

## DAFTAR LAMPIRAN

```
#define vin_pin A1
```

```
#define vout_pin A2
```

```
#define iout_pin A0
```

```
#define boost_pin 5
```

```
#define buck_pin 6
```

```
int raw_vin=0, raw_vout=0, raw_iout=0;
```

```
float Vout_max=13.0, Iout_max=1.0, Vout_min=11.1,  
Iout_min=0.1, Vin_thresold=10.5;
```

```
float Iout_sense;
```

```
float Vout_sense;
```

```
float Vin_sense;
```

```
uint8_t duty_cycle = 25;
```

```
String mode="";
```

```
bool startup=true;
```

```
unsigned int count=0;
```

```
void setup() {
```

```
// put your setup code here, to run once:
```

```
Serial.begin(115200);
```

```
//pinMode(buck_pin,OUTPUT);
```

```
//pinMode(boost_pin,OUTPUT);
```

```
TCCR0B = TCCR0B & 0b11111000 | 0x01;
```

```
analogWrite(buck_pin,255);
```

```
analogWrite(boost_pin,0);
```

```
}
```

```
void loop() {
```

```
if(Serial.available()) {
```

```
String data = Serial.readString();

Vout_max = data.toInt();

Vout_max = Vout_max/10;

Serial.print("Vout_max= ");

Serial.println(Vout_max);
}

// put your main code here, to run repeatedly:

for(int i=0;i<10;i++) {

raw_iout += analogRead(iout_pin)-513;

raw_vin += analogRead(vin_pin);

raw_vout += analogRead(vout_pin);

}

raw_iout=raw_iout/10;
```

```
raw_vout=raw_vout/10;

raw_vin=raw_vin/10;

Iout_sense=float(raw_iout)*0.0586;

Vout_sense=float(raw_vout)*0.046;

Vin_sense=float(raw_vin)*0.046;

if(count>100) {

Serial.print("Vin= ");Serial.println(Vin_sense);

Serial.print("Vout= ");Serial.println(Vout_sense);

Serial.print("Iout= ");Serial.println(Iout_sense);

Serial.print("Duty cycle= ");Serial.println(duty_cycle);

Serial.print("Converter MODE : ");Serial.println(mode);

count=0;

}

if(startup==false) {
```

```
regulate(Iout_sense, Vin_sense, Vout_sense);

auto_cutoff(Iout_sense, Vin_sense, Vout_sense);

}

else {

    soft_start();

}

delay(600);

count++;

}

void regulate(float Iout, float Vin, float Vout) {

if(Vout_max<Vin) {

    mode="";

    mode="Buck mode";
```

```
analogWrite(boost_pin,0);

if((Iout<Iout_max && Iout>Iout_min) && (Vout<Vout_max)) {

    if(duty_cycle<250) {

        duty_cycle+=2;

    }

    analogWrite(buck_pin,255-duty_cycle);

}

else if((Iout>Iout_max) || (Vout>Vout_max)) {

    if(duty_cycle>2) {

        duty_cycle-=2;

    }

    analogWrite(buck_pin,255-duty_cycle);

}

//Serial.print("Duty cycle= ");Serial.println(duty_cycle);
```

```
}
```

```
else if(Vout_max>Vin) {
```

```
mode="";
```

```
mode="Boost mode";
```

```
analogWrite(buck_pin,0);
```

```
if((Iout<Iout_max) && (Vout<Vout_max)) {
```

```
if(duty_cycle<220) {
```

```
duty_cycle+=2;
```

```
}
```

```
analogWrite(boost_pin,duty_cycle);
```

```
}
```

```
else if((Iout>Iout_max) || (Vout>Vout_max)) {
```

```
if(duty_cycle>4) {
```

```
duty_cycle-=2;
```

```
}  
  
analogWrite(boost_pin,duty_cycle);  
  
}  
  
//Serial.print("Duty cycle= ");Serial.println(duty_cycle);  
  
}  
  
}  
  
void auto_cutoff(float Iout,float Vin, float Vout){  
  
if((Vout>=Vout_max) && (Iout<Iout_min) || (Vin<Vin_threshold)) {  
  
analogWrite(boost_pin,0);  
  
analogWrite(buck_pin,255);  
  
Serial.println("Charging Completed.");  
  
delay(64000);
```



```
}  
  
}
```

```
void soft_start() {
```

```
    if(Vout_sense<=Vout_min) {
```

```
        regulate(Iout_sense, Vin_sense, Vout_sense);
```

```
        Serial.print("Vin= ");Serial.println(Vin_sense);
```

```
        Serial.print("Vout= ");Serial.println(Vout_sense);
```

```
        Serial.print("Iout= ");Serial.println(Iout_sense);
```

```
        Serial.print("Duty cycle= ");Serial.println(duty_cycle);
```

```
        Serial.print("Converter MODE : ");Serial.println(mode);
```

```
        Serial.println("Soft Start Activated");
```

```
        delay(64000);
```

```
    }
```

```
else {  
  
startup=false;  
  
}  
  
}
```

