

Système d'analyse de Fourier
par recouvrement
de blocs compatible IEEE 488

Sliding blocks Fourier analysis system coupled to the IEEE 488 bus



Patrick FURON

GERSIC, ISMRa, Esplanade de la Paix, 14032 CAEN CEDEX

Ingénieur de l'École Nationale Supérieure d'Électronique et d'Électromécanique de Caen (ENSEEC 1977). Docteur-ingénieur en électronique de l'Université Paris-XI. Assistant d'automatique à l'ISMRa de Caen.



Daniel BLOYET

GERSIC, ISMRa, Esplanade de la Paix, 14032 CAEN CEDEX

Docteur en Électronique (1970) et ès Sciences physiques de l'Université Paris Sud (1976). Actuellement professeur d'électronique à l'ISMRa de Caen. Dirige au sein du GERSIC l'activité « Instrumentation scientifique et médicale ».

RÉSUMÉ

Cet article présente un système multiprocesseur d'analyse spectrale interfacé au bus IEEE 488. Un processeur de transformation de Fourier rapide (TFR) à géométrie constante, réalisé autour de multiplieurs-accumulateurs, dispose d'une arithmétique flottante par blocs. Il utilise une table trigonométrique en notation binaire complémentée à deux où la représentation des phases 0 et $\pi/2$ est exacte. Un second processeur effectue les opérations arithmétiques de fenêtrage et de calcul du spectre de puissance; il gère le glissement des blocs de données d'entrée et est chargé des échanges avec le bus IEEE 488. L'ensemble, piloté par un microprocesseur 8 bits, peut exécuter un nouveau traitement toutes les 3 ms.

MOTS CLÉS

Transformation de Fourier rapide, géométrie constante, virgule flottante par bloc, glissement, IEEE 488, multiplieur-accumulateur.

SUMMARY

This work is dedicated to a spectral analysis system coupled to the standard IEEE 488 parallel bus. A constant geometry fast Fourier transformation (FFT) is implemented using multiplier-accumulator devices, including a block-floating point arithmetic and a binary 2's complement trigonometric table with an exact representation of the 0 and $\pi/2$ phases. A dedicated processor performs the pre et post-processing operations—windowing, spectral power computation, input blocks sliding management—and controls the exchanges with the IEEE 488 bus. This set, controlled by an eight bits microprocessor, is able to execute a new treatment every 3 ms.

KEY WORDS

Fast Fourier transformation, constant geometry, block-floating point, input blocks sliding, IEEE 488, multiplier-accumulator.

TABLE DES MATIÈRES

- 1. Introduction
- 2. Le processeur TFR ⁽¹⁾
- 3. Le processeur de pré et post-traitement (PPT)
- 4. Les performances du système
- 5. Conclusion
- Bibliographie

1. Introduction

La recherche de processeurs de Fourier rapides conduit le plus souvent à des systèmes onéreux [1] et dépendants du bus de l'ordinateur hôte. La réalisation décrite ici trouve sa place dans le créneau apparaissant entre les processeurs vectoriels rapides et les systèmes de mesure tels que les analyseurs de spectre digitaux. Son temps de calcul actuel est de 3 ms pour un tableau de 1024 points de 16 bits; sa structure permet le traitement en continu de signaux d'entrée réels, les transferts s'effectuant suivant le standard d'instrumentation IEEE 488. Cette étude a été menée dans le cadre du traitement numérique des signaux acoustiques en vue d'une analyse factorielle.

L'analyse d'un flot continu ou enregistré de données nécessite son découpage en blocs dont la taille dépend de la résolution fréquentielle immédiate recherchée et de la capacité de mémoire du système utilisé. De plus, le traitement des blocs de données consécutifs entraîne l'apparition de raies parasites liées à la troncature temporelle du phénomène étudié [2]. Elles sont généralement éliminées par application d'une fenêtre dont l'inconvénient (*fig. 1 a*) est de minimiser le rôle des données situées à proximité des extrémités des blocs. Une analyse fréquentielle précise ne tolère pas une telle perte d'information, aussi opère-t-on en outre par glissement d'une fraction de bloc, ce qui augmente le nombre de transformées de Fourier. Un cycle élémentaire de traitement se déroule selon les opérations suivantes (*fig. 1 a*) : transfert de données de l'ordinateur hôte, stockage local et glissement, fenêtrage, calcul du spectre — et du carré de son module éventuellement —, retour du résultat vers l'ordinateur hôte.

L'étude des mécanismes de transfert et des temps de calcul liés au traitement complet d'un bloc de données fait apparaître les deux goulets d'étranglement suivants :

— le premier est lié à la vitesse de transfert des données avec l'ordinateur hôte qui est limitée à 1 Moctets/s dans le cas du bus IEEE 488;

⁽¹⁾ Voir la liste des abréviations en fin d'article.

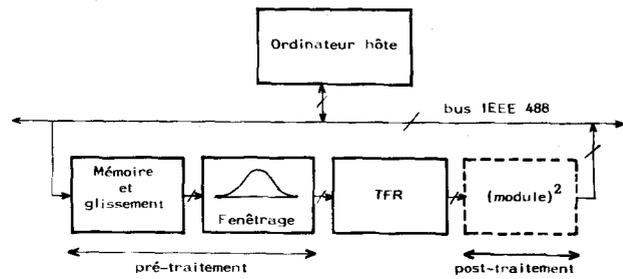


Fig. 1 a. — Représentation symbolique des principales étapes d'un traitement.
Fig. 1 a. — Symbolic representation of the main steps of the involved computation.

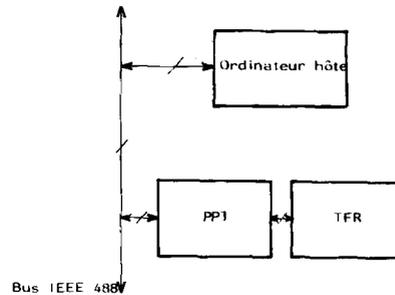


Fig. 1 b. — Schéma bloc d'une réalisation en deux niveaux de profondeur du processeur de Fourier.
Fig. 1 b. — Two blocs realization of the processing unit.

— le second est relatif au temps de calcul d'une transformée de Fourier qui dépend de l'architecture et de la technologie mises en œuvre.

En raison de ces deux contraintes, nous avons adopté une structure en deux blocs (*fig. 1 b*) communiquant entre eux et avec l'ordinateur hôte par un protocole d'échanges asynchrone. Un bloc « TFR » est affecté au calcul de la transformée de Fourier proprement dite. L'autre bloc « PPT » est dédié à l'exécution des tâches de pré-traitement (glissement, fenêtrage) et de post-traitement (calcul du spectre de puissance). Ce dernier, qui exécute simultanément des opérations arithmétiques et des transferts, opère de manière totalement transparente lors des échanges avec les zones voisines et ne ralentit pas le système. Le temps total de traitement sera donc limité soit par la durée constante d'exécution d'une TFR, soit par le temps de transfert des données au cas où l'ordinateur hôte en limiterait le rythme. Aussi, afin de limiter les échanges avec le bus IEEE 488, le bloc PPT est-il muni d'une structure de mémorisation des données et de gestion du glissement.

2. Le processeur TFR (The FFT processing unit)

2.1. TECHNIQUES DE CALCUL DES TRANSFORMÉES DE FOURIER DISCRÈTES (Discrete Fourier transforms main computation techniques)

APPLICATIONS

La transformée de Fourier discrète TFD a pour expression :

$$(1) \quad y_k = \sum_{n=0}^{N-1} W_N^{nk} x_n, \quad k=0, \dots, N-1,$$

avec

$$(2) \quad W_N = e^{-j2\pi n/N},$$

où les termes x_n désignent les échantillons temporels et y_k les composantes spectrales.

Son calcul nécessite N^2 multiplications et $N(N-1)$ additions complexes. Plusieurs méthodes permettent de réduire efficacement la quantité d'opérations intervenant dans le calcul de la TFD :

- les techniques TFR, dérivées de l'algorithme de Cooley-Tukey [3] présentent les avantages liés aux calculs récursifs. Elles sont particulièrement bien adaptées à la réalisation de processeurs à structure pipe-line et/ou parallèles;
- les algorithmes résultant d'une décomposition de N en facteurs premiers ont été comparés aux algorithmes TFR sous l'angle des réalisations matérielles associées par K. S. Anderson [4]. Ils économisent près de 30 % des opérations, mais n'étant pas liés à une méthode de calcul récursive, ils mènent à de lourdes structures matérielles et à des microprogrammes complexes. Nous ne les avons pas retenus et avons porté notre choix sur la réalisation d'un algorithme TFR à base 2.

2.2. TECHNIQUES TFR A BASE 2 (The radix 2 FFT)

Les techniques TFR à base 2 utilisent les propriétés cycliques des coefficients utilisés lors du calcul de la transformée de Fourier discrète d'un tableau de dimension $N=2^p$. Son équation $y = Wx$ peut se décomposer en un produit de p matrices selon les formes (3), (4), (5) où I_b est la matrice d'« inversion binaire » :

$$(3) \quad y = M_{p-1} \dots M_1 M_0 I_b x,$$

$$(4) \quad y = I_b M_{p-1} M_1 M_0 x,$$

$$(5) \quad y = M_{p-1} M_1 M_0 x.$$

Chaque produit par l'une des p matrices M_i est une opération appelée « étape ». Composée elle-même de $N/2$ opérations semblables appelées « papillons », une étape détermine deux nouveaux points complexes C et D à partir des points complexes d'entrée A et B et des coefficients W_N^q liés au numéro de l'étape et au rang du papillon. Pour un papillon de l'algorithme d'entrelacement temporel, les relations

$$(6) \quad C = A + B W_N^q \quad \text{et} \quad D = A - B W_N^q$$

montrent que l'opération de multiplication complexe précède l'addition, tandis qu'elle la suit dans l'algorithme d'entrelacement fréquentiel associé aux relations

$$(7) \quad C = A + B \quad \text{et} \quad D = W_N^q (A - B)$$

Le calcul complet d'un papillon dans un algorithme TFR de base 2 nécessite donc :

- 2 multiplications et 4 additions complexes, soit 4 multiplications et 6 additions réelles;
- 2 opérations de lecture des coefficients de la table trigonométrique;
- 4 opérations de lecture des données;
- 4 opérations d'écriture des résultats.

Les décompositions (3), (4), (5) définissent également la structure de calcul et aboutissent à diverses réalisations [5].

2.3. TECHNIQUES DE RÉALISATION DES PROCESSEURS TFR (FFT algorithms implementations)

S'il est possible de réaliser, lorsque N est faible, le calcul d'une TFR en continu avec un processeur câblé disposant de $(N/2) \log_2 N$ papillons de base 2 ou de $(N/4) \log_4 N$ papillons de base 4 [6], son coût et sa complexité deviennent extrêmes pour les grandes valeurs de N . Un processeur TFR approche toutefois cette conception : il s'agit du SPS-1000 [7] dont l'architecture modulaire est basée sur un empilement de « tranches de modules papillon » associées en série et/ou en parallèle. En général, les processeurs TFR en temps réel destinés aux applications rapides ont une architecture pipe-line performante mais coûteuse; ils sont issus de la mise en cascade de blocs arithmétiques affectés chacun au calcul des papillons d'une étape ([8], [9], [10]).

Pour les processeurs TFR destinés aux applications basse fréquence ou en temps différé, il est plus économique d'employer un bloc arithmétique unique pour le calcul de l'ensemble des étapes. Dans l'un et l'autre cas, le nombre de multiplieurs et d'additionneurs utilisés dans l'unité de calcul d'un papillon dépend des performances souhaitées et des choix technologiques. On trouve principalement des réalisations comportant :

- 4 multiplieurs et 6 additionneurs-soustracteurs [11];
- 2 multiplieurs et 3 additionneurs-soustracteurs ([4], [12], [13]);
- 1 multiplieur et 2 additionneurs-soustracteurs [13].

Bien adapté à la réalisation des papillons de l'algorithme d'entrelacement temporel, le multiplieur-accumulateur ou MAC est un composant qui intègre un multiplieur suivi d'un additionneur-soustracteur opérant à pleine précision. Il peut être utilisé seul et permettre le calcul d'un papillon en 10 périodes d'horloge [15] ou être associé à d'autres unités arithmétiques ([16], [17]).

Parallèlement à l'architecture du processeur TFR, le mode de représentation des nombres à l'intérieur de la machine et la méthode utilisée pour le contrôle de l'évolution de l'amplitude des données dans le papillon influent sur le temps de traitement et les bruits de calcul. Trois représentations sont utilisées :

- la représentation en virgule fixée qui conduit à la plus grande rapidité mais introduit un bruit de

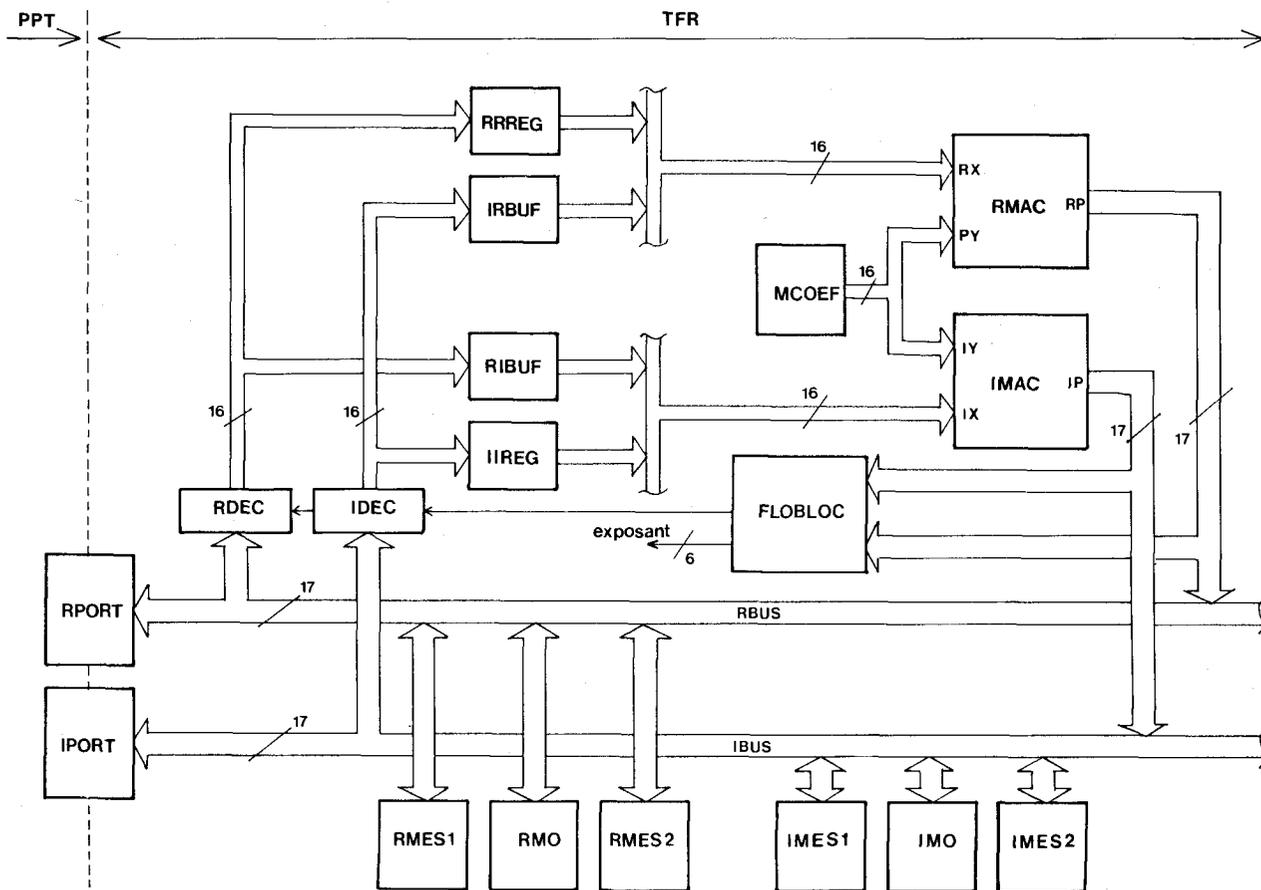


Fig. 2. — Schéma bloc de la zone TFR où l'accent est mis sur le chemin des données.
 Fig. 2. — Block representation of the FFT unit emphasizing the whole data path.

quantification supérieur ([18], [19]) à celui qui résulte de la conversion analogique-numérique des données d'entrée;

- la représentation en virgule flottante qui possède les meilleures performances en bruit de calcul [19], mais mène soit à une grande lourdeur de son arithmétique, soit à de moindres performances en vitesse de traitement;

- la représentation en virgule flottante par blocs ([18], [20]) basée sur une arithmétique en virgule fixée dans laquelle un décalage approprié, appliqué globalement à l'ensemble des résultats d'une étape, prévient tout débordement des nombres. Le tableau-résultat est accompagné d'un exposant, résultat du décompte des décalages utilisés. Cette représentation intéressante possède d'excellentes performances en bruit de calcul qui devient négligeable par rapport au bruit d'entrée [18]. Simple à implanter, elle est aussi rapide qu'une structure en virgule fixée.

Cette analyse des solutions proposées précédemment nous a conduit à effectuer les choix suivants bien adaptés au traitement en temps réel des signaux acoustiques :

- adressage à géométrie constante;
- arithmétique en mode flottant par bloc;

- traitement des papillons par deux multiplieurs-accumulateurs.

2. 4. LE PROCESSEUR TFR (Design of the FFT unit)

2. 4. 1. Structure arithmétique et chemin des données (Arithmetic assembly and data paths)

Le processeur TFR réalisé possède une unité arithmétique unique, opérant en mode flottant par bloc, séquentiellement affectée au calcul des $N/2 \log_2 N$ papillons.

Le schéma proposé en figure 2 est une représentation par blocs du processeur réalisé où l'accent est porté sur le chemin des données. Le système est articulé autour de deux bus de 17 bits (RBUS, IBUS) véhiculant, en représentation en complément à 2, les parties réelles et imaginaires des données. L'algorithme de calcul à géométrie constante nécessite, contrairement à l'adressage en place, une mémoire source et une mémoire résultat dont le rôle est permuté à chaque étape. Ce rôle est tenu par les mémoires RMES_i, RMO et IMES_i, IMO pour les parties réelles et imaginaires. Chaque TFR utilise les mémoires RMO (IMO) et RMES1 (IMES1) ou RMES2 (IMES2) suivant que son numéro d'ordre est pair ou impair; les blocs mémoire laissés disponibles sont alors utilisés

APPLICATIONS

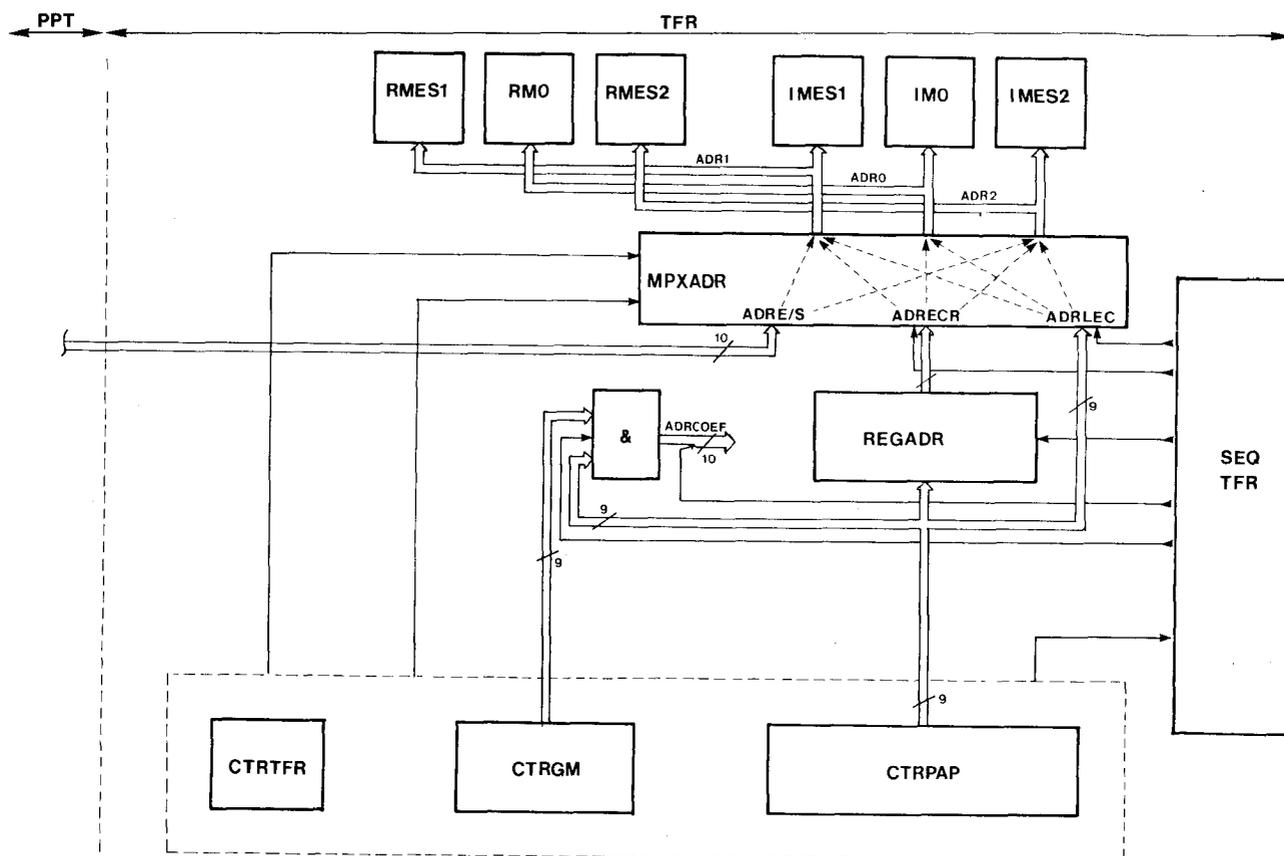


Fig. 3. — Système de génération des adresses de la zone TFR.

Fig. 3. — Address generation for the FFT unit.

pour l'évacuation des résultats précédents et l'introduction des données suivantes venant de la zone PPT.

Deux multiplieurs-accumulateurs sont affectés au calcul des parties réelles (RMAC) et imaginaires (IMAC) des papillons. Les données émanant des mémoires correspondantes sont décalées et formatées à 16 bits par les registres RDEC et IDEC sous contrôle du module FLOBLOC de gestion du mode flottant par bloc. Ce module enregistre les dépassements à 17, 16 et 15 bits lors d'une étape et applique des décalages en conséquence au cours de l'étape suivante. Il prévient tout dépassement à 17 bits et, par application de facteurs multiplicatifs, amène les données dans une gamme d'amplitude qui minimise le bruit de calcul dans un papillon. Le module FLOBLOC comptabilise l'ensemble des décalages intervenus dans une FFT et délivre comme résultat le signal « exposant ».

Le déroulement d'un papillon est dérivé de la séquence de calcul proposée par SCHIRM [15], sur la base de l'utilisation d'un seul MAC, permettant le calcul d'un papillon en une séquence de dix périodes d'une horloge de séquençement. L'utilisation de deux multiplieurs-accumulateurs opérant de manière interne sur 35 bits et assurant ainsi le calcul des résultats intermédiaires en pleine précision abaisse dans notre cas le nombre de périodes d'horloge d'un facteur

deux. Les registres trois états RRREG et IIREG assurent à l'ensemble un fonctionnement de type pipeline; le croisement des données s'effectue au niveau des registre et buffer RRREG et IRBUF (RIBUF et IIREG). Les données trigonométriques sont stockées dans la mémoire MCOEF; enregistrées en valeur négative, les données trigonométriques donnent une représentation exacte du nombre -1 et donc des phases 0 et $\pi/2$; elles introduisent un bruit de coefficient limité.

Enfin, à l'examen du déroulement d'un papillon, il apparaît que les bus de données principaux (RBUS et IBUS) sont disponibles pendant un cycle d'horloge. Ce fait est mis à profit pour glisser vers les mémoires non impliquées dans le cycle TFR en cours, une opération de type entrée-sortie. Un calcul élémentaire montre que le nombre de créneaux disponibles par TFR est très largement supérieur au nombre de données à évacuer et à introduire en préparation de la transformée suivante; l'opération n'est donc pas pénalisante.

2.4.2. Adressage et séquençement (Address generation and sequencing)

La figure 3 représente un schéma par bloc du système de génération d'adresses pour une structure à géométrie constante. Suivant un tel algorithme [2], les adres-

ses nécessaires à l'exécution d'un papillon – lecture, écriture, coefficient – dérivent simplement des indices n ($0 \leq n \leq 511$ pour $N=1024$) du papillon et p ($0 \leq p \leq 9$ pour $N=1024$) de l'étape. Les adresses de lecture sont $2n+0$ et $2n+1$, tandis que les adresses d'écriture s'écrivent $0+n$ et $512+n$. Enfin, si l'on utilise l'ordre croissant de rangement dans la mémoire des coefficients, l'adresse de ces coefficients (ADRCOEF) est obtenue par intersection binaire entre le numéro du papillon n et un masque binaire attaché à l'indice p de l'étape. Un multiplexeur 3 voies 10 bits assure l'adressage complet des trois paires de mémoires de données du système. L'entrée ADRE/S du multiplexeur est couplée au module PPT et sert à l'acquisition et à la restitution des données par vol d'un cycle du papillon.

2.4.3. Mécanisme des échanges entre les modules PPT et TFR (PPT-FFT communications)

Les mémoires et les bus principaux (RBUS, IBUS) sont disponibles (2.4.1) durant un cycle d'horloge de chaque papillon et sont utilisés pour effectuer durant cette période une opération d'entrée-sortie. Le protocole d'échanges totalement asynchrone et le calcul des adresses effectué par le module PPT permettent de considérer pour ce type d'opération la zone TFR comme une mémoire. Ainsi, pendant qu'une TFR est effectuée sur le i -ième tableau, peuvent être respectivement lus et écrits les résultats du $(i-1)$ -ième calcul et les données du $(i+1)$ -ième. Le parallélisme des opérations est indiqué en figure 4 suivant que le goulet d'étranglement se situe en zone PPT ou TFR. Un protocole d'échange assure le fonctionnement correct du système dans chacun des cas.

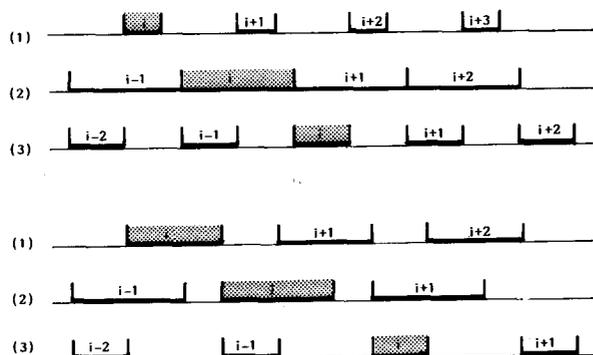


Fig. 4. – Enchaînement des tâches d'entrée des données et prétraitement (1), de transformation de Fourier (2), de post-traitement et de sortie des résultats (3) suivant que les échanges avec le BUS IEEE 488 sont plus rapides (fig. 4a) ou plus lents (fig. 4b) que le temps de calcul d'une TFR.

Fig. 4. – Time diagram of the tasks linkage (input of data and pre-processing (1), FFT (2), post-processing and sending back of data (3)) as a function of the relative speed of the IEEE 488 throughput rate and of the FFT calculation time. Figure 4a: the FFT calculation time is the limiting parameter; figure 4b: the exchange time is the bottleneck.

3. Le processeur de pré et post-traitements (PPT) (The pre and post-processing unit)

Nous avons introduit précédemment l'intérêt, pour une analyse factorielle correcte de données, des opérations de fenêtrage et de glissement d'une fraction de bloc qui doivent être simultanées avec l'exécution des transformations de Fourier. Le système est donc muni d'un ensemble de pré et post-traitement réalisant ces opérations ainsi, qu'au choix, le calcul du spectre de puissance. Il communique avec le bus IEEE 488 et avec le bloc TFR qui pour lui n'est qu'une simple mémoire à temps d'accès élevé.

Les actions à mener dans la zone PPT sont de divers types :

- transferts et mise au format des données par rapport aux zones amont et aval;
- enregistrement de la fonction de fenêtrage définie par l'utilisateur dans l'ordinateur hôte;
- exécution du fenêtrage;
- gestion du glissement des blocs;
- calcul du carré du module du spectre.

Le module PPT est représenté en figure 5. Il est piloté par un processeur de contrôle et de commande (PCC), chargé de l'exécution et de l'enchaînement des tâches, qui est secondé par deux processeurs rapides spécialisés, l'un (SEQTRAP) dans la gestion des transferts rapides, l'autre (SEQCALC) dans l'exécution des opérations arithmétiques. Il comporte deux mémoires de 1024 mots de 16 bits: l'une (MF) contient la fonction fenêtrage, l'autre (MD) stocke les données en provenance de l'ordinateur hôte. Un séquenceur (ADRCALC) élabore les adresses destinées à la zone TFR et aux mémoires MF et MD; il assure la gestion du glissement des données.

Le PCC, outre ses fonctions de contrôle, sert à l'initialisation de l'ensemble de la carte, et peut, bien que ce rôle soit réservé en mode normal au processeur rapide (SEQTRAP), le remplacer dans les échanges avec les zones voisines. Par demande « d'accès direct mémoire » au PCC, SEQTRAP est capable de réaliser une liaison bidirectionnelle au rythme de 2 Moctets/s entre le bus IEEE 488 (soit le double de la vitesse maximale normalisée [21]) et les zones de mémorisation ou de calcul du PPT. Il commande également par le biais de EPORT les transformations de format 8/16 bits entre le bus IEEE 488 et la zone de mémorisation.

Le processeur de calcul possède une structure simple définie pour ne pas ralentir les échanges avec la zone d'entrée et la zone TFR. Son chemin de données est constitué d'un bus unique de 16 bits assurant les transferts entre les zones mémoires, le multiplieur-accumulateur et les ports de liaison (EPORT, RPORT, IPORT). Avant d'effectuer le pré-traitement, il transfère l'ensemble des nouvelles données en mémoire MD, à des adresses successives définies modulo N; l'opération de glissement est ainsi simultanément effectuée. Il multiplie alors chaque donnée par son coefficient de fenêtrage et délivre immédiate-

APPLICATIONS

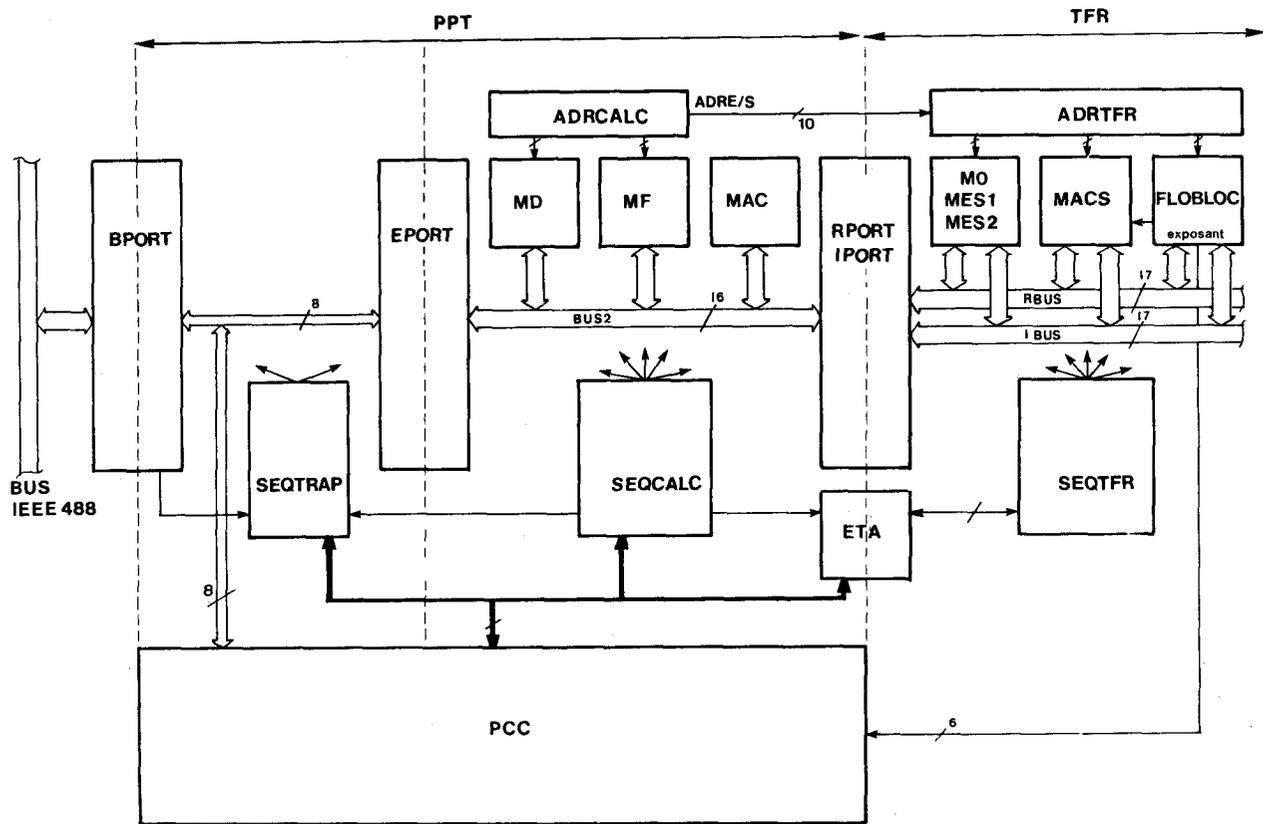


Fig. 5. — Bloc diagramme du système complet; les différentes zones communiquent via des ports. Le PPT comprend une zone de transfert en format 8 bits et une zone de traitement en 16 bits.

Fig. 5. — Block diagram representation of the whole system showing the 8 and 16 bits parts of the PPT unit and the communication ports.

ment le résultat à la mémoire réelle d'entrée sortie concernée (RMES1 ou RMES2) via le port réel RPORT. Dès que la transformation de Fourier de ce bloc de données est terminée — ce que signale le processeur TFR via le registre d'état du système (ETA) — le processeur de calcul entre en phase de post-traitement. Il appelle séquentiellement les valeurs complexes y_k dans la zone TFR, et effectue optionnellement le calcul des $|y_k|^2$ qu'il renvoie vers l'ordinateur hôte. Suivant la vitesse de transfert du bus IEEE 488, il attend alors le nouveau jeu de données ou entame immédiatement une nouvelle acquisition.

4. Les performances du système (System performances)

Nous allons maintenant nous intéresser aux performances en vitesse d'exécution et en niveau de bruit du système. Dans la version réalisée, le temps de calcul (120 ns) des multiplieurs-accumulateurs utilisés (TDC 1010J (TRW)) constitue le facteur limitatif de la zone TFR. Un papillon s'exécute en 600 ns, ce qui correspond à un temps de calcul d'environ 3 ms par TFR d'un bloc de 1024 données. Compte tenu de l'enchaînement des tâches illustré en figure 4, les per-

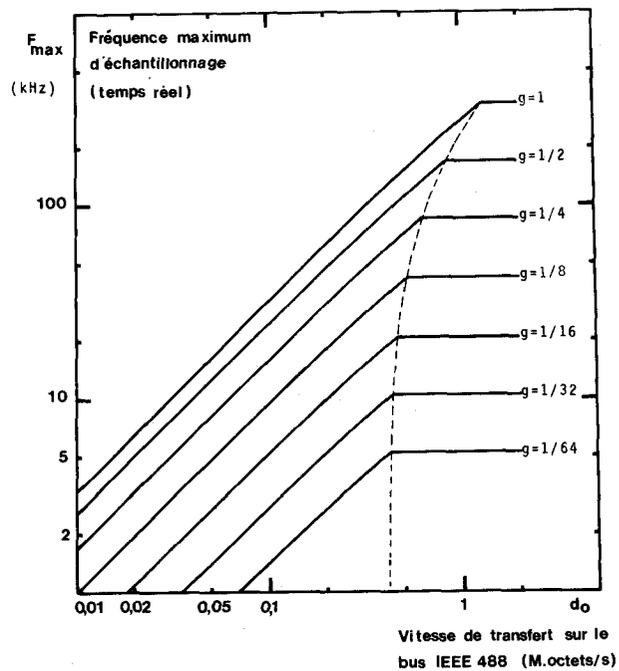


Fig. 6. — Evolution de la fréquence d'échantillonnage du système réalisé en fonction du taux de glissement g désiré et du débit d_0 du bus IEEE 488.

Fig. 6. — Maximal sampling rate of the system as a function of the sliding coefficient g and of the IEEE 488 throughput rate.

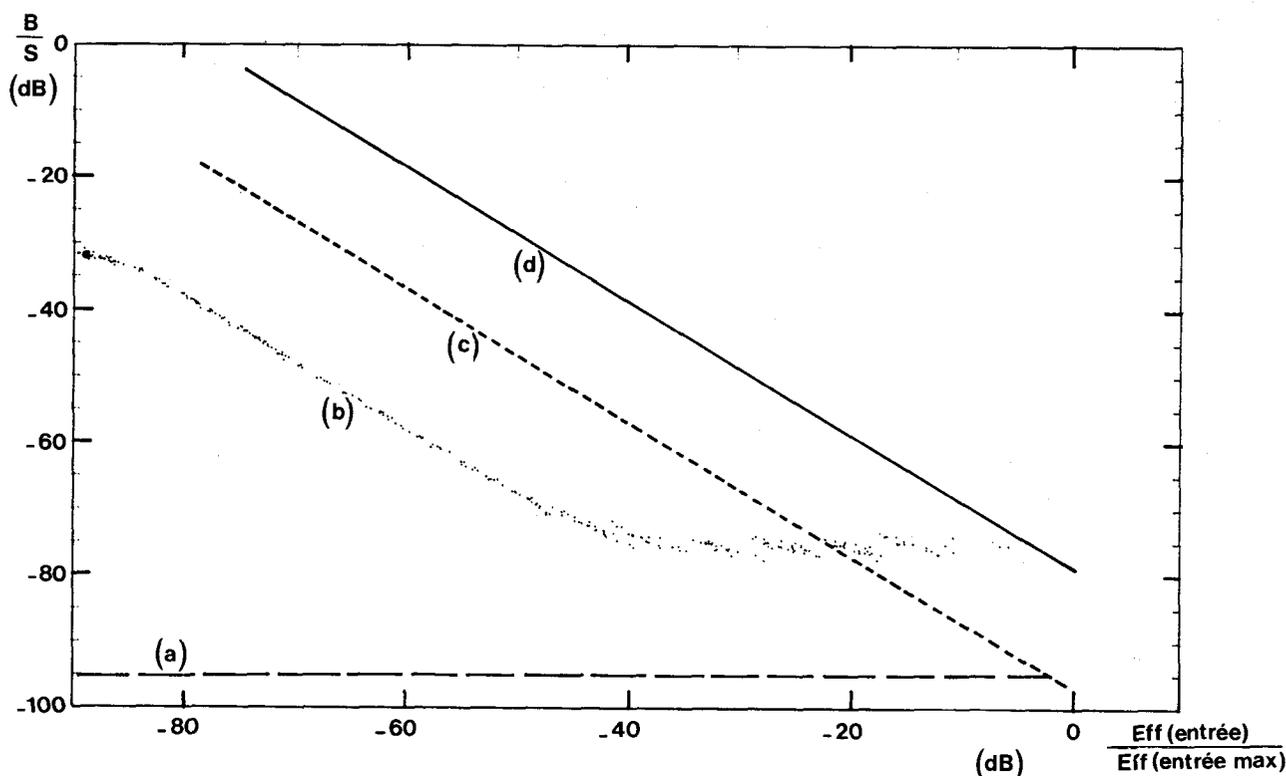


Fig. 7. — Evolution des composantes de bruit du système en fonction de la valeur efficace du signal d'entrée: (a) bruit de troncature de la table trigonométrique MCOEF; (b) bruit arithmétique; bruit de troncature d'entrée pour des mots d'entrée de 16 bits (c) et de 12 bits (d).

Fig. 7. — Noise components evolution of the system as a function of the r.m.s. value of the input signal. Curve (a) shows the truncating effect of the trigonometric table (MCOEF); curve (b) is related to the arithmetic noise; curves (c) and (d) show the effect of the truncation input noise for 16 bits and 12 bits data respectively.

formances en vitesse d'exécution du système sont limitées soit par la zone TFR, soit par le débit des données sur le bus IEEE 488.

La figure 6 illustre ce fait en le rapportant à la fréquence maximale d'échantillonnage permise sur ce système — où l'on suppose que l'ordinateur hôte dispose d'une mémoire de stockage temporaire — en fonction du débit d_0 et du taux g de glissement des blocs désirés. Dans la partie horizontale des courbes, le temps global de traitement du flot continu de données est limité par la zone TFR, alors que dans les parties croissantes il est lié au débit d_0 du bus de transfert avec l'ordinateur hôte.

Le bruit de calcul d'un tel système fonctionnant en mode flottant par blocs est également à considérer. L'erreur globale résulte de la contribution des bruits d'entrée de calcul et de coefficient ([18], [19], [20]). Par utilisation d'un micro-ordinateur muni d'une arithmétique flottante (HP 85) et délivrant un résultat de transformation de Fourier « exact », nous avons pu isoler ces trois contributions et comparer leurs importances relatives en fonction de l'énergie du bloc de données traité [18]. Ce travail qui sera détaillé dans un article ultérieur est résumé en figure 7 où est porté le rapport B/S de l'écart type de l'erreur sur une des composantes (réelle ou imaginaire) et de la valeur efficace du signal traité en fonction de cette dernière. Le signal considéré est un bruit blanc uniformément

réparti en amplitude entre deux limites programmables $\pm M$; sa valeur efficace est donc $M/\sqrt{3}$. Aucune fenêtre n'est appliquée dans ce type de spécifications.

Ces courbes montrent que l'erreur de coefficient (MCOEF) est toujours négligeable tandis que la propagation dans le calcul par FFT du bruit de quantification d'entrée contribue de façon dominante excepté pour les plus fortes valeurs efficaces de signaux d'entrée définis sur 16 bits signés. Le calcul montre [19] que pour une arithmétique en virgule flottante avec mantisse de 16 bits signés le rapport B/S, indépendant de l'énergie du bloc d'entrée, vaut -84 dB. L'utilisation d'un tel dispositif arithmétique n'est ainsi justifiée, par rapport au mode flottant par blocs, que pour des données d'entrée provenant d'un convertisseur analogique digital d'entrée de grande précision opérant en virgule flottante.

5. Conclusion (Conclusion)

Nous avons présenté la réalisation d'un processeur de Fourier rapide interfacé au bus IEEE 488. Sa structure à deux niveaux de profondeur lui permet d'effectuer de manière simultanée les opérations d'entrée sortie sur les blocs $i-1$ et $i+1$ et le calcul du spectre

du bloc i. Les fonctions et les performances du système multiprocesseur sont adaptées au traitement en temps réel de signaux acoustiques. Pour des applications demandant un traitement plus rapide, les limites actuelles du système proviennent soit de la vitesse de transfert des données sur le bus IEEE 488, soit du temps d'exécution d'une TFR. Dans le premier cas, il est envisageable de modifier la seule zone d'accès au bus pour rendre le système, ou sa seule zone TFR, compatible à un autre bus standardisé (MULTIBUS, QBUS, VERSABUS...). Dans le second cas, l'introduction des multiplieurs-accumulateurs compatibles les plus récents tels ceux de la série WTL 2010 de Weitek permettrait de ramener le temps de calcul de la transformée de Fourier d'un bloc à moins de 2 ms. Enfin, ce système a été spécialement conçu pour accroître la vitesse d'analyse de signaux acoustiques enregistrés. L'adaptation au traitement en temps réel est particulièrement simple quand le goulet d'étranglement n'est pas dû au transfert des données sur le bus. Il est alors très envisageable de munir le processeur de calcul d'une chaîne de conversion analogique numérique qui permettrait, avec les multiplieurs-accumulateurs les plus récents, une analyse spectrale jusqu'à des fréquences d'échantillonnage égales à 512. g kHz, g étant le taux de glissement.

Manuscrit reçu le 24 juillet 1985.

BIBLIOGRAPHIE

- [1] E. CATIER, Le turbo-ordinateur, *Electronique industrielle*, n° 62, janvier 1983, p. 72-77.
- [2] S. D. STEARNS, Digital signal analysis, *Hayden Book Company, Inc.*, Rochelle Park, New Jersey.
- [3] J. W. COOLEY et J. W. TUKEY, An algorithm for machine calculation of complex Fourier series, *Math. Comp.*, 19, avril 1965, p. 297-301.
- [4] K. S. ANDERSON, Real time considerations for DFT algorithms, *Proc. SPIE Int. Soc. Opt. Eng. (U.S.A.)*, 431, 1983, p. 230-238.
- [5] Z. ALI, FFT algorithms speed digital signal processing, *Electronic Design*, juillet 1980, p. 111-118.
- [6] G. D. BERGLAND, Fast Fourier transform hardware implementations, an overview, *IEEE trans. on Audio and Electro-acoustics*, AU-17, juin 1969, p. 104-108.
- [7] R. A. COLLESIDIS, T. A. DUTTON et J. R. FISCHER, An ultra high speed FFT processor, *ICASSP 80*, 1980, p. 784-787.
- [8] N. U. CHOWDARY et W. STEENAART, Very high speed Fourier transform processor realization techniques, *ICASSP 83*, 1983, p. 28-31.
- [9] B. GOLD et T. BIALLY, Parallelism in fast Fourier transform hardware, *IEEE Trans. Audio and Electroacoustics*, AU-21, n° 1, février 1973, p. 5-16.
- [10] L. MINTZER, FFT Implementation techniques, *ICASSP 83*, 1983, p. 408-412.
- [11] S. SHIMIZU et M. KITAGAWA, A study of high-speed FFT using highly multi-dimensional memory structure, *The transactions of the IECE of Japan*, E-64, n° 3, mars 1981, p. 160-167.
- [12] Z. ALI, A high speed FFT processor, *IEEE trans. on communications*, 26, n° 5, mai 1978, p. 690-696.
- [13] K. J. HAUSER, Design of high capacity fast Fourier transform unit, *Thorn EMI Electronics Australia Pty, Ltd*, Elisabeth, South-Australia, Australia.
- [14] M. GINDRE et D. HOSTE, Processeur FFT temps réel utilisant la famille Am 29500: exemple de réalisation, *Minis et Micros* n° 223, 1984, p. 40-45.
- [15] L. SCHIRM IV, Packing a signal processor onto a single digital board, *TRW Inc., TRW LSI Products*, La Jolla, California 92038, USA.
- [16] R. J. KARWOSKI, An introduction to digital spectrum analysis including a high speed FFT processor design, *TRW LSI Products*, La Jolla, California 92038.
- [17] R. J. KARWOSKI, A four cycle butterfly arithmetic architecture, *TRW LSI Products*, La Jolla, California 92038, USA.
- [18] P. FURON, Système multiprocesseur d'analyse spectrale en temps réel d'un flot continu de données acoustiques couplé au bus parallèle IEEE 488, *Thèse de doctorat-ingénieur*, Université de Paris-Sud, centre d'Orsay, juin 1984.
- [19] A. V. OPPENHEIM et C. J. WEINSTEIN, Effects of finite register length in digital filtering and the fast Fourier transform, *Proc. IEEE (Invited Paper)*, 60, 1972, p. 957-976.
- [20] P. D. WELCH, A fixed point fast Fourier transform error analysis, *IEEE Trans. Audio Electroacoustics*, AU 17, 1969, p. 151-157.
- [21] American National Standard, IEEE Standard Digital Interface for Programmable Instrumentation, ou ANSI/IEEE Std 488, 1978, publié par "The Institute of Electrical and Electronics Engineers, Inc. 345 East 47th Street, New York, NY 10017".

LISTE DES ABRÉVIATIONS

ADRCOE	:	bus d'adresse de la table de coefficients (défini dans le texte, page 13)
PPT	:	processeur de pré et post-traitements (défini dans le titre du paragraphe 3, page 16)
MPXADR	:	multiplexeur d'adresses
ADRE/S	:	adresse d'entrée/sortie
ADRECR	:	adresse d'écriture
ADRLEC	:	adresse de lecture
REGADR	:	registre d'adresse
SEQTFR	:	séquenceur de la TFR
CTRFR	:	compteur du nombre de TFR
CTRGM	:	compteur et générateur de masque
CTRPAP	:	compteur de papillons