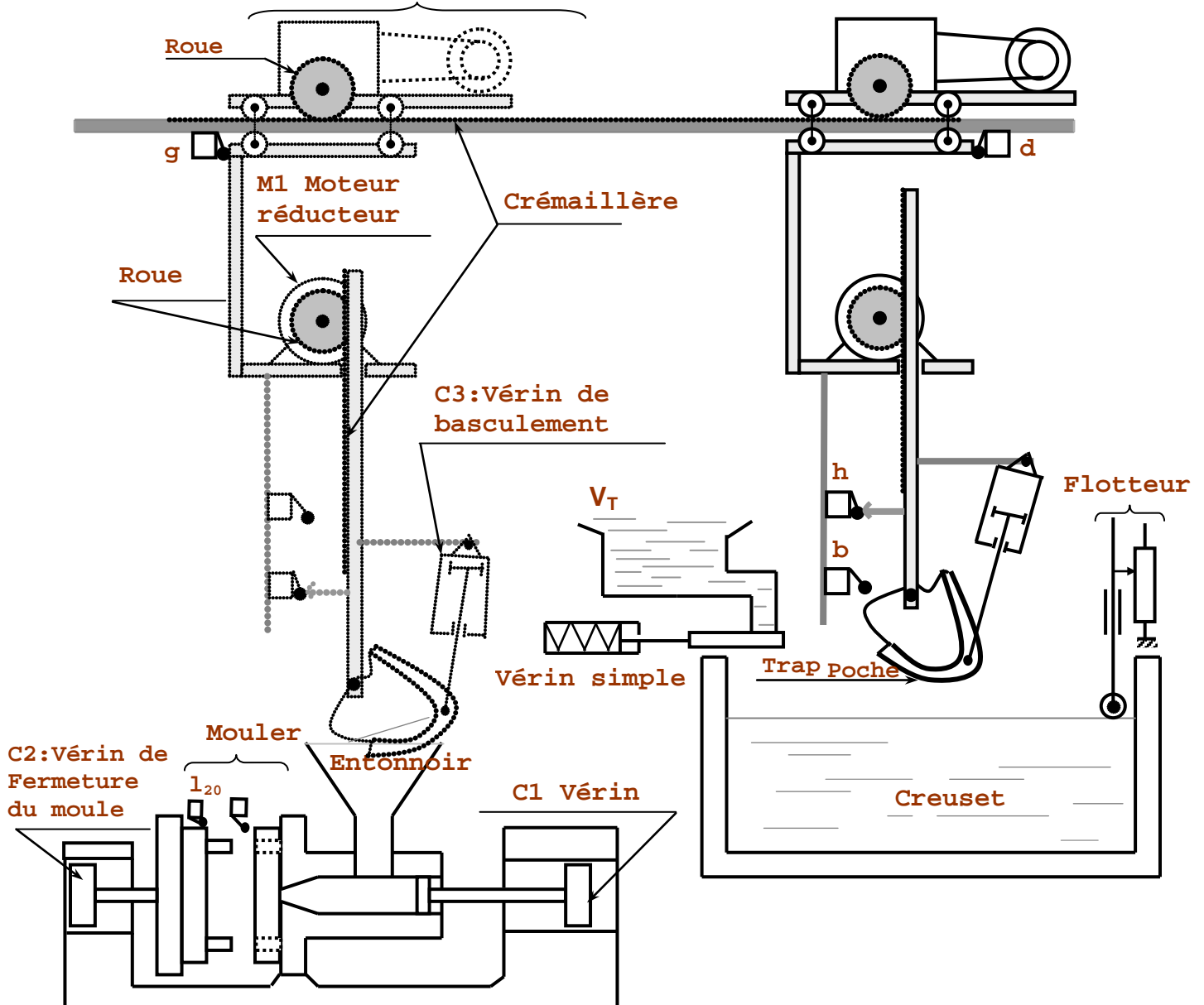


## SYSTEME AUTOMATIQUE DE MOULAGE

### I- PRESENTATION:

Le système représenté ci-dessous permet de mouler des pièces sous pression dans un moule alimenté par une poche à partir d'un bac contenant du métal (Aluminium) en fusion arrivant d'un four d'alimentation.

**M2 Moteur réducteur-embayage- frein**  
(Déplacement latéral(horizontal))



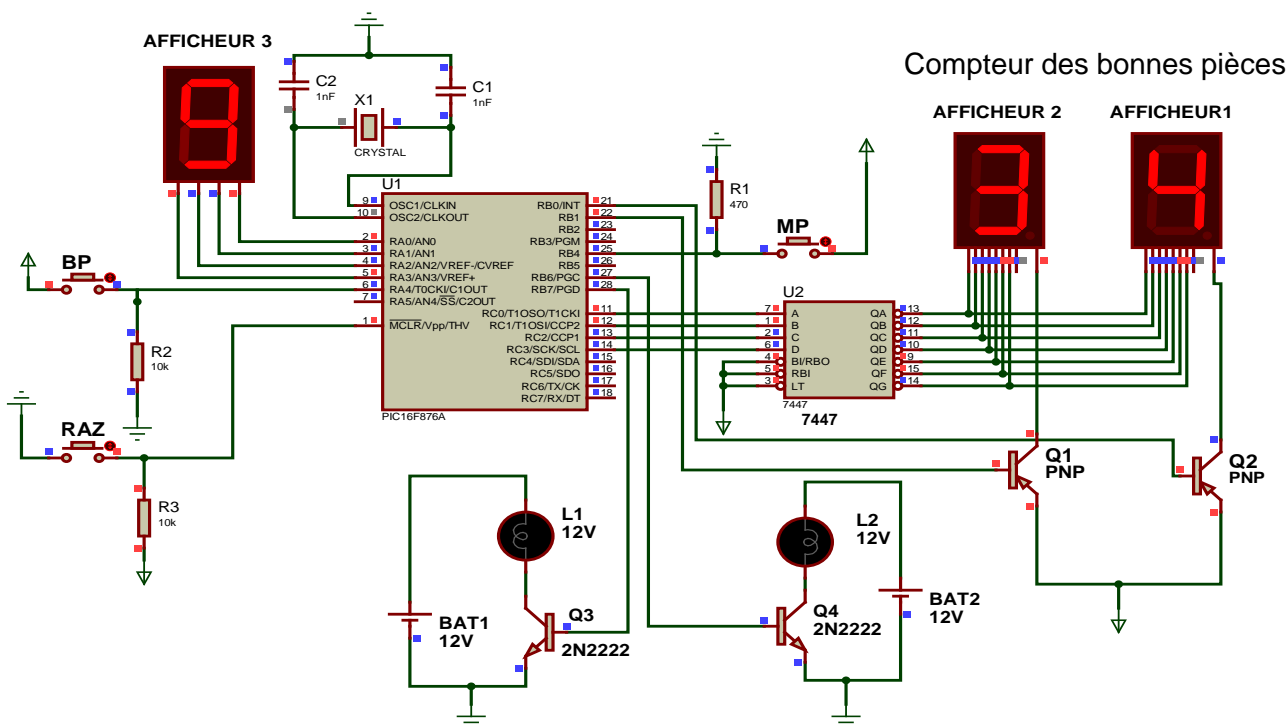
### II- CONSTITUTION :

Le système ci-dessus est constitué principalement :

- D'un moteur M1 à deux sens de marche entraînant la poche verticalement.
- D'un moteur M2 et un système pignon crémaillère entraînant la poche horizontalement.
- D'un vérin d'injection (C1).
- D'un vérin de fermeture du moule (C2).
- D'un vérin de basculement de la poche (C3)

### III- COMPTAGES DES PIÈCES PRODUITES:

Compteur des mauvaises pièces



Les pièces moulées sont triées en deux catégories

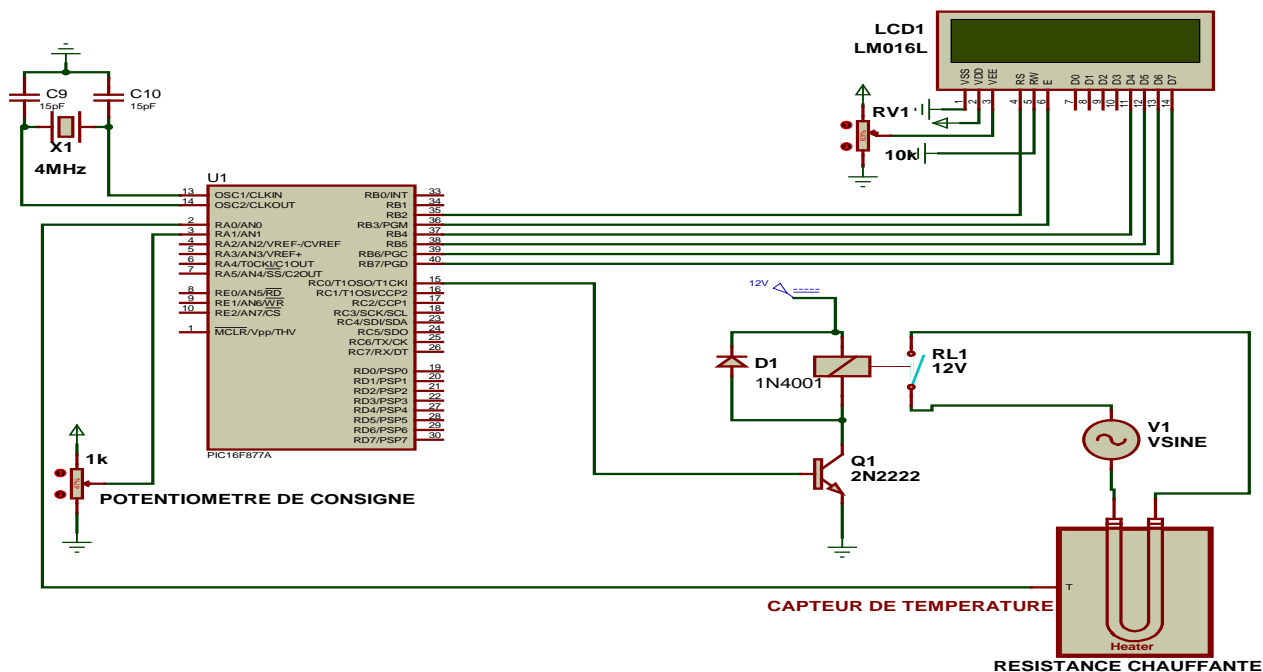
- Pièces bonnes qui sont détectées par un capteur BP.
- Pièces mauvaises qui sont détectées par un capteur MP.

Les bonnes pièces sont emballées par lot de 100 et qui sont comptées le compteur TMR0 qui s'incrémente à chaque front montant sur la broche **RA4** (Elle est reliée au capteur **BP**) du microcontrôleur 16F876A ; la formation d'un lot de 100 bonnes pièces moulées est signalée par la lampe **L2** qui s'allume pendant une **seconde** pour informer l'employeur.

Les mauvaises pièces sont comptées par un compteur qui s'incrémente à chaque changement d'état de la broche **RB4** (Interruption externe via le **port b**) qui est reliée au capteur **MP**.

La formation d'un lot de **10 pièces** est signalée par une lampe **L1** qui s'allume pendant 5s pour avertir l'employeur qui va intervenir en appuyant sur le bouton **RAZ** pour arrêter le système puis le réparer.

### IV- CONTROLE ET COMMANDE DE LA TEMPERATURE DU FOUR:



La résistance chauffante du four d'alimentation est commandée par un PIC 16F877A comme indique la figure ce dessus.

La résistance chauffante envoie à travers le capteur de température une tension proportionnelle à la température du four d'une manière que 5000mV correspond à 250°C

L'opérateur fixe la valeur de la température à laquelle on désire chauffer le four en agissant sur un potentiomètre de consigne d'une manière que 5000mV correspond à 250°C aussi

Le microcontrôleur en recevant les deux tensions à travers ces entrées analogiques AN0 et AN1 -prévenant du capteur de température (image de la température réelle du four) sur AN0.

- prévenant du potentiomètre de consigne (image de la température que l'on désire chauffer le four) sur AN1.

Il convertit ces deux tensions en deux nombres puis en deux valeurs correspondantes de température ( $T_f$  : pour la température réelle du four et  $T_c$  : pour la température imposée par l'opérateur à travers le potentiomètre)

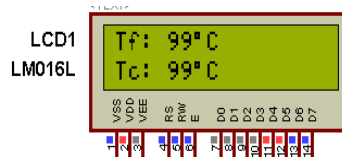
Le microcontrôleur réalise l'opération suivante **erreur =  $T_c - T_f$** .

Si l'erreur est supérieure à zéro on donne l'ordre de chauffer le four à travers la broche RC0 qui commande le four à travers un relais électromagnétique RL1, sinon on donne l'ordre d'arrêter le chauffage.

Cette opération se réalise d'une manière continue tant que le système fonctionne.

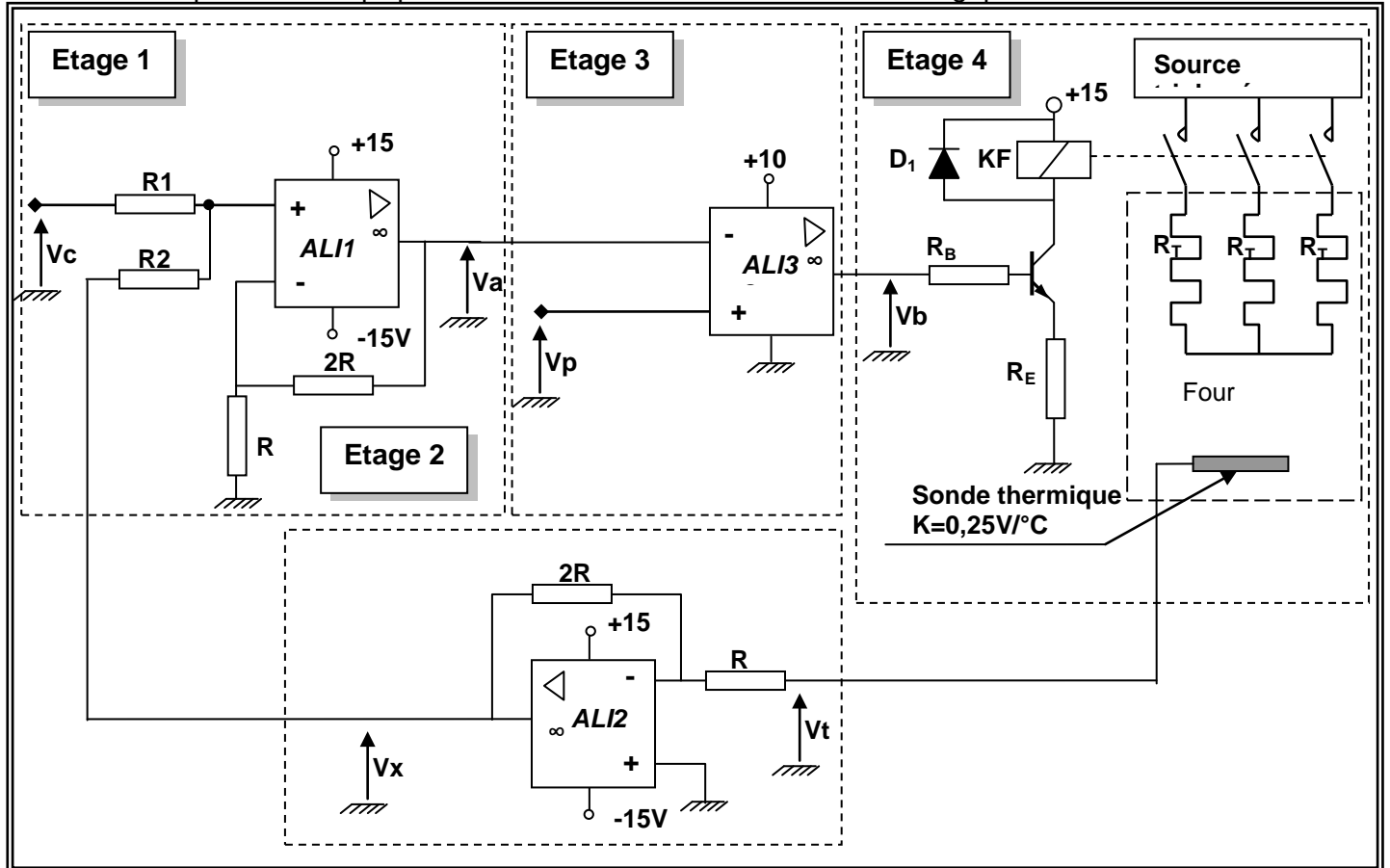
Sur un afficheur LCD, les valeurs des températures doivent être affichées de la manière suivante.

**NB : Un afficheur LCD LM 016L comporte 2 lignes**



## V- PRODUCTION D'UNE SOLUTION DE MODIFICATION:

Au lieu d'utiliser une commande numérique pour commander et contrôler la température du four ; on utilise la carte électronique suivante qui permet une commande et un contrôle analogique



## VI- ANNEXE:

### 1- Configuration registre ADCON1 :

Bit 7	<b>ADFM :</b> Bits de sélection du format de résultat de conversion Analogique / Numérique 1 = Justifié à droite. Les 6 bits de poids fort du registre ADRESH sont lus comme '0' 0 = Justifié à gauche. Les 6 bits de poids faible du registre ADRESL sont lus comme '0'
Bit 6 : 4	<b>Non implémentés :</b> Lire comme '0'
Bit 3 : 0	<b>PCFG3:PCFG0 :</b> Configuration des E/S et des tensions de références.

Ces bits permettent de choisir le partage entre entrées analogiques et digitales sur les PORTS A et E. Ils permettent également de choisir pour  $V_{REF+}$  entre  $V_{DD}$  et RA3 et pour  $V_{REF-}$  entre  $V_{SS}$  et RA2.

PCFG3..0	RE2	RE1	RE0	RA5	RA3	RA2	RA1	RA0	$V_{REF+}$	$V_{REF-}$
0000	A	A	A	A	A	A	A	A	$V_{DD}$	$V_{SS}$
0001	A	A	A	A	$V_{REF+}$	A	A	A	RA3	$V_{SS}$
0010	D	D	D	A	A	A	A	A	$V_{DD}$	$V_{SS}$
0011	D	D	D	A	$V_{REF+}$	A	A	A	RA3	$V_{SS}$
0100	D	D	D	D	A	D	A	A	$V_{DD}$	$V_{SS}$
0101	D	D	D	D	$V_{REF+}$	D	A	A	RA3	$V_{SS}$
011x	D	D	D	D	D	D	D	D	$V_{DD}$	$V_{SS}$
1000	A	A	A	A	$V_{REF+}$	$V_{REF-}$	A	A	RA3	RA2
1001	D	D	A	A	A	A	A	A	$V_{DD}$	$V_{SS}$
1010	D	D	A	A	$V_{REF+}$	A	A	A	RA3	$V_{SS}$
1011	D	D	A	A	$V_{REF+}$	$V_{REF-}$	A	A	RA3	RA2
1100	D	D	D	A	$V_{REF+}$	$V_{REF-}$	A	A	RA3	RA2
1101	D	D	D	D	$V_{REF+}$	$V_{REF-}$	A	A	RA3	RA2
1110	D	D	D	D	D	D	D	A	$V_{DD}$	$V_{SS}$
1111	D	D	D	D	$V_{REF+}$	$V_{REF-}$	D	A	RA3	RA2

Ces entrées n'existent pas sur le 16F876.

**A** : broche configurée en entrée analogique.

**D** : broche configurée en entrée ou en sortie numérique.

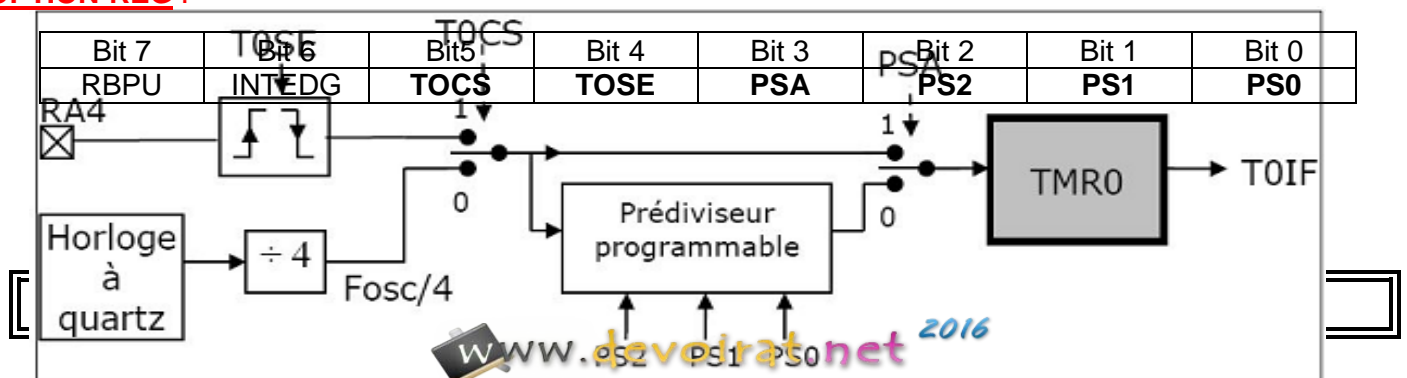
## 2- Configuration registre INTCON :

INTCON	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
	<b>GIE</b>	<b>EEIE</b>	<b>TOIE</b>	<b>INTE</b>	<b>RBIE</b>	<b>TOIF</b>	<b>INTF</b>	<b>RBIF</b>

Bit 7	<b>GIE</b>	Global Interrupt Enable : <b>autorise toute les sources d'interruption</b>
Bit6	<b>EEIE</b>	EEPROM Interrupt Enable : Interruption à la fin d'écriture sur l'EEPROM
Bit5	<b>TOIE</b>	Timer0 Overflow Interrupt Enable : Interruption suite au débordement du Timer0
Bit4	<b>INTE</b>	RB0 Interrupt Enable : <b>autorise l'interruption sur RB0</b>
Bit3	<b>RBIE</b>	PortB Interrupt Enable : <b>autorise l'interruption suite au changement de RB4..7</b>
Bit2	<b>TOIF</b>	Timer0 Overflow Interrupt Flag : drapeau mis à 1 suite au débordement du Timer0 (ff-00)
Bit1	<b>INTF</b>	RB0 Interrupt Flag : <b>drapeau mis à 1 si l'interruption sur RB0 est active</b>
Bit0	<b>RBIF</b>	PortB Interrupt Flag : <b>drapeau mis à 1 si l'interruption sur RB4..7 est active</b>

## 3- Configuration registre Option\_reg :

### OPTION-REG :



**\*RBPU** : mode Pull up :Ce bit est utilisé sauf si le port B est configuré en entrée (TRISB = 0)

-RBPU = 1 : le mode est désactivé (portb fonctionne avec la logique positive : **1**)

-RBPU = 0 : le mode est activé (portb fonctionne avec la logique négative : **0**)

**\*INTEDG** : choisir le type du front de RBO/INT pour provoquer une interruption

-INTEDG = 1 : front montant sur RBO

-INTEDG = 0 : front descendant sur RBO

**\*TOCS** : choisir la source d'horloge du TMR0 (ou le mode de fonctionnement)

-TOCS = 1 : l'horloge externe (RA4) « mode compteur »

-TOCS = 0 : l'horloge interne « mode temporisateur »

**\*TOSE** : choisir le type du front de RA4 en mode compteur

-TOSE = 0 : le TMR0 s'incrémente à chaque front montant sur RA4

-TOSE = 1 : le TMR0 s'incrémente à chaque front descendant sur RA4

**\*PSA** : Affecter le Pré-diviseur

-PSA = 0 : le pré-diviseur est affecté au watchdog

-PSA = 1 : le pré-diviseur est affecté au TMR0

**\*PS2, PS1 et PS0** : choisir le pré-diviseur selon le tableau suivant

<b>PS0</b>	0	1	0	1	0	1	0	1
<b>PS1</b>	0	0	1	1	0	0	1	1
<b>PS2</b>	0	0	0	0	1	1	1	1
<b>Pré-diviseur</b>	2	4	8	16	32	64	128	256

#### 4- Tableaux code ASCII des caractères :

b7- b3- b0	0000	0010	0011	0100	0101	0110	0111	1010	1011	1100	1101	1110	1111
CG/RAM (1)		0	1	2	3	4	5	6	7	8	9	A	B
(2)	!	1	A	Q	a	q	.	7	z	4		ä	q
(3)	"	2	B	R	b	r	「	イ	ツ	×		ß	0
(4)	#	3	C	S	c	s	」	ウ	テ	モ		ø	ø
(5)	\$	4	D	T	d	t	、	エ	ト	フ		µ	Ω
(6)	%	5	E	U	e	u	・	オ	ナ	1		€	Ü
(7)	&	6	F	V	f	v	ヲ	カ	ニ	ヨ		ρ	Σ
CG/RAM (8)	'	7	G	W	g	w	フ	キ	ヌ	ラ		g	π
CG/RAM (1)	(	8	H	X	h	x	イ	ウ	ネ	リ		J	×
(2)	)	9	I	Y	i	y	ウ	エ	ケ	ル		°	y
(3)	*	:	J	Z	j	z	エ	コ	ン	レ		j	¢
(4)	+	;	K	[	k	[	オ	サ	ヒ	ロ		°	¥
(5)	,	<	L	¥	l	l	ハ	シ	フ	ワ		φ	円
(6)	-	=	M	]	m	]	ユ	ズ	ヘ	ン		£	÷
(7)	.	>	N	^	n	^	ヨ	セ	ホ	°		ñ	
CG/RAM (8)	/	?	O	_	o	_	ウ	リ	マ	°		ö	■

Section : ..... Classe : ..... N° : .....  
 Nom et Prénom : .....

**ETUDE DE COMTAGES DES PIECES :**

En se référant au schéma structurel de module de comptage voir dossier technique page 2 :

1- Identifier le type de l'oscillateur utilisé : (0.25pt)

.....

2- Que se passe t il si on appui sur le bouton RAZ : (0.25pt)

.....

3- Indiquer le type et le rôle de chacun des transistors suivants : (1pt)

Transistors	Q1	Q2	Q3	Q4
Rôle	..... .....	..... .....	..... .....	..... .....
Type	..... .....	..... .....	..... .....	..... .....

4- Quel est le rôle du CI 7447 ; (0.25pt)

.....

5- Identifier le type des afficheurs utilisés pour le comptage des bonnes pièces : (0.25pt)

.....

6- Compléter la configuration des registres TRIS correspondant à notre application : (1.5pts)

TRISA	-	-	RA5	RA4	RA3	RA2	RA1	RA0
	.....	.....	.....	.....	.....	.....	.....	.....

TRISB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0
	.....	.....	.....	.....	.....	.....	.....	.....

TRISC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0
	.....	.....	.....	.....	.....	.....	.....	.....

7- Le comptage des mauvaises pièces est assuré par une interruption sur la broche RB4 :  
 a- Pour activer cette interruption il faut écrire dans le registre INTCON la combinaison suivante : (0.25pt)

INTCON	...	...	...	...	...	...	...	...	...	=\$.....
--------	-----	-----	-----	-----	-----	-----	-----	-----	-----	----------

b- Si l'interruption est en cours d'exécution ce registre prend automatiquement la combinaison suivante : (0.25pt)

INTCON	...	...	...	...	...	...	...	...	...	=\$.....
--------	-----	-----	-----	-----	-----	-----	-----	-----	-----	----------

c- Pour sortir de l'interruption et revenir au programme principal il faut écrire la combinaison suivante: (0.25pt)

INTCON	...	...	...	...	...	...	...	...	...	=\$.....
--------	-----	-----	-----	-----	-----	-----	-----	-----	-----	----------

8- Le comptage des bonnes pièces est assuré par le TMR0 qui s'incrémente à chaque front Montant sur la broche RA4, Compléter alors la configuration du registre option \_reg : (0.5pt)

Option_reg	...	...	...	...	...	...	...	...	...	=\$.....
------------	-----	-----	-----	-----	-----	-----	-----	-----	-----	----------

9- En s'aidant des questions précédentes et en respectant les conditions du cahier de charge (voir dossier technique page 2), compléter le programme correspondant : (5pt)

<u>programme</u>	<u>commentaires</u>	<u>Programme</u>	<u>commentaires</u>
program compteur;	Nom programme	.....	Configurer TMRO
var i ,j,k,etat,uni,dix:byte;	Déclarations des variables	j:=0;	
const	Déclarations des constantes	i:=0;	
aucun_afficheur :byte=3;		tmr0:=0;	.....
afficheur_1 :byte=2;		while true do	.....
afficheur_2 :byte=1;		begin	
procedure interrupt ;	.....	i :=tmr0;	
etat:=portb ;	.....	if tmr0=99 then	
.....	. Afficher contenu Compteur sur portA	begin	
j:=j+1;	.....	portb.6:=1;	.....
if j=10 then		.....	Pause 1s
begin		portb.6:=0;	<b>Eteindre</b> la lampe L2
.....	Initialiser le compteur	end;	
.....	Allumer L1	if tmr0=100 then tmr0:=0;	.....
delay_ms(5000);	.....	.....	Calculer les unités
portb.7:=0;	.....	dix:=(i div 10)mod 10;	.....
end;		for k:=0 to 10 do	
.....	Desactiver le Flag	begin	
end;		portb:= aucun_afficheur;delay_ms(1)	Désactiver les afficheurs Pendant 1ms
begin		portc:=uni;portb:= afficheur_1 ; delay_ms(10);	.....
.....	Configuration des Registres TRIS	.....	Désactiver les afficheurs Pendant 1ms
.....		portc:=dix;portb:= afficheur_2 ; delay_ms(10);	Afficher les dizaines
.....		end;	
intcon:=\$..... ;	Activer l'interruption	end;	
.....	Initialiser les sorties	end.	
.....	PortA numérique		

## II- ETUDE DE CONTROLE ET COMMANDE NUMERIQUE DE LA TEMPERATURE :

- 1- En se référant au dossier technique page 3 trouver la relation entre la tension V(mV) et la température T(°C) : (0.75pt)

---

- 2- Trouver la relation entre V et N(résultat de la conversion), en déduire celle de la température et N. (0.75pt)

---

- 3- Donner alors la configuration convenable a notre application du registre ADCON1 : (0.75pt)

ADCON1	...	...	...	...	...	...	...	...	= \$.....
--------	-----	-----	-----	-----	-----	-----	-----	-----	-----------

- 4- Compléter alors le programme qui permet la commande du four : (4pts)

program four;	Commentaires
var LCD_RS : sbit at RB2_bit; LCD_EN : sbit at RB3_bit; LCD_D4 : sbit at RB4_bit; LCD_D5 : sbit at RB5_bit; LCD_D6 : sbit at RB6_bit; LCD_D7 : sbit at RB7_bit; LCD_RS_Direction : sbit at TRISB2_bit; LCD_EN_Direction : sbit at TRISB3_bit; LCD_D4_Direction : sbit at TRISB4_bit; LCD_D5_Direction : sbit at TRISB5_bit; LCD_D6_Direction : sbit at TRISB6_bit; LCD_D7_Direction : sbit at TRISB7_bit;	connections de l'afficheur Lcd
valeur_conversion : word ;	.....
variable_calcul : ..... ;	4 octets pour le calcul pour ne pas avoir un dépassement de taille lors de la multiplication
Tc, Tf, erreur : .....;	1 octet car la température est comprise entre 0 et 250
valeur_afichage : string[3];	.....
begin	
trisc:=0; portc:=0;	.....
.....	initialisation de l'afficheur LCD
LCD_cmd(_LCD_CURSOR_OFF);	.....
ADCON1:=\$.....;	.....
..... ;	initialisation du convertisseur CAN
LCD_out(1,1,'Tf:');	.....
LCD_out(2,1,'Tc:');	.....
while true do	
begin	
valeur_conversion := adc_read(0);	lecture de la consigne de température Tf
variable_calcul := ..... .....	
Tf:= byte(variable_calcul );	.....
byteToStr(Tf,valeur_afichage);	conversion de la température calculée en texte
lcd_out(1,4,valeur_afichage);	affichage de la température Tf
lcd_chr(1,7,%11011111);	affichage du symbole degré: °



lcd_chr(1,8,'C');	// affichage de C pour Celsius
valeur_conversion := .....	// lecture de la consigne de température Tc
variable_calcul := (valeur_conversion * 250)/1023 ;	
Tc:= .....	
byteToStr(Tc,valeur_afichage);	conversion de la température calculée en texte
lcd_out(2,4,valeur_afichage);	affichage de la température Tc
lcd_chr(2,7,.....);	affichage du symbole degré: °
lcd_chr(2,8,'C');	affichage de C pour Celsius
erreur := .....	calcul de l'erreur
if erreur >0 then ..... else .....	
end;..	
end.	

### III- ETUDE DE CONTROLE ET COMMANDE ANALOGIQUE DE LA TEMPERATURE :

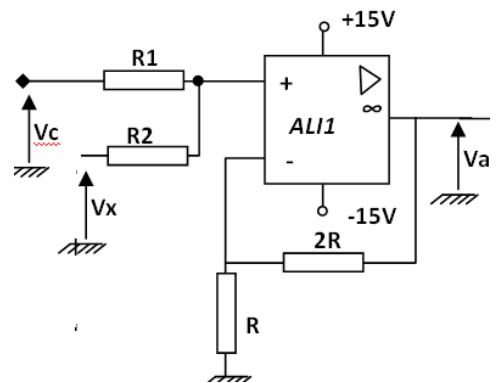
En se référant, dans cette partie, au schéma structurel de la page 5 du dossier technique.

1- Mettre une croix devant la bonne réponse: (1.5pts)

	Polarisation		Régime		Boucle	
	Asymétrique	Symétrique	linéaire	Saturé	Ouvert	Fermé
ALI1						
ALI2						
ALI3						

2- Exprimer Va en fonction de Vc et Vx : (1.5pt)

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

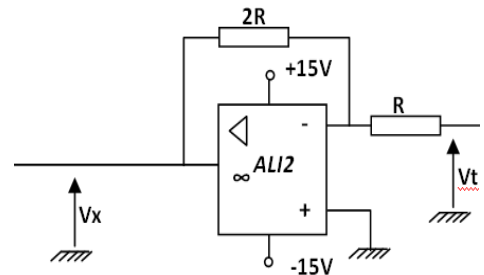


3- En déduire la fonction réalisée par ALI1 : (0.5pt)

.....

5- Exprimer Vx en fonction de Vt : (1.5pts)

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



6- En déduire la fonction réalisée par ALI2 : (0.5pt)

.....