

ポリフェーズ・フィルタ・バンクを用いた基数 2^k FFT に関して 電波望遠鏡用分光器への適用

中原 啓貴[†] 中西 裕之^{††} 笹尾 勤^{†††}

[†] 鹿児島大学 大学院 理工学研究科 電気電子工学専攻 〒 890-0065 鹿児島県鹿児島市郡元 1-21-40

^{††} 鹿児島大学 大学院 理工学研究科 物理宇宙専攻 〒 890-0065 鹿児島県鹿児島市郡元 1-21-40

^{†††} 九州工業大学 大学院 情報工学研究科 情報創成工学専攻 〒 820-8502 福岡県飯塚市大字川津 680-4

あらまし 電波望遠鏡における分光器 (spectrometer) は宇宙から受信した電波をスペクトルに変換する重要な箇所であり、広帯域かつ高速なフーリエ変換が必要とされている。本論文ではポリフェーズ・フィルタ・バンクを用いた基数 2^k FFT 回路について述べる。既存の基数 2^2 FFT 回路の解析を行い、メモリが実装のボトルネックになることを示す。提案回路は、ポリフェーズ・フィルタ・バンクを用いて荒い帯域分割を行った後、基数 2^k FFT 回路を用いてスペクトルに変換する。4 並列化した提案回路を Altera 社 FPGA ボードに実装し、サンプル点 2^{27} の FFT を 0.048 秒で行えることを示す。提案回路は、サンプル点 2^{27} FFT に関して、SETI spectrometer よりも 20.81 倍高速であった。

On a Radix 2^k FFT Using a Polyphase Filter Bank Application to a Spectrometer for a Radio Telescope

Hiroki NAKAHARA[†], Hiroyuki NAKANISHI^{††}, and Tsutomu SASAO^{†††}

[†] Faculty of Engineering, Kagoshima University, 1-21-40, Korimoto, Kagoshima 890-0065, Japan

^{††} Faculty of Science, Kagoshima University, 1-21-40, Korimoto, Kagoshima 890-0065, Japan

^{†††} Department of Computer Science and Electronics, Kyushu Institute of Technology
680-4, Kawazu, Iizuka, Fukuoka, 820-8502 Japan

Abstract In radio telescope, the spectrometer analyzes the radio frequency (RF) received from celestial objects at the frequency domain by performing the fast Fourier transform (FFT). The FFT can be realized by the radix 2^k FFT ($R2^k$ FFT). In Radio Astronomy, the number of points for the FFT is larger than that for the general purpose one. Thus, the size of memory using a conventional method is too large to implement. We use the polyphase filter bank to reduce the size of the $R2^k$ FFT. We implemented the 2^{27} point FFT by the $R2^k$ FFT with the polyphase filter bank. Compared with the SETI spectrometer for 2^{27} -FFT, the four parallelized proposed ones for 2^{27} -FFT is 20.81 times faster.

1. はじめに

1.1 電波望遠鏡における分光器 [1], [9], [18]

電波天文学 (Radio astronomy) は、天体から放射される電波を電波望遠鏡で観測し、研究を行う天文学の一分野である。1931 年、ベル研究所のカール・ジャンスキーによる銀河系からの電波 (波長 14.6m) の初観測以降、電波望遠鏡による宇宙の観測が行われており、初期宇宙に存在する生まれたての銀河、星の誕生、太陽系のような惑星系の誕生、宇宙にある物質の進化の解明が進んでいる。

図 1 に電波望遠鏡の構成を示す。天体からの電波 (RF: Radio Frequency) を受信アンテナ上のいくつかの鏡によって反射し、

受信機へ伝送する。次に、フィードホーン (Feedhorne) によって電磁波を空間モードから導波管モードに変換し、受信機へと転送する。受信機は、増幅器とミキサを用いて扱いやすい信号 (IF: Intermediate Frequency) に変換する。最後に、**分光器 (Spectrometer)** を用いて受信電圧からパワースペクトラムに変換し、測定を行う。従って、分光器の周波数帯域 (f) と分解能 (Δf) が重要なパラメータである。初期の分光器は各周波数を直接解析し、同期回路を用いて集計するフィルタ・バンク方式を採用していた。フィルタ・バンク方式では分解能が低いため、レーザー光の回折効果を利用した音響光学偏向素子による周波数解析を行う音響光学型スペクトル分析器 (AOS: Acousto-Optical Spectrometer) が提案された。AOS は圧電素

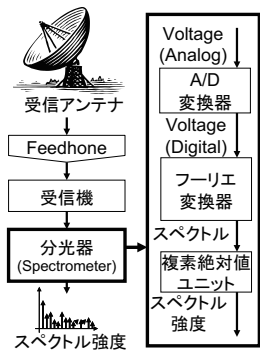


図 1 電波望遠鏡とデジタル分光器。

表 1 FFT 解析点数 N の比較。

	アプリケーション	N
一般用途 [28]	ベースバンド	2^{10} - 2^{11}
	3GPP	
	OFDM	2^8
	CT スキャナ 高音質サウンド	2^{10} - 2^{13}
電波望遠鏡	OH 輝線	2^{20}
	ゼーマン効果 SETI Spectrometer	2^{27}

子等のアナログ素子による非線形特性の調整が困難である。現在の分光器は、性能のばらつきが殆どなく調整も不要なデジタル分光器が用いられている。図 1 にデジタル分光器を示す。デジタル分光器では、まず、受信機からの中間周波数信号が A/D 変換器でサンプルされデジタル信号に変換される。次に、**フーリエ変換 (FFT)** を行い、スペクトルを得る。最後に、複素絶対値 (Magnitude of a complex number) を求め、スペクトル強度を得る。近年、高速 A/D 変換器が開発されており、高周波の信号を扱えるようになった。自己相関演算はフーリエ変換と比較して軽い処理であることから、デジタル分光器における処理の重い演算はフーリエ変換である。

1.2 電波望遠鏡における要求

広帯域解析: 近年、広帯域周波数解析の有用性が報告されている [19]。第一の有用性は、広帯域のスペクトルを得ることにより基本物理パラメータの測定範囲が広がることである。例えば、輝線観測では星間ガスに存在する色々な種類の原子、分子、電離ガスからの輝線を一度に測定することで、新しい星間ガス成分の探査、温度測定が可能となる。第二の有用性は、望遠鏡の感度が向上することである。観測データのノイズは帯域を f とすると $1/\sqrt{f}$ に比例する。従って、 f を大きくするほど感度は向上し、より暗い天体を検出することが可能となる。

表 1 に電波望遠鏡用分光器と一般用途の FFT 解析点数 N の比較を示す。フーリエ変換器を FPGA で実現した場合、既存の手法では $O(\log_2 N)$ の乗算器、LUT、及び $O(N)$ の組込みメモリが必要である [30]。表 1 より N が非常に大きい電波望遠鏡では、一般用途の帯域にチューニングした FFT では組込みメモリがボトルネックとなり、実現できない。従って、組込みメモリをコンパクトにする手法が要求される。

高速処理: 電波望遠鏡は開口合成法を用いて天体から受信した RF から電波図を作成する。開口合成法は地球の自転を利用してアンテナの位置を移動させ、高詳細な図を得る手法であり、スキャナがヘッドをずらしながら図面を得るのと同じ原理である。従って、1 データを解析する時間を短くするほど高詳細な図を得ることができる。よって、電波望遠鏡では高速な FFT が求められる。

省面積: 天体が放射する電波は大気中の水分子や酸素分子によって吸収されるため、空気が薄く乾燥した標高 4,000m~5,000m のハワイ島マウナケア山頂、チリ・アンデス山脈のアタカマ砂漠^(注1)等に電波望遠鏡が設置されている。また、短波より

(注1) : 現在、国際共同プロジェクトで計画中の ALMA (アタカマ大型ミリ波サブミリ波干渉計) 計画では、チリのアタカマ砂漠にパラボナアンテナ 80 台を設置し、2012 年から本格的な観測が始まる予定である。

波長が長い電波は電離層で反射されるため、宇宙空間上の衛星で受信する必要がある。1997 年に電波天文衛星 HALCA が打ち上げられた [8]。従って、電波望遠鏡は高原や宇宙空間に設置するため、広い敷地を必要とするスーパーコンピュータは使えない。アンテナのみを設置して保存したデータを転送するという手段 (VLBI) も考えられるが、ごく一部のデータしか観測できない。

低消費電力: 高原や宇宙空間では電源や電力送信ケーブルの敷設に限られるため、低消費電力であることが必須である。例えば、NASA の MARVEL (Mars Volcanic Emission and Life) ミッションでは消費電力を 10 W 以下にすることが要求されている。GPU は価格対性能比が FPGA よりも優れているが [11]、消費電力が数 100 W と大きい。

従って、これらの条件を満たす FPGA ボード上に分光器を実装する。

1.3 関連研究

現在、国際連携プログラム SKA (Square Kilometer Array) が集光面積 1 平方キロメートルのアンテナを使った電波望遠鏡を開発中である [23]。地球外生命体探査 (SETI: Search for Extra-Terrestrial Intelligence) のプロジェクトの 1 つである SERENDIP プロジェクト [24] では地球外生命体からの信号を解析するため、SETI Spectrometer は $f = 200$ MHz 帯域の電波を $\Delta f = 2$ Hz の分解能で 1 秒間に解析できるフーリエ変換器を用いている [20]。

Cooley と Tukey はフーリエ変換を高速に行う**高速フーリエ変換 (FFT: Fast Fourier Transform)** を提案した [5]。Duhamel らは Split-Radix に基づく FFT を提案した [6]。積和演算に適した FFT 演算に関して、Linzer らは基数 2,4,8、及び Split-Radix の FFT を [16]、Karner らは基数 2,3,5 の FFT を [13]、Goedecker は基数 2,3,4,5 の FFT を提案した [7]。Kondo らは SETI Spectrometer と同等の分光器を NVIDIA 社 Tesla S1070 (GPU4 台) [26] 上に実装した [14]。専用回路実装に関しては、He らが基数 2^2 に基づく FFT 回路を提案した [10]。また、FFT の誤差に関する解析や [15], [21], FPGA に実装した場合のハードウェア量の解析 [30]、及び、FFT の実装でボトルネックとなる回転因子のメモリを CORDIC [2], [27] や区分線形近似回路 [12], [22], [25] で実現する手法が提案されている。

1.4 提案手法と本論文の貢献点

提案手法は実時間で実行広帯域フーリエ変換を FPGA ボード上に実現する。既存の FFT 回路はメモリ量が実装のボトルネックとなるため、ポリフェーズ・フィルタ・バンクを用いて、荒い帯域分割を行った後に各帯域に対して FFT を行う。提案手法は分割した帯域に対して FFT を逐次行うため、FFT のサイズを大幅に削減可能である。本論文の貢献点は以下の通り。

従来 FFT 回路の面積を解析: 基数 2^k FFT 回路の面積を解析し、メモリ量がボトルネックになることを示した。

ポリフェーズ・フィルタ・バンクと基数 2^k FFT を組合せ、コンパクトかつ高速な広帯域 FFT を実現: 2^{27} 点 FFT を 1 台の FPGA ボード上に実装し、性能を明らかにした。筆者の知る限りでは、1 台の FPGA ボード上に 2^{27} 点もの FFT を実装した報告はない。商用の FFT ライブラリでは高々 2^{16} 点 FFT までしか実現できない [3]。ポリフェーズ・フィルタ・バンクを用いた FFT が提案されているが、商用の FFT ($R2^k$ FFT) を使っており、FFT の基数を 2^k に拡張しポリフェーズ・フィルタ・バンクと組合せたのは本報告が初めてである。

既存手法と比較を行い、高速かつ低消費電力であることを実証:

既存の電波望遠鏡用 SETI spectrometer と比較を行い、提案手法は同一帯域で 20.81 倍高速であることを示した。GPU を 4 台用いた GPU spectrometer と比較を行い、4.37 倍高速かつ約 1 桁低消費電力であることを示した。

第 2 章では基数 2^k の高速フーリエ変換器について述べ、面積の解析を行う。第 3 章ではポリフェーズ・フィルタ・バンクについて述べる。第 4 章では実験結果を示し、第 5 章で本論文のまとめを行う。

2. 基数 2^k 高速フーリエ変換器 ($R2^k FFT$)

2.1 離散フーリエ変換

$x(n), (n = 0, 1, \dots, N - 1)$ を入力信号とすると N 点離散フーリエ変換 (以降、 N -DFT と表記) は

$$X(m) = \sum_{n=0}^{N-1} x(n)W_N^{nm}, (m = 0, 1, \dots, N - 1) \quad (1)$$

で定義される。 W_N^{nm} は回転因子 (Twiddle Factor) といい

$$W_N^{nm} = e^{-j\frac{2\pi}{N}nm} = \cos\left(\frac{2\pi}{N}nm\right) + j\sin\left(\frac{2\pi}{N}nm\right) \quad (2)$$

で定義される。 j は虚数単位を表す。本論文では、入力信号 $x(n)$ は複素数のストリーム入力、 N は 2 のべき乗であると仮定する。

2.2 高速フーリエ変換

Cooley と Tukey らは乗算器を $O(N\log_2 N)$ 個に削減する N 点高速フーリエ変換 (Fast Fourier Transform: N -FFT) を提案した。本論文では回路実現が容易な時間間引き (decimation-in-time)^(注2) FFT を用いる。

N 個の入力を偶数番目と奇数番目に分けると、式 (1) は

$$X(m) = \sum_{n=0}^{N/2-1} x(2n)W_N^{2nm} + \sum_{n=0}^{N/2-1} x(2n+1)W_N^{(2n+1)m}$$

となる。ここで $W_N^{2mn} = e^{-j(2\pi/N)2mn} = e^{-j(2\pi/(N/2))mn} = W_{N/2}^{mn}$ を用いると

$$\begin{aligned} &= \sum_{n=0}^{N/2-1} x(2n)W_{N/2}^{nm} + W_N^m \sum_{n=0}^{N/2-1} x(2n+1)W_{N/2}^{(2n+1)m} \\ &= DFT_{even}(m) + W_N^m DFT_{odd}(m), \end{aligned}$$

である。ここで $m = 0, 1, \dots, \frac{N}{2} - 1$ である。式 (1) は N -DFT が 2 個の $\frac{N}{2}$ -DFT に分解できることを示している。 N -DFT に対して上式を 1-DFT になるまで $s = \log_2 N$ 回再帰的に適用することで N -FFT が得られる。このとき、 s をステージ数と定義する。 N -FFT では 1 ステージ毎に $\frac{N}{2}$ 個の乗算器が必要である。 N -FFT は s ステージで構成されるため、 $O(N\log_2 N)$ 個の乗算器が必要である。 N -DFT では乗算器数が $O(N^2)$ 個であったので、 N -FFT を用いることでフーリエ変換をコンパクトに実現できる。

2.3 基数 2^k バタフライ演算器 ($R2^k Butterfly$)

N -FFT ではバタフライ演算が基本演算単位となる。図 3 にバタフライ演算を示す。バタフライ演算器は複素加算回路 (図 4)、及び複素乗算回路 (図 5) から構成される。また、回転因子の値は回転因子メモリで保持する。図 3 (a) に示すバタフライ演算では、乗算が 2 回必要であるが、回転因子の対称性 $W_N^{m+N/2} = -W_N^m$

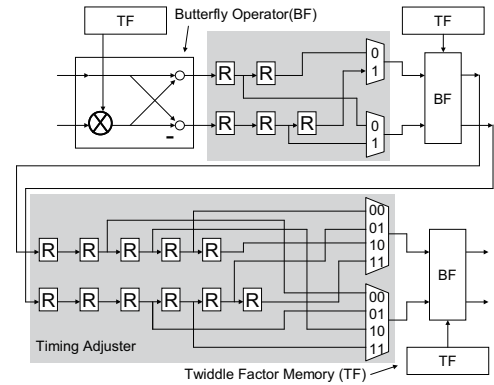


図 6 基数 2^3 バタフライ演算回路 ($R2^3 Butterfly$).

を用いると乗算を共有化できる (図 3 (b)).

入力側から出力側方向にステージのインデックスを $1, 2, \dots, \log_2 N$ とする。このとき、ステージ 1 では隣接した 2 点をバタフライ演算器に入力し、隣接した 2 点の出力を得る (図 2 (a)). ステージ 2 では 2 点離れた 2 点をバタフライ演算器に入力し、2 点離れた 2 点の出力を得る (図 2 (b)). ステージ 3 では 4 点離れた 2 点をバタフライ演算器に入力し、4 点離れた 2 点の出力を得る (図 2 (c)). すなわち、各ステージ間でタイミングを調整する調整回路が必要となる。調整回路をシフトレジスタとセレクタで構成する手法が提案されており、バタフライ演算器と組合わせて N -FFT を実現できる [10]. これを基数 2^k バタフライ演算回路 ($R2^k Butterfly$) という。

[例 2.1] 図 6 に図 2 に示した 8-FFT を実現する基数 2^3 バタフライ演算回路を示す。ただし、バタフライ演算は図 3 (b) を用いている。

基数 2^k バタフライ演算回路のハードウェア量を見積もる。乗算器数 Mul_{R2^k} ^(注3) は

$$Mul_{R2^k} = 4k = 4\log_2 N : O(\log_2 N),$$

である。加算器数 Add_{R2^k} は各バタフライ演算器の 2 個の複素加算器に 4 個、1 個の複素乗算器に 2 個必要であるから

$$Add_{R2^k} = (4 + 2)k = 6\log_2 N : O(\log_2 N),$$

である。レジスタ数 Reg_{R2^k} はステージ i 毎に $2^i + 2i - 1$ 個必要である。ただし、ステージ 1 は不要であるため

$$Reg_{R2^k} = \sum_{i=2}^k (2^i + 2i - 1) : O(2^k),$$

である。セレクタに用いる 2 マルチプレクサ数 Mux_{R2^k} はステージ i 毎に 2^i 個の信号を選択するセレクタが 2 個必要であるから

$$Mux_{R2^k} = 2 \times \sum_{i=2}^k (2^i - 1) : O(2^k),$$

である。回転因子メモリ量 $TwiddleMem_{R2^k}$ は

$$TwiddleMem_{R2^k} = Ns : O(N\log_2 N)$$

である。従って、 N -FFT を基数 2^k バタフライ演算器で実現した場合、 $O(2^k)$ 個必要なレジスタと 2 マルチプレクサが実装のボトルネックとなる。

(注3) : 本論文では FPGA の乗算器ブロック 1 個で乗算を実現するため、乗算回数と乗算器数を同一と見なして見積る。

(注2) : 周波数間引き (decimation-in-frequency)FFT も存在する。

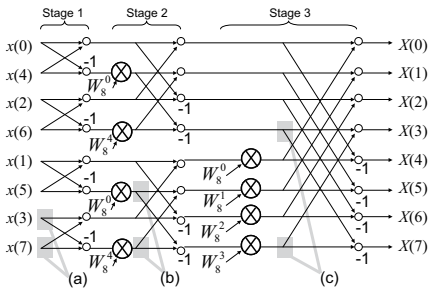


図2 8-FFTのデータフロー.

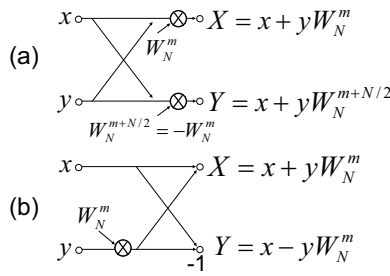


図3 バタフライ演算器.

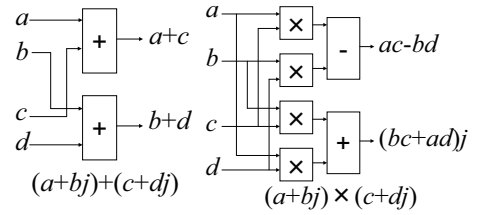


図4 複素加算器. 図5 複素乗算器.

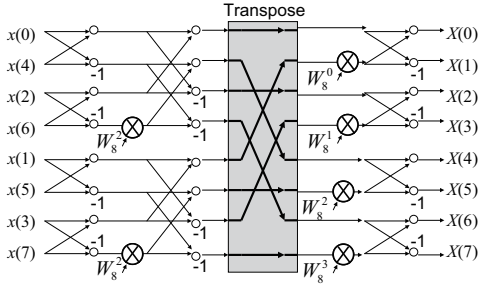


図7 転置の例.

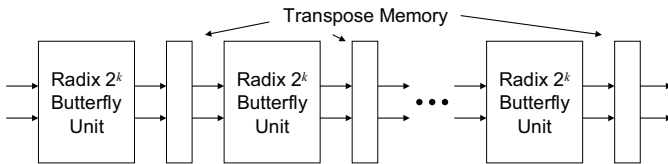


図8 基数 2^k FFT.

基数 2^k バタフライ演算回路のレジスタとセレクタが増加する原因は、ステージ毎に異なる距離の2点を計算するからである。従って、ステージ間に各点のインデックスを置き換える回路を挿入することで、隣接した2点のバタフライ演算で実現でき、調整回路を削減することができる。インデックスを置き換えることを**転置 (Transpose)**といい、**転置メモリ**で実現する。[例 2.2] 図7に図2に示したデータフローのステージ2とステージ3の間に転置メモリを挿入した例を示す。転置を行うことで、ステージ3の演算は隣接した2点のバタフライ演算で計算でき、タイミング調整回路は不要である。 ■

$s = \lceil \log_2 N \rceil$ をステージ数とする。図8に、基数 2^k バタフライ演算回路間に転置メモリを挿入した**基数 2^k FFT ($R2^k FFT$)**を示す。基数 2^k FFT 回路では、 k ステージ毎に転置メモリを配置するため、 $\frac{s}{k}$ 個の転置メモリが必要である。従って、 k を大きくすれば転置メモリ数を削減できる。

ステージ数 $s = \log_2 N$ の N -FFT は基数 2^k バタフライ演算器を $q = \frac{s}{k}$ 個用いて実現できる。基数 2^k 転置 FFT 回路の乗算器数 $Mul_{R2^k FFT}$ は

$$Mul_{R2^k FFT} = 4s = 4\log_2 N : O(\log_2 N),$$

加算器数 $Add_{R2^k FFT}$ は

$$Add_{R2^k FFT} = 6s = 6\log_2 N : O(\log_2 N),$$

レジスタ数 $Reg_{R2^k FFT}$ は

$$Reg_{R2^k FFT} = q \sum_{i=2}^k (2^i + 2i - 1) : O(2^k), \quad (3)$$

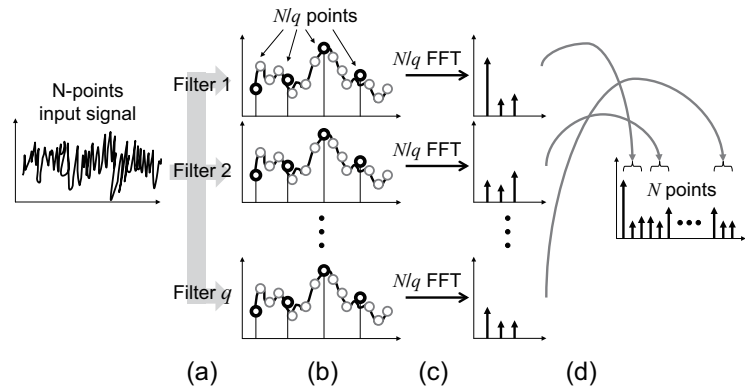


図9 フィルタ・バンクを用いた $R2^k$ FFT.

セレクタに用いる2マルチプレクサ数 $Mux_{R2^k FFT}$ は

$$Mux_{R2^k FFT} = q \sum_{i=2}^k (2^i - 1) : O(2^k), \quad (4)$$

であり、転置メモリ量 $TransMem_{R2^k FFT}$ は

$$TransMem_{R2^k FFT} = Nq : O(N) \quad (5)$$

である。 N -FFTのステージ数 s は一定であるから、転置メモリ数を削減するために k を大きくすると、式(3)、(4)が示すようにレジスタと2マルチプレクサが増加してしまう。従って、与えられたハードウェアの制約を満たす適切な k を決めなければならない。本論文では、 k を実験的に求める。回転因子メモリ量 $TwiddleMem_{R2^k}$ は

$$TwiddleMem_{R2^k FFT} = Ns : O(N\log_2 N) \quad (6)$$

である。

以上の議論より $R2^k FFT$ において、実装のボトルネックとなるのは $N \gg k$ より回転因子メモリ量である。従って、本論文ではポリフェーズ・フィルタ・バンクを用いて N そのものを削減することを提案する。

3. ポリフェーズ・フィルタ・バンクを用いた $R2^k FFT$

3.1 フィルタ・バンクを用いた $R2^k FFT$ の削減

$R2^k FFT$ の解析結果より、面積は解析点数 N に比例して増加する。従って、 N を削減することで面積を削減できる。図9にフィルタ・バンクを用いた $R2^k FFT$ を示す。周波数解析を行う帯域を Δf とし、解析点数を N とする。まず、 q 個のフィルタを用いて荒い帯域分割を行う(図9(a))。各フィルタにより、帯域は $\Delta f/q$ に分割される。次に、ダウン・サンプリングを行い、

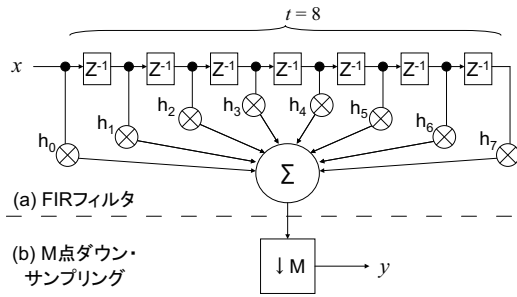


図 10 FIR フィルタを用いた間引き回路.

N 点のデータを N/q 点に削減する (図 9 (b)). フィルタリングとダウン・サンプリングを行うことを**間引き (Decimation)**という. 次に, N/q -FFT を用いてフィルタリングした帯域のみ周波数変換を行う (図 9 (c)). FFT 演算を行う前にダウン・サンプリングを行っているので周波数領域に変換された N/q 点のデータを周波数シフトし元の帯域に戻す (図 9 (d)). これらの操作を各帯域毎に行って N 点スペクトラムを構成する.

3.2 ポリフェーズ・フィルタ [17]

ここではフィルタ・バンクを構成するポリフェーズ・フィルタについて述べる. 間引き回路はフィルタとダウン・サンプリングで構成される. デジタルフィルタでは FIR (有限インパルス応答) フィルタと IIR (無限インパルス応答) フィルタの 2 種類に大別される. IIR フィルタはフィードバックループを用いるため, インパルス応答は再帰的となり無限に持続する. 一方, FIR フィルタはフィードバックループを用いないため, インパルス応答は有限となる. 従って, 同数のタップ数に関しては IIR フィルタの方が FIR フィルタよりも高精度なフィルタを構成できる. しかし, FIR フィルタは全ての周波数範囲において群遅延が完全に一定であることや安定性が高いことで, IIR フィルタよりも優れている. 電波望遠鏡では長期間の測定が行われ, 安定した出力が求められることから本論文では FIR フィルタを採用する. 図 10 (a) に FIR フィルタを示す. 時刻 t における入力を $x(t)$ とすると, FIR フィルタでは遅延要素の z 変換に相当するレジスタ z^{-1} を通過し, タップ係数を乗算した値全てを加算する. 係数の値 h_i とタップ数 t を変えることにより, 任意の周波数応答特性を実現することができる.

入力信号に対して直接ダウン・サンプリングを行った場合, 高周波成分による影響 (エイリアス) を受け, 余分な成分が含まれてしまうことがある. 従って, フィルタリングによる高周波成分の除去が必要となる. 図 10 に FIR フィルタを用いた間引き回路を示す. FIR フィルタに M 点ダウン・サンプリング回路 (図 10 (b)) を付加することで間引き回路を実現できる. M 点ダウン・サンプリング回路は M 点サンプル毎に $M-1$ 点サンプルを捨てる.

図 10 に示した間引き回路は図 11 に示すサブフィルタ回路に分割できる. 図 11 に示すサブフィルタも FIR フィルタである. ダウン・サンプリングを行ってからフィルタリングを行っても出力は変わらないので, M 点ダウン・サンプリング回路を入力側に移動する (図 12). 図 12 において, 各サブフィルタへ入力を分配している回路 (図 12 灰色部分) を回転スイッチで置き換えることで図 13 に示す**ポリフェーズ・フィルタ**が得られる. 元の FIR フィルタのタップ数を t とすると, ポリフェーズ・フィルタは M 分割したタップ数 t/M のサブフィルタを時分割実行していると言える. 従って, 図 14 に示す各サブフィルタを逐次模擬する時分割方式ポリフェーズ・フィルタ (以降, 単に

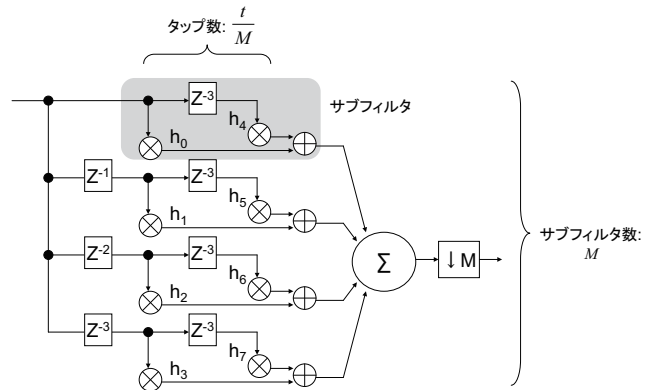


図 11 サブフィルタ分割.

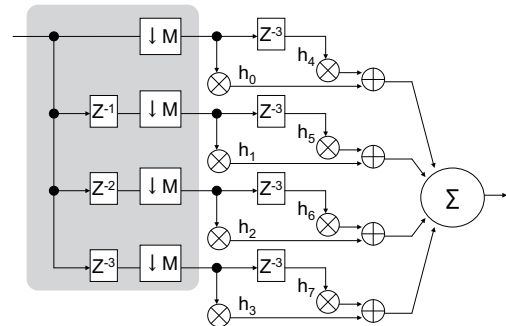


図 12 ダウン・サンプリング回路の移動.

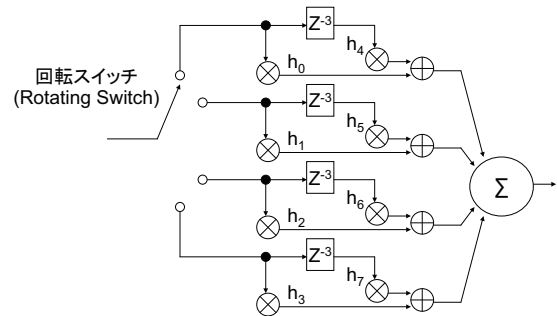


図 13 ポリフェーズ・フィルタ.

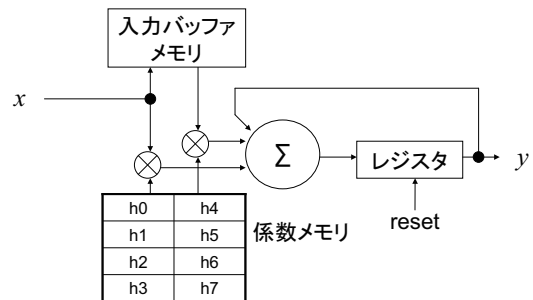


図 14 時分割方式ポリフェーズ・フィルタ.

ポリフェーズ・フィルタと表記) で実現できる. この回路では, **入力バッファメモリ**で遅延要素 z^{-1} による過去の入力を保持する. **係数メモリ**に各時刻毎の係数を格納する. そして, 乗算器と加算器を用いて各時刻毎のサブフィルタの出力を求める. レジスタに計算結果を格納する.

ポリフェーズ・フィルタのハードウェア量を見積もる. 元の FIR フィルタのタップ数を t とし M 点ダウン・サンプリングを行うとする. 乗算器数 Mul_{Poly} は

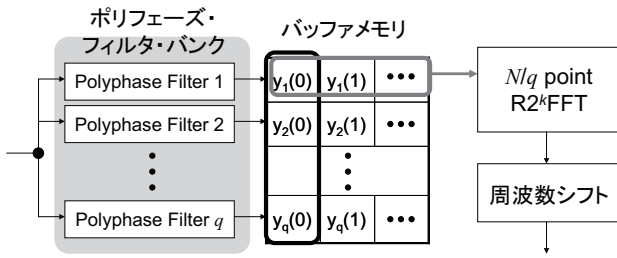


図 15 ポリフェーズ・フィルタ・バンクを用いた $R2^k$ FFT.

$$Mul_{Poly} = \frac{t}{M} \quad (7)$$

であり、加算器 Add_{Poly} は

$$Add_{Poly} = 1 \quad (8)$$

であり、レジスタ数 Reg_{Poly} は

$$Reg_{Poly} = 1 \quad (9)$$

であり、入力バッファメモリ $MemBuf_{Poly}$ は

$$MemBuf_{Poly} = t \quad (10)$$

であり、係数メモリ $MemCoef_{Poly}$ は

$$MemCoef_{Poly} = t \quad (11)$$

である。

3.3 ポリフェーズ・フィルタ・バンクを用いた $R2^k$ FFT

図 15 にポリフェーズ・フィルタ・バンクを用いた $R2^k$ FFT を示す。入力信号は q 個のポリフェーズ・フィルタからなるポリフェーズ・フィルタ・バンクにより q 個の帯域に分割される。転置メモリに各フィルタからの出力 $y_i(t)$, $i = 1, 2, \dots, q$ が各アドレス (列) に格納される。 $R2^k$ FFT は転置メモリの行に格納された各フィルタの出力を逐次読み出し、 N/q -FFT を行う。最後に、元の帯域に戻すために周波数シフトを行う。周波数シフト回路は FFT の出力をシフトするだけなので、 $R2^k$ FFT やポリフェーズ・フィルタ・バンクと比較して無視できるほど小さい。

ポリフェーズ・フィルタ・バンクのハードウェア量を見積もる。元の FIR フィルタのタップ数を t とし M 点ダウン・サンプリングを行うとする。ポリフェーズ・フィルタ数を $q = \frac{N}{M}$ とすると乗算器数 Mul_{PFB} は式 (7) より

$$Mul_{PFB} = \frac{t}{M}q : O(q) \quad (12)$$

であり、加算器 Add_{PFB} は式 (8) より

$$Add_{PFB} = q : O(q) \quad (13)$$

であり、レジスタ数 Reg_{PFB} は式 (9) より

$$Reg_{PFB} = q : O(q) \quad (14)$$

であり、入力バッファメモリ $MemBuf_{PFB}$ は式 (10) より

$$MemBuf_{PFB} = qt : O(q) \quad (15)$$

であり、係数メモリ $MemCoef_{PFB}$ は式 (11) より

$$MemCoef_{PFB} = qt : O(q) \quad (16)$$

である。従って、ポリフェーズ・フィルタ数 q に対してハード

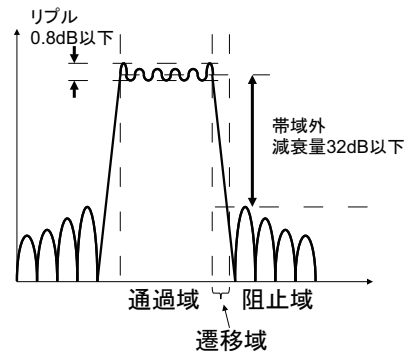


図 16 ポリフェーズ・フィルタの振幅特性.

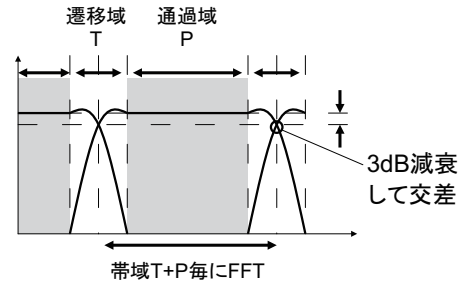


図 17 ポリフェーズ・フィルタ・バンクの振幅特性.

ウェア量は線形的に増加する。フィルタの性能を示すタップ数 t はなるべく小さくしたいがフィルタの精度に影響するので、本論文では実験的に求める。

各フィルタの出力は独立なので、複数のフィルタ毎に転置メモリと $R2^k$ FFT を用意することで容易に並列化可能である。従って、FPGA の空きリソースを利用して高速な分光器を構成できる。

4. 実験結果

4.1 実装環境

基数 2^k 転置 FFT 回路を Altera 社 DE4 Development and Education Board に実装した。搭載 FPGA は Altera 社 Stratix IV GX (ALUT 数: 424,960 個, レジスタ数: 424,960 個, M9k (9k ビット組込みメモリ) 数: 1,280 個, 18×18 DSP 数: 1,024 個) であり、DDR2SO-DIMM コネクタを 2 個, SSRAM を 1 個, PCIeExpress (x8) コネクタを 1 個, 及び A/D 変換器と接続するための HSMC コネクタを 2 個持つ。FPGA 合成ツールは Altera 社 Quartus II v.11.0 を用いた。また、DDR2SDRAM コントローラは Altera 社の MegaCore Function を用いて生成した。

式 (3), (4) より、レジスタ数と MUX 数は $O(2^k)$ で増加する。各ステージの基数 2^k がなるべく等しくなるように、 k を求めた。実装に用いた FPGA ボードは DDR2SO-DIMM を 2 個実装可能なので、3 つの基数 2^k バタフライ演算回路間に転置メモリとして用いた。最終段の転置メモリは実装していない。従って、 $k = \lceil \frac{\log_2 N}{3} \rceil$ とした。

4.2 ポリフェーズ・フィルタ・バンクの設計

ポリフェーズ・フィルタの元となる FIR フィルタの係数とタップ数を実験的に求めた。図 16 にポリフェーズ・フィルタの振幅特性を示す。衛星搭載用フィルタ・バンクの設計事例 [29] を引用し、通過域のリップルを 0.8 dB 以下、帯域外減衰量を 32 dB 以下とした。また、設計の自由度を考慮して窓関数は Kaiser 窓

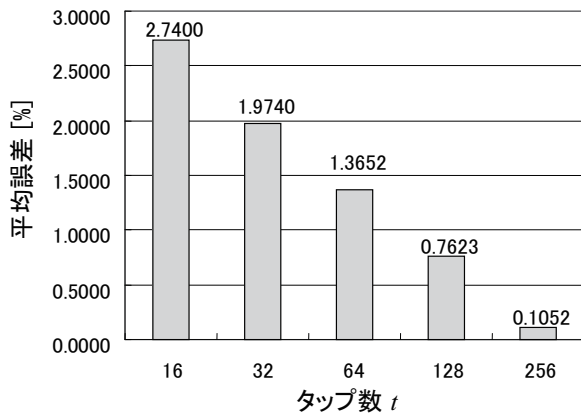


図 18 提案手法の誤差解析.

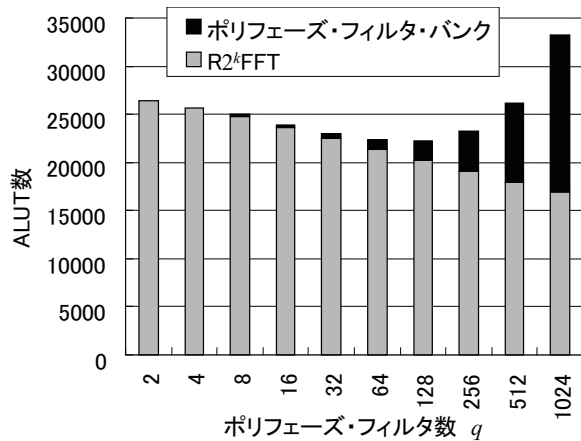


図 19 ALUT 数.

を選択した. 図 17 にポリフェーズ・フィルタ・バンクの振幅特性を示す. この特性も衛星搭載用フィルタ・バンクの設計事例を引用した. タップ数 t を固定し, 各フィルタの遷移域が 3 dB 減衰する点が交差するように各フィルタバンクの係数を求めた. 従って, 提案回路は帯域 $T + P$ で N_q -FFT を行う.

ポリフェーズ・フィルタ・バンクの解析より, タップ数 t はなるべく小さい値が望ましいがタップ数を小さくするとフィルタの遷移域が広がり, 他の帯域の成分が混入するため FFT の誤差が増えてしまう. 図 18 に直接 FFT を行った結果に対して, ポリフェーズ・フィルタ・バンクを行い FFT を行った結果との平均誤差を求めた結果を示す. ただし, この結果は Altera 社 DSP ブロック (丸め誤差回路を含む) を用いた結果であり, 実装に依存することに注意されたい. 図 18 より, タップ数 t を増やすことで平均誤差を削減できる.

4.3 ポリフェーズ・フィルタ数 q とハードウェア量の関係

ポリフェーズ・フィルタ数 q とタップ数 t を変化させ, FPGA のハードウェア量を測定した. 図 19 に ALUT 数を示す. 元の解析点数を N とする. ポリフェーズ・フィルタ数 q に対して提案回路の FFT の解析点数は N/q となるので, R^{2^k} で消費する ALUT 数は線形的に減少する. しかし, ポリフェーズ・フィルタで加算器を用いるため, ポリフェーズ・フィルタ・バンク数 q が指数関数的に増えると加算器が消費する ALUT 数も指数関数的に増加する. 従って, ALUT 数に関してトレード・オフがあり図 19 より, $q = 64 \sim 256$ でバランスが取れていることがわかる. 図 20 に組込みメモリ (M9K) 数を示す. タップ数 t を増やすとポリフェーズ・フィルタで間引きできるサンプル数 M が

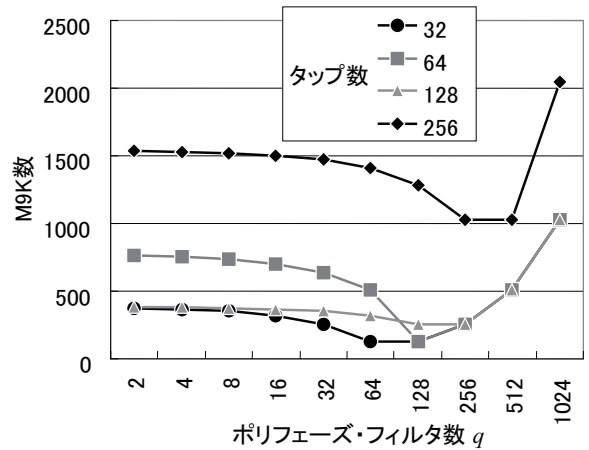


図 20 M9K 数.

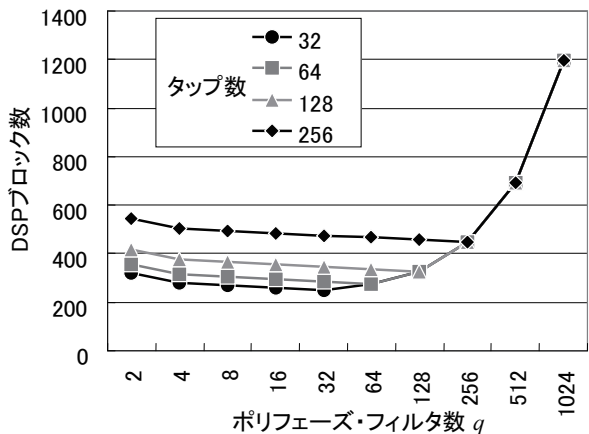


図 21 DSP 数.

表 2 実装結果.

	ALUT 数	M9k 数	DSP ブロック数
ポリフェーズ・フィルタ・バンク	2048	256	128
2^k FFT(4 台)	80,936	0	320
DDR2 SDRAM コントローラ (2 台)	6,036	6	0
合計	89,020	262	448

増えるため, M9K を削減できる. しかし, タップ数 t 以上は間引きできないのでこれ以上は M9K 数が増える. 従って, M9K に関してはフィルタ数 q を増やしたとしても常に削減できるとは限らない. 図 21 に DSP ブロック (乗算器+加算器) 数を示す. M9K と同様に, 間引きできるサンプル数の限界によりタップ数 t までしか削減できない. ただし, M 9 K, DSP ブロックの実験結果は FPGA の構成要素の変更により変わる値であることに注意されたい.

4.4 設計した分光器

SETI spectrometer と同じ 2^{27} FFT を行う分光器を設計した. 図 19 より, FPGA 実装のボトルネックとなる ALUT 数のバランスが取れているタップ数 $t = 128$, ポリフェーズ・フィルタ数 $q = 128$ とし, R^{2^k} FFT を 4 並列に実装し, 高速化を行った. 実装結果を表 2 に示す. また, FPGA 以外に DDR2-SODIMM を 2 個, SSRAM を 1 個を使用した. 最大動作周波数は 375.2 MHz であったが, DDR2SO-DIMM を 667 MHz で動作させるので, PLL を用いてシステムクロックを 333 MHz とした. 提案回路

は1クロックで2個のデータを処理できるので、333 MHz 動作させる場合は $\frac{1}{333 \times 2^{20}}$ 秒で2個のデータを処理できる。提案回路は4並列化しているため、 $N = 2^{27}$ サンプルをFFTする時間は $\frac{2^{27}}{333 \times 2^{20}} \times \frac{1}{2} \times \frac{1}{4} = 0.048$ [sec] である。

FFT演算では N 点のデータに対して N 回加算が行われるため、最悪 $\log_2 N$ ビット桁上げが生じてしまう。従って入力信号の精度を r ビット (符号なし) とすると、最悪の場合、FFTの出力は $r + \log_2 N + 1$ (符号あり) ビットとなる [7], [21]。提案回路はA/D変換器からの入力を8ビットとし、512個のポリフェーズ・フィルタでフィルタリングした 2^{18} 点のデータに対してFFT演算を逐次行い、 2^{27} 点のスペクトラムを得る。よって、最悪の場合の出力は $8 + \log_2 2^{18} + 1 = 27$ ビットとなる。実装した回路は、内部で18ビットの実数と18ビットの虚数に拡張しFFTを行っている。 $R2^k$ FFTは3ステージ構成で実装したので、各ステージの出力で3ビットシフトしスケール調整を行っている。また、誤差を減らすため、乗算器を実装したDSPブロック出力に丸め回路を用いている。SETI SpectrometerもFFT内部で36ビットの複素数 (実部18ビット、虚部18ビット) を用いており、スケール調整を行っているため実用上問題は無い。

4.5 他の手法との比較

ポリフェーズ・フィルタ・バンクと $R2^k$ FFTを用いたSETI spectrometerが提案されている [20]。SETI spectrometerは 2^{27} FFT演算を1.00秒で行うため、提案手法は20.81倍高速である。また、SETI spectrometerは $R2^k$ FFTを用いるため、5台のFPGAを必要とする。一方、提案回路は $R2^k$ FFTを用いるためFFT回路部をコンパクトに実現でき、空きリソースを並列化に適用できる。

5. まとめ

本論文ではポリフェーズ・フィルタ・バンクを用いた $R2^k$ FFT回路について述べた。既存の基数 2^2 FFT回路の解析を行い、メモリが実装のボトルネックになることを示した。ポリフェーズ・フィルタ・バンクを用いて荒い帯域分割を行った後、 $R2^k$ FFT回路を用いてスペクトルに変換する。4並列化した提案回路をAltera社FPGAボードに実装し、サンプル点 2^{27} のFFT演算を0.048秒で行えることを示した。提案回路は、サンプル点 2^{27} FFT演算に関して、SETI spectrometerよりも20.81倍高速であった。

本論文で実装した回路のボトルネックは外付けメモリの帯域である。メモリ数を増加し、帯域を増やすことで更なる高速化が可能である。今後の課題は、電波望遠鏡に適したメモリ帯域が広いFPGAボードを開発し、その性能を明らかにすることである。また、本研究の成果によるFFTライブラリは電波望遠鏡国際開発コミュニティであるCASPER [4]で公開する予定である。

6. 謝 辞

本研究は、一部、日本学術振興会・科学研究費補助金、及び日本学術振興会・頭脳循環プロジェクト補助金による。

文 献

- [1] 赤羽 賢司, 海部 宣男, 田原 博人, “宇宙電波天文学,” 共立出版株式会社, 1988.
- [2] R. Andraka, “A survey of CORDIC algorithms for FPGA based computers,” *FPGA1998*, pp. 191-200, 1998.
- [3] Altera Corp., “FFT MegaCore Function v11.0”, 2011.
- [4] CASPER: Collaboration for Astronomy Signal Processing and

Electronics Research,

<https://casper.berkeley.edu/>

- [5] J. W. Cooley and J. W. Tukey, “An algorithm for the machine computation of complex fourier series,” *Mathematics of Computation*, Vol. 19, pp. 297-301, 1965.
- [6] P. Duhamel and H. Hollmann, “Split-radix FFT algorithm,” *Electron. Lett.*, Vol. 20, pp. 14-16, 1984.
- [7] S. Goedecker, “Fast radix 2,3,4 and 5 kernels for fast fourier transformations on computers with overlapping multiply-add instructions,” *SIAM Journal Sci. Comput.*, Vol. 18, pp. 1605-1611, 1997.
- [8] HALCA: Highly Advanced Laboratory for Communications and Astronomy, <http://www.vsop.isas.jaxa.jp/top.html>
- [9] 半田 利弘, “基礎からわかる天文学: 太陽系から銀河, 観測技術や宇宙論まで,” 誠文堂新光社, 2011.
- [10] S. He and M. Torkelson, “A new approach to pipeline FFT processor,” *IPPS1996*, pp. 766-770, 1996.
- [11] D. H. Jones, A. Powell, C. S. Bouganis, P. Y. K. Cheung, “GPU Versus FPGA for High Productivity Computing,” *FPL2010*, pp. 119-124, 2010.
- [12] V. K. Jain, S. A. Wadkar, and L. Lin, “A universal nonlinear component and its application to WSI,” *IEEE Trans. Components, Hybrids, and Manufacturing Technology*, Vol. 16, No. 7, pp. 656-664, Nov., 1993.
- [13] H. Karner, et al., “Multiply-add optimized FFT kernels,” *Math. Models and Methods in Appl. Sci.*, Vol. 11, pp. 105-117, 2001.
- [14] H. Kondo, E. Heien, M. Okita, D. Werthimer, K. Hagihara, “A multi-GPU spectrometer system for real-time wide bandwidth radio signal analysis,” *ISPA2010*, pp. 594-604, 2010.
- [15] W. R. Knight and R. Kaiser, “A Simple Fixed-Point Error Bound for the Fast Fourier Transform,” *IEEE Trans. Acoustics, Speech and Signal Proc.*, Vol. 27, No. 6, pp. 615-620, 1979.
- [16] E. N. Linzer and E. Feig, “Implementation of efficient FFT algorithms on fused multiply-add architectures,” *IEEE Trans. Signal Processing*, Vol. 41, pp. 93-107, 1993.
- [17] R. G. Lyons, “Understanding Digital Signal Processing,” *Pearson Education*, 1996.
- [18] 中井直正, 坪井昌人, 福井康雄, “宇宙の観測 II(16): 電波天文学,” 日本評論者, 2009.
- [19] A. Hirota, N. Kuno, N. Sato, H. Nakanishi, T. Tosaki, and K. Sorai, “Variation of molecular gas properties across the spiral arms in IC 342: Large-scale 13CO (1-0) emission”, *PASJ2010*, No. 62, pp. 1261-1275, 2010.
- [20] A. Parsons et.al, “PetaOp/Second FPGA signal processing for SETI and radio astronomy,” *Proc. of the Asilomar Conference on Signals, Systems, and Computers*, 2006.
- [21] L. R. Rabiner and B. Gold, “Theory and Application of Digital Signal Processing,” *Prentice-Hall Inc.*, 1975.
- [22] T. Sasao, S. Nagayama and J. T. Butler, “Numerical function generators using LUT cascades,” *IEEE Trans. on Comput.*, Vol.56, No.6, June 2007, pp.826-838.
- [23] SKA: Square Kilometre Array, <http://www.skatelescope.org/>
- [24] SERENDIP: The Search for extra terrestrial intelligence at UC Berkeley, <http://seti.berkeley.edu/SERENDIP/>
- [25] M. J. Schulte and J. E. Stine, “Approximating elementary functions with symmetric bipartite tables,” *IEEE Trans. Comput.*, Vol. 48, No. 8, pp. 842-847, Aug., 1999.
- [26] NVIDIA Corp., “TESLA S1070 GPU COMPUTING SYSTEM,” <http://www.nvidia.com/>
- [27] J. E. Volder, “The CORDIC trigonometric computing technique,” *IRE Trans. on Electronic Comput.*, pp. 330-334, 1959.
- [28] Xilinx Inc., “LogiCORE IP fast fourier transform v7.1”, 2011.
- [29] 山下 史洋, 風間 宏志, 中須賀 好典, “衛星搭載用帯域幅可変FFTフィルタバンクの提案と基本動作特性,” 電子情報通信学会論文誌 B, 通信 J85-B(12), pp. 2290-2299, 2002.
- [30] B. Zhou, Y. Peng, and D. Hwang, “Pipeline FFT architectures optimized for FPGAs,” *Int'l Journal of Reconfigurable Computing*, Vol. 2009, 2009.