

```

; Programme digicode

ERRORLEVEL -302                ; suppression du message bank select
LIST      P=PIC16F88           ; Définition de processeur
#include <p16F88.inc>           ; fichier include

__CONFIG __CONFIG1, _CP_OFF & _CCP1_RB0 & _DEBUG_ON & _WRT_PROTECT_OFF & _CPD_OFF & _LVP_OFF &
_BODEN_OFF & _MCLR_ON & _PWRTE_ON & _WDT_OFF & _HS_OSC

__CONFIG __CONFIG2, _IESO_OFF & _FCMEN_OFF

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

OPTIONVAL      EQU      B'00000000'
INTCONVAL      EQU      B'00000000'
ANSELVAL       EQU      B'00000000'

DIRPORTA       EQU      B'00000000' ; Direction PORTA (1=entrée)
DIRPORTB       EQU      B'11110000' ; Direction PORTB (0=sortie)

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

#DEFINE LED          PORTA, 4      ; Commande LED
#DEFINE GACHE        PORTB, 0      ; Commande gache
#DEFINE APPTOUCHE    touche , 4    ; Touche appuyée
#DEFINE APPLONG      touche , 5    ; Touche longue
#DEFINE PROGRAMME    touche , 6    ; Touche de programmation

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

BANK0 macro                ; passer en banque0
        bcf      STATUS,RP0
        bcf      STATUS,RP1
    endm

BANK1 macro                ; passer en banque1
        bsf      STATUS,RP0
        bcf      STATUS,RP1
    endm

BANK2 macro                ; passer en banque2
        bcf      STATUS,RP0
        bsf      STATUS,RP1
    endm

BANK3 macro                ; passer en banque3
        bsf      STATUS,RP0
        bsf      STATUS,RP1
    endm

; OPERATION EN EEPROM
; -----

REEEPROM macro            ; lire eeprom (adresse dans adeeprom & résultat en w)
    clrwdt                ; reset watchdog
    bcf      STATUS,RP0    ; passer en banque2
    bsf      STATUS,RP1
    movf    adeeprom,w     ; charger adresse eeprom (passage de la valeur par une variable)
    movwf   EEADR          ; pointer sur adresse eeprom
    bsf      STATUS,RP0    ; passer en banque3
    bcf      EECON1,EEPGD   ; pointer sur eeprom
    bsf      EECON1,RD     ; ordre de lecture
    bcf      STATUS,RP0    ; passer en banque2
    movf    EEDATA,w       ; charger valeur lue
    bcf      STATUS,RP1    ; passer en banque0
    endm

WEEPROM macro            ; la donnée se trouve dans W
    LOCAL   loop
    BANK2                ; passer en banque2

```

```

movwf EEDATA ; placer data dans registre
movf addwrite,w ; charger adresse d'écriture (passage de la valeur par une variable)
movwf EEADR ; placer dans registre
bsf STATUS , RP0 ; passer en banque3
bcf EECON1 , EEPGD ; pointer sur mémoire data
bsf EECON1 , WREN ; autoriser accès écriture
bcf INTCON , GIE ; interdire interruptions
movlw 0x55 ; charger 0x55
movwf EECON2 ; envoyer commande
movlw 0xAA ; charger 0xAA
movwf EECON2 ; envoyer commande
bsf EECON1 , WR ; lancer cycle d'écriture
bsf INTCON , GIE ; réautoriser interruptions

loop
btfsc EECON1 , WR ; tester si écriture terminée
goto loop ; non, attendre
bcf EECON1 , WREN ; verrouiller prochaine écriture
bcf STATUS , RP0 ; passer en banque0
bcf STATUS , RP1
endm

```

```

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

```

```

; Zone de 16 bytes commune
; -----

```

```

CBLOCK 0x70 ; Début de la zone (0x70 à 0x7F)
adeeprom : 1
addwrite : 1
ENDC

```

```

; Zone de 80 bytes en banque 0
; -----

```

```

CBLOCK 0x20 ; Début de la zone (0x20 à 0x6F)
tempo1 : 1
tempo2 : 1
tempo3 : 1
touche : 1
touche1 : 1
touche2 : 1
touche3 : 1
touche4 : 1
toucheP : 1
toucheA : 1
comptprog : 1
touche1_P : 1
touche2_P : 1
touche3_P : 1
touche4_P : 1
ENDC ; Fin de la zone

```

```

;*****
;
;          MEMOIRE EEPROM
;*****

```

```

org 0x2100 ; Adresse de départ EEPROM
DE 0x10
DE 0x10
DE 0x10
DE 0x10
DE 0x3b
DE 0x1a

```

```

;*****
;
;          DEMARRAGE SUR RESET
;*****

```

```

org 0x000 ; Adresse de départ après reset
goto init ; Initialiser

```

```

;*****
;
;          ROUTINE INTERRUPTION
;*****

```

```

    org 0x004                ; adresse d'interruption
    return

; ////////////////////////////////////////////////////////////////////

;                               P R O G R A M M E

; ////////////////////////////////////////////////////////////////////

;*****
;                               INITIALISATIONS
;*****
init

                ; initialisation PORTS (banque 0 et 1)
                ; -----
BANK0          ; sélectionner banque0
clrf   PORTA   ; Sorties PORTA à 0
movlw  b'11111111'
movwf  PORTB   ; sorties PORTB à 1
BANK1          ; passer en banque1
movlw  DIRPORTA ; Direction PORTA
movwf  TRISA   ; écriture dans registre direction
movlw  DIRPORTB ; Direction PORTB
movwf  TRISB   ; écriture dans registre direction

                ; Registre d'options (banque 1)
                ; -----
movlw  OPTIONVAL ; charger masque
movwf  OPTION_REG ; initialiser registre option

                ; registres interruptions (banque 1)
                ; -----
movlw  INTCONVAL ; charger valeur registre interruption
movwf  INTCON    ; initialiser interruptions
movlw  ANSELVAL  ; charger masque
movwf  ANSEL     ; initialiser le registre du convertisseur

                ; Effacer RAM banque 0
                ; -----
BANK0          ; sélectionner banque 0
movlw  0x20     ; initialisation pointeur
movwf  FSR      ; d'adressage indirect

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

                ; Configuration digicode
                ; -----
bcf   GACHE
bcf   LED

clrf  adeeprom
REEPROM
movwf touche1
incf  adeeprom
REEPROM
movwf touche2
incf  adeeprom
REEPROM
movwf touche3
incf  adeeprom
REEPROM
movwf touche4
incf  adeeprom
REEPROM
movwf toucheP
incf  adeeprom
REEPROM
movwf toucheA

goto  start    ; programme principal

;*****

```

```

;                               PROGRAMME PRINCIPAL                               *
;*****
start
    clrwdt                        ; effacer watch dog

;----- TEST TOUCHE 1 OU PROG -----
    call    testclavier
    movf    touche,w
    subwf   toucheP,w
    btfsc   STATUS,Z
    goto    programmation
ici
    movf    touche,w
    subwf   touche1,w
    btfss   STATUS,Z
    goto    start

;----- TEST TOUCHE 2 OU PROG -----
    call    testclavier
    movf    touche,w
    subwf   toucheP,w
    btfsc   STATUS,Z
    goto    programmation
    movf    touche,w
    subwf   touche2,w
    btfss   STATUS,Z
    goto    ici

;----- TEST TOUCHE 3 OU PROG -----
    call    testclavier
    movf    touche,w
    subwf   toucheP,w
    btfsc   STATUS,Z
    goto    programmation
    movf    touche,w
    subwf   touche3,w
    btfss   STATUS,Z
    goto    ici

;----- TEST TOUCHE 4 OU PROG -----
    call    testclavier
    movf    touche,w
    subwf   toucheP,w
    btfsc   STATUS,Z
    goto    programmation
    movf    touche,w
    subwf   touche4,w
    btfss   STATUS,Z
    goto    ici

;----- OUVERTURE -----
    bsf     LED
    bsf     GACHE
    movlw   d'10'
    call    tempoV
    bcf     GACHE
    bcf     LED
    goto    start

;-----
;----- PROGRAMMATION -----
;-----
programmation
    movlw   d'3'
    call    tempoV
    bsf     LED
    movlw   d'3'
    call    tempoV
    bcf     LED
    movlw   d'3'
    call    tempoV
    bsf     LED
    movlw   d'3'
    call    tempoV
    bcf     LED

;----- TEST TOUCHE 1 OU ANNUL -----

```

```

    call    testclavier
    movf   touche,w
    subwf  toucheA,w
    btfsc  STATUS,Z
    goto   annulation
    movf   touche,w
    subwf  touche1,w
    btfss  STATUS,Z
    goto   erreur

;----- TEST TOUCHE 2 OU ANNUL -----
    call    testclavier
    movf   touche,w
    subwf  toucheA,w
    btfsc  STATUS,Z
    goto   annulation
    movf   touche,w
    subwf  touche2,w
    btfss  STATUS,Z
    goto   erreur

;----- TEST TOUCHE 3 OU ANNUL -----
    call    testclavier
    movf   touche,w
    subwf  toucheA,w
    btfsc  STATUS,Z
    goto   annulation
    movf   touche,w
    subwf  touche3,w
    btfss  STATUS,Z
    goto   erreur

;----- TEST TOUCHE 4 OU ANNUL -----
    call    testclavier
    movf   touche,w
    subwf  toucheA,w
    btfsc  STATUS,Z
    goto   annulation
    movf   touche,w
    subwf  touche4,w
    btfss  STATUS,Z
    goto   erreur

;----- NOUVEAU CODE -----
    bsf    LED
    movlw  d'10'
    call   tempoV
    bcf    LED

;----- ENTREE TOUCHE 1 OU ANNUL -----
    call    testclavier
    movf   touche,w
    subwf  toucheA,w
    btfsc  STATUS,Z
    goto   annulation
    movf   touche,w
    movwf  touche1_P

;----- ENTREE TOUCHE 2 OU ANNUL -----
    call    testclavier
    movf   touche,w
    subwf  toucheA,w
    btfsc  STATUS,Z
    goto   annulation
    movf   touche,w
    movwf  touche2_P

;----- ENTREE TOUCHE 3 OU ANNUL -----
    call    testclavier
    movf   touche,w
    subwf  toucheA,w
    btfsc  STATUS,Z
    goto   annulation
    movf   touche,w
    movwf  touche3_P

;----- ENTREE TOUCHE 4 OU ANNUL -----

```

```

    call    testclavier
    movf   touche,w
    subwf  toucheA,w
    btfsc  STATUS,Z
    goto   annulation
    movf   touche,w
    movwf  touche4_P

;----- CONFIRMATION DU CODE -----
    bsf    LED
    movlw  d'10'
    call   tempoV
    bcf    LED

;----- TEST TOUCHE 1 OU ANNUL -----
    call   testclavier
    movf   touche,w
    subwf  toucheA,w
    btfsc  STATUS,Z
    goto   annulation
    movf   touche,w
    subwf  touche1_P,w
    btfss  STATUS,Z
    goto   erreur

;----- TEST TOUCHE 2 OU ANNUL -----
    call   testclavier
    movf   touche,w
    subwf  toucheA,w
    btfsc  STATUS,Z
    goto   annulation
    movf   touche,w
    subwf  touche2_P,w
    btfss  STATUS,Z
    goto   erreur

;----- TEST TOUCHE 3 OU ANNUL -----
    call   testclavier
    movf   touche,w
    subwf  toucheA,w
    btfsc  STATUS,Z
    goto   annulation
    movf   touche,w
    subwf  touche3_P,w
    btfss  STATUS,Z
    goto   erreur

;----- TEST TOUCHE 4 OU ANNUL -----
    call   testclavier
    movf   touche,w
    subwf  toucheA,w
    btfsc  STATUS,Z
    goto   annulation
    movf   touche,w
    subwf  touche4_P,w
    btfss  STATUS,Z
    goto   erreur

;----- ENREGISTREMENT CODE -----
    bsf    LED
    clrf   addwrite
    movf   touche1_P,w
    WEEPROM
    incf   addwrite
    movf   touche2_P,w
    WEEPROM
    incf   addwrite
    movf   touche3_P,w
    WEEPROM
    incf   addwrite
    movf   touche4_P,w
    WEEPROM

;----- FIN PROGRAMMATION -----
    movlw  d'10'
    call   tempoV
    bcf    LED

```

```

        movlw  d'3'
        call   tempoV
annulation
        bsf    LED
        movlw  d'3'
        call   tempoV
        bcf    LED
        movlw  d'3'
        call   tempoV
        bsf    LED
        movlw  d'3'
        call   tempoV
        bcf    LED
        movlw  d'3'
        call   tempoV
        bsf    LED
        movlw  d'3'
        call   tempoV
        bcf    LED
        goto   init

;----- ERREUR PROGRAMMATION -----
erreur
        call   tempo
        call   ledx3
        call   tempo
        call   ledx3

        goto   start          ; boucler

;*****
;                               SOUS PROGRAMME                               *
;*****

;SOUS PROGRAMME CLAVIER
;-----
testclavier
        clrf   touche
        clrf   comptprog
        bcf    PORTB,1
        bsf    PORTB,2
        bsf    PORTB,3

;-----
test1
        btfs   PORTB,4
        goto   test4
        movlw  0x11
        call   applong
        goto   test1

test4
        btfs   PORTB,5
        goto   test7
        movlw  0x14
        call   applong
        goto   test4

test7
        btfs   PORTB,6
        goto   testA
        movlw  0x17
        call   applong
        goto   test7

testA
        btfs   PORTB,7
        goto   test2
        movlw  0x1a
        call   applong
        goto   testA

;-----
test2
        bsf    PORTB,1
        bcf    PORTB,2
        bsf    PORTB,3

        btfs   PORTB,4
        goto   test5
        movlw  0x12

```

```

        call    applong
        goto    test2
test5
        btfs   PORTB,5
        goto    test8
        movlw  0x15
        call    applong
        goto    test5
test8
        btfs   PORTB,6
        goto    test0
        movlw  0x18
        call    applong
        goto    test8
test0
        btfs   PORTB,7
        goto    test3
        movlw  0x10
        call    applong
        goto    test0

;-----
test3
        bsf     PORTB,1
        bsf     PORTB,2
        bcf     PORTB,3

        btfs   PORTB,4
        goto    test6
        movlw  0x13
        call    applong
        goto    test3
test6
        btfs   PORTB,5
        goto    test9
        movlw  0x16
        call    applong
        goto    test6
test9
        btfs   PORTB,6
        goto    testB
        movlw  0x19
        call    applong
        goto    test9
testB
        btfs   PORTB,7
        goto    fintest
        movlw  0x1b
        call    applong
        goto    testB

;-----
fintest
        btfs   APPTOUCHE
        goto    testclavier
        call   ledx1
        return

;SOUS PROGRAMME APPLONG
;-----
applong
        movwf  touche
        call   tempo
        incf  comptprog
        btfs  comptprog , 5
        return
        bsf   APPLONG
        bsf   LED
        return

;SOUS PROGRAMME TEMPO
;-----
tempo   movlw  d'1'           ;tempo fixe minimal
tempoV  movwf  tempo3       ;tempo variable de 1 à 255

```



```
        movlw d'150'
tempo_3 movwf tempo2
        movlw d'200'
tempo_2 movwf      tempo1
tempo_1 decfsz tempo1
        goto  tempo_1
        decfsz tempo2
        goto  tempo_2
        decfsz tempo3
        goto  tempo_3
        return

;SOUS PROGRAMME LED
;-----
ledx3
    bsf      LED
    call    tempo
    bcf      LED
    call    tempo
ledx2
    bsf      LED
    call    tempo
    bcf      LED
    call    tempo
ledx1
    bsf      LED
    call    tempo
    bcf      LED
    return

;*****
END                ; directive fin de programme
```