



Matière :	Science de l'Ingénieur - <b>A.T.C</b> -	Pr.MAHBAB
Section :	Sciences et Technologies Électriques	<b>Système n° 2</b>

### ❖ **Sujet :**

- ◆ Le sujet comporte au total **19** pages.
- ◆ Le sujet comporte 3 types de documents :

📄 **Pages 01 à 04** : Socle du sujet comportant les situations d'évaluation (SEV).

📄 **Pages 05 à 10** : Documents ressources portant la mention.

**DRES XX**

📄 **Pages 11 à 19** : Documents réponses portant la mention.

**DREP XX**

19 pages

### ❖ **7 Fiches cours :**

- ◆ Fiche cours n°3 « **Capteurs de températures** »
- ◆ Fiche cours n°5 « **Les mémoires** »
- ◆ Fiche cours n°6 « **Les interfaces** »
- ◆ Fiche cours n°7 « **Le microprocesseur** »
- ◆ Fiche cours n°8 « **Le PIC 16F84** »
- ◆ Fiche cours n°9 « **Langage assembleur** »
- ◆ Fiche cours n°10 « **Utilisation des ports A et B** »

19 pages

## SYSTÈME D'ENCAISSAGE

## SYSTEME D'ENCAISSAGE

### 1- DESCRIPTION DU SYSTEME :

Le système est principalement composé (voir **DRES 01 page 05**) de :

- ❖ Un tapis roulant amenant les boîtes à encaisser.
- ❖ Un vérin  $C_1$  muni d'un plateau horizontal  $p_1$  servant à transférer les boites en rangées de trois.
- ❖ Un vérin  $C_2$  muni d'un plateau vertical  $p_2$  servant à pousser, par bloc de 3 rangées, les boîtes dans le carton.
- ❖ Un vérin  $C_3$  permettant de maintenir le carton pendant le chargement, une fois ce carton est rempli, le vérin  $C_3$  le transfère sur les rouleaux transporteurs.
- ❖ Des rouleaux transporteurs permettant l'évacuation des cartons pleins.

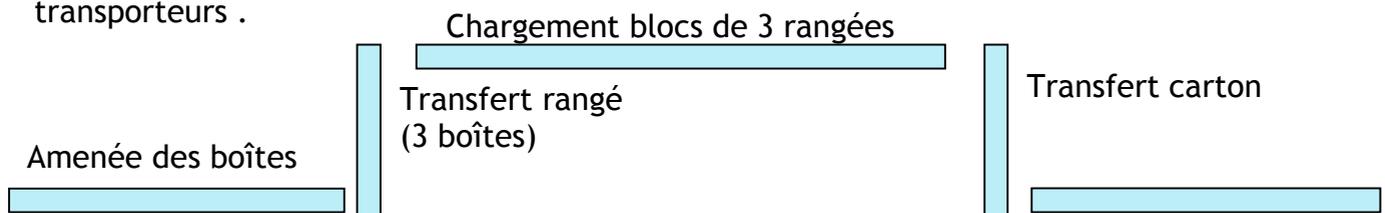
### 2- FONCTIONNEMENT :

#### 2.1- Amenée des boîtes :

Les boîtes sont amenées par l'intermédiaire d'un tapis roulant .Ce tapis est entraîné par un moteur  $M_1$  asynchrone triphasé à rotor en court circuit commandé par un discontacteur tripolaire  $KM_1$ .

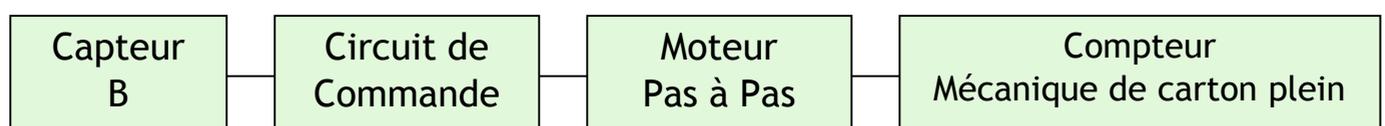
#### 2.2- Transfert des boîtes :

- ❖ L'opérateur place un carton vide sur la machine, sa présence est détectée par le capteur infra rouge  $P$  et fait démarrer le cycle par action sur un interrupteur départ cycle (Dcy).
- ❖ Les boîtes arrivent sur le plateau horizontal du vérin  $C_1$ . L'action des boîtes sur le capteur  $v_1$  provoque le transfert d'une rangée de boîtes grâce au vérin  $C_1$ . Cette rangée, une fois soulevée, est empilée devant le vérin  $C_2$  sur un support élastique. Ce support est constitué de deux lames ressorts qui maintiennent les boites devant le plateau du vérin  $C_2$ .
- ❖ Cette opération se répète jusqu'à l'obtention d'un bloc de trois rangées, l'action sur le capteur  $v_2$  provoque le chargement du bloc dans le carton grâce au vérin  $C_2$ . A la suite du deuxième chargement, un capteur  $v_3$  , actionné sous l'effet du poids , provoque grâce au vérin  $C_3$  le transfert du carton plein en le faisant pivoter jusqu'au chemin des rouleaux transporteurs .



#### 2.3- Evacuation et comptage :

Le transfert des cartons sur les rouleaux transporteurs, met le moteur  $M_2$ , commandé par un discontacteur tripolaire  $KM_2$ , en marche, ce qui permet d'évacuer le carton. Le passage du carton évacué devant un capteur infrarouge  $B$ , arrête le moteur  $M_2$  et incrémente un compteur mécanique piloté par un moteur pas à pas suivant la synoptique suivante :



SEV 1	<b>ANALYSE FONCTIONNELLE</b>	/ pts
-------	------------------------------	-------

RESSOURCES A EXPLOITER : DRES 01 page 05,  
'Description' et 'Fonctionnement' page 01.

1. Sur le document DREP 01 page 11, compléter l'actigramme A-0 du système.
2. Le F.A.S.T. du DREP 01 page 11 définit la fonction globale (FG) du système étudié. Indiquer sur ce document, pour chacune des fonctions principales le processus qui lui est associé.

SEV 2	<b>ETUDE PARTIELLE DE CHAÎNE D'INFORMATION</b>	/ pts
-------	--	-------

Tâche 1	<b>GRAF CET</b>	/ pts
---------	-----------------	-------

1. En se référant au DRES 02 page 06 et au document de la page 01, compléter sur le document DREP 02 page 12, le GRAFCET du point de vue système traduisant le fonctionnement normal du système.
2. En se référant au DRES 02 page 06 et au GRAFCET du point de vue système, compléter sur le document DREP 03 page 13, le GRAFCET du point de vue partie commande traduisant le fonctionnement normal du système.

Tâche 2	<b>ORGANIGRAMME et PROGRAMME</b>	/ pts
---------	----------------------------------	-------

1. En se référant au DRES 02 page 06 et au GRAFCET du point de vue commande, compléter sur le document DREP 04 page 14, l'organigramme traduisant le fonctionnement normal du système.
2. En se référant à l'organigramme traduisant le fonctionnement normal du système et aux documents DRES 03 et 04 pages 07 et 08, compléter le programme du document DREP 05 page 15.

Tâche 3	<b>Dispositif de sécurité</b>	/ pts
---------	-------------------------------	-------

Le produit à encaisser est sensible à la température, pour cela on envisage d'équiper le système d'un détecteur de température. Si la température dépasse un seuil de référence, un dispositif met en marche une alarme et arrête le système immédiatement.  
Sur le document DRES 06 page 10, on donne le schéma synoptique du système de sécurité.

**1. Acquisition et amplification de température :**

Répondre sur le document DREP 06 page 16

L'acquisition de la température se fait grâce au capteur LM135, voir DRES 06 page 10.

- 1.1. Donner le nom du montage autour de l'amplificateur opérationnel.
- 1.2. Donner l'expression de  $V$  en fonction de la température en  $^{\circ}K$ .
- 1.3. Sachant que l'AOP est supposé idéal, donner l'expression de la tension de sortie  $V_S$  en fonction des entrées  $V$  et  $V_{ref}$ .

- 1.4. En déduire l'expression de  $V_s$  en fonction de  $T$  (température en °k).
- 1.5. Sachant que  $V_{ref} = 2.73$  V donner alors l'expression de  $V_s$  en fonction de  $t$  (température en °c).
- 1.6. Calculer  $R_2$  pour avoir une sensibilité de  $50\text{mV}/^\circ\text{c}$ .
- 1.7. Quel est alors le rôle du montage.
- 1.8. Donner l'expression de  $V_{ref}$  en fonction de  $R_3$ ,  $R_4$  et  $V_{cc}$  sachant que le courant dans la résistance  $R_1$  est très inférieur au courant dans la résistance  $R_4$ .
- 1.9. Calculer alors  $R_4$  pour  $V_{cc} = 12$  V et  $R_3 = 4.7$  k $\Omega$

## 2. Dispositif de sécurité :

Répondre sur le document DREP 02 page 12

La tension  $V_s$ , image de la température est comparée à la tension  $V_c$ , image de la température consigne ( $25^\circ\text{c}$ ) , voir DRES 06 page 10.

- 2.1. L'amplificateur opérationnel fonctionne en régime non linéaire. Pourquoi ?
- 2.2. Pour  $V_s < V_c$  donner l'expression numérique de  $U$ , l'état du transistor T (saturé ou bloqué) et l'état des contact  $K_1$  et  $K_2$  (fermé ou ouvert).
- 2.3. Pour  $V_s \geq V_c$  donner l'expression numérique de  $U$ , l'état du transistor T (saturé ou bloqué) et l'état des contact  $K_1$  et  $K_2$  (fermé ou ouvert).
- 2.4. Calculer alors  $V_c$  sachant que la température consigne est égale à  $25^\circ\text{c}$ .

SEV 3

## ETUDE DE LA PARTIE OPERATIVE

/ pts

RESSOURCES A EXPLOITER : DRES 05 page 09

Tâche 1

Schéma cinématique

/ pts

Le document DRES 05 page 09, représente le tambour moteur du tapis roulant. L'arbre (1) du moteur électrique transmet son mouvement de rotation au tambour (20) grâce au réducteur d'engrenage constitué par les deux couples 1-28 et 27-24.

1. Compléter le tableau du DREP 07 page 17 en indiquant les éléments correspondants aux différentes classes d'équivalence B, C et D (groupe d'éléments cinématiquement liés).

Tâche 2

Calcul de prédétermination

/ pts

Répondre sur le document DREP 08 page 18

Considérons le réducteur à engrenages représenté par le schéma du DREP 07 page 17 et par le dessin du DRES 05 page 09.

On donne :

- ❖ Les nombres de dents :  $Z_1 = 17$  dents ;  $Z_{28} = 125$  dents ;  $Z_{27} = 20$  dents ;  $Z_{24} = 55$  dents.
- ❖ Vitesse de rotation du moteur ( $M_1$ ) :  $N_m = 1500$  tr/mn.
- ❖ Diamètre du tambour (20) :  $D = 220$  mm.

1. Calculer le rapport de réduction ( $r$ ) entre l'arbre moteur et le tambour (20).
2. Calculer la vitesse de rotation du tambour (20).
3. Calculer la vitesse de translation des boîtes transférées par le tapis roulant.

Tâche 3

Cotation fonctionnelle

/ pts

Répondre sur le document **DREP 08 page 18**

1. Tracer la chaîne de cotes relative à la condition B
2. Calculer la valeur de la cote A9 sachant que :  $0,2 \leq A \leq 0,5$

Tâche 3

Etude de conception

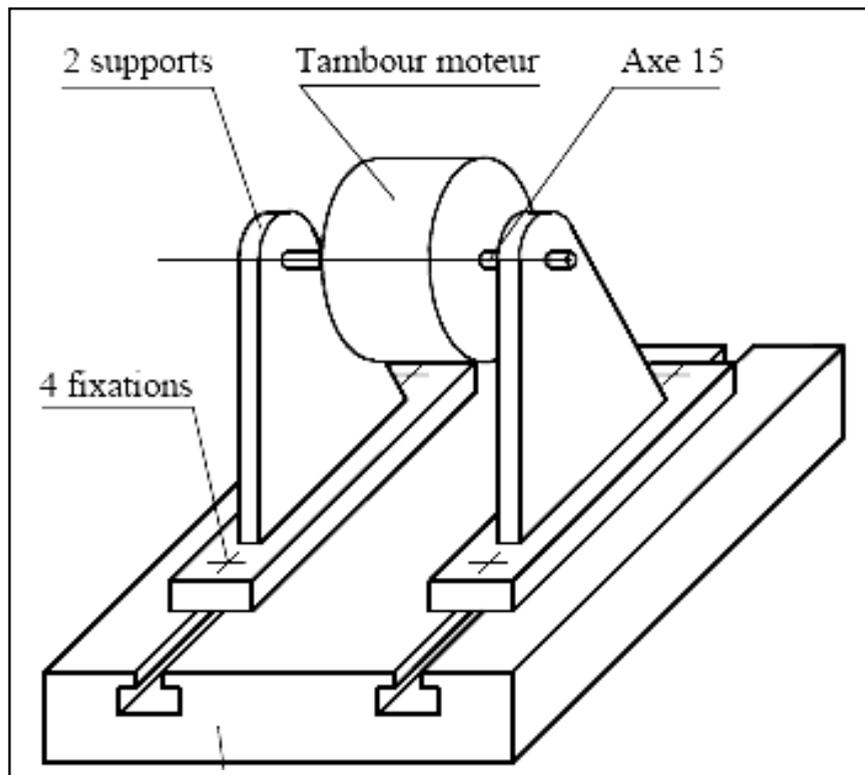
/ pts

Répondre sur le document **DREP 09 page 19**

Le tambour moteur représenté sur le document DRES 05 page xx, est maintenu par les deux supports.  
Les deux rainures en T de la table de la machine permettent de tendre le tapis roulant (voir dessin ci-contre).

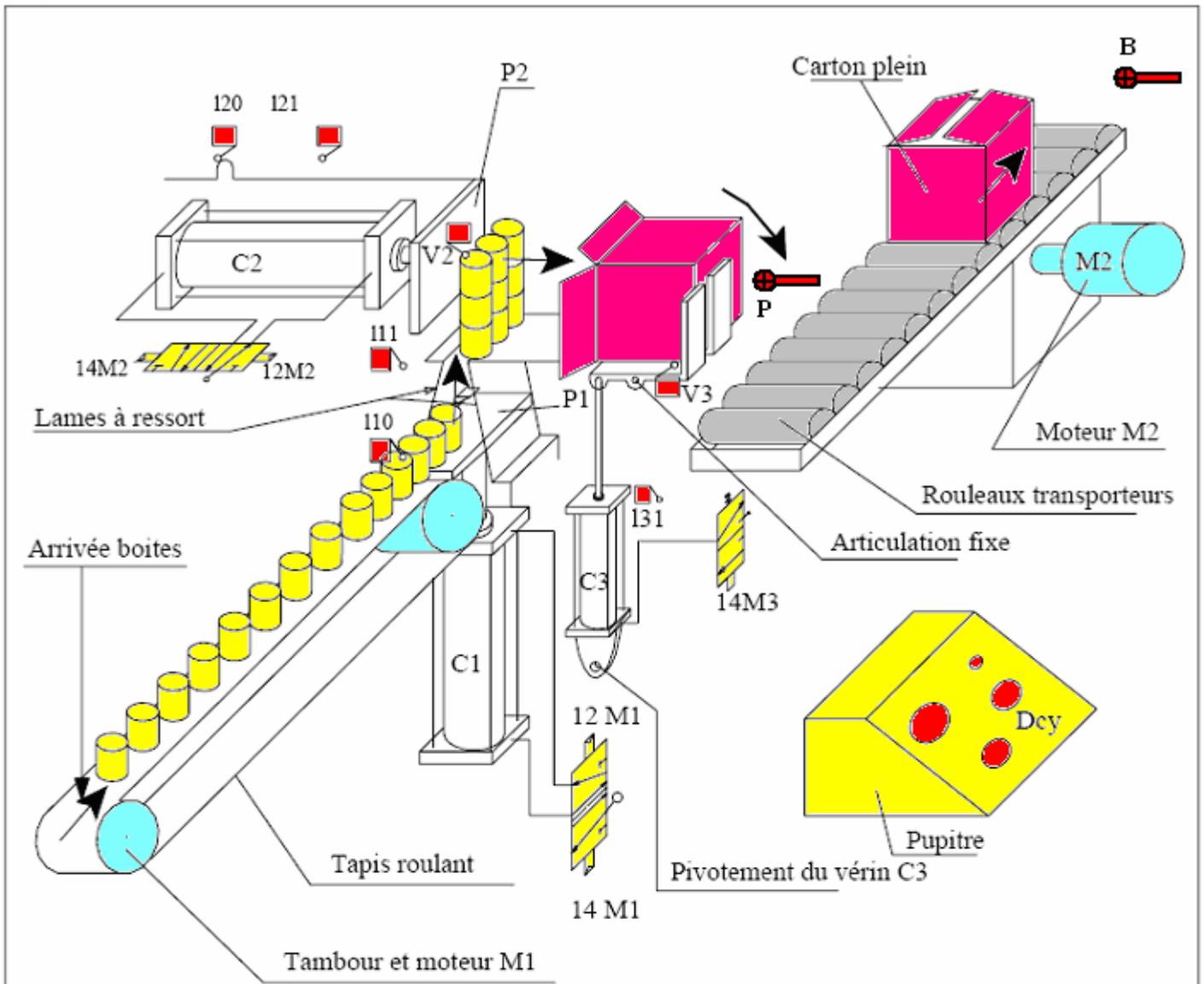
Compléter sur le dessin document **DREP 09 page 19**, échelle 1:2

1. La liaison encastrement de l'arbre (15) avec le support.
2. La liaison encastrement du support avec la table de la machine.  
Cette liaison doit permettre le réglage de la tension du tapis roulant.

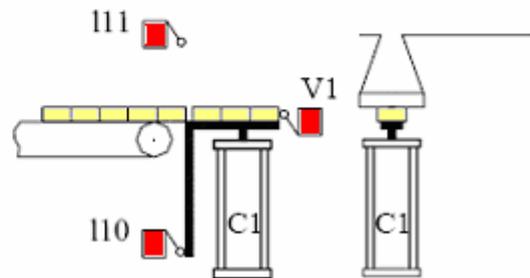
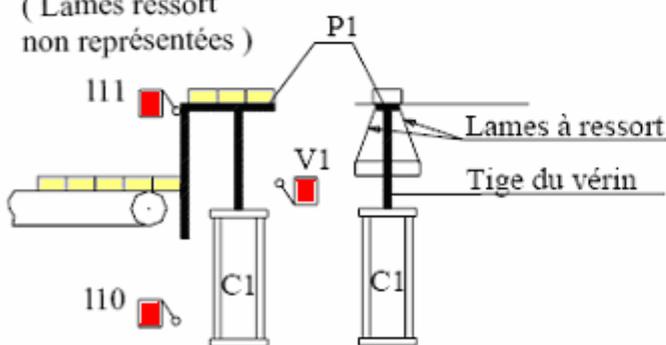


DRES 01

SYSTEME D'ENCAISSAGE



( Lames ressort non représentées )



1 Boîte

1 Rangée

1 Bloc

1 ensemble de 2 blocs

1 ensemble de 2 blocs

**DRES 02**

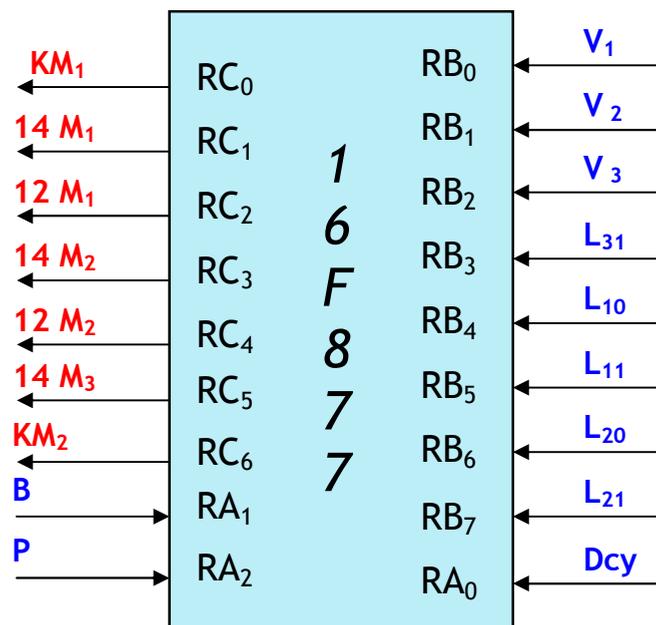
## IDENTIFICATION DES ELEMENTS DU SYSTEME

Action	Actionneur	Préactionneurs
Préparer une rangée	M <sub>1</sub>	KM <sub>1</sub>
Préparer un bloc	Vérin C <sub>1</sub>	14 M <sub>1</sub>
		12 M <sub>1</sub>
Remplir le carton	Vérin C <sub>2</sub>	14 M <sub>2</sub>
		12 M <sub>2</sub>
Basculer le carton	Vérin C <sub>3</sub>	14 M <sub>3</sub>
Evacuer le carton	M <sub>2</sub>	KM <sub>2</sub>

Compte-rendu et Ordre	Capteur	
Départ cycle	Bouton poussoir	Dcy
Présence carton	Détecteur photoélectrique	P
Rangée préparée	Détecteur mécanique à levier	V <sub>1</sub>
Bloc préparé	Détecteur mécanique à levier	V <sub>2</sub>
Carton plein	Détecteur mécanique à levier	V <sub>3</sub>
Carton évacué	Détecteur photoélectrique	B
Carton basculé	Détecteur mécanique à levier	L <sub>31</sub>
Position du plateau P1	Détecteur mécanique à levier	L <sub>10</sub>
	Détecteur mécanique à levier	L <sub>11</sub>
Position du plateau P2	Détecteur mécanique à levier	L <sub>20</sub>
	Détecteur mécanique à levier	L <sub>21</sub>

## COMMANDE DU SYSTEME

La commande du système est faite par le PIC 16F84



**DRES 03**

## Plan mémoire du 16F877

Le plan mémoire des données et des registres internes est découpé en 4 zones ou Bank de 128 octets, pour accéder à une zone il faut positionner les bits **RP<sub>0</sub>** (bit 5) et **RP<sub>1</sub>** (bit 6) du registre **STATUS**.

RP <sub>1</sub> : RP <sub>0</sub>	BANK sélectionnée
00	Bank 0 de 000 <sub>H</sub> à 07F <sub>H</sub>
01	Bank 1 de 080 <sub>H</sub> à 0FF <sub>H</sub>
10	Bank 2 de 100 <sub>H</sub> à 17F <sub>H</sub>
11	Bank 3 de 180 <sub>H</sub> à 1FF <sub>H</sub>

Indirect Addr	000 <sub>H</sub>	Indirect Addr	080 <sub>H</sub>	Indirect Addr	100 <sub>H</sub>	Indirect Addr	180 <sub>H</sub>
TMRO	001 <sub>H</sub>	OPTION	081 <sub>H</sub>	TMRO	101 <sub>H</sub>	OPTION	181 <sub>H</sub>
PCL	002 <sub>H</sub>	PCL	082 <sub>H</sub>	PCL	102 <sub>H</sub>	PCL	182 <sub>H</sub>
STATUS	003 <sub>H</sub>	STATUS	083 <sub>H</sub>	STATUS	103 <sub>H</sub>	STATUS	183 <sub>H</sub>
FSR	004 <sub>H</sub>	FSR	084 <sub>H</sub>	FSR	104 <sub>H</sub>	FSR	184 <sub>H</sub>
PORTA	005 <sub>H</sub>	TRISA	085 <sub>H</sub>		105 <sub>H</sub>		185 <sub>H</sub>
PORTB	006 <sub>H</sub>	TRISB	086 <sub>H</sub>	PORTB	106 <sub>H</sub>	TRISB	186 <sub>H</sub>
PORTC	007 <sub>H</sub>	TRISC	087 <sub>H</sub>		107 <sub>H</sub>		187 <sub>H</sub>
PORTD	008 <sub>H</sub>	TRISD	088 <sub>H</sub>		108 <sub>H</sub>		188 <sub>H</sub>
PORTE	009 <sub>H</sub>	TRISE	089 <sub>H</sub>		109 <sub>H</sub>		189 <sub>H</sub>
PCLATCH	00A <sub>H</sub>	PCLATCH	08A <sub>H</sub>	PCLATCH	10A <sub>H</sub>	PCLATCH	18A <sub>H</sub>
INTCON	00B <sub>H</sub>	INTCON	08B <sub>H</sub>	INTCON	10B <sub>H</sub>	INTCON	18B <sub>H</sub>
PIR1	00C <sub>H</sub>	PIE1	08C <sub>H</sub>	EEDATA	10C <sub>H</sub>	EECON1	18C <sub>H</sub>
PIR2	00D <sub>H</sub>	PIE2	08D <sub>H</sub>	EEADR	10D <sub>H</sub>	EECON2	18D <sub>H</sub>
TMR1L	00E <sub>H</sub>	PCON	08E <sub>H</sub>	EEDATH	10E <sub>H</sub>	Reserved	18E <sub>H</sub>
TMRL2	00F <sub>H</sub>		08F <sub>H</sub>	EEADRH	10F <sub>H</sub>	Reserved	18F <sub>H</sub>
T1CON	010 <sub>H</sub>		090 <sub>H</sub>	General Purpose Register 16 Bytes	110 <sub>H</sub>	General Purpose Register 16 Bytes	190 <sub>H</sub>
TMR2	011 <sub>H</sub>	SSPCON2	091 <sub>H</sub>		111 <sub>H</sub>		191 <sub>H</sub>
T2CON	012 <sub>H</sub>	PR2	092 <sub>H</sub>		112 <sub>H</sub>		192 <sub>H</sub>
SSPBUF	013 <sub>H</sub>	SSPADD	093 <sub>H</sub>		113 <sub>H</sub>		193 <sub>H</sub>
SSPCON	014 <sub>H</sub>	SSPSTAT	094 <sub>H</sub>		114 <sub>H</sub>		194 <sub>H</sub>
CCPR1L	015 <sub>H</sub>		095 <sub>H</sub>		115 <sub>H</sub>		195 <sub>H</sub>
CCPR1H	016 <sub>H</sub>		096 <sub>H</sub>		116 <sub>H</sub>		196 <sub>H</sub>
CCP1CON	017 <sub>H</sub>		097 <sub>H</sub>		117 <sub>H</sub>		197 <sub>H</sub>
RCSTA	018 <sub>H</sub>	TXSTA	098 <sub>H</sub>		118 <sub>H</sub>		198 <sub>H</sub>
TXREG	019 <sub>H</sub>	SPBRG	099 <sub>H</sub>		119 <sub>H</sub>		199 <sub>H</sub>
RCREG	01A <sub>H</sub>		09A <sub>H</sub>		11A <sub>H</sub>		19A <sub>H</sub>
CCPR2L	01B <sub>H</sub>		09B <sub>H</sub>		11B <sub>H</sub>		19B <sub>H</sub>
CCPR2H	01C <sub>H</sub>		09C <sub>H</sub>		11C <sub>H</sub>		19C <sub>H</sub>
CCP2CON	01D <sub>H</sub>		09D <sub>H</sub>		11D <sub>H</sub>		19D <sub>H</sub>
ADRESH	01E <sub>H</sub>	ADRESL	09E <sub>H</sub>		11E <sub>H</sub>		19E <sub>H</sub>
ADCON0	01F <sub>H</sub>	ADCON1	09F <sub>H</sub>		11F <sub>H</sub>		19F <sub>H</sub>
General Purpose Register 96 Bytes	020 <sub>H</sub>	General Purpose Register 80 Bytes	0A0 <sub>H</sub>	General Purpose Register 80 Bytes	120 <sub>H</sub>	General Purpose Register 80 Bytes	1A0 <sub>H</sub>
	07F <sub>H</sub>		0EF <sub>H</sub>		16F <sub>H</sub>		1EF <sub>H</sub>
BANK 0		BANK 1		BANK 2		BANK 3	
		Accesses 070 <sub>H</sub> - 07E <sub>H</sub>	0F0 <sub>H</sub>	Accesses 070 <sub>H</sub> - 07E <sub>H</sub>	170 <sub>H</sub>	Accesses 070 <sub>H</sub> - 07E <sub>H</sub>	1F0 <sub>H</sub>
			0FF <sub>H</sub>		17F <sub>H</sub>		1FF <sub>H</sub>

**DRES 04**

## Résumé du jeu d'instruction du 16F877

Mnemonic, operands	Description	Cycles	14-bit opcode				Status affected	Notes	
			MSB	LSB					
<b>BYTE ORIENTED FILE REGISTER OPERATIONS</b>									
ADDWF	f,d	Add W and f	1	00	0111	dfff	ffff	C, DC, Z	1,2
ANDWF	f,d	AND W with f	1	00	0101	dfff	ffff	Z	1,2
CLRF	f	Clear f	1	00	0001	1fff	ffff	Z	2
CLRFW	-	Clear W	1	00	0001	0xxx	xxxx	Z	
COM	f,d	Complement f	1	00	1001	dfff	ffff	Z	1,2
DECF	f,d	Decrement f	1	00	0011	dfff	ffff	Z	1,2
DECFSZ	f,d	Decrement f, skip if 0	1(2)	00	1011	dfff	ffff		1,2,3
INCF	f,d	Increment f	1	00	1010	dfff	ffff	Z	1,2
INCFSZ	f,d	Increment f, skip if 0	1(2)	00	1111	dfff	ffff		1,2,3
IORWF	f,d	Inclusive OR W with f	1	00	0100	dfff	ffff	Z	1,2
MOVF	f,d	Move f	1	00	1000	dfff	ffff	Z	1,2
MOVWF	f	Move W to f	1	00	0000	1fff	ffff		
NOP	-	No operation	1	00	0000	0xx0	0000		
RLF	f,d	Rotate Left f through Carry	1	00	1101	dfff	ffff	C	1,2
RRF	f,d	Rotate Right f through Carry	1	00	1100	dfff	ffff	C	1,2
SUBWF	f,d	Subtract W from f	1	00	0010	dfff	ffff	C, DC, Z	1,2
SWAPF	f,d	Swap nibbles in f	1	00	1110	dfff	ffff		1,2
XORWF	f,d	Exclusive OR W with f	1	00	0110	dfff	ffff	Z	1,2
<b>BIT ORIENTED FILE REGISTER OPERATIONS</b>									
BCF	f,d	Bit clear f	1	01	00bb	bfff	ffff		1,2
BSF	f,d	Bit set f	1	01	01bb	bfff	ffff		1,2
BTFSC	f,d	Bit test f, skip if clear	1(2)	01	10bb	bfff	ffff		3
BTFSS	f,d	Bit test f, skip if set	1(2)	01	11bb	bfff	ffff		3
<b>LITERAL AND CONTROL OPERATIONS</b>									
ADDLW	k	Add literal and W	1	11	111x	kkkk	kkkk	C, DC, Z	
ANDLW	k	AND literal With W	1	11	1001	kkkk	kkkk	Z	
CALL	k	Call subroutine	2	10	0kkk	kkkk	kkkk		
CLRWDI	-	Clear watchdog Timer	1	00	0000	0101	0100	TO, PD	
GOTO	k	Go to address	2	10	1kkk	kkkk	kkkk		
IORLW	k	Inclusive OR literal With W	1	11	1000	kkkk	kkkk	Z	
MOVLW	k	Move literal to W	1	11	00xx	kkkk	kkkk		
RETFIE	-	Return from interrupt	2	00	0000	0000	1001		
RETLW	k	Return with literal to W	2	11	01xx	kkkk	kkkk		
RETURN	-	Return from subroutine	2	00	0000	0000	1000		
SLEEP	-	Go into standby mode	1	00	0000	0110	0011	TO, PD	
SUBLW	k	Subtract W from literal	1	11	110x	kkkk	kkkk	C, DC, Z	
XORLW	k	Exclusive OR literal With W	1	11	1010	kkkk	kkkk	Z	

### Configuration des PORTs

Tous les ports sont pilotés par deux registres :

- ❖ Le registre de **PORTx**, si le **PORT x** ou certaines lignes de **PORT x** sont configurées en sortie, ce registre détermine l'état logique des sorties.
- ❖ Le registre **TRISx**, c'est le registre de direction. Il détermine si le **PORTx** ou certaines lignes de Port sont en entrée ou en sortie. L'écriture d'un **1** logique correspond à une **entrée** (1 comme Input) et l'écriture d'un **0** logique correspond à une **sortie** (0 comme Output).

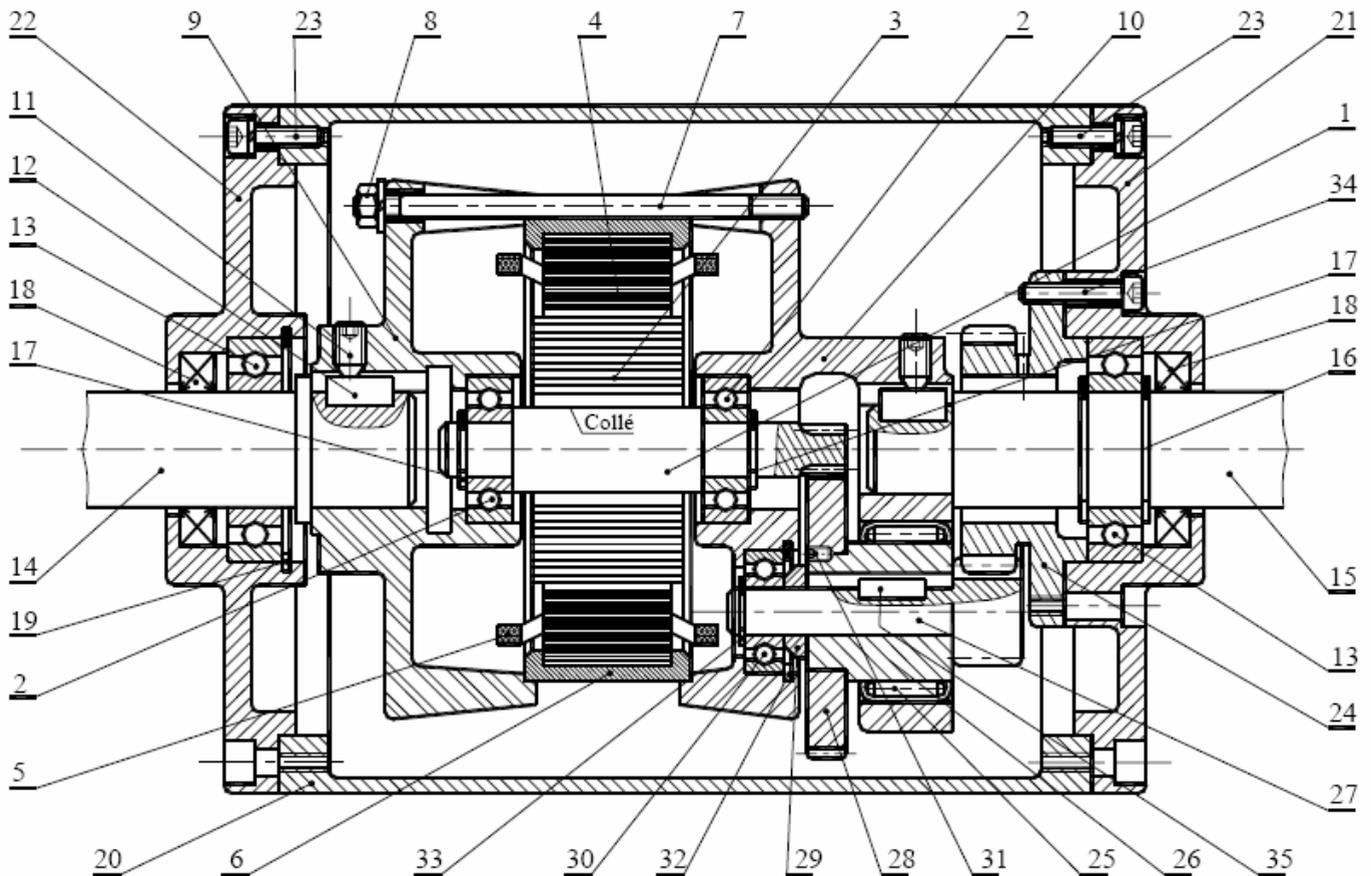
#### Remarque :

Les registres **TRISx** appartiennent à la **BANQUE 1** des **SFR**. Lors de l'initialisation du  $\mu\text{C}$  il ne faut pas oublier de changer de page mémoire pour les configurer.

Pour accéder aux banques mémoire, on utilise le bit  $\text{RP}_0$  et le bit  $\text{RP}_1$  (5<sup>ème</sup> et 6<sup>ème</sup> bit du registre STATUS).

Voir le document ressource **DRES 03**.

DRES 05

TAMBOUR-MOTEUR

18	2	JOINT A LEVRES
17	2	ANNEAU ELASTIQUE POUR ARBRE
16	2	ANNEAU ELASTIQUE POUR ARBRE
15	1	AXE SUPPORT
14	1	AXE SUPPORT
13	2	ROULEMENT A BILLES
12	2	CLAVETTE PARALLELE
11	2	VIS DE PRESSION HC
10	1	CARTER DU REDUCTEUR
9	1	CARTER DU REDUCTEUR
8	3	ECROU H
7	3	TIRANT (GOIJON)
6	1	BAGUE ENTRETOISE
5		BOBINAGE DU STATOR
4	1	STATOR
3	1	ROTOR
2	2	ROULEMENT A BILLES
1	1	AXE ROTOR (PIGNON ARBRE)
REP	NB	DESIGNATION

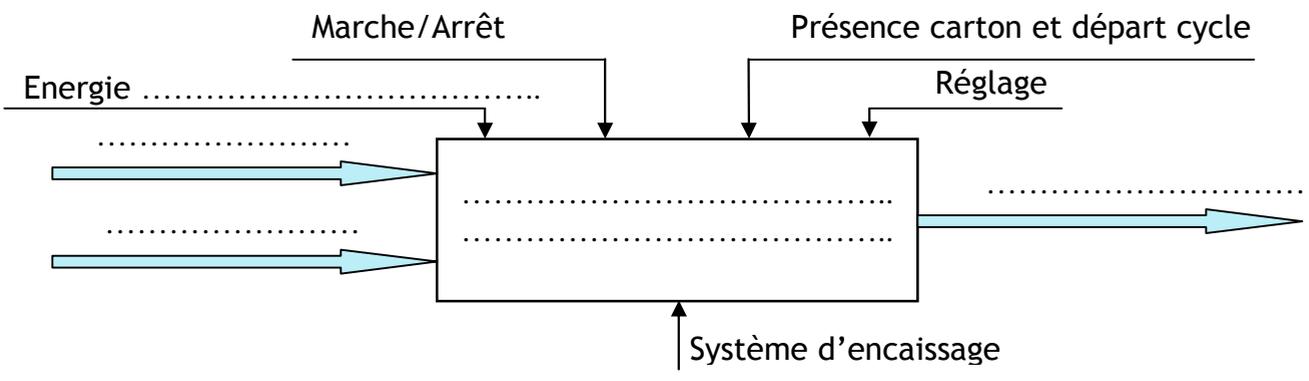
35	1	CLAVETTE
34	6	VIS HC
33	1	ANNEAU ELASTIQUE POUR ARBRE
32	1	ANNEAU ELASTIQUE POUR ALESAGE
31	1	VIS DE PRESSION
30	1	ROULEMENT A BILLES
29	1	BAGUE
28	1	ROUE DENTEE
27	1	PIGNON ARBRE
26	1	BAGUE
25	1	ROULEMENT A AIGUILLES
24	1	ROUE DENTEE
23	8	VIS CHc + RONDELLE
22	1	FLASQUE
21	1	FLASQUE
20	1	TAMBOUR
19	1	ANNEAU ELASTIQUE POUR ALESAGE
REP	NB	DESIGNATION



**DREP 01**

DOCUMENT A RENDRE

**Actigramme A-0**



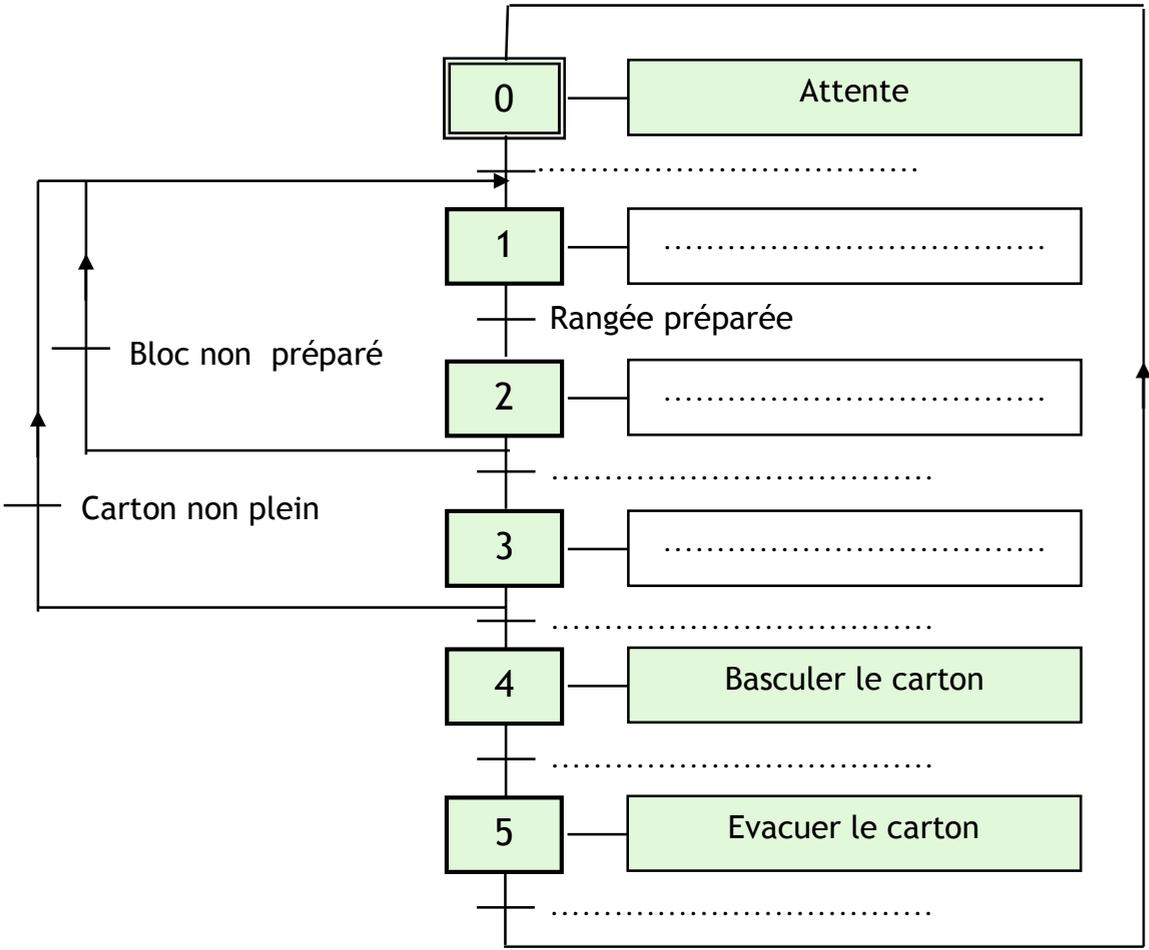
**F.A.S.T du système**

Phase de cycle	PROCESSUS
<b>Préparer une rangée</b>	
Transformer l'énergie électrique en énergie mécanique	.....
Entraîner le tambour moteur	.....
Amener les boîtes	.....
<b>Préparer un bloc</b>	
Soulever une rangée de 3 boîtes	.....
Maintenir les boîtes	.....
<b>Charger les boîtes dans le carton</b>	
Amener un bloc de 3 rangées	.....
<b>Transférer les cartons sur les rouleaux transporteurs</b>	
Basculer les cartons	.....
<b>Evacuer les cartons</b>	
Transformer l'énergie électrique en énergie mécanique	.....
Déplacer les cartons	.....
Saisir le nombre de cartons	<b>Compteur</b>

DREP 02

DOCUMENT A RENDRE

GRAFSET du point de vue système



Dispositif de sécurité

**2.1.** L'amplificateur opérationnel fonctionne en régime non linéaire. Pourquoi ?  
 .....  
 .....

**2.2.** Pour  $V_s < V_c$  donner l'expression numérique de U, l'état du transistor T (saturé ou bloqué) et l'état des contact  $K_1$  et  $K_2$  (fermé ou ouvert).  
 $V_s < V_c \quad \rightarrow U = \dots \quad \rightarrow T : \dots$   
 $\quad \quad \quad \rightarrow K_1 : \dots \quad \quad \rightarrow K_2 : \dots$

**2.3.** Pour  $V_s \geq V_c$  donner l'expression numérique de U, l'état du transistor T (saturé ou bloqué) et l'état des contact  $K_1$  et  $K_2$  (fermé ou ouvert).  
 $V_s \geq V_c \quad \rightarrow U = \dots \quad \rightarrow T : \dots$   
 $\quad \quad \quad \rightarrow K_1 : \dots \quad \quad \rightarrow K_2 : \dots$

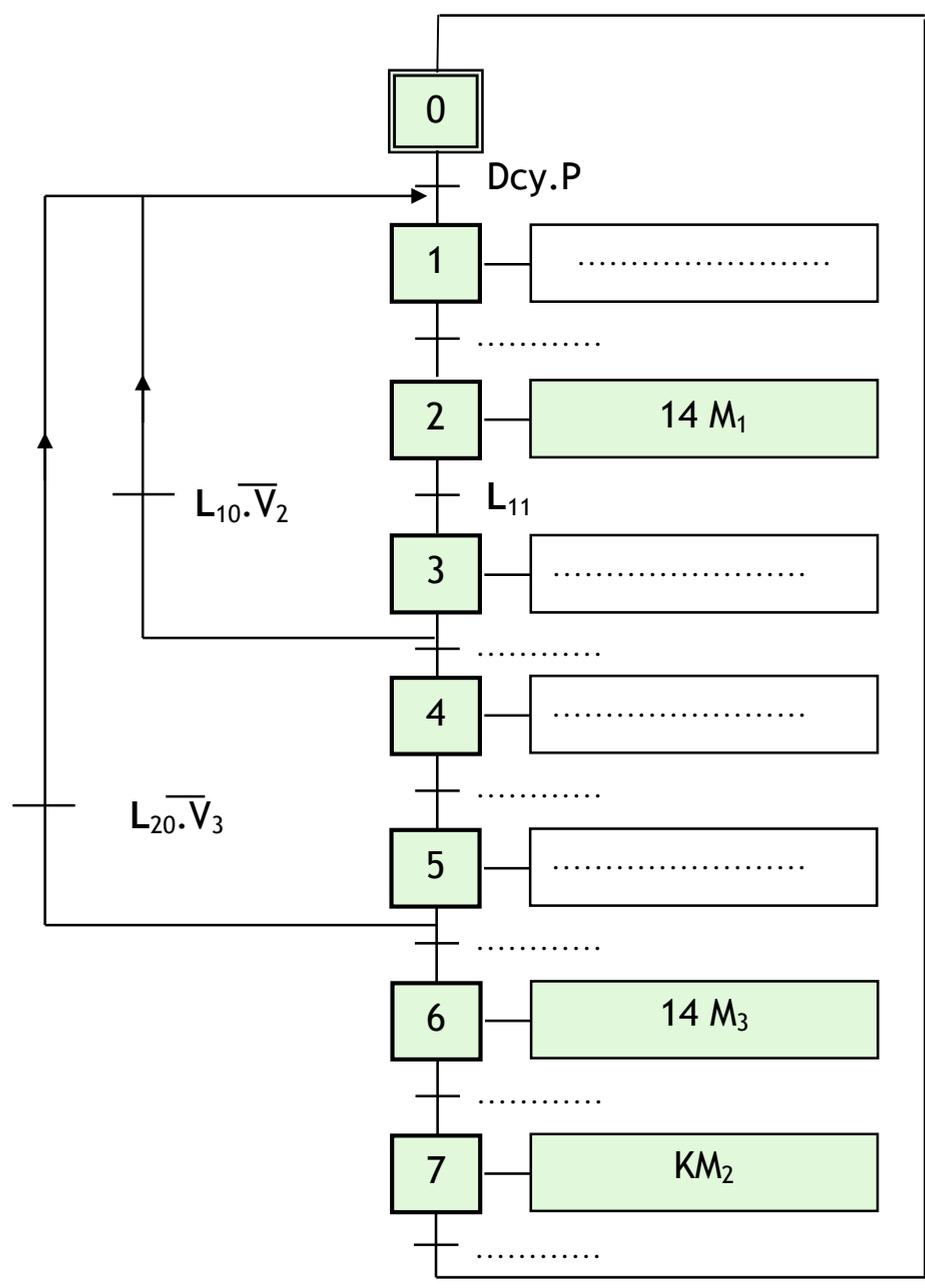
**2.4.** Calculer alors  $V_c$  sachant que la température consigne est égale à 25°c.  
 .....  
 .....

DREP 03

DOCUMENT A RENDRE

**GRAFCET du point de vue partie commande**

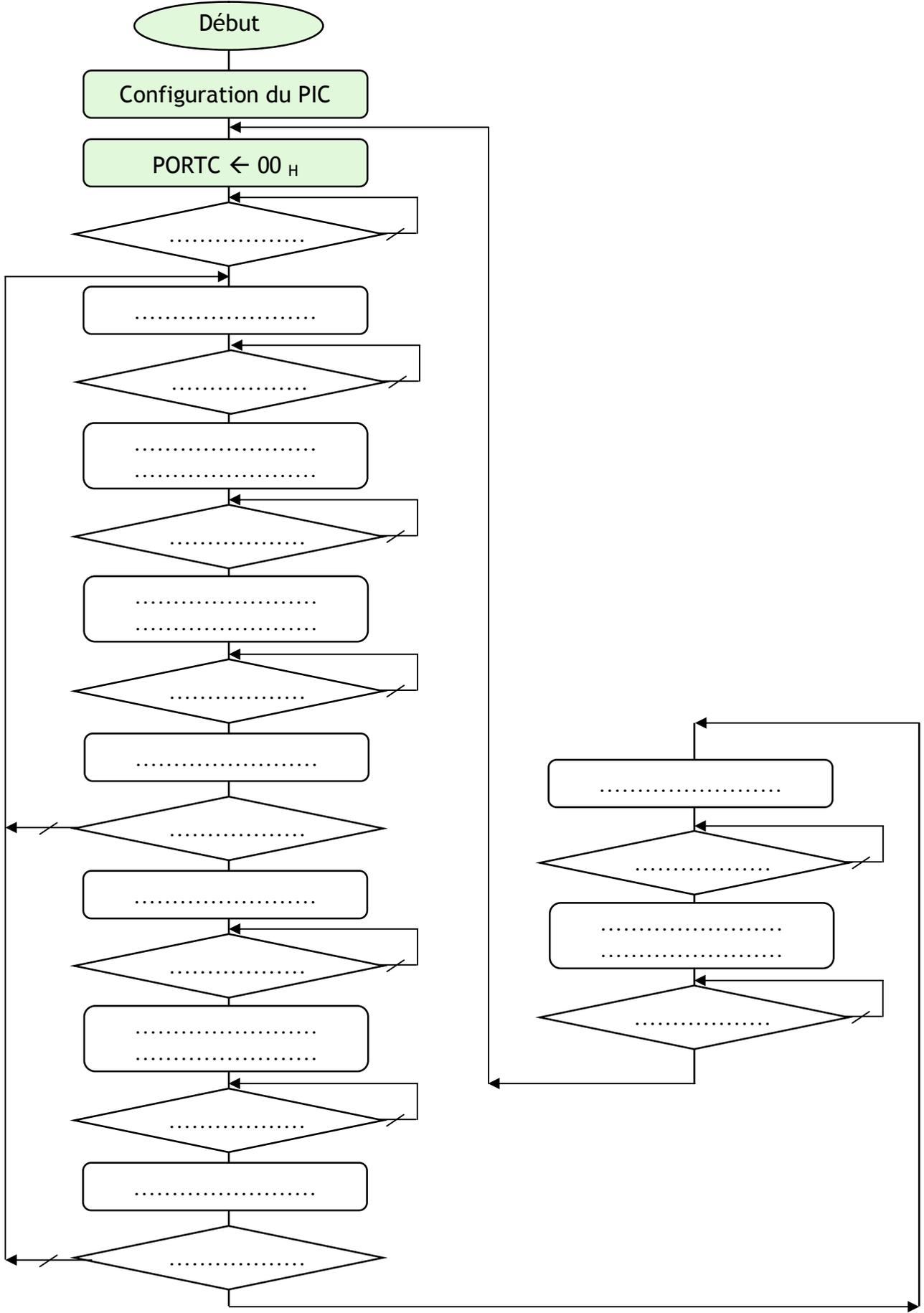
- ❖ Les vérins  $C_1$  et  $C_2$  sont de type double effet à amortissement élastique en fin de course.
- ❖ Le vérin  $C_3$  est de type simple effet amortissement élastique.
- ❖ On donne pour le vérin  $C_i$  (avec  $i = 1, 2$  ou  $3$ )
  - 12  $M_i$  : pilotage du retour de  $C_i$
  - 14  $M_i$  : pilotage de la sortie de  $C_i$
  - Li0 : capteur position tige rentrée de  $C_i$
  - Li1 : capteur position tige sortie de  $C_i$



DREP 04

DOCUMENT A RENDRE

Organigramme



**DREP 05**

**DOCUMENT A RENDRE**

**Initialisation**

```

Init      ..... ;
          ..... ; accès à la BANK 1
          ..... ; PORTC en sortie
          ..... ;
          ..... ; PORTB en entrée
          ..... ; PORTA en entrée
          ..... ; accès à la BANK 0
    
```

**Programme principal**

```

DEBUT      ..... ; état de repos (aucune action)
LAB1       ..... ; Présence barre et Départ cycle
          ..... ;
          ..... ;
          BTFSS   STATUS, Z
          GOTO   LAB1
          ..... ;
LAB2       ..... ; préparer une rangée
LAB3       ..... ; rangée préparée
          ..... ;
          ..... ;
          ..... ; sortir la tige du vérin C1
LAB4       ..... ; tige sortie
          ..... ;
          ..... ;
          ..... ; entrer la tige du vérin C1
LAB5       ..... ; tige entrée
          ..... ;
          ..... ;
          ..... ; bloc préparé
          ..... ;
          ..... ; sortir la tige du vérin C2
LAB6       ..... ; tige sortie
          ..... ;
          ..... ;
          ..... ; entrer la tige du vérin C2
LAB7       ..... ; tige entrée
          ..... ;
          ..... ;
          ..... ; carton chargé
          ..... ;
          ..... ; basculer le carton
LAB8       ..... ; carton basculé
          ..... ;
          ..... ;
          ..... ; évacuer le carton
LAB9       ..... ; carton évacué
          ..... ;
          ..... ;
          END      ..... ; fin du fichier
    
```

DREP 06

DOCUMENT A RENDRE

Acquisition et amplification de température

- 1.1. Donner le nom du montage autour de l'amplificateur opérationnel.  
.....
- 1.2. Donner l'expression de V en fonction de la température en °K.  
.....
- 1.3. Sachant que l'AOP est supposé idéal, on se propose de calculer la tension de sortie VS en fonction des entrées V et Vref.  
.....  
.....  
.....  
.....  
.....
- 1.4. En déduire l'expression de Vs en fonction de T (température en °k).  
.....  
.....  
.....
- 1.5. Sachant que Vref = 2.73 V donner alors l'expression de Vs en fonction de t (en °c).  
.....  
.....  
.....
- 1.6. Calculer R2 pour avoir une sensibilité de 50mV/°c.  
.....  
.....  
.....
- 1.7. Quel est alors le rôle du montage.  
.....
- 1.8. Donner l'expression de Vref en fonction de R3, R4 et Vcc sachant que le courant dans la résistance R1 est très inférieur au courant dans la résistance R4.  
.....  
.....  
.....
- 1.9. Calculer alors R4 pour Vcc = 12 V et R3 = 4.7 kΩ  
.....  
.....  
.....

DREP 07

DOCUMENT A RENDRE

Schéma cinématique

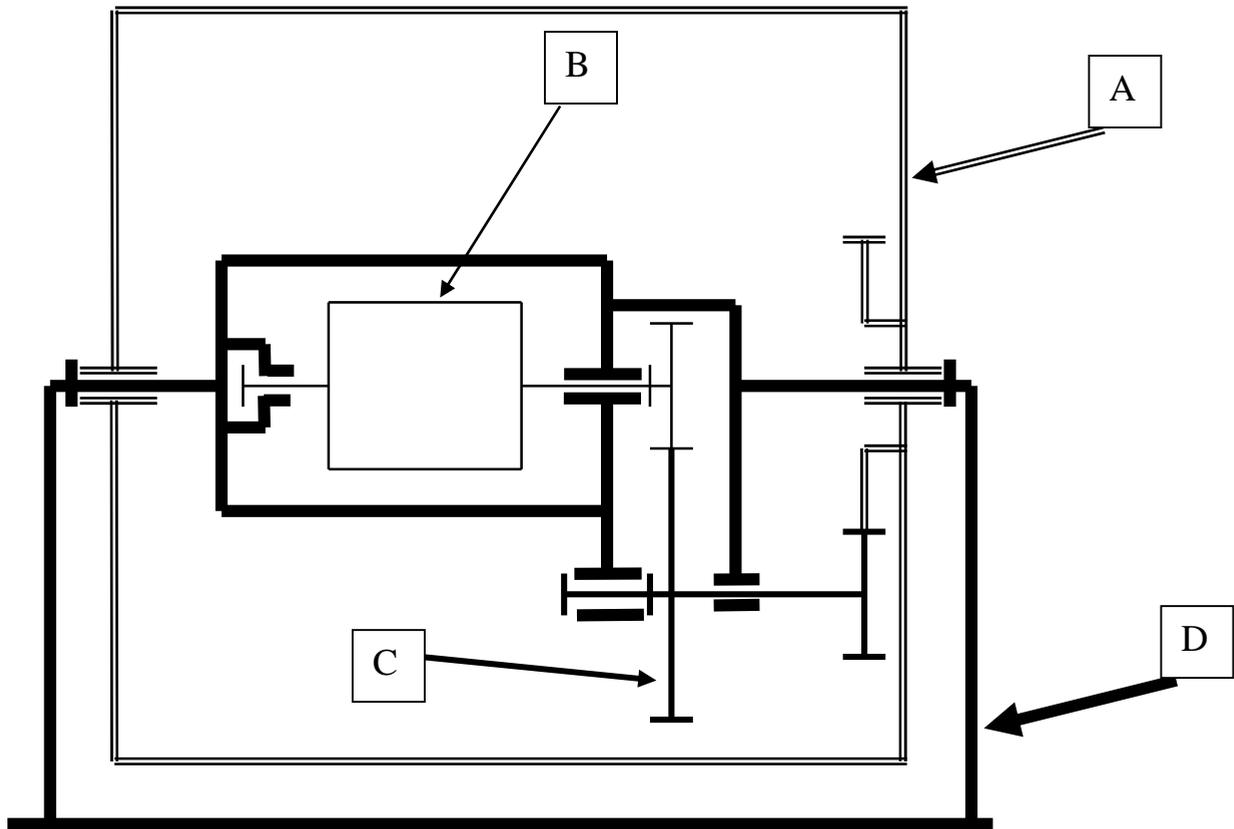


Tableau à compléter

Groupe	Repère des éléments cinématiquement liés
A	B.E13 - 18 - 19 - 20 - 21 - 23-24 - 34 - 22
B	.....
C	.....
D	..... .....

**NB :** BE : bague extérieure    BI : bague intérieure

DREP 08

DOCUMENT A RENDRE

### Calcul de prédétermination

1. Calculer le rapport de réduction ( $r$ ) entre l'arbre moteur et le tambour (20).

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

2. Calculer la vitesse de rotation du tambour (20).

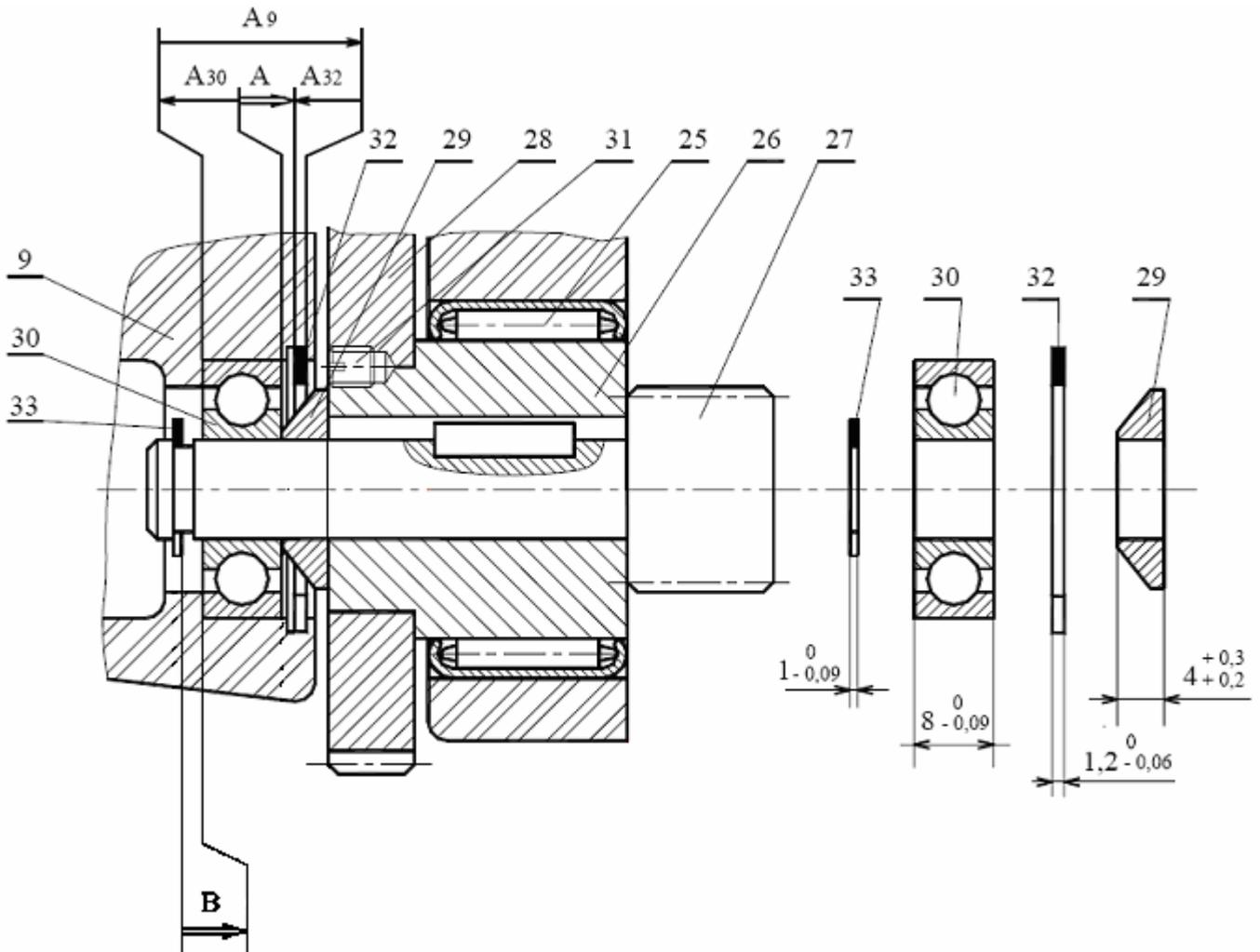
.....

3. Calculer la vitesse de translation des boîtes transférées par le tapis roulant.

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

### Cotation fonctionnelle

1. Tracer la chaîne de cotes relative à la condition B



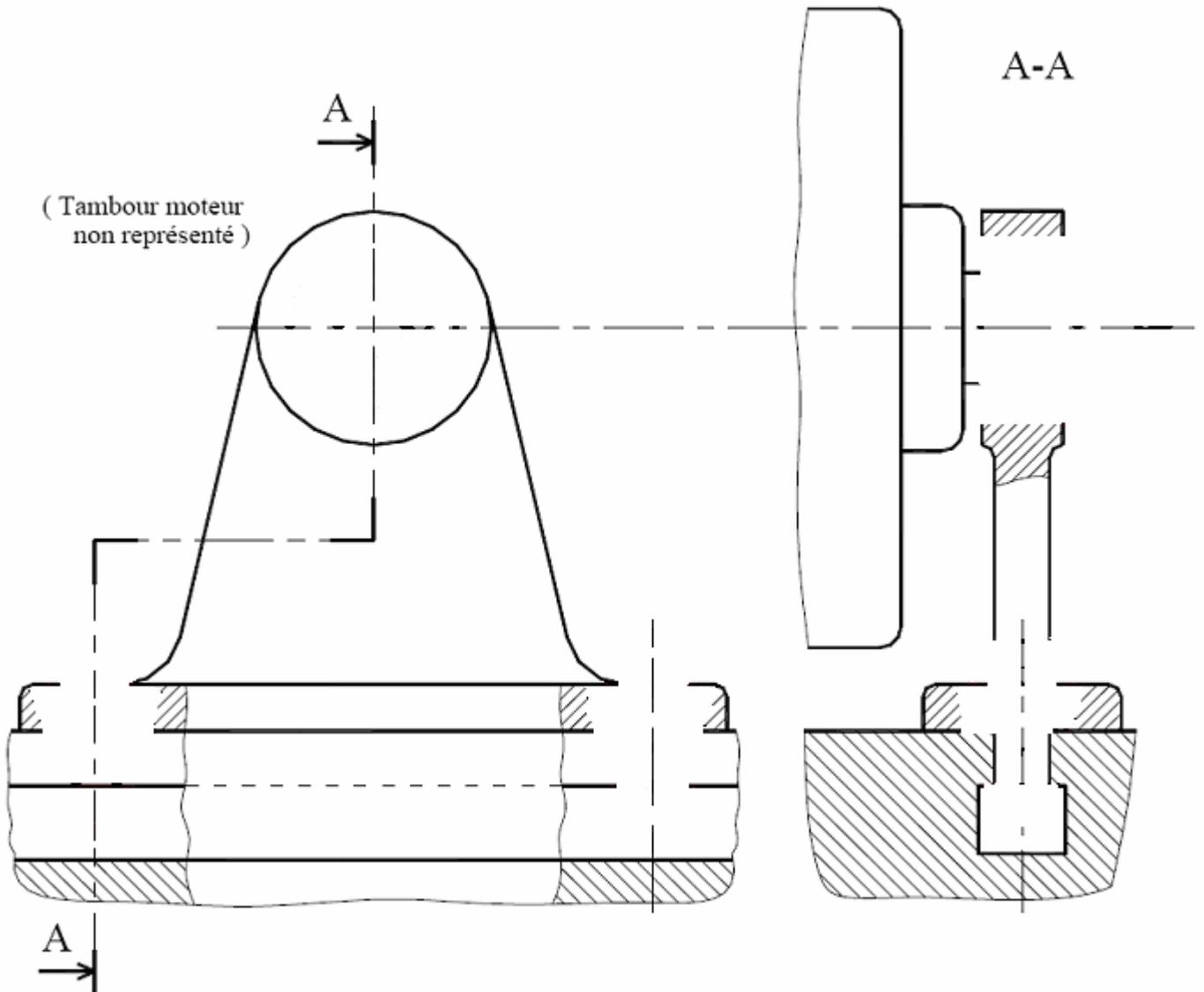
2. Calculer la valeur de la cote A9 sachant que :  $0,2 \leq A \leq 0,5$

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

DREP 09

DOCUMENT A RENDRE

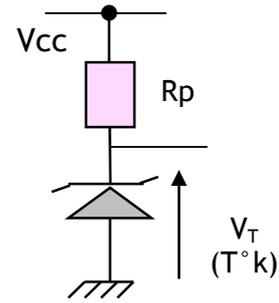
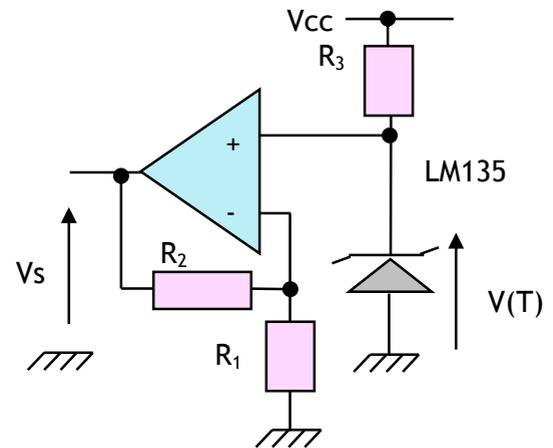
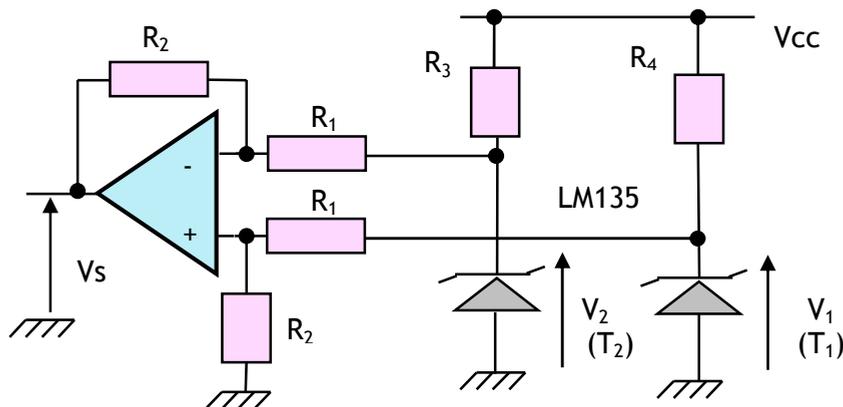
Etude de conception





**Caractéristique du capteur**

Sensibilité  $s = dV/dT = 10\text{mV}/^\circ\text{K}$   
 $V$  : tension aux bornes du LM135 en mV.  
 $T$  : température en  $^\circ\text{K}$   
 $T^\circ\text{K} = t^\circ\text{C} + 273$   
 $V_T$  est alors proportionnelle à la  
température absolue en  $^\circ\text{K}$

**2.2- Exemple d'application :****2.2.1. Mesure de température :****2.2.2. Mesure d'une différence de température :****3- C.T.N :****3.1- Définition:**

C'est un élément passif, non linéaire dont la résistance décroît rapidement en fonction



# Les mémoires électroniques

## 1- Généralité:

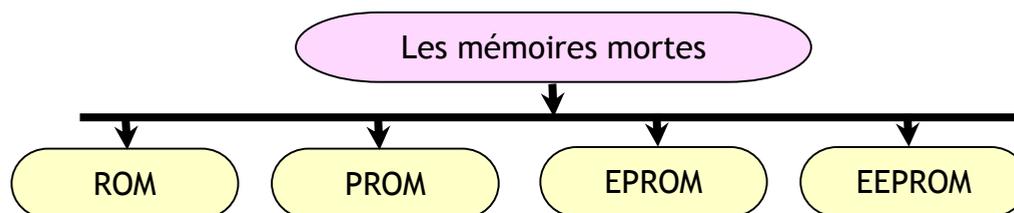
Une cellule mémoire est un élément bistable qui permet de mémoriser un bit d'information. EX : bascule, Disquette...

## 2- Mémoires électroniques:

### 2.1- Mémoire morte (ROM):

C'est une mémoire à lecture seule, son contenu est non modifiable, elle reste inchangée même s'il y a coupure d'alimentation. On dit alors qu'elle est non volatile.

### 2.2- Différent types de ROM:



- ❖ **ROM** : Mémoire programmé par le constructeur.
- ❖ **PROM** : Programmable ROM, c'est une ROM programmable une seule fois par l'utilisateur. La programmation consiste à faire fondre ou non un fusible par application d'une tension calibrée (30v) pendant des temps déterminé.
- ❖ **EPROM** : Erasable PROM, c'est une ROM effaçable et programmable à volonté par l'utilisateur autant de fois qu'il souhaite. Le procédé de programmation est relativement complexe (25v appliqué pendant un temps de 50ms), on utilise en général un programmeur d'EPROM, l'effacement se fait avec des rayons UV à haute énergie.
- ❖ **EEPROM** : Electricly EPROM, c'est un type d'EPROM effaçable électriquement.

### 2.3- Mémoires vives (RAM):

C'est une mémoire à accès aléatoire, on peut à chaque instant changer son contenu. Les RAM perdent leurs informations si on coupe l'alimentation, on dit qu'elles sont volatiles. Il existe deux types de RAM :

- ❖ **RAM statique** : elle est à base de bascules donc de transistors. Elle garde ses informations tant que l'alimentation n'est pas coupée. (Capacité ne dépasse pas 64ko).
- ❖ **RAM dynamique** : c'est une mémoire à base de capacité, elle perd ses informations mêmes si l'alimentation n'est pas coupée. Elle doit être rafraîchi périodiquement (t=1ms).

## 3- Organisation interne:

### 3.1- Capacité:

C'est le nombre total de bits qu'on peut stocker dans une mémoire.

Exemple:

512 octets	
16 ko	$1k = 2^{10}$
2Mo	$1M = 2^{20}$
1Go	$1G = 2^{30}$ .

### 3.2- Longueur de mot:

C'est la façon avec laquelle les bits sont organisés ou rangés, en général par mot de 8 bits ou de 4bits.

- ❖ Un mot de 8bits :1 Octet
- ❖ Un mot de 4bits : 1 Quartet

### 3.3- Adresse:

Pour identifier les mots on donne à chacun une adresse, on dit alors case mémoire d'adresse 40, case mémoire d'adresse FF .....etc.

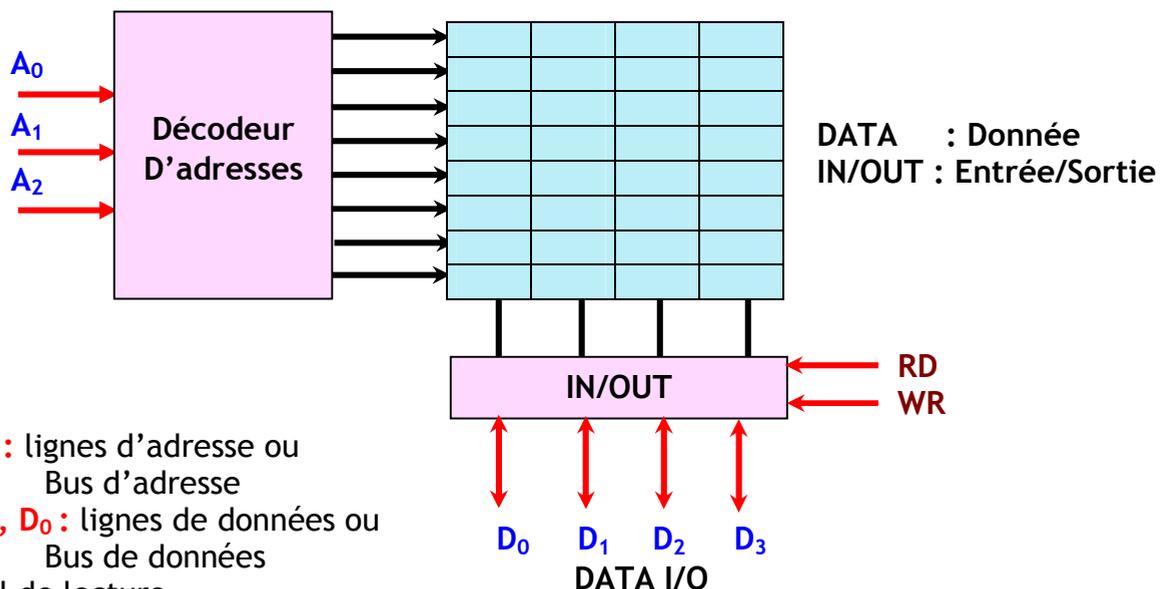
7				
6				
5				
4	1	0	1	0
3				
2				
1				
0	x	x	x	x

8 mots de 4 bits ou 8 quartets  
8 mots de 4 bits = 32 bits

3							
2							
1	x	x	x	x	x	x	x
0							

4 mots de 8 bits ou 4 octets  
4 mots de 8 bits = 32 bits

### 3.4- Décodage d'adresse:



$A_2, A_1, A_0$  : lignes d'adresse ou  
Bus d'adresse

$D_3, D_2, D_1, D_0$  : lignes de données ou  
Bus de données

RD : signal de lecture

WR : signal d'écriture

Capacité =  $2^{\text{nombre de lignes d'adresse}} \times \text{nombre de lignes de données}$

Pour l'exemple ci-dessus :

Capacité =  $2^3 \times 4 \text{ bits} = 8 \times 4 \text{ bits} = 32 \text{ bits}$

Capacité =  $8 \times 4 \text{ bits} = 8 \text{ q} (1 \text{ q} = 4 \text{ bits})$

# Les interfaces

## 1- Introduction:

### 1.1- Définition:

Une interface est un circuit d'adaptation entre le  $\mu p$  et le périphérique. Il établit la compatibilité entre les entrées sorties du  $\mu p$  et celles des périphériques, à plusieurs niveaux :

- ❖ Au niveau des types de transmission série ou parallèle.
- ❖ Au niveau de la vitesse de transmission ; car les périphériques sont très lents par rapport au  $\mu p$ .

### 1.2- Un périphérique :

On appelle périphérique tout organe ou dispositif externe au système minimum qui échange des informations avec le  $\mu p$ . il existe deux types de périphérique.

- ❖ **Organe d'entrée** : Il envoie les informations vers le  $\mu p$  (le clavier).
- ❖ **Organe de sorties** : Il reçoit les informations du  $\mu p$  (l'écran).

## 2- Périphériques d'entrées sorties parallèles:

### 2.1- Définition:

Une liaison // consiste à envoyer l'information de mot de n bits, véhiculés de façon simultanée sur n fils.

### 2.2- Structure et fonctionnement :

Une interface parallèle programmable comporte :

- ❖ Des registres tampon de sorties : Les transferts d'entrées sorties se ramènent à une lecture ou écriture de ces registres appelés ports.
- ❖ Registre de commandes : qui permet de configurer le port en entrée ou en sortie selon l'utilisation.

### 2.3- Programmation de l'interface :

Une interface s'adaptera à un périphérique donné par une programmation de l'interface. Cette programmation consiste à configurer les ports de l'interface en entrée ou en sortie selon l'utilisation.

## 3- Périphériques d'entrées sorties séries:

### 3.1- Définition:

Une liaison série consiste à transmettre les données de façon successive (bit par bit) sur un seul et unique fil.

Les avantages de la liaison série sont énormes :

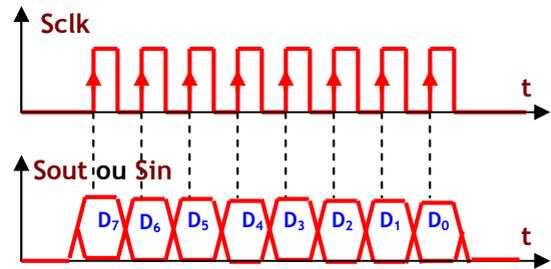
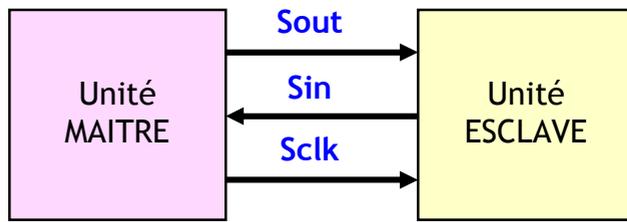
- ❖ utilisation d'un nombre réduit de fil.
- ❖ Fiabilité de l'information due au contrôle de mot du l'émetteur jusqu'au récepteur.



Il y a 2 types de liaison série : **synchrone** et **asynchrone**.

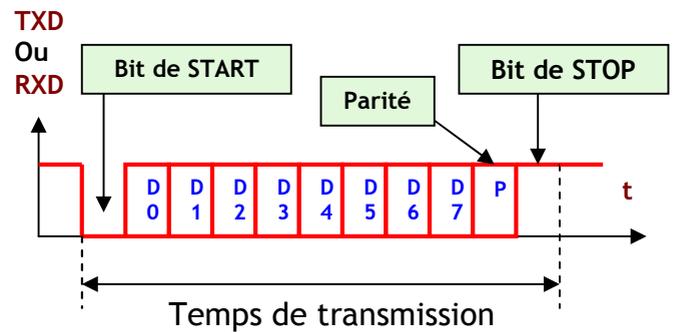
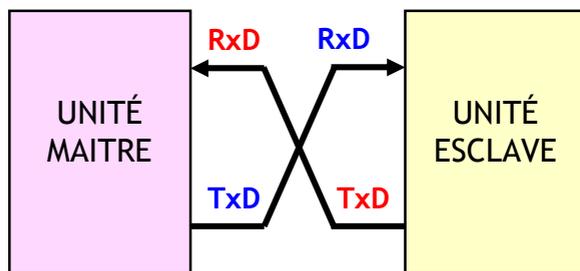
### 3.2- Liaison série synchrone :

Dans ce dispositif la transmission est synchronisée par un signal d'horloge émis par l'unité maître.



### 3.3- Liaison série asynchrone :

Ce dispositif ne possède pas de signal d'horloge de synchronisation. Les unités en liaison possèdent chacune une horloge interne cadencée à la même fréquence. Lorsqu'une unité veut émettre un mot binaire, elle génère un front descendant sur sa ligne émettrice. A la fin de l'émission de ce mot, la ligne repasse au niveau haut. La donnée à transmettre peut contenir un bit supplémentaire appelé "parité" et servant à la correction d'erreurs.



### 3.4- Exemples :

❖ Liaison // :

Port **Centronics** (Pour imprimante parallèle)

❖ Liaison série :

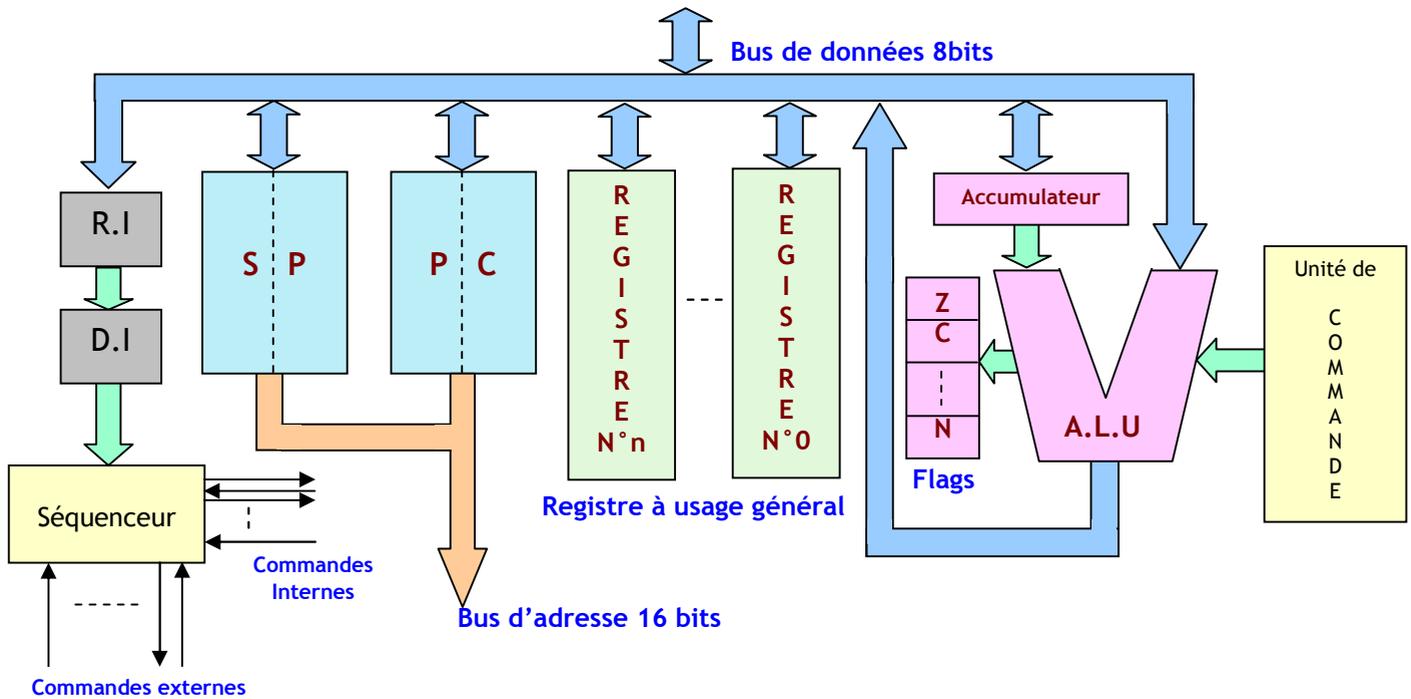
Port **USB** (Universal Serial Bus)

Liaison **RS232** (liaison série point à point)

Liaison **RS485** (liaison série multipoints)

# Le microprocesseur

## 1- Structure interne :



## 2- Unité arithmétique et logique :

L'ALU accomplit les opérations arithmétique et logiques. Elle est dotée à l'une de ces entrées d'un registre spécial appelé "Accumulateur".

## 3- Registre d'état :

C'est le registre des indicateurs d'état (Flags) ; leur rôle est de mémoriser les situations exceptionnelles qui peuvent survenir pendant le fonctionnement du  $\mu P$ .

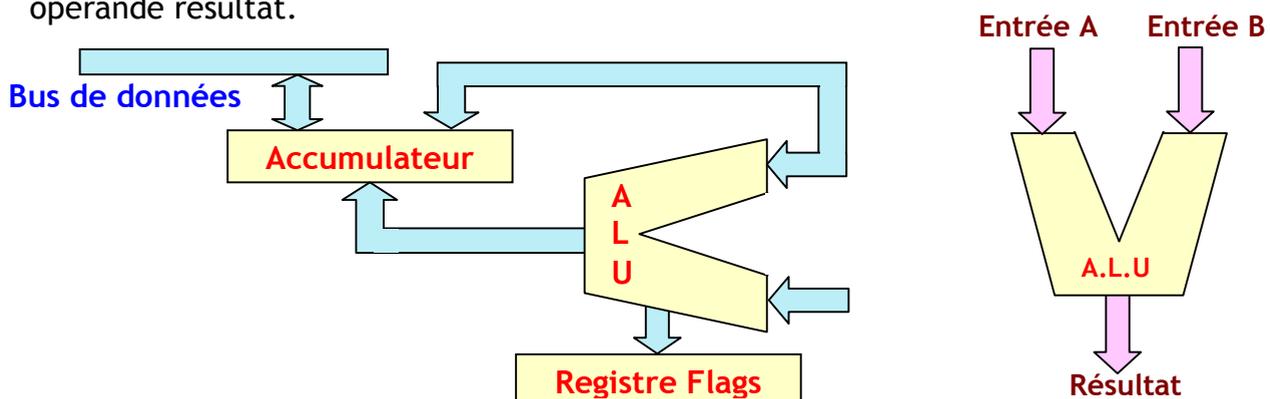
C : Carry (si  $c = 1$  il y a retenue, si  $c = 0$  pas de retenue)

Z : Zéro (si  $z = 1$  résultat = 0, si  $z = 0$  résultat  $\neq 0$ )

Le contenu du registre d'états est testé en général par des instructions spécialisées tel que les instructions conditionnelles.

## 4- Accumulateur:

C'est le registre principal d'un microprocesseur 8bits, il présente la particularité de pouvoir être désigné, dans la même instruction, à la fois comme opérande et comme opérande résultat.



### 5- Registre à usage universel (général):

Leur rôle est de permettre à l'A.L.U de manipuler les données à grande vitesse. Ils sont généralement numérotés de 0 jusqu'à n, leur rôle n'est pas défini à l'avance. C'est pour quoi on les appelle registres à usage universel. Ils peuvent contenir n'importe quelle valeur utilisée par le programme.

### 6- Le compteur de programme :

Le compteur de programme ou Program Counter est présent dans tous les processeurs, il contient toujours l'adresse de la prochaine instruction à exécuter. Une fois le code de l'instruction est récupéré, le PC est incrémenté automatiquement pour pointer l'instruction suivante.

### 7- Le pointeur de pile :

Le pointeur de pile ou Stak Pointer contient toujours une adresse particulière ou la valeur du PC est stockée temporairement. Le SP est indispensable au fonctionnement des sous programmes et des interruptions.

### 8- Registre d'instructions :

Le registre d'instructions est un registre 8bits, dont le rôle est de stocker le code opération de l'instruction en cours d'exécution, ce registre n'est pas programmable.

### 9- Décodeur d'instructions :

Son rôle est d'identifier le code opération de l'instruction présente sur le RI, et générer la séquence correcte des signaux interne et externe qui permettra l'exécution de l'instruction.

### 10- Les cycles de fonctionnement d'un microprocesseur :

Quel que soit le processeur, le cycle de fonctionnement se fait en 3 cycles :

- ❖ Récupération de l'instruction.
- ❖ Décodage de l'instruction.
- ❖ Exécution de l'instruction.

## Le PIC 16 F 84

### 1- Caractéristiques:

- ❖ 1K de mémoire programme
- ❖ 68 octets de RAM
- ❖ 64 octets D'EEPROM
- ❖ 13 entrée/sortie réparties en 2 ports
- ❖ PORTA et PORTB
- ❖ 4 sources d'interruption
- ❖ 1 Timer/Compteur
- ❖ 1 Chien de garde
- ❖ MODE SLEEP (pour une faible consommation)
- ❖ 4 Sources d'oscillateur sélectionnable
- ❖ Protection du code
- ❖ 35 instructions seulement

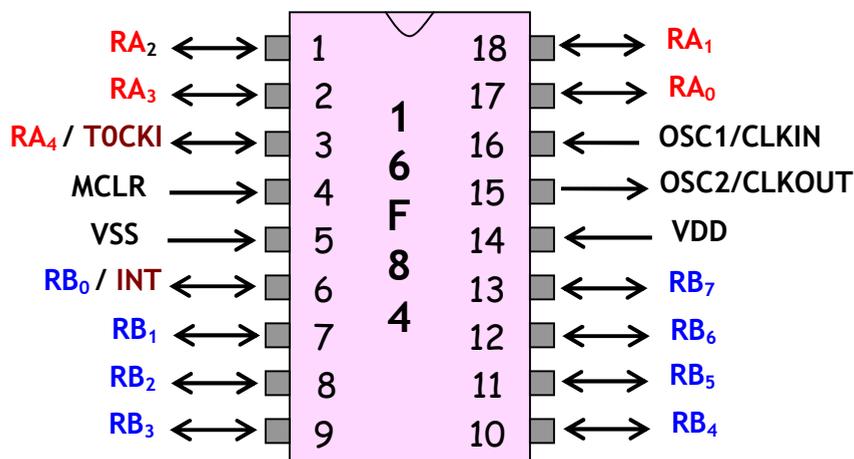
### 2- Structure externe du PIC 16F84 :

#### 2.1- Description:

Le PIC16F84 est un microcontrôleur 8 bits de faible coût, mais il permet toutefois de créer de nombreuses applications. De plus, avec son jeu d'instructions réduit, il est très agréable à utiliser lorsqu'on désire s'initier à l'étude d'un microcontrôleur.

#### 2.2- Brochage:

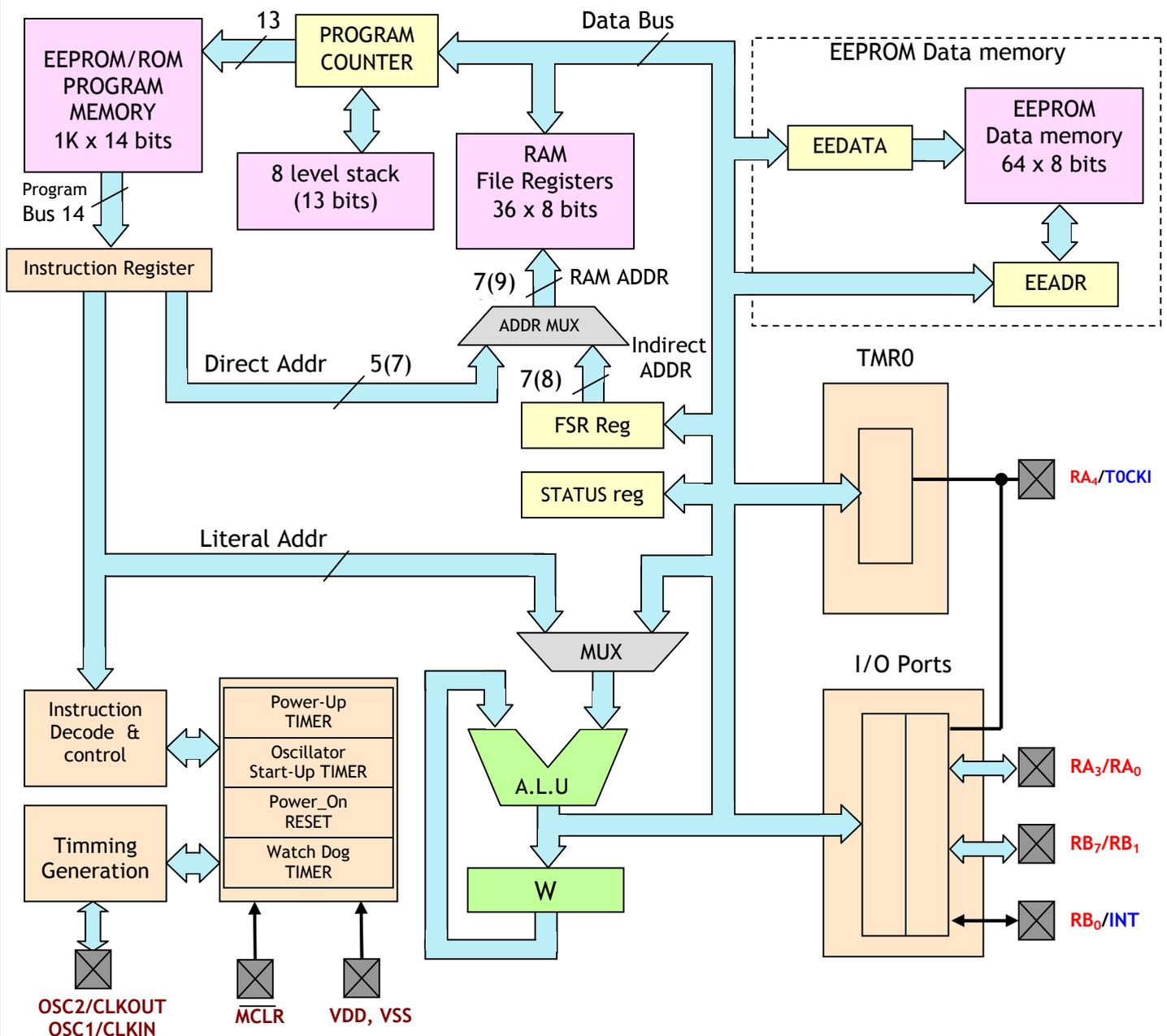
Le PIC16F84 est logé dans un boîtier 18 broches



- VDD, VSS : Broches d'alimentation (3 à 5.5 v).  
 OSC1, OSC2 : Signaux d'horloge. Peuvent recevoir un circuit RC ou un résonateur.  
 CLKINT : Peut être connectée à une horloge externe de 0 à 4, 10 ou 20 Mhz.  
 MCLR : Reset ou Master Clear.  
 TOCKI : entrée d'horloge externe du Timer 0.  
 INT : entrée d'interruption externe.  
 RA0..RA4 : 5 E/S du PORTA.  
 RB0..RB7 : 8 E/S du PORTB.

## 3- Structure interne du PIC 16F84 :

## 3.1- Synoptique:



## 3.2- Les Mémoires:

## 3.2.1. EEPROM de Programme :

Cette mémoire de 1K stock le programme. L'emplacement de celui-ci peut être à n'importe quel endroit de la mémoire. Suite à un RESET ou lorsqu'on l'alimente, le PIC16F84 commence toujours à l'adresse 0000H (Vecteur RESET). De plus, lorsqu'il y a une interruption, et si celle-ci est validée, le microcontrôleur va à l'adresse 0004H (Vecteur d'interruption).

## 3.2.2. RAM de données:

La mémoire de donnée est divisée en deux parties. Une partie appelée SFR, l'autre le GPR, qui sont encore divisées en deux pages (Bank0 et Bank1). Toutes les données de la mémoire sont appelées registres y compris les données utilisateurs.

❖ **LE SFR (Special Function Registers):**

Le SFR est l'ensemble des registres qui permet de configurer tous les modules internes du PIC16F84 (les PORTS, le TIMER, la gestion des interruptions, etc....).

❖ **LE GPR (General Purpose Register):**

Le GPR est l'ensemble des registres à usage général, utilisé par le programmeur pour stocker les variables et les données.

**3.2.3. EEPROM de données:**

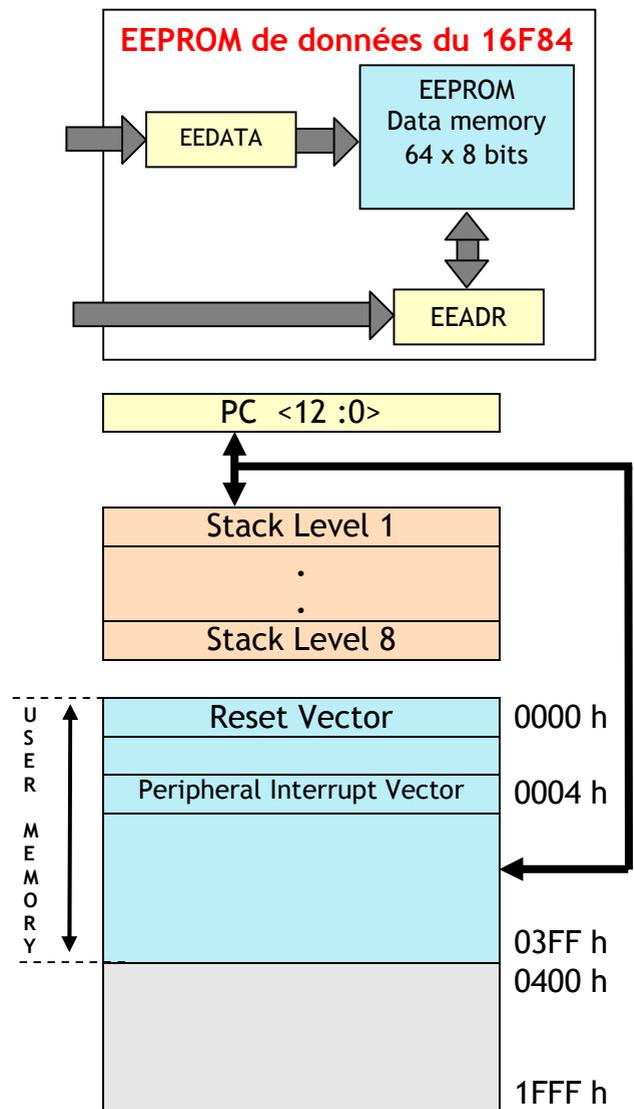
Une EEPROM de 64 octets est disponible pour y stocker les données semi permanentes. Pour accéder à cette mémoire, on utilise les registres EEDATA, EEADR. 2 registres de contrôle sont associés à cette mémoire EECON1 et EECON2.

**3.2.4. La pile:**

Un groupe de 8 registres de 13 bits, son rôle est de sauver temporairement le contenu du PC lors de l'appel d'un sous programme ou du service d'une interruption. La pile n'est pas dans le fichier des registres, donc elle n'est pas manipulable par le programmeur.

00 h	INDF	INDF	80 h
01 h	TMRO	OPTION	81 h
02 h	PCL	PCL	82 h
03 h	STATUS	STATUS	83 h
04 h	FSR	FSR	84 h
05 h	PORTA	TRISA	85 h
06 h	PORTB	TRISB	86 h
07 h			87 h
08 h	EEDATA	EECON1	88 h
09 h	EEADR	EECON2	89 h
0A h	PCLATH	PCLATH	8A h
0B h	INTCON	INTCON	8B h
0C h			8C h
	36	Mapped	
	General	(accesses)	
	Purpose	In Bank 0	
	Registers		
	(SRAM)		
2F h			AF h
30 h			B0 h
7F h			FF h
<b>ADDR</b>	<b>BANK 0</b>	<b>BANK 1</b>	<b>ADDR</b>
<b>FILE</b>			<b>FILE</b>

RAM du 16F84



EEPROM programme du 16F84

### 3.3- Les Registres :

Ces registres spéciaux font partie du SFR (Special Function Registers) et configurent le  $\mu$ C. Certains registres initialisent les périphériques alors que d'autres sont utilisés par le CPU.

#### ❖ Les registres utilisés par le CPU :

- ✓ Le registre **INDF** (adresse 00, 80H)
- ✓ Le registre **FSR** (adresse 04, 84H)
- ✓ Le registre **STATUS** (adresse 03 ET 83H)
- ✓ Le registre **INTCON** (adresse 0B et 8BH)
- ✓ Le **PC** (Program Counter) ; Il est composé de deux registres:
  - Le registre **PCL** (adresse 02, 82H)
  - Le registre **PCLATH** (adresse 0A, 8AH)

#### ❖ Les registres utilisés par le PORTA :

- ✓ Le registre **TRISA** (adresse 85H)
- ✓ Le registre **PORTA** (adresse 05H)

#### ❖ Les registres utilisés par le PORTB :

- ✓ Le registre **TRISB** (adresse 86H)
- ✓ Le registre **PORTB** (adresse 06H)

#### ❖ Les registres utilisés par le TIMER :

- ✓ Le registre **TMRO** (adresse 01H)
- ✓ Le registre **OPTION** (adresse 81H)

#### ❖ Les registres utilisés par l'EEPROM :

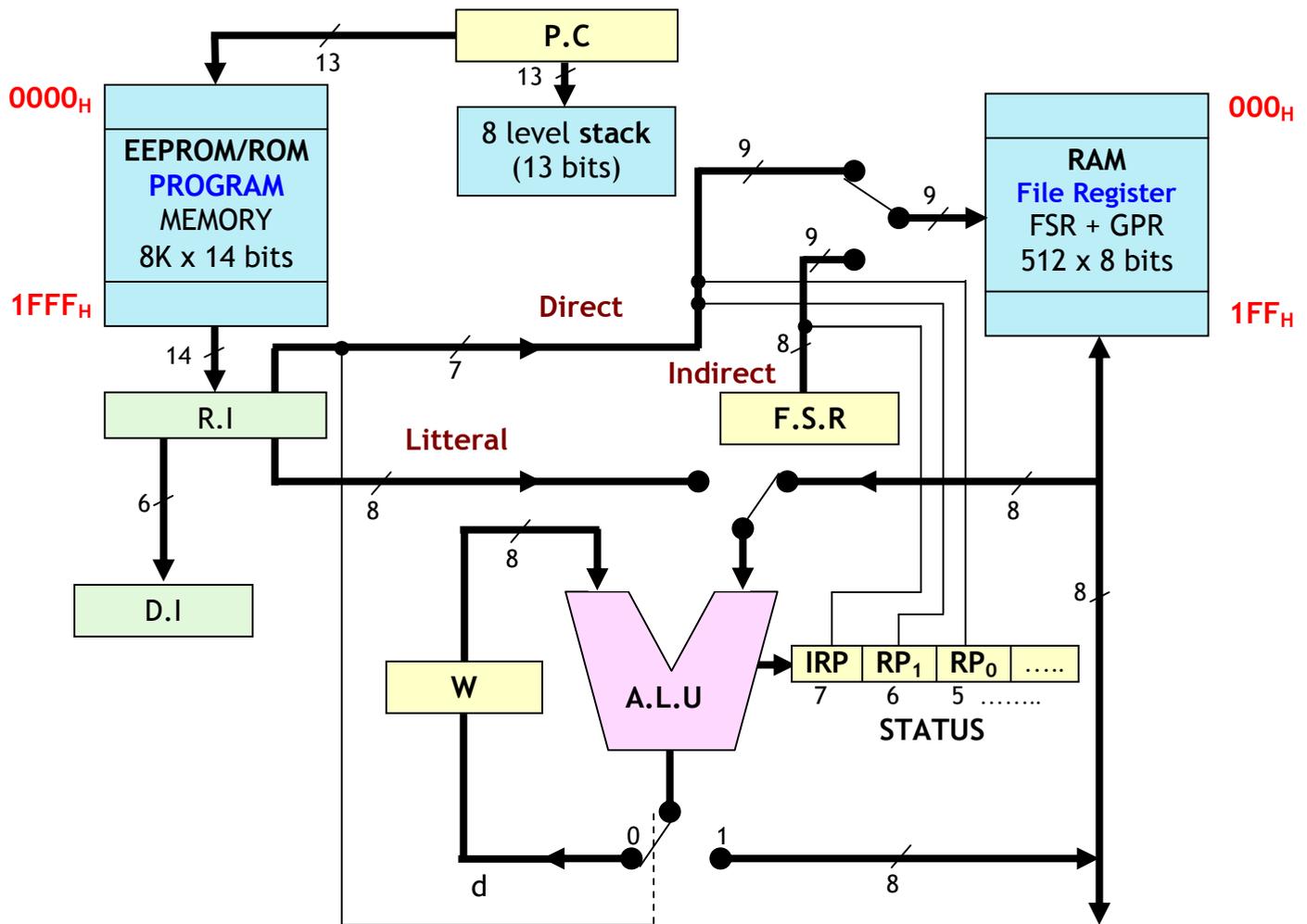
- ✓ Le registre **EEDATA** (adresse 08H):
- ✓ Le registre **EEADR** (adresse 09H)
- ✓ Le registre **EECON1** (adresse 88H)
- ✓ Le registre **EECON2** (adresse 89H)

### 4- Remarques:

- ❖ Chaque E/S est compatible TTL.
- ❖ La configuration de chaque BIT du port est déterminée avec les registres TRISA et TRISB.
- ❖ La broche RA4 est multiplexée avec l'entrée d'horloge du registre TMRO.
- ❖ La broche RB0 est multiplexée avec l'interruption INT.
- ❖ Les broches RB4 à RB7, à condition qu'elles soient configurées en ENTREE, peuvent générer une interruption lorsqu'elles changent d'états.
- ❖ Toutes les broches du PORTB bénéficient d'un "tirage au plus" interne.

## Jeu d'instruction

### 1- Structure des PIC 16F...:



### 2- Le registre STATUS :



C'est le registre d'état, il contient :

- ✓ 5 bits, témoins (drapeaux) caractérisant le résultat de l'opération réalisée par la CPU (lecture seule).
  - ❖ /TO : (Time Out) débordement du timer WDT.
  - ❖ /PD : (Power Down) caractérise l'activité du chien de garde WDT.
  - ❖ Z : (zéro) résultat nul pour une opération arithmétique et logique.
  - ❖ DC : (digit carry) retenue sur un quartet (4 bits).
  - ❖ C : (carry) retenue sur un octet (8 bits).
- ✓ 2 bits - RP<sub>1</sub> et RP<sub>0</sub> - de sélection de banc (Ecriture/lecture).
- ✓ 1 bit - IRP - de sélection de page (Ecriture/lecture).

Au reset, seul le bit RP<sub>0</sub> de sélection de banc est fixé à 0.

### 3- Le jeu d'instruction:

Un microcontrôleur est caractérisé non seulement par sa structure matérielle mais aussi par son jeu d'instructions qui permet de traiter les données.

### 3.1- Les modes d'adressages:

❖ Littéral :

Dans ce mode d'adressage, la donnée est intégrée avec le code de l'opération.

**MOVLW 0x55** ; charger la valeur 0x55 dans le registre W

❖ Direct :

Les 7 premiers bits de l'adresse du fichier, sont intégrés avec le code de l'opération.  
Les 2 bits RP0, RP1 du registre STATUS sont utilisés pour compléter l'adresse sur 9 bits.

**MOVF 0x55, W** ; charger le contenu de l'adresse 0x55 dans W

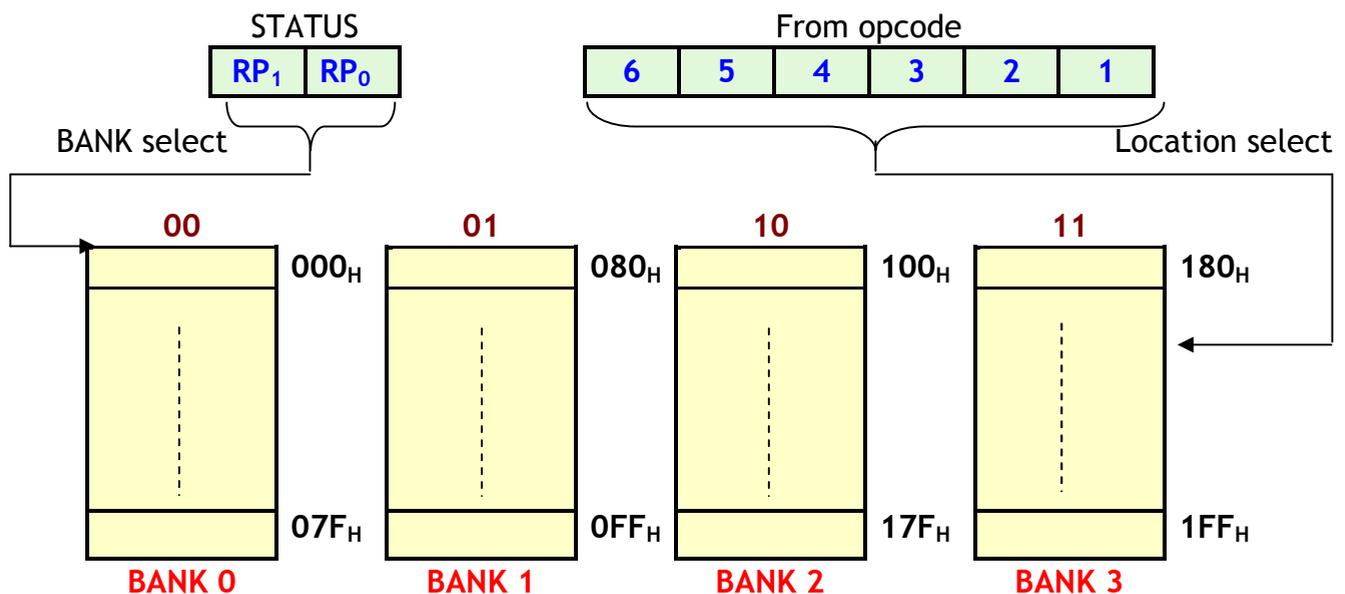
❖ Indirect :

L'adresse du fichier est formée par le contenu du FSR et le bit IRP du registre STATUS.

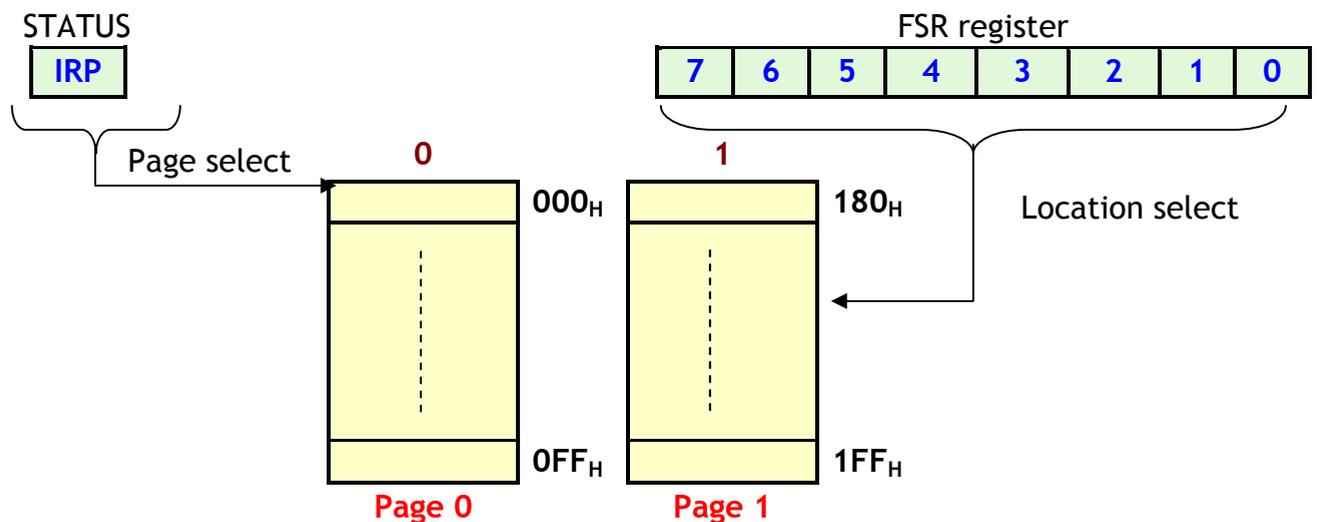
**MOVF INDF, W** ; charger le contenu de la case mémoire pointée par le FSR.

### 3.2- Plan mémoire :

#### 3.2.1. Plan mémoire en mode direct :



#### 3.2.2. Plan mémoire en mode indirect :



### 3.3- Le jeu d'instruction :

Le PIC16F84 a un jeu d'instructions relativement limité mais possède une architecture interne (RISC) qui permet une programmation efficace et rapide (toutes les instructions, exceptées les sauts, s'exécute en un cycle d'horloge).

Instructions opérant sur les registres		STATUS	Cycles
ADDWF F,d	$W+F \rightarrow \{W,F ? d\}$	C, DC, Z	1
ANDWF F,d	$W \text{ and } F \rightarrow \{W,F ? d\}$	Z	1
CLRF F	Clear F	Z	1
COMF F,d	Complémente F $\rightarrow \{W,F ? d\}$	Z	1
DECF F,d	Décrémente F $\rightarrow \{W,F ? d\}$	Z	1
DECFSZ F,d	Décrémente F $\rightarrow \{W,F ? d\}$ skip if 0		1(2)
INCF F,d	Incrémente F $\rightarrow \{W,F ? d\}$	Z	1
INCFSZ F,d	Incrémente F $\rightarrow \{W,F ? d\}$ skip if 0		1(2)
IORWF F,d	$W \text{ or } F \rightarrow \{W,F ? d\}$	Z	1
MOVF F,d	$F \rightarrow \{W,F ? d\}$	Z	1
MOVWF F	$W \rightarrow F$		1
RLF F,d	Rotation à gauche de F à travers C $\rightarrow \{W,F ? d\}$	C	1
RRF F,d	Rotation à droite de F à travers C $\rightarrow \{W,F ? d\}$	C	1
SUBWF F,d	$F - W \rightarrow \{W,F ? d\}$	C, DC, Z	1
SWAPF F,d	Permute les 2 quartets de F $\rightarrow \{W,F ? d\}$		1
XORWF F,d	$W \text{ xor } F \rightarrow \{W,F ? d\}$	Z	1
Instructions opérant sur les bits		STATUS	Cycles
BCF F,b	Mise à 0 du bit b e F	C, DC, Z	1
BSF F,b	Mise à 1 du bit b de F	Z	1
BTFSC F,b	Teste le bit b de F, si 0 saute une instruction	Z	1(2)
BTFSS F,b	Teste le bit b de F, si 1 saute une instruction	Z	1(2)
Instructions opérant sur les constantes		STATUS	Cycles
ADDLW K	$W + K \rightarrow W$	C, DC, Z	1
ANDLW K	$W \text{ and } K \rightarrow W$	Z	1
IORLW K	$W \text{ or } K \rightarrow W$	Z	1
MOVLW K	$L \rightarrow W$	Z	1
SUBLW K	$K - W \rightarrow W$	C, DC, Z	1
XORLW K	$W \text{ xor } K \rightarrow W$	Z	1
Autres instructions		STATUS	Cycles
CLRW	Clear W	Z	1
CLRWDT	Clear Watchdog timer	TO', PD'	1
CALL L	Branchement à un sous programme de label L		2
GOTO L	Branchement à la ligne de label L		2
NOP	Pas d'opération		1
RETURN	Retour d'un sous programme		2
RETFIE	Retour d'interruption		2
RETLW K	Retour d'un sous programme avec K dans W		2
SLEEP	Se met en mode standby	TO', PD'	1

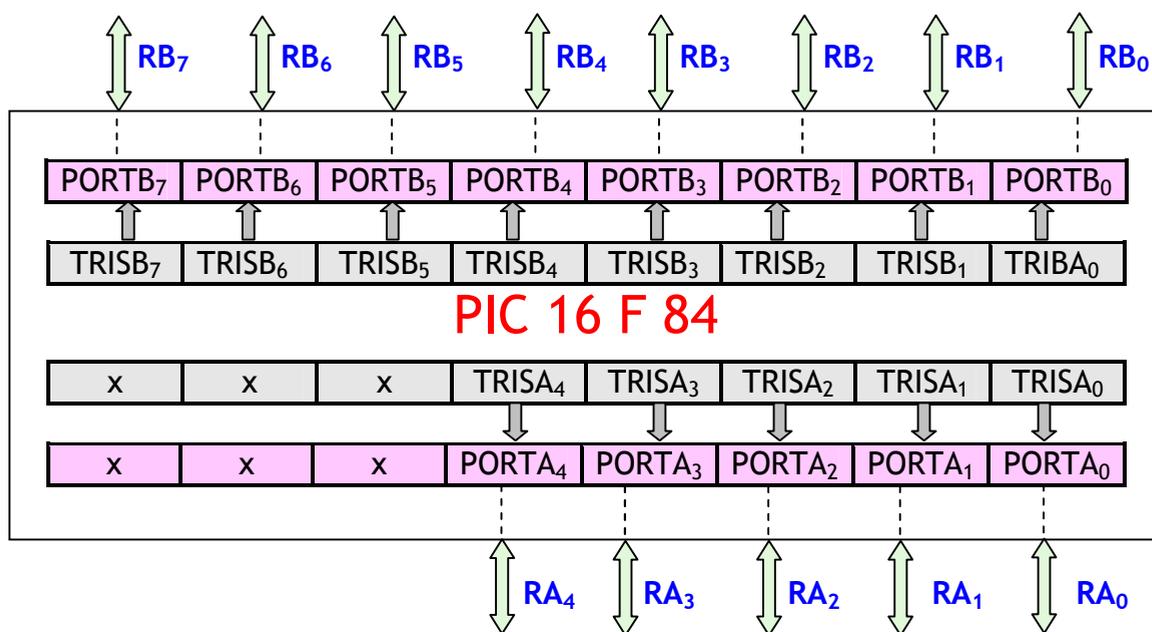
## Les Ports A et B

### 1- Structure:

Pour communiquer avec l'extérieur le PIC16f84 dispose de 2 ports : **PORTA** et **PORTB**. Les ports sont **bidirectionnels**, ce qui signifie qu'ils peuvent être configurés et utilisés comme des entrées ou des sorties. Le microcontrôleur reçoit les informations sur un port d'entrée :

- ❖ informations logiques issues de capteurs sur un ou plusieurs bits d'un port d'entrée,
- ❖ informations numériques codées sur 8 bits sur un port entier.
- ❖ informations analogiques variables dans le temps, si le PIC est doté d'un convertisseur analogique / numérique.

Le microcontrôleur traite ces données et les utilise pour commander des circuits qui sont connectés sur un port de sortie.



### 2- Le registre TRISA (Adresse 85H):

Le registre TRISA configure chaque E/S du PORTA en entrée ou en sortie. Après un RESET, toutes les E/S sont en entrée. Si le bit associé à la porte est à "1", alors elle sera configurée en entrée. Si le bit est à "0", elle sera en sortie. La broche RA4 est multiplexée avec l'entrée d'horloge du registre TMRO.

### 3- Le registre PORTA (Adresse 05H) :

Le PORTA est un port de 5 bits (RA0 à RA4). Chaque E/S est compatible TTL. La configuration de chaque BIT du port est déterminée avec le registre TRISA.

### 4- Le registre TRISB (Adresse 86H) :

Le registre TRISB configure chaque E/S du PORTB en entrée ou en sortie. Après un RESET, toutes les E/S sont en entrée. Si le bit associé à la porte est à "1", alors elle sera configurée en entrée. Si le bit est à "0", elle sera en sortie. La broche RB0 est multiplexée avec l'interruption INT.

Les broches RB4 à RB7, à condition qu'elles soient configurées en entrée, peuvent générer une interruption lorsqu'elles changent d'états. Enfin, toutes les broches du PORTB bénéficient d'un "tirage au plus" interne.

**5- Le registre TRISB (Adresse 86H) :**

Le PORTB est un port bidirectionnel de 8 bits. Toutes les broches sont compatibles TTL. La configuration du PORTB est réalisée avec le registre TRISB.

**6- Configuration d'un PORTx :**

Les registres TRISx appartiennent à la **Bank 1** des SFR. Lors de l'initialisation du µC, il ne faut pas oublier de changer de page mémoire pour les configurer. Pour configurer un PORTx, il faut :

- ❖ Accéder à la Bank 1.
- ❖ Déterminer le mot à maître dans le registre TRISx.
- ❖ Mettre ce mot dans le registre de travail W.
- ❖ Transférer le contenu de W dans le registre TRISx.
- ❖ Accéder à la Bank 0, pour pouvoir accéder au PORTx.

**7- Exemple Application :**

**7.1- Exemple 1 :**

Configurer le PORTB en entrée,  
Lire le contenu du PORTB.  
Mettre le résultat dans la case mémoire d'adresse 0C<sub>H</sub>.

..... ; accès à la Bank1  
 ..... ;  
 ..... ; PORTB en entrée  
 ..... ; accès à la Bank0  
 ..... ; W ← PORTB  
 ..... ; (0C<sub>H</sub>) ← W

Bit n°	7	6	5	4	3	2	1	0
PORTB	...	...	...	...	...	...	...	...
TRISB	...	...	...	...	...	...	...	...
Hex	....				....			

**7.2- Exemple 2 :**

Configurer le PORTB en sortie.  
Initialiser le PORTB à FF<sub>H</sub>.

..... ; accès à la Bank1  
 ..... ;  
 ..... ; PORTB en sortie  
 ..... ; accès à la Bank0  
 ..... ; W ← FF<sub>H</sub>  
 ..... ; PORTB ← W

Bit n°	7	6	5	4	3	2	1	0
PORTB	...	...	...	...	...	...	...	...
TRISB	...	...	...	...	...	...	...	...
Hex	....				....			

**7.3- Exemple 3 :**

Configurer les broches RB<sub>1</sub>, RB<sub>3</sub>, RB<sub>5</sub>, RB<sub>7</sub> du PORTB en entrée et les broches RB<sub>0</sub>, RB<sub>2</sub>, RB<sub>4</sub>, RB<sub>6</sub> du PORTB en sortie.

..... ; accès à la Bank1  
 ..... ;  
 ..... ; PORTB en sortie  
 ..... ; accès à la Bank0

Bit n°	7	6	5	4	3	2	1	0
PORTB	...	...	...	...	...	...	...	...
TRISB	...	...	...	...	...	...	...	...
Hex	....				....			

