

# A szolgáltatások szerepe a BHG kiskapacitású EPM alközpontjainak programfejlesztésében

Dr. Darabos Zoltán  
BHG Fejlesztési Intézet

## ÖSSZEFOGLALÁS

A cikk a szolgáltatások szerepével foglalkozik a BHG kiskapacitású mikroprocesszoros vezérlésű alközpontjainak programfejlesztési folyamatában. Ismerteti a központok programozható szolgáltatásainak különböző szintjét, a fejlesztési környezetet és a programozónak a programrendszer felépítése során nyújtott szolgáltatásokat.

## 1. Bevezetés

Az EP típusú elektronikus, tárolt program vezérlésű alközpontok /EP128, EP512/ gyártása mintegy 10 éves múltra tekint vissza a BHG Híradástechnikai Vállalatnál. A központokban a MAT 512/2 célvezérlőt alkalmazták, mely ára miatt nem tette lehetővé a kis kapacitástartomány lefedését, ezért az EP család gazdaságos alkalmazása a 100 mellékállomás fölötti tartomány volt sokáig. 1986-ban befejeződött a mikroprocesszoros vezérlésű kiskapacitású családtagok hardver és szoftver fejlesztése és 1987-ben megindult a központok sorozatgyártása. A kiskapacitású EPM központok családtagjait az alábbi táblázat tartalmazza, a tipikus kiépítések megadásával fővonal/mellékállomás jelöléssel.

név	EP8M	EP16M!	EP32M	EP64M	EP128M!
kiépítés	2/8	4/16	6/30	12/64	10/100
memória	24K	24K	62K	62K	62K

A táblázatban ! jellel jelölt típusok nem kerültek Magyar Postal approbációra, így azok hazai forgalmazása nem folyik (1. ábra). A család EP32M és attól felfelé lévő tagjai 80 %-ban az EP család kártyaválasztékából állnak, így ezek gyártásbavezetőse nagyobb nehézségek nélkül megoldott. A legkisebb központok /EP8M, EP16M/ viszont csak néhány kártyából épülnek fel és hordozható méretűek, így azok üzemeltetése és szervize szintén egyszerű.

## 2. A szolgáltatások két oldala

Ha egy programrendszer fejlesztési folyamatában szolgáltatásokról beszélünk, meg kell különböztetni a kifejlesztett berendezés szolgáltatásain és a fejlesztéshez nyújtott szolgáltatásokon belül az alábbi részterületeket.

### 2.1. A fejlesztőrendszer szolgáltatásai

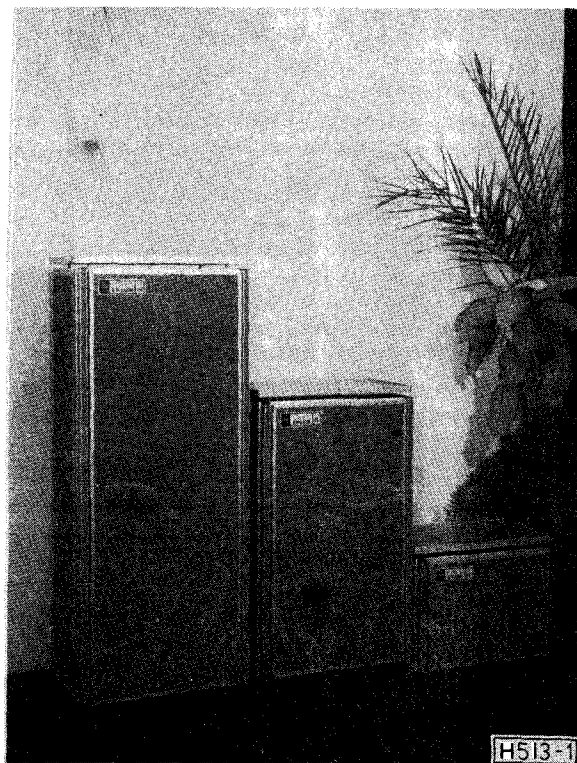
Beérkezett: 1989. 1. 30. (\*)



Dr. Darabos Zoltán

1977-ben végezte a BME Villamosmérnöki karát, majd a BHG ösztöndíjasaként 2 évig nappali szakmérnöki tanulmányokat folytatott. 1980-ban egyetemi doktori fokozatot szerzett az Integrált távközlés területén. A

BHG-ban az EP központok programfejlesztői környezetének kialakítása után az EPM mikroprocesszoros alközpontcsalád programfejlesztését vezette. Jelenleg a BHG Fejlesztési Intézet EP fejlesztésén működő vezetőfejlesztési osztály vezetője.



1. ábra. A kiskapacitású mikroprocesszoros EPM család 3 legnagyobb tagja

- hardverrel biztosított szolgáltatások, melyek
  - a fejlesztendő berendezéstől függetlenül rendelkezésre álló mérő és vizsgálóeszközök, jel és jelzőgenerátorok, emulátorok stb,
  - a fejlesztendő berendezés labormodellje esetleges speciális átalakításokkal és a
  - környezetszimulátorok
- szoftver feltételek /elsősorban a programozónak nyújtott szolgáltatások/
  - szoftver fejlesztő programok /szerkesztők, fordítók, debuggerek/ általában nem a fejlesztés alatt álló berendezésen futnak.

— a fejlesztés alatti programrendszerbe beépített monitor programok, berendezésvizsgáló programok és a fejlesztés alatti programrendszer struktúrája, mint a programozó közvetlen környezete.

### *2.1.1. A fejlesztőrendszer hardver felépítése*

#### *2.1.1.1. A központ vezérlője*

Az EPM központokban alkalmazott vezérlő 8085 típusú mikroprocesszort tartalmaz, mely a táblázatban szereplő maximális memóriakapacitást tudja kezelni. A vezérlőn 3 csatornás időzítő egység szolgáltatja a szükséges óramegszakításokat, periféria illesztő áramkör kezeli a régebbi EP központok kártyái által igényelt perifériaidőzítés szekvenciáját. A vezérlő 20mA áramhurkos duplex csatlakozással rendelkezik, melyre szerviz eszközök /EPROM programozható, monitor/ csatlakoztatható. A vezérlő tranzlens hibák elleni védelméről NMI megszakításra kapcsolódó újraindítható időzítés gondoskodik, a RAM memóriák pedig tápfeszültség kimaradása esetén a védett területeket megőrzik. A fejlesztés idejére a programfelügyeletet végző időzítő /watchdog/ áramkört kikapcsoljuk, hogy a tetszőleges helyen és időpontban megállítható és újraindítható legyen a program az emulációs mérések során.

#### *2.1.1.2. Jel és jelzés generátorok*

A fejlesztés során felhasználhattuk a szokásos telefontechnikai jelgenerátorokat és teszttereket, de legnagyobb segítséget a már kifejlesztett, az EP családba tartozó berendezések jelentették. Mivel ezek nagyobb kapacitásúak, mint a család ezen kisebb tagjai, nem okozott az új berendezést teljes kapacitásáig terhelni a már működő berendezésekből keltett forgalommal.

#### *2.1.1.3. Környezetszimulátorok*

A fejlesztés során nem került sor környezetszimulátor berendezés használatára, mely a reális forgalmi és szolgáltatás igénybevételi körülményeket utánozó programozott berendezést igényelt volna. Ennek fejlesztési ráfordításait nem tudtuk biztosítani, így vállaltuk a nehezebb utat, a kísérleti forgalmi periódust, melyet szakértő környezetben, a Magyar Posta Budapest Vidéki Igazgatóságának szervízosztályán, Budaörsön végeztünk.

#### *2.1.1.4. A programfejlesztés és tervezés eszközei*

A programrendszer tervezésének beindítása 1985-re nyúlik vissza. A programrendszer kidolgozásával egyidejűleg folyt a mikroprocesszoros technika hardver feltételeinek kialakítása is. A programfejlesztést alapvetően 8085-ös In-circuit

emulátor alkalmazására építettük. Az emulátor parancsnyelve és kialakítása biztosította a vezérlő real-time emulációját oly módon, hogy közben eredeti helyén a központ berendezésében működött. Így a hardverközeli programok belövésében nem kellett szoftver szimulációt alkalmazni. Az emuláció biztosította ugyan az utasítás szintű programfejlesztést több megszakításos real-time környezetben, de nem nyújtott támogatást a magasabb szintű programfejlesztési feladatok megoldásához. Ezért bizonyos fokig indokolt a programfejlesztés eszközeihez sorolni a program olyan strukturálását, mely lehetőleg optimális módon tudja a rendelkezésre álló programozói erőforrásokat a feladat megoldására koncentrálni, és a lehető legkevesebb energiát igényli a fejlesztői támogató programok elkészítéséhez, s ugyanakkor elegendő segítséget ad a gyártható minőségű szoftver hibáinak időben történő felderítéséhez. Ebben a nehezen tervezhető, intuíciót igénylő folyamatban támaszkodhattunk a távbeszélőközpontok MAT 512 bázisú programfejlesztésében elért BHG eredményekre, a már több ezer példányban eladott programcsoportokra és nem utolsósorban a MAT 512 programfejlesztés támogató rendszerének felhasználói tapasztalataira.

### *2.1.2. A fejlesztőrendszer szoftver felépítése*

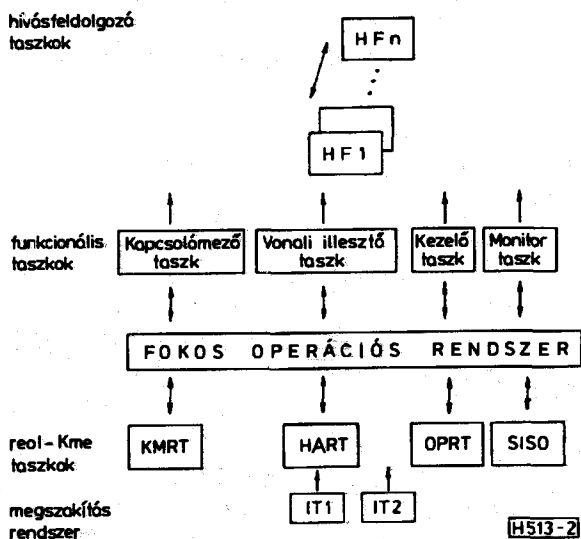
#### *2.1.2.1. Alkalmazott nyelvek*

A vezérlőprogramok írásakor mindig felmerülő kérdés az alacsony vagy magasszintű nyelvek alkalmazása. Ezt a hardver, sebességi, dokumentációs, programozói és követési kérdések halmaza befolyásolja. Esetünkben elsősorban hardver és sebességi kérdések döntöttek. A programrendszer így assembly nyelven íródott, biztosítva ezzel a vezérlő lehetőségének maximális kihasználását, de a programstruktúrát úgy alakítottuk ki, hogy a legfelső programszinten, a hívásfeldolgozó szinten egy olyan makronyelvet alkalmazzunk, mely áttekinthetővé teszi a szolgáltatások megfogalmazását, azok esetleges későbbi módosítását. A nyelvet úgy alakítottuk ki, hogy annak feldolgozására ne kelljen külön fordítóprogramot írni a fejlesztői programcsomagba és ugyanakkor a központi programrendszerében se legyen szükség interpreter programra a makronyelven írt programrészek futtatásához. A nyelvi fordítók és szerkesztők BHG gyártmányú, az In-circuit emulátor is tartalmazó VFR berendezésen futnak.

#### *2.1.2.2. Az EPM programrendszer struktúrája*

Az EPM központokban működő programrendszer multiprocesszoros működésre készült, multitasking felépítésű. A legtöbb programmodul valamennyi központban azonos, a szükséges módosításokat a szolgáltatások kiterjedése ill. a vezérlendő hardverben meglévő különbségek indokol-

ják. A rendszer taszkjait az alábbi ábra mutatja (2. ábra).



2. ábra. A programrendszer felépítése

A rendszert összetevő taszkok alapvetően két részre tagolhatók. A funkcionális taszkok statikusak, a rendszer indításakor keletkeznek és nem szűnnek meg a normális működés során. A hívásfeldolgozást végző taszkok dinamikusak, csak addig élnek a rendszerben, míg a hívás fennáll. A két fajtát együtt normál taszkoknak nevezzük, hogy megkülönböztethessük az általunk igénybe vehető operációs rendszer szolgáltatásokat a real-time taszktól.

A funkcionális taszkok követik a hardver tagolódását, külső interfészek olyanok, hogy a feldolgozó szoftver hardverfüggetlenségét biztosítani tudják.

A hívásfeldolgozó taszkok szabványos üzenetekben cserélnék információt a funkcionális taszkokkal, ill. egymással olyan esetekben, mikor a másik hívás adataira szükségük lehet. Jó példa erre a figyeltetés szolgáltatás, ahol annak engedélyezése előtt meg kell győződni a figyeltetni szándékozó hívó magasabb jogáról a másik kapcsolatban résztvevő valamennyi partnerrel szemben.

A kezelőt, nagyobb feldolgozás igénye miatt külön funkcionális taszk tartja kézben, mely üzeneteket a vonali illesztő szűrése után kapja meg. A monitor taszk valamennyi taszkkal kapcsolatot teremthet az üzenetküldő paranccsal segítségével, normális működése során azonban csak az operációs rendszer belső forgalmát figyeli.

Valamennyi funkcionális taszkhoz a szigorú /10ms nagyságrendű, időzítéseket ellátó real-time szintű program tartozik. Ezek önállóan végzik a jelszűrést, események üzenetek útján való jelzését a funkcionális taszkoknak és ellátják a vezérelt hardver időkritikus működtetési feladatait. Az operációs rendszerhez tartozó időzítő real-time taszk /OSRT1-2/ végzi a funkcionális megsza-

kítási rendszer ütemezését és a normál taszkok behívását.

### 2.1.2.3. Az operációs rendszer szolgáltatásai

- SEND – Üzenet küldés adott számú normál taszknak. Szabványos üzenet összeállítás után hívható meg.
- RECFROM – Üzenet vétel felfüggesztéssel vagy anélkül, adott számú és kulcsú normál taszktól.
- RECANY – Üzenet vétel felfüggesztéssel vagy anélkül bármely feladótól.
- LOOKFROM – Üzenet vétel felfüggesztés nélkül adott számú és kulcsú normál taszktól.
- LOOKANY – Üzenet vétel felfüggesztés nélkül bármely feladótól.
- READY – Normál taszk futásának felfüggesztése azonnali újraütemezéssel.
- YESIT – Real-time taszk futásának letiltása normál és real-time taszk közötti kommunikációban a kölcsönös kizárás biztosítására.
- NOIT – Real-time taszk futásának engedélyezése normál és real-time taszk közötti kommunikációban a kölcsönös kizárás biztosítására.
- SETPRI – Normál taszk prioritásának beállítása
- GETFEL – Relatív időmérés
- CREATE – Normál taszk létrehozása
- ENDTASK – Normál taszk megszüntetése

### 2.1.2.4. A monitor taszk szolgáltatásai

A fejlesztés során nélkülözhetetlennek bizonyult a rendszer magasszintű működését nyomonkövető MONITOR program. Ennek üzenetkövetési szolgáltatását használva a telefonos események real-time jellege és időbeli követhetlensége megszűnik, rögzíthetővé és elemezhetővé válik. Az üzenetküldési szolgáltatással az egyes funkcionális taszkok működése vizsgálható.

#### DUMP parancs

A memória tartalmának kinyomtatása. Az adott címtől kezdődően a memória 64 byte-ját egy időszelvényben lekérdezi, majd utána megjeleníti a képernyőn. A kinyomtatás után lehetőség van az adott terület módosítására: a megnyitott 64 byte méretű ablakban a cursor billentyűvel lehet mozogni és bármelyik byte felülírató. A módosított byte azonnal felülíródik a memóriában is. Mivel a memóriaterületben helyezkedik el a perifériacímterület is, a paranccsal perifériacímterületeket is írhatunk.

#### INCREMENT parancs

Az előző DUMP parancsban megadott cím értékét 64 byte-tal megnöveli, és indítja a DUMP parancsot.

#### DECREMENT parancs

Az előző DUMP parancsban megadott cím értékét 64 byte-tal csökkentli és indítja a DUMP parancsot. Egy DUMP parancs után kiadott INCREMENT és DECREMENT parancsokkal lapozni lehet a memóriában.

#### SAMPLE parancs

Dinamikus DUMP. A DUMP parancs után kiadott S parancs a 64 byte-os ablak ciklikus lekérdezését és megjelenítését végzi. A megjelenítési sebesség kb. 20 kép/s.

#### CLEAR parancs

A DUMP parancsban kijelölt 64 byte-os terület kinullázása a memóriában és az eredmény klírása. A parancs után automatikusan DUMP parancs indul.

#### LOAD parancs

Memóriaterület módosítása. Az adott címtől kezdődően max 32 byte-ot egy időszelvényben a memóriába ír. Így lehet olyan kódokat is javítani, melyeken a javítás alatt is fut a program.

#### MESSAGE parancs

M parancsokkal előre összeállított üzeneteket küldhetünk el tetszőleges címettel a rendszerben tetszőleges taszk, mint feladó nevében. Az üzenet sikeres elküldését a monitor hangjelzéssel nyugtázza. Az üzenet többször is elküldhető az M parancs ismételtetésével. Ha az üzenetet a MONITOR taszk nevében küldjük el, melynek száma 1, a válasz is a MONITOR-hoz fog jönni ha a feladott üzenet válaszköteles volt. A monitorról feladott üzeneteket is megjeleníthetjük a P parancsokkal.

#### PRINT parancs

Rendszerüzenetek klírása. Az operációs rendszer a taszkok által átvett üzeneteket, valamint a taszkok rendszerhívásainak leldőzítési eseményt egy 16 elemű listán gyűjti. A P parancs ezt a listát üríti ki a megjelenítés során úgy, hogy ha valaha azt találja, hogy a lista tele van, akkor egy üres sor beiktatásával jelzi, hogy túlsordulás történt, tehát ott lehet olyan üzenet a rendszerben, melyet nem tudott megjeleníteni. A megfigyelt üzenetfolyamatot szűrési feltételek, feladó, címett és név szerint szelektálhatjuk, így a parancs üzem közben is alkalmas egy adott áramkör viselkedésének megfigyelésére.

## 2.2. A fejlesztő berendezés szolgáltatásai

A fejlesztendő berendezéssel kapcsolatos elvárásokat részben vagy egészében szoftver úton kell realizálni. Ez utóbbin belül is megkülönböztetjük az alábbiakat:

- fix, mintegy hardver jellegű megoldások, melyek egyrészt a
- gyártáskor rögzített szolgáltatások, másrészt
- üzembehelyezéskor, ill. a kiképzett szerviz személyzet által elvégezhető szolgáltatás módosítások, továbbá

- a berendezés folyamatos működése során a felhasználó által bármikor elvégezhető programozási szolgáltatások, úgy mint a
  - speciális készülékhez kötött programozási és a
  - mindenki rendelkezésére álló programozási szolgáltatások.

### 2.2.1. A fejlesztett berendezés beépített programozási szolgáltatásai

#### 2.2.1.1. Gyártási előprogramozás

A központok programrendszere gazdagon paraméterezett. Gyakorlatilag valamennyi hardverfüggetlen adat definiált struktúrában a felhasználó rendelkezésére áll az ún. helyszíntől független adatterületen. Ez az adatterület EPROM-ba égetett, bár egyes részlet csak alap értelmezésnek számítanak, mert a működés során végrehajtott mellékállomással és speciális készülékről végzett programozások módosíthatják őket. A EPM32M központtól felfelé a memória, tápfeszültség kimaradás esetén is megőrzi a védett adatokat. A központ hardverének kezelése a szolgáltatások adott paraméter környezetben való végrehajtása, tehát az algoritmus a rögzített szolgáltatások körébe tartozik, módosítására csak a gyártónak van joga és lehetősége.

Az EPM központok hazai forgalmazása típus helyszíntől függetlenül történik, a gyárban előprogramozva. Ezzel végzik el a központok vizsgálatát is. A berendezés felhasználói igényekhez való illesztése az üzembehelyezéskor és az üzemeltetés során folyamatosan a helyszíntől független adatok módosításával történik.

#### 2.2.1.2. Szolgáltatásmódosítások a helyszínen

A helyszíni szolgáltatásmódosítások EPROM égetéssel végezhetőek. A központ dokumentációjában részletes leírás szolgál a módosítások végrehajtásának és a közben szükséges ellenőrzéseknek az elvégzésére. Ezek az alábbiakra terjednek ki.

##### Hardver konfiguráció

A központ által kezelt mellékek és fővonalak száma és jellege. A kapcsolódó fővonalak irányok és áramköreinek száma

##### Számozási rendszer

A mellékek hívószámainak, a fővonalak egyéni hívószámainak a kezelőnek és szolgáltatások kódszámainak teljes köre.

##### Kategóriaosztályok

Maximum 15 kategóriaosztály szerkeszthető össze a helyszíni igények szerint.

##### Mellék kategóriák

A mellékek átsorolható új kategóriaosztályba.

##### Számkorlátozás

A fővonalakon kiadott hívószámsorozatok és azok hossza függetlenül rugalmasan korlátozható, belőlük 4 felhasználós telefonközpont

alakítható ki önállóan használt fővonalakkal, önálló korlátozásokkal és közös kezelővel igény esetén.

Forró fővonalak és mellékek, kezdeti átirányítások

PBX mellékállomási csoportok

A szolgáltatásokkal kapcsolatos valamennyi időzítés

A melléknek és fővonalnak adott jelzések időzítése

**2.2.2. Felhasználói programozás szolgáltatások**

**2.2.2.1. Programozási szolgáltatás mellékállomásai készülékről**

Az EPM központokban három különféle átirányítási szolgáltatás és készülékhez rendelt rövidített hívószám tár áll rendelkezésre. Az átirányítási szolgáltatásként lehet kérni a rendszertől az alábbiakat.

Feltétel nélkül

A hívott mellék figyelmeztető hangot kap, de az érkező hívás azonnal a beprogramozott célkészülékre megy.

Mellék nem válaszol esetén működő

A hívott mellék rövidített idejű csengetést kap. Ha a hívott nem jelentkezik, a hívás a beprogramozott célkészülékre megy.

Mellék foglalt vagy nem válaszol esetén működő

Megegyezik a fenti esettel, de rövidebb idejű csengetés után, ha a hívott mellék nem válaszol, a hívás a célállomásra megy.

Az átirányításokból láncok képezhetők. A láncokban különböző típusú átirányítással szerepelhetnek készülékek és a láncok hosszát helyszíntől függő adatokban lehet meghatározni.

A mellékállomások rövidített hívószám tárral rendelkezhetnek /általában 2 vagy 4 darab/ melyek a központ vezérlő memóriájában vannak elhelyezve. Mind az átirányítások, mind a rövidített hívószám tárral speciális szám beadása után módosíthatók. A rövidített hívószámok programozása során programozható a 2. és 3. tárcsahangra való várakozás helye illetve mód van a rövidített kladott szám folytatására.

**2.2.2.2. Programozás speciális készülékről**

Az EPM központokban az üzem közbeni programozáshoz az OP-SET kezelői készüléket, vagy kijelölt melléket vagy mellékeket használhatunk, abban az esetben, ha azokat a programozási joggal ellátták. A kezelői készülék használatának előnye, hogy kijelzéseivel jobb visszajelzést biztosítanak, mint a normál telefon akusztikus jelzéssel, melyről ezért csak a programozási szolgáltatások korlátozott köre vehető igénybe. A speciális készülék ill. jog használatát az indokolja, hogy az innen végrehajtott programozás nemcsak a progra-

mozó készülékre vonatkozik, mint a fenti esetekben, hanem más készülékeket is érint, valamint az, hogy a távhívaskorlátozások feloldása például az üzemeltetőnek pénzébe is kerülhet.

**2.2.2.2.1. Programozói jogú mellék szolgáltatásai**

Új kategóriák beállítása

Három darab előprogramozott kategória közül adhat egy újat a programozó készülék bárkinek a központban azzal a feltétellel, hogy a kijelölt kategóriában lévő távhívás jogosság nem lehet nagyobb a sajátjánál. A módosított kategória természetesen visszaállítható az eredetire is.

Forró fővonalak és mellékek

A központ bármely fővonala vagy melléke forróvá tehető valamelyik másik mellékre. Az egyik felprogramozott fővonalon bejövő hívás, vagy a forró mellék híváskezdeményezése azonnal a programozáshoz kijelölt melléket csengeti. Az eredeti állapot visszaállítható.

Átirányítások

A központ tetszőleges melléke a fenti három átirányítási fajta közül tetszőleges másra átirányítható, és átirányított állapota meg is szüntethető függetlenül attól, hogy azt őnmaga vagy más hozta létre.

**2.2.2.2.2. Programozói jogú kezelői készülék IOP-SET/ szolgáltatásai**

A fentiekén kívül az OP-SET további lehetőségekkel rendelkezik.

Erőszakos bontás

Veszélyhelyzetben szükség lehet arra, hogy valamennyi fővonal foglaltsága esetén a kezelő szabad kimenő vonalhoz jusson. A fővonallal hívás erőszakos bontásával az ott folyó beszélgetés megszakad és azt a kezelő lefoglalhatja.

Kizárás

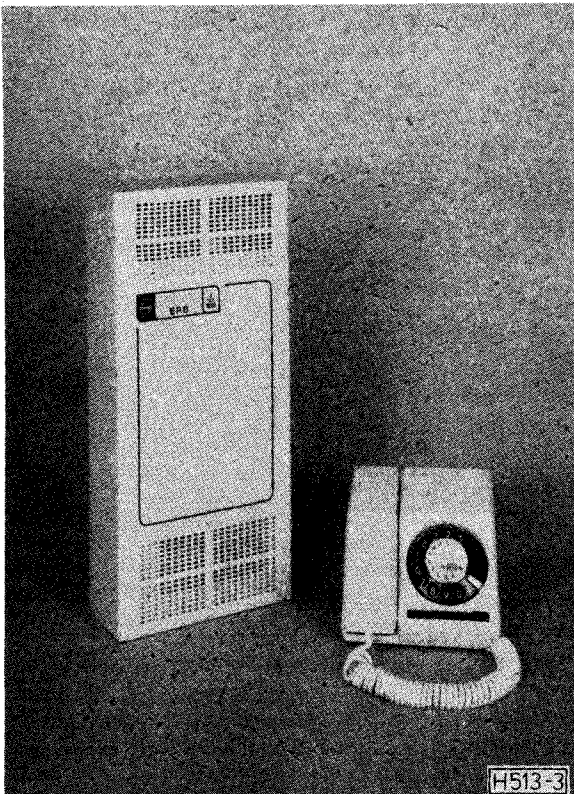
Meghibásodás vagy egyéb okból bármely fővonal vagy mellék kizárható a forgalomból. Ilyenkor a kizárt vonal semmilyen szolgáltatást nem kap. A kizárt áramkör újra felszabadítható.

Dátum beállítás

A kezelő órájának ill. a központ naptárának beállítása.

Vizsgálóprogramok hívása

A központ három vizsgálóprogramjának beindítása a kezelői funkciók időleges felfüggesztése mellett. A programok futása idejére a hívásokat a kijelölt mellék kezeli. A kapcsolómező működésvizsgálatát, az illesztőáramkörök vizsgálatát és a MONITOR taszk szolgáltatásait biztosító három program valamelyikének futtatásakor a hívásokat kijelölt mellék kezeli.



3. ábra. Az EP8M a legkisebb mikroprocesszoros BHG alközpont

### 3. A szoftver megoszlása

A fejlesztés során létrejövő eljárások, módszerek hagyományos berendezéseknél is jelentős szellemi erőfeszítések eredményeként születtek, ma azonban, mivel nagyrészt programozható eszközök állnak már rendelkezésünkre ezen eredmények algoritmusaí gépi úton is rögzíthető szoftverek. A teljes fejlesztési ráfordításnak tehát csak egy részét képezi a berendezésbe épített szoftver. A gyártónak mind nagyobb érdeke fűződik ahhoz, hogy a beépített szoftver racionális kialakításával a két rész arányát az eladhatóság felé tolja el, azaz a melléktermékekből minél többet használjon fel, integrálja azokat vagy egy független vizsgáló-fejlesztő rendszerbe, vagy pedig magába a berendezésbe. Mi magunk az EPM központok fejlesztésénél ez utóbbira törekedtünk.

### 4. Továbbfejlesztési lépések

Az EPM központokon keresztül bevezettük a mikroprocesszoros vezérlést az EP család kiskapacitású berendezéselbe. Ezen folyamatot továbbfejlesztve a közeljövőben valamennyi EP berendezés mikroprocesszoros vezérlésének lehetősége is megteremtődik. Emellett folyik az ismertetett szoftver rendszer további szolgáltatásbővítése. Ezek közül ki kell emelni a intelligens trafikációs és az integrált főnök-titkári szolgáltatást, melyek számlázó számítógépek és új, speciális billentyűs készülék csatlakoztatását teszik lehetővé.