



## Infra Red Monitor

Author: Mark Lemay  
Quebec, Canada  
email: MLemay@CollegeShawinigan.qc.ca

### APPLICATION OPERATION

The application of my IR monitor is to check an IR emitting device such as a TV Video remote controller. This IR monitor requires only 7 resistors, 4 LEDs, 1 miniature switch, 1 IR phototransistor and 1 PIC12C671 uC.

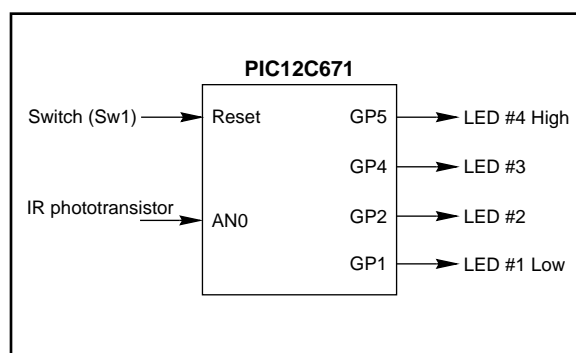
This circuit uses an analog converter from PIC12C671 to measure the infra-red intensity from the IR phototransistor. The intensity is displayed on the bargraph LEDs.

### OPERATION

1. Put an emitting device in front of the IR monitor (from 1 inch to 2 feet).
2. Press the switch on the IR monitor once to wake-up the microcontroller.
3. Press one key on the TV video controller and watch the 4 monitor LEDs. The LEDs blink if data is received, if all 4 LEDs stay off, no IR is sensing. If high IR power is received, more LEDs will be on. If IR monitor didn't receive anything for 17 seconds, it will turn off (sleep mode).

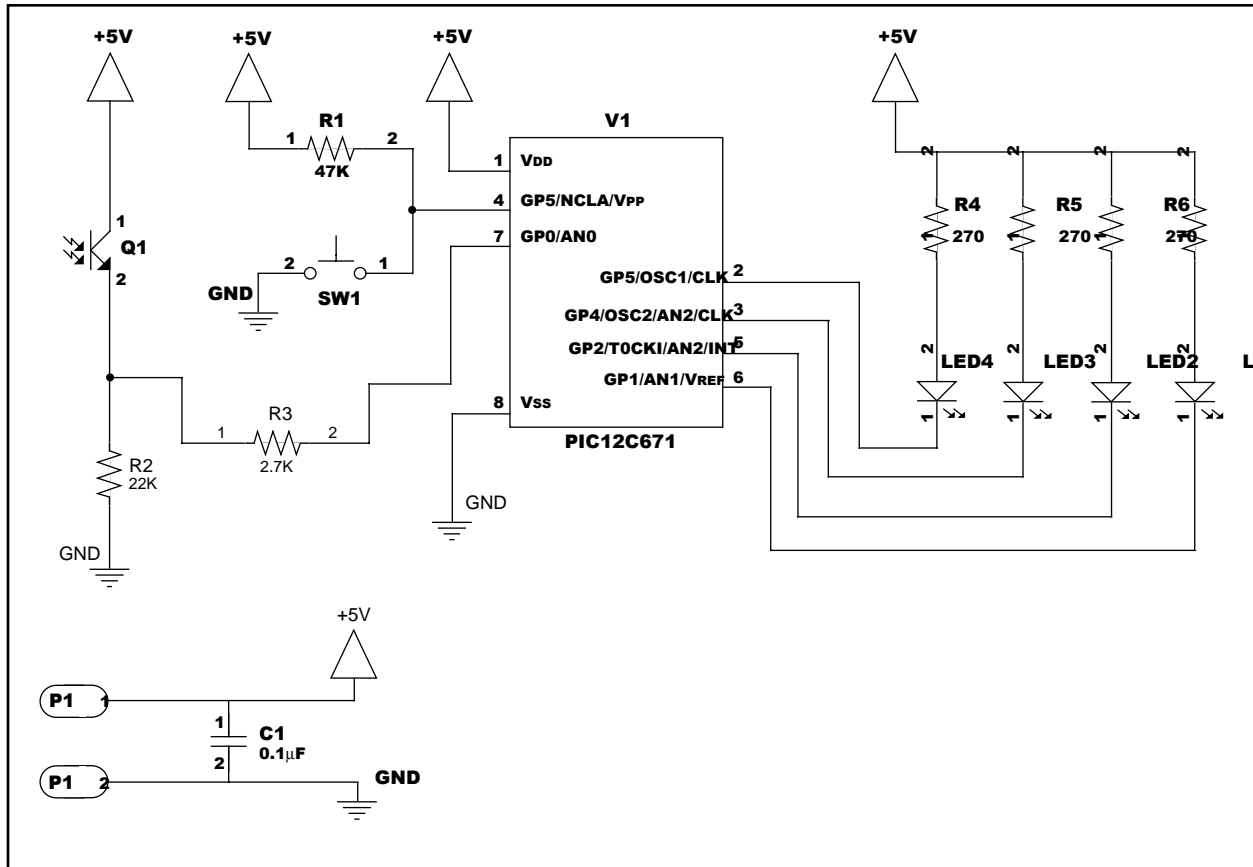
Note: The OPTEK IR phototransistor must be protected from daylight source to avoid false bargraph level.

### Block Diagram:



# Infra Red

## Graphical Hardware Representation



APPENDIX A: SOURCE CODE

MPASM 01.50 Released AD672.ASM 12-27-1997 17:23:34 PAGE 1

```

LOC OBJECT CODE      LINE SOURCE TEXT
VALUE
00001  ;*****
00002  ;
00003  ;   Marc Lemay
00004  ;   Electro Technician
00005  ;   221 St-Isidore
00006  ;   St-Etienne-des-Gres
00007  ;   Quebec, Canada
00008  ;   G0X 2P0
00009  ;
00010  ;   Tel: (819) 535-4117
00011  ;
00012  ;   Project: infra-red monitor with PIC12C672
00013  ;   date   : december 28 1997
00014  ;
00015  ;*****
00016
00017
Warning[217]: Hex file format specified on command line.
00018          list p=l2c672, f=inhx8m ;uC number
00019          ;and inhx8m output format file
00020
2007 3FE4      00021      __configH'3fe4'          ;master clear enable
00022                                     ;power-up timer enable
00023                                     ;oscillator = rc interne
00024
00000001      00025  f          equ      0x01
00000000      00026  w          equ      0x00
00000002      00027  z          equ      0x02          ;bit #2
00000005      00028  rp0       equ      0x05          ;bit #5
00000000      00029  adon      equ      0x00          ;bit #0
00000002      00030  go        equ      0x02          ;bit #2
00000002      00031  toif     equ      0x02          ;bit #2
00032
0000001E      00033  adres   equ      0x1e          ;a/d converter result
0000001F      00034  adcon0  equ      0x1f          ;ad converter
0000009F      00035  adcon1  equ      0x9f          ;input analog or digital register
00000005      00036  gpio    equ      0x05          ;adrs io
00000001      00037  tmr0    equ      0x01          ;adrs timer
00000003      00038  status  equ      0x03          ;status register adrs
0000008F      00039  osccal  equ      0x8f          ;oscillator calibration register
00000081      00040  optionreg equ      0x81
00000085      00041  trisreg equ      0x85
0000000B      00042  intcon  equ      0x0b          ;interrupt register
00043
00000020      00044  sleep1  equ      0x20          ;timer before entering sleep mode
00000021      00045  adresult equ      0x21
00000022      00046  lastvalue equ      0x22
00047
0000          00048      org          0000
00049          ;begin init
00050
0000          00051  begin
00052          ;movwf          osccal          ;save oscillator calibration value

0000 280A      00053          goto      begin2
00054
00055
000A          00056          org          0x0a
000A          00057  begin2
00058
000A 1683      00059          bsf          status,rp0;select bank 1

```

# Infra Red

MPASM 01.50 Released AD672.ASM 12-27-1997 17:23:34 PAGE 2

```
LOC OBJECT CODE      LINE SOURCE TEXT
VALUE
000B 3009 00060      movlw 0x09      ;gp1 gp2 gp4 gp5 output gp3 et gp0 input
Message[302]: Register in operand not in bank 0. Ensure that bank bits are correct.
000C 0085 00061      movwf trisreg   ;
000D 30D7 00062      movlw 0xd7      ;gp2 --> io port
Message[302]: Register in operand not in bank 0. Ensure that bank bits are correct.
000E 0081 00063      movwf optionreg ;prescale /256 au tmr0
000F 3006 00064      movlw 0x06      ;gp0 is analog Vref = Vdd
Message[302]: Register in operand not in bank 0. Ensure that bank bits are correct.
0010 009F 00065      movwf adcon1
0011 1283 00066      bcf  status,rp0 ;select bank 0
0012 110B 00067      bcf  intcon,toif ;clear tmr0 overflow flag
0013 3041 00068      movlw 0x41      ;
0014 009F 00069      movwf adcon0    ;clock/8 ch 0      ad on
00070      ;Tad = 2usec
0015 01A0 00071      clrfsleep1
0016 01A2 00072      clrflastvalue
00073      ;end init
00074
00075      ;begin master prog
0017      00076 master
0017 1D0B 00077      btffs intcon,toif ;check if tmr0 overflow
0018 281E 00078      goto nooverflow
00079
00080      ;here there is an overflow
00081
0019 110B 00082      bcf  intcon,toif ;clear bit toif
001A 0FA0 00083      incfsz sleep1,f ;inc f and skip next inst. if result is zero
00084      (overflow)
001B 281E 00084      goto nooverflow ;is result<>0 pas overflow
00085
00086      ;here the 16.67 sec have pass and we go in sleep mode
001C 01A0 00087      clrfsleep1
001D 2846 00088      goto sleepmode
00089
00090
001E      00091 nooverflow
001E 0000 00092      nop
00093      ;sampling time 19 usec wait time
001F 0000 00093      nop
0020 0000 00094      nop
0021 0000 00095      nop
0022 0000 00096      nop
0023 0000 00097      nop
0024 0000 00098      nop
0025 0000 00099      nop
0026 0000 00100     nop
0027 0000 00101     nop
0028 0000 00102     nop
0029 0000 00103     nop
002A 0000 00104     nop
002B 0000 00105     nop
002C 0000 00106     nop
002D 0000 00107     nop
002E 0000 00108     nop
002F 0000 00109     nop
0030 0000 00110     nop
00111
0031 151F 00112     bsf  adcon0,go   ;begin conversion
0032      00113 noend
0032 191F 00114     btffc adcon0,go
0033 2832 00115     goto noend
00116
00117      ;here the ad conversion is end
```

```

MPASM 01.50 Released          AD672.ASM   12-27-1997  17:23:34          PAGE   3
LOC  OBJECT CODE             LINE SOURCE TEXT
VALUE

```

```

00118          ;ir low = 255 and ir max = 00 in the adres register
00119          ;only the 4 high bits are kept
00120
0034 081E     00121      movf  adres,w          ;mov adres in w register
0035 39F0     00122      andlw 0xf0          ;kill 4 low bits
0036 00A1     00123      movwf adresult
0037 02A2     00124      subwf lastvalue,f    ;f - w --> f
0038 1D03     00125      btfss status,z       ;check if 2 value are equal
0039 01A0     00126      clrf  sleep1         ;sleep1 <-- 0
00127
003A 00A2     00128      movwf lastvalue      ;init the lastvalue with the last ad
                                conversion
003B 30FF     00129      movlw 0xff
003C 0085     00130      movwf gpio           ;shut off the 4 output leds
003D 1BA1     00131      btfsc adresult,0x07  ;test bit high high = low ir receive
003E 2850     00132      goto  ledx4          ;light on the 4 leds
003F 1B21     00133      btfsc adresult,0x06  ;test bit high
0040 2851     00134      goto  ledx3          ;light on the 3 low leds
0041 1AA1     00135      btfsc adresult,0x05  ;test bit high
0042 2852     00136      goto  ledx2          ;light on the 2 low leds
0043 1A21     00137      btfsc adresult,0x04  ;test bit high
0044 2853     00138      goto  ledx1          ;light on the low led
00139
0045 284F     00140      goto  fin            ;signal too low here all leds off
00141
00142
00143          ;here we enter the sleep mode
00144          ;all io are input and we stop ad converter
0046          00145      sleepmode
0046 1683     00146      bsf   status,rp0     ;select bank 1
0047 30FF     00147      movlw 0xff           ;gp1 gp2 gp4 gp5 gp3 et gp0 input
Message[302]: Register in operand not in bank 0. Ensure that bank bits are correct.
0048 0085     00148      movwf trisreg        ;
0049 1283     00149      bcf   status,rp0     ;select bank 0
004A 3040     00150      movlw 0x40           ;
004B 009F     00151      movwf adcon0         ;clock/8 ch 0 at ad off
                                ;Tad = 2usec
00152
004C 0063     00153      sleep
004D 0000     00154      nop
004E          0155      ici
004E 280A     00156      goto  begin2         ;never pass here (it is in sleep mode)
00157
004F          00158      fin
004F 2817     00159      goto  master
00160
0050 1285     00161      ledx4  bcf   gpio,5        ;high led on
0051 1205     00162      ledx3  bcf   gpio,4        ;led #3 on
0052 1105     00163      ledx2  bcf   gpio,2        ;led #2 on
0053 1085     00164      ledx1  bcf   gpio,1        ;low led on
00165
00166
0054 2817     00167      goto  master
00168
00169          end

```

# Infra Red

MPASM 01.50 Released AD672.ASM 12-27-1997 17:23:34 PAGE 5

## SYMBOL TABLE

LABEL	VALUE
__12C672	00000001
adcon0	0000001F
adcon1	0000009F
adon	00000000
adres	0000001E
adresult	00000021
begin	00000000
begin2	0000000A
f	00000001
fin	0000004F
go	00000002
gpio	00000005
ici	0000004E
intcon	0000000B
lastvalue	00000022
ledx1	00000053
ledx2	00000052
ledx3	00000051
ledx4	00000050
master	00000017
noend	00000032
nooverflow	0000001E
optionreg	00000081
osccal	0000008F
rp0	00000005
sleep1	00000020
sleepmode	00000046
status	00000003
tmr0	00000001
toif	00000002
trisreg	00000085
w	00000000
z	00000002

## MEMORY USAGE MAP ('X' = Used, '-' = Unused)

```
0000 : X-----XXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX
0040 : XXXXXXXXXXXXXXXXXXXX XXXXX-----
2000 : -----X-----
```

All other memory blocks unused.

Program Memory Words Used: 76

Program Memory Words Free: 1972

Errors : 0

Warnings : 1 reported, 0 suppressed

Messages : 4 reported, 0 suppressed