

Introduction aux systèmes embarqués

EMB7000

Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Introduction aux systèmes embarqués

EMB7000, Automne 2017

Professeur : Mounir Boukadoum

Bureau : PK-4540

Téléphone : 514-987-3000 #4565

Courriel: boukadoum.mounir@uqam.ca

Site du cours : www.labunix.uqam.ca/~boukadoum_m

Auxiliaire: Ouajdi Brini

Lundi, de 17h30 à 20h30, local SH-3540 (cours)

Lundi, de 13h30 à 16h30, local PK-4765, PK-4770 (laboratoires)

Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Objectifs, moyens et prérequis

- Introduire l'étudiant aux systèmes embarqués et leurs applications dans les systèmes de mesure et de commande en temps-réel
 - L'enseignement couvre les composants et les différentes étapes de conception et réalisation d'un système à microcontrôleur(s), du point de vue matériel et logiciel
- Le cours magistral est complété par des expériences de laboratoire pour la pratique
- Bagage souhaité : Connaissances de base en électronique analogique et numérique, langages c et assembleur
- Les discussions en séance sont encouragées!

Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Plan

1. Introduction
2. Familles de MCU : MSP430 et ARM
3. Périphériques
4. Convertisseurs de format
5. Capteurs
6. Prétraitement
7. Actionneurs et interfaces de puissance
8. Commande par MCU
9. Circuits d'alimentation
10. Exécutifs logiciels

Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Laboratoires

1. Environnements de développement *Code Composer Studio* de Texas Instrument ou *PSOC Creator* de Cypress
2. Ports d'E/S externes : claviers et afficheurs
3. Ports d'E/S internes : I2C
4. Conversion de format : générateur de signaux
5. Traitement du signal : filtres numériques
6. Actionneurs et interfaces de puissance : régulateur PID
7. Conception de circuits imprimés : régulateur de tension
8. Exécutifs logiciels : RTOS
9. Capteur sans fil à lien Bluetooth

Traduit et adapté de *Embedded Systems Design*, (c) 2000 Vahid/Givargis, et autres sources

Chapitre 1 : Introduction

Traduit et adapté de *Embedded Systems Design*, (c) 2000 Vahid/Givargis, et autres sources

Objectifs d'apprentissage

- Survol des systèmes embarqués
- Le défi du design : optimisation de métriques opérationnelles souvent contradictoires
- Les technologies de conception et de réalisation
 - Processeurs/Circuit intégrés
 - Outils de conception

Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

C'est le règne des ordinateurs, mais lesquels?

- Nous pensons surtout aux ordinateurs de "bureau"
 - PC 
 - Portables 
 - Serveurs 
- Un type beaucoup plus répandu existe!
 - Téléphones portables, appareils photos numériques, consoles de jeux, imprimantes
 - Systèmes de commande et capteurs sans fil

Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Système embarqué

- Difficile à définir précisément
Système à micro-ordinateur « enfoui »
- Pratiquement tout ce qui n'est pas PC
 - Compact (haut niveau d'intégration)
 - Faibles dimensions
 - OS invisible, réduit ou non existant
 - E/S dédiées
 - Fonctions prédéfinies
- $>10^{12}$ unités/an versus $<10^7$ PC !
 - Environ, 50+ par maison et automobile

On les trouve ici...



ici...



ici...



et même ici...



Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Une "petite" liste

Anti-lock brakes	Modems
Auto-focus cameras	MPEG decoders
Automatic teller machines	Network cards
Automatic toll systems	Network switches/routers
Automatic transmission	On-board navigation
Avionic systems	Pagers
Battery chargers	Photocopiers
Camcorders	Point-of-sale systems
Cell phones	Portable video games
Cell-phone base stations	Printers
Cordless phones	Satellite phones
Cruise control	Scanners
Curbside check-in systems	Smart ovens/dishwashers
Digital cameras	Speech recognizers
Disk drives	Stereo systems
Electronic card readers	Teleconferencing systems
Electronic instruments	Televisions
Electronic toys/games	Temperature controllers
Factory control	Theft tracking systems
Fax machines	TV set-top boxes
Fingerprint identifiers	VCR's, DVD players
Home security systems	Video game consoles
Life-support systems	Video phones
Medical testing systems	Washers and dryers



et beaucoup plus encore !

Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Un meneur



Apple iPhone 4 - Front

- Skyworks SKY77381 GSM/GPRS Front End Module
- Triquint TQM66692 Power Amp
- Skyworks SKY77422 Wi-CDMA FEM
- Triquint TQM67031 Power Amp
- Apple D380028 Infineon GSM/WCDMA Transceiver
- Skyworks SKY77420 3x1.5x1.5mm for Quad-Band GSM / GPRS / EDGE
- Apple A501 STMicro 3-axis digital gyroscope
- Apple A4 Processor
- Broadcom BCM4329KJRG 802.11n with Bluetooth 2.1 + EDR and FM receiver
- Broadcom BCM43750UB0 single-chip GPS receiver



TECHINSIGHTS



Apple iPhone 4 - Back

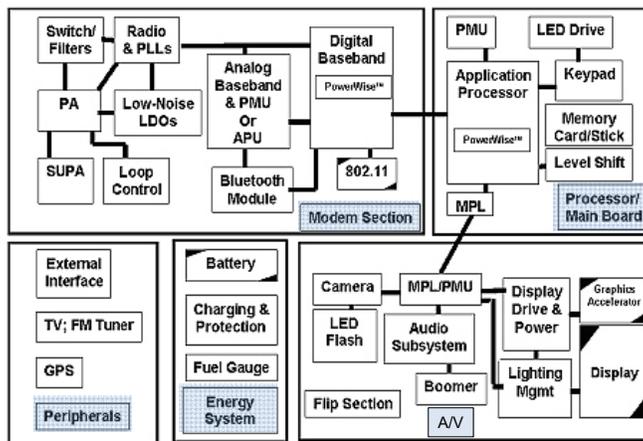
- Apple S4301490 - Texas Instruments Touchscreen controller
- The One App 03050500 audio codec
- Samsung K9PGC08UM 2560 DR. 48 FLASH MEMORY
- 3305007 Shing (Die mark D1816) (Naher) Power Management Unit
- 3383 Infineon X-GOLD 616 Baseband Processor
- Intel 3MAYEF - ELPIDA 128 Mbits Mobile DDR SDRAM 4 25F125F IntelNumalink NOR



TECHINSIGHTS

Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Architecture de téléphone intelligent



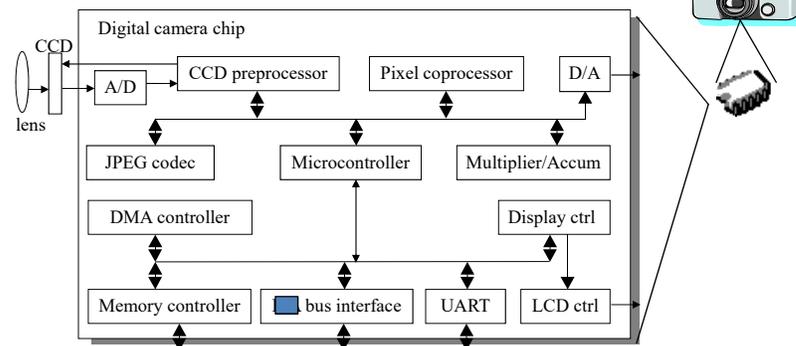
Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Caractéristiques communes

- Composants et fonctions intégrés
- Contraintes serrées
 - Faible consommation d'énergie
 - Petites dimensions physiques,
 - Temps de réponse optimal, etc.
- Réactif et opérant en temps-réel ou proche
 - Réagit continuellement aux conditions d'opération
 - Répond en dedans de contraintes de temps

Traduit et adapté de *Embedded Systems Design*, (c) 2000 Vahid/Givargis, et autres sources

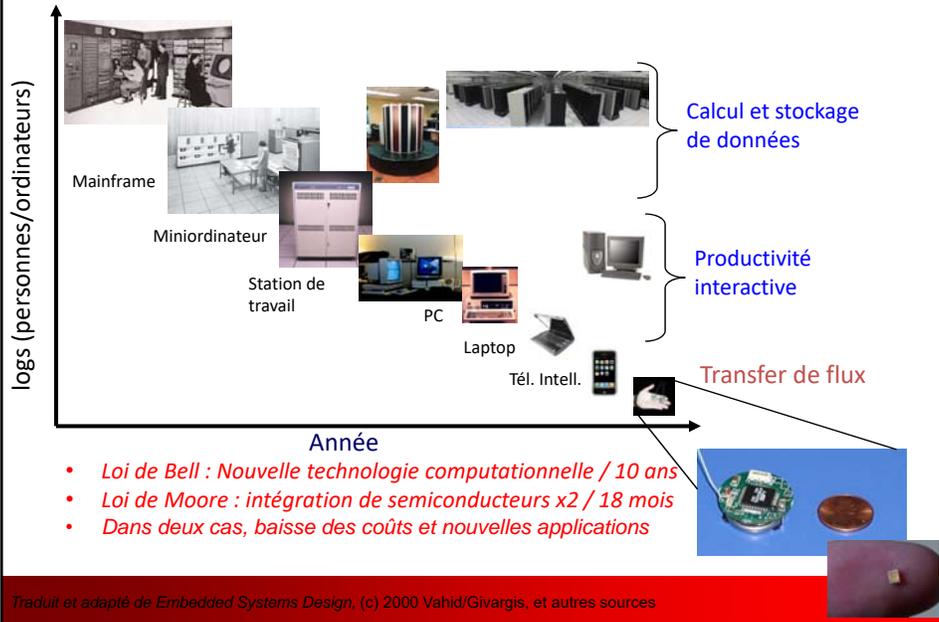
Exemple de système embarqué : Caméra numérique



- Fonction dédiée : Oui, toujours une caméra
- Contraintes serrées : oui, abordable, opération à piles, petites dimensions, réponse rapide
- Réactif et opération en temps réel : jusqu'à un certain point

Traduit et adapté de *Embedded Systems Design*, (c) 2000 Vahid/Givargis, et autres sources

Comment est-ce possible?



Développement d'un système embarqué

- Il faut satisfaire plusieurs contraintes simultanées!
 - But difficile, car les contraintes sont généralement mutuellement exclusives
 - Ex. petites dimensions physiques, grande puissance de calcul, faible émission de chaleur et grande autonomie énergétique
- Problème d'optimisation multiple portant sur un ensemble de métriques de design (*traits mesurables à l'implémentation*)

Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Métriques de conception communes

- **Coût unitaire** : coût de manufacturer chaque unité en excluant les charges non récurrentes
- « **Non-Recurring Engineering cost** » (NRE) : coût de conception initial
- **Taille**: dimension physique du système
- **Performance** : temps de réponse du système
- **Énergie consommée** : alimentation du système
- **Flexibilité**: capacité de modifier le système à coût raisonnable (NRE faible)
- **Fiabilité et sécurité d'utilisation**

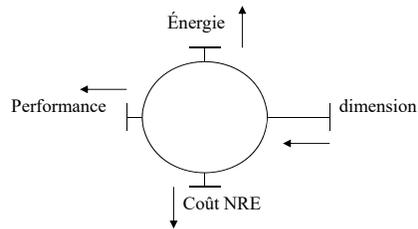
Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Il existe aussi des métriques de marché

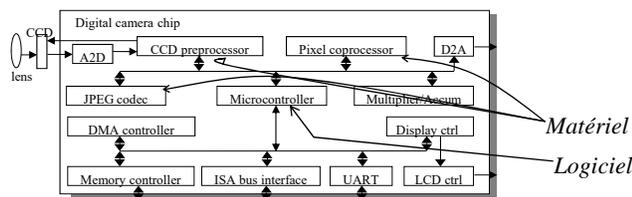
- **Temps de prototypage**: temps requis pour monter une version opérationnelle du système
- **Temps de mise en marché** : temps requis pour développer une version commercialisable.
- **Maintenabilité**: capacité de modifier le système après la mise en marché
- **Durabilité, sources d'approvisionnement, etc.**

Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Améliorer une métrique peut avoir un effet négatif sur une autre !

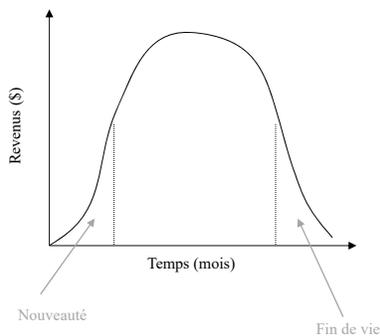


- L'expertise simultanée en *logiciel et matériel* est souvent requise
 - Codesign



Traduit et adapté de *Embedded Systems Design*, (c) 2000 Vahid/Givargis, et autres sources

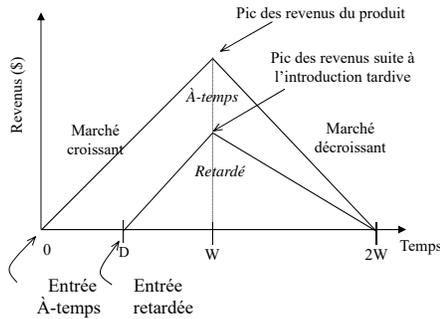
Une métrique exigeante : Le temps de mise en marché



- Temps requis pour développer une version commercialisable
- Doit rencontrer la fenêtre d'opportunité
 - Période durant laquelle le produit réaliserait le plus gros volume de ventes
- Durée moyenne de 8 mois et moins aujourd'hui!
- Les délais peuvent coûter très cher !

Traduit et adapté de *Embedded Systems Design*, (c) 2000 Vahid/Givargis, et autres sources

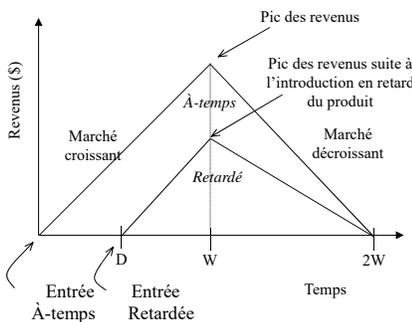
Pertes dues à une mise en marché tardive



- **Modèle simplifié des revenus**
 - Durée de vie du produit = $2W$, sommet des ventes atteint à W
 - Le temps de mise en marché définit un triangle représentant la pénétration du marché
 - L'aire du triangle représente les revenus
- **Perte due au retard**
 - La différence de surface entre les triangles *À-Temps* et *Retardé*

Traduit et adapté de *Embedded Systems Design*, (c) 2000 Vahid/Givargis, et autres sources

Pertes due à une mise en marché tardive



- **Aire du triangle** (en supposant $y=x$)
 - *À-Temps* = $0.5 * 2W * W$
 - *Retardé* = $0.5 * (2W - D) * (W - D)$
 - **Pourcentage de perte de revenu** = $(D(3W - D) / 2W^2) * 100\%$
 - **Exemples**
 - Durée de vie d'un an ($2W=1$) et délais de 0, 4 et 10 semaines
 - $D = 0$: $(0 * (3 * 26) / 2 * 26^2) = 0\%$
 - $D = 4$: $(4 * (3 * 26 - 4) / 2 * 26^2) = 22\%$
 - $D = 10$: $(10 * (3 * 26 - 10) / 2 * 26^2) = 50\%$
- Les délais coûtent cher !**

Traduit et adapté de *Embedded Systems Design*, (c) 2000 Vahid/Givargis, et autres sources

NRE, coût unitaire et coût total

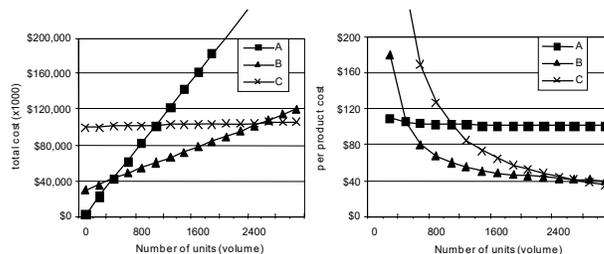
- Coûts:
 - Coût unitaire (CU): le coût de manufacturer chaque unité du systèmes, à l'exclusion des coûts non récurrents (NRE)
 - NRE (Non-Recurring Engineering cost): Le coût de conception du système
 - $\text{Coût total} = \text{NRE} + \text{CU} * \text{\#d'unités}$
 - $\text{Coût total par unité} = \text{coût total} / \text{\#d'unités}$
 $= (\text{NRE} / \text{\#d'unités}) + \text{coût unitaire}$
- NRE peut reléguer CU au second plan
 - NRE=\$20 000, coût unitaire=\$100
 - Pour 10 unités
 - Coût total = \$20 000 + 10*\$100 = \$21 000
 - Coût total par unité = $\underbrace{\$20\,000/10}_{\$2000} + \$100 = \2100

L'amortissement du NRE sur toutes les unités ajoute un coût \$2000 par unité!

Traduit et adapté de *Embedded Systems Design*, (c) 2000 Vahid/Givargis, et autres sources

NRE et coût unitaire

- Comparaison de technologies – c'est la quantité qui compte!
 - Technologie A: NRE=\$2,000, coût unitaire=\$100
 - Technologie B: NRE=\$30,000, coût unitaire = \$30
 - Technologie C: NRE=\$100,000, coût unitaire = \$2



- Il faut aussi considérer le temps de mise en marché !

Traduit et adapté de *Embedded Systems Design*, (c) 2000 Vahid/Givargis, et autres sources

Les métriques de performance*

- **Largement utilisées, souvent mal**
 - Fréquence d'horloge, # instructions par seconde : mesures souvent inefficaces!
 - Par exemple, un usager se préoccupe plus du temps de réponse d'une caméra que de la fréquence du processeur.
- **Latence (temps de réponse)**
 - Temps entre le début et la fin d'une tâche (par ex. : 0.25 s par prise de vue pour une caméra x)
- **Rapidité**
 - Tâches exécutées par seconde (ex. Camera A traite 4 images/s)
 - Peut être plus efficace que la latence si parallélisme de tâches
 - Par ex. : Camera B peut capturer 8 images/s en capturant une nouvelle image pendant qu'elle sauve la précédente).

Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

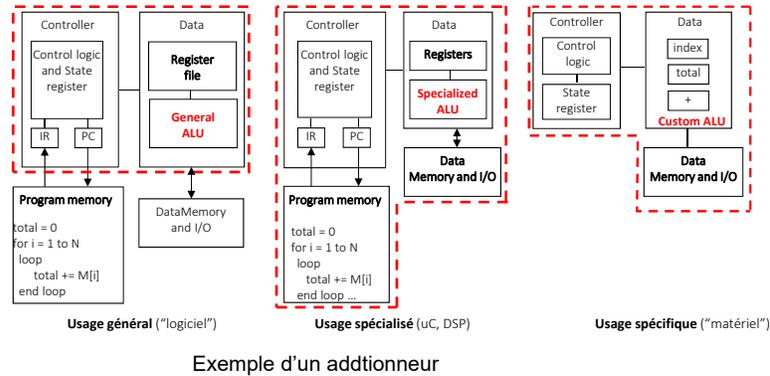
Aspect technologiques

- Technologie : façon d'accomplir une tâche en utilisant un ensemble de processus, méthodes, connaissances techniques
- **Trois aspects à considérer pour un système embarqué**
 - Architecture de traitement
 - Technologie d'implémentation
 - Architecture de design

Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

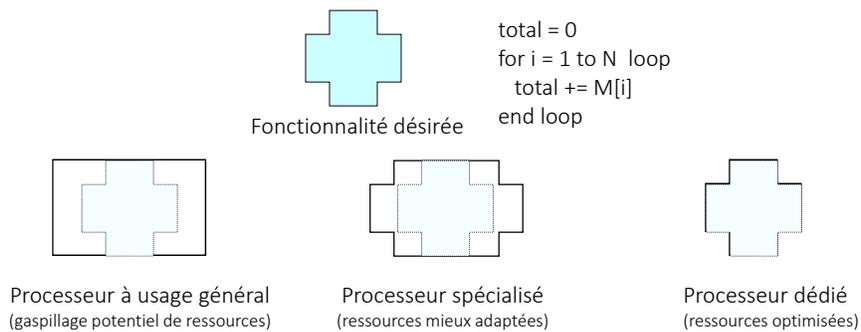
Processeurs

- Trois types d'unités de calcul
 - Usage général, spécialisé ou dédié
 - Programmables ou préprogrammés



Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Quel processeur utiliser?

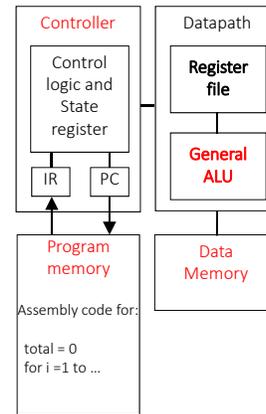


- Le choix se fait en fonction des contraintes de design, dont le coût

Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Processeur à usage général

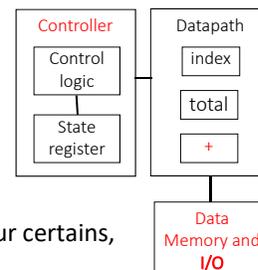
- Souvent appelé microprocesseur
- Reconfigurable
- Traits
 - Mémoire de programme
 - Grand nombre et variété de registres
 - UCT à usage général
- Bénéfices pour l'utilisateur
 - Temps de mise en marché et NRE **bas**
 - Grande **flexibilité**
- Inconvénients
 - Consommation et effets thermiques élevés
 - Interfaces complexes
- Les familles de processeurs d'Intel (pentium) et AMD (Ryzen) les plus connues



Traduit et adapté de *Embedded Systems Design*, (c) 2000 Vahid/Givargis, et autres sources

Processeur dédié

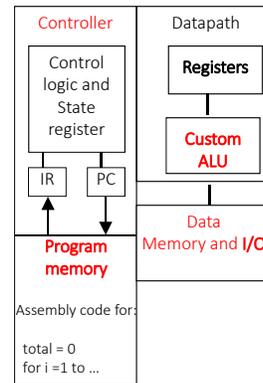
- Coprocesseur, accélérateur ou périphérique
- Souvent associé avec un autre type
- Traits
 - Contient uniquement les composants requis pour l'application
 - *Non programmable au sens habituel*
- Bénéfices pour l'utilisateur
 - Rapidité, consommation d'énergie optimisée pour certains, dimensions physiques réduites
- Inconvénients
 - Temps de mise en marché et NRE **moyens à élevés**
 - **Rigidité**
- Ex. ASIC, FPGA



Traduit et adapté de *Embedded Systems Design*, (c) 2000 Vahid/Givargis, et autres sources

Processeur à usage spécialisé

- Reprogrammable et optimisé pour une classe d'applications
 - Compromis entre l'usage général et l'usage dédié ; le uC et les DSP sont des exemples
- Traits
 - Mémoire de programme intégrée
 - Circuits de données optimisés
 - Unités fonctionnelles spécialisées
- Bénéfices pour l'utilisateur
 - Un peu de flexibilité, bonnes performances, dimension et consommation d'énergie réduites
 - temps de mise en marché **moyen**
- Inconvénients
 - NRE **faible à élevé**



Traduit et adapté de *Embedded Systems Design*, (c) 2000 Vahid/Givargis, et autres sources

Autre facteurs : jeu d'instructions

- CISC (*complex instruction set*)
 - Grand nombre d'instructions
 - Opérations de haut niveau en microcode (simplifient l'usage de langages de programmation évolués comme c++, java...)
 - Ex.: la majorité des processeurs pour ordinateurs de bureau (Pentium, Ryzen...)
- RISC (*reduced instruction set*)
 - Nombre réduit d'instructions
 - Opérations simples uniquement
 - Pas de microcode
 - Ex. : ARM, PowerPC, MIPS, uC récents

	CISC (Intel 486)	RISC (MIPS R4000)
#instructions	235	94
Addr. modes	11	1
Inst. Size (bytes)	1-12	4
GP registers	8	32

Traduit et adapté de *Embedded Systems Design*, (c) 2000 Vahid/Givargis, et autres sources

Architectures computationnelles

- Accumulateur
 - Un argument de chaque instruction vient de l'accumulateur
 - Facilite la conception de compilateurs c mais peu efficace pour la rapidité
- Registre-Mémoire
 - Les arguments des instructions peuvent avoir des origines diverses (registres internes, mémoire, ports d'entrée-sortie)
 - Architecture courantes dans les machines CISC
- « Load-Store »
 - Les arguments des instructions viennent de registres internes; des instructions de chargement (load) ou d'emmagasinage (store) interagissent avec la mémoire
 - Architecture courante dans les machines RISC

Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

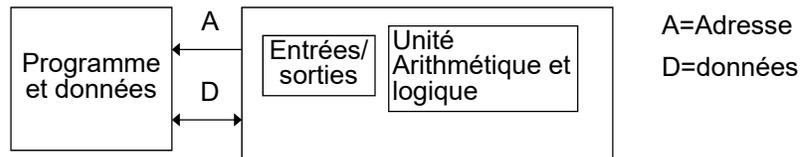
Particularités des machines RISC

- Seules les instructions les plus utilisées et sans grande complexité de réalisation existent en matériel.
- Quatre règles de conception majeures :
 - Instructions : Exécution en un cycle d'horloge/longueur fixe
 - Registres : grand nombre/à usage général
 - Architecture Load/store : La manipulation de données se fait seulement dans les registres ; des instructions de chargement/emmagasinage assurent le transfert vers/de la mémoire.
 - Pipeline : décodage en une étape/pas de microcode
- Avantages : design relativement simple qui permet une fréquence d'horloge optimale et une faible consommation d'énergie

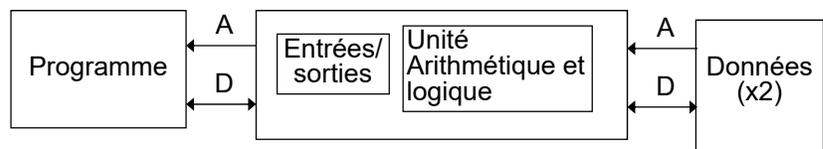
Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Autre facteurs : accès à la mémoire

Machine de Von Neuman



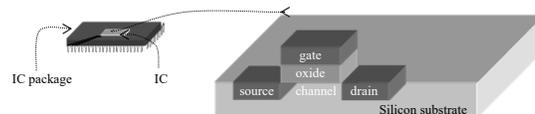
Architecture de Harvard (aussi due à Von Newman!)



Traduit et adapté de *Embedded Systems Design*, (c) 2000 Vahid/Givargis, et autres sources

Autre facteurs : approche technologique

- La manière dont un design peut être réalisé dans un CI
- Concept distinct de l'architecture des processeurs!
- Se distingue par
 - La méthode de fabrication (matériaux, couches, largeur de traits, etc.)
 - La généralité des CI obtenus



Traduit et adapté de *Embedded Systems Design*, (c) 2000 Vahid/Givargis, et autres sources

Technologie de réalisation

- Trois types
 - CI complètement dédié (ASIC/VLSI)
 - ASIC semi dédié (« gate array » et « standard cell »)
 - PLD (« Programmable Logic Device »)

Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

ASIC/VLSI

- Toutes les couches sont optimisées pour la fonctionnalité du système considéré
 - Dimension et placement des transistors
 - Largeur et routage des liens
- Bénéfices
 - Excellente performance, grande densité, petites dimensions, consommation en courant optimale
- Limitations
 - NRE élevé (\$1 M+), temps de mise en marché élevé

Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Semi-dédié

- Les couches inférieures sont partiellement ou complètement réalisées
 - Il reste à faire le routage des fils et éventuellement ajouter des blocs fonctionnels additionnels
- Bénéfices
 - « ASIC du pauvre » avec NRE plus petit (aussi bas que \$10k)
- Limitations
 - Demande des semaines à des mois de temps de développement

Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

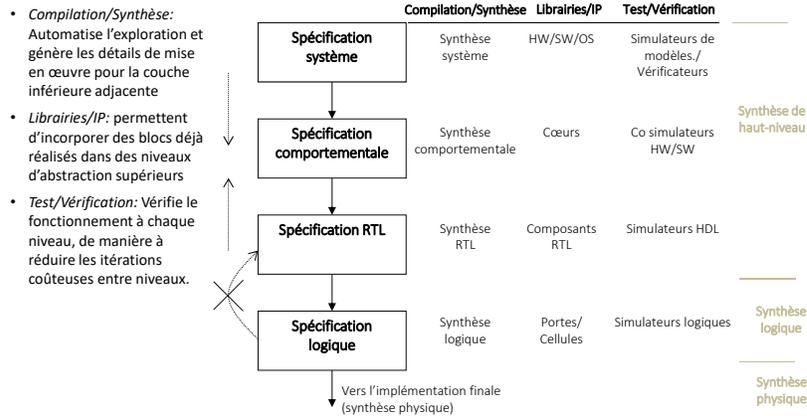
PLD

- Les couches sont déjà réalisées mais les composants sont à configurer et interconnecter
- Des liens sont faits ou défaits pour réaliser la fonctionnalité désirée
 - « Field-Programmable Gate Array (FPGA) » très populaire
- Bénéfices
 - Disponibles « on the shelf »
 - NRE bas, produit final disponible rapidement
- Limitations
 - Dimensions physiques et consommation d'énergie plus grandes, potentiellement plus lents ou plus coûteux à l'unité

Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Cycle de design

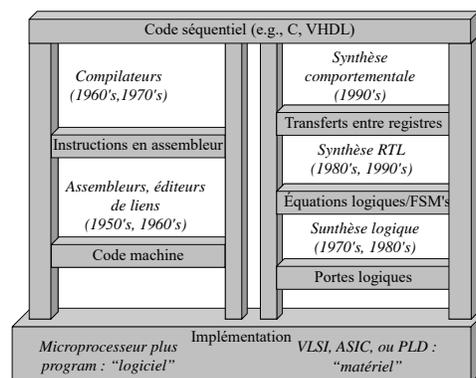
- Le processus pour convertir une fonctionnalité désirée en design concret



Traduit et adapté de *Embedded Systems Design*, (c) 2000 Vahid/Givargis, et autres sources

Le codesign

- Les outils de synthèse récents permettent une vue unifiée du matériel et du logiciel
- Le choix matériel versus logiciel pour une fonction particulière est un compromis entre différentes métriques de design : performance, puissance consommée, dimensions physiques, coûts, NRE et flexibilité. Fondamentalement, il n'existe pas de différence entre ce que le matériel et le logiciel peuvent réaliser tant qu'on ne dépasse pas les limites technologiques



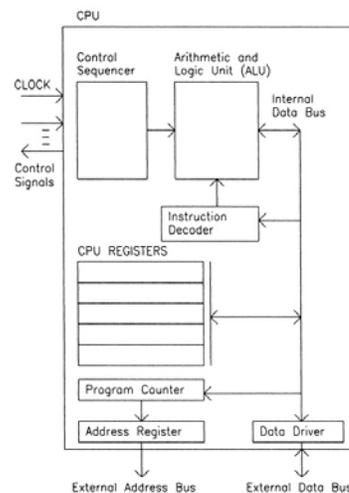
Traduit et adapté de *Embedded Systems Design*, (c) 2000 Vahid/Givargis, et autres sources

Exemples de designs

Traduit et adapté de l'anglais

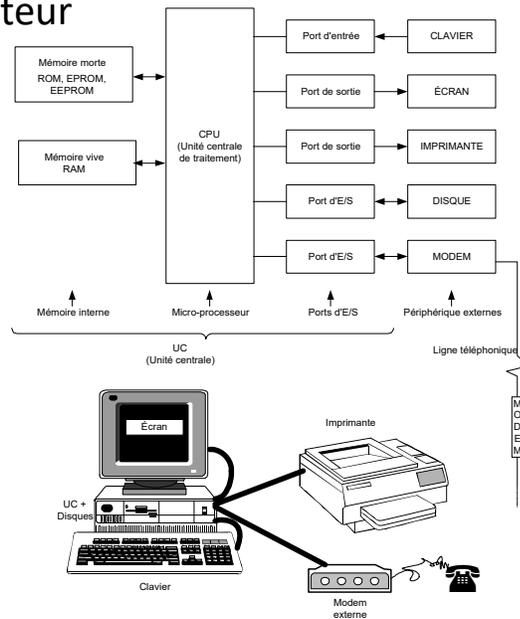
Caractéristiques communes

- Le CPU et son horloge (« métronome »)
- Accumulateur/registre(s) et ALU (unité arithmétique et logique)
- Registres auxiliaires aux fonctions diverses
- **Un CPU enrichi de mémoires externes et d'interfaces au monde extérieur forme un micro-ordinateur**



Traduit et adapté de *Embedded Systems Design*, (c) 2000 Vahid/Givargis, et autres sources

Micro-ordinateur



Traduit et adapté de *Embedded Systems Design*, (c) 2000 Vahid/Givargis, et autres sources

Et ceux-ci ?

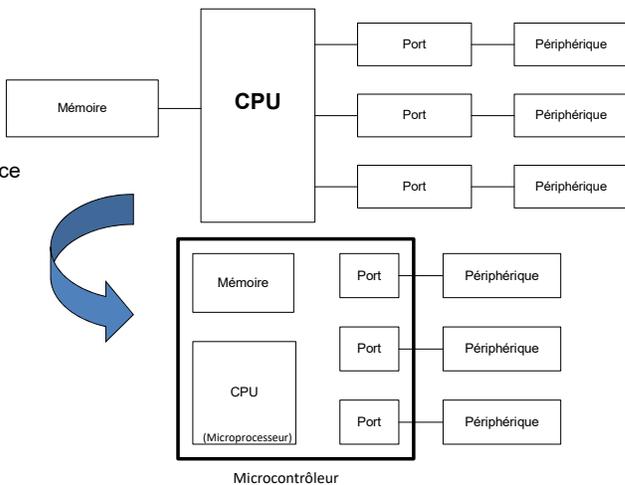


- Les systèmes embarqués comptent pour 94% des ventes
- Étendent les applications bien au delà des applications bureautiques ou de gestion

Traduit et adapté de *Embedded Systems Design*, (c) 2000 Vahid/Givargis, et autres sources

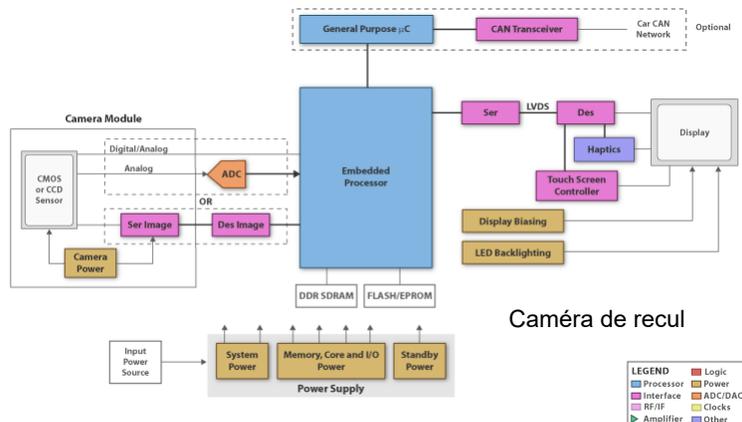
Microcontrôleur

- Micro-ordinateur à ressources réduites
 - Composantes fondamentales d'un micro-ordinateur intégrées sur une puce
- « CPU » des systèmes embarqués



Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Système de vision pour véhicule automobile

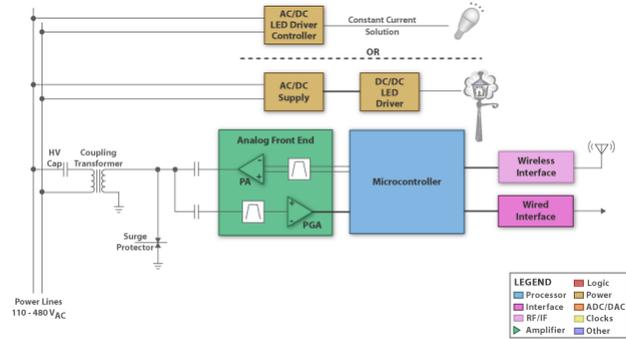


- Traite l'information venant de caméras, lasers, sonars et autres capteurs pour réaliser des tâches comme l'avertissement de changement de ligne, de somnolence, ou l'aide au stationnement.
- Les résultats peut être affichés sur écran ou annoncés par haut-parleur/sirène.

<http://www.ti.com/general/docs/lit/getliterature.tsp?literatureNumber=spry199&fileType=pdf>

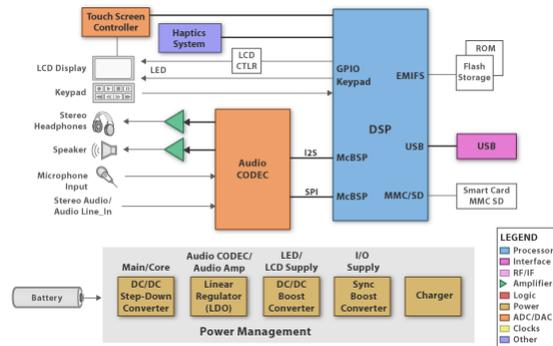
Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Système d'éclairage public par LED à haute intensité



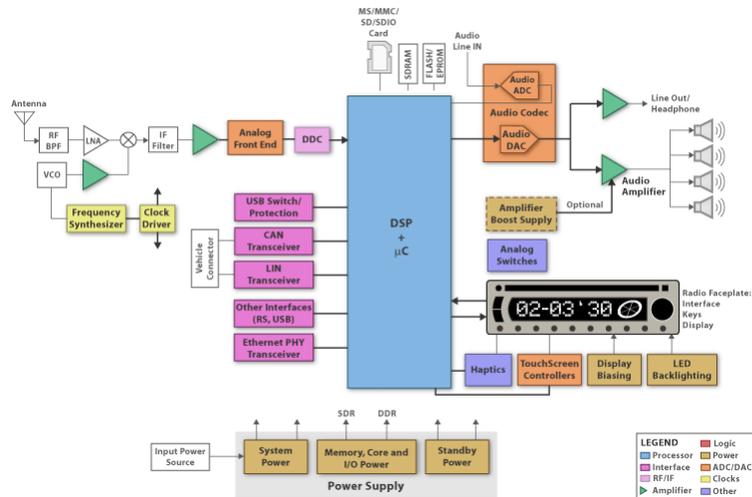
Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Lecteur/enregistreur MP3



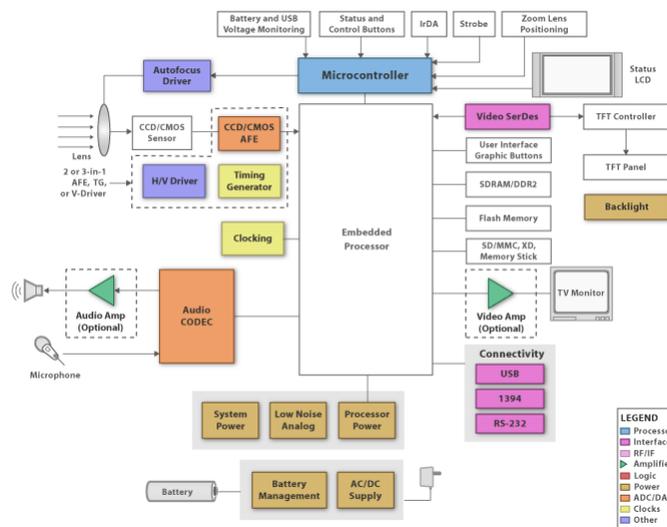
Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Radio numérique



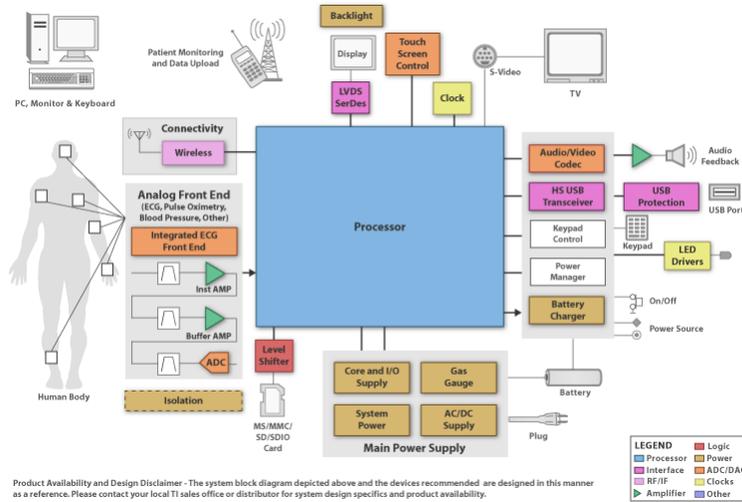
Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Appareil photo numérique



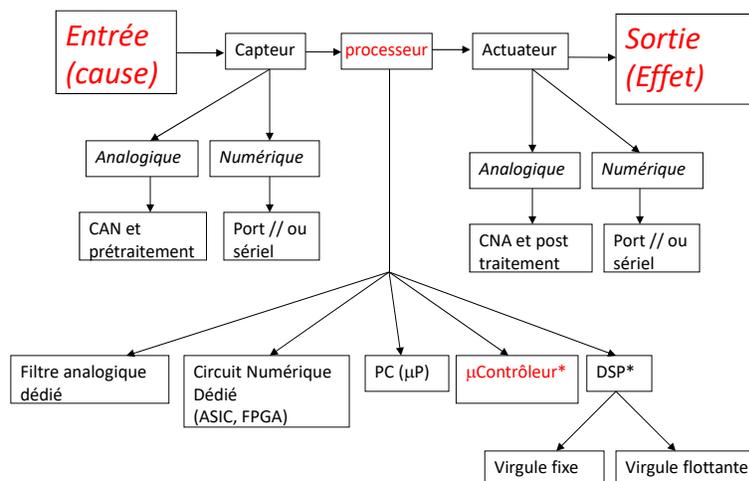
Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Surveillance de patient



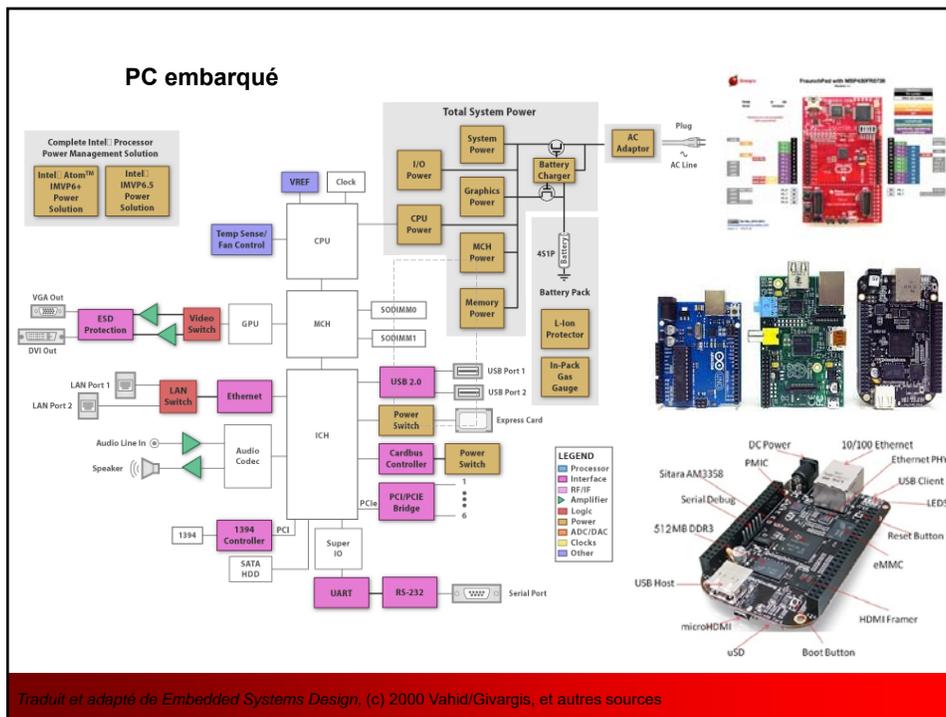
Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources

Digramme d'un système embarqué



* Comprend aussi des ports d'e/s et des mécanismes de gestion de la mémoire

Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources



En somme...

- Les systèmes embarqués sont partout
- Métriques de design souvent en conflit les unes avec les autres
- Nécessité d'une vue unifiée du matériel et du logiciel pour optimiser le processus de développement
- Trois aspects technologiques clés :
 - Processeur : usage général, spécialisé, dédié
 - Puce : personnalisée, reconfigurable, programmable
 - Design: Matériel, logiciel, mixte avec optimisation top-down et tests de validation

Traduit et adapté de Embedded Systems Design, (c) 2000 Vahid/Givargis, et autres sources