

# **EMBEDDED SYSTEMS PROJECT (EE3401)**

## **Embedded System for Energy-Efficient Street Lighting Control Using 8051**

*Electrical Engineering Project submitted to the*

*National Institute of Technology Rourkela*

*In partial fulfillment of*

*Bachelors in technology in*

*Electrical Engineering*

*By*

*LAKSHMI NARAYANA GUPTHA GRANDHI – 121EE0970*

*CHALL SAI KUSHAL– 121EE0838*

*YESHESWINI VISUKAMURI – 121EE1073*

Under the supervision of

**Dr. Supratim Gupta**



Department of Electrical Engineering

**National Institute of Technology Rourkela**

**TOPIC: AUTOMATIC STREET LIGHT CONTROL USING  
SENSORS AND MICRO CONTROLLER**

## **ABSTRACT:**

Electricity being one of the primitive resources, must be utilised carefully, as the number of vehicles during night time are less frequent, building up a sensor-based street lights helps in saving electricity. The digital World we are living in allows us to use different technologies to automatically perform certain tasks. Such automation is very useful in certain areas like energy consumption, reducing human efforts, improving the standard of living etc. The project implemented here is one such project where the microcontroller based system automatically controls the street lights.

## **OBJECTIVE:**

The aim of this project is to automatically turn on or off the street lights by detecting vehicle movement. We implemented this project using an [8051 Microcontroller](#) and two Infrared (IR) sensors.

The job of the circuit is to turn on the first street light when a vehicle's movement is detected and to turn off that light as soon as the same vehicle's movement is detected by the second street light. The goal is to conserve energy by lighting up only the necessary sections of the highway based on the presence and movement of vehicles..

## **WORKING**

In this project, an automatic street light controlling system is developed using 8051 microcontroller. The working of the project is explained here.

The main component of the project is IR Sensor and we have used two of them. The placement of the sensors is important as it will determine the functioning of the project.

When vehicles passes the Sensor 1, it detects the vehicle an turns on the street light. This action will indicate to the 8051 Microcontroller that a vehicle has passed through first street light.

Hence, the microcontroller will turn on the light will keep the light turned on.

When the vehicle starts travelling from street light one to two, Sensor 2 detects the vehicle coming, Sensor 1 detects that a vehicle has left. The microcontroller will not turn off the street light 1 unless the sensor 2 detects the vehicle incoming.

So to summarize, as sensor 1 detects a vehicle's movement, street light one will get switched on and then when the sensor 2 detects the vehicle's movement, and sensor 1 detects the departure of the vehicle, the street light one will get turned off and the street light 2 will get turned on.

### **COMPONENTS REQUIRED:**

- AT89C51 Microcontroller
- 8051 Development Board
- 2 x Infrared Sensors
- 16 x 2 LCD Display
- 5V Relay Module • Lamp
- Connecting Wires
- Power Supply
- Proteus
- Bread Board

### **CIRCUIT DESCRIPTION:**

A 16 x 2 LCD Display, two IR Sensors and a 5V Relay Module must be connected to the 8051 Microcontroller.

First, connect the 8 data pins of the LCD to PORT1 pins i.e. P1.0 to P1.7.

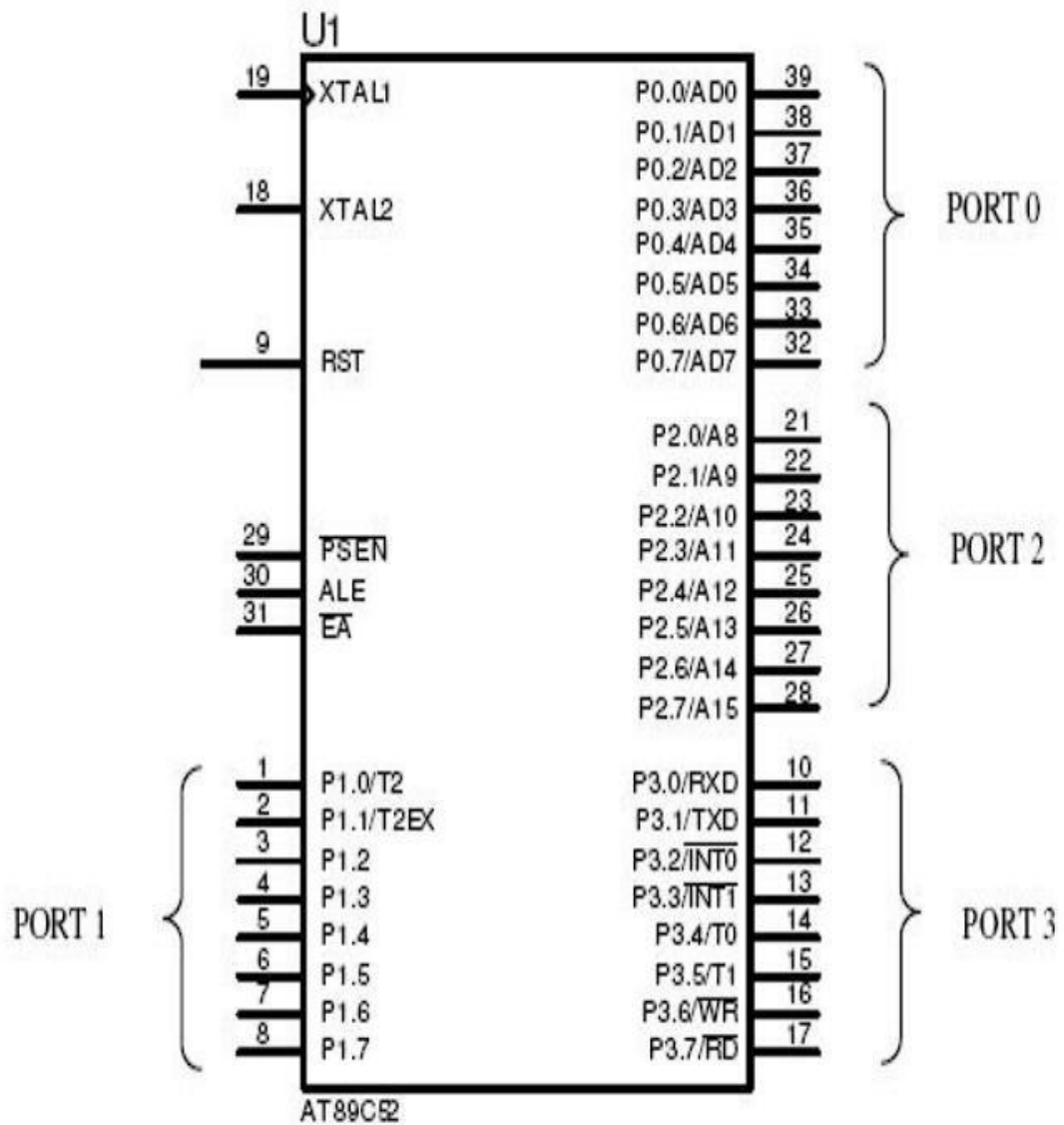
The 3 control pins of LCD i.e. RS, RW and E are connected to P3.6, GND and P3.7 pins respectively. A 10 K $\Omega$  Potentiometer is connected to contrast adjust pin of LCD i.e. its pin 3.

Two Reflective type IR Sensors are connected to PORT2 pins i.e. P2.0 and P2.1. Detailed circuit of the IR Sensor is mentioned in the Component Description.

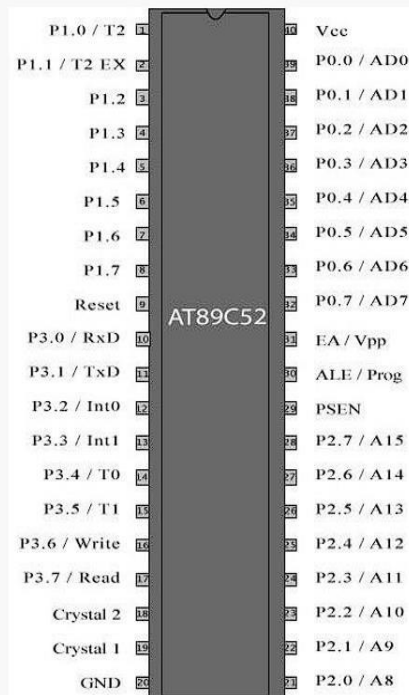
The input of the 5V Relay is connected to PORT0 pin P0.0. The detailed circuit of the 5V Relay module used in the project is explained in the component description section.

Alternatively, you can construct the circuit as per the circuit diagram (which consists of 5V Relay, Transistor, Diode and a Resistor).

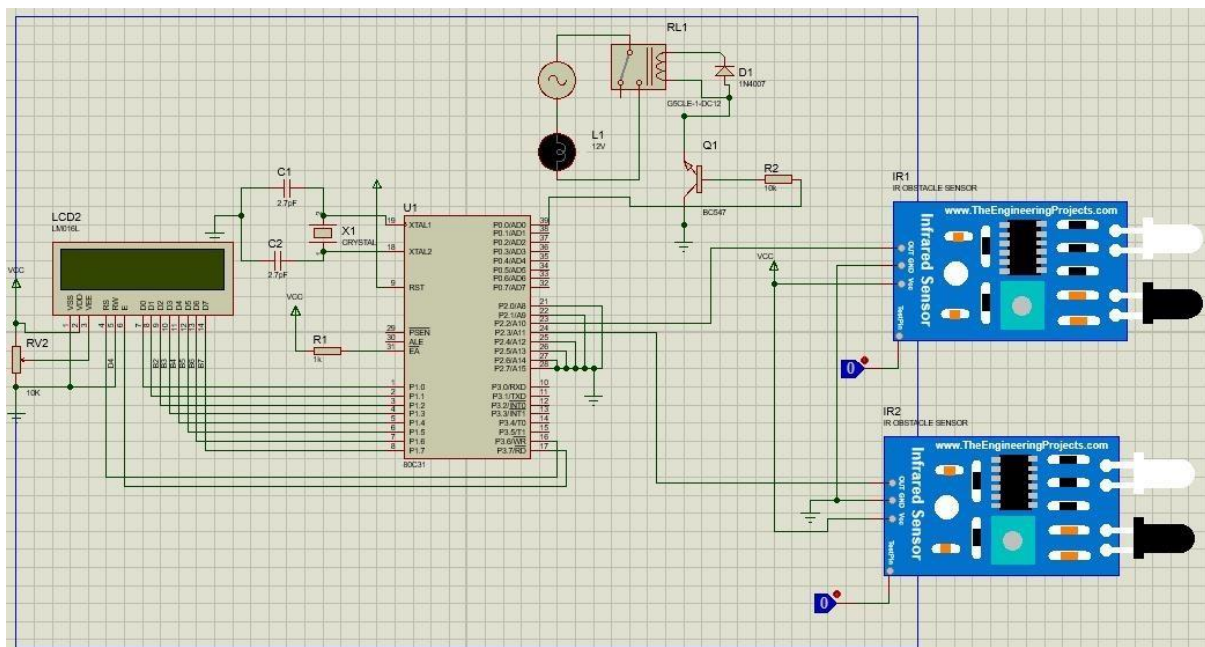
### I/O Port Pin and microcontroller connection diagram:



### PIN DIAGRAM OF AT89C52 MICROCONTROLLER :



**CIRCUIT DIAGRAM :**



## **CODE**

; storing essential information on ROM at 0400h (in range of 4K)

;MAKING LOOKUP TABLE

Table: org

0400H

DB "STREET LIGHT VEHICLE EXIT VEHICLE ENTRY COUNT: NO VEHICLE  
INVALID OPERATION"

;START OF PROGRAM FROM 0000h

org 0000H

MOV DPTR,#0400H ; its used as data pointer that means its locates at 0400H  
ADDRESS of ROM

MOV P2,#0FFH ; assigning PORT 1 as input

MOV P0,#00h

MOV R6,#00h ; R6 ACTS AS COUNTER

; LCD SETUP,START DISPLAY,CHECKING SENSORS

ACALL Lcd\_setup

ACALL Display

ACALL Check

Lcd\_setup:

MOV A,#38H ;setup 2 line 5\*7 matrix display

ACALL command

MOV A,#0CH ;Display ON and cursor OFF

ACALL command

MOV A,#01H ;Clear the old data

ACALL command

;MOV A,#06H;if used then cursor increment mode

;ACALL command

MOV A,#80H ;cursor home and starts left most point

ACALL command

RET

command:

MOV P1,A ;command on port A

CLR P3.6 ; register select as 0 for command

;CLR P3.1 ; display mode as write mode(R!/W)

SETB P3.7 ; Make latch as 1

CLR P3.7 ; to falling edge

ACALL Delay

RET

work:

MOV P1,A ;data on port A

SETB P3.6 ; register select as 1 for data

;CLR P3.1 ; display mode as write mode(R!/W)

SETB P3.7 ; Make latch as 1

CLR P3.7 ; to falling edge

ACALL Delay

RET

```

Delay:                                ;SOME delay for LCD
MOV TMOD, #01H                        ;Program TMOD -->(0000 0001)2 ...
    Timer0 Mode1
MOV TL0, #0D4H                        ;Load lower byte of Count
MOV TH0, #050H                        ;Load upper byte of Count
MOV TCON, #10H                        ;Program TCON --> (0001 0000)2 ...
    start Timer0
WAIT: JNB TCON.5, WAIT                ;Wait for overflow
MOV TCON, #00H                        ;Stop Timer0

RET

```

;welcome message Display:

```

MOV R3,#0Ch                          ;display of welcome message
MOV R2,#00h

```

```

ACALL lcd_displayer
RET

```

;checking

Check:

```

CLR C
ACALL Delay
MOV A,P2                            ;read the data
MOV B,#0CH
CJNE A,B,find                       ;check the data if 00h(initial case)
SJMP Check

```



;identifying sensor

find:

ACALL Delay

CLR C

CJNE A,#00h,goon

ACALL Check

goon:

CLR C

CJNE A,#08h,EXIT ;checking with entry if not equal than its must be exit

CLR C

CJNE A,#04h,ENTRY ;checking with eXIT if not equal than its must be ENTRY

ACALL Check

; ENTRY MODE

ENTRY:

SETB P0.0

ACALL Delay

MOV A,#01H ;Clear the old data

ACALL command

MOV R2,#19h ; displaying of person entering

MOV R3,#20h

ACALL lcd\_displayer

ACALL Entry\_count ; counting of persons

CLR C

SJMP Check

;entry count

Entry\_count:

MOV A,#0C0H ;FORCE CURSOR TO SECOND LINE

ACALL command

MOV R2,#21h ;displaying of "COUNT:"

MOV R3,#2ch

ACALL lcd\_displayer

;count increment

CLR C

MOV A,R6 ; getting data from R6 register

ADD A,#01 ; adding "1"

MOV R6,A ;new data stored back to R6 counter register

DA A ; converting from hex to decimal value (after addition only)

MOV R2,A

ACALL ConvertDisplay ; converting data to ASCII code

RET

; exit mode

EXIT:

ACALL Delay

MOV A,#01H ;Clear the old data

ACALL command

CLR C

CJNE R6,#00h,counter ; check the counter

MOV R2,#39h

MOV R3,#3Fh

```

ACALL lcd_displayer          ; if zero then give error message "INVALID"

MOV A,#0C0H                  ; Force cursor to second line
ACALL command

MOV R2,#41h                  ; showing of error message "Operation"
MOV R3,#49h
ACALL lcd_displayer

SJMP Check

counter:                      ;
MOV R2,#0Dh
MOV R3,#18h
ACALL lcd_displayer

ACALL Exit_count             ;
CLR C
CJNE R4,#00H,moveon
CLR P0.0   moveon:
SJMP Check
;exit counting Exit_count:
MOV A,#0C0H
ACALL command
;count decrement
CLR C
CJNE R6,#00h,start          ;

```

ACALL message

RET

start:

MOV A,R6 ;get data from counter

CLR C ; clear carry other wise it subtraction would be with carry

SUBB A,#01 ; subtract with "01"

MOV R6,A ; store to R6

DAA ; converting from hex to decimal value (after addition only)

MOV R4,A ; lets store it at R4 (DAA data) upcoming operations may  
distrub A

CLR C

CJNE R6,#00h,continue ;BEFORE MOVING ON LETS CHECK IF ANY  
PERSON IS THERE

clr P0.0

ACALL message

continue: ;Displaying "count"

SETB P0.0

MOV R2,#28h

MOV R3,#2Dh

CLR C

ACALL lcd\_displayer

MOV A,R4 ;value showing

MOV R2,A

ACALL ConvertDisplay

ACALL Check

ConvertDisplay:

CLR C ; Clear carry flag

MOV B,#10h

MOV A,R2

DIV AB ; Divide A by B, quotient in A, remainder in B

ADD A, #30h ; Convert quotient to ASCII

ACALL work ; Display the ASCII character

MOV A, B ; Move the remainder back to A

ADD A, #30h ; Convert remainder to ASCII

CALL work ; Display the ASCII character

RET

message: ;

MOV R2,#2Fh

MOV R3,#37h

ACALL lcd\_displayer

ACALL Check

RET

lcd\_displayer: ;sending bit by bit to LCD display

MOV A,R2

MOV B,R3 do:

MOVC A,@A+DPTR

ACALL work

INC R2

CLR C

MOV A,R2

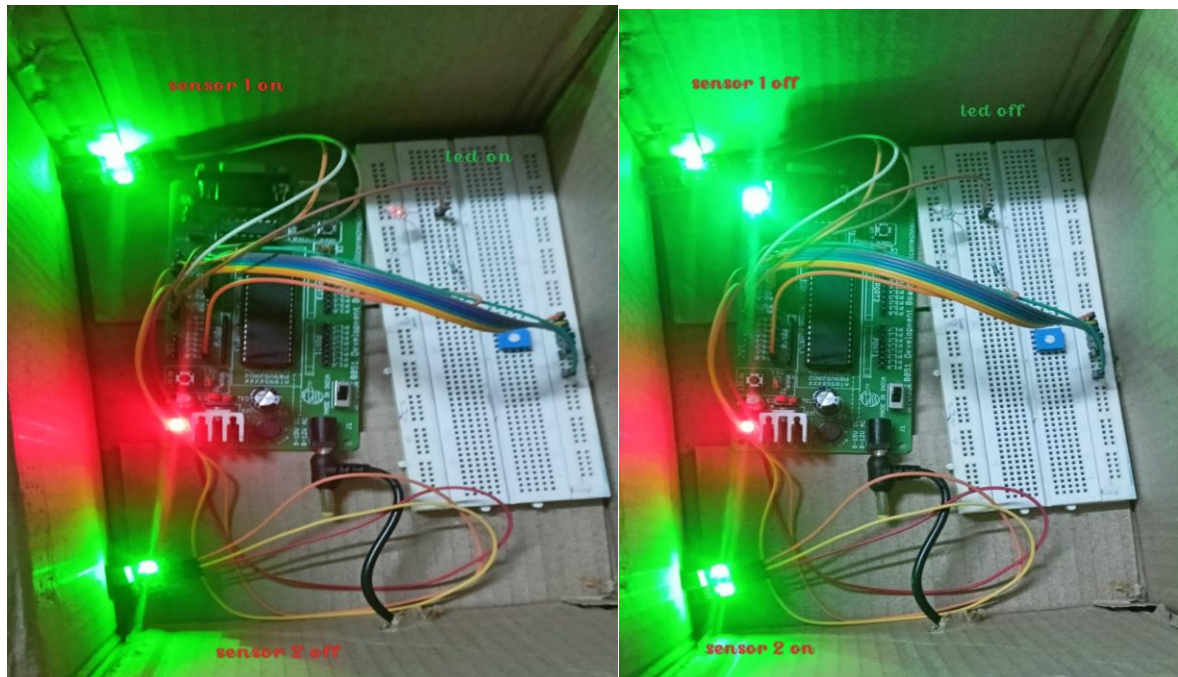
CJNE A,B,do

RET

Here: SJMP Here

END

### Simulation pics:



### ADVANTAGES:

- Reduced Light pollution, Energy conservation
- Cost-effective, longevity of street lights
- Enhanced Safety

## **CONCLUSION:**

By sensing vehicle movement, the created embedded system provides a workable alternative for energy-efficient street lighting. By installing such systems on highways, substantial energy savings and the promotion of sustainable urban infrastructure practises can be achieved.