Code

P10.ASM

```
;**********************************************************************
;                                                                     *
;   This file is a basic code template for assembly code generation   *
;   on the PIC16F84A. This file contains the basic code              *
;   building blocks to build upon.                                    *
;                                                                     *
;   Refer to the MPASM User's Guide for additional information on     *
;   features of the assembler (Document DS33014).                     *
;                                                                     *
;   Refer to the respective PIC data sheet for additional             *
;   information on the instruction set.                               *
;                                                                     *
;**********************************************************************
;                                                                     *
;    Filename:        P10.asm                                         *
;    Date:            2024-05-01T20:00:00-04:00                       *
;    File Version:    v1.0.0                                          *
;                                                                     *
;    Author:          Trevor Nichols                                  *
;    Company:         Case Western Reserve University                 *
;                                                                     *
;                                                                     *
;**********************************************************************
;                                                                     *
;    Files required: P16F84A.INC                                      *
;                                                                     *
;                                                                     *
;                                                                     *
;**********************************************************************
;                                                                     *
;    Notes:                                                           *
;                                                                     *
;                                                                     *
;                                                                     *
;                                                                     *
;**********************************************************************


    list      p=16F84A             ; list directive to define processor
    #include <p16F84a.inc>         ; processor specific variable
definitions
```

```
    __CONFIG   _CP_OFF & _WDT_ON & _PWRTE_ON & _RC_OSC

; '__CONFIG' directive is used to embed configuration data within .asm
file.
; The lables following the directive are located in the respective .inc
file.
; See respective data sheet for additional information on configuration
word.

;***** VARIABLE DEFINITIONS
w_temp        EQU     0×0C        ; variable used for context saving
status_temp   EQU     0×0D        ; variable used for context saving
ctr           EQU     0×0E
ctr_2         EQU     0×0F

;**********************************************************************
RESET_VECTOR      CODE    0×0000  ; processor reset vector
        goto    start             ; go to beginning of program

ISR               CODE    0×0004  ; interrupt vector location

Interrupt:

        movwf  w_temp             ; save off current W register contents
        movf   STATUS,w           ; move status register into W register
        movwf  status_temp        ; save off contents of STATUS register

;  Place ISR Here

        movf   status_temp,w      ; retrieve copy of STATUS register
        movwf  STATUS             ; restore pre-isr STATUS register
contents
        swapf  w_temp,f
        swapf  w_temp,w           ; restore pre-isr W register contents
        retfie                    ; return from interrupt

MAIN_PROGRAM    CODE

start:

; remaining code goes here

        bsf STATUS, RP0
        movlw 0×FF
        movwf TRISA
        movlw 0×00
        movwf TRISB
        bcf STATUS, RP0

PRES1:
        clrf PORTB
        bsf PORTB, 0
        movlw 0x E4
        movwf ctr
        movlw 0x09
        movwf ctr_2
S1:
        btfsc PORTA, 1
        goto PRESERR
        btfsc PORTA, 2
        goto PRESERR
        btfsc PORTA, 3
        goto PRESERR
        btfsc PORTA, 0
        goto PRESOK
        decfsz ctr, 1
        goto S1
        movlw 0xFF
        movwf ctr
        decfsz ctr_2, 1
        goto S1
        goto PRES2

PRES2:
        clrf PORTB
        bsf PORTB, 1
        movlw 0xE4
        movwf ctr
        movlw 0x09
        movwf ctr_2
S2:
        btfsc PORTA, 0
        goto PRESERR
        btfsc PORTA, 2
        goto PRESERR
        btfsc PORTA, 3
        goto PRESERR
        btfsc PORTA, 1
        goto PRESOK
        decfsz ctr, 1
        goto S2
        movlw 0xFF
        movwf ctr
        decfsz ctr_2, 1
        goto S2
        goto PRES3

PRES3:
        clrf PORTB
        bsf PORTB, 2
        movlw 0x09
        movwf ctr_2
S3:
        btfsc PORTA, 0
        goto PRESERR
        btfsc PORTA, 1
        goto PRESERR
        btfsc PORTA, 3
        goto PRESERR
        btfsc PORTA, 2
        goto PRESOK
        decfsz ctr, 1
        goto S3
        movlw 0xFF
        movwf ctr
        decfsz ctr_2, 1
        goto S3
        goto PRES4

PRES4:
        clrf PORTB
        bsf PORTB, 3
        movlw 0xE4
        movwf ctr
        movlw 0x09
        movwf ctr_2
S4:
        btfsc PORTA, 0
        goto PRESERR
        btfsc PORTA, 1
        goto PRESERR
        btfsc PORTA, 2
        goto PRESERR
        btfsc PORTA, 3
        goto PRESOK
        decfsz ctr, 1
        goto S4
        movlw 0xFF
        movwf ctr
        decfsz ctr_2, 1
        goto S4
        goto PRES1

PRESOK:
        clrf PORTB
        bsf PORTB, 5
        movlw 0xE4
        movwf ctr
        movlw 0x09
        movwf ctr_2
SOK:
        btfsc PORTA, 0
        goto PRESOK
        btfsc PORTA, 1
        goto PRESOK
        btfsc PORTA, 2
        goto PRESOK
        btfsc PORTA, 3
        goto PRESOK
        decfsz ctr, 1
        goto SOK
        movlw 0xFF
        movwf ctr
        decfsz ctr_2, 1
        goto SOK
        goto PRES1


PRESERR:
        clrf PORTB
        bsf PORTB, 4
        movlw 0xE4
        movwf ctr
        movlw 0x09
        movwf ctr_2
SERR:
        btfsc PORTA, 0
        goto PRESERR
        btfsc PORTA, 1
        goto PRESERR
        btfsc PORTA, 2
        goto PRESERR
        btfsc PORTA, 3
        goto PRESERR
        decfsz ctr, 1
        goto SERR
        movlw 0xFF
        movwf ctr
        decfsz ctr_2, 1
        goto SERR
        goto PRES1

END:
        goto END
```

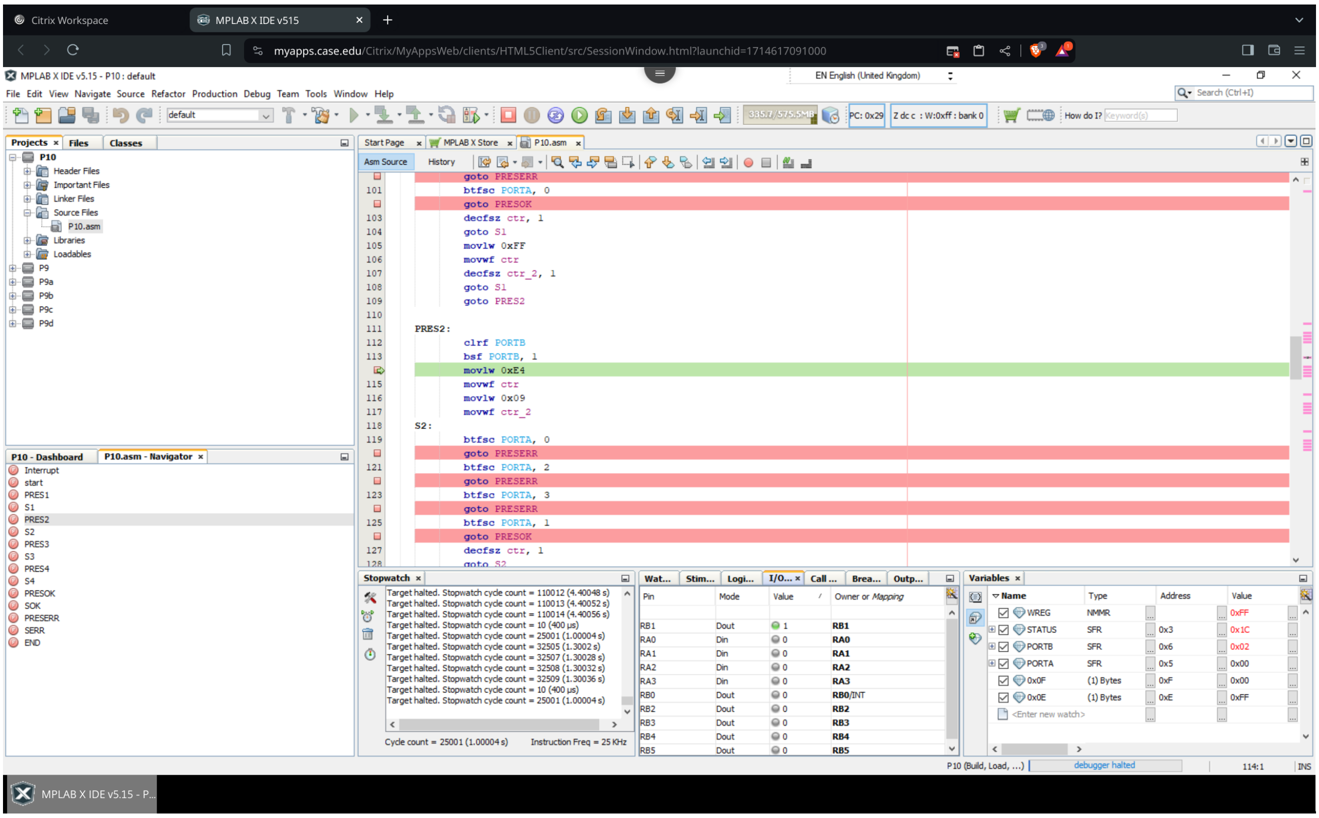
This code will advance to the next state if nothing is pressed after 1 second of inactivity.

If an incorrect button is pressed, the state will immediately shift over to the error light. If a correct button is pressed and no incorrect buttons are pressed, then the state will immediately shift over to the OK light.

While in the error state or OK state, the state will not change until all buttons are unpressed. A one second delay will begin once all buttons are unpressed. Pressing buttons during this cooldown state will reset the cooldown.

Once the cooldown is finished in the ERR or OK state, return to the S1 state.

This code when run on 25kHz instruction frequency has cooldowns accurate to exactly $0.99996\ \mathrm{s}$



```
        goto PRESERR
        btfsc PORTA, 0
        goto PRESOK
        decfsz ctr, 1
        goto S1
        movlw 0xFF
        movwf ctr
        decfsz ctr_2, 1
        goto S1
        goto PRES2

PRES2:
        clrf PORTB
        bsf PORTB, 1
        movlw 0xE4
        movwf ctr
        movlw 0x09
        movwf ctr_2
S2:
        btfsc PORTA, 0
        goto PRESERR
        btfsc PORTA, 2
        goto PRESERR
        btfsc PORTA, 3
        goto PRESERR
        btfsc PORTA, 1
        goto PRESOK
        decfsz ctr, 1
        goto S2
```

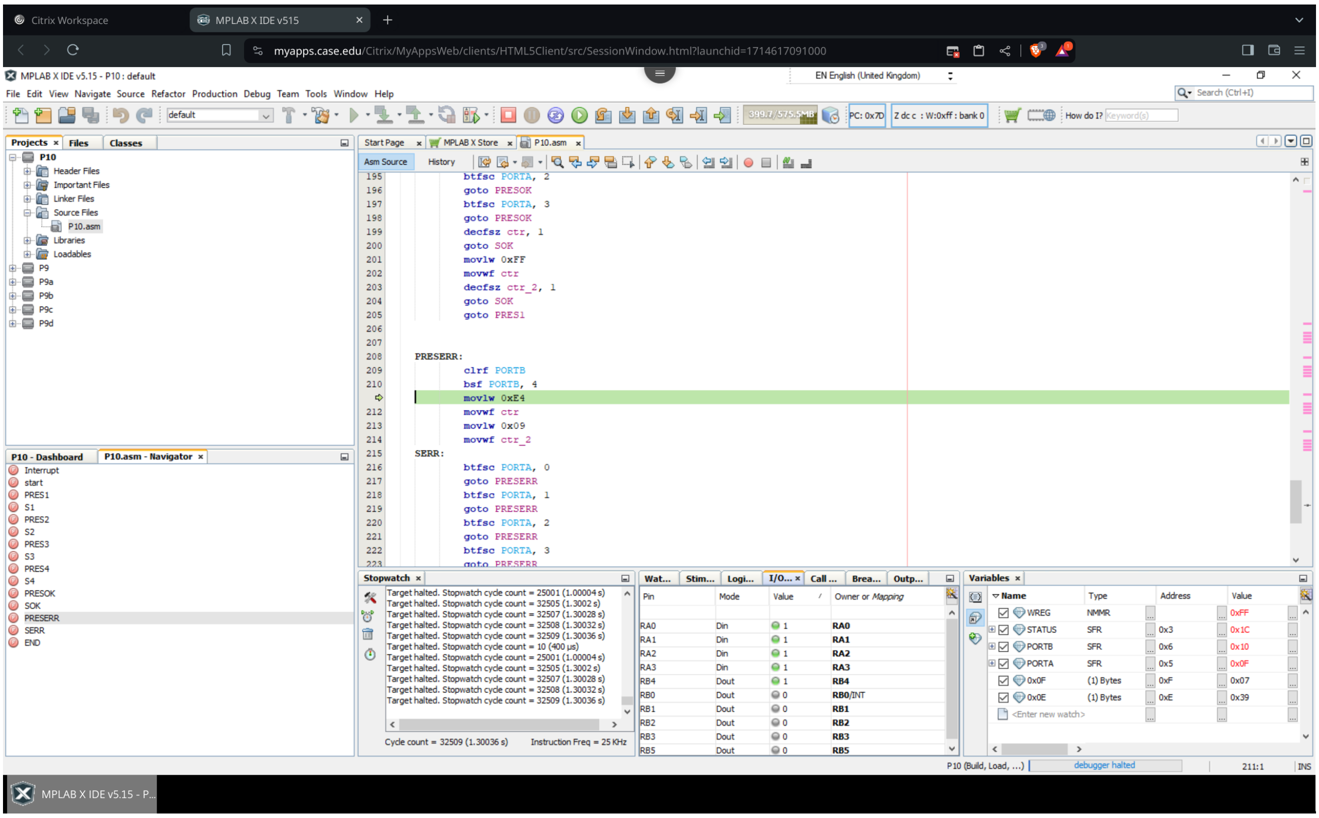
```
Target halted. Stopwatch cycle count = 110012 (4.40048 s)
Target halted. Stopwatch cycle count = 110013 (4.40052 s)
Target halted. Stopwatch cycle count = 110014 (4.40056 s)
Target halted. Stopwatch cycle count = 10 (400 µs)
Target halted. Stopwatch cycle count = 25001 (1.00004 s)
Target halted. Stopwatch cycle count = 32505 (1.3002 s)
Target halted. Stopwatch cycle count = 32507 (1.30028 s)
Target halted. Stopwatch cycle count = 32508 (1.30032 s)
Target halted. Stopwatch cycle count = 32509 (1.30036 s)
Target halted. Stopwatch cycle count = 10 (400 µs)
Target halted. Stopwatch cycle count = 25001 (1.00004 s)

Cycle count = 25001 (1.00004 s)        Instruction Freq = 25 KHz
```

Pin	Mode	Value	Owner or *Mapping*
RB1	Dout	● 1	**RB1**
RA0	Din	● 0	**RA0**
RA1	Din	● 0	**RA1**
RA2	Din	● 0	**RA2**
RA3	Din	● 0	**RA3**
RB0	Dout	● 0	**RB0**/INT
RB2	Dout	● 0	**RB2**
RB3	Dout	● 0	**RB3**
RB4	Dout	● 0	**RB4**
RB5	Dout	● 0	**RB5**

Name	Type	Address	Value
☑ WREG	NMMR		0xFF
☑ STATUS	SFR	0x3	0x1C
☑ PORTB	SFR	0x6	0x02
☑ PORTA	SFR	0x5	0x00
☑ 0x0F	(1) Bytes	0xF	0x00
☑ 0x0E	(1) Bytes	0xE	0xFF
<Enter new watch>			

P10 (Build, Load, ...) debugger halted 114:1 INS

At $1.00004\ \mathrm{s}$, we progress into state 2, with the its corresponding LED on, as there has been no interaction with the device yet



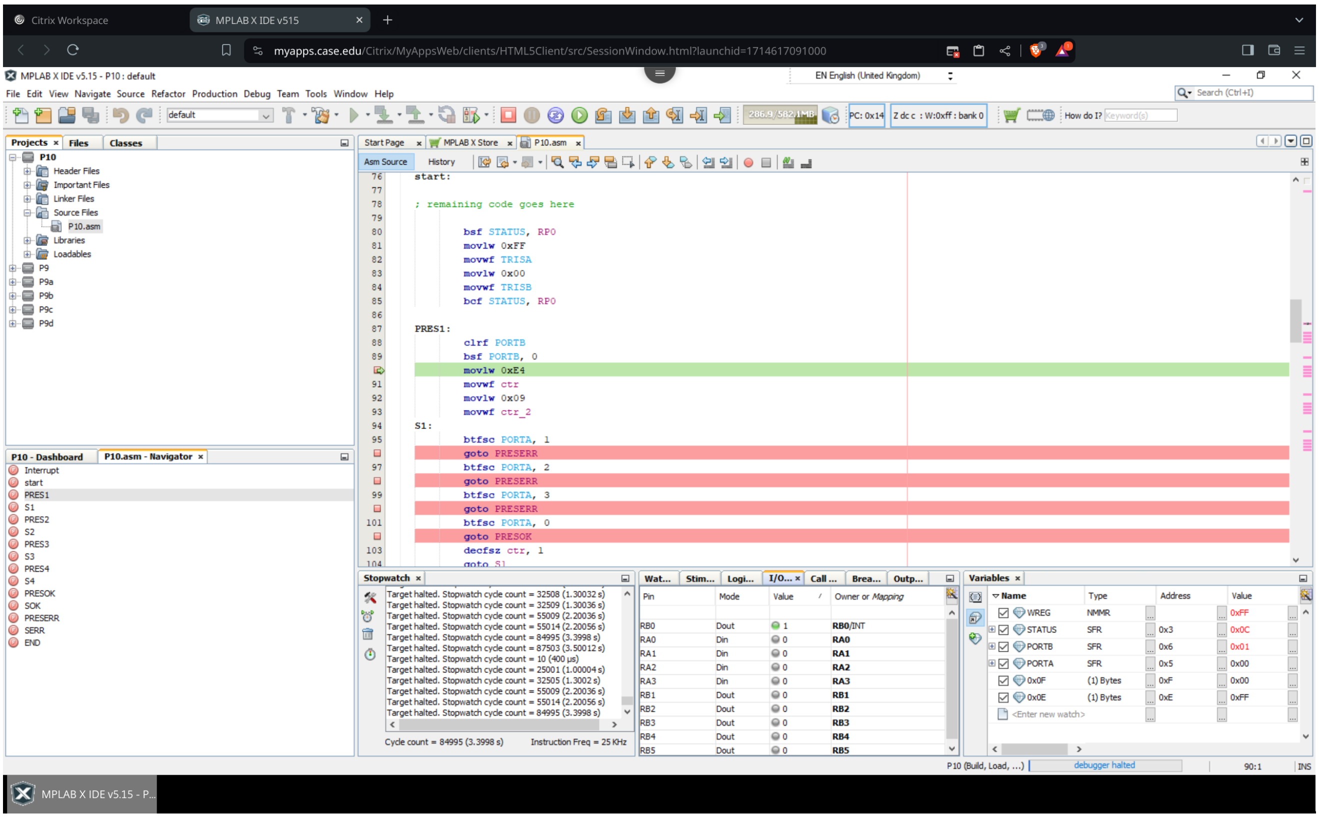
```
        btfsc PORTA, 2
        goto PRESOK
        btfsc PORTA, 3
        goto PRESOK
        decfsz ctr, 1
        goto SOK
        movlw 0xFF
        movwf ctr
        decfsz ctr_2, 1
        goto SOK
        goto PRES1


PRESERR:
        clrf PORTB
        bsf PORTB, 4
        movlw 0xE4
        movwf ctr
        movlw 0x09
        movwf ctr_2
SERR:
        btfsc PORTA, 0
        goto PRESERR
        btfsc PORTA, 1
        goto PRESERR
        btfsc PORTA, 2
        goto PRESERR
        btfsc PORTA, 3
        goto PRESERR
```

```
Target halted. Stopwatch cycle count = 25001 (1.00004 s)
Target halted. Stopwatch cycle count = 32505 (1.3002 s)
Target halted. Stopwatch cycle count = 32507 (1.30028 s)
Target halted. Stopwatch cycle count = 32508 (1.30032 s)
Target halted. Stopwatch cycle count = 32509 (1.30036 s)
Target halted. Stopwatch cycle count = 10 (400 µs)
Target halted. Stopwatch cycle count = 25001 (1.00004 s)
Target halted. Stopwatch cycle count = 32505 (1.3002 s)
Target halted. Stopwatch cycle count = 32507 (1.30028 s)
Target halted. Stopwatch cycle count = 32508 (1.30032 s)
Target halted. Stopwatch cycle count = 32509 (1.30036 s)

Cycle count = 32509 (1.30036 s)     Instruction Freq = 25 KHz
```

Pin	Mode	Value	Owner or *Mapping*
RA0	Din	1	**RA0**
RA1	Din	1	**RA1**
RA2	Din	1	**RA2**
RA3	Din	1	**RA3**
RB4	Dout	1	**RB4**
RB0	Dout	0	**RB0**/INT
RB1	Dout	0	**RB1**
RB2	Dout	0	**RB2**
RB3	Dout	0	**RB3**
RB5	Dout	0	**RB5**

Name	Type	Address	Value
☑ WREG	NMMR		0xFF
☑ STATUS	SFR	0x3	0x1C
☑ PORTB	SFR	0x6	0x10
☑ PORTA	SFR	0x5	0x0F
☑ 0x0F	(1) Bytes	0xF	0x07
☑ 0x0E	(1) Bytes	0xE	0x39
<Enter new watch>			

At 1.30036 s, we progress into the error state as all buttons were pressed at 1.3 s while the correct state was 2



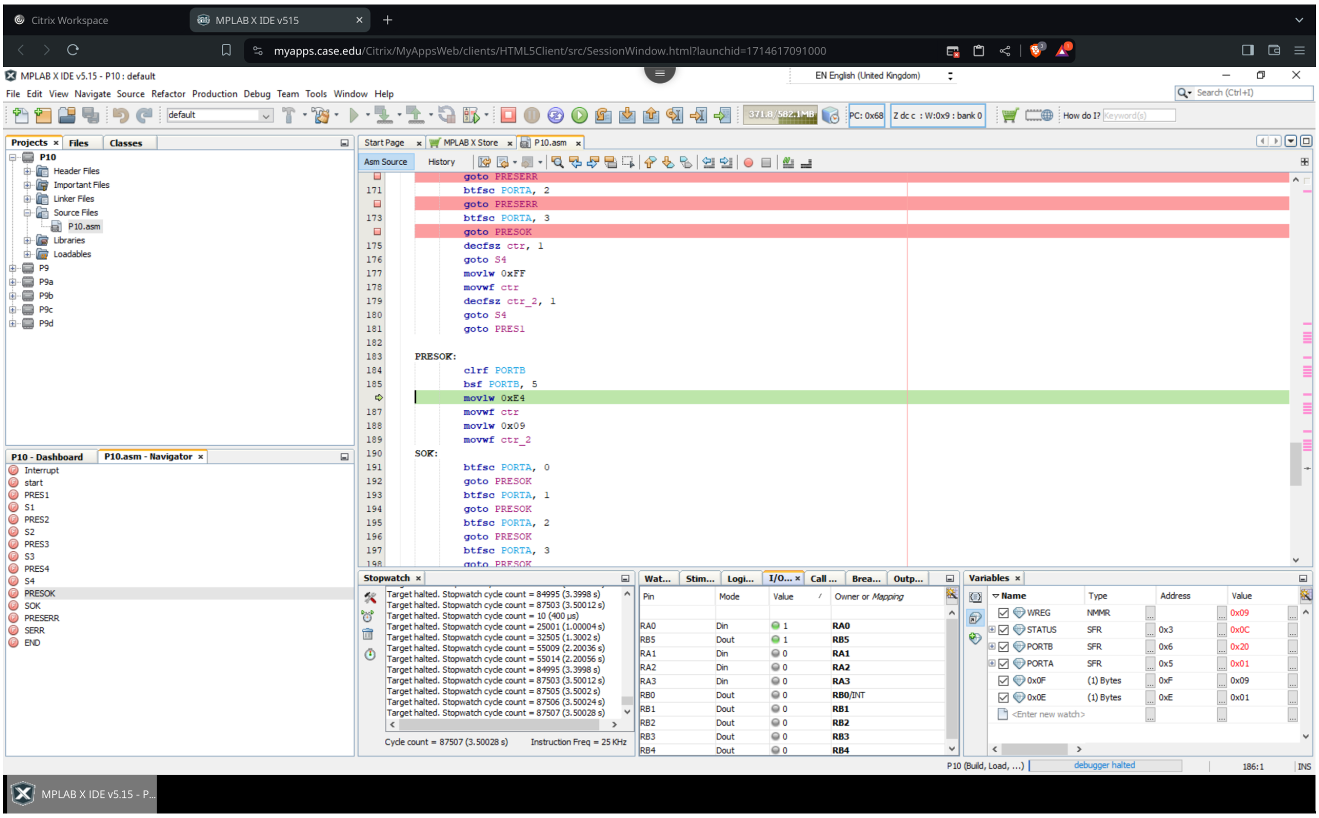
```
start:

; remaining code goes here

        bsf STATUS, RP0
        movlw 0xFF
        movwf TRISA
        movlw 0x00
        movwf TRISB
        bcf STATUS, RP0

PRES1:
        clrf PORTB
        bsf PORTB, 0
        movlw 0xE4
        movwf ctr
        movlw 0x09
        movwf ctr_2
S1:
        btfsc PORTA, 1
        goto PRESERR
        btfsc PORTA, 2
        goto PRESERR
        btfsc PORTA, 3
        goto PRESERR
        btfsc PORTA, 0
        goto PRESOK
        decfsz ctr, 1
        goto S1
```

```
Target halted. Stopwatch cycle count = 32508 (1.30032 s)
Target halted. Stopwatch cycle count = 32509 (1.30036 s)
Target halted. Stopwatch cycle count = 55009 (2.20036 s)
Target halted. Stopwatch cycle count = 55014 (2.20056 s)
Target halted. Stopwatch cycle count = 84995 (3.3998 s)
Target halted. Stopwatch cycle count = 87503 (3.50012 s)
Target halted. Stopwatch cycle count = 10 (400 µs)
Target halted. Stopwatch cycle count = 25001 (1.00004 s)
Target halted. Stopwatch cycle count = 32505 (1.3002 s)
Target halted. Stopwatch cycle count = 55009 (2.20036 s)
Target halted. Stopwatch cycle count = 55014 (2.20056 s)
Target halted. Stopwatch cycle count = 84995 (3.3998 s)

Cycle count = 84995 (3.3998 s)     Instruction Freq = 25 KHz
```

Pin	Mode	Value	Owner or *Mapping*
RB0	Dout	1	**RB0**/INT
RA0	Din	0	**RA0**
RA1	Din	0	**RA1**
RA2	Din	0	**RA2**
RA3	Din	0	**RA3**
RB1	Dout	0	**RB1**
RB2	Dout	0	**RB2**
RB3	Dout	0	**RB3**
RB4	Dout	0	**RB4**
RB5	Dout	0	**RB5**

Name	Type	Address	Value
☑ WREG	NMMR		0xFF
☑ STATUS	SFR	0x3	0x0C
☑ PORTB	SFR	0x6	0x01
☑ PORTA	SFR	0x5	0x00
☑ 0x0F	(1) Bytes	0xF	0x00
☑ 0x0E	(1) Bytes	0xE	0xFF
<Enter new watch>			

At 3.3998 s, we progress back into S1 as all buttons were released at 2.4 s



```
        goto PRESERR
        btfsc PORTA, 2
        goto PRESERR
        btfsc PORTA, 3
        goto PRESOK
        decfsz ctr, 1
        goto S4
        movlw 0xFF
        movwf ctr
        decfsz ctr_2, 1
        goto S4
        goto PRES1

PRESOK:
        clrf PORTB
        bsf PORTB, 5
        movlw 0xE4
        movwf ctr
        movlw 0x09
        movwf ctr_2
SOK:
        btfsc PORTA, 0
        goto PRESOK
        btfsc PORTA, 1
        goto PRESOK
        btfsc PORTA, 2
        goto PRESOK
        btfsc PORTA, 3
        goto PRESOK
```

Stopwatch

```
Target halted. Stopwatch cycle count = 84995 (3.3998 s)
Target halted. Stopwatch cycle count = 87503 (3.50012 s)
Target halted. Stopwatch cycle count = 10 (400 μs)
Target halted. Stopwatch cycle count = 25001 (1.00004 s)
Target halted. Stopwatch cycle count = 32505 (1.3002 s)
Target halted. Stopwatch cycle count = 55009 (2.20036 s)
Target halted. Stopwatch cycle count = 55014 (2.20056 s)
Target halted. Stopwatch cycle count = 84995 (3.3998 s)
Target halted. Stopwatch cycle count = 87503 (3.50012 s)
Target halted. Stopwatch cycle count = 87505 (3.5002 s)
Target halted. Stopwatch cycle count = 87506 (3.50024 s)
Target halted. Stopwatch cycle count = 87507 (3.50028 s)
```

Cycle count = 87507 (3.50028 s)   Instruction Freq = 25 KHz

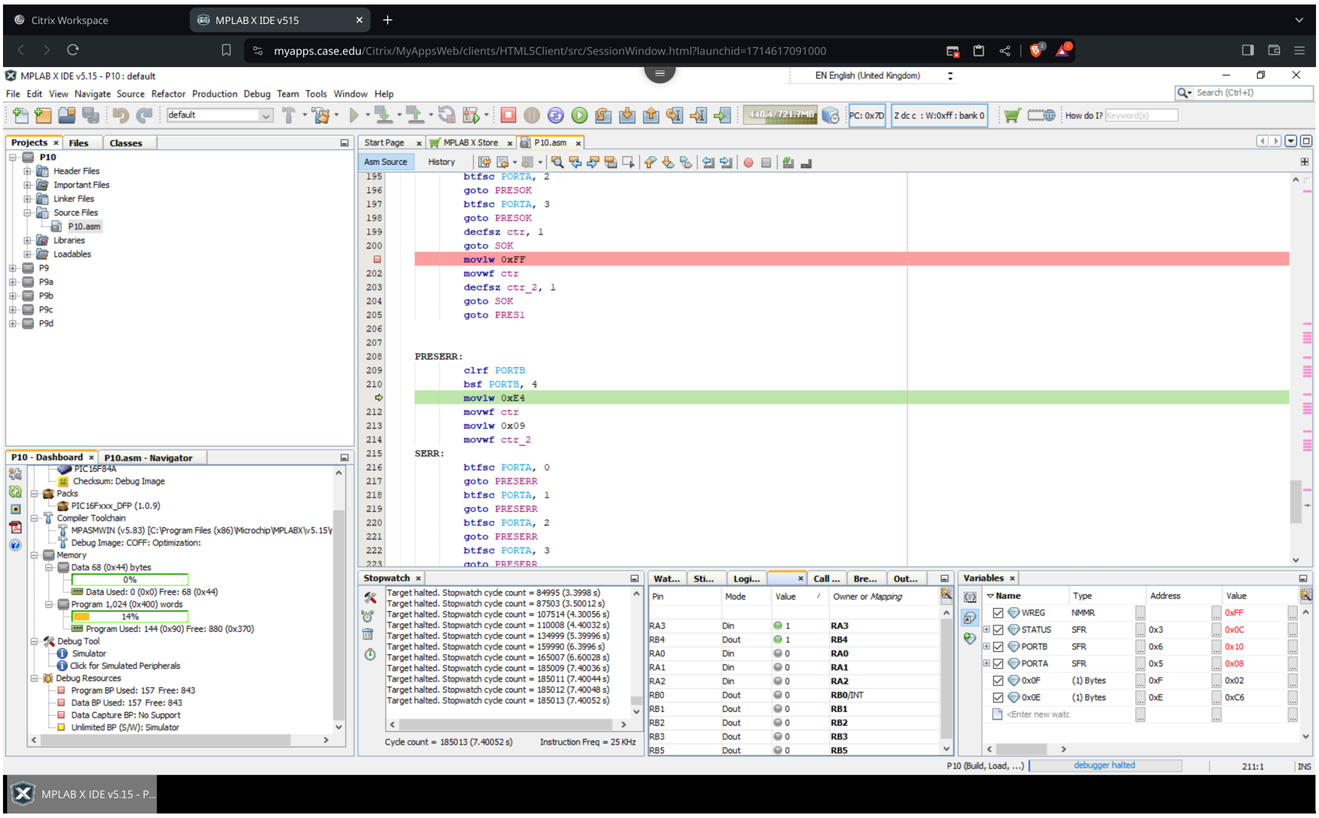
Pin	Mode	Value	Owner or *Mapping*
RA0	Din	● 1	**RA0**
RB5	Dout	● 1	**RB5**
RA1	Din	● 0	**RA1**
RA2	Din	● 0	**RA2**
RA3	Din	● 0	**RA3**
RB0	Dout	● 0	**RB0**/INT
RB1	Dout	● 0	**RB1**
RB2	Dout	● 0	**RB2**
RB3	Dout	● 0	**RB3**
RB4	Dout	● 0	**RB4**

Variables

Name	Type	Address	Value
☑ WREG	NMMR		0x09
☑ STATUS	SFR	0x3	0x0C
☑ PORTB	SFR	0x6	0x20
☑ PORTA	SFR	0x5	0x01
☑ 0x0F	(1) Bytes	0xF	0x09
☑ 0x0E	(1) Bytes	0xE	0x01
<Enter new watch>			

P10 (Build, Load, ...) debugger halted 186:1 INS

At 3.50028 s, we progress into the success state, as the first button was pressed at 3.5 s, 0.1 s after moving to S1.



PIC16F84A
Checksum: Debug Image
Packs
PIC16Fxxx_DFP (1.0.9)
Compiler Toolchain
MPASMWIN (v5.83) [C:\Program Files (x86)\Microchip\MPLABX\v5.15\
Debug Image: COFF: Optimization:
Memory
Data 68 (0x44) bytes
0%
Data Used: 0 (0x0) Free: 68 (0x44)
Program 1,024 (0x400) words
14%
Program Used: 144 (0x90) Free: 880 (0x370)
Debug Tool
Simulator
Click for Simulated Peripherals
Debug Resources
Program BP Used: 157 Free: 843
Data BP Used: 157 Free: 843
Data Capture BP: No Support
Unlimited BP (S/W): Simulator

```
        btfsc PORTA, 2
        goto PRESOK
        btfsc PORTA, 3
        goto PRESOK
        decfsz ctr, 1
        goto SOK
        movlw 0xFF
        movwf ctr
        decfsz ctr_2, 1
        goto SOK
        goto PRES1


PRESERR:
        clrf PORTB
        bsf PORTB, 4
        movlw 0xE4
        movwf ctr
        movlw 0x09
        movwf ctr_2
SERR:
        btfsc PORTA, 0
        goto PRESERR
        btfsc PORTA, 1
        goto PRESERR
        btfsc PORTA, 2
        goto PRESERR
        btfsc PORTA, 3
        goto PRESERR
```

Stopwatch

```
Target halted. Stopwatch cycle count = 84995 (3.3998 s)
Target halted. Stopwatch cycle count = 87503 (3.50012 s)
Target halted. Stopwatch cycle count = 107514 (4.30056 s)
Target halted. Stopwatch cycle count = 110008 (4.40032 s)
Target halted. Stopwatch cycle count = 134999 (5.39996 s)
Target halted. Stopwatch cycle count = 159990 (6.3996 s)
Target halted. Stopwatch cycle count = 165007 (6.60028 s)
Target halted. Stopwatch cycle count = 185009 (7.40036 s)
Target halted. Stopwatch cycle count = 185011 (7.40044 s)
Target halted. Stopwatch cycle count = 185012 (7.40048 s)
Target halted. Stopwatch cycle count = 185013 (7.40052 s)
```

Cycle count = 185013 (7.40052 s)   Instruction Freq = 25 KHz

Pin	Mode	Value	Owner or *Mapping*
RA3	Din	● 1	**RA3**
RB4	Dout	● 1	**RB4**
RA0	Din	● 0	**RA0**
RA1	Din	● 0	**RA1**
RA2	Din	● 0	**RA2**
RB0	Dout	● 0	**RB0**/INT
RB1	Dout	● 0	**RB1**
RB2	Dout	● 0	**RB2**
RB3	Dout	● 0	**RB3**
RB5	Dout	● 0	**RB5**

Variables

Name	Type	Address	Value
☑ WREG	NMMR		0xFF
☑ STATUS	SFR	0x3	0x0C
☑ PORTB	SFR	0x6	0x10
☑ PORTA	SFR	0x5	0x08
☑ 0x0F	(1) Bytes	0xF	0x02
☑ 0x0E	(1) Bytes	0xE	0xC6
<Enter new watc			

P10 (Build, Load, ...) debugger halted 211:1 INS

At 7.40052 s, we progress back into the error state as the wrong button was pressed