

```
list P=PIC16F690
#include "p16F690.inc"
__config(_CP_OFF & _WDT_OFF & _PWRTE_ON & _INTOSCIO )
```

```
    cblock 0x72
    ;DataByte      ; Transmit data
    SwitchVar     ; SwitchVariable
    RXData        ; Received data
    ChargeState   ; State of battery charge
    CharDischar   ; Charge/Discharge status
    LampStatus    ; Lamp status
    DutyCycle     ; Duty Cycle
    ERR           ; Testing error
    endc
```

```
DataByte     equ     0x71; Transmit data
```

```
; Set up SSP, master mode OSC/64, look for 1st edge and transmit happens on rising edge
Period       equ     d'194' ; PWM period of 6.25ms, 40Hz (PR2)
SPI_Mode     equ     0x32   ; 0011 0010 set SSPEN=1|CKP=1
TRISASetup  equ     0xFF   ; 1111 1111 set RA0,1,2 as inputs for Lamp status
TRISBSetup  equ     0x30   ; 0001 0000
TRISCSetup  equ     0x02   ; 0000 0010
CCPsetup    equ     0x0C   ; 00__ 1100 set Compare mode to set output on match
T2Setup     equ     0x05   ; 0000 0101 set prescaler to 4
;TM1setup   equ     0x31   ; 00110001 setup for Timer1 (enable timer 1, use internal clock,
prescale 8)
;
```

```
    org     0
    Goto   Main
;org     4
;Goto   ISR_ROUTINE
    org     5
```

```
Main:
```

```
;
;
```

```
INIT:
```

```
    CALL    OSCCON_INIT
    CALL    ANSEL_CLEAR
    CALL    TRIS_INIT
    CALL    SPI_INIT
    CALL    PWM_INIT
    CALL    TMR_INIT
```

```
    BANKSEL PORTA
    CLRF   PORTA
    BANKSEL PORTB
    CLRF   PORTB
    BANKSEL PORTC
    CLRF   PORTC
    CLRF   SwitchVar ; Initialize SwitchVar to 0
    CLRF   DataByte
```

```

        CLRF      ERR
;        BSF      PORTC,2

;TEST TIMING
;Test      BTFSS   INTCON,T0IF      ; Check that TM0 Overflow flag is set;
;          Goto    Test
;          BCF     PORTC,2          ;
;          BCF     INTCON,T0IF
;Test2     BTFSS   INTCON,T0IF      ;
;          Goto    Test2          ;
;          BSF     PORTC,2
;          BCF     INTCON,T0IF
;          Goto    Test

;
; SS line has to be deasserted for at least 2ms between transfers

START_LOOP:
        BANKSEL  PORTC          ;
        BTFSS   PORTC,1        ; If Command is for battery status (pin state 0)
        CALL    BattStatus     ; Set DataByte for 1st Byte to send (0x3F)
        BTFSC   PORTC,1        ; If Command is for lamp (pin state 1)
        CALL    LampState      ; Set DataByte for 1st Byte to send (0xF0-F7)

START_SPEC_COMM:
        PAGESELW SwitchDecision
        MOVF    SwitchVar,W
        CALL    SwitchDecision ; Call Switch cases for SPECIAL Comm
        MOVF    SwitchVar,W
        XORLW   0x05           ; if SwitchVar is 4 (4th byte is sent/received)
        BTFSS   STATUS,Z       ;
        GOTO    START_SPEC_COMM ; If not done with communication, keep looping through
        cases
        BANKSEL SwitchVar      ;
        CLRF    SwitchVar      ; Reset SwitchVar back to 0
        BCF     INTCON,T0IF    ; Clear T0IF flag
        BANKSEL TMR0
        CLRF    TMR0           ; And restart TMR0 to insure SS is high for at least 2 ms
        GOTO    START_LOOP     ; Return to Start of Loop

;*****
;***** Switch Case for SPECIAL comm states *****
;*****
;
SwitchDecision:
        ADDWF   PCL,f
        GOTO    SendCommand
        GOTO    FirstByte
        GOTO    SecondByte
        GOTO    ThirdByte
        GOTO    FourthByte

```

```
SendCommand:
```

```

BTFSS      INTCON,T0IF      ; Check that TM0 Overflow flag is set
Goto        SendCommand      ;
BANKSEL    PORTC            ;
BCF        PORTC,6          ; RC6 Assert SS line
MOVWF      DataByte,W       ;
BANKSEL    SSPBUF           ; Bank 0
MOVWF      SSPBUF           ;

INCF      SwitchVar,f      ;
RETURN

```

FirstByte:

```

BANKSEL    SSPSTAT          ; Bank 1
BTFSS      SSPSTAT,BF       ; is buffer full? (receive complete)
RETURN      ; Return if buffer is not ready to be read (will continue
to loop)

;***Read Data***
BANKSEL    SSPBUF           ; Bank 0
MOVWF      SSPBUF, W        ; WREG reg = contents of SSPBUF

;***Transmit***
MOVLW      0x00             ;
MOVWF      SSPBUF           ;

INCF      SwitchVar,f      ; Increment SwitchVar
RETURN

```

SecondByte:

```

BANKSEL    SSPSTAT          ; Bank 1
BTFSS      SSPSTAT,BF       ; is buffer full? (receive complete)
RETURN      ; Return if buffer is not ready to be read (will continue
to loop)

;***Read Data***
BANKSEL    SSPBUF           ; Bank 0
MOVWF      SSPBUF, W        ; WREG reg = contents of SSPBUF
XORLW      0xFF             ; Check if received byte is correct
BTFSS      STATUS,Z         ;
GOTO        DEBUG_ERR        ; FOR NOW
;BSF        PORTB,5          ; Light error LED

;***Transmit***
MOVLW      0x00             ;
MOVWF      SSPBUF           ;

INCF      SwitchVar,f      ; Increment SwitchVar
RETURN

```

ThirdByte:

```

BANKSEL    SSPSTAT          ; Bank 1
BTFSS      SSPSTAT,BF       ; is buffer full? (receive complete)
RETURN      ; Return if buffer is not ready to be read (will continue
to loop)

;***Read Data***

```

```

BANKSEL    SSPBUF        ; Bank 0
MOVWF     SSPBUF, W     ; WREG reg = contents of SSPBUF
BANKSEL    PORTC        ;
BTFSS     PORTC,1       ; Check if Command is battery or lamp (1 = lamp, 0 = bat)
CALL      ChargeStat    ;

```

```

;***Transmit***

```

```

BANKSEL    SSPBUF
MOVLW     0x00          ;
MOVWF     SSPBUF        ; Transmit 0x00

```

```

INCF      SwitchVar,f   ; Increment SwitchVar

```

```

RETURN

```

FourthByte:

```

BANKSEL    SSPSTAT      ; Bank 1
BTFSS     SSPSTAT,BF    ; is buffer full? (receive complete)
RETURN     ; Return if buffer is not ready to be read (will continue
to loop)

```

```

;***Read Data***

```

```

BANKSEL    SSPBUF        ; Bank 0
MOVWF     SSPBUF, W     ; WREG reg = contents of SSPBUF
BANKSEL    PORTC
BTFSS     PORTC,1       ; If Command is for battery status
CALL      BattLastByte ;
BTFSC     PORTC,1       ; If Command is for lamp
CALL      LampLastByte ;

```

```

BSF       PORTC,6       ; Raise SS line

```

```

INCF      SwitchVar,f   ;

```

```

RETURN

```

```

; *****
; ***** Initialization Functions *****
; *****

```

OSCCON_INIT:

```

BANKSEL    OSCCON        ; Set int osc speed to 500 kHz (so PWM CCPR1L:CCP1CON<5:4>
does not exceed 10 bits)

```

```

BCF       OSCCON,IRCF2 ;

```

```

BSF       OSCCON,IRCF1 ;

```

```

BSF       OSCCON,IRCF0 ;

```

```

RETURN

```

ANSEL_CLEAR:

```

BANKSEL    ANSEL        ; Bank 2

```

```

CLRF     ANSEL          ; clear ansel bits

```

```

CLRF     ANSELH         ; clear ansel bits

```

```

RETURN

```

TRIS_INIT:

```

BANKSEL    TRISA        ;

```

```

MOVLW    TRISASetup    ;

```

```

MOVWF    TRISA         ;

```

```

MOVLW    TRISBSetup    ;

```

```

MOVWF    TRISB         ;

```

```

MOVLW    TRISCSetup    ;

```

```
MOVWF TRISC
```

```
RETURN
```

```
SPI_INIT:
```

```
BANKSEL SSPSTAT ; Bank 1
```

```
CLRF SSPSTAT ;
```

```
BANKSEL SSPCON ;
```

```
MOVLW SPI_Mode ;
```

```
MOVWF SSPCON ;
```

```
RETURN
```

```
TMR_INIT:
```

```
BANKSEL OPTION_REG ;
```

```
BCF OPTION_REG,T0CS ; Set T0CS to use internal instruction clock for TMO
```

```
BANKSEL T2CON ; Setup Timer 2 for PWM
```

```
MOVLW T2Setup ;
```

```
MOVWF T2CON ;
```

```
RETURN
```

```
PWM_INIT:
```

```
BANKSEL PR2 ;
```

```
MOVLW Period ;
```

```
MOVWF PR2 ;
```

```
BANKSEL CCP1CON ;
```

```
MOVLW CCPsetup ;
```

```
MOVWF CCP1CON ;
```

```
BSF CCP1CON,5 ; Initialize PWM to 50% (For testing)
```

```
BCF CCP1CON,4
```

```
MOVLW b'01100001'
```

```
MOVWF CCPR1L ; Set 8 MSBs of Duty Cycle
```

```
RETURN
```

```
; *****  
; ***** COMM Functions *****  
; *****
```

```
BattStatus:
```

```
MOVLW 0x3F ; 0011 1111 send byte to get Battery status
```

```
MOVWF DataByte ;
```

```
RETURN
```

```
LampState:
```

```
BANKSEL PORTA ;
```

```
MOVWF PORTA,W ; Copy PortA to WREG (State of bits 0-2)
```

```
ANDLW 0x07 ; 0000 0111 isolating bits 0-2 of PortA
```

```
IORLW 0xF0 ;
```

```
MOVWF DataByte ;
```

```
RETURN
```

```
BattLastByte:
```

```
MOVWF CharDischar ;
```

```
BTFSC STATUS,Z ;
```

```
BCF PORTC,0 ; Clear PortC0 for Discharging
```

```
XORLW 0xFE ;
```

```
BTFSC STATUS,Z ;
```

```
BSF PORTC,0 ; Set PortC0 for Charging
```

```
RETURN
```

LampLastByte:

```
;MOVF PORTA,W ;TEST
MOVWF LampStatus ;
ANDLW b'00000111' ; 0000 0111 clear all bits but 3 LSBs
BCF STATUS,C ;
RLF LampStatus,F ; shift Lamp status bits left twice
RLF LampStatus,F ;

MOVF PORTC,W ;
ANDLW b'11100011' ; 1110 0011 to maintain other bits
IORWF LampStatus,W ;
MOVWF PORTC ;
RETURN
```

ChargeStat:

```
;MOVLW d'0' ;TESTING
BANKSEL ChargeState ;
MOVWF ChargeState ;
SUBLW d'242' ; WREG = 242 - ChargeStatus
BTFSS STATUS,C ; Skip if ChargeStatus <= 242
GOTO HUNDREDPER ; Charge at 100%

MOVF ChargeState,W ;
SUBLW d'229' ; WREG = 229 - ChargeStatus
BTFSS STATUS,C ; Skip if ChargeStatus <= 229
GOTO NINTYFIVEPER ; Charge at 95%

MOVF ChargeState,W ;
SUBLW d'216' ; WREG = 216 - ChargeStatus
BTFSS STATUS,C ; Skip if ChargeStatus <= 216
GOTO NINTYPER ; Charge at 90%

MOVF ChargeState,W ;
SUBLW d'204' ; WREG = 204 - ChargeStatus
BTFSS STATUS,C ; Skip if ChargeStatus <= 204
GOTO EIGHTYFIVEPER ; Charge at 85%

MOVF ChargeState,W ;
SUBLW d'191' ; WREG = 191 - ChargeStatus
BTFSS STATUS,C ; Skip if ChargeStatus <= 191
GOTO EIGHTYPER ; Charge at 80%

MOVF ChargeState,W ;
SUBLW d'178' ; WREG = 178 - ChargeStatus
BTFSS STATUS,C ; Skip if ChargeStatus <= 178
GOTO SEVENTYFIVEPER ; Charge at 75%

MOVF ChargeState,W ;
SUBLW d'165' ; WREG = 165 - ChargeStatus
BTFSS STATUS,C ; Skip if ChargeStatus <= 165
GOTO SEVENTYPER ; Charge at 70%

MOVF ChargeState,W ;
SUBLW d'153' ; WREG = 153 - ChargeStatus
```

```

BTFSS STATUS,C ; Skip if ChargeStatus <= 153
GOTO SIXTYFIVEPER ; Charge at 65%

MOVF ChargeState,W ;
SUBLW d'140' ; WREG = 140 - ChargeStatus
BTFSS STATUS,C ; Skip if ChargeStatus <= 140
GOTO SIXTYPER ; Charge at 60%

MOVF ChargeState,W ;
SUBLW d'127' ; WREG = 127 - ChargeStatus
BTFSS STATUS,C ; Skip if ChargeStatus <= 127
GOTO FIFTYFIVEPER ; Charge at 55%

MOVF ChargeState,W ;
SUBLW d'114' ; WREG = 114 - ChargeStatus
BTFSS STATUS,C ; Skip if ChargeStatus <= 114
GOTO FIFTYPER ; Charge at 50%

MOVF ChargeState,W ;
SUBLW d'102' ; WREG = 102 - ChargeStatus
BTFSS STATUS,C ; Skip if ChargeStatus <= 102
GOTO FORTYFIVEPER ; Charge at 45%

MOVF ChargeState,W ;
SUBLW d'89' ; WREG = 89 - ChargeStatus
BTFSS STATUS,C ; Skip if ChargeStatus <= 89
GOTO FORTYPER ; Charge at 40%

MOVF ChargeState,W ;
SUBLW d'76' ; WREG = 76 - ChargeStatus
BTFSS STATUS,C ; Skip if ChargeStatus <= 76
GOTO THIRTYFIVEPER ; Charge at 35%

MOVF ChargeState,W ;
SUBLW d'63' ; WREG = 63 - ChargeStatus
BTFSS STATUS,C ; Skip if ChargeStatus <= 63
GOTO THIRTYPER ; Charge at 30%

MOVF ChargeState,W ;
SUBLW d'51' ; WREG = 51 - ChargeStatus
BTFSS STATUS,C ; Skip if ChargeStatus <= 51
GOTO TWENTYFIVEPER ; Charge at 25%

MOVF ChargeState,W ;
SUBLW d'38' ; WREG = 38 - ChargeStatus
BTFSS STATUS,C ; Skip if ChargeStatus <= 38
GOTO TWENTYPER ; Charge at 20%

MOVF ChargeState,W ;
SUBLW d'25' ; WREG = 25 - ChargeStatus
BTFSS STATUS,C ; Skip if ChargeStatus <= 25
GOTO FIFTEENPER ; Charge at 15%

MOVF ChargeState,W ;

```

```

SUBLW      d'12'           ; WREG = 12 - ChargeStatus
BTFSS      STATUS,C       ; Skip if ChargeStatus > 12
GOTO       TENPER         ; Charge at 10%

MOVWF      ChargeState,W  ;
SUBLW      0              ; WREG = 0 - ChargeStatus
BTFSS      STATUS,C       ; Skip if ChargeStatus <= 0
GOTO       FIVEPER        ; Charge at 5%

```

```
ZERO:        ; 00000010 00 - d'8' for 1% DC (so C32 does on capture strange period on IC)
```

```

BANKSEL    CCP1CON
BCF        CCP1CON,5      ; Set lower 2 LSBs of Duty Cycle
BCF        CCP1CON,4
MOVLW      b'00000010'
MOVWF      CCPR1L        ; Set 8 MSBs of Duty Cycle
RETURN

```

```
HUNDREDPER: ; 11000001 00 - d'772' for 99% DC (100% = '780')
```

```

BANKSEL    CCP1CON
BCF        CCP1CON,5      ; Set lower 2 LSBs of Duty Cycle
BCF        CCP1CON,4
MOVLW      b'11000001'
MOVWF      CCPR1L        ; Set 8 MSBs of Duty Cycle
RETURN

```

```
NINETYFIVEPER: ; 10111001 01 - d'741' for 95% DC
```

```

BANKSEL    CCP1CON
BCF        CCP1CON,5      ; Set lower 2 LSBs of Duty Cycle
BSF        CCP1CON,4
MOVLW      b'10111001'
MOVWF      CCPR1L        ; Set 8 MSBs of Duty Cycle
RETURN

```

```
NINTYPER:    ; 10101111 10 - d'702'
```

```

BANKSEL    CCP1CON
BSF        CCP1CON,5      ; Set lower 2 LSBs of Duty Cycle
BCF        CCP1CON,4
MOVLW      b'10101111'
MOVWF      CCPR1L        ; Set 8 MSBs of Duty Cycle
RETURN

```

```
EIGHTYFIVEPER: ; 10100101 11 - d'663' for 85% DC
```

```

BANKSEL    CCP1CON
BSF        CCP1CON,5      ; Set lower 2 LSBs of Duty Cycle
BSF        CCP1CON,4
MOVLW      b'10100101'
MOVWF      CCPR1L        ; Set 8 MSBs of Duty Cycle
RETURN

```

```
EIGHTYPER:   ; 10011100 00 - d'624'
```

```

BANKSEL    CCP1CON
BCF        CCP1CON,5      ; Set lower 2 LSBs of Duty Cycle
BCF        CCP1CON,4
MOVLW      b'10011100'
MOVWF      CCPR1L        ; Set 8 MSBs of Duty Cycle
RETURN

```

```
SEVENTYFIVEPER: ; 10010010 01 - d'585' for 75% DC
```

```

BANKSEL    CCP1CON
BCF        CCP1CON,5      ; Set lower 2 LSBs of Duty Cycle

```



```

    BSF          CCP1CON,4
    MOVLW       b'10010010'
    MOVWF      CCPR1L      ; Set 8 MSBs of Duty Cycle
    RETURN
SEVENTYPER:   ; 10001000 10 - d'546'
    BANKSEL    CCP1CON
    BSF        CCP1CON,5   ; Set lower 2 LSBs of Duty Cycle
    BCF        CCP1CON,4
    MOVLW     b'10001000'
    MOVWF     CCPR1L      ; Set 8 MSBs of Duty Cycle
    RETURN
SIXTYFIVEPER: ; 01111110 11 - d'507' for 65% DC
    BANKSEL    CCP1CON
    BSF        CCP1CON,5   ; Set lower 2 LSBs of Duty Cycle
    BCF        CCP1CON,4
    MOVLW     b'01111110'
    MOVWF     CCPR1L      ; Set 8 MSBs of Duty Cycle
    RETURN
SIXTYPER:     ; 01110101 00 - d'468'
    BANKSEL    CCP1CON
    BCF        CCP1CON,5   ; Set lower 2 LSBs of Duty Cycle
    BCF        CCP1CON,4
    MOVLW     b'01110101'
    MOVWF     CCPR1L      ; Set 8 MSBs of Duty Cycle
    RETURN
FIFTYFIVEPER: ; 01101011 01 - d'429' for 55% DC
    BANKSEL    CCP1CON
    BCF        CCP1CON,5   ; Set lower 2 LSBs of Duty Cycle
    BSF        CCP1CON,4
    MOVLW     b'01101011'
    MOVWF     CCPR1L      ; Set 8 MSBs of Duty Cycle
    RETURN
FIFTYPER:     ; 01100001 10 - d'390'
    BANKSEL    CCP1CON
    BSF        CCP1CON,5   ; Set lower 2 LSBs of Duty Cycle
    BCF        CCP1CON,4
    MOVLW     b'01100001'
    MOVWF     CCPR1L      ; Set 8 MSBs of Duty Cycle
    RETURN
FORTYFIVEPER: ; 01010111 11 - d'351' for 45% DC
    BANKSEL    CCP1CON
    BSF        CCP1CON,5   ; Set lower 2 LSBs of Duty Cycle
    BCF        CCP1CON,4
    MOVLW     b'01010111'
    MOVWF     CCPR1L      ; Set 8 MSBs of Duty Cycle
    RETURN
FORTYPER:     ; 01001110 00 - d'312'
    BANKSEL    CCP1CON
    BCF        CCP1CON,5   ; Set lower 2 LSBs of Duty Cycle
    BCF        CCP1CON,4
    MOVLW     b'01001110'
    MOVWF     CCPR1L      ; Set 8 MSBs of Duty Cycle
    RETURN
THIRTYFIVEPER: ; 01000100 01 - d'273' for 35% DC

```

```

    BANKSEL    CCP1CON
    BCF        CCP1CON,5    ; Set lower 2 LSBs of Duty Cycle
    BSF        CCP1CON,4
    MOVLW     b'01000100'
    MOVWF     CCPR1L        ; Set 8 MSBs of Duty Cycle
    RETURN

THIRTYPER:  ; 00111010 10 - d'234'
    BANKSEL    CCP1CON
    BCF        CCP1CON,5    ; Set lower 2 LSBs of Duty Cycle
    BSF        CCP1CON,4
    MOVLW     b'00111010'
    MOVWF     CCPR1L        ; Set 8 MSBs of Duty Cycle
    RETURN

TWENTYFIVEPER: ; 00110000 11 - d'195' for 25% DC
    BANKSEL    CCP1CON
    BCF        CCP1CON,5    ; Set lower 2 LSBs of Duty Cycle
    BSF        CCP1CON,4
    MOVLW     b'00110000'
    MOVWF     CCPR1L        ; Set 8 MSBs of Duty Cycle
    RETURN

TWENTYPER:   ; 00100111 00 - d'156'
    BANKSEL    CCP1CON
    BCF        CCP1CON,5    ; Set lower 2 LSBs of Duty Cycle
    BSF        CCP1CON,4
    MOVLW     b'00100111'
    MOVWF     CCPR1L        ; Set 8 MSBs of Duty Cycle
    RETURN

FIFTEENPER:  ; 00011101 01 - d'117' for 15% DC
    BANKSEL    CCP1CON
    BCF        CCP1CON,5    ; Set lower 2 LSBs of Duty Cycle
    BSF        CCP1CON,4
    MOVLW     b'00011101'
    MOVWF     CCPR1L        ; Set 8 MSBs of Duty Cycle
    RETURN

TENPER:      ; 00010011 10 - d'78'
    BANKSEL    CCP1CON
    BCF        CCP1CON,5    ; Set lower 2 LSBs of Duty Cycle
    BSF        CCP1CON,4
    MOVLW     b'00010011'
    MOVWF     CCPR1L        ; Set 8 MSBs of Duty Cycle
    RETURN

FIVEPER:     ; 00001001 11 - d'39' for 5% DC
    BANKSEL    CCP1CON
    BCF        CCP1CON,5    ; Set lower 2 LSBs of Duty Cycle
    BSF        CCP1CON,4
    MOVLW     b'00001001'
    MOVWF     CCPR1L        ; Set 8 MSBs of Duty Cycle
    RETURN

; *****
; ***** Other Functions *****
; *****

DEBUG_ERR:   MOVLW     0xFF
             MOVWF     ERR

```

END