

الامتحان الوطني الموحد للبكالوريا
المسالك المهنية
الدورة العادية 2023

PPPPPPPPPPPPPPPPPPPPPP-PPP

الموضوع

NS 216A

4h

مدة الإجابة

اختبار توافقي في المواد المهنية (الجزء الأول) - الفترة الصباحية

المادة

10

المعامل

شعبة الهندسة الكهربائية مسلك النظم الإلكترونية والرقمية

الشعبة أو المسلك

GYROPODE

☞ Le sujet comporte au total 18 pages.

☞ Le sujet comporte 3 types de documents :

- Pages 02 à 07 (Couleur Verte) : Socle du sujet comportant les situations d'évaluation (SEV) ;
- Pages 08 à 11 (Couleur Rose) : Documents ressources portant la mention **DRES XX** ;
- Pages 12 à 18 (Couleur Blanche) : Documents réponses portant la mention **DREP XX**.

☞ Le sujet comporte 4 situations d'évaluation (SEV) :

- SEV1 : ÉTUDE PARTIELLE DE LA CHAÎNE D'ACQUISITION ET DE CONDITIONNEMENT /21 pts
- SEV2 : ÉTUDE PARTIELLE DE LA CHAÎNE DE TRAITEMENT ET D'AFFICHAGE/36 pts
- SEV3 : ÉTUDE PARTIELLE DE L'ALIMENTATION DE LA CARTE ÉLECTRONIQUE/8 pts
- SEV4 : NOTIONS SUR LA MICRO-ÉLECTRONIQUE ET LA NANOTECHNOLOGIE/5 pts

☞ N.B :

Les quatre SEV sont indépendantes et peuvent être traitées dans un ordre quelconque après lecture de la présentation, de la description et du fonctionnement.

La numérotation des questions est continue : de la question 1 (Q.1) à la question 45 (Q.45).

- Toutes les réponses doivent être rédigées sur les documents réponses **DREP XX** ;
- Les pages portant en haut la mention **DREP XX** doivent être obligatoirement jointes à la copie du candidat même si elles ne comportent aucune réponse ;
- Le sujet est noté sur 70 points.

- *Aucun document n'est autorisé.*
- *Sont autorisées les calculatrices non programmables.*

1. PRÉSENTATION DU SYSTÈME :

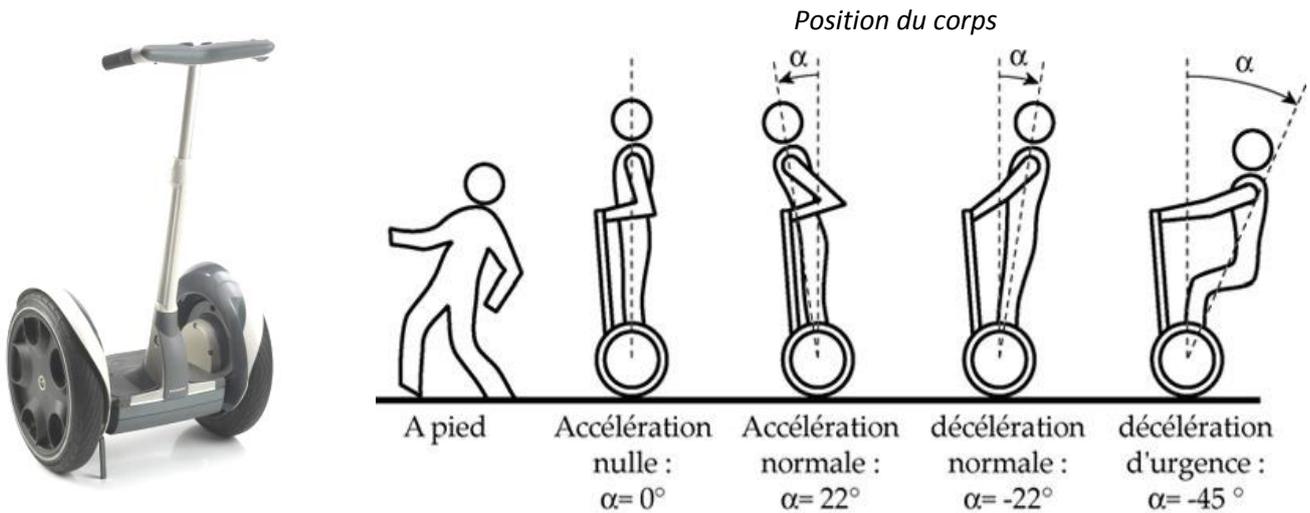


Figure 1 : Postures de commande du gyropode.

Un gyropode est un véhicule électrique monoplace silencieux et non polluant, on l'appelle aussi transporteur personnel, sa fonction est de déplacer une charge utile maximale (une personne) de 100 kg sur un chemin goudronné, de pente maximale 5 %, à une vitesse maximale de 30 km/h en toute sécurité et une portée moyenne pouvant aller jusqu'à 40 km. Le temps pour recharger les batteries, via une prise secteur classique est de 3 à 5 h.

2. DESCRIPTION DU SYSTÈME : (voir *Figure 1* ci-dessus et le document *DRES 01*)

Le gyropode, objet de l'étude, se compose de :

- Un **chariot** (châssis + 2 roues) ;
- Deux **motoréducteurs** entraînant les roues (un par roue) ;
- Un **ensemble** constitué :
 - D'un **gyromètre** délivrant une information sur la vitesse de déviation ;
 - D'un **accéléromètre** délivrant une information sur l'angle d'inclinaison α , du châssis par rapport à la verticale, qui renseigne sur la vitesse de déplacement.
- Une **carte électronique** à base d'un microcontrôleur **PIC16F877** élaborant, à partir des informations issues des capteurs, les consignes de commande des groupes motoréducteurs ;
- Des **batteries** fournissant l'énergie aux divers composants ;
- Un **système d'affichage LCD** pour l'affichage de la vitesse et le niveau des batteries ;
- Un **capteur de vitesse** permettant d'acquérir la vitesse de rotation ;
- Un **détecteur** de présence.

3. FONCTIONNEMENT :

Par l'intermédiaire de son corps, l'utilisateur génère l'inclinaison des plateformes par rapport à la verticale de manière indépendante (voir *Figure 1*). La mesure de cette inclinaison (**angle α**), est interprétée par le microcontrôleur comme une consigne de vitesse. Ce dernier génère alors le signal de commande des moteurs.

Le gyropode se trouve en permanence en équilibre dynamique. En cas de chute de la source d'alimentation au-dessous d'un seuil limite, le microcontrôleur donnera l'ordre de ralentissement jusqu'à l'arrêt complet du gyropode.

L'étude sera limitée à un déplacement rectiligne du gyropode.

4. SITUATIONS D'ÉVALUATION :

SEV 1

ACQUISITION ET CONDITIONNEMENT

21 POINTS

Pour mesurer la vitesse du moteur, on utilise un codeur incrémental solidaire à son arbre de rotation.

Le codeur est constitué d'une roue dentée, ayant **15 trous** équidistants, la roue est partiellement insérée dans une fourche optique.

La fourche optique est un capteur constitué d'un dispositif d'émission de lumière infrarouge (LED) et d'un récepteur (phototransistor) placés en vis à vis (voir **Figure 2**).

La roue dentée tourne dans la fourche (voir **Figure 3**), la coupure périodique du faisceau lumineux provoque une impulsion à chaque passage de dent.

Le capteur délivre une tension U_T périodique dont la fréquence est proportionnelle à la vitesse de rotation du moteur.

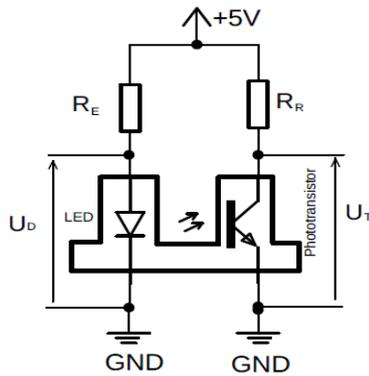


Figure 2 : fourche optique

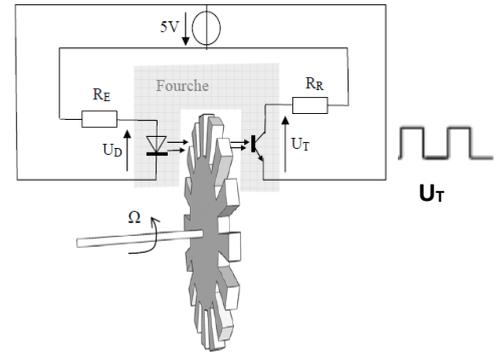


Figure 3 : codeur incrémental

Q.1- Quel est le rôle du codeur incrémental dans le gyropode ?

1 pt

Q.2- Quelle est le rôle de la fourche optique ?

1 pt

Q.3- La fourche optique est un capteur actif ou passif ? Justifier votre réponse.

2 pts

Q.4- La tension aux bornes de la LED est $U_D = 1,2 \text{ V}$. Déterminer la valeur de R_E permettant d'établir un courant d'intensité $I_D = 10 \text{ mA}$ dans le circuit d'émission de lumière.

2pts

On considère que le phototransistor fonctionne comme un interrupteur :

- **Fermé** quand il reçoit de la lumière : transistor est saturé, $V_{CE} = 0 \text{ V}$ et le courant dans le collecteur $I_C = 1 \text{ mA}$;
- **Ouvert** quand il ne reçoit pas de lumière : transistor est bloqué, $V_{CE} = 5 \text{ V}$ et le courant dans le collecteur $I_C = 0 \text{ A}$.

Q.5- Déterminer la valeur de la résistance R_R .

1 pt

Q.6- Quel est le rôle de la résistance R_R ?

1 pt

Q.7- Complétez le tableau.

4 pts

Q.8- Quel est le nombre d'impulsions N_R fournies par le codeur pour **1 tour** de rotation du moteur.

1 pt

Q.9- Donner le nom de la caractéristique du codeur représenté par le nombre N_R .

1 pt

Le moteur tourne à une vitesse de rotation $n = 360 \text{ tr/min}$:

Q.10- Calculer la période de rotation du moteur T_m (en ms).

2 pts

Q.11- Déduire la période T_i (en ms) et la fréquence f_i (en Hz) des impulsions.

2 pts

Q.12- Déduire la période T (en ms) du signal U_T .

1 pt

Q.13- Tracer l'oscillogramme du signal U_T à la sortie du capteur optique pour un seul tour de la roue, en précisant sur cet oscillogramme les périodes T_i et T_m .

2 pts

SEV 2

CHAÎNE DE TRAITEMENT ET D'AFFICHAGE

36 POINTS

Le schéma de principe du système étudié est donné comme suit :

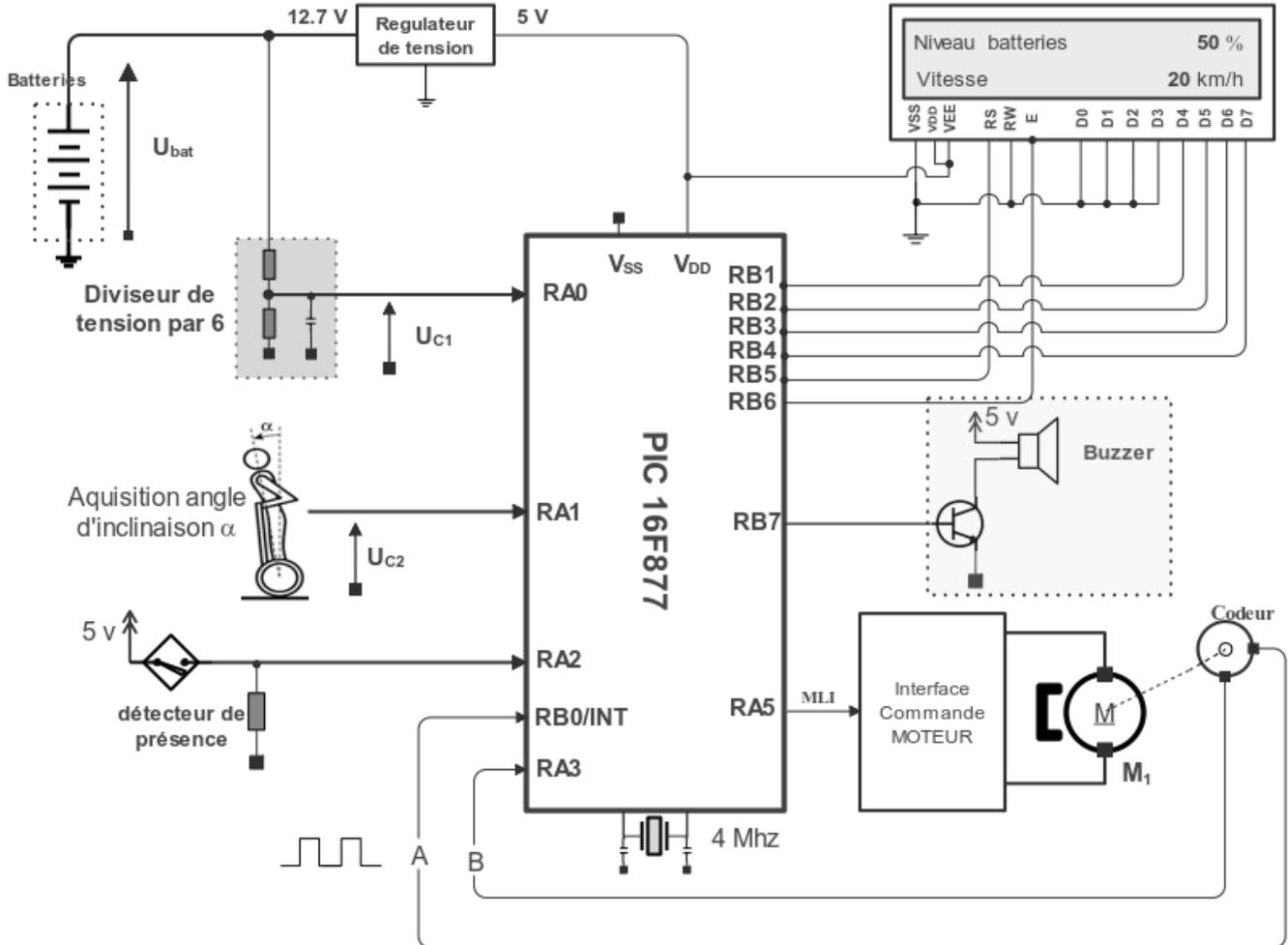


Figure 4 : Schéma de principe du Gyropode

Ce schéma permet :

- ✓ L'acquisition de la tension issue des batteries ;
- ✓ L'acquisition de l'angle d'inclinaison (α) ;
- ✓ L'acquisition de la vitesse ;
- ✓ La commande de l'afficheur **LCD 2x32** ;
- ✓ La commande du moteur **M1** ;
- ✓ La génération d'un signal sonore (Buzzer).

L'organigramme de la chaîne de traitement et d'affichage est donné sur (*DRES 02, page 9*) avec :

- **Acquisition_Vbatt** : Sous-programme qui convertit le signal U_{c1} image de la tension U_{BAT} en un nombre N_{batt} sur 8 bits, stocké dans la case mémoire **adr_Nbat** ;
- **Aquisition_angle_alpha** : Sous-programme qui convertit l'angle d'inclinaison α en un nombre signé N_{α} sur 8 bits, stocké dans la case mémoire **adr_Nalpha** ;
- **Affichage_LCD** : Sous-programme qui informe l'utilisateur via un afficheur **LCD 2x32** sur le niveau de charge des batteries (en %), ainsi sur la vitesse de déplacement du gyropode (en **km/h**) ;
- **SP_buzzer** : Sous-programme qui informe l'utilisateur par une signalisation sonore que la charge est insuffisante (batteries déchargées) ;

- **Commande_MLI** : Sous-programme qui génère un signal MLI (Modulation de Largeur d'Impulsion) pour commander le moteur **M1** ;
- **Commande_Arrêt** : Sous-programme qui ralentit puis arrête le gyropode.

Vu l'aspect temps réel de la mesure de la vitesse du gyropode, alors elle sera traitée par *programme d'interruption*, qui détermine la vitesse du Gyropode, le résultat est ensuite stocké dans la case mémoire **Vitesse_Gyropode**.

☞ *Les sous-programmes Commande_Arrêt, Commande_MLI et SP_buzzer ne seront pas traités.*

Tâche 1 : Traduction de l'organigramme de fonctionnement en assembleur :

Q.14- A partir de la **Figure 4 page 4**, identifier la nature (*digitale ou analogique*) des entrées **RA0, RA1, RA2** et **RB0**. **1 pt**

Q.15- Déterminer les mots en binaire et en hexadécimal à donner aux registres **TRISA** et **TRISB** pour configurer les différents ports utilisés dans le système (les bits non utilisés sont *mis à 1*). **2 pts**

Q.16- En utilisant le document (**DRES 03, page 10**), Compléter alors le programme Assembleur relatif à l'organigramme de fonctionnement du gyropode. **4 pts**

Tâche 2 : Surveillance de la tension des batteries (sous -programme Acquisition_Vbatt) :

Le gyropode se trouve en permanence en équilibre dynamique. La défaillance de la source provoque la chute de la personne. Cette défaillance risque de se produire lorsque la tension aux bornes des batteries chute de plus de **10 %** de la valeur nominale (**12,7 V**).

On souhaite que lors d'une décharge trop importante des batteries, le **PIC** donnera l'ordre de ralentissement puis d'arrêt du gyropode.

La tension continue U_{bat} est appliquée à l'entrée **RA0** par l'intermédiaire d'un pont diviseur de tension qui divise la tension par **6** (voir **Figure 4, page 4**).

Q.17- Calculer $U_{bat,d}$ la valeur de la tension U_{bat} qui doit déclencher l'arrêt du gyropode. **1 pt**

Q.18- Sachant que $U_{bat,d} = 11,43 V$, calculer la tension $U_{C1,d}$ qui doit déclencher l'arrêt. **1 pt**

La tension U_{C1} est convertie par le module **CAN** (Convertisseur Analogique Numérique) intégré au microcontrôleur, la valeur numérique **N_batt** résultat de la conversion est donnée par la relation suivante :

$$N_{batt} = 255 \cdot \frac{U_{C1} - V_{ref-}}{V_{ref+} - V_{ref-}}$$

Q.19- Sachant que $V_{REF+} = 5 V$ et $V_{REF-} = 0 V$. Calculer la valeur décimale du nombre **N_batt**, pour $U_{bat} = 12,7V$ puis pour $U_{bat} = 11,43 V$. **1,5 pt**

La valeur **N_batt** résultat de la conversion est stockée dans le registre **ADRESH** qui sera transférée par la suite à la case mémoire **Adr_Nbat**.

Q.20- Compléter le sous-programme **Acquisition_Vbatt** correspondant à l'organigramme de la **Figure 7** du document **DRES 01**. **3 pts**

Tâche 3 : Acquisition de la vitesse :

Vu l'aspect temps réel de la mesure de la vitesse, alors elle sera traitée par interruption.

Cette interruption est générée :

- ✓ Chaque fois que le **Timer0** associé au Prescaler (Pré-diviseur) atteint **65536 μ s** ;
- ✓ Chaque fois que l'entrée **RB0/INT** reçoit un front montant du signal **A** issue du codeur.

Dans ce programme d'interruption :

- ✓ Une case mémoire **Count_A** s'incrmente à chaque période du signal **A** du codeur ;
- ✓ Une case mémoire **Count_Is** initialisée à une valeur **X1** se décrmente à chaque fois que le **Timer0** associé au Prescaler atteint la valeur **65536**. Il en résulte alors, à une bonne approximation près, qu'à chaque **seconde** la case mémoire **Count_Is** passe à **0** ;
- ✓ Un sous-programme **Calcul_Vitesse_Gyropode** détermine la vitesse du groupe propulsion du Gyropode à partir de la case mémoire **Count_A**, le résultat est ensuite stocké dans une case mémoire **Vitesse_Gyropode**.

Q.21- Sachant que $f_{osc} = 4 \text{ MHz}$, calculer la durée T_c d'un cycle machine. ($f_c = \frac{f_{osc}}{4}$)

1 pt

Q.22- Calculer la valeur du pré-diviseur $N_{pré-div}$ pour avoir **65536 μ s** ($65536 = 256 \times 256$).

1 pt

Q.23- Sachant que la case mémoire **Count_Is**, se décrmente à partir de **X1** et passe à **0** après l'écoulement d'une seconde, calculer la valeur **X1**.

2 pts

Q.24- En utilisant les documents (**DRES 03** et **DRES 04**), compléter le programme Assembleur du programme d'interruption.

4 pts

Q.25- Lors d'une routine d'interruption on doit sauvegarder éventuellement quelques registres, parmi les registres proposés cocher les bonnes réponses.

1 pt

Tâche 4 : Programmation du PIC en MikroC :

Q.26- Compléter les lignes de définition du **PIC16F877** en MikroC.

1,5 pt

Q.27- Compléter les lignes de déclarations des variables externes.

1,5 pt

Q.28- Compléter en MikroC les lignes de configuration de la fonction **Acquisition_Vbatt** relative à l'organigramme **Figure 7** du document **DRES 01**.

2 pts

La gestion de l'affichage dans **LCD** est comme suit :

- On affiche dans **la ligne 1** le niveau de charge des batteries en % ;
- On affiche la vitesse de déplacement du gyropode en (**km/h**) dans **la ligne 2**.

La vitesse de déplacement du gyropode est stockée dans la case mémoire **Vitesse_Gyropode**.

Le niveau de charge des batteries est stocké dans la case mémoire **Adr_Nbat**.

Q.29- Compléter les lignes de configuration de la fonction **Affichage_LCD** en MikroC.

3,5 pts

Q.30- Compléter les lignes de la fonction principale correspondant à l'organigramme de fonctionnement du gyropode (**Figure 8, DRES 02**).

5 pts

SEV 3

ALIMENTATION DE LA CARTE ÉLECTRONIQUE

8 POINTS

Pour alimenter la carte électronique du gyropode, un régulateur de tension est utilisé pour convertir la tension des batteries en une tension stable de 5 V.

Le schéma électrique de ce régulateur est le suivant :

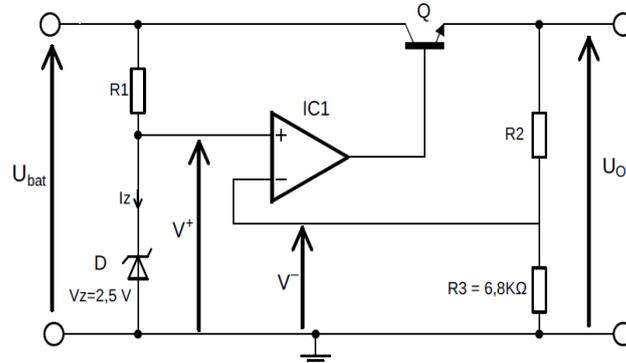


Figure 5 : schéma régulateur de tension

Q.31- Quel est le rôle du transistor Q ?

0,5 pt

Q.32- Quel est le rôle de la diode D ?

0,5 pt

Q.33- Quel est le rôle du diviseur de tension obtenu par R_2 et R_3 ?

0,5 pt

Q.34- Quel est le rôle du circuit $IC1$?

0,5 pt

Q.35- Donner l'expression de la tension V^+ .

1 pt

Q.36- Donner l'expression de la tension V^- en fonction de R_2 , R_3 et U_o .

1 pt

Q.37- Dédurre que $R_2 = R_3 \cdot \left(\frac{U_o}{V_z} - 1 \right)$.

1 pt

Q.38- Calculer la valeur de R_2 pour avoir une tension de sortie $U_o = 5$ V.

1 pt

La diode Zener D a une tension $V_z = 2,5$ V et une puissance $P_z = \frac{1}{4}$ W, $U_{bat} = 12,7$ V

Q.39- Calculer le courant I_z .

1 pt

Q.40- Déterminer la valeur de R_1 .

1 pt

SEV 4

MICRO-ÉLECTRONIQUE ET NANOTECHNOLOGIE

5 POINTS

Q.41- Cocher la bonne réponse.

1 pt

Q.42- Cocher la bonne réponse.

1 pt

Q.43- Cocher la bonne réponse.

1 pt

Q.44- Cocher la bonne réponse.

1 pt

Q.45- Mettre une croix dans la case convenable.

1 pt

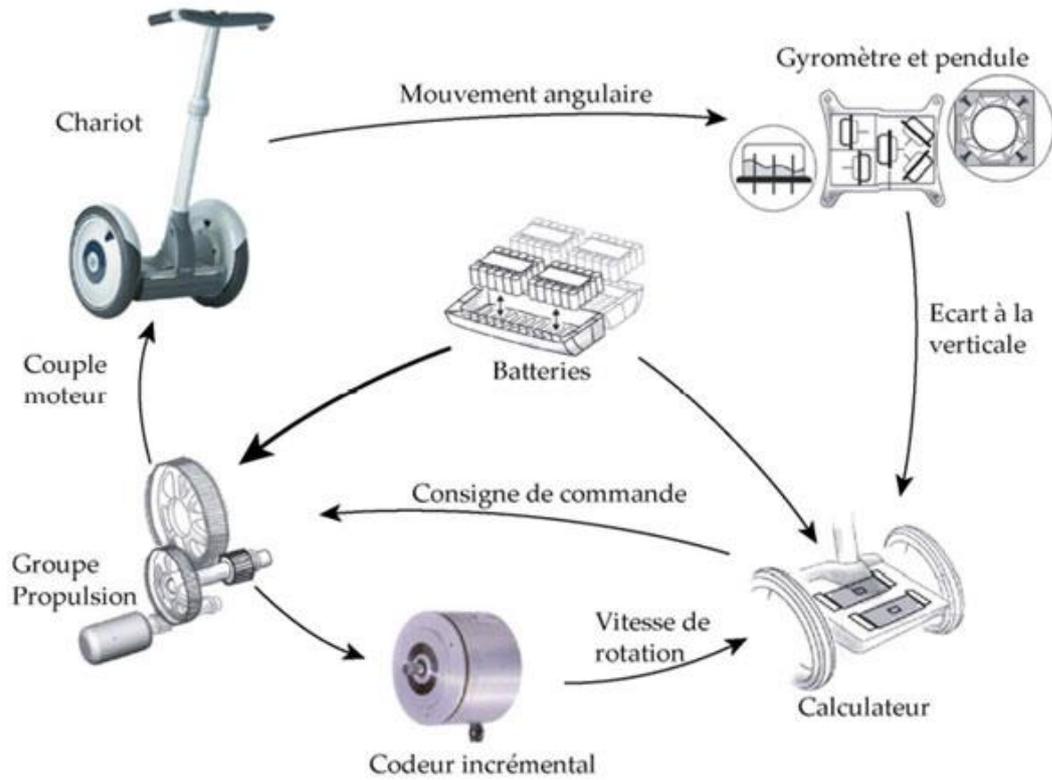


Figure 6 : Structure du gyropode

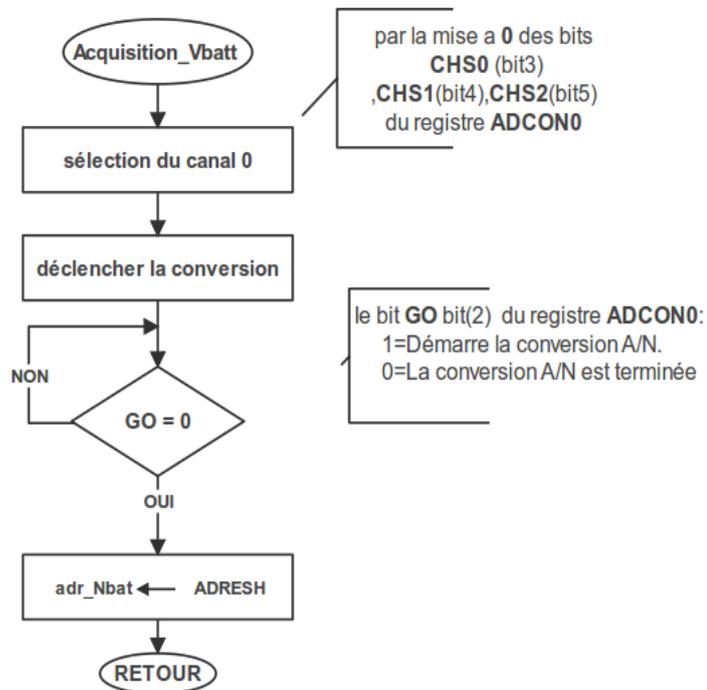


Figure 7 : Organigramme Acquisition_Vbatt

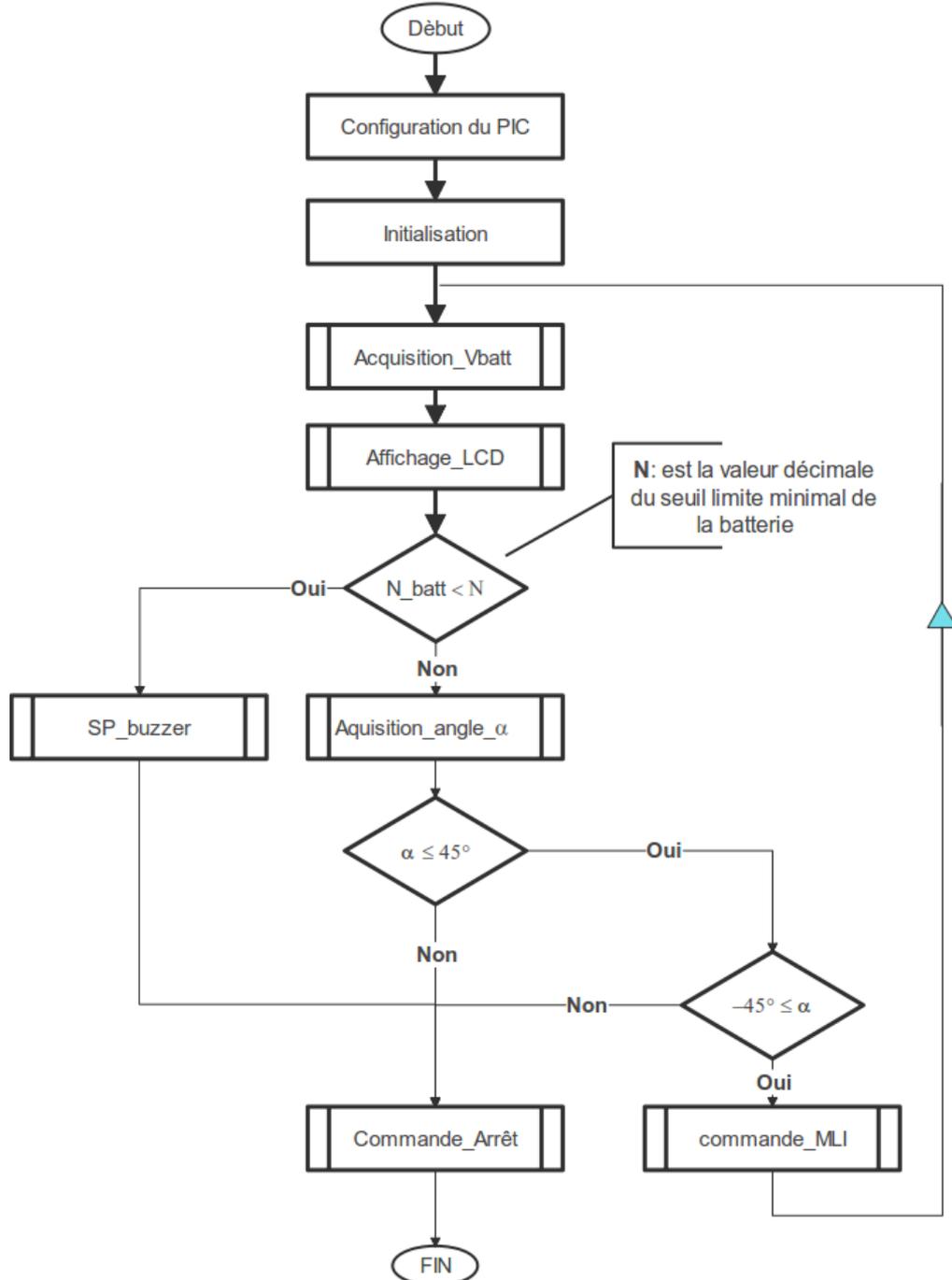


Figure 8: Organigramme de fonctionnement

Jeu d'instructions du microcontrôleur 16F877

DRES 03

1. Jeu d'instructions :

INSTRUCTIONS OPERANT SUR UN REGISTRE			Indicateurs	Cycles
ADDWF	F, d	W+F → {W, F ? d}	C, DC, Z	1
ANDWF	F, d	W AND F → {W, F ? d}	Z	1
CLRF	F	0 → F	Z	1
CLRW		0 → W	Z	1
CLRWD		0 → WDT	TO', PD'	1
COMF	F, d	Complément F → {W, F ? d}	Z	1
DECF	F, d	Décrémente F → {W, F ? d}	Z	1
DECFSZ	F, d	Décrémente F → {W, F ? d} et saut si 0		1(2)
INCF	F, d	Incrémente F → {W, F ? d}	Z	1
INCFSZ	F, d	Incrémente F → {W, F ? d} et saut si 0		1(2)
IORWF	F, d	W OR F → {W, F ? d}	Z	1
MOVF	F, d	F → {W, F ? d}	Z	1
MOVWF	F	W → F		1
RLF	F, d	Rotation à gauche de F à travers C → {W, F ? d}	C	1
RRF	F, d	Rotation à droite de F à travers C → {W, F ? d}		1
SUBWF	F, d	F - W → {W, F ? d}	C, DC, Z	1
SWAPF	F, d	Permute les 2 quartets de F → {W, F ? d}		1
XORWF	F, d	W XOR F → {W, F ? d}	Z	1
INSTRUCTIONS OPERANT SUR UN BIT				
BCF	F, b	Mise à 0 du bit b du registre F		1
BSF	F, b	Mise à 1 du bit b du registre F		1
BTFSC	F, b	Teste le bit b de F et saut si 0		1(2)
BTFSS	F, b	Teste le bit b de F et saut si 1		1(2)
INSTRUCTIONS OPERANT SUR UNE DONNEE				
ADDLW	K	W + K → W	C, DC, Z	1
ANDLW	K	W AND K → W	Z	1
IORLW	K	W OR K → W	Z	1
MOVLW	K	K → W		1
SUBLW	K	K - W → W	C, DC, Z	1
XORLW	K	W XOR K → W	Z	1
INSTRUCTIONS GENERALES				
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		Mode standby	TO', PD'	1

Figure 9

2. Configuration des PORTS :

Tous les ports sont pilotés par deux registres : TRISx et PORTx

- 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) ;
- Les registres TRISx appartiennent à la BANK 1 des SFR. Lors de l'initialisation du µC, il ne faut pas oublier de changer de bank mémoire pour les configurer.

3. Description du registre d'état STATUS :

IRP	RP1	RP0	/TO	/PD	Z	DC	C
-----	-----	-----	-----	-----	---	----	---

- Pour le passage entre les BANKs 0 et 1, on positionne les bits correspondants comme suit :
 - ✓ RP1 RP0 = 00 → Accès à la BANK 0 ;
 - ✓ RP1 RP0 = 01 → Accès à la BANK 1.

REGITRE DE CONFIGURATION DES INTERRUPTIONS (INTCON) :

DRES 04

Le registre INTCON (INTerrupt CONtroller) est le registre principal de contrôle et de gestion des interruptions.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
GIE	EEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF

Registre INTCON

GIE : « Global Interrupt Enable » mis à 1 autorise toutes les interruptions non masquées par leur bit individuel.

EEIE : « EEPROM write completed Interrupt Enable » : autorise les interruptions de fin d'écriture dans l'EEPROM.

TOIE : « Timer 0 Interrupt Enable » : mis à 1 autorise les interruptions dues au débordement du timer 0.

INTE : « INTerrupt Enabled » : mis à 1, autorise les interruptions sur RB0/INT. L'interruption a lieu sur le front montant de l'impulsion si le bit INTEG (INTerrupt Edge) du registre OPTION est à 1 ; elle a lieu sur le front descendant si ce bit est à 0.

RBIE : « RB Interrupt Enable » : mis à 1, autorise les interruptions sur RB4 à RB7.

TOIF : « Timer 0 Interrupt Flag » : est mis à 1 en cas de débordement du timer 0.

INTF : « INTerrupt Flag » : est mis à 1 si une interruption est générée sur RB0/INT.

RBIF : « RB Interrupt Flag », est mis à 1 lors d'un changement d'état sur une des lignes RB4 à RB7.

Quelques variables et fonctions sous MIKROC

Variables

unsigned short int	Sur 8 bits	0 à 255
unsigned int	Sur 16 bits	0 à 65535
unsigned long int	Sur 32 bits	0 à 4294967295
signed char	Sur 8 bits	-128 a 127
signed int	Sur 16 bits	-32768 a 32767
float	Sur 32 bits	+/-1.17549435082E-38 à +/- 6.80564774407E38
Char [n]	Chaîne de n caractères	

Quelques fonctions

ADC_Init();	Initialise le module ADC avec les paramètres par défaut
ADC_Read(i);	Lecture de la valeur numérique du signal analogique présent sur le canal i
Delay_ms (valeur);	Temporisation pour une valeur donnée en milliseconde
Delay_us (valeur);	Temporisation pour une valeur donnée en microseconde
Lcd_Init();	Initialise le module LCD
Lcd_Out (ligne, colonne, text);	Affiche le text sur le Lcd à la position spécifiée par la ligne et la colonne spécifiées.
Lcd_chr(row ,column, 'char')	Affiche le caractère char sur le Lcd à la position spécifiée par la ligne et la colonne spécifiées.
Lcd_chr_Cp('char')	Affiche le caractère char sur le Lcd à la position courante du curseur.
FloatToStr (n,txt)	Convertir un nombre réel n en une chaîne de caractère txt
intToStr (n,txt)	Convertir un nombre entier n en une chaîne de caractère txt
wordToStr (n,txt)	Convertir un mot non signé en chaîne
Lcd_Cmd(_LCD_CLEAR)	Clear Lcd Display
Lcd Cmd(_LCD_CURSOR_OFF)	Cursor off
Lcd Cmd(_LCD_CURSOR_ON)	Cursor on
Lcd_Cmd(_LCD_SHIFT_LEFT)	Move text to the left
Lcd_Cmd(_LCD_SHIFT_RIGHT)	Move text to the right

Q.1- Rôle du codeur incrémental dans le gyropode :

.....

Q.2- Rôle de la fourche optique :

.....

Q.3- La fourche optique est un capteur (actif ou passif) : Cocher la bonne réponse.

Passif	actif	Justification :
.....

Q.4- La valeur de R_E :

.....

Q.5- La valeur de la résistance R_R :

.....

Q.6- Le rôle de la résistance R_R :

.....

Q.7- Complétez le tableau :

Partie du disque en face du rayonnement	Rayonnement reçu par le phototransistor (Oui/Non)	Etat du phototransistor (Saturé/bloqué)	Etat de l'interrupteur équivalent du phototransistor (Ouvert/fermé)	Valeur de U_T (V)
Est pleine
A une ouverture

Q.8- Le nombre d'impulsions N_R fournies par le codeur pour **1 tour** de rotation :

.....

Q.9- Le nom de la caractéristique du codeur représenté par le nombre N_R :

.....

Q.10- La période de rotation du moteur T_m (en *ms*) :

.....

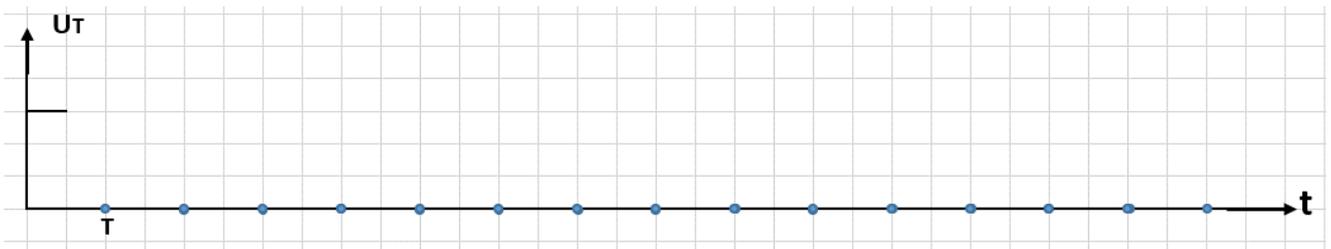
Q.11- La période T_i (en *ms*) et la fréquence f_i (en **Hz**) des impulsions :

.....

Q.12- La période T (en *ms*) du signal U_T :

.....

Q.13- L'oscillogramme du signal U_T avec précision de T_i et T_m :



Q.14- La nature (*digitale* ou *analogique*) des entrées RA0, RA1, RA2 et RB0 :

RA0 :

RA1 :

RA2 :

RB0 :

Q.15- Les mots en binaire et en hexadécimal des registres **TRISA** et **TRISB** :

DREP 02

Bit	Mot en binaire								Mot en hexadécimal
	7	6	5	4	3	2	1	0	
TRISA
TRISB

Q.16- Le programme Assembleur relatif à l'organigramme de fonctionnement du gyropode :

Label	instruction	Commentaire	
	BCF STATUS, RP1	; Activer la page 1 (Bank 1)	
	MOVLW 0XDF MOVWF TRISA MOVLW 0X81	; Configuration PORTA ; Configuration PORTB	
; lignes de Configuration des registres: ADCON1 , OPTION et INTCON			
	BCF STATUS, RP0	; Activer la page 0 (Bank 0)	
; lignes de Configuration du registre : ADCON0			
LAB3	BCF PORTA, RA5	; Initialisation des sorties RB7 et RA5 du système	
	CALL Acquisition_Vbatt	; Appel du sous-programme <i>Acquisition_Vbatt</i>	
	CALL Affichage_LCD	; Appel du sous-programme <i>Affichage_LCD</i>	
	MOVF Adr_Nbat, W	; $W \leftarrow N_batt$	
	SUBLW D'N'	; $W \leftarrow N - N_batt$; tester si le bit C = 1	tester si $N > N_batt$
	GOTO LAB1	; Si oui sauter à LAB1	
	CALL Aquisition_angle_α	; Appel du sous-programme <i>Aquisition_angle_α</i>	
	MOVF Adr_Nα, W	; $W \leftarrow N_α$; $W \leftarrow 45 - N_α$	tester si $N_α < 45$
	BTFSC STATUS, C	; tester si le bit C = 0 ; Si oui sauter à LAB2	
	ADDLW D'45' BTFSC STATUS, C	; $W \leftarrow N_α$; $W \leftarrow 45 + N_α$; tester si le bit C = 0	tester si $-45 < N_α$
	GOTO LAB2	; Si oui sauter à LAB3	
	CALL Commande_MLI	; Appel du sous-programme <i>Commande_MLI</i>	
	GOTO LAB3		
LAB1 LAB2	CALL SP_buzzer	; Appel du sous-programme <i>SP_buzzer</i> ; Appel du sous-programme <i>Commande_Arrêt</i>	
	END		

Q.17- Valeur de la tension U_{bat_d} :

.....

Q.18- Valeur de la tension U_{Cl_d} :

.....

Q.19- Valeurs décimales du nombre N_{batt} :

U_{bat} (V)	11,43	12,7
U_{Cl} (V)	1,905
N_{batt} en décimal

Q.20- Le sous-programme **Acquisition_Vbatt** :

<i>Label</i>	<i>instruction</i>	<i>Commentaire</i>
Acquisition_Vbatt	
	BSF ADCON0, GO	; sélection du canal 0 ; déclencher la conversion
Wait	BTFSC ADCON0, GO	
	; Attendre fin de conversion
	MOVF ADRESH, w	; lecture du résultat
	; $Adr_Nbat \leftarrow W$
	; retour

Q.21- La durée T_c d'un cycle machine :

.....

Q.22- La valeur du pré-diviseur :

.....

Q.23- La valeur $X1$:

.....

Q.24- Le programme Assembleur du programme d'interruption :

<i>Label</i>	<i>instruction</i>	<i>Commentaire</i>
	ORG 0x004	; adresse d'interruption
	;-----sauvegarder les registres-----	
	MOVWF SAVE_W	; sauver registre W
	SWAPF STATUS , W	; swap status avec résultat dans w
	MOVWF SAVE_STATUS	; sauver status swappé
	;-----Traitement de l'interruption de RB0/INT (codeur)---	
	; Test si interruption RB0/INT ?
	GOTO Test_INT_TMR0	; si non , aller Test_INT_TMR0

	; Effacer flag d'interruption INTF
	INCF Count_A, F	;incrémenter Count_A
; -----Traitement de l'interruption de TMR0 (Vitesse Moteur) ---		
Test_INT_TMR0	BTFSS INTCON, TOIF	; Test si interruption Timer 0 ?
	;si non, fin interruption
	BCF INTCON, TOIF	; Effacer flag interruption TOIF
; Mesure de la vitesse		
	; Test d'écoulement d'une seconde ?
	GOTO RestoreStatus	;si non, fin interruption
	; W ← Count_A
	; Appel Calcul_Vitesse_Gyropode
	; Vitesse_Gyropode ← W
;-----Restaurer les registres-----		
RestoreStatus	SWAPF SAVE_STATUS, W	; swap ancien status, résultat dans W
	MOVWF STATUS	; restaurer status
	SWAPF SAVE_W, F	; sans modifier Z
	SWAPF SAVE_W, W	; W restauré
	; retour d'interruption

DREP 04

Q.25- Les registres à sauvegarder lors d'une routine d'interruption : (Cocher les bonnes réponses)

- | | |
|-------------------------------------|---------------------------------|
| <input type="checkbox"/> Registre W | <input type="checkbox"/> ADCON0 |
| <input type="checkbox"/> PORTB | <input type="checkbox"/> STATUS |
| <input type="checkbox"/> TRISC | <input type="checkbox"/> OPTION |

Q.26- Compléter les lignes de définition du PIC16F877 en MikroC :

```

..... //définir la référence du PIC
..... //Fichier de définition des registres de PIC de la famille 16F877
include<delay.h> //Fichier de fonctions prédéfinies pour temporisations logiciels
..... // Directives d'assemblage

```

Q.27- Compléter les lignes de déclaration des variables externes :

```

unsigned int Vitesse_Gyropode ; // déclaration d'un entier non signé nommé Vitesse_Gyropode.
..... // déclaration de deux entiers non signés nommés adr_Nbat et adr_Nα.
..... // déclaration d'une variable réelle N égale 11,43.
int Count_A, Count_1s ; // déclaration de deux entiers nommés Count_A et Count_1s.
..... // déclaration d'une chaîne de 16 caractères nommée txt.

```

Q.28- Compléter les lignes de configuration de la fonction **Acquisition_Vbatt** en MikroC :

```

void Acquisition_Vbatt (void)
{ ADCON0.F3 =0 ;
.....
..... //sélection du canal0

```

```

..... //démarrer la conversion
delay_us(10); //attente de 10 µs
..... //attendre fin conversion
..... //Vitesse_Gyropode ← ADRESH
}

```

Q.29- Les lignes de configuration de la fonction **Affichage_LCD** en MicroC :

```

void ..... (void)

{ // connexion de l'afficheur LCD 2x32
  sbit LCD_RS at RB5_bit;
  sbit LCD_EN at RB6_bit;
  .....
  .....
  sbit LCD_D7 at RB4_bit;
  sbit LCD_RS_Direction at TRISB5_bit;
  sbit LCD_EN_Direction at TRISB6_bit;
  .....
  .....
  .....

  // gestion de l'affichage sur le LCD 2x32
  ..... // Initialisation d'afficheur LCD
  ..... // effacer l'écran LCD
  ..... // curseur OFF
  Lcd_Out (1,1,"Niveau batteries");
  IntToStr (Adr_Nbat,txt) ;
  Lcd_Out (1,24,txt) ;
  Lcd_Chr (1,28,"%");
  ..... // Écrire le texte "vitesse" à la ligne 1, colonne 2
  ..... //Convertir Vitesse_Gyropode en une chaîne de caractère txt
  ..... //afficher la chaîne de caractère txt à la ligne 24,colonne 2
  Lcd_Out (2,28,"km/h");
}

```

Q.30- Les lignes de la fonction principale correspondant à l'organigramme :

```

Void main (void)
{
  ..... // configuration du PortA
  ..... // configuration du PortB
  ..... // mise à 0 du bit RA5
  ..... // mise à 0 du bit RB7
  ..... // appel de la fonction Acquisition_Vbatt
  ..... // appel de la fonction Affichage_LCD

  if (N_batt < N)
  {

```

```

..... //appelle de la fonction SP_buzzer
..... //appelle de la fonction Commande_Arret
return;
}
Else
{ ..... //appelle de la fonction acquisition_angle_alpha
if ( alpha <= 45)
{ if (-45 <= alpha)
{ ..... //appelle de la fonction commande_MLI
return;
}
else
{ Commande_Arret();
return;
}
else
{ Commande_Arret();
return;
}
}
}

```

Q.31- Le rôle du transistor **Q** :

.....

Q.32- Le rôle de la diode **D** :

.....

Q.33- Le rôle du diviseur de tension obtenu par **R₂** et **R₃** :

.....

Q.34- Le rôle du circuit **IC1** :

.....

Q.35- L'expression de la tension **V⁺** :

.....

Q.36- L'expression de la tension **V⁻** en fonction de **R₂**, **R₃** et **U₀** :

.....

Q.37- L'expression de **R₂** :

.....

Q.38- La valeur de **R₂** pour avoir une tension de sortie **U₀ = 5V** :

.....

Q.39- Le courant **I_z** :

.....

Q.40- La valeur de **R₁** :

.....

Cocher la bonne réponse :

DREP 07

Q.41- La microélectronique s'intéresse à :

- La fabrication des composants électroniques à base des produits chimiques ;
- La fabrication de composants électroniques à l'échelle micrométrique à partir de matériaux semi-conducteurs ;
- La fabrication, la manipulation, la caractérisation, l'imagerie, ainsi qu'aux outils de modélisation-simulation utilisant une échelle comprise typiquement entre 1 et 100 nanomètres.

Q.42- Comment appelle-t-on la salle de fabrication des puces dans les laboratoires :

- Salle verte.
- Salle blanche.
- Salle rose.

Q.43- La technologie qui permet l'intégration de nombreuses fonctions électroniques sur un même morceau de silicium (ou autre semi-conducteur) est :

- L'encapsulation.
- L'instrumentation.
- La photolithographie.

Q.44- Pourquoi on procède au nettoyage fréquent des plaquettes de silicium ?

- Pour forcer le dopage des semi-conducteurs.
- Pour éliminer les impuretés déposées.
- Pour réaliser des connexions conductrices entre zones.
- Pour arracher un atome de la surface par collision élastique avec un ion incident.

Q.45- Mettre une croix dans la case convenable :

L'intégration de la nanotechnologie dans les technologies d'affichage présente :	Inconvénient	Avantage
Une consommation énergétique plus faible que les écrans LCD.
Une mise en boîtier onéreuse requise, car les molécules sont sensibles à l'humidité.

الامتحان الوطني الموحد للبكالوريا
الممالك المهنية
الدورة العادية 2023

PPPPPPPPPPPPPPPPPPPP-PPP

مخاض الإجابة

NR 216A

4h مدة الإجابة

اختبار توليفي في المواد المهنية (الجزء الأول) - الفترة الصباحية

المادة

10 المعامل

شعبة الهندسة الكهربائية مسلك النظم الإلكترونية والرقمية

الشعبة أو المملك

Eléments de correction

Q.1- Rôle du codeur incrémental dans le gyropode : **1pt**

Mesure la vitesse de rotation du moteur

Q.2- Rôle de la fourche optique : **1pt**

détecte la position de la roue dentée

Q.3- La fourche optique est un capteur (actif ou passif) : cocher la bonne réponse **2pts (1pt x2)**

Passif	actif	Justification : son principe est fondé sur un effet physique photoélectrique .
	X	

Q.4- La valeur de R_E : **2pts**

$$R_E = (5-1,2) / 0,010 = 380 \Omega.$$

Q.5- Valeur de la résistance R_R . **1pt**

$$R_R = (5-0)/0,001 = 5000 \Omega = 5 k\Omega$$

Q.6- Rôle de la résistance R_R **1pt**

Limiter le courant dans le transistor (protection)

Q.7- Complétez le tableau : **4pts (0,5pt x8)**

Partie du disque en face du rayonnement	Rayonnement reçu par le phototransistor (Oui/Non)	Etat du phototransistor (Saturé/bloqué)	Etat de l'interrupteur équivalent du phototransistor (Ouvert/fermé)	Valeur de U_T (V)
Est pleine	Non	bloqué	Ouvert	5
A une ouverture	Oui	Saturé	fermé	0

Q.8- Le nombre d'impulsions N_R . **1pt**

$$N_R = 15 \text{ impulsions}$$

Q.9- Le nom de la caractéristique du codeur représenté par le nombre N_R . **1pt**

Résolution

Q.10- La période de rotation du moteur T_m (en ms) . **2pts**

$$n=360/60 \text{ tr/s donc } T_m= 60/360 =166,7 \text{ ms}$$

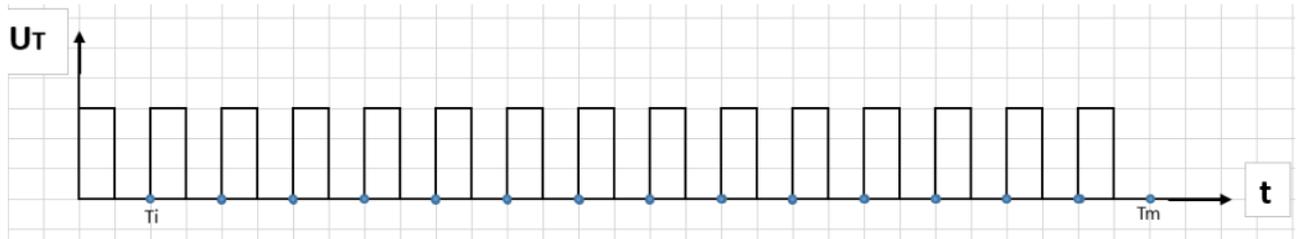
Q.11- La période T_i (en ms) et la fréquence f_i (en Hz) des impulsions. **2pts (1pt x2)**

$$T_i = T_m/15 = 166,7 /15 =11,11 \text{ ms} \quad ; \quad f_i = 1/ T_i = 90 \text{ Hz}$$

Q.12- La période T (en ms) du signal U_T . **1pt**

$$T = T_i = 11,11 \text{ ms}$$

Q.13- L'oscillogramme du signal U_T avec précision de T_i et T_m **2pts (1pt x2)**



Q.14- La nature (*digitale* ou *analogique*) des entrées RA0, RA1, RA2 et RB0 : **1pt (0,25pt x4)**

RA0 : *analogique*

RA1 : *analogique*

RA2 : *digitale*

RB0 : *digitale*

Q.15- Les mots en binaire et en hexadécimal du registres TRISA ,TRISB: **2pts (0,5pt x4)**

Bit	Mot en binaire								Mot en hexadécimal
	7	6	5	4	3	2	1	0	
TRISA	1	1	0	1	1	1	1	1	0XDF
TRISB	0	0	0	0	0	0	0	1	0X01

Q.16- Le programme Assembleur relatif à l'organigramme de fonctionnement du gyropode :

4pts (0,5pt x8)

Label	Instruction	Commentaire
	BCF STATUS, RP1	
	BSF STATUS, RP0	; Activer la page 1 (Bank 1)
	MOVLW 0XDF	
	MOVWF TRISA	; Configuration PORTA
	MOVLW 0X01	
	MOVWF TRISB	; Configuration PORTB
; lignes de Configuration des registres: ADCON1 ,OPTION et INTCON		
	BCF STATUS, RP0	; Activer la page 0

; lignes de Configuration du registre : ADCON0

LAB3	BCF	PORTA, RA5			
	BCF	PORTB, RB7		; Initialisation des sorties RB7 et RA5 du système	
	CALL	Acquisition_Vbatt		; Appel du sous-programme <i>Acquisition_Vbatt</i>	
	CALL	Affichage_LCD		; Appel du sous-programme <i>Affichage_LCD</i>	
	MOVF	Adr_Nbat, W		; $W \leftarrow N_batt$	
	SUBLW	D'N'		; $W \leftarrow N - N_batt$	tester si $N > N_batt$
	BTFSS	STATUS, C		; tester si le bit C = 1	
	GOTO	LAB1		; Si oui sauter à LAB1	
	CALL	Aquisition_angle_α		; Appel du sous-programme <i>Aquisition_angle_α</i>	
	MOVF	Adr_Nα, W		; $W \leftarrow N_α$	
	SUBLW	D'45'		; $W \leftarrow 45 - N_α$	tester si $N_α < 45$
	BTFSC	STATUS, C		; tester si le bit C = 0	
	GOTO	LAB2		; Si oui sauter à LAB2	
	MOVF	Adr_Nα, W		; $W \leftarrow N_α$	
	ADDLW	D'45'		; $W \leftarrow 45 + N_α$	tester si $-45 < N_α$
BTFSC	STATUS, C		; tester si le bit C = 0		
GOTO	LAB2		; Si oui sauter à LAB3		
CALL	Commande_MLI		; Appel du sous-programme <i>Commande_MLI</i>		
LAB1	CALL	SP_buzzer	; Appel du sous-programme <i>SP_buzzer</i>		
LAB2	CALL	Commande_Arrêt	; Appel du sous-programme <i>Commande_Arrêt</i>		
	END				

Q.17- Valeur de la tension U_{bat_d} : **1pt**

$$U_{bat_d} = 12,7 \times 0,9 = 11,43 \text{ v}$$

Q.18- Valeur de la tension U_{C1_d} : **1pt**

$$U_{C1_d} = \frac{11,43}{6} = 1,905 \text{ v}$$

Q.19- Valeurs décimales du nombre N_batt :

1,5pt (0,5pt x3)

U_{bat} (V)	11,43	12,7
U_{C1} (V)	1,905	2,116
N_batt en décimal	97	108

Q.20- Le sous-programme Acquisition_Vbatt : **3pts (0,5pt x6)**

<i>Label</i>	<i>instruction</i>	<i>Commentaire</i>
Acquisition_Vbatt Wait	BCF ADCON0, CHS2	
	BCF ADCON0, CHS1	
	BCF ADCON0, CHS0	; sélection du canal 0
	BSF ADCON0, GO	; déclencher la conversion
	BTFSC ADCON0, GO	
	GOTO Wait	; Attendre fin de conversion
	MOVF ADRESH,w	; lecture du résultat
	MOVWF Adr_Nbat	; Adr_Nbat ← W
	RETURN	; retour

Q.21- La durée T_c d'un cycle machine. **1pt**

$$T_c = \frac{1}{f_c} = \frac{4}{f_{osc}} = 1\mu s$$

Q.22- La valeur du pré-diviseur : **1pt**

$$N_{\text{pré-div}} = \frac{65536}{256} = 256$$

Q.23- La valeur $X1$: **2pts**

$$X1 = \frac{1s}{65536 \mu s} = \frac{1000000}{65536} = 15,25 \approx 15$$

Q.24- Le programme Assembleur du programme d'interruption : **4pts (0,5pt x8)**

<i>Label</i>	<i>instruction</i>	<i>Commentaire</i>
	ORG 0x004	; adresse d'interruption
;-----sauvegarder les registres-----		
	MOVWF SAVE_W	; sauver registre W
	SWAPF STATUS , W	; swap status avec résultat dans w
	MOVWF SAVE_STATUS	; sauver status swappé
;-----Traitement de l'interruption de RB0/INT (codeur)---		
	BTFSS INTCON, INTF	; Test si interruption RB0/INT ?
	GOTO Test_INT_TMR0	; si non , aller Test_INT_TMR0
	BCF INTCON, INTF	; Effacer flag d'interruption INTF

	INCF	Count_A, F	;incrémenter Count_A
; -----Traitement de l'interruption de TMR0 (Vitesse Moteur) ---			
Test_INT_TMR0	BTFSS	INTCON, TOIF	; Test si interruption Timer 0 ?
	GOTO	RestoreStatus	;si non, fin interruption
	BCF	INTCON, TOIF	; Effacer flag interruption TOIF
; Mesure de la vitesse			
	DECFSZ	Count_1s	; Test d'écoulement d'une seconde ?
	GOTO	RestoreStatus	;si non, fin interruption
	MOVF	Count_A, W	; W ← Count_A
	CALL	Calcul_Vitesse_Gyropode	; Appel Calcul_Vitesse_Gyropode
	MOVWF	Vitesse_Gyropode	; Vitesse_Gyropode ← W
;-----Restaurer les registres-----			
RestoreStatus	SWAPF	SAVE_STATUS, W	; swap ancien status, résultat dans W
	MOVWF	STATUS	; restaurer status
	SWAPF	SAVE_W, F	; sans modifier Z
	SWAPF	SAVE_W, W	; W restauré
	RETFIE		; retour d'interruption

Q.25- Les registres à sauvegarder Lors d'une routine d'interruption :(cocher les bonnes réponses)

1pt (0,5pt x2)

- | | | | |
|-------------------------------------|------------|-------------------------------------|--------|
| <input checked="" type="checkbox"/> | Registre W | <input type="checkbox"/> | ADCON0 |
| <input type="checkbox"/> | PORTB | <input checked="" type="checkbox"/> | STATUS |
| <input type="checkbox"/> | TRISC | <input type="checkbox"/> | OPTION |

Q.26- Compléter les lignes de définition du PIC16F877 en MikroC :

1,5pt (0,5pt x3)

```
List p=16F877 //définir la référence du PIC
include<p16F877.inc> //Fichier de définition des registres de PIC de la famille 16F877
include<delay.h> //Fichier de fonctions prédéfinies pour temporisations logiciels
Config H'1FF9' // Directives d'assemblage
```

Q.27- Compléter les lignes de déclaration des variables externes :

1,5pt (0,5pt x3)

```
unsigned int Vitesse_Gyropode ; // déclaration d'un entier non signé nommé Vitesse_Gyropode.
unsigned int adr_Nbat, adr_Na; // déclaration de deux entiers non signés nommés adr_Nbat et adr_Na.
float N=11,43 ; // déclaration d'une variable réelle N égale 11,43.
int Count_A, Count_1s ; // déclaration de deux entiers nommés Count_A et Count_1s.
Char txt[16] ; // déclaration d'une chaîne de 16 caractères nomme txt.
```

Q.28- Compléter les lignes de configuration de la fonction **Acquisition_Vbatt** en MiKroC.

2pts (0,5pt x4)

```
void Acquisition_Vbatt (void)
{
  ADCON0.F3 =0 ;
  ADCON0.F4 =0 ;
  ADCON0.F5 =0 ;           // sélection du canal0
  ADCON0.F2 =1 ;           //démarrer la conversion
  delay_us(10);           //attente de 10 µs
  while(ADCON0.F2 ==0);   //attendre fin conversion
  Vitesse_Gyropode = ADRESH; // Vitesse_Gyropode ← ADRESH
}
```

Q.29- Les lignes de configuration de la fonction **Affichage_LCD** en MiKroC.

3,5pts (0,25pt x14)

```
void Affichage_LCD (void)
{
  // connexion de l'afficheur LCD 2x32
  sbit LCD_RS at RB5_bit;
  sbit LCD_EN at RB6_bit;
  sbit LCD_D4 at RB1_bit;
  sbit LCD_D5 at RB2_bit;
  sbit LCD_D6 at RB3_bit;
  sbit LCD_D7 at RB4_bit;
  sbit LCD_RS_Direction at TRISB5_bit;
  sbit LCD_EN_Direction at TRISB6_bit;
  sbit LCD_D4_Direction at TRISB1_bit;
  sbit LCD_D5_Direction at TRISB2_bit;
  sbit LCD_D6_Direction at TRISB3_bit;
  sbit LCD_D7_Direction at TRISB4_bit;

  // gestion de l'affichage sur le LCD 2x32
  Lcd_Init ( );           // Initialisation d'afficheur LCD
  Lcd_Cmd ( _LCD_CLEAR); // effacer l'écran LCD
  Lcd_Cmd ( _LCD_CURSOR_OFF); // curseur OFF
  Lcd_Out (1,1,"Niveau batteries");
  IntToStr(Adr_Nbat,txt) ;
  LCD_Out(1,24,txt) ;
  Lcd_Chr (1,28,"%");
  Lcd_Out (2,1,"vitesse"); // Écrire le texte "vitesse" à la ligne 1, colonne 2
  IntToStr (Vitesse_Gyropode,txt) ; //Convertir Vitesse_Gyropode en une chaîne de caractère txt
  LCD_Out(2,24,txt) ;      // afficher la chaîne de caractère txt à la ligne 24, colonne 2
  Lcd_Out (2,28,"km/h");
}
```

Q.30- Les lignes de la fonction principale correspondant à l'organigramme :

5pts (0,5pt x10)

Void **main** (void)

```

{
    TRISA=0XDF;           // configuration du Port A
    TRISB=0X01;          // configuration du Port B
    PORTA.F5 =0;         //mise à 0 du bit RA5
    PORTB.F7 =0;         //mise à 0 du bit RB7
    Acquisition_Vbatt (); //appel de la fonction Acquisition_Vbatt
    Affichage_LCD ();   //appel de la fonction Affichage_LCD

    if (N_batt < N)
    {
        SP_buzzer();    //appel de la fonction SP_buzzer
        Commande_Arret(); //appel de la fonction Commande_Arret
        return;
    }
    Else
    { acquisition_angle_α(); //appel de la fonction acquisition_angle_α
      if ( alpha <= 45)
        { if (-45 <= alpha)
          { commande_MLI(); //appel de la fonction commande_MLI
            return;
          }
          else
          { Commande_Arret();
            return;
          }
          else
          { Commande_Arret();
            return;
          }
        }
    }
}

```

Q.31- Le rôle du transistor Q : **0,5 pt**

Élément de contrôle – transistor ballast -élément terminal

Q.32- Le rôle de la diode D : **0,5 pt**

Tension de référence

Q.33- Le rôle du diviseur de tension obtenu par R₂ et R₃ : **0,5 pt**

Echantillon de la tension de sortie

Q.34- Le rôle du circuit IC1 : **0,5 pt**

Comparateur

Q.35- La tension de l'entrée non inverseuse V⁺ : **1 pt**

V⁺ = V_Z = 2,5 v

Q.36- La tension V⁻ en fonction de R₂, R₃ et U₀ : **1 pt**

En applique le diviseur de tension $V^- = U_0 \cdot \frac{R_3}{R_3 + R_2}$

Q.37- Expression de R_2 : **1 pt**

$$\begin{cases} V^+ = V_Z \\ V^- = U_0 \cdot \frac{R_3}{R_3 + R_2} \end{cases} \text{ OR } V^+ = V^- \implies R_2 = R_3 \cdot \left(\frac{U_0}{V_Z} - 1 \right)$$

Q.38- La valeur de R_2 : **1 pt**

$$R_2 = 6,8 \text{ K}\Omega$$

Q.39- Le courant I_Z : **1 pt**

$$P_Z = V_Z \cdot I_Z \text{ donc } I_Z = \frac{P_Z}{V_Z} = 100 \text{ mA}$$

Q.40- La valeur de R_1 . **1 pt**

$$R_1 = \frac{U_{bat} - V_Z}{I_Z} = 102 \Omega$$

Cocher la bonne réponse :

Q.41- La microélectronique s'intéresse à : **1 pt**

- La fabrication des composants électroniques à base des produits chimiques ;
- La fabrication de composants électroniques à l'échelle micrométrique à partir de matériaux semi-conducteurs ;
- La fabrication, la manipulation, la caractérisation, l'imagerie, ainsi qu'aux outils de modélisation-simulation utilisant une échelle comprise typiquement entre 1 et 100 nanomètres.

Q.42- Comment appelle-t-on la salle de fabrication des puces dans les laboratoires **1 pt**

- Salle verte
- Salle blanche
- Salle rose

Q.43- La technologie qui permet l'intégration de nombreuses fonctions électroniques sur un même morceau de silicium (ou autre semi-conducteur) est :

- L'encapsulation **1 pt**
- L'instrumentation
- La photolithographie

Q.44- Pourquoi on procède au nettoyage fréquent des plaquettes de silicium ? **1 pt**

- Pour forcer le dopage des semi-conducteurs.
- Pour éliminer les impuretés déposées.
- Pour réaliser des connexions conductrices entre zones.
- Pour arracher un atome de la surface par collision élastique avec un ion incident.

Q.45- Mettre une croix dans la case convenable : **1 pt(0,5pt x2)**

L'intégration de la nanotechnologie dans les technologies d'affichage présente :	Inconvénient	Avantage
Une consommation énergétique plus faible que les écrans LCD.		X
Une mise en boîtier onéreuse requise, car les molécules sont sensibles à l'humidité.	X	