**SELF STUDY REPORT**

ON

**“WIRELESS NOTICE BOARD USING GSM AND ARDUINO”**

Submitted in fulfilment of the requirements for the completion of

**SELF STUDY FOR COMMUNICATION THEORY-2 COURSE (16EC6DCCT2)**

IN

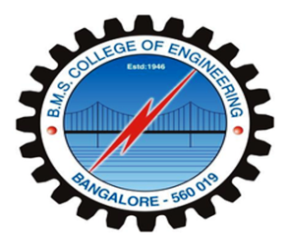
**ELECTRONICS AND COMMUNICATION ENGINEERING**

SUBMITTED BY:

|  |  |
| --- | --- |
| **Name of the Student** | **USN** |
| **Jaffar Aleem** | **1BM16EC037** |
| **Karthik Raj S S** | **1BM16EC043** |
| **Gagan L Naik** | **1BM16EC034** |
| **Likith S** | **1BM16EC052** |

**Name of the course instructor: G PRAPULLA**

**2018 - 19**

****

Department of Electronics and Communication Engineering

**B.M.S COLLEGE OF ENGINEERING**

(Autonomous College Affiliated to Visvesvaraya Technological University, Belgaum)

Bull Temple Road, Basavanagudi, Bangalore-560019

**CONTENTS**

**Chapter-1**

**1.1** Aim of the project…………………………………………………….

**1.2** Area of application……………………………………………………

**1.3** Introduction……………………………………………………………

**Chapter-2**

**2.1** Components used...……………………………………………………. **2.2** Circuit Diagram......……………………………………………………

**Chapter-3**

**3.1** Code description……………………………………………………….. **3.2** Code……………………………………..….…………………………..

**Chapter-4**

**4.1** Final implemented circuit and output...........………………………

**4.2** Conclusion……………………………………………………………

**Chapter 1**

**1.1 AIM**

The aim of the project is to build a wireless notice board which can display a message or notice using some display device like LCD.

This message can be easily set or changed from anywhere in the world, just by using the SMS facility of a mobile handset.

**1.2 AREA OF APPLICATION**

* Hotels
* Malls
* College
* Offices
* Home

**1.3 INTRODUCTION**

A notice board (pin board or bulletin board ) is a surface intended for the posting of public messages, for example, to advertise items wanted or for sale, announce events, or provide information. Notice boards are often made of a material such as [cork](https://en.wikipedia.org/wiki/Cork_(material)) to facilitate addition and removal of messages, as well as a writing surface such as [blackboard](https://en.wikipedia.org/wiki/Blackboard) or [whiteboard](https://en.wikipedia.org/wiki/Whiteboard). A notice board which combines a pin board (corkboard) and writing surface is known as a combination notice board. Notice boards can also be entirely in the digital domain and placed on computer networks so people can leave and erase messages for other people to read and see, as in a [bulletin board system](https://en.wikipedia.org/wiki/Bulletin_board_system).

Notice boards are particularly prevalent at universities. They are used by many sports groups and [extracurricular](https://en.wikipedia.org/wiki/Extracurricular) groups and anything from local shops to offices. [Dormitory](https://en.wikipedia.org/wiki/Dormitory) corridors, well-trafficked hallways, lobbies, and freestanding kiosks often have cork boards attached to facilitate the posting of notices. At some universities, lampposts, bollards, trees, and walls often become impromptu posting sites in areas where official boards are sparse in number

**Chapter 2**

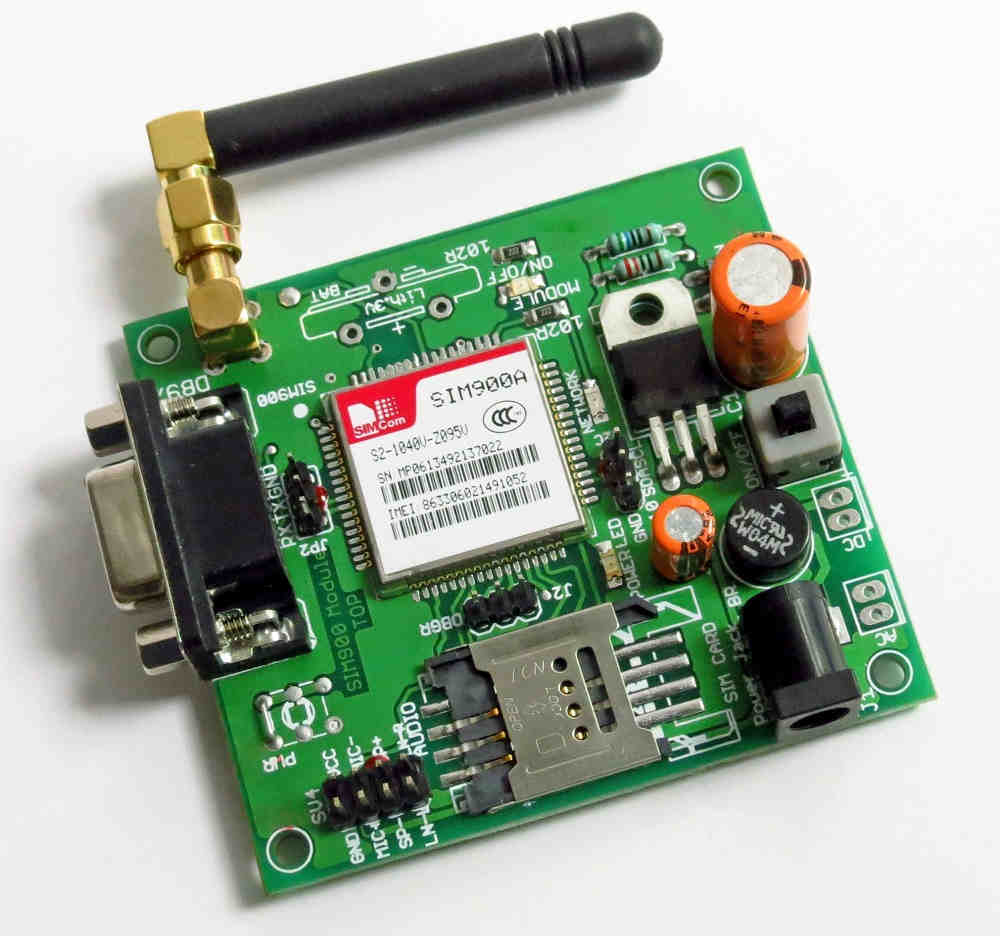
**2.1 COMPONENTS USED**

**1. GSM module (SIM900A)**

This is an ultra compact and reliable wireless module. The SIM900A is a complete Dual-band GSM/GPRS solution in a SMT module which can be embedded in the customer applications. Featuring an industry-standard interface, the SIM900A delivers GSM/GPRS 900/1800MHz performance for voice, SMS, Data, and Fax in a small form factor and with low power consumption. With a tiny configuration of 24mmx24mmx3mm, SIM900A can fit in almost all the space requirements in user applications, especially for slim and compact demand of design.

**Features**

* Dual-Band 900/ 1800 MHz
* GPRS multi-slot class 10/8GPRS mobile station class B
* Compliant to GSM phase 2/2+Class 4 (2 W @850/ 900 MHz)
* Class 1 (1 W @ 1800/1900MHz)
* Control via AT commands (GSM 07.07 ,07.05 and SIMCOM enhanced AT Commands)
* Low power consumption: 1.5mA(sleep mode)'
* Operation temperature: -40°C to +85 °C
* Status indicator (D5): It will flashes continuously whenever the call arrives otherwise it is left ON.
* Network LED(D6):This led will blink every second which indicates that the GSM module is not connected to the mobile network. Once the connection is established successfully, the LED will blink continuously every 3 seconds.

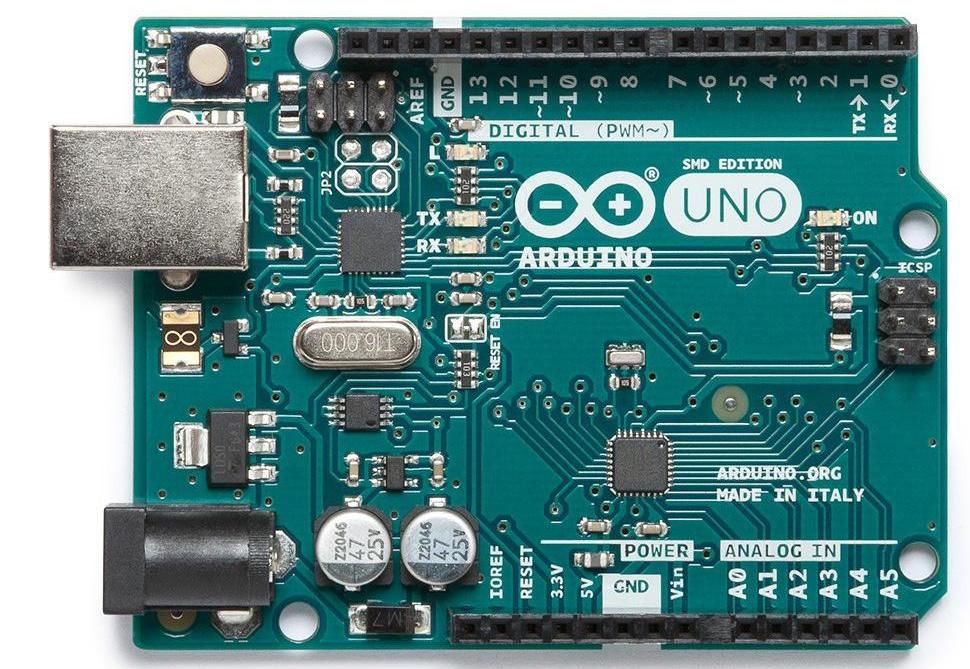


**2. Arduino UNO**

The Arduino UNO is an open-source microcontroller board based on the [Microchip](https://en.wikipedia.org/wiki/Microchip_Technology) [ATmega328P](https://en.wikipedia.org/wiki/ATmega328P) microcontroller and developed by [Arduino.cc](https://en.wikipedia.org/wiki/Arduino).  The board is equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards (shields) and other circuits. The board has 14 Digital pins, 6 Analog pins, and programmable with the [Arduino IDE](https://en.wikipedia.org/wiki/Arduino#Software) (Integrated Development Environment) via a type B USB cable. It can be powered by a USB cable or by an external 9 volt battery, though it accepts voltages between 7 and 20 volts. It is also similar to the Arduino Nano and Leonardo. The hardware reference design is distributed under a [Creative Commons](https://en.wikipedia.org/wiki/Creative_Commons) Attribution Share-Alike 2.5 license and is available on the Arduino website. Layout and production files for some versions of the hardware are also available. "Uno" means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0. The Uno board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases. The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform. The ATmega328 on the Arduino Uno comes preprogrammed with a bootloader that allows uploading new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol. The Uno also differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it uses the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

**Features**

* [Microcontroller](https://en.wikipedia.org/wiki/Microcontroller): [Microchip](https://en.wikipedia.org/wiki/Microchip_Technology) [ATmega328P](https://en.wikipedia.org/wiki/ATmega328P) [[7]](https://en.wikipedia.org/wiki/Arduino_Uno#cite_note-website-7)
* Operating Voltage: 5 Volts
* Input Voltage: 7 to 20 Volts
* Digital I/O Pins: 14 (of which 6 provide PWM output)
* Analog Input Pins: 6
* DC Current per I/O Pin: 20 mA
* DC Current for 3.3V Pin: 50 mA
* [Flash Memory](https://en.wikipedia.org/wiki/Flash_Memory): 32 KB of which 0.5 KB used by [bootloader](https://en.wikipedia.org/wiki/Booting" \l "BOOT-LOADER" \o "Booting)
* [SRAM](https://en.wikipedia.org/wiki/Static_random-access_memory): 2 KB
* [EEPROM](https://en.wikipedia.org/wiki/EEPROM): 1 KB
* Clock Speed: 16 MHz



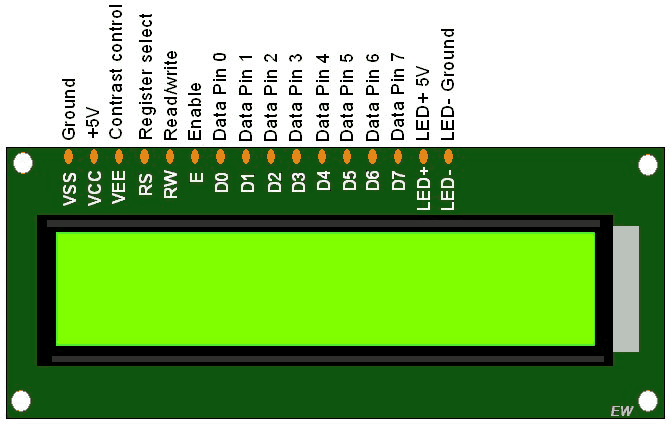
**3. LCD**

LCD (Liquid Crystal Display) screen is an electronic display module and find a wide range of applications. A 16x2 LCD display is very basic module and is very commonly used in various devices and circuits. These modules are preferred over [seven segments](http://www.engineersgarage.com/content/seven-segment-display) and other multi segment [LED](http://www.engineersgarage.com/content/led)s. The reasons being: LCDs are economical; easily programmable; have no limitation of displaying special & even [custom characters](http://www.engineersgarage.com/microcontroller/8051projects/create-custom-characters-LCD-AT89C51) (unlike in seven segments), [animations](http://www.engineersgarage.com/microcontroller/8051projects/display-custom-animations-LCD-AT89C51) and so on.

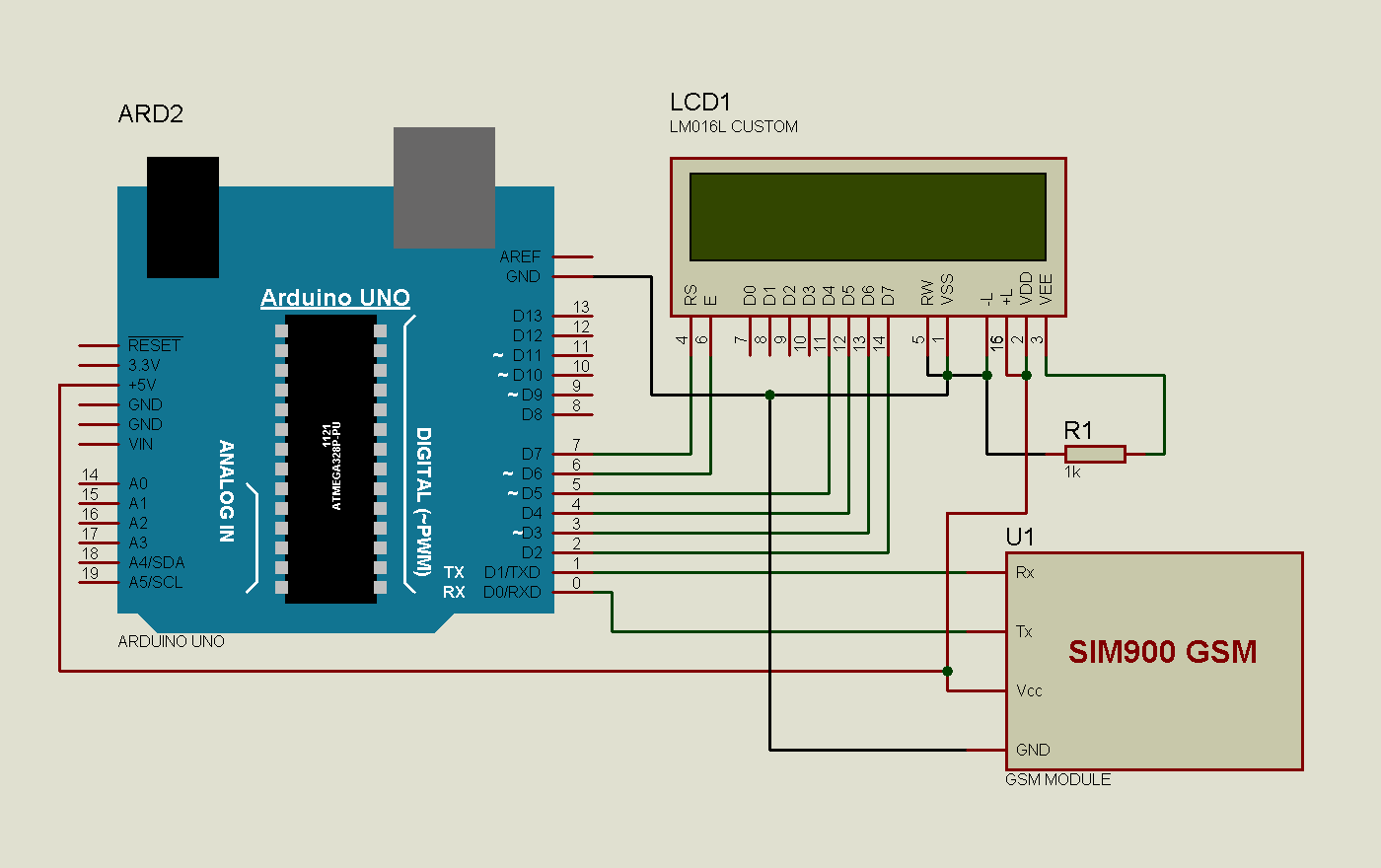
A **16x2 LCD** means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix. This LCD has two registers, namely, Command and Data.

The command register stores the command instructions given to the LCD. A command is an instruction given to LCD to do a predefined task like initializing it, clearing its screen, setting the cursor position, controlling display etc. The data register stores the data to be displayed on the LCD. The data is the ASCII value of the character to be displayed on the LCD. Click to learn more about internal structure of a [LCD](http://www.engineersgarage.com/insight/how-lcd-works).

|  |  |  |
| --- | --- | --- |
| Pin No | Function | Name |
| 1 | Ground (0V) | Ground |
| 2 | Supply voltage; 5V (4.7V – 5.3V) | Vcc |
| 3 | Contrast adjustment; through a variable resistor | VEE |
| 4 | Selects command register when low; and data register when high | Register Select |
| 5 | Low to write to the register; High to read from the register | Read/write |
| 6 | Sends data to data pins when a high to low pulse is given | Enable |
| 7 | 8-bit data pins | DB0 |
| 8 | DB1 |
| 9 | DB2 |
| 10 | DB3 |
| 11 | DB4 |
| 12 | DB5 |
| 13 | DB6 |
| 14 | DB7 |
| 15 | Backlight VCC (5V) | Led+ |
| 16 | Backlight Ground (0V) | Led- |



**2.2 CIRCUIT DIAGRAM**



Connections of **Wireless Notice Board using GSM and Arduino** are simple and shown in the figure below. Here a liquid crystal display (LCD) is used for display the “Notice” or message, which is sent though the mobile phone as SMS. Data pins of LCD namely RS, EN, D4, D5, D6, D7 are connected to arduino digital pin number 7, 6, 5, 4, 3, 2. And Rx and Tx pin of GSM module is directly connected at Tx and Rx pin of Arduino respectively. And GSM module is powered by using a 12 volt adaptor.

**CHAPTER 3**

**3.1 CODE DESCRIPTION**

The code of the program is easily understandable; the new thing here is GSN initialization function gsm\_init(), which is explained in the end.

In the program, first of all we include library for liquid crystal display (LCD) and then we defines data and control pins for LCD and some variables.

**#include <LiquidCrystal.h>**

**LiquidCrystal lcd(7,6,5,4,3,2);**

**int led=13;**

**int temp=0,i=0,x=0,k=0;**

**char str[100],msg[32**];

After this, serial communication is initialized at 9600 bps and gives direction to used pin. And initialize GSM Module in setup loop.

**void setup()**

**{**

**lcd.begin(16,2);**

**Serial.begin(9600);**

**pinMode(led, OUTPUT);**

**digitalWrite(led, HIGH);**

**lcd.print("GSM Initilizing...");**

**gsm\_init();**

**lcd.setCursor(0,0);**

**lcd.print("Wireless Notice");**

 For receiving data serially we use two functions, one is Serial.available which checks any serial data is coming or not and other one is Serial.read which reads the data that comes serially.

**void serialEvent()**

**{**

**while(Serial.available())**

**{**

**char ch=(char)Serial.read();**

**str[i++]=ch;**

**if(ch == '\*')**

**{**

**temp=1;**

**lcd.clear();**

**lcd.print("Message Received");**

**delay(1000);**

**}**

**}**

**}**

After receiving data serially, we store it in a string and this string is checked for ‘#’ and ‘\*’, to find the starting and ending of the Notice or message. Then finally Notice is printed on LCD using lcd.print:

**void loop()**

**{**

**for(unsigned int t=0;t<60000;t++)**

**{**

**serialEvent();**

**if(temp==1)**

**{**

**x=0,k=0,temp=0;**

**while(x<i)**

**{**

**while(str[x]=='#')**

**{**

**x++;**

**while(str[x]!='\*')**

**{**

**msg[k++]=str[x++];**

Initialization function ‘gsm\_init()’ for GSM is important here, where firstly, GSM module is checked whether it is connected or not by sending ‘AT’ command to GSM module. If response OK is received, means it is ready. System keeps checking for the module until it becomes ready or until ‘OK’ is received. Then ECHO is turned off by sending the ATE0 command, otherwise GSM module will echo all the commands. Then finally Network availability is checked through the ‘AT+CPIN?’ command, if inserted card is SIM card and PIN is present, it gives the response +CPIN: READY. This is also check repeatedly until the network is found. This can be clearly understood by the Video below.

**void gsm\_init()**

**{**

**lcd.clear();**

**lcd.print("Finding Module..");**

**boolean at\_flag=1;**

**while(at\_flag)**

**{**

**Serial.println("AT");**

**while(Serial.available()>0)**

**{**

**if(Serial.find("OK"))**

**at\_flag=0;**

**}**

**delay(1000);**

**}**

**3.2 CODE**

#include <LiquidCrystal.h>

LiquidCrystal lcd(7,6,5,4,3,2);

int led=13;

int temp=0,i=0,x=0,k=0;

char str[100],msg[32];

void setup()

{

lcd.begin(16,2);

Serial.begin(9600);

pinMode(led, OUTPUT);

digitalWrite(led, HIGH);

lcd.print("GSM Initilizing...");

gsm\_init();

lcd.setCursor(0,0);

lcd.print("Wireless Notice");

lcd.setCursor(0,1);

lcd.print(" Board ");

delay(2000);

lcd.clear();

lcd.print("Circuit Digest");

delay(1000);

lcd.setCursor(0,1);

lcd.print("System Ready");

Serial.println("AT+CNMI=2,2,0,0,0");

delay(500);

Serial.println("AT+CMGF=1");

delay(1000);

digitalWrite(led, LOW);

}

void loop()

{

for(unsigned int t=0;t<60000;t++)

{

serialEvent();

if(temp==1)

{

x=0,k=0,temp=0;

while(x<i)

{

while(str[x]=='#')

{

x++;

while(str[x]!='\*')

{

msg[k++]=str[x++];

}

}

x++;

}

msg[k]='\0';

lcd.clear();

lcd.print(msg);

delay(1000);

temp=0;

i=0;

x=0;

k=0;

}

}

lcd.scrollDisplayLeft();

}

void serialEvent()

{

while(Serial.available())

{

char ch=(char)Serial.read();

str[i++]=ch;

if(ch == '\*')

{

temp=1;

lcd.clear();

lcd.print("Message Received");

delay(1000);

}

}

}

void gsm\_init()

{

lcd.clear();

lcd.print("Finding Module..");

boolean at\_flag=1;

while(at\_flag)

{

Serial.println("AT");

while(Serial.available()>0)

{

if(Serial.find("OK"))

at\_flag=0;

}

delay(1000);

}

lcd.clear();

lcd.print("Module Connected..");

delay(1000);

lcd.clear();

lcd.print("Disabling ECHO");

boolean echo\_flag=1;

while(echo\_flag)

{

Serial.println("ATE0");

while(Serial.available()>0)

{

if(Serial.find("OK"))

echo\_flag=0;

}

delay(1000);

}

lcd.clear();

lcd.print("Echo OFF");

delay(1000);

lcd.clear();

lcd.print("Finding Network..");

boolean net\_flag=1;

while(net\_flag)

{

Serial.println("AT+CPIN?");

while(Serial.available()>0)

{

if(Serial.find("+CPIN: READY"))

net\_flag=0;

}

delay(1000);

}

lcd.clear();

lcd.print("Network Found..");

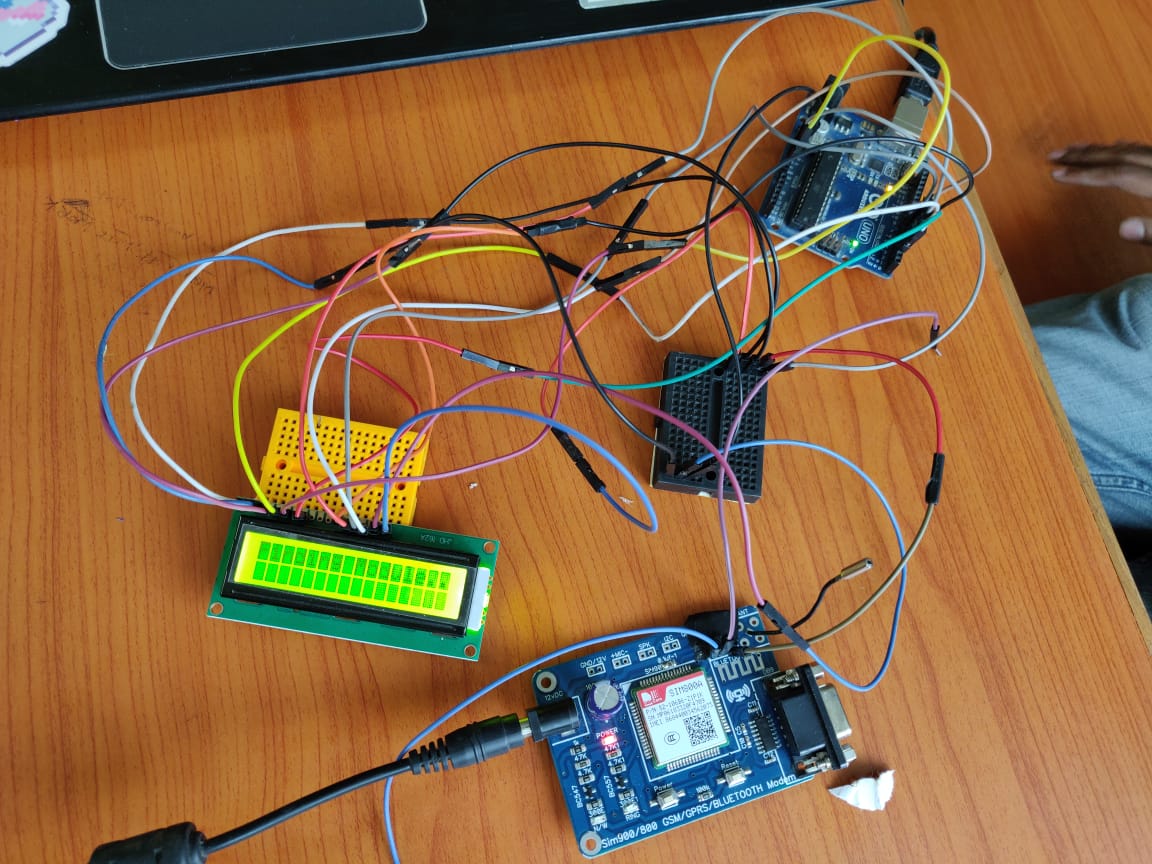
delay(1000);

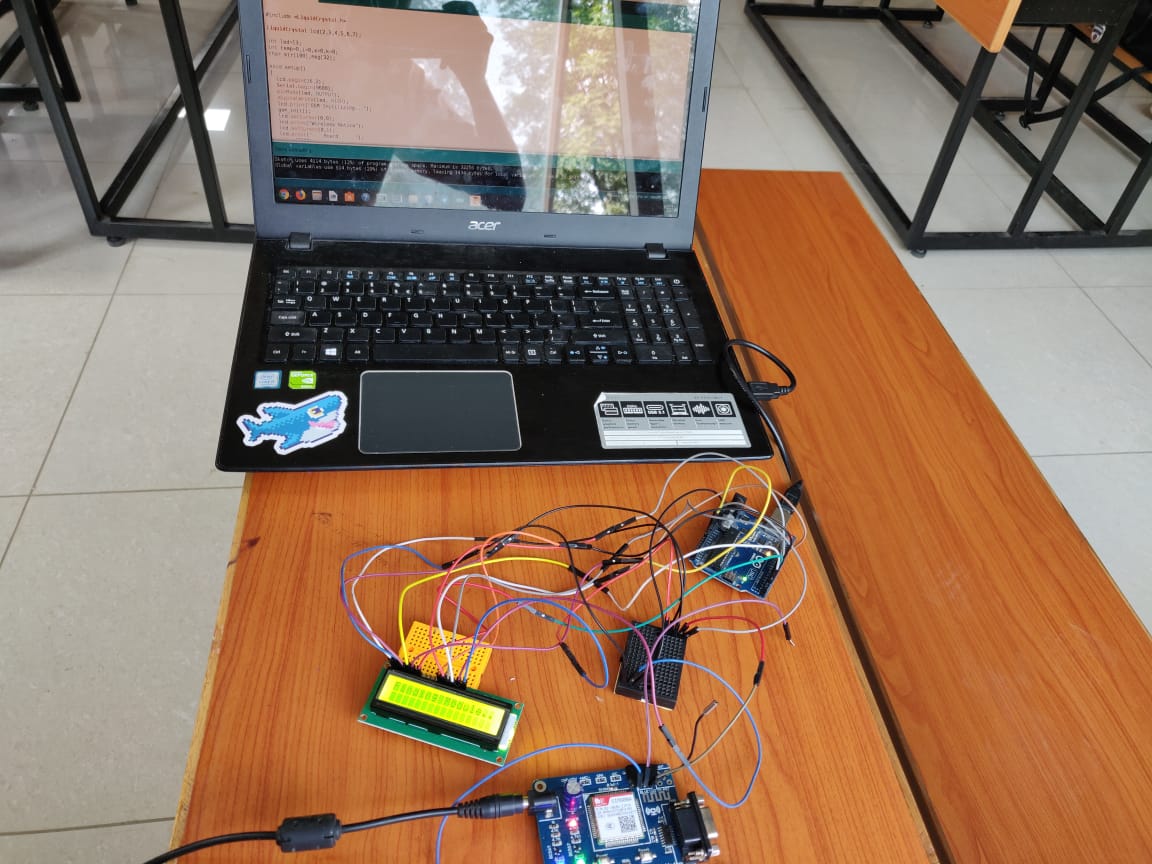
lcd.clear();

}

**CHAPTER 4**

**4.1 FINAL IMPLEMENTED CIRCUIT AND OUTPUT**





**4.2 CONCLUSION**

The progress in science and technology is a nonstop process. New things and new technology are being invented. As the technology grows day by day we can imagine about the future in which thing that may occupy every place. The proposed system based on ATMEGA microcontroller is found to be more compact, user friendly and less complex which can readily be used in order to perform. Several tedious and repetitive tasks. Though it is designed keeping in mind about the need for industry, it can extended for other purposes such as commercial and research applications. Due to probability of high technology (ATMEGA microcontroller) used this GPRS BASED WIRELESS NOTICE BOARD USING ARDUINO system is fully software controlled with less hardware circuit. The feature make this system is the base for future system. The future scope of this project can be extended to many different fields for commercial purpose also, with slight variations according to the application which it is to be designed for some of the examples are as follows to Manage Traffic Metropolitan cities are prone to high traffic congestion. One way to avoid this would be to inform people beforehand to take alternate routes. A wireless notice board using arduino serves well for this support. Crime Prevention Display boards can be put on roads to display tips on public security, accident prevention, and information on criminals on the run and also to flash messages regarding vehicle thefts as and when they occur. This project can be implemented in offices, colleges, schooles etc…