Dallas Personal Robotics Group
IR Detector

By: Darren Sawicz

Original Documentation By: Jeff Koenig

This document is an update reflecting the changes made to Jeff Koenig's Infrared detector circuit. The assembly code and circuit are unchanged but the printed circuit artwork and documentation have been upgraded to make assembly easier.

Theory of Operation

The PIC12C509 generates a 38 kHz square wave, which delivers current to the infrared LEDs. The infrared energy is then reflected off of an obstacle, if present, and is detected by the Sharp IR receiver.

The program does the following:

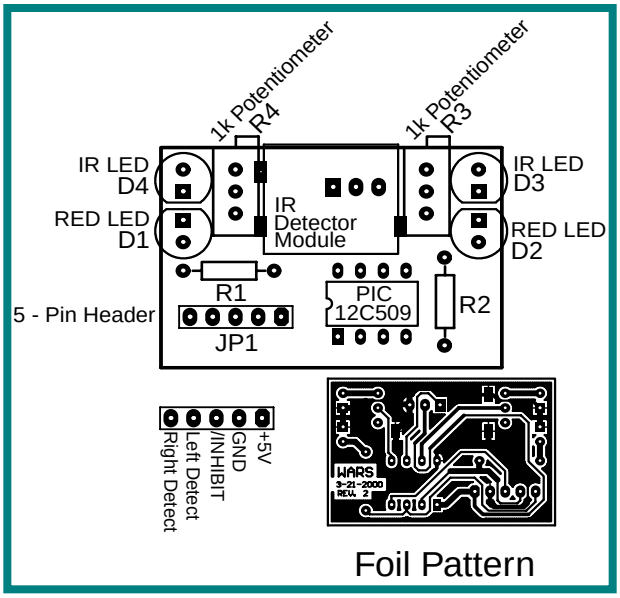
1. Flash the 38 kHz square wave through the Left infrared LED for 600 microseconds.
2. Check the Sharp detector – if it detects IR light, go to step 3. If not, go to step X.
3. Good! A reflection was detected. But, it might be from another source, so we will turn off the Left LED and see if the Sharp detector is still seeing infrared illumination (from another source).
4. Keep the Left infrared LED off for 600 microseconds.
5. Check the Sharp detector – there shouldn't be any IR detected, since we turned off the LED. If there is IR detected, go to step X,
6. A valid reflection was detected, so turn on the left LED (red) and bring header pin 4 high. Skip to step 8.
7. No valid detection was made.
8. Return to step one, but this time substitute "right" for "left"

The New Board

The new circuit board is smaller than Jeff Koenig's original board. Each new board measures 1.5" x 1". This allows 16 small boards to be cut from a single 4" x 6" board.

The original ASCII artwork showed none of the component connections, only component layout. I created a schematic drawing of the circuit to make it easier to see how the circuit is connected. The schematic is included on the last page.

The new board is laid out as follows:



Foil Pattern

The signals available at the header pins are:

1. **+5VDC** - Supply +5 VDC to this pin.
2. **GND** - Supply ground to this pin.
3. **/INHIBIT** - If you want the IRPROX unit to stop pulsing infrared light, use your micro to take this pin low (ground).
4. **LEFT DETECT** - Output from the IRPROX module. High (+5V) if reflection detected.
5. **RIGHT DETECT** - Output from the IRPROX module. High (+5V) if reflection detected.

Parts List

The parts list has been revised to include part numbers from distributors in the Winnipeg Area.

Part Description	*Qty.*	*Distributor Part Number*
IR Detector Board	1	Winnipeg Area Robotics Society
U1 - PIC 12C508 or 12C509	1	Active # 185-0084 Digi-Key # PIC12C508A-04/P-ND
JP1 - 5 x 1 Header	1	Active # 666-2758 Digi-Key # S1012-36-ND
R1 & R2 - 330 Ohm Resistor	2	Digi-Key # 330QBK-ND Active – 330 ohm
D1 & D2 - Red LED	2	Digi-Key # 67-1402-ND Active # 303-3033
R3 & R4 - 1k Potentiometer	2	Active # 468-7582
Sharp IR Module 36 kHz	1	Radio-Shack # 276-137
D3 & D4 - IR LED	2	Active # LD271 Digi-Key # QED423QT-ND

For More Information Please Visit: http://www.winnipegrobotics.com

Appendix A

Source Code

```
;
       TITLE 'Infra Red Proximity Detector - uses Sharp GP1U581Y'
;
       LIST P = 12C509, F = INHX8M
; P12C509.INC  Standard Header File, Version 1.02    Microchip Technology, Inc.
       NOLIST
;==========================================================
;       Verify Processor
;==========================================================
       IFNDEF __12C509
          MESSG "Processor-header file mismatch.  Verify selected processor."
       ENDIF
;==========================================================
;       Register Definitions
;==========================================================
W                           EQU     H'0000'
F                           EQU     H'0001'
;----- Register Files ------------------------------------------------
INDF                        EQU     H'0000' ; Uses FSR to address data mem.
TMR0                        EQU     H'0001' ; 8 bit real time clock/counter
PCL                         EQU     H'0002' ; Low order 8 bits of PC
STATUS                      EQU     H'0003' ; STATUS
FSR                         EQU     H'0004' ; Indirect data memory addr pointer
OSCCAL                      EQU     H'0005' ; Calibration data for osc.
GPIO                        EQU     H'0006' ; General Purpose I/O
;----- STATUS Bits ----------Page 14-----------------------------------
GPWUF                       EQU     H'0007' ; GPIO reset bit
PA0                         EQU     H'0005' ; Program Page preselect
NOT_TO                      EQU     H'0004' ; Time Out bit
NOT_PD                      EQU     H'0003' ; Power Down bit
ZERO                        EQU     H'0002' ; Zero bit
DC                          EQU     H'0001' ; Digit carry/*borrow bit
CARRY                       EQU     H'0000' ; carry/*borrow bit
;----- OPTION Bits ----------Page 15-----------------------------------
NOT_GPWU                    EQU     H'0007' ; Enable wake-up on pin change
NOT_GPPU                    EQU     H'0006' ; Enable weak pull-ups
T0CS                        EQU     H'0005' ; Timer0 clock source select
T0SE                        EQU     H'0004' ; Timer0 sources edge select
PSA                         EQU     H'0003' ; Prescalar assignment bit
PS2                         EQU     H'0002' ;\
PS1                         EQU     H'0001' ; > Prescalar rate select bits
PS0                         EQU     H'0000' ;/
;==========================================================
;       RAM Definition
;==========================================================
       __MAXRAM H'3F'
;==========================================================
;       Configuration Bits
;==========================================================
_MCLRE_ON                   EQU     H'0FFF'
_MCLRE_OFF                  EQU     H'0FEF'
_CP_ON                      EQU     H'0FF7'
_CP_OFF                     EQU     H'0FFF'
_WDT_ON                     EQU     H'0FFF'
_WDT_OFF                    EQU     H'0FFB'
_LP_OSC                     EQU     H'0FFC'
_XT_OSC                     EQU     H'0FFD'
_IntRC_OSC                  EQU     H'0FFE'
_ExtRC_OSC                  EQU     H'0FFF'
       __CONFIG      ( _MCLRE_OFF & _CP_OFF & _WDT_OFF & _IntRC_OSC )
;==========================================================
;       Program Variables
;==========================================================
```

```
       CONSTANT       DELAYTIME=D'150'      ; Timer value

#define LEFTDETECT    GPIO,5 ; Pin 2 = GP5 = Bit 5
#define RIGHTDETECT   GPIO,4 ; Pin 3 = GP4 = Bit 4
#define INHIBIT              GPIO,3  ; Pin 4 = GP3 = Bit 3  NOTE:  INPUT ONLY!
#define RIGHTLED      GPIO,2 ; Pin 5 = GP2 = Bit 2
#define IRDETECT      GPIO,1 ; Pin 6 = GP1 = Bit 1
#define LEFTLED              GPIO,0 ; Pin 7 = GP0 = Bit 0

              LIST
WAIT          EQU    9      ;  Location for counter for time delay loop

;=================================================================

;       Code Begins
;=================================================================

start
       ORG    H'00'
;      MOVLW  H'7F'          ; -- USED FOR OFFSET IN ERASABLE PARTS ---
       MOVWF  OSCCAL         ; Store the factory osc. calibration value
       MOVLW  B'00001010'    ; Set pins 4 as inputs, 2,3,5,6,7 as outputs
       TRIS   GPIO           ; Configure pins as either I or O
       MOVLW  B'00000000'    ; Set OPTION bits
       OPTION                ; Implement OPTION bits

       BCF    LEFTLED
       BCF    LEFTDETECT
       BCF    RIGHTLED
       BCF    RIGHTDETECT

main

noflash
       BTFSC  INHIBIT        ; Don't flash the IR Leds if the INHIBIT pin is low
       GOTO   do_left        ;
       BTFSC  IRDETECT       ; Check the IR detector
       GOTO   NF1
       GOTO   NF2

NF1    BCF    LEFTDETECT     ; No 38 KHz IR detected
       BCF    RIGHTDETECT    ; So turn on off right and left LED

NF2    BSF    LEFTDETECT     ; 38 KHz IR detected
       BSF    RIGHTDETECT    ; So turn on right and left LED

       GOTO   noflash        ;


do_left
       CALL   pulseleft      ; Pulse the left IR LED at 38 KHz for 600 microseconds
       BTFSC  IRDETECT       ; Read the Sharp Module. Skip next inst. if detection
(0=detect)
       GOTO   no_left        ; IR detector didn't sense reflection with Left LED on

       CALL   delayloop      ; Wait 600 microseconds
       BTFSS  IRDETECT       ; Read the Sharp Module. Skip next inst. if no detection
(0=detect)
       GOTO   no_left        ; Now check for an obstacle on the right side

       CALL   pulseleft      ; Pulse the left IR LED at 38 KHz for 600 microseconds
       BTFSC  IRDETECT       ; Read the Sharp Module. Skip next inst. if detection
(0=detect)
       GOTO   no_left        ; IR detector didn't sense reflection with Left LED on

       CALL   delayloop      ; Wait 600 microseconds
       BTFSS  IRDETECT       ; Read the Sharp Module. Skip next inst. if detection
(0=detect)
       GOTO   no_left        ; Now check for an obstacle on the right side

       CALL   pulseleft      ; Pulse the left IR LED at 38 KHz for 600 microseconds
```

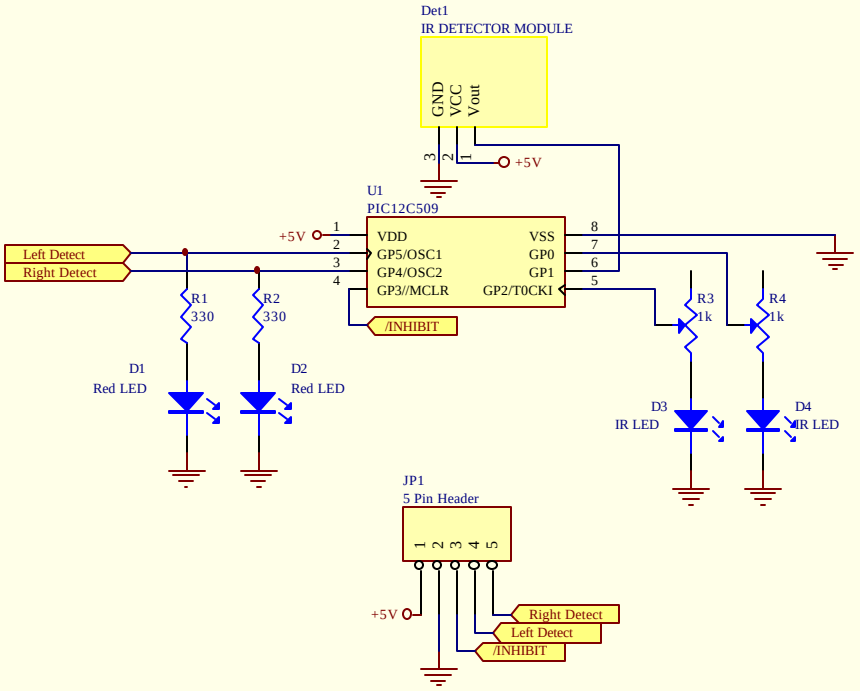
```
       BTFSC   IRDETECT       ; Read the Sharp Module. Skip next inst. if detection
(0=detect)
       GOTO    no_left        ; IR detector didn't sense reflection with Left LED on
       BSF     LEFTDETECT     ; Turn on the left Visible LED
       GOTO    do_right       ;

no_left                       ; No obstacle to the left, so
       BCF     LEFTDETECT     ; Turn off the left visible LED


do_right
       CALL    pulseright     ; Pulse the right IR LED at 38 KHz for 600 microseconds
       BTFSC   IRDETECT       ; Read the Sharp Module. Skip next inst. if detection
(0=detect)
       GOTO    no_right       ; IR detector didn't sense reflection with Right LED on

       CALL    delayloop      ; Wait 600 microseconds
       BTFSS   IRDETECT       ; Read the Sharp Module. Skip next inst. if no detection
(0=detect)
       GOTO    no_right       ; Now check for an obstacle on the left side

       CALL    pulseright     ; Pulse the right IR LED at 38 KHz for 600 microseconds
       BTFSC   IRDETECT       ; Read the Sharp Module. Skip next inst. if detection
(0=detect)
       GOTO    no_right       ; IR detector didn't sense reflection with Right LED on

       CALL    delayloop      ; Wait 600 microseconds
       BTFSS   IRDETECT       ; Read the Sharp Module. Skip next inst. if detection
(0=detect)
       GOTO    no_right       ; Now check for an obstacle on the left side

       CALL    pulseright     ; Pulse the right IR LED at 38 KHz for 600 microseconds
       BTFSC   IRDETECT       ; Read the Sharp Module. Skip next inst. if detection
(0=detect)
       GOTO    no_right       ; IR detector didn't sense reflection with Right LED on
       BSF     RIGHTDETECT    ; Turn on the left Visible LED

       GOTO    main

no_right                      ; No obstacle to the right, so
       BCF     RIGHTDETECT    ; Turn off the right visible LED
       GOTO    main


delayloop                     ;-----------------------
       MOVLW   D'195'         ;
       MOVWF   WAIT           ;  This creates a 600 microsecond delay
loop                          ;
       NOP                    ;
       DECFSZ  WAIT,F         ;
       goto    loop           ;
       RETLW   0              ;-----------------------


pulseleft                     ;-----------------------
       MOVLW   D'24'          ;
       MOVWF   WAIT           ;  Pulses the left IR led at 38 KHz
leloop BSF     LEFTLED        ;  for 600 microseconds
       NOP                    ;
       NOP                    ;
       NOP                    ;
       NOP                    ;
       NOP                    ;
       NOP                    ;
       NOP                    ;
       NOP                    ;
       NOP                    ;
       NOP                    ;
       NOP                    ;
       NOP                    ;
       BCF     LEFTLED        ;
       NOP                    ;
```

```
        NOP                 ;
        NOP                 ;
        NOP                 ;
        NOP                 ;
        NOP                 ;
        NOP                 ;
        NOP                 ;
        NOP                 ;
        NOP                 ;
        DECFSZ  WAIT,F      ;
        GOTO    leloop      ;
        RETLW   0           ;-----------------------

pulseright                  ;-----------------------
        MOVLW   D'24'       ;
        MOVWF   WAIT        ;  Pulses the right IR led
riloop  BSF     RIGHTLED    ;  for 600 microseconds
        NOP                 ;
        NOP                 ;
        NOP                 ;
        NOP                 ;
        NOP                 ;
        NOP                 ;
        NOP                 ;
        NOP                 ;
        NOP                 ;
        NOP                 ;
        BCF     RIGHTLED    ;
        NOP                 ;
        NOP                 ;
        NOP                 ;
        NOP                 ;
        NOP                 ;
        NOP                 ;
        NOP                 ;
        NOP                 ;
        NOP                 ;
        DECFSZ  WAIT,F      ;
        GOTO    riloop      ;
        RETLW   0           ;-----------------------

        end
```

APPENDIX B

Hex Listing

```
:1000000025000A0C0600000C02000604A6044604A3
:10001000860466061 30A26060E0A100AA60486043B
:10002000A6058605090A3E092606240A3809260778
:10003000240A3E092606240A38092607240A3E090E
:10004000260624 0AA605250AA6045A092606360A03
:1000500038092607360A5A092606360A38092607B5
:10006000360A5A092606360A8605090A8604090A46
:10007000C30C29000000E9023A0A0008180C290004
:1000800006050000000000000000000000000065
:10009000000000000000000060400000000000056
:1000A000000000000000000000000000E90265
:1000B000400A0008180C2900460500000000000056
:1000C000000000000000000000000000000030
:1000D000460400000000000000000000000D6
:0C00E000000000000000E9025C0A0008BB
:021FFE00EA0FE8
:00000001FF
```



Winnipeg Area Robotics Society

Title: *DPRG IR Proximity Detector*			Meetings at U of M Saint John's College Room 118 92 Dysart Road
Size: Letter	Number: WARS002	Revision: 0	
Drawn By: Darren Sawicz	Date: 21-Mar-2000	Time: 22:22:55	Sheet 1 of 1