```
;==========================================================================================
;
;    Filename:      NerfCannonLights.asm
;    Date:          4/4/2015
;    File Version:  1.0d1
;
;    Author:        David M. Flynn
;    Company:       Oxford V.U.E., Inc.
;    E-Mail:        dflynn@oxfordvue.com
;    Web Site:      http://www.oxfordvue.com/
;
;==========================================================================================
;    NerfCannonLights receives a pulse from the gun controller and
;      flashes the lights down the barrel.
;
;
;    History:
;
; 1.0d1   4/4/2015     First code. Copied from StepperTest.
;
;==========================================================================================
; Options
;
;==========================================================================================
;==========================================================================================
; What happens next:
;   At power up the system LED will blink and the barrel lights cycle once.
;
;
;==========================================================================================
;
;   Pin 1 (RA2/AN2) Address A2 (output)
;   Pin 2 (RA3/AN3) Enable Servos 0..7 (active low output)
;   Pin 3 (RA4/AN4) Enable Servos 8..15 (active low output)
;   Pin 4 (RA5/MCLR*) N.C.
;   Pin 5 (GND) Ground
;   Pin 6 (RB0) SW3/LED3 (Active Low Input/Output)
;   Pin 7 (RB1/AN11/SDA1) I2C Data
;   Pin 8 (RB2/AN10/RX) SW2/LED2 (Active Low Input/Output)
;   Pin 9 (RB3/CCP1) Pulse output for Servos 0..7
;
;   Pin 10 (RB4/AN8/SLC1) I2C Clock
;   Pin 11 (RB5/AN7)  SW1/LED1 (Active Low Input/Output)(System LED)
;   Pin 12 (RB6/AN5/CCP2) N.C.
;   Pin 13 (RB7/AN6) N.C.
;   Pin 14 (Vcc) +5 volts
;   Pin 15 (RA6) N.C.
;   Pin 16 (RA7/CCP2) Pulse output for Servos 0..7
;   Pin 17 (RA0) Address A0 (output)
;   Pin 18 (RA1) Address A1 (output)
;
;==========================================================================================
;
;
                       list                   p=16f1847,r=hex,W=0    ; list directive to def
                       nolist
                       include                p16f1847.inc           ; processor specific va
                       list
;
                       __CONFIG _CONFIG1,_FOSC_INTOSC & _WDTE_OFF & _MCLRE_OFF & _IESO_OFF
;
;
; INTOSC oscillator: I/O function on CLKIN pin
; WDT disabled
; PWRT disabled
; MCLR/VPP pin function is digital input
; Program memory code protection is disabled
; Data memory code protection is disabled
; Brown-out Reset enabled
; CLKOUT function is disabled. I/O or oscillator function on the CLKOUT pin
; Internal/External Switchover mode is disabled
; Fail-Safe Clock Monitor is enabled
;
                        __CONFIG _CONFIG2,_WRT_OFF & _PLLEN_OFF & _LVP_OFF
;
; Write protection off
; 4x PLL disabled
; Stack Overflow or Underflow will cause a Reset
; Brown-out Reset Voltage (Vbor), low trip point selected.
; Low-voltage programming enabled
;
; '__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.
;
                        constant                oldCode=0
                        constant                useRS232=0
;
#Define                 _C                      STATUS,C
#Define                 _Z                      STATUS,Z
;
CCPCON_Clr              EQU                     b'00001001'             ;Clear output on match
CCPCON_Set              EQU                     b'00001000'             ;Set output on match
kMidPulseWidth          EQU                     d'3000'                 ;1500uS
kMinPulseWidth          EQU                     d'1800'                 ;900uS
kMaxPulseWidth          EQU                     d'4200'                 ;2100uS
kServoDwellTime         EQU                     d'50000'                ;2.5mS/Channel
;============================================================================================
                        nolist
                        include                 F1847_Macros.inc
                        list
;
;    Port A bits
PortADDRBits            EQU                     b'01100000'
PortAValue              EQU                     b'00011000'
;
#Define                 Servo_A0                LATA,0                  ;Output
#Define                 Servo_A1                LATA,1                  ;Output
#Define                 Servo_A2                LATA,2                  ;Output
#Define                 Enable0_7               LATA,3                  ;Output
#Define                 Enable8_15              LATA,4                  ;Output
#Define                 RA5_In                  PORTA,5                 ;unused
#Define                 RA6_In                  PORTA,6                 ;unused
#Define                 RA7_Out                 PORTA,7                 ;CCP2 Output
;
Servo_AddrDataMask      EQU                     0xF8
;
;
;    Port B bits
PortBDDRBits            EQU                     b'11110111'             ;LEDs Out Others In
PortBValue              EQU                     b'00000000'
;
#Define                 SW3_In                  PORTB,0                 ;SW3/LED3
#Define                 RB1_In                  PORTB,1                 ;I2C Data
#Define                 SW2_In                  PORTB,2                 ;SW2/LED2
#Define                 RB3_In                  PORTB,2                 ;CCP1 Output
#Define                 RB4_In                  PORTB,4                 ;I2C Clock
#Define                 SW1_In                  PORTB,5                 ;SW1/LED1
```

```
#Define                RB6_In                  PORTB,6                ;N.C.
#Define                RB7_In                  PORTB,7                ;N.C.
LED1_Bit               EQU                     5                      ;LED1 (Active Low Outpu
LED2_Bit               EQU                     2                      ;LED2 (Active Low Outpu
LED3_Bit               EQU                     0                      ;LED3 (Active Low Outpu
#Define                LED1_Tris               TRISB,LED1_Bit         ;LED1 (Active Low Outpu
#Define                LED2_Tris               TRISB,LED2_Bit         ;LED2 (Active Low Outpu
#Define                LED3_Tris               TRISB,LED3_Bit         ;LED3 (Active Low Outpu
;
;
;=========================================================================================
;=========================================================================================
;
;Constants
All_In                 EQU                     0xFF
All_Out                EQU                     0x00
;
TMR0Val                EQU                     0xB2                   ;0xB2=100Hz, 0.000128S/
LEDTIME                EQU                     d'100'                 ;1.00 seconds
LEDErrorTime           EQU                     d'10'
kWDTime                EQU                     d'200'                 ;2 seconds
;
T1CON_Val              EQU                     b'00000001'            ;PreScale=1,Fosc/4,Time
TMR1L_Val              EQU                     0x3C                   ; -2500 = 2.5 mS, 400 s
TMR1H_Val              EQU                     0xF6
;
;TMR1L_Val             EQU                     0x1E                   ; -1250 = 1.25 mS, 800
;TMR1H_Val             EQU                     0xFB
;
;TMR1L_Val             EQU                     0x8F                   ; -625 = 0.625 mS, 1600
;TMR1H_Val             EQU                     0xFD
;
TXSTA_Value            EQU                     b'00100000'            ;8 bit, TX enabled, Asy
RCSTA_Value            EQU                     b'10010000'            ;RX enabled, 8 bit, Con
; 8MHz clock low speed (BRGH=0,BRG16=1)
Baud_300               EQU                     d'1666'                ;0.299, -0.02%
Baud_1200              EQU                     d'416'                 ;1.199, -0.08%
Baud_2400              EQU                     d'207'                 ;2.404, +0.16%
Baud_9600              EQU                     d'51'                  ;9.615, +0.16%
BaudRate               EQU                     Baud_9600
;
;
DebounceTime           EQU                     d'10'
;
;============================================================================================
;***** VARIABLE DEFINITIONS
; there are 256 bytes of ram, Bank0 0x20..0x7F, Bank1 0xA0..0xEF, Bank2 0x120..0x16F
; there are 256 bytes of EEPROM starting at 0x00 the EEPROM is not mapped into memory but
;  accessed through the EEADR and EEDATA registers
;============================================================================================
;  Bank0 Ram 020h-06Fh 80 Bytes
;
Bank0_Vars             udata                   0x20
;
LED_Time               RES                     1
lastc                  RES                     1                      ;part of tickcount timm
tickcount              RES                     1                      ;Timer tick count
;
StatLED_Time           RES                     1
Stat_Count             RES                     1
;
                       if useRS232
TXByte                 RES                     1                      ;Next byte to send
RXByte                 RES                     1                      ;Last byte received
```

```
WorkingRXByte          RES                    1
RS232Flags             RES                    1
#Define                DataSentFlag           RS232Flags,0
#Define                DataReceivedFlag       RS232Flags,1
                       endif
;
;
;
EEAddrTemp             RES                    1                      ;EEProm address to read
EEDataTemp             RES                    1                      ;Data to be writen to E
;
;
Timer1Lo               RES                    1                      ;1st 16 bit timer
Timer1Hi               RES                    1                      ; one second RX timeiou
;
Timer2Lo               RES                    1                      ;2nd 16 bit timer
Timer2Hi               RES                    1                      ;
;
Timer3Lo               RES                    1                      ;3rd 16 bit timer
Timer3Hi               RES                    1                      ;GP wait timer
;
Timer4Lo               RES                    1                      ;4th 16 bit timer
Timer4Hi               RES                    1                      ; debounce timer
;
;
SysFlags               RES                    1
#Define                SW1_Flag               SysFlags,0
#Define                SW2_Flag               SysFlags,1
#Define                SW3_Flag               SysFlags,2
#Define                LED2_Flag              SysFlags,3
#Define                LED3_Flag              SysFlags,4
#Define                ServoOff0_7            SysFlags,5             ;Set to disable all ser
#Define                ServoOff8_15           SysFlags,5             ;Set to disable all ser
;
;#Define               FirstRAMParam          MinSpdLo
;#Define               LastRAMParam           SysFlags
;
;
;===========================================================================================
;  Bank2 Ram 120h-16Fh 80 Bytes
;
; I2C Stuff is here
;Note: only upper 7 bits of address are used
I2C_ADDRESS            EQU                    0x30                   ; Slave address
RX_ELEMENTS            EQU                    .32                    ; number of allowable a
TX_ELEMENTS            EQU                    .8                     ; Status nibble for eac
I2C_TX_Init_Val        EQU                    0xAA                   ; value to load into tr
I2C_RX_Init_Val        EQU                    0xAA                   ; value to load into re
;
Bank2_Vars             udata                  0x120
I2C_ARRAY_TX           res                    RX_ELEMENTS            ; array to transmit to
I2C_ARRAY_RX           res                    TX_ELEMENTS            ; array to receive from
;
;===========================================================================================
;  Bank4 Ram 1A0h-1EFh 80 Bytes
;
Bank3_Vars             udata                  0x1A0
ServoMaxSpeed0_7       res                    .8                     ;0=no Accel, 1..255 cou
ServoMaxSpeed8_15      res                    .8
ServoAccelValue0_7     res                    .8                     ;1..8 counts/20mS squar
ServoAccelValue8_15    res                    .8
ServoCurSpeed0_7       res                    .8                     ;0=Stopped, MSb=Directi
ServoCurSpeed8_15      res                    .8
ServoActive0_15        res                    .8
```

```
;
;==========================================================================================
;  Bank4 Ram 220h-26Fh 80 Bytes
;
Bank4_Vars             udata                  0x220
CMDServoIDX0           res                    1
CMDServoIDX1           res                    1
CMDSigTime0_7          res                    .16                    ;Commanded position
CMDSigTime8_15         res                    .16
MinTime0_7             res                    .16                    ;Minimum pulse time (9
MinTime8_15            res                    .16                    ;Minimum pulse time (9
;
;==========================================================================================
;  Bank5 Ram 2A0h-2EFh 80 Bytes
;
Bank5_Vars             udata                  0x2A0
ServoIDX0              res                    1                      ;Index 0..7
ServoIDX1              res                    1
ServoFlags             res                    .8                     ;4 bits per servo
ValueSentFlag0_7       EQU                    0
ValueSentFlag8_15      EQU                    4
CalcdDwell             res                    1                      ;scratch var
CalcdDwellH            res                    1
SigOutTime0_7          res                    .16                    ;Current position
DwellTime0_7           res                    .16                    ;Next dwell time
SigOutTime8_15         res                    .16
DwellTime8_15          res                    .16
;
;==========================================================================================
;  Bank6 Ram 320h-26Fh 80 Bytes
;
Bank6_Vars             udata                  0x320
MaxTime0_7             res                    .16                    ;Maximum pulse time (2
MaxTime8_15            res                    .16                    ;Maximum pulse time (2
;
;==========================================================================================
;  Common Ram 70-7F same for all banks
;      except for ISR_W_Temp these are used for paramiter passing and temp vars
;==========================================================================================
;
                       cblock                 0x70
                       Param70
                       Param71
                       Param72
                       Param73
                       Param74
                       Param75
                       Param76
                       Param77
                       Param78
                       Param79
                       Param7A
                       Param7B
                       Param7C
                       Param7D
                       Param7E
                       Param7F
                       endc
;
#Define                INDEX_I2C              Param70                ;I2C Data Pointer
#Define                GFlags                 Param71
#Define                I2C_TXLocked           Param71,0              ; Set/cleared by ISR,
#Define                I2C_RXLocked           Param71,1              ; Set/cleared by ISR,
#Define                I2C_NewRXData          Param71,2              ; Set by ISR, The new
```

```
;
;=========================================================================================
;Conditions
HasISR                 EQU                    0x80                   ;used to enable interu
;
;=========================================================================================
;==========================================================================================
; ID Locations
;                      ORG                    0x2000
;                      DE                     '1','.','0','0'
;
;==========================================================================================
; EEPROM locations (NV-RAM) 0x00..0x7F (offsets)
                       cblock                 0x0000
;
nvMinSpdLo;            RES                    1                      ;0x1E; -1250 = 1.25 mS
nvMinSpdHi;            RES                    1                      ;0xFB
nvMaxSpdLo;            RES                    1                      ;0x8F; -625 = 0.625 mS
nvMaxSpdHi;            RES                    1                      ;0xFD
;
nvSysFlags;            RES                    1
                       endc
;
#Define                nvFirstParamByte       nvMinSpdLo
#Define                nvLastParamByte        nvSysFlags
;
;
;==========================================================================================
;==========================================================================================
;
;
                       ORG                    0x000                  ; processor reset vect
                       CLRF                   STATUS
                       CLRF                   PCLATH
                       goto                   start                  ; go to beginning of p
;
;==========================================================================================
; Interupt Service Routine
;
; we loop through the interupt service routing every 0.008192 seconds
;
;
                       ORG                    0x004                  ; interrupt vector loc
                       CLRF                   BSR                    ; bank0
;
;
                       btfss                  INTCON,T0IF
                       goto                   SystemBlink_end
;
                       movlw                  TMR0Val                ;256x39+16 cycles (10,
                       addwf                  TMR0,F                 ; reload TMR0 with -40
                       bcf                    INTCON,T0IF            ; reset interupt flag
;------------------
; These routines run 100 times per second
;------------------
;Decrement timers until they are zero
;
                       call                   DecTimer1              ;if timer 1 is not zer
                       call                   DecTimer2
                       call                   DecTimer3
                       call                   DecTimer4
;
;-----------------------------------------------------------------
; blink LEDs
```

```
                       MOVLW                  LOW TRISB
                       MOVWF                  FSR0L
                       MOVLW                  HIGH TRISB
                       MOVWF                  FSR0H
; All LEDs off
                       BSF                    INDF0,LED1_Bit
                       BSF                    INDF0,LED2_Bit
                       BSF                    INDF0,LED3_Bit
;
; Read SW's
                       BCF                    SW1_Flag
                       BCF                    SW2_Flag
                       BCF                    SW3_Flag
                       BTFSS                  SW1_In
                       BSF                    SW1_Flag
                       BTFSS                  SW2_In
                       BSF                    SW2_Flag
                       BTFSS                  SW3_In
                       BSF                    SW3_Flag
; Dec LED time
                       DECFSZ                 tickcount,F            ;Is it time?
                       GOTO                   SystemBlink_end        ; No, not yet
;
                       MOVF                   LED_Time,F
                       MOVWF                  tickcount
; Flash LEDs
                       BCF                    INDF0,LED1_Bit
                       BTFSC                  LED2_Flag
                       BCF                    INDF0,LED2_Bit
                       BTFSC                  LED3_Flag
                       BCF                    INDF0,LED3_Bit
;
;
SystemBlink_end
;
;=========================================================================================
IRQ_Servo1             MOVLB                  0                      ;Bank0
                       BTFSS                  PIR1,CCP1IF
                       GOTO                   IRQ_Servo1_End
;
                       BTFSS                  ServoOff0_7            ;Are we sending a puls
                       GOTO                   IRQ_Servo1_1           ; Yes
;
;Oops, how did we get here???
                       MOVLB                  0x05
                       CLRF                   CCP1CON
                       GOTO                   IRQ_Servo1_X
;
IRQ_Servo1_1           MOVLB                  0x05
                       BTFSC                  CCP1CON,CCP1M0         ;Set output on match?
                       GOTO                   IRQ_Servo1_OL          ; No
; An output just went high
;
                       LSLF                   ServoIDX0,W
                       ADDLW                  LOW SigOutTime0_7
                       MOVWF                  FSR0L
                       MOVLW                  HIGH SigOutTime0_7
                       MOVWF                  FSR0H
;
                       MOVIW                  FSR0++                 ;Put the pulse into th
                       ADDWF                  CCPR1L,F
                       MOVIW                  FSR0--
                       ADDWFC                 CCPR1H,F
                       MOVLW                  CCPCON_Clr             ;Clear output on match
                       MOVWF                  CCP1CON                ;CCP1 clr on match
;Calculate dwell time
                       MOVLW                  LOW kServoDwellTime
                       MOVWF                  CalcdDwell
                       MOVLW                  HIGH kServoDwellTime
                       MOVWF                  CalcdDwellH
                       MOVIW                  FSR0++                 ;SigOutTime0_7
                       SUBWF                  CalcdDwell,F
                       MOVIW                  FSR0++                 ;SigOutTime0_7+1
                       SUBWFB                 CalcdDwellH,F
;
                       LSLF                   ServoIDX0,W
                       ADDLW                  LOW DwellTime0_7
                       MOVWF                  FSR0L
                       MOVF                   CalcdDwell,W
                       MOVWI                  FSR0++
                       MOVF                   CalcdDwellH,W
                       MOVWI                  FSR0++
;Set Value Sent Flag
                       MOVF                   ServoIDX0,W
                       ADDLW                  LOW ServoFlags
                       MOVWF                  FSR0L
                       BSF                    INDF0,ValueSentFlag0_7
;
                       GOTO                   IRQ_Servo1_X
;
; output went low so this cycle is done
IRQ_Servo1_OL          LSLF                   ServoIDX0,W
                       ADDLW                  LOW DwellTime0_7
                       MOVWF                  FSR0L
                       MOVLW                  HIGH DwellTime0_7
                       MOVWF                  FSR0H
;
                       MOVIW                  FSR0++
                       ADDWF                  CCPR1L,F
                       MOVIW                  FSR0++
                       ADDWFC                 CCPR1H,F
;
                       MOVLW                  CCPCON_Set             ;Set output on match
                       MOVWF                  CCP1CON
                       INCF                   ServoIDX0,F
                       MOVLW                  0x07
                       ANDWF                  ServoIDX0,F
;
IRQ_Servo1_X           MOVLB                  0
                       BCF                    PIR1,CCP1IF
IRQ_Servo1_End:
;=========================================================================================
IRQ_Servo2             MOVLB                  0                      ;Bank0
                       BTFSS                  PIR2,CCP2IF
                       GOTO                   IRQ_Servo2_End
;
                       BTFSS                  ServoOff8_15           ;Are we sending a puls
                       GOTO                   IRQ_Servo2_1           ; Yes
;
;Oops, how did we get here???
                       MOVLB                  0x05
                       CLRF                   CCP2CON
                       GOTO                   IRQ_Servo2_X
;
IRQ_Servo2_1           MOVLB                  0x05
                       BTFSC                  CCP2CON,CCP2M0         ;Set output on match?
                       GOTO                   IRQ_Servo2_OL          ; No
; An output just went high
```

```
;
                       LSLF                   ServoIDX1,W
                       ADDLW                  LOW SigOutTime8_15
                       MOVWF                  FSR0L
                       MOVLW                  HIGH SigOutTime8_15
                       MOVWF                  FSR0H
;
                       MOVIW                  FSR0++                 ;Put the pulse into th
                       ADDWF                  CCPR2L,F
                       MOVIW                  FSR0--
                       ADDWFC                 CCPR2H,F
                       MOVLW                  CCPCON_Clr             ;Clear output on match
                       MOVWF                  CCP2CON                ;CCP1 clr on match
;Calculate dwell time
                       MOVLW                  LOW kServoDwellTime
                       MOVWF                  CalcdDwell
                       MOVLW                  HIGH kServoDwellTime
                       MOVWF                  CalcdDwellH
                       MOVIW                  FSR0++                 ;SigOutTime0_7
                       SUBWF                  CalcdDwell,F
                       MOVIW                  FSR0++                 ;SigOutTime0_7+1
                       SUBWFB                 CalcdDwellH,F
;
                       LSLF                   ServoIDX1,W
                       ADDLW                  LOW DwellTime8_15
                       MOVWF                  FSR0L
                       MOVF                   CalcdDwell,W
                       MOVWI                  FSR0++
                       MOVF                   CalcdDwellH,W
                       MOVWI                  FSR0++
;Set Value Sent Flag
                       MOVF                   ServoIDX1,W
                       ADDLW                  LOW ServoFlags
                       MOVWF                  FSR0L
                       BSF                    INDF0,ValueSentFlag8_15
;
                       GOTO                   IRQ_Servo2_X
;
; output went low so this cycle is done
IRQ_Servo2_OL          LSLF                   ServoIDX1,W
                       ADDLW                  LOW DwellTime8_15
                       MOVWF                  FSR0L
                       MOVLW                  HIGH DwellTime8_15
                       MOVWF                  FSR0H
;
                       MOVIW                  FSR0++
                       ADDWF                  CCPR2L,F
                       MOVIW                  FSR0++
                       ADDWFC                 CCPR2H,F
;
                       MOVLW                  CCPCON_Set             ;Set output on match
                       MOVWF                  CCP2CON
                       INCF                   ServoIDX1,F
                       MOVLW                  0x07
                       ANDWF                  ServoIDX1,F
;
IRQ_Servo2_X           MOVLB                  0
                       BCF                    PIR2,CCP2IF
IRQ_Servo2_End:
;=========================================================================================
;-----------------------------------------------------------------------------------------
; I2C Com
IRQ_4                  MOVLB                  0x00
                       btfss                  PIR1,SSP1IF            ; Is this a SSP interr
                       goto                   IRQ_4_End               ; if not, bus collisio
                       banksel                SSP1STAT
                       btfsc                  SSP1STAT,R_NOT_W        ; is it a master read:
                       goto                   I2C_READ                ; if so go here
                       goto                   I2C_WRITE               ; if not, go here
I2C_READ_Return:
I2C_WRITE_Return       movlb                  0x00
                       bcf                    PIR1,SSP1IF             ; clear the SSP interr
IRQ_4_End
;-----------------------------------------------------------------------------------------
; I2C Bus Collision
IRQ_5                  MOVLB                  0x00
                       btfss                  PIR2,BCL1IF
                       goto                   IRQ_5_End
                       banksel                SSPBUF
                       clrf                   SSPBUF                  ; clear the SSP buffer
                       movlb                  0x00                    ;banksel PIR2
                       bcf                    PIR2,BCL1IF             ; clear the SSP interr
                       banksel                SSPCON1
                       bsf                    SSPCON1,CKP             ; release clock stretc
                       movlb                  0x00
;
IRQ_5_End:
;
;--------------------------------------------------------------------
;
                       retfie                                         ; return from interrup
;
;
;============================================================================================
;============================================================================================
;
                       include                F1847_Common.inc
                       include                I2C_SLAVE.inc
;
;============================================================================================
;
start                  MOVLB                  0x01                    ; select bank 1
                       bsf                    OPTION_REG,NOT_WPUEN    ; disable pullups on p
                       bcf                    OPTION_REG,TMR0CS       ; TMR0 clock Fosc/4
                       bcf                    OPTION_REG,PSA          ; prescaler assigned t
                       bsf                    OPTION_REG,PS0          ;111 8mhz/4/256=7812.5
                       bsf                    OPTION_REG,PS1          ;101 8mhz/4/64=31250hz
                       bsf                    OPTION_REG,PS2
;
                       MOVLB                  0x01                    ; bank 1
                       MOVLW                  b'01110000'             ; 8 MHz
                       MOVWF                  OSCCON
                       movlw                  b'00010111'             ; WDT prescaler 1:6553
                       movwf                  WDTCON
;
                       MOVLB                  0x03                    ; bank 3
                       CLRF                   ANSELB                  ;Digital I/O
;
; setup timer 1 for 1uS/count
;
                       MOVLB                  0x00                    ; bank 0
                       bcf                    T1CON,TMR1CS0           ; Fosc/4 = 2Mhz
                       bcf                    T1CON,TMR1CS1
                       bsf                    T1CON,T1CKPS0           ; prescale /2
                       bcf                    T1CON,T1CKPS1
                       bsf                    T1CON,NOT_T1SYNC        ;not sync'ed
                       bsf                    T1CON,TMR1ON            ;always on
                       bcf                    T1GCON,TMR1GE
```

```
;
;
                       MOVLB                  0x00                   ;Bank 0
; setup data ports
                       movlw                  PortBValue
                       movwf                  PORTB                  ;init port B
                       movlw                  PortAValue
                       movwf                  PORTA
                       MOVLB                  0x01                   ; bank 1
                       movlw                  PortADDRBits
                       movwf                  TRISA
                       movlw                  PortBDDRBits           ;setup for programer
                       movwf                  TRISB
;
                       if useRS232
; setup serial I/O
                       MOVLW                  TXSTA_Value
                       MOVWF                  TXSTA
                       MOVLW                  BaudRate
                       MOVWF                  SPBRG
                       MOVLB                  0x00                   ; bank 0
                       MOVLW                  RCSTA_Value
                       MOVWF                  RCSTA
                       endif
;
                       CLRWDT
; clear memory to zero
                       CALL                   ClearRam
;-----------------------
; Setup CCP1 & CCP2 for compare
;
                       MOVLW                  LEDTIME
                       MOVWF                  LED_Time
;
                       CLRWDT
                       MOVLB                  0x00
                       call                   Init_I2C               ;setup I2C
;
                       bsf                    INTCON,PEIE            ; enable periferal int
                       bsf                    INTCON,T0IE            ; enable TMR0 interupt
                       bsf                    INTCON,GIE             ; enable interupts
;
;=========================================================================================
;=========================================================================================
;  Main Loop
;
;=========================================================================================
MainLoop               CLRWDT
;
                       CALL                   I2C_DataInturp
;
                       CALL                   I2C_DataSender
;
                       goto                   MainLoop
;
;=========================================================================================
;=========================================================================================
; Parse the incoming data and put it where it belongs
; Even byte: data type nibble, 4 MSb
; Odd byte: data
kServoPosCmd           EQU                    0x80                   ;Position Command CMDS
kServoMaxSpd           EQU                    0x90                   ;LSB is ServoMaxSpeed
kServoAccel            EQU                    0xA0                   ;LSB is ServoAccelValu
kServoON               EQU                    0xB0                   ;Set ServoActive
```

```
kServoOFF              EQU                    0xC0                    ;Clr ServoActive
kServoMinTime          EQU                    0xD0                    ;Minimum pulse time (9
kServoMaxTime          EQU                    0xE0                    ;Maximum pulse time (2
;
;
I2C_DataInturp         BTFSC                  I2C_RXLocked
                       RETURN
                       BTFSS                  I2C_NewRXData           ;Data is new?
                       RETURN                                         ; No
                       BCF                    I2C_NewRXData
                       CLRF                   Param79                 ;offset
I2C_DataInturp_L1      LOADFSR0               I2C_ARRAY_RX,Param79
                       MOVIW                  FSR0++
                       MOVWF                  Param78
                       ANDLW                  0xF0
                       MOVWF                  Param7A
                       MOVLW                  0x0F
                       ANDWF                  Param78,F
; *** kServoPosCmd ***
                       MOVF                   Param7A,W
                       SUBLW                  kServoPosCmd
                       SKPZ
                       GOTO                   I2C_DataInturp_1
                       LOADFSR1               CMDSigTime0_7,Param79
I2C_DI_Mov2            MOVF                   Param78,W
                       MOVWI                  FSR1++
                       MOVIW                  FSR0++
                       MOVWI                  FSR1++
                       GOTO                   I2C_DataInturp_Next
; *** kServoMaxSpd ***
I2C_DataInturp_1       MOVF                   Param7A,W
                       SUBLW                  kServoMaxSpd
                       SKPZ
                       GOTO                   I2C_DataInturp_2
                       LSRF                   Param79,W
                       LOADFSR1               ServoMaxSpeed0_7,WREG
I2C_DI_Mov1            MOVIW                  FSR0++
                       MOVWI                  FSR1++
                       GOTO                   I2C_DataInturp_Next
; *** kServoAccel ***
I2C_DataInturp_2       MOVF                   Param7A,W
                       SUBLW                  kServoAccel
                       SKPZ
                       GOTO                   I2C_DataInturp_3
                       LSRF                   Param79,W
                       LOADFSR1               ServoAccelValue0_7,WREG
                       GOTO                   I2C_DI_Mov1
; *** kServoON ***
I2C_DataInturp_3       MOVF                   Param7A,W
                       SUBLW                  kServoON
                       SKPZ
                       GOTO                   I2C_DataInturp_4

                       GOTO                   I2C_DataInturp_Next
; *** kServoOFF ***
I2C_DataInturp_4       MOVF                   Param7A,W
                       SUBLW                  kServoOFF
                       SKPZ
                       GOTO                   I2C_DataInturp_5

                       GOTO                   I2C_DataInturp_Next
; *** kServoMinTime ***
I2C_DataInturp_5       MOVF                   Param7A,W
                       SUBLW                  kServoMinTime
                       SKPZ
                       GOTO                   I2C_DataInturp_6
                       LOADFSR1               MinTime0_7,Param79
                       GOTO                   I2C_DI_Mov2
; *** kServoMaxTime ***
I2C_DataInturp_6       MOVF                   Param7A,W
                       SUBLW                  kServoMaxTime
                       SKPZ
                       LOADFSR1               MaxTime0_7,Param79
                       GOTO                   I2C_DI_Mov2
;
I2C_DataInturp_7:
I2C_DataInturp_Next    INCF                   Param79,F
                       INCF                   Param79,F
                       MOVLW                  .32
                       SUBWF                  Param79,W
                       SKPZ
                       GOTO                   I2C_DataInturp_L1
                       MOVLB                  0x00
                       RETURN
;
;==============================================================
;
I2C_DataSender         BTFSC                  I2C_TXLocked
                       RETURN
;
                       CLRF                   Param78
;                      BTFSS                  SW1BtnBit
;                      BSF                    Param78,0
;
                       CLRF                   Param79                ;offset
                       LOADFSR0               I2C_ARRAY_TX,Param79
                       MOVF                   Param78,W
                       MOVWF                  INDF0
;
                       RETURN
;=========================================================================================
;
;=========================================================================================
; 10mS Delay
; 8MHz = 500uS/instruction
;
Light_Delay_10mS       MOVLW                  .10
                       MOVWF                  Param78
;
Delay_W_mS             CLRF                   Param77
Light_Delay_L1         NOP
                       NOP
                       NOP
                       NOP
                       NOP
                       DECFSZ                 Param77,F
                       GOTO                   Light_Delay_L1         ;4uS/Loop
                       DECFSZ                 Param78,F
                       GOTO                   Light_Delay_L1         ;1025uS/Loop
;
;=========================================================================================
; Set CCP1/2 to go high in 0x100 clocks
;
StartServos            MOVLB                  0                      ;bank 0
                       BTFSS                  ServoOff0_7
                       RETURN
                       BCF                    ServoOff0_7
;
```

```
;                      CALL                   SetMiddlePosition
;                      CALL                   Copy7CToSig
;
                       MOVLW                  0x00                   ;start in 0x100 clocks
                       MOVWF                  TMR1L
                       MOVLW                  0xFF
                       MOVWF                  TMR1H
;
                       MOVLB                  0x05
                       CLRF                   CCPR1H
                       CLRF                   CCPR1L
                       MOVLW                  CCPCON_Set
                       MOVWF                  CCP1CON                ;go high on match
                       MOVLB                  0x00                   ;Bank 0
                       RETURN
;
; Don't disable interrupts if you don't need to...
SetMiddlePosition      MOVLW                  LOW kMidPulseWidth
                       MOVWF                  Param7C
                       MOVLW                  HIGH kMidPulseWidth
                       MOVWF                  Param7D
                       Return
;
;=========================================================================================
; ClampInt(Param7D:Param7C,kMinPulseWidth,kMaxPulseWidth)
;
; Entry: Param7D:Param7C
; Exit: Param7D:Param7C=ClampInt(Param7D:Param7C,kMinPulseWidth,kMaxPulseWidth)
;
ClampInt               MOVLW                  high kMaxPulseWidth
                       SUBWF                  Param7D,W              ;7D-kMaxPulseWidth
                       SKPNB                                         ;7D<Max?
                       GOTO                   ClampInt_1             ; Yes
                       SKPZ                                          ;7D=Max?
                       GOTO                   ClampInt_tooHigh       ; No, its greater.
                       MOVLW                  low kMaxPulseWidth     ; Yes, MSB was equal c
                       SUBWF                  Param7C,W              ;7C-kMaxPulseWidth
                       SKPNZ                                         ;=kMaxPulseWidth
                       RETURN                                        ;Yes
                       SKPB                                          ;7C<Max?
                       GOTO                   ClampInt_tooHigh       ; No
                       RETURN                                        ; Yes
;
ClampInt_1             MOVLW                  high kMinPulseWidth
                       SUBWF                  Param7D,W              ;7D-kMinPulseWidth
                       SKPNB                                         ;7D<Min?
                       GOTO                   ClampInt_tooLow        ; Yes
                       SKPZ                                          ;=Min?
                       RETURN                                        ; No, 7D>kMinPulseWidt
                       MOVLW                  low kMinPulseWidth     ; Yes, MSB is a match
                       SUBWF                  Param7C,W              ;7C-kMinPulseWidth
                       SKPB                                          ;7C>=Min?
                       RETURN                                        ; Yes
;
ClampInt_tooLow        MOVLW                  low kMinPulseWidth
                       MOVWF                  Param7C
                       MOVLW                  high kMinPulseWidth
                       MOVWF                  Param7D
                       RETURN
;
ClampInt_tooHigh       MOVLW                  low kMaxPulseWidth
                       MOVWF                  Param7C
                       MOVLW                  high kMaxPulseWidth
                       MOVWF                  Param7D
```

```
                        RETURN
;
                        if oldCode
;=============================================================================================
;========================================================================================
;
MoveTo78                MOVWF                   FSR0L
                        MOVF                    INDF0,W
                        MOVWF                   Param78
                        INCF                    FSR0L,F
                        MOVF                    INDF0,W
                        MOVWF                   Param79
                        RETURN
;
;========================================================================================
;
MoveTo7C                MOVWF                   FSR0L
                        MOVF                    INDF0,W
                        MOVWF                   Param7C
                        INCF                    FSR0L,F
                        MOVF                    INDF0,W
                        MOVWF                   Param7D
                        RETURN
;
;========================================================================================
;
Move78To7C              MOVF                    Param78,W
                        MOVWF                   Param7C
                        MOVF                    Param79,W
                        MOVWF                   Param7D
                        RETURN
;
;========================================================================================
;
MoveFrom7C              MOVWF                   FSR0L
                        MOVF                    Param7C,W
                        MOVWF                   INDF0
                        INCF                    FSR0L,F
                        MOVF                    Param7D,W
                        MOVWF                   INDF0
                        RETURN
;
;========================================================================================
; Less or Equal
;
; Entry: Param7D:Param7C, Param79:Param78
; Exit: Param77:0=Param7D:Param7C<=Param79:Param78
;
Param7D_LE_Param79      CLRF                    Param77                 ;default to >
                        MOVF                    Param79,W
                        SUBWF                   Param7D,W               ;Param7D-Param79
                        SKPNB                                           ;Param7D<Param79?
                        GOTO                    SetTrue                 ; Yes
                        SKPZ                                            ;Param7D>Param79?
                        RETURN                                          ; Yes
                        MOVF                    Param78,W               ; No, MSB is a match
                        SUBWF                   Param7C,W               ;Param7C-Param78
                        SKPNB                                           ;Param7C<Param78?
                        GOTO                    SetTrue                 ; Yes
                        SKPZ                                            ;LSBs then same?
                        RETURN                                          ; No
;
SetTrue                 BSF                     Param77,0
                        RETURN
```

```
;
;=====================================================================================
; Greater or Equal
;
; Entry: Param7D:Param7C, Param79:Param78
; Exit: Param77:0=Param7D:Param7C>=Param79:Param78
;
Param7D_GE_Param79     CLRF                   Param77                ;default to <
                       MOVF                   Param79,W
                       SUBWF                  Param7D,W              ;Param7D-Param79
                       SKPNB                                         ;Param7D<Param79?
                       RETURN                                        ; Yes
                       SKPZ                                          ;Param7D>Param79?
                       GOTO                   SetTrue                ; Yes
Param7D_GE_Param79_1   MOVF                   Param78,W              ; No, MSB is a match
                       SUBWF                  Param7C,W              ;Param7C-Param78
                       SKPNB                                         ;Param7C<Param78?
                       RETURN                                        ; Yes
                       GOTO                   SetTrue                ; No
;
;======================================================================================
;
EqualMin               CLRF                   Param77
                       MOVLW                  high kMinPulseWidth
                       SUBWF                  Param7D,W
                       SKPZ
                       RETURN
                       MOVLW                  low kMinPulseWidth
                       SUBWF                  Param7C,W
                       SKPNZ
                       BSF                    Param77,0
                       RETURN

;
Subtract1000           MOVLW                  low kMinPulseWidth
                       SUBWF                  Param7C,F
                       SUBBF                  Param7D,F
                       MOVLW                  high kMinPulseWidth
                       SUBWF                  Param7D,F
                       RETURN
;
Subtract1500           MOVLW                  low d'1500'
                       SUBWF                  Param7C,F
                       SUBBF                  Param7D,F
                       MOVLW                  high d'1500'
                       SUBWF                  Param7D,F
                       RETURN
;
X2                     CLRC
                       RLF                    Param7C,F
                       RLF                    Param7D,F
                       RETURN
;
Add1000                MOVLW                  low kMinPulseWidth
                       ADDWF                  Param7C,F
                       ADDCF                  Param7D,F
                       MOVLW                  high kMinPulseWidth
                       ADDWF                  Param7D,F
                       RETURN
;
                       endif
;=========================================================================================
;=========================================================================================
;
```

```
;
;
;
;
                        END
;

```