#include <SoftwareSerial.h>

SoftwareSerial debug(3, 2); // RX, TX

#include <LiquidCrystal.h>

const int rs = 8, en = 14, d4 = 12, d5 = 11, d6 = 10, d7 = 9;

LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

int L\_level;

int R\_level;

int S\_level;

int count;

void setup()

{

debug.begin(9600);

Serial.begin(9600);

lcd.begin(16, 2);

delay(100);

lcd.print(" SOLAR ");

lcd.print(" TRACKER ");

delay(1500);

Serial.println("AT");

delay(700);

Serial.println("ATE1");

delay(700);

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

delay(700);

}

void loop()

{

L\_level = AnalogRead(0);

R\_level = AnalogRead(1);

S\_level = AnalogRead(2);

if(L\_level<100);

{

rotate\_panel\_L();

send\_msg="Panel Rotated Left";

SerialsendSMS(send\_msg);

}

else if(R\_level<100);

{

rotate\_panel\_R();

send\_msg="Panel Rotated Right";

SerialsendSMS(send\_msg);

}

else if(S\_level<100);

{

rotate\_panel\_S();

send\_msg="Panel Rotated Straight";

SerialsendSMS(send\_msg);

}

get\_battery\_V();

lcd.print(battery\_V);

count=count+1;

delay(100);

if(count>=200)

{

clean\_panel();

count=0;

}

}

void SerialsendSMS(char \*number1)

{

Serial.flush();

delay(500);

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

delay(1500);

Serial.print("AT+CMGS=");

Serial.print((char)34);

Serial.print(number1);

Serial.println((char)34);

delay(500);

Serial.print(send\_msg);

delay(2500);

Serial.print((char)26);

delay(3000);

}

/\* Function for reading the analog pins for current transformer output \*/

int AnalogRead(int readPin)

{

int temp\_V;

temp\_V = analogRead(readPin);

delay(150);

return temp\_V;

}

void upload\_data(char \*data)

{

Serial.println("AT+SAPBR=3,1,\"Contype\",\"GPRS\"\r");

Verify.println(get\_response(5000));

Serial.println("AT+SAPBR=3,1,\"APN\",\"www\"\r");

Verify.println(get\_response(5000));

Serial.println("AT+SAPBR=1,1\r");

Verify.println(get\_response(5000));

Serial.println("AT+HTTPINIT\r");

Verify.println(get\_response(5000));

Serial.print("AT+HTTPPARA=\"URL\",\"serverSOLAR.000webhostapp.com/Project10/index.php?data=");

delay(5000);

Serial.print(data);

Serial.println("\"\r");

Serial.println("AT+HTTPACTION=0\r");

Verify.println(get\_response(15000));

Serial.println("AT+HTTPTERM");

Verify.println(get\_response(500));

}

void clean\_panel()

{

digitalWrite(PUMP,HIGH);

for(int k=0;k<5;k++)

{

delay(1000);

}

digitalWrite(PUMP,LOW);

}

void rotate\_panel\_L()

{

if(L\_limit)

{

while(!L\_limit)

{

digitalWrite(M1,LOW);

digitalWrite(M2,HIGH);

}

}

delay(10);

digitalWrite(M1,LOW);

digitalWrite(M2,LOW);

}

void rotate\_panel\_R()

{

if(R\_limit)

{

while(!R\_limit)

{

digitalWrite(M1,HIGH);

digitalWrite(M2,LOW);

}

}

delay(10);

digitalWrite(M1,LOW);

digitalWrite(M2,LOW);

}

void rotate\_panel\_S()

{

if(L\_limit)

{

digitalWrite(M1,HIGH);

digitalWrite(M2,LOW);

delay(1500);

digitalWrite(M1,LOW);

digitalWrite(M2,LOW);

}

else if(R\_limit)

{

digitalWrite(M1,LOW);

digitalWrite(M2,HIGH);

delay(1500);

digitalWrite(M1,LOW);

digitalWrite(M2,LOW);

}

}