Light Meter

Author: B. M. Dhananjaya
Karnataka, India
email: teekays@pentagon.com

APPLICATION OPERATION

This design displays irradiance of the light source (10 $\mu W/cm^2$ to 1000 $\mu W/cm^2$) on the computer with a serial port (with very little modification a serial LCD display module can also be used).

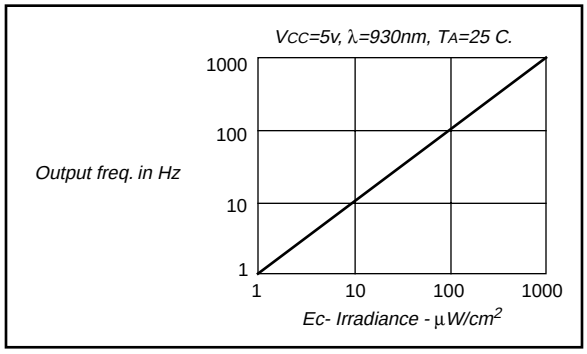
The circuit is designed keeping in view to make it the smallest light sensor in the world (using SOIC packages).

TSL220 from Texas Instruments is an integrated light-to-frequency converter. The frequency of whose digital output depends on the quantity of the light falling on it. TSL220 comprises a large photo diode and a complete current-to-frequency converter. The combination of these allows light to be converted directly into the digital signal of variable frequency. The relationship between the irradiance of the light and the output frequency is shown below.

The heart of the **"Light Meter"** is PIC12C508 that powers the sensor and measures the frequency. The measured frequency is converted to light intensity and transmitted to the PC at the rate of 2400 baud. The TSL220 is powered through one of the GPIO pins to reduce power consumption (TSL220 consumes 7 to 8mA of current). PIC12C508 has one 8-bit timer which is to be used here as a frequency counter, which reads the frequency from 5 Hz to 1250 Hz is implemented in this application. The timer is configured to measure the input frequency at TOCKI pin of PIC12C508. The input frequency is gated for a precise duration of time. The precise gate is implemented in software as an accurate delay.

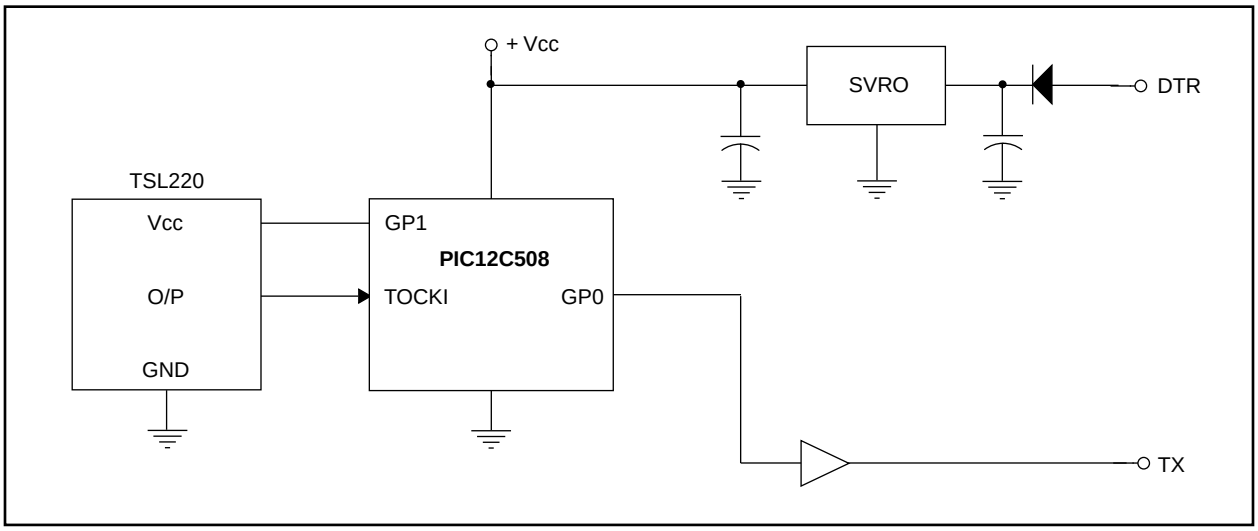
To keep the circuit simple and small the internal 4MHz clock is used with satisfying accuracy. In case of very high accuracy measurements an external oscillator is recommended.

The frequency is converted to the intensity of light and transmitted to the computer which displays the irradiance of light with suitable software on the computer.

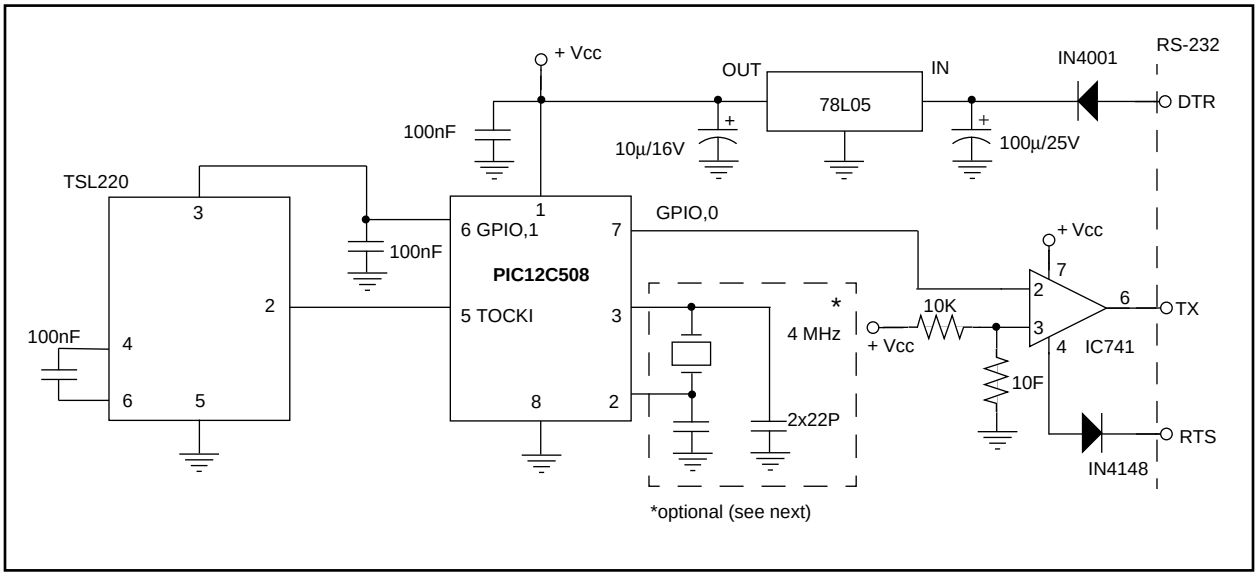


Sensor Interface

Block Diagram:



Schematic Diagram of Light Meter:



APPENDIX A: SOURCE CODE

```
            title  “Light Meter”
            list p=12c508
            include “p12c508.inc”
;*****************************************************************
;           Designed by B.M.Dhananjaya
  config _MCLRE ON&_CP_ON&_WDT OFF&_IntRC_OSC
;*****************************************************************
clockrate   equ        .4000000
baudrate    equ        .2400
fclk        equ        clockrate/4
baudconst   equ        ( (fclk/baudrate) /3-2)
mulplr      equ        .5
;*****************************************************************
cblock      0x08
            txreg,count
            delay,temp
            mulcnd,h byte
            1 byte
endc
;
            org        0
            movlw      b’11111000’    ;transition on TOCKI pin
            option
            movlw      b’00111100’    ;gpio<5:2>inputs
            tris       gpio           ;gpio<1:0>outputs
;
start
            call       delay200ms
            clrf       tmr0           ;clear timer
            bsf        gpio,1         ;power up sensor
            call       delay200ms     ;count for 200ms
            bcf        gpio,1         ;power down sensor
            movf       tmr0,w
            movwf      mulcnd
            call       multiply       ;multiply with constant to give the reading
                                       in mW/cm2
            movf       1 byte,w
            movwf      txreg          ;send low byte of
            call       transmit       ;16 bit value to pc
            movf       h byte,w       ;send high byte of
            movwf      txreg          ;16 bit value to pc
            call       transmit
            goto       start          ;repeat

;*****************************************************************
;           This sub-routine, send a byte to
;           the pc at 2400 bauds (8N1)
;*****************************************************************
transmit
            bcf        gpio,0         ;send start bit
            movlw      baudconst
            movf       delay
            movlw      .9
            movwf      count
tx_wait
            decfsz     delay          ;delay one bit
            goto       tx_wait
            movlw      baudconst
            movwf      delay
            decfsz     count          ;all bit transmitted?
            goto       next bit       ;no, then repeat
            movlw      .9
            movwf      count
            bsf        gpio,0         ;send stop bit
```

```
                return
next bit
                rrf         txreg
                btfss       status,c
                goto        setlo
                bsf         gpio,0
                goto        tx wait
setlo
                bcf         gpio,0
                goto        tx wait
;*****************************************************************
;               This routine gives a delay of 200ms
;               with 4Mhz clock
;*****************************************************************

delay1mS
                movlw       .197
                movwf       count
                nop
                goto        S+1         ;delay 1mS
                goto        S+1
dly1mS
                goto        S+1
                decfsz      count, F
                goto        dly1mS
                retlw       0
;
delay200mS
                movlw       .200
                movwf       temp
dly200mS
                call        delay1mS
                decfsz      temp, F
                goto        dly200mS
                movlw       .64
                movwf       count       ;delay 200x1ms=200ms
loop200mS
                decfsz      count, F
                goto        loop200mS
                retlw       0
;*****************************************************************
;               This routine multiplies register mulcnd & mulplr
;               and stores the 16 bit value in h_byte & 1 byte
;*****************************************************************
multiply
                clrf        h byte
                clrf        1 byte
                movlw       .8
                movwf       count
                movf        mulcnd,w
                bcf         status,c
loop            rrf         mulplr
                btfsc       status,c
                addwf       h_byte,f
                rrf         h_byte,f
                rrf         h_byte,f
                decfsz      count
                goto        loop
                retlw       0
;*****************************************************************

                end
```