

```

; HORLOGE COMTOISE Version 2.0 (Mai 2020) *

LIST P=PIC16F631 ; Définition de processeur
#include <pl6F631.inc> ; fichier include
ERRORLEVEL -302 ; suppression du message bank select
ERRORLEVEL -305 ; suppression du message absence de destination

__CONFIG _FCMEN_OFF & _IESO_OFF & _BOD_OFF & _CPD_OFF & _CP_OFF & _MCLRE_OFF & _PWRTE_OFF &
_WDT_OFF & _INTOSCIO

;*****
; ASSIGNATIONS SYSTEME *
;*****

OPTIONVAL EQU B'00000000' ; Configuration registre OPTION
INTCONVAL EQU B'11000000' ; GIE, PEIE, RABIF validés
OSCCONVAL EQU B'01100001' ; Valeur "110": 4 Mhz
ANSELVAL EQU B'00000000' ; Digital I/O
PIE1VAL EQU B'00000001' ; Timer1 en fonctionnement
T1CONVAL EQU B'10001111' ; Configuration TIMER1
IOCAVAL EQU B'00000000' ; Pas d'interruption en PORTA
IOCBVAL EQU B'00000000' ; Interruption sur RB5
WPUAVAL EQU B'00000100' ; Résistance de Pull-Up sur RA2
WPUBVAL EQU B'01000000' ; Résistance de Pull-Up sur RB6

DIRPORTA EQU B'00110100' ; Direction PORTA (1=entrée)(0=sortie)
DIRPORTB EQU B'01110000' ; Direction PORTB (1=entrée)(0=sortie)
DIRPORTC EQU B'00000110' ; Direction PORTC (1=entrée)(0=sortie)

;*****
; ASSIGNATIONS PROGRAMME *
;*****

LCD_FUNCTION_SET EQU B'00101000' ; Mode 4bits, 2 lignes
LCD_CURSOR_DISPLAY_SHIFT EQU B'00011100' ;
LCD_DISPLAY_CONTROL EQU B'00001100' ;
LCD_ENTRY_MODE_SET EQU B'00000110' ;

;*****
; DEFINE *
;*****

#DEFINE CARRY STATUS,C ; Débordement
#DEFINE ZERO STATUS,Z ; Valeur ZERO

#DEFINE VALID PORTA,2 ; Bouton poussoir "VALID"
#DEFINE B_PLUS PORTB,6 ; Bouton poussoir "+" (Mise à l'heure)
#DEFINE B_MOINS PORTC,2 ; Bouton poussoir "-" (réglage calibration)
#DEFINE ELECTRO PORTC,0 ; Electro-aimant

#DEFINE CELG PORTB,4 ; Cellule de gauche
#DEFINE CELS PORTB,5 ; Cellule sonnerie (grande aiguille à 12h)

#DEFINE LCD_RS PORTC,5 ; RS Sélection registre de l'afficheur
#DEFINE LCD_E PORTC,4 ; E Validation
#DEFINE LCD_D4 PORTB,7 ; D4
#DEFINE LCD_D5 PORTC,3 ; D5
#DEFINE LCD_D6 PORTC,7 ; D6
#DEFINE LCD_D7 PORTC,6 ; D7

#DEFINE SNEGx REG_MATH,7 ; Calculs mathématiques
#DEFINE SNEGy REG_MATH,6
#DEFINE SNEGs REG_MATH,5
#DEFINE SNEGAs REG_MATH,4
#DEFINE SNEGaz REG_MATH,3
#DEFINE INIFLAG REG_MATH,2

#DEFINE SNEG REG_LCD,7

;*****
; MACRO *
;*****

#include <MACRO.asm>

;*****
; DECLARATION DES VARIABLES *
;*****

```



```

movlw  WPUAVAL
movwf  WPUA
movlw  IOCAVAL
movwf  IOCA
movlw  OPTIONVAL
movwf  OPTION_REG      ; Initialisation registre OPTION
movlw  OSCCONVAL
movwf  OSCCON          ; Initialisation registre OSCCON
movlw  PIE1VAL
movwf  PIE1            ;

BANK2          ; Sélection bank2
clrf  ANSEL          ; Initialisation registre ANSEL
movlw  WPUBVAL
movwf  WPUB
movlw  IOCBVAL
movwf  IOCB

; Effacer RAM bank 0
; -----
BANK0
movlw  0x40          ; sélectionner bank 0
movwf  FSR           ; initialisation pointeur
init1  movwf  FSR      ; d'adressage indirect

clrf  INDF           ; effacer ram
incf  FSR,f          ; pointer sur suivant
btfss FSR,7          ; tester si fin zone atteinte (>7F)
goto  init1          ; non, boucler

; Initialisation programme
; -----

call  LCD_INIT      ; Initialisation de l'afficheur LCD

;*****
;
;          PROGRAMME PRINCIPAL
;*****
call  Tempo_200ms
call  LEC_EEPROM    ; Permet la calibration de l'horloge
call  REGLAGEH      ; Réglage de l'heure

start  call  AFF_LIGNE1
call  AFF_LIGNE2
btfsc B_PLUS
goto  start5
call  REGLAGEH
goto  start1

start5 btfsc B_MOINS
goto  start1
call  REGLAGEC      ; Réglage de la calibration

start1 call  Tempo_10ms
movf  MINUTE,w
sublw d'59'          ; 59 moins MINUTE
btfss ZERO          ; "MINUTE" égal à 59 ?
goto  start3
btfsc CELS          ;
goto  start
btfsc flag,3
goto  start2
movf  SECONDE,w
sublw d'60'
movwf AVANCE        ; Résultat soustraction dans "AVANCE"
clrf  RETARD        ; Effacement "RETARD"
bsf  ELECTRO        ; Blocage du balancier
bsf  flag,3

start2 movf  SECONDE,w
andlw 0xFF
btfss ZERO          ; Registre "SECONDE" = 0 ?
goto  start         ; Bouclage en attente SECONDE = 0

start3 bcf  ELECTRO      ; Déblocage balancier si bloqué
movf  MINUTE,w
andlw 0xFF          ; Fonction "ET"
btfss ZERO          ; "MINUTE" égal à Zéro
goto  start4
btfsc CELS

```

```

        goto    start
        btfsc   flag,3
        goto    start
        btfsc   flag,4
        goto    start
        movf    SECONDE,w
        movwf   RETARD                ; SECONDE transféré dans "RETARD"
        clrf    AVANCE                ; Effacement "AVANCE"
        bsf     flag,4
        goto    start
start4
        bcf     flag,3
        bcf     flag,4
        goto    start

;
;                               FIN DU PROGRAMME PRINCIPAL
;
;*****
;                               SOUS PROGRAMMES
;*****

#include <q4mhz.asm>

RTC:
        incf    SECONDE,f
        movf    SECONDE,w
        sublw   d'60'                ; 60 moins w
        btfss   ZERO                 ; Test si "SECONDE" = 0
        goto    rtc2                ; Vers calibration TMR1H chaque seconde
        clrf    SECONDE
        incf    MINUTE,F
        movf    MINUTE,w
        sublw   d'60'                ; 60 moins w
        btfss   ZERO                 ; Test si "MINUTE" = 0
        goto    rtc1
        clrf    MINUTE
        incf    HEURE,f
        movf    HEURE,w
        sublw   d'24'                ; 24 moins w
        btfss   ZERO                 ; Test si "HEURE" = 0
        goto    rtc1
        clrf    HEURE

rtc1
        movlw   0x7F                 ; Calibration (chaque minute)
        movwf   TMR1H                ; Calibration
        comf    CALIBRE,w            ; Complément registre "CALIBRE"
        movwf   TMR1L                ; Calibration chaque minute (65536 - 32768 - "CALIBRE")
        return

rtc2
        bsf     TMR1H,7              ; Paramètre pour comptage des secondes. Valeur: 32768
                                        ; dans le registre "TMR1" pour compter jusqu'à 65536
        return

BLOCAGE:
        movwf   AVANCE
        movwf   cmpt0
        bsf     ELECTRO              ; Cellule gauche atteinte: Blocage du balancier
blocage1
        call    Tempo_1s             ; Tempo 1 seconde
        call    AFF_LIGNE1           ;
        call    AFF_LIGNE2           ;
        decfsz  cmpt0,f              ; Décrémenter registre "cmpt0" (égal à "ECART")
        goto    blocage1             ; Non (Bouclage)
        bcf     ELECTRO              ; Temps écoulé: Oui ... Déblocage du balancier
        return

REGLAGEH:
        bcf     flag,3
        bcf     flag,4
        call    LCD_INIT
        bcf     INTCON,7             ; Arrêt de toutes les interruptions
        clrf    AVANCE
        clrf    RETARD
        clrf    SECONDE
        call    Tempo_1s

Heu1

```

```

        call    LCD_EFFACE
        call    AFF_LIGNE1
        call    Tempo_200ms
        call    Tempo_100ms
Heu2
        btfsc   B_PLUS
        goto    Heu3
        movf    cmpt0,0
        sublw   d'22'
        btfss   CARRY
        goto    Heu21
        incf    cmpt0
        movf    cmpt0,0
        movwf   HEURE
        goto    Heu1
Heu21
        movlw   d'0'
        movwf   cmpt0
        movwf   HEURE
        goto    Heu1
Heu3
        btfsc   B_MOINS
        goto    Heu4
        movf    cmpt0,0
        sublw   d'0'
        btfsc   CARRY
        goto    Heu31
        decf    cmpt0
        movf    cmpt0,0
        movwf   HEURE
        goto    Heu1
Heu31
        movlw   d'23'
        movwf   cmpt0
        movwf   HEURE
        goto    Heu1
Heu4
        btfsc   VALID
        goto    Heu1
        movf    cmpt0,0
        movwf   HEURE
        call    Tempo_1s
Min1
        call    AFF_LIGNE1
        call    Tempo_200ms
Min2
        btfsc   B_PLUS
        goto    Min3
        movf    cmpt0,0
        sublw   d'58'
        btfss   CARRY
        goto    Min21
        incf    cmpt0
        movf    cmpt0,0
        movwf   MINUTE
        goto    Min1
Min21
        movlw   d'0'
        movwf   cmpt0
        movwf   MINUTE
        goto    Min1
Min3
        btfsc   B_MOINS
        goto    Min4
        movf    cmpt0,0
        sublw   d'0'
        btfsc   CARRY
        goto    Min31
        decf    cmpt0
        movf    cmpt0,0
        movwf   MINUTE
        goto    Min1
Min31
        movlw   d'59'
        movwf   cmpt0
        movwf   MINUTE
        goto    Min1
Min4
        btfsc   VALID
        goto    Min1

```

```

        clr    TMR1H
        clr    TMR1L
        bsf    INTCON,7           ; Validation de toutes les interruptions
        return

REGLAGEC:
        call   LEC_EEPROM
        movwf  cmpt0
        call   LCD_EFFACE

Regc1
        call   AFF_LIGNEC
        call   Tempo_500ms

Regc2
        btfsc  B_PLUS
        goto   Regc3
        incf   cmpt0
        movf   cmpt0,0
        goto   Regc1

Regc3
        btfsc  B_MOINS
        goto   Regc4
        decf   cmpt0
        movf   cmpt0,0
        goto   Regc1

Regc4
        btfsc  VALID
        goto   Regc1
        goto   Regc7

Regc7
        movf   cmpt0,0
        call   ECR_EEPROM
        return

LEC_EEPROM:           ; Lecture de la calibration en EEPROM
        BANK2
        movlw  0x00           ; Adresse 0x00 de la mémoire EEPROM
        movwf  EEADR         ; 0x00 est l'adresse de l'emplacement mémoire à lire
        BANK3
        bsf    EECON1,0       ; Lance l'opération de lecture
        BANK2
        movf   EECON1,w       ; La valeur lue dans l'EEPOM à l'adresse 0x00 est mise dans "w"
        BANK0
        movwf  CALIBRE       ; La valeur est ensuite transférée dans le registre "CALIBRE"
        return

ECR_EEPROM:           ; Ecriture d'une valeur de calibration en EEPROM
        BANK2
        movwf  EECON1         ; La valeur à écrire est placée dans le registre EECON1
        movlw  0x00           ; La valeur de l'emplacement EEPROM est indiquée ici: 0x00
        movwf  EEADR         ; 0x00 est l'emplacement dans lequel la valeur sera écrite
        BANK3
        bcf    INTCON,GIE     ; Désactivation de toutes les interruptions
        bsf    EECON1,WREN     ; Autorisation de l'écriture
        btfsc  INTCON,GIE     ; Test si le bit de désactivation des interruptions est positionné
        goto   $-2            ;
        movlw  0x55           ; Séquence obligatoire
        movwf  EECON2         ; Séquence obligatoire
        movlw  0xAA           ; Séquence obligatoire
        movwf  EECON2         ; Séquence obligatoire
        bsf    EECON1,WR       ; Lance l'opération d'écriture
        bsf    INTCON,GIE     ; Autorise à nouveau toutes les interruptions
        bcf    EECON1,WREN     ; Interdit toutes les opérations d'écriture
        BANK0
        call   Tempo_200ms
        return

AFF_LIGNEC:
        movlw  0x00
        call   LCD_LOCATE
        movlw  0x43           ; Affichage signe "C" (pour calibration)
        call   LCD_DONNEE
        movlw  0x3A           ; Affichage signe ":"
        call   LCD_DONNEE
        clr    x+2
        clr    x+1
        movf   cmpt0,w

```

```

movwf x+0
movlw d'4'
call AFFICHE_24BS ; Affichage de la valeur de calibration de l'horloge
return

```

AFF_LIGNE1:

```

movlw 0x00
call LCD_LOCATE
movlw "0"
call LCD_DONNEE
clrf x+2 ;
clrf x+1 ;
movf HEURE,w ;
movwf x+0 ;
movlw d'1' ;
call AFFICHE_24BS ; Affichage "HEURES" de l'horloge à quartz
movlw 0x68 ;
call LCD_DONNEE ; Affichage signe "h"
movlw "0"
call LCD_DONNEE
movlw "0"
call LCD_DONNEE
clrf x+2 ;
clrf x+1 ;
movf MINUTE,w ;
movwf x+0 ;
movlw d'4' ;
call AFFICHE_24BS ; Affichage "MINUTES" de l'horloge à quartz"
movlw 0x3A ;
call LCD_DONNEE ; Affichage signe "deux points" ":"
movlw "0"
call LCD_DONNEE
movlw "0"
call LCD_DONNEE
clrf x+2 ;
clrf x+1 ;
movf SECONDE,w ;
movwf x+0 ;
movlw d'7' ;
call AFFICHE_24BS ; Affichage "SECONDES" de l'horloge à quartz"
return

```

AFF_LIGNE2

```

movlw d'64'
call LCD_LOCATE
movlw B'01000001'
call LCD_DONNEE ; Affichage signe "A" (pour avance)
movlw "0"
call LCD_DONNEE
movlw "0"
call LCD_DONNEE
clrf x+2 ;
clrf x+1 ;
movf AVANCE,w ;
movwf x+0 ;
movlw d'66'
call AFFICHE_24BS ; Affichage du registre "AVANCE"
movlw B'01110011'
call LCD_DONNEE ; Affichage signe "s" (pour seconde)
movlw d'68'
call LCD_LOCATE
movlw B'01010010'
call LCD_DONNEE ; Affichage signe "R" (pour retard)
movlw "0"
call LCD_DONNEE
movlw "0"
call LCD_DONNEE
clrf x+2 ;
clrf x+1 ;
movf RETARD,w ;
movwf x+0 ;
movlw d'70'
call AFFICHE_24BS ; Affichage du registre "RETARD"
movlw B'01110011'
call LCD_DONNEE ; Affichage signe "s" (pour seconde)
return

```

; AFFICHEUR LCD 2x8 ou 2x16 en mode 4 bits

```

LCD_INIT:                                ;Initialisation de l'afficheur
    call    Tempo_30ms
    movlw  h'33'                          ;Function set
    call   LCD_CONTROLE                    ;Ecriture
    movlw  h'28'                          ;Function set
    call   LCD_CONTROLE                    ;Ecriture
    movlw  LCD_FUNCTION_SET                ;Function set
    call   LCD_CONTROLE                    ;Ecriture
    movlw  LCD_DISPLAY_CONTROL            ;Display ON/OFF
    call   LCD_CONTROLE                    ;Ecriture
    call   LCD_EFFACE
    movlw  LCD_ENTRY_MODE_SET             ;Entry mode set
    call   LCD_CONTROLE                    ;Ecriture
    return

LCD_CONTROLE:                             ; Envoi d'une commande vers l'afficheur
    bcf    LCD_RS
    call   LCD_4B
    return

LCD_DONNEE:                               ; Envoi d'une donnée vers l'afficheur
    bsf    LCD_RS
    call   LCD_4B
    return

LCD_LOCATE:                               ; Positionnement de la donnée à écrire
    addlw  d'128'
    CALL   LCD_CONTROLE
    return

LCD_EFFACE:                               ; Effacement de l'afficheur
    movlw  h'01'
    call   LCD_CONTROLE
    call   Tempo_1ms
    return

LCD_4B:                                   ; Transfert d'un octet en mode 4 bits
    movwf  LCD_DATA
    bcf    LCD_D4
    bcf    LCD_D5
    bcf    LCD_D6
    bcf    LCD_D7
    btfsc  LCD_DATA,7
    bsf    LCD_D7
    btfsc  LCD_DATA,6
    bsf    LCD_D6
    btfsc  LCD_DATA,5
    bsf    LCD_D5
    btfsc  LCD_DATA,4
    bsf    LCD_D4
    bsf    LCD_E
    nop
    nop
    bcf    LCD_E
    bcf    LCD_D4
    bcf    LCD_D5
    bcf    LCD_D6
    bcf    LCD_D7
    btfsc  LCD_DATA,3
    bsf    LCD_D7
    btfsc  LCD_DATA,2
    bsf    LCD_D6
    btfsc  LCD_DATA,1
    bsf    LCD_D5
    btfsc  LCD_DATA,0
    bsf    LCD_D4
    bsf    LCD_E
    nop
    nop
    bcf    LCD_E
    call   Tempo_11us
    return

;*****

AFFICHE_8BS:                              ; Affichage d'une valeur sur 1 octet avec signe
    clrf  x+2                              ; Effacement de "x+2"
    clrf  x+1                              ; Effacement de "x+1"

```

