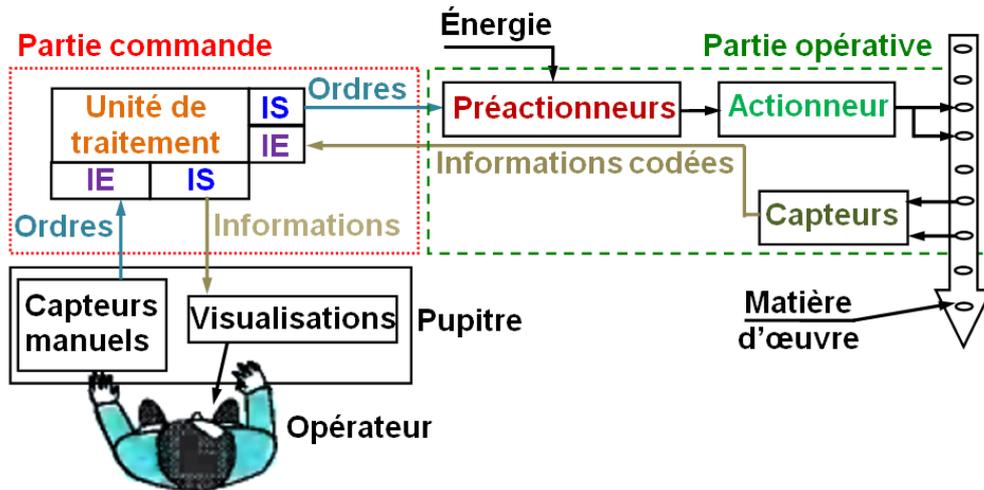


1- STRUCTURE GÉNÉRALE D'UN SYSTÈME AUTOMATISÉ :

Un système automatisé se compose d'une partie commande (unité de traitement et préactionneurs), une partie opérative (actionneurs, capteurs et matière d'œuvre) et un pupitre (boutons de commandes divers, signalisations, alarmes, écrans, etc.) permettant le dialogue avec l'opérateur.



1.1- Unité de traitement :

C'est le cerveau de l'automatisme. Les interfaces (IE = Interface d'Entrée, IS = Interface de Sortie) sont employés pour l'échange des données avec les autres éléments ainsi que la protection de l'unité. **Exemples :** Ordinateur ; processeur ou microprocesseur ; automate programmable (API) ; Séquenceurs.

1.2- Préactionneurs :

Ils représentent les gares de triage de l'énergie. Celle-ci est canalisée vers les actionneurs sur ordre de l'unité de traitement. **Exemples :** Distributeurs ; Contacteurs ; Variateurs.

1.3- Actionneurs :

Ce sont les muscles et les mains. Ils reçoivent leur énergie (électricité, air comprimé, etc.) par l'intermédiaire des préactionneurs. **Exemples :** Vérins ; moteurs ; robots ; vannes ; électro-aimants.

1.4- Capteurs :

C'est le service de surveillance et de renseignement. Ils contrôlent, mesurent, surveillent et informent l'unité de traitement sur l'état et l'évolution de l'automatisme.

Exemples : Interrupteur de position ; Détecteurs de proximité ; Compteurs ; Système de pesage ; Manomètres ; Thermomètres ; Dynamomètres ; Tachymètre.

2- PRINCIPALES CATÉGORIES DE SYSTÈMES AUTOMATISÉS :

3.1- Système combinatoire :

À une combinaison des entrées correspond une seule combinaison des sorties (**logique combinatoire**). Ces systèmes n'utilisent aucun mécanisme de mémorisation ; ils n'ont pas de mémoire. Les outils utilisés pour les concevoir sont l'**algèbre de Boole** ; **les tables de vérité** ; **les tableaux de Karnaugh** ; **Logigrammes** ; **Chronogrammes**.

2.2- Système séquentiel :

Un système séquentiel est un système qui évolue par étapes, contrairement aux systèmes combinatoires, une même combinaison des variables d'entrée peut donner des résultats différents, en fonction de l'évolution du système dans le temps.

Un système séquentiel possède donc une mémoire qui enregistre les étapes d'évolution du système. Le **GRAFCET** est l'un des principaux outils de conception de ces systèmes.

2.3- Système asservi :

Dans ce système, on désire que la sortie suive avec précision les variations de l'entrée et ceci avec un temps de réponse réduit. **Exemples :** Direction assistée d'automobile ; Contrôle de température.

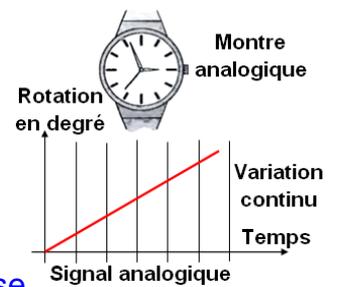
3- SIGNAL NUMÉRIQUE ET SIGNAL ANALOGIQUE :

3.1- Signal analogique :

Si à une grandeur d'entrée, ou signal, on fait correspondre un grandeur de sortie qui lui est directement proportionnelle (allure analogique), la grandeur de sortie est dite analogique.

Les grandeurs analogiques varient de façon continue à l'intérieur d'une gamme ou d'un intervalle de valeurs.

Exemples : Position (angle de rotation) de l'aiguille d'un compteur de vitesse d'automobile proportionnelle à la vitesse du véhicule ; montre analogique à aiguilles ; microphone ; thermostat à bilames.



3.2- Signal numérique, ou digital :

Il a pour base deux états logiques (marche ou arrêt, oui ou non, etc.) représentés par les chiffres 1 ou 0.

Exemples : Montre digitale ; données des systèmes informatiques...

Niveau logique 1	Niveau logique 0
Tout	Rien
Vrai	Faux
Oui	Non
Marche	Arrêt
Fermé	Ouvert
Haut	Bas
On	Off
Bon	Mauvais



Les automates programmables et tous les systèmes à microprocesseurs ne manipulent que ces deux grandeurs matérialisées par des signaux électriques :

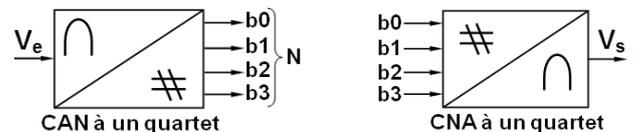
le courant passe = 1 ; le courant ne passe pas = 0.

Remarques :

- Les signaux numériques sont plus faciles à exploiter par les systèmes informatiques usuels, qui sont eux-mêmes basés sur des données ou des codes numériques. De plus, et contrairement aux informations analogiques, ils permettent de transférer des données sans brouillage ou perturbation.
- Afin de rendre un signal analogique exploitable (par un système informatique...) on utilise un

Convertisseur Analogique Numérique (CAN) qui échantillonne le signal suivant des intervalles réguliers et le convertit en un signal numérisé.

L'opération inverse est également possible au moyen d'un **Convertisseur Numérique Analogique (CNA)**.



3.3- Vocabulaire :

- Ces informations à deux états et leur manipulation définissent le système **binaire**.

- L'information unitaire est appelée le **bit** (binary digit = chiffre binaire).

- Les bits peuvent être regroupés en mots.

- Le mot le plus courant est l'**octet** qui regroupe **8 bits** (un octet = un byte).

- Dans un octet, le bit le plus à droite est appelé **bit de poids faible** et celui le plus à gauche est appelé **bit de poids fort**.

Bit de poids fort = 2^7 Bit de poids faible = 2^0

Un octet :	b7	b6	b5	b4	b3	b2	b1	b0
Exemple :	1	1	0	1	1	0	0	1

- Le **kilo-octet** (Ko ou KB) = 1024 octets ;

- Le **méga-octet** (Mo ou MB) = 1024 Ko ;

- Le **giga-octet** (Go ou GB) = 1024 Mo ;

- Le **téra-octet** (To ou TB) = 1024 Go.

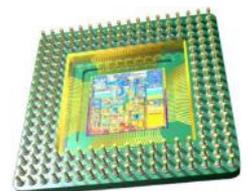


3.4- Codage et représentation des nombres :

Le binaire est également un outil de codage des grandeurs.

Il permet, entre autre, de représenter les nombres entiers. Ainsi un octet (ou tout autre mot de taille différente) peut représenter un nombre et être manipulé en tant que tel.

Les microprocesseurs manipulent des octets, mais également des mots bien plus grands de 16, 32 ou 64 bits.



4- REPRÉSENTATION BINAIRE DES INFORMATIONS :

L'ordinateur est un **calculateur** (computer) qui ne reconnaît que **deux états**, **0** et **1**, c'est le **bit** qui est soit ouvert soit fermé. L'ordinateur, constitué de circuits logiques, travaille en **base 2** alors que les résultats exploitables doivent être donnés en 10 symboles (0, 1, 2, 3, 4, 5, 6, 7, 8, 9) **base 10**.

Compter en base 2 (en binaire) est difficile.

Exemple : le nombre **décimal 300_D** se lirait **0100101100_B**.

Nécessité d'une nouvelle **base 16** dite **hexadécimal** comprenant 16 symboles numériques : (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F).

En **base 16**, un symbole représente **4 bits** d'où nouvelle conversion.

Exemple : le nombre **décimal 300_D** se lira **12C_H**.

4.1- Tableau d'équivalences :

Décimal	Binaire	Hexadécimal	BCD	Code Gray
0	0000	0	0000	0000
1	0001	1	0001	0001
2	0010	2	0010	0011
3	0011	3	0011	0010
4	0100	4	0100	0110
5	0101	5	0101	0111
6	0110	6	0110	0101
7	0111	7	0111	0100
8	1000	8	1000	1100
9	1001	9	1001	1101
10	1010	A	0001 0000	1111
11	1011	B	0001 0001	1110
12	1100	C	0001 0010	1010
13	1101	D	0001 0011	1011
14	1110	E	0001 0100	1001
15	1111	F	0001 0101	1000
16	1 0000	10	0001 0110	1 1000
17	1 0001	11	0001 0111	1 1001
18	0 1001	12	0001 1000	1 1011
...				

Axe de réflexion du bit de droite

Axe de réflexion des deux bits situés à droite

Axe de réflexion des trois bits situés à droite

Axe de réflexion des quatre bits situés à droite

4.2- Code binaire codé décimal (BCD) :

Un nombre décimal est une association de chiffres : unités, dizaines, centaines...

Pour représenter un nombre décimal de **0 à 9**, il faut un **quartet (4 bits)**.

Le **codage BCD** consiste à utiliser un quartet pour les unités, un autre pour les dizaines, un autre pour les centaines... **Exemple :** $N_{10} \Rightarrow 192 \Rightarrow 0001 \quad 1001 \quad 0010$

$$1 \cdot 10^2 \quad 9 \cdot 10^1 \quad 2 \cdot 10^0$$

4.3- Généralités (exemple d'écriture) :

Base décimal $\Rightarrow 200_D$ ou $N_{10} \Rightarrow 200$
 Base binaire $\Rightarrow 1100\ 1000_B$ ou $N_2 \Rightarrow 1100\ 1000$ ou $\% 1100\ 1000$
 Base hexadécimal $\Rightarrow C8_H$ ou $N_{16} \Rightarrow C8$ ou $\$ C8$
 Base BCD $\Rightarrow 0001\ 0010_{BCD}$ ou $N_{BCD} \Rightarrow 0001\ 0010$

4.4- Conversion binaire-décimale :

Principe :

Pour réaliser la conversion d'une base binaire en base décimale, il faut tout d'abord ordonner l'écriture du code binaire en écrivant à gauche le bit de poids fort et le bit de poids faible à droite. Chaque symbole représente une puissance de **2** qui va s'incrémenter de **1** en **1** en partant de 0.

Mode opératoire :

← Bit de poids fort Bit de poids faible →

$$1100\ 1000_B \Rightarrow \text{bit 7} \quad \text{bit 6} \quad \text{bit 5} \quad \text{bit 4} \quad \text{bit 3} \quad \text{bit 2} \quad \text{bit 1} \quad \text{bit 0}$$

$$N_2 = 1100\ 1000 \Rightarrow 1 \cdot 2^7 + 1 \cdot 2^6 + 0 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0$$

$$N_2 = 1100\ 1000 \Rightarrow 128 + 64 + 0 + 0 + 8 + 0 + 0 + 0$$

$$N_2 = 1100\ 1000 \Rightarrow 200_D$$

Binaire	Décimal	Binaire	Décimal
0101 1010	90	1101 1010 1010	3 498
1000 0000	128	1 0111 0011 1010	5 946
1111 1111	255	1011 0001 1011 1011	45 499

Exemples de conversion :

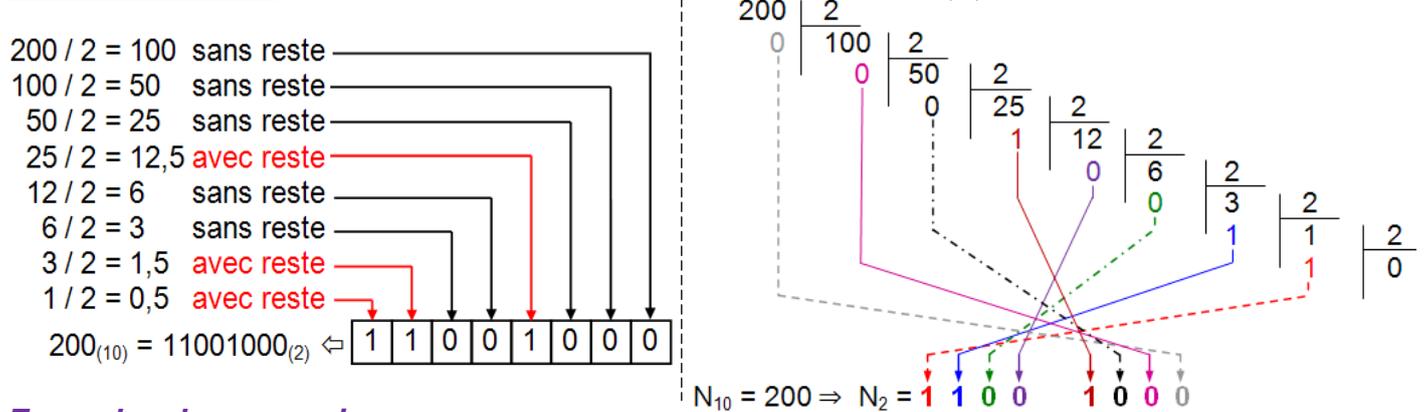
4.5- Conversion décimale-binaire :

Principe :

On trouve le nombre N en **base 2** en réalisant une série de divisions par 2.
Le reste de la première division représente le bit de poids faible alors que le reste de la dernière division représente le bit de poids fort.

- Si le résultat d'une division est un entier, le bit correspondant vaut 0.
- Si le résultat d'une division est décimal, le bit correspondant vaut 1.

Mode opératoire : On souhaite convertir en binaire le nombre $200_{(10)}$



Exemples de conversion :

Décimal	Binaire	Décimal	Binaire
220	1101 1100	2 000	111 1101 0000
255	1111 1111	10 123	10 0111 1000 1011

Remarque :

Utiliser la méthode simple ci-dessous pour convertir le binaire en décimale et inversement :

Conversion binaire-décimale									Conversion décimale-binaire									
256	128	64	32	16	8	4	2	1	108	256	128	64	32	16	8	4	2	1
1	0	0	1	0	1	1	0	0										

4.6- Conversion binaire-hexadécimale :

Principe :

La conversion du binaire en hexadécimale s'effectue simplement. Il faut regrouper le mot binaire en quartet en partant du bit de poids faible et donner pour chaque quartet le chiffre hexadécimal correspondant à l'aide de la table de correspondance.

Mode opératoire : $N_2 = 11\ 0111\ 1011\ 1010 = \underline{0011}\ \underline{0111}\ \underline{1011}\ \underline{1010}$
↓ ↓ ↓ ↓
3 7 B A

$N_2 = 11\ 0111\ 1011\ 1010 \Rightarrow N_{16} = 37BA$

Exemples de conversion :

Binaire	Hexadécimal	Binaire	Hexadécimal
0101 1010 1111	5AF	11 0110 1101 0010 1001 1010	36D29A
1 0000 0001 1001	1019	101 1111 0110 1001 0000 0111	5F6907
111 1111 1101 0101	7FD5	10 1110 0111 0101 1001 0011 0111	2E75937

4.7- Conversion hexadécimale-binaire :

Principe :

La conversion hexadécimale en binaire s'effectue simplement. Il faut utiliser la table de conversion et faire correspondre à chaque symbole hexadécimal le quartet binaire.

Mode opératoire : $N_{16} = F5DC = \underline{F}\ \underline{5}\ \underline{D}\ \underline{C}$
↓ ↓ ↓ ↓
1111 0101 1101 1100

$N_{16} = F5DC \Rightarrow N_2 = 1111\ 0101\ 1101\ 1100$

Exemples de conversion :

Hexadécimal	Binaire	Hexadécimal	Binaire
200	10 0000 0000	1A2C5	1 1010 0010 1100 0101
1234	1 0010 0011 0100	2F8ED8	10 1111 1000 1110 1101 1000
ABCD	1010 1011 1100 1101	118D4F4	1 0001 1100 1101 0100 1111 0100

4.8- Conversion hexadécimale-décimale :

Principe :

Pour réaliser la conversion d'une base hexadécimale vers une base décimale, procéder de la même façon que la conversion binaire-décimale.

Chaque symbole représente une puissance de 16 qui va s'incrémenter de 1 en 1 en partant de 0.

Mode opératoire :

$$\begin{aligned}
 0C4F2B_H &\Rightarrow C \cdot 16^4 + 4 \cdot 16^3 + F \cdot 16^2 + 2 \cdot 16^1 + B \cdot 16^0 \\
 N_{16} = 0C4F2B &\Rightarrow 12 \cdot 16^4 + 4 \cdot 16^3 + 15 \cdot 16^2 + 2 \cdot 16^1 + 11 \cdot 16^0 \\
 N_{16} = 0C4F2B &\Rightarrow 786\,432 + 16\,384 + 3\,840 + 32 + 11 \\
 N_{16} = 0C4F2B &\Rightarrow 806\,699_D
 \end{aligned}$$

Exemples de conversion :

Hexadécimal	Décimale	Hexadécimal	Décimale
0ABCD	43 981	12345	74 565
10000	65 536	0FFFFFF	1 048 575
200000	2 097 152	0FEDCBA	16 702 650

Remarque : Lorsqu'un nombre hexadécimal commence par une **lettre**, il convient de mettre un **0** devant ce nombre.

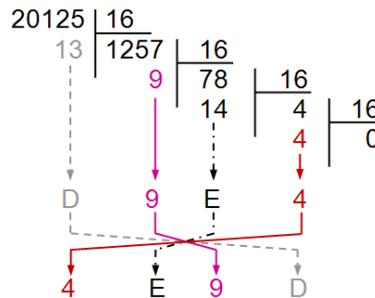
4.9- Conversion décimale-hexadécimale :

Principe :

On trouve le nombre N en **base 10** en effectuant une série de divisions par **16**. Le reste de la première division représente le chiffre de poids faible alors que le reste de la dernière division représente le chiffre de poids fort. Il convient de convertir le résultat ou le reste en hexadécimal.

Mode opératoire :

$$N_{10} = 20125$$



$$N_{10} = 20125 \Rightarrow N_{16} = 4E9D$$

Exemples de conversion :

Décimale	Hexadécimal	Décimale	Hexadécimal
420	01A4	320 000	04E200
65 535	0FFF	123 456 789	075BCD15