



デジタル信号処理の動向*

田中公男** 青山友紀**

1. はじめに

デジタル数値の代表的演算により信号のろ波・変復調・等化などの機能を実現する手段、および信号にフーリエ変換・ヒルベルト変換などの処理を施す手法はデジタル信号処理 (Digital Signal Processing, DSP) と総称される。DSPの基本技術であるデジタルフィルタと高速フーリエ変換 (FFT) は1960年代前半から中頃にかけて登場し、以後電子計算機によるシミュレーション技法として音声情報処理の研究を中心にその理論体系の拡充が進められてきた。1970年代に入り本格的 IC 時代、さらには近年の LSI の発展期を迎えるに及び、DSP は単なるシミュレーション技法から論理・記憶機能を駆使して実際の通信システムを構成する一基本技術へと大きく進展するに至った。

DSP は機能の実現手段 (Implementation) の一つであり表-1 に示す特徴を有している。最大の長所は、デジタル処理であるためアナログの手段では構成不可能な機能を高精度・高安定に実現しうる点にある。また、AD・DA 変換を除き回路は論理・記憶素子で構成されるため LSI 技術との親和性が高く、音声帯域のような低周波領域においてはアナログ回路に比し小形化・低価格化が達成しうる可能性がある。このように DSP 技術の適用は、システムの機能・特性面における著しい発展を促し、今後とも極めて広い分野への応用が期待されている。

本文においては、DSP の基本となるデジタルフィルタ、および各種変換技術のうち最近の話題である Winograd Fourier Transform と Number Theoretic Transform について概説し、また DSP のリアルタイム処理を実行するプロセッサおよびそれを構成す

る LSI について述べるとともに、マイクロプロセッサによる信号処理についてもふれることとしたい。さらに、DSP の通信分野への応用例として時分割多重信号と周波数分割多重信号の相互変換を行うトランスマルチプレクサについて紹介する。

2. デジタルフィルタ^{1,2)}

2.1 デジタルフィルタの記述と分類

フィルタの機能は通信のあらゆるところに用いられている極めて重要な基本機能である。したがってその理論も DSP における最も重要な部分を占めている。デジタルフィルタは一般に、多入力数値系列 $U(n) = (u_1(n), \dots, u_l(n))$ を多出力数値系列 $Y(n) = (y_1(n), \dots, y_m(n))$ に変換するブラックボックス X で表わすことができる。入出力数値系列の1次元変数 n (n は時間的順序、位置的順序その他何を表わしてもよい) を、 n_1, n_2, \dots と2次元以上に拡張することもでき、近年は2次元デジタルフィルタ³⁾の研究が活発に行われている。 X は線形/非線形、時変/時不変いずれでもよい。ここではデジタルフィルタ理論が最も成功している線形・時不変の1次元デジタルフィルタに限定して紹介する。

ブラックボックス X が加算、係数乗算および単位遅延の3つの演算子のみから成る場合、 X は線形・時不変システムであり、状態方程式を用いて表現できる。

$$\left. \begin{aligned} X(n+1) &= AX(n) + BU(n) \\ Y(n) &= CX(n) + DU(n) \end{aligned} \right\} \quad (1)$$

(1)式における記述の規則を図-1 (次頁参照) の例を

表-1 デジタル信号処理の特徴

利 点	欠 点
○高精度の機能が実現できる	○能動素子が多量に必要となり回路が複雑化する
○高安定な特性が実現できる	○AD, DA 変換器が必要となる
○均一な特性が実現できる	○適用周波数範囲が低い
○可変特性が容易に実現できる	○有限遅延処理による雑音が発生する
○機能をプログラム化できる	
○回路がデジタル化され LSI 化しやすい	

* An Overview of Digital Signal Processing by Kimio TANAKA and Tomonori AOYAMA (YOKOSUKA Electrical Communication Laboratory Nippon Telegraph and Telephone Public Corporation).

** 日本電信電話公社 横須賀電気通信研究所

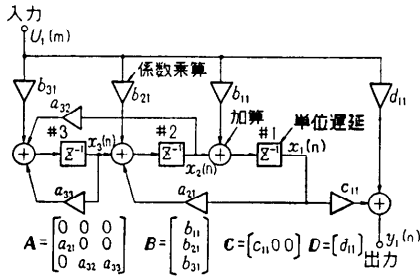


図-1 デジタルフィルタの状態方程式表示例

用いて説明する。

状態ベクトル $X(n) = (x_1(n), \dots, x_N(n))$ の要素 $x_i(n)$ は i 番目の単位遅延の出力数値系列である。行列 $A = [a_{ij}]$ の要素 a_{ij} は j 番目の単位遅延の出力から i 番目の単位遅延の入力に直結されている加算器に至るブランチの重み係数値 (例えば図-1 の a_{21} , a_{32} , a_{31}) を与える。同様に行列 $B = [b_{ij}]$ は入力から加算器への (例えば b_{11} , b_{21} , b_{31}), $C = [c_{ij}]$ は単位遅延の出力からフィルタ出力への (例えば c_{11}), $D = [d_{ij}]$ は入力から出力への (例えば d_{11}), 各ブランチの重み係数値である。以下議論を 1 入力, 1 出力のデジタルフィルタに限れば, A は $N \times N$, B は $N \times 1$, C は $1 \times N$ の行列で, D はスカラーとなる。

z 変換表示された次式。

$$H(z) = D + C(zI - A)^{-1}B, \quad I: \text{単位行列} \quad (2)$$

を伝達関数と呼び, $z = e^{j\omega}$ を代入した $H(e^{j\omega})$ は周波数応答, $H(z)$ を逆 z 変換した $h(n)$ はインパルス応答を与える。(2)式の伝達関数は z の有理関数となり,

$$H(z) = \sum_{r=0}^R \alpha_r z^{-r} / (1 + \sum_{k=1}^K \beta_k z^{-k}) \quad (3)$$

で与えられる。上式で $\beta_i, i \neq 0$ の少くとも 1 つが 0 でない場合を“巡回型”と呼ぶ。 $\beta_i = 0, i \neq 0$ の場合

$$H(z) = \sum_{r=0}^R \alpha_r z^{-r} = \sum_{n=0}^R h_n z^{-n} \quad (4)$$

となり, 重み係数値 α_r はフィルタのインパルス応答 h_n に一致する。これを“非巡回型”フィルタと呼ぶ。一方, デジタルフィルタのインパルス応答が無限に継続するものを“IIR (Infinite Impulse Response)”, 有限個しか継続しないものを“FIR (Finite Impulse Response)”と呼ぶ。これらは用途に応じて使い分ける必要がある。

デジタルフィルタの演算は計算機が専用プロセッサで実行されるから, すべての数値は有限語長にま

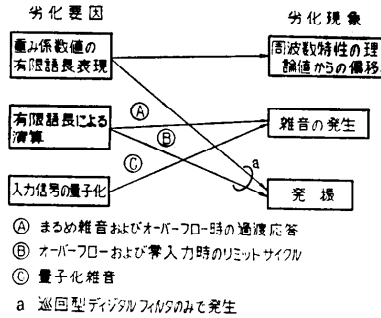


図-2 デジタルフィルタ特性劣化

められる。その結果フィルタは正規の動作から若干の偏移を生じ, 図-2 に示すようなデジタルフィルタ特性の劣化を引き起こす。

一方, 近年 Charge Coupled Device (CCD) や Bucket Brigade Device (BBD) の電荷転送素子 (Charge Transfer Device: CTD) や Switched Capacitor⁴⁾ を用いたサンプル値フィルタ (Sampled Filter) が注目されている。サンプル値フィルタにおいては数値系列や重み係数値は転送される電荷量や抵抗値などのアナログ量を表わしており, 実際の回路では電荷の漏洩や抵抗値の製造誤差により特性が劣化する。このようなサンプル値フィルタでの劣化やデジタルフィルタでの有限語長表現による劣化のない理想状態では両者の記述式は同一であり区別する必要はない。両者を合わせて“Discrete Time System¹⁾”と呼ぶことがある。以上のデジタルおよびサンプル値フィルタの分類を図-3 に示す。

2.2 設計

デジタルフィルタの設計は仕様が決めると (3) 式または (4) 式を用いて理想特性の近似を行って伝達関数 $H(z)$ を決定し (近似), 次にその伝達関数を与える係数乗算, 加算および単位遅延の 3 演算子の結線のしかた (これをデジタルフィルタの構造と呼び, シグナルフローグラフで表わす) を定め, 最後に係数値をしかるべき語長に定める。以上をデジタルフィルタの“Synthesis”と呼ぶ。Synthesis によって得られたデジタルフィルタが実際に仕様を満足しているか否かをチェックするために, 周波数特性,

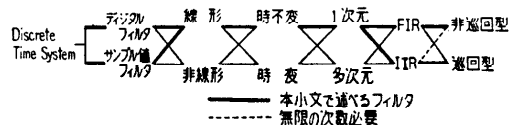


図-3 デジタルフィルタおよびサンプル値フィルタの分類

表-2 実用上重要なデジタルフィルタの近似法

分類	近 似 法	IIR/FIR	備 考	公開されているプログラム
I	双一次 ϵ 変換法 ¹⁾²⁾	IIR	矩形状減衰特性のローパス・ハイパス・バンドパス・バンドストップフィルタは大部分本手法で設計される	Gray & Markel (双一次 ϵ 変換による楕円フィルタの)設計用
	最小二乗近似法 ²⁾	"	矩形状以外の任意特性の近似に用いる	Steiglitz ⁴⁾ (Fletcher-Powell 法による最適化)
	時間域逐次近似法	"	所望のインパルス応答を最小二乗近似するフィルタを得る	
II	Remez Exchange 法 ²⁾	FIR	Minimax の意味で最適な直線位相 FIR フィルタが設計できる	McClellan, Parks & Rabiner ⁴⁾
	Frequency Sampling 法 ²⁾	"	Remez Exchange 法で設計できないような高次の FIR フィルタの設計に用いる	
	Herrmann & Schuessler 法 ³⁾	"	Minimax の意味で最適な最小位相推移 FIR フィルタの設計に用いる。次数を上げるには工夫が必要である	

I : s-平面におけるアナログフィルタの近似法を利用する手法

II : z-平面において直接近似を行う手法

インパルス応答, まるめ雑音などを求める手法が必要となる。これをデジタルフィルタの“Analysis”と呼ぶ。

近似問題は, 近似対象(減衰, 位相, 群遅延の各特性, インパルス応答など), 理想特性の形(矩形状, 直線状, 平坦など), 近似に用いる関数系(多項式, 有理関数), 近似の測度(ミニマックス, 最小二乗近似など)の条件の組合せに対応して数多くの手法が提案されている^{1,2)}。代表的な手法を表-2 に示す。

最近の近似問題のトピックスには最小位相推移(Minimum Phase) FIR フィルタと2次元デジタルフィルタ³⁾の設計問題がある。FIR フィルタは一般に次数が高くなり, それにともなって信号の遅延が増大する。音声を対象とする通信システムでは直線位相の条件は必ずしも必要なく, 遅延時間の増加がより重大であり, 最小位相推移特性が重要となる。また直線位相の条件を除くことによって同じ規格を得るための次数が低減され金物の節約が図れる。最近 Minimax の意味で最適な減衰特性を有する最小位相推移 FIR フィルタの近似法がいくつか試みられている^{5,6)}。

2次元以上の近似問題は多項式の因数分解が必ずしも可能でなく, IIR フィルタの場合の安定性判別も困難な問題となる。また近似論で重要な“Haar の条件”が満足されないので最適近似のユニーク性が保証されなくなるなど, 単なる1次元の拡張では問題は解決しない³⁾。2次元 FIR フィルタは安定性の問題がなく IIR フィルタに比べて研究が進んでいる⁷⁾。

デジタルフィルタの構成論も数多くの研究がある⁴⁾。近似によって得られた $H(z)$ に対して(2)式を解き, 行列 A, B, C, D が求まれば, フィルタ構成は機械的に書き下させるので, Synthesis は終了す

表-3 デジタルフィルタの主な構造⁴⁾

構 造	行列 A	構 成 法
直接形	同伴行列	有理関数の直接構成 係数感度が高く実用されない
縦続接続形	ブロック 三角行列	1次・2次への因数分解 ベアリング・オーダリング必要 実用上最も多用される
並列接続形	ブロック 対角行列	部分分数量開
梯子形	三重対角 行列	伝達関数の連分数量開 2端子対回路の縦続分解 行列 A, B, C, D の等価変換
格子形		伝達関数分子多項式の直交多項式の和による表現 PARCOR 音声分析合成で用いられる
多重フィード バック形	Comrad 行列その他	アクティブフィルタにおける多重フィードバック形構成法に準ずる方法 多重フィードバック項追加による極の修正法
Wave デジタル フィルタ		LC 梯子形フィルタの Wave 量のデジタルシミュレーション
アナログ回路 網の s- ϵ 変換 による構成		アナログ回路の電圧-電流領域, 電圧領域, 電圧・電流の線形変換された領域におけるシグナルフローグラフのデジタルシミュレーション

るが, (2)式の解は一意ではなく, 同じ伝達関数を与える無数のシグナルフローグラフ(以下 SFG と略す)が存在するので, 有限語長劣化を最小にする SFG は何かという命題が生ずる。これまでの数多くの試みにもかかわらずこの問題の一般解は未だ得られていない。表-3 は主なデジタルフィルタの構成とそれに対する行列 A の形を示す。状態方程式はデジタルフィルタの構成を表現できるのでそれを用いてフィルタ構成論を統一的に扱おうとする試みがいくつか発表されている⁸⁾。

デジタルフィルタの Synthesis としては更に, オーバーフローを生じさせないで信号対まるめ雑音比を最大にするスケールリング法, ベアリング・オーダリング法, 大振幅および小振幅のリミットサイクル, など

の諸問題がある²⁾。

2.3 解析と CAD

Synthesis によって得られたデジタルフィルタのシグナルフローグラフより、その周波数応答、時間応答、まるめ雑音特性などを、どのような構成に対しても能率よく計算するのが Analysis の問題である。SFG が与えられると、それに対応する節点方程式は容易に得られる。

$$W = FW + GU \quad (5)$$

ここに、 $W = (\omega_1, \omega_2, \dots, \omega_r)^t$ であり、 ω_i は SFG の入力を除く i 番目のノードである。 U は入力ベクトル、 F 、 G はブランチの重み係数を記述する行列である。

(5) 式の節点方程式から直接周波数応答を求める直接法はノード数が増すと計算量、必要メモリ容量が膨大となる。節点方程式より伝達関数を求め、 $z = e^{j\omega}$ を代入する方法は演算量は減るが、 z^{-1} なる変数を含む行列演算が必要となり、また一般固有値問題を解く必要がある。これは Purdue 大学の Belter らによって開発された CAD システム “DINAP II” で用いられている方法である⁹⁾。これに対して節点方程式より一旦状態方程式を求め、それより伝達関数、周波数応答を順次計算する手法および節点方程式より状態方程式を導びく代数的手法とグラフ的手法が筆者らにより提案されている^{10,11)}。本手法は定数行列の数値処理のみで解析が行われ、演算時間、所要メモリ容量に優れており、現在通研(日本電信電話公社電気通信研究所)で開発中の CAD システム “USAP-D”¹²⁾ の解析プログラムで用いられている。

時間応答は伝達関数 $H(z)$ の逆 z 変換、あるいは状態方程式を解いて $x(n)$ を求める方法があるが、シグナルフローグラフを計算機シミュレーションする方法が演算回数が少なく最も有利である。本方法は結局差分方程式を計算することであるが、ケースバイケースでプログラムを書いていたのでは CAD の意味がない。筆者らは SFG が与えられると Crochiere の方法によってその “Precedence Form”¹³⁾ を求め、それに従ってプログラムを機械的に書くアルゴリズムを提案している¹³⁾。

まるめ雑音の解析は Jackson のモデル¹⁴⁾ を用いてまるめが生ずる乗算器出力よりデジタルフィルタの出力までの伝達関数 $G_j(z)$ を求め、次式により雑音電力スペクトルを得る。

$$N_v(e^{j\omega}) = \sigma_v^2 \sum_j K_j |G_j(e^{j\omega})|^2 \quad (6)$$

表-4 デジタルフィルタ CAD システムの例

CAD システム	開発機関	特 徴
DINAP-II ⁹⁾	Purdue 大学	Symbolic Analysis によりデジタルフィルタの定常応答、過渡応答、まるめ雑音等の解析を行う
CADNAP	MIT	デジタルフィルタの解析を行うプログラムパッケージである
FILSYN ¹⁰⁾	Rockwell International	本システムは LC、アクティブおよびデジタルフィルタの Synthesis および Analysis を行う CAD システムである
USAP-D ¹²⁾	通研	近似・構成・解析を含むデジタルおよびサンプル値フィルタ用 CAD システムである 解析は状態方程式を用い、周波数応答時間応答、まるめ雑音スペクトラムが求められる
DINETS ¹¹⁾	日本電気	近似・構成・解析を行う CAD システムであり、解析は実時間シミュレーションをベースにしている

$G_j(z)$ は前述の節点方程式→状態方程式→伝達関数→周波数応答の計算プロセスより容易に得られる。

デジタルフィルタやサンプル値フィルタを実際の装置に適用する上で、計算機を駆使して Synthesis と Analysis を行う CAD (Computer Aided Design) 技術の必要性が極めて高く、近年はデジタルフィルタ用 CAD の研究がトピックスとなっており、表-4 に示すようなシステムが発表されている^{9,12,15,16)}。

3. 高速フーリエ変換と数論変換

3.1 FFT と WFT

今日、工学のあらゆる分野において各種数学的変換が必要欠くべからざるものとなっているが、それらは計算機を用いてデジタル的に処理される場合が多く、DSP の重要な基本技術の一つとなっている。これらの諸変換を図-4 に示す。このうちフーリエ変換はその重要性から工学の基礎といってもよく、計算機処理に適する離散的フーリエ変換 (DFT) の効率的演算法である FFT アルゴリズム発見の与えたインパクトは図り知れない。この 1965 年の発見より 10 余年が経過し、FFT を中心とする計算機を利用する変換技術の進歩は目覚ましいものがある。ここでは FFT に関する最近のトピックスである Winograd のアルゴ

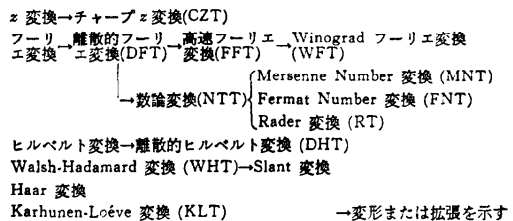


図-4 デジタル信号処理で用いられる諸変換

リズム (WFT と呼ばれる)¹⁷⁾と、たたみ込みを高速に行える数論変換にしばってその概要を紹介するに留める。

1975 年に Winograd によって提案された WFT はある場合には FFT に比べて乗算回数が 20% に低減でき、加算回数の増加はほとんどないという優れたアルゴリズムであり、各方面の注目を集めている。

サンプル数 N の DFT を行列表示すると、

$$\mathbf{X} = \mathbf{D}_N \mathbf{x} \quad (7)$$

$$D_N(m, n) = [W_N^{mn}], \quad m, n \in [0, N-1] \quad (8)$$

$$W_N = \exp(-j2\pi/N) \quad (9)$$

ここで \mathbf{D}_N の Canonical 分解を行う。

$$\mathbf{S}_N \mathbf{C}_N \mathbf{T}_N = \mathbf{D}_N \quad (10)$$

ここに \mathbf{T}_N は $J \times N$ 行列でその要素は $[0, 1, -1]$ のみから成る incidence 行列, \mathbf{C}_N は $J \times J$ の対角行列, \mathbf{S}_N は $N \times J$ の incidence 行列である。

J が $J=N^2$ 程度に大なる時 (10) 式の分解が存在することは容易に示せるが, $J < N^2$ の時 (10) 式を解いて incidence 行列を定めることは容易でない。Winograd は $N=2, 3, 4, 5, 7, 8, 9, 16$ の 8 種類 (これらは素数ないし素数の整数乗である) に対して, $J \approx N$ なる J について (10) 式の解を見出した。

これは (7) 式に必要な乗算回数が $O(N)$ であることを意味する。これを “small- N DFT” と呼ぶ。表-5 に small- N DFT に必要な乗算および加算回数を示す。Winograd は更に, 大きな N に対して small- N を組合せて適用することにより, 全体で大きな演算量の低減が得られるアルゴリズムを示した。 N が互いに素な m 個の数の積とする。

$$N = N_m \cdot N_{m-1} \cdots N_1 \quad (11)$$

まず N 個の入出力データを然るべく並べ換えた列ベクトルを $\mathbf{x}_{N'}$, $\mathbf{x}_{N'}$ とする。このとき

$$\mathbf{x}_{N'} = (\mathbf{D}_{N_m} * \mathbf{D}_{N_{m-1}} * \cdots * \mathbf{D}_{N_1}) \mathbf{x}_{N'} \quad (12)$$

が得られる。ここに * はクロネッカー積であり, \mathbf{D}_{N_i} は N_i 点 DFT の係数マトリクスである。今 (12) 式の N を small- N DFT の中から選べば (10) 式より,

$$\mathbf{D}_{N_i} = \mathbf{S}_{N_i} \cdot \mathbf{C}_{N_i} \cdot \mathbf{T}_{N_i} \quad (13)$$

とすることができる。したがって (12) 式は,

$$\mathbf{x}_{N'} = \mathbf{S}_N \cdot \mathbf{C}_N \cdot \mathbf{T}_N \cdot \mathbf{x}_{N'} \quad (14)$$

ここに, $\mathbf{S}_N = (\mathbf{S}_{N_m} * \mathbf{S}_{N_{m-1}} * \cdots * \mathbf{S}_{N_1})$

$$\mathbf{C}_N = (\mathbf{C}_{N_m} * \mathbf{C}_{N_{m-1}} * \cdots * \mathbf{C}_{N_1})$$

$$\mathbf{T}_N = (\mathbf{T}_{N_m} * \mathbf{T}_{N_{m-1}} * \cdots * \mathbf{T}_{N_1})$$

で与えられる。この時, N_i 点 small- N DFT の乗算

表-5 Winograd small- N DFT の乗算加算回数

N	乗算回数	加算回数	N	乗算回数	加算回数
2	2	2	7	9	36
3	3	6	8	8	26
4	4	8	9	13	44
5	6	17	16	18	74

(注) 入力データ実数時の値
入力データ複素数の場合は 2 倍する
乗算回数には $W^0=1$ の乗算も含む

回数を M_{N_i} とすると, N 点 DFT に要する全乗算回数 M は

$$M = M_{N_m} \cdot M_{N_{m-1}} \cdots M_{N_1} \quad (15)$$

以上が WFT アルゴリズムの概要である。詳細は文献 17) を参照されたい。WFT の特徴を下記に示す。

- (1) 乗算回数の減少は WFT 可能な長さの FFT に対して約 20% 程度である。
- (2) WFT の複素データに対する乗算回数は実データのその丁度 2 倍にしかならない。
- (3) WFT では FFT の数倍のメモリ容量を必要とし, 処理手順も FFT より複雑となる。

3.2 数論変換¹⁸⁾

数論変換 (Number Theoretic Transform, 以下 NTT と略す) は有限語長の離散的たたみ込みを高速にしかも誤差なく計算する方法として提案された変換である。

有限長の離散的たたみ込みは次式で定義される。

$$y(n) = \sum_{m=0}^{N-1} h(n-m)x(m) \quad n=0, 1, \dots \quad (16)$$

上式を $y(n) = h(n) \otimes x(n)$ と記す。 $x(n)$ と $h(n)$ の DFT に対して,

$$\text{DFT}[h \otimes x] = \text{DFT}[h] \cdot \text{DFT}[x] \quad (17)$$

が成り立つとき, Cyclic Convolution Property (以下, CCP と略す) を有するという。このとき,

$$y(n) = \text{IDFT} \{ \text{DFT}[h] \cdot \text{DFT}[x] \} \quad (18)$$

となり, 2 回の DFT と 1 回の IDFT で (16) 式のたたみ込みが計算できる。そこで CCP の性質を保存した新しい変換を導入し, それを用いて (18) 式の計算を効率的に計算することを考える。次の変換,

$$\left. \begin{aligned} X(K) &= \sum_{n=0}^{N-1} x(n)\alpha^{nK} \quad [\text{Mod } M] \\ x(n) &= \frac{1}{N} \sum_{K=0}^{N-1} X(K)\alpha^{-nK} \quad [\text{Mod } M] \end{aligned} \right\} \quad (19)$$

が CCP を有するためには, N が,

$$\alpha^N = 1 \quad [\text{Mod } M] \quad (20)$$

を満たす最小の正整数である, すなわち α は 1 の原

始 N 乗根であることが必要十分である。複素数体上では $\alpha=e^{-j2\pi/N}$ が、すなわち DFT が CCP を有する唯一の変換であるが、別の体上では CCP を有する新しい変換が可能である。計算機による処理を前提にすると M を法とする (Mod M) 整数環上で定義すると演算のまるめによる誤差が発生せず興味深い変換が得られる。 N, M, α を適当に選べば、変換に乗算が不要で加算とシフトのみでよく、しかも FFT に類似する高速演算アルゴリズムの適用可能な極めて有用な変換が得られる。以上のような変換を総称して NTT と呼ぶ。

$M=2^p-1$ (P : 素数) なる M を Mersenne 数と呼ぶ。Mersenne 数を法とする変換、

$$\left. \begin{aligned} X(K) &= \sum_{n=0}^{P-1} x(n)2^{nK} \text{ [Mod } M] \quad K=0, 1, \dots, P-1 \\ x(n) &= R \sum_{K=0}^{P-1} X(K)2^{nK} \text{ [Mod } M] \quad n=0, 1, \dots, P-1 \end{aligned} \right\} \quad (21)$$

$$R \cdot P=1 \text{ [Mod } M]$$

を Mersenne 数変換 (MNT) と呼ぶ。MNT は $P(P-1)$ 回の加算と $(P-1)^2$ 回のシフトのみによって実行され、巡回たたみ込みは 3 回の MNT と $2P$ 回の乗算によって計算できる。しかも演算中まるめによる誤差は生じない。しかし変換点数 P は素数の積に分解できないから FFT に類似の高速算法は適用できない。

M を Fermat 数 F_t に選ぶこともできる。ここに、

$$M=F_t=2^{2^t}+1, \quad b=2^t \quad (22)$$

である。Fermat 数を法とする変換を Fermat 数変換 (FNT) と呼ぶ。FNT の中で α を 2 の冪に選んだ場合、変換が極めて効率的に行えるので、これを特に Rader 変換 (RT) と呼ぶ¹⁷⁾。表-6 に FNT におけるパラメータの可能な組合せをいくつか示す。RT も MFT と同様に加算とビット処理のみで計算され、誤差なしの巡回たたみ込みが可能であり、Parseval の法則等 DFT と類似の性質を有する。しかも MNT と異なり N が素因数分解可能であり、FFT に類する高速算法が適用できる。

FNT はこのように多くの利点を有するが、次のような重大な欠点を有する。

- (1) 語長とデータ長に関係があり自由に選べない。
- (2) 巡回たたみ込みの入出力データの大きさが $M/2$ を超えることは許されない。

$\alpha=2$ の FNT に対しては語長の 2 倍、 $\alpha=\sqrt{2}$ の

表-6 Fermat 数変換のパラメータと演算時間

(a) Fermat 数変換におけるパラメータの組合せの例

t	b	F_t	N^* ($\alpha=2$)	N ($\alpha=\sqrt{2}$)	N_{max}	α for N_{max}
3	8	2^3+1	16	32	256	3
4	16	2^4+1	32	64	65536	3
5	32	2^5+1	64	128	128	$\sqrt{2}$
6	64	2^6+1	128	256	256	$\sqrt{2}$

* Rader Transform に相当する。

(b) FNT と FFT によるたたみ込み演算時間の比較

N	FFT (ms)	FNT or RT (ms)
32	16	3.3
64	31	7.4
128	60	16.6 ^a
256	123	40.0 ^b
256	123	80.0 ^c
512	245	166.0 ^c
1,024	530	340.0 ^c
2,048	1,260	720.0 ^c

^aUsing $\alpha=\sqrt{2}$.

^bUsing 2 by 128 convolution.

^cUsing two-dimensional RT.

FNT に対しては語長の 4 倍にデータ長が限定されるので変換長 N を大きくすると語長の増加が著しい。これを避けるため、データ長の大きな巡回たたみ込みは 2 次元 FNT を用いた 2 次元巡回たたみ込みを適用すると効果的である¹⁸⁾。表-6 に長さ N の実数データの巡回たたみ込みに要する時間を FFT と FNT または RT とで比較した結果を示す。

最近、FNT を実行する専用プロセッサの研究も試みられている。McClellan は語長 16 ビットの 64 点 FNT プロセッサを試作している¹⁹⁾。ECL 10K を含む 283 個の IC を用い 38MHz のクロックで動作する。64 点 FNT を 6 個用いて 1,024 点の実信号たたみ込みが可能である。

4. デジタル信号処理用プロセッサと LSI

4.1 専用プロセッサと LSI

通信の分野に適用する場合のようにリアルタイムな処理のコストパフォーマンスが問題となる場合には専用のハードウェアを構成する必要がある。デジタル信号処理と通常の情報処理は共通点もあるが、次のような相異点もあるのでハードウェア構成上留意する必要がある。

- (1) 命令の種類が少なく、 $A_1x_1+A_2x_2$ (A_1, A_2 は定数、 x_1, x_2 は変数) のような定まった演算の繰返しが大部分を占める。
- (2) 乗算命令の頻度が高い。
- (3) 使用メモリは通常の情報処理に比して少な

い。

(4) ワード単位の処理が大部分でありビット処理の行われることは少ない。

DSP 用プロセッサの設計に際し決定すべきパラメータとして下記の項目が考えられる²⁰⁾。

- (1) 完全ハードワイヤードかプログラム制御型プロセッサか、あるいはその中間か。
- (2) 直列演算か並列演算か。
- (3) 固定/浮動/ブロック浮動小数点のいずれか。
- (4) 数値表現形式。
- (5) IC, LSI の品種。

上記(1)と(2)は相関がある。直列演算はプログラム制御との整合が良くなく、一方並列演算でハードワイヤードプロセッサを構成すると所要ゲート数が膨大となるので、直列演算-ハードワイヤード、並列演算-プログラム制御の組合せがとられることが多い。

直列演算と並列演算の比較を DSP で最もゲート数および処理時間を要する乗算器について検討してみる。アレー型並列乗算器²¹⁾、直並列乗算器²¹⁾およびパイプライン乗算器²¹⁾について比較すると図-5が得られる。同図より、乗算器をパイプライン化してスループットの向上を図れば直列演算のコストパフォーマンスは並列演算より優れていることがわかる²⁰⁾。また直列演算は端子数やバスの数量が大幅に減少し、実装密度が向上する。以上より直列演算によるハードワイヤード構成は決まった演算の繰返しの多いリアルタイム DSP に対しては最もコストパフォーマンスの高い方法である。

一方、並列演算-プログラム制御方式は異なる機能に対してハードウェアの共通化が図れる点やプログラムの変更のみでパラメータの変更や機能の拡張が可能であるなどハードワイヤード方式にはない優れた特徴を有している。

近年、DSP を主たるマーケットとした LSI の開発が活発化している。これらは、(a) 乗算器、(b) $\sum A_i x_i$ 型演算器、(c) 可変長シフトレジスタ²²⁾、(d) 2次巡回型デジタルフィルタ²³⁾およびその他の機能ブロック、に分類できる。乗算器は汎用性が高いので多くの品種が発表されている。図-6 はそれらの性能を比較したものである。

超 LSI 時代の到来を迎え、今後の DSP 用 LSI の動向は次の2つが考えられる。一つはデータ用 MO-

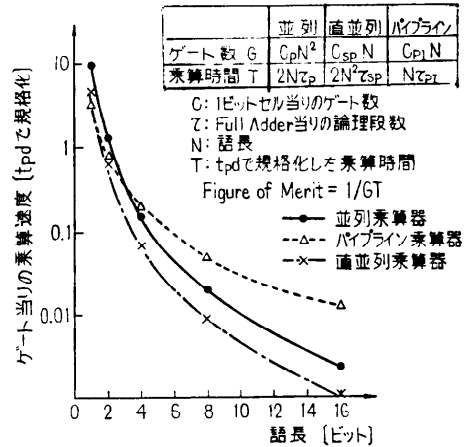


図-5 並列乗算器、直並列乗算器、パイプライン乗算器の比較

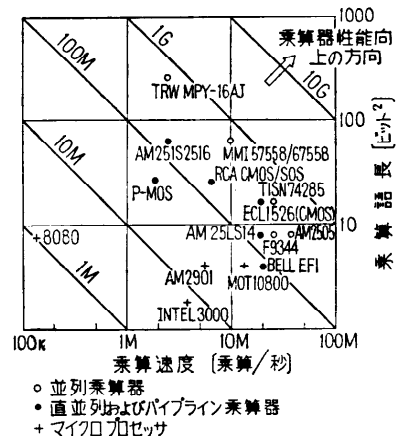


図-6 乗算器の性能比較

DEM²⁴⁾や音声合成装置など、複雑な機能を1チップ化したカスタム LSI であり、他の一つはより汎用性を考慮した DSP 用プロセッサの LSI 化である。TRW 社の SPAU²⁵⁾や通研で開発中の LAP (Linear Arithmetic Processor)²⁶⁾は後者の傾向に沿うものである。

従来、デジタルフィルタや FFT について、ROM を用いた構成²⁷⁾や乗算器を用いない構成²⁸⁾など、多くの工夫やアイデアが提案され、回路の単純化が追求されてきた。しかしながらこれらは従来の SSI や MSI 時代を背景とした考え方のもので多く、今後の超 LSI 時代に必ずしもマッチしない。そこでは特定の回路のみに当てはまるようなアイデアや最適化より、かえってオーソドックスな構成で標準化し、汎用性、量産性を向上したり、回路の repeatability を上げ、設計

や検査の手間を省き、歩留りの向上を図ることが重要である。

デジタルフィルタの設計期間短縮のため、2.3で述べたデジタルフィルタ用CADシステム出力をそのままROM書き込み入力に使用したり、LSIマスク設計とリンクさせるなど設計から製造まで一貫して自動化を図る技術の確立も今後の重要な課題である¹²⁾。

DSPは音声のようなアナログ信号に適用されることが多く、その場合にはA/D変換およびD/A変換が必要となる。近年この分野の技術的進歩も著しく、モノリシックA/D、D/A変換器の登場が相次いでおり、コストダウンも著しい。

4.2 マイクロプロセッサによる処理

近年のマイクロプロセッサ(以下 μP と略す)の著しい進歩を背景として μP をデジタル信号処理に利用する試みが活発に行われている²⁹⁻³¹⁾。 μP をDSPに適用するねらいは主として下記の点にある。

- (1) 機能をソフトウェア化またはファームウェア化して処理の融通性を高めたり、ハードウェアの共通化、標準化を図る。
- (2) 低価格化の著しい μP を用いることによって装置の経済化を図る。
- (3) μP を適用することによって装置の開発期間を短縮し、設計コストを低減する。

μP をリアルタイムDSPに適用するとき最大の問題は μP でどの程度までの機能が処理できるかにある。今、処理しようとする信号の帯域を W 、処理に必要な各命令の平均ステップ数を S 、1ステップ当りの命令実行時間を τ とすると、リアルタイム性を維持するためには平均して、次の I_{∞} 個の命令しか処理できない。

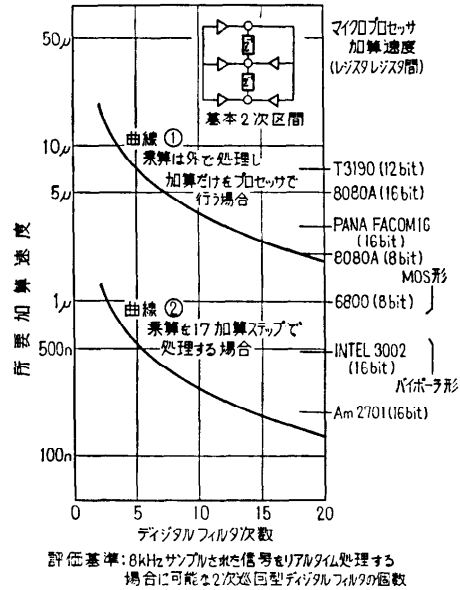
$$I_{\infty} = 1/2 W \tau S \quad (23)$$

I_{∞} が小さい程簡単な処理しか実行できないから、 I_{∞} を大きくする必要がある。それには(23)式より、

- (1) τ の小さな μP を用いる。
- (2) 命令に要するステップ数 S を小さくする。
- (3) W の小さな低周波帯域の信号を対象とする。

現在発売されている代表的な μP についてどの程度DSP処理能力があるかを評価すると図-7のごとくなる。図より、 μP にすべての処理を実行させる場合にはMOS型8ビット μP では不十分であり、 $\tau=100$ ns程度のバイポーラビットスライス μP が対象となる。

DSPに必要な命令のうち最も S の大きい命令は乗



算命令であり、これが μP の適用を制限している大きな要因である。 μP による乗算は通常Shift and Accumulationで行われるが、(加算+シフト)を1ステップで実行できれば $N \times N$ ビット乗算が $(N+1)$ ステップで処理できる(データのSettingに1ステップ)。Am 2901はパイプラインレジスタを利用して16ビット乗算を17ステップででき、カタログ値で $3.3 \mu s$ 、実験値で最大 $1.74 \mu s$ の乗算速度が得られている。

DSPでは $A_1x_1 + A_2x_2$ 型の演算がひんぱんに実行される。今、 $A_1 = \sum_{i=0}^N a_{1i}2^i$ 、 $A_2 = \sum_{i=0}^N a_{2i}2^i$ 、 a_{1i} 、 $a_{2i} \in [0, 1]$ と表わすとき、 $A_1x_1 + A_2x_2$ を、

$$P = \sum_{i=0}^N (a_{1i}x_1 + a_{2i}x_2)2^i = \sum_{i=0}^{N+2} (f_i x_1 + f_i x_2) c_i 2^i \quad (24)$$

ここに、 $f_i \in [0, 1]$ 、 $c_i \in [0, \pm 1]$ と変形して演算する方法が提案されている³²⁾。(24)式は次の漸化式より求められる。

$$P_0 = 0, P = P_{N+2}, P_i = P_{i-1} \times 2 + c_i (f_i x_1 + f_i x_2) \quad (25)$$

P_{i-1} から P_i を得るには P_{i-1} のシフトと x_1 または x_2 のいずれかの加減算で実行できるので、シフトと加減算が同時に処理できれば、 $A_1x_1 + A_2x_2$ の乗算2回と加算1回が $(N+2)$ ステップで処理できる³²⁾。

以上に述べたように、 τ の小さな高速 μP を用い、命令ステップ数 S の低減化を図ってもなお I_{∞} が

不足している場合には、次の2つの方法が考えられる。

- (1) マルチプロセッサ化する。
- (2) ステップ数を要する命令をハードウェア化する。

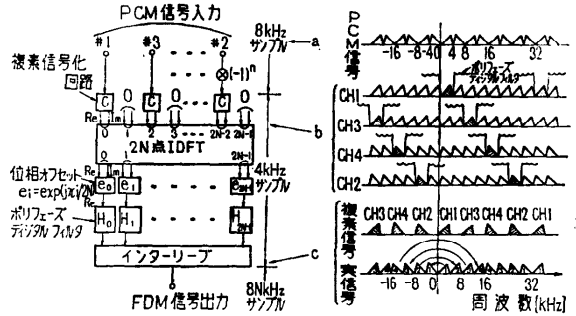
最近マルチマイクロプロセッサ構成により性能向上を図る試みが活発化しているが³³⁾、プロセッサ間の結合方式、各プロセッサの機能分担、メモリ上のプログラム配置、割込み制御などについて設計手法が確立されているとはいえ、今後の研究に待つところが多い。特にプロセッサ数が増えたと各プロセッサ間のデータ転送の割合が増加してプロセッサの効率が低下してプロセッサ数に見合っていない問題がある。

一方、最も多くのプログラムステップを要する乗算をハードウェア化して外付けすると飛躍的に性能を向上できる。市販の16×16ビット1チップ乗算器の乗算速度は160ns程度であるから、1~2ステップで乗算が可能となり $I_{0.1}$ は著しく向上する。図-7に乗算器外付けの場合の μP の処理能力を示す。8080等のMOS形 μP でも10次以上のIIRデジタルフィルタのリアルタイム処理(8kHzサンプル)が可能である。DSPでは8ビット語長は通常不十分であり、今後主流となる16ビット μP と乗算器の組合せは検討に値しよう。

5. デジタル信号処理の応用例

通信の分野でリアルタイムデジタル信号処理を適用した装置が実用済か実用化に近い例としては、(i)MODEM(データおよびFAX用)、(ii)トランスマルチプレクサ³⁴⁾(Transmultiplexer, TMUXと略す)、(iii)PB・MF(プッシュボタン信号および多周波信号)受信器、(iv)エコーサプレッサおよびエコーキャンセラ、(v)音声分析合成装置、(vi)画像帯域圧縮符号化装置、(vii)測定器(スペクトル分析器その他)などがある。これら各装置において前述したデジタルフィルタ、FFTおよびその他の諸変換が組み合わせられ、各装置に必要な機能はデジタル演算によって実現されている。演算を実行する金物は各々の目的に合致したハードウェアの専用プロセッサが多いが、今後はマイクロプロセッサによる構成が検討されよう。ここではリアルタイムDSP技術を全面的に適用した装置の例としてTMUXを紹介する。

TMUXは時分割多重化(TDM)信号と周波数分割



(a) TDM-FDM 変換アルゴリズム
(b) $N=4$ の場合のスペクトル
図-8 TDM-FDM 変換デジタル信号処理過程の一例

多重化(FDM)信号の双方向変換をDSP技術を用いてデジタル的に行う装置である。時分割多重化された4kHz帯域の電話信号をSSB(Single Sideband)変調して周波数軸上に並べる機能はデジタルフィルタとFFTの組み合わせで実現される³⁵⁾。処理の流れと各処理段階でのスペクトルを図式的に図-8に示す。この処理を z 変換表示で示すと次式で与えられる³⁶⁾。

$$Y(z) = \sum_{i=0}^{N-1} z^{-i} H_i(-z^N) \quad \text{ポリフェーズ DF 演算}$$

$$\sum_{K=0}^{N-1} \exp\left[j2\pi\left(K + \frac{1}{2}\right)i/N\right] X_K(z^N) \quad \text{FFT 演算} \quad (26)$$

ここに z は f_i に対する z 変換、 f_i はSSB-FDM信号のサンプリング速度、 $N=f_i/[kHz]/4[kHz]$

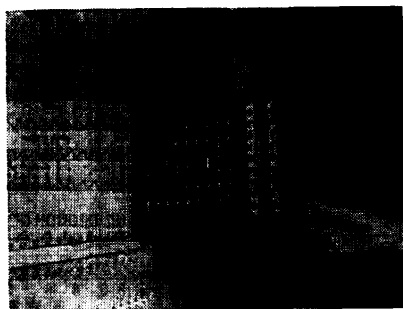
$H_i(z^N)$ は通過域0~1.7kHz、阻止域2.3kHz~ $f_i/2$ の低域フィルタ $H(x)$ を、

$$H(x) = \sum_{i=0}^{N-1} H_i(z^N) z^{-i} \quad (27)$$

により N 分解したフィルタであり、全体でポリフェーズデジタルフィルタ(P-DF)と呼ばれる³⁵⁾。P-DFとFFTとは一体として4kHz間隔で周波数軸上に並んだフィルタバンクを形成する。 $X_K(z^N)$ のスペクトルは正負対称ではなく、したがって時系列 $x_K(nNT)$ は複素信号であり入力PCM信号に、ウィーバ変調器³⁵⁾に相当する演算を施して得られる。出力 $Y(z)$ に対する時系列 $y(nT)$ も同様に複素信号であり、その実数部をとる。アルゴリズムの詳細は文献35, 36)を参照されたい。

図-8の方法の他数多くのアルゴリズムが提案されている³⁷⁾。

筆者らは $N=144$ 、 $f_i=576kHz$ のTMUXを試



2次巡回型デジタルフィルタが2個とう載されている。4kHz サンプルング 72 多重クロック 6.912MHz データ語長 24 ビット係数語長 16 ビット

図-9 SG 帯域トランスマルチプレクサに用いるデジタルフィルタ

作した³⁸⁾。本装置は 8.192Mb/s の TDM 信号 (120 CH の PCM 信号) を 2 つの超群帯域 (312~552kHz) の FDM 信号に変換および逆変換する。図-9 は本装置のデジタルフィルタ部であり、プリント板 1 枚に 2 つの 2 次巡回型フィルタがとう載される。本デジタルフィルタは 1 チャンネル当り 4kHz サンプルング速度で動作し、72 チャンネル分多重処理する。本デジタルフィルタには通研で開発した CMOS 16 ビットパイプライン乗算器³⁹⁾が用いられている。このようにデジタルフィルタと FFT はコストパフォーマンスに優れている直列演算のハードワイヤードプロセッサで構成される。

6. むすび

デジタル信号処理発展の歴史は、比較的短期であるにもかかわらず、これまでに蓄積された技術体系は膨大を極めており、また最近の進歩は特に目覚ましいものがある。本文では通信システムへの適用を指向したごく狭い分野の、しかも最近の動向を中心に述べたにすぎず、特に写真やパターンの処理など DSP を駆使する非リアルタイム処理については全くふれなかった。非リアルタイム処理においては計算機によりアナログ処理では実現困難な極めて複雑な演算処理が施こされており、その処理過程には興味深いものがあるが、著者の浅学菲才の点もあり割愛させていただいた。

DSP の本質は冒頭で述べたごとく機能の実現手段にあり、原理的に非リアルタイム処理・リアルタイム処理を問わず極めて広範な分野への適用が可能なものである。本文で主として述べたリアルタイム処理を必要とする分野においては、LSI 技術の進歩と相まって、今後その技術は深く浸透していくものと期待されてお

り、特にとり扱う信号の周波数帯域が低い領域においては、実現手段の一つのオルタネティブとして一度は DSP の適用を検討してみることをお奨めして本文のしめくりとしたい。

参考文献

- 1) A. Oppenheim & R. Schaffer, Digital Signal Processing, Prentice-Hall, N. J. (1975).
- 2) L. Rabiner & B. Gold, Theory and Application of Digital Signal Processing, Prentice-Hall, N. J. (1975).
- 3) R. Mersereau & D. Dudgeon, "Two-Dimensional Digital Filtering", IEEE Proc. Vol. 63, No. 4, pp. 610~623 (April 1975).
- 4) B. Hosticka, R. Brodersen & P. Gray, "MOS Sampled Data Recursive Filters Using Switched Capacitor Integrators", IEEE Journal SC, Vol. 12, No. 6, pp. 600~608 (Dec. 1977).
- 5) 土屋, 志田, 伏見, "阻止域準チェビシェフ最小位相推移 FIR フィルタ", 電子通信学会技術研究報告, CST 78-16, 1978 年 5 月。
- 6) 上田, 青山, "最適振幅特性を有する最小位相推移 FIR フィルタの一設計法", 電子通信学会, 通信方式研究会, 1978 年 9 月報告予定。
- 7) D. Harris & R. Mersereau, "A Comparison of Algorithms for Minimax Design of Two-Dimensional Linear Phase FIR Digital Filters", IEEE Trans. ASSP, Vol. 25, No. 6, pp. 492~500 (Dec. 1977).
- 8) 例えば, 小野, 青山, "状態変数法によるデジタルフィルタの構成", 昭和 52 年度電子通信学会総合全国大会, S2-2, 1977 年 10 月。
- 9) S. Belter & S. Bas, "Computer-Aided Analysis and Design of Digital Filters with Arbitrary Topology", IEEE Trans. CAS, Vol. 22, No. 10, pp. 810~819 (Oct. 1975).
- 10) 小野, 青山, "デジタルフィルタ CAD システムのアルゴリズムの基礎検討", 電子通信学会技術研究報告, CST 77-30, 1977 年 6 月。
- 11) 小野, 青山, "状態方程式を用いたデジタルフィルタの周波数域解析", 電子通信学会技術研究報告, CS 77-184, 1978 年 2 月。
- 12) 青山, 小野, 笹島, 進藤, 大村, "デジタルフィルタおよびサンプル値フィルタの設計・解析プログラム-USAP-D-", 電子通信学会技術研究報告, CS 77-185, 1978 年 2 月。
- 13) 小野, 青山, "クロスソストを用いたデジタルフィルタのエミュレーション" 電子通信学会通信方式研究会, 1978 年 9 月発表予定。
- 14) L. Jackson, "Roundoff-Noise Analysis for Fixed-Point Digital Filters Realized in Cascade or Parallel Form", IEEE Trans. AU, Vol. 18, No. 2, pp. 107~122 (June 1970).

- 15) G. Szentirmai, "FILSYN—A General Purpose Filter Synthesis Program", IEEE Proc. Vol. 65, No. 10, pp. 1443~1458 (Oct. 1977).
- 16) 日比野, 中山, 水上, "デジタルフィルタ設計プログラムシステム: DINETS", 昭和 53 年度電子通信学会総合全国大会, No. 1592, 1978 年 3 月.
- 17) H. Silverman, "An Intraduction to Programming the Winograd Fourier Transform Algorithm (WFTA)", IEEE Trans. ASSP, Vol. 25, No. 2, pp. 152~165 (April 1977).
- 18) R. Agarwal & C. Burrus, "Number Theoretic Transforms to Implement Fast Digital Convolution", IEEE Proc. Vol. 63, No. 4, pp. 550~560 (April 1975).
- 19) J. McClellan, "Hardware Realization of a Fermat Number Transform", IEEE Trans. ASSP, Vol. 24, No. 3, pp. 216~225 (June 1976).
- 20) 青山, 真野, 若林, 吉田, "デジタル信号処理用 LSI の諸条件", 電子通信学会技術研究報告, CS 76-68, 1976 年 8 月.
- 21) 宮川ほか, デジタル信号処理, 電子通信学会編, 1975 年.
- 22) 大和田ほか, "可変語長シフトレジスタ CMOS LSI", 電子通信学会通信方式研究会, 1978 年 9 月発表予定.
- 23) 例えば, 菅田ほか, "デジタルフィルタ LSI", 電子通信学会昭和 53 年度総合全国大会, No. 1853, 1978 年 3 月.
- 24) 例えば, 村野ほか, "LSI プロセッサによるデータ MODEM 用自動等化器", 電子通信学会技術研究報告, CS 76-186, 1977 年 2 月.
- 25) J. Buie & T. Zimmerman, "Very Large Scale Integrated Circuits for Digital Signal Processing", Circuits and Systems, pp. 2~7 (April 1977).
- 26) 大和田ほか, "線形演算プロセッサ CMOS LSI", 電子通信学会通信方式研究会 1978 年 9 月発表予定.
- 27) A. Peled & B. Liu, "A New Hardware Realization of Digital Filters", IEEE Trans. ASSP, Vol. 22, No. 6, pp. 456~462 (Dec. 1974).
- 28) 後藤, "乗算器のないデジタルフィルタについて", 電子通信学会, 技術研究報告 CS 75-106, 1975 年 9 月.
- 29) 三木ほか, "ビットスライスプロセッサによる実時間変係数デジタルフィルタ", 電子通信学会技術研究報告, CST 77-25, 1977 年.
- 30) 西谷, "実時間信号処理マイクロプロセッサシステムについて", 電子通信学会技術研究報告, CS 77-138, 1977 年.
- 31) 入江, 鶴沢, "マイクロプロセッサによるデジタルフィルタの構成", 昭和 52 年度電子通信学会情報部門全国大会, No. 135, 1977 年.
- 32) 西谷, "2 項乗算手法について", 昭和 53 年度電子通信学会総合全国大会 No. 1593, 1978 年.
- 33) 例えば, 星子他, "マルチマイクロプロセッサによる信号処理", 昭和 52 年度電子通信学会情報部門全国大会, No. S4-8, 1977 年.
- 34) 青山, "FDM-TDM トランスマルチプレクサの検討", 電子通信学会技術研究報告 CS 75-166, 1976 年 1 月.
- 35) M. Bellanger & J. Daguët, "TDM-FDM Transmultiplexer: Digital Polyphase and FFT", IEEE Trans. ASSP, Vol. 22, No. 9, pp. 199~1205 (Sept. 1974).
- 36) 青山, 真野, 若林, "TDM-FDM 変換用デジタル信号処理過程の検討", 電子通信学会技術研究報告 CS 76-193, 1977 年 2 月.
- 37) 例えば, 丸田, 金政, 友沢, "デジタル TDM-FDM 変換装置 (トランスマルチプレクサ) の方式構成に関する検討", 電子通信学会技術研究報告 CS 75-190, 1976 年 2 月.
- 38) 真野, 青山, 若林, "SG 帯 TDM-FDM 変換装置の設計", 電子通信学会通信方式研究会, 1978 年 9 月発表予定.
- 39) 大和田ほか, "CMOS パイプライン乗算器の特性", 電子通信学会技術研究報告 SSD 77-91, 1978 年 3 月.
- 40) 鎌田, 三谷, "デジタルフィルタの構成方法", 昭和 52 年電気四学会連合大会. No. 161, 1977.
- 41) A. Peled & B. Liu, Digital Signal Processing, John Wiley & Sons, New York, 1976.

(昭和 53 年 7 月 25 日受付)

(昭和 53 年 8 月 21 日再受付)