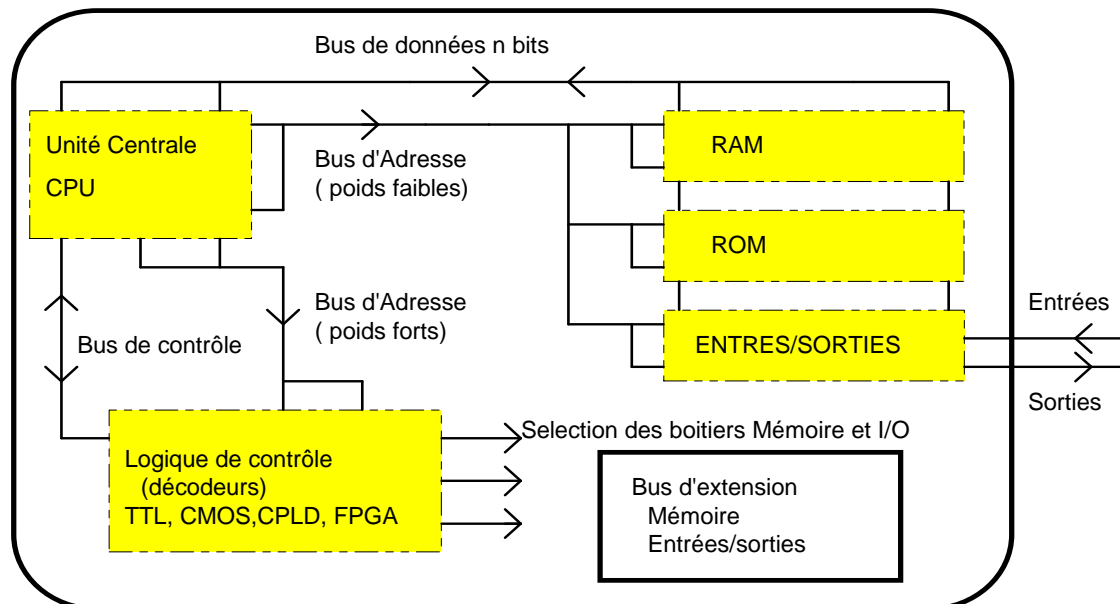


Architecture Générale d'un Système Microprocesseur ou Microcontrôleur

A) Principe généraux des unités centrales

1) Architecture générale d'un système

Tout système informatique peut être représenté par le schéma ci-dessous :



CPU veut dire : Central Processor Unit

Pour caractériser un système il faut préciser :

- Le type de CPU : 386 , 486, Pentium, 68k, Power PC, Monochip, DSP ...
- Les fréquences d'horloge.
- Les capacités mémoires RAM et ROM.
- Le nombre et la nature des dispositifs d'entrées sorties.
- La taille et le nombre des bus de données (8, 16, 32, 64)
- La taille le nombre des bus d'adresse (16, 24, 32, 64).
- Le type des bus d'extension : ISA, PCI, PCM, VME, PXI, Compac PCI, VXI...
- Les systèmes d'exploitation utilisés : MSDOS, Windows, UNIX, noyaux temps réels.

2) Comparaison Microprocesseur et Microcontrôleur

- Les Microprocesseurs ne sont constitués de l'unité centrale et contiennent un peu de mémoire (cache vois plus loin). Ils disposent d'un bus d'adresse, de données et d'un bus de contrôle des signaux. Ils n'ont pas de mémoire de type ROM ou Flash intégrée et ne disposent pas d'entrées sorties vers l'extérieur. Ils sont très puissants et généralistes.

Le choix d'un microprocesseur est plus lié à sa puissance de calcul et ses grandes capacités mémoire.

- Les microcontrôleurs (monochip) sont très variés et composent des familles de composants suivant la configuration choisie. L'architecture générale est toujours la même que pour un système à microprocesseur, mais les éléments qui le composent sont sur **un seul circuit intégré**.
 - L'unité centrale
 - Les mémoires RAM, ROM, EPROM, EEPROM, Flash
 - Les entrées sorties tout ou rien, analogiques, timers, bus de communication etc..

Le choix d'un microcontrôleur est très dépendant de l'application qu'il doit gérer.

3) Les types de processeurs

a) Les types de processeur

Il existe sur le marché plusieurs types de processeurs.

Les CISC : Complex Instruction Set Computer

Peu de registres internes, jeu d'instructions très complet avec la mémoire : opérations arithmétiques complexes, modes d'adressage complets.

Intel 386, 486, Pentium.

Motorola 68k, Coldfire.

Pour faire des calculs il y a donc beaucoup d'échanges de données avec la mémoire. Cela peut avoir pour effet de ralentir le système.

Les RISC : Reduced Instruction Set Computer

Beaucoup de registres internes, Jeu d'instructions très réduit avec la mémoire : les seules opérations sont des échanges de données en lecture ou écriture en utilisant au mieux la technique des pointeurs.

ARM, Intel et HP P6

Motorola IBM et Apple : Power PC.

Pour faire des calculs il faut donc beaucoup de registres internes au processeur. On trouve trois grandes familles de registres : les registres de calculs en entiers, de calcul en flottant et les pointeurs pour les échanges de données avec la mémoire. Tous les calculs se font donc à la vitesse du processeur.

Les DSP : Digital Signal Processor

Beaucoup de registres internes, plusieurs bus et ALU internes, jeu d'instructions orienté traitement du signal.

Texas Instrument : TMS320 C xx

Analog Devices ADVxx

ST Micoelectronics

Motorola

Les DSP disposent de plusieurs unités de calculs spécialisés et en particulier d'une unité « MAC », Multiplieur Accumulateur pour le calcul de filtres numériques. Ils sont aussi très spécialisés et le choix du type de DSP se fait en fonction de l'application.

Les coeurs IP : Propriété Intellectuelle

Ce sont des coeurs de processeurs décrits dans un langage de description matérielle HDL (Hardware Description Language) ou en « C++ . »

ARM, Coware

Ce type de processeur est réservé à la production en grand volume car les coûts d'études et de production sont très élevés.

b) Analyse comparative et domaine d'utilisation.

Jusque vers les années 1990/95 le coût d'une machine **RISC** était beaucoup plus élevé qu'une machine **CISC**. Avec les progrès de l'intégration les prix des deux types de machines sont très voisins. Ils sont utilisés pour :

- Les ordinateurs de bureau.
- Les systèmes de contrôle industriel (automate, machine VME, routeurs etc.)

Les **DSP** sont très puissants en calcul pour le traitement des filtres numérique, du son, des images et de la commande de moteur.

Les processeurs **IP** sur mesure sont adaptés à la production en grand volume sur un type d'application précis compte tenu du coût de leur développement.

Pour les systèmes **portables** se sont les **microcontrôleurs**, les **DSP** ou les processeurs **IP** à faible consommation qui sont utilisés.

c) **Le micro-ordinateur.**

Un micro-ordinateur est construit autour d'une MONOCARTE qui comprend tous les éléments indiqués ci-dessus.

Les mémoires et les dispositifs d'entrées sorties sont câblées sur la même carte de circuit imprimé que le microprocesseur.

Elle dispose de connecteurs d'extension qui permettent de brancher tous les périphériques du marché (écran, clavier, disques, imprimantes, souris, liaison RS232, Réseaux, Modem Fax).

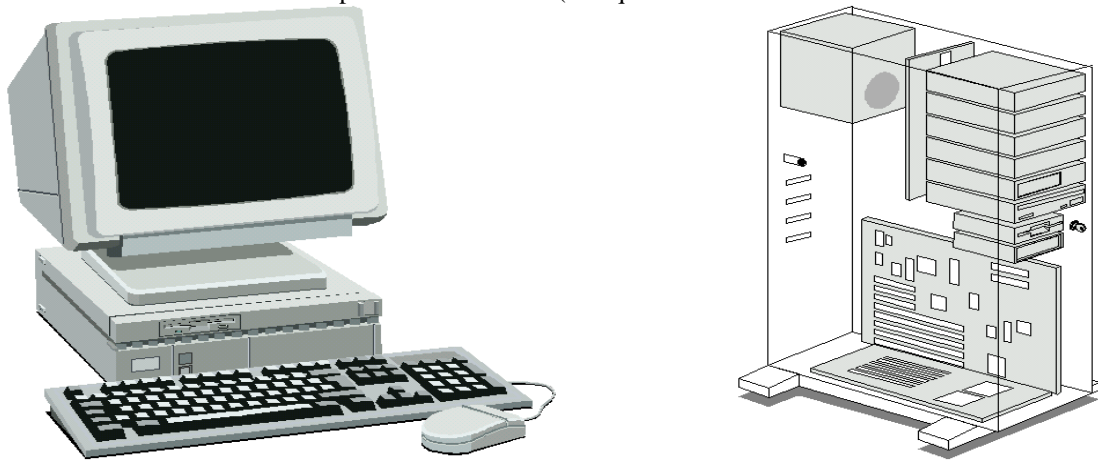
On trouve plusieurs grandes familles sur le marché actuel :

- Compatible PC : Micro Intel 386, 486, Pentium.
- Apple Macintosh : Micro Motorola Power PC.
- Sun, IBM et Motorola : Power PC, Sparc etc.

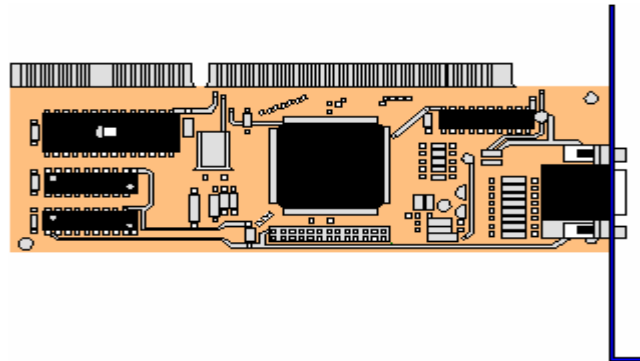
Les systèmes d'exploitation sont :

- MSDOS, Windows pour les compatibles IBM PC.
- Apple System xx pour les Macintosh
- UNIX pour les stations de travail industrielles.

Ces micro-ordinateurs sont des machines de bureau utilisés pour le développement et non pour le contrôle de processus industriel (manque de fiabilité dans un environnement sévère).



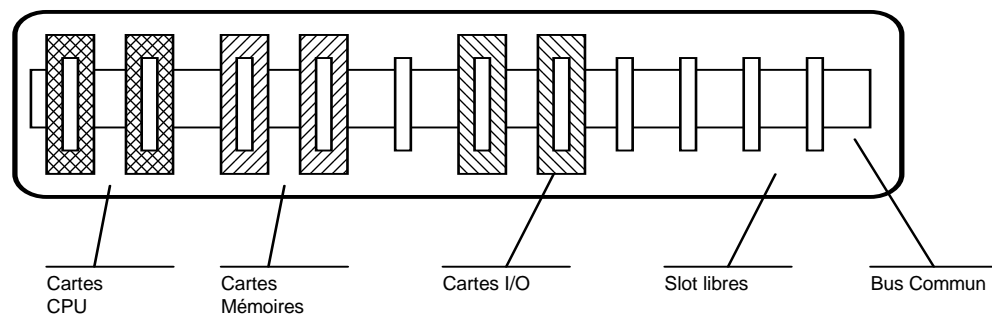
Carte d'extension



On trouve souvent sur ce type de carte des processeurs spécifiques (DSP, Graphique ou autres) dont le rôle est de décharger le processeur central de la carte mère de tâches de calcul.

d) **Les automates industriels.**

Un automate industriel comprend les mêmes éléments qu'un micro-ordinateur mais il est **modulaire** et construit autour d'un **CHASSIS** et d'un **FOND de PANIER** suivant le schéma ci-dessous.



Le châssis mécanique, les alimentations et les cartes électroniques sont prévus pour résister à des ambiances industrielles sévères.

Suivant la complexité de l'application, il peut y avoir une ou plusieurs MONOCARTE CPU, une ou plusieurs cartes mémoires, une ou plusieurs carte I/O (Input Output).

Le bus de fond de panier est commun à toutes les cartes : il est géré par un Maître situé en général sur une carte CPU.

Chaque carte doit respecter la norme du bus sur lequel elle est installée. Il est possible de mélanger des cartes de différents constructeurs sur le même bus.

On trouve sur le marché :

- des micro-ordinateur de type IBM PC (ils sont durcis).
- des systèmes au bus VME (Versatil Module Europ).
- des systèmes au bus Compac PCI (Bus PCI sur un fond de panier) ou VXI.
- des Automates Constructeurs : April, Télémécanique, Siemens etc...

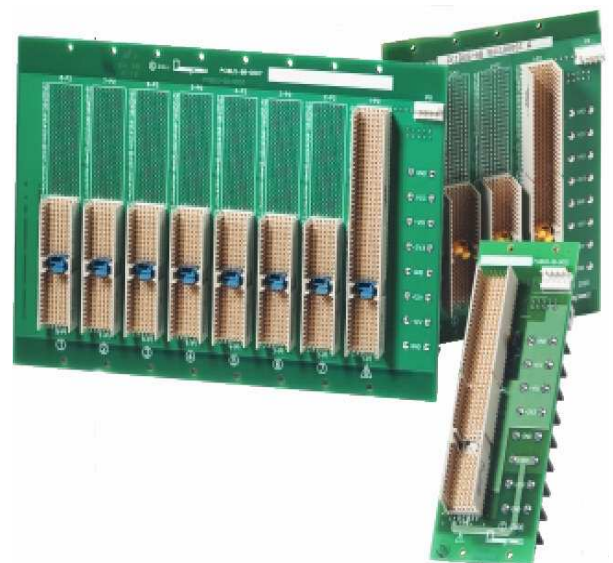
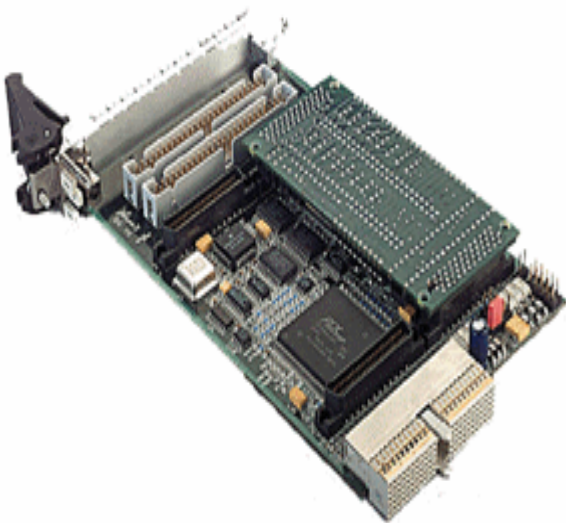
Les systèmes d'exploitation sont essentiellement UNIX et des noyaux temps réel comme PSos, VRTX, VxWorks, Labview pour les machines ouvertes.

Les automates constructeurs disposent de logiciels propriétaires.

Les applications principales de ce type de machines sont le contrôle de processus industriels spécifiques (automatismes, robotique, régulation, contrôle de réseaux etc...)



CPU Système PCI : Carte CPU et Châssis Compac PCI



Carte fille support de modules d'entrées sorties et fond de panier
(Documents GESPAC)

e) **Les systèmes embarqués.**

On trouve de plus en plus d'électronique dans les produits grand public : machine à laver, téléphone, télévision, radio, compact disque, automobile etc...

Tous ces systèmes ont besoin de composants programmables très bon marché réalisant une fonction bien précise.

C'est le domaine des MONOCHIP.

Le Monochip est un composant électronique qui comprend **sur une même puce de silicium** tous les éléments constituant un système, à savoir **le CPU, la RAM, la ROM et les I/O.**

Les capacités mémoires disponibles sont très faibles (1k à 32k).

Le CPU peut être de type CISC, RISC, DSP ou IP.

Ils sont produits en très grand nombre pour réduire les coûts de production.

4) Architecture interne d'un microprocesseur CISC

Tout microprocesseur est construit autour d'une ou plusieurs unités de calcul ALU (Arithmétique Logic Unit), et d'un ensemble de registres plus ou moins nombreux suivant la puissance désirée. L'ALU ainsi que les registres peuvent utiliser de mots de 8, 16 32 ou 64 bits.

a) Unité arithmétique et logique, Instruction, Registre d'Etat.

Une Unité Arithmétique et Logique (ALU) réalise des opérations de calcul sur des *opérandes organisés en mots de n bits.*

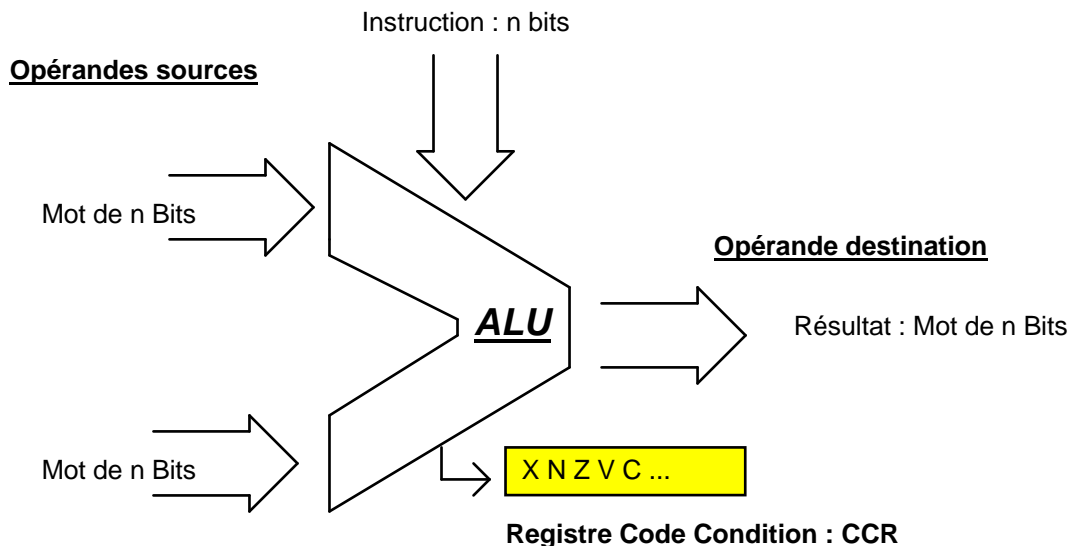
La taille des opérandes "n" dépend de la capacité de traitement du microprocesseur. On trouve des microprocesseurs qui traitent de mots de 8, 16, 32 ou 64 bits.

L'instruction à exécuter est elle aussi codée en binaire. La taille du *mot instruction* dépend de la complexité du jeu d'instruction du microprocesseur : elle souvent un multiple de 8 bits.

Le résultat des instructions est rangé la plupart du temps dans un mot de taille identique aux opérandes sources : On parle d'opérande destination.

Un registre interne au microprocesseur indique la nature du résultat : *Registre Code Condition : CCR*. Il indique si le résultat est nul, positif, s'il y a eu débordement etc...

Les valeurs des bits de ce registre sont mises à jour lors de l'exécution de chaque instruction. Elle sont utilisées lors de test de branchement conditionnel.

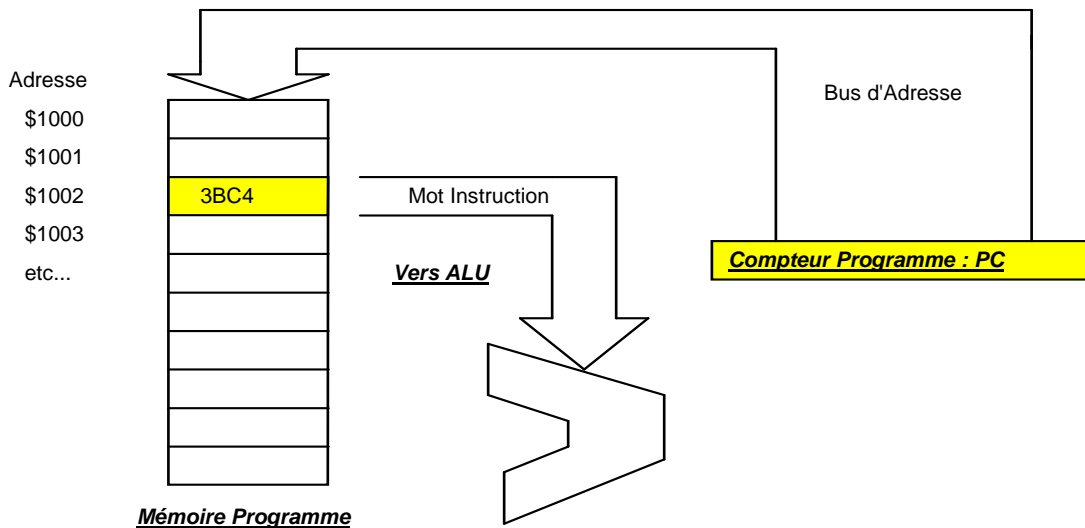


b) La mémoire programme : le Compteur Programme PC.

La suite des opérations élémentaires à effectuer par le microprocesseur est contenue dans une mémoire programme. Celle ci doit être chargée du **code binaire de chaque instruction.**

Chaque mot de n bits, constituant une instruction, est extrait de cette mémoire en utilisant un registre compteur binaire : le **Compteur Programme PC.**

Le nombre de bits de ce compteur est égal au nombre de bits d'adresse nécessaires pour accéder à la mémoire programme.



Pour exécuter l'instruction suivante le compteur programme PC est incrémenté de une unité :
 (1, 2 ou plusieurs adresses mémoires, en fonction de la taille des mots instruction).

Lors d'un branchement à une adresse, un mécanisme analogue à un chargement parallèle de compteur est réalisé.

Exemples :

Famille Motorola		Registres Internes	Bus de données	Bus d'Adresse
Nom				
6809		8 bits	8 bits	16 bits
68040		8/16/32 bits	32 bits	32 bits

c) Les registres de données, la mémoire de données.

Dans chaque microprocesseur il existe des registres internes spécialisés qui contiennent des données de calcul. La taille de ces registres est liée à la taille des données que peut manipuler l'ALU.

Dans une architecture CISC le nombre de ces registres est limité :

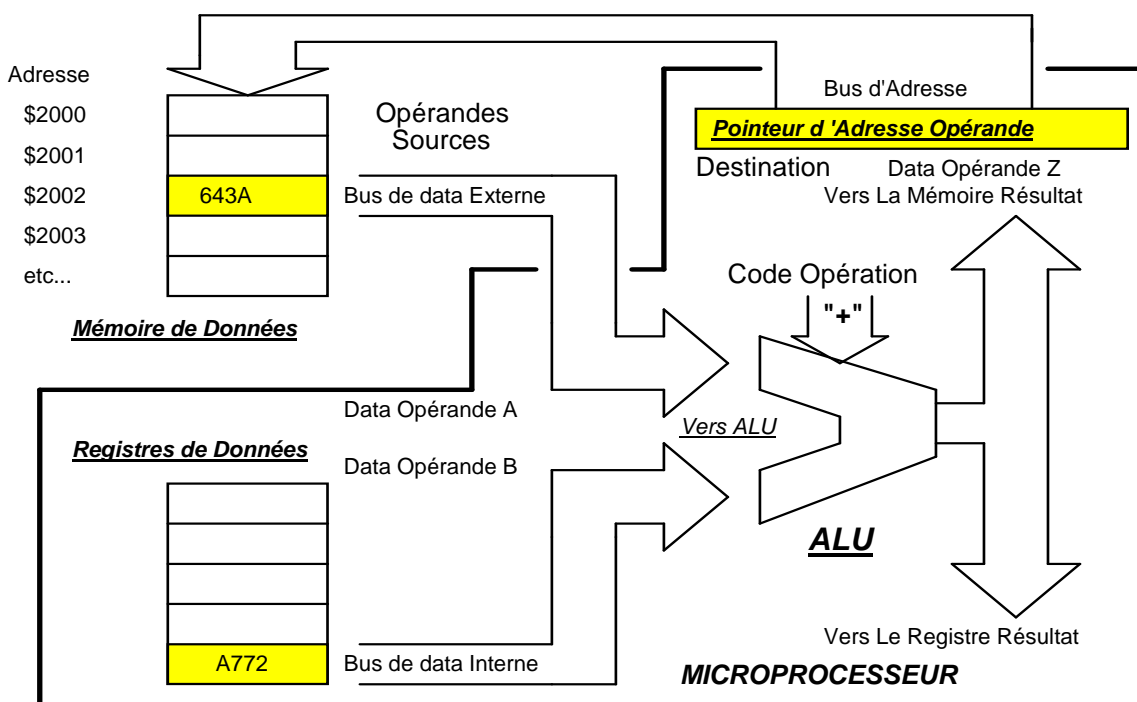
- 8 registres 32 bits pour le 68000. : registre D0 à D7

Toute opération arithmétique ou logique utilise souvent un de ces registres soit en opérande source soit en opérande destination.

Les opérandes sources peuvent aussi provenir d'une mémoire de données externe au microprocesseur.

Le résultat, de chaque opération, opérande destination, est rangé soit dans un registre interne du microprocesseur, soit dans une mémoire de donnée, suivant la nature de chaque instruction.

C'est le **mode d'adressage de l'instruction** qui indique l'**Adresse Effective (AE)** des opérandes, mémoire ou registre.



Possibilités de mode d'adressage:

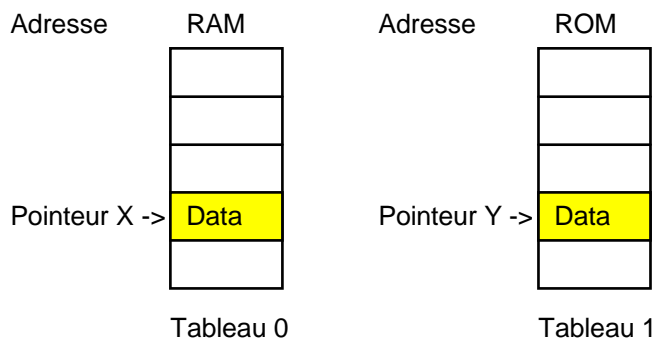
Source X	Source Y	Destination Z
Mémoire	Mémoire	Mémoire
Registre	Mémoire	Mémoire
Registre	Registre	Mémoire
Registre	Registre	Registre
etc...		

Pour un microprocesseur donné toutes les combinaisons d'AE ne sont pas possibles pour toutes les instructions. Il faut se reporter à la documentation technique du constructeur qui précise quels sont les modes d'adressage effectifs de chaque instruction.

A chaque instruction le CCR est mis à jour.

d) Les registres pointeurs.

Les programmes gèrent souvent des **tableaux de données en mémoire**. Pour accéder à un élément du tableau il suffit de connaître l'adresse mémoire du premier élément : c'est le rôle des **pointeurs d'Adresse**.

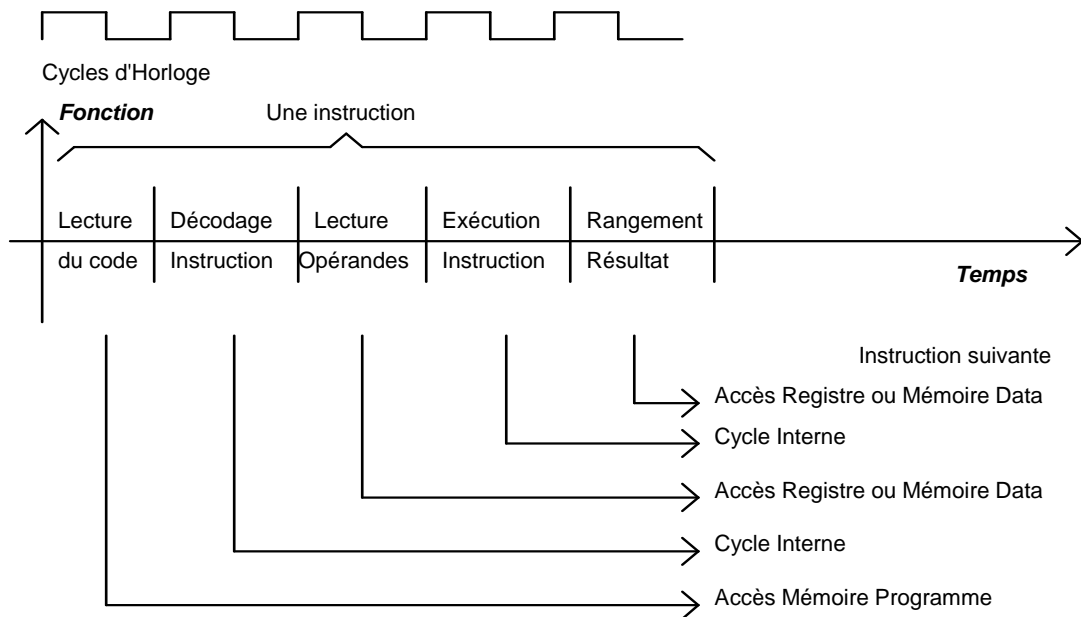


Dans une architecture CISC le nombre de ces registres est limité :

- 2 pointeurs 8 bits pour le 6809 : Registres X et Y
- 8 pointeurs 32 bits pour le 68000. : Registres A0 à A7

5) Séquencement d'une instruction

Le schéma ci-dessous résume le séquencement d'une instruction :

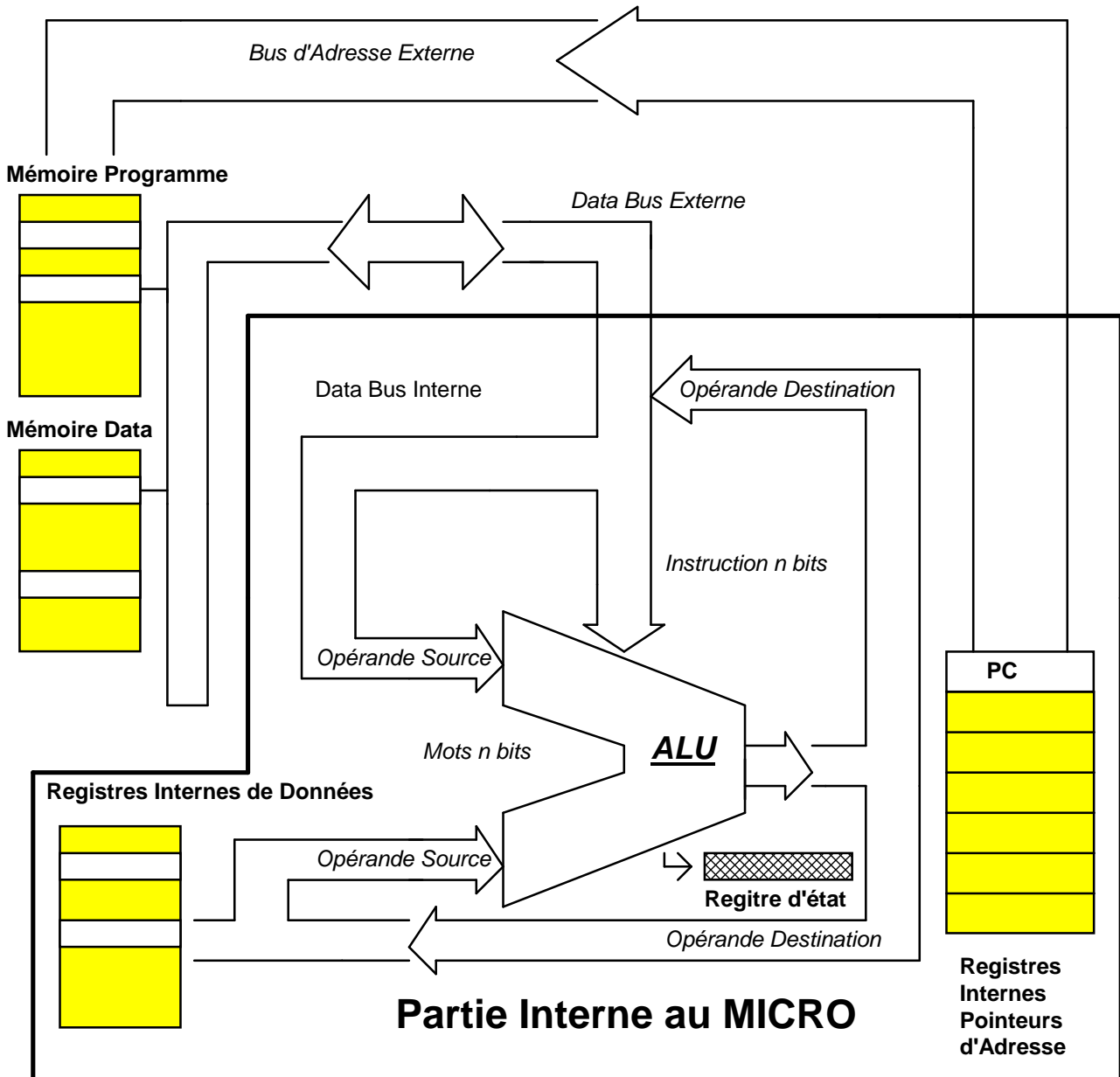


On constate sur ce schéma qu'il faut cinq cycles élémentaires d'horloge pour exécuter une instruction.

6) Schéma simplifié d'un microprocesseur CISC

- Un seul bus d'adresse vers la mémoire externe au processeur.
- Un seul bus données qui véhicule à la fois les codes instructions et les données de la mémoire externe vers l'unité centrale.
- Une seule unité arithmétique et logique.
- De nombreuses et très puissantes instructions.
- Des modes d'adressages très variés.
- Un nombre de registres de calcul internes limité
- Un nombre de pointeurs d'adresses limité.

C'est une machine de type Von Neuman



7) Architecture Harvard et Von Neuman

Dans l'exemple présenté ci dessus le programme et les opérandes sont rangées dans le même plan mémoire. Le processeur doit donc faire la distinction entre un code binaire représentant une instruction et un code binaire représentant une donnée.

Cette distinction est fait dans le code de l'instruction et le microprocesseur doit gérer l'aiguillage des données et du programme.

C'est le principe d'une architecture **Von Neuman**.

Dans une architecture **Harvard** les données et le programme sont rangés dans au moins deux mémoires différentes. Il n'y a donc plus besoin du dispositif d'aiguillage précédent.

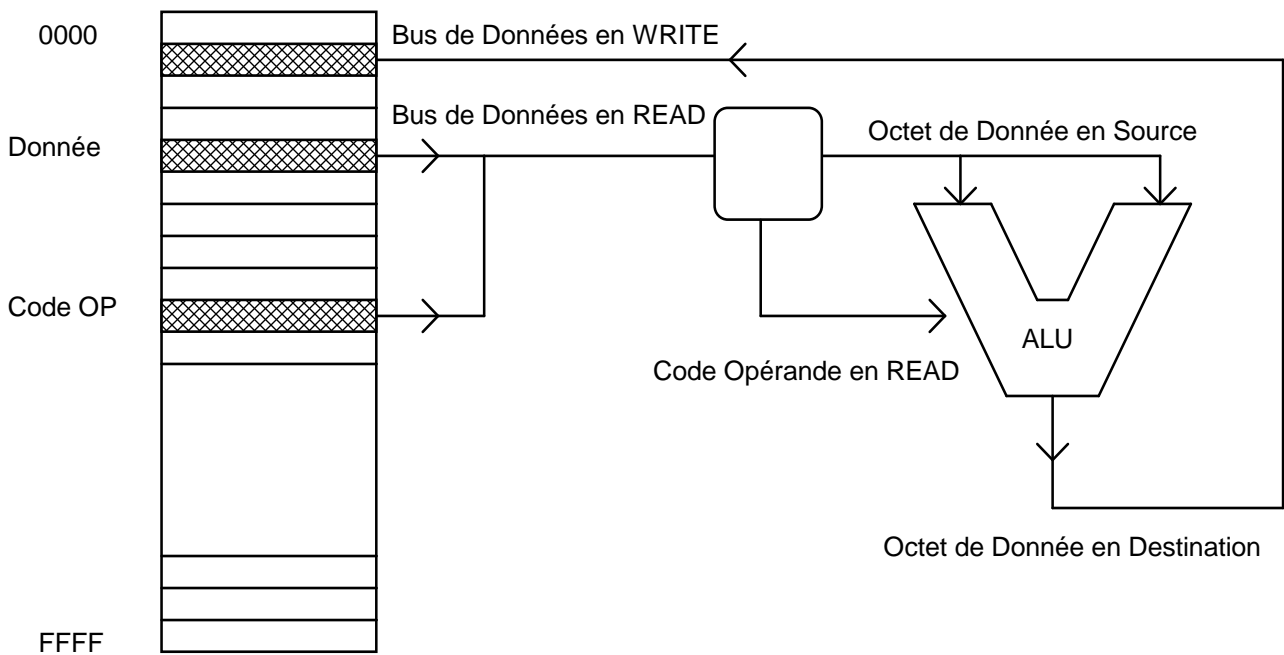
Il cependant synchroniser les accès à ces deux mémoires.

Cela présente l'avantage de pouvoir aller chercher du code et des données en même temps dans les deux mémoires.

Le schéma de principe des deux architectures est résumé ci après.

Architecture de type Von Neuman

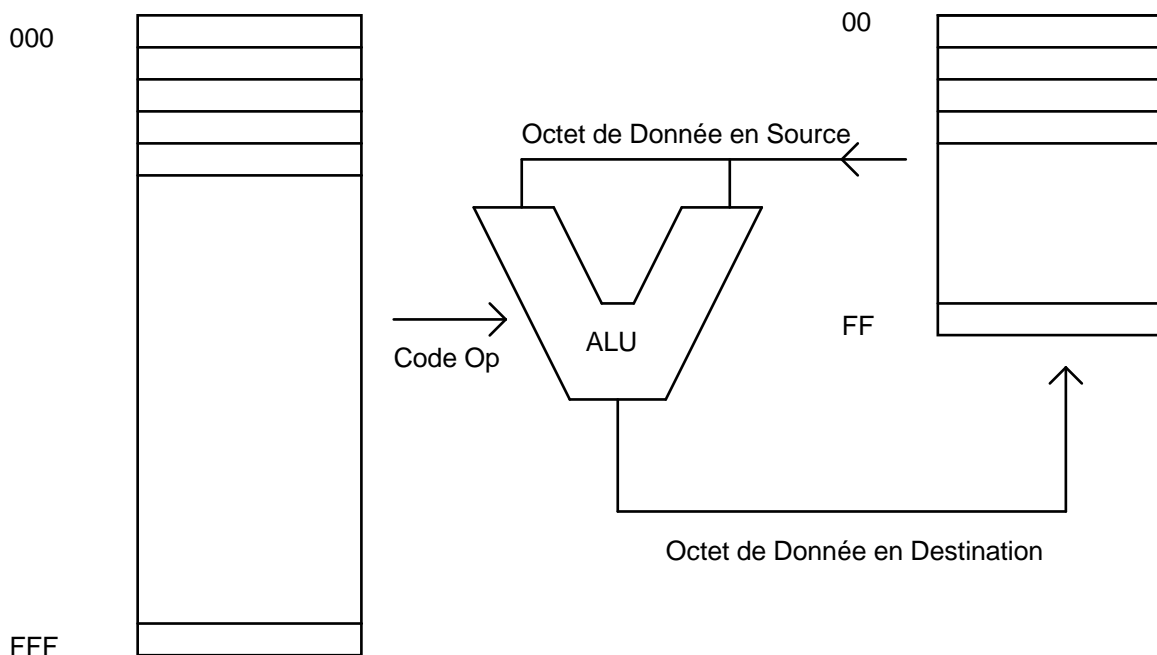
Mémoire Programme / Data 64 K Octets



Architecture de type Harvard

Mémoire Programme 4K Octets

Mémoire Data 256 Octets



8) Amélioration des performances

a) Paramètres physiques.

Les paramètres que l'on peut améliorer sont :

- la fréquence d'horloge.
- la taille et le nombre des registres internes.
- la taille des bus de données.
- la taille des bus d'Adresse etc...

b) Mémoires caches.

L'accès aux mémoires externes est souvent lent par rapport à la fréquence du microprocesseur.

• Cache Instruction

Un programme est souvent constitué de boucles exécutées un grand nombre de fois.

Un cache instruction est une mémoire très rapide, de faible temps d'accès adaptée à la vitesse du microprocesseur. Elle contient les dernières instructions utilisées par le microprocesseur.

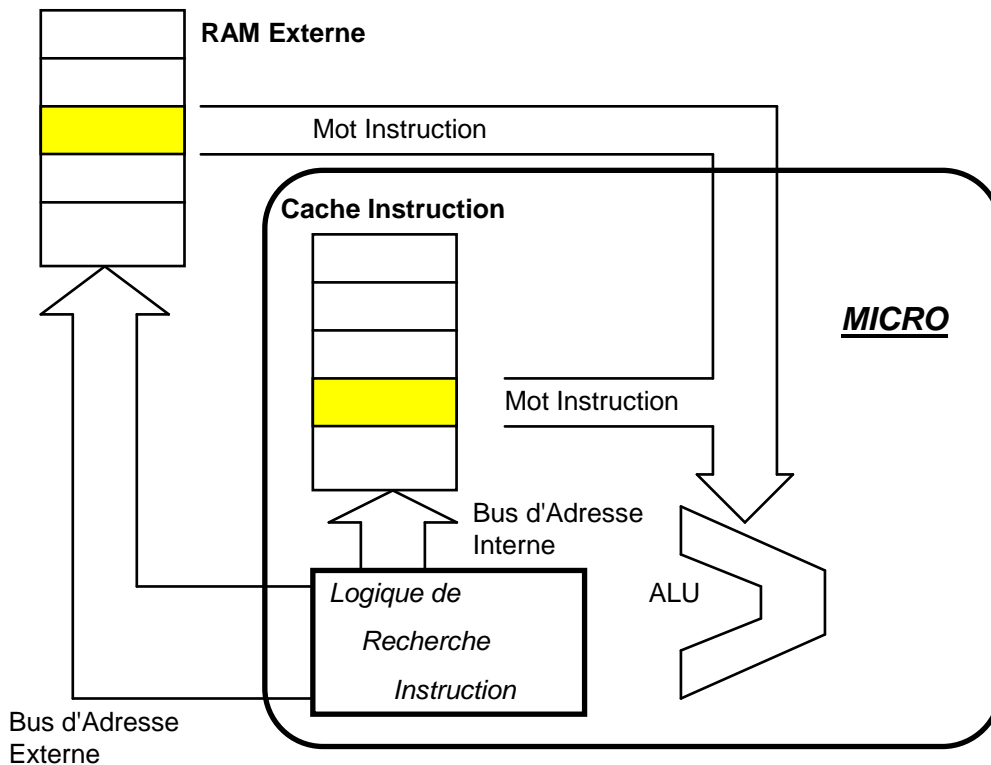
Elle est implantée soit à proximité du microprocesseur, soit sur la puce de silicium elle même.

On parle de cache externe ou interne.

Sa taille est petite : 256 octets dans les microprocesseur de bas de gamme et peut atteindre 32k sur les microprocesseur plus puissants.

Lors du cycle horloge de recherche d'une d'instruction, une logique interne au microprocesseur lance en parallèle une ouverture dans la mémoire cache et dans la mémoire externe. Deux cas peuvent alors se produire :

- la prochaine instruction est dans le cache:
le microprocesseur exécute cette instruction, et abandonne l'accès externe.
- la prochaine instruction n'est pas dans le cache:
le microprocesseur cherche son instruction dans la mémoire externe et abandonne l'accès cache. La mémoire cache est mise à jour par recopie de la mémoire externe.



Cache Données

Le principe d'une mémoire cache de données est identique à celui de la mémoire cache instruction.

Elle agit sur les dernières données utilisées par le processeur.

c) Structure Superscalaire.

Les instructions exécutées par un processeur sont très variées:

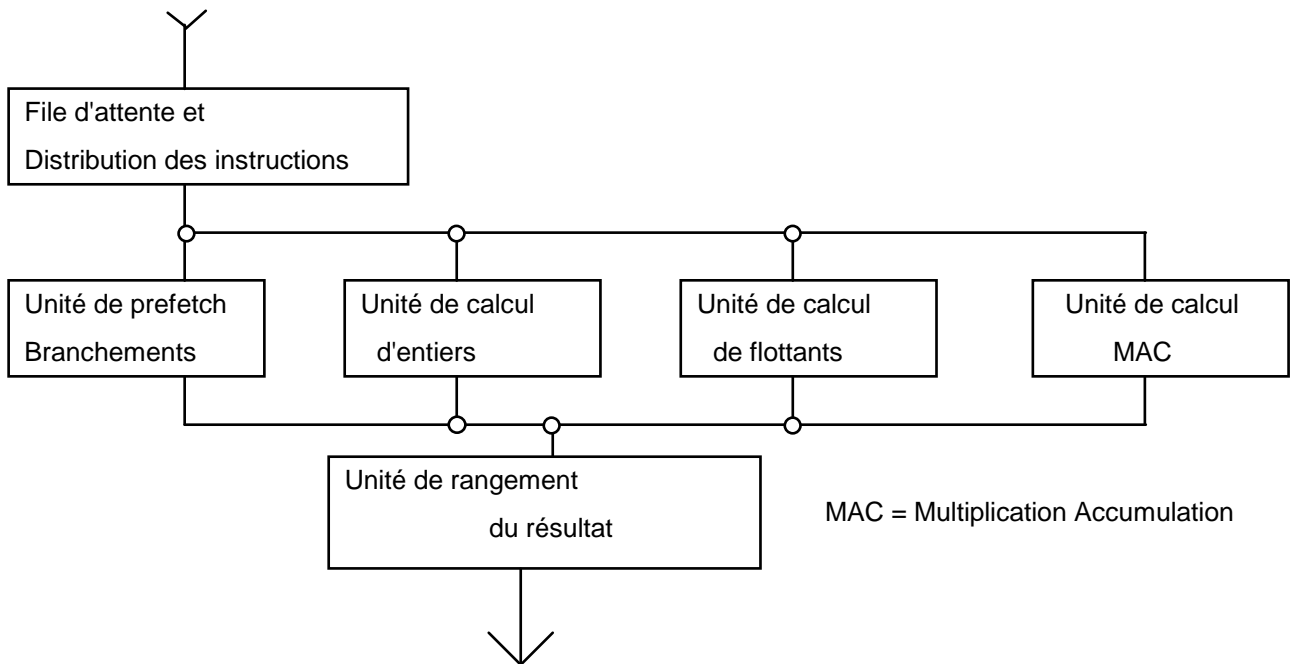
- instructions logiques (ET, OU, NON, Décalages etc...)
- instructions arithmétiques sur des entiers signés ou non.
- instructions arithmétiques en virgules fixes.
- instructions arithmétiques en virgules flottantes.
- calculs d'adresse etc...

Une unité arithmétique et logique qui exécute tous ces types d'instructions est complexe et lente car peu optimisée.

On peut donc multiplier les ALU , une pour le calcul d'adresses, une pour le calcul des entiers, une pour le calcul des multiplication, une pour le calcul des fonctions mathématiques (coprocesseur) etc...

Le mot instruction peut devenir de plus en plus grand : on parle alors de **WLSI** « Very Large Scale Instruction »

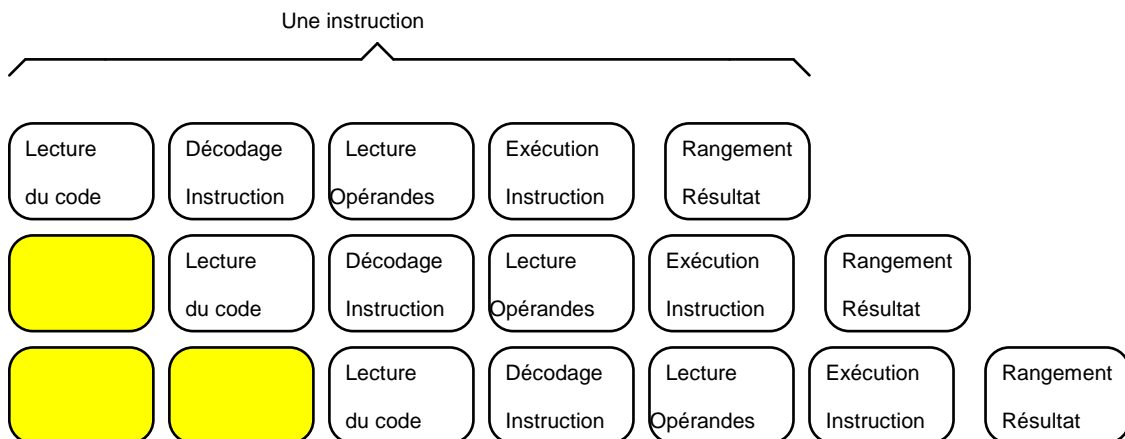
On arrive à la structure suivante:



Toutes ces unités fonctionnent en parallèle.

d) Structure Pipeline

Dans un paragraphe précédent nous avons vu que l'exécution d'une instruction nécessitait par exemple cinq cycles d'horloge. Comme ces cinq cycles sont indépendants on peut imaginer la structure ci-dessous :



e) branchement prédictifs

Lors des branchements conditionnels, il est possible de gagner du temps en **exécutant en parallèle le branchement et le test** : En effet, un programme réalise des milliers de boucles quand le résultat du test est vrai, et ne sort qu'une seule fois si le test est faux.

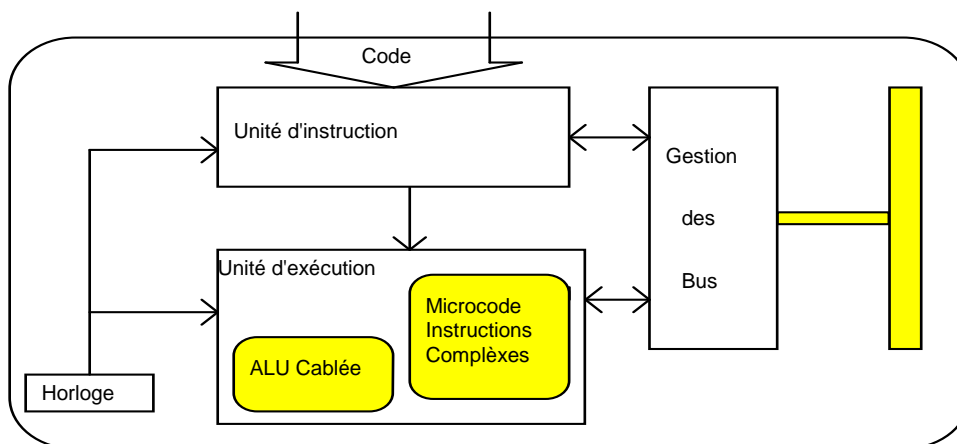
f) structure multi-bus

Le parallélisme des unités de calcul, et l'augmentation de la largeur des instructions obligent le constructeur à multiplier les bus internes. On peut avoir :

- Un bus d'adresses et un bus de données pour la lecture du programme.
- Un bus d'adresses et un bus de données pour la lecture des opérandes.
- Un bus d'adresses et un bus de données pour la l'écriture des opérandes.
-

9) Comparaison CISC / RISC

a) Schéma simplifié du cœur du microprocesseur CISC.



- L'unité d'instruction lit les codes en mémoire centrale, les décode puis les transmet à l'unité d'exécution.
- L'unité d'exécution exécute chaque instruction.
- L'unité de gestion des bus contrôle les accès aux bus d'adresse, de données et de contrôle.

b) Instructions simples et complexes : microcode

- Instruction Simple

Une instruction simple entre registres (ET, OU, Décalage, ADDITION) peut directement être réalisée par une ALU câblée. C'est très rapide.

- Instruction Complexe

Une instruction complexe avec la mémoire (incrémenter le contenu d'une mémoire, addition de mémoire à mémoire, modes d'adressage complexes etc...) ne peut plus être câblée : le nombre de circuits serait trop important. Elle est fractionnée en un nombre restreint de micro instructions qui sont exécutées séquentiellement.

Ce microcode est inscrit dans le silicium. Autour de ce microcode sont installés des circuits logiques qui organisent l'exécution séquentielle des micro instructions.

Il faut donc plusieurs cycles d'horloge pour exécuter une instruction complexe.

- Avantages

Les instructions sont puissantes, le programme d'application est court.

Il faut moins de mémoire centrale.

Les compilateurs sont simples à mettre en concevoir.

La taille des instructions est variable et l'on peut optimiser le code binaire.

-Inconvénients

- Les temps d'exécutions de chaque instruction sont très variables : PB Temps Réel.
- L'analyse du microcode ralentit le processeur.
- Le microcode occupe beaucoup de place sur la puce de silicium.
- Le nombre de registres internes est limité.
- Un faible nombre d'instructions complexes est en fait réellement utilisé.
- Le processeur utilise beaucoup d'instructions de chargement de registres à mémoire.

c) Principe des RISC.

Idées de base : faire des choses simples mais rapidement

- supprimer le microcode : toutes les instructions sont câblées.
(un cycle d'horloge par instruction)
- construire un jeu d'instruction de taille fixe.
(simplification de l'unité d'instruction)
- réduire le nombre des instructions complexes avec la mémoire.
(ne garder que LOAD STORE BRANCH avec la mémoire)
- remplacer une instruction complexe par une suite d'instructions simples câblées.
(la puissance de traitement est dans le compilateur)
- augmenter considérablement le nombre de registres pour limiter les accès RAM.
(réaliser les calculs complexes de registres à registres)
- le format des instruction est sur trois opérandes ou plus.
(ADD DEST,SOURCE_1,SOURCE_2)
- utiliser au maximum :
les branchements prédictifs, la structure superscalaire, les pipelines, les caches Programme et Données.

Prix à payer :

Augmenter la taille de la mémoire centrale.

Ce problème a longtemps été LE FREIN au développement des RISC. Ce n'est plus le cas depuis les années 1995.

Réaliser des compilateurs très efficaces.

La gestion du pipeline, de la structure superscalaire, la multiplication des bus internes, la structure VLSI rend la mise en place du code binaire très délicate.

Réaliser des « debuggers » très efficaces.

Les outils de mise au point du programme doivent permettre de démasquer les erreurs lors du fonctionnement en parallèle des unités et de mettre au point les problèmes liés au temps réel d'une application.

c) Exemples de RISC : Le power PC

En 1991 Apple IBM et Motorola on décidé de construire une famille de processeur RISC : les Power PC dont les caractéristiques sont les suivantes :

- Architecture RISC.
- Bus 32/64/128 bits.
- Superscalaire.
- Pipeline.
- Caches Mémoires.
- Horloge de 400MHz à 2 GHz.

B) Exemples d'architectures

Les exemples de processeurs qui suivent sont choisis de manière à montrer une large gamme de processeurs. Ils n'ont pas la prétention de couvrir toutes les familles de tous les constructeurs et leurs différents composants.

Le choix du constructeur et du composant doivent être dictés par la nature de l'application et les solutions peuvent être autres que celles présentés ci-dessous.

Les documents présentés sont issus des informations données par les constructeurs que l'on peut facilement retrouver sur les sites Internet.

Nous avons volontairement limité la présentation à quelques composants et à cinq constructeurs dont voici les adresses Internet :

ST Microelectronics	http://www.st.com
Microchip	http://www.microchip.com
Motorola	http://www.motorola.com
Texas Instruments	http://www.ti.com

1) Les familles de microcontrôleurs

a) Exemples de CISC

ST Microelectronics

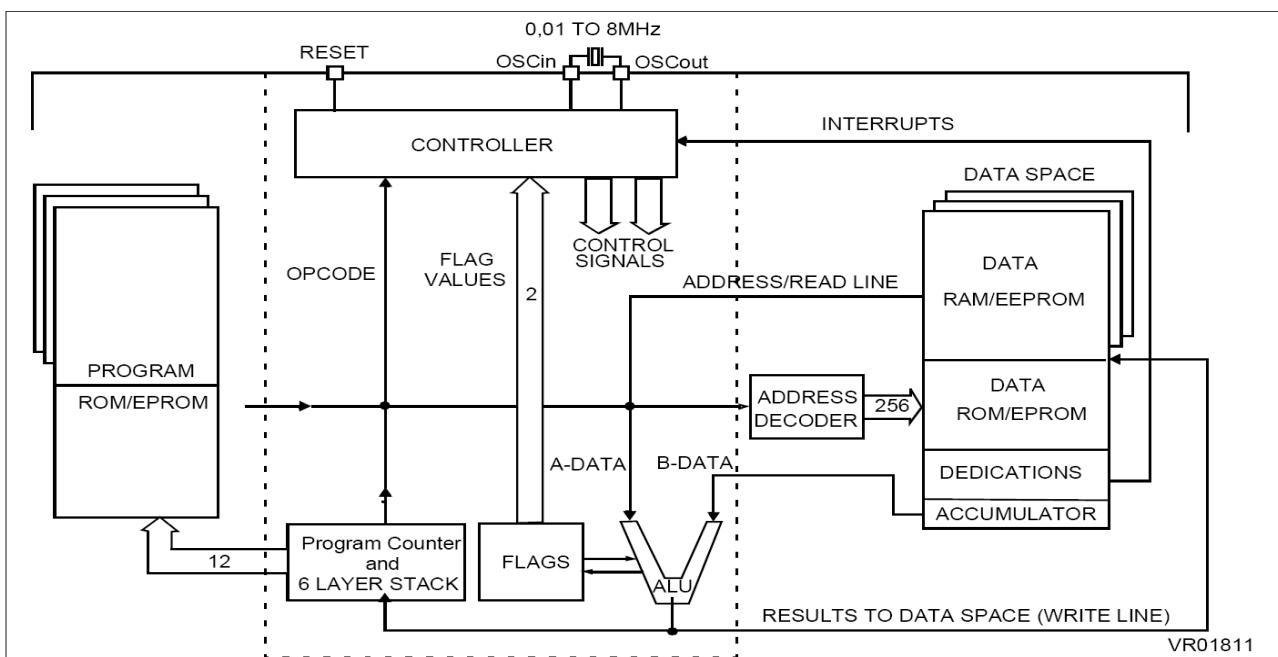
Les famille ST62xx et ST7xx

Ces familles sont construites autour d'un cœur commun et les différents composants se distinguent par leurs capacité mémoire et les dispositifs d'entrées sorties (il en est de même pour toutes les familles, y compris les processeurs RISC et les DSP)

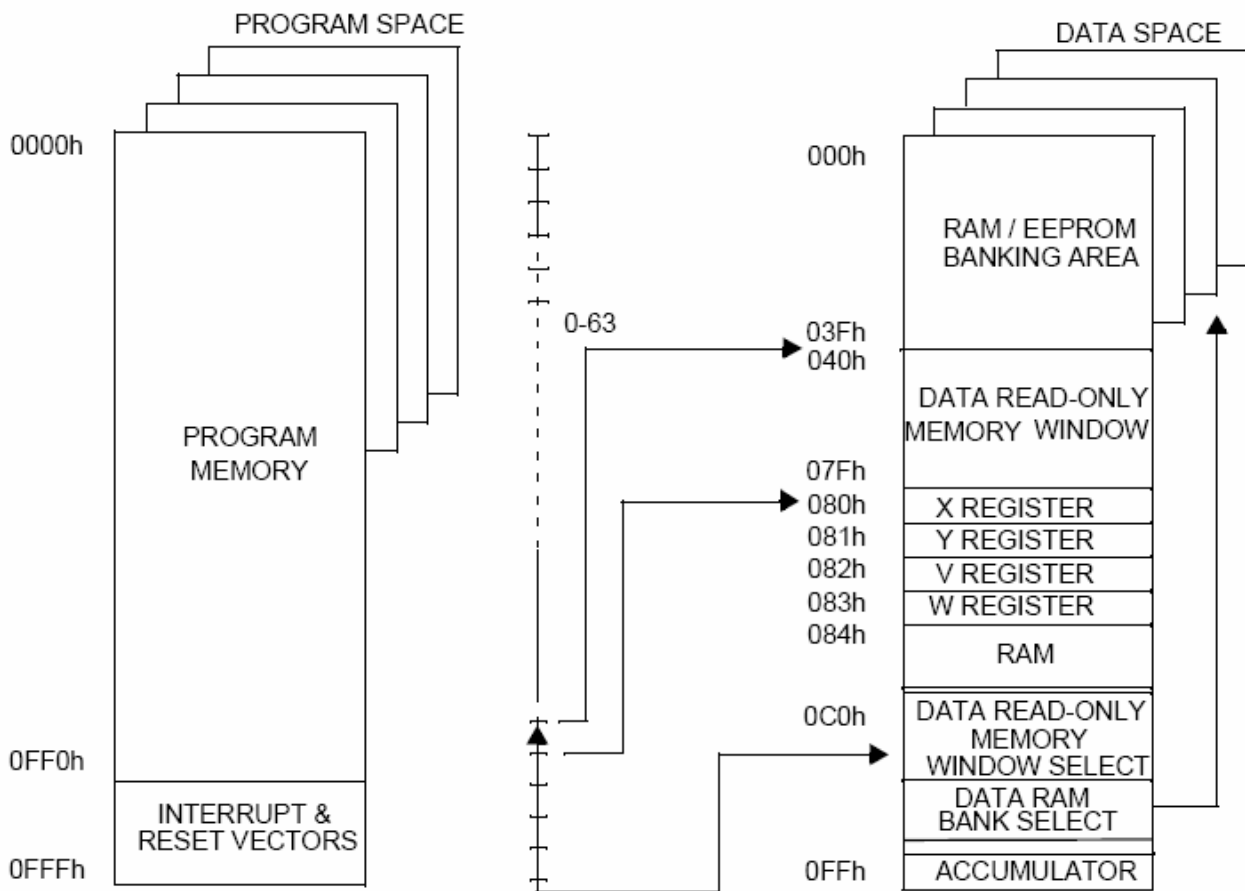
Les caractéristiques principales du cœur ST62xx sont les suivantes :

- Architecture Harvard Une mémoire programme organisée par page de 4k octets. Mémoire de données de 256 octets. Un compteur programme de 12 bits. Un pointeur de pile 12bits et une pile de 6 mots de 12 bits
- ALU 8 bits Traitement sur bits et mots de 8bits non signés. 32 Instructions et 9 modes d'adressage
- Registres Horloge 8Mhz : une instruction comprend plusieurs cycles. Un Accumulateur A 8 bits Deux registres généraux 8bits V et W. Deux pointeurs d'adresses 8bits.
- Entrées sorties Ports de 8bits, DAC, Timers, LCD etc...

Figure 7. ST6 Core Block Diagram



Plan Mémoire du ST62xx



Les caractéristiques principales du cœur ST7xx sont les suivantes :

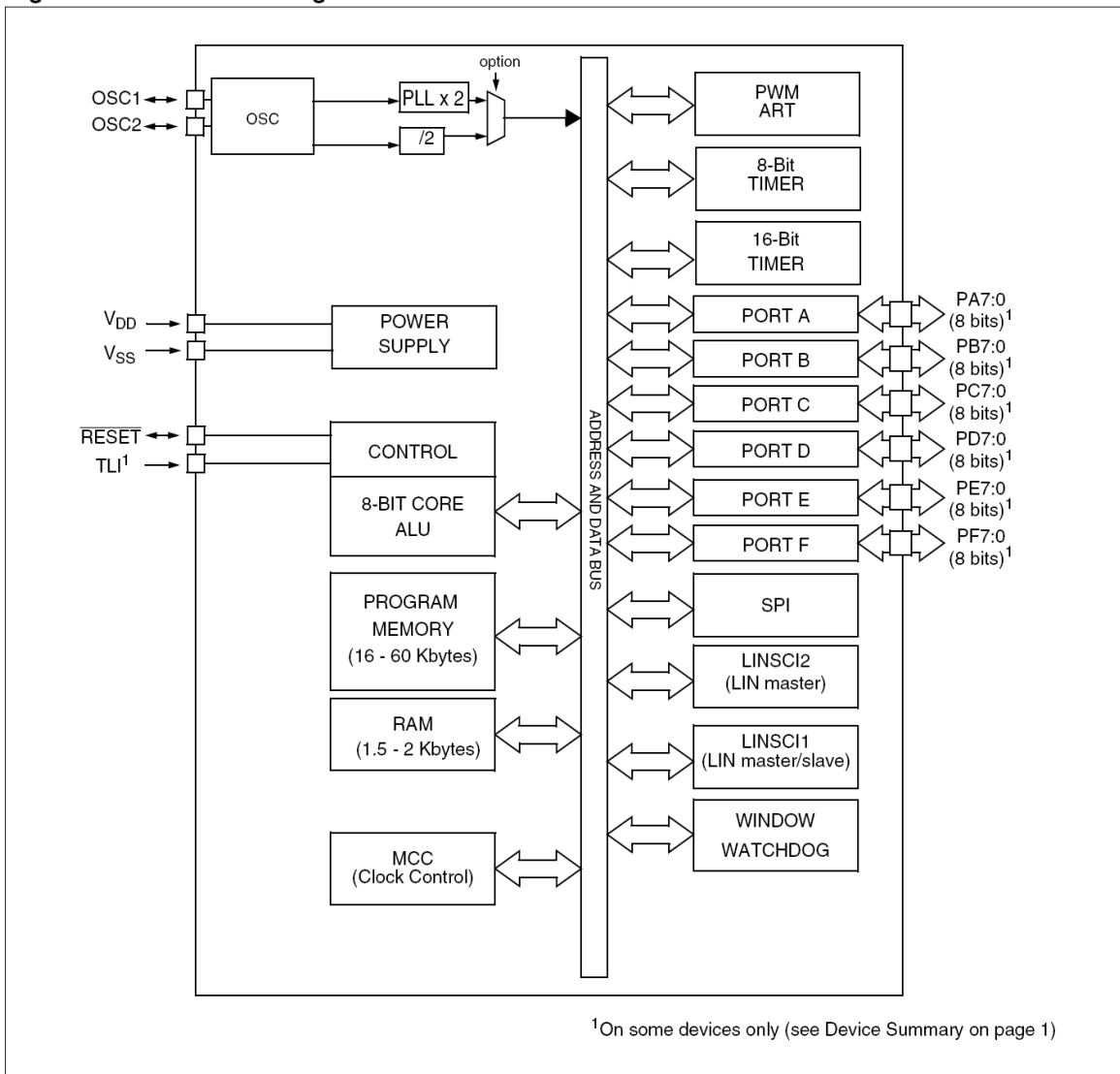
- Machine Von Neuman Un seul plan mémoire pour la ROM, la RAM, et les entrées sorties. Un compteur programme et un pointeur de pile de 16 bits. Une taille mémoire de 64k adressable.
- ALU 8 bits Traitement sur bits et mots de 8 bits signés. 63 instructions et 17 modes d'adressage. Multiplication signée 8*8 intégrée. Horloge 8 MHz.
- Registres Un Accumulateur A 8 bits . Deux pointeurs d'adresses 8 bits X et Y.
- Entrées sorties Ports de 8bits, DAC, Timers, SPI, SCI, I2C, CAN...

❖ Remarque

- Pour ces deux familles la taille de la mémoire est fixée lors du choix du composant.
- Pour d'autres familles les ports d'entrées sorties peuvent servir pour accéder à des mémoires externes (ROM ou RAM supplémentaires).
- C'est le cas des familles HC11 et HC16 de Motorola et des PIC de Microchip qui peuvent accéder à des mémoires externes.

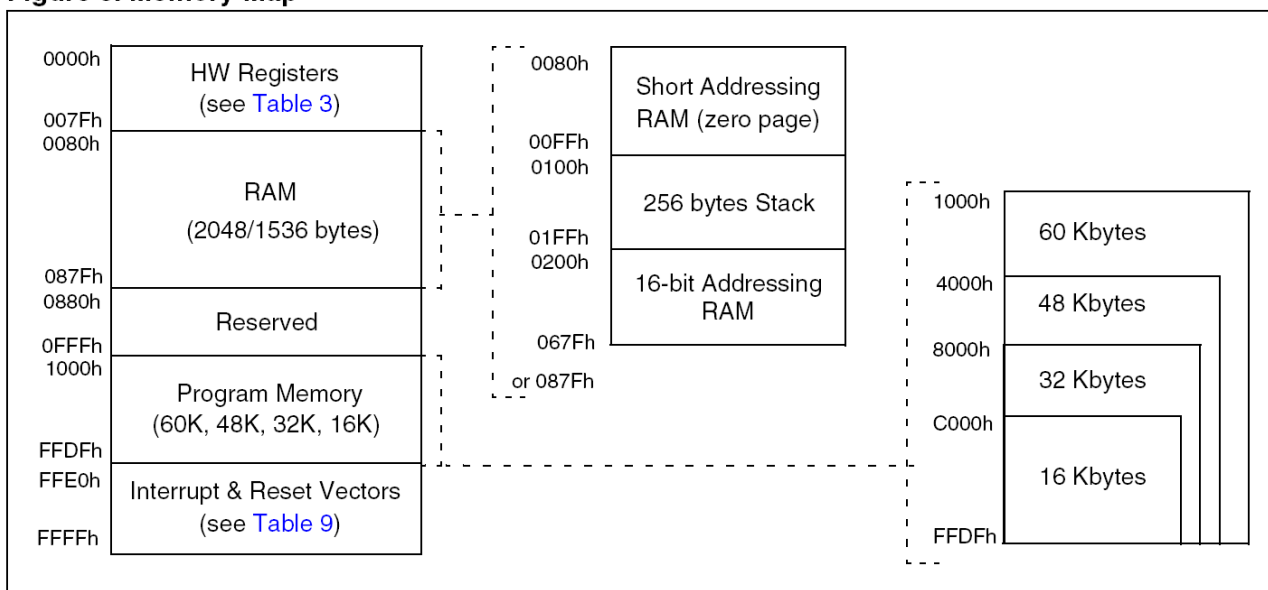
Architecture du ST7

Figure 1. Device Block Diagram



Plan Mémoire du ST7

Figure 5. Memory Map



b) Exemples de microcontrôleur RISC

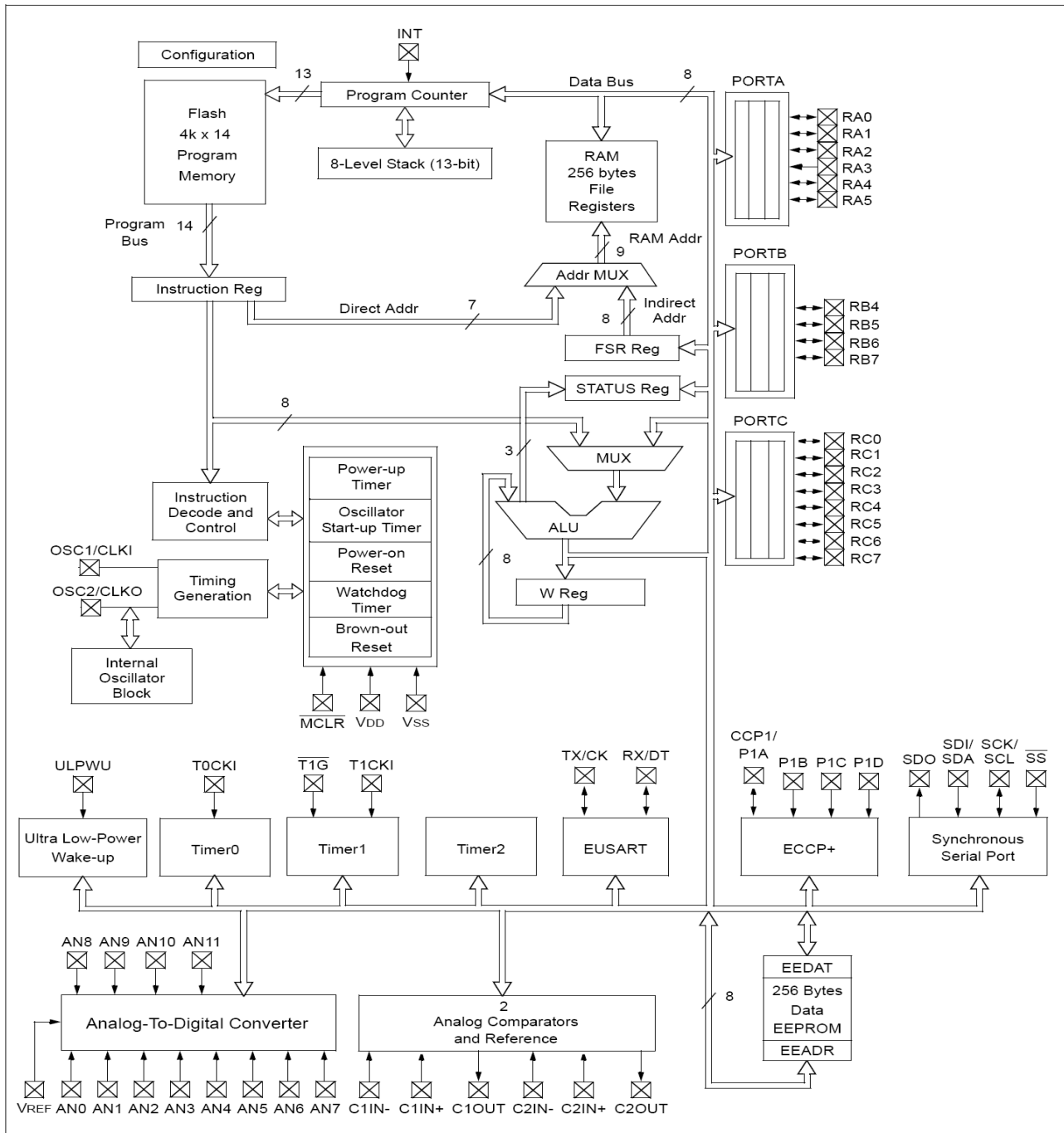
La famille MICROCHIP

Les caractéristiques principales du cœur PIC16f87xx sont les suivantes :

- Architecture Harvard Une mémoire programme FLASH organisée par mots de 14 bits (2k mots).
Mémoire de données de 512 octets.
Un compteur programme de 13 bits.
Un pointeur de pile 13 bits et une pile de 8 mots de 13 bits
- ALU RISC 8 bits Traitement sur bits et mots de 8 bits non signés.
35 Instructions et 3 modes d'adressage.
Une instruction par cycle d'horloge à 20 MHz.
- Registres Un Accumulateur W 8 bits et pointeurs SR.
128 registres généraux.
4 banques mémoires de 128 octets.
- Entrées sorties Ports de 8bits, DAC, Timers, SPI, SCI.

Structure Interne d'un PIC

FIGURE 1-5: PIC16F690 BLOCK DIAGRAM



Les registres d'un PIC

FIGURE 2-4: PIC16F631 SPECIAL FUNCTION REGISTERS

File Address	File Address	File Address	File Address
Indirect addr. ⁽¹⁾ 00h	Indirect addr. ⁽¹⁾ 80h	Indirect addr. ⁽¹⁾ 100h	Indirect addr. ⁽¹⁾ 180h
TMR0 01h	OPTION_REG 81h	TMR0 101h	OPTION_REG 181h
PCL 02h	PCL 82h	PCL 102h	PCL 182h
STATUS 03h	STATUS 83h	STATUS 103h	STATUS 183h
FSR 04h	FSR 84h	FSR 104h	FSR 184h
PORTA 05h	TRISA 85h	PORTA 105h	TRISA 185h
PORTB 06h	TRISB 86h	PORTB 106h	TRISB 186h
PORTC 07h	TRISC 87h	PORTC 107h	TRISC 187h
08h	88h	108h	188h
09h	89h	109h	189h
PCLATH 0Ah	PCLATH 8Ah	PCLATH 10Ah	PCLATH 18Ah
INTCON 0Bh	INTCON 8Bh	INTCON 10Bh	INTCON 18Bh
PIR1 0Ch	PIE1 8Ch	EEDAT 10Ch	EECON1 18Ch
PIR2 0Dh	PIE2 8Dh	EEADR 10Dh	EECON2 ⁽¹⁾ 18Dh
TMR1L 0Eh	PCON 8Eh	10Eh	18Eh
TMR1H 0Fh	OSCCON 8Fh	10Fh	18Fh
T1CON 10h	OSCTUNE 90h	110h	190h
11h	91h	111h	191h
12h	92h	112h	192h
13h	93h	113h	193h
14h	94h	114h	194h
15h	WPUA 95h	WPUB 115h	195h
16h	IOCA 96h	IOCB 116h	196h
17h	WDTCON 97h	117h	197h
18h	98h	VRCON 118h	198h
19h	99h	CM1CON0 119h	199h
1Ah	9Ah	CM2CON0 11Ah	19Ah
1Bh	9Bh	CM2CON1 11Bh	19Bh
1Ch	9Ch	11Ch	19Ch
1Dh	9Dh	11Dh	19Dh
1Eh	9Eh	ANSEL 11Eh	SRCON 19Eh
1Fh	9Fh	11Fh	19Fh
20h	A0h	120h	1A0h
3Fh	EFh	16Fh	1EFh
General Purpose Registers 40h	EFh	16Fh	1EFh
64 Bytes 6Fh	accesses 70h-7Fh F0h	accesses 70h-7Fh 170h	accesses 70h-7Fh 1F0h
7Fh	FFh	17Fh	1FFh
Bank 0	Bank 1	Bank 2	Bank 3

Unimplemented data memory locations, read as '0'.

Note 1: Not a physical register.

Les familles de microprocesseurs

b) Exemples de CISC : la famille 68K de Motorola

La famille 16/32 bits : 68000, 68010, 68012. Adresses 24 bits / données 16 bits.
La famille 32 bits : 68020, 68030, 68040, 68060. Adresses 32 bits / données 32 bits.

Ces familles ont été mises sur le marché dès le début des années 1980 et ont été très utilisées dans tous les systèmes de contrôle au bus G96 et VME ainsi que dans les ordinateurs MAC de Apple.

- Architecture Harward Une mémoire externe 16 Mega (680xx) dès 1980 (24 lignes d'adresses).
 Une mémoire externe de 4 Giga à partir du 68020 (1984).
 Deux modes de fonctionnement :
 - Système pour « l'Operating System » .
 - User pour le programmes d'application.
 Un compteur programme de 32 bits.
 Deux pointeurs de pile 32 bits.
- ALU CISC 32 bits Traitement sur bits et mots de 8 /16/32 bits signés.
 Calcul sur flottants IEEE avec un coprocesseur externe puis interne
 113 Instructions et 18 modes d'adressage (68020).
 Une instruction peut prendre plusieurs cycles d'horloge à 100 MHz.
- Registres 8 registres de données 32 bits.
 9 pointeurs d'adresse et un compteur programme 32 bits.
 Un registre d'état de 16 bits (8 bits système et 8 bits utilisateur).
- Caches Caches instructions et caches de données à partir du 68030.

CETTE FAMILLE APRES 20 ANS EST EN FIN DE VIE REMPLACEE PAR LES COLDFIRE

c) Les COLDFIRE de Motorola

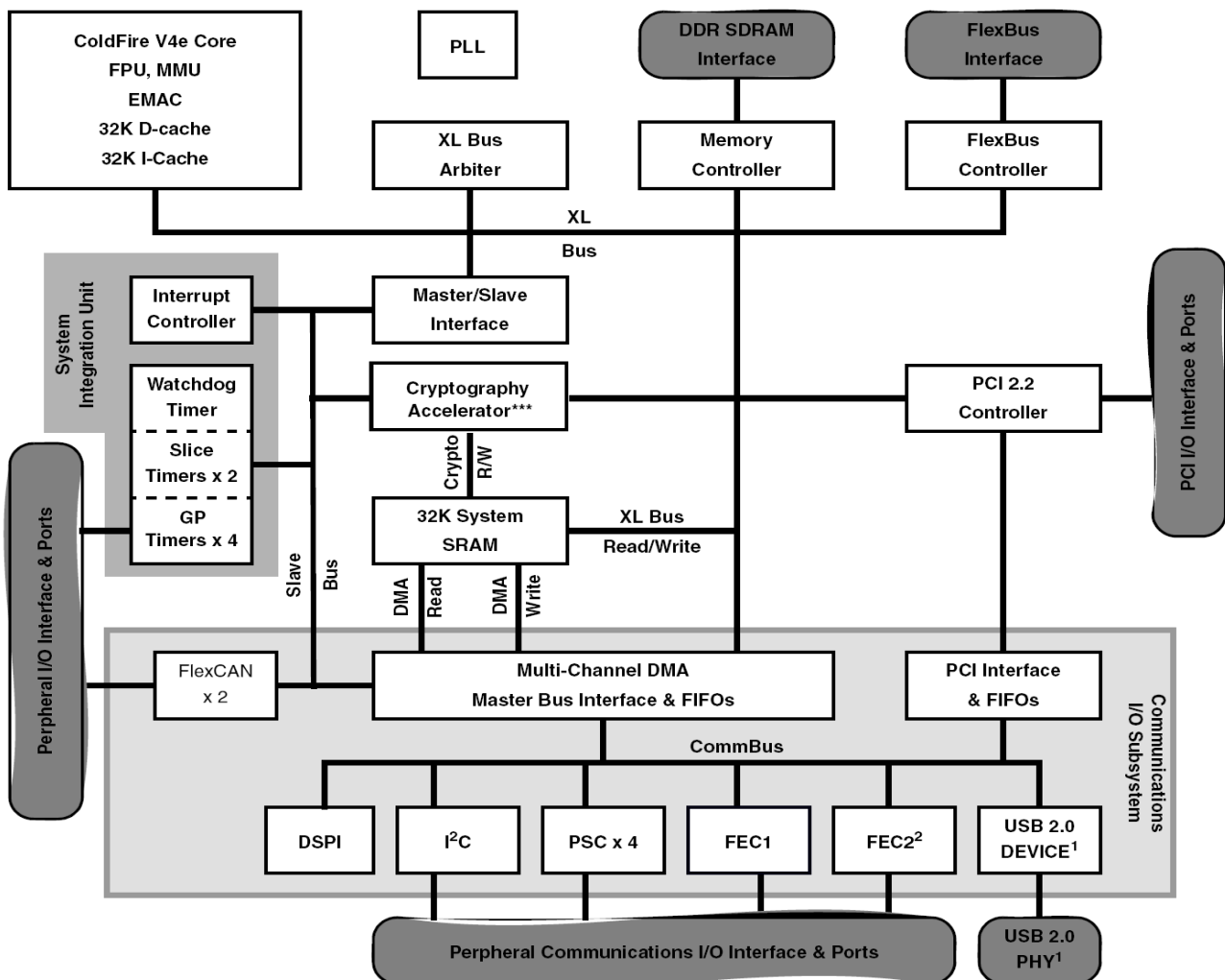


Figure 1. MCF548X Block Diagram

d) Exemples de RISC : La famille Power PC de Motorola

Exemple : la famille MPC 74xx

Ces familles ont été mises sur le marché dès le début des années 1990 et ont pris leur essor vers 1995. Elles remplacent la famille 68k dans les systèmes de contrôle et VME et Compac PCI ainsi que dans les ordinateurs MAC de Apple.

Il supportent tous le noyau et les « OS » du marché : Winxx, UNIX, LINUX, PSOS, VRTX, QNX etc.

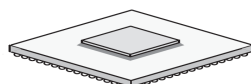
Leur conception est telle que sur les systèmes industriels il n'y a pas besoin de système de refroidissement sophistiqué.

De nombreuses variantes existent, y compris des versions monochip avec RAM, ROM et I/O intégrées à faible consommation pour les systèmes portables.

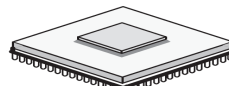
- **Architecture Harvard** Adressage sur 32 bits (128 bits en interne).
Bus de données de 64 bits.
Adressage virtuel avec unité de translation mémoire.
Une mémoire externe virtuelle ou réelle de 4 Giga ou plus.
Deux modes de fonctionnement :
 - Système pour « l'Operating System ».
 - User pour les programmes d'application.
 - Passage en mode superviseur sur exception.Gestion de mémoire intégrée MMU (Memory Management Unit).
 - une unité pour les instructions.
 - une unité pour les données.Une unité d'interface vers les bus externes.
- **Registres** 32 registres de données 32 bits entiers GPR (General Purpose Register).
32 registres de données 32 bits flottants FPR (Floating Point Register).
Registres spéciaux SPR (Special Purpose Register)
Registres liés au matériel HDI (Hardware Implantation Register).
- **ALU RISC 32 bits** Deux unités de calculs sur entiers dont une avec multiplication.
Une unité de calculs flottants.
Une unité de transfert des données en mémoire.
Une unité de calcul des branchements.
Une unité de calcul des vecteurs d'interruption « AltiVec ».
Une unité de distribution des instructions gérée en pile 8 niveaux.
Une unité de contrôle système général.
Les instructions ont une longueur fixe pour le parallélisme.
Instructions à 1, 2, 3 ou 4 opérandes.
Calcul sur flottants IEEE 754 simple ou double.
Les instructions complexes n'utilisent que les registres internes.
Deux modes d'adressage vers la mémoire :
 - Indirect plus déplacement avec registre
 - Indexé à registre.Avec la mémoire il n'y a que des instructions de transfert.
- **Caches** Caches instructions et caches de données séparés à 5 états.
 - 32k de cache de niveau 1 (data/instructions)
 - 512k 1M 2M de cache de niveau 2 associatif.
 - Mode burst pour les accès SRAM avec pipeline.
- **Opérandes** Opérandes de 8/16/32/64/128 bits.

Screening

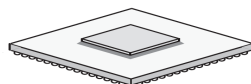
- CBGA Upscreenings Based on Atmel Standards
- Full Military Temperature Range ($T_j = -55^{\circ}\text{C}, +125^{\circ}\text{C}$).
- Industrial Temperature Range ($T_j = -40^{\circ}\text{C}, +110^{\circ}\text{C}$)
- CI-CGA Package Version, HITCE Package Version



G suffix
CBGA 360
Ceramic Ball Grid Array

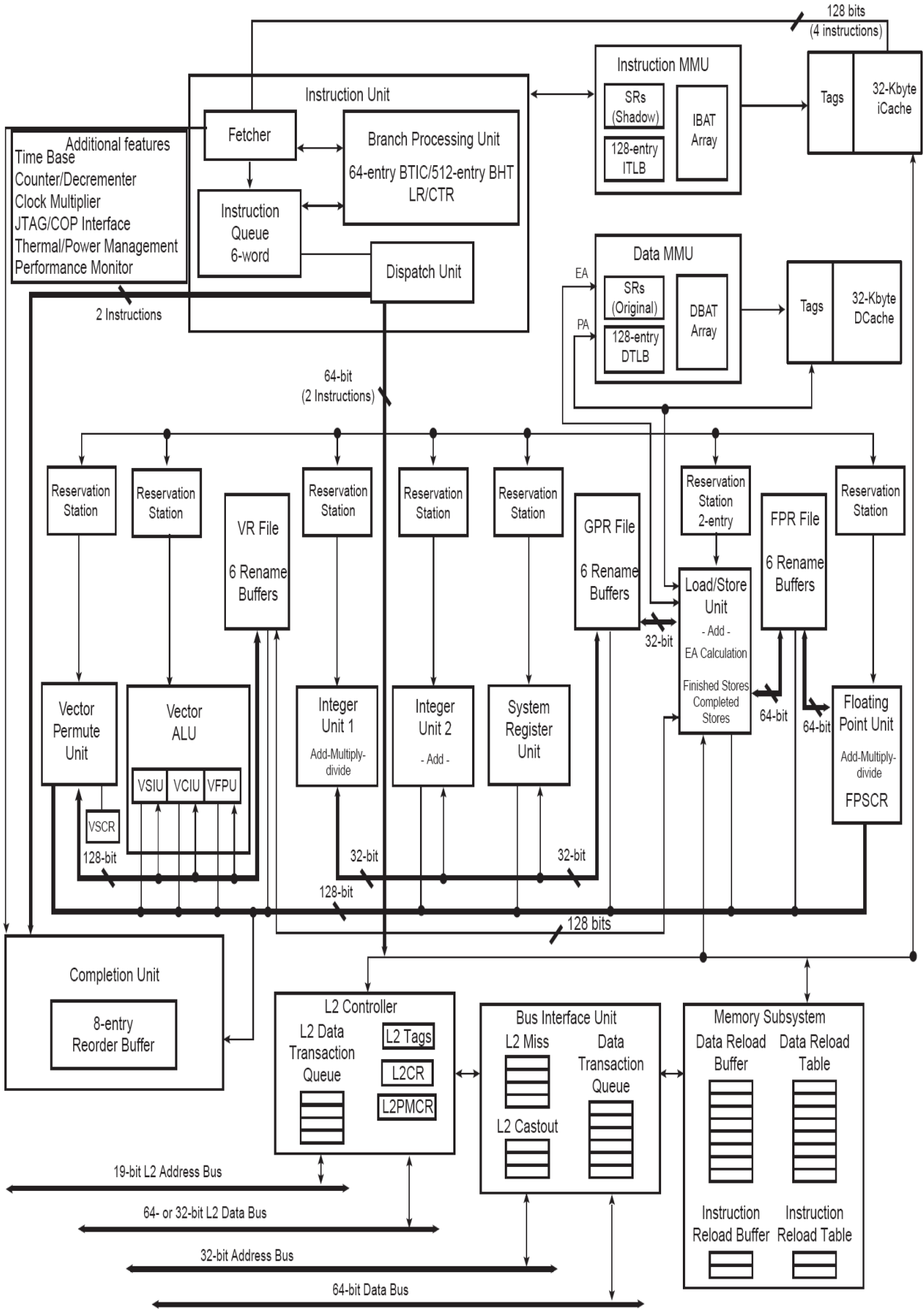


GS suffix
CI-CGA 360
Ceramic Ball Grid Array
with Solder Column Interposer (SCI)

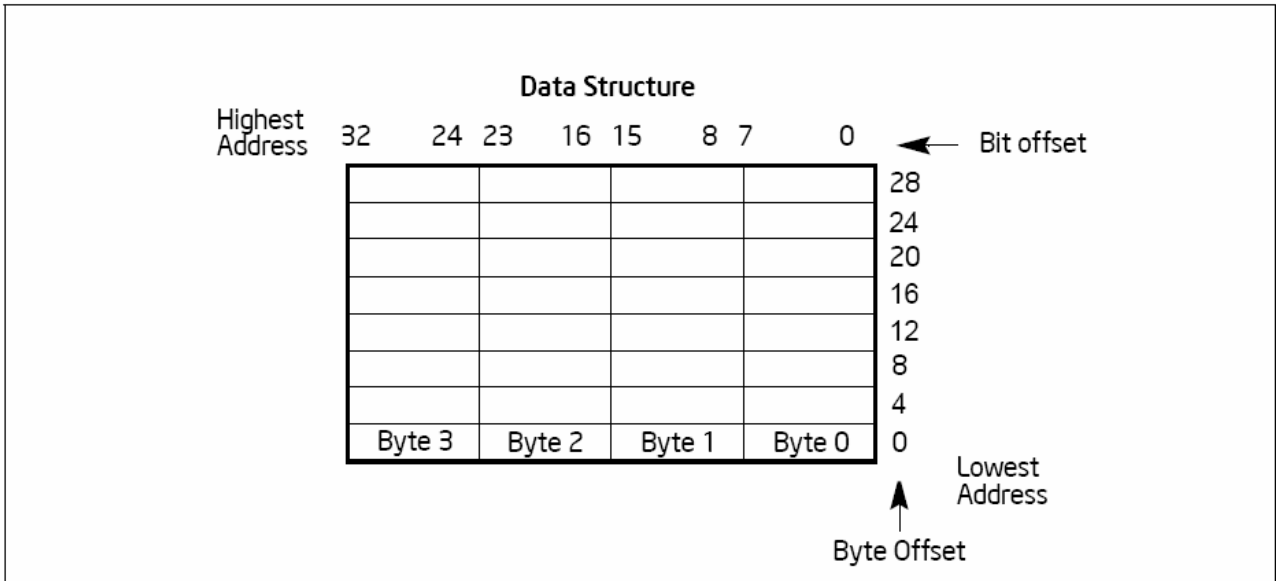


GH suffix
HITCE 360
Ceramic Ball Grid Array

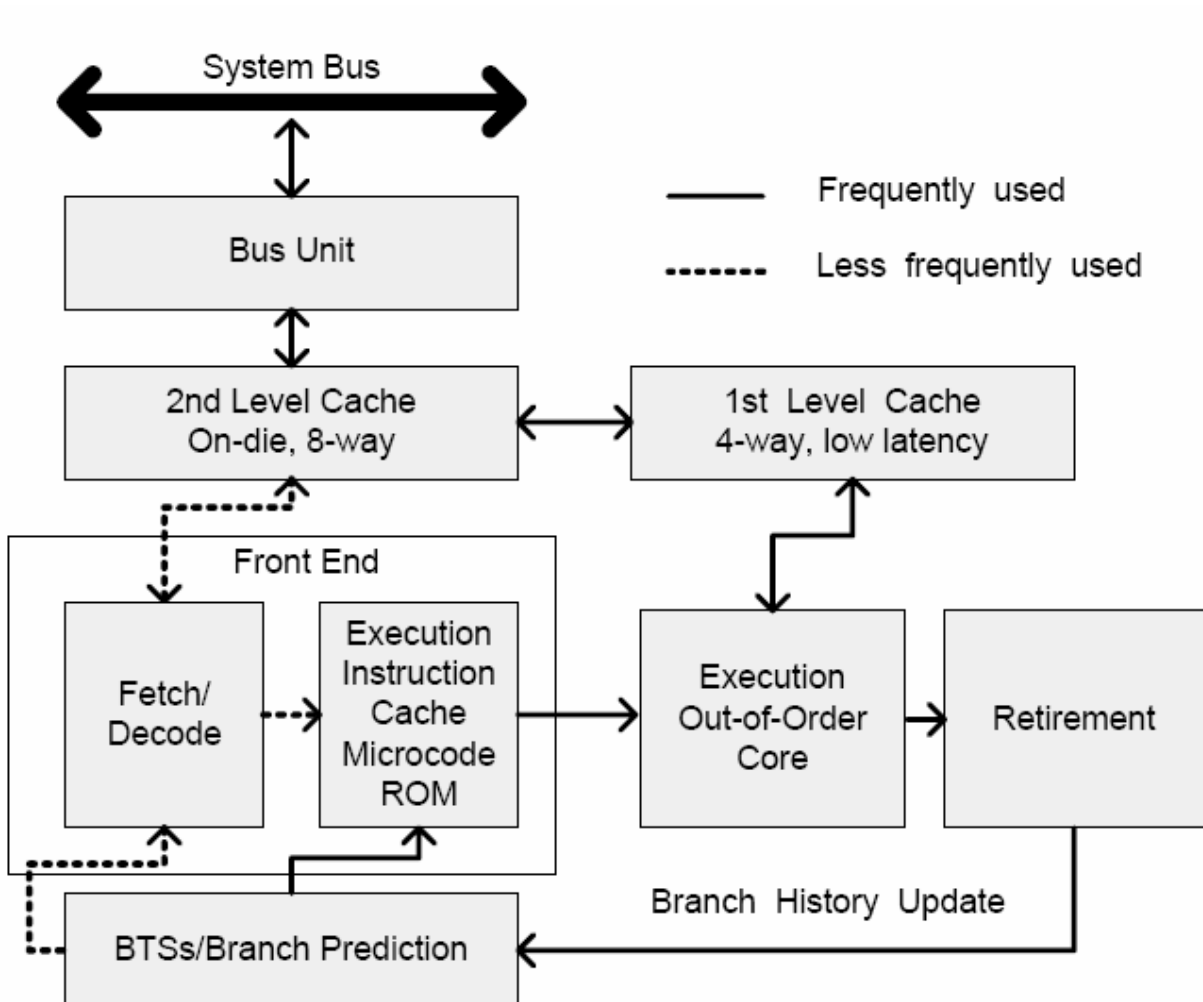
Architecture d'un Power PC



Les Pentium de Intel : Organisation des données



Les Pentium de Intel : Architecture interne



Les familles de DSP

Exemple : La famille Texas Instruments

La famille C2xx

La famille C2xx est orientée pour le contrôle de processus : elle comprend une unité centrale DSP et toute une famille d'interfaces d'entrées sorties associées : ADC/DAC, SCI, SPI, Timer, PWM, CAN etc.

Caractéristiques du DSP :

- architecture virgule fixe 16 bits
- une performance de 100 MIPS en version 3V, 0.25-micron
- temps de cycle instructions de 10ns
- bus de données 32-bit on-chip : analyse deux mots en un simple cycle
- accumulateur 32-bit individuellement adressable en words ou bytes
- 16x16-bit multiplieur.
- multiplication et accumulation (MAC) en un cycle.
- option d'intégration flexible de mémoire: flash, static random access memory (RAM), read only memory (ROM)
- librairie complète de périphériques logiques et analogiques : support de DSP "full custom" (cDSP™) et produits standards.
- structure Statique CMOS, usage efficace de la mémoire, mode power-down (idle) qui minimize la consommation de puissance.
- port JTAG, non-intrusif, permettant l'émulation temps réel
- technique "Enhanced Real-time Data Exchange" (RTDX™) pour une meilleure visibilité des performances de l'application.
- technique de Microcontroller haute performance
- bus de programme et de données séparés (architecture Harvard modifiée) configurable en mode Von Neumann (mode bus combinés) supportant à la fois des opérations de type DSP et MCU
- espace d'adressage égal à 4M mots de 16bits pour à la fois les instructions et les données avec des instructions sur 16-bits ou 32-bits
- opérations de registre-à-registre
- opérations mémoire-à-mémoire simple-cycle, simple-word, pour réduire la taille du code et la performance en vitesse
- modes d'adressage efficaces pour réduire les besoins en mémoire programme
- pointeurs de pile spécifiques.
- gestionnaires d'interruptions multiples rapides : les opérations se font avec 80ns d'attente et 160ns pour une sauvegarde complète du contexte
- format d'instruction de type MCU avec les mnemonics: INSTR dest[,src1[,src2]]

La famille C5xx

La famille C5xx est orientée faible consommation pour les systèmes portable (téléphones GSM). Sa puissance de calcul est supérieure à celle de la famille C2xx. Par contre la puissance consommée par MIPS est très faible.

Le TMS320C55x digital signal processor (DSP) représente la dernière génération de DSP 'C5000 DSPs de Texas Instruments.

Le 'C55x est construit sur la compatibilité avec le 'C54x et il est compatible au niveau code source avec le 'C54x, cela protège l'investissement logiciel de l'utilisateur. En suivant les principes mis en place sur le 'C54x, le 'C55x est optimisé pour une faible consommation, un faible coût du système, et de meilleures performances pour les budgets très exigeants.

Avec une dissipation de la puissance du cœur aussi faible que 0.05 mW/MIPS à 0.9V, et une performance jusqu'à 800 MIPS (400 MHz), the TMS320C55x offre une solution faible coût dans les applications personnelles et portables ainsi que dans les infrastructure de communications digitales avec des budgets en puissance limités.

Comparé à un 120-MHz 'C54x, un 300-MHz 'C55x donne approximativement une performance de 5X supérieure et dissipe un sixième de la puissance d'un cœur de 'C54x.

La très faible dissipation de puissance d'un cœur 'C55x de 0.05mW/MIPS est obtenue à travers une intense attention portée au "low-power design" et aux techniques avancées de management de puissance. Les concepteurs du 'C55x ont implémenté une possibilité de configurer un niveau non-parallèle de "power-down" et un management de la puissance sans précédent qui se met en place de manière automatique, totalement transparente à l'utilisateur.

Le cœur du 'C55x est deux fois plus efficace que le 'C54x au travers d'une architecture double MAC (multiply-accumulate) avec un parallélisme des instructions, des accumulateurs supplémentaires, des ALUs, et de registres de données.

Un jeu d'instructions avancées, une évolution de celui du 'C54x, combiné avec une extension de la structure des bus complète les nouvelles unités d'exécution "hardware".

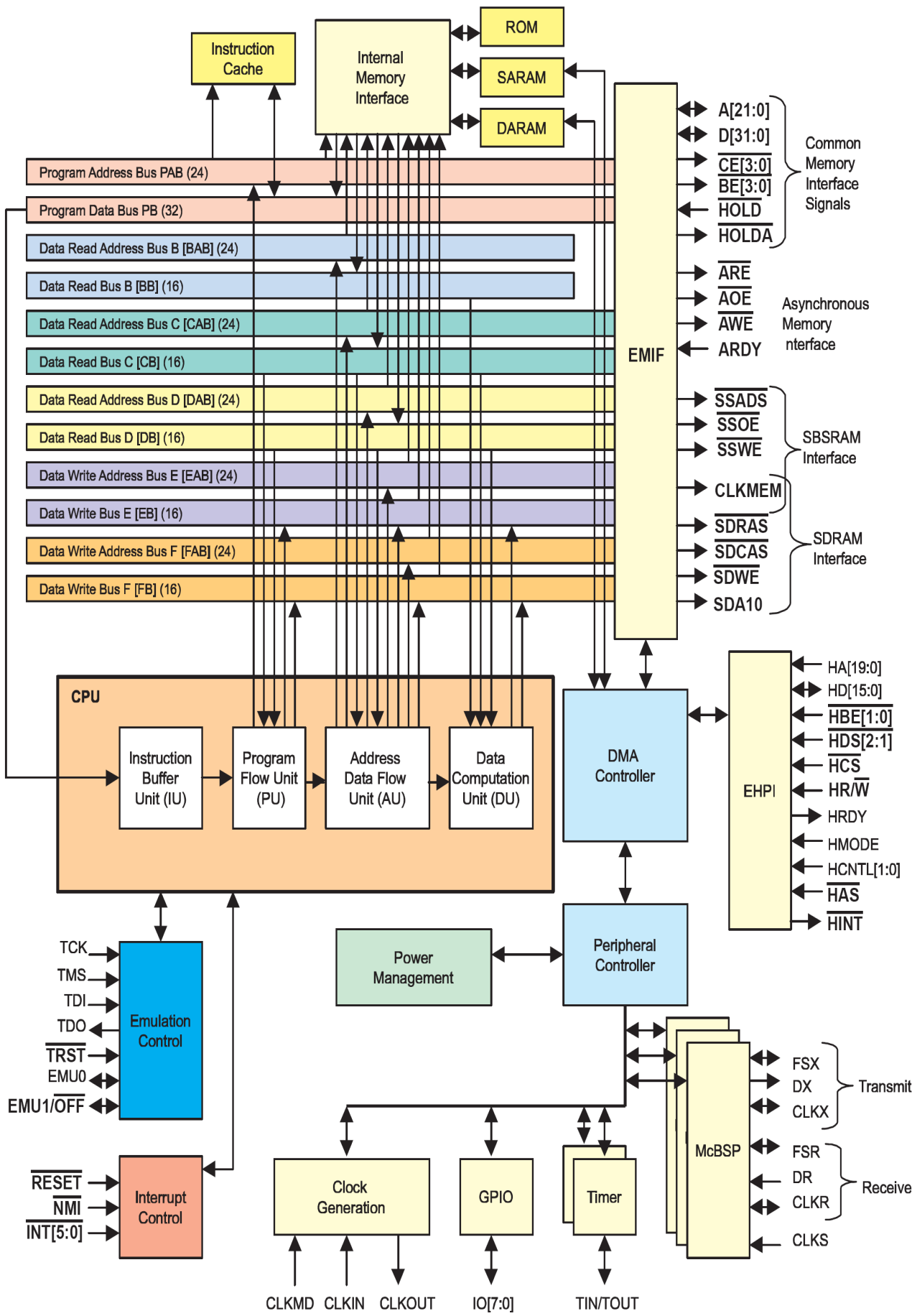
Le 'C55x continue le standard mis en place par le 'C54x en terme de densité de code et de leadership dans les systèmes faible coût. Les instructions du 'C55x sont de longueur variable en taille de 8 bits à 48 bits. Avec ce mot instruction de longueur variable, le 'C55x peut réduire la taille du code de control par fonction jusqu'à 40% par rapport à un 'C54x. La réduction de la taille du code de contrôle signifie la réduction des besoins en mémoire pour les systèmes faible coût.

Feature(s) Benefit(s)

A 32 x 16-bit Instruction buffer queue	Buffers variable length instructions and implements efficient block repeat operations
Two 17-bit x17-bit MAC units	Execute dual MAC operations in a single cycle
One 40-bit ALU	Performs high precision arithmetic and logical operations
One 40-bit Barrel Shifter	Can shift a 40-bit result up to 31 bits to the left, or 32 bits to the right
One 16-bit ALU	Performs simpler arithmetic in parallel to main ALU
Four 40-bit accumulators	Hold results of computations and reduce the required memory traffic
Twelve independent buses:	Provide the instructions to be processed as well as the operands for the various computational units in parallel to take advantage of the 'C55x parallelism.
– Three data read buses	– Two data write buses
– One program read bus	– One program address bus
User-configurable IDLE Domains	Improve flexibility of low-activity power management

Byte Address† (Hex)	Memory Blocks	Block Length
000000	DARAM‡ (8 blocks)	65,536 bytes
010000	SARAM§ (32 blocks)	262,144 bytes
050000	External¶ – $\overline{CE0}$	3,866,624 bytes
400000	External¶ – $\overline{CE1}$	4,194,304 bytes
800000	External¶ – $\overline{CE2}$	4,194,304 bytes
C00000	External¶ – $\overline{CE3}$	4,161,536 bytes
FF8000	ROM# if $\overline{MP/MC}=0$ (1 block) External¶ – $\overline{CE3}$ if $\overline{MP/MC}=1$	32,768 bytes
FFFFFF		

† Address shown represents the first byte address in each block.
‡ Dual-access RAM (DARAM): two accesses per cycle per block, 8 blocks of 8K bytes.
§ Single-access RAM (SARAM): one access per cycle per block, 32 blocks of 8K bytes.
¶ External memory spaces are selected by the chip-enable signal shown ($\overline{CE}[0:3]$). Supported memory types include: asynchronous, synchronous DRAM (SDRAM), and synchronous burst SRAM (SBSRAM).
Read-only memory (ROM): one access every two cycles, one block of 32K bytes.



La famille C6xx

La famille des composants C6xx est celle des hautes performance en particulier pour le traitement video . On trouvera ci-dessous le résumé des caractéristiques du C64xx.

Nous vivons dans un monde où les données à traiter sont nombreuses: données financières, médicales, sportives etc.

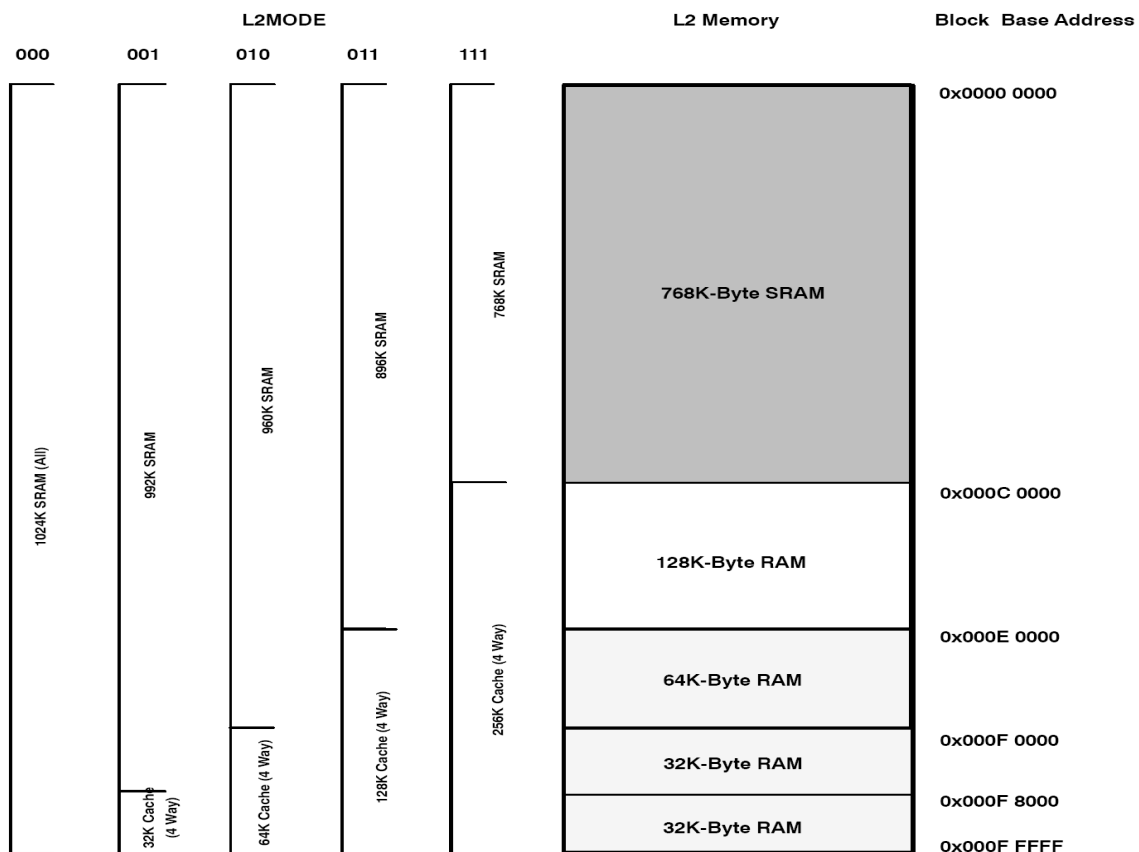
Dans cet univers les données qui peuvent être audio, video, ou écrites, sont transmises sur un unique medium. Ce medium peut être de technologie sans fil, satellite, cable, ou boucle digitale (DSL). Tous ces médias ont en commun le fait de devoir traiter les données très rapidement.

La famille 'C6000 avec l'architecture "VelociTI" correspond à ces demandes. Introduite pour la première fois en 1997 avec les coeurs 'C62x and 'C67x , la famille 'C6000 utilise l'architecture avancée de type "very long instruction word" (VLIW).

Cette architecture contient de multiples unités d'exécution fonctionnant en parallèle, ce qui permet de réaliser des instructions multiples en un seul cycle d'horloge. Le parallélisme est la clef pour réaliser de très hautes performances. A une fréquence d'horloge de 200 MHz et 1600 million d'instructions par seconde (MIPS) au début, le 'C6201 permettait des performances de dix fois celles des solutions précédentes (DSP). Aujourd'hui, la famille 'C62x permet d'atteindre 2400 MIPS à une fréquence d'horloge de 300 MHz, ce qui accroît les performances de 50% par rapport aux trois années précédentes.

Le nouveau membre de la famille 'C6000 , le 'C64x, accroît encore le plus haut niveau de performance pour le traitement des données. A une fréquence d'horloge de 1.1 GHz et plus, the 'C64x peut traiter les informations à une vitesse de 8800+ MIPS ou presque neuf billion d'instructions par seconde. Cette année (2001), les échantillons de CPU à 600-800 MHz, proposent un niveau de performance initial de 4800-6400 MIPS. En plus de la fréquence d'horloge, le traitement en un cycle peut être réalisé grace aux extensions VelociTI.2 de l'architecture VelociTI. Ces extensions incluent de nouvelles instructions pour accélérer les performances de l'application et utiliser le parallélisme de l'architecture.

La structure de l'unité centrale 'C64x (CPU) décrite sur la Figure 1, consiste en huit unités fonctionelles, deux files de registres, et deux chemins de données. Comme le 'C62x/'C67x, deux de ces huit unités fonctionelles sont des multiplieurs. Le multiplieur du 'C64x a été conçu pour être capable de réaliser deux multiplications 16-bit x 16-bit à chaque cycle d'horloge. Cela double par rapport à un 'C62x/'C67x; quatre multiplications 16-bit x 16-bit peuvent être exécutées à chaque cycle sur le 'C64x. En utilisant 750 MHz ce qui représente à peu près les performances du 'C64x, cela signifie que trois billion de multiplications 16-bit peuvent être réalisées chaque seconde. De plus, chaque multiplieur sur le 'C64x a la possibilité de réaliser quatre multiplications 8-bit x 8-bit à chaque cycle d'horloge. A 750 MHz, cela est équivalent à six billion de multiplications 8-bit par seconde. Les données de type huit-bit sont courantes dans la cadre du traitement des images, l'une des applications que peut prendre en charge un 'C64x.



Unité de Multiplications

.M unit (.M1, .M2)

16 x 16 multiply operations
16 x 32 multiply operations
Quad 8 x 8 multiply operations
Dual 16 x 16 multiply operations
Dual 16 x 16 multiply with add/subtract operations
Quad 8 x 8 multiply with add operations
Bit expansion
Bit interleaving/de-interleaving
Galois Field Multiply
Rotation
Variable shift operations

Unité Arithmétique et Logique : ALU

.L unit (.L1, .L2)

32/40-bit arithmetic and compare operations
32-bit logical operations
Leftmost 1 or 0 counting for 32 bits
Normalization count for 32 and 40 bits
Byte shifts
Data packing/unpacking
5-bit constant generation
Dual 16-bit arithmetic operations
Quad 8-bit arithmetic operations
Dual 16-bit min/max operations
Quad 8-bit min/max operations
Quad 8-bit subtract with absolute value

Registre à Décalage et ALU

.S unit (.S1, .S2)

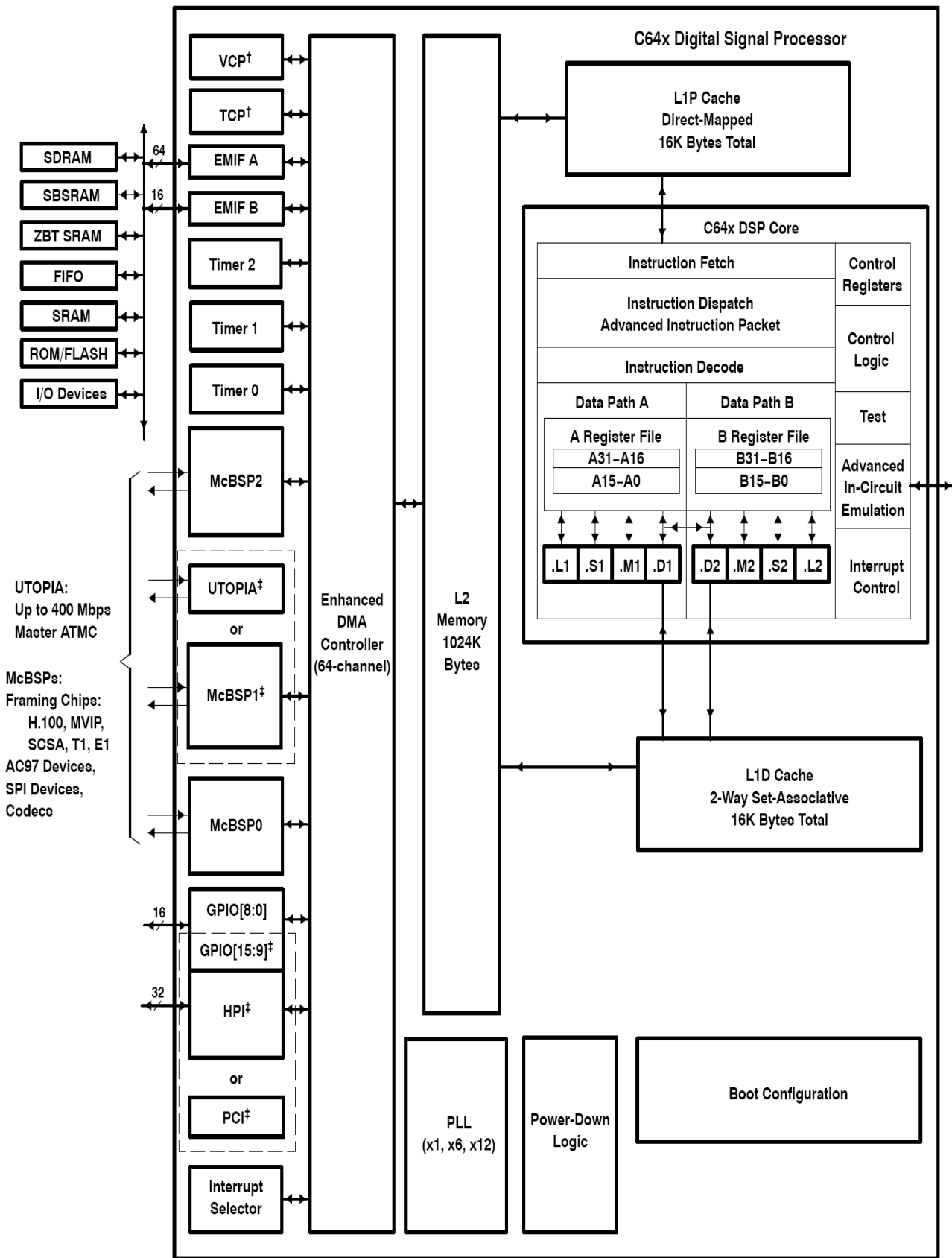
32-bit arithmetic operations
32/40-bit shifts and 32-bit bit-field operations
32-bit logical operations
Branches
Constant generation
Register transfers to/from control register file (.S2 only)
Byte shifts
Data packing/unpacking
Dual 16-bit compare operations
Quad 8-bit compare operations
Dual 16-bit shift operations
Dual 16-bit saturated arithmetic operations
Quad 8-bit saturated arithmetic operations

Générateur d'Adresses

.D unit (.D1, .D2)

32-bit add, subtract, linear and circular address calculation
Loads and stores with 5-bit constant offset
Loads and stores with 15-bit constant offset (.D2 only)
Load and store double words with 5-bit constant offset
Load and store non-aligned words and double words
5-bit constant offset generation
32-bit logical operations

functional block and CPU (DSP core) diagram



† VCP and TCP decoder coprocessors are applicable to the C6416 device only.

‡ For the C6415 and C6416 devices, the UTOPIA peripheral is muxed with McBSP1, and the PCI peripheral is muxed with the HPI peripheral and the GPIO[15:9] port. For more details on the multiplexed pins of these peripherals, see the Device Configurations section of this data sheet.