

Digitális jelfeldolgozó- és mérőrendszer

ÖSSZEFOGLALÁS

A cikk ismerteti egy TMS 32010 mikroprocesszor alapú rendszer felépítését és alkalmazási lehetőségeit, valamint a rendszer tesztelésére kifejlesztett analóg hangolású digitális szűrő elvét. A cikk annak a dolgozatnak az alapján készült, amely 1987-ben a BME villamoskari TDK konferenciáján II. díjban részesült.

Bevezetés

A '80-as évek elején piacra került nagybonyolultságú (VLSI) áramkörök a digitális jelfeldolgozás fejlődésére is hatással voltak; a jelfeldolgozó mikroprocesszorok megjelenése adta meg a szükséges technológiai hátteret a spektrumanalízis, a digitális szűrés és általában az analóg jelek számítógéppel segített feldolgozásának újabb, gyakorlati fejlődéséhez.

Egy ilyen jelfeldolgozó mikroprocesszor a TEXAS cég 1982-ben megjelent TMS 32010 (a továbbiakban: TMS) áramköre is, mely max. 20 MHz-es órajelével, speciális utasításkészletével és a hagyományostól eltérő belső felépítésével (mely az ún. módosított Harvard struktúrát követi) lehetővé teszi digitális jelfeldolgozási feladatok hatékony végrehajtását.

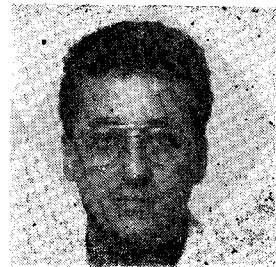
A Harvard architektúra lényege, hogy az adat- és a programfeldolgozó egységek különválasztottak. Ez azt jelenti, hogy külön sínek köré építve találjuk meg a program-memóriát az utasításkódoló-egységgel illetve az aritmetikai-logikai egységet az adatmemóriával. Ez a strukturális különválasztás lehetővé teszi, hogy a processzor egyidejűleg hajtsa végre egy utasítás program-memóriából való lehívását és dekódolását, valamint egy előzőleg lehívott utasítás végrehajtását.

A TMS 320 processzor-család esetében ezt a struktúrát úgy módosították, hogy a két egység között egy adatutatót létesítettek. Ez a módosítás lehetővé teszi, hogy a felhasználó a program-memóriában adatokat tároljon, oda írjon, illetve onnan olvasson. A processzorhoz tartozó esetleges perifériák és az adat-memória illetve a program-memória közötti adatforgalmat gépi utasítások szabályozzák.

A TMS speciális architektúrája, utasításkészlete, feladatorientáltsága miatt önmagában nem támogatja egy intelligens, felhasználóközelí fejlesztőrendszer kialakítását. A TMS kifejezetten jelfeldolgozó feladatok végrehajtására készült. Belső felépítése és utasításkészlete az e területen leggyakrabban előforduló utasítások (szorzás, lép-



FODOR GÁBOR



SOMOGYI GÁBOR

tetés, adat-tömbök kezelése stb.) gyors végrehajtását szolgálják. A TMS maximálisan 8 be- és kimeneti port csatlakoztatását teszi lehetővé egy szintű megszakítási rendszerrel, ami több periféria gyors kiszolgálását nem támogatja. Ezen kívül a TMS nem rendelkezik olyan kivezetésekkel, amelyek segítségével a program-végrehajtás menete közvetlenül szabályozható lenne (pl. a hagyományos mikroprocesszoroknál megszokott HOLD vagy WAIT jelekkel sem). Egy cél-orientált áramkörben nincs is szükség ilyenfajta kivezetésekre, ám ha fejlesztőrendszerrel van szó, melyen szükség van egy új program belövését segítő MONITOR-funkciók megvalósítására (pl. lépésenkénti végrehajtás, töréspontnál való megállítás), akkor e kivezetések hiánya kifejezetten hátrány.

A TMS memória-kapacitása (4 kszó) sem teszi lehetővé, hogy az olyan tágirányú programok, mint általában a MONITOR programok és assemblerek, közvetlenül a TMS-n fussanak.

Mindezek indokolják, hogy az ilyen többretű feladatok ellátását, a felhasználóval való közvetlen kapcsolattartás feladatát célszerű egy általános célú mikroprocesszorra bízni: az így létrejövő rendszert joggal nevezhetjük jelfeldolgozó fejlesztő rendszernek.

Az általunk bemutatott fejlesztőrendszerben a host-funkciókat egy i8085 mikroprocesszor alapú számítógép (továbbiakban P85) biztosítja. Ez alkalmas TMS gépi kódú felhasználói programok fejlesztésére, memóriába töltésére, tárolására és egyéb, felhasználót segítő funkciók megvalósítására (pl. regiszterek tartalmának megjelenítése, töréspontok definiálása, lépésenkénti program-végrehajtás vezérlése stb.).

A TMS-P85 rendszer HW felépítése

A fejlesztő rendszer funkcionálisan három nagyobb egységből épül fel, melyek együttesen teszik lehe-

tővé a fent említett funkciók megvalósítását és általában bizonyos jelfeldolgozó feladatok real-time (valós idejű) végrehajtását.

A három egység vázlatos felépítését, funkcionális kapcsolatát mutatja az 1. ábra.

Az első egység a P85 alapú számítógép, melynek memória-kapacitása maximális kiépítettségben 64 kB. Ezen memória egy része ROM, ebben helyezkedik el a P85-öt közvetlenül vezérlő ún. monitor-program, melynek főbb szolgáltatásai a következők:

- 8080/85 assembler
- editor és file-kezelő
- töréspont és lépésenközi végrehajtás
- DUMP, GO és REGISTER parancsok.

Ugyancsak ROM-ban helyeztük el a TMS-t vezérlő monitorprogramot is, melynek szolgáltatásai az előző monitورهoz hasonlóak, természetesen azzal a különbséggel, hogy ez egy TMS cross-assemblert tartalmaz.

A P85-höz csatlakoztatott RAM egy része a P85 saját használatú területe, a másik rész közös a TMS-sel, azaz a két processzor egy szorosan csatolt rendszert alkot. A közös memória-területen keresztül történik a két processzor közti adatforgalom; a P85 ide helyezi be a cross-assembler által generált TMS kódot, és a TMS kimeneti adatai szintén ide kerülnek.

A második egység a TMS alapú mikrogép, melynek memóriája (maximális kiépítettségben 4 kszó) teljes egészében a P85-tel közös RAM. Ez a modul tartalmaz egy $4k \times 1$ bites RAM-ot is, (BREAK memória), amelyet a TMS futása közben állandóan címvez. Így lehetőség van a TMS tetszőleges címen való megállítására, ugyanis a P85 a TMS elindítása előtt ebbe a RAM-ba tetszőleges címre „1”-et helyezhet, ami ezen bit címzése esetén megszakítást generál a TMS felé.

A fejlesztői rendszer kialakításánál gondot jelentett, hogy — mint említettük — a TMS nem rendelkezik a hagyományos processzoroknál meg-

szokott HOLD „lábbal”, valamint az, hogy írás esetén nem különbözteti meg azt, hogy az írás a memóriába történik, vagy pedig adatkivitel történik a perifériára.

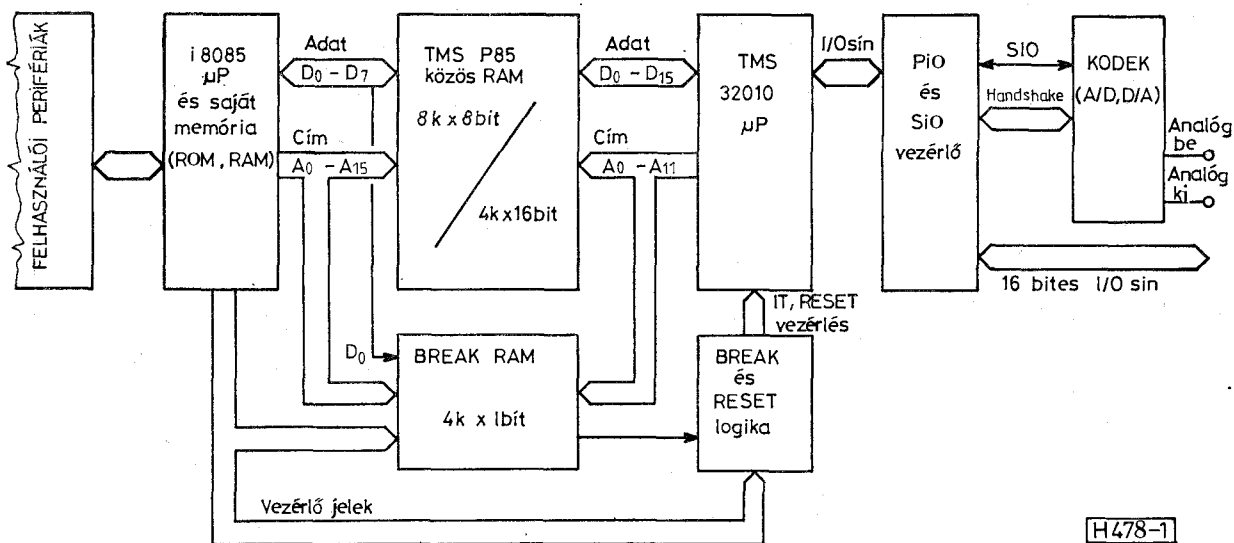
A TMS elindítását és megállítását ezért egy ún. be- és kirakódó rutin végzi, melyet a P85 helyez el a TMS memóriájában. A kirakódó feladata, hogy a TMS megállítása esetén annak belső állapotát a P85-tel közös memóriába mentse, a processzort „reset” állapotba juttassa, ennek megtörténtét pedig jelezze a P85-nek. A berakódó feladata ennek éppen ellentéte; az, hogy a TMS-t „reset” állapotából elindítsa, a közös memóriából betöltse a megfelelő belső állapotot és a TMS-t a felhasználói program végrehajtására irányítsa.

A rendszer harmadik nagyobb eleme egy KODEK kártya, melynek kódolási/dekódolási szabálya a logaritmikus kvantálás szerinti ún. „A”-törvény. Ez azt jelenti, hogy ez a kártya az analóg/digitális átalakítás két elvileg szükséges lépése — kvantálás és kódolás — közül az elsőt logaritmikus karakterisztika szerint végzi el. Erre a karakterisztikára jutunk, ha a kvantálás során fellépő relatív hibát kívánjuk minimalizálni, vagyis logaritmikus kvantálási törvény esetén valószínűleg meg a szintfüggetlenül állandó jel/kvantálási zaj viszony. A gyakorlatban általában a logaritmikus karakterisztikát törtvonalas karakterisztikával közelítik. Erre javasolja a CCITT a táblázatos formában definiált „A” törvényt.

A KODEK modul teszi lehetővé, hogy analóg jelek real-time feldolgozása megtörténjen. A TMS és a KODEK közötti adatforgalom a TMS-hez kialakított soros I/O porton keresztül, handshake jelekkel időzítve történik.

A rendszer eme harmadik egységének TMS-hez való illesztését látja el az ún. KERNAL programcsomag. Ez a programcsomag alapvetően három feladatot old meg.

1. A processzor belső állapotának elmentése/betöltése. (Erre a programok fejlesztése során van szükség.)



1. ábra. TMS 32010 alapú fejlesztő rendszer felépítése

2. A KODEK-ről történő olvasás, illetve KODEK-re történő írás esetén a handshake jelek figyelmével az adatforgalom szabályozása. (Erre a KODEK és a TMS eltérő működési sebessége miatt van szükség.)

3. A KODEK kódolási/dekódolási szabályából adódó linearizálás és logaritmálás. A KODEK által szolgáltatott adatokat a TMS-sel való számítások elvégzése előtt linearizálni kell és a megjelenítés céljából a KODEK felé elküldött adatokat is „elő kell készíteni”, azaz logaritmálni kell.

E programcsomag egyes rutinjai bármilyen felhasználói programból közvetlenül hívhatók, így a rendszer felhasználója tulajdonképpen egy analóg I/O csatornával rendelkező TMS programfejlesztői környezetet lát.

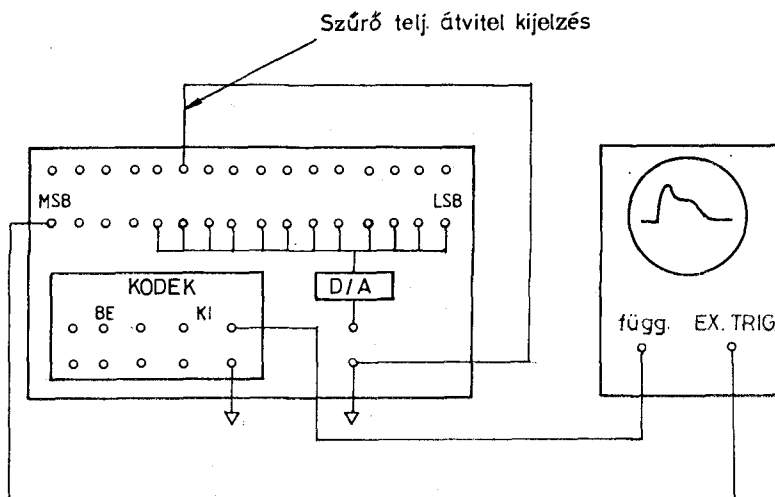
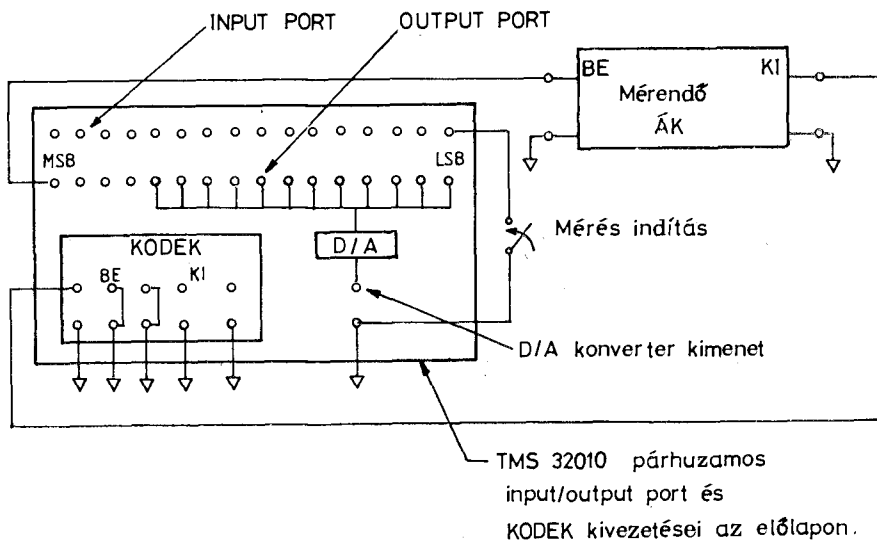
A továbbiakban e fejlesztőrendszer egy konkrét felhasználását mutatjuk be, mely egy általunk kifejlesztett áramkört analizál.

Áramköri méréseket megvalósító SW

A fejlesztőrendszer szolgáltatásait egy mérési feladaton kívántuk bemutatni, melyben a TMS-P85 rendszer mint mintavételező-kiértékelő-adatfeldolgozó egység működik. A mérést vezérlő SW (áramkör-analízis programcsomag) a TMS-en fut, külső parancsokat a TMS párhuzamos input portjáról fogad el.

A SW a következő mérési lehetőségeket nyújtja:

1. trigger jel kiadása, majd mintavételezés (8 kHz órajellel)
2. numerikusan végzett deriválás vagy integrálás a mintákon, mely funkciók segítségével lineáris hálózatok esetén az egységugrás gerjesztésre adott válaszból előállítható az impulzusválasz, illetve a lineáris gerjesztésre adott válasz.
3. a mintasorozaton (annak deriváltján vagy integráltján) FFT (Fast Fourier Transform) végzése, a teljesítményspektrum számítása.



H478-2

2. ábra. Mérési elrendezések egységugrás-válasz felvételéhez és a felvett adatok megjelenítéséhez

A teljesítmény-spektrum a diszkrét Fourier-transzformáltból közvetlenül számítható, ugyanakkor szemléletes képet ad arról, hogy a mintavételezett jel teljesítménye hogyan oszlik meg a különböző frekvenciájú összetevők között. Ezt a mennyiséget igen gyakran használják jelek jellemzésére, például a beszédfelismerésben, rezgés-analízisben is.

A kijelzés a KODEK kártyán keresztül oszcilloszkópra történik, a tárolt adatsorozat periodikusan ismételt kiírásával.

A mérőrendszer alkalmazása

Az ismertetett konfiguráció és SW alkalmas egyes áramkörök analizisére is. Minthogy a KODEK egység átviteli frekvenciatartománya 300—3400 Hz (a mintavételi frekvencia 8 kHz), így természetesen az analizálandó áramkör frekvenciasávja sem lehet ennél szélesebb. Ennek figyelembevételével olyan szűrő áramkört terveztünk, mely a mérőrendszerrel jól mérhető, ugyanakkor változtatható karakterisztikájú és olcsón megvalósítható, így alkalmas hallgatói mérési célokra és (esetleg hibridizálva) különféle hangfrekvenciás szűrési feladatokra is.

Hallgatói mérési elrendezést mutat be a 2. ábra. A mérendő áramkör lehet például a később ismertetésre kerülő digitális szűrő is. Ebben az esetben az áramköranalízis programcsomag minden funkciója tanulmányozható, ugyanakkor megfigyelhető az is, hogy a szűrő egyes paramétereit (pl. szorzótényezők) változtatva hogyan változnak a szűrő átvitelét jellemző mennyiségek (pl. egységugrásra adott válasz teljesítmény-spektruma).

Lényeges szempont, hogy a szűrő-karakterisztika a felhasználó számára egyszerű módon (potenciométerek és banándugók elrendezésével) tág határok között változtatható. Az alábbiakban

azt is látni fogjuk, hogy egy olyan olcsó áramkör-ről van szó, mely moduláris felépítésű, így esetleges meghibásodás esetén az alkatrészcsere sem okoz problémát. (A kezelői hibákból származó meghibásodások javításának fontosságát a szerzők saját tapasztalatukból is jól ismerik.)

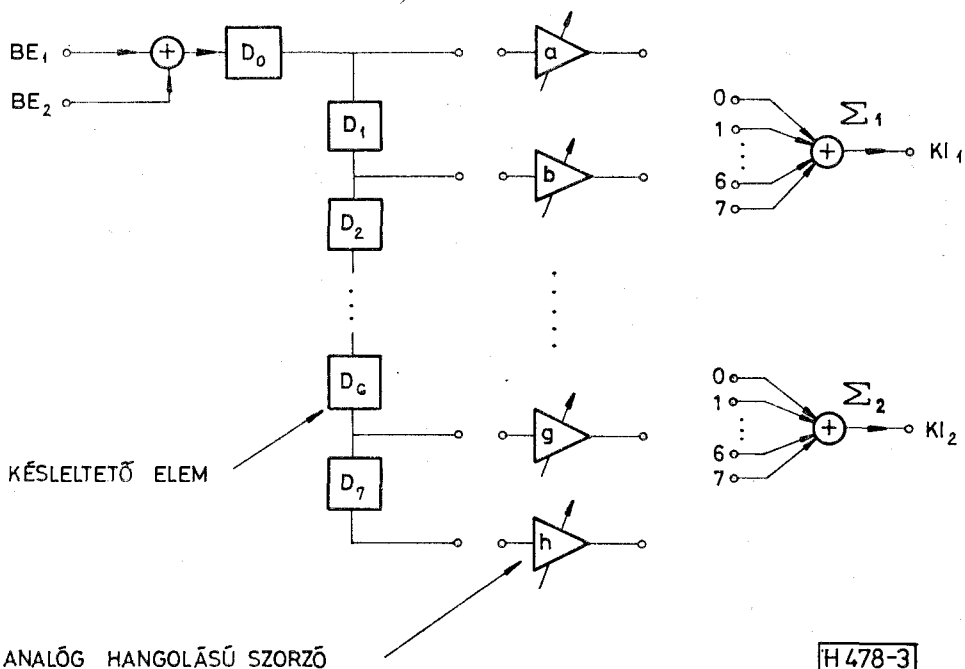
Az effajta kísérletezések nemcsak azért tanulságosak, mert kézzelfogható közelségbe hozzák a hallgatók számára az egyik legalapvetőbb digitális áramkört, hanem mert egyúttal egy digitális mérőrendszer használatát is megismertetik.

Analóg hangolású digitális szűrő

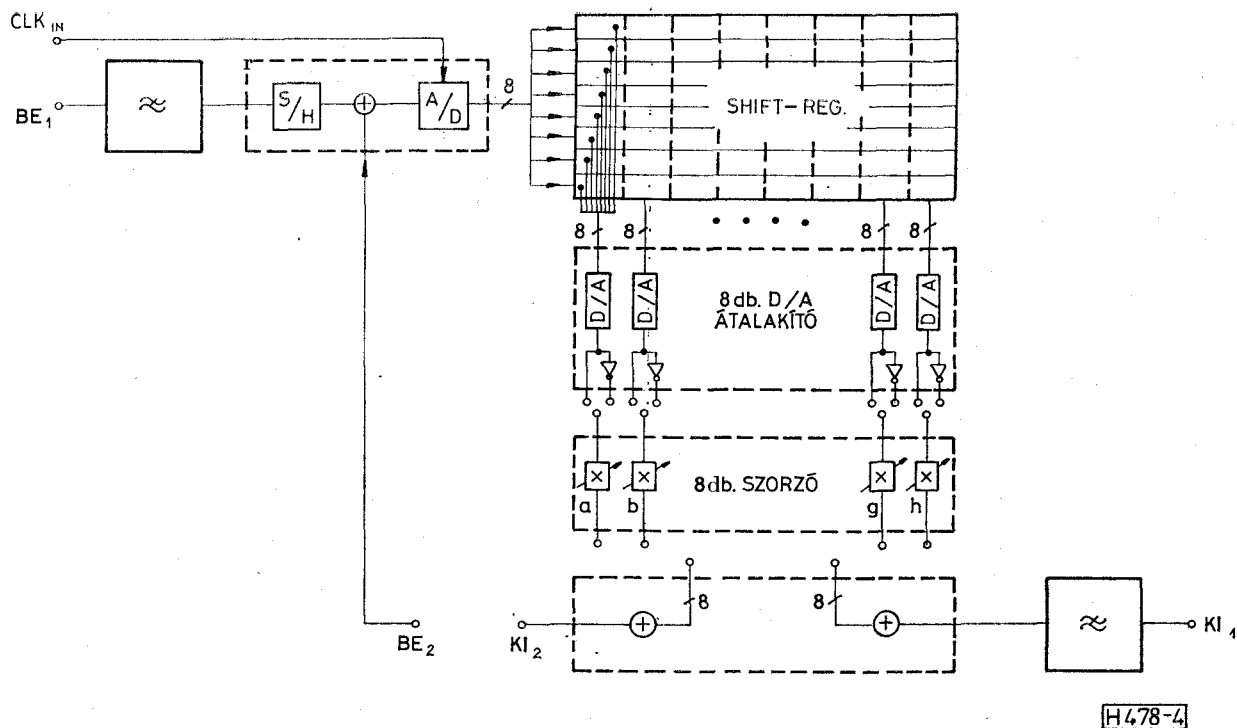
A digitális szűrők alapáramkörei a digitális szorzó, összeadó és a késleltetéseket megvalósító shift-regiszter áramkörök. Gyors működés eléréséhez vagy több szorzó együttes működtetése, vagy egy igen gyors szorzómű használata és az adatforgalom bonyolult megszervezése szükséges. Mindkét megoldás igen költséges. Ezért vetődött fel a gondolata egy olyan digitális szűrő megépítésének, ahol a késleltetések kivételével az adatok feldolgozása tisztán analóg módon történik. Ez viszont felveti a következő problémát:

— minden digitális késleltető-elem kimenetén gondoskodni kell a digitális adatok analóggá konvertálásáról. Amennyiben ez, a tisztán digitális megoldással összevetve, olcsóbban, de kielégítő pontossággal megoldható, akkor egy „vegyes” felépítésű szűrő áramkör gazdaságosan megépíthető. A konstans értékekkel történő szorzás analóg jelek esetében egyetlen ellenállással megvalósítható, így összehasonlíthatatlanul olcsóbb, mint egy digitális szorzómű használata.

Célunk volt univerzális, azaz tág határok között változtatható karakterisztikájú és variábilis strukturájú szűrő felépítése. Ennek érdekében a következő megfontolásokat tettük:



3. ábra. Analóg hangolású digitális szűrő elvi felépítése



4. ábra. Analóg hangolású digitális szűrő áramköri blokkjai

1. Transzverzális, direkt strukturájú szűrőkkel (FIR és IIR) minden elvileg lehetséges diszkrét idejű szűrő karakterisztika realizálható. Ennek alapján úgy döntöttünk, hogy a késleltetőelemeket sorba kötjük (transzverzális struktúra).
2. Minthogy a szűrő viselkedését alapvetően befolyásolja a szűrő rekurzív vagy nem rekurzív volta, a visszacsatolási pontok helye stb., ezért a fentiek a felhasználó által változtathatóak kell legyenek.
3. A szűrő kapcsolódása a „külvilághoz” minden ponton analóg jelekkel történjék és a digitális belső működés maradjon „rejtve” a felhasználó előtt.
4. A szűrővel legyen realizálható minden olyan transzverzális szűrő, amelyet a rendelkezésre álló áramkörkészlet megenged, a szűrő-együtthatók legyenek folyamatosan változtathatók.

Mindezek alapján a 3. ábrán látható struktúra mellett döntöttünk, amelyet a 4. ábrán látható áramköri blokkok valósítanak meg. Ez utóbbi ábráról leolvasható, hogy a szűrőn áthaladó jelek csupán a shift-regiszter tömbben digitálisak, a szűrő minden további része analóg.

Úgy véljük, hogy a fenti elvek alapján megvalósított szűrőáramkörök — éppen a realizáció olcsósága miatt — gazdaságosan alkalmazhatók olyan nagy darabszámban iparilag előállított, — pl. közszükségleti célú — berendezésekben is, ame-

lyekben a hangfrekvenciás tartományban szükséges az üzem közbeni áthangolhatóság.

Köszönetnyilvánítás

A szerzők köszönetüket fejezik ki konzulensüknek, dr. Jagudits László adjunktusnak a munka során nyújtott értékes segítségért.

Fodor Gábor
Somogyi Gábor
BME Villamosmérnöki Kar

IRODALOM

- [1] Dr. Simonyi Ernő: Digitális szűrők. Műszaki Könyvkiadó, Budapest, 1984.
- [2] Ronald N. Bracewell: The Fourier Transform and its Applications, McGraw-Hill Book Company 1978.
- [3] TMS 32010 Assembly Language Programmer's Guide TEXAS Instruments Inc. 1984.
- [4] TMS 32010 User's Guide. TEXAS Instruments Inc. 1983.
- [5] TMS 32010 Data Sheet. TEXAS Instruments Inc. 1983.
- [6] Pasztirák Gábor: Fejlesztőrendszer tervezése TMS 32010 mikroprocesszorhoz. (Diplomaterv) BME-HEI, 1985.
- [7] Tóth János: TMS 32010 fejlesztőrendszer integrálása. (Diplomaterv) BME-HEI, 1986.
- [8] Körösi Sándor: Assembler TMS 32010 mikroprocesszorhoz (Diplomaterv) BME-HEI, 1985.
- [9] Rajkai Gábor: Hazai gyors Fourier analízátor. Magyar Elektronika II. évf. 2. szám 16—24. old.