

Le circuit processeur de signal

du CNET, Grenoble

TABLE DES MATIÈRES

1. Présentation du circuit intégré UPTS et des outils de développement
2. Les applications typiques du UPTS
3. L'architecture générale du circuit
4. Le séquençement interne
5. Le port B
6. Le port C
7. Le jeu d'instructions du UPTS
 - 7.1. Les registres du UPTS
 - 7.2. Les types d'instructions
8. Méthodologie de développement d'une application pour le UPTS
9. Un exemple d'utilisation, le filtrage transversal
 - 9.1. Le filtrage transversal
 - 9.2. Programme d'exécution SPL 1
 - 9.3. La simulation
 - 9.4. Le filtre en temps réel

1. Présentation du circuit intégré UPTS ⁽¹⁾ et des outils de développement

Ce circuit est un microprocesseur dont l'architecture a été adaptée à une classe de fonctions particulières : les calculs arithmétiques réguliers sur des tableaux de données codées en virgule fixe.

Sa puissance de calcul est donnée par une séparation des données en deux mémoires RAM, distinctes de la mémoire ROM contenant le programme, permettant ainsi l'accès de deux données pendant l'accès d'une instruction. Sa puissance de calcul est aussi donnée par des unités de calcul spécialisées : unité de calcul des adresses des mémoires, multiplieur cablé, unité de calculs arithmétiques et logiques. En outre pour que ce circuit s'intègre facilement à son environnement il est pourvu de deux bus bidirectionnels de 16 bits chacun. Un premier bus

⁽¹⁾ UPTS : Microprocesseur pour le traitement du signal (voir photo 1).

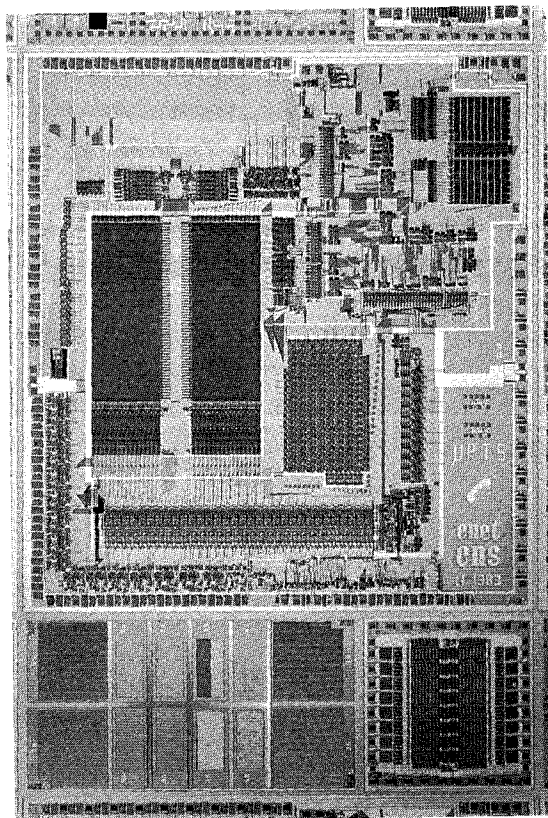


Photo 1.

permet le transfert des données par un échange par des boîtes aux lettres, le circuit a alors un espace d'adressage de 64 K mots. Le deuxième bus permet un contrôle aisé du circuit par un microprocesseur d'usage général (arrêt, accès aux registres et mémoires internes, redémarrage à un point d'entrée quelconque de programme) ou permet l'entrée des instructions du UPTS soit dans un mode d'extension de la mémoire ROM interne soit dans un mode où les instructions sont systématiquement stockées à l'extérieur (émulation, petites séries).

Le jeu d'instruction permet une programmation aisée des algorithmes de traitement du signal. Le circuit est capable de traiter jusqu'à 3,33 millions d'instructions par seconde soit une puissance maximale de calcul de 6,67 millions d'opérations en virgule fixe par seconde. (Les mots traités peuvent être codés sur 16 bits à 32 bits.) Une instruction permet en 300 ns : une modification des pointeurs des mémoires données, un accès de deux données dans ces mémoires, un calcul d'un produit, un calcul arithmétique ou logique suivi du rangement du résultat en mémoire.

Un langage de programmation a été défini : SPL 1 (signal processing language n° 1) et des outils logiciels de traduction en binaire puis de simulation complète au niveau

des registres et mémoires du circuit sont disponibles sur VAX (VMS), Micromega (Unix), IBM 43. Un émulateur fonctionnant en temps réel peut être chargé par ce code binaire et permet l'émulation du circuit dans une application matérielle et une mise au point aisée du programme : points d'arrêts, analyse des transferts de données aux broches du circuit, trace de la valeur des registres internes en cours d'émulation par vol de quelques cycles (inférieur à 3 μ s). L'analyse des résultats peut être faite sur l'émulateur lui-même ou après transferts sur l'ordinateur principal. Ces outils de développement permettent par leurs puissances importantes de réduire considérablement les temps et les coûts de développement d'un programme pour le UPTS adapté à une application précise.

2. Les applications typiques du UPTS

La souplesse et la puissance du UPTS permettent son emploi dans une gamme étendue d'applications où les microprocesseurs universels manquent de puissance de calcul pour traiter aisément le problème. Les applications potentielles s'étendent depuis le contrôle des asservissements mécaniques jusqu'aux radars sans oublier l'imagerie, le traitement de la parole, l'instrumentation, les télécommunications, les traitements sonores dans la bande audio.

3. L'architecture générale du circuit

Le UPTS est un microprocesseur ayant des caractéristiques adaptées au traitement numérique à haute rapidité. Ce circuit n'a pas une architecture de von Neuman habituellement rencontrée dans les microprocesseurs d'usage

général, mais il a une architecture dite de Harvard pour obtenir à la fois vitesse et souplesse d'emploi. Dans cette structure les données et le programme sont sur des supports séparés permettant le recouvrement des accès et le décodage de l'instruction avec l'exécution de l'instruction précédente. Le UPTS possède aussi des éléments de calcul spécialisés et optimisés pour les opérations arithmétiques en virgule fixe (additionneur, multiplieur). Le chemin des données n'a pas une largeur fixe : il est adapté aux problèmes liés à l'emploi de l'additionneur (débordement) et du multiplieur (augmentation de la précision) sa largeur minimale est de 16 bits avec une largeur maximale de 32 bits. La figure 1 représente les principaux blocs de cette architecture. Le UPTS possède deux bus servant aux transferts des données :

- le bus AMONT affecté aux transferts des opérandes avant un calcul;
- le bus AVAL affecté aux transferts des résultats d'une opération.

Une unité arithmétique de largeur de 32 bits prend un opérande sur le bus AMONT et donne un résultat sur le bus AVAL, le deuxième opérande est contenu dans l'unité arithmétique qui elle-même comporte un accumulateur et quatre registres de 32 bits chacun.

Le multiplieur reçoit directement les données des RAM ou de l'accumulateur et donne le résultat sur le bus AMONT pour être utilisé comme opérande de l'unité arithmétique et logique.

Le port B est un organe d'interface entre les bus AMONT et AVAL internes et l'extérieur, sa fonction principale est une fonction de transfert de données.

Les deux RAM C et D entièrement symétriques, chacune comporte un plan de RAM de 128 mots de 16 bits et deux registres, les adresses des RAM sont calculées par le séquenceur.

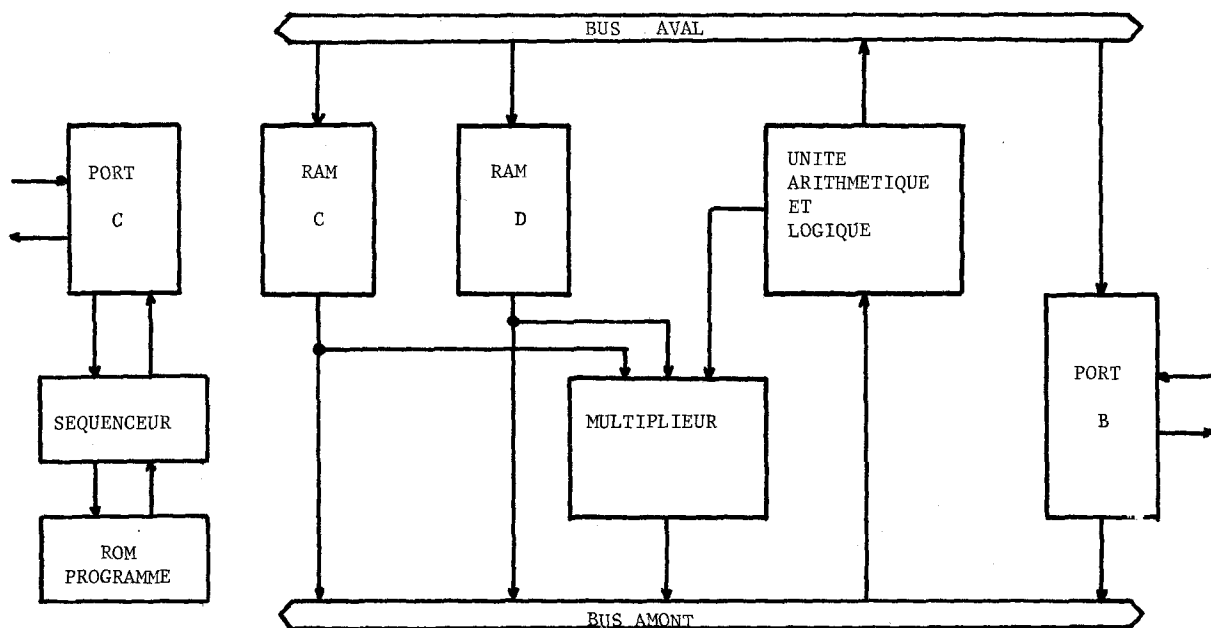


Fig. 1. — Architecture du processeur de signal μ PTS.

Le port C est un bloc d'interface entre le séquenceur du circuit et l'extérieur, sa fonction principale est une fonction de contrôle du circuit par l'environnement du circuit.

La ROM programme est un élément de stockage interne du programme contrôlant le fonctionnement du circuit.

Le séquenceur constitue le bloc de contrôle du circuit et de génération des commandes des autres blocs du circuit.

Il comprend un compteur pour le programme, une pile, un ensemble de compteurs pour les boucles, et un registre contenant les drapeaux.

4. Le séquençage interne

Le cycle interne est découpé en six phases d'égale durée (la durée d'une phase est donnée par la fréquence d'oscillation de l'horloge imposée au circuit). Cette durée est de 50 ns (minimum) pour la durée du cycle interne ou cycle instruction. Toutes les instructions du microprocesseur sont exécutées en un seul cycle interne ou cycle instruction et les différentes opérations sont largement actives simultanément (cf. fig. 2).

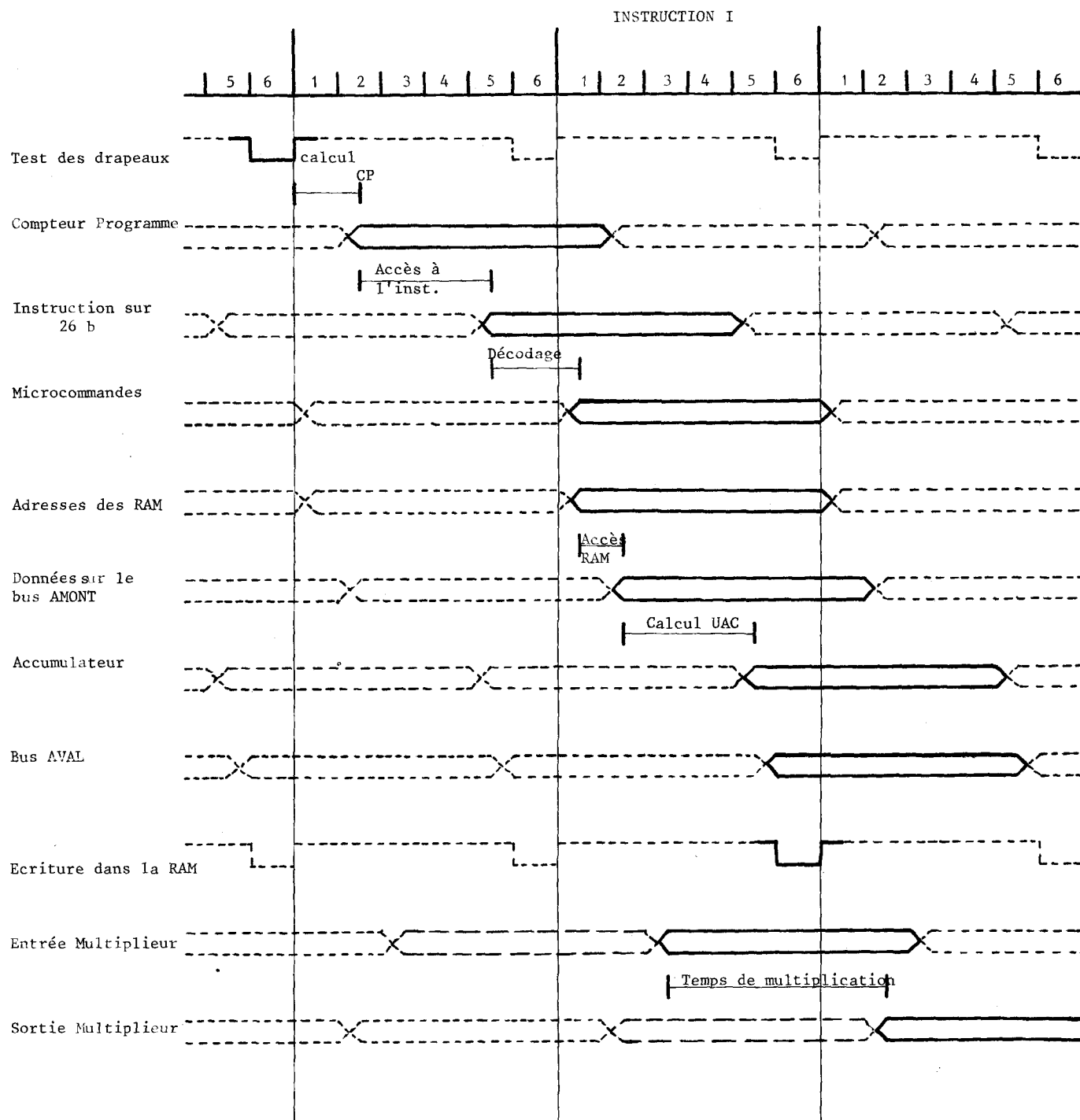


Fig. 2. — Le séquençage interne du μ PTS.

On considère le cycle machine où l'instruction est exécutée. Pour cela il faut anticiper le calcul du compteur programme puis l'accès à l'instruction proprement dite, cette instruction doit être disponible à la phase 5 du cycle précédant l'exécution, puis décodée de façon à ce que les microcommandes ainsi que les adresses des RAM nécessaires à l'exécution de cette instruction soient prêtes en phase 1 du cycle lui-même. La phase 2 est employée à accéder aux données elle-même (l'accès des mots en RAM étant le plus lent) les phases 3 et 4 sont nécessaires aux calculs arithmétiques dans l'UAL (l'addition est l'opération la plus longue) l'accumulateur est chargé en phase 5 et le résultat est ensuite disposé sur le bus AVAL pendant la phase 6 pour une réécriture en RAM. Ce déroulement de ces opérations se répète ainsi à chaque cycle avec une légère avance du séquenceur sur les parties opératives mais cette avance n'est pas sensible pour le programmeur. Par contre, le multiplieur est un bloc dont le temps de traversée 250 ns est voisin du temps de cycle (pour l'addition seules 100 ns suffisent) il est nécessaire de faire travailler le multiplieur et les autres parties opératives en parallèle de manière à éviter d'attendre le résultat de la multiplication pour continuer les actions de l'instruction en cours. Par conséquence, l'utilisateur du cycle instruction fournit les entrées du multiplieur, par contre le résultat de la multiplication effectuée avec les données du cycle instruction précédent est disponible et peut être utilisé dans l'unité arithmétique et logique. Cette particularité ne crée qu'un ralentissement mineur pour une série de calculs répétitifs.

5. Le port B

Le port B est un bloc d'interface entre les bus internes AMONT et AVAL et l'extérieur. Vu de l'intérieur, il comporte un ensemble de six registres accessibles en lecture ou écriture *via* les bus AMONT et AVAL, les mouvements des données sont contrôlés par le programme du UPTS. Ce port B présente, vu de l'extérieur, un bus de 16 bits (B0..B15) un ensemble de sept broches de contrôle (ASB, DSB, R/WB, DTACKB, BR, BGIN, BGOUT) et deux horloges sur deux broches (T1, T2).

Ce protocole permet l'échange de données entre le UPTS et l'extérieur sous le contrôle du programme interne. Le bus de 16 bits avec un dialogue de type boîte aux lettres permet un échange d'un mot de 16 bits à un débit de 1,3 Mmots/s. La gestion du protocole d'accès au bus est assurée par le port B sous forme d'automates câblés libérant le programme principal des tâches élémentaires de gestion du bus. Le port B comporte aussi un automate câblé convertisseur entre la loi MIC (loi A) et la loi linéaire libérant ainsi le programme principal de cette tâche. Deux horloges distinctes et identiques assurent la génération d'impulsions présentées sur les broches T1 ou T2; dans un autre mode ces deux broches peuvent déclencher en interne des mises à UN de drapeaux testables par le logiciel du UPTS.

6. Le port C

Ce port entièrement distinct du port B est un organe de contrôle du UPTS. Il présente vu de l'extérieur un bus

de 16 bits C0..C15 et un ensemble de quatre broches (ASC, DSC, R/WC, DTACKC) de contrôle des échanges et de six broches de contrôle (RST, INT, XTL, EXTL, SYNC, EXT) et un port de sortie du compteur de programme CP0..CP10. Ce port présente trois utilisations distinctes :

- l'entrée des instructions stockées à l'extérieur du UPTS;
- l'accès en écriture ou lecture d'une grande partie des registres internes;
- l'accès direct en écriture et lecture des deux mémoires RAM internes à une vitesse élevée (un transfert par cycle de 300 ns).

L'entrée des instructions de 26 bits au rythme nominal d'un cycle tous les 300 ns est réalisé avec un protocole très simplifié en deux temps 16 bits + 10 bits. Le temps maximal d'accès du support externe des instructions est de 100 ns.

L'accès aux registres internes est réalisé par un protocole compatible avec le dialogue du port B d'un même UPTS ou d'un autre distinct ou par un protocole avec acquittement compatible avec la plupart des microprocesseurs standards.

L'accès direct aux mémoires internes du UPTS est réalisé après une courte phase d'initialisation selon un protocole réduit avec acquittement permettant une large gamme de vitesse de transfert jusqu'à une vitesse maximale de 3,33 Mmots/s.

7. Le jeu d'instructions du UPTS (tableau 1)

Le circuit est contrôlé par un programme stocké soit dans la mémoire ROM interne, soit sur un support externe (PROM, ROM, RAM, EPROM, etc.). Ce programme est constitué d'un ensemble d'instructions codées sur un mot de longueur fixe (26 bits). Chaque instruction est exécutée pendant un cycle du UPTS (attention un mot quelconque de 26 bits ne constitue pas nécessairement une instruction valide et peut donc donner un résultat erroné). Le développement d'un programme peut être réalisé directement en binaire mais il est préférable d'employer les outils de développement disponibles et de travailler en SPL 1 (signal processing langage n° 1). Les instructions peuvent être aussi décrites en mnémoniques mais leur emploi est lourd.

7.1. LES REGISTRES DU UPTS

La figure 3 représente l'ensemble des registres du UPTS avec leurs tailles et noms respectifs et avec une indication des positions relatives des registres entre eux. Les noms des registres sont employés directement par le langage SPL 1.

Exemple : $A := A + P$; $C (+1) := A$.

7.2. LES TYPES D'INSTRUCTIONS (cf. fig. 4)

Une instruction codée sur 26 bits est classée en quatre types différents regroupant chacun un ensemble d'instruc-

APPLICATIONS

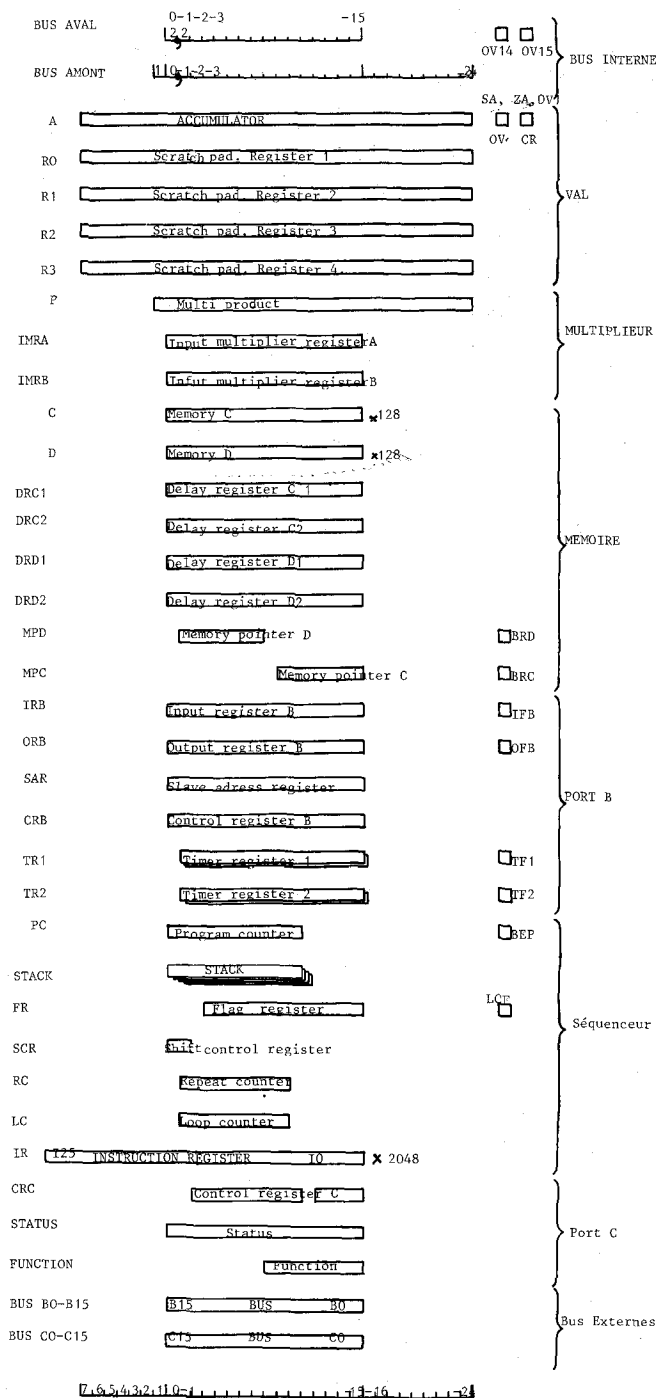


Fig. 3. — Les registres, mémoires et drapeaux du PTS.

tions ayant des caractéristiques communes :

TYPE 1 = les chargements immédiats d'une donnée.

TYPE 2 = les ruptures de séquence.

TYPE 3 = les opérations arithmétiques et logiques.

TYPE 4 = les transferts généralisés.

7.2.1. Les chargements immédiats

Une donnée indiquée en adressage immédiat par 16 bits de l'instruction (I0-I15) est chargée dans le registre ou le mot mémoire indiqué comme destination. En outre, dans le cas de chargement en mémoire un incrément préalable du pointeur correspondant peut être indiqué.

7.2.2. Les ruptures de séquences

Les différentes ruptures de séquences sont regroupées dans le type 2, le point d'arrivée du compteur programme est une donnée immédiate codée sur 11 bits de l'instruction (I5-I15). En outre un transfert (cf. type 3) peut être exécuté en même temps de façon indépendante du test (ce transfert est réalisé que le saut soit effectif ou non, une modification des drapeaux par ce transfert n'est pas prise en compte pour cette rupture de séquence).

7.2.3. Les opérations arithmétiques et logiques

Ce type d'instruction permet d'activer les opérateurs de calcul du UPTS : pointeurs mémoire, mémoires, multiplieur, unité arithmétique et logique. Ce type d'instruction contient trois champs principaux qui seront décrits séparément : le champ multiplieur (I5-I7), le champ opération arithmétique et logique (I8-I15) et le champ transfert (I0-I4 commun au type 2 et 3).

7.2.4. Les transferts généralisés

Les instructions de type 2 et 3 autorisent un nombre restreint de transferts de données à l'intérieur du UPTS (les plus usités dans les algorithmes de traitement du signal).

Pour obtenir un jeu d'instruction plus puissant, il est possible d'utiliser une instruction de type 4 pour transférer une donnée entre deux registres quelconques du UPTS, aucune opération arithmétique ou rupture de séquence n'est alors réalisée pendant une instruction de type 4 (sauf lors d'un transfert d'un mot vers le registre PC donnant une rupture de séquence). L'instruction indique la source (I10-I15) et la destination (I21-I16) de la donnée.

IR	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Type 1	0	1	0	0	0	0	CHARGEMENTS IMMÉDIATS				DONNÉE SUR 16 BIT															
Type 2	0	0	RUPTURES DE SÉQUENCE					DONNÉE SUR 11 BIT										TRANSFERT								
Type 3	1	ADRESSAGE					OPÉ. AR. ET LOG.							MULT.			TRANSFERT									
Type 4	0	1	1	0	DESTINATION				SOURCE																	

Fig. 4. — Les types d'instruction du μ PTS.

8. Méthodologie de développement d'une application pour le UPTS

L'utilisation d'un microprocesseur nécessite l'emploi d'outils de développements sophistiqués et d'une méthodologie de travail propre à donner un produit complet: le circuit lui-même avec son programme. Ce produit peut être ainsi introduit dans une application particulière et apporte la puissance et la souplesse du UPTS. Ce chapitre décrit les outils de développement entourant le

UPTS: langage SPL 1, l'analyseur et le générateur de code, le simulateur et l'émulateur matériel. Ensuite, on trouvera un exemple simple d'application.

La figure 5 donne l'organigramme d'une méthodologie de développement d'une application employant le UPTS en vue de résoudre un problème particulier. Le problème étant clairement spécifié, il est nécessaire dans un premier temps de décrire l'algorithme qui sera employé (cette description peut consister en un programme écrit en FORTRAN ou en PASCAL et validé par quelques essais sur un ordinateur). Ensuite, le futur utilisateur du UPTS

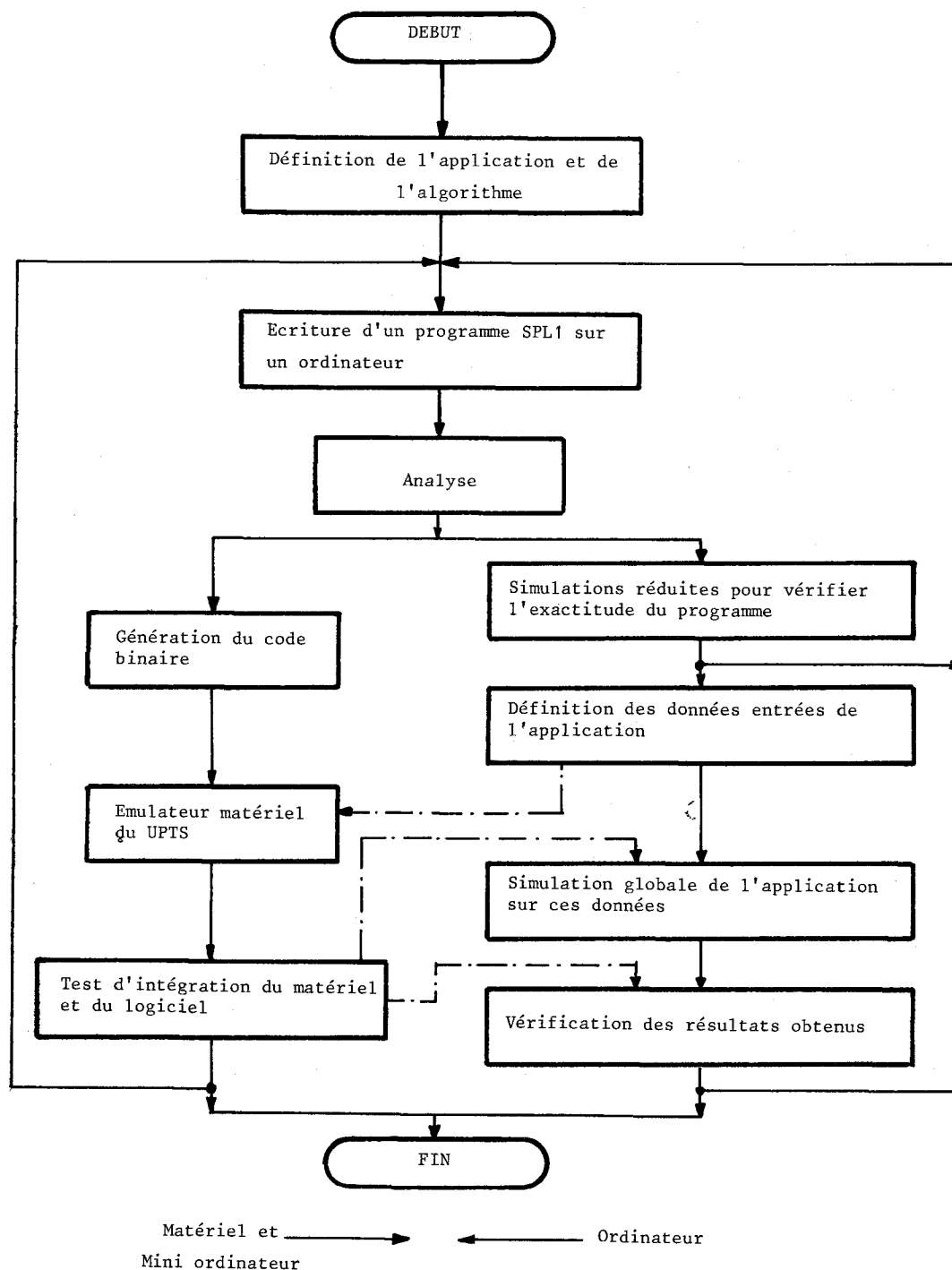


Fig. 5. — Organigramme donnant une méthodologie de développement d'une application.

décrit ce même algorithme dans le langage de programmation du circuit. Ce langage appelé SPL 1 (signal processing language n° 1) est particulier à ce microprocesseur mais est d'un emploi et d'une grammaire souple et efficace. L'analyseur disponible sur plusieurs matériels informatiques (VAX/VMS, IBM 43-VM CMS, Micromega-Unix, sous CP/M) permet de traiter ce langage et de fournir un fichier servant dans un premier temps à la simulation du UPTS au niveau registre. Cette simulation permet de vérifier le programme dans un premier temps en utilisant les nombreuses commandes d'aide à la mise au point puis en utilisant les facilités de description des entrées ou des sorties du UPTS pour une simulation plus globale de l'application. Le programme étant parfaitement au point grâce à ce simulateur disponible sur VAX/VMS, IBM 43-VM CMS, Micromega-Unix, il est alors possible de produire le code binaire correspondant aux instructions du programme écrit en SPL 1. Le futur UPTS est alors remplacé par l'émulateur qui reçoit le code ainsi produit et l'utilisateur a la possibilité de finir la mise au point de son application et de vérifier l'intégration parfaite du matériel et du logiciel. En outre, les données peuvent aisément être échangées entre l'émulateur temps réel du circuit et les outils logiciels de simulation ce qui autorise toutes vérifications croisées et donne d'avantage de souplesse et d'intégration à ces ensembles d'outils de développement.

9. Un exemple d'utilisation, le filtrage transversal

9.1. LE FILTRAGE TRANSVERSAL

Dans une opération de filtrage transversal, la suite $y(n)$ des échantillons de sortie est reliée à la suite $x(n)$ des échantillons d'entrée par la relation :

$$y(n) = \sum_{i=0}^{N-1} a_i \cdot x(n-i),$$

a_i , coefficient du filtre; N , ordre du filtre.

L'exemple traité est celui d'un filtre passe-bas, à phase linéaire sa fréquence de coupure à 3 dB est à 800 Hz. Le filtrage sera réalisé à une fréquence d'échantillonnage de 9 600 Hz.

9.2. PROGRAMME D'EXÉCUTION SPL 1

Le programme exécutable en SPL 1 sera organisé de la façon décrite par l'organigramme de la figure 6.

9.2.1. Remarque

L'initialisation du programme comprend : l'initialisation du μ PTS et le chargement des coefficients du filtre en mémoire. La figure 7 représente la liste du code exécutable en SPL 1 correspondant à l'organigramme de la figure 6.

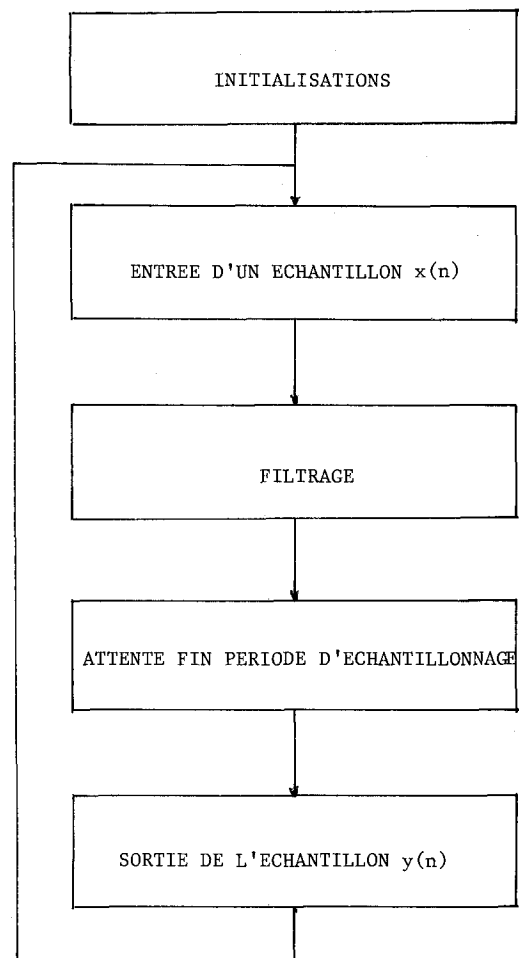


Fig. 6.

9.3. LA SIMULATION

Afin de vérifier si le filtre ainsi réalisé correspond aux spécifications, il suffit d'en étudier la réponse impulsionnelle (fig. 8) et la transformée de Fourier (fig. 9).

9.4. LE FILTRE EN TEMPS RÉEL

Transféré en ROM, le code étudié précédemment permet d'obtenir les résultats de la photo 2.

Annexe

Fiche signalétique du produit

Deux mémoires RAM de 128 mots de 16 bits (une lecture et une écriture de chaque RAM par cycle).

Une mémoire ROM de 256 instructions de 26 bits.

Un multiplieur câblé 16 bits $\times$ 16 bits $\dots$ > 26 bits (temps de traversée 300 ns).

Une unité arithmétique et logique de 32 bits (une opération par cycle).

Un accumulateur et quatre registres de brouillon de 32 bits.

APPLICATIONS

DBA1: BALESTRO.STAGE RIF.SOU;2

PAGE : 1
4-SEP-1984 10:13:19.20

```
*****
** SPL1 TRANSLATOR - VERSION : 2.0 SYSECA 1984 **
*****
***** TRANSLATION *****
*****
1 BEGIN
2 !-----!
3 ! INITIALISATIONS !
4 !-----!
5
6 !-----!
7 ! INITIALISATION MPTS !
8 !-----!
9 000 SAR:=H'2000' !adresse CAN,CNA
10 001 CRB:=H'0060' !registre controle port B
11 002 TR1:=347 !F echantillonnage = 9600 Hz
12
13 !-----!
14 ! COEFFICIENTS DU FILTRE !
15 !-----!
16 003 C(127)
17 004 C(+1):=-0.0038615
18 005 C(+1):=-0.0013109
19 006 C(+1):=-0.0007378
20 007 C(+1):=-0.0042494
21 008 C(+1):=-0.0080390
22 009 C(+1):=-0.0101650
23 00A C(+1):=-0.0085878
24 00B C(+1):=-0.0021384
25 00C C(+1):=-0.0085509
26 00D C(+1):=-0.0206032
27 00E C(+1):=-0.0292308
28 00F C(+1):=-0.0289718
29 010 C(+1):=-0.0154367
30 011 C(+1):=-0.0128516
31 012 C(+1):=-0.0533805
32 013 C(+1):=-0.0996940
33 014 C(+1):=-0.1427437
34 015 C(+1):=-0.1732454
35 016 C(+1):=-0.1842662
36
37 !-----!
38 ! ENTREE D'UN ECHANTILLON x(n) !
39 !-----!
40
41 017 ENT: A:=IRB; D(O):=A
42
43 !-----!
44 ! FILTRAGE !
45 !-----!
46 018 P:=C(O)*D; A:=0
47 019 REPEAT 18
48 01A P:=C(+1)*D(+1); A:=P+A; D:=DRD1
```

DBA1: BALESTRO.STAGE RIF.SOU;2

PAGE : 2
4-SEP-1984 10:13:19.20

```
49 01B REPEAT 18
50 01C P:=C(-1)*D(+1); A:=P+A; D:=DRD1
51 01D A:=P+A
52
53 !-----!
54 ! ATTENTE FIN PERIODE ECHANTILLONNAGE !
55 !-----!
56
57 01E FIL: IF TF1=0 THEN GOTO FIL
58
59 !-----!
60 ! SORTIE D'UN ECHANTILLON y(n) !
61 !-----!
62
63
64 01F GOTO ENT; ORB:=A SAT
65 END
```

```
! LABEL TABLE !
! NAME ADDRESS !
! ENT 017 !
! FIL 01E !
```

NUMBER OF LINES READ : 65
NUMBER OF TRANSLATED LINES : 32

*** NO ERRORS **

Fig. 7.

Deux réseaux de décalage de quatre à sept positions (en entrée et sortie de l'UAL).

Un bus de transfert de données sur 16 bits (un transfert en 700 ns) avec 64 K mots adressables (22,9 Mbits/s).

REPONSE TEMPORELLE

TEMPS	AMPLITUDE	-0.29E-01	0.77E-01	0.18E+00
0.000000E+00	-0.387573E-02	+		
0.104167E-03	-0.131226E-02	+		
0.208333E-03	0.732422E-03	+		
0.312500E-03	0.421143E-02	+		
0.416667E-03	0.799561E-02	+		
0.520833E-03	0.101318E-01	+		
0.625000E-03	0.854492E-02	+		
0.729167E-03	0.210571E-02	+		
0.833333E-03	-0.854492E-02	+		
0.937500E-03	-0.205994E-01	+		
0.104167E-02	-0.292358E-01	+		
0.114583E-02	-0.289612E-01	+		
0.125000E-02	-0.154419E-01	+		
0.135417E-02	0.128174E-01	+		
0.145833E-02	0.533447E-01	+		
0.156250E-02	0.996704E-01	+		
0.166667E-02	0.142700E+00	+		
0.177083E-02	0.173218E+00	+		
0.187500E-02	0.184235E+00	+		
0.197917E-02	0.173218E+00	+		
0.208333E-02	0.142700E+00	+		
0.218750E-02	0.996704E-01	+		
0.229167E-02	0.533447E-01	+		
0.239583E-02	0.128174E-01	+		
0.250000E-02	-0.154419E-01	+		
0.260417E-02	-0.289612E-01	+		
0.270833E-02	-0.292358E-01	+		
0.281250E-02	-0.205994E-01	+		
0.291667E-02	-0.854492E-02	+		
0.302083E-02	0.210571E-02	+		
0.312500E-02	0.854492E-02	+		
0.322917E-02	0.101318E-01	+		
0.333333E-02	0.799561E-02	+		
0.343750E-02	0.421143E-02	+		
0.354167E-02	0.732422E-03	+		
0.364583E-02	-0.131226E-02	+		
0.375000E-02	-0.387573E-02	+		
0.385417E-02	0.000000E+00	+		
0.395833E-02	0.000000E+00	+		
0.406250E-02	0.000000E+00	+		
0.416667E-02	0.000000E+00	+		
0.427083E-02	0.000000E+00	+		
0.437500E-02	0.000000E+00	+		

Fig. 8.

ATTENUATION DE LA FFT DU SIGNAL

FREQUENCE	ATTENUATION	-0.91E+02	-0.46E+02	-0.66E-02
0.000000E+00	-0.662623E-02			
0.750000E+02	-0.208861E-01			
0.150000E+03	-0.562539E-01			
0.225000E+03	-0.917684E-01			
0.300000E+03	-0.100116E+00			
0.375000E+03	-0.660036E-01			
0.450000E+03	-0.122942E-01			
0.525000E+03	-0.183622E-01			
0.600000E+03	-0.216843E+00			
0.675000E+03	-0.773433E+00			
0.750000E+03	-0.186809E+01			
0.825000E+03	-0.369238E+01			
0.900000E+03	-0.646987E+01			
0.975000E+03	-0.105113E+02			
0.105000E+04	-0.163582E+02			
0.112500E+04	-0.253082E+02			
0.120000E+04	-0.443810E+02			
0.127500E+04	-0.448190E+02			
0.135000E+04	-0.648729E+02			
0.142500E+04	-0.456244E+02			
0.150000E+04	-0.469526E+02			
0.157500E+04	-0.600369E+02			
0.165000E+04	-0.453263E+02			
0.172500E+04	-0.463677E+02			
0.180000E+04	-0.914042E+02			
0.187500E+04	-0.464090E+02			
0.195000E+04	-0.451960E+02			
0.202500E+04	-0.554372E+02			
0.210000E+04	-0.491756E+02			
0.217500E+04	-0.446378E+02			
0.225000E+04	-0.488243E+02			
0.232500E+04	-0.568511E+02			
0.240000E+04	-0.454469E+02			

Fig. 9.

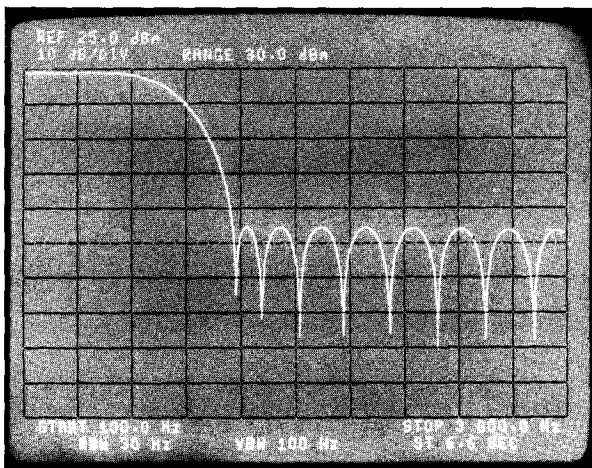
Un bus de contrôle de 16 bits (arrêt du programme, accès aux registres internes, redémarrage).

Une fonction DMA des mémoires RAM internes avec un débit de 3,33 mots/s (53,3 Mbits/s).

Un niveau d'interruption masquable.

Quatre niveaux de sous-programmes enchaînés.

APPLICATIONS



► Photo 2.

L'extension du programme à l'extérieur jusqu'à 2 K instructions.

Un convertisseur MIC → linéaire et linéaire → MIC câblé et autonome.

Deux horloges fournissant des impulsions à l'extérieur.

Deux broches directement testables par le programme.

NMOS 3 μ m 80 000 transistors 8,6 mm $\times$ 9,4 mm = 80 mm². 64 broches DIL.

Alimentation 0,5 V-1,3 W.

Alain ROSET
CNET, GRENOBLE

```
*****
**TRADUCTEUR DU LANGAGE SPL1 - VERSION : 1.0 SYSECA 1983*
*****
```

```
*****
***** T R A D U C T I O N *****
*****
```

```
1      !
2      !
3      !
4      BEGIN
5      !
6      !LES CHARGEMENTS IMMEDIATS: TYPE 1 (I10-I25)
7      !
8      000      A := 0.54
9      001      C := 0.54
10     002      C(+1) := 0.54
11     003      CRB := 0.54
12     004      CRC := 0.54
13     005      D := 0.54
14     006      D(+1) := 0.54
15     007      IMRA := 0.54
16     008      IMRB := 0.54
17     009      LC := 54
18     00A      ORB := 0.54
19     00B      RC := 54
20     00C      SAR := 0.54
21     00D      SHIFT /2
22     00E      SHIFT /4
23     00F      SHIFT AUTO
24     010      SHIFT STOP
25     011      TR1 := 0.54
26     012      TR2 := 0.54
27     !
28     !LES RUPTURES DE SEQUENCE: TYPE 2 (I5-I25)
29     !
30     013      IF A = 0 THEN GOTO FIN      !GOSUB FIN 00 RETURN 00 ENDT
31     014      IF A <> 0 THEN GOTO FIN      !GOSUB FIN 00 RETURN 00 ENDT
32     015      IF CR = 0 THEN GOTO FIN      !GOSUB FIN 00 RETURN 00 ENDT
33     016      IF CR = 1 THEN GOTO FIN      !GOSUB FIN 00 RETURN 00 ENDT
34     017      IF IFB = 0 THEN GOTO FIN      !GOSUB FIN 00 RETURN 00 ENDT
35     018      IF IFB = 1 THEN GOTO FIN      !GOSUB FIN 00 RETURN 00 ENDT
36     019      IF LC = 0 THEN GOTO FIN      !GOSUB FIN 00 RETURN 00 ENDT
37     01A      IF LC <> 0 THEN GOTO FIN      !GOSUB FIN 00 RETURN 00 ENDT
38     01B      IF LC = 0 THEN C(+1) GOTO FIN !GOSUB FIN 00 RETURN 00 ENDT
39     01C      IF LC <> 0 THEN C(+1) GOTO FIN !GOSUB FIN 00 RETURN 00 ENDT
40     01D      IF LC = 0 THEN D(+1) GOTO FIN !GOSUB FIN 00 RETURN 00 ENDT
41     01E      IF LC <> 0 THEN D(+1) GOTO FIN !GOSUB FIN 00 RETURN 00 ENDT
42     01F      IF LC = 0 THEN CD(-1) GOTO FIN !GOSUB FIN 00 RETURN 00 ENDT
43     020      IF LC <> 0 THEN CD(-1) GOTO FIN !GOSUB FIN 00 RETURN 00 ENDT
44     021      IF OFB = 0 THEN GOTO FIN      !GOSUB FIN 00 RETURN 00 ENDT
45     022      IF OFB = 1 THEN GOTO FIN      !GOSUB FIN 00 RETURN 00 ENDT
46     023      IF OV = 0 THEN GOTO FIN      !GOSUB FIN 00 RETURN 00 ENDT
47     024      IF OV = 1 THEN GOTO FIN      !GOSUB FIN 00 RETURN 00 ENDT
48     025      IF OV1 = 0 THEN GOTO FIN      !GOSUB FIN 00 RETURN 00 ENDT
```

APPLICATIONS

```

49 026      IF OV1 = 1 THEN GOTO FIN      !GOSUB FIN ou RETURN ou ENDIT
50 027      IF OV14 = 0 THEN GOTO FIN     !GOSUB FIN ou RETURN ou ENDIT
51 028      IF OV14 = 1 THEN GOTO FIN     !GOSUB FIN ou RETURN ou ENDIT
52 029      IF OV15 = 0 THEN GOTO FIN     !GOSUB FIN ou RETURN ou ENDIT
53 02A      IF OV15 = 1 THEN GOTO FIN     !GOSUB FIN ou RETURN ou ENDIT
54 02B      IF SA = 0 THEN GOTO FIN       !GOSUB FIN ou RETURN ou ENDIT
55 02C      IF SA = 1 THEN GOTO FIN       !GOSUB FIN ou RETURN ou ENDIT
56 02D      IF TF1 = 0 THEN GOTO FIN     !GOSUB FIN ou RETURN ou ENDIT
57 02E      IF TF1 = 1 THEN GOTO FIN     !GOSUB FIN ou RETURN ou ENDIT
58 02F      IF TF2 = 0 THEN GOTO FIN     !GOSUB FIN ou RETURN ou ENDIT
59 030      IF TF2 = 1 THEN GOTO FIN     !GOSUB FIN ou RETURN ou ENDIT
60 031      IF ZA = 0 THEN GOTO FIN       !GOSUB FIN ou RETURN ou ENDIT
61 032      IF ZA = 1 THEN GOTO FIN       !GOSUB FIN ou RETURN ou ENDIT
62 033 FIN:  ENDIT
63 034      GOTO FIN
64 035      GOSUB FIN
65 036      RETURN
66 037      REPEAT 5
67 038      REPEAT
68          !
69          ILES TRANSFERTS SIMPLES: TYPE 2/3 (I0-I5)
70          !
71 039      C := A
72 03A      CD := A
73 03B      D := A
74 03C      C := A SAT
75 03D      CD := A SAT
76 03E      D := A SAT
77 03F      C := A /S
78 040      CD := A /S
79 041      ORB := A
80 042      ORB := A SAT
81 043      ORB := A /S
82 044      D := A /S
83 045      C := DRC1
84 046      C := DRC2
85 047      D := DRD1
86 048      D := DRD2
87 049      R0 := A
88 04A      R1 := A
89 04B      R2 := A
90 04C      R3 := A
91 04D      TR1 := A
92 04E      TR2 := A
93          !
94          ILES MULTIPLICATIONS: TYPE 3 (I5-I7)
95          !
96 04F      P := A*D
97 050      P := C*A
98 051      P := C*C
99 052      P := C*D
100 053      P := C*IMRB
101 054      P := D*D
102 055      P := IMRA*D
103 056      P := IMRA*IMRB
104          !
105          ILES OPERATIONS ARITHMETIQUES ET LOGIQUES: TYPE 3 (I8-I15)
106          !
107 057      A := 0
108 058      A := -A
109 059      A := A+1
110 05A      A := A / 2
111 05B      A := A * 2
112 05C      A := A * 2 + CR
113 05D      A := ABS A
114 05E      A := -ABS A
115 05F      A := +(CR) A
116 060      A := -(CR) A
117 061      A := NOT A
118 062      A := R0
119 063      A := R1
120 064      A := R2
121 065      A := R3
122 066      CR := 0
123 067      CR := 1
124          !
125 068      A := C      ! C ou D ou IRB ou P
126 069      A := C /S   ! C ou D ou IRB ou P
127 06A      A := C + ABS A ! C ou D ou IRB ou P

```

APPLICATIONS

```

128 06B      A := C - ABS A      ! C ou D ou IRB ou F
129 06C      A := C + (CR) A     ! C ou D ou IRB ou F
130 06D      A := C - (CR) A     ! C ou D ou IRB ou F
131          !
132 06E      A := C + A          ! C ou D ou IRB ou F; A ou R0,R1,R2,R3
133 06F      A := C - A          ! C ou D ou IRB ou F; A ou R0,R1,R2,R3
134 070      A := C /S + A       ! C ou D ou IRB ou F; A ou R0,R1,R2,R3
135 071      A := C /S - A       ! C ou D ou IRB ou F; A ou R0,R1,R2,R3
136 072      A := C NOR A        ! C ou D ou IRB ou F; A ou R0,R1,R2,R3
137 073      A := C NAND A       ! C ou D ou IRB ou F; A ou R0,R1,R2,R3
138 074      A := C XOR A        ! C ou D ou IRB ou F; A ou R0,R1,R2,R3
139          !
140          !LES TRANSFERTS GENERALISES: TYPE 4 (110-125)
141          !
142 075      A := source
143 076      dest := A
144 077      A /C := source
145 078      dest := A /C
146 079      A /S := source
147 07A      dest := A /S
148 07B      C := source
149 07C      dest := C
150 07D      C(+1) := source
151 07E      dest := C(+1)
152 07F      D := source
153 080      dest := D
154 081      D(+1) := source
155 082      dest := D(+1)
156 083      IRB := source
157 084      dest := IRB
158 085      IRB /F := source
159 086      dest := IRB /F
160 087      ORB := source
161 088      dest := ORB
162 089      ORB /F := source
163 08A      dest := ORB /F
164 08B      MPCD := source
165 08C      dest := MPCD
166 08D      PC := source
167 08E      dest := PC
168 08F      TR1 := source
169 090      dest := TR1
170 091      TR2 := source
171 092      dest := TR2
172          !
173 093      dest := DRC1
174 094      dest := DRC2
175 095      dest := DRD1
176 096      dest := DRD2
177 097      dest := FR
178 098      dest := P /H
179          !
180 099      A /L := source
181 09A      CRB := source
182 09B      CRC := source
183 09C      LC := source
184 09D      MPC := source
185 09E      MPD := source
186 09F      RC := source
187 0A0      SAR := source
188          !
189          END

```

Un accord de coopération

Un accord de coopération entre THOMSON et Le CNET a été signé en juillet 1984 portant sur l'échantillonnage du microprocesseur de traitement du signal. Les partenaires s'engagent à conclure un ou plusieurs contrats de licence pour l'industrialisation du circuit dans les technologies les plus appropriées et pour les logiciels et matériels associés, avant l'échéance de l'accord en juillet 1985.

I L A B E L T A B L E I

NAME	ADDRESS
FIN	033

```

number of lines read      :      189
number of translated lines :      161
*****
*** NO ERRORS ***
*****

```

► Les instructions du μ PTS.