Device Usage Monitor using AT89C2051

S.R. Jadhav, G.B.Waghmare, S.S.Nirmale, Dr. P.A.Kadam, Dr. R.R.Mudholkar
Department of Electronics, Shivaji University, Kolhapur, Maharashtra, 416004 India
e-mail: pak_eln@unishivaji.ac.in

Abstract:

Manufacturers provide usage guidelines for critical devices in working hours. The device has been guaranteed to work within these usage limits. It is very important that the specification closely matches with the actual device performance. Otherwise the device finds limited applicability, reliability and thus fails to meet the application prospects.

The system described in this article monitors the device usage using optical sensor[1]. In this work LDR[2] has been selected as a sensor to monitor usage of the LCD monitor. These readings have been useful for comparing the performance of LCD monitor. The system has been designed around the ATMEL microcontroller AT89C2051, the on-chip analog comparator[3] of which reduces need of external components. The 16*2 LCD display shows the elapsed time and total usage time.

Keywords: Embedded systems, 8051, device usage monitor, LDR

1.1 Introduction:-

Embedded systems are getting more and more popular with their integration in large number of applications. Broadly speaking today every application has got at least one embedded component integrated inside it e.g. consumer electronics, automobiles, industrial automation, networking etc. The major benefits of using embed system improved reliability, performance, efficiency, scalability and low cost. Such systems can be found very useful in many applications where direct Human intervention is not possible or at least not necessary.

The manufacturers of time critical appliances specify device usage specification in hours. In actual practice the device may fail to meet the specification provided by the manufacturer, which may severely affect the user application. Hence some practical solution for this situation is required. This paper describes design of device monitoring system using microcontroller AT89C2051, which senses the device status using LDR[2] sensor and displays the usage status on 16*2 LCD display. Thus this system has been found useful for the devices which give visual indication when they are turned ON. The unique advantage of the system is that the system is totally isolated from the device to be monitored and thus it is free from any type of switching disturbances generated by the device. The system uses very few components, thanks to the on-chip analog comparator feature provided by AT89C2051[3]; this provides higher reliability and usability with battery power.

In this study, different devices such as display monitor, microwave oven have been monitored for their usage time. By simply tuning LDR signal conditioning system block for differing intensities the system adjusts itself with different appliances. The output of LDR has been compared to the reference voltage by on-chip analog comparator. The reference voltage decides the intensity level at which switching should occur. The

microcontroller using polling mode reads the comparator output and counts accordingly using on-chip timer. The time duration has been displayed on the 16*2 LCD.

1.2 System block diagram-:

The system block diagram is shown in Fig. 1. It consists of LDR as a sensor, a microcontroller for controlling systems activities, a LCD for local real-time display of on time of display. The individual blocks are described Below.
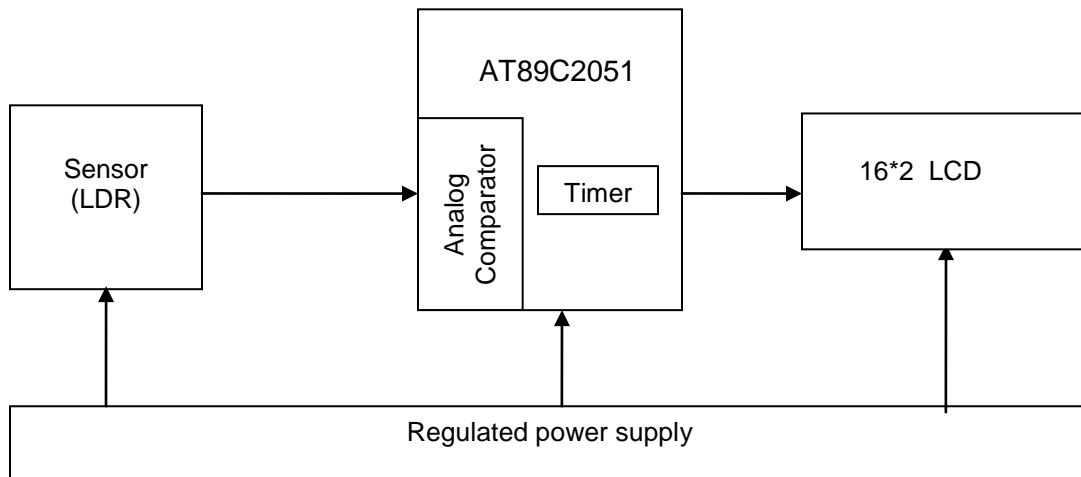

Fig. 1. System Block Diagram.

The sensor unit monitors light intensity with the help LDR, which has been connected in a voltage divider circuit. Sensor produces linear output within the specified range of maximum and minimum light intensities.

The output from the voltage divider has been connected to the analog comparator of the microcontroller. The port pins P1.0 and P1.1 act as a non-inverting and inverting inputs respectively and P3.6 as an output of an analog comparator. The reference level has been applied using a 10k preset to the inverting input, which varies in the range of 0-5V. The reference level is adjusted to detect the minimum level of the incident light. When the LDR receives the light, its resistance decreases. This increases the potential at the non-inverting input of analog comparator and the comparator switches the output.

The microcontroller has been clocked at 12MHz for accurate time calculations using on-chip timer. The timer forms the basis of time calculations by providing ticks (interrupts) at regular intervals. The timer configuration provides 50 msec period which has been multiplied by soft-timer to get 1 sec count. Similarly the microcontroller calculates minutes and hours and displays the usage time onto the LCD display.

The detailed schematic diagram of the system is shown in fig.2.
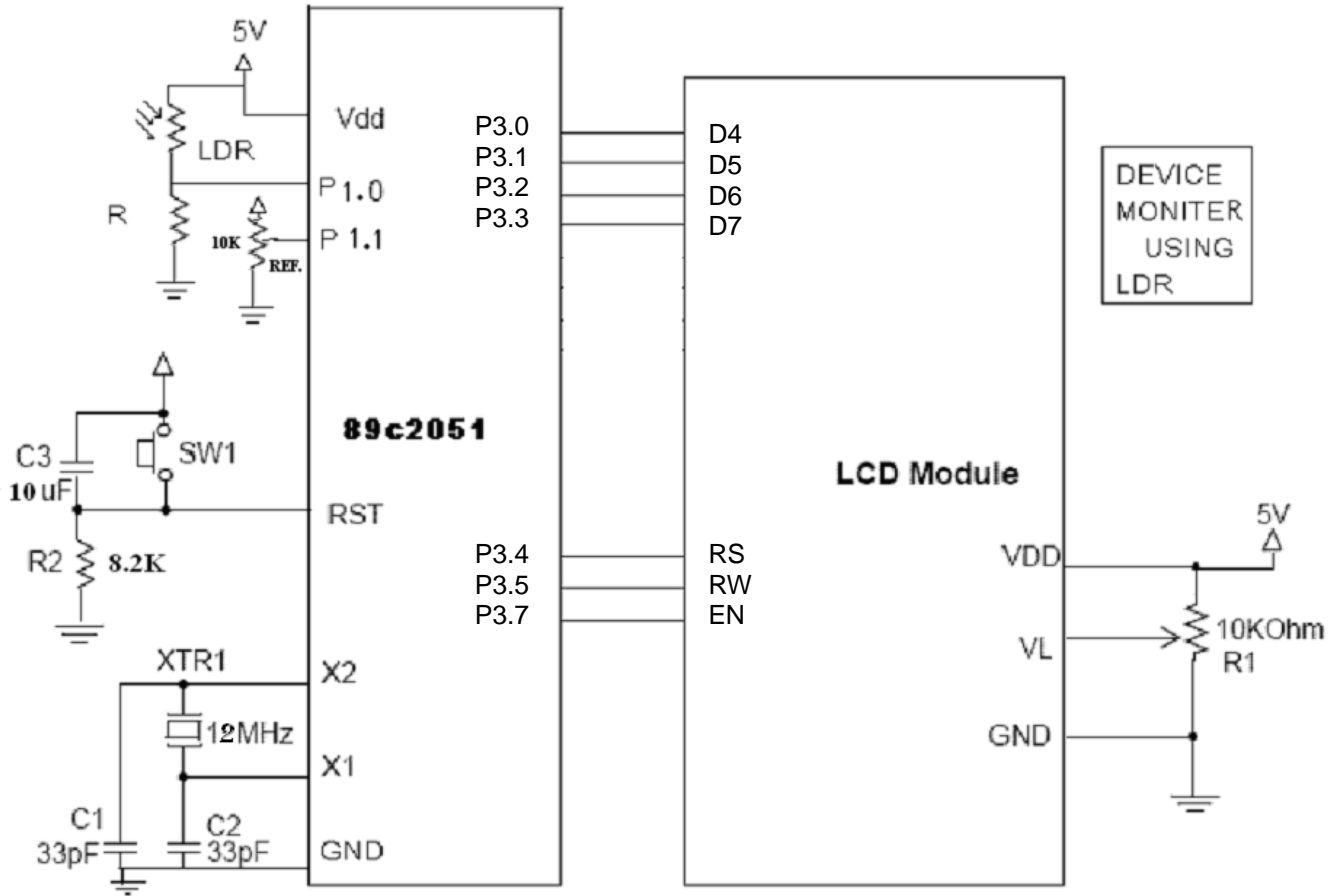
1.3 Circuit Diagram:

Fig. 2. Schematic of System. (IC1) AT89C2051, LCD 16*2.

1.4: Theoretical Aspects:-

1.4.1:Microcontroller AT89C2051: [3]
        The Port 1 is an 8-bit bi-directional I/O port. P1.0 and P1.1 require external pull-ups. P1.0 and P1.1 also serve as the positive input (AIN0) and the negative input (AIN1), respectively, the pin P3.6 is reserved for comparator output, this pin is not available at port 3 for any external application, and only available inernally. When 1 is written to Port 1 pins, they can be used as inputs.
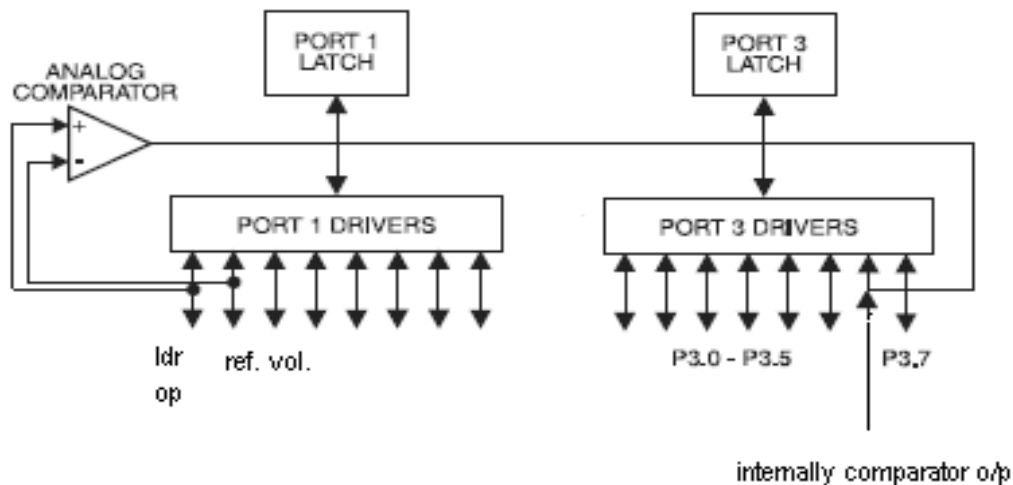
The comparator action is shown in fig. 3.

Fig. 3 Internal Comparator of IC 89C2051.

1.4.2 Sensor :
Light Dependent Resistor – LDR[2]

Two cadmium sulphide(cds) photoconductive cells with spectral responses similar to that of the human eye. The cell resistance falls with increasing light intensity.
LDR specifications-:
Operating Temperature -60 - +75 Deg. C.
Power Dissipation 100 mW max.
Maximum Current - - 75 mA max.
Maximum Voltage applied AC/DC Peak - - 320 V max.
Dark Resistance - - 1 - M Ohm.
Light resistance – 1 ohm.
Dark Capacitance - - 3.5 – pF.
It has high sensitivity.

        A photo resistor is a sensor whose resistance varies with light intensity. Most decrease in resistance as the light intensity increases. In a typical microcontroller

application, this resistance must be converted to a voltage so that an A-D converter can measure it. The easiest way to do this is with a voltage divider circuit.

A voltage divider is just two resistors in series connected between a voltage supply and ground as shown in fig 4. If R1 is connected to the voltage supply and R2 is connected to ground then the voltage at the junction between the two resistors is:

$$V = V_{CC} * \frac{R_2}{R_1 + R_2}$$

If R1 is the photo resistor, the voltage will increase with increasing light intensity. If R2 is the photo resistor, the voltage will decrease with increasing light intensity. [2]
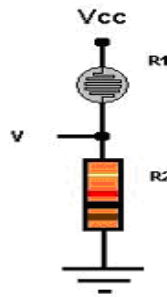

Fig 4. LDR connections:

1.5: Software Description:-

On power-up the initialization part of the system software configures various on chip peripherals such as timers, interrupts, LCD, comparator etc. and polls for the sensor output. The flowchart of the software has been shown in fig 5 and program code has been given in the listing 1. The microcontroller starts counting when it reads comparator output as LOW.[4]

For real time processing the timer-0 of AT89C2051 has been used. It has been configured in 16 bit timer mode, timer high(TH0) and low(TL0) counts are loaded for providing 50msec delay; which multiplied by 20, in timer ISR, gives accurate 1 sec Delay. The timer ISR further multiplies the 1 sec delay to get minutes, hours. The 16*2 LCD display shows time in the hh:mm:ss format.

The counting proceeds until the device to be monitored is ON. When the device turned OFF, the LDR receives less light (ambient light if not properly shielded). This increases resistance of LDR and lowers potential at the non-inverting input of analog comparator. The comparator again toggles the output, turning off the timer. Microcontroller then displays the time elapsed.

Table 1 gives LCD control signals and commands used in the program. Table 2 gives program structure and table 3 shows port configurations.
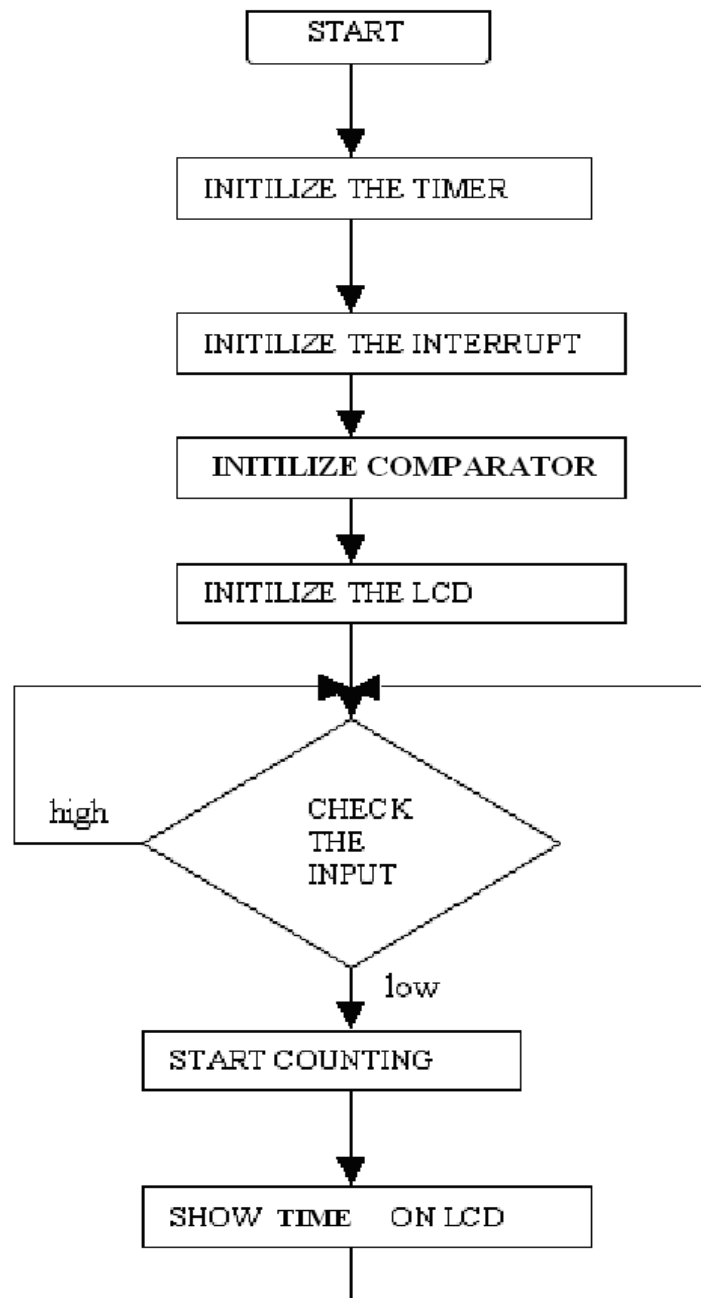
fig. 5. Software flow

1.5.1 Real Time Clock Calculations:

 The timer 0 of AT89C2051 has been used to form time base of real time clock. It has been configured to operate in 16 bit counter mode and timer interrupt has been enabled to generate timer tick interrupt. The count to be loaded in TH and TL registers has been calculated as follows.

        Timer 0,16 bit mode → maximum count 65535.

        Crystal frequency = 12MHz

Timer increments every machine cycle i.e. after 1 Machine cycle period = 1 μsec.

        So adjust timer count to get 50,000 μsec time base.

$$\therefore \text{ Timer count} = [\text{TH:TL}] = \text{Max. Count} - 50000 + 1$$
$$= 65535 - 50000 + 1$$
$$= 15536_d$$
$$= 3CB0 \text{ H}$$

        Thus timer registers have been loaded as TH0 = 3CH, TL0=B0H.

The timer vector routine performs following tasks when invoked at timer tick event.

        1) Reload timer registers again for the next counting cycle, and run the timer.

        2) Multiply timer tick period by 20 to get 1sec clock base.

        3) Process clock data in the form of HH:MM:SS.

        The display routine in the task context displays HH:MM:SS.

Listing 1: Device monitor program routiunes

```
/*----------------------------------------------------------------*/
                    // program to find Device Usage.
/*----------------------------------------------------------------*/

#include <REG2051.H>
void lcd_init();                        //Funtion to initialise LCD
void lcd_cmd(char cmd_value);        //Funtion to send commands  LCD
#define rs P3_4
#define rw P3_5
#define en P3_7
#define lcdport P3
#define busy_bit P3_3
unsigned char sec=0x00,min=0x00,hour=0x00;      //INITILISING FROM
                                                    ZERO TIMING
unsigned char cnt=0;
void lcdcmd(unsigned char a);         //TO SEND COMMAND TO LCD
void lcddata(unsigned char a);     //TO SEND DATA TO LCD
void timer0 (void) interrupt 1    //TO CALCULATE 1 SECOND DELAY
{
      TH0=0X3c;              //load higher and lower count in timer
      TL0=0Xb0;
      TR0=1;                 //reinitialize and run timer
      cnt++;
      if(cnt>19)              //multiply 50 msec delay 20
      {                       time to get require 1 sec delay
          cnt=0;
          sec++;                          //increment sec.
```

```c
                if(sec>59)

                        sec=0;
                        min++;                          //increment min.
                        if(min>59)
                        {
                                min=0;
                                hour++;                 //increment hour.
                        }
                }
        }
}
/*-------------------------------------------------------------------*/
void lcd_busy(void)                     //Funtion to check busy flag
{
        busy_bit = 1;
        rs=0;
        rw=1;
        en=0;
        en=1;
        while(busy_bit)
        {
                en=0;
                en=1;
        };
        en=0;
}
/*-------------------------------------------------------------------*/
void lcd_cmd(char cmd)                  //FUNCTION TO SEND COMMAND TO LCD
{
        lcdport = (cmd>>4) & 0x0f;   //TO CONVERT LCD 8 BIT DATA INTO 4
                                      BIT
        lcd_busy();                   //check busy flag
        rs = 0;
        rw = 0;
        en = 0;
        en = 1;
        en = 0;

        lcdport = cmd & 0x0f;
        rs = 0;
        rw = 0;
        en = 0;
        en = 1;
        en = 0;
}
/*-------------------------------------------------------------------*/
void lcd_data(char cmd)            //Funtion to send data to LCD
{
 lcdport = (cmd>>4) & 0x0f;        // TO CONVERT LCD 8 BIT DATA INTO 4 BIT
        lcd_busy();               // check busy flag
        rs = 1;
        rw = 0;
        en = 0;
        en = 1;
        en = 0;
```

```c
        lcdport = cmd & 0x0f;
        rs = 1;
        rw = 0;
        en = 0;
        en = 1;
        en = 0;
}
/*----------------------------------------------------------------*/
void lcd_init()                 //function for initialize
{
        lcd_cmd(0x02);          //to initialize LCD in 4-bit mode.
        lcd_cmd(0x28);          //to initialize LCD in 2 lines, 5X7 dots.
        lcd_cmd(0x0C);             and 4bit mode
        lcd_cmd(0x06);
        lcd_cmd(0x83);
}
/*----------------------------------------------------------------*/
void msg1(void)
{
                lcd_data('C');
                lcd_data('O');
                lcd_data('U');
                lcd_data('N');
                lcd_data('T');
                lcd_data('I');
                lcd_data('N');
                lcd_data('G');
                lcd_data(' ');
                lcd_data('T');
                lcd_data('I');
                lcd_data('M');
                lcd_data('E');
                lcd_data(':');
}
/*----------------------------------------------------------------*/
void main()
{
        AOUT = 1;
        TMOD=1;                 //timer 0 mode 1
        IE=0x82;                //timer0 intr enable
        lcd_init();
        lcd_cmd(0x80);          //selecting memory location 80 of LCD
        msg1();

        TCON=0x10;              //start timer
        while(1)
        {
                if(AOUT==0)                 //when comparator output == 0
                {
                        TCON=0;                 //stop timer
                }
                lcd_cmd(0xc0);              //set location of lcd
                lcd_data((hour/10)+0x30);     //to display hour
                lcd_data((hour%10)+0x30);
                lcd_data(':');
                lcd_data((min%10)+0x30);      //to display min
                lcd_data((min%10)+0x30);
```

```
            lcd_data(':');
            lcd_data((sec%10)+0x30);        //to display sec
            lcd_data((sec%10)+0x30);
        }
}
/*----------------------------------------------------------------*/
```

Table 1: LCD control signals and commands: [4]

Command	Function
rs	Register select rs=0(status reg)&rs=1(data reg)
rw	Read/write rw=0(write to display,rw=1(read from display)
en	Enable LCD en=1(enable display interface)
0x38	16*2 LCD select
0x01	Clear display
0x06	Shift cursor to right
0x0E	Screen on and cursor on

Table 2: Program structure

No.	Function	Description
1	void lcdinit()	LCD initialize
2	void timer 0 (void) interrupt 1	RTC processing in ISR context
3	void lcdcmd()	To provide commands to LCD
4	void lcddata()	To provide data to LCD
5	lcd_data_int(int time_val)	Function to send number on LCD
6	void main()	Entry point

Table 3 : Port pin configuration : [3]

Sr. no.	Port pin no.	Port pin descriptions
1	P1.0,P1.1,P3.6	Input and output of on-chip analog comparator.
2	P3.4,P3.5,P3.7	Control signal of LCD (en,rs,rw).
3	P3.0-P3.3	Data lines of LCD

1.6. Observations:

The data of output resistance of LDR is recorded for various light intensity of light source as indicated by table no 4. and resented graphically in Fig. 6. Intensity of light has varied by changing the voltage to the light source and respective reading for the intensity and LDR resistance has taken.

Table 4: Light Intensity Versus LDR Resistance

| obs. No.	Voltage of dimmer in V	Light intensity in lux	Resistance of LDR in ohm			
			R1	R2	R3	Mean (R)
1	40	0.1	118000	118000	117995	118000
2	60	0.5	42000	42000	42000	42000
3	80	6.7	9000	8988	9000	9000
4	90	51.3	2100	2100	20994	2100
5	100	110.5	1597	1600	1600	1600
6	110	280	1070	1070	1070	1070
7	120	460	892	895	895	895
8	130	502	475	475	477	475
9	140	850	470	475	475	475
10	150	1400	305	300	300	300
11	160	1500	270	270	270	270
12	180	1580	230	225	230	230
13	200	2100	110	100	100	100
14	210	5500	80	80	80	80
15	220	6500	50	45	45	45
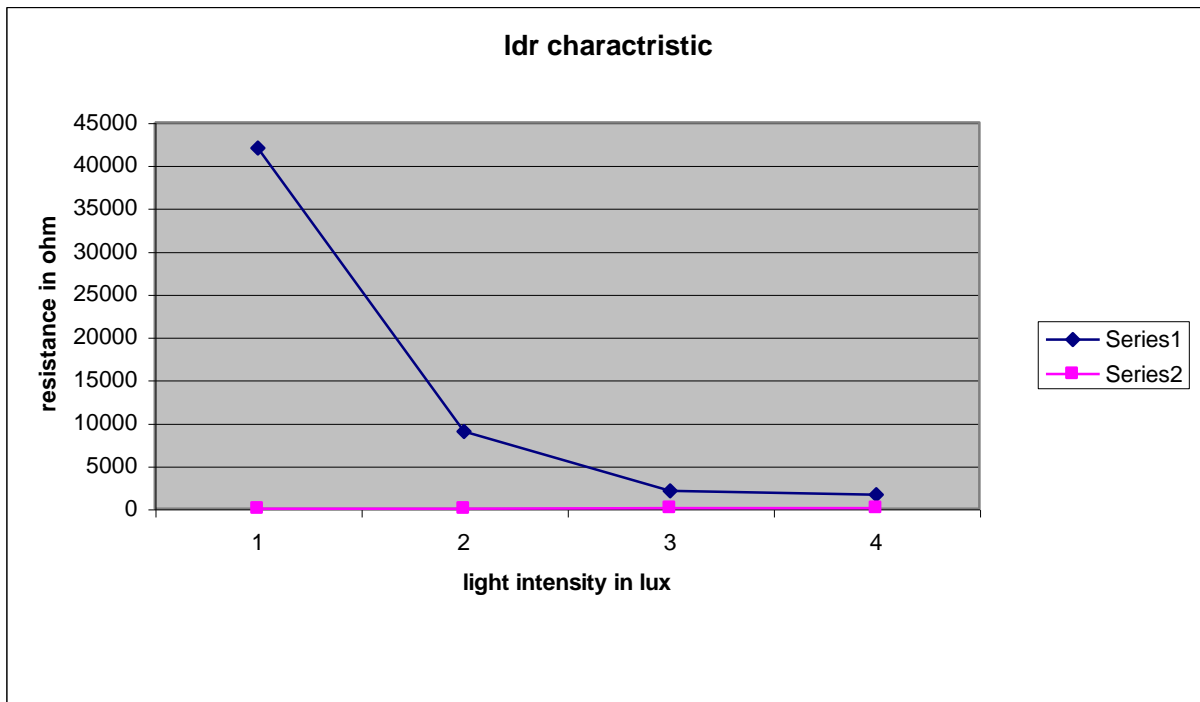
Fig 6.Graph of resistance of LDR verses light intensity.

1.7 onclusion:

The data provided by device monitor has been found useful for testing truthfulness of manufacturer specifications. The system can be modified to accommodate EEPROM to store the values for future reference. A serial/Ethernet[5] link with PC can also be used for PC data logging.

REFERENCES

1. J. Jayapandian, O.K. Sheela, A. Thiruarul, S. Kalavathi, 'Embedded Design and Virtual sensor interfacing', Journal of the Association for Laboratory Automation, Volume 11, Issue 5, October 2006,Pages 304-308

2. Photoresistor, From Wikipedia, the free encyclopedia, **http://en.wikipedia.org/wiki/Light_Dependent_Resistor**, Wikimedia Foundation, Inc. (visited on 7-Jan-2011)

3. feature and application of analog comparator, 2008 Atmel Corporation ltd **www.pdf/pdf/56224/ATMEL/89C2051.html** Page no. 1-3 Date:3 Jan. 2011.

4. Document no. :32217 LCD command and specification **http://www.futurlec.com/LCDdisp.shtml** 16×2 character LCD   from page 3 to 8 Date: 7 Jan 2011

5. P.A.Kadam et.al. Instruments and Experimental Techniques, 2009 **Embedded Web Server for Monitoring Environmental Parameters** *Vol. 52, No. 6, pp. 784–787* , Date: 11 Jan 2011 ISSN 0020_4412