

Chapitre I.

1^{ère} partie

Introduction aux systèmes embarqués

1. Introduction

La traduction la plus souple et la plus fidèle du terme systèmes embarqués, du français à l'anglais, est « Embedded Systems ». Embedded signifie : Enfoui ou intégré. Et selon la langue de Molière, le mot embarqué, est dérivé du verbe « embarquer » qui Désigne le fait de « mettre quelque chose à bord d'un navire, d'un avion ou d'un véhicule ». Autrement dit ces systèmes sont cachés ou invisibles [1].

Les systèmes embarqués envahissent notre espace de vie, ils sont avec nous là où nous allons et partout autour de nous, voire même à l'extérieur ou à l'intérieur de notre corps. Sans le savoir ou sans se rendre compte, nous en croisons des dizaines quotidiennement. Ils sont donc là fidèles aux postes, prêts à nous rendre service, discrets, efficaces et dédiés à ce à quoi ils sont destinés. Omniprésents, ils le sont déjà et le seront de plus en plus. Ils sont bourrés d'électronique plus ou moins complexe et d'informatique plus ou moins évoluée. Dans ce cours nous présentons une définition plus précise d'un système embarqué ainsi que ces principales fonctions et caractéristiques.

2. Définition d'un système embarqué

Un système embarqué est un système complexe qui intègre du logiciel et du matériel conçus ensemble afin de fournir des fonctionnalités données. Il n'est pas visible en tant que tel, mais il est intégré dans un équipement doté d'une autre fonction. Le système matériel et l'application sont intimement liés et immergés dans le matériel ; c'est à dire le soft est noyé dans le hard ; et ne sont pas aussi facilement discernables comme dans un environnement de travail classique de type ordinateur .Un système embarqué est autonome et ne possède pas des entrées/sorties standards tels qu'un clavier ou un écran. Contrairement à un PC, l'interface homme machine d'un système embarqué peut être aussi simple qu'une diode électroluminescente LED qui clignote ainsi qu'un afficheur à cristaux liquides LCD de structure généralement simple ou aussi complexe qu'un système de vision de nuit en temps réel, un système avionique critique ou un robot.

Le logiciel a une fonctionnalité fixe à exécuter qui est spécifique à une application. Et, par conséquent, l'utilisateur n'a pas la possibilité de le modifier, une fois le système est conçu,

indépendamment de ses contraintes. Bien souvent, l'exploiteur n'a pas conscience d'utiliser un système à base de microprocesseurs. Les systèmes embarqués contiennent généralement un ou plusieurs microprocesseurs destinés à exécuter un ensemble de programmes définis lors de la conception et stockés dans des mémoires.

Ils sont désormais utilisés dans des applications diverses tels que le transport (avionique, automobile, ferroviaire), dans les équipements électriques et électroménagers : télévision, lave linge, lave vaisselle...

Afin d'optimiser les performances et la fiabilité de ces systèmes, des circuits numériques programmables, des circuits dédiés à des applications spécifiques ou des modules analogiques sont utilisés. Nous allons voir quelques exemples plus loin. .

3. Exemple typique de système embarqué

Dans ce qui suit quelques exemples de systèmes embarqués que vous ne saviez peut-être pas qu'ils sont à la base d'un microprocesseur et un logiciel très sophistiqués

- *Grand public* : distributeurs automatiques de boissons, Appareils photographiques et caméras, lecteurs DVD, chauffage et climatisation, éclairage, électroménager, domotique, sécurité (incendie, intrusion, surveillance), ascenseurs, audio et vidéo, consoles de jeux, décodeurs, etc.
- *Transports*: Automobile, aéronautique, spatial, marine, assistance à la conduite ou au pilotage, maintenance, signalisation, contrôle du trafic aérien, maritime, distributeur de billets, radar, etc.
- *Défense*: Contrôle de trajectoire, lanceur, etc.
- *Secteur manufacturier et industrie* : Chaînes de production, automates, production et distribution d'électricité, réacteurs chimiques, réacteurs nucléaires, raffineries, dispositifs de sécurité, aide à la maintenance, etc.
- *Information et communication* : Imprimante, périphérique, téléphone, répondeur, fax, routeurs, téléphonie mobile, satellites...
- *Santé* : Imagerie médicale, diagnostique, soins, implants, handicapés, etc.
- *Autres* : Carte à puce, distributeurs, etc.

4. Domaines d'application des systèmes embarqués

Domaines « traditionnels »

- Avionique
- Robotique

- Automobile
- Militaire

Domaines « nouveaux »

- Jeux et loisirs
- Téléphonie, Internet mobile
- Implants, pacemaker (santé sécurité)
- Immeubles intelligents
- Villes intelligentes

5. *La meilleure conception d'un système embarqué*

Du point de vue technique, la conception d'un système embarqué demande à son concepteur d'être pluridisciplinaire : Electronique, informatique, réseaux, sécurité ... Mais le concepteur se doit aussi d'être un bon gestionnaire car concevoir un système embarqué revient finalement à un exercice d'optimisation : minimiser les coûts de production pour des fonctionnalités optimales. Alors le système embarqué se doit d'être :

- ✓ Robuste aux chocs, à la température, à l'humidité ...
- ✓ Simple.
- ✓ Fiable.
- ✓ Fonctionnel : Le système doit toujours fonctionner correctement.
- ✓ Sûr de fonctionner et à accorder une confiance justifiée au service surtout si la sécurité des personnes est en jeu.
- ✓ Tolérant aux fautes et résistant aux pannes : La panne d'un composant ne remet pas en cause la vie du système.
- ✓ Autonomie : les systèmes embarqués doivent être auto gérables et indépendants.
- ✓ Temps Réel : Fonctionnent généralement en Temps Réel

D'autres contraintes sont aussi à prendre en compte :

- ✓ L'encombrement. Mémoire limitée, pas de disque en général
- ✓ Le poids.
- ✓ Le packaging : difficulté de faire cohabiter dans un faible volume, électronique analogique, électronique numérique et RF sans interférences.
- ✓ Sensibilité effective à l'environnement extérieur.
- ✓ La consommation électrique.

- ✓ Le système embarqué nomade doit être de faible consommation car il est alimenté par des batteries. Une consommation excessive augmente le prix de revient du système embarqué car il faut alors des batteries de plus forte capacité.
- ✓ Le temps de développement. Dans un marché concurrentiel, il convient d'avoir un système opérationnel le plus rapidement possible pour être le premier sur le marché [2].

Devant toutes ces contraintes, le concepteur adopte des règles de bon sens :

- ✓ Faire simple.
- ✓ Utiliser ce que l'on a déjà fait ou fait par d'autres.
- ✓ Utiliser des technologies éprouvées qui ont fait leur preuve. Surtout si la sécurité des personnes est en jeu...

6. *Qualité et métriques des systèmes embarqués*

Si un produit de type système embarqué est commercialisé le client doit vérifier les conditions suivantes dont le concepteur doit tenir compte :

- ✓ *Puissance de calcul*: selon la fréquence du processeur et la largeur des registres.
- ✓ *Débit*: capacité d'exécution des tâches en thread.
- ✓ *Temps de réponse*: le système doit réagir vite
- ✓ *Mémoire*: l'estimation la plus précise possible de l'utilisation mémoire
- ✓ *Consommation d'énergie*: la conception doit prendre en compte la plus faible consommation possible.
- ✓ *Durée de vie* : paramètre important lors de la décision des composants à inclure dans le système
- ✓ *Installation du programme* : requiert des outils spécifiques
- ✓ *Testabilité et débogage* : très dur sans entrées : sorties de type clavier, écran...

7. *Classification des systèmes embarqués :*

La figure.1 illustre les différents types de systèmes embarqués, classés selon leurs tâches et leurs fonctionnements.

7.1 Système Transformationnel : produit des sorties en fonction d'entrées selon un processus de calcul indépendant de l'environnement ou avec un faible couplage.

Les données en entrée et les résultats engendrés sont gérés à l'initiative du système et Les traitements internes sont généralement importants.

7.2 Système Réactif : produit des sorties en fonction d'entrées et en réaction à des événements produits par l'environnement, c'est à dire il est en interactions permanentes avec l'environnement. En réponse aux stimuli le système provoque des réactions, il ne réagit que lors de l'élaboration des réactions et les instants de production des résultats sont contraints par la dynamique du procédé.

7.2.1 Système Interactif : les stimuli provoquent des réactions, leur prise en compte reste à l'initiative du système selon les événements reçus et par l'état courant le rythme de l'interaction est déterminé par le système et non par l'environnement.

7.2. 2 Systèmes temps réel : Système dont les entrées et les sorties sont soumises à des contraintes temporelles en plus des contraintes de correction fonctionnelles classiques, La plupart des systèmes sont multi-rate (Les opérations doivent être gérées à des vitesses différentes). On en distingue deux catégories :

7.2.2.1 Système temps réel Durs ou stricts (Hard real time) : le manquement des échéances provoque une faute.

7.2.2.2 Système temps réel souple (Soft real time) : le manquement des échéances cause des dégradations de performances.

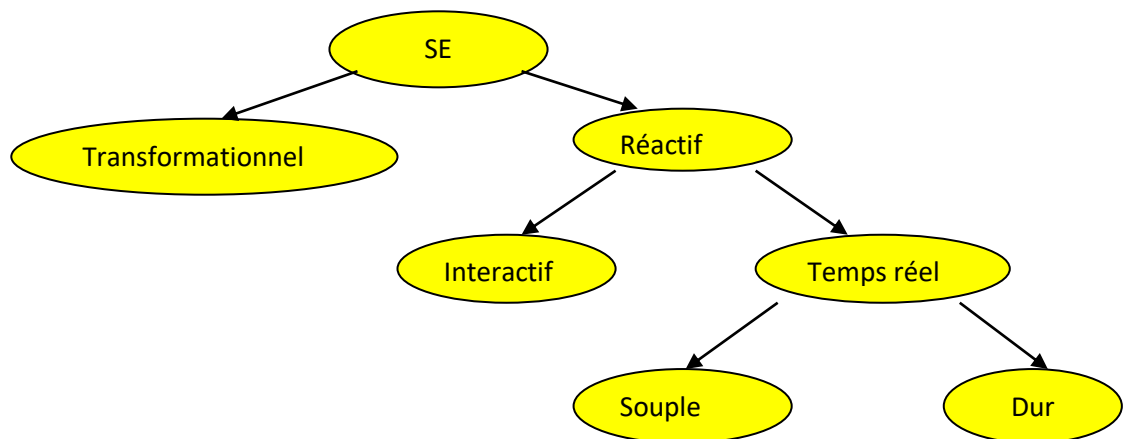


Figure .1 les différents types de SE

8. Architecture interne d'un système embarqué

Quelle que soit la nature et la complexité du système embarqué, il est toujours constitué de :

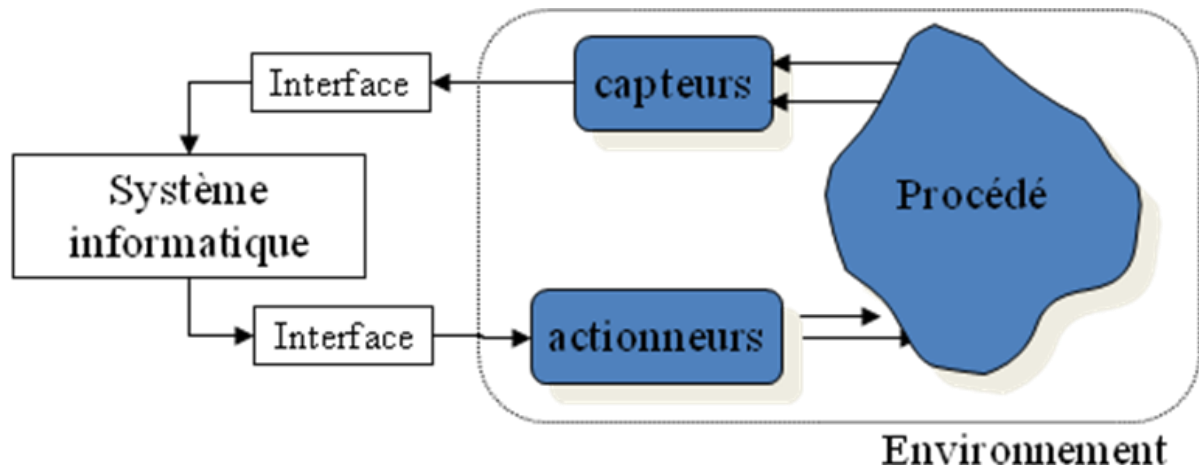


Figure.2 architecture interne d'un système embarqué

8.1 Le système contrôlé : Environnement ou autre systèmes équipés d'une instrumentation qui réalise l'interface avec le système de contrôle .Se sont des éléments qui remplacent les entrées/sorties dans un micro ordinateur standards, désignés généralement par des capteurs en entrées et des actionneurs en sortie.

8.1.1 Les actionneurs

Un actionneur est un système qui transforme l'énergie qui lui est fournie en un phénomène physique (en chaleur, champs magnétique, lumière, mouvement, position, pression, son) qui fournit un travail. L'actionneur fait partie de la partie opérative. Citons à titre d'exemple :

- ✓ Lumière à partir d'un courant électrique (Ecran LCD, diode électroluminescente, lampe...).
- ✓ Sons à partir d'un courant électrique (vibreux, avertisseur sonore...).
- ✓ Champ magnétique à partir d'un courant électrique (électro aimant).
- ✓ Rayonnement infrarouge à partir d'un courant électrique (diode émissive infrarouge).
- ✓ Chaleur à partir d'un courant électrique (résistance chauffante).
- ✓ Mouvement à partir d'un courant électrique (moteur électrique).
- ✓ Mouvement à partir d'un fluide sous pression (cas d'un vérin pneumatique ou hydraulique)[3].

8.1.2 Les capteurs

Les capteurs sont les éléments de base des systèmes d'acquisition de données. De se fait ils sont souvent utilisés comme des entrées dans les systèmes embarqués.

Un capteur est un dispositif qui transforme l'état d'une grandeur physique observée en une grandeur utilisable, exemple : une tension électrique, une hauteur de mercure, une intensité, la déviation d'une aiguille...

on peut classer les capteurs selon leurs apports énergétiques en capteurs **passifs** comme : la thermistance, le potentiomètre, le thermomètre à mercure : n'ont pas besoin de source d'énergie et capteurs **actifs** comme : chronomètre mécanique, jauge... Ils sont constitués d'un ensemble de transistors qui nécessitent une énergie pour fonctionner.

Comme on peut trouver aussi des capteurs analogiques qui génèrent des sorties analogiques de type courant ou tension et capteurs numériques qui délivrent des impulsions ou train d'impulsions.

8.2 Le système de contrôle :

8.2.1 Aspect Hardware

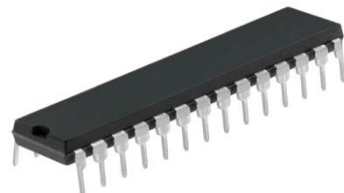
Le système de contrôle est un ensemble de composants électroniques reliés entre eux afin de contrôler le bon fonctionnement du système embarqué suivant un programme noyé dans la mémoire du microcontrôleur. Il est constitué de :

8.2.1.1 Le Microcontrôleur

En réalité, un système embarqué n'est pas doté d'un microprocesseur mais d'un microcontrôleur. On peut dire qu'un microcontrôleur est un ordinateur miniaturisé constitué d'un microcalculateur (microprocesseur) dédiée à faire une tâche précise que le concepteur traduit en programme chargeable afin de conduire le procédé à un état donné alors qu'un ordinateur est destiné à effectuer une variété de tâches de calcul , ce qui explique la différence des prix. D'un autre côté, il est très judicieux de noter la différence significative au niveau de la vitesse d'horloge.



Un processeur Intel 80486 SX 25 D



9306AC Micro peripheral: miscellaneous

Figure 1.2. Différence entre microcontrôleur et microprocesseur

Les microcontrôleurs sont conçus pour consommer moins d'énergie car ils sont intégrés à d'autres appareils qui peuvent avoir une plus grande consommation d'énergie dans d'autres zones.

Les schémas des figures suivants montrent la différence entre un microprocesseur et un microcontrôleur. On ne peut pas se contenter de la forme sous laquelle un circuit est présenté, pour juger mais l'architecture interne et le fonctionnement détaillés en data-sheet. La figure.5 montre la différence du point de vue architecture.

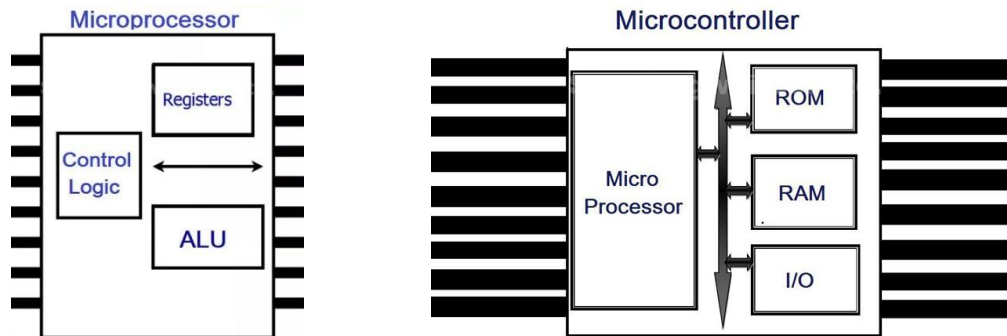


Figure1.3 structure interne d'un micro processeur et un microcontrôleur

8.2.1.2 La mémoire

Trois types de mémoire : RAM, ROM et la mémoire cache sont des éléments très importants des systèmes embarqués. La RAM est l'endroit où les composantes du programme sont stockés temporairement pendant l'exécution. La ROM contient les routines d'entrée-sortie de base qui sont nécessaires par le système au démarrage. Le cache est utilisé par le processeur comme un stockage temporaire lors du transfert de la transformation et de données

8.2.1.3 L'Horloge système

Comme tous les autres éléments, l'horloge du système est une partie très importante d'un système embarqué, tous les processus dans un système embarqué fonctionnent sur des cycles d'horloge. Cette horloge est généralement constituée d'un oscillateur et autres certains circuits associés.

8.2.1 Les ports

Un port est un dispositif capable de recevoir des données provenant d'un périphérique, d'un processeur ou d'un autre système extérieur. Un port permet d'envoyer des données vers l'extérieur à partir d'instructions dans le programme chargé et exécutées par le processeur. La connexion du port vers le processeur se fait via un bus (parallèle ou série), sur lequel on échange les informations. On trouve aussi des bus de données, des

bus de contrôle ou des bus d'instructions. ET, selon la manière avec laquelle le rythme de la communication se déroule on distingue deux caractéristiques des bus : synchrone ou asynchrone. Le premier suit les impulsions de l'horloge quant au deuxième transmet les données sur le canal de communication selon leurs validités.

8.2.2 Aspect software

8.2.2.1 Mode de fonctionnement des systèmes embarqué [6]

Ils existent plusieurs types de fonctionnement sur lesquels est basé le type de programme qui doit être installé sur système embarqué, citons à titre d'exemple :

8.2.2.1 .1 mode de fonctionnement général : Boucle infinie

Exemple

```
Tant que TOUJOURS faire
    Lecture_des_entrees();
    Traitement();
    Ecriture_des_sortie();
Fin tant que
```

8.2.2.1.2 Mode fonctionnement périodique (cyclique) : A une structure régulière, constituée d'un cycle infini, qui exécute un calcul puis se suspend en attendant l'activation périodique suivante.

Exemple :

```
Tant que( top de l'horloge)faire
    Lecture_des_entrees();
    Traitement();
    Ecriture_des_sortie();
Fin tant que
```

8.2.2.1.2 Mode fonctionnement apériodiques (Evénementiel) :

Dans ce cas il n'est pas possible de fixer un temps de séparation minimum entre deux tâches consécutives. De plus, ils n'ont pas de structure particulière. Avec ce type de tâche, nous pouvons modéliser :

- Des tâches qui répondent à des événements qui se produisent rarement.
- Tâches qui répondent à des événements qui se produisent avec une structure irrégulière.

```
lorsque( interruption)faire
    Lecture_des_entrees();
    Traitement();
    Ecriture_des_sortie();
Fin tant que
```

Nous reviendrons sur cette notion en détails dans les chapitres suivant.

9. Langages de programmation :

Un système embarqué doté d'un processeur a besoin de *code* pour faire quelque chose. Ce dernier est placé dans une mémoire accessible par le processeur afin de pouvoir être exécuté, instruction par instruction. Typiquement, le software d'un système embarqué sera présenté sous la forme d'une *image exécutable*, définitive et non modifiable.

Plusieurs langages de programmation se veulent destinés à cet usage parmi lesquels :

- ✓ Ada.
- ✓ Des langages proches de la machine comme le C et C++.
- ✓ Le langage assembleur.

Pour la programmation des systèmes embarqués critiques, des langages construits différemment ont été proposés :

- ✓ les langages sur une sémantique formelle : synchrones Esterel, Lustre, Signal, Lucid Sychrone... Ces langages sont actuellement utilisés avec succès dans l'industrie avionique. Par
- ✓ les langages à base de méthode formelle : méthode B, Coq, etc.
- ✓ Le transfert des fichiers binaires s'effectue généralement dans des formats 'texte' standardisés tel que le format S-Record de Motorola ou le format HEX (Intel) de Intel.

10. Langages de modélisation

Il existe plusieurs langages de spécification et de conception pour l'informatique embarquée :

- ✓ Modeling and Analysis of Real Time and Embedded systems (MARTE)
- ✓ Profil UML, Sysml standardisé par l'Object Management Group (OMG).
- ✓ Vhdl, verilog et systemC
- ✓ Specification and Description Language (SDL) pour les télécoms et plus généralement les systèmes communicants.
- ✓ Architecture Analysis and Design Language (AADL) orienté sur les aspects d'architecture ;
- ✓ Structured Analysis for Real Time (SART) convient à la modélisation des systèmes embarqués temps réel.
- ✓ Méthode formelle : permet de garantir des propriétés notamment celle de sûreté (comme la Méthode B).

La deuxième partie de ce chapitre sera dédié à la modélisation AADL
