

MPDS, a tárolt program vezérlésű távbeszélőközpontok programfejlesztő rendszere

DR. DARABOS
ZOLTÁN —
KISS ENDRE —
SCHULTZ
KRISZTINA
BHG

A korszerű távbeszélőközpontok többsége számítógépes vezérlésű. A központgyártó cégek berendezéseik vezérlésére legtöbbször saját fejlesztésű kissetítőgépet vagy processzort alkalmaznak. A BHG Híradástechnikai Vállalat tárolt program vezérlésű központjaiban saját fejlesztésű, speciális távbeszélőközpont-vezérlő processzort, a MAT 512 típusú processzort használ fel. Ezért a BHG-nak kellett gondoskodnia a fejlesztőrendszer kialakításáról is.

A számítástechnika ipari bevezetésének kezdeti szakaszára jellemző, hogy a számítógépvezérelt berendezések software-ének előállítási, dokumentálási, megbízhatósági és karbantartási körülményeire nem fordítanak kellő figyelmet. A fenti szempontok elhanyagolása általában jelentős, a termék kibocsátása után keletkező, utólagos költségek és megmagyarázhatatlannak tűnő határidőcsúszások formájában jelentkeznek. Ezek a szempontok különösen nem mellőzhetők olyan termékek fejlesztésekor, melyeket hosszú élettartamra terveznek, és megbízhatósági követelményeik szigorúak.

A BHG Fejlesztési Intézetben létrehozott MPDS (MAT 512 Program Development System) programfejlesztő rendszer egy többprocesszoros, interaktív programfejlesztésre is lehetőséget biztosító berendezés, melyen jelenleg 6 programozó dolgozhat egyidejűleg a programfejlesztés különböző fázisaiban. Az MPDS a mikroprocesszoros fejlesztőrendszerek legfontosabb sajátosságait teszi elérhetővé a MAT 512 alapú rendszerek fejlesztésekor.

1. Az MPDS rendszer hardware felépítése

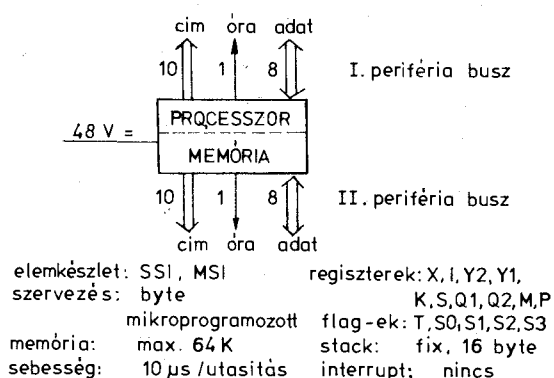
Az MPDS rendszer elsősorban a BHG MAT 512 processzorára írt programok leírására, belövésére és dokumentálására szolgál. A fenti funkcióján felül, sokoldalú csatlakozási felületei és I/O eszközei révén, berendezések programozott tesztelésénél és tetszőleges adatelőkészítési, illetve leírás-készítési feladatoknál is használható. A rendszerben egy TI kissetítőgépet és MAT 512 processzorokat alkalmaztunk. A kissetítőgép ismertetése nem célunk, és meg is haladna jelen cikk kereteit, ezért részletesebben csak a MAT 512 processzorral foglalkozunk.

1.1. A MAT 512 processzor és környezete

Az SSI és MSI elemekből felépített MAT 512 távbeszélőközpont-vezérlő processzort a BHG a 70-es években fejlesztette ki. Az 1. ábrán a processzor külső kapcsolatait és legfontosabb adatait tüntettük fel. A processzor tápellátását a központokban alkalmazott 48 V egyenfeszültségről DC/DC konverterekkel oldották meg.

Látható, hogy a processzor kettő, egymástól függetlenül működtethető perifériabusszal rendelkezik. A perifériabuszok kétirányú kommunikációt biztosítanak a központperifériákkal és különböző karbantartó eszközökkel. A perifériák a 10 bites cibusz dekódolásával határozzák meg, hogy írás vagy olvasás műveletet hajt-e végre a vezérlő. A processzor hardware interrupt-tal nem rendelkezik, a környezeti változások figyelése a 10 ms-os időalappal működő S0—S2 időzítőflag-eket kezelő utasításokkal programból biztosítható. A 10 ms-os ciklus a távbeszélőközpontokban előforduló események kezelésére elegendő. A hardware fix méretű stack-kel rendelkezik, amely nyolc szubrutin egymásba skatulyázását teszi lehetővé.

A processzor utasításkészletének gépi kód szerinti megoszlása látható a 2. ábrán. A MAT 512 utasítás-



B 232-1

1. ábra. A MAT 512 processzor főbb adatai

készletének jellegzetessége a memória szegmensenkénti (256 byte) kezelése, valamint speciális, mikroprogramban megvalósított műveletek jelenléte. Az első sajátosság azt jelenti, hogy a memóriában levő változókra 1 byte-os címekkel lehet hivatkozni úgy, hogy a változók címének felső byte-ját az Y jelű szegmensregiszterekkel képezi a gép. Ez a memória hatékony kihasználását biztosíthatja, de nem kevés programozási nehézséget okoz. Az utasításkészletben alkalmazott címzési módok és speciális műveletek az alábbiakban foglalhatók össze:

Műveletek:

Aritmetikai: +, +1, -1; logikai: \wedge , \vee , ∇ , -
 Speciális: ROTATE, SHIFT, SELECT, CODE, DECODE

Címzési módok: regiszter,
 immediate,
 direkt,
 indexelt,
 indirekt (indexelt).

A processzor programjai a fenti utasításkészletet magába foglaló MATASSY assembly nyelven készülnek.

Az assembly nyelv tartalmaz olyan direktívákat is, melyekkel programmodulonként linkelhető kódot állít elő a fordítóprogram. Ez a nyelv egy előnyös tulajdonságának bizonyult a központprogramok fejlesztésénél, több programozó munkájának összehangolásánál.

1.2 Az MPDS rendszer moduljainak kapcsolata

Az MPDS rendszer moduláris felépítésű. A rendszer

- 1 db főmodulból (TI kisszámítógép), és
- N db almodulból (MAT 512 processzor)

áll.

Az almodulok száma az igények szerint változtatható (jelenleg 5). Az almodulok, kihasználva a MAT 512 processzorok kettős buszrendszerét, a II. perifériabusz révén egy közös buszon át kapcsolódnak össze egymással és a főmodullal. Erre a közös buszra kapcsolódnak mindazok a perifériák, amelyek valamennyi almodul számára hozzáférhetőek (egy lyukszalag lyukasztó-olvasó egység és egy DZM mátrixnyomtató).

Valamennyi almodulhoz egy tv-display kapcsolódik, az egyikhez egy PROM programozó berendezés is rendelkezésre áll. A főmodulban oldjuk meg a nagyobb méretű nyomtatási feladatokat, és ott elégítjük ki a rendszer háttértár igényét, mely célra elsősorban floppy disc-eket használunk.

1.3 Információátvitel az MPDS rendszerben

Az MPDS rendszer moduljait összekötő közös buszt egy időben csak egy almodul használhatja, és ezenkívül az adatáramlás irányára is vonatkoznak megkötések lásd a 3. ábrát.

A közös buszon csak főmodul – almodul, valamint

a gépi kód %-ában

1		üres utasítás
1.5		szubrutin hívó
2		időzítő, indirekció
2		hosszu ugró
8		szegmens beállító
9		input/output
16.5		rövid ugró
60		adatmozgató és műveletvégző

B 232-2

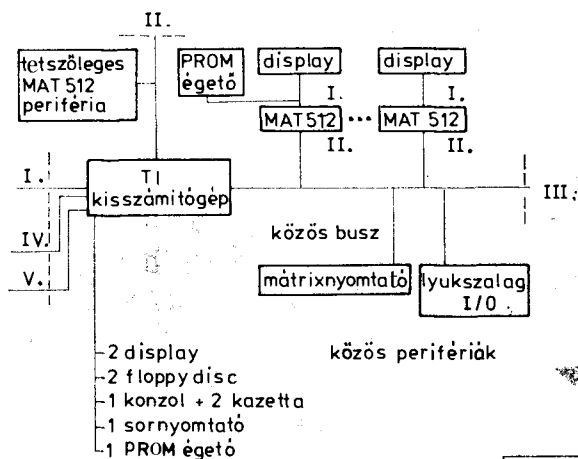
2. ábra. Az utasításkészlet megoszlása

almodul – közös perifériák viszonylatban lehetséges kétirányú kapcsolat. Az almodulok közötti átvitelt, ha szükséges, lyukszalagon, a főmodul valamely háttértárán, vagy a főmodul memóriáján keresztül lehet megoldani. Ez utóbbi a leggyakoribb.

1.4. Az MPDS és környezete

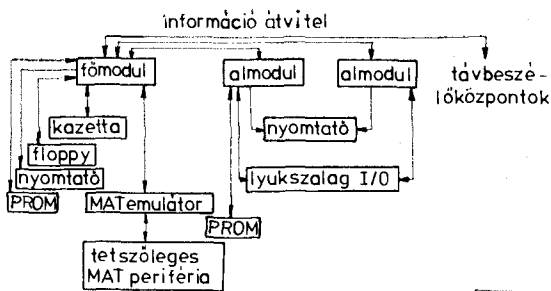
Az MPDS sokoldalú kapcsolatot tart fenn környezetével (lásd 4. és 5. ábrákat). Ez a kapcsolat egyrészt különböző adathordozókon (floppy, kazetta, lyukszalag, PROM, listák) és display-ken keresztül, másrészt elektronikus interface-eken keresztül valósul meg.

A MAT 512 alkalmazói környezetébe kétféle csatlakozási felület vezet (III. és II.). A III. felület maga a közös busz, melyre további almodulok, illetve közvetlenül a célberendezést jelentő központvezérlők kapcsolhatók, így egyszerűen megoldva a programbetöltés problémáját. A II. felületen a MAT 512 perifériabuszait állítottuk elő. Így a főmodulban futó program segítségével tetszőleges, a MAT 512-höz kifejlesztett, vagy kifejlesztendő periféria illeszthető a kisszámítógéphez teszt vagy egyéb célokból. Az I. felület bitsoros adatátvitelt biztosít, melyen keresztül a főmodulban futó programból 4096 input és 4096 output bit bitenkénti manipulációjára nyílik lehetőség.



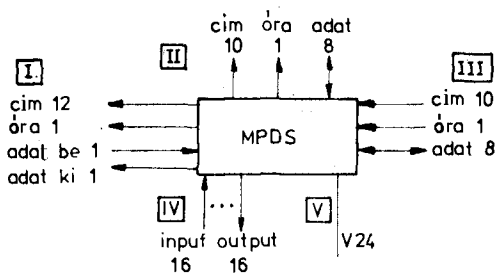
B 232-3

3. ábra. Az MPDS rendszer moduljai



B 232-4

4. ábra. A modulok közti információátvitel



B 232-5

5. ábra. Az MPDS csatlakozási felületei

A legtöbb periféria egyszerűen illeszthető az univerzális párhuzamos (IV.), vagy a V. 24-es (V.) interface-en keresztül.

2. Az MPDS programrendszere

Az MPDS rendszer egyes moduljai között munkamegosztást valósítottunk meg. Az almodulokban történik a programok, leírások előkészítése és a programmodulok belövése, a főmodulban pedig a programok fordítása, linkelése, háttértáron való tárolása, valamint a nyomtatás. A rendszerfunkciók ilyen kiosztását egyrészt a takarékos eszközfelhasználás, másrészt az a tény indokolja, hogy a főmodul általános jellege miatt könnyebben és gyorsabban programozható bonyolultabb adatfeldolgozási feladatok elvégzésére.

A rendszer valamennyi programja interaktív kapcsolatot tart fenn a kezelővel, aki maga a programozó, így lehetőség van arra, hogy a jelzett hibákat azonnal javítani lehessen. Az ily módon felgyorsított fejlesztési ciklus a régebben alkalmazott batch fejlesztéshez képest, amely R20 számítógépen folyt, minőségi változást hozott a központ programrendszerének előállításában, beleértve a labor és gyártás-orientált programcsomagok létrehozását is.

2.1 Az MPDS modulok szolgáltatásai

Az MPDS rendszerben a főmodul—almodul viszonylatban funkcionális, az almodul—almodul viszonylatban pedig terhelésmegosztást alkalmaztunk. Ezt tükrözik az egyes modulok programjai is:

Modul	Programnév	Méret	
		(forrásor)	
Főmodul	TXMONITOR	4000	
	TXEDIT	3000	
	DRBASM/XREF/SYMTAB	8000	
	MATLINK	3000	
	OBJMAN	1500	
Almodulok	DRBMON	2000	
	HIPO/M	5000	
	SMART TRACE	3000	
	DPROG	2000	
	ANAMAT	2000	

Mindkét modultípusban a monitorprogram jelenti a futási környezetet a többi program számára. A monitorprogramok a memória és a perifériák manipulálására szolgáló parancsokon és programindítási lehetőségeken felül logikai I/O kezelést tesznek lehetővé. Ez utóbbi segítségével a monitoron kívüli programok egységes módon kezelhetik a rendszert, előzőekben már ismertetett sokféle perifériáit.

A PROM programozást legtöbbször az almodulokban végzik. A folyamat a display-ról a DPROG programmal vezérelhető. A program az égetendő tok típusa szerint képes a memória törlésére, automatikus újrapróbálkozásra, memóriatartomány folyamatos égetésére stb.

Az ANAMAT visszafordítóprogram gépi kódú MAT 512 programok megfejtésénél használható.

A programozó egy programmodul elkészítésénél a rendszeren töltött idejének mintegy 95%-át két program használatával tölti el. Átlagosan kb. 60%-ot tartózkodik az almodul szövegszerkesztő-programjának (HIPO/M) hatáskörében a forrásfile előállításán és javításán dolgozva, és kb. 35%-ot a nyomkövető program (SMART TRACE) hatáskörében végzett programbelövással tölti. Ezért a fenti két program döntő hatással van az egész rendszer megítélésére.

2.1.1 Szövegszerkesztés

Az MPDS rendszerben a TXEDIT és a HIPO/M programokkal lehet forrásfile-okat, leírásokat készíteni. A főmodul editor programja csak kisebb javításokhoz, és a háttértárakkal való kommunikációhoz használható. Az almodulokban futó HIPO/M szövegszerkesztőprogram kb. kétezer sor méretig kényelmes körülményeket biztosít a szerkesztési feladatok megoldásához. Mivel a programok programmodulokból épülnek fel, és egy modul általában ezer sornál rövidebb, az említett kapacitás bőven elegendő.

A HIPO/M display-s editor egy lapon a szerkesztett szöveg 16 sorát jeleníti meg beállítható hosszúságú sorokban a képernyőn. Egy sor max. 64 karakter hosszú lehet. Egy lapra lapszámmal, vagy előrehátralapozással lehet hivatkozni, de külön kérhető az első és utolsó lap.

A képernyőn levő lapon megengedett a szabad mozgás és javítás, a kép egy sorral történő elmozdítása előre, ill. hátra, karaktorsorozat törlése, beszúrása és tabulátorok használata.

A nagyobb szövegrészek kezelésére a parancsábra parancsai szolgálnak. Azok végrehajtása kijelölt sortartományra és kijelölt karakteroszlopokra kérhető. A fenti parancsokkal szöveg kivitelre, beolvasásra, másolásra, törlésre és áthelyezésre adható utasítás. Leírások, dokumentációk készítésénél nyomdai szedés is elvégezhető.

A parancsábrából indítható a sokoldalú keresőparancs, mellyel a fent említett paraméterek beállítása után tetszőleges karaktersorozat kerestethető úgy, hogy „dont care” karakterpozíciók is megadhatók. A keresett minta helyére vagy elé új karaktersorozat szűrhető be. A HIPO/M parancsai az 1. mellékletben láthatók.

2.1.2 Nyomkövetés

A MAT 512-re írt programok az almodulokba, vagy közvetlenül a központba kerülnek át kipróbálásra. Az almodulok maguk is MAT 512 processzorok, így a programbelövésnél kihasználható az a tény, hogy a futtatás saját környezetben történik. Az almodulokban a belövést a SMART TRACE (MAT 512 Real-Time Trace) program támogatja.

A program két üzemmódban, egy teljes és egy részleges real-time üzemmódban működhet. A teljes üzemmódban a SMART TRACE interpretálja a követendő programot, s így a tényleges futásnál ugyan százszor lassabban futva, de a gépi utasítás szintjénél is mélyebb információt nyújtva segíti a belövést. A teljes üzemmód kialakítására azért volt szükség, mert a processzor rendelkezik olyan regiszterekkel és flagekkel, melyek állapotjelzők egy gépi utasítás végrehajtása után, de nem olvashatók. A teljes trace segítségével írásvédelem is megvalósítható és PROM területen futó programok is nyomon követhetők.

A real-time módú nyomkövetéskor két megállás között teljes sebességgel fut a program, amely sok esetben követelmény ahhoz, hogy kiértékelhető eredmény keletkezzen. Ilyenkor azonban a követett program leállítását célzó töréspontok elhelyezésekor eleget kell tenni a processzor hardware sajátosságai-ból eredő néhány megszorításnak.

Mindkét üzemmódban max. 4 db töréspont definiálható. A töréspontoknál paraméterként megadható egy memória pillanatfelvétel (snapshot) sorszáma és az, hogy hanyadik áthaladáskor álljon meg a nyomkövetés (ciklusok vizsgálata). A törésponti címen szereplő utasítás végrehajtása előtt a nyomkövető program közli a processzor állapotinformációit és a kért memóriaterület pillanatnyi tartalmát. A teljes nyomkövetés történhet lépésenkénti végrehajtással is, és tetszőleges utasítás végrehajtásakor leállítható. A nyomkövetéskor kapott képernyőtartalom a rendszerhez közös perifériaként kapcsolódó nyomtaton kinyomtatható.

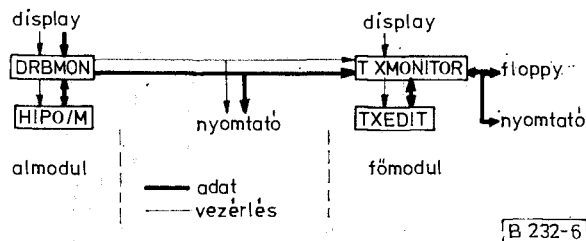
2.2. Az MPDS modulok használata

Az egyes programmodulok kapcsolatát a két leggyakoribb tevékenységen keresztül mutatjuk be részletesen.

2.2.1 Forrásfile készítés és javítás

Az előforduló szövegfile-okat, melyek lehetnek programok, leírások, esetleg object file-ok, ma már kizárólag az almodulokban javítjuk. Az almodulban folyó tevékenység ideje alatt a főmodul és a közös busz más programozók rendelkezésére áll (6. ábra).

Ilyenkor a főmodul, mint a háttértárak kezelője működik és csak a file-ok átvitelének idejére foglaljuk le.



6. ábra. Forrásfile-ok előállítása

2.2.2 Programok fordítása

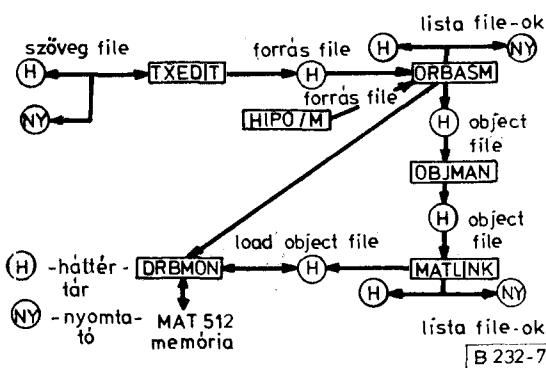
Programfordításkor, legyen az szintaktikus elemzés, vagy object generálás, az információs kapcsolat megegyezik az előző esettel azzal a különbséggel, hogy a TXEDIT helyén a DRASMS/XREF/SYMTAB modulok állnak. A fordítandó file-okat leggyakrabban az almodulok HIPO/M editorában helyezik el úgy, hogy a fordítóprogram és segédprogramjai közvetlenül az almodulból olvassák az input file-t. Ha a fordító hibát észlel, az a főmodul hibalistáján megjelenik, s annak alapján az azonnal javítható az almodulban. Így a háttértárra csak szintaktikailag hibátlan programok kerülnek. A fordításkor a főmodulban lista és object file generálható.

A fordítóprogram kisegítőprogramjai a forrásprogram szimbólumainak definíciós értékét, illetve a szimbólumok sorszáma szerinti előfordulási helyeit adják meg táblázatos formában.

A programmodulok object file-jait egy közös fiúba vonják össze az object manager (OBJMAN) programmal, majd a linkelőprogrammal (MATLINK) állítanak elő a MAT 512 memóriájába betölthető kódot. A linkelő program feloldja az egyes modulok egymásra történő hivatkozásait, megadja a hivatkozások számszerű értékeit, a memóriefoglalást, a feloldozott modulok azonosítóit és betölthető kódot generál.

3. Az MPDS modulok file-szintű együttműködése

A 7. ábrán feltüntettük a rendszer legfontosabb moduljait és a köztük folyó tipikus file-kapcsolatokat. Az egyes modulok között általában háttértáron keresztül valósul meg a kapcsolat. A rendszerben megvalósított logikai periféria-kezelés miatt a csatoló szerepét bármelyik erre alkalmas fizikai eszköz betöltheti. Mivel valamennyi file szövegfile, nyomtaton való megjelenítésük sem okoz gondot.



7. ábra. Az MPDS modulok file-szintű együttműködése

Összefoglalás

Az MPDS rendszer tárgyalása során nem térünk ki azokra a segédeszközökre, melyek nem kapcsolódnak közvetlenül a MAT 512 alapú fejlesztéshez, és csak utaltunk azokra a lehetőségekre, melyek más típusú processzor fejlesztése esetén az MPDS használatát indokolhatják.

Az MPDS rendszeres használata 1981-ben kezdődött el. Az eddigi eredmények biztatóak, a rendszer kiépítése fokozatosan nőtt. Az irodalomban megadunk néhány, az SPC távbeszélőközpont gyártásban alkalmazott software fejlesztői környezetről szóló cikket. Saját lehetőségeink ezekkel való összevetése, a tanulságok levonása azonban meghaladja egy cikk kereteit.

1. melléklet

A HIPO/M szövegszerkesztő program parancsai

A parancsábrában a kívánt parancs kiválasztása a ↑, ↓, space billentyűk valamelyikével történik. A cursor által kijelölt parancs tetszőleges grafikus karakterrel indítható, kivéve a START parancsot, mely csak Y-ra indul. (N) – logikai egység száma.

PARANCSS	PARAMÉTER	JELENTÉS
PRINT	n	0–255 a kiíratandó lap sorszám
START	–	memória teszt, alaphelyzet
BACK	–	visszatérés a monitorhoz
INPUT	elé	az adott sor elé beolvas (7)
OUTPUT	tói, ig	tartományt kivisz (8)
FIELD	n, m	$n \leq m < 64$ – keresési mező
FIND	tói, ig	keresés
COMPOSE	tói, ig	szedés a FIELD mezőben
COPY	tói, ig, elé	tartomány másolás

REMOVE	tói, ig	tartomány törlés
MOVE	tói, ig, elé	tartomány áthelyezés
LIMIT	n	az utolsó írható pozíció 0–63
TABU'S	8 db	állítható tabulátorok

A FIND-parancs

TÍPUS	FORMÁTUM	JELENTÉS
0.	/FI/	csak számol
1.	/F1/P	kiírja a lapot, villog FI
2.	/F1//F2/	számol, cserél
3.	/F1//F2/P	cserél, kiír
4.	/F1//F2/VP	kiír, Y-ra cserél, kiír
5.	/F1//F2/I	FI elé beszúrja F2-t
6.	/F1//F2/PI	FI elé beszúrja F2-t, kiír
7.	/FI/ /F2/VPI	kiír, Y-ra beszúr, kiír

A „/” jel az FI, F2-ben nem szereplő grafikus karaktert jelenti.

1. melléklet

- FI – keresendő karaktersorozat; DEL (kiírva?) esetén az adott karakterpozíción nem vizsgál egyezést
- F2 – beszúrandó karaktersorozat
- P – kiírás kérés
- V – beszúrás előtt kérdezzen

Ha a FIND-parancs kiírt egy lapot, akkor

- SPACE – tovább
- ESC – szabad módosítás a kiírt lapon
- Y – V esetén módosítás kérés

Sorparancsok

- DCA – tabulátor
- CR – sor eleje
- CTRL X – cursor jobbra
- CTRL H – cursor balra
- DEL – sor vége törlés
- NC – tab. mező törlés
- SCO – kar. beszúrás
- ETX – kar. törlés

Lapparancsok

- CTRL Y – cursor fel
- CTRL Z – cursor le
- CTRL L – home
- RES – sor, oszlop = 15,0
- LF – soremelés

Paraméterparancsok

- CTRL A – tói
- CTRL S – ig
- CTRL W – elé

A fenti CTRL karakterek némelyikének speciális billentyűk felelnek meg az almodulok tv display-in.

Lapmozgató parancsok

ALT	— sor törlés
NULL	— sor beszúrás
ULB	— fél lap beszúrás
ERASE	— * sor beszúrás
CTRL N	— vonalzó sor kérés
CTRL I	— sor ismétlés
SWS	— előző lap
SWC	— következő lap
CTRL T	— első lap
CTRL B	— utolsó lap
ESC	— parancsábra

I R O D A L O M

[1] *J. A. Brakel*: Structural aspects of the TGP 36 multiprocessor control of telephony switching system ISS 79. Paris p. 970—977.

[2] *E. B. Daly*: The management of large software development for stored program switching systems ISS 79. Paris p. 1287—1291.

[3] *M. Fulton*: Organization and operation of a software support service for an SPC telephone network ISS 79. Paris p. 1383—1389.

[4] *H. Y. Chang*: Application of software engineering to ESS. ISS 79. Paris p. 1390—1395.

[5] *T. Morisawa*: DEX A—113 TS, TLS program design and evaluations. Review of the ECL Vol. 27. No. 5—6. p. 469—481.

[6] *C. Carelli*: Il sistema PROTEO Telecomunicazioni 1979, No. 71—72. p. 2—68.

[7] *Dr. Darabos Z.*: MPDS MAT 512 telefonközpont-vezérlő programfejlesztő rendszere. HTE előadás, 1981 október 1.