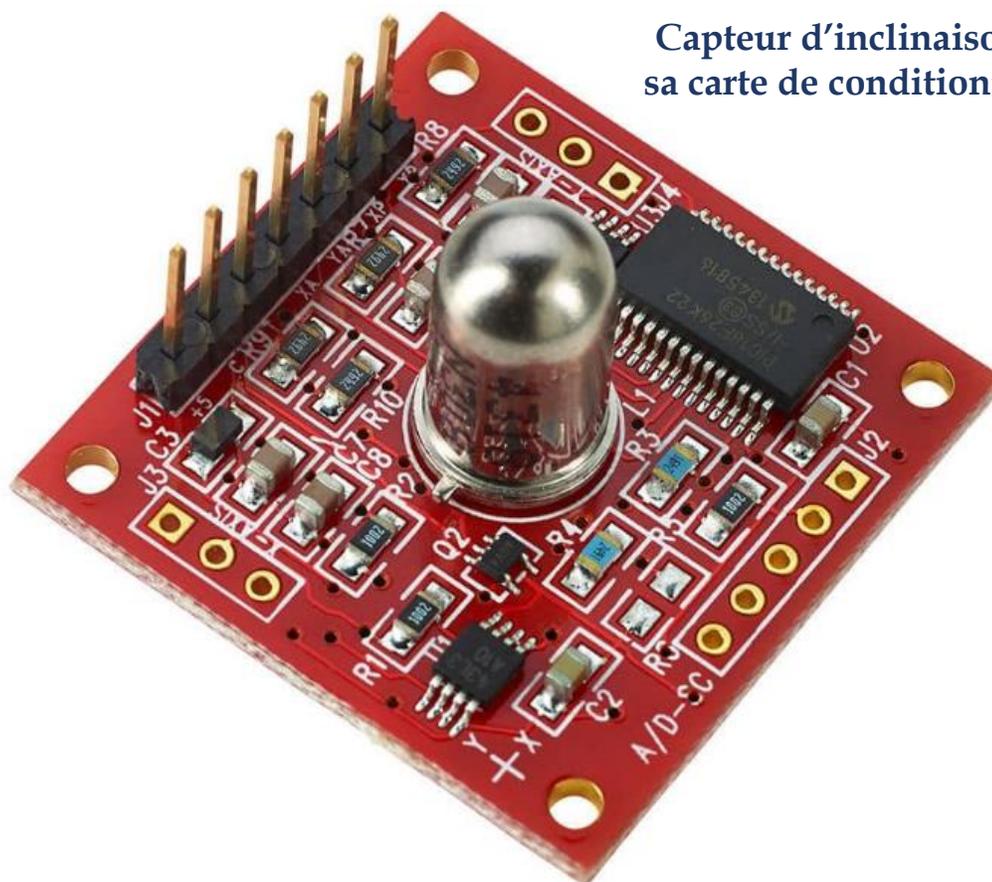


**PARTIE****1**

# La fonction ACQUERIR



Capteur d'inclinaison avec sa carte de conditionnement

## 1

**CAPTEURS****Mise en situation**

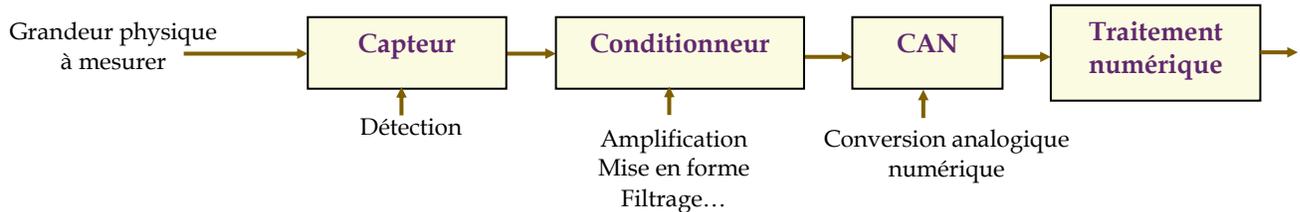
En automobile, tout devient électronique, l'injection, l'allumage, la climatisation, l'indicateur de vitesse du véhicule, la position des sièges, le freinage...

Tout cela est géré par des calculateurs à base de microprocesseur.

Cependant, pour fonctionner, ces calculateurs ont besoin d'informations qui sont prélevées sur le moteur, la boîte de vitesse, les roues... ; c'est le rôle des capteurs.

**Rappel de la chaîne d'acquisition**

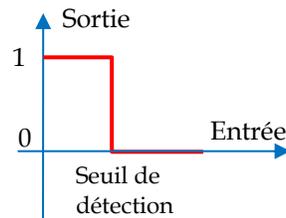
La chaîne d'acquisition d'un système technique permet de transformer une grandeur à mesurer en un signal électrique exploitable. Dans la chaîne d'acquisition, le capteur est l'élément sensible à la grandeur physique :



Selon la nature du signal de sortie, on peut classer les capteurs en 3 grandes catégories :

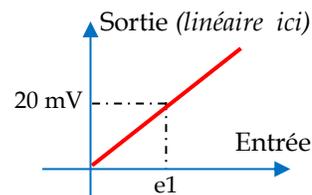
- Capteur **logique** ou Tout Ou Rien (TOR) : la sortie est binaire (information vraie ou fausse).  
Exemples : détecteur de fin de course, thermostat

Exemple de caractéristique de transfert d'un capteur logique



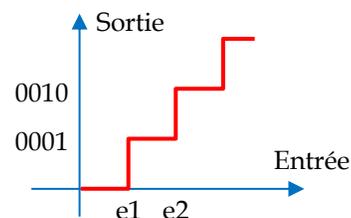
- Capteur **analogique** : la sortie varie de façon progressive (variation continue).  
Exemples : thermocouple, dynamo tachymétrique

Exemple de caractéristique de transfert d'un capteur analogique



- Capteur **numérique** : la sortie varie par échelons.  
Exemple : codeur absolu

Exemple de caractéristique de transfert d'un capteur numérique



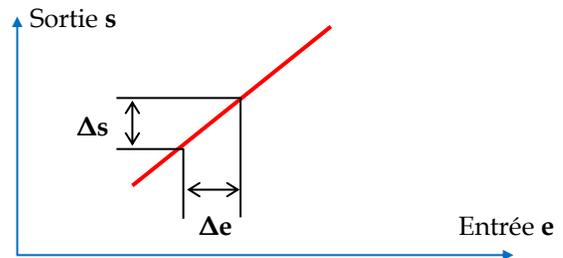
## Caractéristiques des capteurs

### ⇒ Sensibilité

C'est le rapport entre la variation de la grandeur de sortie et celle de la grandeur d'entrée  $S = \Delta s / \Delta e$ .

### ⇒ Linéarité

Un capteur est linéaire si sa caractéristique de transfert  $s = f(e)$  est linéaire donc si sa sensibilité est constante dans toute sa plage d'emploi.



⇒ Autres caractéristiques : étendue de mesure, précision, fidélité et rapidité.

## Les capteurs numériques

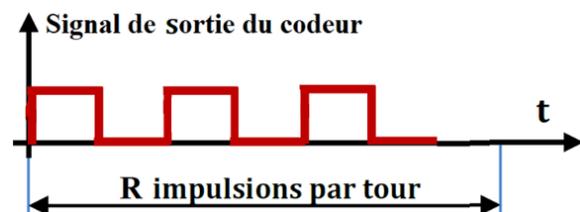
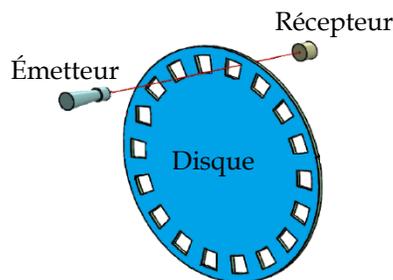
### 1. Codeur Incremental

L'élément principal est un disque portant une piste divisée régulièrement en secteurs alternativement opaques et transparents.

Autour de la piste, sont installés un émetteur et un récepteur de lumière.

Les impulsions émises par le codeur peuvent renseigner sur :

- La **position** ;
- La **vitesse** ;
- Le **sens de mouvement**.



Résolution :

$R$  = nombre de points/tour

Précision angulaire :

c'est l'angle de rotation qui correspond à une seule impulsion ; c'est donc la plus petite position angulaire détectée ; soit  $\theta = \frac{360^\circ}{R}$  ou  $\theta = \frac{2\pi}{R}$

Vitesse de rotation :

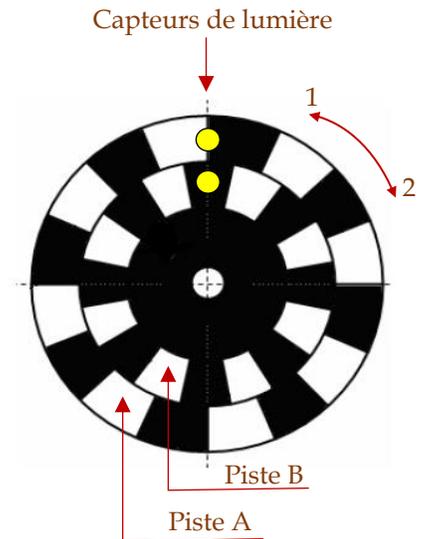
$N = 60 \cdot \frac{f}{R}$  (N en tr/min ; f : fréquence du signal de sortie)



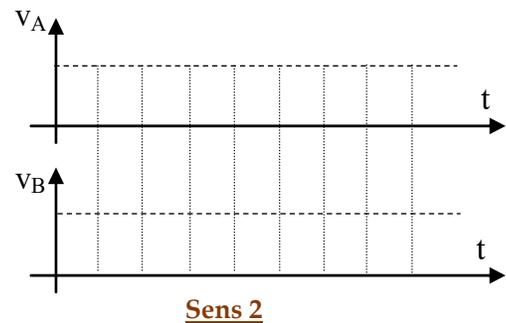
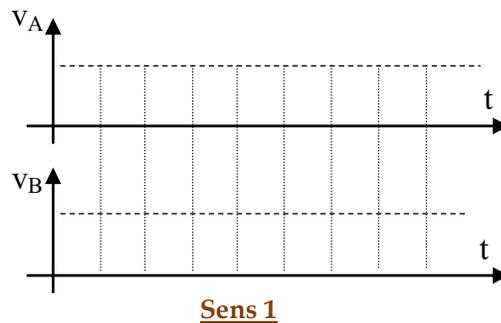
### Application 3 : détection de sens de rotation

**Principe** : Analyse du déphasage entre les signaux des voies A et B

Le codeur est équipé de 2 pistes A et B décalées comme l'indique la figure. Chacune des pistes est dotée d'un capteur de lumière à base d'émetteur-récepteur.

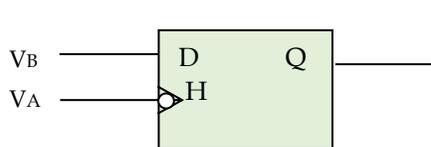


- Tracer l'allure des signaux  $v_A$  et  $v_B$  issus des capteurs A et B.



- On relie les 2 sorties  $v_A$  et  $v_B$  à une bascule D ; constater l'état de la sortie Q à chaque front descendant de l'horloge.

Rappel : pour une bascule D, à chaque front sur H, l'état de l'entrée D est transféré à la sortie Q.



|        | $V_A=H$ | $V_B = D$ | Q     |
|--------|---------|-----------|-------|
| Sens 1 |         | 0         | ..... |
| Sens 2 |         | 1         | ..... |

- Conclure

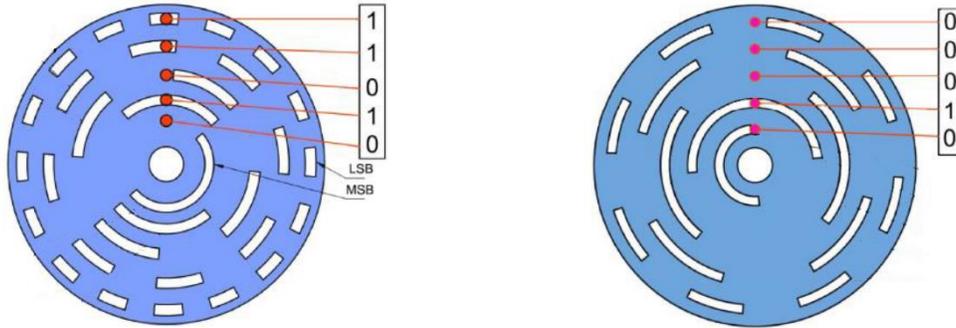
On obtient une information logique qui correspond au sens de rotation. En effet, la sortie Q se maintient toujours à :

- 0 pour le **premier sens**
- 1 pour le **deuxième sens**

**2. Codeur absolu**

Le codeur absolu génère le **code numérique** sur **n bits** correspondant à la **position** d'un mobile, moyennant :

- Un disque codé le plus souvent en Gray et comportant **n pistes** ;
- Un ensemble de **n capteurs** optiques (un capteur par piste).



**Résolution :**

$$R = 2^n \text{ positions/tour (nombre de positions possibles)}$$

**Précision angulaire :**

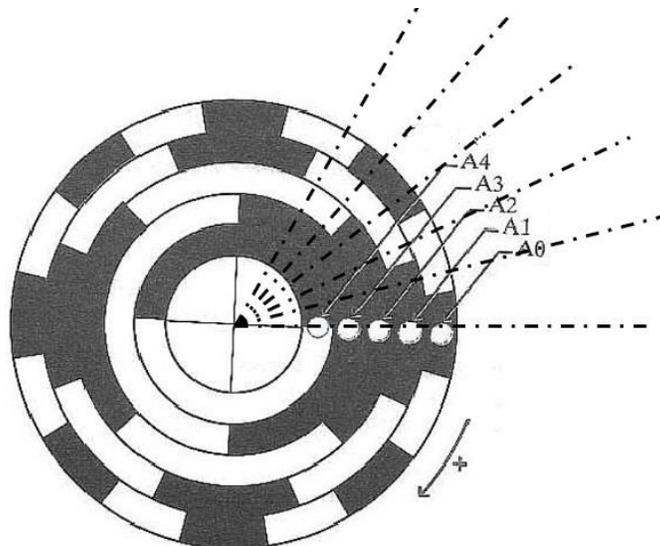
$$\theta = \frac{360^\circ}{2^n} \text{ (angle qui provoque un changement du code de sortie)}$$

**Exercice**

Pour le codeur absolu fourni ci-dessous, donner :

- Le nombre de pistes : .....
- La résolution : .....
- La précision en degrés : .....
- Les codes numériques correspondant aux angles 22,5°, 45° et 46° :

| Angle (°)        | 0         | 22,5  | 45    | 46    |
|------------------|-----------|-------|-------|-------|
| Sortie numérique | 0 0 0 0 0 | ..... | ..... | ..... |

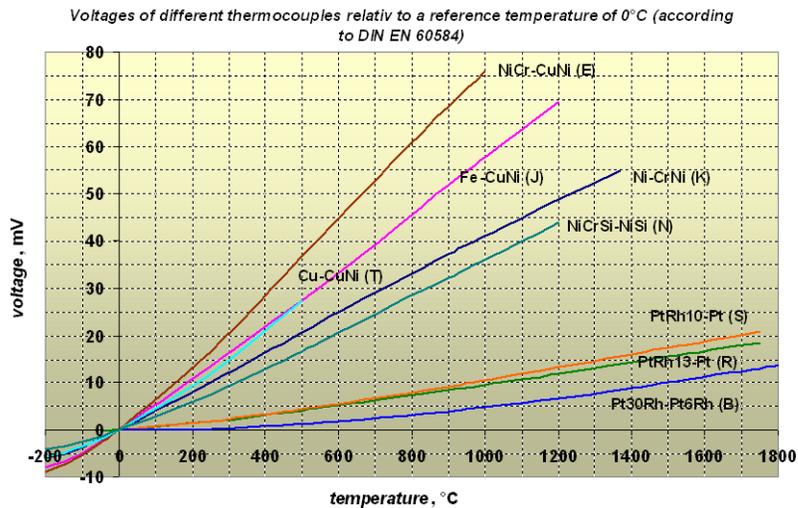
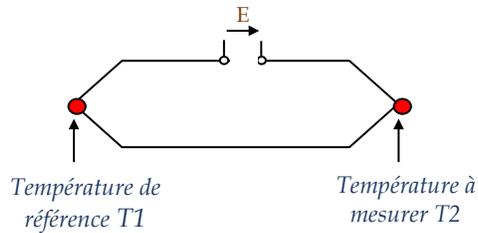


## Capteurs analogiques

### 1. Capteur de température

#### ⇒ Mesure par thermocouple

Un thermocouple est constitué de deux conducteurs de natures différentes reliés en 2 points. Une f.é.m  $E$ , fonction de la différence des températures  $T_1$  et  $T_2$ , apparaît entre ces deux points.



#### ⇒ Mesure par thermistance

Une thermistance est constituée d'un matériau semi-conducteur. Sa résistance varie en fonction de la température.

On distingue deux types de thermistances : les CTN et les CTP

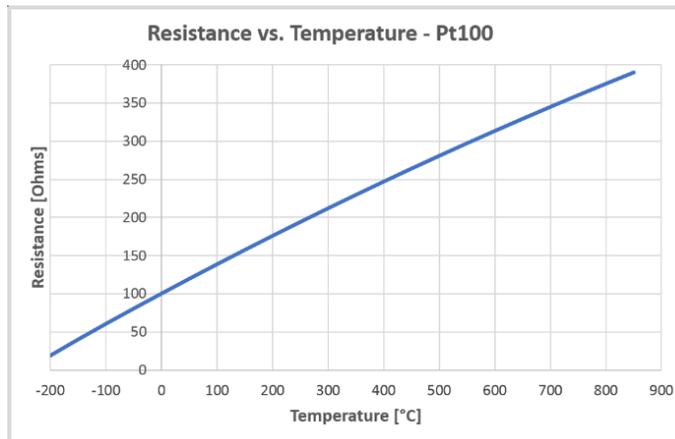
- Les CTN (Coefficient de Température Négatif) sont des thermistances dont la résistance diminue de façon uniforme avec la température.
- Les CTP (Coefficient de Température Positif) sont des thermistances dont la résistance augmente fortement avec la température.

⇒ Mesure par résistor PT100

C'est un résistor à coefficient de température positif. La loi de variation  $R = f(T)$  est parfaitement connue. C'est un standard de la mesure de température.

Constitué d'un film ou d'un fil de platine, sa résistance vaut  $100\ \Omega$  à  $0\ ^\circ\text{C}$  et  $138,5\ \Omega$  à  $100\ ^\circ\text{C}$ .

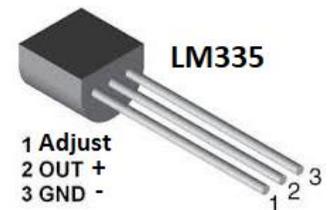
La sensibilité est de  $0,385\ \Omega/^\circ\text{C}$ .

⇒ Mesure par variation de tension Zener

Le LM335 est un composant ayant un comportement de diode Zener.

Sa tension de sortie, recueillie entre les broches + et -, est proportionnelle à la température :  $v_s = 10^{-2} \cdot T_K$  (Sensibilité =  $10\ \text{mV}/^\circ\text{K}$ ).

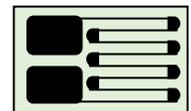
Le LM335 se présente sous forme d'un circuit intégré avec notamment un amplificateur intégré et un circuit de linéarisation.

2. Capteurs à jauges extensométriques

Ils permettent de déterminer les efforts dans les matériaux et sont à l'origine de capteurs de forces, couples, pressions....

Un fil électrique est collé à la surface du matériau. Lorsque celui-ci est soumis à un effort, il se déforme, le conducteur résistif également : sa résistance change.

Jauge



De manière à miniaturiser le capteur, le conducteur est déposé sur un substrat isolant et la piste décrit un certain nombre d'allers-retours.

3. Capteur de vitesse⇒ Dynamo tachymétrique

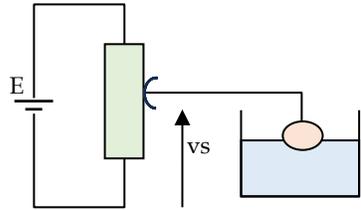
La dynamo tachymétrique est une petite génératrice à courant continu ; elle délivre une fem proportionnelle à la vitesse.



#### 4. Capteur de déplacement ou de position

⇒ Capteur résistif linéaire ou angulaire

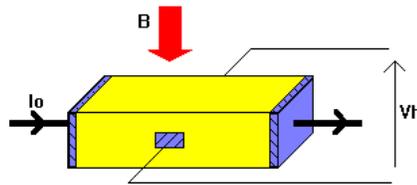
Essentiellement utilisé pour mesurer des déplacements linéaires ou angulaires (Jauge de carburant, niveau de cuve...), il utilise le principe du montage potentiométrique permettant d'obtenir une relation directe entre la tension de sortie  $v_s$  et le déplacement.



#### 5. Effet Hall

Si un courant  $I_0$  traverse un barreau conducteur immergé dans un champ magnétique  $B$  perpendiculaire à  $I_0$ , une tension  $V_h$  dite de Hall, proportionnelle au champ magnétique et au courant  $I_0$ , apparaît sur les faces latérales. Les capteurs à Effet Hall permettent de :

- Réaliser des capteurs de position et de vitesse sans contact, utilisés notamment dans l'automobile (boîte à vitesse, cardans,...) ;
- Mesurer les champs magnétiques (Teslamètres) ;
- Mesurer l'intensité des courants électriques (capteurs de courant à Effet Hall).



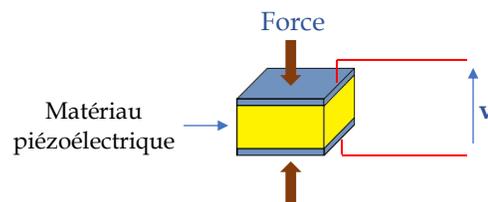
#### 6. Piézoélectricité

La piézoélectricité est la propriété que possèdent certains corps de :

- Générer une tension sous l'action d'une contrainte mécanique ;
- Se déformer sous l'effet d'une tension électrique (effet piézoélectrique inverse).

Le plus connu des matériaux piézoélectriques est sans doute le quartz, toujours utilisé aujourd'hui dans les montres pour générer des impulsions d'horloge.

Application : capteur de pression, d'accélération, de force...



## Exercices

## Exercice 1 : examen national 2021, session normale

L'ADCP est équipé d'un capteur acoustique dit transducteur. Ce dernier émet des ondes ultrasons qui sont réfléchies sur des particules en suspension entrainées par le courant d'eau (Figure 14).

La différence en fréquence des signaux émis et reçu dépend de la vitesse des courants marins : c'est l'effet Doppler.

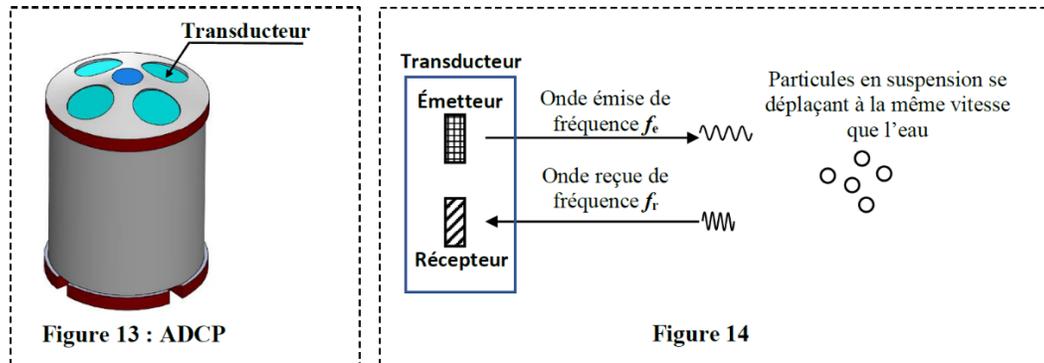


Figure 13 : ADCP

Figure 14

Le décalage Doppler  $\Delta f = f_r - f_e$  (différence entre la fréquence  $f_r$  de l'onde reçue et la fréquence  $f_e$  de l'onde émise) est donné par la relation :

$$\Delta f = \frac{2 \cdot f_e \cdot V}{c}$$

Où :  $\Delta f$  : décalage Doppler (en **Hz**) ;

$f_e$  : fréquence de l'onde émise (en **Hz**) ;

$V$  : vitesse des courants marins (en **m/s**) ;

$c$  : vitesse du son dans le milieu (en **m/s**).

La fréquence d'émission utilisée par le transducteur est  $f_e = 150 \text{ kHz}$ , ce dernier mesure un décalage Doppler  $\Delta f = 300 \text{ Hz}$ , on donne la vitesse du son dans le milieu  $c = 1500 \text{ m/s}$  :

**Question : 30.** calculer la vitesse  $V$  (en **m/s**) des courants marins. 1 pt

**Question : 31.** Calculer, en **kHz**, la fréquence  $f_r$  de l'onde reçue. 1 pt

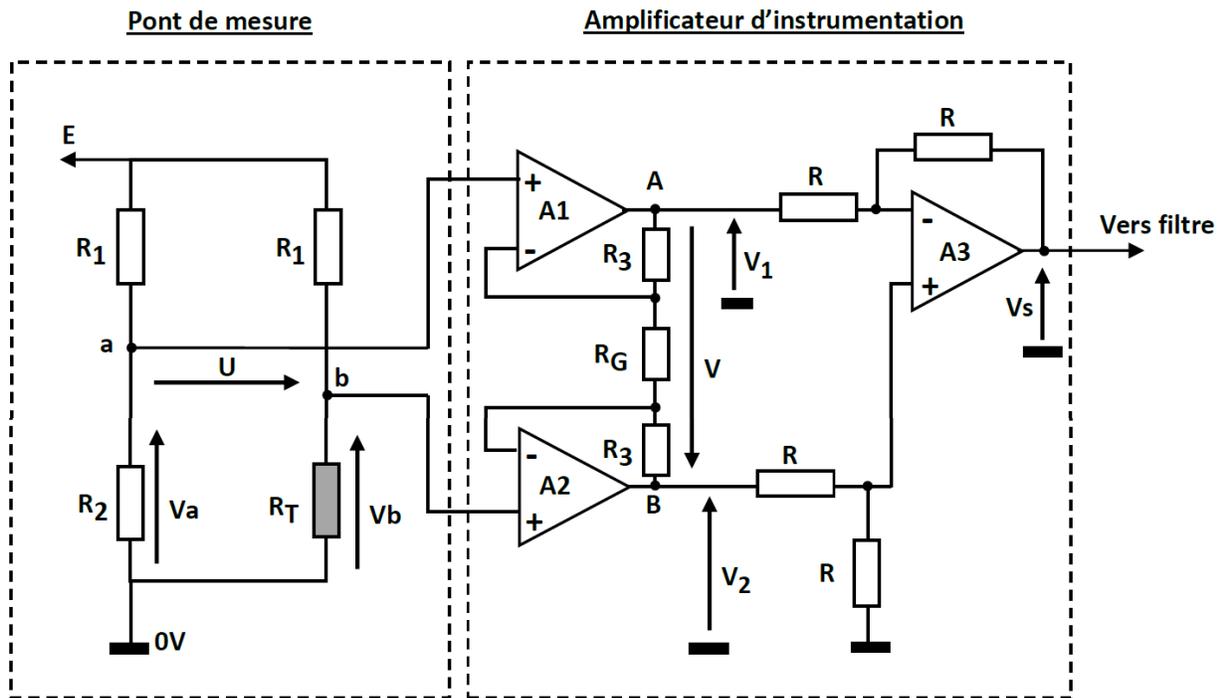
**Question : 32.** Pour un décalage Doppler  $\Delta f$  positif, préciser, en se référant au **DRES 03**, le sens de déplacement des particules 1 pt

La vitesse du son dans l'eau de mer n'est pas constante mais dépend de certains paramètres ambiants dont la température.

Ainsi, il faut renseigner la température de l'eau afin d'en tenir compte.

La mesure de la température se fait par un capteur à base d'une sonde **PT100** associée à un montage de conditionnement. Ce dernier se compose d'un pont de mesure (pont de Wheatstone), d'un amplificateur d'instrumentation et d'un filtre (figure ci-dessous).

Ce conditionneur fournit à sa sortie une tension pratiquement proportionnelle à la température.



Les amplificateurs opérationnels **A1**, **A2** et **A3** sont considérés parfaits.

### Pont de mesure

La sonde **PT100** possède une résistance **RT** qui varie avec la température **T** selon la loi :

$$R_T = R_0 + a \cdot T \quad (\text{avec } T \text{ en } ^\circ\text{C}, a \text{ en } \Omega/^\circ\text{C} \text{ et } R_T \text{ en } \Omega)$$

On donne :

$$R_0 = 100 \, \Omega$$

$$R_1 = 1,5 \, \text{k}\Omega$$

$$a = 0,385 \, \Omega/^\circ\text{C} \text{ (coefficient de température)}$$

$$E = 5 \, \text{V} \text{ (tension d'alimentation du pont)}$$

**Question : 33.** Montrer que  $U = E \cdot \left( \frac{R_T}{R_1 + R_T} - \frac{R_2}{R_1 + R_2} \right)$ . (remarquer que  $U = V_b - V_a$ ).

1,5 pt

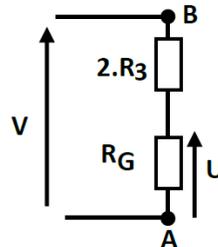
On dit que le pont est en équilibre lorsque sa tension de sortie est nulle ( $U = V_b - V_a = 0 \text{ V} \Rightarrow V_b = V_a$ ).

**Question : 34.** Calculer la valeur de  $R_2$  pour que le pont soit équilibré à la température  $T = 0 \text{ °C}$ . 2 pts

**Question : 35.** Pour  $R_2 = R_0$ , calculer la valeur de la tension  $U$  (en mV) à  $T = 25 \text{ °C}$ . 2 pts

### Amplificateur d'instrumentation

La branche **AB** de l'amplificateur d'instrumentation peut être simplifiée comme le montre la figure suivante :



**Question : 36.** Montrer alors que  $V = \left(1 + \frac{2.R_3}{R_G}\right) \cdot U$  1,5 pt

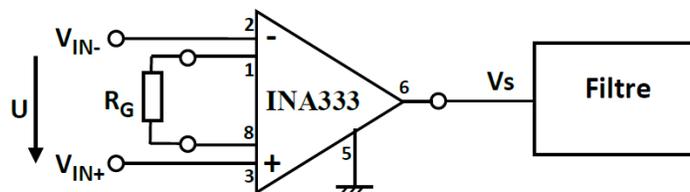
**Question : 37.** Dans le montage construit autour de l'amplificateur opérationnel **A3**, montrer que  $V_s = V$  (remarquer que  $V = V_2 - V_1$ ). 2 pts

On admet que la caractéristique  $U$  en fonction de la température  $T$  peut être assimilée à une droite et a pour équation :  $U = 1,12 \cdot 10^{-3} \cdot T$  (avec  $T$  en  $^{\circ}\text{C}$  et  $U$  en Volt).

**Question : 38.** Montrer que l'expression de la tension de sortie  $V_s$  en fonction de la température  $T$  est

$$V_s = 1,12 \cdot 10^{-3} \cdot \left(1 + \frac{2.R_3}{R_G}\right) \cdot T$$
1,5 pt

La réalisation matérielle de l'amplificateur d'instrumentation est confiée à l'amplificateur intégré **INA333** conformément à la figure suivante :



**Question : 39.** En se référant au schéma interne de l'amplificateur intégré **INA333 (DRES 03)**, relever la valeur de la résistance  $R_3$ . En déduire la valeur à donner à la résistance  $R_G$  pour obtenir une tension  $V_s = 5 \text{ V}$  à la température  $T = 25 \text{ °C}$ . 2 pts





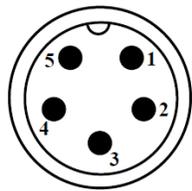
## Exercice 2 : examen national 2021, session de rattrapage

### Tâche 2 : Acquisition et conditionnement du signal

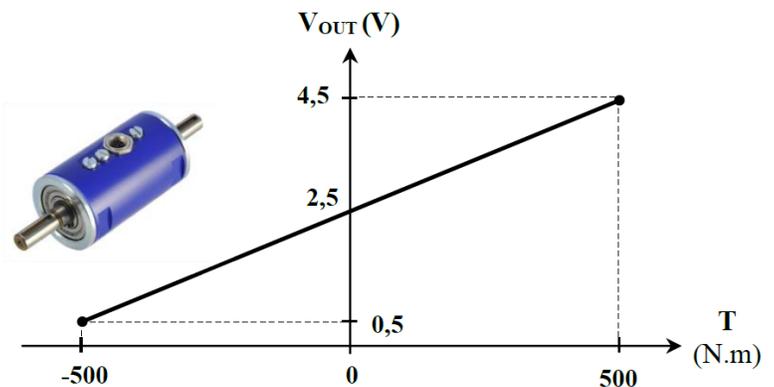
Le moteur **M3** du concasseur fournit un couple nominal  $T_N$  de **240 N.m**. Un capteur de couple mesure en permanence le couple  $T$  du moteur pour arrêter le concasseur en cas de surcharge ou de blocage des mâchoires de concassage.

Couplemètre SM 2200-500 de la série 2000 :

Le SM 2200-500 est un capteur de couple rotatif sans contact à électronique intégrée. Alimenté en **9-12V**, le SM 2200 délivre une sortie analogique de **0,5 à 4,5 V** lorsque le couple mesuré varie de **-500 à +500 N.m**.



- 1- Supply voltage  $V_{CC}$ .
- 2- Signal output  $V_{OUT}$ .
- 3- Ground GND.
- 4- Not used.
- 5- Reference voltage  $V_{REF}$ .



Brochage et fonction de transfert du couplemètre

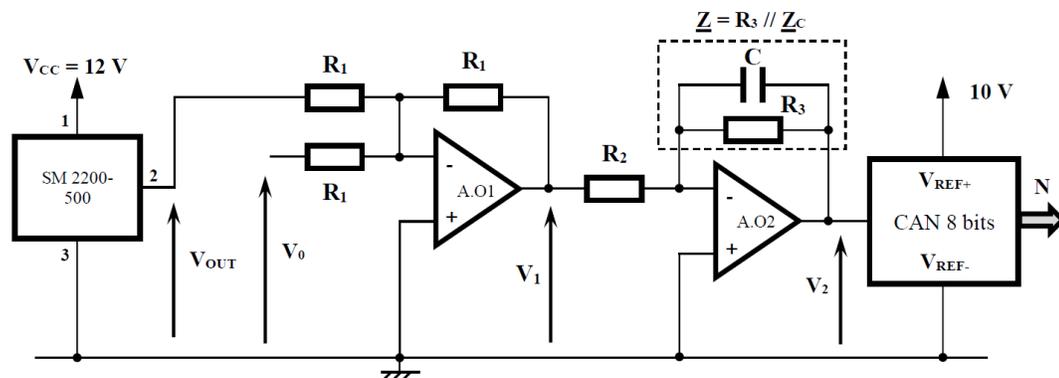
**Q39-** Calculer la sensibilité de ce capteur  $s = \frac{\Delta V_{OUT}}{\Delta T}$  et préciser son unité. 1 pt

**Q40-** Donner alors l'expression de la tension  $V_{OUT}$  (en volt) en fonction du couple  $T$  (N.m). 1 pt

Conditionnement du signal :

La tension  $V_{OUT}$  image du couple moteur  $T$  à la sortie du couplemètre est conditionnée (décalage, amplification et filtrage) avant d'être numérisée par un convertisseur analogique numérique (CAN 8 bits).

Le circuit de conditionnement de la page 8 fournit une tension  $V_2$ , proportionnelle à la tension  $V_{OUT}$ .



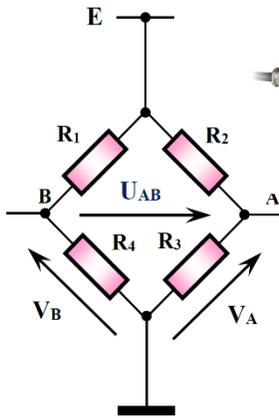


### Exercice 3 : examen national 2022, session normale

#### Tâche 1 : Acquisition et conditionnement du signal

Pour contrôler la masse supportée par la cabine, on utilise un capteur de pesage en traction.

Le corps d'épreuve du capteur se déforme légèrement lorsqu'il est soumis à une charge et revient à sa position initiale quand cette charge est retirée. Ces déformations extrêmement faibles sont acquises par 4 jauges de contrainte  $R_1$ ,  $R_2$ ,  $R_3$  et  $R_4$  identiques montées en pont de Wheatstone.



#### Caractéristiques du capteur de poids

- Etendue de mesure : **1000 Kg**
- Tension d'alimentation max tolérée **15 V**.
- Sensibilité  $s = 2 \text{ mV/V}$  (en pleine charge  $m = 1000 \text{ Kg}$ ).
- Charge statique maximale (% sur la pleine échelle) **150 %**.
- Charge de rupture (% sur la pleine échelle) **300 %**.
- Résistance au repos  $R = 350 \Omega$ .
- $R_1$  et  $R_3$  des jauges qui travaillent en extension, avec :  $R_1 = R_3 = R + \Delta R$
- $R_2$  et  $R_4$  des jauges qui travaillent en compression, avec :  $R_2 = R_4 = R - \Delta R$

#### Étude du capteur :

- La variation relative  $\frac{\Delta R}{R}$  de la résistance d'une jauge est liée à la masse  $m$  par la relation :  $\frac{\Delta R}{R} = K.m$ .
- La tension de déséquilibre du pont de Wheatstone  $U_{AB}$  est liée à la variation  $\Delta R$  par la relation :  $U_{AB} = E \cdot \frac{\Delta R}{R}$ .
- En pleine charge ( $m = 1000 \text{ Kg}$ ), la sensibilité du capteur  $s = \frac{U_{AB}}{E} = 2 \text{ mV/V}$ .
- Tension d'alimentation du pont  $E = 10 \text{ V}$ .

**Q.36** - Donner la valeur numérique de la tension  $U_{AB}$  en pleine charge.

**1,5 pt**

**Q.37** - Calculer le coefficient  $K$  et préciser son unité.

**1,5 pt**

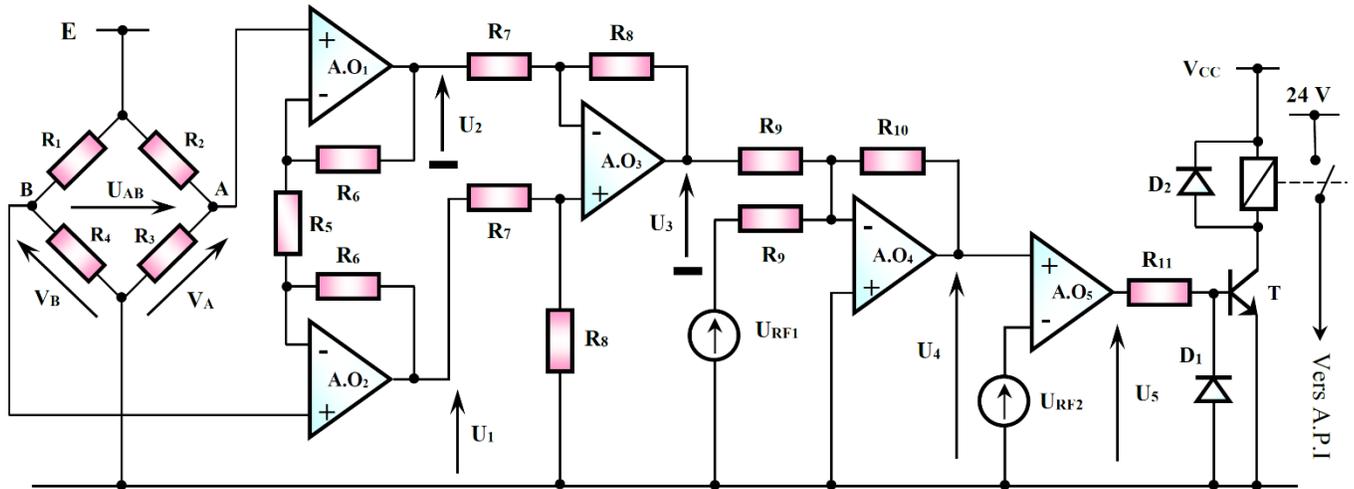
**Q.38** - Exprimer la tension  $U_{AB}$  en fonction de la masse  $m$ .

**1,5 pt**

**Q.39** - Déterminer en **Kg** la charge **statique**  $C_s$  maximale supportée par le capteur, et sa charge de **rupture**  $C_R$  minimale.

**2 pts**

Le montage ci-dessous représente le circuit de conditionnement du signal  $U_{AB}$  permettant d'informer l'A.P.I en cas de surcharge.



Les amplificateurs opérationnels sont supposés parfaits avec  $\pm V_{SAT} = \pm V_{CC} = \pm 15$  V.

#### Étude du conditionneur :

La tension de déséquilibre  $U_{AB}$  est amplifiée et mise en forme avant d'être utilisée par l'A.P.I. Pour cela, on utilise un amplificateur (montage autour de AO1, AO2, AO3 et AO4) associé à un comparateur à seuil (montage autour de AO5).

**Q.40** - Pour le montage autour de AO3, montrer que  $U_3 = \frac{R_8}{R_7} \cdot (U_1 - U_2)$ .

2 pts

On donne :

- $R_8 = R_7$  ;
- $U_1 = 51 \cdot V_B - 50 \cdot V_A$  ;
- $U_2 = 51 \cdot V_A - 50 \cdot V_B$ .

**Q.41** - Donner l'expression de  $U_3$  en fonction de  $U_{AB}$ , puis en fonction de la masse  $m$  (cabine + charge).

2 pts

**Q.42** - Pour le montage autour de AO4, montrer que  $U_4 = -\frac{R_{10}}{R_9} \cdot (U_3 + U_{RF1})$ .

2 pts

On donne :

- $U_3 = -2,02 \cdot 10^{-3} \cdot m$  ;
- $m = m_0 + m_C$ , avec :
  - ✓  $m_0 = 200$  Kg : masse de la cabine vide ;
  - ✓  $m_C$  : masse de la charge transportée ;
  - ✓  $m$  : masse totale (cabine + charge).
- $U_{RF1}$  tension continue de référence ( $U_{RF1} = 0,404$  V).

**Q.43** - Montrer alors que  $U_4 = 2,02 \cdot 10^{-3} \cdot \frac{R_{10}}{R_9} \cdot m_C$ .

1 pt

**Q.44** - On veut que  $U_4$  soit égale à 5 V lorsque  $m_C = 800$  kg, déterminer alors la valeur de  $R_{10}$ , sachant que  $R_9 = 22$  k $\Omega$ .

1,5 pt

A series of horizontal dotted lines for writing.

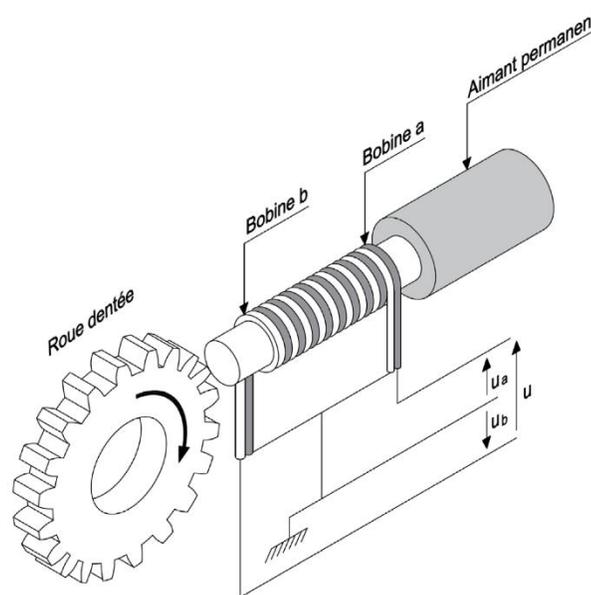
**Exercice 4 : examen national 2022, session de rattrapage**
**Tâche 1 : Acquisition de la vitesse du moteur d'entraînement du bac**

La vitesse de rotation du moteur d'entraînement est surveillée en permanence au cours du déplacement du bac. Le prélèvement de la vitesse est réalisé par un capteur inductif monté sur l'arbre de sortie du moteur.

⇒ **Capteur de vitesse inductif** (figure ci-dessous)

Le capteur intègre deux bobines soumises à l'action du champ magnétique d'un aimant permanent. L'ensemble est placé devant une roue dentée qui, en tournant, module le flux magnétique dans les bobines.

La variation de flux engendre deux tensions  $u_a$  et  $u_b$  en opposition de phase. Le capteur fournit une tension de sortie différentielle  $u = u_a - u_b$  ; cela le rend moins sensible aux signaux parasites.

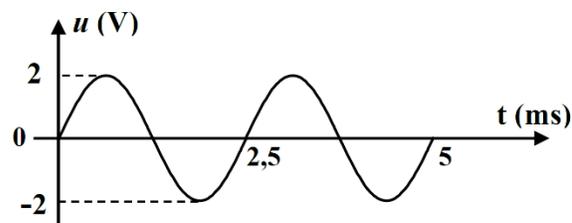


La fréquence  $f$  de la tension  $u$  est proportionnelle à la vitesse de rotation  $n$  :

$$f = \frac{n}{60} \cdot Z \quad \text{avec :} \quad \begin{array}{l} f : \text{fréquence du signal de sortie du capteur (en Hz)} ; \\ n : \text{vitesse de la roue dentée (en tr/min)} ; \\ Z : \text{nombre de dents de la roue.} \end{array}$$

**Q.33)** On donne  $Z = 20$  dents ; calculer la valeur de la fréquence  $f$  (en Hz) qui correspond à une vitesse de rotation du moteur  $n = 1475$  tr/min. **1 pt**

Un relevé de la sortie  $u$  issue du capteur a donné l'oscillogramme suivant :



Q.34) Calculer la vitesse de rotation  $n$  (en **tr/min**) qui correspond à cet oscillogramme ;

**1,5 pt**

Q.35) Pour les signaux  $u_a$  et  $u_b$  proposés, compléter les trois valeurs manquantes du signal  $u(t)$ .

**1,5 pt**

.....

.....

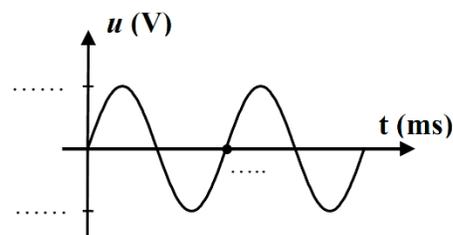
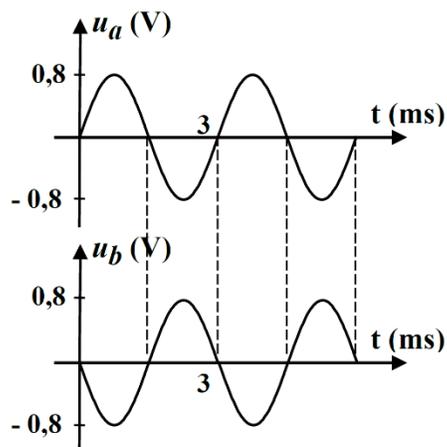
.....

.....

.....

.....

Q.35)



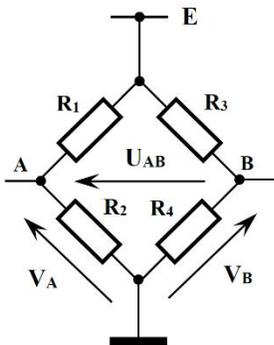
**Exercice 5 : examen national 2023, session normale****Tâche 2 : Acquisition, adaptation et conditionnement du signal**

Le moteur utilisé dans l'escalier fournit un couple nominal  $C_N$  de **72,2 N.m**. La mesure du couple permet de vérifier la puissance réelle du moteur.

Le système de sécurité contrôle en permanence le couple moteur pour arrêter l'escalier en cas de surcharge ou de blocage.

**Acquisition du couple moteur (couplemètre DM-DR1) :**

Ce couplemètre est un capteur de couple à jauges extensométriques inséré sur l'arbre, entre le moteur et la charge à entraîner. Il est constitué d'un barreau cylindrique sur lequel sont collées quatre jauges métalliques identiques.



- ✓ Etendu de mesure : 160 N.m (E.M)
- ✓ Sensibilité : 2 mV/V
- ✓ Résistance du pont : 350  $\Omega$
- ✓ Grande précision : 0,1% de l'E.M
- ✓ Couple ultime minimal de rupture : 250% de l'E.M

Les paires de jauges sont diamétralement opposées de telle sorte qu'une torsion du barreau, proportionnelle au couple exercé sur l'arbre, entraîne une variation symétrique de leurs résistances respectives :

$$R_1 = R_4 = R - \Delta R \text{ et } R_2 = R_3 = R + \Delta R$$

- $R$  est la résistance au repos ( $R = 350 \Omega$ ) ;
- $\Delta R$  est la variation de résistance proportionnelle au couple à mesurer  $C_u$  selon la relation  $\frac{\Delta R}{R} = K \cdot C_u$ .
- Les quatre jauges sont interconnectées en pont de Wheatstone qui est alimenté en continu sous la tension  $E = 10 \text{ V}$ .

**Q.31-Exprimer les tensions  $V_A$  et  $V_B$  en fonction de  $E$ ,  $R$  et  $\Delta R$ .**

**2 pts**

**Q.32-Montrer alors que la tension de déséquilibre  $U_{AB}$  s'écrit sous la forme :  $U_{AB} = E \cdot \frac{\Delta R}{R}$ .**

**1 pt**

En pleine charge ( $C_u = 160 \text{ N.m}$ ), la valeur numérique de la tension de déséquilibre est :  $U_{AB} = 20 \text{ mV}$ .

**Q.33-Mettre l'expression de  $U_{AB}$  sous la forme  $U_{AB} = a \cdot C_u$ . Calculer alors, la valeur du coefficient  $a$  en précisant son unité.**

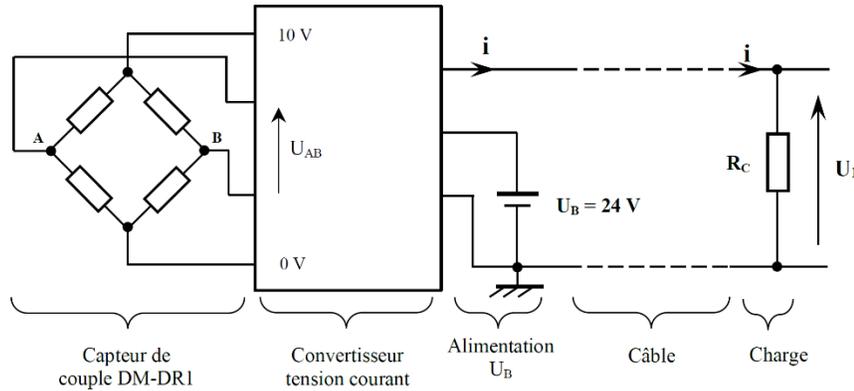
**2 pts**

**Q.34-Déterminer  $C_r$  le couple ultime minimal de rupture en  $N.m$ .**

**1 pt****Adaptation du signal :**

Le capteur de couple DM-DR1 est relié à un convertisseur tension-courant délivrant un courant d'intensité qui varie linéairement avec le couple.

La figure ci-dessous représente le câblage du capteur associé au convertisseur tension-courant.



On a branché une résistance de charge  $R_C$  de  $180 \Omega$  dans la boucle de courant.

On donne l'expression du courant  $i$  (mA) en fonction du couple  $C_u$  (N.m) :  $i = 0,16.C_u + 4$

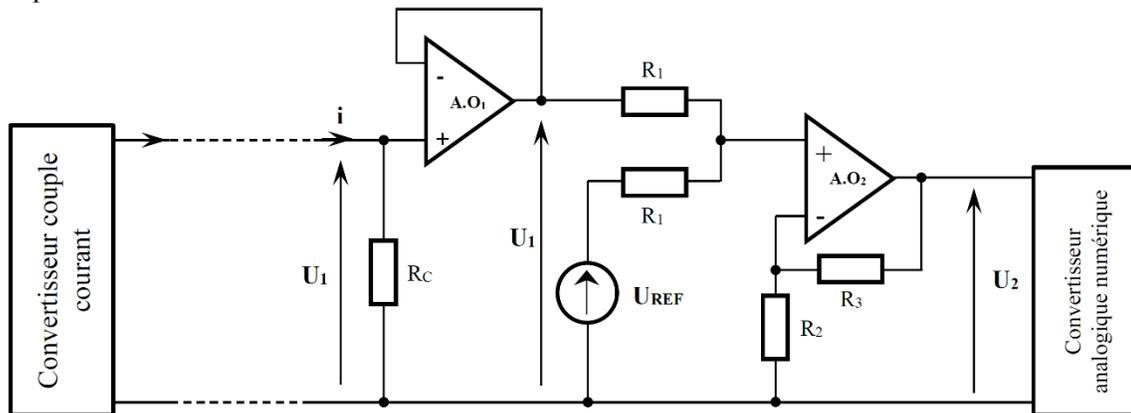
**Q.35-** Donner l'expression de la tension  $U_1$  (mV) en fonction du couple  $C_u$  (N.m). **1,5 pt**

**Q.36-** En déduire la sensibilité  $s$  du montage et donner son unité. **1,5 pt**

Amplification et décalage du signal analogique  $U_1$  :

Le signal  $U_1$  image du couple  $C_u$ , doit être numérisé par un convertisseur analogique numérique (CAN). Le signal à l'entrée du CAN doit être compris entre  $0$  et  $5$  V, lorsque le couple  $C_u$  varie de  $0$  à  $100$  N.m.

L'objectif est de conditionner le signal  $U_1$  pour la conversion analogique numérique, alors on utilise le montage représenté ci-dessous :



Les amplificateurs opérationnels sont supposés parfaits.

On donne :

- $U_{REF}$  tension continue, avec  $U_{REF} = -0,72$  V.
- $U_1$  (V) =  $29.10^{-3}. C_u + 0,72$ .

**Q.37-** Donner l'expression de  $U_2$  en fonction de  $U_1$ ,  $U_{REF}$ ,  $R_2$  et  $R_3$ . **3 pts**

**Q.38-** En déduire l'expression de  $U_2$  en fonction du couple moteur  $C_u$ . **1 pt**

**Q.39-** On veut que  $U_2$  soit égale à  $5$  V pour  $C_u = 100$  N.m pour exploiter toute l'échelle du CAN, calculer alors la résistance  $R_3$ , sachant que  $R_2 = 3,3$  K $\Omega$ . **1,5 pt**



**Exercice 6 : examen national 2023, session de rattrapage**

 ⇒ Acquisition du taux d'oxygène dissous

L'oxygène dissous dans l'eau du bassin d'aération est mesuré à l'aide d'une sonde oxygène dont voici un extrait de ses caractéristiques techniques :

| Sonde pour oxygène dissous |  |
|----------------------------|--|
| Précision de mesure        | $\pm 0,1$ mg/L (milligrammes par litre)    |
| Tension d'alimentation     | 18 ... 30 V DC                             |
| Signal de sortie           | 4-20 mA correspondant à la plage de mesure |
| Principe de mesure         | Optique                                    |
| Plage de mesure            | 0,1 ... 20 mg/L                            |



Sonde oxygène

Q.31) À partir de la fiche technique du capteur, ci-dessus, compléter le tableau.

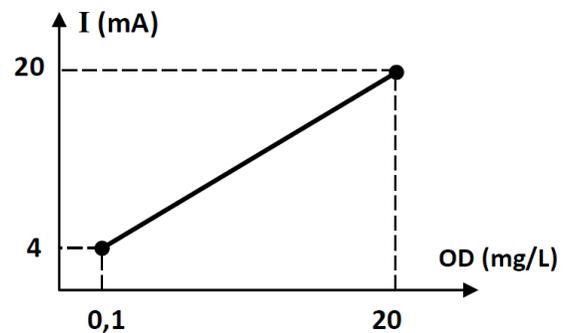
**2 pts**

La sortie **4-20 mA** présente l'avantage de maintenir la précision malgré les chutes de tension dues aux résistances des fils de liaison.

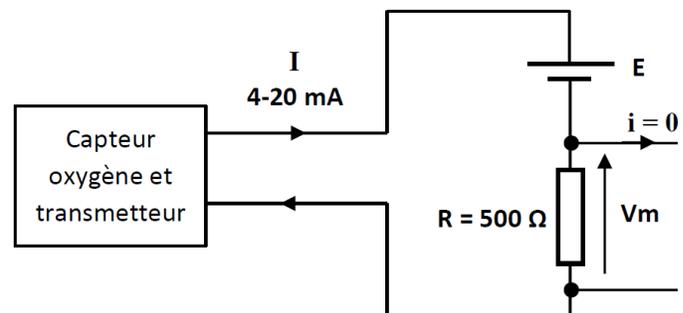
La courbe ci-contre est la caractéristique de transfert **I** en fonction de l'oxygène dissous **OD** du capteur

On note que la courbe a pour équation :  $I = a \cdot OD + 3,92$

(**OD** en **mg/L** et **I** en **mA**)



Le schéma ci-contre est d'un montage typique de conversion du courant **I** délivré par le capteur en une tension **V<sub>m</sub>**.



Q.32) Pour une consigne typique d'oxygène dissous **OD = 2 mg/L**, calculer l'intensité du courant **I** délivrée par le capteur.

**1,5 pt**

Q.33) En déduire la tension **V<sub>m</sub>** correspondante

**1 pt**

|                     |  |
|---------------------|--|
| Grandeur mesurée    | .....  |
| Étendue de mesure   | .....  |
| Grandeur de sortie  | <input type="checkbox"/> Tension <input type="checkbox"/> Courant <input type="checkbox"/> Impédance |
| Nature de la sortie | <input type="checkbox"/> Analogique <input type="checkbox"/> Numérique <input type="checkbox"/> TOR  |

.....

.....

.....

.....

.....

.....

.....

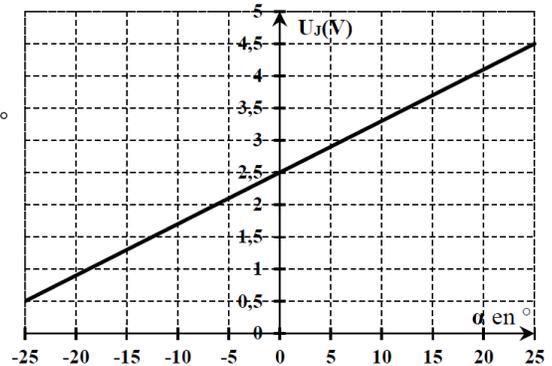
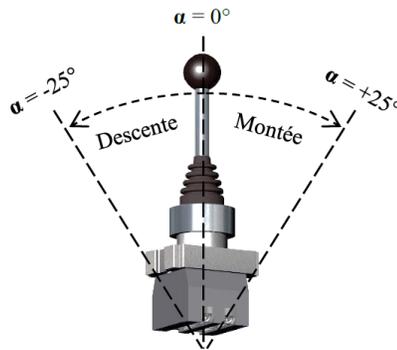
.....

**Exercice 7 : examen national 2024, session normale**
**Tâche 2 : Acquisition de la position du manche de la manette de commande du crochet**

La montée ou la descente du crochet est commandée par l'opérateur grâce à une manette industrielle : plus le manche de la manette est poussé vers l'extrémité plus la vitesse du moteur augmente.

La manette de pilotage du crochet (Joystick **J1**) est de type analogique. Alimentée en **5 V**, elle délivre une sortie analogique  $U_J$  de **0,5 à 4,5 V** lorsque la position du manche  $\alpha$  varie de **-25° à +25°**.

- $\alpha = 0^\circ$  : manche de la manette en position médiane (manche relâché).
- $\alpha = +25^\circ$  : manche de la manette à l'extrémité avant.
- $\alpha = -25^\circ$  : manche de la manette à l'extrémité arrière.



Fonction de transfert de la manette :  $U_J = f(\alpha)$

**Q.40-** Calculer la sensibilité de ce capteur  $s = \frac{\Delta U_J}{\Delta \alpha}$  et préciser son unité, en déduire l'expression de la tension  $U_J$  (en Volt) en fonction de la position du manche  $\alpha$  (en °). **2 pts**

**ADAPTATION DU SIGNAL ISSU DU JOYSTICK**

Le signal  $U_J$ , à la sortie de la manette, est numérisé par un convertisseur analogique numérique (CAN interne du PIC 16F876). Le signal à l'entrée du CAN doit être compris entre **0** et **5 V**, lorsque la position du manche de la manette  $\alpha$  varie de **-25° à +25°**.

**Tâche 3 : Étude du circuit de décalage du signal  $U_J$** 

L'objectif est de conditionner le signal  $U_J$  pour exploiter toute l'échelle du convertisseur analogique numérique, alors on utilise le montage autour de l'amplificateur **AO2** représenté sur le document **DRES 03**.

**Q.41-** Donner l'expression de  $U_S$  en fonction de  $U_J$ ,  $U_d$ ,  $R_2$  et  $R_3$ . En déduire que l'expression de  $U_S$  en fonction de la position  $\alpha$  du manche de la manette s'écrit :  $U_S = \frac{R_3 + R_2}{2 \cdot R_2} \cdot (0,08 \cdot \alpha + U_d + 2,5)$ . **3 pts**

**Q.42-** Pour  $\alpha = -25^\circ$ , calculer la valeur de la tension  $U_d$  pour avoir  $U_S = 0 V$ . **1,5 pt**

**Q.43-** Sachant que  $R_2 = 22 k\Omega$  et  $U_S = \frac{R_3 + R_2}{2 \cdot R_2} \cdot (0,08 \cdot \alpha + 2)$ , calculer la valeur de la résistance  $R_3$  pour avoir  $U_S = 5 V$  pour  $\alpha = +25^\circ$ . **1,5 pt**

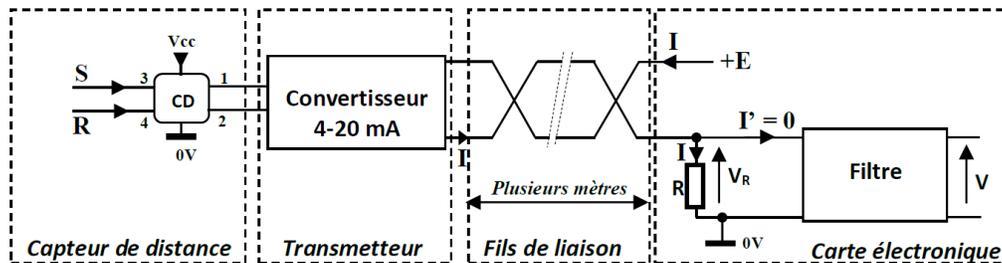


### Exercice 8 : examen national 2024, session de rattrapage

Le capteur de distance (CD) a une étendue de mesure de **0 à 4800 mm**. Il est situé assez loin de l'armoire de commande contenant l'ensemble de l'appareillage ainsi que les cartes électroniques.

Pour éliminer l'effet négatif de la chute de tension due à la longueur importante des câbles, on utilise un transmetteur de type « **4-20 mA** ».

Le schéma de principe du transmetteur est donné ci-dessous.



Le capteur analogique de distance **CD**, qui mesure la longueur du tube à découper, possède six bornes : deux sorties pour la mesure, deux entrées **S** et **R** (**S** : lancement de la mesure ; **R** : réinitialisation de la mesure) et deux bornes d'alimentation **Vcc** et **0 V**.

#### Tâche 1 : Étude du transmetteur

Le transmetteur comporte un convertisseur qui génère un courant **I** analogique, image de la longueur mesurée, qui varie entre **4 mA** et **20 mA** :

- **I = 4 mA** pour la première valeur de l'échelle de mesure (**L = 0 mm**)
- **I = 20 mA** pour la dernière valeur de l'échelle de mesure (**L = 4800 mm**).

Le courant **I** varie entre **4** et **20 mA** en fonction de la longueur **L** suivant la relation :  $I = a.L + b$ .

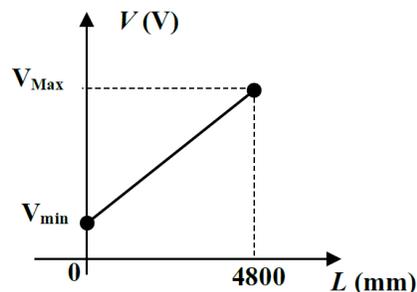
**Q35.** Calculer les valeurs de **a** (sous forme d'une fraction) et de **b** en précisant leurs unités. **2 pts**

**Q36.** Exprimer  $V_R$  (en **V**) en fonction de **R** (en  $\Omega$ ) et **L** (en **mm**). **2 pts**

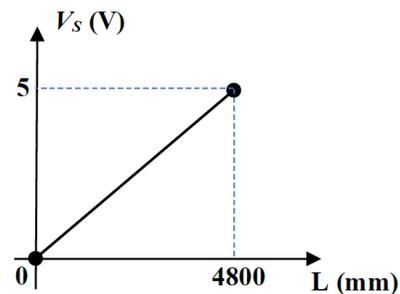
**Q37.** Calculer les valeurs des tensions  $V_{Rmin}$  et  $V_{Rmax}$  de  $V_R$  pour **R = 275  $\Omega$** . **1 pt**

Pour un traitement ultérieur, on est amené à adapter le signal **V** de façon à obtenir une tension  $V_S$  de **0 V** lorsque la distance mesurée est de **0 mm** et de **5 V** lorsque la distance mesurée est de **4800 mm**.

Les courbes ci-dessous représentent la réponse du capteur avant et après adaptation du signal.



Réponse avant adaptation



Réponse après adaptation



## 2

**FILTRAGE ANALOGIQUE****Généralités**

La fonction filtrage consiste à atténuer les signaux de fréquences indésirables.

L'intervalle des fréquences autorisées par le filtre s'appelle bande passante.

On distingue :

- Les filtres **passifs** : constitués seulement de dipôles passifs (résistances, condensateurs, bobines) ;
- Les filtres **actifs** : constitués de dipôles passifs et de dipôles actifs (amplificateurs opérationnels, transistors...).

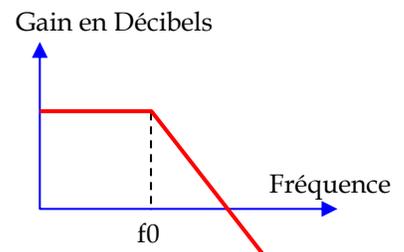
Les filtres actifs ont l'avantage de :

- Avoir des caractéristiques indépendantes de la charge ;
- Avoir une amplification possible dans la bande passante.

**1. Types de filtres**

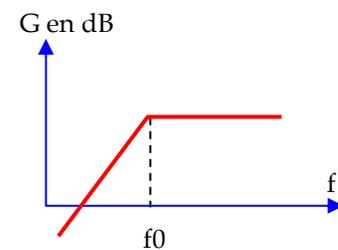
⇒ Filtre passe-bas

Laisse passer les signaux de fréquence **inférieure** à une fréquence déterminée **f<sub>0</sub>** dite **fréquence de coupure**.



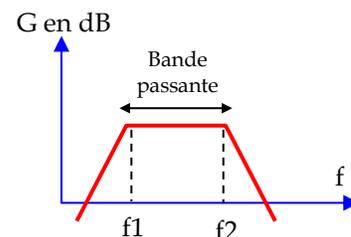
⇒ Filtre passe-haut

Laisse passer les signaux de fréquence **supérieure** à sa fréquence de coupure **f<sub>0</sub>**.



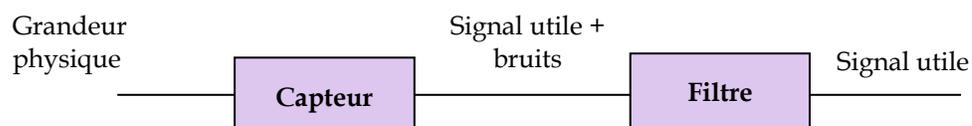
⇒ Filtre passe-bande

Laisse passer les signaux de fréquence **comprise** entre ses deux fréquences de coupure **f<sub>1</sub>** et **f<sub>2</sub>**.



Parmi les principales applications des filtres, on peut citer :

- Extraire la valeur moyenne d'un signal périodique par un passe-bas ;
- Changer la tonalité d'un son (éliminer les sons aigus par un passe-bas, éliminer les sons graves par un passe-haut) ;
- Sélectionner une fréquence ou une bande de fréquence par un passe-bande (télévision, radio...).

**Cas de la chaîne d'acquisition**

Lois mathématiques et électriques utiles

- Quelques propriétés de la fonction logarithme décimal

$$\log(ab) = \log a + \log b \quad \log 1 = 0$$

$$\log\left(\frac{a}{b}\right) = \log a - \log b \quad \log 10 = 1$$

$$\log(a^n) = n \cdot \log a \quad \log 100 = 2$$

- Quelques propriétés des nombres complexes

Soit  $z$  un nombre complexe,  $z = a + jb$  où  $a, b$  sont des réels et  $j^2 = -1$

Module de  $z$  :  $\|z\| = \sqrt{a^2 + b^2}$

Argument de  $z$  :  $\arg(z) = \arctg\left(\frac{b}{a}\right)$

$$\|z_1 \cdot z_2\| = \|z_1\| \cdot \|z_2\|$$

$$\arg(z_1 \cdot z_2) = \arg(z_1) + \arg(z_2)$$

$$\left\|\frac{z_1}{z_2}\right\| = \frac{\|z_1\|}{\|z_2\|}$$

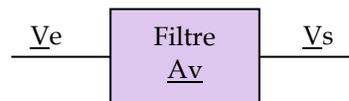
$$\arg\left(\frac{z_1}{z_2}\right) = \arg(z_1) - \arg(z_2)$$

- Impédances en notation complexe

L'impédance d'une résistance  $R$  est :  $Z_R = R$

L'impédance d'une bobine d'inductance  $L$  est :  $Z_L = jL\omega$

L'impédance d'un condensateur de capacité  $C$  est :  $Z_C = \frac{1}{jC\omega}$

**2. Fonction de transfert**

La fonction de transfert d'un filtre, en notation complexe  $\underline{A_v} = \frac{V_s}{V_e}$

- **L'amplification  $A_v$**  est le module de  $\underline{A_v}$  soit  $A_v = \|\underline{A_v}\|$

- **La phase  $\varphi$**  (déphasage entre l'entrée et la sortie) est l'argument de  $\underline{A_v}$  soit  $\varphi = \arg(\underline{A_v})$

Plutôt que d'utiliser l'amplification  $A_v$ , on préfère parler du **gain  $G$**  en décibels dB :

$$G = 20 \log(A_v)$$

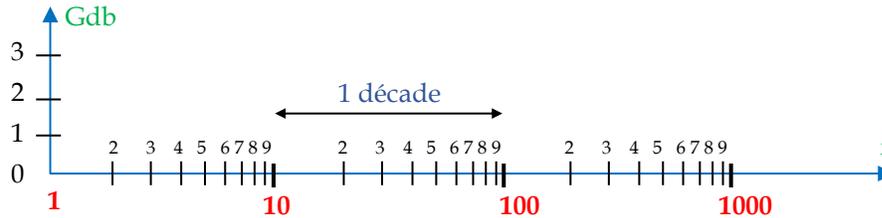
On en déduit l'amplification en fonction du gain

$$A_v = 10^{\frac{G}{20}}$$

**3. Diagramme de Bode**

Un axe gradué linéairement est mal adapté pour représenter une grandeur qui s'étend sur un intervalle très large, c'est le cas du domaine des fréquences d'un filtre. Pour cela, on recourt à une échelle logarithmique.

Le diagramme de Bode est la représentation du gain dB et de la phase en fonction de la fréquence sur une échelle semi-logarithmique.



Une décade se définit par l'intervalle  $[f, 10f]$ .

Par exemple,  $[20\text{Hz}, 200\text{Hz}] = 1$  décade et  $[15\text{kHz}, 15\text{MHz}] = 3$  décades

**4. Fréquence de coupure à -3db**

La fréquence de coupure à -3 dB d'un filtre est telle que  $Av = Av_0/\sqrt{2}$

En décibels, cela se traduit par  $G = G_0 - 3\text{db}$ .

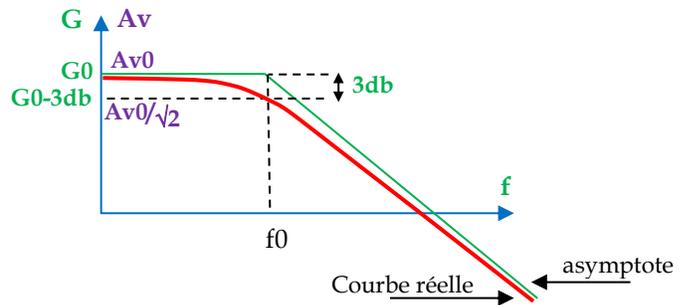
En effet,  $Av = Av_0/\sqrt{2}$

$$\Rightarrow G = 20 \cdot \log(Av_0/\sqrt{2})$$

$$\Rightarrow G = 20 \log(Av_0) - 20 \log\sqrt{2} = G_0 - 3\text{db}$$

$G_0$  : gain maximal (ou gain dans la bande passante ou encore gain statique)

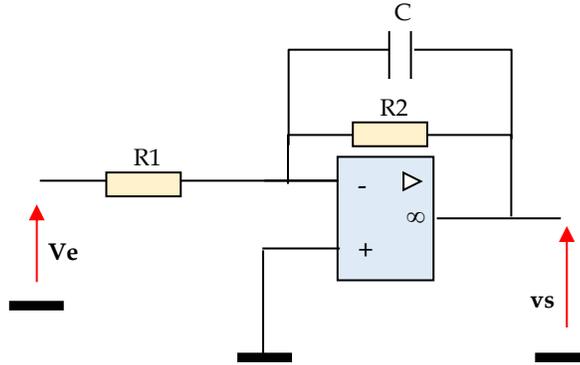
$Av_0$  : amplification maximale



**5. Formes canoniques des filtres du 1<sup>e</sup> ordre**

Les fonctions de transferts, en notation complexe, des filtres du 1<sup>e</sup> ordre peuvent s'exprimer sous les formes canoniques suivantes :

| Passe-bas  | Passe-haut   | Passe-bande  |
|--|--|--|
| $\underline{Av} = Av_0 \cdot \frac{1}{1 + j\frac{\omega}{\omega_0}}$   | $\underline{Av} = Av_0 \cdot \frac{j\frac{\omega}{\omega_0}}{1 + j\frac{\omega}{\omega_0}}$      | $\underline{Av} = Av_0 \cdot \frac{1}{(1 + j\frac{\omega}{\omega_2})} \cdot \frac{j\frac{\omega}{\omega_1}}{(1 + j\frac{\omega}{\omega_1})}$   |
| <p><math>Av_0</math> : Amplification maximale</p> <p><math>f_0 = \frac{\omega_0}{2\pi}</math> : fréquence de coupure</p> | <p><math>Av_0</math> : Amplification maximale</p> <p><math>f_0</math> : fréquence de coupure</p> | <p><math>Av_0</math> : Amplification maximale</p> <p><math>f_1</math> : fréquence de coupure basse</p> <p><math>f_2</math> : fréquence de coupure haute (<math>f_2 \gg f_1</math>)</p> |

**Étude des filtres actifs****Filtre passe-bas**

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

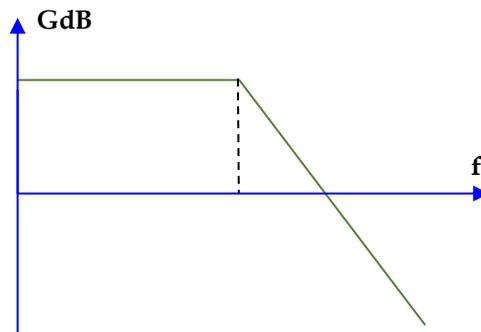
.....

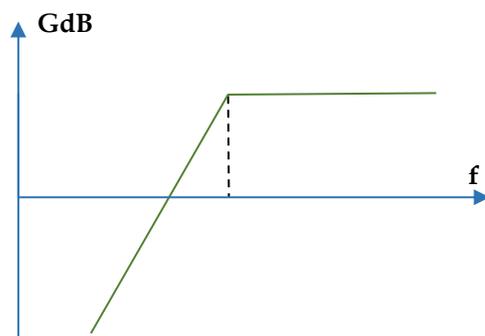
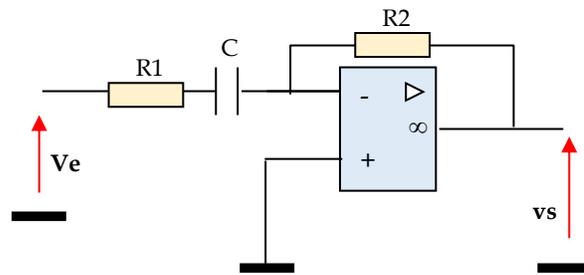
.....

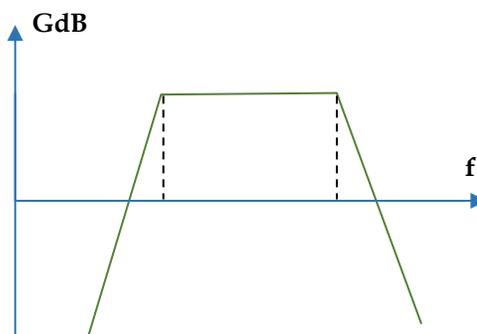
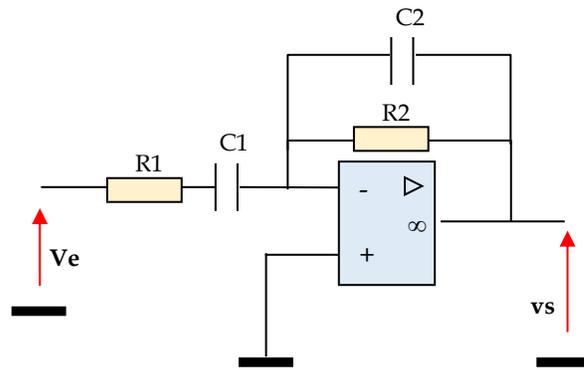
.....

.....

.....

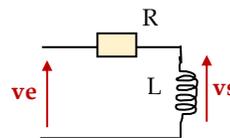


Filtre passe-haut

**Filtre passe-bande**

**Exercice 1 : filtre passe-haut passif**

1. Déterminer la fonction de transfert complexe  $A_v$  et déduire le type du filtre ;
2. Calculer la fréquence de coupure  $f_0$  et le gain maximal  $G_0$  ;  
On donne  $R = 2 \text{ k}\Omega$  et  $L = 10,6 \text{ mH}$  ;
3. Esquisser le diagramme asymptotique du gain ;
4. Calculer le gain  $G$  en dB pour  $f = 1 \text{ KHz}$ .



.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

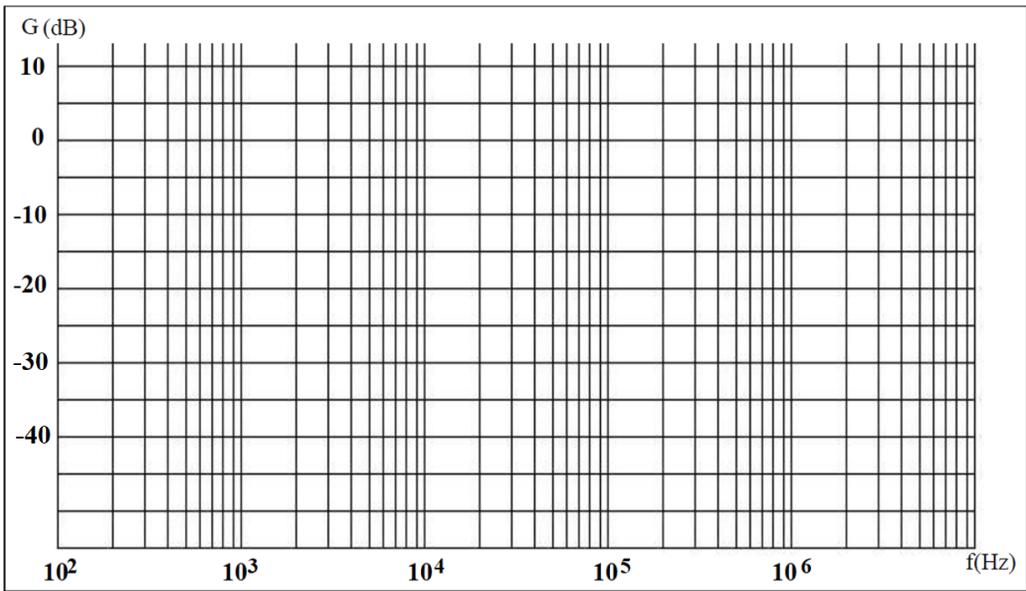
.....

.....

.....

.....

.....



$A_{v0} = \dots\dots$

$G_0 = \dots\dots$

$f_0 = \dots\dots$

Pente = .....

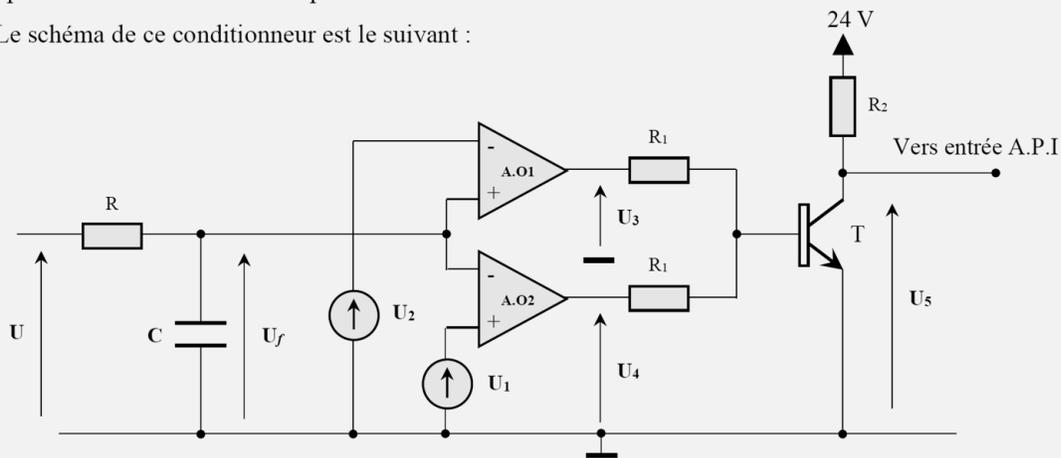


**Exercice 3 : examen national 2020, session de rattrapage**

**Conditionnement du signal**

Le circuit de conditionnement du signal de sortie du capteur à ultrasons est constitué d'un filtre, d'un comparateur à fenêtre et d'un adaptateur de tension.

Le schéma de ce conditionneur est le suivant :



Afin de supprimer tout signal parasite, la fréquence de coupure à -3 dB du filtre RC est fixée à  $f_0 = 0,5 \text{ Hz}$ .

Q39. Exprimer  $\underline{Av} = \frac{U_f}{U}$  en fonction de R et  $\underline{Zc}$  ( $\underline{Zc}$  : impédance du condensateur C). **1 pt**

Q40. Mettre  $\underline{Av}$  sous la forme  $\underline{Av} = \frac{1}{1 + j\frac{f}{f_0}}$  et exprimer  $f_0$  en fonction de R et C. **2 pts**

Q41. Pour  $R = 100 \text{ k}\Omega$ , calculer la capacité C du condensateur en  $\mu\text{F}$ . **1,5 pt**

Q42. Tracer la courbe du gain (diagramme asymptotique). Donner la nature du filtre (passe-bas, passe-haut, passe-bande) et sa bande passante à -3 dB ainsi que sa pente d'atténuation. **3,5 pts**

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

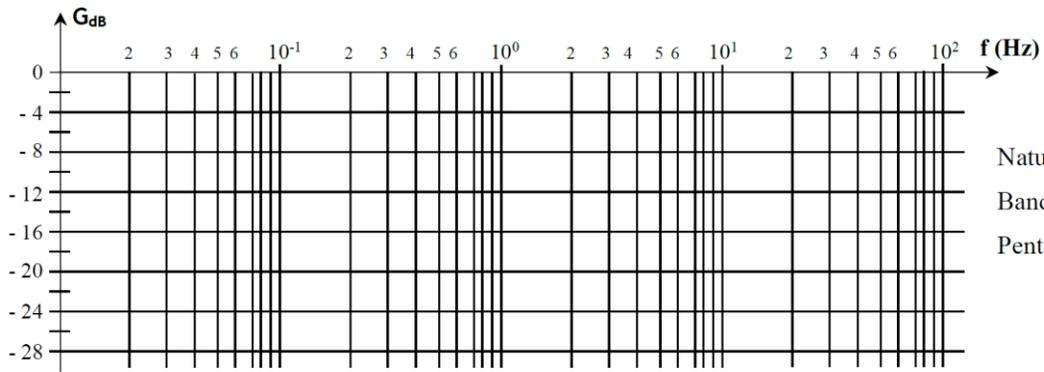
.....

.....

.....

.....

.....



Nature du filtre : .....

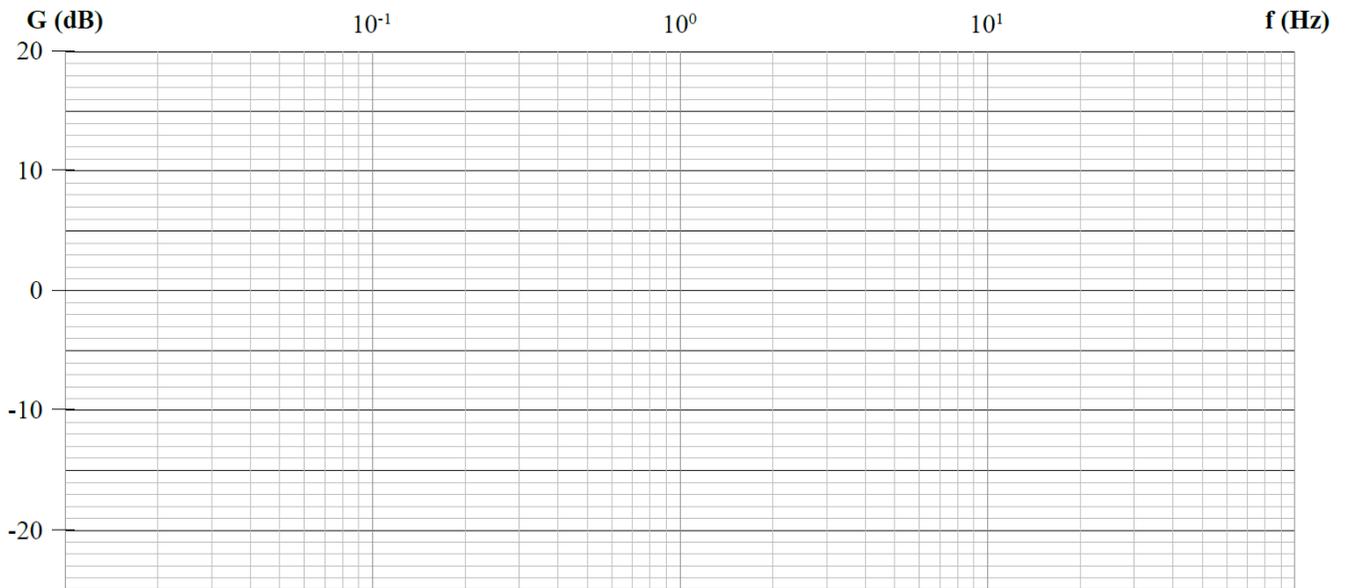
Bande passante = .....

Pente d'atténuation = .....  
dB / décade





.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....



Nature du filtre : .....

Bande passante : .....

**Exercice 6 : examen national 2024, session de rattrapage**

**Tâche 2 : Étude du Filtre**

Pour réduire les parasites probables à la réception, on utilise un filtre **RC** dont la fonction de transfert est :

$$\underline{T} = \frac{1}{1 + jRC\omega}$$

**Q38.** Préciser la nature de ce filtre : passe-bas ou passe-haut. **1 pt**

**Q39.** Déterminer le module **T** de la fonction de transfert de ce filtre en fonction de **f** et **f<sub>0</sub>**; avec **f<sub>0</sub>** fréquence de coupure à **-3 db**. **2 pts**

**Q40.** Monter que le gain **G** (en **db**) peut se mettre sous la forme suivante :  $G = -10 \cdot \log \left( 1 + \left( \frac{f}{f_0} \right)^2 \right)$ . **2 pts**

**Q41.** Calculer le gain **G** pour **f = 0**, **f = 10.f<sub>0</sub>** et **f → ∞** et tracer le diagramme asymptotique **G** en fonction de la fréquence **f** (Diagramme de Bode). **3,5 pts**

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

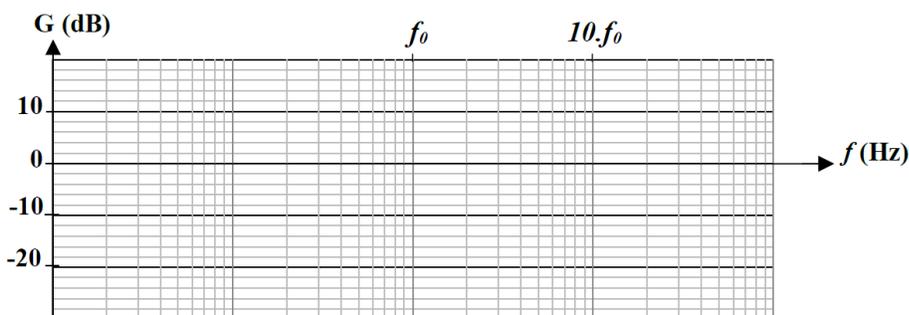
.....

.....

.....

**Q41.**

|                               |       |                         |              |
|-------------------------------|-------|-------------------------|--------------|
| Fréquence <b>f</b>            | 0     | <b>10.f<sub>0</sub></b> | <b>f → ∞</b> |
| Gain <b>G</b> (en <b>dB</b> ) | ..... | .....                   | .....        |



## 3

**MISE EN FORME****Régimes de fonctionnement de l'amplificateur opérationnel****1. Régime linéaire**

Ce régime se reconnaît lorsque la sortie est reliée à l'entrée "-" de l'AOP. On parle de contre réaction négative.

Dans ce cas,  $v_+ = v_-$

⇒ Exemples d'application du régime linéaire : fonctionnement en amplificateur, filtre.

**2. Régime non linéaire (ou de saturation)**

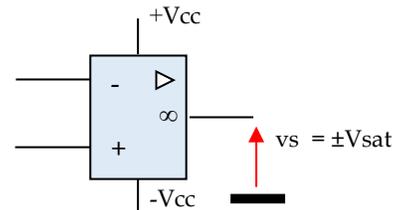
Ce régime se reconnaît lorsque la sortie est reliée à l'entrée "+" de l'AOP (contre réaction positive) ou lorsque l'AOP est en boucle ouverte.

La sortie de l'AOP ne peut prendre que 2 valeurs possibles :

$$\begin{cases} +V_{sat} & \text{si } v_+ > v_- \\ -V_{sat} & \text{si } v_+ < v_- \end{cases}$$

Le **basculement** se produit à l'instant où  $v_+ = v_-$

⇒ Exemple d'application du régime de saturation : fonctionnement en comparateur.

Notes

Si l'alimentation est symétrique c'est-à-dire par  $+V_{cc}$  et  $-V_{cc}$ , les tensions de saturation seront  $+V_{sat}$  et  $-V_{sat}$  ;

Si l'alimentation est simple c'est-à-dire par  $+V_{cc}$  et la **masse**, les tensions de saturation seront  $+V_{sat}$  et  $0\text{ V}$  ;

La tension de saturation  $V_{sat}$  est presque égale à la tension d'alimentation de l'AOP ; soit  $V_{sat} \approx V_{cc}$ .

**Comparateur à un seul seuil****1. Comparateur non inverseur**

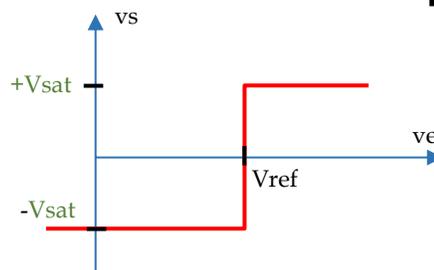
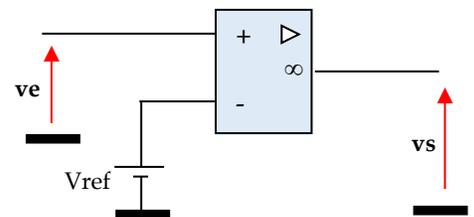
On utilise un comparateur à base d'AOP en boucle ouverte.

Le signal d'entrée  $v_e$  est appliqué à l'entrée non inverseuse "+" de l'AOP

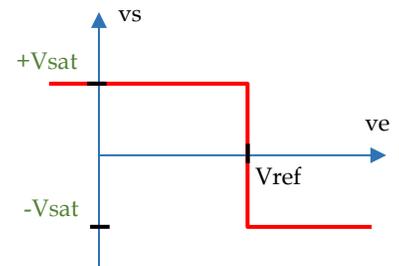
Deux cas sont possibles :

Si  $v_e > V_{ref}$  alors  $v_s = +V_{sat}$

Si  $v_e < V_{ref}$  alors  $v_s = -V_{sat}$

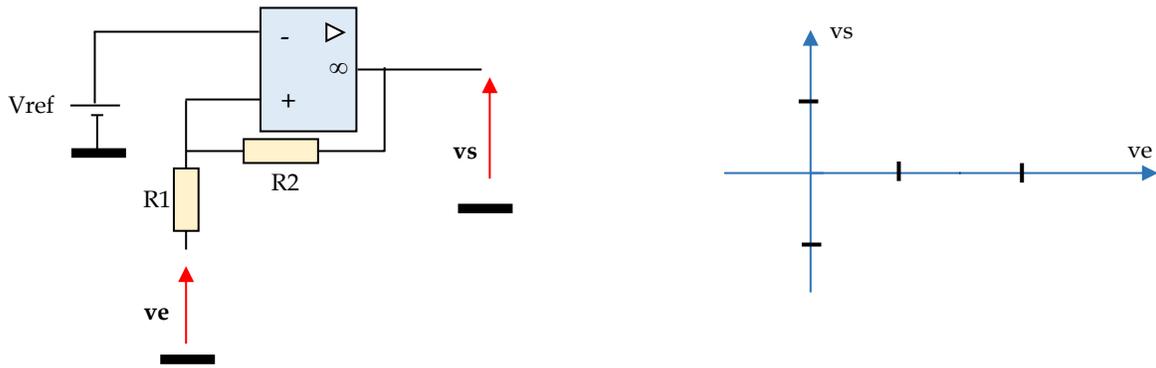
Caractéristique de transfert du comparateur**2. Comparateur inverseur**

En appliquant  $v_e$  à l'entrée inverseuse "-", on obtient un comparateur inverseur dont voici la caractéristique de transfert.





2. **Trigger non inverseur**



.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

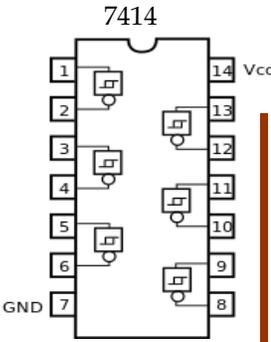
.....

.....

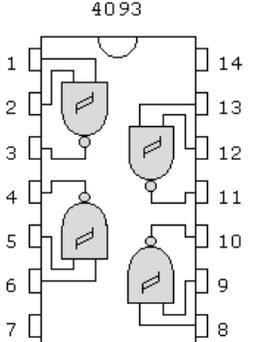
.....

.....

**Réalisation à l'aide de circuits logiques**



Le circuit intégré 7414 fait partie de la série des circuits intégrés 7400 utilisant la technologie TTL. Ce circuit est composé de six portes logiques indépendantes inverseuses avec trigger de Schmitt.

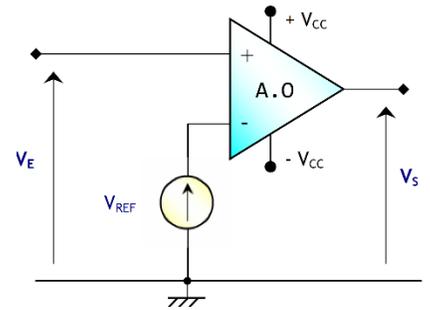


Le 4093 fonctionne comme un 4011 (quadruple NAND) dont toutes les entrées seraient dotées d'un trigger de Schmitt.

Exercices

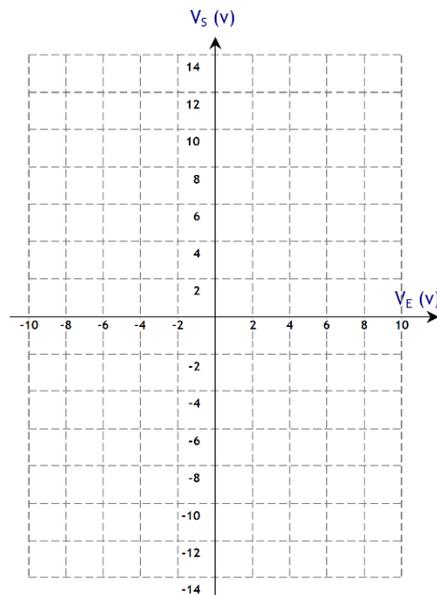
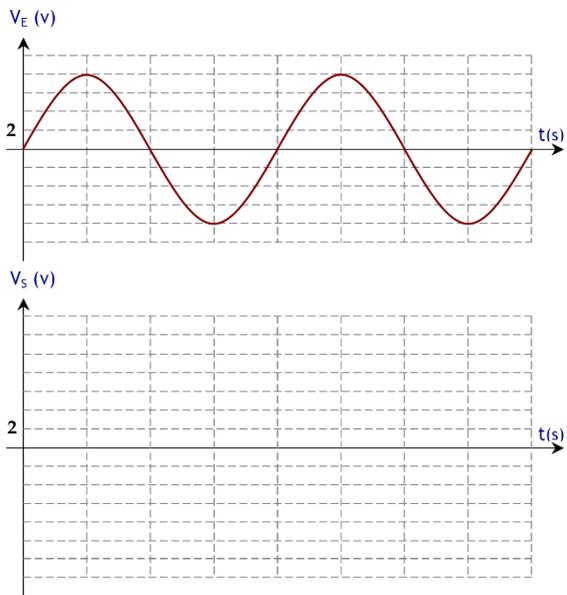
**Exercice 1 : comparateur à un seuil**

Le montage ci-contre est d'un comparateur ;  
 L'entrée inverseuse est prise comme référence ;  
 Le signal d'entrée  $V_E$  est appliqué à l'entrée non inverseuse ;  
 $\pm V_{CC} = \pm 12 \text{ V}$  ;  $V_E(t) = 8 \cdot \sin(2\pi f \cdot t)$  (en V).

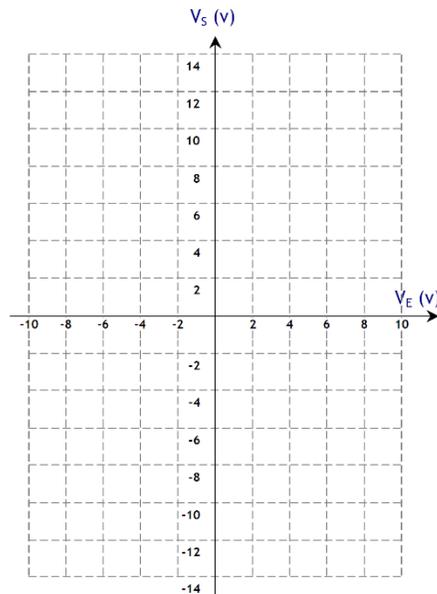
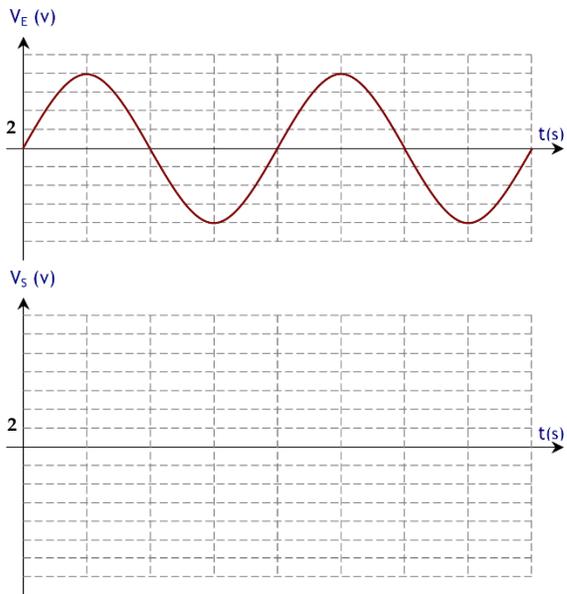


- Compléter :  
 C'est un comparateur .....  
 Pour  $V_E < V_{REF}$  .....  
 Pour  $V_E > V_{REF}$  .....

- Compléter le tracé du signal de sortie  $V_S$  et celui de la caractéristique de transfert pour  $V_{REF} = 0 \text{ V}$



- Compléter le tracé du signal de sortie  $V_S$  et celui de la caractéristique de transfert pour  $V_{REF} = 6 \text{ V}$



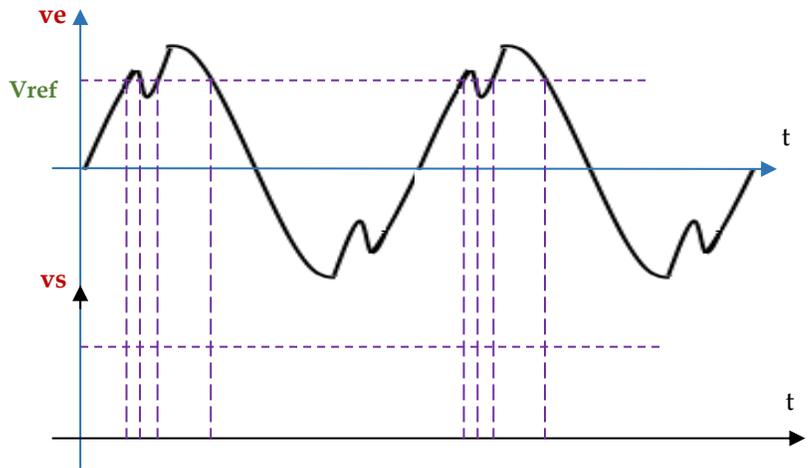
**Exercice 2 : Avantage du trigger**

Dans les deux cas, l'AOP est alimenté en 12 V et 0 V. L'entrée  $v_e$  est un signal sinusoïdal bruité.  
Tracer la réponse  $v_s$ .

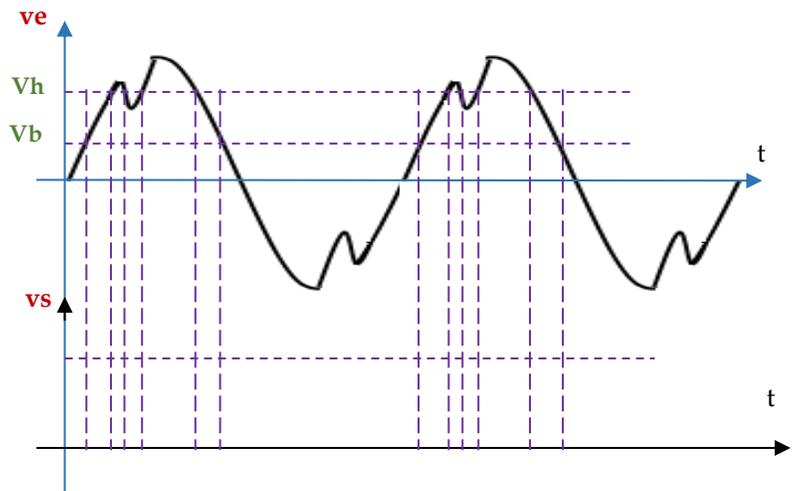
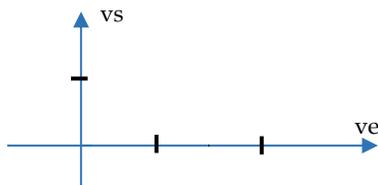
1<sup>er</sup> cas : comparateur à un seuil non inverseur

Si  $v_e < V_{ref}$ ,  $v_s = \dots\dots\dots$

Si  $v_e > V_{ref}$ ,  $v_s = \dots\dots\dots$



2<sup>e</sup> cas : trigger non inverseur

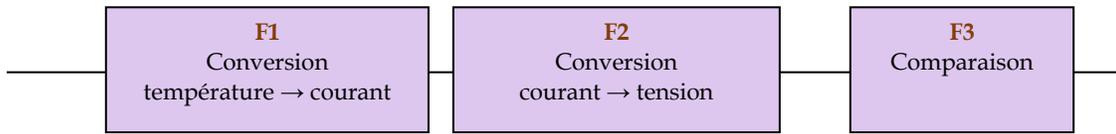


Conclusion : .....

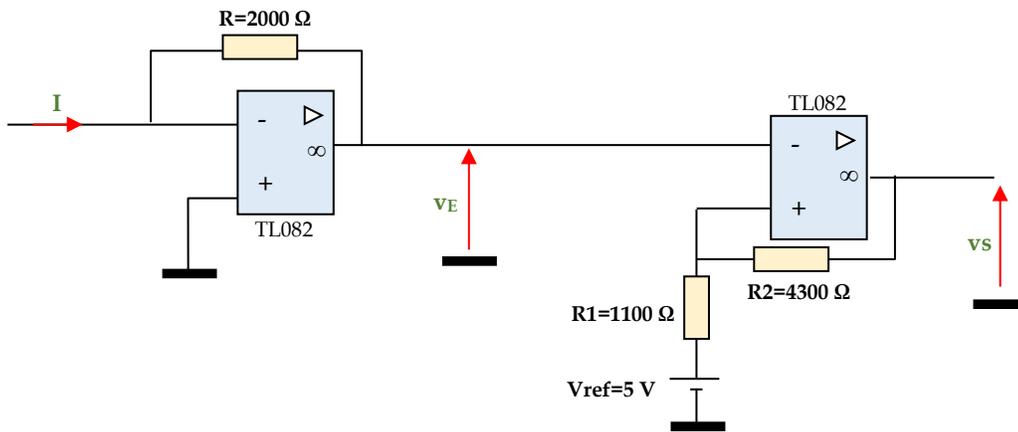
.....

**Exercice 3 : régulation TOR de température**

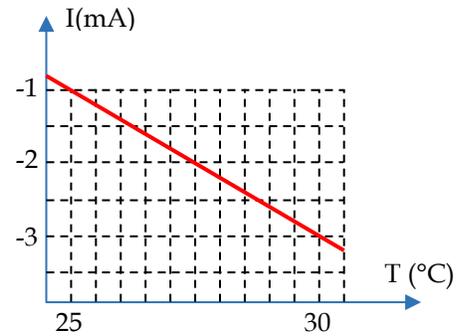
Soit la chaine fonctionnelle :



et le schéma électronique correspondant :



L'étude du capteur matérialisant la fonction F1 n'est pas demandée. On fournit, ci-contre, sa caractéristique de transfert.



Étude fonctionnelle

- Entourer sur le schéma les parties qui matérialisent les fonctions F2 et F3.

Étude de F2

- Exprimer  $V_E$  en fonction de  $I$

.....  
 .....

- Que vaut  $V_E$  pour les températures  $T = 25^\circ\text{C}$  et  $T = 30^\circ\text{C}$  ?

.....  
 .....  
 .....  
 .....

Étude de F3

- Quel est le régime de fonctionnement de l'AOP ?

.....

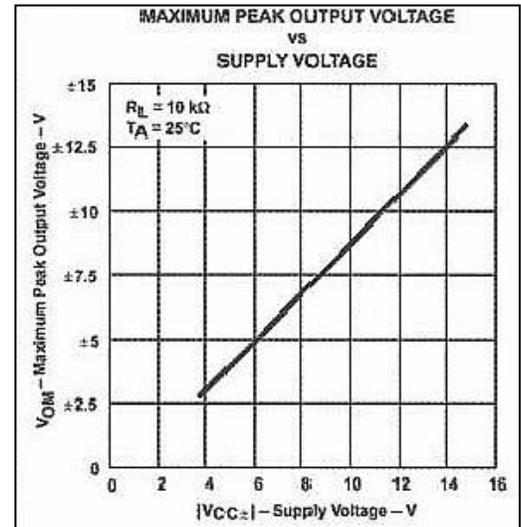
.....

- Cet AOP est un TL082. D'après sa documentation technique, relever la valeur de  $V_{sat}$  (Maximum Peak Output Voltage), sachant qu'il est alimenté (Supply Voltage) sous  $\pm 11V$ .

.....

.....

.....



- Donner les expressions des potentiels  $v_+$  et  $v_-$ .

.....

.....

.....

.....

- Déduire les valeurs des seuils de basculement  $V_{SH}$  et  $V_{SL}$  du comparateur.

.....

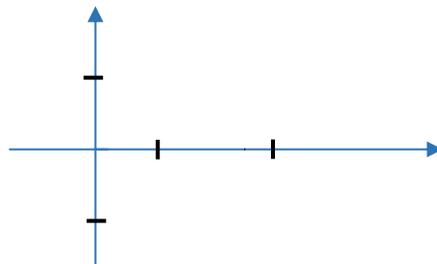
.....

.....

.....

.....

- Tracer le cycle d'hystérésis de  $v_s = f(V_E)$  pour  $V_E$  variant entre 0 et 10 V.



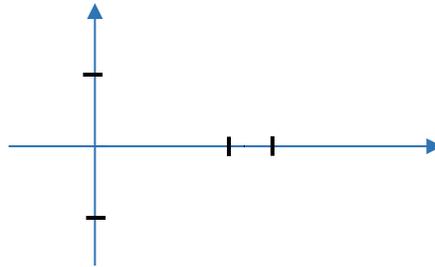
Étude de l'ensemble

- À quelle température correspond chacun des deux seuils de basculement ?

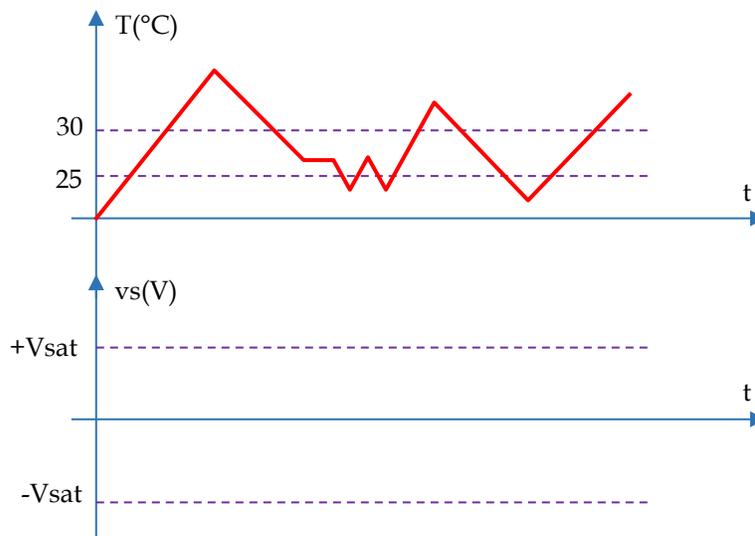
.....

.....

- Tracer le cycle d'hystérésis de  $v_s = f(T)$  pour  $T$  variant entre  $0$  et  $40^\circ\text{C}$ .



- Compléter la courbe de la sortie  $v_s(t)$ .



- Si la sortie  $v_s$  commande le chauffage d'une pièce, expliquer le principe de la régulation de température dans cette pièce.

.....

.....

.....

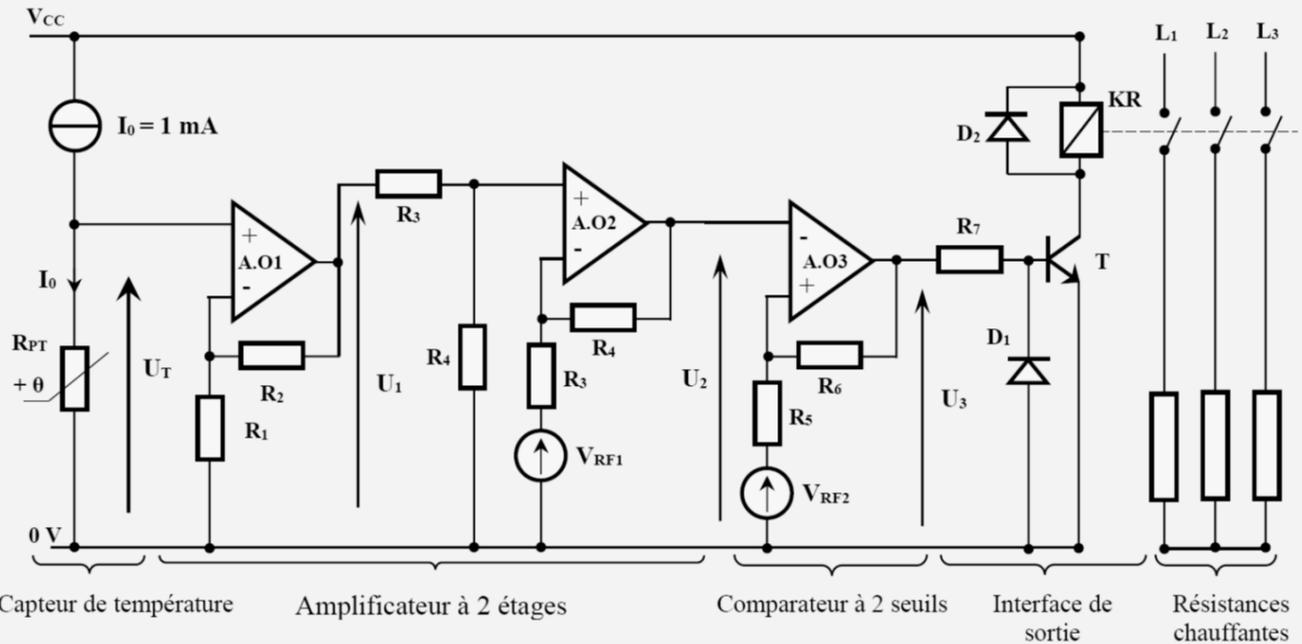
.....

### Exercice 4 : examen national 2020, session normale

Le mélange des produits s'effectue à une température comprise entre  $60^{\circ}\text{C}$  et  $80^{\circ}\text{C}$ .

#### Tâche 1 : Commande du circuit de chauffage

Le chauffage est assuré par 3 résistances chauffantes. La température est prélevée par le capteur PT100 dont la résistance  $R_{PT}$  varie avec la température. Le schéma de commande du dispositif de chauffage est le suivant :



On considère que :

- Les amplificateurs opérationnels sont parfaits avec  $\pm V_{sat} = \pm V_{CC} = \pm 15\text{ V}$  ;
- Le générateur de courant fournit un courant  $I_0 = 1\text{ mA}$  ;
- $V_{RF1}$  et  $V_{RF2}$  sont des tensions continues de référence.

#### Étude du comparateur à 2 seuils (montage autour de AO3) :

Le comparateur à 2 seuils (**trigger inverseur**) agit de la façon suivante :

- Si la température est inférieure à  $60^{\circ}\text{C}$ , le contacteur  $KR$  est fermé donc les résistances chauffantes sont alimentées et la température  $\theta$  augmente ;
- Si la température est supérieure à  $80^{\circ}\text{C}$ , le contacteur  $KR$  est ouvert donc les résistances chauffantes ne sont plus alimentées et la température  $\theta$  baisse.

On donne :

- $U_2 = 100 \cdot 10^{-3} \cdot \theta$ .
- $R_5 = 3,3\text{ K}\Omega$
- L'expression du seuil haut du trigger :  $V_{SH} = \frac{V_{RF2} \cdot R_6 + V_{CC} \cdot R_5}{R_6 + R_5}$ .
- L'expression du seuil bas du trigger :  $V_{SL} = \frac{V_{RF2} \cdot R_6 - V_{CC} \cdot R_5}{R_6 + R_5}$ .

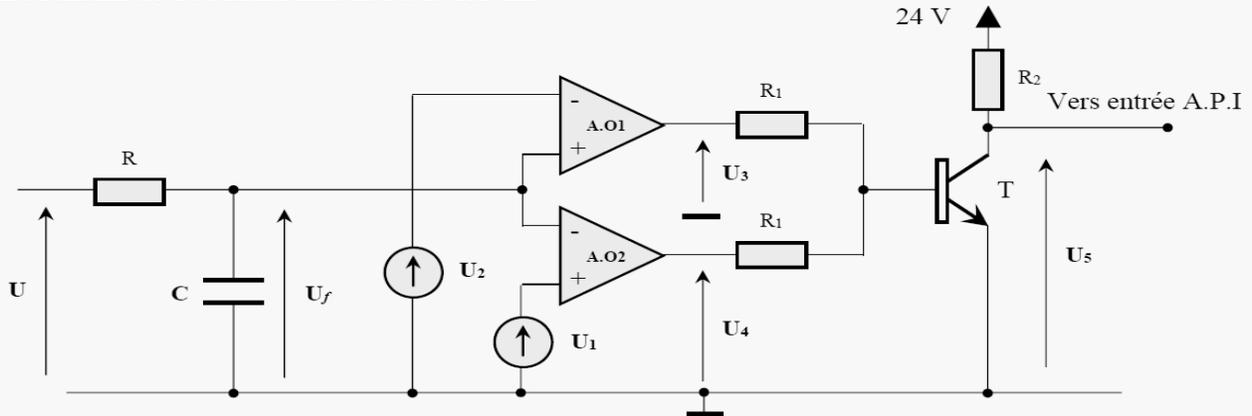


### Exercice 5 : examen national 2020, session de rattrapage

#### Conditionnement du signal

Le circuit de conditionnement du signal de sortie du capteur à ultrasons est constitué d'un filtre, d'un comparateur à fenêtre et d'un adaptateur de tension.

Le schéma de ce conditionneur est le suivant :



Le conditionneur (montage page 07) fournit une information logique lorsque la pièce à détecter se trouve à une distance du capteur comprise entre les distances limites  $d_1$  et  $d_2$ .

Les amplificateurs **A.O1** et **A.O2**, ainsi que le transistor **T** sont considérés parfaits. On donne :

- Tensions de saturation des amplificateurs A.O1 et A.O2 :  $V_{sat}^+ = 12\text{ V}$  et  $V_{sat}^- = 0\text{ V}$
- Tensions de référence :  $U_1 = 1,76\text{ V}$  et  $U_2 = 8,81\text{ V}$

**Q43.** Donner le nom du montage autour de l'**A.O1** et tracer sa caractéristique de transfert ( $U_3$  en fonction de  $U_f$ ). **1,5 pt**

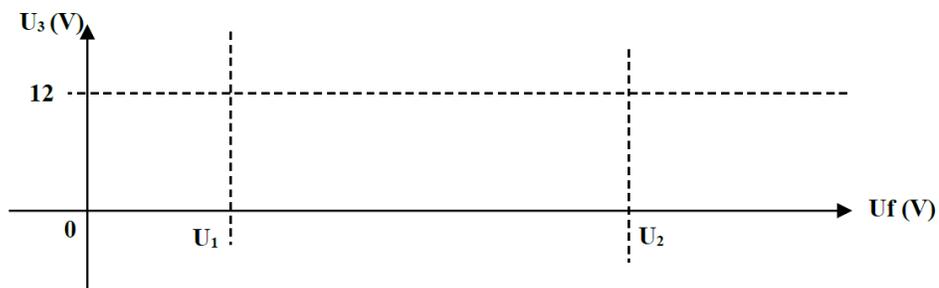
**Q44.** Donner le nom du montage autour de l'**A.O2** et tracer sa caractéristique de transfert ( $U_4$  en fonction de  $U_f$ ). **1,5 pt**

**Q45.** Le transistor **T** fonctionne en commutation. Compléter le tableau des valeurs de  $U_5$  pour les valeurs de  $U_3$  et  $U_4$ . **3 pts**

**Q46.** Tracer, à partir des deux graphes précédents, le graphe  $U_5$  en fonction de  $U_f$ . **1 pt**

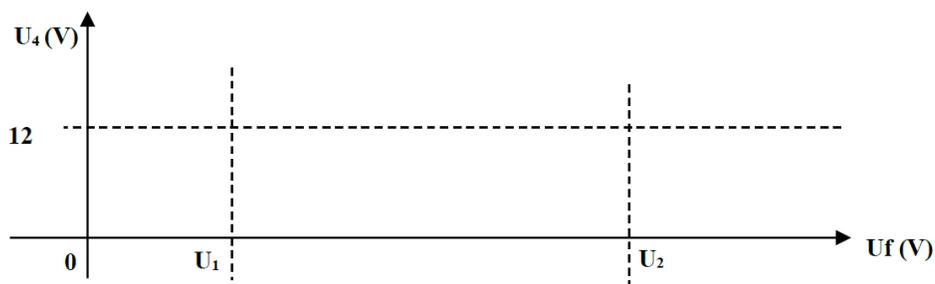
Q43- [1,5 pt ]

Nom du montage : .....



Q44- [1,5 pt ]

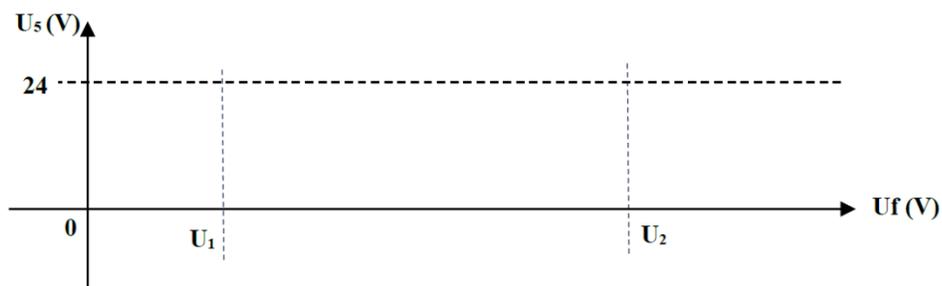
Nom du montage : .....



Q45- [3 pts]

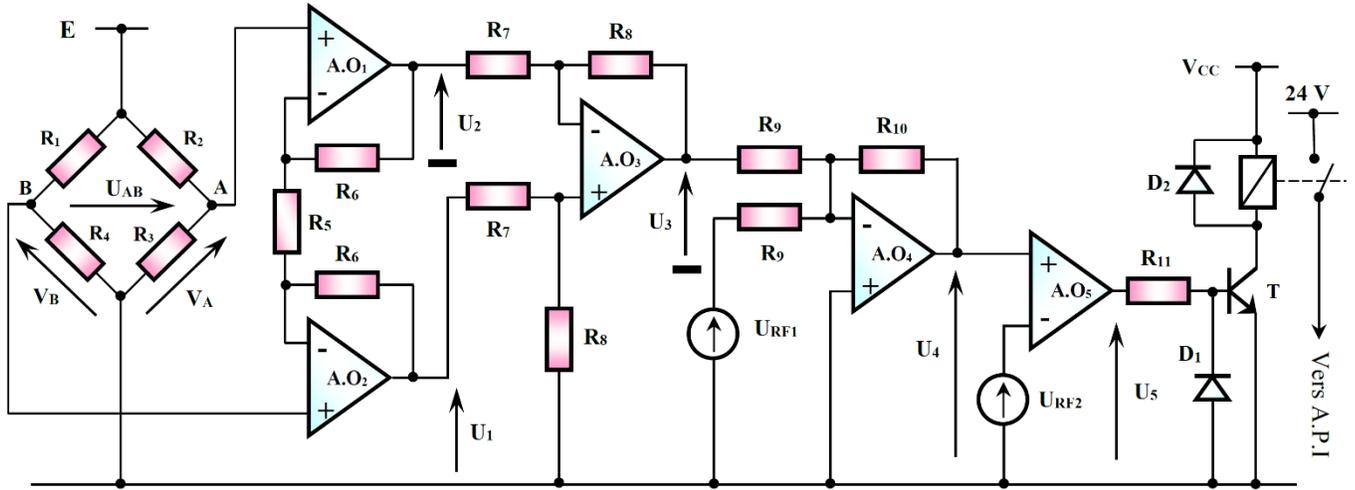
| $U_3(V)$ | $U_4(V)$ | T (bloqué ou saturé) | $U_5(V)$ |
|----------|----------|----------------------|----------|
| 0        | 0        | .....                | .....    |
| 0        | 12       | .....                | .....    |
| 12       | 0        | .....                | .....    |

Q46- [1 pt ]



**Exercice 6 : examen national 2022, session normale**

Le montage ci-dessous représente le circuit de conditionnement du signal  $U_{AB}$  permettant d'informer l'A.P.I en cas de surcharge.



Les amplificateurs opérationnels sont supposés parfaits avec  $\pm V_{SAT} = \pm V_{CC} = \pm 15\text{ V}$ .

$U_4 = 2,02 \cdot 10^{-3} \cdot \frac{R_{10}}{R_9} \cdot m_C$  et  $R_9 = 22\text{ k}\Omega$

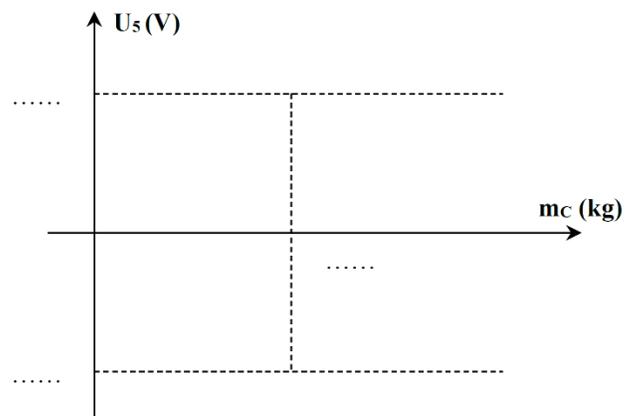
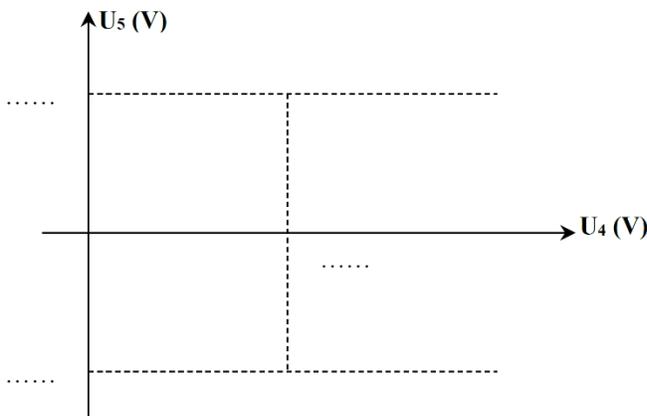
**Q.45 -** Sachant que le comparateur à seuil (montage autour de AO5) bascule lorsque la charge à déplacer  $m_C$  est supérieure à  $700\text{ kg}$  et que  $R_{10} = 68\text{ k}\Omega$ , calculer la valeur de la tension continue de référence  $U_{RF2}$ .

**1,5 pt**

**Q.46 -** Tracer la fonction de transfert du comparateur à seuil  $U_5$  en fonction de  $U_4$ , en déduire alors la caractéristique  $U_5 = f(m_C)$ .

**3 pts**

.....  
 .....  
 .....



## 4

**CONVERTISSEURS**

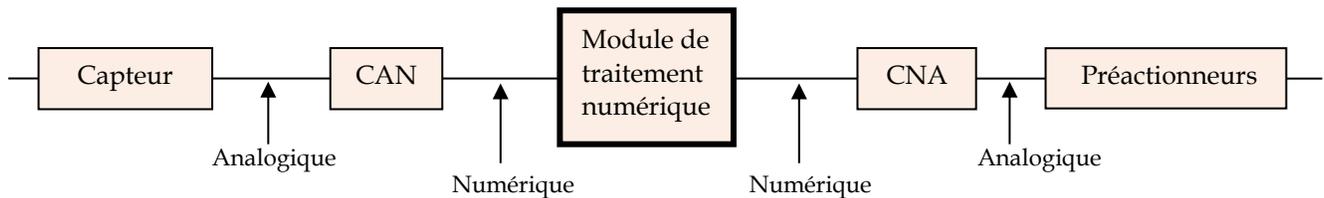
Les convertisseurs permettent de changer la nature d'une grandeur électrique. Ceux étudiés ici sont :

- Les convertisseurs **analogique-numérique** et **numérique-analogique** ;
- La convertisseurs **courant-tension** ;
- Les convertisseurs **tension-fréquence** et **fréquence-tension**.

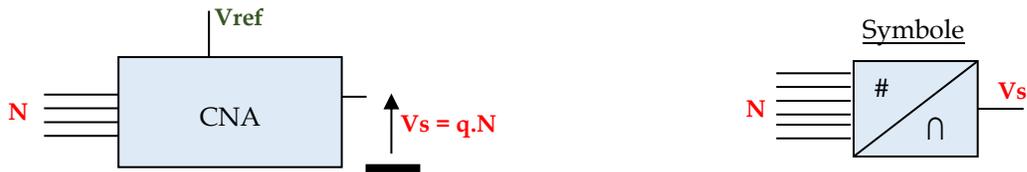
**Conversion numérique-analogique**

De plus en plus, les systèmes techniques utilisent la technologie numérique à base de microprocesseur du fait des avantages qu'elle présente.

Ainsi, lorsque les signaux issus des capteurs sont analogiques ou que les actionneurs doivent être commandés par des signaux analogiques, il est nécessaire de procéder à des conversions.

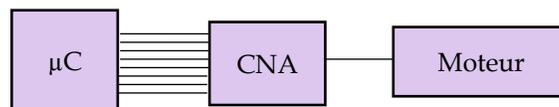


Un CNA transforme une valeur numérique (mot binaire) en une autre analogique (tension ou courant)

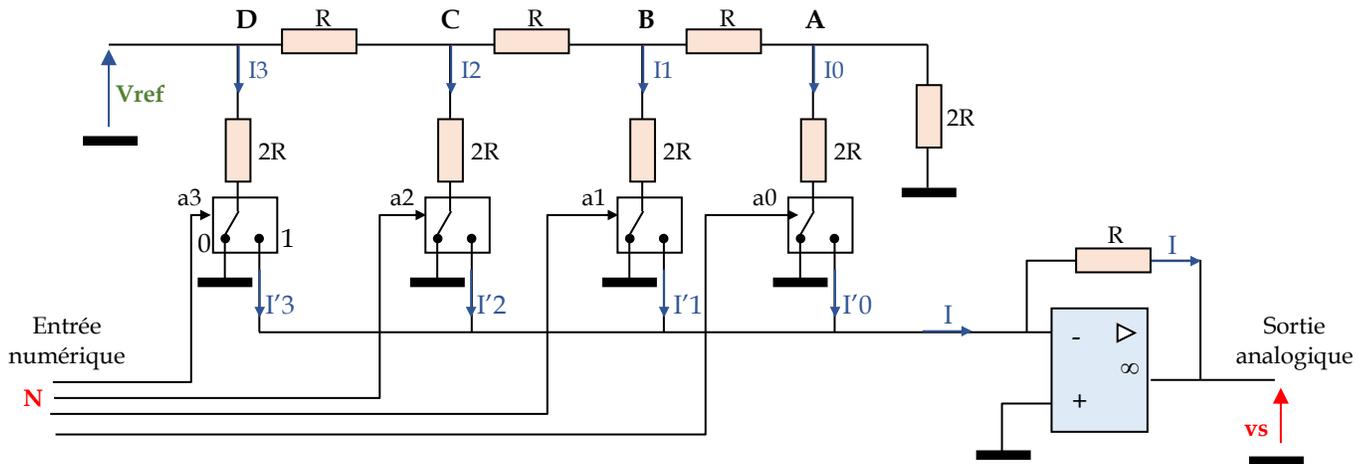


Vref : tension de référence  
 n : nombre de bits du convertisseur  
 q : quantum (ou résolution)

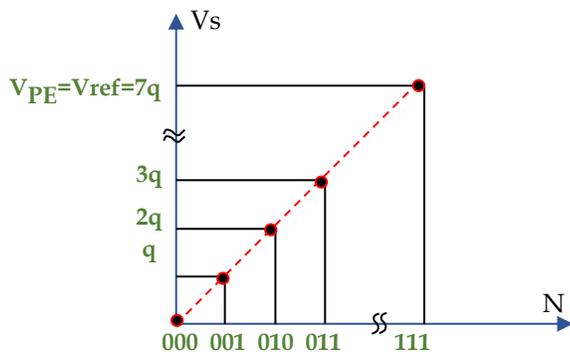
Application : variation de vitesse d'un moteur électrique par microcontrôleur



⇒ CNA à réseau R/2R



Caractéristique de transfert (cas de n = 3 bits)



Quantum q et fonction de transfert

Le quantum q est la plus petite variation de la tension de sortie vs ; vs augmente donc de q lorsque N augmente de 1

|                              |                   |                     |
|------------------------------|-------------------|---------------------|
| $q = \frac{V_{PE}}{2^n - 1}$ | $V_s = q \cdot N$ | $N_{max} = 2^n - 1$ |
|------------------------------|-------------------|---------------------|

n : nombre de bits  
 V<sub>PE</sub> : tension pleine échelle (tension maximale que peut prendre la sortie), elle est fixée par Vref c'est à dire V<sub>PE</sub> = Vref

**Exercice 1** On applique Vref = 8 V à un CNA 6 bits. Calculer Vs pour N = 100101, 001001 et 111111

.....

.....

.....

.....

.....

**Exercice 2** Sachant que Vref = 10 V et que c'est CNA 8 bits, calculer Vs pour N = 10011001 et N = 00010110

.....

.....

.....

.....

**Exercice 3**

Quelle est la **pleine échelle** d'un CNA qui fournit  $v_s = 1V$  pour  $N = 00010111$  ?

.....

.....

.....

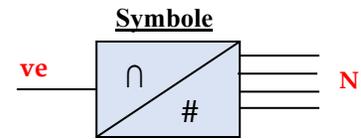
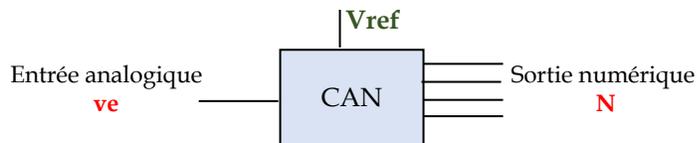
.....

.....

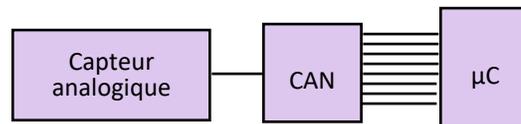
.....

### Conversion analogique-numérique

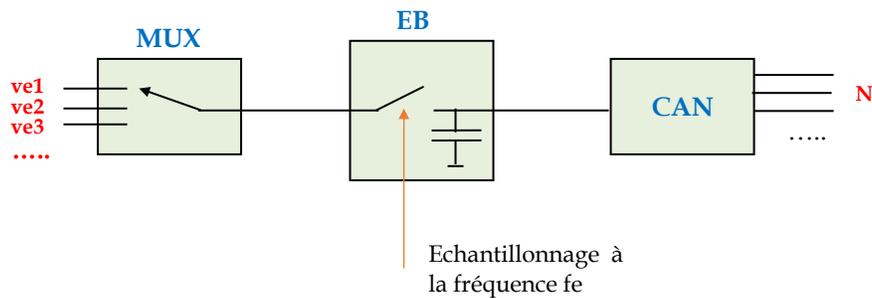
Le CAN permet d'obtenir en sortie une grandeur numérique codée sur  $n$  bits dont la valeur est représentative de la grandeur analogique présente à l'entrée.



Application : traitement numérique du signal issu d'un capteur analogique



### Chaîne de conversion



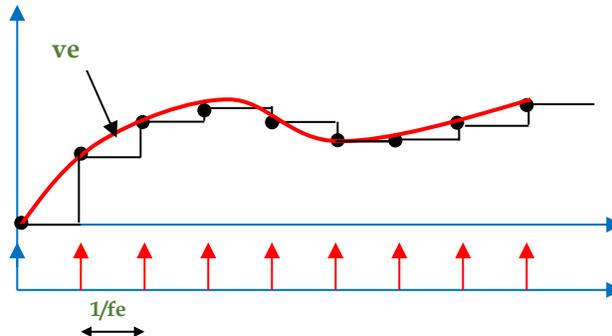
MUX : multiplexeur

Le multiplexeur est nécessaire lorsqu'il s'agit de convertir des signaux analogiques provenant de plusieurs sources. Il permet d'utiliser, en temps réel, un même convertisseur CAN.

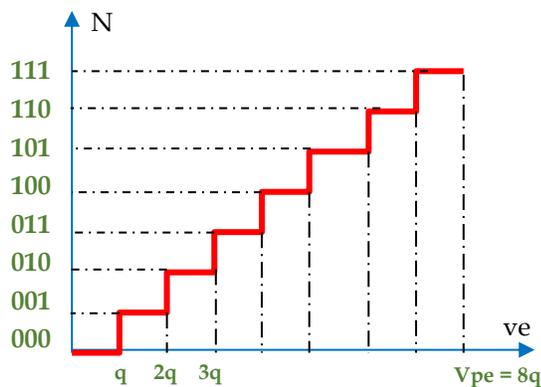
EB : échantillonneur bloqueur

L'EB assure l'échantillonnage qui consiste à prélever, à divers instants, une valeur de l'entrée  $v_e$  et la bloquer pour la conversion puisque celle-ci n'est pas instantanée.

- Le prélèvement d'un échantillon s'obtient par fermeture de l'interrupteur électronique ;
- Pendant que l'interrupteur s'ouvre, le condensateur conserve l'échantillon lors de la conversion.



$f_e$  : fréquence d'échantillonnage

Caractéristique de transfert (Cas de  $n=3$  bits)Quantum et fonction de transfert

Le quantum  $q$  est la plus petite variation de  $v_e$  qui provoque une incrémentation du code numérique  $N$  de sortie.

$$q = \frac{V_{PE}}{2^n}$$

$$V_e = q \cdot N$$

$q$  : quantum  
 $n$  : nombre de bits

$V_{PE}$  : tension pleine échelle (tension maximale tolérée en entrée)

**Exercice**

On applique à l'entrée d'un CAN 8 bits une tension d'entrée  $v_e = 7,224V$ , la tension de référence est de 10 V.

- Calculer le quantum
- Déterminer, en décimal puis en binaire, le résultat  $N$  de la conversion

.....

.....

.....

.....

.....

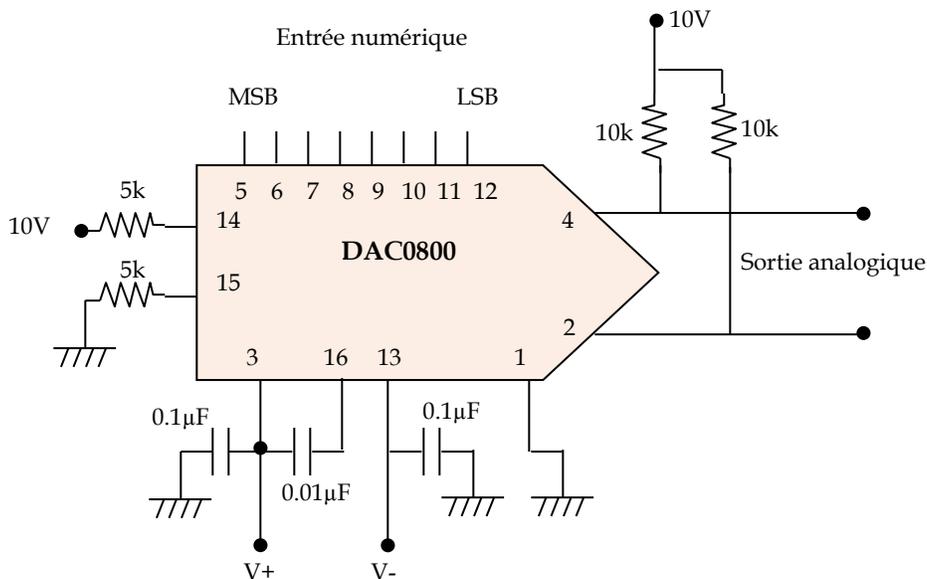


## Conversion analogique ↔ numérique

### Réalisation à l'aide de circuits intégrés

1. Le convertisseur numérique analogique DAC0800 de Motorola est un convertisseur 8 bits rapide. La tension de sortie peut atteindre 20 V.

#### Symbole et câblage minimal du DAC0800



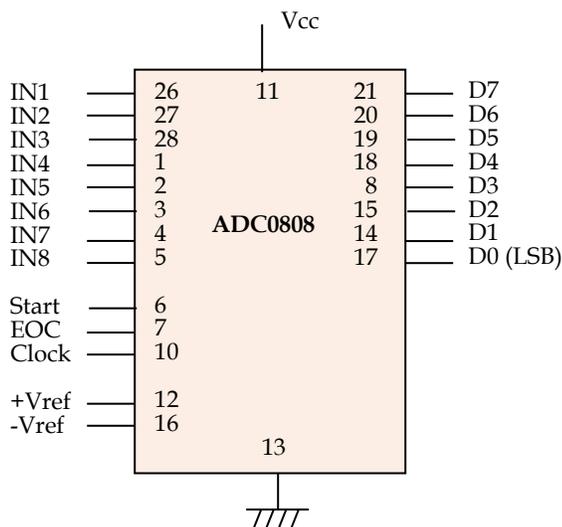
MSB : bit du poids fort  
LSB : bit du poids faible

#### Caractéristiques

Temps de conversion : 100 ns  
Tension de sortie jusqu'à 20 V  
Alimentation de  $\pm 4.5$  V à  $\pm 18$  V  
Consommation : 33 mW à  $\pm 5$  V  
Coût modéré

2. Le convertisseur analogique numérique ADC0808 est un convertisseur à approximations successives 8 bits muni d'un multiplexeur 8 voies. Le CNA interne est à base du réseau R/2R.

#### Principales broches du ADC0808



IN1 ... IN8 : entrées analogiques multiplexées  
D0...D7 : sortie numérique  
Start et EOC : signaux de début et fin de conversion  
+Vref et -Vref : tensions de référence qui permettent de fixer les valeurs min et max de l'entrée

#### Caractéristiques

Alimentation :  $V_{ccmax} = 5$  V  
Faible consommation : 15m W  
Durée d'un cycle de conversion : 100  $\mu$ s  
Température de fonctionnement :  $-40$  °C à  $+85$  °C  
Erreur :  $\pm \frac{1}{2}$  LSB  
Nombre de broches : 28

**Exercice**

Le microcontrôleur PIC 16F877 est doté d'un CAN qui permet de traiter des informations analogiques. Ce CAN a une résolution de **10 bits** et une tension de référence de **5V**.

- Calculer le quantum de ce convertisseur

.....

- Compléter ce tableau de conversion

| ve(V) | N décimal | N binaire |     |     |    |    |    |   |   |   |   |
|-------|-----------|-----------|-----|-----|----|----|----|---|---|---|---|
|       |           | 512       | 256 | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |
| 1     | .....     | .....     |     |     |    |    |    |   |   |   |   |
| 2     | .....     | .....     |     |     |    |    |    |   |   |   |   |
| 3     | .....     | .....     |     |     |    |    |    |   |   |   |   |
| 4     | .....     | .....     |     |     |    |    |    |   |   |   |   |
| 5     | .....     | .....     |     |     |    |    |    |   |   |   |   |

- Le convertisseur étant à approximations successives, compléter le tableau de conversion correspondant à l'entrée  $v_e = 3V$

| N     |       |       |       |       |       |       |       |       |       |       | v = q.N |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|---------|
| 512   | 256   | 128   | 64    | 32    | 16    | 8     | 4     | 2     | 1     |       |         |
| ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | .....   |
| ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | .....   |
| ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | .....   |
| ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | .....   |
| ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | .....   |
| ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | .....   |
| ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | .....   |
| ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | .....   |
| ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | .....   |
| ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | .....   |
| ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | .....   |

Résultat de la conversion : .....

**Exercice : examen national 2019, session normale**

**Tâche 4 : Etude de la fonction FP4**

On utilise un convertisseur analogique numérique (CAN) pour convertir la tension  $V_4$ , image de la force  $F$ , en un signal numérique. On donne l'expression du nombre  $N$  résultat de la conversion en fonction de  $V_4$  et du quantum  $q$  :  $N = V_4/q$

**Question : 40.** Sachant que  $N = 200$  pour  $V_4 = 1$  V et que  $N_{\max} = 1023$ , calculer la valeur pleine échelle PE définie par  $PE = q.(N_{\max} + 1)$  ; [2 pts]

**Question : 41.** Calculer la résolution numérique (nombre de bits)  $n$  du CAN. [2 pts]

**Exercice : examen national 2021, session de rattrapage**

Numérisation :

La tension  $V_2$  image du couple  $T$  est numérisée par un convertisseur analogique numérique (CAN 8 bits) avant d'être utilisée par l'A.P.I. On donne :

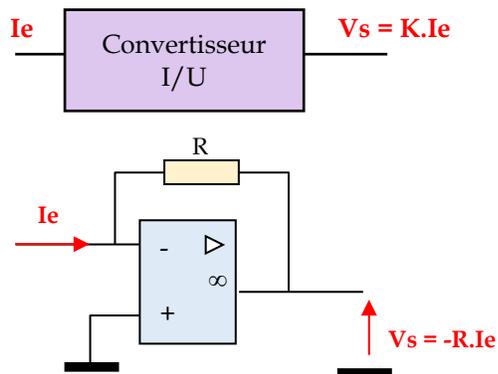
- La valeur numérique  $N$  de la conversion correspondante au couple  $T$  :  $N = \frac{V_2 - V_{REF-}}{V_{REF+} - V_{REF-}} \cdot 255$  ;
- $V_{REF+} = 10$  V ;  $V_{REF-} = 0$  V ;
- Pour les signaux de basse fréquence :  $V_2 = 20 \cdot 10^{-3} \cdot T$  (en Volt).

**Q48-** Calculer la résolution analogique  $q$  (quantum) de ce convertisseur en mV. [1 pt]

**Q49-** Déterminer les valeurs  $N1$  et  $N2$  du mot  $N$  (valeurs en décimal) correspondantes aux couples :  $T_1 = 240$  N.m et  $T_2 = 300$  N.m. [2 pts]



## Conversion courant-tension



Ce convertisseur fournit une tension proportionnelle au courant qu'il reçoit.

Il peut être utilisé pour obtenir une sortie tension à partir d'un circuit à sortie courant comme une photodiode, un CNA à sortie courant...

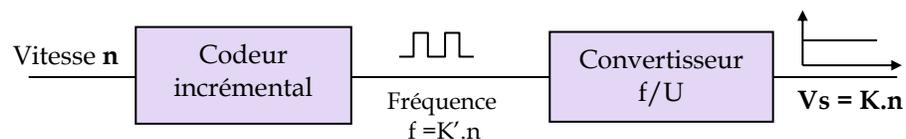
## Conversion fréquence-tension



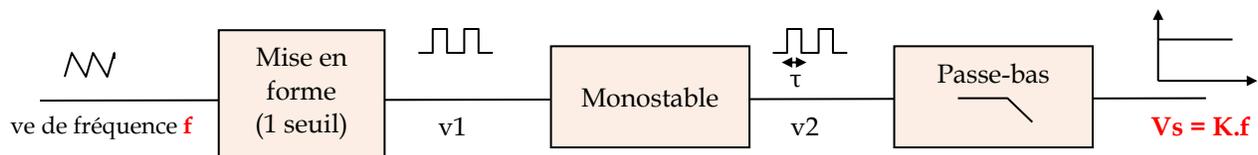
Permet de produire une tension continue proportionnelle à la fréquence du signal d'entrée.

### Exemple d'application

Conversion en tension continue du signal en créneaux de sortie d'un codeur incrémental.



### Principe

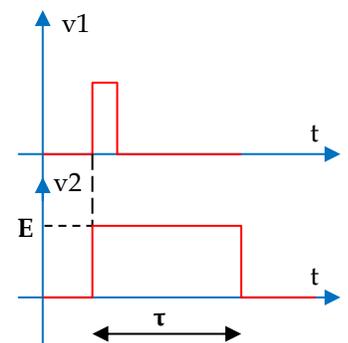


- Le **monostable** génère des impulsions calibrées de durée  $\tau$  déterminée et d'amplitude  $E$  ;
- Le **filtre passe-bas** permet d'extraire la **valeur moyenne** du signal délivré par le monostable.

En effet selon la loi de Fourier, tout signal périodique est décomposable en sa composante continue et une série de composantes sinusoïdales.

$$V(t) = V_{\text{moy}} + V_1 \cdot \sin(2\pi f_1 \cdot t + \varphi_1) + V_2 \cdot \sin(2\pi f_2 \cdot t + \varphi_2) + \dots$$

Un filtre passe-bas de fréquence de coupure inférieure à la fréquence  $f_1$ , laisse passer seule la valeur moyenne  $V_{\text{moy}}$  (composante continue).

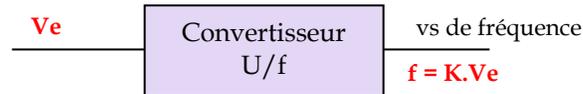




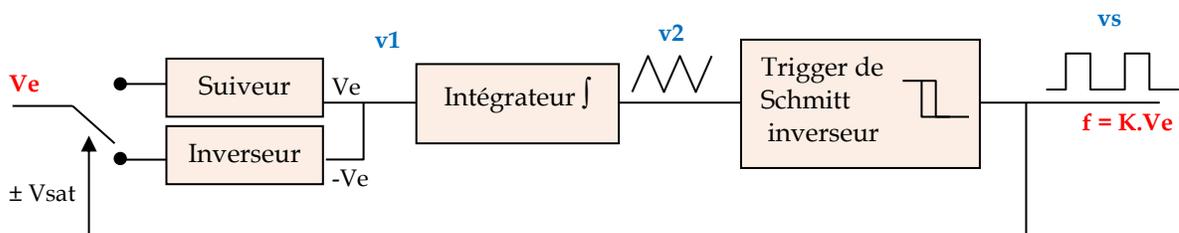
## Conversion tension-fréquence

Cette conversion permet de produire un signal périodique de fréquence proportionnelle à la tension d'entrée.

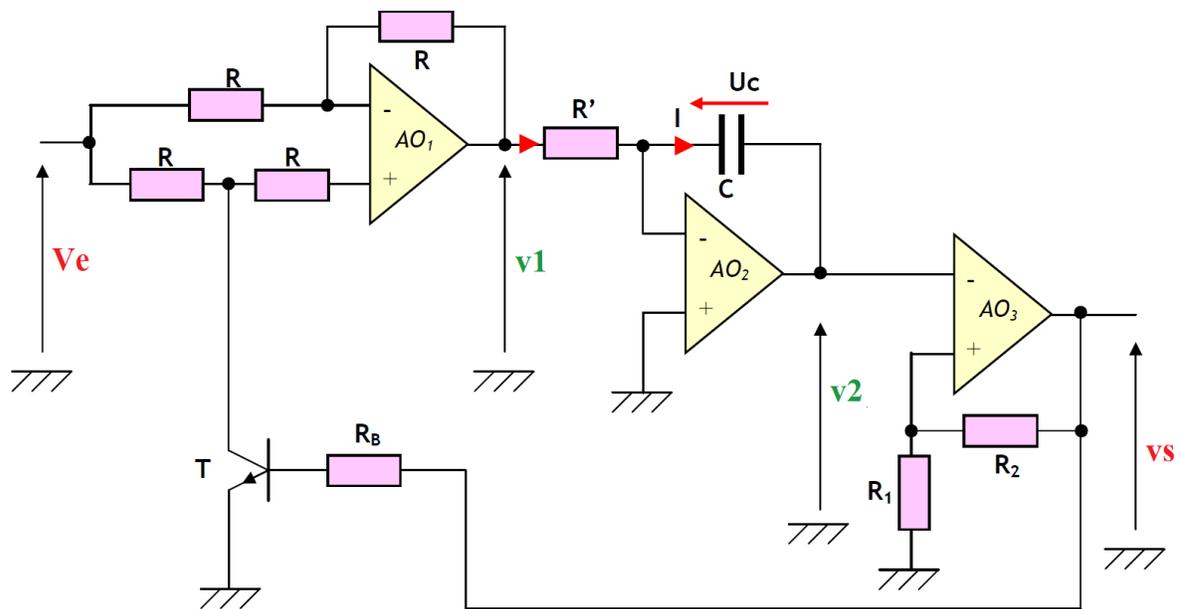
Application : réglage de la fréquence d'un générateur de signaux



⇒ Synoptique de la conversion



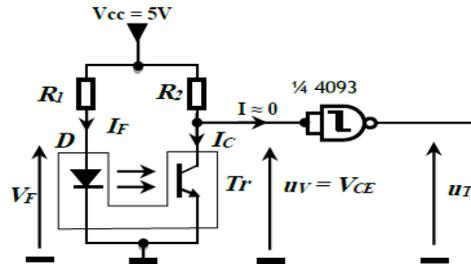
⇒ Exemple de réalisation



**Exercice : examen national 2018, session de rattrapage**
**Tâche 3 : Mesure de la vitesse**

La vitesse de rotation du moteur est acquise au moyen d'un codeur optique incrémental qui génère un signal carré de fréquence proportionnelle à la vitesse du moteur. Ce codeur est constitué :

- D'un disque comportant 72 trous (figure a) et fixé sur l'arbre de sortie du réducteur ;
- D'un détecteur optoélectronique à fourche lié au bâti et alimenté comme indiqué dans la figure b.


**Figure a**

**Figure b**

Les éléments du détecteur ont pour caractéristiques principales:

- Intensité du courant dans la diode électroluminescente :  $I_F = 20 \text{ mA}$  ;
- Tension aux bornes de la diode **D** électroluminescente :  $V_F = 1,5 \text{ V}$  ;
- Intensité maximale du courant dans le collecteur du phototransistor **Tr** lorsqu'il est saturé :  $I_{C\text{MAX}} = 0,5 \text{ mA}$  ;
- Tension  $V_{\text{CESAT}}$  entre le collecteur et l'émetteur du phototransistor **Tr** lorsqu'il est saturé :  $V_{\text{CESAT}} \approx 0 \text{ V}$ .

**Q31)** Calculer les valeurs des résistances  $R_1$  et  $R_2$ .

4 pts

**Q32)** Suivant l'état du phototransistor (saturé ou bloqué), donner la valeur de la tension  $u_V$  en volts.

1 pt

**Q33)** On admet que la fréquence  $f$  du signal  $u_V$  est liée à la vitesse  $N_a$  du disque du codeur par la relation :

$$f = R \cdot N_a \text{ où } R = 72 \text{ (résolution du codeur) et } N_a \text{ étant en tr/s.}$$

Montrer que la fréquence  $f$  du signal  $u_V$  est liée à la vitesse du moteur  $N_{\text{mot}}$  (tr/min) par :  $f = \frac{N_{\text{mot}}}{60} \cdot K \cdot R$

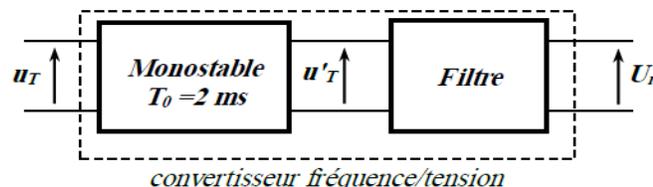
où  $K = 1/25$  est le rapport de réduction.

2 pts

**Q34)** Sachant que la vitesse du moteur  $N_{\text{mot}}$  varie entre 200 et 4000 tr/min, calculer alors les fréquences  $f_{\text{min}}$  et  $f_{\text{max}}$  correspondantes.

2 pts

Après la mise en forme du signal  $u_V$ , le signal  $u_T$  est appliqué à l'entrée d'un convertisseur fréquence/tension constitué d'un monostable suivi d'un filtre comme l'indique la figure suivante :



Le monostable génère une impulsion de 2 ms à chaque front montant de  $u_T$ .

Les chronogrammes des signaux  $u_T$  et  $u'_T$  sont représentés sur le document DRES 04.

On admet pour la suite que la fréquence  $f$  du signal  $u_T$  varie entre  $f_{\text{min}} = 10 \text{ Hz}$  et  $f_{\text{max}} = 200 \text{ Hz}$ .

**Q35)** Exprimer  $U'_{T\text{moy}}$  (valeur moyenne de  $u'_T$ ) en fonction de  $f$ .

1 pt

**Q36)** La tension  $u'_T$  peut être considérée comme la somme d'une composante continue  $U'_{T\text{moy}}$  (de fréquence nulle) et de plusieurs tensions sinusoïdales dont les fréquences sont supérieures ou égales à la fréquence  $f$ .

Quel est le type de filtre (passe-bas, passe-haut ou passe-bande) qui convient pour avoir une tension continue  $U_r = U'_{T\text{moy}}$  ?

1 pt

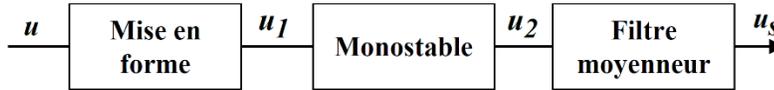


**Exercice : examen national 2022, session de rattrapage**

⇒ Conditionnement

Le circuit de conditionnement du signal du capteur est conçu à base d'un convertisseur **fréquence-tension** et délivre une tension continue  $u_s$  proportionnelle à la vitesse de rotation  $n$  du moteur.

Voici, ci-dessous, une présentation synoptique du principe de la conversion fréquence-tension :

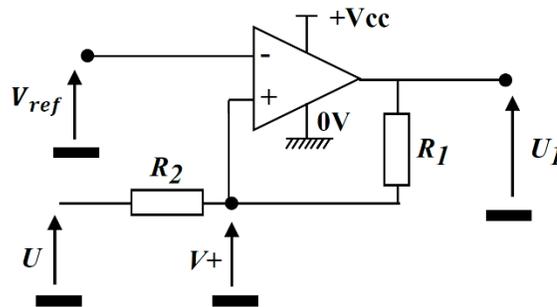


Mise en forme

Le circuit de mise en forme est un comparateur à deux seuils (trigger non inverseur) à base d'amplificateur opérationnel.

On admet que l'amplificateur opérationnel est parfait et que ses tensions de saturation sont  $+V_{cc}$  et  $0\text{ V}$ .

$V_{ref}$  est une tension de référence continue positive



Q.36) Donner l'expression de  $V^+$  en fonction de  $U$ ,  $U_1$ ,  $R_1$  et  $R_2$  ;

1 pt

Q.37) En déduire que, au moment du basculement,  $U = \frac{(R_1 + R_2)}{R_1} \cdot V_{ref} - \frac{R_2}{R_1} \cdot U_1$  ;

1 pt

Q.38) En déduire les expressions des seuils de basculement  $V_B$  (seuil bas) et  $V_H$  (seuil haut) ;

1 pt

Q.39) Compléter le tracé de la caractéristique de transfert  $u_1 = f(u)$  en indiquant les sens d'évolution ;

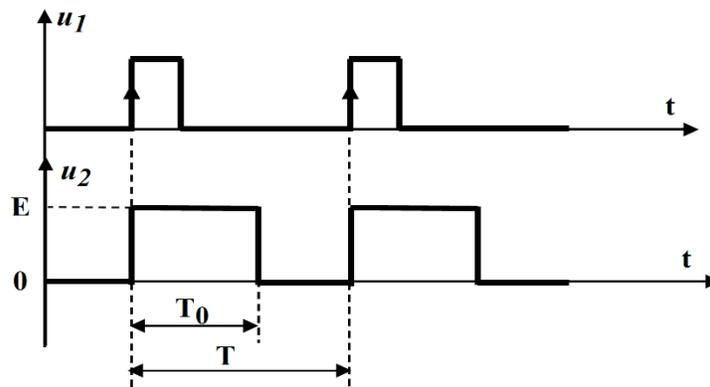
1,5 pt

Q.40) Tracer la courbe  $u_1(t)$  en concordance de temps avec le signal  $u(t)$  fourni.

2 pts

Monostable

Le circuit monostable génère à sa sortie une impulsion de durée calibrée  $T_0$  et d'amplitude  $E$  à chaque fois qu'il reçoit un front sur son entrée. Les chronogrammes suivants illustrent la réponse  $u_2$  d'un monostable actif sur le front montant :



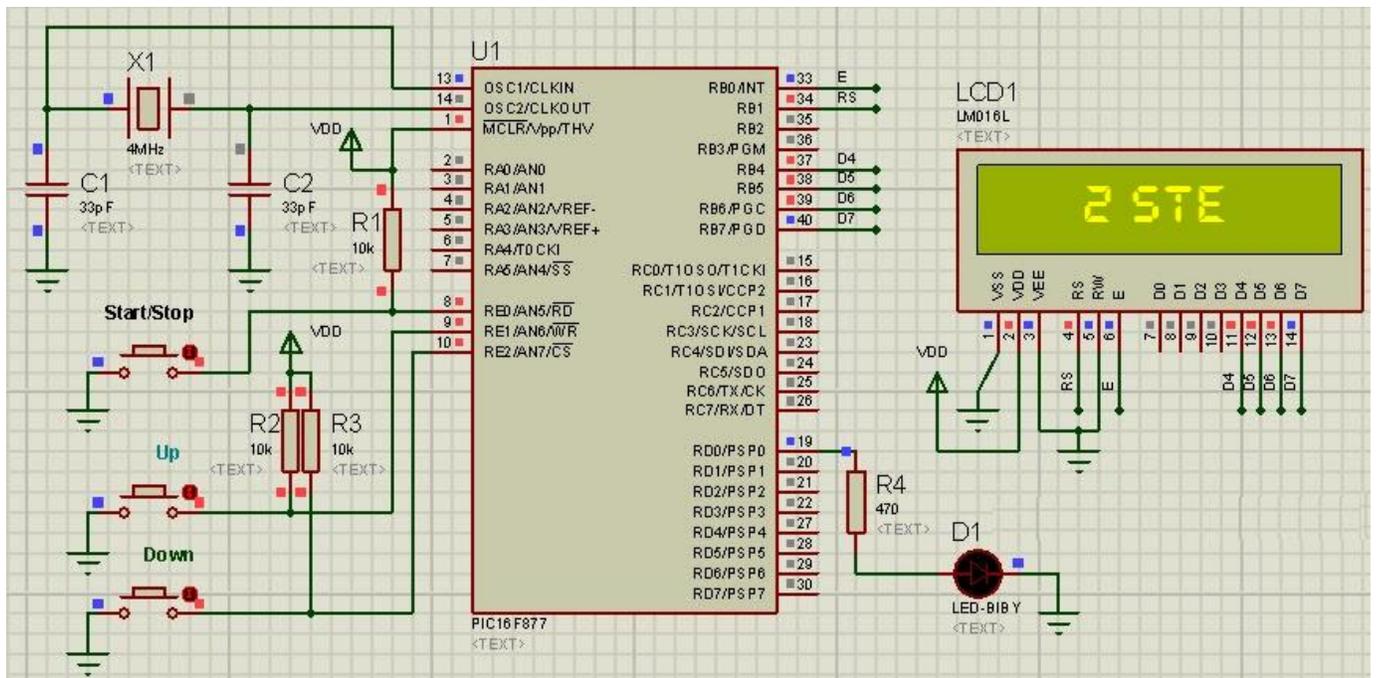




PARTIE

2

# La fonction TRAITER



## 5

**CIRCUITS LOGIQUES PROGRAMMABLES**

Un circuit logique programmable **PLD** (Programmable Logic Device) est un composant logique dont on peut programmer le schéma selon la fonction souhaitée, combinatoire et/ou séquentielle.

Il apporte un gain dans la mesure où :

- Un seul circuit peut remplacer plusieurs ;
- La taille et le coût de la fonction implantée sont moindres.

**Architecture d'un PLD**

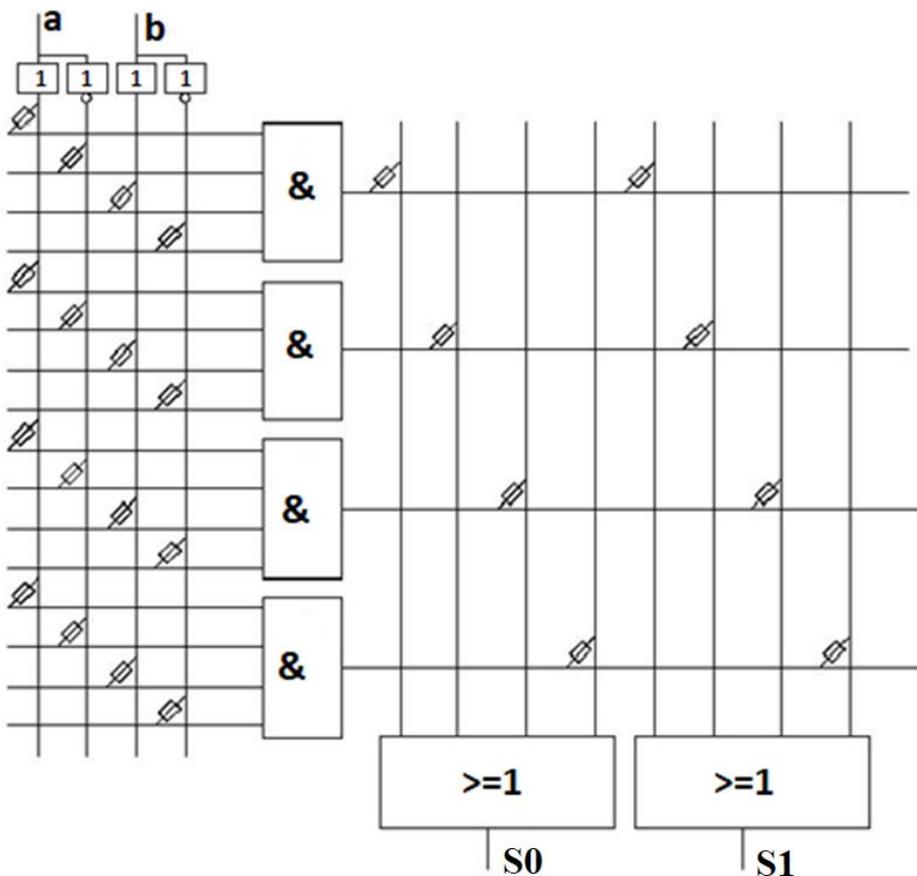
Toute fonction logique peut être réalisée par seules les portes logiques fondamentales ET, OU et NON.  
Un PLD est bâti autour de ce concept.

La plupart des PLD adoptent l'architecture suivante :

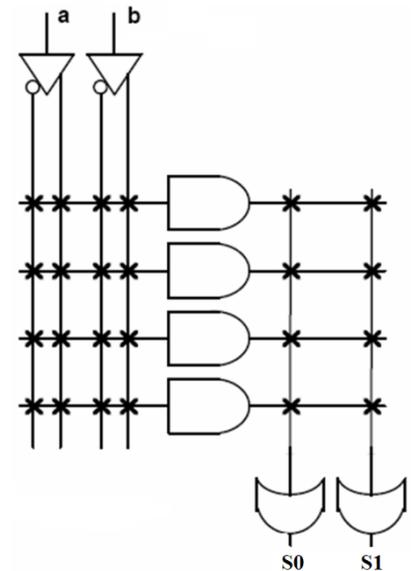
- Une matrice d'opérateurs ET sur lesquels viennent se connecter les variables d'entrée ;
- Une matrice d'opérateurs OU sur lesquels les sorties des opérateurs ET sont connectées ;
- Une éventuelle structure de sortie (Portes inverseuses, bascules...).

Schéma de principe très simplifié (cas d'un PLD à 2 entrées et 2 sorties combinatoires)

⇒ Avant la programmation



Représentation simplifiée

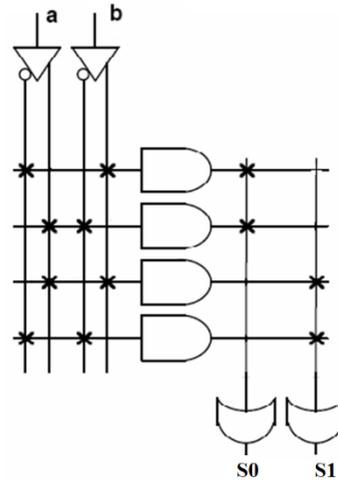


⇒ Après programmation

Les fonctions programmées ci-contre sont :

$$S0 = \bar{a}b + a\bar{b}$$

$$S1 = ab + \bar{a}\bar{b}$$



### Exercice : générateur de parité à 3 entrées

S = 1 si le nombre de 1 à l'entrée est impair ;  
S = 0 sinon.

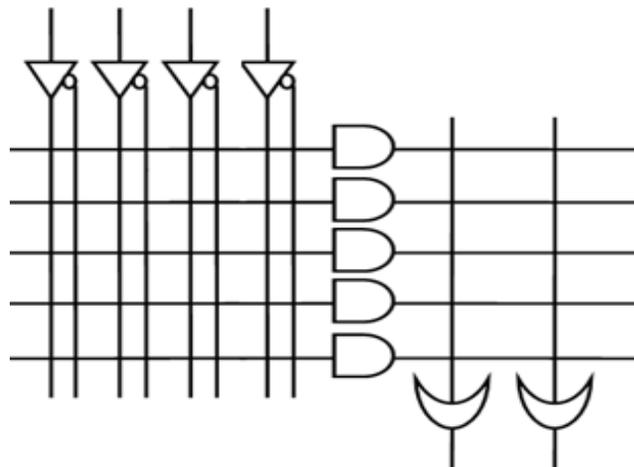
Table de vérité

| a | b | c | S     |
|---|---|---|-------|
| 0 | 0 | 0 | ..... |
| 0 | 0 | 1 | ..... |
| 0 | 1 | 0 | ..... |
| 0 | 1 | 1 | ..... |
| 1 | 0 | 0 | ..... |
| 1 | 0 | 1 | ..... |
| 1 | 1 | 0 | ..... |
| 1 | 1 | 1 | ..... |

Équation

S = .....

État du PLD après programmation





### ⇒ Structure d'un programme ABEL

On y trouve :

- La section d'entête → **module, title** ;
- La section des déclarations → **declarations** (device, E/S, bus...) ;
- La section de description → **equations, truth\_table, state\_diagram** ;
- La déclaration de fin du fichier → **end**.

### ⇒ Déclarations

- **Pin** : permet d'affecter une broche d'E/S à une variable.

Exemples

|                          |   |
|--------------------------|---|
| e1, e2, A1 pin 2, 3, 4;  | : entrées affectées aux broches 2, 3 et 4     |
| d4 pin ;                 | : entrée sans désignation de numéro de broche |
| s1 pin 12 istype 'com';  | : sortie combinatoire associée à la broche 12 |
| S0..S6 pin istype 'com'; | : sorties combinatoires                       |
| S1 pin istype 'reg';     | : sortie séquentielle                         |

- On peut déclarer un bus ; il s'agit d'un ensemble de variables

Exemples **A = [b7..b0]** ;      **N = [D2, D1, D0]** ;  
L'affectation N = 5 entraîne D2 = 1, D1 = 0 et D0 = 1.

### ⇒ Opérateurs

| <u>Opérateurs logiques</u> | <u>Opérateurs relationnels</u> | <u>Opérateurs arithmétiques</u>   |
|----------------------------|--------------------------------|-----------------------------------|
| <b>!</b> NON               | <b>==</b> Egal                 | <b>-</b> négation ou soustraction |
| <b>&amp;</b> ET            | <b>!=</b> Différent            | <b>+</b> Addition                 |
| <b>#</b> OU                | <b>&lt;</b> Inférieur          | <b>*</b> Multiplication           |
| <b>\$</b> OU exclusif      | <b>&lt;=</b> Inférieur ou égal | <b>/</b> Division                 |
|                            | <b>&gt;</b> Supérieur          | <b>%</b> Reste de la division     |
|                            | <b>&gt;=</b> Supérieur ou égal |                                   |

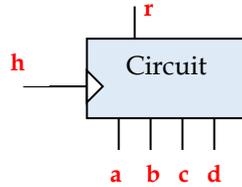
### ⇒ Instructions d'affectation et de test

- **S = a & !b # !a & b;**      affectation pour une sortie combinatoire (symbole = )  
**A0 := !D1 & A0 & !D7;**      affectation pour une sortie séquentielle (symbole := )  
**X = a < b**      X reçoit 1 si a < b et X reçoit 0 sinon
- La structure conditionnelle      **WHEN (condition) THEN équation1 ; ELSE équation2 ;**  
Exemple      when (E1 == 0) then S = a # b ; else S = a \$ b ;

### ⇒ Extensions

Les pins peuvent être suivis d'une extension leur associant une fonction particulière.

|                               |                                       |
|-------------------------------|---------------------------------------|
| <b>.CLK</b> : Entrée horloge  | <b>.D</b> : Entrée D d'une bascule D  |
| <b>.AR</b> : Reset asynchrone | <b>.J</b> : Entrée J d'une bascule JK |
| <b>.OE</b> : Output Enable    | <b>.K</b> : Entrée K d'une bascule JK |



Dans cet exemple, on déclare que :

- L'entrée **h** est l'horloge des bascules de sortie ;
- L'entrée **r** effectue une mise à 0 asynchrone des sorties.

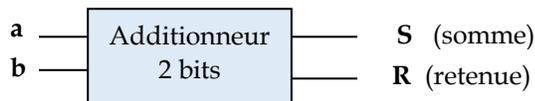
```

..
Declarations
  h, r pin ;
  a, b, c, d pin istype 'reg';
  S = [a, b, c, d];
Equations
  S.CLK = h ;
  S.AR = r ;
..

```

## Exercices

### Exercice : additionneur 2 bits



| a | b | S     | R     |
|---|---|-------|-------|
| 0 | 0 | ..... | ..... |
| 0 | 1 | ..... | ..... |
| 1 | 0 | ..... | ..... |
| 1 | 1 | ..... | ..... |

S = .....

R = .....

#### Description par la table de vérité

```

MODULE additionneur
DECLARATIONS
  additionneur DEVICE 'P22V10' ;
  a, b PIN ;
  S, R PIN ISTYPE 'COM' ;
TRUTH_TABLE ( [ ..... ] ->[.....] )
  .....
  .....
  .....
  .....
END additionneur

```

#### Description par les équations logiques

```

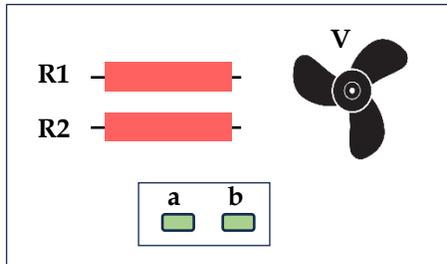
MODULE additionneur
DECLARATIONS
  additionneur DEVICE 'P22V10' ;
  a, b PIN 4, 5 ;
  S, R PIN 15, 16 ISTYPE 'COM' ;
EQUATIONS
  S = .....
  R = .....
END additionneur

```

**Exercice : radiateur de chauffage électrique**

Le système de chauffage est équipé d'un radiateur électrique à deux allures de chauffe, comportant deux résistances **R1** et **R2** et un ventilateur **V**. Il est commandé par deux interrupteurs **a** et **b**.

- Par action sur **a** seul, la résistance **R1** seule est mise sous-tension ;
- Par action sur **b** seul ou sur **a** et **b** à la fois, les **deux résistances** et le **ventilateur** sont mis sous-tension.



```

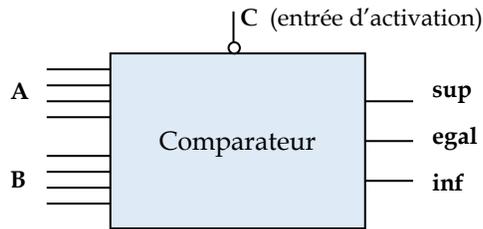
module .....
Declarations
.....
.....
.....

Truth_Table (.....)
.....
.....
.....
.....

End .....
    
```

**Exercice : comparateur 4 bits**

$$\begin{aligned}
 sup &= 1 \text{ si } A > B \\
 inf &= 1 \text{ si } A < B \\
 egal &= 1 \text{ si } A = B
 \end{aligned}$$



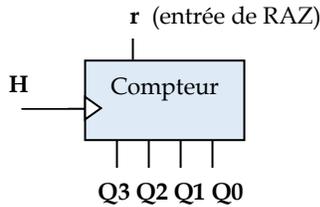
```

module compareur
Declarations
.....
.....
.....
.....

Equations
.....
.....
.....
.....

End .....
    
```

**Exercice : compteur 4 bits avec mise à 0 asynchrone**

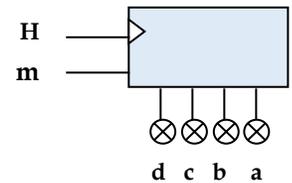


```

module .....
Declarations
.....
.....
.....
.....
equations
.....
.....
.....
End .....
    
```

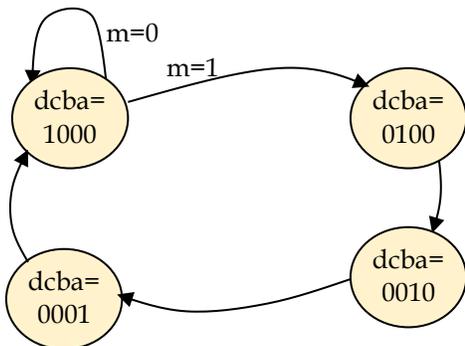
**Exercice : chenillard à 4 LED**

Au départ la Led **d** est allumée.  
L'appui sur le bouton **m** lance une séquence de défilements vers la droite.



Le diagramme d'état est une représentation symbolique de l'enchaînement séquentiel des états de sortie d'un système.

Le diagramme d'état suivant illustre le cycle d'allumage désiré :



```

Module chenillard
DECLARATIONS
    chenillard device 'P22V10' ;
    .....
    .....
    .....
EQUATIONS
    .....
STATE_DIAGRAM .....
    State [1,0,0,0] : IF (m==1) THEN goto [0,1,0,0] ; else goto [1,0,0,0] ;
    .....
    .....
    .....
END chenillard
    
```

## 6

## GRAFCET

## Rappel

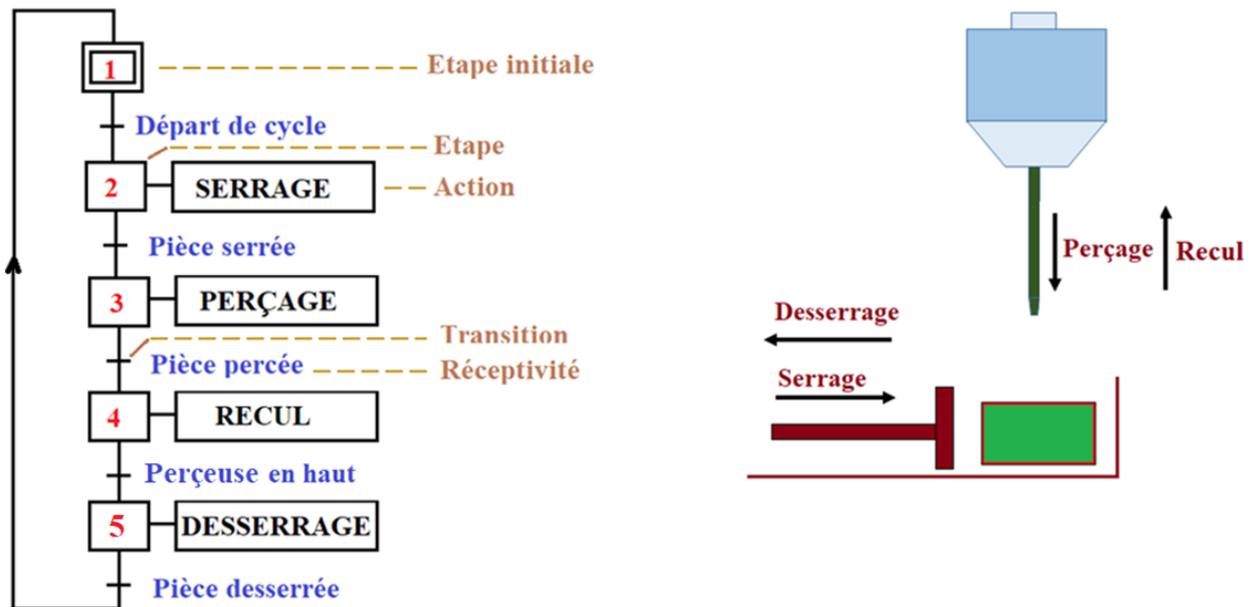
Le GRAFCET (Graphe Fonctionnel de Commande Étapes-Transitions) ou diagramme fonctionnel :

- ⇒ Est un outil de représentation du fonctionnement des systèmes automatisés séquentiels ;
- ⇒ Est un langage de programmation des automates programmables (API).

### 1. Concepts

Le GRAFCET décrit les cycles de fonctionnement par une succession d'étapes, de transitions et de liaisons orientées. Aux étapes, sont associées des actions et aux transitions, des réceptivités.

Exemple :



### Évolution du grafcet

- L'étape 1 est initialement active ;
- Lorsque la réceptivité "départ de cycle" est vraie, la transition correspondante "1-2" est franchie ce qui provoque l'activation de l'étape 2 qui alors provoque la désactivation de l'étape 1 ;
- Pendant que l'étape 2 est active, se déroule l'action correspondante "SERRAGE " et si la réceptivité "pièce serrée" devient vraie, l'étape 3 s'active et l'étape 2 se désactive et ainsi de suite.

### 2. Structures du GRAFCET

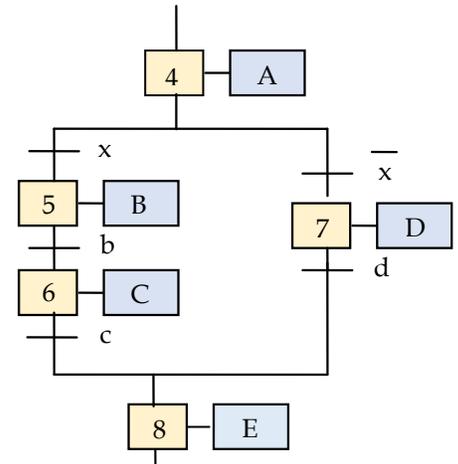
On distingue :

- GRAFCET à séquence unique (linéaire) ;
- GRAFCET à séquences multiples :
  - GRAFCET à sélection de séquences : séquences exclusives, saut d'étapes, reprise d'étapes ;
  - GRAFCET à séquences simultanées.

### Séquences exclusives (aiguillage en OU)

Permet d'effectuer un choix unique d'évolution entre plusieurs séquences d'étapes.

Ici, il y a sélection entre la séquence 4-5-6-8 et la séquence 4-7-8 selon l'état de la réceptivité  $x$ .



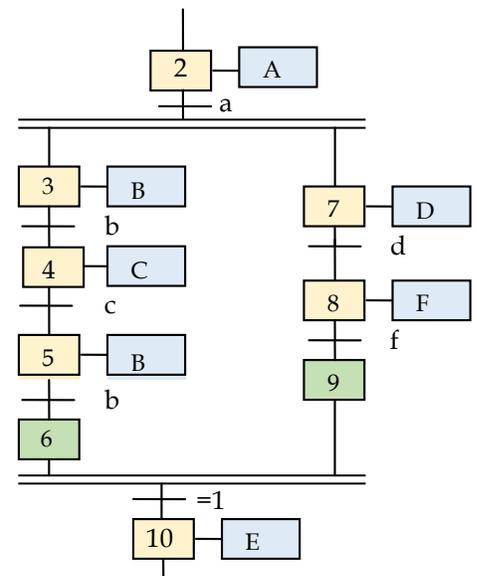
### Séquences simultanées (aiguillage en ET)

Le franchissement d'une transition conduit à activer simultanément plusieurs séquences d'étapes qui s'exécuteront parallèlement mais indépendamment.

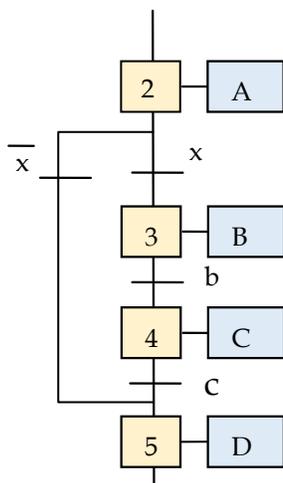
Ici, le franchissement de la transition "a" conduit à l'exécution simultanée des séquences 3-4-5-6 et 7-8-9.

Les séquences 3-4-5-6 et 7-8-9 évoluent indépendamment : par exemple, il n'est pas nécessaire que les étapes 4 et 8 s'activent en même temps.

Les étapes 6 et 9 sont des étapes d'attente. Une fois toutes les deux actives, l'étape 10 s'active immédiatement.



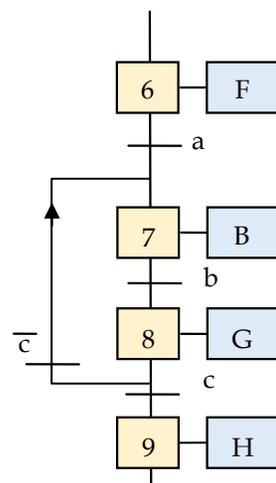
### Saut d'étapes



Le saut d'étapes est un saut conditionnel d'une séquence d'étapes.

Ici, il y a saut de la séquence 3-4 si  $x = 0$ .

### Reprise d'étapes



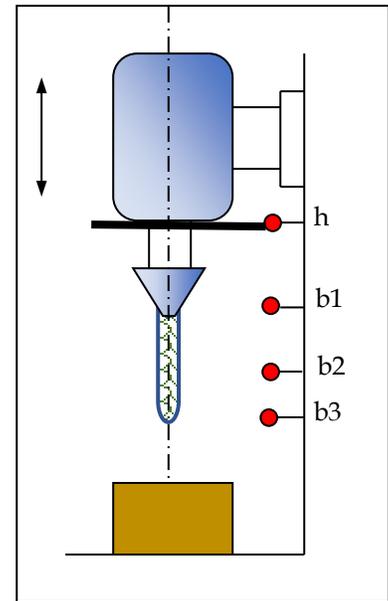
La reprise d'étapes permet de reprendre une ou plusieurs fois la même séquence.

Ici, la séquence 7-8 est répétée tant que  $c = 0$ .

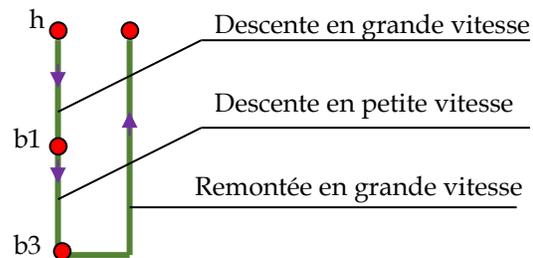
### Exercice : perceuse automatique

Suivant la nature des pièces, l'opérateur de la perceuse peut choisir entre deux modes : le perçage avec déburrage ou sans déburrage.

Établir le GRAFCET du point de vue partie commande.  
(La commande de la rotation de la broche n'est pas prise en compte)

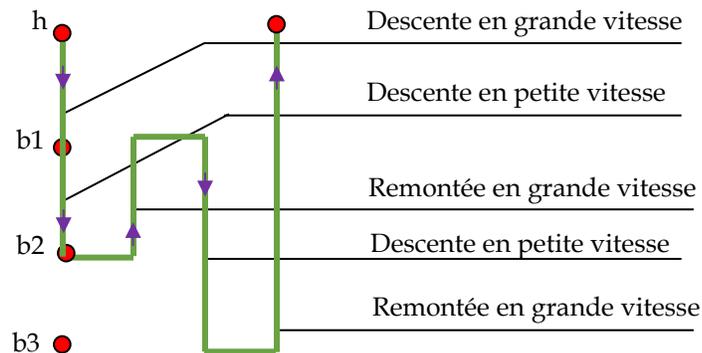


#### Cycle sans déburrage



#### Cycle avec déburrage

Pendant le déroulement du cycle, la broche effectue une remontée à une position intermédiaire (détectée par b2) afin de dégager le foret avant de terminer le perçage.

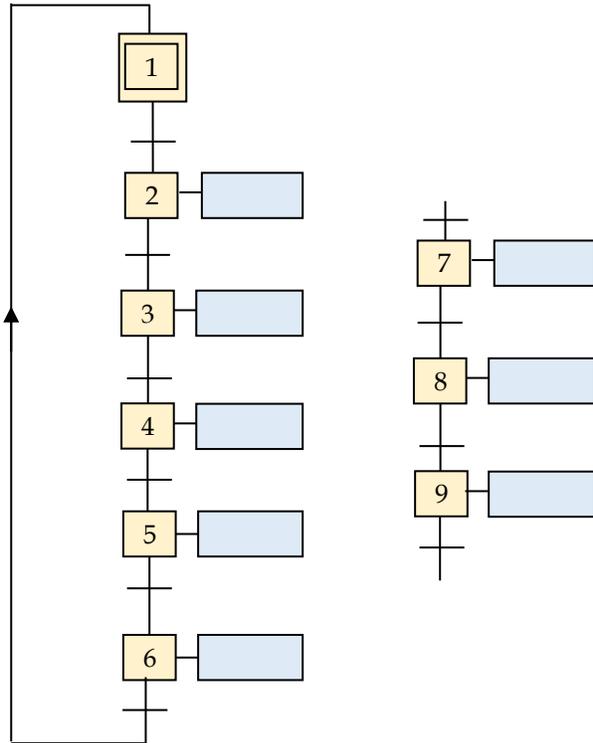
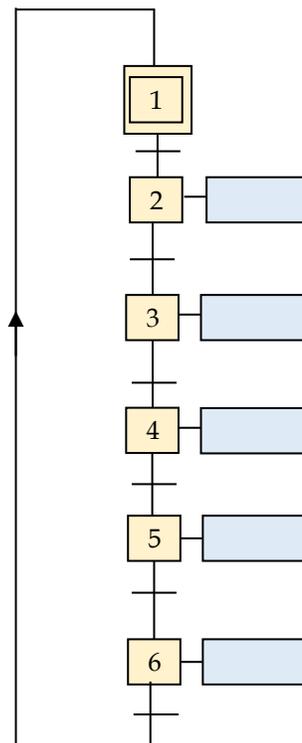


#### Actions

- KM1** : contacteur descente en grande vitesse
- KM2** : contacteur descente en petite vitesse
- KM3** : contacteur montée en grande vitesse

#### Capteurs et éléments du pupitre

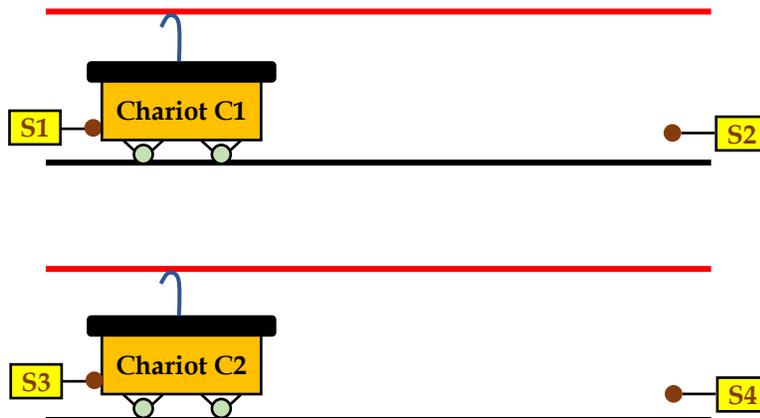
- h** : position haute
- b1** : approche terminée
- b2** : position intermédiaire
- b3** : perçage terminé
- br** : broche en rotation
- db $\bar{r}$ /dbr** : commutateur avec déburrage/sans déburrage
- dcy** : Bp départ de cycle

SolutionVariante**Exercice : Aller et retour de deux chariots**

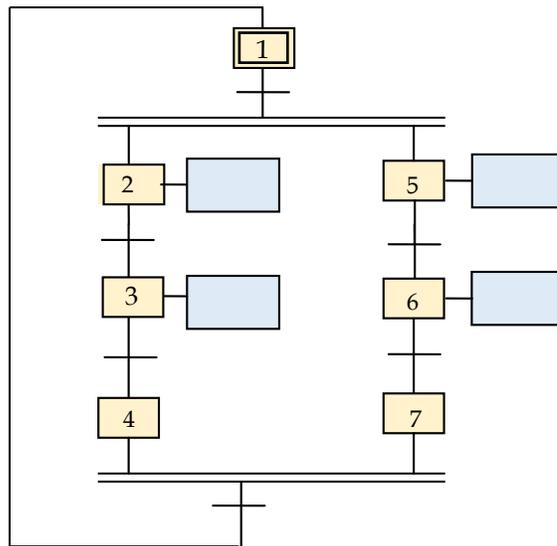
Suite à l'action sur le bouton **S0**, les deux chariots se déplacent pour effectuer chacun un aller et retour. Le cycle se termine lorsque les deux chariots se retrouvent à leurs positions initiales.

- Fournir le Grafcet de point de vue partie commande.

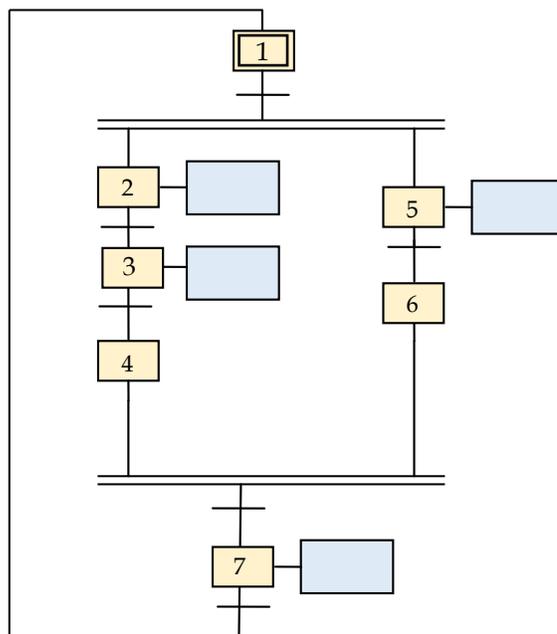
**S0**



| Mouvement                 | Actionneur | Préactionneur | Capteur                         |
|---------------------------|------------|---------------|---------------------------------|
| Déplacement avant de C1   | M1         | KM1           | S2 : fin de déplacement avant   |
| Déplacement arrière de C1 |            | KM2           | S1 : fin de déplacement arrière |
| Déplacement avant de C2   | M2         | KM3           | S4 : fin de déplacement avant   |
| Déplacement arrière de C2 |            | KM4           | S3 : fin de déplacement arrière |



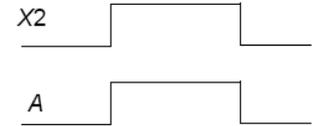
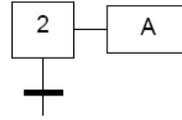
- Modifier le Grafcet si, maintenant, le retour du chariot C2 ne peut se faire qu'après le retour du chariot C1 à sa position initiale.



## Actions associées aux étapes

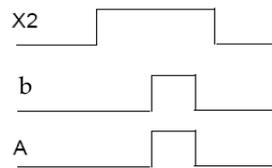
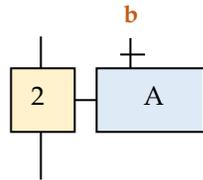
### ⇒ Action continue

L'action se poursuit tant que l'étape à laquelle elle est associée est active.



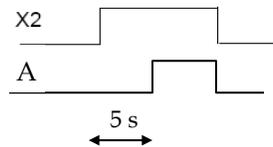
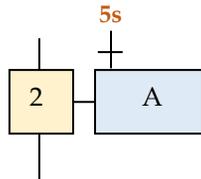
### ⇒ Action conditionnelle

Une action conditionnelle n'est exécutée que si l'étape est active et si la condition associée à cette étape est vraie.



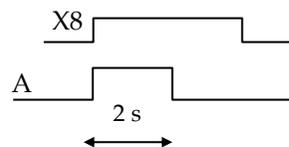
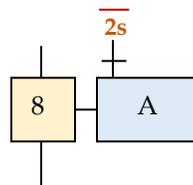
### ⇒ Action retardée

L'action est retardée par rapport à l'instant de l'activation de l'étape.



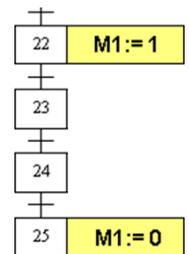
### ⇒ Action limitée

L'action limitée fonctionne pendant une durée depuis l'activation de l'étape.



### ⇒ Action mémorisée

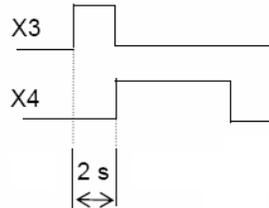
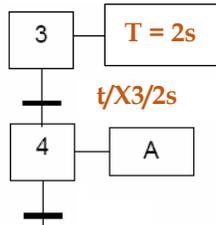
L'action se déroule sur plusieurs étapes. Il suffit d'indiquer les étapes de début et de fin de l'action. L'action M1 est active aux étapes 22, 23 et 24.



## Réceptivités particulières

### ⇒ Réceptivité fonction du temps

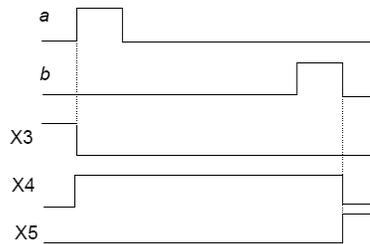
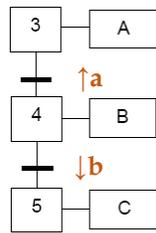
La réceptivité dépend d'une durée écoulée depuis l'activation d'une étape.



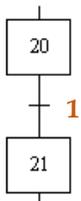
### ⇒ Réceptivité front

Il est fréquent que l'on ait à détecter dans un système le changement d'état d'une variable.

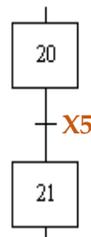
On représente ce changement d'état par :  $\uparrow a$ , pour le front montant de a et  $\downarrow a$  pour le front descendant de a.



### ⇒ Réceptivité toujours vraie

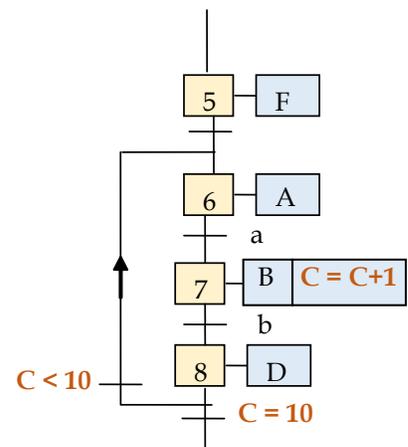


### ⇒ Variable associée à une étape



La variable X5, associée à l'étape 5, prend 1 si l'étape 5 est active et 0 dans le cas inverse.

### ⇒ Comptage



Ici, la séquence 6-7-8 est répétée 10 fois.

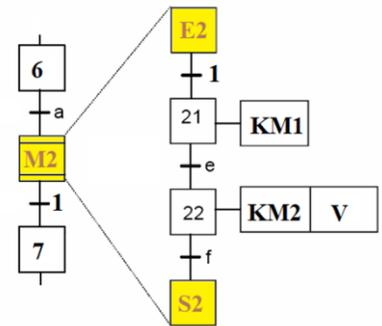
### Structuration du GRAFCET

⇒ Macro étape

Une macro-étape est la représentation unique d'une portion du GRAFCET appelée **expansion**. Elle se substitue à une étape du GRAFCET.

L'expansion de macro-étape comporte une étape d'entrée notée E et une étape de sortie notée S.

L'utilisation des macro-étapes permet de simplifier l'écriture du GRAFCET.

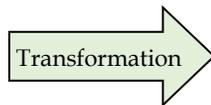
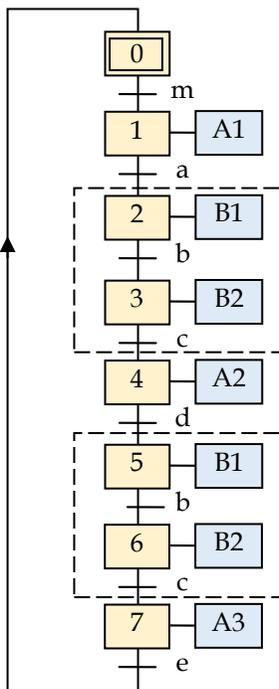


⇒ Synchronisation par tâche

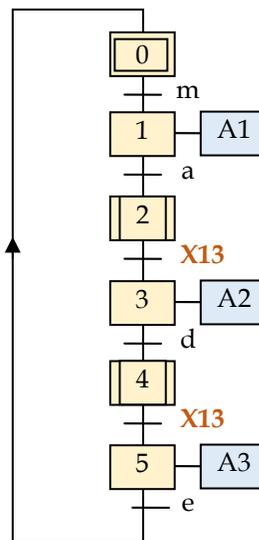
Dans le but de simplifier l'écriture de certains GRAFCET ou lorsqu'une suite de séquences apparaît plusieurs fois dans un GRAFCET, on est amené à définir des **tâches** que l'on peut appeler à partir d'un GRAFCET principal dit **GRAFCET de coordination** des tâches ou **Grafcet principal**.

Dans l'exemple suivant, on va déléguer les séquences 2-3 et 5-6 à une tâche évitant ainsi la répétition.

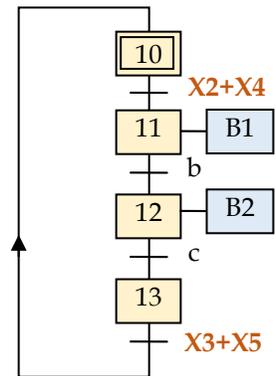
GRAFCET initial



GRAFCET de coordination



GRAFCET de tâche



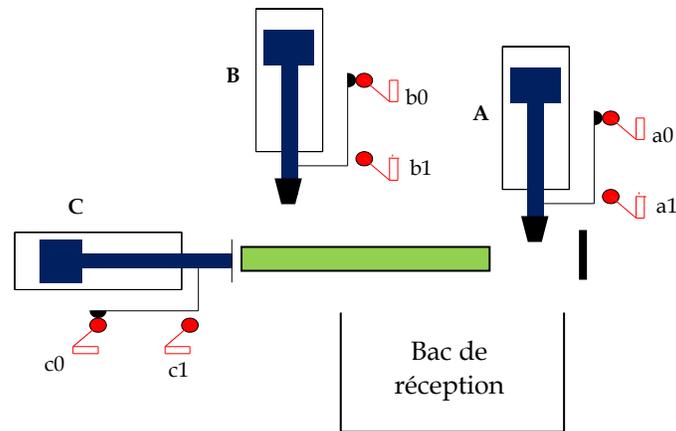
**X2+X4** : information de début de la tâche

**X13** : information de fin de la tâche

**X3+X5** : information de réinitialisation de la tâche

### Exercice : cintreuse de tubes

Le système permet le cintrage de tubes d'aluminium. Ces tubes sont ensuite intégrés à un système d'arrosage rotatif de jardin. C'est grâce au cintrage de ces tubes que la rotation de l'arroseur est effectuée.



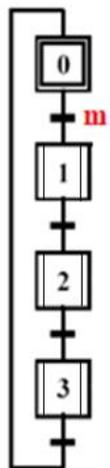
Dès l'appui sur un bouton poussoir **m**, le vérin C agit pour mettre le tube en position de cintrage. Le vérin A descend alors pour effectuer l'opération de cintrage. Le tube traité est, ensuite, évacué dans le bac de réception grâce au vérin B.

L'objectif est d'établir un Grafcet décomposé en 3 tâches :

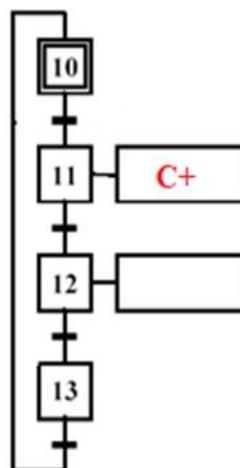
- Positionner le tube en butée ;
- Cintrer le tube ;
- Évacuer le tube dans le bac de réception.

Compléter cette solution.

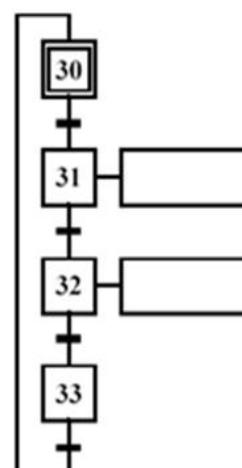
#### GRAF CET de coordination des tâches



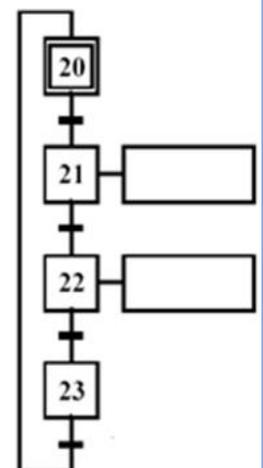
#### Tâche POSITIONNER



#### Tâche EVACUER



#### Tâche CINTRER



### Implantation du GRAFCET dans un automate

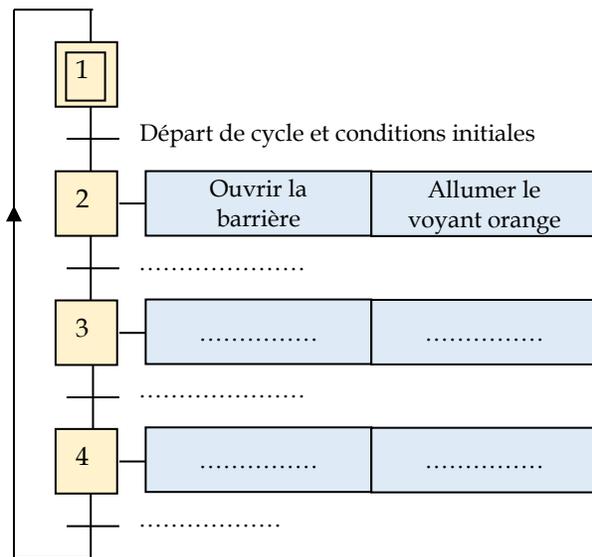
On considère le support technique "barrière automatique" qu'on désire commander selon les séquences suivantes :

- Le système étant au repos, si on appuie sur la télécommande **U2**, la barrière s'ouvre (**KA4**) en même temps que le voyant orange **H4** s'allume ;
- Une fois la barrière ouverte (**S1**), le voyant vert **H1** s'allume ;
- Après 8 s, la barrière se referme (**KA5**) et le voyant orange **H4** s'allume ;
- Lorsque la barrière est fermée (**S2**), le cycle se termine.

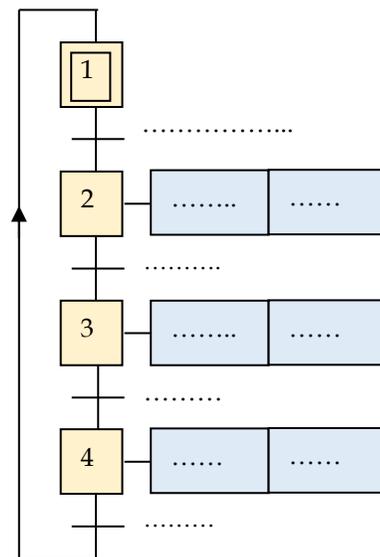
⇒ Étude du GRAFCET

- Compléter le GRAFCET d'un point de vue système et celui d'un point de vue partie commande.

GRAFCET de point de vue système



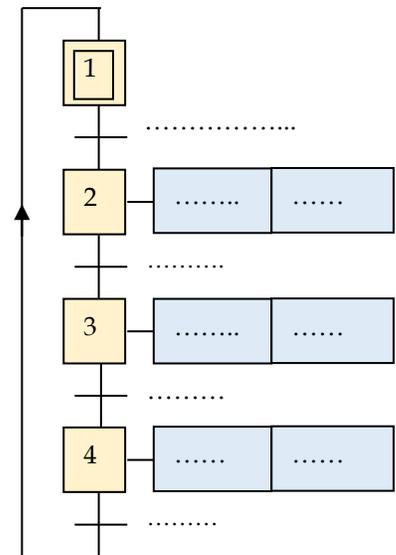
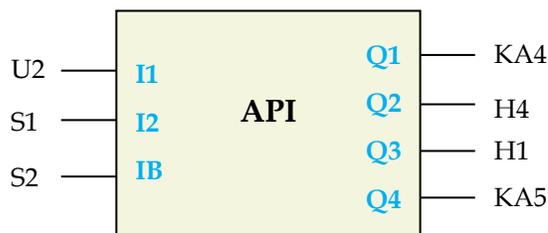
GRAFCET de point de vue partie commande



⇒ GRAFCET de point de vue automate

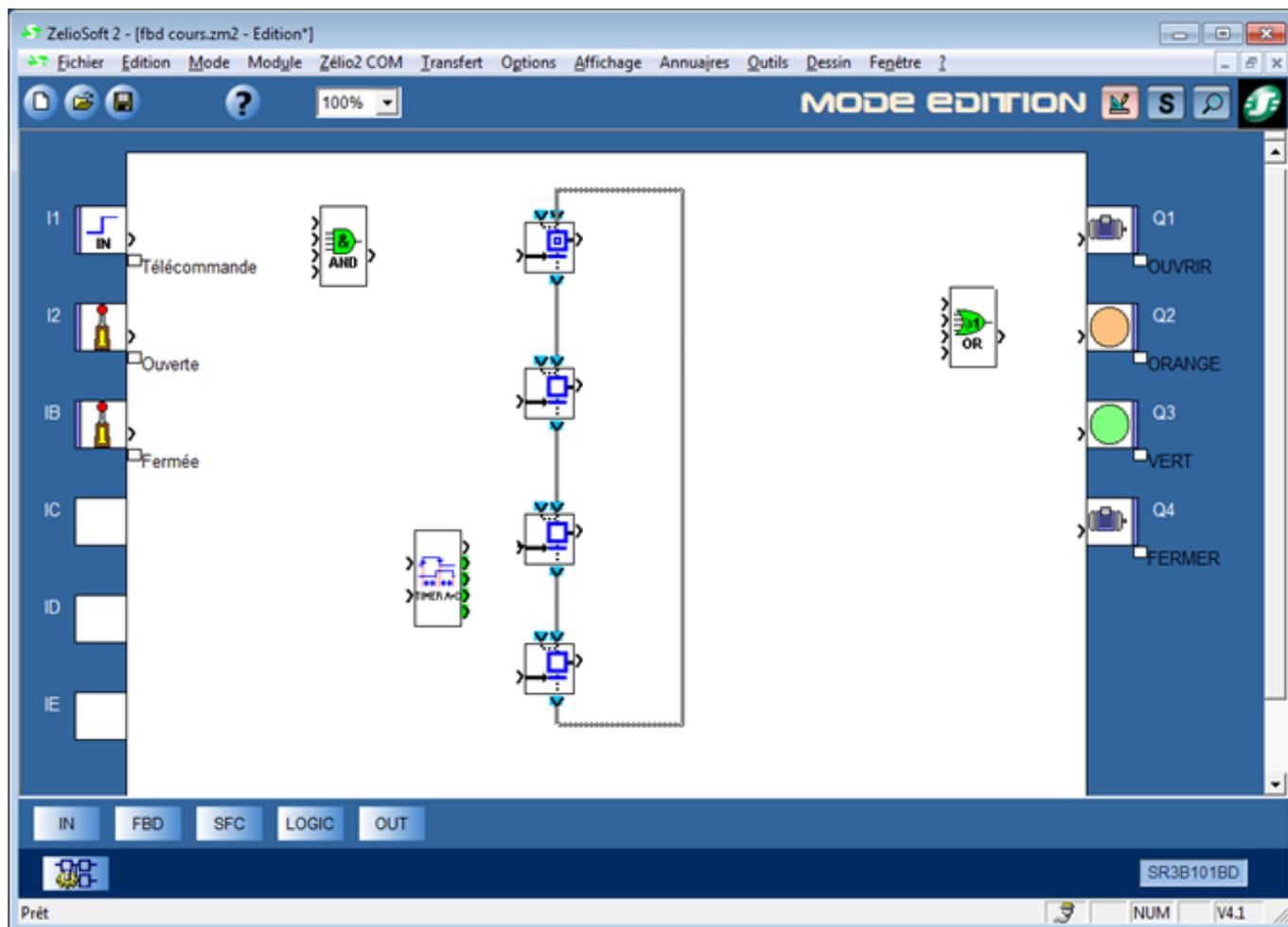
L'API équipant la barrière automatique est le module SR3B101BD doté de 6 entrées (I1, I2, IB, IC, ID, IE) et de 4 sorties (Q1, Q2, Q3, Q4).

- Le schéma bloc suivant, propose une affectation des entrées / sorties à l'automate, compléter le GRAFCET du point de vue API.



⇒ Programmation de l'API par le langage FBD

- Ci-après le schéma FBD à implanter dans l'automate via le logiciel ZelioSoft ; compléter-le



⇒ Programmation de l'API par le langage LADDER

- Effectuer la mise en équations du GRAFCET.

| Etape | Activation  | Désactivation |
|-------|-------------|---------------|
| 1     | SM1 = ..... | RM1 = .....   |
| 2     | SM2 = ..... | RM2 = .....   |
| 3     | SM3 = ..... | RM3 = .....   |
| 4     | SM4 = ..... | RM4 = .....   |

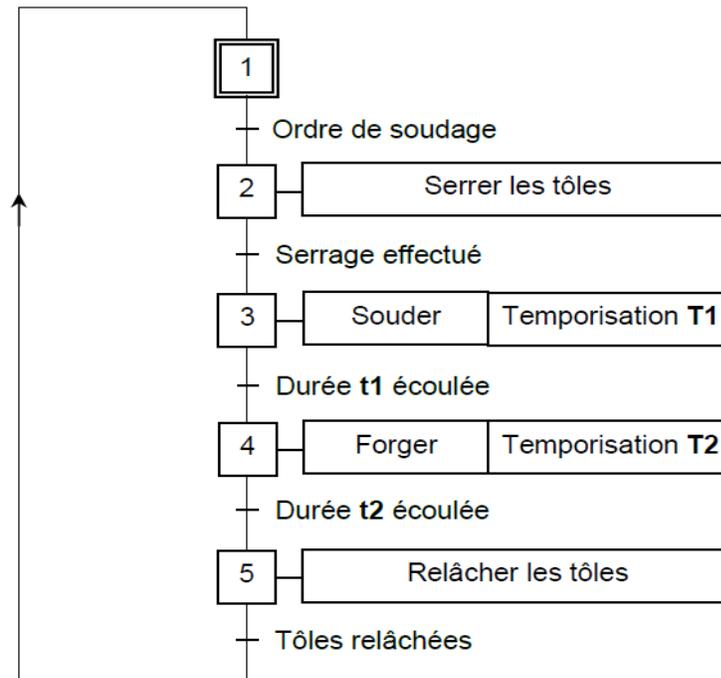
| Action     |
|------------|
| Q1 = ..... |
| Q2 = ..... |
| Q3 = ..... |
| Q4 = ..... |
| T1 = ..... |

- Compléter le programme LADDER

| Contact 1 | Contact 2 | Contact 3 | Contact 4 | Contact 5 | Bobine | Commentaire           |
|-----------|-----------|-----------|-----------|-----------|--------|-----------------------|
| M4        | IB        |           |           |           | SM1    | Activation étape 1    |
| M2        |           |           |           |           | RM1    | Désactivation étape 1 |
| .....     | .....     | .....     | .....     |           | SM2    | Activation étape 2    |
| .....     |           |           |           |           | RM2    | Désactivation étape 2 |
| .....     | .....     |           |           |           | SM3    | Activation étape 3    |
| .....     |           |           |           |           | RM3    | Désactivation étape 3 |
| .....     | .....     |           |           |           | SM4    | Activation étape 4    |
| .....     |           |           |           |           | RM4    | Désactivation étape 4 |
| .....     |           |           |           |           | Q1     | OUVRIR                |
| .....     |           |           |           |           | Q2     | ORANGE                |
| .....     |           |           |           |           | Q3     | VERT                  |
| .....     |           |           |           |           | Q4     | FERMER                |
| .....     |           |           |           |           | TT1    | TEMPORISATEUR         |

**Exercice 1 : examen national 2019, session normale**
**B. ETUDE DE LA COMMANDE DE LA TETE DE SOUDAGE**

Le cycle de soudage d'un point peut être décrit par le grafctet fonctionnel suivant.(on suppose que l'accostage est déjà effectué)



La commande de la tête de soudage se fait à l'aide d'un automate programmable (API) dont les affectations des entrées / sorties sont données par le tableau suivant :

| <i>Action / Information</i> | <i>Entrée/Sortie API</i> |
|-----------------------------|--------------------------|
| Serrage et forgeage         | <b>Q1</b>                |
| Relâchement                 | <b>Q2</b>                |
| Soudage                     | <b>Q3</b>                |
| Ordre de soudage            | <b>I1</b>                |
| Serrage effectué            | <b>I2</b>                |
| Relâchement effectué        | <b>I3</b>                |

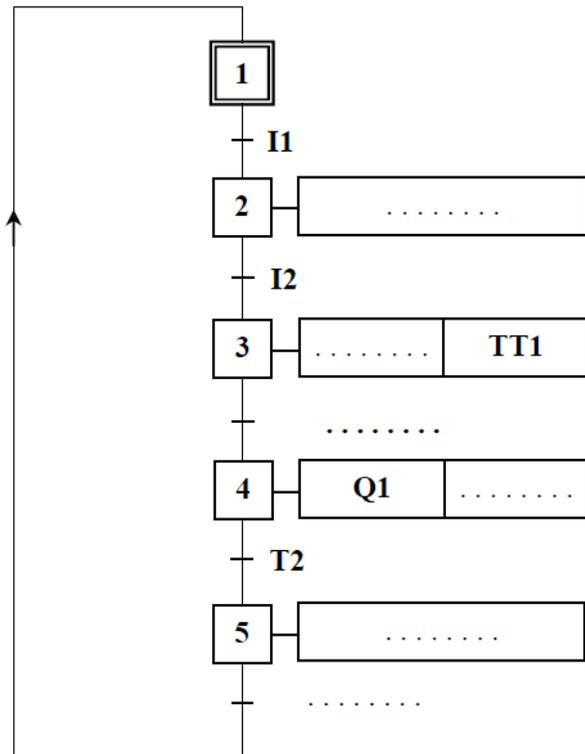
**Question : 42.** Compléter le grafctet de point de vue API.

[3 pts]

**Question : 43.** Compléter le programme, en langage **LADDER**, traduisant le grafctet point de vue API.

[3 pts]

**Question : 42.** [3 pts]

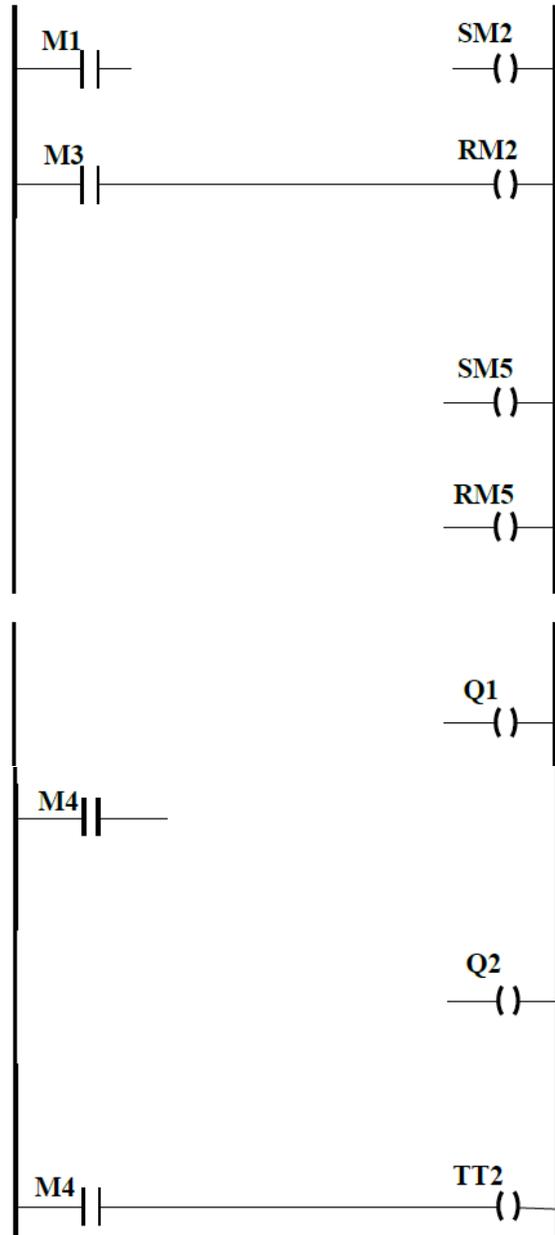


**Rappel :**

À chaque temporisateur sont associés 2 bobines (TTx et RTx) et un contact Tx :

- bobine **TTx** : entrée de déclenchement de la temporisation numéro x (x =1,2 ou 3...);
- bobine **RTx** : entrée de remise à zéro ;
- Contact **Tx** : sortie de fin de temporisation.

**Question : 43** [3 pts]



### Exercice 2 : examen national 2020, session normale

#### Tâche 2 : GRAFCET et programme LADDER

En se référant aux documents ressources **DRES 03** et **DRES 04** ; compléter :

**Q47-** Le GRAFCET du point de vue API.

**4,5 pts**

**Q48-** Le programme Ladder correspondant :

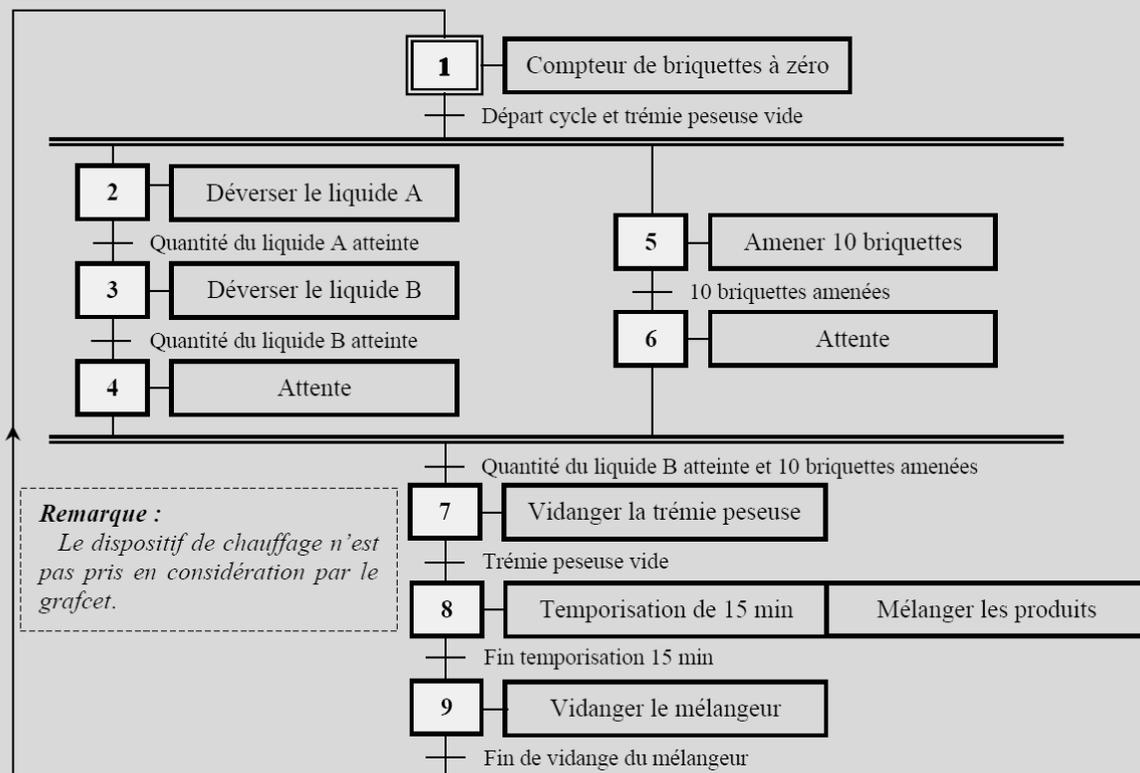
**5,5 pts**

- aux étapes 1, 2, 5 et 7.
- aux actions CCI, Q5 et Q6.

#### GRAFCET fonctionnel du système

**DRES 03**

Selon le cahier des charges les différentes fonctionnalités séquentielles du système sont décrites par le GRAFCET du point de vue système suivant :



#### Configuration matérielle de l'A.P.I

Identification et affectation des sorties

| Actions                               | Actionneurs                        | Préactionneurs | Sorties API |
|---------------------------------------|------------------------------------|----------------|-------------|
| Déverser le liquide A                 | Électrovanne V <sub>A</sub>        | Contacteur KVA | Q1          |
| Déverser le liquide B                 | Électrovanne V <sub>B</sub>        | Contacteur KVB | Q2          |
| Vidanger la trémie peseuse            | Électrovanne V <sub>C</sub>        | Contacteur KVC | Q3          |
| Amener une briquette                  | Moteur MR2                         | Contacteur KMI | Q4          |
| Mélanger les produits                 | Moteur MR1                         | Contacteur KM2 | Q5          |
| Vidanger le mélangeur                 | Électrovanne V <sub>D</sub>        | Contacteur KVD | Q6          |
| Compter le nombre de briquettes       | Compteur interne n°1 de l'API      |                | CC1         |
| Remettre à zéro le compteur           |                                    |                | RC1         |
| Définir le temps de malaxage (15 min) | Temporisateur interne n°1 de l'API |                | TT1         |

## Identification et affectation des entrées

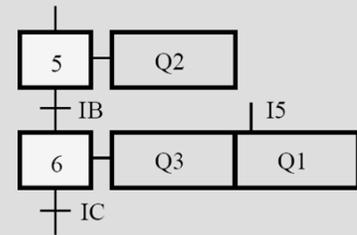
| Consignes/Comptes rendus                    | Boutons/Capteurs                           | Entrées API          |
|---|--|----------------------|
| Initialisation                              | Bouton poussoir <b>Init</b>                | <b>I1</b>            |
| Départ cycle                                | Bouton poussoir <b>Dcy</b>                 | <b>I2</b>            |
| Fin de vidange du mélangeur                 | Bouton poussoir <b>Fv</b>                  | <b>I3</b>            |
| Quantité du liquide A atteinte              | Capteur de force +<br>Conditionneur        | <b>P<sub>A</sub></b> |
| Quantité du liquide B atteinte              |  | <b>P<sub>B</sub></b> |
| Trémie peseuse vide                         |  | <b>P<sub>0</sub></b> |
| Briquette amenée                            | Détecteur optique <b>Pi</b>                | <b>I7</b>            |
| Nombre de briquettes dans le mélangeur = 10 | Contact interne <b>C1</b> du compteur      |                      |
| Fin de la temporisation de 15 min           | Contact interne <b>T1</b> du temporisateur |                      |

## Langage LADDER

DRES 04

Action conditionnelle :

- L'action **Q1** est conditionnelle ;
- L'action **Q1** est effective lorsque l'étape **6** est active et la condition **I5** est vraie ;
- L'équation de Q1 est : **Q1 = M6.I5**

Bloc compteur :

L'A.P.I possède plusieurs blocs compteurs de C1 ... Cn.

Le bloc fonction compteur permet de compter ou décompter des impulsions jusqu'à une valeur de présélection.

À chaque compteur sont associées 3 bobines (**RCx**, **CCx** et **DTx**) et un contact **Cx** :

- Bobine **CCx** : entrée de comptage ou de décomptage de numéro x (x =1,2 ou 3... ) ;
- Bobine **DCx** : entrée du sens de comptage (le bloc décompte si cette entrée est activée) ;
- Bobine **RCx** : entrée de remise à zéro du compteur ;
- Contact **Cx** : sortie de fin de comptage.

Le compteur **1** est paramétré à **10** donc le contact associé **C1** se ferme lorsque le compteur atteint la valeur de présélection **10**.

Bloc temporisateur :

L'A.P.I possède plusieurs blocs temporisateurs de T1 ... Tm.

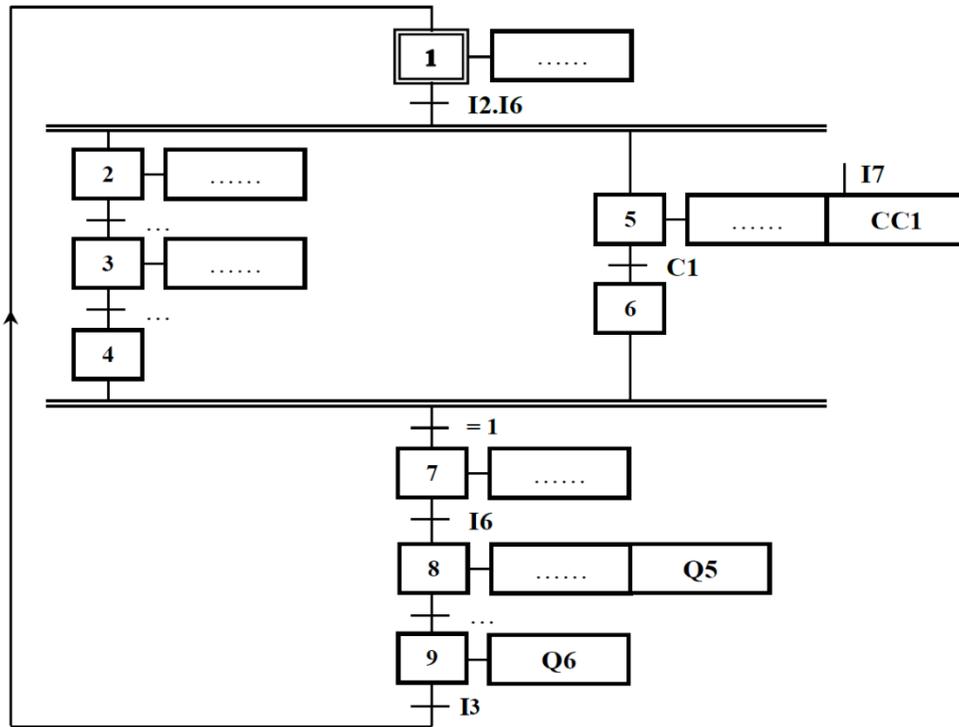
Le bloc fonction temporisateur permet de temporiser des actions.

À chaque temporisateur sont associées 2 bobines (**TTx** et **RTx**) et un contact **Tx** :

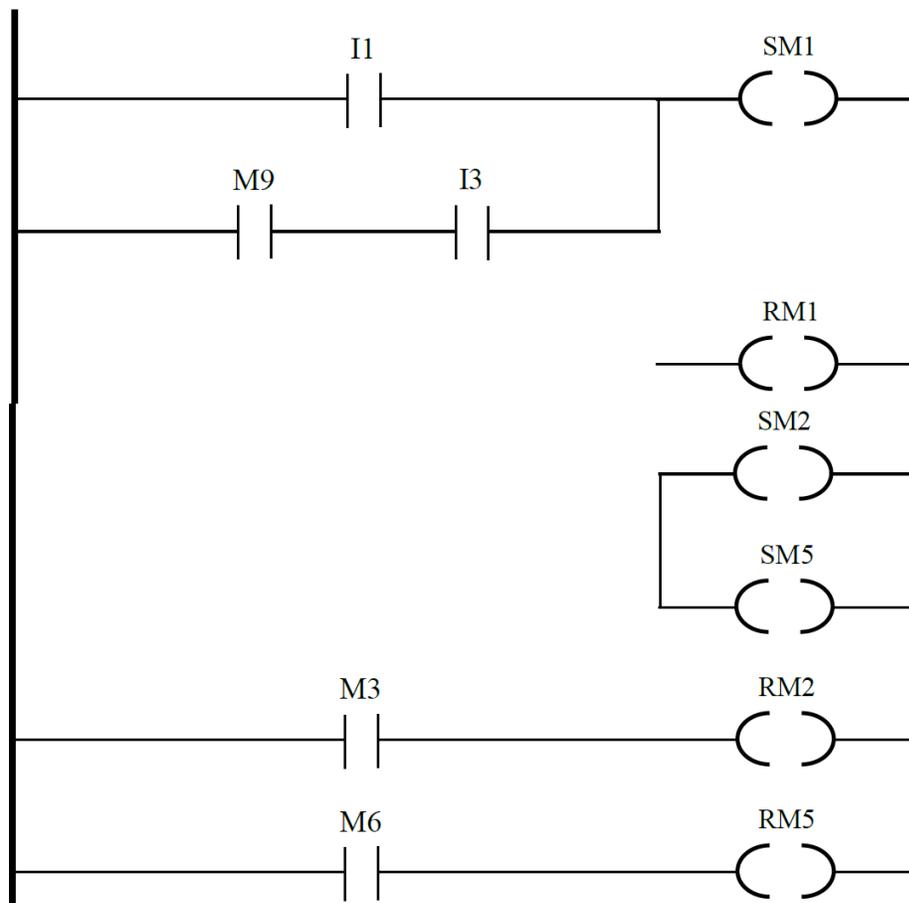
- Bobine **TTx** : entrée de lancement de la temporisation de numéro x (x =1,2 ou 3... ) ;
- Bobine **RTx** : entrée de remise à zéro ;
- Contact **Tx** : sortie de fin de temporisation.

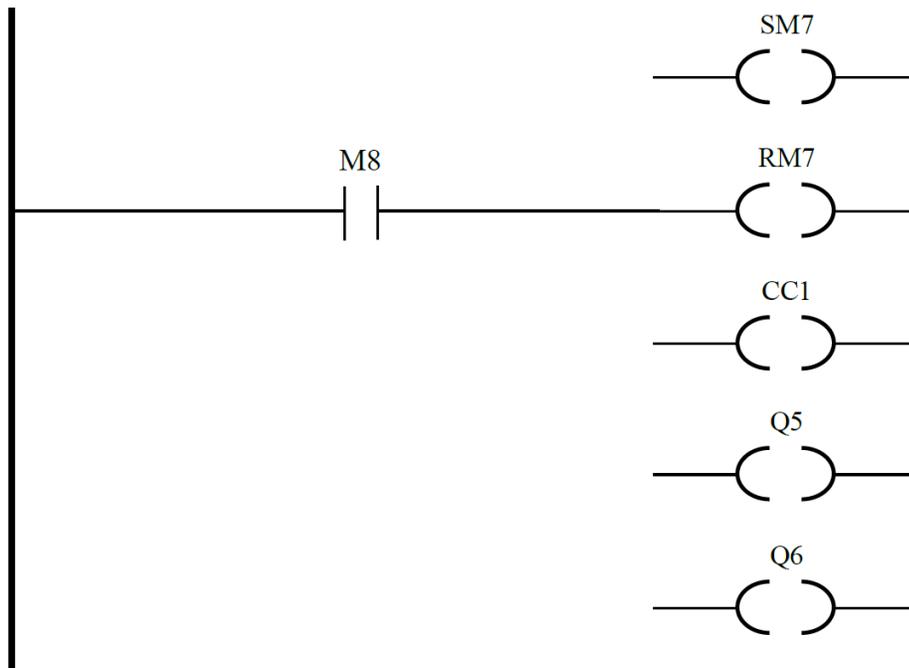
Le temporisateur **1** est paramétré à **15** min donc le contact associé **T1** se ferme **15** min après le déclenchement de la temporisation.

Q47- [4,5 pts ]



Q48- [5,5 pts]





### Exercice 3 : examen national 2021, session de rattrapage

Les différentes fonctionnalités séquentielles du système sont décrites par le GRAFCET global du point de vue fonctionnel donné au document ressources **DRES 03**.

Le GRAFCET de sécurité arrête la production (force le GRAFCET de production à son état initial) en cas d'éventuel défaut ou après un arrêt d'urgence.

#### **Tâche 1 : GRAFCET et programme LADDER**

En se référant aux documents ressources **DRES 03** et **DRES 04**, compléter :

**Q37-** Le GRAFCET du point de vue *API*.

**5 pts**

**Q38-** Le programme Ladder correspondant :

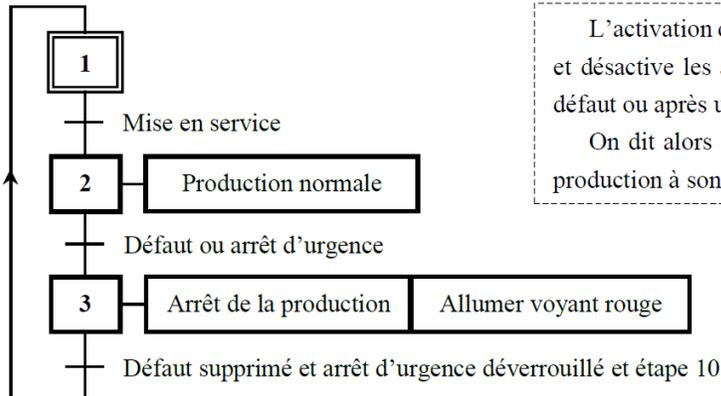
**6 pts**

- aux étapes **10,11** et **15** ;
- aux actions **TT4** et **Q6**.

## GRAFCET fonctionnel du système

**DRES 03**

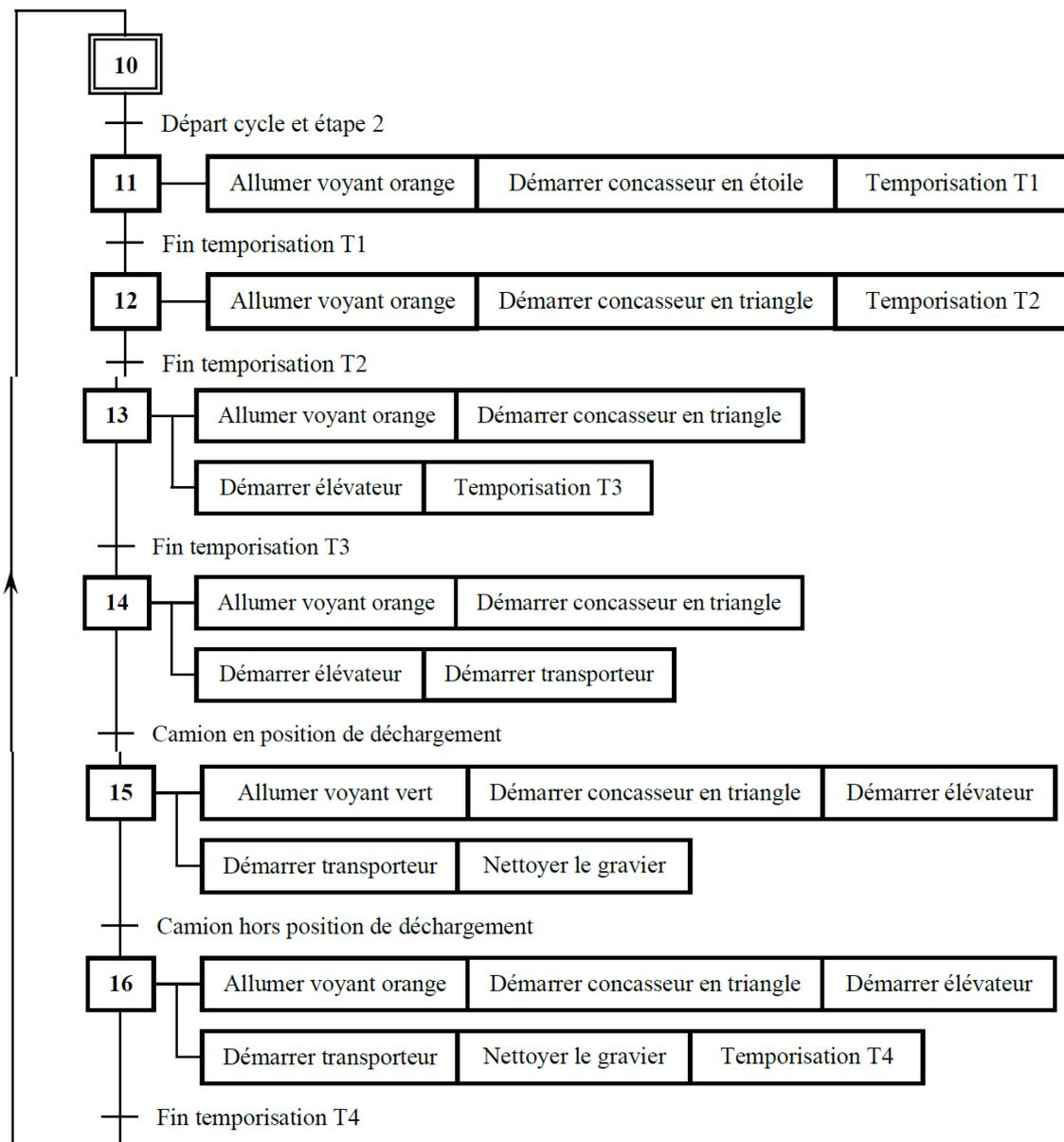
### GRAFCET de sécurité



L'activation de l'étape 3 du GRAFCET de sécurité active l'étape 10 et désactive les autres étapes du GRAFCET de production en cas de défaut ou après un arrêt d'urgence.

On dit alors que le GRAFCET de sécurité force le GRAFCET de production à son état initial lorsque l'étape 3 est active.

### GRAFCET de production



### Configuration matérielle de l'A.P.I et langage LADDER

**DRES 04**

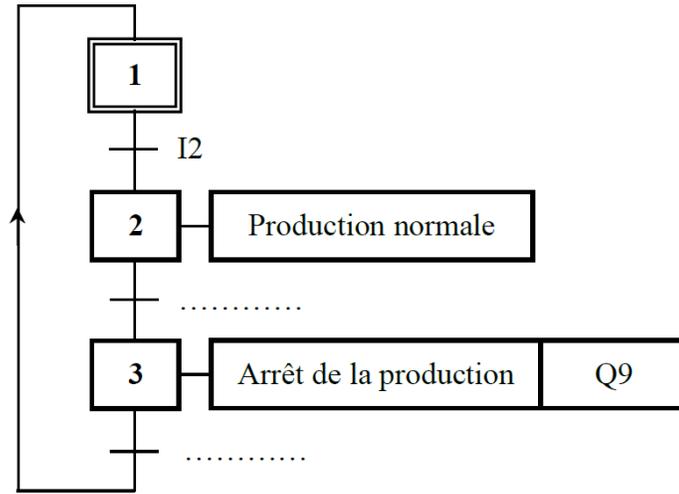
#### Identification et affectation des sorties

| Actions/Messages/Temporisations                | Actionneurs                        | Préactionneurs                 | Sorties API |
|--|------------------------------------|--------------------------------|-------------|
| Démarrer élévateur                             | Moteur <b>M1</b>                   | Contacteur de ligne <b>KM1</b> | <b>Q1</b>   |
| Démarrer transporteur                          | Moteur <b>M2</b>                   | Contacteur de ligne <b>KM2</b> | <b>Q2</b>   |
| Démarrer concasseur en étoile                  | Moteur <b>M3</b>                   | Contacteur de ligne <b>KM3</b> | <b>Q3</b>   |
|  |                                    | Contacteur étoile <b>KM4</b>   | <b>Q4</b>   |
| Démarrer concasseur en triangle                | Moteur <b>M3</b>                   | Contacteur de ligne <b>KM3</b> | <b>Q3</b>   |
|  |                                    | Contacteur triangle <b>KM5</b> | <b>Q5</b>   |
| Nettoyer le gravier                            | Électrovanne <b>EV</b>             |                                | <b>Q6</b>   |
| Absence du camion en position de déchargement  | Voyant orange <b>H1</b>            |                                | <b>Q7</b>   |
| Présence du camion en position de déchargement | Voyant vert <b>H2</b>              |                                | <b>Q8</b>   |
| Défaut ou un arrêt d'urgence                   | Voyant rouge <b>H3</b>             |                                | <b>Q9</b>   |
| Temporisation 5 s                              | Temporisateur interne n°1 de l'API |                                | <b>TT1</b>  |
| Temporisation 4 s                              | Temporisateur interne n°2 de l'API |                                | <b>TT2</b>  |
| Temporisation 3 s                              | Temporisateur interne n°3 de l'API |                                | <b>TT3</b>  |
| Temporisation 10 min                           | Temporisateur interne n°4 de l'API |                                | <b>TT4</b>  |

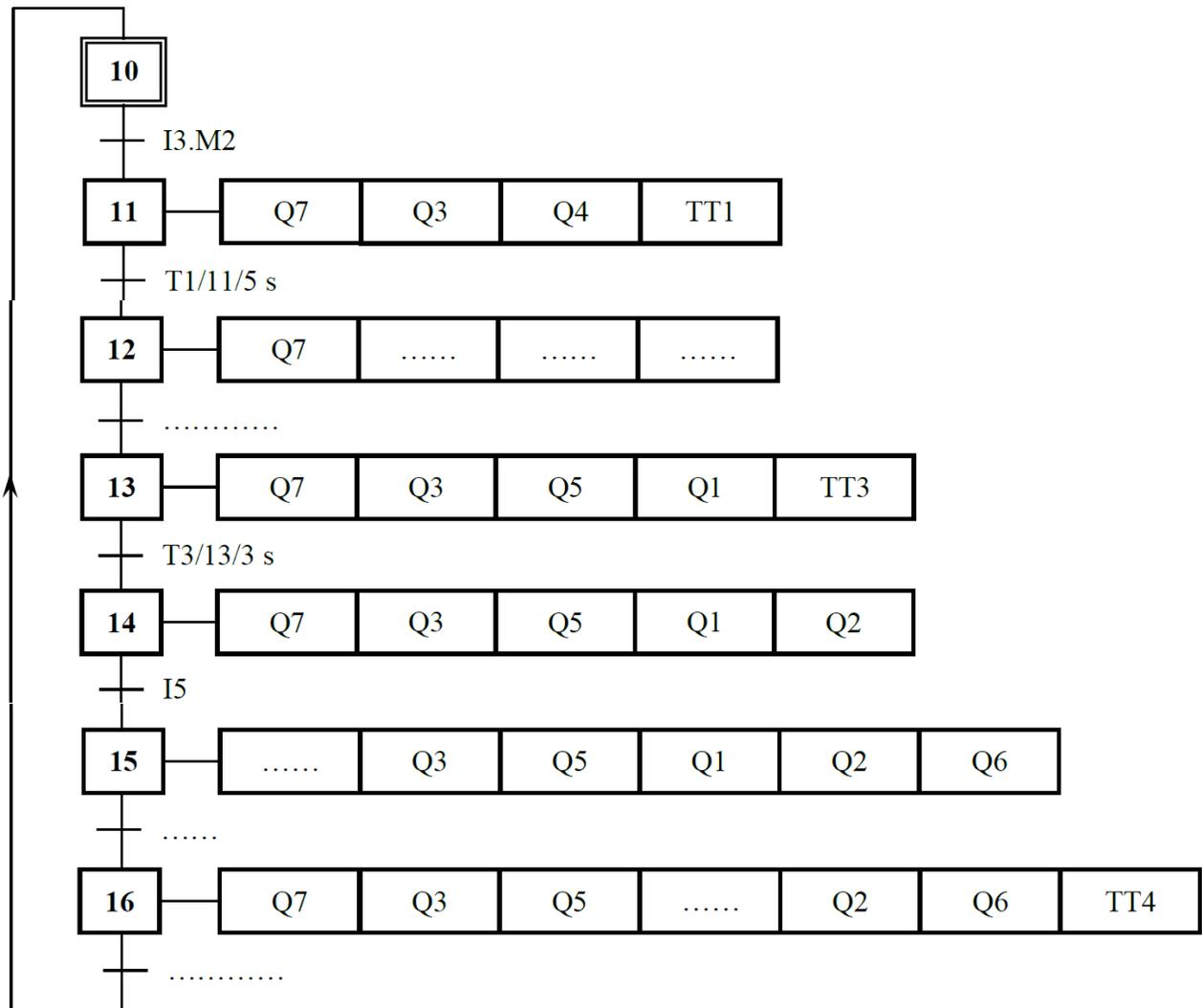
#### Identification et affectation des entrées

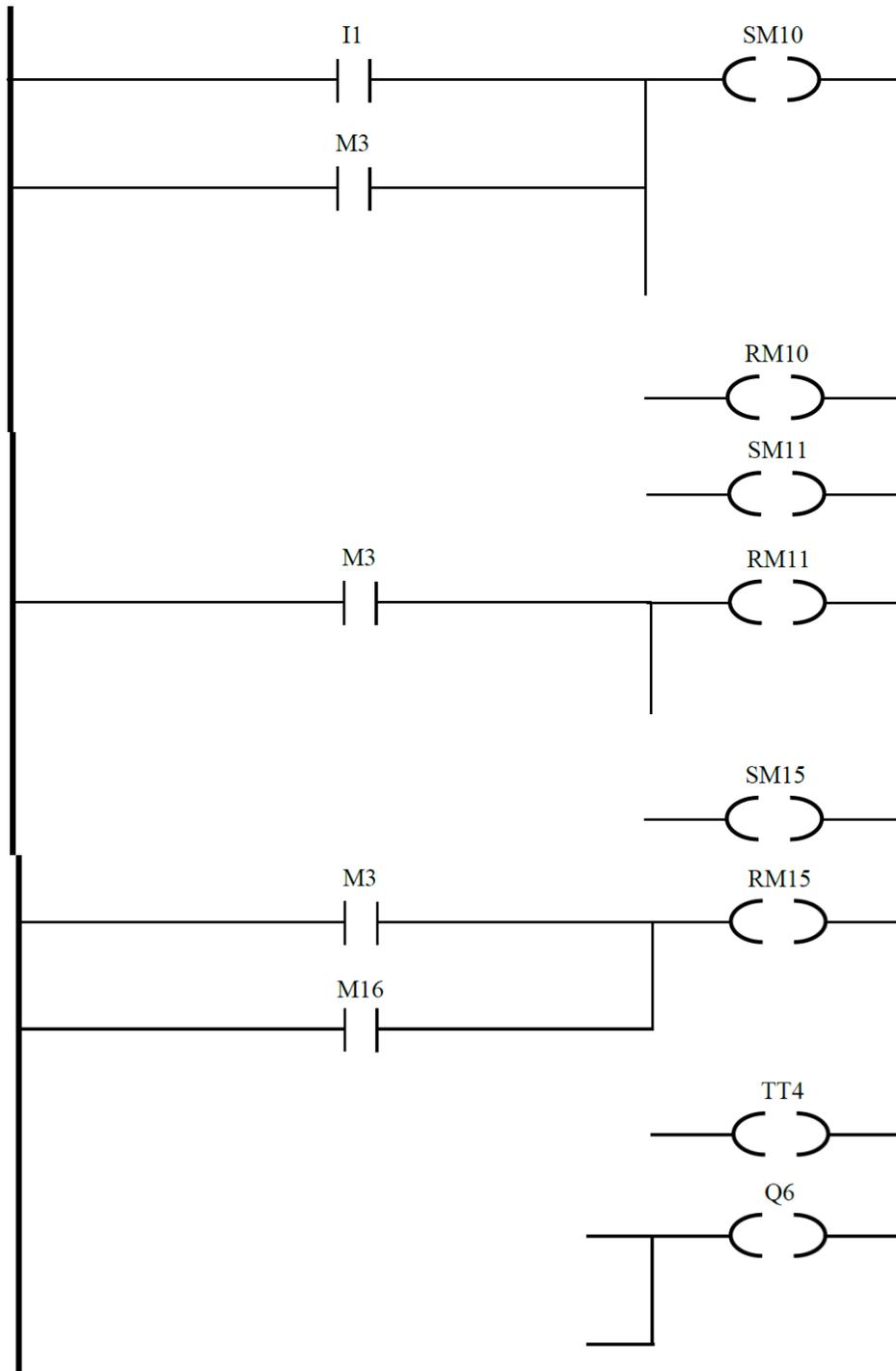
| Consignes/Comptes rendus           | Boutons/Capteurs                               | Entrées API |
|------------------------------------|--|-------------|
| Initialisation                     | Bouton poussoir <b>Init</b>                    | <b>I1</b>   |
| Mise en service du système         | Bouton poussoir <b>S1</b>                      | <b>I2</b>   |
| Départ cycle                       | Bouton poussoir <b>S2</b>                      | <b>I3</b>   |
| Arrêt d'urgence                    | Bouton poussoir <b>S3</b>                      | <b>I4</b>   |
| Camion en position de déchargement | Détecteur électromécanique <b>P</b>            | <b>I5</b>   |
| Présence d'un défaut               | Capteurs (couple, courant...)                  | <b>I6</b>   |
| Fin de la temporisation de 5 s     | Contact interne <b>T1</b> du temporisateur n°1 |             |
| Fin de la temporisation de 4 s     | Contact interne <b>T2</b> du temporisateur n°2 |             |
| Fin de la temporisation de 3 s     | Contact interne <b>T3</b> du temporisateur n°3 |             |
| Fin de la temporisation de 10 min  | Contact interne <b>T4</b> du temporisateur n°4 |             |

GRAFCET de sécurité



GRAFCET de production



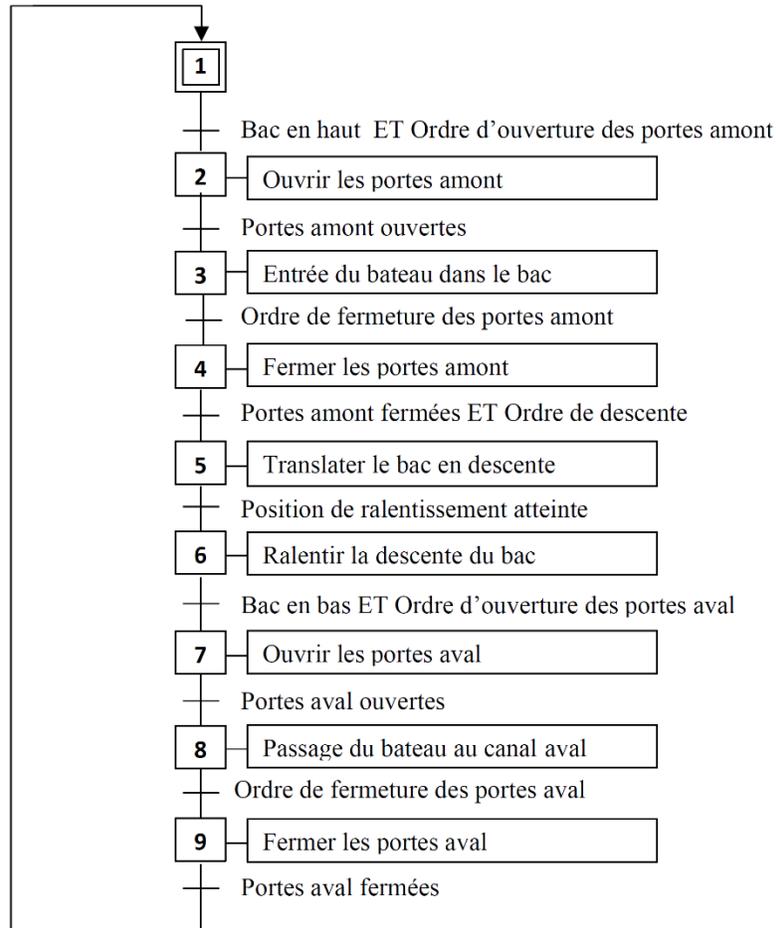


### Exercice 4 : examen national 2022, session de rattrapage

#### Tâche 2 : Cycle de manœuvre de l'ascenseur

Un opérateur, depuis son poste de commande et via un pupitre, contrôle le cycle de manœuvre, descente ou montée, du bac

Le Grafcet suivant décrit uniquement le cycle de la descente du bac :



**Q.47)** En se référant au Grafcet ci-dessus et au **DRES 04**, compléter le Grafcet de point de vue A.P.I. **3 pts**

Dans ce qui suit, on va déléguer la séquence d'étapes "4-5-6-7" à une tâche qu'on va appeler "Translation descendante".

**Q.48)** En partant du Grafcet de point de vue API (**Question 47**), compléter le Grafcet de la tâche "Translation descendante" ;

**3 pts**

**Q.49)** À partir du Grafcet modifié (**Question 48**), compléter le programme Ladder.

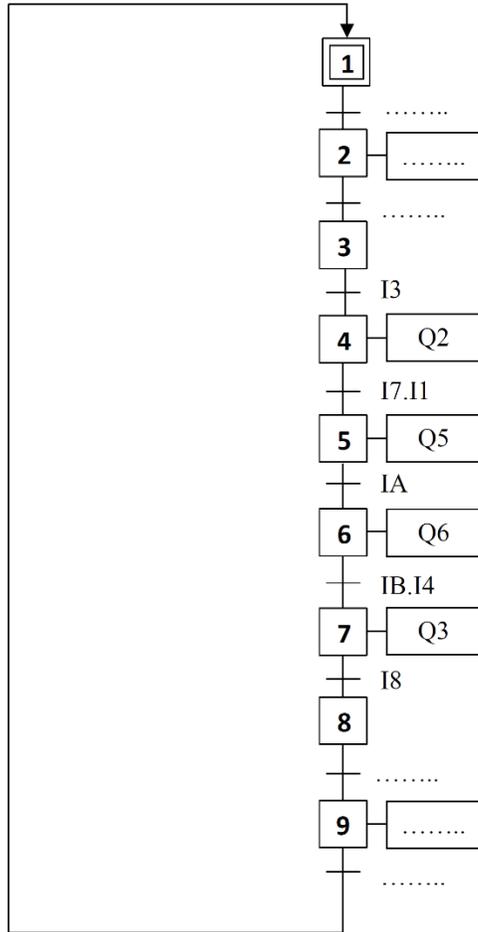
**3 pts**

**DRES 04****Tableaux d'affectation des entrées/sorties de l'automate**

| <b>Entrées</b>                      |   |                        |
|-------------------------------------|---|------------------------|
| <b>Information</b>                  | <b>Éléments de consigne et capteurs</b> | <b>Entrée automate</b> |
| Ordre de descente                   | <b>S1</b>                               | <b>I1</b>              |
| Ordre d'ouverture des portes amont  | <b>S2</b>                               | <b>I2</b>              |
| Ordre de fermeture des portes amont | <b>S3</b>                               | <b>I3</b>              |
| Ordre d'ouverture des portes aval   | <b>S4</b>                               | <b>I4</b>              |
| Ordre de fermeture des portes aval  | <b>S5</b>                               | <b>I5</b>              |
| Portes amont ouvertes               | <b>pmo</b>                              | <b>I6</b>              |
| Portes amont fermées                | <b>pmf</b>                              | <b>I7</b>              |
| Portes aval ouvertes                | <b>pvo</b>                              | <b>I8</b>              |
| Portes aval fermées                 | <b>pvf</b>                              | <b>I9</b>              |
| Position de ralentissement atteinte | <b>b</b>                                | <b>IA</b>              |
| Bac en bas                          | <b>a</b>                                | <b>IB</b>              |
| Bac en haut                         | <b>d</b>                                | <b>IC</b>              |
| Bouton d'initialisation             | <b>S6</b>                               | <b>ID</b>              |

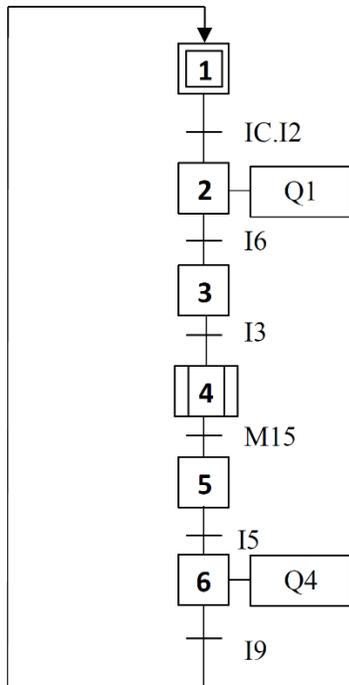
| <b>Sorties</b>                |                        |                        |
|-------------------------------|------------------------|------------------------|
| <b>Action</b>                 | <b>Pré-actionneurs</b> | <b>Sortie automate</b> |
| Ouvrir les portes amont       | <b>OPM</b>             | <b>Q1</b>              |
| Fermer les portes amont       | <b>FPM</b>             | <b>Q2</b>              |
| Ouvrir les portes aval        | <b>OPV</b>             | <b>Q3</b>              |
| Fermer les portes aval        | <b>FPV</b>             | <b>Q4</b>              |
| Translater le bac en descente | <b>DGV</b>             | <b>Q5</b>              |
| Ralentir la descente du bac   | <b>DPV</b>             | <b>Q6</b>              |

Q.47)

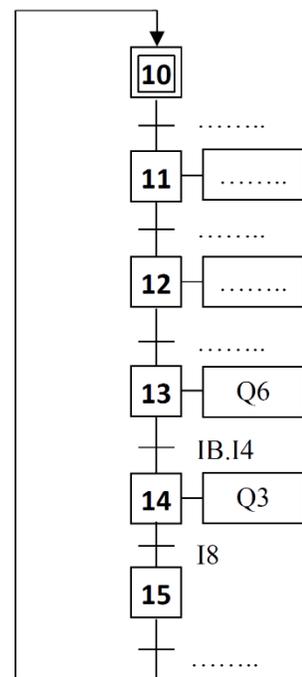


Q.48)

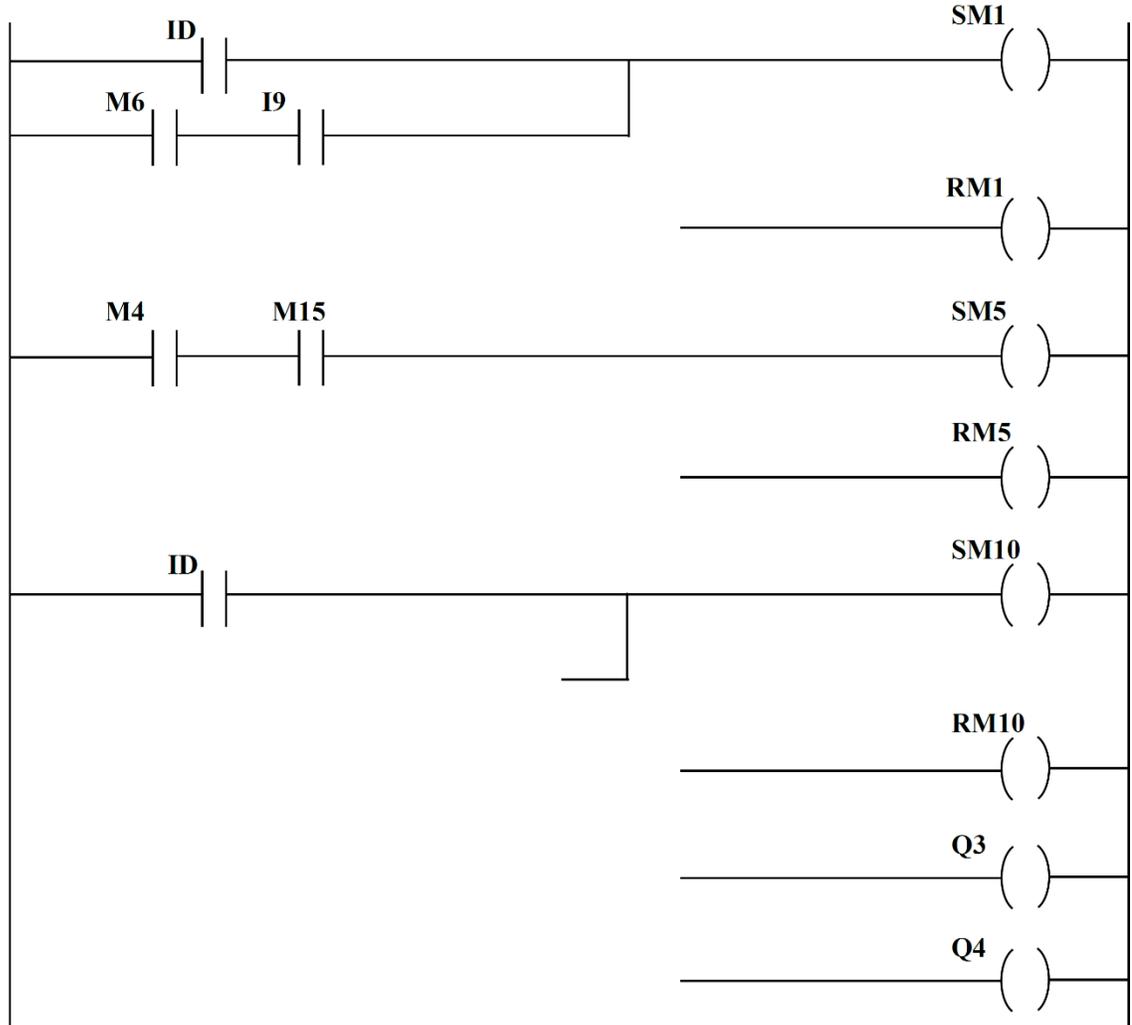
Grafcet principal



Tâche "Translation descendante"



Q.49)



### Exercice 5 : examen national 2023, session normale

#### Tâche 1 : GRAFCET de fonctionnement de l'escalier et programme LADDER

Les différentes fonctionnalités séquentielles du système sont décrites par le GRAFCET fonctionnel donné au document ressources **DRES 05**.

La commande de l'escalier mécanique est assurée par un automate programmable industriel (**A.P.I**) dont les affectations des entrées / sorties sont données par les tableaux du document ressources **DRES 05**.

**Q.29-** Compléter le GRAFCET du point de vue **A.P.I** traduisant le fonctionnement de l'escalier.

**3,5 pts**

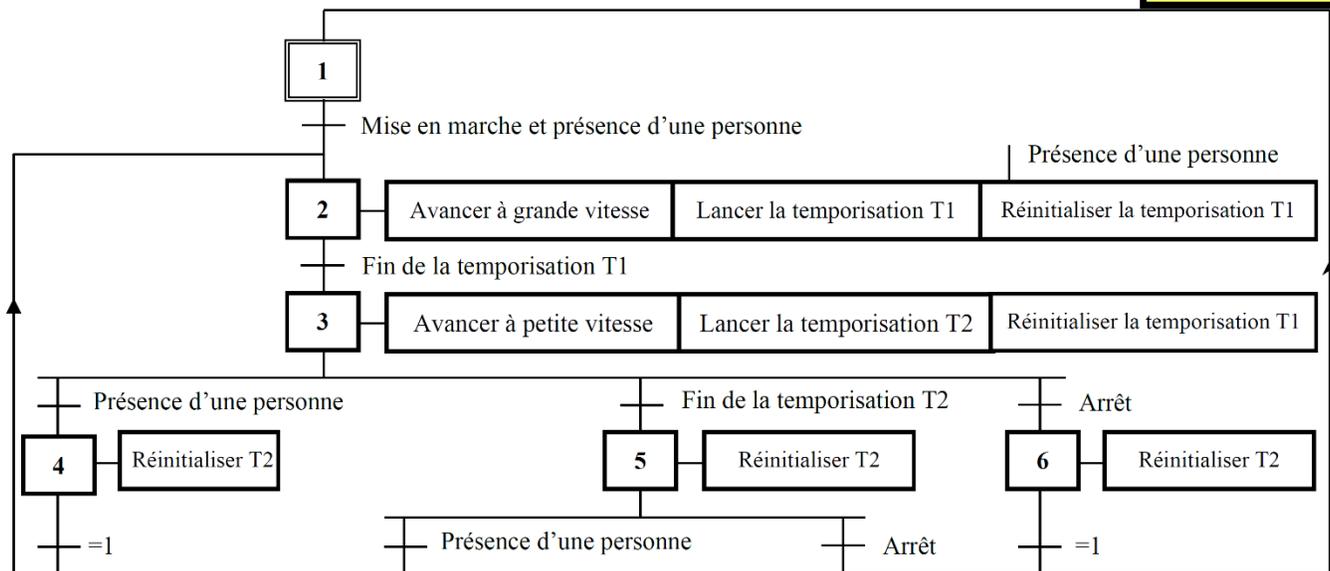
**Q.30-** Compléter le programme LADDER correspondant :

**4 pts**

- Aux étapes 2, 3 et 5.
- Aux actions **RT1** et **Q1**.

#### GRAFCET fonctionnel de l'escalator

**DRES 05**



L'escalier mécanique est prêt à fonctionner lorsqu'on bascule l'interrupteur bistable Marche/Arrêt sur la position Marche (l'étude sera limitée au sens de marche **avant**) :

- Lorsqu'une personne est détectée au niveau du palier d'entrée par le détecteur de présence, le moteur accélère jusqu'à la grande vitesse  $V_1 = 0,6 \text{ m/s}$  (mode normal) ;
- Si aucune personne ne se présente dans un délai  $T_1$ , le moteur décélère jusqu'à la petite vitesse  $V_2 = 0,28 \text{ m/s}$  (mode veille) ;
- Si aucune personne ne se présente dans un délai  $T_2$ , le moteur s'arrête ;
- Lorsqu'une personne se présente à nouveau au niveau du palier d'entrée, le moteur accélère jusqu'à la grande vitesse  $V_1 = 0,6 \text{ m/s}$ , qu'il soit à l'arrêt ou en mode veille ;
- La prise en compte d'un arrêt (en basculant l'interrupteur bistable Marche/Arrêt sur la position Arrêt) se fait durant le mode veille ou l'arrêt du moteur.

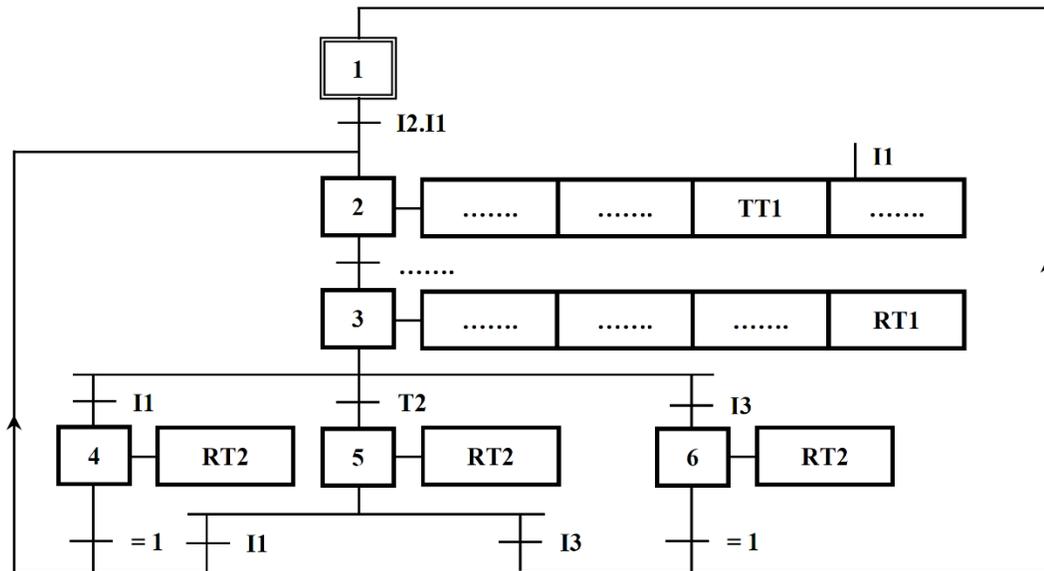
### Identification et affectation des sorties

| Actions                             | Actionneurs             | Préactionneurs                        | API |
|-------------------------------------|-------------------------|---------------------------------------|-----|
| Avancer l'escalier à grande vitesse | Moteur                  | Variateur de vitesse Altivar 31 (LI1) | Q1  |
|                                     |                         | Variateur de vitesse Altivar 31 (LI4) | Q4  |
| Avancer l'escalier à petite vitesse | Moteur                  | Variateur de vitesse Altivar 31 (LI1) | Q1  |
|                                     |                         | Variateur de vitesse Altivar 31 (LI3) | Q3  |
| Lancer la temporisation Tx          | Temporisateurs T1 et T2 |                                       | TTx |
| Réinitialiser la temporisation Tx   |                         |                                       | RTx |

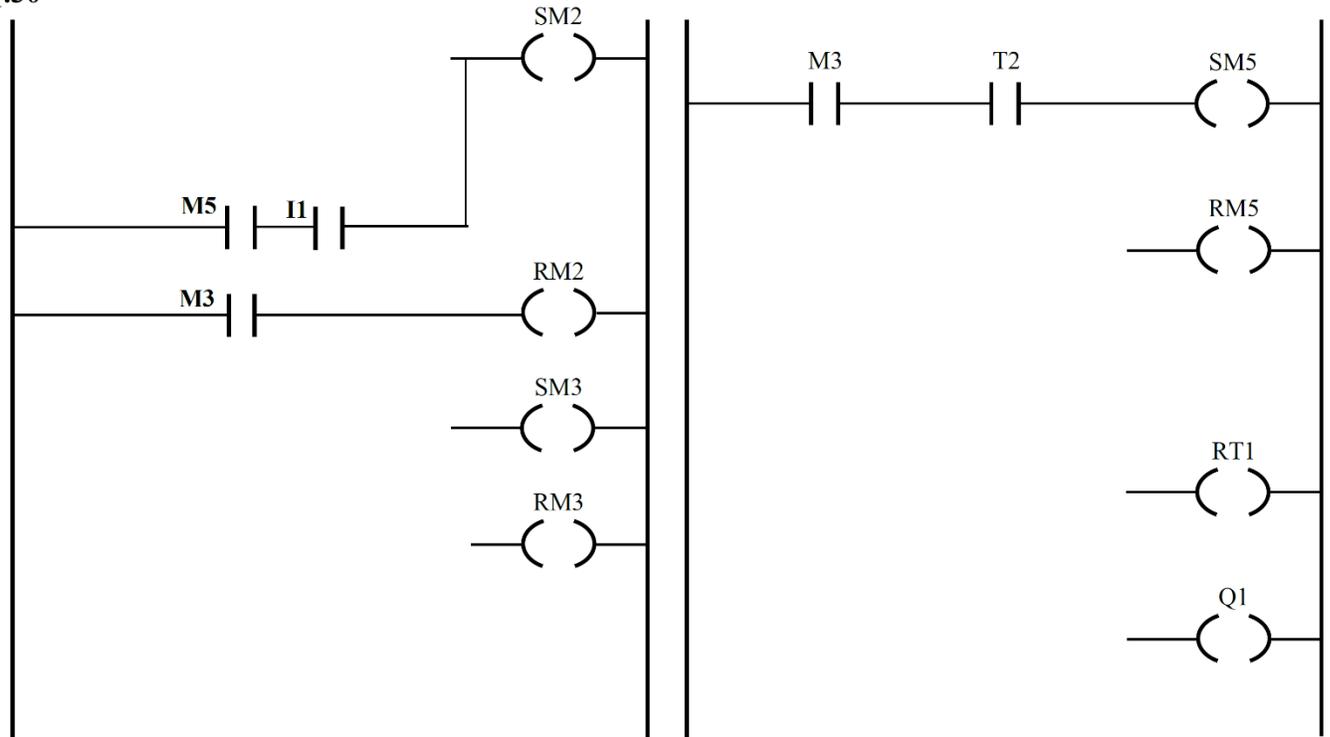
### Identification et affectation des entrées

| Compte-rendu et consigne | Capteur et bouton     | API |
|--------------------------|-----------------------|-----|
| Présence d'une personne  | Détecteur optique     | I1  |
| Marche                   | Interrupteur bistable | I2  |
| Arrêt                    |                       | I3  |
| Fin de temporisation Tx  | Temporisateurs 1 et 2 | Tx  |

Q.29 -



Q.30 -



**Exercice 6 : examen national 2024, session normale**

Le déplacement vertical de la charge (levage) est géré par un automate programmable industriel (**A.P.I.**).

Les tableaux donnés en **DRES 03** précisent les affectations des entrées et des sorties de l'**A.P.I.**

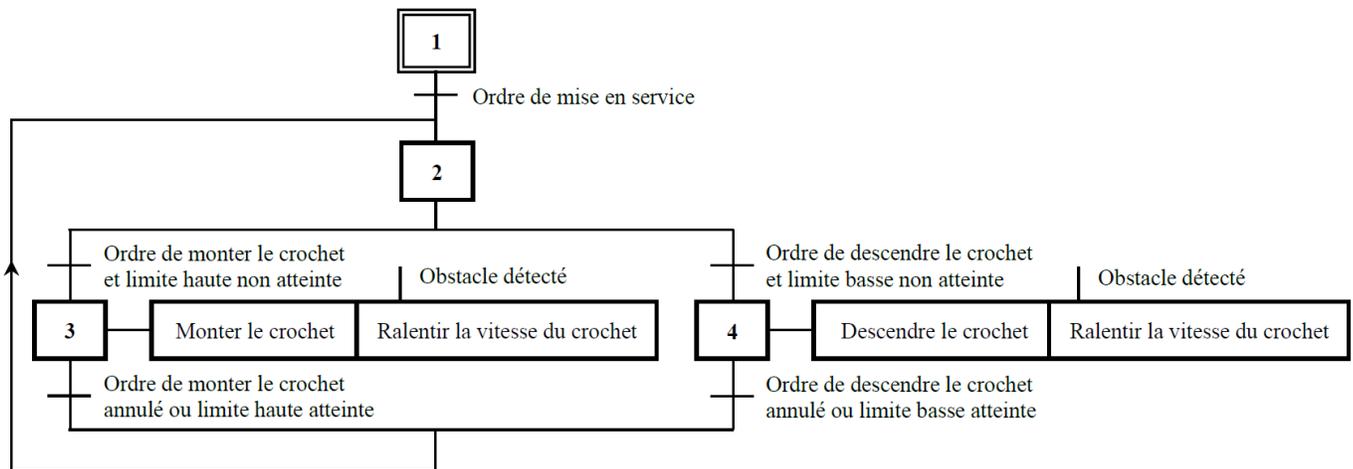
Le schéma synoptique de commande de déplacement vertical du crochet est donné en **DRES 03**.

On note que le moteur **M1** ralentit à la vitesse **V<sub>R</sub>** lorsque le capteur **P** détecte, à une distance **d** prédéfinie, la présence d'un obstacle.

**Tâche 1 : Étude du GRAFCET de déplacement vertical du crochet (levage)**

On se propose d'établir le programme ladder permettant la commande de déplacement vertical de la charge.

Le cycle de déplacement vertical de la charge suivant l'axe **Z** est décrit par le GRAFCET fonctionnel suivant :



En se référant aux tableaux d'affectations des entrées et des sorties de l'A.P.I. du document **DRES 03**, compléter :

**Q.38**-Le GRAFCET du point de vue **A.P.I.**

**4 pts**

**Q.39**-Le programme Ladder correspondant aux étapes 2, 3 et 5 et à l'action **Q3**.

**4 pts**

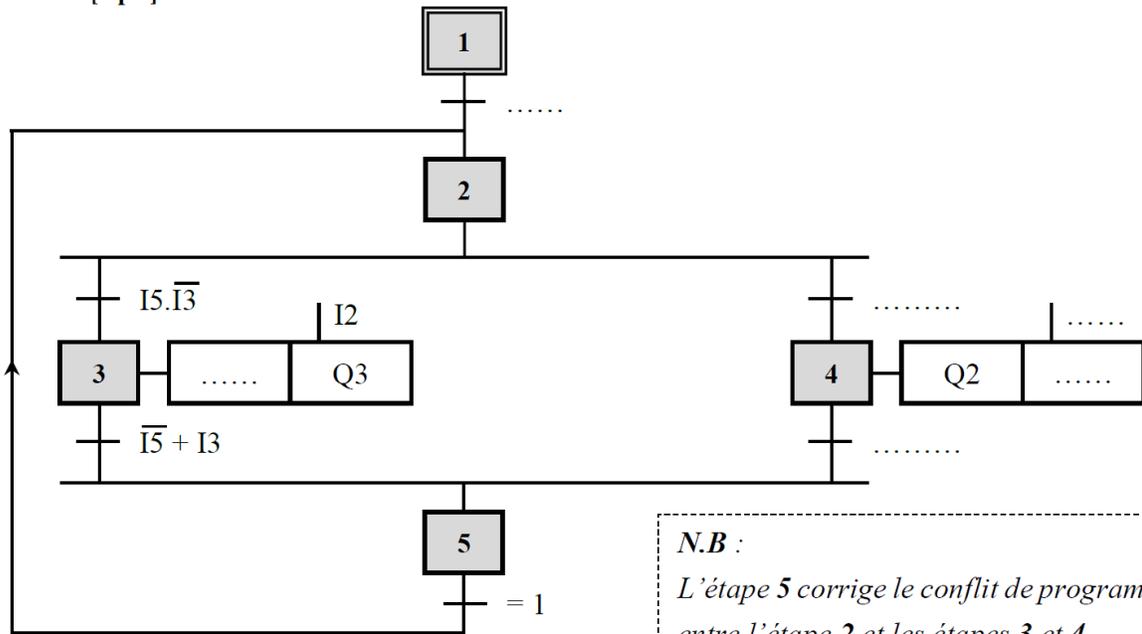
**Tableaux d'affectations des entrées et des sorties de l'API**

**DRES 03**

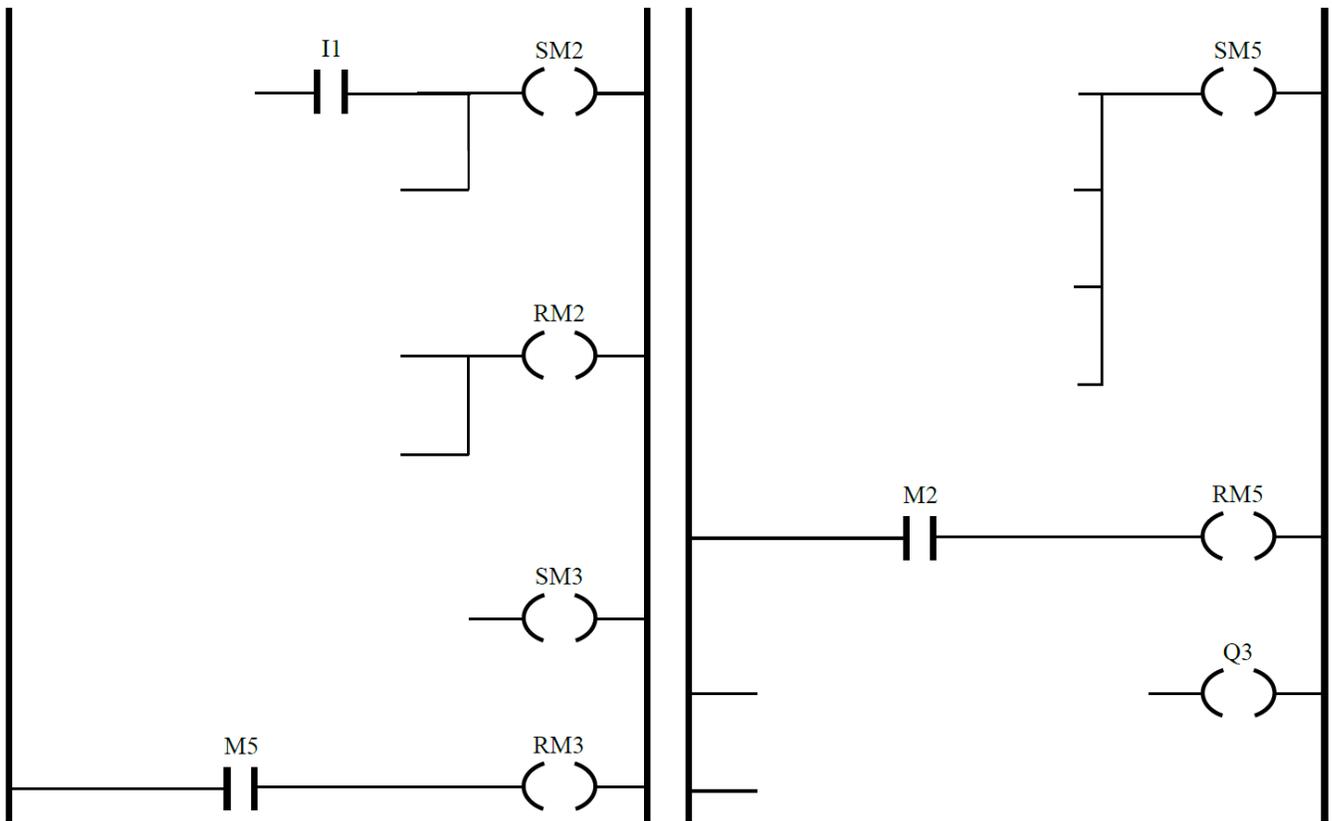
| Consigne ou compte rendu |                |                               | Entrée API |
|--------------------------|----------------|-------------------------------|------------|
| Bouton                   | S1             | Mise en service du système    | I1         |
| Détecteur d'obstacle     | P              | Obstacle détecté              | I2         |
| Fin de course            | F <sub>H</sub> | Limite haute du crochet       | I3         |
|                          | F <sub>B</sub> | Limite basse du crochet       | I4         |
| Joystick J1 à un axe     |                | Ordre de monter le crochet    | I5         |
|                          |                | Ordre de descendre le crochet | I6         |

| Ordre                          |     | Sortie API |
|--------------------------------|-----|------------|
| Monter le crochet              | KA1 | Q1         |
| Descendre le crochet           | KA2 | Q2         |
| Ralentir la vitesse du crochet | KA3 | Q3         |

Q.38- [4 pts]



Q.39- [4 pts]



**Exercice 7 : examen national 2024, session de rattrapage**
**Tâche 4 : Étude du grafcet point de vue commande**

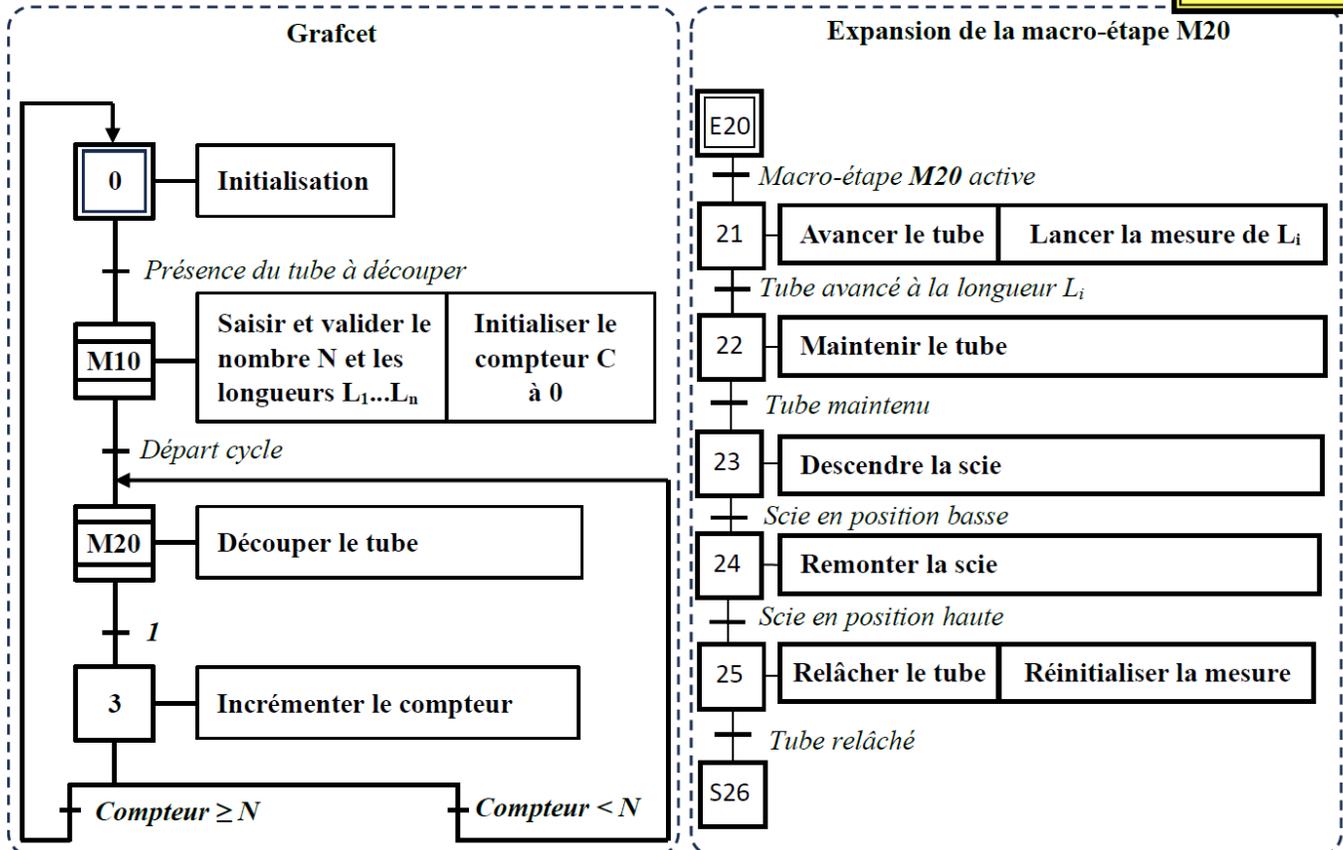
À partir du Grafcet (DRES 02) et des données des tableaux du DRES 03 :

Q46. Compléter, de point de vue commande, l'expansion de la macro-étape M20.

2 pts

**Grafcet de point de vue système**

**DRES 02**


**Tableau d'affectation des entrées**

**DRES 03**

| Fonction de l'entrée               | Capteur/Interface d'entrée                 |
|------------------------------------|--|
| Départ cycle                       | Dcy  |
| Présence tube                      | cp   |
| Longueur $L_n$ prédéfinie atteinte | m1 (Bit 0 de la case mémoire interne CMI)  |
| Macro-étape M20 active             | m20 (Bit 1 de la case mémoire interne CMI) |
| Tube maintenu                      | tm   |
| Tube relâché                       | tl   |
| Scie en position basse             | sb   |
| Scie en position haute             | sh   |

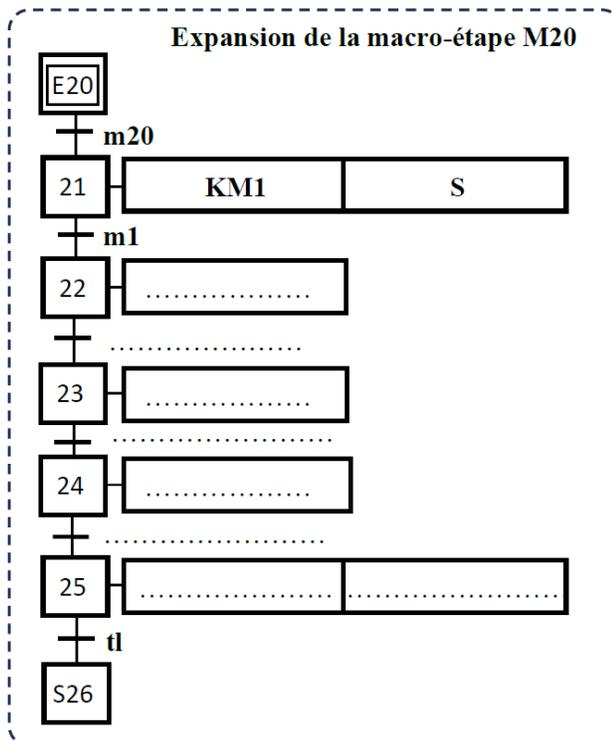
**Tableau d'affectation des sorties**

| Action                   | Actionneur                   | Pré actionneur |
|--------------------------|------------------------------|----------------|
| <i>Avancer le tube</i>   | <i>Moteur électrique M1</i>  | <b>KM1</b>     |
| <i>Maintenir le tube</i> | <i>Vérin double effet V2</i> | <b>V2+</b>     |
| <i>Relâcher le tube</i>  |                              | <b>V2-</b>     |
| <i>Remonter la scie</i>  | <i>Vérin double effet V1</i> | <b>V1-</b>     |
| <i>Descendre la scie</i> |                              | <b>V1+</b>     |

**Commande du capteur de distance**

| Fonction                       | Contact de commande |
|--------------------------------|---------------------|
| <i>Lancer la mesure</i>        | <b>S (Set)</b>      |
| <i>Réinitialiser la mesure</i> | <b>R (Reset)</b>    |

**Q46. Grafcet point de vue commande**



## 7

**MICROCONTROLEURS****Environnement microinformatique minimal**

Le contrôle d'un processus technique fait souvent appel à un dispositif de traitement programmable composé de deux éléments complémentaires indissociables :

- **Logiciel** : programme qui décrit la gestion du processus technique ;
- **Matériel** : dispositif électronique, à base de  $\mu P$ , qui héberge le programme et lui donne vie.

Ci-contre, la structure d'un système programmable minimal :

**1. Microprocesseur**

(ou CPU pour Central Processing Unit)

Le  $\mu P$  exécute le programme situé dans la mémoire de programme. Toute l'activité du  $\mu P$  est cadencée par une horloge.

**2. Mémoires**

Ce sont des composants chargés de conserver des contenus. En fonction des propriétés des mémoires, on distingue :

|        | Lecture | Écriture  | Volatile |
|--------|---------|---|----------|
| RAM    | Oui     | Oui   | Oui      |
| ROM    | Oui     | Non (programmée une seule fois au moment de la fabrication)                 | Non      |
| PROM   | Oui     | Programmable une seule fois par l'utilisateur                               | Non      |
| EEPROM | Oui     | Effaçable électriquement à l'aide d'une broche spécifique et reprogrammable | Non      |

- Une mémoire volatile perd son contenu à la coupure de l'alimentation ;
- En principe, la RAM sert de mémoire de données tandis que la ROM sert de mémoire de programme ;
- Une variante de l'EEPROM est la mémoire FLASH qui est rapide et accepte d'être reprogrammée un grand nombre de fois.

**3. Ports d'entrée/sortie**

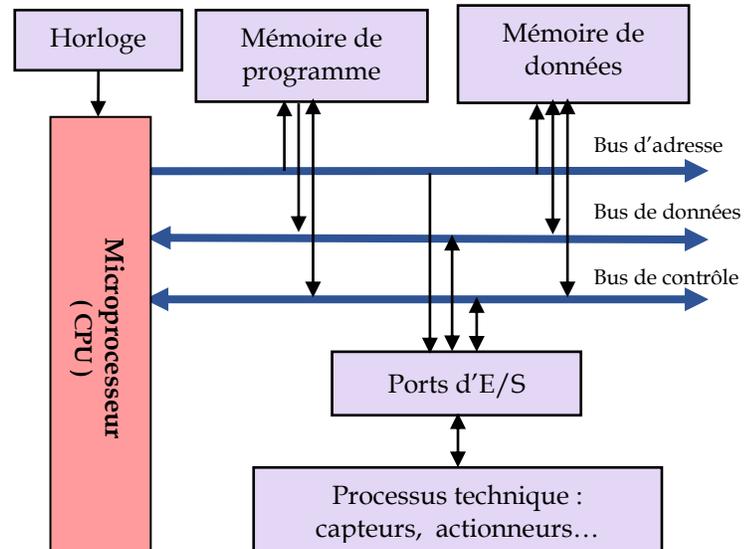
Ils ont pour rôle d'interfacer les périphériques afin de dialoguer avec le processus à gérer.

- ⇒ **Port parallèle** : les données sont transmises par paquets de 8, 16 ...bits.
- ⇒ **Port série** : les données sont transmises bit par bit sur un fil unique (exemples : USB, RS232, RS485).

**4. Bus**

Un bus est ensemble de fils destiné à véhiculer les données entre les différents composants du système. On distingue :

- **Bus de données** : à travers lequel transitent les données ;
- **Bus d'adresse** : à travers lequel le  $\mu P$  applique l'adresse de la donnée à lire ou à écrire ;
- **Bus de contrôle** : à travers lequel le  $\mu P$  dialogue avec le composant adressé.

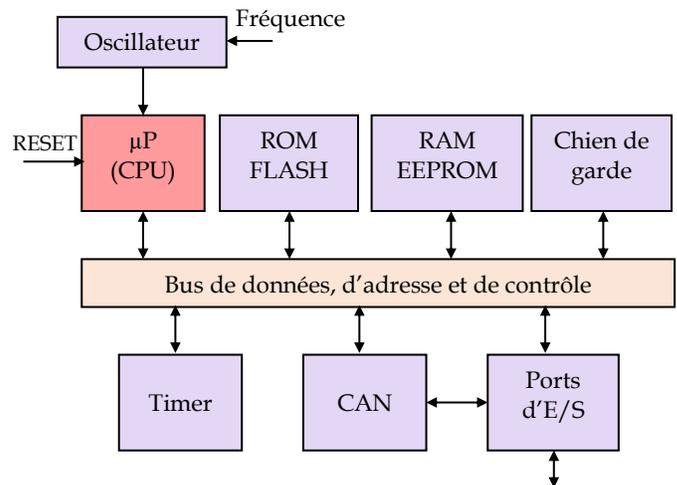


## Microcontrôleur PIC 16F877

### Généralités

Le  $\mu$ C se présente sous la forme d'un circuit intégré réunissant tous les éléments d'une structure à base de  $\mu$ P ainsi que certains modules auxiliaires :

- Un  $\mu$ P ;
- De la mémoire de données (RAM, EEPROM) ;
- De la mémoire de programme (FLASH, EEPROM) ;
- Des ports pour la connexion des entrées/sorties ;
- Des timers pour générer des impulsions, compter des évènements, mesurer des fréquences... ;
- Des CAN pour traiter des signaux analogiques ;
- Un chien de garde : qui détecte s'il y a un problème logiciel et provoque un RESET ;

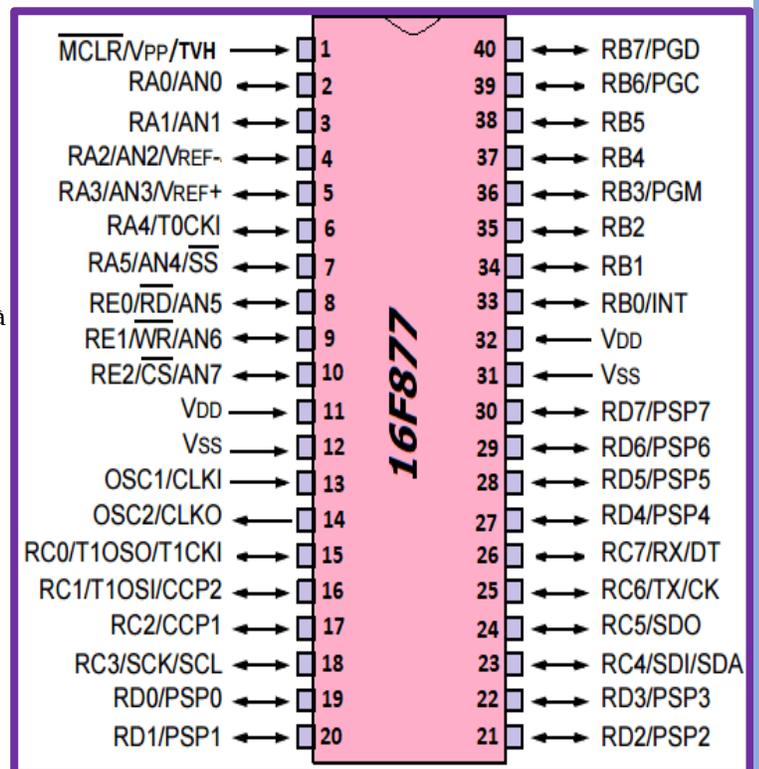


Aujourd'hui on trouve les  $\mu$ C presque partout : cafetière, réfrigérateur, four à micro-ondes, téléviseur, téléphone portable, imprimante, scanner, voiture (airbags, climatisation, alarme...)...

### Architecture externe (brochage) du PIC 16F877 de Microchip

La popularité du 16F877 vient de sa facilité de mise en œuvre et la disposition d'un environnement de développement complet totalement gratuit : MPLAB.

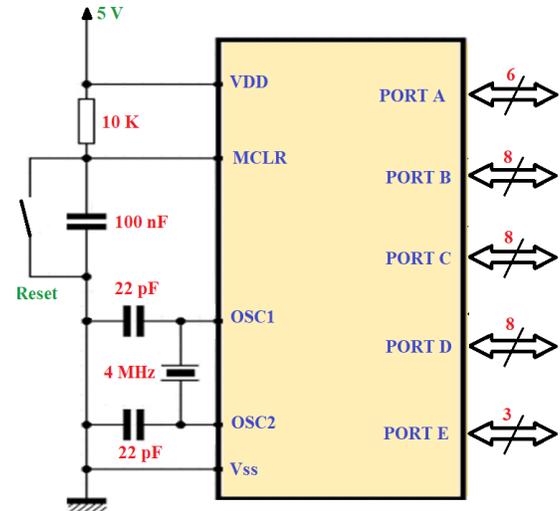
- L'alimentation du circuit est assurée par les pins  $VDD = +5V$  et  $VSS = 0V$  ;
- **OSC1** et **OSC2** : broches qui reçoivent un quartz ou un circuit RC pour l'horloge interne ;
- **MCLR** (Master Clear Reset) : broche de remise à zéro, active au niveau bas (0 V) ;
- **RA0 à RA5** : 6 broches E/S du **PORT A** ;  
**RB0 à RB7** : 8 broches E/S du **PORT B** ;  
**RC0 à RC7** : 8 broches E/S du **PORT C** ;  
**RD0 à RD7** : 8 broches E/S du **PORT D** ;  
**RE0 à RE2** : 3 broches E/S du **PORT E**.



### Mise en œuvre

Ci-contre, un montage minimal avec :

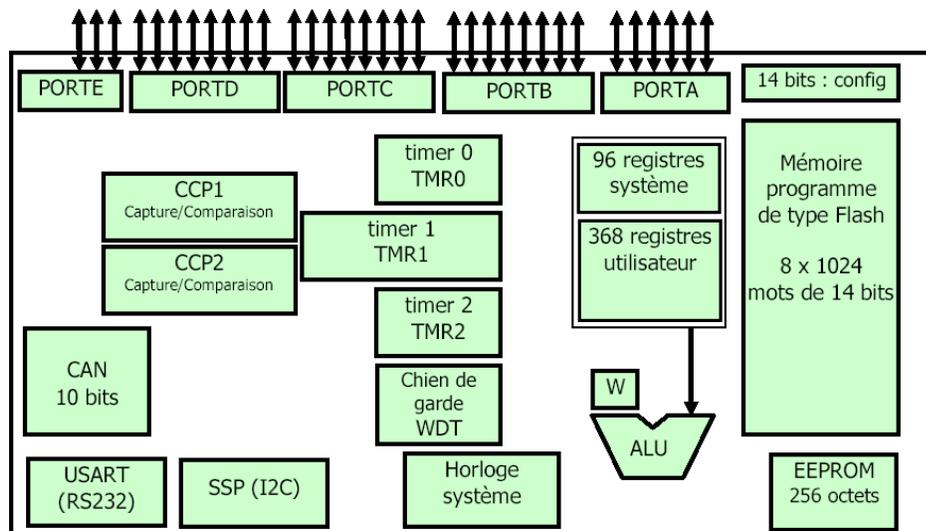
- Alimentation du composant (VDD = +5V et VSS = 0V) ;
- Pilotage par quartz, ici de 4 MHz (OSC1 et OSC2) ;
- RESET manuel par un bouton poussoir ;
- RESET automatique à chaque mise sous tension.  
En effet, le condensateur est initialement déchargé (MCLR = 0), puis se charge jusqu'à VDD (MCLR = 1).



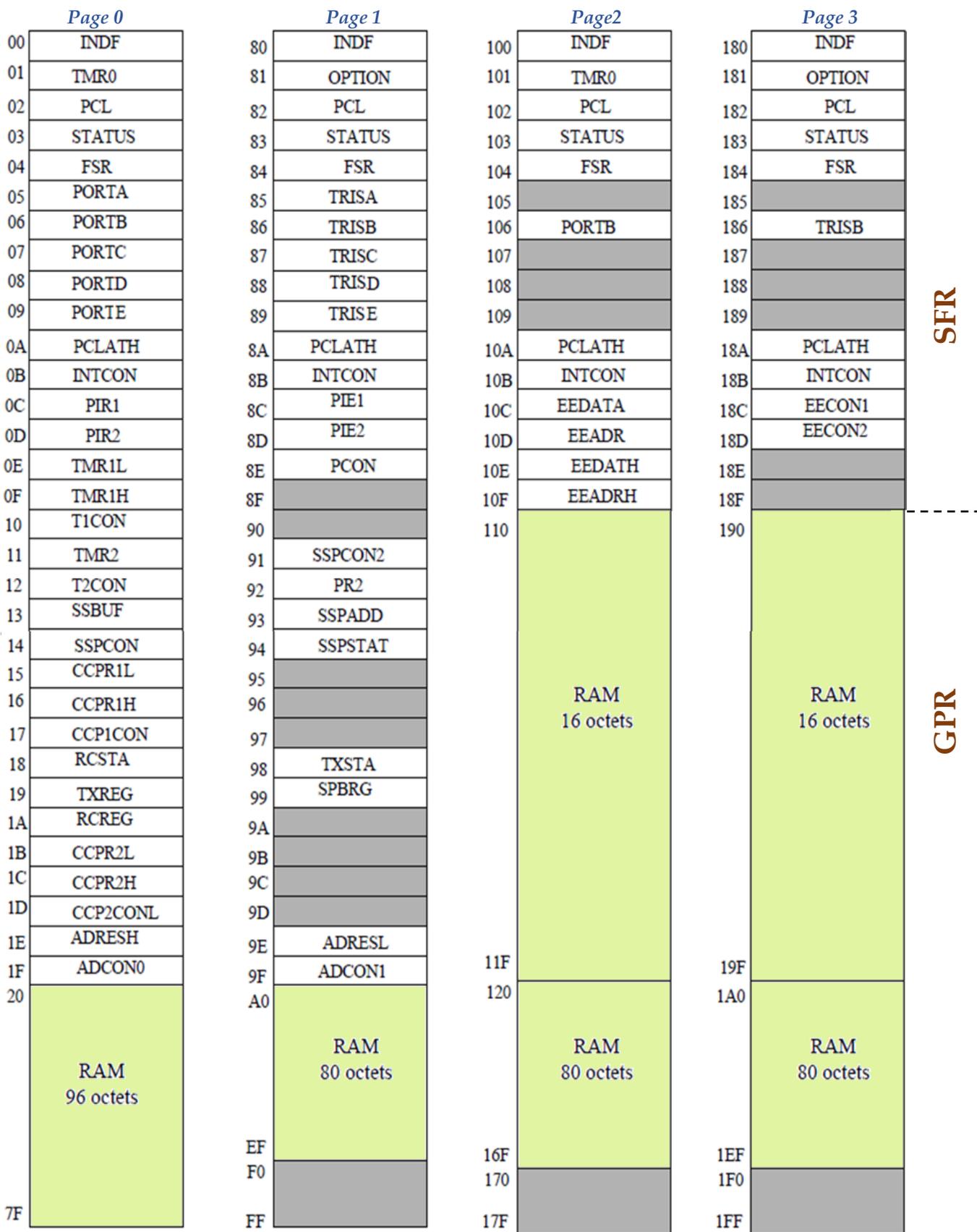
### Architecture interne

Les caractéristiques principales du 16F877 sont :

- Une **mémoire programme** de type flash de **8K mots de 14 bits** de h'0000' à h'1FFF' (1K = 2<sup>10</sup> = 1024) ;
- Une **mémoire RAM** constituée :
  - **96 registres SFR** (Special Function Register) : sont les registres de fonctionnement du PIC et ayant des fonctions spéciales ;
  - **368 registres GPR** (General Propose Register) : sont des positions mémoire que l'utilisateur peut utiliser pour stocker ses variables.
- Une **mémoire EEPROM** Data de données de **256 octets** ;
- **3 Timers** avec prédiviseur programmable : 2 timers 8 bits **TMR0**, **TMR2** et un autre 16 bits **TMR1** ;
- Un chien de garde **WDT** ;
- **CAN 10 bits** à approximations successives à 8 entrées multiplexées ;
- 33 Entrée-Sortie bidirectionnelles réparties sur **5 ports** : **PORTA** à 6 bits, **PORTB**, **PORTC**, **PORTD** à 8 bits et **PORTE** à 3 bits ;
- 1 registre de travail (accumulateur) **W** à 8 bits ;
- Bus DATA sur 8 bits ;
- Jeu de **35 instructions** de durée 1 ou 2 cycles ;
- Plusieurs sources d'interruption ;
- Autres modules : **USART** pour transmission série, interface **I2C**, 2 modules pour **PWM**...



La RAM de données



## Programmation du PIC 16F877 en langage assembleur

### Jeu d'instructions du 16F877

| <i>INSTRUCTIONS OPERANT SUR UN REGISTRE</i> |      |   | Indicateurs | Cycles |
|---|------|---|-------------|--------|
| <b>ADDWF</b>                                | F, d | W+F → {W, F ? d}                                | C, DC, Z    | 1      |
| <b>ANDWF</b>                                | F, d | W AND F → {W, F ? d}                            | Z           | 1      |
| <b>CLRF</b>                                 | F    | 0 → F   | Z           | 1      |
| <b>CLRW</b>                                 |      | 0 → W   | Z           | 1      |
| <b>CLRWD</b>                                |      | 0 → WDT   | TO', PD'    | 1      |
| <b>COMF</b>                                 | F, d | Complément F → {W, F ? d}                       | Z           | 1      |
| <b>DECF</b>                                 | F, d | Décrémente F → {W, F ? d}                       | Z           | 1      |
| <b>DECFSZ</b>                               | F, d | Décrémente F → {W, F ? d} et saut si 0          |             | 1(2)   |
| <b>INCF</b>                                 | F, d | Incrémente F → {W, F ? d}                       | Z           | 1      |
| <b>INCFSZ</b>                               | F, d | Incrémente F → {W, F ? d} et saut si 0          |             | 1(2)   |
| <b>IORWF</b>                                | F, d | W OR F → {W, F ? d}                             | Z           | 1      |
| <b>MOVF</b>                                 | F, d | F → {W, F ? d}                                  | Z           | 1      |
| <b>MOVWF</b>                                | F    | W → F   |             | 1      |
| <b>RLF</b>                                  | F, d | Rotation à gauche de F à travers C → {W, F ? d} | C           | 1      |
| <b>RRF</b>                                  | F, d | Rotation à droite de F à travers C → {W, F ? d} |             | 1      |
| <b>SUBWF</b>                                | F, d | F - W → {W, F ? d}                              | C, DC, Z    | 1      |
| <b>SWAPF</b>                                | F, d | Permute les 2 quartets de F → {W, F ? d}        |             | 1      |
| <b>XORWF</b>                                | F, d | W XOR F → {W, F ? d}                            | Z           | 1      |
| <i>INSTRUCTIONS OPERANT SUR UN BIT</i>      |      |   |             |        |
| <b>BCF</b>                                  | F, b | 0 → bit b du registre F                         |             | 1      |
| <b>BSF</b>                                  | F, b | 1 → bit b du registre F                         |             | 1      |
| <b>BTFSZ</b>                                | F, b | Test du bit b de F et saut si 0                 |             | 1(2)   |
| <b>BTFS</b>                                 | F, b | Test du bit b de F et saut si 1                 |             | 1(2)   |
| <i>INSTRUCTIONS OPERANT SUR UNE DONNEE</i>  |      |   |             |        |
| <b>ADDLW</b>                                | K    | W + K → W                                       | C, DC, Z    | 1      |
| <b>ANDLW</b>                                | K    | W AND K → W                                     | Z           | 1      |
| <b>IORLW</b>                                | K    | W OR K → W                                      | Z           | 1      |
| <b>MOVLW</b>                                | K    | K → W   |             | 1      |
| <b>SUBLW</b>                                | K    | K - W → W                                       | C, DC, Z    | 1      |
| <b>XORLW</b>                                | K    | W XOR K → W                                     | Z           | 1      |
| <i>INSTRUCTIONS GENERALES</i>               |      |   |             |        |
| <b>CALL</b>                                 | L    | Branchement à un sous-programme de label L      |             | 2      |
| <b>GOTO</b>                                 | L    | Branchement à la ligne de label L               |             | 2      |
| <b>NOP</b>                                  |      | Pas d'opération                                 |             | 1      |
| <b>RETURN</b>                               |      | Retour d'un sous-programme                      |             | 2      |
| <b>RETFIE</b>                               |      | Retour d'interruption                           |             | 2      |
| <b>RETLW</b>                                | K    | Retour d'un sous-programme avec K dans W        |             | 2      |
| <b>SLEEP</b>                                |      | Mode standby                                    | TO', PD'    | 1      |

La notation {W, F ? d} signifie que le résultat va soit dans W si d=0 ou w, soit dans F si d= 1 ou f.

⇒ Accès à la RAMRegistre STATUS

|     |     |     |    |    |   |    |   |
|-----|-----|-----|----|----|---|----|---|
| IRP | RP1 | RP0 | T0 | PD | Z | DC | C |
|-----|-----|-----|----|----|---|----|---|

Les bits **RP1** et **RP0** du registre **STATUS** permettent la sélection des pages de la RAM selon ce tableau :

| RP1 | RP0 | Page sélectionnée |
|-----|-----|-------------------|
| 0   | 0   | Bank 0            |
| 0   | 1   | Bank 1            |
| 1   | 0   | Bank 2            |
| 1   | 1   | Bank 3            |

⇒ Expression des nombres

L'assembleur considère les nombres dans la base indiquée par la spécification adjointe :

| Base        | Indicateur | Exemple              |
|-------------|------------|----------------------|
| Décimal     | d          | d'27'                |
| Hexadécimal | h<br>0x    | h'1B' ou 0x1B<br>1Bh |
| Binaire     | b          | b'00011011'          |

**Exercice**

En utilisant le plan d'organisation de la RAM, trouver la page concernée.

| Registre | Bank  |
|----------|-------|
| PORTA    | ..... |
| TRISA    | ..... |
| TMR0     | ..... |
| ADCON1   | ..... |

**Exercice**

| Action                   | Code assembleur   |
|--------------------------|---|
| Sélectionner la page 0   | BCF 0x03, 6                      ou mieux                      BCF STATUS, RP1<br>.....             |
| Sélectionner la page 2   | .....<br>.....  |
| Mettre à 0 le registre W | ..... ; par CLRW<br>autrement<br>..... ; par AND<br>ou encore<br>..... ; charger la valeur 0 dans W |

|  |  |
|--|--|
| <b>Incrémenter le registre GPR d'adresse 0x20</b>    | <pre> ..... ; copier le registre 0x20 dans W ..... ; Incrémenter le registre W ..... ; copier W dans 0x20 ou mieux ..... ; incrémenter le registre 0x20 par incf </pre>                                    |
| <b>Charger le registre PORTB par le mot 10000001</b> | <pre> ..... ; RAZ du registre PORTB ..... ; forcer le bit b<sub>0</sub> à 1 ..... ; forcer le bit b<sub>7</sub> à 1 autrement ..... ; charger l'octet '10000001' dans W ..... ; copier W dans PORTB </pre> |
| <b>Copier le contenu de PORTA dans PORTB</b>         | <pre> ..... ; copier le contenu de PORTA dans W ..... ; copier W dans PORTB </pre>   |

⇒ Ports d'entrées/ sortie

**PORTA** : pins RA0 à RA5  
**PORTB** : pins RB0 à RB7  
**PORTC** : pins RC0 à RC7  
**PORTD** : pins RD0 à RD7  
**PORTE** : pins RE0 à RE2

- La direction des données sur les ports est programmée à travers les registres **TRISA**, **TRISB**, **TRISC**, **TRISD** et **TRISE** de la façon suivante :
  - 0** → configure la broche correspondante en **sortie** ;
  - 1** → configure la broche correspondante en **entrée**.

Exemples : la valeur '00000011' dans TRISB configure les broches RB0 et RB1 en entrée et les autres en sortie.

- La manipulation des données (lecture/écriture) sur les ports est programmée à travers les registres PORTA, PORTB, PORTC, PORTD et PORTE.

Exemple :

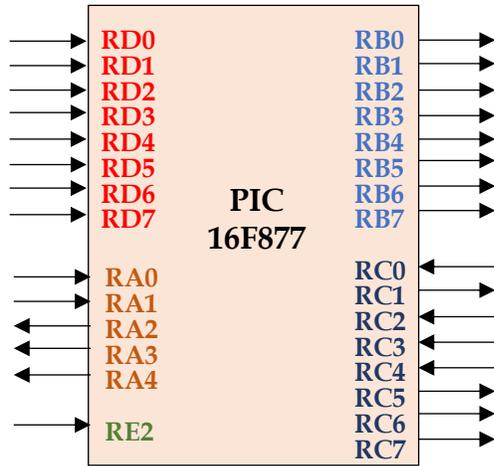
```

bsf PORTC, 7 → positionne la broche RC7 à 1 (écriture).
movwf PORTC → permet de sortir le contenu de W sur PORTC (écriture).
movf PORTC, w → récupère l'état de PORTC dans W (lecture).

```

**Exercice**

Ecrire le programme assembleur permettant la configuration suivante :



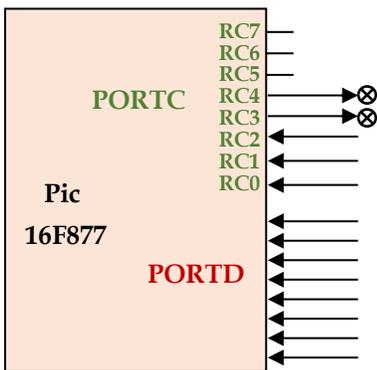
TRISA = .....  
 TRISB = .....  
 TRISC = .....  
 TRISD = .....  
 TRISE = .....

```

..... ; activer banque 1
.....
..... ; configurer port A
.....
..... ; configurer port B
..... ; configurer port C
..... ; configurer port D
..... ; configurer port E
..... ; activer banque 0
.....
    
```

**Exercice**

- Configurer les ports conformément à la figure
- Allumer les 2 LED du PORTC
- Lire le contenu de PORTD pour le stocker dans le registre d'adresse 7F



|       |       |
|-------|-------|
| TRISC | ..... |
| TRISD | ..... |

```

..... ; activer page 1
.....
..... ; configurer le port C
.....
..... ; configurer le port D
.....
..... ; retour à page 0
.....
..... ; allumer les 2 Leds du portC
.....
..... ; lire et ranger la valeur du port D
..... ; dans le registre 7F
    
```

**Exercice**

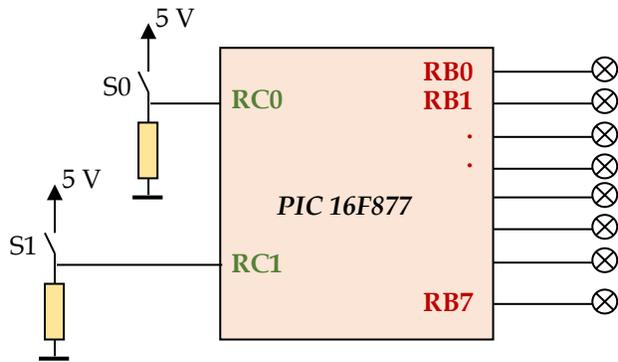
Au repos les LEDs sont éteintes.

Une action sur le bouton poussoir **S1** allume toutes les LEDs.

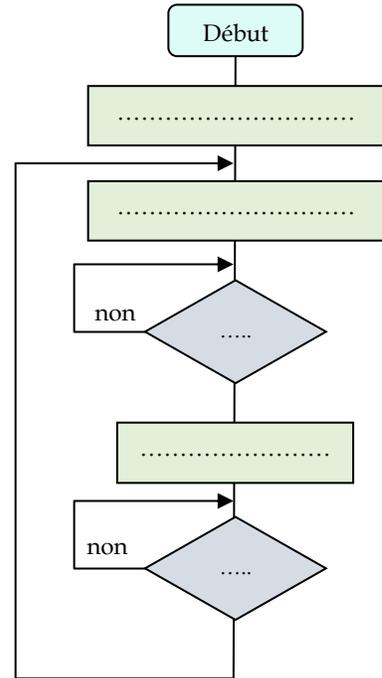
Une action sur le bouton poussoir **S0** éteint toutes les LEDs.

Et ainsi de suite.

Fournir l'organigramme et le programme



|       |       |
|-------|-------|
| TRISB | ..... |
| TRISC | ..... |



```

..... ; page 1
.....
..... ; configurer PORTB en sortie
..... ; configurer PORTC
.....
..... ; page 0
.....

boucle1 ..... ; éteindre les LEDs
boucle2 ..... ; Si RC1 = 1 (signifiant si S1 appuyé)
..... ; sinon se brancher à "boucle2"
..... ; si oui allumer les LEDs
.....

boucle3 ..... ; Si RC0 = 1
..... ; sinon se brancher à "boucle3"
..... ; Se brancher à "boucle1" pour éteindre de nouveau
    
```

⇒ Techniques de comparaison des registres

**Bits indicateurs d'état Z et C du registre STATUS**

Le bit **Z** est l'indicateur de **Zéro** : **Z = 1** si le résultat d'une opération arithmétique ou logique est **zéro** ;  
**Z = 0** si le résultat est **différent de zéro**.

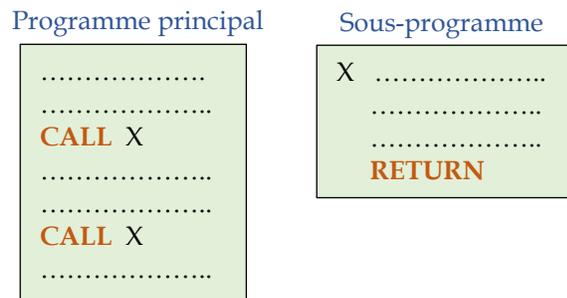
Le bit **C** est l'indicateur de **signe** : **C = 1** si le résultat d'une soustraction est **positif** ;  
**C = 0** si le résultat est **négatif**.

F1 et F2 étant deux registres, compléter ce tableau :

| Action     | Code assembleur  |
|------------|--|
| Si F1 = 0  | ..... ; copier F1 dans W puis contrôler le bit Z<br>.....                                  |
| Si F1 = F2 | ..... ; effectuer la soustraction entre F1 et F2 puis contrôler le bit Z<br>.....<br>..... |
| Si F1 < F2 | ..... ; effectuer la soustraction entre F1 et F2 puis contrôler le bit C<br>.....<br>..... |

⇒ Temporisation et sous-programme

Un sous-programme est une séquence d'instructions chargée d'effectuer un traitement particulier.  
 Les sous-programmes permettent ainsi de décomposer un programme en plusieurs sous-parties.



**CALL** : instruction d'appel du sous-programme.

**RETURN** : instruction de retour au programme appelant.

**Exercice**

Faire clignoter à la fois et à 1 Hz, les 8 LED supposées montées sur PORTB.  
On suppose déclaré un sous-programme "tempo" qui réalise une temporisation de 0,5 s.

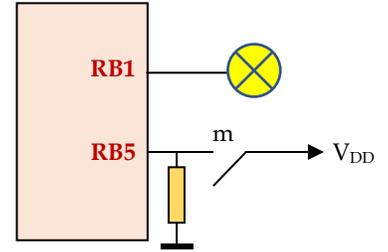
```
loop ..... ; Éteindre les LED
..... ; Temporiser 0,5 s
..... ; Allumer les LED
.....
..... ; Temporiser 0,5 s
..... ; répéter
```

Variante utilisant l'instruction **comf F, d**

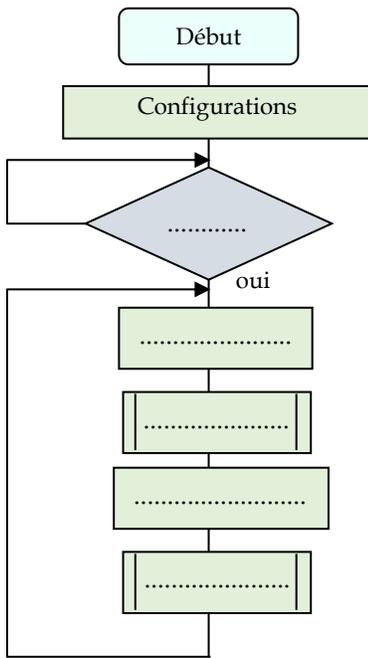
```
..... ; Éteindre les LED
loop ..... ; Temporiser 0,5 s
..... ; complémenter PORTB
..... ; répéter
```

**Exercice**

Faire clignoter la LED du montage ; *m* est le bouton marche.  
 Réaliser la temporisation par une boucle de retard (temporisation logicielle) et la confier à un sous-programme.



Programme principal

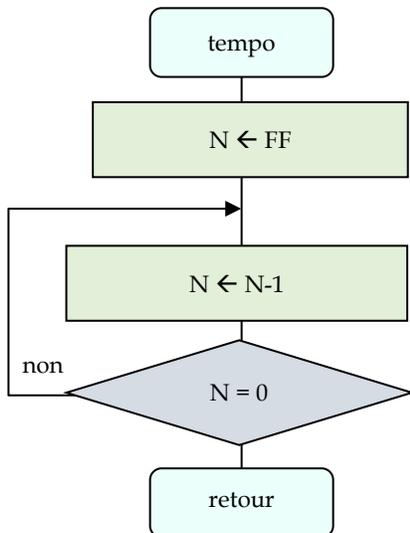


```

N equ 0x20           ; Initialisation des variables
.....
.....              ; page 1
.....
.....              ; configurer port B
.....
.....              ; page 0
test .....          ; tester si RB5 = 1 et saut si vrai
.....              ; revenir à 'test'
lab .....           ; allumer
.....              ; appel du sous-programme 'tempo'
.....              ; éteindre
.....              ; temporisation
.....              ; revenir à 'lab'

tempo Movlw 0xFF     ; sous prog. de temporisation
      movwf N
boucle Decfsz N,f
      Goto boucle
      Return
    
```

Sous-programme de temporisation



```

tempo .....         ; N ← FF
.....
loop .....          ; décrémenter N et tester si N = 0
.....              ; sinon retour à l'étiquette 'loop'
return .....        ; retour au programme principal
    
```

### Durée de temporisation du sous-programme

La fréquence  $f$  de fonctionnement interne du PIC est le quart de celle du quartz ; soit  $f = F_{osc}/4$ .

- Pour un quartz de  $F_{osc} = 20 \text{ MHz} \Rightarrow f = 5 \text{ MHz} \Rightarrow 1 \text{ cycle} = 1/f = 0,2 \mu\text{s}$  ;
- Pour un quartz de  $F_{osc} = 4 \text{ MHz} \Rightarrow f = 1 \text{ MHz} \Rightarrow 1 \text{ cycle} = 1 \mu\text{s}$ .

- Pour un quartz de **4MHz**, 1 cycle instruction = **1 $\mu$ s** ;
- Toutes les instructions du sous prog. prennent 1 cycle sauf goto et return qui en prennent 2.

Calcul de la durée la temporisation :

.....

.....

.....

.....

.....

.....

### Multiples durées de temporisation

Augmenter la durée de temporisation peut être obtenu par imbrication des boucles de retard.

```
tempo .....
.....
loop .....
.....
return
```

Durée T = ..... avec un quartz de .....

```
tempo1 .....
.....
loop1 .....
.....
return
```

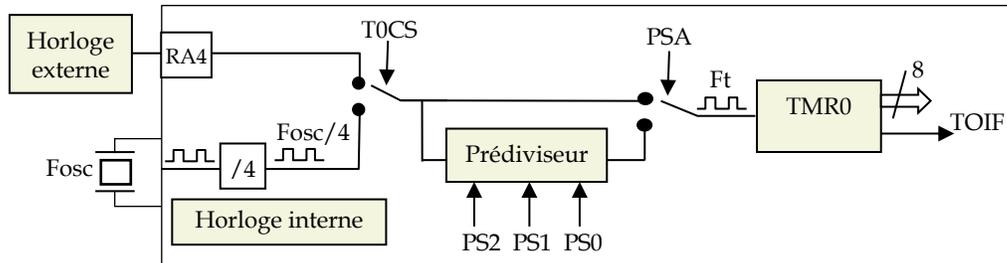
Durée T1 = .....

```
tempo2 .....
.....
loop2 .....
.....
return
```

Durée T2 = .....

### ⇒ Module Timer 0

- Le Timer 0 est un compteur 8 bits qui peut servir de temporisateur (timer) ou de compteur d'évènements ;
- TMR0 est un registre associé au Timer 0.



### Configuration

Timer 0 est configuré par le registre OPTION\_REG

| Registre <b>OPTION_REG</b><br>(pages 1 et 3) | RBPU | INTEDG | <b>T0CS</b> | TOSE | <b>PSA</b> | <b>PS2</b> | <b>PS1</b> | <b>PS0</b> |
|--|------|--------|-------------|------|------------|------------|------------|------------|
|--|------|--------|-------------|------|------------|------------|------------|------------|

- L'horloge peut être interne ou externe :
  - T0CS = 0** → horloge interne (de fréquence Fosc/4) ;
  - T0CS = 1** → horloge externe appliquée à la broche RA4.
- On peut ou non utiliser le prédiviseur :
  - PSA = 0** → le prédiviseur est affecté au timer TMR0 ;
  - PSA = 1** → le prédiviseur est affecté au Watchdog WDT.

Le facteur de division de la fréquence est fixé par les bits PS2, PS1 et PS0

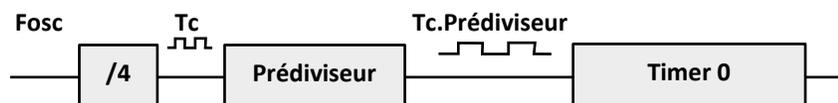
| <b>PS2 PS1 PS0</b>                       | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |
|--|-----|-----|-----|-----|-----|-----|-----|-----|
| <b>Division de fréquence de TMR0 par</b> | 2   | 4   | 8   | 16  | 32  | 64  | 128 | 256 |

### Timer 0 en mode timer

Dans ce mode, Timer 0 est piloté par l'horloge interne du microcontrôleur et sert de temporisateur.

La durée de temporisation est  **$T = T_c \cdot \text{Prédiviseur} \cdot (256 - \text{TMR0})$**

- T<sub>c</sub>** : durée d'un cycle horloge.
- Prédiviseur** : valeur de prédivision.
- TMR0** : valeur de départ du registre TMR0.



### Temporisation par scrutation du débordement du Timer 0

Au débordement du timer (passage de FF à 00), le bit **TOIF** du registre **INTCON** passe à 1.

| Registre <b>INTCON</b><br>(pages 0, 1, 2 et 3) | GIE | PEIE | TOIE | INTE | RBIE | <b>TOIF</b> | INTF | RBIF |
|--|-----|------|------|------|------|-------------|------|------|
|--|-----|------|------|------|------|-------------|------|------|

**Exercice**

On désire obtenir une temporisation de **10 ms**, trouver le mot de configuration (*OPTION\_REG*) et la valeur initiale de *TMR0*.  
On donne **Fosc = 4 MHz**.

Écrire, ensuite, un sous-programme qui met en œuvre cette temporisation en scrutant le bit *TOIF*.

En mode timer, on utilise l'horloge interne : *T0CS* = ....

Et si on choisit une prédivision = ....., le mot de configuration sera :

| OPTION_REG | RBPB  | INTEDG | TOCS  | TOSE  | PSA   | PS2   | PS1   | PS0   |
|------------|-------|--------|-------|-------|-------|-------|-------|-------|
| .....      | ..... | .....  | ..... | ..... | ..... | ..... | ..... | ..... |

Avec un quartz de  $F_{osc} = 4\text{MHz}$ , la durée d'un cycle horloge est .....

.....  
 .....  
 .....

Sous-programme de temporisation de 10 ms par scrutation du bit *TOIF*

```

...
..... ; configurer Timer 0 via OPTION_REG
.....
...
tempo10ms ..... ; valeur de départ du registre TMR0
.....
..... ; baisser le drapeau càd mettre TOIF = 0
attente ..... ; tester si TOIF = 1
..... ; sinon revenir à 'attente'
..... ; retour au programme principal

```

**Exercice**

Donner la valeur initiale de *TMR0* et trouver le mot de configuration(*OPTION\_REG*) pour obtenir la temporisation maximale possible. On donne **Fosc = 4 MHz**.

Écrire, ensuite, un sous-programme qui met en œuvre cette temporisation.

.....  
 .....

| OPTION_REG | RBPB  | INTEDG | TOCS  | TOSE  | PSA   | PS2   | PS1   | PS0   |
|------------|-------|--------|-------|-------|-------|-------|-------|-------|
| .....      | ..... | .....  | ..... | ..... | ..... | ..... | ..... | ..... |

Sous-programme de temporisation maximale

```

...
..... ; configurer Timer 0 via OPTION_REG
.....
...
tempomax ..... ; valeur de départ du registre TMR0
..... ; baisser le drapeau càd mettre TOIF = 0
attente ..... ; tester si TOIF = 1
..... ; sinon revenir à 'attente'
..... ; retour au programme principal

```

⇒ Le module WDT Watchdog Timer ou chien de garde

C'est un compteur 8 bits incrémenté par une horloge indépendante de l'horloge système (même si le  $\mu$ C est en mode sleep). Lorsqu'il déborde (après 18 ms), deux situations sont possibles :

- Si le  $\mu$ C est en fonctionnement normal, cela provoque un RESET ;
- Si le  $\mu$ C est en mode SLEEP, cela provoque un WAKE-UP (l'exécution du programme reprend).

Configuration

|  |      |        |      |      |            |            |            |            |
|--|------|--------|------|------|------------|------------|------------|------------|
| Registre <b>OPTION_REG</b><br>(pages 1 et 3) | RBPU | INTEDG | T0CS | T0SE | <b>PSA</b> | <b>PS2</b> | <b>PS1</b> | <b>PS0</b> |
|--|------|--------|------|------|------------|------------|------------|------------|

Il est possible d'augmenter la durée par utilisation du prédiviseur (celui partagé avec le timer 0). Le coefficient de prédivision est fixé par les bits PS0, PS1 et PS2 :

- **PSA = 0** → le prédiviseur est affecté au timer TMR0 ;
- **PSA = 1** → le prédiviseur est affecté au Watchdog WDT.

Le coefficient de prédivision de la fréquence est fixé par les bits PS2, PS1 et PS0 :

|                                  |     |     |     |     |     |     |     |     |
|----------------------------------|-----|-----|-----|-----|-----|-----|-----|-----|
| <b>PS2 PS1 PS0</b>               | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |
| <b>Division de fréquence par</b> | 1   | 2   | 4   | 8   | 16  | 32  | 64  | 128 |

Le WDT est activé par la directive `__CONFIG __WDT_ON`.

Mode SLEEP

Dans ce mode, l'horloge système est arrêtée ce qui suspend l'exécution du programme.

Pour sortir du mode SLEEP et donc continuer l'exécution, il faut provoquer un WAKE-UP par :

- Un RESET externe dû à l'initialisation du PIC ; dans ce cas, l'exécution recommence ;
- Ou par un débordement du chien de garde WDT s'il est activé ; dans ce cas, l'exécution continue.

Temporisation par le WDT combiné au mode Sleep

On met le  $\mu$ C dans le mode Sleep ; le programme est alors suspendu et reprendra après débordement du WDT.

**Exercice**

Faire clignoter une LED montée sur la ligne RC0.

On optera pour une prédivision par ..... donc PSA = ..... et PS2 PS1 PS0 = .....

Donc

|                   |       |        |       |       |            |            |            |            |
|-------------------|-------|--------|-------|-------|------------|------------|------------|------------|
| <b>OPTION_REG</b> | RBPU  | INTEDG | T0CS  | T0SE  | <b>PSA</b> | <b>PS2</b> | <b>PS1</b> | <b>PS0</b> |
|                   | ..... | .....  | ..... | ..... | .....      | .....      | .....      | .....      |

La durée du WDT étant de .....  
Avec le prédiviseur elle devient .....

```

..... ; activer WDT
...
..... ; configurer WDT via OPTION_REG
.....
...

loop ..... ; allumer la LED
..... ; passer en mode sleep. Réveil après débordement de WDT
..... ; éteindre la LED
..... ; passer en mode sleep. Réveil après débordement de WDT
goto loop

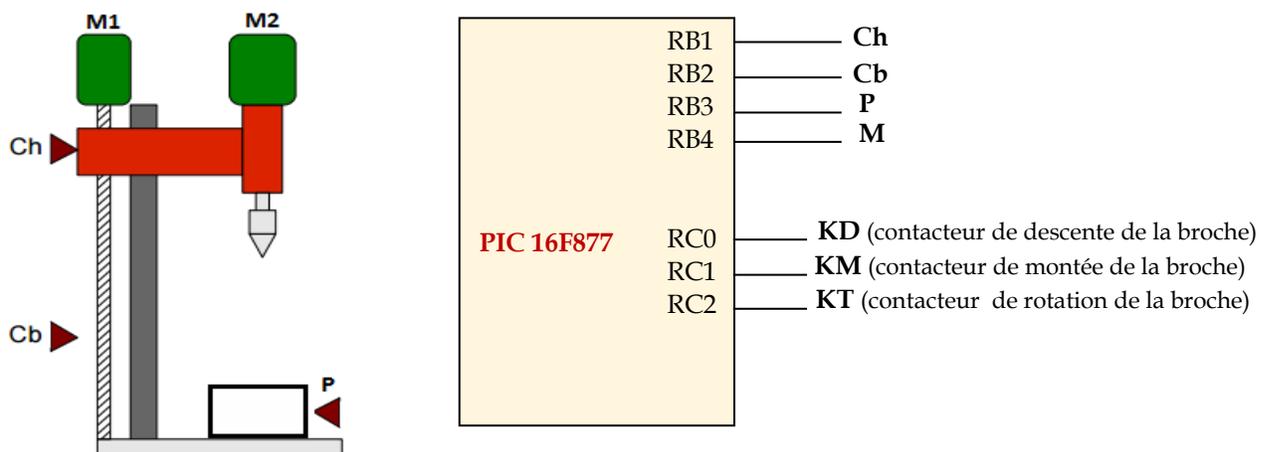
```

### Durées possibles du WDT

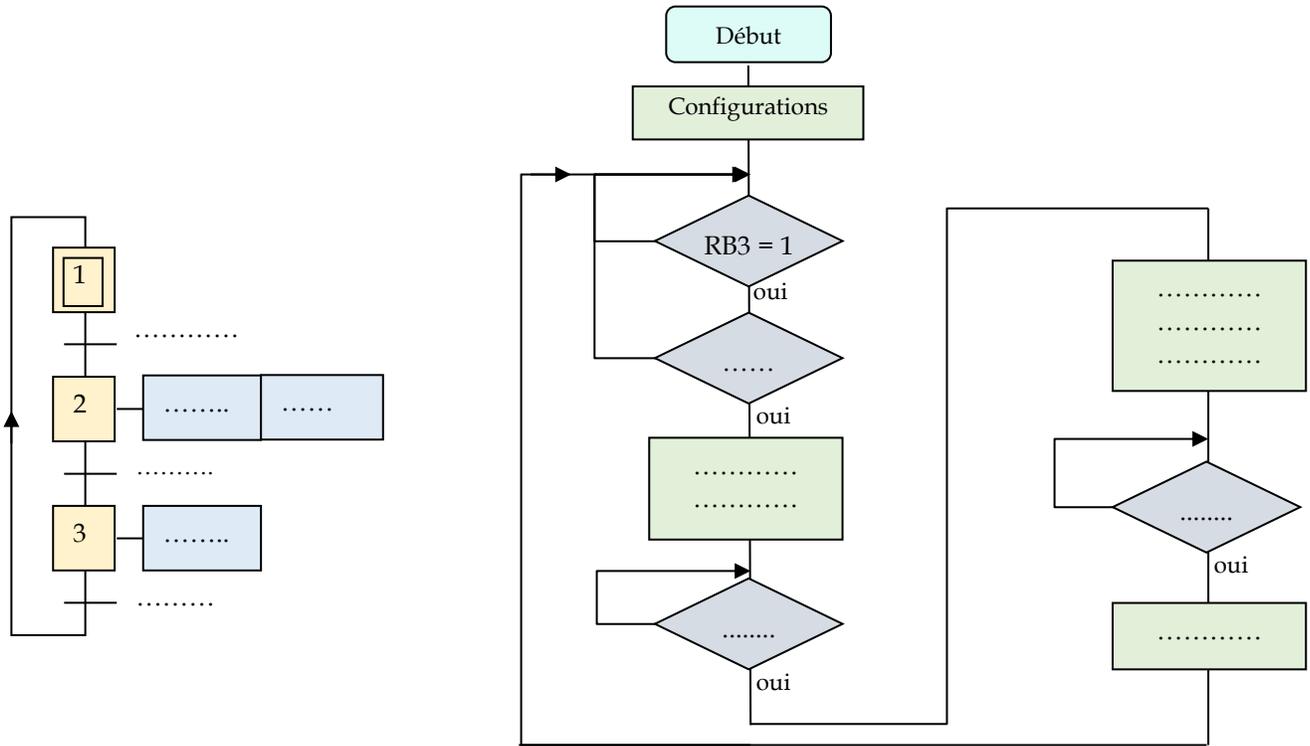
| PSA | PS2, PS1,<br>PS0 | Taux de<br>prévision du<br>WDT | Mot de configuration<br>(OPTION_REG) | Durée indicative |
|-----|------------------|--------------------------------|--------------------------------------|------------------|
| 1   | 000              | .....                          | .....                                | .....            |
| 1   | 001              | .....                          | .....                                | .....            |
| 1   | 010              | .....                          | .....                                | .....            |
| 1   | 011              | .....                          | .....                                | .....            |
| 1   | 100              | .....                          | .....                                | .....            |
| 1   | 101              | .....                          | .....                                | .....            |
| 1   | 110              | .....                          | .....                                | .....            |
| 1   | 111              | .....                          | .....                                | .....            |

### Implantation du GRAFCET dans un microcontrôleur

Le système est une perceuse automatique simple. Lorsque la pièce à usiner est en place (P), un appui sur le bouton M lance le cycle : la machine descend jusqu'à Cb puis remonte pour s'arrêter en Ch.



Fournir le GRAFCET, le traduire en organigramme puis en un programme assembleur pour le microcontrôleur PIC16F877.



```

bcf STATUS, RP1                ; page 1
bsf STATUS, RP0
.....                          ; configurer le port B
.....
.....                          ; configurer le port C
bcf STATUS, RP0                ; page 0
bcf STATUS, RP1

test1 .....                   ; si RB3 = 1
.....
.....                          ; si RB4 = 1
.....
.....                          ; RC0 ← 1
.....                          ; RC2 ← 1
test2 .....                   ; si RB2 = 1
.....
.....                          ; RC0 ← 0
.....                          ; RC2 ← 0
.....                          ; RC1 ← 1
test3 .....                   ; si RB1 = 1
.....
.....                          ; RC1 ← 0
.....                          ; recommencer
    
```

### ⇒ Interruptions

Une interruption provoque l'arrêt du programme principal pour aller exécuter un sous-programme, dit d'interruption, qui se trouve à l'adresse **0004**.

À la fin de cette procédure, le microcontrôleur reprend le programme principal à l'endroit où il l'a laissé.

### Sources d'interruption

| <b>Interruption</b> | <b>Source d'interruption</b>   | <b>Bit de validation</b> | <b>Flag</b> | <b>Interruption périphérique</b> |
|---------------------|--|--------------------------|-------------|----------------------------------|
| <b>TOI</b>          | Débordement de Timer0  | TOIE                     | TOIF        | Non                              |
| <b>INT</b>          | Front sur la broche RB0/INT<br>(le front de déclenchement est configuré par le bit INTEDG) | INTE                     | INTF        | Non                              |
| <b>RBI</b>          | Front sur une broche RB4-RB7   | RBIE                     | RBIF        | Non                              |
| <b>ADI</b>          | Fin de conversion A/N  | ADIE                     | ADIF        | Oui                              |
| <b>TMR1I</b>        | Débordement de Timer1  | TMR1E                    | TMR1F       | Oui                              |
| <b>TMR2I</b>        | Timer2 a atteint la valeur maxi  | TMR2E                    | TMR2F       | Oui                              |
| <b>EI</b>           | Fin d'écriture en EEPROM   | EEIE                     | EEIF        | Oui                              |

- ☞ Toutes les interruptions peuvent être validées/interdites à la fois par le bit GIE ;
- ☞ Chaque interruption peut être validée/interdite par son bit de validation individuel ;
- ☞ Un bit drapeau (flag) permet de savoir si l'évènement déclenchant l'interruption arrive.

### L'interruption INT

Cette interruption externe arrive via la broche RB0/INT, sa mise en œuvre nécessite de :

- Mettre à 1 le bit **GIE** pour l'autorisation globale de toutes les interruptions ;
- Mettre à 1 le bit **INTE** afin d'autoriser l'interruption sur RB0/INT ;
- Choisir le front (montant ou descendant) qui déclenche l'interruption par le bit **INTEDG** du registre OPTION\_REG.

Le bit **INTF** (flag) est un indicateur, il est mis à 1 si une interruption arrive sur RB0/INT. Il doit être remis à 0 dans le programme de traitement de l'interruption.

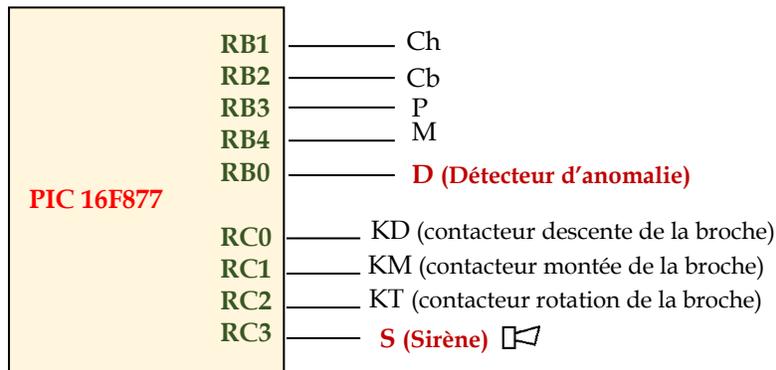
Registre **INTCON**

|     |      |      |      |      |      |      |      |
|-----|------|------|------|------|------|------|------|
| GIE | EEIE | TOIE | INTE | RBIE | TOIF | INTF | RBIF |
|-----|------|------|------|------|------|------|------|

### Exercice : mise en œuvre de l'interruption INT

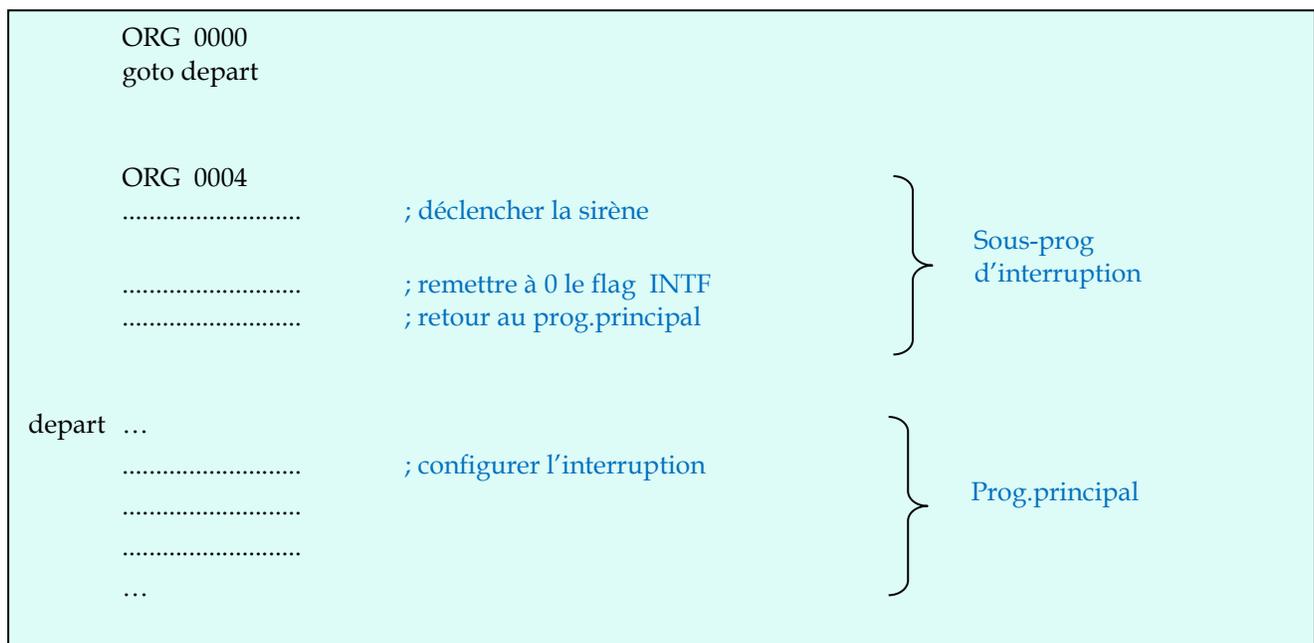
Il s'agit d'alerter les responsables en cas de danger autour de la perceuse de l'exercice traité plus haut : en cas d'anomalie signalée par le détecteur D sur la broche RB0, la sirène S montée sur RC3 entre en action

On va charger le programme principal de l'opération de perçage et la routine d'interruption du traitement de l'anomalie



Configuration de l'interruption INT (sur RB0)

| Registre   | bit        | Action  |
|------------|------------|---|
| INTCON     | GIE = 1    | Autorisation générale de toutes les interruptions |
| INTCON     | INTE = 1   | Autorisation de l'interruption INT (sur RB0)      |
| OPTION_REG | INTEDG = 1 | Choix du front montant sur RB0                    |



**Exercice 1 : examen national 2018, session de rattrapage**

**Tâche 4 : Test du niveau de la charge de la batterie** (voir document DRES 04)

A chaque mise sous tension du chariot, l'utilisateur est informé du niveau de la charge de la batterie par une signalisation sonore. Pour cela, un sous-programme « Test » permet de tester la tension de la batterie  $E_{Bat}$  et d'émettre des bips sonores :

- $E_{Bat} \geq 12 V$  → charge convenable : 1 bip ;
- $11 V \leq E_{Bat} < 12 V$  → charge moyenne : 2 bips ;
- $E_{Bat} < 11 V$  → charge insuffisante (batterie déchargée) : 3 bips.

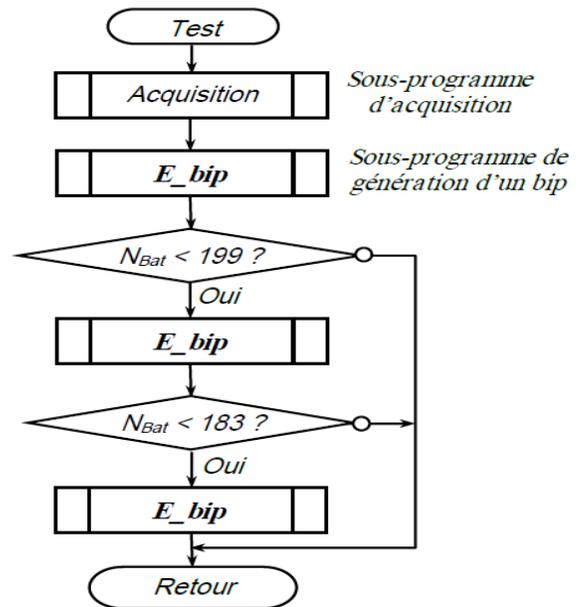
Le principe de ce test consiste à acquérir la tension  $U_{Bat}$  (image de la tension de la batterie  $E_{Bat}$ ) et à la convertir en une valeur numérique  $N_{Bat}$  ; ensuite, on compare cette valeur aux seuils correspondants définis dans le tableau suivant :

|                      |                  |                  |
|----------------------|------------------|------------------|
|                      | $E_{Bat} = 11 V$ | $E_{Bat} = 12 V$ |
| $U_{Bat} (V)$        | 3,59             | 3,92             |
| $N_{Bat}$ en décimal | 183              | 199              |

L'organigramme du sous-programme « Test » est représenté ci-contre :

On note  $adr\_Nbat$  l'adresse de la case mémoire contenant la valeur numérique  $N_{Bat}$ .

**Q39)** En utilisant le jeu d'instructions du microcontrôleur donné sur le document DRES 05, compléter le programme Assembleur correspondant à cet organigramme. 6 pts



**Q39)** Programme assembleur :

| Label | Mnémonique | Opérande   | Commentaire                           |
|-------|------------|------------|---------------------------------------|
| Test  | .....      | .....      | ; Appel au sous-programme Acquisition |
|       | CALL       | E_bip      | ; Appel au sous-programme E_bip       |
|       | .....      | .....      | ; Charger W par la valeur 199         |
|       | SUBWF      | Adr_NBat,W | ; Comparer (adr_NBat) à W             |
|       | .....      | .....      | ; Sauter si N_Bat < 199               |
|       | GOTO       | Fin        | ; Aller à la fin                      |
|       | CALL       | E_bip      | ; Appel au sous-programme E_bip       |
|       | .....      | .....      | ; Charger W par la valeur 183         |
|       | .....      | .....      | ; Comparer (adr_NBat) à W             |
|       | BTFSC      | STATUS,C   | ; Sauter si N_Bat < W                 |
|       | .....      | .....      | ; Aller à la fin                      |
|       | CALL       | E_bip      | ; Appel au sous-programme E_bip       |
| Fin   | RETURN     |            |                                       |

### Exercice 2 : examen national 2019, session de rattrapage

#### Tâche 2 : Programme de régulation du degré d'acidité pH

Le programme de régulation agit selon l'**algorithme** suivant :

- Acquisition du niveau de pH ;
- Affichage du niveau de pH sur l'indicateur (BARGRAPH) ;
- Si le pH mesuré est inférieur à 6,9 ( $N < N1$ ) alors :
  - ✓ Injection pendant 5 minutes du correcteur de pH+ par la mini-pompe doseuse pH+ ;
  - ✓ Arrêt de la mini-pompe pendant 15 minutes, ce qui permet de stabiliser l'eau traitée ;
- Si le pH mesuré est supérieur à 7,7 ( $N > N2$ ) alors :
  - ✓ Injection pendant 5 minutes du correcteur de pH- par la mini-pompe doseuse pH- ;
  - ✓ Arrêt de la mini-pompe pendant 15 minutes, ce qui permet de stabiliser l'eau traitée.

On dispose des sous programmes suivants :

- **Acquisition\_pH** : qui permet de convertir le signal  $U_3$  image du pH en un nombre N sur 8 bits, stocké dans la case mémoire appelée **Val\_pH** ;
- **Affichage\_pH** : qui permet d'afficher le pH mesuré sur le BARGRAPH à diodes LED connecté sur le port B ;
- **TEMPO\_T1** : est un sous-programme de temporisation  $T1 = 5$  min ;
- **TEMPO\_T2** : est un sous-programme de temporisation  $T2 = 15$  min.

**Q.43-** En vous aidant de l'algorithme ci-dessus, compléter l'**organigramme** correspondant.

[4,5 pts]

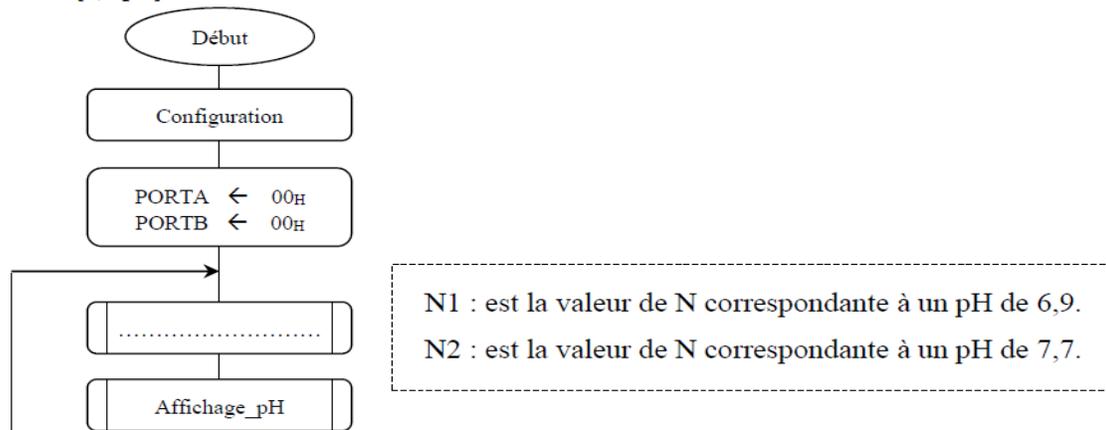
La configuration du PIC16F876 est comme suit :

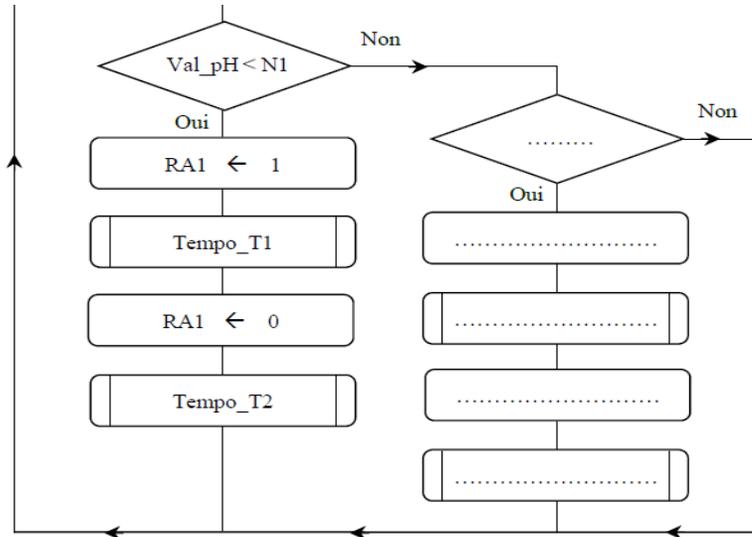
- RA2 : Sortie logique pour la commande de la mini-pompe doseuse pH- ;
- RA1 : Sortie logique pour la commande de la mini-pompe doseuse pH+ ;
- RA0 : Entrée analogique pour l'acquisition de la tension  $U_3$  image du pH de l'eau de la piscine ;
- RB0...RB7 : Sorties logiques pour la commande du BARGRAPH.

**Q.44-** En vous aidant du jeu d'instructions fourni en **DRES 04** page 14, compléter le programme assembleur de régulation du pH. (Les bits non utilisés du registre **TRISA** sont à 0).

[5,5 pts]

**Question.43:** [4,5 pts]



**Question.44:** [5,5 pts]

```

BCF          STATUS, 6
.....      ; Accès à la banque 1
.....
.....      ; Configuration PORTA
.....      ; Configuration PORTB

MOVLW       0X0E
MOVWF       ADCON1      ; Configuration ADCON1
BCF         STATUS, 5   ; Accès à la banque 0
MOVLW       0X81
MOVWF       ADCON0      ; Configuration ADCON0
CLRF        PORTB
CLRF        PORTA      ; Initialisation des sorties du système
LAB1 CALL    Acquisition_pH ; Appel du sous-programme Acquisition_pH
CALL        Affichage_pH ; Appel du sous-programme Affichage_pH
.....
.....
.....
GOTO        LAB2      ; Si non sauter à LAB2
.....
.....
.....
.....

LAB2 GOTO    LAB1      ; Reprendre
MOVF       Val_pH, W   ; Lecture du résultat de la conversion
SUBLW      N2          ; W = N2 - W
BTFSC     STATUS, C   ; W > N2?
GOTO      LAB1        ; Si non sauter à LAB1
BSF       PORTA, 2    ; Injection du pH-
CALL      TEMPO_T1    ; Appel du sous-programme de temporisation 5 min
BCF       PORTA, 2    ; Arrêt d'injection du pH-
CALL      TEMPO_T2    ; Appel du sous-programme de temporisation 15 min
GOTO      LAB1        ; Reprendre
  
```

### Exercice 3 : examen national 2021, session normale

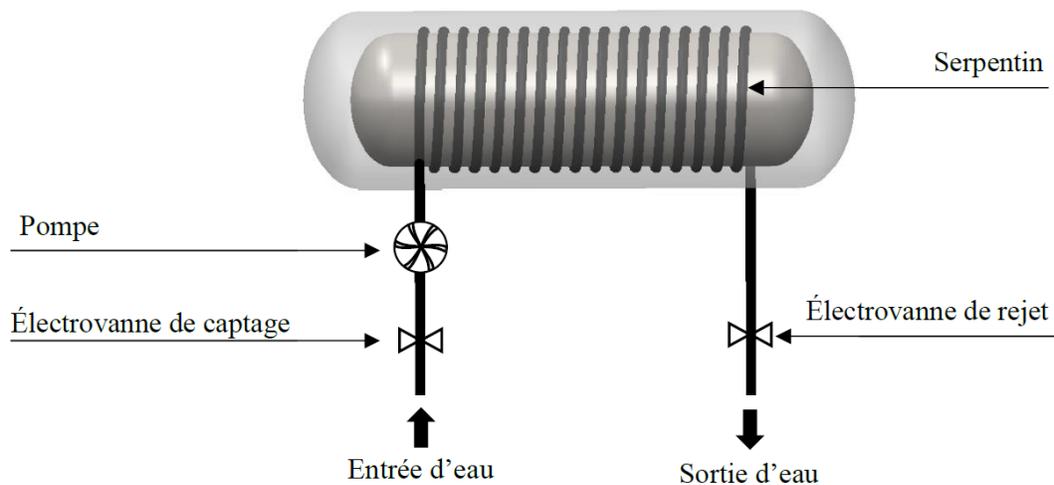
#### **Tâche 2 : Commande du système de refroidissement à l'eau de mer**

Les constituants électriques de l'hydrolienne, en particulier les convertisseurs électroniques de puissance, sont confinés à l'intérieur du convertisseur AC/AC (l'enceinte posée au fond marin et abritant l'ensemble du matériel électrique de l'hydrolienne).

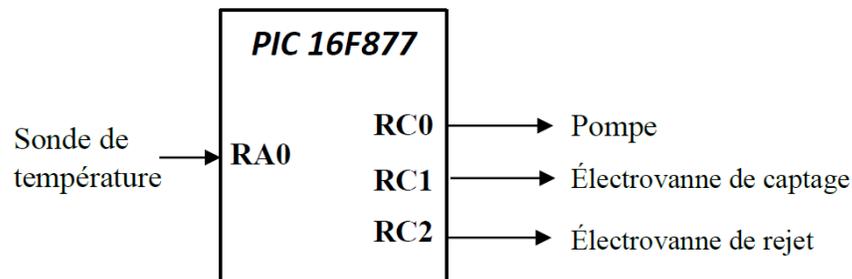
De ce fait, les composants électroniques peuvent être portés à des températures excessives. Cela nécessite de prévoir un moyen de refroidissement afin de protéger l'électronique embarquée.

On envisage de maintenir la température à l'intérieur du convertisseur AC/AC entre 20 °C et 30 °C par un système de refroidissement à l'eau de mer.

Moyennant deux électrovannes et une pompe à eau de mer, le système fait circuler l'eau dans un serpentin (tube métallique enroulé en spirale) inséré dans la double coque du convertisseur AC/AC. C'est un refroidissement en circuit ouvert ; en effet, après circulation dans le serpentin, l'eau est rejetée dans la mer. Une sonde mesure la température à l'intérieur du convertisseur AC/AC.



La commande de ce système de refroidissement est réalisée par un microcontrôleur PIC 16F877 selon l'architecture suivante :



### Programme de commande de ce système de refroidissement

#### Éléments du programme

- **Lecture\_T** : sous-programme chargé de lire, sur la ligne **RA0**, l'information issue de la sonde de température et de ranger le code numérique correspondant dans le registre **NT**.
- **NT** : registre qui contient le code numérique de la température mesurée par la sonde.
- **NT<sub>min</sub>** et **NT<sub>max</sub>** : sont les codes numériques qui correspondent respectivement à **20 °C** et **30 °C**.

#### Séquences du programme

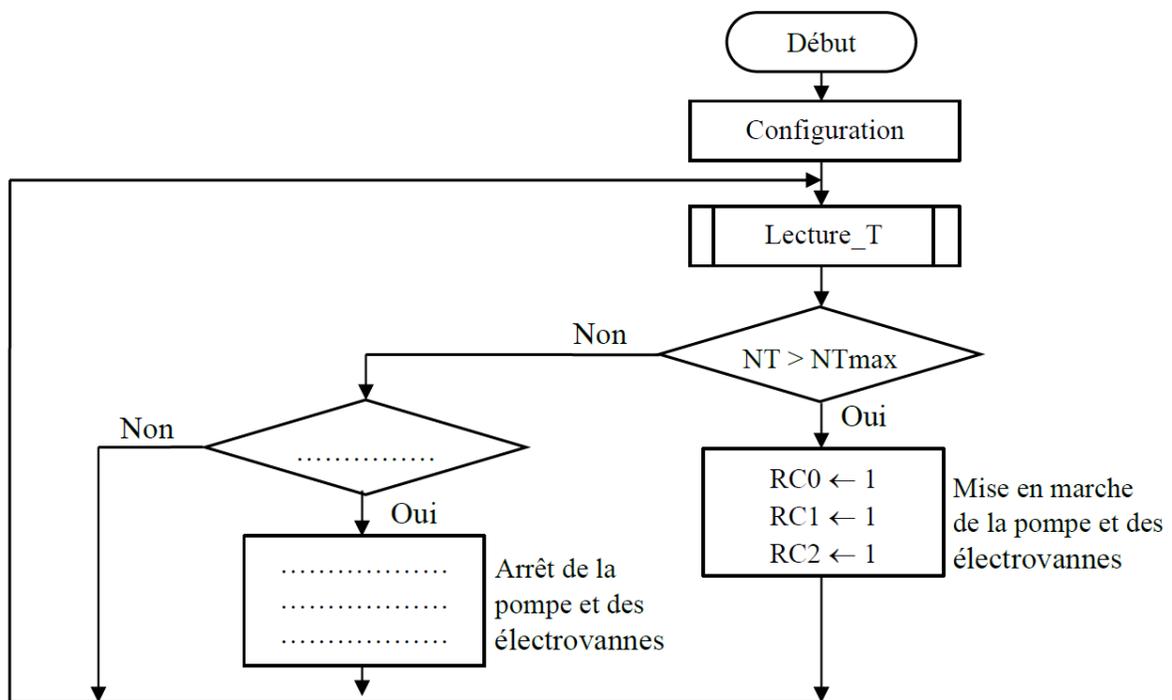
- Appel du sous-programme **Lecture\_T** ;
- Evaluation de la température lue :
  - Si  $NT > NT_{max}$ , on met en marche la pompe et les deux électrovannes ;
  - Si  $NT < NT_{min}$ , on arrête la pompe et les deux électrovannes ;
- Reprendre une nouvelle lecture de la température.

**Question : 43.** Compléter l'organigramme traduisant les séquences ci-dessus.

2 pts

**Question : 44.** En vous aidant du jeu d'instructions du **DRES 04**, compléter le programme assembleur correspondant.

6 pts



| Instruction   |  | Commentaire   |                           |
|---|--|---|---------------------------|
| <b>BCF STATUS, RP1</b><br>.....                                   |  | ; Activer la page 1   |                           |
| .....   |  | ; Configurer PORTC en sortie  |                           |
| <b>BSF TRISA, 0</b>   |  | ; Configurer la ligne RA0 en entrée   |                           |
| <b>BCF STATUS, RP1</b><br><b>BCF STATUS, RP0</b>                  |  | ; Activer la page 0   |                           |
| <b>repete</b><br>.....  |  | ; appel du sous-programme "Lecture_T"   |                           |
| <b>MOVLW NTmax</b><br><b>SUBWF NT, w</b><br>.....                 |  | ; $W \leftarrow NTmax$<br>; $W \leftarrow NT - NTmax$<br>; tester si le bit C = 1 | tester si<br>$NT > NTmax$ |
| <b>GOTO saut</b>  |  | ; si non, se brancher à l'étiquette "saut"  |                           |
| <b>BSF PORTC, 0</b><br><b>BSF PORTC, 1</b><br><b>BSF PORTC, 2</b> |  | ; si oui, mettre en marche la pompe et les deux électrovannes                     |                           |
| <b>GOTO repete</b>  |  | ; reprendre   |                           |
| <b>saut</b><br>.....<br>.....<br>.....                            |  | ; $W \leftarrow NTmin$<br>; $W \leftarrow NT - NTmin$<br>; tester si le bit C = 0 | tester si<br>$NT < NTmin$ |
| <b>GOTO repete</b>  |  | ; si non, reprendre   |                           |
| <b>BCF PORTC, 0</b><br><b>BCF PORTC, 1</b><br>.....               |  | ; si oui, arrêter la pompe et les deux électrovannes                              |                           |
| <b>GOTO repete</b>  |  | ; reprendre   |                           |

**Exercice 4 : examen national 2022, session normale****Tâche 2 : Affichage de la masse à déplacer**

Le monte-charge étudié est équipé d'une carte d'acquisition et d'affichage de la masse  $m_C$  à déplacer (voir document ressources **DRES 04**). Cette carte est à base du microcontrôleur **PIC 16 F 876**.

L'affichage se fait en décimal sur 3 afficheurs à 7 segments, équipés chacun d'un décodeur BCD/7 segments et d'un Latch 4 bits (verrou à 4 bascules D).

On dispose des sous programmes suivants :

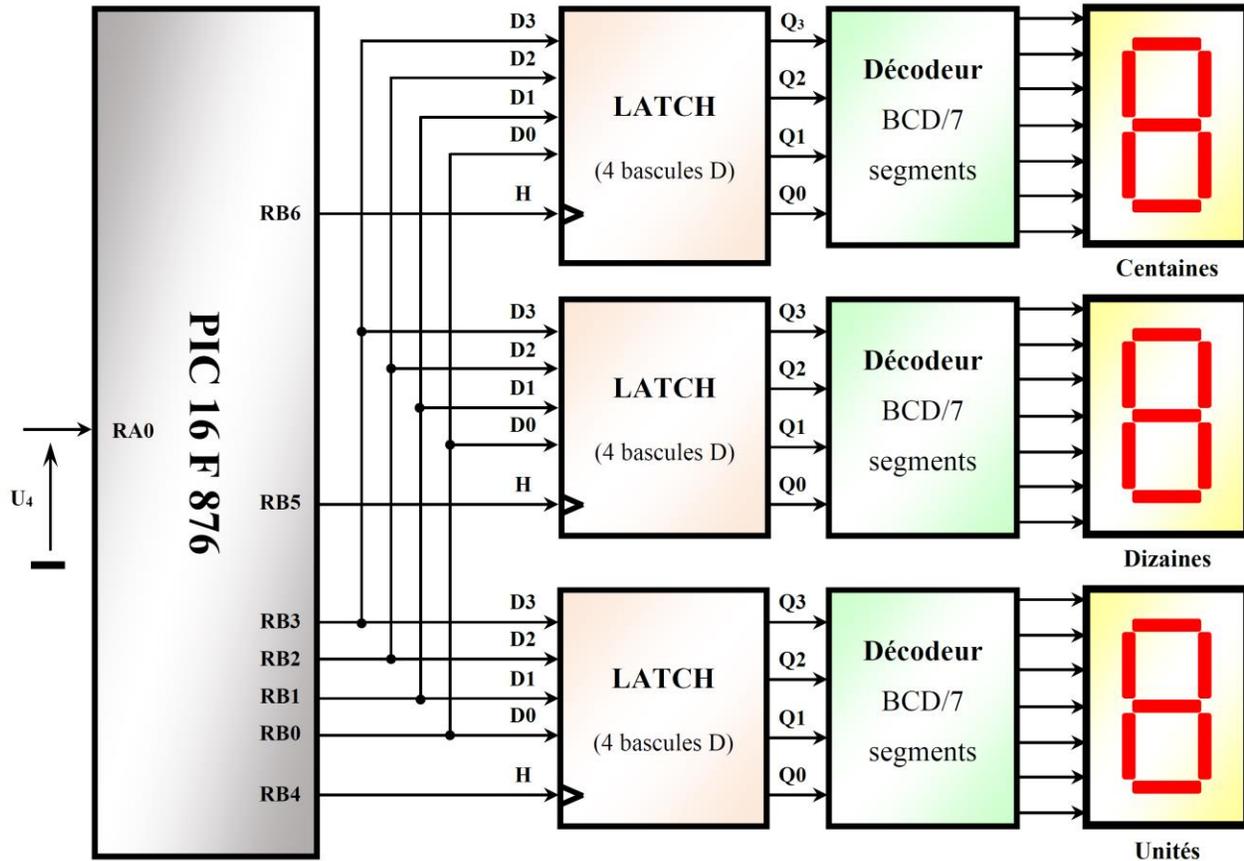
- **Acquisition** : Sous-programme qui convertit le signal  $U_4$  image de la masse  $m_C$  à déplacer en un nombre  $N$  sur **10 bits**, stocké dans les registres internes du CAN (**ADRESH : ADRESL**).
- **Conv\_Kg** : Sous-programme qui convertit la valeur numérique  $N$  résultat du convertisseur CAN (**ADRESH : ADRESL**) en un nombre  $N'$  exprimé en **Kg**, stocké dans deux cases mémoires appelées **Val\_mc\_H** et **Val\_mc\_L**.
- **Conv\_BCD** : Sous-programme qui convertit le nombre  $N'$  en décimal (code BCD), stocké dans les cases mémoires appelées **Unite\_mc** (pour les unités), **Dizaine\_mc** (pour les dizaines) et **Centaine\_mc** (pour les centaines) ;

**Q.47** - En vous aidant du document ressources **DRES 04**, compléter l'**organigramme** de l'acquisition et de l'affichage de la masse  $m_C$  à déplacer. **4 pts**

**Q.48** - En vous aidant du jeu d'instructions fourni en document ressources **DRES 05**, compléter le **programme assembleur** correspondant. **5 pts**

## Carte d'acquisition et d'affichage de la masse à déplacer

DRES 04



## Principe d'affichage de la masse à déplacer

Si  $N' = 2DE_H$  alors  $N' = 734$  en décimal donc  $N' = 0111.0011.0100$  en BCD

Donc, après appel du sous-programme "Conv\_BCD" :

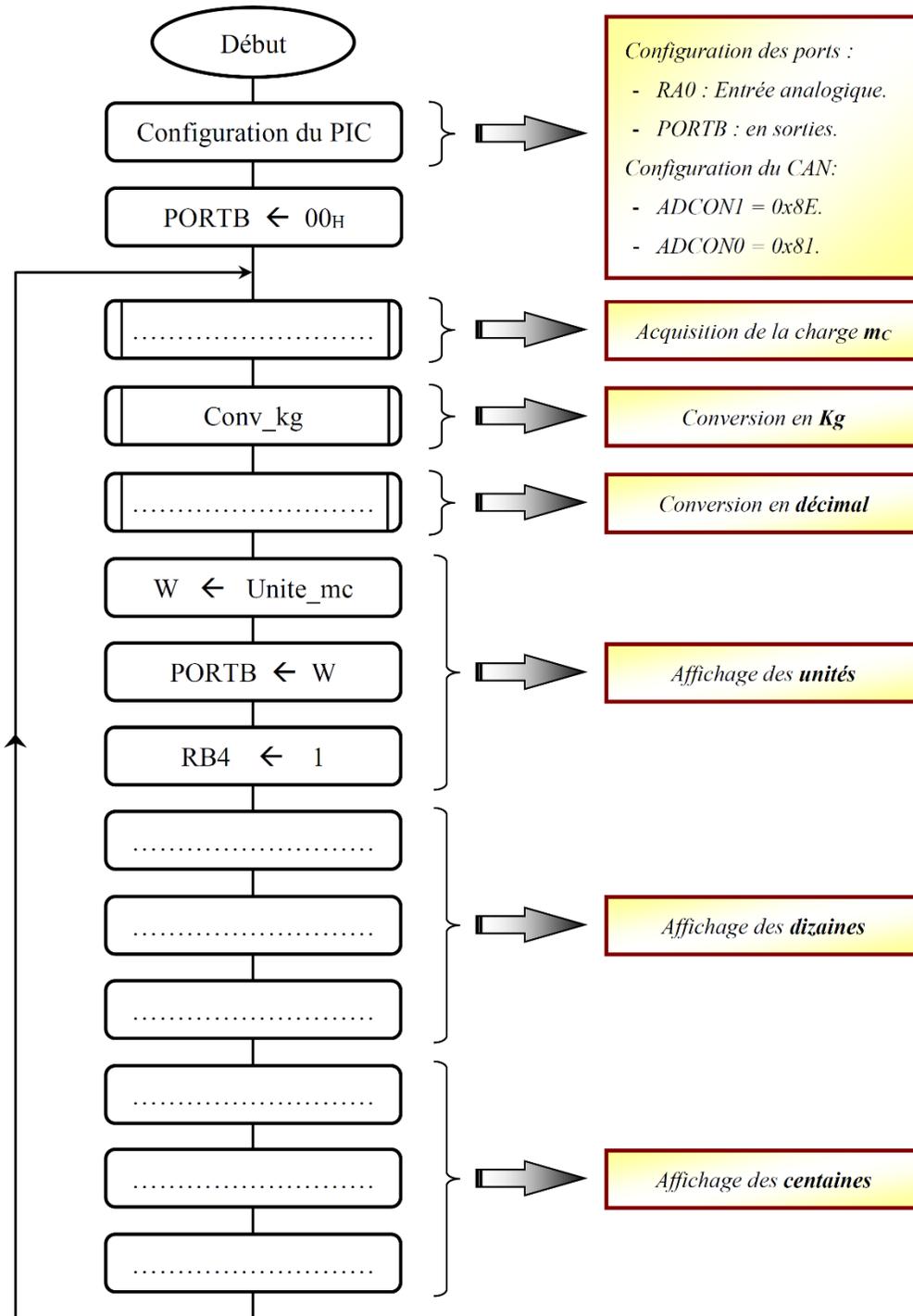
- Unité\_mc = 0 0 0 0 . 0 1 0 0 (4)
- Dizaine\_mc = 0 0 0 0 . 0 0 1 1 (3)
- Centaines\_mc = 0 0 0 0 . 0 1 1 1 (7)

Pour afficher la masse à déplacer  $m_C$  en décimal sur les **trois** afficheurs ( $N' = 2DE_H = 734$  en décimal), on effectue les opérations suivantes :

- Acquisition de la masse à déplacer par appel du sous-programme "Acquisition" ;
  - Convertir le nombre  $N$  résultat de la conversion analogique numérique en un nombre  $N'$ , exprimé en **kg** par appel du sous-programme "Conv\_Kg" ;
  - Convertir  $N'$  en BCD par appel du sous-programme "Conv\_BCD" ;
  - Écrire la valeur des unités '4' sur les entrées  $D_3..D_0$  de l'afficheur des unités avec  $H = 0$  ;
  - Mettre  $H$  à 1 pour mémoriser la valeur des unités par les sorties  $Q_3..Q_0$  (créer un front montant sur l'entrée d'horloge des unités) ;
  - Écrire la valeur des dizaines '3' sur les entrées  $D_3..D_0$  de l'afficheur des dizaines avec  $H = 0$  ;
  - Mettre  $H$  à 1 pour mémoriser la valeur des dizaines par les sorties  $Q_3..Q_0$  (créer un front montant sur l'entrée d'horloge des dizaines) ;
  - Écrire la valeur des centaines '7' sur les entrées  $D_3..D_0$  de l'afficheur des centaines avec  $H = 0$  ;
- Mettre  $H$  à 1 pour mémoriser la valeur des centaines par les sorties  $Q_3..Q_0$  (créer un front montant sur l'entrée d'horloge des centaines).

Q.47 -

**DREP 09**



Q.48 -

**DREP 10**

```

BCF      STATUS, 6      ;
.....      ; accès à la BANK 1
.....      ; PORTB en sortie
.....      ; Mot de commande du registre TRISA
MOVWF   TRISA          ; RA0 en entrée
MOVLW   0x8E          ; Mot de commande du registre ADCON1
MOVWF   ADCON1        ; Configuration du CAN interne
BCF     STATUS, 5      ; Retour en banque mémoire 0
MOVLW   0x81          ; Mot de commande du registre ADCON0
MOVWF   ADCON0        ; Configuration du CAN interne
CLRF    PORTB         ; Initialisation des sorties
Loop    CALL    Acquisition ; appel du sous-programme "Acquisition"
.....
CALL    Conv_BCD      ; appel du sous-programme "Conv_BCD"
.....
.....
.....
.....
.....
MOVF    Centaine_mc, W ; Lecture de la valeur des centaines
MOVWF   PORTB        ; Ecriture des centaines dans le PORTB
BSF     PORTB, 6      ; Affichage des centaines
GOTO   Loop          ;

```

**Exercice 5 : examen national 2023, session de rattrapage****Tâche 2 : programmation du cycle de nettoyage des diffuseurs fines bulles**

Le colmatage (bouchage des orifices des membranes des diffuseurs causé par les dépôts) est le problème majeur observé dans le système d'insufflation par diffuseurs fines bulles.

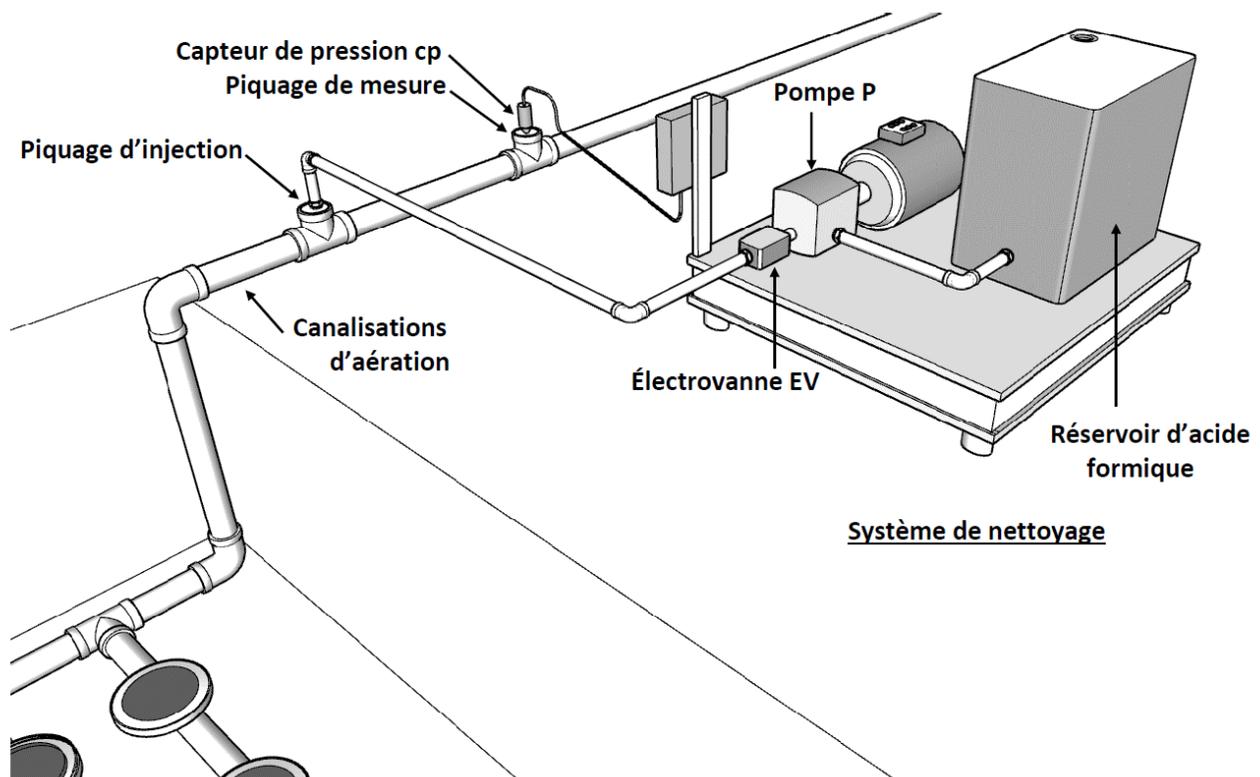
Il est donc fortement recommandé de procéder de façon régulière à un nettoyage préventif des canalisations d'aération

Le nettoyage par injection de l'acide formique est le remède le plus couramment utilisé.

**Système de nettoyage**

Le système de nettoyage (figure ci-dessous) est équipé d'une pompe **P** et d'une électrovanne d'isolement **EV** ; la pompe puise l'acide formique dans un réservoir. Le produit est pulvérisé et injecté dans les canalisations d'aérations par le biais d'un piquage.

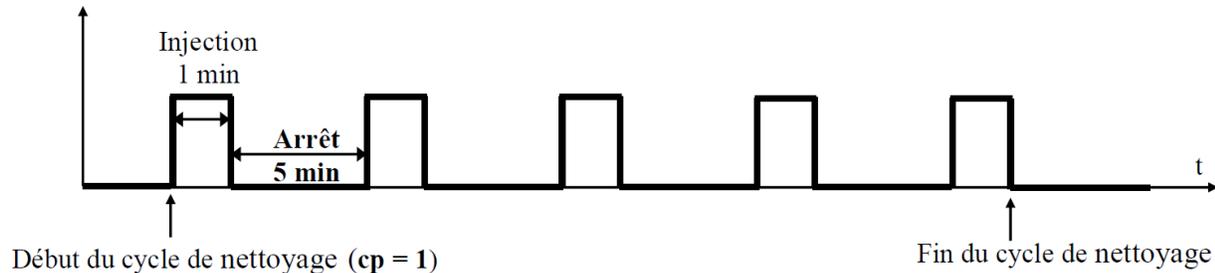
Le cycle de nettoyage se déclenche lorsque la perte de charge (chute de pression), détectée par un capteur de pression **cp**, dépasse un certain seuil



Le dispositif de nettoyage est géré par microcontrôleur (PIC 16F877). Le cycle de fonctionnement de ce dispositif est réalisé de la manière suivante :

Dès que le capteur « cp » détecte une perte de charge de **30 mbar** ( $cp = 1$ ), on réalise **5 injections** d'acide formique d'une minute chacune (**1 min**), avec un arrêt de **5 minutes** entre deux injections successives.

Le chronogramme suivant illustre les étapes du cycle de nettoyage :



L'équipement électrique du nettoyeur est connecté au microcontrôleur comme suit :

|                               |              |
|-------------------------------|--------------|
| Capteur de pression <b>cp</b> | → <b>RC1</b> |
| Pompe <b>P</b>                | → <b>RC2</b> |
| Électrovanne <b>EV</b>        | → <b>RC3</b> |

**Q.40)** Quelle est la durée totale  $T_n$  d'un cycle de nettoyage (en *min*) ?

**1 pt**

**Q.41)** Compléter l'organigramme de ce cycle (sont déclarés : une variable  $N$  de comptage ainsi que deux sous-programmes  $T\_1min$  et  $T\_5min$  qui réalisent respectivement des temporisations de **1 min** et de **5 min**) ;

**2 pts**

**Q.42)** Compléter le programme assembleur correspondant (le jeu d'instructions du microcontrôleur est fourni en **DRES 02**).

**3 pts**

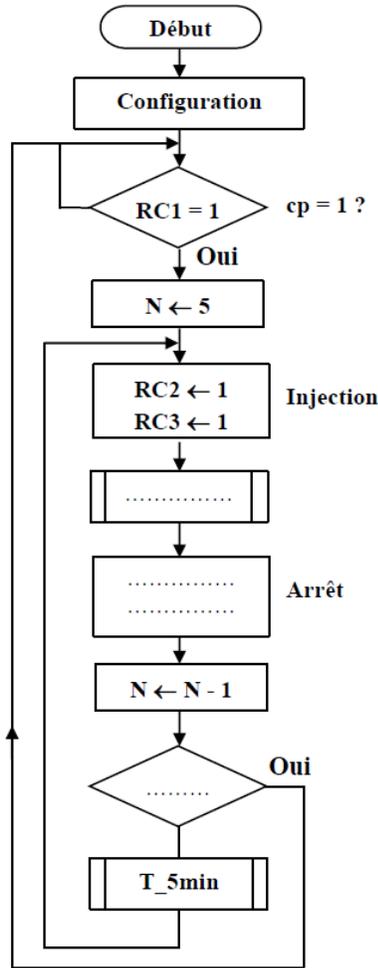
Les deux sous-programmes de  $T\_1min$  et  $T\_5min$  font appel à une routine qui se répète plusieurs fois pour réaliser la temporisation désirée. Le code assembleur de la routine est le suivant :

| <u>Routine de temporisation</u> |              |
|---------------------------------|--------------|
|                                 | MOVLW d'233' |
|                                 | MOVWF N1     |
| b4                              | DECFSZ N1, f |
|                                 | GOTO b4      |

**N1** est une variable de comptage



Q.41)



Q.42)

|    |                                    |  |
|----|------------------------------------|--|
| b1 | .....<br>GOTO    b1                | Tester si RC1 = 1  |
|    | .....<br>.....                     | N ← 5  |
| b2 | BSF    PORTC, 2<br>BSF    PORTC, 3 | Injection  |
|    | CALL    T_1min                     | Temporisation de 1 min   |
|    | BCF    PORTC, 2<br>BCF    PORTC, 3 | Arrêt de l'injection   |
|    | DECFSZ    N, f<br>GOTO    b3       | Décrémenter N et tester si N = 0<br>Si non, se brancher à "b3" |
|    | .....                              | Si oui, fin de cycle   |
| b3 | .....<br>.....                     |  |

### Exercice 6 : examen national 2024, session normale

#### Tâche 6 : Calcul de la vitesse de consigne et détermination du sens de déplacement

Pour calculer la vitesse de consigne et déterminer le sens de déplacement du crochet (montée ou descente), le PIC agit selon l'algorithme suivant :

- Acquérir la position  $\alpha$  du manche de la manette de commande par un sous-programme appelé 'Acquisition' qui convertit le signal  $U_s$  image de  $\alpha$  en un nombre  $N$  sur 9 bits, stocké dans les cases mémoires appelées  $N_H$  ( $N_{H0}$  bit de poids fort de  $N$ ) et  $N_L$  (8 bits de poids faible de  $N$ ) ;
- Déterminer à partir de  $N_{H0}$  (bit 0 de  $N_H$ ) et de  $N_L$  :
  - Le code de la vitesse consigne  $N_C$  (sur 8 bits) :  $N_C = N_L$  si  $\alpha > 0$  et  $N_C = \bar{N}_L$  si  $\alpha < 0$  ;
  - Le sens de déplacement du crochet (montée ou descente) en positionnant les bits RA1 et RA2.
- Mettre  $N_C$  à 0 si  $N_C < N_{Cm}$  ( $N_{Cm}$  : code de la vitesse de déplacement minimal du crochet) ;
- Transférer le code  $N_C$  au convertisseur numérique analogique via le **PORTB**, pour restituer le signal  $U_c$  de commande du variateur de vitesse du moteur **M1**.

#### Exemple :

|   | $N_H$     | $N_L$     |   |
|---|-----------|-----------|---|
| $\alpha_{\max} = +25^\circ$<br>( $\alpha > 0$ ) | 0000 0001 | 1111 1111 | Vitesse maximale de montée : $N_C = N_L = 255$<br>Bit 0 de $N_H = 1$ : ordre de monter le crochet.            |
| $\alpha_{\min} = -25^\circ$<br>( $\alpha < 0$ ) | 0000 0000 | 0000 0000 | Vitesse maximale de descente : $N_C = \bar{N}_L = 255$<br>Bit 0 de $N_H = 0$ : ordre de descendre le crochet. |

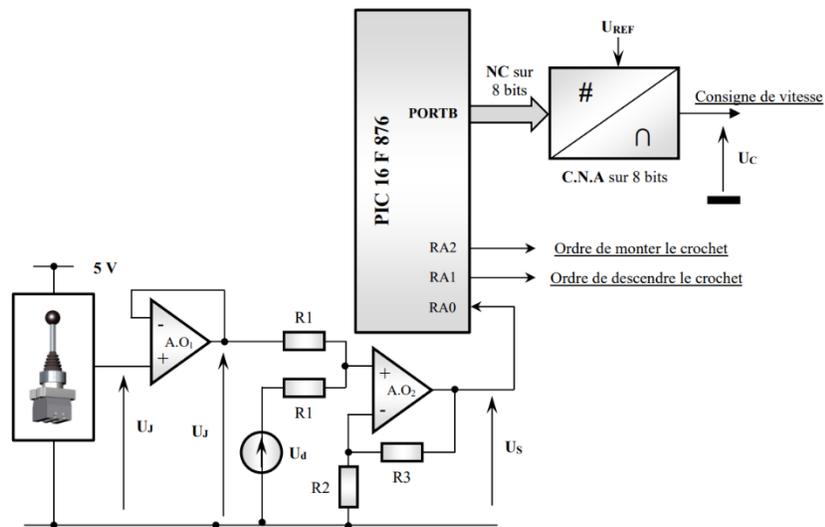
En vous aidant de l'algorithme ci-dessus et du jeu d'instructions fourni en **DRES 04**, compléter :

**Q.48-** L'organigramme correspondant.

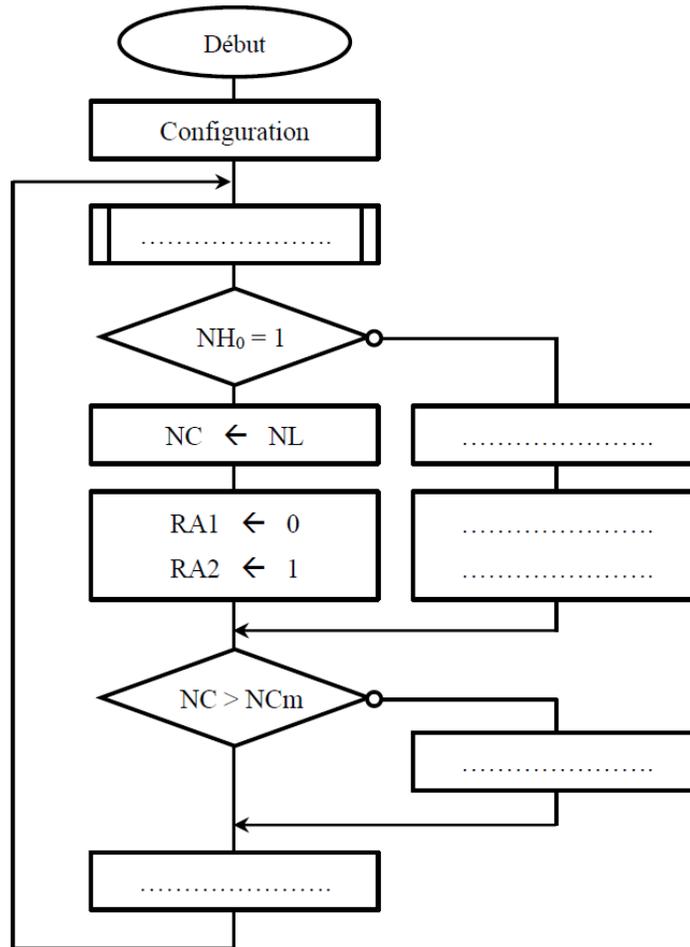
**3 pts**

**Q.49-** Le programme de configuration du microcontrôleur **PIC 16F876**. (Mettre les bits non utilisés du registre **TRISA** à 0).

**3 pts**



Q.48- [3 pts]



Q.49- [3 pts]

```

..... ;
..... ; Accès à la BANK 1
..... ;
..... ; RA0 entrée, RA1 et RA2 des sorties
..... ; PORTB en sortie
MOVLW    0x8E ;
MOVWF    ADCON1 ; Configuration du CAN interne
BCF      STATUS, 6 ;
..... ; Accès à la BANK 0
MOVLW    0x81 ;
MOVWF    ADCON0 ; Configuration du CAN interne
  
```

### Exercice 7 : examen national 2024, session de rattrapage

#### Tâche 5 : Programme assembleur

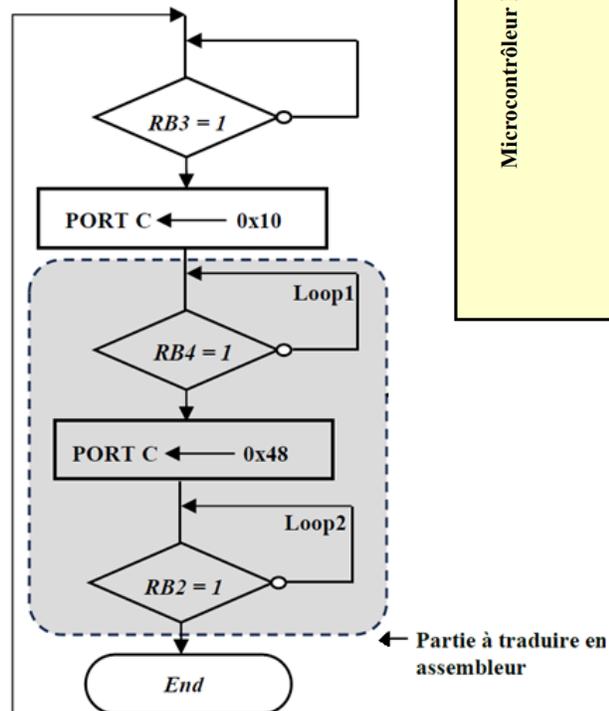
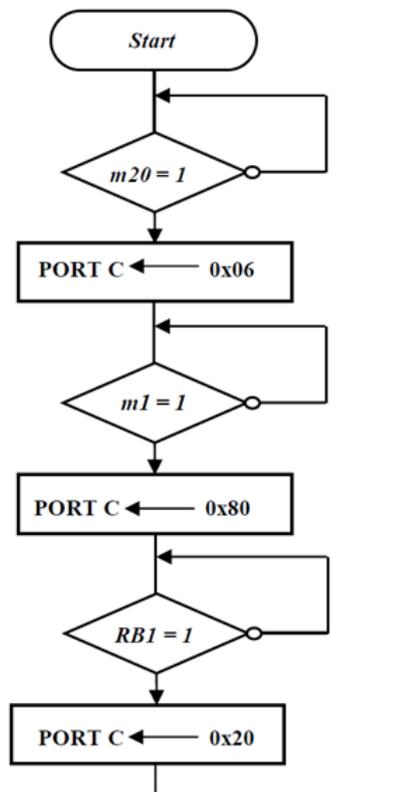
À partir :

- Du schéma de câblage des entrées-sorties du **DRES 03** ;
- De l'extrait de la documentation technique du microcontrôleur donné en **DRES 04**,

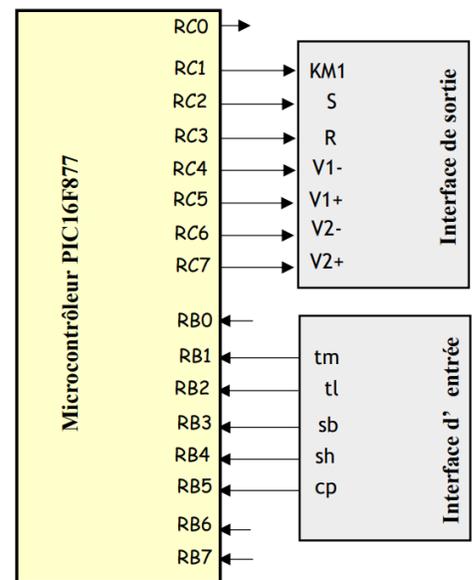
**Q47.** Compléter les parties du programme assembleur permettant de :

**6,5 pts**

1. Configurer les ports **B** et **C** du microcontrôleur ;
2. Réaliser la partie encadrée de l'organigramme de l'expansion de la macro-étape **M20** donné ci-dessous.



#### Schéma de câblage des entrées-sorties



**Q47. Programme Assembleur**

| Ligne  | Label        | Mnémonique   | Opérande           |
|--|--------------|--------------|--------------------|
| <b>Configuration des ports B et C</b>                    |              |              |                    |
| 1  |              | <b>BSF</b>   | <b>STATUS, RP0</b> |
| 2  |              | .....        | .....              |
| 3  |              | .....        | .....              |
| 4  |              | .....        | .....              |
| 5  |              | .....        | .....              |
| 6  |              | <b>BCF</b>   | <b>STATUS, RP0</b> |
| <b>Programme de la partie encadrée de l'organigramme</b> |              |              |                    |
| 7  | <b>Loop1</b> | <b>BTFSS</b> | <b>PORTB, RB4</b>  |
| 8  |              | .....        | .....              |
| 9  |              | .....        | .....              |
| 10   |              | .....        | .....              |
| 11   | <b>Loop2</b> | .....        | .....              |
| 12   |              | .....        | .....              |

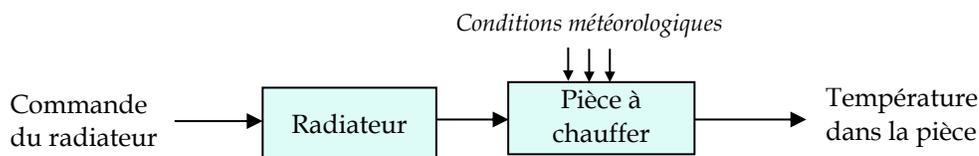
## 8

**SYSTEMES ASSERVIS****Généralités****1. Commande en boucle ouverte**

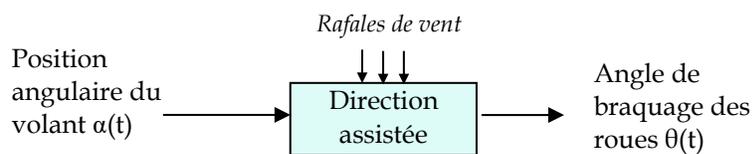
Un système est en boucle ouverte lorsque la commande est élaborée indépendamment de la sortie. Lorsqu'une perturbation intervient sur le système, la sortie s'écarte de la consigne (sortie souhaitée) : la précision de ce système est faible et il est moins fiable.

**Exemple 1 : commande de la température d'une salle**

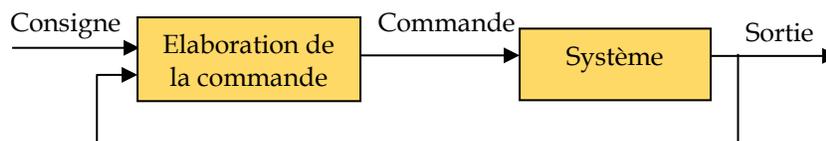
Suite à un changement de la température ambiante, la température atteinte dans la salle sera différente de la consigne.

**Exemple 2 : direction assistée d'un véhicule automobile**

Lors de rafales de vent, l'angle de braquage souhaité ne sera pas obtenu.

**2. Commande en boucle fermée**

En boucle fermée, la sortie est comparée en permanence à la consigne puis corrigée.



### 3. Systeme bouclé

Un système bouclé a donc un dispositif de retour permettant de compenser le manque de fidélité d'un système physique. Un système bouclé peut fonctionner en régulation ou en asservissement :

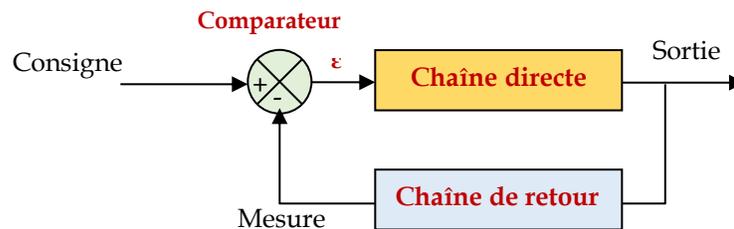
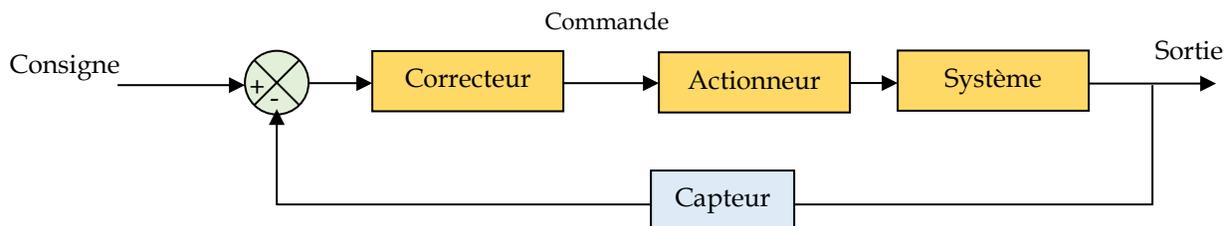
- On parle de régulation lorsque la **consigne est constante** et que le système doit maintenir une sortie constante quelques soient les perturbations.

Exemples : régulation de vitesse d'un véhicule, de température d'un local.

- On parle d'asservissement lorsque la **consigne varie** avec le temps. Le système doit ajuster en permanence la sortie au signal d'entrée.

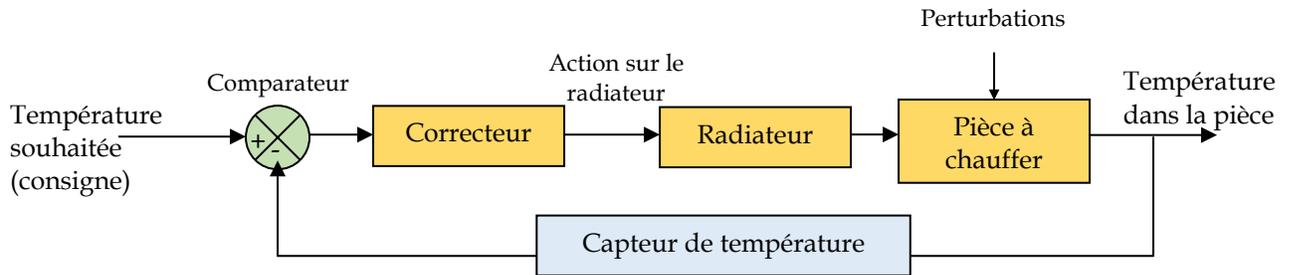
Exemples : asservissement de position d'un radar de poursuite, suivi de trajectoire d'un missile.

Le schéma bloc d'un système asservi est le suivant :

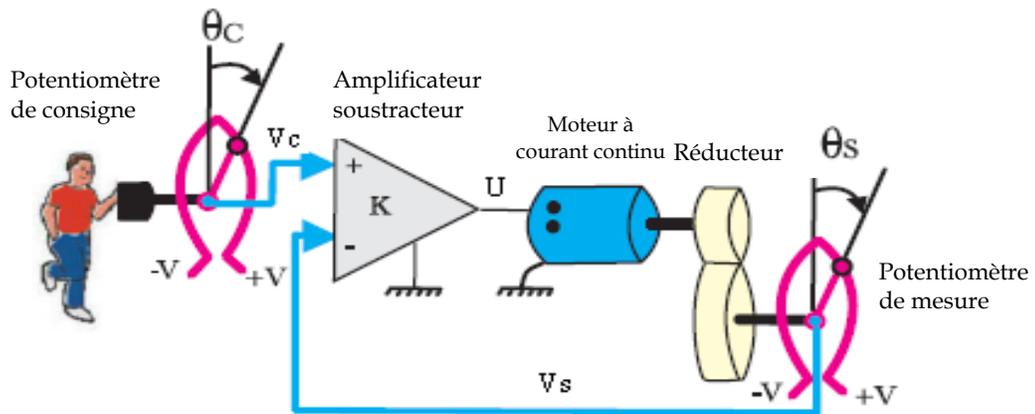


- Le comparateur calcule l'écart entre la consigne et la sortie ;
- À partir de l'écart  $\epsilon$  constaté, le correcteur élabore un signal de commande ;
- L'écart  $\epsilon$  (ou erreur) caractérise la qualité de l'asservissement. Le but de l'asservissement est d'annuler en permanence cet écart, de manière à ce que la sortie suive l'entrée.

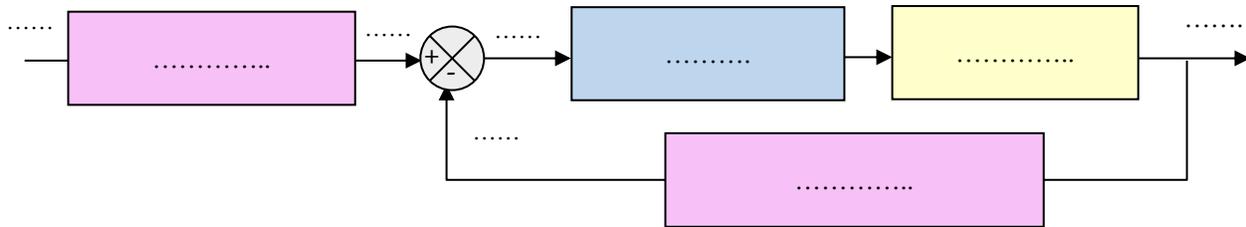
**Exemple : régulation de température d'une pièce**



**Exercice : représentation d'une régulation de position par un schéma bloc**



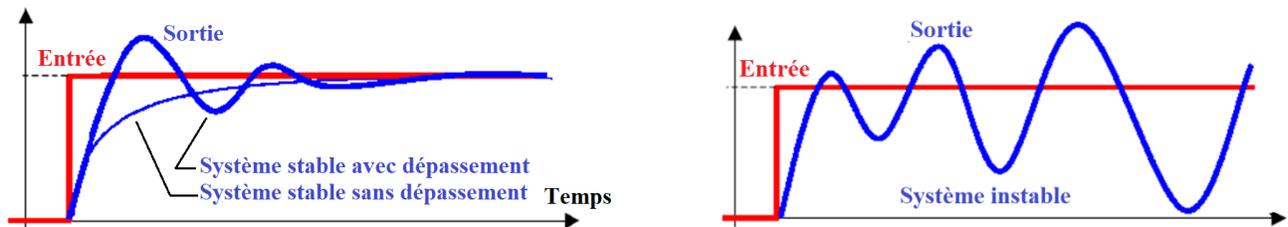
Compléter la représentation en blocs de cette régulation.



#### 4. Performances d'un système asservi

Le cahier des charges d'un système asservi impose généralement un certain nombre de contraintes sur le comportement du système (pour le passage d'une position à une autre sur un radar par exemple). Ces contraintes portent sur : la *stabilité*, la *précision* et la *rapidité*.

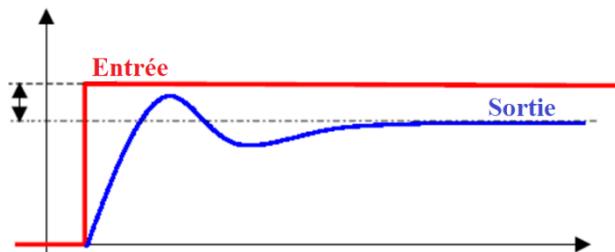
##### ⇒ Stabilité



Un système est stable si pour une entrée constante  $e(t)$ , la sortie du système  $s(t)$  tend vers une constante. Un système asservi instable se traduit par des variations importantes de la sortie qui peuvent causer la dégradation.

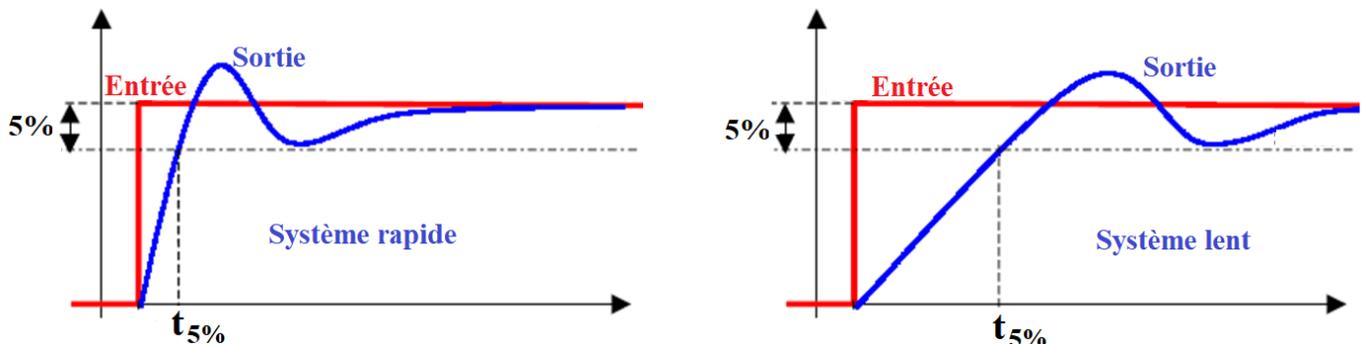
##### ⇒ Précision

C'est la différence entre la sortie et l'entrée en régime permanent.



##### ⇒ Rapidité

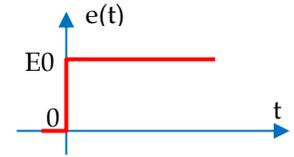
La rapidité se traduit par le temps mis par le système pour que la sortie atteigne la valeur finale. On définit, pour caractériser la rapidité, le temps de réponse à 5%.



## Systèmes linéaires

- La mise en équations d'un système  $s(t) = f(e)$  aboutit souvent à une équation différentielle du premier ordre ou du deuxième ordre ;
- Analyser les performances d'un système c'est étudier son comportement vis-à-vis des signaux d'entrée particuliers. Une entrée typique est l'échelon, la réponse de  $s(t)$  à un échelon s'appelle une réponse indicielle.

Entrée échelon



### 1. Système du premier ordre

Un système physique d'entrée  $e(t)$  et de sortie  $s(t)$  est du premier ordre s'il est régi par une équation différentielle du premier ordre du type :

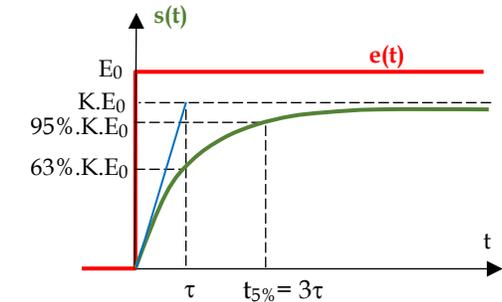
$$\tau \cdot \frac{ds(t)}{dt} + s(t) = K \cdot e(t)$$

$k$  : gain statique du système  
 $\tau$  : constante de temps.

Réponse indicielle : l'entrée est un échelon d'amplitude  $E_0$  pour  $t > 0$

L'équation devient :  $\tau \cdot \frac{ds(t)}{dt} + s(t) = K \cdot E_0$

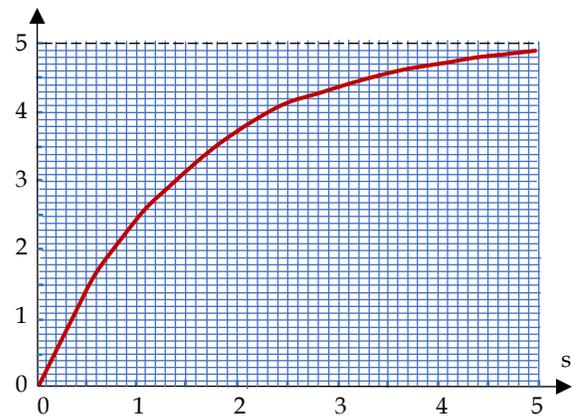
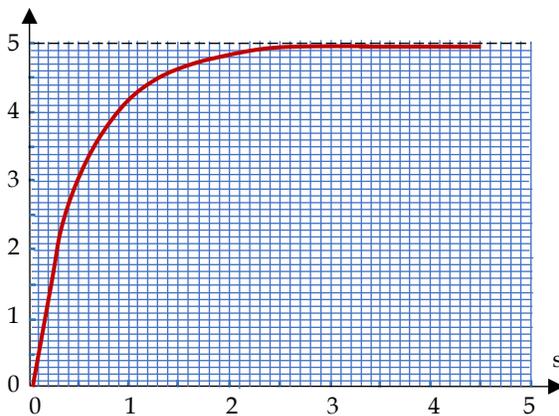
Ce qui donne :  $s(t) = K \cdot E_0 \cdot (1 - e^{-\frac{t}{\tau}})$



- Le comportement du système est non oscillant (stable), il tend vers la valeur  $K \cdot E_0$  sans jamais la dépasser. La propriété de non dépassement est très recherchée dans certains asservissements où le dépassement est interdit ;
- Le temps de réponse à 5% est tel que  $s(t) = 0,95 \cdot K \cdot E_0 \Rightarrow s(t) = K \cdot E_0 \cdot (1 - e^{-\frac{t}{\tau}}) = 0,95 \cdot K \cdot E_0 \Rightarrow t(5\%) = 3\tau$  ;
- L'erreur statique est  $\varepsilon = 1 - k$  ;  
 En effet,  $\varepsilon = (e(t) - s(t))/e(t)$  en régime statique ; soit  $\varepsilon = (E_0 - kE_0)/E_0 = 1 - k$  en % ;
- La **tangente à l'origine** coupe l'asymptote horizontale  $y = K \cdot E_0$  à  $t = \tau$ . **À  $t = \tau$ ,  $s(t) = 0,63 \cdot K \cdot E_0$ .**

### Exercice

Déterminer le temps de réponse à 5% et la constante de temps de ces systèmes du premier ordre.



## 2. Systeme du deuxieme ordre

Un système physique d'entrée  $e(t)$  et de sortie  $s(t)$  est du deuxième ordre s'il est régi par une équation différentielle du deuxième ordre du type :

$$\frac{1}{\omega_n^2} \cdot \frac{d^2 s(t)}{dt^2} + \frac{2z}{\omega_n} \cdot \frac{ds(t)}{dt} + s(t) = K \cdot e(t)$$

$K$  : gain statique

$\omega_n$  : pulsation propre

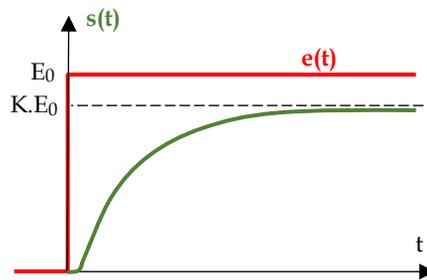
$z$  : facteur d'amortissement

Réponse indicielle : l'entrée est un échelon d'amplitude  $E$  pour  $t > 0$

Il existe 3 cas selon le facteur d'amortissement :

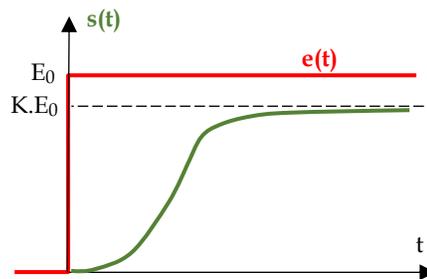
### 1<sup>er</sup> cas : $z > 1$ , régime aperiodique amorti

Le comportement du système est non oscillant, il tend vers la valeur  $K.E_0$  sans jamais la dépasser.



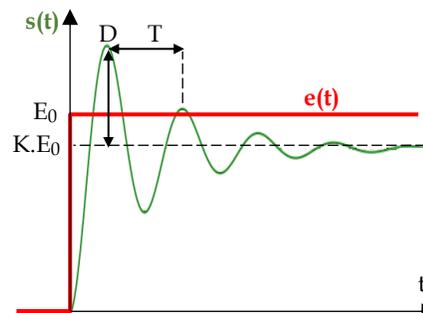
### 2<sup>e</sup> cas : $z = 1$ , régime aperiodique critique

La réponse est non oscillante, c'est le régime aperiodique le plus rapide.



### 3<sup>e</sup> cas : $z < 1$ , régime oscillatoire

La réponse présente la forme d'une sinusoïde amortie.



$D$  : dépassement

$T$  : pseudo-période

## Asservissement en vitesse d'un moteur à CC

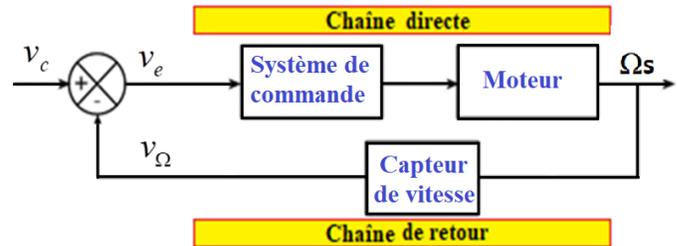
Le système est décrit par le schéma bloc ci-contre :

Supposons qu'une perturbation provoque une diminution de la vitesse de sortie  $\Omega_s$  (la consigne  $v_c$  étant constante).

La chaîne de retour fait alors apparaître une diminution de la grandeur de retour  $v_\Omega$ .

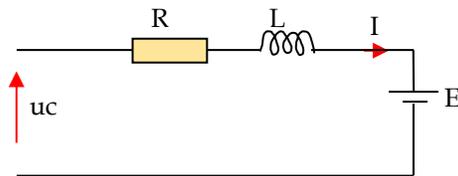
La grandeur d'entrée  $v_e$  va donc augmenter et la vitesse  $\Omega_s$  également. Elle va augmenter jusqu'à ce que la tension  $\varepsilon = v_e = v_c - v_\Omega$  s'annule.

Remarquons que ce résultat est obtenu sans qu'il soit nécessaire de connaître l'origine de la perturbation.



⇒ Mise en équation du moteur et sa charge

La mise en équation du système "moteur+charge" est réalisable à partir du modèle électrique de la machine à courant continu et de la relation fondamentale de la dynamique.



$R, L$  : résistance et inductance de l'induit  
 $E$  : force contre électromotrice

On a  $U_c = E + RI + L \frac{di}{dt}$  avec  $E = k' \Phi \Omega = k \Omega$  (à flux constant)  $\Rightarrow U_c = RI + L \frac{di}{dt} + k \Omega$  (1)

Le principe fondamental de la dynamique s'écrit :  $J \Omega' = C_u - C_r = C_{em} - C_f - C_r = C_{em} - C_r$  (si on néglige  $C_f$ , le couple des frottements secs)

Avec  $J$  : moment d'inertie des masses en mouvement

$C_{em}, C_u$  et  $C_r$  : respectivement couple électromagnétique, couple utile et couple résistant

On aussi  $C_{em} = kI$  et on suppose  $C_r = f \Omega$  où  $f$  est le coefficient de frottement visqueux

Ce qui donne  $J \Omega' = kI - f \Omega \Rightarrow I = \frac{J}{k} \cdot \Omega' + \frac{f}{k} \cdot \Omega$  (2)

⇒ Analyse temporelle

On néglige l'inductance  $L$  de l'induit

L'équation (1) devient  $U_c = RI + k \Omega$  or (2)  $I = \frac{J}{k} \cdot \Omega' + \frac{f}{k} \cdot \Omega \Rightarrow U_c = \frac{RJ}{k} \cdot \Omega' + \frac{Rf + k^2}{k} \cdot \Omega$

$\Rightarrow \frac{RJ}{Rf + k^2} \cdot \Omega' + \Omega = \frac{k}{Rf + k^2} \cdot U_c$

C'est une équation du 1<sup>er</sup> ordre avec la constante de temps  $\tau = \frac{RJ}{Rf + k^2}$  et le gain statique  $K = \frac{k}{Rf + k^2}$ .

La réponse indicielle (à un échelon)  $U_c = E_0$  pour  $t > 0$  est  $\Omega(t) = K \cdot E_0 \cdot (1 - e^{-\frac{t}{\tau}})$  (voir plus haut).

Le système est stable et ne risque aucun dépassement. Le temps de réponse à 5% est  $3\tau$ .

### On tient compte de l'inductance L de l'induit

En tenant compte des équations (1) et (2), il vient  $U_c = R \left( \frac{J}{k} \cdot \Omega' + \frac{f}{k} \cdot \Omega \right) + L \left( \frac{J}{k} \cdot \Omega'' + \frac{f}{k} \cdot \Omega' \right) + k\Omega$

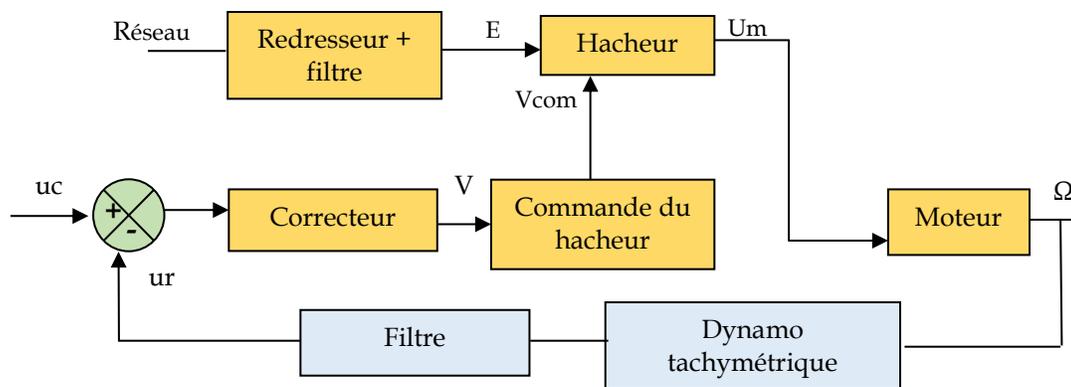
Ce qui donne  $\frac{LJ}{Rf+k^2} \cdot \Omega'' + \frac{RJ+Lf}{Rf+k^2} \cdot \Omega' + \Omega = \frac{k}{Rf+k^2} \cdot U_c$

C'est une équation du 2<sup>ème</sup> ordre.

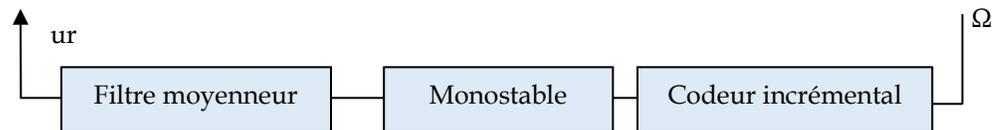
La pulsation propre, le facteur d'amortissement et le gain statique sont tels que (voir plus haut) :

$$\frac{1}{\omega_n^2} = \frac{LJ}{Rf+k^2} \quad , \quad \frac{2z}{\omega_n} = \frac{RJ+Lf}{Rf+k^2} \quad \text{et} \quad K = \frac{k}{Rf+k^2}$$

### ⇒ Montage d'asservissement

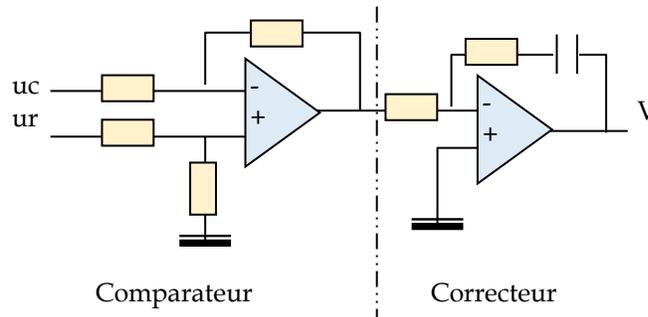


### Autre structure de la chaîne de retour



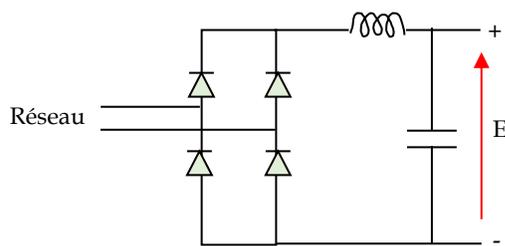
- La dynamo tachymétrique, capteur de vitesse, est en général très bruitée ; il va falloir filtrer sa sortie avec un passe-bas qui élimine une partie suffisante du bruit ;
- Pour qu'un asservissement soit correct, il faut d'abord qu'il assure la **stabilité** du système. Ceci étant, on recherche souvent à ce qu'en régime permanent, la sortie tende vers la consigne (**précision**). On peut aussi s'imposer un temps de réponse court (**rapidité**). Pour atteindre ces objectifs, on introduit un **correcteur** dans la chaîne directe.

L'ensemble "comparateur (soustracteur) + correcteur" peut avoir la structure suivante :

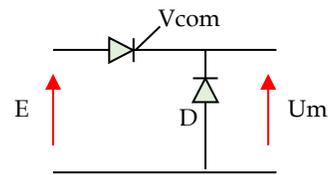


### Alimentation du hacheur

Cette alimentation est réalisée au moyen d'un pont redresseur et d'un filtre :



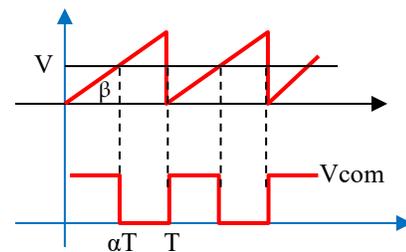
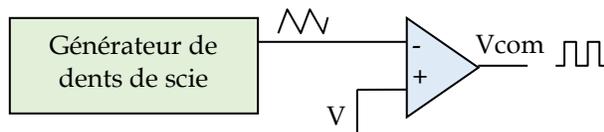
### Hacheur



Le hacheur débite au moteur une tension  $U_m = \alpha \cdot E$  où  $\alpha$  est le rapport cyclique proportionnel à la tension de commande  $V$ .

$D$  est une diode de roue libre.

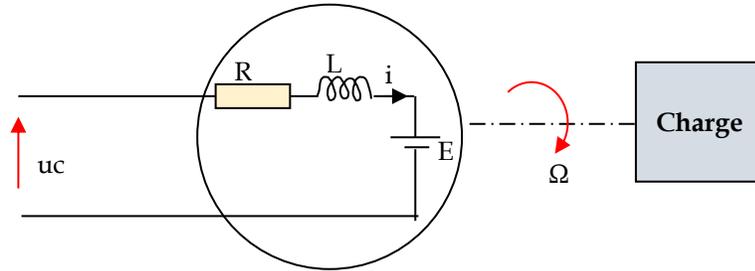
Pour obtenir un rapport cyclique  $\alpha$  proportionnel à la tension  $V$ , on compare celle-ci à un signal en dents de scie. La sortie du comparateur est le signal  $V_{com}$  envoyée au thyristor.



On a  $\text{tg}\beta = V/(\alpha \cdot T) \Rightarrow \alpha = V/(\text{tg}\beta \cdot T) = K \cdot V$   
 ( $\beta$  et  $T$  étant des caractéristiques du générateur de dents de scie)

**Exercice : Modélisation du moteur à CC par un schéma bloc**

L'objectif est de représenter l'ensemble "moteur à courant continu + charge" par un schéma bloc en vue d'en étudier l'asservissement en vitesse. On négligera l'inductance L du bobinage de l'induit.



| Relation   | Schéma fonctionnel |
|--|--------------------|
| <p><u><i>i</i> en fonction de <i>uc</i> et <i>E</i></u></p> <p>.....</p> <p>.....</p> <p>.....</p>   |                    |
| <p><u><i>E</i> en fonction de <math>\Omega</math></u></p> <p>.....</p> <p>.....</p>  |                    |
| <p><u><math>\Omega</math> en fonction de <i>Cem</i> et de <i>Cr</i></u></p> <p>.....</p> <p>.....</p> <p>.....</p>   |                    |
| <p><u><i>Cr</i> en fonction de <math>\Omega</math></u></p> <p>On admet que le couple résistant <i>Cr</i> est proportionnel à <math>\Omega</math>.<br/>Soit, <math>Cr = f \cdot \Omega</math> où <i>f</i> est le coefficient de frottement visqueux</p> |                    |
| <p><u><i>Cem</i> en fonction de <i>i</i></u></p> <p>.....</p> <p>.....</p> <p>.....</p>  |                    |
| <p><u>Schéma bloc</u></p>  |                    |

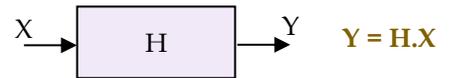
**Schéma bloc (ou diagramme fonctionnel)**

Un schéma fonctionnel est une représentation simplifiée d'un processus mettant en évidence les différentes fonctions mises en œuvre.

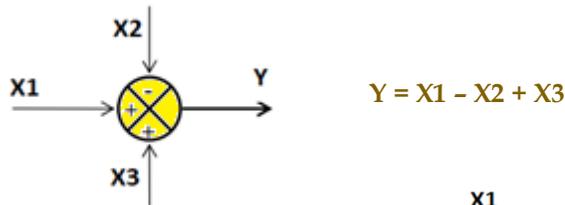
**1. Formalisme**

⇒ **Bloc**

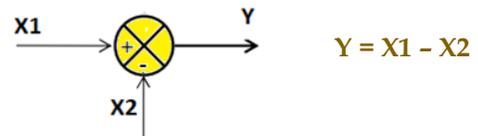
Un bloc est caractérisé par sa fonction de transfert ou transmittance H



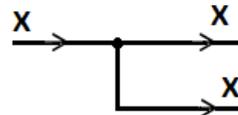
⇒ **Sommateur**



⇒ **Comparateur**



⇒ **Point de prélèvement**



**2. Simplification des schémas blocs**

|                             |  |  |
|-----------------------------|--|--|
| Blocs en cascade (en série) |  |  |
| Blocs en parallèle          |  |  |
| Structure en boucle fermée  |  |  |
| Cas du retour unitaire      |  |  |

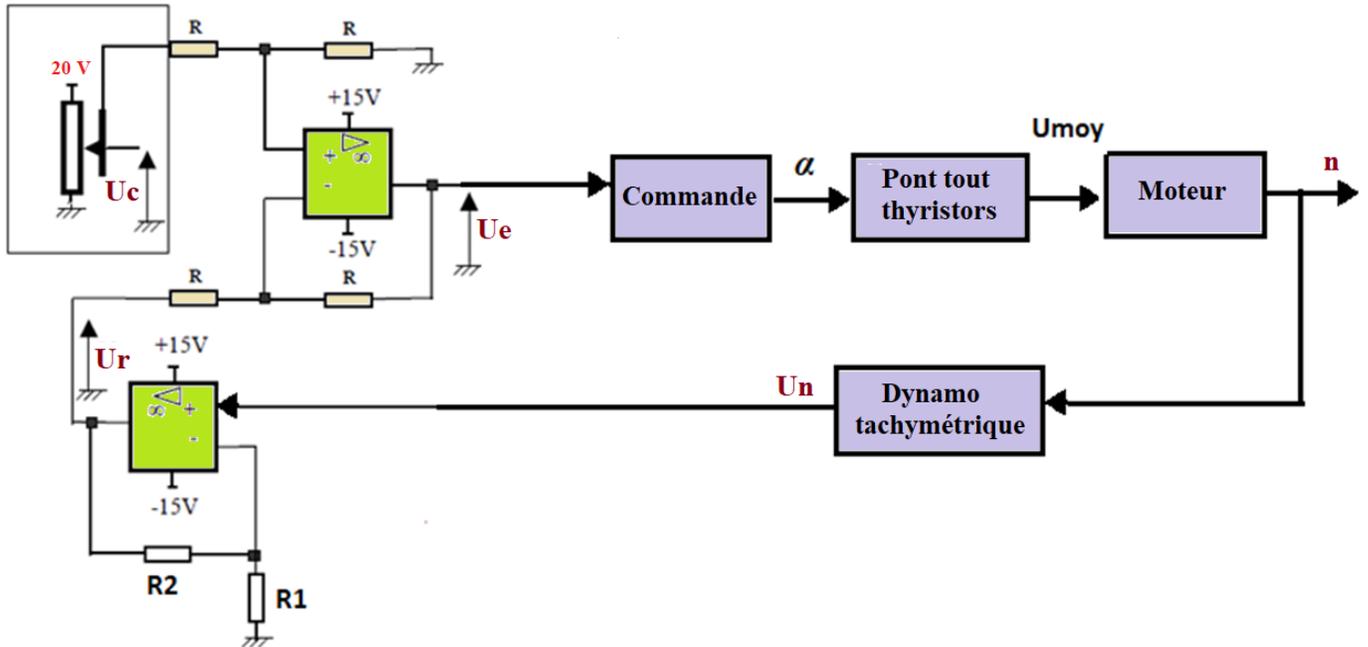




## Exercice 2 : régulation de vitesse de rotation d'un moteur à CC

L'exercice propose d'étudier la régulation de vitesse d'un moteur à courant continu, afin d'éviter des variations importantes de vitesse lors d'une perturbation. Le descriptif du système de régulation est le suivant :

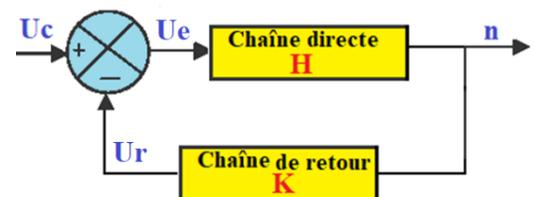
Potentiomètre de consigne



La régulation par le système bouclé peut être schématisée comme suit :

La chaîne de retour est constituée par l'ensemble "dynamo tachymétrique-amplificateur"

La dynamo tachymétrique délivre une tension  $U_n$  proportionnelle à la vitesse de rotation du moteur :  $U_n = 5 \cdot 10^{-3} \cdot n$  (n en tr/min et  $U_n$  en V).



### Compareur

Les amplificateurs opérationnels sont supposés parfaits et fonctionnent en régime linéaire, montrer que  $U_e = U_c - U_r$ .

### Chaîne de retour

- Montrer que  $U_r = \left(1 + \frac{R_2}{R_1}\right) \cdot U_n$
- On donne  $R_1 = 10 \text{ k}\Omega$ , calculer  $R_2$  pour que la transmittance de la chaîne de retour soit  $K = 0,01 \text{ V}/(\text{tr}/\text{min})$ .

### Régulation de vitesse

$U_e$  est une tension de commande qui sert à régler l'angle de retard à l'amorçage des thyristors et donc à régler la tension moyenne  $U_{moy}$  à la sortie du pont. La relation liant la tension  $U_e$  à la tension moyenne est  $U_{moy} = 100 \cdot U_e$ .

On négligera la chute de tension dans l'induit du moteur ; la vitesse de rotation est alors pratiquement proportionnelle à  $U_{moy}$  ; soit  $n = 50 \cdot U_{moy}$  ( $U_{moy}$  en V et n en tr/min).

- Calculer la transmittance  $H$  de la chaîne directe.
- La tension de consigne  $U_c$  étant maintenue constante, calculer  $U_e$  et  $U_r$  pour une vitesse de  $n = 1\,000 \text{ tr}/\text{min}$ . En déduire la tension de consigne  $U_c$  permettant le réglage de cette vitesse.
- Une perturbation tend à diminuer la vitesse du moteur. Donner le sens de variation des grandeurs  $U_r$ ,  $U_e$  et n puis conclure sur l'intérêt du bouclage réalisé.



**Exercice 3 : examen national 2018, session de rattrapage**
**SEV 3 : Asservissement de vitesse et traitement de l'information**

Le réglage de la vitesse du moteur se fait en agissant sur le rapport cyclique de la tension du moteur  $u_m(t)$ .

En plus, la vitesse dépend de plusieurs paramètres tels que :

- ✓ La charge du chariot ;
- ✓ La pente et l'état du terrain.

On se propose d'asservir la vitesse du chariot à la vitesse choisie par l'utilisateur (consigne).

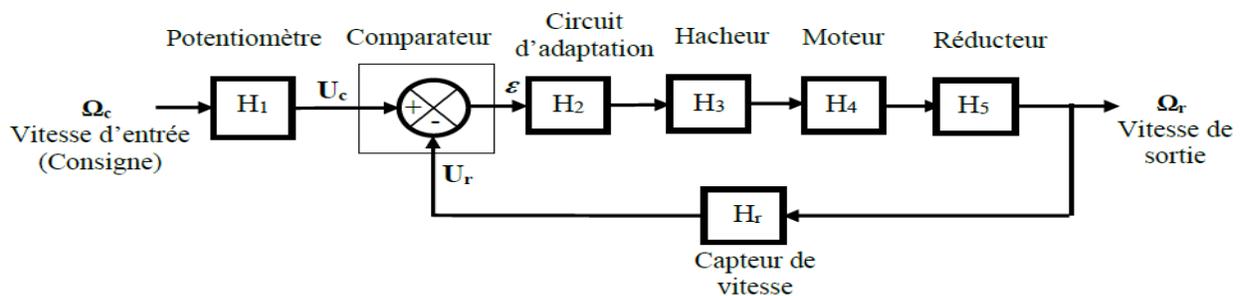
Le principe de cet asservissement consiste donc à :

- Mesurer la vitesse du chariot et délivrer une tension  $U_r$  image de la vitesse ;
- Comparer cette tension à la tension  $U_c$  image de la vitesse désirée ;
- Agir sur le moteur en fonction de l'écart entre  $U_c$  et  $U_r$ .

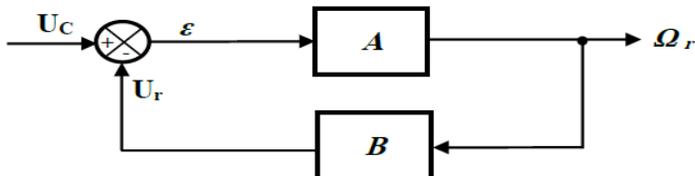
La comparaison des tensions et la commande du hacheur s'effectuent par le microcontrôleur.

**Tâche 1 : Modélisation du système**

Le modèle équivalent est représenté par le schéma bloc suivant :



Ce schéma-bloc peut être simplifié et remplacé par le schéma équivalent suivant :



Avec :  $U_c = H_1 \cdot \Omega_c$

On note :

- $A$  : la fonction de transfert de la chaîne directe ;
- $B$  : la fonction de transfert de la chaîne de retour.

**Q26)** Exprimer  $A$  et  $B$  en fonction de  $H_2$ ,  $H_3$ ,  $H_4$ ,  $H_5$  et  $H_r$ .

**1,5 pt**

**Q27)** Donner en fonction de  $A$  et de  $B$  :

**Q27-1)** La fonction de transfert en boucle ouverte FTBO :  $T_{BO} = U_r / \varepsilon$ .

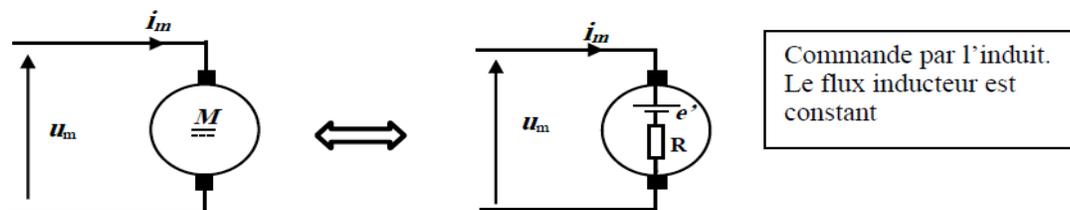
**1,5 pt**

**Q27-2)** Donner la fonction de transfert en boucle fermée FTBF :  $T_{BF} = \Omega_r / U_c$ .

**2 pts**

**Tâche 2 : Modélisation du moteur**

Le schéma équivalent du moteur à courant continu est le suivant :



- $u_m = Ri_m + e'$
- $e' = a \cdot \Omega_m$
- $C_m = a \cdot I_m$

avec :

- $u_m$  : tension aux bornes de l'induit du moteur (V).
- $R$  : résistance de l'induit du moteur ( $\Omega$ ).
- $e'$  : f.c.é.m du moteur (V).
- $a$  : constante de la f.c.é.m (V.s/rad) ou constante de couple (N.m/A).
- $\Omega_m$  : vitesse angulaire du moteur (rad/s).
- $C_m$  : couple moteur (N.m).

En appliquant la loi fondamentale de la dynamique au moteur, on obtient :  $C_m - C_r - C_f = J \frac{d\Omega_m}{dt}$

avec J : le moment d'inertie de l'ensemble mobile en rotation.

On suppose que :

- ✓ Le moteur est à vide : couple résistant  $C_r = 0$  ;
- ✓ Les frottements mécaniques  $C_f$  sont négligeables.

**Q28)** Montrer que la vitesse du moteur  $\Omega_m$  est liée à la tension  $u_m$  par l'équation différentielle suivante :

$$\left(\frac{RJ}{a^2}\right) \frac{d\Omega_m}{dt} + \Omega_m = \frac{u_m}{a}$$

2 pts

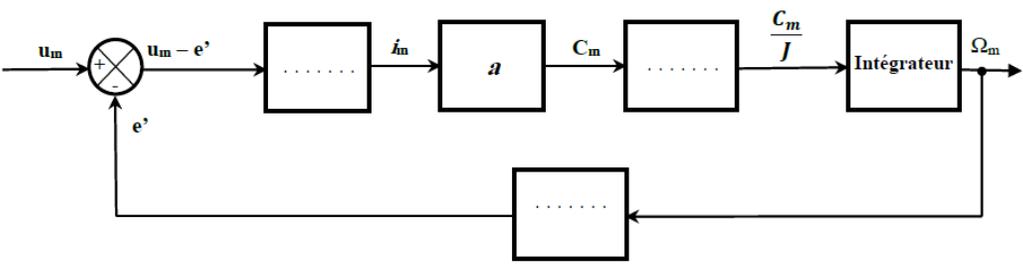
**Q29)** S'agit-il d'un système de premier ou de deuxième ordre ?

0,5 pt

**Q30)** Sur le document **DREP 06**, compléter alors le schéma-bloc du moteur.

1,5 pt

.....  
 .....  
 .....  
 .....  
 .....  
 .....  
 .....  
 .....  
 .....  
 .....  
 .....  
 .....  
 .....  
 .....  
 .....  
 .....  
 .....  
 .....



**Exercice 4 : examen national 2023, session de rattrapage**

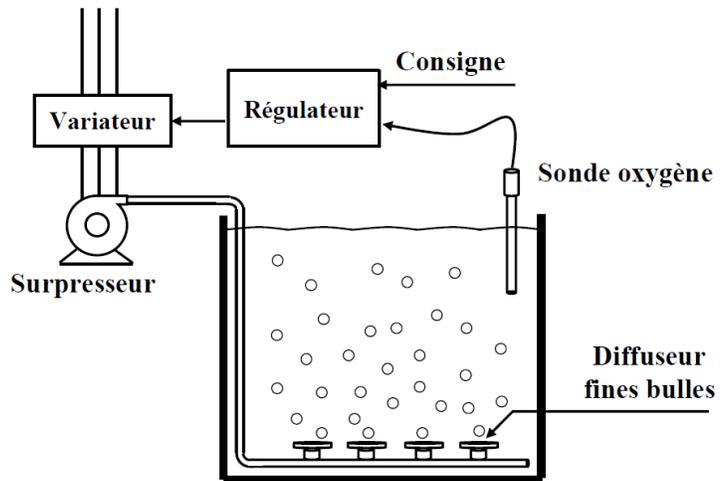
Pour réaliser un traitement biologique efficace, il est nécessaire de réguler le taux d'oxygène dissous dans le bassin d'aération et donc de contrôler la vitesse du surpresseur selon le besoin en oxygène.

**Tâche 1 : régulation du taux d'oxygène dissous (OD) dans le bassin d'aération**

La figure ci-contre fournit le schéma de la boucle de régulation du taux d'oxygène dissous (OD).

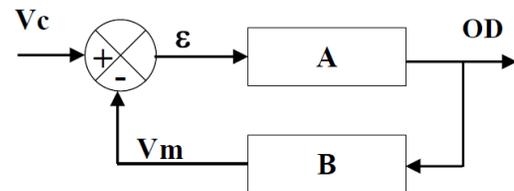
⇒ **Boucle de régulation**

Le régulateur compare la valeur mesurée par la sonde oxygène à la consigne et élabore le signal de commande envoyé au variateur de vitesse. La modification de vitesse du surpresseur se traduit par celle du débit d'air soufflé dans le bassin et donc du taux d'oxygène OD



**Boucle de régulation de l'oxygène dissous**

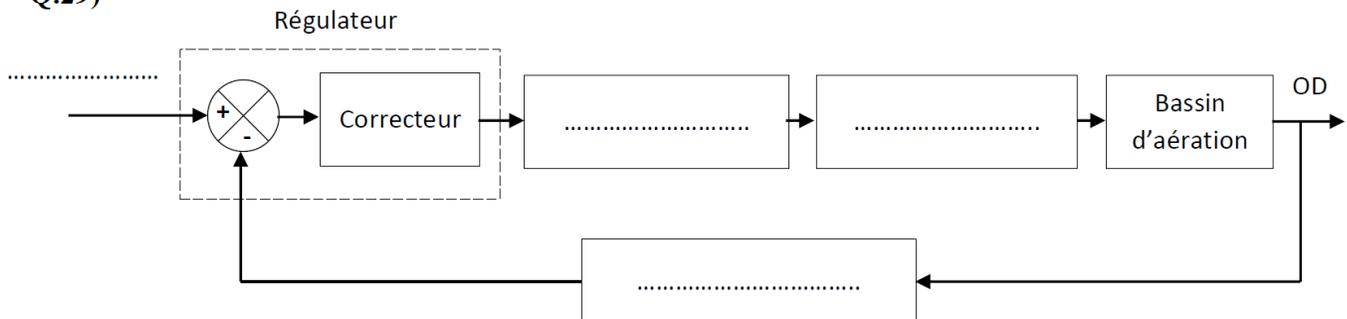
Le schéma fonctionnel de cette boucle de régulation peut être modélisé comme le montre la figure ci-contre.



**Q.29)** À partir du schéma de la boucle de régulation ci-dessus, compléter le schéma-bloc fonctionnel de cette régulation (utiliser les propositions : surpresseur, sonde oxygène, variateur, consigne) **2 pts**

**Q.30)** Parmi les propositions données, lesquelles sont correctes ? **1,5 pt**

**Q.29)**



Q.30) Cocher les cases appropriées

| Proposition   | Vrai                     | Faux                     |
|---|--------------------------|--------------------------|
| Un système régulé est un système bouclé pour lequel la sortie poursuit une consigne variable. | <input type="checkbox"/> | <input type="checkbox"/> |
| Un système régulé est un système bouclé qui maintient la sortie conforme à une consigne fixe. | <input type="checkbox"/> | <input type="checkbox"/> |
| Le bloc <b>A</b> correspond à la chaîne directe du système bouclé.                            | <input type="checkbox"/> | <input type="checkbox"/> |
| Le bloc <b>B</b> correspond à la chaîne de retour du système bouclé.                          | <input type="checkbox"/> | <input type="checkbox"/> |
| La fonction de transfert en boucle fermée est : $\frac{A}{1+A.B}$                             | <input type="checkbox"/> | <input type="checkbox"/> |
| La fonction de transfert en boucle ouverte est : $A + B$                                      | <input type="checkbox"/> | <input type="checkbox"/> |

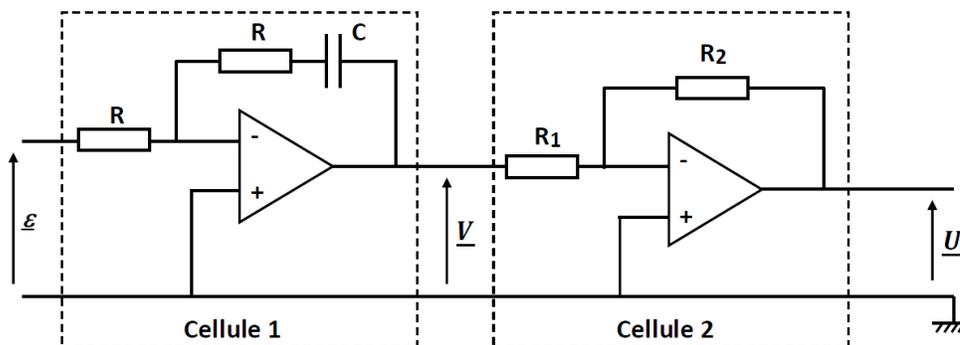
⇒ Synthèse d'un correcteur PI

Dans une boucle de régulation, le correcteur élabore le signal de commande à partir du signal d'erreur. Son but est d'améliorer les performances de la régulation.

Q.34) Quelles sont ces performances visées par le correcteur (cocher les bonnes réponses)

1,5 pt

Le montage suivant représente un correcteur **PI** dit à architecture mixte :



Q.35) Montrer que la fonction de transfert complexe de la cellule 1 est  $\frac{V}{\varepsilon} = -(1 + \frac{1}{jRC\omega})$ ;

2 pts

Q.36) Donner, en fonction de  $R_1$  et  $R_2$ , la fonction de transfert complexe  $\frac{U}{V}$  de la cellule 2 ;

2 pts

Q.37) En déduire que la fonction de transfert complexe  $\underline{H} = \frac{U}{\varepsilon}$  du correcteur s'écrit  $\underline{H} = \frac{R_2}{R_1} (1 + \frac{1}{jRC\omega})$ ;

2 pts

La forme canonique de la fonction de transfert de ce correcteur est :  $\underline{H} = K_p (1 + \frac{1}{T_i j\omega})$ .

$K_p$  et  $T_i$  s'appellent respectivement gain proportionnel et constante de temps d'action intégrale.

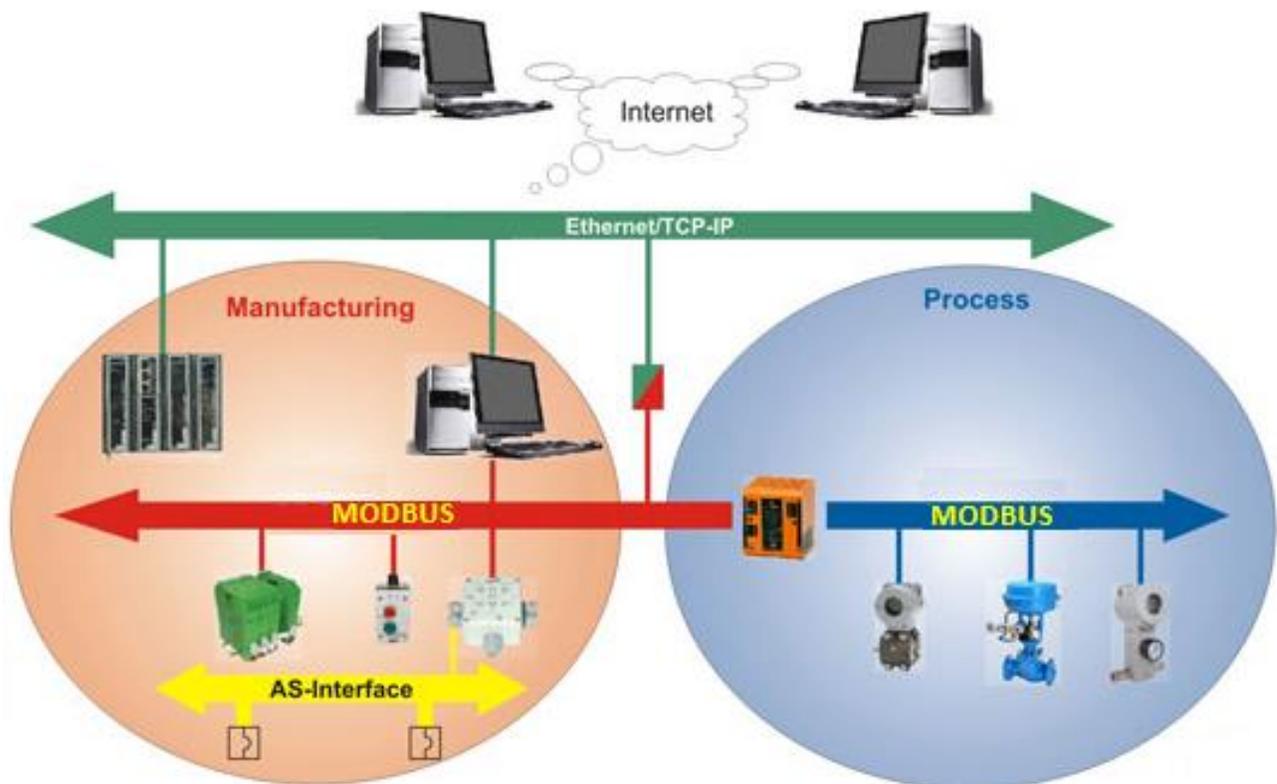
On se propose de déterminer ces deux paramètres par une méthode expérimentale.

Cette méthode de réglage des correcteurs repose sur le principe de la recherche des oscillations en faisant varier le gain proportionnel. On note, au point d'oscillations, le gain  $K_{osc}$  et la période des oscillations  $T_{osc}$



**PARTIE****3**

# La fonction COMMUNIQUER



## 9

## COMMUNICATION DE L'INFORMATION - RESEAUX INDUSTRIELS -

### Généralités sur les réseaux informatiques

Un réseau est un ensemble de machines interconnectées dans le but d'échanger des informations et de partager des ressources (fichiers, imprimante...).

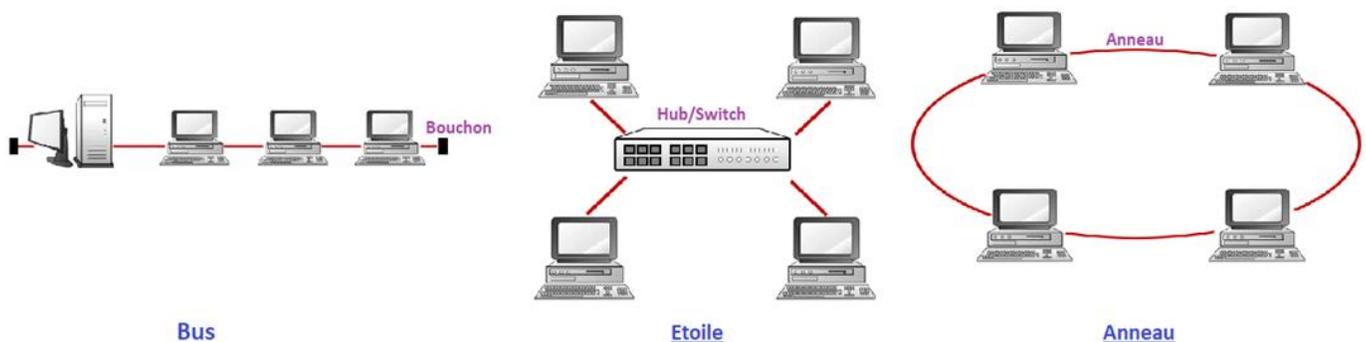
En fonction de la localisation, la distance et le débit, les réseaux sont classés en trois types :

- **LAN** (Local Area Network) : réseau local, intra entreprise permettant l'échange de données et le partage de ressources (ex : Ethernet) ;
- **MAN** (Metropolitan Area Network) : réseau métropolitain qui permet la connexion de plusieurs sites à l'échelle d'une ville ;
- **WAN** (Wide Area Network) : réseau à l'échelle d'un pays ou d'un continent, généralement celui des opérateurs. Le plus connu des WAN est Internet.

#### ⇒ Topologie

Par topologie, nous entendons la façon dont on connecte les machines au serveur. Il y en a trois principales :

- **Topologie en bus** : tous les ordinateurs sont reliés à une même ligne par le biais d'un câble ;
- **Topologie en étoiles** : la communication entre deux ordinateurs passe par le matériel central (routeur, commutateur, concentrateur, ...)
- **Topologie en anneau (ring)** : les équipements sont reliés entre eux en formant une boucle.



#### ⇒ Méthodes d'accès

Une méthode d'accès est un ensemble de règles définissant la façon dont un ordinateur accède au support tout en évitant les collisions.

On distingue les méthodes :

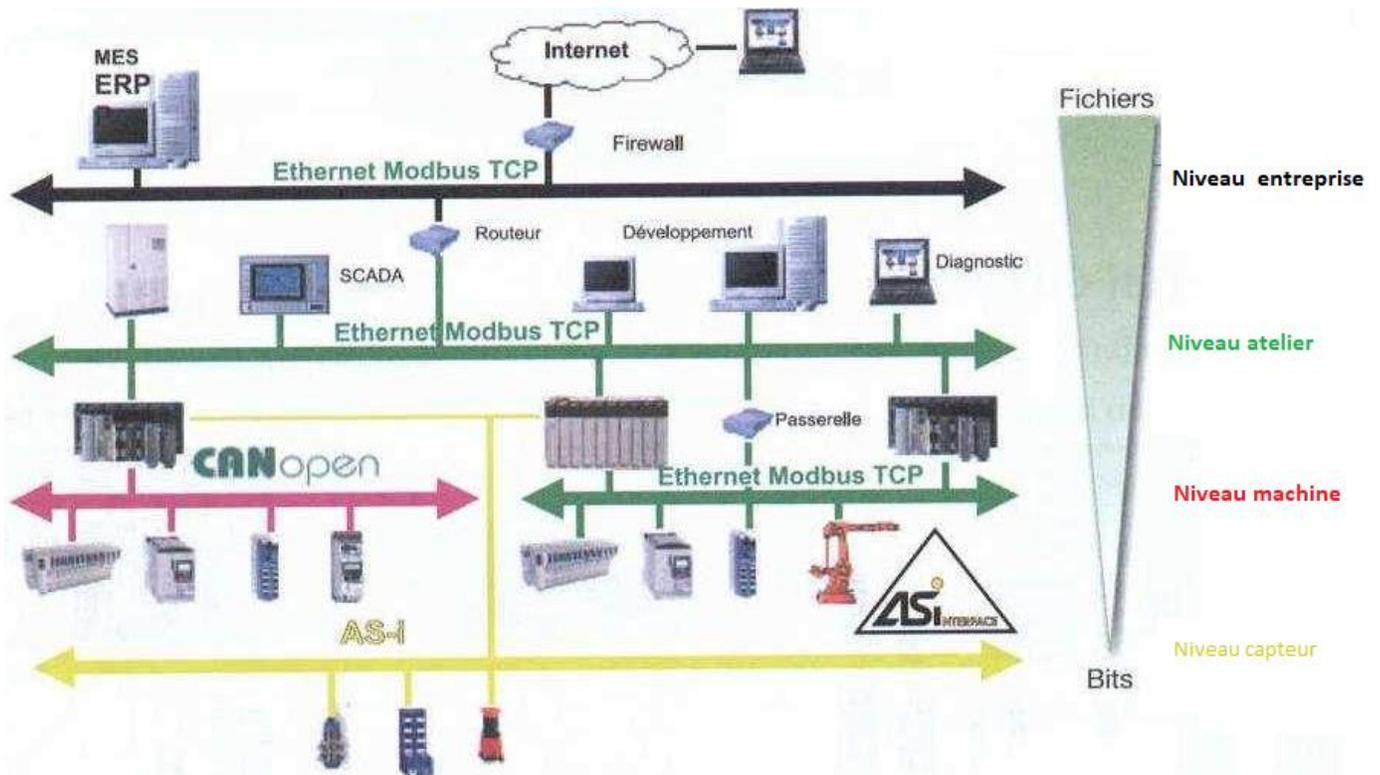
- **CSMA/CD** (Carrier Sense Multiple Access with Collision Detect) pour les réseaux en bus et en étoile. Méthode utilisée par Ethernet ;
- **Token Ring** (anneau à jeton) pour les réseaux en anneau.

⇒ **Pyramide CIM (Computer Integrated Manufacturing)**

La technologie CIM définit le modèle d'usine industrielle au sens de l'automatisation et de la communication, c'est-à-dire que tous les équipements de l'usine fonctionnent sous le contrôle permanent des ordinateurs, API et autres dispositifs numériques.

La CIM propose quatre niveaux. Plus on s'élève dans la pyramide, plus le niveau de décision est important. Un niveau supérieur décide ce qu'un niveau inférieur exécute.

|   | <i>Fonction</i>   | <i>Réseau</i>   |
|---|---|---|
| <p><i>Niveau 3</i></p> <p><b>Entreprise</b></p>         | <p>Informatique de gestion de l'entreprise (PC, serveur)</p>              | <p>Réseau informatique (Ethernet, TCP)</p>            |
| <p><i>Niveau 2</i></p> <p><b>Atelier</b></p>            | <p>Gestion locale<br/>Contrôle de production (PC industriel, serveur)</p> | <p>Réseau local industriel (Ethernet, TCP Modbus)</p> |
| <p><i>Niveau 1</i></p> <p><b>Machine</b></p>            | <p>Traitement d'automatisation (API, <math>\mu</math>C, régulateur)</p>   | <p>Réseau de terrain (Modbus)</p>                     |
| <p><i>Niveau 0</i></p> <p><b>Capteur Actionneur</b></p> | <p>capteurs/ actionneurs</p>  | <p>Réseau de capteurs/ actionneurs (ASI)</p>          |



## Réseau Ethernet

La technologie Ethernet permet de connecter les différents équipements entre eux dans un LAN (Local Area Network). Plusieurs réseaux peuvent communiquer entre eux par des routeurs pour former un WAN (Wide Area Network). La topologie la plus courante pour un réseau local est une topologie en étoile. Ethernet utilise le mécanisme CSMA/CD.

### ⇒ Équipements

- Carte réseau dans le PC ;
- Prises murales de type RJ45 ;
- Câble en cuivre à paires torsadées ou en fibres optiques ;
- Concentrateurs (hub), commutateurs (switchs), routeurs ;

Selon l'importance et le type de réseau, on utilise le plus souvent un des trois concentrateurs les plus généraux : les Hubs, Switchs et Routeurs.

**Hub** (concentrateur) : utilisé pour un réseau local avec un nombre très limité de machines. Il n'est ni plus ni moins qu'une multiprise RJ45 qui amplifie le signal réseau.

**Switch** (ou commutateur) : distribue les données à chaque machine destinataire, alors que le hub envoie les données à toutes machines qui répondent. Il élimine les collisions de paquets éventuelles.

**Routeur** : autorise l'utilisation de plusieurs classes d'adresses IP au sein d'un même réseau. Il permet ainsi la création de sous-réseaux.

### ⇒ Protocoles de communication

Un protocole est un ensemble de règles et de procédures à respecter pour émettre et recevoir des données sur un réseau. Le plus souvent, c'est la suite de protocoles **TCP/IP** qui est utilisée.

Voici quelques exemples de protocoles utilisant TCP/IP :

- **HTTP** (HyperText Transfer Protocol) : protocole de navigation sur le web ;
- **FTP** (File Transfer Protocol) : protocole pour le transfert de fichiers.

TCP/IP (Transmission Control Protocole / Internet Protocole) :

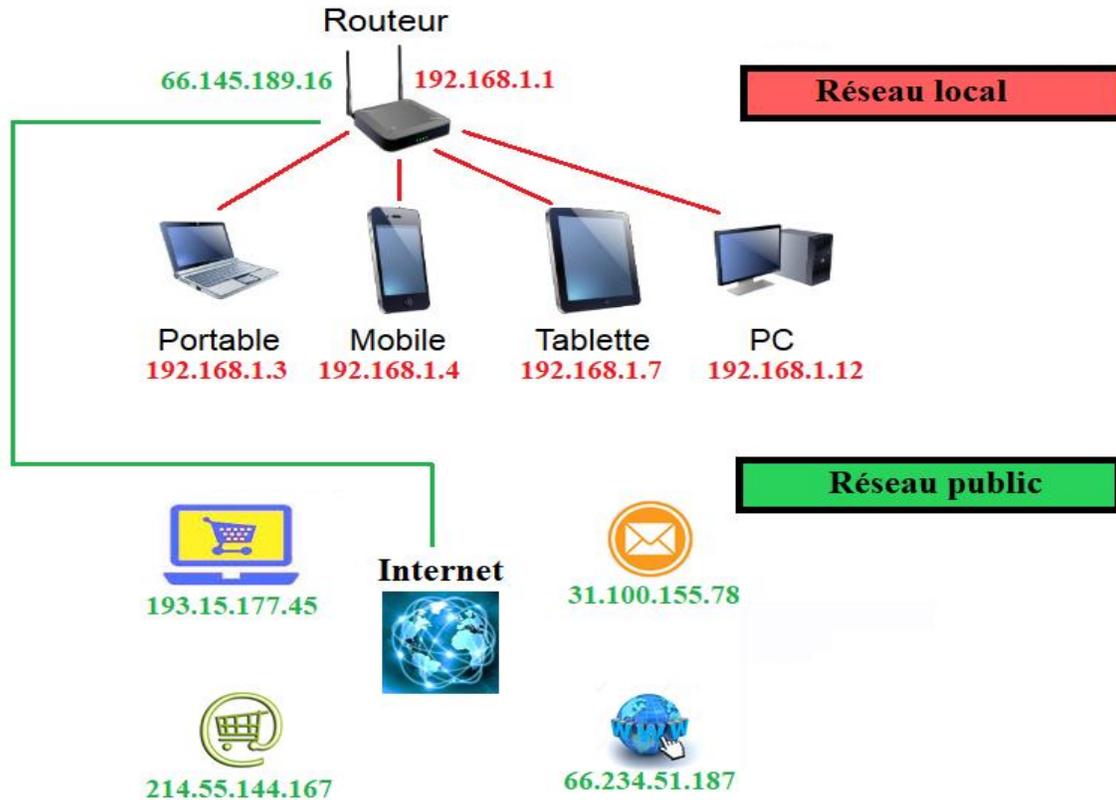
- Le protocole **IP** se base sur l'adressage IP, c'est-à-dire le fait de fournir une adresse IP à chaque machine du réseau afin de pouvoir acheminer des paquets de données ;
- Le protocole **TCP** assure le contrôle de la transmission des données.

### ⇒ Adressage IP

Une adresse IP est le numéro qui identifie chaque ordinateur connecté à Internet, ou plus généralement, l'interface avec le réseau de tout matériel informatique (PC, routeur, imprimante) connecté à un réseau informatique utilisant l'Internet Protocol.

En attendant la généralisation de la nouvelle version d'IPv6 avec un adressage sur 128 bits, IPv4 utilise 32 bits représentés par 4 valeurs décimales chacune allant de 0 à 255.

L'adresse est constituée de deux parties : un identificateur de réseau (netId) et un identificateur de la machine (hostId) pour ce réseau.



Il existe trois classes d'adresses, chacune permettant de coder un nombre différent de réseaux et de machines :

- **Classe A** : 1 octet identificateur du réseau, 3 octets d'hôtes.  
1<sup>er</sup> Octet compris entre 1 et 127.
- **Classe B** : 2 octets réseau, 2 octets d'hôtes.  
1<sup>er</sup> Octet compris entre 128 et 191.
- **Classe C** : 3 octets réseau, 1 octet d'hôtes.  
1<sup>er</sup> Octet compris entre 192 et 223.

**Classe A : De 0.0.0.0 à 127.255.255.255**

|               |           |             |           |
|---------------|-----------|-------------|-----------|
| 0XXX XXXX     | XXXX XXXX | XXXX XXXX   | XXXX XXXX |
| octet 1       | octet 2   | octet 3     | octet 4   |
| partie réseau |           | partie hôte |           |

**Classe B : De 128.0.0.0 à 191.255.255.255**

|               |           |             |           |
|---------------|-----------|-------------|-----------|
| 10XX XXXX     | XXXX XXXX | XXXX XXXX   | XXXX XXXX |
| octet 1       | octet 2   | octet 3     | octet 4   |
| partie réseau |           | partie hôte |           |

**Classe C : De 192.0.0.0 à 223.255.255.255**

|               |           |           |             |
|---------------|-----------|-----------|-------------|
| 110X XXXX     | XXXX XXXX | XXXX XXXX | XXXX XXXX   |
| octet 1       | octet 2   | octet 3   | octet 4     |
| partie réseau |           |           | partie hôte |

Masque de sous-réseau

Il permet d'identifier le réseau associé à une adresse IP.

Pour connaître, par exemple, l'adresse du réseau associé à l'adresse IP **34.208.123.12** (classe A), le masque est **11111111.00000000.00000000.00000000** ; soit **255.0.0.0** (/8)

Remarque : si on ajoute le nombre entre parenthèses, c'est pour indiquer la taille de l'adresse réseau.

La valeur binaire de 34.208.123.12 est 00100010.11010000.01111011.00001100 .

Un ET logique entre l'adresse IP et le masque donne ainsi le résultat suivant :

```

00100010.11010000.01111011.00001100
ET   11111111.00000000.00000000.00000000
=    00100010.00000000.00000000.00000000
soit   34.0.0.0

```

Il s'agit bien du réseau associé à l'adresse 34.208.123.12 .

En généralisant, il est possible d'obtenir les masques par défaut correspondant à chaque classe d'adresse :

- Pour une adresse de **Classe A**, le masque est **255.0.0.0**
- Pour une adresse de **Classe B**, le masque est **255.255.0.0**
- Pour une adresse de **Classe C**, le masque est **255.255.255.0**

**Exercice**

- Quelle est la classe d'adresse de la machine dont l'adresse IP est 144.19.74.201 ? .....
- Quelle est l'adresse du réseau ? .....
- Quel est l'identificateur de la machine dans le réseau ? .....
- Quel est le masque réseau ? .....

**Exercice**

Compléter le tableau.

Note : on enlève toujours deux adresses :

- L'adresse du réseau : la partie hôte contient uniquement des 0 ;
- L'adresse de diffusion (broadcast) : la partie hôte contient uniquement des 1.

| Classe   | Adresse mini | Adresse maxi | Masque par défaut | Nombre de réseaux | Nombres de machines par réseau |
|----------|--------------|--------------|-------------------|-------------------|--------------------------------|
| <b>A</b> | .....        | .....        | .....             | .....             | .....                          |
| <b>B</b> | .....        | .....        | .....             | .....             | .....                          |
| <b>C</b> | .....        | .....        | .....             | .....             | .....                          |

Création de sous-réseaux

Soit l'adresse **34.208.123.12** et supposons que l'on désire que les deux premiers bits du deuxième octet permettent de désigner le sous-réseau. Le masque à appliquer sera alors : **11111111.11000000.00000000.00000000** c'est-à-dire **255.192.0.0**.

Ce masquage divise donc un réseau de classe A en 4 sous-réseaux ; d'où le nom de *masque de sous-réseau*. Si on note N le nombre de bits attribués en plus au réseau, le **nombre de sous-réseaux** sera  $2^N$

**Exercice**

- ☞ Pour l'adresse **145.245.45.225**, donner
- La classe d'adresse ;
  - Le masque de sous-réseau et l'adresse du réseau ;
  - Le nombre de machines possibles sur ce réseau ;
- ☞ Pour une subdivision de ce réseau en 64 sous-réseaux, donner :
- Le masque modifié ;
  - L'adresse du sous-réseau auquel appartient la machine **145.245.45.225**.
  - Le nombre de machines possibles par sous-réseau.

- a) Le premier octet de l'adresse est ..... : il s'agit d'une classe .....
- b) Le masque de sous-réseau d'une classe ..... est ..... (/.....). Nous aurons en binaire **11111111.11111111.00000000.00000000**.

Pour trouver l'adresse réseau, on va appliquer le masque de sous-réseau à l'adresse IP au travers d'une fonction "ET". Nous aurons .....

- c) Le nombre de machines possibles sur ce réseau est .....
- d) Pour obtenir 64 subdivisions du réseau, nous devons augmenter le masque réseau de ..... bits. Le masque de sous-réseau devient donc en binaire **11111111.11111111.11111100.00000000** soit ..... (/.....).
- e) Notre adresse IP 145.245.45.225 sera répartie comme suit :

|         |            |            |           |            |
|---------|------------|------------|-----------|------------|
| Décimal | <b>145</b> | <b>245</b> | <b>45</b> | <b>225</b> |
| Binaire |            |            |           |            |

Pour trouver l'adresse de sous-réseau qui contient la machine 145.245.45.225, nous allons appliquer le masque de sous-réseau à l'adresse IP au travers d'une fonction "ET". Nous aurons :

.....

**ET** .....

= .....

soit, en décimal .....

- f) Le nombre de machines possibles par sous-réseau est .....

## Réseau de terrain

Un bus de terrain est une ligne numérique, reliant différents types d'équipements d'automatisme au sein d'une zone géographique délimitée (usine, voiture, photocopieur ...). Les équipements de sécurité (frein ABS, capteur de vitesse, airbag...) d'une voiture moderne, communiquent à l'aide d'un bus de terrain.

Parmi les avantages des bus de terrain, on peut citer :

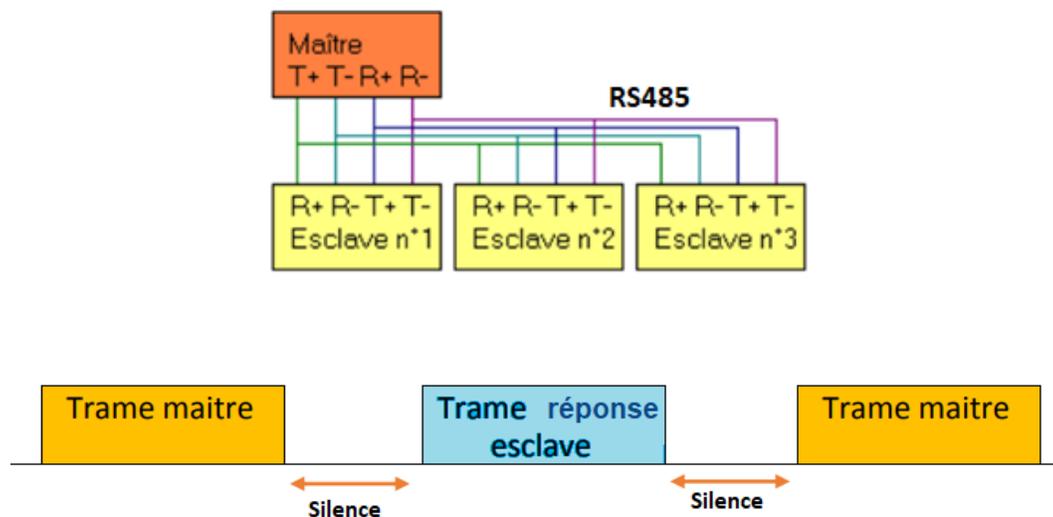
- Réduction massive du câblage, du temps d'installation, du matériel requis et des coûts de maintenance ;
- Plus de précision grâce à la numérisation.

### ⇒ Bus Modbus

Modbus (marque déposée par Modicon) est un protocole de communication utilisé pour des réseaux d'automates programmables. Il fonctionne selon une structure hiérarchisée entre un maître et plusieurs esclaves.

- Le maître envoie une demande et attend une réponse ;
- Deux esclaves ne peuvent dialoguer ensemble.

Le protocole MODBUS utilise généralement la transmission sérielle asynchrone RS232, RS422 ou RS485.

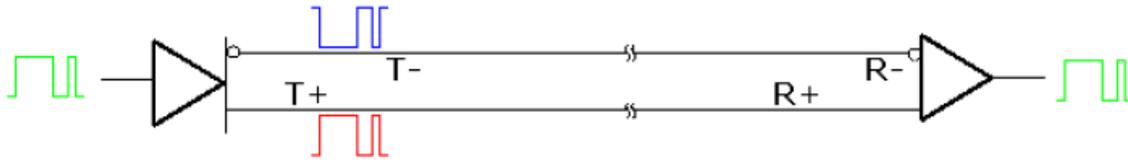


### RS-485

C'est une liaison série, de type asynchrone, multipoints et différentielle ; débit élevé (jusqu'à 10 Méga bits/s) sur une distance importante (jusqu'à 1200 m). Elle dispose de 2 bornes d'émission polarisées notées T+, T- et de 2 bornes de réception polarisées notées R+, R-.

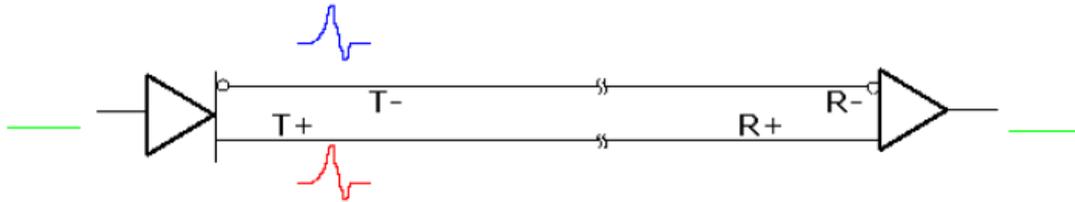
### Liaison différentielle

Sur une liaison différentielle (ou symétrique) les signaux (T+ et T-) sont transmis en opposition de phase. Le récepteur réalise la différence de ces deux signaux (R+ et R-) pour obtenir le signal utile.



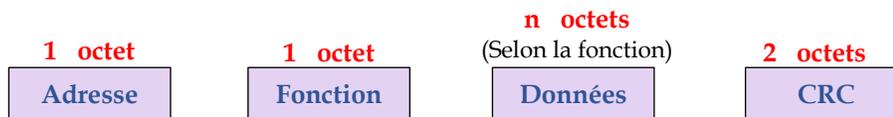
La paire de fils étant dans le même environnement. Donc, Si un parasite se produit, il se présente nécessairement sur les deux fils avec la même polarité.

Le récepteur, réalisant la différence des deux signaux, neutralise le parasite.



### Format d'une trame Modbus

Le format de la trame RTU est en général composé de 4 champs :



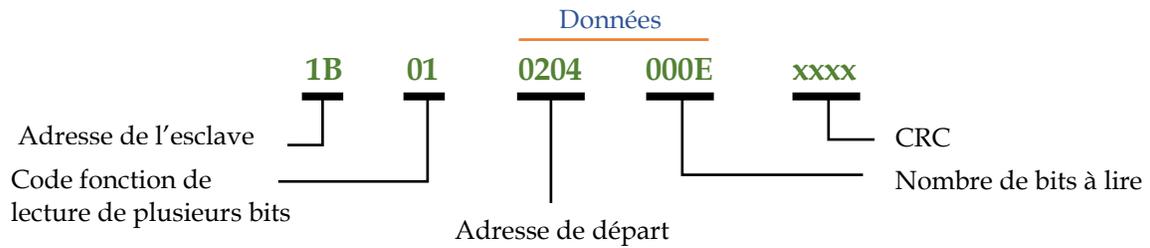
- Adresse : N° de l'esclave concerné par la requête ;
- Fonction : type d'action à réaliser ;
- Données : champ d'informations contenant les paramètres liés à la fonction : adresse mot, valeur de mot, nombre de mots, ...
- CRC (Cyclic Redundancy Check). : mot de contrôle de l'intégrité de la trame :
  - Le maître calcule le CRC codé sur 2 octets ;
  - L'esclave, lorsqu'il reçoit la trame, calcule le CRC et le compare au CRC reçu. En cas d'erreur, l'esclave répond par une trame d'exception.

La liste de quelques fonctions MODBUS est présentée dans le tableau suivant :

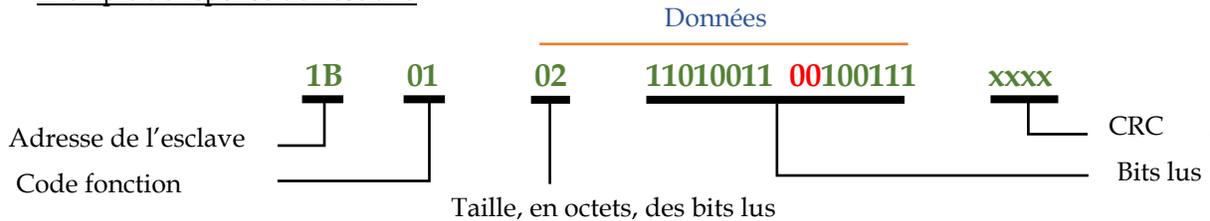
| CODE en Hexa | FONCTION                     |
|--------------|------------------------------|
| 01           | Lecture de N bits de sortie  |
| 02           | Lecture de N bits d'entrée   |
| 03           | Lecture de N mots de sortie  |
| 04           | Lecture de N mots d'entrée   |
| 05           | Ecriture d'un bit de sortie  |
| 06           | Ecriture d'un mot de sortie  |
| 0F           | Ecriture de N bits de sortie |
| 10           | Ecriture de N mots           |

**Exemple de transaction maitre-esclave Modbus**

- Lecture par le maitre des bits 204<sub>h</sub> à 211<sub>h</sub> (soit 14 bits ou 0E en hexa) sur l'esclave N° 27 (1B en hexa)



- Exemple de réponse de l'esclave

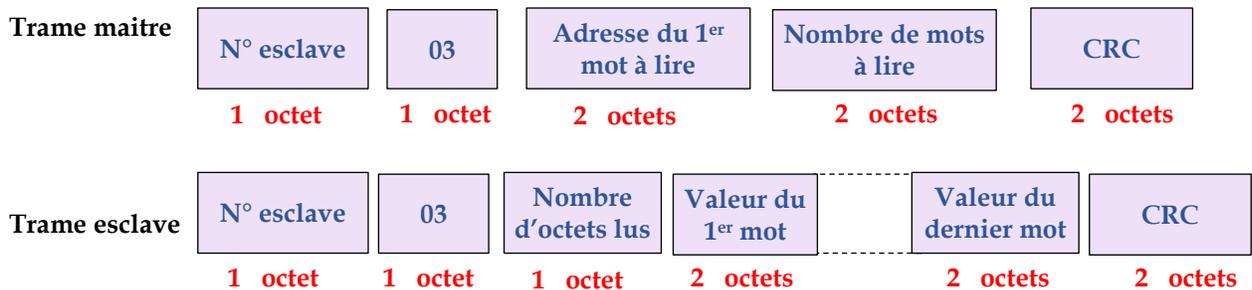


**Exercice**

On considère les trames Modbus suivantes :

|                             |    |    |    |    |    |    |    |    |
|-----------------------------|----|----|----|----|----|----|----|----|
| Requête émise par le maitre | 04 | 03 | 1C | 02 | 00 | 01 | 25 | 9F |
| Réponse émise par l'esclave | 04 | 03 | 02 | 09 | D5 | 74 | DE |    |

Pour la fonction 03 (lecture de plusieurs mots), les trames maitre et esclave sont montées comme suit :



Compléter le décodage de ces messages Modbus.

Trame maitre

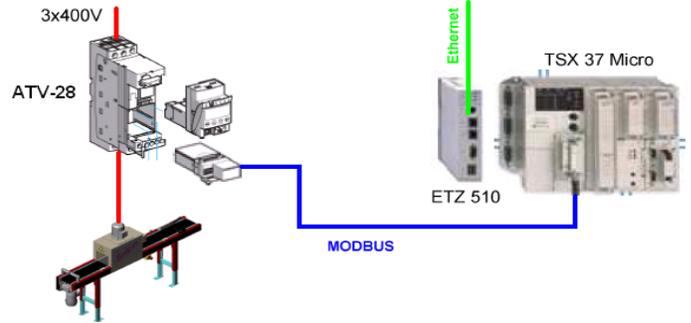
- Adresse de l'esclave : .....
- Code fonction : .....
- Nombre et adresse des mots à extraire : .....
- CRC : .....

Trame réponse esclave

- Adresse de l'esclave : .....
- Code fonction : .....
- Valeur du mot envoyé au maitre : .....
- CRC : .....

**Exercice**

Donnez la trame MODBUS de lecture par le maître des mots qui se trouvent sur la plage d'adresse W450<sub>(10)</sub> à W456<sub>(10)</sub> sur un ATV-28 (ALTIVAR), esclave n°2. (Le CRC n'est pas demandé)



$(450)_{10} = (\dots\dots\dots)_{16}$

Nombre de mots = .....

|                 |                                  |                  |                |               |
|-----------------|----------------------------------|------------------|----------------|---------------|
| .....           | .....                            | .....            | .....          | XX XX         |
| N° de l'esclave | Instruction de lecture de N mots | Adresse de début | Nombre de mots | Valeur du CRC |

**Exercice**

Donnez la trame MODBUS pour envoyer une question à l'esclave 4 pour charger la valeur 10<sub>(10)</sub> dans le registre W252<sub>(10)</sub>. La partie "Données", dans le cas de l'écriture de 1 mot, est composée de 4 octets : 2 octets de l'adresse de destination et de 2 autres de la valeur envoyée.

$(252)_{10} = (\dots\dots\dots)_{16}$

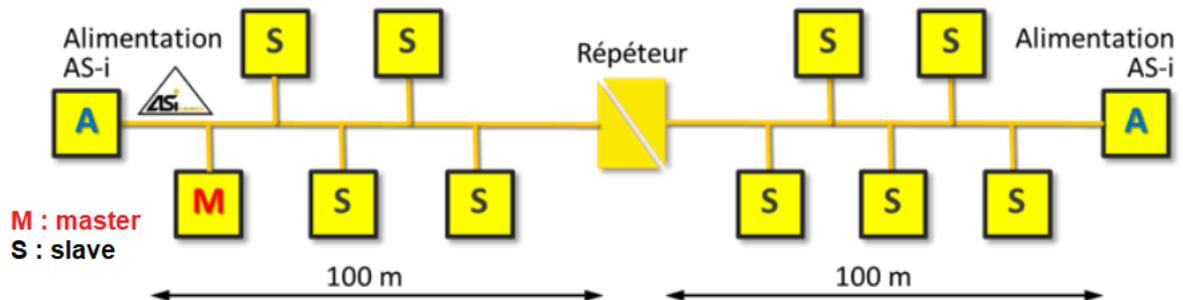
Valeur envoyée = .....

|                 |                                 |                        |                |               |
|-----------------|---------------------------------|------------------------|----------------|---------------|
| .....           | .....                           | .....                  | .....          | XX XX         |
| N° de l'esclave | Instruction d'écriture d'un mot | Adresse de destination | Valeur envoyée | Valeur du CRC |

⇒ **Bus As-i**

Un bus AS-i (en anglais Actuators Sensors Interface) est un bus de terrain de niveau 0 sur la pyramide CIM. Le bus AS-i étant un bus série, il permet de diminuer la longueur de câble nécessaire à l'utilisation d'un grand nombre de capteurs ou de pré-actionneurs. La liaison physique est composée d'une simple paire de fils gainés d'un isolant jaune. Sur ces fils transitent les données ainsi que l'alimentation électrique.

Exemple de topologie

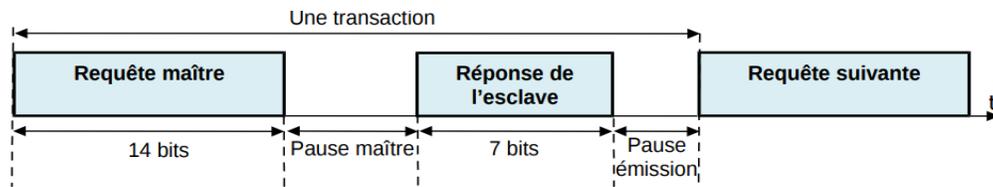


Exemple d'équipements

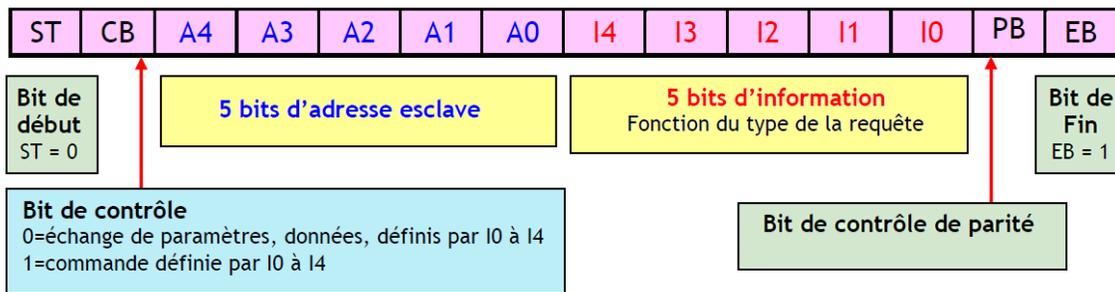


Protocole de communication AS-i

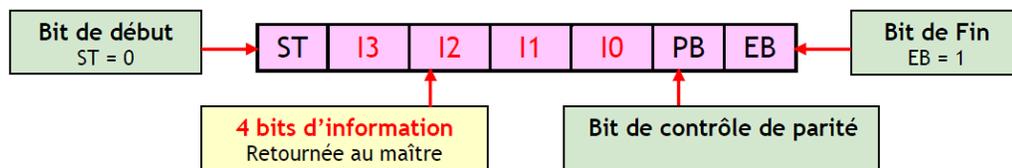
La méthode d'accès du bus AS-i est de type maître-esclave. Le maître interroge chacune des stations esclaves sur le bus, il met à jour les sorties et acquiert l'état des entrées des esclaves.



La trame de requête du **maître** contient toujours **14 bits** organisés de la façon suivante :



La trame de réponse de **l'esclave** contient toujours **7 bits** organisés de la façon suivante :



## Bus USB

La technologie USB (Universal Serial Bus) est une interface d'entrée-sortie beaucoup plus rapide que les ports série standards. Si l'on veut exploiter plusieurs périphériques, on utilise un hub.

### Hub USB



### Types de connecteurs USB



Les avantages de L'USB sont nombreux :

- Faible coût ;
- Indépendance vis à vis des machines hôtes ;
- Hot Plug & Play (c'est à dire branchement et débranchement sans avoir besoin d'arrêter le PC) ;
- Jusqu'à 127 périphériques possibles ;
- Fiabilité et sécurité ...

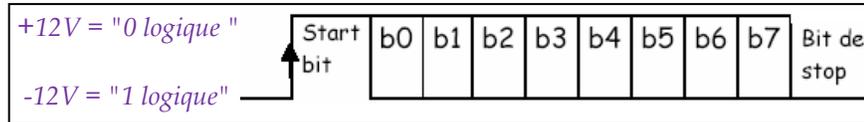
### Brochage des connecteurs USB

| Broche | Désignation      | Couleur du fil |
|--------|------------------|----------------|
| 1      | Alimentation +5V | Rouge          |
| 2      | Données (D-)     | Blanc          |
| 3      | Données (D+)     | Vert           |
| 4      | Masse (GND)      | Noir           |



### Exercice : Décodage d'une frame RS232

Dans une transmission série type RS232, un caractère est émis comme ceci :



La trame d'envoi d'un caractère est donc composée de 10 bits :

- 1 bit de start ;
- 8 bits du caractère émis b0 à b7 ;  
Exemple : si  $(b_7b_6b_5b_4b_3b_2b_1b_0) = (01000111) = (71)_{10}$ . D'après la table ASCII, le caractère transmis est G
- 1 bit de stop.

### Table du code ASCII

| Décimal | Octal | hexa | binaire  | caractère | Décimal | Octal | hexa | binaire  | caractère |
|---------|-------|------|----------|-----------|---------|-------|------|----------|-----------|
| 033     | 041   | 21   | 00100001 | !         | 059     | 073   | 3B   | 00111011 | ;         |
| 034     | 042   | 22   | 00100010 | "         | 060     | 074   | 3C   | 00111100 | <         |
| 035     | 043   | 23   | 00100011 | #         | 061     | 075   | 3D   | 00111101 | =         |
| 036     | 044   | 24   | 00100100 | \$        | 062     | 076   | 3E   | 00111110 | >         |
| 037     | 045   | 25   | 00100101 | %         | 063     | 077   | 3F   | 00111111 | ?         |
| 038     | 046   | 26   | 00100110 | &         | 064     | 100   | 40   | 01000000 | @         |
| 039     | 047   | 27   | 00100111 | '         | 065     | 101   | 41   | 01000001 | A         |
| 040     | 050   | 28   | 00101000 | (         | 066     | 102   | 42   | 01000010 | B         |
| 041     | 051   | 29   | 00101001 | )         | 067     | 103   | 43   | 01000011 | C         |
| 042     | 052   | 2A   | 00101010 | *         | 068     | 104   | 44   | 01000100 | D         |
| 043     | 053   | 2B   | 00101011 | +         | 069     | 105   | 45   | 01000101 | E         |
| 044     | 054   | 2C   | 00101100 | ,         | 070     | 106   | 46   | 01000110 | F         |
| 045     | 055   | 2D   | 00101101 | -         | 071     | 107   | 47   | 01000111 | G         |
| 046     | 056   | 2E   | 00101110 | .         | 072     | 110   | 48   | 01001000 | H         |
| 047     | 057   | 2F   | 00101111 | /         | 073     | 111   | 49   | 01001001 | I         |
| 048     | 060   | 30   | 00110000 | 0         | 074     | 112   | 4A   | 01001010 | J         |
| 049     | 061   | 31   | 00110001 | 1         | 075     | 113   | 4B   | 01001011 | K         |
| 050     | 062   | 32   | 00110010 | 2         | 076     | 114   | 4C   | 01001100 | L         |
| 051     | 063   | 33   | 00110011 | 3         | 077     | 115   | 4D   | 01001101 | M         |
| 052     | 064   | 34   | 00110100 | 4         | 078     | 116   | 4E   | 01001110 | N         |
| 053     | 065   | 35   | 00110101 | 5         | 079     | 117   | 4F   | 01001111 | O         |
| 054     | 066   | 36   | 00110110 | 6         | 080     | 120   | 50   | 01010000 | P         |
| 055     | 067   | 37   | 00110111 | 7         | 081     | 121   | 51   | 01010001 | Q         |
| 056     | 070   | 38   | 00111000 | 8         | 082     | 122   | 52   | 01010010 | R         |
| 057     | 071   | 39   | 00111001 | 9         | 083     | 123   | 53   | 01010011 | S         |
| 058     | 072   | 3A   | 00111010 | :         | 084     | 124   | 54   | 01010100 | T         |
|         |       |      |          |           | 085     | 125   | 55   | 01010101 | U         |
|         |       |      |          |           | 086     | 126   | 56   | 01010110 | V         |
|         |       |      |          |           | 087     | 127   | 57   | 01010111 | W         |
|         |       |      |          |           | 088     | 130   | 58   | 01011000 | X         |
|         |       |      |          |           | 089     | 131   | 59   | 01011001 | Y         |
|         |       |      |          |           | 090     | 132   | 5A   | 01011010 | Z         |

On considère la trame RS232 donnée ci-dessous.

1. Combien de caractères porte la trame ?

La trame est composée de ..... bits, elle est donc porteuse de ..... caractères (puisque l'émission d'un seul caractère nécessite ..... bits)

2. Calculez la durée d'un bit sachant que le débit de transmission est de **9600 bits/s**.

Le débit étant de 9600 bits/s  $\Rightarrow$  la durée d'un bit = .....

3. En déduire la durée totale de la trame.

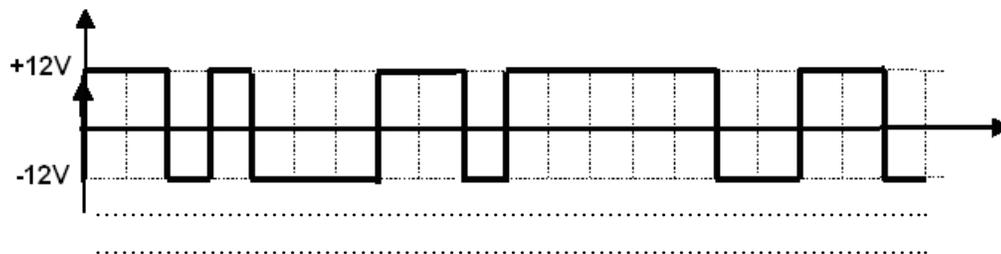
La durée totale de la trame = .....

4. Relever les bits ( $b_7b_6b_5b_4b_3b_2b_1b_0$ ) pour chaque caractère.

Pour le premier caractère, on relève ( $b_7b_6b_5b_4b_3b_2b_1b_0$ ) = (.....) = (.....)<sub>16</sub>

Pour le deuxième caractère, on relève ( $b_7b_6b_5b_4b_3b_2b_1b_0$ ) = (.....) = (.....)<sub>16</sub>

5. Utiliser la table ASCII pour déduire les caractères transmis.



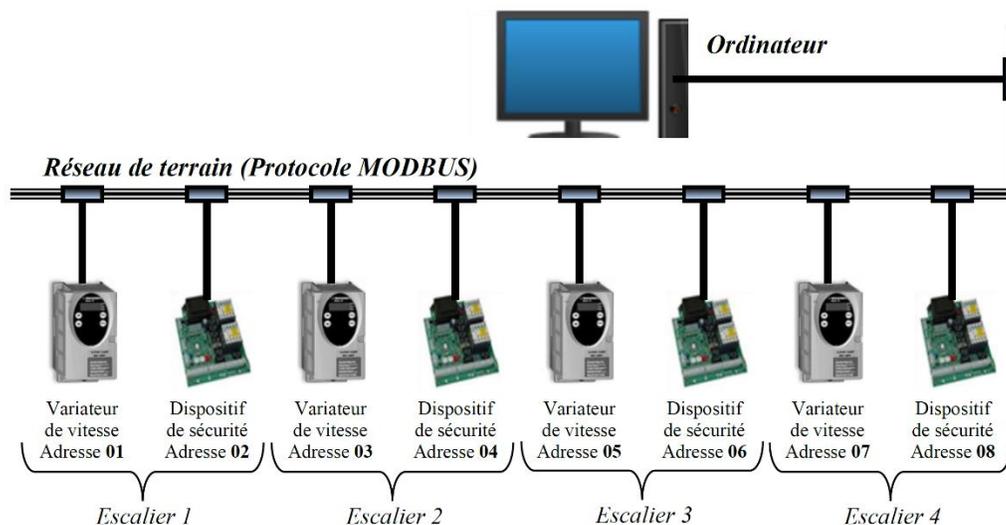
D'après la table ASCII, le premier caractère est " ....." alors que le second est " ....." .





**Exercice : examen national 2023, session normale**
**Tâche 3 : Analyse d'une trame MODBUS**

Un espace commercial utilise quatre escaliers mécaniques supervisés à distance par **ordinateur** et une liaison différentielle de type **RS485**, selon la configuration suivante :



On peut ainsi se renseigner sur l'état général des quatre escaliers, comme on peut les contrôler à distance.

Lire le document ressources **DRES 06** avant de répondre aux questions.

On désire vérifier à l'aide d'un analyseur de trame **MODBUS**, la configuration des variateurs **ALTIVAR** qui pilotent les différents moteurs des 4 escaliers.

Ces variateurs gèrent la vitesse des moteurs (accélération, décélération, grande vitesse, petite vitesse...) d'après les paramètres stockés sous forme de mots (double octet) dans leurs registres de configuration (mémoire interne).

On a relevé sur le bus les trames suivantes codées **en hexadécimal** :

**Requête du maître (Ordinateur)**

|    |    |      |      |     |
|----|----|------|------|-----|
| 05 | 03 | 2C92 | 0002 | CRC |
|----|----|------|------|-----|

**Réponse de l'esclave (Variateur)**

|    |    |    |      |      |     |
|----|----|----|------|------|-----|
| 05 | 03 | 04 | 0064 | 0096 | CRC |
|----|----|----|------|------|-----|

2C92 = 11410 en décimal

0064 = 100 en décimal

0096 = 150 en décimal

Répondre aux questions suivantes en cochant les bonnes réponses.

**Q.40-** Avec quel esclave le maître entame-t-il un **dialogue** ?

**1 pt**

**Q.41-** Quel est le **but** de la requête du maître ?

**1 pt**

**Q.42-** Quels sont les **paramètres** lus ou écrits ? donner alors, leurs **valeurs** en Hz ?

**2 pts**

On souhaite changer le paramètre **SP2** de 100 (10 Hz) à 250 (25 Hz).

**Q.43-** Elaborer la trame **MODBUS** (requête du maître et réponse de l'esclave) permettant de configurer à partir du maître le paramètre **SP2** (2<sup>ème</sup> vitesse) à 250 (25 Hz) du variateur de vitesse de l'**escalier 4**.

**2 pts**

## MODBUS

DRES 06

### 1. Présentation

Le tableau suivant indique quelques fonctions du protocole de communication **MODBUS** entre le maître (Ordinateur) et l'esclave (**Altivar 31**).

| Code fonction (décimal) | Nom des fonctions            | Valeur maxi de N |
|-------------------------|------------------------------|------------------|
| 3                       | Lecture de N mots de sortie  | 29 mots maxi     |
| 6                       | Ecriture d'un mot de sortie  |                  |
| 16                      | Ecriture de N mots de sortie | 27 mots maxi     |

### 2. Ecriture d'un mot de sortie : fonction 6 (06<sub>H</sub>)

Cette fonction permet d'écrire un mot. Les formats des trames dans ce cas sont identiques.

#### Requête du maître et réponse de l'esclave

| Adresse de l'esclave | N° fonction | Adresse du mot |    | Valeur du mot |    | CRC      |    |
|----------------------|-------------|----------------|----|---------------|----|----------|----|
|                      |             | PF             | Pf | PF            | Pf | PF       | Pf |
| 1 octet              | 1 octet     | 2 octets       |    | 2 octets      |    | 2 octets |    |

**N.B:**

PF = poids fort      Pf = poids faible      1 mot = 2 octets = 2 x 8 bits = 16 bits

CRC (Cyclic Redundancy Code) : 2 octets de contrôle de l'intégrité de la trame (non exploité dans le sujet).

### 3. Lecture de N mots de sortie : fonction 3 (03<sub>H</sub>)

Cette fonction permet de lire un ou plusieurs mots, la lecture s'effectue à partir de l'adresse du premier mot puis par incrémentation de l'adresse.

#### Requête du maître

| Adresse de l'esclave | N° fonction | Adresse du premier mot |    | Nombre de mots |    | CRC      |    |
|----------------------|-------------|------------------------|----|----------------|----|----------|----|
|                      |             | PF                     | Pf | PF             | Pf | PF       | Pf |
| 1 octet              | 1 octet     | 2 octets               |    | 2 octets       |    | 2 octets |    |

#### Réponse de l'esclave

| Adresse de l'esclave | N° fonction | Nombre d'octets lus | Valeur premier mot |    | Valeur dernier mot |    | CRC      |    |
|----------------------|-------------|---------------------|--------------------|----|--------------------|----|----------|----|
|                      |             |                     | PF                 | Pf | PF                 | Pf | PF       | Pf |
| 1 octet              | 1 octet     | 1 octet             | 2 octets           |    | 2 octets           |    | 2 octets |    |

### 4. Liste de quelques mots configurables de l'ALTIVAR 31 par trame MODBUS

| Adresse MODBUS                             | Code       | Nom / Description / Valeurs possibles   |
|--|------------|---|
| 11410 <sub>10</sub> = 2C92 <sub>Hexa</sub> | <b>SP2</b> | <b>2<sup>ème</sup> vitesse présélectionnée</b> <ul style="list-style-type: none"> <li>Unité : 0,1 Hz</li> <li>Réglage usine 100 : (100 × 0,1 = 10 Hz)</li> <li>Plage de réglage : 0 à 5000</li> </ul> |
| 11411 <sub>10</sub> = 2C93 <sub>Hexa</sub> | <b>SP3</b> | <b>3<sup>ème</sup> vitesse présélectionnée</b> <ul style="list-style-type: none"> <li>Unité : 0,1 Hz</li> <li>Réglage usine 150 : (150 × 0,1 = 15 Hz)</li> <li>Plage de réglage : 0 à 5000</li> </ul> |
| 11412 <sub>10</sub> = 2C94 <sub>Hexa</sub> | <b>SP4</b> | <b>4<sup>ème</sup> vitesse présélectionnée</b> <ul style="list-style-type: none"> <li>Unité : 0,1 Hz</li> <li>Réglage usine 200 : (200 × 0,1 = 20 Hz)</li> <li>Plage de réglage : 0 à 5000</li> </ul> |

Q.40 -

- Esclave d'adresse 01 : Variateur de vitesse du moteur de l'escalier 1
- Esclave d'adresse 03 : Variateur de vitesse du moteur de l'escalier 2
- Esclave d'adresse 05 : Variateur de vitesse du moteur de l'escalier 3
- Esclave d'adresse 07 : Variateur de vitesse du moteur de l'escalier 4

Q.41 -

- Lecture de 2 mots de sortie (adresses : 2C92<sub>H</sub> et 2C93<sub>H</sub>)
- Ecriture de 2 mots de sortie (adresses : 2C92<sub>H</sub> et 2C93<sub>H</sub>)
- Ecriture d'un mot de sortie (adresse : 2C92<sub>H</sub>)

Q.42 -

Paramètres lus ou écrits :

- 1<sup>ère</sup> vitesse présélectionnée (SP1)
- 2<sup>ème</sup> vitesse présélectionnée (SP2)
- 3<sup>ème</sup> vitesse présélectionnée (SP3)
- 4<sup>ème</sup> vitesse présélectionnée (SP4)

Valeurs des paramètres lus ou écrits en Hz :

- SP1 = 0 Hz
- SP2 = 10 Hz
- SP3 = 15 Hz
- SP4 = 20 Hz
- SP1 = 25 Hz
- SP2 = 100 Hz
- SP3 = 150 Hz
- SP4 = 200 Hz

Q.43 -

$$2^{\text{ème}} \text{ vitesse : } 25 \text{ Hz} \rightarrow SP2 = 25/0,1 \rightarrow SP2 = 250$$

Requête du maître (Ordinateur) et réponse de l'esclave (Variateur)

|                             |                             |                               |                               | CRC |
|-----------------------------|-----------------------------|-------------------------------|-------------------------------|-----|
| <input type="checkbox"/> 01 | <input type="checkbox"/> 03 | <input type="checkbox"/> 2C92 | <input type="checkbox"/> FA00 |     |
| <input type="checkbox"/> 03 | <input type="checkbox"/> 06 | <input type="checkbox"/> 2C93 | <input type="checkbox"/> 0064 |     |
| <input type="checkbox"/> 05 | <input type="checkbox"/> 0A | <input type="checkbox"/> 2C94 | <input type="checkbox"/> 6400 |     |
| <input type="checkbox"/> 07 | <input type="checkbox"/> 10 | <input type="checkbox"/> 2C95 | <input type="checkbox"/> 00FA |     |