

Project: Controlling and monitoring changes via  
SMS.

Name: Karim Ayyad.

Abstract:

- In locations such as remote areas, it's not the best option to control solenoids, pumps or motors manually, especially in areas with extreme climate when you could get better, cheaper and hence more logical choice. This project is developed to control output devices with SMS as the user can send specified commands which the hardware performs and sends the feedback and to monitor any changes occurred to any input switch such as limit switches. Naturally it requires an area with network coverage to connect the SIM card and communicate with the controlling users.

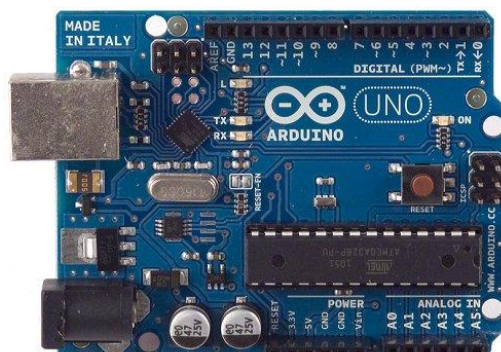
## 1. Introduction:

- The main component in the project is Arduino Uno. A GSM module is added to connect the SIM Card with the network. A relay module is added to the project to perform the procedure which the user sends.

## 2. Components:

### 2.1 Arduino Uno R3:

- It is an open-source microcontroller that could be programmed by C++ with the official IDE software. Arduino have several analog and digital I/O. Interfacing them lead to write and read data from different types of sensors, switches and modules.



### 2.2 SIM808 module :

- It's a GSM and GPRS module integrated with GPS and comes with quad band. In this project, we will use it to send and receive SMS using AT commands and hence we will install an antenna to GSM slot. The module is connected with the Arduino by using 4 pins. 2 of them are set for power (VCC-GND) as it requires power range 3.4~4.4V. The other 2 pins (i.e. Rx for receiving and Tx for sending data) are connected to digital pins of the Arduino. An SIM card is installed to the module. It could be in different sizes except nano-sim and e-sim.



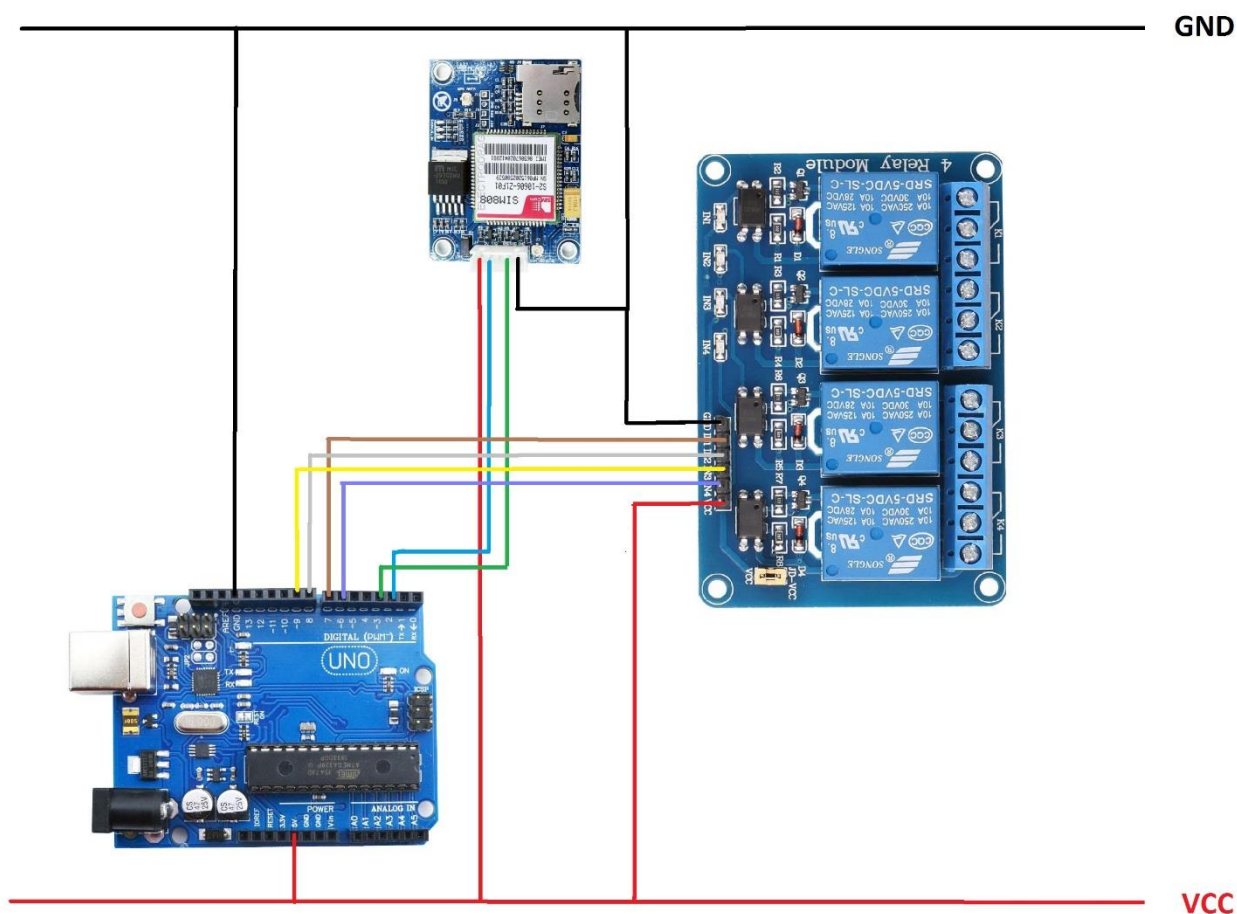
- The antenna must be put in an area with network coverage to enable the SIM card to connect to the network and communicate with the user by SMS.

### 2.3 Relay module:

- Relays are electromechanical components used in controlling. Basically it has a normally-opened point and a normally-closed point with a common point. It has 3 input pins, two of them are for power (5v-GND) and the third one is for the input signal from the output pins .



### 3. Diagram:



### 4. Bill of Materials:

Type	Price/unit (EGP)	Quantity
Arduino Uno R3	140	1
SIM808 module	700	1
Jumpers F-M 20cm	0.5	5
Jumpers F-F 20cm	0.5	5
Plastic Box 250×150×60	50	1
Relay Module 5Vdc – 4 channels	60	1
Total =	955 EGP	

Table. (The total cost of the device)

## 5. Appendix:

### - C++ Arduino code:

```
// Include Software Serial library to communicate with GSM
#include <SoftwareSerial.h>

// Configure software serial port (Tx,Rx)
SoftwareSerial SIM808(2, 3);

// Variables
String textMessage;

String StateOutput1 = "off";
String StateOutput2 = "off";
String StateOutput3 = "off";
String StateOutput4 = "off";
String StateInput1 = "ON";
String StateInput2 = "ON";
String StateInput3 = "ON";
String StateInput4 = "ON";
bool inputSent1 = false;
bool inputSent2 = false;
bool inputSent3 = false;
bool inputSent4 = false;
const int relay1 = 7;
const int relay2 = 8;
const int relay3 = 9;
const int relay4 = 6;
int reading1, reading2, reading3, reading4;
void setup() {
    delay(1000);

    // Set relays as OUTPUT
    pinMode(relay1, OUTPUT);
```

```

pinMode(relay2, OUTPUT);
pinMode(relay3, OUTPUT);
pinMode(relay4, OUTPUT);
//Set inputs pins
pinMode(10, INPUT_PULLUP);
pinMode(11, INPUT_PULLUP);
pinMode(12, INPUT_PULLUP);
pinMode(13, INPUT_PULLUP);

// By default the relays are off
digitalWrite(relay1, LOW);
digitalWrite(relay2, LOW);
digitalWrite(relay3, LOW);
digitalWrite(relay4, LOW);

// Automatically turn on the shield
Serial.begin(9600);
SIM808.begin(9600);

// Give time to your GSM shield log on to network
Serial.print("SIM808 ready...");

// AT command to set SIM808 to SMS mode
SIM808.print("AT+CMGF=1\r");
delay(500);
// Set module to send SMS data to serial out upon receipt
SIM808.print("AT+CNMI=2,2,0,0,0\r");
delay(500);
}

void loop() {

// read the message received
if (SIM808.available() > 0) {

```

```

    textMessage = SIM808.readString();

    Serial.print(textMessage);

    delay(1000);
}

//Controlling the 1st relay

if (textMessage.indexOf("1ON") >= 0 || textMessage.indexOf("1on") >= 0 ||
textMessage.indexOf("1On") >= 0 || textMessage.indexOf("1oN") >= 0) {

    // Turn on relay and save current state

    digitalWrite(relay1, HIGH);

    StateOutput1 = "on";

    Serial.println("Relay 1 set to ON");

    textMessage = "";

    SendFeedbackSMS();

}

if (textMessage.indexOf("1OFF") >= 0 || textMessage.indexOf("1off") >= 0 ||
textMessage.indexOf("1Off") >= 0) {

    // Turn off relay and save current state

    digitalWrite(relay1, LOW);

    StateOutput1 = "off";

    Serial.println("Relay 1 set to OFF");

    textMessage = "";

    SendFeedbackSMS();

}

//Controlling the 2nd relay

if (textMessage.indexOf("2ON") >= 0 || textMessage.indexOf("2on") >= 0 ||
textMessage.indexOf("2On") >= 0 || textMessage.indexOf("2oN") >= 0) {

    digitalWrite(relay2, HIGH);

    StateOutput2 = "on";

    Serial.println("Relay 2 set to ON");

    textMessage = "";

```

```

    SendFeedbackSMS();
}

if (textMessage.indexOf("2OFF") >= 0 || textMessage.indexOf("2off") >= 0 ||
textMessage.indexOf("2Off") >= 0) {
    digitalWrite(relay2, LOW);
    StateOutput2 = "off";
    Serial.println("Relay 2 set to OFF");
    textMessage = "";
    SendFeedbackSMS();
}

//Controlling the 3rd relay
if (textMessage.indexOf("3ON") >= 0 || textMessage.indexOf("3on") >= 0 ||
textMessage.indexOf("3On") >= 0 || textMessage.indexOf("3oN") >= 0) {
    digitalWrite(relay2, HIGH);
    StateOutput3 = "on";
    Serial.println("Relay 3 set to ON");
    textMessage = "";
    SendFeedbackSMS();
}

if (textMessage.indexOf("3OFF") >= 0 || textMessage.indexOf("3off") >= 0 ||
textMessage.indexOf("3Off") >= 0) {
    digitalWrite(relay2, LOW);
    StateOutput3 = "off";
    Serial.println("Relay 3 set to OFF");
    textMessage = "";
    SendFeedbackSMS();
}

//Controlling the 4th relay

```



```

    if (textMessage.indexOf("4ON") >= 0 || textMessage.indexOf("4on") >= 0 ||
textMessage.indexOf("4On") >= 0 || textMessage.indexOf("4oN") >= 0) {

        digitalWrite(relay4, HIGH);

        StateOutput4 = "on";

        Serial.println("Relay 4 set to ON");

        textMessage = "";

        SendFeedbackSMS();

    }

    if (textMessage.indexOf("4OFF") >= 0 || textMessage.indexOf("4off") >= 0 ||
textMessage.indexOf("4Off") >= 0) {

        digitalWrite(relay4, LOW);

        StateOutput4 = "off";

        Serial.println("Relay 4 set to OFF");

        textMessage = "";

        SendFeedbackSMS();

    }


// Send the current states of the relays and the input states

    if (textMessage.indexOf("STATES") >= 0 || textMessage.indexOf("States") >=
0 || textMessage.indexOf("STATUS") >= 0 || textMessage.indexOf("Status") >=
0) {

        delay(5000);

        SendStatusSMS(StateOutput1, StateOutput2, StateOutput3, StateOutput4);

        textMessage = "";

    }


//Checking inputs and send its feedback
while (true)
{ reading1 = digitalRead(10);

    if (reading1 == LOW)

    { Serial.println("Input 1 off");

        if (inputSent1 == true)

            break;

        SendFeedback1();

        textMessage = "";

```

```
        StateInput1 = "off";
        inputSent1 = true;

    }
    else {
        inputSent1 = false;
        break;
    }
}

while (true)
{ reading2 = digitalRead(11);
  if (reading2 == LOW)
  { Serial.println("Input 2 off");
    if (inputSent2 == true)
        break;
    SendFeedback2();
    textMessage = "";
    StateInput2 = "off";
    inputSent2 = true;

  }
  else {
      inputSent2 = false;
      break;
  }

}

while (true)
{ reading3 = digitalRead(12);
  if (reading3 == LOW)
```

```
{ Serial.println("Input 3 off");
  if (inputSent3 == true)
    break;
  SendFeedback3();
  textMessage = "";
  StateInput3 = "off";
  inputSent3 = true;

}
else {
  inputSent3 = false;
  break;
}

}

while (true)
{ reading4 = digitalRead(13);
  if (reading4 == LOW)
  { Serial.println("Input 4 off");
    if (inputSent4 == true)
      break;
    SendFeedback4();
    textMessage = "";
    StateInput4 = "off";
    inputSent4 = true;

  }
  else {
    inputSent4 = false;
    break;
  }
}
```

```

    }

}

// Function that sends SMS
void SendStatusSMS(String ost1, String ost2, String ost3, String ost4)

{
    Serial.println("Sending SMS...");
    SIM808.print("AT+CMGF=1\r");
    delay(100);
    //replace the Xs with the user's number including the international key
    SIM808.print("AT+CMGS=\"+XXXXXXXXXXXXX\"\r");
    delay(500);
    SIM808.println("OutputState1:");
    SIM808.println(ost1);
    SIM808.println("OutputState2:");
    SIM808.println(ost2);
    SIM808.println("OutputState3:");
    SIM808.println(ost3);
    SIM808.println("OutputState4:");
    SIM808.println(ost4);

    delay(500);
    SIM808.print((char)26); // (required according to the datasheet)
    delay(500);
    SIM808.println();
    Serial.println("Text Sent.");
    delay(500);
}

```

```
// Sending feedback of commands
void SendFeedbackSMS( )
{
    Serial.println("Sending SMS...");
    SIM808.print("AT+CMGF=1\r");
    delay(100);
    SIM808.print("AT+CMGS=\"+XXXXXXXXXXXXX\"\\r");
    delay(500);
    SIM808.print("Command done!!");
    delay(500);
    SIM808.print((char)26);
    delay(500);
    SIM808.println();
    Serial.println("Text Sent.");
    delay(500);
}
```

```
// Sending feedback for changes in inputs
```

```
void SendFeedback1( )
{
    Serial.println("Sending SMS...");
    SIM808.print("AT+CMGF=1\r");
    delay(100);
    SIM808.print("AT+CMGS=\"+XXXXXXXXXXXXX\"\\r");
    delay(500);
    SIM808.print("input 1 off");
    delay(500);
    SIM808.print((char)26);
    delay(500);
    SIM808.println();
}
```

```
    Serial.println("Text Sent.");
    delay(500);
}

void SendFeedback2( )
{
    Serial.println("Sending SMS...");
    SIM808.print("AT+CMGF=1\r");
    delay(100);
    SIM808.print("AT+CMGS=\"+XXXXXXXXXXXXX\"\\r");
    delay(500);
    SIM808.print("input 2 off");
    delay(500);
    SIM808.print((char)26);
    delay(500);
    SIM808.println();
    Serial.println("Text Sent.");
    delay(500);
}

void SendFeedback3( )
{
    Serial.println("Sending SMS...");
    SIM808.print("AT+CMGF=1\r");
    delay(100);
    SIM808.print("AT+CMGS=\"+XXXXXXXXXXXXX\"\\r");
    delay(500);
    SIM808.print("input 3 off");
    delay(500);
    SIM808.print((char)26);
    delay(500);
    SIM808.println();
    Serial.println("Text Sent.");
    delay(500);
}
```

```
}

void SendFeedback4( )
{
    Serial.println("Sending SMS...");
    SIM808.print("AT+CMGF=1\r");
    delay(100);
    SIM808.print("AT+CMGS=\"+XXXXXXXXXXXXX\"\r");
    delay(500);
    SIM808.print("input 4 off");
    delay(500);
    SIM808.print((char)26);
    delay(500);
    SIM808.println();
    Serial.println("Text Sent.");
    delay(500);
}
```