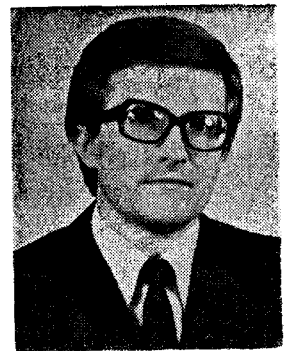


# Periodogram alapú spektrumbecslés korszerű digitális jelfeldolgozó mikroprocesszor felhasználásával

DR. KOCSIS FERENC  
Távközlési Kutató Intézet

MARX FERENC—  
DR. SOLYMOSI JÁNOS  
BME Híradástechnikai Elektronika Intézet



## ÖSSZEFOGLALÁS

A cikk korszerű digitális jelfeldolgozó processzorra kifejlesztett periodogram alapú spektrumbeeslő program tervezésének kérdéseit tárgyalja, ül. az eredményeket mutatja be. Kifejlesztésre kerültek olyan gyors DFT-modulok ( $N=64, 128$  és  $256$ ), amelyek sebessége meghaladja az ismerteket. Felépítésüket a TMS32010 DSP architektúráis jellemzőinek kihasználásával optimaltuk. A fennmaradt tárterület lehetővé teszi, hogy ugyanazon processzoron egy, a DFT-t, ill. spektrumbecslést felhasználó főprogram futhasson.

Meghatároztuk a DFT számításánál a véges pontos-ságú aritmetika miatt fellépő hibák hatásait is.

A programok alkalmazási lehetősége széles körű: beszédfeldolgozás, rezgések vizsgálata, rőszsávú kódolók, beszédttikcsítás, geofizikai vizsgálatok stb.

## 1. Bevezetés

A híradástechnikában, de a különböző mérnöki tudományokban is gyakran merülnek fel spektrumbecslési feladatok. A diszkrét minták alapján történő spektrumbecslésre két alapvető módszer ismert. Egyik a rendelkezésre álló diszkrét minták alapján becsli a jel autokorrelációs függvényét, s a keresett spektrumbecslés azután a korrelációbecslés Fourier-transzformáltjaként áll elő [2]. A másik módszer a jelminták alapján spektrumbecslésre az ún. periodogramot használja fel, amely a kiindulási jel mintái Fourier-transzformáltjainak számításával állítható elő. A periodogram nem konzisztens becslés [7], azonban simítás és átlagolás alkalmazásával a becslés javítható [14]. Mindkét eljárás a Fourier-transzformáció alkalmazására épül. Véges mintasorozatok esetén a Fourier-transzformált megadható a diszkrét Fourier-transzformált (DTF) értékekkel is.

A cikk jelfeldolgozó mikroprocesszoron (TMS 32010 DSP) programozott úton megvalósított spektrumbeeslő programrendszert mutat be. Elsőként a periodogram alapú becslés kérdéseiről lesz szó, amit a TMS32010 mikroprocesszor néhány jellemzőjének bemutatása követ. A negyedik rész tárgya a becsléshez szükséges diszkrét Fourier-transzformációt számító igen gyors DFT-modulok kiválasztásának szempontjai és felépítése. A kidolgozott DFT-modulok pontszáma  $N=64, 128$  és  $256$ . Részletesen elemeztük a DFT meghatározása közben a véges pontosságú aritmetika

A dolgozat részben a BME Villamosmérnöki Kar 1986. év i Tudományos Diákköri Pályázatára benyújtott, s I. díjat nyert dolgozat alapján készült [4].

## DR. KOCSIS FERENC

1975-ben szerzett villamosmérnöki diplomát a BME Villamosmérnöki Karán, majd a Távközlési Kutató Intézetben (TKI) kezdett dolgozni. Egyetemi doktori értekezését 1976-ban védte meg. 1983—86 között a BME—HEI-ben dol-

gozott tudományos ösztöndíjasként, ahol digitális jelfeldolgozás és jelszintézis algoritmusok kérdéseivel foglalkozott. Jelenleg ismét a TKI dolgozója. Szakmai érdeklődési köre: rendszertechnika, digitális jelfeldolgozás, számítástechnika, algoritmusok elmélete.

használata miatt keletkező hibákat. Végül a spektrumbecslést a Welch-módszer alapján, Bartlett-ablakkal számító program ismertetése kerül sorra (a mintahossz  $N=2048$ , a maximális kötött idejű jelfeldolgozási frekvencia  $f_{max}=20,1$  kHz).

## 2. Periodogram alapú spektrumbecslés

A hírközlésben sokszor előforduló alapfeladatok egyike valamely adatsorozat spektrális jellemzőinek meghatározása. A spektrumbecslés egyik alapmódszere az ún. periodogram alkalmazása ([7], [14]).  $N$  hosszúságú mintasorozat esetén a  $P_N(\omega)$  periodogram definíciója

$$P_N(\omega) = \frac{1}{N} |X(e^{j\omega})|^2 \quad (2-1)$$

ahol  $X(e^{j\omega})$  az  $\{x(n) | 0 \leq n \leq N-1\}$  bemeneti adatsorozat Fourier-transzformáltja:

$$X(e^{j\omega}) = \sum_{n=0}^{N-1} x(n)e^{-j\omega n} \quad (2-2)$$

Az  $S_x(\omega)$  spektrumbecslésként első közelítésben választható a  $P_N(\omega)$  periodogram, azonban ez bebizonyíthatóan annak eltolt és nem konzisztens becslése [7]. A becslés jellemzői átlagolással és ablakolással javíthatók. A szórás csökkentésére szolgáló átlagolásnál a bemeneti  $\{x(n)\}$  adatsorozatot  $K$  db  $U$  hosszúságú adatszégmensre vágjuk ( $N=KU$ ), azaz képezzük az

$$x^{(i)}(n) = x(n+iU-U) \quad 0 \leq n \leq U-1, 1 \leq i \leq K \quad (2-3)$$

részsorozatokat, amelyekből azután a Bartlett-féle [1] átlagolt periodogram képezhető

$$P_U^{(i)}(\omega) = \frac{1}{U} \left| \sum_{n=0}^{U-1} x^{(i)}(n)e^{-j\omega n} \right|^2 \quad 1 \leq i \leq K \quad (2-4)$$

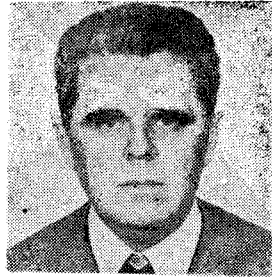
A BME Villamosmérnöki Kar Híradástechnika szakának V. éves hallgatója. 1984 óta a BME-HEI-ben digitális jelfeldolgozási feladatok megoldásával foglalkozik. Szakmai érdeklődési köre: mikroprocesszorok hardware és software problémái, digi-



tális jelfeldolgozás, hálózatelmélet.

DR. SOLYMOSI JÁNOS

1936-ban született és 1961-ben szerzett villamosmérnöki oklevelet a Budapesti Műszaki Egyetemen. Egyetemi doktori címet 1967-ben, kandidátusi fokozatot, 1978-ban kapott. Egyetemi tanulmányai elvégzése óta a BME-n dolgozik. Jelenleg a Híradástechnikai Elektrotechnika Intézet docense és



hálózatelmélettel, ill. jelfeldolgozással foglalkozik.

s az átlagolással módosított  $S_x(\omega)$  spektrumbecslés:

$$S_x(\omega) = \frac{1}{K} \sum_{i=1}^K P_{U^{(i)}}(\omega). \quad (2-5)$$

Az átlagolás eredményeképp ugyan csökken a becslés szórása, de ennek ára az eltolás növekedése és a felbontás csökkenése (a periodogramok rövidebb adatsorozatokról állnak elő). A becslés simításának másik lehetősége időtartománybeli ablakolás (súlyozás) alkalmazása a periodogramok számításakor [14]. Eszerint a (2-4) alatti módosított periodogramok új alakja:

$$P_U^{(i)}(\omega) = \frac{1}{U \cdot A} \left| \sum_{n=0}^{U-1} x^{(i)}(n)w(n)e^{-j\omega n} \right|^2 \quad (2-6)$$

ahol

$A = \frac{1}{U} \sum_{n=0}^{U-1} w^2(n)$ , és a  $\{w(n)\}$  az időtartománybeli ablakozó sorozat. A Bartlett-féle átlagolás tekintetbe  $\{w(n) = 1 \mid 0 \leq n \leq U-1\}$  sorozattal való ablakolásnak is.

A spektrumbecslés a (2-6) szerinti módosított periodogramok felhasználásával ismét a (2-5) összefüggéssel számítható [7]. Ablakolásra nagyszámú ablakolófüggvény ismert. A pontosság szempontjából spektrumbecslésre sok esetben a legkedvezőbb az ún. Hamming-ablak, amelynek definíciója.

$$w(n) = 0,54 + 0,46 \cos\left(\frac{\pi \cdot n}{N-1}\right)$$

$$-(N-1) \leq n \leq (N-1). \quad (2-7)$$

Alkalmazásához állandók tárolására vagy ismételt kiszámítására van szükség. Számítástechnikai szempontból kedvezőbb a háromszög- vagy Bartlett-ablak [1]

$$w(n) = 1 - \frac{|n|}{N} \quad -(N-1) \leq n \leq (N-1) \quad (2-8)$$

A periodogram alapú spektrumbecslés kiszámítása történhet a DFT felhasználásával, ugyanis a DFT éppen az  $X(e^{j\omega})$  Fourier-transzformáltat adja meg a frekvenciatengelyen egymástól egyenlő távolságban lévő  $\omega_k = \left(\frac{2\pi}{N}\right)k$  pontokban ( $0 \leq k \leq N-1$ ).

Ily módon a keresett spektrumbecslést megadó új összefüggések:

$$S_x^w\left(\frac{2\pi}{U}k\right) = \frac{1}{K} \sum_{i=1}^K P_U^{(i)}\left(\frac{2\pi}{U}k\right) \quad 0 \leq k \leq U-1$$

$$P_U^{(i)}\left(\frac{2\pi}{U}k\right) = \frac{1}{KU} \left| \sum_{n=0}^{U-1} x^{(i)}(n)w(n)e^{-j\left(\frac{2\pi}{U}\right)kn} \right|^2 \quad (2-9)$$

Azaz a spektrumbecslés számításának menete:

1. az  $\{x(n) \mid 0 \leq n \leq N-1\}$  hosszúságú bemeneti adatsorozat részekre osztása a (2-3) kifejezésnek megfelelően,

2. az egyes részsorozatokhoz tartozó

$$X_U^{(i)}(k) = \sum_{n=0}^{U-1} x^{(i)}(n)w(n)e^{-j\left(\frac{2\pi}{U}\right)kn} \quad (2-10)$$

$$0 \leq k \leq U-1$$

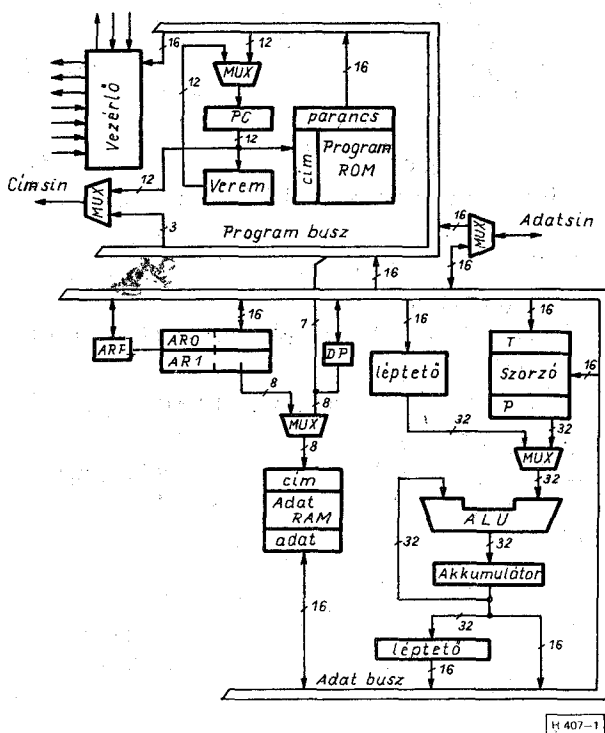
diszkrét Fourier-transzformáltak számítása az adott alkalmazásban célszerű  $\{w(n)\}$  ablakozófüggvény megválasztása után

3. a periodogramok és az  $\left\{S_x^w\left(\frac{2\pi}{U}k\right)\right\}$  spektrumbecslés számítása (2-9) alapján.

Amint látható, a számítás kulcseleme a 2. lépésben a diszkrét Fourier-transzformáltak meghatározása, amely a becslési eljárás számításigényének döntő részét adja. Kötött idejű (real-time) spektrumbecslés megvalósításához így elsősorban igen gyors DFT számítási eljárásokra van szükség. A becslést digitális jelfeldolgozó mikroprocesszorral (TMS 32010), programozott úton valósítottuk meg. A következő fejezet röviden bemutatja a TMS32010 néhány, számításaink szempontjából alapvető jellemzőjét.

3. A TMS32010 második generációs digitális jelfeldolgozó mikroprocesszor felépítése, fontosabb jellemzői

A TMS32010 második generációs DSP, amely 2,7  $\mu\text{m}$  esíkszólességű NMOS technológiával készül, s egyetlen +5V-os tápfeszültségről üzemeltethető. Fogyasztása 900 mW.



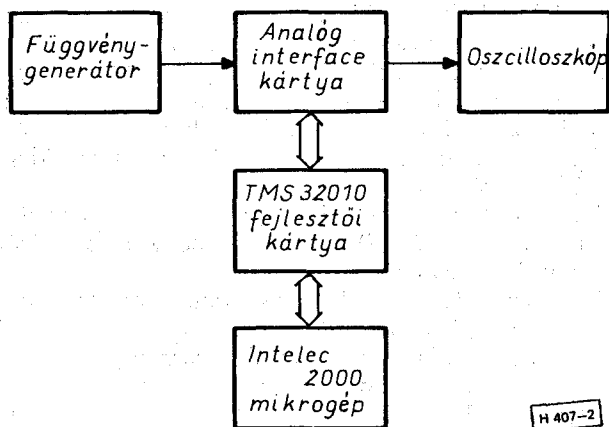
1. ábra. TMS32010 CPU felépítése

A processzor külön memóriával rendelkezik az adatok és az utasítások tárolására (1. ábra), s lehetőség van a két memória közti adatmozgatásokra (a programmemória adatokat is tartalmazhat, ami fordítva nem igaz). A gép 16-bites, de lehetőség van byte-ok elérésére és mozgatására is. Az utasítások max. 4Kszó kapacitású külső tárban helyezhetők el, míg a belső adat RAM mérete 144 szó.

Az utasításlelvétel és az utasításvégrehajtás egymással teljesen átlapoltak. Az utasítások többségének elvégzéséhez elegendő egyetlen órajelciklus (minimális értéke 200 ns). Az utasítás-készlet 60 elemű [10] kialakításánál messzemenően figyelembe vették a digitális jelfeldolgozási feladatok megoldása során felmerülő számítások műveletigényét: a gép felépítése és utasításai révén különösen alkalmas a  $\sum a(i)b(i)$  súlyozott véges összeg meghatározására (FIR-szűrés).

A fixpontos aritmetika négy fő eleme a 32-bites ALU, a 32-bites akkumulátor, a 16-bites léptető-regiszter és a  $16 \times 16$ -bites párhuzamos szorzó. Az ALU egyik operandusa betöltés közben 0–15 bittel idővesztés nélkül balra léptethető. Az akkumulátorban keletkező eredmény szavanként elmenthető, s a felső 16-bites szó eközben 1 vagy 4 bittel balra léptethető. Adatmozgatásokkal együtt egyetlen szorzáshoz 4 utasításra (800 ns), egyetlen összeadáshoz pedig 3 utasításra (600 ns) van szükség.

Az adatok háromféle címezési móddal érhetőek el (direkt, indirekt és közvetlen). Két óraciklus alatt nyolc különböző periféria címezhető és érhető el. Kiszolgáláskérésük két kivezetésen vizsgál-



2. ábra. A fejlesztői környezet blokkvázlata

ható: egyikén maszkolható megszakításkérés érkezik, a másikat feltételes ugróutasítás teszti.

A programkészítéshez fejlesztői környezetként egykártyás, a TMS32010 köré épülő mikrogép [11], és analog interface kártya [12] állt rendelkezésre. A mikrogép rendszerprogramjai: szövegszerkesztő, monitor és gépi kódra fordító program. A saját kazettás adattároló lassúsága, valamint a szövegszerkesztő hiányos szolgáltatásai miatt a forrásszöveg javítása, és tárolása egy 8085  $\mu P$  alapú lemezegységgel ellátott gépen történt, amely soros vonalon csatlakozik a fejlesztői kártyához. Az analog kártya tartalmazza az A/D és D/A átalakításhoz szükséges összes áramkört. A mintavételi frekvencia programozható. Az „A/D átalakítás vége” jel a TMS32010 megszakításkör bemenetére csatlakozott. A fejlesztői környezet vázlatát a 2. ábra szerinti.

#### 4. Gyors DFT-modulok

A gyors, kötött időben történő, periodogram alapú spektrumbecslés programozott megvalósításának alapja, hogy megfelelően gyors, a diszkrét Fourier-transzformáltat számító programok álljanak rendelkezésre. A DFT kiszámítására nagyszámú algoritmus ismert ([3], [5], [7], [9]). Közülük az optimális kiválasztását jelentősen befolyásolják a TMS32010 DSP lehetőségei. A megoldandó feladat szempontjából a TMS32010 legfontosabb architektúráis jellemzői:

- a gyorstár mérete mindössze 144 szó,
- a szorzások és egyéb aritmetikai-logikai műveletek (pl. összeadás) időigényének aránya 4 : 3,
- közvetlen címezést alkalmazva egy szorzási művelet végrehajtásához mindössze 400 ns-ra van szükség (MPYK-utasítás),
- a kiegészítő félvezető RAM mérete 4 Kszó (adatok és utasítások tárolása egyaránt lehetséges).

Következésképp a leggyorsabb DFT-modulok akkor adódnak, ha minimális a műveletigényük, ill. az adatok lehető legnagyobb része a műveletvégzés idejére a gyorstárban helyezhető el. Ugyanakkor a programoknak és az adatoknak (az átlapolt DFT számításához szükséges kettős pufferrel

együtt) el kell férniük a 4 Kszó méretű programtárban. Mivel mindig ugyanazon algoritmus változó adatokkal történő végrehajtásáról van szó, a programstruktúra (utasítások, állandók, címek) valójában előre rögzített, következőképp valamilyen módon előre optimalizálható. Az ismert legfontosabb algoritmusok:

a) A DFT közvetlen kiértékelése [3] négyzetes lépésszáma ( $O(N^2)$  szorzás és összeadás) miatt nem jöhet számításba.

b) A fokozatos részekre osztáson alapuló módszerek [3] számításigénye csupán  $O(N \log N)$  rendű. Az algoritmosztály nagy előnye, hogy szabályos felépítésű, fixpontos aritmetika esetén kedvezőek a skálázási és zajjellemzői. Ismert (pl. [3]), hogy a hasonló típusú eljárások közül a négy szerinti részekre osztást alkalmazó (radix—4) algoritmusok műveletigénye a legkedvezőbb (a DFT(4) modul csupán  $\pm 1$  és  $\pm j$  értékekkel történő triviális szorzásokat tartalmaz), s nincs szükség az eredmények átmeneti tárolására. A DFT(8) modulokra épülő algoritmusok számításigénye ugyan némileg kisebb, azonban a számítás során keletkező átmeneti eredmények tárolására viszonylag sok tárolórekeszre van szükség.

c) A Winograd-típusú DFT algoritmusok [15] műveletigénye  $O(2N)$  nagyságrendű. A szóba-jöhető kisméretű moduloknál DFT(4) esetére a műveletszám azonos a fokozatos részekre osztás DFT(4) moduljával, míg a DFT(16) modulnál a különbség elhanyagolható (komplex bemeneti adatsorozatnál 20 valós szorzás és 120 valós összeadás) a radix—4 DFT(16) eljárásához képest, azonban mintegy másfélszer nagyobb adattárterületre lenne szükség, és a szükséges adatrendezések is jelentős időt vesznek igénybe. A radix—4 algoritmus mellett szól a már említett szabályossága, ami megkönnyíti a program optimális kialakítását.

A fenti megfontolások alapján a választás a radix—4 DFT algoritmusra esett (az eljárás egyébként a számítási pontosság szempontjából is előnyösebb).

Példaképp röviden nézzük meg a 64-pontos transzformáltat számító algoritmust (ez a modul szolgál alapul a 128- és 256-pontos eljárásokhoz is). A transzformációs alapösszefüggések:

$$X(k) = \sum_{n=0}^{63} x(n) e^{-j \left( \frac{2\pi}{64} \right) kn} \quad 0 \leq n, k \leq 63 \quad (4-1)$$

A

$$k = 4^2 k_3 + 4k_2 + k_1 \quad 0 \leq k_1, k_2, k_3 \leq 3 \quad (4-2)$$

$$n = 4^2 n_3 + 4n_2 + n_1 \quad 0 \leq n_1, n_2, n_3 \leq 3 \quad (4-2)$$

indextranszformációt végrehajtva és azt a transzformációs összefüggésbe helyettesítve:

$$X(k_1, k_2, k_3) = \sum_{n_1=0}^3 e^{-j \left( \frac{2\pi}{4} \right) k_3 n_1} e^{-j \left( \frac{2\pi}{64} \right) (4k_2 + k_1) n_1} \times$$

$$\times \sum_{n_2=0}^3 e^{-j \left( \frac{2\pi}{4} \right) k_2 n_2} e^{-j \left( \frac{2\pi}{16} \right) n_2 k_1} \sum_{n_3=0}^3 x(n_1, n_2, n_3) \times e^{-j \left( \frac{2\pi}{4} \right) k_1 n_3} \quad (4-3)$$

Az utolsó összegzés az ún. DFT(4) pillangó:

$$\begin{aligned} \hat{X}(0) &= x(0, n_2, n_1) + x(1, n_2, n_1) + x(2, n_2, n_1) + \\ &+ x(3, n_2, n_1) \\ \hat{X}(1) &= x(0, n_2, n_1) - jx(1, n_2, n_1) - x(2, n_2, n_1) + \\ &+ jx(3, n_2, n_1) \\ \hat{X}(2) &= x(0, n_2, n_1) - x(1, n_2, n_1) + x(2, n_2, n_1) - \\ &- x(3, n_2, n_1) \\ \hat{X}(3) &= x(0, n_2, n_1) + jx(1, n_2, n_1) - x(2, n_2, n_1) - \\ &- jx(3, n_2, n_1) \end{aligned} \quad (4-4)$$

ahol

$\hat{X}(k_1)$  az összegzés  $k_1$ -től függő értéke.

Amint látható, a pillangó csak triviális szorzásokat tartalmaz. A teljes transzformáció ilyen pillangóknak és közbülső forgatásoknak a számítására bontható.

A korábban már említett MPYK szorzóutasításnál a forgatásoknál szükséges állandók 12-bites értékei közvetlenül az utasításban helyezhetők el, s nincs szükség külön memóriaterületre az együtt-tárolására. A program végrehajtása gyorsítható: ciklusokat nem tartalmazó, ún. kiterített kód (straight line code) alkalmazásával elmaradnak a ciklusszervezéshez használt döntéshozatali utasítások. Hátrány, hogy a program terjedelme lényegesen megnő, azonban így is elfér a rendelkezésre álló tárterületen.

1. táblázat

	Tár- szükség- let (prog- ram + pufferek) (szó)	Egy teljes transz- formá- ciós ciklus ideje (msec)	Egy pontra eső fel- dolgozási idő ( $\mu$ sec)	Adat- vesztés nélkül meg- enged- hető max. jelfrek- vencia (kHz)
64-pontos komplex DFT	2661	0,532	8,31	60,1
128-pontos komplex DFT	3454	3,14	24,5	20,4
256-pontos komplex DFT	3940	8,42	32,8	15,2
128-pontos valós DFT	3171	2,16	16,9	29,5
256-pontos valós DFT	3524	4,91	19,2	26,0

A külön végrehajtási időt nem igénylő léptetés hatékony és gyors, a túlsordulások elleni védekezéshez szükséges skálázást tesz lehetővé. Az alkalmazott módszer a módosított lépésenkénti skálázás, amely az ismert eljárások közül a legkedvezőbb jel-zaj viszonyra vezet [8].

Eddig 64-, 128- és 256-pontos komplex bemenetű és 128- és 256-pontos valós bemenetű gyors DFT-modulok készültek el. A programok jellemzőit az 1. táblázat foglalja össze. A programok részletes leírását és folyamatábráikat [4] tartalmazza.

## 5. A DFT-modulok számítási hibáinak becslése

A véges hosszúságú aritmetikát alkalmazó jelfeldolgozó rendszereknek a mérnöki gyakorlat szempontjából egyik alapvető jellemzője az elérhető pontosság. A DFT jelfeldolgozó processzorral történő meghatározásakor a számítási láncban először fellépő hiba az A/D átalakítással kapcsolatos. Az elkészült programok 12-bites adatokat dolgoznak fel: az A/D átalakítás eredménye úgy kerül egy 16-bites szóba, hogy az utolsó négy bit értéke határozatlan. Az esetleges kellemetlen hatások ellen az első adatátvitelkor a 11 értékes bitet 4 bittel jobbra léptetjük ( $2^{-4}$  értékű skálázás). A 12-bites lineáris kvantálásnál fellépő hiba a gyakorlatban rendszerint teljesülő feltételek esetén lineáris additív zajforrással modellezhető. A hiba statisztikai jellemzői az irodalomból [7] jól ismertek, s ezért a továbbiakban a kvantálási zajhatásait nem foglalkozunk.

Kimutatható [6], hogy a szorzókonstansok véges hosszúságából származó hiba áttranszformálható egy olyan hiba-jel viszonyra, amely a pontszám-mal csak lassan nő, és nagyobb pontszámokra már elhanyagolható a többi hibákhoz képest. A konstansokat ezért pontosnak tételezzük fel.

Jelölje  $e(k) = X(k) - \bar{X}(k)$  a transzformált sorozat  $k$ -ik tagjának hibáját.  $X(k)$  a pontos,  $\bar{X}(k)$  pedig a program által számított érték. Keressük a hiba  $E[e(k)]^2$  négyzetes várható értékét, és a

$$\sum_{k=0}^{N-1} E[e(k)]^2 \text{ teljes négyzetes várható értékét.}$$

Első lépés a lehetséges hibaforrások azonosítása.

A számítások során a 16-bites szavakat fix-pontos bináris törtekként kezeljük, azaz az előjel-bit után mindjárt a fixpont következik (a számok abszolút értéke tehát kisebb egynél). Két 16-bites szám szorzásakor a 32-bites eredményt 16-bitre csonkítjuk, azaz hiba keletkezik. Diszkrét, egyenletes eloszlású hibát feltételezve a csonkolási hiba várható értéke  $m = -2^{-15}/2$ , szórásnégyzete  $\sigma^2 = 2^{-30}/12$ . Két komplex szám szorzásakor az eredmény hibája  $m_M = -2^{-16}(1+j)$  és  $\sigma^2_M = 2^{-30}/6$  (mivel a 32-bites aritmetika miatt csak kétszer kell csonkolni).

A következő hibaforrás az összeadások utáni esetleges túlsordulások elleni védekezésül alkalmazott skálázás, ami minden lépésben két bittel

való jobbra léptetést, azaz 4 értékkel való szorzást jelent (minden lépésben négy komplex számot összegzünk, így az értékes bitek száma lépésenként legfeljebb kettővel nőhet). A két bittel való jobbra léptetéskor keletkező hiba az előző hibamodell

szerint  $m_s = -\frac{3}{8}2^{-15}(1+j)$  és  $\sigma_s^2 = \frac{5}{32}2^{-15}$  értékkel jellemezhető.

A számítások során azzal a szokásos feltevéssel élünk, hogy az egyes szorzások és leosztások során keletkező hibák statisztikusan függetlenek. A  $k$ -ik transzformált érték hibájának négyzetes várható értéke tehát:

$$E|e(k)|^2 = VM(k) + |MM(k) + MS(k)|^2 + VS(k) \quad (5-1)$$

ahol  $VM(k)$  a szorzási csonkolásokból eredő hiba szórásnégyzete,  $MM(k)$  a várható értéke,  $VS(k)$  a leosztási csonkolásból eredő hiba szórásnégyzete,  $MS(k)$  pedig a várható értéke a  $k$ -ik transzformált-ra nézve.

Következik a négy hibaösszetevő értékének számítása.  $VM(k)$  értékének számításakor azt kell figyelembe venni, hogy a forgatási tényezőktől függően háromféle pillangó különböztethető meg [4]. Az első típusban nincs szorzás, ezért kimeneti pontjainak szorzási csonkolásokból eredő hiba szórásnégyzete  $C_{1F} = 0$ . A második csoportba tartozó pillangóknál a 3 szorzás közül 1 triviális, ami nem okoz hibát, a szórásnégyzet  $C_{2F} = 2\sigma_M^2$ . A harmadik fajta pillangók 3 db hibát okozó komplex szorzást tartalmaznak:  $C_{3F} = 3\sigma_M^2$ .

$$\text{Legyen } q(k) = \min(i : k_i \neq 0) \quad \left( k = \sum_{i=0}^3 4^i k_i \right).$$

Belátható [3], hogy a  $k$ -ik transzformált előállításában az első  $q(k)$  lépésben  $C_{1F}$  típusú pillangók szerepelnek, a  $q(k)+1$ -ikben  $C_{2F}$  típusúak, ha  $k_{q(k)}$  páros, és  $C_{3F}$  típusúak, ha  $k_{q(k)}$  páratlan, végül a  $q(k)+2$ -ik lépéstől kezdődően az összes pillangó  $C_{3F}$  típusú. Könnyen megmutatható az is, hogy az  $m$ -ik lépésben  $4^{M-m}$  pillangó ( $M = \log_4 N$  a fokozatok száma és  $N$  a transzformáció hossza) kimeneti értékei szerepelnek a  $k$ -ik transzformált számításában. Mivel a forgatási tényezők abszolút értéke 1, a kimenet felé terjedő hibák szórásnégyzete nem változik (a másodrendű hibáktól eltekinthetünk), csak a skálázási léptetések hatására értékük fokozatonként  $\frac{1}{16}$  értékű szorzótényezővel csökken.

Az előzőeket összefüggések formájában összefoglalva:

$$VM(k) = 0 \quad k=0, \quad \frac{N}{4}, \quad \frac{N}{2}, \quad \frac{3N}{4}$$

$$VM(k) =$$

$$C_{3F} \left[ \frac{1}{16^{M-2}} \sum_{m=q(k)+1}^3 4^{M-m} + \sum_{m=4}^M \frac{4^{M-m}}{16^{M-m+1}} \right],$$

ha  $k_{q(k)}$  páratlan

$$VM(k) =$$

$$C_{3F} \left[ \frac{1}{16^{M-2}} \sum_{m=q(k)+2}^3 4^{M-m} + \sum_{m=4}^M \frac{4^{M-m}}{16^{M-m+1}} \right] + C_{2F} \frac{4^{M-q(k)-1}}{16^{M-q(k)}}, \quad (5-2)$$

ha  $k_{q(k)}$  páros.

Továbbá

$$\sum_{k=0}^{N-1} VM(k) = C_{3F} \left[ 11,6 + \sum_{m=4}^M \left( 4^{m-2} - \frac{1}{2} \right) \right] + 4C_{2F} \left[ 1,0625 + \sum_{m=4}^M 4^{2(M-m)} \right]. \quad (5-3)$$

Másodikként  $VS(k)$  számítása következik. Vegyük észre, hogy az  $m$ -ik lépés utáni leosztás miatt  $4^{M-m}$  pont hibája terjed a  $k$ -ik kimeneti pont felé, de csak  $\frac{1}{16^{M-m}}$  értékkel leosztott szórásnégyzettel ér el a kimenetre.

Tehát:

$$VS(k) = \sigma_S^2 \sum_{m=3}^M \frac{4^{M-m}}{16^{M-m}} \quad (5-4)$$

és

$$\sum_{k=0}^{N-1} VS(k) = \alpha_S^2 \frac{4^{M+1} - 64}{3}. \quad (5-5)$$

A hiba várható értékének számítására a [13]-beli módszert alkalmaztuk: a várható értékeket előbb a bemenetre transzformáltuk vissza, majd összegzésük után egy  $N$ -pontos diszkrét Fourier-transzformációval a kimenetre transzformáljuk.

Már szó volt arról, hogy [3] szerint egy  $N$ -pontos DFT számítása  $N/4$  hosszúságú transzformáltak számítására vezethető vissza, s ez a felbontás  $N/4 = 4$ -ig folytatható. Az  $m$ -ik lépés utáni skálázási leosztás hibájának várható értéke minden leosztott pontra  $m_S$  nagyságú. A bemenet felé való transzformációhoz ezt  $4^{m-2}$  értékkel súlyozni kell, mivel ekkora leosztás már volt, amikor a hiba fellépett.

A  $4^m$ -pontos inverz transzformációkat elvégezve és  $m$  szerint összegezve az  $n$ -ik bemeneti ponthoz tartozó ekvivalens hiba várható értéke:

$$E[es(n)] = m_S \sum_{m=3}^{f(n)} 4^{m-2} \quad (5-6)$$

ahol

$$f(n) = \begin{cases} M & \text{ha } n = 0 \\ M - \max_i (i: n_i \neq 0) & \text{egyébként.} \end{cases} \quad (5-7)$$

Továbbá

$$MS(k) = \text{DFT}\{E[es(n)]\}_n \quad 0 \leq n \leq N-1. \quad (5-8)$$

Hasonló módon járunk el  $MM(k)$  számításánál is. A szorzásokból eredő hibák várható értékét a forgatott pontok hibájának tekintve a pillangók bemenetére transzformáljuk vissza. Figyelembe kell venni azonban, hogy nem mindegyik pont rendelkezik majd hibával a triviális szorzások miatt. Az  $m$ -ik lépésben keletkező szorzási hibákat  $4^{m-3}$  értékkel való súlyozás után  $4^{m-1}$  pont-

számú inverz transzformációkkal a bemenetre transzformáljuk. Összegzés után:

$$E[e_M(n)] = m_M \sum_{m=2}^3 n_{M-m+1} \cdot 4^{1-m} \times \left[ \frac{W_m^{g(n,m)} - j^{g(n,m)}}{1 - W_m^{g(n,m)}} - h(n,m) \cdot (-1)^{g(n,m)} \right] + \frac{m_M}{16} \sum_{m=4}^M n_{M-m+1} \times \left[ \frac{W_m^{g(n,m)} - j^{g(n,m)}}{1 - W_m^{g(n,m)}} - h(n,m) \cdot (-1)^{g(n,m)} \right] \quad (5-9)$$

ahol

$$W_m = e^{j \frac{2\pi}{4^m}}$$

és

$$g(n,m) = \sum_{i=1}^m n_{M-m+i} \cdot 4^{i-1};$$

$$h(n,m) = \begin{cases} 1 & \text{ha } n_{M-m+1} = 2 \\ 0 & \text{egyébként} \end{cases} \quad (5-10)$$

Megfontolásaink eredménye:

$$MM(k) = \text{DFT}\{E[e_M(n)]\}_n \quad 0 \leq n \leq N-1 \quad (5-11)$$

Az előzőekben számított  $MS(k)$  és  $MM(k)$  értékek részletes számítására nincs szükség, ha csak a teljes négyzetes hiba várható értékére vagyunk kíváncsiak. A Parseval-tétel alapján ugyanis

$$\sum_{n=0}^{N-1} |X(k)|^2 = N \sum_{n=0}^{N-1} |x(n)|^2 \quad (5-12)$$

a teljes négyzetes hiba várható értéke pedig

$$\sum_{k=0}^{N-1} E[e(k)]^2 = \sum_{k=0}^{N-1} [VM(k) + VS(k)] + N \sum_{n=0}^{N-1} (E[es(n)] + E[e_M(n)])^2 \quad (5-13)$$

2. táblázat

$N$	A teljes négyzetes hiba várható értéke
64 (komplex)	$7,4 \cdot 10^{-8}$
128 (komplex)	$1,6 \cdot 10^{-7}$
128 (valós)	$8,7 \cdot 10^{-8}$
256 (komplex)	$5,3 \cdot 10^{-7}$
256 (valós)	$2,4 \cdot 10^{-7}$

Eddigi számításaink komplex bemeneti adatsorozatú, radix-4 szerinti DFT-programokra vonatkoznak. Valós bemeneti adatok, ill. kevert radixú számítás (128-pontos DFT!) esetén a transzformációs algoritmusok az utolsó lépésben némiképp

eltérnek a vázolt struktúrától. A teljes négyzetes hiba azonban összefüggéseink alapján könnyen számítható. A számszerű eredményeket a 2. táblázat foglalja össze.

## 6. A spektrumbecslés számítása

Az elméleti és gyakorlati alapozás után következhet a spektrumbecslés konkrét meghatározása. A kifejlesztett gyors DFT-modulokra épülő program  $N = 2048$  elemű mintákból állítja elő a módosított, ablakolt periodogramok alapján a spektrumot (2—9). A mintákat  $K = 8$  db, egymással nem átlapolódó részsorozatra bontottuk. Számítástechnikai egyszerűsége miatt a háromszög alakú (Bartlett-féle) ablakozófüggvényre (2—3) esett a választás. A számításhoz 256-pontos DFT-modulokra volt szükség. A modul ki lett egészítve az utolsó transzformációs lépésben az abszolútérték-számítással és az átlagolással.

A program tárfoglalása a négy pufferral együtt 3918 szó, s egy becslési ciklus végrehajtási ideje 50,88 ms. Azaz a maximális jelfrekvencia, amely mellett még nincs adatvesztés  $f_{\max} = 20,1$  kHz.

## 7. Köszönetnyilvánítás

Ezúton szeretnénk köszönetet mondani dr. Simonyi Ernőnek, a Távközlési Kutató Intézet főosztályvezetőjének, aki a munkához szükséges TMS32010 fejlesztőrendszert a rendelkezésünkre bocsátotta.

## IRODALOM

[1] Bartlett, M. S.: „An Introduction to Stochastic Processes with Special Reference to Methods and

Applications”, Cambridge University Press, New York (1985).

- [2] Blackman, R. B.—Tukey, J. W.: „The Measurement of Power Spectra”, Dover Publications, Inc., New York (1958).
- [3] Kocsis F.: „Gyors eljárások a diszkrét Fourier-transzformáció számítására”. Híradástechnika, vol. 35. no. 12. pp. 544—549 (1984), vol. 36. no. 1. pp. 31—39 és no. 5. pp. 214—219 (1985).
- [4] Marx F.: „Gyors DFT-modulokra épülő periodogram alapú spektrumbecslés”. TDK dolgozat (1986).
- [5] Nussbaumer, H. J.: „Fast Fourier Transform and Convolution Algorithms”, Springer Verlag (1981).
- [6] Oppenheim, A. V.—Weinstein, C. J.: „Effects of Finite Register Length in Digital Filtering and the Fast Fourier Transform” Proceedings of the IEEE, vol. 60. no. 8. (1972)
- [7] Oppenheim, A. V.—Schäfer, B. W.: „Digital Signal Processing”, Prentice Hall, Inc., Englewood Cliffs, NJ (1975).
- [8] Prakash, S.—Rao, V. V.: „Fixed-point Error Analysis of Radix—4 FFT”. Signal Processing, no. 3. 23pp. 1—133 (1981).
- [9] Simonyi E.: „Digitális szűrők”. Műszaki Könyvkiadó (1984).
- [10] TMS32010 User's Guide, Texas Instruments, Inc.
- [11] TMS32010 Evaluation Module User's Guide
- [12] TMS32010 Analog Interface Board User's Guide
- [13] Tran-Thong-Liu, B.: „Fixed-point Fast Fourier Transform Error Analysis”. ASSP—24, vol. 24. no. 6. pp. 563—573 (1976).
- [14] Welch, P. D.: „The Use of Fast Fourier Transform for the Estimation of Power Spectra”. AU—15, vol. 15. no. 6. pp. 70—73 (1975).
- [15] Winograd, S.: „On Computing the Discrete Fourier Transform”. Math. Comput., vol. 32. pp. 176—199 (1978).

**21-ÉVEL ÖN IS NYER!**

**MECMAN 21 ÉVES SVÉD-MAGYAR KOOPERÁCIÓ**

**FINOMSZERELVÉNYGYÁR EGER**