

Chapitre III: Processeurs de traitement numérique du signal (DSP)

Objectif du chapitre

Se familiariser avec les processeurs de traitement numérique du signal (DSP)

Contenu du chapitre

III.1. Domaine d'utilisation :

Les DSP (digital signal processor) sont, comme leur nom l'indique, des processeurs dédiés au traitement des signaux numériques sont nombreux et variés (traitements du son, de l'image, synthèse et reconnaissance vocale, analyse, compression de données, télécommunications, automatisme, etc.) Chacun de ces domaines nécessite un système de traitement numérique, dont le cœur est un (parfois plusieurs) DSP ayant une puissance de traitement adaptée, pour un coût économique approprié.

Voici quelques exemples des applications les plus courantes faisant appel aux DSP :

- ❖ **Dans le domaine des télécommunications** : modems, multiplexeurs, récepteurs de numérotation DTMF, télécopieurs, codeurs de parole GSM, modems radio...).
- ❖ **Dans le domaine des interfaces vocales** : codeurs vocaux pour répondeurs, reconnaissance automatique de parole, synthèse vocale
- ❖ **Dans le domaine militaire** : guidage des missiles, navigation, communications cryptées, traitement radar....
- ❖ **Dans le domaine des multimédias et du grand public** : compression des signaux audios (CD), compression des images, cartes multimédias pour PC, synthèse musicale, jeux ...
- ❖ **Dans le domaine médical** : compression d'image médicale (IRM, échographie...) , équipements de monitoring.
- ❖ **Dans le domaine de l'électronique automobile** : Equipements de contrôle moteur, aide à la navigation, commande vocale, détection de cliquetis pour avance à l'allumage...
- ❖ **Dans le domaine de l'automatisation et du contrôle de processus** : surveillance et commande de machine, contrôle de moteurs, robots, servomécanismes...
- ❖ **Dans le domaine de l'instrumentation** : Analyseurs de spectre, générateurs de fonction, interprétation de signaux sismiques...

III.2 Les spécificités des DSP :

Un DSP est un type particulier de microprocesseur. Il se caractérise par le fait qu'il intègre un ensemble de fonctions spéciales. Ces fonctions sont destinées à le rendre particulièrement performant dans le domaine du traitement numérique du signal. Comme un microprocesseur classique, un DSP est mis en œuvre en lui associant de la mémoire (RAM, ROM) et des périphériques. Un DSP typique

a plutôt vocation à servir dans des systèmes de traitements autonomes. Il se présente donc généralement sous la forme d'un microcontrôleur intégrant, selon les marques et les gammes des constructeurs, de la mémoire, des timers, des ports séries synchrones rapides, des contrôleurs DMA, des ports d'E/S divers.

III.2.1 Le cœur d'un système de traitement numérique du signal

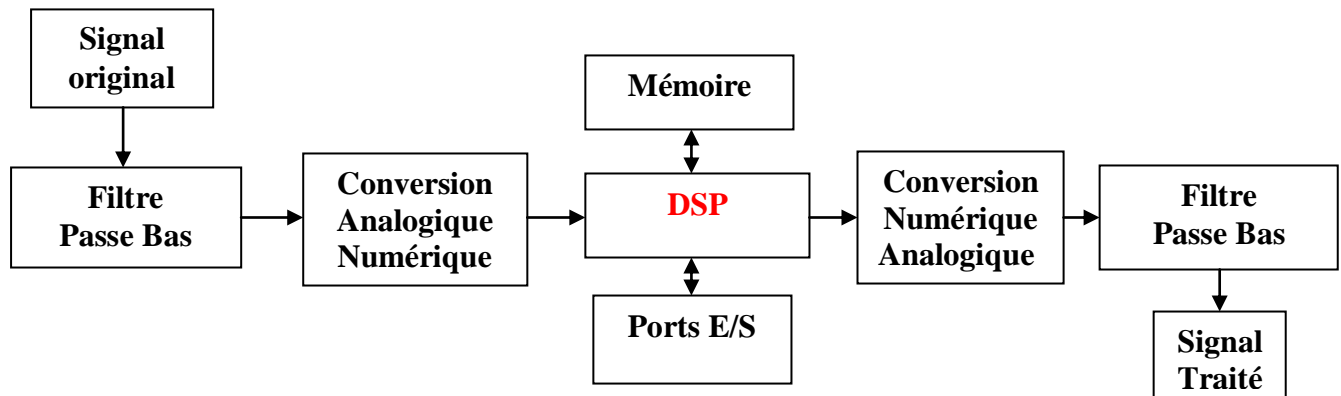


Figure III.1 : Chaîne complète typique d'un système de traitement numérique du signal.

III.3 Principales distinctions entre un microprocesseur et un DSP

En quoi un DSP est-il différent d'un microprocesseur ? Pour répondre à cette question, il nous faut prendre en considération les éléments suivants.

III.3.1. L'opération MAC

Après avoir été numérisé, le signal se présente sous la forme d'une suite de valeurs numériques discrètes. Cette suite de valeurs (ou échantillons) est apte à être stockée et traitée par un système informatique. Par nature, le traitement numérique du signal revient à effectuer essentiellement des opérations arithmétiques de base du type $A = (B \times C) + D$.

Un microprocesseur classique va nécessiter plusieurs cycles d'horloge pour effectuer un tel calcul, par exemple, un 68000 à besoin de :

10 cycles d'horloge pour effectuer une addition,

70 cycles d'horloge pour effectuer une multiplication.

Soit 80 cycles pour seulement calculer A. Si ce temps est admissible dans des applications informatiques courantes, il n'est pas acceptable pour faire du traitement rapide du signal. Les DSP sont donc conçus pour optimiser ce temps de calcul. À cet effet, ils disposent de fonctions optimisées permettant de calculer A beaucoup plus rapidement.

Dans la pratique, la plupart des DSP ont un jeu d'instructions spécialisé permettant de lire en mémoire une donnée, d'effectuer une multiplication puis une addition, et enfin d'écrire en mémoire le résultat, le tout en un seul cycle d'horloge. Ce type d'opération est nommé MAC, de l'anglais Multiply and Accumulate.

Effectuer une opération MAC en un seul cycle n'est malgré tout pas satisfaisant si le cycle d'horloge est trop « long ». Le principal objectif d'évolution des DSP a toujours été d'améliorer le temps de calcul d'un MAC. La **Figure III.2** présente le progrès réalisé dans ce domaine depuis plus de trois décennies.

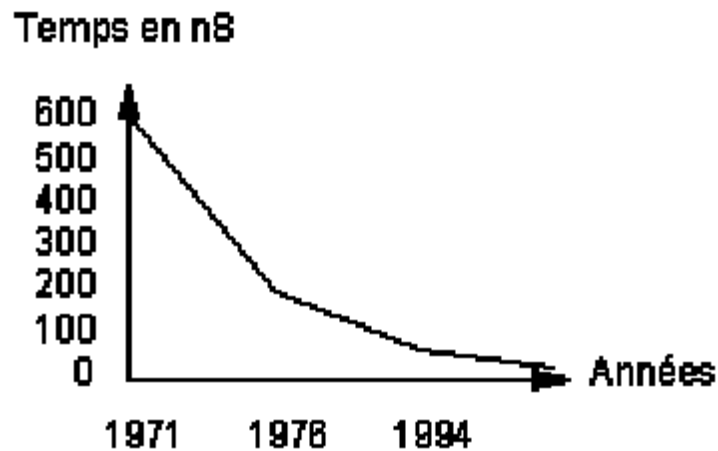


Figure III.2 : Evolution du temps d'exécution d'une opération MAC selon Texas Instrument.

Outre le temps d'exécution d'une opération MAC, un autre problème se pose. L'opération MAC étant une multiplication suivie d'une addition, un débordement de l'accumulateur est toujours possible. Pour contourner ce problème, certains DSP possèdent un accumulateur adapté au MAC. Ces accumulateurs ont un format spécial incorporant des bits supplémentaires (bits de garde) par rapport à la taille des données à manipuler. Les problèmes de débordements sont alors contournés, car un d'opérations MAC telles qu'un résultat excède la capacité élargie de l'accumulateur.

III.3.2. L'accès à la mémoire

Outre l'opération MAC, une autre caractéristique des DSP est leurs capacités à réaliser plusieurs accès mémoire en un seul cycle. Ceci permet à un DSP de chercher ranger le résultant du MAC précédent. Le gain de temps est évident. Toutefois, sur certains DSP basiques, ce type d'opération simultanée est généralement limité à des instructions spéciales. Ces instructions utilisent un mode d'adressage restreint, c'est à dire ne portant que sur de la mémoire vive intégrée au DSP.

Les modes d'adressages des données sont un point particulier des DSP. Un DSP peut posséder plusieurs unités logiques de génération d'adresse, travaillant en parallèle avec la logique du cœur du DSP. Une unité logique de génération d'adresse est paramétrée une seule fois via les registres appropriés. Elle génère alors toute seule, en parallèle avec l'exécution d'une opération arithmétique, les adresses nécessaires à l'accès des données.

Ceci permet non seulement de réaliser les accès mémoires simultanés en un seul cycle, comme décrit plus haut, mais également d'incrémenter automatiquement les adresses générées. Ce mode d'adressage particulier, généralement appelé adressage indirect par registre avec post (ou pré) incrément, est très utilisé pour effectuer des calculs en mémoire une instruction et ses données

réalisant un MAC, et simultanément, d'y répéter sur une série de données rangées séquentiellement en mémoire.

III.3.3 Contrôle du processeur – le pipeline (principe de fonctionnement) :

Nous prendrons l'exemple du TMS320C64XX. Si on considère les opérations nécessaires à l'exécution d'une instruction, elles peuvent se décrire ainsi :

PFetch	Positionnement de l'adresse de la mémoire programme et incrémentation du PC
Fetch	Recherche de l'instruction par accès à la mémoire programme et écriture dans le registre d'instruction
Decode	Décodage de l'instruction et des adresses de l'opérande
Access	Adressage des opérandes situés en mémoire par les bus
Read	Lecture des opérandes en mémoire donnée et positionnement des adresses pour l'écriture du résultat
Execute	Exécution de l'opération et écriture du résultat

Afin de gagner du temps lors de l'exécution de séries d'instruction, il est donc nécessaire d'optimiser ces différentes étapes en les parallélisant ou en les « pipelinant ».

Le principe retenu est celui utilisé dans les usines de production et qui consiste à découper le travail en tâches élémentaires (**Figure III.3**) :

Temps d'instruction	t	t+1	t+2	t+3	t+4	t+5	t+6	t+7	t+8
n	PF	F	D	A	R	E			
n+1		PF	F	D	A	R	E		
n+2			PF	F	D	A	R	E	
n+3				PF	F	D	A	R	E

PF : Pfetch
F : Fetch

D: Decode
A: Access

R: Read
E: Execute

Figure III.3 : Cycle d'exécution d'une instruction.

A la lecture de ce tableau, on peut noter qu'à partir de la 5eme instruction, à chaque nouveau cycle machine, une nouvelle instruction est réalisée au lieu d'une instruction tous les 5 cycles machines. Un programme est donc exécuté environ 5 fois plus vite !

III.4 Architecture des processeurs :

L'architecture d'un microprocesseur, et donc d'un DSP, est un élément important qui conditionne directement les performances d'un processeur. Il existe deux types fondamentaux de structures, dites

« Von Neumann » et « Harvard », telles que présentées par la (**Figure III.4**) :

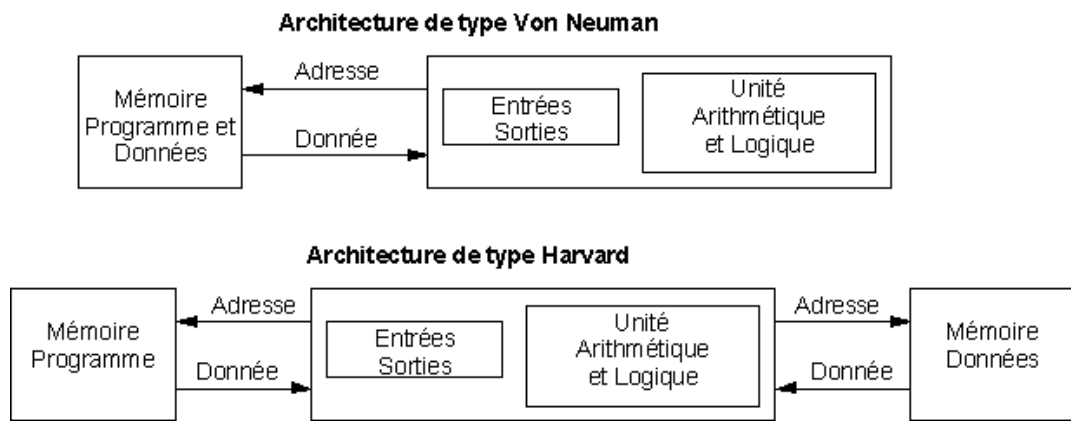


Figure III.4 : Représentation schématique des structures de Harvard et de Von Neumann.

III.4.1 structure de Von Neumann

Un microprocesseur basé sur une structure Von Neumann stocke les programmes et les données dans la même zone mémoire. Une instruction contient le code opératoire et l'adresse de l'opérande. Ce type de microprocesseur incorpore principalement deux unités logiques de base :

- ❖ l'Unité Arithmétique et Logique (ou ALU en anglais), chargée de réaliser les opérations centrales (de type multiplications, additions, soustractions, rotation, etc.),
- ❖ l'unité en charge des Entrées/Sorties, qui commande le flux de données entre le cœur du microprocesseur et les mémoires ou les ports.

III.4.2 structure de Harvard

Cette structure se distingue de l'architecture Von Neumann par le fait que les mémoires se fait via un chemin distinct. Cette organisation permet de transférer une instruction et des données simultanément, ce qui améliore les performances.

III.4.3. Utilisation de ces structures dans les DSP

L'architecture généralement utilisée par les microprocesseurs est la structure Von Neumann (exemples : la famille Motorola 68XXX, la famille Intel 80X86). L'architecture Harvard est plutôt utilisée dans des microprocesseurs spécialisés pour des applications temps réels, comme les DSP.

Pour réduire le coût de la structure Harvard, certains DSP utilisent l'architecture dite « Structure de Harvard modifiée ». À l'extérieur, le DSP ne propose qu'un bus de données et un bus d'adresse, comme la structure Von Neumann. Toutefois, à l'intérieur, la puce DSP dispose de deux bus distincts de données et de deux bus distincts d'adresses. Le transfert des données entre les bus externes et internes est effectué par multiplexage temporel. C'est le cas du TMS320C6711.

III.5 Les formats des données utilisés dans les DSP :

Un autre point essentiel des DSP est la représentation des nombres (les données) qu'ils peuvent manipuler. Il est possible de distinguer deux familles :

- ❖ Les DSP à virgule fixe : les données sont représentées comme étant des nombres fractionnaires à virgule fixe, (exemple -1.0 à +1.0), ou comme des entiers classiques. La représentation de ces nombres fractionnaires s'appuie la méthode du « complément à deux » (explication **Figure III.5** et **Figure III.6**). L'avantage de cette représentation (qui n'est qu'une convention des informaticiens) est de permettre facilement l'addition binaire de nombres aussi bien positifs que négatifs.

Poids des bits	MSB			LSB	
Valeurs	-2^3	2^2	2^1	2^0	
	-8	4	2	1	
	0	1	0	1	= 4+1 = +5
	1	1	0	1	= -8+4+1 = -3
le moins positif	0	0	0	1	= +1
le plus positif	0	1	1	1	= 4+2+1 = +7
le moins négatif	1	1	1	1	= -8+4+2+1 = -1
le plus négatif	1	0	0	0	= -8

Figure III.5 : Exemple de nombres codés sur 4 bits en complément à 2.

Poids des bits	MSB			LSB	
Valeurs	-2^0	2^{-1}	2^{-2}	2^{-3}	
	-1	½	¼	1/8	
	0	1	0	1	= 0,5+0,125 = 0,625
	1	1	0	1	= -1+0,5+0,125 = -0,375
le moins positif	0	0	0	1	= +0,125
le plus positif	0	1	1	1	= 0,5+0,25+0,125 = +0,875
le moins négatif	1	1	1	1	= -1+0,5+0,25+0,125 = -0,125
le plus négatif	1	0	0	0	= -1

Figure III.6 : Exemple de nombres fractionnaires codés sur 4 bits en complément à 2

- ❖ Les DSP à virgule flottante : les données sont représentées en utilisant une mantisse et un exposant. La représentation de ces nombres s'effectue selon la formule suivante: $n = \text{mantisse} \times 2^{\text{Exposant}}$. Généralement, la mantisse est un nombre fractionnaire (-1.0 à +1.0), et l'exposant est un entier indiquant la place de la virgule en base 2 (c'est le même mécanisme qu'en

base 10). Un exemple de nombre à virgule flottante est donné en figure 7 pour un nombre sur 8 bits avec une mantisse sur 5 bits et un exposant sur 3 bits.

Représentation binaire		Valeur décimale
Mantisse M	Exposant	
01110	010	1,75
01100	100	0,046875
10010	011	-7
01000	100	0,03125
01111	011	7,5

Figure III.7 : Représentation de nombres à virgule flottante pour une mantisse de 8 bits et un exposant sur 3 bits.

III.5.1 Les DSP à virgule flottante

Les DSP à virgule flottante sont plus souples et plus faciles à programmer que les DSP à virgule fixe. Un DSP comme le TMS320C30 manipule des nombres formés avec une mantisse de 24 bits et un exposant de 8 bits (taille de la donnée en mémoire : 32 bits). Les valeurs intermédiaires des calculs sont mémorisées dans des registres avec un format de 32 bits de mantisse et un exposant de 8 bits (taille du registre : 32 + 8 bits supplémentaires). La dynamique disponible est très grande, elle va de -1×2^{128} à $(1-2^{-23}) \times 2^{127}$, toutefois la résolution reste limitée à 24 bits au mieux. Outre les nombres fractionnaires, ce DSP sait également manipuler les entiers avec une précision de 32 bits.

La très grande dynamique proposée par les DSP à virgule flottante permet virtuellement de ne pas se soucier des limites des résultats calculés lors de la conception d'un programme. Cet avantage a cependant un prix, à savoir qu'un système basé sur un DSP à virgule flottante a un coût de fabrication supérieur par rapport à un système basé sur DSP à virgule fixe. La puce d'un DSP à virgule flottante nécessite à la fois une surface de silicium plus importante (cœur plus complexe), et un Le système revient donc plus cher (exemple : 2 x 32 broches ne serait-ce que pour les bus de données externes avec une architecture Harvard de base).

Un DSP à virgule flottante est plutôt adapté (sans être impératif) à des applications dans lesquelles :

- les coefficients varient dans le temps (exemple : les filtres adaptatifs),
- le signal et les coefficients ont besoin d'une grande dynamique,
- la structure mémoire est importante (exemple : traitement d'image), nombre de broches supérieur, car la mémoire externe est elle aussi au format 32 bits.
- la précision est recherchée sur toute une gamme dynamique importante (exemple : traitements audio phoniques de qualité professionnelle).

De part leurs facilités de programmation, ils peuvent également se justifier dans des projets où le temps et la facilité de développement sont des facteurs importants. On les trouve également dans des produits de faible volume de production, pour lesquels le prix du DSP n'est pas significatif.

III.5.2 Les DSP à virgule fixe

Un DSP à virgule fixe est un peu plus compliqué à programmer qu'un DSP à virgule flottante. Dans un DSP à virgule fixe typique comme le TMS320C25, les nombres sont codés sur 16 bits (rappel : des entiers classiques ou des fractionnaires).

Toutefois, sur ce DSP, les calculs sont effectués avec des accumulateurs de 32 bits. Lorsque les résultats doivent être stockés en mémoire, les 16 bits les moins significatifs sont perdus. Ceci permet de limiter les erreurs d'arrondis cumulatives. Il est toujours possible de stocker séparément en mémoire les 16 bits de poids faibles puis les 16 bits forts s'il n'y a plus de registres libres lors d'une étape de calcul.

Cette particularité n'est pas toujours disponible sur tous les DSP. Dans ce cas, les calculs requérant absolument une double précision sont réalisés en chaînant deux à deux des instructions spéciales manipulant des données 16 bits en simple précision, au détriment du temps d'exécution.

La précision des calculs est un point critique des DSP à virgule fixe, car le concepteur de programmes doit rester vigilant à chaque étape d'un calcul. Il doit rechercher la plus grande dynamique possible (c.à.d. exploiter au mieux la gamme des nombres disponibles), pour conserver une bonne précision des calculs, tout en évitant autant que faire ce peu les débordements du ou des accumulateurs. Les bits supplémentaires des accumulateurs (les bits de garde) prévus à cet effet permettent de réduire cette contrainte.

Les programmeurs contournent les limites des DSP à virgule fixe en déterminant à l'avance, et avec soin, la précision et la dynamique nécessaire (par méthode analytique ou avec des outils de simulation) pour réaliser leurs projets.

Il est également possible d'effectuer des opérations en virgule flottante dans un DSP à virgule fixe par le biais de routines logicielles adéquates. Cette approche est néanmoins pénalisante en temps d'exécution, même sur un DSP à virgule fixe très rapide.

Les DSP à virgule fixe sont les plus utilisés, car ils sont moins chers que les DSP à virgule flottante. On les trouve dans tous les produits de grande diffusion où le coût est un facteur important.

III.6 Les performances des DSP :

Plus que pour un microprocesseur classique, les performances d'un DSP conditionnent son domaine d'application. La plupart des DSP sont particulièrement destinés à des applications « temps réel » et spécialisées, c'est à dire des applications où le temps de traitement est bien sûr primordial, mais où la diversité des événements à traiter n'est pas notablement importante. De ce point de vue, l'approche DSP s'apparente plus à une étude « électronique » visant à réaliser une ou des fonctions de traitements de signal, que d'une approche informatique temps réel et/ou multitâche traditionnelle.

Il existe cependant des applications où le DSP assure à la fois des fonctions de traitements numériques du signal et les fonctions générales d'un microprocesseur au cœur d'un système informatique classique. Dans tous les cas, les performances du DSP sont critiques. Le concepteur d'un système à base de DSP doit évaluer d'une part la « puissance » nécessaire pour réaliser les traitements numériques voulus, et d'autre part les performances des DSP disponibles pour réaliser son application.

III.7 Les produits Texas instrument:

Les familles les plus récentes des DSP de Texas instrument sont :

- TMS320C54x, DSP format fixe;
- TMS320C20x, DSP format fixe;
- TMS320C24x, DSP format fixe;
- TMS320C62x, DSP format fixe à architecture VLIW (Very long instruction word);
- TMS320C67x, DSP format flottant à architecture VLIW ;

Ces nouvelles familles sont regroupées en 3 classes appelées plates-formes. Ces trois classes sont appelées :

- TMS320C5000, formée de la famille C54x et des C54xx ;
- TMS320C2000, formée des familles C20x et C24x ;
- TMS320C6000, formée des familles C62x et C67x ;

La **Tableau III.1** résume les principales caractéristiques de ces trois classes.

Application		Type de DSP	Caractéristiques
TMS320C6000 DSP hautes performances			
C62x	Applications exigeantes en vitesse : stations de base des réseaux de communications mobiles, équipement de radiodiffusion, réseaux informatiques	16 bits virgule fixe architecture VLIW	1200-2400 MIPS
C67x	Applications exigeantes en précision, dynamique et vitesse : Antennes adaptatives des stations de base, imagerie médicale, reconnaissance de parole, graphisme ...	32 bits virgule flottante architecture VLIW	600MFLOPS-1GFLOPS
TMS320C5000 DSP optimisés en consommation			
C54x	Applications de télécommunications exigeantes en coût, consommation, vitesse : terminaux mobile, voix sur IP, alphapages ...	16 bits virgule fixe	0,54 mW/MIPS 30-200 MIPS
TMS320C2000 DSP optimisés pour les applications de contrôle			
C20x	Applications de grands volume en téléphonie, électronique grand public, appareils photos numériques ou contrôleurs de disques durs ...	16 bits virgule fixe	PLL, UART, timers, mémoire flash intégrée 20-40MIPS
C24x	Applications de contrôle moteur, automatisation, robotique, contrôle d'appareils électroménagers... Bon compromis prix/performance	16 bits virgule fixe	Port série SCI, SPI et CAN, Convertisseur Analogique /numérique 20MIPS

Tableau III.1 : Principales caractéristiques des 3 familles de DSP.