

Jihen Thabet

Module:

Informatique embarquée

CHAPITRE 1:

INTRODUCTION AUX SYSTÈMES

EMBARQUÉS

DEFINITION D'UN SYSTÈME EMBARQUÉ

- ▶ Un système embarqué est défini comme un système **électronique** (matériel) et **informatique** (logiciel) autonome, spécialisé dans des tâches bien précises. Ses ressources sont généralement **limitées**.
- ▶ Système embarqué:
 - ▶ Comprend une partie matérielle et une partie logicielle
 - ▶ Architecture spécifique dédiée. Effectuer des tâches précises
 - ▶ Certains doivent répondre à des contraintes de temps réel. D'autres ayant peu de contraintes au niveau performances permettent de simplifier le système et de réduire les coûts de fabrication
 - ▶ En contact avec l'environnement
 - ▶ N'est pas toujours un module indépendant. Le plus souvent il est intégré dans le dispositif qu'il contrôle
 - ▶ Possède des entrées/sorties spécifiques et réactives
- ▶ Le logiciel créé pour les systèmes embarqués est appelé **firmware**. Il est stocké dans une mémoire en lecture seule (ROM) ou dans une mémoire flash. Il fonctionne le plus souvent avec des ressources matérielles limitées : un petit, voire pas de clavier, un petit écran et peu de mémoire.

DOMAINES D'UTILISATION (1)

3



DOMAINES D'UTILISATION (2)

- ▶ Domaine grand public:
 - ▶ smart phone, console de jeux, appareil photos, lecteur audio, décodeurs TV ...
- ▶ Moyens de transport:
 - ▶ gestion moteur, ordinateur de bord, ABS, GPS, système navigation, système d'aide automobiles, avions, trains, bateau, véhicule électrique, ...
- ▶ Equipement médicaux:
 - ▶ imagerie (rayon X, IRM), endoscopie, monitoring, lasers, chirurgie, stimulateur cardiaque, ...
- ▶ Equipement de télécommunications:
 - ▶ station mobile, routeur, gateway, satellite, ...
- ▶ Equipements industriels:
 - ▶ commande, contrôle réparti, capteurs intelligents, ...
- ▶ ...

SPÉCIFICITÉS D'UN SYSTÈME EMBARQUÉ (1)

- ▶ Caractéristiques spécifiques:
 - ▶ Dédié pour une application spécifique
 - ▶ Coût réduit (de production)
 - ▶ Espace restreint (volume, capacité mémoire)
 - ▶ Capacité de calcul appropriée et adaptée
 - ▶ Exécution temps réel (la plupart des systèmes)
 - ▶ Fiabilité et sûreté de fonctionnement
 - ▶ Consommation d'énergie maîtrisée, voir très faible en cas d'utilisation des batteries

SPÉCIFICITÉS D'UN SYSTÈME EMBARQUÉ (2)

- ▶ Différences avec un ordinateur de bureau :
 - ▶ L'interface IHM (Interface Homme Machine) est adaptée selon l'application. Cela peut-être de simples leds et boutons jusqu'à un écran tactile. Généralement pas de clavier, et l'écran est petit
 - ▶ Le système embarqué dispose de capteurs spécifiques pour son application: (température, courant, pression, module GSM-GPS, ...)

SPÉCIFICITÉS D'UN SYSTÈME EMBARQUÉ (3)

- ▶ Architecture d'un système embarqué :
 - ▶ Il est réalisé généralement autour d'un microcontrôleur
 - ▶ Un microcontrôleur est un système à processeur dans un seul chip avec de très bon rapport performance/prix
 - ▶ Comprend: processeur, mémoire, décodeur d'adresse, GPIO (entrées/sorties simples configurables) , contrôleur d'écran, contrôleur d'interruption, ...
 - ▶ Ensemble compact (volume optimisé)
 - ▶ Démarrage autonome du système (boot). Pas de disque dur, utilisation de mémoire flash
 - ▶ Généralement pas d'extension possible par l'ajout de cartes ou modules supplémentaires

DÉVELOPPEMENT DE SYSTÈMES EMBARQUÉS

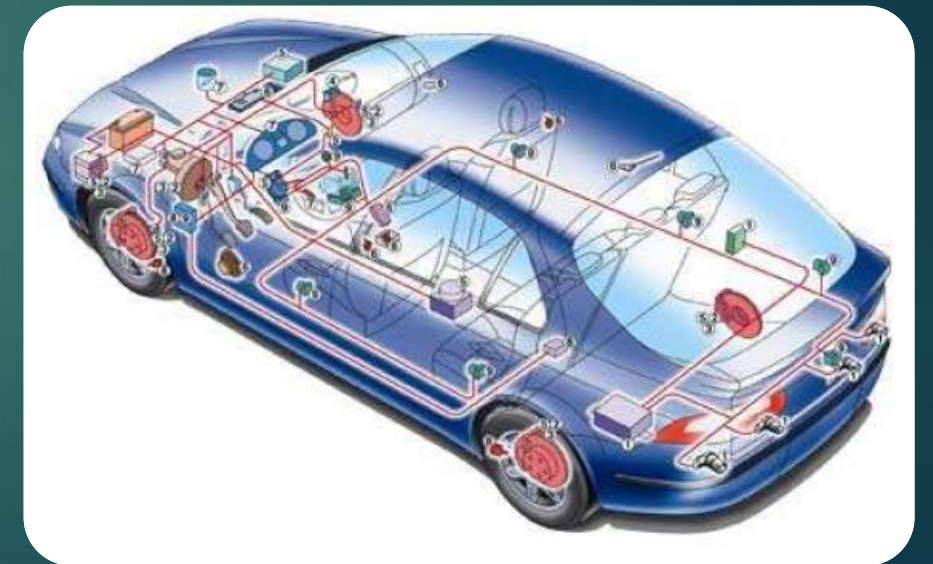
Le développement d'un système embarqué nécessite des connaissances à la fois en **électronique** et en **informatique**. Parmi le matériel nécessaire pour réaliser un système embarqué on trouve :

- ▶ La documentation sur les composants utilisés (datasheets)
- ▶ L'outillage de base de l'électronicien: fer à souder, ...
- ▶ Les outils d'analyse temporelle : oscilloscope, analyseur logique...
- ▶ Des composants de base: résistances, condensateurs, inductances,...
- ▶ Un microprocesseur ou un microcontrôleur
- ▶ Un compilateur croisé: pour compiler le noyau de l'OS, les applications, ...
- ▶ Un programmeur de microcontrôleur
- ▶ Un émulateur (*In Circuit Emulator*). Considéré comme l'équipement roi pour le debug matériel et logiciel
- ▶ Une sonde JTAG: pour la phase de développement

CARACTÉRISTIQUES

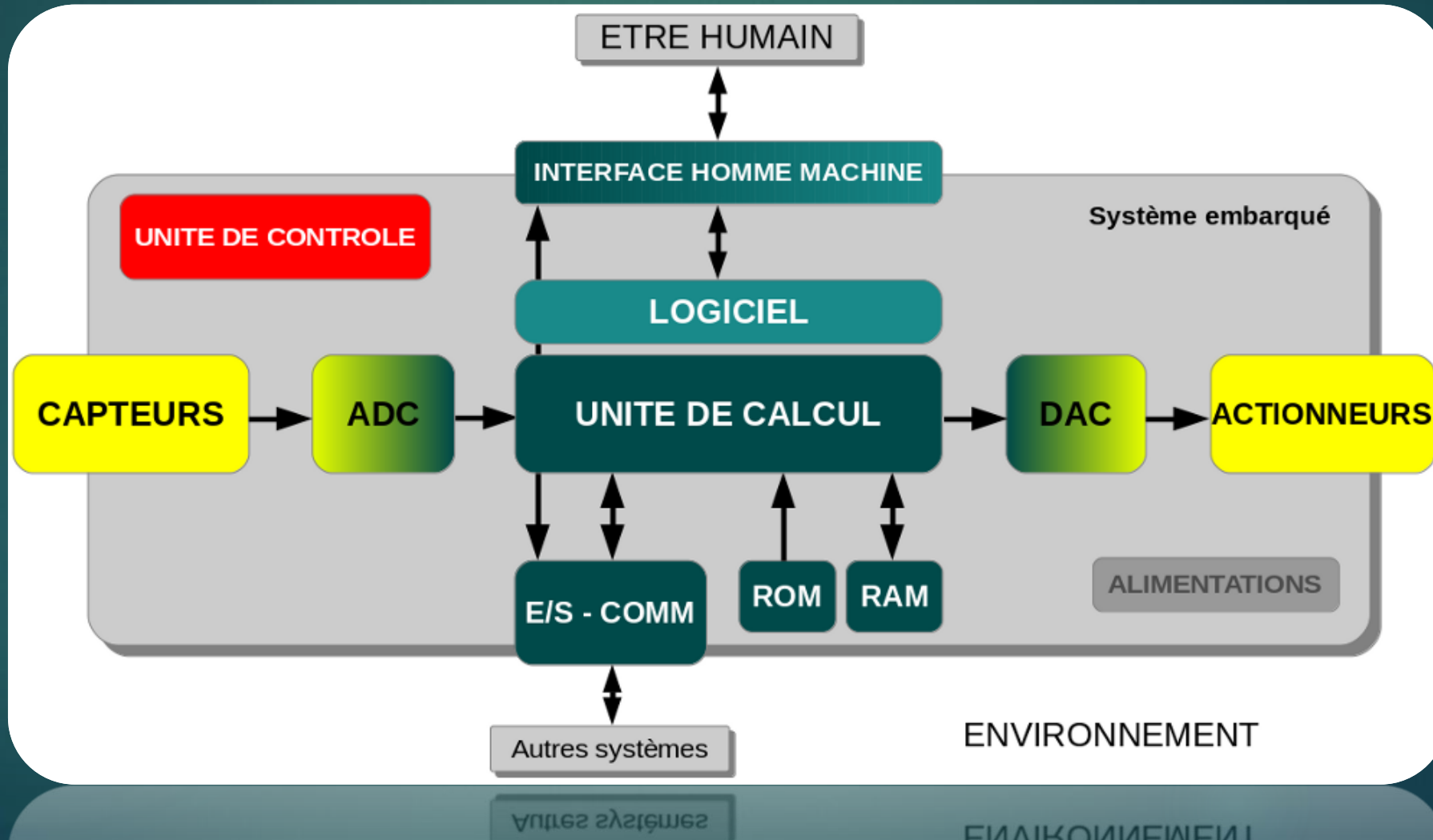
9

- ▶ Un système embarqué est principalement numérique pouvant intégrer une partie analogique (conditionnement de signaux, modulation, filtrage, ...)
- ▶ Système embarqué = une partie d'un système plus complexe
- ▶ Exemple: Dans automobile, entre 40 et 100 systèmes embarqués pour assurer :
 - ▶ La sécurité (ABS,...)
 - ▶ Le confort (auto-radio, ouverture centralisée...)



STRUCTURE D'UN SYSTÈME EMBARQUÉ

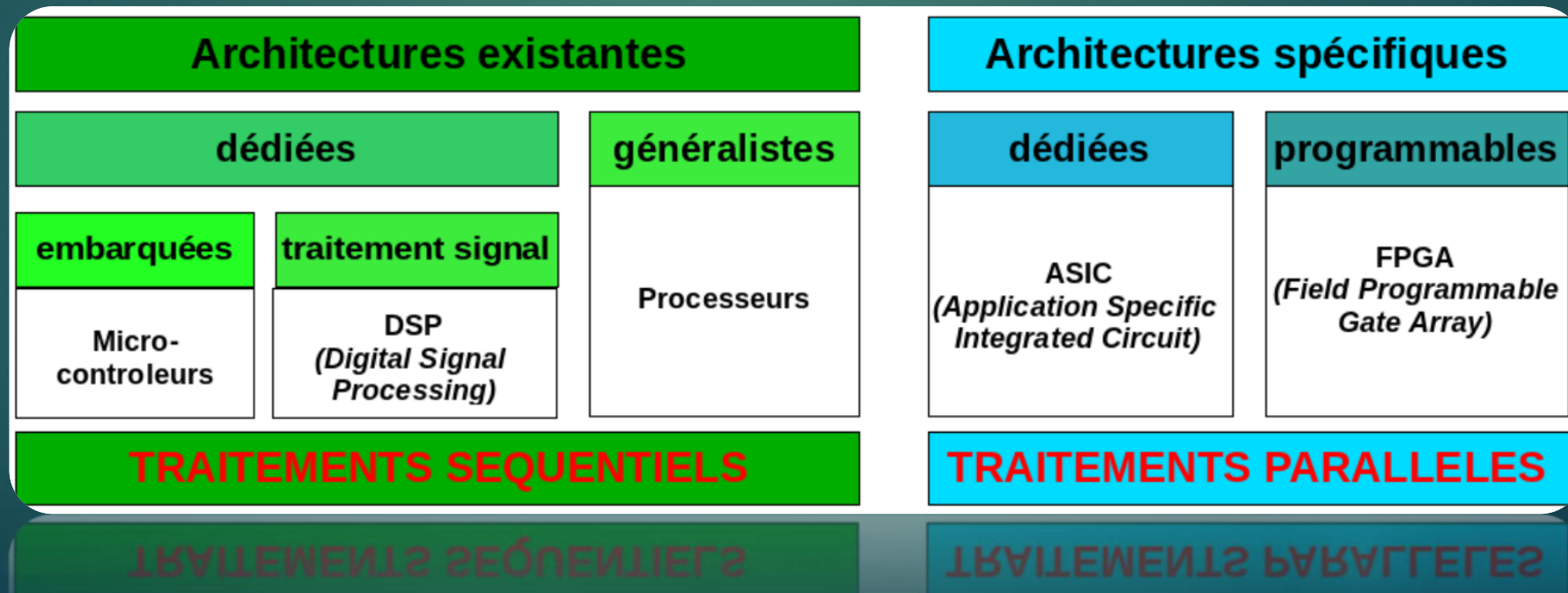
10



CONSTITUTION D'UN SYSTÈME EMBARQUÉ

11

- ▶ Pour concevoir un système embarqué "intelligent", deux solutions principales :
 - ▶ des architectures existantes (ex: microcontrôleurs)
 - ▶ des architectures spécifiques (FPGA, ASIC)



EVOLUTION DES PROCESSEURS

12

Date	Description	Transistors
1971	Intel 4004, CPU 4 bits, 108 KHz Adressage 640 Bytes, 0.06 Mips	2'300 10 μm
1979	Motorola 68000, CPU 32 bits, 8 MHz Adressage 16 MB, 0.7 Mips	68'000 3.5 μm
1990	Motorola 68040, CPU 32 bits, 50MHz, Adressage 4 GB, MMU, FPU, 44Mips	1.2 million 0.8 μm
1999	Pentium III, CPU 32 bits, 500MHz Adressage 4 GB, MMU, L1 cache 16KB	9.5 million 0.25 μm
2005	Pentium D, CPU 64 bits, Dual cores, 2.8GHz	290 million 0.09 μm
2008	Core i7, CPU 64 bits, 4 cores, 2.8GHz	781 million



EVOLUTION DES PROCESSEURS ARM

13

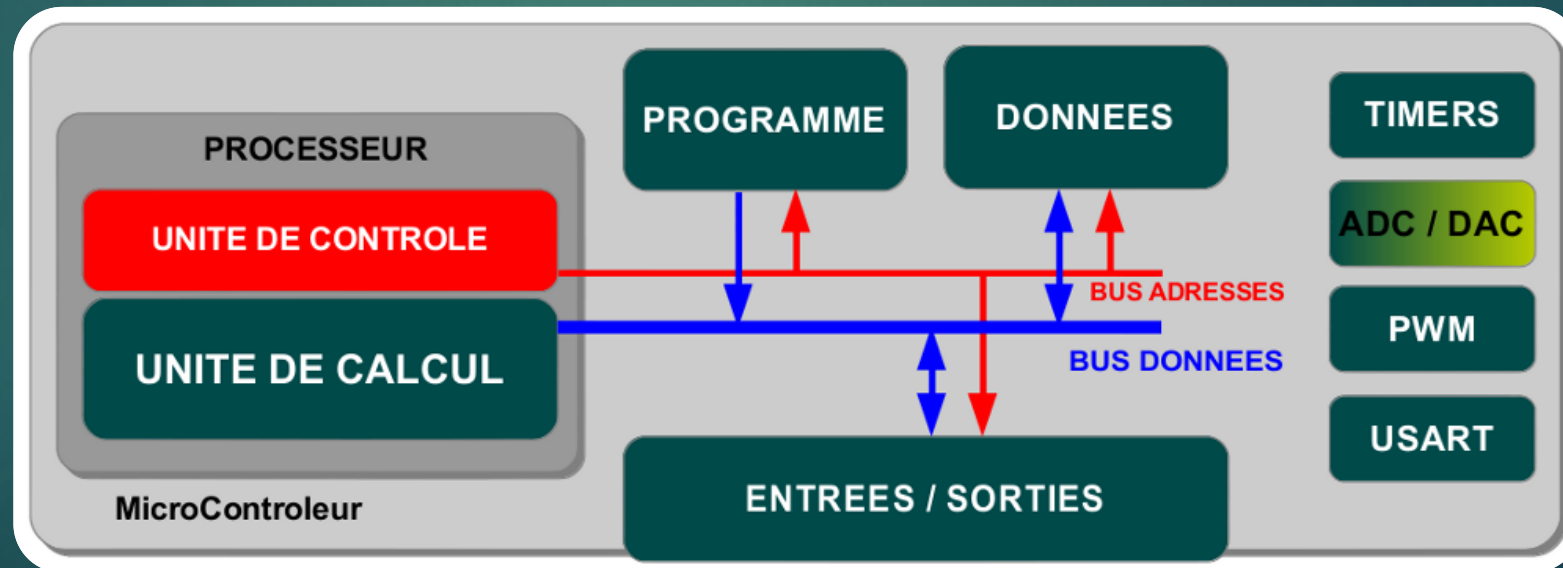
Processor	Architecture	features
ARM7TDMI	ARMv4T	
ARM920	ARMv4T	MMU
ARM926EJ-S	ARMv5E	MMU, DSP, Jazelle
ARM11 MPCore	ARMv6	MPCore, MMU, DSP, Jazelle
Cortex-M1	ARMv6-M	Intégration dans FPGA
Cortex-M3	ARMv7-M	MPU (MMU), NVIC
Cortex-A8	ARMv7-A	MMU, DSP, Jazelle, NEON + floating
ARM Cortex-A9	ARMv7-A	MPCore, MMU, DSP, Jazelle, NEON + floating

MMU: Memory Management Unit
MPCore: MultiProcessor Core
Jazelle: hardware Java Virtual Machine
NVIC: Nested Vectored Interrupt Controller

CONSTITUTION D'UN SYSTÈME EMBARQUÉ – MICROCONTROLEUR (1)

14

- ▶ Microcontrôleur
 - ▶ Unité de calcul séquentiel pré-cablée
 - ▶ Exécution d'une séquence d'instructions
 - ▶ Entrées/Sorties spécifiques (numériques et analogiques)



CONSTITUTION D'UN SYSTÈME EMBARQUÉ – MICROCONTROLEUR (2)

15

- ▶ Avantages:
 - ▶ Large choix de composants
 - ▶ Modules déjà existants (ADC, PWM...)
 - ▶ Gestion d'horloge – oscillateur interne de quelques MHz à quelques centaines de MHz
 - ▶ Facilité de mise en œuvre
- ▶ Inconvénients:
 - ▶ Exécution séquentielle des calculs. Moins rapide pour le traitement de données
 - ▶ Instructions prédéfinies
 - ▶ Utilisation réservée de certains modules (entrées-sorties limitées)

LES CIRCUITS LOGIQUES PROGRAMMABLES

16

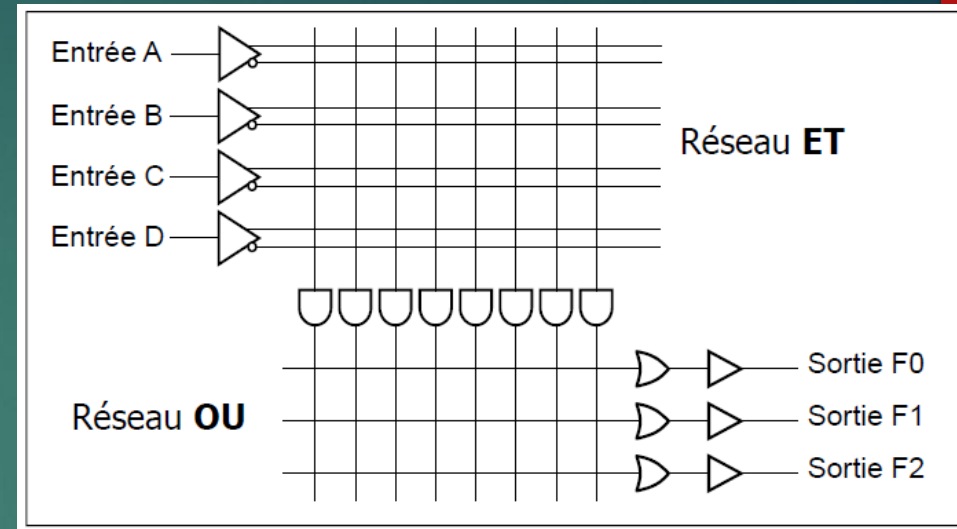
- ▶ Un **circuit logique programmable** est un circuit intégré logique qui peut être « reprogrammé » après sa fabrication
- ▶ Il serait impropre de parler de programmation au sens logiciel (contrairement à un microprocesseur, il n'exécute aucune ligne de code). Dans ce cas, il est mieux de parler de « **reconfiguration** » plutôt que de reprogrammation. On modifie des connexions ou le comportement du composant. Il s'agit bien de **réseaux logiques reconfigurables et modifiables**
- ▶ Un circuit logique programmable est composé de nombreuses **cellules logiques** élémentaires et **bascules logiques** librement connectables,
- ▶ Avantages:
 - ▶ Exécution parallèle des calculs
 - ▶ Gestion d'horloge avancée de quelques MHz à quelques GHz
 - ▶ Reconfiguration dynamique
- ▶ Inconvénients:
 - ▶ Aucun module pré-cablé
 - ▶ Entrées-sorties numériques seulement

STRUCTURE DES CIRCUITS LOGIQUES PROGRAMMABLES

17

Une fonction logique peut-être exprimée par une équation logique :

- **F = somme de produits**
 - Exemple : $F0 = (D \text{ and } B \text{ and } A)$ or $(\text{not}C \text{ and } A)$ or $(D \text{ and } \text{not}C)$
- ▶ Circuit universel :
 - **Un réseau de ET suivi d'un réseau de OU**



Famille	Réseau ET	Réseau OU	Circuits
FPLA Field Prog. Logic Array	Programmable	Programmable	GAL
PAL Prog. Array Logic	Programmable	Fixe	SPLD, CPLD
FPGA Field Prog. Gate Array	Programmable	Inexistant	FPGA
PLE Prog. Logic Element	Fixe	Programmable	PROM

APPLICATION SPECIFIC INTEGRATED CIRCUIT (ASIC)

18

- ▶ Un ASIC (*Application-Specific Integrated Circuit*) est un circuit intégré spécialisé. En général, il regroupe un grand nombre de fonctionnalités uniques ou sur mesure.
- ▶ L'intérêt de l'intégration est de réduire les coûts de production et d'augmenter la fiabilité.
 - ▶ Avantages:
 - ▶ un contrôle total du produit
 - ▶ un coût de production réduit
 - ▶ Inconvénients :
 - ▶ un coût de développement élevé (notamment pour la fabrication des masques de gravure)
 - ▶ un délai de développement de plusieurs mois
- ▶ On qualifie les gros ASIC de SoC (*System-on-Chip*, ou système sur silicium), lorsqu'ils intègrent des processeur(s), interfaces, mémoires, etc., totalisant plusieurs millions de portes logiques, et qu'ils assurent la quasi-totalité des fonctions de la carte.

CHOIX DE L'ARCHITECTURE

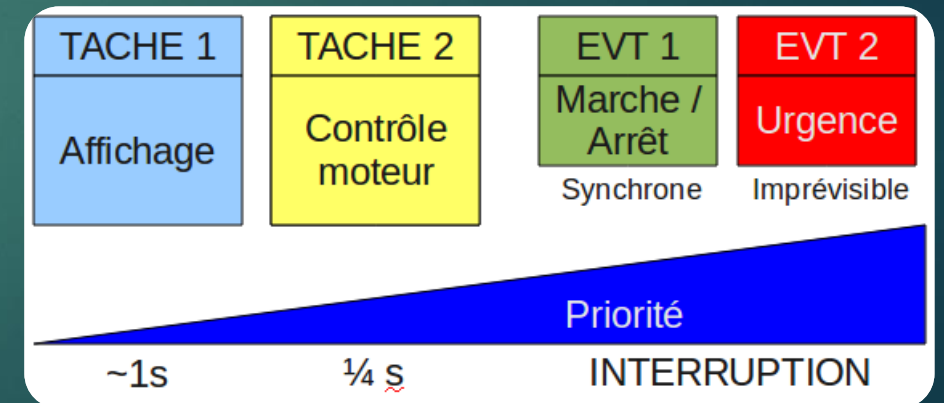
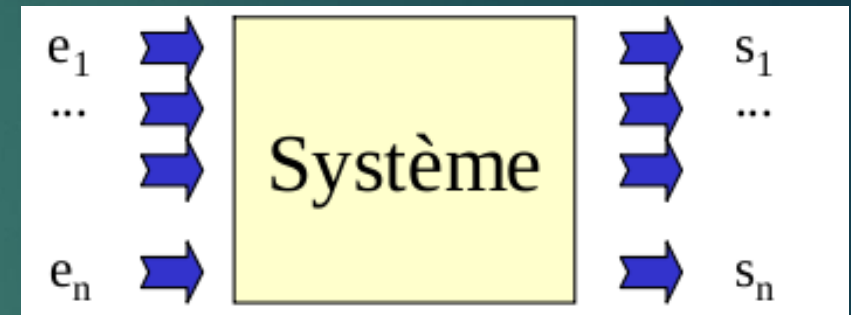
19

- ▶ Les critères à prendre en compte sont les suivants :
 - ▶ le nombre de fonctions à réaliser
 - ▶ le nombre d'entrées-sorties nécessaires
 - ▶ la vitesse d'exécution de ces fonctions
 - ▶ la consommation électrique
 - ▶ l'aspect temps réel et multitâches

NOTIONS IMPORTANTES

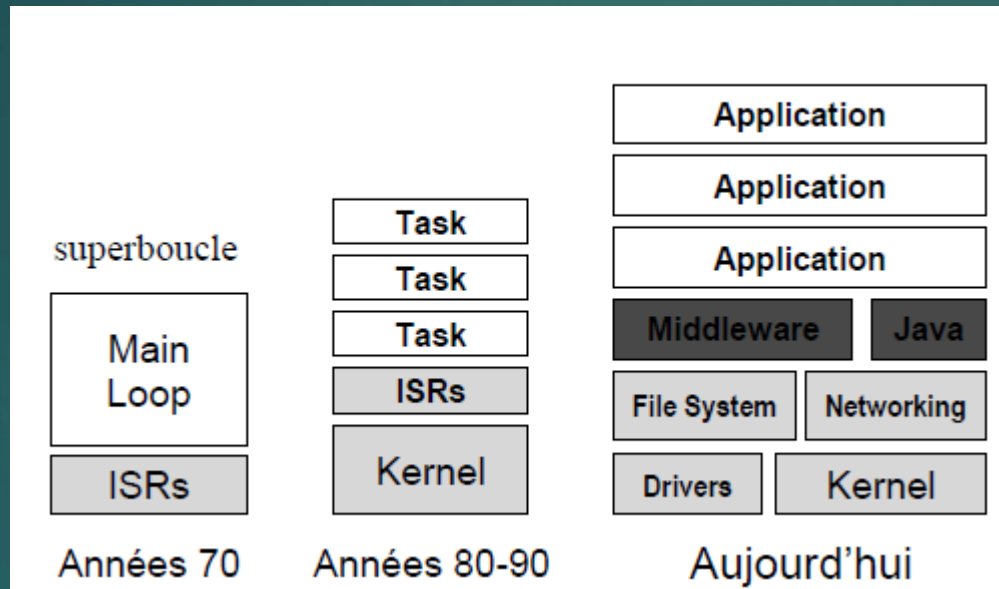
20

- ▶ Un système est caractérisé par :
 - ▶ sa relation entrées-sorties
 - ▶ son temps de réponse
- ▶ Gestion de tâches :
 - ▶ priorité
 - ▶ ordonnancement
- ▶ Gestion d'évènements (indiquant l'évolution d'un système) :
 - ▶ synchrones
 - ▶ asynchrones (non prédictibles) - interruptions
- ▶ Systèmes multitâches / systèmes temps réel

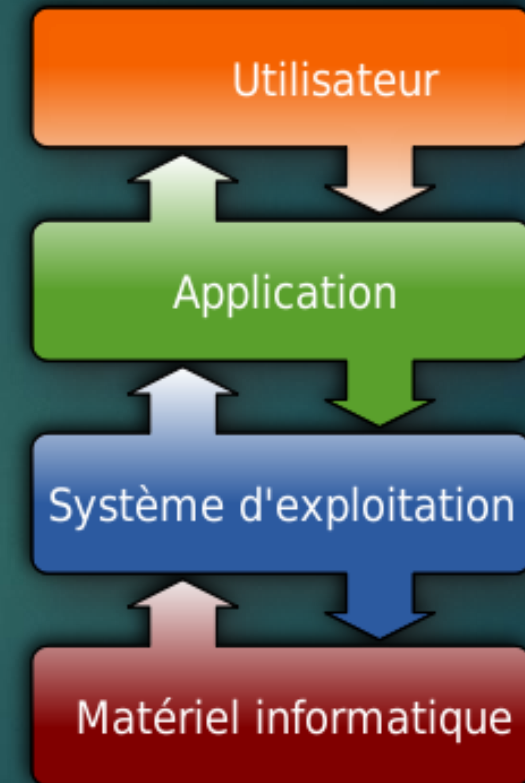


LES RTOS ET LES SYSTÈMES EMBARQUÉS

21



- ▶ Un système d'exploitation temps réel :
 - ▶ Apport du multitâche
 - ▶ Maîtrise des contraintes temporelles
 - ▶ Développement de pilotes des périphériques
 - ▶ Système de fichiers



PLAN DE COURS (1)

22

- ▶ **Chapitre 1 : Introduction aux systèmes embarqués**
- ▶ **Chapitre 2 : Les processeurs**
 - ▶ Architecture: Harvard / Von Neuman, Unités de calcul, Unité de contrôle,
 - ▶ Séquenceur, Registres, ...
 - ▶ Composition & classification des processeurs
 - ▶ Les opérations du processeur, Parallélisme, Adressage
 - ▶ Les processeurs spécialisés (DSP, GPU,...)
- ▶ **Chapitre 3 : Les processeurs ARM**
 - ▶ Présentation, Architecture & Technologies des processeurs ARM
 - ▶ Processeurs graphiques ARM
 - ▶ Divers processeurs ARM

PLAN DE COURS (2)

23

▶ **Chapitre 4 : Les microcontrôleurs**

- ▶ Architecture interne
- ▶ Les microcontrôleurs à base d'ARM (exemple STM32)
- ▶ Environnements de programmation
- ▶ Familles de microcontrôleurs

▶ **Chapitre 5 : La programmation C/ C++ appliquée aux systèmes embarqués**

- ▶ Rappel sur le langage C/C++
- ▶ La programmation C/C++ appliquée aux microcontrôleurs
- ▶ Les « Best Practices »