



www.fee.bzh

White paper For Engineer Eyes

FEE-WP-012Bf – Dec.2021–Mai.2024

Auteur: J. Pontois

DFTs & FFTs

SOMMAIRE

1.	TRANSFORMEE DE FOURIER DISCRETE.....	3
1.1.	DEFINITION	4
1.2.	CALCUL DE QUELQUES DFT ELEMENTAIRES	4
1.2.1.	<i>Multiplication complexe</i>	4
1.2.2.	<i>DFT de 2 points</i>	5
1.2.3.	<i>DFT de 3 points</i>	5
1.2.4.	<i>DFT de 4 points</i>	6
1.2.5.	<i>DFT de 5 points</i>	6
2.	PRINCIPAUX ALGORITHMES DE TRANSFORMEE DE FOURIER RAPIDE	9
2.1.	ALGORITHME DE COOLEY-TUKEY	9
2.2.	ALGORITHME DE GOOD-THOMAS.....	13
2.3.	CONSIDERATIONS SUR LA DYNAMIQUE	15

1. OBJET

Ce document présente l'implémentation de transformées de Fourier discrètes.

Note : document interne 010-NT-001A de Mars 2013, passage initial en « White Paper » en décembre 2021.

Ce document de FEE est fourni pour information, sans aucune garantie. Sa copie partielle n'est pas autorisée.

2. TRANSFORMEE DE FOURIER DISCRETE

2.1. DEFINITION

La Transformée de Fourier Discrète (TFD ou DFT pour Discrete Fourier Transform) de longueur N fournit le spectre en fréquence d'un signal discret x_n , considéré périodique de période N (ou de fréquence $\frac{f_e}{N}$).

Soit le coefficient $W_N = e^{j\frac{2\pi}{N}}$.

On définit le spectre discret (X_k) du signal (x_n) par la relation :

$$X_k = \sum_{n=0}^{N-1} x_n e^{-j\frac{2\pi nk}{N}} = \sum_{n=0}^{N-1} x_n W_N^{-nk}, \text{ pour } k = 0..N-1$$

X_k représente la valeur de la composante de (x_n) pour la fréquence $k\frac{f_e}{N}$, obtenue par la transformée directe.

La transformée inverse s'écrit :

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k e^{j\frac{2\pi nk}{N}} = \frac{1}{N} \sum_{k=0}^{N-1} X_k W_N^{nk}, \text{ pour } n = 0..N-1$$

2.2. CALCUL DE QUELQUES DFT ELEMENTAIRES

On optimise ici quelques DFT complexes élémentaires courantes ($N = 2, 3, 4, 5$) pour une utilisation dans les algorithmes précédents, écrites de manière à minimiser le nombre de multiplications réelles à leur calcul (qui est le facteur limitant pour une implémentation en FPGA).

L'entrée de la DFT est noté (x_n) pour $n = 0..N-1$

La sortie de la DFT est noté (X_k) pour $k = 0..N-1$

On pose par définition : $X_k = \sum_{n=0}^{N-1} x_n W_N^{-nk}$, avec $W_N = e^{j\frac{2\pi}{N}}$

2.2.1. MULTIPLICATION COMPLEXE

En marge du calcul des DFT proprement dit, on donne ici une écriture de la multiplication complexe (qui utilise basiquement 4 multiplications et 2 additions réelles) en une expression utilisant 3 multiplications et 5 additions.

On cherche à effectuer la multiplication $z = (a + jb) \cdot (c + js)$, a, b, c, s réels.

$$\text{En posant } z = x + jy, \text{ on a : } \begin{cases} x = a \cdot c - b \cdot s \\ y = a \cdot s + b \cdot c \end{cases}$$

dont le calcul direct s'effectue en deux étapes : 4 multiplications suivies de 2 additions.

On peut réécrire cette expression selon par exemple :

$$\begin{cases} x = (a+b) \cdot c - b \cdot (c+s) \\ y = (a+b) \cdot c - a \cdot (c-s) \end{cases}$$

Le calcul du produit complexe s'effectue alors en trois étapes :

- 3 additions : $a + b$, $c + s$, $c - s$
- 3 multiplications : $(a+b) \cdot c$, $b \cdot (c+s)$, $a \cdot (c-s)$
- 2 additions

L'implémentation du multiplieur complexe ne nécessite plus que 3 multiplications. Néanmoins, la représentation des multiplicandes nécessite un bit de plus que dans le cas basique pour garder la même précision (si a et b sont codés sur p bits, $a+b$ est codé sur $p+1$ bits).

2.2.2. DFT DE 2 POINTS

Le calcul de la DFT pour $N = 2$ est direct et ne nécessite que deux additions :

$$\begin{cases} X_0 = x_0 + x_1 \\ X_1 = x_0 - x_1 \end{cases}$$

A la normalisation par 2 près, la DFT inverse est égale à la DFT.

2.2.3. DFT DE 3 POINTS

Le calcul de la DFT pour $N = 3$ s'écrit :

$$\begin{cases} X_0 = x_0 + x_1 + x_2 \\ X_1 = x_0 + W_3^{-1}x_1 + W_3^{-2}x_2 \\ X_2 = x_0 + W_3^{-2}x_1 + W_3^{-1}x_2 \end{cases}$$

Sous cette forme, la DFT nécessite 4 multiplications complexes et 6 additions complexes.

En écrivant $W_3 = c_1 + js_1$ avec
$$\begin{cases} c_1 = \cos\left(\frac{2\pi}{3}\right) = -\frac{1}{2} \\ s_1 = \sin\left(\frac{2\pi}{3}\right) = \frac{\sqrt{3}}{2} \approx 0.8660254038... \end{cases}$$

et en remarquant que $W_3^2 = (W_3^1)^*$, on peut écrire :

$$\begin{cases} X_0 = x_0 + x_1 + x_2 \\ X_1 = x_0 + c_1(x_1 + x_2) - js_1(x_1 - x_2) \\ X_2 = x_0 + c_1(x_1 + x_2) + js_1(x_1 - x_2) \end{cases}$$

En posant $\begin{cases} P_0 = x_1 + x_2 \\ P_1 = x_1 - x_2 \end{cases}$, ce qui nécessite 2 additions complexes, on obtient :

$$\begin{cases} X_0 = x_0 + P_0 \\ X_1 = (x_0 + c_1P_0) - js_1P_1 \\ X_2 = (x_0 + c_1P_0) + js_1P_1 \end{cases}$$

En posant $\begin{cases} R_0 = x_0 + c_1P_0 \\ R_1 = s_1P_1 \end{cases}$, ce qui nécessite 4 multiplications réelles et 1 addition complexe, on

obtient finalement :

$$\begin{cases} X_0 = x_0 + P_0 \\ X_1 = R_0 - jR_1 \\ X_2 = R_0 + jR_1 \end{cases}$$

Le calcul de la DFT demande donc encore 3 additions complexes (les multiplications par j sont transparentes), soit au total :

- 4 multiplications réelles (dont deux multiplications par $c_1 = -\frac{1}{2}$ qui sont implémentées trivialement en base 2, soit 2 multiplications non triviales),
- 6 additions complexes (12 additions réelles)

A la normalisation par 3 près, la DFT inverse s'obtient par $j \rightarrow -j$:

$$\begin{cases} X_0 = x_0 + P_0 \\ X_1 = R_0 + jR_1 \\ X_2 = R_0 - jR_1 \end{cases}$$

2.2.4. DFT DE 4 POINTS

Le calcul de la DFT pour $N = 4$ s'écrit :

$$\begin{cases} X_0 = x_0 + x_1 + x_2 + x_3 \\ X_1 = x_0 - jx_1 - x_2 + jx_3 \\ X_2 = x_0 - x_1 + x_2 - x_3 \\ X_3 = x_0 + jx_1 - x_2 - jx_3 \end{cases}$$

Sous cette forme, la DFT nécessite 12 additions complexes et 2 multiplications triviales (par j), réalisées par 24 additions réelles.

En posant $\begin{cases} P_0 = x_0 + x_2 \\ P_1 = x_0 - x_2 \\ P_2 = x_1 + x_3 \\ P_3 = x_1 - x_3 \end{cases}$, on obtient $\begin{cases} X_0 = P_0 + P_2 \\ X_1 = P_1 - jP_3 \\ X_2 = P_0 - P_2 \\ X_3 = P_1 + jP_3 \end{cases}$

Le calcul de la DFT se fait donc en deux étapes :

- 4 additions complexes
- 1 multiplication par j et 4 additions complexes réalisées par 8 additions réelles (la multiplication par j est triviale, elle est cachée dans les additions réelles).

Au total, la DFT d'ordre 4 nécessite 16 additions réelles.

A la normalisation par 4 près, la DFT inverse s'obtient par $j \rightarrow -j$:

$$\begin{cases} X_0 = P_0 + P_2 \\ X_1 = P_1 + jP_3 \\ X_2 = P_0 - P_2 \\ X_3 = P_1 - jP_3 \end{cases}$$

2.2.5. DFT DE 5 POINTS

Le calcul de la DFT pour $N = 5$ s'écrit :

$$\begin{cases} X_0 = x_0 + x_1 + x_2 + x_3 + x_4 \\ X_1 = x_0 + W_5^{-1}x_1 + W_5^{-2}x_2 + W_5^{-3}x_3 + W_5^{-4}x_4 \\ X_2 = x_0 + W_5^{-2}x_1 + W_5^{-4}x_2 + W_5^{-1}x_3 + W_5^{-3}x_4 \\ X_3 = x_0 + W_5^{-3}x_1 + W_5^{-1}x_2 + W_5^{-4}x_3 + W_5^{-2}x_4 \\ X_4 = x_0 + W_5^{-4}x_1 + W_5^{-3}x_2 + W_5^{-2}x_3 + W_5^{-1}x_4 \end{cases}$$

Sous cette forme, la DFT nécessite 16 multiplications complexes et 20 additions complexes.

$$\text{En écrivant } W_5 = c_1 + js_1 \text{ avec } \begin{cases} c_1 = \cos\left(\frac{2\pi}{5}\right) \\ s_1 = \sin\left(\frac{2\pi}{5}\right) \end{cases}, W_5^2 = c_2 + js_2 \text{ avec } \begin{cases} c_2 = \cos\left(\frac{4\pi}{5}\right) \\ s_2 = \sin\left(\frac{4\pi}{5}\right) \end{cases}$$

et en remarquant que $W_5^3 = (W_5^2)^*$ et $W_5^4 = (W_5^1)^*$, on peut écrire :

$$\begin{cases} X_0 = x_0 + x_1 + x_2 + x_3 + x_4 \\ X_1 = x_0 + c_1(x_1 + x_4) - js_1(x_1 - x_4) + c_2(x_2 + x_3) - js_2(x_2 - x_3) \\ X_2 = x_0 + c_2(x_1 + x_4) - js_2(x_1 - x_4) + c_1(x_2 + x_3) + js_1(x_2 - x_3) \\ X_3 = x_0 + c_2(x_1 + x_4) + js_2(x_1 - x_4) + c_1(x_2 + x_3) - js_1(x_2 - x_3) \\ X_4 = x_0 + c_1(x_1 + x_4) + js_1(x_1 - x_4) + c_2(x_2 + x_3) + js_2(x_2 - x_3) \end{cases}$$

$$\text{En posant } \begin{cases} P_0 = x_1 + x_4 \\ P_1 = x_1 - x_4 \\ P_2 = x_2 + x_3 \\ P_3 = x_2 - x_3 \end{cases}, \text{ ce qui nécessite 4 additions complexes, on obtient :}$$

$$\begin{cases} X_0 = x_0 + P_0 + P_2 \\ X_1 = x_0 + c_1P_0 - js_1P_1 + c_2P_2 - js_2P_3 \\ X_2 = x_0 + c_2P_0 - js_2P_1 + c_1P_2 + js_1P_3 \\ X_3 = x_0 + c_2P_0 + js_2P_1 + c_1P_2 - js_1P_3 \\ X_4 = x_0 + c_1P_0 + js_1P_1 + c_2P_2 + js_2P_3 \end{cases}$$

$$\text{En posant } \begin{cases} Q_0 = P_0 + P_2 \\ Q_1 = P_0 - P_2 \\ Q_2 = P_1 + P_3 \end{cases}, \text{ ce qui nécessite 3 additions complexes, et en définissant}$$

$$\left\{ \begin{array}{l} q_0 = \frac{c_1 + c_2}{2} = \frac{\cos\left(\frac{2\pi}{5}\right) + \cos\left(\frac{4\pi}{5}\right)}{2} = -\frac{1}{4} \\ q_1 = \frac{c_1 - c_2}{2} = \frac{\cos\left(\frac{2\pi}{5}\right) - \cos\left(\frac{4\pi}{5}\right)}{2} = \frac{\sqrt{5}}{4} \approx 0.559016994... \\ q_2 = s_1 + s_2 = \sin\left(\frac{2\pi}{5}\right) + \sin\left(\frac{4\pi}{5}\right) = \frac{\sqrt{5+2\sqrt{5}}}{2} \approx 1.538841769... \\ q_3 = s_1 - s_2 = \sin\left(\frac{2\pi}{5}\right) - \sin\left(\frac{4\pi}{5}\right) = \frac{\sqrt{5-2\sqrt{5}}}{2} \approx 0.363271264... \end{array} \right. ,$$

on obtient :

$$\left\{ \begin{array}{l} X_0 = x_0 + Q_0 \\ X_1 = x_0 + q_0 Q_0 + q_1 Q_1 - js_1 Q_2 + jq_3 P_3 \\ X_2 = x_0 + q_0 Q_0 - q_1 Q_1 + js_1 Q_2 - jq_2 P_1 \\ X_3 = x_0 + q_0 Q_0 - q_1 Q_1 - js_1 Q_2 + jq_2 P_1 \\ X_4 = x_0 + q_0 Q_0 + q_1 Q_1 + js_1 Q_2 - jq_3 P_3 \end{array} \right.$$

En posant $\left\{ \begin{array}{l} R_0 = (x_0 + q_0 Q_0) + q_1 Q_1 \\ R_1 = (x_0 + q_0 Q_0) - q_1 Q_1 \\ R_2 = s_1 Q_2 - q_2 P_1 \\ R_3 = s_1 Q_2 - q_3 P_3 \end{array} \right.$, ce qui nécessite 10 multiplications réelles et 5 additions complexes, on

obtient finalement :

$$\left\{ \begin{array}{l} X_0 = x_0 + Q_0 \\ X_1 = R_0 - jR_3 \\ X_2 = R_1 + jR_2 \\ X_3 = R_1 - jR_2 \\ X_4 = R_0 + jR_3 \end{array} \right.$$

Le calcul de la DFT demande donc encore 5 additions complexes (les multiplications par j sont transparentes), soit au total :

- 10 multiplications réelles (dont deux multiplications par $q_0 = \frac{1}{4}$ qui sont implémentées trivialement en base 2, soit 8 multiplications non triviales),
- 17 additions complexes (34 additions réelles)

A la normalisation par 5 près, la DFT inverse s'obtient par $j \rightarrow -j$:

$$\left\{ \begin{array}{l} X_0 = x_0 + Q_0 \\ X_1 = R_0 + jR_3 \\ X_2 = R_1 - jR_2 \\ X_3 = R_1 + jR_2 \\ X_4 = R_0 - jR_3 \end{array} \right. .$$

3. PRINCIPAUX ALGORITHMES DE TRANSFORMEE DE FOURIER RAPIDE

Les techniques de Transformée de Fourier Rapide (TFR ou FFT pour Fast Fourier Transform) permettent de calculer la transformée de Fourier discrète pour des valeurs particulières de N plus efficacement que la DFT "de base", qui demande N^2 multiplications et $N(N-1)$ additions (complexes).

La transformée de Fourier discrète (X_k) d'ordre N de la séquence (x_n) s'écrit :

$$X_k = \sum_{n=0}^{N-1} x_n W_N^{-nk} \text{ pour } k = 0..N-1, \text{ avec } W_N = e^{j\frac{2\pi}{N}}$$

3.1. ALGORITHME DE COOLEY-TUKEY

On suppose que N s'écrit comme le produit de deux entiers : $N = PQ$.

On pose : $n = q + pQ$, avec $q = 0..Q-1$ et $p = 0..P-1$.

On peut alors réécrire la DFT selon :

$$X_k = \sum_{n=0}^{N-1} x_n W_N^{-nk} = \sum_{q=0}^{Q-1} \sum_{p=0}^{P-1} x_{pQ+q} W_N^{-k(pQ+q)} = \sum_{q=0}^{Q-1} \sum_{p=0}^{P-1} x_{pQ+q} W_N^{-k(pQ+q)}$$

En posant maintenant $k = p' + q'P$, avec $q' = 0..Q-1$ et $p' = 0..P-1$, on a :

$$X_{p'+q'P} = \sum_{q=0}^{Q-1} \left[W_N^{-(p'+q'P)q} \sum_{p=0}^{P-1} x_{pQ+q} W_N^{-(p'+q'P)pQ} \right] = \sum_{q=0}^{Q-1} \left[W_N^{-qq'P} W_N^{-p'q} \sum_{p=0}^{P-1} x_{pQ+q} W_N^{-pp'Q} \right],$$

soit $X_{p'+q'P} = \sum_{q=0}^{Q-1} \left[W_Q^{-qq'} W_N^{-p'q} \sum_{p=0}^{P-1} x_{pQ+q} W_P^{-pp'} \right]$, avec $W_P = e^{j\frac{2\pi}{P}}$ et $W_Q = e^{j\frac{2\pi}{Q}}$

Sous cette forme, la DFT initiale est équivalente à une ré-indexation des N points d'entrée, suivie de l'application de Q DFT de P points, de N multiplications (par $W_N^{-p'q}$), de l'application de P DFT de Q points, et enfin d'une ré-indexation des valeurs de sortie.

Exemple pour $P = 3$ et $Q = 4$:

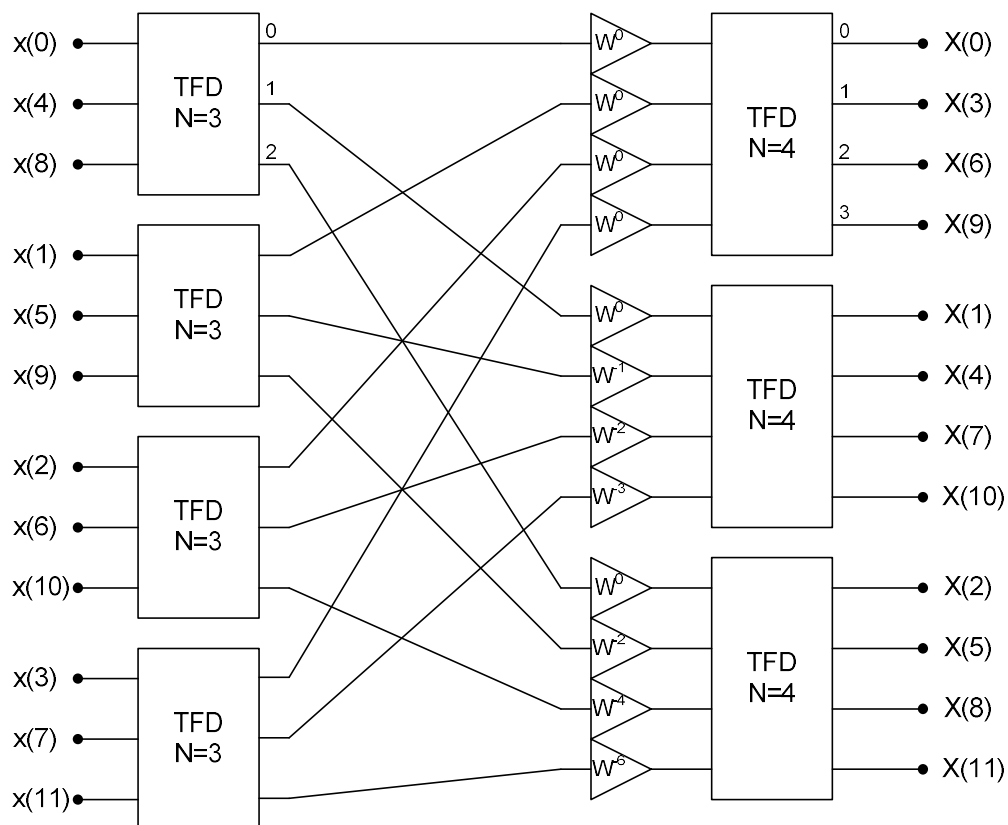


Figure 1 - FFT 12 points par Cooley-Tukey

La DFT initiale demandait $N^2 = P^2Q^2$ multiplications, la décomposition de Cooley-Tukey ramène le nombre de multiplications à $QP^2 + N + PQ^2 = N(P + Q + 1)$. Si les DFT de P et Q points peuvent elles-mêmes être décomposées en DFT plus petites, la puissance de calcul nécessaire diminue encore.

En particulier, cette méthode permet de traiter, par itérations successives, le cas $N = R^S$. Le nombre M_N de multiplications nécessaires (hors simplifications) passe alors à :

$$M_N = M_R \cdot S \cdot R^{S-1} + R^S (S - 1) = S \cdot R^S \left(1 - \frac{1}{S} + \frac{M_R}{R} \right) = N \cdot \log_R(N) \cdot \left(1 - \frac{1}{\log_R(N)} + \frac{M_R}{R} \right),$$

où M_R est le nombre de multiplications nécessaires pour effectuer la TFD élémentaire d'ordre R .

Dans le cas général d'une FFT de S étages d'ordre $N = R^S$, la ré-indexation en entrée s'écrit :

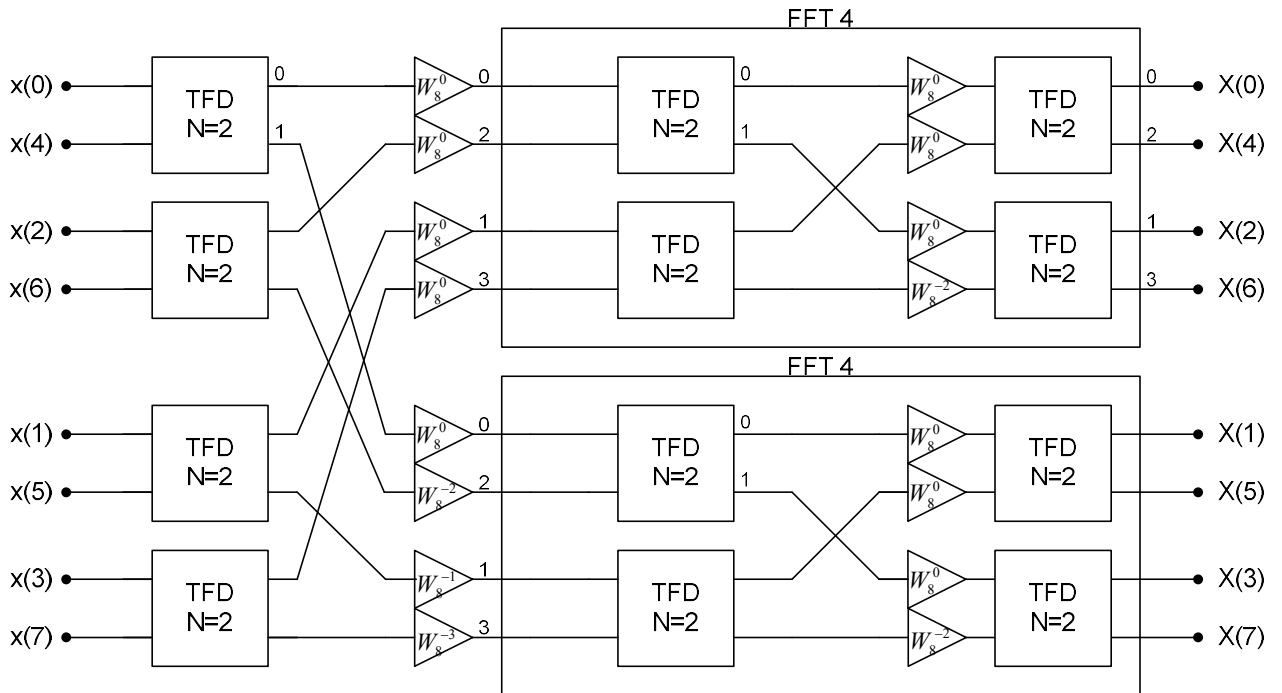
$$n = m_0 R^{S-1} + \dots + m_{S-2} R + m_{S-1}, \text{ avec } m_{S-1}, \dots, m_0 \text{ entiers dans } 0..R-1.$$

$m = m_{S-1} R^{S-1} + \dots + m_1 R + m_0$ est le numéro de l'entrée du bloc FFT. En base R , m et n s'écrivent avec les mêmes chiffres, mais en ordre inversé.

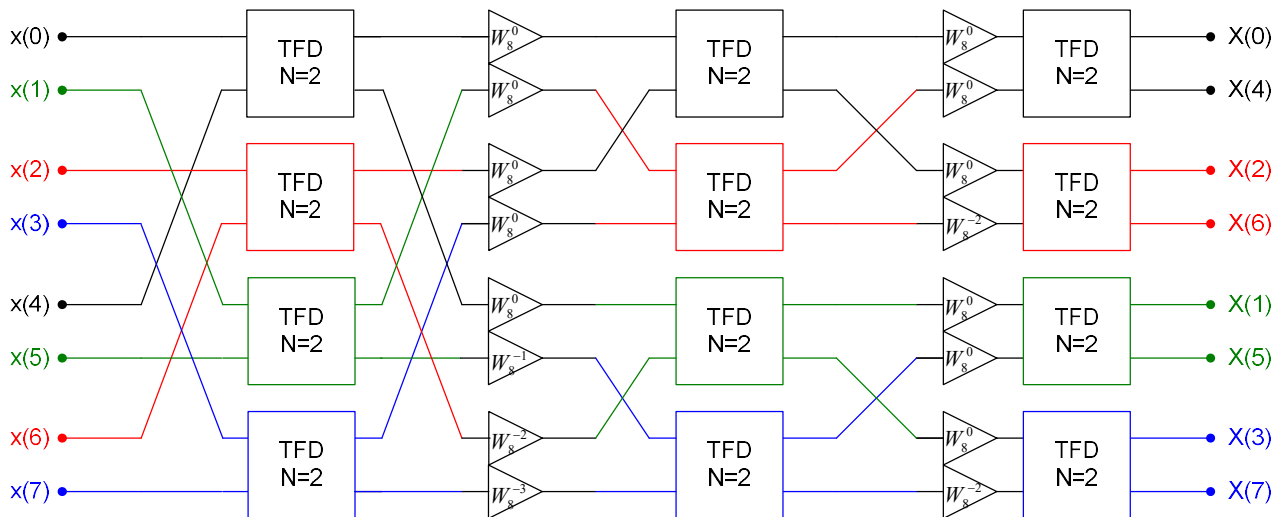
La ré-indexation en sortie s'écrit de la même manière.

Cet algorithme est à la base de la FFT « classique » pour $R = 2$ ou $R = 4$. En effet, les DFT pour $N = 2$ et $N = 4$ ne demandent aucune multiplication autre que par ± 1 ou $\pm j$: on considère que $M_2 = 0$ et $M_4 = 0$, d'où asymptotiquement $M_N \approx N \cdot \log_2(N)$ pour $R = 2$ ou $M_N \approx N \cdot \log_4(N)$ pour $R = 4$.

L'exemple ci-dessous illustre le cas d'une FFT de 8 points pour $R = 2$ et $S = 3$, décomposée en 4 DFT de 2 points suivies de 2 FFT de 4 points.



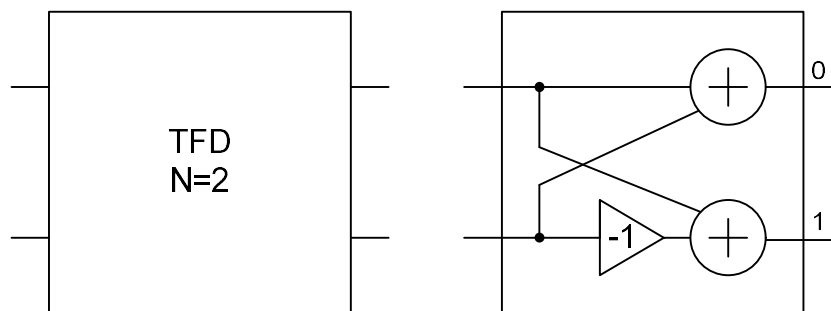
En intégrant la ré-indexation en entrée de chaque étage, on construit la structure suivante :



Cette présentation montre que l'on peut calculer la FFT étage par étage, en remplaçant à chaque fois les données entrantes par les données sortantes. On évite ainsi la ré-indexation en entrée (elle est réalisée durant la phase de calcul), mais les données de sortie sont toujours entrelacées.

Outre la structure de FFT présentée ici, due à la décomposition itérative « à gauche », par $P = R$ et $Q = R^{S-1}$, on peut aussi définir la structure symétrique (obtenue en itérant $P = R^{S-1}$ et $Q = R$), où l'on opère sur un bloc de données entrelacées pour obtenir directement le spectre. La première est nommée à entrelacement fréquentiel, la seconde à entrelacement temporel.

La DFT élémentaire d'ordre 2 se résume à une addition et une soustraction, selon le schéma suivant :



Dans cet exemple de FFT à 8 points, seules 2 multiplications complexes sont non triviales (par W_8^{-1} et W_8^{-3}). La FFT nécessite par ailleurs, dans les 12 DFT élémentaires de 2 points, 24 additions complexes.

Quand R est une puissance de 2, la ré-indexation est simple à réaliser dans un FPGA : elle consiste simplement à inverser l'ordre des bits composant les indices. Pour une taille de FFT fixée, cette opération peut se résumer en un câblage, sans consommation de ressource. En fonction du processeur utilisé, une implémentation logicielle de la ré-indexation peut demander un certain nombre d'opérations ou l'utilisation de tables pré calculées. Les processeurs dédiés au traitement de signal (DSP) intègrent la « propagation de retenue inverse » qui permet d'effectuer la ré-indexation de manière transparente.

Un dernier exemple montre la réalisation d'une FFT de 25 points à partir d'une DFT élémentaire de 5 points :

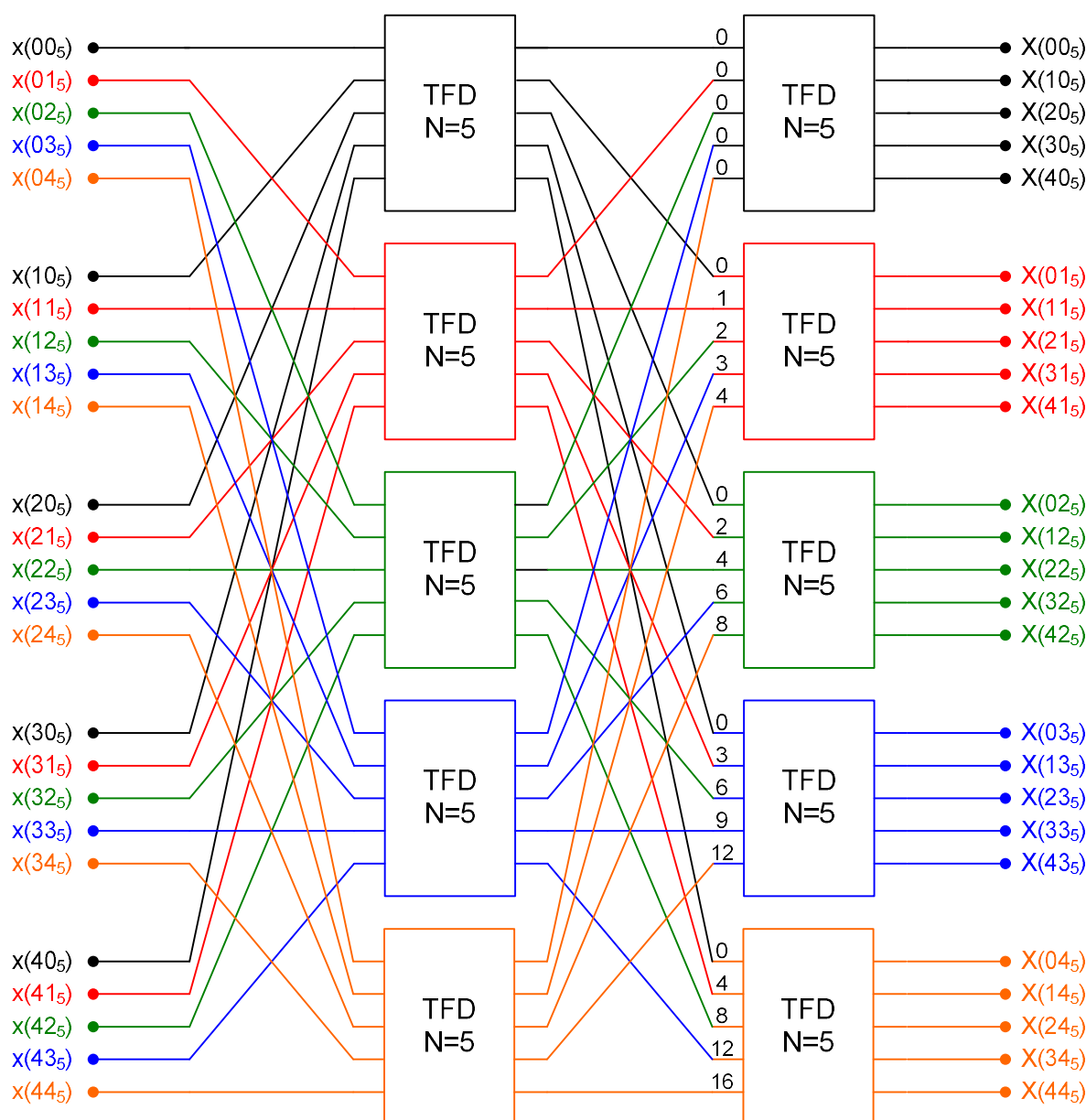


Figure 2 - FFT 25 points par Cooley-Tukey

Les indices des éléments d'entrée-sortie sont données en base 5, les nombres en entrée des DFT du second étage donne la valeur du multiplicateur à appliquer sur la branche ($m \leftrightarrow W_{25}^{-m}$). On conserve la structure vue pour la FFT d'ordre 2^S , ce qui permet d'effectuer la FFT de 25 points « sur place » en deux étapes de calcul, en réutilisant la mémoire des données d'entrée. Les données d'entrée sont fournies dans l'ordre naturel, les données de sortie sont entrelacées. Les indices entrée-sortie s'obtiennent l'un l'autre par permutation des chiffres.

3.2. ALGORITHME DE GOOD-THOMAS

On suppose que N s'écrit comme le produit de deux entiers premiers entre eux : $N = PQ$.

P et Q étant premiers entre eux, il existe p_0 et q_0 tels que $q_0P + p_0Q = 1$.

Pour $n = 0..N-1$, on définit $p = (n \cdot p_0) \bmod P$ et $q = (n \cdot q_0) \bmod Q$.

On peut écrire $n \cdot p_0 = p + P \cdot u$ et $n \cdot q_0 = q + Q \cdot v$, u et v entiers

d'où $n(p_0Q + p_0P) = n = pQ + qP + N(u + v)$

On en déduit $n = (qP + pQ) \bmod N$: à tout $n = 0..N-1$ on peut donc faire correspondre un couple unique (p, q) , avec $p = 0..P-1$ et $q = 0..Q-1$, tel que $n = (qP + pQ) \bmod N$

On peut alors réécrire la DFT selon :

$$X_k = \sum_{n=0}^{N-1} x_n W_N^{-nk} = \sum_{q=0}^{Q-1} \sum_{p=0}^{P-1} x_{(pQ+qP) \bmod N} W_N^{-k[(pQ+qP) \bmod N]} = \sum_{q=0}^{Q-1} \sum_{p=0}^{P-1} x_{(pQ+qP) \bmod N} W_N^{-k(pQ+qP)}$$

$$\text{Soit } X_k = \sum_{q=0}^{Q-1} \left[W_N^{-kqP} \sum_{p=0}^{P-1} x_{(pQ+qP) \bmod N} W_N^{-kpQ} \right] = \sum_{q=0}^{Q-1} \left[W_Q^{-kq} \sum_{p=0}^{P-1} x_{(pQ+qP) \bmod N} W_P^{-kp} \right],$$

$$\text{avec } W_P = e^{j\frac{2\pi}{P}} \text{ et } W_Q = e^{j\frac{2\pi}{Q}}$$

$$\text{On pose } y_{qP+p} = x_{(pQ+qP) \bmod N}, \text{ on a : } X_k = \sum_{q=0}^{Q-1} \left[W_Q^{-kq} \sum_{p=0}^{P-1} y_{qP+p} W_P^{-kp} \right]$$

Pour $k = 0..N-1$, on définit $p' = k \bmod P$ et $q' = k \bmod Q$.

On peut écrire $k = p' + P \cdot u$ et $k = q' + Q \cdot v$, u et v entiers, d'où :

$$p' p_0 Q + q' q_0 P = kp_0 Q - PQp_0 u + kq_0 P - PQq_0 v = k(p_0 Q + q_0 P) - N(q_0 v + p_0 u) = k - N(q_0 v + p_0 u)$$

On en déduit $k = (p' p_0 Q + q' q_0 P) \bmod N$: à tout $k = 0..N-1$ on peut donc faire correspondre un couple unique (p', q') , avec $p' = 0..P-1$ et $q' = 0..Q-1$, tel que $k = (q' q_0 P + p' p_0 Q) \bmod N$

D'où :

$$X_{(p' p_0 Q + q' q_0 P) \bmod N} = \sum_{q=0}^{Q-1} \left[W_Q^{-q(q' q_0 P + p' p_0 Q)} \sum_{p=0}^{P-1} y_{qP+p} W_P^{-p(q' q_0 P + p' p_0 Q)} \right] = \sum_{q=0}^{Q-1} \left[W_Q^{-qq' q_0 P} \sum_{p=0}^{P-1} y_{qP+p} W_P^{-pp' p_0 Q} \right]$$

Comme $p_0 Q = 1 \bmod P$ et $q_0 P = 1 \bmod Q$, l'expression se simplifie en :

$$X_{(p' p_0 Q + q' q_0 P) \bmod N} = \sum_{q=0}^{Q-1} \left[W_Q^{-qq' q_0 P} \sum_{p=0}^{P-1} y_{qP+p} W_P^{-pp' p_0 Q} \right]$$

On pose $Y_{p'Q+q'} = X_{(p'p_0Q+q'q_0P) \bmod N}$, avec $p' = 0..P-1$ et $q' = 0..Q-1$, et on a alors finalement :

$$Y_{p'Q+q'} = \sum_{q=0}^{Q-1} \left[W_Q^{-qq'} \sum_{p=0}^{P-1} y_{qP+p} W_P^{-pp'} \right]$$

Avec $y_{qP+p} = x_n$ pour $p = (n \cdot p_0) \bmod P$ et $q = (n \cdot q_0) \bmod Q$
 et $Y_{p'Q+q'} = X_k$ pour $p' = k \bmod P$ et $q' = k \bmod Q$
 (avec $q_0P + p_0Q = 1$ et $N = PQ$)

Sous cette forme, la DFT initiale est équivalente à une ré-indexation des N points d'entrée, suivie de l'application d'une DFT à deux dimensions de Q lignes et P colonnes (réalisée par Q DFT de P points puis P DFT de Q points), et enfin d'une ré-indexation des valeurs de sortie.

Par rapport à la méthode précédente (Cooley-Tukey), celle-ci permet d'éviter les N multiplications par $W_N^{-p'q}$ au prix d'une plus grande complexité dans l'indexation en entrée et en sortie, ainsi que d'une contrainte supplémentaire (P et Q premiers entre eux).

Exemple pour $P = 3$ et $Q = 4$: on prend $p_0 = 1$ et $q_0 = -1$ pour $p_0Q + q_0P = 4 - 3 = 1$

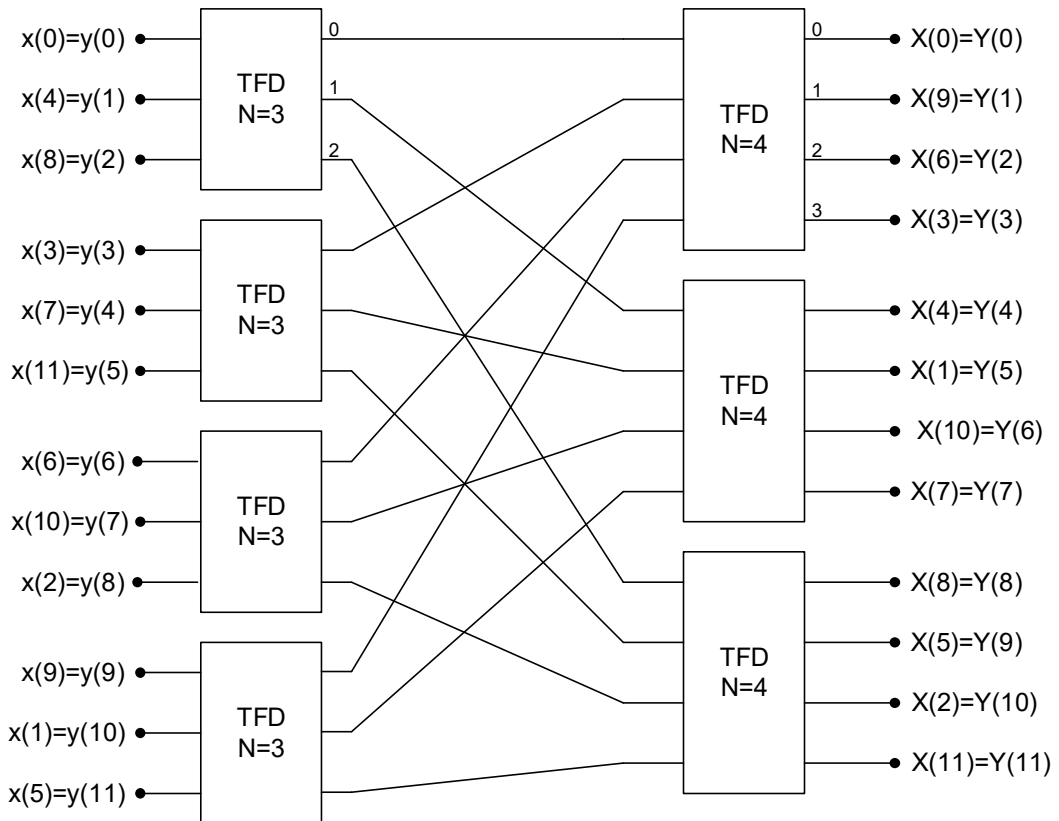
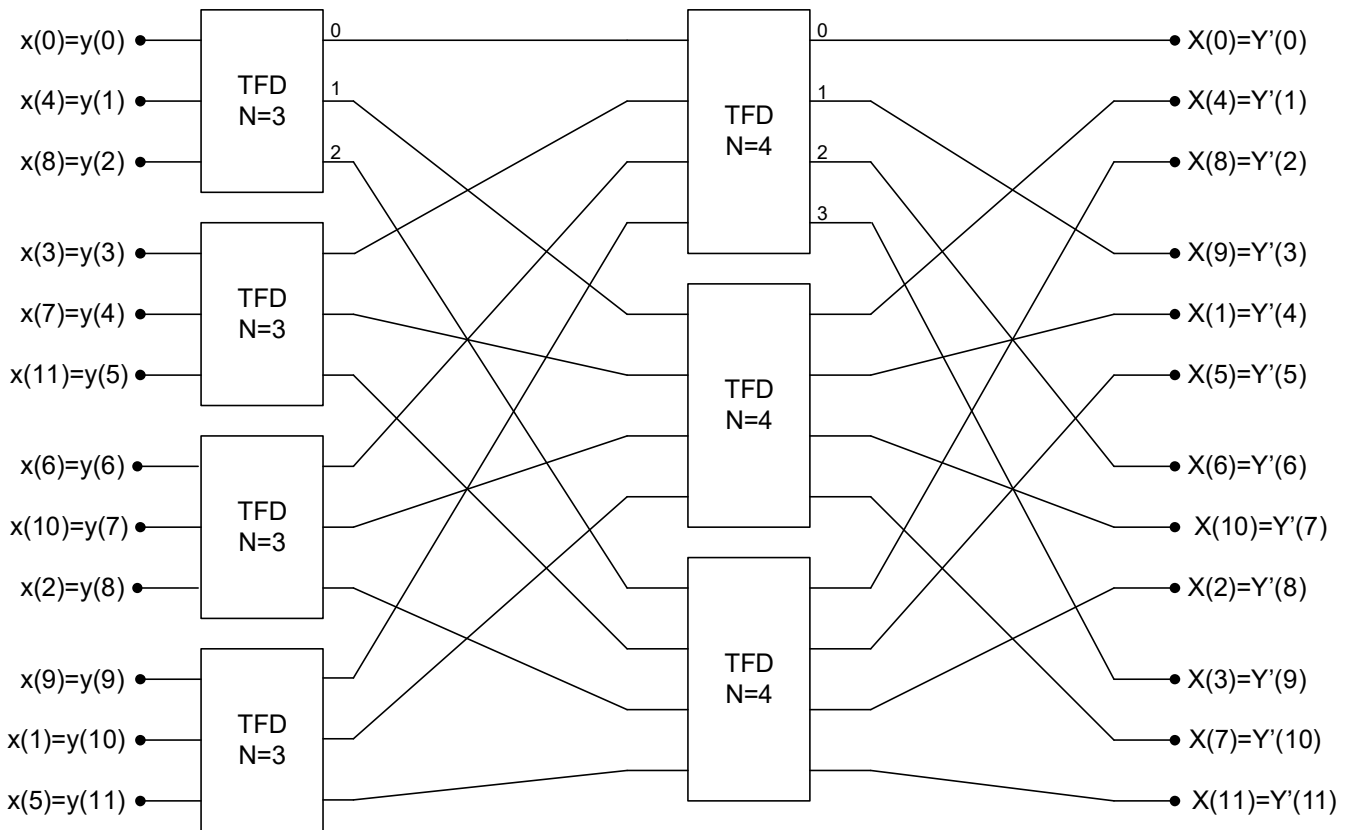


Figure 3 - FFT 12 points par Good-Thomas

La DFT initiale demandait $N^2 = P^2Q^2$ multiplications, la décomposition de Good-Thomas ramène le nombre de multiplications à $QP^2 + PQ^2 = N(P+Q)$.

Le réindexation en sortie peut être arrangée pour permettre le calcul « in-place » des transformées de Fourier du second étage :



En notant les sorties entrelacées Y'_n , on a : $Y'_{p'Q+q'} = X_k = Y'_{p'+q'Q}$ pour $p' = k \bmod P$ et $q' = k \bmod Q$.

3.3. CONSIDERATIONS SUR LA DYNAMIQUE

L'implémentation d'une FFT de grande taille peut poser un problème de dynamique. Pour une FFT classique de $N = 2^p$ points effectuée à partir de DFT élémentaires d'ordre 2, par exemple, chacun des p étages demande une représentation des données sur au moins 1 bit de plus que l'étage précédent pour limiter le bruit de calcul. En pratique, en virgule fixe, la sortie des filtres transversaux et de chaque étage de la FFT est tronquée à un nombre de bits maximum (16 ou 18 par exemple), en fonction des capacités du matériel. Ceci induit un bruit supplémentaire (blanc en première approximation) ajouté à chaque étage, limitant les performances théoriques de détection de signaux faibles. Néanmoins, l'implémentation de la FFT en virgule fixe peut tout de même permettre la gestion d'une pseudo virgule flottante qui consiste à fournir un paramètre de recadrage (exposant) pour chaque bloc de sortie. Les échantillons d'un bloc sont ainsi représentés avec une dynamique "instantanée" maximale et le bruit de calcul est minimisé.