

# CHAPITRE I: GENERALITES SUR LES SYSTEMES EMBARQUES

Les systèmes embarqués existent depuis longtemps. L'Apollo Guidance Computer, développé en 1961 au laboratoire d'instrumentation du *Massachusetts Institute of Technology (MIT)*, est le premier système embarqué. Il était utilisé en temps réel par les astronautes afin de recueillir et de fournir des informations sur le vol, mais aussi pour effectuer des contrôles automatiques de toutes les fonctions de navigation du vaisseau spatial.

De nos jours, ces objets communicants, des plus petits placés sur des objets ou sous la peau jusqu'aux plus gros (satellites), en passant par les téléphones portables, PDA, montres, Mp3, vêtements, les appareils médicaux, appareils électroménagers, consoles de jeux vidéo, automobiles, etc... ont profondément modifié notre comportement.

Les systèmes embarqués sont en forte croissance. Ils s'emparent à lui tout seul 95% des puces produites aujourd'hui. Ces systèmes jouent un rôle de plus en plus important dans de nombreux domaines d'applications. On citera par exemple le domaine de la santé pour construire des systèmes de soin, économiser l'énergie, contrôler le stock des aliments, etc.

## I - 1 CATEGORIES

On peut classer les systèmes embarqués en trois catégories : système transformationnel, système interactif et système réactif ou temps réel [1].

### I - 1 - 1 Le système transformationnel

*Le système transformationnel* est une activité de calcul, ils utilisent des données fournies à l'initialisation par l'utilisateur. Les données, leurs traitements et l'obtention du résultat n'ont aucune contrainte temporelle. Ainsi, les données sont disponibles au lancement et les instants de productions des résultats ne sont pas contraints.

### **I - 1 - 2 Le système interactif**

*Le système interactif* est en interaction quasi permanente avec son environnement, y compris après l'initialisation du système. La réaction du système est déterminée par les événements reçus et par l'état courant.

Dans les systèmes interactifs, les données sont produites par interaction avec l'environnement sous différentes formes (clavier, fichier, réseaux, souris, etc.). Mais le temps n'intervient pas en tant que tel si ce n'est avec un aspect confort de travail ou qualité de service.

### **I - 1 - 3 Le système réactif ou temps réel (SETR)**

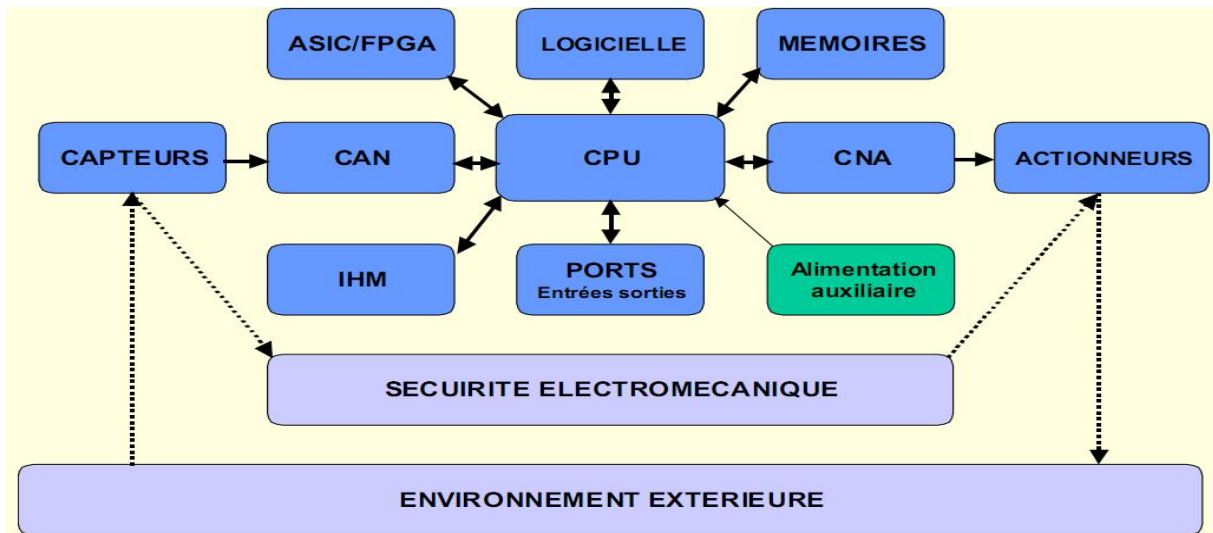
*Le système Réactif ou Temps Réel (SETR)* est en interaction permanente avec son environnement, y compris après l'initialisation du système. La réaction du système est fonction des événements et des réactions passées mais le rythme de l'interaction est déterminé par l'environnement et non par le système.

Les systèmes de contrôle-commande ou réactifs sont en relation avec l'environnement physique réel pour les données capteur/actionneur; l'aspect « temps » a une place importante sous la forme d'un temps de réaction, d'une échéance à respecter, etc.

Le SETR est le plus utilisé dans le domaine de la domotique. Ainsi, dans toute la suite nous allons axés notre analyse en se basant sur ce type de système.

## **I - 2 ARCHITECTURE MATERIELLE D'UNE SETR**

L'architecture matérielle d'un SETR est composé de plusieurs bloques comme le montre la Fig.1.2 suivant.

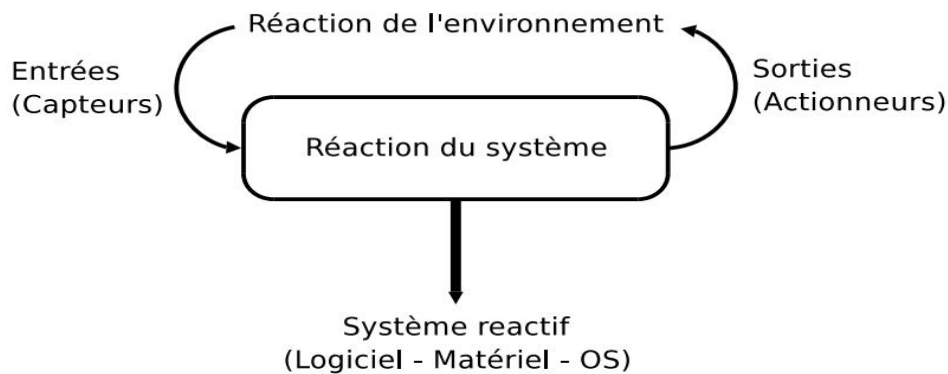


***Figure 1. 1 : Structure typique d'un système embarqué [2].***

Quelle que soit la nature et la complexité du système, on décompose un système embarqué en système contrôlé et en système de contrôle.

### **I - 2 - 1 Le système contrôlé**

Le système contrôlé est l'environnement du système réactif. C'est un procédé équipé d'une instrumentation qui réalise l'interface avec le système de contrôle. Le fonctionnement du système réactif est illustré par la Fig.1.2.



***Figure 1. 2 : Fonctionnement du système réactif.***

## I - 2 - 2 Le système de contrôle

Le système de contrôle concerne les éléments matériels et logiciels dont la mission est d'agir sur le procédé via les actionneurs en fonction de l'état de ce procédé indiqué par les capteurs de manière à maintenir ou à conduire le procédé dans un état donné, généralement, appelé « architecture matérielle », c'est l'ensemble des composants physiques qui est nécessaire d'ajouter à un processus pour réaliser une application temps réel. Il est constitué d'un calculateur et des transducteurs.

### a- **Calculateur**

Le calculateur peut être composé d'un ou plusieurs microprocesseurs, de circuits intégrés spécialisés comme les ASIC ou les FPGA ainsi que des médias de communication [3].

#### i. ASIC et FPGA

Les *Application Specific Integrated Circuits* (ASIC) et les *Field Programmable Gate Array* (FPGA) sont des circuits intégrés que l'on peut configurer pour réaliser à bas coût des fonctions « câblées » simples dont les temps de réaction sont extrêmement courts. La programmation de ces circuits n'est pas évidente car elle nécessite l'utilisation de programmeurs ou de procédures spécialisés. Ce circuit n'intègre pas de séquenceur d'instructions, ils ne sont capables que d'exécuter une seule fonction.

#### ii. Microprocesseurs, microcontrôleurs

Un microprocesseur est composé d'un Central Processing Unit (*CPU*) et d'*unités de communication* pour communiquer avec des périphériques externes ou d'autres microprocesseurs.

Le **CPU** est une machine séquentielle constituée généralement d'un séquenceur d'instruction (SI), d'une unité de traitement (UT) et d'une mémoire. Les dernières générations

de microprocesseurs peuvent aussi intégrer une unité de calcul flottant (FPU) dont le but est d'accéder à certains calculs mathématique (addition, multiplication, fonction trigonométrique, etc.).

Un *microcontrôleur* est un microprocesseur intégrant un certain nombre d'interfaces supplémentaires (mémoires, timers, Parallel Input Output, décodeurs d'adresse, etc.). Ces nombreuses entrées-sorties garantissent un interfaçage aisé avec un environnement extérieur tout en nécessitant un minimum de circuits périphériques ce qui les rends particulièrement bien adaptés aux applications temps réels embarquées. Du fait de leur forte intégration en périphérique, les microcontrôleurs sont souvent moins puissants que les microprocesseurs.

### iii. Médias de communication

Lorsque le calculateur intègre plusieurs processeurs on dit que l'architecture matérielle du système est multiprocesseur ou parallèle. Les médias de communications sont les éléments qui permettent aux processeurs d'un calculateur multiprocesseur d'échanger des données. On peut classer ces médias de communication selon trois catégories principales :

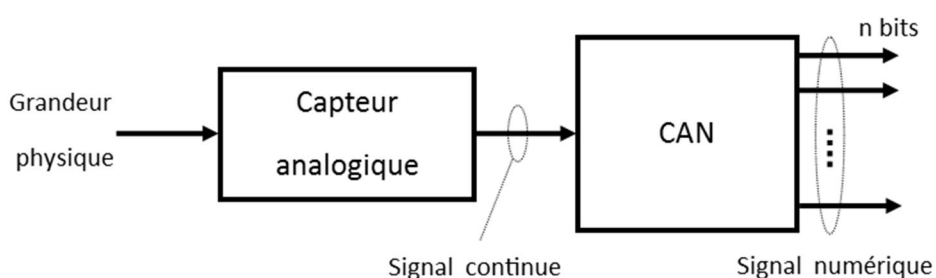
- média point à point Séquentiel Accès Memory (SAM) aussi appelée *lien* : c'est la liaison bidirectionnelle entre deux mémoires SAM de deux processeurs différents. L'accès à ce type de liaison est de type FIFO ou First In First Out, pour lequel l'ordre d'émission des données est aussi l'ordre de réception ;
- média multipoint SAM ou bus SAM : c'est aussi une liaison multidirectionnelle qui relie les mémoires SAM de plus de deux processeurs ;
- média multipoint RAM ou bus RAM : c'est aussi une liaison multidirectionnelle, mais ici contrairement aux médias précédents la communication est réalisée à travers une mémoire commune de type RAM.

### *b-* Transducteurs

Le calculateur interagit avec le processus par l'intermédiaire de transducteurs. Ceux qui permettent d'observer le processus sont appelés capteurs et ceux qui permettent d'agir sur le processus sont appelés actionneurs.

i. Capteur

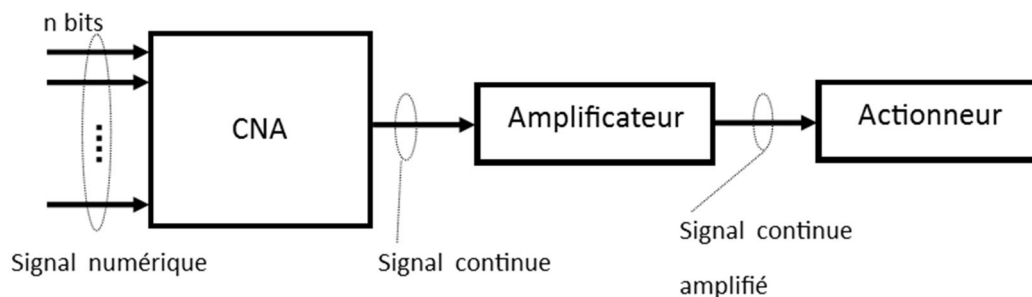
Un capteur est un dispositif conçu pour mesurer une grandeur physique en la convertissant en une tension ou un courant électrique. Le signal électrique issu d'un capteur est traité pour être converti en signal numérique pour pouvoir être traité par le calculateur. La numérisation est réalisée par le Convertisseur Analogique Numérique (CAN) qui est utilisé pour échantillonner le signal issu du capteur. La figure 1.3 montre l'utilisation d'un capteur dans un système embarqué.



***Figure 1. 3 : Codage d'un signal issu d'un capteur analogique.***

ii. Actionneur

L'actionneur est un dispositif qui convertit un signal électrique en un phénomène physique (moteur, vérin électrique, voyant, haut-parleur, etc.) censé modifier l'état courant du processus. Le Signal de commande fourni par le calculateur est un signal numérique que le Convertisseur Numérique Analogique ou CNA va convertir en signal électrique analogique. La figure 1.4 montre la commande d'un actionneur.



***Figure 1. 4 : Exemple de commande d'un actionneur.***

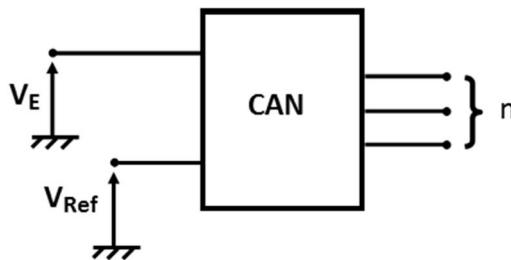
### I - 2 - 3 CNA/CAN

Longtemps, l'électronique avait resté exclusivement analogique, mais vue le progrès de la technologie, nous sommes désormais à la fin d'une période de passage vers une électronique numérique. Ainsi, dans différents appareils High-Tech, on retrouve soit de la CAN ou de la CNA. Les conversions analogique-numérique et numérique-analogique sont des ponts entre le monde numérique et le monde analogique.

#### a- CAN

Le convertisseur analogique-numérique (CAN) est un dispositif électronique permettant la conversion d'un signal analogique (signal continu en temps et en amplitude) en un signal numérique (signal échantillonné et quantifié, discret en temps et en amplitude).

La figure 1.5 suivante montre le symbole d'un CAN à n bits :



**Figure 1. 5 : Symbole d'un CAN.**

n : nombre de bits en sortie.

$V_{Ref}$  : la tension de références du convertisseur (Elle permet de fixer les amplitudes maximales et minimales de la grandeur d'entrée à convertir.)

$V_E$  : la tension d'entrée à convertir.

Un CAN est caractérisé essentiellement par :

- **Sa résolution**, la plus petite variation en entrée correspond à un changement de code en sortie. Elle est donnée par la valeur du quantum « Q ».

D'une manière générale pour un CNA théorique, on exprime Q de la manière suivante :

$$Q = \frac{V_{Ref}}{2^n - 1}$$

Q : Le quantum (en Volt), pour un CAN réel, Q est donné dans la documentation du constructeur.

n : nombre de bits en sortie du convertisseur.

La valeur binaire de sortie s'obtient par la transformation de la partie entière de  $N_d$  en binaire :

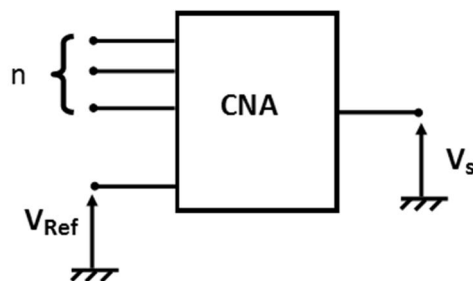
$$N_d = \frac{V_E}{Q}$$

- Son **temps de conversion** : lorsqu'on numérise un signal, on envoie au CAN un ordre de conversion, et on récupère la valeur binaire en sortie au terme d'un délai appelé temps de conversion.

- La **plage de conversion** : pour fonctionner, un CAN nécessite toujours une tension limite pour laquelle la tension d'entrée ne devrait pas dépasser. En dehors de cette zone, il ne fonctionnera plus correctement : sortie numérique incorrecte et des éventuelles dégradations.

#### b- CNA

Le convertisseur Numérique/Analogique est un dispositif électronique permettant d'obtenir en sortie une Tension dont la valeur est représentative du mot binaire présenté en entrée et est symbolisé par la figure suivante pour un CNA de n bits :



**Figure 1. 6 : Symbole du CNA.**



- n : nombre de bits en entrée
- $V_{Ref}$  : tension de référence
- $V_s$  : tension de sortie

Un CNA est caractérisé essentiellement par :

- La **tension pleine échelle** que peut prendre la sortie. Elle est obtenue en appliquant à l'entrée de CNA la valeur numérique la plus grande, c'est à dire lorsque tous les bits d'entrée sont au niveau logique haut. En théorie (CNA idéal) la tension pleine échelle est égale à la tension de référence mais dans la réalité (CNA réel), elle est plus souvent inférieure à la tension de référence. Elle est fixée à la fabrication de composant et est donnée dans la caractéristique technique du constructeur.

- La **résolution** d'un CNA ou pas de progression ou quantum est la plus petite variation en sortie et correspond au LSB. La valeur de sortie  $V_s$  va augmenter de  $Q$  lorsque  $N$  va augmenter de 1. Elle est exprimé par:

$$Q = \frac{V_{Ref}}{2^n - 1}$$

La tension de sortie est en fonction de la valeur binaire  $N$  appliquée à l'entrée associée à la valeur décimale  $(N)_{10}$ . On a :

$$V_s = (N)_{10} Q$$

- La **temps de conversion** qui sépare le début d'un cycle de conversion et la disponibilité de la tension résultante en sortie.

- La **précision** qui est l'Ecart maximal entre la valeur théorique attendue en sortie et la valeur réelle. En général, la précision vaut  $\pm \frac{1}{2} Q$ .

- *La tension de décalage* : C'est la tension ou courant qui existe en sortie, lorsqu'on applique 00.0 en entrée. Idéalement, le CNA doit afficher 0V mais en réalité on mesure une petite tension qui est due à l'erreur de décalage de l'ampli opérationnel en sortie.

### **I - 3 CARACTERISTIQUES D'UN SYSTEME EMBARQUE**

Un système embarqué requiert une multitude d'exigences.

#### **I - 3 - 1 Exigences fonctionnelles**

Les systèmes embarqués fonctionnent généralement en temps réel. Les opérations de ces systèmes sont faites en réponse à un événement extérieur. Les opérations doivent avoir lieu dans des délais précis et avant une échéance, afin de garantir le bon fonctionnement du système.

Selon l'application, on distingue deux types de systèmes embarqués. En premier, les systèmes temps réel stricts où le non-respect des échéances peut avoir des conséquences graves sur le fonctionnement du système ou sur son environnement. Ensuite, les systèmes temps réel où le non-respect des échéances ralentit le système sans conséquences graves.

Ces systèmes doivent garantir une très haute fiabilité, réagir en cas de panne de l'un de ses composants, et assurer les besoins de sécurité.

#### **I - 3 - 2 Exigences techniques**

Les systèmes embarqués doivent réagir dans des délais connus, garantis et interagir avec leur environnement à une vitesse qui est imposée par ce dernier. L'exécution des tâches se fait en temps réel. Les systèmes embarqués doivent en général être autonomes, c'est-à-dire capables de fournir des services sans intervention humaine.

Ses critères d'autonomie s'appliquent d'abord sur le démarrage du système du fait de l'utilisation de mémoire flash et manque de disque dur, et ensuite, sur le fonctionnement du système en entier. L'autonomie énergétique fait aussi partie des exigences techniques.

Les systèmes devraient garantir un minimum de services dans tous les cas en dépit des erreurs éventuelles d'exécution ou des défaillances possibles du matériel (étanche, choc, etc.).

### **I - 3 - 3 Exigences économiques**

Les systèmes embarqués doivent garantir la fonctionnalité et la qualité à un coût de construction acceptable (rapport coût/efficacité). Beaucoup de systèmes embarqués sont fabriqués en grande série et doivent avoir des prix de revient extrêmement faibles.

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.

Dans un marché concurrentiel, il convient d'avoir un système opérationnel le plus rapidement possible pour être le premier sur le marché.

### **I - 3 - 4 Contraintes**

Un système embarqué fait face à certains faits contraignant :

#### **a- Contrainte au niveau de la taille**

Avec un poids, de volume et d'un espace mémoire limité de l'ordre de quelques Mo maximum, l'espace est compté. Cependant, il y a une difficulté de faire cohabiter dans un faible volume l'électronique analogique, l'électronique numérique et les composantes RF (*Radiofréquence*) sans interférences. Ils font appel, très souvent, à l'informatique et aussi aux systèmes « temps réel ».

#### **b- Contrainte temporelle**

Dans le domaine Temporelle, le temps d'exécution des tâches est déterminé. Le temps réel exprime une qualité de service à fournir, pour certains et une nécessité de garantir des délais de réponse connus à l'avance, pour d'autres. Toutes les applications temps réel ont une caractéristique commune qui réside dans l'existence de contraintes temporelles dont il faut tenir compte. Ces contraintes peuvent prendre diverses formes (échéances, intervalles de temps, durée de validité, etc.) et s'appliquer à des objets variés. En effet, dans les applications temps réel, les données ont une durée de vie limitée et deviennent obsolètes après un certain temps. Les événements apparaissent à des instants particuliers et doivent être pris en compte au bout de délais connus à l'avance, ainsi les traitements (ou actions) ont souvent des instants de débuts, de fin et des durées d'exécutions fixés [5].

#### **c- Contrainte environnementale**

L'évolution du système n'est pas contrôlée dans le milieu extérieur. Il est exposé à des variations et d'autres contraintes environnementales susceptibles d'induire des défaillances. Des contraintes d'implémentation physique sont liées à la consommation de ressources et au contexte de déploiement. Par exemple, la température, les vibrations, les variations d'alimentation, les interférences RF, la corrosion, l'eau, le feu, les radiations, etc.

#### **d- Contrainte énergétique**

Le système doit consommer de l'énergie le plus faible possible par l'utilisation de sources autonomes, batteries, les panneaux solaires et la communication des matériels intégré au sein du système. Il est nécessaire d'aborder ce sujet du fait que c'est une des raisons de ce travail qui est la domotique.

### **I - 4 TACHES**

Une tâche est généralement caractérisée par un temps de calcul, une échéance qui est la date à laquelle la tâche doit avoir terminé son exécution. Et dans le cas des tâches

périodiques, elle est caractérisée par une période qui représente la durée séparant les instants d'activation. Une exécution de la tâche est appelée une instance [6].

#### **I - 4 - 1 Contraintes**

Les tâches accèdent à des ressources communes, s'échangent des messages via un réseau et s'exécutent sur un ou plusieurs processeurs. Ils doivent respecter différentes contraintes à savoir :

- la cohérence des ressources partagées
- la tolérance aux fautes : la puissance de calcul du système embarqué doit être suffisante afin de fournir les fonctionnalités pour lesquelles ce dernier a été conçu.
- la sécurité : la protection des données de l'utilisateur doit être assurée, grâce à la mise en œuvre de systèmes prédictibles.
- Et le temps de réponse : elle est directement liée à la dynamique du système.

#### **I - 4 - 2 Les différents types de tâches**

On distingue plusieurs types de tâches.

##### ***a-* Tâches d'entrées/sorties**

Ces tâches permettent d'accéder aux données externes par l'intermédiaire de cartes d'entrées/sorties et ensuite de capteurs et d'actionneurs directement liés au procédé géré. Elles peuvent être activées de façon régulière ou par interruption.

##### ***b-* Tâches de traitement**

Ces tâches constituent le cœur de l'application. Elles intègrent des traitements de signaux (analyse spectrale, corrélation, traitement d'images, etc.) ou des lois de commande (Régulation tout ou rien, régulation du premier ordre, régulation PID, etc.).

#### **c- Tâches de gestion de l'interface utilisateur**

Ces tâches permettent de présenter l'état du procédé ou de sa gestion à l'utilisateur. En réponse, l'opérateur peut modifier les consignes données ou changer les commandes. Elles peuvent être très complexes et coûteuses en temps de calcul si l'interface gérée est de taille importante (tableau de bord) ou de type graphique (représentation 3D).

#### **d- Tâches de communications**

Ces tâches sont destinées à gérer les messages envoyés ou reçus à travers un ou plusieurs réseaux ou bus de terrain. Si ce type de tâches existe, l'application est dite distribuée ou répartie.

#### **e- Tâches de sauvegarde**

Ces tâches permettent de stocker l'état du système à des instants fixés. Cette sauvegarde peut être utilisée a posteriori pour analyser le fonctionnement de l'application ou lors d'une reprise d'exécution à une étape précédente.

### **I - 4 - 3 Les relations entre les tâches**

Les tâches obtenues, qui constituent l'application de contrôle-commande, ne sont pas des entités d'exécution indépendantes. En effet certaines tâches sont connectées vers l'extérieur pour les entrées/sorties. De plus elles peuvent être liées par des relations de type.

#### **a- Synchronisation**

La synchronisation se traduit par une relation de précédence d'exécution entre les tâches. Une tâche peut se synchroniser avec une autre tâche (ou une ISR) avec un sémaphore. Dans ce cas un drapeau initialisé à 0 signale un événement. Par exemple, une tâche peut lancer une opération d'I/O et attendre le sémaphore. Quand l'opération d'I/O est terminée, une ISR (ou une autre tâche) fait un signal sur le sémaphore (incréméntation), et la tâche peut

reprandre. Si le noyau supporte les sémaphores compteur, il accumule les évènements non traités. Plus d'une tâche peut être en attente d'évènement, le noyau dans ce cas signale à la tâche de plus haute priorité en attente puis à la première tâche en attente de l'évènement.

#### ***b-* Communications**

Les tâches peuvent avoir besoin de communiquer avec une autre tâche par le transfert de données globales ou messages. Les données globales nécessitent un accès exclusif à la variable. Si une tâche est impliquée, il faut interdire les interruptions. Si deux tâches partagent des données, elles peuvent interdire puis autoriser les interruptions (accès exclusif) ou utiliser un sémaphore.

Une tâche peut uniquement communiquer des informations à une ISR via des variables globales, la tâche n'est pas mise au courant du changement par l'ISR sauf si l'ISR signale à la tâche par un sémaphore ou que la tâche effectue une scrutation périodique de la variable. Pour corriger ce problème, on utilise soit une boîte aux lettres de messages (mail box) soit une queue de message.

#### ***c-* Partage de ressources**

Les tâches utilisent des éléments mis en commun au niveau du système comme des zones mémoire, des cartes d'entrées/sorties, cartes réseau, etc. Certaines de ces ressources, comme par exemple les zones mémoire, ne sont pas ou ne doivent pas être accessibles, pour avoir un fonctionnement correct, par plus d'une tâche à la fois, elles sont dites ressources critiques.

### **I - 5 EXEMPLES D'APPLICATIONS DES SETR**

Traditionnellement, l'étude sur les systèmes embarqués se focalisait dans l'avionique, Robotique, Automobile et Militaire. Mais de nos jours, grâce à l'avancé de la technologie, ils sont de plus en plus répandu dans nos quotidiens.

### **I - 5 - 1 Aviation**

Dans le champ encore largement inexploré du développement durable, les systèmes embarqués facilitent la réduction de la consommation d'énergie dans les avions (« moteurs propres »). Ils sont partie prenante des technologies spatiales, sur les plates-formes satellitaires et au cœur des instruments d'observation et d'analyse de la terre et de l'espace etc.

### **I - 5 - 2 Automobile**

L'électronique et l'informatique embarquées dans les voitures constituent un réseau tellement complexe que de nouveaux outils informatiques sont nécessaires pour le tester et diagnostiquer les pannes ou télé diagnostique : l'allumage électronique, injection, système antiblocage, coussins gonflables de sécurité, correcteurs de trajectoire et d'autre encore. L'introduction massive de fonctions embarquée a provoqué ces dernières années une véritable révolution dans l'industrie automobile.

### **I - 5 - 3 Domotique**

Dans le domaine de la domotique, on citera des systèmes embarqués qui sont intégrés à des équipements afin d'apporter une surveillance intelligente d'une maison : un dispositif de surveillance qui permet de signaler un événement comme par exemple une intrusion. On peut gérer aussi l'énergie consommé dans un foyer pour plus d'économie possible. Contrôler le stock des aliments n'est plus à l'écart quand il s'agit de la domotique à l'heure actuelle et il y en a d'autre encore.

Un système embarqué est un système complexe qui intègre du logiciel et du matériel conçus ensemble afin de fournir des fonctionnalités données dont le secteur de la domotique. Dans le prochain chapitre, nous allons étudier une des applications au système embarqué dont la domotique.