

Dr. Sebestyén Béla

Számítógép,
irányítási
mérő,
rendszerek

Műszaki Könyvkiadó

Dr. Sebestyén Béla

Számítógép-irányítású mérőrendszerek

Dr. Sebestyén Béla

Számítógép-irányítású mérőrendszerek

Műszaki Könyvkiadó, Budapest, 1976

Lektorálta:

Dr. Bohus Miklós

okl. villamosmérnök

© Dr. Sebestyén Béla, 1976

ETO: 681.2

681.31

ISBN: 963 10 1463 0

Felelős kiadó: Solt Sándor igazgató

Felelős szerkesztő: Kemény Tamásné okl. villamosmérnök

Tartalomjegyzék

Előszó

1. Mérőrendszer és számítógépes irányítása — Mérőhálózatok	14
1.1. Mérőrendszerek	14
Mérőkészülékek, ellenőrzőkészülékek, mérő- és ellenőrző rendszerek	14
A mérőrendszer és a tárgykapcsolata	17
Moduláris és integrált mérőrendszer	18
Mérési algoritmus; a mérőrendszer általánosítása	18
Kötött és kötetlen módon programozott rendszer	20
1.2. Számítógép-irányítású mérőrendszer	21
A rendszerek osztályozása	21
A kisszámítógép és alternatívái	22
A számítógép-irányítású mérőrendszer sajátosságai	25
1.3. Mérőhálózatok	26
A hálózat felépítése	26
A gépcsatlakozó	28
A tárgycsatlakozó	29
Analog mérőhálózatok	30
Digitális rendszerek	43
Általános jellegű hálózatok	50
Többmérőhelyes hálózatok	51
2. Mérőhálózati alapelemek	57
2.1. Analog építőelemek	57
Műveleti erősítő	57
Mérőerősítő	60
Szigetelt bemenetű jelerősítő	62
Szaggató-stabilizált erősítő	62
Varaktorhíd bemenetű erősítő	63
Töltésérzékeny erősítő	64
Tároló mintavevők	64
Csúcsdetektorok	66
Komparátorok	67
2.2. D—A átalakítók	70
Funkcionális jellemzés	70

Kódtípusok	73
A D—A átalakítók osztályai	74
Párhuzamos közvetlen D—A átalakítók	74
Soros közvetlen D—A átalakítók	77
Közvetett D—A átalakítók	78
Szorzó D—A átalakítók és nem lineáris átalakítók	78
2.3. A—D átalakítók	79
Funkcionális jellemzés	79
Kvantálás és kódolás	81
Az A—D átalakítók osztályai	82
Paralel kvantálású párhuzamos A—D átalakítók	83
Szervo rendszerű átalakító	84
A—D átalakító szukcesszív approximációval	84
Közvetett A—D átalakító lineáris időkonverzióval	85
Két- és háromlejtős A—D átalakító	86
Bipoláris feszültség—frekvencia átalakító	87
Logaritmikus A—D átalakító	88
2.4. Kapcsolók és kapcsolórendszerek	89
Analóg kapcsolók	89
Analóg kapcsolók alkalmazása	92
Kapcsolótömbök	93
Analóg multiplexerek	94
Digitális multiplexerek	97
Pneumatikus multiplexerek	98
3. A rendszerorientált számítógép — Szerkezet és architektúra	
3.1. A szerkezet	102
3.1.1. A ROV struktúra	102
Építőelemek, hálózatok, műveletek	102
ROV-szintű adathálózatok	103
Vezérlőhálózatok és vezérlők	104
Mikroprogramozott vezérlők	106
Adat- és vezérlővonalak	107
Szolgáltatkozó vonalak és hálózataik	108
3.1.2. Memóriák	109
A főmemória	109
Közbenső tárolók	112
Letároló	113
Töltőmemóriák	114
Regisztértömbök	115
Gyors kisegítőmemóriák	116
3.1.3. A központi processzor	116
Regiszterkészletek	116
Operatív tulajdonságok	119
Végrehajtási szintek	121
Felépítésmód	122
Kommunikációs kapcsolatok	124
Mikroprocesszorok	124
Modularitás	125
3.1.4. Készülékvezérlők, periférikus processzorok és periférikus készülékek	126
A készülékvezérlő	126
Csatornaprocesszorok	126

Társoperátorok és társprocesszorok	126
Periferikus készülékek	127
3.1.5. Az MPK struktúra	128
A ROV és MPK struktúra kapcsolata	128
Adat- és vezérlőhálózatok	128
Az MPK hálózat elemeinek kapcsolatai	131
Hálózati konfliktusok, prioritási hierarchia	132
Moduláris MPK hálózatok	134
3.2. Az architektúra	137
3.2.1. A gépek osztályozása	137
Közös jellemzők	137
Jellegzetes kisgép-architektúrák	138
3.2.2. Utasításszinten programozott gépek architektúrája	140
Szószervezésű gépek	140
Byte-szervezésű gépek	143
Byte-orientált szószervezésű gépek	143
Vegyesszervezésű gépek	145
Virtuális memória	145
Szubrutinkezelés	147
4. Kapcsolat- és csatlakozásmódok	152
4.1. Számítógép—készülék kapcsolat	152
4.1.1. Csatornarendszerek	152
Átvitel mód és útvonal	152
Összeköttetési módok	152
Csatornák	153
4.1.2. A programozott csatorna	153
A csatorna szerkezete	153
A csatornát működtető utasítások	155
Az adatátvitel változatai	155
A megszakítórendszer	156
Szinkron, aszinkron, félszinkron kapcsolat	159
4.1.3. A programfüggetlen csatorna	161
Csatorna-útvonalak	161
Működésbeli sajátosságok	162
Átvitel módok	162
A csatorna felépítése	163
A munkamenet	165
4.1.4. Az autonóm csatornarendszer	166
Az autonóm csatorna felépítése	166
Üzem módok	167
Vezérlőutasítások és állapotvektorok	167
A csatorna munkamenete	168
Szolgáltatás-kérés és megszakítás-kérés	168
4.2. Csatlakozás (Interface)	169
4.2.1. A mérőhálózat elemeinek a géphez és egymáshoz való csatlakoztatása	169
A gépcsatlakozás problematikája	169
A mérőrendszer elemei közötti csatlakozás	170
A mérőrendszerben alkalmazott csatlakozást determináló tényezők	171
Egységes csatlakozási rendszerek	172
4.2.2. A HP—IEC csatlakozás	175
A sín	175

Byte-átvitel	175
Rendszerállapotok; szervezés mód	176
Parancsok	177
Szolgáltatkozás és kezelése	178
A készülék kezelésmódjának megválasztása	178
Szerkezeti paraméterek	178
Alkalmazási szempontok	178
4.2.3. A CAMAC csatlakozás	179
A rendszer felépítése	179
Az adatpálya	180
Adatpálya-parancsok	182
Parancsok végrehajtásának időzítése az adatpályán	183
A figyelemkérés (LAM) kezelése	184
Állapotvizsgálat	185
Blokkátvitel	185
Az ágfővonal	185
Ágparancsok	186
Ágfővonal szolgálatkérés	187
Blokkátvitel az ágon	187
Soros fővonal	187
A fővonalak megvalósításának módjai és lehetőségei	188
CAMAC—számítógép kapcsolat	189
5. A mérőhálózat gépi irányítása	192
5.1. A programirányíthatóság problematikája	192
Analog struktúrájú mérőkészülékek jelvezérlése	192
Közvetlen digitális vezérlésű készülékek	193
Jelvezérelhetőség programirányíthatóság, rendszerkompatibilitás	194
A teljes hálózat programirányítása	195
5.2. Mérőkészülék-elemek és mérőkészülékek programozása	196
Programozható ellenállások és potenciométerek	196
Ellenálláshíd programvezérlése	199
Programozható erősítők	202
Tápforrások programvezérlése	204
Impulzusgenerátor programozása	207
Programozható szógenerátorok	210
Programozható frekvenciaosztók, -szintetizálók és -sokszorozók	214
Digitális jelszintetizálók	216
5.3. A mérőkészülék illesztése a rendszerhez	218
A készülékvezérlő	218
Az IEC csatlakozóegység	220
Az IEC csatlakozás egy alkalmazása: DVM rendszer illesztése	221
5.4. A mérőhálózat lehetőségeinek kiterjesztése	223
Automatikus tartományváltás	223
A mérőhálózat memóriájának bővítése	227
Mikroprogramozott vezérlés alkalmazása	229
Mikroprocesszor alkalmazása	230
6. Programrendszerek	233
6.1. Kiszámítógépek programrendszerei	233
6.1.1. Nyelvek és implementációik	233

Assembly nyelvek	233
Felsőbb szintű általános nyelvek	234
Alkalmazásorientált nyelvek	235
6.1.2. Rendszerprogramok	235
Assemblerek és makroprocesszorok	235
Compilerek	236
Kereszt-assemblerek és kereszt-compilerok	236
Töltők	237
Interpreterek	237
Szubrutinok	238
Forrásszerkesztő	238
Hibátlanító programok	239
File-kezelő rutinok	239
Egzekutív program	239
6.1.3. Működtetési módok	240
Manuálisan működtetett rendszerek	240
Készülékegzekutív	241
Diszk operációs rendszerek	241
Többprogramú operációs rendszerek	242
Időreális operációs rendszerek	244
Kiegészítés	245
6.2. A mérőrendszer programellátása	245
6.2.1. A programozási nyelv a mérés technikában	245
Helyzetáttekintés	245
Nyelvi szint és feladatapertúra	246
A nyelv megválasztásának és értékelésének szempontjai	247
A nyelvi probléma megoldásának különböző irányzatai	248
6.2.2. Alkalmazott nyelvek	249
ATLAS	249
ATS BASIC	256
A HP ATS BASIC kiterjesztése digitális vizsgálatokra	261
ATLAS és BASIC származékok, dialektusok, adaptációk	265
Munkaleíró nyelvek	267
Egyéb nyelvek	267
6.2.3. A mérőrendszer működtetése	268
Nyelvi szintek	268
Programozási sémák	269
Időreális diszk operációs rendszer	272
Interpretív operációs rendszer	273
7. A könyvben alkalmazott szakkifejezések szótára	278
8. A szövegben és az ábrákon alkalmazott rövidítések	282

Nap mint nap tanúi lehetünk a számítógép térhódításának. Az elmúlt évtized egyik legfontosabb eredménye a gép rendszerkapcsolatú alkalmazása. A rendszerkapcsolatú gép a feldolgozandó információt nem az embertől, hanem valamely (általában mérő) rendszertől kapja, s tennivalója bizonyos sajátos feladatokra (pl. vezérlés, szabályozás, jelalak-szintézis stb.) is kiterjed. E területnek számos önállóan fejlődő — bár egymással kölcsönhatásban is levő — ága van, hogy csak a természettudományos kutatást, a laboratóriumi technikát, az orvostudományt és biológiát, továbbá az ipari mérés technikát említsük. A könyv az ipari mérések két nagy területe, a folyamatirányítás, illetve a termékellenőrzés közül az utóbbival foglalkozik.

A termékellenőrzést a számítógép kevésbé látványosan hódította meg, mint a folyamatirányítást. Számítógépet — ellenőrző mérésekhez — hosszabb időn keresztül csak a haditechnikában, az űrkutatásban és a repülőgépiparban alkalmaztak. A mindennapok praxisában nemigen, ami a gépek kezdeti viszonylag magas árában, s az ipar igényeinek késlekedésében lelheti magyarázatát. Az ipari ellenőrzés szférája csak az árak lényeges csökkenésével s a mérés technika lehetőségeinek egyidejű bővülésével vált a gépek számára befogadóképesé. Az áttörés valamikor a 70-es évek fordulóján történt meg. Az a körülmény, hogy a kisgépek alapkonfigurációinak világszertei árnyója tíz esztendő alatt mintegy a tizedére esett (vegyük figyelembe a fizetési eszközök időközben végbement értékromlását is), lehetővé teszi, hogy mára a számítógépet az átlagos mérőkészülék árszínvonalán szerezhessék be. Ez a lehetőség a termékellenőrzés automatizálásának új távlatait tárta fel. A következményképpen jelenleg zajló csendes forradalom — mert általános érvényű, s nem csak az ipar egyes ágait érinti — jelentőségét tekintve semmiképp se marad el az ipari automatizálás más szektoraitól. A minőségellenőrzésen kívül a termékre közvetlenül irányuló technológiai jellegű mérések, a fejlesztő laboratórium új termék létrehozásával kapcsolatos mérései, a típusvizsgálat, majd a termék használatba vétele utáni karbantartási vizsgálatok, illetve üzemközbeni folytonos ellenőrző mérések mind-mind ebbe a témakörbe tartoznak. Néhány esztendeje még könnyen felsorolhattuk azokat az iparágakat, melyek számítógépes ellenőrzés technikát alkalmaznak. Ma egyszerűbb volna a felsorolást kizáró értelemben elvégezni.

A termékellenőrzés automatizálásának máris igen kiterjedt területe irodalmi szempontból legalábbis három irányból közelíthető meg: a termékellenőrzés, a mérés technika vagy a számítógép oldaláról. Mindhárom megközelítésmód azonban más-más tematikai töltést sugall. A termékellenőrzés oldaláról kiindulva elsősorban a gazdaságosság és szükségszerűség kérdését kell elemezni, megfigyelés alá vonva az

összes — manuálistól az automatizált módszerekig terjedő — lehetőségeket. Kimutatni, hol és miért, melyik eljárást célszerű alkalmazni. Ilyenszerű vizsgálatok és következtetések azonban sokkal inkább egy-egy speciális területre irányítva, semmint általánosságban végezhetők. Ugyanígy az általánostól elfordul és a speciális felé irányul a mérés technikai megközelítés is. A szerző a harmadik utat választotta, annál inkább, mert vizsgálatait a kiszámítógép alkalmazási lehetőségeinek oldaláról indította, s ezen az úton bukkant a termékellenőrzésre mint a kiszámítógépek hazai alkalmazhatósága szempontjából igen perspektivikus területre. Figyelme elsősorban azon általános, strukturális, szervezési és működésbeli összefüggések felé irányult, melyek ismerete a számítógépes ellenőrző rendszerek létrehozása kapcsán mindenképp szükséges. Hogy hol és miért célszerű számítógépet alkalmazni? — ez a kérdés a könyv látóhatárán túl fekszik. A tematika szűkítése érdekében a rendelkezésre álló irányító eszközök közül az anyag egyedül a kiszámítógépre koncentrált; ennek alternatíváit (asztali kalkulátor, kontroller) csak szószavúán érinti. A mikrogép a kisgép fejlődési vonalán, annak egyenes folytatásaként, lényegében a könyv kialakulásának időszakában jelentkezett. Alkalmazástechnikája a termékellenőrzés terén egyelőre inkább ígéret, mintsem realitás.

A különböző korlátozási tényezők által körülhatárolt anyag hat meglehetősen önálló fejezetben öltött formát. Az érintett témakörök összefoglaló feldolgozása — talán a 2. fejezet anyagának s a 4. fejezet egyes részeinek kivételével — a magyar irodalomból mindeddig hiányzott. Mindemellett az író szorongva gondol arra a hatalmas ismeretanyagra, mely a tématerületen az elmúlt évek során felhalmozódott, s amelynek egy része kényszerűen a könyv keretein kívül rekedt. Ugyanakkor egyes témakörök tárgyalása során éppen a rendelkezésre álló információk egyenetlensége, hézagossága okozott nehézséget.

A mérések automatizálásának, a számítógép általános alkalmazásának lehetőségeit döntő módon befolyásolta a detektálási technika fejlődése, a nemvillamos mennyiséget villamos mennyiségre áttevő érzékelő-átalakítók kialakulása. Ezen átalakítók birtokában nyílt mód arra, hogy a korábban egymástól távoleső mérés technikai területeket közös elvi-metodikai alapokra helyezték. Lehetővé vált, hogy a méréseket a villamos, elsősorban az elektronikus mérés technika eszközeivel végezzék, és hogy a mérések automatikus lebonyolítására a villamos mérés technikában kialakult fejlett módszereket alkalmazzák. A számítógép-irányítású mérőrendszerek struktúrája világosan bizonyítja, hogy az egységesítés irányában milyen jelentős előrelépés történt. Tulajdonképpen az összes számítógépes mérőrendszer egyetlen alapsémára vezethető vissza, bár az egyes alkalmazási területek konkrét igényei a részletek jelentékeny eltéréseiben tükröződnek. A könyv első fejezete a mérőrendszerekkel kapcsolatos alapvető ismereteket összefoglalva, a számítógéppel irányított mérőrendszerek sajátosságait és felépítésük általános vonásait tekinti át, majd a rendszereket mérőhálózataik tulajdonságai alapján osztályozza. Egyes rendszertípusok szemléltetése konkrét, megvalósított berendezésekre való hivatkozással történik.

A második fejezetben ismertetett mérőhálózati alapelemeknek legalábbis egy részével magyar- és idegennyelvű művek sora foglalkozik. Ami itt mégis eredetinek tekinthető: az anyagnak a mérőrendszer szemszögéből való sajátos csoportosítása és az áramköri részleteket elhanyagoló, a funkcionális sajátosságokra koncentrált előadásmód.

A legnehezebb írói gondot kétségkívül a kiszámítógép feldolgozása jelentette. Bár a kisgépekkel foglalkozó rész egyben a legnagyobb terjedelmű fejezet is, az e téren összegyűlt óriási ismeretanyagnak öt ívben való kivonatolása csak erős tömörítéssel és az olvasó általános számítástechnikai-elektronikai tájékozottságának feltételezésével vált lehetségessé. E helyzet ellenére — az összefüggések megőrzése érdekében — néhány olyan részlet áttekintésére is sor került, ami közismertnek tűnhet.

Külföldi vagy hazai kisgépek részletes leírása, specifikációs adatainak megadása, eleve nem lehetett cél. A szerző azokat az elveket, struktúra és architektúra vonásokat igyekezett rendszerezni, melyek a kisgépek világában meghatározó jelentőségűek, s konkrét géptípusok megemlézésére csupán mint egyik vagy másik tulajdonság jellegzetes képviselőjére került sor. Bár a figyelem az ismertebb gépek jelentékeny hányadára kiterjedt, arra természetesen nem volt lehetőség, hogy a becslés szerint legalább kétszáz különböző géptípus mindegyikére kiterjedjen. Ezért a szerző nyitva hagyja megállapításait és általánosításait, hallgatólagosan mindegyikhez hozzáfűzve: „az általam ismert esetekben”.

A mérőrendszer szervezése (szervezhetősége) szempontjából igen lényeges kapcsolati-csatlakozási kérdéseket tárgyaló 4. fejezetből hiányzik a kisgépek konkrét csatlakozásmódjainak leírása. Ennek áttekintését azonban — mivel az legalább akkora teret igényelt volna, mint a fejezet megmaradó része — mellőzni kellett. Az egységes csatlakozási rendszerek (HP—IEC; CAMAC) ismertetése során elsődlegesen a logikai összefüggések megvilágítása, a működési filozófiák megértetése volt a cél. A pontos, formális leírások listája az irodalomjegyzékben az olvasó rendelkezésére áll.

A klasszikus szerkezetű mérőkészülék analóg jellegű beavatkozási és a gép közti D — A átmenetet kell beiktatni. Kialakulóban vannak azonban digitálisan közvetlenül vezérelhető mérőkészülékek is, melyek a gép vezérlőjeleit közvetlenül értelmezik. Ezek közül ismertet néhányat az 5. fejezet, elsősorban a jelvezérelhetőség, programirányíthatóság lehetőségeinek érzékeltetésére, egyszersmind azonban felhívja a figyelmet arra az új szerepkörre, melyet a számítógép mint a generált jelalak közvetlen meghatározója a jövőben betölthet. Perspektivikusan igen nagy jelentőségűek a mérőhálózat intelligenciafokának emelésére, az elosztott intelligenciájú mérőhálózatok létrehozására irányuló törekvések (5. 4.) E törekvéseket a könnyen hozzáférhető memóriaelemek, a mikroszámítógépnek kártya szinten való elérhetősége, valamint az áramköri-tok méretű „számítógép” közeli lehetősége teljes mértékben indokolja. A jelenlegi gépcentrikus mérőrendszerek helyét minden bizonnyal az elosztott vezérlésű moduláris vagy hierarchikus szervezési formák veszik majd át éppúgy, mint ahogy a processzor-centrikus számítógép is lassanként a modulokból álló hálózatnak ad helyet. A máris megragadható különböző lehetőségekre a fejezet, ha csak röviden is, ugyancsak kitér.

A programozás problematikáját felölelő zárófejezet első — előkészítő — része a kisgépek nyelvi és programozási sajátosságainak áttekintésével nem csupán a hazai szakirodalom egy hiányát kívánja pótolni. A mérőrendszer tervezőjének világosan látnia kell, mik azok a programozási eszközök, melyek a kisgéppel együtt általában rendelkezésre állnak, s mi az, ami ebből mérés-technikai célra felhasználható. Ez utóbbi kérdés már átvezet a fejezet második, speciálisabb részébe, mely igyekszik összefoglalni a mérőrendszer programozásának nyelvi, szervezési kérdéseit. E második rész belső arányait talán leginkább befolyásolja a szerző rendelkezésére álló információs anyag egyenetlensége. Az ATS BASIC ismertetése azonban pl. nem ezért kapott viszonylag nagyobb teret. A szerző úgy véli, hogy a mérés-technikai programozási példák megértéséhez mindenképpen szükséges az alapnyelv ismerete. Egyúttal azonban nem várható el, hogy egy adott nyelvet minden olvasó ismerjen, vagy hogy a programozási példák megértése érdekében viszonylag terjedelmes előtanulmányokat folytasson. Bizonyos meglehetősen fontos kérdések (pl. CAMAC nyelvek) taglalása ez esetben is kényszerűen elmaradt.

A szerzőnek nem állt szándékában, hogy a szöveget a gyorsan változó s esetleg már a könyv kihozatalának időpontjára elavuló (specifikációs) adatokkal terhelje meg. Tervbe volt véve azonban egy hetedik fejezet összeállítása, mely az első hat anyagának szintézisét, az egyes fejezetek ismereteinek a mérőrendszer szempontjai

szerinti értékelését adhatta volna. Sajnos e szándék megvalósítására már nem kerülhetett sor.

A szövegben ritkán fordulnak elő irodalmi utalások, bár a könyv — bővebb hazai tapasztalatok híján — nagymértékben támaszkodik az (idegennyelvű) irodalomra. A források anyaga azonban gyakran egymással és a szerző saját eredményeivel igen szorosán ötvöződik, ami helyenként megnehezítette volna az irodalmi hivatkozást. Áthidaló megoldásként: az egyes fejezetek végéhez csatolt irodalomjegyzék tételeinek sorrendje, ahol lehet, nagyjából követi a szöveg menetét.

A számítástechnika nyelvi szempontból igen nagy nehézségekkel küzd. A határainkon kívül megszülető fogalmak és szakkifejezések tömege zúdul ránk, s ezek adaptációja mindenképpen hatalmas terhet jelent. A fő problémát mégis talán az okozza, hogy a nyelvi adaptáció mechanizmusa jelenleg igen csikorogva működik. A helyzetet valamiféle aggályoskodás és határozatlanság jellemzi, meg az alapelvek és tennivalók tisztázatlansága. Ma aligha merné valaki is az általános felhördülés kockázata nélkül pl. a „Bolzen” szót természetes közvetlenséggel „bolcni”-ra magyarosítani. Következésképpen a terület szakembere időnként már-már azzal a képzettel küzd, hogy valamely távoli és egykori gyarmat kikötőjének keverék nyelvét hallja. Az eredeti idegen szakkifejezések használatának kétségkívül vannak objektív okai. A szakember e kifejezésekkel irodalmi tanulmányai közben ismerkedik meg. Az egyazon irodalmi töről táplálkozó réteg az eredeti kifejezések alapján könnyen azonosítja a fogalmakat. Ha ugyanazt a fogalmat valaki egy, még általánosan el nem terjedt magyar szóval jelöli meg — nem értik, és gyakran ismét csak utalnia kell az eredeti kifejezésre. Közrejátszhat itt azonban a közgondolkodásban jelentkező valamelyes nyelvi sznobság és a szakmai misztifikáció szándéka is. Másképp hogyan fordulhatna elő, ami pl. egy nemrég megtartott hazai előadáson, hogy az előadó olyan tösgyökeres magyar szó helyett, mint a „kiégni” — „burn out”-ot mondott, a „nyitott” helyett „open”-t használt.

A minden áron való magyarosítás természetesen értelmetlen, már azért is, mert feleslegesen terheli a magyar szókincset, és egyúttal elszakadást is jelent a fogalmakat teremtő közegtől. Az erőszakolt magyarítás főleg gyakran olyan szerencsétlen és nehézkes kifejezéseket eredményez, melyek idegenebbül hangzanak az idegennél. Egyáltalán, az idegen szó nem minden esetben zavaró és nem azáltal zavaró, mert egy más nemzet szókincséhez tartozik. Zavar akkor, ha nem illeszkedik a magyar nyelv hangzásvilágához; groteszkül vagy egyáltalán nem ragozható. A magyarból és adaptálatlan idegenből álló kotyvalék nyelv alkalmatlan a gondolat megfelelő tisztaságú megfogalmazására és kifejezésére.

A szerző a könyv megírása során arra törekedett, hogy a fogalmak megjelölésére magyar kifejezéseket vagy csak olyan idegen szavakat használjon, melyeket a magyar már adaptált vagy adaptálni képes. Hogy a tisztázatlan szóhasználatból származó zavarokat elkerülhesse, az általa javasolt kifejezések szótárát a könyvhöz csatolva mellékeli.

A szótár mellett a könyv végén a szövegben és ábrákon alkalmazott rövidítések gyűjteménye található.

A szerző köszönetet mond mindazoknak, akik könyve megírásában bátorították. Köszönetet mond a könyv lektorának Bohus Miklós docensnek, továbbá Erényi István tudományos munkatársnak és Lukács József tudományos főmunkatársnak, akik a szöveg egyes részeinek átolvasása során hasznos tanácsaikkal segítettek; végül a kézirat gépelésének fárasztó munkájáért feleségének, Juditnak.

Budapest, 1975. október 19.

1.

Mérőrendszer és számítógépes irányítása — Mérőhálózatok

1.1. Mérőrendszerek

Mérőkészülékek, ellenőrzőkészülékek, mérő- és ellenőrző rendszerek

A műszaki méréstechnika két alapvető eszköze: az adatközlő mérőkészülék és az ellenőrző készülék. Az előbbi a vizsgált mennyiség számszerű értékét adja, az utóbbi pedig arra szolgál, hogy megállapítsa, vajon a vizsgált mennyiség nagysága valamely adott értékkel vagy értékekkel összehasonlítva megfelel-e az előírásnak, s mint ilyen, jó/nem jó eredményt ad.

Az *adatközlő mérőkészülék* funkcionális szempontból: érzékelő, átalakító, értékképző és adatközlő elemekből áll. A funkcionális elemek nem mindig választhatók el egymástól szerkezetileg. Az érzékelő pl. egységet alkothat az átalakítóval, az értékképző az adatközlővel. Más esetben, az egymástól távol fekvő elemeket adatátviteli vonal köti össze.

Az adatközlő mérőkészülék működésének alapja a közte és a mérés tárgya között létrejövő valamely olyan kölcsönhatás, melyben a mérendő mennyiség szerepet játszik. A mérendő mennyiségre vonatkozó kvantitatív információt a kölcsönhatásban létrejövő jel viszi át a mérőkészülékhez. A jel, mely maga is fizikai mennyiség, nem információ, az információnak csupán anyagi hordozója. A jel fizikai természetét illetően lehet a mérendő mennyiséggel azonos nemű, de lehet — és a gyakorlatban ez a jellegzetesebb — attól eltérő is. A jelképzés általánosságban egy mennyiségnek egy más mennyiségre való leképezése, a két mennyiség kvantitatív attribútumai között fennálló kapcsolat alapján. A leképezést végző elemet átalakítónak nevezik, a mérés tárgyával közvetlen kölcsönhatásban álló elemet érzékelőnek. Az átalakító kimenő jelének valamely paramétere a mérendő mennyiség kvantitatív képviselője. A mennyiség és képviselője közötti reláció a

$$z = f(x)$$

függvénykapcsolattal jellemezhető, melyben x a bemenő érték, z pedig annak átalakítás utáni képviselője. A lineáris átalakítók lineáris kapcsolatot realizálnak. A másféle kapcsolatot (kvadrátikus, logaritmikus stb.) követő átalakítókat függvényátalakítóknak nevezik.

A mérendő mennyiség lehet analóg (A) vagy lehet diszkrét természetű (D). Az analóg mennyiség a vizsgált tartományon belül bármely értéket felvehet, szemben a diszkrét természetű mennyiséggel, mely csak meghatározott értékszinteket foglalhat el. A jel több különböző mennyiségre, analóg és diszkrét mennyiségekre vonatkozó információt egyidejűleg is közvetíthet. Az energiaérzékeny nukleáris detektor kimenő impulzusai pl. a folytonos értéktartományra eloszló impulzusamplitúdóban a részecskék energiájára vonatkozó analóg információt hordoznak, ugyanakkor az impulzusok időegységre eső száma a sugárzás intenzitásának mértéke.

A leképzés módjától függően, A—A, D—D, A—D és D—A átalakító különböztethető meg (az első betű az átalakító bemenő, a második kimenő mennyiségének természetére utal).

Azon A — A átalakítókat, melyek bemenő értékének és kimenő értékének egymástól eltérő spektruma van, külön névvel, modulátoroknak nevezik.

Az érzékelő-átalakító egység kimenő jele és az általa érzékelt mennyiség közötti kapcsolat gyakran nem közvetlen, hanem közvetett; az információ egymás után többször is hordozót cserél, míg végül a kimenő jelre ültetődik (az ampermérőnél pl. az áramerősség először forgatónyomatékká transzformálódik, majd szögelfordulássá, ill. körívvé). Az érzékelő-átalakító jele — a mérőkészülék felépítésétől függően — további átalakítókra kerülhet, közben folytonosan változtatva megnyilvánulási formáját, megőrizve azonban az érzékelt mennyiség nagyságára vonatkozó információ-tartalmát. Az átalakítók (átalakítók lánc) végül a jelet a mérőkészülék további elemei számára olyan alakban állítják elő, melyről a mért mennyiség nagyságára vonatkozó információ az ember vagy gép számára értelmezhető módon leválasztható.

Közvetlen, ill. szintkiegyenlítéssel átalakító mérőkészülék ismeretes. A közvetlen átalakításmódra az átalakítók kaszkád kapcsolása jellemző; az érzékelő-átalakító kivételével minden egyes átalakító bemenő jelül a sorban előtte álló kimenő jele szolgál. A szintkiegyenlítő (vagy más néven kompenzációs) átalakításmódot alkalmazó mérőeszköz direkt átalakítólánccból és inverz átalakítólánccból épül fel. Az utóbbi az átalakítólánc kimenő mennyiségét a kiindulási mennyiségre transzformálja, és a bemenetre csatolja vissza. A visszatranszformált érték a bemenő értékből levonódik úgy, hogy a direkt átalakítóláncre csak a kettő különbsége jut.

Bizonyos átalakítókat egynél több bemenet, azaz

$$z = f(x_1, x_2, x_3, \dots, x_n)$$

függvénykapcsolat jellemez. Az ún. operatív átalakítóknak pedig az a sajátosága, hogy kimenő értékük nem csupán a bemenő értéknek, hanem az időnek is függvénye (integráló, differenciáló átalakítók).

A mérőkészülék érzékelő + átalakító szakaszának kimenetén kapott jel valamely jellemzője kvantitatíve a mért mennyiséget tükrözi. A jellemző számszerű értékét a mérőkészülék értékképző eleme állapítja meg, a készülékbe épített etalonnal való összevetés alapján, vagy előre hitelesített skálára való rávetítés útján. Mindkét eljárás a mennyiség értékét bizonyos mértékegységben határozza meg.

A mért mennyiség számszerű értékét a mérőkészülék adatmegjelenítő eleme közli; az ember (vagy gép) számára értelmezhető módon. Az adatmegjelenítés történhet analóg alakban (l. mutatós műszerek, görberajzolók, analóg oszcilloszkópok stb.) vagy digitális alakban (digitális kijelzők, nyomtatók, digitális rajzolók, KS-csöves adatközlők).

Az ellenőrző készülék, eltérően az adatközlő mérőkészüléktől, nem a vizsgált mennyiség számszerű értékét adja, hanem a mennyiség nagyságának ugyanazon mennyiség egy vagy több előre meghatározott értékszintjéhez való viszonyítására és a viszonyítás eredményének közlésére szolgál; tehát nem mérőszámot, hanem (valamely mennyiségi adatot vagy adatokat magában záró kérdést eldöntő) igen/nem választ nyújt. A kérdések ilyen természetűek: a mennyiség értéke azonos-e a megadottal; a mennyiség értéke meghatározott szint alatt vagy felett van-e; két meghatározott szint közé esik-e; egy adott szinttől való \pm irányú relatív eltérése bizonyos százalékos értéket meghalad-e, vagy sem. Igen vagy nem? Az ellenőrző készülék éppúgy rendelkezik érzékelő- és átalakító szervekkel, mint az adatközlő mérőkészülék (értékképző és megjelenítő elemmel csak akkor, ha egyúttal adatközlő szerepkört is el kell, hogy lásson). Az ellenőrző készülék jellegzetes elemei: az érték-össz-

szehasonlító és az igen/nem (jó/nem jó) vizsgálat kimenetéről tájékoztató eredményközlő.

Az érzékelő primer mérőjelének képzéséhez szükséges energiát vagy a vizsgált objektum, vagy a mérőkészülék szolgáltatja. *Passzív érzékelésről* beszélünk, ha a jelképzés során az érzékelő energiát vesz fel, és *aktív érzékelésről*, ha energiát ad le. A passzív, ill. aktív érzékelés példaként megemlítjük a hőmérséklet termoelemmel, ill. ellenálláshőmérővel való mérését. Aktív érzékelés során a mérőkészülék az érzékelő-átalakítóhoz kérdőjelet küld; a mérőjel a kérdőjelre adott válaszként adódik.

Az aktív érzékelés előfeltételeként a mérőkészülék belső kérdőjel-forrással kell, hogy rendelkezzen.

Az aktív érzékeléstől meg kell különböztetni az *aktív mérést*, melynek célja, hogy megállapítsa a mérés tárgyának valamely meghatározott külső ingerre adott válaszát. Az aktív mérés példaként az átviteli tulajdonságok vizsgálatát vagy pl. a digitális áramkörök logikai vizsgálatát említhetjük meg.

Az aktív méréshez külön jelgenerátor vagy jelgenerátorok alkalmazása szükséges. A jelkeltés villamos úton történik, de ha a feladat megkívánja, a villamos jelet más természetű fizikai jellé transzformáló (inverz) átalakítókat, esetleg beavatkozókat is alkalmaznak.

Bizonyos esetekben a mérés objektumának tulajdonságait számos munkapontban kell megvizsgálni, ill. ellenőrizni (pl. félvezető eszközök jelleggörbéit). Az ilyen változtatható munkapontú objektumok a munkapont állítására szolgáló bemenettel is rendelkeznek. A munkapont beállítására szolgáló jelet alapjelnek, az alapjelet adó forrást alapjel-generátornak nevezzük. Az alapjel-generátor és a mérés tárgya között — szükség szerint — ugyancsak átalakítók, ill. beavatkozók helyezkedhetnek el. Az alapjel esetenként nem közvetlenül, hanem a környezet mint átvivőszerv befolyásolása révén hat a mérés tárgyára. A tárgy ellátására szolgáló tápforrások, bizonyos általánosítással, ugyancsak az alapjel-generátorokhoz sorolhatók.

A technológiai folyamat (akár üzemi, akár laboratóriumi), művelet és mérés zárt visszacsatolóláncként szemlélhető. Hagyományos módon a visszacsatoló lánc az emberen keresztül záródik. Mind több olyan technológiai rendszert is alkalmaznak azonban, melyet a mérőkészülék kimenetének a műveletre való közvetlen visszahatása, tehát a termék paramétereinek előírt érték alapján végzett zártláncú szabályozása jellemez, amikor is a mérőkészülék kimenetét és a mérési objektumot szabályozási szakasz kell, hogy összekapcsolja. A szabályozási szakasz jelenléte révén másrészt bizonyos mérési paraméterek állandó értéken tartása is lehetővé válik.

A műszaki méréstechnika — gyakran összetett feladatkörű — passzív és aktív mérőkészülékei, ill. mérőberendezései (különböző paramétermérők, ill. ellenőrzők, jelvizsgálók, jelleggörbe- és átviteligörbe-regisztrálók, funkcionális és diagnosztikai vizsgálóeszközök stb.) végső soron mind az adatközlő mérőkészülék és ellenőrző készülék elvén épülnek fel. A mérésre, ill. ellenőrzésre szolgáló elemek — szükség szerint — jelgenerátorokkal, valamint speciális kiegészítő elemekkel bővíthetnek ki; átalakító, értékképző, értékösszehasonlító és megjelenítő szerveik nem ritkán adatfeldolgozó rendszer bonyolultságúak. Változatskálájuk a kézi irányítású készüléktől az automatikusan működő berendezésig, az indikátortól a mérőrendszerig terjed.

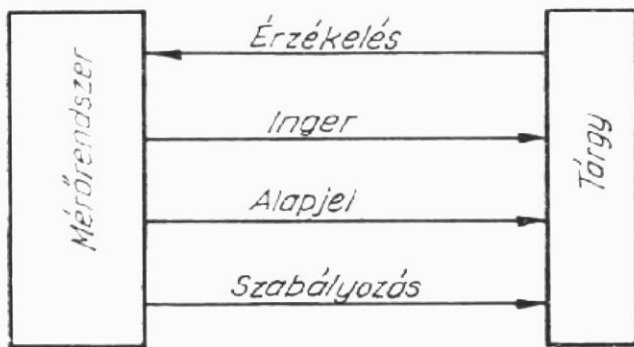
Rendszernek általános megfogalmazás szerint olyan elemek együttesét nevezik, melyeket legalább egy reláció egybefűz. *Mérőrendszeren* adatközlő mérőkészülékekből, ellenőrző készülékekből és jelgenerátorokból, valamint kiegészítő elemekből kiválasztott olyan együttest értünk, mely pontosan körülírt mérési feladatok ellátására szolgál. A mérőrendszer elnevezést célszerűségi okokból a továbbiakban a tulajdonképpeni mérőrendszer villamos részére alkalmazzuk; a mérés tárgyával közvetlen érintkezésben levő nemvillamos bemenetű mérő, ill. nemvillamos kimenetű hatásáta-

dó átalakítókat (ha ilyenek jelen vannak) pedig mint kihelyezett egységeket, a (villamos) mérőrendszer és a tárgy csatlakoztató (interface) elemeiként fogjuk fel. Az egyszerű mérő/ellenőrző készülékkel szemben, a mérőrendszert felépítésének, feladatkörének, irányításának, külső kapcsolatainak, adatfeldolgozási problematikájának sokrétűsége különbözteti meg.

Az ellenőrző készülék analógiájára a jó/nem jó választ adó mérőrendszert speciális elnevezéssel *ellenőrző rendszernek* (test system) is nevezik.

A mérőrendszer és a tárgy kapcsolata

A mérőrendszert a vizsgálat objektumával (melyet a továbbiakban egyszerűen tárgynak nevezünk) az érzékelés, az inger, az alapjel, valamint a szabályozás hatásvonala kötheti össze (1.1.) ábra. A hatásvonalak azonos nevű átviteli csatornák alakjában realizálódnak.



1.1. ábra

A mérőrendszert és a mérés tárgyát összekötő hatásvonalak

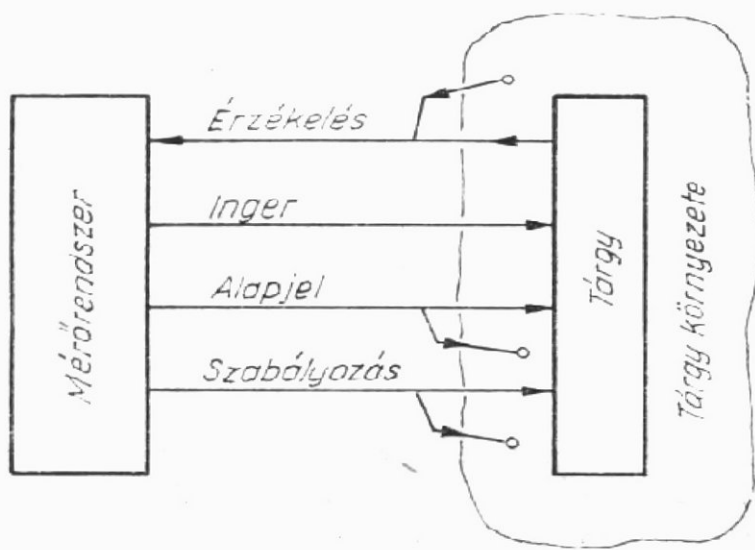
Bár az aktív érzékelés folyamatában a tárgy energiát ad le, a passzív érzékelés folyamatában energiát vesz fel, a rendszer helyes szervezése esetén az érzékelés a tárgy állapotát csupán elhanyagolható mértékben befolyásolja. Az érzékelés mozzanatában a tárgynak a mérőrendszerre gyakorolt hatása a jellemző. Az érzékelés hatásvonalát megtestesítő érzékelő csatornán passzív és aktív érzékelőktől származó mérőjelek haladnak a mérőrendszer irányában, és kérdőjelek haladhatnak a tárgy irányában.

Az inger-, alapjel- és szabályozócsatornákon át a mérőrendszer hat a tárgyra. Esetenként a mérőrendszer — a szabályozás hatásvonalán át — a tárgynak a vizsgálatot megelőző tulajdonságait a mérési algoritmusban szereplő előírásnak megfelelően meg is változtathatja (pl. sávszűrő hangolása).

A lehetséges négy csatornatípus közül minden mérőrendszerben megtalálható az érzékelőcsatorna. A többi csatorna jelen lehet, vagy esetleg hiányozhat is, a tárgy természetétől, az elvégzendő vizsgálat követelményeitől függően.

Egyes mérőrendszerek érzékelőcsatornáin olyan vonalakat is tartalmaznak, melyek segítségével a mérőrendszer az általa kibocsátott ingerjelek vagy alapjelek paramétereit a tárgy bemenetein korrekció céljából ellenőrzi. Erre az esetben lehet szükség, ha feltehető, hogy e jelek az átvivő vonalakon a programozotthoz képest el nem hanyagolható torzulást szenvednek.

A tárgyat és a mérőrendszert összekötő közvetlen hatásvonalak — esetenként — a tárgy környezete és a mérőrendszer közötti hatásvonalakkal egészülhetnek ki (1.2. ábra). Az érzékelőcsatornán át érzékelheti, az alapjel-csatornán át változtathatja, és a szabályozó-csatornán át stabilizálhatja a mérőrendszer pl. a környezet hőmérsékletét.



1.2. ábra.

A mérőrendszer és a tárgy közvetlen és közvetett kapcsolata

A mérőrendszer feladata, mint látható, bár egyes esetekben pusztán adatgyűjtésre korlátozódik, általában azonban ennél lényegesen tágabb körre terjed ki.

Moduláris és integrált mérőrendszer

A mérőrendszer felépítését tekintve lehet moduláris vagy integrált.

A moduláris mérőrendszer valamely tágabb méréstechnikai feladatkör ellátásához szükséges összes elemet építőmodulok alakjában öleli fel. Az aktuális modulok kiválasztása az elemek együtteséből, a kiválasztott modulok egymással, ill. a tárggyal való kapcsolatának megszervezése, esetenként az elvégzendő mérésnek megfelelően történik. A moduláris mérőrendszer további jellemzője még, hogy építőelemeit a rendszerből kiemelve, azok funkcionális egységként, önállóan is alkalmazhatók.

Az integrált mérőrendszer feladatorientált és fordítva, a feladatorientált rendszer általában integrált; benne az egyes funkcionális egységek fizikailag nem különülnek el; önálló egységként nem alkalmazhatók. E tulajdonságai alapján az integrált mérőrendszer bonyolult felépítésű speciális mérőkészülékként is felfogható.

Bizonyos rendszerek az integrált és a moduláris közötti átmenetet képeznek.

Mérési algoritmus: a mérőrendszer általánosítása

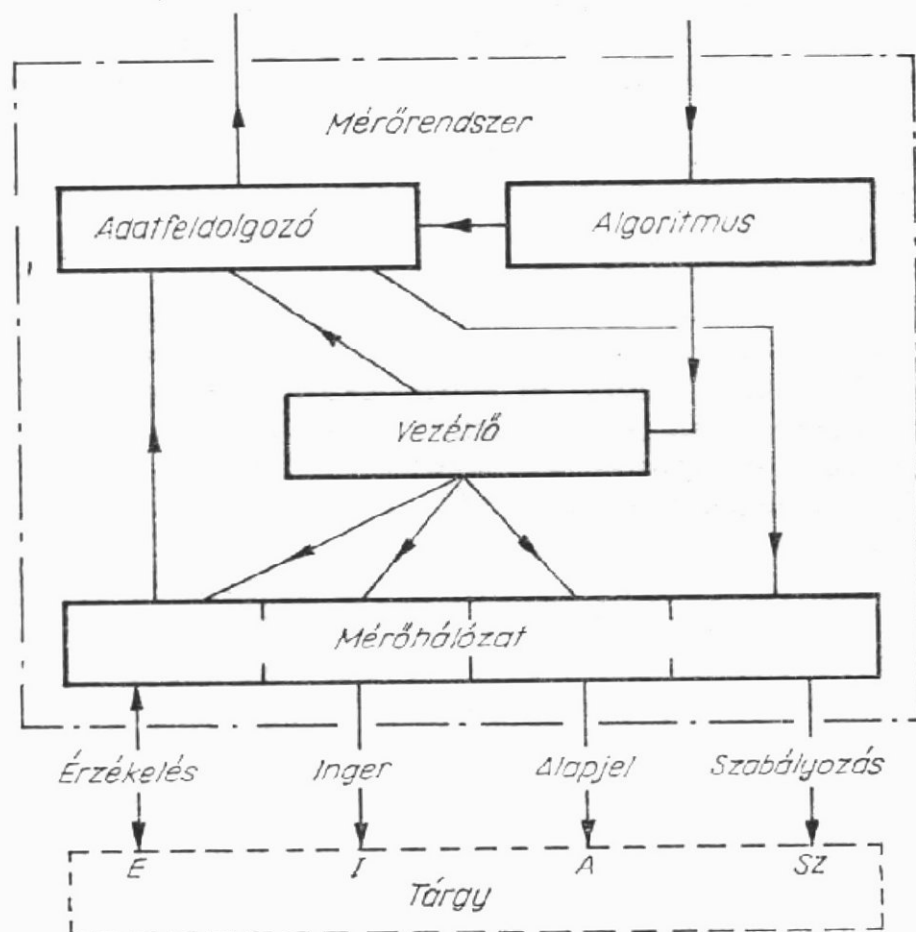
Bármely mérési feladat végrehajtásához mindenekelőtt a feladat pontos megfogalmazása, leírása, majd algoritmizálása szükséges. Az algoritmizálás nem más, mint a feladatnak időben egymást követő műveleti lépések sorára való felbontása. Az algoritmus első változata mindig tárgyorientált, melyet a jelek szintjén fogalmazzunk; a feladat végrehajtásához szükséges mérőrendszert nem vagy csak hozzávetőlegesen vesszük figyelembe. Az algoritmust konkrét mérőrendszerhez közelítve annak mérőrendszer-orientált alakja áll elő, mely az adott mérőrendszerrel már végrehajtható. A mérőrendszer-orientált algoritmus a mérési feladatokat a mérőrendszer alkotó-elemeinek megnevezésével fogalmazza meg. A mérőrendszer-orientált algoritmus a következő elemi feladatokra terjedhet ki:

—A méréshez szükséges készülékek kiválasztása a rendelkezésre álló készülékek halmazából.

- Mérőhálózat kialakítása a kiválasztott készülékeknek egymással és a tárggyal való (a momentán mérési feladatnak megfelelő) összekapcsolása révén.
- A tárgy munkafeltételeinek kialakítása (tápfeszültségek beállítása, terhelő-ellenállások csatlakoztatása stb.).
- Jelgenerátorok jelparamétereinek, a mérő- és ellenőrző készülékek mérismódjának és tartományának beállítása.
- Jelek kibocsátása a tárgyhoz; érzékelt jelek begyűjtése.
- Mérési adatok, ellenőrzési eredmények képzése és megjelenítése.
- Primer mérési adatok rögzítése.
- Primer mérési adatok feldolgozása, kiértékelése, átfogó mérési jellemzők képzése, jegyzőkönyvezése.
- Valamely előírás vagy a begyűjtött adatok alapján a mérési körülmények esetleges korrekciója.

A mérési feladat végrehajtásának folyamatában meg kell különböztetnünk a szorosabb értelemben vett mérés (mely általánosabb értelemben az ingerkeltést is magában foglalja), a munkapont-állítást, szabályozást, adatfeldolgozást és megjelenítést mozzanatait, amelyek a legváltozatosabb módon fűződhetnek egymáshoz. A mérőrendszeren belül az egymással és a tárggyal meghatározott kapcsolatban álló, a mérés végrehajtására szolgáló végrehajtó elemek (mérő- és ellenőrző készülékek, generátorok stb.) együttesét mérőhálózatnak nevezzük. A hagyományos mérőrendszer mérőhálózatának működtetése, mérési algoritmusának lebonyolítása manuálisan történik. Az automatizált rendszerben az irányítás feladatát teljes mértékben az algoritmust ismerő szerkezet látja el. A két határeset között a félautomatizált rendszerek számos változata található.

A mérési algoritmus predeterminált műveleteket ölel fel. Amennyiben tehát az algoritmust ember bonyolítja le, szerepe gépies. A mérőrendszer egységes tárgyalása szempontjából célszerű az algoritmust lebonyolító embert mint rendszervezérlőt és adatfeldolgozót, de magát az algoritmust is a mérőrendszer részeként tekinteni.



1.3. ábra.
A mérőrendszer általános alakja

Ezen általánosító értelmezés alapján a mérőrendszert az alábbi funkcionális egységekre oszthatjuk fel (1.3. ábra):

- mérőhálózat;
- rendszervezérlő;
- algoritmus;
- adatfeldolgozó.

Ezek közül minden mérőrendszerben megtalálható a mérőhálózat, a rendszervezérlő és az algoritmus. Adatfeldolgozó ott szükséges, ahol a mérőhálózat által közvetlenül szolgáltatott adatok szortírozására, tárolására, szintézisére, jegyzőkönyvezésére stb. van szükség. Az adatfeldolgozó tulajdonképpen felfogható átalakító, értékképző, értékösszehasonlító, adat-, ill. eredményközlő elemek valamely összetettebb, magasabb színvonalú reprezentációként is. Másrészt bonyolultsága, sajátos szerepe és számítástechnikai vonatkozásai miatt célszerű külön egységként megjelölni.

Kötött és kötetlen módon programozott rendszer

A mérési algoritmust a rendszervezérlő bonyolítja le. Az algoritmust a rendszervezérlővel, általa értelmezhető nyelven, utasítások, ill. utasításokból felépített program alakjában közlik. A vezérlő a programot a mérőrendszer végrehajtó elemeinek nyelvére fordítja le.

A program és a vezérlő viszonyát illetően két, karakterisztikusan eltérő eset különböztethető meg. A program tartozhat állandó jelleggel a vezérlőhöz, mintegy a szerkezetben rögzítve, vagy abba esetenként betáplálható és tárolható. A rögzített programú vezérlőt kötött (hardware), a szabadon változtatható programút kötetlen (software) programozású vezérlőnek nevezzük. Ennek megfelelően kötött, illetve kötetlen programozású mérőrendszert különböztetünk meg.

A legkezdetlegesebb manuális rendszerek kötetlen programozásúak, ezekben az ember a vezérlő. Az automatizálás első irányzatát kötött programozású rendszerek jellemzik. A számítógép alkalmazása a méréstechnikában: visszatérés a kötetlen programozásra.

A kötött és a kötetlen programozású vezérlők között különböző átmeneti változatok találhatók. Egyrészt bizonyos vezérlési műveleteket még az ember tarthat kézben, míg mások kötött program szerint futnak. Másrészt a kötött programozás merevsége feloldható olyan szerkezeti megoldások alkalmazásával, amelyek megengedik a vezérlő struktúrájának egyszerű módon való megváltoztatását, variálását. Ez utóbbi törekvések jegyében születtek meg a különböző dugaszmezők, változtatható mintájú programkapcsolók stb. Kiterjedtebb programoknak a struktúrában való rögzítése, a már rögzített programnak új programmal való helyettesítése rendkívül nehézkes feladat. A nehézségek csökkentése érdekében kezdték alkalmazni a különböző fix memóriákat, a lyukszalagot és lyukkártyát, a mágneskártyát, mágnesszalagot és díszeket mint a mérési program hordozóját.

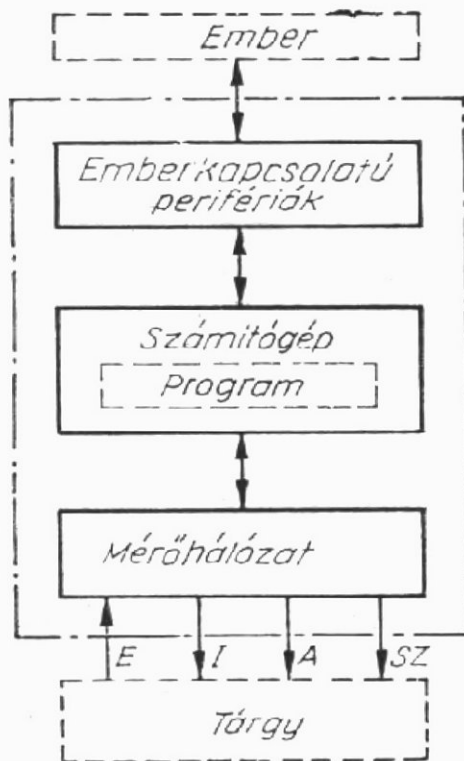
A mérőrendszer automatizálásának követelményei sorában — a vezérlés gépesítése mellett — az adatfeldolgozás gépesítése is jelentkezik. A rendszervezérlő szerepkörét, éppúgy mint az adatfeldolgozóét, memóriával kiegészített processzor veszi át. A processzor — adottságai révén — bekapcsolódhat a munkapont-állítás, szabályozás, sőt a jelgenerálás folyamatába is. Azzal a képességgel is rendelkezik, hogy az előző mérés eredményétől függően maga válassza ki a soron következő végrehajtandó programszakaszt.

A mérőrendszer automatizálásának igénye a mérőhálózat oldalán, a végrehajtó elemek (készülékek) vezérelhetőségének problematikáját veti fel. A manuálisan állítható elemek helyébe program által irányítható elemek lépnek.

1.2. Számítógép-irányítású mérőrendszerek

A rendszerek osztályozása

A legmagasabb színvonalú mérőrendszerekben a vezérlés és adatfeldolgozás feladatkörét, s az esetlegesen felmerülő szabályozási feladatokat is számítógép látja el. A számítógép adatfeldolgozóként részt vehet összetett, átfogó mérési adatok, ill. eredmények képzésében, továbbá jelalak-generálásban is. Az általa ellátott komplex feladatkört irányításnak nevezzük, s ilyen értelemben beszélünk számítógép-irányítású mérőrendszerrel, melynek leegyszerűsített vázlatát az 1.4. ábra mutatja be. A részletek (a tárggyal való kapcsolat mélysége, a mérőhálózat felépítése és természete, az alkalmazott számítógép milyensége és szerepköre, valamint az emberrel vagy másik számítógéppel való kommunikáció) tekintetében az egyes konkrét rendszerek — gyakran igen szélsőséges módon — különböznek egymástól.



1.4. ábra.

Számítógépes mérőrendszer alapsémája

E — érzékelés; I — inger; A — alapjel; SZ — szabályozás

A tárggyal csupán az érzékelőcsatornán (É) összeköttetésben álló rendszert *adatgyűjtő* rendszernek nevezik. Adatgyűjtő rendszert abban az esetben alkalmaznak, amikor a tárgy vizsgálatához (a mérőjelek előállításához esetleg szükséges, érzékelőcsatornán belüli kérdőjeleken kívül) a mérőrendszer oldaláról más ráhatás nem szükséges, vagy ha a ráhatást az ember közvetlenül gyakorolja. Valamely elektronikus berendezés huzalkeretének vizsgálata pl. nem igényel sem inger (I), sem alapjel (A), sem szabályozó (SZ) jelet. A Diesel-motor adott esetben kézi vezérléssel is végigvezethető a munkapontok során, miközben egy-egy munkapont üzemi adatait az adatgyűjtő rendszer felveszi, feldolgozza, ellenőrzi stb. Ugyancsak kézi úton elvégezhető a nyert mérési adatok alapján bizonyos jusztirozási műveletek is.

Az *aktív mérőrendszert* a tárggyal az érzékelő-csatornán kívül az I és A vonalaknak legalább egyike összeköti.

Közvetlen szabályozó mérőrendszerről akkor beszélünk, ha az E mellett legalább az SZ hatásvonal is megtalálható.

A számítógépes mérőrendszer automatizáltságának mértéke — az elmondottakból következően — relatív. Az adatgyűjtő rendszer a tisztán adatgyűjtési feladatok szempontjából az automatizáltság legmagasabb fokán állhat, míg ugyanaz a rendszer munkapont-állítást és szabályozást igénylő vizsgálat kapcsán csupán részlegesen automatizáltnak nevezhető.

A tárgy és a mérőrendszer kapcsolatának mélysége nemcsak a hatásvonalak számában és milyenségében, hanem a csatlakozás módjában is különbözhet. Valamilyen konkrét tárgy vizsgálata során az általában számos feladatra (és különböző tárgyak vizsgálatára) szolgáló mérőrendszer csatlakozási vonalait hozzá kell rendelni és kapcsolni a tárgy vizsgálati pontjaihoz, továbbá biztosítani kell a tárgy üzemi feltételeit (pl. félvezető áramkörök vizsgálatakor a tápfeszültségeket és a terhelőellenállásokat). Ki végzi el a vonalak és mérőpontok egymáshoz rendelését, összekapcsolását, ki biztosítja a vizsgálat üzemi feltételeit: a számítógép vagy az ember? A választól függően ugyancsak a mérőrendszer automatizáltságának (bár az előzőtől különböző) mértékéről beszélhetünk.

A mérőhálózat az E, I, A, SZ csatornákon áthaladó jelek milyenségétől függően analóg, digitális vagy vegyesen, analóg és digitális jellegű. Ennek megfelelően analóg, digitális vagy általános jellegű mérőrendszer különböztethető meg.

A mérőrendszer lehet központosított vagy lehet elosztott vezérlésű. Központosított vezérlés esetében a mérőhálózat belső vezérlési feladatainak túlnyomó hányadát a számítógép látja el. Elosztott vezérlés esetén viszont a vezérlési feladatok jelentős része a mérőhálózati elemekhez rendelt készülékvezérlőkre hárul. A készülékvezérlők kötött (szerkezetben rögzített) vagy kötetlen (mikroprocesszor) programozásúak. A mikroprogramozott készülékvezérlő átmeneti típust képvisel. A vezérlési feladatokhoz hasonlóan az adatfeldolgozási feladatok is koncentrálnak, vagy (legalább bizonyos mértékben) a rendszer elemei között szét is oszthatók.

A számítógép szerepkörében — adott esetben — alternatívaként mikroszámítógép, programozható controller, valamint asztali kalkulátor is szóba jöhet, attól függően, mennyire kiterjedt a mérési feladat, ill. a mérőhálózat, milyen volumenű vezérlési és számítástechnikai munkát kell ellátni, mik a mérési időre vonatkozó követelmények stb.

Az ember—gép kapcsolatot szolgáló periferikus készülékek száma és színvonala a rendszer méreteinek egészével áll arányban.

A kisszámítógép és alternatívái

A *kisszámítógép* a mérőkészülékek ár- és méretkategóriájába eső olyan számítástechnikai eszköz, mely struktúrája, architektúrája, szervezés- és működésmódja tekintetében a számítógépek általános jellemzőivel írható le. A „mérőkészülékek ár- és méretkategóriája” helyébe „áramköri lap ár- és méretkategóriáját” állítva, a *mikroszámítógép* meghatározásához jutunk.

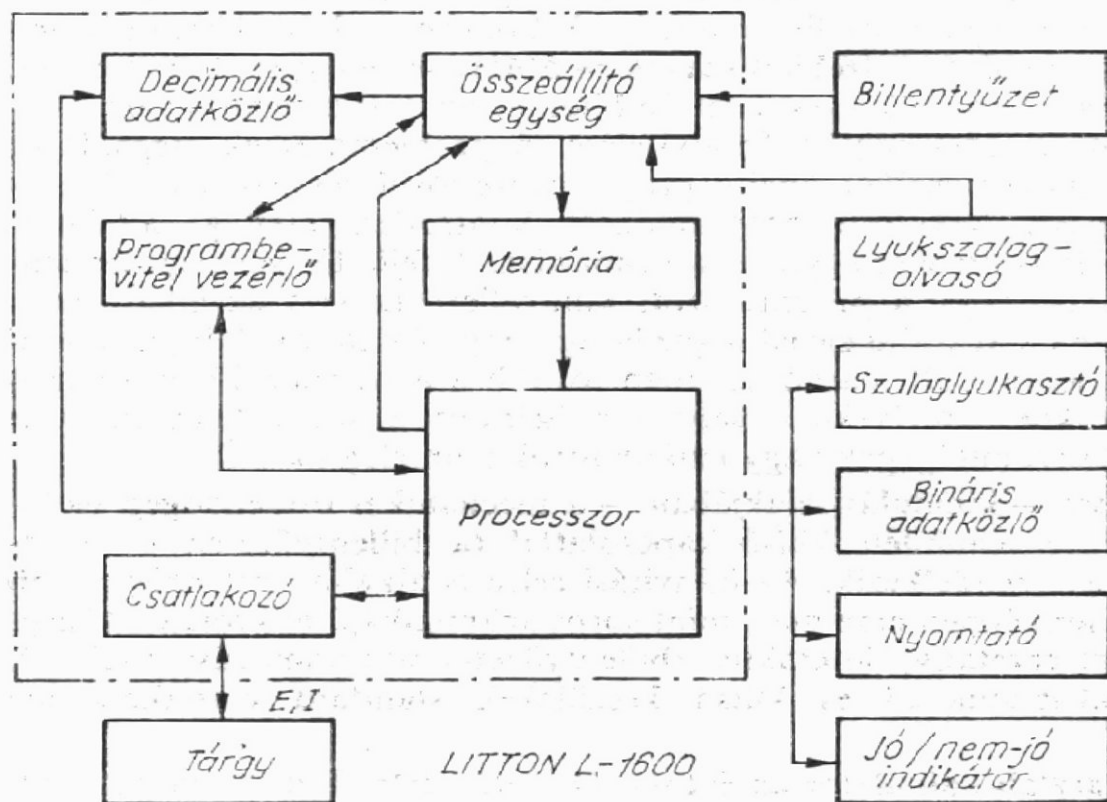
A kisszámítógépet — mérőrendszer szempontjából — a készülékek egységes csatlakoztathatósága és cserélhetősége, a csatlakoztatható készülékek nagy száma és e szám változtathatósága; általános utasításkészlet; kötetlen programozhatóság, kiterjedt programok végrehajtásának, több program szimultán futtathatóságának lehetősége; nagy feldolgozási sebesség, jelentékeny csatornaátviteli sebesség jellemzi. A kisszámítógép a moduláris mérőrendszer legáltalánosabban alkalmazott vezérlő-adatfeldolgozó eleme.

A mikroszámítógép (mikroprocesszor) struktúrája és architektúrája tekintetében a kisszámítógéppel rokon, s mint ilyen, moduláris mérőrendszer irányító egységként

használható. A mikroprocesszor-elemek azonban ugyanolyan jól alkalmazhatók integrált mérőrendszerekben is. Joggal feltételezhető, hogy ennek következményeképpen a jövőben gazdaságossá és előnyössé válik a feladatorientált mérőrendszerek alkalmazása újabb, az automatizálás által eddig nem érintett mérés-technikai területeken is. A mikroprocesszorra különösen fontos szerep vár az elosztott intelligenciájú (elosztott vezérlésű és adatfeldolgozású) rendszerekben.

Kontrolleren, eredeti értelemben a relékből, ill. diszkrét félvezető elemekből felépülő ipari vezérlőt értik; mai, programozott változata az igen egyszerű nyelvű vezérlő-programot tárolt alakban őrzi. Kezdeti formái elsősorban az általa vezérelt objektummal állnak kapcsolatban; az emberrel való kommunikáció szegényes. A programozott változatnál azonban ez a kommunikációs kapcsolat bővül ki, miáltal a kontroller a kiséghez közelít. A programozott kontroller kialakulásának másik területe éppen a mérés-technika, ahol a korábbi lyukszalag-, mágnesszalag-vezérlés szerepkörét veszi át. Mérés-technikai értelemben (és a mérés-technikában használt) programozható kontrolleren az igen egyszerű, néhány utasítást tartalmazó, feladatorientált utasításkészlettel és szegényes kommunikációs lehetőségekkel rendelkező kis- (mikro-) számítógépet értjük.

A programozható kontroller, leginkább a feladatorientált (integrált) mérőrendszer irányító eszköze, bár önálló egységként előfordul moduláris mérőrendszerben is, főleg periferikus szerepkörben. Az általános jellegű, a feladatok tágabb horizontjára tekintő kis- és mikrogéppel szemben a kontroller szerkezeti és programozási szempontból egyaránt csak az adott feladat ellátásához szükséges elemeket tartalmazza. Utasításkészletében kizárólag a feladat végrehajtásához szükséges, kevés számú utasítás található. Gyakran szerkezeti-ileg sem különül el a mérőhálózattól, hanem általában azzal szoros egységet képez. Az 1.5. ábra — az elmondottak illusztráció-



1.5. ábra.
Kontroller-vezérlésű integrált mérőrendszer

jaként — kontroller vezérlésű, közepes- és nagy integráltságú áramkörök, valamint áramköri kártyák digitális és analóg funkcionális ellenőrzésére szolgáló, feladatorientált rendszer felépítését szemlélteti. Láthatók a számítógép jól ismert egységei, de

a memória a kisgép szokásos min. 4 K kapacitása helyett mindössze 512 rekesz 12 bit kapacitású, a processzor pedig jelgeneráló, jelvevő, jelvizsgáló és összehasonlító mérőhálózati elemeket is tartalmaz; ugyanakkor aritmetikai képességei nincsenek. Az utasításkészlet — a kisgépeknél átlagos 50...100 utasítás helyett — mindössze 20 utasítást tartalmaz, melyek közül 8 vezérlés jellegű és 12 operatív jellegű. A program a processzor közreműködésével billentyűzetről vagy szalagolvasóról vihető be a mérőrendszerbe, ahol is az összeállító egységben az utasítások a rendszer belső formátumára transzformálódnak át. A vezérlőutasítások a programbevitel-vezérlőn át közvetlenül jutnak a processzorra; ezek alapján kezeli a processzor a periférikus készülékeket, és ezek alapján — még a program operatív részének végrehajtása előtt — választja ki a tárgycsatlakozó érintkezői közül a bemenő és kimenő pontokat. Itt ilyen utasítások találhatók: az xx pont kimenő pont; szalag kezdete; szalag vége; a lyukszalag-karakter vagy a memória adatregiszter tartalmának törlése stb. Az operatív utasítások (ezekből épül fel a vizsgálati program) a memóriában tárolódnak, s a berendezés kezelője által megválasztható ütemben hajtódnak végre. A billentyűzetről a memóriába ültetett program közvetlenül lyukszalagra is átvihető. A vizsgálati eredményt jó/nem jó indikátor nyújtja. Az ellenőrző programok és ellenőrzési eredmények nyomtatóval naplózhatók.

A programozható kontroller feladatorientált mérőrendszerekben való alkalmazása gazdasági előnyökkel járhat. A kontroller egyedi tervezésének költségei messzenőven visszatérülhetnek a mérőrendszer egészének ökonomikus felépítése révén. Alkalmazását azonban nem egyedül gazdasági szempontok indokolhatják. Esetenként valamely csekély kiterjedésű, viszonylag kevés elemet felölelő mérőhálózatot az általános jellegű számítógép a cél szempontjából fölösleges alkatelemeivel, hozzáférhetőségének problematikájával indokolatlanul túlterheli. Máskor az általános gép tulajdonságai éppen az adott feladat követelményeit nem elégítik ki. Még a kiemelkedően nagy sebességű átvitelrel rendelkező Lockheed SUE kisszámítógép sínrendszere sem teszi lehetővé pl. 16 bites szavak $> 5\text{MHz}$ sebességgel való átvitelét. Bizonyos esetekben azonban szükséges lehet 16 bitesnél hosszabb bináris vektoroknak pl. 10 vagy 20 MHz sebességgel való átvitelére. A probléma egyik megoldása: általános jellegű számítógép mellett önálló programvezérlővel rendelkező szógenerátor, a másik: egyedi tervezésű programozott kontroller, esetleg: általános jellegű kisszámítógép és kontroller együttes alkalmazása. A fejlődés jelenlegi tendenciáit tekintve úgy tűnik, hogy a programozott kontrollert azokon az alkalmazási területeken, ahol használatát eddig egyedül gazdasági tényezők indokolták, a mikroprocesszor váltja fel; létjogosult marad azonban továbbra is mindazokban a speciális alkalmazási körökben, melyeknek sebességi igényeit a sorozatban gyártott mikroprocesszorok, kisszámítógépek vagy kalkulátorok nem elégítik ki.

Az *asztali kalkulátor* — kiindulási alakjában — a mechanikai összeadó gép elektronikus változata, mely semmiféle külső kapcsolattal (a billentyűzeten és adatmegjelenítőn kívül) nem rendelkezik. Az irányítási célra is alkalmazott legfejlettebb változat viszont tulajdonképpen nem más, mint soros aritmetikájú mikroszámítógép, magas színvonalú interpretatív, interaktív (billentyűzet-) programozási nyelvvel, jelentékeny memóriakapacitással és külső készülékek standard csatlakoztatási lehetőségével.

A programozható asztali kalkulátor az évtized kezdetén érte el azt a színvonalat, amely mérőrendszerben, irányító egységként való alkalmazásának gondolatát felvette. Ezzel együtt napirendre került és megoldódott a külső készülékek és a kalkulátor egymáshoz csatlakoztatásának problematikája is. Míg a korábbi kalkulátorokat a billentyűzeten át a matematika nyelvén programozták, s az általuk végezhető művelettípusok száma viszonylag csekély, a memóriakapacitás kicsiny (300...400 rekesz) és a feldolgozási sebesség is mérsékelt, addig a korszerűbb változatok

tulajdonságai sok tekintetben a kisszámítógépekével vetekednek. Röviden jellemezve pl. az irányítási igények figyelembevételével megalkotott HP 9830A típusú kalkulátort (1972): Alfa-numerikus billentyűzetén át BASIC nyelven programozható; 7,5 K-rekesz fixmemóriával és 2 K-rekesz olvas/ír (read-write) memóriával rendelkezik (az előbbi 16 K, az utóbbi 4 K-ig bővíthető). 16 bit szóméretű soros mikroprocesszorának órafrekvenciája 8 MHz. Utasításkészlete 75 gépi utasítást tartalmaz. Belső, tehát inkorporált periférikus készülékei sorában — a billentyűzetén kívül — mágneskazettát; 32-jegyű alfa-numerikus adatközlőt; 250 sor/min (1 sor 80 betűhely) teljesítményű sornyomtatót találunk. Az utasításkészlet tíz kimenő utasítását külön csatornaprocesszor értelmezi és hajtja végre. A belső periférikus készülékeken kívül további négy külső csatlakoztatható; bővítőt alkalmazva e szám négy fölé növelhető. Lehetőségeit tekintve a programozható kalkulátor e fejlett változata már igen közel áll a kisszámítógépéhez (annak tulajdonképpen speciális változata) s mint ilyen, moduláris mérőrendszer irányító egységeként alkalmazható. Alkalmazása kapcsán figyelembe kell venni alábbi jellegzetes tulajdonságait, melyek a kisszámítógéptől megkülönböztetik, s amelyek sajátos alkalmazási területeit is meghatározzák:

1. Programozási technikája lényegesen egyszerűbb, mint a kisszámítógépé.
2. Eleve tartalmazza a legszükségesebb periférikus készülékeket; ezek csatlakoztatási problémáival a rendszertervezőnek egyáltalán nem kell foglalkoznia. A perifériák csatlakoztatási előírásai könnyen áttekinthetők, a csatlakozó kártyák (készülékvezérlők) felépítése egyszerű.
3. BASIC nyelve és soros átviteli csatornája révén minden további nélkül összekapcsolható időosztásos számítógéppel.
4. Paralel be/kimeneti csatornája nem teszi lehetővé több, mint 10...100 adat/s begyűjtését, ill. kiadását.
5. Be/kimeneti struktúrája egyszerű; nem tartalmaz hibaellenőrzést; a periférikus egységek számának növelésével a megbízhatóság csökken.

A jelenlegi kalkulátorok alkalmazása mindenekelőtt a nem kifejezetten elektronikus számítógépes környezetben mutatkozhat előnyösnek; olyan területen, ahol gyakran kell újabb és újabb programokat készíteni, tehát: ahol a vizsgálat tárgya gyakran változik, a mérőrendszer adatforgalma mérsékelt, a méréstechnikai feladat jelentékeny mennyiségű számítással jár együtt, s ugyanakkor központi, időosztásos gépre (terminálként) való csatlakoztatás lehetősége fennáll. Elsősorban laboratóriumi felhasználása jöhet szóba, bár természetesen bizonyos üzemi mérések irányítási feladatainak ellátására is alkalmazható.

A programozható kontroller, a programozható asztali kalkulátor, a mikrogép és a kispép a számítógép alsó teljesítményspektrumának egy-egy képviselője. Sajátos tulajdonságaik alapján a mérésautomatizálásban egymástól eltérő igényeket elégítenek ki. Eredeti célkitűzésünknek megfelelően a továbbiakban elsősorban a kisszámítógép mérőrendszerbeli alkalmazásának problematikáját tartjuk szem előtt.

A számítógép-irányítású mérőrendszer sajátosságai

A számítógép-irányítású (moduláris) mérőrendszer legjellegzetesebb vonásai az alábbiakban foglalhatók össze:

- A mérési feladatot automatikusan bonyolítja le.
- Programozása kötetlen, miáltal tetszőleges számú különböző mérési feladat algoritmusát a szerkezet megváltoztatása nélkül képes végrehajtani. E tulajdonsága révén igen flexibilis.

- Memóriájában egyidejűleg egynél több program is tárolható, melyek akár egymás után, akár időosztásos módon szimultán futtathatók. A számítógép egynél több mérőhálózatot, a mérőrendszer több mérőhelyet is elláthat.
- Számítástechnikai adottságai képessé teszik a primer mérési adatok tömörítésére, a mért értékekből kiindulva átfogó paraméterek értékének képzésére, adatok összehasonlítására, szortírozására, statisztikai értékelésére; középérték, szórás, trend stb. vizsgálatára; jegyzőkönyv készítésére.
- Képes önnön hitelesítésére.
- Képes a mért értékek, esetleg mérési jelleggörbék korrekciójára.
- Programozási eszközökkel ellenőrizheti önnön üzemképességét, s a kezelővel együttműködhet esetleges saját működési hibáinak felderítésében.
- Döntési képességei révén adaptív természetű. A már lefolytatott vizsgálat eredményétől függően az általa végrehajtott program menetét önmaga módosíthatja.
- A mérés eredményétől függően diagnosztikai vizsgálatot végezhet.
- A kvázi-egyidejűség és az időrealitás feltételeinek a feladatok igen széles körében eleget tesz.
- Igen sok részletre kiterjedő, az ember áttekintőképességét meghaladó vizsgálatot végezhet.
- Lehetővé teszi az egy feladat által lekötött magas képzettségi színvonalú méréstechnikusok számának csökkentését.
- A feladatorientált rendszerekkel szemben rendszertechnikai tekintetben egységes területhez tartozik. Minden egyes konkrét rendszer létrehozása kapcsán a tervező meríthet az általánosból, minden realizáció egyúttal gyarapítja az általánost is.
- A számítógép-irányítású műszaki mérőrendszerek családja beletartozik a rendszerkapcsolatú számítógépes struktúrák nagyobb törzsébe (pl. a folyamat-szabályzó, orvosi és biológiai, kísérleti fizikai, stb. rendszerekkel együtt), ami akár a szerkezeti megoldások, akár a programozás szempontjából, az analógiák révén további információhátteret nyújt.
- A moduláris rendszer, a feladatorientálttal szemben, jórészt sorozatban gyártott egységekből építhető fel; megvalósítása nem igényel kiterjedt áramköri tervezői munkát, emiatt tervezési ideje viszonylag rövid; konstruktíve kiforrottabb alakban áll elő.
- Modularitása révén összetétele egyszerűen módosítható; újabb mérőkészülékek kapcsolhatók hozzá, míg mások elhagyhatók. Ezen lehetősége műszaki elavulásának időpontját kitolja.

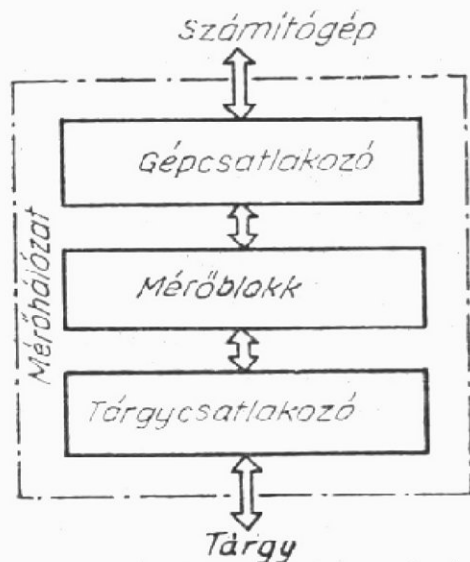
1.3. Mérőhálózatok

A hálózat felépítése

A számítógépes mérőrendszer mérőhálózatához (1.4. ábra) a mérőblokkon kívül, mely a szűkebb értelemben vett mérésorientált elemeket öleli fel, gépcsatlakozó és tárgycsatlakozó is tartozik (1.6. ábra). Az előbbi a mérőhálózat és a számítógép, az utóbbi a mérőhálózat és a tárgy összekapcsolására szolgál.

A mérőblokknak az érzékelőcsatornához tartozó, meghatározott mérési feladatokat ellátó egységeit összefoglaló névvel vevőknek, a többi csatornához tartozó funkcionális egységeit adóknak nevezzük. Bizonyos esetekben a vevők mint mérőkészülékek, az adók mint jelgenerátorok szerkezetileg is önálló egységet képezhetnek. Más esetben a mérőblokkon, mérőhálózaton, vagy tágabb körre tekintve a mérő-

rendszeren belül, az egységeknek csak funkcionális határai vonhatók meg, anélkül, hogy fizikai elkülönítésük lehetséges volna. A mérőblokk továbbá tartalmazhat a vevők és adók esetleges egymás közötti összeköttetésére szolgáló belső kötőelemeket, valamint a tárgy tényleges üzemi körülményeinek szimulációjához szükséges elemeket is. Az előbbieket sorában vezetékeket és programvezérlésű kapcsolókat, az utóbbiak sorában terhelőellenállásokat stb. találhatunk.



1.6. ábra.
Számítógéppel irányított mérőhálózat

A mérőhálózat integrált egységként vagy modulokból épülhet fel. Ugyanúgy, mint az egész mérőrendszer vonatkozásában, az integrálás a mérőhálózat esetében is egyrészt szűk feladatorientáltsággal, másrészt az építőelemek szoros strukturális kapcsolatával jár együtt; az integrált mérőhálózat nem bontható fel önállóan is alkalmazható egységekre.

A moduláris építésmód jellegzetességét a fizikailag is elkülönülő funkcionális egységek adják. A modularitásnak két szintje is megkülönböztethető: a készülékszint és az építőkocka-szint. A készülékszinten moduláris rendszer hagyományos készülékekből (pl. digitális voltmérő, impulzusgenerátor, multiméter stb.) áll. Az építőkocka-szinten a készülékeknél alárendeltebb feladatkörű áramköri egységeket találunk (pl. A—D átalakító, D—A átalakító, műveleti erősítő, tároló-mintavevő stb.).

A készülékszinten moduláris mérőhálózat a legáltalánosabb, legkevésbé feladat-orientált, viszont egy bizonyos mérés-technikai feladat szempontjából redundáns lehet. Építőkockákból a mérőhálózat gazdaságosabban építhető fel, de az építőkockákból álló rendszer kevésbé általános. Új rendszer legegyszerűbben a készülék-modularitás szintjén szervezhető. Legnagyobb szellemi befektetést az integrált egység megvalósítása igényel. Az építőkocka-módszer a kettő között helyezkedik el.

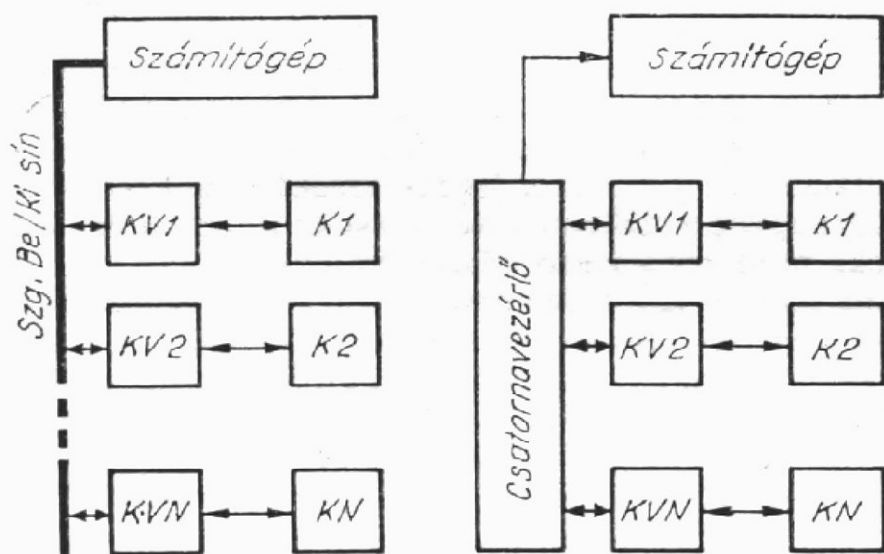
A fenti megfontolás természetesen csupán az esetben érvényes, ha a mérőhálózatot a rendszer tervezőjének kell összeállítania. Bizonyos mérőhálózatokat realizáló integrált berendezések azonban standard alakban is rendelkezésre állnak, az építőkockák létrehozásával kapcsolatban pedig általában együttjár rendszerszervezési elveik kimunkálása is; rendelkezésre állnak tehát azok a sémák, melyek alapján meghatározott feladatkörű mérőkészülékek, ill. mérőhálózatok belőlük összeállíthatók.

A gépcsatlakozó

A gépcsatlakozó a számítógép által a mérőhálózathoz címzett parancsokat átveszi, értelmezi és végrehajtja; a végrehajtás során a mérőhálózat által szolgáltatott mérési, azonosítási és ellenőrzési adatokat a számítógéphez továbbítja. A gépcsatlakozó tartalma és felépítmódja a mérőhálózat megvalósításának módjától és a tárgycsatlakozó tartalmától függően változik. Az esetben, amikor a számítógéphez egyetlen, vevő típusú mérőberendezés társul, a tárgycsatlakozó pedig nem igényel vezérlést, a gépcsatlakozó egyetlen készülékvezérlőből áll. Ezzel szemben a készülékszinten moduláris mérőhálózat minden egyes készülékéhez saját készülékvezérlő járul, s külön készülékvezérlő szolgálja ki a mérőblokkon belüli és a tárgycsatlakozóhoz tartozó programvezérlésű kapcsolóelemeket is. Végül építőköcska-szinten moduláris mérőhálózat esetében a teljes mérőblokkot ismét csak egyetlen készülékvezérlő kezelheti, ha az építőelemeket egységes rendszerszervezési elvek fűzik össze.

A gépcsatlakozó, a gépcsatlakozóhoz tartozó készülékvezérlők egyik oldalon a vezérelt mérőhálózati elemekre, másik oldalon a számítógép be/kimenetére csatlakoznak. A különböző számítógépek egymástól eltérő csatlakozási paraméterekkel rendelkeznek, azonkívül az egyes vezérelhető elemek — természetesen — sajátos vezérlési követelményeket támasztanak, ami nehézségek forrása. Bár számos, az iparban előállított mérőkészülékhez és moduláris építőköcska-rendszerhez rendelkezésre állnak a leginkább ismert kisszámítógép-típusokhoz szolgáló készülékvezérlők, ez a géphez való csatlakoztatás problémáját általánosságban nem oldja meg. A kis- és mikrogépek a rohamos fejlődés és változás állapotában lévén, csatlakozási paramétereik egységesítése a közeli jövőben nem várható. Áthidaló megoldásként a gép és a készülékvezérlők közé csatornavezérlőt iktatnak; a számítógép és a csatornavezérlő közötti csatlakozásmód egyedi, a csatornavezérlő és készülékvezérlők csatlakoztatásmódja pedig megállapodás szerinti, egységes. A kétféle megoldást, készülékszinten moduláris mérőhálózat esetére, az 1.7. ábra szemlélteti. A készülékoldalon egységesített csatornavezérlő alkalmazásának egy példája, a 4. fejezetben részletesebben leírt CAMAC-rendszer.

A csatornavezérlő alkalmazása nem csak a csatlakoztatás egységesítését, hanem áramköri elemek megtakarítását is eredményezi, minthogy bizonyos valamennyi készülékvezérlőre nézve közös funkciók a csatornavezérlőre háríthatók át. A készülékvezérlőkben e funkciók elemei elmaradnak.



1.7. ábra.

A készülékek számítógép-csatlakoztatásának kétféle módja

KV — készülékvezérlő; K — mérőkészülék

A tárgycsatlakozó

A mérőblokk és a tárgy között létesít összeköttetést. Tartalmában és felépítésmódjában mérőrendszerenként más-más lehet. A gázturbina pl. csatlakoztatása tekintetében természetesen egész más követelményeket támaszt, mint valamely digitális integrált áramkör. Általános esetben a mérőblokk M -számú, a tárgy pedig K -számú csatlakozóponttal rendelkezik. Az M és a K nagysága, e számok egymáshoz való viszonya, a kétoldali csatlakozópontok meghatározott individumokból felépülő részhalmazainak egymáshoz rendelése és összekapcsolása azok az elsőrendű feladatok, amelyeket a tárgycsatlakozónak el kell látnia. E mellett, gyakran nem pusztán galvanikus, hanem átvitel- vagy mérés technikai szempontból illesztett összeköttetést kell létesítenie olyan körülmények között, amikor az átviendő jelek spektruma az egyenáramtól a nanoszekundumos impulzustartományig terjed.

A tárgycsatlakozónak rendkívül lényeges paramétere a csatlakoztatási idő, amely alatt a tárgy a mérőrendszerrel összekapcsolódik. A csatlakoztatás idejének arányban kell állnia a mérési idővel. Összetettebb tárgyak esetében, melyeknek hosszú a vizsgálati ideje, megengedhető a kézi úton való csatlakoztatás; az egyszerűbb, igen rövid mérés idejű tárgyak tömeges mérése viszont automatikus adagolók alkalmazását teszi szükségessé. Tranzisztorok sorozatvizsgálata történhet pl. úgy, hogy ellenőrizendő tranzisztorokat százas mérőpaklikba fogják össze. A paklit azután kézi úton kapcsolják a rendszerre. Az első tranzisztor levizsgálása után a tárgycsatlakozó részét képező kapcsolórendszer automatikusan átvált a második tranzisztorra, a harmadikra és így tovább. A kapcsolórendszer helyettesíthető adagolóval, amely a tranzisztort automatikusan a mérőpontokra kapcsolja, mérés után automatikusan leválasztja a mérőpontokról, veszi a következő tranzisztort stb. Vannak olyan objektumok, amelyek racionális tömeges mérése automatikus adagoló nélkül elképzelhetetlen. A 0,3 mm ferritmagnak a mérőrendszerre való kézi „rákapcsolása” megoldhatatlan. A korszerű adagolók 130 000 mag/h teljesítményt érnek el; a magokat a ns-os időtartományban vizsgálják.

A tárgy és mérőrendszer csatlakoztatásának sajátos feladata merül fel abban az esetben, amikor egynemű (elektromos) mennyiséget a tárgy számos pontján kell vizsgálni. Ilyenkor nem alkalmaznak minden mérőponthoz külön mérőkészüléket, hanem egyetlen mérőkészülék idejét megosztva, a mérőpontokat méréspontváltó segítségével letapogatják.

A tárgycsatlakozóban gyakran előforduló programvezérlésű individuális, kapcsolócsomagok, mátrixok, méréspontváltók, kapcsolófák stb. a számítógépes mérőrendszer nélkülözhetetlen elemei. Sorukban kis- és nagyteljesítményű, lassú és gyors, kis- és nagyfeszültségű, kis- és nagyfrekvenciás (koaxális) változatok találhatóak, amelyek száraz és higanyal nedvesített kontaktusú reed-relé vagy félvezető kapcsolóelemekből épülnek fel. A kapcsolórendszerek fontos minőségi jellemzője a flexibilitás. A kapcsoló annál flexibilisebb, minél többféle kapcsolási variáció valósítható meg általa. A kapcsolórendszernek — a nem kívánatos helyzetek elkerülése végett — minden esetben a programrendszer ellenőrzése alatt kell állnia. Csak így kerülhető el, hogy a mérőkör egy vevőjének bemenete a tárgy két vagy több különböző pontjára kapcsolódjék egyidejűleg, vagy hogy több adó jele egyidejűleg jusson a tárgy egy csatlakozási pontjára. A programrendszer minden a feladatprogram által kezdeményezett akciót kell, hogy ellenőrizzen, s ha ez utóbbi meg nem engedhető kapcsolat létrehozására ad utasítást, a mérést hibajelzés kíséretében le kell, hogy állítsa.

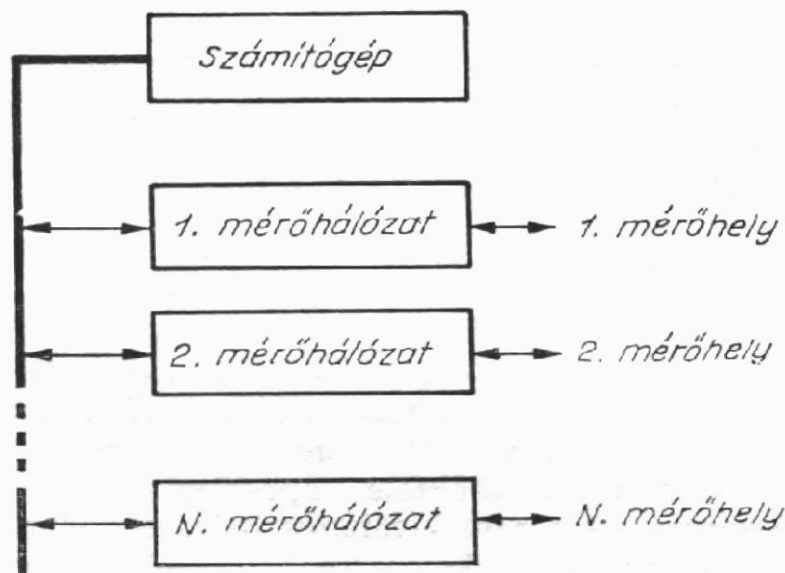
A tárgy és a rendszer mérőpontjainak egymáshoz rendelésére és összeköttetésére esetenként nem kapcsolókat, hanem összekötő adaptereket használnak. Az adapter egyik oldalán a mérőrendszer rögzített elrendezésű mérőpontjaira csatlakozik, másik oldalán pedig a tárgyra. Minden mérési objektumhoz individuális adapter

tartozik, az illető tárgy mérőpontjaihoz vezető huzalozással. Az adapter a huzalozáson kívül egyéb elemeket (pl. munkaellenállásokat), sőt a rendszernek szóló programellemeket is tartalmazhat (1.35. ábra).

Az analóg áramkörök általános vizsgálatára szolgáló számítógépes mérőrendszer-től lényegesen nagyobb flexibilitást követelnek meg, mint a digitális áramkörvizsgálóktól, aminek oka: az analóg áramkörök jellemzőinek igen tág tartománya. Teljesen más jellegű vevőket és adókat követel ui. az egyenfeszültségű stabilizátor-körök vizsgálata, mint pl. az impulzuserősítő mérése, s a példát vég nélkül sorolhatnánk. A tapasztalat azt mutatja, hogy a flexibilitás követelménye egyedül a mérőblokk oldaláról nem elégíthető ki. A tárgycsatlakozó részeként — a mérőkör és a tárgy közé — ún. áramköri csomagokat kell elhelyezni. A csomag a tárgyak egy körének vizsgálatához szükséges kiegészítő elemeket tartalmazza. Az elemek között található detektorok, jelforrások, pufferkörök, szűrők, kompenzálókörök, áramérzékelő ellenállások, csatlakozók stb. A tárgyak egy egész osztályát kiszolgáló csomagot anyalemeznek (mother board), a csak egy bizonyos objektumhoz (terméktípushoz) tartozó csomagot tárgylemeznek (device board) nevezik. A tárgy közvetlen csatlakoztatására az adapter (socket adapter) szolgál. Az audio—stereo anyalemez pl. az összes hangfrekvenciás készülékek (stereo demodulátorok, TV/FM hangrendszerek) vizsgálata esetében változatlan, de különböző tárgylemezekkel egészül ki. A tv/fm hangrendszerek vizsgálatára szolgáló tárgylemez pl. RF detektort, hangfrekvenciás puffert és külső jelgenerátor-csatlakozást hordozhat. Az adapter sem mindig puszta csatlakozó; csatolásmentesítő és/vagy kompenzálóköröket foglalhat magában.

Az áramköri csomagok egy része rendszeren kívüli készülékek (pl. jelforrások) csatlakoztatására alkalmas kapcsokkal is rendelkezik, előre nem látható követelményekkel is számolva.

A számítógéphez adott esetben egynél több mérőhálózat, egynél több mérőhely is csatlakoztatható (1.8. ábra) egymást kizáró vagy időosztásos üzemmódban.



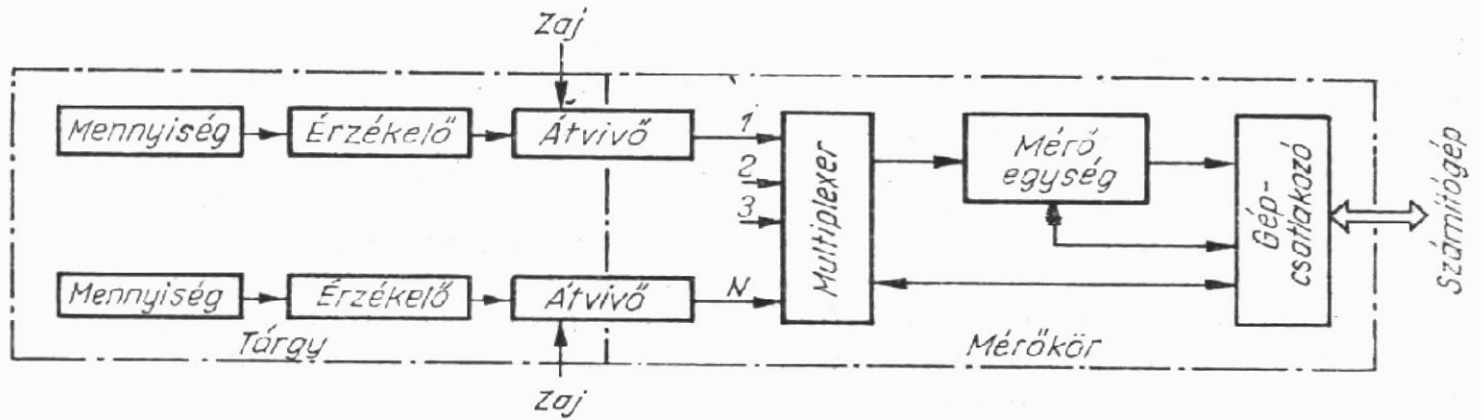
1.8. ábra.

Több-mérőhelyes számítógépes mérőrendszer

Analóg mérőhálózatok

A folytonos analóg mennyiségek mérésére szolgáló adatgyűjtő hálózatot legáltalánosabb alakjában az 1—9. ábra szemlélteti. Az N -számú mérőponthoz tartozó mérőcsatornák a multiplexer segítségével időben egymás után kapcsolódnak az egyetlen mérőegységre, amely ily módon hely és idő szerinti multiplex mérést végez.

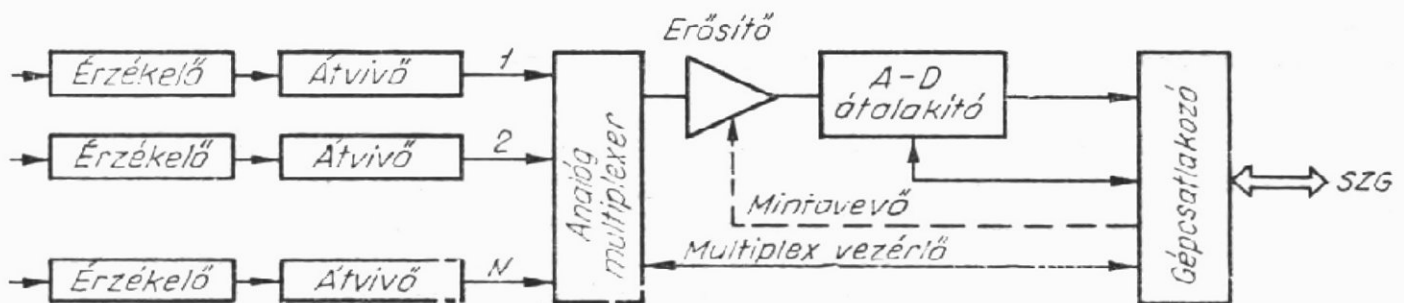
A hálózat vezérlését a számítógéptől kapott parancsok alapján a gépcsatlakozó látja el; ugyanez továbbítja (a multiplexelés ütemében) feldolgozás céljából a mérőegység által szolgáltatott adatokat a számítógéphez. Az érzékelő-átalakítók által



1.9. ábra.
Analog adatgyűjtő hálózat általános sémája

szolgáltatott jel lehet a mért mennyiség nagyságával (megváltozása mértékével) analóg feszültség vagy áram, de lehet impulzussorozat vagy szinuszos váltakozófeszültség (áram) is, mely utóbbiak (ismétlődési) frekvenciája (periódusideje) analóg az érzékelt mennyiséggel. Ennek megfelelően kitérés-analóg és frekvencia-analóg mérőhálózatok [27, 28] különböztethetők meg. A kétféle mérőrendszer közül a kitérés-analóg a sokkal elterjedtebb, ezért először és részletesebben a kitérés-analóg módszert tekintjük át.

A kitérés-analóg hálózat legelterjedtebb alakját az 1.10. ábra mutatja be. Az érzékelő-átalakítók kimenő pontjai és a multiplexer bemenő pontjai közötti szakasz a jelátvitelre szolgáló vezetéseken kívül kondicionáló (zavarszűrő, léptékbeállító,

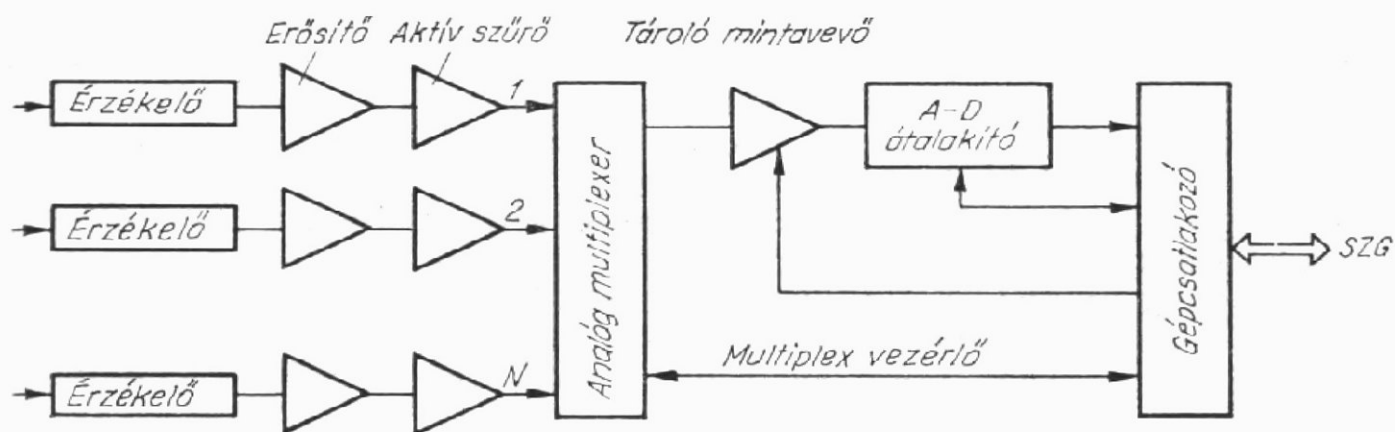


1.10. ábra.
Központi erősítőt és A—D átalakítót alkalmazó analóg mérőhálózat

jelleggörbe módosító) elemeket is tartalmazhat. A multiplexer kimenete és az A—D átalakító bemenete között, a valamennyi mérőcsatornára nézve közös (programozható) erősítő található, mely a jelet az átalakító által igényelt szintre emeli. A gépcsatlakozó a multiplexert előírt ütemben lépteti; a léptetés kötött vagy tetszőleges sorrendű is lehet. Az A—D átalakítónak meghatározott időre van szüksége a bemeneti analóg jel konvertálásához. A multiplexer léptetési üteme a konvertálás időtartamával összhangban kell, hogy álljon. A mérőhálózat 1.10. ábrabeli alakjának — különösen alacsony jelszint esetén — számos hátrányos vonása van, és a vele kapcsolatban felmerülő, gyakran ellentmondó követelmények nehezen egyeztetethetők össze: Költséges és körülményes a mérőcsatorna kondicionálása, fémkontaktusú multiplexert kell alkalmazni, ami korlátozza a letapogatás ütemét, és fellép az apertúraidő problémája is.

Az A—D átalakító adott felbontóképessége csak abban az esetben érvényesül, ha az átalakítás ideje alatt a mért érték relatív megváltozása nem haladja meg az átalakító felbontását. Azt az időt, mely alatt a relatív változás éppen a felbontással megegyező értéket ér el, apertúraidőnek nevezik. Az apertúraidő, mint látható, egyrészt a felbontás mértékének, másrészt a vizsgált jel összetételének (időbeli menetének) függvénye. Az A—D átalakító konvertálási ideje az apertúraidőnél kisebb kell, hogy legyen. Érzékeltetésül: tiszta szinuszos 1000 Hz-es jel és 12 bit felbontású átalakító esetén: az apertúraidő: 40 ns. Látható, hogy viszonylag lassan változó folyamat esetében is, a jel közvetlen multiplexelése igen gyors átalakítót igényel. Az apertúraidőből származó problémák erősítővel kombinált tároló-mintavevő segítségével hidalhatók át. A tároló mintavevő az 1.10. ábra erősítőjének helyére kerül; rövid kapuzási ideje alatt rögzíti a mért mennyiség pillanatértékét, majd a felvett értéket azonos szinten tartva, a jelet viszonylag hosszú időn át bocsátja az A—D átalakító bemenetére.

A kondicionálási problémák csökkentése, a jelbegyűjtő vezetékezés egyszerűsítése érdekében alkalmazzák a mérőhálózat 1.11. ábrán bemutatott felépítmódját.



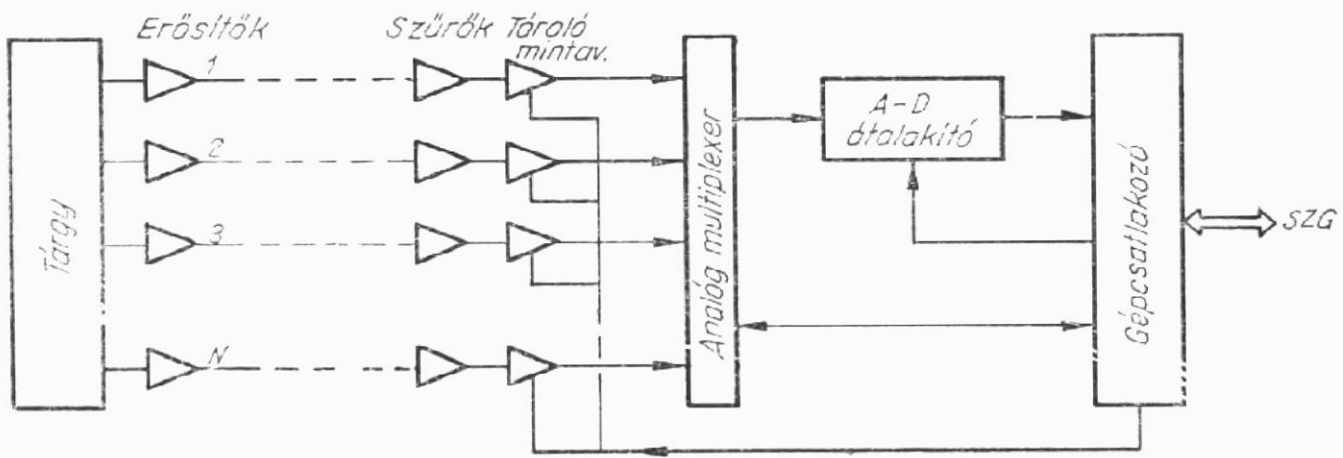
1.11. ábra.

Méréspontonként erősítőt és szűrőt alkalmazó analóg mérőhálózat

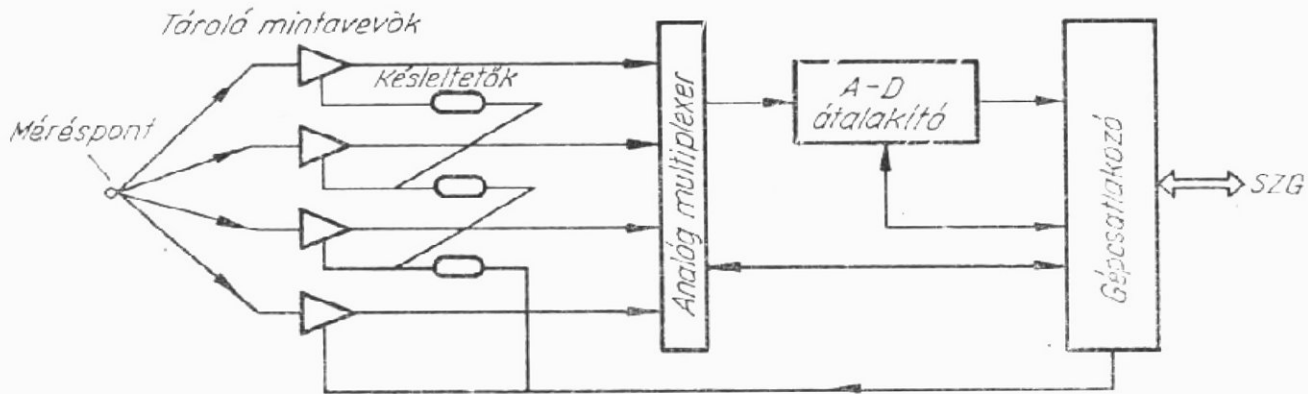
Valamennyi jelcsatornában az érzékelő-átalakító mellett erősítő, a multiplexer mellett aktív szűrő helyezkedik el. Az átvitel nagy jelszinten történik; a maradék zajtartalmat szűrők csökkentik. Az erősítőnek az érzékelőkhöz, az érzékelőkkel együtt a tárgyhoz való rendelése voltaképpen a mérőhálózat elemeinek elosztását jelenti. Az 1.9. ábrán a tárgy és a mérőhálózat határolóvonalának helyzete éppen erre a lehetőségre utal, ti. bizonyos esetekben a mérőhálózat egyes elemeit mérés technikai célszerűségi okokból a tárggyal eleve összeépítik. A csatornánkénti mérőáramkört elemek számának szaporítására, elsősorban az integrált áramkörti technológia fejlődésének eredményeként, az áramkörök egységköltségének csökkenése folytán nyílt lehetőség.

Az esetben, amikor nem csak az egyes pontokon felvett értékek, hanem az egyes mennyiségek egymáshoz viszonyított mérési pontossága is követelmény, egyidejű rögzítést kell végezni (hely szerinti multiplexelés). Egyidejű rögzítés a valamennyi mérőcsatornában elhelyezett tároló mintavevő segítségével oldható meg (1.12. ábra). Más körülmények között, gyors lefutású folyamat rögzítésére, egy mérőponthoz egynél több tároló mintavevőt (időmultiplexert) alkalmaznak. (1.13. ábra).

Az egyes csatornában elhelyezett erősítők nem csupán a jel/zaj viszony javítása szempontjából jelentenek előnyt; erősítésük változtatható, miáltal az egyes csatornák léptékhelyes beállítása elvégezhető. A csatornában nem csak lineáris erősítők, hanem szükség esetén linearizálást, többszörözést vagy más korrekciót végző analóg



1.12. ábra.
N-számú méréspont egyidejű érzékelésére szolgáló analóg mérőhálózat



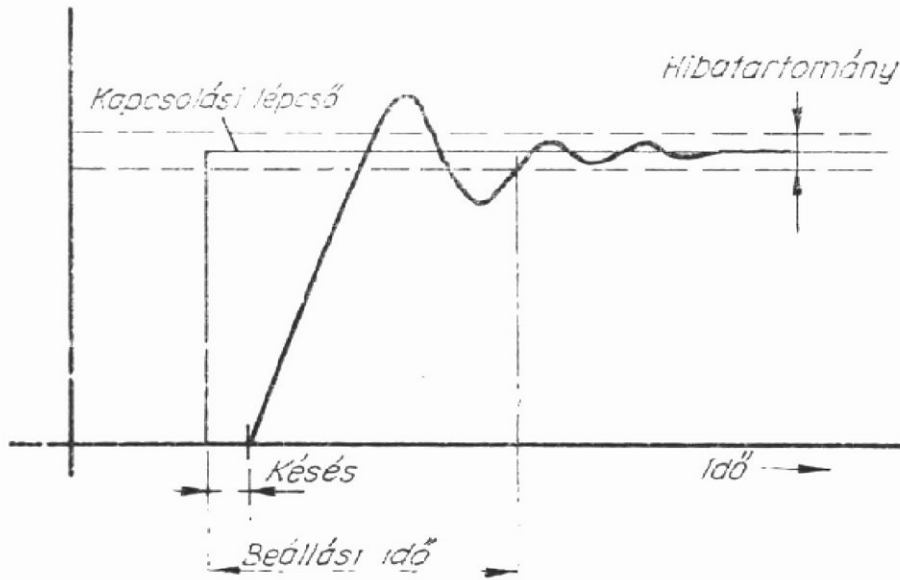
1.13. ábra.
Gyors folyamat időbeli lefutását vizsgáló mérőhálózat

szorzók is alkalmazhatók. Ha az érzékelő által szolgáltatott jel közös feszültsége jelentékeny, differenciaerősítőt alkalmaznak. Amennyiben a hálózat egyetlen, a multiplexert követő differenciaerősítőt tartalmaz, kettős (esetleg hármas) multiplexert kell alkalmazni, és a kondicionálási problémák fokozottabb mértékben lépnek előtérbe. A csatornánként alkalmazott szimmetrikus bemenetű, jelentékeny közös feszültségnyomással rendelkező egyikemenetű csatornaerősítők a kondicionálási és multiplexelési problémákat egyaránt leegyszerűsítik.

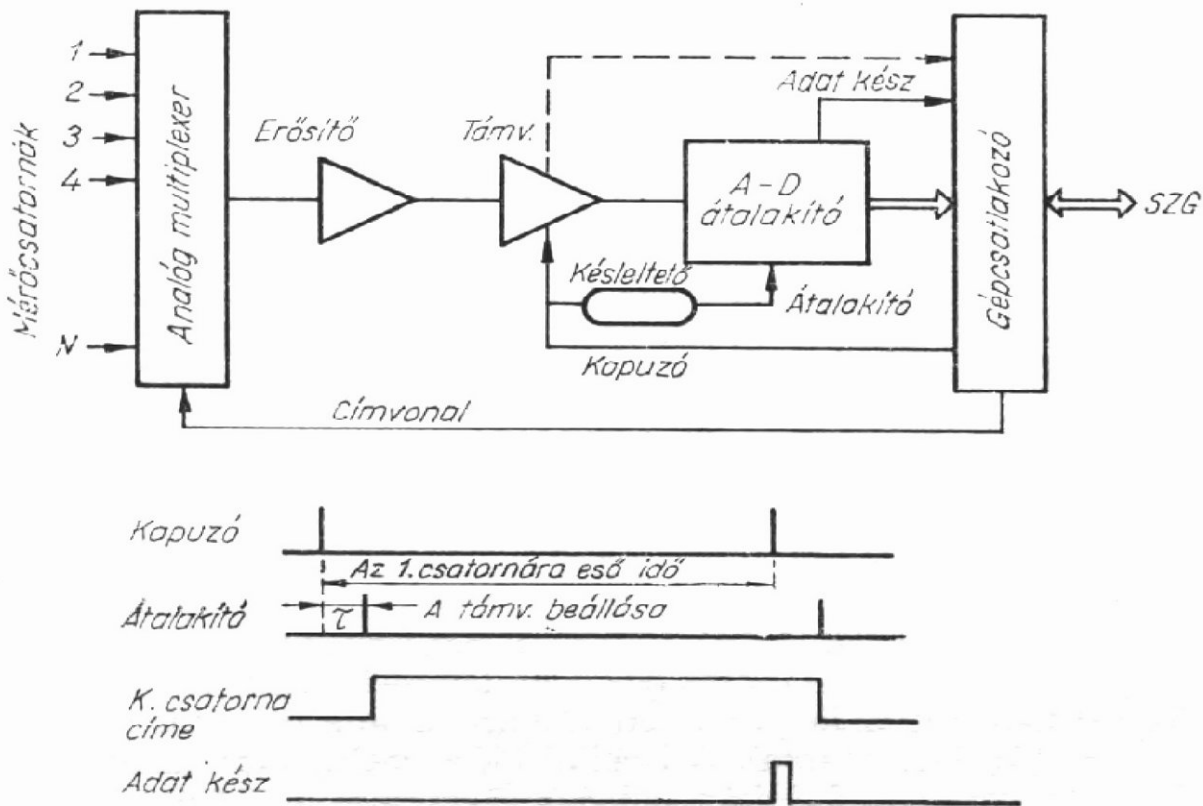
Az analóg adatgyűjtő hálózatok igen fontos jellemzője: a mérőcsatornák beállási ideje. Az átkapcsolással együttjáró tranziensnek bizonyos előírt tartományból már nem szabad kilépnie, amikor az A—D átalakítás megkezdődik. A kapcsolási lépcső frontjától, a tranziensnek a hibatarományba való érkezéig eltelt időt nevezik beállási időnek. Beérkezésen azt az időpontot értik, amely után a tranziens a hibatarományt már nem hagyja el (1.14. ábra). A beállási idő a multiplexer, az erősítő, az A—D átalakító és a csatornák felépítésmódjának függvénye. A $\pm 0,01\%$ hibatarományhoz tartozó, tipikusan $1 \mu\text{s}$ beállási idő, 300 ns -re csökkenthető az ún. előcímezési módszer segítségével, melynek lényege az 1.15. ábra alapján világítható meg. A kapujel hatására a tároló mintavevő rögzíti az erősítő kimenetén észlelt szintet. A mintavétel ideje, τ múlva az átalakító megkezdte munkáját, s ugyanakkor a soron következő kapcsolóállás megnevezésével a címvonal máris megkezdte a multiplexer átkapcsolását. Mire a konverter munkájával elkészül, és sor kerül a következő csatorna mintavételére, a tranziens az erősítő, illetve a tároló mintavevő körében már a szükséges mérvűre csillapodik.

A digitális jelátvitel az analóg jelátvitellel szemben ismert előnyökkel rendelkezik, amelyek alapján célszerűnek bizonyulhat az egyes mérőcsatornáknak nem

csak erősítőket, hanem a tárgy mérőpontjaihoz csatolt A—D átalakítókat is alkalmazni (1.16. ábra). Míg a fejlődés korábbi szakaszában a multiplexelés célja éppen az, hogy a számos mérőpontra kiterjedő mérést egyetlen mérőblokk segítségével



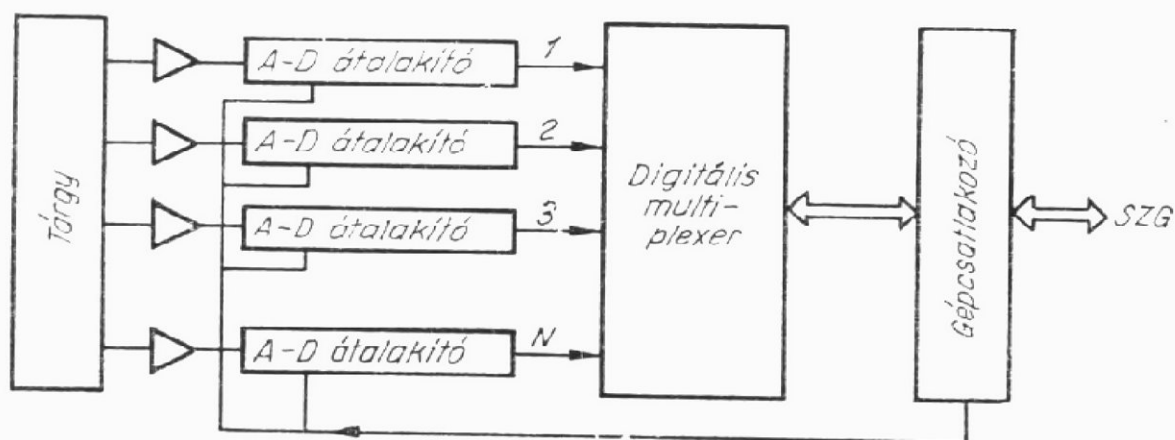
1.14. ábra.
Kapcsolási tranziens és beállási idő



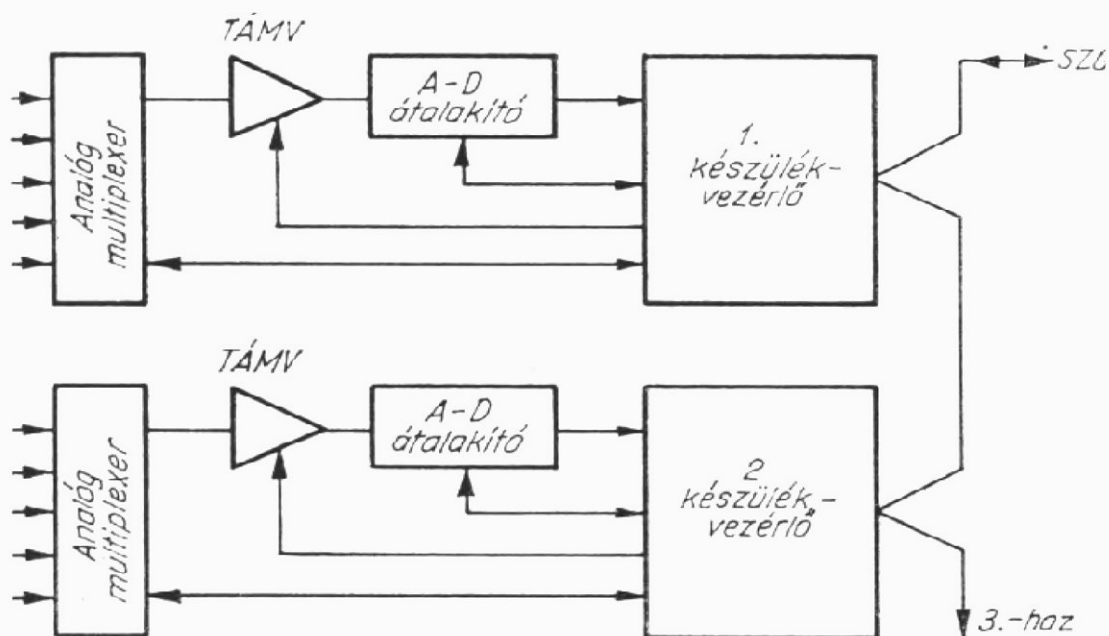
1.15. ábra.
Beállási idő csökkentése a multiplexer előcímezésével;
TÁMV — tároló mintavevő

oldja meg, az integrált áramköri technológia előrehaladtával az individuális erősítő-kön túl A—D konverterek is gazdaságosan alkalmazhatók. A mérőcsatornáknak végzett A—D átalakítás esetén az adatoknak a számítógéphez való továbbítása digitális multiplexer segítségével történik.

Analóg és digitális multiplex-rendszer közötti átmeneti megoldást is alkalmaznak. (1.17. ábra). Ez esetben az összes analóg csatornát csoportokra osztják, és minden



1.16. ábra.
Méréspontonként A—D átalakítót alkalmazó mérőhálózat



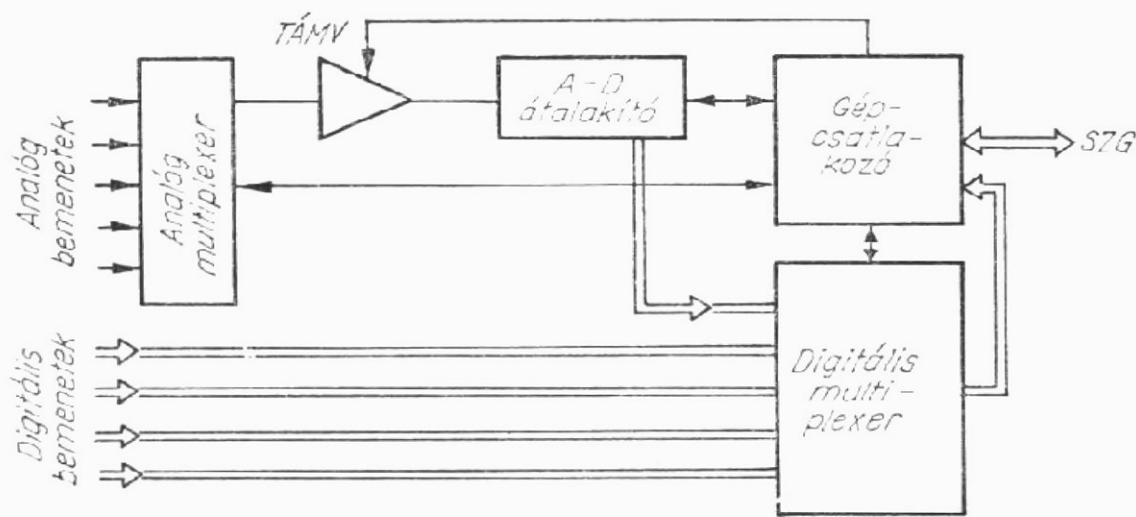
1.17. ábra.
A méréspontok elosztása több mérőblokk között;
készülékvezérlők együttese mint digitális multiplexer

csoporthoz külön analóg multiplexert, tároló-mintavevőt és A—D átalakítót, valamint önálló készülékvezérlőt rendelnek. A digitális multiplexer szerepkörét a számítógép látja el, az egyes készülékvezérlőkkel folytatott kommunikáció útján.

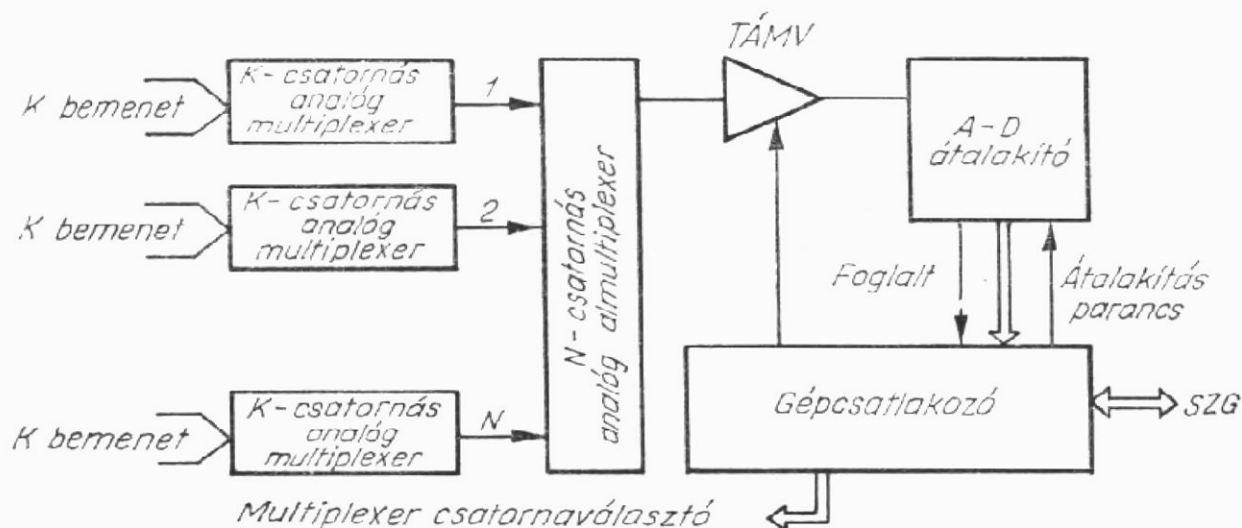
Analóg és digitális multiplexerek együtt is alkalmazhatók (1.18. ábra). A digitális multiplexer minden bemenetére analóg multiplexelt A—D átalakítót, de digitális kimenetű mérőpontok is csatlakoztathatók, pl. az 1.16. ábrán látható módon. A digitális multiplexer bemenetére csatlakozhatnak a közvetlen digitális kimenetű érzékelő-átalakítók, valamint a digitális állapotjelzők együttese is, melyek — esetenként — az analóg kimenetű érzékelőkkel együtt fordulhatnak elő.

Mind az analóg, mind a digitális multiplexerek általában modulárisan építhetők fel. A multiplexer modulmérete és a modulok egymáshoz csatlakoztatása rendszerenként változhat. Az 1.19. ábrán pl. K -bemenetű analóg multiplexerek csatlakoznak N -bemenetű analóg almultiplexerhez ($K=32$, $N=8$ esetén a mérőpontok száma max. 256-ig terjeszthető ki).

A számítógépes analóg adatgyűjtő rendszer gyakorlati példjaként távolsági földgázvezeték-rendszerhez tartozó, gázturbina hajtású kompresszor gyártása



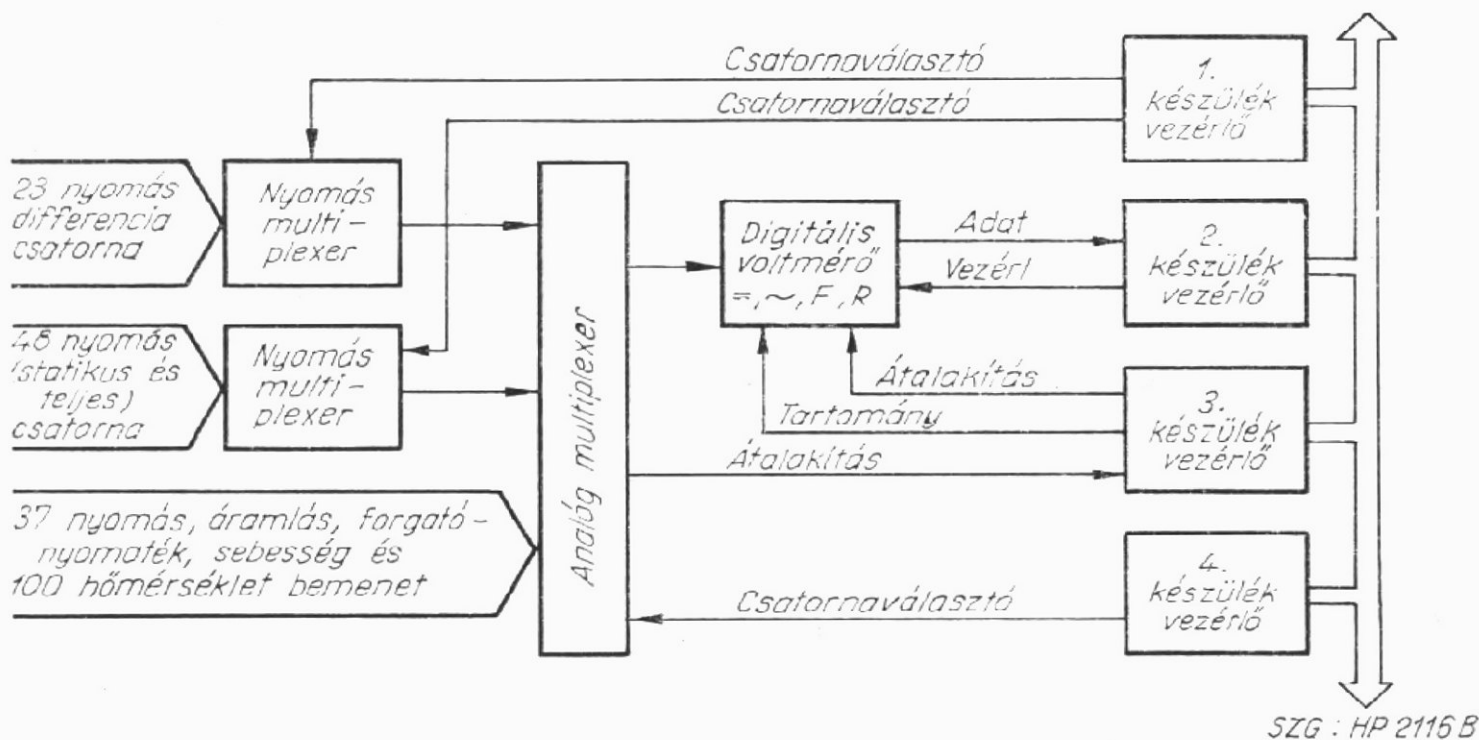
1.18. ábra.
Analog és digitális multiplexer együttes alkalmazása



1.19. ábra.
Analog multiplexerek többszintű moduláris rendszere

kapcsán alkalmazott mérőrendszert (DRESSER INDUSTRIES Machinery Group Gas Turbine Division, Houston, Texas) mutatunk be (1.20. ábra). A gépegység üzemét összesen mintegy 200 érzékelő segítségével ellenőrzik, melyek között nyomás, áramlás, sebesség, forgatónyomaték- és hőmérsékletmérők találhatóak. A nyomásmérés részben individuális érzékelők, részben pedig két nyomásmultiplexer útján történik. A nyomásmultiplexer nem más, mint pneumatikus letapogató, mely egyetlen érzékelő-átalakítót (nyúlásmérőt) tartalmaz; ez utóbbira a nyomáscsatornák időbeli egymás utánban oly módon kapcsolódnak rá, mint a villamos kimenetű érzékelők feszültsége (árama) a villamos multiplexer közös pontjára. A számítógép a digitális voltmérő által nyújtott adatokat mindenekelőtt technikai egységekre számítja át, majd a konvertált adatok a dátummal, időponttal, vizsgálatszámmal és csatornaszámmal együtt mágnesszalagra kerülnek. A mérőrendszer kezdeti változatánál a mágnesszalagra rögzített adatokat nagyszámítógép értékeli ki. Fejlettebb változata — elképzelés szerint — bizonyos adatredukciót, valamint a működési analízis egy részét is a turbina futási idejében, az adatgyűjtéssel párhuzamosan végzi el; a teljes analízis továbbra is a nagygépre hárul.

A *frekvencia-analog* jel a mért mennyiségre vonatkozó információt nem a jelnek valamely nívótól való kitérésében, hanem az egységnyi időre eső nullátmenetek számában vagy két nullátmenet között eltelt időben hozdozza; amplitudó-zavarokra ez az átviteli forma kevésbé érzékeny; az adatátvitel rosszminőségű csatornán is

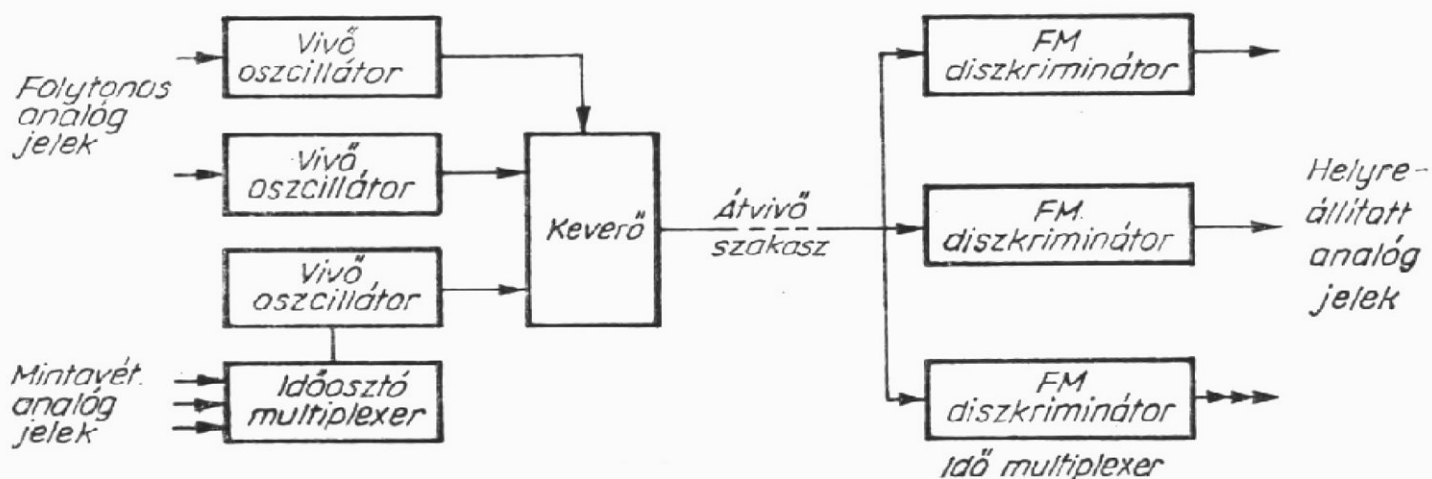


1.20. ábra.
Komplex analóg adatgyűjtő hálózat

elvégezhető. Kontaktusproblémák, munkapont-eltolódási jelenségek itt nem játszanak szerepet. A frekvencia-analóg jel átvitele nagyobb sávszélességet és hosszabb átviteli időt igényel, mivel azonban a sávszélesség-igény nem haladja meg a kitérés-analóg átvitelnél általánosságban használt vonalak lehetőségeit, a sávszélesség-követelmény nem tekinthető hátránynak. Jellemző, hogy míg a kitérés-analóg átvitel mérési hibája 0,5...1% körül mozog, addig a frekvencia-analóg átvitel módban a mérési hiba mindössze 0,5...0,1%. A kitérés-analóg érzékelésmód — tipikusan — 1...100 ms, a frekvencia-analóg 1 s mérési időt igényel. Ez utóbbi relatíve hosszú mérési idő lecsökkenthető, ha az információ leválasztása frekvenciamérés helyett periódusméréssel történik (mérési idő 1...10 ms). A periódusmérés a frekvenciamérésnél zavarérzékenyebb, pontossága azonban növelhető, ha nem egy, hanem több ciklus átlagát veszik a mérés alapjául. A zavarérzékenység a jelfront meredekségének növelése révén is csökkenthető (szinuszos helyett impulzus jelforma), ezzel azonban fokozódik a sávszélesség-követelmény. A zavarelnyomás szempontjából a frekvencia-analóg konverzió további előnyös tulajdonsága még az is, hogy a jelamplitúdó az átviteli szakasz elején tetszőleges szintre emelhető (pl. 10 V), melyhez viszonyítva a zavaró effektusok háttérbe szorulnak. A frekvencia-analóg jel váltakozó áramú átviteli csatornát igényel (vivőfrekvenciás vagy egyszerű telefonvonal), mely egyidejűleg felhasználható az érzékelő-átalakító által igényelt egyenáramú segédenergia átvitelére. A frekvencia-analóg jel digitalizálása egyszerű; számlálástechnikán alapul, amely bonyolult eszközök nélkül is jelentős pontosságot nyújt.

A számos mérőponttól származó frekvencia-analóg jel a mérőhálózathoz külön-külön csatornán érkezik, a jelek frekvenciatartománya jellegzetesen 10 KHz alatt van; a csatornák a kitérés-analóg csatornához hasonlóan multiplexelhetők egyetlen digitalizáló egység bemenetére. Az esetben, amikor a tárgy a számítógéptől és a számítógép mellett levő mérőhálózati egységektől távolabb helyezkedik el, nem gazdaságos az egyes mérőpontoktól származó jelek mindegyikét külön csatornán vinni a mérőhálózatra; célszerűbb az érzékelés helyén frekvencia-osztásos multiplexelést végezni, és az összes szimultán jelet közös csatornán átvinni a mérőponthoz. (1.21. ábra) A frekvenciaosztásos multiplexelés esetében minden egyes jelforráshoz

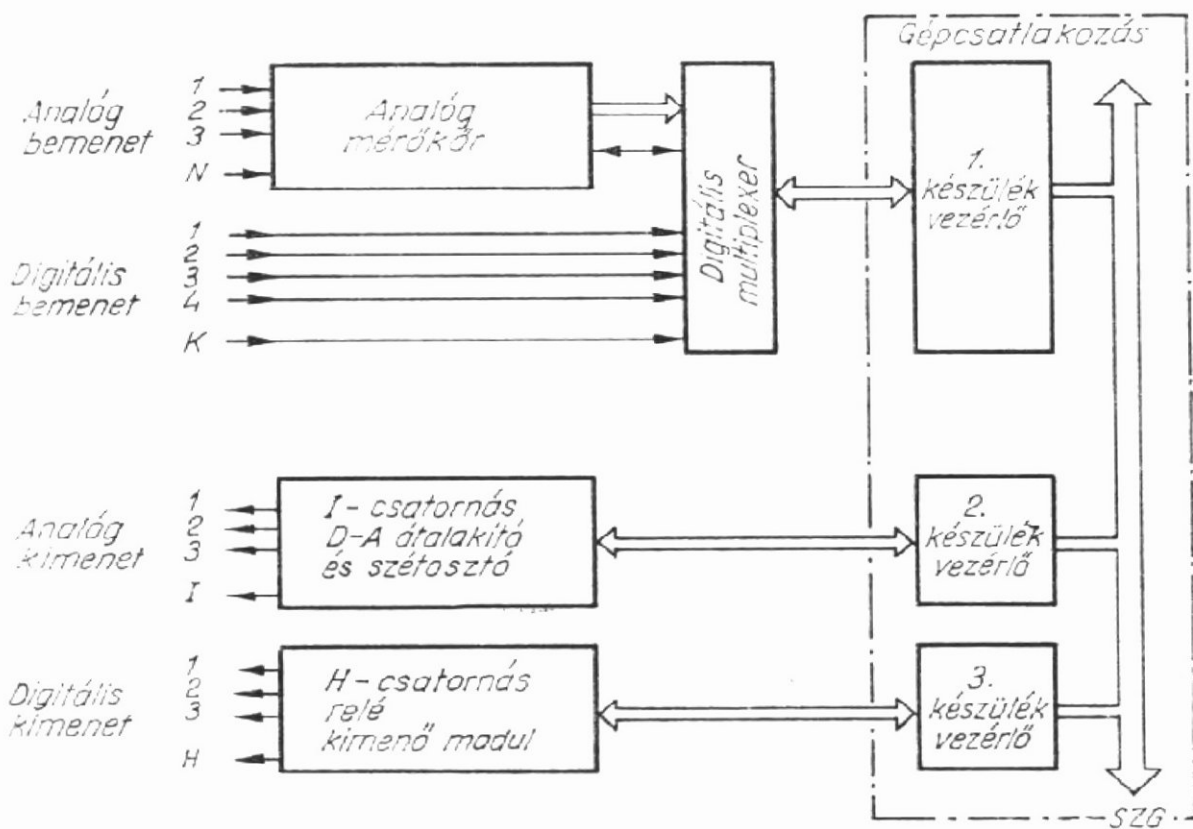
külön, egymást át nem lapoló frekvenciatartományt kell rendelni; minden egyes jelforráshoz saját vivőfrekvencia tartozik, s e vivőfrekvenciát modulálja a (kondicionált) jel. A modulációs sávokat szigetelésávok választják el egymástól, az áthallás lehetőségének elkerülése érdekében. Végeredményként egy sor jelmodulált vivőfrekvencia adódik. A moduláció módja számos; legelterjedtebbek a feszültség—frekvencia átalakítók. Átvitel előtt az összes jelforrás modulált jelét keverő foglalja össze; a közös jel egyetlen vezetéken keresztül jut a mérőközpontba. A közös jelről az egyes érzékelők jelét az FM diszkriminátor bemenetén elhelyezkedő sávszűrők választják le. A sávszűrőket követő FM demodulátorok végül a jelet feszültséganalóg alakban állítják elő, mely a továbbiakban a már ismertetett kitérés-analóg módszerek valamelyikével dolgozható fel.



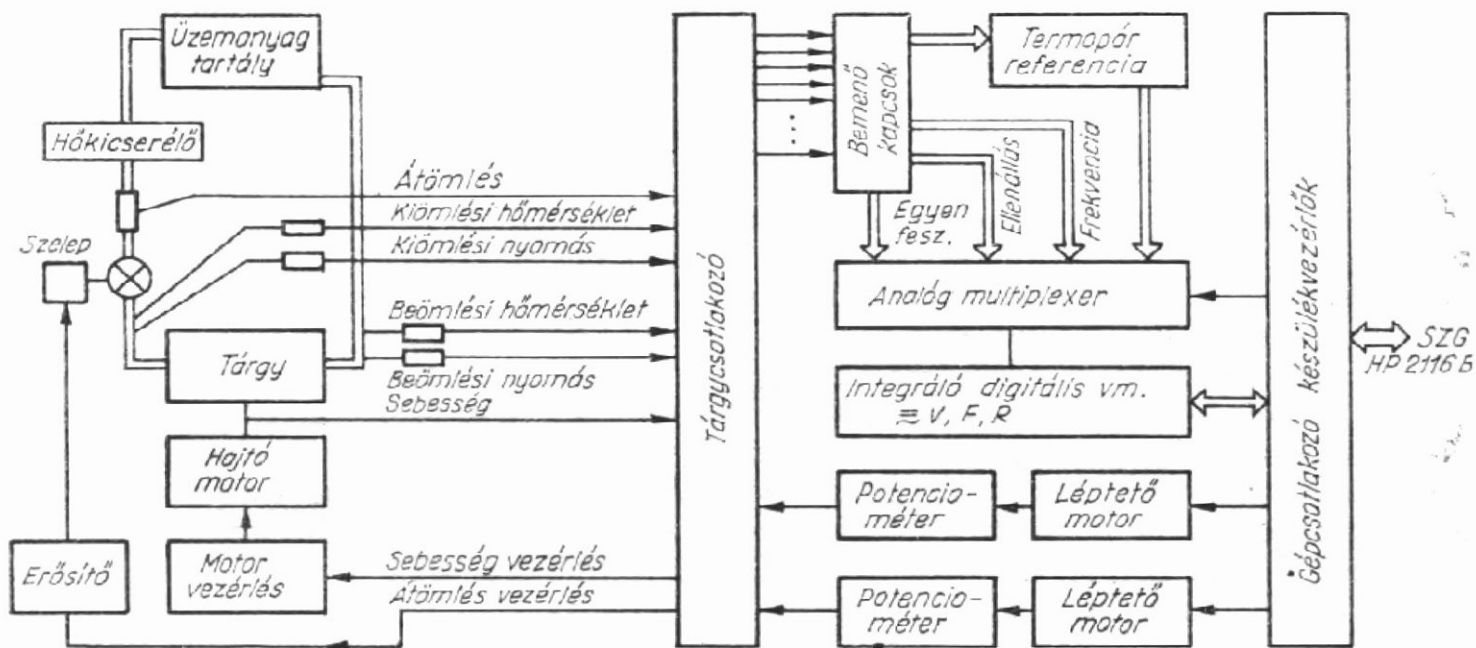
1.21. ábra.
Frekvencia-analóg mérőhálózat bemeneti szakasza

A mérésadatgyűjtő *aktív analóg mérőrendszerre* lép elő, ha a tárgyra hatást gyakorló áramköri kimenő egységeket is tartalmaz. Az adókörök felépítmódja nem mutatja fel az általánosságoknak azokat a jegyeit, melyek a vevőkörökkel kapcsolatban kimutathatók. Az 1.22. ábra analóg és digitális adatgyűjtő hálózattal, valamint analóg és digitális (alapjel/szabályzó) kimenettel rendelkező aktív mérőrendszert mutat be. Minden egyes analóg kimenő csatornához egy a készülékvezérlő által címezhető tárolóregiszter, valamint a regiszterhez csatlakozó D—A átalakító tartozik. A készülékvezérlő az egyes csatornák regisztereit kötött vagy tetszőleges sorrendben tölti fel. Az analóg kimenetek egymástól galvanikusan függetlenek. A H-csatornás relé-kimenetű modul H számú (kétértékű) logikai szintet állít elő kontaktuspárok útján.

Az aktív analóg mérőrendszert általánosan alkalmazzák pl. a repülőgépiparban. Az 1. 23. ábra repülőgép- és rakéta-vezérlőelemek (üzemanyag-pumpák, üzemanyag-szabályzók, üzemanyag-vezérlőszelepek, szükségállapot üzemanyag-szabályzók, elektrohidraulikus rendszerek stb.) gyártásában alkalmazott mérőrendszer vázlatát mutatja be (Colt Industries; Chandler Evans Control Systems Div.). A felsorolt vezérlőelemeket különböző üzemi állapotokban (munkapontokban) ellenőrzik, jelleggörbéiket felveszik, miközben bizonyos változókat állandó értéken tartanak. A vizsgálatok — a tapasztalat szerint — kézi vezérléssel nem nyújtanak kielégítő pontosságot és reprodukálhatóságot; elsősorban ezért kellett áttérni a mérőrendszer számítógépes irányítására. A rendszer hőmérséklet-, nyomás-, áramlás-, szögsebesség-, forgónyomaték-, elmozdulás- és feszültségértékeket mér. A hőmérsékleti érzékelők feszültség szintje mV, a nyomás- és forgatónyomaték-érzékelőké 0...5 V nagyságrendű; az áramlás-érzékelő 50...1500 Hz frekvencia-analóg jelet ad. A rendszer vezérlő kimenetein át, motorsebességet, üzemanyag-áramlást, nyomást,



1.22. ábra.
Aktív mérőrendszer mérőhálózata

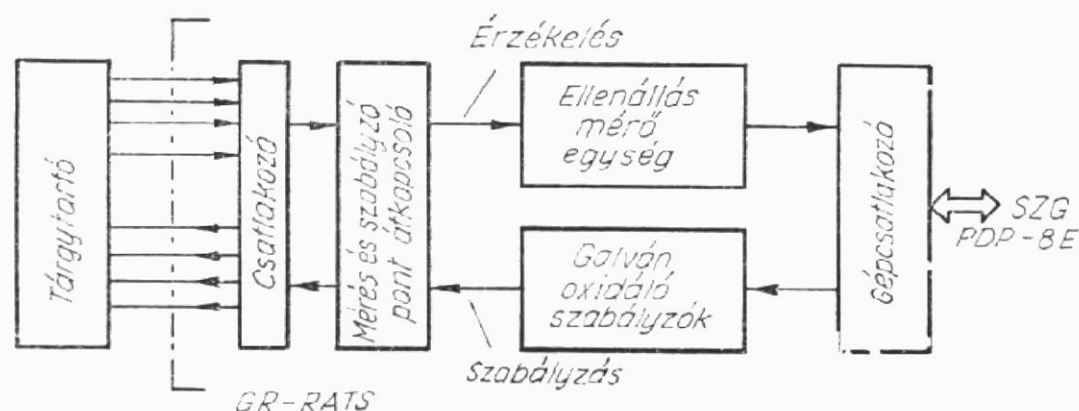


1.23. ábra.
Repülőgépiparban alkalmazott aktív rendszer mérőhálózata

hőmérsékletet stb. állít program szerint. A D—A átalakítás léptetőmotorok és potenciométerek alkalmazásával történik. A potenciométerek csúszkájáról nyert analóg feszültség definiálja a meghajtó motor sebességét, valamint a különböző szeleppállásokat.

Közvetlen szabályzó analóg mérőrendszert mutat be az 1.24. ábra. A rendszer precíziós tantállenállások gyártásának irányítására szolgál, s mint ilyen, a technológiai mérőrendszer jellegzetes képviselője. A tárgyartón összesen 240 ellenállás

helyezkedik el, melyek közül a mérés- és szabályzópont-átkapcsoló egyidejűleg tizenkettőt különít el. Az elkülönített ellenállások mindegyikére individuális mérő-híd és individuális szabályozókör kapcsolódik. Az egyes ellenállások oxidációs fo-

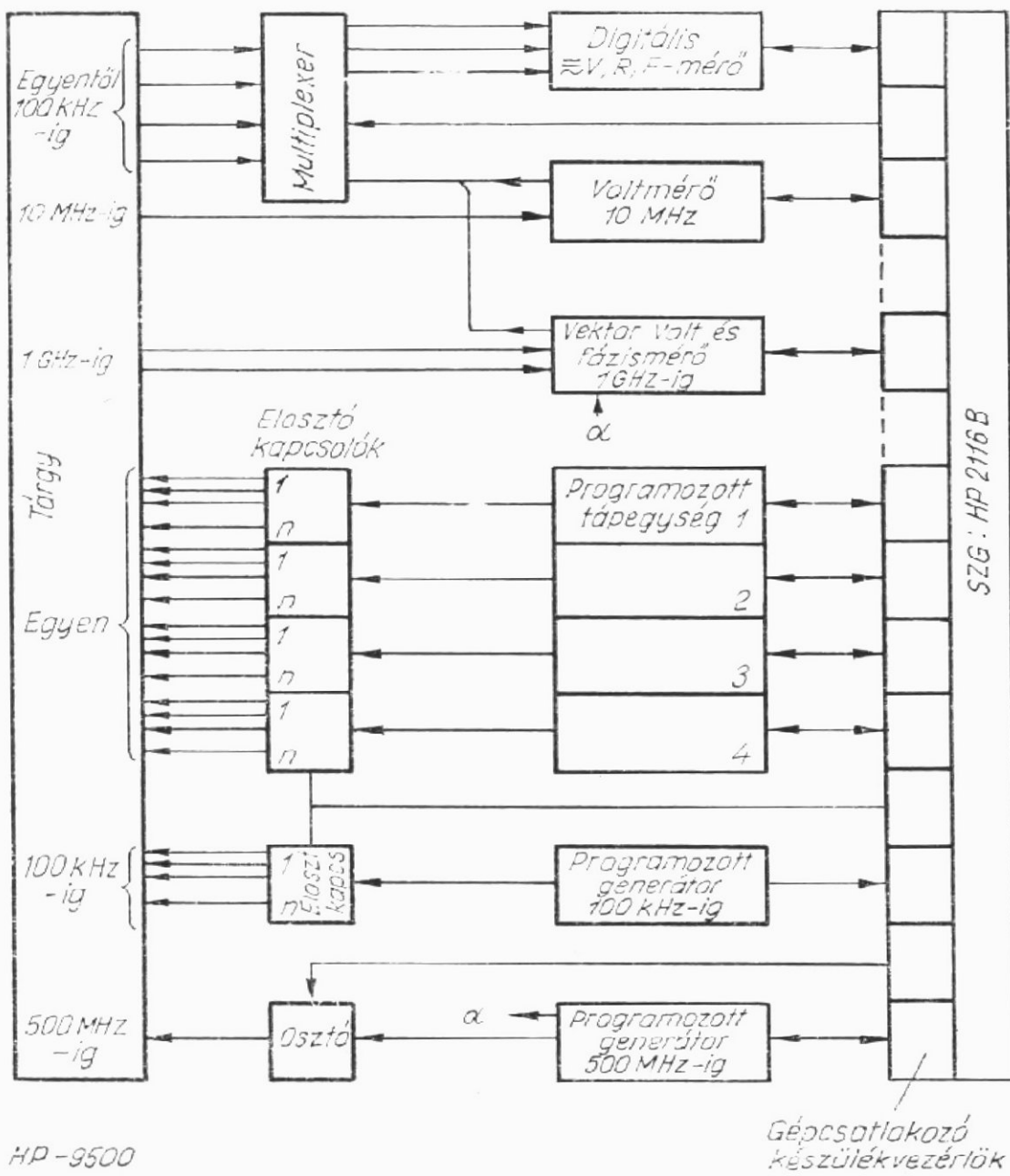


1.24. ábra.
Közvetlen szabályzórendszer mérőhálózata

lyamatát — a mért ellenállásértékek alapján — a számítógép irányítja. Az ellenállás-mérő egység tizenkét, Kelvin csatlakozású, digitális kimenetű, automatikus mérő-hidat tartalmaz; ugyancsak tizenkettő a programozható áramforrásokra alapozott szabályozókörök száma is. A munkafolyamat megindítása előtt a megadott adatok alapján a rendszer ellenőrzi, hogy a kiválasztott ellenállások kiindulási értéke a trimmelhetőség tartományába esik-e; ezt követően megindítja a munkafolyamatot, egyenként (és szimultán) beállítja az értékeket, végellenőrzést tart, végül az átkapcsoló segítségével a következő ellenálláscsoportra kapcsol át. Az adatgyűjtő rendszerek ismertetett formáival szemben — amelyeket számos mérőpont és egyetlen mérőegység jellemez — itt minden szimultán mérőcsatornának külön mérő- és szabályozóköre van. E szimultán rendszer a húsz ($20 \times 12 = 240$) ellenálláscsoportra vonatkozóan természetesen továbbra is multiplexelődik.

A készülékek szintjén moduláris aktív analóg mérőrendszer jellegzetes képviselője az analóg áramkörök vizsgálatára szolgáló, s az 1.25. ábrán bemutatott berendezés. A méréstartomány egyenfeszültségtől 1 GHz-ig terjed. A frekvenciasáv alsó területét kombinált digitális voltmérő szolgálja ki, amely feszültségmérésen kívül ellenállás- és frekvenciamérést is végezhet (ugyanezen voltmérő szerepel az 1.20. és 1.23. ábrán bemutatott rendszerekben is). Feszültségmérőként felbontóképessége a legalacsonyabb mérési tartományban $1 \mu\text{V}$, felső méréshatára 1000 V ; a méréstartomány automatikusan vagy program szerint változtatható. A frekvenciatartomány 100 KHz -ig terjed. A mérésidő 25 ms . Ellenállásmérőként $0,01 \Omega$ -tól $10 \text{ M}\Omega$ -ig, frekvenciamérőként $5 \text{ Hz} \dots 200 \text{ KHz}$ között mér. Külön voltmérő szolgál a 10 MHz -ig terjedő frekvenciatartományban végzendő mérésekre, míg az $1 \dots 1000 \text{ MHz}$ közötti sávhoz vektoriális volt- és fázismérő tartozik, mely utóbbival nemcsak feszültség, hanem fáziseltolás is vizsgálható (a fázismérés kapcsán α referencia kapcsa a megfelelő jelgenerátor hasonló nevű kapcsáról nyeri a nulla fázisszögű feszültséget). Mind a 10 MHz -es voltmérő, mind a vektoriális volt- és fázismérő egyenfeszültségű kimenetet ad, amely a multiplexeren keresztül a digitális voltmérő bemenetére kerül digitalizálás céljából.

A mérőrendszer adótípusú egységeinek frekvenciatartománya $0 \dots 500 \text{ MHz}$ -ig terjed. Egyenfeszültségű forrásként négy programozható tápegység szolgál, melyek a tárgy tápkörének, ingerjel, alapjel vagy szabályzójel adójának szerepkörét egyaránt elláthatják. A tápegységek áramkorlátja, éppúgy, mint kimenő feszültségük, program útján választható. A kimenő kapcsok elosztó kapcsolókon át csatlakoznak a



1.25. ábra.
Készülékek szintjén moduláris aktív analóg mérőrendszer

tárgy vizsgálati pontjaira. Váltakozó áramú ingerjel előállítására 100 KHz, ill. 500 MHz felső frekvenciahatárú jelgenerátor szolgál.

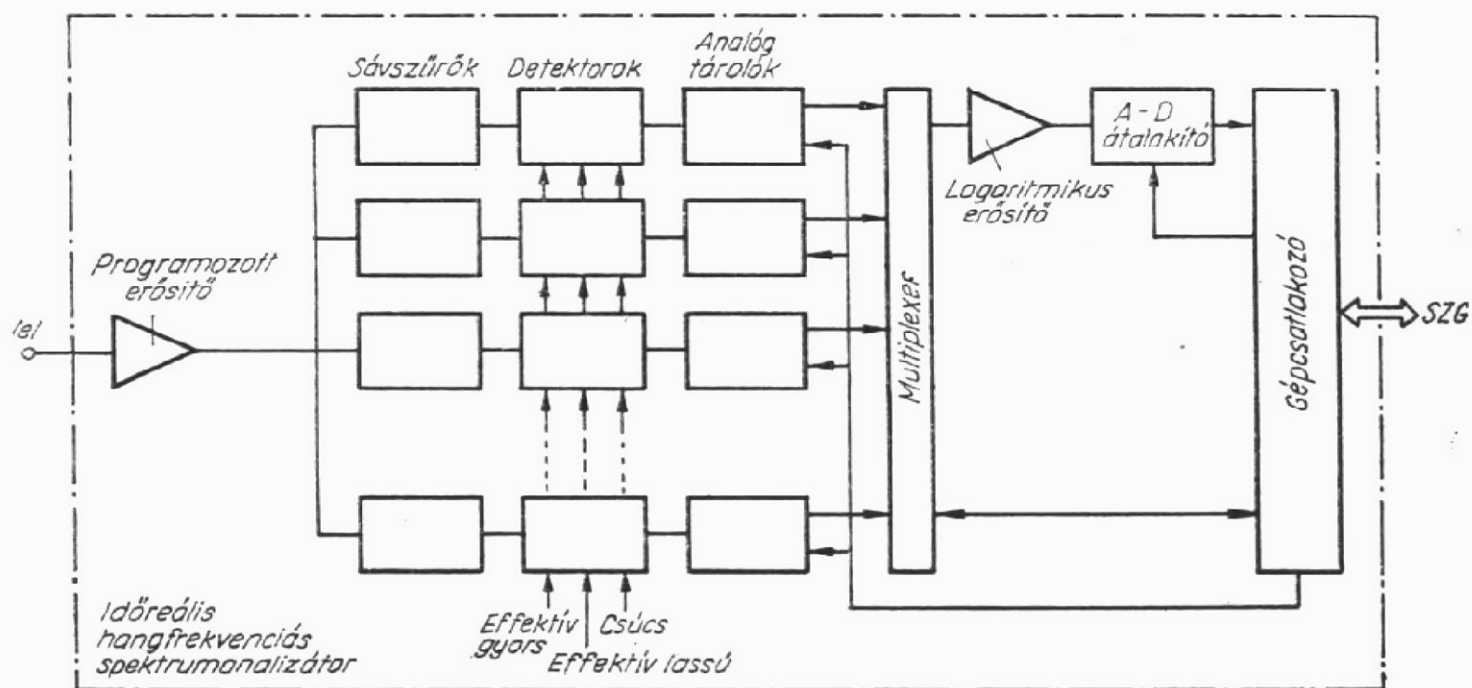
A rendszernek a tárgyhöz való csatlakozása individuális módon történik. A kapcsolórendszer lehetővé teszi, hogy a mérés lebonyolítása során a vevők bemeneteit és az adók kimeneteit bizonyos, a kapcsolók lehetőségei által megengedett variációs tartományon belül a tárgy különböző vizsgálati pontjaihoz lehessen rendelni.

A közvetlen analóg be/kimenetek mellett, a rendszer az ábrán nem látható, korlátozott digitális be/kimeneti lehetőségekkel is rendelkezik. Duplex regiszteregység szolgáltat 16-bites paralel bemenetet és 16-bites paralel kimenetet, 16-bites relélap pedig szigetelt paralel kimenetű digitális vezérlőjelet szolgáltat. A digitális kimenetek különböző vezérlési feladatokat látnak el, mint amilyenek pl. az automatikus adagolók működtetése során merülnek fel.

A rendszer modulárisan bővíthető (vagy szűkíthető); frekvencia-, ill. időmérő, torzításmérő, különböző nemszinuszos jelgenerátorok, impulzusgenerátorok, hullámalak-analizátor stb. csatlakozhat a mérőkészülékek készletéhez. A rendszer egy e célra szolgáló kiegészítő egységgel digitális, valamint analóg/digitális áramkörök

vizsgálatára is alkalmassá tehető, mely egység leírására a digitális mérőrendszerek tárgyalásakor térünk ki.

Az *integrált adatgyűjtő mérőhálózat* egy megtestesítője az időreális hangfrekvenciás spektrumanalizátor (1.26. ábra). Az egyetlen bemenetre érkező hangfrekvenciás



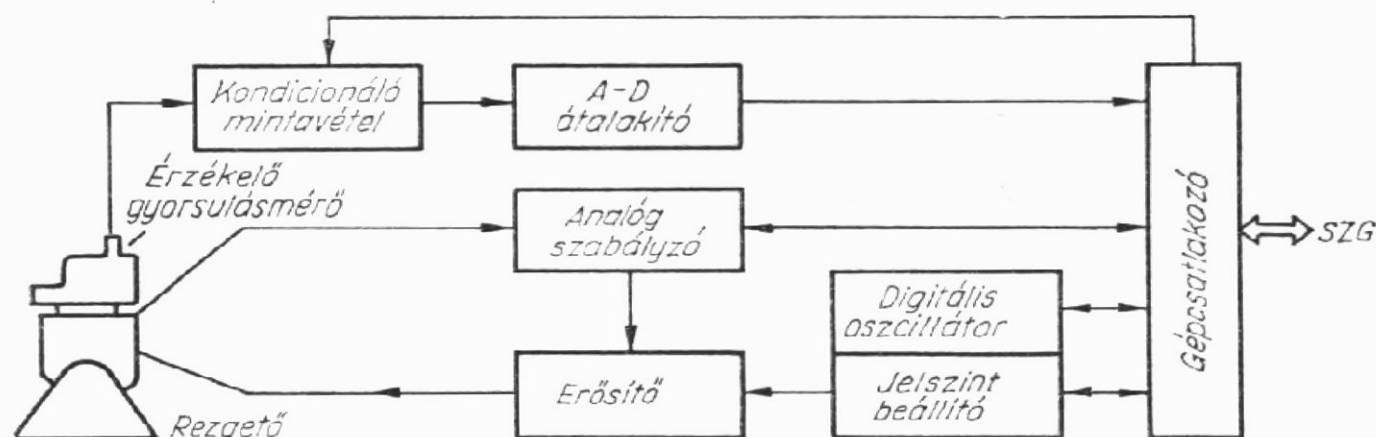
1.26. ábra.
Integrált analóg mérőhálózat

ciás jel (mely származhat mikrofontól, más átalakítótól vagy pl. mágnesszalagról) a programozás útján beállítható erősítőn át hangfrekvenciás sávszűrőkre jut. Az egyes szűrők sávközép-frekvenciáit egymáshoz képest oly módon eltolva választják meg, hogy $1/3$ -oktávós sávjaik a teljes mérési tartományt lefedjék. A sávszűrőket, az egyes csatornában program szerint választhatóan, csúcs- vagy effektív értéket mérő detektorok követik. A detektorok jeleinek a tároló-mintavevőkkel rögzített értékeit a multiplexeren át az A—D átalakító sorban digitalizálja; a digitalizált értékeket a gépcsatlakozó a számítógép tárolójába továbbítja feldolgozásra. A vázolt rendszer számos alkalmazási lehetősége közül a járművek zajosságának mérését, valamint a dízelmotor-vizsgálatot említjük meg. A dízelmotor különböző jellegzetes üzemiállapotaihoz és hibaállapotaihoz tartozó referenciaspektrumokat a számítógép tárolójában tartva, a vizsgált motor zajspektruma ezekkel összehasonlítható, a motor üzemének gyors kiértékelése érdekében.

Az 1.27. ábra *analóg szabályozóval kombinált aktív számítógépes rezgésvizsgáló* mutat be. A mérőrendszer vobbulációs technikával dolgozik. A számítógép — egy a gépcsatlakozóhoz tartozó órajel-generátor időzítésének megfelelően — program útján vezeti át a digitális oszcillátort (és a rezgetőt) az előírt frekvenciatartomány mérési pontjain. Az egyes bemenő értékekhez tartozó kimenő értékek mérése a gyorsulásmérő és az A—D átalakítóval kiegészülő kondicionáló-mintavevő egység útján történik. Az erősítő révén a rezgető bemenő jelének amplitúdóját korrigálva, az analóg szabályozó egység gondoskodik a rezgető amplitúdó megfelelő szinten tartásáról. A szintet a számítógép definiálja, másrészt a szabályozó az általa végrehajtott korrekció értékét visszajelenti a számítógépnek. E korrekciós érték alapján módosítja a gép az érzékelőn át nyert mérési adatokat.

A számítógépes analóg mérőrendszer további jellegzetes képviselője az idősorozat-analízis módszere alapján működő *dinamikus rezgésvizsgáló*, mely a számítógépes

gyors Fourier-transzformáció algoritmusának reciprok tulajdonságán alapul. A rezgésvizsgálatokkal kapcsolatban az a probléma merül fel, hogy általános esetben a rezgető és a rezgetett test együttesének átviteli függvénye ismeretlen, ezzel együtt arra vonatkozó támpont sem áll rendelkezésre, milyen bemenő jelet kell alkalmazni a rezgető bemenő kapcsain, hogy a tárgy az előírt rezgésterhelést kapja. A dinamikus rezgésvizsgáló mindenekelőtt megállapítja a momentán átviteli függvényt, majd ennek birtokában az előírt jelformát korrigálva a rezgető bemenetén olyan jelet generál, amely az aktuális átviteli függvénnyel módosítva a tárgyon éppen az előírt rezgés képet hozza létre. A mérőrendszer a következő lépéseket hajtja végre



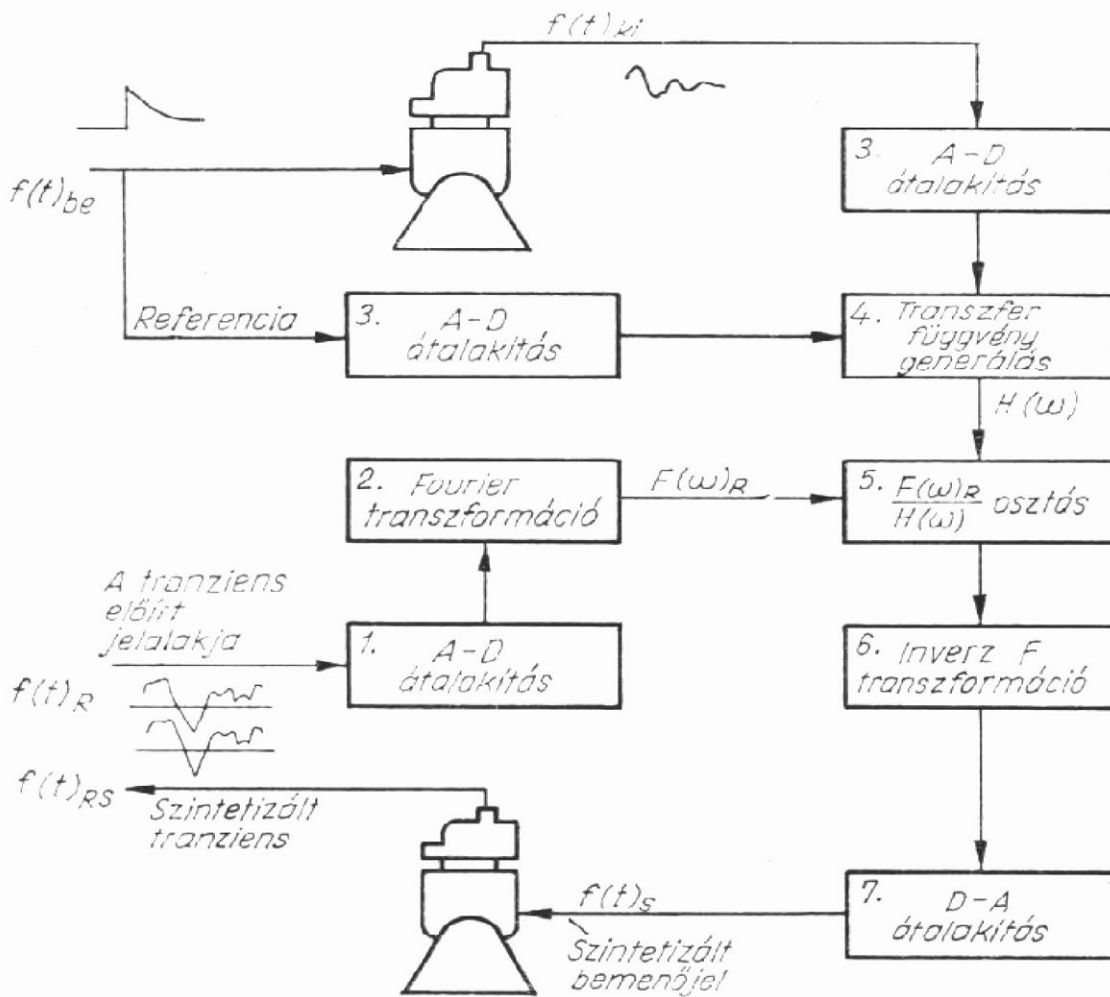
1.27. ábra.
Analog szabályzóval kombinált aktív számítógépes rezgésvizsgáló

- 1—2. Mintavételezi az *előírt* tranzienszt és Fourier-transzformáltját képezi.
- 3—4. Valamely referenciatranziens, valamint e tranziens által a rezgető kimenetén létrehozott jel Fourier-transzformáltjait képezi.
4. A 3—4. során nyert Fourier-komponensek alapján számítja a $H(\omega)$ aktuális átviteli függvényt.
5. Az átviteli függvénnyel korrigálja az 2. lépésben nyert Fourier-transzformáltat.
- 6—7. Inverz Fourier-transzformációval és D—A átalakító útján generálja a korrigált bemenő tranzienszt, mely a rezgető kapcsaira jutva a tárgyon az előírt rezgés képet állítja elő.

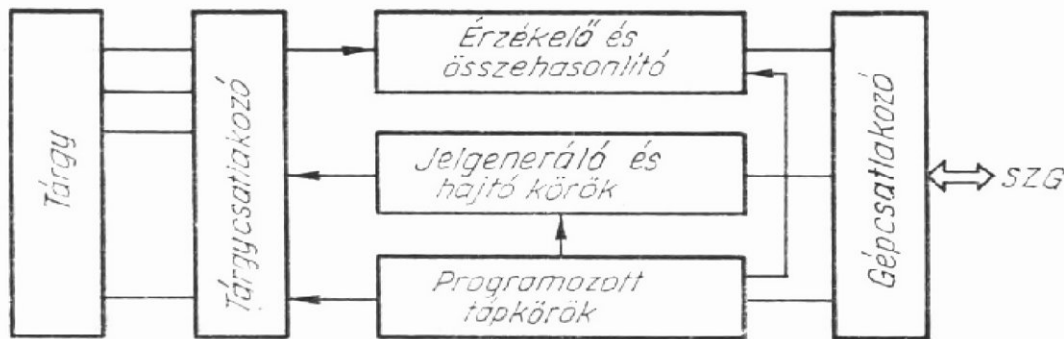
Az eljárás a rezgőrendszer időinvarianciájának és linearitásának feltételezésén alapszik. Az eljárás vázlatát (nem magát a mérőrendszert) az 1.28. ábra szemlélteti; a kockákba írt számok az egyes műveleti lépések sorrendjét jelzik. A tárgy tulajdonképpeni rezgésvizsgálata természetesen az előírt tranziens előállítás után kezdődik.

Digitális rendszerek

A digitális mennyiségeket vizsgáló rendszerek kivétel nélkül aktív mérőrendszerek, amelyeket a tárggyal inger- és érzékelőcsatorna köt össze. Alkalmazási spektrumuk lényegesen szűkebb, mint az analóg rendszereké: a digitális technikára (integrált áramkörök, digitális áramköri kártyák és egységek vizsgálatára) korlátozódik. A digitális mérés: funkcionális ellenőrzés. A jelgeneráló és hajtó (driver) elemek (1. 29. ábra) a tárgy bemenetére bináris — kérdő — vektorok sorozatát adják, az érzékelő és összehasonlító körök pedig ellenőrzik, hogy a tárgy által az egyes kérdővektorokra adott válaszvektorok előírás szerinti-e? Az előírt válaszvektoroktól való eltérés hibát jelez. Elemi alakjában tehát a digitális mérőrendszer jó/nem jó típusú ellenőrzést végez; elkülöníti a hibás egyedeket a hibátlanoktól.



1.28. ábra.
Dinamikus rezgésvizsgáló működési sémája



1.29. ábra.
Digitális ellenőrző berendezés mérőhálózatának vázlata

Integrált áramkörök esetében a hiba okának felderítése legfeljebb a gyártó szempontjából lényeges, hiszen a hiba a tokon belül nem javítható. Az áramköri kártyák, a nagyobb szerkezeti egységek ezzel szemben javíthatók: a hibás egységeket diagnosztizálni kell. A magasabb színvonalú ellenőrző rendszerek diagnózist is nyújtanak; a hibát nemcsak észlelik, hanem lokalizálják is.

Az analóg mérőrendszerek — mint láthattuk — esetenként kibővülnek bizonyos digitális mennyiségek vizsgálatára szolgáló elemekkel. Ugyanígy — legtöbb esetben — a digitális rendszerek is tartalmaznak analóg elemeket. Mindenekelőtt a tárgyra belépő logikai vektor jelkomponenseinek szintje program szerint állítható, az érzékelő összehasonlító áramkörök pedig a tárgyról érkező válaszjelek digitális tartalmán

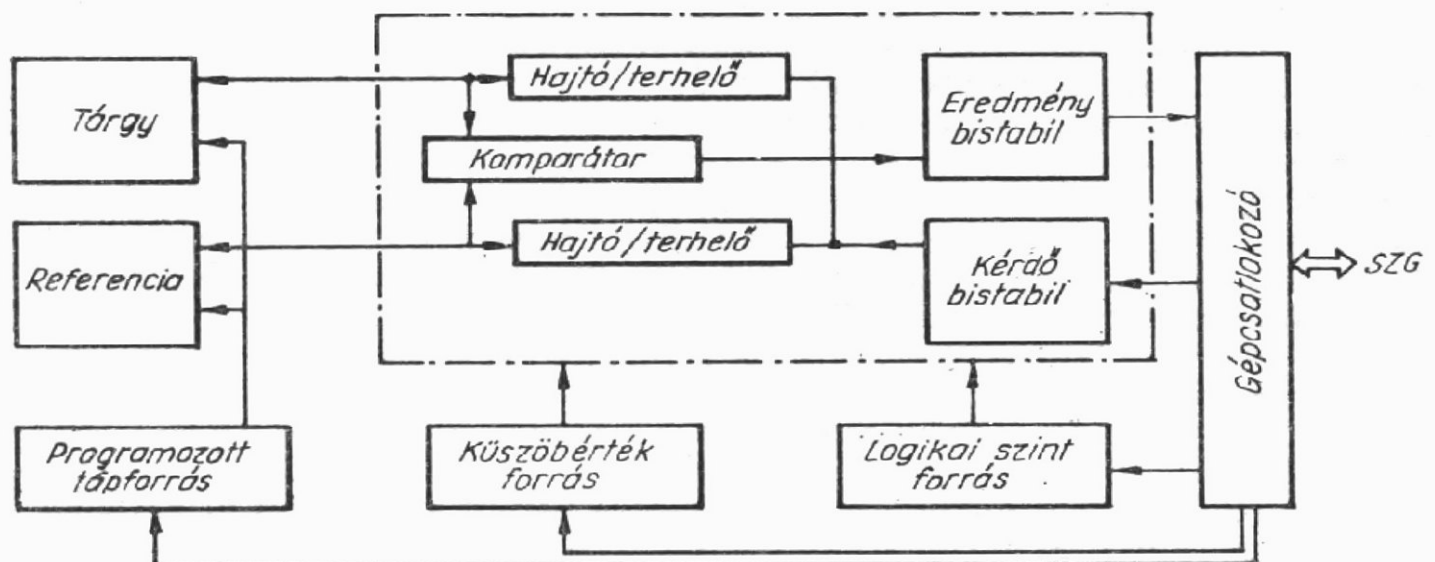
túl, azok analóg szintjét is ellenőrzik. A szintek beállítását a számítógép a programozott tápegységblokk útján végzi. A számítógépes digitális ellenőrző rendszerek többnyire kiegészülnek egyenáramú, valamint dinamikus (impulzus) jellemzőket mérő eszközökkel is.

A különböző digitális mérőhálózatok elsősorban az alkalmazott ellenőrzési stratégia, másodsorban parametrikus mérőhálózatuk kiépítettsége tekintetében térhetnek el egymástól.

A digitális vizsgálat alapjául, általában, a vizsgált áramköri egység logikai leíró egyenlete, ill. egyenletei szolgálnak. Az egyszerűbb kombinációs hálózatnál a kérdő- és válaszvektorok között egyértelmű kapcsolat áll fenn. Belső tárolókat is tartalmazó sorrendi hálózat esetében, valamely kérdővektorhoz tartozó válaszvektor az előéletet rögzítő belső állapotnak is függvénye, más szóval attól is függ, milyen áramköri kezdőállapotból indul a vizsgálat, s milyen kérdőértékeken, milyen sorrendben halad át.

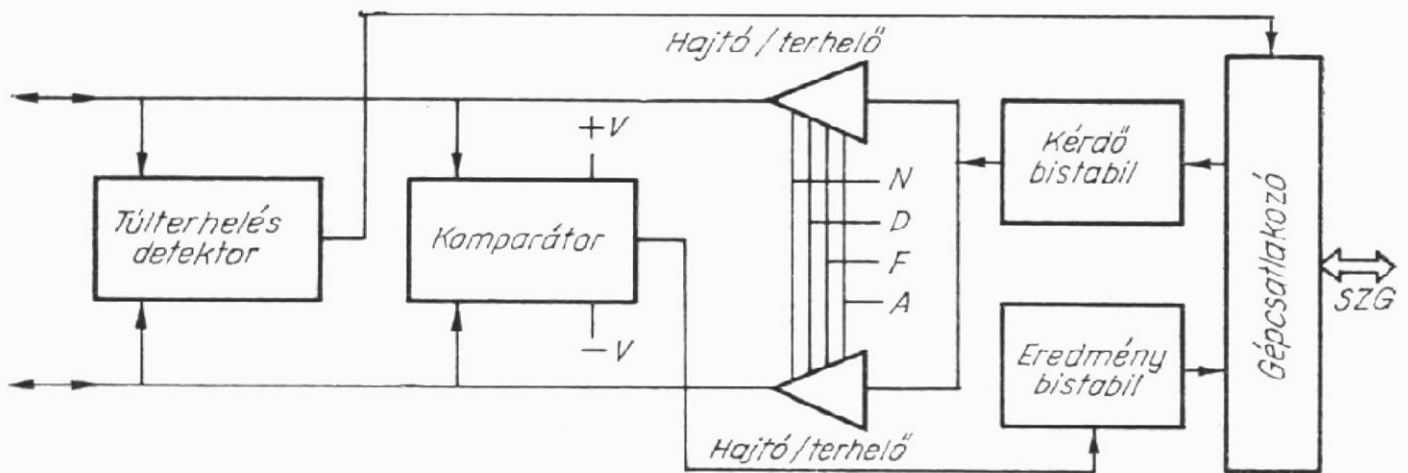
Legkézenfekvőbb vizsgálati módszer a teljes vizsgálat, mely a bemenetre az összes lehetséges bináris kombinációt (vektort) reáviszi, s a számítógép tárolójában elhelyezett igazságtáblázat alapján ellenőrzi az egyes bemenetekhez tartozó kimenetek helyességét. A teljes vizsgálat módszere számos szempontból problematikus. Kombinációs hálózatnál, k -számú bemenő pont esetében a teljes vizsgálat 2^k kérdővektor alkalmazását teszi szükségessé. Példaképpen 60 bemenő pontot és $t=0,4 \mu s$ vizsgálati időt tételezve fel, a teljes vizsgálat ideje kb. 14 000 esztendő, de még ha $k=30$, akkor is 7 percet tesz ki. Elsőként azonban nem is a vizsgálati idő, hanem az igazságtáblázat méretének problémája jelentkezik. Már $k=16$ esetén is az igazságtáblázat sorainak száma 65 536. Olyan környezetben, amely nélkülözi az áramköri szimuláció lehetőségét, ilyen méretű táblázat manuális eszközökkel való kiszámítása elfogadhatatlan terhet jelent (a táblázat óriási memóriaterület-igénye háttértárolóval még csak megoldható). Szekvenciális köröknél a vizsgálati kombinációk száma, az ellenőrzéshez szükséges táblázatok területe, a belső állapotok számától függően a kombinációs körökhöz viszonyítva még megsokszorozódik.

A felvázolt problémák a kérdővektoroknak szerkezeti, ill. program útján történő algoritmikus generálása, a „helyes válaszok” szerkezeti generálása útján részben kiküszöbölhető. Ezen az alapon működik az 1.30. ábrán bemutatott digitális mérőhálózat, amelynek jellegzetes vonása a tárgyjal teljesen azonos felépítésű (ellenőrzötten kifogástalan működésű) referenciaegység jelenléte. Mind a tárgy, mind a referencia-



1.30. ábra.
Digitális mérőhálózat szerkezeti referenciával

egység összes csatlakozópontjára azonos felépítésű hajtó/terhelő áramkörök kapcsolódnak, a tárgy, ill. a referenciaegység azonos nevű csatlakozópontjai között pedig egy-egy komparátor helyezkedik el. A program, annak ismeretében, hogy a tárgy (referencia) mely csatlakozópontja bemenet és mely csatlakozópontja kimenet, a hajtó/terhelő egységekre vagy hajtó, vagy terhelő szerepkörét ruházza rá. A komparátor a hajtók mellett monitorként, a terhelők mellett ellenőrzőként működik. Az 1.30. ábra egyetlen csatlakozóvonalat tüntet fel, mely tehát a program választásától (a vizsgált egység adottságaitól) függően, a tárgyra (referenciára) irányuló ingervonal, vagy a tárgytól a hálózat felé irányuló érzékelővonal szerepkörét egyaránt betöltheti. A mérőcsatorna részletesebb vázlata az 1.31. ábrán látható. A hajtó/terhelő áramkörök vezérlőpontjai a felső (F), azaz pozitívabb, ill. az alsó (A), azaz negatívabb logikai szint értékének, valamint a pozitív (P) és negatív (N) áramkorlát értékének beállítására szolgálnak. A komparátor $+V$ és $-V$ kapcsain megadhatók a válaszjel elfogadható tartományát a nem elfogadhatótól elválasztó feszültség szintek. A logikai és komparációs szintek beállítására a tárgy és a referencia áramellátására programozott tápforrások állnak rendelkezésre. (1.30. ábra.) A kérdővektort, ill. a tárgy és a referencia válaszvektorának komparációs eredményét egy-egy regiszter tárolja; ezek egy mérőcsatornához tartozó sejt-eit a csatornavédő túlterhelésetektorokkal együtt az 1.31. ábra mutatja.



1.31. ábra.

A szerkezeti referenciát alkalmazó mérőhálózat egy érintkezőhöz tartozó áramköre (pin electronics)

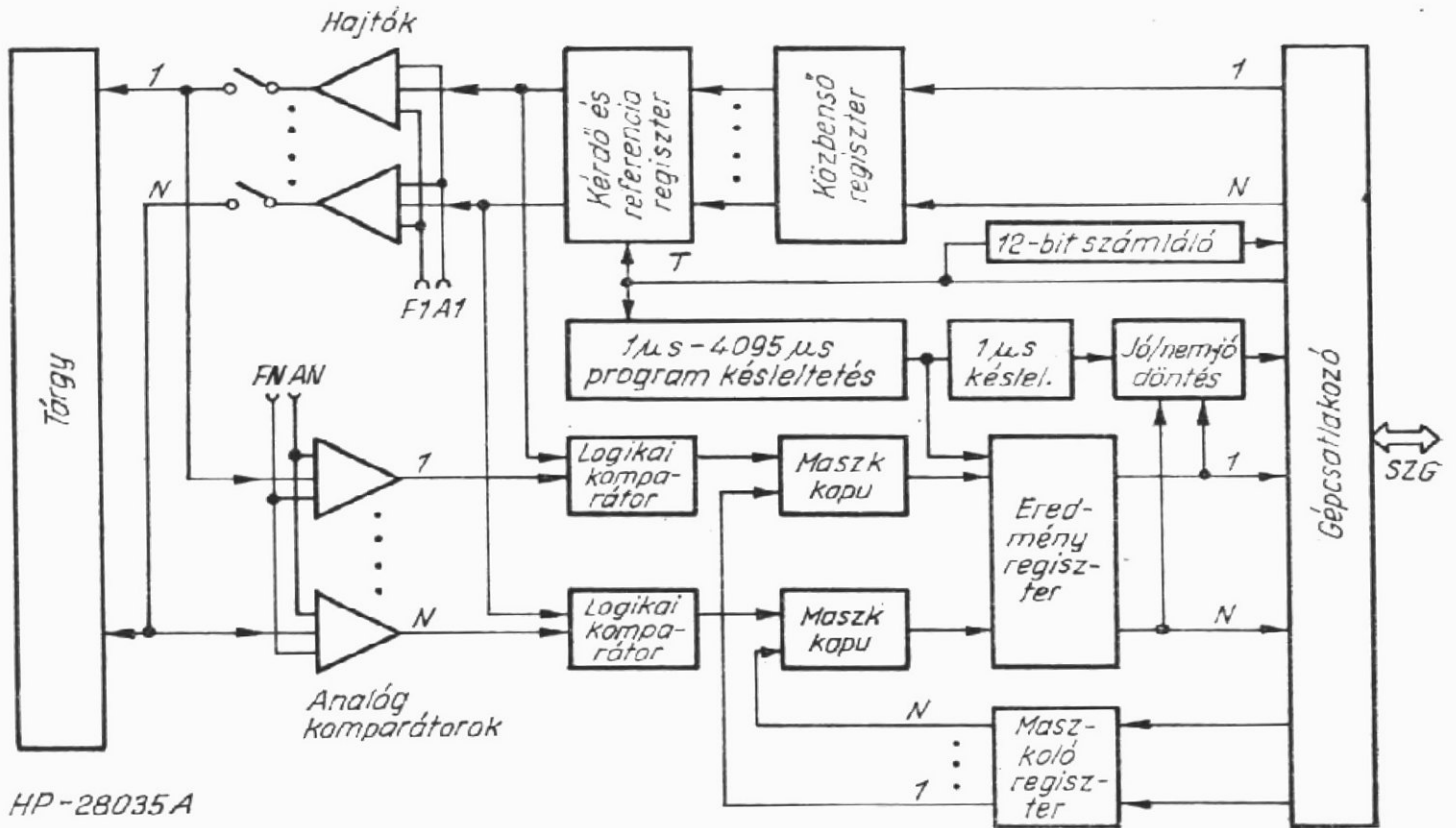
A szerkezeti referenciával működő, teljes vizsgálatot végző ellenőrző rendszer hátrányos vonásai között továbbra is szerepel a nagyobb bemenő kapocsszám esetén szükséges hosszú mérésidő (más szóval: az időkorláthoz kötött kapocsszám korlát). További negatívum, hogy a teljes ellenőrzés nem ölel fel diagnosztikai lehetőségeket. Sorrendi hálózatok vizsgálata során általában arra sincs biztosíték, hogy a mérőhálózat üzembe helyezése után a referenciaegység a tárgyéval azonos belső állapotot foglaljon el. Ez azzal járhat, hogy hibátlan tárgy esetén is, a két áramköri egységtől eltérő válasz érkezik. A sorrendi hálózatokat, célszerűen, valamely visszahelyező vektorsorozat segítségével viszik kiindulási helyzetbe, de a különböző állapotból indított egységek más-más visszahelyező sorozatot igényelnek. Szerkezeti összehasonlítás esetében végül, a referenciaegység üzemképes állapotának ellenőrzésére nincs mód. A felsorolt hátrányokat bizonyos mértékben kiküszöbölik a tárolt referenciájú rendszerek.

A tárolt referenciával működő áramkör-ellenőrzőknek két változatát különböztetjük meg, a tanuló- és a szimulációs rendszert, amelyek nem annyira mérőhálózatuk felépítésében, mint inkább programrendszerük szervezésében térnek el egymástól.

A tanuló-rendszer a kérdővektorok referencia-válaszpárjait a tárgy helyére állított referenciaegységgel generálja. A garantáltan jó referenciaegység válaszait tárolva a megfelelő kérdő- és válaszvektorokat táblázatba foglalja. Ellenőrzés során a rendszer a tárgy válaszait a memóriában őrzött referenciaválaszokkal hasonlítja össze.

A szimulációs rendszer (on-line pattern generátor) a tárgy logikai működését szimuláló programon alapul. A szimulációs program a bemenetére adott kérdővektorra éppúgy a helyes válaszvektort adja, mint a szerkezeti referenciaegység. A tárgy szimuláltját szimulált vagy program-referenciaegységnek is nevezhetjük. Mind a tanuló-, mind a szimulációs rendszer mentes az eltérő kezdeti állapot problémájától, s a szerkezeti referenciaegység meghibásodási lehetőségének problematikája is elesik.

Tárolt referenciájú mérőhálózatot mutat be az 1.32. ábra. Az N -számú vizsgálati



HP-28035A

1.32. ábra.
Tárolt referenciájú digitális ellenőrző mérőhálózata
A — alsó logikai szint; F — felső logikai szint

ponthoz ugyanannyi analóg komparátor, ill. hajtó tartozik. A hajtók csak a tárgy kérdőpontjaira kapcsolódnak rá, a válaszpontokat a hajtóktól kapcsolók választják le. Az összevont kérdő- és referenciavektor a gépcsatlakozóból először közbenő regiszterre lép. A mérés a T indítójel hatására kezdődik, mely az összevont vektort a kérdő- és referenciaregiszterre viszi át. A T jel továbbhalad a program által beállított késleltető szakaszon, és ΔT múlva kapuzza az eredményregisztert. Időközben a válaszvektor a kérdővektorral együtt az analóg komparátorokra, majd a logikai komparátorra kerül, amely összehasonlítja az eredeti öszetett vektorral. Az összehasonlítás eredményéből a maszkolóregiszter elkülöníti a kérdővektor komponenseit, úgy, hogy az eredményregiszterre már csak a válasznak a referenciával történő összevetése kerül. A jó/nem jó döntés az összes válaszcatorna eredményei alapján születik meg. A vizsgálati lépések sorszámát (12-bites) számláló jegyzi fel.

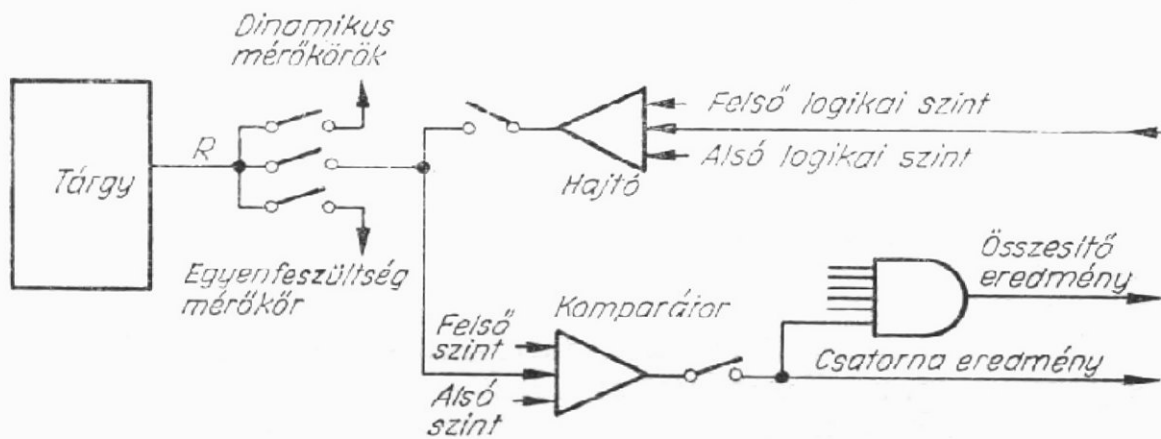
A tanuló-rendszer vagy a szimulált referenciaegység módszerének alkalmazása nem érinti a teljes logikai vizsgálatához fűződő problémákat. A teljes logikai vizsgá-

lattal kapcsolatos nehézségeket különböző hibaelemzési és felderítési stratégiák alkalmazásával igyekeznek megkerülni. E stratégiák abból a megfontolásból indulnak ki, hogy a teljes vizsgálat redundáns, a hozzá tartozó összes kérdővektorok egy részhalmaza is elegendő a jó/nem jó döntés megalkotásához. A részhalmaz egyedeinek az egészből való kiválasztására különböző (hibamátrix, D-algoritmus, Boole-különbség, állapottáblázatok, search) módszereket alkalmaznak. Minthogy a részhalmaz egyedeinek kiválogatása a konkrét tárgy bizonyos feltételezett hibáinak megfelelően történik, e stratégiák nem csak a vizsgálati lépések számát redukálják, hanem a diagnosztizálás lehetőségét is magukba foglalják.

Az egyenáramú paraméterek mérése a digitális technikában elsősorban az integrált, de természetesen más áramkörök vizsgálatával kapcsolatban is szükséges lehet. A logikai (funkcionális) vizsgálatok önmagukban nem mindig szűrik ki a hibás példányokat. A funkcionális vizsgálat pozitív eredményt hozhat, ugyanakkor az integrált áramkör mégsem mindenben felel meg üzemi előírásainak. A bemenő kapcsokon jelentkező szivárgási áramtöbblet pl. a tényleges áramköri környezetben zavart okozhat, bár az áramkör logikai vizsgálatának körülményei között minden rendbenlévőnek látszott. Az egyenáramú parametrikus mérés a szivárgási áram, maradék feszültség, kapcsolási küszöbérték, bemenő áram, kimenő áram, a bemenet és kimenet terhelési adatai, statikus zajtűrés, teljesítményfelvétel, letörési feszültség, statikus logikai szintek megállapítására irányulhat.

A dinamikus mérőáramkörök (impulzusparaméter-vizsgálók) fel- és lefutási idő, átfutási idő és átfutási idők különbsége, túllövés, alálövés, dinamikus zajtűrés, zajáthallás, impulzusszélesség, lépcsőfüggvényre adott válasz, max. ismétlődési frekvencia stb. mérésére térhetnek ki.

Bár a logikai vizsgálat a statikus és dinamikus paramétermérésekkel csak jelentékeny nehézségek árán hozható össze, ismereteseek mindhárom fajta mérést végző mérőhálózatok is. A különböző fajta mérések általában nem egyidejűleg, hanem egymást kizáró módon végezhetők; az aktuális mérőeszközöket a kiválasztott mérőponttal, erre szolgáló reléegység (1.33. ábrán R) köti össze.



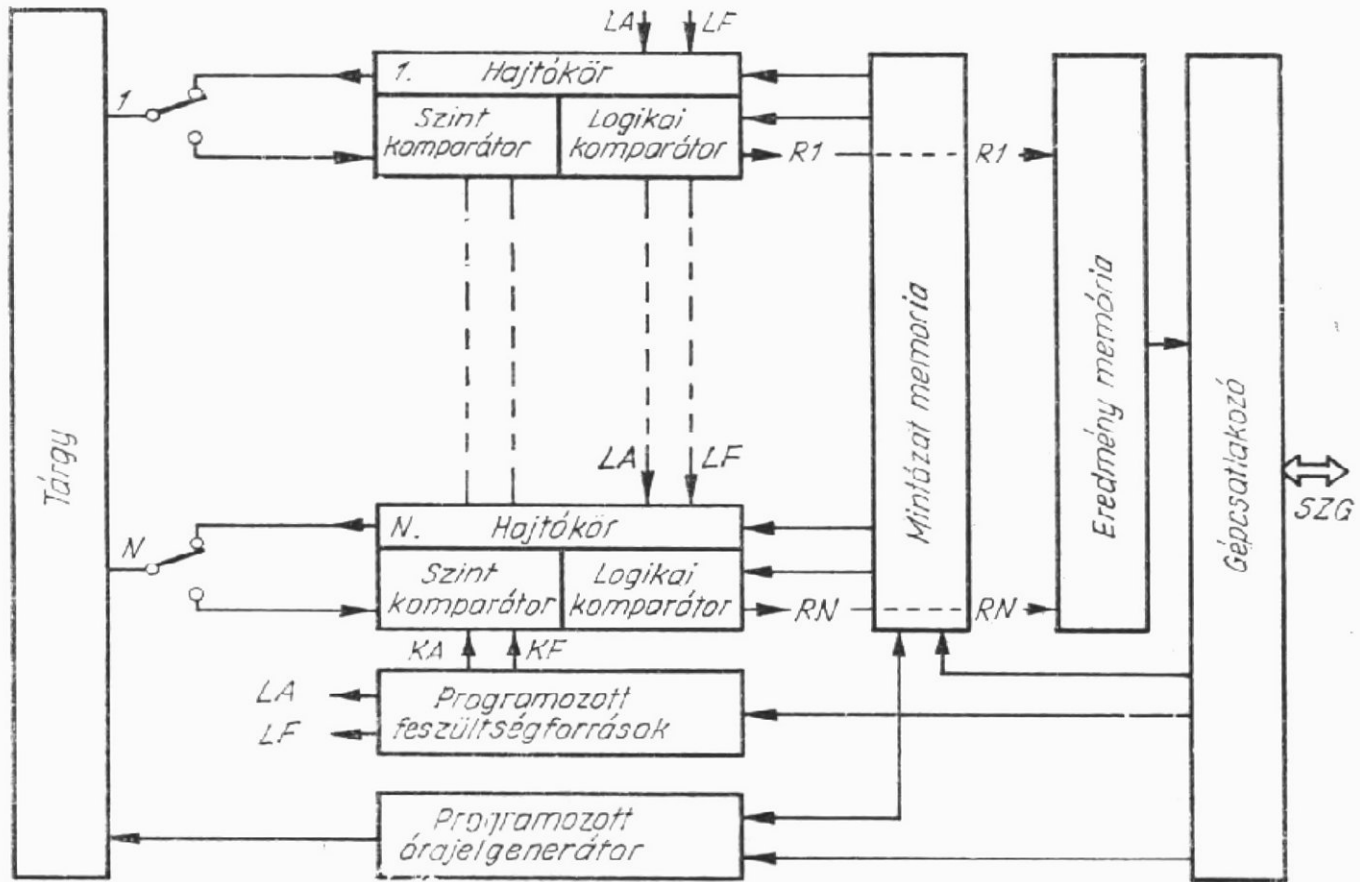
1.33. ábra.

Logikai ellenőrzést, statikus és dinamikus paramétervizsgálatot egyaránt végző mérőrendszer egy érintkezőhöz tartozó áramköre

A paramétermérők ellenőrző vagy adatközlő üzemmódban működhetnek; jó/nem jó döntést vagy pontos mérési adatot jegyzőkönyvi alakban közölnek.

A digitális áramköri egységek üzemi körülmények között végzett funkcionális ellenőrzését gyakran a számítógépcsatorna átviteli lehetőségeit meghaladó ütemben kell végezni. Ebben az esetben a mérőblokkot — a feladattól függően — közbenső tárolóval vagy szógenerátorral (vektorgenerátor) kell kiegészíteni. A közbenső tároló a gépcsatlakozó irányából a számítógép átviteli lehetőségeinek megfelelő

ütemben töltődik fel a vizsgálat bitmintázatával (kérdő- és válaszreferencia-vektorok), a vizsgálat lebonyolítása idején pedig lényegesen nagyobb sebességgel ürítődik a tárgy irányában. Az eredmények közbenső tárolására, az ellenkező irányú sebességkonverzióra, egy második közbenső memória szolgálhat (1.34. ábra). A mintázat-

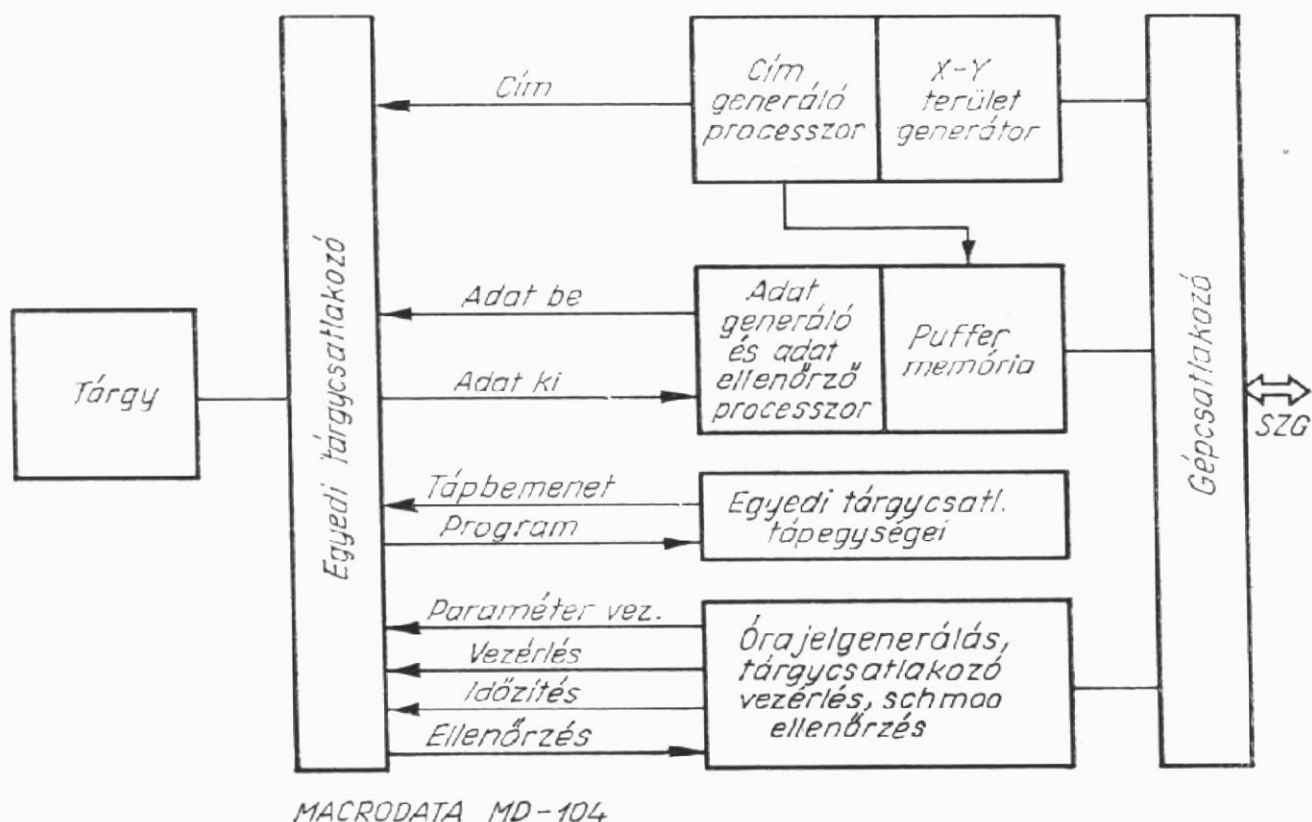


MINATO LSI - 1002

1.34. ábra.
Digitális mérőhálózat ki- és bemeneti közbenső memóriával

memóriából való olvasás, az eredménymemóriába történő írás a közbenső memóriarendszer saját vezérlési ütemében történik (programozott órajel-generátor). Közbenső memóriaként kötött elérési sorrendű regisztermemória is alkalmazható, azonban kötetlen elérésű olvas/ír memória előnyösebbnek bizonyulhat. Egyes digitális elemek (pl. léptetőregiszter) vizsgálata rövid, kevés szóból álló mintázatokat igényel, melyeket azonban számtalanszor meg kell ismételni. Célszerű, ha az egyes mintázatszakaszok kezdő- és végpontja, a mintázatok hossza, a belső ciklicitás, mindez program által irányítható, ami azonban csak kötetlen elérési sorrendű közbenső memóriával valósítható meg; a regisztermemóriák egyedül belépő és kilépő peremükön közelíthetők meg.

Az integrált áramkörök egy részének áramköri és logikai szerkezete bizonyos szabályosságot mutat. Ide tartoznak a különböző fixmemóriák, olvas/ír memóriák, léptetőregiszterek stb. Ellenőrzés szempontjából ezeket a struktúrákat a vizsgálati mintázatok sorozatának generálhatósága, algoritmizálhatósága jellemzi. Ahelyett, hogy az ellenőrző mintázat generálását a számítógép végezné, hogy aztán a generált mintázatot közbenső tárolóba tegye át, vagy hogy a mérőhálózatban rögzített szerkezeti vizsgálati rutinokat alkalmaznának, egyes digitális mérőhálózatok a mintagenerálást a részüket képező paralel processzorok (kontrollerek) segítségével végzik (1.35. ábra). Az ilyen többprocesszoros ellenőrző rendszerek igen nagy követési frekvenciával (jellemzően 10...20 MHz) állítják elő a címek, adatok, referen-



MACRODATA MD-104

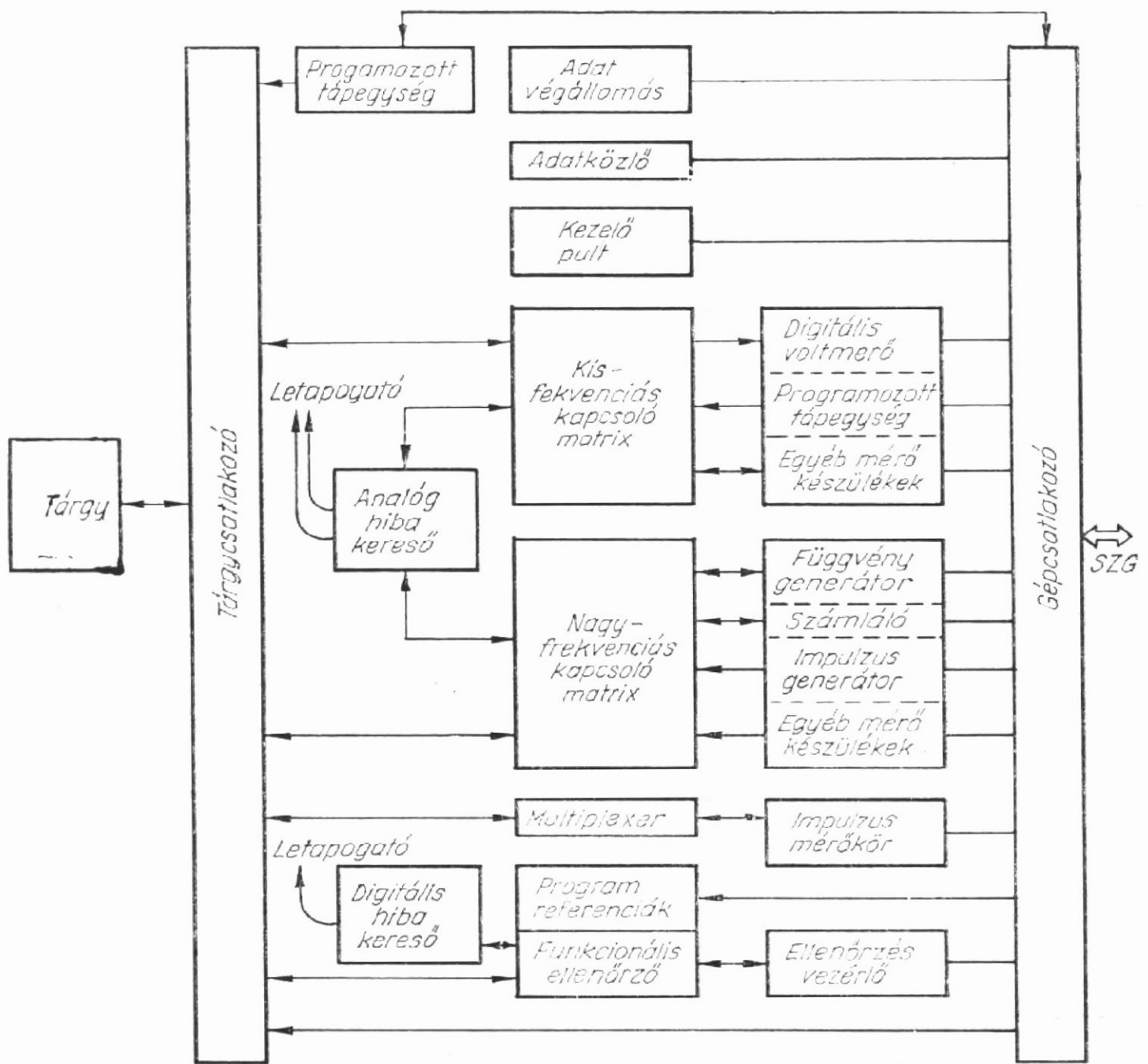
1.35. ábra.
Digitális mérőhálózat periferikus processzorokkal

ciaválaszok, valamint vezérlővektorok (különböző generálási algoritmusok által definiált) szekvenciáit. Az 1.35. ábra mérőhálózatában kiemelt szerepe van az egyedi tárgycsatlakozónak. Mindazokat a feladatokat, amelyek speciálisan az integrált áramkör egy adott típusával kapcsolatosak, az egyedi tárgycsatlakozó látja el. Így pl. a belső időzítést, a jelszintek beállítását, a tápfeszültség-ellátást, az elektromos és mechanikai csatlakoztatást.

Általános jellegű hálózatok

A mérőhálózatot általános jellegűnek akkor nevezzük, ha kiterjedt analóg és digitális mérési lehetőségekkel egyaránt rendelkezik, s az analóg és a digitális mérőhálózati elemek nagyjából hasonló súlyt képviselnek. A moduláris rendszer kiépíthetőségének gyakran nem a készülékeknek vagy építőmoduloknak a számítógéphez illeszthetősége, hanem a tárgycsatlakozót alkotó elemek tulajdonságai szabnak határt. Az általános jellegű hálózatok eleve úgy épülnek fel, hogy egyenáramtól a nanoszekundumos impulzustartományig elláthassák a tárgynak a választás szerint adódó mérőkészülékekhez való csatlakoztatását. Ideálisan általánosnak azt a mérőrendszert nevezhetnénk, amelynek strukturális modellje rendelkezik a műszaki mérés-technika bármely igényéhez való adaptáció lehetőségével. Ilyen rendszerstruktúra azonban ez időben még gondolatban sem alakult ki. Jelenleg tapasztalhatók arra irányuló törekvések, hogy egy-egy mérőrendszer lehetőleg legalább egyetlen alkalmazási terület szükségleteit elégítse ki általános érvényűen.

Az 1.36. ábra elektronikus alkatrészeket, modulokat és szerelvényeket vizsgáló általános jellegű mérőrendszert mutat be (hasonló még a HP-9510D és a Systron-Donner 3600 CATS), melyben egyaránt megtalálhatók, ill. beépíthetők a korábban ismertetett analóg és digitális áramkör-ellenőrző hálózatok elemei.



TEXAS INSTRUM. ATS-960

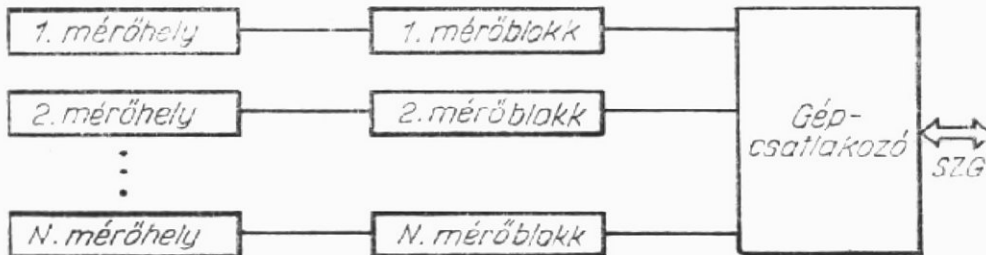
1.36. ábra.

Analog és digitális vizsgálatokra egyaránt alkalmas általános mérőrendszer

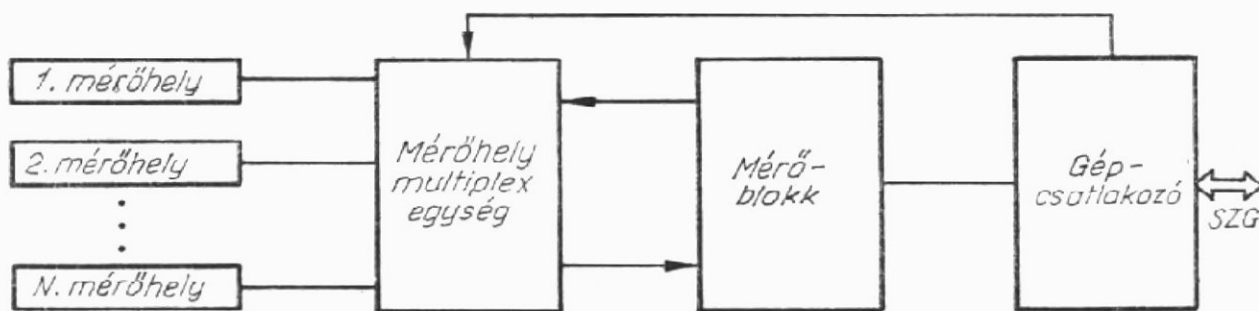
Többszörös helyes hálózatok

Többszörös helyes hálózatokat elsősorban gazdasági megfontolások alapján alkalmazzák, a mérőrendszerben felhasznált szerkezeti elemek kedvezőbb kihasználása, a rendszer központi szervei teljes feldolgozóképeségének hasznosítása céljából. Többszörös helyes rendszer bevezetése olyan környezetben kerülhet szóba, amelyben az egyes mérőhelyek a rendszer teljes idejének viszonylag csekély hányadát foglalják le mérés-technikai, adatátviteli, adatfeldolgozási stb. feladataikkal. A rendszer szerkezeti egységeinek több mérőhely között való megosztásának különböző szintjei lehetségesek. Bizonyos esetekben (1.37. ábra) minden mérőhelyhez külön mérőblokk tartozik, az egyes mérőhelyek egyedül a számítógépen osztoznak. Más esetben a mérőhelyek nem csupán a számítógépen, hanem a mérőblokkon is osztoznak

(1.38. ábra). E két megoldási lehetőség között különböző átmeneti variánsokat találunk: N számú mérőhely osztozhat pl. $M < N$ számú mérőblokkon, vagy — bár a rendszer egyetlen mérőblokkal rendelkezik — egyes mérőhelyekhez különböző kiegészítő elemek, készülékek tartoznak. Az egy mérőblokkal, több mérőhellyel rendelkező rendszer tipikus példája az 1.39. ábra laboratóriumi anyagvizsgáló mérőrendszere. A lassú működésű, kis adathozamú és relatíve csekély számítástechnikai igényű anyagvizsgáló blokkok számítógépes mérésautomatizálása úgy oldható meg gazdaságosan, ha több anyagvizsgáló blokkot egy mérőrendszer, egy számítógép irányítása alá vonnak. (Az ábrán látható *KS*-csöves adatközlő egységek a rendszerrel való konverzációra szolgálnak.)



1.37. ábra.
Időosztásos mérőrendszer mérőhelyenként önálló mérőblokkal

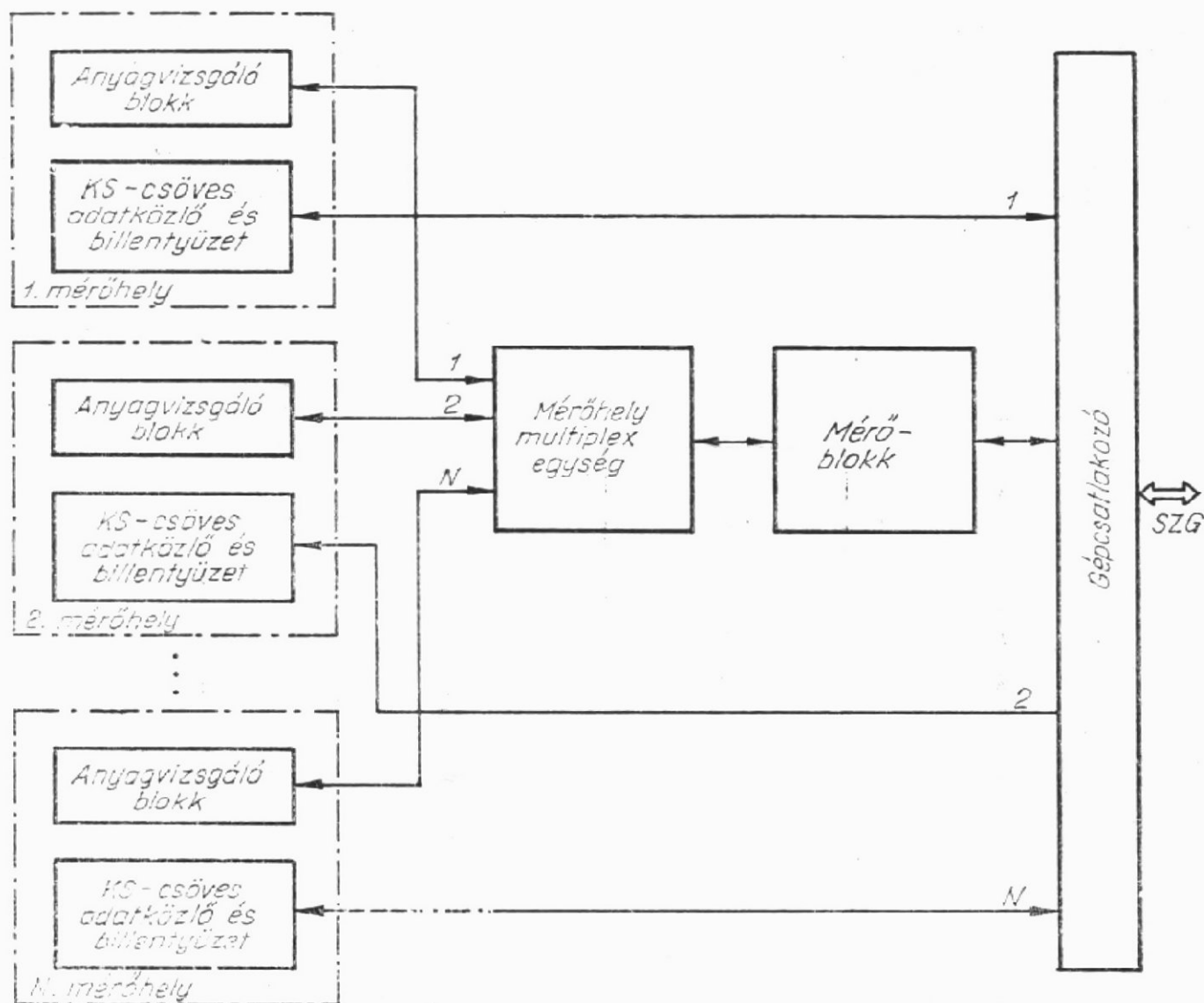


1.38. ábra.
Időosztásos mérőrendszer központi mérőblokkal

A központi szervek időmegosztása különböző mérőhelyek között — az adott konkrét alkalmazási környezettől függően — más és más módon történhet. A több-mérőhelyes rendszer legegyszerűbb változata mérőpaklit alkalmaz. A mérőpakliban bizonyos számú azonos tulajdonságú mérési objektumot helyeznek el annak érdekében, hogy a tárgy kézi cserélgetésével a mérés folyamán ne kelljen a mérőrendszer idejét lefoglalni. A mérőpakliban levő egyedeket (pl. tranzistorok) multiplexer kapcsolja rá sorban a mérőrendszer bemenetére. Valamennyi egyed vizsgálati ideje azonos lévén, a rendszer minden egyed vizsgálatára bizonyos időt oszt ki, reális idejű óra segítségével.

Mechanikai adagolók időkiosztását, mechanikai reciklálását a programrendszer végezheti programbázisú számláló, vagy reális idejű óra útján. Ez azonban csak egyszerűbb működési struktúrák esetében ad kielégítő eredményt; célszerűbb, ha a számítógép megszakító hálózatát működtetik az adagolók mechanikája által.

Ez utóbbinál az adott mérőhely munkaciklusának indítása, a számítógéppel a megszakító hálózaton át közölt „mérésre kész” jel segítségével történik. Miután a számítógép a jelet küldő mérőhelyet azonosítja, a mérőblokkot multiplex egység útján kapcsolja a mérőhelyre. A vizsgálat ezt követően úgy zajlik le, mint az egyetlen mérőhelyes rendszernél.

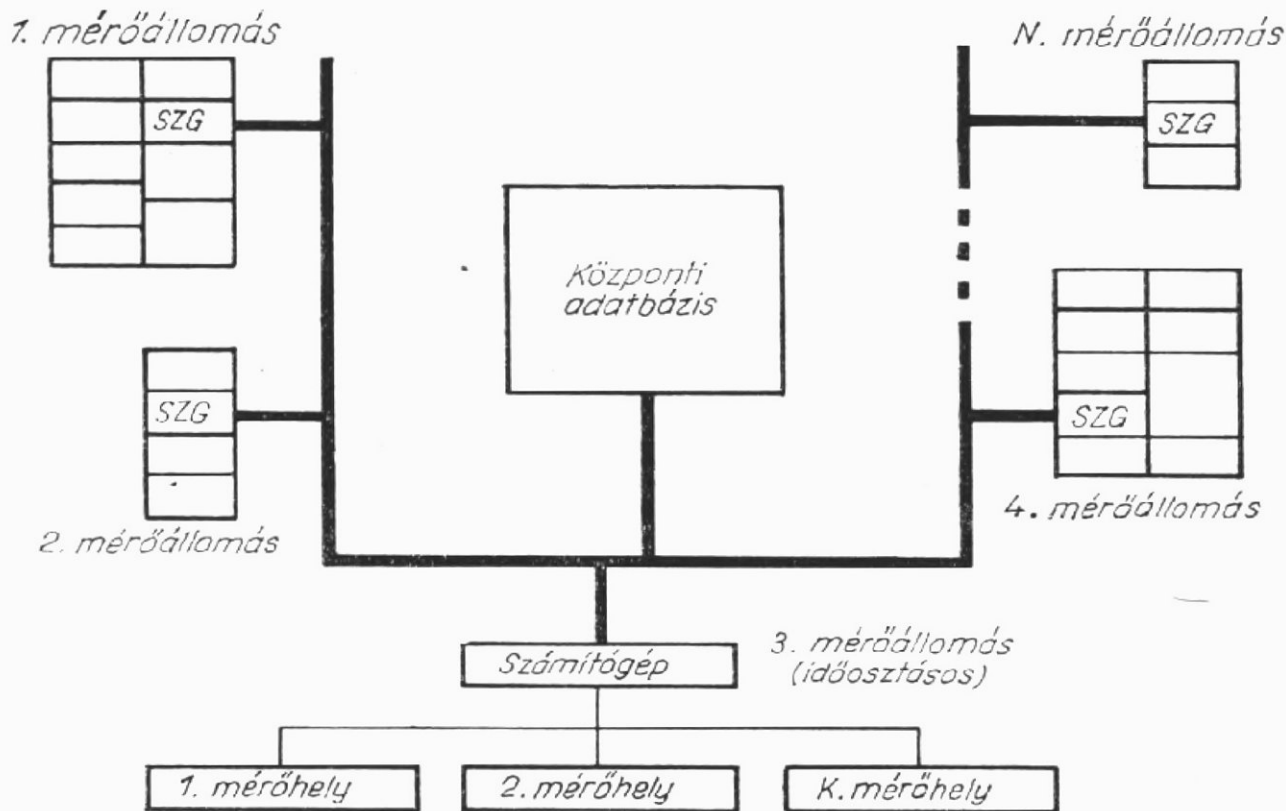


1.39. ábra.
Anyagvizsgáló laboratórium időosztásos mérőrendszere

A kiszámítógépek egyes típusainál a megszakításkérés eleve definiálja a megszakítást kérő forrást (ez esetben mérőhelyet), más gépek megszakító rendszere a forrást azonosítási eljárással identifikálja. Az egyszerűbb struktúrájú megszakító rendszer kiegészülhet szerkezeti letapogatóval is, amely meghatározott sorrendben ciklikusan végigkérdezi a mérőhelyeket. Amennyiben a letapogató valamelyik mérőhelyet szolgálatkérés állapotában találja, megáll, és a mérőhely azonosítási számának közlésével együtt megszakításkérést küld a számítógéphez. A letapogató program útján is végezhető, de ez az eljárás nehézkes és időtrabló.

A mérőrendszer legfejlettebb változata a többmérőállomású központi adatbázisú integrált ellenőrző rendszer (1.40. ábra). Az egyes önálló számítógéppel irányított mérőállomásokat (ezek sorában általában egymérőhelyes és időosztásos többmérőhelyes változat egyaránt előfordul) a központi adatbázissal nagysebességű (pl. 250 KHz) soros átvivő vonal köti össze. A központi adattároló és az egyes mérőállomások közötti távolság viszonylag nagy (pl. 500 m) lehet. Az egyes mérőállomások autonóm működésűek. A központi adatbázis a programkönyvtár, valamint a jegyzőkönyvezett adatok tömegtárolását végzi, osztott diszk-file alakjában. A mérőállomások az aktuális ellenőrző programokat, szubrutinokat és végrehajtó programokat a központi adatbázisból hívják le. A programok univerzális jellegűek, minden mérőállomáshoz illeszkednek. A mérés befejezte után a mérőállomás a tárolójában összegyűjtött jegyzőkönyvi adatokat a központi adatbázishoz továbbítja. E rendszer

gazdasági előnyei nyilvánvalók (bizonyos központi feladatok ellátására egyetlen szerkezeti elemet alkalmaz, ezáltal az egyes mérőállomásokon jelentékeny megtakarítás érhető el), azonban munkaszervezési és információfeldolgozási szempontból is — a számos egyedi ellenőrző egységgel szemben — magasabb színvonalat képvisel. Egységes programformátum, központi adminisztráció, egységes adatformátum tünteti ki. Közvetlenül bekapcsolható a vállalati termelésirányítás információs rendszerének egészébe.



1.40. ábra.
Többszámítógépes integrált üzemi ellenőrző rendszer

Irodalom az 1. fejezethez

- [1] Новицкий, П. В.: Основы информационной теории измерительных устройств, Энергия, Ленинград, 1968.
- [2] Долинский Е. Ф.: Классификация измерений, Измерительная техника, 2 (1972) 8... 11. old.
- [3] Сотсков Б. С.: Измерения и информационно-измерительные системы. Problems of Control and Information Theory, Vol. 1(2) (1972) 103...115. old.
- [4] Finkelstein, L.: Grundgedanken zum Messen, mrs. 16. H.6 (1973) 201...205. old.
- [5] Münch, G.: Allgemeine Betriebsmesstechnik, Industrie-Anzeiger 94.Jg. Nr.89v. 24.10 (1972) 2131...2135. old.
- [6] Peklenik, J.: Automatisierung der Messwertverarbeitung in der fertigungstechnischen Forschung und Entwicklug, wt-Z. ind. Fertigg. 61. (1971) 154...157. old.
- [7] Astrop, A. W.: Metrology and inspection — some present and future trends, Machinery and production engineering, 14 February (1973) 214...220. old.
- [8] Steinhorst, W.: Der Systembegriff und seine Bedeutung in der Technik, Feinwerktechnik, 75. H3 (1971) 97...144. old.
- [9] Test Equipment, Mintech/Marconi Company Production Testing Study Vol. 2 (1969).

- [10] *Sebestyén B.*: Számítógép-bázisú automatikus ellenőrző rendszerek, *Mérés és Automatika*, 18 (1970) 6. sz. 172...176; 8.sz. 270...274. old.
- [11] *McAleer, H. T.*: A look at automatic testing, *IEEE spectrum*, May (1971) 64...78. old.
- [12] *Sebestyén B.*: Számítógép a műszaki ellenőrzéstechnikában, *Mérés és Automatika*, 20. 12. sz. (1972) 447...451. old.
- [13] *Andreiev, N.*: Programmable logic controllers — An update, *Control Engineering*, Sept. (1972) 45...47. old.
- [14] *Franck, D. T.*: Programmable controllers — today and tomorrow, *IEEE Transaction on Industrial Electronics and Control Instrumentation*, Nov. (1973) 195...196. old.
- [15] *Hilton, R.*: Minicomputer controller is inexpensive, *Electronics*, Jan. 18 (1973) 186...187. old.
- [16] *Occhiogrosso, J. J.*: Interfacing to desk-top calculators, *Research/Development*, April (1971) 22...25. old.
- [17] *Grimes, J. D.—Hill, R. V.*: Data acquisition, processing and display — Calculator-based. *Wescon vol. 17. 27/3* (1973) 1...4. old.
- [18] *Stone, P. S.*: Systems type measurements in the laboratory using programmable bench instruments under calculator control, *Wescon vol. 17. 27/3* (1973) 1...7. old.
- [19] *Jennings, R.*: Calculator-based systems: plug-in data acquisition and process control, *Wescon vol. 17. 27/1* (1973) 1...5 old.
- [20] *Herzog, R. E.*: Calculator-based systems challenge minis, *Machine Design*, January 24 (1974) 96...100. old.
- [21] *Jackson, P. C.*: Integrating nonstandard peripheral devices into automated test systems, *Computer Design*, Sept. (1973) 83...88. old.
- [22] *Forbes, B.*: Computer architecture for instrumentation, 1971 *Computer Designer's Conf.* 55...60. old.
- [23] DATEL System 256, *Bulletin 256A410307*
- [24] *Ruggeri, L.*: Multiplex signals the analog way, *Electronic Design* 15, July 19 (1974) 74...77. old.
- [25] *Strassberg, D. D.*: Multiplex and grounding in analog-digital data acquisition system, *ISA Transaction*, vol. 11, No. 3. 259...273. old.
- [26] Computerized data acquisition aids final testing, *Hewlett-Packard Application Note 135-1* (1971).
- [27] *Gossel, D.*: Frequenzanalogue, *ETZ—A. Bd. 93. H.10.* (1972) 577...581. old.
- [28] *Richter, W.*: Frequenzanalogue Meßsysteme, *mrs. 15. H.8.* (1972) 280...282. old.
- [29] *Vorce, R. G.*: Frequency-division multiplexing, *The Electronic Engineering*, Jan. (1969) 82...88. old.
- [30] Closed-Loop Production Testing, *Hewlett-Packard Application Note 135-4.*
- [31] *Grimm, R. A.*: An automatic test system utilizing standard instruments and abbreviated English language, *Proc. of the Automatic Support Systems Symposium, 1D* (1968) 1...10. old.
- [32] *Herz, F.—Stawinski, H.—Elsasser, W.*: Acoustic data collection and evaluation with the aid of small computer, *Brüel & Kjaer Technical Review*, No. 1 (1970) 3...18. old.
- [33] *Kapuskar, W. T.—Balmforth, C. J.*: Real-time measurement and on-line processing of acoustical and other audio-frequency spectra, *Hewlett-Packard Journal*, July (1969) 2...10. old.
- [34] *Bosso, F. C.*: Computer controlled test system, *Instrum. and Control Systems*, Nov. (1970) 25...26. old.
- [35] *Favour, J. D.—Lebrun, J. M.*: Transient synthesis for mechanical testing, *Instrum. and Control Systems*, Sept. (1970) 125...127. old.
- [36] *Trimble, Ch. R.*: Digital testing — an instrumentation viewpoint, *ASSC Record* (1970) 138...141. old.
- [37] *Bohus, M.—Geher, K.*: Logikai hálózatok számítógépes vizsgálata. *Híradástechnika XXIII. éf. 7. sz.* (1972) 199...203. old.

- [38] *Reggins, C.:* Automatic testing for logic circuits — benefits and drawbacks, Nat. Electronic Packaging and Production Conf. (1971) 335...339. old.
- [39] *Van Veen, F.:* An introduction to IC testing, IEEE Spectrum, Vol. 8. No. 12, Dec. (1971) 28...37. old.
- [40] *Cargile, W. P.:* A computer-controlled system for testing digital logic modules, Hewlett-Packard Journal, March (1969) 14...19. old.
- [41] *Gudnitz, L.—Tsuda, H.:* General-purpose test system gets digital capability, Hewlett-Packard Journal, July (1971) 2...9. old.
- [42] LSI digital test system functions under computer control, Computer Design, Sept. (1972) 46...48. old.
- [43] Model LSI-1002 LSI test system specifications, Minato Electronics.
- [44] *Sebestyén B.:* Kiszámítógépek alkalmazása a műszaki ellenőrzésben, KGM ISZSZI—VIDEOTON, III. kötet okt. (1972).
- [45] *Litzinger, J. E.—Friedmann, R. E.:* Hardware/software considerations in multiplexed element test systems, Proc. Comput. Designer's Conf. Anaheim, Calif. Jan. (1971) 199...209. old.
- [46] *Chapman, J. S.—Wichland, Ch. E.:* Utilization of a time shared central data system in a manufacturing test environment, WESCON (1971) 18/1, 1...7. old.
- [47] *Hogan, Ph. A.—Nelson, A. E.:* A time shared computer data system for automated test stations, WESCON (1970) 16/4.
- [48] *Allain, W. L.:* Integrated instrument setup test intricate assemblies, Electronics, Oct. 9 (1972) 104...109. old.

2.

Mérőhálózati alapelemek

2.1. Analóg építőelemek

Műveleti erősítő

A műveleti erősítő aszimmetrikus (ritkábban szimmetrikus) kimenetű differenciaerősítő és esetenként változó külső csatolóelemek együttese, melyre első közelítésben az alábbi feltételezések fogadhatók el:

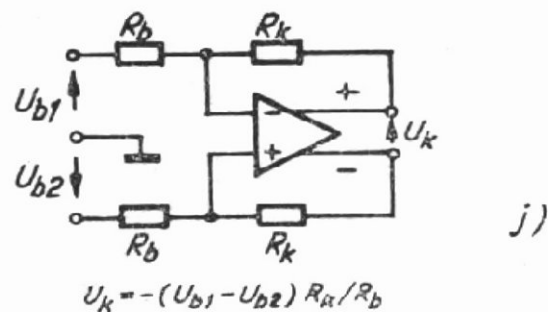
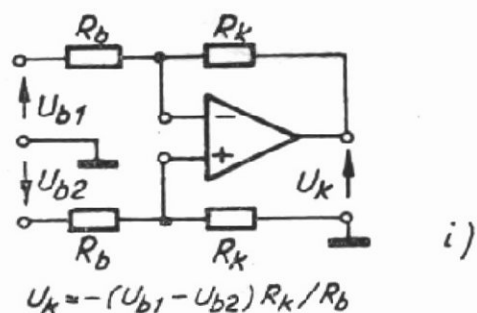
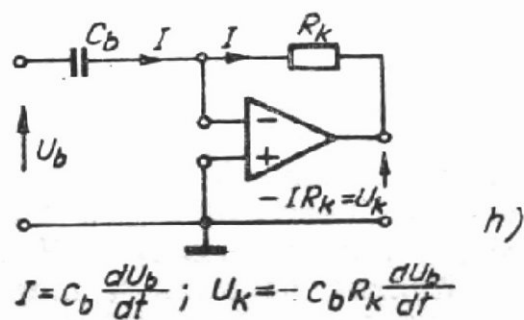
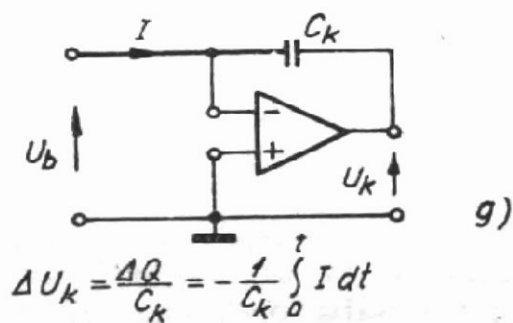
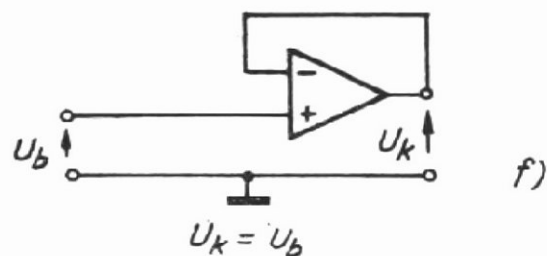
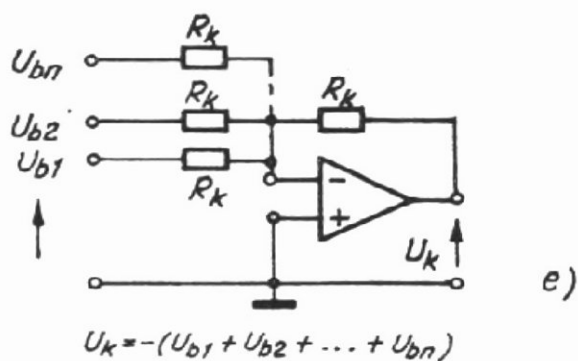
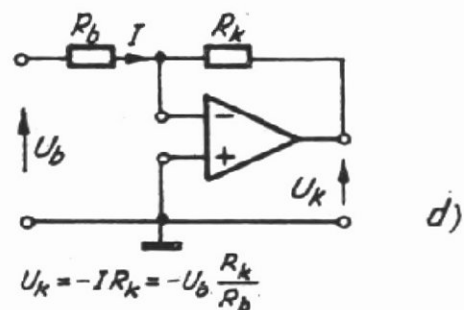
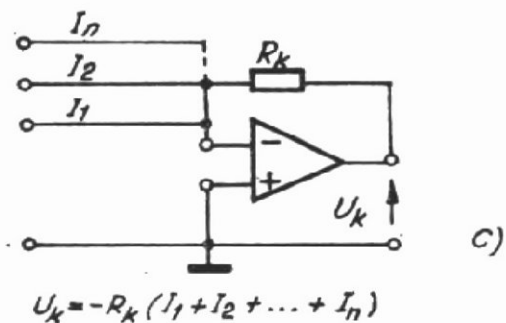
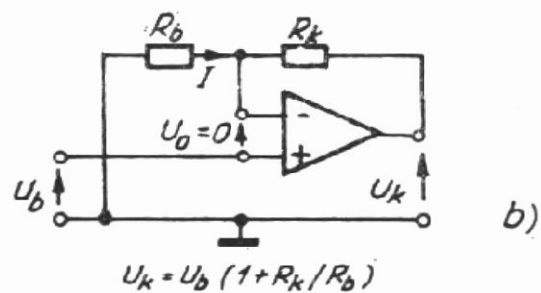
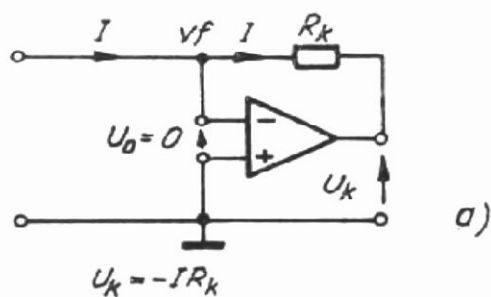
1. Nyitott hurkú erősítése (A) megközelíti a végtelent.
2. Alaperősítőjének (nyitott hurkú) sáv szélessége megközelíti a végtelent.
3. Alaperősítőjének bemenő impedanciája végtelen.
4. Kimeneti impedanciája zérus.
5. Válaszideje (a be- és kimenet közötti késés) zérus.

E jellemzőkkel leírt — ideális — műveleti erősítő jelentős alkalmazási területeken mint modell, a valóság első közelítésének tekinthető. Egyszerűségénél fogva igen alkalmas arra, hogy vele a műveleti erősítő áramköri változatainak funkcionális tulajdonságait áttekinthessük. Mivel a következőkben e funkcionális tulajdonságokra helyezük a súlyt, a műveleti erősítő valóságos viszonyait csak röviden érintjük.

A műveleti erősítő: visszacsatolt erősítő. Változatai két alapáramkörből, a 2.1a ábra invertáló áramkövetőjéből, valamint a 2.1b ábra nem invertáló feszültség-erősítőjéből származtathatók le. Az ábrákon a nem invertáló bemenetet +, az invertáló — jelöli. Az R_k és R_b — általában ohmos — csatolótagok helyén esetenként reaktív vagy aktív elem is állhat.

Az *áramkövető* a bemenetére érkező áramot teljes egészében (3. feltétel) az R_k ágba tereli, melynek árama tehát független R_k értékétől. Az 1. feltétel értelmében az erősítő lineáris működési tartományában mindenkor $U_0=0$ kell, hogy legyen, s ezért vf pont I bármely értékénél földpotenciálú. E sajátosságáért, a vf pontot virtuális földnek nevezik. A mondottakból következően az áramkövető kimenetén $U_k = -IR_k$ feszültség lép fel. A virtuális föld nulla potenciálja folytán az I több független komponensből is összetehető anélkül, hogy az egyes áramutak egymásra hatnának (2.1c ábra); a vf pontot ezért *áramösszegező pontnak* is nevezik. Az áramösszegező pontra R_b ellenállást kapcsolva, R_b bemenő ellenállású, R_k/R_b erősítésű, invertáló feszültség-erősítőt kapunk (2.1d ábra). Az áramösszegező ponton n számú R_k nagyságú ellenállással az áramkövető feszültségösszegezőként működik (2.1e ábra).

A *nem invertáló feszültség-erősítő* (2.1b ábra) invertáló bemenetén $U_k \cdot R_b / (R_b + R_k)$ feszültség uralkodik, mivel azonban feltételezésünk szerint $U_0=0$, egyúttal $U_b = U_k \cdot R_b / (R_b + R_k)$. Következően: $U_k = U_b(1 + R_k/R_b)$; a zárójeles tényező nem más, mint az erősítés. $R_k \rightarrow 0$ és $R_b \rightarrow \infty$ közelítéssel impedanciáttranszformációt megvalósító (a végtelen nagy bemenő impedanciát nullára transzformáló) feszültségkövető (2.1f ábra) áll elő.



2.1. ábra.

A műveleti erősítő alapáramkörei és alkalmazási változataik

(a) invertáló áramkövető; (b) neminvertáló feszültség erősítő; (c) áramösszegező; (d) invertáló feszültség erősítő; (e) feszültségösszegező; (f) feszültségkövető; (g) áramintegrátor; (h) differenciáló elem; (i) szimmetrikus bemenetű, aszimmetrikus kimenetű erősítő; (j) szimmetrikus bemenetű, szimmetrikus kimenetű erősítő

Az áramkövető az R_k ellenállás helyén C_k kapacitással áramintegrátorként (2.1g ábra), a bemenetre kapcsolt C_b kapacitással pedig differenciáló elemként (2.1h ábra) működik. Ez utóbbit illetően azonban meg kell jegyezni, hogy ebben az egyszerű alakjában a stabil működés követelményeinek ellentmondó frekvenciaátviteli görbével rendelkezik. A valóságban alkalmazott differenciáló elemek — az instabilitás elkerülése érdekében — csatolótagjaik felépítésében bizonyos mértékben eltérnek a 2.1h ábra kapcsolásától.

Végül az $R_b - R_k$ tagot az erősítő másik bemenetén is alkalmazva, egyszerű módon alakítható ki *szimmetrikus bemenetű*, aszimmetrikus (2.1i ábra), ill. szimmetrikus kimenetű erősítő (2.1j ábra).

A valóságban az imént felsorolt öt feltétel nem elégül ki.

Az alaperősítő erősítése nem közelíti meg a végtelent, bár értéke általában jelentős ($A = 10\,000 \dots 500\,000$). A visszacsatolt operatív erősítő G erősítése a visszacsatolási tényező figyelembevételével az ismert

$$G = \frac{A}{1 + A\beta}$$

összefüggésnek tesz eleget. $A \rightarrow \infty$ esetén $G \rightarrow 1/\beta$. A valóságos erősítő annál jobban megközelíti az ideálist, minél nagyobb az $A\beta$, ún. hurokerősítés, tehát ha $A\beta \gg 1$. $A = 10^5$ és $\beta = 10^{-1}$ esetén pl. a G értéke már csak a negyedik számjegyben tér el az ideális erősítőhöz képest.

A nyitotthurkú erősítő frekvenciamenete a valóságban a 2. feltételezéstől eltérően a Bode-görbét követi; a zárthurkú erősítő sáv szélessége a nyitthurkú sáv szélességének közelítően $A\beta$ -szorosa.

A rendszertervezés szempontjából fontos adat: a *beállási idő*. A beállási idő részben az erősítő tulajdonságaitól, részben azonban az üzemi körülményektől, továbbá, az előírt hibakorlától is függ. Értékének előre meghatározása, ill. megadása ezért nehézségekbe ütközik. Egyes esetekben a különböző nagyságú hibakorlátokhoz tartozó beállási idő értékét — az ugrásjel amplitudójának függvényében — görbék alakjában adják meg.

A nyitotthurkú erősítőnek különbségi, ill. közös bemenő ellenállása különböztethető meg. A *közös ellenállás* az egymáshoz zárt bemenetek és a föld között jelentkezik; értéke általában $> 10^7 \Omega$. A *különbségi ellenállás* a bemenő kapcsok között jelentkezik; értéke az alkalmazott erősítő felépítésétől függően, $10^4 \dots 10^{13} \Omega$. A zárthurkú erősítő bemenő impedanciája ehhez képest jelentősen eltér. Az áramkövető bemenő impedanciája — jó közelítéssel — az áramösszegező pontra kapcsolt R_b ellenállás értékével egyezik meg. A nem invertáló feszültségerősítő bemenő impedanciája pedig az $A\beta$ hurokerősítéssel szorzott nyitotthurkú bemenő impedancia.

A nyitotthurkú *kimenő impedancia* néhány száz ohm nagyságrendű. Értékét azért kell figyelembe venni, mert a kimeneten általában jelenlevő parazitív kapacitással képzett időállandója befolyásolja a kimenő jel emelkedési arányát. Másrészt a kimenő impedancia, terhelés esetén (a terhelés mértékétől függő) feszültségesést idéz elő, miáltal csökken a nyitotthurkú erősítés. A zárthurkú erősítő kimenő impedanciája a nyitotthurkú erősítőéhez képest — jó közelítéssel — a hurokerősítés arányában csökken. Néhány száz ohm kimenő impedancia és > 1000 hurokerősítés esetén a zárthurkú erősítő kimenő impedanciája $< 1 \Omega$, ami a legtöbb alkalmazási esetre vonatkoztatva indokolja a 4. sz. feltételezést. Itt és minden más esetben is, szem előtt kell tartani azonban $A\beta$ frekvenciafüggését.

A *virtuális föld* potenciálja a valóságban a földpotenciáltól eltér, tehát $U_0 \neq 0$. Az eltérés mértékét a tényleges viszonyoknak megfelelő értékekkel szemléltetve: $A = 100\,000$ és $U_{k\max} = 10\text{ V}$ esetén, $U_0 = 10^{-4}\text{ V}$. Feltételezve $10^5 \Omega$ különbségi bemenő ellenállást, az U_0 hatására ez ellenálláson folyó áram $I_0 = 10^{-4}/10^5 = 10^{-9}\text{ A}$. Tehát

U_0 és I_0 a gyakorlatban szokásos áramköri viszonyok mellett, az alkalmazási esetek nagy részében, valóban elhanyagolható. Figyelembe kell venni azonban, hogy A értékének csökkenése mértékében U_0 és I_0 értéke is arányosan nő, és ez a növekedés akkor is bekövetkezik, ha A csökkenése a véges sávzélesség következménye.

A sávzélesség helyett — a linearitás tartományában — ekvivalens jellemzőként az erősítő felfutási ideje, tehát egy ugrásjelre kapott válaszjel *felfutási ideje* is megadható.

Az esetben, amikor a bemenő jel az operatív erősítő fokozatainak valamelyikét áramtartományának határáig vezérli ki, az erősítő kimenetének az időegységre eső változása maximumot ér el. Ezt a maximumot *emelkedési aránynak* nevezik.

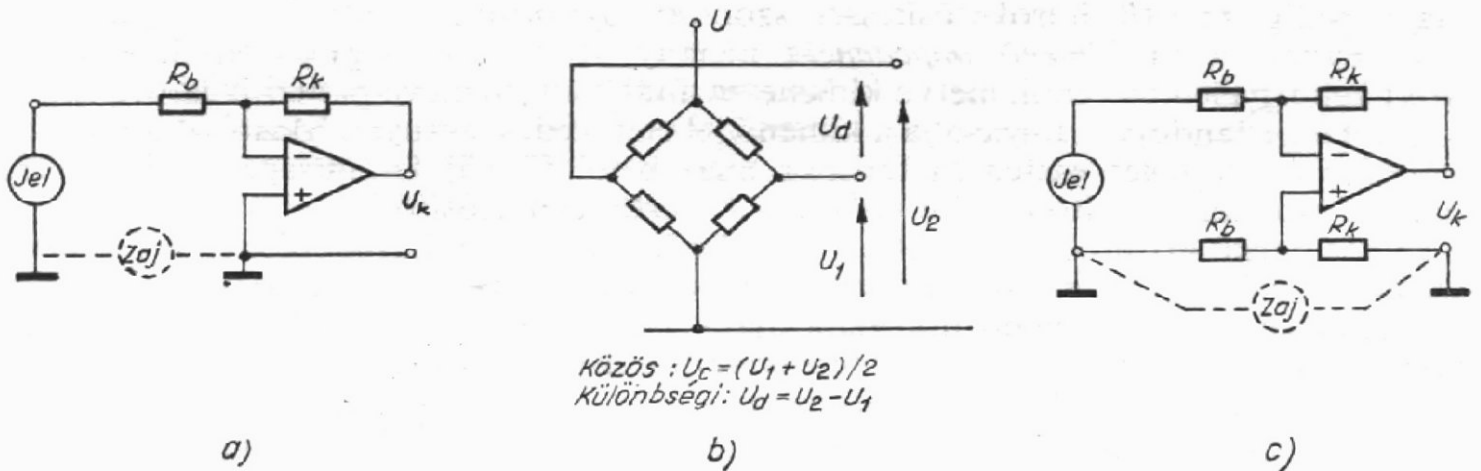
Az ideális műveleti erősítő kimenő feszültsége $U_k=0$, ha $U_b=0$. A valóságban $U_b=0$ -nál a kimeneten bizonyos feszültség található. Azt a bemenő feszültséget, mely ezt a kimeneti feszültséget nullára állítja vissza, *nullponthibának* (offset) nevezik. Bizonyos alkalmazási esetekben a nullponthiba elhanyagolható. Nagyobb igényű erősítők beépített nullpont-korrekcióval rendelkeznek. Szükség esetén a korrekciós áramkör az erősítőhöz kívülről is applikálható.

Az ideális erősítő a földhöz képest mindkét bemenetén jelentkező közös jelkomponenst nem érzékeli. A valóságban az áramkörök aszimmetriája miatt a közös feszültség is okoz kimenő feszültség változást. *Közös feszültség elnyomási tényezőnek* (KFE) az U_c közös feszültség, és az általa kiváltott kimenő feszültséget semlegesítő U_{bs} bemenő differenciajel hányadosát nevezik, db-ben kifejezve, azaz $KFE = -20 \log_{10} (U_c/U_{bs})$. Az egyik bemenetén földelt erősítőnél közös feszültség nem léphet fel. A közös feszültség által okozott hibajel nem a invertáló és a differenciaerősítőként működtetett műveleti erősítő sajátja.

Az összes felsorolt jellemző többé-kevésbé hőmérséklet- és tápfeszültség-érzékeny (drift; vándorlás). Precíziós mérések kapcsán szükséges lehet a tápfeszültség, sőt a hőmérséklet stabilizálása is, vagy esetleg olyan korrekciós áramköröket kell alkalmazni, melyek segítségével az erősítő mérés előtt hitelesíthető.

Mérőerősítő

Az áramkövető műveleti erősítő, feszültségerősítőként alkalmazva kizárja, hogy a bemenő kapcsokra közös feszültség jusson. Amennyiben azonban a jelforrás az erősítőtől nagyobb távolságra helyezkedik el és külön földelése van, úgy a két földpont között fellépő (pl. kúszóáramok által előidézett) zajfeszültség hozzáadódik a jelhez (2.2a ábra). Vannak másrészt olyan jelforrások, melyek jele eleve tartalmaz

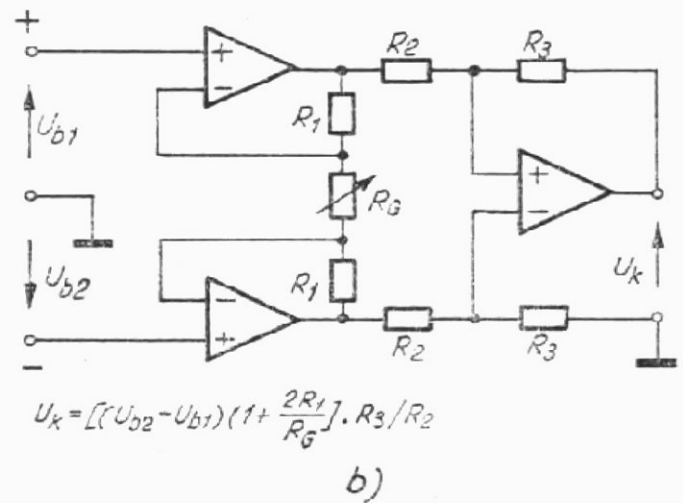
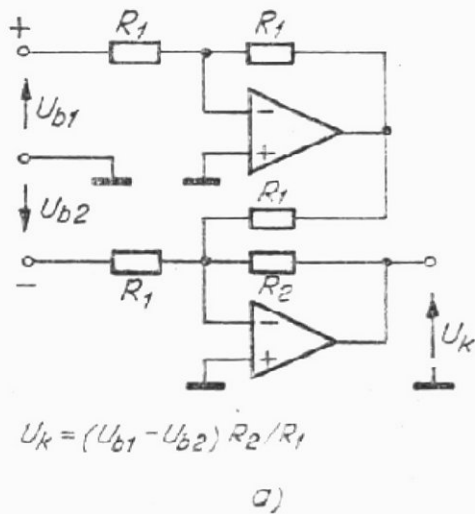


2.2. ábra.

Zajfeszültség — közös feszültség

(a) zajként jelentkező közös feszültség kettős földelés esetén; (b) érzékelőátalakító által keltett közös feszültség; (c) a zaj (közös feszültség) eliminálása ideálisan szimmetrikus erősítővel

közös feszültség komponensét (2.2b ábra), s ilyen esetben aszimmetrikus bemenetű erősítő nem is használható. A műveleti erősítőt szimmetrikus bemenettel alkalmazva a közös feszültség a csatolóágak és az erősítő belső elemeinek teljes szimmetriája esetén nem érvényesül (2.c ábra). A csatolóágak adjusztálásával az eredetileg többé-kevésbé kiegyenlített rendszer is elvileg elég jól kiegyenlíthető. Problémát jelenthet azonban, ha a jelforrás paraméterei (belső ellenállás, jelszint) és a mérési pontosság megadott követelménye alapján az R_b értékét $M\Omega$ nagyságrendűre kell választani, s ugyanakkor az erősítő jelentékeny erősítéssel kell, hogy rendelkezzen. Ez esetben ui. az R_k értéke a tíz vagy száz $M\Omega$ -os tartományba esik. Ismeretes, hogy ezen nagy értékű ellenállások adjusztálása, de még inkább hőmérsékleti és időbeli stabilitása problematikus. A nagy bemenő ellenállás egyébként zaj és hőmérsékleti stabilitás szempontjából az erősítőre általában is kedvezőtlen. A legnagyobb nehézség a közös feszültség elnyomás követelménye tekintetében akkor lép fel, ha az erősítést valamilyen oknál fogva változtatni kell. Az erősítés változtatását ui. éppen azokban az ellenállásokban végzik, melyek adjusztálásával a KFE-t optimumra állítják. A nehézségek kiküszöbölése érdekében több műveleti erősítőből összetett erősítőket alkalmaznak. A 2.3a ábrán két műveleti erősítőből álló együttes



2.3. ábra.

Közös feszültség elnyomásra optimalizált erősítők

(a) szimmetrikus bemenetű erősítő egyszeres erősítésű fázisfordítóval; (b) mérőerősítő

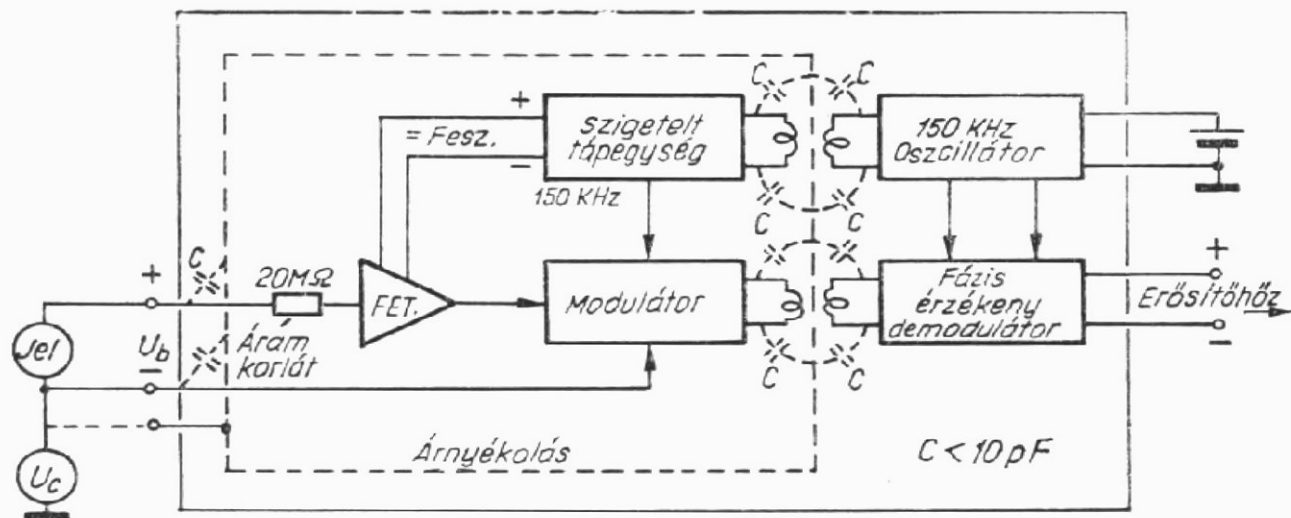
látható, mely a bemenő ellenállással kapcsolatos problémákat ugyan nem oldja meg, de erősítése az R_2 változtatásával olyképpen módosítható, ami a KFE-re optimalizált bemenő szakasz kiegyenlítését nem érinti.

A 2.3b ábra három egységből álló erősítője mind a bemenő ellenállás, mind a KFE, mind pedig az erősítés-beállítás oldaláról mutatkozó problémák kedvező megoldását nyújtja. A 2.3b ábrán vázolt erősítőt egyetlen áramköri egységként tekintve, *mérőerősítőnek* nevezik. A mérőerősítőt mint integrált áramköri egységet, R_1 , R_2 , R_3 ellenállásaival együtt, KFE szempontjából optimalizálva állítják elő. Hozzá csupán egyetlen külső ellenállást, R_G -t kell csatolni. A mérőerősítők általában rendelkeznek két olyan (referencia és érzékelő) kivezetéssel is, melyek alkalmazása a műveleti erősítőknél nem szokásos. E kivezetések felhasználásával bizonyos feladatok kapcsán a mérőkör egyszerűsödik.

A program által vezérelhető mérőerősítőt *adaterősítőnek* nevezik.

Szigetelt bemenetű jelerősítő

A mérőerősítő közös feszültség tartománya néhány voltra korlátozódik. Nagyobb (száz volt, esetleg kV nagyságrendű) közös feszültség esetén, vagy ha a mérőerősítővel elérhető KFE nem kielégítő, szigetelt erősítőket használnak. A fázismodulációt alkalmazó szigetelt erősítő (2.4. ábra), vagy helyesebben nevezve erősítőszigetelő



2.4. ábra.
Szigetelt erősítőbemenet

(Analog Devices), árnyékoló dobozban helyet foglaló izolált előfokozatból és hozzá csupán inductíve csatlakozó detektoregységből áll. Az előfokozat lényegi része: a jel által vezérelt fázismodulátor. A fázismodulátor alapjelét külső (pl. 150 kHz-es) oszcillátor állítja elő. Ugyanez az oszcillátor szolgáltatja a szigetelt tápegység energiáját is. A detektor: fázisérzékeny demodulátor, amely az U_b bemenő jelet polaritáshíven képezi le a tulajdonképpeni erősítő bemenetére. A felvázolt erősítőrendszer közös bemenő impedanciája $>10^{12} \Omega$, közös feszültségelnyomási tényezője pedig — hálózati frekvenciára vonatkoztatva — mintegy 120 db.

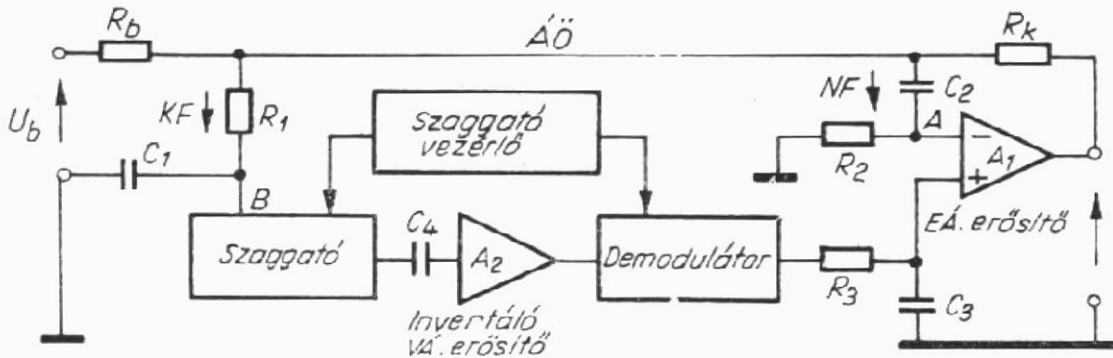
Az erősítő bemenetének szigetelésére optikai csatolókat is alkalmaznak.

A szigetelőerősítők közös feszültségtartománya általában 1...2 kV határig terjed, közös feszültségelnyomási tényezőjük pedig frekvenciafüggő. Az optikai csatolást fényemittáló dióda és fototranzisztor közé iktatott száloptikával oldva meg, nő a közös feszültség elnyomás, s értéke egyúttal frekvenciafüggetlenné is válik, a szigetelési feszültség pedig a MV tartományig terjeszthető ki.

Szaggató-stabilizált erősítő

A szaggatott bemenetű erősítő az egyenáramú komponenst is tartalmazó jelet váltakozó feszültséggé alakítja át, váltakozó áramú erősítőként erősíti, majd a felerősített jelet egyenáramú komponensével együtt rekonstruálja. Az erősítő egyenáramú nullponthibája és hőmérsékleti munkapont-vándorlása az eljárás folyamán nem szuperponálódik a jelre, s ezért a szaggatott bemenetű erősítő igen jó munkapont-stabilitással rendelkezik. Hátránya viszont, hogy mert a szaggató frekvenciája szokás szerint $<500 \text{ Hz}$, sáv szélessége rendkívül kicsiny. A szaggatott bemenetű erősítő előnyeit — a sáv szélesség megőrzése mellett — a szaggató-stabilizált erősítő hasznosítja (2.5. ábra). Működésének az a lényege, hogy az operatív erősítő áramösszegezési pontját két irányba ágaztatja el. A jel nagyfrekvenciás komponensére

vonatkozóan, az ábrán A-val jelölt pont tekinthető áramösszegező pontként. Ugyanakkor a kisfrekvenciás komponensre vonatkoztatva az áramösszegező pont B lesz. Mind a kisfrekvenciás, mind a nagyfrekvenciás komponensre vonatkozó zárthurkú erősítés az áramösszegező pont e kettőssége ellenére $\approx R_k/R_b$. Látható azonban, hogy míg a nagyfrekvenciás komponens hurokerősítése βA_1 , a kisfrekvenciás komponensé $\beta A_1 A_2$. Az egyenáramú erősítő nullponthibája, munkapont-vándorlása az ÁÖ pontra tekintve elvileg A_2 arányában csökken. A gyakorlatban a felvázolt stabilizálási eljárással mind a nullponthiba, mind a hosszú idejű stabilitás tekintetében egy-két nagyságrendnyi javulás érhető el.



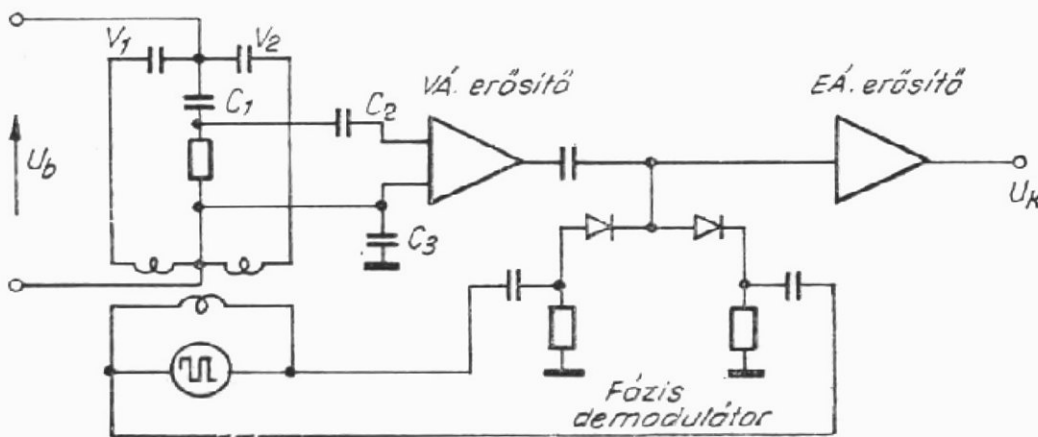
2.5. ábra.

Szaggató-stabilizált erősítő

EA — egyenáramú; VÁ — váltakozó áramú; KF — kisfrekvenciás; NF — nagyfrekvenciás

Varaktorhid bemenetű erősítő (2.6. ábra)

Jellemzője 10^{-14} A bemenő átvezetési áram, amelyben felülmúlja a térvezérlésű tranzisztort, vagy elektrométercsövet alkalmazó erősítőket. A nagyfrekvenciás varaktordiódás (V_1, V_2) transzformátorhíd $U_b = 0$ bemenő jelnél teljesen szimmetrikus. A varaktordiódák kapacitása az U_b nagyságától és polaritásától függő mértékben és irányban változik, s a híd e változásnak megfelelő kimenő jelet bocsát (C_1, C_2 elválasztó kondenzátorokon át) az őt követő váltakozó áramú erősítő bemenetére. A váltakozó



2.6. ábra.

Varaktorhid bemenetű erősítő

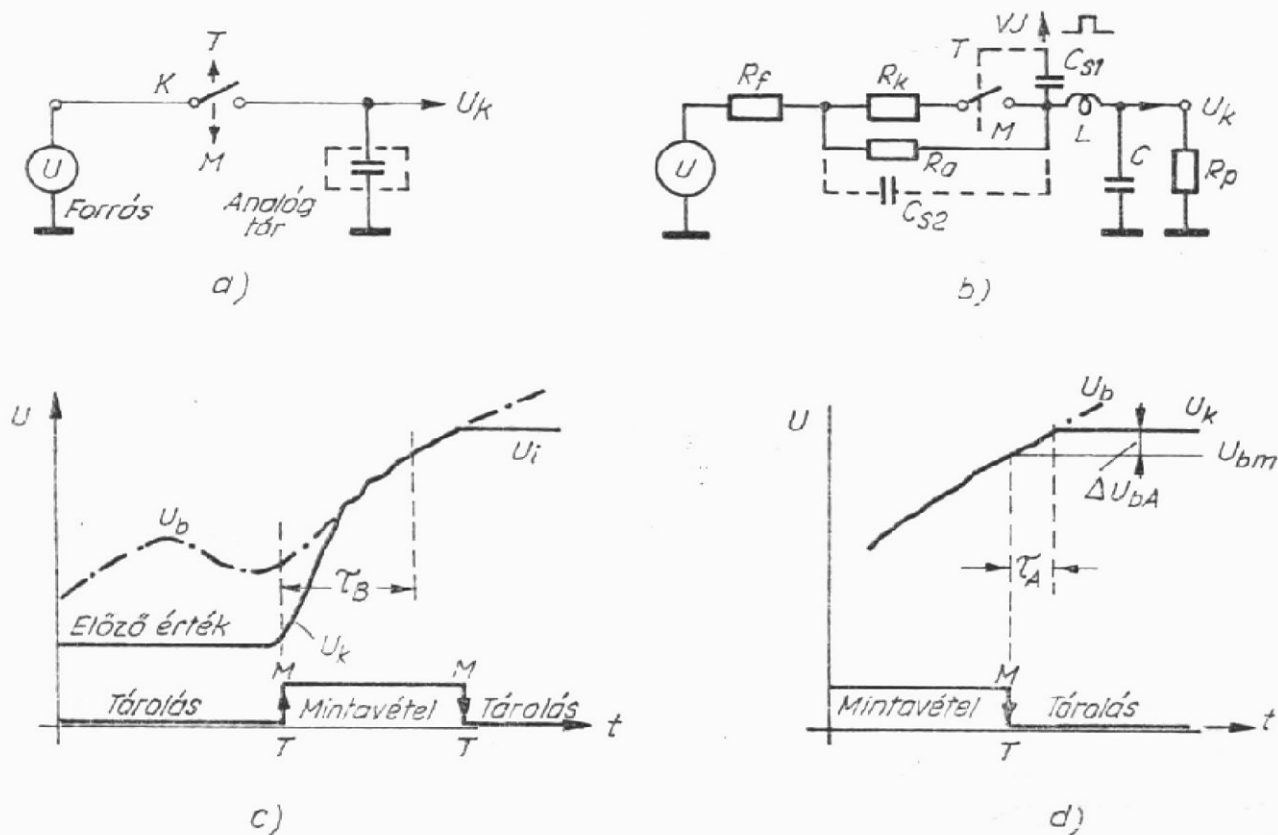
áramú erősítőfokozaton a kisfrekvenciás zajkomponens nem jut át. A felerősített és fázishelyesen rekonstruált jel egyenáramú erősítő bemenetére kerül. Az erősítő megengedhető közös feszültsége a transzformátor és a kondenzátorok megválasztásától függően néhány száz volt lehet. A közös feszültség elnyomási tényező értéke jelentékeny.

Töltésérzékeny erősítők

Bizonyos jelforrások (pl. piezoelektromos érzékelők; félvezető részecske detektorok) a vizsgált mennyiséggel arányos töltést szolgáltatnak. Ha a töltés teljes egészében kondenzátorban halmozható fel, e kondenzátor feszültsége a töltés nagyságát tükrözi. Töltés—feszültség átalakítása céljára a 2.1g ábra műveleti erősítőjének változatait alkalmazzák.

Tároló mintavevők

A tároló mintavevő — vagy röviden mintavevő — valamely jelforrás t_i időponthoz tartozó U_i feszültségének felvételére, a felvett feszültség relative hosszú idejű tárolására és szolgáltatására rendelt áramkör. Lényeges elemei: az analóg tár és a K kapcsoló (2.7a ábra). A mintavétel időpillanatában a kapcsoló zár (M állapot),



2.7. ábra.

Tároló mintavevő

(a) elvi működési séma; (b) a mintavétel pontosságát befolyásoló áramkörtényezők; (c) a begyűjtési idő (τ_B) szemléltetése; (d) a bontási idő (τ_A) és az apertúrahiba (ΔU_{bA}) összefüggése

a tárolás ideje alatt nyitva áll (T állapot). Ideális esetben a mintavétel végtelen rövid ideig tart, és a felvett feszültség tetszőlegesen hosszú időn át változatlan szinten marad.

A valóságban (2.7b ábra) a kapcsolóval sorban fekvő R_f forrásellenállás, R_k „kontaktus ellenállás”, valamint a szórt reaktanciák jelenléte miatt, mintavételkor bizonyos időt vesz igénybe, míg az analóg tárolóként alkalmazott C kondenzátor feszültsége a forrás feszültségét megközelíti. Azt az időtartamot, amely alatt a kondenzátor feszültsége (a szórt induktivitás és kapacitás okozta tranzienseket is figyelembe véve) a forrás feszültségét előírt hibataromány átlépésével közelíti meg, begyűjtési időnek (τ_B) nevezik (2.7c ábra). A begyűjtési idő az RC időállandó csökkentésével, egyúttal a szórt elemek lehető kiküszöbölésével rövidíthető (R a kondenzátor eredő töltő-

ellenállása az M időszakban). A begyűjtési idő nem csak áramkör-, hanem jelfüggő is. Értékét a teljes kitérés nagyságú feszültségváltozásra definiálják.

A kapcsolót működtető logikai jel $T \rightarrow M$ frontja a szórt kapacitáson át a kondenzátorba töltést injektál, az $M \rightarrow T$ front a kondenzátorból töltést von el. Ez utóbbi a felvett értékre $M \rightarrow T$ eltolódási hibaként szuperponálódik. A kapcsolási frontok által előidézett tüskék is — bizonyos esetekben — zavarólag hatnak.

A mintavétel záró időpontját a kapcsolójel $M \rightarrow T$ átmenete jelöli ki. A véges kapcsolási sebesség miatt azonban a bontás csak τ_A idővel később következik be; közben a forrás jele megváltozhat. Azt a ΔU_{bA} értéket, mellyel a ténylegesen felvett U_k jel az $M \rightarrow T$ időpontbeli U_{bm} értéktől eltér, *apertúrahibának* nevezik (2.7d ábra). Látható, hogy az apertúrahiba τ_A -n kívül a jel változási sebességének is függvénye. Tervezélrészű tranzisztorkapcsoló esetében τ_A néhány (átlagosan 5...40) nanoszekundum nagyságú, ami a mintavevő alkalmazásának legtöbb esetében nem okoz figyelembe veendő hibát. Hatása egyébként, értékének pontos ismeretében azáltal kompenzálható, ha az $M \rightarrow T$ frontot τ_A idővel előbbre hozzuk. Kompenzálás után adott esetben már csak a τ_A értékében mutatkozó határozatlanság okoz hibát.

Az apertúrahiba értéke szinuszos jel feltételezésével egyszerűen kiszámítható. Figyelembe véve, hogy a szinuszgörbe a nullátmenetnél a legmeredekebb, a τ_A idő alatt bekövetkező feszültségváltozás:

$$\Delta U_{bA} = \tau_A \left(\frac{dU_b}{dt} \right)_{t=0} = \tau_A \frac{d}{dt} (U_{b0} \sin \omega t)_{t=0} = U_{b0} \omega \tau_A.$$

A relatív apertúrahiba pedig:

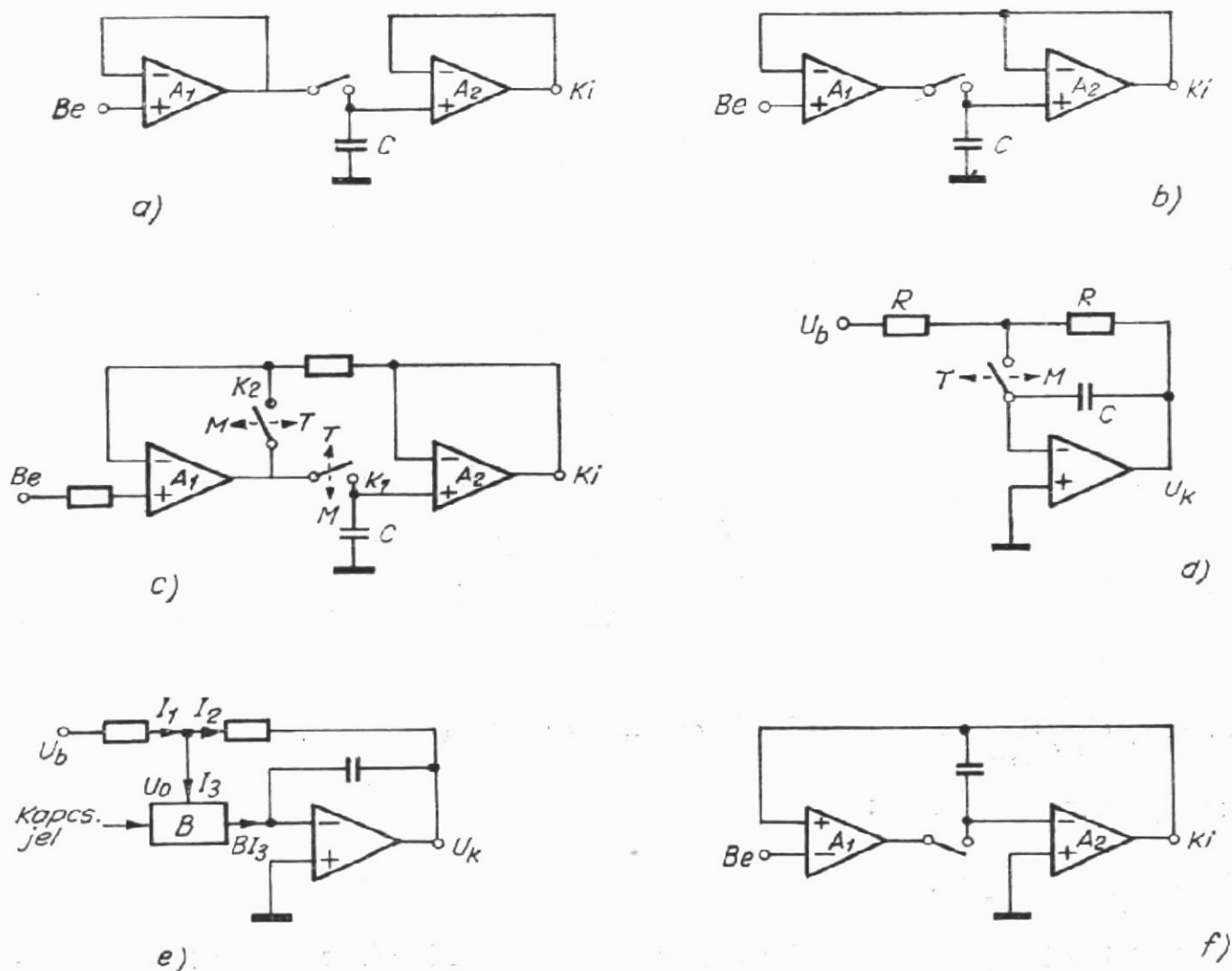
$$RAH = \frac{\Delta U_{bA}}{U_{b0}} = 2\pi f \tau_A.$$

A kapcsoló késésének kompenzálása esetén τ_A helyére a határozatlanság $|\Delta\tau|$ értéke kerül.

Az analóg tárolás viszonyai éppúgy nem eszményiek, miként a mintavételéi sem. A tárolás időszakában a kondenzátort a kapcsoló R_a átvezetése, másrészt saját szivárgási ellenállása és a hozzá csatlakozó mérőkör bemenetének R_p ekvivalens ellenállása terheli. E terhelés következtében a mintavétel folyamán nyert feszültség a kondenzátor sarkain az idővel változik. A változás mértékszámát esési sebességnek ($\mu\text{V/s}$) nevezik. A feszültségváltozás okozta hiba az idővel, az esési sebesség arányában nő.

A 2.8a...c ábra neminvertáló mintavevőket mutat be. Az egyszerű modelltől ezek abban térnek el, hogy a tárolókondenzátor mindkét irányban impedancia-transzformátorral csatlakozik a külvilághoz. Az A_1 nagy bemenő ellenállású feszültségkövető a jelforrást nem terheli, ugyanakkor a mintavételi ciklus során a tárolókondenzátort igen kis belső ellenállású forrásként tölti. A töltés időállandóját C -n kívül lényegében a kapcsolóként alkalmazott tervezélrészű tranzisztor vezető állapotú ellenállása determinálja, feltéve, hogy a töltőáramot nem az erősítő kimeneti határárama vagy a kapcsolóként alkalmazott tervezélrészű tranzisztor nyelő telítési árama korlátozza. A külső terhelést illesztő tervezélrészű tranzisztor erősítő bemenő ellenállása a kondenzátor számára elhanyagolható szivárgási utat jelent; az esési sebességet nem is annyira ez, hanem a nyitott tranzisztor-kapcsoló átvezetése, továbbá a kondenzátor saját szivárgási árama determinálja. A tárolási időnek a gyakorlatban számbajövő értékei mellett az esés (mely növekedés is lehet) okozta hiba elhanyagolható.

A 2.8a ábra mintavevőjének negatív vonása: a két erősítő-kaszád nullpont-hibájának összegeződése. A 2.8b ábrán látható változat a visszacsatoló hurok kiterjesztésével ezt a negatívumot kiküszöböli, ugyanakkor más előnyt is nyújt, ameny-



2.8. ábra.
Tároló mintavevő különböző változatai
(a...c) neminvertáló, (d...f) invertáló

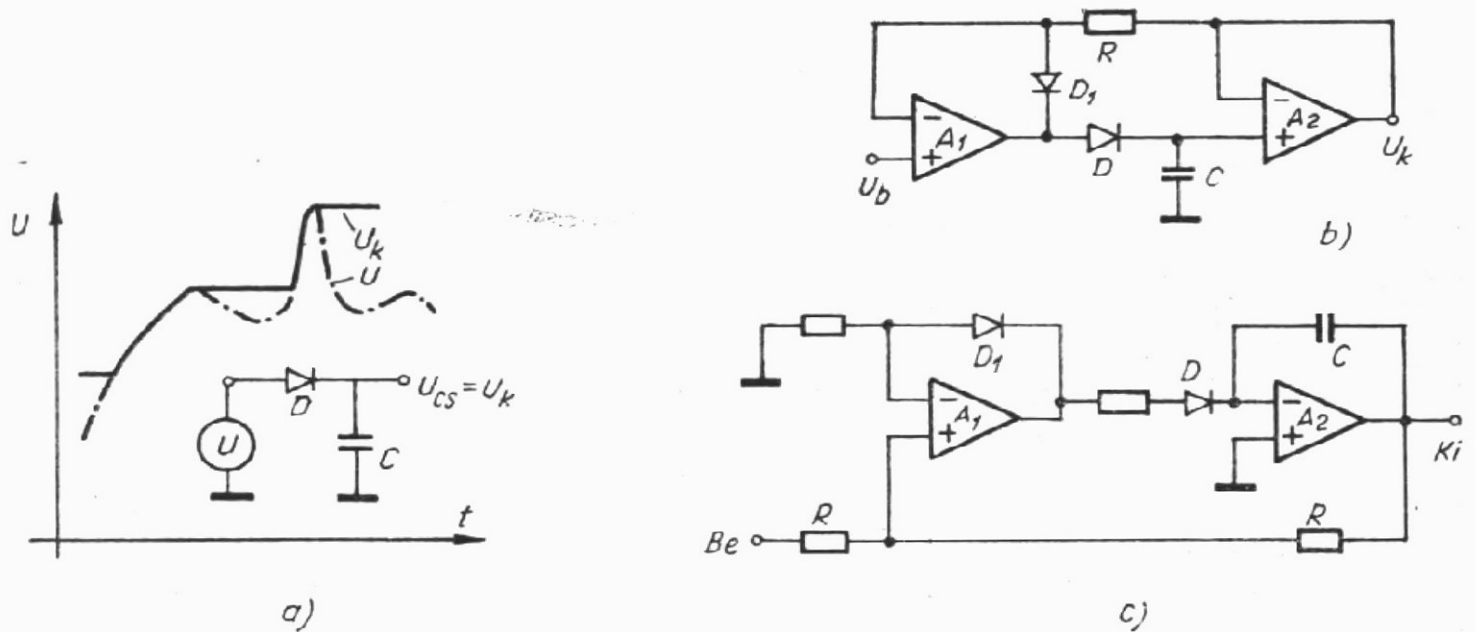
nyiben a kondenzátort végig a bemenő erősítő határáramával tölti, miáltal a bebegyűjtési idő lerövidül. További módosítást mutat be a 2.8c ábra. A K_2 kapcsoló a tárolási ciklusban zár, mintavételnél nyit, meggátolva, hogy K_1 kapcsaira nyitott állapotában jelentős feszültség essék, másrészt, hogy a bemeneti erősítő túlterhelődjék.

A nem invertálóval szemben, az invertáló mintavevő (2.8d...f ábra) tároló kondenzátorának kapcsoló felőli oldala, és ezzel a kapcsoló is földpotenciálon (virtuális föld) található. Ez a körülmény mind a kapcsolási idő, mind az átvezetési áram szempontjából kedvező. Az egyszerű invertáló mintavevő (2.8d ábra) nem elégíti ki a sebességre és bemenő ellenállásra vonatkozó egyidejű követelményeket; az RC időállandót R -rel csökkentve a bemenő ellenállás is esik. Az ellentmondás feloldásának egyik módja: B erősítésű áramerősítő kapcsoló alkalmazása (2.8e ábra); másik módja: erősítő előfokozat alkalmazása (2.8f ábra).

Csúcsdetektorok

A csúcsdetektor az időben változó feszültség által elért maximum rögzítésére szolgáló speciális mintavevő. A mintavevő fent ismertetett formáitól lényegében az által tér el, hogy a vezérlőjellel működtetett kapcsoló helyén szintdetektort (D) tartalmaz, mely akkor nyit, ha a forrás feszültsége valamely korábban már regisztrált maximumot meghalad (2.9a ábra).

Nem invertáló csúcsdetektort mutat be a 2.9b ábra. Az A_1 és A_2 erősítő szerepe ez esetben is impedanciáttranszformáció. Ezen kívül azonban az alkalmazott hurok a D szintdetektor nemlinearitásából és feszültségeséséből eredő mérési hibát is



2.9. ábra.
Csúcsdetektor

(a) működési séma; (b) nem invertáló, (c) invertáló változat

redukálja, mégpedig az A_1 nyitotthurkú erősítés arányában. D_1 dióda feladata, hogy $U < U_k$ esetében (amikor is D nyitott) az A_1 erősítő túlterhelődését meggátolja. A csúcsdetektorban alkalmazott erősítők túlszillapított állapotban kell, hogy működjenek; bármely túllövés ui. meghamisítja a C kondenzátorban tárolt értéket. Invertáló csúcsdetektort a 2.9c ábra szemléltet.

A bemutatott áramkörökben a D polaritását felcserélve, negatív csúcsdetektorhoz jutunk. Közös bemenetre kapcsolódó pozitív és negatív csúcsdetektorok kimeneteit szimmetrikus bemenetű műveleti erősítőre véve, csúcstól-csúsig detektor áll elő, mely két szomszédos (egy pozitív, egy negatív) csúcs közötti különbséget mér.

A tároló kondenzátor feszültségét minden mérési ciklus végén vissza kell állítani nullára; máskor a mérési eredményt hosszabb ideig tárolni kell. A csúcsdetektorokat e feladat ellátására szolgáló visszaállító és tárolás-üzemmód kapcsolók, ill. e kapcsolókat reprezentáló áramkörök egészíthetik ki. A tárolás-üzemmód kapcsoló a tároló kondenzátort leválasztja a bemenetről, a visszaállító kapcsoló a kondenzátort kisüti.

Komparátorok

Az analóg komparátor kétbemenetű erősítő áramkör; az egyik bemenetére adott jel szintjét a másik bemenetén uralkodó referenciaértékkel hasonlítja össze. Kimenete két logikai állapot egyikét veheti fel; egyik állapotból a másikba akkor kapcsol, amikor bemenetén a vizsgált jel áthalad a referenciaszinten. Ideális körülmények feltételezésével az átkapcsolás momentán módon, pontosan a komparálási szint elérésekor következik be, és differencia vezérlőfeszültséget nem igényel. A valóságban, a véges erősítés miatt, az átkapcsoláshoz a jel- és referenciaszint közötti bizonyos differenciafeszültség szükséges, továbbá figyelembe kell venni az erősítő nullpont-hibájának és bázisáramának a komparálási szintre gyakorolt hatását is. A nullponthiba egy adott hőmérsékletre kiegyenlíthető, a bázisáram hatása is jól kompen-

zálható egy a referenciaágba iktatott ellensúlyozó ellenállás alkalmazásával. Minden esetben tekintettel kell lenni azonban arra, hogy a hőmérséklet változásával a nullponthiba nagysága is megváltozik, s a megváltozás a kiegyenlített értékhez képest ismét csak hibajelként szuperpolálódik a referenciaszintre. A felsorolt tényezőkből származó hiba annál inkább érvényesül, minél kisebb a referenciaszint, s így leginkább a nullátmenet detektornál, melynek referencia pontja földpotenciálon van. A 2.10a ábrán bemutatott nullátmenet detektornál a bázisáramot kompenzáló ellenállás szerepét R_2 tölti be. A ZD zéner-dióda — határolóként — pozitív irányban a vezetési, negatív irányban a zéner-szintre korlátozza az erősítő kimenő feszültségét. E nullátmenet detektor hibája, hogy az $U_b \approx 0$ állapotban a bemenetre szuperponálódó zaj a kimenetet két állapota között ide-oda kapcsolhatja. Ennek elkerülése céljából az eredeti áramkört visszacsatoló ellenállás alkalmazásával (R_3) olyképpen módosítják, hogy a komparálási jelleggörbe hiszterézishurkot tartalmazzon (2.10b ábra). E komparátor határoló eleme, szemben az előbb látottal, szimmetrikus; mindkét irányban azonos értéken limitálja az U_k -t. A 2.10c ábra $U_r \neq 0$ referenciájú komparátort ábrázol karakterisztikájával együtt. A tetszőleges referenciaszinten működő komparátor a nullátmenet detektortól inkább belső tulajdonságai, mint felépítésmódjának logikája alapján különbözik; megfelelő közös feszültségűréssel és nagy közös feszültségelnyomási tényezővel kell, hogy rendelkezzenek.

A komparátor erősítő invertáló áramkövető kapcsolásban áramkomparátorként is alkalmazható. Az áramösszegező pontra a komparálási szintnek megfelelő nagyságú, de ellentétes irányú referenciaáramot vezetve a kimenet ott vált előjelet, ahol a jeláram a referenciaáramot éppen kiegyenlíti, azaz, amikor az áramösszegező ponton a $\Sigma I = 0$ helyzet áll elő. Az áramösszegező pontra ellenállásokat kapcsolva, az áramkomparátor feszültségkomparátorrá alakul át (2.10d ábra). Az áramösszegezés alapján működő komparátor az U_r értékétől függetlenül mindig földpotenciálon marad, és ez bizonyos gyakorlati alkalmazások kapcsán előnyt jelent a 2.10c ábra feszültségkomparátorával szemben. Többek között a közös feszültség kezelésének problematikája is elesik.

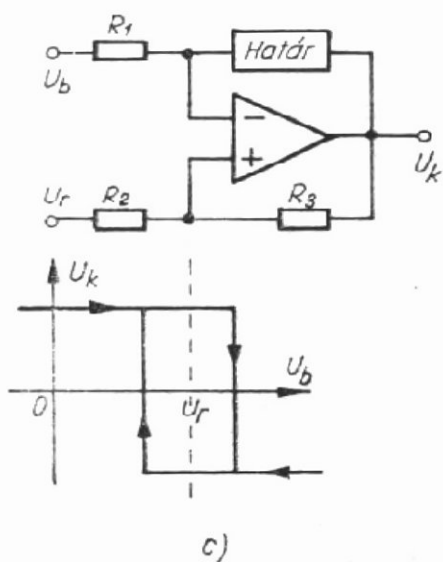
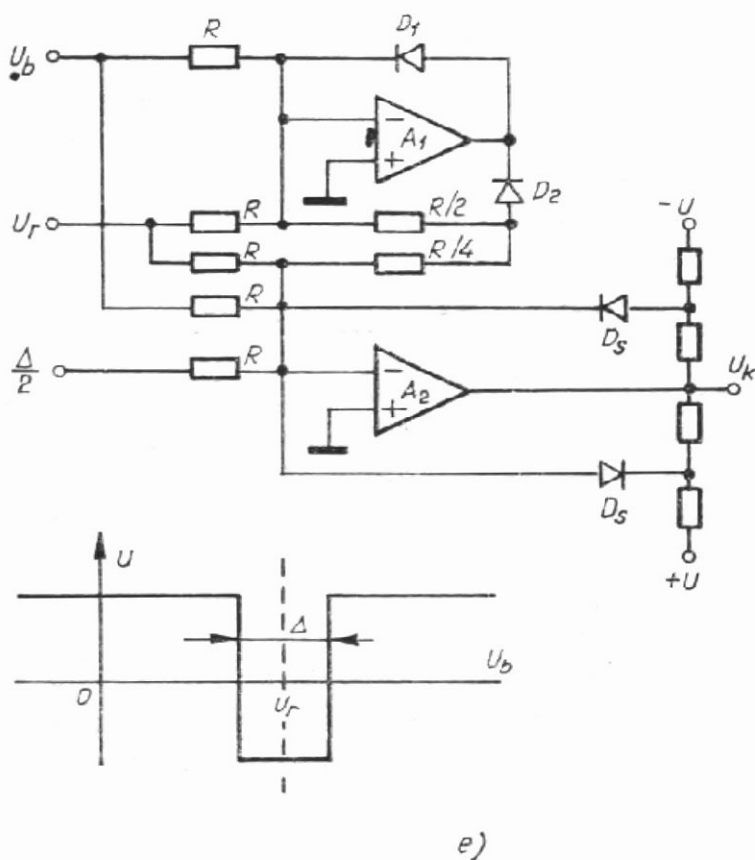
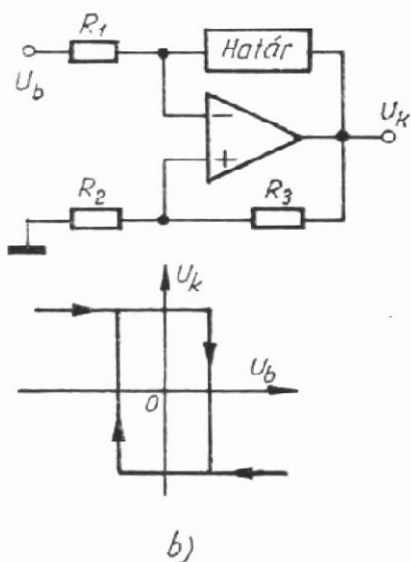
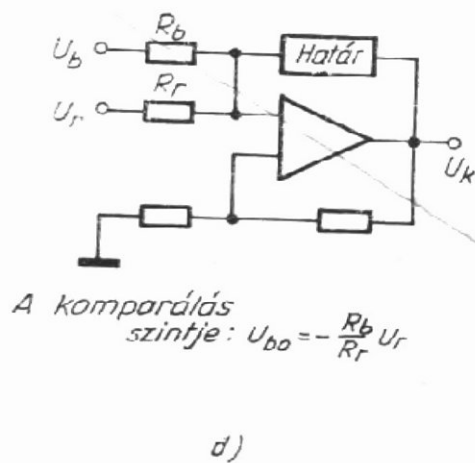
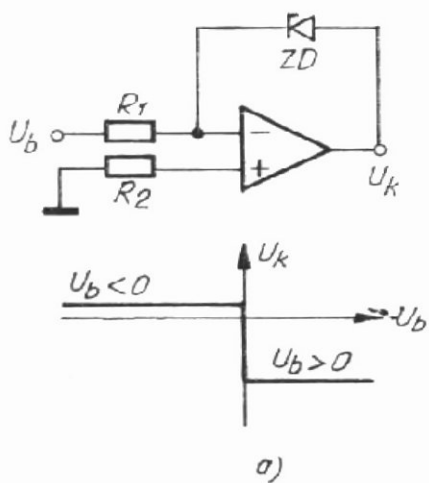
Az integrális komparátorokon kívül, melyek tehát azt jelzik, ha a jelszint egy bizonyos értéket elér, ill. meghalad, vannak differenciális vagy más néven ablakkomparátorok is. Az ablakkomparátornak két összehasonlítási szintje van; a szintek közötti sáv: az ablak. A komparátor ez esetben arról ad információt, vajon a jel értéke az ablakon belül vagy azon kívül esik-e. Az ablakkomparátorok feszültségösszehasonlító vagy áram-összehasonlító típusú integrális komparátorokból építhetők fel.

A 2.10e ábra ablakkomparátora két áram-összehasonlító komparátorból áll. Három bemenő pontja közül az első a jelszint (U_b), a második a referenciabemenet (U_r), a harmadik pedig az ablakbemenet. Δ ablakszélesség esetén az ablakbemenet $\Delta/2$ feszültséget kap. Az ablakfeszültség előjele olyan, hogy az A_2 komparátor szintjét negatív irányba tolja el. A_2 tehát $\Delta/2$ -vel alacsonyabb szinten kapcsol, mint A_1 . Az A_1 átkapcsolási szintje $U_{b1} = -U_r$, az A_2 -é tehát $U_{b2} = -U_r - \Delta/2$. Figyelembe veendő, hogy mindaddig, míg A_1 nem kapcsol át, az $R/4$ ellenálláson keresztül nem folyik áram, az átkapcsolást követően viszont ez az ág

$$- \frac{U_b + U_r}{R} \cdot \frac{R}{2} \Big/ \frac{R}{4}$$

áramot vezet az A_2 áramösszegező pontjára, ahová ily módon

$$\Sigma I = \frac{U_b}{R} + \frac{U_r}{R} + \frac{\Delta}{2R} - \frac{U_b + U_r}{R} \cdot 2$$



2.10. ábra. Komparátorok

(a) nullátmenet-detektor; (b) nullátmenet-detektor hiszterézissel; (c) U_r feszültség szint átlépését hiszterézissel detektáló áramkör; (d) áramkomparátor; (e) ablakkomparátor

összegáram folyik. $\Sigma I=0$ elérésekor az A_2 komparátor visszakapcsol eredeti állapotába. Ez, a ΣI -re felírt egyenlet alapján, $U_b + U_r = \Delta/2$ értéknél következik be. Az A_1 kimenetének logikai állapota az $(U_b + U_r) < 0$ állapotot különbözteti meg az $(U_b + U_r) > 0$ állapottól. Az A_2 kimenetének logikai állapota pedig azt jelzi, vajon az U_b szintje az ablakon belül vagy kívül fekszik-e.

A komparátorként alkalmazott erősítők, bár bizonyos áramköri kompozícióik a műveleti erősítők elvei alapján épülnek fel, a műveleti erősítőktől eltérő jellemzőkkel rendelkeznek. Nyílthurkú erősítési tényezőjük mindössze néhány ezer. A fokozatok erősítési tényezőjének növekedésével ui. nő a kisjelű felfutási idő, a fokozatok számának szaporítása meg a késési időt növeli. A komparátor erősítők további jellegzetessége a jelentős kisjelű sáv szélesség, valamint a meredek nagyjelű, emelkedési görbe. Tág, közös feszültség tartomány, ill. differenciafeszültség-tartomány bizonyos alkalmazási esetekben ugyancsak kívánatos. Az erősítő kimenetének a szóba kerülő logikai áramkörökhöz illeszkednie kell. Bár ismeretesek a komparátorok integrált áramköri változatai, a nagyobb igényű komparátorok gyakran jelenleg is kiemelkedő tulajdonságú diszkrét elemekből épülnek fel.

2.2. D—A átalakítók

Funkcionális jellemzés

A D—A átalakító digitális jel (mennyiségi) információ tartalmának analóg jelre való leképzésére szolgáló illesztőelem. A digitális jel n elemű bináris vektor, melynek konkrét számértékét az egyes bináris komponensek nagyságán (0 vagy 1) túl egyrészt a választott kód (bináris, binárisan kódolt decimális stb.) és a számformátum (fixpontos, lebegőpontos) definiálja. A D—A átalakítók túlnyomó többségének számformátuma fixpontos törtszám, melyben a törtvessző a legnagyobb helyértékű bit-től balra áll. Bármilyen kódról legyen is szó, tehát az X_D digitális szám értéke a $0 \leq |X_D| < 1$ tartományba esik. Bináris kód esetén pl.:

$$X_D = x_1 2^{-1} + x_2 2^{-2} + \dots + x_n 2^{-n}.$$

Az analóg jel feszültség vagy áram alakjában, a szám, feszültség- vagy áramerősség szintként jelenik meg. Ennek megfelelően kétféle: *feszültségkimenetű*, ill. *áramkimenetű* átalakító különböztethető meg.

A bináris vektor X_D értéke és az Y_A analóg jelszint közötti összefüggést általánosságban az

$$Y_A = f(X_D)$$

függvény írja le. Az X_D digitális mennyiség természetéből következően csupán meghatározott diszkrét értékeket vehet fel, következésképpen az Y_A sem folytonos.

Unipolárisnak azt az átalakítót nevezik, melynek analóg értéktartománya a nullponthoz képest vagy pozitív, vagy negatív irányban terül el. A nullpont mindkét oldalára kiterjedő értéktartománnyal a *bipoláris* átalakító rendelkezik (Y_A pozitív és negatív egyaránt lehet). Az $Y_A = f(X_D)$ összefüggés nem invertáló, az $Y_A = -f(X_D)$ invertáló átalakítót határoz meg.

A gyakorlatban alkalmazott D—A átalakítók csaknem kivétel nélkül lineáris karakterisztikával rendelkeznek. Ezért D—A átalakítón a továbbiakban — ha külön nem jelezzük — lineáris jelleggörbéjű átalakítót értünk.

A neminvertáló lineáris D—A átalakító be- és kimenő értékei között az $Y_A = S X_D$

összefüggés áll fenn; S az ún. skálatényező, az átalakító tényleges kimenő érték tartományát definiálja. Az S más néven „végkitérésnek” is nevezhető. Ez utóbbi értelemben az X_D egy konkrét értéke azt határozza meg, hogy a hozzá tartozó Y_A hányadrésze a végkitérésnek. Tekintettel arra, hogy X_D dimenzió nélküli szám, az S dimenziója megegyezik Y_A -ével; áramkimenetű átalakító esetében tehát S áramerősséget, feszültségkimenetű átalakító esetében pedig feszültséget jelent.

A D—A átalakítók mindegyikének alapvető eleme a referenciaforrás. A referenciaforrás U_r feszültsége és az S között arányossági kapcsolat áll fenn, azaz

$$\text{feszültségkimenetű konverternél: } S = k \cdot U_r,$$

$$\text{áramkimenetű konverternél: } S = \frac{1}{R} U_r.$$

A k egyszerű viszonzszámot, R pedig ellenállást jelöl. Az S előjele megegyezik U_r előjével.

A D—A átalakítók tulajdonságait célszerű a normalizált karakterisztika alapján vizsgálni, mely

$$Y_s = \frac{Y_A}{S} = X_D$$

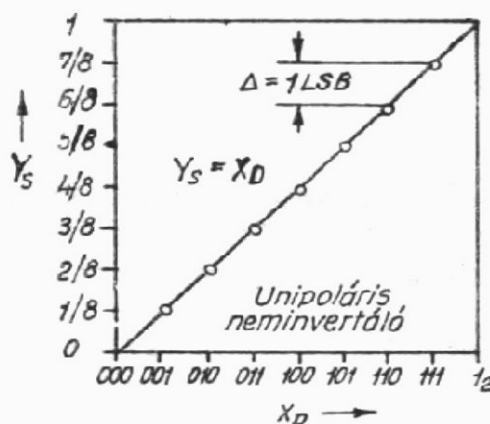
alakú. Az elmondottak illusztrálásaképpen a 2.11a, b ábra 3 bit bemenetű, unipoláris, neminvertáló átalakító összetartozó (X_D , Y_S) értékpárjait, valamint normalizált karakterisztikáját tünteti fel: a 2.11c ábrán pedig invertáló és neminvertáló bipoláris átalakítók jelleggörbéi láthatók. A normalizált jelleggörbéből az átalakító tényleges kimenő értéke a skálatényezővel való szorzás útján nyerhető.

Megállapítható, hogy az X_D értékek során végighaladva, X_D két szomszédos értéke közötti különbség a legkevesbé szignifikáns bit helyértékével 2^{-n} -nel egyezik meg, melyet általában *LSB*-vel jelölnék. Az analóg érték relatív növekménye, a D—A átalakító *felbontóképességének* mértékszám, ugyancsak *1LSB*. Az *LSB* értéke annál kisebb, a felbontás annál jobb, minél nagyobb az n .

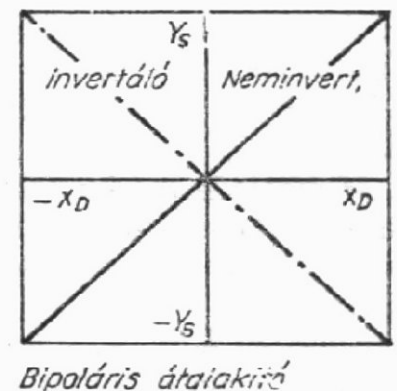
A D—A átalakító tényleges kimenő értéke a szomszédos X_D értékek közötti felező pontokban mutat legnagyobb eltérést a névlegestől; a relatív hiba értéke ott: *LSB/2*. Ezt az átalakító *felbontási hibájának* nevezik.

X_D	Y_s
000	0
001	1/8
010	2/8
011	3/8
100	4/8
101	5/8
110	6/8
111	7/8

a)



b)



c)

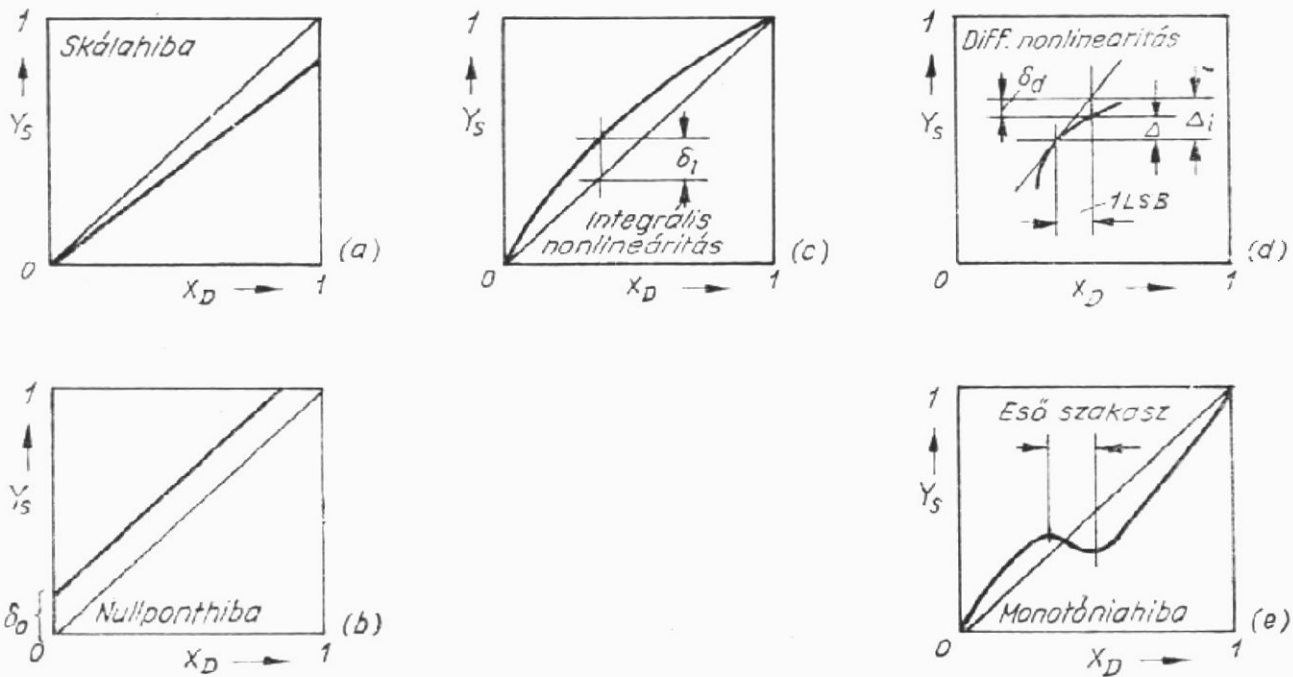
2.11. ábra.

D A átalakító normalizált jelleggörbéi:

(a) a normalizált jelleggörbe értéktáblázata; (b) és (c) ideális jelleggörbék

A felbontóképesség javulásával (n növekedésével) a karakterisztika érvényes pontjai sűrűsödnek, csökken a felbontási hiba, és az átalakító lehetséges maximális kimenő értéke egyre inkább megközelíti a végkitérést, bár azt elvileg soha nem érheti el.

A D—A átalakító jelleggörbéje a valóságban többé-kevésbé eltér az ideális, a (0,0) és az (1,1) pontokon áthaladó egyenestől. A legismertebb karakterisztikahibák: a skálahiba, a nullponthiba, a nemlinearitás és a monotonia hiba (2.12. ábra).



2.12. ábra.

D—A átalakító különböző jelleggörbe-hibái

A *skálahiba*: az S értékhibája, mely helytelen beállítás (kalibráció) vagy az átalakító üzemi paramétereinek a környezethatásokra reagáló megváltozása (pl. a referenciafeszültség hőmérséklet okozta eltolódása), esetleg alkatrészöregedés következményeként adódhat.

A *nullponthiba* (eltolódási hiba) a jelleggörbe párhuzamos eltolódásában jelentkezik. Helytelen munkapont-beállítás vagy környezethatásokból adódó munkapont-eltolódás következménye. Elsősorban bipoláris átalakítóknál fordul elő.

A *nemlinearitás* a tényleges karakterisztikának az egyenestől való eltéréseben jelentkezik. Két mértékszámát különböztetik meg, melyek közül az integrális nemlinearitás a karakterisztikának az ideálistól való max. eltérését a végkitéréshez viszonyítva adja meg (a 2.12c ábrán δ_i). A differenciális nemlinearitás pedig a karakterisztika bármely (értelmezett) pontjában, az előző analóg értékhez viszonyított Δ növekménynek a Δ_i ideálistól ($1LSB$) való δ_d eltérését adja meg, LSB egységben kifejezve (2.12d ábra).

A *monotonia hibája* annak az átalakítónak van, melynek karakterisztikája eső szakaszt is tartalmaz; az analóg érték csökken, miközben a digitális érték nő. Ilyen helyzet ott áll elő, ahol a negatív értelmű differenciális nemlinearitás meghaladja az $1LSB$ -t.

A *abszolút hiba*: az egyes analóg szintek tényleges értékének a névleges értéktől való (fizikai egységben számított) eltérése.

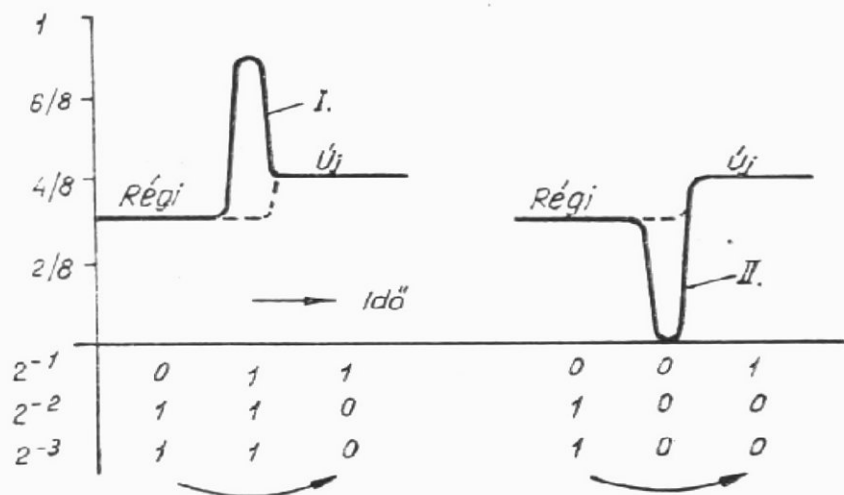
A *relatív hiba*: az abszolút hiba értéke a végkitéréshez viszonyítva. Esetenként (pl. külső referenciaforrás használata kapcsán) egyedül a relatív hibának van jelentősége; az abszolút hiba értékének ismerete érdektelen.

A jellemzőknek dinamikus üzemi körülmények között mért értéke a statikus állapotban mért értékektől lényegesen eltérhet; a jellemzők teljes meghatározásához

hozzátartozik az a működési sebességhatár is, ameddig a szóban forgó adatok érvényesek.

A beállási idő fogalmát az analóg mérőhálózatok ismertetése során az 1.14. ábrával kapcsolatban már meghatároztuk. A D—A átalakító beállási idejét általában $LSB/2$ nagyságú hibahatár figyelembevételével rögzítik.

Átkapcsolási tüskének azokat a tranzienseket nevezik, melyek a digitális bemenő érték változtatása során az analóg kimeneten jelennek meg. A tüskék előidézésében többféle tényező is szerepet játszhat. Ezek közül elsősorban az átalakítón belül zajló kapcsolási folyamatoknak a kimenetre gyakorolt kapacitív áthatását kell megemlíteni. A tüskéképződés igen nagy mértékben függ a digitális vezérlőjel megváltozásának időbeli lefolyásmódjától és az alkalmazott kódtól is. A 2.13. ábra az utóbbira mu-



2.13. ábra.

Az értékváltozás helytelen időbeli menete által előidézett tüskék a D—A átalakító kimenetén

tat két példát. Mindkét esetben a bináris digitális érték 011-ről 100-ra változik. A túlvívés jellegű átkapcsolási tüske (I) akkor jön létre, ha a digitális bemeneten a legnagyobb helyértéken az 1 logikai szint olyan időpontban jelenik meg, melyben az alacsony helyértékek szintje még nem esett 0-ra. Alálívés jellegű tüske (II) viszont akkor jön létre, ha az új érték beírása előtt a régi érték törlődik. Az átkapcsolási tüskék szempontjából valamely kódtípus annál előnyösebb, minél kevesebb helyértéken következik be változás X_D két szomszédos értéke között. A Gray-kód esetében pl. — amint ez ismeretes — a számtartomány bármely két szomszédos értéke közötti átmenetnél a változás csupán 1 bitnyi. A digitális bemeneten azonban nem csak két szomszédos érték közötti átmenet fordulhat elő, s bármely kód esetében is előfordulhat sok helyértéket érintő értékcseré. A tüskéképződés szempontjából az a kódtípus a legelőnyösebb, melynek lehetséges „átkapcsolási tranziensei” minél kevésbé térnek el az új értéktől. Az átkapcsolási tüskék egyedül a kód oldaláról teljesen nem eliminálhatók, ezért a jelváltás helyes időzítése optimális kódtípus alkalmazásakor sem nélkülözhető. Esetenként — az átkapcsolási tüskék eliminációja céljából — külön erre a célra szolgáló szűrőket is alkalmaznak, vagy a D—A átalakítót tároló mintavevővel is kiegészíthetik.

Kódtípusok

A D—A átalakító egyik meghatározó jellemzője, amely az átalakító belső felépítésére és alkalmazhatóságának lehetőségeire is kihat, digitális bemenetének kódolásmódja. A három legfontosabb kód: a bináris, a Gray- és a binárisan kódolt decimális (BCD).

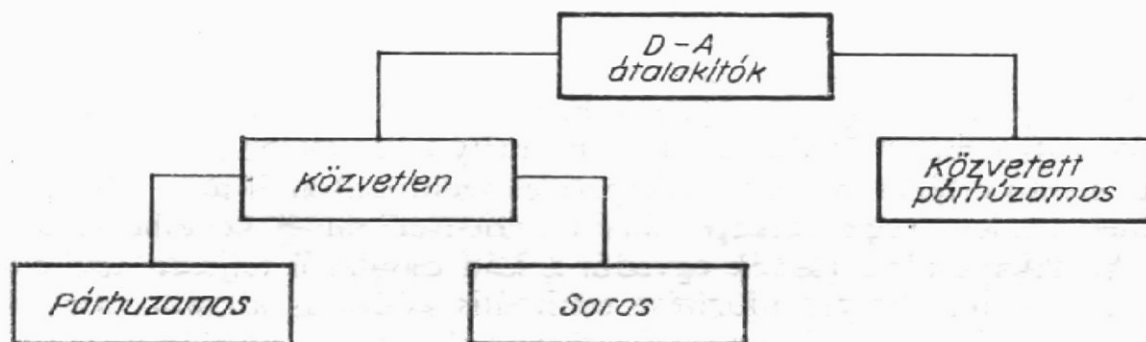
A D—A átalakítók technikájában a bináris kódoknak több változatát is alkalmazzák, melyek egymástól a negatív számérték ábrázolása, továbbá az előjel megadásának tekintetében különböznek. E bináris kódok közül legismertebbek az egyes és a kettes komplement kód, az előjel + szám és az ellensúlyozott bináris kód. Mind a négyféle bináris kódban a pozitív számok alakilag azonosak. Az első három kódtípusnál a pozitív előjelet bináris 0, a negatív előjelet bináris 1 jelöli; az ellensúlyozott kód előjelének ábrázolása éppen fordított. Az egyes és kettes komplement kódok negatív számértékei a megfelelő pozitív értékek egyes, ill. kettes komplementesei. Az előjel + szám kód negatív számértékei megegyeznek a pozitív számértékekkel. Az ellensúlyozott bináris kód a kettes komplement bináristól csak felcserélt előjegyével tér el. Az n jegyű ellensúlyozott kód alakilag egybevágó az $(n+1)$ jegyű bináris alapkóddal. A vele megvalósított n -bités S skálatényezőjű, bipoláris D—A átalakító voltaképpen — S eltolású $(n+1)$ bites $2S$ skálatényezőjű bináris átalakítóként fogható fel. A felsorolt négy kódunk invertált előjegyű és teljesen invertált változatait is alkalmazzák; az utóbbit elsősorban áramkörüi technológiai megfontolások alapján.

A számítógépes rendszerekben való alkalmazhatóság szempontjából figyelembe kell venni, hogy a legtöbb kisszámítógép kettes komplement aritmetikájú, amihez a kettes komplement kódolású átalakító jól illeszkedik. Mind a többi kód és a számítógép közé kódátalakító szükséges, ami több-kevesebb szerkezeti kiegészítést vagy program általi adminisztrációt kíván.

A D—A átalakítók osztályai

Közvetlen és közvetett átalakítók különböztethetők meg. A közvetlen átalakító a digitális értéket direkt módon képezi le áramerősség- vagy feszültség szintté. A közvetett átalakító ezzel szemben a digitális értéket először valamely közbenső analóg mennyiségre (pl. impulzusszélesség) viszi át, s csak ezt követően (A—A átalakítás útján) áramerősség/feszültség szintre.

A közvetlen átalakítók sorában párhuzamos és soros átalakítók különböztethetők meg, a közvetett átalakítók mind párhuzamos rendszerűek. A párhuzamos átalakító paralel kódot, a soros átalakító soros kódot fogad el, ill. dolgoz fel.



2.14. ábra.
D—A átalakítók osztályai

Párhuzamos közvetlen D—A átalakítók

A D—A átalakítók családjából legáltalánosabban a közvetlen párhuzamos rendszerű átalakítót alkalmazzák. Az átalakító lényegi része: az átalakító hálózat, mely U_r referenciaforrással, tárolóregiszterrel és erősítővel egészül ki. A gyakorlatban ez utóbbiaknak nem mindegyike képezi minden esetben az (áramkörileg integrált) D—A

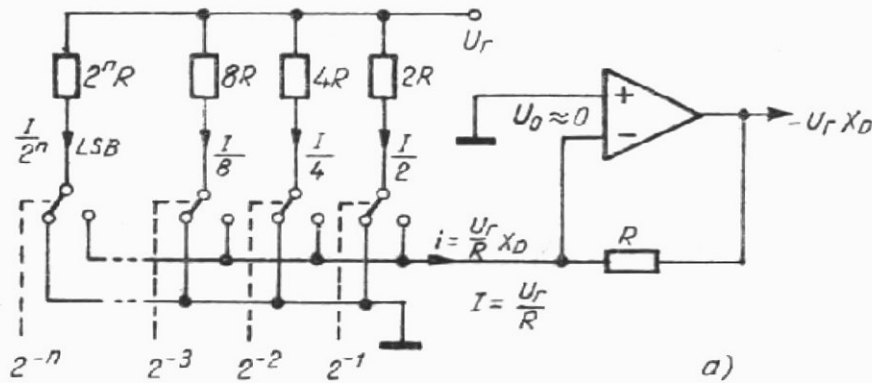
átalakító részét; a hiányzó egység — az alkalmazási szempontoknak megfelelően — kívülről csatlakoztatható. Az átalakító hálózat — felépítésének elvétől függően — áram- vagy feszültségkimenetű. A hálózatot követő erősítőt azonban általában oly módon választják meg, hogy végső soron feszültségkimenetet adjon; az áramkimenetű hálózatot tehát áramösszegező műveleti erősítővel, a feszültségkimenetű hálózatot pedig feszültségkövetővel csatlakoztatják a kimenethez.

A közvetlen párhuzamos D—A átalakítók három alaptípusa ismeretes:

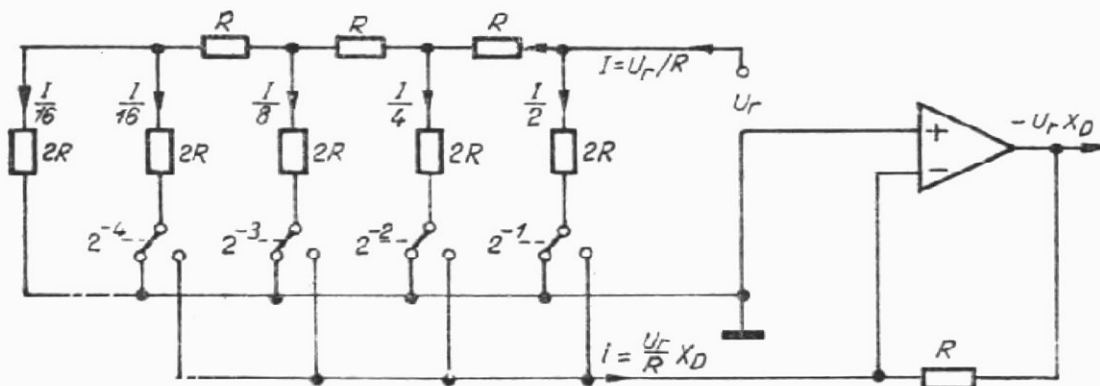
- a súlyozott ellenálláshálózatú,
- a létrahálózatú,
- a súlyozott áramgenerátor hálózatú.

Mindegyik típuson belül számos variáns található.

A *súlyozott ellenálláshálózatú átalakító* a bináris kód helyértékeinek számával azonos számú ellenállást tartalmaz, melyek értéksora 2 hatványai szerint növekszik. Ezen ellenállások bármelyikén, U_r feszültség hatására, helyértékszámával arányos áram jön létre (2.15a ábra). Mindazokon a helyértékeken, ahol a bináris kódban 1 áll,



a)



b)

2.15. ábra.

Párhuzamos közvetlen D—A átalakítók:

(a) súlyozott ellenálláshálózattal; (b) R—2R létrahálózattal

az átalakító kapcsolórendszere a helyérték-áramot az átalakító áramösszegező pontjára, ahol pedig 0 áll, a földre irányítja. Ily módon az áramösszegező pontra éppen $i = (U_r/R)X_D$ áram lép. A műveleti erősítő visszacsatoló ellenállását R -nek választva a kimeneten $-U_r X_D$ feszültség adódik. Az U_r negatív értéke nem invertáló, az U_r pozitív értéke invertáló unipoláris átalakítót definiál.

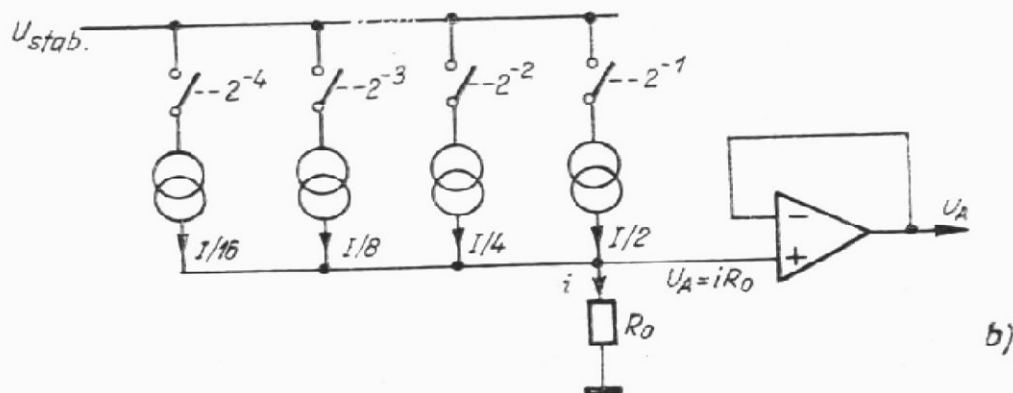
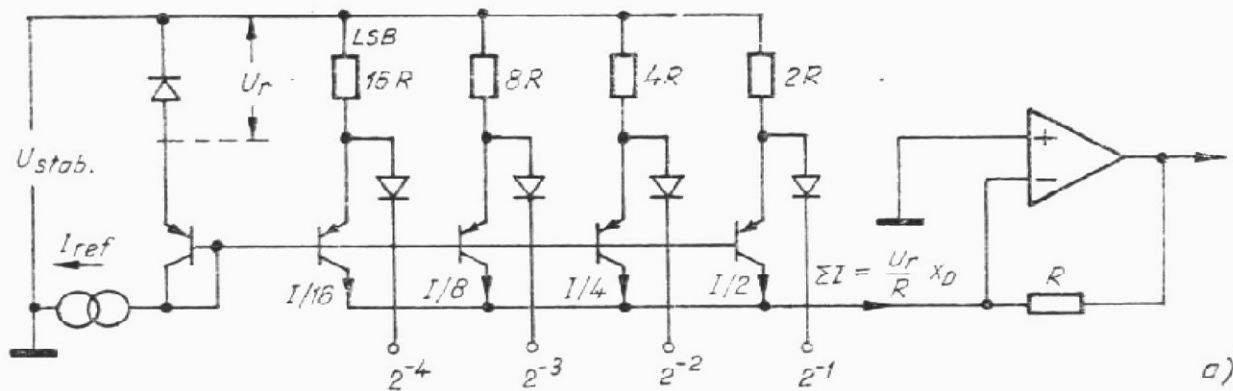
A súlyozott ellenállásokból épített hálózat igen nagy hátránya, hogy minden ellenállásának más-más értéke van, továbbá, hogy a legkisebb és legnagyobb ellenállás között az értékarány $1:2^{n-1}$. 12-bites átalakító esetén pl. ez annyit jelent, hogy míg

a legkisebb érték a $10^3 \Omega$, a legnagyobb a $10^6 \Omega$ értéktartományba esik. Egymástól távoli értéktartományba tartozó ellenállások hőmérsékleti együtthatója viszont nehezen tartható azonos szinten, figyelembe véve még, hogy a legkisebb ellenállással szembeni pontossági követelmények annál szigorúbbak, minél nagyobb az n . A súlyozott ellenálláshálózat említett problémáit a teljes hálózat ellenálláskvádokra (quadruplett) való osztásával oldják meg. Minden kvád azonos felépítésű, és azonos négy ellenállásból áll. A csökkenő helyérték irányában fekvő kvádot a szomszédos nagyobb helyértékűvel osztóellenállás köti össze. A kvádok nem csak bináris, hanem BCD kódot is elfogadnak, az összekötő osztó-ellenállás értékének megfelelő választása esetén.

A létrahálózat vagy más néven R - $2R$ hálózat, a súlyozott ellenállásokból álló hálózattal szemben csupán R és $2R$ értékű ellenállásokat tartalmaz. A 2.15b ábra létrahálózatában az U_r vonalának minden csomópontjából lefelé, ill., balra azonos $2R$ értékű ellenállás ágazik ki, tehát a csomópontok mindegyikén a jobb felől érkező áram két egyenlő részre oszlik el. Jobbról bal felé haladva a függőleges ágak árama így módon a 2 negatív hatványainak megfelelően csökken.

A súlyozott áramgenerátor-hálózat a súlyozott ellenálláshálózattól csak abban tér el, hogy az ellenállások helyén egy-egy áramgenerátor áll (2.16a ábra). A súlyozott áramgenerátor-hálózat egyetlen (R_0) ellenállás alkalmazásával feszültségkimenetű hálózattá alakítható át (2.16b ábra).

Az összes eddig felsorolt D—A átalakító unipoláris. A 2.17a ábra ellensúlyozott bináris kódolású bipoláris hálózat felépítmódját szemlélteti. A pozitív értéktartományban az 1 előjegynek megfelelő $I = I_e$ áram teljes egészében kiegyensúlyozza az áramösszegező pontra kapcsolódó áramgenerátor $I_e = U_r/R$ áramát. Kiegyensúlyozott állapotban az átalakító hálózat pontosan úgy viselkedik, mint a 2.15a unipolá-

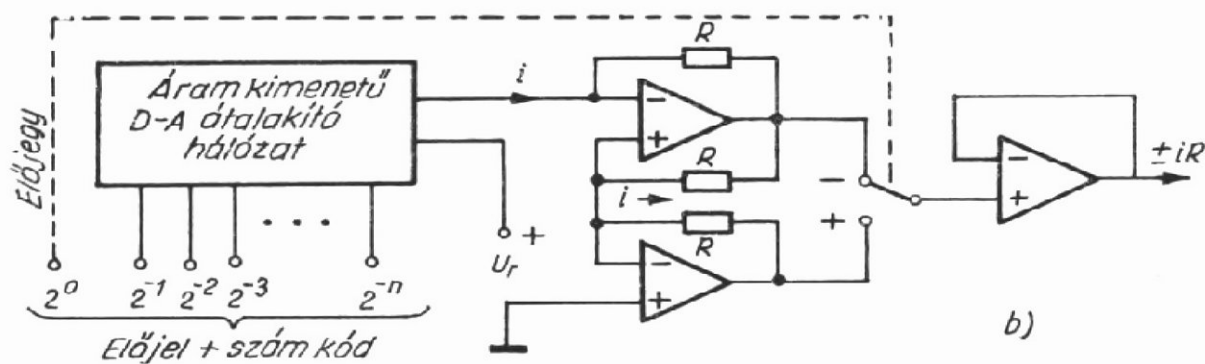
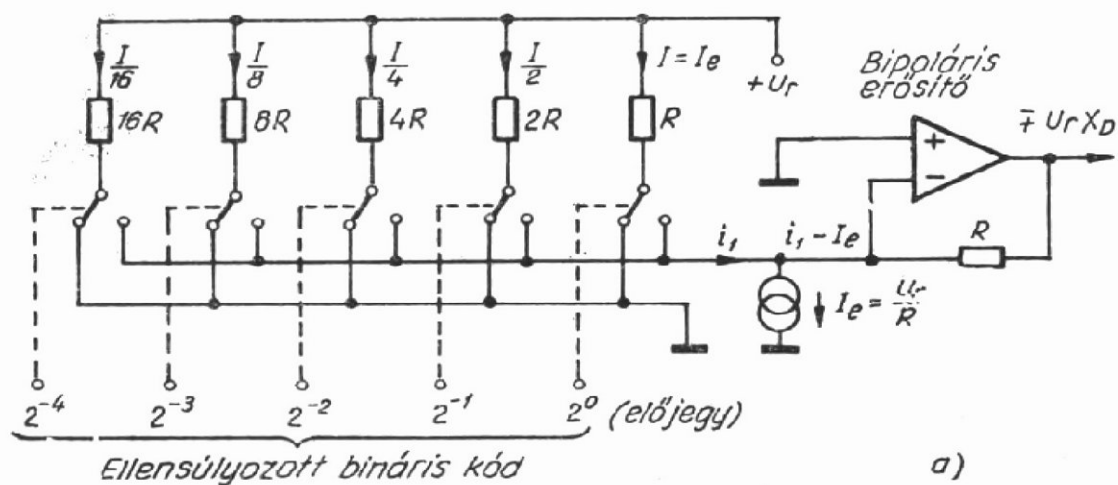


2.16. ábra.

Súlyozott áramgenerátor hálózatú, párhuzamos közvetlen D—A átalakító

(a) áramkövető kimenet; (b) feszültségkövető kimenet

ris hálózat. Az előjegy 0 értékénél (negatív értékek tartománya) az áramösszegező pont I_e -vel negatív irányba tolódik el. Így (X_D) bármely értékénél az összegezési ponton negatív áram adódik.



2.17. ábra.

Bipoláris D—A átalakítók

(a) ellensúlyozott bináris kódolású; (b) előjel + szám kódolású

Előjel + szám kódolású bipoláris hálózatot mutat be a 2.17b ábra. Az előjegy ez esetben a polaritásváltó erősítő kimenő kapcsolóját alternálja.

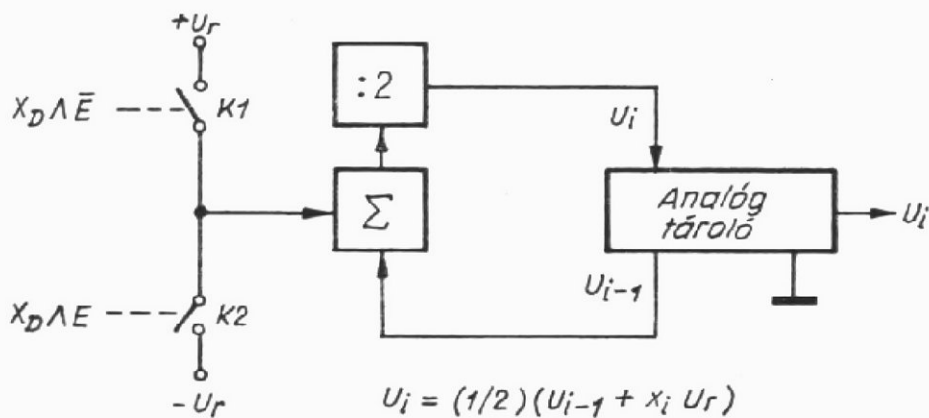
Soros közvetlen D—A átalakítók

Az X_D paralel kódtól való megkülönböztetéseként jelöljük a soros kódot X_{DS} -sel. Jelentése:

$$X_{DS} = x_k 2^{-1} + x_{k-1} 2^{-2} + \dots + x_2 2^{-(n-1)} + x_1 2^{-n},$$

bipoláris alakja pedig $X_{DS}^* = x_E \cdot 2^0 + X_{DS}$.

Az 1, 2 ..., k számsor az időbeli sorrendet adja meg, amelyben a kód egyes helyi értékei az átalakító bemenetén megjelennek. Először tehát *LSB*, legutoljára pedig a 2^{-1} súlyú x_k bit érkezik be, ill. bipoláris kód esetében az x_E előjelbit. A bipoláris kód előjelét külön megjelölő impulzus tünteti ki, melynek értéke x_E mellett E , egyébként \bar{E} . Ezek előrebocsátása után a kettes komplement kódolású bipoláris soros D—A átalakítók működési elve a 2.18. ábra alapján a következőkben világítható meg:



2.18. ábra.
Soros közvetlen D—A átalakítók általános működési sémája

A K1, ill. K2 kapcsoló minden olyan óraciklusban zár, amikor az $x_i \wedge \bar{E} = 1$, ill. $x_i \wedge E = 1$ logikai feltétel teljesül. Ilyenkor az áramkör bemenete $+U_r$, ill. negatív szám (1 előjegy) esetében $-U_r$, egyébként 0 feszültséget kap. Az analóg tároló kimenő kapcsain az i -edik ciklus kezdetén az $(i-1)$ -edik ciklusból származó U_{i-1} feszültség uralkodik, melyet Σ áramkör hozzáad az i -edik bemenő értékhez ($\pm U_r$ vagy 0). Az összegezés eredményét az osztókör, 2-vel leosztva, az analóg tárolóba helyezi, ahol tehát

$$U_i = (U_{i-1} \pm x_i U_r) 2^{-1}$$

új érték képződik. A zárójelen belül a $+$ előjel helyébe $-$ csak akkor lép, ha $i = E$ és $x_E = 1$.

Közvetett D—A átalakítók

A közvetett D—A átalakítás két eljárása ismert általánosan. Az egyik szerint a digitális érték — első lépésben — meghatározott periódusidejű impulzussor pontosan definiált amplitúdójú impulzusának szélességére képződik le, majd második lépésben a kimenő feszültség ez impulzussorozat egyenáramú középértékeként adódik. A másik eljárás: a digitális értéket definiált amplitúdójú és szélességű impulzusok időegységre eső számára képezi le; az analóg egyenfeszültségű szint — ez esetben is — középérték-képzéssel áll elő.

Szorzó D—A átalakítók és nem lineáris átalakítók

A D—A átalakító $Y_A = SX_D = kU_r X_D$ összefüggése alapján a kimenet az $U_r X_D$ szorzattal arányos; a D—A átalakító szorzóként használható. U_r -ként pl. rögzített amplitúdójú váltakozó feszültséget alkalmazva a D—A átalakító az X_D értékével programozott amplitúdójú szinuszos jelgenerátorra alakul át. Az átalakítót egyéb jelformával rendelkező generátorok építőelemeként is alkalmazzák. A bipoláris D—A átalakító négynegyedes, azaz minden előjelkombinációra érvényes szorzóként is felfogható.

A D—A átalakítóval a lineáristól eltérő jelleggörbe is megvalósítható. U_r -ként programozható forrást választva, mindkét szorzóbemenetet program által vezérelve, máris ilyen átalakító áll előttünk.

2.3. A—D átalakítók

Funkcionális jellemzés

Az A—D átalakító, fordítva mint a D—A átalakító, az analóg jel (mennyiségi) információtartalmának digitális alakban való leképzésére szolgál. Analóg jeltartományának határpontját — a végkiterést — S -sel jelölve, a lineáris átalakító kimenetén megjelenő Y_D fixpontos törtszám az analóg értéket a végkiterés hányadában adja meg, azaz:

$$Y_D = \frac{X_A}{S}.$$

Ebben az értelemben S -et skálatényezőnek is nevezik. Az A—D átalakító által mért analóg mennyiség értéke fizikai egységben nyerhető, ha Y_D -t S -sel szorozzuk. Az X_A : feszültség- vagy áramszintet ad meg, s ennek megfelelően S is áram-, ill. feszültség-dimenziójú. Az S mindig kapcsolatban áll valamely U_r referenciafeszültséggel úgy, hogy végeredményben:

$$Y_D = \frac{X_A}{kU_r},$$

ahol k dimenzió nélküli, ill. vezetőképesség dimenziójú arányossági tényező. Az A—D átalakító tulajdonságainak áttekintésekor célszerű az analóg mennyiségnek a végkiterésre normalizált $X_s = X_A/S$ értékét figyelembe venni, miáltal a fenti egyenlet az $Y_D = X_s$ alakot nyeri.

Az Y_D számhoz tartozó bináris vektor a választott kód függvénye, mely utóbbi a D—A átalakítók kapcsán már ismertetett kódok egyike lehet. Minthogy a bináris komponensek n száma — tehát a szóhossz — minden konkrét esetben adott, a kód meghatározza az Y_D által felvehető értékek készletét. Az ezekhez az értékekhez tartozó szinteket az A—D átalakító kvantálási szintjeinek, a szomszédos szinteket elválasztó különbséget pedig kvantálási lépcsőnek nevezik. Nyilvánvaló, hogy Y_D csak a kvantálási szinteken adja meg X_s pontos értékét, ha pedig X_s értéke két kvantálási szint közé esik $Y_D \approx X_s$ érvényes; az A—D átalakító a kvantálási szintekre végez le-, ill. felkerekítést. A kerekítés által adódó hiba annál nagyobb, minél messzebb esik a lépcsőn belül az analóg érték attól a kvantálási szinttől, amelyikre a kerekítés történik. Egyirányú kerekítés esetén a maximális hiba egy kvantálási lépcső, amennyiben azonban az átalakító kétirányú kerekítést végez, a hiba a kvantálási lépcső fele. Minthogy a kvantálási lépcső normalizált értéke 1 *LSB*, tehát a kvantálási hiba maximális relatív értéke kétirányú lekerekítés esetében: *LSB*/2. Kétirányú lekerekítés lehetőségének figyelembevételével az A—D átalakító ideális jelleggörbáját az $n=3$ helyértékű bináris kód esetére a 2.19. ábra mutatja be.

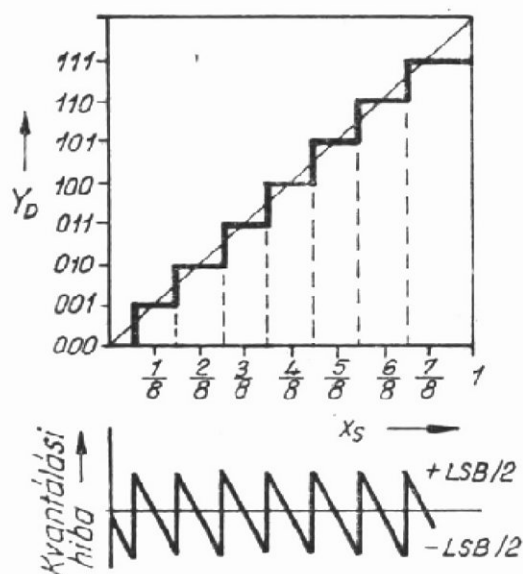
Az *unipoláris* A—D átalakító csak pozitív vagy csak negatív jelet fogad el. Bipoláris jelek feldolgozására alkalmas A—D átalakítók is ismeretesek.

Az A—D átalakító a D—A átalakítóhoz hasonló jellemzőkkel írható le.

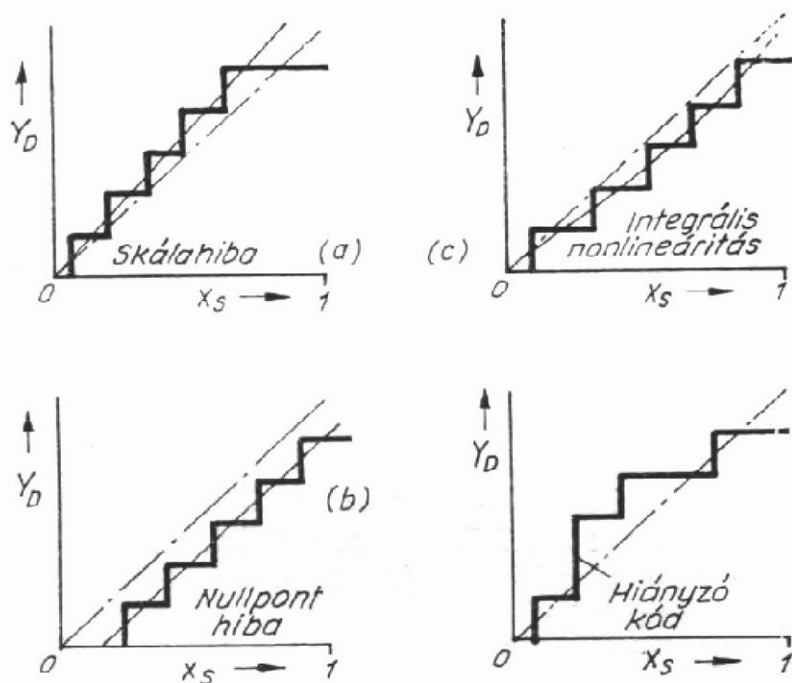
Felbontóképessége a kvantálási lépcső relatív értékével, *LSB*-vel jellemezhető. Az *LSB* bináris kód esetében, 2^{-n} és 10^{-d} a d -számjegyű BCD kód esetében. Az átalakító valóságos felbontóképessége különböző karakterisztikahibák, zajérzékenység stb. miatt esetenként lényegesen kisebb lehet, mint az n által megengedett elméleti érték.

A különböző jelleggörbe-hibákat a 2.20. ábra szemlélteti. A *skálahiba* az analóg tartományt nyomja össze vagy nyújtja meg arányosan. A *nullponthiba* a jelleggörbe paralel eltolódását idézi elő (az ábrán pl. pozitív irányban). Az *integrális nemlinearitást* a kvantálási lépcsők eredeti helyzetükhöz viszonyított elmozdulási maximumának a végkiterésre vonatkoztatott értékeként definiálják. A *differenciális nemlinearitás*:

a kvantálási lépcső tényleges és ideális értéke közötti differencia. A normalizált jelleggörbén a felsorolt hibák értéke *LSB* egységben adódik. Meg kell még említeni a monotonia hibának megfelelő *hiányzó kód* esetét, mely akkor lép fel, ha az A—D átalakító egy számértéket kihagy.



2.19. ábra.
A—D átalakító ideális normalizált jelleggörbéje



2.20. ábra.
A—D átalakító jelleggörbe-hibái

Az A—D átalakító abszolút hibáján az átalakító által mért értéknek ($Y_D \cdot S$) az analóg mennyiség tényleges értékétől való eltérését nevezzük. Az abszolút hiba az összes felsorolt hibákat magában foglalja. A relatív hiba az abszolút hibának a végkitérésre vonatkoztatott értéke.

Az A—D átalakító lényeges jellemzője az *átalakítási idő* és az *átalakítási ütem*. Az előzőn azt az időtartamot értjük, mely az analóg mennyiségnek az átalakító bemenetére vitelétől a digitális ekvivalens megjelenítésig telik el. Az átalakítási

ütem pedig azt a legrövidebb periódusidőt definiálja, mellyel az átalakítási ciklusok egymást követhetik. E periódusidő, az A—D átalakító feléledési követelményeinek figyelembevételével, mindig hosszabb az átalakítási időnél.

Kvantálás és kódolás

Az A—D átalakító közelebbi feladata abban áll, hogy megállapítsa: a bemenő érték melyik két kvantálási szint közé esik, képezze az analóg mennyiség le- vagy felkerekített értékét, s ezt az értéket előírtan kódolt bináris vektorként szolgáltatassa. Feladata tehát kettős: kvantálás és kódolás. Bizonyos átalakítók e két feladatot egymástól elkülönülő mozzanatként, más átalakítók egyetlen műveletként hajtják végre. A kvantálás a kódolással együtt lényegében nem más, mint az (ismeretlen) analóg mennyiségnek a megadott felbontóképesség által definiált finomságú és az előírt kód szerint leszarmazott skálára való rávetítése.

A kvantálás során minden átalakító az ismeretlen X_A mennyiséget bizonyos komparálási szintekkel hasonlítja össze abból a célból, hogy az X_A -t két szomszédos kvantálási szint közé skatulyázza. A beskatulyázási eljárás mikéntjétől függően három alapvető kvantálási módszert különböztetnek meg.

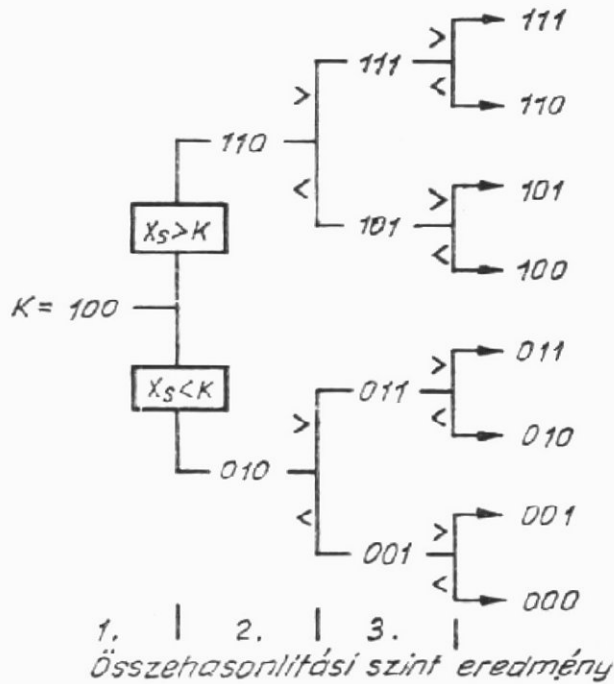
Paralel kvantálás esetében X_A -t, az ismeretlen analóg mennyiséget egyidejűleg az összes kvantálási szinttel összehasonlítják. Az összehasonlítás eredménye közvetlenül, N kvantálási szint esetében N -ből az egy kódban adódik. Az összehasonlító egységhez külön kódoló áramkör társul, mely az eredményt az adatfeldolgozás céljára alkalmas kódra képezi le.

A *számláló kvantálás* esetében az A—D átalakító az X_A -t a kvantálási lépcsővel inkrementált lépcsős függvényvel közelíti meg, a nulla szintről kiindulva. Az inkrementumok számát (mely az inkrementummal szorozva a momentán összehasonlítási szintet adja meg) számláló regisztrálja. Amikor az összehasonlítási szint az X_A értékét $\cong |LSB/2|$ pontossággal megközelíti, az inkrementálás folyamata leáll, s a számláló a rendszer által definiált kódban (pl. bináris) szolgáltatja az analóg szint digitális (közelítő) értékét.

Az *iteratív kvantálási eljárás* a kódolással áll szoros kapcsolatban, amennyiben a komparálási értékeket a választott kódtól függő súlyozott inkrementumokból állítja össze. Bináris kód esetén pl. az inkrementumoknak az S -re vonatkoztatott relatív értékei a bináris helyértékekkel egyeznek meg. Az egyes komparálási szintek a súlyozott inkrementumokból tevődnek össze, mégpedig oly módon, hogy minden újabb összehasonlítási szint az első kivételével függ a megelőző összehasonlítás eredményétől. A kvantálás algoritmusát 3-bites konverter esetére a 2.21. ábra mutatja be. Látható, hogy az első összehasonlítás a legnagyobb inkrementummal történik a komparálás szintje $K=2^{-1}=1/2$. Amennyiben $X_S > K$, úgy Y_D legnagyobb helyértékű bitje eggyel egyenlő, a következő komparálási szint pedig $(2^{-1}+2^{-2})=3/4$. Ellenkező esetben Y_D legnagyobb helyértékű bitje nullával egyenlő, a soron következő komparálási szint pedig $2^{-2}=1/4$. Az eljárás a továbbiakban ugyanezen módon folytatódik. A harmadik összehasonlítás után máris az eredmény adódik.

Míg a paralel kvantálás egyetlen lépésben történik, de — a nulla nívót is beleszámítva — a kvantálási szintekével megegyező számú összehasonlítási szint egyidejű jelenlétét igényli, addig a számláló kvantálás, szélső esetben, a kvantálási szintekével megegyező számú összehasonlítási ciklust igényel úgy, hogy minden összehasonlítási ciklus komparálási szintje a megelőzőétől egy kvantálási lépcsővel különbözik. A számláló kvantálás összehasonlítási ciklusainak száma — általánosságban — az analóg jelszint függvénye. Iteratív bináris kvantálás esetében az összehasonlítási ciklusok száma n -nel, a bináris helyértékek számával egyezik meg. A lehetséges

különböző összehasonlítási szintek közül döntési eljárással az A—D átalakító mindig csak n -et alkalmaz. Az átalakítási idő szempontjából legelőnyösebb a paralel kvantálás, majd az iteratív eljárás következik, legvégén a számláló eljárás áll.



2.21. ábra.
Az iteratív bináris kvantálás algoritmusá

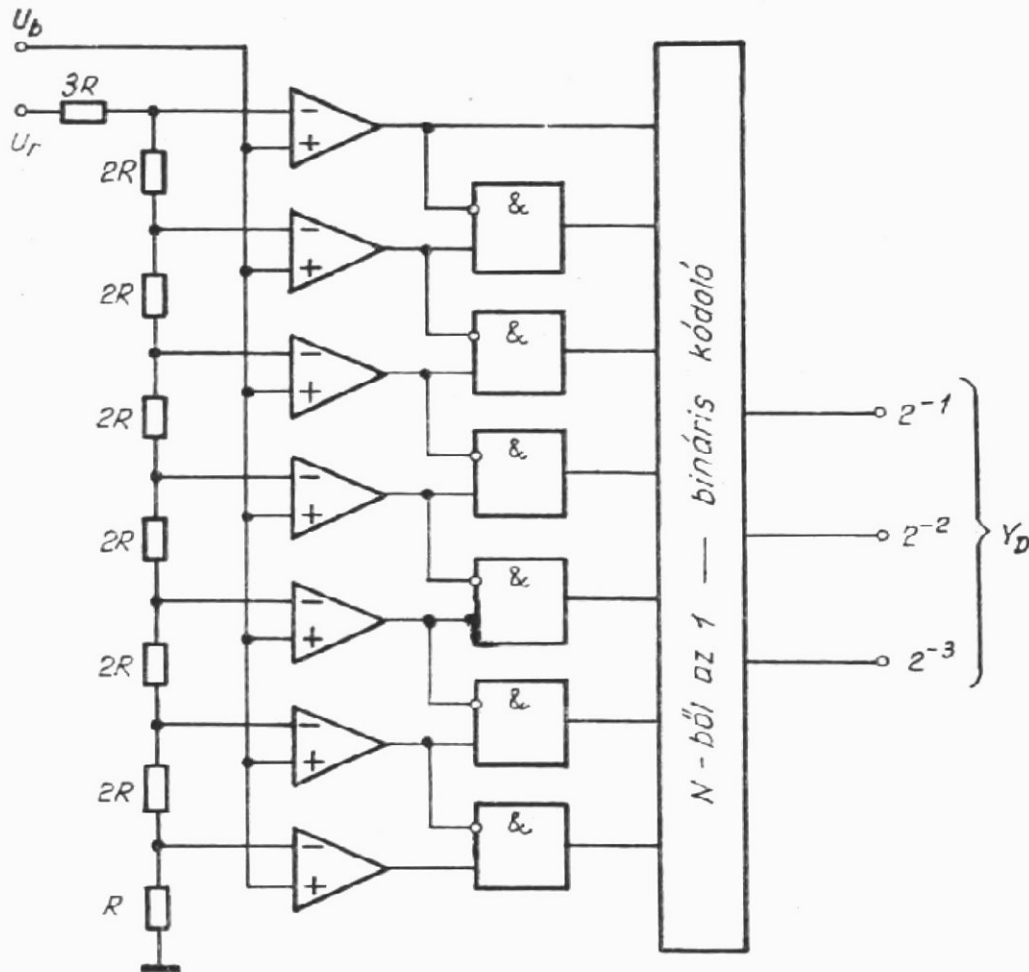
Az A—D átalakítók osztályai

A kvantálás az A—D átalakítók egy részénél közvetlen módon történik, ezek a *közvetlen átalakítók*, más rendszerekben pedig közvetett módon úgy, hogy az analóg mennyiség mindenekelőtt más analóg mennyiségre (pl. feszültség időtartamra) transzformálódik, s az átalakító e közvetítő mennyiséget kvantálja, ill. képezi le digitális mennyiségre. Ez utóbbiak a *közvetett A—D átalakítók*. A *párhuzamos átalakítók* a digitális mennyiséget paralel kód, a *soros átalakítók* soros kód alakjában szolgáltatják.

Az átalakítók megkülönböztethetők kvantálási eljárásaik alapján is. A kvantálás módja és az átalakítás ideje között szoros összefüggés áll fenn. A paralel kvantálással dolgozó átalakító az átalakítást egyetlen ciklusban hajtja végre, a másik két kvantálási eljárás több, egymást követő átalakítási ciklust igényel. A paralel kvantálás a digitális vektor minden komponensét egyidejűleg szolgáltatja, míg a másik két eljárás esetében a digitális vektor komponensei az időben egymás után jelennek meg. Attól függően, hogy az egyes komponensek megjelenésének időpontjai kapcsolatban állnak-e valamely óragerátor impulzusaival vagy sem, beszélünk *szinkron*, ill. *aszinkron* rendszerű átalakítókról. Az átalakítók értékkövetési képességük tekintetében is eltérnek egymástól, amennyiben egyes rendszerek képesek a bemenő kapcsón uralkodó analóg érték időbeli nyomon követésére, míg más rendszerek helyes működésének alapfeltétele, hogy az átalakítás időtartama alatt a bemenő érték csak elhanyagolható mértékben változzék. Egyes átalakítótípusokat az analóg értéknek a bemenetre való visszavezetése jellemez, míg más típusaiknál az átalakítás visszavezetés nélkül megy végbe.

Paralel kvantálású párhuzamos A—D átalakítók

Az $1/N$ felbontóképességű paralel kvantálású párhuzamos átalakító $N - 1$ komparátort, bináris kódolás, $N = 2^n$ esetén tehát $2^n - 1$ komparátort tartalmaz. Szemléltetésül a 2.22. ábra 2^{-3} felbontóképességű átalakítót mutat be. A hét feszültségkomparátor



2.22. ábra.
Paralel kvantálású párhuzamos A—D átalakító

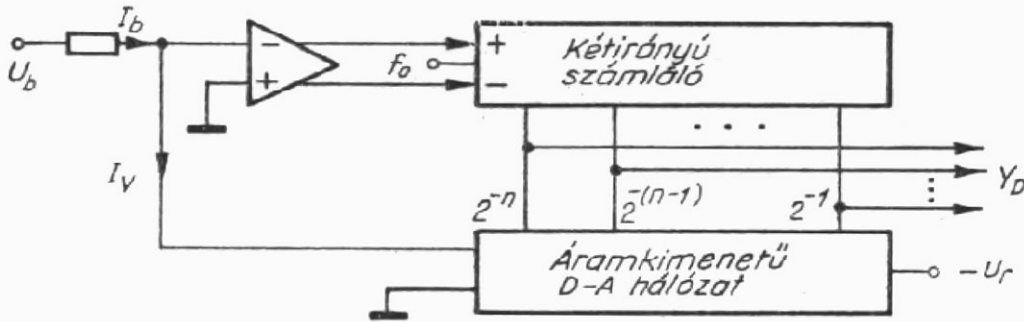
szintjei a kvantálási lépcsők szimmetriavonalára vannak beállítva, minek folytán a kvantálási lépcső alsó felén az átalakító lefelé, felső felén felfelé kerekít. Az analóg jel szintje alatti összes komparátor kimenete logikai 1 állapotban, a felette fekvőké logikai 0 állapotban van. Ily módon csak a közvetlenül a jelszint alatt fekvő komparátor logikai kapuja ad 1 kimenetet. Az átkódoló hálózat az N -ből egy értéket bináris kóddá alakítja át. A bináris kód helyett természetesen másféle kód (pl. Gray) is választható. Ez esetben az átkódoló hálózat módosul. Az átalakító, elvileg teljesen azonos módon, feszültségkomparátorok helyett áramkomparátorokkal is felépíthető.

A paralel kvantálású párhuzamos átalakító előnye nagy működési sebessége (az átalakítási ciklus ideje egyedül a kvantáló és kódoló áramkörök beállási idejének függvénye), hátrányos vonása viszont, hogy kvantálási lépcsőnként egy-egy komparátort igényel, s N növekedésével a dekódoló felépítése is egyre bonyolultabbá válik.

Az iteratív kvantálású párhuzamos átalakítók, melyeket itt nem részletezünk, a paralel kvantálású átalakítókhoz képest $\approx N/n$ arányban kevesebb áramköri elemet igényelnek, viszont lassúbb működésűek.

Szervo rendszerű átalakító

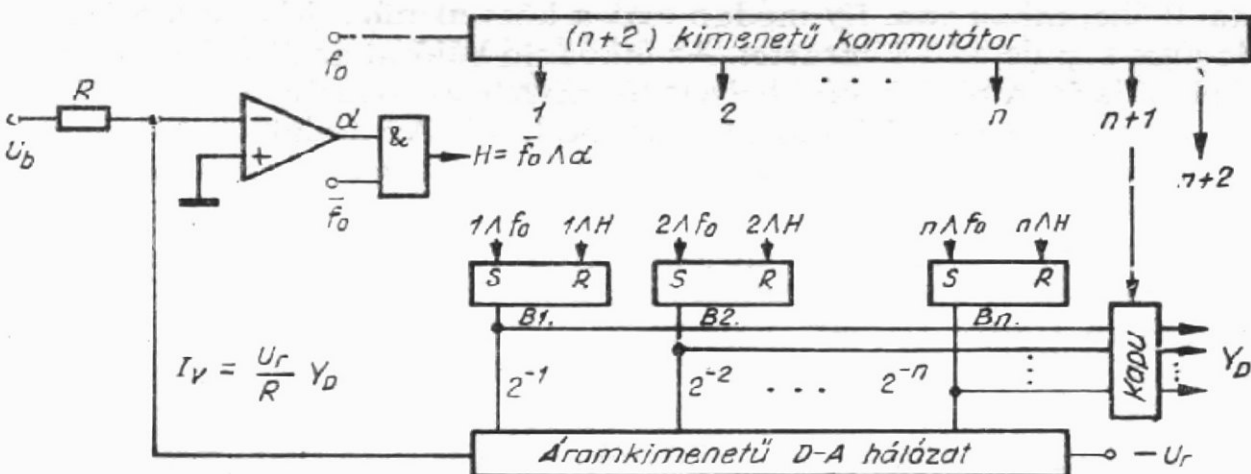
A szervo rendszerű A—D átalakító kettős kimenetű komparátorból, kétirányú számlálóból, valamint D—A hálózatból épül fel, és számláló kvantálással dolgozik (2. 23. ábra). Kezdő állapotban a komparátor visszavezetett árama $I_V = 0$, a számláló nyugalmi állapotban van és tartalma zérus. A bemenetre jelfeszültséget helyezve, a komparátor egyensúlya felborul, a számláló az előre (+) bemenetére érkező vezérlőjel hatására akkumulálja az f_0 generátor impulzusainak számát. Annak arányában, ahogy a számláló feltöltődik I_V növekszik, míg ki nem egyenlíti az I_b jeláramot. Ekkor a komparátor visszanyeri egyensúlyát, a számlálási folyamat leáll; a kimeneten az U_b -nek megfelelő Y_D érték rendelkezésre áll. Amennyiben U_b változva megnövekednek, a számlálási folyamat az előbbi módon újra indul, ha pedig U_b csökkenése miatt $I_b < I_V$ helyzet állna elő, a komparátor a számláló vissza (–) bemenetét vezérli, s az f_0 impulzusai a számláló tartalmából leszámolódnak. Ez mindaddig folytatódik, míg a komparátor egyensúlya ismét helyre nem áll. A szervo rendszerű átalakító átalakítási ideje — a számláló kvantálásra tekintettel — lényegesen hosszabb, mint az előbb ismertetett átalakítóké. Előnyös tulajdonsága viszont, hogy áramköreinek sebességi határain belül képes időben változó szintű jel követésére.



2.23. ábra.
Szervo rendszerű A—D átalakító

A—D átalakító szukcesszív approximációval

A szervo átalakítókhöz hasonlóan, ez az átalakító (2.24. ábra) is a kimenetről a bemenetre visszavitt jellel, de iteratív kvantálással dolgozik. Komparátorból, $(n+2)$ kimenetű kommutátorból (n az Y_D számjegyeinek száma), n bistabilból és a komparátorra kapcsolódó D—A hálózatból áll. Az átalakítási ciklus kezdetével



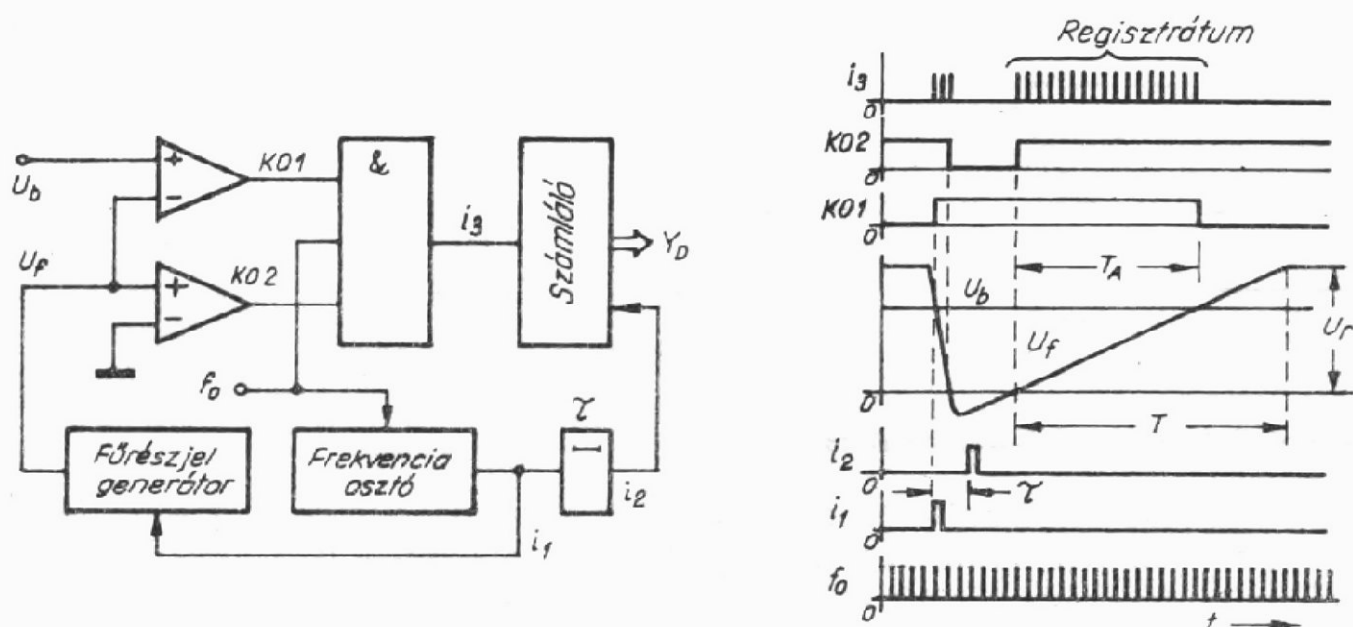
2.24. ábra.
A—D átalakító szukcesszív approximációval (iteratív kvantálással)

f_0 első impulzusa a kommutátor bemenetére lép; a kommutátor 1. kimenetén $1/f_0$ időtartamú impulzus áll elő. Az f_0 első félperiódusában a B1 bistabil logikai egy állapotba billen (s bemenet), és a D—A átalakító az U_b bemenő feszültséggel $U_r/2$ feszültséget állít szembe. Amennyiben $U_b > U_r/2$, a komparátor kimenetét követő ÉS kapu kimenete az f_0 második félperiódusában $H=0$, és B1 megőrzi állapotát. $U_b < U_r/2$ esetében azonban f_0 második félperiódusában $H=1$ adódik, miáltal B1 logikai null állapotba tér vissza. Az f_0 periódusának végére, így módon az Y_D legnagyobb helyértékű jegye már B1-ben található. Az f_0 következő ciklusában a kommutátor 2. kimenetén jelenik meg impulzus, és az előbb leírt összehasonlítási eljárás (B1 állapotának változatlanul hagyása mellett) B2 bevonásával zajlik le; eredményeképpen B2-ben a 2^{-2} helyértéksúlyú y_2 számjegy marad vissza és így tovább. A 2^{-n} helyértékű számjegy meghatározása után már Y_D minden számjegye rendelkezésre áll; a kommutátor $(n+1)$ -edik csatornájának jele az Y_D -t paralel kód alakjában bocsátja ki. Az $(n+2)$ impulzus az összes bistabilt nullázza. Ezután az átalakítási folyamat újra kezdhető. Szemben a szervo átalakító számlálási folyamatával, az átalakítási ciklusok száma itt állandó, éspedig $n+2$; a szukcesszíve approximáló átalakító azonban U_b -t nem követi; az átalakítás ideje során U_b értéke számottevően nem változhat.

Mind a szervo, mind a szukcesszíve approximáló átalakító szinkron működésű; az átalakítási ciklusok periodicitását f_0 határozza meg. Tekintettel arra, hogy a szukcesszíve approximáló átalakítónál egyrészt az y_1, y_2 stb. számjegyek f_0 -val szinkron időbeli egymásutánban adódnak, másrészt, mivel ezeket az értékeket regiszter gyűjti össze, ez az átalakító soros vagy párhuzamos átalakítóként egyaránt felfogható.

Közvetett A—D átalakító lineáris időkonverzióval

A közvetett átalakítás egy módszere az analóg mennyiségnek egy vele lineárisan arányos T_A időtartamra való leképzése, és az időtartam mérése valamely f_0 óragenerátor T_A -ra eső impulzusszámának regisztrálásával (számláló kvantálás). Az átalakító működésmódja a 2. 25. ábra alapján a következőkben világítható meg. Az átalakítási ciklust a frekvenciaosztó kimenő jele (i_1) indítja. Hatására a fűrészjel-generátor



2.25. ábra.
Közvetett A—D átalakító lineáris időkonverzióval

az U_r szintről a nullszint alá mozdul (közben KO1 és KO2 komparátor is átbillen), majd rátér lineárisan növekvő szakaszára. Mielőtt ismét áthaladna a nullszinten, a számláló i_2 engedélyező jelet kap. A nullszinten való áthaladás időpillanatában a komparátor ÉS kapuja nyílik, és a számláló megkezdí az f_0 órajelek számlálását. A számlálást KO1 zárja T_A idő múlva, a fűrészelnek az U_b szinten való áthaladásakor. Az ábra alapján látható, hogy

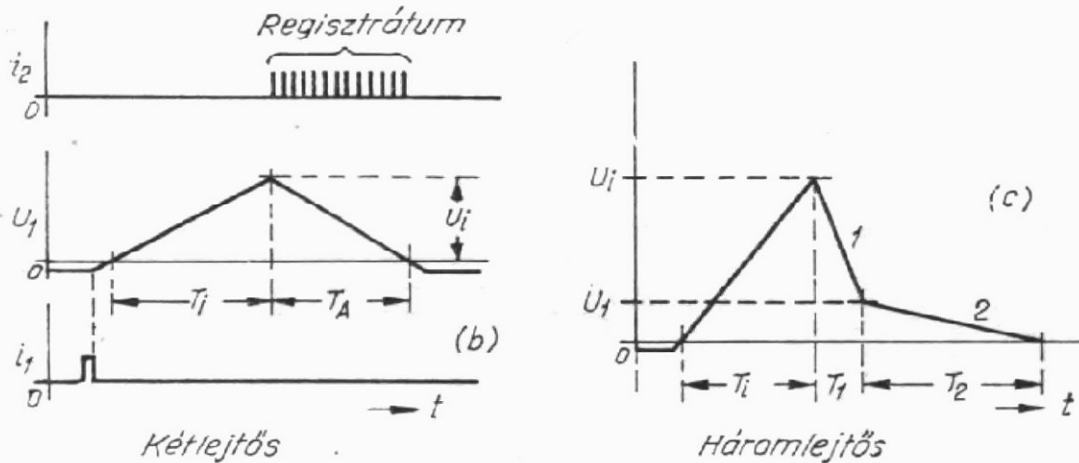
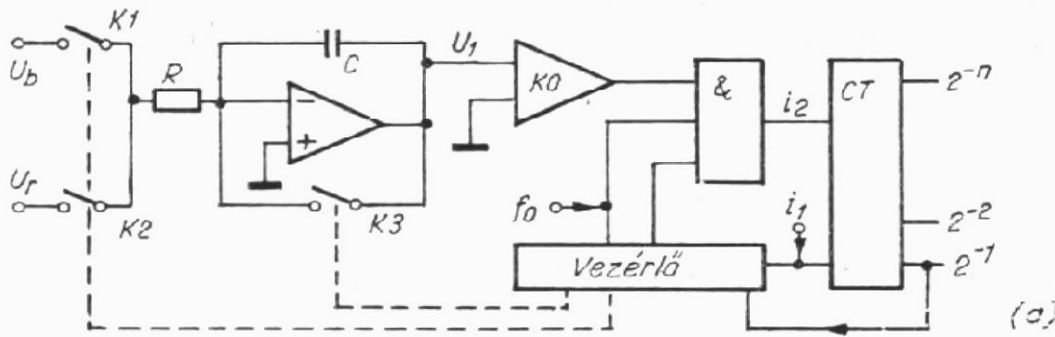
$$U_b = U_r \frac{T_A}{T} = U_r \frac{SZ_A/f_0}{SZ/f_0} = U_r \frac{SZ_A}{SZ},$$

ahol SZ_A a számláló regisztrátuma, SZ pedig a T időre eső óraimpulzusszám (konstans).

Az átalakító egyszerűbb változataiban fűrészel-generátorként Miller-integrátort, precízebb megoldásaiban D—A átalakítóra épülő s az óragenerátorral szinkron lépcsőgenerátort alkalmaznak.

Két- és háromlejtős A—D átalakító (2.26. ábra)

A kétlejtős integráló átalakító ugyancsak közvetett átalakító; az analóg értéket előbb időtartamra, majd az időtartammal arányos impulzusszámra transzformálja. Működésének lényege, hogy a bemenő értéket analóg tárolóba helyezi, majd a



2.26. ábra.
Két- és háromlejtős integráló A—D átalakító

tárolóként használt kondenzátort pontosan előírt árammal kisüti. A kisütés ideje arányos a tárolt analóg mennyiséggel, melynek nagyságát végül a kisütési idő alatt regisztrált óraimpulzusok száma reprezentálja. Figyelemre méltó, hogy az analóg bemenő érték tárolása nem mintavételszerűen történik, hanem pontosan rögzített T_i idő alatt végzett integrálással, minek következményeként a tárolóban nem pillanatérték, hanem a bemenő jel T_i időre vonatkozó középértéke rögzítődik.

Az átalakítási folyamat i_1 indítójellel kezdődik, melynek mellső frontja nullázza a CT számlálót, hátsó frontja nyitja a K3 és zárja a K1 kapcsolót (K2 nyitva marad). T_i idő eltelte után K1 kinyílik és K2 zárul. Az U_b -vel ellenkező előjelű U_r forrás és az R által definiált referenciaáram megkezd C kisütését. Ugyanakkor a vezérlő nyitja a számláló bemeneti ÉS kapuját, s a számláló hozzákezd f_0 impulzusainak regisztrálásához. A regisztrálás U_1 -nek a nullvonalon való áthaladásával zárul, amit K0 érzékel.

A középértékképzés a bemenet zavar szintjének csökkentése szempontjából jár előnnyel. Feltételezve, hogy az U_b bemenő jelre U_{sz} amplitúdójú szinuszos zavarjel szuperponálódik, a T_i időre vonatkozó középérték:

$$\bar{U}_b = \frac{1}{T_i} \int_0^{T_i} [U_b + U_{sz} \sin(\omega t)] dt,$$

azaz

$$\bar{U}_b = U_b + \frac{U_{sz}}{T_i} \left[-\frac{\cos \omega t}{\omega} \right]_0^{T_i}.$$

A váltakozófeszültség-elnyomási tényezőt

$$E_v = \frac{\omega T_i}{1 - \cos \omega T_i}$$

egyenlőséggel definiálva, az

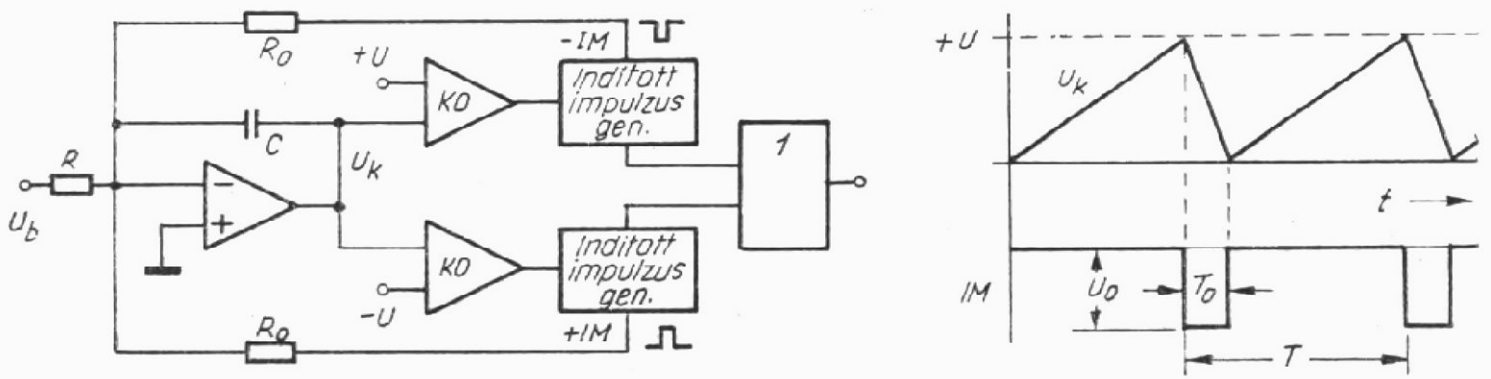
$$\bar{U}_b = U_b + \frac{U_{sz}}{E_v}$$

összefüggéshez jutunk. Minden olyan helyen, ahol $\omega T_i = k \cdot 2\pi$ ($k = 1, 2, 3, \dots$) értéket vesz fel, E_v a végtelenhez közelít, és $\bar{U}_b = U_b$. A T_i értékének a hálózat periódusához való megfelelő illesztésével a hálózati zavaroknak a mérésre gyakorolt hatása eliminálható.

Az integráló átalakító viszonylag hosszú átalakítási idejének csökkentésére olyan áramköri megoldást is alkalmaznak, mely a kisütési szakaszt két részre bontja (2.26.c ábra). Az 1. szakaszban, az U_1 feszültség szintig, nagy kisütő áramot alkalmaznak. A T_1 időre eső óraimpulzusok számát a nagy helyértékek számlálója regisztrálja. Az U_1 meghatározott feszültség szint elérésekor a kisütő áram lecsökken, s az átalakító rátér a 2. kisütési szakaszra. A T_2 időre eső óraimpulzusokat a kis helyértékek számlálója gyűjti össze. Ahhoz, hogy a két számláló együtt adhassa az eredményt, a kisütési áramok arányát előírt módon kell beállítani. A háromlejtős átalakító átalakítási ideje csaknem egy nagyságrenddel rövidebb lehet a kétlejtős átalakítóénál.

Bipoláris feszültség—frekvencia (v — f) átalakító

A feszültség nem csak impulzusszámra, hanem impulzusekvenciára is konvertálható. A v — f átalakítók egyik legismertebb képviselője az integráló feszültség—frekvencia átalakító. Bipoláris változatát a 2.27. ábra mutatja be. A bemenő fokozatot képező integrátor kimenete $+U$ és $-U$ küszöbszintű komparátorokra csatlakozik. Amikor $U_k = \pm U$, az előjelnek megfelelően indított impulzusgenerátor egyrészt a kimenő VAGY kapura, másrészt a bemenetre impulzust bocsát. Ez utóbbi



2.27. ábra.
Bipoláris feszültség—frekvencia ($v-f$) átalakító

a C kondenzátort előírt T_0 időn át meghatározott árammal kisüti. Kimutatható, hogy a feltöltésből és a kisütésből álló teljes integrálási ciklus ideje

$$T = \frac{R}{R_0} \cdot \frac{U_0 T_0}{U_b}$$

ami

$$f = \frac{R_0}{R} \cdot \frac{U_b}{U_0 T_0}$$

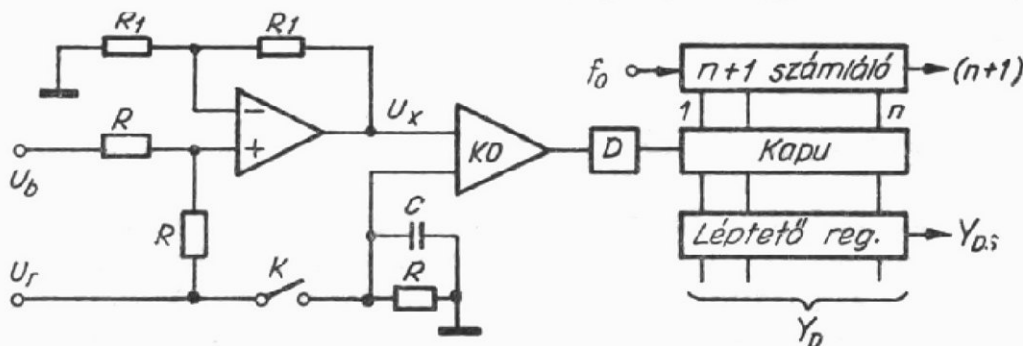
ismétlődési frekvenciát definiál. Amennyiben az átalakító az $U_0 T_0 = \text{konstans}$ feltételt kielégíti, az f a bemenő feszültséget lineárisan képezi le. A teljes A—D átalakításhoz a $v-f$ konverzió túl szükség van még az f mérésére is, ehhez külön frekvenciamérőt kell alkalmazni.

Logaritmikus A—D átalakító

A logaritmikus jelleggörbéjű átalakító nem az analóg érték, hanem az analóg érték logaritmusának digitális ekvivalensét adja. A logaritmikus összefüggés megvalósításának egyik eljárása az R ellenállással kisütött kapacitás exponenciális jelleggörbéjén alapul. A kisütést az U_r referenciafeszültségről indítva, a kondenzátor momentán feszültsége és a kisütés kezdetétől eltelt idő között

$$\Delta t = -RC \ln U_x / U_r$$

összefüggés áll fenn. Amennyiben a kisülő kondenzátor olyan komparátor referenciaforrását képezi, melynek másik kapcsán U_x feszültség van jelen, a komparátor a kisütés megkezdésétől számított Δt idő múlva ad jelzést (billen át). A 2.28. ábrán



2.28. ábra.
Logaritmikus A—D átalakító

látható erősítőelrendezéssel a bemenő feszültség és U_x között $U_x = U_r - U_b$ transzformációt létesítve, a

$$\Delta t = -RC \ln \frac{U_r - U_b}{U_r}$$

összefüggés az U_b -t logaritmikusan képezi le az időre.

Az átalakítónak két azonos időtartamú ($T = n/f_0$) munkafázisa van. Az első munkafázis az indítójellel kezdődik: a K kapcsoló zárul; f_0 az $(n+1)$ -bités számláló bemene-tére kerül; a kondenzátor töltődik: Az első munkafázis befejeztére, a második megkezdésére n óraperiódus múlva, azaz a számláló $(n+1)$ helyértékéről nyert jel hatására kerül sor: a K kapcsoló nyílik, az U_r -re feltöltött kondenzátor megkezdí kiséletét. (Az RC időállandót úgy választják meg, hogy a kondenzátor feszültsége T idő alatt a nullszintet előírt határon belül közelítse meg). Amikor a kondenzátor az U_x szinten áthalad, a komparátor a D differenciáló elem útján a számláló tartal-mát a léptetőregiszterbe kapuzza. A léptetőregiszterben tárolt szám Δt -vel, tehát az U_b bemenő feszültség logaritmusával arányos.

2.4. Kapcsolók és kapcsolórendszerek

A mérőhálózat lényeges építőelemei a különféle kapcsolók és kapcsolórendszerek. Általuk szervezi a számítógép a hálózat egységeit a momentán feladatnak meg-felelő mérőrendszerre, létesíti a tárggyal való kapcsolatot, hajtja végre a mérőké-szülékek programozását, s a különféle átalakítóknak is szerepet játszanak.

Analóg kapcsolók

Az analóg kapcsoló a hálózat analóg szakaszainak elektromos vezérlőjelek hatása alapján való összeköttetésére vagy szétválasztására szolgáló eszköz. A digitális kap-csoló feladata viszonylag egyszerű; általános esetben két logikai érték¹ valamelyikét kell továbbítania, s ezt a feladatát az átvitt jel szintjének lényeges torzítása mel-lett is hibátlanul végrehajthatja. Az analóg kapcsoló a jelet számottevően nem befo-lyásolhatja; ideális esetben, megszakított állapotában tökéletes szigetelést kell adnia, zárt állapotában tökéletes összeköttetést kell létesítenie a két vonalszakasz között. Természetesen sem a tökéletes szigetelés, sem a tökéletes összeköttetés feltétele a va-lóságban nem valósítható meg. A követelményeket ezért mindenkor az adott felté-telekhez kell mérni, melyek nagy jelszint esetében pl. egészen mások lehetnek, mint kis jelszint esetében. Az analóg kapcsolóknak két nagy osztályát különböztethet-jük meg:

- a mechanikus és
- a félvezető

kapcsolókat.

A *mechanikus kapcsolók* sorában az idők folyamán számos változat került előtérbe az egyszerű relétől kezdve a keresztrudas (crossbar) kapcsolón át a körpályán mozgó kontaktusú letapogató kapcsolókig. Jelenleg a mérés technikában csaknem ki-zárólag a reed-relét és származékait alkalmazzák.

A reed-relének nyugalmi állapotban nyitott (A alak), nyugalmi állapotban zárt (B alak) és két álló érintkezős (C alak) alaptípusait különböztetik meg. Ismeretesek ez alaptípusok továbbfejlesztett módosulatai és az alaptípusokból felépülő kombi-nált változatok is.

Száraz, ill. higanyal nedvesített kontaktusú relék különböztethetők meg. A higanybevonat előnye a kontaktus stabilitása, a zárás és nyitás igen meredek homloka, továbbá a pergésmentesség; a korábbi változatok nagy hátrányától a pozícióérzékenységtől (az üvegkapszula munkahelyzetéből pl. $\pm 15^\circ$ -nál nagyobb szögben nem fordítható el) az újabb relékonstrukciók már nem szenvednek.

A reed-relé elektromágnes vagy állandómágnes segítségével működtethető. Megbízhatóságát a kontaktusok védettségeinek köszönheti, átmeneti ellenállása csekély, kapcsolási ideje rövid, az elektródák közötti kapacitás kicsiny, a szigetelési ellenállás pedig nagy. Ugyancsak jelentékeny szigetelési ellenállás választja el a kontaktusokat a vezérlőrendszerrel (elektro- vagy állandómágnes).

A reed-relének kis jelszintű áramkörökben való alkalmazásánál bizonyos olyan szempontokat is figyelembe kell venni, melyek általánosságban elhanyagolhatók. Így: a termoelektromos, elektrosztatikus, induktív, piezoelektromos, magnetostruktív és triboelektromos jellemzőket. Míg az általános célú reed-relék termoelektromos eltolási hibája elérheti az $500 \mu\text{V}$ -t, a kisjelű változatoknál ez az érték $5 \mu\text{V}$, ami speciális kapcsolástechnikával tovább javítható. A kontaktusoknak a záródás után az elektromágneses térben való továbbmozgása induktív és magnetostruktív jellegű zajt generálhat. A kontaktusok szétválasztása utáni elektródapergés pedig a kondenzátormikrofonhoz hasonló effektussal, ugyancsak zajteljesítményt szolgáltat. Ezeknek a zajoknak a frekvenciatartománya 1 KHz szomszédságában, attól felfelé fekszik, hatásuk tehát az egyenfeszültségű és az igen nagy frekvenciás jeltartományban elhanyagolható. Figyelemmel kell lenni a záródással együtt járó elektródapergés jelenségére is, ami bizonyos időn át a kontaktuspár ismételt zárásával, ill. nyitásával, de legalábbis a kontaktusellenállás modulációjával járhat együtt. Erre való tekintettel alkalmaznak higanyal nedvesített kontaktusokat. Kisjelű reed-relék esetében különös figyelmet kell fordítani a mágneses árnyékolás problematikájára is. Nem megfelelő árnyékolás esetében nem csak a környező szórt terek (pl. hálózati transzformátorok), hanem még a földmágnesség is zavarhat.

A jellegzetes reed-relé kontaktus átmeneti ellenállása $\approx 100 \text{ m}\Omega$, a kisjelű reléké $< 5 \text{ m}\Omega$ és a változás $10\,000$ kapcsolás után sem nagyobb $1 \text{ m}\Omega$ -nál. A kontaktus átmeneti ellenállás nagysága és stabilitásának kérdése főleg bizonyos analóg mérőáramkörök (pl. nyúlásmérőbéllyeg) csatlakoztatása kapcsán áll előtérben.

A reed-relék jellemző adatai közül meg kell említeni még a be- és kikapcsolási időt, amely 1 ms , ill. $0,5 \text{ ms}$ (l. 2.1. táblázat).

A félvezető analóg kapcsolók három különböző alapeleme a bipoláris tranzisztor, a térvezérlésű tranzisztor és a fototranzisztor. A bipoláris tranzisztor kis jelszintű áramkörökben az átmenet feszültségesése miatt közvetlenül nem alkalmazható. Az átmenet hatása bizonyos mértékben kompenzálható ugyan, de teljesen meg nem szüntethető. A bipoláris tranzisztor másik hátrányos vonása, analóg kapcsolóként való alkalmazása szempontjából, nem vezető állapotú jelentékeny visszarama. A fototranzisztort ugyanazok a negatív tulajdonságok terhelik, mint a bipoláris tranzisztort. Egyes alkalmazási esetek kapcsán azonban szigetelt vezérlőelektródája (fényemittáló dióda) az áramkörtervező számára speciális előnyöket nyújthat.

A három félvezető kapcsolóelem közül, legelterjedtebben, a térvezérlésű tranzisztort (FET) alkalmazzák, mégpedig mind a JFET (záróréteges, junction FET), mind MOS (metall oxid semiconductor FET), mind pedig a CMOS (complementary MOS) változatait. A térvezérlésű tranzisztorokhoz mint kapcsolóelemekhez, vezérlőelemként esetenként bipoláris tranzisztor is társulhat.

A térvezérlésű tranzisztor — többségi hordozóval működő félvezető eszköz lévén — kapcsolóként való alkalmazásának előnye az, hogy nincs átmeneti feszültsége (nullponteltolása), másrészt, hogy bilaterális vezető. A CMOS elem (mely p-csatornás és n-csatornás MOS FET tranzisztorokból áll) sajátossága a kicsiny meghajtó-

teljesítmény-igény, a csekély disszipáció, az elhanyagolható szivárgási áram és az igen nagy kivezérlési tartomány, melynek határa a tápfeszültség közelében fekszik.

A tervezérlésű tranzisztorkapcsolók között, ugyanúgy, mint a reed-relék esetében, A, B, C kapcsoló formákkal, ill. ezek különféle multiplikatjaival találkozunk. Négy különböző felépítésmódú és alkalmazáskörű FET kapcsoló különböztethető meg:

- a hibrid (bipoláris + MOSFET vagy JFET),
- a CMOS,
- a virtuális föld és
- a pozitív jel

rendszerű kapcsolók.

A *hibrid kapcsoló* bipoláris monolitikus hajtó áramkörből (lapkából) és MOS FET vagy JFET kimenő egységből (lapkából) épül fel. Csaknem minden szempontból felülmúlja ezt a monolitikus CMOS *kapcsoló*, mely a hibrid technikával szemben gazdaságosabb, megbízhatóbb, és érvényesül benne a CMOS elem által nyújtott összes előbb felsorolt előny. Mind a hibrid, mind a CMOS kapcsolók alkalmasak nem csak egyenáramú, hanem üzemi határaikon belül váltakozó áramú jel átvitelére is.

A virtuális föld rendszerű kapcsoló a műveleti erősítő virtuális földpontján végez áramterelő kapcsolást. A kapcsolóelem p-csatornás JFET, mely TTL logikával közvetlenül vezérelhető; külön hajtót tehát nem igényel. A kapcsoló nyitott állapotában a forráspontot dióda fogja meg. Az erősítő visszacsatoló ágában elhelyezkedő JFET hőmérsékletkövető kompenzálást végez.

Amint a virtuális föld rendszerű kapcsoló, ugyanúgy a pozitív jel rendszerű kapcsoló is az általánosság feladásával egy speciális esetre, a pozitív jeltartományra optimalizált megoldás.

A 2.1. táblázat a reed-relé jellemzőinek tipikus értékeihez viszonyítva foglalja össze a különböző FET kapcsolók főbb tulajdonságait.

2.1. táblázat

Jellemzők	Kapcsolótípusok				
	hibrid	CMOS	virtuális föld	pozitív jelű	reed-relé
Jeltartomány: V	± 8	± 14	± 15	0 -tól + 10	± 300
Kapcsoló zárt ellenállása: Ω	30	75	100	100	0,005—0,1
Kapcsolási idő: μs (be/ki)	0,5/1,0	1,0/0,5	0,5/0,5	0,2/0,2	1000/500
Kapcsolási állapot nyugalmi árama: mA	3,5	0,1	nincs	nincs	10 mA, 15 V
Logikai kompatibilitás	van	van	van	van	nincs

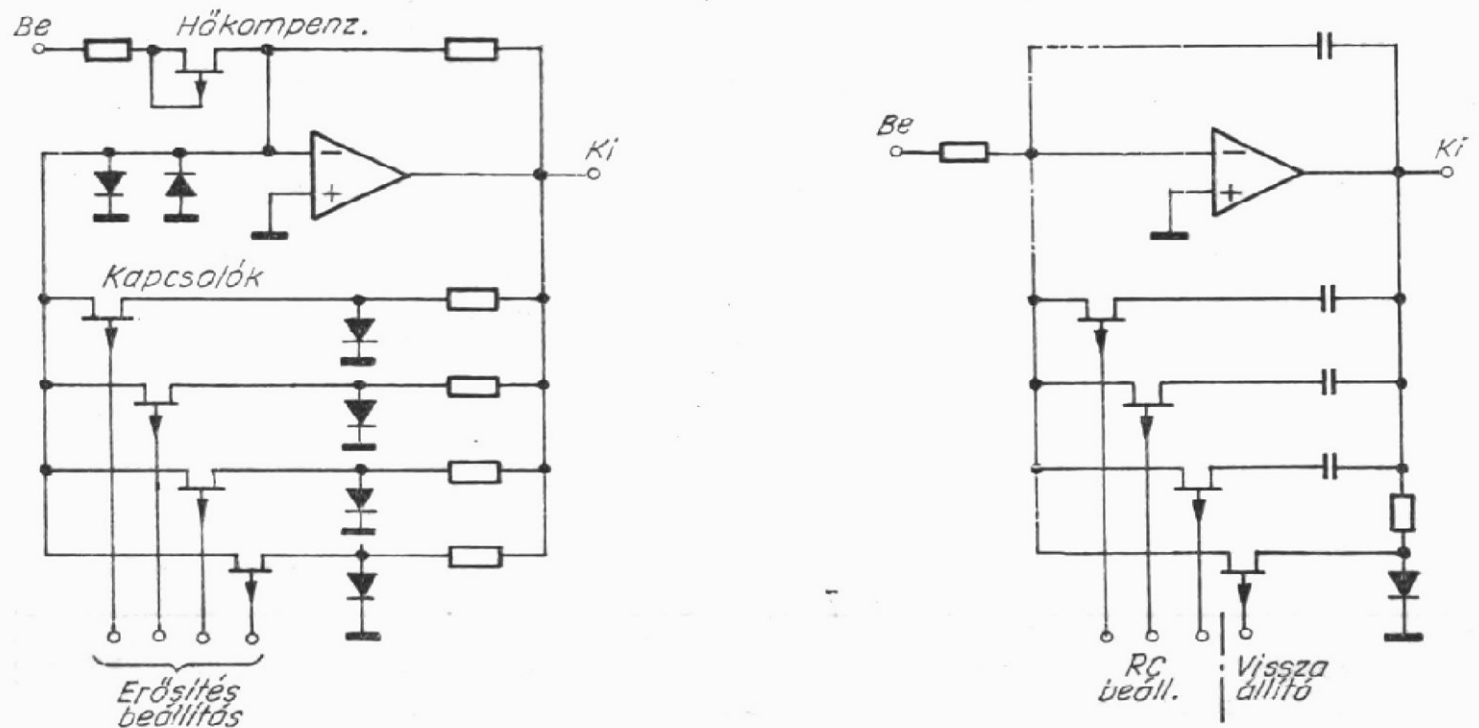
Analóg kapcsolók alkalmazása

Az analóg kapcsolókat — mint a 2.4. pont bevezető soraiban említettük — az automatizált rendszer mérőhálózatán belül számos különböző feladatra használják. Közlebről tekintve, négy fő alkalmazási terület különböztethető meg:

1. Kapcsolatlétesítés a digitális és az analóg világ között.
2. A mérőhálózat funkcionális egységeinek jelvezérlése.
3. A mérőhálózat analóg egységei közötti, a program előírásai szerinti kapcsolat létesítése.
4. A tárgy, ill. a tárgy környezetének a mérőhálózzal való, előírás szerinti kapcsolat-létesítése.

Az 1. feladatkörbeli alkalmazásra példaként a D—A és A—D átalakítókat említjük meg. Az előbbieken kapcsolók képezik le a digitális jel információ tartalmát annak analójjára, az utóbbiakban a kvantálási folyamat egyes szintjein végzett döntések eredményeit csatolják a soron következő döntésekhez, s részesei a mennyiségnek analógról digitálisra való leképzése folyamatának.

Az analóg kapcsolók 2. feladatkörére a hálózat funkcionális egységeinek programirányításáról szóló fejezet részletesen kitér. A műveleti erősítő erősítése, valamint az integrátor időállandója kapcsoló által való vezérlésének példáit itt előrebocsátjuk (2.29. ábra).



2.29. ábra.

Analóg kapcsolók alkalmazása programvezérlés végrehajtó eszközeként (két példa)

A mérőhálózat építőelemei (mérőkészülékek, modulok, tartozékok) közötti kapcsolatok — a rendszer flexibilitásának mértékétől függően — változtathatók. A rendszer moduláris építőelemét képező számos analóg—analóg mérőkészülék közül a program által kiválasztott egység kimenetét analóg kapcsolórendszer köti össze az A—D átalakítást ellátó egység bemenetével.

Az automatizált mérőrendszer általában kapcsolók segítségével szervez és létesít összeköttetést a tárggyal és környezetével. Kapcsolók rendelik a hálózat méréspontjait és hatáspontjait a tárgy mérés-, ill. hatáspontjaihoz. Kapcsolók végzik a hálózat valamely méréspontja és a tárgy számos méréspontja közötti multiplifikálást, több mérőhely esetén ők valósítják meg a kiválasztott mérőhellyel való összeköttetést.

Analóg kapcsolók által hajtja végre továbbá a mérőrendszer a tárgy üzemi környezetének szimulációját is.

Az analóg kapcsolók alkalmazása a negyedik szerepkörben természetesen nem kizárólagos; a digitális mérőrendszerekben vagy az általános mérőrendszerek D—D szakaszain az analóg kapcsolónál általában egyszerűbb és lényegesen olcsóbb digitális kapcsolókat használnak.

Kapcsolótömbök

A reed-relé mint diszkrét elem könnyebb alkalmazhatósága érdekében egyes mérés-technikai orientációjú relétípusokat modulárisan beépíthetően alakítanak ki. Az ilyen *moduláris rendszerű relékből* igen egyszerűen (pl. dugaszolással) képezhetők kapcsolótömbök.

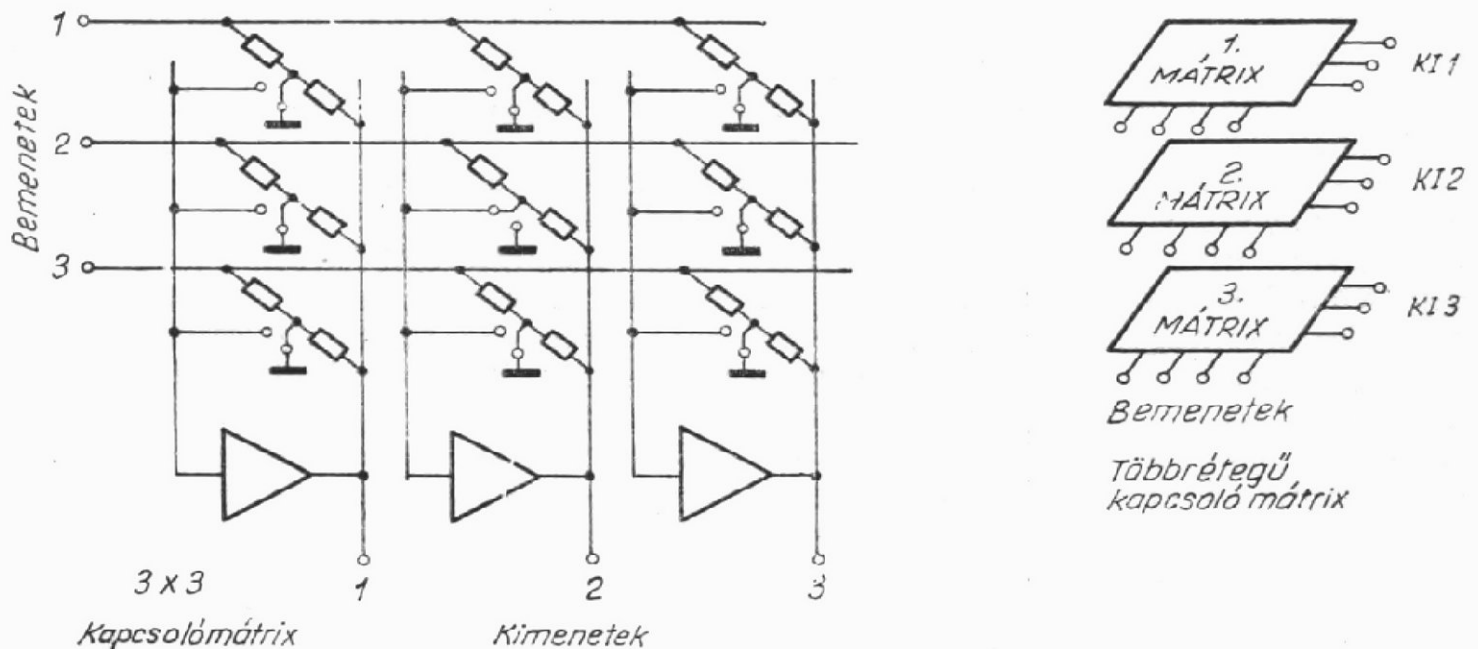
A félvezető kapcsolók sorában diszkrét elemek éppúgy szerepelnek, mint nagyobb tömbök. Gyakran alkalmazott modul pl. a kapcsoló kvád. Azonban számos más kombináció is előfordul.

A több kapcsolóból álló tömbök sajátos típusánál a *multiplex kapcsolóegység*nél a kapcsolók egyik fegyverzete közös. A multiplex kapcsoló bizonyos változatainak alkalmazásával csak az imént is találkoztunk (l. 2.29. ábra), a legjelentősebb szerepet azonban minden bizonnyal a tárgy és a mérőhálózat összeköttetésében játsszák.

A kapcsolótömbök ugyancsak jelentős képviselője a *mátrix-rendszerű* kapcsoló, mely egy M és egy N halmazból tetszés szerint kiválasztott egy-egy vonal összeköttetésére szolgál. A többretegű mátrix-kapcsolóval a rétegek számától függő több-dimenziós összeköttetés valósítható meg (2.30. ábra).

Különleges osztályt képeznek a koaxiális kapcsolók, ill. kapcsolótömbök, melyeknek sajátossága ki- és bemenetük nagyfrekvenciás illesztettsége és az igen nagy sáv-szélesség (pl. 0...300 MHz).

Míg a mátrixkapcsolókban félvezető és reléelemek egyaránt szerepelnek, a koaxiális kapcsolókban egyedül reléelemeket alkalmaznak.

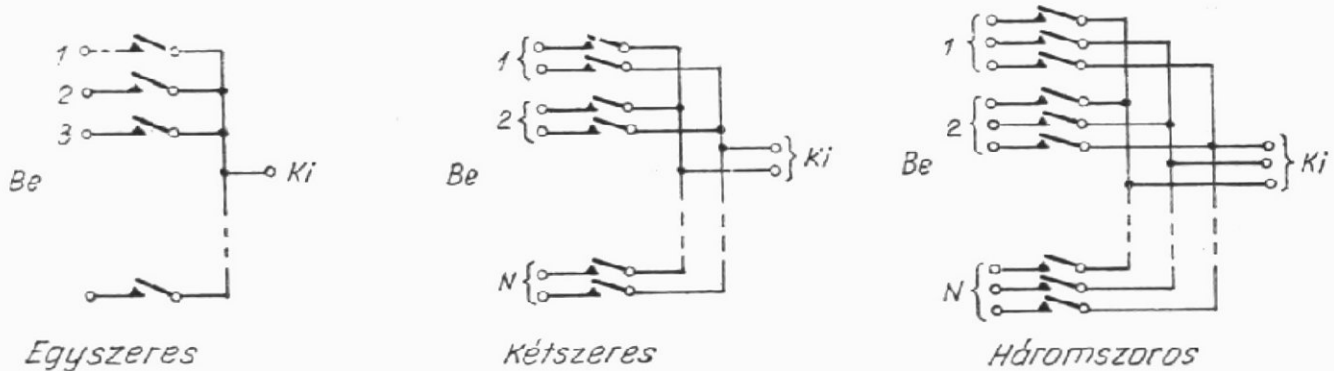


2.30. ábra.
Egy- és többretegű kapcsolómátrixok

Analóg multiplexerek

A jelátvitelre szolgáló analóg multiplexerrel, vagy amint e speciális vonatkozásban nevezik: méréspontváltóval, jelentőségének arányában külön kell foglalkoznunk.

Az analóg multiplexer jellemzői közül első helyen a csatornaszámot, második helyen az egy mérőcsatornához rendelt kapcsolók számát kell megemlíteni. Egy-, két és háromszoros kapcsolópontú multiplexereket különböztetünk meg (2.31. ábra).



2.31. ábra.

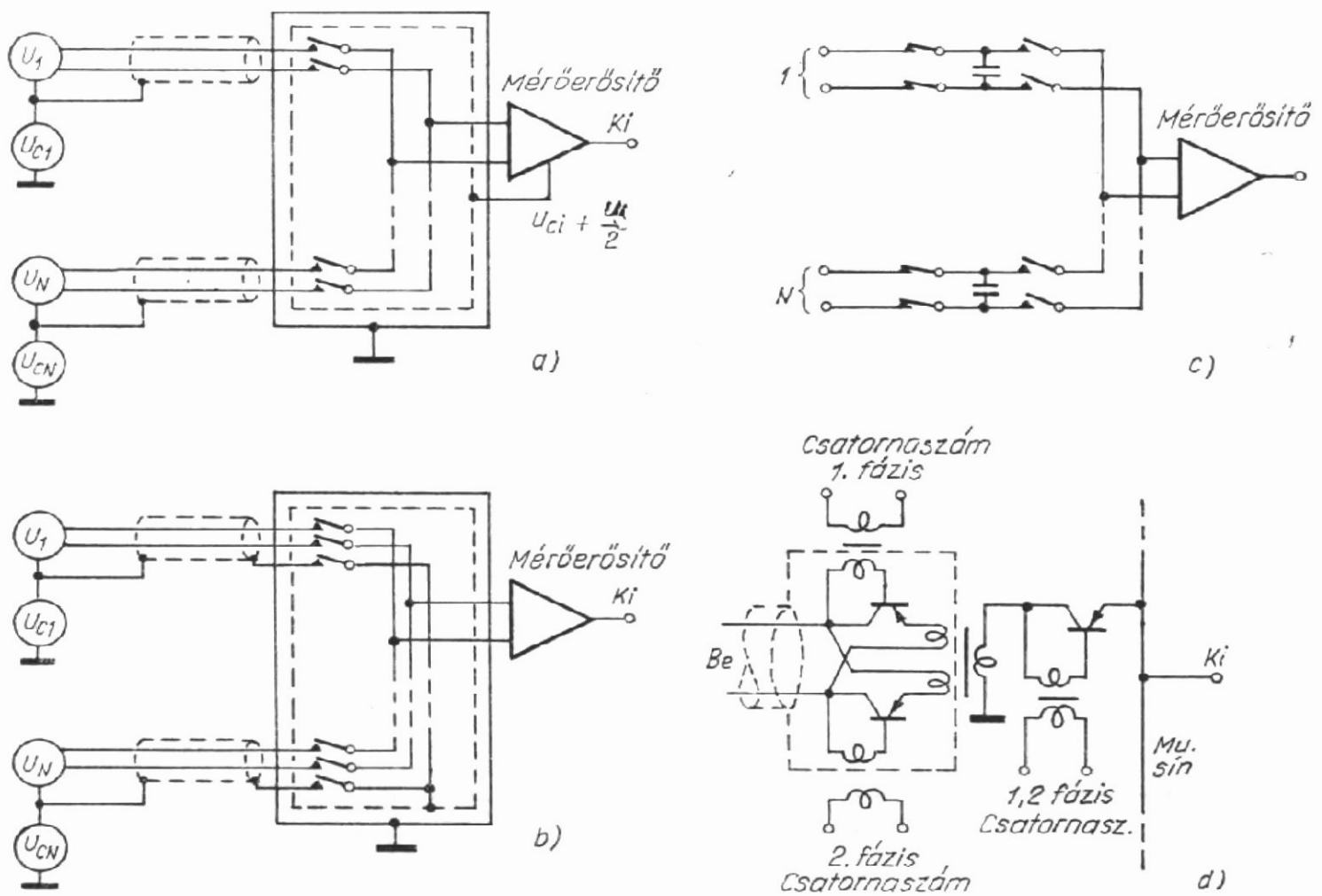
Egy- és többszörös kapcsolópontú analóg multiplexerek

A csatornákat sorszámmal megjelölve beszélhetünk tetszőleges, ill. kötött kiválasztási sorrendű multiplexerről. A kötött kiválasztási sorrendű multiplexer sajátos képviselője a ciklikus letapogató (scanner), mely a csatornák során ciklikusan ismétlődve halad át. A ciklikus letapogató korszerű elektromechanikus rendszerei hengerpaláston koszorúban elhelyezkedő reed-reléből épülnek fel. A reed-relét a hengerpalást mentén szinkron- vagy léptetőmotor által mozgatott állandómágnes kapcsolgatja. A ciklikus letapogató a csatornát a mérőrendszer részére tengelykódoló (shaft encoder) segítségével azonosítja. A tetszőleges kiválasztási sorrendű multiplexer, fordítva nem adja, hanem a mérőrendszertől kapja a csatornaazonosítót, s a csatornaazonosító értelmezésére, a kiválasztás végrehajtására szolgáló áramkörökkel kell, hogy rendelkezzenek.

Egyes multiplexerek minden mérőcsatornájához engedélyező/tiltó kulcs is tartozik, melynek állásától függően a multiplexer az adott csatornát érzékeli, vagy figyelmen kívül hagyja. Ciklikus letapogató esetén pl. ezáltal lehetővé válik, hogy bizonyos mérés kapcsán a multiplexer csak az engedélyezett csatornákat tapogassa le.

Kisjelű mérőhálózatban részben a zavarmentesség, részben közös feszültség elnyomása érdekében különleges csatornakiépítéssel bíró multiplexereket alkalmaznak. Éppen az ilyen speciális rendszerek céljára szolgálnak a két- és háromszoros kapcsolópontú, más néven differencia multiplexerek is. A számos lehetséges változat közül négy különböztet a 2.32. ábra mutat be. A 2.32a ábrán kétszeres kapcsolópontú multiplexer és nagy közös feszültség elnyomású differencia erősítő együttese látható. A multiplexer két kapcsolópontja a csatorna mindkét lebegő mérővezetékét kapcsolja a nagy közös feszültség elnyomású, szimmetrikus bemenetű mérőerősítőre. A jelvezetékek árnyékolása a jelforrás közös feszültség pontjára csatlakozik, a multiplexer belső árnyékolása pedig az erősítő szimetriapontjára. A háromszoros kapcsolópontú multiplexer (2.32b ábra) a momentán kiválasztott jelvezetékek köpenyét a multiplexer belső árnyékolásához köti.

Néhány voltnál nagyobb közös feszültség, ill. a zavarmentesség tekintetében fennálló fokozottabb követelmények esetében a 2.32b ábra szimmetrikus erősítője helyett szigetelt bemenetű erősítőt alkalmaznak. Más multiplexer megoldásokban (2.32c és d) a multiplexer minden egyes csatornabemenete szigetelt. Kétségtelen

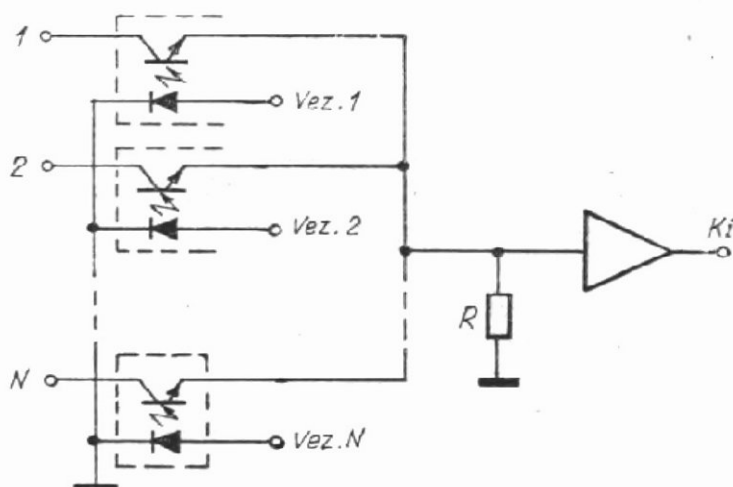


2.32. ábra.
Közös feszültség elnyomását célzó multiplexer hálózatok

(a) kétszeres kapcsolópontú multiplexerrel; (b) háromszoros kapcsolópontú multiplexerrel; (c) komplementis kapcsolópárú multiplexerrel és lebegő kondenzátorral; (d) szimmetrikus modulátor-demodulátor típusú szigetelőbemenettel

azonban, hogy ezek a módszerek mind megbízhatóság, mind gazdaságosság szempontjából elmaradnak a szigetelt bemenetű erősítő mögött.

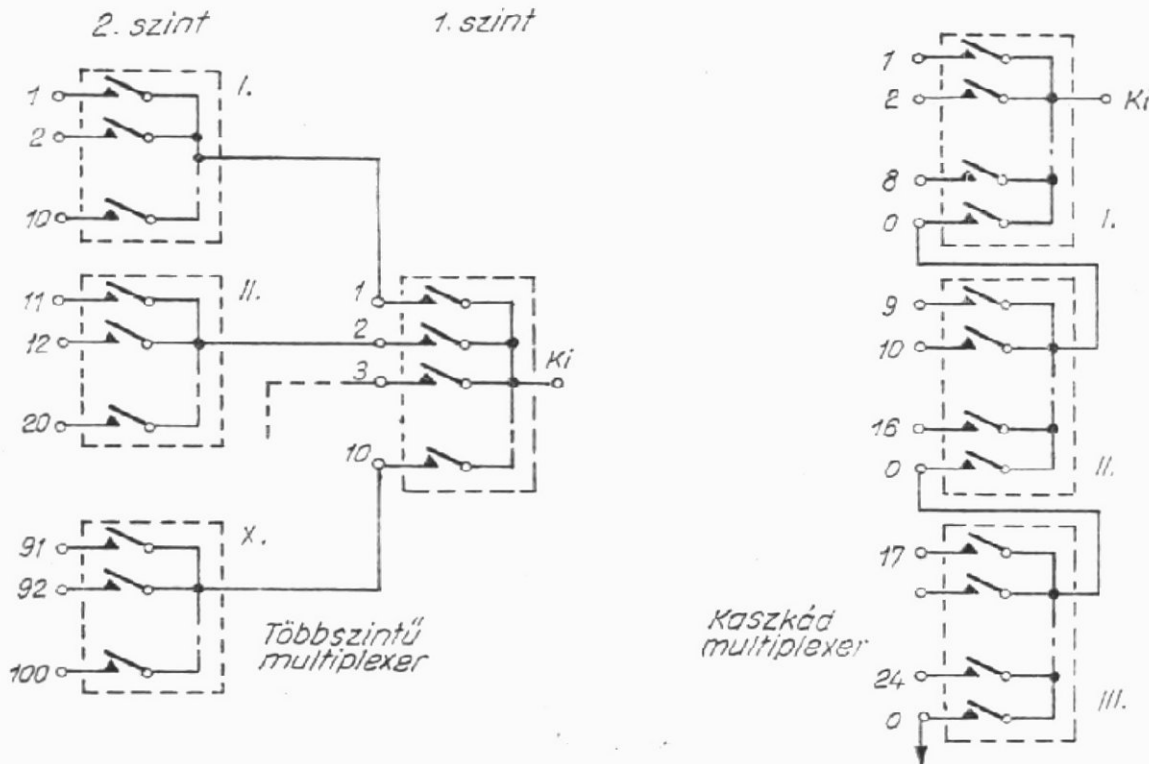
A jelvezetékek és árnyékolások elrendezésének feladatával egyidőben jelentkezik a kapcsolórendszer és az azt vezérlő áramkörök szigetelésének problémája is. A nem megfelelően felépített vezérlőáramkör egyrészt kapcsolási tüskéket vihet be a jeláramkörbe, másrészt ronthatja a rendszer közös feszültség elnyomási tulajdonságait is. A probléma egy megoldásként optikai szigetelőt alkalmaznak (2.33. ábra).



2.33. ábra.
Multiplexer optikailag csatolt kapcsolóvezérléssel

A jel és vezérlőáramkörök közötti igen jó szigetelésen kívül, az optikailag csatolt multiplexer további előnye még az is, hogy vezérlőkapcsai a szokásos logikai áramkörökhöz közvetlenül csatlakoztathatók, továbbá, hogy felépítmódja igen gazdaságos. Az egyetlen hátrány a fototranzisztor relatíve nagy áteresztési árama a tranzisztor nem vezető állapotában. Az átvezetési áram hatása azonban az ábrán R -rel jelölt levezető ellenállás értékének alkalmas megválasztásával jelentékeny mértékben csökkenthető.

A multiplexerek általában moduláris felépítésűek. Attól függően, hogy a modularitás bináris vagy dekadikus struktúrájú-e, az alapmodul kapcsolószáma kettő valamely egész számú hatványával vagy tízzel lehet egyenlő. A modulokból elvileg tetszés szerinti számú bemenő kapoccsal rendelkező multiplexer építhető fel. A modulok összekapcsolásának módjától függően többszintű, ill. kaszkád multiplexereket különböztetünk meg (2.34. ábra).



2.34. ábra.
Többszintű és kaszkád multiplexer

Az analóg multiplexer legfontosabb jellemzői: a beállási idő, az átviteli pontosság, az átviteli ütem, a visszatérési idő, az áthallás, a jeltartomány, a közös feszültségűtés, a ponttól-pontig eltolási hiba, a bemenő impedancia, a zajosság.

A *beállási idő*: a korábbi definíciónak megfelelően itt az alkalmazott kapcsolók kapcsolási időire vonatkozik.

Az *átviteli pontosság*: a bemenő és kimenő érték közötti különbséget a bemenő értékre vonatkoztatva definiálja. Függvénye a kapcsolóimpedanciának, a forrásimpedanciának, a kimenő kapocs terhelésének, a csatornák felépítmódjának, és a jel spektrális összetételének.

Az *átviteli ütem*: a multiplikálás időegységre eső azon maximális ciklusszáma (csatornaszám/s), mely mellett a multiplexer még az előírt átviteli pontosságon belül működik. Értéke — lényegében — a beállási idő függvénye.

A *visszatérési idő*: az a legkisebb időtartam, amelyen belül egy csatorna kétszer nem hívható.

Az *áthallás*: definíció szerint: egy vagy több nyitott kapcsolón át a kiválasztott csatorna jelére szuperponálódó jelhányad. E jellemző értéke erősen frekvencia-

függő, ezért pontos értelmezéséhez a vizsgáló jel spektrális tulajdonságainak definíciója is hozzátartozik.

A jeltartomány: a multiplexer által kezelhető (a minimumtól a maximumig terjedő) feszültségtartomány. Az alsó határt a multiplexer zaja, nullponthibája, áthallása és a közös feszültség áthatás, a felső határt a kapcsoló működési korlátai definiálják. Mérőrendszerekben az átviendő jel minimumára vonatkozó legszigorúbb követelmény termopár és nyúlásmérő bélyeg érzékelőkkel kapcsolatban merül fel, ahol <1 mV (≈ 1 μ V felbontással) jelszint átvitele szükséges. Másrészt 10 V-nál nagyobb jel átvitele csak kivételesen szükséges.

A közös feszültség túrés: az a maximális közös feszültség, melyet a kétszeres, ill. háromszoros kapcsolópontú differencia multiplexerek jellemzőik torzulása nélkül elviselnek. A csatornakiépítés (pl. lebegő kondenzátor) és a multiplexerhez csatlakozó erősítő (pl. szigetelt bemenetű) jellemzőinek is függvénye.

Ponttól-pontig eltolási hibán az összes bemenet földre zárt állapotában, a megengedett átviteli ütemben működő multiplexer bármely kapocspárján szobahőmérsékleten jelentkező feszültséget értenek. A kapcsolótípustól függően eredete lehet termoelektromos erő, szivárgási áram stb.

A bemenő impedanciának különböző értékeiről beszélhetünk. Így a zárt, ill. nyitott kapcsolóálláshoz tartozó csatornaimpedanciáról, csatorna különbségi impedanciáról és közös bemenő impedanciáról. Kieső hálózatnál a forrást terhelő multiplexer-impedancia ismerete is — bizonyos esetekben — szükséges lehet.

A zaj. A külső eredetű zajoktól eltekintve, az erősítővel kiegészülő multiplexernek belső (termikus) zajai is vannak. Ezek a zajok az erősítő sáv szélességének szűkítésével csökkenthetők, ezzel azonban a beállási idő növekedése jár együtt. A zajleválasztás lehetséges másik módja az átlagolás: A számítógép a több ciklusban egymás után felvett csatornaértékek átlagát képezi. Azonban ez a módszer is a mérési idő megnövekedéséhez vezet, azonkívül számítógépidőt is fogyaszt. Általában számolni kell a pontosságra és átviteli ütemre vonatkozó követelmények között kötendő kompromisszummal.

Digitális multiplexerek

A digitális multiplexer különböző digitális jelforrásokból származó adatok egyetlen vonalon való átvitelére (egyetlen vonalra terelésére) szolgáló berendezés. A multiplexer az átvivő vonal idejét vagy átviteli frekvenciasávját osztja meg a különböző források között. Ennek megfelelően idő- és frekvenciaosztású multiplexerekről beszélnek. A digitális multiplexereknek a számítógépben és a számítógép környezetében igen sokféle változata van; világos osztályozásukat és megnevezésbeli megkülönböztetésüket sajnos mindmáig nélkülözzük. Három fő kategóriát:

1. adatátviteli,
2. periferikus készülékek közvetlen csatlakoztatására szolgáló, és
3. egyszerű

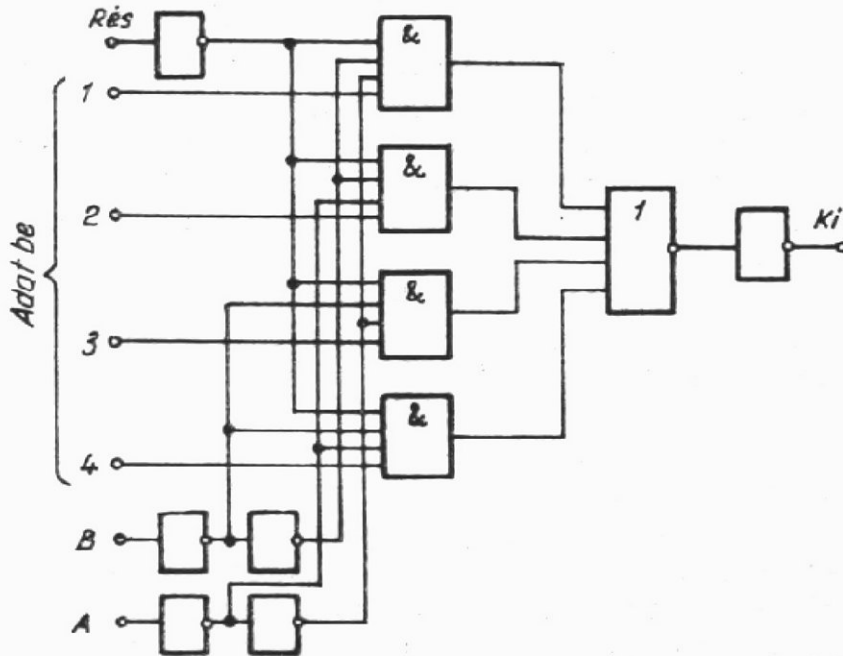
digitális multiplexert különböztetünk meg. Itt most csak a legutóbbi, tehát az egyszerű multiplexerekkel foglalkozunk, mely az előbb ismertetett analóg multiplexer digitális megfelelője.

A digitális multiplexer N számú digitális bemenettel és egyetlen digitális kimenettel rendelkező időosztású eszköz; a bemeneti csatornák közül a címrendszere révén kiválasztott csatornát kapcsolja a kimenetre. Kiválasztási sorrendje tetszőleges, s letapogatóként is működtethető.

Az átvitt digitális kód alakja szerint soros, ill. párhuzamos multiplexert különböztetnek meg. A soros multiplexer felépítmódja elvileg megegyezik az analóg

multiplexerével. Gyakorlatban a különbség a kapcsolókban mutatkozik; a digitális multiplexerben lényegesen egyszerűbb és olcsóbb digitális kapcsolókat alkalmaznak. A paralel multiplexer, kézenfekvő módon, a bitben kifejezett vonalszélességnek megfelelő számú, egymásra rétegzett soros multiplexerből szerveződhet. Alkalmaznak azonban közvetlen paralel multiplexert is, mely áramkörileg gazdaságosabb a soros multiplexerekből álló megoldásnál.

A digitális multiplexerek gyakorlati kivitelei — éppúgy, mint az analóg egységek — moduláris felépítésűek. Különböző csatornaszámú (négy, nyolc, tizenhat) és különböző csatornaszélességű modulokkal találkozunk. A 2.35. ábra négycsatornás



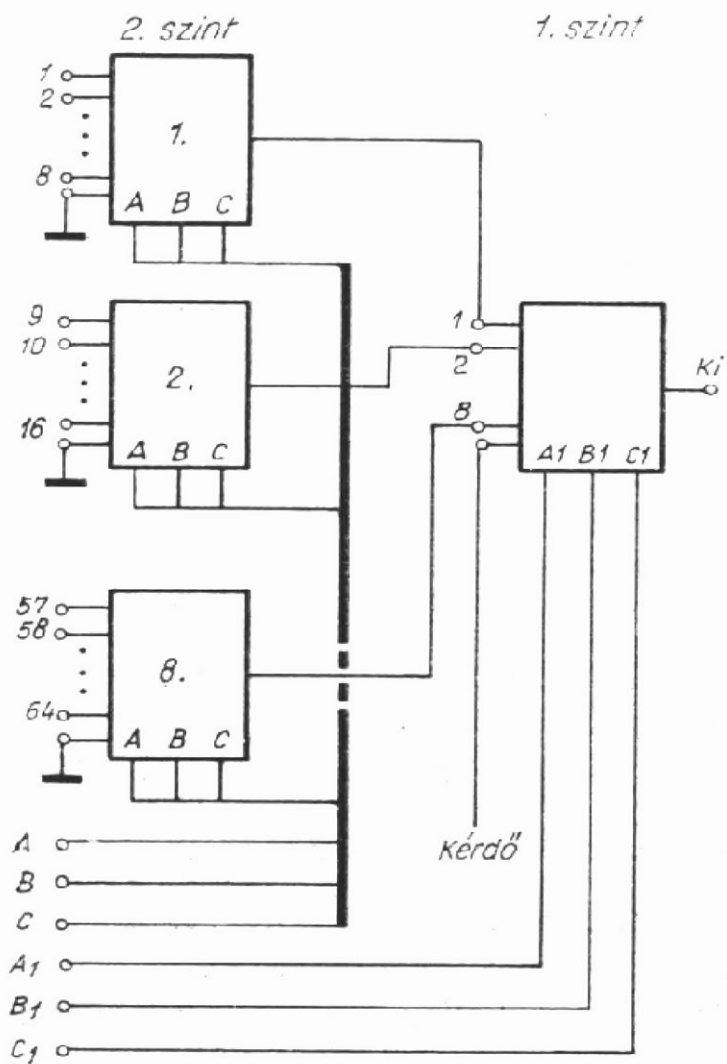
2.35. ábra.
Soros digitális multiplexer

soros multiplexer modul felépítésmódját szemlélteti. A modulokból elvileg tetszőleges csatornaszámú multiplexerek állíthatók össze. A 2.36. ábra nyolccsatornás soros modullal felépített, kétszintű, soros multiplexert szemléltet.

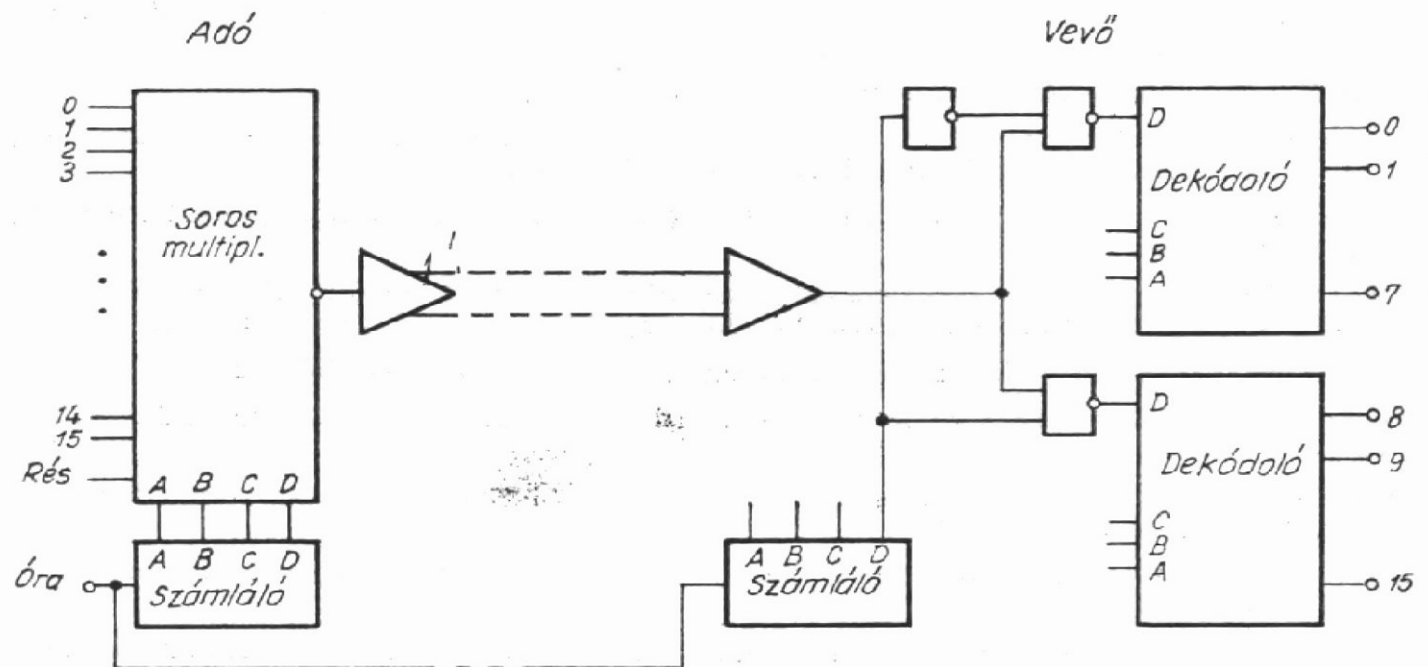
A digitális multiplexer a mérőáramkörben csatornaátkapcsoláson kívül egyéb, különleges feladatokat is elláthat. Alkalmazható különböző memóriák, logikai függvénygenerátorok és aritmetikai áramkörök alapelemeként. A 2.37. ábra pl. párhuzamos — soros átalakító szerepkörében mutatja be. A multiplexer csatornaki-választó vonalaihoz bináris számláló csatlakozik; az így keletkező együttes az órajel hatására sorban letapogatja a bemenetre adott 16 bites paralel kód egyes helyértékeit, s e helyértékek órajellel szinkronizált soros kód elemeiként kerülnek az átvivő-vonalra. A vonal másik végén, ugyanazzal az órajellel meghajtott számláló, két nyolc bites dekódolóval a soros kódot paralel kóddá alakítja vissza.

Pneumatikus multiplexerek

A pneumatikus multiplexer (Scannivalve) a ciklikus analóg letapogató speciális alakja. Alkalmazásának célszerűsége ott kerül előtérbe, ahol egymáshoz közel eső számos nyomáspontot kell letapogatni. A hagyományos eljárás minden nyomásponton külön érzékelő-átalakítót alkalmaz, és az egyes pontokat villamos úton, analóg multiplexerrel kérdezi le. A pneumatikus multiplexer a nyomáspontokat pneumatikus kapcsoló segítségével tapogatja le. A pneumatikus kapcsoló tulajdon-



2.36. ábra.
Többszintű, soros digitális multiplexer



2.37. ábra.
Soros digitális multiplexer alkalmazásán alapuló paralel-soros, soros-paralel átalakító

képpen nem más, mint egy radiális csatornával rendelkező tárcsa. A tárcsát tengelye mentén léptetőmotor forgatja. Forgás közben a csatorna egymás után halad el a körkerületen elhelyezkedő csatlakozónyílások alatt, s eközben az érzékelőt sorban kölcsönhatásba hozza az egyes nyomáspontokkal. A letapogató csatorna-pozícióját tengelykódoló azonosítja.

Annak következtében, hogy a pontonkénti érzékelő és jelvezeték rendszere helyett itt mindössze egyetlen érzékelő és egyetlen jelvezeték szükséges, adott esetben a pneumatikus multiplexer a hagyományos megoldásnál lényegesebben gazdaságosabb, egyszerűbb és megbízhatóbb megoldást nyújthat.

Irodalom a 2. fejezethez

- [1] *Welk, S. A.—Matthews, T. A.*: Understanding operational amplifiers, *Microelectronics*, 4, 2. 1972, 29...48. old.
- [2] *Demrow, R. I.*: Settling time of operational amplifiers, *Analog Dialogue*, 4, 1. 1970, 1...11. old.
- [3] *Hoft, D.*: Applying the analog differentiator, *Analog Dialogue*, 4, 1. 1970, 12...15. old.
- [4] *Schick, L. L.*: Linear circuit applications of operational amplifiers, *IEEE spectrum*, Apr. 1971, 36...50. old.
- [5] *Zicko, C. P.*: New applications open up for the versatile isolation amplifier, *Electronics*, March 27. 1972, 96...100. old.
- [6] High speed optically coupled isolators, Hewlett-Packard application note 939. 11. 1972.
- [7] What is new in operational amplifiers? Analog-Devices application note ED-10/68.
- [8] Signal isolation by fiberoptic transmission, Motorola közlemény, *Electronic Engineering*, Dec. 1970, 65.
- [9] *Artur, K.*: Transducer signal conditioning, *Machine Design*, Sept, 21. 1972, 133...139. old.
- [10] *Patston, W.—Dunbar, C.*: Choosing a sample-and-hold amplifier is not as simple as it used to be, *Electronics*, Aug. 2. 1973, 101...104. old.
- [11] *Davis, S.*: Linear ICs for digital interface applications Part 1: Monolithic amplifiers, *Computer Design*, July 1973, 59...67. old.
- [12] *Naylor, J. R.*: Digital and analog signal applications of operational amplifiers-II, *IEEE spectrum*, June 1971, 38...46. old.
- [13] *Cate, T.*: Designing sample/hold circuits, *Electronic Design*, Oct. 25. 1967.
- [14] *Meyer-Brötz, G.—Kley, A.*: The common-mode rejection of transistor differential amplifiers, *IEEE Transactions*, Vol. CT-13, No. 2. June 1966, 171...175. old.
- [15] *Sheingold, D. H.—Ferrero R. A.*: Understanding A/D and D/A converters, *IEEE spectrum*, Sept. 1972, 47...56. old.
- [16] *Cholet, J.*: La conversion digitale-analogique EMI 150/15—1—(1972) 27...33; 151/15—2—(1972) 43...47. old.; 152/1—3—(1972) 49...58. old; 154/1—4—(1972) 95...100. old.
- [17] *Schmid, H.*: D—A conversion, *Electronic Design* 22, Oct. 24. 1968, 49...88. old.
- [18] *Smith, B. K.*: Digital-to-analog converters and their performance specifications, *EEE* — Nov. 1970.
- [19] *Pastoriza, J. J.*: Specifying measuring and using very high-speed digital-to-analog converters, *Analog Devices E 055—10—4/70*.
- [20] *Gimmel, B.*: AD- und DA- Wandler — Verfahren und ihre Anwendung, *Neue Technik*, Nrn. 3 und 4. 1972.
- [21] *Grzybowski, W.—Wagner, F.*: Nonlinear functions from D—A converters, *Electronic Engineering*, July 1971, 48...51. old.
- [22] *Bonfig, K. W.—Weimann, G.*: Verfahren der elektronischen Analog-Digital-Umsetzung, *Automatik* 1971, Juni, 185—189; Juli, 219...223. old.

- [23] *Aniebona, E. N.—Brathwaite, R. T.*: A review of analog-to-digital conversion, *Computer Design*, Dec. 1969, 45...54. old.
- [24] *Naylor, J. R.*: Digital applications of operational amplifiers, *IEEE spectrum*, May 1971, 79...87. old.
- [25] *Schmid, H.*: A/D conversion, *Electronic Design* 1968 Dec. 5., 49...72; Dec. 19, 57...76; 1969 Jan. 4, 97...112. old.
- [26] *Lugos A.—Tóthmátyás I.*: Nagypontosságú analóg-digitál átalakító ipari alkalmazásra. *Mérés és Automatika*, XX. éf. 5. sz. 175...180. old.
- [27] *Малиновский В. Н.*: Цифровые вольтметры интегрирующего типа (обзор), *Приборы и системы управления*, 2. 1973. 10...14. old.
- [28] *Тарасов, В. Ф.—Шахов, Э. К.*: Полупроводниковые преобразователи напряжение-частота (обзор), *Приборы и системы управления*, 4. 1971, 9...14. old.
- [29] *Grossman, M.*: Focus on reed relays, *Electronic Design*, July 6. 1972, 50...59. old.
- [30] *Bery, S.*: Consider dry reed switches, *Automation*, Oct. 1971, 37...41. old.
- [31] *Fullagar, D.*: Analog switches replace reed relays, *Electronic Design* 13, June 21. 1973, 98...101. old.
- [32] *Foster, M.*: Scannareed — a new approach to rotary scanners, *Instrum. Practice*, March 1970, 170...171. old.
- [33] *Strassberg, D. D.*: Multiplexing and grounding in digital data acquisition systems, *ISA Transactions*, Vol. 11. No. 3. 259...273. old.
- [34] *Mahan, R. E.*: Low-level multiplexing, *Instrum. and Control Systems*, Oct. 1969, 133...136. old.
- [35] *Naylor, J. R.*: Multiplexers and converters, *IEEE spectrum*, May 1971, 79...87. old.
- [36] *Das, S.*: Multiplexing analogue signals with optically coupled isolator, *Int. J. Electronics* Vol. 34, No. 4. 1973, 569...570. old.
- [37] *Krigman, A.*: Guide to selecting digital multiplexers, *Instrum. and Control Systems*, Nov. 1973, 63...68. old.
- [38] *Krausener, J. M.*: Multiplexeurs digitaux TTL et leurs applications, *Electricque Industrielle* 1970, Avril, 218...224; Mai, 293...295. old.
- [39] *Baker, W. J.—Hunt, H. C.*: Minimizing costs in computerizing a pneumatically instrumented plant, *Proc. 27th Annual Conf. and Exhibit. of ISA*, 1972, 572, 1...5. old.

3.

A rendszerorientált számítógép — Szerkezet és architektúra

3.1. A szerkezet

A számítógépet szerkezetének (hardware) felépítésmódja, valamint programozási, működési tulajdonságai — architektúrája — jellemzik. Szerkezet és architektúra között nincs egyértelmű kapcsolat. Adott architektúra több, egymástól különböző szerkezettel is megvalósítható, másrészt a mikroműveletek szintjén egyetlen szerkezettel számos különböző architektúra is emulálható. Bizonyos területeken a gép használatához az architektúra ismerete egyedül is elegendő. A mérőrendszerben alkalmazott gépet azonban mint az építőelemek egyikét, más rendszerelemekkel kell összekapcsolni. A rendszertervezéshez ezért a gép fizikai struktúrájának ismerete annál inkább szükséges, mivel a fejlődés a gépek zárt formáitól a moduláris számítógéprendszer irányába halad; nemcsak a mérőrendszert, de magát a számítógépet is, a konkrét szükségleteknek megfelelő modulokból, esetenként a rendszertervező kell, hogy összeállítsa.

Attól függően, hogy mely szerkezeti egységeket tekintünk építőelemnek, a gép különböző fizikai strukturális szintjei különböztethetők meg. Ezek sorából az alábbiakban a tárgyunk szempontjából lényeges Memória—Processzor—Készülék (MPK) és Regiszter—Operátor—Vezérlő (ROV) szintet tanulmányozzuk.

3.1.1. A ROV struktúra

Építőelemek, hálózatok, műveletek

A ROV struktúra alapelemei: a regiszter, az operátor és a vezérlő, melyeket különböző adatátvivő és vezérlővonalak egészítenek hálózattá. A magasabb szintű funkcionális egységek ROV struktúráját ezen építőelemek tulajdonságai, továbbá hálózataik szervezésmódja jellemzi.

A számítógépen belül mind az MPK, mind a ROV szinten, adathálózat és vezérlőhálózat különböztethető meg. Az MPK szinten e hálózatok gyakran nem különülnek el egymástól teljesen; bizonyos átvivő vonalak mind az adathálózat, mind a vezérlőhálózat részét képezhetik. A ROV szinten azonban a kétféle hálózat elkülönülése sokkal világosabb. A számítógépnek a feladattal kapcsolatos — lényegi — folyamatai az adathálózat(ok)ban zajlanak le. Az adathálózat folyamatait a vezérlőhálózat vagy hálózatok útján a vezérlő, vagy vezérlők bonyolítják le.

A ROV szintű *elemi (piko) műveletek*: regiszter- és operátorműveletek. Az adathálózatban végrehajtható ezen elemi műveletek összessége szerkezeti adottság. A pikoműveletekből az adatprocesszálas szempontjából értelmezhető *mikroműveletek*

állíthatók össze. Az adathálózat által végrehajtható mikroműveletek összessége a gép strukturális jellemzője. A mikroműveletek minden értelmezett részhalmaza egy-egy mikroarchitektúrát definiál. A gép utasításszintű architektúráját a mikroműveletek zárt készletéből felépített algoritmusrendszer határozza meg.

ROV-szintű adathálózatok

A ROV-szintű adathálózat alapelemei: operátorok (műveletvégzők), regiszterek és vonalak.

Az *operátorok* általában többműveletűek. Az operátor által elvégzendő momentán műveletet — pikoutasítással — a vezérlő jelöli ki.

Egyoperandusú átmenő operátor (AOP) és többoperandusú aritmetikai-logikai operátor (ALOP) különböztethető meg. Az AOP egy bemenettel és egy kimenettel, az ALOP egynél több bemenettel, eredménykimenettel, valamint bizonyos állapotkimenetekkel (pl. túlfolyás) is rendelkezik. Az n -bemenetű ALOP általában $\cong n$ operandusszámú műveletet végezhet.

A *regiszter* az adathálózat tárolóeleme. A regiszterben tárolt adat a regiszter bemenetére, ill. kimenetére vonatkozóan több részadatból is összetevődhet (részadatok összekapcsolása egyetlen adattá; egyetlen adat felbontása részadatokká). A regiszter szélessége szerkezeti jellemző, mely nem szükségképpen azonos a hozzá csatlakozó átvivő vonalak, vagy a programozási szinten kezelt adatok szélességével.

Regiszterműveletek: a törlés, az írás és az olvasás. *VAGY írású* és *másoló írású* regiszter különböztethető meg. Az előbbiben beírás után az eredeti tartalom és a beírt adat logikai összege képződik, az utóbbiban a regiszter eredeti tartalma az új adat beírásával megsemmisül. A *VAGY írású* regiszterek írás előtti törlése egyes esetekben (pl. ferritmemóriák) roncsoló olvasással, más esetekben (pl. bizonyos processzorok akkumulátorainál) külön törlő utasítás segítségével történik (szerkezet megtakarítása programozási eszközökkel). Vannak *roncsoló olvasású* regiszterek, melyek az olvasással együtt tartalmukat is elvesztik, és *nem roncsoló olvasású* regiszterek, melyek tartalmukat az olvasás után is megőrzik. Amennyiben roncsoló olvasású regiszter tartalmát meg kell őrizni, úgy kiolvasás után az eredeti tartalmat vissza kell írni a regiszterbe.

Azt a regisztert, mely saját tartalmán különböző műveleteket (inkrementálás, komplementálás, eltolás) képes végezni, operatív regiszternek nevezzük. A regiszter állapotkijelzőkkel (pl. tartalom = nulla) is rendelkezhet. A számos változat sorában olyan adathálózattal is találkozhatunk, melynek valamennyi regisztere egységesen azonos operatív regiszter.

Az adathálózat a *ponttól-ponthoz*, ill. a *sínszervezés* elvét követheti. A ponttól-ponthoz szervezésű hálózat regisztereit, ill. operátorait egymással közvetlen, individuális vonalak kötik össze, melyek elején és végén (de legalább egyik oldalt) a regiszterek vonalzáró kapui állnak. A sínszervezésű hálózatban az adatátvitelre egy vagy több olyan közös sín szolgál, melyek mindegyikét nagyobb számú (esetleg az összes) regiszter, ill. operátor használhatja. A regiszterek és sínek között itt is kapuk helyezkednek el.

Adatátadás kapcsán a kiindulási regisztert forrásregiszternek, az érkezési regisztert rendeltetési regiszternek nevezik. Két regiszter közötti adatátadás a hálózat-szervezés mindkét módja esetében a forrásregiszter és a rendeltetési regiszter megfelelő vonalzáró kapuinak nyitásával történik (regiszterolvasás és regiszterbeírás). Két regiszter közötti adatátvitel kapcsán az adat módosulhat, ha átmenő operátoron halad át. A hálózatban helyet foglaló ALOP-hoz egyidejűleg két vagy több forrásregiszter és egy vagy több rendeltetési regiszter is tartozhat.

Számos kisgép adathálózataiban mind a ponttól-ponthoz, mind a sínszervezés elvét alkalmazzák.

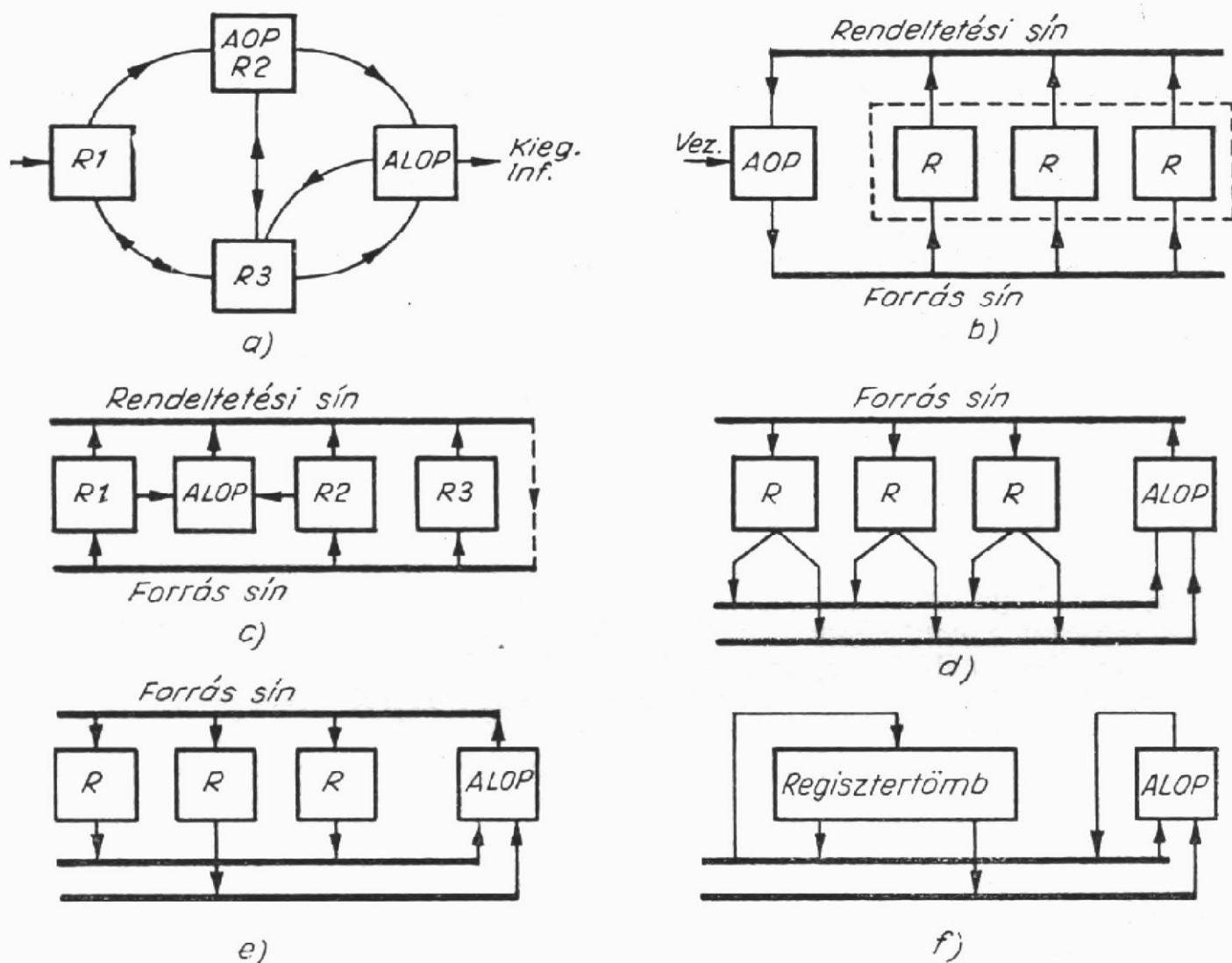
A 3.1a ábra a ponttól-ponthoz szervezőmódot szemlélteti, példaképpen három regiszterrel (közülük R2 operatív). ALOP hálózatbeli elhelyezkedéséből kitűnik, hogy R3 forrásregiszter és rendeltetési regiszter szerepét egyaránt betölti. Az ábra további része a sínszervezés különböző lehetőségeit mutatja be.

Vezérlőhálózatok és vezérlők

A vezérlőhálózat felépítésmódja a *radiális-* vagy a *sínszervezés* elvét követheti (3.2. ábra). A radiális hálózat az adathálózat minden vezérlőpontjához a vezérlő egy-egy kimenő pontját rendeli, míg sínhálózat esetében az adathálózat összes elemei vezérlőpontjaikkal a vezérlőből kiinduló közös sínhez csatlakoznak.

A radiális hálózat a pikoutasításokat explicit alakban, a sínszervezésű vezérlőhálózat kódolt formában továbbítja a vezérlőtől az adathálózatához. A radiális hálózat az adathálózat elemeit közvetlenül identifikálja; sínszervezés esetén az adathálózat elemeinek külön azonosító áramkörrel is kell rendelkezniük.

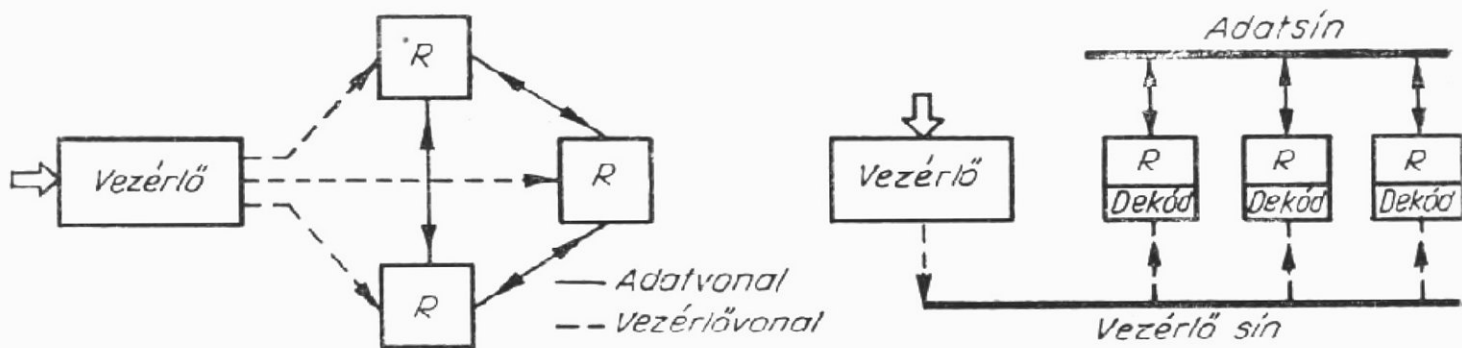
A vezérlő: sorrendi logikai hálózat. Az általánosságban alkalmazott szinkron sorrendi hálózatra tekintve, a 3.3. ábra a vezérlő leegyszerűsített (a visszacsatoló hur-



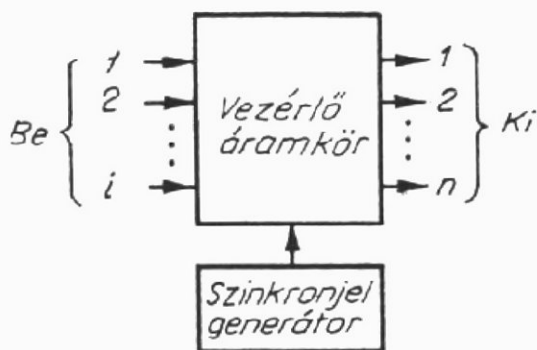
3.1. ábra.

Regiszter (R)—operátor (OP)—vezérlő szintű adathálózatok

(a) ponttól-ponthoz szervezésű hálózat átmenő operátorral (AOP) és aritmetikai-logikai operátorral (ALOP); (b) sínszervezésű hálózat átmenő operátorral; (c...f) aritmetikai-logikai operátor elhelyezésének lehetőségei sínszervezésű adathálózatban



3.2. ábra.
Vezérlőhálózat szervezésmódjai
(a) radiális hálózat; (b) sínhálózat



3.3. ábra.
A vezérlő leegyszerűsített vázlata

kot implicál) sémáját szemlélteti. A vezérlő minden a gép architektúrájára értelmezett állapothoz és bemenő értékhez meghatározott kimenő értéket, bináris vezérlővektort vagy vektorok egymást követő sorát rendel. A vezérlővektorok a szinkron generátor által időzítve jelennek meg a kimeneten.

A mindenkor bemenet különböző komponensekből tevődik össze: az utasítás műveleti kódjából, belső állapotjelzők jeleiből, szolgálatkérő jelekből stb. A vezérlő a különböző forrásokból származó jeleket együttesen is értelmezheti (pl. feltételes utasítást és állapotjelzőket), de e jeleknek egymást kizáró értelme is lehet (pl. a megszakításkérés a folyamatban levő utasítás lezártáig figyelmen kívül marad).

A vezérlővektor formátuma és információtartalma gépenként változik. Reprezentálhat egy vagy több pikoutasítást, vagy mikroutasítást is. A vezérlővektor komponenseinek (a vezérlő kimenő pontjainak) száma általánosságban az adathálózat vezérlőpontjainak számától, továbbá a kódolásmódtól függ.

A számos pikoutasítást tartalmazó vezérlővektor az ún. *horizontális*, a néhány pikoutasítást képviselő vezérlővektor a *vertikális* vezérlésmód jellemzője.

A vezérlővektorok, ill. a vektorokból álló algoritmusok a vezérlőben tárolódnak. A tárolás módjától függően, *kötött- és tároltprogramú* (mikroprogramozott) vezérlő különböztethető meg. Az előbbinél az algoritmusrendszer a szerkezet struktúrájában őrződik (huzalozott programozás), míg az utóbbinál a tárolás céljára külön ún. vezérlőmemória szolgál. A kötöttprogramú vezérlő algoritmusrendszere a szerkezet átépítése nélkül nem módosítható, míg a tároltprogramú vezérlő esetében ez pusztán a tár cseréje, esetleg a tár tartalmának átírása útján végezhető.

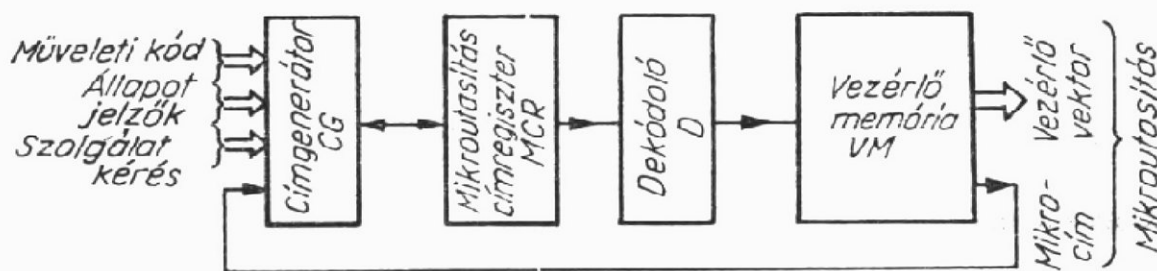
A kétféle vezérlő eltér az algoritmusrendszer strukturáltságának tekintetében is. A kötöttprogramú vezérlő strukturáltságáról tulajdonképpen nem is beszélhetünk; algoritmusrendszerét ezért a logikai hálózat szintjén kell leírni. A tároltprogramú vezérlő ezzel szemben lehetővé teszi, hogy az algoritmusokat strukturált mikroprogramok az algoritmusrendszert mikroprogramrendszer alakjában építsék fel.

A kötöttprogramú vezérlő szerkezeti szempontból is már csak a logikai szinten strukturálható, ROV szinten tovább nem bontható. Ugyanakkor a tároltprogramú vagy általánosabb nevén mikroprogramozott vezérlő maga is regiszterekből, operátorokból és vezérlőből (szinkrongenerátor) állván, mintegy mikro ROV struktúrát képez.

Mikroprogramozott vezérlők

A mikroprogramrendszer kötetlen elérési sorrendű tárban (vezérlőmemória) tárolható, melyből az egyes rutinok szükség szerint hívhatók le és futtathatók. A vezérlővektor (explicit vagy kódolt) a mikroutasítás része. Valamely mikroművelet végrehajtása lényegében a megfelelő mikroutasítás kiolvasásából és az általa tartalmazott vezérlővektorok a vezérlőhálózatra való továbbításából áll. A mikroutasítás a vezérlővektoron kívül többnyire egyéb adatokat is tartalmaz.

A mikroprogramozott vezérlő (3.4. ábra) funkcionális egységei általában a kö-



3.4. ábra.
Mikroprogramozott vezérlő felépítésmódja

tetlen elérési sorrendű vezérlőmemória (VM), a mikroutasítás címregiszter (MCR), ill. ennek dekódolója (D), valamint a címgenerátor (CG).

A vezérlő működésével kapcsolatban mindenekelőtt két kérdés merül fel: hogyan áll elő egy-egy meghatározott mikrorutin kezdő címe, és mi módon történik a címregiszter tartalmának változtatása a kiválasztott mikrorutin mentén.

A kezdő címre utaló adatot a vezérlő a bemenetén át nyeri. Programvégrehajtási alapállapotban pl. ez az adat a gép adathálózatából érkezik; az utasításle hívási ciklus végén, az utasítás műveleti kódja (az előforduló kiegészítő adatokkal) és a gép rendszerétől függő — esetleges — állapotjelzők együtteseként jelenik meg a vezérlő bemenetén. Az utasítások műveleti zónájának kiterjedése még kis kapacitású vezérlőmemória esetén sem elegendő általában a teljes címtartomány átfogására, elegendő azonban VM kezdőszektorának címzésére. Ebben a kezdő szektorban helyezik el a mikroprogramrendszer összes rutinjainak kezdőcím-listáját. Valamely utasítás végrehajtása, rutinja effektív kezdő címének e listából való felkeresésével és MCR-be való átvitelével indul.

A kezdő címek listájának tárolására külön címgeneráló memóriát vagy vele egyenértékű programozható gyors logikai egységet is alkalmaznak.

Szolgálatkérő (pl. megszakításkérő) jel rutinjának felkeresése a fent említett módszerek egyikével történhet, ha a szolgálatkérő jelhez meghatározott kódot — mint a kezdő címre utaló adatot — rendelnek. A szolgálatkérő jel kezdő címe különösen egyszerűen és gyorsan generálható programozható gyors logikai egység útján.

Az MCR-tartalom automatikus változtatásának egy ismert megoldása: minden mikroutasítás tartalmazza a soron következő címét is, mely a vezérlővektorral

együtt olvasódik ki VM-ből, majd a 3.4. ábrán látható módon, azaz a CG-n át, új mikrocímként MCR-re kerül.

A mikroutasítások hosszának, VM tárolóterületének csökkentése érdekében a mikroutasítás, teljes cím helyett, csonka címet tartalmazhat. A mikrorutinok túlnyomó többsége ugyanis korlátozott terjedelmű; címtartományuk VM teljes címtartományának csekély hányada. Így, minden mikrocím egy, a kiválasztott rutinra jellemző bázisértékből és az ezt kiegészítő eltolásból áll. A bázisértéket a kezdő cím adja meg, az eltolást pedig — csonka címként — a mikroutasítás nyújtja. Az eltolási címtartomány határainak átlépésére különleges mikroutasítás szolgál.

Nem tartalmaznak utasításcímet (csonka címet sem) azoknak a vezérlőknek mikroutasításai, amelyek mikroprogram-számlálót alkalmaznak. A mikroprogram-számláló MCR-nek olyan operatív változata, mely külső vezérlőjel hatására párhuzamos bemenetén át beírt tartalmát inkrementálja, ill. dekrementálja. Ha több növekményérték (± 1 , ± 2 stb.) közötti választás is lehetséges, kisebb programhurkok egyszerű módon képezhetők. A mikroprogram-számláló alkalmazása áramkörileg bonyolítja a vezérlőt, viszont ugyanakkor VM-terület megtakarítással jár. A mikroprogramrendszeren belüli feltételes vagy feltétel nélküli elágazásokhoz, az elágazási címek megadásához, programszámláló alkalmazása esetén is, különleges mikroutasítások szolgálnak.

A mikroutasítás gazdaságos felépítésmódjának követelménye nem áll mindig összhangban az alkalmazott vezérlő-, ill. adathálózat természetével. Ezért a vezérlő kimenete és a vezérlőhálózat közé célszerű lehet dekódoló egységet iktatni. Dekódoló szükséges pl., ha radiális vezérlőhálózat alkalmazása ellenére a memóriaterület megtakarítása érdekében a vezérlőinformációt nem explicit, hanem kódolt alakban tárolják VM-ben. A kódolás alapjául az a lehetőség szolgál, hogy az adathálózat egymást kölcsönösen kizáró vezérlőpontjai egy-egy csoportba összefoghatók. Az egy csoportba tartozó bármely pont kijelölése bináris kóddal és bináris dekódoló segítségével végezhető.

A vezérlővektor időszakos tárolására általában a vezérlőmemóriát követő puffer-regisztert alkalmaznak.

A vezérlőmemória — rendszerét illetően — fixtár, átírható fixtár vagy olvas/ír tár. Az előző két tártípust statikus, az utóbbit dinamikus programozású vezérlőmemóriának nevezik. A fixtár rögzített mikroprogramrendszert, rögzített architektúrát jelent. Az átírható fixtár megengedi az architektúra esetenkénti módosítását, míg az olvas/ír tár a gép felhasználója számára is lehetővé teszi a mikroutasításszintű programozást.

Adat- és vezérlővonalak

Az adathálózat vonalainak szerkezete az adattovábbítás és processzálás módjától függ. Ismeretesen soros és párhuzamos számítógép különböztethető meg. Tekintettel a soros gépek jelenleg elhanyagolható jelentőségére, leírásunk folyamán mindig párhuzamos gépet tartunk szem előtt. Elvileg a párhuzamos gépben az adattovábbítás a regiszterek szélességének megfelelő számú vezetékből álló vonalakon át történik. Aligha van azonban olyan gép, mely — ebben az értelemben — a párhuzamosság kritériumának kizárólagosan tenne eleget, már csak azért is, mivel gyakran a regiszterek is különböző szélességűek. Az adathálózat egyes szakaszain soros, ill. soros-paralel átvitel móddal a párhuzamos gépeknél is találkozhatunk; az egyes adatvonalak vezeték száma egy gépen belül is változó lehet.

Bizonyos adatvonalak egyirányúak, míg más vonalakon mindkét irányban továbbítható adat. Az irány fogalmát általánosan a vonalhoz kötik, bár nem a vonal, hanem a hálózat attribútuma.

Az adatvonal belső szerkezetére bizonyos mértékben a hálózat szervezőmódja is kihat.

A legkisebb ROV-szintű információegység, amelyet a vezérlő az adathálózathoz továbbít: egy pikoparancs. A pikoparancs több elemből tevődik össze. Tartalmazza a végrehajtandó művelet azonosítóját, valamint annak a regiszternek vagy operátornak az azonosítóját, melynek a műveletet végre kell hajtania. Bizonyos kiegészítő elemeket is magában foglalhat, pl. a feladat lebonyolításának időzítésére vonatkozó információt.

Mint ahogy a radiális vezérlőhálózatban minden pikoparancs továbbítására külön, a vezérlőt az érintett regiszter vagy operátor megfelelő vezérlőpontjával közvetlenül összekötő vonal szolgál, a radiális vezérlőhálózat minden vonala egy és csakis egyféle pikoparancsot képvisel; e vonal jelének hatására mindig ugyanaz a pikoművelet hajthatódik végre. Ezért is beszélünk a radiális hálózattal kapcsolatban a pikoparancs (a vezérlővektor) explicit voltáról. A tiszta radiális vezérlőhálózat vonalai egyvezetékek.

A sínszervezésű vezérlőhálózat — szélső esetben — csupán egyetlen vezérlővonalból (sín) áll, s ennek kell eljuttatnia az adathálózat összes vezérlőpontjára a pikoparancsokat. A sín nem egy, hanem az összes pikoparancshoz „hozzátartozik”, azok egyikét sem indentifikálja, ezzel szemben képesnek kell lennie az összes lehetséges pikoparancs információelemeinek átvitelére. A sín műveleti vonalat, címvonalat és szolgálati vonalat kell hogy tartalmazzon. Az utóbbi kettő segítségével a vezérlő a pikoparancs lényegi részének (művelet) a kijelölt vezérlőpontra való átvitelét adminisztrálja. A címvonal vezetékszámát az adathálózat címtartományának megfelelő bináris címkód, a műveleti vonal vezetékszámát a művelet kódformátuma határozza meg. A sínrendszeren át az adathálózat egészéhez eljuttatott, de csupán egyetlen elemi vezérlőpontnak szóló pikoparancs dekódoló segítségével választódik ki.

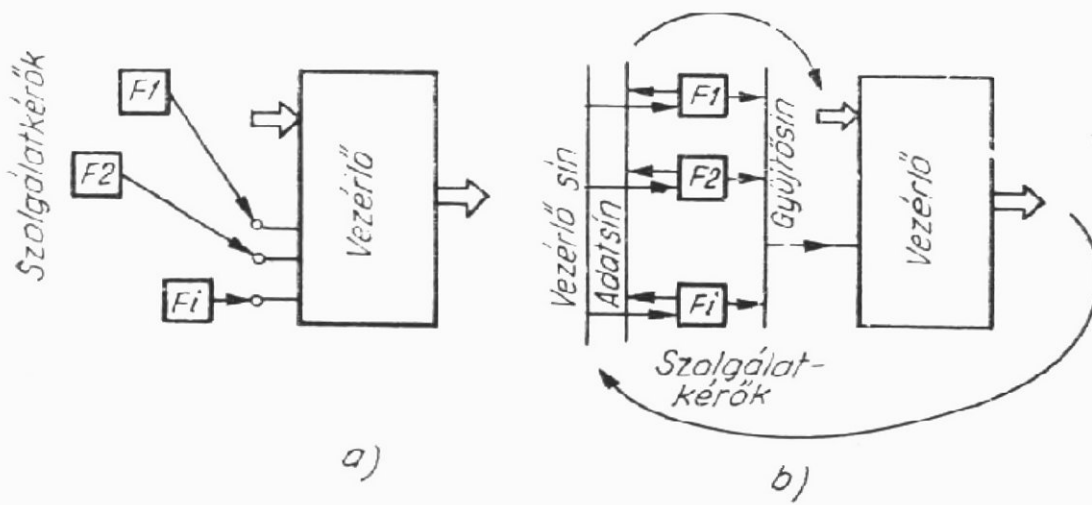
Az egyetlen pikoparancsot reprezentáló vezérlővektor a vezérlés hatékonysága szempontjából rendkívül szegényes. A határfok növelése érdekében multiplikatív vezérlőszint is alkalmaznak, mellyel egyidőben több pikoparancs (pl. egy a forrás, egy a rendeltetési regiszter számára) vihető át.

Szolgáltatkozó vonalak és hálózataik

A szolgáltatkozó jelekhez megszakítástkérő jelek, cikluselvétel kezdeményező jelek, az esetleges autonóm csatorna kezdeményező jelei stb. tartoznak. E jelek mindegyike a vezérlő bemenetére irányul, s mindegyikük meghatározott gépi akciót kezdeményez. A szolgáltatkozó jelek begyűjtő hálózatainak szervezése a vezérlőhálózatokénak mintegy fordított képét mutatja. A radiális hálózat fordítottja a radiális gyűjtő rendszer, melyben minden jelforrást a vezérlőbe irányuló közvetlen vonala azonosít (3.5a ábra). A sínszervezésű vezérlőhálózat megfelelője a sínrendszerű gyűjtőhálózat, melynél számos jelforrás szolgáltatkozó jele egyazon vonalra fut be, s a jelforrás azonosításáról, a forrás címének az adathálózaton át történő bekérésével a processzorvezérlő gondoskodik (3.5b ábra).

A gyakorlatban a számítógépek hálózataiban a fent ismertetett szervezési elvek ritkán érvényesülnek teljes tisztaságukban. A vezérlésben explicit forma keveredik kódolt formával, radiális hálózat sínszervezésével. Az adathálózatban is gyakran együtt realizálódik a ponttól-ponthoz és a sínszervezés elve, méghozzá úgy, hogy ponttól-ponthoz vonalak számos kisebb nagyobb hatósugarú sínnel vegyesen fordulnak elő.

Leírásunk elsősorban a ROV hálózatok felépítésmódjára, nem pedig a folyamatokra irányult. Ezért, az időbeli viszonyok áttekintését mellőztük. Célunk szempontjából



3.5. ábra.
Szolgáltatókérő vonalak hálózatai
(a) radiális; (b) sínrendszerű

ezen a helyen kielégítő, ha azzal a közelítő feltételezéssel élünk, hogy a vezérlővektor minden eleme egyidejűleg hat. A valóságban az egyetlen vezérlési ciklushoz tartozó pikoműveletekkel kapcsolatban általában sorrendiséget kell feltételezni; az egyik művelet végrehajtási idejének a másikéhoz viszonyított eltolása szükséges, tekintettel az áramkörü elemek véges működési sebességére.

A vezérlővektort az adathálózat végül mindig explicit alakban kapja meg. Az explicit és a kódolt forma — mint megkülönböztetés — a vezérlő kimenetére vonatkozik. De mit nevezünk a vezérlő kimenetének? A vezérlőt az adathálózattól esetenként annál nehezebb elhatárolni, mivel — mint ez gyakran megtörténik — nem zárt blokként, hanem az adathálózat mentén eloszló áramkörü egységek alakjában realizálódik. Elhatárolást azon az alapon végezhetünk, ha a vezérlőhálózat vonalain fekvő azon áramköröket, melyek az adathálózat egy és csakis egy elemét szolgálják az adathálózathoz, azokat pedig, amelyek egynél több elemet szolgálnak, a vezérlőrendszerhez tartozónak gondoljuk.

A ROV hálózatban regisztereken, operátorokon és vezérlőkön kívül egyéb hálózati elemek is előfordulhatnak, így: paritásképzők és paritásellenőrzők, soros-parallel, ill. parallel-soros átalakítók, multiplexerek stb. Ezek azonban a ROV-szintű leírás szempontjából másodlagos jelentőségűek.

3.1.2. Memóriák

Memóriának a processzorral közvetlen operatív kapcsolatban álló tárákat nevezzük. A „memória” tehát a funkcióra utaló megnevezés. A memóriaként alkalmazott tárák, jellemzően, azonos tulajdonságú regiszterekből álló homogén regiszterhálózatok. A regiszterek a külvilággal közös csatornán (csatornákon), a memória-vezérlő irányítása alatt kommunikálnak. Memóriaként, legáltalánosabban, ferrit-, illetve félvezetőtípusú tárákat alkalmaznak.

A főmemória

A gép központi szerepkörű memóriája; speciális esetektől eltekintve kötetlen elérési sorrendű (RAM), változó tartalmú (olvas/ír) tár. Kommunikációs csatornája adatvonalat, címvevonalat és vezérlővonalat ölel fel. A memória bejáratánál (port) címregiszter és adatregiszter található.

Elemi műveletei: az írás és az olvasás. Követelmény, hogy a kiválasztott rekesz (regiszter) előző tartalma írás kapcsán törlődjék, az olvasás azonban a tartalmat ne semmisítse meg. E követelmény a másoló írású, nem roncsoló olvasású félvezető tár esetében minden további nélkül teljesül. A ferritgyűrűs táruk ezzel szemben roncsoló olvasású VAGY írású rekeszekkel rendelkeznek, ezért náluk mind írás, mind olvasás esetében teljes (írás és olvasás fázisból álló) memóriaciklust kell lefuttatni. Az áramkörök megegyező működési sebességét feltételezve, félvezető tárral tehát kétszer akkora feldolgozási sebesség érhető el, mint ferrittárral, ugyanakkor vezérlési struktúrája is egyszerűbb.

A ferrittár átírási képességének növelése érdekében egyes főmemóriák „olvasás—adatmódosítás—írás” műveletet is végezhetnek, mely művelet időmegtakarítást ad, ha utasításvégrehajtás során a processzor a memóriából olyan adatot hív le, melyre felhasználása után már nincs szükség, s a processzor eredménye a már kiválasztott, üres rekeszbe írható.

A memória műveletidejétől meg kell különböztetni elérési idejét. Műveletidő: a teljes időtartam, mely a művelet kezdeményezésének időpontjától a memória-áramkörök nyugalmi állapotának helyreálltáig telik el, s mely után új művelet kezdhető. Ezzel szemben elérési időn a művelet kezdeményezésének pillanatától a kiválasztott rekesz és az adatregiszter összekapcsolódásáig eltelt időtartamot nevezik. Az elérési idő végén a kiolvasott adat az adatregiszterben már rendelkezésre áll, ill. az adat átírása az adatregiszterből már kezdetét veszi.

A rekeszt, a memória címvonalára adott cím s a vezérlővonalára adott parancs alapján, a memóriavezérlő választja ki. Ugyanő végzi a kiválasztott rekesz tartalmának az adatregiszterbe vagy az adatregiszter tartalmának a rekeszbe való átmásolását. Kiválasztás és átírás, félvezető és ferritgyűrűs tár esetében egyaránt egymással rendkívül szorosan összefonódó, több elemi fázisból álló folyamat.

A főmemória környezetével szinkron vagy aszinkron kapcsolatban állhat. A szinkron kapcsolatú főmemóriához forduló környezetnek „ismernie” kell a memória-műveletek időbeli tagozódását; a memória és a környezet folyamatainak ui. egymással szinkronban kell futniuk. E feltétel maradék nélkül akkor teljesül, ha a főmemória vezérlőjét a környezetből érkező, a memória tulajdonságait figyelembe vevő órajelek irányítják. Fordítva, az órajel ciklusidejét a memória fizikai folyamatainak lefutási időihez kell igazítani. Mivel a memória, általában, a leglassúbb áramköri elem, a szinkron kapcsolat a környezeti folyamatok sebességét többnyire az áramkörök által megengedett lehetőségek alatt korlátozza.

Az aszinkron kapcsolatú főmemória belső folyamatai a környezettől függetlenül, önállóan zajlanak. Memóriaműveletet természetesen ez esetben is csak a környezet kezdeményezhet. A kezdeményezett művelet végrehajtásáról a memória a kezdeményező szervnek visszajelzést ad. E jelzés vételéig terjedő időn át a kezdeményező szerv saját működését felfüggeszti, egyébként azonban önmaga áramköri sebességével működhet. Megvan annak is a lehetősége, hogy a kezdeményező szerv (pl. processzor) a memóriával szimultán dolgozzék. Az aszinkron kapcsolat előnye továbbá, hogy a főmemória a számítógép egyéb áramköreinek érintése nélkül cserélhető, a főmemórián belül különböző sebességű modulok alkalmazhatók, továbbá, hogy vele olyan üzemmódok is létesíthetők, amelyeket a szinkron kapcsolat kizár.

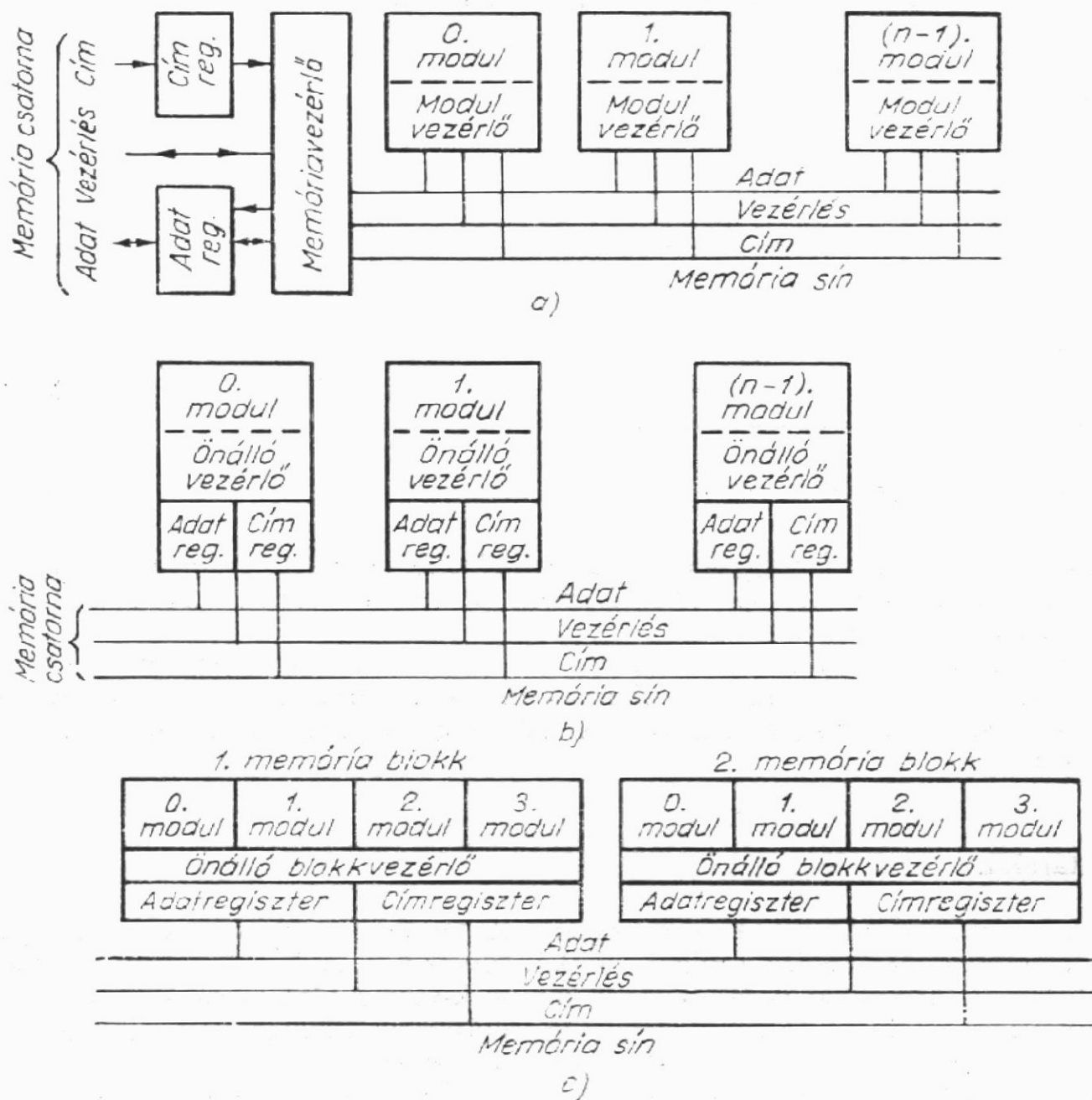
A főmemória fontos jellemzői még: bitben mért rekeszmérete, valamint tagozódásának módja. Építőegysége a modul. Technológiai modul (t-modul) és funkcionális modul (f-modul) különböztethető meg. A t-modul bitkapacitását általában a modul gyártási eljárása szabja meg. Az f-modul (legtöbbször 4 vagy 8 K; újabban 16 vagy 32 K; ritkábban 1 vagy 2 K) nagyságát elsősorban alkalmazástechnikai szempontok határozzák meg. A ferritgyűrűs memória f-modulja szokás szerint a t-modullal egybeesik. A félvezető memória t-modulja viszont f-moduljánál lé-

nyegesen kisebb kapacitású. Az f-modul mindig szó vagy byte szervezésű, ezzel szemben a félvezető t-modul, általában bitszervezésű; a szó (byte) szervezésű f-modul számos t-modulból épül fel.

Az f-modul rendszerint síkmátrix címzésrendszerű; bármely rekesz modulon belüli címét két bináris vektor együttese adja meg.

A főmemória kapacitástartományát a számítógép névtartománya definiálja. A kapacitástartományon belül a főmemória, a szükséglet mértékében, a rendelkezésre álló f-moduloknak megfelelő inkrementummal építhető ki.

A főmemória lehet egybefüggő vagy részenként független. Az *egybefüggő memória* f-moduljai vezérlés szempontjából nem függetlenek; a memóriavezérlő egy része külön egységként, valamennyi modul számára közös (3.6a ábra). A *részenként független memória* önálló vezérlőjű modulokra (3.6b ábra), vagy közös vezérlővel rendelkező modulcsoportokra (blokkokra) oszlik fel (3.6c ábra). A modul (blokk) függetlensége abban áll, hogy a modul (blokk) a memóriaműveletek végrehajtásához szükséges minden elemmel rendelkezik.



3.6. ábra.

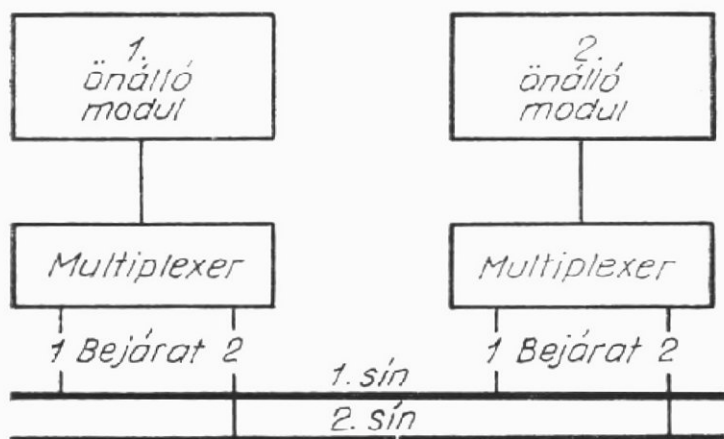
A főmemória különböző szervezőmódjai

(a) egybefüggő memória közös vezérlővel; (b) részenként független memória, modulonként önálló vezérlővel; (c) részenként független memória, blokkonként önálló vezérlővel

Az egybefüggő memória egyidejűleg csupán egyetlen memóriaműveletet végezhet. A részenként független memória ezzel szemben szimultán műveleteket is megenged. Kétféle szimultán üzemmód ismeretes: a láncolódás és az átfedés.

A *láncolódásos* üzemmód feltételeként a főmemória két önálló egységre oszlik, melyek közül az első a páros című, a második a páratlan című rekeszeket tartalmazza. Ha — amint az a főmemória üzemében általában szokásos — az egymást követő memóriáhozfordulások váltakozva páros/páratlan cím szerint történnek, minden egyes memóriaművelet olyan időpontban kezdhető, amelyben az őt megelőző még folyamatban van. A láncolódásos üzemmódban dolgozó, részenként független memória különösen ferritgyűrűs tár esetében, lényegesen nagyobb átbocsátóképességű, mint az egybefüggő memóriarendszer.

A részenként független memória *átfedéses* üzemmódjában az egyes önálló memóriaelemek más-más számítógépszervvel tartanak fenn egyidejű kapcsolatot. Amennyiben csupán egyetlen memóriasín áll rendelkezésre, a működésbeli függetlenség csak részleges lehet. Valóságos szimultaneitás részenként független, két-vagy több-bejárátú memóriarendszerrel valósítható meg. (3.7. ábra).



3.7. ábra.

Kétbejárátú, részenként független memória

Az olvas/ír táruk közül a ferritgyűrűs *tartalomőrző*, a félvezető *tartalomvesztő*. A félvezető memóriához, emiatt, minden esetben háttéráramforrást (akkumulátort vagy galvánelemet) kell alkalmazni.

A főmemórián belül (állandó programok és adatok tárolására) esetenként rögzített tartalmú (ROM) modulokat is alkalmaznak. A fixtár előnye: nagyobb megbízhatósága és általában nagyobb működési sebessége, viszonyítva a változtatható tartalmú tárukhoz.

Közbenső tárolók

A közbenső tároló (k-tároló) a számítógép két különböző működési sebességű egységének időbeli illesztésére szolgál. Ritkábban két eltérő szélességű vonalat illesztő közbenső tárolókkal is találkozhatunk. A k-tároló alkalmazásával elkerülhető, hogy valamely lassú működésű egység, akár csak ideiglenesen is, a gyors működésű egységekre kényszerítse saját működési sebességét.

A k-tároló szervezőmódját az általa illesztett egységek kapcsolatának mibenléte határozza meg. A szóátvitelű k-tároló pl. egyetlen regiszterből áll, míg a blokkátvitelű a blokkban szereplő szavak (karakterek) számának megfelelő számú rekeszből épül fel. A blokkátvitelű k-tárolóval mint időbeli illesztőelemmel mind a lassú,

mind a gyors egység a maga ütemében kommunikálhat. Két irányból hívható, pl. úgy, hogy először a gyors hálózat lefoglalja, s feltölti egy adatblokkal. Majd a gyors hálózat és a memória kapcsolata megszűnik, hogy a lassú egység a saját ütemében egymás után hívja le és dolgozza fel az egyes rekeszek tartalmát. A feltöltés és kiürítés fázisa nem minden esetben különül el. Írás- és olvasásműveletek egymás közé ékelődve is előfordulhatnak; a tároló kiürítése a teljes feltöltés előtt már megkezdődhet vagy fordítva is (összefonódó üzemmód).

A k-tárolók feltöltési és kiürítési sorrendje általában kötött. Írás parancs alapján a soron következő üres rekeszt, olvasás parancs alapján a soron következő kiolvasandó rekeszt a tárvezérlő választja ki; a k-tárolóval nem kell címet közölni. A feltöltés és kiürítés egymáshoz viszonyított sorrendjét illetően két lehetőség kínálkozik:

- a kiürítés a feltöltés sorrendjében történik,
- a kiürítés sorrendje a feltöltés sorrendjének fordítottja.

Így azonos sorrendű és fordított sorrendű k-tárolóról beszélhetünk.

Amint általában a kötött elérési sorrendű tárolóknál, a k-tárolónál is a tárvezérlő ellenőrzi a memória foglaltsági és feltöltöttségi állapotát, biztosítva, hogy se adatok egymásra írása, se üres rekeszekből való olvasás ne történhessen.

A k-tárolónak két, egymástól elvileg lényegesen eltérő rendszere ismeretes. Az egyik, tárként, kötetlen elérési sorrendű modult használ, mely a kiválasztó logikához kapcsolódó címszámlálóval (számlálókkal) egészül ki; a másik, recirkulációs léptetőregiszterekből épül fel.

A főmemória és a központi processzor időbeli illesztésére alkalmazott asszociatív k-tárolót *memóriapuffernek* (memory buffer; cache) nevezik. A memóriapuffer az egyszerű k-tárolótól, működésmódját tekintve, lényegesen különbözik. Segítségével a viszonylag lassú főmemóriával rendelkező kisméretű gépek feldolgozási teljesítménye a központi processzor sebességi lehetőségeinek tartományába emelhető.

A memóriapuffer alkalmazásával kiadódó eredő ciklusidő lényegesen rövidebb lehet, mint a főmemória ciklusideje, de a ciklusidő értéke nem kizárólag szerkezeti jellemző; függ a főmemória és a puffer ciklusidőinek értékétől, a puffer kapacitásától, valamint a programok szerkezetétől.

A közbenső tárolók sorában meg kell még említeni az egymástól eltérő szélességű adatvonalak illesztésére szolgáló tárolókat. E tárolók segítségével a soros adat paralel adattá — és fordítva — alakítható. Átmeneti formáik paralel adat időben egymást követő részekre tördelését vagy részekből összetevését végezhetik.

Letároló (stack)

A letároló, szervezésmódjában a k-tárolóval áll rokonságban. Elérési sorrendje kötött; a kiürítés a feltöltéshez viszonyítva fordított sorrendű. Rögzítettperemű (alul nyitott) és mozgóperemű (alul zárt) változata különböztethető meg. (A perem a letárolónak azon helye, melyen át környezetével kommunikál). Az előbbi: irányváltó léptetőregiszterekből, az utóbbi: kötetlen elérési sorrendű tár és irányváltó címszámláló együtteséből épül fel.

A letároló alapműveletei: a betétel és a kivétel; ritkábban előforduló műveletei: az írás és az olvasás. A 3.8. ábra e műveleteket a rögzített-, illetve a mozgóperemű letároló esetére értelmezi, egyúttal a kétféle letároló működési mechanizmusát is bemutatja. A rögzítettperemű letárolónál minden művelet a perem melletti ugyanazon rekeszre irányul. Betétel, ill. kivétel kapcsán az egész memóriatartalom egy rekeszrel lefelé, ill. felfelé mozdul. Mivel a környezet mindig azonos rekeszrel áll kapcsolatban, a műveletek címet nem igényelnek. A mozgóperemű letároló tartalma a memóriaműveletek folyamán a rekeszekhez viszonyítva helyben marad,

Művelet előtt	Olvasás	Kivétel	Írás	Betétel	Rögzített perem
	D ↑	D ↑	E ↓	E ↓	
D	D	C	E	E	
C	C	B	C	D	
B	B	A	B	C	
A	A	—	A	B	
—	—	—	—	A	
—	—	—	—	—	
—	—	—	—	—	

	D ↑	D ↑	E ↓	E ↓	Mozgó perem
—	—	—	—	—	
—	—	—	—	E	
D	D	—	E	D	
C	C	C	C	C	
B	B	B	B	B	
A	A	A	A	A	

3.8. ábra.
Rögzített-, ill. mozgóperemű letároló működési sémái

a peremcím viszont állandóan változik. A peremcímet külön regiszter, a peremmutató őrzi. Eltérő cím tartozik azonban egyrészt az olvasás, az írás és a kivétel, másrészt a betétel műveletéhez. Célszerűen a peremmutató a legfelső, tartalommal rendelkező rekeszre mutat. Betétel előtt a peremmutató tartalmát +1-gyel, kivétel után -1-gyel inkrementálni kell.

A rögzítettperemű letároló alul, a mozgóperemű fölül — elvileg — nyitott. A letároló azonban véges kapacitású. Megszakítójellel jelezheti, ha teljesen feltöltött állapotban betételre, ill. teljesen üres állapotban kivételre kap parancsot.

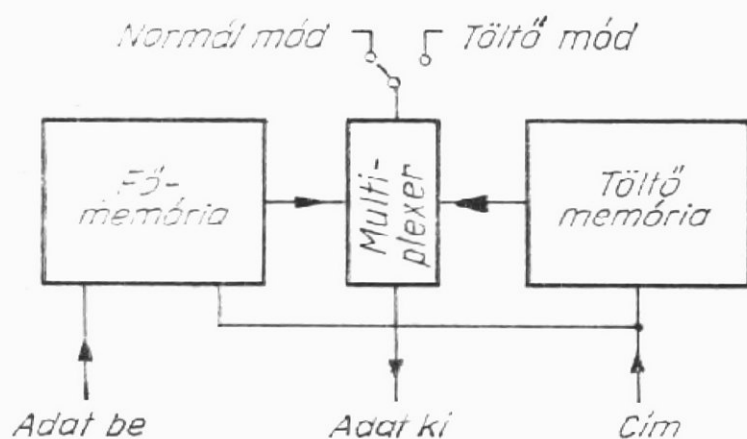
Ismeretesek vegyes szerkezetű letárolók is, melyek néhány rekesz kapacitású rögzítettperemű (gyors) letároló és hozzá csatlakozó nagyobb kapacitású (lassúbb) mozgóperemű letároló együtteséből állnak. A két rész közötti automatikus adatátviteli kapcsolatot a tárvezérlő hozza létre. Az összetett letároló, környezetével a rögzített peremen át kommunikál. A környezet a perem alatti szerkezeti inhomogenitásról nem vesz tudomást.

Bár a letárolónak funkcionálisan egészen más a szerepe, mint a közbenső tárolónak, látható, hogy az ismertetett mindkét típusú letároló fordított sorrendű közbenső memóriaként is alkalmazható. A letároló programmegszakítás és szubrutinhívás alkalmával adatok és címek időszaki tárolására, valamint letároló rendszerű adatstruktúrák kezelésére szolgál. Letárolóként nem csak külön erre a célra rendelt tárat, hanem főmemória területet is használnak (emulálnak).

Töltőmemóriák

A korszerű kisgép töltőprogramját fixtárban, ún. töltőmemóriában tárolják. A töltőmemória csatlakoztatásának módja gépenként változó. Egy megoldást a 3.9. ábra szemléltet. A töltőmemória címtartománya a főmemória címtartományának

kezdő területére esik, azonban a töltőmemória nem része a főmemóriának, hanem azzal paralel helyzetet foglal el. A töltőmemória jelenléte egyedül a főmemória olvasóköreit érinti, amennyiben az olvasóvezetékek nem közvetlenül, hanem multiplexeren át kapcsolódnak a memória adatcsatornájára. A multiplexer másik bemenetére a töltőmemória kimenete lép. A két memória közös cím vonalra csatlakozik. A multiplexer a kezelőpulton elhelyezett kapcsoló segítségével normál-, ill. töltőmódba kapcsolható. Töltőmódban a programszámláló is nullázandó, ami a töltőprogram kezdő címét egyértelműen meghatározza. Töltőmódban a gép az utasításokat a töltőmemóriából olvassa ki, viszont az utasítások hatására betöltött program a főmemóriába kerül. A töltőmemóriában több periférikus egység mint bemeneti készülék töltőprogramja is elhelyezhető, melyek kezdő címei a programszámlálóba a kezelőpult kapcsolóregiszterein át vihetők be. Bármely periférikus készülék töltőprogramja azonban a neki megfelelő, külön, cserélhető töltőmemóriában is tartható.



3.9. ábra.

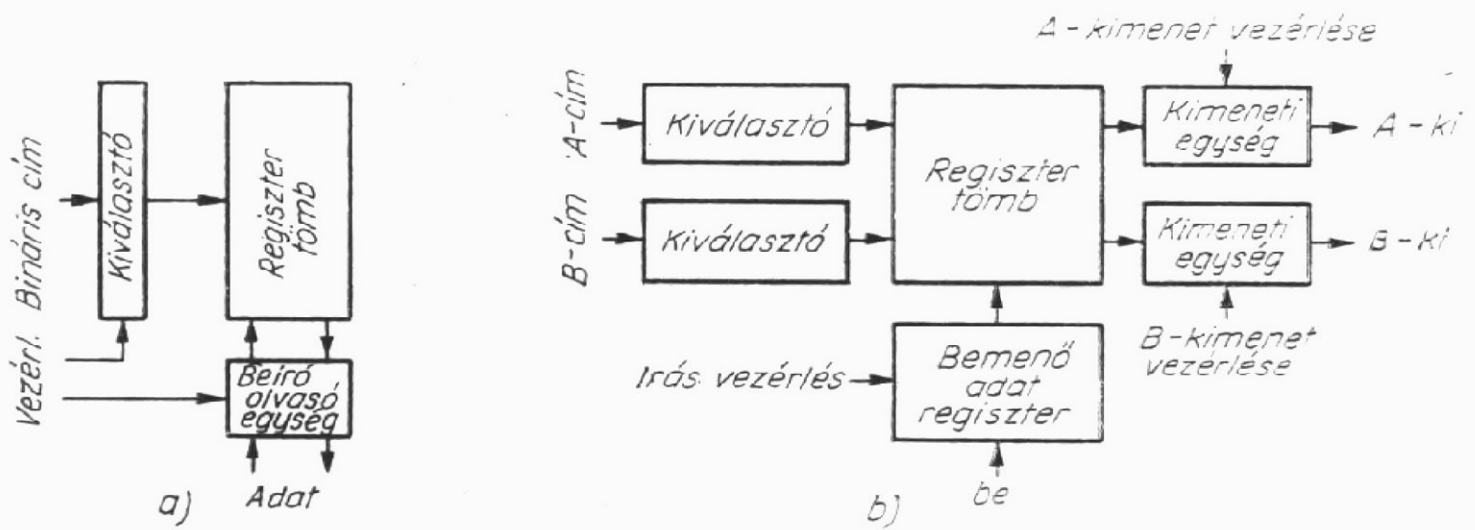
Töltőmemória és főmemória kapcsolata

Regisztertömbök

A főmemória fölé és alá épülő táruk hierarchiája számos kisgépnél a központi processzor belsejéig terjed, regisztertömb (regiszter file) alakjában. A regisztertömb: adatok időszakos tárolására szolgáló, tipikusan 8...32 gyorsregiszterből álló memória, mely a központi processzor vezérlője által individuálisan kezelt regiszterek csoportjától eltérően közös kiválasztó-, beíró- és olvasórendszerrel — tárvezérlővel — rendelkezik.

A 3.10. ábra két regisztertömböt mutat be. Az egyszerűbb (a) változatnál írás és olvasás egymást kizáró művelet. Az összetettebb (b) két különböző rekesz egyidejű olvasását is megengedi; az A címmel kiválasztott rekesz tartalma az A kimeneten, a B-vel kiválasztotté a B kimeneten jelenik meg. Egyidejű írás és olvasás is lehetséges, mégpedig úgy, hogy az A címmel a beírandó, B címmel a kiolvasandó adat helye jelölhető ki.

A regisztertömb homogén szerkezetű (egyforma regiszterek). Az egyes regiszterek gépen belüli funkciói azonosak, azonban egymástól eltérők is lehetnek. A regiszterek processzoron belüli feladatát a processzorvezérlő szabja meg. A processzorvezérlő által kiosztható szerepköröket természetesen a processzor felépítésmódja korlátozza. A regisztertömböt letároló szervezésmódban is alkalmazzák (letároló aritmetika).



3.10. ábra.

Regisztertömbök

(a) egyszerű regisztertömb; (b) kettős szimultán elérésű regisztertömb

Gyors kisegítőmemóriák (gyk-memóriák)

E memóriatípus a kiskapacitású diszk kiváltására (ahol ez utóbbiak sebessége nem kielégítő) szolgál. A táruk hierarchiájában, a csökkenő sebesség irányában, közvetlenül a főmemória mögötti szinten helyezkedhet el. A gyakorlatban a gyk-memóriát MOS technológiájú, dinamikus léptetőregiszterekből szervezett, kötetlen elérési sorrendű recirkulációs tár alapjában valósítják meg. A gyk-memória egyrészt működés-módja, másrészt adattárolásának és elérésének szervezés-módja tekintetében igen emlékeztet a diszkre. Míg azonban a diszkben az adat helyzete az információhordozóhoz viszonyítva rögzített, és az információhordozó mozog, addig a gyk-memóriában az információhordozó mozdulatlan, s benne a tárolt adat kering.

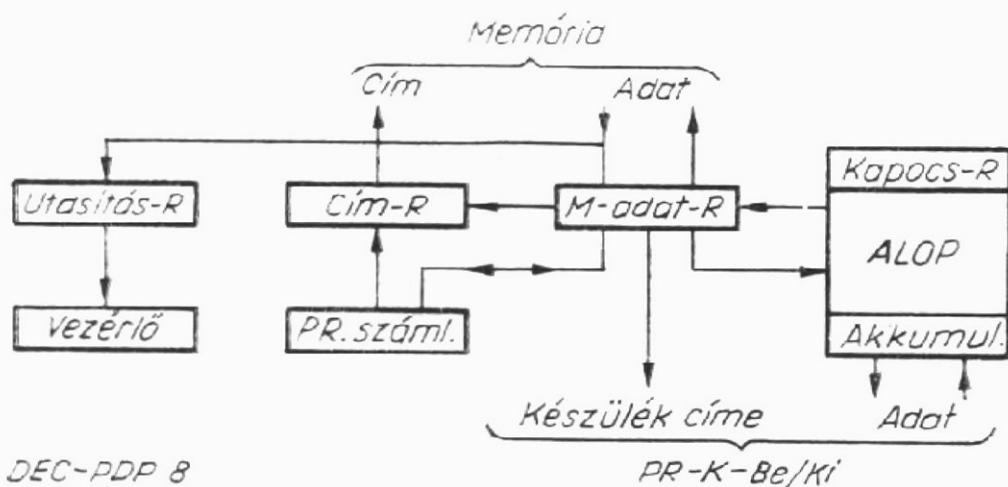
A vezérlők tárgyalása során már érintettük a mikroprogram tárolására szolgáló vezérlőmemóriát; ugyanott egy más szerepkörű, címgeneráló memória is felbukkant. A kisgéprendszerekben számos egyéb szerepkörű memória is előfordulhat, mint pl. kódátalakítók, karaktergenerátorok stb. Az adott keretek között ezek részletezésére nem térhetünk ki.

3.1.3. A központi processzor

A kisgépek processzorai, regisztereik és operátoraik száma és neve, hálózataik felépítésmódja, a környezettel való kapcsolataik szervezés-módja, a végrehajtási szint, valamint technológiai kivitelük tekintetében különbözhetnek egymástól. A processzorok szerkezeti sajátosságait (bár nem kizárólagosan) a 3.11., ... 3. 19. ábrák alapján elemezzük, melyek mindegyike valamely (esetleg több) szempontból karakterisztikus processzort mutat be. Az áttekinthetőség kedvéért az ábrákon lényegében csak a regiszter—operátor hálózatok szerepelnek; a vezérlők specifikumaira, a vezérlőhálózatok szervezésének részleteire, a felmerülő szükség szerint térünk ki.

Regiszterkészletek

A minimum regiszterkészletű processzor két címrendeltetésű regiszterrel (címgiszter; programszámláló), két munkaregiszterrel (adatregiszter; akkumulátor), egy utasításregiszterrel és (1-bites) kapocsregiszterrel (link) rendelkezik (3.11. ábra).



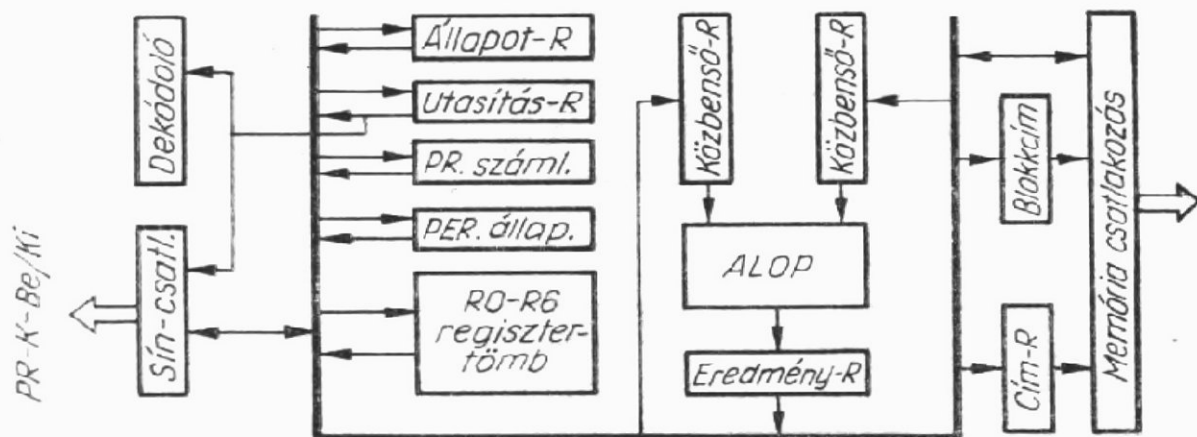
DEC-PDP 8

3.11. ábra.
Ponttól-ponthoz szervezésű processzor adathálózata

Takarékossági szempontból a processzor osztozhat a memóriával az adatregiszteren és a címregiszteren (PDP 8), vagy a processzor a memória egyes rekeszeit saját munkaregisztereként is használhatja (CII 10010).

A nagyobb teljesítményű processzorok kettőnél több munkaregisztert, különböző címképző regisztereket (pl. index- és bázisregiszter), címmutatókat, számos állapotra kiterjedő (egy vagy több) állapotregisztert, valamint időszakos tárolásra szolgáló regisztereket tartalmazhatnak. A regiszterek feladatköre lehet általában rögzített; gyakran pedig csupán néhány regiszter kötött feladatkörű, s a többi ún. általános regiszter. Az általános regisztereknek — a vezérlő által emulált architektúra keretében — a programozó választásának megfelelően többféle szerepköre is lehet.

Elterjedt az operátor és egyedi szerepkörű regiszterek mellett regisztertömböt alkalmazó processzorforma (3.12., 3.14., 3.15., 3.16., 3.17., 3.18., 3.19. ábra).

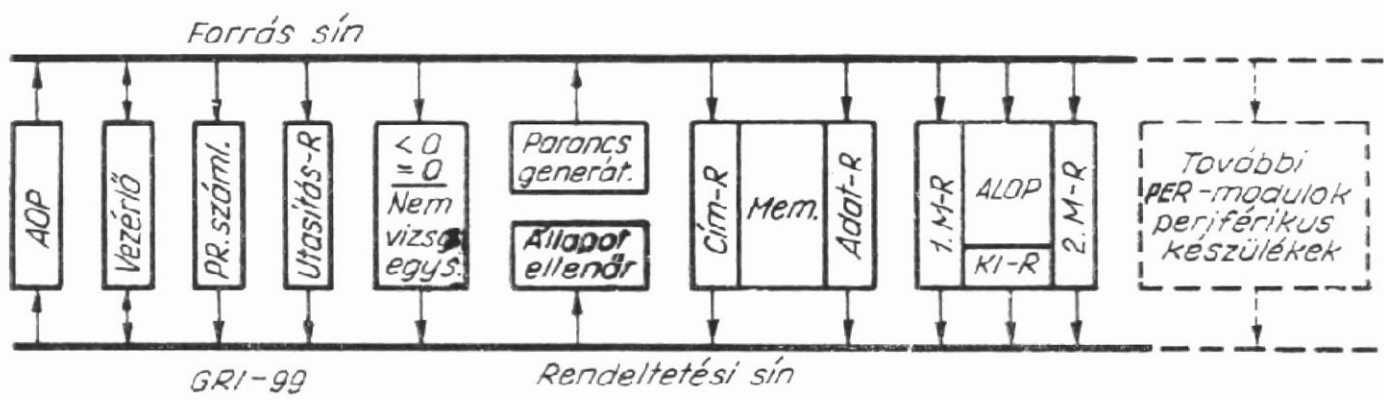


MBM-K 202

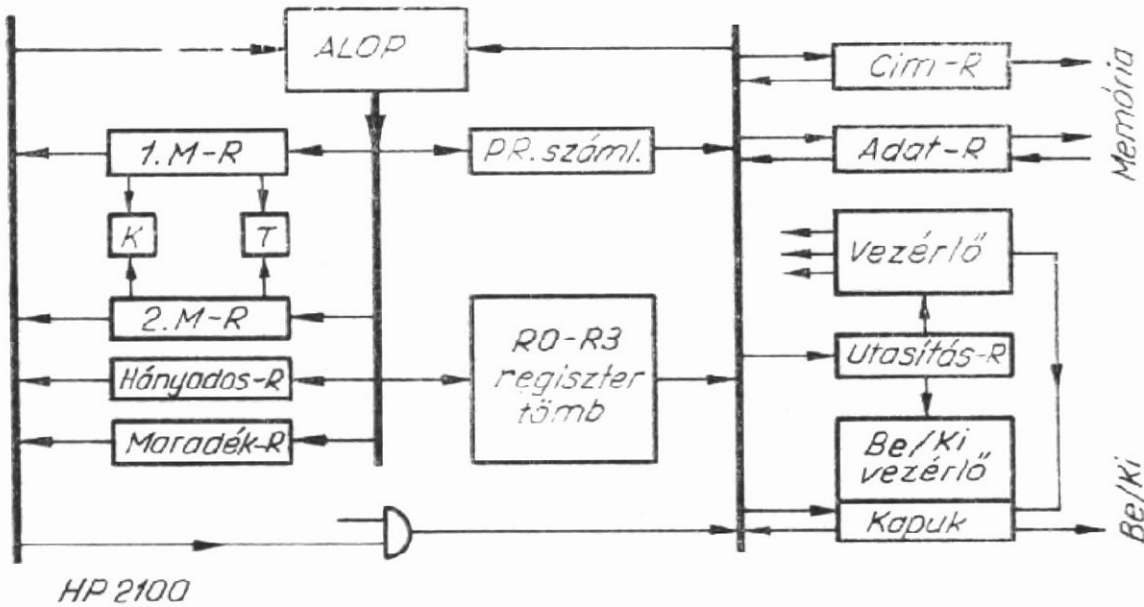
3.12. ábra.
Egysín-rendszerű processzor

A regisztertömb bizonyos regisztereinek a vezérlő egyedi feladatkört szab meg, más részük általános regiszterként áll a programozó rendelkezésére. Processzoron belüli letároló céljára is alkalmaznak regisztertömböt (MICRODATA 32/S), de a processzor rögzített peremű speciális letárolóval is rendelkezhet (NS IMP 16C).

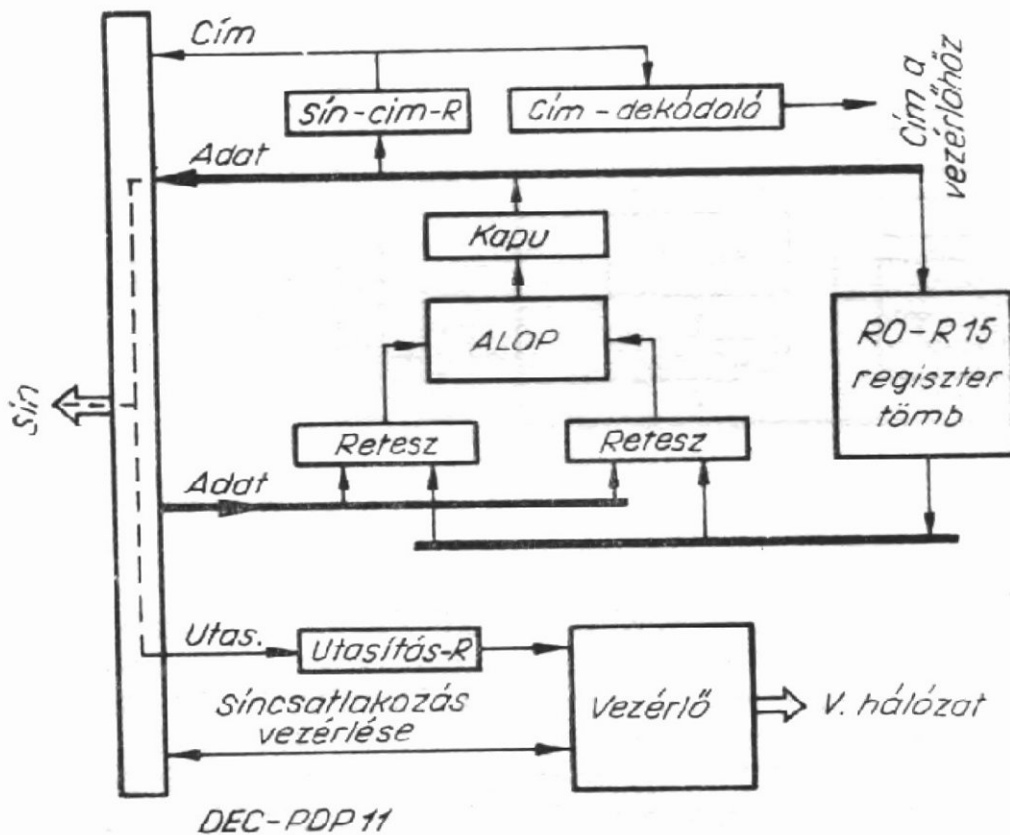
Egyes kisgépekben (SPC 16, PDP 11/45) a gépi kontextust tartalmazó regisztereket „állapottömb” fogja össze. A processzornak egynél több állapotömbje van, ami



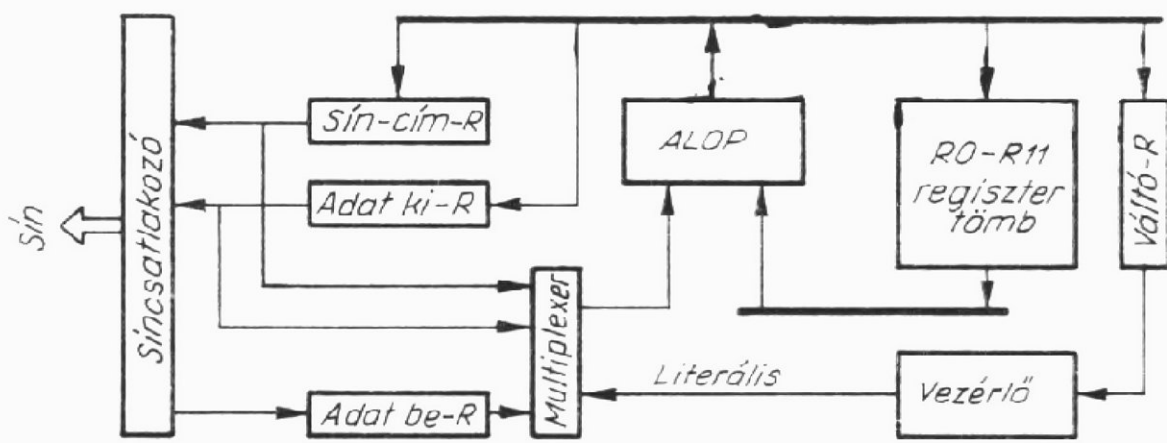
3.13. ábra.
Kétsín-rendszerű processzor



3.14. ábra.
Háromsín-rendszerű processzor, megosztott vezérléssel



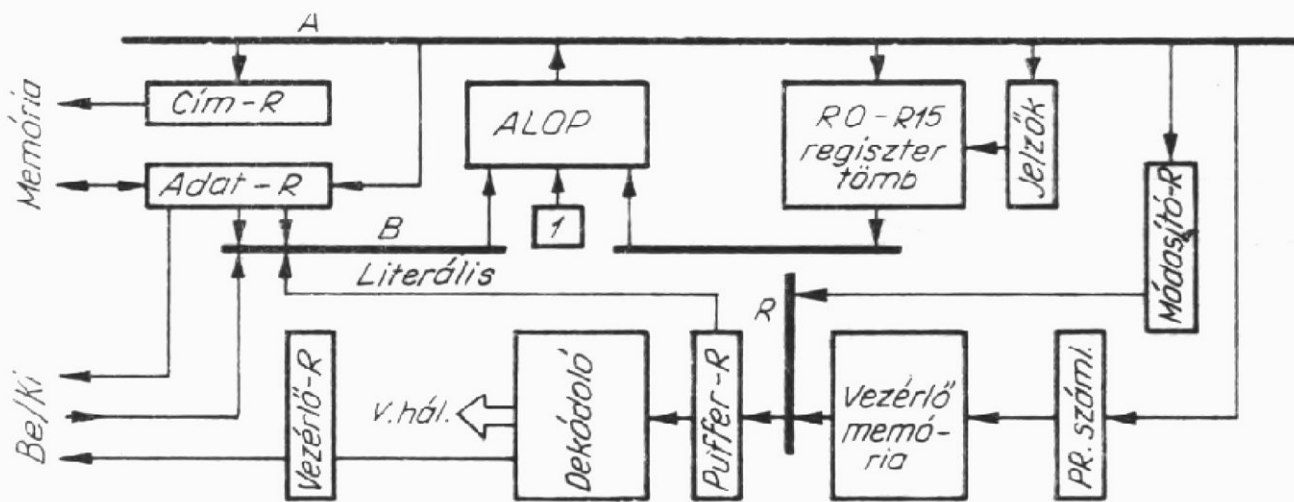
3.15. ábra.
Elfajult háromsínű processzor



LOCKHEED - SUE

3.16. ábra.

Elfajult háromsínű processzor, előretekintő utasítás-szekventálással



MICRODATA - MICRO 800

3.17. ábra.

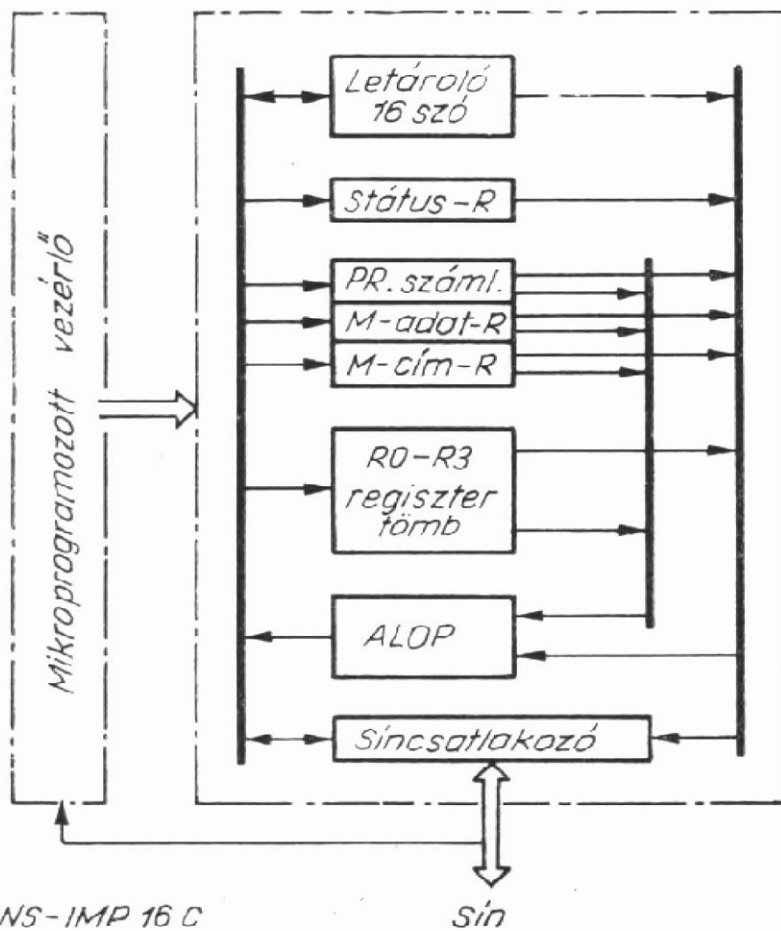
Mikroutasítás-szinten végrehajtó processzor

arra jó, hogy megszakítás esetén, egyik tömbről a másikra idővesztés nélkül lehessen áttérni.

Meg kell említeni még a gép kezelőpultjához tartozó kapcsoló- és megjelenítő-regisztereket, melyek segítségével egyrészt a gépbe adatok vihetők, másrészt a gép (esetleg csak a processzor) regisztereinek tartalma megjeleníthető. A kezelőpult regiszterei szorosan a processzorhoz tartozhatnak, más esetben pedig periférikus készülékként kapcsolódnak a gép sínrendszerére.

Operatív tulajdonságok

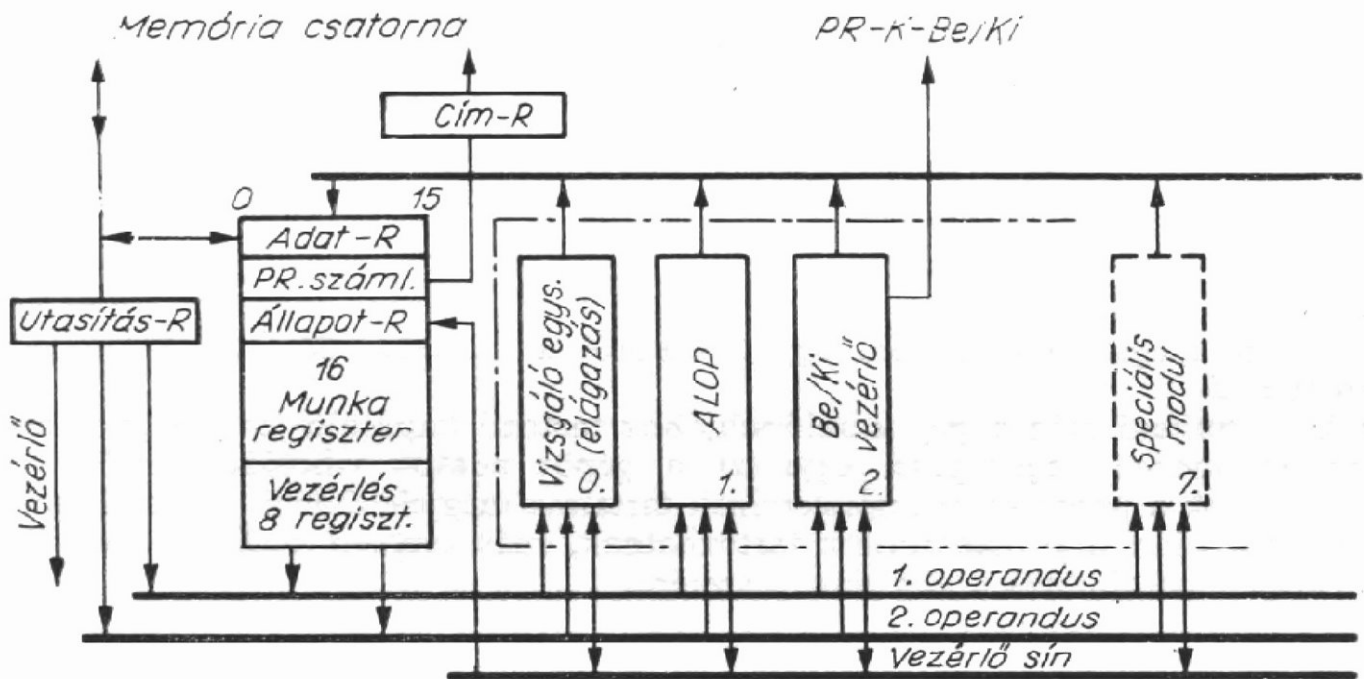
Ismerünk alapkiépítésében csak átmenő operátort tartalmazó processzort (GRI 99; 3.13. ábra), általában azonban fordított a helyzet: a processzorokban külön átmenő operátor egyáltalán nem található, jobban mondva, az átmenő operátort is az aritmetikai—logikai operátor, vagy/és operatív regiszterek helyettesítik. A PDP 8-nak például három operatív regisztere (adatregiszter, programszámláló, címregiszter) van az egyoperandusú műveleteket is végző ALOP mellett.



NS-IMP 16 C

sín

3.18. ábra.
Bitszeletelt modulokból felépített mikroprocesszor



INTERDATA 80

3.19. ábra.
Mérsékeltlen nyitott (moduláris) strukturájú processzor

Vertikális és horizontális jellegű aritmetikai—logikai operátorokat különböztethetünk meg. A vertikális jellegű operátor csak legalacsonyabb szintű műveleteket végez. A vertikális operátorral felsőbb szintű művelet a műveletnek az operátor

műveleteire történő lebontásával, algoritmizálásával (pl. szorzás ismételt összeadással), programozási úton végezhető. A horizontális jellegű operátor a felsőbb szintű műveleteket (pl. szorzás, osztás) közvetlenül hajtja végre.

A processzorok, műveleti lehetőségeik kiterjeszhetősége tekintetében, legtöbbször modulárisan bővíthetők. A bővíthetőségre többféle megoldást dolgoztak ki. Mikroprogramozott vezérlőjű processzoroknál pl. a horizontalitás irányába való továbblépés a processzor adathálózatának érintése nélkül, a vezérlés szintjén történhet. Pusztán a vezérlőmemória bővítése, a mikroprogramrendszer kiterjesztése útján, a gép utasításkészlete a magasabb szintű műveletek utasításaival kiegészíthető.

A vezérlés szintjén történő bővíthetőség egy további lehetőségével találkozunk a GRI 99 számítógépnél, mely ún. külső utasításokat is elfogad. A processzor (a gép) sínrendszerére, a többi egység érintése nélkül, szerkezeti programgenerátor kapcsolható. A programgenerátor, meghatározott gépi utasítás alapján, az általa képviselt magasabb szintű művelet mikroprogramját külső utasítások sorozataként generálja.

A műveleti adottságok kiterjeszhetőségének más módja: olyan kiegészítő egységet adni a processzorhoz, mely eredeti ROV struktúrájának megváltoztatásával jár mind a regiszter-operátor, mind a vezérlés szintjén is. A bővítőegység paraméterei, a hozzákapcsolás módja eleve meghatározottak, ezért ez a megoldás az előbbi kettőnél lényegesen merevebb.

A felsorolt módszereken túlmenően, bővítésként kiegészítő operátorokat és processzorokat is alkalmaznak.

Végrehajtási szintek

Végrehajtási szint tekintetében ötféle processzor különböztethető meg:

1. utasításszinten végrehajtó,
2. mikroutasításszinten végrehajtó,
3. dinamikus mikroprogramozású,
4. két szinten végrehajtó,
5. compiler orientált (letároló aritmetikájú).

A gépek túlnyomó többsége utasításszinten végrehajtó processzorral rendelkezik; az utasítások és adatok a főmemóriában foglalnak helyet. A rögzített utasításkészlet minden utasításának a mikroszinten előírt vezérlési algoritmus felel meg.

A mikroutasításszinten végrehajtó processzor (pl. MICRO 800; 3.17. ábra) egyedül mikroutasításokat ismer. A programok mikroutasításokból épülnek fel, és a konstansokkal együtt a vezérlőmemóriában található. A főmemóriában csupán változók, mutatók és jelzők tárolódnak. Ennek a processzortípusnak nincs utasításregisztere és programszámlálója, ill. ezeknek a regisztereknek a feladatát a vezérlőmemória programszámlálója (címregisztere) veszi át. Szerepel viszont egy különleges feladatkörű módosítóregiszter, mely a vezérlő memória kimenetének dinamikus módosítására szolgál. A MICRO 800 processzornak további specialitása az előretekinthető utasításszekventálás; a soron következő mikroutasítás lehívása megkezdődik, mielőtt még a folyamatban levő végrehajtása befejeződne.

Előretekinthető logikával más kisgépekben is találkozhatunk, pl. a Lockheed—SUE processzorában (3.16. ábra). Mivel azonban a SUE utasításszinten végrehajtó gép, az előretekinthető logika nem mikroutasítás-, hanem utasításszinten funkcionál. Az előretekinthető logika két utasításregisztert igényel, melyek közül az első az RO—R11 regisztertömb eleme, a második (az ábrán Váltó—R) individuális regiszter.

A dinamikus mikroprogramozású processzor (HP 2100S, INTERDATA 85) bár utasításszinten dolgozik, de utasításkészlete nem rögzített; a vezérlőmemória tartalma változtatható. A mikroprogramrendszer rutinjainak teljes készlete a fő-

memóriában foglal helyet, ahonnan mindig a pillanatnyi szükségletnek megfelelő részhalmoz kerül át a vezérlőmemóriába, míg a fölösleges mikrorutinok a főmemóriába térnek vissza (swapping).

A két szinten végrehajó processzor (GRI 99) utasításainak (éppúgy, mint a korábban említett külső utasításoknak) a formátuma megegyezik a mikroutasítások formátumával. A mikroutasítások, utasítások és külső utasítások tehát azonos készlethez tartoznak. A felhasználói-, ill. rendszerprogramok a főmemóriában foglalnak helyet. A főmemóriában tárolt utasítások (eredetükre utalva f-utasításoknak nevezzük), valamint a szolgálatkérő jelek kezelését a mikroprogramozott vezérlő végzi. Az f-utasítások, végrehajtás során, a processzor utasításregiszterébe kerülnek; őket a vezérlő értelmezi. Ugyanakkor azonban az f-utasítások közvetlen vezérlőelemeket is tartalmaznak, az utasításregiszter a vezérlő kimenetével mintegy VAGY kapcsolatban áll.

A compiler orientált processzor operátor—regiszter hálózata letároló struktúrájú (MICRODATA 32/S). A compiler sokkal könnyebben kezeli a letároló struktúrát, mint a kisgépeknél általános ALOP + regisztertömb együttesét, ahol az adatok helyének nyilvántartásával kapcsolatban bonyolult adminisztrációt kell végeznie. Rögzített-peremű letároló és a peremmenti regiszterekkel kapcsolatban álló aritmetikai-logikai operátor használata esetén sem a betétel, sem a kivétel nem igényel regisztercímet, s ALOP művelet kapcsán sincs szükség operanduscímekre; ez az, ami a compiler munkáját lényegesen leegyszerűsíti.

Felépítésmód

A processzorok sorában mind a tiszta ponttól-ponthoz adathálózatú, mind a tiszta sínszervezésű, de a kettő közötti átmenet is megtalálható. Vezérlőrendszereikben radiális és sínszervezésű vezérlőhálózatok egyaránt előfordulnak.

A ponttól-ponthoz adathálózatú, radiális vezérlőhálózatú, utasításszinten végrehajtó processzor jellegzetes képviselője a PDP 8 (3.11. ábra). Vezérlője kötött programú. Újabb változata a PDP 8/E, elődjétől eltérően, már kiterjedten alkalmazza a sínszervezést.

Az egysínű processzor (3.12. ábra) jellemzője az ALOP bemenő és kimenő pontjain elhelyezkedő időszaki tárolóregiszter. E regiszterekre amiatt van szükség, mivel a sínen egyidejűleg csak egy adat vihető át. A processzor a belső és a külső (periferikus) állapotok rögzítésére külön-külön regisztert (Állapot-R; PER. állapot.) használ. A regisztertömb általános célú regisztereket ölel fel. A rendszer sajátossága két címregisztere. A processzorból kiinduló memóriasínhez, elvileg max. 64, egyenként 16...64 K kapacitású memóriablokk csatlakoztatható multiprogramozás céljából. Az egyes blokkok kijelölésére a blokkcím-regiszter (Blokkcím), a blokkon belüli címzésre a címregiszter (Cím-R) szolgál. A processzornak kötött programú, aszinkron rendszerű vezérlője és radiális vezérlőhálózata van.

A 3. 13. ábra kétsín-rendszerű processzora a gép fő sínjeit használja belső adatátvitel és vezérlés céljára is. A vastagon kihúzott vonal ez esetben nem csupán adatvonalat, hanem adatvonalból és vezérlővonalból álló komplex sánt jelöl. A sínek a processzoron túl, az egész rendszeren is keresztülhaladnak. A sínek bekötő vonalai a sínekkel való kapcsolatot ezért éppen csak szimbolizálják. A kétsín-rendszerű processzor mint a korábbiakban az ábrára való hivatkozással már említettük, különböző szintű utasításokkal dolgozik. Másik különlegessége felépítésmódjának (és működésének) nagymérvű szimmetriája. A két sánt áthidaló átmenő operátor világosan és egyértelműen elkülönül ALOP-tól és a regiszterektől egyaránt. Bármely két regiszter közötti átvitel mindig ALOP-on keresztül történik, mely az adatot

meghagyhatja változatlanul, de átvitel közben műveletet is végezhet rajta (inkrementálás, komplementálás, eltolás). Ezzel szemben ALOP csak két operandusú műveleteket végez. A vizsgálóegység, a vezérlő kiegészítő részeként, feltételes f-utasítások kapcsán, a sínrendszeren található bármely regiszter (ALOP kimenő regiszterét is beleértve) tartalmán végezhet vizsgálatot. Adott feltétel teljesülése (vagy a feltétel negációja) esetén a programszámláló tartalmát saját letárolójába helyezi át, a programszámlálóba pedig az f-utasításban szereplő memóriacímet helyezi. A parancsgenerátor, f-utasítás hatására az f-utasításban megcímzett egységhez parancskódot továbbít (pl. ALOP-hoz „összeadás” parancsot). Az állapot ellenőr a készülékek (általában a sínrendszerre kapcsolódó egységek) állapotkódjainak vizsgálatát végzi. Előírt feltételek teljesülése (vagy a feltételek negációja) esetén utasításátlépést hoz létre. ALOP a parancsgenerátortól kapott parancsnak megfelelő műveletet hajtja végre az 1. M—R, 2. M—R munkaregiszterek tartalmán. Az eredmény ALOP kimenő regiszterébe kerül. Az ábra nem tünteti fel a különböző állapotjelzőket, melyek az egységek mentén szétszórva találhatók, de amelyek tartalma — mint a gépi kontextus része — adott esetben egyetlen regisztertartalomként tevődik félre, ill. áll vissza. Nem látható az ábrán a megszakítórendszer programkapcsolatú megszakítás—állapotregisztere, az indexregiszter és a kezelőpult-regiszterek. A kétsínű processzor adatátvitel orientációjú, mérsékelt aritmetikai-logikai műveletvégző képességű rendszer. Felépítésmódjának sajátossága az igen nagymérvű moduláris flexibilitás. Különböző elemekkel (pl. az előzőekben említett szerkezeti programgenerátorral) egyszerűen bővíthető.

A háromsínű processzorok tipikus képviselőjét a 3.14. ábra szemlélteti. Két ekvivalens munkaregiszterével, regiszterek közötti műveleteket is végezhet. Az RO-R3 tömb a vezérlő által használt, programozási úton nem hozzáférhető, időszaki tároló. A hányados- és maradékregiszter osztás alkalmával szerepel. Radiális hálózatu vezérlőrendszerének érdekessége, hogy két jól elválasztott részből tevődik össze, melyek közül a mikroprogramozott vezérlő a belső adatprocesszálást, míg a logikai áramkörökből álló Be/Ki vezérlő a programkapcsolatú adatátvitelt irányítja.

A vezérlőnek önálló feladatot ellátó két vagy több egységre való elkülönülésével más processzoroknál (mint pl. az előbb tárgyalt GRI 99, PDP 11, és Lockheed SUE) is találkozhatunk. A PDP 8 ellenpélda: ott a processzor és a memória vezérlése fonódik össze, szinte szétválaszthatatlanul.

Elfajult háromsínű processzora van a PDP 11 és a SUE gépeknek. Az utóbbinál a harmadik sín multiplexer helyettesíti, mely nem más, mint sínbe inkorporált kapcsolórendszer. [Elvileg mindegy, hogy a kapcsolók (kapuk) a síncsatlakozó vezetékek melyik végén helyezkednek el]. A PDP 11 processzorában három sín található ugyan, de ALOP, működése során esetenként csak két sinnel áll kapcsolatban. Minthogy ALOP bemenetei (alternatív) egy sínre is csatlakozhatnak, az operandusok időszaki tárolására ez esetben is ugyanúgy szükség van, mint az egyadatsínű processzor esetén. Az időszaki tárolást rezeszek végzik.

A PDP 11 regisztertömbje 16 regisztert ölel fel. Szerepük szerint közülük egy állapotregiszter, egy programszámláló, egy speciális peremmutató, hat általános célú regiszter, kettő időszaki tároló. SUE regisztertömbjének 12 regiszteréből egy állapotregiszter, egy programszámláló, egy utasításregiszter, hét általános regiszter, kettő időszaki tároló szerepkörét tölti be. Az általános regiszterek munkaregiszterként, címképző regiszterként, ill. peremmutatóként szolgálhatnak mindkét gépnél. A processzoron belül ugyan sem a PDP 11-nek, sem SUE-nek nincs letárolója, de autoinkrementáló utasításaik segítségével a főmemórián belül egynél több mozgóperemű letároló is képezhető. A PDP 11 speciális peremmutatója a programkitérések (megszakítás, szubrutin) folyamán automatikusan képződő letároló peremére mutat. Mindkét processzor-egysínrendszerű kisgéphez tartozván, közös jellemzője, hogy vezérlőik-

től — önálló szerepkörrel — elkülönül a gép sínkezelő egysége. A PDP 11 kötött programú, SUE mikroprogramozott vezérlővel rendelkezik.

SUE-hoz szerkezetileg igen közel áll a már említett MICRO 800 is (3. 17. ábra); az eltérések főleg a két gép végrehajtási szintjének különbözőségét tükrözik.

Több, mint három adatsínű processzorok (pl. VARIAN 520/i; 620/i) is vannak. Adott kiterjedésű hálózaton belül a sínek számának szaporodása a ponttól-ponthoz adathálózatba való fokozatos átmenettel jár.

Kommunikációs kapcsolatok

A processzor egy vagy több egymástól független csatornán át kommunikál környezetével. A csatornák számától függően egy- és több-bejáratú processzorokról beszélnek. Amikor a bejáratok számáról van szó, világosan meg kell különböztetni a processzort a központi egységtől, mely utóbbi a processzoron kívül memóriát, különböző járulékos egységeket, kezelőpultot, stb. is felölel.

Egybejáratúak az egysín-rendszerű gépek processzorai (PDP 11, SUE). Az egybejáratú processzor mind a főmemóriával, mind a készülékekkel, mind a kezelőpulttal egyetlen csatornán át kommunikál; a közvetlen memóriakapcsolat útvonala a processzort teljesen elkerüli. A több-bejáratú processzornál a főmemória, a konzol-írógép, a kezelőpult, a periférikus készülékek kollektívája, külön-külön csatornával csatlakozhat, sőt a processzor és a főmemória szoros egysége miatt tulajdonképpen még a közvetlen memóriakapcsolat útvonala is félig-meddig a processzoron keresztül halad (1. PDP 8). A jellegzetes több-bejáratú processzor két kommunikációs útvonallal: programozott adatátviteli csatornával és memóriasínnel rendelkezik, a közvetlen memóriakapcsolat útvonalával pedig nem érintkezik.

Mikroprocesszorok

A mikroprocesszorok kalkulátor, ill. számítógép orientációjúak. Ez utóbbiak a kisgépekhez viszonyítva nem elvi, hanem méretkategóriabeli eltérést mutatnak. A méretek zsugorítására való törekvés a félvezető technológia oldaláról adódó követelményekkel együtt, bizonyos strukturális kihatásokkal is jár. Az első speciális tényezőt az integrált áramkörtok kivezetésszámának korlátai jelentik. Az egysín-rendszer kedveltsége a mikrogépek körében ezzel magyarázható.

A processzor szervezőmódja a technológiai kivitel függvénye. Négy jellegzetes kivitelmodot említhetünk meg:

1. Az egész processzor egyetlen félvezető lapkán helyezkedik el (INTEL 8080).
2. A processzor két (esetleg három) lapkára oszlik el, melyek közül az egyik a regiszter-operátor hálózat található, a másik (kettő) a mikroprogramozott vezérlőt hordozza (AMI 7300, AMI 7200).
3. A processzor regiszter-operátor hálózata bitszeletelt modulokból áll (egy modul pl. 4 bit szélességű). A modulok száma az eredő processzor regiszterszélességének megfelelően alakul (8-bites regiszterszélesség, 4-bites modul esetén pl. a modulszám: 2). A vezérlő külön lapkán foglal helyet (NS IMP 16 C).
4. A regiszter—operátor egység és a vezérlő is egyaránt bitszeletelt (Computer Automation-Naked Mini LSI).

Az egyes lapkákon belüli hálózatok a már ismertetett szervezési elveknek felelnek meg. A processzor egészének felépítésmódját azonban sajátosan befolyásolják a lapkák összeköttetésére vonatkozó követelmények. A bitszeletelt rendszernél pl. megjelennek a modulokat összefogó gyűjtősínek, valamint az egyes modulok

operátorait összekötő, helyérték átvitelt biztosító gyöngyvezetékek (Naked Mini LSI).

A 3.18. ábra a 3. megvalósítási módnak megfelelő mikroprocesszor szerkezetét mutatja, mely sínrendszere szerencsés belső kiképzése révén, a modulok külön (külső) összekötő sínjeit nélkülözhetővé teszi. Az IMP 16 C regiszter-operátor hálózatának elemi modulja 4 bit szélességű; a processzor négy modult tartalmaz (regiszterszélesség: 16 bit). A processzor adathálózata háromsín-rendszerű. Itt találkozunk — az eddigiek folyamán elsőízben — kettős kimenetű regiszterekkel, valamint a processzor részét képező 16 szó kapacitású, rögzítettperemű letárolóval. Ez utóbbi egyébként más mikroprocesszornál is előfordul.

Megemlítjük a több lapkára eloszló mikroprocesszor még egy sajátosságát. A vezérlővektor, ugyancsak a kivezető lábak számának kötöttsége miatt, kódolt alakban lép át a vezérlőmodultól a regiszter-operátor modulhoz (modulokhoz).

Modularitás

A processzorok kétféle modularitása technológiai és funkcionális. A t-modul olyan univerzális építőelem, melynek alkalmazásával az építőelemek választéka minimumra csökkenthető. A t-modulokat azon elvek együttesének keretében alakítják ki, melyek alapján a modulokból különböző processzorstruktúrák képezhetők. Az univerzális t-modul példaképpen a Raytheon AS—80, nyolcműveletű operatív regisztert említjük meg.

Az áramköri integráció növekedésének irányában, a t-modul szintje is a nagyobb egységek irányában tolódik el. A moduláris (tehát nem egy lapkán helyet foglaló) mikroprocesszor — mint említettük — bitszeletelésű regiszter — operátor és vezérlőmodulokból épül fel. A kicsiny regiszterszélességű egységek egymáshoz kapcsolhatók, miáltal mind az adathálózat regisztereinek és operátorainak, mind a vezérlőmemória rekeszeinek szélessége, bizonyos határok között, tetszés szerint alakítható.

Funkcionális modularitáson a processzor azon tulajdonságát értjük, mely alapján architektúrája, bizonyos a processzorhoz kapcsolt modulokkal bővíthető. A zárt struktúrájú processzorok, alapösszeállításukhoz viszonyítva, csupán előre meghatározott, korlátozott számú funkcionális egységgel bővíthetők. A nyitott struktúrájú processzorok ezzel szemben, ha nem is korlátlan (mint az gyakran elhangzik), de jelentékeny számú, előre nem meghatározott funkcionális egységgel bővíthetők.

A ponttól-ponthoz adathálózatú, radiális vezérlőhálózatú processzorok zárt struktúrájúak. A PDP 8 processzorának pl. csupán egyetlen kiegészítő aritmetikai-logikai operátora van.

A nyitott struktúrájú, tehát nagymértékben moduláris processzornak mind az adat-, mind pedig a vezérlőhálózata (ez utóbbi legalábbis részben) sínszervezésű. Teljesen nyitott struktúrájú pl. a GRI 99 (3.13. ábra) processzora, míg az INTERDATA 80 (3.19. ábra) a mérsékelten nyitott struktúra példája; sínrendszerére a három alapoperátor mellett (ezek közül kettő a vezérlőnek az operátor—regiszter hálózatba sajátosan illeszkedő egysége) tetszőlegesen további öt kapcsolható.

A modularitás elvén, a processzorok, az operatív elemeken túl, esetenként más egységekkel (így: regiszterek és regisztertömbök, paritásellenőrző, időmérő, hálózat-kiesés elleni védelem stb.) is bővíthetők.

A moduláris flexibilitás nem csak a hálózatok, hanem a vezérlő felépítésmódjának is függvénye, hiszen új funkciók csak a vezérlő algoritmusrendszerének kiegészítése révén realizálhatók. A moduláris flexibilitás követelményeit leginkább a mikroprogramozott vezérlők elégítik ki.

3.1.4. Készülékvezérlők, periferikus processzorok és periferikus készülékek

A készülékvezérlők és periferikus processzorok a ROV szinten a már leírt elvek alapján szerveződnek. Az alábbiakban tulajdonságaikat, működésmódjukat és feladatkörüket tekintjük át.

A készülékvezérlő

A processzor, ill. a memória és valamely készülék vagy a rendszerbe tartozó készülékek egymás közötti kommunikációs összeköttetésének megvalósítására, valamint a készülék feladatainak lebonyolítására szolgál. Főbb részei: a csatlakozó-, a vezérlő- és a végrehajtó egység. A csatlakozó- (interface) egység a készülékvezérlő és számítógép összekapcsolására szolgál. A végrehajtó a készüléket működteti. A vezérlő a számítógéptől kapott parancs által meghatározott vezérlési algoritmus(oka)t állít elő, a végrehajtó bemenetére adott jelsorozat formájában. Az egységek ellenőrző és jelző áramkörökkel egészülnek ki, melyek a készülék és a készülékvezérlő állapotát, a vezérlési folyamatok menetét, a számítógép—készülék adatátvitelt ellenőrzik, ill. jelzik. Az alapszintű készülékvezérlő műveletet nem végez, az adatot mindkét irányban változatlanul továbbítja. (A soros-paralel, paralel-soros átalakítást, két adatelem összekapcsolását vagy szétválasztását, nem tekintjük műveletnek).

Csatornaprocesszorok

A készüléket a központi egységgel összekötő adatvonalba a központi processzort tehermentesítő műveletvégző elem: csatornaprocesszor is beiktatható. A csatornaprocesszort *preprocesszornak* vagy *postprocesszornak* nevezik aszerint, hogy bemenő vagy kimenő adaton végez-e műveletet. A pre-, ill. postprocesszor a készülékvezérlővel egyetlen egységet is képezhet, másrészt egyetlen csatornaprocesszor pre- és postprocesszor feladatkörét is elláthatja. Alkalmazható: kódformátum konvertálására (pl. a géppel nem kompatibilis mágnesszalag-egység mellett); multiplexelt A/D átalakító által szolgáltatott adatok bináris—lebegőpontos konverziójára; fordított szerepkörben: a számítógép és a D/A átalakító között; adatok összehasonlítására és előszortírozására; mérési adatok linearizálására; mérési adatok dimenzionálására; numerikus vezérlésű készülékek, így pl. grafikus megjelenítő eszközök készülékorientációjú geometriai algoritmusainak a számítógéptől kapott parancsok alapján való képzésére stb.

A csatornaprocesszor és a készülékvezérlő egyaránt lehet határozott processzálsí/vezérlési feladatokkal rendelkező kötött programú berendezés. A célberendezés helyett mindinkább előtérbe kerül a mikroprocesszor (mikrogép), mely az aktuális feladathoz program szinten illeszthető. A készülék-csatlakoztatási probléma sajátos megoldását nyújtja a DEC, meghatározott építőelemkészletre alapozott, PDP 16 számítógépes tervezőrendszere, mely a megadott kiindulási paraméterek alapján a feladathoz illeszkedő készülékvezérlőt, ill. csatornaprocesszort generál.

Társoperátorok és társprocesszorok

A műveletvégző képesség nemcsak a központi processzor moduláris bővítése, hanem a géphez periferikus készülékként csatlakozó társoperátorok és társprocesszorok révén is kiterjeszthető. A csatornaprocesszortól eltérően a társprocesszor

egyedül a központi processzorral és memóriával áll kapcsolatban, feladata pedig az, hogy a központi processzor által csak körülményesen és lassan elvégezhető komplex műveleteket hajtson végre nagy sebességgel. A tárprocesszor tehát nem periférikus feladatokra, hanem ugyancsak központi processzállásra szolgál, sőt esetenként a viszony megfordulhat, amikor is a központi processzor végzi a tárprocesszor mellett az elő- és utóprocesszállási feladatokat. A tárprocesszor szerkezeti bonyolultsága megközelítheti, sőt túl is szárnyalhatja a központi processzorét. Különbséget tesznek tárprocesszor és tároperátor között. Az utóbbit, az előbbinél lényegesen egyszerűbb architektúra, ill. szerkezet jellemzi.

A tároperátor a központi processzorral egyedül programkapcsolatú átvitel módban kommunikál. A tárprocesszor a programozott csatornán kívül a közvetlen memóriakapcsolatot is igénybe veszi. Tárprocesszor alkalmazása egyedül abban az esetben indokolt, ha mindkét processzor ideje kielégítően hasznosítható, ami szimultán működésmódot tételez fel. A programkapcsolatú adatátvitel azonban a központi processzort lefoglalja; a tárprocesszor intenzív tevékenysége a központi processzort megbéníthatja. Az együttes munka ilyen formája nemcsak gazdasági szempontból hátrányos, zavart okozhat a mérés-technikai feladatok lebonyolításában is. Ezért a programkapcsolatot csupán a műveletkezdeményezéshez alkalmazzák; az utasítások, az operandusok és az eredmény adatainak átvitelére a közvetlen memóriakapcsolatot használják. A tárprocesszor külső adatforgalmának csökkentése érdekében egyes tárprocesszorokhoz igen jelentékeny kapacitású (4...8 K) operatív memóriát is kapcsolnak.

A különböző rendeltetésű tároperátorok és tárprocesszorok sorából a szorzást és osztást végző fixpontos aritmetikai operátort, a lebegőpontos aritmetikai tárprocesszort, valamint a gyors Fourier-processzort említjük meg. A lebegőpontos processzorok egyszerűbb változatai összeadás, szorzás és osztás műveletűek; fejlettebb változataik hatványsorok értékelésére, \sin , \cos , \arctg és logaritmus számítására, komplex számokon végzett műveletekre és gyors Fourier-transzformációra is képesek. Bizonyos Fourier-processzorok másrészt a Fourier-transzformáción kívül a transzformációval kapcsolatban nem álló aritmetikai műveleteket is ismernek.

A funkcionális modularitás, ugyanúgy, mint a mikroprogramozott vezérlés elve, a tárprocesszorok körében is alkalmazásra talál.

Periférikus készülékek

Két nagy csoport különböztethető meg:

1. háttértárolók és
2. be/kimenő készülékek.

Egyesek közülük háttértárolásra és adat be/kivitelre egyaránt szolgálhatnak.

A periférikus készülék általában periférikus eszközből és készülékvezérlőből áll. A periférikus eszköz lényegében átalakító, mely a gép és valamely közbenső hordozó, a gép és a külvilág közötti jelalak konverziót végez.

A be/kimenő készülékek sorában hagyományos, valamint speciális készülékeket találunk. A számítógép-irányítású mérőrendszerben az előbbiek az ember és a gép, ill. a gép és egy másik gép közötti kommunikációra szolgálnak. A speciális készülékekhez a mérőhálózat különböző elemei tartoznak. A számítógépes mérőrendszer létrehozásának folyamatában a két nagy egység, — a számítógép és a mérőhálózat — között a hangsúly sajátosan tolódik el a mérőhálózattól a géphez. Kezdetben a mérőrendszer oldaláról fogalmazódnak meg az alkalmazandó számítógéppel szembeni követelmények, majd amikor a gép és a mérőhálózat együttesének szervezésére kerül sor, a mérőkészülék a gép periferiájaként jelenik meg. Ennek megfelelően beszélünk

mérőrendszer-orientációjú számítógépről, ill. számítógép-orientációjú mérőkészülékről. A számítógép-orientációjú mérőkészülékkel, valamint ezeknek a számítógéphez való csatlakoztatása kérdéseivel külön fejezet foglalkozik. A hagyományos periférikus eszközökre és a háttértárolókra a könyv részletesebben nem tér ki.

3.1.5. Az MPK struktúra

A ROV és MPK struktúra kapcsolata

A számítógép regiszter — operátor — vezérlő (ROV) szintjén elhatárolhatók az egyes vezérlők irányítása alatt álló területek. Az egy vezérlő hatáskörébe tartozó, ilyen ún. egyszerű ROV hálózat pl. a processzor, a memória és a készülékvezérlő. Első közelítésben ezek az egyszerű ROV hálózatok a gép memória — processzor — készülék (MPK) struktúrájának építőelemeiként tekinthetők. Az egybevágóság azonban nem teljes. Az egyszerű hálózatok ugyanis nemcsak az MPK szinten, hanem esetenként más ROV hálózat részeként, sőt a ROV szint alatt is megjelennek. A regisztertömb, pl. bár saját külön tárvezérlője van, a processzor részét képezi (az inhomogenitás csak az integrálódás fokozódásával mosódik el). A mikroprogramozott vezérlő maga is processzorként fogható fel, ezért is nevezi gyakran az irodalom mikroprocesszornak, bár ezt az elnevezést más értelemben a miniatürizált központi processzor köti le. A pre- és postprocesszorok a készülékvezérlő részeként, ugyancsak ROV-szintű elemként jelennek meg. A készülékvezérlő pedig mint a periférikus készülék részegysége, valamivel az MPK szint alatt helyezkedik el. A hálózati szintek és alkatelemeik pontos egymáshoz tartozása jelenleg tehát még nem kristályosodott ki. A technológiai viszonyok gyors változása bizonyos mérvű állandó átrendeződéssel és terminológiai határozatlansággal jár, mely azonban — ha figyelembe vesszük — nem zavarja a gép felépítés- és működésmódjának kvalitatív áttekintését.

Adat- és vezérlőhálózatok

Az MPK szintet hálózatai sínszervezése jellemzi; a ponttól-ponthoz és a radiális hálózatszervezési elvet itt ritkábban alkalmazzák. A komplex, adat- és vezérlővonalakból álló, MPK sínek száma a kisgép egyik alapvető struktúrális jellemzője.

Legkézenfekvőbb (korábban a legáltalánosabban alkalmazott) a *háromsínű* MPK hálózat (3.20a ábra). A sínek egyike a processzor és a készülékek összeköttetésére szolgál, a másikon a memóriamodulok osztoznak, a harmadik a közvetlen memóriakapcsolatot látja el. A *kétsínű* rendszerben (3.20b ábra) a készülékek két csoportra oszlanak; a közvetlen memóriakapcsolatú készülékek a memóriasíntre csatlakoznak, s ezek a processzorral és a memóriával egyaránt a memóriasínen át érintkeznek. Végül az *egysínű* gép (3.20c ábra) összes egysége egyetlen sínen át kommunikál.

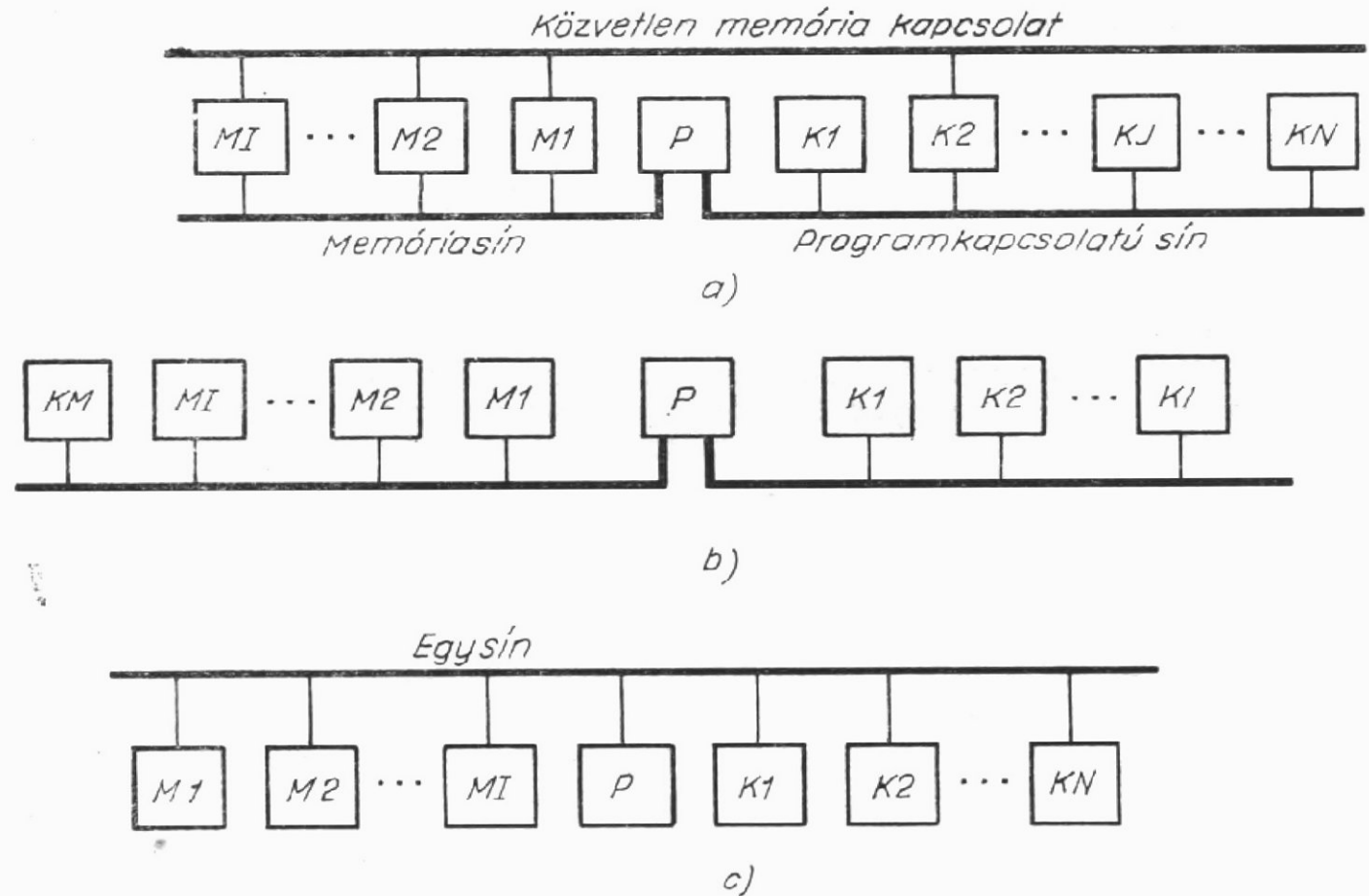
A sínszám csökkentése a korábban önálló feladatú sínek funkcionális összevonásával történik. A funkcionálisan egyesített síntől meg kell különböztetnünk a technikai sínt, melybe több, eredeti funkcióját megtartó sínt az áramköri kártyák csatlakoztatásának egyesítése érdekében vonnak össze (pl. PDP-8/E). A technikai egysínhez csatlakozó kártyák pozíciója kötetlen lehet ugyan, világos azonban, hogy az egymástól eltérő kártyák az egysín más-más vezetékére csatlakoznak. Fordított esettel is találkozhatunk: A GRI 99 gép processzorának két belső sínje a processzorból kilépve a MPK szint többi elemével való kommunikációra szolgál.

Minthogy az elemek egységes módon csatlakoznak, a processzor sínjei az MPK struktúrában tulajdonképpen egyetlen sít alkotnak.

Meg kell jegyeznünk, hogy az MPK struktúra sínjeinek száma általában eltér a hálózat elemét alkotó processzor sínszámától; e sínszámok között nincs szükségszerű összefüggés.

Megkülönböztetünk *teljes sínű* és *csonka sínű* hálózatot. Az előbbiben a funkcionális egységek közötti kommunikációs kapcsolatot egészében sín látja el; az utóbbiban a vezérlési feladatok egy része radiális hálózatra hárul. A teljes sín általában külső, a csonka sín mindig belső sín. A belső sín a központi egység szekrényéből nem lép ki.

A teljes és a csonka sínrendszer felépítésbeli sajátosságait, többsínű gép processzor—készülék hálózatának bemutatásával a 3.21. ábra szemlélteti. A teljes sín



3.20. ábra.

Kisszámítógépek memória—processzor—készülék (MPK) szintű hálózatai

(a) háromsín-rendszer; (b) kétsín-rendszer; (c) egysín-rendszer; P — processzor; M — memóriamodul; K — készülék

mint külső sín a központi processzorból indul ki, befut az első készülékhez, onnét a másodikhoz és így tovább, míg csak az utolsó készüléket el nem éri. Újabb készülékek a sín (melynek fizikai realizációja kábel) meghosszabbításával csatlakoztathatók. Természetesen az összes készülék számát egyrészt a készülékcímek tartománya, másrészt a kábel megengedhető max. fizikai hossza korlátozza. Processzorkezdeményezés kapcsán a sínre küldött készülékcím választja ki a hívott készüléket; készülékkezdeményezés esetén, a kezdeményező jel a sín egy erre szolgáló gyűjtővezetékén át jut a processzorra. A teljes sínű rendszer jellemzője az is, hogy a sínre csatlakozó készülékvezérlő a periferikus eszközzel szoros egységet képez.

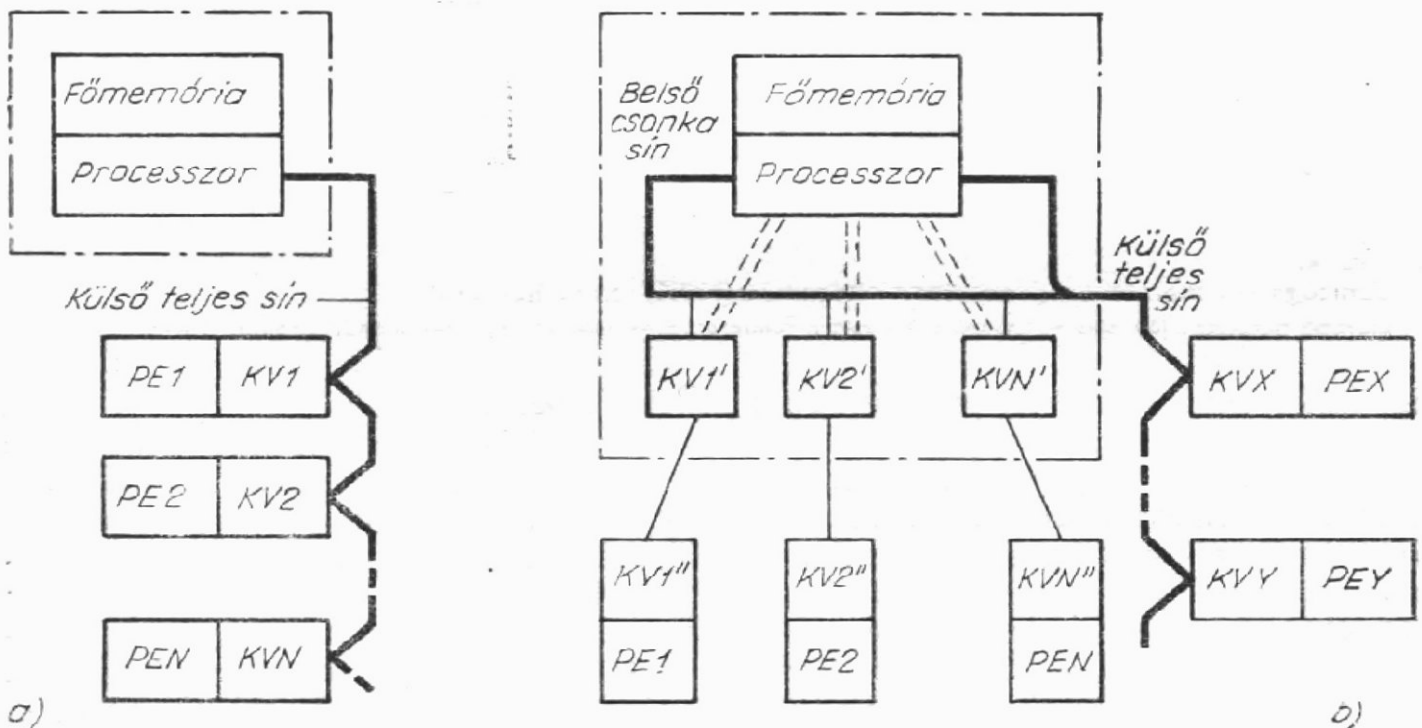
A csonka sín mint belső sín a valamennyi készülékhez tartozó összes adatvonalat és a vezérlővonalak egy részét öleli fel, hiányzik azonban belőle a címvonal és a kez-

deményező jelek gyűjtővonalai. A készülékek kiválasztását a processzor radiális hálózattal végzi; a hívás nem kódolt címmel, hanem a készülékre irányuló explicit jellel történik. A készülék kezdeményező jele sem általánosságban, hanem eredetét kifejező módon mutat a processzorra, elkerülhetővé téve a teljes sínrendszer használata esetében szükséges, a kezdeményezést (megszakításkérés) követő azonosítási folyamatot. Radiális hálózatok alkalmazására a periferia és a processzor csatlakozási pontjai közötti csekély geometriai távolság ad lehetőséget.

A csonka sínű rendszerben a készülékvezérlő két különálló részből áll, melyek közül az egyik a központi egység szekrényében a sínhez és a radiális hálózatokhoz csatlakozva foglal helyet, míg a másik, átvivő kábel túlsó végén, a periferikus eszközzel képez szoros egységet. A kábelek adat- és vezérlővonalból állnak. A központi egységet magasabb szintű építőelemként tekintve, valami ponttól-ponthoz, ill. radiális hálózat körvonalai rajzolódnak elénk. Figyelembe kell venni azonban, hogy ezek a kábelek a készülékvezérlők belső összekötő vezetékét tartalmazzák, s mint ilyenek ROV-szintű és nem MPK-szintű összeköttetések.

Belső sín teljes sín is lehet, ami csupán kábelezésbeli, elrendezésbeli különbséget jelent. Vegyes sínrendszerű MPK hálózat is alkalmazható, melynek a központi egységen belül fekvő szakasza csonka sínként, külső szakasza teljes sínként funkcionál. A vegyes sínrendszer a készülék címzés mindkét módját (kódolt és explicit) ismeri.

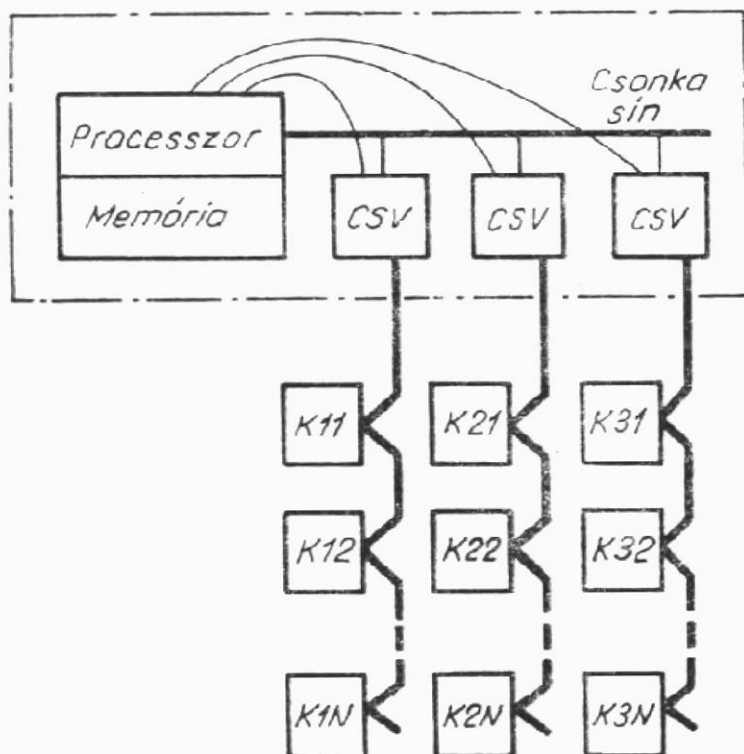
Egyes gépekben csatornavezérlőket (CSV) alkalmaznak (3.22. ábra). A csatornavezérlők belső (csonka) sínre csatlakoznak, s mindegyikükből meghatározott számú készülék között megoszló külső sín indul ki. Ez a szervezőmód megengedi, hogy a készülékvezérlők bizonyos feladatait a csatornavezérlőkre háríthassuk át, miáltal a készülékvezérlők egyszerűsödnek. Ugyanakkor a processzor vezérlőjének szerkezete is egyszerűbbé válik, lévén, hogy a sínmultiplexeléssel járó feladatok egy részét a csatornavezérlők veszik át. A csatornavezérlőknek a készülékvezérlőkénél nagyobb szabadsági foka lehetővé teszi az egyes csatornákon dolgozó készülékek időmultiplex üzemét is.



3.21. ábra.

Kisgépekben alkalmazott sántípusok

(a) külső, teljes sín; (b) belső, csonka sín: KV — készülékvezérlő; PE — periferikus eszköz



3.22. ábra.
Késszámítógép csatornavezérlőkkel
CSV — csatornavezérlő; K — készülék

Az MPK hálózat sínjeinek fontos jellemzője: szélessége, azaz az általa felölelt vonalak, ill. vezetékek száma. (Vonalakon bizonyos típusú információk — pl. cím — átvitelére szolgáló, egy vagy több szálból álló vezetékrendszert értünk). Egyugyanazon feladat különböző szélességű sínekkel is megoldható. Itt is a horizontalitás vertikálitás problémájába ütközünk; ha ugyanis az átvivő vezetékek számát csökkentjük, csökken az egy ciklusban átvihető információ mennyisége is; a gazdaságosabb sínelépítés tehát az átviteli sebesség csökkenésével, egyúttal bonyolultabb síncsatlakozó áramkörökkel jár. A lehetőségeket tekintve: két ellentétes irányú adatvonalat pl. helyettesíthet egyetlen kétirányú vonal, továbbá a vonal keskenyebb is lehet, mint az átvitelben alkalmazott adat. Egyetlen vonal használható adat- és vezérlőinformáció átvitelére is, ha a kommunikációban résztvevő egységeket külön csatornán értesítjük a többfeladatú vonalon momentán áthaladó adatelem természetéről (l. egyes mikrogépeket).

Az MPK hálózat elemeinek kapcsolatai

Az MPK hálózatnak — a ROV hálózattól eltérően — nincs külön vezérlője, mely a többi elem műveleteit, a hálózat egészének folyamatait szervezné. Az MPK hálózat két eleme között mindig egyikük kezdeményezése alapján létesülhet kapcsolat. Kezdeményezést bármely hálózati elem elfogadhat, de kapcsolatot csak folyamatállapotú processzor, ill. készülék kezdeményezhet, memória nem. A kapcsolat kezdeményezés, valamint a létrejövő kapcsolat adminisztrációjának jeleit az MPK struktúra vezérlővonalai közvetítik a hálózat elemei között.

Processzor és készülék különbözik egymástól folyamataik autonómiájának tekintetében. A processzor folyamatai — azoktól az időszakoktól eltekintve, melyekben a processzor a készülék által előírt feladatot hajt végre — kötetlenek; a mindenkorinak megfelelően futnak. A készülék viszont egyedül a reáljellemző kevés számú feladat végrehajtására képes. A processzor, a memória vagy

bármely készülék irányában, kötetlenül élhet kezdeményezéssel, a készülék ön-állósága azonban korlátozott, mivel létesíthető kapcsolatait egyrészt a hálózat szervezőmódja, másrészt a processzor predeterminálja. A memória autonómiája még korlátozottabb.

Az MPK hálózat elemei általában munkaidejüknek csak bizonyos töredékében állnak összeköttetésben más elemekkel, egyébként folyamataik egymástól elkülönülve zajlanak; az elemek párhuzamosan dolgoznak.

A processzor — készülékkel — mindig az általa végrehajtott utasítás alapján kezdeményez kapcsolatot. A készüléket annak explicit vagy kódolt címével választja ki a kollektívából, s a kiválasztott készülékkel a kapcsolatot szolgálati jelei útján realizálja. A kapcsolat létesítésének célja általában valamely adatelem processzor → készülék vagy készülék → processzor irányú átvitele. Az adatelem lehet a feldolgozási rendszer része, de lehet a processzornak szóló állapotkód vagy a készüléknek szóló parancs is. Miután az átvitel megtörtént, a processzor—készülék kapcsolat megszakad, a processzor folytatja munkáját, a készülék pedig várakozó állapotba kerül, vagy saját vezérlőjének irányítása alatt a processzortól a kapcsolat idején kapott feladatát végzi.

Készülék processzorra szolgálatkérő jele útján kezdeményez kapcsolatot. Kétféle szolgálatkérő jel különböztethető meg: megszakításkérés és blokkátvitelen belüli átvitelkérés. Előírt feltételek teljesülése esetén, szolgálatkérő jel hatására, a processzor momentán programvégrehajtási feladatát megszakítja, s áttér a szolgálatkérő jel által definiált kötött szerkezeti (vagy mikro) program végrehajtására. Ez a szerkezeti program a blokk egy elemének átvitelét vagy megszakításkérés kapcsán a processzornak a megszakítási programra való rátérését hajtja végre. A kisgépek túlnyomó többségének szervezőmódja kizárja, hogy készülék processzorra adatátviteli kapcsolatot létesítsen; a készülék kezdeményezése ezen gépeknél nem több, mint kapcsolat megvalósítására irányuló kérés. Az adatátviteli összeköttetést és magát az adatátvitelt a szolgálatkérés alapján a processzor valósítja meg.

Egyes hálózatok szervezőmódja két készülék közvetlen kapcsolatát is lehetővé teszi.

Processzor—memória kapcsolatot csak processzor kezdeményezhet és valósíthat meg. A készülék—memória kapcsolat kezdeményezője közvetlenül a készülék, közvetve a processzor; készülék a memóriához általában csak akkor fordulhat, ha megelőzően, programozási úton a processzor erre már előkészítette.

Az MPK hálózat elemei közötti kapcsolat szinkron vagy aszinkron módon jöhet létre (erre vonatkozó részletes leírás a 4.1. pont alatt).

Hálózati konfliktusok, prioritási hierarchia

A szimultán folyamatok konfliktus-helyzeteket idézhetnek elő; pl. valamely elem egy már folyamatállapotú másikkal fordul, vagy valamely elemhez egyszerre több kezdeményezés érkezik. A konfliktuslehetőségek mindegyikének az MPK hálózat struktúrájába ágyazott megoldása van.

Folyamatállapotú készülékkel a processzor általában nem létesíthet kapcsolatot; a készülék a processzor kezdeményezését visszautasítja. A rendszer vagy kezdeményezés előtt ellenőrzi a készülék üzemállapotát, vagy kezdeményezés után vizsgálja meg, hogy az érvényes kapcsolat létrejött-e. Az ellenőrzés eredményfüggő elágazással rendelkező program útján történik.

Ismeretes olyan kisgép (pl. CII 10010) is, melynek processzora, erre szolgáló utasítással, folyamatállapotú készülékkel is létesíthet kapcsolatot, abból a célból, hogy a készülék folyamatát leállítsa, s a készüléket várakozó állapotba helyezze.

A processzort a készülékekkel összekötő sínen (vagy sínrendszeren) egyidejűleg több készülék is lehet folyamatállapotban, egyidejűleg egynél több készülék is élhet tehát szolgálatkéréssel. A processzornak kell eldöntenie, hogy a jelentkező készülékek közül melyeket részesíti előnyben saját momentán folyamatával szemben, s hogy az előnyben részesített készülékek közül melyiknek a prioritását ismeri el. Mindez a gép prioritási hierarchiája alapján történik. A prioritási hierarchia egyes gépeknél kötött, és teljes egészében a szerkezetben őrződik, más gépeknél — csaknem teljességgel — a programrendszer definiálja, míg átmeneti esetben a szerkezetben rögzített hierarchia — programozási úton — korlátozott mértékben módosítható. Bármely, a processzorral szemben prioritást élvező készülék megszakíthatja a processzor folyamatát, ill. várakozási állapotát is, és a processzort új irányba térítheti. Másrészt, több készülék egyidejű szolgálatkérése közül, a processzor mindig a legmagasabb prioritásút fogadja el. A gépek jelentékeny hányada nem csak a processzor alapfolyamatát, hanem a készülékek által kezdeményezett folyamatot is megszakíthatja, egy az eddigieknél magasabb prioritásrangú készülék szolgálatkérése alapján.

A prioritásrangok rögzítése a processzorhoz és minden egyes készülékhez rendelt prioritási szint alakjában történik. A processzor és a készülékek sokaságában két elem soha sem rendelkezhet azonos prioritási szinttel, ellenben adott esetben egy készülékhez több szint is rendelhető.

A prioritási szintek függetlenek, ha bármely szintről érkező szolgálatkérés megszakíthatja a nálánál alacsonyabb szintű által indított processzor-folyamatot. A gépek egy részénél a prioritási szintek osztályokba oszlanak el úgy, hogy az egy osztályba eső szintek egymástól nem függetlenek, ami azt jelenti, hogy adott osztály szolgálatkérése által indított folyamatot ugyanazon osztályba tartozó más szolgálatkérés nem, csupán magasabb osztályba tartozó forrás jele szakíthat meg. Amennyiben a processzor valamely osztály szolgálatkérését hajlandó elfogadni, úgy az osztályban egyidejűleg jelentkező források közül mindig a legmagasabb szintűt választja ki.

Több készülék egyidejű kezdeményezéséből származó konfliktushelyzet azáltal oldódik fel, hogy míg a gép valamely készülék elfogadott kezdeményezését kezeli, a többi készülék sorára várakozik. A processzor a szolgálatra várakozó készülékek szolgálatkérő jeleit a prioritási hierarchiának megfelelő sorrendben, egymás után dolgozza fel. A várakozók közé lép minden olyan újabb kezdeményező, mely azonos vagy alacsonyabb osztályú, ill. alacsonyabb szintű, mint a momentán elfogadott készülék. A momentán elfogadott készülék is a várakozók sorába kerül, ha időközben nálánál magasabb osztállyal vagy magasabb független szinttel rendelkező készülék jelentkezik szolgálatkéréssel.

A közvetlen memóriakapcsolat a processzorral szemben prioritást élvez ugyan, de a processzor által kezdeményezett memóriaműveletet nem szakíthatja félbe. A memória konfliktusainak előfordulási gyakorisága a memória szervezőmódjának is függvénye. A konfliktusok gyakorisága egybefüggő memóriánál a legnagyobb, mivel ennél műveleti szimultaneitás nem lehetséges. Kedvezőbb a helyzet részenként önálló, több-bejáratú memória esetében. A processzor és a közvetlen csatorna ennél nem ütközik, ha egyidejűleg ugyan, de más-más önálló modulhoz (blokkhoz) fordulnak. A processzor és a közvetlen csatorna konfliktushelyzetében a közvetlen csatorna a processzort reteszeli, miközben a processzortól a memóriának egy vagy több ciklusát eltulajdonítja.

Tekintettel arra, hogy a közvetlen csatorna a processzort reteszeli, a közvetlen szolgálatkérés a készülékek minden egyéb szolgálatkérésével szemben prioritást élvez.

Egyes gépeknél a közvetlen csatorna a memória néhány rekeszét saját szolgálati regisztereként használhatja; egy-egy adatelem átvitelét többciklusú folyamat kíséri,

melynek kapcsán e memóriarekeszek tartalmát a processzor inkrementálja. Ilyenkor időszakosan a processzor a közvetlen csatorna által vezetett egységgé válik, mely saját folyamatai vagy készülék-kezdeményezés szempontjából reteszeltnek tekinthető.

Egynél több memóriakapcsolatú készülék esetén prioritási rangsort kell felállítani. A készülék-kezdeményezések rangsorolására, a készülékek és a memória közötti forgalom lebonyolítására csatornamultiplexert alkalmaznak. A közvetlen memóriacsatornához tartozó osztály szolgálatkérő jelei a gép hierarchiájában minden más szolgálatkérő jelet megelőznek.

Moduláris MPK hálózatok

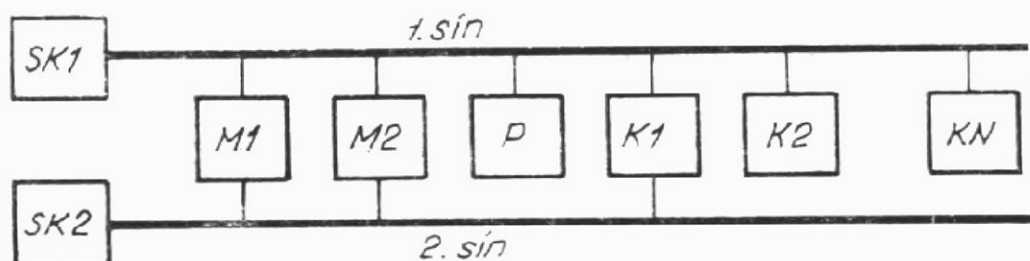
A kezdeti kisgépet strukturális merevség jellemzi. A főmemória a központi processzorral olyan szoros egységet képez (szinkron kapcsolat), hogy az irodalom a „központi processzor” elnevezéssel gyakran a processzor—memória együttest illeti. (Jelenleg ezt az együttest a processzortól való megkülönböztetésül központi egységnek nevezik.) A processzor sebességét a memória ciklusideje korlátozza. A memória egybefüggő; egyetlen bejáratát a processzor és a közvetlen kapcsolat csak egymást kizáró értelemben használhatja: közvetlen adatátvitel esetén a processzor blokkolódik; bővíthetőség szempontjából moduláris ugyan, de mivel egybefüggő, egynél több művelet egyidejűleg nem végezhet. Periferikus processzort kezdettől fogva alkalmaznak, de a két processzor kapcsolata merev; a DEC LINC 8 rendszerénél pl. (ahol a PDP 8 központi processzorához a programkapcsolatú és a közvetlen átviteli csatornára egyaránt csatlakozó LINC elnevezésű periferikus processzor társul) a két processzor időben egymást kizáró módon működik.

A kezdeti korszak merev strukturájú, modulárisan igen korlátozottan bővíthető, MPK hálózatait fokozatosan felváltják a flexibilitás szempontjait érvényesítő moduláris hálózatok. A moduláris hálózat általánosságban már nem egyetlen processzort és nem egyetlen főmemóriát, hanem processzorok és memóriamodulok (blokkok) sorát tartalmazza, melyek egymáshoz, és amelyekhez a hálózat periferikus készülékei, a legkülönbözőbb módon kapcsolódnak.

A korlátozottan bővíthető három-, ill. kétsínű gépekkel szemben a hálózati flexibilitás irányában jelentékeny előrelépést jelent az egysínű gép, mely egynél több processzort és számos, funkcionálisan önálló memóriaegységet foghat össze, s bármely újabb MPK elem egységes csatlakozási filozófia alapján kapcsolható a sínre. Az egysínű gépnél a processzorcentrikusság háttérbe szorul. A főmemória funkcionális egysége is fellazul; gyakran már nem egy, hanem több egymástól független, funkcionálisan is önálló memóriát láthatunk magunk előtt.

Az egysín elnevezéssel sajátos ellentmondást képez az a lehetőség, hogy az egysínű hálózatból kiindulva a sínek száma modulárisan bővíthető. Az egysín bizonyos üzemi körülmények között túlterhelődhet. Több-bejáratú memória birtokában az egysín kiegészítésére egy azonos szerkezetű második sín is bevezethető, amint az a 3.23. ábrán látható. A 3.24. ábra a sínek számának további növelésével nyert moduláris hálózatot mutat be. Új elemként jelenik meg az 1. és 2. sín (processzor) közötti kapcsolat létesítésére szolgáló ablak (A), valamint a hálózati multiplexer (μ), mely utóbbi segítségével a közvetlen memóriakapcsolatú K készülék, program útján, az 1. vagy a 2. sínről is előkészíthető. A bővített hálózatban az egyidejű folyamatok mellett egyidejű kapcsolatok is lehetségesek. Így pl. P1 az M2-vel, P2 az M3-mal kommunikálhat, míg K az M1-ből adatblokkal töltődik fel. Az egysínrendszer így vezet át általános jellegű, többsínű hálózatokba.

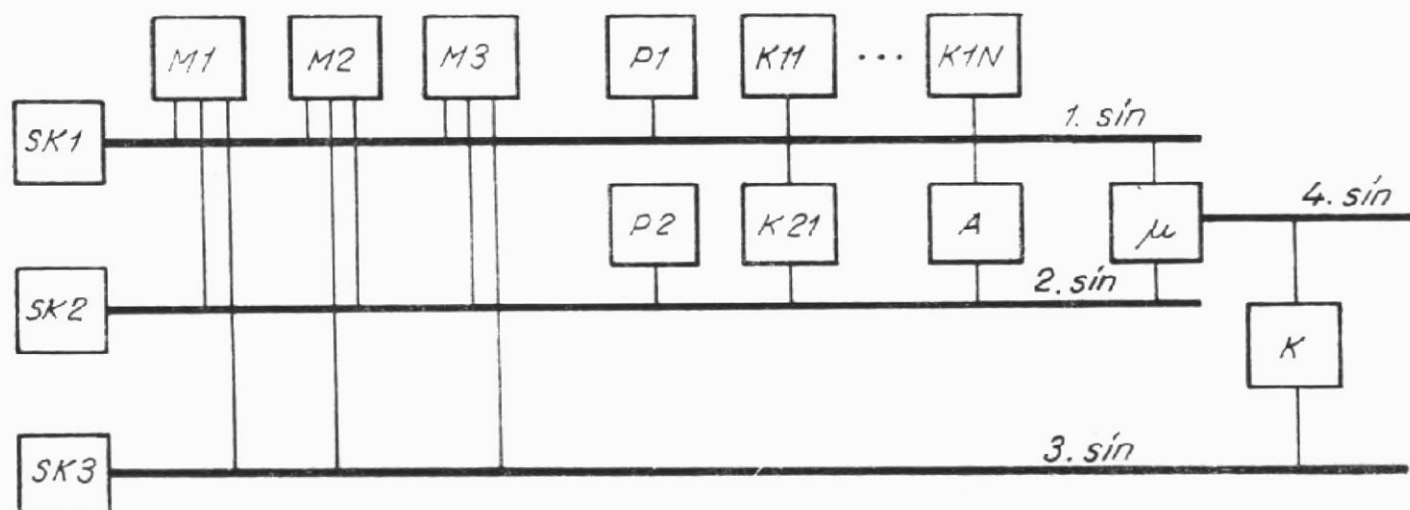
A modularitás követelményeinek megfelelően nem csak a sínek száma, hanem azok szélessége, ezen belül az adatvonal és a címvonal szélessége is (esetleg szakaszonként) változtatható.



3.23. ábra.

Kettős sínrendszerű moduláris MPK hálózat

SK — sínkezelő; M — memória; P — processzor; K — készülék



3.24. ábra.

Többcsatornás sínrendszerű moduláris MPK hálózat

SK — sínkezelő; M — memória; P — processzor; K — készülék; A — ablak; μ — hálózati multiplexer

A modularitás általános követelményeihez tartozik a hálózat elemeinek aszinkron kapcsolata. Az aszinkron kapcsolat eltérő működési sebességű elemek egységes csatlakoztatását is lehetővé teszi. Míg a szinkron hálózatban az elemek működési sebessége az egymással való összeköttetésben meghatározó szerepet játszik (így pl. a szinkron kapcsolatú memória minden moduljának azonos ciklusidővel kell rendelkeznie), addig az aszinkron hálózatban az ökonómiai követelményeknek megfelelő legkülönbözőbb sebességű (és kapacitású) memória modulok helyezhetők el.

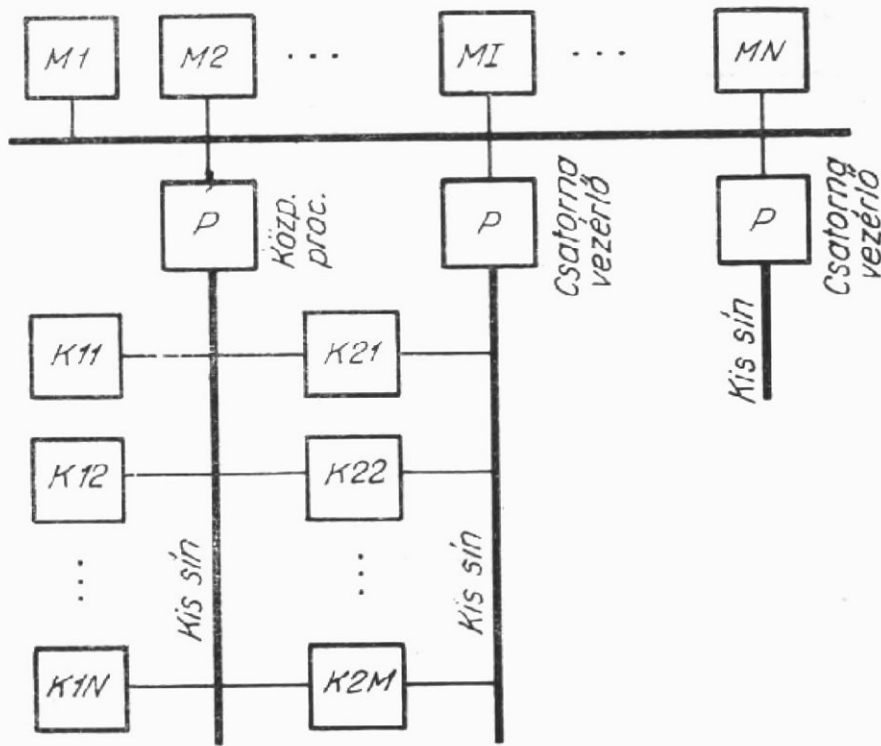
Az ismertetett sínrendszereken kívül moduláris hálózatok létesítésére egyéb szervezési módszereket is alkalmaznak. Ismeretes a fa, mátrix, iteratív és szimmetrikus hálózat.

A moduláris hálózatra igen jellemző processzorainak hierarchiája. A hálózat két processzora egymáshoz viszonyítva lehet mellérendelt, de fölé- alárendeltségi viszony is fennállhat. Ennek megfelelően paralel, ill. műveletvonal (pipeline) processzorokat különböztethetünk meg. A paralel processzorok valamely nagyobb feladat önálló részfeladatait végzik, s a feladat természetétől függően esetlegesen kommunikálnak egymással. A műveletvonal processzorok adott feladat algoritmusa „mentén” sorba állnak. Mindegyik processzor az algoritmus más-más blokkját hajtja végre; a műveletvonal minden processzora az őt megelőzőtől kapja bemenő adatait, és az utána következőnek szolgáltatja saját processzálásának eredményét.

A memóriamodulokat összefoghatja egyetlen címmező, de más-más processzorok által más-más módon használt modulokhoz egymással paralel címmezők is tartozhatnak. A központi processzor mellett elhelyezkedő nagyteljesítményű Fourier-processzor például dinamikusan mikroprogramozott speciális processzor és olyan memória együttesének fogható fel, melynek címmezeje a főmemória címmezejével párhuzamosan fekszik. A moduláris hálózat memóriamodulja állandó jelleggel

egy adott processzorhoz tartozhat, vagy megoszolhat több processzor között is. A megosztott használatú modul címmezőjén belül is lehetnek egy vagy más processzor által privilegizált területek, de lehetnek több processzor által közösen használt szektorok is.

A modularitás irányába haladó fejlődés egy másik eredményét az univerzális építőelemek jelentik. Különböző, gyakran nehezen meghatározható szintű modulokkal a processzorok és memóriák tárgyalása során már találkoztunk. A moduláris hálózatok univerzális építőelemként a processzort alkalmazzák. Példaként a 3.25. ábrán látható MITRA 15/30 gépet említjük meg, melynek mikroprocesszor-modulja a belé tárolt mikroprogramtól függően, központi processzor vagy csatornavezérlő szerepkörét láthatja el.



3.25. ábra.
Moduláris MPK hálózat univerzális processzor elemmel

A redundáns moduláris hálózat tartalék processzor és memóriamodulokat tartalmaz. Meghibásodás esetén a meghibásodott elem programozási úton kiiktatódik, és helyébe automatikusan a tartalékelem lép. A nagymegbízhatóságú moduláris hálózat egy más rendszere ugyancsak egynél több processzort alkalmaz, azonban mindegyiküket állandó jelleggel igénybe veszi. Meghibásodás esetében a meghibásodott elem kiiktatódik, s szerepkörét a maradék elemek látják el. A gépi teljesítmény természetesen ezt követően csökken.

A moduláris hálózatszervezés, melynek elvei és módszerei a könyv megírásának időszakában vannak kialakulóban, előreláthatóan felbontja majd a számítógép máris nehezen felrajzolható, hagyományos kereteit. A gép tervezésének és alkalmazás-technikájának új dimenziói bontakoznak ki, melyek látószögében a hálózatnak már nemcsak funkcionális, hanem technológiai építőelemei is mikroprocesszorok, mikromemóriák és készülék mikrovezérlők; ez utóbbiak — feltehetően — maguk is mikroprocesszorok.

3.2. Az architektúra

Architektúrán, mint említettük, a gépnek a programozó által érzékelt (működési) tulajdonságait, azaz koncepcionális struktúráját értik. Az architektúra leírása során a szerkezetet és a programrendszernek a szerkezethez szorosan tartozó elemeit — esetenként — egységként tekintik. A gépek rendszerezése szempontjából azonban célszerűbb a szerkezeti tulajdonságokat a software tulajdonságoktól elkülöníteni, annál inkább, mivel a későbbiekben a programozási rendszereket már nem általánosságban, hanem kizárólag a mérőrendszerek szempontjai alapján tekintjük át.

A legnagyobb teljesítményű számítógépek architektúráját a megvalósításuk időszakára jellemző tudományos, technikai és technológiai színvonal determinálja. A kisméretű gépek architektúráját a lehetőségek tartományán belül elsősorban a rögzített költség- és méretszint korlátozza. Az igen gyors technológiai fejlődés eredményeképpen, a kezdetben megvont költség- és méretszinten egyre fejlettebb architektúrával rendelkező kisgépek jelennek meg, másrészt a kezdeti architektúrát mind olcsóbban, s mind kisebb térfogatban valósítják meg. A fejlődés leglényegesebb koordinátái tehát: a költség, a méret és a „minimum architektúra”.

3.2.1. A gépek osztályozása

Architektúra tekintetében a gépeket mindenekelőtt programozási szintjük jellemzi:

1. utasításszinten programozott,
2. mikroutasítás-szinten programozott,
3. dinamikusan mikroprogramozott,
4. utasítás- és mikroutasítás-szinten programozott,
5. compiler-orientált (letároló aritmetikájú)

gépek különböztethetők meg. A gépek túlnyomó többsége az első csoportba tartozik, ezért leírásunkban bővebben csak az e kategóriába tartozó gépekkel foglalkozunk.

Közös jellemzők

A gépek architektúráját leíró néhány jellemző független a programozás szintjétől. Ilyen általános jellemzők: az adatformátum, valamint a memóriarekeszek, procesz-szorregiszterek és operátorok bitben számított mérete. A kezdeti korszak gépeiben a rekesz, a munkaregiszter és az adat azonos méretű; a „szóhossz” kifejezést hármas értelemben: a rekesz, a munkaregiszter és az adat méretének jellemzésére használják. Jelenleg a szóhossz ilyen általános jelzőként már nem alkalmazható, mivel számos gépben a regiszterek és a memóriarekeszek mérete nem azonos, ráadásul a gép különböző méretű adatokat is kezelhet. *Szónak* a gép által kezelt legjellegzetesebb adatformátumot nevezve, lehet beszélni egyszavas és többszavas adatról, félszóról, ill. byte-ról. A szó fogalmát ily módon rögzítve a kisgépek az alábbi módon osztályozhatók:

— Szószervezésű gépek: rekeszeik és regisztereik szóméretűek.

— Byte-szervezésű gépek: rekeszméret 8 bit; regiszterméret: 8 vagy 16 bit; adatméret 8 bit és ennek többszöröse.

— Byte-orientált szószervezésű gépek: byte műveleteket is végző szószervezésű gépek.

— Vegyesszervezésű gépek: egyszavas munkaregiszterek; félszavas memóriarekeszek; egyszavas, félszavas és byte műveletek.

Az operátorok a munkaregiszterek méretéhez igazodnak. Számos mérés-technikai orientációjú gépnek közvetlen bitmanipuláló utasításai is vannak.

A fenti osztályozást a tárgykörünkbe tartozó gépek jellegzetes adataival a 3.1. táblázat foglalja össze. A táblázatban sz szó, mr memóriarekesz, pr processzorregiszter méretet, k és n pedig természetes egész számot jelöl. A jellegzetes adatok rovatán a zárójel arra utal, hogy a 18 és 19 bit szóméret ritkán fordul elő. A paritásellenőrzéssel és memóriavédelemmel rendelkező gépek tényleges rekesz- és regisztermérete az effektívénél egy vagy két bittel nagyobb lehet. A gépek jellemzésekor célszerű az effektív méretet venni figyelembe.

Egyes gépekben a processzor összes regiszterének mérete a munkaregiszterek méretével azonos (az állapotjelzők általában kivételt képeznek). Számos esetben azonban az akkumulátortól eltérő méretű regisztereket is találunk. A gépek rendszerezése ebből a szempontból nem nyújt lényeges információt.

3.1. táblázat

-szervezésű	A gép jellemzője	A processzált adatok	Jellegzetes adatok (bit)		
			rekesz	regiszter	szó
Byte-	mr = 1 byte pr = 1 vagy 2 byte	byte; n. byte	8	8, 16	8, 16 24, 32
Szó-	mr = pr = sz	szó; n. szó	12, 16, (18, 19), 24		
Byte-orientált	mr = pr = sz sz = k. byte	byte; szó; n. szó	16	16	16
Vegyes-	mr ≠ pr = sz sz = k. byte k > 1	byte; félszó; szó	16	32	32

Jellegzetes kisgép-architektúrák

Az első kisgépek architektúráját a költségminimum szempontja alapján, az akkori technológiai viszonyoknak megfelelően a szerkezet zsugorítására való törekvés determinálja. A két fő követelmény: 1. kis regiszter- és rekeszméret, 2. minimum regiszterkészlet. A gépek szószervezésűek, utasításaik egységesen szóhosszúságúak.

Az utasításhossz csökkentése (a nagyobb gépekhez viszonyítva): a gép flexibilitását korlátozó elvek bevezetése, az utasítások algoritmikus vertikálisának növelése, továbbá a processzorok megfelelő módosítása révén vált lehetségessé. Az ún. rövidszavas gép címtartalmú utasításai egyetlen memóriahelyet jelölnek meg. Kétooperandusú műveletek azzal a feltételezéssel hajthatók végre, hogy az egyik operandus megállapodás szerinti helyen tartózkodik, és hogy az eredmény a művelet elvégzése után ugyanerre a helyre kerül. A megállapodás szerinti hely a processzor akkumulátora, melynek címét tehát az utasítás implicite tartalmazza. A vertikális itt abban áll, hogy a művelet elvégzését megelőzően az 1. sz. operandust külön adatmozgató utasítással kell az akkumulátorba tölteni; az eredményt külön utasítással kell a memóriába helyezni. A tárolóterületben elért nyereség emiatt nem teljesen arányos az utasítás (rekesz) méretcsökkenésével.

A vizsgálatok tanúsága szerint a memóriavonatkozású utasítások mintegy 80... 90%-ának hivatkozási helye a folyó utasításcímhez viszonyított ± 128 memóriahelyen

belül fekszik. A legtöbb hivatkozási hely tehát rövid, relatív címmel elérhető. Ez a rövid cím már elfér a kezdetben választott 12 biten belül. A memória szerkezeti címét — az abszolút címet — a processzor számítja ki, a relatív címnek valamely megállapodás szerinti mutatóhoz való hozzákapcsolása révén. A relatív címet a mutatóra vonatkoztatva eltolásnak nevezik. Az eltolás értéktartománya: a lap. A gép névtartománya (címeinek értéktartománya) egész számú lapra osztható. Mutatóként valamely lap kezdő címét használva, az eltolás e lapon belüli címet ad meg. Az eltolás a laphatárokon kívüli hivatkozást nem enged meg; lapok közötti kapcsolat közvetett címezéssel létesíthető.

A fenti megfontolások a szöszervezésű, egycímű, egyakkumulátoros, rövidszavas, lapszervezésű architektúrát eredményezték.

A kisgépek számítástechnikai teljesítményének növelése során alakultak ki a két, ill. több akkumulátorral rendelkező processzorok. Egynél több akkumulátort (program által elérhető munkaregisztert) alkalmazva, az akkumulátor(ok) implicit címezsmódja már nem lehetséges; az utasításnak meg kell neveznie, hogy a művelet melyik akkumulátorra irányul.

A processzor regisztereinek szaporodásával nemcsak a munkaregiszterek száma nő, a kisgépekben is alkalmazást nyernek különböző címmutatók: bázis-, index- és peremregiszterek. A program által elérhető regiszterek speciális feladatkörű csoportokba tömörülhetnek, vagy minden felmerülő feladatra használható általános regiszterként szerepelhetnek. Az utasításban explicite megadott regisztereken kívül a gépekben továbbra is találhatunk csak bizonyos utasítások által érintett, implicit címezett regisztereket is (pl. a programszámlálót az ugró utasítások általában implicit címezik).

Az explicite címezett regisztereket az utasításban sorszám nevezi meg. A sorszám és a regiszter szerkezeti címe között direkt kapcsolat van; a sorszám közvetlenül a regiszterre mutat. A kisgépek utasításaiban több, mint két regiszternév ritkán fordul elő.

Az explicite megnevezett regiszter szerepkörét egyrészt a műveleti kód határozhatja meg, másrészt nevének az utasításon belüli pozíciója, harmadsorban pedig a címezsmód. A processzor regisztereinek címtartománya a főmemória címtartományától — általában — független. Az általános regisztereknek egyetlen közös címtartományuk van, amennyiben azonban a processzoron belül több, speciális feladatkörű regisztercsoport (pl. munkaregiszterek, indexregiszterek) található, mindegyik csoport önálló címtartománnyal rendelkezhet. Az általános regisztereket alkalmazó processzor programozási szempontból flexibilisebb, de az utasításokban a regiszterek megnevezése, feladatkörük megjelölése nagyobb helyet foglal el, mint amikor a regiszterek feladatkör szerint csoportosulnak. Mindenekelőtt az explicite címezett regiszterek alkalmazása vonta magával a gépek szóhosszának növelését (12 bitről 16, 18 bitre). Két regiszternek az utasításban való megnevezésével azonban még 16 vagy 18 bit szóhossz mellett is alig marad helye az eltolásnak. Részben emiatt, részben a lapszervezés hátrányainak kiküszöbölése érdekében visszatértek a memória abszolút címezsmódjára, egyszersmind a gazdaságosság követelményeit is figyelembe véve, elvetették az utasításkészlet formai egységének koncepcióját. Így alakultak ki az utasításkészletükben különböző hosszúságú, egy-, esetleg kétcímű utasításokat tartalmazó kisgépek. Később, bizonyos gépcsaládok létrehozása kapcsán, a memória rekeszei és a processzor munkaregisztereinek kongruenciájának követelményét is elejtik.

Az architektúra alakulását befolyásoló elsődleges tényezők mellett bizonyos másodlagos tényezők is hatnak. Nem csupán a számítástechnikai teljesítmény fokozásának, hanem az algoritmikai struktúra finomításának problematikája is állandóan előtérben áll. A tiszta eljárás, az újra kezdhető program követelménye alapján formálódott ki pl. az automatikus letárolót alkalmazó szubrutinkezelési

mód. Az adatstruktúrák processzálására a közvetett mellett a többszintű közvetett címzés mód, az egyszeres indexelés mellett pedig a többszörös indexelés, valamint különböző autoinkrementálási, ill. -dekrementálási lehetőségek jelennek meg.

Az időrealitási követelmények kielégítéséhez az algoritmikai struktúra tisztulása a programok lerövidülésén, egyszerűsödésén keresztül járul hozzá. Az időrealitással kapcsolatos tényezők közül azonban elsősorban a megszakító rendszerek fejlődésére kell rámutatnunk. A kontextus letárolóba helyezése, a kontextusváltás többszörös regiszterkészlet alkalmazásával azok a módszerek, melyek a fejlettebb kisgép-architektúrákat kitüntetik.

A mikroprogramozott vezérlők előretörése a gépi utasítás szintjéről kétirányú — lefelé és felfelé — való eltérést vont maga után. Egyrészt strukturált mikroutasítás-készlettel rendelkező, mikroutasítás szinten programozott gépek, másrészt compiler-orientált architektúrák alakultak ki. Mindkét irányzat mögött — bár egészen különböző aspektusú — programozásökonómiai törekvések hatnak.

A gépi architektúra kibontakozó komplexitásának időszakában megfigyelhető, hogy a legújabb gépek sorában továbbra is megjelenik sok tekintetben a korábbi színvonallal jellemezhető architektúra. Ezek a gépek rendszerint egy-egy számítógépcsaládnak a korábbi tagokkal programozási szempontból kompatibilis, technológiailag tökéletesített egyedei.

3.2.2. Utasításszinten programozott gépek architektúrája

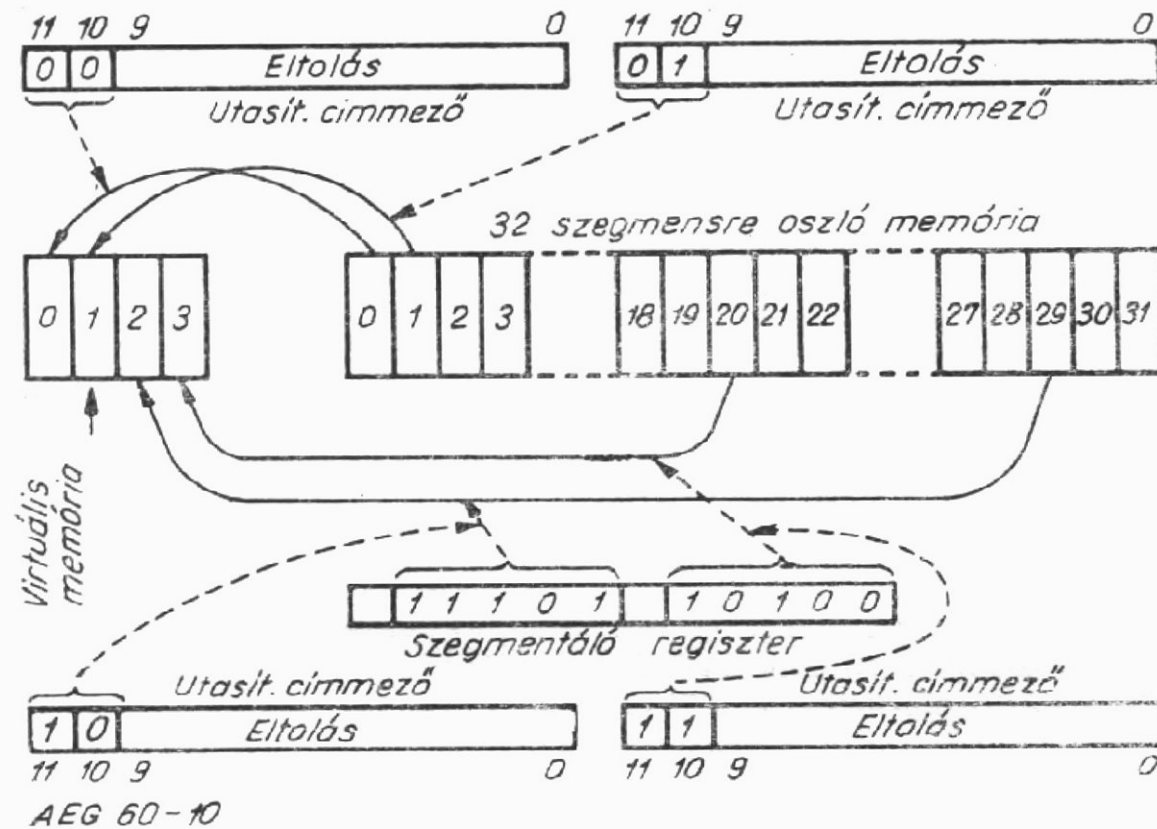
Szószervezésű gépek

A szószervezésű kisgépek jellegzetes képviselői a *12 bitszavas*, egycímes, egyakkumulátoros PDP 8 család tagjai és a velük kompatibilis egyéb gépek (DCC D 112 és D 112H; Honeywell H 112). E csoportot egyszavas utasításformátum, továbbá lapszervezésű címrendszer jellemzi. A címtartalmú utasítások hét eltolásbitje 128 memóriahelyű lapméretet határoz meg. A négy címzés mód: folyó lapra, ill. kezdőlapra vonatkozó, közvetlen és közvetett. A közvetett címzés egyszintű. A névtartomány 4 K. A kezdőlap a többi lap közötti kapocs szerepét játssza; bármely lapról rajta keresztül közvetett címzéssel hívhatók pl. szubrutinok. A kezdőlap további szerepe: utasítással közvetlenül elérhető számlálók, munkaregiszterek, időszaki tárolók, konstansok szolgáltatása. E minimum regiszterkészletű gépcsaládnak nincs indexregisztere. A közvetett címzés nemcsak a laphatárok átlépésére, hanem az indexregiszter pótlására is szolgál. A kezdőlap $10_8 \dots 17_8$ című rekeszeit közvetett címzéssel nevezve meg, e rekeszek tartalma, mielőtt effektív címként felhasználódna, inkrementálódik (autoindex). Az indexregiszter mint hurokszámológó pótlására a feltételes átlépő utasítás használható, mely a címmezőjében megnevezett memóriarekesz tartalmát, mielőtt értékét megvizsgálja, inkrementálja. A 4 K névtartomány átlépésére programozási úton kezelhető blokkregiszterek szolgálnak.

A 12 bitszavas, egycímes, egyakkumulátoros gépek egy más rendszerét (AEG DARK 2, AEG 60—10), kétszavas (2×12 bit) utasítások jellemzik. Az utasítás első szava: operatív rész, a második: címmező. A hat bites műveletkódnak 64 kombinációja lévén, ezek a gépek lényegesen gazdagabb utasításkészlettel rendelkeznek, mint a PDP 8. Indexregiszterként a memória 1...7 című rekesze szolgál. A 12 bites címzóna 4 K területet fog át. A közvetlen, abszolút címzés módon kívül közvetett és programrelatív címzés mód is van. Programrelatív címzés mód használatakor az utasítás címzónájának értéke mint eltolás a programszámláló momentán tartalmához kapcsolódva képez abszolút címet. A címzóna értéktartományát ebben a vonatkozásban, programrelatív lapnak is nevezik. A programrelatív lap helyzete nem rögzített, hanem a programszámláló tartalmával együtt mozog. A 4 K címhatár

átlépése egyrészt relatív címzés, másrészt a névtartomány és a memória címtartományának szegmentálása, szegmentáló regiszter alkalmazása révén lehetséges.

A memória kezelése szegmentálással, a következő módon történik (3.26. ábra). Feltételezzük, hogy a valóságos memória mellett a gépnek a névtartománnyal



3.26. ábra. Memóriakezelés szegmentálással

egybevágó virtuális memóriája is van. Mind a virtuális, mind a valóságos — mondjuk 32 K kapacitású — memória 1 K nagyságú szegmensekre tagolódik. Az utasítás címrésze virtuális címként értelmeződik úgy, hogy tizedik és tizenegyedik bitje szegmensszámot, a többi pedig szegmensen belüli eltolást jelöl. A 00_2 és a 01_2 szegmensszám a virtuális és a valóságos memória két alsó szegmensét állandó jelleggel rendeli egymáshoz (a szegmensszám közvetlenül kapcsolódik az eltoláshoz). A szegmensszám 10_2 , ill. 11_2 értékének utasításbeli megjelenése esetén viszont az eltolás, értelemszerűen a szegmentáló regiszter 6...10, ill. 0...4 bitjeinek tartalmával (mint szegmensmutatóval) egészül ki abszolút címmé. A szegmentáló regiszter az akkumulátorból, külön e célra rendelt utasítással tölthető fel. A programozó dönti el, hogy a virtuális memória 2., ill. 3. szegmenséhez a valóságos memória mely szegmenseit rendeli (természetesen a 0. és 1. kivételével). Többszörös indexelés, valamint többszintű közvetett címzés mód alkalmazható. Az utasításkészlet ún. kváziutasításokat is tartalmaz, melyek formailag a gépi utasításokkal megegyező makroutasítások. Az egyes kváziutasítások műveletkódja az utasításhoz rendelt makroprogramra közvetlenül mutat. A makroprogramok a memóriában előre meghatározott címeken helyezkednek el.

A 16 bitszavas szószervezésű gépek legáltalánosabban ismert képviselői a Hewlett—Packard 2110 család tagjai: 2114, 2115, 2116 és fejlesztett módosulataik, melyeknek a PDP 8-éhoz igen hasonló architektúrájuk van. Jellemzőjük az egységes egyszavas utasításformátum és a lapszervezésű címrendszer. A nagyobb szóhossz (a PDP 8-hoz viszonyítva) az utasítások számának növelését és a lapméret bővítését (1 K) teszi lehetővé. Lényeges újítás az előzményekhez képest: egy akkumulátor helyett két

akkumulátor. Az akkumulátorok implicit címzés módja miatt mindazon utasítások, melyek mindkét akkumulátorra vonatkoznak, duplán szerepelnek. Takarékosági szempontok alapján bizonyos művelettipusok csak az egyik akkumulátorral végezhetők. Így is azonban számos utasítás alternatív, és emiatt a hosszabb műveletkód által nyújtott gazdagabb kombinációs lehetőség nem használható ki optimálisan. Az akkumulátorokhoz, regiszterek közötti műveletek kapcsán alkalmazott explicit címek is tartoznak. Explicit címként a gép névtartományának első két neve (0000₁₆ és 0001₁₆) szolgál. A 2110 családnak nincs sem indexregisztere, sem autoindex lehetősége, autoinkrementáló utasítása azonban van, közvetett címzés módja pedig többszintű. Az újabb kiadású HP 2100 sorozat utasításszinten ugyan kompatibilis elődeivel, de alap-utasításkészletében bizonyos olyan aritmetikai műveletek kétszavas utasításait is tartalmazza, melyek a 2110 sorozatnál csupán opcióként állnak rendelkezésre.

Az egységes utasításformátummal való szakítás megtestesítői a VARIAN 620 család 16 bitszavas szószervezésű gépei (620/i; 620/f, 620/L). Főbb adottságaik sorából az egy- és kétszavas utasításformátumot, az explicite címezhető két akkumulátort és az indexregisztert kell kiemelni. A címzés módok gazdag választéka (közvetlen operandusú; közvetlen abszolút; programrelatív; indexelt; többszintű közvetett) nélkülözhetővé teszi a lapszervezést. Az utasításformátum flexibilitásának eredménye a hatásos utasításkészlet, melynek sajátos tagját képezik a végrehajtó utasítások. (A végrehajtó utasítás meghatározott feltétel mellett a címe által megnevezett helyen található utasítást hajtja végre, a programszámláló kitérése nélkül).

A 18 bit adatméretű szószervezésű gépek közül a VARIAN 622-t kell első helyen említeni, melynek utasításformátumai a 620 család többi tagjaiéval azonosak; az utasítások ezért 18 bitből csupán 16-ot foglalnak el (két bit kihasználatlan marad). A 18 bit előnye: nagyobb adatpontosság.

A 18 bitszavas gépek közül az Electronic Processor, Inc. EPI—118, egyszavas formátumú utasításkészletével bizonyos értelemben kezdetleges, de sajátos architektúrát képvisel. A gép sem indirekt címzéssel, sem indexelési lehetőséggel nem rendelkezik, kétféle (utasításonként változó) lapmérete pedig címrendszerének kezelését megnehezíti. Az egy- és kétszavas utasításokkal egyaránt rendelkező EPI—218 címrendszere gyökeresen eltér az elődétől; programrelatív, indexelt, közvetett és közvetlen címzés módjai elkerülik a lapszervezéssel járó nehézségeket. Mindkét gép specialitása, hogy az adatok nem csak 18-bites egészként, hanem 3-bites byte-okban is processzálhatók (oktális aritmetika).

A lapszervezés sajátos megvalósításmódjával találkozunk a 19 bitszavas Telemechanique T 2000 gépnél. Utasításai egyöntetűen egyszavas formátumúak. A címtartalmú utasítások 12-bites címzónája 4 K, azaz blokkméretű lapot definiál. Címzés módjai: blokkon belüli közvetlen; közvetlen indexelt; közvetett; előindexelt közvetett. A közvetett cím is modifikálható azonban, miáltal többszintű elő- és utóindexelt közvetett címzés mód realizálható. A blokkhatárok (laphatárok) átlépésére a közvetett címzés mód szolgál. A gépnek egyetlen, bővített akkumulátora van.

A szószervezésű 24 bitszavas gépek jellemző példjaként az újabb korból származó GE—PAC 4400-t érdemes bemutatni. Utasításkészletének formátuma homogén, egyszavas; a 14-bites címzónával 16 K memóriaterület címezhető közvetlenül. A memória 1...7 rekesze indexregiszterként szolgál. A memória teljes címtartománya 128 K. A géphez közvetlenül rendelhető memória max. kapacitása 64 K. A 64 K-n felüli memóriaterület más gépekkel közösen használható. A 16 K közvetlenül elérhető memóriaterület határa indexeléssel léphető át. A relatív címzés módja által definiált címrelatív lap, a programszámláléhoz viszonyítva +8191, ill. -8191 memóriahelyek között fekszik. A közvetlen cím mellett a relatív cím is indexelhető. A gép egyetlen, bővített akkumulátorral rendelkezik. Az utasításkészlet bitmanipuláló, valamint kváziutasításokat is felölel.

Byte-szervezésű gépek

A byte-szervezésű gépek alapvető adatformátuma és rekeszmérete: 8 bit. Az egyes gépek munkaregisztereik méretében (és számában), továbbá adatkezelési lehetőségeik tekintetében különböznek egymástól. Jellegzetes képviselőik egyikét, a CII 10010-t, *8 bit akkumulátorméret* jellemzi. Az akkumulátor egy byte méretű bővítő regiszterrel egészül ki. Indexregiszter nincs, a memória névtartománya 64 K. A címrendszer lapszervezésű. A címezsmódok: kezdőlapon belüli közvetlen és közvetett; folyólapon belüli közvetlen; és akkumulátor közvetítésű. A gép egy- és kétbyte-os adatokat processzál. Az utasítások egységesen két byte méretűek és valamennyi azonos formátumú.

A *16 bit akkumulátorméretű* byte-szervezésű gép példája a VARIAN 520/i. Bővítő-regiszterével együtt az akkumulátor max. 32 bit méretű adatot kezelhet. A programozó négy különböző (1, 2, 3, ill. 4 byte) adatméret között választhat. Az adatméret egy e célra szolgáló regiszter tartalmának feltöltésével jelölhető ki. A processzoron belüli összes adatvonal 8 bit széles; több byte méretű adatok processzálása bytesorosan történik. Az utasításkészlet egy, két és három byte hosszúságú utasításokat ölel fel. A címrendszer lapszervezésű (1 K lapméret). A legelső négy lap közvetlenül címezhető. További címezsmódok: lapon belüli közvetlen; indexelt; közvetett a kezdőlapon át. A megszakítórendszer által végrehajtott kontextusváltás céljára két állapotömb (akkumulátor és bővítőregiszter; programszámláló; indexregiszter; túlfolyásjelző; adathosszjelző) áll rendelkezésre.

Egyes byte-szervezésű gépek (pl. BIT 480) nem csak 1...4, hanem szinte korlátlan számú byte-ból álló operandusokat is kezelnek, a bináris aritmetika mellett pedig decimális aritmetika is előfordul. A byte-szervezés jól illeszkedik a decimális aritmetikához, tekintettel arra, hogy két BCD számjegy éppen egy byte-ot foglal el.

Byte-orientált szószervezésű gépek

Byte-orientáltak a szó és byte operandusú műveleteket egyaránt végző szószervezésű gépeket nevezzük. Szóméretük általánosságban 16 bit. Közülük egyesek (pl. Honeywell H 316, H 516) byte-műveletei a processzor regisztereire szorítkoznak. Byte-műveleteik pl: az akkumulátor két byte-jának felcserélése, az akkumulátor egyik (jobb vagy bal oldali) byte-jának törlése és ehhez hasonlók. Fejlettebb képviselőik a szószervezésű memória rekeszeinek tartalmát is képesek byte-onként kezelni. A PDP 11 byte-műveleteinek köre pl. csaknem az összes adatprocesszáló utasításra kiterjed, a Lockheed SUE utasításkészlete pedig a PDP 11-nél hiányzó byte-operandusú aritmetikai műveleteket is tartalmaz.

A PDP 11 az eddig ismertett gépektől regisztereinek készletével, címrendszere sajátosságával és utasításkészlete kiterjedtségével (pl. kétcímes utasítások) is eltér. Regisztertömbjének (16 bit méretű) regisztereinek közül nyolc (RO—R7) programozási úton általános regiszterként kezelhető; (RO—R5), a programozó választása szerint akkumulátor, címmutató, autoinkrementáló, ill. autodekrementáló peremmutató, valamint indexregiszter feladatát láthatja el. A további két regiszter egyike: programszámláló (jelölésszinonímái: R7, ill. PC), a másik: szubrutinhívás és megszakítás-kérés kezelése kapcsán szerepet játszó letároló automatikus peremmutatója (jelölésszinonímái: R6, ill. SP). Az architektúra egyik érdekessége, hogy bizonyos korlátozással ugyan, de e két utóbbi regiszter is hozzáférhető programozási úton; pl. az összes (RO—R7) regiszter kezelhető peremmutatóként. A gép névtartománya 64 K-ig terjed. A készülékeknek nincs önálló címtartománya; a készülékcímek a 64 K névtartomány legfelső szektorát foglalják le. Önálló címtartománya van ezzel szemben a processzor regisztereinek (RO—R7).

A PDP 11 címrendszere programrelatív közvetlen, valamint regisztercímzésű címzésmodot ismer. A regisztercímzés sajátossága, hogy az utasítások végrehajtása során, az effektív címképzés mechanizmusában a processzor regiszterei alapvető szerepet játszanak. Az aktuális regiszter nevét és szerepét az utasításban címmeghatározó zóna írja le. A regisztercímzésű utasítások formátuma változó; hosszuk a címmeghatározó zóna (zónák) jelentésétől függően egytől három szóig terjedhet. A címmeghatározó zónában szereplő regiszter típusától függően általános regisztercímzés és programszámláló (PC) címzés különböztethető meg. A címmeghatározó zóna szerkezetét, a regisztercímzésű utasítások címzés módjait a 3.2. táblázat mutatja be.

A PDP-11 regisztercímzésű utasításainak címzés módjai

3.2. táblázat

		<table border="1" style="display: inline-table; vertical-align: middle;"> <tr> <td style="width: 15px; height: 15px;"></td> <td style="width: 15px; height: 15px;"></td> <td style="width: 15px; height: 15px;"></td> <td style="width: 15px; height: 15px;"></td> <td style="width: 15px; height: 15px;"></td> <td style="width: 15px; height: 15px;"></td> </tr> <tr> <td style="text-align: center;">Mód</td> <td></td> <td style="text-align: center;">Regiszterszám</td> <td></td> <td></td> <td></td> </tr> </table>								Mód		Regiszterszám				Az utasítás 6-bites címmeghatározó zónája		
Mód		Regiszterszám																
	Mód	Címzés mód megnevezése	A regiszter szerepe	Szimbólum														
Általános regisztercímzés	0	Regiszter közvetlen	akkumulátor	R														
	1	Regiszter közvetett	címmutató	@ R vagy (R)														
	2	Autoinkrementáló	peremmutató	(R) +														
	3	Autoinkrementáló közvetett	peremmutató	@(R) +														
	4	Autodekrementáló	peremmutató	-(R)														
	5	Autodekrementáló közvetett	peremmutató	@-(R)														
	6	Indexelt	indexregiszter	X(R)														
	7	Előindexelt	indexregiszter	@X(R); @(R)														
Programszámláló (PC) címzés			<table border="1" style="display: inline-table; vertical-align: middle;"> <tr> <td style="width: 15px; height: 15px;"></td> <td style="width: 15px; height: 15px;"></td> <td style="width: 15px; height: 15px;"></td> <td style="width: 15px; height: 15px;"></td> <td style="width: 15px; height: 15px;"></td> <td style="width: 15px; height: 15px;"></td> </tr> <tr> <td style="text-align: center;">Mód</td> <td></td> <td style="text-align: center;">PC(7)</td> <td></td> <td></td> <td></td> </tr> </table>								Mód		PC(7)				Az utasítás címmeghatározó zónája	
	Mód		PC(7)															
	2	Közvetlen operandusú	peremmutató	# n														
	3	Abszolút	peremmutató	@ # A														
6	Relatív	bázisregiszter	A															
7	Relatív közvetett	bázisregiszter	@ A															

Regiszter közvetlen címzés módnál a megnevezett regiszter akkumulátorként szerepel, mint ilyen vagy operandust tartalmaz, vagy adatátviteli művelet forrás/rendeltetési regisztereként szolgál.

Regiszter közvetett címzés mód esetében a megnevezett regiszter tartalmát a gép abszolút címként használja fel.

Autoinkrementáló címzés mód használatakor a gép a megnevezett regiszter tartalmát először abszolút címként alkalmazza, majd — attól függően, hogy az utasítás byte- vagy szóvonatkozású-e — eggyel, ill. kettővel inkrementálja.

Autoinkrementáló közvetett címzés módnál a regiszter tartalmát a gép először közvetett címként alkalmazza, majd kettővel növeli.

Autoinkrementáló címzés alkalmazásakor a megnevezett regiszter tartalma először csökken (eggyel, ill. kettővel), majd abszolút címként szolgál.

Autoinkrementáló közvetett címzés esetében a regiszter tartalma kettővel csökken, majd közvetett címként szolgál.

Az általános regisztercímzés módozatai közül a 0...5 mindig egyszavas utasítást tételez fel. A 6, 7 azonban címbővítő vonzattal rendelkezik.

A címbővítőben elhelyezkedő szám (X) a regisztertartalommal indexelve képez címet. Az indexelt címzés mód közvetlen címet, az előindexelt címzés mód előindexelt közvetett címet definiál. A kétoperandusú — két címmeghatározó zónát tartalmazó — utasítások adott esetben tehát két címbővítővel is kiegészülhetnek, melyek közül sorban az első a forráscímhez, a második a rendeltetési címhez tartozik.

A PC címzés módok mindegyike bővített címet tételez fel. A gép a címbővítő rekesz tartalmát közvetlen operandusú címzés mód esetén operandusként, abszolút címzés mód esetén abszolút címként, relatív címzés mód kapcsán pedig a programszámláló tartalmához kapcsolódó eltolásként értelmezi. A relatív címzés mód két módosulata közvetlen, ill. közvetett címet definiál. A PC címzés tulajdonképpen az általános regisztercímzés 2, 3, 6 és 7 módjának kiterjesztése a programszámlálóra.

Az összetartozó autoinkrementáló és autodekrementáló címzés módok segítségével a főmemória bármely területe letárolóként kezelhető.

Vegyesszervezésű gépek

A vegyesszervezésű gép regisztermérete eltér a (> 1 byte) memóriarekesz méretétől. Az INTERDATA 7/32 esetében pl. a rekesz 16 bit, a munkaregiszterek 32 bit méretűek. A gép byte-orientált; szómérete 32 bit, de 16 bit félszavas és byte-műveletei is vannak. A szokatlanul széles regiszter azokat a nehézségeket hivatott kiküszöbölni, melyek a feldolgozási adat hosszához mérten keskeny munkaregiszter alkalmazásával állnak elő.

Tizenhat általános (32 bit) regiszter két készlete áll rendelkezésre, gyors kontextusváltás céljából. A regiszterek, a nulljelű kivételével munkaregiszter (R) vagy index-regiszter (X) szerepét — a programozó választása szerint — tölthetik be; a nulljelű csak munkaregiszter lehet. A különböző méretű utasításformátumok egyike 24-bites címzónával rendelkezik, ami 0...16 777 215 névtartománynak felel meg. A ténylegesen kihasznált névtartomány azonban ennél kisebb: 1 048 576 byte (524 288 félszó).

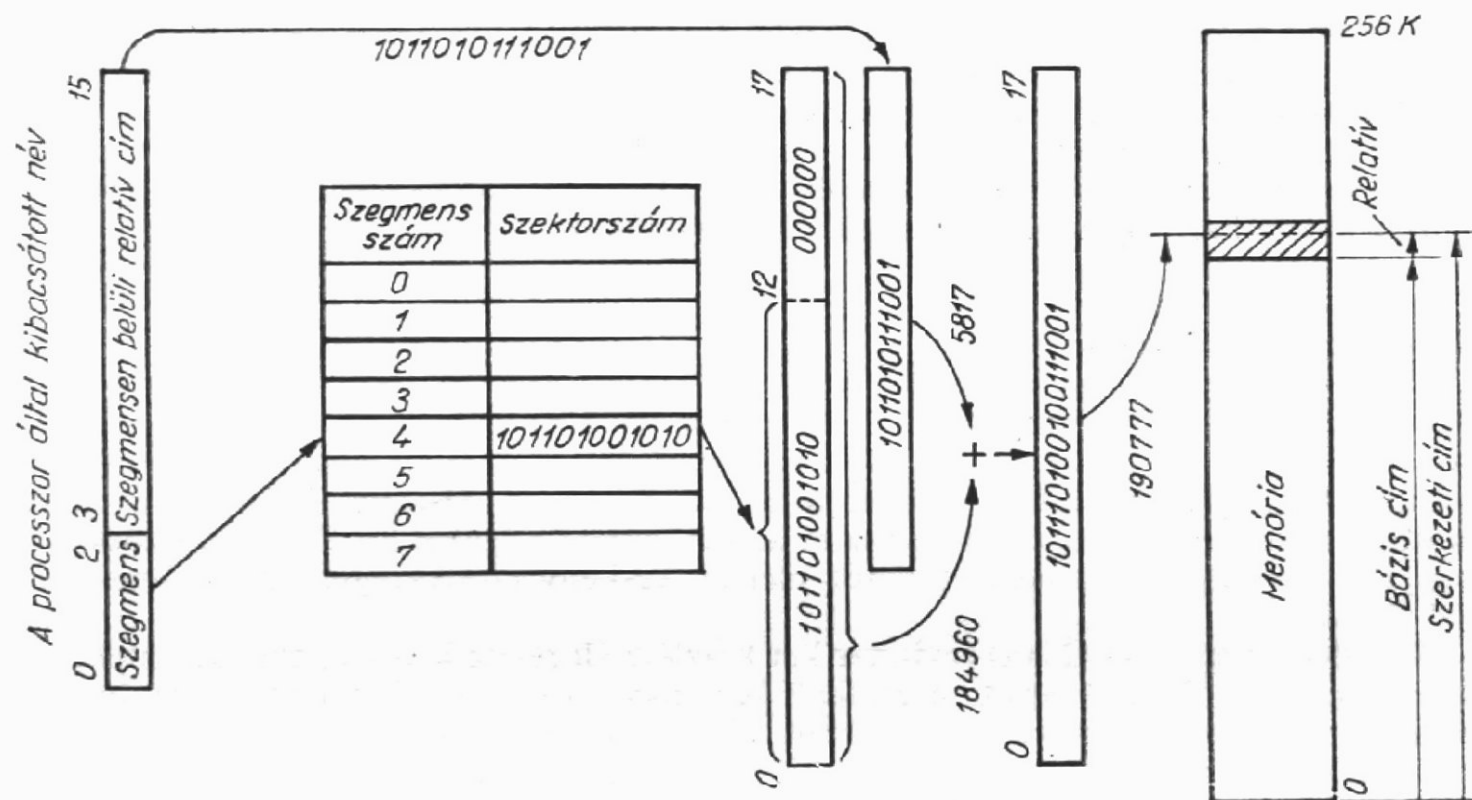
Virtuális memória

A számítógép címrendszere meghatározza az utasítás által elérhető memóriarekeszek, a programban megnevezhető memóriahelyek maximális számát: a névtartományt. A névtartománynál lényegesen kisebb címtartományú főmemória, kisegítő tároló-

val (lemez, dob, félvezető) a teljes névtartományra kiterjedő virtuális memóriává egészíthető ki. Másrészt a névtartománynál lényegesen nagyobb címtartományú főmemória bizonyos eszközök igénybevételével a névtartomány által definiált virtuális memóriaként kezelhető. Az előbbi eljárást a címtartomány kiterjesztésének, az utóbbit zsugorításának nevezzük.

Kisgépek esetében a címtartomány kiterjesztése a lapigénylés (page demand) módszerével történik. A névtartomány (a program) és a memória címtartománya egybevágó méretű lapokra oszlik fel. A teljes névtartományra kiterjeszhető program a kisegítő tárolón helyezkedik el; végrehajtása során az épp aktuális lapok a főmemóriába másolódnak át, miközben a nem használt programlapok a kisegítő tárolóba kerülnek vissza. Menetközben a főmemóriában a programlapok állandóan cserélődnek; a memória adott lapjához időben egymás után más-más programlapok rendelődnek. Az automatikus lapcserélgetés révén a programozó a címrendszert transzparensnek látja, a teljes névtartományra kiterjedő főmemóriát érzekei. Ez az érzékelt, de a valóságban nem levő főmemória: a virtuális memória.

A memóriaszegmentálással képzett virtuális memória általánosabb megoldását nyújtja a PDP—11/45, melynek 64 Kbyte névtartománya 256 Kbyte szerkezeti (memória) címtartománnyal áll szemben. A névtartomány nyolc egyenlő (8 K byte nagyságú) szegmensre, a címtartomány pedig 4096 (egyenként 64 byte nagyságú) szektorra oszlik fel. Minden szegmenshez egy (a teljes névtartományhoz tehát összesen nyolc) szegmentáló regiszter tartozik. A szegmensek bármelyike a memória címtartományára szegmentáló regisztere segítségével képezhető le, a 3.27. ábrán illusztr-



3.27. ábra.

Virtuális memória képzése szegmentálással és szektorokra osztással

rált módon. A processzor által kibocsátott név a 0...2 zóna szegmensszámára és a 3...15 zóna relatív címére bomlik. A szegmensszám kiválasztja a neki megfeleltetett szegmensregisztert (pl. a 4. sorszámút); a regiszter az általa tartalmazott szektorszámot a címképzőbe továbbítja. A címképző a szektorszámból nullák hozzáadásával 18-bites báziscímet, majd a relatív cím hozzáadásával szerkezeti címet képez. A névszegmens a memóriaterületen bázisrelatív szegmensként jelenik meg. A bázis

64 byte-os inkrementumokkal a memória tetszőleges pontján jelölhető ki, a relatív cím pedig pozitív vagy negatív előjellel is értelmezhető úgy, hogy a leképzett szegmens a báziscímhez viszonyítva előre vagy visszafelé is terjedhet. A relatív cím finomszerkezete a valóságban az itt bemutatottnál összetettebb, a szegmentáló regiszter pedig a szektorszámokon kívül különböző kiegészítő adatokat is tartalmaz. A processzornak három programállapota van, és mindhárom állapothoz a szegmentáló regiszternek két-két (nyolc regiszterből álló) készlete tartozik; állapotonként egy regiszterkészlet a program, egy az adatkímek leképzéséhez. A szegmentáló regiszterek feltöltése, tartalmuk módosítása programozási úton végezhető, s így a szerkezeti címtartomány bármely területe elérhető segítségükkel.

Szubrutinkezelés

A kisszámítógépek körében legáltalánosabban ismert, hagyományos szerkezetű szubrutin üres memóriahellyel (Y) kezdődik, ezt követi a szubrutin törzse (tulajdonképpen rutin), majd az üres memóriahelyet közvetett címmel megnevező szubrutin-visszatérő utasítás ($SRV\ I, Y$), mely azonos a feltétel nélküli ugró utasítással. A szubrutin speciálisan erre rendelt utasítással ($SRH\ Y$) hívható, melynek címrésze a szubrutin üres memóriahelyére mutat. Az $SRH\ Y$ végrehajtása során, a vezérlés a hívó program X címéről a szubrutin $Y + 1$ címére adódik át, miközben a program soron következő utasításának $X + 1$ címe az Y helyen tárolódik. E feltételek mellett az $SRV\ I, Y$ utasítás éppen az $X + 1$ effektív ugráscímet definiálja, s ilyenképpen vezet vissza a főprogramra.

A szubrutinkezelésnek ez a módja nem elégíti ki a tiszta eljárás követelményeit, mivel a hívás a szubrutin programterületén a visszatérési cím bevitelével változást idéz elő. Következésképpen a szubrutin nem újrakezdhető és rekurzív sem lehet; a szubrutint két vagy több, egymástól független program közösen nem használhatja.

A tiszta eljárás követelménye — szubrutin kezeléssel kapcsolatban — pl. címmutató regiszter és letároló alkalmazásával teljesíthető (PDP 11). Az $SRH\ R, Y$ alakú szubrutinhívó utasítás Y címe (címmeghatározója) a hívott szubrutin elsőként végrehajtandó utasítására mutat, R pedig azt az általános regisztert (címmutatót) nevezi meg, amelyikbe a visszatérési cím kerül.

Külön utasítás szolgál a szubrutin-visszatérésre is ($SRV\ R$), mely egyedül a visszatérési címet tartalmazó általános regisztert tünteti fel.

Az $SRH\ R, Y$ utasítás a következőket hajtja végre:

1. $R \downarrow$: Az R tartalma a letárolóba kerül.
2. $(PC) \rightarrow R$: A programszámláló aktuális ($X + 1$) értéke R -be kerül.
3. $Y \rightarrow PC$: A címmeghatározó által definiált cím a programszámlálóba kerül; a programvégrehajtás Y -nál folytatódik.

Az $SRV\ R$ végrehajtása kapcsán:

1. $(R) \rightarrow PC$: A visszatérési cím R -ből a programszámlálóba kerül.
2. $\uparrow R$: Az R eredeti tartalma a letárolóból visszatér.

A szubrutin ily módon való processzálása a tiszta eljárás követelményeinek eleget tesz: A szubrutin újra kezdhető (akár önmaga kezdeményezése alapján is); több program közösen használhatja, nem csak kezdő utasítása, hanem bármely közbeni pontja is hívható; egymásba ágyazott szubrutinok láncá képezhető egyugyanazon címmutató regiszter alkalmazásával.

A szubrutin hagyományos szervezés- és kezelésmódjának további, eddig még nem említett nehézsége az esetleges operandus(ok)nak a szubrutinhoz, az eredménynek

a szubrutint használó programhoz való átvitelével kapcsolatos. Az operandus helyének a szubrutin számára ismertnek kell lennie. Hívás előtt az operandust vagy operandusokat a hívó program a megállapodás szerinti helyre kell, hogy vigye. Egyakkumulátoros gépeknél, s ha csak egyetlen operandusról van szó, ez a hely az akkumulátor; a szubrutin a hívó program számára szóló eredményt is az akkumulátorban hagyhatja vissza. Operandus(ok) és eredmény(ek) elhelyezésére általánosan alkalmazzák a hívó program SRH utasítását követő memóriahelyeket. Ez esetben a szubrutin üres (címmutató) rekeszébe kerülő cím nem a visszatérési pontra, hanem az első operandusra mutat. Az operandusok e cím inkrementálásával tapogathatók végig. A cím, melynek értéke eközben változik, az összes operandus letapogatása, az eredmény összes elemének elhelyezése után végül a visszatérési pontra mutat. A címmutató memóriarekesz tartalmának inkrementálása elkerülhető, utóindexelt közvetett címezsmód alkalmazásával, de ezzel nagyon kevés kisgép rendelkezik. A PDP 11 szubrutinkezelési rendszerében az operandusok és eredményelemek helyei ugyancsak SRH utasítás után következnek. Ezek azonban rögzített elérési sorrend esetén az (R)+ autoinkrementáló címezsmóddal, a címmutató regiszteren át közvetlenül hozzáférhetőek. Az operandus—eredmény tárolóterület tetszőleges pontja az X(R) ndexelt címezsmóddal ugyancsak egyszerűen elérhető.

Irodalom a 3. fejezethez

- [1] *Bell, C. G.—Newell, A.:* Computer structures: readings and examples. McGraw-Hill, Jun. 1971, 668 old.
- [2] *Hill, F. J.—Peterson, G. R.:* Digital systems: Hardware organization and design. John Wiley and Sons, Inc. 1973, 481 old.
- [3] Журавлев, Ю. П.: Системное проектирование управляющих ЦВМ. Советское радио, 1974, 368. old.
- [4] Константинов В. П. — Сухачев, В. А.: Структурные особенности малых ЭМВ. Приборы и системы управления, № 11, 1969. 6...10. old.
- [5] *Kaenel, R. A.:* Minicomputers — a profile of tomorrow's component. IEEE Trans. Vol. AU—18, Dec. 1970, 354...379. old.
- [6] *Bell, C. G.—Grason, J.:* The register transfer module design concept. Computer Design, May 1971, 87...94. old.
- [7] *Bell, C. G.—Eggert, J. L. — Grason, J. — Williams, P.:* The description and use of register-transfer modules (RTM's). IEEE Trans. Vol. C—21, May 1972, 495...500. old.
- [8] *Grason, J.—Bell, C. G. — Eggert, J.:* The commercialization of register transfer modules. Computer, Oct. 1973, 23...27. old.
- [9] *House, D. L.:* Micro level architecture in minicomputer design. Computer Design, Oct. 1973, 75...80. old.
- [10] *Fantauzzi, F. G.:* Minicomputers and microprogramming: A general approach. Nov. 1971. Dept. of Mat. The University of Iowa, Project. No 0014—68—A—0500.
- [11] *Janson, P.:* Common bus structure for minicomputers improves I—O flexibility. Control Engineering, Jan. 1971, 50...53. old.
- [12] *Chertkow, D.—Cady, R.:* Unified bus maximizes minicomputer flexibility. Electronics, Dec. 21. 1971, 47...52. old.
- [13] *Jármai F.—Szakállas Cs.:* Sínrendszerű moduláris számítógépek. Infelior Közlemények, 6. sz. 95...122. old.
- [14] *Wilkes, M. V.:* The present status and potential of microprogramming. 1971 IEEE Internat. Convention Digest, 130...131. old.
- [15] *Davies, P. M.:* Reading in microprogramming. IBM Syst. J. No.1. (1972) 16...40. old.
- [16] *Strickland, D.:* Understanding microprogramming. Instrum. and Control Syst., 1973, 69...70. old.
- [17] *Ramamoorthy, C. V.—Tsuchiya, M.:* A study of usermicroprogrammable computers. AFIPS 36, Spring Joint Computer Conf. 1970, 165...181. old.
- [18] *Strout, F. D.:* Microprogramming in the hierarchy of peripheral control. Fifth Annual 1971 IEEE Internat. Comput. Soc. Conf. Digest, No. 71C41—C, 111...112. old.
- [19] *Roberts, W.:* Microprogramming concepts and advantages as applied to small digital computers. Computer Design, Nov. 1969, 147...150. old.

- [20] *Archdale, D. E. W.*: Mikroprogrammierung — praktische Anwendung bei Minicomputern. *Computer-Praxis*, 9. 1972, 267...268. old.
- [21] *Dollhoff, T. L.*: Microprogrammed control for small computers. *Computer Design*, May 1973, 91...97. old.
- [22] Design of microprogrammable system. Scientific Micro Systems, Inc. Application Notes 0052, July 1973.
- [23] *Priel, U.—Holland, P.*: Application of high speed programmable logic array. *Computer Design*, Dec. 1973, 94...96. old.
- [24] *Mrazek, D.—Morris, M.*: PLAs replace ROMs for logic designs. *Electronic Design* 22, Oct. 25. 1973, 66...70. old.
- [25] *Laliotis, T. A.*: Main memory technology. *Computer*, Sept. 1973, 21...27. old.
- [26] *Knoke, P.*: An analysis of buffered memories. *Proc. 2nd Hawaii Internat. Conf. on Syst. Sci.* 1969, 397...400. old.
- [27] *Smiley, Ch. R. Jr.*: Digital data systems work faster when a storage buffer is used. *Electronic Design* 24, Nov. 23, 1972, 150...154. old.
- [28] *Bell, C. G.—Casasent, D.*: Implementation of a buffer memory in minicomputers. *Computer Design*, Nov. 1971, 83...89. old.
- [29] *Hoagland, A. S.*: Mass storage: past, present and future. *Computer*, Sept. 1973, 29...37. old.
- [30] *Davis, S.*: Selection and application of semiconductor memories. *Computer Design*, Jan. 1974, 65...77. old.
- [31] *Carli, G. — Marcon, J. Jr. — Rosborough, J. R.*: Read-Only Memory loads process computer. *Control Eng.* Febr. 1969, 89...91. old.
- [32] *Howard, J. A.—Pfeifer, L.*: An ROM bootstrap loader for small computers. *Computer Design*, Oct. 1970, 95...97. old.
- [33] *Percival, R.*: ROMs are versatile in digital systems. *Electronic Design* 12. June 8, 1972, 66...71. old.
- [34] *Beuter, R.*: Einführung in die Halbleiterspeichertechnik. *Grundig Technische Informationen*, 4, 1973, 236...238. old.
- [35] *Lapidus, G.*: Forum on FAMs. *IEEE spectrum*, July 1973, 50...54. old.
- [36] PDP 8/E and PDP 8/M small computer handbook. Digital Equipment Corp. 1972.
- [37] PDP 11 handbook. Digital Equipment Corp. 1969.
- [38] *Bell, G.—Cady, R. — McFarland, H. — Delagi, B. — O'Laughlin, J. — Noonan, R.*: A new architecture for mini-computers — The DEC PDP 11. *Spring Joint Computer Conf.* 1970, 657...675. old.
- [39] Honeywell Bull Computer Kompendium. 5. Auflage, Juli, 1972.
- [40] A pocket guide to Hewlett-Packard Computers. No 5950—8313 /7/70.
- [41] A pocket guide to interfacing HP computers. No. 5950 —8718 /4/70.
- [42] A pocket guide to the HP 2100 A computer. No. 5951—4423/1972.
- [43] Lockheed Electronics SUE computer handbook. No. 5M 4/72.
- [44] MBM (M. B. Metals Limited) — K 202 modular computer system-handbook.
- [45] The value of power... GA SPC—16/40, Fourth edition, 1974. General Automation, Inc.
- [46] MITRA 15 manuel de presentation. No 4029 P1/FR, Compañie internationale l'informatique, 1972.
- [47] VARIAN DATA 620/i computer manual. No 605—A, 4/68 50.
- [48] VARIAN 520/i computer handbook. No 520, 11/69 20.
- [49] Microprogramming handbook. Microdata No 30M 4/72.
- [50] INTERDATA User's manual. No 29—261Ro2, 1974.
- [51] GRI-99 System reference manual. No 08—73—200. 1972.
- [52] English, W.: How to use the Nova and the Supernova. Data General Corp. 1969.
- [53] 6135 computer system reference manual. EMR No PP0200—023 EMR computer, 1972.
- [54] 6145 computer system reference manual. EMR. No PP1300—001G. EMR computer 1972.
- [55] GE PAC 4400 system summary manual. GET—6469, General Electric, 1973.
- [56] *Dürr, D.—Hirsch, W.*: Das Prozessrechnersystem AEG 60—10 und seine Einsatzmöglichkeiten, *Tech. Mitteilungen AEG-Telefunken*, 1968. 1. Beih. Datenverarb., 5. 38...44. old.
- [57] *Dinman, S. B.*: The direct function processor concept for system control. *Computer Design*, March 1970. 55...60. old.
- [58] *Wallis, P. J. L.*: The INTERDATA series. *Data processing Magazine*, Febr. 1969. 32...50. old.
- [59] *Michels, J.*: The mega-mini succeeds the modelt. *Datamation*, Febr. 1974, 71...74. old.
- [60] *Lackner, A.*: Systemstruktur und Aufbau des Siemens Prozessrechners 320. *Siemens-Zeitschrift*, Okt. 1971, 776...779. old.
- [61] *Dittman, J.*: Struktur und Eigenschaften der Zentraleinheit des Prozessrechners 330. *Siemens-Zeitschrift*, Mai 1973, 359...364. old.

- [62] *Offer, U.—Schneider, U.*: Siemens Prozessrechner 310. Siemens-Zeitschrift, Heft 9, 1974, 675...679. old.
- [63] *Bogdány J.*: A TPA—70 kisszámítógép-család. Mérés és Automatika, XXI. évf. 10 sz. 1973, 376...379. old.
- [64] *Német P.*: Az R-10 (ESZ-1010) modell kialakításának rendszertechnikai elvei. Információ Elektronika 3. 1973, 166...172. old.
- [65] *Гришин, А. И.*: Новые средства управляющей вычислительной техники. Приборы и системы управления, №. 3, 1973. 16...17. old.
- [66] *Забара, С. С. и др.*: Основные характеристики модели М—4030 АСВТ—М. Приборы и системы управления, №. 11, 1974. 17...18. old.
- [67] *Schultz, G. W. — Holt, R. M. — McFarland, H. L. Jr.*: A guide to using LSI microprocessors. Computer, June 1973, 13...19. old.
- [68] *Holt, R. M.—Lemas, M. R.*: Current microcomputer architecture. Computer Design. Febr. 1974. 65...73. old.
- [69] *Davis, S.*: A fresh view of mini- and microcomputers. Computer Design, May 1974, 67...79. old.
- [70] *Csákány A.,—Vajda F.*: Mikroszámítógépek. Műszaki Könyvkiadó 1976.
- [71] *Langley, F. J.*: Small computer design using microprogramming and multifunction LSI arrays. Computer Design, Apr. 1970, 151...157. old.
- [72] *Langley, F. J.*: The universal function unit concept for computing applications. Computer Design, Apr. 1972. 87...91. old.
- [73] *Genke, R. M.*: Static and dynamic control memory in microprogrammable computers. 1972 IEEE Internat. covention digest, Paper 6CI. 3/1—2.
- [74] *Eggert, J. L.*: PDP 16 functional computer concepts. Fifth Annual 1971 IEEE Internat. Comp. Soc. Conf. No 71 C41—C, 107...108. old.
- [75] *Parrish, E. A. Jr. — Chang Lee, Y.*: A microcomputer preprocessor /postprocessor for analog signals. IEEE Trans. Vol. IECI-21, No 1. Febr. 1974, 38...41. old.
- [76] *Teicher, S.—Hughes, L.*: Implementation of a hardware floating point processor. Computer Design, May 1972, 122...128. old.
- [77] *Dobbins, R. W.*: A hardware floating point processor for a minicomputer. Symposium on minicomputers, South Africa, 1973.
- [78] Fast Fourier transform arithmetic unit 5471A, HP No 5952—0683, 1971.
- [79] Fourier processor 5470A, HP No 02—5952—0682, 1971.
- [80] *Cole, L. S.*: Preprocessor for remote supervisory control stations. IEEE Transact. Vol. IECI-20, No 1. Febr. 1973, 9...11. old.
- [81] *Hinsenkamp A. — Hubert B.*: Gyors lebegőpontos műveletvégző egység az R-10 számítógép-hez. Információ Elektronika, 2, 1974, 87...90. old.
- [82] Systems design handbook. Modular computer systems, 1974.
- [83] *Arbuckle, W. L.—Mattson, R. C.*: Marco modularity: a design concept to end computer generation gaps. Computer Design, Aug. 1970.
- [84] *Loréтан, R. P.—Coakley, F. P.*: Detailed proposals for multicomputer bus system. University of Essex, Report No 74, Dec. 1972.
- [85] *Larkin, R.*: A minicomputer multiprocessing system. Proc. 1971 Computer Designer's Conf., 231...235. old.
- [86] *Swoboda, J.*: Was ist pipelining? Elektron. Rechenanl. 2, 1975, 80...83. old.
- [87] *Wulf, W. A.—Bell, C. G.*: C.mmp-A multi-mini-processor. Fall Joint Comp. Conf. 1972, 765...777. old.
- [88] *Shar, L. E.—Davidson, E. S.*: A multiminiprocessor system implemented thorough pipelining. Computer, Febr. 1974, 42...51. old.
- [89] *Riley, W. B.*: Minicomputer networks — a challenge to maxicomputers? Electronics, March 29, 1971, 56...62. old.
- [90] *Hamer Hodges, K. J.*: A fault-tolerant multiprocessor design for real-time control. Computer Design, Dec. 1973, 75...81. old.
- [91] *Young, N. F.*: Distributed computer systems, Automation, Oct. 1969.
- [92] *Büttner, P.—Chaoui, A.*: Hierarchische computersysteme. Technische Rundschau, Nr 43, 20. Oct. 1972, 41...45. old.
- [93] *Bell, C. G.*: Minicomputer architecture, description and design. 1971 IEEE Internat. Convention Digest, 134...135. old.
- [94] *Riberio, S.*: Talking to the minicomputer. IEEE Transact. Vol. IECI-18, No. 2, May 1971, 67...72. old.
- [95] *Seligman, L.*: LSI and minicomputer system architecture. Spring Joint Comput. Conf. 1972, 767...773. old.
- [96] *Kondela, J. Jr.*: The past, present, and future of minicomputers: a scenario. Proc. IEEE, Vol. 61, No. 11, Nov. 1973, 1526...1534. old.
- [97] *Hakozaki, Katusya, et al.*: Design and evaluation system for computer architecture. National Computer Conf. 1973, 81...86. old.

- [98] *Leis, Ch. T.*: Minicomputer hardware architecture. Proc. IEEE, Vol. 61, No 11, Nov. 1973, 1535...1538. old.
- [99] *Burns, R.—Savitt, D.*: Microprogramming, stack architecture ease minicomputer programmer's burden. Electronics, Febr. 15. 1973, 95...101. old.
- [100] *Boutwell, E. O. Jr.*: Comparing the compacts. Datamation, Dec. 1965, 61...73. old.
- [101] *Hanlon, A. G.*: Content-addressable and associative memory systems. IEEE Transact. Vol. EC-15. No. 4, Aug. 1966, 509...521. old.
- [102] *Журавлев, Ю. П.—Забубенов, В. Н.*: Выбор системы форматов команд управляющей ЭЦВМ. Автоматика и вычислительная техника 2. 1973. 71/76. old.
- [103] Virtual memory system for minicomputers. Computer Design, March 1973, 92...95. old.
- [104] *Chase, T. D. — Glorioso, R. M.*: Virtual memories for mini-computers. Proc. of the ACM. — Annual Conf. Aug. 1972, 6...11. old.
- [105] *Glorioso, R. M.—Chase, T. D.*: Design of virtual memory for small computers. Computer Design, Dec. 1973, 67...72. old.
- [106] *Townsend, J. F.*: Overlaid memory simplifies programs, has hidden nooks for diagnostics. Electronics, April 26, 1973, 116...119. old.
- [107] *Wojtkowak, H.*: Strukturelle Abhängigkeiten in einem virtuellen Speicher. Elektronische Rechenanlagen, Heft 1, 1974, 26...32. old.
- [108] *Sweet, W. B.*: Addressing technique unlocks minicomputer's extended memory. Electronics, Nov. 8, 1973, 107...112. old.
- [109] *Riley, W. B.*: 32-bit mini addresses megabyte directly. Electronics, Sept. 27. 1973.

4.

Kapcsolat- és csatlakozásmódok

4.1. Számítógép—készülék kapcsolat

4.1.1. Csatornarendszerek

Átvitel mód és útvonal

A készülék és a gép közötti forgalomban: 1. programkapcsolatú és 2. programfüggetlen átvitel mód különböztethető meg. A programfüggetlen átvitel módot, esetenként autonóm átvitel módnak (ADT— Autonomous Data Transfer) is nevezik, mely elnevezés, közvetve ugyancsak a programtól való függetlenségre — autonomiára — utal. Programkapcsolatú átvitel módban minden egyes adat átvitele utasítás alapján történik. Programfüggetlen átvitel módban az adat átvitelét a készülékvezérlő (a memóriavezérlő vagy csatornavezérlő, esetenként a processzor bevonásával) bonyolítja le. Adaton itt az átvitel rögzített formátumú alapelemeként szolgáló, szó- vagy byte (karakter) -szélességű, bináris vektort értünk.

A programkapcsolatú átvitel útvonala a processzorban végződik, de legalábbis azon át vezet a memóriához, esetleg egy másik készülékhez (l. egysín-rendszerű gépet). Ha az átvitel útvonala keresztülhalad a processzoron, közbenső tárolóként egyes gépeknél az akkumulátor, másoknál mikroművelet-szinten kezelt (a programozó által nem elérhető) regiszter szolgál.

A programfüggetlen átvitel útvonala a processzor érintése nélkül közvetlenül, de a processzoron át is vezethet a memóriához. Vannak gépek, melyekben mindkét (tehát a memóriához közvetlenül és a processzoron keresztül haladó) programfüggetlen átviteli útvonalat is használják. A készülék—készülék programfüggetlen átvitel csak az egysín-rendszerű gépeknél lehetséges, s a processzort mindig elkerüli.

Összeköttetési módok

A készüléket dedikált vonal vagy sín köti össze a géppel. A dedikált vonal csupán egyetlen kommunikációs kapcsolatot szolgál ki, a sít az egynél több átviteli reláció jellemzi. A ponttól-ponthoz szervezési elvet követő kezdeti számítógépben minden egyes készüléknek dedikált vonala van. A kisgépekben, melyek jellegzetesen sín-szervezésűek, dedikált vonallal elvétve találkozunk. Dedikált vonala van egyes esetekben pl. a kezelőpult kapcsoló és adatközlő regiszterének, a minimál-konfigurációhoz tartozó írógépnek, a csupán egyetlen készüléket fogadó közvetlen memóriakapcsolatnak. A dedikált vonal és a hozzá tartozó készülékvezérlő ugyan egyszerű, de egyedi szerkezetű. Számos készülék egységes paraméterek és elvek alapján való csatlakoztatása egyedül sín-szervezéssel valósítható meg. A csatlakozás egyetemessége messzemenően kompenzálja azt a hátrányt, amit a sínnek és készülékvezérlőjének bonyolultabb szerkezete jelent.

A rendszert, melynek révén a készülékek egy csoportja valamely sínen át géppel — esetleg egymással is — meghatározott átvitel módban kommunikál, csatornának nevezik. A gépcsatlakozás és a géppel való kommunikáció három különböző szintjét képviselő csatornatípus különböztethető meg:

1. a programozott csatorna,
2. a programfüggetlen csatorna,
3. az autonóm csatorna.

Közülük az első minden gépnél kötelezően megtalálható. A programfüggetlen csatorna gyakran csak opcióként áll rendelkezésre, míg autonóm csatornával a kisgépek körében elvétve találkozhatunk.

A programfüggetlen csatornát az irodalom számos különböző elnevezéssel illeti (direct memory access; data break; cycle stealing transfer channel; data channel); leginkább közvetlen memóriacsatornának hívják. A terminológiai határozatlanság oka: az átvitel kétféle — processzort elkerülő és processzoron áthaladó — lehetséges útvonala. A különböző elnevezésekben részben az egyes útvonalakkal társuló másodlagos csatornatulajdonságok tükröződnek.

Az autonóm csatornának ugyancsak egy sor különböző neve van (block I/O; concurrent I/O; multiplex channel; standard channel). Blokkátvitel, éppúgy mint több készülék multiplex üzeme azonban a programfüggetlen csatornán is megvalósítható; az elnevezések tehát kevésbé jellemzők. Az autonóm csatorna is programfüggetlen adatátvitelt végez, s a programfüggetlen csatornától csupán adatforgalmának magasabb színvonalú szervezőmódja, üzemének zártsága, *autonómiája* tekintetében tér el. Míg a programfüggetlen csatorna egyik oldala lehet nyitott, melyen át külső kezdeményezést és előkészítést is elfogadhat, az autonóm csatorna kizárólag a géppel áll kapcsolatban; mindig a program előkészítése és indítása alapján végzi (programfüggetlen átvitelek sorából álló) munkáját.

4.1.2. A programozott csatorna

A csatorna szerkezete

A programozott csatornához a sínen kívül a gép oldalán a sínvezérlő, a készülék oldalán a készülékvezérlő, ill. a készülékvezérlő bizonyos része is hozzátartozik.

A *sínvezérlő* lehet a processzorvezérlő nehezen elhatárolható része, a processzorvezérlőtől elkülönülhet, de a processzortól független is lehet. Faladatköre az MPK hálózat felépítésmódjával bizonyos mértékben változik. Mindenkor alapvető funkciói a következők:

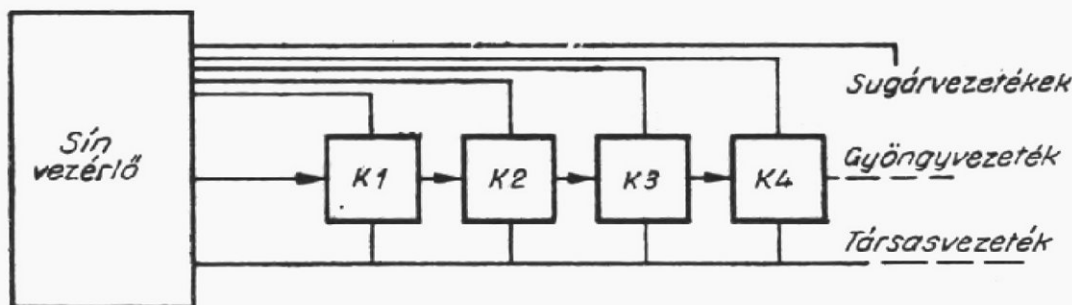
1. A csatornát működtető utasítás alapján létrehozza a processzor—készülék kapcsolatot, vagy legalább is részt vesz e kapcsolat létesítésében, és esetleg az utasítás által előírt csatornafeladat (pl. átvitel) végrehajtásában.
2. Kezeli a megszakításkérést; részt vesz a prioritáshierarchiának megfelelő legmagasabb rangú készülék kiválasztásában és a programmegszakítással kapcsolatos adminisztrációban.

A sínvezérlőn belül, ennek megfelelően, két fő funkcionális egység különböztethető meg: a sínkezelő és a megszakításkezelő.

A készülékvezérlő három funkcionálisan önálló része (síncsatlakozó-, vezérlő-, végrehajtó egység) közül adott rendszeren belül tulajdonképpen csak a csatlakozóegység (interface) tartozik a csatornához, s ennek *szervezőmódja* egységes; a másik

kettő készülékenként egyedi. A „szervezőmódra” azért kell hangsúlyt helyezni, mivel a gépek átlagánál a csatlakozóegységek egymással nem pontosan azonosak, hanem készüléktípusonként több-kevesebb eltérést mutatnak. Egységesek azonban a sokpólusú csatlakozók és bekötéseik, a kártyaméretük, a logikai és impedanciaszintek, s ami a leglényegesebb: a kommunikációs tulajdonságok, melyek alapján a csatlakozóegység és a sín értenek egymás nyelvén.

A sín — a csatorna átvivő szerve — funkcionálisan: címvonalból, adatvonalból, és szolgálati vonalból áll. E vonalak lehetnek fizikailag is önállóak, amikor is mindegyik vonalhoz meghatározott állandó vezetékészlet tartozik. Más esetben a sín bizonyos vezetői egynél több funkciót is elláthatnak; a csatorna működése során pl. egyszer adatot, máskor címet vagy parancsot szállítanak. A sínben három különböző típusú vezetékkel találkozhatunk: A társasvezeték (party line) minden készülékvezérlő azonos módon csatlakozik; a sugárvezeték (unique line) a sínvezérlőt és egy készülékvezérlőt közvetlenül köt össze; a gyöngyvezeték (daisy chain line) a készülékvezérlőket gyöngyszemként fűzi fel. A 4.1. ábra a különböző vezeték típusokat szemlélteti.



4.1. ábra.
A számítógép sínrendszerének különböző vezeték típusai
K1, K2, ... készülékek

A csatornához tartozó összes készüléknek saját (egy, esetleg több) címe van. A készüléket a program címével hívhatja, megszakításkérés kezelése kapcsán cím szolgál azonosítóul. A készülék és a sín kapcsolata a címvonalra küldött készülék-címmel aktivizálható. A készülék cím a processzorban mindig kódolt alakban áll elő. A teljes sínű csatornarendszerben maga a kód jelenik meg a címvonalon, s mivel a címvonalat ez esetben társasvezetékek realizálják, valamennyi csatlakozóegység címbemenetén is. A kódot egyedül a megcímezett készülék címfelismerő (dekódoló) eleme ismeri fel. Miután ez megtörtént (ha előírt feltételek nem zárják ki), aktivizálódik a csatornával való kapcsolat. Csonka sín esetében a címdekódolás már a sínvezérlőben megtörténik. A dekódolás folyamata a sínvezérlőt a megcímezett készülékkel összekötő sugárvezeték kijelöléséből, és e vezetékre jel kibocsátásából áll. A csonka sínű csatorna csatlakozóegysége ezért nem címfelismerő, hanem (1-bites) kijelölő áramkört tartalmaz; a sínrel való kapcsolatot e kijelölő áramkör aktivizálja.

Adatátviteli utasítás végrehajtása során, a kapcsolat aktivizálása után, a csatlakozóegység és a processzor előírt regisztere között a sín adatvonalán át összeköttetés és adatátadás létesül. Az összeköttetés és adatátvitel létrehozásával kapcsolatos eljárást vagy a sínvezérlő, vagy a processzor közvetlenül vezeti, miközben a szolgálati vonal vezérlő- (időzítő, kapuzó- és jelző-) vezetői útján a csatlakozóegység irányítást kap. A vezérlővezetéseket egyes gépeknél a készülékvezérlőnek szóló parancs továbbítására is felhasználják, a parancsközlésre azonban a szolgálati vonal külön vezetékcsoportja is rendelkezésre állhat. A szolgálati vonalhoz tartozhatnak még állapotjelző, megszakításkérő és megszakításkérés-kezelő vezeték, szerkezeti precedencialánc gyöngyvezeték, különböző visszaállító és szinkronjel-vezetékek stb. A külön-

böző csatornasínek sajátosságai leginkább a szolgálati vonal szervezőmódjában jutnak kifejezésre.

Különbség tehető programorientált (software oriented) és szerkezetorientált (hardware oriented) csatorna között. Az előbbi a csatorna-adminisztrációval járó feladatok jelentékeny részét a programra hárítja; ellentétként a csatorna (csatorna-vezérlő, készülékcsatlakozó és sín) szerkezete igen egyszerű. A szerkezetorientált csatornát a szerkezeti ráfordítás árán elért programtakarékoság jellemzi.

A csatornát működtető utasítások

Az utasításkészletben megjelölhetők azok az utasítások, melyek a csatornát működtetik. A két- és háromsínű gépeknél ezek a Be/Ki utasítások független csoportját alkotják. A Be/Ki utasítások között minden esetben megtalálható egy-egy vagy többféle, adatbeviteli, illetve kiviteli utasítás, melyeket esetenként, különösen a programorientált csatornával rendelkező gépeknél, számos csatorna-kezelő (állapotellenőrző, törlő stb.) utasítás egészíthet ki. Az egysínű gépeknek nincsenek külön Be/Ki utasításai; programozott csatornájuk mindazokkal a címtartalmú utasításokkal működtethető, melyek a készülékvezérlők szerkezeti adottságai alapján értelemmel bírnak.

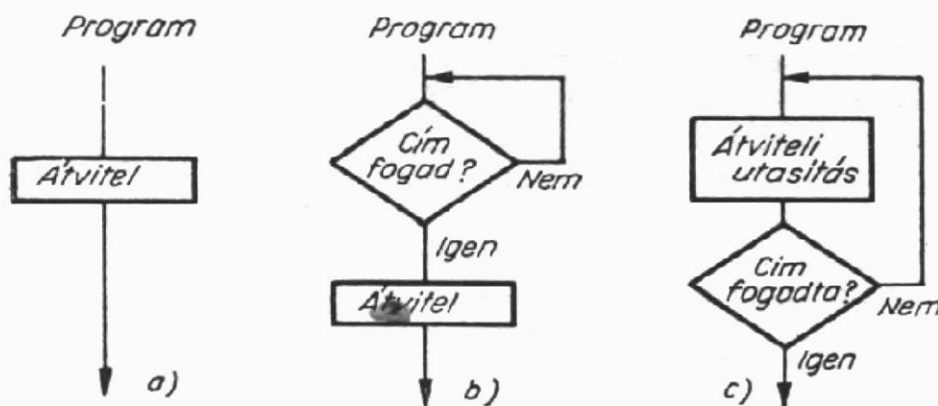
A két- és háromsínű gépek készülékeinek címtartománya a memória címtartományától független; a memóriasín a programozott csatorna sínjétől elkülönül, s mindkét sínnek önálló címvonala van. Ezért különülnek el — szükségképpen — a memóriavonatkozású utasítások, ill. a Be/Ki utasítások. Az egysínűeknek azonban egyetlen, a memóriának és a készülékeknek egyaránt szóló címvonala, a gépeknek egyetlen címtartománya van. E címtartományon belül helyezkednek el a memóriacímek és a készülékcímek is. A processzorcentrikusság háttérbe szorulását az utasításkészlet homogenitása is tükrözi.

Az adatátvitel változatai

Programkapcsolatú átvitelt kezdeményezhet a folyó program vagy megszakításkérő jelével maga a készülék. Akár program, akár készülék kezdeményezi is azonban az átvitelt, annak végrehajtása mindig ugyanazon adatmozgató utasítás(ok) alapján történik.

Az adatátvitel két változata különböztethető meg: 1. állapotellenőrzés nélkül, 2. állapotellenőrzéssel végrehajtott átvitel.

Állapotellenőrzés nélkülinek azt az átvitelt nevezik (4. 2. a ábra), melyet megelőzően



4.2. ábra.

A programozott csatorna átviteltípusai

(a) állapotellenőrzés nélkül; (b) előzetes állapotellenőrzéssel; (c) utólagos állapotellenőrzéssel végrehajtott átvitel

vagy követően a periferikus készülék állapotát — fogadókészsége szempontjából — nem ellenőrzik; az átviteli utasítás végrehajtására annak feltételezésével kerül sor, hogy a készülék fogadóképes. A fogadóképeség nyomon követése azonban pusztán az időbeli viszonyok számbavétele alapján rendkívül nehéz; célszerűbb valami módon minden egyes átvitel kapcsán ellenőrizni a készülék állapotát.

Állapotellenőrzéssel végrehajtott átvitel esetében vagy az adatmozgató utasítás végrehajtása előtt, vagy azt követően, a készülék állapotát a program ellenőrzi (4.2. ábra).

Esetenként a programorientált csatornarendszer adatvonala félig nyitott, ami annyit jelent, hogy a csatorna a készülék állapotától függetlenül mindig végrehajtja az adatmozgató utasításokat. Amennyiben átvitel kapcsán a program nem ismerné a készülék állapotát, hibás — egy megelőző készülékműveletet is megzavaró — művelet hajthat végre. Ezt elkerülendő, a csatornát érintő adatmozgató utasítások elé itt feltételes kérdő utasítást kell elhelyezni. A feltételes kérdő utasítás a program által adatátvitelre kiszemelt készüléket címével kérdezi meg foglaltsági állapotáról. A lekérdező utasítás mindaddig ciklikusan ismétlődik, míg a készülék átvitelre kész állapotot nem jelez. Csak ekkor kerül sor az adatmozgató utasítás végrehajtására (4.2b ábra).

Gépekben, melyeknek készülékvezérlője foglalt állapotban az adatmozgató utasítást nem hajtja végre, a készülék folyamatban levő üzemét a program nem zavarhatja meg. Minden adatmozgató utasítást követően a processzor jelzést kap: elfogadta vagy visszautasította-e a készülék az utasítást. A jel a processzor egy regiszterébe (pl. kapocs) kerül, ahol értéke feltételes (ugró vagy átlépő) utasítással megvizsgálható. A készülék foglaltságának idejére — ez esetben is — zárt ciklus alakul ki (itt azonban az adatmozgató utasítás ismétlődésével) mindaddig, míg csak a készülék az utasítást el nem fogadja (4.2c ábra).

A zártciklusú, elő- vagy utóellenőrzéssel kapcsolódó programkezdeményezésű átvitel mód súlyos hátránya, hogy — különösen a lassú készülékek — a processzor munkáját igen hosszú időszakokra blokkolják. Gyakran célszerűbb a készülék átvitelre kész állapotát megszakításkéréssel jelezni, az átvitelt programkezdeményezés helyett a készülék kezdeményezése alapján állapotellenőrzés nélküli utasítással végrehajtani. A megszakításkérés alapján végrehajtott átvitel a processzor idejét csak a megszakításkérés kezelésének, s a megszakítási (átviteli) rutin végrehajtásának időtartamára foglalja le. Bármennyire rövid is ez az idő, az időegységre eső átvitt adatok számának növekedésével tekintélyes értékévé integrálódhat, s ilyen esetben célszerű programfüggetlen vagy autonóm átviteli csatorna alkalmazása.

A megszakítórendszer

Az egyes gépek megszakítórendszerei egymástól igen sok tekintetben eltérhetnek. A legegyszerűbb esetben a gépeknek csupán egyetlen (DEC PDP 8), más esetben (HP 2100) számos független megszakítási szintje van. Nagyobb számú forrás megszakításkérésének kezelése az egy független megszakítási szintű rendszerben jórészt programozási úton, ahol pedig a független megszakítási szintek száma a megszakításkérő források számát eléri vagy meghaladja, ott teljes mértékben szerkezet segítségével történik. Mint a csatorna egésze, a megszakítórendszer is lehet program- ill. szerkezetorientált, de a két szélsőséges megoldás közötti átmenettel is találkozunk.

A processzor csak azokban a pillanatokban fogadhat el megszakításkérést, melyekben állapota — a programkontextus — kimenthető. Ez a pillanat a legtöbb gépnél az éppen végrehajtott utasítás befejezésének időpontjával esik egybe, bár utasításvégrehajtás közben megszakítható gép is ismeretes (Philips P855; P 860).

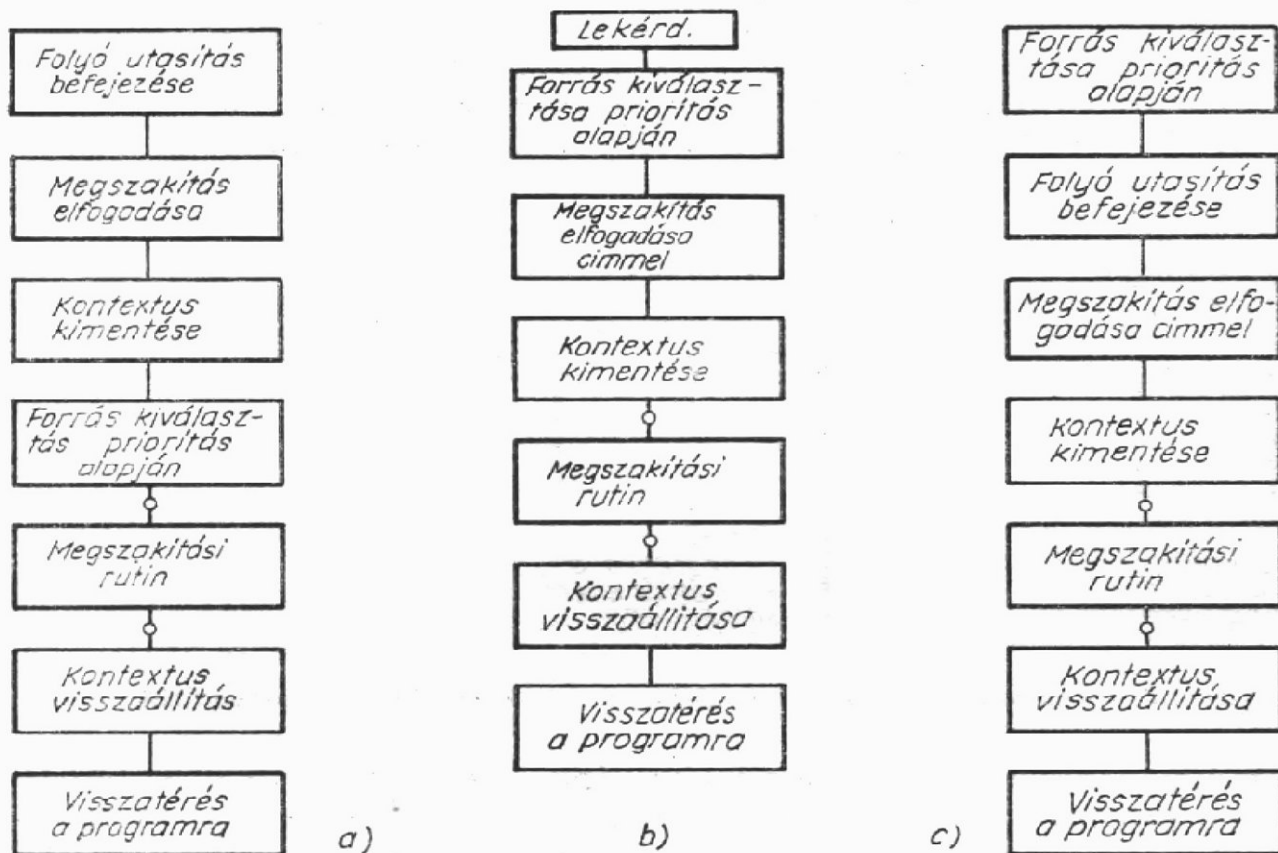
Általános esetben számos forrás jelentkezik megszakításkéréssel. A gép fogadó-képességének minden időpontjában megvizsgálja, vannak-e a jelentkező források között olyanok, amelyeket a folyó programmal szemben előnyben kell részesítenie. Ha ilyen források vannak, meg kell állapítania, melyik jelentkezőnek van prioritása, s reá kell térnie az illető forrás megszakítási programjára, miközben a folyó program kontextusa is kimentődik. Megszakításkérés-kezelésen a megszakítási eljárásnak, a megszakításkérés jelentkezésétől a megszakítási rutinra való rátéréséig tartó szakaszát értjük, a kezelés idejét pedig *válaszidőnek* nevezzük.

A legmagasabb rangú forrás identifikációjának két alapvetően eltérő módja különböztethető meg: 1. az utó kiválasztás és 2. az előkiválasztás.

Az *utó kiválasztással* működő megszakítórendszerben az összes megszakításkérő forrás jele a processzornál egyidejűleg jelentkezik. A processzor csak azt érzékeli: megszakításkérés van, de nem tudja, egy vagy több forrás jelentkezik-e, melyek a jelentkezők, s hogy közülük melyiket részesítse előnyben. A processzor a megszakításkérés elfogadása után kell, hogy identifikálja a kérelmezők közül a legmagasabb rangút (4.3a ábra).

Az *előkiválasztást* alkalmazó megszakítórendszerben a megszakításkérések nem lépnek a processzorra, hanem a forrásoknál várákznak. A processzor, fogadóképesége időpontjaiban, lekérdező jelet bocsát a forrásokat prioritási rangsorban felfűző gyöngyvezetékre (precedencialánc). A várákzók közül a folyó programmal szemben prioritást élvező legmagasabb rangú forrás a processzor kérdésére megszakításkérő jelével (mely egyúttal címét is identifikálja) válaszol. Itt tehát a processzorhoz mindig csak a már kiválasztott és azonosított forrás kérése jut el (4.3b ábra).

Az előkiválasztásnak egy más eljárásával az egysínű gépeknél találkozhatunk, melyeknél a forrásokat a processzortól független sínkezelő rangsorolja (4.3c ábra).



4.3. ábra.

A megszakításkérés kezelésének különböző módjai

Forrásazonosítás: (a) utólagos kiválasztással; (b) előkiválasztással; (c) előkiválasztással, sínkezelő útján.

Az előkiválasztással dolgozó megszakítórendszert *vektoriális megszakítórendszernek* is nevezik, mivel a megszakítójellel együtt a forrás rutinjára mutató cím is megjelenik.

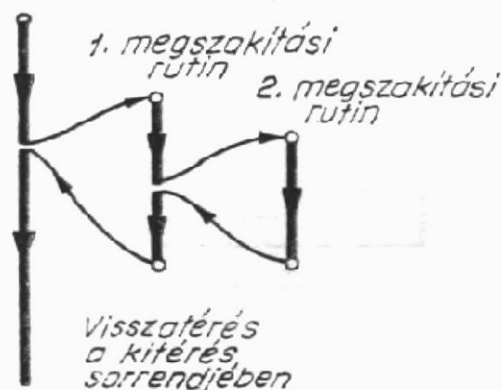
A megszakítási, tehát a szigorúan a forrás által igényelt rutint mindig kontextus-visszaállítás és az elhagyott programra való visszatérés mozzanata követi. A megszakítási eljárás három szakasza tehát:

1. megszakításkérés-kezelés,
2. megszakítási rutin,
3. zárószakasz.

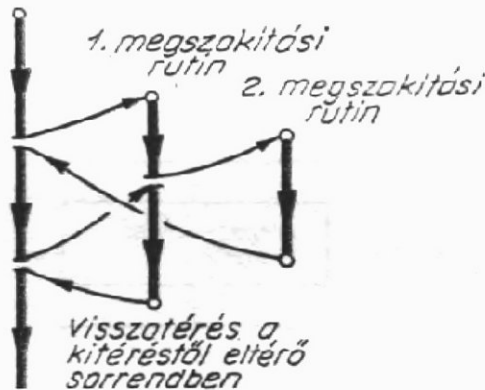
Az 1. és 3.-at együtt programrátétnek (overhead) nevezzük. Alkalmazói szempontból nagy jelentősége van a programrátét időtartamának, lévén ez a csatorna átbocsátó képességének lényeges tényezője. A programrátét időtartamát elsősorban a válaszidő szabja meg. A válaszidőnek gépfüggő és feladatfüggő komponense is van. A gépfüggő komponens megszakítórendszerként igen különböző; elsősorban a megszakítórendszer (program/szerkezet) orientáltságától, a forráskiválasztás (azonosítás) módjától, a kontextus kiterjedtségétől és kimentésének módjától, az utasítások végrehajtási idejétől függ. A feladatfüggő komponens változhat a megszakításkérő források számával, a független megszakítási szintek számával (bár ez önmagában a gép tulajdonsága, de adott feladat sajátosságainak megfelelően hat), az egyes források által időegység alatt kibocsátott megszakításkérések mennyiségével, a programfüggetlen csatorna igénybevételével.

A prioritási hierarchiát programban vagy szerkezetben rögzítik. Általában mód van a hierarchia menet közbeni, program útján való módosítására. Ha pl. egyes források megszakításkérése program útján maszkolható, ez máris a szerkezetben rögzített hierarchikus rend módosításának eszközéül szolgálhat. Vannak gépek, melyekben a megszakítórendszer egésze, másokban egyes prioritási osztályok maszkolhatók, utasítás segítségével. Általában megengedett a többszörös megszakítás, ami a megszakítási rutinok egymásba ágyazódásával jár. A visszatérés szokásosan a kitéréssel azonos sorrendben történik, egyes gépeknél azonban a visszatérési sorrend a kitérési sorrendhez viszonyítva megváltoztatható (4.4. ábra).

Alapprogram



Alapprogram



4.4. ábra.

Visszatérés megszakítás állapotából a kitérés sorrendjében és a kitéréstől eltérő sorrendben

Különbség tehető belső és külső megszakításkérés között. A belső megszakításkérés forrásai a processzor—memória együttesen belül helyezkednek el; kezelésük a külső megszakításkéréssel együtt, de attól elkülönítve is történhet. Belső megszakítást általában valamely hibaállapot vagy üzemzavar (pl. hálózatkiesés), esetleg reális idejű óra generál; bizonyos körülmények között és bizonyos gépeknél pedig maga a program is megszakíthatja önmagát (trap).

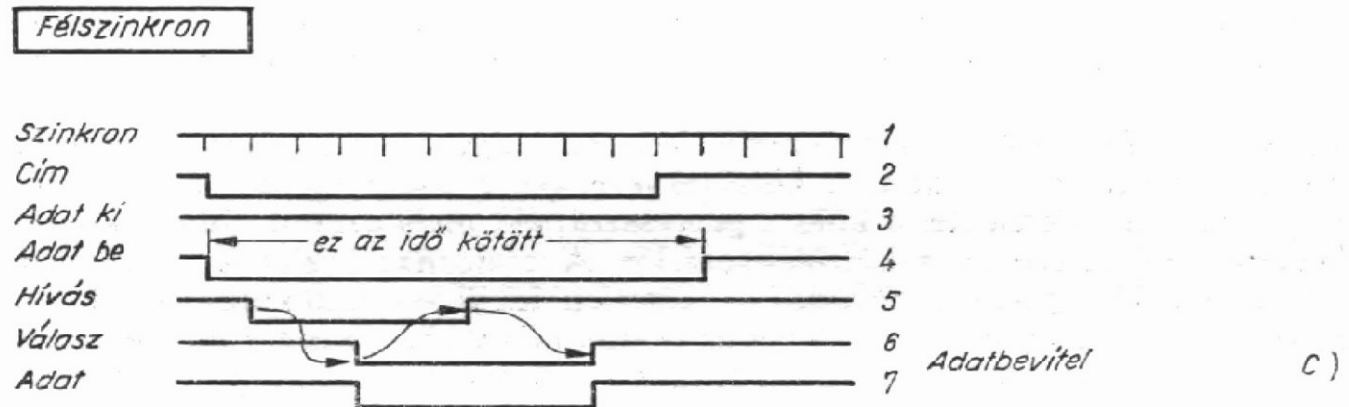
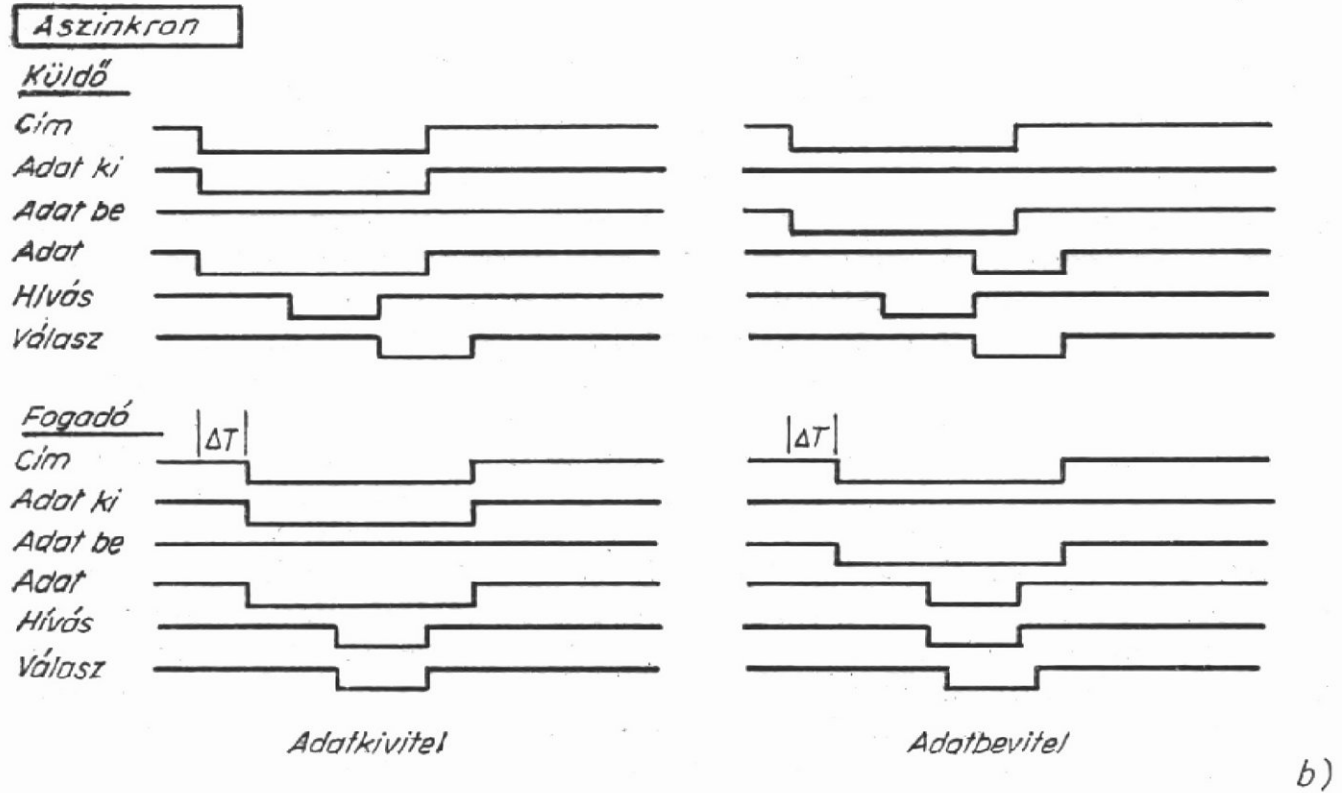
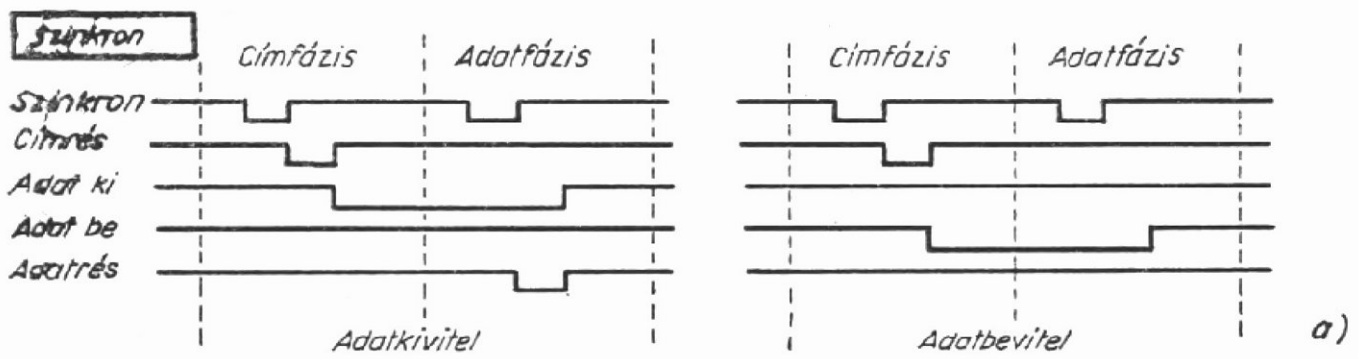
Külső megszakításkéréssel folyamatállapotú készülék, vagy a számítógépen kívül eső forrás jelentkezhethet. A programozott csatorna meghatározott külső megszakítási szintekkel, ill. osztályokkal rendelkezik.

Szinkron, aszinkron, félszinkron kapcsolat

A processzor—készülék kapcsolatot (speciális esetektől eltekintve) mindig a processzor hozza létre, valamely csatornaművelet végrehajtása céljából. A kapcsolatlétesítésnek szinkron, aszinkron és félszinkron módja ismeretes. A kapcsolatban megvalósuló csatornaművelet egy vagy több munkafázisból áll. A szinkron csatorna munkafázisainak s így a csatornaműveleteknek az időtartama is előre meghatározott, a műveleten belüli eseményeket óragenerátor jelei időzítik. Az aszinkron rendszerben a munkafázisokat a kapcsolatban résztvevő egységek által generált koordináló jelek határolják; egy-egy munkafázis, maga a csatornaművelet időtartama is a résztvevő egységek funkciójától és működési sebességétől függően változhat.

A *szinkron* működési mechanizmus szemléltetésére a 4.5a ábra az SPC 16 számítógép adat ki- és bevitelének módját mutatja be. Az egyszerűség kedvéért az ábra csak a vezérlővezetékek jeleit tünteti fel. A szinkron rendszert jellemzően a vezérlő jelek mindig a processzortól a készülék felé haladnak. Adott esetben az adatvonal szolgál a cím átvitelére is, ezért mind a kiviteli, mind a beviteli művelet címfázisra és adatfázisra különül el (általánosságban ez nem szükségszerű). A cím, a címfázis szinkronjelével kerül a vonalra. A készülékvezérlő címfelismerő egységét a CÍMRÉS kapuzza. A CÍMRÉS visszaállításával induló ADAT KI, ill. ADAT BE jel vezet át a második munkafázisba. ADAT KI a készülékvezérlő adatbemenetét, ADAT BE pedig adatkimenetét készíti elő. Az adatvonalat kivitel során az ADATRÉS, bevitel során a SZINKRON jel kapuzza. Visszajelzés híján, a processzor a címfázis eredményének konkrét ismerete nélkül tér rá az adatfázisra. A helyes működés feltétele a szinkronjel periódusidejének, valamint az egyes jelek hosszának összehangolása a csatorna-áramkörök sebességi adottságaival. Ebből következik, hogy adott számítógép esetében a csatornához csatlakoztatható készülékek működési sebességének bizonyos alsó korlátja van. A szinkroncsatorna működését átgondolva az is kiténik, hogy míg adatkivitelnél a kábelhossz nem játszik szerepet, adatbevitelnél a kábel által okozott késleltetés bizonyos értéket nem léphet túl, hiszen az adatnak a processzorhoz a soron következő szinkronjel meghatározott időbeli környezetében kell beérkeznie.

Aszinkron kapcsolat esetén a csatornaműveletet nem egyedül a processzor vezérli. A műveletet a processzor indítja, a készülékvezérlőnek szóló feladattal. Utóbbi, a feladat teljesítéséről visszajelzést küld a processzornak, mely ezután rátér a következő feladatra, amiről értesíti a készülékvezérlőt. A csatornaművelet a processzor és a készülékvezérlő egymásba fűződő — kötetlen idejű — munkafázisaiból áll. Az egymásnak küldött koordináló jeleket az irodalom parola- (handshake) jeleknek nevezi. Láncolódás nélküli, félig láncolódó és láncolódó parolajelek különböztethetők meg attól függően, hogy e jelek mennyire szorosan követik egymást. A 4.5b ábra láncolódó parolajelű aszinkron csatorna működésmódját szemlélteti be/kiviteli műveleten. ΔT késleltetésű kábel feltételezésével, az ábra mind a processzor (KÜLDŐ), mind a készülék (FOGADÓ) állapotát nyomon követi. Kivitel során a processzor munkafázisa a cím, a műveletjel (ADAT KI) és az adat egyidejű sínrehelyezésével kezdődik. Bizonyos késéssel a processzor parolajelével (HÍVÁS) jelzi, hogy a sínrehelyezés megtörtént. A HÍVÁS jel egyrészt a készülékvezérlőt kapuzza, másrészt indítja a készülékvezérlő VÁLASZ parolajelét. VÁLASZ frontja a processzornál visszaállítja a HÍVÁS, a CÍM, ADAT és ADAT KI vonalat. ΔT idő múlva a HÍVÁS



4.5. ábra.
MPK hálózat elemei közötti kapcsolatlétesítés különböző módjai
(a) szinkron; (b) aszinkron; (c) félszinkron

lefutása visszaállítja VÁLASZ-t, majd újabb ΔT múlva a VÁLASZ lefutásának hatására a processzoroldalon zárul a csatornaművelet. Látható, hogy a parolajelek élleikkel láncolódnak egymáshoz. Az adatbevitel hasonlóképpen zajlik le.

A félszinkron csatornarendszerben a csatornajeleket részben óragerátor szinkronjelei időzítik, s a csatornaműveletek időtartama maximálva van. Ugyanakkor

a csatornaművelet lebonyolítása éppúgy parolajelek közvetítésével történik, mint az aszinkron rendszerben. A 4.5c ábra félszinkron csatorna adatbeviteli műveletét szemlélteti, a jelfrontokat felléptük helyén mutatva; az 1—5 jel a processzor kimenetén, a 6, 7 a készülékvezérlő kimenetén mutatkozik. A félszinkron rendszer a hozzá csatlakozó áramköri elemek sebességére, az alkalmazható kábel hosszára nézve ugyanúgy korlátokat állít, mint a szinkron rendszer. Ugyanakkor, a parolajelek alkalmazásával, üzeme a szinkronrendszerénél megbízhatóbb.

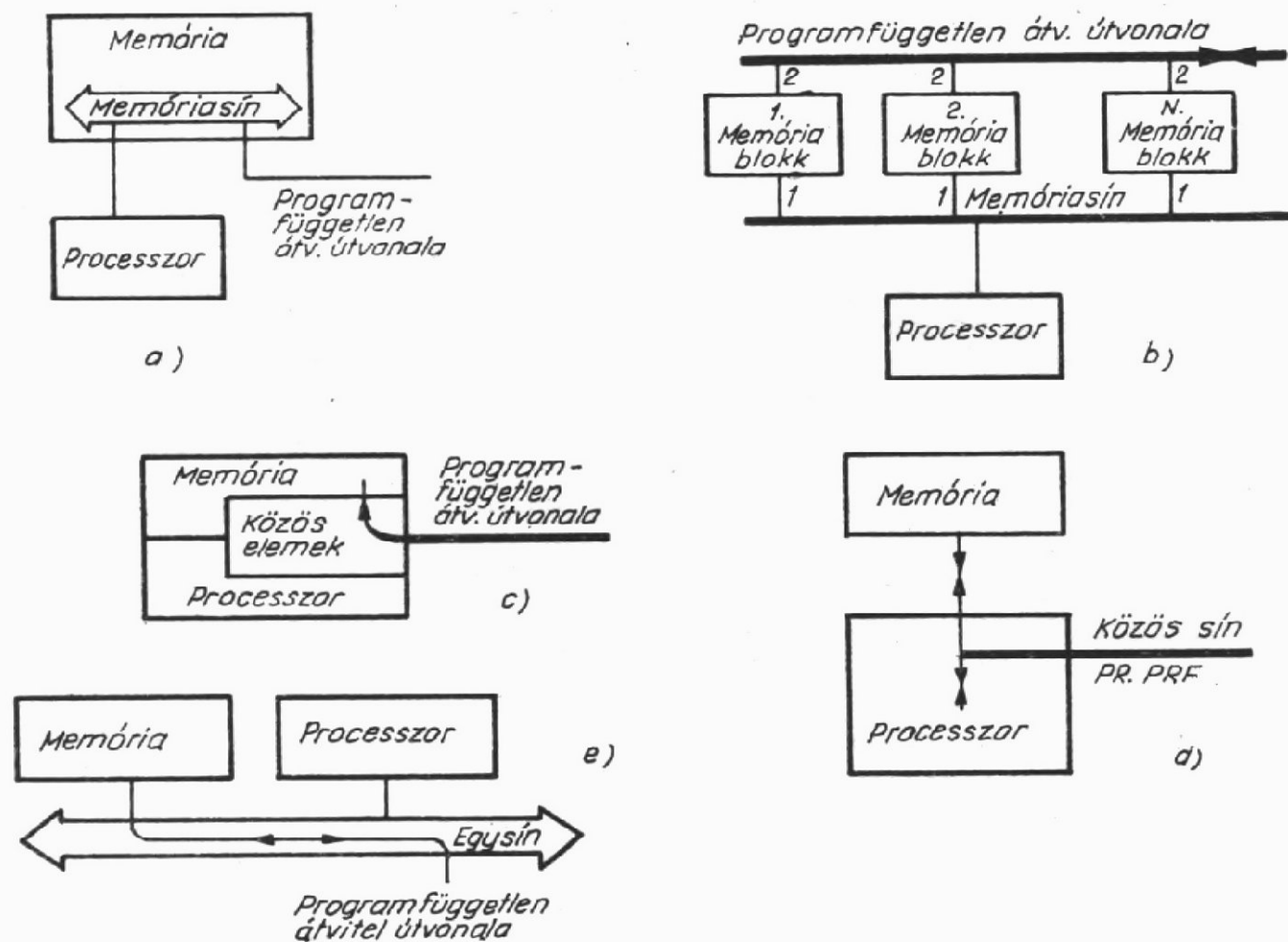
Az aszinkron csatorna megbénul, és általa a processzor is reteszeli, ha a HI-VÁS-ra nem érkezik VÁLASZ. Ennek elkerülésére időmérést kell végezni. Ha a VÁLASZ késlekedése bizonyos kritikus küszöböt túllép, a csatorna figyelőáramköre (watch-dog) hibajel kibocsátásával felszabadítja a processzort. A félszinkron csatorna egyszerűbb, mivel a parolajeleknek a csatornaművelet kötött időtartamán belül kell lefutniuk; külön időellenőrzésre nincs szüksége.

A szinkron csatorna szerkezeti kevésebb bonyolult, mint az aszinkron, azonban kevésebb megbízható; a félszinkron a kétféle megoldás adottságai közötti kompromisszumot nyújt.

4.1.3. A programfüggetlen csatorna

Csatorna-útvonalak

Amint a 4.1.1. pontban röviden említettük, a programfüggetlen adatátvitel a processzort elkerülő és a processzort érintő útvonalon végezhető. A gyakorlatban mindkét lehetőség felhasználásával számos csatornamegoldás jött létre.



4.6. ábra.

A programfüggetlen adatátvitel lehetséges útvonalai

(a), (b) és (e) processzort elkerülő; (c) és (d) processzoron áthaladó útvonalak

A 4.6a ábrán egybefüggő memóriát, processzort elkerülő útvonalat látunk. A processzor és a csatorna a memóriához csak egymást kizárva fordulhat. A programfüggetlen csatorna a processzorral szemben mindig előnyt élvez. Ez nem azt jelenti, hogy a processzor által már megkezdett memóriaciklus megszakítható, de ha a jelentkezés egyidejű, a memóriát a csatorna nyeri el (cikluselvétel), a processzor pedig egy memóriaciklus idejére blokkolódik. Nem minden csatornaművelet jár azonban együtt — adott elrendezés esetében sem — cikluselvétellel; a processzor nem memóriavonatközös feladatát programfüggetlen átvitel folyamán is végezheti.

A 4.6b ábra a részenként önálló memória és a processzort elkerülő útvonal esetét szemlélteti (pl. CII—10010). A programfüggetlen csatorna itt cikluselvétellel csak akkor él, ha történetesen valamelyik memóriablokkot a processzorral egyidőben igényli. Amennyiben a processzor, ill. a csatorna más-más blokkot használ, vagy a processzor memóriaigénnyel nem él, munkájuk szimultán; cikluselvétel nincs.

A 4.6c ábra hálózatában (pl. PDP 8) a programfüggetlen csatorna a memóriával bizonyos a memória és processzor által közösen használt elemeken (adatregiszter, címregiszter) át kommunikál. Nyilvánvalóan minden csatornaművelet blokkolja a processzort; a csatorna minden művelete cikluselvétellel jár.

A 4.6d ábra olyan gépet illusztrál (pl. SPC 16), mely programozott (PR) és programfüggetlen (PRF) csatornát közös, funkcionálisan összevont sínnel valósít meg. A sítet a programfüggetlen csatorna prioritása mellett mindkét csatorna használhatja. A programfüggetlen csatorna útvonala a processzoron belül a processzor—memória összeköttetésbe torkollik. A programfüggetlen csatorna minden művelete e rendszerben cikluselvétellel jár.

Végül a 4.6e ábrán látható esetben mind a programozott, mind a programfüggetlen átvitel útvonala is az egysínen át vezet. Cikluselvételre itt — éppúgy, mint a 4.6a elrendezésben — akkor kerül sor, ha a processzor a programfüggetlen csatornával egyidejűleg igényli a sítet.

Működésbeli sajátosságok

A programfüggetlen csatorna készülék—memória adatátvitelt a programkontextus érintése, program igénybevétele, ill. megszakítása nélkül tesz lehetővé; az adatátvitel a csatornához tartozó készülékvezérlő(k) kezdeményezése (szolgáltatás kérése) alapján történik. Ezért a programfüggetlen csatorna átbocsátóképessége lényegesen nagyobb, mint a programozott csatornáé. A cikluselvétel nagyobb tömegű adat átvitele esetén a processzort viszonylag hosszú időre blokkolhatja ugyan, de a processzor idejének egy adatra eső vesztesége nagyságrendekkel kisebb lehet, mint amit a programozott csatorna okoz. A processzort elkerülő csatorna üzemének helyes szervezése esetén a csatorna által okozott processzoridő kiesés minimálisra csökkenthető.

A csatorna mindig a memória egy kijelölt tartományával kommunikál, mely tartomány helyét egyrészt a program, másrészt a csatorna pontosan kell, hogy ismerje. A csatornával, működésének megindítása előtt, minden esetben közölni kell munkájának feltételeit, elsősorban azt, hogy a memória mely területére hány adatot kell bevinnie vagy onnan kivinnie. Azt az eljárást, melynek során a csatornával közlik munkafeltételeit, a csatorna előkészítésének nevezik.

Átvitel módok

A csatorna alapfeladata: az adatátvitel. Működésének mechanizmusából következően, a csatorna az előkészítés által meghatározott számú adatot visz át egy kezdőcímtől sorakozó memóriarekeszekbe vagy memóriarekeszekből, tehát blokkátvitel-

módban dolgozik. A processzort elkerülő útvonal esetén az adatátvitel a csatorna által végezhető egyetlen művelet, ha azonban az útvonal a processzoron halad át, különböző egyéb műveletek is lehetségesek, melyek végrehajtásában — természetesen a programkontextus megzavarása nélkül — a processzor is részt vesz. Az SPC 16 számítógépnél pl. az adat be-, ill. kivitelen kívül még a következő csatornaműveleteket találjuk:

- bevitt adat hozzáadása a memóriarekesz tartalmához,
- a bevitt adat és a rekesztartalom logikai összeadása,
- a memóriarekesz tartalmának inkrementálása.

Az előkészítő információ két leglényegesebb eleme — az átviendő adatok száma és a kommunikációban érintett memóriaterület kezdőcíme — a csatornához tartozó két regiszterbe kerül. A csatorna működése során, minden egyes adat átvitelekor e regiszterek tartalma inkrementálódik (dekrementálódik). Ilyenképpen a *címmutatóként* szolgáló regiszter végigjárja a kijelölt memóriaterületet, az *adatszám-regiszter* pedig a végrehajtott műveletek leszámolásával a zérus felé tart. A címmutató és adatszám-regiszter általában a készülékvezérlőben található.

A csatorna szerkezeti egyszerűsíthetősége érdekében azok a gépek, melyek az átvitelen túl egyéb csatornaműveleteket is végeznek, a készülékvezérlő adatszám- és címmutató regisztere helyett a memória egy-egy rekeszét is használhatják. A készülékvezérlő csak e két rekesz helyét kell, hogy definiálja. Az adatátvitel itt három memóriaciklusból áll, melyek közül egy a folyó címet aktualizálja, egy az adatszámot aktualizálja, és ellenőrzi is egyszerre mind, egy pedig közvetlenül az adatátvitelre szolgál. Az ilyen rendszert többciklusú programfüggetlen csatornának nevezzük. A többciklusú csatorna járulékos ciklusaival együttjáró inkrementálás és feltételvizsgálat a processzorra hárul, a processzor terhelése ezáltal — az egyszerű cikluselvételhez képest — megháromszorozódik, s vezérlőrendszerének struktúrája is bonyolultabbá válik, ugyanakkor a csatorna átbecsátóképessége harmadára csökken. Ez az ára a készülékvezérlő egyszerűsödésének.

Attól függően, hogy a csatorna egy vagy több készüléket szolgál-e ki, *egyszeres és többszörös blokkátvitelről* beszélünk. A többszörös blokkátvitel tekintetében továbbá különbséget teszünk *szelektív* és *multiplex* üzemmód között. Szelektív üzemmódban a csatornához tartozó készülékek közül egyszerre csak egy lehet folyamatállapotban; a különböző készülékekhez tartozó blokkok átvitele időben egymástól elkülönítve történik. Multiplex üzemmód esetén egynél több készülék is lehet aktív; az egyes készülékekhez tartozó blokkok átvitele időben fedheti egymást.

A csatorna felépítése

A programfüggetlen csatorna felépítése és működésmódja az adatátvitel útvonalától, a processzornak a csatornafeladatokban való részvétele mértékétől, a gép MPK hálózatának sínszámától, a kiszolgált készülékek számától, az átvitel módtól (egy- vagy többciklusú) és a csatorna üzemmódjától függően változik.

Legegyszerűbb esetben a csatorna egyetlen készüléket szolgál ki, mely dedikált vonallal csatlakozik. A csatorna alkotórészei: a csatornakezelő, a vonal és a készülékvezérlőnek a csatorna munkájában részvevő elemei (4.7a ábra).

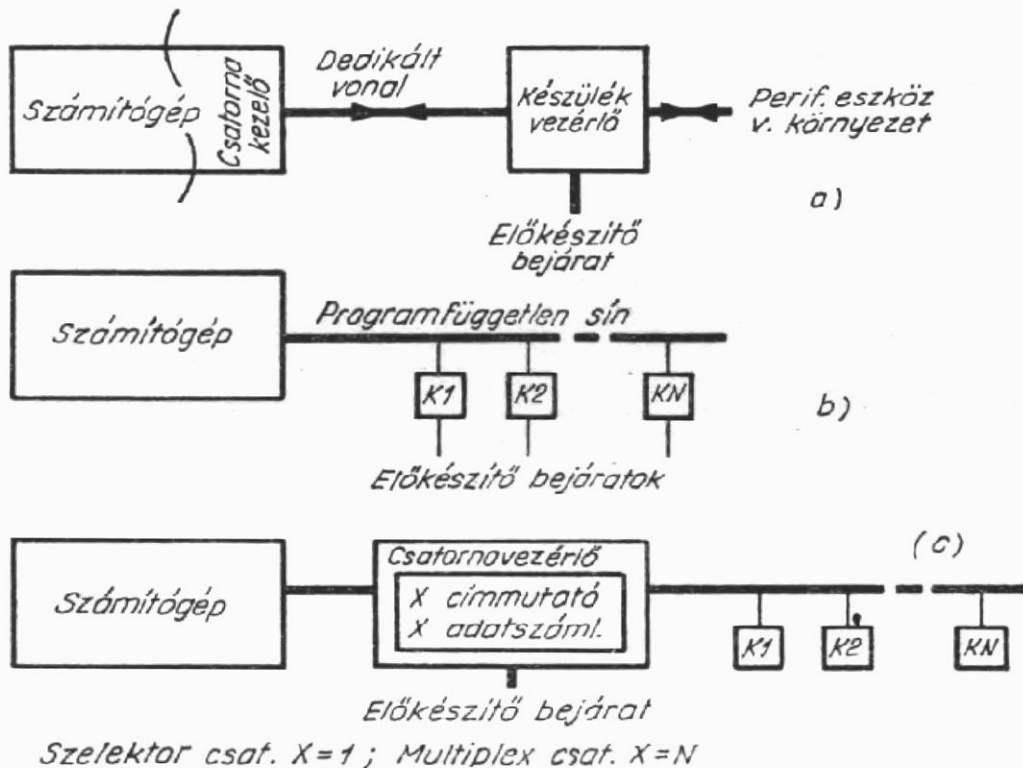
A csatornakezelőt áramkörileg a legtöbb esetben igen nehéz körülhatárolni, mivel a különböző MPK hálózatokban súlypontja hol a processzorra, hol a memóriára helyeződik, míg az egysínű gépnél szerepét a sínvezérlő látja el. A hálózat sínszámától és szervezésmódjától függően változik a készülék(ek) szolgálatkérését kezelő eljárás is. Ahol a csatorna munkájában a processzor is részt vesz, a processzorvezérlő minden gépi ciklusban engedélyező jelet küld a csatornára. Ha az engedélyező

jel szolgálatkérést észlel, a processzor a következő memóriaciklust a csatornának engedi át. A szolgálatkérés kezelésének nyilvánvalóan egész más módja szükséges azonban pl. a 4.6b hálózatban, ahol a kérés elfogadása a hívott memóriablokk foglaltsági állapotának függvénye, vagy a 4.6e hálózatban, ahol a szolgálatkérés sínigénylő jelként hat.

A készülékvezérlő három fő funkcionális eleme a programfüggetlen csatornánál is: a csatlakozó-, a vezérlő- és a végrehajtó-egység. A csatlakozóegységnek azonban itt két bejárata van, melyek egyike csatornakkommunikáció, a másik előkészítés céljára szolgál (4.7a ábra). Az előkészítés történhet a gép programozott csatornáján keresztül program útján, de végezhető külső forrásból is (pl. egy másik számítógép segítségével). A program útján előkészített készülékvezérlő két bejárata egybeesik, ha a programozott és a programfüggetlen csatorna közös, funkcionálisan összevont sínt használ. A készülékvezérlő másik sajátága, hogy nem csak a periférikus eszközt, hanem a csatornavezérlés (sínvezérlés) feladatát is ellátja.

A csatorna üzemmódjától függően, a több készüléket kiszolgáló csatornát (4.7b ábra) szelektor, ill. multiplex csatornának nevezik. Mindkét csatornatípus átvívőszerve: sín. A kétféle csatorna egymástól abban különbözik, hogy míg a multiplex csatorna rendelkezik a készülékek szimultán folyamatállapotából következő konfliktusok feloldásának eszközeivel, a szelektor csatornánál ezek az eszközök hiányoznak, ezért egyidejűleg egynél több készüléke nem aktivizálható. A multiplex csatorna készülékei a számítógép hierarchiájának egészében rangsorolódnak. Több készülék egyidejű szolgálatkérésének kezelése hasonló elvek alapján történik, mint a megszakító rendszernél, de természetesen kizárólag szerkezeti úton, a programkontextus érintése és program segítsége nélkül. A szolgálatkérés kezelése a gépek jelentős többségénél független a megszakítórendszertől; az egysínű gépeknél a megszakításkéréskezeléssel együtt történik. A programfüggetlen csatorna egésze prioritást élvez az MPK hálózat minden más elemével szemben.

Mind a szelektor-, mind a multiplex csatorna felépítése esetenként egyszerűsíthető a gép és a készülékeket összefogó sín közé iktatott csatornavezérlő alkalmazásával.



4.7. ábra.
Készülékcsatlakozás a programfüggetlen csatornára
(a) egyetlen készülék; (b) és (c) egynél több készülék esetében

A csatornavezérlőbe vonhatók össze a készülékvezérlők közös (pl. előkészítő) áramkörei. N számú készüléket kiszolgáló szelektorcsatorna esetében így, N adatszám-regiszter, ill. címmutató helyett mindössze egy-egy regiszter szükséges. Multiplex csatornánál a csatornavezérlő bevezetése a regiszterek számában ugyan nem nyújt megtakarítást, de éppúgy, mint a szelektorcsatorna esetében is, igen leegyszerűsödik az előkészítés (4.7c ábra). A csatornavezérlő alkalmazásának jelentősége háttérbe szorul azokban a gépekben, melyekben a programozott és programfüggetlen csatornát egyetlen, funkcionálisan összevont sín szolgálja ki.

A többciklusú programfüggetlen átvitel módban is működő gép esetében előkészítő regiszterként a memória bizonyos rekeszei alkalmazhatók; a csatornavezérlő funkcióinak jelentékeny hányada pedig a processzor—memória együttesre hárítható át.

A munkamenet

A csatorna egy blokk átvitelével kapcsolatos munkájában négy munkaszakasz követi egymást:

1. előkészítés,
2. aktivizálás,
3. blokkátvitel,
4. lezárás.

Az előkészítés során a készülékvezérlővel (csatornavezérlővel) mindazokat az adatokat közölni kell, melyek az adatblokk automatikus átviteléhez szükségesek. Ezek közül az adatok közül eddig kettőt emeltünk ki: az adatszámot és az átvitel kezdő-címét. Adatszám helyett az adatot befogadó memóriaterület végcíme is megadható, mely esetben a készülékvezérlő minden egyes adat átadása kapcsán a folyó címet e végcímmel összeveti, s az átvitelt ott fejezi be, ahol megegyezést talál. Meghatározatlan számú adatból álló blokk végére zárókaraktert helyeznek el. A készülékvezérlőnek képesnek kell lennie e zárókarakter felismerésére. Rögzített formátumú blokkal rendelkező periférikus készülékek (pl. lemeztár) előkészítése során az adatszám esetenkénti megadására nincs szükség. A csatornaadatokon kívül a készülékvezérlővel a periférikus *eszköznek* szóló (esetleges) parancsokat is közölni kell. Így pl. lemeztár esetén meg kell adni az átvitelben szereplő blokk lemeztárbeli helyét (címét), ahová vagy ahonnan az átvitel történik.

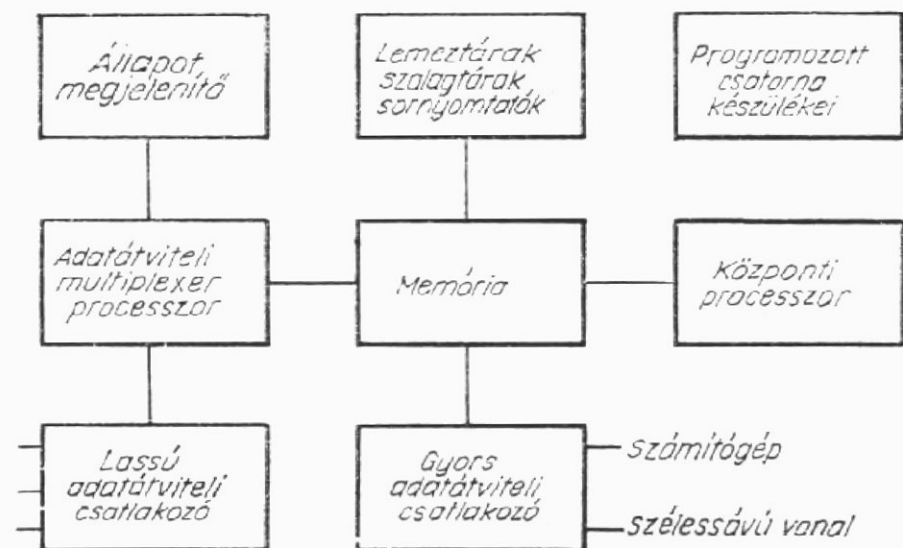
Az előkészített készülékvezérlőt külön utasítás aktivizálja, mely — ha eddig ez nem történt volna meg — az átvitel irányát is meghatározza. A készülék az előkészítés előírásainak megfelelő számú szolgálatkérést bocsát ki; minden egyes szolgálatkérés kapcsán a blokk egy adatának átvitelét hajtja végre a memória és a készülék között, programbeavatkozás nélkül.

Az átvitel az adatszám, a végcím vagy zárókarakter előírásainak megfelelően zárul. A készülék a feladat elvégzéséről a processzort megszakításkérés útján értesíti. A processzor a megszakításkérés alapján az adott készülék szolgálati rutinjára vagy a bevitt blokk feldolgozása céljából, vagy kivitel esetén a memória kiürített puffertérületének újrafeltöltése céljából (ha ez szükséges) tér rá. Megszakításkérés alkalmazása a blokkátvitel végének jelzésére nem minden esetben szükséges. Ha a programfüggetlen csatorna azonnali újraindítása nem követelmény, a program is megbízható azzal, hogy előírt "időpontban" megvizsgálja, vajon véget ért-e vagy sem az átvitel.

Az előkészítés és aktivizálás feladatát nemcsak a gép láthatja el programozott csatornáján keresztül, hanem a számítógéptől független rendszer, pl. másik számítógép is. Ismeretesek hálózatok, melyekben egy készüléket (pl. lemeztár) két számítógép is használ, s ennek megfelelően a készülék egyrészt mindkét gép programfügget-

len csatornájával kommunikál, másrészt mindkét irányból előkészíthető és aktíválható.

A számos lehetőséget felvillantandó, bár a berendezés nem mérés-technikai orientációjú, mégis érdemes egy pillantást vetni a Texas Instruments EMS II üzenet kapcsoló (message switching) rendszerére, melynek négybejártú memóriájához három különböző programfüggetlen csatorna csatlakozik (4.8. ábra).



4.8. ábra. Négybejártú memória, három programfüggetlen csatornával

4.1.4. Az autonóm csatornarendszer

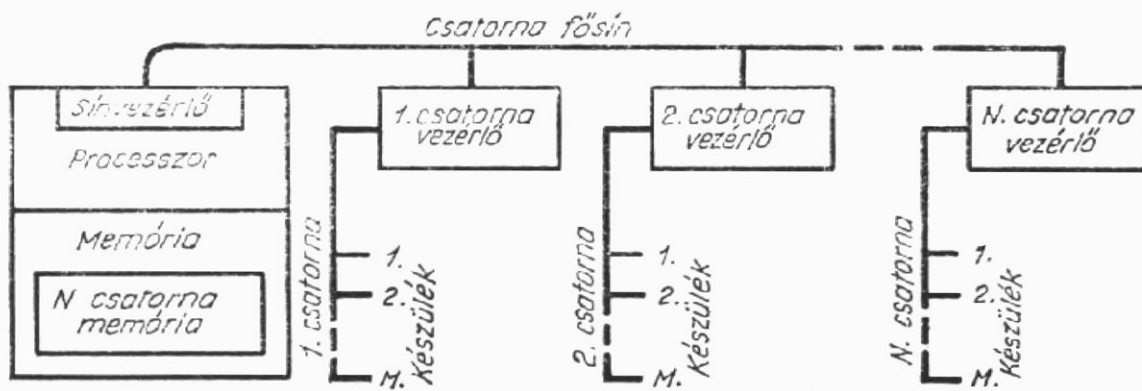
Az autonóm csatorna felépítése

Az autonóm csatorna a programfüggetlen csatorna továbbfejlesztése, jelentékeny számú blokkátvitelű periferikus készülék adatforgalmának lebonyolítására. Blokkátvitelű periferikus készülékek nagyobb (4...32 egységből álló) együttesére jelentékeny mennyiségű adat tárolása, ill. feldolgozása kapcsán lehet szükség, ami sokkal inkább nagy-, semmint kiszámítógép feladat. Ez a magyarázata annak, hogy autonóm csatornával kiszámítógépnél elvéve, s akkor is csak többsín-rendszerű gépnél találkozunk. Az egysínű gép a készülékek multiplex üzemét eleve lehetővé teszi; külön autonóm csatorna alkalmazása ott szükségtelen.

Az autonóm csatorna a processzort érintő programfüggetlen adatátviteli útvonalra épül fel, szerkezzetakarékossági megfontolás alapján többciklusú átvitel móddal, a programkontextus megzavarása és a program kitérítése nélkül a processzort is belevonva az adatátvitel munkájába. Tekintetbe véve még az autonóm csatorna által kiszolgált készülékek számát is világos, hogy a csatorna átbocsátóképessége a programozott csatorna és a programfüggetlen csatornáé közé esik.

Az autonóm csatorna egy lehetséges felépítésmódját a 4.9. ábra mutatja be. A programfüggetlen csatornához viszonyított legfeltűnőbb eltérés az egynél több (N) autonóm csatornavezérlő, melyek a géppel a fősínen át kommunikálhatnak, s melyek mindegyike legfeljebb M számú készüléket (készülékvezérlőt) összefogó csatornával áll kapcsolatban. Az autonóm csatorna mint rendszer egymástól függetlenül is dolgozni képes N csatorna szuperpozíciója.

A csatornarendszer minden egyes csatornájához: kezdőfeltételek, készülékparancs és készülékállapot tárolására szolgáló regiszterkészlet — csatornamemória — tartozik. A csatornaszámnak megfelelő N csatornamemória a főmemória állandó címekkel



4.9. ábra.
Autonóm csatornarendszer felépítése

meghatározott szektorában helyezkedik el. Egy csatornamemóriához pl. a következő regiszterek tartozhatnak:

- adatszám
- címmutató
- parancs
- végállapot.

Az első kettő szerepe ugyanaz, mint a programfüggetlen csatorna esetében; a parancsregiszter készüléknek szóló parancsot tartalmaz; a végállapot-regiszter adatátvitel befejeztével a csatorna által szolgáltatott készülékállapot elhelyezésére szolgál. A rendszer jellegzetessége, hogy egy csatornának egyidejűleg csak egy készüléke lehet aktív; a csatornamemória egyetlen készülékhez tartozó adatokat tartalmaz.

Üzem módok

Háromféle: szelektív, multiplex és reteszelt multiplex üzem mód lehetséges.

Szelektív üzem módban az egész csatornarendszernek csupán egyetlen készüléke lehet aktív, folytathat kommunikációt a memóriával. A kommunikáció lebonyolításában a processzor is részt vesz, ezért a blokkátvitel teljes időtartamára reteszelt (programfelfüggesztés).

Multiplex üzem módban minden egyes csatorna autonóm, egyidejűleg dolgozhat (csatornánként egy készülék aktivitása mellett) és pedig programfelfüggesztés nélkül. Mivel az adatátvitel munkájában a processzor ez esetben is részt vesz, a program csak azokban az időszakokban folytatódhat, melyekben az autonóm (vagy a programfüggetlen) csatorna nem él cikluselvéttel.

A *reteszelt multiplex* üzem mód az előbbtől csak abban különbözik, hogy az autonóm csatorna bármely folyamatának teljes időtartamára a programvégrehajtás felfüggesztődik.

Az üzem mód utasítással választható meg. A csatornarendszer átbocsátóképessége szelektív üzem módban a legnagyobb, multiplex üzem módban a legkisebb.

Vezérlőutasítások, állapotvektorok

A programfüggetlen átvitel módban dolgozó autonóm csatornának átviteli utasításai nincsenek, de hozzátartoznak bizonyos vezérlőutasítások. E vezérlőutasítások sorában

- indító és leállító,
- ellenőrző,
- megszakításkezelő.

utasítások különböztethetők meg.

Az indító utasítás a csatornarendszer meghatározott készülékét munkaállapotba helyezi, a leállító utasítás a készülék munkáját félbeszakítja. Az aktivizáló utasítás az üzemállapotot is meghatározza.

Az ellenőrző utasítás hatására a csatornarendszer adott készülék előírt állapotvektorát a processzorba továbbítja.

A megszakításkezelő utasítás a megszakításkéréssel jelentkező készülék azonosítása során játszik szerepet. Az autonóm csatorna az adatátvitelt programbeavatkozás nélkül végzi, megszakításkérést csupán valamely készülék feladatának befejeztével vagy hibahelyzetben állít.

A csatornarendszerhez meghatározott állapotvektorok tartoznak, melyek mindegyike az egyes készülékekre külön-külön értelmezhető. Az állapotvektorok bizonyos csatornaműveletek kapcsán automatikusan kerülhetnek a processzorba vagy csatornamemóriába, de közülük egyesek az ellenőrző utasítások segítségével is behívhatók.

A csatorna munkamenete

A csatorna alapfeladata: blokkátvitel. E feladat végrehajtásának menetében ugyanaz a négy fő munkaszakasz különböztethető meg, mint a programfüggetlen csatorna esetében: 1. az előkészítés, 2. az aktivizálás, 3. a blokk átvitele és 4. a lezárás.

Az előkészítést a program végzi, feltöltve annak a csatornának a memóriáját, melyhez az átvitelben résztvevő készülék tartozik. A memória feltöltése után, ugyancsak a program, aktivizáló utasítással indítja a készüléket. Ha a készülék — az előírt feltételek együttes teljesülése esetén — az utasítást elfogadja, az utasítás által előírt üzemmódban megkezdí munkáját. A blokkátadás lényegi részét megelőzően a csatornamemóriából lehívja a készülékparancsot, s esetleg a processzorhoz küldi az állapotvektorok egyikét. Ezután, időben egymás után, szolgálatkérő jeleket bocsát ki. Minden egyes szolgálatkérő jel hatására a processzor a csatornával együtt egy-egy adat átvitelét hajtja végre. Adatátvitel kapcsán a vonatkozó csatornamemória címmutatója automatikusan tovább lép, adatszámregiszterének tartalma pedig eggyel csökken; a processzor, ez utóbbi tartalom feltételes vizsgálatot is végez. Az adatátvitel az adatszám nullára csökkenésével fejeződik be. Ezt követően a csatorna beküldi a készülék végállapot-vektorát, majd a processzornál megszakításkéréssel jelentkezik.

Szolgálatkérés és megszakításkérés

Nem célunk az autonóm csatorna üzemi lehetőségeinek részleteire kitérni, de a szolgálatkérés és megszakításkérés kezelésének módjára mégis röviden rá kell mutatnunk. A szolgálatkérő jeleket a megszakító rendszertől függetlenül a csatornarendszer sínvezérlője és a processzorvezérlő együtt értelmezi. Tekintettel arra, hogy csatornánként mindig csak egy készülék lehet aktív, csatornán belüli rangsorolást nem kell végezni; rangsort csak a csatornák között kell felállítani.

A szolgálatkérés-kezelés a prioritással bíró csatorna szerkezeti úton való kiválasztásából és szolgálatának elfogadásból áll. További sajátosság, hogy a csatorna feladatától (állapotvektor-bevitel, parancskivitel, adatátvitel) függően a szolgálatkérő jelnek többféle változata van.

A szolgálatkérés kezelésével kapcsolatos eljárástól a megszakításkezelésé legszembetűnőbben abban tér el, hogy az utóbbi esetében a prioritási rangsor alapján történő csatornakiválasztást a megszakítást kérő készülék azonosításának is követnie kell; ez feltétele annak, hogy a processzor a készülékhez rendelt megszakítási rutinra rátérhessen.

A szolgálatkérések és a megszakításkérések kötött szintjeikkel a számítógép szolgálati hierarchiájában egy-egy osztályt foglalnak le. Az autonóm csatorna szolgálatkéréseinek osztálya az összes megszakítási szintnél magasabban, de a programfüggetlen csatorna szolgálatkéréseinek osztálya alatt helyezkedik el.

4.2. Csatlakozás (Interface)

4.2.1. A mérőhálózat elemeinek a géphez és egymáshoz való csatlakoztatása

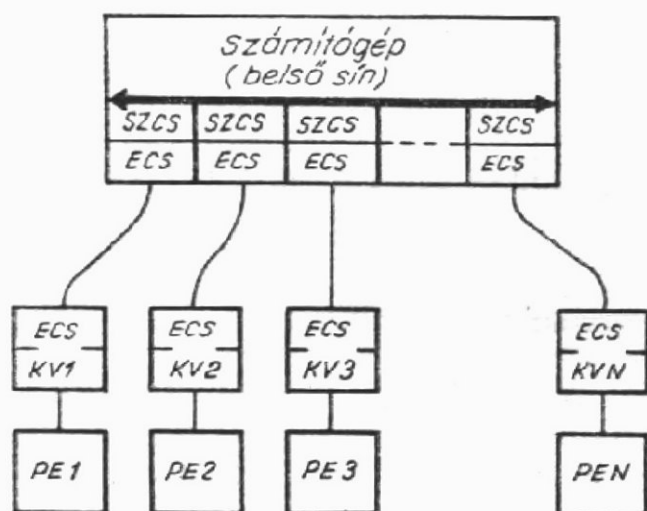
A gépcsatlakozás problematikája

A számítógép MPK hálózatához annak egy eleme (pl. készülék) meghatározott előírások betartásával csatlakoztatható. Csatlakozáson általánosságban valamely rendszer elemének a rendszer egészével való összeköttetése és kommunikációja módját értjük. A csatlakozást

1. a mechanikai előírás (csatlakozók, összekötő vezetékek, esetleges csatlakozókártyák specifikációi),
2. az elektromos előírás (jelek és esetleges táppontok feszültség- és áramszintjei),
3. a funkcionális előírás (a jelvezetékek és szerepük leírása; azoknak az eljárásoknak rögzítése, melyekkel a jelvezetékekre kapcsolódó egységek egymással kapcsolatba léphetnek és kommunikálhatnak; a csatlakozás nyelvének leírása)

definiálja. Ezek az előírások függetlenek a csatlakoztatott elem tulajdonságaitól. Az elemfüggetlen előírások esetenként elemfüggőkkel egészülhetnek ki, így pl. az adatforgalomban alkalmazott kód, valamint a csatlakozás kezelését végző programozási (software) eszközök definíciójával.

A különböző kis- és mikrogépek, irányítási célra alkalmazott kalkulátorok csatlakozása, sajnos ahány annyiféle. De még egy gépen belül is a többféle lehetséges átvitmódnak többféle csatlakozás felel meg. Az egységesség hiánya már az esetben is problémát jelenthet, ha meglévő rendszerhez kell új készüléket csatlakoztatni. Bár a gyártók gépeikhez többnyire a készülékek nagy választékát illesztik, az alkalmazónál mégis gyakran előáll egy még nem illesztett egység csatlakoztatásának nehézsége. A probléma megsokszorozódik, ha azzal az igénnyel élnek, hogy egy sor



4.10. ábra.

A készülékcsatlakozás egységesítése a csatlakozóegység kétfelé választása útján

SZCS — számítógép-csatlakozó; ECS — egységes csatlakozó; KV — készülékvezérlő; PE — periferikus eszköz

különböző készüléket számos különböző típusú gépet felölelő környezetben lehessen alkalmazni. N -féle készülék, M -féle gép esetén ugyanis e követelmény kielégítéséhez $(N \cdot M)$ -féle készülékvezérlőt kell kidolgozni. Ezzel a nehézséggel minden önálló készülékgyártó (sorukban a mérőkészülékgyárak), de a készülékek alkalmazói is közvetlenül szembetalálkoznak. Egy megoldásként: a készülékvezérlőt két részre bontják, csatlakozóegységét a géppel, vezérlő- és végrehajtó egységét a készülékkel építik össze, s a két rész közötti csatlakozást egységesítik (4.10. ábra). Ily módon géptípusonként egy (összesen tehát N) és készüléktípusonként egy (összesen M), mindössze tehát $(N + M)$ különböző egység szükséges.

A csatlakozás ökonomiáját ezen az úton keresve alakult ki és nyert a korábbi években alkalmazást az Egységes Angol Csatlakozás [British Standard Interface (B.S.I.): B.S. 4421, A Digital Input/Output Interface for Data Collection Systems]. A BSI elsősorban az adatfeldolgozás hagyományos periférikus eszközeinek illesztésére készült. Mérőrendszerekben általánosságban nem terjedt el, aminek egyik oka, hogy elektromos specifikációja a TTL környezettel inkompatibilis. Bár az inkompatibilitást megszüntető módosított változatokat is dolgoztak ki (pl. Network Standard Interface), időközben azonban a méréstechnika követelményeit figyelembe vevő egységes sínrendszerek nyomultak előtérbe.

A mérőrendszer elemei közötti csatlakozás

A rendszer elemének a rendszer egészéhez való csatlakoztatásának, az elemek egymáshoz kapcsolásának feladata a számítógéptől függetlenül, a méréstechnika terén is felmerült. Mielőtt még a számítástechnika betört volna, a méréstechnikában a csatlakozás elvi módszereinek és gyakorlati megoldásainak önálló, bár természetesen a digitális technika egészének hatása alatt álló fejlődése zajlott le.

Az elemek mérőhálózattá való összekapcsolásának első formája az elemek közötti egyirányú adatátviteli kapcsolat; az adatátadásban csak digitális alakú, készülékjellegű (mérési) adatok szerepelnek, parancsjellegű (program) adatok nem.

A mérőhálózat, ezen belül a csatlakozás fejlődésének második szakaszát a központi szerkezet (vezérlő) általi irányíthatóságra való törekvés determinálja. A mérőhálózat programirányítható elemekből áll, melyek a központi vezérlőszerkezethez készülék- és parancsjellegű adatokat egyaránt továbbító, dedikált vonalakkal csatlakoznak. A mérőhálózat és vezérlője (mely utóbbi háttérben szalagolvasó vagy kártyaolvasó áll) egyedi, és az egyes mérőkészülékek (modulok) csatlakozása is individuális.

A fejlődés következő lépcsőjén, egy-egy rendszeren belül egységesítik a csatlakozást. Kialakulnak olyan hálózatsémák is, melyek a központi irányítás elvének megtartása mellett a hálózati elemek láncszerű összekapcsolását is lehetővé teszik.

Végül a számítógéppel való találkozás során a fejlődésnek ez a vonala is az előbb említett egységes sínrendszerekhez vezet.

A mérőrendszerben alkalmazott csatlakozást determináló tényezők

Különböző célokra szolgáló, más-más szempontok alapján létrehozott mérőrendszerek hálózatai egymástól sok tekintetben eltérhetnek. Az alkalmazott csatlakozás adottságainak összhangban kell lennie a mérőrendszer követelményeivel. A csatlakozást meghatározó főbb tényezők:

- a mérőrendszer hálózati struktúrája,
- a programirányítású elemek milyensége és mennyisége,

- a mérőhálózaton belüli, valamint a hálózat és a vezérlő közötti információ-áram nagysága,
- a programirányítású hálózati elemek intelligenciafoka,
- az alkalmazói környezet,
- az alkalmazott vezérlő,
- a gazdasági megfontolások,
- a megbízhatósági követelmények.

A mérőhálózat struktúrája a különböző rendszerekre tekintve változatos képet mutat. A csatlakozás szempontjából leglényegesebb strukturális kérdés az elemek egymással való kapcsolata. A vezérlőcentrikus hálózatokban ez a kapcsolat szegényes, a hálózati elemek átviteli útvonalai direkt a központi adatfeldolgozó szervhez kapcsolódnak. Más hálózatokban az elemek láncszerűen kapcsolódnak egymáshoz, az elemek közötti adatáramlás jelentékeny. Egyéb kapcsolati lehetőségek is vannak. Az adatvonalak gyakran egyirányúak, esetenként azonban mindkét irányban folyik adatátvitel. Nyilvánvaló, hogy a különböző hálózati struktúrák a csatlakozással szemben is más-más követelményeket támasztanak.

Korábban említettük már, hogy a hálózat különböző jellegű modulokból, építő-kocka vagy mérőkészülék jellegű elemekből, esetenként mérőberendezés méretű integrált mérőkészülékként épülhet fel. A hálózat különböző építésmódjainak különböző optimális csatlakozásmódok felelnek meg.

A csatlakozás előírásaira jelentős befolyást gyakorol a rendszer programozható elemeinek száma. Fordítva, a csatlakozás specifikációjában általában megtalálható a maximális csatlakozóhelyek, a csatlakoztatható elemek száma.

A csatlakozási előírásokra a hálózatban alkalmazott elemek működési sebességével kapcsolatos követelmények is kihatnak. A működési sebesség szempontjából homogén hálózatban szinkron kapcsolatú csatlakozás alkalmazható, ahol azonban az egyes elemek működési sebessége jelentékenyen eltérhet, s az elemek közötti távolság is határozatlan, egyedül az aszinkron kapcsolat nyújt kielégítő megoldást.

A csatlakozás fizikai realitásához hozzátartozik a rajta különböző körülmények között átbocsátható információáram. Fordítva, az alkalmazott csatlakozás minimális átbocsátóképességét — követelményként — a hálózat információforgalmának intenzitása determinálja.

Az automatizált mérőrendszer információforgalmában készülékjellegű adatok mellett vezérlés- (program-) jellegű adatokat is találunk. Mind a programjellegű, mind az adatjellegű információáramok nagyságát jelentékeny mértékben befolyásolhatja a vezérelt (programirányított) elemek intelligenciaszintje. Az áramköri integrációs technika előrehaladtával kialakul a kiterjedt lokális vezérlési és adatredukciós lehetőséggel rendelkező intelligens mérőkészülék típusa. Az ezekből felépített hálózat elemei közötti adatforgalom lényegesen kisebb intenzitású, mint a hagyományos elemekből összeállított hálózaté. A hálózati elemek intelligenciaszintjének növelése a csatlakozás átbocsátóképessége tekintetében támasztott követelmények enyhítésével jár.

Az alkalmazói környezet a csatlakozás flexibilitásának kérdését érinti. Laboratóriumi viszonyok között gyakran merül fel újabb és újabb feladatra alkalmas mérőrendszer iránti igény. A skála is széles: egy-két mérőkészülék és egyszerű vezérlő együttesétől, számos készülék és számítógép együtteséig terjedhet a skála. Az is fontos, hogy az elképzelt és megtervezett rendszer gyorsan összeépíthető legyen. Termelői környezetben, adott feladatkörben relatíve hosszú időn át használják a mérőrendszert, amely azért ugyan nagyobb kezdőerőfeszítéssel (és merevebb struktúrával), ugyanakkor azonban gazdaságosabb felépítésmóddal tervezhető. Nyilvánvaló, hogy

a laboratóriumi környezet a csatlakozást illetően a termelői körülményekénél lényegesen nagyobb flexibilitási igényt támaszt.

A flexibilitás kérdéséhez nem csak a mérőhálózat, hanem a vezérlés problematikája is hozzátartozik. A csatlakozás a különböző szintű rendszervezérlőkhöz való gazdaságos adaptálhatósága szempontjából is jellemezhető.

A csatlakozás egyik fontos értékmérője a csatlakozási helyenként felmerülő költséghányad, továbbá e költséghányad függése a csatlakozóhelyek számától. A méret és költségskála alsó tartományában elhelyezkedő mérőrendszerek nyilvánvalóan azzal a követelménnyel élnek, hogy az egy csatlakozóhelyre eső költség kevés csatlakozóhely esetében is mérsékelt legyen. Nagyobb rendszereknél a csatlakozás kezdőköltsége lehet nagyobb, ha az egy csatlakozási helyre eső költség a rendszer méreteinek arányában egyébként ésszerű szinten mozog.

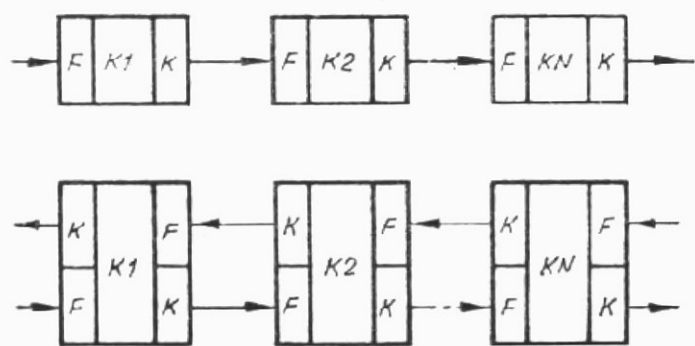
A rendszerre előírt általános megbízhatósági követelmények a csatlakozásra is kiterjednek. A csatlakozási helyek számának és a rendszer geometriai kiterjedtségének növekedésével a megbízhatóság problémája mindinkább előtérbe lép. A fokozott megbízhatósági követelmények különösen a sokpólusú csatlakozók érintkezőszáma és az adatátvitel technikája (parolatechnika) tekintetében állíthatnak követelményeket.

Egységes csatlakozási rendszerek

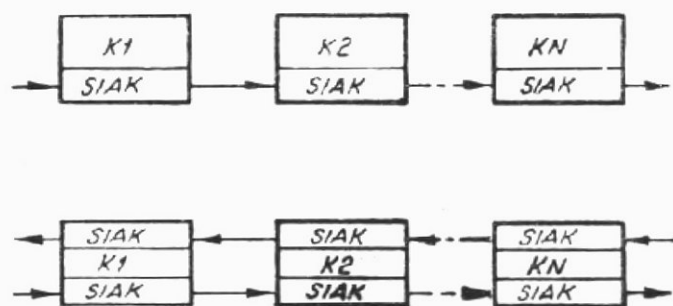
A mérőrendszer tervezési és megvalósítási idejének lerövidítése, a költségek csökkentése, a különböző gyártók által előállított mérőhálózati elemek (mérőkészülékek stb.) rendszerekben való általános alkalmazhatósága (kompatibilitása) szempontjából alapvető jelentőségű a csatlakozás lehetőleg nemzetközi szintű egységesítése. E felismerés jegyében számos országban, számos intézmény keretében folytatott egyidejű munka eredményeképpen különböző csatlakozási rendszerek alakultak ki. Minden a mérőrendszerek oldaláról felmerülő (előbb főbb vonásaiban áttekintett) követelményt optimálisan kielégítő csatlakozás nincs — a többféle kikristályosodott irányzatnak is ez a magyarázata.

A szélesebb körben elsőként rendszeresített BSI csatlakozás azon túl, hogy elektromos paraméterei a jelen áramköri technológiával inkompatibilisek, a mérőrendszer szemszögből nézve egyébként sem előnyös tulajdonságú. A 4.10. ábrára tekintve, s felidézve a kisgépek csatornarendszereinek felépítésmódját, látható, hogy minden számítógép-oldali BSI csatlakozóegység mögött még egy belső síncsatlakozó egységnek is meg kell húzódnia. Ez természetesen nem valami gazdaságos. Ehhez jön még, hogy a BSI byte-soros átvitel módja általában nem illeszkedik sem a gép, sem a mérőkészülék adatformátumához, ezért többnyire a belső csatlakozóegységek (SZCS) mindegyikében adatkonverziót is végre kell hajtani. A BSI csak egyirányú (transmitter → receiver) adatforgalmat tesz lehetővé. Amennyiben a gép és a készülék között kétirányú adatforgalmat kell megvalósítani, minden készülékhez két pár BSI csatlakozó szükséges. A BSI-t készülékek láncszerű csatlakoztatására lehet ugyan alkalmazni, de egyirányú adatforgalom esetén készülékenként egy-egy küldő (transmitter), ill. fogadó (receiver) típusú csatlakozóegység szükséges; kétirányú forgalom esetén pedig a készülékenkénti csatlakozóegységek száma négy (4.11. ábra).

Kifejezetten mérőrendszerek céljára dolgozták ki a KGST keretén belül a BSI-hez hasonló SIAK rendszert; elsősorban láncszerű hálózatokon belüli csatlakoztatásra. A SIAK a BSI-nél valamivel gazdaságosabb. Egyirányú adatforgalom esetén készülékenként csupán egy, kétirányú adatforgalom esetén készülékenként két csatlakozóegység alkalmazását igényli (4.11. ábra). Különbözik a két rendszer adatvonalának



BSI



SIAK

4.11. ábra.

A BSI és SIAK csatlakozásmód;

K1...KN készülékek; F — BSI fogadó; K — BSI küldő

szervezése is, amennyiben a BSI vonalszélessége kötött (1 byte), a SIAK vonalszélessége tetszőlegesen választható. A BSI aszinkron (parola), a SIAK szinkron kapcsolatú.

A sínrendszerű SIAL a SIAK társrendszere, mely nemcsak egyes készülékek, hanem a SIAK segítségével láncszerűen összekapcsolt hálózati elemek együttesének számítógéphez vagy más központi vezérlőhöz való csatlakoztatására is szolgál.

E könyv írásakor küszöbön áll két csatlakozási rendszer, a CAMAC és a Hewlett-Packard, IEC (International Elektrotechnical Commission) nemzetközi szabványként való elfogadása.

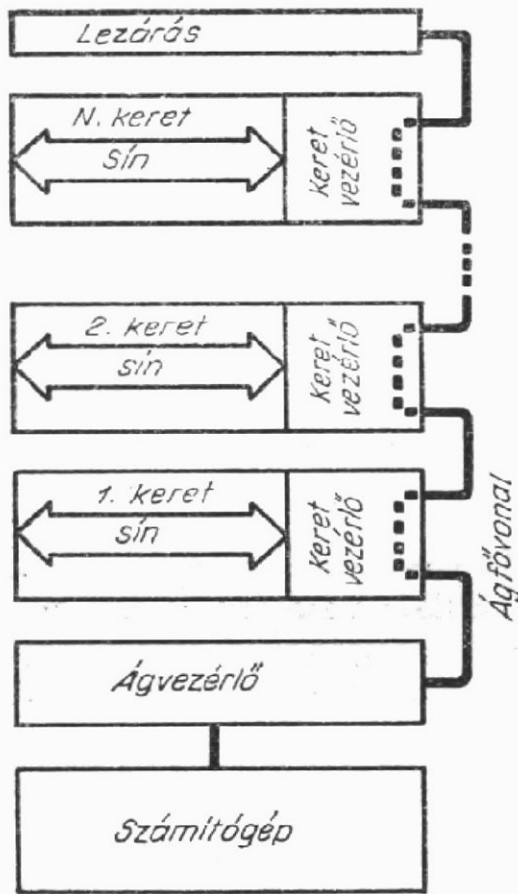
A CAMAC-ot a magfizikai mérőrendszerek szükségletei alapján az EURATOM keretében működő ESONE bizottság dolgozta ki, majd a későbbiekben bizonyos az ESONE-val együttműködésben végzett apróbb változtatások után, az Egyesült Államok Atomenergia Bizottsága is átvette (AEC NIM Committee). A rendszert számos országban, így a szocialista országokban is hosszú ideje alkalmazzák, és pedig nemcsak a magfizikai méréstechnikában, hanem a folyamatirányítás és a műszaki méréstechnika területén is.

A Hewlett—Packard-rendszer a műszaki méréstechnika igényei és szükségletei szerint alakult ki, a vállalat több évtizedes méréstechnikai tapasztalatai, valamint a számítógép-orientált mérőrendszerek területén folytatott igen széleskörű praxisa alapján.

A két csatlakozási rendszer egymástól sok tekintetben különbözik, s a méréstechnika más-más szükségleteit elégítik ki. A CAMAC kifejezetten számítógéporientációjú, modulárisan bővíthető csatlakozás, mellyel igen sok elemet magába foglaló mérőrendszerek valósíthatók meg. A hálózat mérőmoduljait összefogó sín szinkron csonka sín, tehát kötelezően belső sín, melyhez előírt méretű és meghatározott számú modul befogadására szolgáló keret tartozik. Számos keret csatlakoztatására egy második sínrendszer szolgál (4.12. ábra). E fizikai realizációja alapján a CAMAC elsősorban kisméretű mérőhálózati modulok csatlakoztatására, vagy nagyobb mérőkészülékek áramköri kártyán elhelyezett készülékvezérlőinek, esetleg csatlakozóegységeinek befogadására alkalmas. További lényeges jellemzője a vezérlőcentrikusság. A mérőhálózati modulok egymással közvetlenül nem vagy legalább is nagyon korlátozott és egyedi módon kommunikálhatnak. Egyik modultól a másikhoz adatátadás csak a keretvezérlőn át történhet.

A HP-csatlakozásnak — ellentétben a CAMAC-kal csak egy csatlakozási szintje van, és ezen is kevesebb (14) a csatlakozási hely, mint a CAMAC-nál (ahol 23).

Ez természetesen nem zárja ki annak lehetőségét, hogy nagyobb rendszerek kialakításánál egynél több HP-sín csatlakozzék a számítógépre. A HP-csatlakozás teljes sít alkalmaz, mely belső és külső sínként egyaránt alkalmazható (4.12. ábra). A készülékvezérlőt belső sín alkalmazása esetén sem kell a sín mellé rendelni — a sín és a reá csatlakozó funkcionális egység összekapcsolása sokpólusú csatlakozóval



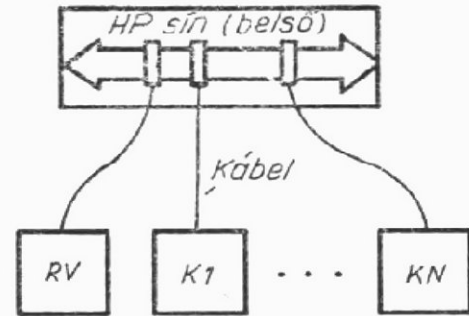
CAMAC ágfővonalal

4.12. ábra.
CAMAC és HP—IEC csatlakoztatásmódok

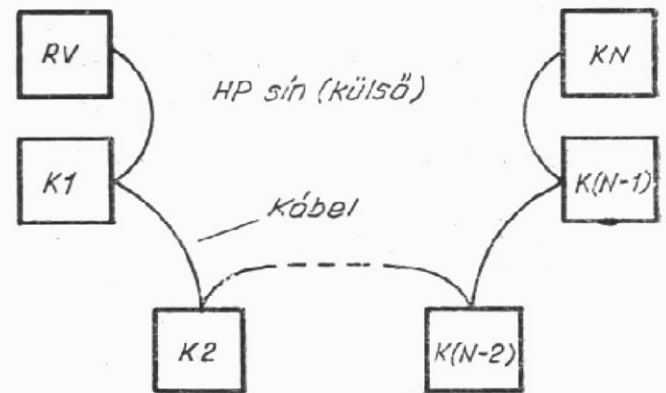
és kábellel oldható meg. A csatlakozás ezen adottsága inspirálja, hogy a mérőrendszer hálózati moduljaiként standard készülékvezérlőkkel (melyek intelligenciafoka tetszőlegesen emelhető) egybeépített zárt mérőkészülékeket alkalmazzanak. A HP-csatlakozáson át a készülékek egymással nemcsak a vezérlőn keresztül, hanem közvetlenül is kommunikálhatnak. E tulajdonság révén a csatlakozás vezérlőcentrikus és láncszerű információkapcsolatok létesítésére egyaránt alkalmas.

A CAMAC információ-átbocsátó képessége lényegesen nagyobb, mint a HP-csatlakozásé, egyrészt, mert a CAMAC adatvonala 24 bit, míg a HP csupán 8 bit szélességű, másrészt, mivel a cím és parancsátvitel a CAMAC-nál külön vonalak útján, a HP-rendszerben azonban ez is az adatvonal útján történik.

A CAMAC és a HP rendszerre jelenlegi, de még inkább várható jelentőségükre való tekintettel részletesebben kitérünk. Főként a funkcionális tulajdonságokat ismertetjük és az alkalmazó átfogó tájékoztatása lebeg szemünk előtt. A pontos, formális leírásra az irodalomjegyzék utal.



RV - rendszervezérlő; K - készülék

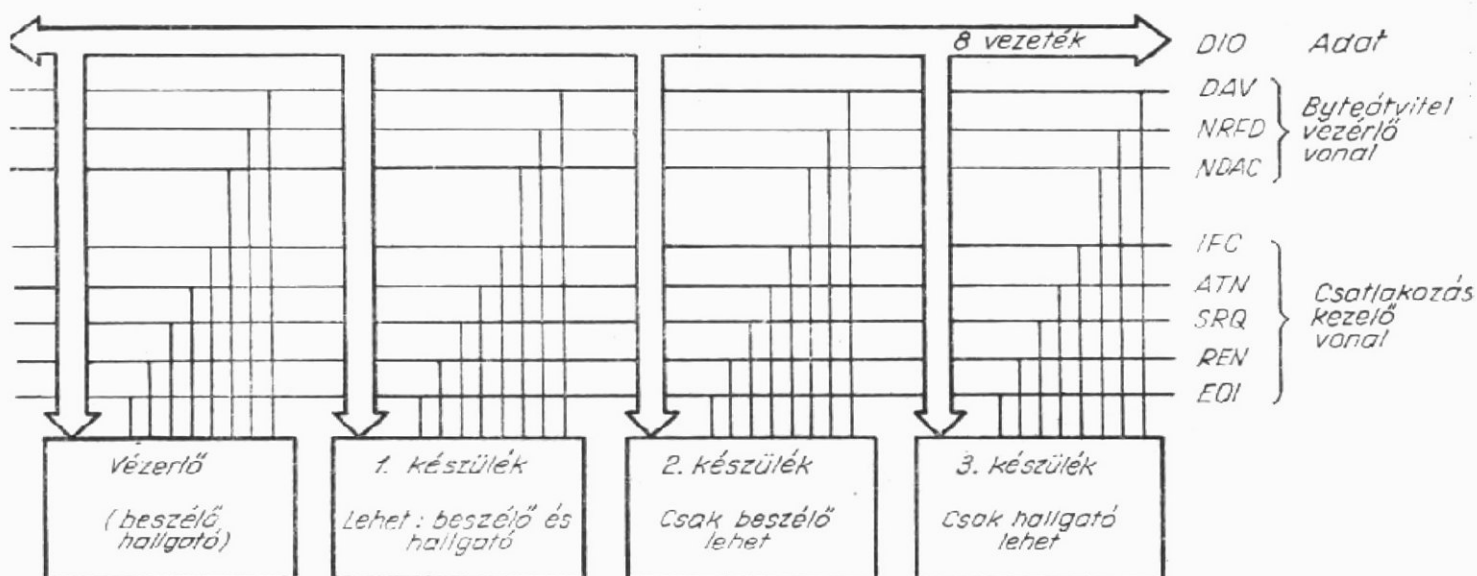


A HP-csatlakozás két alkalmazásmódja

4.2.2. A HP—IEC csatlakozás

A sín

A HP-rendszer sínje 8-vezetékes adatvonalból és 8-vezetékes szolgálati vonalból áll. Az összesen 16 vezeték kivétel nélkül társasvezeték (4.13. ábra). Az adatvonal kétirányú, aszinkron kapcsolatú, s egyaránt szolgál készülékvonatkozású (mérési adatok, programadatok, állapotvektorok) és parancsjellegű, irányító adatok átvitelére. A szolgálati vonal a háromvezetékes byte-átvitel-vezérlő és az ötvezetékes kezelővonalra különíthető el. Az előbbi, parolajeleivel, byte átvitelének lebonyolítását végzi, az utóbbi különböző üzemállapotok meghatározására, ill. jelzésére szolgál.



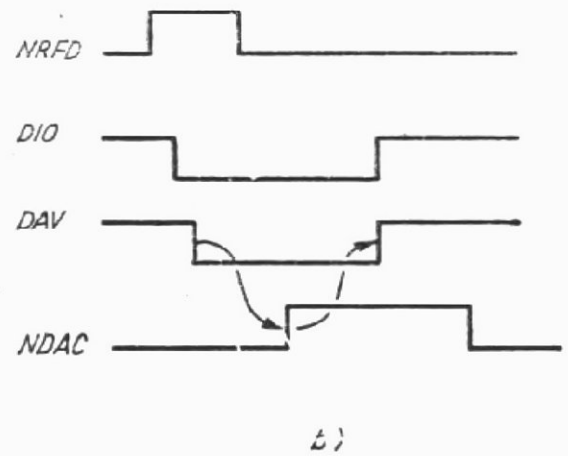
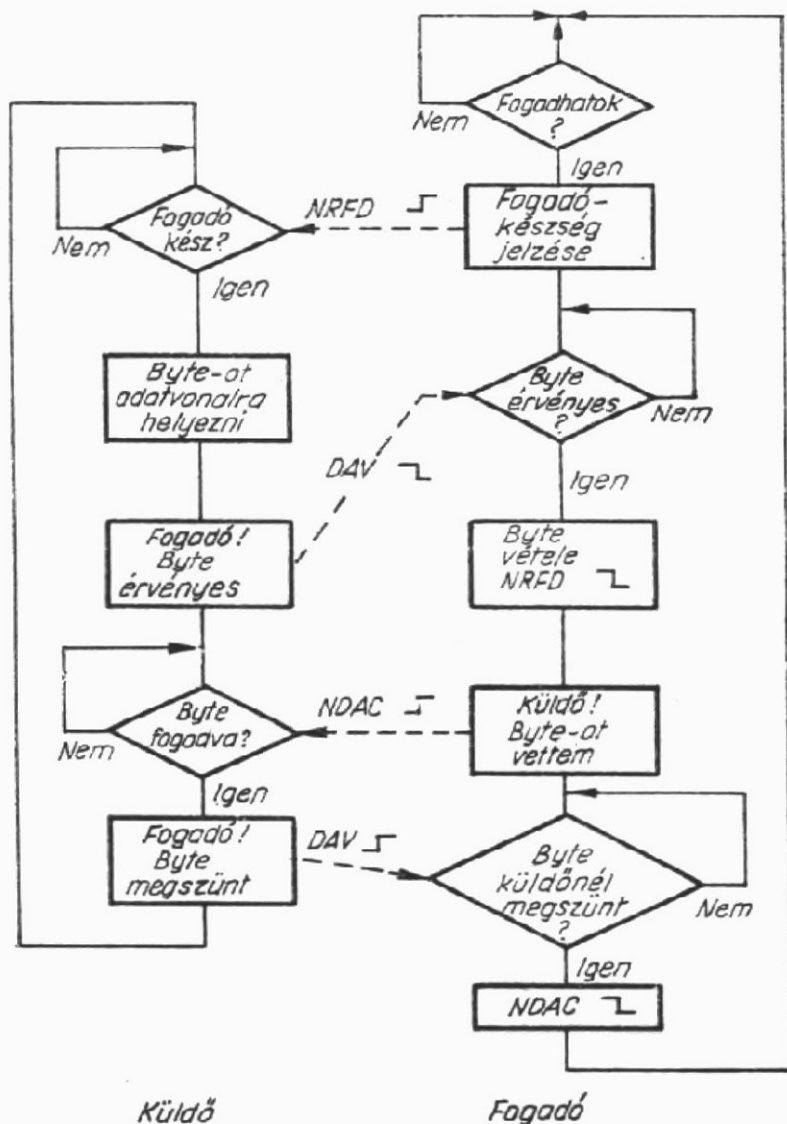
4.13. ábra.
HP—IEC sín

Byte-átvitel

Az adatvonal forgalmában az átvitel alapeleme a 8-bites byte. Ennél hosszabb adatok (üzenetek) továbbítása byte-soros módon történik. A byte átvitele — tartalmától függetlenül — egységesen, parolajelek által koordinálva történik. Küldőnek nevezve a byteot kibocsátó, és fogadónak a byteot vevő egységet és feltételezve, hogy egyidejűleg csak egy küldő, de egynél több fogadó is lehet, a byte átvitele a következőképpen megy végbe. Az NRFD vezeték mindazoknak az egységeknek állapotát regisztrálja, melyek fogadóként szerepelnek. A vezeték alsó (feszültség) szintje vételkésztség hiányát, felső szintje az összes fogadó vételkésztségét jelzi. A küldő az adatvonalra csak az esetben küldhet byte-ot, ha az NRFD vezetéken felső logikai szintet észlel; a byte kibocsátását és érvényességét a fogadó(k)nak a DAV vezeték (szintjének levitelével) jelzi. Azt, hogy az összes fogadó vette a byte-ot, az NDAC vezeték alsó → felső szintváltása igazolja. Ez a front a küldőhöz érve DAV-ot is visszaállítja. Miután NDAC alsó szintje is helyreállt, az NRFD vezeték szintjétől függően új átvitel kezdhető. Az átvitel lefolyása a 4.14. ábra alapján követhető nyomon.

Rendszerállapotok; szervezőmód

A sínre maximálisan 15 funkcionális egység csatlakoztatható. Az egységek egyike — eredeti rendeltetését tekintve — rendszervezérlő, a többi pedig készülék. Az eredeti rendeltetéstől függetlenül a rendszer működése során az egységek 1. vezérlő, 2.



4.14. ábra.
 Byte-átvitel a HP—IEC csatlakozási rendszerben
 (a) a byte-átvitel algoritmus; (b) küldő és fogadó kapcsolatlétesítésének jelábrája

beszélő vagy 3. hallgató szerepkörét tölthetik be (természetesen csak azt, amelyet alapfunkciójuk és logikai felépítésük eleve megenged).

Az ATN vezeték alsó-felső szintjével a rendszer két különböző állapota: a parancs-állapot, ill. az adatállapot definiálható. Az előbbiben a rendszer az adatvonal byte-ját parancsjellegű, az utóbbiban pedig készülékjellegű adatként értelmezi.

Parancsállapotban mindig csak egy vezérlő működhet, melynek üzeneteire nézve az összes többi egység fogadóképes. Adatállapotban a sínen háromféle egység különböztethető meg: beszélő, mely üzenetet bocsát ki magából, hallgató, mely veszi az üzenetet és passzív egység, mely a kommunikációban nem vesz részt. Beszélő (ahogy vezérlő is) mindig csak egy, hallgató azonban egynél több is lehet.

Minden olyan egység, mely (momentán) vezérlő szerepét tölti be, a többi egységekkel egységes módon kommunikál. Míg azonban a rendszervezérlő autonóm, a rendszer munkájába bármikor, előkészítés nélkül beavatkozhat, a készülék csupán a rendszervezérlő által reáruházott módon és időszakosan láthat el meghatározott vezérlési feladatot, melynek lezárulta után az irányítás visszaszáll a rendszervezérlőre. Egyedül a rendszervezérlő privilegiuma az IFC vonal, mellyel a rendszer teljes egészét — annak momentán állapotától függetlenül — bármikor kiindulási állapotba viheti vissza, továbbá a REN vonal, mellyel a készülékek helyivezérlés/távvezérlés üzemi-állapota választható meg.

A byte-átvitel mechanizmusa minden esetben egységes. Parancsállapotban küldő lehet a momentán vezérlő, fogadó pedig a kibocsátott parancs értelmétől függően egy vagy valamennyi jelenlevő többi egység. Adatállapotban a küldő szerepét a beszélő, a fogadót a hallgató (esetleg több hallgató) játssza. Az NRFD és az NDAC vonal a reákapcsolódó fogadókkal ÉS kapcsolatban áll. Ez annyit jelent, hogy NRFD csak akkor jelez fogadóképességet, ha a reákapcsolódó összes fogadó (pl. hallgató) már fogadóképes. Ugyanígy az NDAC jele a küldő felé csak akkor fut ki, ha már a leglassúbb fogadó is vette a számára küldött byte-ot. A byte-átvitel időtartamát az átvitelben szereplő leglassúbb fogadó determinálja.

Parancsok

Parancsot a rendszer parancsállapotában (ATN logikai igen) vezérlő generálhat. Formailag az összes parancs 7-bites, a byte nyolcadik bitje kihasználatlan marad. A tartalmat illetően általános és címtartalmú parancsok különböztethetők meg. Az általános parancsok az összes a sínen levő egységnek szólnak, a címtartalmúak viszont csupán annak az egységnek, melynek címét tartalmazzák. A rendszer irányítására szolgáló legalapvetőbb parancsokat a 4. 1. táblázat sorolja fel. A hallgató-címző és a beszélő-címző parancsok az általuk megcímzett egységre a hallgató, ill. beszélő szerepkörét ruházzák; az egységek — a rendszer adatállapotában — mint ilyenek hajtják végre feladatukat. A címző parancsok 5-bites címrésze 32 különböző címet nevezhet meg, de ezek közül csak 31 használható fel; az 1111 cím alkalmazása tilos. Tekintve, hogy a rendszernek csak egyetlen beszélő egysége lehet, egynél több, egymás után végrehajtott beszélő-címző parancs közül mindig csak a legutolsó érvényesül, a megelőzők hatása törlődik. Ezzel szemben egymás után végrehajtott

4.1. táblázat

Parancs	Értelme
X 0 0 A ₅ A ₄ A ₃ A ₂ A ₁	általános parancsok
X 0 1 A ₅ A ₄ A ₃ A ₂ A ₁	hallgató-címző
X 1 0 A ₅ A ₄ A ₃ A ₂ A ₁	beszélő-címző
X 0 1 1 1 1 1	hallgató-visszaállító
X 1 0 1 1 1 1	beszélő-visszaállító

hallgató-címző parancsokkal a lehetőségek határán belül tetszőleges számú hallgató (max. 14) jelölhető ki. Az 1111 cím az aktív készülékekhez, és pedig 01₂ kóddal társítva az összes hallgatóhoz, 10₂ kóddal társítva az egyetlen beszélőhöz szól. A visszaállító parancsokkal az aktív egységek passzív állapotba helyezhetők.

Tekintettel a 7-bites formátumra, a parancsok a paritásbit nélküli ASCII kód jeleivel azonosíthatók, a rendszer ASCII jelekkel vezérelhető.

Szolgáltatás és kezelése

Bármely folyamatállapotú készülék a rendszervezérlőtől, az SRQ vezeték alsó szintre állításával kérhet szolgáltatást. A készülékek az SRQ vezetékkel huzal-VAGY kapcsolatban állnak. A rendszervezérlő a számára megengedhető időpontban forrás-

azonosítási eljárással lát hozzá a szolgálatkérés(ek) feldolgozásához. Mindenekelőtt a rendszert ATN vonallal parancsállapotba helyezi, s ugyanakkor az EOI vezeték is alsó szintjére kapcsolja. Ez utóbbi hatására a szolgálatkérésre jogosult készülékek mindegyike a hozzá rendelt DIO vezetéken megjelenteti SRQ állapotát. Az így adódó bináris állapotvektor a rendszervezérlőbe átvihető és előírt módon megvizsgálható, majd a vizsgálat eredményétől függő eljárás lefolytatható.

Az EOI vezeték nemcsak a forrásazonosítás folyamatában, hanem a rendszer adatállapotában is szerepet játszhat, amennyiben byte-blokk átvitele kapcsán az átvitel végét jelzi.

A készülék kezelésmódjának megválasztása

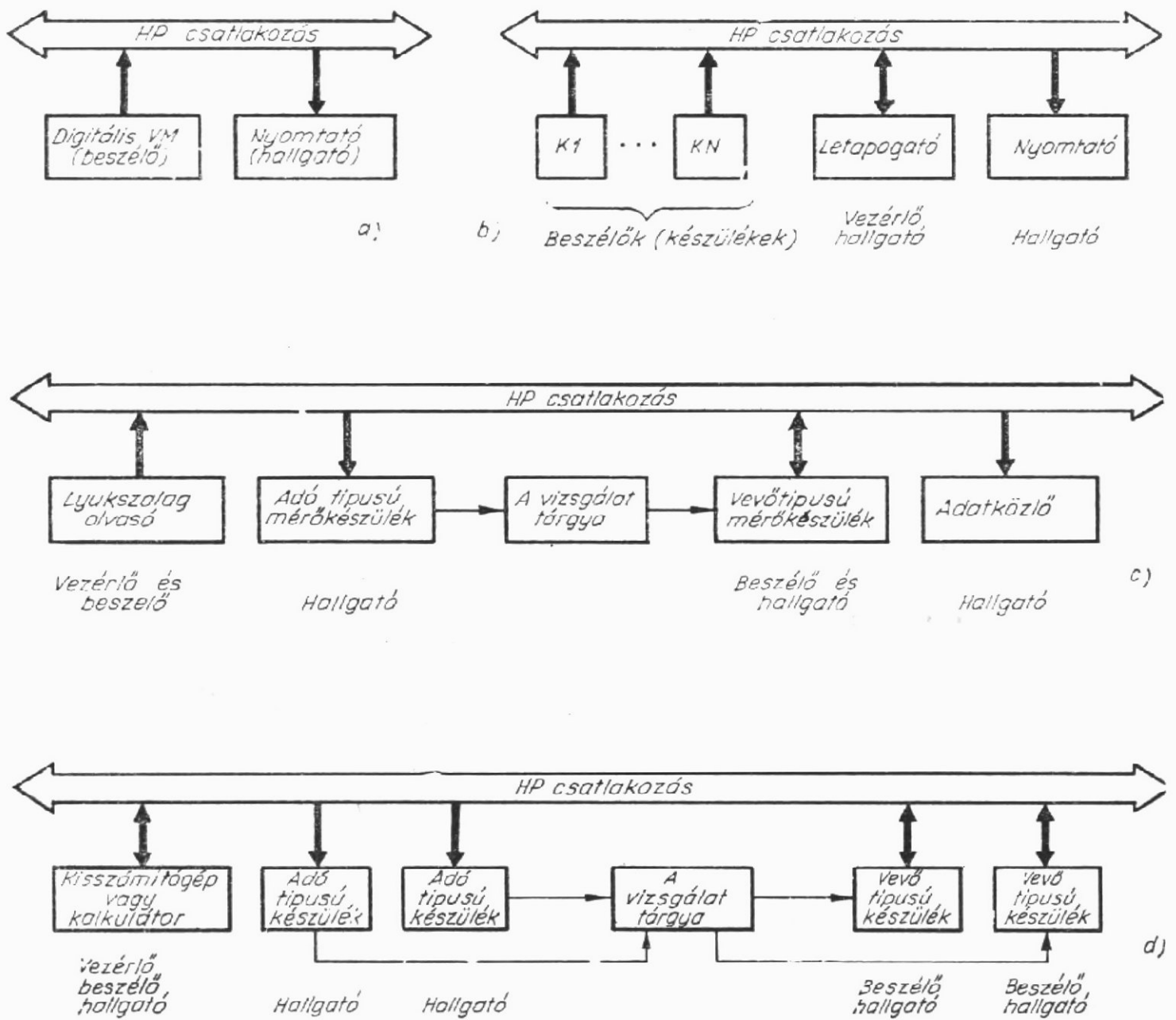
A készülékek saját előlapszerveikkel vagy távvezérléssel kezelhetők („programozhatók”). A helyi/táv üzemmód a rendszervezérlő irányítása alatt álló REN vezeték szintjével központilag választható meg. Egyes készülékek rendelkeznek olyan helyi átkapcsolóval, mellyel a REN vezeték rendelkezése megmásítható, REN távvezérlés üzemmódjában pl. az adott készülék helyi üzemmódba kapcsolható. Másrészt az általános jellegű parancsok egyikével, e helyi kapcsoló állásától függetlenül, a rendszervezérlő az összes készüléket helyi, egy másikkal pedig táv-üzemmódba állíthatja. A REN, a kapcsoló és a parancs növekvő rangsort képez.

Szerkezeti paraméterek

A hajtó- és vevőáramkörökre nézve a TTL logikai szintek kötelezők (felső szint $> 2,4 \text{ V}$, alsó szint $< 0,4 \text{ V}$). Schmitt-vevők és nyitott kollektorú hajtók alkalmazhatók általános célokra, fokozott sebességi követelmények esetén pedig háromállapotú hajtók használhatók. Az összes jelvonal az alkalmazott csatlakozókábel hullámellenállásánál lényegesen nagyobb ohmos terheléssel van lezárva. Ily módon a terhelés a funkcionális egységek között megoszlik, és a csatlakoztatásra passzív kábelek szolgálhatnak.

Alkalmazási szempontok

A HP-csatlakozás rendszervezérlőjének elsődleges feladata, hogy kijelölje a sínen át kommunikáló egységeket, egyidejűleg megadva szerepkörüket. Eltérően más megoldásoktól, a rendszervezérlő nem feltétlenül programozza a készülékeket, vagy végez adatfeldolgozást. A mérőrendszer a HP-csatlakozás felhasználásával a legkülönbözőbb vezérlési szinteken építhető ki. A legalacsonyabb szinten, amikor az egymáshoz csatlakozó készülékek állandó feladatkört látnak el, vezérlőre egyáltalán nincsen szükség. (A vezérlő feladatát a szerepkörök kiosztásával az ember látja el.) A 4.15. ábra különböző szintű rendszereket szemléltet. Az (a) rendszernek nincs vezérlője; a digitális voltmérő beszélőként, a sornyomtató hallgatóként működik, állandó jelleggel. A (b) hálózat vezérlője egyszerű letapogató, mely a beszélő szerepkörét időben egymás után osztja ki a vevő jellegű mérőkészülékeknek; állandó hallgató a sornyomtató. A (c) vázlat az előző két, „kézi programozású” hálózattal szemben lyukszalag-programozású rendszert mutat be. Végül a (d) ábrán egy a csatlakozás minden lehetőségét kihasználó összeállítás látható.



4.15. ábra.
HP—IEC csatlakozással szervezett különböző szintű mérőrendszerek

4.2.3. A CAMAC csatlakozás

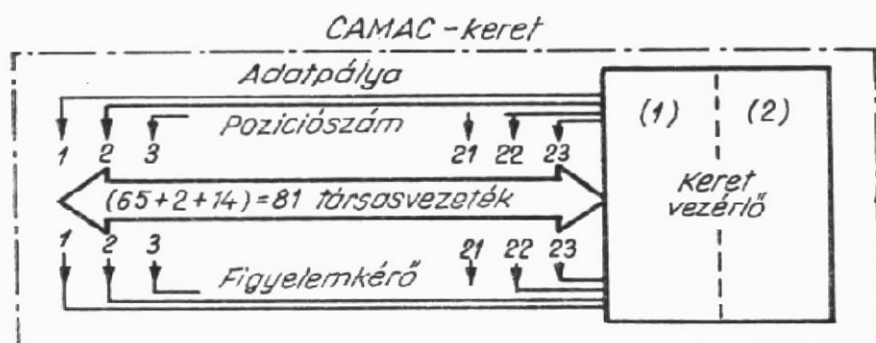
A rendszer felépítése

A mérőhálózat elemeinek (moduljainak) egységes csatlakoztatására szolgáló sín a CAMAC leírás — külön névvel — adatközlőnek (dataway) nevezi. Az adatközlő forgalmát lebonyolító, ún. keretvezérlő (crate controller) lehet önálló, vagy csatlakozhat számítógépre. A rendszerkiépítés magasabb szintjén számos adatközlő keretvezérlője előírt jellemzőjű CAMAC fővonalra csatlakozik, melyet vonalvezérlő illeszt a számítógéphez.

Az adatközlő társasvezetékekből és sugárvezetékekből komponált csonka, belső sín. A sín a CAMAC előírás szerinti keret sokpólusú lapcsatlakozóinak huzalozásában realizálódik. A keret 25 egyenlő szektorra oszlik fel, melyek közül (legalább) kettőt a keretvezérlő foglal el, a többi 23 csatlakozási pozícióként áll rendelkezésre. Minden pozícióba egy-egy önálló csatlakozóegység, teljes készülékvezérlő vagy akár teljes

funkcionális egység dugaszolható, egy funkcionális egység azonban több pozíciót is lefoglalhat. Az egy vagy több pozíciót elfoglaló egységet a CAMAC szóhasználat modulnak nevezi. Adott esetben a keretvezérlő kettőnél több szektorra is kiterjedhet, lefoglalva a 23 csatlakozási pozíció bizonyos hányadát.

A sín 81 társasvezetékéből 65 kötött feladatú jelvezeték, 2 szabadon felhasználható, 14 tápvezeték. 23 sugárvezeték az N pozíciószámok (station number) megadására, és ugyancsak 23 sugárvezeték az L figyelemkérő (look-at-me, LAM) jelek begyűjtésére (4.16. ábra) szolgál. Mindezek a vezetékek a pozíciónként 86 pólusú csatlakozó



4.16. ábra.
CAMAC keret szervezési sémája

83 érintkezőjét kötik le. A maradék 3 érintkező a mérőhálózaton belüli eseti, egyedi összeköttetésekre használható fel.

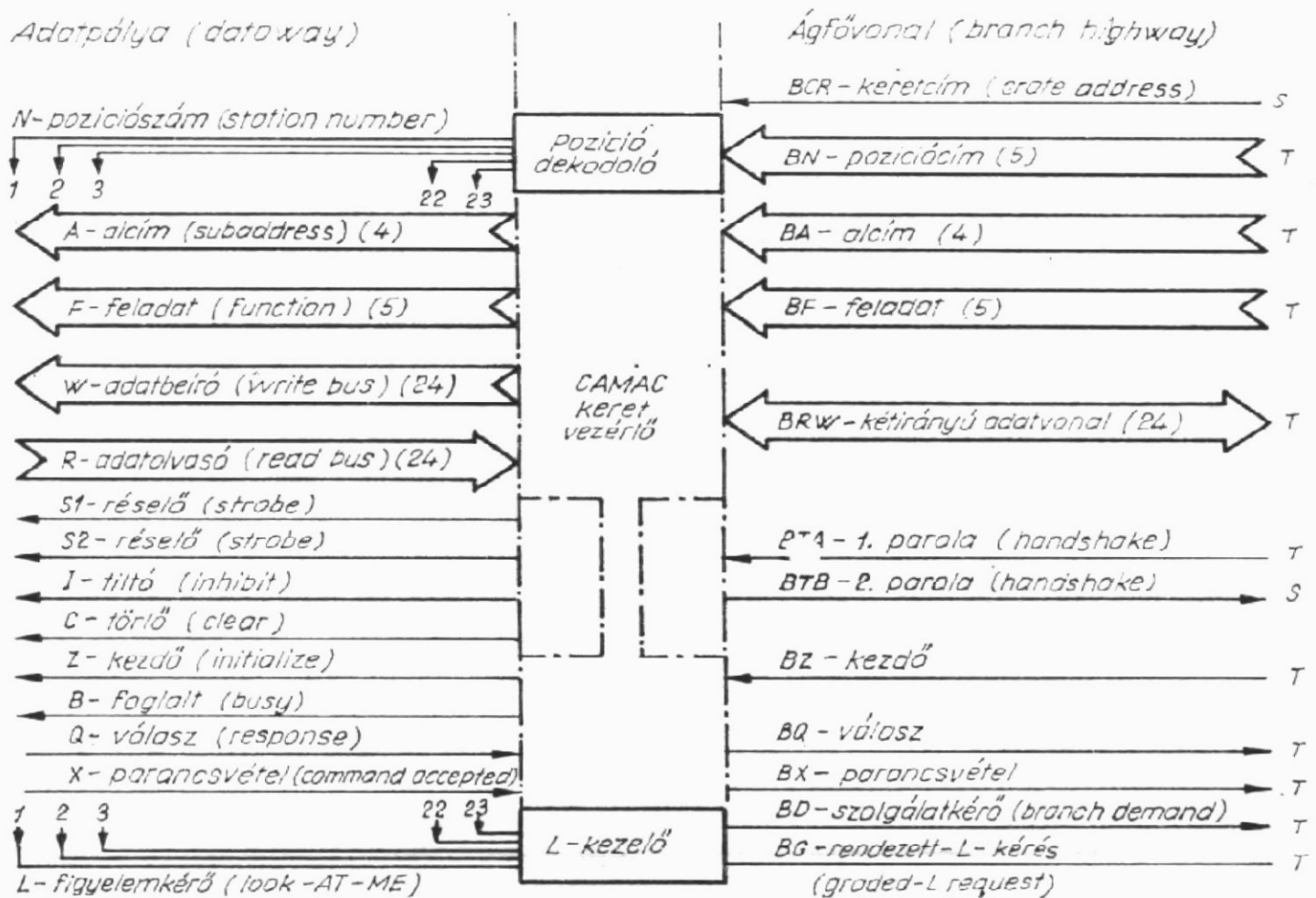
A fővonal paralel változatát a CAMAC rendszerben ágfővonalnak (branch highway) nevezik. Az ágfővonalon kívül a rendszerben (bit- vagy byte-átvitelű) soros fővonal is szerepel.

Bármely keretvezérlő adatpálya felőli oldala (1) szabványos, CAMAC előírásnak megfelelő. A különböző keretvezérlők egyrészt külső csatlakozásuk módjában, másrészt autonómiájuk tekintetében különböznek egymástól. A teljesen önálló keretvezérlőnek nincs külső kapcsolata. Azokat a keretvezérlőket, melyek közvetlenül számítógéphez kapcsolódnak, U-típusú keretvezérlőknek nevezik. Az egyes számítógépek, tekintettel csatornarendszereik sokféleségére, más-más U keretvezérlőt igényelnek. Az ágfővonalra kapcsolódó keretvezérlő mindkét csatlakozása CAMAC rendszerű. A keretvezérlő egy konkrét, ág-csatlakozású realizációja a CAMAC-ban „A” típusjellel szerepel. A soros fővonalhoz soros keretvezérlő tartozik.

A gyakorlatban megvalósított különböző keretvezérlők közül egyesek csupán a rendszervezérlő által kiadott parancsokat hajtják végre, míg mások kiterjedt önálló vezérlési algoritmusokkal, esetleg tárolt programok végrehajtásának lehetőségével is rendelkeznek. A keretvezérlő teljes mikroszámítógépet is magába foglalhat, mely esetben a keret nem más, mint CAMAC csatlakozású kis- (mikro-) számítógép.

Az adatpálya

Az adatpálya két — egyenként 24 bit szélességű — egyirányú adatvonala: a W adatbeíró és az R adatolvasó vonal (4.17. ábra). A beírás és kiolvasás itt az adatpálya által kiszolgált modulok regisztereire vonatkozik. Az adatvonalakon történő átvitel egyik végpontja minden esetben vezérlő; az adatpályára csatlakozó normál modulok közvetlenül nem kommunikálhatnak egymással, csupán vezérlőn keresztül. Az összes adatvezeték kihasználása nem kötelező. Akár a saját vagy a rendszervezérlő,



4.17. ábra.
CAMAC adatpálya és ágfővonal szerkezete

akár a modulok figyelemkérő jeleinek kezdeményezése alapján is, de az adatpályára parancsokat csak a vezérlő bocsáthat ki. A vezérlés nem átruházható.

Az adatpályához általános és címtartalmú parancsok tartoznak. Címtartalmú parancsot a vezérlő a pozíciószámnak megfelelő N vezeték kijelölésével, egyidejűleg az F, ill. A vonalra küldött feladatkóddal, ill. pozícióban belüli alcímmel adhat. Egynél több N vezeték egyidejű aktivizálásával, a parancs egynél több pozícióhoz is eljuttatható. Általános jellegű parancsokat a vezérlő a Z és C individuális vezetékek útján adhat ki. A Z kezdő vezetékkel az adatpálya összes modulja meghatározott kezdő helyzetbe állítható vissza. C azokon a modulokon belül, melyek használják, bizonyos regiszterek tartalmának törlésére szolgálhat. A megkezdett pályaműveletet a vezérlő B foglalt vezetéken át jelzi a moduloknak.

Az I tiltó vonal bizonyos vezérlési feladatokat lát el azoknál az egységeknél, melyekre jele értelmezve van. Az I azonban nem parancs-, hanem vezérlővezeték.

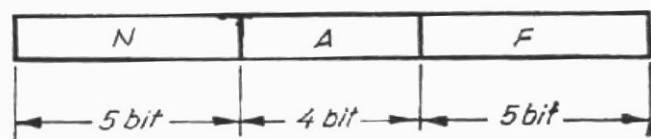
A modulok két állapotjelző vezetékkel, Q és X, állnak HUZAL-VAGY kapcsolatban. Az X-en át a modul a vezérlőnek parancs vételét és végrehajtását jelzi vissza. A Q egyrészt a vezérlő által végrehajtott állapotvizsgálatokra a modul által adott választ szállítja a vezérlőhöz, másrészt blokkátvitel koordinálására szolgál, de egyéb, közelebbről esetenként meghatározandó feladatokat is elláthat.

Az L, pozícióként egy, figyelemkérő vezeték a modulokban zajló autonóm folyamatok meghatározott állapotát jelzi abból a célból, hogy a keretvezérlő vagy a rendszervezérlő előírt szerkezeti vagy/és programrutinjainak végrehajtását kérje.

Az adatpálya szinkron kapcsolatú: a címtartalmú parancsok, valamint Z és C általános parancs végrehajtásával kapcsolatos időzítési-résnyitási feladatokat az S1 és S2 réselő vezeték látja el.

Adatpálya-parancsok

A címtartalmú parancsok formátuma egységes (4.18. ábra). A parancs az 5-bites F feladatkódból, a 4-bites A alcímből és az 5-bites N pozíciócímből tevődik össze. N dekódolása a keretvezérlőben, az A és F dekódolása a modulokban történik. N közvetlenül a pozícióra mutat, A a pozíción belüli funkcionális elemet nevez meg, az F pedig megadja az elem által végrehajtandó feladatot.



4.18. ábra.

Adatpálya-parancsok egységes formátuma:

F — feladat; A — alcím; N — pozíciószám

A feladatkódok jelentését felsoroló 4.2. táblázat alapján a feladatkódok három csoportja különíthető el:

1. standard CAMAC kódok,
2. jövőbeni fejlesztés céljára rezervált kódok,
3. az alkalmazó által tetszőlegesen értelmezhető, nemstandard kódok.

4.2. táblázat.

Sor- szám	Feladat	F				
		16	8	4	2	1
0	Olvas 1. RCS*	0	0	0	0	0
1	Olvas 2. RCS	0	0	0	0	1
2	Olvas töröl 1. RCS	0	0	0	1	0
3	Olvas 1. RCS komplem	0	0	0	1	1
4	Nemstandard	0	0	1	0	0
5	Rezervált	0	0	1	0	1
6	Nemstandard	0	0	1	1	0
7	Rezervált	0	0	1	1	1
8	L-vizsgálat	0	1	0	0	0
9	Törlés 1. RCS	0	1	0	0	1
10	L-törlés	0	1	0	1	0
11	Törlés 2. RCS	0	1	0	1	1
12	Nemstandard	0	1	1	0	0
13	Rezervált	0	1	1	0	1
14	Nemstandard	0	1	1	1	0
15	Rezervált	0	1	1	1	1

Sor- szám	Feladat	F				
		16	8	4	2	1
16	Ráírás 1. RCS	1	0	0	0	0
17	Ráírás 2. RCS	1	0	0	0	1
18	1. RCS szelektív beírás	1	0	0	1	0
19	2. RCS szelektív beírás	1	0	0	1	1
20	Nemstandard	1	0	1	0	0
21	1. RCS szelektív törlés	1	0	1	0	1
22	Nemstandard	1	0	1	1	0
23	2. RCS szelektív törlés	1	0	1	1	1
24	Felfüggesztés	1	1	0	0	0
25	Végrehajtás	1	1	0	0	1
26	Engedélyezés	1	1	0	1	0
27	Állapotvizsgálat	1	1	0	1	1
28	Nemstandard	1	1	1	0	0
29	Rezervált	1	1	1	0	1
30	Nemstandard	1	1	1	1	0
31	Rezervált	1	1	1	1	1

*RCS jelentése: regisztercsoport

A kódkészlet a regiszterek két csoportját különbözteti meg. Az adatregiszterek mindig az 1. regisztercsoporthoz, a status és vezérlés rendeltetésű regiszterek a 2. csoporthoz tartoznak. Valamely regisztert a különböző feladatkódok mellett célszerű egy és ugyanazon alcímmel megnevezni. Bizonyos feladatkódok mellett azonban az alcímek újraértelmezése is lehetséges. Nem követelmény, hogy minden egyes modul az összes feladatkód értelmezésére képes legyen. A konkrét CAMAC modulok használata előtt ezért mindig tisztázni kell, hogy a modulokra vonatkozóan mely standard kódok és mely speciális kódok érvényesek. Ugyancsak nem kötelező a használata, ha egy adott feladatot a modul csak egyféleképpen tud értelmezni. A címtartalmú parancsokat a vezérlőn belül $N(i) A(j) F(k)$ kanonikus alakban adják meg, ahol i, j , és k az N, A , ill. F sorszámval megadott értéke. A modulok adatlapjain ugyanez a cím $NA(j) F(k)$ alakban jelenik meg. $N \equiv N(i)$ csak akkor, ha a modul az i -edik pozíción csatlakozik.

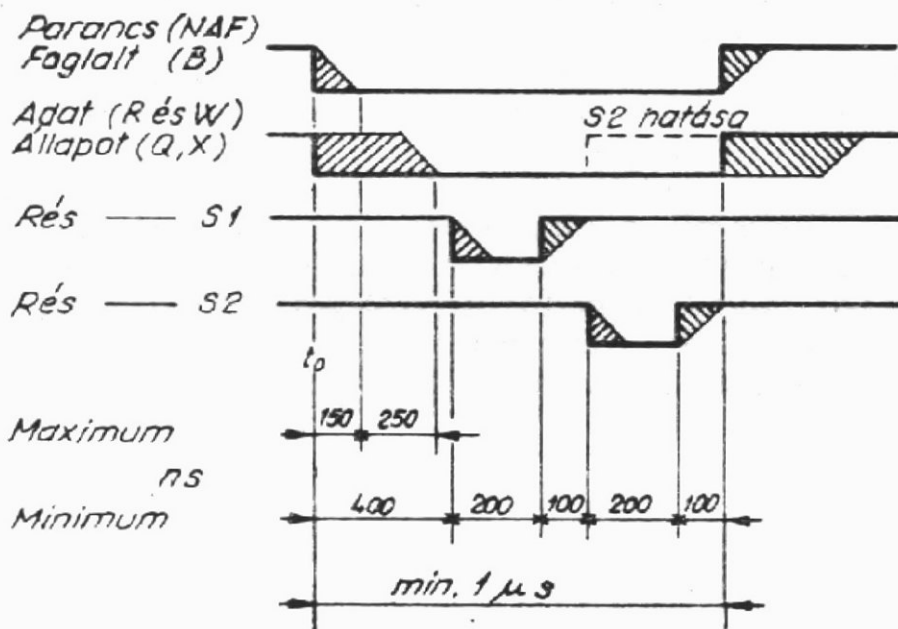
Az egyetlen társasvonalon kiadható Z általános jellegű parancs az adatpályára csatlakozó modulokat passzív definiált kezdőállapotba állítja vissza. Adatot és szolgáltatást éppúgy, mint ahogy vezérlőjellegű regiszterek tartalmát is, törölheti. Z -vel állítható be a rendszer hálózati bekapcsolás utáni kiindulási állapota, s általa végezhető újratevés is.

A C törlő parancs — melynek ugyancsak egy külön társas vezeték felel meg — szelektív feladatokra szolgál, mint pl. blokkátvitel után, meghatározott regisztercsoport tartalmának egyidejű törlésére.

A Z minden modulra kötelező, a C viszont csak azokra, melyekre értelmezve van.

Parancsok végrehajtásának időzítése az adatpályán

A címtartalmú parancsok N, A és F komponense a B jellel egyidejűleg jelenik meg az adatpályán (4.19. ábra), s együttesük az N -edik pozíción csatlakozó modulra közvetlenül hat. Az R és W vonalakat használó, adatátadással együtt járó parancsok az adat moduloldali, ill. vezérlőoldali regiszterének kapuzására az $S1$ résjelet használják. Az $S1$ a t_0 -hoz képest minimum 400 ns eltolással jelenik meg. Ez idő alatt a kifelé irányuló adatnak a modul bemenetén (W), a befelé irányuló adatnak



4.19. ábra. CAMAC adatpálya-parancsok végrehajtásának időbeli lefutása

a vezérlő bemenetén (R) meg kell állapodnia. Az S2 részjel a keretvezérlő beavatkozási lehetőségeit terjeszti ki; segítségével lehet kettős műveleteket végrehajtani. Ilyen kettős művelet pl. az olvasás törléssel [F(2) feladat], melynek végrehajtása során először S1 nyitja az R vonalak keretvezérlő oldali bemenetét, s miután az adat a keretvezérlő adatregiszterébe került, az S2 jel törli a kiolvasott modulregiszter tartalmát.

A parancsok elfogadása és végrehajtása esetén az X állapotjel minden esetben automatikusan adódik. A Q jel különböző feladatokat láthat el. Egyik szerepköre: állapotvizsgáló parancs által feltett kérdésre ad választ. E válasz-szerepkörben időbeli megjelenése és réselése (mintavétele) az R vonal adatáéhoz hasonló módon történik. Bizonyos szerepkörökben Q nem feltétlenül időzített.

Az általános, Z és C parancsok végrehajtása kapcsán az S2 jel kötelező, az S1 megengedett. A Z, ill. C vonal jele csak az S2 részen át fér hozzá az általuk kezelt modulokhoz; ezáltal csökken annak a valószínűsége, hogy a Z és C vezeték zavartranziensei a modulokban nemkívánt állapotváltozásokat idézzenek elő.

A címtartalmú parancsok végrehajtási ideje $\cong 1 \mu\text{s}$, az általános parancsok végrehajtási ideje $\cong 0,75 \mu\text{s}$.

A figyelemkérés (LAM) kezelése

A keret minden egyes normál csatlakozási pozíciójához individuális figyelemkérő (L) vezeték tartozik. A több pozíciót elfoglaló modulok tehát egynél több L jelet is előállíthatnak. Tekintettel az L jelek begyűjtő hálózatának radiális szervezőmódjára, a vezérlő közvetlenül érzékeli, melyik pozíció kér figyelmet. Egy L vezeték azonban több, modulon belüli forrást is képviselhet, s ilyenkor a forrás individuumának azonosítását is el kell végezni. A modulon belül minden forráshoz jelző tartozik, melyet a forrás állít át, s mely megfelelő paranccsal vagy Z jellel állítható vissza. Az egyes forrásokhoz különböző A(j) címeket rendelve, a jelzők az N(i)A(j)F(8) paranccsal sorban lekérdezhetők. Az egymás után feltett kérdésekre a választ a modul a vezérlőnek a Q vezetéken át adja meg. Az azonosításnak egy más módja is alkalmazható. A jelzők összességét regiszterként kezelve, s a regiszternek a 2. regisztercsoportban konkrét alcímet adva, tartalma az N(i)A(j)F(1) paranccsal az R vonalon át olvasható ki. A kiolvasott bináris vektor vizsgálata alapján a szolgálatkérő forrás(ok) identifikálása elvégezhető. (Általában különbséget tesznek LAM állapot és LAM kérés között. E részletekre nem térünk ki.)

A LAM regiszter teljes tartalma törölhető Z jellel. Az egyes jelzőbitek egyenként az F(10), csoportosan az F(23) művelettel törölhető. Adott jelzőbitet automatikusan törölhet a modul által végrehajtott művelet is.

Bármely modulnak, mely L jellel élhet, rendelkeznie kell azokkal az eszközökkel is, melyekkel L tiltható vagy engedélyezhető. A tiltás/engedélyezés az F(24), F(26) műveletekkel vagy a modul részét képező maszkregiszter feltöltésével végezhető.

L jelet minden meg nem címzett modul szabadon, tetszőleges időben állíthat. A megcímzett modul L jelét, ha a modul olyan műveletet végez, mely L-t törölné, a művelet idejére tiltani kell. Ez, legegyszerűbben, az $N(i) \wedge L(i) = 0$ logikai feltétel betartásával biztosítható.

Állapotvizsgálat

Állapotvizsgálat valamely kiválasztott modul megcímzett állapotjelzőjén F(27) művelettel végezhető. A vizsgálat eredményét a modul a Q vonalon át jelenti. Az F(27) művelettel LAM állapotjelzők is megvizsgálhatók. Állapotjelzés és LAM igénylés között éles különbséget kell tenni. Ez utóbbi éppúgy, mint L jel, sem vizsgálható egyedül csak F(8) művelettel.

Blokkátvitel

A CAMAC leírása a blokkátvitel

1. címletapogatóssal,
2. ismétléssel és
3. zárójelzéssel

végzett módját különbözteti meg.

Címletapogatóssal végzett blokkátvitelben a modulok oldalán bármely regisztercsoport részt vehet. A csoport minden egyes regisztere csak egy adat átvitele kapcsán szerepel. A regiszterek a modulok között megoszolhatnak. Az érintett modulok mindegyike 1...16 regisztert tartalmazhat; üres pozíciók is előfordulhatnak. A rendszervezérlőnek elegendő az első és utolsó regiszter helyét és az adatszámot ismernie.

A címletapogatósi módszer olyan modulokra alkalmazható, melyek regiszterei az A(0)-tól sorban következő alcímekkel érhetők el, s melyek Q=1 jellel válaszolnak minden olvasás vagy írásparancsra, ha az meglévő regisztert címez meg, de Q=0 jelet adnak, ha a megadott címen regiszter nem található. Amennyiben valamelyik modul <16 regisztert tartalmazna, akkor a címmutató Q=0 észlelésekor az üres helyeket automatikusan átlépi, s a következő pozíció A(0) címére ugrik. Lehet azonban, hogy a soron következő pozíció üres; ez esetben a címmutató ennek teljes tartományát átugorva lép tovább.

Az *ismétléssel* végzett blokkátvitelben, moduloldalán egyetlen regiszter vesz részt. Q=1 jelzi, ha az átvitel sikeres. Ez esetben Q a modulregiszter foglaltságjelzőjének szerepkörét játssza.

A rendszervezérlő számára ismeretlen hosszúságú blokkok esetében használják a *zárójelzéssel* végzett átvitelt. A blokkátvitel végét a modul jelzi, a Q vonal 1→0 logikai szintváltásával.

Az ágfővonal

Az ágfővonal aszinkron kapcsolható csomópont, melyre max. 7 keretvezérlő csatlakozhat. A sín szerkezetét (az adatpályáéval együtt) a 4.17. ábra szemlélteti. T társasvezeték, S sugárvezeték jelöl. A lehetséges hét keret mindegyikéhez egy-egy külön BCR keretcím vezeték tartozik. Az N pozíciószám az ágfővonalon binárisan kódolt alakban 5 vezetéken (BN), jelenik meg. BA, BF vonalak, s a BZ, BQ, BX vezeték az adatpálya A, F vonalainak, ill. Z, Q, X vezetékének felelnek meg. Az adatpálya R és W vonalait az ágfővonalon kétirányú BRW adatvonal képviseli. A BD szolgálatkérő vonal a fővonalon elhelyezkedő keretek összes L jelét VAGY kapcsolatba fogja össze. A BG vonal a forrásazonosításban játszik szerepet. A BTA az ágvezérlő (branch driver) által kibocsátott parolajelet szállítja a kerethez. Minden keretnek saját BTB válaszvonala van (az összes BTB vonalak száma tehát 7). A megcímzett keret a BTA jelre BTB vonalán át válaszol. Az I, C, B, S1, S2 adatpálya vezetéknek az ágfővonalon nincs megfelelője. Az utóbbi hármat kizárólag a keretvezérlő generálja. Az I és C vonalak működtetése viszont fővonalon parancsok útján történik.

Ágparancsok

A fővonalai parancsok tartalmilag az adatpálya parancsaitól a C keretcímmel térnek el. Alakjuk tehát: CNAF. A 7 keretnek megfelelően C bináris kódja 3-bites, ily módon a teljes fővonalai parancs (ágvezérlőn belüli) mérete 17 bit. C az ágvezérlőben, NAF a keretben dekódolódik. Tekintettel arra, hogy az N ötszámjegyű bináris szám 32 kombinációt determinál, ugyanakkor a kereten belül csak 23 pozíció van, a fennmaradó kombinációk rendelkezésre állnak, s ezekkel a fővonalai parancsok különleges csoportja képezhető (4. 3. táblázat), melyek interpretációja nem a modul(ok)ban, hanem a keretvezérlőben történik. A $C(a)N(26)A(j)F(k)$ parancs az $F(k)$ műveletet az összes normál pozíció (1...23) $A(j)$ alcímén érvényesíti. A keretvezérlő részét képező pozíciószám-regiszter az $N(30)$ pseudo-pozíciószámot tartalmazó parancscsal tölthető fel; e regiszter tartalmával viszont tetszőlegesen, kereten belüli pozíciók preszelektálhatók. A $C(a)N(24)A(j)F(k)$ alakú parancs az összes preszelektált pozíción az $F(k)$ műveletet egyidejűleg hajtja végre.

4.3. táblázat

Az ágfővonal pozíciókódjainak értelmezése

N	Értelme	B, S1, S2	Megjegyzés
N(0)	Rezervált		
N(1) – N(23)	Normál pozíciócímek	igen	A keretvezérlő által lefoglalt normál pozíciókat nem kell címezni
N(24)	Preszelektált normál pozíciók	igen	
N(26)	Az összes normál pozíció	igen	
N(28)	Egyedül a keretvezérlőnek szól	igen	
N(30)	Egyedül a keretvezérlőnek szól	nem	Nincs adatpálya-művelet
N(25, 27) N(29, 31)	Rezervált		

Az általános jellegű törlőparancsnak és tiltó jelnek, melyeket az adatpályán egy-egy (C, ill. I) társasvezeték továbbít, a fővonalon címtartalmú parancsok felelnek meg. A törlő parancs $C(a)N(28)A(9)F(26)$. Az I vonal tiltó/engedélyező állapota (egy bistabilt érintő) fővonalai parancscsal vezérelhető, éspedig az átállítás $C(a)N(30)A(9)F(26)$, a visszaállítás $C(a)N(30)A(9)F(24)$ útján. Az $A(9)$ alcím, az A típusú keretvezérlő tervezésénél megválasztott belső cím. A Z általános parancs átvitelére a fővonalon is egyetlen vezeték, BZ szolgál, ugyanakkor $C(a)N(28)A(8)F(26)$ parancscsal is generálható.

A BX vezeték a fővonalra futó összes X vezeték VAGY kapcsolatba fogja össze. A parancselfogadás jele így az ágvezérlőnek közvetlenül rendelkezésére áll.

Ágfővonal szolgálatkérés

Az ágfővonalhoz tartozó keretek bármelyikében jelentkeznek figyelemkérés, arról az ágvezérlő a BD szolgálatkérő (társas) vezeték újtán értesül. Az ágvezérlő a BD jelre válaszképpen a keretekhez BG, rendezett L-kérés jelet küld ki. Eredményül a BRW vonalon át az ágvezérlőhöz egy az összes keret L-állapotát tükröző, rendezett állapotvektor érkezik. Minden keretvezérlőhöz tartozik egy ún. LAM-rendező, mely az adott keret pozícióazonosítóit megfelelően kódolva a BRW vezetékhez rendeli. Az állapotvektort esetenként kell értelmezni. Legegyszerűbb alakjában az állapotvektor a BRW vonalnak csak hét vezetékét foglalja le; a hét bit mindegyike egy-egy keretet identifikál. Ha a rendszerhez pl. csak egyetlen keret tartozik, az állapotvektor 23 bitjének mindegyike e keret egy-egy pozícióját képviselheti. A pozíciószám az állapotvektorban nemcsak explicit, hanem kódolt alakban is helyet foglalhat. Amikor egy pozícióhoz több szolgálatkérő forrás, a rendszerhez pedig egynél több keret is tartozik, a forrásazonosításhoz szükséges minden adat általában nem fér el az állapotvektorban. Ilyenkor az állapotvektor beérkezését követően az ágvezérlőnek (rendszervezérlőnek) még bizonyos kiegészítő, lekérdeztetési eljárást kell lefolytatnia, mely az adatszámításán végzett azonosítási eljárással analóg, s melynek egyik módozatában a BQ vezeték a Q vezetékek jeleit továbbítja. A BQ vezeték az egyes keretek Q vezetékét VAGY kapcsolatba fogja össze.

Blokkátvitel az ágon

Az adatszámításán lehetséges három különböző típusú blokkátvitel mindegyikét az jellemzi, hogy folyamatukban az adatok átvitele a keretvezérlő által kibocsátott parancsok alapján történik. Ágrendszer esetében a parancsok a keretvezérlőhöz az ágvezérlőből érkeznek, a parancsok generálása menetében pedig BQ a Q-val kongruens. Ágrendszerben a címletapogató tárgyát képező regiszteregység egynél több keret között is megoszolhat.

Soros fővonal

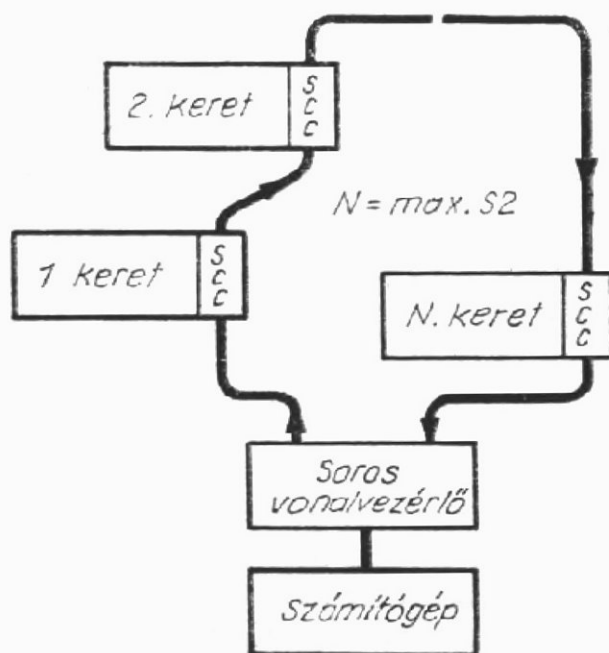
A CAMAC előírás a keretvezérlők és a számítógép összeköttetésére szolgáló soros fővonalat (SH — serial highway) is definiál, mely akár bitsoros, akár byte-soros üzemmódban használható, s amelyre maximum 62 keret kapcsolható. A soros fővonalhoz (az alkalmazó választása szerint) bitsoros, vagy byte-soros üzemmódban dolgozó, előírás szerinti, soros keretvezérlő járul.

Az ágfővonalal helyettesítése soros fővonalal a következő esetekben indokolt:

- amikor a keretet (vagy kereteket) a számítógéptől jelentős távolság választja el,
- zajos környezetben, ahol az átviteli hibák elleni hibavédelem problémája előtérben áll,
- amikor a gép és a keretek közötti információforgalom csekély, akár mert a mérőhálózat adatárama kicsiny, akár mert intelligens keretvezérlők adatredukciót végeznek.

A soros fővonalat a számítógéphez soros vonalvezérlő illeszti, de ahol a követelmények mérsékeltek, a legtöbb számítógéphez hozzátartozó távíró-bejárathoz is csatlakoztatható, egyszerű adapterrel.

A soros fővonal zárt hurkot képez (4.20. ábra), melyen az üzenetek egyirányban haladnak. Minden keretvezérlőnek egy bejárata és egy kijárata van. A keretvezérlő csak a neki szóló üzeneteket hallja meg, s a neki szóló üzenet kivételével minden, a bejárathoz érkező üzenetet a kijáratán reprodukál.



4.20. ábra.
CAMAC rendszer soros fővonallal

Az üzenetek három típusa: a soros vonalvezérlő által kibocsátott parancsüzenet, a keretvezérlő által a parancsra adott válaszüzenet és a szolgálatkérő üzenet, mely a keret figyelemkérő jelei által definiált információt hordoz. Az üzenetek mindegyike byte-strukturált; a byte paralel vagy soros módon vihető át. A CAMAC előírás rögzíti az üzenetformátumokat, az üzenetátadás procedúráit és a soros keretvezérlő bejárati és kijárat jeleinek specifikációját.

A keretek egy csoportja a soros fővonallal közös vevő-adó adapterrel csatlakozhat, míg a csoporton belüli kommunikáció az előírásoknak megfelelő módon megy végbe.

A hurokhálózatra való tekintettel bármelyik keretvezérlő hibás működése vagy leválasztása az egész rendszert megbéníthatná. Ennek elkerülésére különböző standard eljárások szolgálnak.

A rendszer sebességfüggetlen logikával rendelkezik; 100 bit/sec...5 Mbyte/s sebességhatárok között megkötés nélkül használható.

A fővonalak megvalósításának módjai és lehetőségei

A fővonalak nem csupán szervezésmódjukban, hanem az alkalmazott vezetékek milyensége tekintetében is különbözhetnek egymástól:

Az U vezérlők mindig az alkalmazott számítógép sínrendszerére kapcsolódnak a gépre előírt konvenciók szerint.

Az ágfővonal két jellegzetes változata az aszimmetrikus (kiegyenlítetlen) és a szimmetrikus (kiegyenlített). A szabványos, az A keretvezérlővel kapcsolatban definiált vonal aszimmetrikus. Az aszimmetrikus vonal maximális megengedhető hossza a vezeték keresztmetszettől függően 16...66 m. A kábelhosszat a jelszint esése, a logikai szintek feszültségtartománya és a megengedhető (vagy inkább feltételezett) zajamplitúdó korlátozza.

Szimmetrikus vezetékpárokból álló vonallal, szimmetrikus átvitel módban igen kedvező jel/zaj viszony érhető el, minek folytán 2 000 m vagy annál nagyobb kábelhossz is megengedhető. A szimmetrikus vonal a keretvezérlőkben és az ágvezérlőkben is szimmetrikus vevő- és hajtóáramköröket igényel. Annak érdekében, hogy a ki-

egyenlítően vonal fogadására kialakított A keretvezérlőt, valamint ágvezérlőt alkalmazni lehessen, de ugyanakkor a szimmetrikus vonal előnyét is kihasználhassák, a vonal és a vezérlők közé aszimmetrikus/szimmetrikus átalakítókat iktatnak.

Soros vonalként a megkövetelt átviteli sebességtől függően igen sokféle vezeték-fajta alkalmazható. Figyelemreméltók a galvanikusan nem vezető vonalak. Ezek sorában megemlíthetők a közvetlen fényutak és optikai szálvezetők fénykapcsolóikkal, a lézersugár, a rádiókapcsolat, a transzformátorcsatolás. E nem vezető vonalak alkalmazása azokban a mérőrendszerekben válhat szükségessé, melyekben a mérőhálózat egyes területeit különböző okokból egymástól el kell szigetelni.

CAMAC—számítógép kapcsolat

A legkisebb kiterjedésű CAMAC rendszernek egyáltalán nincs számítógép-kapcsolata, egyetlen keretből áll. Ez esetben a keret által összefogott mérőrendszer algoritmusait maga a keretvezérlő bonyolítja le. Az algoritmus és a vezérlő is lehet igen egyszerű. Példaképpen az adatnaplózó rendszer említhető: A vezérlő olvasás — beírás parancspárokat generál, az előbbi valamelyik modul regiszterének tartalmát átírja a vezérlő adatregiszterébe, majd az utóbbi az így nyert adatot egy másik — írógép, szalaglyukasztó, mágneses szalagtár stb. készülékvezérlőjeként szolgáló — modulhoz továbbítja. Címletapogatással végzett blokkátvitel üzemmódban a vezérlő meghatározott regiszteregyüttest járhat végig, a regiszterek tartalmát rendre közbenső hordozóra víve át.

A keretvezérlő fixtárral vagy átírható tartalmú fixtárral kiegészítve képessé tehető bonyolultabb strukturájú algoritmusok lebonyolítására, melyekbe szubrutinok, feltételes elágazások stb. is beleépíthetők. Szélső esetben a keretvezérlő képességei mikroszámítógép színvonaláig fejleszthetők.

Az önálló keret legegyszerűbb módon adat-végállomás módján kapcsolódhat számítógépre. Az összeköttetés az írógéphez szolgáló soros aszinkron csatlakozást nyújtó készülékvezérlő útján valósítható meg.

A következő csatlakozási szintet az U keretvezérlő, majd az ágfővonalon át történő csatlakozás jelenti. A kettő működési szempontból egyenértékű. Azt, hogy adott esetben melyik lehetőséggel érdemes élni, részben a rendszer kiépítettségével összefüggő költségtényezők, részben az üzemi követelmények döntenek el. Kisebb CAMAC konfiguráció esetén U vezérlő, nagyobb konfiguráció esetén ágfővonal, A-típusú keretvezérlők és ágvezérlő alkalmazása látszik gazdaságosabbnak. Az üzemi adottságokat tekintve az ágfővonal — több keretre kiterjedő blokkátvitel kapcsán — a számítógép szemszögéből nézve transzparens és ezért használata célszerű. Viszont több modullal szimultán végzett blokkátvitelhez az U-típusú csatlakozás látszik előnyösebbnek. Az U-típusú számítógép-csatlakozás használatakor a figyelemkéréssel kapcsolatos forrásazonosítás is egyszerűbb.

A CAMAC adatstruktúrája a számítógépeketől, nyelve a gépek csatornáinak nyelvétől eltér. Az eltéréseket a 4. 4. táblázat foglalja össze. Az időzítési problémákat általában közbenső tárolással oldják meg. Ez különösen fontos, ha a CAMAC-ot szinkron kapcsolatú gép mellett alkalmazzák, melynek saját, a CAMAC-étól eltérő működési ritmusa van. A közbenső regiszter mindkét oldala saját (CAMAC, ill. számítógép) ritmusában kommunikálhat, de egyidejűleg csak egyik oldallal állhat kapcsolatban. Aszinkron kapcsolat esetén a CAMAC ciklus beleágyazható a csatorna munkaciklusába.

A CAMAC — számítógép csatlakozóegység szerkezeti felépítése nem csak a számítógép típusával változik, hanem függvénye a CAMAC — számítógép közötti kommunikáció módjának (programkapcsolatú, programfüggetlen) és a keretvezérlő, ill. ágvezérlő intelligenciája mértékének is.

Összehasonlítás tárgya	CAMAC jellemzők	A jellegzetes kisgép csatlakozási jellemzői
Műveletek	Kétirányú adatátvitel A gépétől eltérő, modulvonatkozási parancsok Figyelemkérés-szolgáltatás	Kétirányú programkapcsolatú, ill. programfüggetlen adatátvitel Csatornaműködtető utasítások; készülékvonatkozási parancsok Megszakítás; programfüggetlen csatorna szolgáltatás
Alapciklus ideje	$\cong 1 \mu s$	Gépenként változik; tipikusan: $1 \mu s$.
Műveletek időzítése	Saját előírás szerint	Gépenként különböző, CAMAC-tól eltérő
Adatméret	1...24 bit között választható	8, 12, 16 bit; ritkábban 18, 24 bit
Szolgáltatás	A keretvezérlőnek 23 figyelemkérő szint, az ágvezérlőnek 1 szolgáltatás szint áll rendelkezésre; GL jel hatására utóbbi 24-bites szolgáltatás állapotvektort kap: további információ lekérdezéssel nyerhető	Megszakításkérés: egy vagy több független szint Programfüggetlen csatorna szolgáltatásai

Irodalom a 4. fejezethez

- [1] Rinder, R.: The input/output architecture of minicomputers, *Datamation*, May 1970, 119...124. old.
- [2] DiMauro, J.: Input-Output the basic options, *EE systems engineering today*, May 1973, 98...99. old.
- [3] Kocher, Ch. P.: Interrupt driven I/O, *EE systems engineering today*, May 1973, 100...101. old.
- [4] Knoblock, D. E.: I/O channels for minis: fast and simple, *EE systems engineering today*, May 1973, 102...103. old.
- [5] Barber, D. L. A.: Experience with the use of the British Standard Interface in Computer Peripherals and Communication Systems, *Proc. ACM Symposium on Problems in the Optimization of Data Communications Systems*, Pine Mountain, Georgia Oct. 13—16, 1969, 271...289. old.
- [6] Hampel A.—Zimányi I.: Korszerű műszerfelépítés — Korszerű interface. *Mérés és Automatika XXI. éf. 8. sz. 1973*, 319...323. old.
- [7] Zimányi I.: Mérésautomatizálási célra alkalmas műszerek interface kialakításai a SIAK és BSI ajánlások tükrében, *Mérés és Automatika, XX. éf. 8. sz. 1972*, 312...316. old.
- [8] Naumann, G.: Standard-Interface, eine Voraussetzung zum Aufbau automatisierter elektronischer Systeme, *mstr 15 ap. 1972*, H. 12. 270...272; *mstr 16 apr. 1973*, H. 2. 41...44. old.
- [9] Kurtze, P. et al.: Stand und Entwicklungstendenzen der Interfaces der elektronischen Messtechnik, *Radio Fernsehen Elektronik 21. 1972*, H. 3. 73...77. old.
- [10] Williams, T. J.: Interface problems for process control, *IFAC/IFIP Internat. Conf. on Digital Applications to Process Contr. P. III. 1974*, Zürich, 39...79. old.
- [11] Vachon, B.—Weiske, W. Jr.: Interfacing: A balancing act of hardware and software, *Electronic Design 11*, May 27. 1971, 58...63. old.

- [12] *Hoffman, A. A. J.—French, R. L. — Lang, G. M.*: Minicomputer interfaces: know more, save more, IEEE spectrum, Febr. 1974, 64...68. old.
- [13] The Proceedings Minicomputer Interfacing, A One-day symposium at the Polytechnic of Central London loth Nov. 1973, MINICONSULT 35 Richmond Ave, Islington London N1.
- [14] *Souček, B.*: Minicomputers in data processing and simulation (5. Interfacing Fundamentals), Wiley-Interscience 1972.

A HP — csatlakozásra vonatkozó irodalom:

- [15] *Grossman, S. E.*: Instrument makers seek a language, Electronics, Sept. 19. 1974, 67...68. old.
- [16] *Nelson, G. E.—Ricci, D. W.*: A practical interface system for electronic instruments, Hewlett—Packard Journal, October 1972, 2...7. old.
- [17] *Loughry, D. C.*: A common digital interface for programmable instruments: the evolution of systems, Hewlett—Packard Journal, October 1972, 8...11. old.
- [18] *Loughry, D. C.*: The Hewlett—Packard interface bus: current perspectives, Hewlett-Packard Journal, January 1975, 2...4. old.
- [19] *Ricci, D. W.—Stone, P. S.*: Putting together instrumentation systems at minimum cost, Hewlett—Packard Journal, January 1975, 5...11. old.
- [20] Interface system for programmable measuring apparatus, byte-serial bit-parallel; International Electronic Commission, Technical Committee No 66: Electronic Measuring Equipment, June 1975.

A CAMAC csatlakozásra vonatkozó irodalom:

- [21] CAMAC A Modular Instrumentation System for Data Handling EUR 4100e, 1972; AEC Report TID—25877, July 1972.
- [22] CAMAC Organization of Multi-Crate System, EUR 5600e, 1972; AEC Report TID—25876, March 1972.
- [23] Supplementary Information on CAMAC Instrumentation System; AEC Report TID—25877, Dec. 1972.
- [24] CAMAC Serial System Organization ,ESONE/SH/01; AEC Report TID—26488, Dec. 1973.
- [25] CAMAC Specification of Amplitude Analogue Signals within a 50 Ω System, EUR 5100, 1974; AEC Report TID—26614, Oct. 1974.
- [26] CAMAC The Definition of IML a Language for Use in CAMAC systems, ESONE/IML/01; AEC Report TID—26615 Jan. 1975.
- [27] Proc. of the 1972 Nuclear Science Symposium IEEE Transactions on Nuclear Science, Vol. NS—20, No. 1. Febr. 1973.
- [28] Proc. of the 1973 Nuclear Science Symposium, IEEE Transactions on Nuclear Science, Vol. NS—21, No. 1. Febr. 1974.
- [29] Proc. of the first International Symposium on CAMAC, Luxemburg, Dec. 4—6. 1973.
- [30] CAMAC Bulletin, az ESONE bizottság időszaki közleménye.
- [31] *Bisby, H.*: The CAMAC Interface and some Applications, The Radio and Electronic Engineer, Vol. 41, No. 12, Dec. 1971. 527...537. old.
- [32] *Hetthésy J.*: A CAMAC rendszer általános rendszertechnikai felépítése. Mérés és Automatika, XX. évf. 5. sz. 1973, 167...171. old.
- [33] *Biri J.—Blasovszky M.—Lukács J.—Somlai L.—Vashegyi Gy.*: KFKI—CAMAC rendszerek nem-nukleáris alkalmazásokban. Mérés és Automatika, XXII. évf. 7. sz. 1974, 257...260. old.
- [34] *Arató P.—Kalmár P.—Kondorosi K.—Kőrösi J.—Lantos B.*: Folyamatillesztő rendszer multiplex és programozott számítógép-csatornákra. Mérés és Automatika, XX. évf. 8. sz. 1972, 307...311. old.
- [35] *Tradowsky-Thal, I.*: CAMAC—Bibliographie, Kernforschungszentrum Karlsruhe. KFK 1471, 1971.
- [36] *Biri J.—Lukács J.*: CAMAC Perifériarendszer, Műszaki Könyvkiadó 1976.
- [37] *Виноградов, В. И.*: Современное развитие программно-управляемых модульных структур для автоматизации измерений и управления экспериментами, Автометрия №. 4. 1972, 3...11. old.
- [38] *Бобко, В. Д. и др.*: Магистральная система обмена информацией, Автометрия №. 1974, 9...19. old.
- [39] *Виноградов, В. И. и др.*: Особенности организации магистрального канала программно-управляемой модульной структуры, Автометрия, №. 4, 1974, 31...39. old.
- [40] *Бредихин, С. В.—Песляк, П. М.*; Средства программирования для САМАС, Автометрия, №. 4. 1974, 39...50. old.

5.

A mérőhálózat gépi irányítása

5.1. A programirányíthatóság problematikája

Analóg strukturájú mérőkészülékek jelvezérlése

A manuális beállítású (adó/vevő típusú) mérőkészülékek beavatkozó szervei részben analóg, részben digitális jellegűek. Az analóg szerv legegyszerűbb esetben folyamatosan változtatható értékű kapacitás, induktivitás, ellenállás vagy potenciométer; a digitális jellegű szerv két-, ill. többállású kapcsoló, mely két pont között ellenállást, kapacitást vagy induktivitást válthat előírt fokozatokban; ellenállás-, kapacitás-, induktivitáslánc vagy transzformátortekercs megcsapolási pontjait választja ki, áramköri útvonalakat szakít meg, köt össze, módosít, rendez át a készüléken belül. A koncentrált paraméterű analóg beavatkozó szervek mellett esetenként osztott paraméterű beavatkozók (pl. művonal) is előfordulnak, a digitális jellegű beavatkozók sorában pedig különböző dugaszokkal és dugaszmezőkkel is találkozunk. A beavatkozókhoz a kezelhetőséget lehetővé tevő szervek (kapcsoló- és forgatógombok, diálok stb.) tartoznak, melyek továbbá a hozzájuk rendelt skálákkal, esetleg az általuk beállított értéket mutató mérőműszerekkel egészülhetnek ki.

Funkciójukat tekintve a digitális jellegű beavatkozók tartomány és üzemmód állítására, az analóg jellegű beavatkozók a mérőkészülékek valamely paraméterének finombeállítására, nullpont-, ill. végkitérés-korrekcióna vagy kiegyenlítésre (pl. híd) szolgálnak.

Annak érdekében, hogy a mérőkészüléket számítógép-irányítású rendszerben lehessen alkalmazni, szükséges, hogy jelek által vezérelhető legyen. A jelvezérelhetőség kézenfekvő lehetősége: a beavatkozók kezelőszerveihez elektromechanikus eszközök (pl. léptetőmotor) adaptálása. Ez a megoldásmód azonban egyáltalán nem elégíti ki az egyre szigorúbb rendszerkövetelményeket, s bár igen szűk körben jelenleg is szerepel, az elektronikus módszerek mellett azonban háttérbe szorul.

A kapcsolók közvetlenül helyettesíthetők digitálisan vezérelhető reed-reléekkel vagy félvezető elemekkel, ill. ezek nagyobb együtteseivel, az analóg jellegű beavatkozók pedig a digitális értéket analóg rendszerjelre leképező, D—A átalakítás technikáján alapuló digitális ellenállásokkal, potenciométerekkel, kondenzátorokkal, feszültség- és áramátalakítókkal stb.

Az első generációs mérőkészülékek általánosságban analóg jellegűek. Analóg jellegről szólva itt elsősorban nem a készülékek adatszolgáltatásának formájára gondolunk, hanem általánosságban a mérőkészülékek áramköri struktúrájára. Bár e készülékek manuálisan kezelt beavatkozói nehézség nélkül helyettesíthetők jelvezérelt beavatkozókval, maga az analóg áramköri struktúra ritkán illeszkedik a digitális gépi irányítás követelményeihez, és a számítógép-irányítású mérőrendszer pontossági követelményeinek kielégítését sem teszi lehetővé. A legnagyobb problémát az okozza, hogy az analóg strukturájú készülékben alkalmazott jelvezérlésű

beavatkozó digitális bemenő értéke nem közvetlenül a beavatkozó által befolyásolt mennyiség értékére, hanem valamely analóg rendszerjel (ellenállás, kapacitás stb.) értékére képződik le. Ilyen helyzet állna elő akkor, ha valamely manuális kezelőszerv skáláján pl. nem impulzusszélesség, hanem ellenállásérték szerepelne, ami aztán szükségessé tenné, hogy az impulzusszélesség és az ellenállás összetartozó értékeit külön táblázat adja meg. A manuális készülék előregyártott skálája fixtár, egyedi úton hitelesített skálája változtatható tartalmú tár alkalmazásával (fenti említett táblázatnak e tárukban való elhelyezésével) jelvezérelt beavatkozó szerv esetében is imitálható, mindez azonban költséges és rendkívül nehézkes is. A leggondosabb eljárás sem véd a hitelesítésben a környezeti hatások és készüléköregedés okozta hibáktól. E hibák kézbentartásával és korrekciójával járó tevékenység pedig a gépet jelentékenyen terheli.

Az analóg strukturájú mérőkészülékek sorában a gépi vezérelhetőség szempontjából kivételes helyzetet foglalnak el azok, melyeknél a beavatkozó analóg rendszerjele és e rendszerjel által befolyásolt mennyiség értéke között lineáris összefüggés áll fenn. Ez esetben ugyanis a rendszer lineáris D—A konverterre redukálható, melynél a kimenő jel értéke és a beavatkozó digitális bemenő értéke közötti kapcsolatot konstans skálatényező definiálja. Ez a helyzet pl. a programozható tápforrások esetében.

Közvetlen digitális vezérlésű készülékek

Részben a digitális technika folyamatosan születő eredményei, részben a fokozódó, ill. egyre újabb követelmények s különösen a számítógép-bázisú mérés-technika ösztönző hatására is, az analóg mellett kialakultak a digitális strukturájú készülékek is. Digitális struktúrán elsősorban ismét nem az adatszolgáltatás formájára gondolunk, hanem a készülék áramköri, működési mechanizmusára és felépítésmódjára. Igaz, hogy az adatközlés formájának megváltozásával a legtöbb mérőkészülék ma már legalábbis vegyes, analóg és digitális felépítésű, számos azonban a tisztán digitális szerkezetű készülék is. Ez utóbbiak sorából elég megemlíteni a digitális frekvencia-, idő- és impulzusszám mérőket, a digitális jelszintetizáló generátorok különböző változatait, a digitális impulzusgenerátorokat és szógenerátorokat, melyek egy része korábban analóg rendszerű mérőkészülékek feladatkörét vette át, más részük merőben új rendeltetésű.

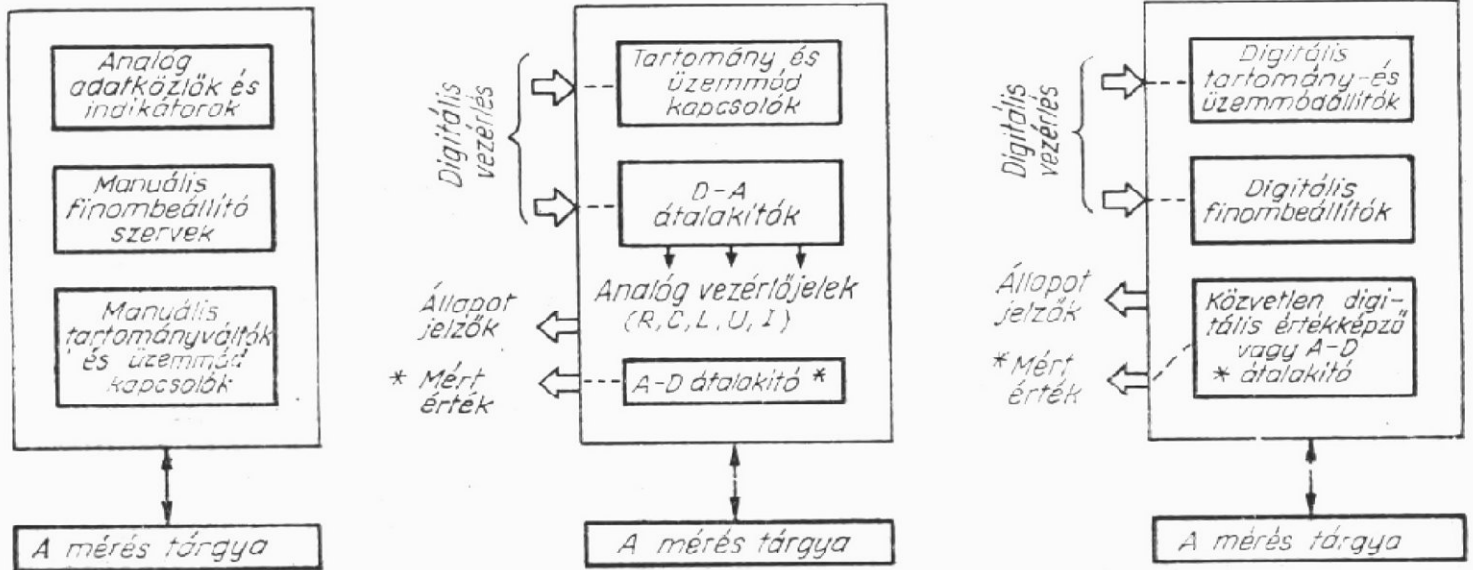
A digitális strukturájú mérőkészülékek működési paramétereinek értékállítása közvetlenül, digitális jelek útján történik, a digitális vezérlőjel és az általa befolyásolt paraméterek értéke között közvetlen kapcsolat áll fenn, sőt a paraméter értékét maga a szám definiálja, az semmi mástól sem hitelesítési, sem környezeti, sem öregedési, sem más tényezőktől nem függ. Érzékeltetésül a preszetelhető, vagyis számlálási kapacitása szempontjából programozható számlálót mint frekvenciaosztót említjük meg, mely a bemenő jel frekvenciáját az előre beállított digitális értéknek megfelelően osztja le.

Az 5.1. ábra a mérőkészülék három változatát szemlélteti. Végül megjegyezzük, hogy a gépi úton vezérelhető mérőkészülékek általában rendelkeznek manuális kezelőszervekkel is.

Manuális készülék

Analog strukturájú jelvezérelt készülék

Közvetlen digitális vezérelt készülék



* Csak vevő típusú mérőkészülékeknek

5.1. ábra.

Mérőkészülék manuális és jelvezérelt változatai

Jelvezérelhetőség, programirányíthatóság, rendszerkompatibilitás

A mérőkészülék programirányíthatóságával, a „programozott”, „programozható” kifejezésekkel kapcsolatban a szakirodalomban általában bizonyos terminológiai és fogalmazásbeli bizonytalanság tapasztalható. Az általánosítás skáláját szétterítve egyes szerzők minden mérőkészüléket, mely egyáltalán rendelkezik beavatkozó szervevel, programozhatónak neveznek, így a hagyományos készülékeket „kézi programozású-”nak. Ez utóbbi kifejezés azonban sokkal inkább azoknak a készülékeknek megnevezésére alkalmas, melyek bizonyos kapcsoló- vagy dugaszmezőkkel manuálisan programozható, egyszerűbb vezérlőelektronikát is magukba foglalnak.

Még gyakoribb a *jelvezérelhetőség* (elektromos jelek útján való távvezérlés) és a programirányíthatóság fogalmának azonosítása. A jelvezérelhetőség a programirányíthatóságnak elengedhetetlen, de nem egyedüli feltétele. A programirányíthatóság a számítógép és a készülék közötti kommunikáció magasabb szintű követelményrendszerének kielégítését is megköveteli.

A következőkben a jelvezérelhető készüléket *programozhatónak*, a számítógéphez (vagy más vezérlőszerkezethez) közvetlenül csatlakoztatható és azzal kommunikálni képes készüléket pedig *programirányíthatónak* nevezünk.

A számítógép a külvilággal csupán digitális (kétszintű) jelek útján érintkezhet. A programozható készüléket (teljes, analóg és digitális szerkezetű skálájukra tekintve) vezérlőjelként digitális és analóg rendszerjelet egyaránt igényelhetnek, ill. szolgáltathatnak, s általános esetben még digitális jelek szintje, formátuma (szóhosszúsága) és kódolása is különbözik a számítógépétől, amiért is ahhoz közvetlenül nem csatlakoztathatók. Más szóval a készülékek a számítógéppel nem kompatibilisek; nyelveik különböznek. A kompatibilitás, az egymással való kommunikáció lehetősége megteremtésére szolgál a számítógép és a mérőkészülék közé elhelyezett készülék-vezérlő.

Az egyedi eseten túltekintve felmerül a készülékek általános rendszerkompatibilitásának problémája; az az igény, hogy bármely készülék bármely számítógépet alkalmazó mérőrendszerhez, külön illesztési munka nélkül csatlakoztatható legyen.

A probléma súlyát érzékelhetjük, ha meggondoljuk, hogy a különböző kisgép- (és mikrogép-) típusok száma a százat már régen meghaladta, s hogy ezek csatlakozási rendszerei úgyszólván kivétel nélkül eltérnek egymástól. Ugyanakkor a használatban levő programozható mérőkészülékek jelentékeny hányada a programirányíthatóság feltételeit csupán többé-kevésbé vagy esetleg egy-egy géppel kapcsolatban teljesíti. Az egységes csatlakozási rendszerek (CAMAC, HP—IEC) alkalmazása ezért különös jelentőségű.

A teljes hálózat programirányítása

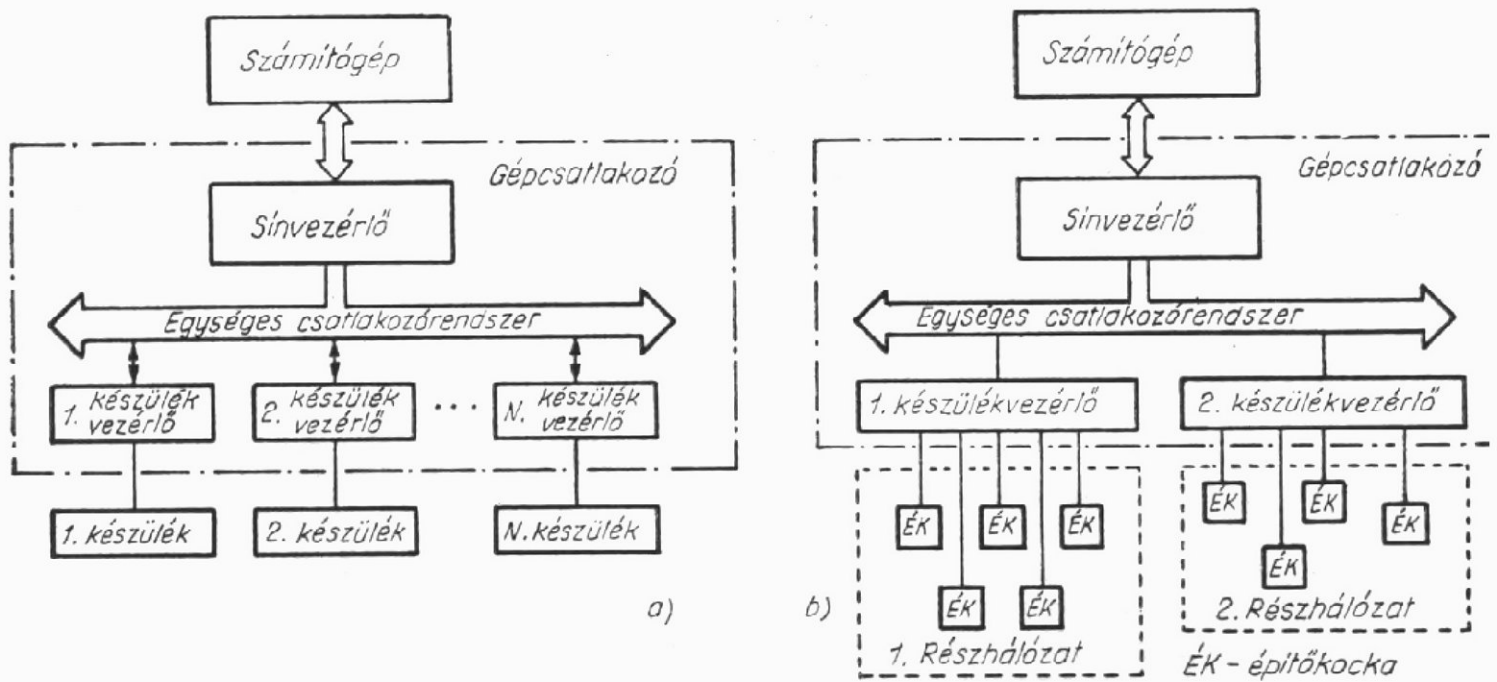
A számítógép-irányítású mérőrendszer a mérőkészüléket mint valamely meghatározott feladat ellátására szolgáló zárt egységet, az őt megelőző korszaktól kapta örökségül. A konszolidálódott területeken, a mérési módszerek és eszközök fejlődésétől függetlenül, a korábban és újabban kialakult mérőkészülék-formák tovább élnek. A mérőkészülékek alapján felépített mérőhálózat a számítógép-irányítású mérőrendszer moduláris szervezőmódjának azonban csak egyik — klasszikus — alakja. Az idők során az olyan természetű méréstechnikai feladatokkal kapcsolatban, melyekhez szokványos mérőkészülékek nem álltak rendelkezésre, egy másik, építőkövekre alapított moduláris szervezőmód is kialakult. Miután bizonyos — kezdetben ad hocnak tűnő — struktúra az építőkövekből komponált mérőhálózatban belül stabilizálódik, kikristályosodik, ez megint csak mérőkészülékké integrálható. E folyamatnak lehetünk tanúi a könyv megírásának időpontjában az adatgyűjtő hálózatokat illetően, melyek kiviteli alakjai a teljesen integrált áramköri egység irányában haladnak.

A mérőhálózat építőelemeként — a zárt formájú készülék és az építőkocka mellett — számolni kell még a kettő keresztezésésként tekinthető moduláris mérőkészülékkel. A mérőkészüléknek modulokra bontását flexibilitási és gazdasági szempontok támasztják alá.

Legtermészetesebb, hogy adott esetben egy-egy hálózatban belül mindháromfajta építőelem, tehát készülék, moduláris készülék és építőkocka egyaránt előfordulhat.

A mérőkészülék, készülékvezérlőjével — a periferikus készülékek mintájára — a számítógéppel közvetlenül vagy egységes csatlakozási rendszeren át köthető össze. A tisztán mérőkészülékekből szervezett hálózat is tartalmaz általában kiegészítő elemeket, pl. kapcsolók programozható együtteseit, programozható terheléseket stb. Ez utóbbiakat — a készülékek mintájára — ugyancsak a számítógéppel való kommunikációt lehetővé tevő készülékvezérlőkkel kell ellátni. A készülékvezérlők együttesét, ami egységes csatlakozási rendszer alkalmazása esetén még a számítógép és a csatlakozás között fekvő csatornavezérlővel vagy esetleg (mint a CAMAC-nál) vezérlőkkel egészülhet ki, nevezzük összefoglalóan gépcsatlakozónak (5.2a ábra).

Ideális esetben a rendszerorientációjú mérőkészülékek mindegyike eleve tartalmazza készülékvezérlőjét, s a rendszerhez előkészületi munka nélkül csatlakoztatható. Az építőkövek önálló készülékvezérlővel nem rendelkeznek, az építőkövekből szervezett mérőhálózat csatlakoztatását a rendszertervezés során tehát el kell végezni. Világos, hogy a csatlakoztatást nem építőköveként kell megoldani; a hálózatot méréstechnikai szempontból funkcionálisan összefüggő, önálló rész-hálózatokra kell felbontani, s egy-egy készülékvezérlőt e rész-hálózatokhoz kell rendelni (5.2b ábra). Ebben az értelemben a rész-hálózatok egy-egy önálló mérőkészülék szerepkörét töltik be. Az esetben, amikor az építőkövekből összeállított mérőhálózat funkcionálisan nem bontható, s egyetlen összefüggő egészet alkot, a hálózat és a gép közötti kapcsolatot egyetlen készülékvezérlő kell, hogy ellássa.



5.2. ábra.

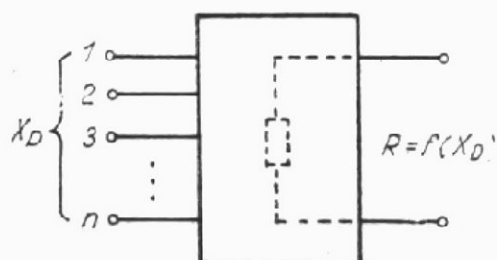
Mérőkészülékek (a), valamint modulokból álló részhálózatok (b) számítógéphez való csatlakoztatása

5.2. Mérőkészülék-elemek és mérőkészülékek programozása

Programozható ellenállások és potenciométerek

Programozható ellenálláson ellenállásokból, analóg kapcsolókból és kiegészítő elemekből (pl. tárolóregiszter, dekódoló stb.) álló olyan elrendezést értünk (5.3. ábra), mely a vezérlő bemenetére adott n komponensű bináris vektor X_D mennyiségi információtartalmát az általa realizált,

$$R = f(X_D)$$



5.3. ábra.

Programozható ellenállás

összefüggésnek megfelelő ellenállásértékre képezi le. Más szóval a programozható ellenállást szám \rightarrow ellenállás D—A konverternek is nevezhetjük. Általános esetben (itt is csak erre tekintünk) az R és X_D között

$$R = R_S X_D$$

lineáris összefüggés áll fenn. A programozható ellenállások különböző változatai egymástól mindenképp digitális kódjuk és szóformátumuk tekintetében térhetnek el. X_D lehet binárisan kódolt, pozitív fixpontos törtszám vagy lebegőpontos pozitív egész szám:

$$\text{fixpontos bináris törtszám: } X_D = \sum_{i=1}^n x_i 2^{-i},$$

$$\text{lebegőpontos bináris egész szám: } X_D = M \cdot 2^E.$$

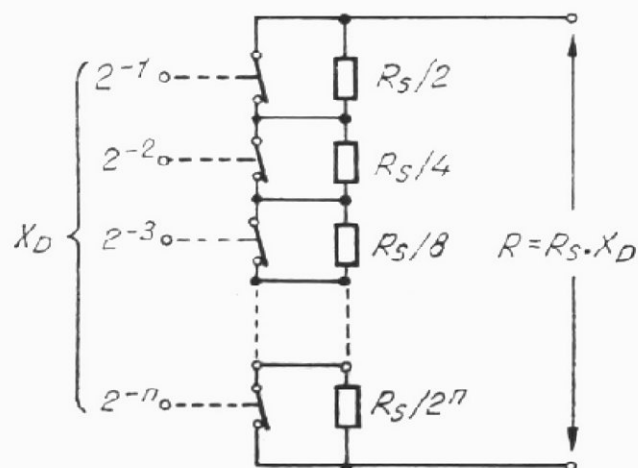
Az utóbbiban M az m -jegyű mantissza, E az e -jegyű karakterisztika, azaz

$$M = \sum_{i=0}^{m-1} x_i 2^i,$$

$$E = \sum_{i=0}^{e-1} x_i 2^i.$$

A kifejezésekben x_i a bináris számrendszernek megfelelően 0 vagy 1 értéket vehet fel.

Az 5.4. ábra fixpontos számformátumú binárisan kódolt ellenállás egy lehetséges megvalósítási alakját szemlélteti ($x_i=0$ -nak a kapcsoló zárt, $x_i=1$ -nek nyitott állása felel meg).



5.4. ábra.
Fixpontos számformátumú, binárisan kódolt ellenállás

A fixpontos számformátumú programozható ellenállás egyik, gyakran problémát okozó tulajdonsága, hogy kvantálási hibája $LSB/2$, a teljes skála mentén azonos. Ez azzal jár, hogy a tényleges kimenő értékek a névlegesre vonatkoztatott relatív kvantálási hibája, a nullpont felé haladva egyre növekszik. A lebegőpontos szám normalizált alakját figyelembe vevő, és M -met egészként kezelő lebegőpontos számábrázolás alapján ezzel szemben olyan programozható ellenállás valósítható meg, melynek értéktartománya:

$$(2^{m-1} R_b) \cong R \cong (2^m - 1) 2^E R_b,$$

ahol $E \cong (2^e - 1)$ és ezen tartományon belül, a kvantálási hiba $1/2 \cdot 2^E$, a relatív kvantálási hiba pedig

$$q = 2^{-m},$$

azaz állandó.

Annak feltételezésével, hogy E -nek az R rögzített felső határához tartozó értékére az $E_{\max} = \epsilon \cong m$ kikötés érvényes, a fenti feltételeknek megfelelő bináris lebegőpontos számformátumú ellenállás az 5.5. ábra ellenálláshálózatával realizálható. A hálózat mantisszaszektorból és karakterisztikaszektorból épül fel. A mantisszaszektor összesen m ellenállást ölel fel, melyek értéke:

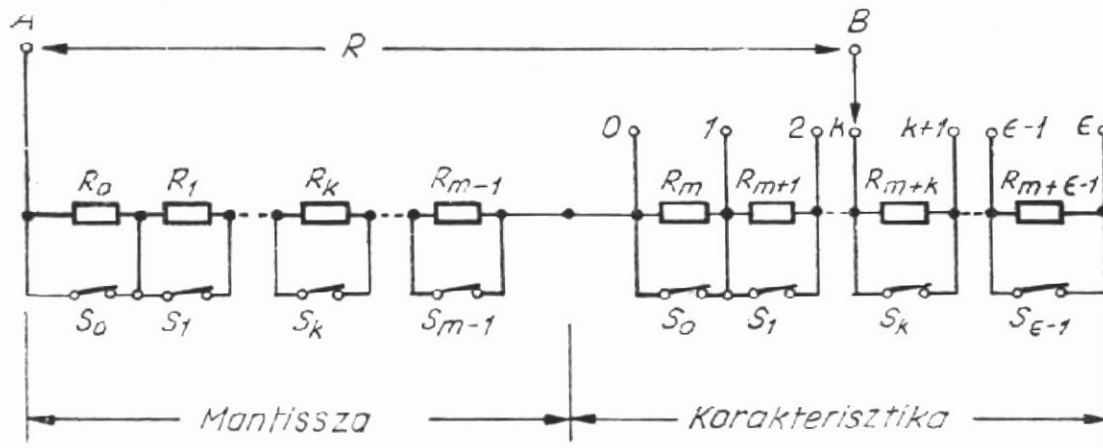
$$R_i = 2^i R_b \quad i = 0, 1, \dots, m-1,$$

a karakterisztikaszektor pedig ϵ ellenállást tartalmaz, melyek értéke:

$$R_{m+i} = (2^m - 1) 2^i R_b \quad i = 0, 1, \dots, \epsilon - 1.$$

Adott lebegőpontos számhoz tartozó ellenállás értéke meghatározott s jelű kapcsolók nyitásával és a $0, 1, \dots, \epsilon$ karakterisztikapontok közül a megfelelő kiválasztásával állítható be az A—B kapcsok között. A kapcsolók nyitott állapotához bináris 1 értéket rendelve és feltételezve, hogy az i -vel megnevezett kapcsolók a két szektorban paralel dolgoznak, az A pont és a k -edik karakterisztikapont közötti ellenállás:

$$R = \sum_{i=0}^{m-1} s_i R_i + \sum_{i=0}^{k-1} s_i R_{m+i}.$$



5.5. ábra. Lebegőpontos számformátumú, bináris ellenállás

Behelyettesítve R_i és R_{m+i} fent megadott értékeit, elvégezve az összevonást és az összeg alól kiemelve $2^k R_b$ -t, eredményül

$$R = \left(\sum_{i=k}^{m-1} s_i 2^{i-k} + \sum_{i=0}^{k-1} s_i 2^{m-k+i} \right) 2^k R_b$$

adódik. A $j=i-k$ transzformációval, a kifejezés

$$R = \left(\sum_{j=0}^{m-k-1} s_{j+k} 2^j + \sum_{j=m-k}^{m-1} s_{j+k-m} 2^j \right) 2^k R_b$$

alakot nyeri. A $(k+j)_m = k+j$ moduló jelöléssel végül az

$$R = \sum_{j=0}^{m-1} (s_{(j+k)_m} 2^j) 2^k R_b$$

eredményre jutunk. Ez R -t a lebegőpontos számformátumnak megfelelő alakban adja meg, de az eredetitől, mely

$$R = R_b X_D = \left(\sum_{i=0}^{m-1} s_i 2^i \right) 2^E R_b,$$

eltérő módon. A megegyezés helyreállítása céljából $k=E$ -t kell választani, a mantisszát pedig, mielőtt az s kapcsolókat működtetné, E pozícióval körkörösén balra kell eltolni. Az eltolás művelete vagy a számítógéppel, vagy az ellenállást kiegészítő működtető elektronikával hajtható végre.

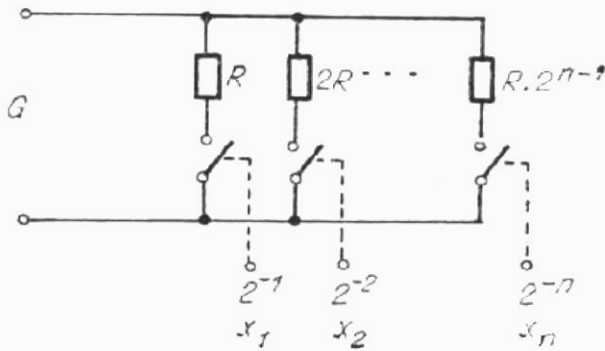
Esetenként ellenállás helyett célszerűbb a vezetőképességet programozni. Az 5.6. ábra alapján írható, hogy

$$G = x_1 \frac{1}{R} + x_2 \frac{1}{2R} + \dots + x_n \frac{1}{R \cdot 2^{n-1}}$$

és

$$G = \frac{2}{R} (x_1 2^{-1} + x_2 2^{-2} + \dots + x_n 2^{-n}) = \frac{2}{R} X_D,$$

$x = 1$ a zárt, $x = 0$ a nyitott kapcsolóállásnak felel meg.



5.6. ábra.

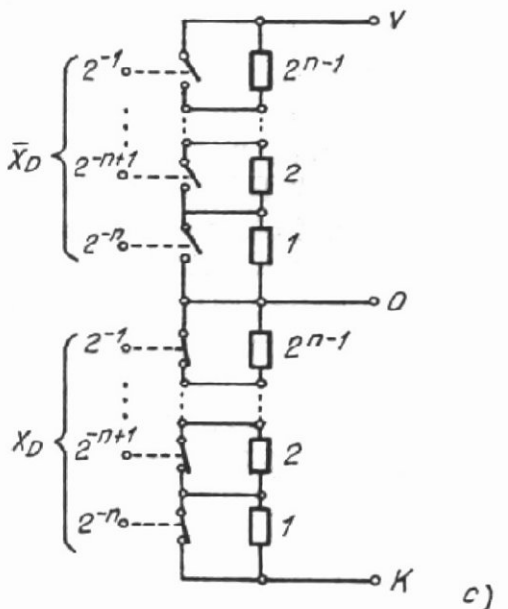
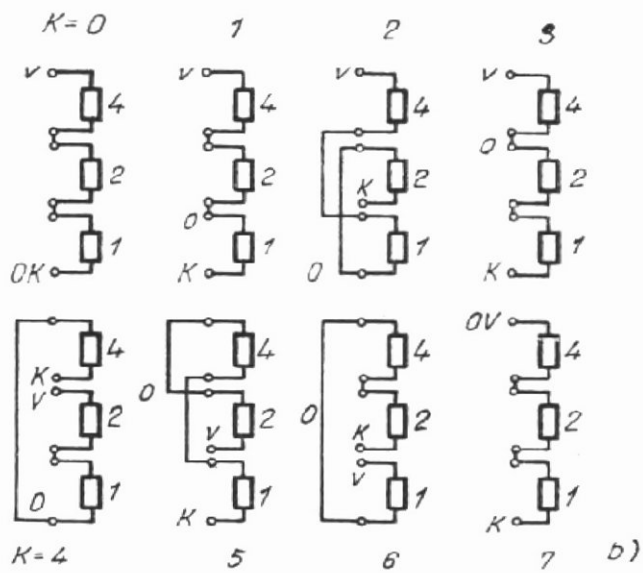
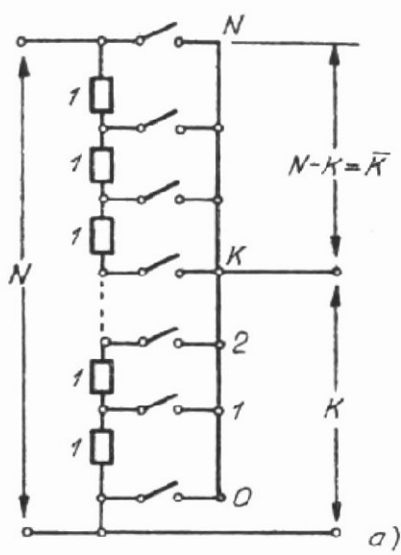
Fixpontos számformátumú, binárisan kódolt vezetőképesség

A *potenciométer*, eredeti értelemben háromkapocspontú ellenállás, melynek K kezdő és V végpontja között O osztáspontot képező mozgó kontaktus futtatható végig. Jellemzője, hogy a kezdőpont és az osztáspont közötti szakasz R_{KO} ellenállása az osztáspont és végpont közötti szakasz R_{VO} ellenállásának a teljes R ellenállásra vonatkozó komplemente, azaz $R_{KO} = \bar{R}_{VO} = R - R_{VO}$. A programozható potenciométerek (éppúgy, mint az ellenállások is) diszkrét ellenállásokból épülnek fel. A felépítésforma a választott kód és számformátum függvénye. Az alábbiakban csak a fixpontos számformátumú lineáris potenciométerekre térünk ki — általában ezt alkalmazzák — bár lebegőpontos számformátumú potenciométerek is megvalósíthatók az ellenállásokra fent elmondottak alapján; a lineáristól eltérő karakterisztika is lehetséges.

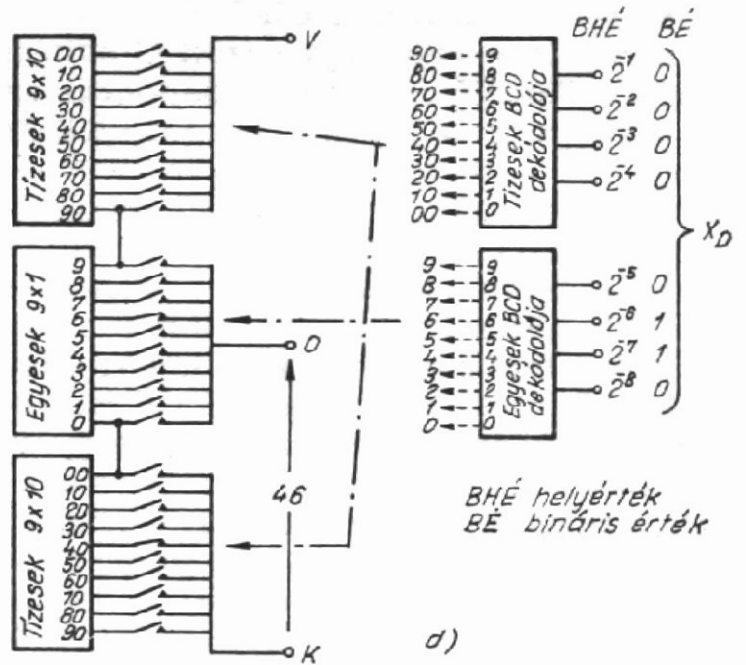
Az 5.7. ábra, melyen az ellenállásokat relatív értékük jelöli, különböző potenciométer-megoldásokat mutat be. Legegyszerűbb szerkezete az *a* jelű, egyforma ellenállásokból komponált potenciométernek van. Az osztáspont az $N+1$ kapcsolóból egy kiválasztásával határozható meg. Az *effajta* potenciométernek egyrészt az a hátránya, hogy az azonos felbontóképességű, más kódrendszerű potenciométerekhez viszonyítva igen sok ellenállást és kapcsolót tartalmaz, másrészt különösen a felbontás (tehát az ellenállások és kapcsolók számának) növekedtével, egyre nehezebb — bináris/ N -ből az egy — dekódolót igényel. A *b* jelű (példaképpen $1/(2^3 - 1)$ felbontású) bináris potenciométer az ellenállások száma tekintetében gazdaságos, azonban kapcsolórendszere és annak kezelése bonyolult. Az ellenállások és kapcsolók számát illetően kielégítően gazdaságos, ugyanakkor dekódoló nélkül, közvetlenül a számítógép binárisan kódolt X_D vektorával vezérelhető a *c* jelű potenciométer. A *d* jelű, dekadikus (tehát decimális kódolású) potenciométer egy egyes és két tizes ellenállásdekádból épül fel. A három ellenállásdekád mindegyikében egyidejűleg csak egy-egy kapcsoló lehet zárva, és pedig a tizes dekádokban az azonos nevűek. Így az egyik tizes dekádból mindig ugyanannyi ellenállás iktatódik ki, mint amekkora a másikkól a potenciométer körében jelen van. A BCD kódolású X_D magasabb helyértékű számjegye az egyes dekádnak a potenciométer ellenállásláncában elfoglalt helyét, a kisebb helyértékű számjegy pedig az egyesek láncán belüli megcsapolás helyét definiálja.

Ellenálláshíd programvezérlése

Az 5.8. ábra ellenálláshíd két változatát mutatja: az ismeretlen ellenállás meghatározására szolgáló R_x mérőt és az ismert ellenállás megváltozása vagy az ellenállás névleges értékétől való eltérése megállapítására szolgáló ΔR mérőt.



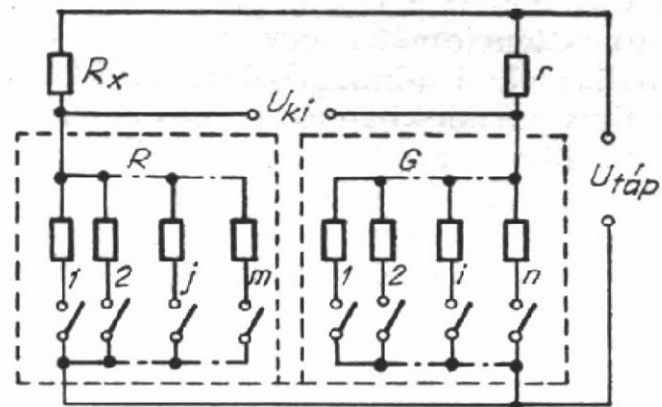
$x_i = 0$ zárt, $x_i = 1$ nyitott kapcsoló



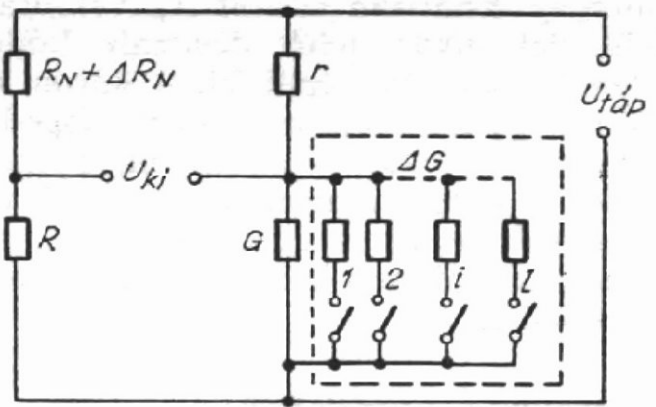
5.7. ábra.

Digitális potencióméterek

(a) N-ből a K kiválasztású; (b) bináris értékláncú; (c) bináris és komplementer kódolású; (d) dekadikus



Méréstartomány; kiegyenlítés
 R_x -mérő



ΔR -mérő

5.8. ábra.

Programozható ellenállás-, ill. deviációmérő híd

Az R_x mérő mérési tartománya az R ellenállás értékének m -ből az 1 választásával program útján állítható be. Az ismeretlen R_x , az R és r ellenállások, valamint a G vezetőképesség között a híd kiegyenlített állapotában az

$$R_x = rR_jG$$

összefüggés érvényes. Kiegyenlítő ágként G -t választva, s az ábrán látható módon programozható vezetőképességként képezve ki, a fenti összefüggés

$$R_x = rR_jG(X_D)$$

alakot veszi fel, ahol X_D a hidat vezérlő bináris vektornak a híd kiegyenlített állapotához tartozó értéke. A híd kiegyenlítésével egyúttal X_D -t, s így közvetve R_x -et is megkapjuk. G a közvetlen párhuzamos D ...A átalakítók ellenálláshálózataihoz hasonlóan képezhető ki bináris vagy pl. BCD kódolással. A 5. 8. ábra a bináris kódolásmódot szemlélteti, mely esetben

$$G(X_D) = G_S X_D = G_S \sum_{i=1}^n x_i 2^{-i},$$

ahol G_S a korábbiakból már ismert skálatényező.

A ΔR mérő az R_x mérőtől csupán a G mellett fekvő ΔG vezetőképességgel különbözik. A mérés a hídnek az ellenállás R_N névleges (vagy kezdő) értékére való „kiegyenlítésével” kezdődik. A program kijelöli az R_N -nek megfelelő méréstartományt (R_j), és az R_N -re kiegyenlített hídnek megfelelő G értéket (közben ΔG kapcsolói nyitott állapotban találhatók), azaz teljesíti az

$$R_N = rRG$$

feltételt. Ezután ΔG értékének beállításával a híd tényleges kiegyenlítése következik ($R_N + \Delta R_N$)-re. Kiegyenlített állapotban:

$$R_N + \Delta R_N = rR(G + \Delta G).$$

A két utolsó egyenlet alapján:

$$\Delta R_N = rR\Delta G \quad \text{és} \quad \frac{\Delta R_N}{R_N} = \frac{\Delta G}{G}.$$

ΔG bináris kódolása esetén tehát:

$$\Delta R_N = rR\Delta G_S Y_D = rR\Delta G_S \sum_{i=1}^l y_i 2^{-i}$$

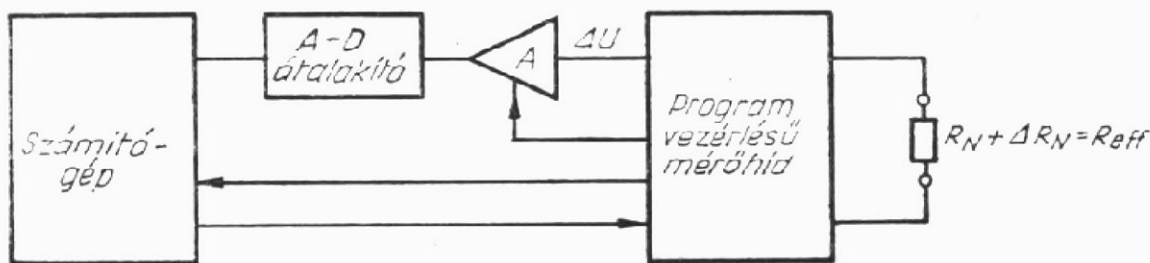
és

$$\frac{\Delta R_N}{R_N} = \frac{\Delta G_S}{G_S} \cdot \frac{Y_D}{X_D} = \frac{\Delta G_S}{G_S} \cdot \frac{\sum_{i=1}^l y_i 2^{-i}}{\sum_{i=1}^n x_i 2^{-i}}.$$

Mérőátalakító rendeltetésű hidakban R és G egyaránt állandó értékű.

Deviációmérő (ΔR) híd számítógéppel pl. az 5.9. ábrának megfelelően az alábbi módon működhet együtt. A mérési feladat:

1. Megállapítani, hogy az ellenállás tényleges értéke az előírt toleranciahatáron ($\pm 10\%$) belül van-e.
2. Feltéve, hogy az eltérés $< 10\%$, megállapítani az eltérés pontos értékét.



5.9. ábra.
Számítógép-irányítású deviációmérő

A mérés első lépéseként a számítógép a memóriájában őrzött R_N értékkel programozza a hidat. A hídnek két érzékenységi állapota van, melyek közül a megfelelőt, az U_{ki} kimenő feszültség értéke alapján a híd maga választja ki, részben az $U_{táp}$ tápfeszültség, részben az A erősítésének beállításával. A kevésbé érzékeny állapotban az A—D átalakító végkitérése 10%, az érzékenyebb állapotban 0,1% ellenálláseltérésnek felel meg. A mindenkor érvényben levő végkitérés nagyságát a híd a számítógéppel közli.

Amennyiben a vizsgált ellenállás eltérése $>0,1\%$ de $<10\%$, a híd a kevésbé érzékeny üzemállapotba kapcsol és mér. A mérési eredmény a hídérzékenység adataival együtt a számítógépbe kerül, mely a hídmodell alapján számítja ΔR_N közelítő értékét, majd az ennek megfelelő ΔG értékkel a hidat kiegyenlíti. Feltételezés szerint e durva kiegyenlítés után a kimenő feszültség a 0,1% eltérésnek megfelelő érték alá esik, s a híd az érzékenyebb mérési állapotra vált át. Az új mérési eredmény és a durva mérés során nyert $\approx \Delta R_N$ alapján a számítógép az eltérés pontos ΔR_N értékét számítja. Ha a híd kimenőfeszültsége már a mérés kezdetén $<0,1\%$ eltérést jelez, úgy a mérés eleve nagyobb hídérzékenységgel történik, és a számítógép közvetlenül ΔR_N -t számítja.

Ismereteseek nem csupán ellenállás, hanem ellenállás, induktivitás és kapacitás mérésére szolgáló automatikus, programozható mérőhidak is (General Radio 1683), s a számítógép, esetenként, számos mérőhíd egyidejű irányítását is elláthatja.

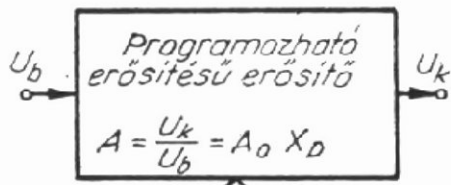
Programozható erősítők

Az erősítő, melynek erősítése digitális jellel állítható, olyan szorzó D—A átalakítóként tekinthető, melynek kimenő feszültsége a bemenő jel nagyságával és az X_D digitális mennyiséggel arányos. Az 5.10. ábra különböző erősítőket és az erősítés programozhatóságának különböző módjait szemlélteti. Az ábrán megadott összefüggésekben az A erősítést jelöl.

A műveleti erősítő erősítése kézenfekvően az R_b bemenő ellenállás (vagy inkább bemenő vezetőképesség) változtatásával programozható. R_b struktúrája a választott kódtól függően alakul. A *b)* ábra bináris kódnak megfelelő struktúrát tüntet fel, melynek kapcsolórendszere közvetlenül binárisan kódolt digitális jellel vezérelhető.

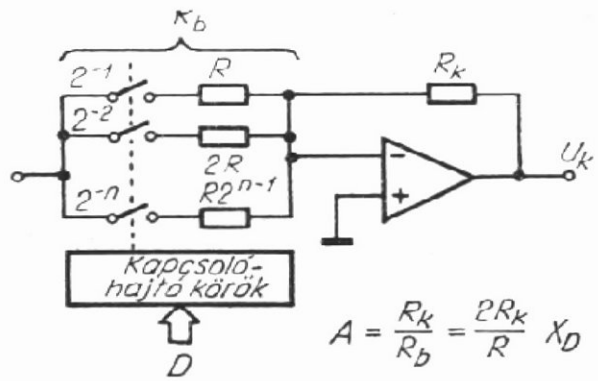
A kapcsolórendszer megvalósításával kapcsolatos gyakorlati szempontok alapján esetenként célszerű lehet a *c)* rajzon látható megoldást választani, melynek jellegzetesége, hogy az összes kapcsoló közös pólusa földpotenciálon található.

A *d)* áramkör elsősorban az erősítésnek nagy lépésekben való változtatására (tartományváltás) alkalmazható. Az általában N számú kapcsoló közül itt mindig csak egyetlen egy zár, a többi kapcsoló ugyanakkor nyitva marad. A zárt, i -edik kapcsoló és a föld között fekvő ellenállásszakaszt R_{bi} -vel, a zárt kapcsoló és a kimenet közötti szakaszt R_{ki} -vel jelölve, az erősítés $A = (R_{ki}/R_{bi}) + 1$. Az A_1, A_2, A_3 érték az ábra konkrét áramköri viszonyaira adódik. Rá kell mutatni a *d)* erősítő-



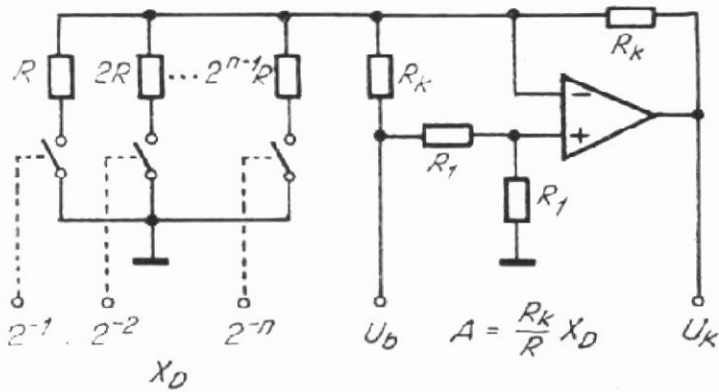
Digitális jel : D

a)

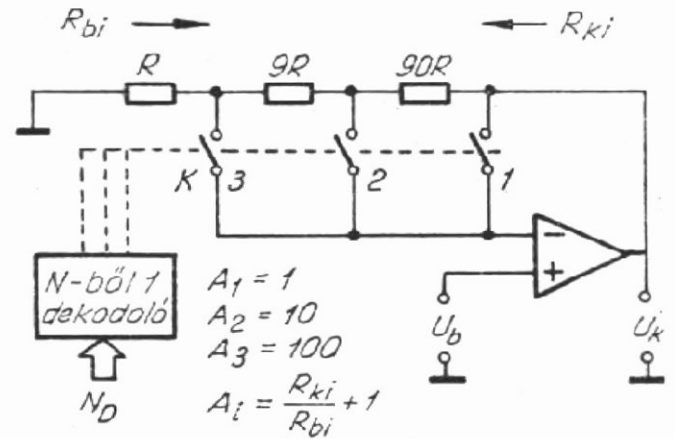


b)

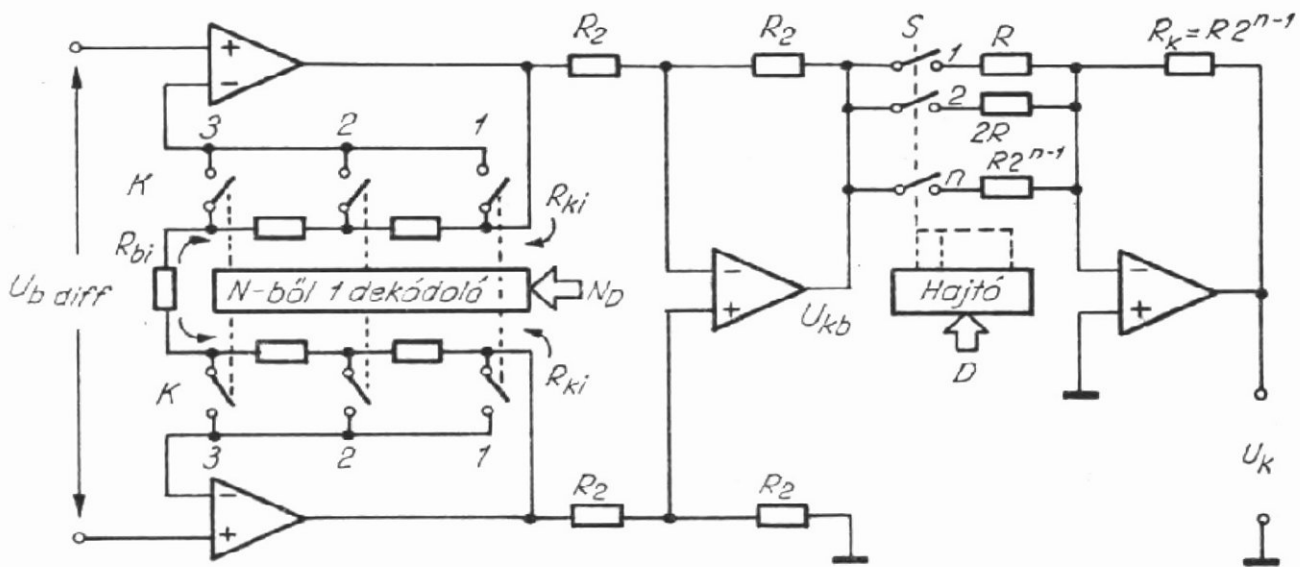
$$A = \frac{R_k}{R_b} = \frac{2R_k}{R} X_D$$



c)



d)



$$A = A_1 A_2; \quad A_1 = \frac{U_{kb}}{U_{b,diff}} = 1 + \frac{2R_{ki}}{R_{bi}}; \quad A_2 = \frac{2R_k}{R} X_D = 2^n X_D$$

e)

5.10. ábra. Programozható erősítők

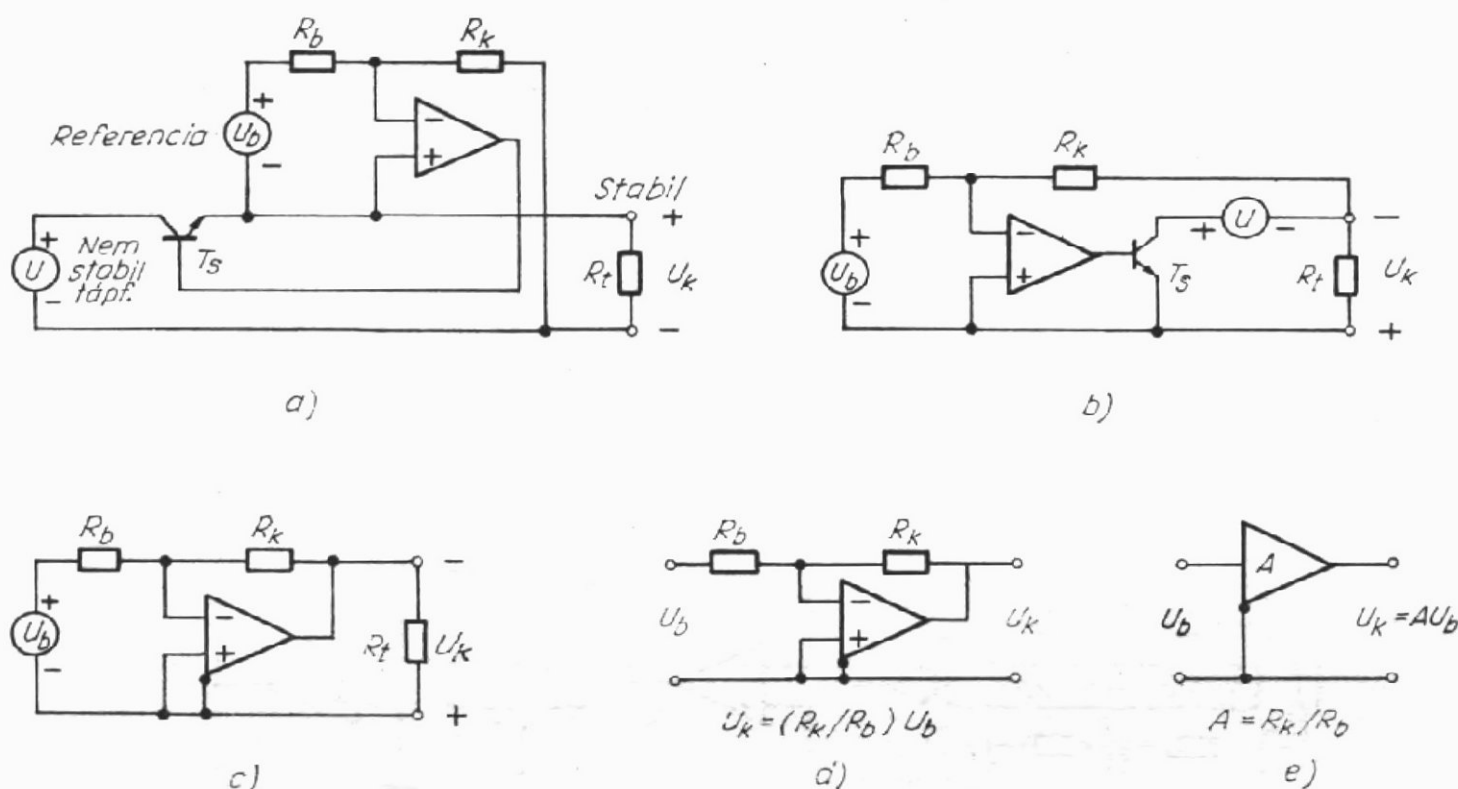
(a) alapséma; (b...d) műveleti erősítő programozási lehetőségei; (e) adaterősítő (programozott mérőerősítő)

megoldás még egy igen fontos tulajdonságára: bemenő ellenállása az előbb ismertetett operatív erősítők bemenő ellenállásánál lényegesen nagyobb.

Az *e*) erősítő szimmetrikus bemenetű mérőerősítő (2.17. ábra), valamint a *b*) ábra operatív erősítőjének együttese. Az erősítés programozása ez esetben két fozatban történik. Az N -ből 1 kódolású K jelű kapcsolók (melyek közül az egynevéűek együtt zárnak, ill. nyitnak) az erősítéstartomány beállítására szolgálnak; az N_D digitálisan kódolt számot a dekódoló kapcsolószámra képezi le. Az erősítés egy tartományon belüli finombeállítása a D -vel vezérelt S kapcsolórendszer útján történik.

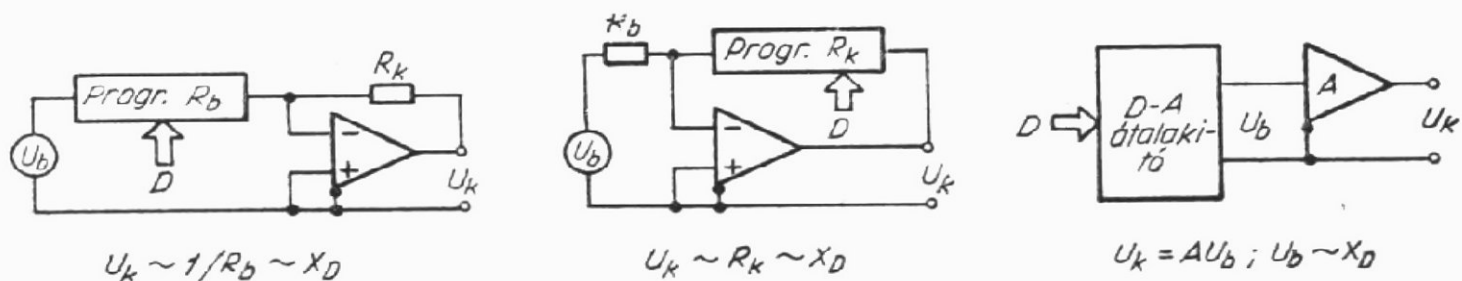
Tápforrások programvezérlése

A soros szabályozású feszültségstabilizátor — a referenciafeszültségre vonatkozóan — operatív erősítőként fogható fel. Az 5.11. ábrán a betűjelölések sorrendjében nyomon követve a rajzokat, állításunk helyessége belátható. Az 5.11a rajz a tápegység kapcsolási vázlatát mutatja be, az 5.11d és e) pedig már a végső, redukált vázlatok. Ez utóbbiak alapján kitűnik, hogy a kimenő feszültség: $U_k \sim 1/R_b$ vagy



5.11. ábra.
Egyenfeszültségű soros tápegység mint műveleti erősítő
(levezetés az *a*...*e* betűk sorrendjében)

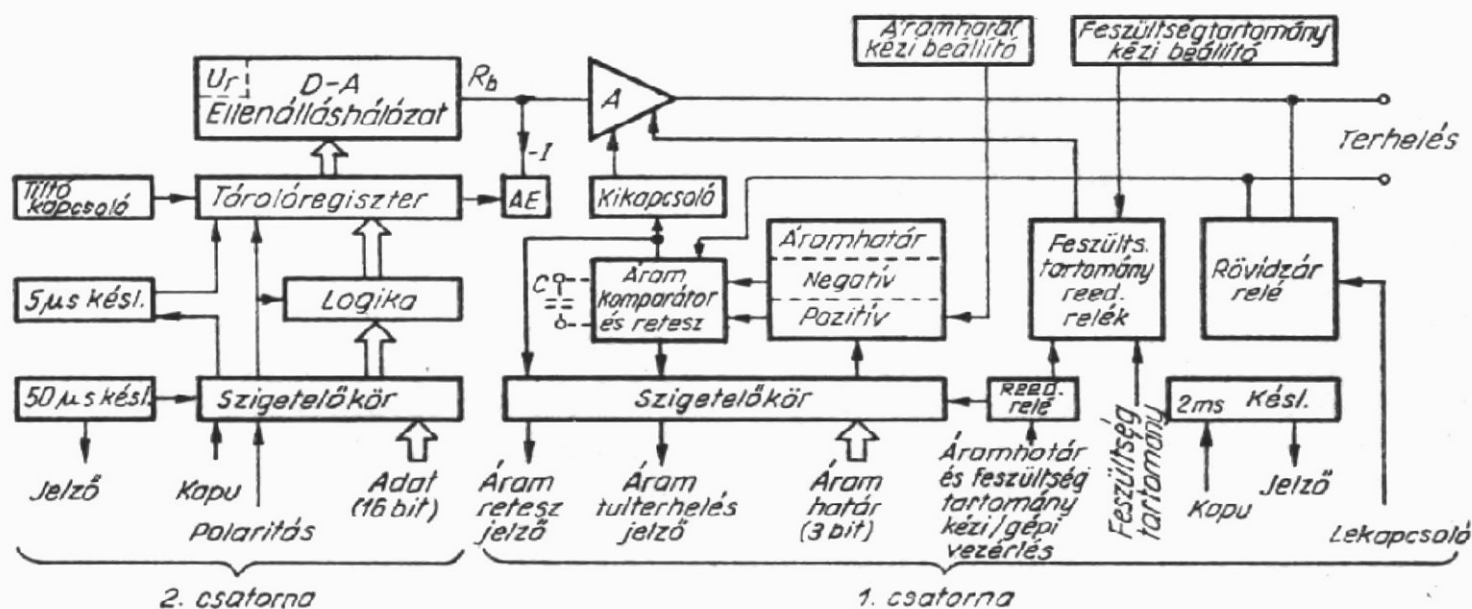
$U_k \sim R_k$ vagy $U_k \sim U_b$; a tápforrás tehát háromféle programozási lehetőséggel rendelkezik (5.12. ábra). $\sim 1/R_b$, ill. $\sim R_k$ programozásmód esetén, programozható ellenállások alkalmazásával a tápegység teljesítménykimenetű D—A átalakítónak alakul át. Mindkét esetben $U_b = \text{konstans}$. Az $\sim U_b$ programozásmód esetében a referenciafeszültség helyére külön D—A átalakító analóg kimenete csatlakozik. Gyakran egy tápegységben belül kétféle programozásmódot (pl. $\sim U_b$ és $\sim R_k$) is alkalmaznak; az egyiket tartományváltásra, a másikat tartományon belüli finombeállításra.



5.12. ábra.

Az egyenfeszültségű (műveleti) tápegység kimenő feszültségének programozása

(a) $1/R_b$ arányos; (b) R_k arányos; (c) U_b arányos



5.13. ábra.

$1/R_b$ és R_k arányos programozású egyenfeszültségű tápforrás

Az 5.13. ábra $\sim 1/R_b$, $\sim R_k$ programozású stabilizált feszültségforrás felépítésmódját mutatja be [12]. Az operatív tápforrás alapáramkörét az ábrán A betű jelzi. A feszültségtartomány reed-relés kapcsolórendszerrel program útján vagy manuálisan állítható be R_k változtatásával. A kimenő feszültség programvezérlése ellensúlyozott bináris vagy BCD kódolású D—A ellenálláshálózat útján történik (2.47a ábra). A bináris adatvektor 16 bit hosszúságú. A polaritásváltásra külön bit szolgál.

A számítógép által küldött vezérlőinformációt a D—A hálózat részére tárolóregiszter rögzíti. A tárolóregiszter lehetővé teszi adott kimenő feszültség hosszú idejű fenntartását a számítógép lefoglalása nélkül. A regiszterbe való beírást a tápegységhez tartozó, de az ábrán nem látható címfelismerő áramkör kapujele végzi. A kapujel a regiszteren az adat beérkezése után $5 \mu s$ késéssel jelenik meg, miután a stacionárius viszonyok a regiszter bemenetén már kialakultak. A tápegység által táplált áramkörökre veszélyes tüskék elkerülése érdekében fontos, hogy az új adat digitális komponensei a D—A ellenálláshálózatra megfelelő időzítéssel kerüljenek.

A tápforrás áramkorlátozó köre egyrészt rövidzár-védelmet nyújt, másrészt lehetővé teszi a max. kimenő áram program útján való meghatározását. Az áramkomparátorra a szükségletnek megfelelő C kapacitás kapcsolható a komparátor működésének késleltetése céljából, ha a tápforrás a programozott áramhatáron túlterjedő rövididejű áramigényeket (terheléslökéseket) kell, hogy kielégítsen.

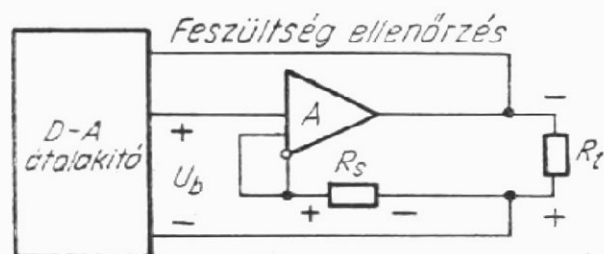
A rövidzár állapotának bekövetkeztéről az áramretesz-jelző, a programozott áramhatár túllépéséről az áramtúlterhelés jelző tudósítja a számítógépet.

A számítógépről lekapcsolt tápforrás kimenetét rövidzárrelé söntöli.

A digitális rendszert (gépcsatlakozót) a tápegységtől szigetelőkörök választják el. A szigetelés elsősorban azért szükséges, mivel a potenciálkép szempontjából át nem gondolt kétoldali földelés rövidzárt okozhat, vagy egyoldali földelés esetén a digitális rendszer a földtől eltérő potenciálra kerülhet. Az elválasztás másrészt abból a szempontból is kívánatos, nehogy a kétoldali földelés következtében elektromágneses tér, földelési kúszóáramok stb. révén a D—A átalakítóba hibajel kerüljön. A szigetelést relék, *RF* csatolás, induktív vagy optikai csatolás útján oldják meg. A relé lassú működésű; mérsékelt sebességi követelményeket támastó áramutakon alkalmazható. Az *RF* és induktív szigetelők kielégítik ugyan az itt adódó sebességi követelményeket, de viszonylag bonyolultak; helyüket a fejlődés újabb szakaszában az optikai csatolók foglalták el.

Az ábrából látható, hogy a tápforrásnak a gépcsatlakozó (készülékvezérlő) irányában két külön csatornája van, melyek mindegyikének önálló címe (kapu) van. Az 1. (lassú) csatornán át történik a tápegység feszültségtartományának és áramhatárának programozása (választás szerint ez kézi úton is végezhető). Az adatok beprogramozása, tekintettel a relékapcsolókra, 2 ms időt vesz igénybe, melynek elteltéről a 2 ms késleltető jelző kimenete tudósít. A 2. (gyors) csatorna, mely az egy feszültségtartományon belüli finombeállítást végzi, 50 μ s műveleti időt igényel. Másodpercenként tehát 20 000 változtatás végezhető a D—A hálózat bemenetén. Ez lehetővé teszi, hogy a tápforrást programozott erősítő minőségében a számítógép által vezérelt teljesítménykimenetű jelgenerátorként is lehessen alkalmazni.

Áramforrás programozása a feszültségforrás programozásához hasonlóképpen végezhető. Az 5.14. ábrán áramforrás kimenő áramának D—A átalakító analóg feszültségével való programozásmódja látható. Áramforrás esetében a kimenet fe



5.14. ábra.
Programozott egyenáram-forrás

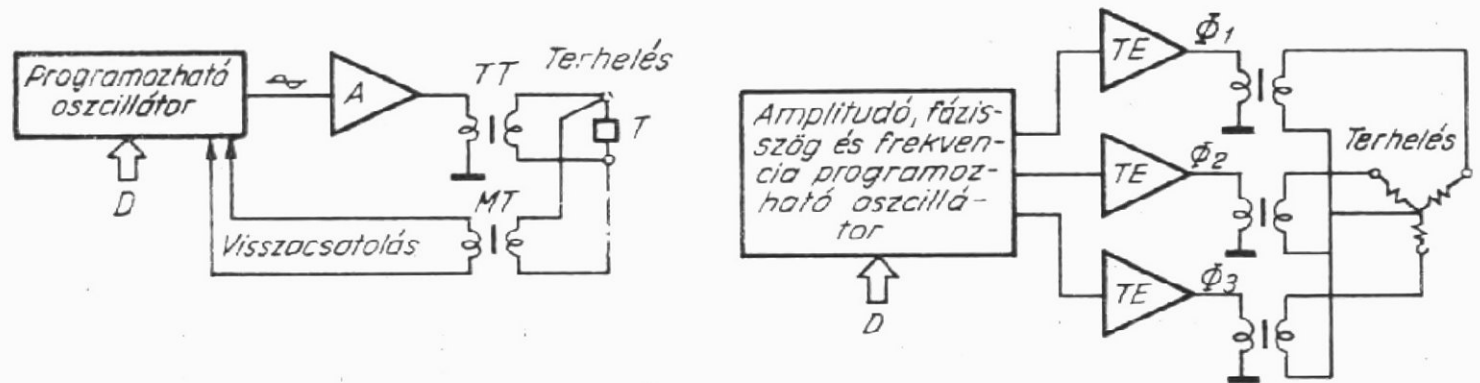
szültségét egyrészt a teljesítményerősítő védelme, másrészt az R_t terhelés felszültségvédelme szempontjából korlátozni, ill. ellenőrizni kell.

Automatikus mérőrendszerek gyakran igénylik, hogy a tápforrás programozott kimenő egyenfeszültségének (egyenáramának) szintjére meghatározott amplitúdójú váltakozóáramú komponenst lehessen felvinni. Célszerűen a programozható egyenáramú tápegységnek e jelkomponens ráültetésére szolgáló analóg bemenettel is kell rendelkeznie.

A műszaki mérés technika számos vizsgálata során szükség lehet változtatható jelparaméterű váltakozóáramú tápforrás alkalmazására. Vizsgálni kell pl. a hálózatról táplált elektronikus, ill. elektromos készülékeknek a hálózati tranziensekre való érzékenységét vagy e készülékek viselkedését olyan körülmények között, amikor a tápfeszültség paraméterei eltérnek a névlegestől. Az ilyen jellegű vizsgálatoknak, különösen járműveken, hajókon, repülőgépeken alkalmazott készülékek ellenőrzése során van jelentősége, mely készülékek a földi viszonyoknál lényegesen rosszabb tápköri viszonyok között kell, hogy dolgozzanak. A nagyteljesítményű,

változtatható paraméterű, váltakozóáramú tápforrások másik jelentékeny alkalmazási köre a különböző rázó- és rezgésvizsgálatok területére esik. A mérések automatizálása megköveteli e tápforrások programozhatóságát.

A programozható váltakozóáramú tápforrás programozható oszcillátorból és teljesítményerősítőből vagy erősítőkből épül fel, melyek TT teljesítménytranszformátorral kapcsolódnak a terhelésre. Fokozottabb pontossági igények esetén a terhelés kapcsain megjelenő feszültség mérőtranszformátoron át visszakerül az oszcillátorra, a névlegessel való összehasonlítás, a kimenő feszültség esetleges korrekciója céljából. A tápforrás lehet egy- vagy többfázisú (5.15. ábra). Több fázis



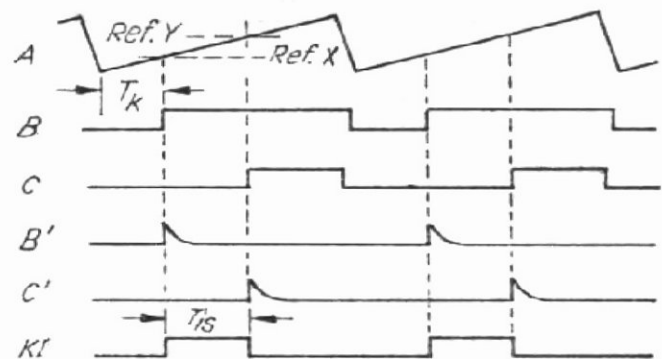
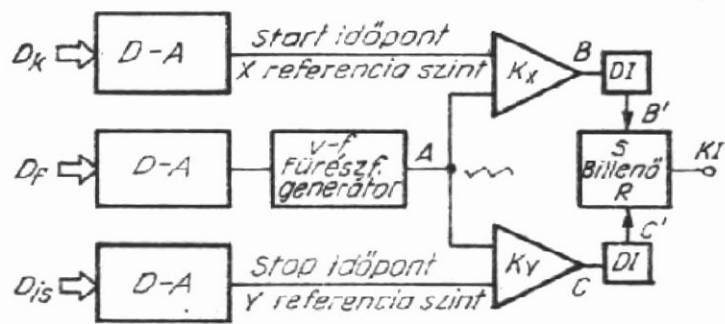
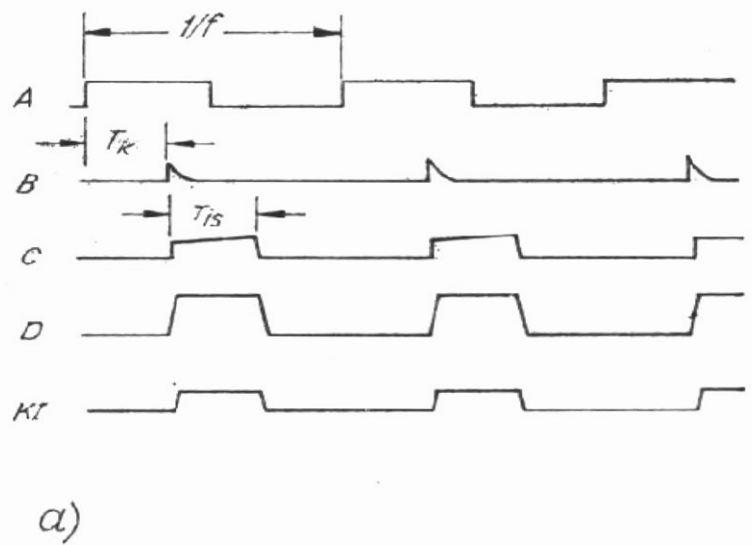
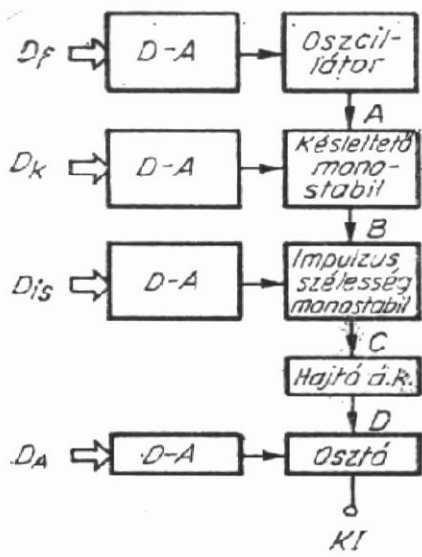
5.15. ábra.
Egy- és többfázisú programozott váltakozóáramú feszültségforrás

esetén minden egyes fázis külön érzékelő—visszacsatoló körrel rendelkezhet. Általánosságban programozható a frekvencia, az egyes fázisok kimenő feszültsége külön-külön, továbbá az egyes fázisoknak egymáshoz vagy egy referenciafázishoz viszonyított szöge is.

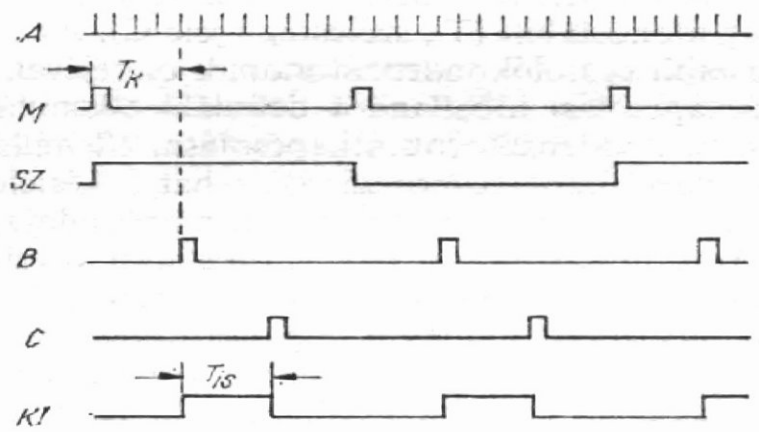
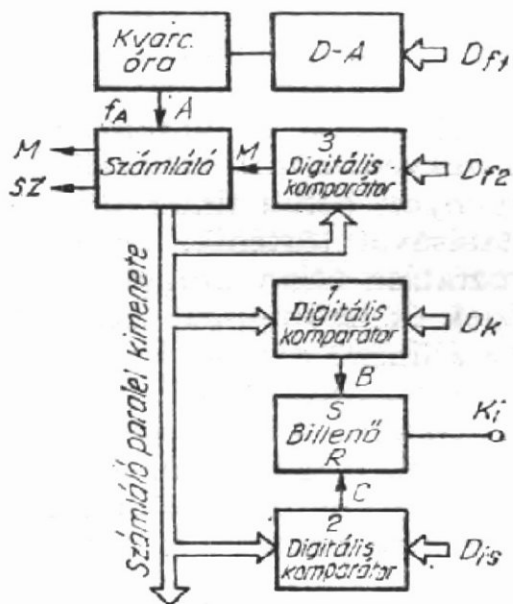
Impulzusgenerátor programozása

A klasszikus felépítményű impulzusgenerátor (5.16. ábra) szabadonfutó multivibrátorból (oszcillátor), késleltetést, ill. impulzusszélességet meghatározó monostabilokból, teljesítménykimenetet nyújtó hajtóáramkörből és osztóból épül fel. Az oszcillátor A jelének felfutó éle indítja a késleltető monostabilt; a késleltető monostabil T_k tartalmú jelének hátsó éle (differenciált megfelelője B) indítja az impulzusszélesség monostabilt (T_{is} szélességű jele C). Az f ismétlési frekvencia tartományállítás az astabil csatolókapacitárainak cseréjével, a tartományon belüli finombeállítás az átkapcsolási időállandót definiáló ellenállás változtatásával történik. Hasonlóképpen, kondenzátorok átkapcsolása, ellenállások változtatása útján állítják a két monostabil impulzushosszának, tehát a késleltetési időnek és az impulzusszélességnek a nagyságát is. A kimenő jel amplitúdójának durva állítását osztó-fokozatkapcsolóval, finomállítását potenciométerrel végzik. Bár a kondenzátorok átkapcsolása és az ellenállásbeállítás is analóg kapcsolórendszerek közvetítésével, digitális jelekkel elvégezhető, s így a jelparaméterek programozhatósága megoldható, a klasszikus felépítményű sem a digitális vezérléstechnikához, sem a korszerű technológiához alapjában nem illeszkedik, s a kimenő jel paramétereinek pontossága sem elégti ki a korszerű mérés technikai követelményeket.

A módosított analóg szerkezetű impulzusgenerátorban az oszcillátor szerepét nagy pontosságú fűrészel-generátor tölti be, mely a 2.27. ábrán bemutatott $v-f$ átalakítóhoz hasonló módon épülhet fel, s melynek frekvenciája D_f digitális jellel állítható, egyrészt az integráló kapacitás átkapcsolása, másrészt a fűrészel-generátor mint



b)



c)

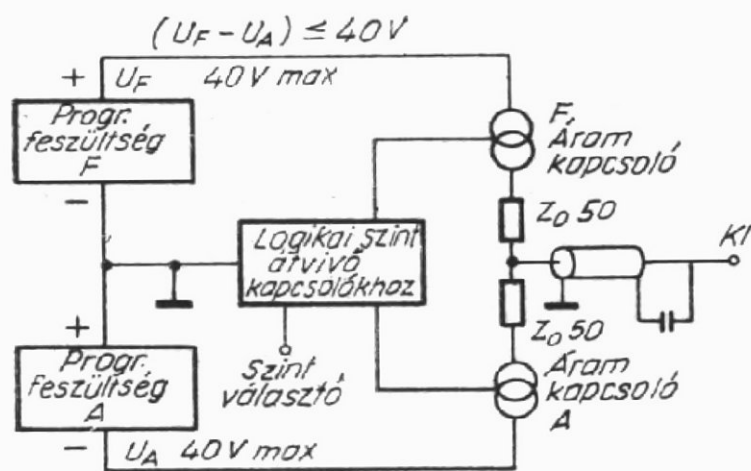
5.16. ábra.
Impulzusgenerátor

(a) klasszikus; (b) módosított analóg és (c) digitális strukturájú változat

v—f átalakító analóg bemenetének változtatása révén. Az állandó amplitúdójú lineáris fűrészjelet az X és Y szinten egy-egy komparátor figyeli. A K_x komparátor az X szint elérésekor a DI differenciáló tagon keresztül átállítja a kimenetéhez csatlakozó billenőkört; a K_y ugyanezt a billenőkört az Y szint elérésekor visszaállítja. Tekintettel a $T_k \sim X$ és $(T_k + T_{is}) \sim Y$ kapcsolatra, \bar{D}_k -val az impulzus késleltetése, D_k és D_{is} -sel együttesen az impulzus szélessége programozható. Az impulzusgenerátornak ezen felépítésmódja mind a programozhatóság, mind a korszerű technológia követelményeihez lényegesen jobban idomul, mint a klasszikus szerkezet, és jelparamétereinek pontossága tekintetében is felülmúlja azt, mégis, lehetőségeiben messze elmarad a tisztán digitális szerkezetű impulzusgenerátor mögött.

A digitális szerkezetű impulzusgenerátor alap-jelforrásaként kvarcoszcillátor-bázisú precíziós óragenerátor szolgál, melynek frekvenciája dekadikusan programozható (pl. 100 Hz...100 MHz között). Az óragenerátor jeleit N számkapacitású szinkron számláló regisztrálja. A számláló bináris paralel kimenetére három digitális komparátor csatlakozik. Ha a számláló tartalma a komparátorok valamelyikében tárolt számmal egyenlővé válik, úgy e komparátor kimenő jelet ad. A 3. sz. komparátor M kimenete a számláló nullázó bemenetére lép, D_{f2} -vel tehát a számláló által ténylegesen regisztrálható $n_3 \leq N$ szám programozható. Az órajel frekvenciája és n_3 (tehát D_{f1} és D_{f2}) együttesen határozza meg az impulzus f_A/n_3 ismétlődési frekvenciáját. Az 1. és 2. sz. komparátor n_1 , ill. n_2 tartalma ($n_1 < n_2 < n_3$) a billenőkör átállítási, ill. visszaállítási idejét adja meg, s így D_k a késleltetési időt, D_k és D_{is} együttesen az impulzus szélességét definiálja.

Az impulzusgenerátor digitális szerkezete lehetőséget ad mind a periódusidő, mind a késleltetési idő, mind pedig az impulzusszélesség igen finom fokozatokban való változtatására (a XINCOM PINTOFILÉ generátor esetében pl. mindhárom paraméter inkrementumának minimuma 10 ns). Hasonló felbontóképesség és ennek megfelelő pontosság kívánatos az impulzus amplitúdójának állíthatósága tekintetében is, mégpedig nem is a generátor kimenetére, hanem a vizsgálat tárgyának bemenetére vonatkoztatva. E követelmények az 5.17. ábrán látható, vagy ahhoz hasonló kimenet-



5.17. ábra.
Impulzusgenerátor programozott kimenő fokozata

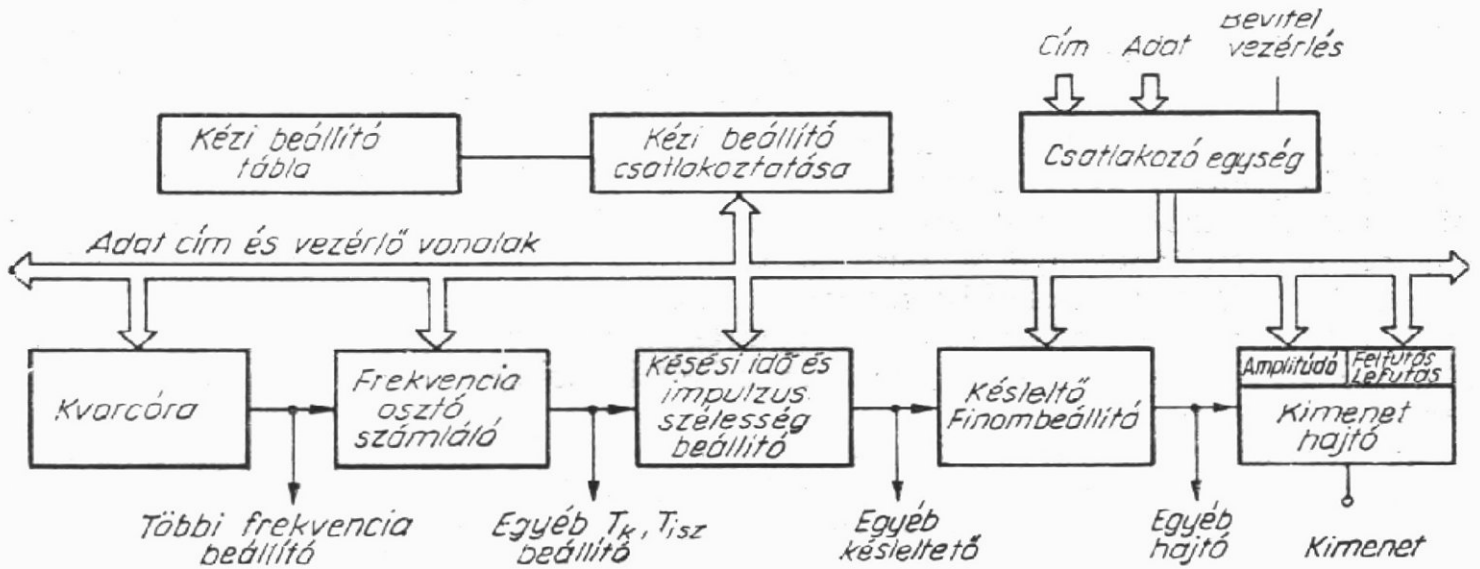
hajtó áramkörrel valósíthatók meg. Az áramkörben a kimenő impulzus alsó és felső szintjének feszültsége az A és F tápforrás U_A , ill. U_F feszültségével programozható. A logikai szintátvívő a TTL/DTL logikai szinteket az áramkapcsolók vezérlőjeleivé, végső fokon a kimenet logikai szintjeivé transzformálja. A csatlakozókábel lezárása a generátoroldalon helyezkedik el; külső végén az impedancia tetszőleges.

Bizonyos esetekben szükség van a felfutási és lefutási idő program általi állíthatóságára is. Ilyenkor a kimenő-hajtó fokozatot olyan lejtőgenerátorként képezik ki,

melynek feszültség/időegységben számított emelkedési/esési szöge program útján változtatható.

Az impulzusgenerátor frekvenciaosztó számlálójának paralel bináris kimenetére nemcsak egy, hanem számos komparátorpáros is elhelyezhető, melyek mindegyikének T_k , T_{is} ideje önállóan programozható, miáltal többkimenetű, többfázisú impulzusgenerátor áll elő.

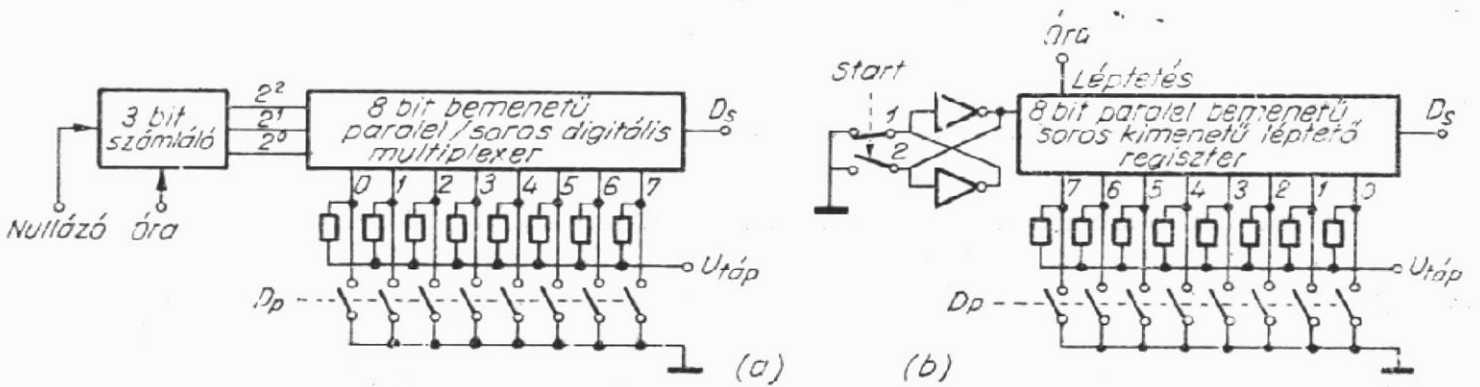
Az impulzusgenerátor minden programozható elemét külön-külön kell a csak neki szóló adattal feltölteni. Ezért valamennyi elemnek azonosíthatónak kell lennie, valamennyi címmel kell, hogy rendelkezzen. A digitális szerkezetű impulzusgenerátor belső szervezését az 5.18. ábra vázolja.



5.18. ábra. Digitális strukturájú, moduláris felépítésű programozott impulzusgenerátor

Programozható szógenerátorok

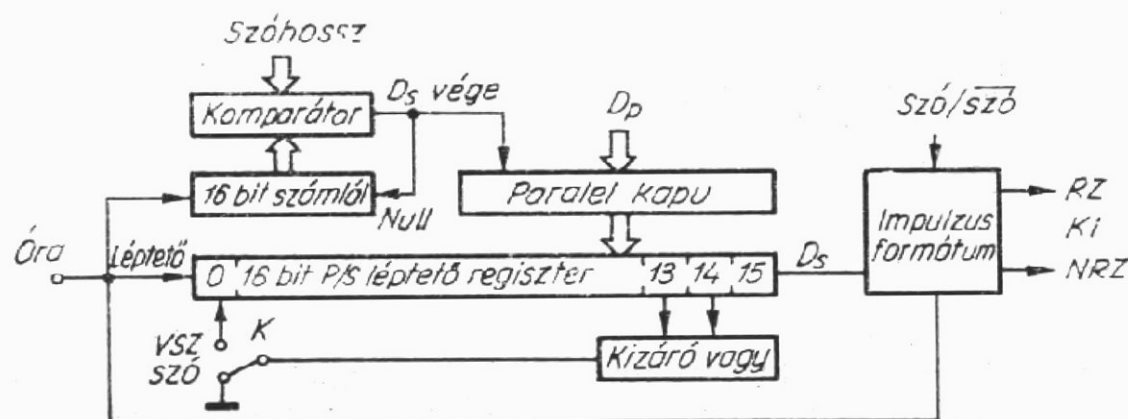
A szógenerátor N komponensű, soros vagy párhuzamos, bináris szavak képzésére szolgáló speciális impulzusgenerátor. A számítógép mellett elsődleges feladata, hogy a gép lehetőségeit meghaladó sebességgel állítson elő vizsgáló mintázatokat. Az 5.19. ábra soros szógenerátor két alapvető változatát mutatja be. A paralel/soros digitális multiplexer (a) ábra) vezérlő bemenetére számlálót kapcsolva, paralel adatbemenetére adatot véve, majd a nullázó felengedésével az órajeleket engedélyezve a D_p paralel szó komponensei a D_s soros szó komponenseiként egymás után a ki-



5.19. ábra. Multiplexer típusú (a) és léptetőregiszter-típusú (b) soros szógenerátor

menetre lépnek. A számlálót ciklikusan működtetve, a kimeneten a bemeneten rögzített szó ismétlődik, az órafrekvenciától és a szó hosszúságától függő gyakorisággal. A bemenő adat kapcsolókkal manuálisan vagy tárolóregiszterben gépi úton rögzíthető. Szükség esetén a bemenet gépi úton, ciklusonként átírható.

Hasonló jellegű szógenerátor hozható létre paralel bemenetű, soros kimenetű léptetőregiszter alkalmazásával (5.19b ábra) is. A startkapcsoló működtetésével a D_p szó (mely ismét csak kapcsolókkal manuálisan vagy gépi úton automatikusan definiálható) a léptetőregiszterbe tárolódik, majd az óragenerátor ritmusában, komponenseként, D_s soros szó alakjában a kimenetre kerül.



5.20. ábra.

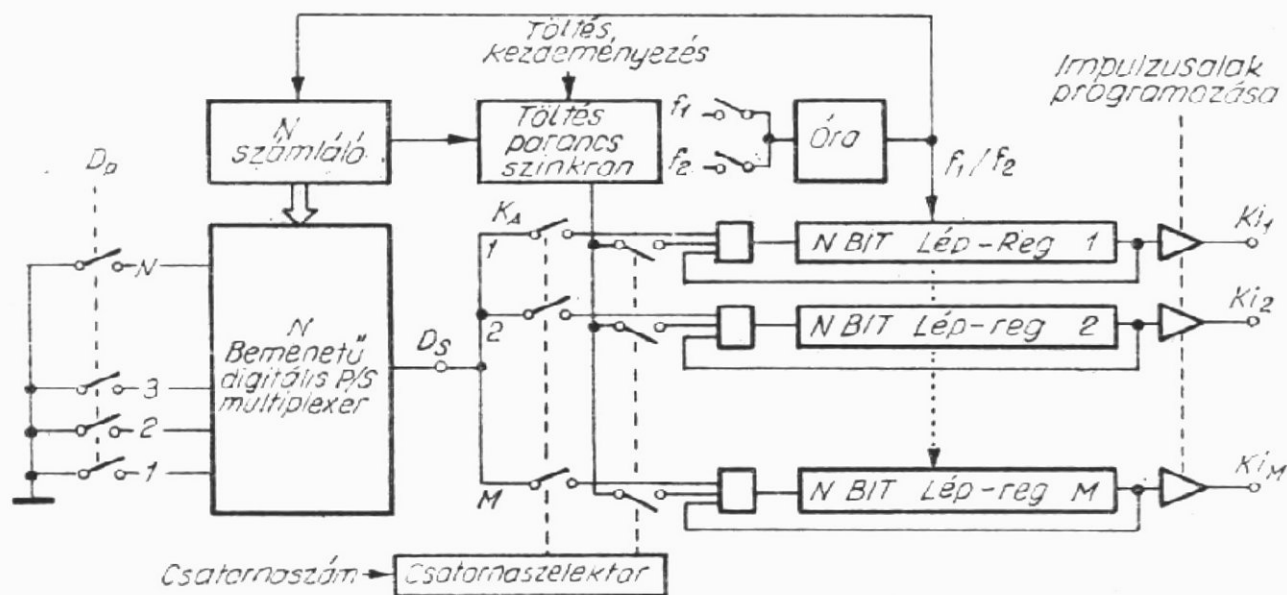
Léptetőregiszter-típusú, programozható szóhosszúságú soros szógenerátor

Az 5.20. ábra léptetőregiszteres szógenerátora az előbb ismertetett generátortól mindenekelőtt a programozható szóhosszúságban különbözik. A komparátorba a H szóhosszt programozva, a léptetőregiszter tartalma H óraimpulzus után újrafródik. A komparátor elhagyható, ha a szógenerátor működési ciklusának megkezdése előtt a szóhossz-számlálóba H komplemente kerül. A szógenerátor impulzusformátum áramkörének RZ (return to zero=alapvonalra visszatérő) és NRZ (non return to zero=alapvonalra nem visszatérő) impulzuskiemelete program útján komplementálható.

A K kapcsolót szóállásból VSZ állásba váltva, a szógenerátor végtelen ciklusban dolgozó álvéletlenszám-generátorra alakul át.

D_p bevitele mindig a komparátor vége jelével történik. Ha a paralel kaput pufferregiszter előzi meg, e pufferregiszteren az adat előkészítése célszerűen a munkaciklus elején történhet, annak érdekében, hogy a vége jel a paralel kapu bemenetén már konszolidált viszonyokat találjon.

Az 5. 21. ábra szógenerátora N számú, M bites paralel szóból álló digitális mintázat szolgáltatására, M darab dinamikus léptetőregisztert alkalmaz. A generátor két fő munkafázisa a töltés és a jelgenerálás. A töltés előkészítéseként a D_p bináris vektort a multiplexer bemenetére kell vinni, a csatorna szelektor közvetítésével K_A kapcsoló útján a feltöltendő léptetőregisztert ki kell választani, az óraoszillátort pedig f_1 üzemmódba kell állítani (mely üzemmódban $f_1 \ll f_2$ frekvenciájú léptetőjelet szolgáltat). A töltés külső kezdeményezés alapján megy végbe. A kezdeményezés hatására az egyébként passzív szinkron egység aktív állapotba kerül. Az aktív állapotú egység kész fogadni a számláló jelét, melynek hatására a kiválasztott léptetőregiszter bemenetére töltésvégrehajtás parancsot bocsát ki. A ciklikusan működő számláló a multiplexer 1. sz. bemeneti kapcsának kiválasztásával egyidejűleg bocsát ki jelet, így a betöltés mindig D_p első bitjével kezdődik, ami biztosítja a betöltött adatok bitparalel együttfutását. A töltésvégrehajtás parancs egészen a számláló soronkövetkező jeléig marad fenn, mely a szinkron egységet passzív állapotába állítja



5.21. ábra.
 M szélességű, N hosszúságú mintázatok generáló paralel szógenerátor

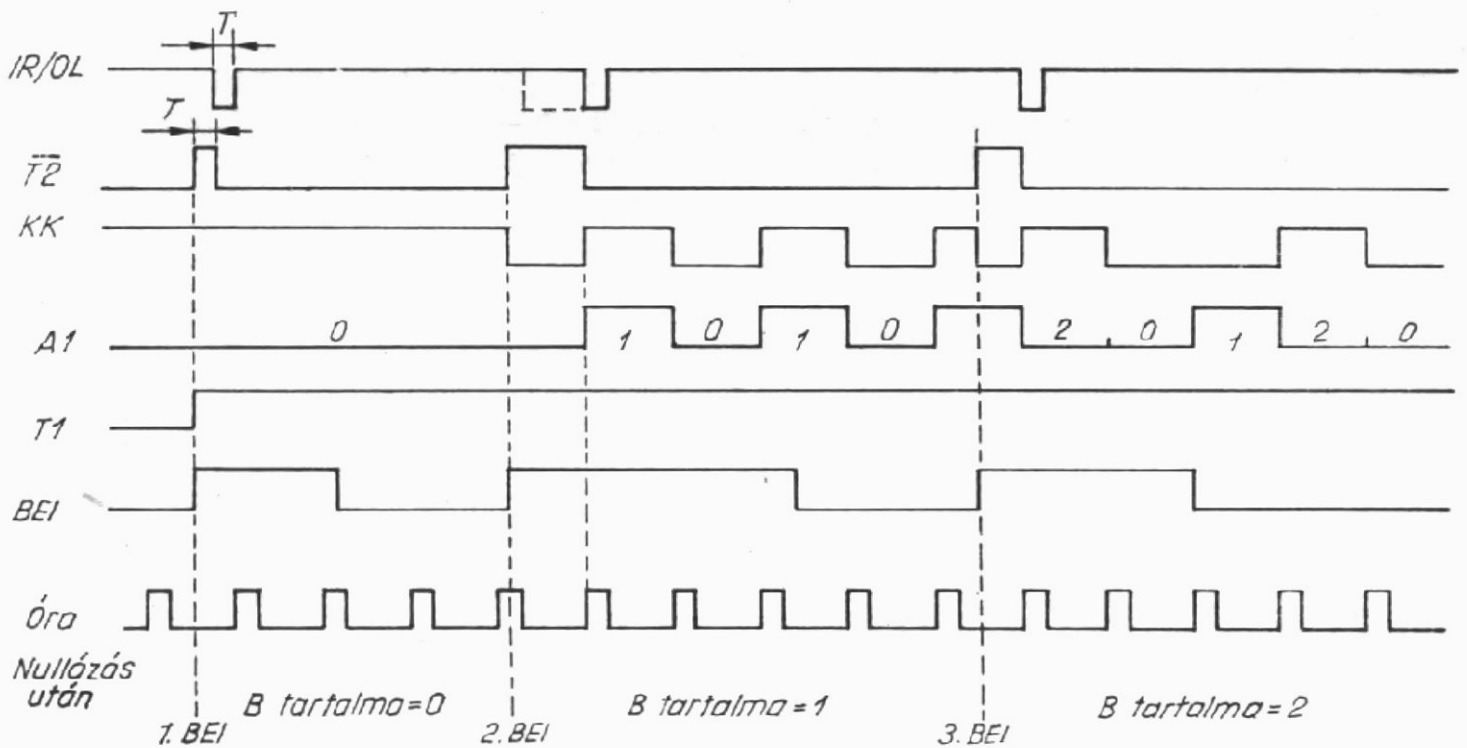
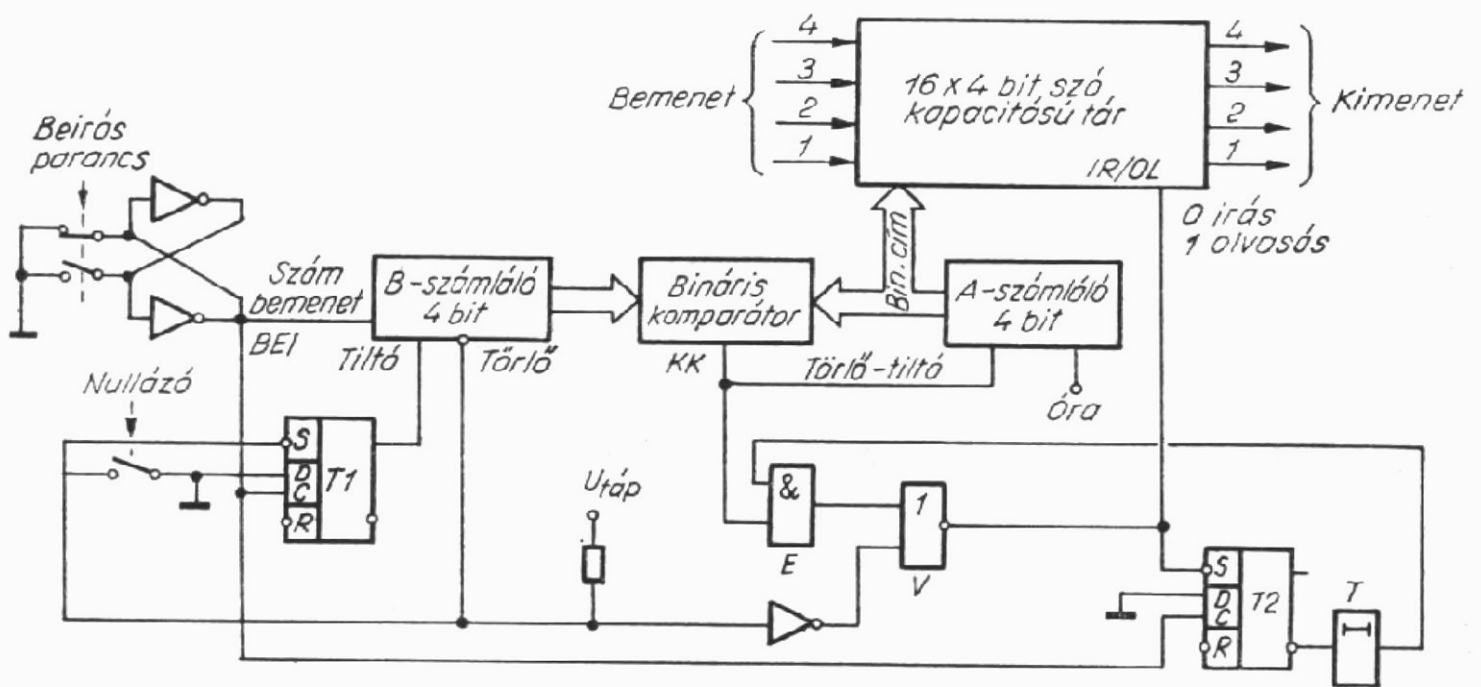
vissza. A töltésvégrehajtó jel időtartama alatt a recirkulációs hurok megszakítódik, de megszűnte után azonnal helyreáll. Így valamennyi léptetőregiszter tartalma az N számlálóval szinkron cirkulál. A jelgenerálás munkafázisa az M -edik regiszter feltöltése után, az órafrekvenciának f_2 -re állításával veszi kezdetét, és tetszés szerinti időn át tartható fenn. A szógenerátor feltöltése manuálisan, kapcsolók beállításával, ill. működtetésével (töltéskezdeményezés) vagy gépi úton történhet.

Az 5.22. ábrán bemutatott szógenerátor változtatható tartalmú tároló alkalmazásán alapul. A példaképpen választott 16×4 bit kapacitású memóriába lassú ritmusban 1...16 szó írható be; a beírt szavak kiolvasása az órafrekvenciának megfelelő (nagy) sebességgel ciklikusan történik. A szógenerátor a nullázóval hozható alapállapotba. Alapállapotban az A és B számláló nullára mutat; a számlálók megegyező tartalma folytán a komparátor 1 kimenetet ad, s tiltja az A számláló órabemenetét; T1 és T2 billenő 1 állapotban van, T1 tiltja a B számláló számbemenetét, T2 tiltja az E kaput.

A memória bemenetére helyezett első 4-bites szó beírása a BEI kapcsolóval kezdeményezhető. BEI hatására T1 és T2 átbillen 0 állapotba. B számláló helyzete egyenlőre változatlan marad (mivel a BEI jel felfutó élénél számbemenete még tiltva van), s így az A számláló is megőrzi alapállapotát, melyben a 0. című tárolórekeszre mutat. T2 átbillenését követő T idő múlva a memória ÍR/OL bemenetén 0 szint jelenik meg, ezzel megtörténik az első szó beírása. A beíró jel T ideig tart, mivel felfutó éle T2-t ismét 1 állapotba helyezi, s így önmagát szünteti meg.

Újabb BEI parancs a második beírandó szónak a memória bemenetére helyezése után adható ki. Hatására T2 ismét $1 \rightarrow 0$ átmenetet végez, kimenő jele azonban az E kapun elakad, mivel a most már szabad bemenetű B számlálóba 1 íródik, ami a bináris komparátor kimenetének $1 \rightarrow 0$ változását idézi elő. E változással az A számláló is felszabadul a tiltás alól; a következő óraimpulzust regisztrálja. Ezzel a komparátor egyensúlya, kimenetének 1 értéke helyreáll. Ez most már átengedi T2 jelét, következésképpen V kimenetén (a memória ÍR/OL pontján) T ideig tartó beíró impulzus jelentkezik, melynek hatására a memóriabemenet a második memóriaregiszterbe kerül.

E helyen kell kiemelni az A számláló törlő-tiltó bemenetének sajátosságát. E bemenet logikai nullszintjén a számláló regisztrálja az órajeleket. A törlő-tiltó bemenet $0 \rightarrow 1$ átmenete a számlálóra közvetlenül nem hat. A nullázás csupán a logikai egy szint megjelenését követő első óraimpulzus hatására következik be, ami az órabemenet egyidejű letiltásával jár. E tulajdonságból következik, hogy miután időközben a



5.22. ábra.
Tetszőleges elérési sorrendű mintázattárolót alkalmazó paralel szógenerátor

komparátor egyensúlya a beírási ciklusban helyreállt, a soron következő órajel-impulzus nullázza az A számlálót, s megtörténik az első memóriarekesz tartalmának kiolvasása: $\overline{IR/OL}=1$. A soron következő órajel az A számláló ismét regisztrálja, s megtörténik a második memóriarekesz tartalmának kiolvasása. Újabb beírás parancs érkezéig a rendszer ciklikusan olvassa az első és második tárolórekesz tartalmát. Általánosságban, i számú beírás esetén a ciklikus olvasás az első i rekeszre terjed ki. Az ábra alsó felén látható impulzusdiagram a belső állapotok alakulását szemlélteti, a nullázás után kiadott beírás parancsok sorszámának és az órajelnek a függvényében. A diagramon T1 és T2 a triggerok felhasznált kimeneteinek, A1 az A számláló 2^0 helyértékű kimenetének állapotalakulását mutatja be. Az A1

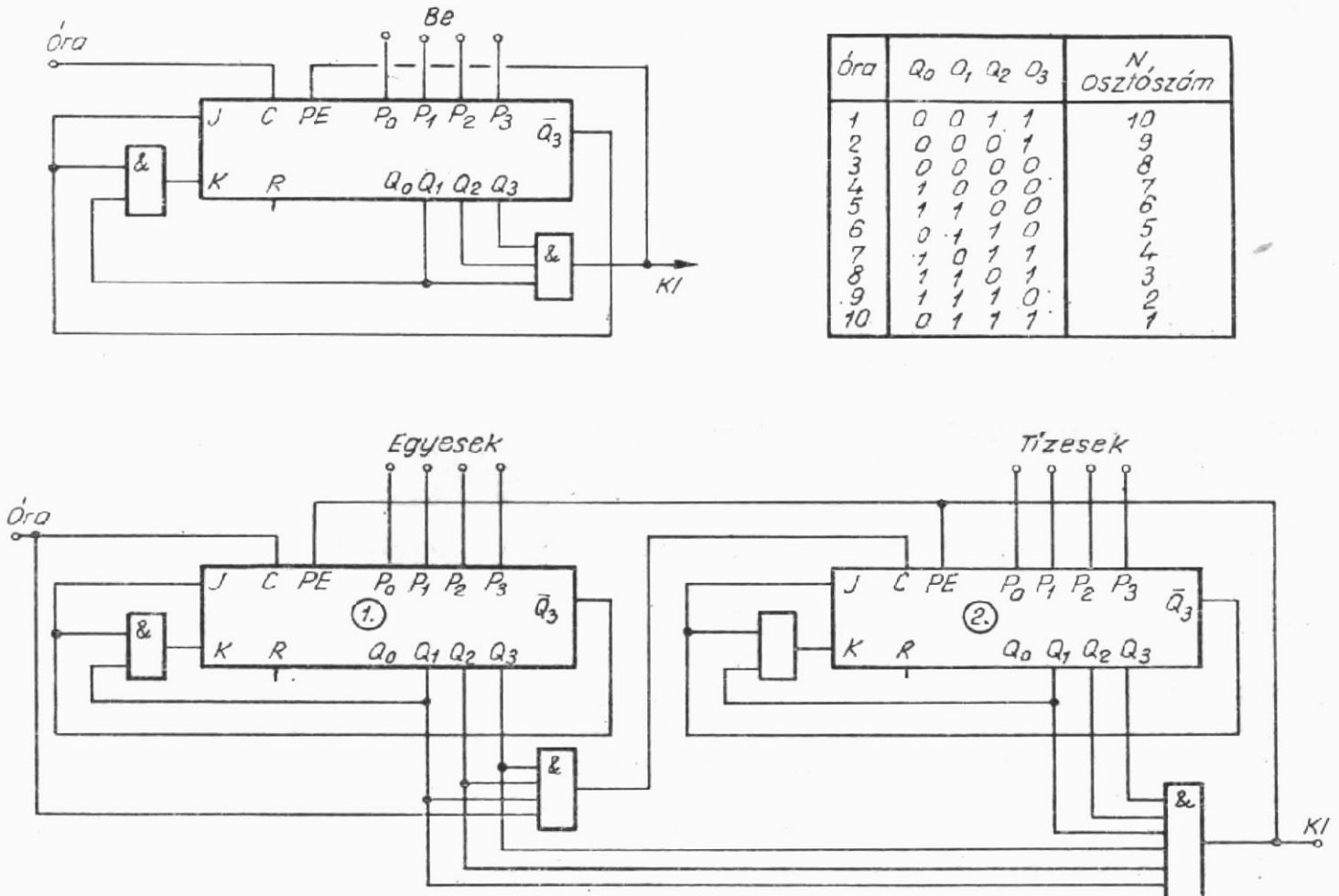
állapotgörbéjének egyes szakaszain szám tünteti fel az A számláló momentán tartalmát.

Szógenerátor fixtár alkalmazásával is képezhető. A tartalom rögzített, beírásról ilyenkor nem kell gondoskodni, az áramköri logikára egyedül az előírt módon végzett kiolvasás feladata hárul. Az a körülmény, hogy a tartalom programozás útján nem módosítható, csökkenti ugyan a flexibilitást, ugyanakkor azonban a fixtár használatával a generátor felépítése és működésmódja egyszerűsödik, működési sebessége pedig emelhető. A szógenerátorok gyakran moduláris szerkezetűek; változtatható a szóhossz és a mintázatban helyet foglaló szavak száma is.

Programozható frekvenciaosztók, -szintetizálók és -sokszorozók

A digitális impulzusgenerátorok ismertetése kapcsán, a programozható frekvenciaosztó egy megoldásával már találkoztunk: számláló kimenetét – másik bemenetén program útján előkészített – digitális komparátor figyeli; amikor a számláló tartalma eléri a komparálási szintet, a komparátor a számláló tartalmát törli. Ha a számlált jel frekvenciája f_1 , a komparátorbemeneten előkészített szám n , a komparátor – számláló együttes kimenő jelének frekvenciája $f_2 = f_1/n$.

A fentitől eltérő rendszerű osztókat is alkalmaznak. Az 5.23. ábra léptetőregiszteres, egy- és kétdekádos decimális osztót mutat be. A négybites léptetőregiszternek soros (J, K) és paralel ($P_0—P_3$) bemenete, soros (Q_3) és paralel ($Q_0—Q_3$) kimenete egyaránt van; jellemzője továbbá, hogy első billenőköre J—K bistabil, a többi D tároló. Paralel beírás PE jellel végezhető, mely beírásakor lezárja, egyébként szabadon hagyja

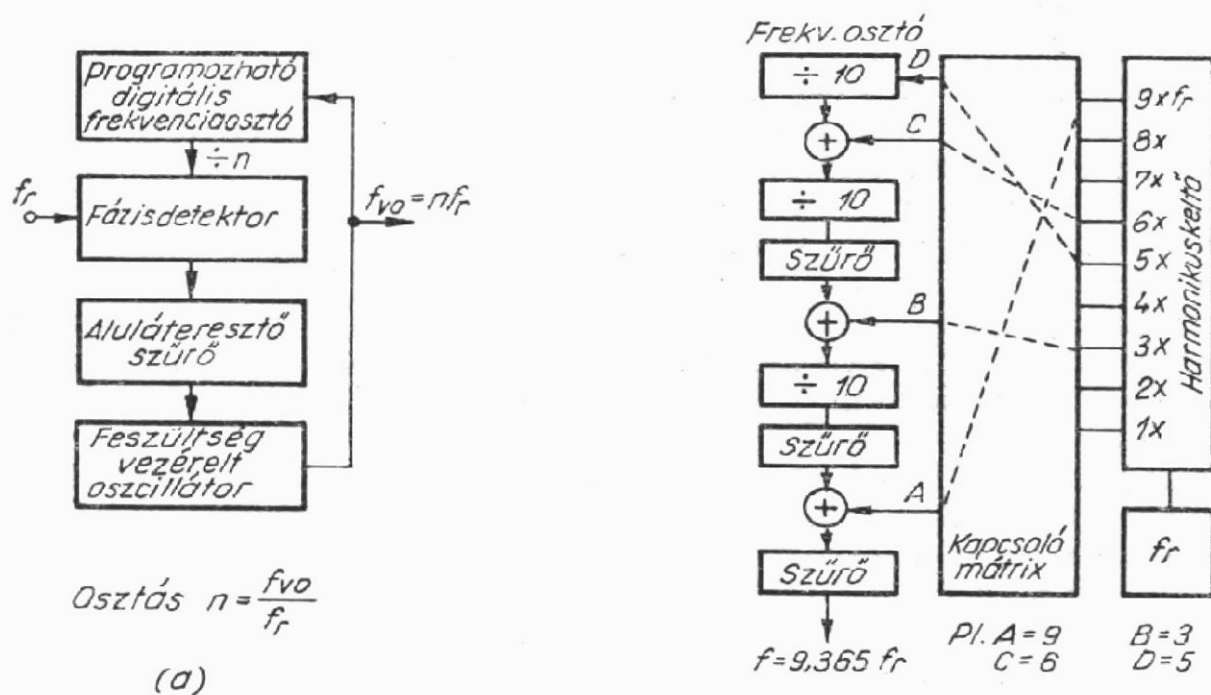


5.23. ábra.
Egy- és kétdekádos digitális frekvenciaosztó

a regiszter belső soros útvonalát. Ha a soros útvonal szabad, az órajel hatására minden egyes bistabil tartalma egy hellyel jobbra tolódik, az első bistabil új tartalmát pedig a J—K bemenet definiálja. Válasszuk a párhuzamos bemenet értékét 0011-nek. Az órajel hatására a bemenet a regiszterbe tárolódik. A soron következő órajelek a regiszter tartalmát a táblázatnak megfelelően alakítják. A 11. órajel hatására ismét a 0011 tartalom áll elő. A 0011 kezdő tartalommal tehát a regiszter tizes osztású. Kezdő tartalomnak a táblázat valamely más értékét választva, pl. 1011-et, a regiszter négyes osztóként működik. A helyes működés feltétele azonban, hogy a paralel bemeneten a választott kezdő érték az osztó működése során mindvégig fennmaradjon. A kétdekádos változatban az 1. regiszter többszörös számlálási ciklust hajt végre. Éspedig az első ciklusban a belétárolt kezdő értéknek megfelelő, a többi ciklusban teljes tizes számlálást végez. Ha pl. a leosztás 37, a regiszterek bemenetére 1000-et, ill. 1101-et kell adni. Az 1. regiszter az első ciklusban hetet számol, majd még háromszor tizedet. A harmadik tizes ciklus végével mindkét regiszter tartalma 0111. Ezután ismét a bemeneti adatok tárolódnak.

A programozható frekvenciaosztónak mint a gépi irányításra kézenfekvően alkalmas elemnek, fontos szerep jut a fáziszárás (phase-lock) elvén alapuló digitális frekvenciaszintetizálóban (5.24a ábra). Itt, feszültségvezérlésű oszcillátor f_{VO} frekvenciáját saját, frekvenciaosztóval leosztott jelének valamely f_r referenciajellel fázisdetektorban képzett hibajele, ill. e hibajellel arányos egyenfeszültség definiálja és stabilizálja. f_{VO} az n programozásával digitálisan változtatható. Megjegyezzük, hogy automatikus mérőrendszerekben való alkalmazás tekintetében, viszonylag lassú működésük miatt a fáziszáró rendszerek a közvetlen átalakítás elvén alapuló frekvenciaszintetizálók (5.24b ábra) mögött mérsékelt szerephez jutnak.

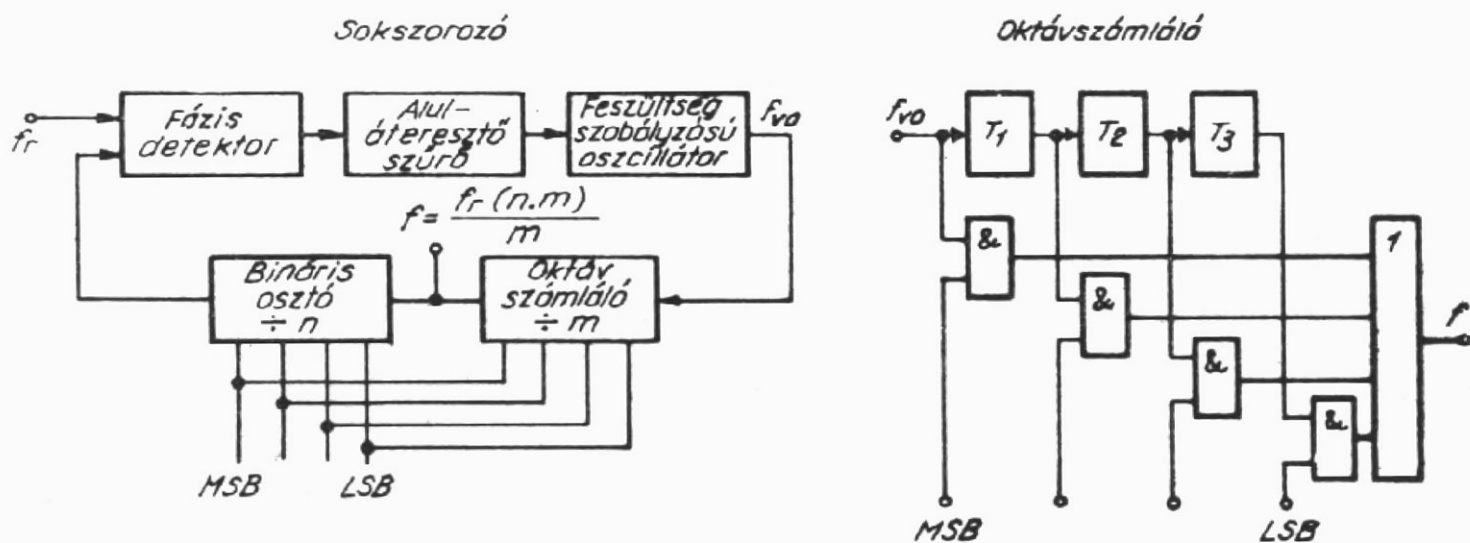
A programozható frekvenciaosztó alkalmazást talál az impulzusfrekvencia sokszorozása terén is. Az 5.25. ábra bináris frekvenciaosztó és oktáv számláló együttesének felhasználásával, fáziszárás elvén alapuló sokszorozót mutat be. Az f kimenőfrekvencia itt is a (bináris) frekvenciaosztó beállításának függvénye, azonban az oktáv számláló jelenléte miatt általában $f \neq f_{VO}$. Az oktáv számláló mint frekvenciaosztó szerepe az, hogy n változásával a vezérelt oszcillátor frekvenciája ne változzék lényegesen. A bináris osztó és az oktáv számláló együttesen mintegy frekvencia-



5.24. ábra.

Frekvenciaszintetizáló

(a) fáziszárás elvén alapuló — digitális frekvenciaosztóval; (b) közvetlen átalakítás elvén alapuló



5.25. ábra.

Fáziszárás elvén alapuló, bináris frekvenciaosztót és oktáv számlálót alkalmazó frekvenciasokszorozó

potenciométert képez. f_{VO} -nak előírt határok között tartása a szabályozó kör stabilitás és válasz tulajdonságai szempontjából előnyös; az áramkör lényegesen gyorsabban követi az osztón át adott parancsokat, mint az 5.24a ábrán bemutatott rendszer.

Digitális jelszintetizálók

A digitális áramköri technika lehetőségeinek, valamint D—A átalakítónak a felhasználásával, a legkülönbözőbb alakú periodikus jelek generálhatók. A bináris programozású erősítő (mint szorzó D—A átalakító) digitális bemenetére egyszerű számláló paralel kimenetét kapcsolva, jelműnyetere rögzített egyenfeszültséget helyezve, majd a számlálóval óragenerátor jeleit akumulálva, fűrészfog jel generátor áll elő (5.26a ábra), mely a fűrészfog lineárisan emelkedő szakaszát a digitális felbontás mértékének megfelelő pontossággal, lépcsős görbével közelíti meg. A jel periódusidejét a számláló kapacitása és az f_c órafrekvencia definiálja; az amplitúdó U_b -vel arányos.

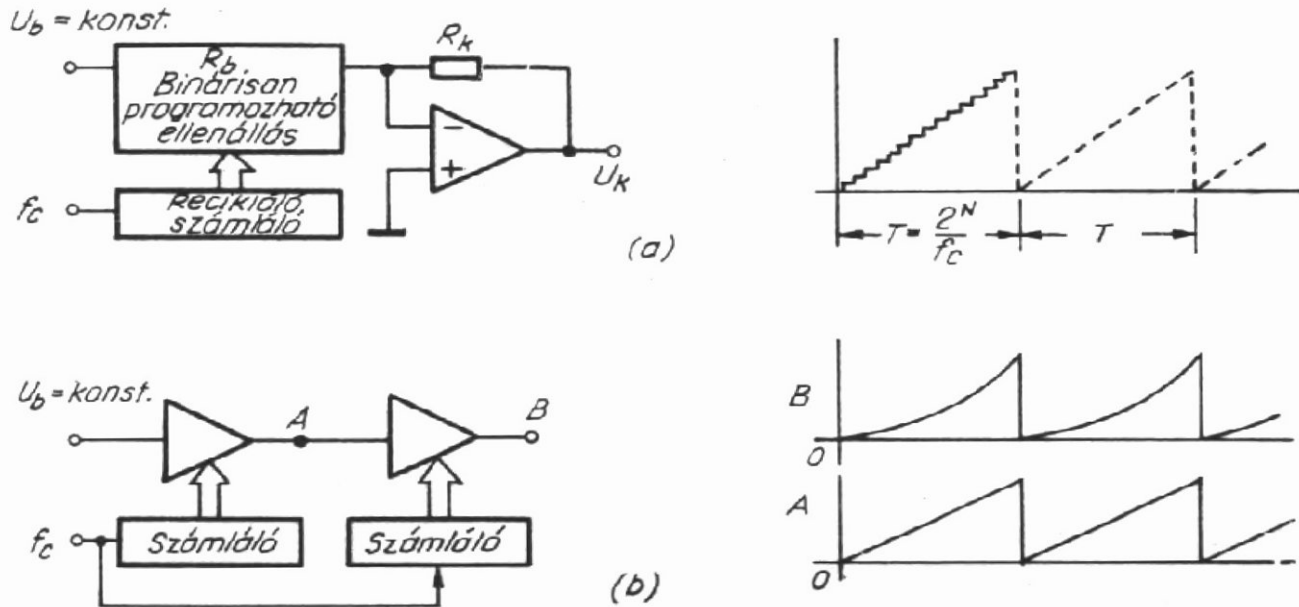
Egyszerű számláló helyett reverzálható számlálót alkalmazva, és e számlálót minden számlálási ciklus végén reverzálva, háromszögjel-generátor áll elő.

Egyes mérések kapcsán szükség lehet időben lineárisan növekvő amplitúdójú szinuszciklus vagy impulzussorozat előállítására. Az U_b bemenetre egyenfeszültség helyett szinuszciklus, ill. impulzussorozat állandó amplitúdójú jelét bocsátva, a kimeneten a kívánságnak megfelelő alakú jelsorozat áll elő.

Bonyolultabb függvénygörbék valósíthatók meg több, az iméntihez hasonló jelgenerátor kaskád kapcsolásával (5.26b ábra). Ha pl. a fűrészfog jelkeltő U_b analóg bemenetere egy vele azonos fűrészfog jelkeltő analóg kimenetét kapcsolják, másodfokú parabolaszakaszokból álló periodikus jel jön létre. Több lineáris jelkeltő egymás után kapcsolásával magasabbfokú görbék is előállíthatók.

A jelszintetizálás lehetőségei D—A átalakító és tároló együttesének alkalmazásával lényegesen kiterjeszthetők (5.27a ábra). A periodikus függvénygörbe időtengelyét, az időtengely menti diszkrét abszcisszapontokat, az óragenerátor és a számláló cím-regiszter együtt képezi; az ordinátaértékek az egyes abszcisszákhöz tartozó tároló-címek alatt helyezkednek el. E sémából kiindulva szinuszciklus szintetizálására szolgáló jelgenerátorokat is létrehoztak. Szinuszciklus szintetizálása kapcsán (5.27b ábra), már tárolóterület megtakarítása érdekében is, figyelembe veszik a szinuszciklus négy negyedének szimmetriatulajdonságait, s csupán a görbe első negyedének érték-készletét tárolják. E kvadráns tároló címek egy teljes szinuszciklus képzése során

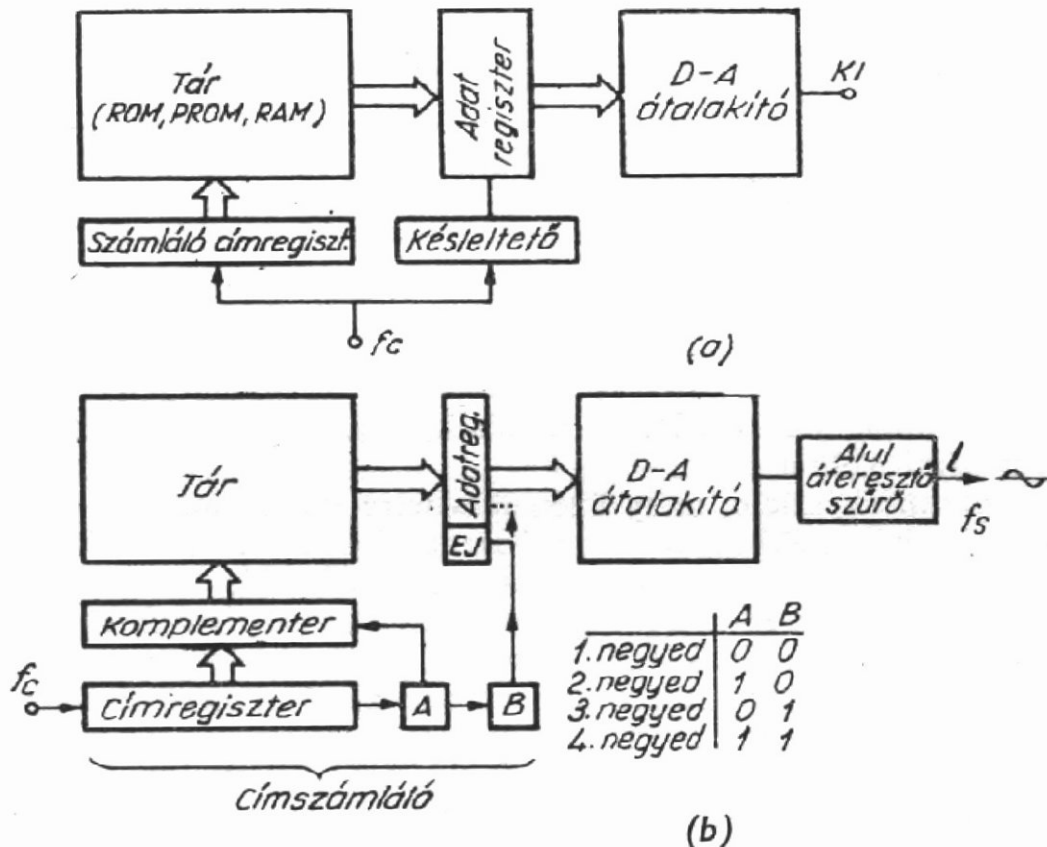
a címszámláló négyszer halad át. Éspedig az első és harmadik negyed képzése során a címek növekvő, a második és negyedik negyed képzése során a címek csökkenő sorrendjében. π és 2π között az ordináta értékét negatív előjellel is el kell látni. A címszámláló két legmagasabb helyértékű bitje ez esetben az aktuális görbenedre mutat, s csak a megmaradó rész szolgál továbbra is a tároló címregisztereként. Az A bit a 2. és 4. negyedben komplementálja a címregiszter kimenetét, a B bit a 3. és 4. negyedben az előjelet negatívra váltja, s a D—A átalakító bemeneti kódjától függően esetleg a tárból érkező adatot is komplementálja.



5.26. ábra.

Jelalak-generátorok

(a) fűrészfog lineárisan emelkedő ágának közelítése lépcsős görbével; (b) parabolajel előállítás



5.27. ábra.

Jelalak-szintetizátor

(a) tetszőleges alakú periodikus függvény előállítására; (b) szinuszjel szintetizálására

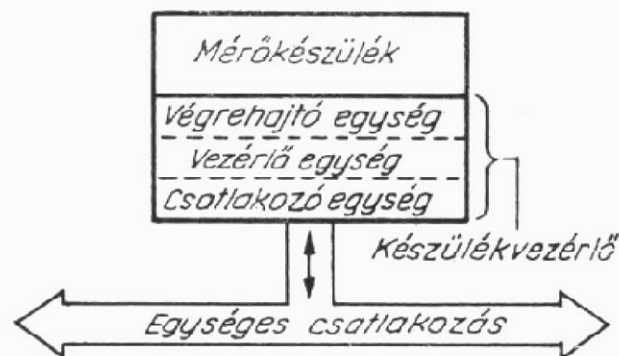
A képzett szinuszjel frekvenciája $f_s = f_c/N$, ahol f_c az órafrekvencia, N pedig a címszámláló teljes kapacitása: $N = 2^{M+2}$; M a kvadráns címregiszter biteinek száma, melyhez A és B-re tekintettel még +2 járul.

A jelalak-szintetizátort változtatható tartalmú (RAM) tárral ellátva, a szintetizátor jelalakja esetenként programozható.

5.3. A mérőkészülék illesztése a rendszerhez

A készülékvezérlő

A mérőkészüléket (mérőkészülékként kezelhető mérőhálózati egységet), mint a számítógép esetében a periferikus készüléket, készülékvezérlő illeszti a mérőrendszerhez. Amint korábban (1. 4. fejezet) már említettük, a készülékvezérlő három — feladatát tekintve önálló — részre különül el, a csatlakozó-, a vezérlő- és a végrehajtó egységre (5.28. ábra).



5.28. ábra.

A készülékvezérlő elhelyezkedése és felépítése

A csatlakozóegység a mérőkészüléknek a rendszerhez való villamos—mechanikus csatlakoztatásán túl a két fél közötti kommunikáció lebonyolítására szolgál. A csatlakozás funkcionális specifikációja, az utóbbival foglalkozva, definiálja a mérőkészülék és a rendszer közötti üzenetváltás módját; nem foglalkozik azonban a csatlakozáson áthaladó üzenet tartalmával. A funkcionális specifikáció leglényegesebb elemei:

- annak leírása, ahogy a rendszer a készülékkel kapcsolatot kezdeményezhet, ahogy, s ahányféleképpen kapcsolatot létesíthet;
- annak leírása, ahogy a készülék a rendszerrel kapcsolatot kezdeményezhet (ahogy a jelentkező készülék identifikációja történik, s ahogy és ahányféleképpen a kapcsolat megvalósulhat);
- a csatlakozórendszer parancskészlete, mely üzenetátvitelre vonatkozó (pl. üzenet átvitele a rendszertől a készülékhez) és adminisztratív jellegű (pl. a csatlakozó egység alapállapotának helyreállítása stb.) parancsokat ölel fel.

A specifikáció a csatlakozási rendszer lehetőségeinek kerete, mely egyúttal definiálja annak módját is, ahogy a lehetőségek realizálhatók. A körülhatárolt lehetőségek keretén kívül és a definícióktól eltérő módon a mérőrendszer és a mérőkészülék nem kommunikálhat egymással. Másrészt adott mérőkészülék illesztése során általában az összes lehetőség kihasználására nincs szükség; elég azokkal élni, melyeket a konkrét helyzet megkövetel. Tovább menve, a csatlakozóegység sem kell, hogy tartalmazza azokat az áramköri elemeket, melyek általa fel nem használt csatlakozási lehetőségek kielégítését szolgálnák. Ha pl. csak egyirányú üzenetátvitel szükséges, a másik irányú átvitelhez tartozó elemek elmaradhatnak.

A *vezérlőegység* a mérőkészülék és a mérőrendszer között az üzenet szintjén tart kapcsolatot; funkciója esetenként nehezen körülhatárolható, mivel a ráháruló feladatok repertoárja általános rendszerszemponatok alapján szűkülhet vagy bővíthet meghatározott mérőkészülék esetében is. Általánosságban a vezérlőre hárul az üzenet rendeltetés szerinti értelmezése, a rendszer és a mérőkészülék esetleg eltérő üzenetformátumának egyeztetése, az üzenet átkódolása, bizonyos mérőkészülék feladatok előírt sorrendben, előírt időprogram szerinti lebonyolítása, a mérőkészülék, ill. a csatlakozóegység állapotának figyelése, reagálás a megfigyelt állapotokra, stb. Azonban, hogy adott esetben vannak-e ilyen feladatok, s ha vannak, azok a vezérlőre hárulnak-e, az már a konkrét adottságoktól és megfontolásoktól függ. Amennyiben pl. a rendszer és a mérőkészülék üzenetformátuma és üzeneteik kódolása eleve megegyezik, az ezzel kapcsolatos feladatok elesnek. Másrészt formátum és kód különbözősége esetén a transzformációt a számítógép maga is végezheti, miáltal e feladatok alól a vezérlő ugyancsak mentesül. Igaz, az üzenet formátumát sokkal inkább a csatlakozási rendszer átvivő vonalainak szélessége determinálja, semmint a gép vagy a mérőkészülék.

Ugyancsak a környezeti feltételektől függ, hogy milyen feladatok hárulnak a vezérlőre az üzenet rendeltetésének értelmezése tekintetében. Az üzenet — értelmét tekintve — lehet a mérőkészüléknek szóló, ill. attól származó adat, lehet a vezérlőnek szóló parancs, vagy lehet állapotvektor. Ezeknek megkülönböztetése többféleképpen történhet. Az első lehetőség: a különféle fajta üzenetek különböző átviteli csatornákon való továbbítása. A második megoldás: egyetlen átviteli csatorna, de egy készülékvezérlőn belül egynél több, egymástól független, külön-külön címezhető átviteli végpont (adatregiszter) alkalmazása. A harmadik lehetőség: az üzenetelemek blokkba kapcsolása, miközben az üzenetelem értelmét a blokkon belül elfoglalt pozíció szabja meg. A CAMAC rendszerben a vezérlőnek szóló parancsok is a csatlakozási rendszer feladat vonalán (F) továbbíthatók; a CAMAC előírásban megadott nemstandard parancsok a vezérlőnek szóló parancsként használhatók. Az adatátvitelre — ide és oda — külön vonalak (W és R) szolgálnak. Ugyanakkor az igen kiterjedt címzési lehetőségek módot adnak arra, hogy az adatvonalakon átvitt üzenetet címével is minősíteni lehessen. A vezetékiszámban takarékos IEC csatlakozás nemcsak, hogy a készülék (és vezérlő) vonatkozású üzeneteket szállítja egyetlen csatornán, de ugyanezt a csatornát a csatlakozási rendszer parancsainak átvitelére is felhasználja. Itt tehát az üzenet azonosításának csak második és harmadik lehetősége áll rendelkezésre.

A vezérlőt az említetteken túl bonyolultabb feladatok is terhelhetik. Puffertárolást és sebességtranszformációt, karakterisztika-linearizálást, sőt előzetes adatfeldolgozást is végezhet. A mögöttünk álló évtizedre tekintve a vezérlőegységgel kapcsolatos szempontok alapvetően módosultak. Kezdetben, a szerkezettakarékosság nevében, minden vezérlési feladatot a számítógépekre igyekeztek áthárítani (programorientáció). Jelenleg minden lehetséges feladatot, s nem csak a szorosabb értelemben vett vezérlési, hanem korábban tisztán számítógép-jellegűnek tekintett feladatokat is, a vezérlőre bíznak (elosztott intelligencia, mikroprocesszorok).

A *végrehajtó egység* a vezérlőegységet, vagy ha ez hiányzik (ami előfordulhat), a csatlakozóegységet közvetlenül illeszti a mérőkészülékhez. A végrehajtó egység tulajdonképpen nem összefüggő egész, hanem az eseti szükséglet által megkövetelt illesztőelemek (D—A és A—D átalakítók, logikai szintváltók, teljesítménypufferok stb.) együttese.

Megjegyezzük, hogy a csatlakozó-, a vezérlő- és a végrehajtó egység elkülönítését elvi alapon végeztük; a gyakorlatban a három rész gyakran elválaszthatatlanul ötvöződik.

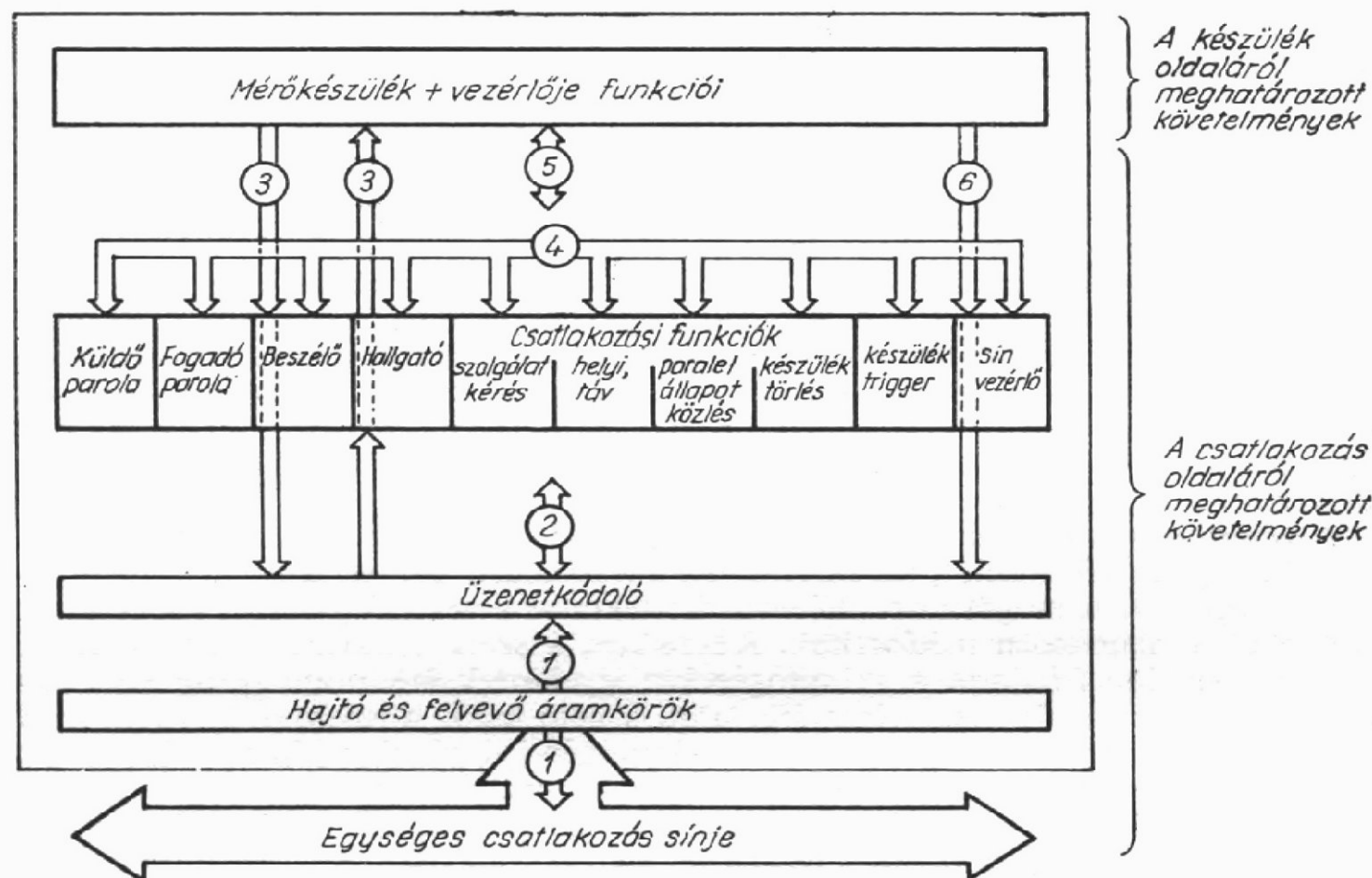
Vizsgáljuk meg a készülék illesztésének, a készülékvezérlőnek a problematikáját valamivel konkrétabb formában, az IEC csatlakozás figyelembevételével.

Az IEC csatlakozóegység

Az 5.29. ábra az IEC csatlakozóegységet legáltalánosabb alakjában ábrázolja. Az egység a közös sínre a csatlakozási specifikációnak megfelelő hajtó- és felvevő elemekkel csatlakozik, melyek a sín és az egység áramköreit villamos szempontból illesztik. Az üzenatkódoló az üzenetek előszelekcióját végzi el, a csatlakozási rendszer ATN vezetékének (l. 4.13. ábra) rendszerállapotot definiáló jele alapján.

Az összes lehetőségek figyelembevételével a csatlakozás tíz különböző funkciót láthat el, melyek közül öt alapfunkció, öt pedig kiegészítő funkció (5.1. táblázat). Alapfunkciónak a leggyakrabban előfordulókat nevezzük. A mellékfunkciók közül a szolgálatkérést mindig a beszélő funkcióval kapcsolva kell használni, mivel a készüléknek a lekérdezés során identifikálnia kell önmagát.

A lehetséges funkciók közül a csatlakozóegységhez egyedül azokat rendelik hozzá, amelyek a mérőkészülék kezeléséhez szükségesek, s hogy melyek szükségesek, azt a mérőkészülék tulajdonságainak és a mérőrendszerben betöltött szerepének elemzése alapján határozzák meg. Dönteni kell továbbá a választott funkciókon belüli lehetőségekről is. Funkciók és lehetőségek bármely készlete kompatibilis a csatlakozási rendszerrel. A csatlakozó egység felépítésmódját illetően, a mérőkészülék oldaláról meghatározott követelményeket a készülékcsatlakozó vezérlőjének struktúrája is befolyásolja.



1. Az egységes csatlakozás sínjének jelvezetékei
2. Kétirányú, csatlakozási funkcióra vonatkozó közlemények
3. Kétirányú készülékvonatközös üzenetek
4. A csatlakozási funkciók közötti állapot-kapcsolatok
5. Csatlakozó egység és készülék közötti kommunikáció
6. Sínvezérlő minőségben készülék által kezdeményezett csatlakozási közlemények

5.29. ábra.

A HP—IEC csatlakozóegység általános funkcionális sémája

Az IEC csatlakozás egy alkalmazása: DVM illesztése

A digitális voltmérő mint vevőtípusú mérőkészülék, mérési adatokat szolgáltat, s mint programozható készülék programozási adatokat igényel. Ezért mind beszélő, mind hallgató szerepkörben dolgozik. Csatlakozóegysége a következő feladatok ellátására kell, hogy alkalmas legyen:

1. Választási lehetőséget kell engednie a DVM helyi/táv üzemmódjai között.
2. A DVM rendszervezérelhető állapotában el kell juttatnia a készülékprogram tartalmú üzeneteket a sínről, a készülék beavatkozóinak digitális bemeneteire.
3. Fogadnia és továbbítania kell a DVM-nek szóló „olvasás” parancs jelentésű üzenetet (ez esetleg általános csatlakozási paranccsal helyettesíthető).
4. Fogadnia kell a DVM által szolgáltatott mérési adatot, valamint állapot adatot, és azokat a sínre kell továbbítania.

A 2...4. feladatokat a készülékvezérlőnek cím szerinti kiválasztás alapján kell elvégeznie. E feladatokon túl még rendelkeznie kell a következő képességekkel is:

5. Központi vezérlőt (számítógépet) nem tartalmazó rendszerben, pl. nyomtatóval együtt működtetve, címzés nélkül is beszélő lehessen.

5.1. táblázat

Alapfunkciók		Kiegészítő funkciók	
Beszélő	<i>Alap beszélő</i> — A készülék adatot küldhet egy másik készülékhez	A mérőkészülék azon képessége, hogy valamely esemény bekövetkeztét jelezze, s hogy a folyamatban levő sínműveletekkel aszinkron eljárás lebonyolítását igényelje	Szolgáltatkozás
	<i>Csak beszélő</i> — A készülék vezérlő nélküli rendszerben beszélőként működhet		
	<i>Saját magának nem beszélhet (MLA)</i> — Beszélő/hallgató funkciójú készülék önmagának nem beszélhet	<i>Alap helyi/táv</i> — Annak lehetősége, hogy a mérőkészülék helyi/táv programozása között lehessen választani	Helyi/táv
<i>Bővített beszélő (TE)</i> — Beszélő funkció kiterjesztett címzési lehetőségekkel	<i>Helyi kizárása</i> — A „visszatérés helyire” lehetőség hatástalanítása		
Hallgató	<i>Soros lekérdezés</i> — A vezérlőhöz küldött állapot-byte-tal a készülék saját szolgáltatkozását azonosítja	<i>Alap</i> — A készülék egy bittel jelezheti a vezérlőnek állapotát. Nyolc készülék egyidejűleg adhat választ. Egy állapotvonalon egynél több készülék ÉS/VAGY választ adhat	Paralel állapotközlés
	<i>Alap hallgató</i> — A készülék egy másik készülék adatát fogadhatja	<i>Átrendező</i> — A vezérlőnek arra való lehetősége, hogy a készülékeket „átrendezze”	
Küldő parola	<i>Csak hallgató</i> — A készülék vezérlő nélküli rendszerben hallgatóként működhet	<i>Alap</i> — A készülék képessége, hogy valamely alapállapotba lehessen visszaállítani. A készülékek visszaállítása paralel történik	Készülék törlés
	<i>Saját magára nem hallgathat (MTA)</i> — Beszélő/hallgató készülék önmagára nem hallgathat	<i>Szelektív</i> — Egyes készülékek szelektív visszaállítása	
	<i>Bővített hallgató (LE)</i> — Hallgató funkció kiterjesztett címzési lehetőségekkel		
	Byte-átvitel lebonyolításakor a küldő által generált időzítő-koordináló jel		

Alapfunkciók		Kiegészítő funkciók	
Fogadó parola	Byte-átvitel lebonyolításakor a fogadó által generált időzítő-koordináló jel		
Vezérlő	<p><i>Rendszervezélő</i> — Lehetővé teszi, hogy a készülék törlő (IFC) és helyi/táv (REN) sínparancsot adjon</p> <p><i>Visszaállítás (IFC)</i> — Megengedi, hogy a sín valamely egysége a rendszervezélő szerepkörét másik vezérlőtől átvegye és törölje a sint</p> <p><i>Távengedélyezés (REN)</i> — Valamely rendszervezélőnek lehetővé teszi helyi/táv parancs kiadását</p> <p><i>Válasz szolgálatkérésre (SRQ)</i> — A vezérlőnek lehetővé teszi szolgálatkérés kezelését</p> <p><i>Csatlakozási üzenet kibocsátása</i> — A vezérlő byte alakú parancsokat bocsáthat ki</p> <p><i>Vezérlés átvétele</i> — A vezérlő egy másik vezérlőtől átveheti a sín irányítását</p> <p><i>Vezérlésátadás</i> — A vezérlő egy másik vezérlőnek átadhatja a sín irányítását</p> <p><i>Paralel végigkérdezés</i> — A vezérlő paralel lekérdezést hajthat végre</p> <p><i>Szinkron vezérlésátvétel</i> — A vezérlő átveheti a sín irányítását anélkül, hogy már folyamatban levő adatátadását megzavarná</p>	Lehetőség egy vagy több készülék valamely műveletének egyidejű indítására	Trigger

6. Szolgálatkéréssel jelentkezhessék, ha a DVM számára kiadott program nem értelmezhető, ill. minden a DVM által elvégzett mérés végén.

Ahhoz, hogy a csatlakozóegység e feladatokat elláthassa, az alábbi funkciókkal, funkciókon belüli lehetőségekkel, ill. teljesítésükhöz szükséges áramköri elemekkel kell, hogy rendelkezzen:

- Helyi/táv üzemmódválasztó (az 1. feladat ellátására).
- Fogadó parolaelem a csatlakozási rendszer parancsainak vételére (szükséges a 2., 3. feladat ellátásához).
- Hallgató funkcionális eleme (a 2. és 3. feladat ellátására). Tekintve, hogy a DVM hallgató-beszélő jellegű készülék, a hallgató funkciót „saját magadra ne hallgass” lehetőséggel kell megvalósítani.
- Beszélő funkció (a 4. feladat megvalósításához, valamint a szolgálatkérés azonosítási műveletében való részvételhez).
- Az 5. feladat ellátása a csak-beszélő lehetőség mint alternatíva révén biztosítható.
- A szolgálatkérés funkcióelemei (a 6. feladat ellátásához).

5.4. A mérőhálózat lehetőségeinek kiterjesztése

Intelligencia színvonalának emelésével a hálózat alkalmassá tehető bizonyos számítógép-feladatok átvállalására. A hálózati intelligencia fokozásával másrészt egy sor olyan méréstechnikai probléma is megoldhatóvá válik, aminek útja egyébként — a számítógép jelenléte ellenére is — elzárva maradna. Az intelligencia-színvonal emelésének módszerei közül, az alábbiakban

1. az automatikus tartományváltás,
2. a memória bővítése,
3. a mikroprogramozott vezérlés alkalmazása,
4. a mikroprocesszor alkalmazása

kérdéseire térünk ki. Mint a bemutatott gyakorlati megoldásokból is kitűnik, az intelligencia problematikája mindig a mérőhálózat (mérőkészülék) vezérlési szférájához kapcsolódik.

Automatikus tartományváltás

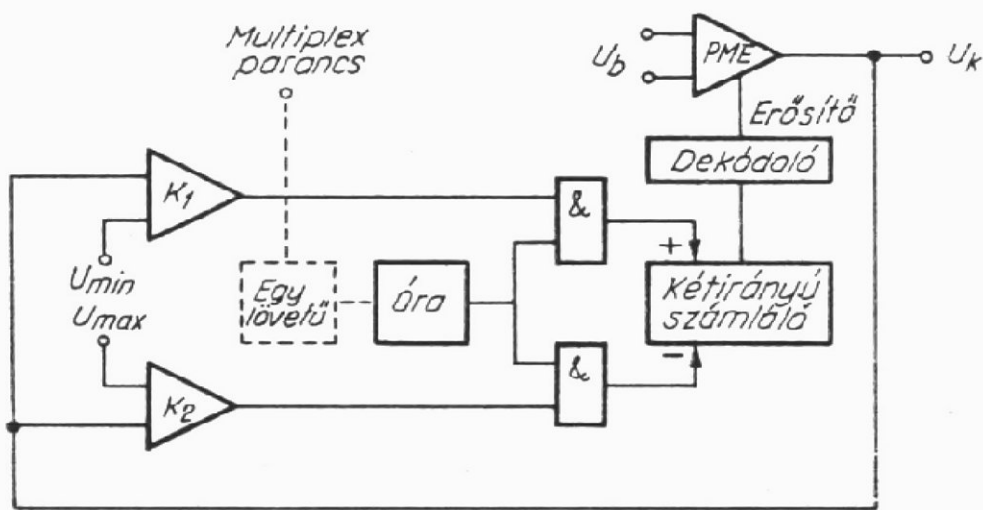
A számítógép tehermentesítésének egyik lehetőségeként, a mérőhálózatban automatikus méréstartomány-beállítással rendelkező mérőkészülékek alkalmazhatók. A mérőkészüléknek az automatikus beállítást végző szerve különböző eljárasmódokat követhet:

1. A legalacsonyabb méréstartománnyal kezd, majd szükség szerint magasabb mérési tartományokra tér át.
2. A legmagasabb tartománnyal kezd, és a mérési eredményektől függően addig lépked alacsonyabb tartományokra, mígnem a mért mennyiség értékéhez leginkább illeszkedő tartományt el nem éri.
3. A valószínűleg legalkalmasabb tartománnyal kezd, s szükség szerint innen lép felfelé vagy lefelé.

Mindig a mérőberendezésen belül dől el, vajon az éppen érvényben levő méréstartomány illeszkedik-e a bemenő értékhez. A méréstartomány kiválasztása e döntés alapján történik, a fenti eljárasmódok valamelyikével.

Önbeállító erősítők. Olyan környezetben, ahol a jelamplitúdó értéktartománya előre nem ismert, alkalmazzák az önbeállító erősítőket, melyek saját kimenő feszültségüket automatikusan, előírt határok között tartják. Lényeges idő és programozási munka takarítható meg, ha az erősítéstartományt nem a számítógépnek kell a jelfeszültséghez illeszteni. Az önbeállító erősítő programozó áramköre a momentán erősítés értékét a számítógép számára folyamatosan szolgáltatja.

Az önbeállítás lehetséges többféle módja közül egyet az 5.30. ábra mutat be. A PME programozható mérőerősítő tartományváltását kétirányú számláló vezérli. A számláló tartalmának növekedtével az erősítés nő, csökkenésével csökken. Feltétel szerint a kimenő feszültségnek, melyet K_1 , K_2 komparátor ellenőriz, az U_{\max} — U_{\min} feszültséghatárok között kell maradnia. $U_k > U_{\max}$ esetén K_2 komparátor $0 \rightarrow 1$ átmenetet végez, és az óragenerátor jelét a számláló levonó bemenetére (—) bocsátja. A számláló tartalmának csökkenésével csökken az erősítés is (pl. az előbbi érték tizedére). $U_k < U_{\min}$ esetben K_1 végez $0 \rightarrow 1$ átmenetet, az erősítést növelve. Az erősítést a rendszer addig változtatja, míg az $U_{\min} < U_k < U_{\max}$ előírás nem teljesül. Ilyenkor mindkét komparátor kimenetén 0 található, a számláló tartalma — az erősítés — állandó. Az óragenerátor periódusidejének illeszkednie kell az erősítő beállási és túlterhelés visszaállási idejéhez annak érdekében, hogy a soron következő óraimpulzus idejére a komparálás reális eredményt adjon. Két méréshatár közötti billegés elkerülése érdekében az U_{\min} , U_{\max} egymáshoz való viszonyát az erősítés-

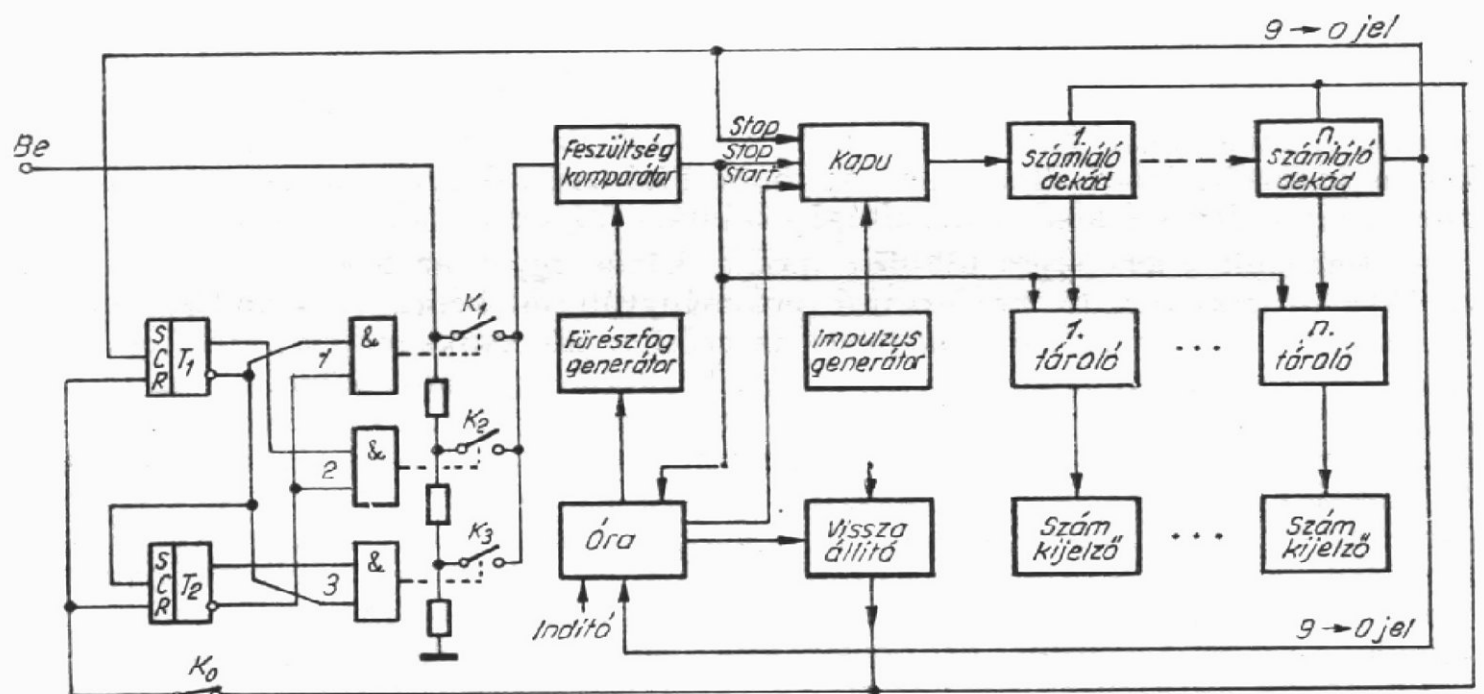


5.30. ábra.
Önbeállító erősítő

fokozatok figyelembevételével kell beállítani, nehogy az egyik komparátor által kezdeményezett tartományváltás egyidejűleg a másik komparátort aktivizálja. Zaj okozta tartományváltás elkerülése érdekében a komparátorok működését késleltetik; a komparátor csak akkor válik hatásossá, ha bemenő jele, legalább bizonyos meghatározott minimális időn át, fennmarad.

Az ábrán szaggatott vonallal jelzett egylövetű áramkör nem szükséges, ha az erősítő állandó jelleggel egyetlen jelforrásra kapcsolódik. Amennyiben azonban az erősítőre multiplexer kimenete csatlakozik, a csatornaváltás idejére a mutliplexer vezérlése az egylövetű áramkör útján letiltva az órajelet, hatástalanítja az önbeállító kört.

DVM automatikus tartományváltása. Az 5.31. ábra lineárisan emelkedő (lépcsős függvény) feszültséggel időtranszformációt végző A—D átalakítón alapuló digitális voltmérőt ábrázol. Az óragenerátor által indított és vezetett mérőjel 0-ról indulva bármely a 0 feszültség és a végkiterés (U_v) közötti feszültség szintet annak nagyságával arányos idő alatt ér el. A feszültséggel arányos időt az n dekádból álló számláló



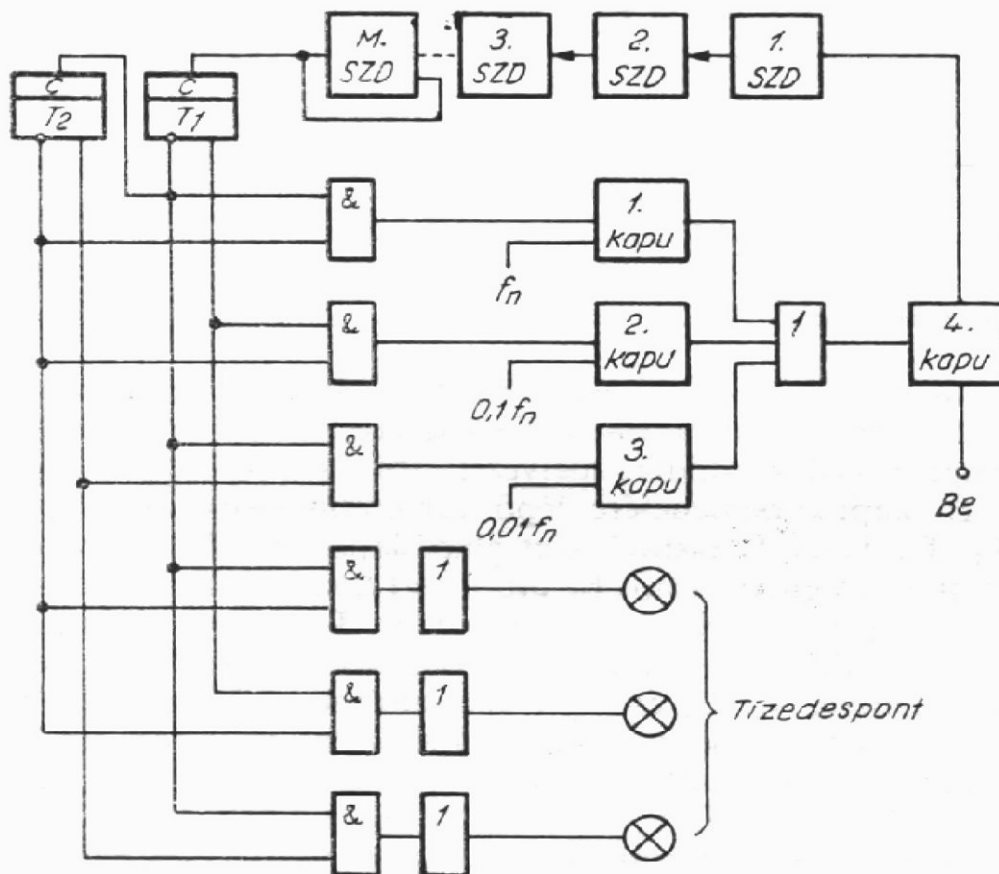
5.31. ábra.
Digitális voltmérő automatikus tartományváltással

az impulzusgenerátor impulzusainak számlálásával regisztrálja. A számlálási folyamatot az óragenerátor, a kaput kinyitva, a lineáris mérőjellel együtt indítja. Az U_v végkiterésnek a számláló feltöltött állapota, az utolsó dekád $9 \rightarrow 0$ átmenete felel meg. A számlálási folyamatot vagy a feszültségkomparátor, vagy a $9 \rightarrow 0$ átmenet jele állíthatja le. A komparátor akkor ad STOP jelet, ha a lineáris mérőjel eléri a bemenő (méréndő) feszültség szintjét.

Az automatikus tartományállítást a T1 és T2 bistabil, valamint a három ÉS kapu, a kapcsoló-működtető körökkel együtt végzi. A rendszer kezdő állapotában (bistabilok visszaállítva) a K1 kapcsoló van zárt állapotban, az indító bemenetről kezdeményezett mérés tehát a legalacsonyabb méréstartományban indul. Ha a méréndő jel ebbe a tartományba esik, a mérés véget ér, amikor a lineáris lejtőjel eléri a bemenő szintet, melynek értéke egyidejűleg a számkijelzőkön megjelenik. Amennyiben $U_{BE} > U_v$, a mérés első fázisa az utolsó dekád $9 \rightarrow 0$ jelével végződik. E jel egyrészt zárja a kaput, másrészt az óragenerátoron át nullára állítja a lineáris lejtőjelet, továbbá átállítja T1 bistabilt; K1 kapcsoló nyílik, K2 zárul. A leosztott bemenő feszültséggel ezután a mérési folyamat az előző módon megismétlődik, s ennek kimenetétől függően a mérési eredményt kapjuk, vagy a készülék $9 \rightarrow 0$ átmenettel a soron következő magasabb mérési tartományra tér át. Az automatikus méréshatárérték a tartományváltással együtt a tizedesvessző helyét is módosítja.

Minden feszültségmérést követően (ill. megelőzően) a rendszert kezdő helyzetébe kell visszaállítani. A visszaállítás során a rendszer ismét a legalacsonyabb méréstartományba áll vissza. Amennyiben egyvalamely méréstartományba eső sorozatmérést kell végrehajtani, az alkalmas méréstartományt (megtalálása után) KO nyitásával lehet rögzíteni.

Időmérő automatikus tartományváltása. Az 5.32. ábrán látható mérőkészülék a BE kapocsra vitt jel időtartamát méri. Alapállapotban (T1 és T2 bistabil nullázva) a 4. sz. kapura a normálgenerátor f_n frekvenciájú, T_n periódusidejű mérőjele jut.



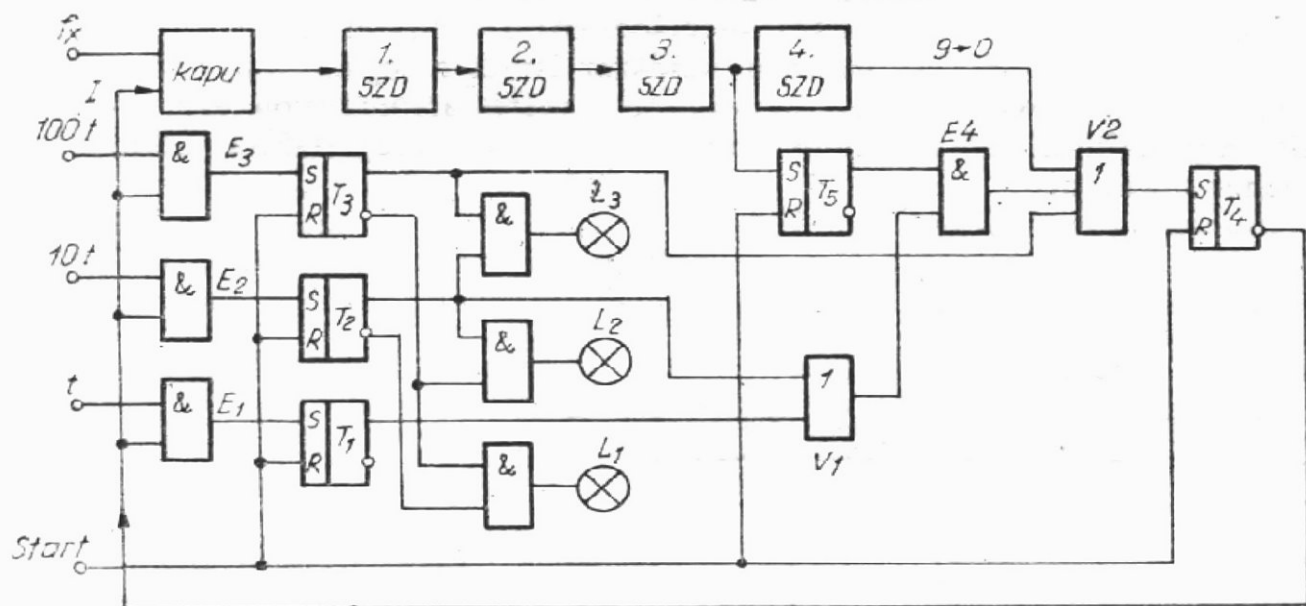
5.32. ábra.
Időmérő automatikus tartományváltással

A 4. sz. kaput a mérendő jel nyitja, melynek T szélessége és a normálgenerátor periodusideje között

$$T = nT_n$$

összefüggés áll fenn; n -t, a kapun átjutó impulzusok számát, az M dekádból álló számláló regisztrálja. Amennyiben $T > N \cdot T_n$, ahol N a számláló kapacitása, úgy a mérés újratekzdése nélkül tartományváltás következik be. A tartományváltást az utolsó (M .) dekád $9 \rightarrow 0$ átmenete hozza létre. A $9 \rightarrow 0$ átmenet jele egyrészt T_1 átbillentésével a 4. sz. kapu bemenetére $0,1f_n$ mérőjelet kapcsol, másrészt az M -edik dekádba $+1$ -et tölt be, a tizedespontot pedig egy pozícióval balra tolja el. A mérés megszakítás nélkül folytatódik. Ha újabb $9 \rightarrow 0$ átmenet jönne létre, a berendezés $0,01f_n$ mérőjelet választ. Így bármelyik mérési tartományban zárul T , a számláló kimenetén T értéke közvetlenül leolvasható. $T > T_{\max}$ esetben (ahol T_{\max} a készülék által mérhető leghosszabb idő) a mérés vagy blokkolható, vagy újra kezdhető, amikor is T értéke a T_{\max} ciklusok számának és a számláló végső tartalmának birtokában számítható ki.

Frekvenciamérő automatikus tartományváltása. Felépítésében a frekvenciamérő emlékeztet az időmérőre (5.33. ábra). Itt azonban a frekvencia a mérendő érték, a normált érték pedig az idő.



5.33. ábra.
Frekvenciamérő automatikus tartományváltással

A mérendő jel a négy dekádból álló számláló (SZD1—SZD4) kapuzott f_x bemenetére kerül. Az I kapuzó bemenet a mérési idő közlésére szolgál. A mérés a startvonalra adott indító impulzussal kezdődik, melyet t , $10t$ és $100t$ idő múlva, sorrendben az E_1 , E_2 , ill. E_3 kapuk bemenetére lépő záróimpulzusok követnek. Az indító impulzus a T_1 , T_2 , T_3 , T_4 és T_5 bistabilokat alapállapotba helyezi, nyitva egyúttal az E_1 , E_2 , E_3 , valamint a bemeneti kaput. Ez utóbbival f_x számlálása kezdetét veszi. Amennyiben a t időpontban megjelenő záróimpulzus T_5 -öt már átállított helyzetben találja, de az utolsó, SZD4, számlálódekád még nem adott $9 \rightarrow 0$ jelet, úgy a záróimpulzus T_1 -et átbillentve, V_1 , E_4 és V_2 -n keresztüljutva, átállítja T_4 -et. A számlálás zárul, a mért jel frekvenciája a számlálón leolvasható, a mérési tartományt L_1 indikálja. Ha azonban t időpontban T_5 még alapállapotában van, úgy E_4 zárva lévén a számláló tovább regisztrál. T_5 állapotának lekérdezésére $10t$ időpontban kerül újra sor. Igen válasz esetén a számláló leáll, nem válasz esetén $100t$ időpontig folytatódik. $100t$ időpontban, a kapcsolási logikából következően, T_5 állapotától teljesen függetlenül, a számlálás mindenképpen befejeződik. Előfordulhat

hogy a t idő alatt beérkező impulzusszám nagyobb, mint a számláló N kapacitása, azaz: $f_x > N/t$. Ez esetben a SZD4 dekád $9 \rightarrow 0$ jele állítja át T_4 -et, s zárja egyúttal a számlálási folyamatot. Ugyanakkor a számláló által mutatott érték: nulla.

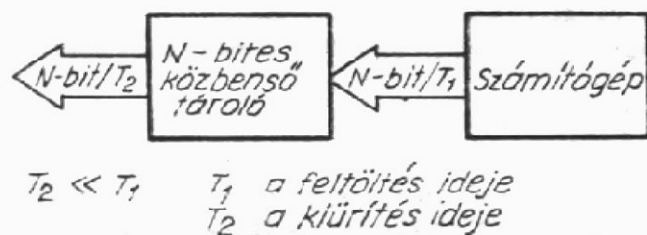
A mérőhálózat memóriájának bővítése

A készülékvezérlőt a külvilággal összekötő adatvonal mentén, a csatlakozóegységben, általánosan használnak egy vagy néhány pufferregisztert az adatok (adatelemelek) időszakos tárolására. Egyes esetekben e regiszterek lehetőségeit lényegesen meghaladó kapacitású közbenső tárolót kell alkalmazni.

A közbenső tároló egyik ismert feladatköre a sebességtranszformáció. A mérőrendszer hosszú időre számított adatáteresztő képessége, feltételezve, hogy a hálózatban nincs adatredukció, a számítógép (programfüggetlen csatornájának) adatáteresztő képességét nem haladhatja meg. Gyakran szükség lehet azonban rövid ideig tartó intenzív adatáramok vételére vagy kibocsátására, máskor pedig a beérkező adatok statisztikus eloszlása az időbeli felbontóképesség tekintetében támaszthat a számítógép adottságait meghaladó követelményeket.

Közbenső tárolón alapulnak a digitális áramkörök nagy sebességű ellenőrző hálózatai, melyeken a mindkét irányú rövididejű adatáram intenzitása a számítógép programfüggetlen csatornájának áteresztő képességét két nagyságrenddel is túllépheti (5.34. ábra). A paralel szógenerátor maga tulajdonképpen a közbenső memória állandósult és a célnak megfelelően működtetett változata. Egyes szógenerátorok (Instrum. Eng. DWG/R) nemcsak az ellenőrző vektorok kibocsátására, hanem a válaszvektorok egyidejű vételére is alkalmasak, betöltve a gép és a tárgy közötti mindkét irányú puffer szerepkörét.

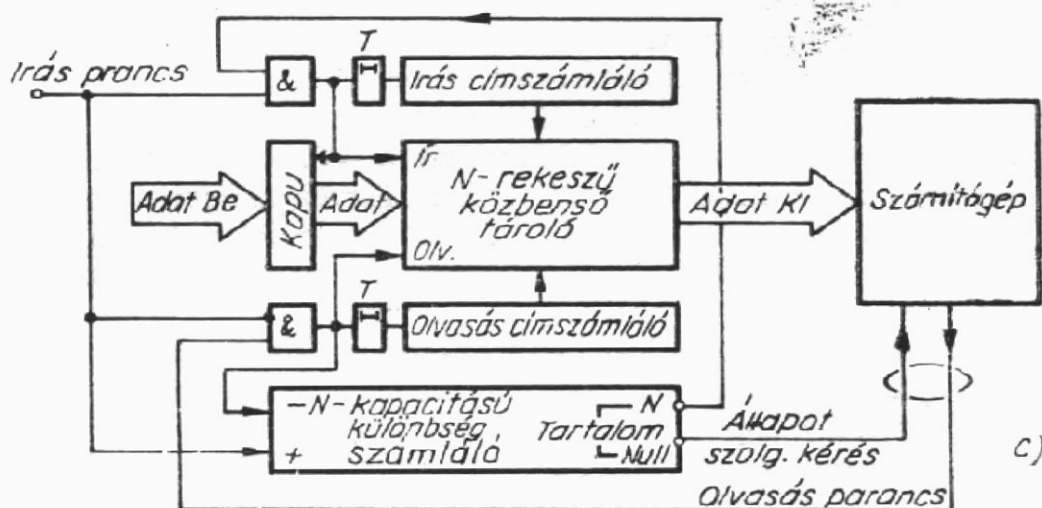
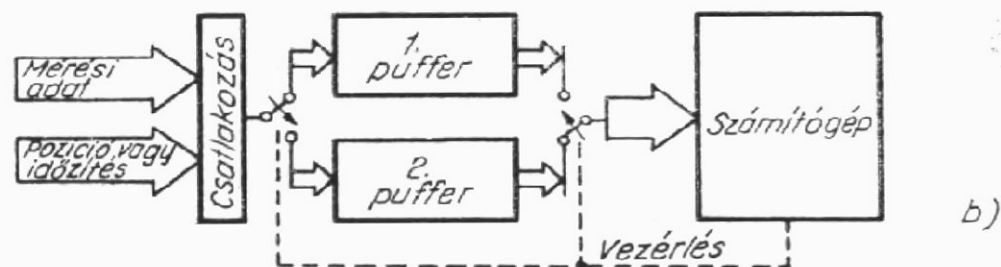
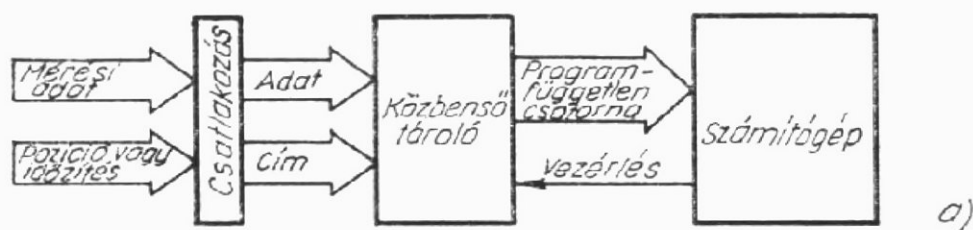
Más vonatkozásban a közbenső tároló gyors adatgyűjtő rendszerekben talál alkalmazást, ahol az adatbegyűjtés időszakában a mintavételezés gyakorisága által definiált adatáram meghaladja a számítógép programfüggetlen csatornájának át-bocsátóképességét. A közbenső tároló adatgyűjtőben való alkalmazásának három lehetséges változatát mutatja be az 5.35. ábra. Tárcapacitás felhasználása szempont-



5.34. ábra.

A mérőrendszer által kibocsátott adatáram intenzitásának növelésére szolgáló közbenső memória

jából kevésbé gazdaságos az *a*) változat, melynek működési logikája: először az egy begyűjtési periódushoz tartozó összes adat a tárolóba kerül, majd a begyűjtési szakasztól elkülönített feldolgozási periódusban a közbenső tároló teljes kiürítésére kerül sor. A *b*) változatnál a közbenső tároló két egyforma részre oszlik fel, s míg az egyik rész feltöltődik, a másiknak tartalma a gép memóriájába kerül át. E tárolótípus alkalmazása csak a bemenő adatok statisztikus eloszlása esetében jár haszonnal, ha a mérőrendszer adatárama nem lépi túl a gép csatornájának át-bocsátóképességét, de a közbenső tár időbeli felbontóképessége lényegesen jobb, mint a számítógépcsatornáé. Elsősorban ugyancsak az időbeli felbontóképesség transzformációjára alkalmazható a *c*) változat. E puffertár működését a kötött és egymással meg-egyező beírási és kiolvasási sorrend mellett még az jellemzi, hogy a beírási és kiol-



5.35. ábra.

A mérőhálózat és a számítógép között sebességkonverziót végző közbenes memóriák

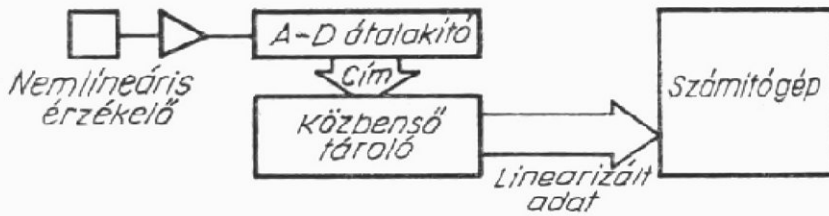
(a) először adatbegyűjtés, majd a számítógép felé kiürítés; (b) míg az egyik (pl. 1.) puffer feltöltődik, a másik a gép felé ürítődik; (c) szimultán töltés és ürítés

vasási fázis időben nem különül el egymástól. A tárolónak két független számláló címregisztere van, melyek közül az egyik a beírásra, a másik a kiolvasásra soron következő memóriarekeszre mutat. Az írás-címregiszter tartalma minden beírás, az olvasás-címregiszteré minden kiolvasás kapcsán +1-gyel nő. Írás bármikor lehetséges, míg van üres tárolórekesz; olvasás bármikor lehetséges, ha van tartalommal rendelkező rekesz. A rekeszek állapota pl. kétirányú (differencia) számláló segítségével követhető nyomon. A kétirányú számláló összegező bemenetével a beírás műveleteket, levonó bemenetével az olvasás műveleteket számlálva, a számláló tartalma a tároló állapotát tükrözi. Ha a tartalom zérus, a tároló üres; vagy még nem történt beírás, vagy az elvégzett írás és olvasás műveletek száma egymással megegyezik. Ha a számláló teljesen feltöltődik (feltételezve, hogy számkapacitása a tároló szókapacitásával azonosan: N), ez azt mutatja, hogy N -nel több beírás, mint ahány kiolvasás művelet történt. A kétirányú számláló tartalma alapján a közbenes tároló három alapállapota különböztethető meg:

- tartalom zérus: minden rekesz üres; csak beírás végezhető;
- tartalom >0 , de $<N$: a rekeszek egy része üres, más részének tartalma van; olvasás, írás egyaránt végezhető;
- tartalom N : minden rekesz foglalt, csak olvasás végezhető.

E tárolási séma lehetővé teszi, hogy olvasás műveletet ékelődjenek írás műveletek közé. Olvasás mindig létrejöhet, ha írás nincs folyamatban és a tároló tartalma a zérustól különbözik.

Tárolót mérőhálózatban *linearizáló elemként* is alkalmaznak. Nemlineáris érzékelő digitalizált kimenő értékeit tároló címtartományához rendelve, s az egyes címek alatt a lineáris jelleggörbének megfelelő értékeket tárolva, a memória a nemlineáris jelleggörbét lineárisra képezi le. Az 5.36. ábrán szemléltetett egyszerű séma azonban



5.36. ábra.

Közbenső tároló alkalmazása érzékelő nemlineáris jelleggörbéjének linearizálására

meglehetősen bonyolulttá válik, ha a mérőrendszer multiplex üzemmódban számos érzékelő jelét gyűjti be, s a linearizáló tároló egynél több érzékelő jelleggörbéjét linearizálja. Mód van arra is, hogy az egyes csatornák határértékeit a linearizáló memória egy szektorában tárolják, s hogy a mért értékeket a határértékkel még a számítógépen kívül összevethessék.

Mikroprogramozott vezérlés alkalmazása

Mérőkészülékhez, mérőkészülékek vagy építőköcskák meghatározott funkciót ellátó együtteséhez mikroprogramozott vezérlő társítható a célból, hogy a számítógéptől kapott parancsok alapján bizonyos, a mérésekkel kapcsolatos standard szekvenciális mérőhálózati műveleteket bonyolítson le. Mikroprogramozott vezérlőnek készülékben való alkalmazása tekintetében, ugyanazok a szempontok érvényesülnek, mint amelyeket a kisszámítógépekkel kapcsolatban már áttekintettünk. Elsősorban flexibilitását kell kiemelnünk, adott feladathoz való optimális illeszthetőségét, mely átírható tartalmú tár és dinamikus mikroprogramozás alkalmazásával még inkább fokozható.

A szokványos vezérlési feladatokon túl a mikroprogramozás elvének felhasználásával igen speciális mérőhálózati problémák is megoldhatók. A lehetőségekre irányítva a figyelmet a mikroprogramozható szógenerátort említjük meg. A szokványos szógenerátor a tárolójába beírt digitális mintázatot változatlan alakban, ciklikusan ismételteti. Szükség lehet azonban a vizsgálati előírásoktól (és feltételektől is) függően a mintázat egy-egy rövidebb-hosszabb szakaszának menet közbeni, előírt számú reciklálására is. A Tau-tron WG 304A szógenerátor az ilyen jellegű előírásokat mikroprogram-vezérléssel oldja meg. A szógenerátor ≈ 10 ns ciklusidejű, olvas/ír rendszerű félvezető tárja 18 bit rekesz-szélességű A szektorból és 16 bit rekesz-szélességű B szektorból áll; mindkét szektor 64 rekesz kapacitású. Az ellenőrzés szóhosszúsága 16 bit; az A szektor rekeszének maradék 2 bitje mikroparancsot tartalmaz. A generátor részét képezi még egy címmutató és egy ún. szinkron-pozíció regiszter is. A generátor alapüzemmódja ugyanolyan, mint az egyszerű paralel szógenerátoroké; B szektor az A mögé kapcsolódik, a kettő 128 szavas együttest képez, és az ebben tárolt 16 bit szélességű digitális mintázat ismétlődik az óragenerátor lépési ritmusával haladva kifelé. Követő üzemmódban, a B szektor működési ritmusát az A szektorban tárolt parancsok szabják meg.

Az A szektorban a szóval együtt tárolt 2-bites mikroparancs első bitje a mintázat szavainak szekvenciálását megállítja, és a generátor címmutató regiszterében található, számítógép által vagy manuálisan propprogramozott memóriacímre való kitérést hoz létre. A második bit a B szektor címvezérlőjének ad inkrementum utasítást. Ily módon lehetővé válik az A szektor tetszőleges szakaszának tetszés szerinti számban való megismétlése, a B szektor tartalmának az ellenőrző rutin részeként való felhasználása, vagy hogy pl. memóriavizsgálat esetén a B szektor a vizsgált elem címeit szolgáltassa.

A fent ismertettnél bonyolultabb mikroprogram strukturájú szógenerátor is ismeretes (Tau-tron WG—15A), melynek mikroutasítása 8 bit hosszúságú; a mikroutasítások tartalmazzák a kitérés címet is, és sorukban feltételes kitérő utasítás is található; a belső ciklusok száma repetíció számláló és programozható repetíció regiszterek útján determinálható.

Mikroprocesszor alkalmazása

A mikroprocesszor a mérés technikára többirányú hatást is gyakorol. Számos vonatkozásban talál alkalmazásra manuális jellegű mérőkészülékekben, kiterjesztve azok lehetőségeit, fokozva teljesítményüket és pontosságukat. Az automatizált mérőrendszereket tekintve a mikroprocesszor egyrészt bizonyos területeken átveszi a kisméretű számítógép feladatkörét, másrészt a mérőhálózatban, kisegítő szerepkörben, felmenti a kisméretű alacsonyabb szintű feladatok terhe alól. Itt most a mikroprocesszorra csak a mérőhálózatot érintő vonatkozásban térünk ki.

A mikroprocesszor a mérőhálózat vezérlési szférájában a mikroprogramozott vezérlőt programozhatóságának flexibilitása és aritmetikai—logikai lehetőségei tekintetében múlja felül. Másrészt azonban azokban az esetekben, amikor nem az ezen tulajdonságok által nyújtott lehetőségek, hanem a vezérlés sebességi követelményei állnak előtérben, mikroprogramozott vagy kötöttprogramú vezérlő alkalmazása előnyösebb lehet. A mikroprocesszor olcsósága és külső megjelenésének egyszerűsége lefedheti áramkörü—logikai strukturájának bonyolultságát és programozásának problematikáját. A mikroprocesszornak adott feladat szempontjából való értékelésről csak e tényezők szem előtt tartásával lehet szó.

A mérőhálózatban alkalmazott mikroprocesszor elláthat általános hálózati feladatokat, de munkaköre korlátozódhat egyetlen mérőkészülékre, vagy a hálózat mérőkészülékként tekinthető elhatárolt területére is.

Általános feladat pl. a mérőrendszer bekapcsolása után a mérőhálózat elemeinek kezdőállapotba helyezése, vagy feszültségkiesés kapcsán a mérőhálózat állapotának memorizálása, majd a feszültség visszatérése után a hálózati állapot visszaállítása. Hasonlóképpen általános feladat a mérőhálózat készülékeinek, ill. a mérőhálózatnak mint összefüggő egésznek kalibrálása. A kalibrálás általában időt rabló folyamat (egy becslések szerint az Egyesült Államokban évi 25 millió munkaórát töltenek el kalibrálással), olyan bonyolultabb mérőhálózatok esetében pedig, melyeknek elemeit számos, esetleg kétirányú reláció fűzi össze, igen nehezen elvégezhető munka, mivel az egyes elemeken végzett korrekció más elemek újrabebiztosítását teszi szükségessé. Az ilyen bonyolult, iterációs eljárást igénylő kalibrációs feladatok ugyancsak mikroprocesszorra háríthatók. Általános feladatkörként a mikroprocesszor elláthat a mérőhálózat—tárgy együttesében szabályzási feladatokat, így a tárgy bizonyos környezeti paramétereinek (pl. hőmérséklet) ellenőrzését, stabilizálását, esetleg a központi program szerinti beállítását (változtatását).

A mikroprocesszorra számos individuális készülékfeladat osztható ki. Első helyen az automatikus tartományillesztés (tartományváltás) feladatkörét említhetjük meg.

Láthattuk azonban, hogy esetenként a tartományváltás igen egyszerű szerkezeti logikával oldható meg, s egyedül ennek helyettesítésére mikroprocesszort alkalmazni általában nem indokolt.

Környezeti hatásoknak a mérőkészülék valamely paraméterére gyakorolt befolyását — hagyományos módon — a hatás kompenzálásával igyekeznek megoldani. Mikroprocesszort alkalmazva merőben más út követhető: megengedik, hogy a környezeti hatás érvényesüljön, az általa a mérési eredményben okozott hibát pedig a mikroprocesszor a készülék környezetfüggésének, valamint a környezeti hatás mennyiségi változásának ismeretében korrigálja. Tranzisztoros erősítő pl. hőmérsékleti kompenzáció nélkül alkalmazható, ha a jelenlévő mikroprocesszor egyrészt ismeri a hőmérsékleti függést, másrészt érzékeli a hőmérsékletnek valamely alapértékhez viszonyított változását, és ezen ismeretei alapján értékcorrekciót végez. Ugyanilyen módon a mikroprocesszor nullpont-kiigazítást is végezhet, és pedig nem egyedül csak a hőmérséklet, hanem a tápfeszültség ingadozására is figyelve.

A mikroprocesszor jelentős szerepet játszhat a „sokdimenziós” készülékek vezérlése területén. A sokdimenziós, tehát számos programozott paraméterű mérőkészülék vezérlésének problémáit elég talán a korábbiakban, az 5.16...5.18. ábrákkal kapcsolatban már bemutatott digitális strukturájú impulzusgenerátoron szemléltetnünk. A vezérlés első problémája abban áll, hogy minden egyes funkcionális modult külön-külön kell megcímezni, sőt egyes modulokhoz (ilyen a kimenethajtó kör) egynél több cím is tartozhat. A központi irányítást végző számítógépnek a paraméterek programadataihoz tehát címeket is kell mellékelnie. Az is nehézséget jelent, hogy bár a digitális programadatok és az impulzusgenerátor jelparaméterei közötti kapcsolat szigorúan determinált, ugyanakkor azonban nem közvetlen. A frekvencia pl. az órafrekvencia és a programozott osztásérték hányadosa: a késési idő az óraciklus és a késleltetést meghatározó számláló programozott számkapacitásának szorzata, az impulzusszélesség pedig két idő különbsége, melyet a késleltetés minden értékéhez újra kell meghatározni. A generátor programozása, mikroprocesszoros vezérlése esetén az impulzusparaméterekkel, címek közlése nélkül történhet; az impulzusparamétereket a mikroprocesszor digitális vezérlőértékekre fordítja le, a vezérlőértékhez csatolva rendeltetési címét is.

A mikroprocesszor számos alkalmazási lehetőségét, melyek egy része a mérőhálózatot érinti, már felsoroltuk a 3.1.4. szakaszban, a csatornaprocesszorok tárgyalásakor.

Irodalom az 5. fejezethez

- [1] *Sebestyén B.*: Programirányítású mérőkészülékek. Híradástechnika XXIII. évf. 4. sz. 1972, 107...111. old.
- [2] *Geraci, E. E.—Tonooka, S.*: A computer-controlled highprecision resistance measuring system. ISA Transactions, Vol. 9, No. 3. 1970, 236...242. old.
- [3] *Parker, Y.*: A binary floating-point resistor. IEEE Trans. Comput., C—20. No. 1. 1971, 7...11. old.
- [4] *Lennartz, A.*: Digital-Potentiometer, Teil—I. Automatik, Jan. 1972, 14...17. old. Teil-II. Automatik, Febr. 1972, 48...54. old.
- [5] *Geraci, E. E.*: A programmable potentiometer system. ISA Transactions, Vol. 8, No. 2. 163...168. old.
- [6] *Малиновский, В. Н.*: Цифровые мосты постоянного тока с бесконтактным уравниванием. Приборы и системы управления, № 2. 1973, 27...30. old.
- [7] *Tobey, G.*: Programmable data amplifiers. ISA Transactions, Vol. 11, No. 3. 250...258. old.
- [8] *Sedra, A.*: Simple digitally-controlled variable gain linear d. c. amplifier. Electronic Engineering, March 1969, 362...365. old.

- [9] *Sperazza, J. L.*: A computer controlled instrumentation amplifier. *Advances in Instrumentation*, Vol. 25, Part 1—4. ISA—70. Silv. Jubil. Conf. 612/1—5.
- [10] *Birman, P.*: Digitally controlled power supplies. *Electronics World*, Oct. 1971, 28...29. old.
- [11] *Birman, P.*: Using operational power supplies. KEPCO, Inc., Publ. No. 146—1281, 1973, 34. old.
- [12] *Nordgren, B. M.*: Systems-oriented digital power sources. *Hewlett—Packard Journal*, June 1968, 9...16. old.
- [13] *Christiansen, R. A.*: Programmable AC power systems... wyes and wherefores. *EID—Electronic Instrumentation*, May, 1972, 18...23. old.
- [14] Multiphase programmable pulse generator. XINCOM corp. Applications manual, Pub. No. 0371617.
- [15] *Sear, B. E.*: Programmable precision pulse drivers for unterminated lines. *Computer Design*, May 1972, 103...110. old.
- [16] *Patterson, E.*: Select pin drivers cautiously. *Electronic Design* 4, Febr. 15. 1975, 82...84. old.
- [17] *Chauvel, G.*: Réalisez votre générateur de mots. *EMI* 191/15—6—74, 43...50. old.
- [18] *Kubinec, J. J.*: Build a programmable word generator. *Electronic Design* 2, Jan. 18. 1969, 62...67. old.
- [19] *Brinton, J.*: Word generator is fast and flexible. *Electronics*, Dec. 18. 1972, 117...118. old.
- [20] *Donn, E.*: Generating words for digital testing. *Hewlett—Packard Journal*. April 1969, 8...13. old.
- [21] *Stokes, V. O.*: Techniques of frequency synthesis. *Proc. IEE*, Vol. 120. No. 10R, Oct. 1973, 1057...1077. old.
- [22] *Pócza A.*: Nagy pontosságú frekvenciagenerátorok. *Híradástechnika XXIV. évf. 2. sz.* 55...60. old.
- [23] *Erdman, A. C.*: Operation of programmable frequency dividers. *Computer Design*, May 1974, 110...112. old.
- [24] *Vandeman, G. S.*: Programmable frequency multiplier uses octave scaler to simplify programming. *Electronic Design* 18, Sept. 1. 1974, 100. old.
- [25] *Elliot, D. F.—Sypherd A. D.*: Simplify function generator design, *Electronic Design* 21, Oct. 11. 1970, 86...89. old.
- [26] *Cooper, H. W.*: Why complicate frequency synthesis? *Electronic Design* 15, July 19. 1974, 80...84. old.
- [27] *Ricci, D. W.—Nelson, G. E.*: Standard instrument interface simplifies system design, *Electronics*, Nov. 14. 1974, 95...106. old.
- [28] *Krause, A.—Vogt, W.*: Einige Beispiele für automatische Messbereichumschaltung bei digitalen elektronischen Messgeräten. *ATM*, März 1967, R25—R29.
- [29] *Laird, J. R.*: Memories in instrumentation. *Instruments and control systems*. May 1970, 21...22. old.
- [30] *Bounds, P.*: Buffering high speed data for minicomputer input. *Computer Design*, July 1973, 69...73. old.
- [31] *Falk, H.*: The microprocessor, *IEEE spectrum*, Nov. 1974, 46...51. old.
- [32] *Allan, R.*: Instruments and test equipment. *IEEE spectrum*, Jan. 1975, 90...94. old.
- [33] *Cole, B.*: Microprocessor runs signal generator. *Electronics*, April 3. 1975, 113...114. old.
- [34] *Runyon, S.*: Instruments with improved 'minds' will be taking over dedicated jobs. *Electronic Design* 1, Jan. 4. 1975, 34...35. old.
- [35] *Sample, P.*: The frequency synthesiser as a signal generator. *Electronic Engineering*, Dec. 1974, 39...43. old.
- [36] *Santoni, A.*: Microprocessors await the call. *Electronics*, March 20. 1975, 76...80. old.

6.

Programrendszerek

6.1. Kiszámítógépek programrendszerei

A kiszámítógép programrendszerén a feladatok egy körének előírt módon történő végrehajtásához szükséges rendszerprogramok összességét értjük. A programrendszer, implementációkba rögzítve tartalmazza a benne alkalmazható nyelvek pontos definícióit is. A programrendszert az általa felölelt programok tartalma, funkcionális struktúrája és architektúrája jellemzi.

Egy adott géprendszerhez különböző célokra szolgáló többféle programrendszer is tartozhat, esetenként pedig valamely feladatkör ellátására is több, egymástól színvonalában eltérő programrendszer állhat rendelkezésre. Az egy géphez tartozó különböző programrendszerek mindegyike bizonyos meghatározott szerkezeti környezetet, a szerkezeti komponensek (tárak, periférius készülékek stb.) meghatározott együttesét tételezi fel.

A rendszerprogramok két válfaja különböztethető meg: 1. egzekutív programok és 2. feladatprogramok. A rendszerhez tartozó feladatprogramok éppúgy proceszszálást végeznek, mint az alkalmazási feladatprogramok. Az egzekutív programok ezzel szemben különböző (rendszer vagy alkalmazási) feladatprogramok végrehajtásával kapcsolatos adminisztratív teendőket látnak el.

6.1.1. Nyelvek és implementációik

Assembly nyelvek

A kisgépek mindegyike rendelkezik assembly nyelvvel. Az assembly nyelvű programokban a gépi utasításokat szimbolikus ekvivalenseik képviselik. Következésképpen, minden gépnek más-más assembly nyelve van. Egy géphez, egy és ugyanazon gépi nyelvhez, több — jelölésrendszere és szintaktikája tekintetében különböző — assembly nyelv is tartozhat. Másrészt egyugyanazon nyelvnek többféle változata is lehetséges, melyek egymástól terjedelmükben különböznek (pl. ASTROL—4 és ASTROL—8). A kisebb terjedelmű változat általában a nagyobb változat részhalmaza. A különböző terjedelmű gépi nyelvek különböző gépi konfigurációkat tételeznek fel.

A nyelv (nyelvi változat) implementációján szűkebb értelemben a fordítóprogram (ez esetben assembler), tágabb értelemben mindazon rendszer-feladatprogramok összessége értendő, melyek felhasználásával az assembler nyelven írt program gépi nyelvre fordítható és a memóriába helyezhető. A különböző nyelvi implementációk eltérhetnek egymástól metodikailag (egyszeres és többszörös átfutású assemblerek), a relokációs lehetőségek, a szerkesztési, hibamentesítési stb. eljárások tekintetében.

Egyes implementációk a direkt utasítások mellett makroutasítások használatát is megengedik. A makro nyelven írt programban a programozó által előre definiált utasítássorozatok egyetlen makroutasítással adhatók meg. A gép programrendszeréhez tartozó makroprocesszor a programban makroutasítást érzékelve, helyére az előre definiált utasítássorozatot építi be.

A mikroprogram szinten programozható gépek mikroassembly nyelvvel és ennek megfelelő assemblerrel (stb.) rendelkeznek.

Felsőbb szintű általános nyelvek

A felsőbb szintű általános nyelvek két fő célkitűzés alapján alakultak ki: 1. a nyelv legyen független az alkalmazott gép szerkezeti adottságaitól, legyen általánosan használható; 2. jelölésrendszere minél inkább közelítse meg a természetes (írott) nyelvet. A meghatározott alkalmazási területekre optimalizált ALGOL, FORTRAN, BASIC, COBOL, PL/1 stb. e célkitűzéseknek a nagygépek környezetében messzeemenően megfelel. E nyelvek természetesen nem egyszerre álltak elő, hanem a fejlődés számos fokozatán haladtak át. Példaképpen elég talán a FORTRAN-nak I, II, III, IV jellel megkülönböztetett, valamint az ASA (USASI) FORTRAN és Basic FORTRAN változatait megemlíteni.

A felsőbb szintű nyelvek kisgép-adaptációja kapcsán a nagygépek körében már elért eredmények egy jelentős hányada elveszett. A kisgép-implementációk igen diszperz képet mutatnak, mivel egyrészt a nyelvek nagygép-változatainak különböző fejlődési szintjeiről ágaznak el, másrészt e változatok leszűkítése esetenként más-más módon történt, s még egy-egy géprendszer többféle konfigurációjához is többféle módosulat tartozik. A FORTRAN-nál maradvá a helyzet illusztrálására megemlíthetők pl. az igen szerény lehetőségű DEC 4K, a Basic FORTRAN-ból leszűkített DEC 8K, az ASA FORTRAN-hoz hasonló HP 4K, és HP 8K, a FORTRAN IV adottságait csaknem elérő Data General 8K és 12K valamint General Automation SPC—16 (12K, ill. 16K) FORTRAN implementációk. Ez azonban csak a számbavétel kezdete, melyet a többi géphez adaptált FORTRAN változatok, valamint a speciális alkalmazási területekre irányuló FORTRAN dialektusok áttekintésével lehetne folytatni.

A nyelvi implementációk számosságára tekintve, a nagygépek körében általános érvényű nyelvek a kisgépek vonatkozásában már aligha mondhatók akár gépfüggetlennek, akár általánosnak, akár egymás között kölcsönösen kompatibilisnek.

Adott kisgéphez implementált felsőbb szintű nyelv nemcsak abból a szempontból korlátos, hogy más kisgépeken nagy valószínűséggel nem használható, de a fordítóprogramok szerény adottságai megfelelő határfokú kódolást sem tesznek lehetővé; a fordítás eredményeképpen kapott gépi programok térfogat- és időredundánsak. Az újabb kisgépek programozástechnikája ugyan lehetővé teszi a nyelvi implementációknak az alapnyelvhez való közelítését, ez azonban természetesen a fordítóprogramok terjedelmének növelésével jár együtt, ami a rendszerprogramozás és a szerkezet költségeinek arányát borítja fel. Ugyanakkor még a legfejlettebb kisgép-implementációk hatásossága is nagyságrendekkel elmarad a nagygépekhez viszonyítva (ha nem másért, hát a sokkal kisebb teljesítményű periférikus készülékek miatt).

A nagygépektől átvett felsőbb szintű általános nyelvek mellett találkozhatunk eredetileg a kisgépek környezetében kifejlesztett nyelvekkel is, ilyen pl. a PDP 8 családdhoz tartozó FOCAL. Az ilyen, egy-egy géptípushoz kötődő nyelvek általában nélkülözik a nagy nyelvcsaládok általános elterjedtségéből és tapasztalati bázisából származó előnyöket.

Alkalmazásorientált nyelvek

A nagyszámítógépek alkalmazási spektruma igen keskeny; a gépek túlnyomó többségét (adatfeldolgozás és/vagy tudományos-technikai-gazdasági) számítástechnikai célra használják. Ez az oka annak, hogy a gépek zöménél ugyanazzal az (előbb már felsorolt) néhány nyelvvel találkozunk, s ezek a felmerülő alkalmazói igényeket ki is elégítik. A kisgépeket számítástechnikai feladatokhoz (részben éppen általános felsőbb szintű nyelvek eddigi rossz hatásfoka miatt) csak az utóbbi időben kezdik alkalmazni. Más irányú alkalmazási spektrumuk azonban igen széles, számos speciális területet (sőt individuális felhasználást) ölel fel. E területek igényeit vagy assembly nyelvvel, vagy/és speciális alkalmazásorientált nyelvekkel elégítik ki. Ez utóbbiakhoz sorolhatjuk mindenekelőtt az általános felsőbb szintű nyelvek alkalmazásorientált dialektusait és számos, egy-egy alkalmazási terület sajátos igényeit valamilyen szempontból optimálisan kielégítő nyelveket is, mint amilyen pl. a PEARL, a CORAL 66 és a műszaki mérés technika céljára kidolgozott ATLAS.

6.1.2. Rendszerprogramok

Assemblerok és makroprocesszorok

Az assembler: fordítóprogram, mely az assembly nyelvű (közbenső hordozón levő) forrásprogramot gépi nyelvű tárgyprogramra teszi át. Egyrészt a szimbolikus utasításokat gépi ekvivalenseikre képezi le, másrészt értelmezi és végrehajtja a forrásprogram pszeudoutasításait (direktíváit). A pszeudoutasítások formailag megegyeznek a forrásprogram utasításaival, azonban nincs gépi ekvivalensük. A program összeállítója általuk adja meg az assemblernek a tárgyprogram összeállítására vonatkozó parancsait.

Az assembler által végzett főbb részfeladatok: szimbólumok értelmezése, műveletjelek helyettesítése meghatározott gépi értékekkel, a címjelentésű szimbólumok helyettesítése számszerű címértékekkel, adatok, változók lokalizálása, adatmezők elhelyezése, literálisok kezelése, utasítások generálása stb.

Egyszeres és többszörös átfutású assemblerek különböztethetők meg. Az egyszeres átfutású assembler a tárgyprogramot (gépi nyelvű programot) a forrásprogram egyszeri processzálásával adja. A többszörös átfutású assembler a forrásprogramon egy vagy több menetben előprocesszálást végez, s a tárgyprogramot csak az utolsó átfutás eredményeként szolgáltatja. A kétszeres átfutású assembler az első átfutás kapcsán pl. szimbólum meghatározást végez, s a második átfutással generálja a kódot. Ismeretesek olyan, látszólag egyszeres átfutású assemblerek is, melyek bizonyos feladataikat (pl. előlegezett szimbólumok meghatározása) a töltő programra hárítják át.

A makroprocesszor az assembly nyelv szintjén makroutasítások alkalmazását lehetővé tevő program. Legegyszerűbb alakjában öt fő feladatot lát el: 1. felismeri a makrodefiníciót; 2. a makrodefiníciót tárolja; 3. felismeri a makroutasításokat (hívásokat); 4. a makroutasításban megadott argumentumokat a makrodefinícióba helyettesíti; 5. a makrodefiníció utasítássorozatát a programba a makrohívás helyére beépíti. A makroban magában is alkalmazható makrohívás, ami azonban a processzor szerkezetét is bonyolultabbá teszi. A makroprocesszor az assemblertől különálló, egyszeres vagy kétszeres átfutású preprocessoroként szerepelhet, de az assembler szerves részét is képezheti.

Compilerek

A compiler: fordítóprogram, mely az általa implementált magasabb szintű nyelven írt és közbenső hordozón levő forrásprogramot mint bemenetet processzálja; eredményként a forrásprogrammal ekvivalens értelmű, gépi (vagy assembly) nyelvű programot ad. A compiler feladatstruktúrája az assemblerénél lényegesen bonyolultabb. Bizonyos jelláncolatokat címke, változó, operátor stb-ként kell felismernie; körül kell határolnia és szintaktikai egységként kell felismernie bizonyos egymással előírt kapcsolatban levő elemeket, egyúttal elvégezve értelmezésüket; el kell végeznie a tárkijelölést, s ki kell jelölnie a változók helyét; végül szolgáltatnia kell a tárgyprogramot. E feladatok kapcsán lexikális és szintaktikai analízist, értelmezést, optimalizálást, tárkijelölést, kódgenerálást és összeállítást kell végrehajtania, s végül kimenetet kell képeznie.

A compilerek ugyancsak egyszeres vagy többszörös átfutásúak (a DEC 4K, és a General Automation FORTRAN compilerek pl. egyszeres, a Data General, a DEC 8K és a HP 8K FORTRAN compilerei kétszeres, a HP 4K FORTRAN compilere négyszeres átfutású). Az átfutások számának csökkentése érdekében bizonyos funkciók a compilerről más szolgálati programokra, így pl. a töltőprogramra vagy az általában rendelkezésre álló assemblerre háríthatók át.

A compiler értékelésével kapcsolatban a kisgépek szempontjából figyelembe veendő az a tapasztalati tény, hogy a compiler által elfoglalt tárterület nagyságának csökkentésével (a compiler zsugorításával) arányosan növekszik az általa képzett kód redundanciája és a kompilálás ideje is.

A compiler adott forrásnyelv és adott gép között létesít kapcsolatot, s ezért mind nyelvfüggő, mind gépfüggő.

Kereszt-assemblerek és kereszt-compiler

A kereszt-assembler adott (tárgy)gép számára végez (assembly szintű) fordítást, de egy a tárgygéptől különböző más gépen futtatható. Alkalmazása elengedhetetlen, ha a tárgygép memóriakapacitása oly kicsiny (pl. mikroprocesszor), hogy abban (saját) assembler nem helyezhető el. Más esetekben alkalmazása különböző előnyökkel járhat. Amennyiben pl. a kisgép periférikus rendszere szegényes, saját assembler használata időtrabló és költséges lehet, nagyobb gépen futtatott kereszt-assembleréhez viszonyítva. A kereszt-assembler alkalmazása a műszaki mérés technika területén különösen gazdaságos olyan környezetben, ahol a mérőrendszerekben számos, azonos típusú kisgépet alkalmaznak, de ahol központi nagy gép áll rendelkezésre programfejlesztés céljára.

A kereszt-assemblereket általában magasabb szintű nyelven (pl. FORTRAN) dolgozzák ki, ami azzal az előnnyel jár, hogy bármely, az illető nyelvet szabatosan implementáló gépen futtatható, azaz messzemenően gépfüggetlen.

A kereszt-compiler ugyancsak idegen gépen futtatott fordítóprogram; magasabb szintű nyelven írott programoknak a tárgygép nyelvére való áttételére szolgál. Kimenetét a tárgygép assembly nyelvén vagy gépi nyelvén szolgáltatja. A kereszt-compiler alkalmazásával mindazok a problémák megkerülhetők, melyeket magasabb szintű nyelvek kisgép adaptációival kapcsolatban megemlítettünk.

A kereszt-assembler és a kereszt-compiler használatával elkerülhető, hogy a kisgéphez programelőkészítő operációs rendszert kelljen alkalmazni.

Töltők

A töltő: rendszerprogram, amelynek feladata a programoknak közbenső hordozóról a memóriába való bevitele. Legegyszerűbb változataik a rögzített memóriahelyre való beültetésen kívül más feladatot nem látnak el. Esetenként igen bonyolult szerkezete van a tárgyprogram beültetésére használt töltőnek. A tárgyprogram, a fordítás során, közbenső információhordozóra (lyukszalag, mágnesszalag stb) kerül; a memóriában általában azért nem helyezhető el közvetlenül, mivel azt a fordítóprogram foglalja le. Ezért, a fordítás befejezte után a most már felszabadult memóriába külön munkafolyamattal (töltővel) kell beültetni. A beültetés helyét egyszerűbb esetben a programozó definiálja a forrásprogramban, s az erre vonatkozó információt az assembler a töltő számára a tárgyprogramban rögzíti. Az ilyen — abszolút — töltési sémát az jellemzi, hogy ha a tárgyprogram elhelyezését módosítani kell, úgy a módosítást a forrásprogramban kell elvégezni, és a szövegfordítást meg kell ismételni. Abszolút töltő alkalmazása különösen nehézkes, ha több önálló programot (pl. egy főprogramot különböző szubrutinokkal) kell összekötni, mivel a kapcsolókat forrásprogram szinten kell definiálni. Ennek az a hátránya, hogy minden egyes programot (a szubrutinokat is) minden esetben újra le kell fordítani. A relokációs töltési sémák az abszolúttal szemben lehetővé teszik a gépi kódban tárolt, számszerű címértékekkel rendelkező programok újraelhelyezését (átcímzését) is. A töltő különböző (direkt, dinamikus) kapcsolatszerkesztő eljárásokat is felölelhet, melyekkel különböző eredetű gépi nyelvű programokat egyetlen működő egészévé szervezhet. A fordítóprogram egyszerűsítését egyes programrendszerekben azáltal érik el, hogy bizonyos assembler, ill. compiler feladatokat is a töltőre hárítanak.

Interpreterek

Az interpretert az irodalom gyakran a compiler alternatívájaként jelöli meg. Az interpreter azonban nem fordító-, hanem végrehajtó program; nem készít tárgyprogramot — értelmezi a forrásprogram utasításait, és gépi utasítások generálásával közvetlenül végre is hajtja azokat. Az interpreterrel ellátott számítógép a kalkulátorra emlékeztető üzemmódban működik.

Olyan felhasználói környezetben, ahol egy-egy programot igen sokszor alkalmaznak, az interpreter megoldás lassú; célszerűbb a program gépi nyelvű alakjának használata, mivel ennek lefuttatása sokkal rövidebb időt vesz igénybe, mint a forrásprogram esetenkénti értelmezése. Másrészt egyszer vagy kévszer futtatott programok végrehajtására az interpreter lehet előnyösebb, tekintettel a compiler implementáció által megkövetelt előkészítő munka körülményességére és a fordítással járó idővesztésre. Az interpreter igen kényelmes abból a szempontból is, hogy használata esetén a forrásprogram javítása, módosítása igen könnyen végezhető. A compilerrel fordított program logikai hibái csak fordítás után, a tárgyprogram végrehajtása során derülnek ki, a forrásprogram módosítása esetleg rekompilálást tesz szükségessé. Esetenként, egyugyanazon gépen mindkétféle implementációt alkalmazzák, kihasználva sajátos előnyeiket. Ismeretes a kétféle implementáció összekapcsolása is, amikor a compiler a forrásprogramot nagyon egyszerű interpretatív nyelvre fordítja le, melyet azután interpreter értelmez, ill. hajt végre.

Az értelmező programok kezdetleges formájával assembly szinten is találkozhatunk (pl. CII 10010). Egy adott nyelven definiált szubrutinok ugyanazon nyelven írt programban a nekik megfeleltetett szimbólumokkal hivatkozhatók (hasonló módon, mint makrók esetében). Fordítás után a (zárt) szubrutinok ugyanúgy, mint a program, a memóriába kerülnek. A szubrutinokra a programban hivatkozó címeket egyszerű, memóriarezidens interpreter értelmezi, s ugyanő teremt kapcsolatot az önálló programrészek között.

Programszubrutin, ill. rendszerszubrutin különböztethető meg. Az előbbi a program keretén belül helyezkedik el, az utóbbi nem tartozik a programhoz, hozzá csak a rendszer szintjén csatlakoztatható. A rendszerszubrutin, mely mindig zárt, lehet memóriarezidens vagy tárolható a memórián kívül is.

A memóriarezidens szubrutin elhelyezése állandó, kezdő címe ismert, s e címen szokványos szubrutinhívó utasítással vagy interpretatív névvel aktivizálható. A szubrutin hívás sajátos alakja az egyes gépeknél megtalálható kváziutasítás. A kváziutasítás alakilag a gépi utasítással megegyezik, a programozó gépi utasítás-ként kezelheti, de műveletkódja nem mikroművelet-láncot definiál, hanem szubrutin memóriabeli kezdő címét.

Memóriarezidens szubrutinkönyvtárat, tekintettel a drága tárolóterületre és a ráírás veszélyére (memória védelemmel nem mindegyik gépkonfiguráció rendelkezik), a múltban általában nem alkalmaztak. A félvezető fixtárak mindkét gátoló tényezőt feloldják, s előtérbe léphetnek a rezidens szubrutintárolással járó előnyök: az egyszerű és gyors kezelhetőség.

A nem memóriarezidens szubrutinokat közbenső tárolóanyagon (lyukszalag, mágnesszalag, mágneslemez) helyezik el, és pedig abszolút töltési séma esetén forrásnyelven, relokációs töltési séma alkalmazása esetén pedig gépi nyelven.

Tartalmát illetően belső, ill. külső, továbbá készülékvonatkozású szubrutinról beszélhetünk. A belső szubrutinok a gép utasításkészletében nem szereplő műveleteket (kétszeres pontosságú, lebegőpontos számkör műveletei stb.) szimulálnak. A külső szubrutinok csoportjába egy bizonyos alkalmazási terület általánosan alkalmazott eljárásai (pl. gyökvonás, szögfüggvények stb.) tartoznak. A készülékvonatkozású szubrutinok különböző Be/Ki eljárásokat testesítenek meg.

Forrásszerkesztő (source editor, symbolic editor)

Új forrásprogram összeállítása során igen problematikus vagy egyáltalán lehetetlen korrekciót végezni, ha a program az írógépről egyenesen közbenső hordozóra kerül. Az ebből fakadó nehézségek (interaktív) forrásszerkesztő alkalmazásával kerülhetők el. A memóriában tárolt forrásszerkesztő az írógépről bemenetére érkező forrásprogramot saját adatterületén helyezi el, majd az írógépről kapott parancsok alapján a programot módosíthatja. A jóváhagyott forrásprogram, az összeállítást követően, a memóriából közbenső hordozóra vihető át. Hasonló módon történik régi programok módosítása is. A szerkesztővel az alábbi jellegzetes feladatok végeztethetők:

- új forrásprogram beolvasása,
- egy vagy több sor kinyomtatása,
- sor eltávolítása,
- sor beszúrása,
- módosított forrásprogram kivitele stb.

A szerkesztő a forrásprogramot annak nyelvi alakjától függetlenül szöveggént kezeli. A kezelésmód tekintetében azonban a különböző szerkesztők eltérhetnek egymástól. Egyesek pl. a szöveget számozott soronként olvassák be a memóriába, a sorokat számokkal identifikálják, míg mások a sorstruktúrát figyelmen kívül hagyják (dot and mark editor, context editor).

Hibátlanító programok

A kisszámítógép kódolt (bináris alakú) programjainak hibátlanítása (módosítása), elemi módon, a kezelőlap erre szolgáló eszközei és lehetőségei (kapcsoló és adatközlő regiszter, utasítások manuális szekventálhatósága, cím szerinti megállás stb.) révén végezhető. Hatásosabb eszközként különböző interaktív hibátlanító (debugging) programok állnak rendelkezésre.

A hibátlanító program az ellenőrzött programmal együtt foglal helyet a memóriában. A gépkezelő a hibátlanító programnak az írógépen át kiadott parancsai útján meghatározhatja az ellenőrzött program futásának menetét, megvizsgálhatja a program egyes utasításait, ill. egyes szakaszait, és módosításokat hajthat végre annak tartalmán.

A hibátlanító programoknak numerikus és szimbolikus változatai különböztethetők meg. A numerikus hibátlanító program használatakor az ellenőrzött programra oktális vagy hexadecimális reprezentációja alapján kell hivatkozni. Az eredményes munka alapja ez esetben a pontos lista.

A szimbolikus hibátlanítók az assembler által készített szimbólumtáblázatot is felhasználják. Így módon a felhasználó a programot mind gépi nyelvű, mind szimbolikus alakjában nyomon követheti, ill. hibátlaníthatja. A szimbolikus hibátlanító assembly nyelvre való visszafordítást is végezhet, ami adott esetben lényegesen megkönnyítheti a program áttekintését.

File-kezelő rutinok

A file-kezelő rutinok a file-t egységként kezelő, tartalmán processzálást nem végző különböző programok. Sorukba tartozik: a file-t adott hordozóról (pl. mágnesszalag) azonos típusú hordozóra, változtatás nélkül átmásoló kopírozó rutin; az adott közbenső hordozóról más típusú hordozóra átíró rutin; a valamely tár tartalmát nyomtatóra vivő szerkesztő rutin; a file adatformátumát átalakító file-konverter; a file átrendezésére szolgáló szortírozó rutin és más ehhez hasonlók.

A file-kezelő programokat utility rutinoknak is nevezik. E kifejezés tartalma azonban eléggé meghatározatlan, mivel az irodalom esetenként a szubrutinokat, a hibajavító és felderítő, valamint üzemellenőrző programokat is utility programként említi.

Egzekutív program

Az egzekutív, melyet gyakran felügyelő (supervisor) programnak, monitornak vagy szervező programnak is neveznek, esetenként változó összetételű rendszerprogram, mely feladatok automatikus lebonyolításával és rendszerkészletek felhasználásával kapcsolatos funkciókat lát el. Az egzekutívra az alábbi általános jellegű kötelességek hárulhatnak:

- készülékfeladatok végrehajtása,
- feladatok menetirányítása,
- táradminisztráció,
- hibafelügyelet,
- memóriavédelem.

Az egzekutív a processzor és a készülékek közötti adatátvitellel, az adatok pufferosásával, átrendezésével, előprocesszálásával, a megszakításkérés-kezeléssel kapcsolatos feladatokat a maga hatáskörében láthat el, felmentve az e feladatokkal kapcsolatos teendők alól mind a programozót, mind a programokat (device scheduling).

Az egzekutív az írógéptől kapott utasítások vagy benne rögzített algoritmusok alapján ellátja bizonyos egymás után következő (típus) feladatok menetirányítását (job scheduling). A felhasználói programban elhelyezett parancsok alapján is végezhet előírt feladatokat.

Az egzekutív esetenként ellátja a memória—háttértár együttes adminisztrációját. Hatásköre az egyszerű programcseréltéstől (swapping) a többprogramú, ill. időosztásos rendszereken belüli memóriaterület hozzárendeléséig terjedhet.

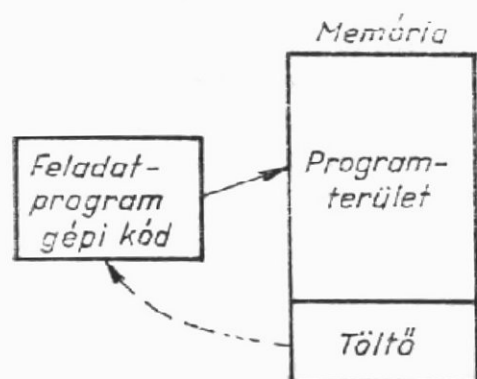
Az egzekutív felügyelő rutinja a programvégrehajtás menetében lehetséges hibaállapotokra felügyel (védett memóriaterület érintése, végtelen hurok. stb.). Hiba észlelése esetén különböző, előírt eljárásokat folytat le a hiba elhárítása vagy okának felderítése érdekében (error handling).

Az egzekutív fontos feladata, hogy önmagát védje a felhasználó tévedései ellen, több felhasználó esetében pedig, hogy kölcsönösen védje a felhasználókat egymás programjainak megkárosításától.

6.1.3. Működtetésmódok

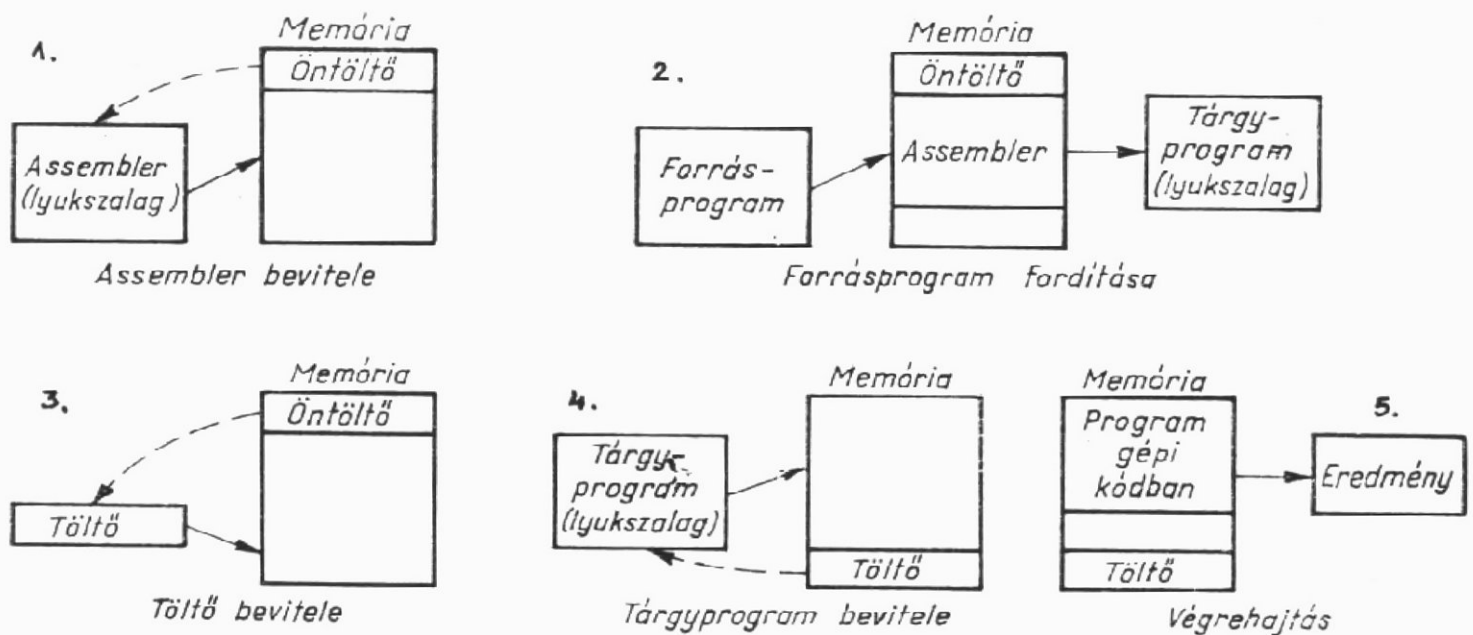
Manuálisan működtetett rendszerek

A legegyszerűbb, nem önálló programrendszer: egyetlen, memóriarezidens töltő (6.1. ábra). A gép ennek birtokában csupán az adott feladattal kapcsolatos, minden teendőt (pl. szubrutinok) magába foglaló, kódolt programokat képes végrehajtani. Nem önálló programrendszer akkor használható, ha kész alkalmazási programcsomagok vagy egy más, programelőkészítésre alkalmas gép szolgáltatása áll rendelkezésre.



6.1. ábra.
Nem önálló programrendszer

Az önálló programrendszer kezdetleges alakját az egzekutív programok hiánya jellemzi, s bár a programelőkészítéshez és futtatáshoz szükséges minden rendszerprogram rendelkezésre áll (pl. lyukszalagon), e rendszerprogramokat manuálisan kell sorba állítani és működtetni (6.2. ábra). A rendszer használhatóságának egyik fontos szerkezeti követelménye, hogy (a fordításhoz) legalább egy bemeneti és egy kimeneti készülék álljon rendelkezésre. A konvencionális minimális konfiguráció esetében mindkét készülék szerepkörét a lyukszalagolvasóval és -lyukasztóval rendelkező írógép látja el. A programelőkészítés ideje gyors lyukszalagegységek alkalmazásával csökken. E rendszer használata megfelelő színvonalú alkalmazói környezetben, és ahol gyakran kell új és új programokat előkészíteni, még akkor is előnyös lehet, ha egyébként számítóközpont szolgáltatásai állnának rendelkezésre.



6.2. ábra.
Egyszerű önálló programrendszer részei és működtetésének menete

Készülékegzekutív

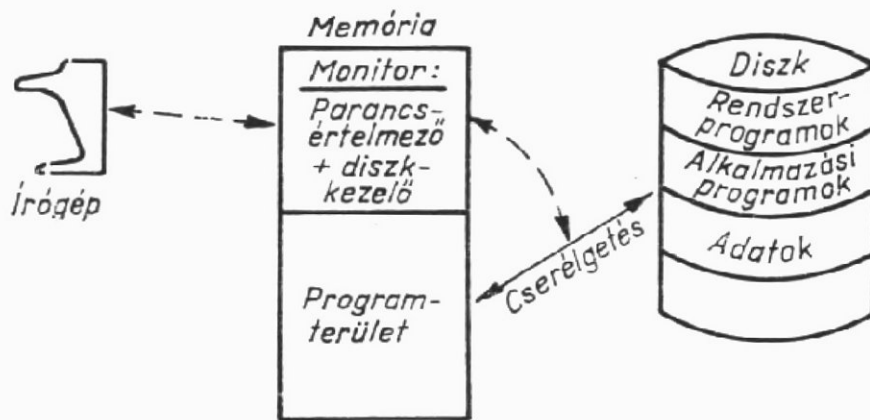
A programrendszer automatizált működtetésének egyik lehetősége a készülékrutinok egységesítése, s egységes programcsatlakozással rendelkező egzekutívra való összefogása. A készülékfeladatok struktúráját tekintve, azon belül különböző hierarchikus szinteket különböztethetünk meg. A legalsó szinten a készülékek működtetésével s a megszakításkérés-kezeléssel járó alapvető feladatokat, valamint az ezekhez kapcsolódó mellékfeladatokat (állapot- és átviteellenőrzés, hibaállapotokkal kapcsolatban előírt eljárások) találjuk. A következő szinten a fizikai adatblokk átvitelével járó adminisztratív feladatok (beleértve a pufferkezelést) találhatóak. A legfelső feladatszinten pedig a fizikai adatblokknak a processzor számára logikailag értelmezhető alakba való konvertálásával, előkészítésével kapcsolatos (és fordított irányú) teendők helyezkednek el. Attól függően, hogy e feladatszintek közül hányat foglal magában, a készülékegzekutívna is több változata különböztethető meg; az egyszerű készülékkezelőtől (device handler) a mindhárom feladatszintre kiterjedő készülékintézőig (device scheduler) terjed a skála.

Diszk-operációs rendszerek

A rendszerprogramok és az alkalmazási programcsomagok a manuálisan működtetett programrendszerben lyukszalagon vagy mágneskazettában tárolódnak. Minőségileg új helyzet áll elő, ha a rendszerprogramok tárolására közvetlen elérésű tárat, a memória és a tár közötti forgalom lebonyolítására, általában a programrendszer működtetésére egzekutív programot alkalmaznak. Háttértárként diszk vagy dob, de közvetlenül címezhető szalagtár is alkalmazható; legáltalánosabb a lemez (diszk), minek alapján az ilyen jellegű programrendszert diszk operációs rendszernek nevezik.

A legegyszerűbb diszk operációs egzekutív a diszk-monitor. A diszk-monitor lényegében a manuálisan működtetett rendszerben az ember által ellátott adminisztratív, feladatszekventálási munkakört veszi át. A gépkezelőtől az írógépen át kapott parancsok alapján, a diszkkal kapcsolatba lépve cserélgeti (swapping) a memória programjait. A diszken a rendszerprogramok mellett alkalmazási programcsomagok

és adatok is tárolódhatnak. A monitor memóriarezidens része parancsértelmezőből és diszk-kezelőből áll. A diszk-kezelőt mind a rendszerprogramok, mind az alkalmazási programok aktivizálhatják, de a diszk kivételével minden más készülék kezelése a rendszeren (monitoron) kívül, az egyes programokon belül történik (6.3. ábra).



6.3. ábra.
Diszk-monitorral működtetett programrendszer

Példaként a PDP-8 család diszk-monitorral működtetett programrendszere említendő meg, melyhez forrászerkesztő, assembler, FORTRAN compiler, szimbolikus és oktális hibátlanító, továbbá különféle file-kezelő rutinok tartoznak. Ehhez járul még egy monitorgeneráló program is, mellyel a nyitottan moduláris architektúrájú monitor a mindenkori szerkezeti konfigurációhoz illeszthető.

Az általános diszk-operációs rendszerek az egyszerű monitortól mindenekelőtt a számítógéprendszer összes készülékére kiterjedő készülékintézőben különböznek, mely egységes készülékkezelést biztosít. A program a készülékintézőhöz három különböző szinten fordulhat:

- a közvetlen elérési szinten egyenesen a készülékkezelőhöz,
- a fizikai szinten, ahol meghatározott puffert érintő blokkátvitel igényelhető,
- logikai szinten, ahol direkt file-kezelés (hívás) lehetséges.

A dinamikus készülék (file) hozzárendelés egyidejű többszörös file-elérést tesz lehetővé, ami akkor előnyös, ha valamelyik feladatprogram egy bizonyos file-t bemenetként használ, ugyanakkor egy másikat kimenetként ír ki. A dinamikus hozzárendelés lényege az, hogy a program nem valóságos készüléket (file-t), hanem szimbolikus azonosítóval pszeudokészüléket (file-t) szólít meg. A szimbolikus azonosítóhoz tényleges file vagy készülékcím hozzárendelését külön szubrutin a program futtatása előtt, vagy maga a program végezheti.

A fejlettebb diszk-operációs rendszerek relokációs töltési sémával, kapcsolatszerkesztővel és természetesen mind a már korábban felsorolt lehetőséggel is rendelkeznek.

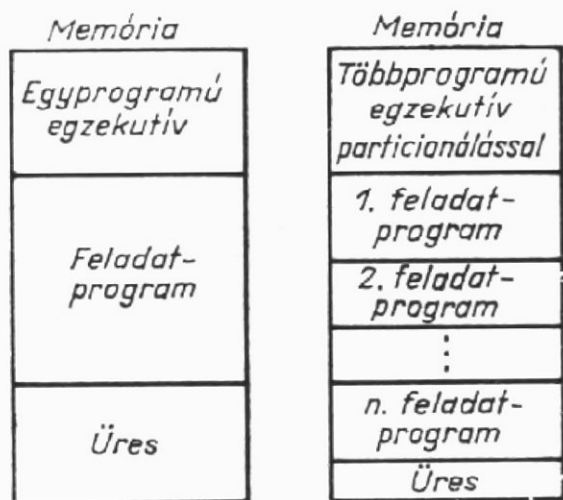
Többprogramú operációs rendszerek

A fentiekben ismertetett operációs rendszerek közös jellemzője, hogy a memóriában egyidejűleg csak egyetlen feladatprogram tartózkodik, csak egyetlen feladatprogram aktív (6.4. ábra). Ezeket egyprogramú vagy egyfeladatú (single job, single task) operációs rendszernek nevezik. A feladat addig köti le a központi egységet, míg le nem zárul, vagy esetleg előálló hiba miatt félbe nem szakad. Az egyprogramú rendszer egyrészt időkihasználás szempontjából lehet előnytelen, ha pl. a program olyan adatokon végez műveleteket, melyek beérkezési ciklusa (a készülék lassúsága

miatt) hosszabb, mint az adatok processzási ideje. Másrészt a készletek kihasználása is rossz, hiszen gyakran marad üres memóriaterület, a várakozási időszakokban pedig a processzor tétlen. E negatívumok kiküszöbölése érdekében alkalmazzák a különböző többprogramú (multiprogramming) operációs rendszereket. Kisgépek esetében a relatíve csekély szerkezeti árak folytán a készletkihasználás problémája háttérbe szorul, ezzel szemben a gépet gyakran kell olyan feladatkörben alkalmazni, mely megköveteli számos program egyidejű munkaállapotát, számos program szimultán futtatását.

A többprogramú rendszer programjainak munkaállapotán azt értjük, hogy a programok (vagy legalábbis mindegyikükből egy-egy rész) egyidejűleg a memóriában tartózkodnak, és végrehajtásuk folyamatban van. Természetesen, az idő egy elhatárolt részében több program közül mindig csak egy aktív, s bizonyos értelemben a programok mindegyike független a többitől. A programok menetirányítását ellátó egzekutívot ez esetben többprogramú felügyelőnek (multiprogramming supervisor) nevezik. A felügyelőnek a menetirányítást közvetlenül ellátó része: a KE (központi egység) diszpécser. Bizonyos prioritási rend alapján a KE diszpécser végzi a KE időbeli hozzárendelését az egyes munkaállapotú programokhoz.

A többprogramú operációs rendszer az alábbiakban körülírt módon funkcionál. A momentán aktív program a soron következő készülékmuveletig fut. Amikor a program készülékkihívással fordul a felügyelőhöz, egyúttal felfüggesztődik mindaddig, míg a megindított készülékfolyamat végig nem fut. A közbenső időre a fel-



6.4. ábra. Egy- és többprogramú operációs rendszer

ügyelő a menetirányítási algoritmusban meghatározott legmagasabb rangú programot (folyamatot) aktivizálja, mely aztán készülékfolyamat befejeztét jelző megszakítás-kérésig fut. A megszakítási rutin a készülékfolyamatot elindító programot a prioritási rangsorolásnak megfelelő helyre, a várakozó programok közé visszasorolja.

Amennyiben adott rutint egynél több program igényel, a tiszta eljárás (újrakezdhetőség) feltételét kielégítő gépi architektúra esetén ez igények a rutinnak egyetlen, a memóriában tartózkodó kópiájával kielégíthetők.

A többprogramú operációs rendszer felügyelőprogramja az egyedül főmemóriát tartalmazó gépi környezetben, egyedül készülékintézőből és KE diszpécserből áll. A helyzet lényegesen bonyolultabb, ha a számítógéphez háttértár is járul, amikor is a felügyelőnek a programcserélgetéssel járó feladatokat is el kell látnia. A memóriaterület felosztására és a különböző programokhoz való hozzárendelésére, a munkaállapotú és a háttértárban esetleg várakozó programok várakozási sorba állítására, különböző rendszerfilozófiák alakultak ki. Többprogramú rendszer esetében az

operációs rendszernek biztosítani kell a programok egymást károsításának lehetőségét kizáró védelmet és a különböző programok közötti kapcsolatfelvétel lehetőségét is.

Időreális operációs rendszerek

Tágabb értelemben véve, minden számítógép-tevékenység végrehajtási ideje korlátos, melyet túllépve a tevékenység elveszíti realitását. A feladatok végrehajtási idejének mindig összhangban kell állnia azon folyamat egészének időléptékével, melynek a feladat részét képezi. Bizonyos alkalmazási területeken (különösen a közvetlen kapcsolatú on-line mérés technikában) az időbeli megkötöttség nem csak a feladat egészére, hanem a részfeladatokra vonatkozóan is igen szigorú; a feladatnak előírt időszerkezetben kell lefutnia. Az ilyen részletes időbeli előírásoknak eleget tevő (és gyakran órával ellenőrzött) végrehajtásmódot *időreálisnak*, az időrealitás követelményeinek eleget tevő végrehajtó rendszert időreális operációs rendszernek, az operációs rendszer végrehajtó programját időreális egzekutívnak nevezik.

Az időreális feladattípusok igen diszperz képet mutatnak; a gép szerkezeti és rendszerprogram konfigurációja tekintetében csaknem minden alkalmazási esetben más-más igény vetődik fel. Az időreális egzekutívot — időbeli vonásain túl — az tünteti ki, hogy tulajdonképpen csak keret, melyet esetenként az egyes alkalmazási igényeket kielégítő, a felhasználó (vagy megbízottja) által kifejlesztett speciális rendszerprogramokkal kell kiegészíteni.

Az időreális terület további jellegzetessége az alkalmazási programok sajátos struktúrája. A feladat általában számos elkülönülő részfeladatra osztható fel, melyek végrehajtását előírt prioritási (és időbeli) sémának megfelelően az egzekutív kell, hogy koordinálja. Egzekutív által, ill. szerkezeti megszakításkérés által kezdeményezett feladatok különböztethetők meg. Amennyiben nincs folyamatban levő megszakítási program, sem várakozó megszakításkérés, az egzekutív feladat-menet-irányítója (task scheduler) mindig a legmagasabb prioritásrangú, nem megszakítás jellegű feladat végrehajtását indítja.

Az egzekutívnak mint a végrehajtás keretének az alkalmazási programok részére általában a következő lehetőségeket kell biztosítani:

- mindkét feladattípust (megszakítási és nem megszakítási) képviselő programok hívhassanak nem megszakítás jellegű rutint;
- bármely program igényelhesse, hogy egy másik program adott óraidőben vagy általa specifikált időintervallum múlva végrehajtsassék;
- feladat, feladatok sorának ciklikus újravégrehajtása;
- adott program adott készüléket kizárólagos használatra igényelhessen, más szóval adott program adott készülék megszakítási rutinjává rendelhesse önmagát;
- a program meghatározatlan időre felfüggeszthesse önmagát, vagy speciális, megszakításra várakozó állapotba léphessen.

Az időreális operációs rendszer egyszerűbb változatainak egzekutívja memóriarezidens. A kiterjedtebb rendszerek rendelkeznek a többprogramú diszk-operációs rendszerek adottságaival, de azzal a módosítással, hogy míg bizonyos (háttér) programok végrehajtásmódja konvencionális, más programoké az időreális végrehajtási sémát követi.

Kiegészítés

A programrendszerek feladat-előkészítés, ill. feladat-végrehajtás rendeltetésűek. Az előbbiek (ide tartoznak a diszk-operációs rendszerek) egzekutívjai az alkalmazási feladatprogram összeállításával, fordításával és hibátlanításával kapcsolatos teendők algoritmusait ölelik fel, míg az utóbbiakban (időreális rendszerek) a feladat lebonyolításával kapcsolatos teendők algoritmusai dominálnak. A kisgép előkészítés rendeltetésű programrendszere nélkülözhető olyan környezetben, ahol számítóközpont-szerű szolgáltatás áll rendelkezésre. Az időreális rendszereknek annak ellenére, hogy alapfeladatuk nem programelőkészítés, mégis rendelkezniük kell bizonyos, a programfejlesztés szempontjából fontos olyan adottságokkal, melyek igénybevételével a feladatprogramok közvetlen kapcsolatú (on-line) kipróbálása, korrigálása, végleges formába öntése elvégezhető.

Az általános célra kidolgozott programrendszerek gyakran igen kevésbé használhatók a kisszámítógép alkalmazásának speciális eseteiben. A felhasználónak a meglévőt ki kell egészítenie, vagy új programrendszert kell létrehoznia. A kiterjedt és bonyolult struktúrájú programrendszerek fejlesztése igen nívós programozási praxis birtokában is meglehetősen sok időt igényel és jelentős anyagi áldozattal jár, különösen ha e terheket egyetlen rendszernek kell viselnie. A programrendszerek tervezésének és létrehozásának automatizálása a kisgépek szempontjából igen nagyjelentőségű már csak azért is, mivel a szerkezeti költségek hosszú idő óta állandóan csökkennek. A probléma egy megoldásmódját jelentik a rendszerprogramok készítésére kifejlesztett és alkalmazott ún. implementációs nyelvek, melyeken compilerek és egész operációs rendszerek írhatók, s melyek ugyanúgy, mint a kereszt-asmblerek és compilerek, nagygépen futtathatók.

6.2. A mérőrendszer programellátása

A kisgép, a 6.1. pontban leírt általános jellegű rendszerprogramok birtokában, mérés-technikai irányítási feladatok ellátására jó hatásfokkal nem használható. A gép alkalmazásával járó lehetőségek optimális kiaknázása érdekében, a programrendszert a mérési feladat követelményeihez és a mérőrendszer szerkezeti adottságaihoz egyaránt illeszteni kell. Az ily módon adaptált programrendszer az általánostól célra orientált nyelvvel, nyelvi implementációjának és operációs rendszerének sajátosságaival általában egyaránt eltér.

6.2.1. A programozási nyelv a mérés-technikában

Helyzetáttekintés

Valamely alkalmazási terület igényeit kielégítő egységes nyelv kialakításának előfeltétele egyrészt a feladatstruktúra konzolidáltsága, másrészt a nyelv felhasználói, gépi környezetének predeterminálhatósága. A mérés-technikát, de a mérési eljárások automatizálásának metodikáját is, hosszú idő óta — és várhatóan még hosszú időn át — a fejlődés, változás állapota jellemzi; viszonylagos stabilitás csupán egy-egy részterületen érzékelhető. Ebből következően kialakulatlan a feladatstruktúra is. A programozási nyelvvel kapcsolatos általános megfontolások síkján predeterminált gépi környezettel nem lehet számolni; a rendszerkonfigurációt legalább részben, az esetenként változó technikai és gazdasági követelmények határozzák meg.

A mérés-technika vezérlési és számítástechnikai igényeinek összességét kielégítő mérés-technikai nyelv kialakítása még stabil feladatstruktúra mellett sem volna kisebb feladat, mint akár a FORTRAN vagy pl. a PL/1 létrehozása jelentett. Ezt a gyakorlat is bizonyítja. A tapasztalat másrészt azt mutatja, hogy „nagy”, általános nyelv csupán nagygépen vagy kis/középgép nagy konfigurációján implementálható szabatosan. A mérés-technikát azonban a kisgépek kis konfigurációi uralják, s ha volna is egységes, az egész mérés-technikára kiterjedő nyelv, alkalmazása e rendszereken ugyanolyan nehézségekbe ütközhetnék, mint amelyeket az ALGOL, FORTRAN és hasonló nyelvekkel kapcsolatban a 6.1. pontban már ecseteltük.

Feltételezve, hogy volna általános nyelv, s hogy a szabatos implementációjához szükséges gépi környezet is rendelkezésre állna, alkalmazásának további komoly akadályát jelentené az egyes mérőrendszerek individuális jellege. A számítástechnika magasabb szintű nyelveinek széles körű használatát nemcsak e nyelvek feladatleírásra való alkalmassága teszi lehetővé, hanem az is, hogy a nyelvek implementációja a számítógéppel azonnal rendelkezésre áll, az implementációt felölelő programrendszert a gép gyártója dolgozza ki, és az ezzel kapcsolatos költségek a gép árában mérsékelt tételként szerepelnek. A mérőrendszerek esetében a gyártó a programrendszer kidolgozásához szükséges paramétereknek nincs birtokában. E paraméterek a mérőrendszer tervezésének folyamatában fogalmazódnak meg, s ha a számítógéppel nyújtott rendszerprogram elemek részben fehasználhatók is, a programrendszer egészét a rendszertervezőnek kell kidolgoznia. A költségek így egy vagy legfeljebb néhány (tucat) mérőrendszerre oszlanak el. Jelentkezik továbbá a programrendszer megalkotásával kapcsolatos időtényező is. Mármost minél általánosabb nyelvet kell a rendszertervezőnek alapul vennie, annál (talán aránytalanul is) nagyobb feladatot jelent a programrendszer megalkotása, s annál kiterjedtebb a szabatos implementációnak megfelelő gépi konfiguráció is. Az így összegeződő hátrányokat az egységes nyelv használatából származó előnyök hozzávetőlegesen sem egyenlíthetik ki.

Az áttekintett körülmények folytán a mérés-technikában nem alakultak ki olyan széles körben elfogadott és használt nyelvek, mint a számítástechnikában; csaknem minden egyes mérőrendszer-családnak, sőt gyakran mérőrendszernek is, saját nyelve van. Bizonyos nemzetközi elismerés szintjére is csupán az egyetlen, erősen korlátozott rendeltetésű ATLAS emelkedett. Implementációs nehézségei miatt azonban ezt is egyrészt inkább feladatleíró nyelvként, másrészt különböző leszűkített nyelvi változatok és dialektusok implementációjának bázisaként, semmint általános programozási nyelvként alkalmazzák.

Nyelvi szint és feladatapertúra

A mérés-technika gyakorlatában négy különböző nyelvi szinttel találkozhatunk. Alkalmazznak

- mérésorientált,
- felsőbb szintű rendszerorientált,
- rendszer gépi,
- számítógép assembly

nyelveket.

A legmagasabb szintű mérésorientált vagy más néven tárgyorientált nyelv a mérési algoritmusokat a vizsgálat, és a vizsgálat tárgyának a mérőrendszertől független terminológiájával írja le.

A felsőbb szintű rendszerorientált (blokkorientált) nyelv a konkrét mérőrendszer nagyobb építőegységeire, blokkjaira való hivatkozásokat is tartalmaz; a mérési algoritmusok leírásánál a mérőrendszer fogalmaival is él.

A rendszer gépi nyelvén a mérőhálózat összes funkcionális eleme hivatkozható, az e nyelven írott program az adott mérésben résztvevő összes hálózati elemnek szóló előírásokat tartalmazza.

A számítógép assembly nyelve ugyancsak mérőrendszer-független, míg azonban a mérésorientált nyelv esetében a programírás során a mérőrendszer elvileg teljesen figyelmen kívül hagyható, assembly nyelv használatakor nemcsak a mérőhálózat, hanem a mérőhálózat és a számítógép belső, funkcionális összefüggéseinek ismerete is elengedhetetlen.

Szemléltetésül a 6.1. táblázat a mérés egy mozzanatát három különböző szinten (az assembly-től eltekintünk) írja le.

A nyelv másik fontos meghatározója: feladatapertúrája, az, hogy érvényessége a feladatok milyen körére terjed ki. Alkalmas-e pl. analóg és digitális mérések leírására egyaránt vagy csak az egyikére? s ezen belül vajon általánosan, vagy esetleg csak leszűkített értelemben használható?

A nyelv annál nehezebben implementálható, minél magasabb szintű és minél szélesebb apertúrájú. Annál könnyebben alkalmazható, minél magasabb szintű és minél szűkebb apertúrájú.

6.1. táblázat

Mérési feladat leírása a VITAL nyelv három különböző szintű dialektusán

Feladat: Alkamazzunk 5 KHz-cel 20%-ra modulált 1Veff. 1MHz szinuszos vivőhullámot a tárgy J—5 kapcsán; J—6 földelve; terhelőimpedancia 50 ohm		
TOL (mérésorientált)	BBOL (blokkorientált)	DCOL (rendszer gépi nyelv)
1. APPLY, AM SIGNAL, 1V, 1MHZ, MOD 20 PC, MOD—F 5KHZ, 50 OHM, CNX HI J—5\$	1. APPLY, BB20—1, 1.8 V, 5 KHZ, OFF, INT, HI (BB21, MIHA) \$ 2. APPLY, BB21, 1 V, 1 MHZ, MOD 20PC, CNX HI J—5 MIHA (BB20—1, HI) \$	1. SET UP DC20—1, 1.8 V, 5KHZ OFF, INT \$ 2. SET UP DC21, 1 V, 1 MHZ, MOD 20 PC \$ 3. CLOSE, BB01, FST, J16—FF 2J25—L \$ 4. CLOSE, BB01, SLW, J3B05T, G5—4 \$ 5. CLOSE, BB20—1, HI \$ 6. CLOSE, BB21, MIHA \$ 7. CLOSE, BB21, HI \$

A nyelv megválasztásának és értékelésének szempontjai

Egységes nyelv és implementációjának hiányában a rendszertervező, a felhasználói követelményekkel összhangban, különböző szempontok optimális kompromisszumaként saját maga kell, hogy megoldást találjon. A konkrét körülményektől függően az optimum esetenként más-más nyelvi megoldás választásával érthető el. Olyan abszolút kritérium, melynek alapján a konkrét körülményektől függetlenül bármely lehetséges nyelvet minősíteni lehetne — nincs. Amennyiben pl. a rendszertervező a felhasználóval azonos gazdasági keretbe tartozik, a nyelv megválasztása kapcsán a tervezési és alkalmazási költségek együttes minimumára kell törekednie, a nyelv implementációjának és alkalmazásának együttes költségeit kell minimumra szorítania. Figyelembe kell vennie, hogy az implementáció költségei a nyelv szintjével és feladatapertúrájának tágításával emelkednek, az alkalmazási programok megírásának

költségei viszont a nyelv szintjének emelésével, apertúrájának szűkítésével csökkennek. A viszonylag stabil összetételű alkalmazási programokkal dolgozó rendszerek esetében, ahol az igények hosszú időn át használt alkalmazási programcsomaggal elégíthetők ki, a programírás költségei és nehézségei háttérbe szorulnak az alternatívaként választható magasabb szintű nyelvek implementációs költségei mellett, s megeshet, hogy az assembly nyelv jelent optimális megoldást. Ezzel ellentétben, ha a mérőrendszer olyan alkalmazási környezetben kerül felhasználásra, ahol minduntalan új és új programokat kell írni, s emellett a programozási és rendszerismeret színvonala is alacsony, a magasabb szintű nyelv használatából származó gazdasági és időrealitási előnyök kiegyenlíthetik az igen jelentős implementációs költségeket.

Egészen más helyzettel áll szemben a független rendszertervező—gyártó, aki berendezését nem egyetlen felhasználó, egyetlen feladatkör, hanem valamely mérés-technikai terület átfogóbb igényeinek figyelembevételével hozza létre. Szükségképpen szélesítenie kell a nyelv feladatapertúráját, de a közérthetőség érdekében megfelelően magas szintű nyelvet is kell választania. Az apertúra szélesítése, a nyelv szintjének emelése kiterjeszti a berendezés felhasználási körét, s ezáltal növeli az eladható berendezések számát, növeli azonban az implementáció költségét is. Nyilvánvaló, hogy ez esetben nem az implementáció teljes, abszolút költsége a mérvadó, hanem az a nyelvi szint és apertúra, mely mellett az egy berendezésre jutó költség a lehető legkisebb. A rendszeroptimumot itt tehát más tényezők szabják meg, mint az előbbi esetben.

Tekintettel arra, hogy az önálló rendszertervező—gyártó vállalkozó által előállított, meghatározott feladatkörre szánt berendezés a konkrét felhasználó igényeitől függetlenül jön létre, tervbe vett alkalmazása esetén egész programrendszerének, s ezen belül a nyelv alkalmasságának a felhasználó szükségleteivel való gondos összevetése elengedhetetlen.

A nyelvi probléma megoldásának különböző irányzatai

A mérés-technika nyelvi problémáinak megoldását különböző irányokból és különböző módszerekkel közelítik meg.

A mérésorientált megközelítés a mérési eljárások szemléletes, s bár rövidítéseket használó, de a napi mérés-technika fogalomkörének megfelelő természetes nyelvi leírásmódjára törekszik. Nem véletlen, hogy e megközelítésmód a repülőgépiparban alakult ki (ATLAS). A repülőgéphez számos, egymástól független alkattelemgyártó járul hozzá. Alapvetően fontos tehát a specifikáció és ellenőrzési eljárásának egységessége és egyértelműsége, az ellenőrzés végrehajtásmódjától független leíró nyelv. Az e nyelven megadott programnak elvileg tetszés szerinti módon, tehát akár manuálisan, akár számítógéppel irányított mérőrendszeren végrehajthatónak kell lennie. Az előzőekben már szó volt azokról a nehézségekről, amelyekkel az ilyen jellegű nyelvek számítógépes implementációja együtt jár. Meg kell azonban külön említeni egy további problémát is: a rendszertől függetlenül megalkotott nyelv, a dolgok természetéből következően, a leggondosabb implementáció esetén sem nyújthat a rendszer- vagy számítógéporientált nyelvekhez hasonló flexibilitást. Átfogó utasításai a rendszer elemeit meghatározott összefüggésben működtetik, de a rendszer elemeihez ezen túl a nyelv nem fér hozzá, és így bizonyos — esetenként szükséges — rendszerfunkciók a nyelvvel nem aktivizálhatók.

Azokon a területeken, melyeken az egységes mérésorientált nyelv használatáról nem lehet lemondani, az implementációval kapcsolatban felsorolt nehézségeket oly módon hidalják át, hogy a nyelvnek az adott felhasználói kör igényeire szűkített részalmazát, esetleg valamely dialektusát adaptálják a konkrét mérőrendszerhez.

A flexibilitás problémáját pedig az adaptált nyelv keretében létrehozott rendszerorientált nyelv, esetleg nyelvek alkalmazásával oldják meg (többszintű nyelv).

A rendszertervezőknek gyakran csupán szűkebb felhasználói kör igényeivel kell számolniuk. Ilyenkor az előbb bemutatott út fordítottját is követhetik. Nem az általánosból származtatják le a konkrétat; ellenkezőleg, a nyelv kialakítása során (mely ez esetben használat közbeni időbeli folyamat) rendszer gépi nyelvi szintről indulnak el, s a praxis alapján alakítanak ki bizonyos típusrutinokat összefogó magasabb szintű utasításokat.

A nyelvi probléma megoldásának harmadik irányzata a számítástechnikában alkalmazott felsőbb szintű nyelvekből indul ki. E nyelvek alkalmazása vezérlési feladatokra nehézkes, s főként nem szemléletes. Kielégítő eredmények érhetők el azonban mérés- (mérőrendszer) orientált dialektusok (a leszűkített nyelv kiegészítése bizonyos mérés-technikai feladat szempontjából lényeges utasításokkal) alkalmazásával. Ilyen jellegű nyelvek használata különösen akkor bizonyulhat előnyösnek, ha a számítógéppel nem csak vezérlési feladatokat, hanem a mért adatokon nagyobb mennyiségű számítást is kell végeztetni.

6.2.2. Alkalmazott nyelvek

ATLAS

A mérésorientált, pseudo-angol ATLAS (Abbreviated Test Language for Avionics Systems) nyelvet az Aeronautical Radio, Inc. (ARINC) Airlines Electronic Engineering Committee (AEEC) széleskörű együttműködés keretében alakította ki, a polgári repülőgépgyártás és karbantartás céljára, mely téren nemzetközi (legalább is Nyugat-Európai és USA) standardként szerepel. Elsősorban repüléstechnikai analóg készülékek (többek között pl. vivőhullámú rádiókészülékek) ellenőrzési algoritmusainak leírására szolgál, de alkalmas a repüléstechnikán kívül eső területek ellenőrzési feladatának leírására is. A digitális mérés-technika szükségleteit eredeti alakjában kevésbé veszi figyelembe; az AEEC munkacsoport a nyelvnek a digitális technikára való kiterjesztését tervbe vette. Főbb jellemzői az alábbiakban összegezhetők:

Angol nyelv származékú mnemonikai rövidítéseket használ (pl. ERRLMT).

Mérőrendszer független; manuális vagy gépi irányítású mérőrendszerek feladatainak megfogalmazására egyaránt alkalmazható.

Az utasítások nyelvi elemeinek elválasztására főként írásjeleket alkalmaz.

Teljességében vagy adaptált (részhalmaz, subset) alakjában egyaránt alkalmazható.

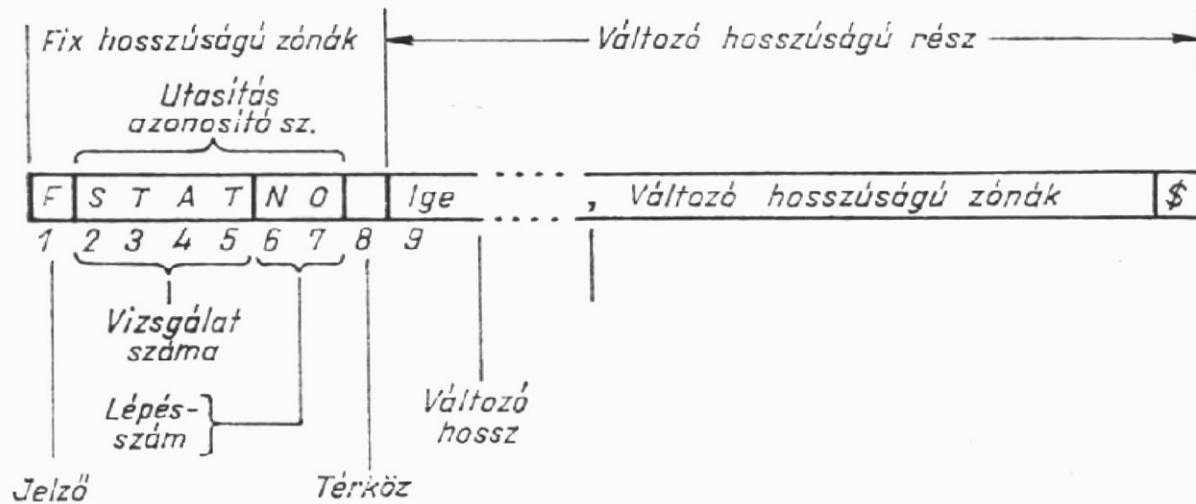
Az ATLAS szótár 238 nyelvi elemet definiál. Ezek közül 42 ige, 35 főnév, 80 értelmző (modifier), 14 kapcsolatmeghatározó nyelvi elem és 67 különböző utasítás-jellemző (jelzők, írásjelek, operátorok és bizonyos korlátolt alkalmazáskörű szavak). A karakterkészlet az angol abc A...Z betűit, a 0...9 számjegyet és különböző írásjeleket * , . = + - \$ / () ' betűköz ölel fel, melyek teljessége az ASC II karakterkészlet részhalmazát képezi. Az ATLAS a karakterek bináris ekvivalenseire nem tesz kikötést, de legtöbb esetben célszerűen az ASC II kód használható.

Az ATLAS specifikáció tartalmazza a decimális, hexadecimális, oktális, bináris és komplex számok, valamint az alfanumerikus karaktersorozatok formátumainak leírását.

A nyelv ' jelek közé helyezett egyedi szavakat címkeként alkalmaz. A címkével definiált információ az utasításon belül elhelyezett címkével hivatkozható.

A változó hosszúságú utasítások rögzített és kötetlen hosszúságú zónákból épülnek fel. Nem kötelező, hogy az utasítás egyetlen sorban elférjen; bármely funkcionális egység után betűközt iktatva, az utasítás a következő sorban folytatható,

és az átvitel többször is megismételhető. Valamennyi utasítás egykarakteres jelző zónával kezdődik, ezt követi a hatkarakteres azonosító zóna, majd a változó hosszúságú igezóna. Az utasításokat \$ jel zárja. Az igezóna és a \$ között, változó hosszúságú zónák helyezkednek el. A jelzőt az azonosítótól sem írásjel, sem térköz nem választja el; az azonosító és az ige között egy betűköz van, az ige a többi zónától és a többi zónát egymástól vessző határolja el (6.5. ábra).



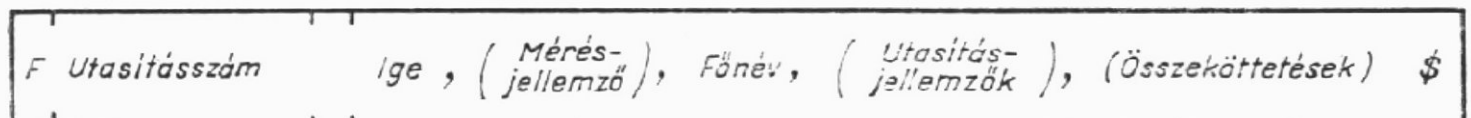
6.5. ábra.
Az ATLAS nyelv utasításainak általános szerkezete

A jelző zónában elhelyezett, meghatározott értelmű betűk bizonyos szempontból, jellemzik az utasítást, a jelző zóna azonban az esetek többségében üres marad. C, E, B, S és M jelzők különböztethetők meg. C a compilernek szóló utasítást jelöli, melyet nem kell lefordítani. E a gépkezelő lehetséges beavatkozási pontjait mutatja. A B-t követő és \$ jellel végződő szöveget a compiler megjegyzésként kezeli, melyet nem fordít le. Az S-sel jelölt utasítást a compiler a fordítás befejeztéig megőrzi. Az M érzékelésre (mérésre) vonatkozó utasítás előtt állhat, s ott azt jelöli, hogy az utasításban meghatározott érzékelési funkciót addig kell fenntartani, míg valamely (REMOVE, OPEN vagy DISCONNECT) leválasztó utasítás végrehajtására nem kerül sor.

Az azonosító zónában elhelyezkedő hatjegyű szám az utasítás identifikálására és programon belüli hivatkozásra szolgál.

A változó hosszúságú ige zóna: az utasítás műveleti zónája — az ige: a művelet megnevezése. Az ATLAS igék jelentését vázlatosan a 6.2. táblázat foglalja össze.

Az ige és a \$ zárójel között az utasítás jellegétől függő, különböző számú és jelentésű, változó hosszúságú zóna található. A vevőtípusú utasításban pl. (6.6. ábra) az ATLAS szótárhoz tartozó valamely főnév a mérés nemét (pl. AC SIGNAL, AM SIGNAL, DC SIGNAL stb.) adja meg, az ige és a főnév közötti zónában pedig az ATLAS szótár által a főnévhez rendelt modifikátorok (l. 6.3. táblázat) egyike kerül, mérésjellemző minőségében. Szemléltetésül: MEASURE, (FREQ), AC SIGNAL jelentése; mérd a váltakozóáram frekvenciáját! A vevőtípusú utasításában továbbá a főnevet az öt kiegészítő utasításjellemzők követik; minden utasításjellemzőnek saját, határolójelekkel (vessző) elkülönített zónája van. Az utasítás legvégén, a \$ előtt, az összeköttetések definíciója áll.



6.6. ábra.
Az ATLAS vevőtípusú utasításainak formátuma

CLOSE	Adó, vevő vagy terhelési funkció aktivizálása vagy kapuzása a mérés tárgyán
CONNECT	A tárgy csatlakozási pontjainak összekötése a tárgycsatlakozó (adó, vevő vagy terhelési) pontjaival
DISCONNECT	CONNECT ellentettje
OPEN	CLOSE ellentettje (lekapcsolás vagy tiltás)
READ	Valamely vevőponton a pillanatnyi érték felvétele és 'MEASUREMENT' címke alatti megőrzése mindaddig, míg a soron következő felvétel új 'MEASUREMENT' értéket nem ad
SETUP	Adó, vevő vagy terhelési funkció előkészítése a tárgy számára
ADJUST TO MINIMIZE TO MAXIMIZE TO REACH	Az ADJUST utasításban megadott adómennyiség változtatása míg a válaszul kapott (vett) mennyiség ki nem elégíti a soron következő TO MINIMIZE, TO MAXIMIZE vagy TO REACH utasításban szereplő feltételt
ALTER	Az ALTER-t megelőző (s jelzővel megjelölt) utasítás egy vagy több meghatározó zónáját módosítja, majd az általa megváltoztatott utasítást végrehajtja
APPLY	A SETUP, CONNECT, CLOSE adó, illetve terhelési feladatoknak a megadott sorrendben való végrehajtása
BEGIN TERMINATE	Az ATLAS eljárás kezdő és záró utasítása
CALCULATE	Valamely (az utasításban megadott) egyenlet jobb oldalának végrehajtása és az eredmény hozzárendelése a bal oldalon álló címkehez
COMPARE	Valamely címkézett mennyiség (ideértve 'MEASUREMENT') értékének összehasonlítása az utasításban megadott (határ) értékkel
DEFINE	Az ellenőrzési eljárás egy szakaszának előállítására, rögzítésére és címkézésére szolgál, mely szakasz akkor hajtódik végre, ha valamely későbbi utasítás a címkeére hivatkozik
DELAY	A soron következő utasítás végrehajtását késlelteti a kiegészítő előírásban megadott idővel
DISPLAY	A berendezés kezelője számára valamely időszakos üzenetet vagy adatot jelentet meg, adatközlő készüléken (KS-cső, alfanumerikus indikátor stb.)
END	A DEFINE ... PROCEDURE sorozat záróutasítása
EXECUTE PREPARE	Együtt használt utasítások; ATLAS utasítások egy sorozatát veszik közre (elől PREPARE, hátul EXECUTE), mely sorozatban az adó, vevő és terhelési utasítások végső kritikus műveleteinek (mint READ és LOAD) végrehajtási időtartamára EXECUTE időlimitet határoz meg.

FINISH	A vizsgálat befejezése, és az ellenőrző berendezés visszaállítása nyugalmi állapotba	
GO TO	Feltétel nélküli vagy feltételes elágazás; utóbbi esetben egy megelőző kiértékelő utasításban megállapított feltételnek megfelelően	
INDICATE	Információ megjelenítése a gépkezelő számára, bármely értelmezhető alakban. Szemben DISPLAY és PRINT-el INDICATE kötetlen a megjelenítést véghezvivő szerv tekintetében	
LEAVE ATLAS RESUME ATLAS	A nem ATLAS-ban írt programszakaszok határoló utasításai, melyek közül a LEAVE felfüggeszti, RESUME újraaktivizálja a compilert	
MEASURE	SETUP CONNECT APPLY READ	} Vevőfunkciót teljesítő utasítássorozat végrehajtása
MONITOR	SETUP CONNECT APPLY READ DISPLAY ,MEASUREMENT' WAIT FOR MANUAL INTERVENTION	
		} vevőfunkciójú utasítássorozat végrehajtása. A két utolsó előtti ciklikusan addig ismétlődik, míg MANUAL INTERVENTION hatására a program a soron következő utasítás végrehajtására nem tér rá
PERFORM	Hatására a DEFINE 'processzor' által definiált utasítások hajtódnak végre előírt sorrendben	
PREPARE	Jelentését l. EXECUT-nál	
PRINT	Valamely programváltozó vagy valamely üzenet másolatának elkészítésére szóló utasítás	
RECORD	Programváltozó adat identifikálása és tárolása olyan esetben, amikor a program futási ideje alatt nincs szükség visszakeresésre	
REMOVE	Adó, vevő vagy terhelési feladat kapcsán	} OPEN DISCONNECT SET UP (nyugalomba) utasítássorozat végrehajtása
REPEAT	Egy vagy több előző utasítás újbóli végrehajtása	
RESUME ATLAS	Jelentését l. LEAVE ATLAS-nál	
SAVE	'MEASUREMENT' jelen értékének egy másik címke alá rendelése abból a célból, hogy ez érték fennmaradjon az ellenőrzési eljárás későbbi céljára	
SPECIFY	Valamely ATLAS főnév egy vagy több utasításjellemzőjéhez érték rendelése abból a célból, hogy a főnév későbbi programbeli említése esetén ez értékek automatikusan a főnévhez kapcsolódjanak	

START WHEN	A keresztülviendő vevő feladatok jelzésére, illetve adott intervallum	
STOP WHEN	(INTERVAL) kezdő és záró feltételeinek megadására szolgáló, egymáshoz tartozó utasításpár; minden START WHEN után STOP WHEN utasításnak kell állnia; a kettő között azonban egyéb utasítások is állhatnak. Az utasításpárt csak olyan megelőző vevő utasítással kapcsolatban, ill. utasításra vonatkozóan lehet használni, amelyben mérési jellemzőként TIME OR SLOP áll.	
TERMINATE	1. BEGIN	
TO MAXIMIZE	1. ADJUST	
TO MINIMIZE	1. ADJUST	
TO REACH	1. ADJUST	
VERIFY	SETUP CONNECT CLOSE READ COMPARE INDICATE EVALUATION if no go	} utasítássorozat végrehajtása
WAIT FOR	Az utasításban előírt feltétel beálltáig az ellenőrzési eljárás végrehajtását felfüggeszti	

GO TO utasítás (amennyiben feltételes elágazást jelent) feltétel zónájában az elágazás helyét megadó szám, majd IF és az előírt feltétel követi egymást.

CALCULATE utasítással aritmetikai műveletek redukálása végezhető, miközben öt műveletjel és zárójel, a végrehajtás sorrendjének jelölésére, alkalmazható. A művelet eredménye elkövetkező számítások céljára *value* vagy *parameter* címke alatt őrizhető meg. Kifejezés elemeiként akár számok, akár címkék alkalmazhatók.

A szintaktikai előírásokat (ezek képezik a compiler implementáció alapját) az ATLAS-ban szintaktikai diagrammok rögzítik.

Az ATLAS nyelven írt program két részre oszlik fel. Az első, deklaratív jellegű bevezető rész (preamble section) a program összes SPECIFY és DEFINE utasításait (ha ilyenek egyáltalán vannak) öleli fel. Minden SPECIFY utasítás (ha van) megelőzi a DEFINE-t. A bevezető rész utasításai nem végrehajthatók, de az általuk leírt információ a program második, procedurális részében tetszésszerűen helyen hivatkozható, ill. felhasználható.

SPECIFY, amint az ATLAS ige-táblázatából is kitűnik, utasításjellemzők általános, az egész programra vonatkozó érték megadására szolgál. A DEFINE utasítás teljes utasítást, utasítás egy részét vagy ATLAS utasítások egy sorát rendel címkéhez, miáltal elkerülhető gyakran használt ATLAS szegmensek újrainírása; ahol erre a procedurális programrészben szükség van, az adott szegmens a DEFINE alatti címkével hívható. A DEFINE utasításnak három változata is van, a DEFINE 'function', a DEFINE 'procedure' és a DEFINE 'message'. A DEFINE 'function' adó-, vevő- vagy terheléstípusú utasítás egy vagy több szomszédos zónáját címkézheti. A címke alatti utasítászónákban szereplő számokat () zárójelpár helyettesíti, jelezve, hogy a szám változó. Ha a procedurális részben az adott 'function' címkét citáljuk, helyére a címke alatt a DEFINE utasításban megadott utasításrész pontos másolata

Néhány ATLAS főnév és modifikátorai

Főnév	Modifikátor	Emlékeztető	Mértékegység
AC SIGNAL	Teljesítmény	POWER	W
	Feszültség	VOLTAGE	V
	Áram	CURRENT	A
	Frekvencia	FREQ	Hz
	Sávszélesség	BANDWIDTH	Hz
	Impedancia	IMP	Ω /szögfok (Hz) $\Omega + -j\Omega - \text{Hz}$
	Harmonikus	HARMONICS	Db v. V v. % v. A
	Nemharmonikus	NON— HARMONICS	Db v. V v. % v. A
	Zaj	NOISE	Db v. V v. % v. A
	Fázis remegés	PHASE—JIT	szögfok
	Teljes torzítás	DISTORTION	Db v. V v. % v. A
	DC eltolódás	DC—OFFSET	V vagy A
	Kvadratura	QUAD	V vagy A
	Fázisban	IN—PHASE	V vagy A
Fázisszög	PHASE— ANGLE	szögfok	
Fázis	PHASE	Egy vagy több fázis	
DC SIGNAL	Teljesítmény	POWER	W
	Feszültség	VOLTAGE	V
	Áram	CURRENT	A
	Impedancia	IMP	Ω /szögfok (Hz) $\Omega + -j\Omega - \text{Hz}$
	Váltakozóáramú feszültség/áram- komponens	AC—COMP	V v. A
	Váltakozóáramú frekv. komponens	AC—COMP— FREQ	Frekv.
	Zaj	NOISE	Db v. V v. % v. A
Teljes torzítás	DISTORTION	Db v. V v. % v. A	
IMPEDANCE	Impedancia	IMP	Ω /szögfok (Hz) $\Omega + -j\Omega - \text{Hz}$
	Frekvencia	FREQ	Hz
	Teljesítményhatár	PWR—LMT	W
	Feszültség határ	VOLT—LMT	V
	Referencia feszültség	REF—VOLT	V
	Disszipációs tényező	DISS— FACTOR	Százalék v. hányad
	Q tényező	Q	Viszonyszám
	Áramhatár	CURRENT— LMT	A
	Ellenállás	RES	Ω
	Kapacitás	CAP	F
Induktivitás	IND	H	

1. SPECIFY használata

000022 SPECIFY, SOURCE, DC SIGNAL, VOLTAGE RANGE 0V TO 100V BY .1V,
AC—COMP .001V\$

888844 APPLY, DC SIGNAL, VOLTAGE -1.005V, CNX HI J2—3 LO J2—4\$

A 000022 utasítás a bevezető részben DC SIGNAL feszültségtartományát és váltakozó áramú komponensét definiálja. Mivel a 888844 utasítás AC—COMP tekintetében semmit sem ír elő, automatikusan a SPECIFY AC—COMP érték jön figyelembe. APPLY elhanyagolja viszont SPECIFY VOLTAGE definícióját, melyre saját, a SPECIFY definíciójától eltérő előírása van.

2. DEFINE használata

000011 DEFINE, 'RZ STIM', SOURCE, AC SIGNAL,
VOLTAGE () RANGE 100V TO 200V BY 1V,
FREQ (), CNX HI H2—1 LO J2—2\$

111133 APPLY, 'RZ STIM', 110V, 400 HZ\$

A 111133 utasítás végrehajtásakor, 'RZ STIM' helyére a DEFINE utasításban az 'RZ STIM' által definiált rész kerül, miközben a VOLTAGE 110V és FREQ 400Hz helyettesítődik. Az összeköttetések CNX definíciójában a melegpont a tárgy H2—1 kapcsához, a föld J2—2 kapcsához rendelődik.

000044 DEFINE, 'MULTIMETER', SENSOR, AC SIGNAL,
IMP 40 + J50 ØHM, CNX HI J1—2 LO J1—3\$

000088 MEASURE, (FREQ), 'MULTIMETER' \$

000098 MEASURE, (VOLTAGE), 'MULTIMETER' \$

000099 MEASURE, (CURRENT), 'MULTIMETER' \$

Látható, miként definiálható valamely vevő (SENSOR) funkció, melynek alapján aztán a procedurális részben, különböző mérésjellemzők bevitelével, különböző mérési műveletek írhatók elő.

3. Általános jellegű példa

000200 BEGIN, ATLAS PROGRAM \$

000201 DEFINE, 'DC1', SOURCE, DC SIGNAL VOLTAGE 28 V ERR LMT + - 1PCS

000202 APPLY, 'DC1', CNX HI J—1 LO J—2\$

000203 MEASURE, (VOLTAGE), DC SIGNAL, VOLTAGE MAX 30 V,
CNX HI J—3 LO J—4 \$

000204 INDICATE, 'MEASUREMENT' \$

000205 TERMINATE, ATLAS PROGRAM \$

A program értelmében, a vizsgálat tárgyának J—1 és J—2 kapocsponjtjára $28\text{ V} \pm 1\%$ kerül, és a J—3 és J—4 pontokon megjelenő feszültség mérése történik. Az INDICATE, 'MEASUREMENT' hatására a legutolsó mérési eredmény, előírt alakban, megjelenik.

6.7. ábra.

Példák az ATLAS nyelv használatára

helyettesítődik, egyúttal a változók konkrét értékei is behelyettesítődnek. A DEFINE 'procedure' egy, vagy egymást követő több ATLAS utasítást címkéz. Ha a program folyamán PERFORM utasításban az adott címkét írjuk le, az általa képviselt utasítás, ill. utasítássor hajtódik végre. A DEFINE 'message' utasítással meghatározott karaktersorozat címkézhető, mely azután a DISPLAY, INDICATE, PRINT vagy RECORD utasítások valamelyikével (az utasításban a címkével) hívható. Az 'üzenet'

()-kel jelölt változókat is tartalmazhat, hasonlóképpen, mint a 'function' meghatározásban.

Tekintettel arra, hogy a bevezető rész összes utasítása SPECIFY vagy DEFINE, egy további szóval jelezni kell, hogy a későbbiekben az adott definíció milyen minőségben kerül felhasználásra. A definiált szegmenst SOURCE vagy LOAD jelöli meg, ha a későbbiekben a szegmenst APPLY, CONNECT, CLOSE vagy ADJUST utasítás használja; MEASURE, VERIFY, MONITOR, WAIT FOR, CONNECT, CLOSE, TO REACH, TO MINIMIZE, TO MAXIMIZE, START WHEN és STOP WHEN utasításokra vonatkozóan pedig a definiált szegmenst SENSOR jelöli.

A nyelv használatát a 6.7. ábrán bemutatott néhány példa szemlélteti.

Az ATLAS-t, bár a gépi implementáció lehetőségét figyelembe véve hozták létre, eredeti alakjában ez idő szerint eljárás leíró nyelvként alkalmazzák; teljes implementációja, terjedelmessége miatt — különösen kisgépeken — egyenlőre nehézségekbe ütközik. Rendszerfüggetlensége, mely bizonyos szempontból előnyös tulajdonságának mondható, másrészt nehézségeket okoz, mihelyt a nyelvet konkrét mérőrendszerhez közelítik, mivel a rendszer elemeinek kezeléséhez szükséges eszközökkel nem rendelkezik. Az implementálhatóság érdekében az ATLAS-t az adott feladatnak megfelelően leszűkítik, miközben a leszűkített nyelvet bizonyos (az eredeti ATLAS-on kívül eső) feladatkörök irányában ki is terjeszthetik, továbbá a rendszerflexibilitás követelményeit kielégítendő, általában alacsonyabb szintű rendszervonatkozású utasításokat is hozzáfűznek.

ATS (Automatic Test System) BASIC

Az ATS BASIC az 1965-ben a Dartmouth College (USA) keretében létrehozott (és azóta széles körben alkalmazott) BASIC nyelvnek a méréstechnika igényeinek megfelelően kiterjesztett változata.

A BASIC-ot első ízben (1968) Hewlett—Packard alkalmazta mérőberendezési programozási nyelveként, később méréstechnikai alkalmazása általánosabbá vált (pl. Systron Donner: SD BASIC és MUSD BASIC). Az alábbiakban a HP—ATS BASIC változatot ismertetjük.

A nyelv karakterkészlete az angol abc A...Z nagybetűiből, a 0...9 számjegyből és különböző írásjelekből + - * / =] [) (> < ' , ; \$ † ? áll. A karakterkészlet az ASC II karakterkészlet részhalmaza.

Az utasítások sorszervezésűek; minden utasítás azonosítási számmal kezdődik. A program az azonosítási számok növekvő értékeinek sorrendjében hajtódik végre még akkor is, ha az utasítások bevitele e sorrendtől eltérő. A számokat a programozási praxisban általában öt vagy tíz többszöröseiként választják meg, hogy a későbbiekben lehetséges legyen utasítások pótlólagos beszúrása, a program újraírása nélkül.

Az utasítás következő eleme: a sorszámától térközzel elválasztott műveletleíró (angol szó, ill. szórövidítés), melyhez különböző vonzatok járulhatnak.

A BASIC kétféle, egyszerű és tömb változót különböztet meg. Az egyszerű változókat A...Z betűk, ill. e betűk bármelyike a 0...9 számjegyek valamelyikével kiegészítve jelölik. Ebben az értelemben pl. B és B8 két egymástól teljesen független változót jelent.

A BASIC a számokat decimális számként definiálja (az előjel választás szerinti); a számok írhatók közvetlen alakjukban, vagy megadhatók az E formátummal is (pl. 1,75E9 jelentése: $1,75 \cdot 10^9$; 1,75E-3 jelentése $1,75 \cdot 10^{-3}$).

Változókból és állandókból a BASIC által ismert operátorok (műveletjelek) igénybevételel numerikus értékek kiszámítására szolgáló kifejezések építhetők fel [pl. $(C+2)/52$].

A matematikai operátorok: $+ - * / \uparrow$ (összeadás, kivonás, szorzás, osztás, hatványozás).

A viszonyító operátorok:

egyenlő	=
nem egyenlő	< > vagy \neq
nagyobb, mint	>
kisebb, mint	<
nagyobb vagy egyenlő	> =
kisebb vagy egyenlő	< =

A logikai operátorok: AND, OR és NOT. AND két kifejezés logikai ÉS kapcsolót képezi. Ha mindkét kifejezés logikailag igaz, az AND kapcsolat is igaz, egyébként valótlan. Az OR operátorral összekapcsolt két kifejezés logikailag igaz, ha legalább egyik kifejezés logikailag igaz. A NOT egy állítás logikai ellentettjének igazságtartalmát vizsgálja.

A HP ATS BASIC műveleteit a 6.4. táblázat foglalja össze. Az egyes utasításokat az alábbiakban értelmezzük:

LET értékadó utasítás, mely változóhoz konkrét értéket vagy kiértékelendő kifejezést rendel pl.:

```
100 LET X=10
110 LET Y=20
120 LET K=(X+Y)/2
130 PRINT "A KÖZÉPÉRTÉK", K
```

PRINT: az idézőjelben álló szöveget változatlan alakban reprodukálja, egyúttal K-nak a 120 alatt számított értékét a szöveg mellé nyomtatja. Használatának másik lehetősége:

```
ssz. PRINT "KÖZÉPÉRTÉK", (X+Y)/2
```

GO TO: feltétel nélküli elágazás. Hatására a program az utasításszámmal definiált utasításra tér rá.

IF—THEN: feltételes elágazás. Ha az IF után álló feltétel teljesül, a program THEN-t követő számmal megadott utasításánál folytatódik, egyébként a soron következő utasításra tér rá.

READ és DATA: összetartozó utasításpár; DATA adatokat sorol fel, READ változókat; a program végrehajtása során DATA adatai READ változóikhoz rendelődnek. Pl.:

```
100 READ X, Y
110 PRINT "KÖZÉPÉRTÉK", (X+Y)/2
120 GO TO 100
130 DATA 10, 20, 15, 25, 30, 40
```

program három középerértéket számít és nyomtat ki; az elsőt a (10, 20) a másodikat a (15, 25), a harmadikat a (30, 40) értékpár alapján. Az értékészlet kimerültét a program ERROR 56 IN LINE 100 üzenettel tudatja.

FOR és NEXT: ugyancsak összetartozó utasításpár; FOR adott változóhoz alsó és felső értékhatárt rendel. A program végrehajtása során a változóhoz először

A HP ATS BASIC műveletei

BASIC alap	elemi BASIC	<p>LET <változó> = <kifejezés> PRINT <üzenet v. kifejezés >, <üzenet v. kifejezés > GO TO <utasítás szám> IF <kifejezés > <viszonyító> <kifejezés> THEN <utasítás szám> READ <változó>, <változó>, ..., <változó> DATA <szám>, <szám>, ..., <szám> FOR <változó> = <kifejezés> TO <kifejezés> STEP <kifejezés> NEXT <változó> END</p>
	kiterjesztés	<p>REM * <bármely karaktersorozat > INPUT <változó, változó, ... > DEF FN <betű> (<változó>) = <kifejezés > GOSUB <utasítás szám > RETURN RESTORE DIM <változó> (<egész sz. >, <egész sz. >) STOP</p>
	mátrixműveletek	<p>MAT READ A Adatok soronkénti beolvasása A-ba DATA alól MAT A = ZER A feltöltése nullákkal MAT A = CON A feltöltése egyesekkel MAT A = IDN A feltöltése egységmátrixként MAT PRINT A A soronkinti kinyomtatása MAT B = A B egyenlővé tétele A-val MAT C = A + B A és B mátrix összeadása MAT C = A - B B kivonása A-ból MAT C = A * B B szorzása A-val MAT C = TRN(A) A transzponáltjának képzése MAT C = INV(A) A invertálása MAT C = (K) * A A skaláris szorzása a K kifejezéssel</p>
HP bővítés	<p>CALL (<szubrutin szám>, <paraméter lista >) WAIT (<formula >) COM < vesszőkkel elválasztott indexelt tömbök ></p>	
további HP bővítés	<p>DCV (< a tápforrás száma >, < paraméter lista >) DVM (< digitális vm. száma >, < paraméter lista >) PAUSE</p>	
HP ATS BASIC kiterjesztése digitális vizsgálatokra	<p>DTUDF (M) ✓ BE/KI pontok kijelölése DTUSD (D) Ellenőrzés időkétszertése DTUTP (P, Y, S, E) ✓ Ellenőrző vektortömb definíciója DTUFX (P,F) ✓ Logikai szintek rögzítése DTURC (R,V) Ellenőrző számláló törlése/kiolvasása DTURS (C, T, P [I], N) P [I] tömb képzése DTUCA (M, A [I], N, X) Ellenőrző vektortömb letárolása DTUTA (Y, S, E, A [I], N, X) Ellenőrző vektorok lehívása letárolóból DTUGP (P, C) ✓ Ellenőrző vektortömb képzése jó kártyával DTUSS (S) Belső/külső szinkronjel választása</p> <p>A ✓ jelölt utasításokat mindig egy vagy több REM utasítás kell, hogy kövesse</p>	

az alsó értékhatár rendelődik, a program ezen értékkel hajtódik végre. A NEXT a változó értékét STEP-pel inkrementálja, s az új értékkel a FOR és NEXT-t közötti programszakasz ismét végrehajtódik. A program a hurokból akkor lép ki, ha a változó (inkrementált) értéke túllépi a számára megadott felső határt. STEP elhagyása esetén az inkrementum automatikusan eggyel egyenlő. Pl.:

```
100 FOR P=2 TO 20 STEP 2
110 PRINT P ↑ 2
120 NEXT P
130 ...
```

a 2 és 20 közötti páros számok négyzetét számítja és nyomtatja, majd rátér a 130 számú utasításra.

END: a BASIC programok záró utasítása.

REM: dokumentáció jellegű, ill. magyarázó szövegnek a programba való beültetése céljára; a REM után álló szöveg a program végrehajtása során figyelmen kívül marad.

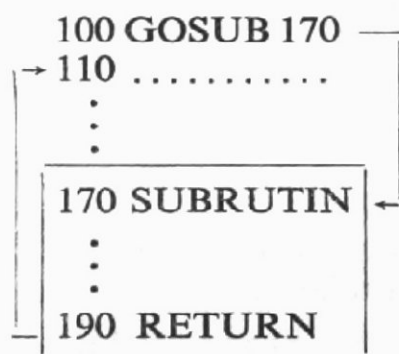
INPUT: módot ad arra, hogy az utasításban felsorolt változók konkrét számértékeit a gépkezelő a program végrehajtása során az írógépről adhassa be. Ahol INPUT utasítás áll, ott a program végrehajtása leáll, és az írógépen ? jelenik meg. A ? megjelenése után a gépkezelő sorra legépel (behelyettesíti) az INPUT alatt álló változók értékeit. Ezt követően a program tovább fut.

DEF FN: függvényt definiáló utasítás abból a célból, hogy az adott függvényt — ismételt használata esetén — ne kelljen újból és újból leírni. A függvény jelölésére az FN után következő egyetlen betű szolgál. Pl.:

```
100 LET X=200
110 DEF FNA (X)=X/20
120 PRINT FNA (X)
130 END
```

A program eredményül 10-et ad.

GOSUB és RETURN: szubrutin-hívó és szubrutin visszatérő utasítás; pl.:



A szubrutin végén RETURN áll, melynek hatására a programvégrehajtás az elágazás helyétől folytatódik.

RESTORE: A READ és DATA utasításokkal kapcsolatban alkalmazzák, amikor egynél több READ utasítás használja a DATA alatt felsorolt adatokat. A READ utasítás az adatmutatót az utolsó tételéig végigfutva elállítja. Mielőtt újabb READ utasítás használhatná az adatokat, RESTORE segítségével vissza kell állítani az adatmutatót. RESTORE tehát mindig két READ utasítás közötti programterületen helyezkedik el.

DIM: Mátrix tárolóterületének kijelölésére szolgál, pl.:

100 DIM A[4,4], B[6,6]

az A mátrix számára 4×4 , a B mátrix számára 6×6 méretű területet jelöl ki. A mátrixműveleteket értelmezésükkel együtt a 6.4. táblázat sorolja fel.

BASIC standard függvényei:

SIN (X)	COS (X)	TAN (X)	ATN (X)
ABS (X)	EXP (X)	LOG (X)	SQR (X)
INT (X)	SGN (X)	RND (X)	TAB (X)

Egyedül az általánostól eltérő jelöléseket magyarázva: ATN (X) és SQR (X) az X arctg, ill. négyzetgyök függvénye; INT (X) az X-hez legközelebb eső egész számot képezi. SGN 1-et generál, ha $X > 0$; 0-t, ha $X = 0$ és -1-et, ha $X < 0$. RND a 0...1 tartományba eső véletlen számokat képez. TAB csupán PRINT utasításban szerepelhet; hatására, mielőtt az írógép kinyomtatáshoz fogna, a kocsit az X által definiált nyomtatási helyzetbe mozdul. A függvények az utasításokban kifejezésként szerepelnek. Pl. 50 LET A=SQR (X ↑ 2+Y ↑ 2).

CALL: A HP BASIC programrendszer alapján interpreterből és előkészítő programból (Prepare BASIC System, PBS) áll. Az előkészítő program tartalmazza a periférikus készülékek kezelő rutinjait, továbbá azokat a rutinokat, melyek a kezelő rutinokat az interpreterrel kapcsolatba hozva, HP BASIC-be szervezik. Mód van arra, hogy a programrendszer alapelemeihez assembly nyelven írott szubrutinokat is lehessen csatolni. E szubrutinokat az előkészítő program ugyanugy a HP BASIC rendszerbe szervezi, mint a rendszer szubrutinokat. A későbbiekben a BASIC programokban ezek az alkalmazási szubrutinok CALL utasítással hívhatók. A CALL utasítás vonatában a hívott szubrutin számát és a szubrutin által igényelt paraméterértékeket kell megadni. CALL alkalmazása lehetővé teszi, hogy a mérőhálózat elemeinek kezelésére, bizonyos mérési eljárások lefolytatására speciális rutinokat alkalmazzunk, és e rutinokat a mérőrendszer mindenkori adottságainak és funkcióinak megfelelően állítsuk össze. Assembly rutinok természetesen nemcsak vezérlési, hanem egyéb célokra is létrehozhatók, miáltal az alap BASIC a legkülönbözőbb irányokban terjeszthető ki.

Assembly szintű rutinok alkalmazási lehetősége a rendszertervező számára maximális flexibilitást, a rendszer alkalmazója részére pedig a CALL utasítás révén, mégis maximális egyszerűséget nyújt. Természetesen a CALL (eltérően a BASIC alaputasításoktól) nem rendszerfüggetlen. Minden egyes, hívószámával azonosított CALL utasításnak csak annak a rendszernek keretében van értelme, melyen belül meghatározták.

WAIT: A mérés két egymást követő lépése közé időeltolást iktat abból a célból, hogy a második lépés a mérés tárgyán állandósult állapotot találjon.

COM: Módot ad az alkalmazási programok szegmentálására. Szegmentálás akkor végezhető, ha a program terjedelme a rendelkezésre álló memóriaterületet felülmúlja, ugyanakkor olyan egységekre bontható szét, melyek mindegyike adatként szolgáltathatja saját eredményeit a soron következő szegmensnek. A méréstechnikában pl. külön szegmensként kezelhető az adatgyűjtést és processzálást végző programszakasz, ill. az adatok analízisét végző programrész. COM, mely mindig az egymással kapcsolatban levő szegmensek legelején áll, a szegmensek közötti adatátadás feladatát látja el.

A fejlődés során a HP BASIC nyelv további méréstechnikai orientációjú utasításokkal egészült ki. Mindenekelőtt célszerűnek látszott bizonyos funkciókat az ál-

talános CALL utasítás körén kívül, közvetlenül a feladatra utaló emlékeztetővel (mnemonic) definiálni. Ilyen utasítások pl. (l. a 6.4. táblázatot) a tápforrások, ill. digitális voltmérők kezelésére szolgáló DCV és DVM, vagy az automatikus hálózat-analizátoroknál a kalibrációs adatok kiolvasását végeztető RCAL és a mérésre és adatkorrekcióra utasító MEAS. A mérés lebonyolítását megkönnyítő lehetőségek is bővültek, amire PAUSE a példa. A program végrehajtása PAUSE utasításnál felfüggesztődik annak érdekében, hogy a gépkezelő valamely rutin feladatot (pl. áramköri kártya behelyezése a mérőfoglatba) elláthasson. A program továbbfolytatására — feladata elvégzése után — a gépkezelő ad parancsot.

Az ATS BASIC (éppúgy, mint maga a BASIC is) konverzációs nyelv, melyet a Hewlett—Packard interpreterként implementál. A számítógép a program összeállításának és végrehajtásának menetében a gép kezelőjével (egymásnak küldött üzenetek révén) konverzációs kapcsolatban áll. A program legépelése és egyidejű bevitele az írógépről, a gép által küldött READY üzenetet követően kezdődhet. A programot az interpreter forrás alakban tárolja. A végrehajtás írógépről adott RUN paranccsal indítható és bárhol, STOP paranccsal állítható le. Az interpreter az utasításokat egyenként értelmezi és hajtja végre. A gépkezelő SCRATCH paranccsal a teljes, folyamatban levő programot törölheti, aminek véghezvitelét a gép READY-vel jelzi. LIST paranccsal a teljes, LIST XXXX-szel az XXXX sorszámú utasítástól az END-ig terjedő lista íratható ki. További három parancs — TAPE, PTAPE, PLIST — a készülékekre vonatkozik. Hibaállapotot, a hiba típusának és helyének feltüntetésével, a gép ERROR üzenet formájában jelez.

A HP ATS BASIC méréstechnikai alkalmazását a 6.8. ábrán bemutatott példa szemlélteti.

```

90 REM "ERŐSÍTÉS MÉRÉSE"
100 INPUT U, I, M, F, G, T
110 CALL (1, U, I)
120 WAIT (T)
130 CALL (2, M, F, I)
140 IF (Y < (1.05 * U * G)) AND (Y > (.95 * U * G)) THEN 180
150 PRINT "AZ ERŐSÍTÉS NEM MEGFELELŐ"
160 PRINT "BEMENET="U; KIMENET="Y
170 GO TO 100
180 PRINT "ERŐSÍTÉS MEGFELELŐ"
190 GO TO 100
200 END

```

A 100 INPUT bekéri az erősítésméréshez szükséges adatokat. Ezek: U — a bemenő feszültség értéke; I — a feszültségforrás áramkorlátja, M — DVM méréstartománya; F — DVM mérés-neme; G — az erősítés értéke; T — a bemenőjel ráhelyezése és az erősítés mérése közé iktatott időkésleltetés. A 110 CALL rutin ráhelyezi az erősítőre a bemenő jelet. 130 CALL a DVM-t aktivizálja. 140 IF THEN megvizsgálja, hogy Y értéke megfelel-e a $G \pm 5\%$ feltételnek. A program további része magától értetődő.

6.8. ábra.

Példa a HP ATS BASIC alkalmazására

A HP ATS BASIC kiterjesztése digitális vizsgálatokra

A digitális technikában — elsősorban a technológiai háttér jelenlegi gyors és folyamatos változása miatt — a vizsgálati módszerek nem konszolidálódtak. A létrehozott mérőrendszereknek csaknem mindegyike más-más ellenőrzési—diagnosztikai meto-

dikát követ, az alkalmazott nyelvek pedig szigorúan kötődnek a metodika és a mérőrendszer realitásaihoz. A HP ATS BASIC digitális technikai kiterjesztése is konkrét rendszer, az 1.32. ábrán bemutatott HP 28035 A típusjelű digitális ellenőrző egység (Digital Test Unit, DTU) és az általa megtestesített vizsgálati metodika alapján történt (1971). A későbbiekben (1973) az azonos elvek szerint felépített HP 9560 digitális áramkörellenőrző rendszerben, ugyanezen nyelvnek csekély mértékben továbbfejlesztett változata nyert alkalmazást.

Adott esetben a vizsgálati eljárás a következő főbb lépésekből áll:

1. Programozási úton be kell állítani a vizsgálat tárgyát képező digitális kártya tápfeszültségét, a vizsgáló jelek logikai szintjeit, valamint a logikai szintek ellenőrzési (egyenáramú komparálási) szintjeit.

2. A tárgycsatlakozó kapcsait a tárgy bemenő, illetve kimenő pontjaiként programozási úton ki kell jelölni: egyúttal meg kell adni azt az időt is, melyet a mérőrendszer a bemenő vektor kibocsátása és a válaszvektor észlelése közé kell, hogy iktasson.

3. A rendszer a programnak megfelelő logikai vektorsorozatot kell, hogy képezzen, minden egyes választ összehasonlítva referenciavektorával. Hiba észlelése esetén a program bizonyos nyomozó rutinra tér rá.

A HP BASIC digitális ellenőrzésre szolgáló utasításait ugyancsak a 6.4. táblázat tartalmazza. Az utasítások értelmezését a 6.9. ábra programjának nyomon követésével végezzük el. Bizonyos DTU utasításokat követően a REM utasításnak a korábbtól eltérő különleges jelentése van.

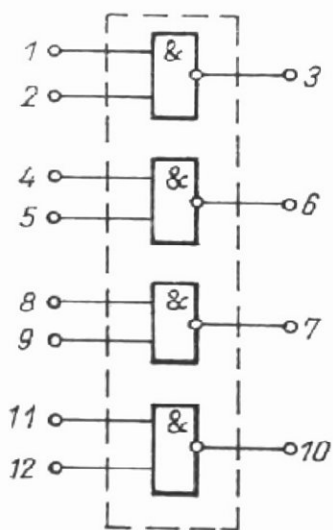
A program REM megjegyzéssel, a program nevével kezdődik. A DCV utasításokban az első szám a rendszeren belüli feszültség (forrás) azonosítója, a második a feszültség szint értéke, a harmadik az áramhatáré. A 110 utasítás a tápfeszültséget, 120 és 130 a logikai szinteket, 140 és 150 a komparálási szinteket adja meg. 160 DTUDF(1) a tárgycsatlakozó kapcsainak a tárgy be-, ill. kimenetként való kijelölését, a közvetlenül utána álló REM IN (tárgybemenet) és REM OUT (tárgykimenet) utasításokkal együtt végzi.

190 DTUSD(5) az ellenőrző vektor kibocsátása és az ellenőrzés megejtése közötti időt 5 μ s-re állítja be.

DTUTP (P, Y, S, E) utasításban P-vel a logikai értékhez alsó/felső szint (polaritás) rendelhető; Y és S a vizsgálat belépő, kilépő helyét határozza meg; E a hibanyomozó rutin elágazási címe. DTUTP mindig a közvetlenül utána következő REM utasítás(ok)al együtt értelmezhető. 230 DTUTP (2, Y, 1, 500) a logikai 1 értékhez a felső feszültség szintet rendeli. A 250...280. sorszámú sorban a kérdő és referencia vektorokból összetett ellenőrző vektorokat találjuk. Az Y=1 érték azt jelzi, hogy a vizsgálatot az első, a 250. ssz. sorban levő vektorral kell kezdeni. Az S=1 arról intézkedik, hogy a 280. ssz. ellenőrző vizsgálat lefolytatása után a 290. ssz. utasítás végrehajtására kell rátérni. Az E=500 a hibanyomozó rutin címét adja meg. A 240 REM PINS a tárgycsatlakozó kapocspontjait és a 250...280 REM utasításokban álló logikai vektorok komponenseit egymáshoz rendeli. Esetünkben pl. a 280. ssz. utasítás végrehajtása során a kapocspontokhoz rendelt logikai értékek:

Kapocspontok:	1	2	3	4	5	6	7	8	9	10	11	12
Logikai értékek:	1	1	0	1	1	0	0	1	1	0	1	1

A jó kártya vizsgálata a 220...310 programszakasz alapján oly módon zajlik le, hogy a számítógép egymás után négyszer (l. 240...280) egy-egy ellenőrző vektort bocsát a mérőhálózatra. Miután az egyes vektorok alapján elvégzett vizsgálatok mindegyike pozitív eredménnyel zárul, az írógépen a KÁRTYA JÓ üzenet jelenik meg. 300 DTURC az ellenőrző lépések sorszámát regisztráló 12-bites számláló tartalmát törli (az utasításban R=1 törlést jelez), majd a program újabb kártyát kér.



SSZ	BE								KI			
	1	2	4	5	8	9	11	12	3	6	7	10
1	0	0	0	0	0	0	0	0	1	1	1	1
2	0	1	0	1	0	1	0	1	1	1	1	1
3	1	0	1	0	1	0	1	0	1	1	1	1
4	1	1	1	1	1	1	1	1	0	0	0	0

100 REM * ÉS NEM KVAD ELLENŐRZÉSE 110 DCV (1, 5, 100) 120 DCV (2, 4, 20) 130 DCV (3, .2, 20) 140 DCV (4, 2, 20) 150 DCV (5, .8, 20)	feszültségszintek beállítása
160 DTUDF (1) 170 REM IN 1, 2, 4, 5, 8, 9, 11, 12 180 REM OUT 3, 6, 7, 10 190 DTUSD (5)	bemenő és kimenő kapcsok kijelölése; az inger ráadása és a válasz mérése közötti idő megadása
200 PRINT "KÉREM A KÁRTYÁT" 210 PAUSE	kártya behelyezése
220 LET Y=1 230 DTUTP (2, Y, 1, 500) 240 REM PINS 1—6, 8, 9, 7, 11, 12, 10 250 REM 001 001 001 001 TEST#1 260 REM 011 011 011 011 TEST#2 270 REM 101 101 101 101 TEST#3 280 REM 110 110 110 110 TEST#4 290 PRINT "KÁRTYA JÓ" 300 DTURC (1, V) 310 GO TO 200	ellenőrző vizsgálat
500 DTURC (2, V) 510 DTURS (3, 2, P[I], 4) 520 PRINT "TEST#"; V; "PINS" 530 FOR I = 1 TO 4 540 IF P[I]=0 THEN 560 550 PRINT P[I] 560 NEXT I 570 PRINT "HIBÁS" 580 LET Y=2 590 GO TO 230	hibanyomozó rutin

6.9. ábra.

Példa a digitális vizsgálatokra kiterjesztett HP ATS BASIC használatára

Amennyiben a kártya hibás, és a hibát pl. a 260. sz. utasítás észleli, úgy a program az 500. sz. utasításra, a hibanyomozó rutinhoz ágazik el. A hibanyomozó rutinnak meg kell állapítania, és az ellenőrzést végző személlyel közölnie kell, hogy a hiba hányadik sorszámú vizsgálati lépésnél (esetünkben a második), továbbá, hogy a tárgy melyik kimenő pontján jelentkezett. 500 DTURC (2,V) hatására mindenekelőtt a 12-bites számláló tartalma V-vel jelzett változóként tárolódik. Az 510 DTURS (3,2,P[I],4) utasítás hatására a mérőhálózat eredményregiszterének tartalmát a gép behívja, majd a P[I] tömb képződik, mely a tárgy kimenő pontjainak megfelelő számú helyet (N=4) tartalmaz, s azoknak a kapcsoknak a számával töltődik fel, melyeken a vizsgálat eredménye hibát jelez (ahol nincs hiba, ott a tömbben 0 jelenik meg). Az 520. sz. utasítás kinyomtatja a TEST# mellé a V tartalmát (ez esetben: 2), továbbá a PINS szót. Ezután a tömb négy lépésből (mivel az összes kimenet száma: 4) álló ellenőrzése következik. Ahol az ellenőrzés nullát talál, ott továbblép, ahol kapocsszámot talál, ott azt kinyomtatja. A teljes ciklus lefutása után a gép "HIBÁS" minősítést ad, majd miután Y=2 értéket definiál, visszatér a 230. sz. utasításhoz. Y=2 azt jelzi, hogy az ellenőrző vizsgálatot most már nem a 250., hanem a 270. sz. utasításnál kell folytatni.

Amennyiben az ellenőrző vektorok sorozatában bizonyos szakaszon át meghatározott kapcsokhoz állandó logikai értékek tartoznak, úgy a polaritás rögzítése után ezek a DTUFX utasítással rendelhetők egymáshoz, ami DTUTP alatti megtakarítást tesz lehetővé. A DTUFX definíciója újra definiálásig érvényes marad.

A DTUCA utasítás használatával tárolóterület takarítható meg. Amennyiben a program elejére helyezett COM utasítással tárolóterületet biztosítunk, a 150. sz. és 160 sz. utasítások közé, valamint a 300. sz. helyére DTUCA utasítást helyezünk, úgy az ellenőrző vizsgálat végrehajtása során a kapocsdefiníció és az ellenőrző vektorok tömbje COM területére tárolódik. Ezek után a 6.9. ábra programja a 6.10. ábrán látható programmal helyettesíthető. Az ellenőrző vizsgálat ez esetben

```

100 COM A [100]
110 DCV (1, 5, 100)
120 DCV (2, 4, 20)
130 DCV (3, .2, 20)
140 DCV (4, 2, 20)
150 DCV (5, .8, 20)
160 DTUDF (1)
170 PRINT "KÉREM A KÁRTYÁT"
180 PAUSE
190 LET Y=1
200 DTUTA (Y, 1, 500, A[I], 100, 600)
210 PRINT "KÁRTYA JÓ"
220 GO TO 170

```

6.10. ábra.

A 6.9. ábra programjának a DTUCA utasítás alkalmazásával egyszerűsített változata

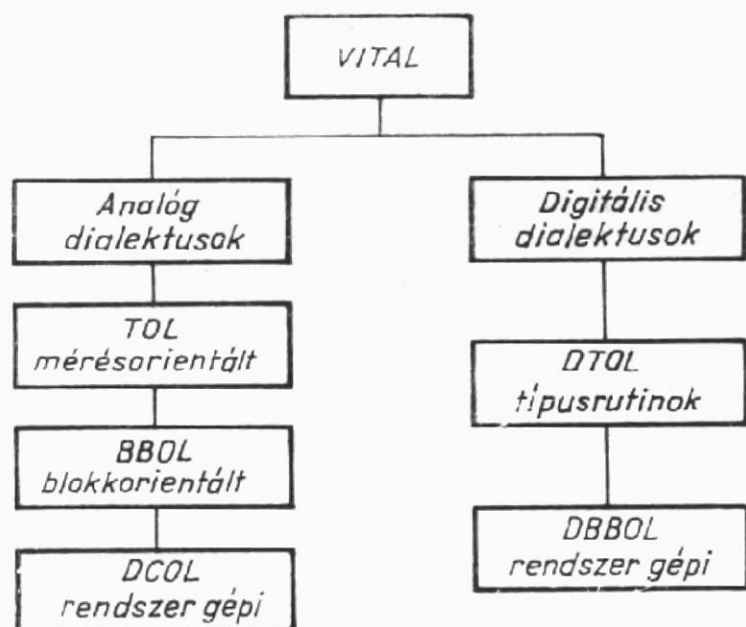
négy utasításra zsugorodik; DTUTP helyett DTUTA áll, melyben A[I], 100 a vektor-tömbre utal, a 600 pedig egy újabb elágazási cím arra az esetre, ha az A tömb valamilyen hibája fordulna elő.

Igen nagyméretű vektortömbök háttértárolón szegmentálva helyezhetők el, és szegmensenként alkalmazhatók.

A DTUGP (P, C) utasítással a mérőrendszer tanulórendszerként működtethető; „jó kártya” segítségével vektortömb generálható.

A direkt alkalmazással kapcsolatos nehézségek ellenére az ATLAS — bázisnyelv-ként — a mérés technikában igen jelentős szerepet tölt be. Egy sor olyan nyelv ismeretes, melyet egy-egy alkalmazási terület igényeire, egy-egy mérőberendezés adottságaira orientálva az ATLAS-ból kiindulva alakítottak ki. Így jöttek létre a különböző ATLAS származéknyelvek, dialektusok, adaptációk és ATLAS-szerű nyelvek.

A származéknyelvek közül a VITAL [27] és a GOAL [34] említésre méltó. A VITAL (VAST Interface Test Application Language) az Egyesült Államok haditengerészeténél alkalmazott VAST (Versatile Avionic Shop Test) rendszer nyelve, melyet ugyancsak repüléstechnikai célokra alkalmaznak. A VITAL kettős — analóg és digitális — feladatkörű nyelv. Analóg része háromszintű, digitális része kétszintű. Az egyes szinteken definiált nyelvi egységeket a VITAL dialektusainak nevezik. Ennek megfelelően három analóg és két digitális, összesen tehát öt VITAL dialektus különböztethető meg (6.11. ábra).



6.11. ábra.
A VITAL nyelv dialektusai

A mérésorientált, rendszerfüggetlen TOL dialektus az ATLAS-hoz igen közel esik; a két nyelv gramatikája csaknem azonos.

A középszintű analóg dialektus, BBOL, blokkorientált rendszernyelv, mely lehetővé teszi a mérőhálózat nagyobb blokkjainak programozási úton való elérését. Emellett BBOL sok TOL vonást megőriz. A jelútvonalak automatikus kiválasztása itt is az összeköttetések definíciója (l. ATLAS) alapján történik, és az igék szótára és meghatározása is lényegében azonos, mint TOL-ban. A fő különbség, hogy míg TOL-ban a főnév jelorientált, BBOL-ban a mérőhálózat blokkjait címezi meg.

A legalacsonyabb szintű VITAL dialektus a DCOL, a mérőrendszer gépi nyelve. Ezen a nyelvi szinten a mérőrendszer minden programirányítású eleme külön-külön elérhető. A DCOL szintű nyelv lehetőséget ad a mérőrendszer belső, funkcionális ellenőrző vizsgálatára. Általa lehetséges továbbá, „forró” állapotban, azaz a tárgycsatlakozó kapcsolórendszerének zárt (aktív) állapotában változtatni a rendszerblokkok jeleit.

A 6.1. táblázat három különböző szintű nyelve a VITAL három analóg dialektusának használatát szemlélteti.

A digitális dialektusok közül DBBOL az alacsonyabb szintű. DBBOL a VAST rendszer digitális alegységének időreális szimbolikus gépi nyelve. Nyolc ige (START, STOP, CLEAR, LOAD, READ, CONFIGURE, SET UP és NO—OP) köré épül fel. E dialektus flexibilis, de igen komplex, nehezen elsajátítható. A digitális vizsgálatokat egyszerűbbé teszi a magasabb szintű DTOL, mely lényegében bizonyos típusvizsgálatok rutinjainak könyvtára. A rutinok CALL utasítással hívhatók, a konkrét feladatnak megfelelő paraméterek mellékelésével. A rutinok között a sorrendi hálózatok ellenőrző eljárásai is megtalálhatók, de DTOL nem rendelkezik időreális vizsgálati lehetőséggel, mivel az időzítések a rutinokon belül rögzítettek.

A VITAL jellegzetessége, hogy dialektusai a programban a programozó igényeinek megfelelően keverhetők. A VITAL terjedelme nem teszi lehetővé a VAST rendszer VARIAN 622/i—24K gépén való közvetlen használatát. Nagygépre implementált kereszt-compiler az öt különböző dialektusban írott VITAL programokat a mérőrendszer VOSC (VAST Operating System Code) nyelvére fordítja le (6.12. ábra). A VOSC nyelvű programok a mérőrendszeren on-line futtathatók.

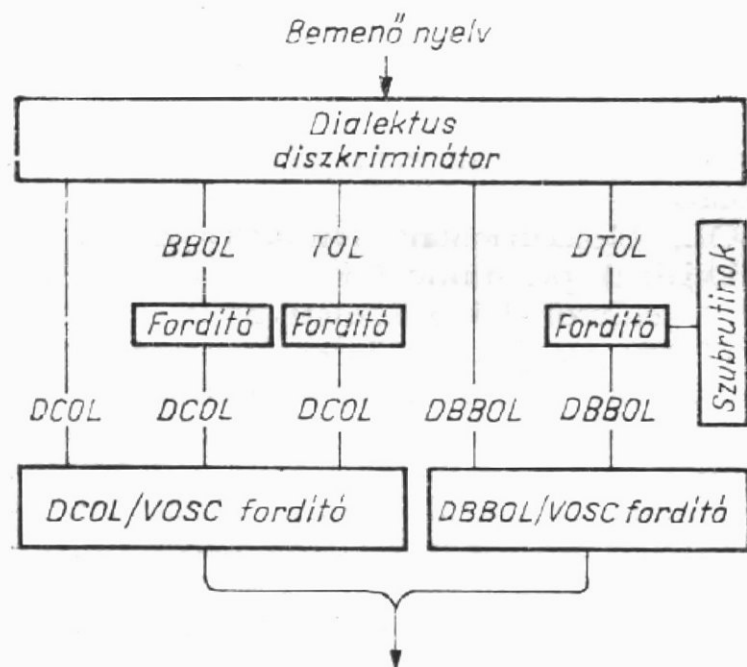
Az ellenőrzéstechnikai orientációjú GOAL (Ground Operations Aerospace Language) [34], melyet a Kennedy űrkutató központban alkalmaznak, a sok tekintetben meghaladott, korábbi ATOLL-t [35] váltja fel. A GOAL az ATLAS-hoz lazán kötődő, más magasabb szintű nyelvek tulajdonságait is implikáló, analóg és digitális feladatkörre kiterjedő nyelv. Űrszerkezetek és elemeik ellenőrzése céljára alkalmazzák.

A brit honvédelem IDA (Intermediate Dialect of ATLAS) elnevezéssel az ATLAS-nak egy általánosított mérőrendszerre orientált dialektusát használja.

A műszaki ellenőrzés köznapiabb területeiről a Texas Instrumentst „ATS—960” [36], az Instrumentation Engineering „System 390” [37], valamint az RCA „EQUATE” [33] rendszerének ATLAS adaptációit kell megemlíteni.

ATLAS alapokra támaszkodik a Rhode—Schwartz számítógép-irányítású mérőrendszerének SESAM II (Symbolische Eingabe-Sprache für Automatische Meßsysteme) nyelve [39].

A Messerschmitt—Bölkow—Blohm elektronikus elemek vizsgálatára szolgáló MMB-mérőrendszerének COUNT nyelve [38] az ATLAS és BASIC együttes származéka.



6.12. ábra.
A VITAL nyelv fordítóprogramjának szerkezete

A Marconi Instruments, AUTOTEST és OE 1759 jelű számítógépes ellenőrző rendszeréhez MI—BASIC nyelvet alkalmaz. A Systron Donner "CATSystem 360" rendszerében az SD BASIC és a MULTI—USER SD BASIC változatokat alkalmazza.

Munkaleíró nyelvek (job languages)

Míg a technika egyes területeit a mérési eljárások szempontjából nagyfokú mozgás és gyors változás, más területeket viszonylagos állandóság jellemez. A relégyártásban pl. évtizedek óta nagyjából ugyanazokat a paramétereket kell vizsgálni, a relék bizonyos csoportokba sorolhatók, az egy csoporton belüli változatokat azonos jellemzők írják le, csupán a paraméterek értéke változik típusonként. Jó ideje hasonló a helyzet a diszkrét félvezető elemek, a ferritmagok, de az elektronikán kívül eső számos egyéb területen is. E méréstechnikai területeket a feladatstruktúra nagyfokú állandósága és a szűk feladatapertúra jellemzi, melyen belül valamennyi mérési eljárás bizonyos elemi rutinokra vezethető vissza. A rutinok összessége alkalmazási programcsomagként a mérőrendszer programrendszerének részét képezi. A programcsomag rutinjai a rendszer munkaleíró nyelvének utasításaival aktivizálhatók. A programnyelv egyaránt emlékeztet az interpretív nyelvekre és az adagolt üzemmódú (batch processing) rendszerek munkaleíró (job description) nyelvére. Az elektronikus és elektromos alkatrész- és szerelvényellenőrző berendezések gyártására specializálódott Teradyne a fent vázolt utat követi, s az utóbbi analógia alapján a munkaleíró nyelven írott programjait "munkaterv"-nek (job plan) nevezi. A 6. 13. ábra példaképpen egyszerű relé munkatervének felépítésmódját (a 2. ellenőrző lépést munkaleíró utasításokkal részletezve) mutatja be [40]. Látható, hogy az ellenőrző lépés szubrutinjának megnevezése mellett meg kell adni az adott relétípus vizsgálati paramétereinek értékét is. A munkatervben a részfeladatok végrehajtására megadott sorrend nem kötelező; tetszés szerint választható meg.

Egyéb nyelvek

Nem térhetünk ki a számos különböző számítógép-irányítású mérő- és ellenőrző rendszer egyedi nyelvére, viszont az őket alkalmazó vállalatok jelentőségére való tekintettel, ha csak röviden is, utalnunk kell a General Electric GETEL (General Test Engineer Language) [41] és az International Telephone and Telegraph üzemeiben alkalmazott PROTEST [42] nyelvekre.

Az eddigi felsorolásban ún. ellenőrző nyelvek (test languages) szerepeltek. Meg kell azonban említeni a méréstechnikai orientációjú nyelveknek egy másik csoportját, az időreális nyelveket is, melyek primer alkalmazási köre ugyan a folyamatszabályozás és a laboratórium, de melyekkel, esetenként, a műszaki méréstechnikában is találkozhatunk. Az időreális nyelvekkel szemben az alábbi általános követelményeket hangsúlyozzák:

- bit és karakterlánc kezelésének, adatstruktúra definiálásának és kezelésének lehetősége;
- paralel folyamatok kezelésére és időzítésére szolgáló nyelvi elemek;
- e folyamatok szinkronizálására szolgáló elemek;
- a szerkezeti kapcsolatok leírásának lehetősége nyelvi szinten;
- lehetőség a nem standard periferikus eszközök kezelésére.

Az időreális nyelvek közül a Purdue Workshop on Standardization of Industrial Computer Languages által, az ANSI X3.9—1966 Standard FORTRAN alapján

1 CÍM R25 TSZ RELÉ VIZSGÁLATÁNAK MUNKATERVE
2
3 ;MEGJ: EGYSÉGEK: VOLT, AMPER, OHM ÉS SEC
4
5 ;CSATLAKOZÓPONTOK HOZZÁRENDELÉSE
6 HASZNÁLD 1 NO 2 NC 3 NO 4 NC
7

8 1 MÉRÉS; TEKERCSELLENÁLLÁS VIZSGÁLATA

9

10

11

12 2 MÉRÉS; KONTAKTUS ELLENÁLLÁS VIZSGÁLATA

13 1 TEKERCSTRE 28 VOLTOT RÁ 10 MS VÁRAKOZÁS

14 HA R (1 KONTAKTUS) $>0,05$ VAGY R (3 KONTAKTUS) $>0,05$ ÚGY ELVETNI

15 1 TEKERCSTRE 28 VOLTOT RÁ 10 MS VÁRAKOZÁS

16 Ha R (2 KONTAKTUS) $>0,05$ VAGY R (4 KONTAKTUS) $>0,05$ Ft ÚGY ELVETNI

17

18 3 MÉRÉS

19

20

21

.

45 N MÉRÉS

46

47

48

49 ELFOGADVA

50 VÉGE

6.13. ábra.

Relé vizsgálatának munkaleíró nyelven összeállított munkaterve

kidolgozott, és ISA standardként publikált ipari FORTRAN [43] említendő meg, melyet a Los Alamos Sci. Lab. CAMAC szubrutinokkal is kiegészített. A CORAL 66 [44, 45] az Angliában rendszeresített időreális nyelv az ALGOL 60-nal rokon, de elsősorban kisgép-implementációra orientált. Végül a folyamatszabályozás és a kísérleti fizika együttes igényeinek szem előtt tartásával, a Német Szövetségi Köztársaságban kidolgozott PEARL (Process and Experiment Automation Realtime Language) [46] érdemel figyelmet.

6.2.3. A mérőrendszer működtetése

Nyelvi szintek

A mérési feladat megfogalmazása, a mérés algoritmusának leírása — eredendően — mindig rendszerfüggetlen módon, a mérést végrehajtó berendezés közelebbi meghatározása nélkül történik. A rendszerfüggetlenség azonban relatív, hiszen

bármely méréstechnikai feladat csupán olyan eljárásként képzelhető el, mely hallgatólagosan feltételezi a végrehajtáshoz szükséges konkrét eszközöket, valamint ezen eszközök egymáshoz és a mérés tárgyához fűződő relációit is. A mérési algoritmus abban az értelemben lehet rendszerfüggetlen, hogy az eszközök megnevezése nélkül írja elő mindazt, amit ez eszközöknek egyenként és együttesen a mérési eljárás lefolytatása során teljesíteniük kell, beleértve meghatározott jeleknek a tárgyra (ill. környezetére) való rávitelét, jeleknek a tárgyon (és környezetében) való érzékelését, mérését stb.

A másik oldalon a feltételezett eszközöket felölelő mérőrendszer áll, mely feladatát a számítógép, az egységes csatlakozórendszer, ill. a mérőhálózat gépi szintjén hajtja végre.

A mérőrendszer működtetési filozófiájához hozzátartozik az az eljárás mód, melynek alapján a rendszerfüggetlen nyelven leírt algoritmus gépi nyelvre tevődik át, s melynek alapján a program végrehajtható.

A mérőrendszer két nyelvi szinttel jellemezhető. *Bemenő szinten* azt a nyelvi szintet értjük, melyen a feladat algoritmusát a rendszerre vihetők; *on-line szinten* pedig azt, melyen az algoritmus közvetlenül végrehajtható.

A bemenő szint a rendszerfüggetlen nyelv és a rendszer gépi nyelve között — e határokat is beleértve — bárhol elhelyezkedhet. Az on-line szint összeeshet a rendszer gépi nyelvével, vagy az interpretív nyelvek valamely alacsonyabb-magasabb szintű változatának felelhet meg.

A rendszerfüggetlen nyelvi szintről a bemenő szintre (ha e kettő nem esik össze) a mérési algoritmust a rendszeren kívül kell áttenni, a bemenő szintű programot az on-line szintre a mérőrendszer operációs rendszere fordítja le. A bemenő szint és on-line szint esetenként összeeshet, továbbá, adott rendszernek egynél több bemenő szintje is lehet (l. 6.12. ábra).

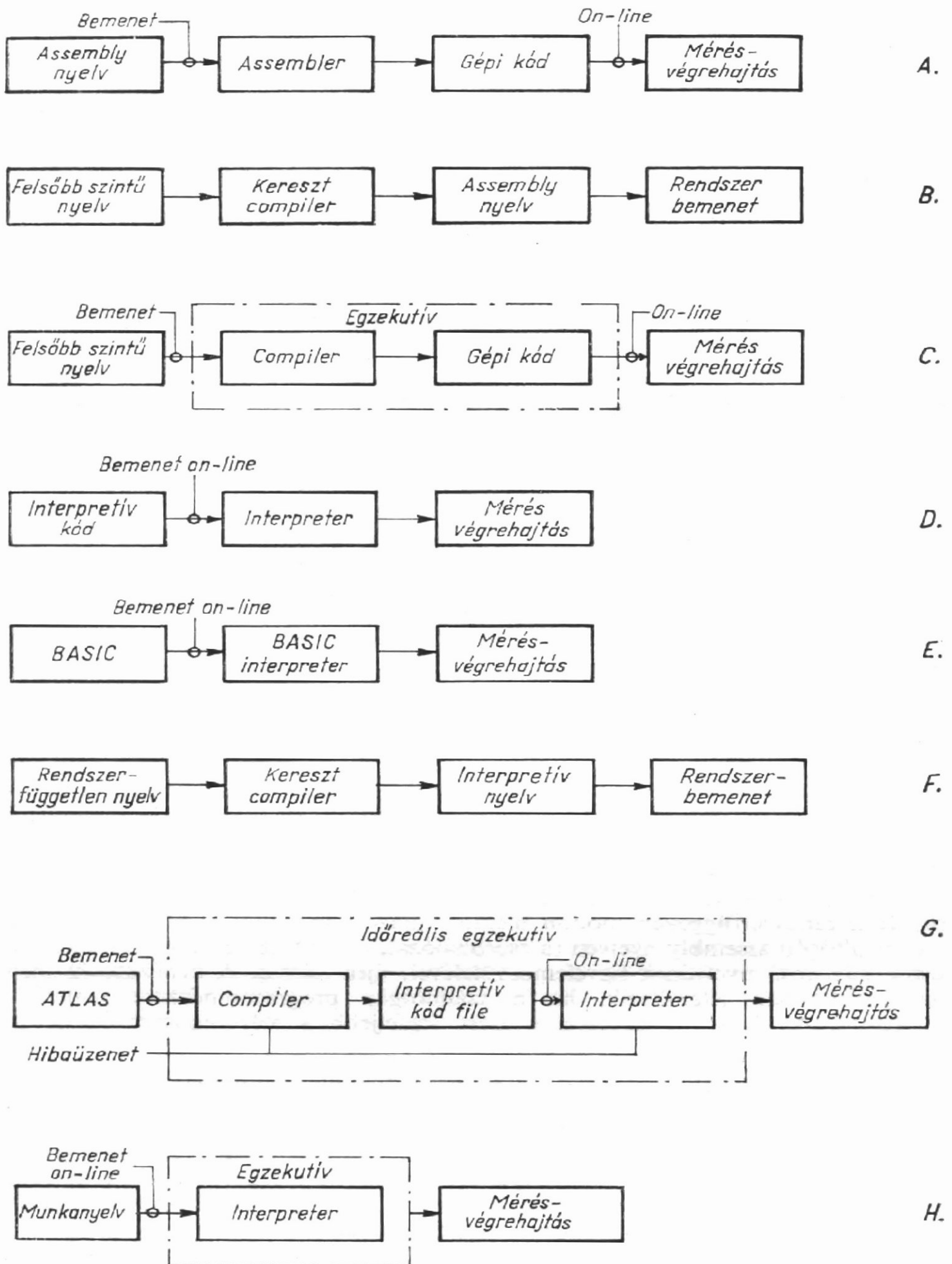
Programozási sémák

Attól függően, hogy a rendszerfüggetlen és a gépi-nyelvi szint által behatárolt tartományban hol helyezkedik el a mérőrendszer bemenő, ill. on-line szintje, egymástól eltérő programozási sémák különböztethetők meg. A legjellegzetesebb sémákat a 6.14. ábra mutatja be.

Az összes közül legkezdetelegesebb az A séma. A programrendszer itt ugyan egyszerű, de a rendszerfüggetlen módon megfogalmazott feladatot a programozónak kell a számítógép assembly nyelvén (a mérőhálózat, esetleg az egységes csatlakozási rendszer közbenső nyelvének figyelembevételével) újra leírnia. A nehézkesség még hangsúlyozottabban jelentkezik, ha a számítógép programrendszere manuális működtetésű. Az A séma használata ezért legfeljebb a mérőrendszer fejlesztési stádiumában jöhet figyelembe.

A programozási séma lényegi megváltoztatása nélkül, különböző programozástechnikai lehetőségek kihasználásával, az assembly módszer lényegesen javítható. Mindenekelőtt kidolgozhatók a mérőhálózat elemeinek és a méréstechnikai feladatstruktúra által meghatározott típuseljárásoknak a makro vagy assembly-interpretív rutinjai. Amennyiben a számítógépnek moduláris diszk operációs rendszere van, úgy e rutinok beépíthetők az operációs rendszerbe. Ily módon a bemenő program jórészt szubrutin-hívásokból és a rutinok bemenő adataiból áll.

Az A séma célszerű alkalmazásának egy másik lehetősége: az adott mérőrendszerhez tartozó, de más (nagyobb) gépen futtatható kereszt-compiler munkába állítása; a B sémának az A séma elé iktatása.



6.14. ábra. Mérőrendszer jellegzetes programozási sémái

Amennyiben a választott felsőbb szintű nyelv terjedelme és strukturális bonyolultsága összhangba hozható a mérőrendszer adottságaival, alkalmazható a C séma, melynél a compiler felsőbb szintű nyelvről közvetlenül gépi kódra fordít. Ez a megoldás látszólag egyszerű, azonban az ilyen közvetlenül fordító compiler meglehetősen bonyolult (esetleg többszörös átfutású), a felsőbb szintű nyelvnek pedig eleve kiszámítógép-orientáltak vagy kiscségekre leszűkítettnek kell lennie.

Az A...C sémák fordítóprogramokat alkalmaznak; on-line szintjük minden esetben gépi kód. A 6.14. ábra a lényegre mutatóan egyszerűsít. A valóságban a sémák különböző szerkesztési, hibátlanítási és programérvényesítési eljárásokkal egészülnek ki. On-line nyelvként gépi kód alkalmazása — időrealitási szempontok alapján — bizonyos mérések kapcsán szükségzerű lehet, mivel az interpretív programvégrehajtás a gépi kóddal szemben mindenkor lassúbb. Ugyancsak időrealitási megfontolások alapján esetenként a legalacsonyabb szintű assembly nyelvet azért kell alkalmazni, mivel mind terjedelemben, mind futási idő szempontjából ez tesz leggazdaságosabb programfelépítést lehetővé.

A fordítóprogram helyett interpretert alkalmazva, a D sémához jutunk, melyet a bemenő és az on-line nyelvi szint egybeesése jellemez. Az interpretív nyelv: numerikus vagy mnemonikus kriptokód. A numerikus kriptokód pl. négyjegyű hexadecimális számok sorozataként adható meg, ahol minden egyes számnak önnönmaga és a környezet által meghatározott jelentése van. E kód a lyukszalag-nyelvekre emlékeztet. A mnemonikus kriptokód számozósítók helyett az élő nyelvek valamelyikéből (pl. angolból) extrahált rövidítésekkel nevezi meg a mérőhálózat emeleit. „A kiscségezésű jelgenerátort 0.1V;400Hz értékre állítani” parancs kétféle kriptokódja pl.:

numerikus:	mnemonikus:
B021	OS
1040	1040

A kiscségezésű jelgenerátor beállítására a numerikus kódban a B021 hexadecimális szám (21 a generátor rendszeren belüli címe), a mnemonikus kódban az OS (oszillátor) emlékeztető utal. A második sorban álló szám mindkét esetben a kimenő feszültséget (0,1V) és a frekvenciát (400 Hz) adja meg. Mindkét kódforma, lényegében a mérőhálózat numerikus, ill. szimbolikus nyelvének felel meg.

A D séma primitív nyelvű interpretív eljárásával szemben, programozástechnikai szempontból lényeges előrelépést jelent az E séma, melynek bemenő on-line nyelve a felsőbb szintű BASIC. Az interpreter mint végrehajtó rendszer előnyeiről a korábbiakban, a 6.1.2. szakaszban már szó volt. Figyelembe kell venni azonban, hogy bár az interpretív nyelv szintjének emelésével a programozással járó nehézségek csökkennek, ugyanakkor a programvégrehajtás ideje növekszik. Egyes esetekben, ez az időrealitás követelményével ellentétben áll.

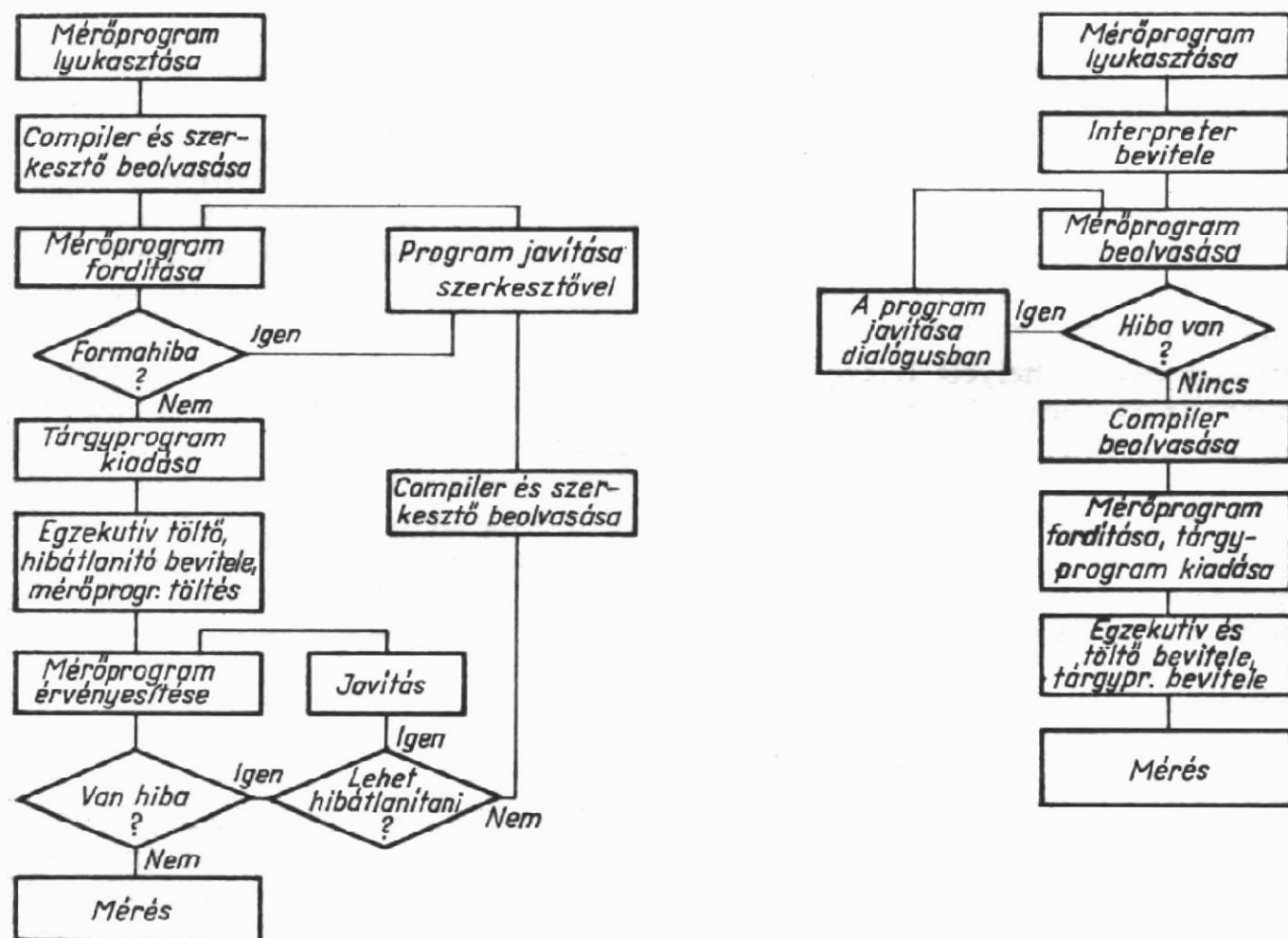
Akár primitív interpretív kód, akár BASIC szolgál bemenő nyelvként, a rendszerfüggetlen módon megfogalmazott feladatot a választott nyelvre le kell fordítani. A manuális munka elkerülése érdekében egyik megoldásként erre — az F séma szerint — kereszt-compiler alkalmazható [49].

A G séma kompromisszum, mely megalkuszik a primitív interpreternek a gépi kódhoz képest viszonylagos lassúságával, hogy ezen az áron a compiler struktúráját a kiscsége implementálhatóság követelményeinek megfelelően egyszerűsíthesse. A compiler kimenete, interpretív kód file alakjában, a rendszer lemeztárolójára kerül, majd a fordítás befejeztét követően on-line végrehajtható [33].

Utolsóként a H sémát kell megemlíteni, melynek on-line bemenő nyelveként valamely legmagasabb szintű rendszerfüggetlen, de igen szűk feladataperturájú munkanyelv szolgál.

Egyes berendezésekben interpretert alkalmaznak program előkészítésére és érvényesítésére, majd az érvényesített programot compilerrel gépi kódra fordítják, s gépi kódként futtatják.

A tiszta compiler és az interpreter+compiler séma munkamenetéről a 6.15. ábra nyújt összehasonlító áttekintést [39].



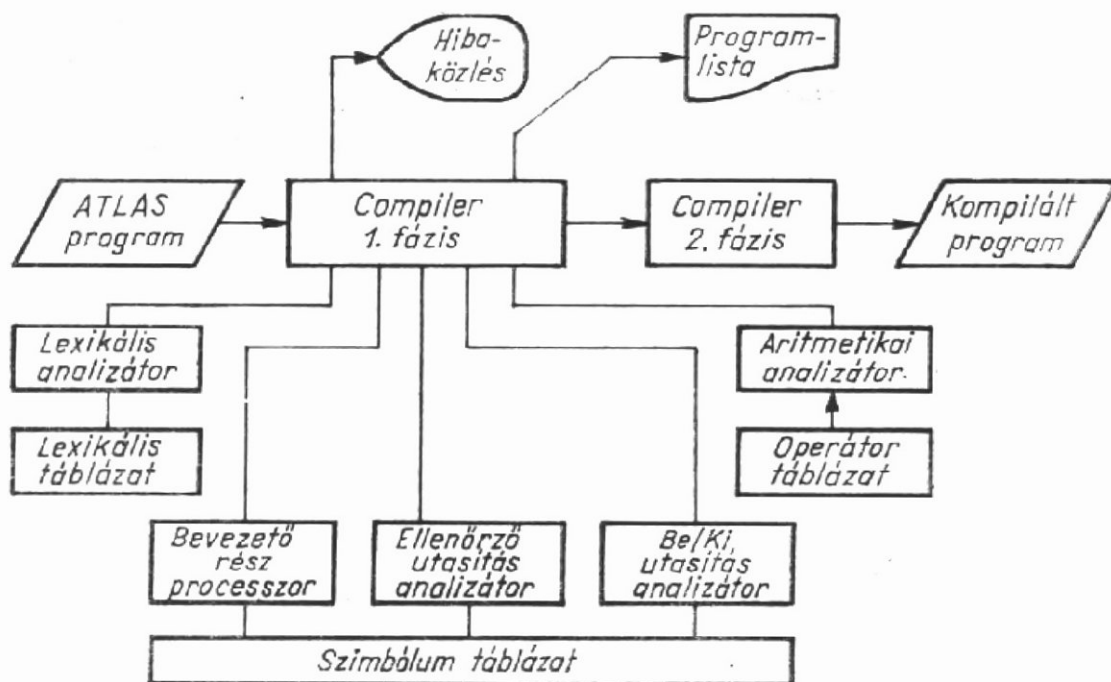
6.15. ábra. Compiler, valamint interpreter + compiler munkamenete

Időreális diszk operációs rendszer (RDOS)

A G programozási séma időreális diszk operációs rendszer keretében realizálódik [33]. RDOS átfogó file-rendszert ölel fel. A felhasználó egyszerű munkautasításokkal adhat parancsot file-ok szerkesztésére, kompilálására, ill. eltávolítására. Az összes periferikus készülék file-ként hívható és kezelhető. RDOS mind egyetlen feladatú, mind többfeladatú operációs rendszerként alkalmazható.

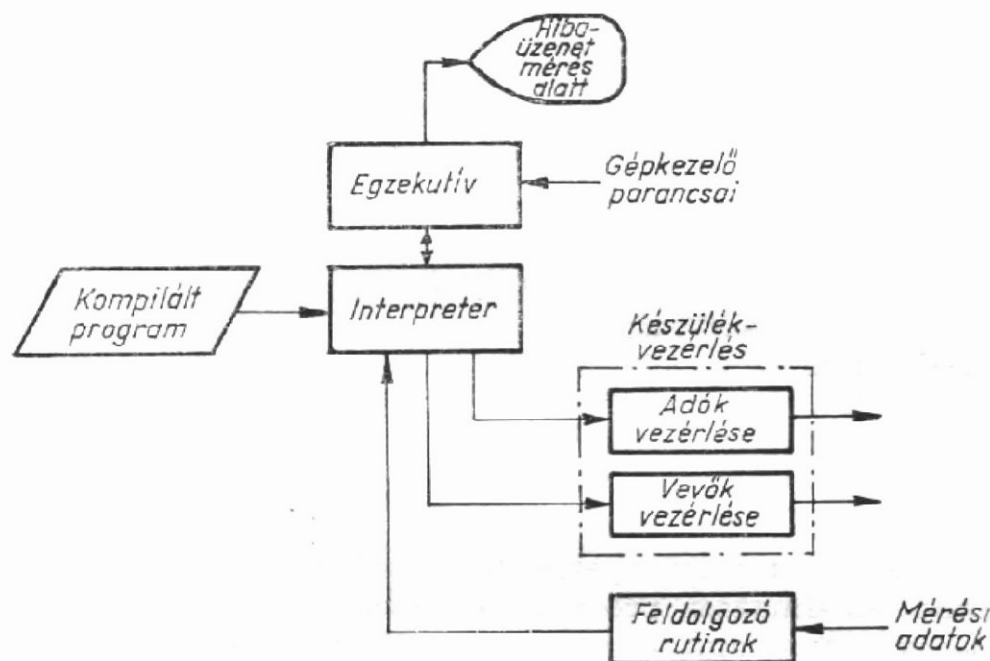
Az ATLAS compiler (6.16. ábra) első fázisa analizálja az ATLAS utasításokat és közbenső kód file-t generál. Második fázisa az utasításszámokat címezzé transzformálja, és on-line mérési programot szolgáltat. A compiler magja a figyelőprogram, mely az összes ATLAS utasítást egyenként processzálja, a mindenkor i igények megfelelő processzort aktivizálja. Az utasítások elemzése során szintaktikai vizsgálat is történik; az esetleg észlelt hibáról a compiler üzenetet küld.

A második fázis által szolgáltatott gépfüggetlen, interpretív kód formátumának struktúrája: Művelet kód, első argumentum, második argumentum, ..., N-edik argumentum.



6.16. ábra.
ATLAS—interpretív kód compiler szerkezete

Az interpretív kód a mérésvégrehajtó programrendszer (6.17. ábra) on-line bemenetére kerül. A mérésvégrehajtó rendszer a RDOS keretébe tartozó feladatként működik. Központjában interpreter áll, ez értelmezi a compiler által generált kód file-t, ill. hajtja végre a készülékek vezérlőrutinjai útján, miközben a primer mérési adatokat a feldolgozó rutinok processzálják.

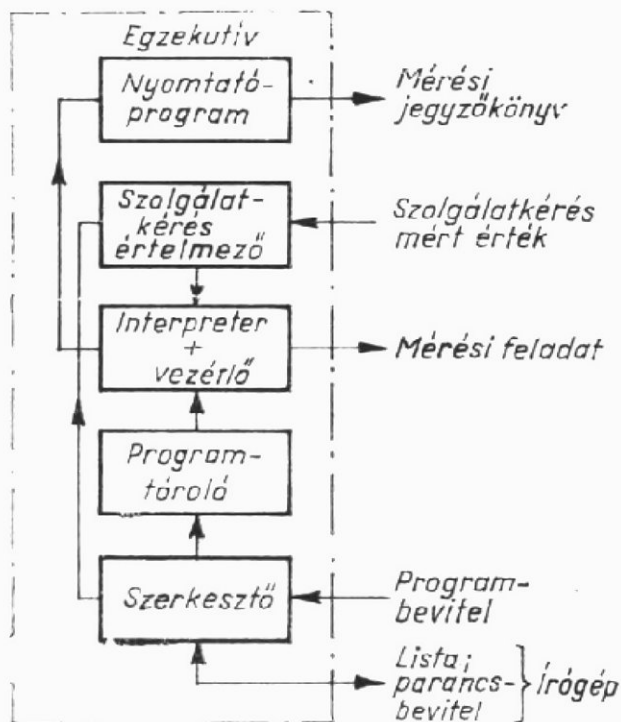


6.17. ábra.
Interpretív kód on-line végrehajtásának vázlata

Interpretív operációs rendszer

A kisméretű általános programrendszereinek tárgyalása során említettük már, hogy az interpreter végrehajtó program. Voltaképpen az, amit interpreterként említünk, nem egyetlen program, hanem programokból álló teljes operációs rendszer, mely

az alkalmazási program előkészítésével és végrehajtásával kapcsolatos minden feladatot ellát. A 6.17. ábra interpretív programvégrehajtó rendszere az interpretív operációs rendszer egyszerű példaként szolgál. Több-mérőállomású ellenőrző rendszer teljes interpretív operációs rendszerét a 6.18. ábra szemlélteti vázlatosan [51]. A rendszer közös egzekutív felügyelete alatt álló szerkesztő, interpreter + vezérlő, szolgálatkérés-értelmező és nyomtató programblokkból áll.



6.18. ábra.

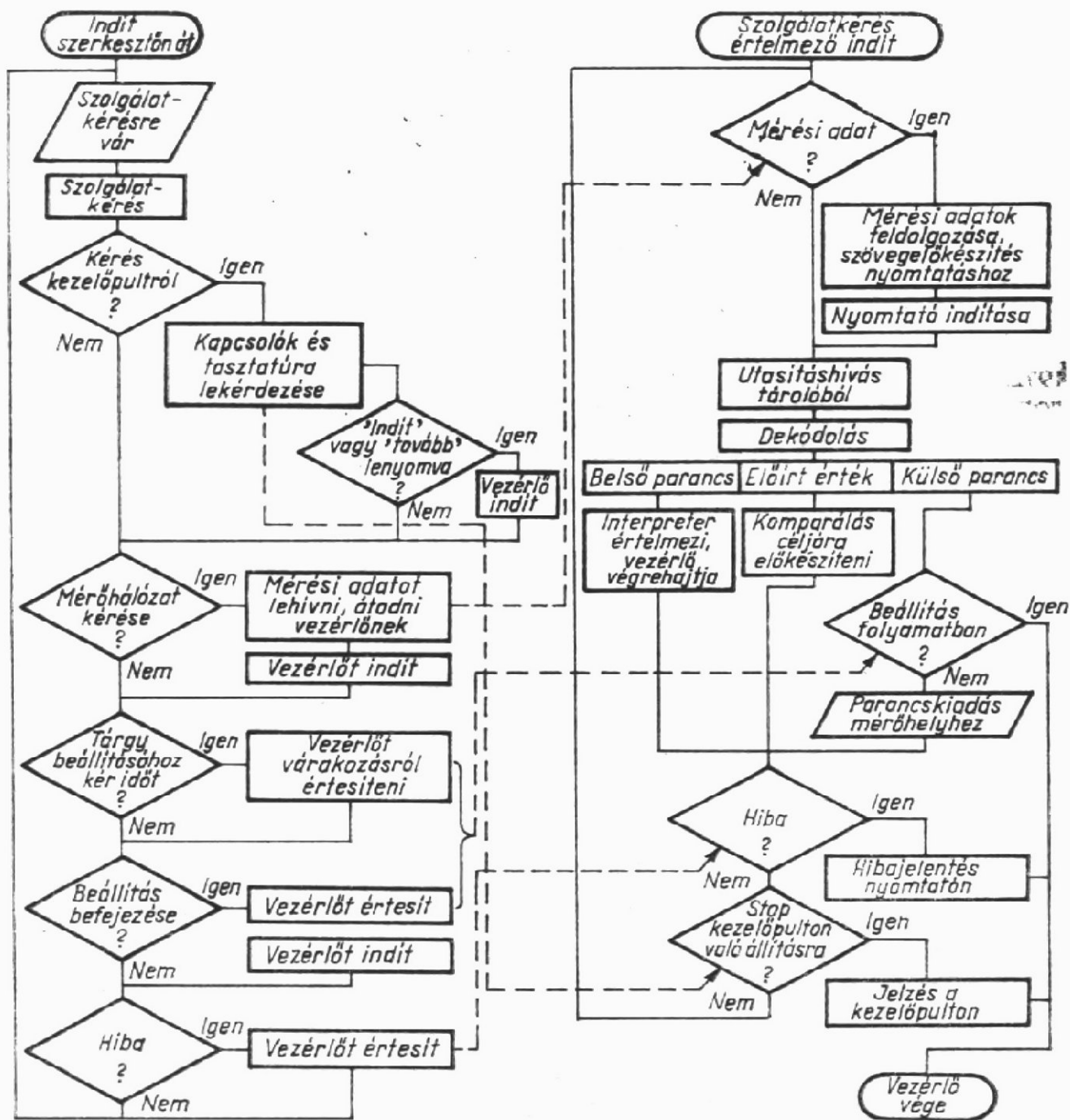
Több-mérőhelyes ellenőrző rendszer teljes interpretív operációs rendszere

A bemenő programot a szerkesztőprogram fogadja, és a programtárolóban helyezi el. Az egyes alkalmazási programok kezdő címét a szerkesztőprogram a szolgálatkérés-értelmező programmal (későbbi felhasználás céljából) közli. A szerkesztőprogram végrehajtja az írógépről adott parancsokat, melyekkel az alkalmazási program korrigálható, esetleg a programtárolóból törölhető.

A szolgálatkérés értelmezőhöz a következő típusú szolgálatkérő jelek érkehetnek: a kezelőpultról adott jelzések, mérőkészülék feladat záró jele, hibahelyzet jele. A program a kezelőpult szerveinek (kapcsolók, gombok) állapotát lekérdezéssel regisztrálja, és a vizsgálat további menetét a regisztrátum alapján szabja meg. A mérési eredményeket ugyancsak lekérdezi, majd továbbfeldolgozásra a vezérlőprogramhoz továbbítja.

Az interpreter + vezérlőprogram a mérési eljárás tulajdonképpeni végrehajtója (6.19. ábra). A végrehajtandó alkalmazási program kezdőcímét a szolgálatkérés-értelmezőtől kapja meg. Lehívja a programtárolóból az utasításokat és értelmezve, különböző rutinjai segítségével végrehajtja azokat. Belső és külső utasításokat, valamint adatokat (referenciaértékeket) kell megkülönböztetnie. A belső, pl. elágazási utasításokat közvetlenül végrehajtja, a külső parancsokat a mérőhálózatra képezi le, a referenciaértékeket a mért értékkel hasonlítja össze, a vizsgálati eredményeket a nyomtatóprogram számára előkészíti. E program a gépkezelő választása szerint dolgozhat folyamatos üzemmódban, minden utasítás után megállás vagy megállás-hiba esetén üzemmódban.

A jegyzőkönyvként kinyomtatandó szöveg tárolására két puffertároló terület áll rendelkezésre. Így lehetőség van arra, hogy míg az egyik tárolóterület tartalmának



Siemens Prüfsystems 70

6.19. ábra.

Az interpreter + vezérlő által végrehajtott mérési eljárás menete

kinyomtatása folyik, addig a másik területre a vezérlőprogram új szöveget helyezzen el. A tárolóterületek címének változtatását és a nyomtatóprogram indítását ugyancsak a vezérlőprogram végzi.

A szolgáltatás-kérés-értelmező, az interpreter + vezérlő és a nyomtatóprogram kezdő-címei az egzekutívon belül hozzáférhetők; e programok egymással szimultán futhatnak, és bizonyos közös tárolóhelyeken át egymással kommunikálhatnak. Az egyes programok eső prioritási sorrendje: 1. szerkesztő-; 2. szolgáltatás-kérés-értelmező; 3. nyomtató-; 4. interpreter + vezérlő. E sorrend alapján a mérési adatbevitel mindig prioritást élvez a kivitellel szemben.

Az operációs rendszer felépítmódja egyrészt megengedi annak bővíthetőségét, másrészt több mérőhely egyidejű használatát is. Több mérőállomás esetén az egyetlen

egzekutív, szerkesztő- és szolgálatkérés-értelmező mellett, a mérőhelyek szimultán munkájának lehetővé tétele érdekében a mérőhelyeknek megfelelő számú interpreter + vezérlő és nyomtató program szükséges.

Irodalom a 6. fejezethez

- [1] *Hooper, R. L.*: The minicomputer, a programming challenge. Fall Joint Computer Conference, 1968, 649...654. old.
- [2] *Ossanna, J. F.*: The current state of minicomputer software. Spring Joint Computer Conference, 1972, 111...118. old.
- [3] *Spencer, H. W.—Shepardson, H. P.—McGowan, L. M.*: Small computer software. IEEE Computer Group News, July/August 1970, 15...20. old.
- [4] *Ferguson, D. E.*: The software and softwhen of mini-computers. IEEE Internat. Convention Digest, March 1970, 23...26. old.
- [5] *Walks, D. J.—Kroneberg, A. B.*: The future of minicomputer programming, Spring Joint Computer Conference, 1972, 103...109. old.
- [6] *Lamb, V. S.*: All about cross-assemblers. Datamation, July 1973, 77...80. old.
- [7] *Goddard, P. J.*: Minicomputer software — an ideal, Symposium on Minicomputers, South Africa 1973, 1...7. old.
- [8] *Falk, H.*: A checkup: minicomputer software. IEEE spectrum, Febr. 1974, 52...56. old.
- [9] *St Johnston, A.*: Software for minicomputers. Software Word — Spring 1971, 51...55. old.
- [10] *Katzan, H.*: Operating systems architecture. Spring Joint Computer Conference 1970, 109...118. old.
- [11] *Mills, D. L.*: Executive systems and software development for minicomputers. Proc. of the IEEE, Vol. 61, No. 11. Nov. 1973, 1556...1562. old.
- [12] *Allan, J. J.*: Minicomputer executives for the development of real-time applications. EUROCON, Lausanne Oct. 22. 1971, 1...18. old.
- [13] *Lewis, A. D.*: Programming mini-computers in a real-time and communication environment. Software World, June 1973, 6...11. old.
- [14] *Baráth Cs.—Papp Gy.*: Mérési adatgyűjtéshez és folyamatirányításhoz kidolgozott programrendszer a VT 1010 számítógépre. Mérés és Automatika, XXII. évf. 9. sz. 1974, 337...348. old.
- [15] *Fine, P.*: Minicomputer software implementation. Software World, March 1973, 2...9. old.
- [16] *Russel, T.—Russel, J.*: MIMIC — Is simulation the answer to minicomputer software development? Software World, March 1973, 10...16. old.
- [17] *Pike, H. E.*: Software production for minicomputers, Proc. IEEE, Vol. 61. No. 11. Nov. 1973, 1544...1555. old.
- [18] *Freeman, P.*: Automating software design. 10th design automation workshop proc., IEEE Catalog Number 73 CHO 752—6C ,62...67. old.
- [19] *Liguori, F.*: The test language dilemma. Proc. of 1971 annual conf. ACM, Chicago, Aug. 3—5. 1971, 388...396. old.
- [20] *Grimm, R. A.*: The past present and future of automatic test languages. 1971 Wescon Technical Papers, 14/1,1...6. old.
- [21] *Ellis, M. T.*: The dialect concept in ATE language design. 1971 Wescon Technical Papers, 14/2, 1...8. old.
- [22] *Ellison, T. A.*: Abbreviated Test Language for Avionic Systems. 1971 Wescon Technical Papers, 14/3, 1...8. old.
- [23] Aeronautical Radio, Inc., Abbreviated Test Language For Avionic Systems (ATLAS), ARINC Specification 416—1, June 1. 1969.
- [24] Aeronautical Radio, Inc., A guide to ATLAS for specification test writers. ARINC Report 418, May 15. 1969.
- [25] "ATE Newsletters" Aeronautical Radio, Inc., Annapolis, Md.
- [26] *Anstead, J. W.—Challenger, M. T.*: The use of ATLAS by an equipment manufacturer. Proc. of the Joint Conf. on Automatic Test Systems, 14—17, Apr. 1970, Birmingham, 365...374. old.
- [27] *Ellis, M. T.*: VITAL — A general purpose, multiple dialect automatic test language. ASSC RECORD 70, 7...18. old.
- [28] *Peterson, G. L.*: BASIC the language of time sharing. Hewlett-Packard Journal, Nov. 1968, 2...8. old.
- [29] *Moley, R. M.*: BASIC at Hewlett—Packard. Hewlett—Packard Journal, Nov. 1968, 9...13. old.

- [30] *Gudniz, L.—Tsuda, H.*: General-purpose test system gets digital capability. Hewlett—Packard Journal, July 1971, 2...9. old.
- [31] A Pocket Guide to the 2100 computers. Hewlett—Packard Co. 9/72, 5951—4423.
- [32] *Sammet, J. E.*: Programming Languages: History and Fundamentals. Prentice—Hall, Inc., Englewood Cliffs, N. J. 1962.
- [33] *Kelly, J.—Toscano, P. M.*: Equate — New concepts in automatic testing. ASSC Record 72. 79...88. old.
- [34] *Mitchell, T. R.*: A standard test language — GOAL. 10th design automation workshop proc. IEEE Catalog No. 73, CHO 752—6C, 87...96. old.
- [35] *Ryle, B. L.*: The ATOLL checkout language. Datamation, Apr. 1965, 33...35. old.
- [36] ATS—960 Automated Test System. Texas Instruments TI—312B—5M—1172 (az ATS—960 technikai leírása).
- [37] Instrumentation engineering user seminar on automatic test system, 1973. "390 System software."
- [38] MMB Test Station COUNT Computer—aided Test Programming, Messerschmitt—Bölkow—Blohm sz. AK. 49E. 10. 73. 05.
- [39] *Martens, D.*: Software für automatische Prüfsysteme. Elektronik, Apr. 1974, 109...115. old.
- [40] Teradyne bulletin No. F533, K167 Relay test system.
- [41] *Kierce, G.*: Programs in Getel speak test engineer's language. Electronics, Aug. 2. 1971, 53...57. old.
- [42] *Gill, M. H.*: PROTEST: A language for testing. Computers and Automation, May 1973, 17...22. old.
- [43] Minutes Eight Workshop on Standardization of Industrial Computer Languages. Purdue University, Oct. 2—5. 1972.
- [44] A standard for real—time. Data Processing, May—June 1974, 155...157. old.
- [45] Official Definition of CORAL 66. Her Majesty's Stationery Office, London 1973.
- [46] *Brandes, J. et al.*: A real—time programming language and its application for measuring processes. IFAC 5th World Congress, June 1972. 10. 4. 1...6. old.
- [47] *Koch, G.—Martin, W. A.*: Application of a Process—oriented Language in Laboratory—Automation. IFAC/IFIP 3. Internat. Conf. 1971 — Helsinki XIII. 3. 1...8. old.
- [48] *Rákosi M.*: Digitális folyamatszámítógépek programozása. Műszaki Könyvkiadó, Budapest, 1975.
- [49] *Matelan, M. N.*: A method for conversational to interpreter test language translation. ASSC Record 70, 27...37. old.
- [50] *Meyer, R. J.*: Software considerations for computer controlled test equipment. ASSC Record 70, 38...49. old.
- [51] *Ploss, R.*: Zentrale Einheiten für Prüfautomaten des Integrierten Prüfsystems 70 mit Rechner des Siemens-Systems 300. Siemens-Z. 44. 1970, Heft. 11. 670—674. old.
- [52] *Standeven, J.—Bowden, K. F.—Edwards, D. B. G.*: An operating system for a small computer providing time—shared data collection, computing and control functions. Information Processing 68 — North-Holland Publishing Company — Amsterdam 1969, 536...541. old.
- [53] *Thomas, R. F.*: Some aspects of CAMAC software, IEEE Transactions on Nucl. Sci. NS—20/2, 1973, 50...68. old.
- [54] *Hooton, J. N.*: An introduction to the CAMAC intermediate language, IEEE Transactions on Nucl. Sci. NS—21/1, 1974, 903...908. old.

A könyvben alkalmazott szakkifejezések szótára

Megjegyzés: a gömbölyű zárójelek között az értelmezést elősegítő szavak, a szögletes zárójelek között szinonímák találhatók.

A

adaterősítő	data amplifier
adatpálya	dataway
ágfővonal	branch highway
ágparancs	branch command
ágvezérlő	branch driver
átállít	set
átfedéses címzés mód	overlapped addressing
azonos sorrendű (tár)	first in first out (store)

B

bejárat	port
beszélő	talker
bitszeletelt	bit sliced

C

cikluselvétel	cycle stealing
címmutató	address pointer
csatlakozás	interface
csatlakozási rendszer	interface system
csatlakozóegység	interface unit
csatornavezérlő	channel controller

D

diszk	disk
-------	------

E

egzekutív [szervező; felügyelő]
(program)
előrettekintő (logika)
egyprogramú [egyfeladatú]
operációs rendszer
egylövetű [egyciklusú]

executive [monitor; supervisor]

look ahead (logic)
single job [single task]
operating system
one-shot

F

fáziszáró
feladat [munka] menetirányítás
felfüggeszt [hatástalanít]
file-kezelő rutin
fordított sorrendű (tár)
forrásszerkesztő

phase locked
job scheduling
disable
utility routin
first in last out (store)
source editor [symbolic editor]

G

gyöngyvezeték

daisy chain

H

hajtó
hallgató
hibátlanító [hibaszűrő; hiba-
nyomozó]

driver
listner

debugging

I

időreális [reális idejű]
implementáció
implementál

real time
implementation
implement

J

jelző

flag

K

kapocs (regiszter)
kérdő
kereszt-assembler
kereszt-compiler
keretvezérlő
készülék (periferikus)
készülékcím
készülékintéző (program)
készülékkezelő (program)

link
interrogate
cross assembler
cross compiler
crate controller
device (peripheral)
device number
device scheduler
device handler

készülékvezérlő
kétlejtős
kontextus (program)
kontroller

kötetlen elérési sorrendű
[kötetlen elérésű]
kötött elérési sorrendű [kötött
[soros] elérésű]
küldő
kvád

L

láncolódásos címzés mód
lapka
letároló [léptető tároló]
literális

M

memóriapuffer
menetirányítás
menetirányító [intéző]
mérőerősítő
műveletvonal (processzor)

N

név (az utasításban megadott cím)
nullhiba [nullponthiba]

O

olvas/ír [változó tartalmú] (tár)
operátor (matematikai v. áramköri)

P

parancs
parolajel
peremmutató
programcserélgetés
programorientált
programrátét [programballaszt]

R

regisztertömb
rés
résjel [résimpulzus, réselő jel]
retesz (regiszter)

device controller
dual slope
context (program)
controller

random access

serial access
source [transmitter]
quad (quadruplet röv.)

interleaved addressing
chip
stack (pushdown)
literal

memory buffer [cache]
scheduling
scheduler
instrumentation amplifier
pipeline (processor)

name
zero offset

read/write (memory)
operator

command
handshake signal
stack pointer
swapping
software oriented
program overhead

register file
strobe
strobe pulse
latch

S

soros fővonal
soros-vonal-vezérlő

serial highway
serial driver-receiver

Sz

szerkezet (mint pl. óra ~)
szerkezetorientált

hardware
hardware oriented

T

társasvezeték
tartalomőrző (tár)
tartalomvesztő (tár)
többprogramú [többfeladatú]
operációs rendszer
töltőmemória

party line
nonvolatil
volatil
multiprogramming [multi task]
operations system
bootstrap memory

V

vezérlőegység
vezetett
vezető
visszaállít

control unit
slave
master
reset

A szövegben és az ábrákon alkalmazott rövidítések

A	1. alapjel; 2. alsó logikai szint; 3. erősítés; 4. erősítő; 5. alakáramkör
A—D	analóg—digitális
AE	áramellensúly
ALOP	aritmetikai—logikai operátor
AOP	átmenő operátor
ATLAS	Abbreviated Test Language for Avionics Systems
ATS	Automatic Test System
BSI	British Standard Interface
CAMAC	jelentés nélküli betűszó; az általa megjelölt csatlakozási rendszer megnevezésére
CSV	csatornavezérlő
D	digitális adat
Dp	paralel alakú digitális adat
Ds	soros alakú digitális adat
DEC	Digital Equipment Corporation
E	érzékelő (csatorna); érzékelés
ECS	egységes csatlakozó
ÉK	építőköcka
F	1. felső logikai szint; 2. szolgálatkérő forrás; 3. frekvencia
FM	frekvenciamoduláció
HP	Hewlett—Packard
I	inger (csatorna)
IEC	International Electrical Commission
K	1. készülék; 2. mérőkészülék; 3. komparátor
KV	készülékvezérlő
LA	logikai alsó szint
LF	logikai felső szint
M	memóriamodul
MPK	memória—processzor—készülék

M-R	munkaregiszter
O	operátor
P	processzor
PE	periferikus eszköz
PER	periferikus
PME	programozott mérőerősítő
PR	program, programozott
PRF	programfüggetlen
PR-K-Be/Ki	programkapcsolatú be/kimenet
p/s	paralel/soros
R	1. regiszter; 2. ellenállás; 3. rezultátum
ROV	regiszter—operátor—vezérlő
RV	rendszervezérlő
SK	sínkezelő
SZ	szabályozás
SZCS	számítógép-csatlakozó egység
SZG	számítógép
TÁMV	tároló mintavevő
V	vezérlő
Vez.	vezérlő
∧	logikai ÉS műveletjele
&	logikai ÉS áramkör jele
∨	logikai VAGY áramkör jele

Tárgymutató

A

- A, B, C alakú reed-relék 89
- ablakkomparátor 68
- abszolút címzés mód 139
- A—D átalakító 79...89
 - , feszültség — frekvencia 87
 - —, integráló 86, 87
 - —, iteratív kvantálású 83
 - —, két- és háromlejtős 86
 - — kvantálás és kódolás 81
 - —, lineáris időkonverzióval 85
 - —, logaritmikus 88
 - — osztályozása 82
 - —, paralel kvantálású 83
 - —, számláló kvantálású 84
 - —, szervo rendszerű 84
 - —, szukcesszív approximációval 84
- adagoló, mechanikai 52
- adapter 29
- adatátvitel 152, 155
 - állapotellenőrzés nélkül 155
 - állapotellenőrzéssel 156
 - -módok 152
 - utólagos ellenőrzéssel 156
 - útvonalak 152
- adatbázis, központi 53
- adaterősítő 61
- adatfeldolgozó 19
- adatgyűjtő 21
- adatközlő mérőkészülék 14
- adatmegjelenítő 15
- adathálózat 104, 128
- adatszám regiszter 163
- adatvonal 107
- aktív érzékelés 16
 - aktív mérőrendszer 21
 - , analóg mérőrendszer 38
- alapjel 16, 17
 - generátor 16
- algoritmikus vektorgenerálás 45
- algoritmus 18, 19
 - , mérési 18
- alkalmazásorientált nyelv 235
- állapottömb 117
- általános jellegű mérőhálózat 50
 - regiszterek 139
- analóg adatgyűjtő 31
 - — általános séma 31
 - — csatornaerősítőkkel 32
 - —, frekvencia-analóg 31, 36
 - —, kitérés-analóg 31, 32
 - — központi erősítővel 31
 - építőelemek 57
 - kapcsolók 89...91
 - kimenő csatorna 38
 - komparálás 46
 - komparátor 67
 - mérőhálózat 30
 - , adatgyűjtő 31
 - , aktív 38
 - , frekvencia-analóg 36
 - , integrált 42
 - , kitérés-analóg 31
 - , közvetlen szabályozó 39
 - , moduláris 40
 - , rezgésvizsgáló 42
 - multiplexer 94
 - stuktúrájú mérőkészülék 192
- anyagvizsgáló mérőrendszer 52
- anyalemez 30
- apertúra idő 31, 32

apertúra hiba 65
 áramkomparátor 68
 áramköri csomag 30
 áramkövető 57
 architektúra 137
 aritmetikai—logikai operátor 103, 120
 assembler 235, 269
 —, kereszt 236
 assembly nyelv 233
 aszinkronkapcsolat 159, 160
 aszinkron kapcsolatú memória 110
 átalakító 14, 70...78, 79...89
 —, A—D 79...89
 —, D—A 70...78
 —, jel 14
 —, modulátor 15
 —, operatív 15
 átfedésező üzemmód, memória 112
 A-típusú vezérlő, CAMAC 180
 ATLAS (nyelv) 246, 248, 249...256, 272
 — dialektusok 265
 — főnév és modifikátorai 254
 — igék 251...253
 — programszerkezet 253
 — származékok 265
 — szintaktika 253
 — szótár 249
 — utasításszerkezet 250
 átmenő operátor 103
 ATOLL (nyelv) 266
 ATS BASIC (nyelv) 256...261, 271
 — — karakterkészlet 256
 — — mátrix műveletek 260
 — — mérés-technikai kiterjesztése 260
 — — műveletek 258
 — — operátorok 257
 — — szubrutinhívás 259
 — — utasításszervezés 256
 — — változók 256
 — — digitális kiterjesztése 261...264
 autodekrementáló címzés mód 145
 automatikus letároló 139
 — tartományváltás 223
 — —, DVM 224
 — —, erősítő 223
 — —, frekvenciamérő 226
 — —, időmérő 225
 automatizált mérőrendszer 19
 autonóm csatorna 166
 — — állapotvektorok 167
 — — felépítésmód 166
 — — megszakításkérés 168
 — — munkamenet 168

autonóm csatorna szolgáltatáskérés 168
 — — üzemmódok 167
 azonos sorrendű tároló 113

B

BASIC 234, 256, 264, 271
 — dialektusok 265, 267
 — származékok 265, 267
 beavatkozó 16
 beállási idő 33, 77, 95
 bejárat, memória 109
 belső megszakításkérés 158
 — sín 129
 — szubrutin 238
 bemenő impedancia 59, 97
 — nyelvi szint 269
 bináris D—A átalakító 75
 bipoláris analóg kapcsoló 90
 — komparátor 70
 bitszelelt mikroprocesszor 120, 125
 blokkátvitel 162, 163, 165, 185, 187
 —, CAMAC 185, 187
 —, egyszeres 163
 —, multiplex 163
 —, programfüggetlen 165
 —, többszörös 163
 blokkorientált nyelvi szint 246
 BSI 173
 byte átvitel, HP—IEC 175
 — -orientált szószervezésű gép 137, 138, 143
 — -szervezésű számítógép 137, 138, 143

C

cache tároló 113
 CAMAC 28, 173, 174, 179...190
 — adatpálya 179, 181
 — adatpálya parancsnok 182
 — ágfővonal 185
 — ágparancsnok 186
 — állapotvizsgálat 185
 — A-típusú vezérlő 180
 — blokkátvitel adatpályán 185
 — — az ágon 187
 — felépítés 179, 180
 — figyelemkérés (LAM) 184
 — fővonal megvalósításmód 188
 — gépi kapcsolat 189
 — soros fővonal 187
 — szolgáltatáskérés az ágon 187
 — szubrutinok 268
 — U-típusú vezérlő 180, 189
 címmutató 139, 163
 címvonal 108

címzés mód 139, 140, 141, 145
CMOS analóg kapcsoló 91
compiler 236, 269, 271
— orientált gép 121, 122, 137

CS

csatlakozás 169
—, CAMAC 28, 173, 174, 179... 190
—, egységes angol (BSI) 169, 172, 173
— — rendszerei 172
—, HP—IEC 174, 175... 179
— meghatározói 171
—, mérőrendszer 171
—, SIAK, SIAL 173
—, számítógép 152
csatlakozóegység 126, 218
—, HP—IEC 220
csatorna 153
—, autonóm 166
—, programfüggetlen 161
—, programozott 153
csatornaerősítő 32
csatorna korrekció 32
csatornamemória 166
csatornaprocesszor 126, 127
csatorna rendszerek, számítógép 152
csatorna vezérlő 28, 130, 131, 164, 165
csonka sín 129
csúcsdetektor 66
—, invertáló 67
—, nem invertáló 67

D

D—A átalakító 70... 78
— — ellenálláshálózatai 75
— — funkcionális jellemzői 70
— — kódtípusai 73
— —, közvetett 78
— — osztályai 74
— —, párhuzamos közvetlen 74
— —, soros közvetlen 77
dedikált vonal 152
deviáció mérő 201
diagnosztika 44
digitális ATS BASIC 261... 264
— jelszintetizáló 216
— mérőrendszer 43
— multiplexer 34, 97, 98, 99
— szerkezetű impulzusgenerátor 209
— — —, moduláris 210
dinamikus mérőáramkör 48
— mikroprogramozás 121, 137
— rezgés vizsgáló 42

diszk-operációs rendszer 241
— — —, időreális 272
Diesel-motor vizsgálat 42
DVM 221, 224
— illesztése 221
— automatikus tartományváltása 224

E

egzekutív program 239
egyakkumulátoros számítógép 139
egybefüggő memória 111
egyenáramforrás, programozott 206
egyenáramú paraméter 48
— mérése 48
egyenfeszültség-forrás, programozott 204
egylapkás processzor 124
egyprogramú operációs rendszer 242
egységes csatlakozási rendszer 172
egysínű MPK hálózat 128, 129
— processzor 122
elérési idő, memória 110
elfajult háromsínű rendszer 123
ellenállás, programozható 196
ellenálláshíd, programozható 199
ellenőrző készülék 14, 15
— rendszer 17
eloszlott vezérlés 22
előkiválasztás, megszakításkérés 157
előrettekintő (processzor) logika 119
építőköcska 27, 195
eredménymemória 49
erősítő 57... 64, 203, 223
—, adat 61
—, komparátor 70
—, mérő 61
—, műveleti 57
—, önbeállító 223
—, programozható 202
—, szaggató stabilizált 62
—, szigetelt bemenetű 62
—, töltésérzékeny 64
—, varaktorhíd bemenetű 63
értékképző 15
érték-összehasonlító 15
érzékelés 16, 17
érzékelő 14
— átalakító 15
explicit vezérlés 104, 107, 109
explicite címzett regiszter 139

F

feladatapertúra 246
feladatorientált mérőrendszer 23

félautomatizált mérőrendszer 19
 felsőbb szintű nyelv 234
 felügyelő program 239
 félvezető analóg kapcsolóelemek 90
 — kapcsoló 89
 feszültség—frekvencia átalakító 87
 file-kezelő rutin 239
 FOCAL nyelv 234
 fordított sorrendű tároló 113
 forrásregiszter 103
 forrásszerkesztő 238
 FORTRAN (nyelv) 234
 —, ipari 268
 főmemória 109
 frekvencia-analóg adatgyűjtő 31
 — —, mérőhálózat 37
 frekvencia-analóg jel 36
 frekvenciamérő automatikus tartományváltása 226
 frekvenciaosztó, programozható 214
 frekvenciasokszorozó, programozható 215
 — szintetizáló, programozható 215
 funkcionális ellenőrzés 43
 — modularitás 125
 független prioritási szint 133

G—GY

gázturbina-kompresszor ellenőrző rendszer 35
 gépcsatlakozás 169
 gépcsatlakozó 26, 28
 gépi architektúra 137
 — irányítás, mérőhálózaté 192
 GETEL (nyelv) 267
 GOAL (nyelv) 265, 266
 gyors kisegítő memória 116
 gyöngyvezetékek 154

H

hajtó/terhelő áramkör 46
 hálózati kapcsolat 159
 — konfliktus 132
 hangfrekvenciás spektrumanalizátor 42
 háromsín rendszerű processzor 118, 123
 — — —, elfajult 118, 123
 háromsínű MPK hálózat 128, 129
 hatásvonal 17
 hibátlanító program 239
 horizontális jellegű operátor 120
 — vezérlésmód 104
 HP—IEC csatlakozás 174, 175...179
 — — alkalmazás 178
 — — byte átvitel 175
 — — kezelésmód választás 178

HP—IEC csatlakozás parancsok 177
 — — sín 175
 — — szerkezeti paraméterek 178
 — — szolgálatkérés kezelés 178
 — csatlakozóegység 220, 222

I

IDA (nyelv) 266
 időmérő automatikus tartományváltása 225
 időosztás 29
 időosztásos mérőrendszer 52
 időreális diszk operációs rendszer 272
 — operációs rendszer 244
 időtranszformáció puffertárolóval 227
 IEC (HP) csatlakozás 174, 175...179
 — csatlakozóegység 220, 222
 implicite címzett regiszter 139
 impulzusgenerátor 207...210
 —, klasszikus 207
 —, módosított analóg szerkezetű 207
 —, digitális szerkezetű 209
 —, moduláris 210
 inger 17
 integrált adatgyűjtő 42
 — mérőhálózat 27
 — mérőrendszer 18
 interpreter 237, 271
 interpretív operációs rendszer 273, 274 ..276
 inverz-átalakító 16

J

jel 14
 — -átalakító 14
 — -érzékelő 14
 — -képzés 14
 — -generátor 16
 — -vezérelhetőség 194
 jó/nem jó ellenőrzés 14, 43

K

kalkulátor 22
 kapcsolat 131
 —, készülék—készülék 132
 —, készülék—memória 132
 —, készülék—számítógép 152
 —, MPK hálózatbeli 131
 —, processzor—készülék 132
 —, processzor—memória 132
 kapcsolat- és csatlakozásmód 152
 kapcsolatkezdeményezés, MPK 131
 kapcsoló 29, 89
 —, analóg 89

kapcsoló, fa 29
 —, koaxiális 29
 —, mátrix 29
 — rendszer 89
 kapcsolótömb 93
 —, koaxiális 93
 —, mátrix 93
 —, multiplexer 93
 kérdő vektor 43
 kereszt-assembler 236
 kereszt-compiler 236, 269
 készülék, periférikus 126, 127
 — -cím 154
 — -egzekutív 241
 — -intéző 242
 — -kezelő 242
 — szintű modularitás 27, 40
 — -vezérlő 28, 126, 153, 218
 — -vonatközös szubrutin 238
 kétsínű MPK hálózat 128
 — processzor 118, 122
 kiegészítő operátor 121
 — processzor 121
 kódolás 81
 kódolt vezérlés 107, 109
 kódtípusok 73
 komparálás 46
 komparátor 67, 208, 211, 212
 —, analóg 67...70
 —, digitális 208
 —, bináris 213
 kompatibilitás 194
 kondicionálás 31
 kontextusváltás 140
 kontroller 22
 kötetlen programozás 20, 25
 kötött programozás 20, 104
 közbenső tároló 48, 112, 113
 közös feszültségnyomás 33, 60
 — feszültségtűrés 97
 központi adatbázis 53
 — egység diszpécser 243
 — processzor 116
 központosított vezérlés 22
 közvetett címzés mód 140
 közvetlen memóriakapcsolat 133
 — szabályozó mérőrendszer 22, 39
 kriptokód 271
 külső megszakítás 158
 — sín 129
 — szubrutin 238
 — utasítás 121
 kvantálás 81

L

laboratóriumi anyagvizsgáló 52
 láncolódásos üzemmód, memória 112
 lapszervezés 139
 letároló 113
 — mozgóperemű 114
 — műveletek 113
 — rögzített peremű 114
 létrahálózat 75
 linearizáló puffertároló 229

M

makro 269
 makroprocesszor 235
 manuális mérőrendszer 19
 másoló írású regiszter 103
 mechanikai adagoló 52
 megszakítási rutin 158
 megszakításkérés 158, 168
 megszakítórendszer 156
 memória 109...112
 —, aszinkron kapcsolatú 110
 —, átfedéses üzemmódú 112
 — bejárat 109
 — elemi műveletei 110
 — elérési idő 110
 —, egybefüggő 111
 —, fő 109
 —, gyors kisegítő 116
 —, láncolódásos üzemmódú 112
 — modularitása 110
 — —, olvas/ír 109
 — —, puffer 113
 — —, RAM 109
 — —, rekeszenként független 111
 — — szervezőmódjai 111
 — —, szinkron kapcsolatú 110
 — —, tartalomőrző 112
 — —, tartalomvesztő 112
 — —, töltő 115
 — —, virtuális 145
 memóriakezelés szegmentálással 141
 memóriarezidens szubrutin 238
 mérési algoritmus 18
 mérésorientált nyelvi szint 246
 méréspontváltó 29
 mérőblokk 27
 mérőerősítő 60
 mérőhálózat 19
 — — adókörei 38
 — — alapelemei 57
 — —, általános jellegű 50
 — —, digitális 43

mérőhálózat felépítése 26
 — —, integrált 27
 — — memóriájának bővítése 227
 — — mikroprogram-vezérlése 229
 — —, moduláris 27
 — — programirányítása 195
 — —, több mérőhelyes 30, 50
 mérőkészülék 14, 15, 195, 218
 — —, adatközlő 14
 — —, közvetlen 15
 —, moduláris 195
 — programirányítása 195
 — rendszer-illesztése 218
 —, szintkiegyenlítő 15
 mérőpakli 29, 52
 mérőrendszer 14, 16...19, 22, 23
 —, aktív analóg 38
 —, általános jellegű 50
 —, automatizált 19
 — automatizáltságának mértéke 22
 —, digitális 43
 — elemei közötti csatlakozás 170
 —, feladatorientált 23
 —, félautomatizált 19
 —, időosztásos 52
 —, integrált 18
 —, közvetlen szabályozó analóg 39
 —, manuális 19
 —, moduláris 18
 — programellátása 245
 — programozási nyelve 245
 —, repülőgép-ipari 38
 —, részlegesen automatizált 22
 —, több mérőhelyes 51, 274
 MÍ—BASIC (nyelv) 267
 mikroprocesszor 124
 — egylapkás 124
 — funkciói mérőhálózatban 230
 — többlapkás 124
 — — bitszeletelt 124
 mikroprogramozott szógenerátor 229
 — vezérlő 106, 107
 mikroszámítógép 22, 23
 mikroutasítás 102
 — szintű processzor 119, 137
 minimum regiszterkészlet 116
 mintázat generálás 49
 — memória 49
 moduláris mérőhálózat 27
 — mérőrendszer 18
 — processzor 121, 125
 — relé 93
 modulátor-átalakító 15

monotónia hiba, D—A átalakító 72
 MPK és ROV struktúra kapcsolata 128
 MPK hálózat 128
 — —, csonka sínű 129
 — —, egysínű 128
 — — elemeinek kapcsolata 131
 — —, háromsínű 128
 — —, kétsínű 128
 — —, moduláris 134
 — —, teljes sínű 129
 — —, többsín rendszerű moduláris 135
 multiplex csatorna 164, 167
 multiplexelés 30
 — frekvencia osztással 37
 — hely és idő szerint 30
 — — szerint 30
 — idő szerint 32
 multiplexer 93...99
 —, analóg 94
 —, digitális 97...99
 —, egykapcsolópontú 94
 — előcímezése 33
 — hálózatok 95
 —, kaszkád 96
 —, kettős és hármas 33
 —, kisjelű 94
 —, moduláris 35, 96, 98
 —, optikailag csatolt 95
 —, pneumatikus 100
 —, többszintű 96, 99
 —, többszörös kapcsolópontú 94
 multiplexer-szógenerátor 210
 multiplex mérés 30
 MULTI USER SD BASIC 267
 működtetésmód, számítógép 240
 műveletek, gépi 102
 műveleti erősítő 57...60
 —, áramkövető 57
 —, nem invertáló 57
 —, szimmetrikus 59
 műveleti vonal 108

N—NY

nem roncsoló olvasás 103
 nyelv 233
 —, ALGOL 268
 —, alkalmazás orientált 235
 —, assembly 233
 —, ATLAS 246, 249...256, 272
 —, ATOLL 266
 —, BASIC 234, 256...264, 267, 271
 —, CORAL—66 268
 —, COUNT 266

nyelv dialektusai 247
—, felsőbb szintű 234
—, FOCAL 234
—, FORTRAN 234
—, GETEL 267
—, GOAL 265, 266
—, IDA 266
— implementációja 233
—, munkaleíró 267
—, PEARLE 268
—, PROTEST 267
—, Purdue FORTRAN 268
—, SESAM II 266
—, VITAL 247, 265
nyitott struktúrájú processzor 125
nullátmenet detektor 68

O—Ö

on-line nyelvi szint 269
operációs rendszer (működtetésmód) 240
— —, diszk 241
— —, egyprogramú 242
— —, időreális 244, 272
— —, interpretív 273
— —, manuális 240
— —, nem önellátó 240
— —, önellátó 240
— —, többprogramú 242
operatív átalakító 15
— regiszter 103
operátor 103, 120
optikai csatoló 62, 95
önbeállító erősítő 223

P

párhuzamos—soros átalakító 98
paraméter-mérés 48
passzív érzékelés 16
PEARL (nyelv) 268
periferikus processzor 50, 126
piko művelet 102
ponttól—ponthoz szervezésű adathálózat 103
prioritási hierarchia 132, 158
— osztály 133
— szint 133
prioritásrang 133
processzor 116
— állapotömbje 117
—, bitszeletelt 120
—, compiler orientált 122, 137
—, egysín rendszerű 117, 122
—, elfajzott háromsín rendszerű 123
—, előrettekintő logikájú 119

processzor, háromsín rendszerű 123
—, kétsín rendszerű 118, 122
—, kétszintű 122, 137
— kommunikációs kapcsolatai 124
—, központi 116
—, mikro 124
—, mikroutasítás szintű 119, 137
—, minimum regiszterkészletű 116
—, moduláris 121, 125
—, periferikus 126
—, ponttól—ponthoz szervezésű 117, 122
— regiszterkészlete 116
— végrehajtási szintjei 121, 137
programellátás, mérőrendszeré 245
programfüggetlen csatorna 161...166
— átvitel módjai 162
— felépítése 163
— munkamenete 165
— útvonalai 161
programirányíthatóság 192, 194
programozási sémák, mérőrendszer 269
programozható ellenállás 196
— kontroller 23
programozott csatorna 153...161
— adatátvitel változatai 155
— kapcsolatlétesítés 159
— megszakítórendszer 156
— utasításai 155
Purdue FORTRAN 268

R

radiális hálózat 104, 108, 130
—, vezérlő 104
—, szolgálatkérő 108
RDOS 272
reed-relé 89
referencia áramkör, digitális 45
regiszter 103
regisztercímezés 144
regisztertömb 115
rekesz, memória 110
relatív címezés 140
— hiba 72
relé kimenet 38
rendeltetési regiszter 103
rendszerfüggetlen nyelvi szint 269
rendszer gépi nyelvi szint 247
rendszervezérlő 19
rendszerorientált számítógép 102
részenként független memória 111
reteszelt multiplex üzemmód 167
roncsoló olvasás 103
ROV szintű adathálózat 104

S

SD BASIC (nyelv) 267
SESAM II (nyelv) 266
SIAK, SIAL csatlakozás 173
sinszervezés 103, 104, 108, 122, 152
sinvezérlő 153
skálahiba 72, 79
skálatényező 71, 79
soros közvetlen D—A átalakító 77
spektrumanalizátor 42
sugárvezeték 154
súlyozott hálózat 75

SZ

szabályozás 16, 17
szaggató-stabilizált erősítő 62
számítógép 102
—, byte orientált 143
—, byte szervezésű 143
—, szószervezésű 140
—, vegyes szervezésű 145
számítógépes rezgésvizsgáló 42, 43
számítógép-irányítás 21
— — sajátosságai 25
számítógép—készülék kapcsolat 152
számítógép névtartománya 139
szelektív üzemmód 167
szelektor csatorna 164
szerkezeti referencia, digitális 45
szigetelt bemenetű erősítő 62
szimmetrikus erősítő 59
szimulációs rendszer, digitális 46
szinkron kapcsolat 110, 159
szógenerátor 48, 210
—, soros 210
—, paralel 211
szolgálati vonal 108
szolgáltatás 132, 168
szolgáltatás-kérő vonal 108
szószervezésű számítógép 137, 140
szorzó D—A átalakító 78
szubrutin 238
szubrutin kezelés 147

T

tanuló rendszer, digitális 46
tárgycsatlakozó 26, 29
tárgylemez 30
tárgyorientált mérési algoritmus 18

tárgyorientált nyelvi szint 246
tároló-mintavevő 32, 64
tárolt referencia 46
társasvezeték 154
technológiai modularitás
teljes sín 129
teljes vizsgálat 45
tisztá eljárás 147
többlapkás mikroprocesszor 124
többmérőhelyes mérőrendszer 30, 51, 274
többprogramú felügyelő 243
többprogramú operációs rendszer 242
töltésérzékeny erősítő 64
töltő memória 115
töltő program 237

U

unipoláris átalakító 70, 79
—, A—D 79
—, D—A 70
utility rutin 239
U vezérlő, CAMAC 180, 189

V

válaszvektor 43
váltakozó áramú programozott tápforrás 207
— feszültség elnyomási tényező 87
varaktorhíd bemenetű erősítő 63
végrehajtó egység 219
vegyesszervezésű számítógép 138, 145
vektorgenerátor 48
vertikális jellegű operátor 120
vertikális vezérlés 105
vezérlés 104, 107, 109
vezérlő 105
—, mikroprogramozott 105
—, tárolt programú 105
— -egység 219
— -hálózat 104, 128
— memória 106
— -utasítás 167
— -vonal 108
virtuális memória 145
visszahelyező vektorsorozat 46
VITAL (nyelv) 265
— dialektusok 247, 265
Z
zárt struktúrájú processzor

Műszaki vezető: Hegedűs Ernő — Műszaki szerkesztő: Molnár József
Borító és kötéstervező: Pázdán Zsuzsa
A könyv ábráit rajzolta: Illés József
A könyv formátuma: B5 — Ívterjedelem: 25,5 (A5)
Ábrák száma: 179 — Példányszám: 2150
Papír minősége: 80 g ofszet — Betűcsalád és méret: gm/gm New Times ant. monó
Azonosítási szám: 60 647 — MŰ: 2376 — k — 7679
Készült az MSz 5601-59 és 5602-55 szerint

Dr. Sebestyén ■ Számítógépirányítású mérőrendszerek

Dr. Sebestyén Béla

Számítógép, irányítású mérő, rendszerek

Műszaki Könyvkiadó