

RTS6: Conception et programmation de Systèmes Embarqués

cours 1: Introduction aux systèmes embarqués

Antoine Fraboulet, Fabrice Jumel, Lionel Morel, Tanguy Risset
tanguy.risset@insa-lyon.fr
Lab CITI, INSA de Lyon

- p. 1/133



Plan du cours 1 (2H)

- Introduction générale aux systèmes embarqués
- Architecture des processeurs embarqués
- Présentation rapides du MSP430
- Systèmes sur puces

Introduction

Processeurs embarqués

introduction au MSP 430

Présentation des SoC

- p. 2/133



Introduction

● Un peu d'histoire

● Aujourd'hui

Processeurs embarqués

introduction au MSP 430

Présentation des SoC

Un peu d'histoire

- p. 3/133



Introduction

● Un peu d'histoire

● Aujourd'hui

Processeurs embarqués

introduction au MSP 430

Présentation des SoC

Un peu d'histoire

- Fin du XIXème siècle : démonstration du potentiel des technologies électroniques pour les transmissions sans fil.
- Seconde guerre mondiale
 - ◆ Application au calcul (décryptage)
 - ◆ Premiers ordinateurs (ENIAC)
- 1947-1954
 - ◆ Invention puis commercialisation des transistors à semi-conducteurs
 - ◆ Les « transistors » remplacent les postes radio à tubes
- Années 60 et 70: premiers circuits intégrés, LSI

- p. 4/133

Un peu d'histoire

- Introduction
- Un peu d'histoire
- Aujourd'hui
- Processeurs embarqués
- introduction au MSP 430
- Présentation des SoC

- 1960-2002
 - ◆ Réduction des tailles de transistors (≈ 10000)
 - ◆ Évolution anticipée (loi de Moore)
 - ◆ Industrie au coeur de la croissance économique des 50 dernières années
 - ◆ Augmentation exponentielle des performances
 - Puissance (Joy) : $MIPS = 2^{\text{année}-1984}$
 - Densité (Moore) : $\text{Transistors par puce} = 2^{\text{année}-1964}$
 - Densité des supports magnétiques «Maximal Areal Density» (Frank): $MAD = 10^{\frac{\text{année}-1971}{10}}$
 - “Stagnation” des performances pour la rapidité d'accès aux disques → RAID (multiplications des disques) et caches.

- p. 5/133

Aujourd'hui

- Introduction
- Un peu d'histoire
- Aujourd'hui
- Processeurs embarqués
- introduction au MSP 430
- Présentation des SoC

- Circuits mixtes analogique/digital (télécommunications)
- Micro-systèmes (capteurs, actionneurs intégrés)
- Nano-technologies (horizon 5 à 10 ans)
- Taille de gravure: 90 nm
- Coûts de R&D de plus en plus élevés
- La demande (ordinateurs, téléphones mobiles, etc.) stagne
- Peu de produits vraiment nouveaux (VHS => DVD, caméscopes => numériques, téléphones fixes => mobiles)
- Crise financière et économique des TIC

- p. 6/133



Processeurs embarqués

- p. 7/133



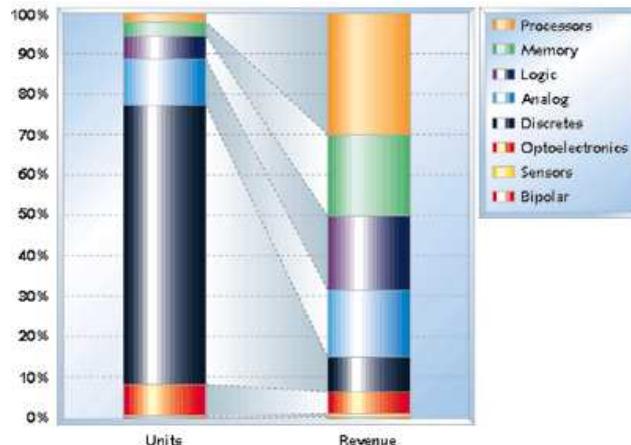
Part de marché

- Quel est le le microprocesseur le plus vendu ?
 - ◆ Réponse classique: "Le Pentium: 92% du marché"
- Faux!.....
 - ◆ En fait les Pentium ne représentent que 2% des microprocesseurs vendus dans le monde.

- p. 8/133

Contradiction ?

- Alors d'où vient la position d'Intel (16% du marché des semi-conducteurs) ?
- processeurs: 2% du silicium, 30% des revenus



- p. 9/133

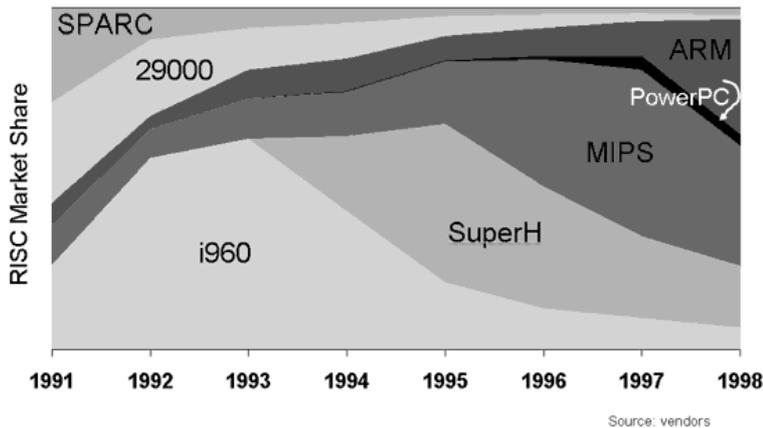
Et au sein des processeurs

- 3 milliards de processeurs 8 bits vendus par an (8051, 6805 etc.)
- 32 bits (Pentium, Athlon, mais aussi PowerPC, 68000, MIPS, ARM etc.)
- La plupart (98%) sont embarqués (3 fois plus d'ARM vendus que de Pentium)

- p. 10/133

Variété des processeurs embarqués

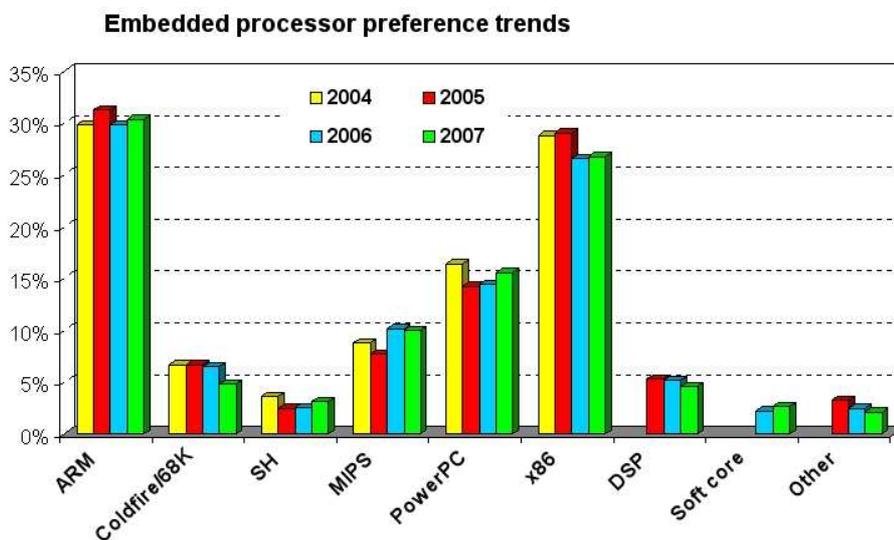
Embedded RISC Lead Swings Constantly



- Les applications sont plus variées que pour les ordinateurs
- Beaucoup de processeurs embarqués sont des processeurs de bureau qui n'ont pas percés (MIPS, 68K, SPARC, ARM, PowerPC)

- p. 11/133

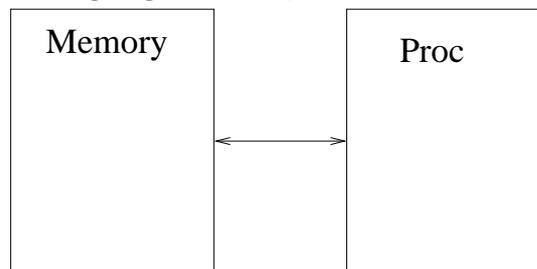
Une image en 2007



- p. 12/133

Architecture "Von Neuman" ou "Princeton"

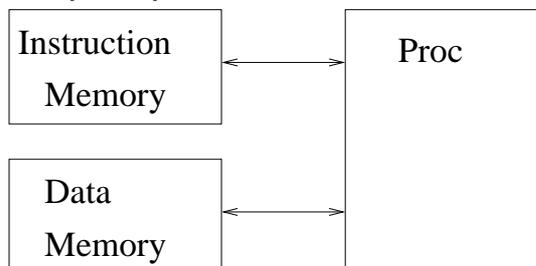
- La mémoire contient les données et les instructions
- L'unité centrale (CPU) charge les instructions depuis la mémoire.
- Un ensemble de registres aide le CPU:
 - ◆ Compteur d'instructions (Program counter: PC),
 - ◆ Registre d'instruction (Instruction register: IR)
 - ◆ Pointeur de pile (stack pointer: SP)
 - ◆ Registres à usage général (Accumulateur: A)



- p. 13/133

Architecture Harvard

- Données et instructions dans des mémoires séparées
- Autorise deux accès simultanés à la mémoire.
- Utilisé pour la plupart des DSP
 - ◆ meilleure bande passante
 - ◆ Performances plus prédictibles



- p. 14/133

Le jeu d'instruction

Introduction

Processeurs embarqués

● Processeurs embarqués

● Économie

● ISA

● Pipeline

● Classification

● Caractéristiques

introduction au MSP 430

Présentation des SoC

- Le *jeu d'instruction* (Instruction Set Architecture: ISA) a une importance capitale
 - ◆ Il détermine les instructions élémentaires exécutées par le CPU.
 - ◆ C'est un équilibre entre la complexité matérielle du CPU et la facilité d'exprimer les actions requises
 - ◆ On le représente de manière symbolique (ex: MSP, code sur 16 bits):

```
                mov r5,@r8 ; commentaire [R8]<-R5
lab:            ADD r4,r5   ; R5<-R5+R4
```

- Deux classes de jeux d'instructions:
 - ◆ CISC: Complex Instruction Set Computer
 - ◆ RISC: Reduce Instruction Set Computer

- p. 15/133

CISC: Complex Instruction Set Computer

Introduction

Processeurs embarqués

● Processeurs embarqués

● Économie

● ISA

● Pipeline

● Classification

● Caractéristiques

introduction au MSP 430

Présentation des SoC

- Une instruction peut designer plusieurs opérations élémentaires.
 - Ex: un load, une opération arithmétique et un store,
 - Ex: calculer une interpolation linéaire de plusieurs valeurs en mémoire.
- Accélération par des mécanismes matériels complexes
- Grandes variations de taille et de temps d'exécution pour les instructions
- Résulte en un code compact mais complexe à générer.
- Vax, Motorola 68000, Intel x86/Pentium

- p. 16/133

Exemple: instructions de l'ISA du Pentium

Introduction

Processeurs embarqués

● Processeurs embarqués

● Économie

● ISA

● Pipeline

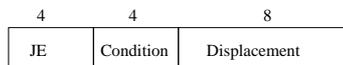
● Classification

● Caractéristiques

introduction au MSP 430

Présentation des SoC

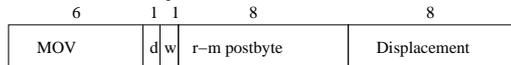
JE EIP + displacement



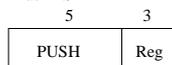
Call



Mov \$EBX, [EDI+displacement]



Push ESI



Add \$EAX, Immediate



Test \$EDX, Immediate



- p. 17/133

RISC: Reduced Instruction Set Computer

Introduction

Processeurs embarqués

● Processeurs embarqués

● Économie

● ISA

● Pipeline

● Classification

● Caractéristiques

introduction au MSP 430

Présentation des SoC

- Petites instructions simples, toutes de même taille, ayant toutes (presque) le même temps d'exécution
- Pas d'instruction complexe
- Accélération en pipelinant l'exécution (entre 3 et 7 étages de pipeline pour une instruction) ⇒ augmentation de la vitesse d'horloge
- Code plus simple à générer, mais moins compact
- Tous les microprocesseurs modernes utilisent ce paradigme: SPARC, MIPS, ARM, PowerPC, etc.

- p. 18/133

Exemple: instructions de l'ISA du MSP

- Introduction
- Processeurs embarqués
 - Processeurs embarqués
 - Économie
 - ISA
 - Pipeline
 - Classification
 - Caractéristiques
- introduction au MSP 430
- Présentation des SoC

1 operand instruction

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	1	0	0	opcode			B/W	Ad		Dest reg.			

relative Jumps

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	1	condition			PC offset (10 bits)									

2 operands instruction

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
opcode			Dest reg.			Ad	B/W	As			Dest reg.				

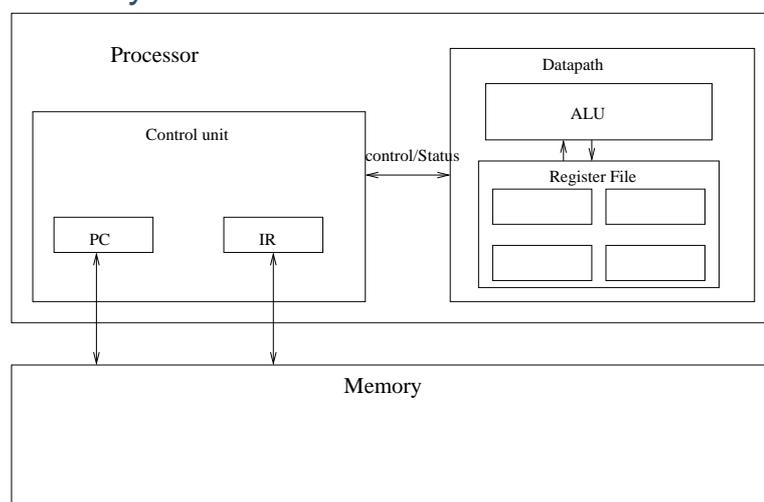
Exemples:

- PUSB.B R4
- JNE -56
- ADD.W R4,R4

Le CPU

- Introduction
- Processeurs embarqués
 - Processeurs embarqués
 - Économie
 - ISA
 - Pipeline
 - Classification
 - Caractéristiques
- introduction au MSP 430
- Présentation des SoC

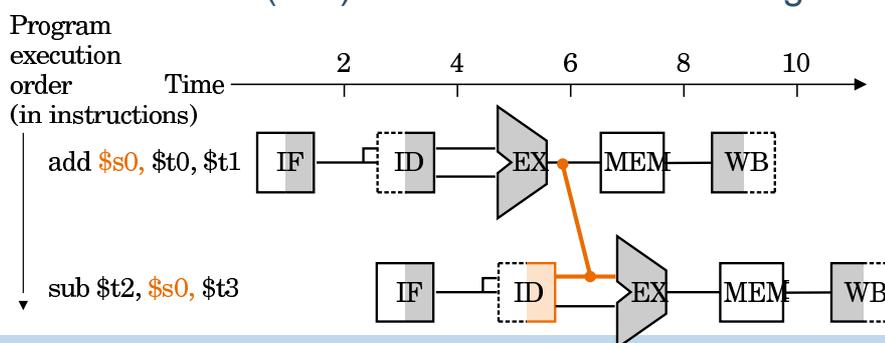
- L'unité de contrôle configure le chemin de donnée suivant l'instruction à exécuter.
- L'exécution d'une instruction est décomposée en plusieurs phases d'un cycle.



Le pipeline RISC: exemple du MIPS

- Introduction
- Processeurs embarqués
 - Processeurs embarqués
 - Économie
 - ISA
 - Pipeline
 - Classification
 - Caractéristiques
- introduction au MSP 430
- Présentation des SoC

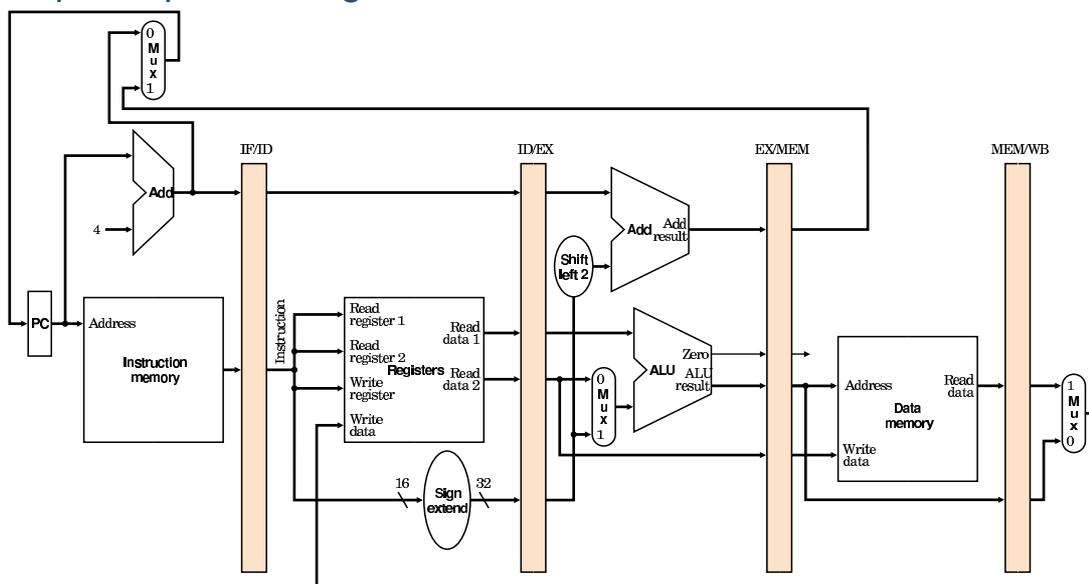
- Le pipeline dépend de l'architecture, pour le MIPS:
 - ◆ Instruction Fetch (IF, Fetch): charge l'instruction dans l'IR
 - ◆ Instruction Decode (ID, Decode): décode l'instruction et met en place le contrôle du chemin de donnée
 - ◆ Execute (Ex): exécute le calcul dans le chemin de donnée.
 - ◆ Memory access (Mem): accède la mémoire
 - ◆ Write Back (WB): écrit dans le banc de registre



Le pipeline RISC: exemple du MIPS

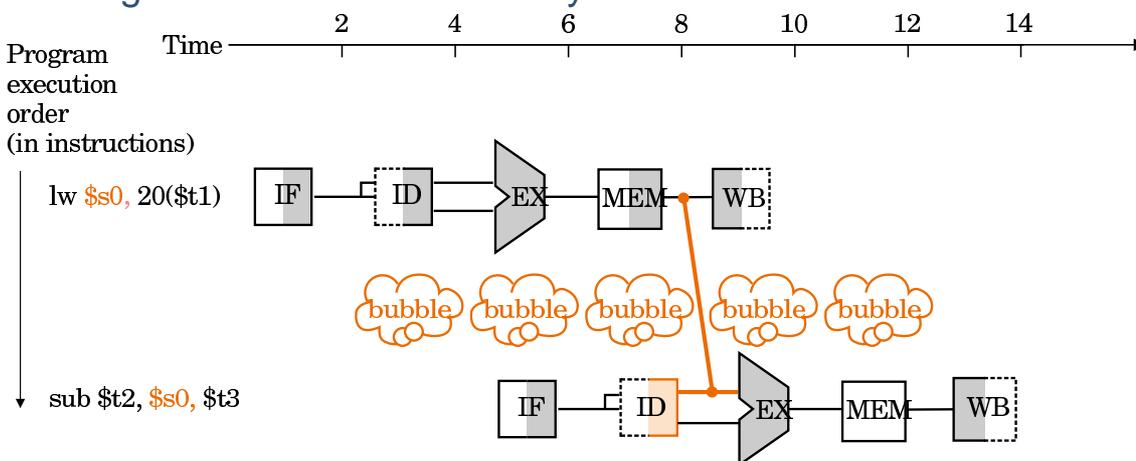
- Introduction
- Processeurs embarqués
 - Processeurs embarqués
 - Économie
 - ISA
 - Pipeline
 - Classification
 - Caractéristiques
- introduction au MSP 430
- Présentation des SoC

- Physiquement, l'architecture du processeur est organisée en calculs combinatoires pour chaque étape de pipeline, séparés par des registres.



Le pipeline RISC: exemple du MIPS

- Lorsque l'instruction suivante ne peut pas être exécutée tout de suite, cela crée une "bulle".
- Par exemple une addition utilisant un registre qui vient d'être chargé doit être retardé d'un cycle.



- p. 23/133

Bilan architecture non pipelinée

- Exécution non pipelinée:
 - ◆ 5 cycles pour exécuter une instruction
 - ◆ ⇒ 15 cycles pour 3 instructions.
- Exécution pipelinée:
 - ◆ 5 cycles pour exécuter une instruction
 - ◆ 8 cycles pour 3 instructions.
 - ◆ ⇒ sans branchement, une instruction par cycle
 - ◆ Un branchement (conditionnel ou pas) interrompt le pipeline car il faut attendre de décoder l'adresse de branchement pour charger l'instruction suivante ⇒ quelques cycles d'inactivité (pipeline stall)
 - ◆ Lors d'un branchement, certain ISA autorisent l'utilisation de ces *delai slots*: une ou deux instructions après le branchement sont exécutées, que le branchement soit pris ou pas (comme si elles étaient écrites avant le branchement).

- p. 24/133

Parallélisme au sein du processeur

Introduction

Processeurs embarqués

● Processeurs embarqués

● Économie

● ISA

● Pipeline

● Classification

● Caractéristiques

introduction au MSP 430

Présentation des SoC

Indépendamment du pipeline, Deux paradigmes dominants:

- Super Scalaire
 - ◆ Duplication des unités,
 - ◆ Répartition au vol des instructions sur les unités disponibles (re-ordonnancement des instructions: *out of order execution*)
 - ◆ Exemple: le PowerPC 970 (4 ALU, 2 FPU)
 - ◆ Efficace mais complexifie l'unité de contrôle (problème des interruptions)
- Very Large Instruction Word (VLIW)
 - ◆ Duplication des unités,
 - ◆ L'ordonnancement des instructions est fixé à la compilation (tout se passe comme si les instructions pouvait être regroupé sur 64 bits, 128 bits etc.)
 - ◆ Inventé par Josh Fisher (Yale) à partir du trace scheduling
 - ◆ Les processeurs VLIW sont tous basés sur les architectures RISC, avec entre 4 et 8 unités.
 - ◆ Exemple: TriMedia (Philips), Itanium IA64 (Intel).

- p. 25/133

Parallélisme au sein du processeur

Introduction

Processeurs embarqués

● Processeurs embarqués

● Économie

● ISA

● Pipeline

● Classification

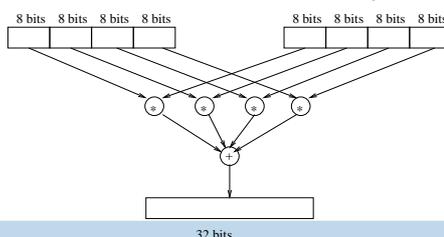
● Caractéristiques

introduction au MSP 430

Présentation des SoC

Une autre approche possible: instructions SIMD.

- Modification du data-path pour proposer des opérations parallèles sur 16 ou 8 bits
- Exemple: Sun Visual Instruction Set, Intel Pentium MMX, Philips TriMedia
- Gains importants sur certains traitements mais très peu utilisé en pratique (difficile à inférer par le compilateur)
 - ◆ Bibliothèques écrites en assembleur (programmes non portables)
 - ◆ Fonction C représentant les instructions assembleurs (*compiler intrinsic*)
 - ◆ Exemple: instruction `ifir8ii R1, R2, R3` du Trimedia:



- p. 26/133

Mémoire

Introduction

Processeurs embarqués

● Processeurs embarqués

● Économie

● ISA

● Pipeline

● Classification

● Caractéristiques

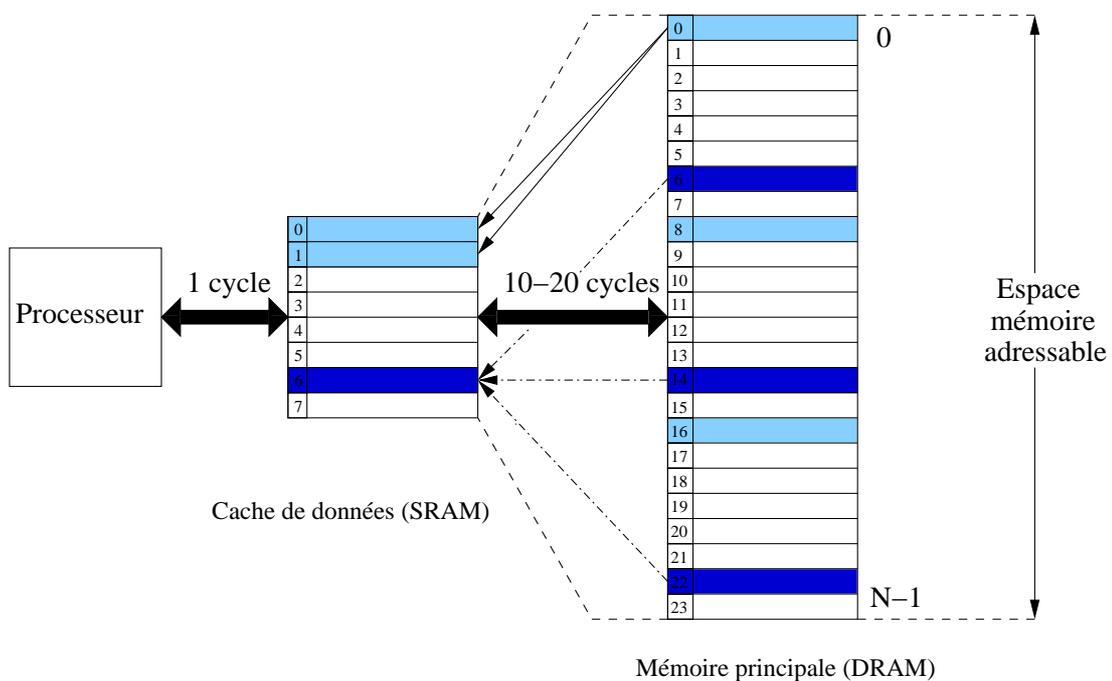
introduction au MSP 430

Présentation des SoC

- Plusieurs technologies pour les mémoires:
 - ◆ Mémoires statiques (SRAM): petites, rapides, consommatrices, peu denses (chères).
 - ◆ Mémoires dynamiques (DRAM): grandes, lentes, très denses, transactions chères
- De plus en plus de place On-Chip pour la mémoire (dans ce cas elles sont moins efficaces que les chips mémoire).
- Ne pas oublier que le code aussi réside en mémoire
- Tous les systèmes ont des caches pour cacher les temps de latence lors de l'accès à la mémoire, en général plusieurs niveaux de caches: hiérarchie mémoire.

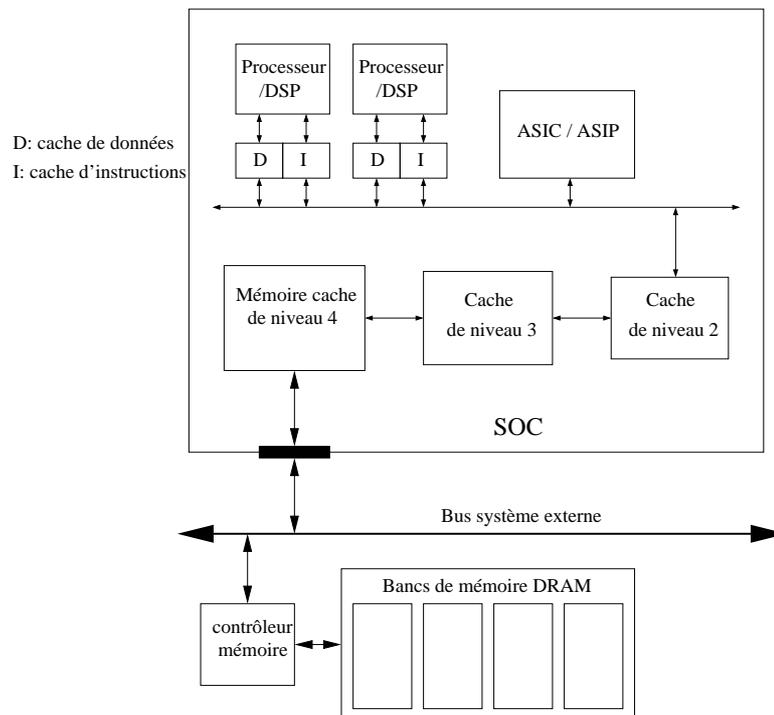
- p. 27/133

Principe du Cache



- p. 28/133

Hiérarchie Mémoire



- p. 29/133

Différents types de processeurs embarqués

- Beaucoup de Processeurs à usage général ayant une ou deux générations
- 4, 8, 16 ou 32 bits (taille des mots)
- RISC et CISC
- DSP: Digital Signal Processor
- ASIP: Application Specific Integrated Processor

- p. 30/133

68000, x86

Introduction

Processeurs embarqués

- Processeurs embarqués
- Économie
- ISA
- Pipeline
- Classification
- Caractéristiques

introduction au MSP 430

Présentation des SoC

- Famille des Motorola 68000
 - ◆ Un des plus vieux processeur embarqué (ex Sun, Mac)
 - ◆ Architecture CISC
 - ◆ ISA propre et les meilleurs outils de développement, beaucoup d'utilisateurs
- Famille des x86
 - ◆ Démarre au 8086 (Intel) puis 80286, 386, 486, Pentium, et Athlon (AMD)
 - ◆ En processeurs embarqués: 5 fois moins que MIPS, ARM ou 68000.
 - ◆ architecture CISC, compatible avec le code du 8086
 - ◆ compatibilité mais mauvaises performances

- p. 31/133

SPARC, 29000 et i960

Introduction

Processeurs embarqués

- Processeurs embarqués
- Économie
- ISA
- Pipeline
- Classification
- Caractéristiques

introduction au MSP 430

Présentation des SoC

- SPARC
 - ◆ Un des premier RISC à avoir été embarqué (pratiquement plus aujourd'hui)
 - ◆ SPARC est une architecture brevetée (soft core, Intellectuel Property: IP), plusieurs compagnies fabriquent des SPARC
- 29000 (AMD)
 - ◆ Le 29000 a eu beaucoup de succès (imprimante laser Apple) grâce à ces 192 registres
 - ◆ AMD a arrêté la production car le développement des outils coûtait trop cher.
- i960 (intel)
 - ◆ Le i960 a été le plus vendu des processeurs embarqués au milieu des années 90 (router réseau et HP Laserjet).

- p. 32/133

MIPS, ARM, SuperH et PowerPC

Introduction

Processeurs embarqués

● Processeurs embarqués

● Économie

● ISA

● Pipeline

● Classification

● Caractéristiques

introduction au MSP 430

Présentation des SoC

- MIPS (microprocessor without interlocked pipeline stages)
 - ◆ Originellement pour les stations puissantes (SGI)
 - ◆ Puis, marché des consoles de jeux (Nintendo N64)
 - ◆ Famille très étendue: du plus gros (MIPS 20Kc, 64 bit) au plus petit (SmartMIPS, 32 bit pour carte à puce)
- ARM (Advanced RISC Machines, ex Acorn)
 - ◆ Un des 32 bits embarqués les plus populaires : téléphones portables
 - ◆ Faible consommation
 - ◆ Le successeur: StrongArm est commercialisé par Intel sous le nom de XScale
- SuperH (ou SH: Hitachi) Utilisé dans les stations Sega et les PDA
- PowerPC autant utilisé en embarqué qu'en ordinateur

- p. 33/133

Et les autres....

Introduction

Processeurs embarqués

● Processeurs embarqués

● Économie

● ISA

● Pipeline

● Classification

● Caractéristiques

introduction au MSP 430

Présentation des SoC

- Plus de 100 processeurs embarqués 32 bits sur le marché
- Les constructeurs de FPGA proposent des soft-processeurs pour configurer les FPGA: Nios (Altera), MicroBlaze (Xilinx)
- Certains processeurs RISC (Crusoe de Transmeta) peuvent exécuter du code CISC (Intel)
 - ◆ Principe: recompilation du code à l'exécution (*runtime compilation*)
 - ◆ Gain obtenu par un mécanisme de cache, d'optimisation poussée des portions de code répétées (boucle), et grâce au parallélisme de niveau instruction
 - ◆ Réduction drastique de la consommation pour des performances équivalentes

- p. 34/133

Micro-contrôleurs

Introduction

Processeurs embarqués

- Processeurs embarqués
- Économie
- ISA
- Pipeline
- Classification
- Caractéristiques

introduction au MSP 430

Présentation des SoC

- Utilisé pour le contrôle embarqué
 - ◆ Censeur, contrôleurs simples
 - ◆ Manipule des événements, quelques données mais en faible quantité
 - ◆ Exemple: caméscope, disque dur, appareil photo numérique, machine à laver, four à micro-onde
- Quelques caractéristiques fréquentes
 - ◆ Périphériques présents sur le circuit (timer, convertisseur analogique numérique, interface de communication), accessible directement grâce aux registres
 - ◆ Programme et données intégrées au circuit
 - ◆ Accès direct du programmeur à de nombreuses broches du circuit
 - ◆ Instructions spécialisées pour les manipulation de bits.
- Le MSP430 appartient à cette catégorie

- p. 35/133

DSP: Digital Signal Processing

Introduction

Processeurs embarqués

- Processeurs embarqués
- Économie
- ISA
- Pipeline
- Classification
- Caractéristiques

introduction au MSP 430

Présentation des SoC

- Utilisés pour les applications de traitement du signal
 - ◆ Grande quantités de données numérisées, souvent organisées en flux
 - ◆ Filtre numérique sur téléphone, TV numérique, synthétiseur de sons
- Relativement proche des GPP, mais quelques caractéristiques en plus:
 - ◆ Bande passante élevée (deux bus)
 - ◆ Instructions dédiées pour les calculs de traitement du signal: multiplication accumulation,
 - ◆ Arithmétique spécifique (mode d'arrondi)
 - ◆ Registres dédiés pour certains opérateurs.
 - ◆ Constructeurs: Texas Instrument, puis Analog Devices, Motorola

- p. 36/133

Quelques mécanismes matériels utiles



- Introduction
- Processeurs embarqués
 - Processeurs embarqués
 - Économie
 - ISA
 - Pipeline
 - Classification
 - Caractéristiques
- introduction au MSP 430
- Présentation des SoC

■ Densité de code:

- ◆ La taille du code est importante pour les codes embarqués car elle influe sur la taille de la mémoire utilisée
- ◆ Un programme C compilé pour SPARC prendra deux fois plus de place en mémoire que le même programme compilé pour le 68030.
- ◆ En général les code RISC sont deux fois moins dense que les codes CISC (ex: instruction `TBLIS` du 68300: *table lookup and interpolate*)
- ◆ Les options de compilation doivent être utilisées avec précaution.
- ◆ Le code est quelquefois stocké compressé et décompressé au vol par du matériel spécifique.

- p. 37/133

Quelques mécanismes matériels utiles



- Introduction
- Processeurs embarqués
 - Processeurs embarqués
 - Économie
 - ISA
 - Pipeline
 - Classification
 - Caractéristiques
- introduction au MSP 430
- Présentation des SoC

■ Manipulations au niveau bit:

- ◆ Utilisé pour les algorithmes de cryptage mais surtout pour les pilotes de périphériques.
- ◆ La plupart des périphériques indiquent leur état au processeur en mettant un certain bit à 1 dans un certain registre.
- ◆ Un processeur standard doit rapatrier le mot de 32 bit, masquer et tester à 0
- ◆ Les instructions `BIC`, `BIT` et `BIS` du MSP430 font des manipulation au niveau bit dans la mémoire

- p. 38/133

Quelques mécanismes matériels utiles



- Introduction
- Processeurs embarqués
 - Processeurs embarqués
 - Économie
 - ISA
 - Pipeline
 - Classification
 - Caractéristiques
- introduction au MSP 430
- Présentation des SoC

■ Données non-alignés

- ◆ De nombreux traitements manipulent des données de taille non-multiple de 32 (paquets TCP/IP, video streams, clés d'encryption, 20 bits, 56 bits)
- ◆ Les processeurs RISC savent uniquement transférer des mots (32 bits) *alignés* (calés sur une adresse multiple de 32 bits).
- ◆ La plupart des architectures CISC (68k, x86) peuvent faire des chargements non alignés

- p. 39/133

Quelques mécanismes matériels utiles



- Introduction
- Processeurs embarqués
 - Processeurs embarqués
 - Économie
 - ISA
 - Pipeline
 - Classification
 - Caractéristiques
- introduction au MSP 430
- Présentation des SoC

■ Gestion spécifique du cache

- ◆ Les caches améliorent les performances mais introduisent du non-déterminisme.
- ◆ Les contraintes spécifiques des systèmes embarqués ont entraîné des mécanismes particuliers pour les cache
- ◆ On peut vouloir bloquer le cache (cache locking): forcer certaines données ou instruction à se charger et rester dans le cache (on parle aussi de mémoire *scratch-pad memory* ou de *software controlled cache*).
- ◆ La plupart des caches utilisent une politique de Write-Back: une donnée modifiée dans le cache n'est pas forcément immédiatement recopiée en memoire. Dans le cas de périphériques mappés en mémoire, il est indispensable de recopier immédiatement (politique *write-through*)

- p. 40/133

Quelques mots sur la consommation

Introduction

Processeurs embarqués

- Processeurs embarqués
- Économie
- ISA
- Pipeline
- Classification
- Caractéristiques

introduction au MSP 430

Présentation des SoC

■ Trois composantes de la consommation d'une porte logique (inverseur)

- ◆ Consommation dynamique : $P_{dyn} = C.V_{CC}^2$
(C capacité de la porte)
- ◆ Consommation statique : $P_{static} = V_{CC}.I_{leak}$
(V_{CC} : tension d'alimentation, I_{leak} intensité des courants de fuite)
- ◆ Consommation de court-circuit $P_{cs} = K.\tau.(V_{CC} - 2V_{Th})^3$.
(K : constante technologique ; V_{Th} : tension seuil ; τ : temps de montée descente du signal)

■ Aujourd'hui (2004) $P_{dyn} \gg P_{static} \gg P_{cs}$

■ Demain (2007) $P_{dyn} \approx P_{static} \gg P_{cs}$

- p. 41/133

Consommation d'un circuit CMOS

Introduction

Processeurs embarqués

- Processeurs embarqués
- Économie
- ISA
- Pipeline
- Classification
- Caractéristiques

introduction au MSP 430

Présentation des SoC

■ Généralisation naïve en prenant en compte une activité moyenne α (nombre moyen de portes commutant)

- ◆ Consommation dynamique : $P_{dyn} = C.V_{CC}^2.\alpha.f$
(f : fréquence du circuit)
- ◆ Consommation statique : $P_{static} = V_{CC}.I_{leak}.N.k_{design}$
(N : nombre de portes, k_{design} constante dépendant du design)

■ Cette modélisation est très imprécise pour un circuit dont le comportement n'est pas stationnaire (ex: processeur)

- p. 42/133

Réduction statique de la consommation

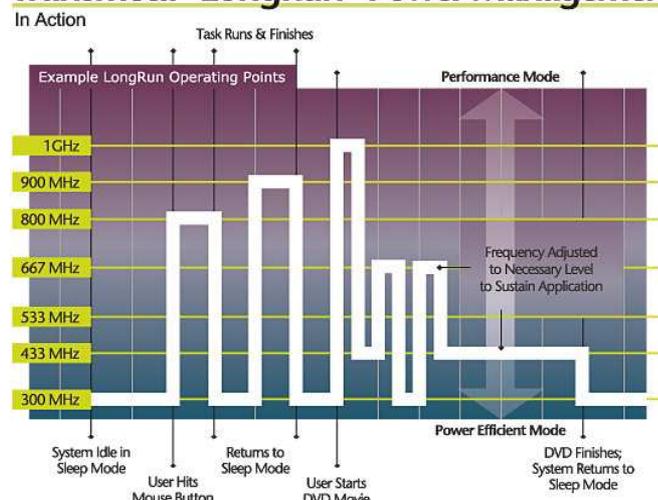
- Le facteur le plus important est la tension d'alimentation (V_{CC}) d'abord 3.3 V puis 2.5 V. Les versions récentes de Xscale (strong ARM, Intel) et les puces smartCard fonctionnent à 0.65 V
- On peut différencier les tensions en fonction du bloc du chip: 1.5 V pour le processeur, 3.3 pour l'interface du bus et les pattes d'entrée/sortie (ex: Strong ARM de Digital)
- Plus la technologie est intégrée, moins elle consomme (capacité diminuée).
- Fréquence d'horloge peu élevée compensée par le parallélisme
- Complexité réduite des différents composants (moins de registres, architectures RISC)

- p. 43/133

Réduction dynamique de la consommation

- Gestion dynamique de la fréquence d'horloge
- Exemple: processeur Crusoe (Transmeta)
Suppression de l'horloge sur un bloc (*Dynamic clock gating*)
- Gestion dynamique de l'alimentation (pas encore réalisé)

Transmeta™ LongRun™ Power Management



- p. 44/133

Low Power Mode pour le MSP430

Introduction

Processeurs embarqués

- Processeurs embarqués
- Économie
- ISA
- Pipeline
- Classification
- Caractéristiques

introduction au MSP 430

Présentation des SoC

- Différent mode pour réduire la consommation
 - ◆ LPM0: le CPU est arrêté
 - ◆ LPM1, LPM2: l'horloge rapide (MCLK) est aussi arrêtée
 - ◆ LPM3 le générateur d'horloge est arrêté
 - ◆ LPM4 : l'oscillateur du crystal est arrêté
- Le temps de reprise est d'autant plus long que la veille est profonde.

- p. 45/133

Introduction

Processeurs embarqués

introduction au MSP 430

- introduction au MSP 430
- TI MSP430 : ez430-rf2500
- Documents techniques
- Architecture
- Périphériques
- Chaîne de développement

Présentation des SoC

introduction au MSP 430

- p. 46/133

TI MSP430 : ez430-rf2500

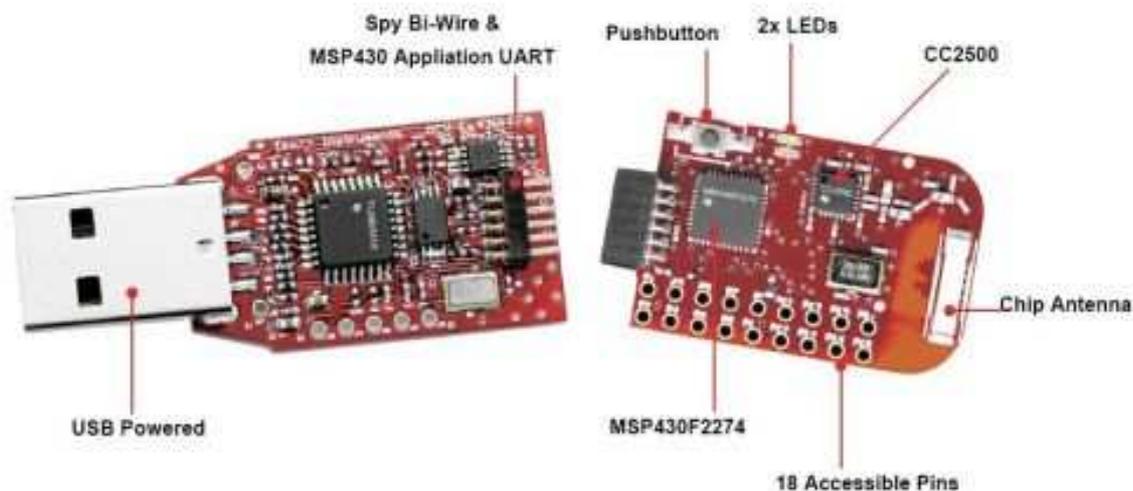
Introduction

Processeurs embarqués

introduction au MSP 430

- introduction au MSP 430
- TI MSP430 : ez430-rf2500
- Documents techniques
- Architecture
- Périphériques
- Chaîne de développement

Présentation des SoC



- p. 47/133

Documents techniques

Introduction

Processeurs embarqués

introduction au MSP 430

- introduction au MSP 430
- TI MSP430 : ez430-rf2500
- Documents techniques
- Architecture
- Périphériques
- Chaîne de développement

Présentation des SoC

- Documents du cours <http://>
- Manuel de programmation du MSP
(MSP430x2xx_Family_User's_Guide_(Rev._D)_slau144d.pdf)
- Manuel du MSP430F2274 (msp430f2274.pdf)
- Feuilles de schématique
- User's Manual EZ430
(ez430-RF2500_User_Guide_SLAU227A.pdf)
- Datasheet du composant radio (cc2500.pdf)
- Documents IAR
 - ◆ IAR compiler reference guide, ...
- Site de Texas Instrument
 - ◆ Exemples de programme / drivers, Notes d'application, ...
- The mspgcc toolchain:
<http://mspgcc.sourceforge.net/>
- Notamment la doc mspgcc (*download puis documentation*)

- p. 48/133

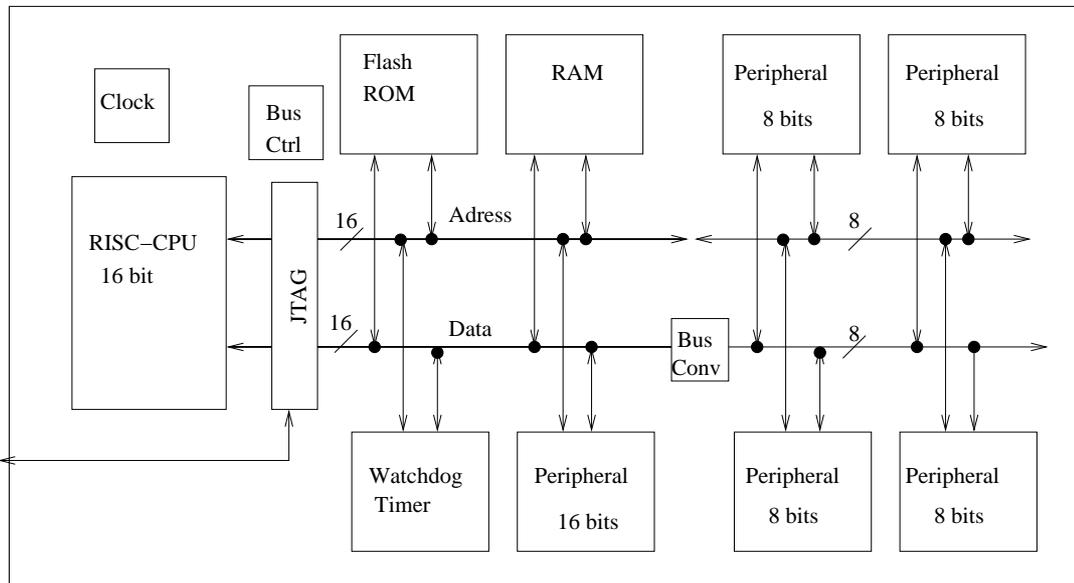
Architecture du MSP

- Introduction

- Processeurs embarqués

- introduction au MSP 430
 - introduction au MSP 430
 - TI MSP430 : ez430-rf2500
 - Documents techniques
 - Architecture
 - Périphériques
 - Chaîne de développement

- Présentation des SoC



CPU RISC 16 bits

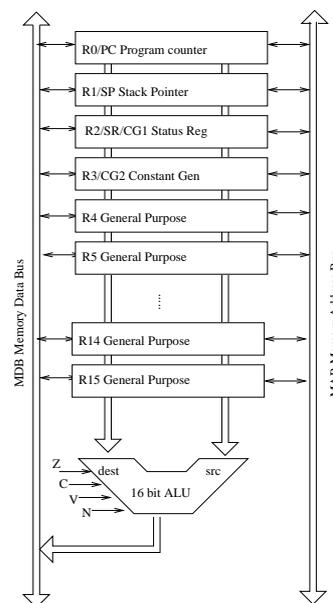
- Introduction

- Processeurs embarqués

- introduction au MSP 430
 - introduction au MSP 430
 - TI MSP430 : ez430-rf2500
 - Documents techniques
 - Architecture
 - Périphériques
 - Chaîne de développement

- Présentation des SoC

- 28 Instructions sur 16 bits
- 64 Ko de mémoire adressable
- Périphériques mappés en mémoire
- 16 registres 16 bits (r0-r16)
 - ◆ r0: PC (Program counter)
 - ◆ r1: SP (Stack pointeur)
 - ◆ r2: SR (status register)
 - ◆ r3: constante 0



R3: Status Register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Reserved								V	SCG1	SCG1	OS-COFF	CPU OFF	GIE	N	Z	GC

Périphérique intégrés

Introduction

Processeurs embarqués

introduction au MSP 430

- introduction au MSP 430
- TI MSP430 : ez430-rf2500
- Documents techniques
- Architecture
- Périphériques
- Chaîne de développement

Présentation des SoC

- timers
- contrôleur LCD
- mutliplieur cablé
- contrôleur de bus
- convertisseur analogique numérique, comparateur
- ports séries

- p. 51/133

Périphérique mappé en mémoire

Introduction

Processeurs embarqués

introduction au MSP 430

- introduction au MSP 430
- TI MSP430 : ez430-rf2500
- Documents techniques
- Architecture
- Périphériques
- Chaîne de développement

Présentation des SoC

- l'écriture à une certaine adresse est interprétée comme une communication avec le périphérique.
- Exemple : le multiplieur matériel
 - ◆ Accessible par des registres mappés entre les adresses $0x0130$ et $0x013F$
 - ◆ Écriture à l'adresse $0x130$, positionne le premier opérande (unsigned mult)
 - ◆ Écriture à l'adresse $0x138$, positionne le deuxième opérande et lance le calcul
 - ◆ Le résultat est à l'adresse $0x013A$, sur 32 bits
- Les autres périphériques sont aussi accessibles par des registres mappé en mémoire: les SFR (Special Function Registers), en C:
 - ◆ écriture vers le périphérique: `SFR = valeur`
 - ◆ lecture des registres du périphérique: `variable = SFR`

- p. 52/133

Exemple: multiplieur câblé

Introduction

Processeurs embarqués

introduction au MSP 430

- introduction au MSP 430
- TI MSP430 : ez430-rf2500
- Documents techniques
- Architecture
- Périphériques
- Chaîne de développement

Présentation des SoC

```
int main(void) {
    int i;
    int *p,*res;

    p=0x130;
    *p=2;
    p=0x138;
    *p=5;
    res=0x13A;
    i=*res;

    nop();
}
```

```
int main(void) {
    int i;
    int *p,*res;

    __asm__("mov #304, R4");
    __asm__("mov #2, @R4");
    // p=0x130;
    // *p=2;
    __asm__("mov #312, R4");
    __asm__("mov #5, @R4");
    //p=0x138;
    // *p=5;
    __asm__("mov #314, R4");
    __asm__("mov @R4, R5");
    //res=0x13A;
    i=*res;

    nop();
}
```

- p. 53/133

Mapping mémoire

Introduction

Processeurs embarqués

introduction au MSP 430

- introduction au MSP 430
- TI MSP430 : ez430-rf2500
- Documents techniques
- Architecture
- Périphériques
- Chaîne de développement

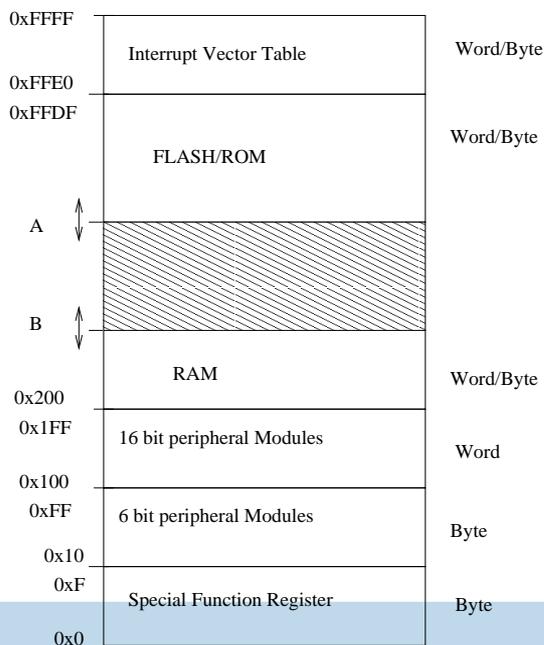
Présentation des SoC

■ Interrupt Vector Table: indique les adresses des fonctions de gestion des interruption

■ Lors du Boot, le MSP va lire l'adresse 0xFFFFE: handler du reset.

■ Sur le MSP430F149:

- ◆ 0x0 à 0x1FF: périphériques
- ◆ 0x200 à B=0x9FF: RAM (2Ko), Données et pile d'exécution
- ◆ 0xC00 à 0xFFFF: Boot mem.
- ◆ 0x1000 à 0x10FF: byte info. mem.
- ◆ A=0x1100 à 0xFFFF: ROM (60 Ko): code.



- p. 54/133

Chaîne de développement

Introduction

Processeurs embarqués

introduction au MSP 430

- introduction au MSP 430
- TI MSP430 : ez430-rf2500
- Documents techniques
- Architecture
- Périphériques
- Chaîne de développement

Présentation des SoC

- Plusieurs configurations
 - ◆ Programmation directe du micro-contrôleur
 - ◆ Simulation du micro-contrôleur (sans les périphériques)
- Plusieurs plateformes de développement
 - ◆ `mspgcc` (GNU)
 - ◆ IAR (texas Instrument)
 - ◆ ..

- p. 55/133

RTS6: Conception et programmation de Systèmes

Introduction

Processeurs embarqués

introduction au MSP 430

Présentation des SoC

- Qu'est ce qu'un SoC
- Exemple SoC: appareil photo numérique
- Autre exemple: iPaQ H5500
- Architecture carte iPaQ H5500
- Architecture S3CA400A01 (companion chip)
- Critères de qualité de conception d'un SOC
- Choix en fonction des contraintes
- Méthodologie de conception
- SocLib
- Organisation logicielle
- Exemple d'une plateforme soclib

Introduction aux systèmes sur puces (SoC)

- p. 56/133

Qu'est ce qu'un SoC

Introduction

Processeurs embarqués

introduction au MSP 430

Présentation des SoC

- Qu'est ce qu'un SoC
- Exemple SoC: appareil photo numérique
- Autre exemple: iPaQ H5500
- Architecture carte iPaQ H5500
- Architecture S3CA400A01 (companion chip)
- Critères de qualité de conception d'un SOC
- Choix en fonction des contraintes
- Méthodologie de conception
- SocLib
- Organisation logicielle
- Exemple d'une plateforme soclib

- "System" : une collection d'éléments en interaction.
- "On Chip" : sur un seul circuit.
- Mais tous les circuits intégrés ne sont pas des SOC, Un SOC implémente une fonction "complète"
 - ◆ Décodeur MPEG2 vidéo + audio + système + transport + graphique + interface utilisateur
 - ◆ Terminal GSM : tout sauf la RF

- p. 57/133

Q'est ce qu'un SoC

Introduction

Processeurs embarqués

introduction au MSP 430

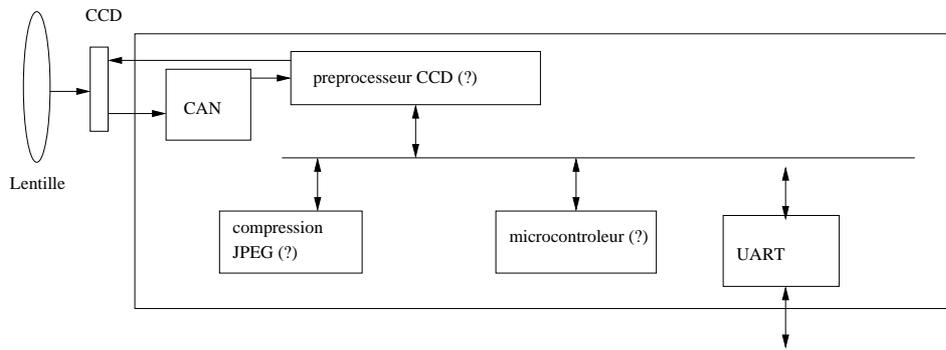
Présentation des SoC

- Qu'est ce qu'un SoC
- Exemple SoC: appareil photo numérique
- Autre exemple: iPaQ H5500
- Architecture carte iPaQ H5500
- Architecture S3CA400A01 (companion chip)
- Critères de qualité de conception d'un SOC
- Choix en fonction des contraintes
- Méthodologie de conception
- SocLib
- Organisation logicielle
- Exemple d'une plateforme soclib

- Une seule fonction
un seul programme, exécuté en boucle
- Contraintes importantes
coût, consommation, taille, performance, . . .
- Prise en compte de l'environnement
doivent réagir en fonction de nombreux paramètres
souvent associé à des contraintes de temps-réel
- Les éléments constitutifs d'un SOC sont "complexes", réutilisables et de nature variée :
 - ◆ Fonctions analogiques (convertisseurs A/N, filtres, etc.)
 - ◆ Fonctions numériques câblées (décodeur de Viterbi)
 - ◆ Fonctions logicielles (sur micro-contrôleur, DSP, RISC)
 - ◆ Composants de base (mémoires)

- p. 58/133

Exemple SoC: appareil photo numérique



- Une seule fonction : prendre des photos
- Contraintes : taille, poids, consommation

- p. 59/133

Autre exemple: iPaQ H5500

- Site <http://www.handhelds.org/>: "encourage and facilitate the creation of open source software for use on handheld and wearable computers"
- Adaptation de linux pour PDA



- p. 60/133

Introduction

Processeurs embarqués

introduction au MSP 430

Présentation des SoC

- Qu'est ce qu'un SoC
- Exemple SoC: appareil photo numérique
- Autre exemple: iPaQ H5500
- Architecture carte iPaQ H5500
- Architecture S3CA400A01 (companion chip)
- Critères de qualité de conception d'un SOC
- Choix en fonction des contraintes
- Méthodologie de conception
- SocLib
- Organisation logicielle
- Exemple d'une plateforme soclib

Introduction

Processeurs embarqués

introduction au MSP 430

Présentation des SoC

- Qu'est ce qu'un SoC
- Exemple SoC: appareil photo numérique
- Autre exemple: iPaQ H5500
- Architecture carte iPaQ H5500
- Architecture S3CA400A01 (companion chip)
- Critères de qualité de conception d'un SOC
- Choix en fonction des contraintes
- Méthodologie de conception
- SocLib
- Organisation logicielle
- Exemple d'une plateforme soclib

iPaQ H5500, une fois démonté

Introduction

Processeurs embarqués

introduction au MSP 430

Présentation des SoC

- Qu'est ce qu'un SoC
- Exemple SoC: appareil photo numérique
- Autre exemple: iPaQ H5500
- Architecture carte iPaQ H5500
- Architecture S3CA400A01 (companion chip)
- Critères de qualité de conception d'un SOC
- Choix en fonction des contraintes
- Méthodologie de conception
- SocLib
- Organisation logicielle
- Exemple d'une plateforme soclib



- p. 61/133

Architecture carte iPaQ H5500

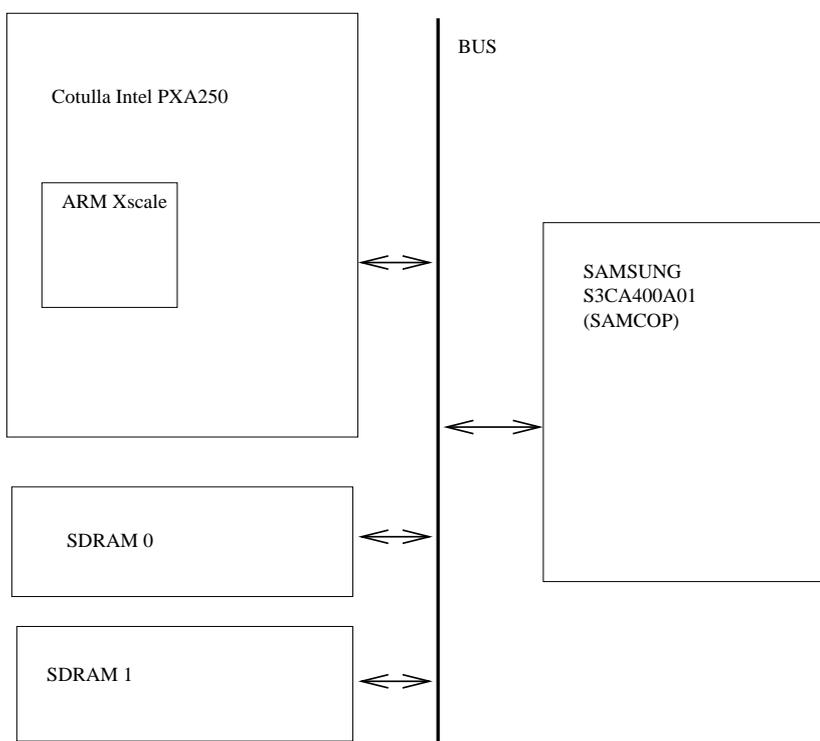
Introduction

Processeurs embarqués

introduction au MSP 430

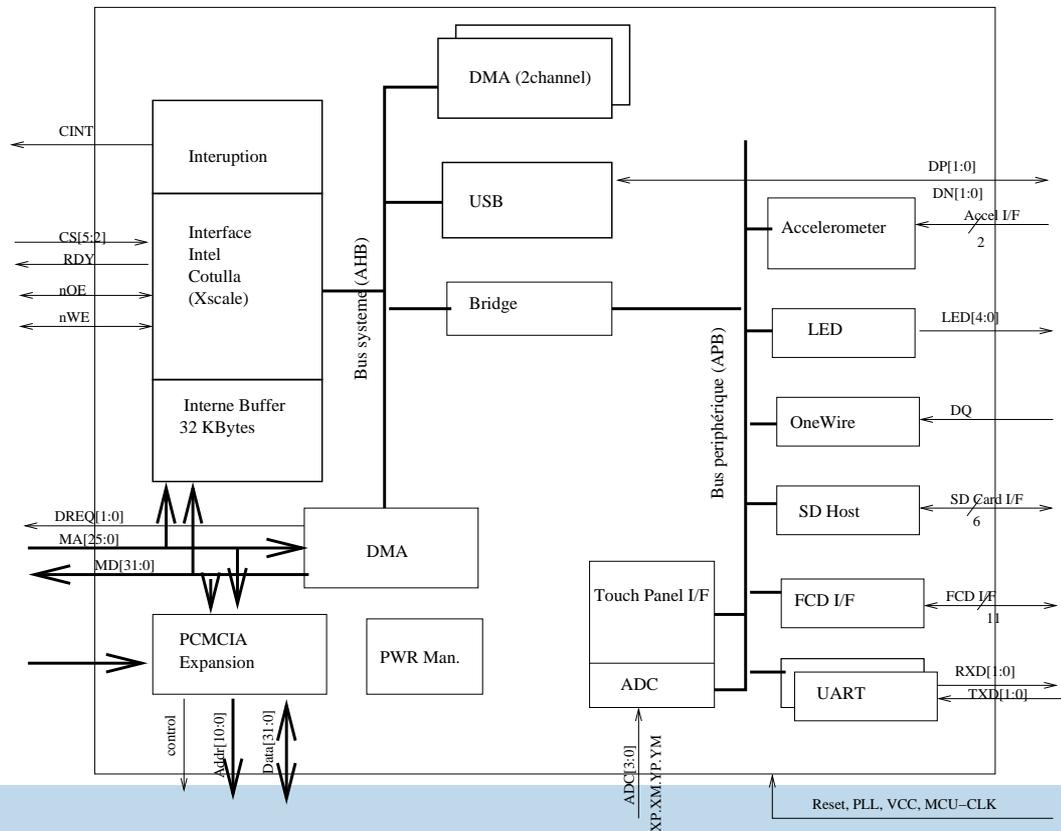
Présentation des SoC

- Qu'est ce qu'un SoC
- Exemple SoC: appareil photo numérique
- Autre exemple: iPaQ H5500
- Architecture carte iPaQ H5500
- Architecture S3CA400A01 (companion chip)
- Critères de qualité de conception d'un SOC
- Choix en fonction des contraintes
- Méthodologie de conception
- SocLib
- Organisation logicielle
- Exemple d'une plateforme soclib



- p. 62/133

Architecture S3CA400A01 (companion chip)



- p. 63/133

Critères de qualité de conception d'un SOC

■ Métriques usuelles

- ◆ Coût à l'unité: coût de fabrication d'une unité sans inclure les coût non récurrents
- ◆ Coûts non récurrents: coût de conception d'un système (coût de mise en place de la première pièce).
- ◆ Taille
- ◆ Performance
- ◆ Consommation
- ◆ Évolutivité: possibilité de faire évoluer le système pour en avoir des versions dérivées.

- p. 64/133

Critères de qualité de conception d'un SOC

Introduction

Processeurs embarqués

introduction au MSP 430

Présentation des SoC

- Qu'est ce qu'un SoC
- Exemple SoC: appareil photo numérique
- Autre exemple: iPaQ H5500
- Architecture carte iPaQ H5500
- Architecture S3CA400A01 (companion chip)
- Critères de qualité de conception d'un SOC
- Choix en fonction des contraintes
- Méthodologie de conception
- SocLib
- Organisation logicielle
- Exemple d'une plateforme soclib

■ Métriques usuelles (suite)

- ◆ Temps de prototypage : temps de mise en place d'un premier système fonctionnel
- ◆ Temps de mise sur le marché : système suffisamment fiable pour être commercialisé
- ◆ Maintenance : possibilité de modifications du système par rapport à sa première version
- ◆ Fiabilité, sûreté de fonctionnement, ...

- p. 65/133

Choix en fonction des contraintes

Introduction

Processeurs embarqués

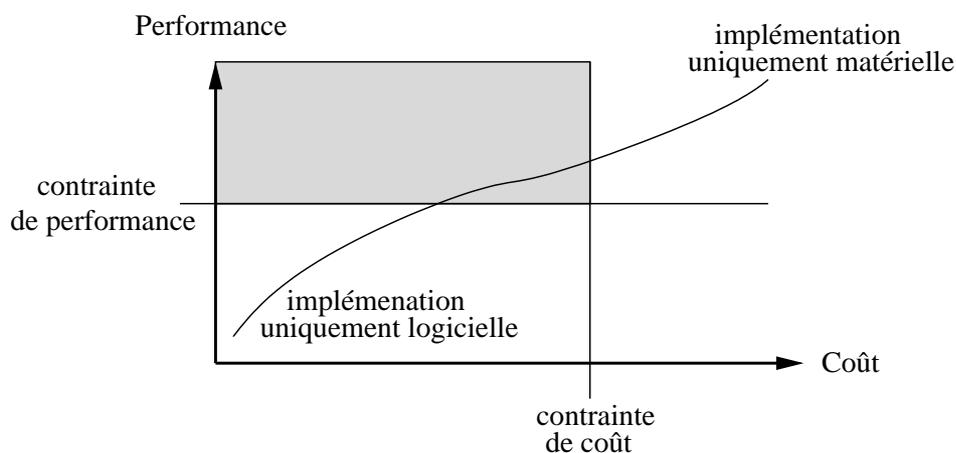
introduction au MSP 430

Présentation des SoC

- Qu'est ce qu'un SoC
- Exemple SoC: appareil photo numérique
- Autre exemple: iPaQ H5500
- Architecture carte iPaQ H5500
- Architecture S3CA400A01 (companion chip)
- Critères de qualité de conception d'un SOC
- Choix en fonction des contraintes
- Méthodologie de conception
- SocLib
- Organisation logicielle
- Exemple d'une plateforme soclib

■ Une grande souplesse de réalisation est possible

- ◆ Il n'y a pas de solution unique



- p. 66/133

Méthodologie de conception

Introduction

Processeurs embarqués

introduction au MSP 430

Présentation des SoC

- Qu'est ce qu'un SoC
- Exemple SoC: appareil photo numérique
- Autre exemple: iPaQ H5500
- Architecture carte iPaQ H5500
- Architecture S3CA400A01 (companion chip)
- Critères de qualité de conception d'un SOC
- Choix en fonction des contraintes
- Méthodologie de conception
- SocLib
- Organisation logicielle
- Exemple d'une plateforme soclib

Intérêts du «codesign»

- Conception rapide de SoC : time to market
 - ◆ Cycle de conception de haut niveau pour réduire les temps d'estimations des solutions
- Réduire la difficulté de validation et débogage
 - ◆ Réutilisation d'IP
 - ◆ Modules reconfigurables
- Converger vers une solution optimale en fonction des contraintes de départ

- p. 67/133

Méthodologie de conception

Introduction

Processeurs embarqués

introduction au MSP 430

Présentation des SoC

- Qu'est ce qu'un SoC
- Exemple SoC: appareil photo numérique
- Autre exemple: iPaQ H5500
- Architecture carte iPaQ H5500
- Architecture S3CA400A01 (companion chip)
- Critères de qualité de conception d'un SOC
- Choix en fonction des contraintes
- Méthodologie de conception
- SocLib
- Organisation logicielle
- Exemple d'une plateforme soclib

- La conception nécessite d'avoir une modélisation complète
- Approche logicielle
 - ◆ les objets migrent vers le HW jusqu'à ce que les contraintes de performances soit atteintes (pour un coût minimum)
- Approche matérielle
 - ◆ les objets migrent en SW tant que les contraintes de performances restent atteintes (pour un coût minimum)
- Le meilleur partitionnement nécessite en général l'expertise d'un concepteur.

- p. 68/133



<http://soclib.lip6.fr/>

- Environnement de conception et de simulation gratuit et open source
- Utilise des modèles de composants écrits en SystemC
 - ◆ processeur
 - ◆ interconnexion
 - ◆ périphériques
 - ◆ ...
- Utilisation d'une chaîne de compilation standard (GNU)

Organisation logicielle

- Système d'exploitation multithread : `Mutek`
 - ◆ Gestion de l'API de thread Posix
 - ◆ Gestion des interruptions
 - ◆ Commutation de contexte rapide
- Logiciel cross-compilé avec `gcc`
- Intégration du système avec l'applicatif à l'édition des liens

Exemple d'une plateforme soclib



Introduction

Processeurs embarqués

introduction au MSP 430

Présentation des SoC

- Qu'est ce qu'un SoC
- Exemple SoC: appareil photo numérique
- Autre exemple: iPaQ H5500
- Architecture carte iPaQ H5500
- Architecture S3CA400A01 (companion chip)
- Critères de qualité de conception d'un SOC
- Choix en fonction des contraintes
- Méthodologie de conception
- SocLib
- Organisation logicielle
- Exemple d'une plateforme soclib

