

3 バンク RAM を用いた高速フーリエ変換回路の実装

前原 崇章[†] 和田 知久[†] 金田 喜共[†]
村上 修二[‡] 水谷 弘幸[‡]

[†] 琉球大学理工学研究科 〒903-0129 沖縄県中頭郡西原町千原1番地
[‡] 株式会社マグナデザインネット 〒901-0152 沖縄県那覇市小禄1831番地1
E-mail: [†] {maehara,kaneda}@lsi.ie.u-ryukyu.ac.jp, [†] wada@ie.u-ryukyu.ac.jp,
[‡] {murakami,mizutani}@magnadesignnet.com

あらまし 日本や欧州の地上波デジタルテレビジョン放送方式として採用された OFDM(直交周波数分割多重)は、多数の直交する搬送波の信号を多重化するデジタル変調方式である。この OFDM 信号の変復調には 4096~8192 ポイントもの高速フーリエ変換を行うため、変復調器ハードウェア規模の増大という問題がある。本稿では地上波デジタル放送対応の高速フーリエ変換回路において RAM を効率的に利用することで回路規模の縮小を実現したので、その詳細について報告する。
キーワード FFT, Radix, Bit Reverse, RAM, ISDB-T, OFDM

FFT (Fast Fourier Transform) Circuit design using 3 bank RAM organization

Takafumi MAEHARA[†] Tomohisa WADA[†] Yoshitomo KANEDA[†]
Shuji MURAKAMI[‡] and Hiroyuki MIZUTANI[‡]

[†] Graduate School of Science and Engineering, University of the Ryukyus
1 Senbaru, Nishihara, Nakagami, Okinawa, 903-0219 Japan
[‡] Magna Design Net, Inc. 1831-1 Oroku, Naha, Okinawa, 901-0152 Japan
E-mail: [†] {maehara,kaneda}@lsi.ie.u-ryukyu.ac.jp, [†] wada@ie.u-ryukyu.ac.jp,
[‡] {murakami,mizutani}@magnadesignnet.com

Abstract The OFDM (Orthogonal Frequency Division Multiplexing) was adapted as a digital modulation method of the Digital Terrestrial Television broadcasting system in Japan and Europe since the OFDM can effectively pack a large number of digital data into the limited RF bandwidth by using thousands of orthogonal sub-carrier signals. Since it modulates thousands of data simultaneously, the increase of encoding/decoding hardware is one of the problems in implementing the OFDM system. One of the big calculations in the OFDM is FFT (Fast Fourier transform). In the Digital Terrestrial Television broadcasting system, the size of 4096 - 8192 points FFT is required. In this paper, the reduction of memory size and the number of multipliers are examined for the OFDM system. The proposed 3 bank RAM organization with pipelined operation has reduced the RAM size by 19% and decreased the number of multipliers from 6 to 4.

Keyword FFT, Radix, Bit Reverse, RAM, ISDB-T, OFDM

1. はじめに

OFDM(Orthogonal Frequency Division Multiplexing)は日本や欧州の地上波デジタルテレビジョン放送方式として採用された変調方式であり、広く注目を集めている。OFDMはその名前が示す通り、多数の直交する搬送波の信号を多重化するデジタル変調方式である。このように多数の搬送波からなる OFDM 変調方式では、その変調及び復調に高速フーリエ変換(FFT: Fast Fourier Transform)を用いて処理を行う。この OFDM 信

号の変復調には 4096~8192 ポイントもの FFT 処理を行うため、変復調器ハードウェア規模の増大という問題がある。

本研究では、Radix-4 バタフライ演算器と 3 バンク構成の RAM を用いた FFT 処理により、RAM や演算器を共有することでハードウェア規模の縮小を検討する。

2. OFDM 変調方式

OFDM 信号を生成する直接的な方法としては N 個のデジタル変調器を用いて個々の基本信号を生成し、そ

れらを加え合わせる事が考えられる。しかし、これを実際に行う場合、搬送波の数が非常に多くなると以下のような問題が生じる。

- ・非常に数多くのデジタル変調器を用いる必要があり、ハードウェアの規模が増大してしまう。特に、数千の搬送波を用いるデジタル地上波テレビジョンにおいては大きな問題となる。
- ・仮に多数のデジタル変調器を用意できるとしても、OFDM 信号を正しく生成するためには、周波数およびタイミングの正確に設定された多数の搬送波を生成する必要がある。しかし、実際にこれを正確に行うためのハードウェアを作成するためには調整の難しさ等の問題がある。

同様の問題は受信機側でも生じ、上記の問題は受信機の小型化に大きく影響する。この問題によって、OFDM はその実現に制約をうけることになっていた。しかし、離散フーリエ変換(DFT: Discrete Fourier Transform)を用いて OFDM 信号を生成することで、OFDM 方式の実現がより現実的なものとなった。

OFDM 変調器の具体的な構成例を図 1 に示す[1]。

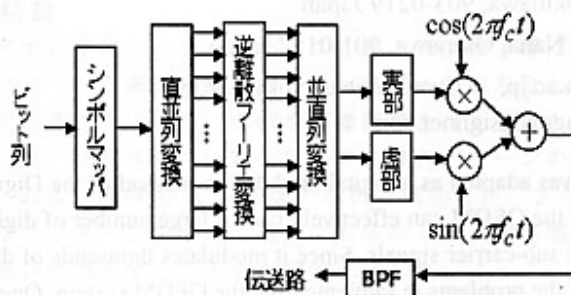


図 1: OFDM 変調器の構成例

図 1 の変調器において、送信すべきデータビット列は最初にシンボルマップによって、各搬送波を変調するための複素シンボル列に変換される。シンボルマップは複数ビットのデータを対応する複素シンボルに変換するためのものであり、シンボルの変調方式に応じた構成をとる。生成された複素シンボル列は直並列変換器に蓄積される。蓄積された N 個のシンボルは逆離散フーリエ変換によって変換され、OFDM シンボルの標本値が生成される。得られた標本値は並直列変換の後、連続信号に変換され、複素ベースバンド OFDM 信号の実部と虚部それぞれに対して搬送波が掛け合わされ、搬送帯域の信号が生成されて、伝送路に送信され

る。図 1 の変調器の構成は主要な構成部分のみを示す簡略化構成であり、実際の変調器はそのほか多くの構成要素から成っている。

3. 高速フーリエ変換

FFT は DFT における計算のむだを省くため、三角関数の周期性をうまく利用した計算技法である。FFT であっても DFT であっても計算結果には差がない(厳密には、FFT での演算回数のほうが少ないのでいくぶん計算精度がよい)。DFT は、すべてのスペクトル $X(k)[k=0,1,2,\dots,N-1]$ を求めるためには N^2 回の積和演算が必要となる。例えば、 $N=1024$ の場合は、約 10^6 回もの膨大な積和演算が必要となる。高速フーリエ変換アルゴリズムを用いた場合、その計算回数は $N \log_2 N$ 回となり、 $N=1024$ とすると計算回数は約 10^4 回となり、大幅に計算回数を減らすことができる。

ただし、FFT は普通、データ点数に制約があり、2 のべき乗 (2,4,8,16,32,64,128,...) にデータ数をとるのが最も一般的である。このような制約がありながらも、実用上は計算速度の点でのメリットが大きいので、必然的に FFT が用いられる。

3.1. バタフライ回路

バタフライ回路とは、FFT の最小構成単位であり、基数を 2 (Radix-2) または 4 (Radix-4) とすることで、一度に処理するデータ数が決まる。以下に、基数 2 および基数 4 に対する 2 種類のバタフライ回路を説明する。

3.1.1. Radix-2 バタフライ

Radix-2 バタフライでは、2 入力 2 出力の計算が行われる。式(1)が radix-2 バタフライ演算を表し、これは DFT の式から導き出せる。図 2 は、Radix-2 バタフライの構成図である[2]。+ で示される加算器のしたにある係数は、2 番目の入力値 $x(1)$ に対して乗算される。

$$X(k) = \sum_{n=0}^{N-1} \left\{ x(n) + (-1)^k x\left(n + \frac{N}{2}\right) \right\} W_N^{nk} \quad (1)$$

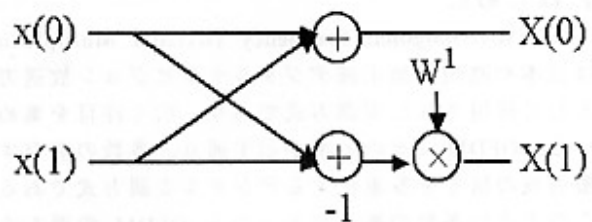


図 2: Radix-2 バタフライ

図3に、Radix-2を用いた8点FFTのシグナルフロー図を示す。 $\log_2 8=3$ より、Radix-2を用いた8点FFTでは3段の直列接続で構成される。各ステージは4つのRadix-2バタフライ回路で構成されている。

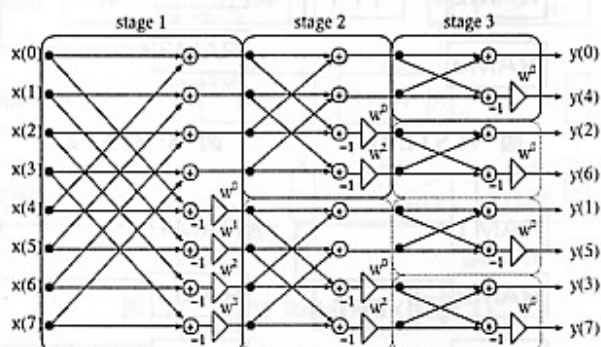


図3: Radix-2 バタフライを用いた8点FFT

3.1.2. Radix-4 バタフライ

Radix-4 バタフライは、Radix-2 バタフライを拡張したもので、2入力での計算を行うRadix-2バタフライに対してRadix-4バタフライは4入力での計算を行う。式(2)がradix-4バタフライ演算を表すものである。Radix-2同様、DFTの式から導き出すことができる。図4は、Radix-4バタフライの構成図である[3]。+で示されるのは加算器であり、その上の4つの係数が入力値のそれぞれに対して乗算される。 W^1, W^2, W^3 はtwiddle factorと呼ばれ、回転に対応する複素数である。

$$X(k) = \sum_{n=0}^{N/4} \left\{ x(n) + (-j)^k x\left(n + \frac{1}{4}N\right) + (-1)^k x\left(n + \frac{2}{4}N\right) + (j)^k x\left(n + \frac{3}{4}N\right) \right\} W_N^{nk} \quad (2)$$

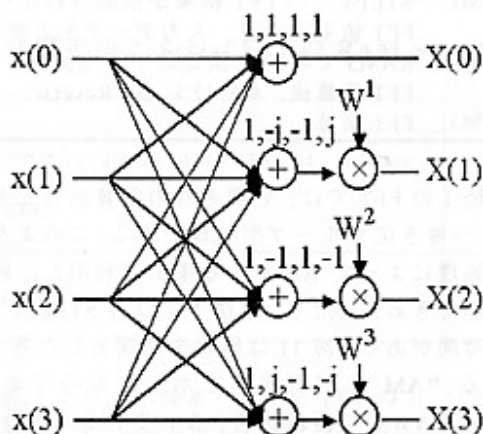


図4: Radix-4 バタフライ

図5に、Radix-4を用いた16点FFTのシグナルフロー図を示す。 $\log_4 16=2$ より、Radix-4を用いた16点FFTでは2段の直列接続で構成される。各ステージは4つのRadix-4バタフライ回路で構成されている。

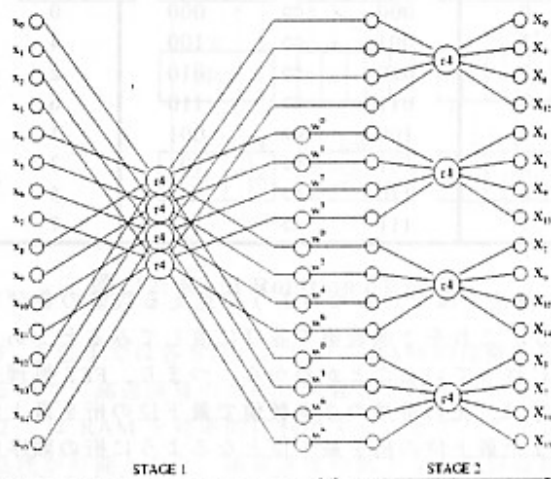


図5: Radix-4 バタフライを用いた16点FFT

Radix-2におけるステージ数は、 $\log_2 N$ 回(シグナルフロー図の段数に相当)なのに対してRadix-4でのステージ数は、 $\log_4 N$ となる。このことから、Radix-2よりもステージ数が少ないのでハードウェアでの実装の際に規模の縮小が期待できる。しかし、データ入力数が4のべき乗(4, 16, 64, 256, 1024, ...)となるのでradix-2のときよりもデータ点数の制約が大きくなる。例えば、データ入力数として $N=2048$ とすると、 N は2のべき乗であるのでradix-2においては問題ないがradix-4で扱う際には $N=4096$ のデータ点数として処理しなければならない。そこで、ステージ数を抑えつつデータ点数の制約を緩やかにする手段として、Radix-2/4を組み合わせてデータ処理を行う。

今回対象とするFFTのポイント数は、 $8k/4k$ なので $8k$ についてはRadix-2/4の混在、 $4k$ はRadix-4で計算を行う。ここでRadix-2による処理は、Radix-4のバタフライ演算回路を用いて実行することができるように構成する。

3.2. ビットリバーサル

FFTアルゴリズムのポイントの一つはバタフライ演算にあるが、もう一つは信号値の系列を二つのグループに次々に分解し、最終的には2点ずつのバタフライ演算に帰着するよう信号値を並べ替えることである。

以下に、4点/8点FFTでのビットリバーサルの例を示す。

表 1: 8点 FFTでのビットリバーサル

元の番号		Bit Reversal		
10進数	2進数	2進数	2進数	10進数
0	000	⇔	000	0
1	001	⇔	100	4
2	010	⇔	010	2
3	011	⇔	110	6
4	100	⇔	001	1
5	101	⇔	101	3
6	110	⇔	011	5
7	111	⇔	111	7

10進法ではでたらめのように見える数値の並びであるが、これを2進数値の並びに直してみると、ある規則に従っていることがわかる。つまり、FFT処理後の番号は、元の番号の2進数値で最下位の桁を最上位に、また最上位の桁を最下位となるように桁の順序関係を逆に並べ替えたものに等しい。FFT出力を正規の順序に戻すためにビットリバーサル処理が最終的に必要になる。

4. 3バンク RAM 構成 FFT

FFTはその処理を大きく分けて、データ入出力、バタフライ演算、Bit Reverseと3つの処理から成る。これら3つの処理時にアクセスする各々のRAMに、データ入出力RAM、演算RAM、Bit Reverse RAMの処理を割り当てる。各々割り当てられた処理は、OFDM信号1シンボル毎にその処理をローテーションすることでFFT処理を行う。図6に、そのローテーション方法を図示する。

図6のローテーションを実現する手法について以下に説明する。

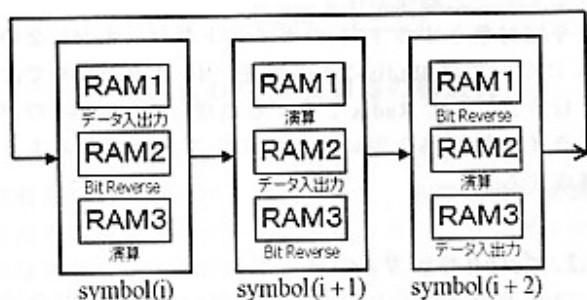


図 6: RAM Rotation

4.1. 手法 1

提案手法を図7~10と表1で説明する。図や表においてSTEP1~4の後、STEP2~4を繰り返す。

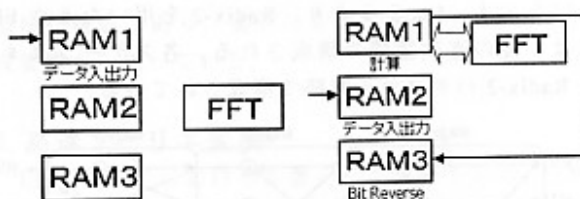


図 7: STEP1

図 8: STEP2

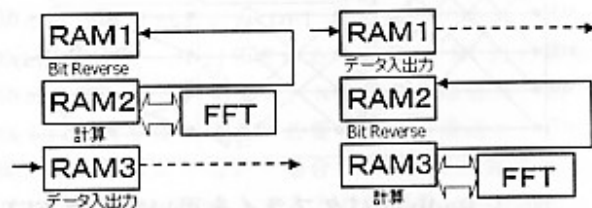


図 9: STEP3

図 10: STEP4

表 2: 手法 1 の説明

○STEP1	RAM1: データ入力 RAM2: 待機 RAM3: 待機
○STEP2	RAM1: FFT 演算 RAM2: データ入力 (2回目以降はFFT値を出力後に上書き) RAM3: RAM1でのFFT演算終了まで待機 FFT演算後、RAM3にBit Reverse
○STEP3	RAM1: RAM2でのFFT演算終了まで待機 FFT演算後、RAM1にBit Reverse RAM2: FFT 演算 RAM3: STEP2でのFFT結果が格納されているので FFT値を出力後、入力データを上書きする
○STEP4	RAM1: STEP3でのFFT結果が格納されているので FFT値を出力後、入力データを上書きする RAM2: RAM3でのFFT演算終了まで待機 FFT演算後、RAM2にBit Reverse RAM3: FFT 演算

手法1のFFTでは、処理過程の演算出力先を同一のRAMへ書き戻すループ型処理である。このようにループ型処理によってRAMを効率的に利用し、回路規模が縮小できる。ただし、手法1では各STEPでRAMの待機時間がある。図11は横軸を時間とした各STEPにおけるRAMの利用状態を示したものである。Bit Reverse RAMにて待機状態となっているのがわかる。

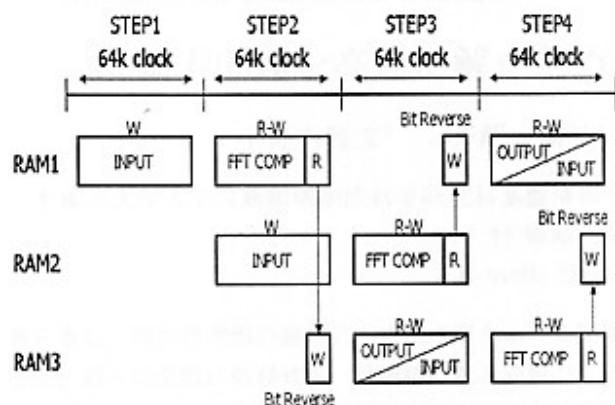


図 11: RAM Rotation (手法 1)

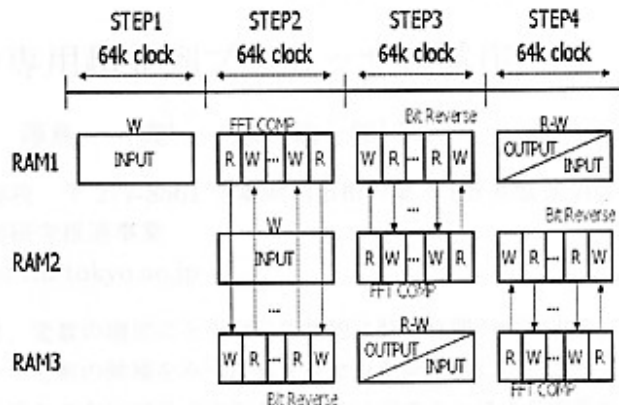


図 12: RAM Rotation (手法 2)

4.2. 手法 2

図 12 で示すように手法 2 は FFT 処理の間、待機状態である Bit Reverse RAM も演算 RAM の一部として利用するものである。その利用方法とは、手法 1 の FFT 処理において演算 RAM は Read, Write とともに計算 RAM 内で行われていたが、手法 2 では出力先を演算 RAM ではなく Bit Reverse RAM へ移す。次のステージでの FFT 処理は、Bit Reverse RAM を入力先、演算 RAM を出力先として入出力先を逆にする。以後これを繰り返す。そして、最後に演算 RAM から Bit Reverse RAM へ Bit Reverse 処理を行うことで FFT 処理の終了となる。このように Read, Write 先を別々にすることで FFT 演算をパイプライン処理化でき、手法 1 の RAM Size 縮小に加え、高速演算または、さらなる回路規模(バタフライ演算器, 乗算器)の縮小が可能となる。

5. まとめ

本研究の 3 バンク RAM を用いた FFT と Serial FFT[4]において、各々の FFT 処理に必要な RAM サイズを表 3 に示す。

表 3: 提案手法と Serial FFT の RAM Size 比較

	RAM Size	総計
Serial FFT	$8192 \times \frac{3}{2} + (4096 + 1024 + 256 + 64 + 16 + 4) \times \frac{7}{4} + 8192$	30035
提案手法 (1.2)	8192×3	24576

上記の表 3 より、本提案と Serial FFT の RAM Size を比較すると、約 19% の RAM サイズを縮小できた。そして表 4 に Serial FFT, 手法 1, 手法 2 の乗算器数を

示す。手法 1 では各 STEP において RAM の待機状態があるので、高速演算のために乗算器が増えている。手法 2 では RAM を効率的に利用することでパイプライン処理が可能となり、演算速度の低下なく乗算器数を削減できた。

表 4: Serial FFT, 手法 1, 手法 2 の乗算器数

	乗算器数
Serial FFT	6 個
手法 1	8 個
手法 2	4 個

研究会では CADENCE 社の S P W (Signal Processing Work bench) システムによる実設計結果を含めて発表を行う予定である。今後の課題として、FFT 演算 Bit 精度やバタフライ演算器数、乗算器数削減の検討を予定している。

6. 謝辞

本研究をまとめるにあたり、日頃より御指導を賜りました和田知久教授に深く感謝致します。また、同じく御支援を賜ったマグナデザインネットの方々、そして、同研究室の方々に感謝致します。

文献

- [1] 伊丹誠, "OFDM 変調技術," トリケップス, pp.43-49, Oct.1999.
- [2] 金城繁徳, 尾知博, "例題で学ぶ デジタル信号処理," コロナ社, Nov.1998.
- [3] John A. C. Bingham, "ADSL, VDSL, and Multicarrier Modulation," A Wiley-Interscience, 2000.
- [4] Christophe Del Toso, et al., "0.5-um CMOS Circuits for Demodulation and Decoding of an OFDM-Based Digital TV Signal Conforming to the European DVB-T Standard," IEEE Journal of Solid-State Circuits, Vol.33, No.11, Nov.1998.