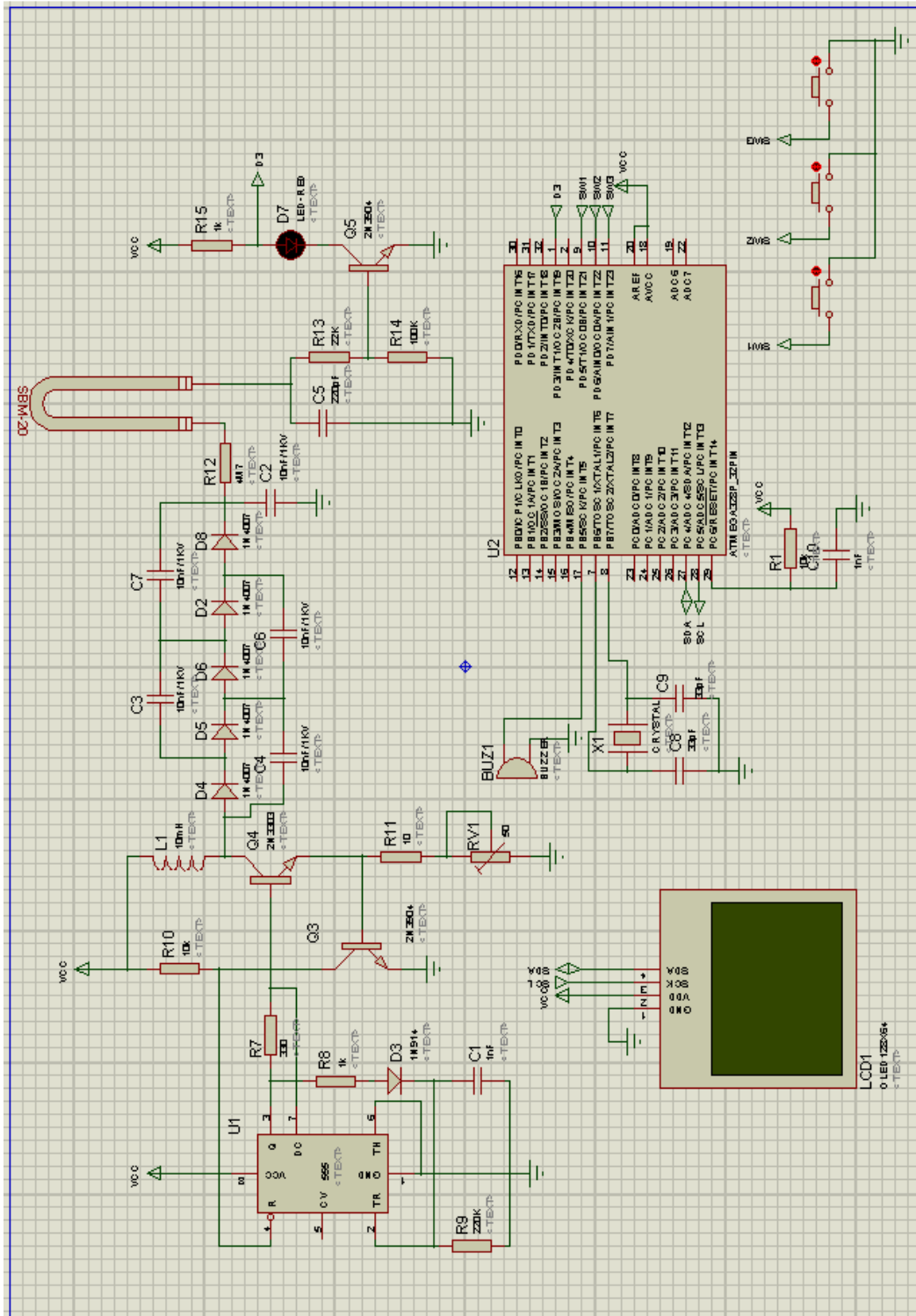
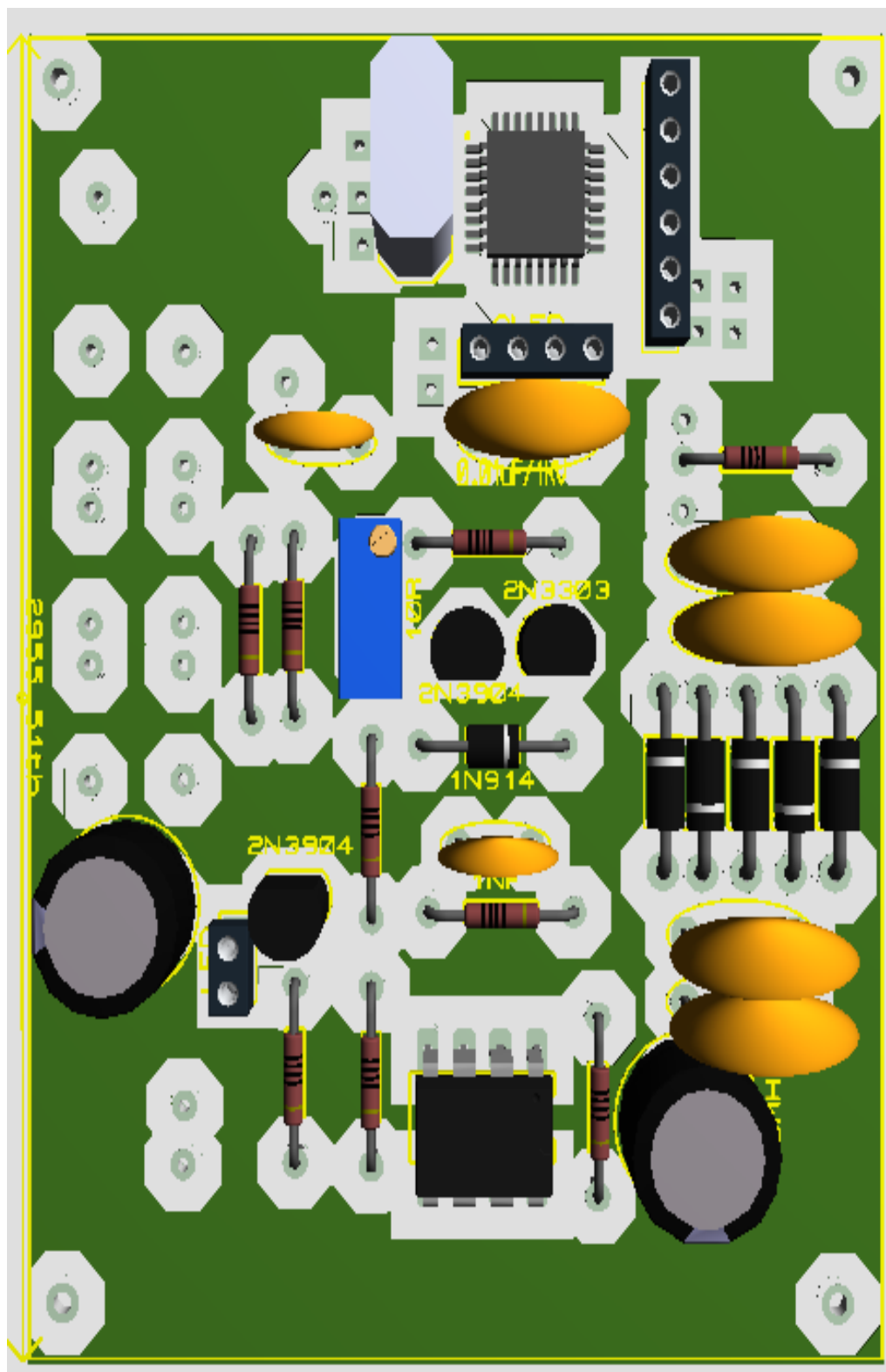


# LAMPIRAN

## A. Rangkaian Sistem Alat Lengkap





**Board Rangkaian Modul Tugas Akhir**

## B. Listing Program Keseluruhan

```
#include "gambar.h"
#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include <TimerOne.h>
#include <EEPROM.h>

#define OLED_RESET 4
Adafruit_SSD1306 display(OLED_RESET);

#define NUMFLAKES 10
#define XPOS 0
#define YPOS 1
#define DELTAY 2

#define LOGO16_GLCD_HEIGHT 16
#define LOGO16_GLCD_WIDTH 16

//read internal voltage
long readVcc() {
  long result;
  // Read 1.1V reference against AVcc
  ADMUX = _BV(REFS0) | _BV(MUX3) | _BV(MUX2) | _BV(MUX1);
  delay(2); // Wait for Vref to settle
  ADCSRA |= _BV(ADSC); // Convert
  while (bit_is_set(ADCSRA, ADSC));
  result = ADCL;
  result |= ADCH << 8;
  result = 1126400L / result; // Back-calculate AVcc in mV
  return result;
}
```

---

```

}

//tampilan indikator tegangan dan baterai

boolean flag_mute=false;
void bat_indicator(){
int h;
double curvolt;

curvolt = double( readVcc() ) / 1000;
display.setTextSize(1);
display.setTextColor(WHITE);
display.setCursor(0,4);
display.print("BAT:");
display.print(curvolt);

display.print(" V");
if (flag_mute==true){
display.print(" MUTE");}
if (curvolt>4.21){h=11;}
else {h=(curvolt-3.2)*10.0/0.9;if(h<0){h=0;} if (h>10){h=10;}}

switch(h){
case 0: display.drawBitmap(104, 0, batery, 24, 12, 1); break;
case 1: display.drawBitmap(104, 0, batery1, 24, 12, 1); break;
case 2: display.drawBitmap(104, 0, batery2, 24, 12, 1); break;
case 3: display.drawBitmap(104, 0, batery3, 24, 12, 1); break;
case 4: display.drawBitmap(104, 0, batery4, 24, 12, 1); break;
case 5: display.drawBitmap(104, 0, batery5, 24, 12, 1); break;
case 6: display.drawBitmap(104, 0, batery6, 24, 12, 1); break;

```

```

case 7: display.drawBitmap(104, 0, batery7, 24, 12, 1); break;
case 8: display.drawBitmap(104, 0, batery8, 24, 12, 1); break;
case 9: display.drawBitmap(104, 0, batery9, 24, 12, 1); break;
case 10: display.drawBitmap(104, 0, batery10, 24, 12, 1); break;
case 11: display.drawBitmap(104, 0, batery11, 24, 12, 1); break;
}
display.display();
}

```

```
//Perhitungan Laju Dosis dan Dosis
```

```

#define buz 13 // buzer di pin13
#define sw1 5 // button 1
#define sw2 6 // button 2
#define sw3 7 // button 3
#define CONV_FACTOR 0.0057 // Conversion Factor SBM-20
float usievert, usiv;
long cpm;
unsigned int cps;

void countPulse(){
  cps++;
}
int tick=0;
void calculatecpm(){
  detachInterrupt(1); //nonaktifkan interrupt 1
  cpm=cps;
  usievert=cpm*CONV_FACTOR*60.0*10;
  usiv=usiv + usievert/3600.0;
  tick++;
}

```

```
if (tick>59){ //setiap satu menit sekali disimpan
writeEEPROM(0, usiv);
tick=0; //reset kembali ke awal untuk count
}
cps=0;
attachInterrupt(1,countPulse,FALLING); //aktifkan INT1 terhubung ke sensor high ke low
}

void beep(){
digitalWrite(buz,HIGH);
delay(50);
digitalWrite(buz,LOW);
delay(50);
}

void reset_usiv(){
while(1){
display.clearDisplay();
display.setCursor(0,16);
display.setTextSize(1);
display.println("Reset Dose?");
display.println("Y/N");
bat_indicator();
delay(100);
if (digitalRead(sw2)==LOW){ beep();reseting(); goto kel;}
if (digitalRead(sw3)==LOW){
beep();
display.clearDisplay();
display.setCursor(0,16);
display.setTextSize(1);
.
```

```
.  
display.println("Canceling.....");  
bat_indicator();  
  
goto kel;}  
  
}  
kel:  
delay(1000);  
}  
  
void resetting() {  
int i;  
for (i=0;i<101;i++){  
display.clearDisplay();  
display.setCursor(0,16);  
display.setTextSize(1);  
display.println("Reseting.....");  
display.print(i);  
display.println(" %");  
bat_indicator();  
}  
writeEEPROM(0, 0.000);  
usiv=0; //nilai dosis reset  
beep();  
display.clearDisplay();  
display.setCursor(0,16);  
display.setTextSize(1);  
display.println("Reset Done");  
bat_indicator();  
delay(1000);
```

```

}

void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);
  pinMode(sw1, INPUT_PULLUP);
  pinMode(sw2, INPUT_PULLUP);
  pinMode(sw3, INPUT_PULLUP);
  pinMode(13, OUTPUT);
  // by default, we'll generate the high voltage from the 3.3v line internally! (neat!)
  display.begin(SSD1306_SWITCHCAPVCC, 0x3C); // initialize with the I2C addr 0x3D (for the 128x64)
  // init done
  usiv=readEEPROM(0);

  // Show image buffer on the display hardware.
  // Since the buffer is intialized with an Adafruit splashscreen
  // internally, this will display the splashscreen.
  display.clearDisplay();
  display.drawBitmap(0, 0, logo_1my, 128, 64, 1);
  display.display();
  delay(5000);

  Timer1.initialize(1000000); //buat calculate cpm tiap 1 detik
  Timer1.attachInterrupt(calculatecpm);
  attachInterrupt(1, countPulse, FALLING);
  // Clear the buffer.
  display.clearDisplay();
}

void loop() {

```