Assembly Code with Comments and Flowchart

**Description of code:**

**Displaying a 0-9 counter on a seven-segment display with a short delay between each digit.**

;:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

; Configuration data

; PICmicro MCU type: 16F88

; Oscillator: RC mode, slow, VR1 fully anticlockwise (min.rate)

; LCD display: off

; 7-segment display: off

; Version 3 board settings: J14 links: Digital

;:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

;

; The following lines embed configuration data into the PICmicro

LIST P=16F88

\_\_CONFIG H'2007', H'3F7A' ; RC mode

\_\_CONFIG H'2008', H'3FFC' ; Clock Fail-Safe disabled

;:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

; setting Port B to output mode and turn on each led in sequence followed by turning each led off in sequence repeatedly.

; a short delay subroutine is called after each led on/off step

ORG 0 ; Reset vector

GOTO 5 ; Goto start of program

ORG 4 ; Interrupt vector

GOTO 5 ; Goto start of program

ORG 5 ; Start of program memory

;ASSEMBLER DIRECTIVES FOLLOW BELOW ;we will use EQU to make the program easier to read,

STATUS EQU H'03' ;by setting hex 03 equivalent to STATUS, we can replace H'03'with

;STATUS in the main program, thus we can read the code and understand

;what the line does

Z EQU 2

ANSEL EQU H'9B'

PORTB EQU H'06' ;make the word “PORTB” equivalent to H'06'

DIRB EQU H'86'

PORTA EQU H'05'

DIRA EQU H'85'

PAGE EQU 5 ; make PAGE equivalent to 5

COUNT EQU H'31'

PCL EQU 2 ;MAKE PCL equivalent to 2

;THE STEPS BELOW SET UP PORTB AND PORTA AS OUTPUT PORTS BY WRITING ALL ZEROS INTO THE PORTS DIRECTION REGISTERS WHICH ARE

;LOCATED IN PAGE1. THUS IT IS NECCESSARY TO CHANGE TO PAGE1 TO DO THIS AND CHANGE BACK TO PAGE0 WHEN DONE.

;PAGE0 IS SELECTED BY DEFAULT.

MAIN CLRF PORTB ; set all Port B pins to logic 0

CLRF PORTA

BSF STATUS,5 ; instruct program that Page 1 commands come next

CLRF ANSEL ; clear ANSEL register to make port pins digital

CLRF DIRA ;set contents of address to binary 0000 0000

CLRF DIRB ;set all Port B pins as outputs from within Page 1 USING PORTB DIRECTION REGISTER DIRB

BCF STATUS,5 ; instruct program that Page 0 commands come next

START MOVLW D'0' ;start of program, move literal decimal value 0 to working register

MOVWF COUNT

RPT MOVFW COUNT

CALL TABLE ;go to TABLE and return with literal hex value in working register, this will

;make the display show the numbers on the LED DISPLAY in order.

MOVWF PORTB

MOVLW D'12'

MOVWF PORTA ;copy the data from W to register f

CALL DELAY ;go to the delay subroutine

INCF COUNT ;add one to the contents of the address

MOVFW COUNT

ADDLW H'F6' ;add W to contents of address, store result at address

BTFSS STATUS,Z ;bit test file in status file check is count is equal to zero, if zero set z

; to satus register.

GOTO RPT ;jump to RPT and repeat that code over again

GOTO START ;go to the START of the program and repeat the code again

TABLE ADDWF PCL,F

RETLW H'C0' ; H'C0' is the corresponding hex code for showing number 0 on the display

RETLW H'F9' ; H'F9' is the corresponding hex code for showing number 1 on the display

RETLW H'A4' ; H'A4' is the corresponding hex code for showing number 2 on the display

RETLW H'B0' ; H'B0' is the corresponding hex code for showing number 3 on the display

RETLW H'99' ; H'99' is the corresponding hex code for showing number 4 on the display

RETLW H'92' ; H'92' is the corresponding hex code for showing number 5 on the display

RETLW H'83' ; H'83' is the corresponding hex code for showing number 6 on the display

RETLW H'F8' ; H'F8' is the corresponding hex code for showing number 7 on the display

RETLW H'80' ; H'80' is the corresponding hex code for showing number 8 on the display

RETLW H'98' ; H'98' is the corresponding hex code for showing number 9 on the display

;DELAY SUBROUTINE FOLLOWS BELOW

DELAY MOVLW H'01' ; this is the delay subroutine which loads

LOOP ADDLW H'FF' ; the working register W with a number delayval

BTFSS STATUS,Z ; and counts it down to zero thus creating a delay.

GOTO LOOP ; when finished it returns to the calling (main) routine

RETURN

END ; final statement, required to end the program

flowchart, this program is in a loop.

Yes

No

Is count = 10 ?

Increment count

Call Delay

Display count variable on 7 SEG. Display

Set up count variable

Set to zero

Set up both ports as output ports