

非均等な時間間隔サンプリングされたデータの周波数成分ベクトルを求める演算回路

田中 優* 梁 維焜 萩原良昭 (崇城大学)
E-mail: hagiwara@cis.sojo-u.ac.jp

Digital Frequency Transformation Circuit for Time-wise Unequally Sampled Data

Masaru TANAKA, Weikun LIANG, and Yoshiaki HAGIWARA

DFT is essential for voice and picture recognition. Normally DFT is processed for Time-wise Equally Sampled Data. This paper reports a challenge to design a DFT circuit for Time-wise Unequally Sampled Data.

キーワード：離散フーリエ変換、再帰的手続き、デジタル回路設計、回路性能予想

Keywords DFT, Recursive Procedure, Digital Circuits Design, Circuit Simulation

1. はじめに、

従来の離散フーリエ変換回路では均等な時間間隔でアナログ情報を複数個サンプルし、その時間軸ベクトルを求め、次に回転因子行列と言われるもので行列演算処理して周波数成分ベクトルを求めている。しかし、非均等な時間間隔で、複数個サンプリングした場合にも同様に周波数成分ベクトルを求める事ができ、その結果回転因子行列に相当する行列式がある事を報告する。その演算処理回路の長所と短所についても、従来の離散フーリエ変換回路と対比して、報告する。

2. 離散フーリエ変換(DFT)の定義

離散フーリエ変換(DFT=Discrete Fourier Transformation)演算とは、ある一定の時間区間 T を等間隔に N 区分し、区分した各離散時間点ベクトル $t[]$ の時点で、アナログ入力信号 $a(t)$ を sampling して、アナログ信号ベクトル $a[]$ を求めて、そのベクトル $a[]$ と、回転因子行列といわれる行列式 $W[][]$ とで、かけ算を実行し、周波数成分ベクトル $A[]=W[][]a[]$ を求めることをいいます。周波数成分ベクトル $A[]$ の各要素値 $A[m]$ は、次の行列式演算をすることにより求められます。m の値は 1 から N 迄の整数とします。

$$A[m] = \sum_{n=1}^N W[m][n]a[n] \dots \dots (1)$$

ここで、この回転因子行列 $W[][]$ の各要素値 $W[m][n]$ は次式で与えられます。

$$W[m][n] = \left(\frac{1}{N}\right) \exp(-2\pi mnj/N) \dots \dots (2)$$

また、この周波数成分ベクトル $A[]$ の値から、もとのアナログ信号ベクトル $a[]$ の値を、この回転行列式 $W[][]$ の逆行列 $invW[][]$ を求めることにより、次式の様に求められます。

$$a[n] = \sum_{m=1}^N invW[n][m]A[m] \dots \dots (3)$$

ここで、この逆回転因子行列 $invW[][]$ の各要素値 $invW[n][m]$ は次式で与えられます。

$$invW[n][m] = \exp(2\pi nmj/N) \dots \dots (4)$$

3. 2点離散フーリエ変換回路 DFT2()の定義

まず2点離散フーリエ変換の場合、 $N=2$ としますと、演算回路 DFT2()を次式で定義できます。

$$A[1] = (1/2) \{ -a[1] + a[2] \} \dots \dots (5)$$

$$A[2] = (1/2) \{ a[1] + a[2] \} \dots \dots (6)$$

この単純な DFT2()回路は、図 1 の様に、加算回路 ADD() 1 個と減算回路 SUB() 1 個と 2 個の

1 bit Shift 回路 Half()で構成されませぬ。

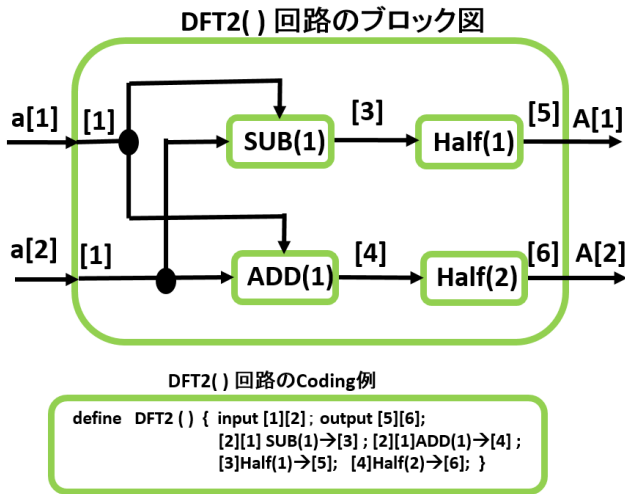


図1 2点入力 DFT 演算回路 DFT2() の定義

4. DFT4() 回路の定義

4点離散フーリエ変換回路 DFT4() 回路は、回転因子行列の定義式(1)と(2)により、図2に示す様な 4 x 4 行列式の演算回路と定義できます。

$$\begin{bmatrix} A[1] \\ A[2] \\ A[3] \\ A[4] \end{bmatrix} = \frac{1}{4} \begin{bmatrix} -j & -1 & j & 1 \\ -1 & 1 & -1 & 1 \\ j & -1 & -j & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} a[1] \\ a[2] \\ a[3] \\ a[4] \end{bmatrix}$$

図2 4 x 4 の回転因子行列を使った 4点入力の DFT4() 演算回路の定義

図3に示す様に、DFT4() 回路は、DFT2() 回路を4つコピーした DFT2(1)回路、DFT2(2)回路、DFT2(3)回路、DFT2(4)回路を基本回路 BLOCK として構成できます。DFT4() 回路はこの行列演算を実行する複素数演算回路です。ここで、回転因子ベクトル W[] と Bo[] ベクトルは次式で定義しています。For m=1→N の値で、

$$W[m] = \exp\left(-\frac{2\pi jm}{N}\right) \dots (7)$$

$$Bo[m] = W[m]Ao[m] = \exp\left(-\frac{2\pi jm}{N}\right) Ao[m] \dots (8)$$

ここで、WN() 回路を、この N 個の複素数のかけ算を実行する演算回路と定義します。しかし、実際には、

奇数項のみに適応されるもので、DFT4() 回路の場合は、W[1]=-j と W[2]=1 のみが、図3の DFT4() 回路の中の複素数かけ算回路 W2(1) 回路の入力となります。実際には、Bo[2]=Ao[2] のままで、Bo[1]= -jAo[1] となります。

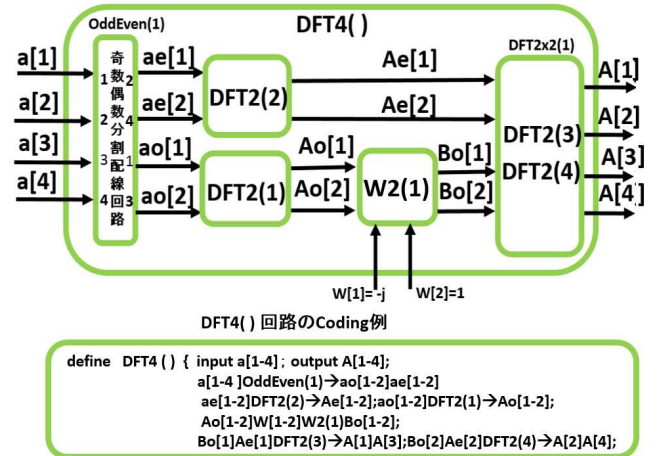


図3 4点入力 DFT 演算回路 DFT4() 回路

実際には、次の関係が成り立ちます。

$$Ae[1] = (1/2) \{ -ae[1] + ae[2] \} \dots (9)$$

$$Ae[2] = (1/2) \{ ae[1] + ae[2] \} \dots (10)$$

$$Ao[1] = (1/2) \{ -ao[1] + ao[2] \} \dots (11)$$

$$Ao[2] = (1/2) \{ ao[1] + ao[2] \} \dots (12)$$

すなわち、ae[1]=a[2], ae[2]=a[4], ao[1]=a[1], ao[2]=a[3] であるので、次式を得ます。

$$Ae[1] = (1/2) \{ -a[2] + a[4] \} \dots (13)$$

$$Ae[2] = (1/2) \{ a[2] + a[4] \} \dots (14)$$

$$Ao[1] = (1/2) \{ -a[1] + a[3] \} \dots (15)$$

$$Ao[2] = (1/2) \{ a[1] + a[3] \} \dots (16)$$

また、図2に示した回転因子行列の値により、次式の関係を得ます。

$$A[1] = (1/2) \{ -Bo[1] + Ae[1] \} \dots (17)$$

$$A[3] = (1/2) \{ Bo[1] + Ae[1] \} \dots (18)$$

$$A[2] = (1/2) \{ -Bo[2] + Ae[2] \} \dots (19)$$

$$A[4] = (1/2) \{ Bo[2] + Ae[2] \} \dots (20)$$

従って、図3において、DFT2(3)回路は式(17)と(18)を演算実行する演算回路と考えられます。また、DFT2(4)回路は式(19)と(20)を演算実行する演算回路と考えられます。

5. 奇数偶数分割法による DFT2N()回路の定義

同様に、8点離散フーリエ変換回路 DFT8()回路、16点離散フーリエ変換回路 DFT16()回路と、順次に設計構築していきたいところです。実際、64点の離散フーリエ変換 (DFT) の場合は、すでに実用化され、既に IEEE802.11a/g/n 等の Wireless LAN にも応用されています。現在 64bit のパソコンが主流ですが、2020 年には 128bit パソコンの出現が期待され、離散フーリエ変換回路も DFT128() の設計構築されることになると期待します。

そこで、奇数偶数分割法という方法を使って、DFT 演算回路のサンプリング数を2のべき乗の数 2^l として、再帰的(recursive)に階層定義することにします。

たとえば、 $2^l=2N$ 点の実時間の入力信号ベクトル $a[]$ を処理する DFT2N()回路を考えます。まず、この $2N$ 個のサンプリング点を持つ入力ベクトル $a[]$ を、偶数ベクトル $ae[]$ と奇数ベクトル $ao[]$ に2分割します。

そして、それぞれを、図4の様に2つの DFTN()回路で処理します。N bit の2つ DFTN()回路で演算処理し、それぞれ、N bit の周波数成分ベクトル $Ae[]$ と $Ao[]$ を求めます。

さらにその2つのベクトルをN個の DFT2()回路を使って、最終的に2N個の要素値を持つ $A[]$ ベクトルを演算処理して求める回路になります。ここでN個の DFT2()回路の前段で $Ao[]$ 信号には回転因子ベクトルの重みづけ演算 $WN()$ 回路が必要となります。

実際には、回転行列式の定義(1)と(2)から、導入計算は複雑ですが、最終的に次の関係が成り立ちます。

For $m=1 \rightarrow N$ の値で、

$$A[m] = \left(\frac{1}{2}\right) \left\{ Ae[m] + \exp\left(-\frac{2\pi jm}{N}\right) Ao[m] \right\} \dots (21)$$

$$A[m+N] = \left(\frac{1}{2}\right) \left\{ -Ae[m] + \exp\left(-\frac{2\pi jm}{N}\right) Ao[m] \right\} \dots (22)$$

ここで $Ao[m]$ と $Ae[m]$ を次式で定義しています。

$$Ao[m] = \left(\frac{1}{N}\right) \sum_{n=1}^N \exp\left(-\frac{2nm\pi j}{N}\right) a[2n-1] \dots (23)$$

$$Ae[m] = \left(\frac{1}{N}\right) \sum_{n=1}^N \exp\left(-\frac{2nm\pi j}{N}\right) a[2n] \dots (24)$$

式(23)と式(24)の演算は、DFTN()回路を2つ使うことにより演算処理が可能です。

また、式(22)と式(1)の演算は、DFT2()回路をN個用意することにより演算実行が可能です。

図4ではN個の DFT2()回路をまとめて DFT2xN()回路として表記しています。

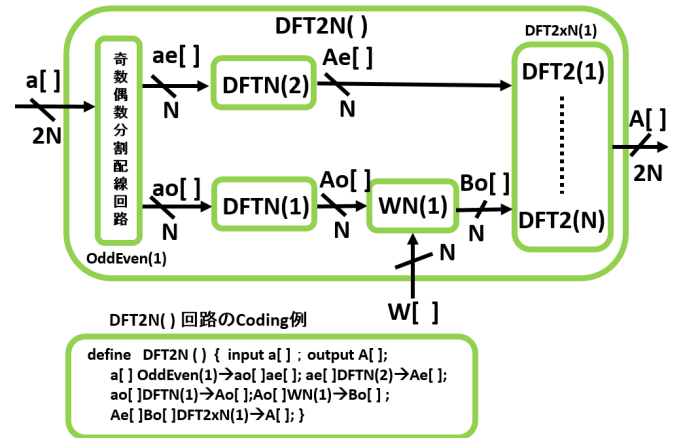


図4 偶数奇数2分割 DFT2N()演算回路アーキテクチャ

ここで、回転因子ベクトル $W[]$ と $Bo[]$ ベクトルは式(7)と(8)で定義しています。

$WN()$ 回路はN個の複素数のかけ算演算回路です。離散フーリエ変換回路の最大の問題点はこの複素数のかけ算回路 $WN()$ 回路の存在です。まずこの回転因子ベクトル $W[]$ の値を、特別に ROM の形などにして保管しておき、それを読み出しかけ算する必要があります。

6. 非等間隔での Sampling 手法の提案

従来の DFT 変換は等間隔の離散時間点でのアナログ入力信号の sampling を行います。ここで発想を転換して、等間隔で Sampling しなくても、つまり非等間隔の Sampling でも、有効な周波数成分ベクトル $A[]$ は求められないかと考えてみます。

そこでまず、sampling 開始時間 $t \sim 0$ の近辺に多くの Sampling する離散時間点を持たせてみます。

- (1) まず、アナログ入力信号の高速 sampling を可能にするパルス(sampling pulse)を発生する回路が必要になります。
- (2) 次に当然、高速にアナログ入力信号を A/D 変換して 2 値ベクトル情報に変換する回路や、
- (3) その前処理をするデジタル回路に、
- (4) 重要な情報を抽出する回路と、
- (5) その抽出情報を判定する回路や、
- (6) 出力保存する回路などが必要となります。

そこで、非等間隔 sampling 手法は自由度が大きすぎますので、単純な非等間隔 sampling 手法として、時間区間 $[0, T]$ の間で、離散時間点ベクトル $t[]$ を次の簡単な式で定義される場合における非等間隔 sampling 手法を考えてみます。n の値は 1 から N 迄の整数とします。

$$t[n] = \frac{T}{(N+1-n)} \quad \text{for } n = 1 \rightarrow N \quad \dots(25)$$

図 5 は N=8 の場合の sampling の timing を示す図です。時間区間 $[0, T]$ の間で、合計 N 個の離散時間点アナログ入力信号を sampling することにします。

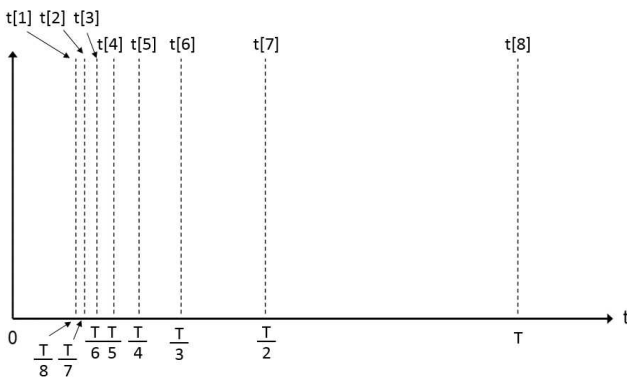


図 5 非等間隔 sampling 手法 (N=8)

これでも $t = 0$ に近い時刻で、比較的多数のアナログ入力信号 $a(t)$ を sampling しています。この手法を適当に $t=0$ 付近高密度 sampling 手法と呼ぶことにします。システム

が sampling を開始しなさいと命令した時刻を $t=0$ としています。

たとえば、標本点が $N=8$ の場合は、離散時間点ベクトル $t[]$ の値は以下に様になります。

$$t[] = \left\{ \frac{T}{8}, \frac{T}{7}, \frac{T}{6}, \frac{T}{5}, \frac{T}{4}, \frac{T}{3}, \frac{T}{2}, T \right\} \dots(26)$$

ここで、例えば、 $t[7] - t[8] = \left(\frac{T}{56}\right)$ となります。

7. 非等間隔 Sampling 手法での周波数変換行列 $H[][]$ の定義

この非等間隔 sampling 手法での周波数変換行列 $H[][]$ が実際に存在します。その値を図 6 に示します。

$N=2 \quad a[] = H2[][] A[]$ $\begin{bmatrix} a[1] \\ a[2] \end{bmatrix} = \begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} A[1] \\ A[2] \end{bmatrix}$	$N=2 \quad A[] = \text{inv}H2[][] a[]$ $\begin{bmatrix} A[1] \\ A[2] \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} a[1] \\ a[2] \end{bmatrix}$
$N=3 \quad a[] = H3[][] A[]$ $\begin{bmatrix} a[1] \\ a[2] \\ a[3] \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ -1 & 1 & -1 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} A[1] \\ A[2] \\ A[3] \end{bmatrix}$	$N=3 \quad A[] = \text{inv}H3[][] a[]$ $\begin{bmatrix} A[1] \\ A[2] \\ A[3] \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & 0 & 2 \\ 1 & 1 & 0 \\ 1 & -1 & -2 \end{bmatrix} \begin{bmatrix} a[1] \\ a[2] \\ a[3] \end{bmatrix}$
$N=4 \quad a[] = H4[][] A[]$ $\begin{bmatrix} a[1] \\ a[2] \\ a[3] \\ a[4] \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ -1 & 1 & -1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} A[1] \\ A[2] \\ A[3] \\ A[4] \end{bmatrix}$	$N=4 \quad A[] = \text{inv}H4[][] a[]$ $\begin{bmatrix} A[1] \\ A[2] \\ A[3] \\ A[4] \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 1 & 1 & 0 & 2 \\ 0 & 0 & 4 & 0 \\ 1 & 1 & 0 & -2 \\ 2 & -2 & 4 & 0 \end{bmatrix} \begin{bmatrix} a[1] \\ a[2] \\ a[3] \\ a[4] \end{bmatrix}$
$N=8 \quad a[] = H8[][] A[]$ $\begin{bmatrix} a[1] \\ a[2] \\ a[3] \\ a[4] \\ a[5] \\ a[6] \\ a[7] \\ a[8] \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & -1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} A[1] \\ A[2] \\ A[3] \\ A[4] \\ A[5] \\ A[6] \\ A[7] \\ A[8] \end{bmatrix}$	$N=8 \quad A[] = \text{inv}H8[][] a[]$ $\begin{bmatrix} A[1] \\ A[2] \\ A[3] \\ A[4] \\ A[5] \\ A[6] \\ A[7] \\ A[8] \end{bmatrix} = \frac{1}{8} \begin{bmatrix} 1 & 1 & 0 & 2 & 0 & 0 & 0 & 4 \\ 0 & 0 & 0 & 0 & 0 & 0 & 8 & 0 \\ 0 & 0 & 4 & 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 8 & 0 & 0 & 0 \\ 1 & 1 & 0 & 2 & 0 & 0 & -4 & -4 \\ 0 & 0 & 4 & 0 & -4 & 0 & -4 & 0 \\ 2 & 2 & -4 & -4 & 0 & -4 & 0 & 0 \\ 4 & -4 & -4 & 0 & -8 & 4 & -8 & 0 \end{bmatrix} \begin{bmatrix} a[1] \\ a[2] \\ a[3] \\ a[4] \\ a[5] \\ a[6] \\ a[7] \\ a[8] \end{bmatrix}$

図 6 行列式 $H[][]$ と逆行列 $\text{inv}H[][]$

従来の DFT 変換の $W[][]$ 行列、すなわち

$W[m][n] = \left(\frac{1}{N}\right) \exp(-2\pi mnj/N)$ に対応するのが、 $\text{invH}[] []$ 行列になります。

すなわち、 $A[] = W[] [] a[]$ に対応して $A[] = \text{invH}[] [] a[]$ となります。

また、 $\text{invW}[n][m] = \exp(2\pi mnj/N)$ に対応するのが、 $H[] []$ になります。

すなわち、 $a[] = \text{invW}[] [] A[]$ に対応して $a[] = H[] [] A[]$ となります。

$A[]$ 従来の DFT 変換では、最終的に、

$$a(t) \sim b(t) = \sum_{m=1}^N A[m] \exp\left(\frac{2\pi mtj}{T}\right) \dots (27)$$

の形で、アナログ入力関数 $a(t)$ を近似関数 $b(t)$ で近似できるものとします。

8. 非等間隔 Sampling 手法での

基本構成関数 $h_m(t)$ の定義

そこで同様に、ゼロ近辺高密度 sampling 手法でも、 $b(t) = \sum_{m=1}^N A[m] h_m(t)$ の形で、アナログ入力関数 $a(t)$ を近似できる関数群 $h_m(t)$ なるものを考えます。 $a[] = H[] [] A[]$ に対応させます。この基本構成関数 $h_m(t)$ の値は次式で定義できます。その値は、単純に -1 か 0 か 1 の pulse 周期関数 になります。

【1】 $\left(\frac{T}{m}\right)$ を周期として繰り返す関数であり、

【2】 $0 \leq t \leq \left(\frac{T}{m}\right)$ においては、

$t=0$ 及び $t = \left(\frac{T}{m}\right)$ の近傍において 1 の値を取る。

【3】 $t = \left(\frac{T}{2m}\right)$ の近傍において -1 の値となり、

【4】 他の t の値については $h_m(t) = 0$ となります。

図 7 に関数 $h_m(t)$ を定義する図を示します。

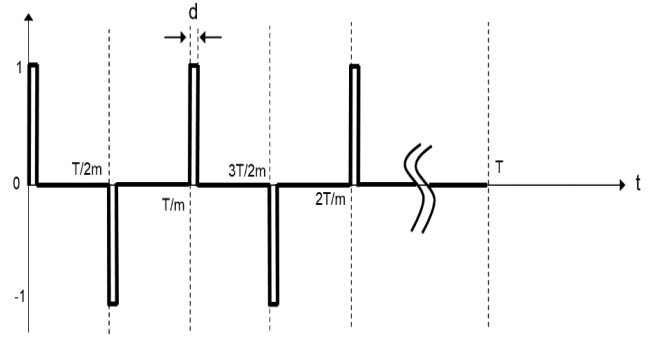


図 7 関数 $h_m(t)$ の定義図

図 6 においては、 $h_m(t) = 1$ または $h_m(t) = -1$ となる箇所に d だけの幅 (t 軸方向の幅) を持たせていますが、幅がないものと考えていることが正しいです。 $d \rightarrow 0$ の極限が $h_m(t)$ を正確に表すものとします。

図 8 に $N=8$ の場合の $t=t[n]$ の場合に求まる関数 $h_m(t)$ の値、すなわち、 $H[n][m] = h_m(t[n])$ の値を示します。

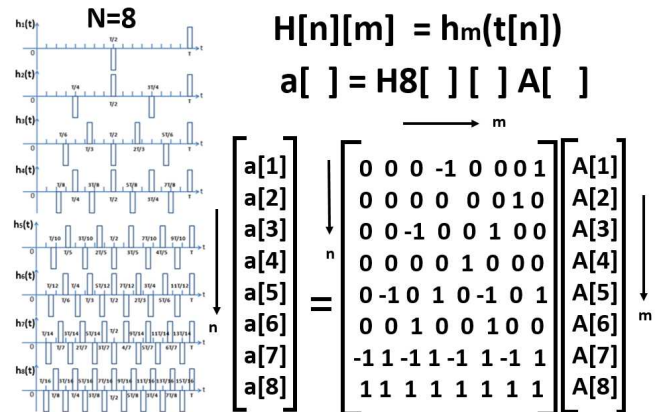


図 8 $N=8$ の場合の $H[n][m] = h_m(t[n])$ の値

DFT 変換やコサイン変換やサイン変換の場合、必ず三角関数の計算が必要でその演算処理時間は無視できません。できれば ROM 回路も不要な簡単な演算処理で済めば最高です。

図 9 は、時間領域 data を周波数領域 data に変換する演算回路の構成図です。

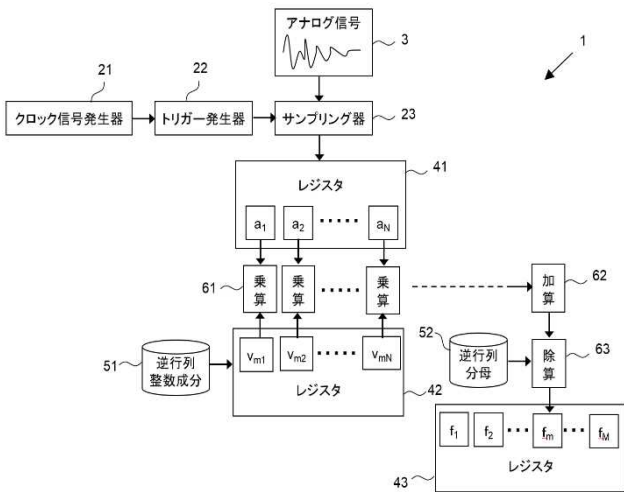


図9 時間領域 data を周波数領域 data に変換する演算回路の構成図

また図10は、sampling回路を示す図です。

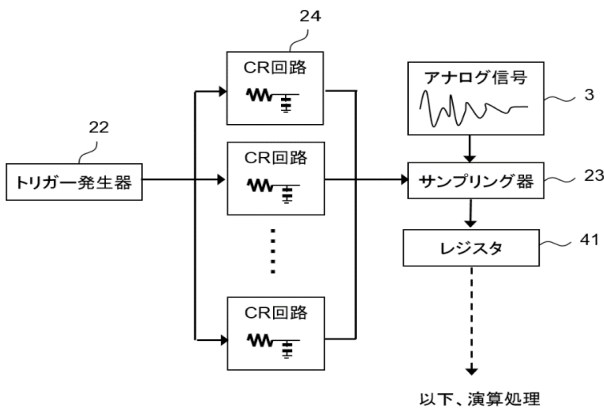


図10 sampling回路のblock図

図9の中での各部品番号の説明は以下のようになります。

- 1 = 時間領域 data を周波数領域 data に変換する演算回路全体
- 3 = analog 信号
- 21= clock 信号発生器
- 22= trigger 発生器
- 23= sampler
- 24= CR 回路
- 41= 時間領域 data 保存 register
- 42= 行列演算中間ベクトル data $V_m[]$ の一時保存用 register
- 43= 周波数領域ベクトル data $f[] (=A[])$ の一時保存用 register

- 51= 逆行列 $invH[][]$ の要素数の分子 (整数成分) の記憶用回路
- 52= 逆行列 $invH[][]$ の要素数の共通分母の記憶用回路
- 53= 乗算器
- 54= 加算器
- 55= 除算器

実際に $a(t)=\cos(4\pi t/T)$ の場合で、従来の DFT 変換で求めて周波数ベクトル $A[]$ と今回提案の $t=0$ 付近高密度 sampling 方式で求めたベクトル $A[]$ の値を図11と図12に示します。

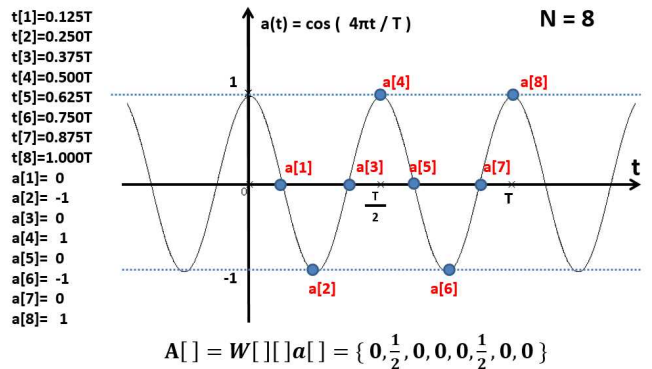


図11 従来の DFT 変換の場合

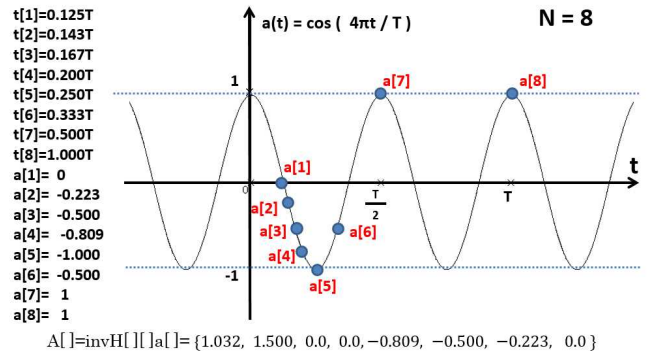


図12 t=0 付近高密度 sampling 方式

離散フーリエ変換は元来 cosine 関数等の周波数成分を求めるのに最適ですが、図12の結果ではかなりの aliasing が目立ちます。

文献

- [1] 64 Point FFT Design Manual, 64 点高速フーリエ変換回路設計仕様書, 和田知久, Design Wave Magazine 2006 年 11 月号, pp.143-156
- [2] 数値計算法, 三井田惇郎, 須田 宇宙共著, 森北出版, 第 8 章離散フーリエ変換, pp.96-110.