```
; PIC16F877A Configuration Bit Settings

; Assembly source line config statements

#include "p16f877a.inc"

; CONFIG
; __config 0x3F3A
 __CONFIG _FOSC_HS & _WDTE_OFF & _PWRTE_OFF & _BOREN_OFF & _LVP_OFF & _CPD_OFF & _WRT_OFF & _CP_OFF

include "p16f877A.inc"

CBLOCK 0x20
        pb1Last
        pb1Curr

        pb2Last
        pb2Curr

        pb3Last
        pb3Curr

        pb4Last
        pb4Curr

        pb5Last
        pb5Curr
```

```
        pb6Last
        pb6Curr

        pb7Last
        pb7Curr

        pb8Last
        pb8Curr

        d1
        d2
        d3
        d4
        d5
        d6
        d7
        d8
        d9
        d10

        pass
        target
        ctr
endc

org 0x000
goto Start
org 0x004
```

```
Interrupt
	retfie

Start
		bsf STATUS, RP0		; Bank 1
	bcf STATUS, RP1

		movlw b'11111111'
	movwf TRISB

		movlw b'00000000'
		movwf TRISC

		movlw b'00000000'
		movwf TRISD

		bcf STATUS, RP0		; Bank 0
		bcf STATUS, RP1

		clrf PORTB
		clrf PORTC
		clrf PORTD

		movlw B'0'
		movwf pb1Last

		movlw B'0'
		movwf pb1Curr
```

```
movlw B'0'
movwf pb2Last

movlw B'0'
movwf pb2Curr

movlw B'0'
movwf pb3Last

movlw B'0'
movwf pb3Curr

movlw B'0'
movwf pb4Last

movlw B'0'
movwf pb4Curr

movlw B'0'
movwf pb5Last

movlw B'0'
movwf pb5Curr

movlw B'0'
movwf pb6Last

movlw B'0'
```

```
        movwf pb6Curr

        movlw B'0'
        movwf pb7Last

        movlw B'0'
        movwf pb7Curr

        movlw B'0'
        movwf pb8Last

        movlw B'0'
        movwf pb8Curr
Main
        ;movlw b'11111111'
        ;movwf PORTC

CheckMain

        movlw B'4'
        andwf ctr

        bsf STATUS, RP0
    bcf STATUS, RP1

        btfss WREG
        goto ContinueChecking

ContinueChecking
```

```
        bcf STATUS, RP0
        bcf STATUS, RP

PB1Check

        btfss PORTB,0
        goto PB1Off

PB1On
        movlw B'00000001'
        movwf PORTC
        movlw D'1'
        addwf pass
        call Delay1s
        incf ctr
        goto PB2Check

PB1Off
        movlw B'00000000'
        movwf PORTC

        goto PB2Check
PB2Check

        btfss PORTB,1
        goto PB2Off

PB2On
        movlw B'00000010'
```

```
        movwf PORTC
        movlw D'2'
        addwf pass
        incf ctr
        call Delay1s
        goto PB3Check

PB2Off
        movlw B'00000000'
        movwf PORTC
        goto PB3Check

PB3Check
        btfss PORTB,2
        goto PB3Off

PB3On
        movlw B'00000100'
        movwf PORTC
        movlw D'3'
        addwf pass
        incf ctr
        call Delay1s
        goto  PB4Check

PB3Off
        movlw B'00000000'
        movwf PORTC
        goto PB4Check
```

```
PB4Check
        btfss PORTB,3
        goto PB4Off

PB4On
        movlw B'00001000'
        movwf PORTC
        movlw D'4'
        addwf pass
        incf ctr
        call Delay1s
        goto  PB5Check

PB4Off
        movlw B'00000000'
        movwf PORTC
        goto PB5Check

PB5Check
        btfss PORTB,4
        goto PB5Off

PB5On
        movlw B'00000001'
        movwf PORTC
        movlw D'5'
        addwf pass
        incf ctr
```

```
        call Delay1s
        goto  PB6Check

PB5Off
        movlw B'00000000'
        movwf PORTC
        goto PB6Check

PB6Check
        btfss PORTB,5
        goto PB6Off

PB6On
        movlw B'00000010'
        movwf PORTC
        movlw D'6'
        addwf pass
        incf ctr
        call Delay1s
        goto  PB7Check

PB6Off
        movlw B'00000000'
        movwf PORTC
        goto PB7Check

PB7Check
        btfss PORTB,6
        goto PB7Off
```

```
PB7On
        movlw B'00000100'
        movwf PORTC
        movlw D'7'
        addwf pass
        incf ctr
        call Delay1s
        goto  PB8Check

PB7Off
        movlw B'00000000'
        movwf PORTC
        goto PB8Check

PB8Check
        btfss PORTB,7
        goto PB8Off

PB8On
        movlw B'00001000'
        movwf PORTC
        movlw D'8'
        addwf pass
        incf ctr
        call Delay1s
        goto  DoneChecking

PB8Off
```

```
        movlw B'00000000'
        movwf PORTC
        goto DoneChecking

Delay1s
        movlw  0x44             ; We put '44' to W (accumulator). '44' data is prepared
        movwf  dc6              ; It will write the contents of the W register (0100 0100) into dc6
(memory address)
        movlw  0x23             ; We put '32' to W (accumulator). '32' data is prepared
        movwf  dc7              ; It will write the contents of the W register (0011 0010) into dc7
(memory address)
        movlw  0x06             ; We put '60' to W (accumulator). '60' data is prepared
        movwf  dc8              ; It will write the contents of the W register (0110 0000) into dc8
(memory address)

dLx
        decfsz   dc6,f          ; Since we use CBLOCK, we can get reduce the number of bits
until the dc6 = f
        goto     dLx            ; Since we use CBLOCK, we can call dLx with the value of dc6
        decfsz   dc7,f          ; Since we use CBLOCK, we can get reduce the number of bits
until the dc7 = f
        goto     dLx            ; Since we use CBLOCK, we can call dLx with the value of dc7

        decfsz   dc8,f          ; Since we use CBLOCK, we can get reduce the number of bits
until the dc8 = f
        goto     dLx            ; Since we use CBLOCK, we can call dLx with the value of dc8
        return

DoneChecking

        goto Main
```

```
end
```