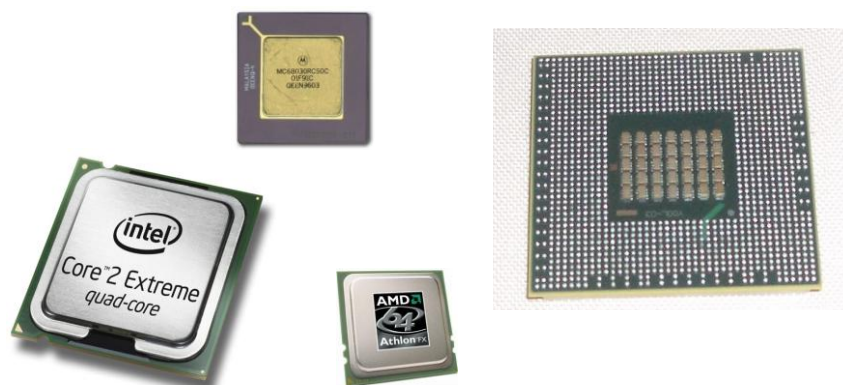




Architecture des systèmes à processeurs – IUT GEII (ISI-II2) -2

Christophe BLANC
www.christophe-blanc.info
IUT de Montluçon
Département Génie Electrique et Informatique Industrielle



Architecture des systèmes à processeurs

INTRODUCTION AUX MICROPROCESSEURS

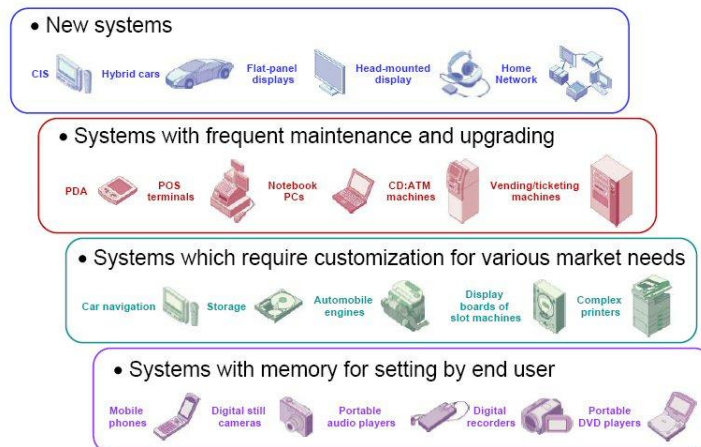
Sommaire

- Le domaine visé, les systèmes embarqués
- Du microprocesseur au microcontrôleur
- Architectures classiques mono processeur
 - Modèle de Harvard (Aiken 1943)
 - Modèle de Von Neumann
- Étude d'une architecture minimum 8 bits
- Déroulement de l'algorithme
- Unité arithmétique et logique
- Modèle de programmation de la famille M32C80
- Registres

I.U.T. Montluçon — Département Génie Electrique et Informatique Industrielle — 3

Le domaine visé : les systèmes embarqués

La plus grande partie des systèmes informatiques utilisés de nos jours sont des systèmes embarqués : téléphones mobiles, horloges, baladeurs, GPS, automobile, ...



I.U.T. Montluçon — Département Génie Electrique et Informatique Industrielle — 4

Le domaine visé : les systèmes embarqués

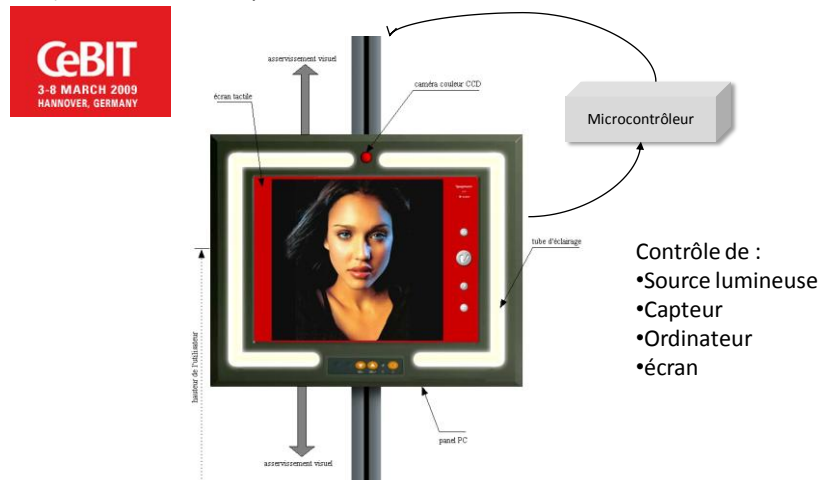
Les systèmes embarqués se démarquent des systèmes informatiques traditionnels selon plusieurs aspects :

- Contraintes de taille (intégration), de consommation électrique (autonomie) et de coûts.
- Dédiés à une tâche précise. Taille des programmes et de la mémoire modestes.
- Doivent communiquer avec des dispositifs d'entrées sorties (IO) : boutons, LED, moteurs électriques, relais, ...
- N'ont parfois aucun dispositif d'interface homme-machine (IHM) : ni clavier, ni écran, ni disques, ni imprimante, ...

I.U.T. Montluçon — Département Génie Electrique et Informatique Industrielle — 5

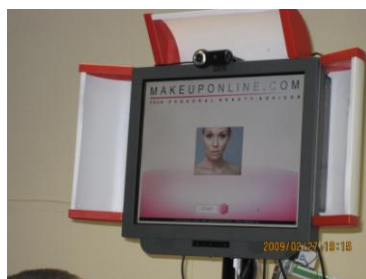
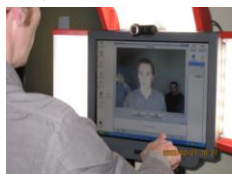
Le domaine visé : les systèmes embarqués

Un exemple : borne interactive asservi – Prototype présenté au Cebit 2009 (3-8 mars), Hanovre, Germany.



Le domaine visé : les systèmes embarqués

Un exemple : borne interactive asservi – Prototype présenté au Cebit 2009 (3-8 mars) , Hannover, Germany.



Du microprocesseur au microcontrôleur

Le processeur est l'élément central d'un système informatique : il interprète les instructions et traite les données d'un programme. Il a besoin de certains éléments externes pour fonctionner :

- Une horloge pour le cadencer
- De la mémoire pour stocker les variables durant l'exécution du programme (mémoire vive RAM) et le programme d'une mise sous tension à l'autre (mémoire morte ROM). Si l'on conçoit un système dédié à une tâche bien particulière (ce qui est généralement le cas des systèmes embarqués), le programme n'est pas amené à changer. Il peut donc être stocké dans une mémoire morte (ROM) ;
- des périphériques (pour interagir avec le monde extérieur).

Du microprocesseur au microcontrôleur

Ces éléments ont besoin de 3 bus pour fonctionner :

- le **bus d'adresse** qui permet au microprocesseur de sélectionner la case mémoire ou le périphérique auquel il veut accéder pour lire ou écrire une information (instruction ou donnée) ;
- le **bus de données** qui permet le transfert des informations entre les différents éléments ; ces informations seront soit des instructions, soit des données en provenance ou à destination de la mémoire ou des périphériques ;
- le **bus de contrôle** qui indique si l'opération en cours est une lecture ou une écriture, si un périphérique demande une interruption pour faire remonter une information au processeur, etc.

I.U.T. Montluçon — Département Génie Electrique et Informatique Industrielle — 9

Du microprocesseur au microcontrôleur

Traditionnellement, ces composants sont intégrés dans des circuits distincts. Le développement d'un tel système à base de microprocesseur se trouve donc pénalisé par (liste non exhaustive) :

- la nécessité de prévoir l'interconnexion de ces composants (bus, câblage, nappes de connexion) ;
- la place occupée physiquement par les composants et les moyens d'interconnexion ;
- la consommation énergétique ;
- la chaleur dégagée ;
- le coût financier.

I.U.T. Montluçon — Département Génie Electrique et Informatique Industrielle — 10

Du microprocesseur au microcontrôleur

Les **microcontrôleurs** améliorent l'intégration et le coût (lié à la conception et à la réalisation) d'un système à base de microprocesseur en rassemblant ces éléments essentiels dans un seul circuit intégré. Un microcontrôleur est donc un composant autonome, capable d'exécuter le programme contenu dans sa mémoire morte dès qu'il est mis sous tension. Selon les modèles et les conditions de fonctionnement, les microcontrôleurs peuvent avoir besoin de quelques composants externes (quartz, quelques condensateurs, parfois une ROM), mais ceci reste très limité.

I.U.T. Montluçon — Département Génie Electrique et Informatique Industrielle — 11

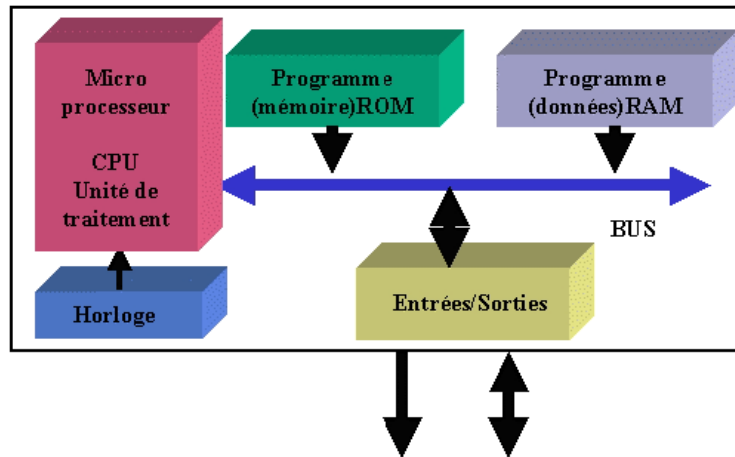
Architectures classiques mono processeur

Les trois éléments fondamentaux d'un système à base de microprocesseur sont:

- **Le microprocesseur**
Il exécute l'instruction qu'il a lue dans la mémoire. C'est le "cerveau" du micro-ordinateur.
- **La mémoire**
Elle stocke et restitue des informations sous forme de mots binaires (1 bits / 4 bits / 8 bits / 16 bits) **ex** : (01011101)_b : mot de 1 octet = (5D)_H = 5DH.
- **Les coupleurs d'entrées/sorties**
Ils servent de "bras" au micro-ordinateur. Ils gèrent l'interface entre le micro-ordinateur et l'extérieur (constitué de périphériques : imprimante, clavier, écran, moteur).

I.U.T. Montluçon — Département Génie Electrique et Informatique Industrielle — 12

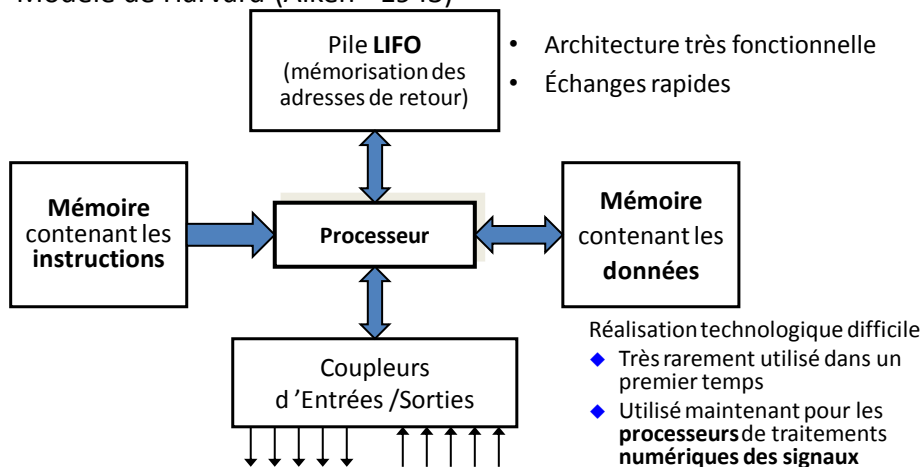
Architectures classiques mono processeur



I.U.T. Montluçon — Département Génie Electrique et Informatique Industrielle — 13

Architectures classiques mono processeur

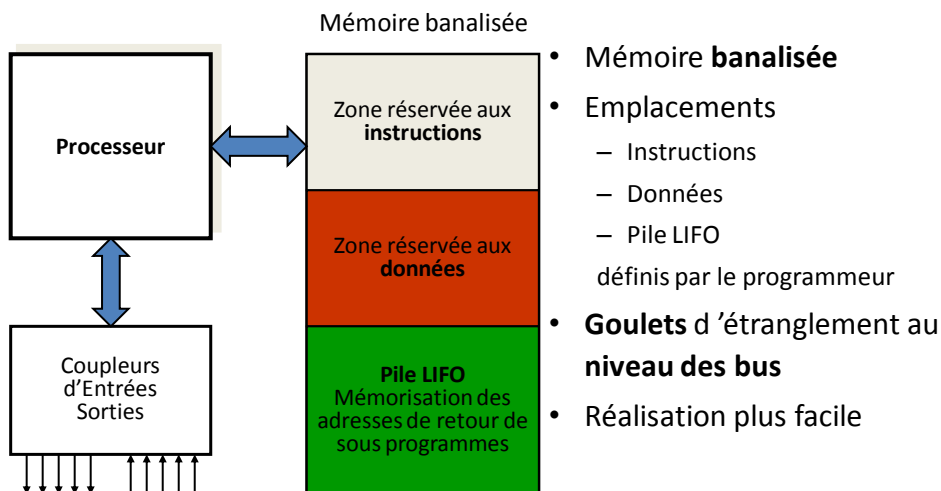
Modèle de Harvard (Aiken - 1943)



I.U.T. Montluçon — Département Génie Electrique et Informatique Industrielle — 14

Architectures classiques mono processeur

Modèle de Von Newmann

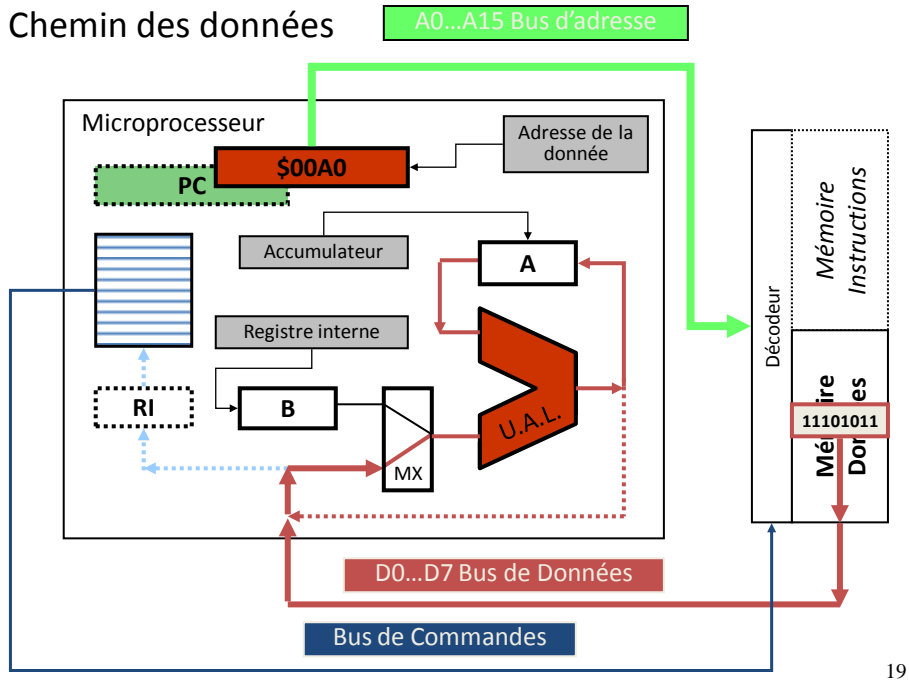


I.U.T. Montluçon — Département Génie Electrique et Informatique Industrielle — 15

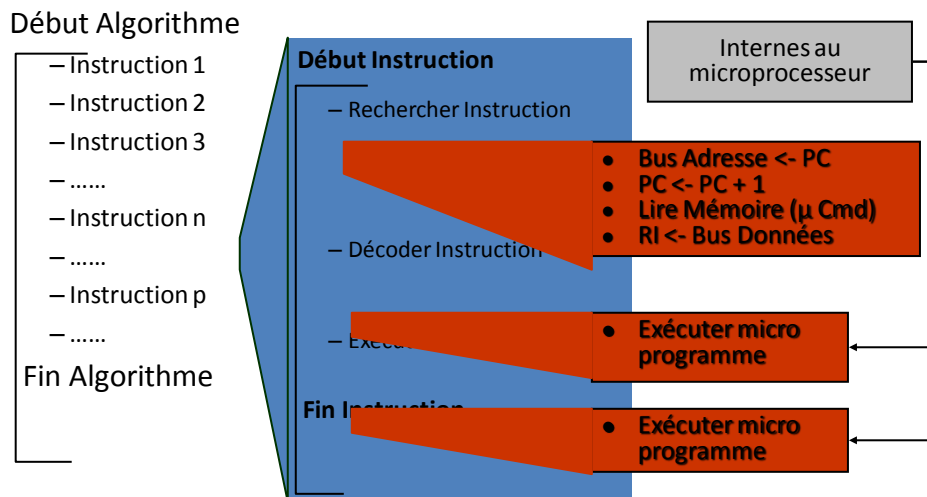
Étude d'une architecture minimum 8 bits

- Un microprocesseur comprend :
 - Une unité de commande (qui assure le séquençement)
 - Le Compteur Ordinal (PC - Program Counter)
 - Contient l'**adresse de la prochaine instruction à exécuter**
 - Un registre d'instruction (IR - Instruction Register)
 - Contient le code de l'instruction à exécuter
 - Une Unité Arithmétique et Logique (U.A.L.)
 - Un registre Accumulateur
 - Registre **servant de Source et de Destination**
 - Des registres internes
 - Équivalent à de la mémoire mais plus rapide et en nombre limité

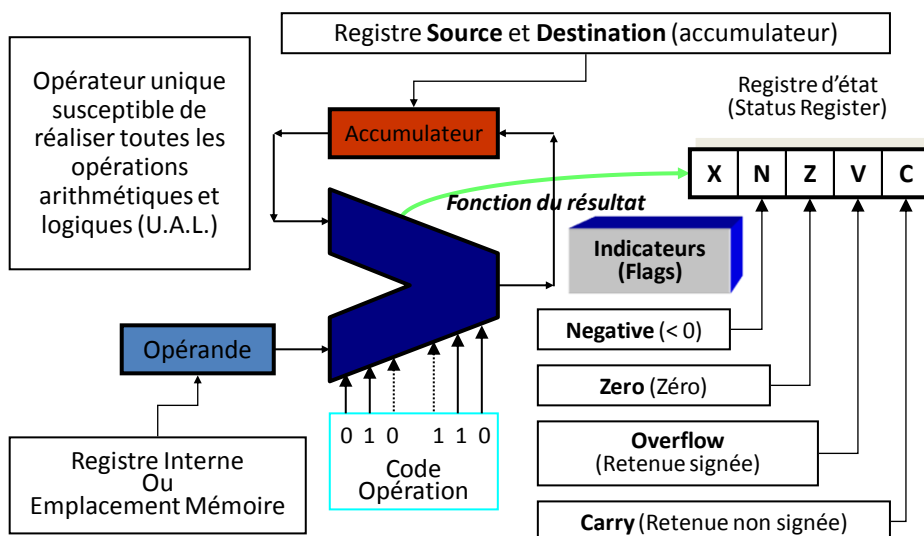
I.U.T. Montluçon — Département Génie Electrique et Informatique Industrielle — 16



Déroulement de l'algorithme



Unité arithmétique et logique



I.U.T. Montluçon — Département Génie Electrique et Informatique Industrielle — 21

Classification des registres

- Registre de donnée
 - Équivalent à une position mémoire
 - Interne au microprocesseur
 - Nombre limité
 - Temps d'accès très rapide (lié à la fréquence d'horloge du μP)
- Registre d'adresse
 - Contient l'adresse d'une donnée
 - Équivalent à un pointeur d'adresse en C, C++ (**sans typage**)
- Registre pointeur de pile
 - Registre d'adresse qui gère une zone mémoire (Pile)
 - Politique de type **LIFO** (Last In, First Out)

I.U.T. Montluçon — Département Génie Electrique et Informatique Industrielle — 22

Modèle de programmation de la famille Renesas M32C80

Everywhere you imagine.



<http://america.renesas.com/>
<http://eu.renesas.com/>



Applications Clefs :

- Audio
- Cameras
- Office Equipment
- Communication/Portable Devices

Modèle de programmation de la famille Renesas M32C80

Everywhere you imagine.



Caractéristiques:

- Fréquence horloge 32MHz : 1 cycle correspond à 31.3ns
- Dvlpmt efficace en c et/ou assembleur (108 instructions)
- Opérations registre/registre
- Opérations registre/mémoire
- Opérations mémoire/mémoire
- Opérations arithmétiques
- Calcul arithmétique rapide (un cycle d'horloge pour effectuer certaines opérations)

Modèle de programmation de la famille Renesas M32C80

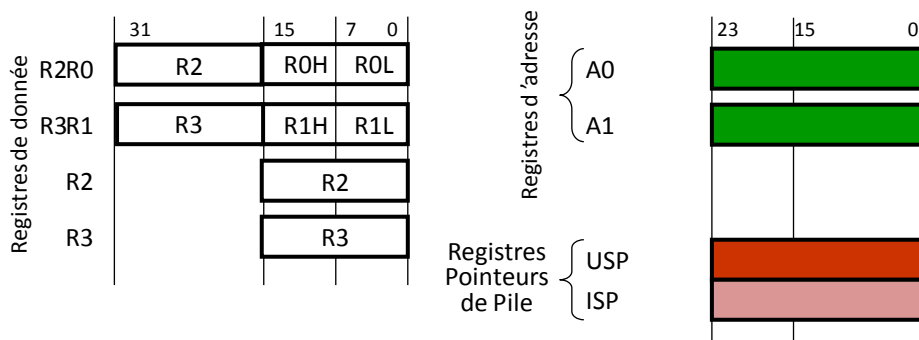
Everywhere you imagine.



■ Speed performance (types incorporating a multiplier, operating at 32 MHz)

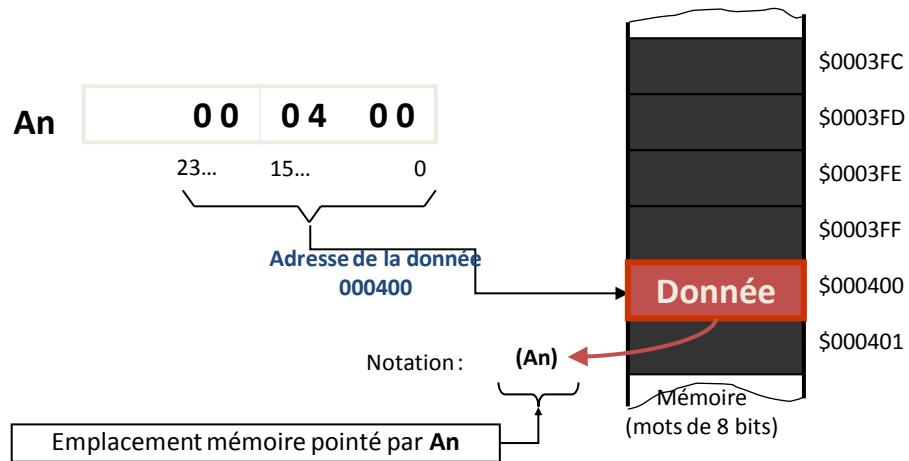
	Cycle	Execution Time
Register-register transfer	1	31.3 ns
Register-memory transfer	1	31.3 ns
Register-register addition/subtraction	1	31.3 ns
8 bits x 8 bits register-register operation	3	93.8 ns
16 bits x 16 bits register-register operation	3	93.8 ns
16 bits / 8 bits register-register operation	18	562.5 ns
32 bits / 16 bits register-register operation	18	562.5 ns

Modèle de programmation de la famille Renesas M32C80



Registre d'adresse

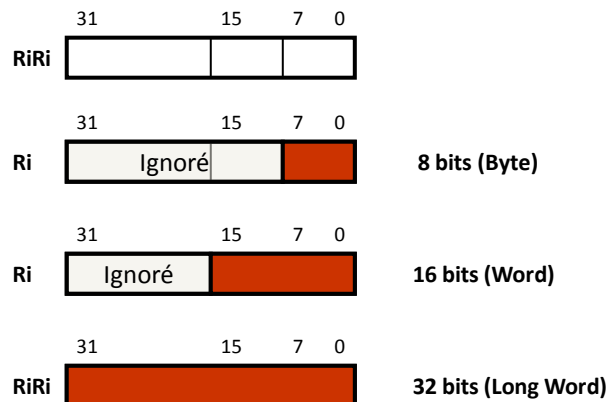
- ☒ Il permet de définir l'adresse d'une donnée



I.U.T. Montluçon — Département Génie Electrique et Informatique Industrielle — 27

Registre de donnée

- Taille des registres : relation entre taille et le nombre de bits



I.U.T. Montluçon — Département Génie Electrique et Informatique Industrielle — 28

Registre des indicateurs (flag register)

- Après instruction de transfert vers **Registre de Donnée**
 - Indicateurs **N** et **Z** positionnés en fonction du **contenu** du registre
 - Indicateurs **V** et **C** positionnés à **0**
- Exemple : **MOV.L:format R3R1, R2R0** **MOV.X:format R1, R0**
– Contenu de R2R0/R0 après exécution de l'instruction

