

波長多重光インターコネクションによる高速フーリエ変換専用コンピューターのアーキテクチャー

安 哲 興・峯 本 工

神戸大学大学院自然科学研究科 〒657 神戸市灘区六甲台町 1-1

Architecture of Fast Fourier Transform Special-Purpose Computer Using Wavelength-Multiplexed Optical Interconnection

Chulheung AN and Takumi MINEMOTO

Graduate School of Science and Technology, Kobe University, 1-1, Rokkodai, Nada-ku, Kobe 657

Fast Fourier transform (FFT) is important for engineering study including image processing and signal processing. For a better improvement of computing speed in current electronic computer, a new architecture of FFT special-purpose computer with compact size, in which 2-D FFT operation of sample points (1024×1024) can be performed in a video rate, has been proposed using a method of wavelength multiplexed optical interconnection.

1. はじめに

フーリエ変換 (FT: Fourier transform) は物理学や工学的な応用にとって重要な原理と手段であることは周知の事実である。特に、空間フィルタリング・コンボリューション・コリレーション・特徴抽出・エッジ強調などを行う光情報処理のアルゴリズムまたはアーキテクチャーは FT やアダマール変換に基づいている¹⁾。そして、電子計算機の中で FT を速く実行する高速フーリエ変換 (FFT: fast Fourier transform) アルゴリズムが開発され、画像処理・信号処理の分野で多く利用されている。また、電子計算機によって 2 次元のレイデータを取り扱うパターン認識および種々の画像処理 (例えば、2 値減算型ジョイントトランスフォルムコリレータ²⁾によるパターン認識では 3 回の FFT の実行が要求されている) を実用化するために、ビデオレートで FFT を実行できることが切に要求されている場合が多い。しかし、FFT はサンプル点数が多くなるにつれ、計算量の爆発を起こす典型的な例としても知られている。すなわち、 (1024×1024) 点数の 2 次元の複素数 FFT は大型のスーパーコンピューターでないとビデオレートで実現できないのが現在の実状である。そのた

め、可能な限り小型の速い計算速度のコンピューターが望まれている。そのためには、問題専用のハードウェアに強く依存したアーキテクチャーにならざるをえない。FFT を基本処理に利用している工学関係の諸研究および他分野の研究においても精度の高い処理を要求するため、処理すべきデータ情報量が多くなってきている。さらに、実時間処理も要求される。このような趨勢はいろいろな研究分野でもこれからの一般的な傾向であると考えられるので、コンパクトでかつ超高速な専用の並列コンピューターがそれぞれの研究や応用分野で簡単に入手し、利用できることが不可欠になると思われる。

レンズを用いた光学的フーリエ変換は電子コンピューターとは比べものにならないほど非常に速い処理が可能であるが、基本的にアナログ処理であるので、精度不足という問題は根本的に避けられない。高い精度の計算が重要な場合は、電子計算機を用いてフーリエ変換を実行しなければならない。また、電子計算機のもうひとつの特徴である柔軟性は非常に重要な要素である。一方、光学的情報処理システムは超高速の実行速度だけではなく、レンズの波面変換作用からフーリエ変換などの積分変換およびフィルタリング作用などの物理的概念が理解

しやすいかたちで導入されるところにその意味があるといえる³⁾。ところで、現在のコンピューターの処理速度は電子コンピューターの構造的な問題からMPU (micro processing unit) チップ自身の処理性能よりもチップ間の通信時間によって処理速度が左右されているのが現状である。この通信ボトルネックを解決するため、光インターコネクションが考えられ、いろいろな階層のインターコネクションに対して種々の方式の光インターコネクション⁴⁾が提案されている。

本研究の目的はボード間レベルでの光インターコネクション^{5,6)}を利用した超高速FFT専用コンピューターの設計である。本研究では2次元FFTのバタフライネットワークのデータの流れをうまく表現できる4波長多重光インターコネクションを用いて、サンプル点数(1024×1024)の2次元FFTをビデオレート(1/30s)で実行することのできるFFT専用コンピューターのアーキテクチャーを提案する。この提案ではMPUチップの入っている16個のボード間を積層型平行反射板を利用した光インターコネクションで接続するので、コンパクトな構造になっているのが特徴である。また、提案された専用コンピューターは現在の技術レベルで実現可能なこともうひとつの長所でもあり同時に、それはFFTの特殊な性質であるバタフライネットワークから考案されたものであるが、一般の並列コンピューターの設計にも十分応用することが可能である。

効率よいFFT専用コンピューターのアーキテクチャーを実現するためには、FFTの数学的な規則から見直すことから始めなければならない⁷⁾。本論文の2節ではFFTの数学的な演算規則を検討し、3節ではFFTの演算規則に忠実に対応するハードウェアアーキテクチャーを新しく提案する。そして、サンプル点数(1024×1024)の2次元FFTをビデオレートで実行するために必要な条件を4節で述べ、5節では提案した本方式のアーキテクチャーが現在の電子技術とうまく整合する光インターコネクションであることを述べ、提案のインター

コネクションが電子技術では実現がほぼ不可能な超高速データ並列伝送技術であることを示す。

2. FFTの原理 (DFTの原理)

1次元のDFT (discrete Fourier transform) は式(1)により表現される。

$$G(u) = \sum_{m=0}^{M-1} g(m) \exp(-j2\pi um/M) \quad (1)$$

ここで、 $g(m)$ は1次元空間でのサンプル値関数、 M はサンプル点数、 $G(u)$ は1次元周波数空間でのスペクトル関数である。この1次元のDFTを2次元へ拡張すると式(2)になる。この式で $g(m, n)$ と $G(u, v)$ は各々2次元空間でのサンプル値関数、スペクトル関数を表している。

$$G(u, v) = \sum_{m=0}^{M-1} \sum_{n=0}^{M-1} g(m, n) \exp[-j2\pi(um+vn)/M] \quad (2)$$

DFTを現在の電子計算機の中で高速で実行するアルゴリズムであるFFTはバタフライネットワークとして知られている⁸⁾。サンプル点数が8個の1次元の場合のバタフライネットワークでの信号の流れをFig.1に示した。すなわち、FFTの計算機プログラムはこのバタフライネットワークの流れをコーディングしているにすぎない。光情報処理における画像認識などの世界では主に2次元のデータが取り扱われるので2次元FFTが重要な対象になる。新しく提案するFFT専用コンピューターのアーキテクチャーにおける数学的な根拠を導出するために2次元DFTの式(2)について、バタフライネットワークの信号流れを詳しく考察する必要がある。FFTアルゴリズムでは周波数間引き法と時間間引き法との2つの方式がある。両方ともバタフライネットワークの考え方・計算量はまったく同じで、周波数間引き法ではバタフライネットワークの最終出力をビットリバーサル (bit reversal) するが、時間間引き法ではバタフライネットワークヘデータをを入力する最初の段階でビットリバーサルを行うことだけが違う。したがって、ここ

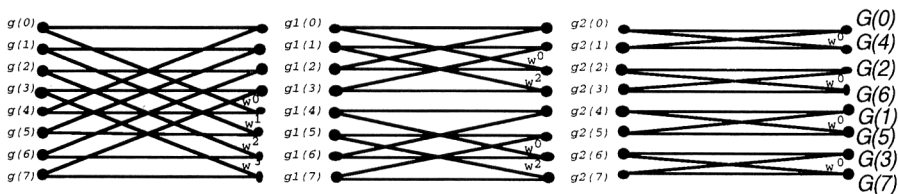


Fig. 1. Signal streams diagram of butterfly network in 1-dimensional FFT (8-points), where $W = \exp(-j2\pi/M)$.

では周波数間引き法に対して考察することにする。
 〈周波数間引き法に基づくバタフライネットワーク〉

まず、イメージ空間のサンプル値関数 $g(m, n)$ と周波数空間のスペクトル関数 $G(u, v)$ を Fig. 2 に示したように4分割して考えることにする。2次元のデータ空間サイズは $M \times M$ であるとして、Fig. 2(a) に示したように4分割したものを $g_1(m, n)$, $g_2(m, n)$, $g_3(m, n)$, $g_4(m, n)$ と書くと、4分割したデータ群は次式により表現される。ただし、 M は偶数整数とする。

$$g_1(m, n) = g(m, n) \quad (3a)$$

$$g_2(m, n) = g\left(m + \frac{M}{2}, n\right) \quad (3b)$$

$$g_3(m, n) = g\left(m, n + \frac{M}{2}\right) \quad (3c)$$

$$g_4(m, n) = g\left(m + \frac{M}{2}, n + \frac{M}{2}\right) \quad (3d)$$

ここで、 $0 \leq m \leq M/2 - 1$, $0 \leq n \leq M/2 - 1$ である。周波数空間を4分割したものを Fig. 2(b) のように表し、2次元FFTの最終段で得られる結果のデータをビットリバーサルして得たものがスペクトル空間でのデータである。サンプル点数 $M \times M$ の2次元FFTを実行するとき、バタフライネットワーク・ステップ (butterfly network step) の数は $\log_2 M$ である。この Fig. 2 のように $M \times M$ 点の2次元フーリエ変換は1番目ステップで4組の $M/2 \times M/2$ 点フーリエ変換に分解できる。同様の手続きを繰り返せば、 $M/2 \times M/2$ 点フーリエ変換はさらに、4組の $M/4 \times M/4$ 点フーリエ変換に分解することができる。順次これを n 回繰り返せば $N/2^n$ 点フーリエ変換となる。最後に 1×1 点フーリエ変換になるまで繰り返したときに、その必要回数が $\log_2 M$ である。また、FFTはバタフライ・ステップが進むにつれ、FFT演算のためのデータは近傍に限られていくので、FFT演算は近傍演算になっていく。そのため、FFTを高速で完了するには、FFTを実行すると

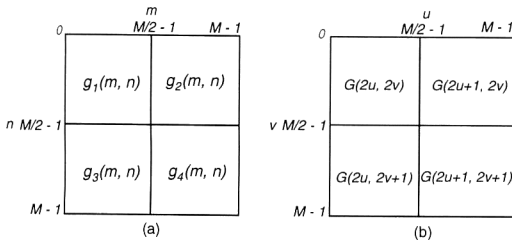


Fig. 2. Four partitions of (a): image space and (b): frequency space in 2-dimensional FFT ($M \times M$ points) in the method of decimation in frequency.

きにデータの遠距離通信が激しい第1, 2のバタフライ・ステップで、何らかのかたちで大容量通信バンド幅を行えばよいことになる。周波数間引き法のFFTではバタフライ最終段階のデータのアドレスをビット逆読み (bit reversal) することにより、最終的に求める周波数空間のデータが得られる。これはイメージ空間のデータをフーリエ変換した結果のスペクトル値は、周波数空間で偶数空間と奇数空間に分解されているためである。Fig. 2(b) はこうした意味の周波数空間である。このフーリエ空間データをイメージ空間データにより求めるため、式(3)を使って式(2)を書き直すと次の式(4)になる。

$$G(u, v) = \sum_{m=0}^{M/2-1} \sum_{n=0}^{M/2-1} [g_1(m, n) + (-1)^u g_2(m, n) + (-1)^v g_3(m, n) + (-1)^{u+v} g_4(m, n)] \times W^{um} W^{vn} \quad (4)$$

ここで、

$$W = \exp(-j2\pi/M)$$

この式(4)を次のように4つの場合に分けて考えると式(5)~(8)になる。

(2.1.a) u : 偶数と v : 偶数の場合は、

$$G(2u, 2v) = \sum \sum [g_1(m, n) + g_2(m, n) + g_3(m, n) + g_4(m, n)] \times W^{2um} W^{2vn} \quad (5)$$

(2.1.b) u : 奇数と v : 偶数の場合は、

$$G(2u+1, 2v) = \sum \sum [g_1(m, n) - g_2(m, n) + g_3(m, n) - g_4(m, n)] W^m \times W^{2um} W^{2vn} \quad (6)$$

(2.1.c) u : 偶数と v : 奇数の場合は、

$$G(2u, 2v+1) = \sum \sum [g_1(m, n) + g_2(m, n) - g_3(m, n) - g_4(m, n)] W^n \times W^{2um} W^{2vn} \quad (7)$$

(2.1.d) u : 奇数と v : 奇数の場合は、

$$G(2u+1, 2v+1) = \sum \sum [g_1(m, n) - g_2(m, n) - g_3(m, n) + g_4(m, n)] W^{m+n} \times W^{2um} W^{2vn} \quad (8)$$

上の式(5)~(8)は $M \times M$ 点の2次元フーリエ変換が $M/2 \times M/2$ 点フーリエ変換に分解されることを示している。この $M/2 \times M/2$ 点フーリエ変換を 1×1 点フーリエ変換になるまで、前述のとおりの手続きを繰り返せば、最終的にFFTができ上がる。このとき、各バタフライネットワーク・ステップでの計算量はまったく同じである。Fig. 3では2次元サンプル点 (8×8) の場合について、周波数間引き法に対するバタフライネットワークを示した。この図からバタフライネットワークでのデータの流れを理解することができる。

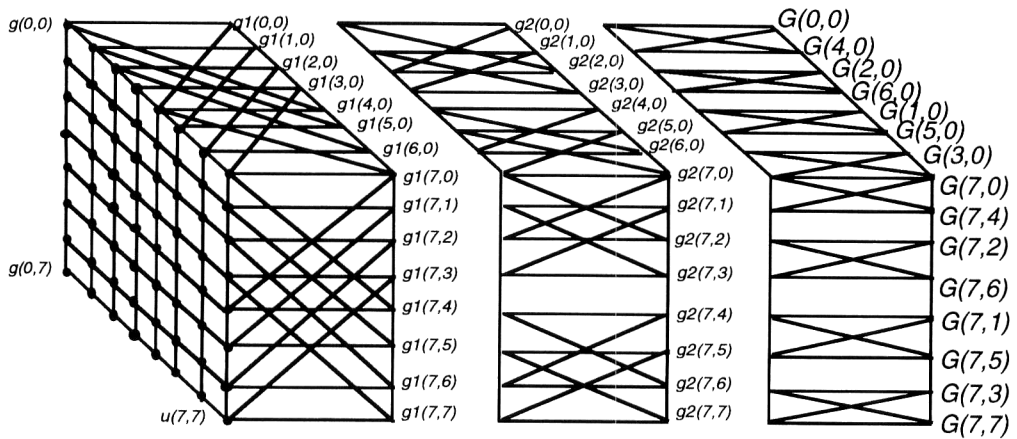


Fig. 3. Signal streams of 2-dimensional butterfly network in the method of decimation in frequency.

3. 波長多重光インターコネクションによる高性能の新しいFFT専用コンピューターアーキテクチャーの提案

フーリエ変換はレンズのフーリエ変換作用からもわかるように、根本的に並列処理である。FFT専用の並列電子計算機では並列演算がなされるのは各々のパタフライ・ステップであり、パタフライ・ステップとステップの間は逐次処理になる。それで、各々のパタフライ・ステップごとに、各サンプル点は同時並列計算の対象になるので、サンプル点数が多ければ多いほど並列計算量は増加する。しかし、多数の演算子間のインターコネクション問題のため、電気配線のみでこの問題を解決して並列度の高い電子コンピューターを製作することは不可能に近い。しかし、現時点で実現可能な実用化計算機システムでは電子計算機が持っているプログラミング性によって、事務処理から科学技術計算やシミュレーションにいたるまであらゆる問題の処理に対応できる。この電子計算機のもつ柔軟性と高い演算精度は、現実的にほかの弱点を補う大きな利点である。また、現状の並列電子計算機には種々の方式が存在するが、その中でマルチプロセッサ型の並列計算機が多くの問題に対応可能な汎用性が高い方式である。それで、逐次処理と並列処理の両方をもつマルチプロセッサ型⁹⁾のFFT専用コンピューターがもっとも柔軟性に富んでいるので、この方式をボードレベルの光インターコネクションによって実現し、その性能を向上させるのが現状では優れた方法のひとつだと思われる。主にFFT専用コンピューターで提案された、このマルチプロセッサ方式の専用コンピューターは、FFTだけではなくほかの多くの問題にも対応可能となるに違いない。これは世の自然系が逐次

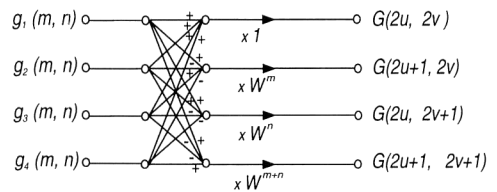


Fig. 4. Mapping representations of equations in relation to Eqs.(5)-(8).

性と並列性両方の表現であることを思い出すとごく自然な考え方とみなせる。

2節で言及したように、周波数間引き法での4つの式(5)~(8)そのままでは、その演算に関わるデータの流れ等に関するイメージをつかむのは難しいので、わかりやすく表現することから始める。この式における関係を理解しやすくするために、図式的に示したのがFig.4である。

このFig.4をよく観察するとサンプル点からの信号は自分自身を含んで4つに分配されていることがわかる。これが(2×2)のラディックスパタフライである⁷⁾。4つのサンプル点がFFT演算に関係していることが簡単にわかる。すなわち、4分割されたデータ空間で1つの空間での各1点に対応するのは別々の3つの空間での各1点である。このラディックスパタフライネットワークの信号の流れをそっくり表しているのがFig.5である。この図からわかるように、これは複数のボード(4つのボード)間の光インターコネクションである¹⁰⁾。また、この光インターコネクションでは2枚の平行反射板の間に多数の受光素子と発光素子を置いたユニットを積層することにより、PE (processing element) またはMPUを積載した複数のボード間のインターコネ

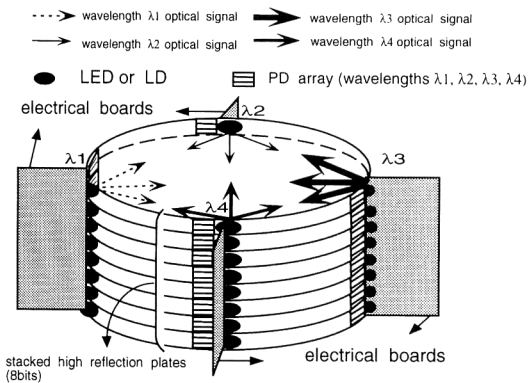


Fig. 5. Four wavelengths multiplexed data communication between 4 MPU boards using 8 stacked plates (8 bits).

クションを実現しようとするものである。光は反射膜の間に閉じ込められて反射しながら伝搬し、情報のやり取りをする。4つのボード間結合のために使う光の波長はボードごとに決めておく。たとえばボード1からは波長 λ_1 の光を出し、ボード2からは波長 λ_2 の光を出すというように割り当てる。そして、各ボードには自分から出す光の波長と異なる波長の光を選択的に受ける3種類のホトセンサーを配置し、他のボードから送られてきたデータを取り込む。発光素子と受光素子を取り付けた1枚の平行平板反射板を情報の各ビット単位に対応させ、これらを積層したかたちのインターコネクションでボード間を結合する。そして、各ボードには分割された各空間が対応する。4つのMPUボードのFig. 5では、ボード間通信はデータ通信量が一番多いパタフライネットワークの第一段階だけで発生するということである。この段階だけでも大容量通信ができることは、チップ間通信時間に悩んでいるFFT専用の電子計算機の性能を飛躍的に向上させる。パタフライの2段階以後からは、ボード間のデータ通信はなくなるので、逐次的に演算することになる。しかし、性能をもっと向上させるためには、FFTされるデータ空間を多く分割して、MPUボードの数を増加させることである。そして、16個分割の空間に対応した16MPUボードを使ったアーキテクチャーをFig. 6で示した。

このFig. 6での(a)は第1ステップのパタフライネットワークでの関係のあるボード間同士の信号の流れを表している。このステップでは、16個に分割された領域全部がFFTされるため16MPUボード間で激しい同時通信が発生していることがわかる。この図の(b)は第2ステップのパタフライネットワークで関係のあるボード

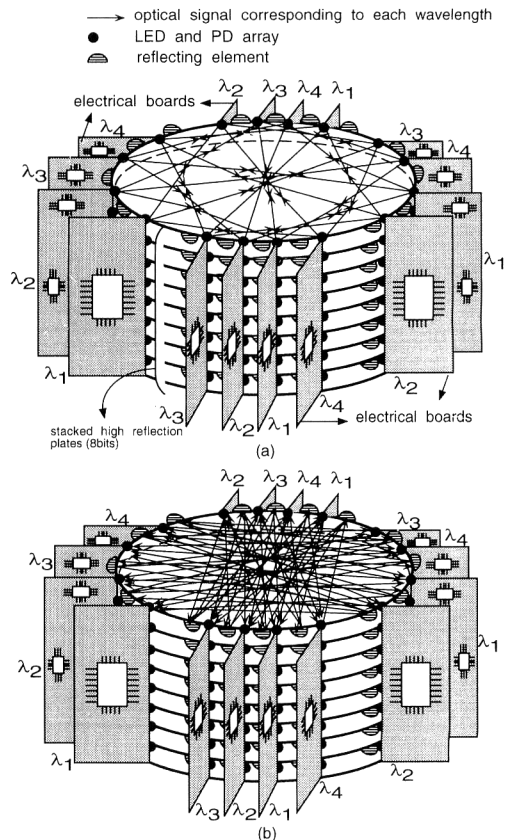


Fig. 6. Mapping of four wavelengths multiplexed data communication between 16 MPU boards; (a): the first stage and (b): the second stage of butterfly network in the method of decimation in frequency.

間の信号の流れを表している。また、Fig. 6ではパタフライネットワークの第3ステップ以後は逐次演算することになる。この段階からMPUボード間のデータ通信量は少なくなって近傍間演算になるので、MPUの通信負担は大幅に減ってしまう。この図の(b)では、隣の4つのボード間同士のデータ通信を行うため、向かい合っているボードの間に小さい反射体を置いて、その反射体からの信号を用いて互いに通信する構想になっている。こうするためには、Fig. 6に示した基本構成とは別に反射信号のためもうひとつの発光素子を設けなければならないが、受光素子は新しくもって設置しなくてもすむのでコンパクトなサイズになる長所がある。実際、受光素子の方が発光素子より現実的な問題が多いのが一般的である。受光素子を減らすとさまざまなメリットがあるので、Fig. 6の(b)の構成は非常に望ましいものであると考えられる。また、Fig. 6では関係のあるボード間のデータ通信の様子を矢印で示した。この図からデータの流

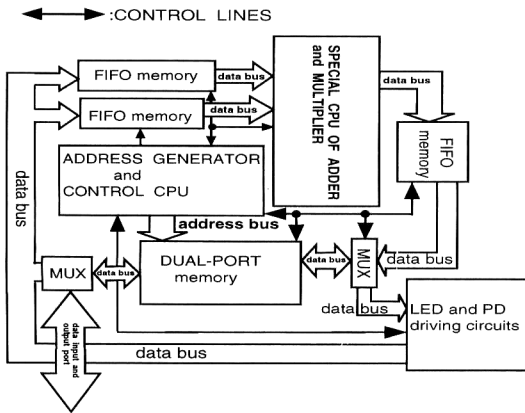


Fig. 7. An example of computing circuit system within intraboard.

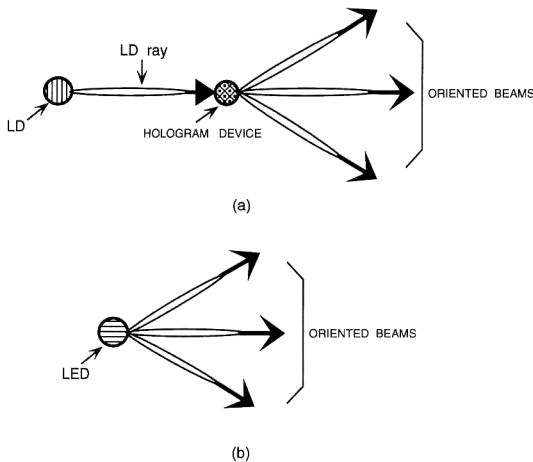


Fig. 8. Efficient designs of light emitting devices; (a): three direction partitions of highly oriented of LD beam by a hologram element and (b): special make-up of LED emitting in three directions.

れ方がラディックスバタフライの構造になっていることがわかる。

1つのMPUボード内での計算回路のブロックダイアグラムの一例をFig. 7に示す。この図のようにデータの入力線バスと出力線バス別々に設計された専用のMPUを使用するならば、さらに高速計算が可能になると思われる。本研究で提案したFFT専用コンピュータのアーキテクチャーで使われる発光素子は、LD (laser diode) よりもLED (light emitting diode) を想定している。LDは周辺の制御回路が必要なので全体的にコンパクトなサイズを追求することに逆行するものになる。しかし、LDを使うとすればFig. 8の(a)のよう

に小さなホログラムデバイスまたは特殊レンズなどを使って信号ビームを3つに分配する必要がある。LEDの場合、Fig. 5では広い放射角をもつものでもかまわないが、光のエネルギー集光効率が悪いので、できればFig. 8の(b)のように3方向に高い指向性をもつように特別設計で製作されたLEDが望ましい。なお、Fig. 6では必ずこのような特別製作された発光素子を用いなければならないことに注意を要する。

4. 提案したFFT専用コンピュータの性能評価

FFTをマルチプロセッサ型の並列コンピュータで計算量を分担して処理する場合には、時間間引き法と周波数間引き法はまったく同一なので、どちらを使ってもよいことになる。前節のアーキテクチャーで4つの波長を選んだことはFFTの演算に参与する4つが一组になっていることに由来する。提案したシステムの評価は、Fig. 5のような4つのMPUボードのマルチプロセッサ型のコンピュータの場合と、Fig. 6のように第2ステップのバタフライまで波長多重設計が容易な16個のMPUボードのマルチプロセッサ型のコンピュータの場合において考えることにする。それで、16個のMPUボードのマルチプロセッサ型システムを用いて、 1024×1024 サンプル点のFFT演算を実時間で行う可能性を考察する。

周波数間引き法の式をよく観察すると、バタフライネットワーク・ステップでの計算は3回の複素加算と1回の複素乗算だけになっている。この3回の複素加算は6回の実数加算であり、1回の複素乗算は4回の実数乗算と2回の実数加算である。したがって、12回の加算と乗算が必要になる。計算機では加算より乗算の方に時間がかかるが、ここでは加算も乗算と同じように演算回数に入れて見積もることにする。以下では、1回の実数加算に要する時間を1回の実数乗算に要する時間と等しいとして、FFT演算を完了するまでに要する時間を見積もることにする。バタフライネットワークの第3番目ステップからは各MPUボードは逐次演算になるので、MPUボード間のデータ通信はなくなる。このときには、式(5)~(8)の全部を演算回数に入れなければならない。そうすると3回の複素乗算と8回の複素加算が行われるのがわかる。この3回の複素乗算は12回の実数乗算と6回の実数加算であり、8回の複素加算は16回の実数加算である。そして、合計34回の実数加算と実数乗算が行われる。

上記の考え方に従ってFFT演算を遂行するとして、

計算が完了するまで総実数加算と乗算の計算量を算出することにする。また、本提案のFFTアーキテクチャーはFig. 4に示したように4つのプロセッサを基本単位にしている。すなわち、基本的な4つのMPUボードを単位ユニットとし、並列演算システムの大規模化と高速化は4を基数とする 4^n 個のMPUを用いたシステムで実現される。ここで、 n は2次元のサンプルデータを $(M \times M)$ 点とすると $1 \leq n \leq \log_2 M$ の範囲の数値である。以上のことを総合してFFT演算される2次元のサンプルデータ数量が $(M \times M)$ の場合、 4^n 個のMPUボードのマルチプロセッサの1つのボードで実行される実数加算の演算回数と実数乗算の演算回数を単純に加え合わせたFFTの総演算回数 F (回)は次の式(9)で表現される。

$$F = \frac{M^2}{4^n} \left[2 \times a \times \frac{b}{4} \times (\log_2 M - n) \right] \quad (9)$$

ここで、 a は最初のバタフライネットワーク・ステップでの実数加算と乗算の回数(=12)であり、 b は残りのバタフライネットワーク・ステップでの実数加算と乗算の回数(=34)である。この式(9)から $(M \times M)$ 点のFFTを1/30秒で完了するのに必要なMPUの浮動小数点処理能力の目安であるMFLOPS (mega floating operations per second) が得られる。したがって、2次元サンプルデータ $(M \times M)$ 点数が 4^n 個のマルチプロセッサ型のシステムでビデオレートで実現されるためには、式(10)のような性能をもったMPUが必要である。

$$30 \times \frac{M^2}{4^n} \left[2 \times a \times \frac{b}{4} \times (\log_2 M - n) \right] \quad (\text{MFLOPS}) \quad (10)$$

この式(10)から4つのMPU($n=1$ の場合)のシステムで (1024×1024) サンプル点のFFTをビデオレートで実行するためには約696 MFLOPSの性能をもったMPUが必要になることがわかる。単体のマイクロプロセッサでこのような性能をもったMPUはまだ存在しない。したがって、MPUの数を増やさねばならないことがわかる。16個のMPU($n=2$ の場合)で構成するFig. 6のシステムでは同じように考えて、 (1024×1024) 点のFFTを1/30秒以下で計算するのに1つのMPUのボードで約181 MFLOPS性能をもったMPUが要求されることがわかる。これらは計算時間の短い加算を、乗算に要する時間に等しいとして見積もったものなので、余裕ある十分な見積り値である。

現在、商用化されている単体のマイクロプロセッサの性能は300 MFLOPSである。181 MFLOPS性能の

MPUを16個使って (1024×1024) サンプル点のFFTをビデオレートで実現することは現実的に可能なことである。しかし、MPUの性能だけ高くてもプロセッサとメモリー、MPUボード間の通信バンド帯域幅が低いとビデオレートの性能は達成できない¹¹⁾。これがいわゆる通信ボトルネックと呼ばれるものである。普通、浮動小数点演算は8バイトの大きさのデータなので、提案しているシステムでは1ボード当たり発光素子・受光素子が64以上のアレイの構成にならざるをえない。181 MFLOPSの性能に合わせて (1024×1024) サンプル点のFFTが実時間で終わるにはプロセッサとメモリー、MPUボード間の通信バンド帯域幅は少なくとも約786.5 Mbyte/s (=65536 \times 10 \times 30 \times 5 \times 8)以上でなければならない。したがって発光素子の変調周波数は32.8 MHz (=786.5/8/3)以上が必要になる。100 MHzの変調速度をもつ発光素子なら2.4 Gbyte/s (=100 \times 3 \times 8)の通信バンド幅が可能なので、十分余裕ある設計になると考えられる。この性能は、ボード間の電気配線では実現が難しい高い通信バンド幅であり、ここで提案したアーキテクチャーは意義があると考えられる。

5. 結 論

並列電子計算機を実現するのに配線の複雑な問題は避けられない。この配線の問題はインターコネクションの問題であり、リングング・クロストーク・CR (capacitance and resistance) 遅延などから派生しているものである。この問題は原理的なことなので、解決するためにはいろいろな階層の光インターコネクションに依存せざるを得ないのが実状である¹²⁻¹⁵⁾。計算機同士のネットワークや長距離データ通信で大成功をおさめた光ファイバーのインターコネクションはボード間レベルでの大容量通信を遂行する役割にも拡張されなければならない。しかし、まだボードレベルの光インターコネクションの定型は決まっていない。本研究では現状の技術で実現しやすく、先に提案しているボード間光インターコネクション¹⁰⁾を用いてのFFT専用の並列コンピューターを提案してその性能を評価した。FFTは計算量が多くて通信が頻繁に発生する典型的な例である。その結果として181 MFLOPSのMPUを16個使うことにより (1024×1024) の2次元サンプル点のFFTが実時間で完成できることがわかった。計算機の演算における加算は、乗算に比べてそれほど時間がかからないことを考慮すると、加算と乗算の演算時間を等しいとおいて演算回数をすべて入れて見積もったMPUの演算能力181

MFLOPS の半分より少し上の 120 MFLOPS の性能で十分だと考えられる。しかし、この性能の MPU を利用して電気配線だけで構成したマルチプロセッサ型のコンピュータでは不可能に近い複雑な配線のため、(1024×1024) の 2 次元 FFT が実時間で実行できない。すくなくとも数百以上の MPU が必要になる。また、本提案システムの高反射率の積層板は、近年並列計算機で深刻な問題になっている熱を発散させるための放熱板として利用し得ることももうひとつの長所であるといえる。これらのことを総合的に判断して、本研究で提案している FFT 専用の並列コンピュータは柔軟性に富んだ実現性の高い非常に強力なシステムであるといえる。

文 献

- 1) D.-G. Sun, N.-X. Wang, L.-M. He, Z.-W. Lu and Z.-H. Weng: "Butterfly interconnection networks and their application in information processing and optical computing: applications in fast Fourier-transform-based optical information processing," *Appl. Opt.*, **32** (1993) 7184-7193.
- 2) Q. Zhan and T. Minemoto: "Successful pattern matching with a large number of reference patterns using a joint Fourier-transform correlator," *Jpn. J. Appl. Phys.*, **32** (1993) 3471-3476.
- 3) F. T. S. Yu: "Optical information processing," *Optical Signal Processing Fourier Optics* (Wiley/Interscience, New York, 1982).
- 4) M. K. Kilcoyne, D. Kasemset, R. Asatourian and S. Becue: "Optical data transmission between high speed digital integrated circuit chips," *Optical Computing*, ed. J. A. Neff, *Proc. SPIE*, **625** (1986) 134-142.
- 5) R. A. Nordin, A. F. J. Levi, R. N. Nottenburg, J. O'Gorman, T. Tanbun-Ek and R. A. Logan: "A systems perspective on digital interconnection technology," *IEEE J. Lightwave Technol.*, **10** (1992) 811-827.
- 6) P. R. Haung, S. Rychnovsky, A. Husain and L. D. Hutcheson: "Optical interconnects for high speed computing," *Opt. Eng.*, **25** (1986) 1076-1085.
- 7) 辻井重男: デジタル信号処理の基礎, 2nd ed. (電子情報通信学会, 1988) pp. 305-308.
- 8) K. Hwang and F. A. Briggs: *Computer Architecture and Parallel Processing*, 4th ed. (McGraw-Hill Co., Singapore, 1988) pp. 367-373.
- 9) T. Sakano, T. Matsumoto and K. Noguchi: "Three-dimensional board-to-board free-space optical interconnects and their application to the prototype multiprocessor system: COSINE-III," *Appl. Opt.*, **34** (1995) 1815-1822.
- 10) 安 哲興, 峯本 工: "高反射率積層板によるボード間波長多重光インターコネクション", *光学*, **24** (1995) 574-580.
- 11) P. B. Berra, A. Ghafoor, M. Guizani, S. J. Marcinkowski and P. A. Mitkas: "Optics and supercomputing," *Proc. IEEE*, **77** (1989) 1797-1815.
- 12) D. Z. Tsang: "One-gigabit per second free-space optical interconnection," *Appl. Opt.*, **29** (1990) 2034-2037.
- 13) W. T. Cathy and B. J. Smith: "High concurrency data bus using arrays of optical emitters and detectors," *Appl. Opt.*, **18** (1979) 1687-1691.
- 14) J. W. Goodman, F. J. Leonberger, S.-Y. Kung and R. A. Athale: "Optical interconnection for VLSI systems," *Proc. IEEE*, **72** (1984) 850-866.
- 15) W. H. Wu, L. A. Bergman, A. R. Johnston, C. C. Guest, S. C. Esener, P. K. L. Yu, M. R. Feldman and S. H. Lee: "Implementation of optical interconnections for VLSI," *IEEE Trans. Devices*, **ED-34** (1987) 706-714.