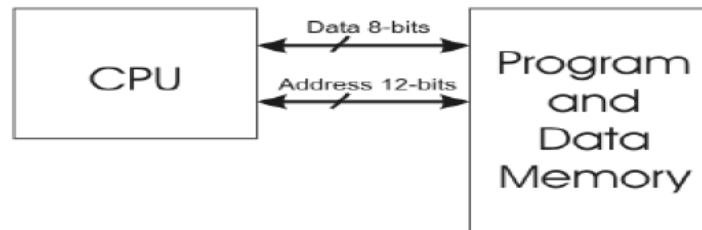


Cours SET (Master 1 ST)

Chapitre 3 : Architecture des processeurs embarqués :

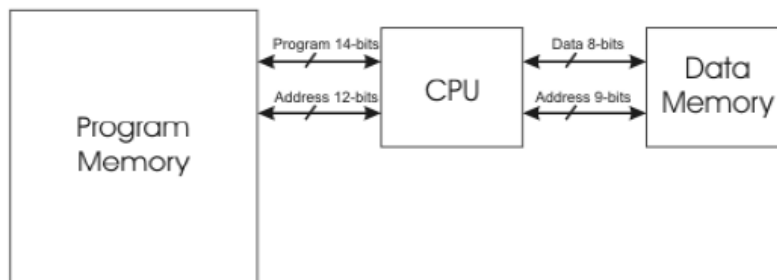
3.1 Principaux concept d'architecture des processeurs embarqués :

Architecture von neumann : L'architecture von neumann était l'une des premières architectures où le programme et les données sont stockés dans la même mémoire et sont accessibles sur le même bus. Par conséquent, la CPU effectue une opération à la fois pour cela il charge les instructions depuis la mémoire.



Architecture Harvard : L'architecture de Harvard est une modification par rapport à l'architecture de Von-Neumann. La mémoire est séparée ; une pour accéder au code (programme) et une pour accéder aux données et cela avec des bus séparés pour chacune. Les deux instructions : programme et les données peuvent être récupérées en même temps.

La vitesse opérationnelle est supérieure à Von Neumann



CISC (Complex Instruction Set Computer) : Pour rendre la programmation plus facile et plus rapide, les ordinateurs supportaient un grand nombre d'instructions. Ces instructions peuvent effectuer des opérations complexes qui résulte en un code compact mais complexe à générer. Une seule instruction peut extraire un ou plusieurs opérandes et effectuer une ou plusieurs opérations sur ces opérandes. Ainsi le temps d'exécution pour les instructions varie. Les processeurs qui utilise ce principe : Vax, Motorola 68000, Intel x86/Pentium

Comme les mémoires étaient très coûteuses, les concepteurs voulaient un ensemble d'instructions denses (pour réduire les besoins en mémoire).

RISC (Reduced Instruction Set) : Au début des années 1980, les concepteurs de processeurs se sont concentrés sur la réduction de la taille et de la complexité des jeux d'instructions. Basé sur des instructions plus simples, toutes de même taille, ayant toutes (presque) le même temps d'exécution pourraient accélérer la conduite et ainsi améliorer les performances. ensuite, un ensemble d'instructions simple implique moins de matériel informatique et donc un coût réduit. Par conséquent, l'objectif de conception était de fournir des instructions simples de base, qui pourraient s'exécuter plus rapidement dans un CPU. Les processeurs qui utilise ce principe : SPARC, MIPS, ARM, PowerPC, etc.

Accélération en pipelinant l'exécution (entre 3 et 7 étages de pipeline pour une instruction) ⇒ augmentation de la vitesse d'horloge

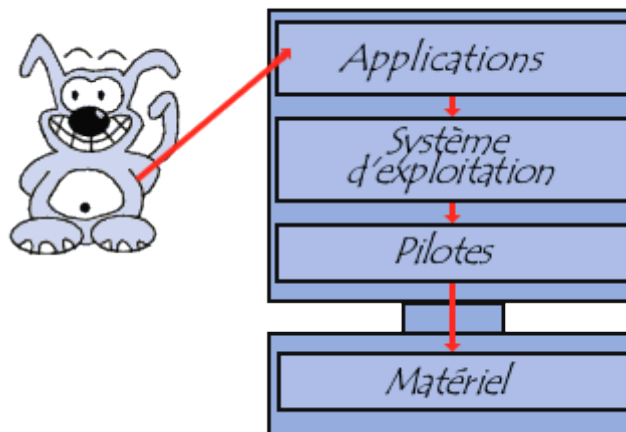
DSPs : ce sont des processeurs spéciaux Utilisés pour les applications de traitement du signal. Les unités MAC (Multiplication et Accumulation) et Shifter (Arithmétique et décalage logique) sont ajoutées aux cœurs DSP car les algorithmes de traitement du signal dépendent fortement de telles opérations. Comme les applications de traitement du signal nécessitent beaucoup de données, la bande passante des E / S de données de ces processeurs est conçue pour être élevée. De nos jours, de nombreux systèmes embarqués utilisent des applications de traitement du signal (téléphones cellulaires, lecteurs multimédia portables, etc.).

VLIW architecture : (**very long Instruction word** : **mot d'instruction très long**) : cette architecture est constituée de plusieurs ALU (UAL : unité arithmétique et logique : fait les calculs au niveau du CPU) en parallèle. Ces architectures ont été conçues pour exploiter le "parallélisme des niveaux d'instruction" dans l'application.

SIMD : "**Single Instruction Multiple Data**". Les architectures SIMD ont plusieurs ALU. Cependant, dans un cycle de processeur unique, la même instruction (opération) doit être exécutée sur toutes les ALU. Le processeur SIMD exécute donc une instruction identique (unique) sur différentes entrées de données (multiples), dans un cycle donné. Les processeurs SIMD exploitent le "Data Level Parallelism" d'une application.

3.2 systèmes d'exploitation pour systèmes embarqués :

Un système d'exploitation embarqué est un système d'exploitation pouvant être installé sur un système embarqué. Ce système d'exploitation est conçu avec des spécificités à gérer afin de répondre à des besoins spécifiques du système embarqué. Un OS sert d'intermédiaire entre l'application logiciel et le matériel, Ainsi lorsqu'un programme désire accéder à une ressource matérielle il lui suffit d'envoyer les informations au système d'exploitation, qui se charge de les transmettre au périphérique concerné via son pilote. En l'absence de pilotes il faudrait que chaque programme reconnaisse et prenne en compte la communication avec chaque type de périphérique !



un système d'exploitation embarqué doit avoir un niveau de robustesse bien au-dessus des exigences d'un système d'exploitation de bureau⁵. Les systèmes d'exploitation embarqués nécessitent une très grande fiabilité, ainsi que de bonnes performances.

Les systèmes d'exploitation pour PC sont conçus avec une interface homme-machine particulière (écran-clavier-souris). Dans le cadre d'un système d'exploitation embarqué, l'interface homme-machine pour pouvoir interagir avec peut être spécifique (clavier à digicode, écrans de smartphones...), voire inexistante (cartes de crédits, cartes sim, ..), auquel cas on utilise une machine intermédiaire (bornes, téléphones).

Académique :

ExoKernel, SPIN, Think
Choices, OSKit, Coyote, PURE, 2K

Commercial:

VxWorks, QNX, Windows CE, Windows XP embarqué
JavaOS, MMLite, icWORKSHOP, Pebble
Open Source de qualité industrielle :
Linux, RTLinux (real time linux), eCos

VxWorks : c'est un système d'exploitation temps réel multitâche, il a été employé par la NASA pour les missions spatiales du programme Discovery, Mars Pathfinder, Stardust, Messenger, Dawn ... Il est principalement employé par la recherche et l'industrie (aéronautique, automobile, transport, télécommunication), mais également dans de nombreux systèmes de communication d'entreprise : Intel, PowerPC, ARM, MIPS. Les langages utilisés avec ce système d'exploitation sont le C, le C++ et l'Ada

Windows CE : Windows CE est optimisé pour les appareils possédant une faible capacité de stockage - le noyau peut tourner avec moins d'un mégaoctet de mémoire vive, son architecture est de 32bits multitâche. Les systèmes sont souvent produits sans disque de stockage. Windows CE est conforme à la définition d'un système d'exploitation temps réel. exemple : PDAs, Téléphones, Consoles de Jeux fixe, Web tablets, Téléviseur interactif , guichets/bornes automatique (billettique), un exemple l'OS de la X-Box.

Linux : Depuis les années 2000, le noyau Linux est utilisé sur du matériel informatique allant des téléphones portables aux super-ordinateurs, exemple : le cas d'Android, qui équipe plus de 80 % des smartphones.

Le noyau Linux équipe la plupart des systèmes embarqués, civils ou militaires (box, robots, aérospatial, drones...).

cibles : UP et UC 16bit et 32bits sans MMU, Motorola 683xx, Motorola ColdFire, Intel i960, ARM7TDMI, Altera NIOS, ...

3.3 Processeurs à usage spécifique et processeur à usage général :

Un processeur est un système numérique dont le but est de traiter des données par une succession d'étapes simples. il doit acquérir des données, les traiter, et produire un résultat.

Les processeurs à usage général :

Ils sont caractérisés par un jeu d'instructions et peuvent être programmés en logiciel. Ils ont la possibilité d'effectuer n'importe quelle tâche sans changer de structure ni de interconnexions. Ils sont à usage général fabriqués à très grande échelle. On les appelle souvent microprocesseur ou unité centrale de traitement. Par exemple Un microcontrôleur, les processeurs Core de Intel et les processeurs FX-8000 de AMD, les ARM, MIPS, TMS320 ... etc.

Les processeurs à usage spécifique :

Ces des processeurs programmable en matériel conçus dans le but de répondre à un besoin unique. Ils sont moins complexes et plus efficaces que les processeurs à usage général. Leur conception demande souvent beaucoup d'efforts. Exemple : DSP, transmetteur RS232 ... etc.

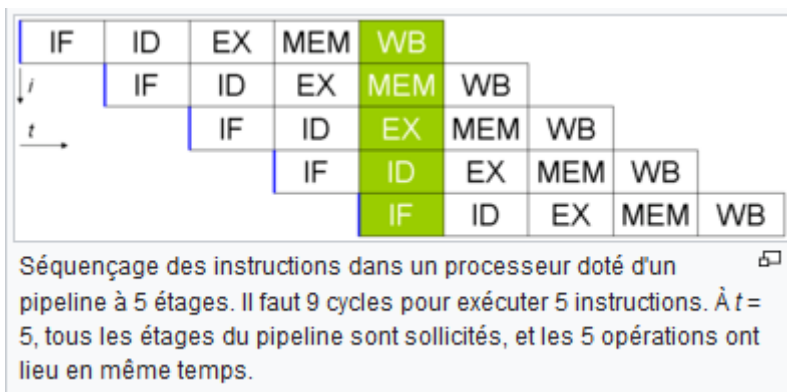
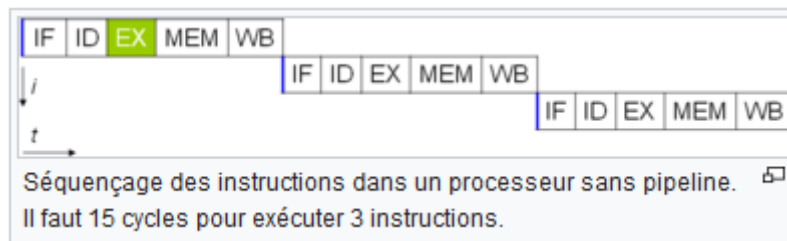
3.4 fonctionnement pipeline :

un pipeline est l'élément d'un processeur dans lequel l'exécution des instructions est découpée en plusieurs étapes. le processeur peut commencer à exécuter une nouvelle instruction sans attendre que la précédente soit terminée. L'architecture en pipeline permet ainsi d'accélérer

l'exécution des instructions. Chacune des étapes d'un pipeline est appelé étage. Le nombre d'étages d'un pipeline est appelé sa profondeur.

Exemple avec un pipeline à 5 étages :

- Lecture de l'instruction (IF)
- Décodage de l'instruction (ID)
- Exécution de l'instruction (EX)
- Accès mémoire (MEM)
- Ecriture du résultat (WB)



3.5 Hiérarchie mémoire :

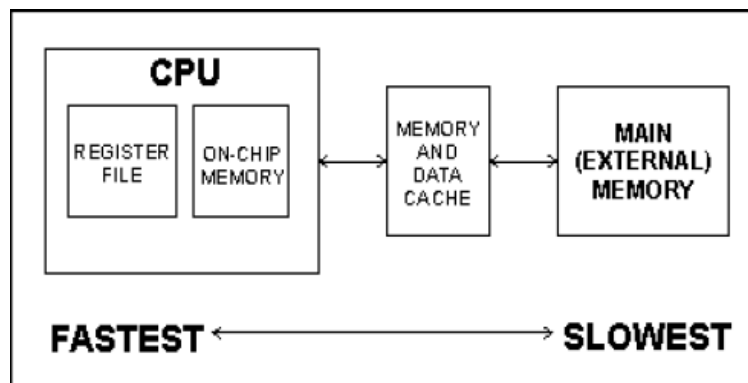
L'un des rôles principaux du concepteur de systèmes embarqués est de faire en sorte que tout ce qui est nécessaire soit le plus près possible du CPU (Principe de localité) Cela donne les résultats suivants :

- Utiliser des techniques telles que l'accès direct mémoire (DMA)
- Utiliser des techniques architecturales

Niveaux de hiérarchie pour la mémoire

- La mémoire interne (“on-chip”) : c’est le premier niveau , elle est constituée par les registres du processeur et contient les variables temporaires et intermédiaires. C’est la mémoire la plus rapide et la plus chère
- Le système cache mémoire : c’est le 2eme niveau, Il est aussi rapide (et cher) mais moins cher que le premier niveau. Il est utilisé pour mettre les insructions et les données près du CPU pour accès rapide aux données utilisées fréquemment ou récemment. Le système de cache utilise de la SRAM
- La mémoire externe : (« off-chip ») : c’est le 3eme niveau, Il tend à être moins rapide (et moins cher) que les autres types de mémoire, c’est l’endroit ou généralement sont sauvegardées les données et instructions non utilisées (stockage à long terme). L’accès à l’information de cette mémoire nécessite plus de “handshake” et de contrôle et donc prend plus de temps.

Exemple d’hiérarchie mémoire pour les DSP :



3.6 Périphériques et interfaces :

Les périphériques servent à élargir les fonctionnalités du système. On distingue deux types de périphériques. Un premier type concerne la communication avec des composants externes à la puce, tels que la mémoire externe, la gestion du temps (oscillateurs, horloges), ou des interfaces externes (ports USB, Ethernet, CAN, SPI,mémoires etc). Le deuxième type de périphériques concerne l'accélération de traitement de données spécifiques. par exemple, les cartes graphiques pour accélérer les algorithmes vidéo.

Du point de vue processeur, un périphérique est généralement accessible par une interface : on dit alors que le périphérique est « adressable », Un logiciel, par l'intermédiaire d'un processeur, peut alors le configurer et l'utiliser par de simples accès mémoire en lecture et écriture.

Exemple de périphériques : disque dur, écran, clavier, souris, carte son, afficheur, leds ...

Un périphérique doit utiliser des protocoles et communiquent à travers des bus.

Bus USB : c'est un Bus polyvalent pour connecter un périphérique électronique

Il y a Au moins deux paramètres impliqués dans communication USB

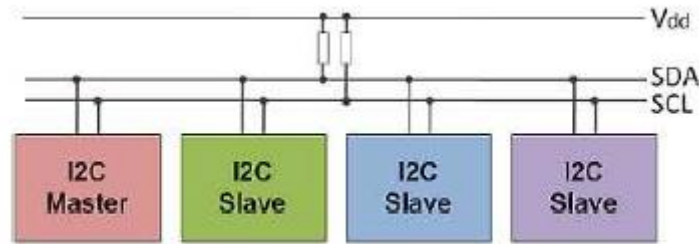
- Hôte dirige le trafic vers les périphériques et interroge régulièrement tous les périphériques USB
- Périphérique répond aux requêtes de l'hôte

Le bus série UART : Universal Asynchronous Receiver/Transmitter

- Communication avec un unique circuit
- Communication série full duplex et asynchrones

Bus I2C : Inter-Integrated Circuit

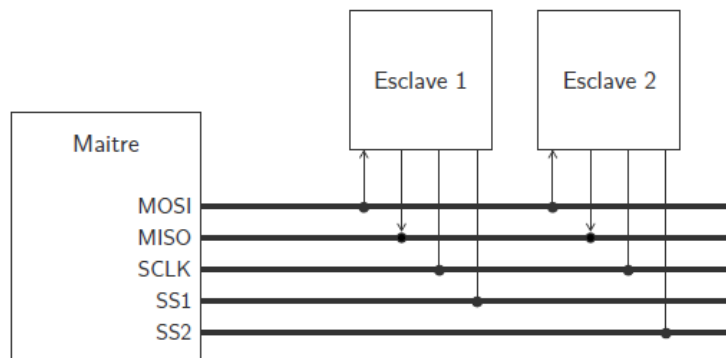
- Communication entre périphériques on-board et le CPU ou Connexion de sous-systèmes d'une même board entre eux Mémoire de type EEPROM, convertisseurs DAC et ADC...
- Bus deux fils Communications maitre-esclave initiées par le Processeur



Débit : **SPI 4Mbps - I2C 400kbps**

Bus Serial Peripheral Interface (SPI) : similaire à I2C

- Représente des Débits plus grands que I2C et UART et transferts bidirectionnels
- Bus trois fils + un slave select par circuit relié
- Communications maitre-esclave initiées par le
- Processeur permettant une Communication dans les deux sens de manière simultanée
- Flux de données efficaces pour application multimédia : LCDs, vidéos, traitement de signal, télécommunication...



Comparaison entre les bus :

Bus	Transmission	Sens	Synchronisation	Débit max
UART	série	full-duplex	asynchrone	M/S 3686,4 kbit/s
USB	série			10 Gbit/s
I ² C	série		synchrone	Ms/Ss de 100 kbit/s à 5 Mbit/s
SPI	série	full-duplex	synchrone	M/Ss 10 Mbits/s

3.7 Mécanismes de communication et protocoles associés :

La communication entre le processeur embarqué et le périphérique se fait à travers des bus. Cette communication est soit série soit parallèle, on dit que la communication est série si on transmet un bit à la fois, elle se dit parallèle si on transmet plusieurs bits en même temps.

La communication peut être synchrone si l'émetteur et le récepteur sont synchrones pour avoir les mêmes données (par exemple connexion réseau internet), elle peut être asynchrone Si L'émission et la réception de chaque message se réalisent en des temps distincts, séparés par un délai plus ou moins long. (le fax et la messagerie électronique).

Les processeur synchrone cumulent à la fois un intérêt de vitesse de traitement, mais aussi d'économie d'énergie. Ils intéressent énormément pour ce dernier point les industriels concevant des systèmes embarqués autonomes (téléphonie mobile, sonde...).

Le half duplex et le full duplex détermine le sens de communication supportés : le half duplex présente un seul fil et un seul sens à la fois (alternance pour une communication bidirectionnelle), le full duplex permet une communication bidirectionnelle simultanée.

Protocoles de communication : Port série : le port série est le port de communication le plus important dans un système embarqué, il se réfère au standard RS232 remplacé sur un ordinateur par un périphérique USB émulateur

un protocole de communication est une spécification de plusieurs règles pour un type de communication particulier.

Concept

Communiquer consiste à transmettre des informations, mais tant que les interlocuteurs ne lui ont pas attribué un sens, il ne s'agit que de données et pas d'information. Les interlocuteurs doivent donc non seulement parler un langage commun mais aussi maîtriser des règles minimales d'émission et de réception des données. C'est le rôle d'un protocole de s'assurer de tout cela.

Exemple d'utilisation

Lorsqu'on est en session X Window avec un ordinateur distant sur une ligne RNIS , la communication est payée au temps. On peut dans la plupart des cas déconnecter la session de bas niveau RNIS au bout de quelques secondes d'inactivité tout en maintenant la connexion de niveau supérieur TCP/IP. Ainsi lorsqu'un message TCP/IP est envoyé, le pilote RNIS réétablit la communication en moins de deux secondes, donnant l'illusion d'une continuité de liaison avec des prix bas. Pour TCP/IP, la liaison n'aura jamais semblé être coupée.

- Un réseau numérique à intégration de services (RNIS,
- TCP (Transmission Control Protocol) et IP (Internet Protocol).
- TCP/IP est l'ensemble des protocoles utilisés pour le transfert des données sur Internet.

Exemple d'architecture

