

**Problem Description:**

The embedded system built in this assignment implements a simple circuit that connects a push-button input (PORT B) to a LED output (PORT D). In addition to this, the LED is also synced to a 16 MHz external oscillator with a preset counter. Every time the counter completes one cycle and issues an interrupt, the LED will blink off and on. When the button is not pushed, the LED will blink at a frequency of about 1Hz. When the button is pushed, the timer will decrease its counter value, thus increasing the frequency to around 10Hz.

The calculations used to acquire the two frequencies are as follows:

$$F_{\text{EXOSC}} = 16 \text{ MHz} = 16000000, \text{ with 4 cycles per instruction}$$

$$F_{\text{OSC}} = 16000000 / 4 = 4000000 \text{ Hz} = 4 \text{ MHz}$$

Pre-scalar set to 1:64

$$F_{\text{TIMER}} = 4000000 / 64 = 62500 \text{ Hz} = 6.25 \text{ KHz}$$

$$T_{\text{TIMER}} = 1 / 62500 = 0.000016 \text{ s} = 16 \mu\text{s}$$

**OFF:**

$$\text{COUNT} = 2^{16} = 65536 \text{ iterations, @ } 16\mu\text{s per iteration}$$

$$T_{\text{OFF}} = 16\mu\text{s} * 65536 = 1.05 \text{ s} \sim 1 \text{ s}$$

$$F_{\text{OFF}} = 1 / 1.05 = 0.95 \text{ Hz} \sim 1 \text{ Hz}$$

**ON:**

$$\text{COUNT (reduced by 90\%)} = 2^{16} - (2^{16} * 0.90) = 65536 - 58983 = 6553, \text{ @ } 16\mu\text{s per iteration}$$

$$T_{\text{ON}} = 16\mu\text{s} * 6553 = 0.105 \sim 0.1 \text{ s}$$

$$F_{\text{ON}} = 1 / 0.105 = 9.52 \sim 10 \text{ Hz}$$

**Assembly Code:**

```

#include <p18f4550.inc>
CONFIG WDT=OFF      ; disable watchdog timer
CONFIG MCLRE = ON   ; MCLR Pin on
CONFIG DEBUG = ON   ; Enable Debug Mode
CONFIG LVP = OFF    ; Low-Voltage programming disabled (necessary for debugging)
CONFIG FOSC = HS    ; External oscillator, port function on RA6
org 0

Start:
CLRF PORTB          ; Clear PORTB
SETF TRISB          ; Set TRISB to input
MOVLW B'11111111'
MOVWF ADCON1
CLRF PORTD          ; Clear PORTD
CLRF TRISD          ; Set PORTD to output
MOVLW B'00000101'   ; 16 bit Timer control set, PRESCALAR OF 64
MOVWF T0CON
CLRF TMR0H
CLRF TMR0L

MainLoop:
BTFSS PORTB,0       ; Test bit if set, skip next instruction
GOTO Push_Button_OFF

Push_Button_ON:
MOVLW B'11100110'   ;
MOVWF TMR0H          ; LOAD THE HIGHER BYTE
MOVLW B'01100110'   ;
MOVWF TMR0L          ; LOAD THE LOWER BYTE
GOTO HERE

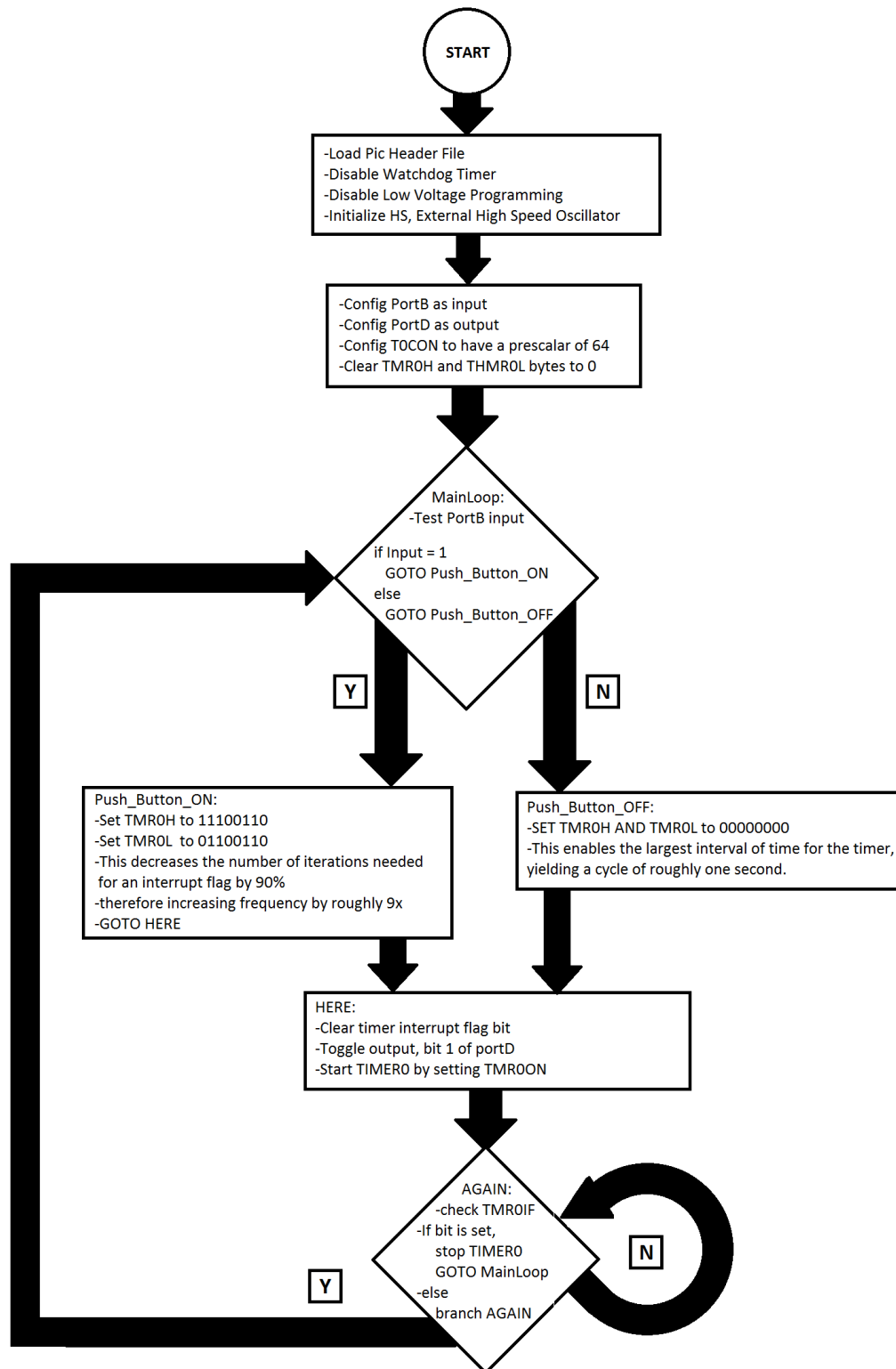
Push_Button_OFF:
MOVLW B'00000000'
MOVWF TMR0H          ; LOAD THE HIGHER BYTE
MOVLW B'00000000'
MOVWF TMR0L          ; LOAD THE LOWER BYTE

HERE:
BCF INTCON, TMR0IF   ; CLEAR TIMER INTERRUPT FLAG BIT
BTG PORTD,1          ; TOGGLE BIT 1 PORT D
BSF T0CON, TMR0ON    ; START TIMER0

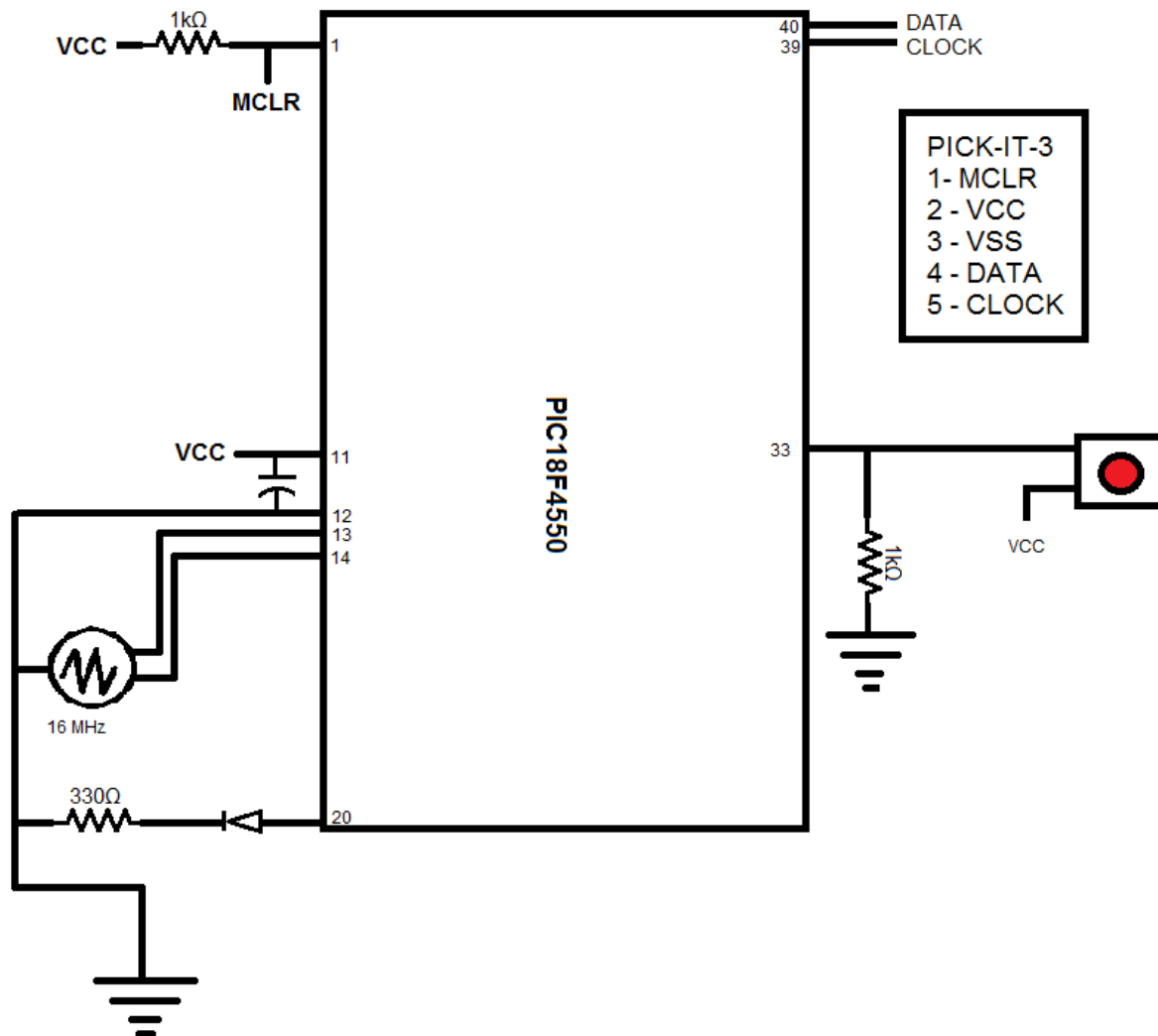
AGAIN:
BTFSS INTCON, TMR0IF ; MONITOR TIMER0 FLAG UNTILL
BRA AGAIN            ; IT ROLLS OVER
BCF T0CON, TMR0ON    ; STOP TIMER0
GOTO MainLoop        ; GOTO mainloop
end

```

**Pseudocode Flow Chart:**



**Circuit Diagram:**



**Circuit Actual Implementation:**

