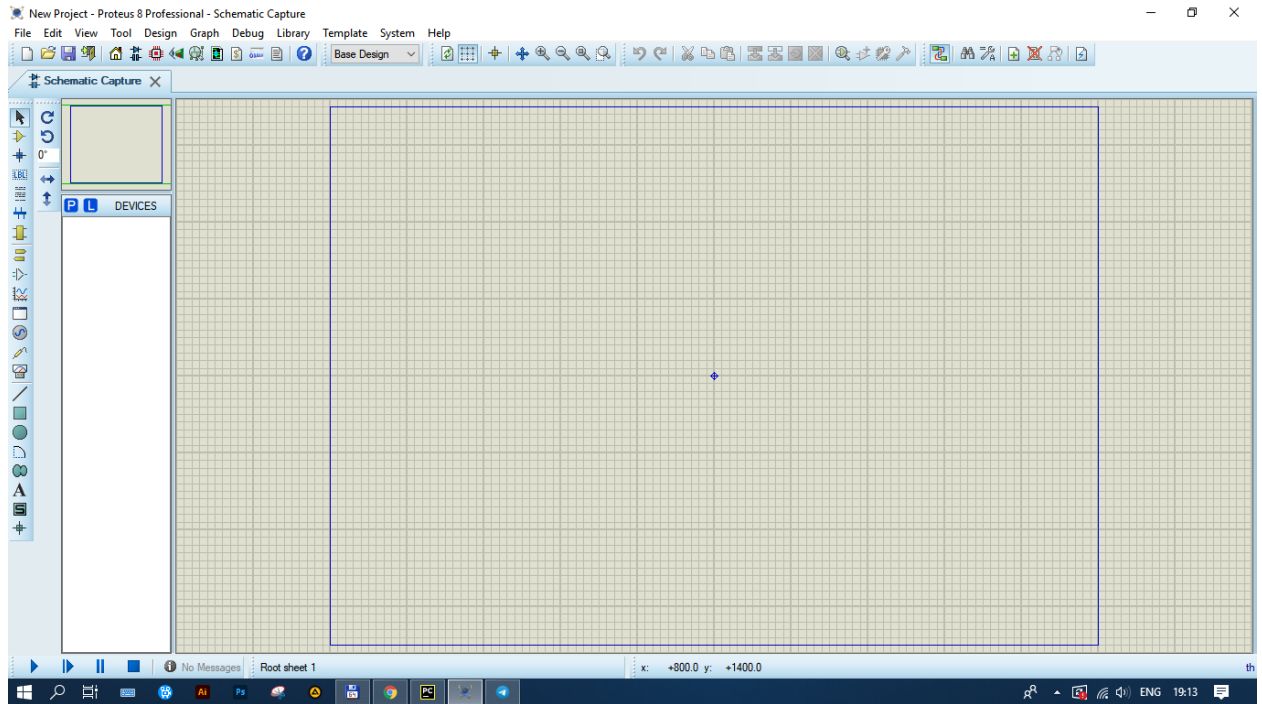



Mavzu: Smart eshik qulfi.

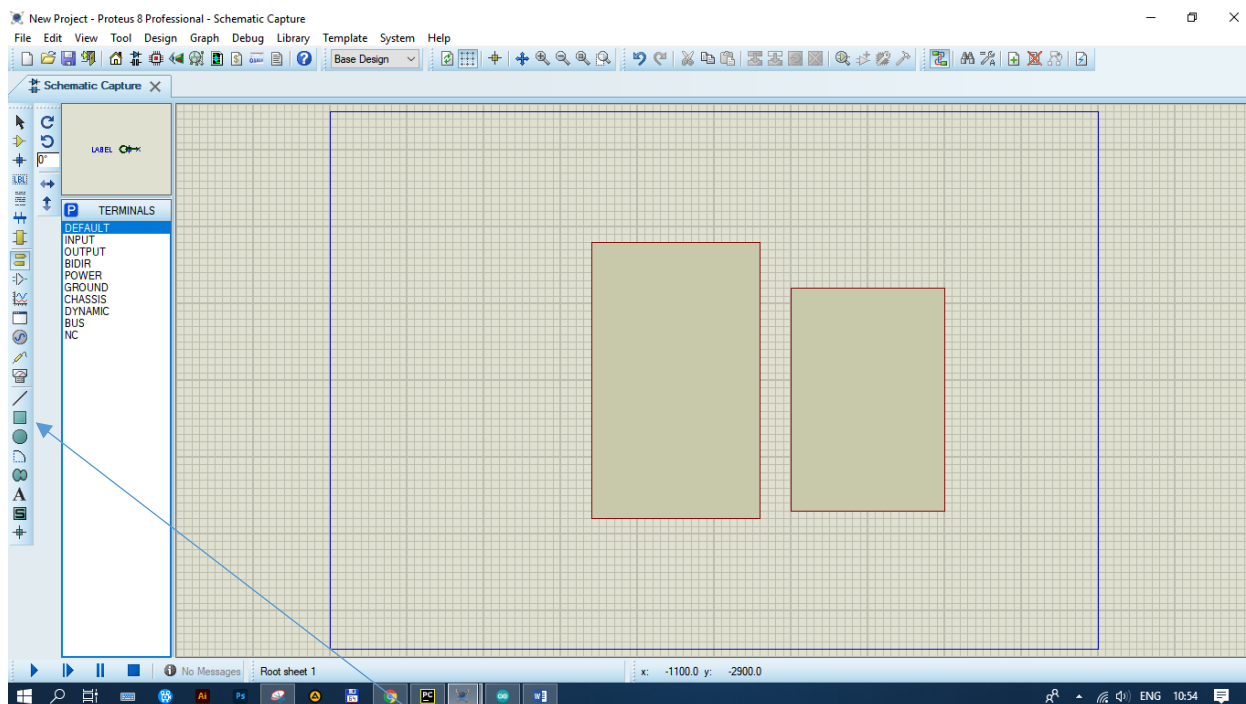


6.1 – rasm. Bosh oyna.

P foydali komponentlarni shu joydan tanlab olamiz. Bizga kerak bo'ladigan komponentlar quyidagilar:

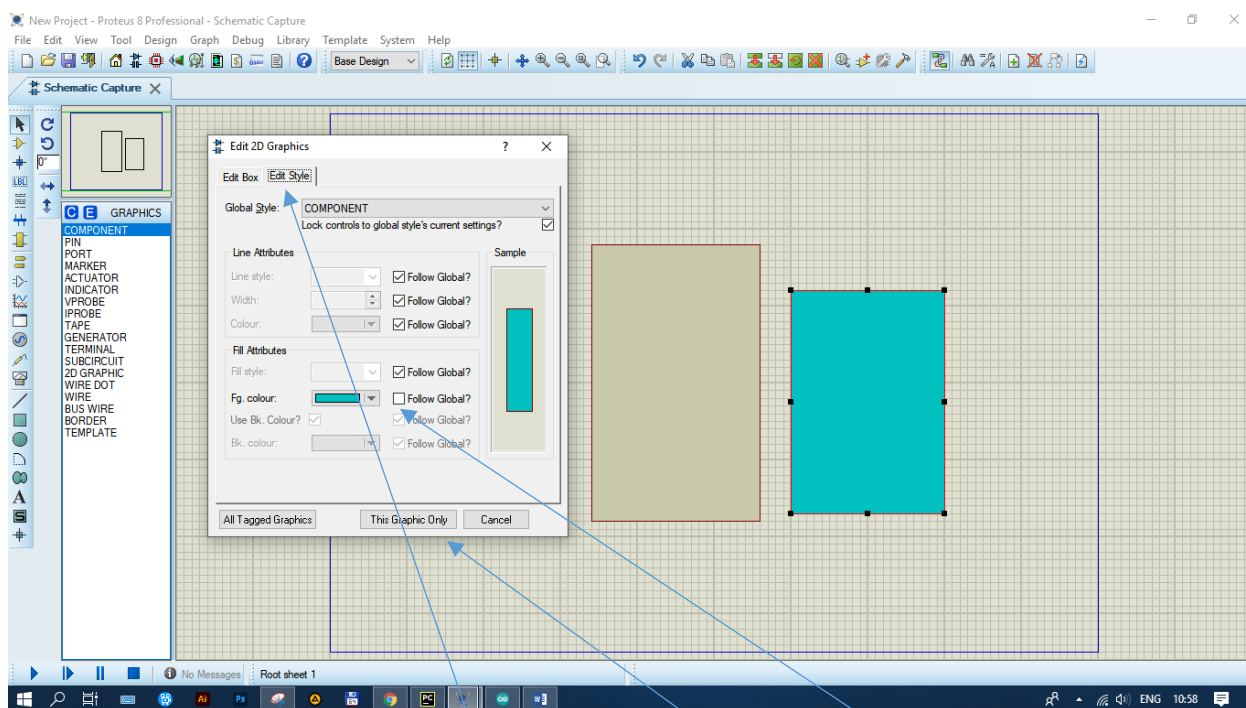
- Ikkita qarshilik. (Res 220R).
- Arduino Uno.
- Bluetooth.
- Avto uzub-ulagich (Relay).
- Ikkita tranzistor. (BC547BP).
- Signal tarqatgich (Buzzer).
- Monitor (LM044L).
- Micro sxema (PCF8574A).
- Diod (1N4001).
- 12 voltlik doimiy quvvat manbai (DC).
- Quvvat yutish uchun yer (Ground).
- O'zimiz yig'gan suv idish.
- Har xil ko'ngilsiz holatlardan saqlash uchun o'chirish tugmasi (Sump Tank)

12 voltli doimiy manba va Ground  shu bo'limdan olinadi. Komponentlarni birin ketin ishchi oynaga o'tkazib loyihamni yig'amiz. Suv idish va Sump Tankni o'zimiz yasaymiz.



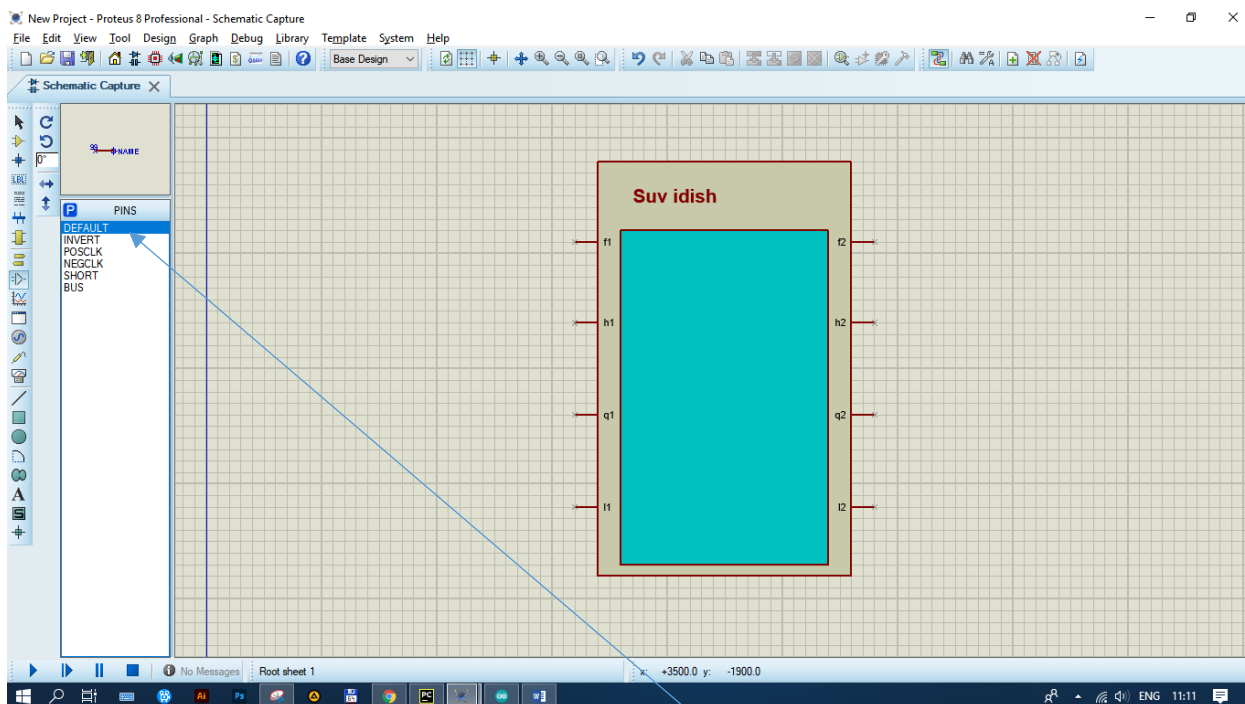
6.2 – rasm. Suv idishni tayyorlash.

Avval ikkita bir biridan kattaligida farq qiladigan to'rtburchaklar chizib olamiz. Bu to'rtburchaklarni chizishda  shu bo'limdan foydalanamiz.




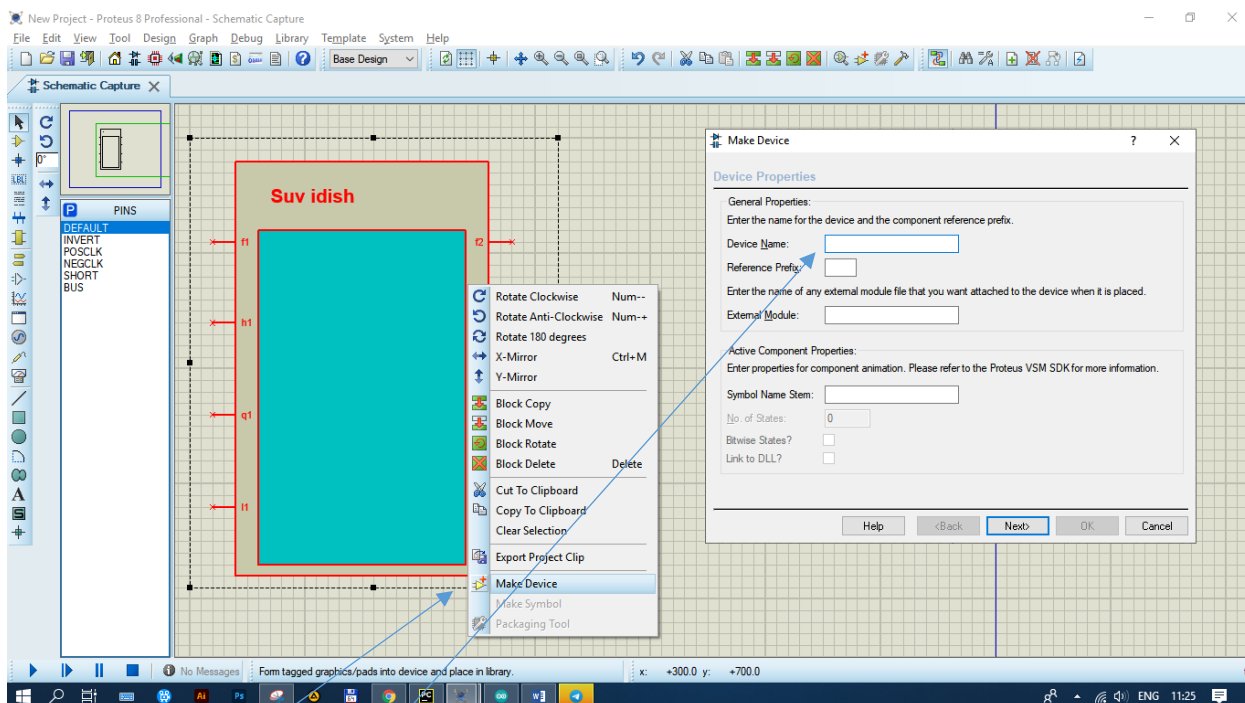
6.3 – rasm. To'rtburchakga rang berish.

Kichik to'rtburchakning sozlamalariga kiramiz. Buning uchun to'rtburchakning chizig'ining ustiga sichqonchaning chap tarafini ikki marta bosish kifoya. Sozlamalar ischidagi **Edit Style** bo'limiga o'tamiz. U yerdagi **Fg. Color** yachekasiga rang beramiz. Sozlamalarni saqlash uchun **This Graphic Only** tugmasini bosasiz.



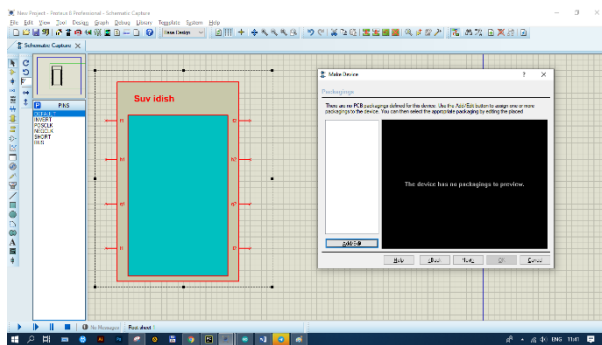
6.4 – rasm. Suv idish ko'rinishi,

Ikkita to'rtburchani ustma-ust qo'yasiz.  Shu bo'limdan foydalanib suv miqdorini aniqlab turuvchi qurulmani joylashishini **Default** oyoqchalar yordmida belgilab chiqmiz va har bir oyoqchalarni nomlab chiqamiz. Suv idish nomini **A** shu bo'lim orqali to'rtburchakning tepasiga yozib qo'yamiz.

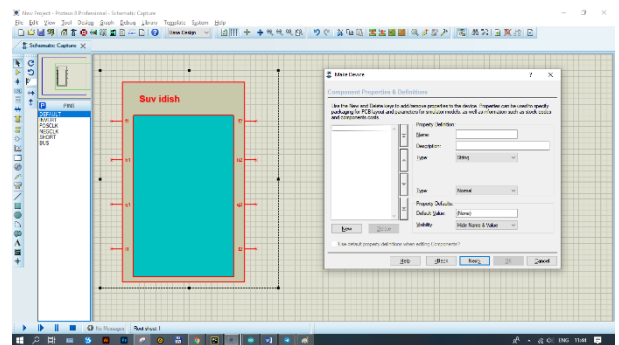


6.5 – rasm. Suv idishni komponentlar qatoriga qo'shish.

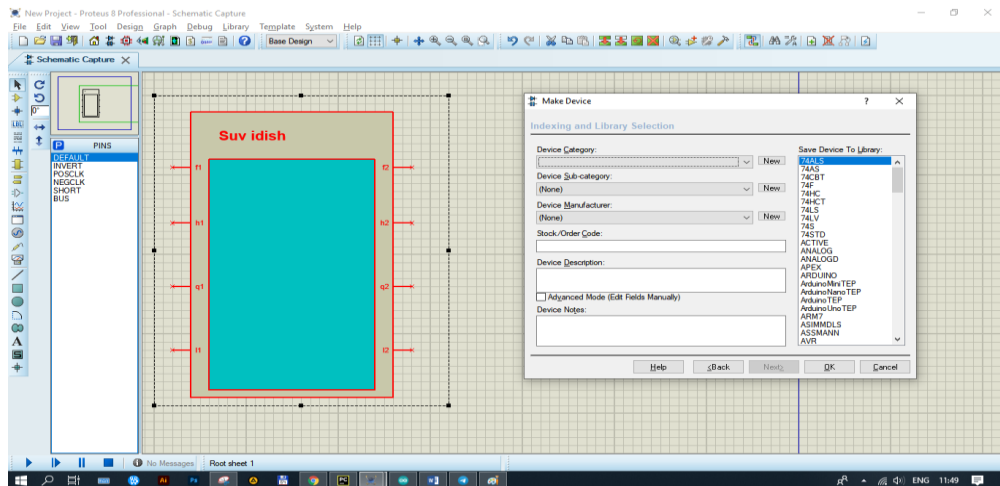
Suv idishni to'liq belgilab sichqonchanning o'ng tarafini bosasiz va **Make Device** ga kirasiz. U bo'limda yangi qo'shayotgan loyihangizni component sifatidagi nomini kiritasiz. Shundan so'ng **Next** tugmasini bosasiz.



6.6 – rasm.

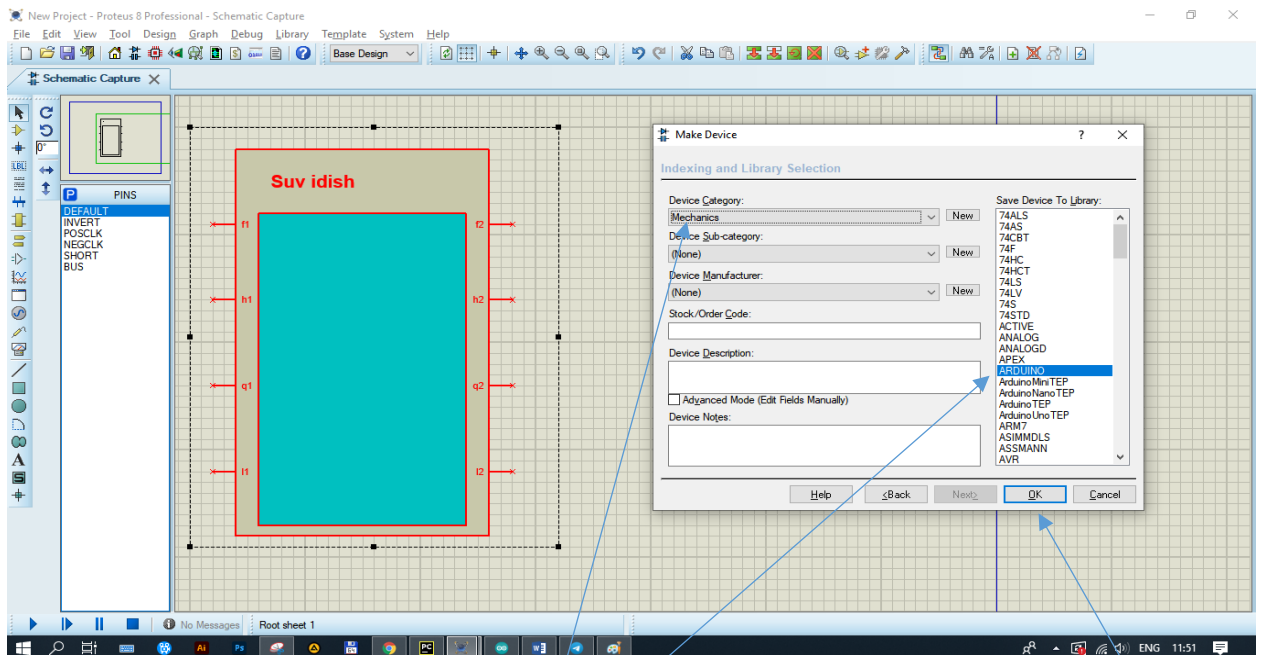


6.7 – rasm.



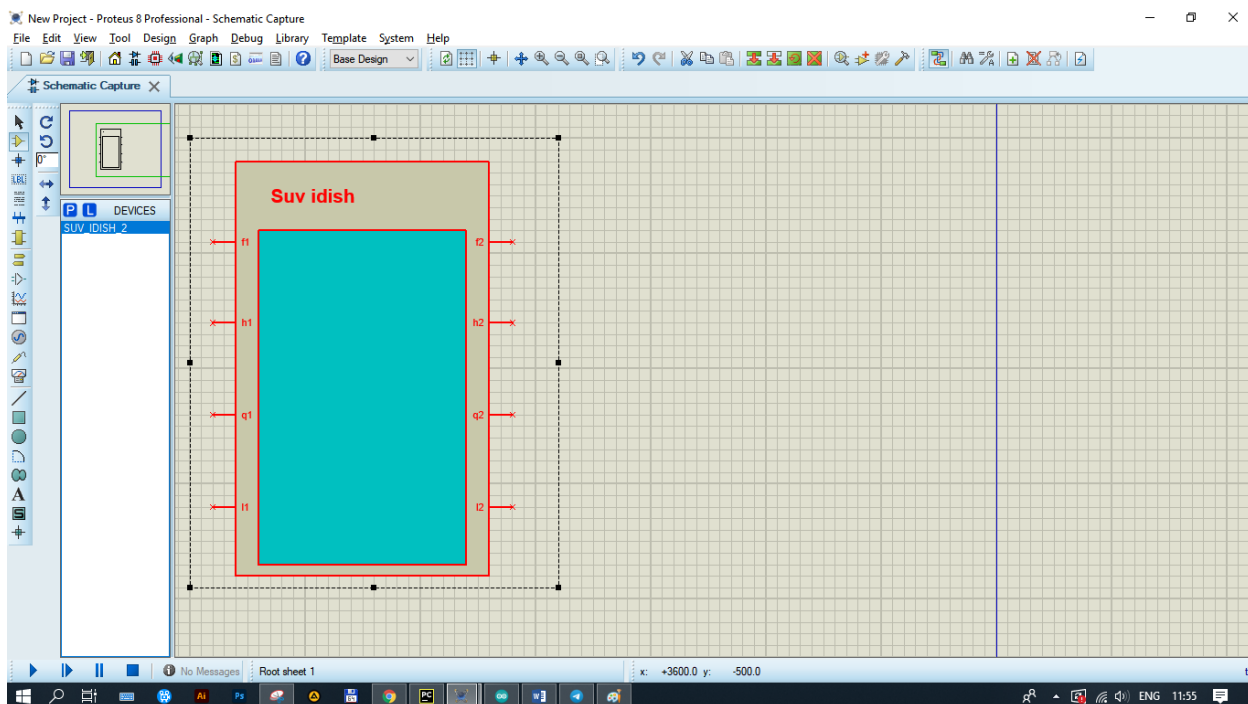
6.8 – rasm.

6.6 – rasm, 6.7 – rasm va 6.8 – rasmdagi holatda shunchaki **Next** tugmasini bosib davom etamiz.



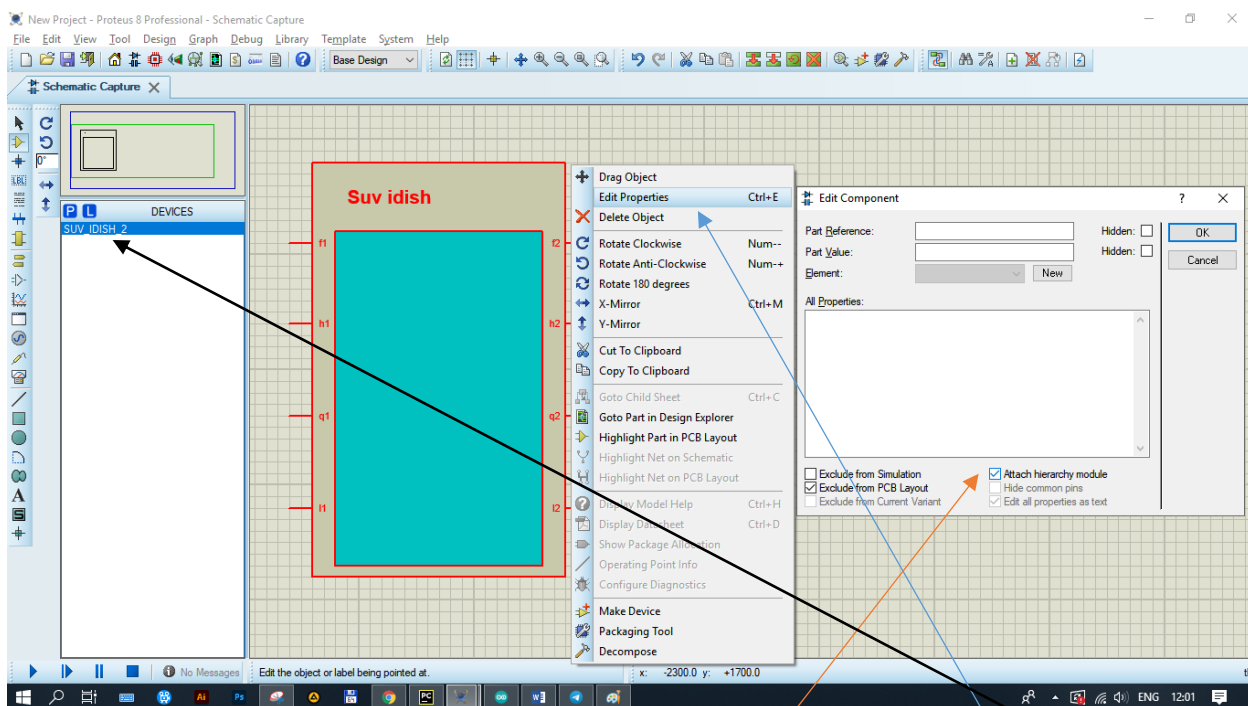
6.9 – rasm. Yig'ilgan sxema turini belgilash.

Yig'gan sxemazini qaysi turga mansub ekanini belgilab saqlaymi. Saqlash uchun OK tugmasini bosasiz.



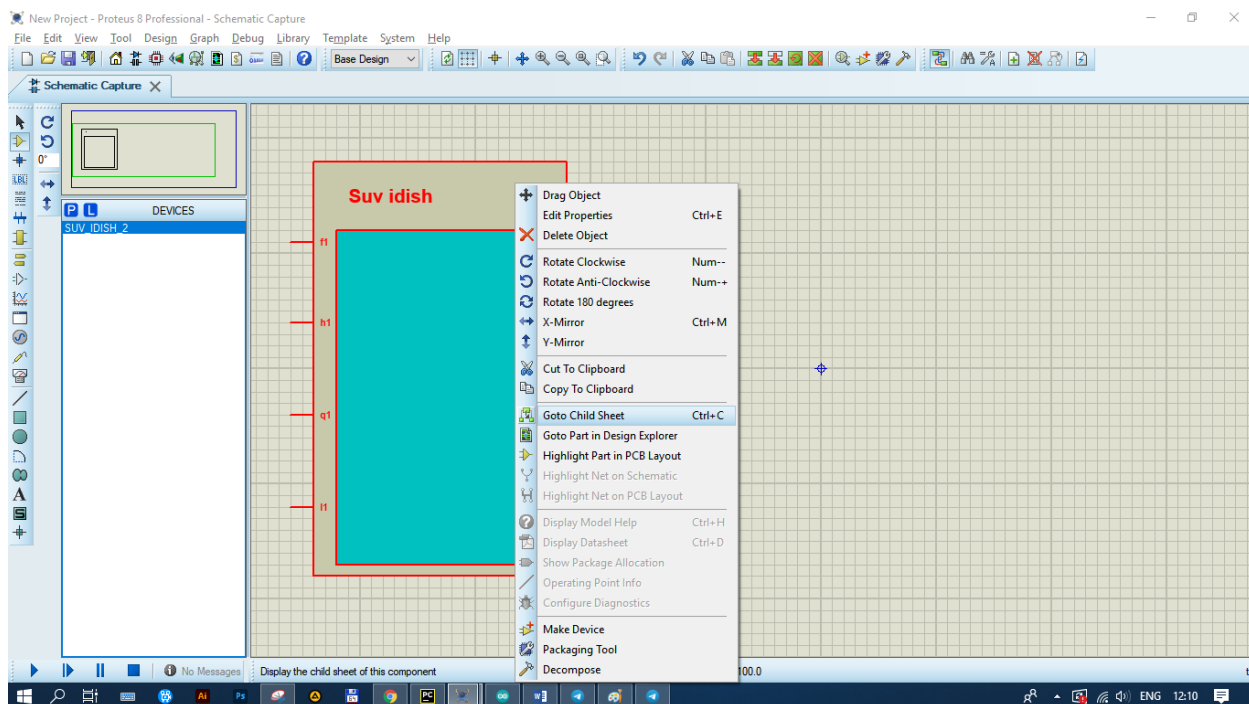
6.10 – rasm. Komponentlar qatoriga qo'shildi.

Qo'shgan komponentingizni vazifasi nima ekanligini belgilashimiz kerak.



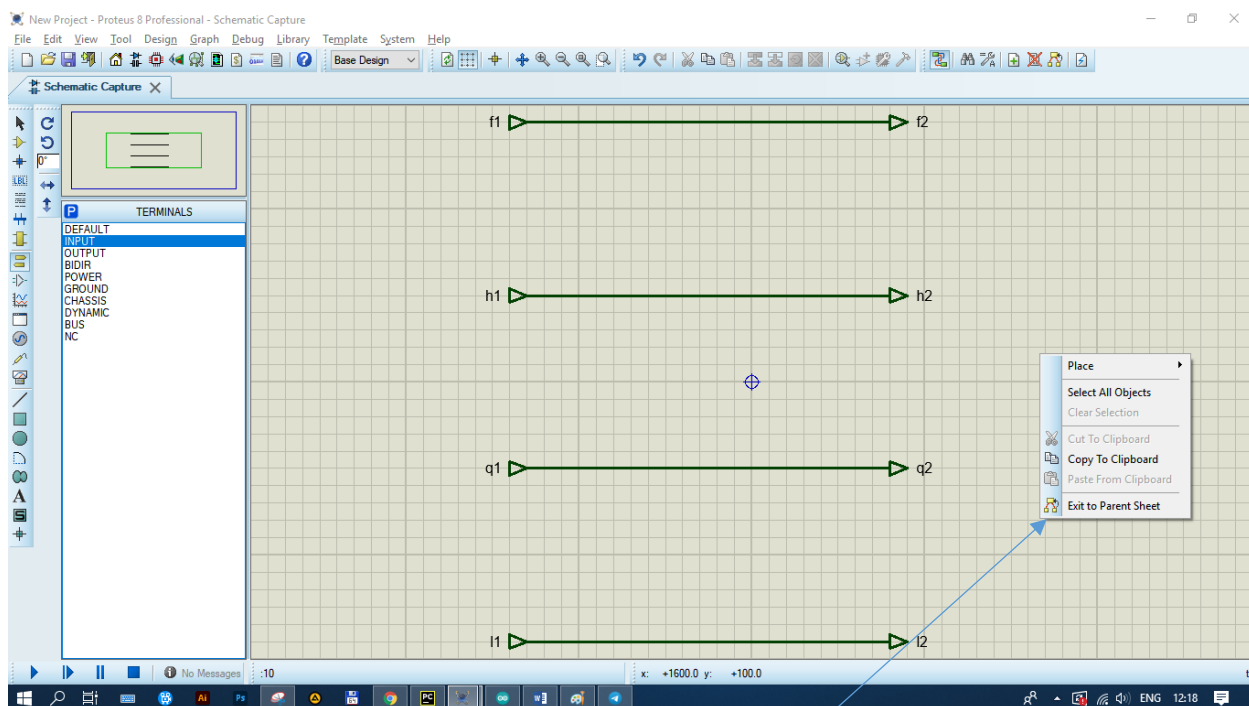
6.11 – rasm. Suv idish sozlamalari.

Yig'gan loyihamizni hammasini tanlab o'chirib yuboramiz. Keyin **P** bo'limidan component sifatida qo'shgan SUV_IDISH ni qidirib topib, ishchi oynaga qo'yamiz. Ishchi oynadagi o'zimiz tuzgan komponentni **Edit Properties** bo'limiga kirib **Attach hierarchy module** funksiyasini yoqib qo'yamiz.



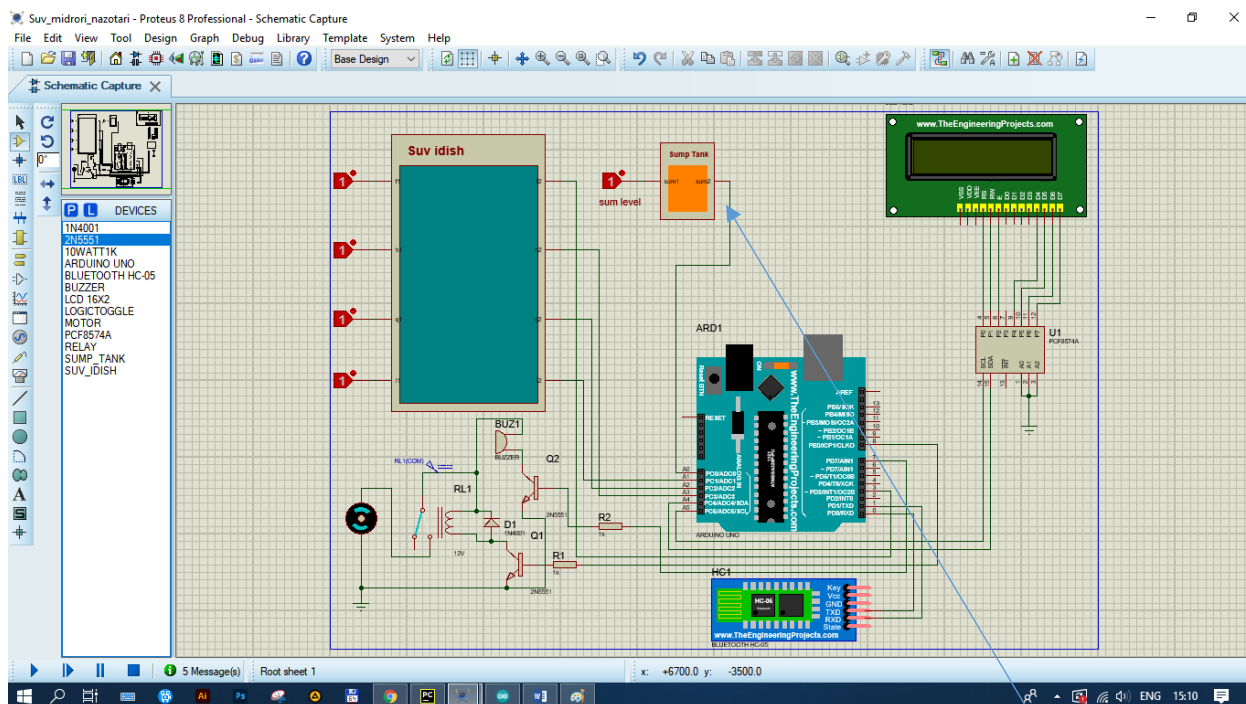
6.12 – rasm. Ishlash funksiyasini kiritish.

Komponentimizning **Goto Child Sheet** kiramiz.



6.13 – rasm. Komponentning funksiyasini belgilash.

O'zimiz tuzgan suv idish funksiyasini 6.13 – rasmda ko'rsatilganidek qilib tuzamiz. Shundan so'ng **Exit to Parent Sheet** tugmasini bosib saqlaymiz. Shundan so'ng bizning suvning miqdorini aniqlovchi qurilmamiz ham tayyor bo'ldi.



6.14 – rasm. Loyiha yig' ilgan holati.

Loyiha sxemasi to'liq yig' ilgan holati shunday. Bu yerdagi **Sump Tank** ham suv idish kabi yig' iladi. Uni ham o'zingiz komponentlar qatoriga qo'shasiz. Faqat uning kirish va chiqish oyoqchalari bittalik bo'ladi. Sump Tank ning maqsadi har xil ko'ngilsiz holatlarni sezib unga qarshi chora ko'rish uchun xizmat qiladi. Bu holatlar qanday turlarda bo'lishlarini o'zingiz hohlagan usullarni qo'shib chiqasiz.

```

water_level_test

#include <Wire.h>
#include <SoftwareSerial.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x38,16,2);

int sump=A0;
int qut=A1;
int hlf=A2;
int thf=A3;
int ful=3;
int motor=8;
int buz=7;
int s;
int s1 = 1, s2 = 1;
int q;
int h;
int t;
int f;
int i; //motor status flag
int b=0; //buzzer flag
int m=0; //motor flag
int c=0; //sump flag
String BT_data; //Bluetooth

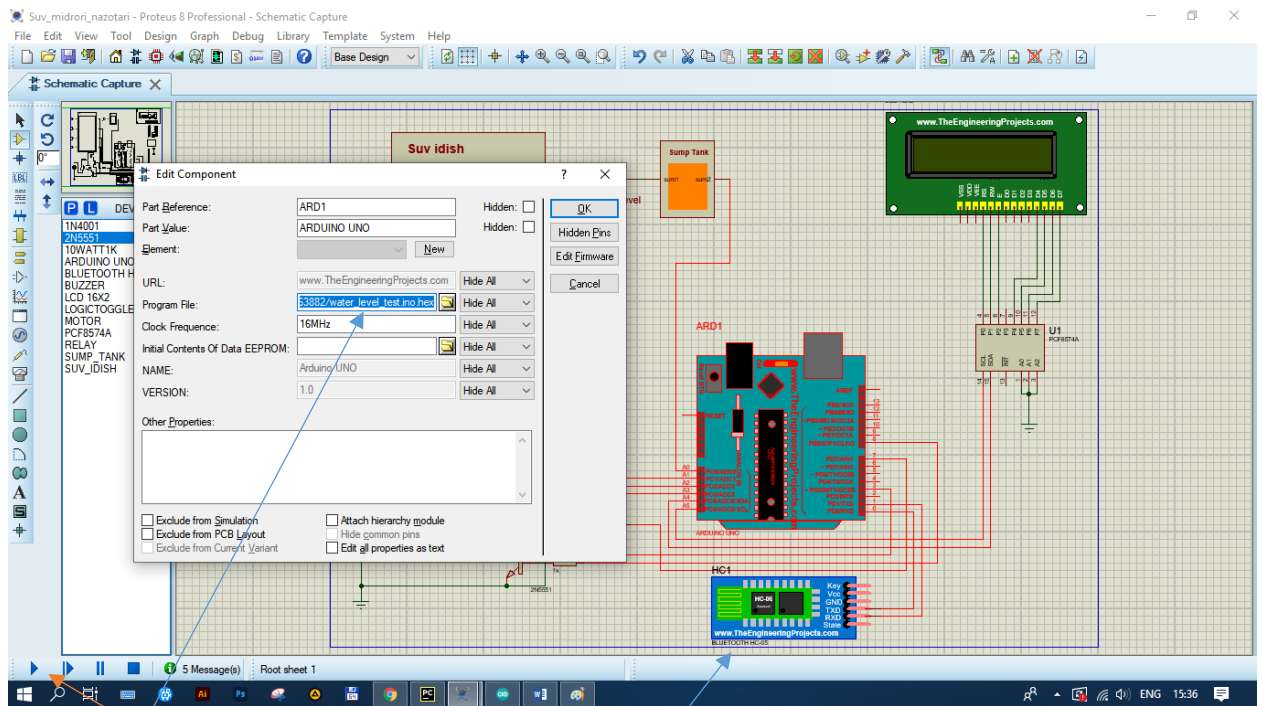
void setup() {
  lcd.begin(16,2);
}

void loop() {
  // ... (rest of the code) ...
}

```

6.15 – rasm. Dastur kodlarini mashina tiliga o'tkazilgan fayl manzili.

Qurulma tiliga o'tkazilgan dastur ko'dining manzilidan nusxa olamiz. U faylning qisqartirmasi **hex** so'zi bilan yakunlanadi.



6.16 – rasm. Arduinoga fayl manzilini kiritish.

Proteusdagi Arduino Uno qurulmasiga mashina tiliga o'tkazilgan fayl manzilini kiritib saqlaymiz. Bluetooth qurulmasiga ham o'ziga tegishli port raqamini kiritib o'tamiz. Shundan so'ng loyihani ishga tushunsak ham bo'ladi. Loyihani ishga tushurish uchun **Start** tugmasini bosamiz.

Dastur kodi:

```
#include <Wire.h>
```

```
#include <SoftwareSerial.h>
```

```
#include <LiquidCrystal_I2C.h>
```

```
LiquidCrystal_I2C lcd(0x38,16,2);
```

```
int sump=A0;
```

```
int qut=A1;
```

```
int hlf=A2;
```

```
int thf=A3;
```

```
int ful=3;
```

```
int motor=8;
```

```
int buz=7;
```



```
int s;  
int s1 = 1, s2 = 1;  
int q;  
int h;  
int t;  
int f;  
int i;    //motor status flag  
int b=0;  //buzzer flag  
int m=0;  //motor flag  
int c=0;  //sump flag  
String BT_data; //Bluetooth
```

```
void setup() {
```

```
    lcd.init();
```

```
    lcd.backlight();
```

```
    Serial.begin(9600);
```

```
    pinMode(qut,INPUT);
```

```
    pinMode(hlf,INPUT);
```

```
    pinMode(qut,INPUT);
```

```
    pinMode(ful,INPUT);
```

```
    pinMode(sump,INPUT);
```

```
    pinMode(motor,OUTPUT);
```

```
    pinMode(buz,OUTPUT);
```

```
    digitalWrite(buz,LOW);
```

```
}
```

```
void loop() {
```

```

if (Serial.available()) {
    BT_data = Serial.readString();
    if (BT_data == "1") {
        s1 = 1;
    } else if (BT_data == "0") {
        s1 = 0;
    }
} else {
    s2 = digitalRead(sump);
}
if ((s1 == 1 && s2 == 1) || (s1 == 0 && s2 == 0)) {
    s = 1;
} else {
    s = 0;
}
i = digitalRead(motor);
q = digitalRead(qut);
h = digitalRead(hlf);
t = digitalRead(thf);
f = digitalRead(ful);
lcd.clear();

if(f == 1 && t == 1 && h == 1 && q == 1 ) {
    lcd.setCursor(0,0);
    lcd.print(char(219));
    lcd.print(char(219));
    lcd.print(char(219));
    lcd.print(char(219));
    lcd.setCursor(5,0);

```

```
    lcd.print("FULL");
    Serial.println("FULL");
    m=0;
    b=0;
}
else {
    if(f==0 && t==1 && h==1 && q==1) {
        lcd.setCursor(0,0);
        lcd.print(char(219));
        lcd.print(char(219));
        lcd.print(char(219));
        lcd.print("_");
        lcd.setCursor(5,0);
        lcd.print("3/4th");
        Serial.println("3/4th");
        b=0;
    }
    else {
        if(f==0 && t==0 && h==1 && q==1) {
            lcd.setCursor(0,0);
            lcd.print(char(219));
            lcd.print(char(219));
            lcd.print("_");
            lcd.print("_");
            lcd.setCursor(5,0);
            lcd.print("HALF");
            Serial.println("HALF");
            m=1;
            b=0;
```

```

}
else
if(f ==0 && t ==0 && h ==0 && q ==1) {
    lcd.setCursor(0,0);
    lcd.print(char(219));
    lcd.print("_");
    lcd.print("_");
    lcd.print("_");
    lcd.setCursor(5,0);
    lcd.print("1/4th");
    Serial.println("1/4th");
    b=0;
}
else {
    if(f == 0 && t == 0 && h == 0 && q == 0) {
        lcd.setCursor(0,0);
        lcd.print("_");
        lcd.print("_");
        lcd.print("_");
        lcd.print("_");
        lcd.setCursor(5,0);
        lcd.print("LOW");
        Serial.println("LOW");
        b=0;
    }
    else {
        digitalWrite(motor,LOW);
        lcd.setCursor(0,0);
        lcd.print("ERROR!");
    }
}

```

```

        Serial.println("ERROR");
        b = 1;
    }
}
}
}
}
if(i == HIGH) {
    lcd.setCursor(0,1);
    lcd.print("Pump ON");
    Serial.println("Pump ON");
}
else {
    lcd.setCursor(0,1);
    lcd.print("Pump OFF");
    Serial.println("Pump OFF");
}

if(s == 1 && m == 1) {
    digitalWrite(motor,HIGH);
}
if(s == 0) {
    digitalWrite(motor,LOW);
    lcd.setCursor(11,0);
    lcd.print("Low");
    lcd.setCursor(11,1);
    lcd.print("Sump");
    Serial.println("\tSump:  Low");
    c=1;
}

```

```
if(s == 1) {  
    c=0;  
}  
if(m == 0) {  
    digitalWrite(motor,LOW);  
}  
if(b == 1 || c == 1) {  
    digitalWrite(buz,HIGH);  
    delay(500);  
    digitalWrite(buz,LOW);  
}  
else {  
    digitalWrite(buz,LOW);  
}  
delay(100);  
lcd.clear();  
}
```