# Description:

Write a program to show Modem details on LCD like Operator Name, Network Range, SIM Status, GPRS Status, Modem Status, IMEI Number.

# Source Code:

#include <LiquidCrystal.h> // include the library code:

LiquidCrystal lcd(11,12,14,15,16,17); // initialize the library with the numbers of the interface pins

int signal\_level;

char network\_name[15] = "";

char imc[16];

String imei;

char c;

int u=0;

void setup()

{

pinMode(13,OUTPUT); //SIM808 wakeup connected on pin 13 in IomaTic board

digitalWrite(13, HIGH); //Initialize the SIM808 Module

delay(1000);

digitalWrite(13, LOW); //Sending wake up signal to SIM808 Module

delay(1000);

digitalWrite(13, HIGH); //Keeping SIM808 in active/wakeup state

delay(5000);

lcd.begin(16, 2); //Initialize the LCD in 16x2 mode

delay(1000);

lcd.setCursor(0,0); //Set cursor at first character/coloumn of first line/row

lcd.print(" IomaTic "); //Print the message as metioned cursor location

lcd.setCursor(0,1); //Set cursor at first character/coloumn of first line/row

lcd.print("GSM Modem Test...."); //Print the message as metioned cursor location

Serial.begin(9600); //Initialize a serial communication with baud rate 9600

delay(1000);

Serial.println("AT"); //check if sim800 module responds

delay(100);

if (Serial.find("OK"))

{

lcd.setCursor(0,1); //Set cursor at first character/coloumn of first line/row

lcd.print("Modem OK............"); //Print the message as metioned cursor location

}

else

{

lcd.setCursor(0,1); //Set cursor at first character/coloumn of first line/row

lcd.print("Modem Not OK........"); //Print the message as metioned cursor location

}

delay(2000); // wait for sim800 to settle a bit

Serial.println("AT+CSMINS?"); // check if SIM card inserted

delay(300);

if (Serial.find("CSMINS: 0,0"))

{

lcd.setCursor(0,1); //Set cursor at first character/coloumn of first line/row

lcd.print("SIM Card Not OK.... "); //Print the message as metioned cursor location

}

else

{

lcd.setCursor(0,1); //Set cursor at first character/coloumn of first line/row

lcd.print("SIM Card OK...."); //Print the message as metioned cursor location

}

delay(2000); // wait for sim800 to settle a bit

Serial.println("AT+CSQ"); // SIGNAL STRENGTH

delay(100);

if (Serial.find(":"))

{

signal\_level = Serial.parseInt();

lcd.setCursor(0,1); //Set cursor at first character/coloumn of first line/row

if(signal\_level==0) lcd.print("Network: Poor ");

if(signal\_level==1) lcd.print("Network: Average ");

if(signal\_level>=2 || signal\_level<=30 ) lcd.print("Network: Good ");

if(signal\_level==31) lcd.print("Network: Excellent ");

if(signal\_level==99) lcd.print("Network: Unknown ");

}

else

{

lcd.setCursor(0,1); //Set cursor at first character/coloumn of first line/row

lcd.print("No Network......."); //Print the message as metioned cursor location

}

// delay(2000);

// Serial.println("AT+COPS?"); // OPERATOR

// delay(100);

// if (Serial.find("\"")) // find operator name between two double quotes

// {

// c = Serial.read();

// while (c != '"' || u < 10)

// {

// network\_name[u] = c;

// c = Serial.read();

// u++;

// }

// lcd.setCursor(0,1); //Set cursor at first character/coloumn of first line/row

// lcd.print("Operator:");

// lcd.print(network\_name);

// }

delay(2000);

Serial.println("AT+CGSN");

delay(100);

while(Serial.available() > 0)

{

c = Serial.read();

if (c=='0' || c=='1' || c=='2' || c=='3' || c=='4' || c=='5' || c=='6' || c=='7' || c=='8' || c=='9' )

{

imc[u] = c;

u++;

}

lcd.setCursor(0,1); //Set cursor at first character/coloumn of first line/row

lcd.print(imc);

}

}

void loop()

{

//This is single task program hence nothing to do in loop,

//everything will be executed in setup function only.

}

# Libraries:

No additional libraries required.

# Functions:

*AT Commands:*

AT Commands are commands which are used to control the modems where AT stands for Attention.