

Devoir Surveillé (DS)

Semestre : 2

d'Ingénierie et de Technologies	Session : Principale	
Module: Architecture des microcontrôle Enseignants: CHERIF Nozha, GHORE SOUAKI Ghofrane, TEBER Feten. Classes: 2A, 2P, 3B		HAOUEL Jihène, JEDIDI Hassen, LABBENE Ali,
Documents autorisés : NON	Nombre de pages : 6 page	es : ENONCE (4 pages) + ANNEXE (2 pages)
Date : 05/03/2016 Heu	ıre: 9h00	Durée: 1h00
ETUDIANT(e) N° Carte : Nom et Prénom :		Classe :
QCM (6 po	oints) Cocher la ou les bonn	re(s) réponse(s) :
1. Un microcontrôleur est u	ın •	
	· les systèmes embarqués.	
<u> </u>	· · · · · · · · · · · · · · · · · · ·	eur et les mémoires d'une machine à distance.
		contenant un microprocesseur capable d'exécuter
		des programmes et données, et des ports pour
communiquer avec le		
	ontient un microprocesseur, m e la carte mère d'un ordinateur	némoires et des ports de gestion des périphériques r.
une instruction est code ☐ (a) asser ☐ (a) binai ☐ (a) asser	6F84 contiennent des inform Se sur (b), et une donnée est combleur (b) 14 bits (c) 12bits. re (b) 12 bits (c) 8bits. mbleur (b) 14 bits (c) 8bits. re (b) 14 bits (c) 8bits.	nations (instructions et données) sous forme (a), rodée sur (c) :
		nnées RAM du PIC16F84, sont des cases
	une à une fonction bien déter	·
-		es sont connectés des périphériques d'entrées et TRISB
pour les périphériq		The Page 1944 (PRO PRO 1974)
, ,	_	ement des pattes RA0 => RA4 et RB0 => RB7 qui
	soit des périphériques d'entrée	
• •	s pour la gestion des états des parties RA0 => RA4 et RB0 =	périphériques d'entrées ou de sorties connectés > RB7.
☐ PORTB pour la ge	stion des états des périphérique	es de sortie uniquement.
4. La fréquence de l'ho	orloge externe utilisée est 10 N	MHz, les instructions GOTO et CALL
-	en (a) cycle-s- machine qui v	
	(b) 0,8 microsecondes.	(-) -
	(b) 0,8 microsecondes.	
	(- , - , 	

□ (a) 2 et (b) 8 microsecondes.□ (a) 2 et (b) 800 nanosecondes.

5.	Si je souhaite réaliser un programme contenant au moins 1050 instructions, est ce que le
	PIC16F84 me conviendrait ?
	☐ Oui non
6.	Le registre spécial STATUS est composé de bits spéciaux, parmi lesquels :
	▼ Un bit permettant le changement entre les banques.
	Un bit qui indique si le résultat de la dernière opération exécutée par le microprocesseu
	est nul ou non-nul.
	☐ Un bit qui indique l'état des périphériques d'entrée/sortie connectés au PIC16F84.
	☐ Un bit qui indique si le résultat de la dernière opération exécutée par le microprocesseu

Exercice (4 points)

est positif ou négatif.

Soient une variable i et une sous-routine tempo permettant de faire une attente logicielle de 0,5s. Soit un afficheur 7 segments de type BCD connecté au PIC16F84 qui n'affiche que des chiffres en fonction de la valeur du registre PORTB : (voir tableau ci-dessous)

Valeur du registre PORTB en binaire	Le chiffre affiché au niveau de l'afficheur 7_seg_BCD
00000000	0
0000001	1
0000010	2
00000011	3
00000100	4
00000101	5
00000110	6
00000111	7
00001000	8
00001001	9

Travail demandé:

Commenter les instructions de la sous_routine ci-dessous **et déduire** l'objectif de cette sous_routine (que voit-on sur l'afficheur ?)

sous_routine
movlw B'00000010';W <2
clrf PORTB; Portb < 0 <==> Afficher 0
call tempo; attente 0,5s
boucle ; étiquette nommée « boucle »
addwf PORTB,f; Portb < Portb + 2 <==> Afficher 2.4,6,8 3.5,7,9
call tempo ;attente 0,5s
Btfss PORTB,3; Tester le bit 3 <==> atteint la valeur 8 où non
goto boucle ; retour à l'étiquette « boucle » au cas où
BTFSC PORTB,0; Tester le bit 0 <==> atteint la valeur 9 où non
return; quitter la sous_routine au cas où
clrf PORTB ; Portb < 0
Incf PORTB,f; Portb < Portb + 1
Call tempo ;attente 0,5s
Goto boucle ; retour à l'étiquette « boucle » au cas où

L'objectif de cette sous routine (que voit-on sur l'afficheur ?)

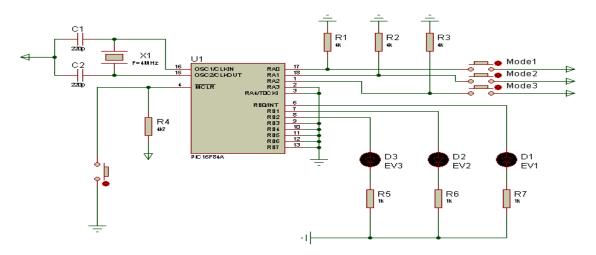
Le but de cette sous-routine			
	* Le cycle impair 1,3,	,5,7,9	

Problème: (10 points)

On se propose de réaliser un système d'arrosage automatique dépendant des conditions climatiques, qui, en fonction du mode choisi (mode1, mode2 ou mode 3), on active la ou les électrovannes correspondantes au mode sélectionné.

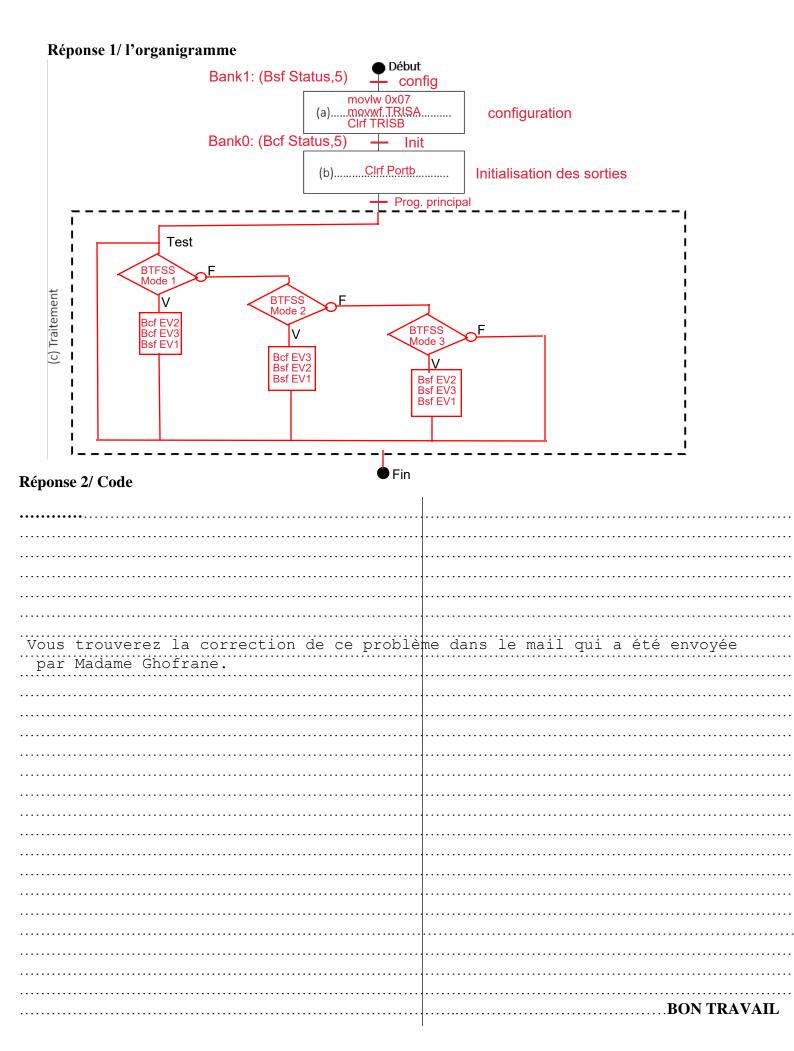
Mode	Electrovanne_1	Electrovanne_2	Electrovanne_3
Mode_1	activée	désactivée	désactivée
Mode_2	activée	activée	désactivée
Mode_3	activée	activée	activée

Comme il est indiqué dans le schéma ci-dessous, les modes sont représentés par des boutons (mode1, mode2 ou mode 3) et les électrovannes par des diodes (EV1, EV2 et EV3).



Travail demandé:

- 1/ Compléter l'organigramme ci-dessous décrivant le fonctionnement souhaité de notre système (Détailler la partie (a),(b) et (c) :Traitement.) (4 points)
- 2/ Ecrire le code assembleur complet qui décrit le fonctionnement souhaité. (6 points : (a)1point, (b)1point, (c) 4points)



ANNEXE(1/2)

Registres spéciaux (SFR : Special Function Registers)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on Power-on Reset	Value on all other resets (Note 3)
Bank 0											
00h	INDF	Uses co	ntents of F	SR to addre	ess data memory	y (not a phys	sical registe	r)			
01h	TMR0	8-bit rea	l-time dock	/counter						XXXX XXXX	uuu uuuu
02h	PCL	Low ord	er 8 bits of	the Program	n Counter (PC)					0000 0000	0000 0000
03h	STATUS (2)	IRP	RP1	RP0	TO	PD	Z	DC	С	0001 1xxx	000q quuu
04h	FSR	Indirect	data memo	ry address	pointer 0	ointer 0				XXXX XXXX	uuu uuuu
05h	PORTA	-	_	_	RA4/T0CKI	RA3	RA2	RA1	RA0	X XXX	
06h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0/INT	XXXI XXXI	unn unn
07h		Unimple	mented loc	ation, read	as'0'						
08h	EEDATA	EEPRO	M dala regi	ster					XXXX XXXX	unn unn	
09h	EEADR	EEPRO	M address i	register							uuu uuuu
0Ah	PCLATH	_	_	_	Write buffer for	upper 5 bit	s of the PC	(1)		0 0000	0 0000
0Bh	INTCON	GIE	EEIE	TOLE	INTE	RBIE	TOLE	INTF	RBIF	0000 000x	0000 000u
Bank 1											
80h	INDF	Uses co	ntents of F	SR to addre	ess data memory	y (not a phys	sical registe	r)			
81h	OPTION_ REG	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
82h	PCL	Low ord	er 8 bits of	Program O	ounter (PC)					0000 0000	0000 0000
83h	STATUS (2)	IRP	RP1	RP0	TO	PD	Z	DC	С	0001 1xxx	000q quuu
84h	FSR	Indirect	data memo	ry address	pointer 0					XXXX XXXX	uuu uuuu
85h	TRISA	_	_	_	PORTA data d	rection regi:	ster			1 1111	1 1111
86h	TRISB	PORTB	data directi	on register						1111 1111	1111 1111
87h		Unimple	mented loc	ation, read	as'0'						
88h	EECON1	-	_	_	EEF	WRERR	WREN	WR	RD	0 x000	0 good
89h	EECON2	EEPRO	M control re	gister 2 (n	ota physical reg	ister)					
0Ah	PCLATH	_	_	_	Write buffer for	iffile buffer for upper 6 bits of the PC ⁽¹⁾					0 0000
0Bh	INTCON	GIE	EEIE	TOLE	INTE	RBIE	TOLE	INTE	RBIF	0000 000x	0000 00Du

Legend: x = unknown, u = unchanged. - = unimplemented read as '0', q = value depends on condition.

- 2: The TO and PD status bits in the STATUS register are not affected by a MCLR reset.
- Other (non power-up) resets include: external reset through MCLR and the Watchdog Timer Reset.

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a slave register for PC<12:8>. The contents of PCLATH can be transferred to the upper byte of the program counter, but the contents of PC<12:8> is never transferred to PCLATH.

ANNEXE(2/2)

Jeu d'instructions

Mnemonic,		Description	Cycles		14-Bit Opcode				Notes
Operar	nds			MSb			LSb	Affected	
		BYTE-ORIENTED FILE REGIS	STER OPE	RATIC	NS				
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	Ζ	1,2
CLRF	f	Clear f	1	00	0001	lfff	ffff	Ζ	2
CLRW	-	Clear W	1	00	0001	0xxx	xxxx	Ζ	
COMF	f, d	Complement f	1	00	1001	dfff	ffff	Ζ	1,2
DECF	f, d	Decrement f	1	00	0011	dfff	ffff	Ζ	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	Ζ	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	Ζ	1,2
MOVF	f, d	Move f	1	00	1000	dfff	ffff	Ζ	1,2
MOVWF	f	Move W to f	1	00	0000	lfff	ffff		
NOP	-	No Operation	1	00	0000	0xx0	0000		
RLF	f, d	Rotate Left f through Carry	1	00	1101	dfff	ffff	С	1,2
RRF	f, d	Rotate Right f through Carry	1	00	1100	dfff	ffff	С	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

		BIT-ORIENTED FILE REGIST	ER OPER	RATION	IS			
BCF	f, b	Bit Clear f	1	01	00bb	bfff	ffff	1,2
BSF	f, b	Bit Set f	1	01	01bb	bfff	ffff	1,2
BTFSC	f, b	Bit Test f, Skip if Clear	1 (2)	01	10bb	bfff	ffff	3
BTFSS	f, b	Bit Test f, Skip if Set	1 (2)	01	11bb	bfff	ffff	3

		LITERAL AND CONTROL	OPERAT	IONS					
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	Ζ	
CALL	k	Call subroutine	2	10	0kkk	kkkk	kkkk		
CLRWDT		Clear Watchdog Timer	1	00	0000	0110	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	Ζ	
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 in 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	