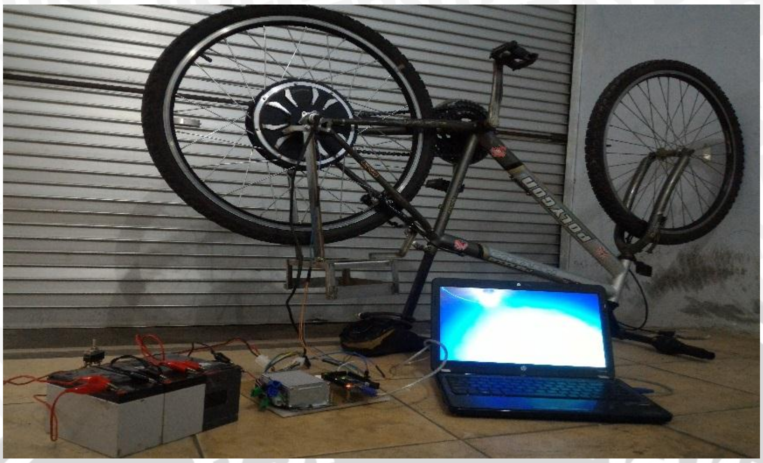
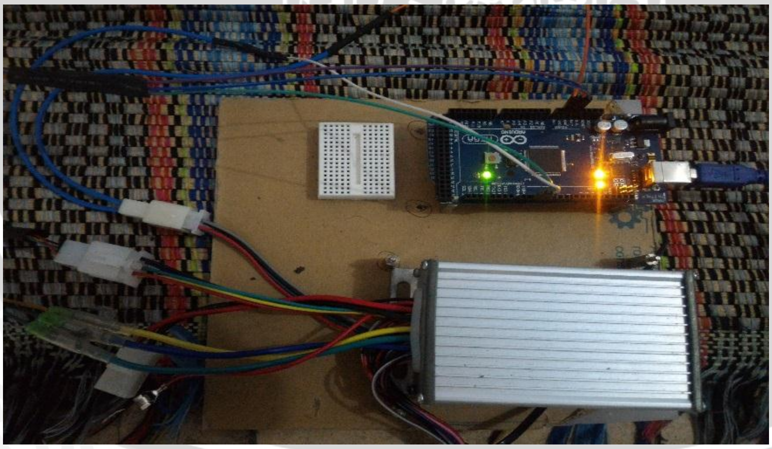
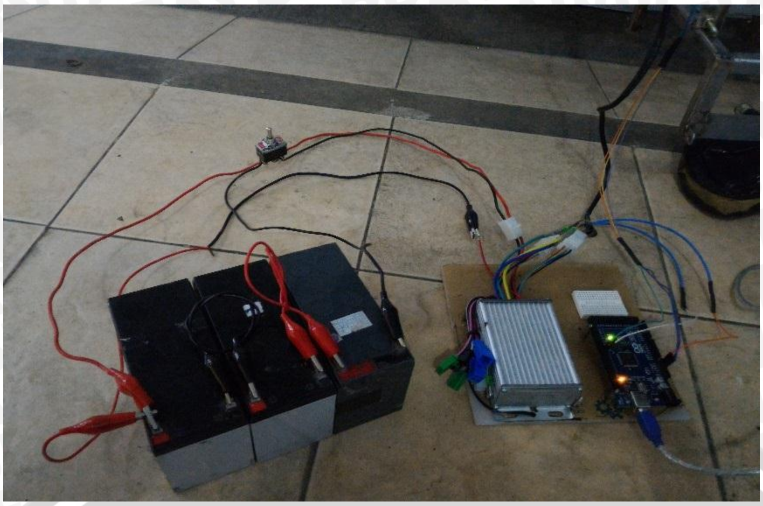
LAMPIRAN I

FOTO ALAT

FOTO ALAT







LAMPIRAN II

LISTING PROGRAM

Listing program MATLAB *Symmetrical Optimum*

```
Gcl=0.596;
Tcl=0.054;
Tmn=1;
D=0.707;
a=2*D+1;
Tcw=a^2*Tcl;
Gcw=(1/(a*Gcl)*(Tmn/Tcl));

s=tf('s');
so=(Gcw*Gcl*Tcw*s+Gcl*Gcw)/(Tcw*Tcl*Tmn*s^3+Tcw*Tmn*s^2);
margin(so);
```

Listing program Arduino sistem kontrol kecepatan sepeda listrik menggunakan metode self-tuning parameter PI dengan metode logika *fuzzy*

```
#define NB 0
#define NK 1
#define ZE 2
#define PK 3
#define PB 4

float value;
float result;
float x0;
float x1;
float x2;
float Ts = 0.01;
float data_Error;
float data_delError;
float Error[5];
float delError[5];
float dke1;
float dke2;
float dkde1;
float dkde2;
float fs1;
float fs2;
float fs3;
float fs4;
float a;
float b;
float tp;
float tn;
float wp;
float wi;
float wap;
float wai;
float y;
float Kp = 3.23465;
float Ki = 10.3245125;
float integrator;
float last1_integrator;
float prop;
float intg;
float rpm;
int hie1;
int hie2;
int hide1;
int hide2;
int hop1;
int hop2;
int hop3;
int hop4;
int hoi1;
int hoi2;
int hoi3;
int hoi4;
int MV;
int pwm;
int setpoint;
volatile int pulsa = 0;


float triangle(float
value, float x0, float
x1, float x2);
float reverse_grade(float
value, float x0, float
x1);
float grade(float value,
float x0, float x1);
void fuzzy();
void
derajat_keanggotaan();
void firing_strength();
void weighted_average();


float triangle(float
value, float x0, float
x1, float x2)
{
  float result = 0;
  float x;
  x = value;

  if ((x <= x0) || (x >=
x2))
  {
    result = 0;
  }

  else if (x == x1)
  {
    result = 1;
  }

  else if ((x >= x0) &&
(x < x1))
  {
    result = ((x - x0) /
(x1 - x0));
  }

  else
  {
    result = (((-x) + x2)
/ (x2 - x1));
  }

  return result;
}
float grade(float value,
float x0, float x1)
{
  float result = 0;
  float x;

  x = value;
  if (x <= x0)
  {
    result = 0;
  }
  else if (x >= x1)
  {
    result = 1;
  }
  else
  {
    result = (x - x0) /
(x1 - x0);
  }

  return result;
}

float reverse_grade(float
value, float x0, float
x1)
{
  float result = 0;
  float x;
  x = value;

  if (x <= x0)
  {
    result = 1;
  }
  else if (x >= x1)
  {
    result = 0;
  }
  else
  {
    result = (-x + x1) /
(x1 - x0);
  }

  return result;
}
```

```
void fuzzy()
{
  Error[NB] =
reverse_grade(data_Error,
-20, -12);
  Error[NK] =
triangle(data_Error, -20,
-12, 0);
  Error[ZE] =
triangle(data_Error, -12,
0, 12);
  Error[PK] =
triangle(data_Error, 0,
12, 20);
  Error[PB] =
grade(data_Error, 12,
20);

  delError[NB] =
reverse_grade(data_delErr
or, -2, -1.2);
  delError[NK] =
triangle(data_delError, -
2, -1.2, 0);
  delError[ZE] =
triangle(data_delError, -
1.2, 0, 1.2);
  delError[PK] =
triangle(data_delError,
0, 1.2, 2);
  delError[PB] =
grade(data_delError, 1.2,
2);
}

void
derajat_keanggotaan()
{
  if (Error[NB] > 0)
  {
    dke1 = Error[NB];
    hie1 = NB;
  }
  if (Error[NK] > 0)
  {
    if (dke1 > 0)
    {
      dke2 = Error[NK];
      hie2 = NK;
    }
    else
    {
      dke1 = Error[NK];
      hie1 = NK;
    }
  }
  if (Error[ZE] > 0)
  {
    if (dke1 > 0)
    {
      dke2 = Error[ZE];
      hie2 = ZE;
    }
    else
    {
      dke1 = Error[ZE];
      hie1 = ZE;
    }
  }
  if (Error[PK] > 0)
```

```
  {
    if (dke1 > 0)
    {
      dke2 = Error[PK];
      hie2 = PK;
    }
    else
    {
      dke1 = Error[PK];
      hie1 = PK;
    }
  }
  if (Error[PB] > 0)
  {
    if (dke1 > 0)
    {
      dke2 = Error[PB];
      hie2 = PB;
    }
    else
    {
      dke1 = Error[PB];
      hie1 = PB;
    }
  }

  if (delError[NB] > 0)
  {
    dkde1 = delError[NB];
    hide1 = NB;
  }
  if (delError[NK] > 0)
  {
    if (dkde1 > 0)
    {
      dkde2 =
delError[NK];
      hide2 = NK;
    }
    else
    {
      dkde1 =
delError[NK];
      hide1 = NK;
    }
  }
  if (delError[ZE] > 0)
  {
    if (dkde1 > 0)
    {
      dkde2 =
delError[ZE];
      hide2 = ZE;
    }
    else
    {
      dkde1 =
delError[ZE];
      hide1 = ZE;
    }
  }
  if (delError[PK] > 0)
  {
    if (dkde1 > 0)
    {
      dkde2 =
delError[PK];
      hide2 = PK;
    }
    else
    {
```

```
      dkde1 =
delError[PK];
      hide1 = PK;
    }
  }
  if (delError[PB] > 0)
  {
    if (dkde1 > 0)
    {
      dkde2 =
delError[PB];
      hide2 = PB;
    }
    else
    {
      dkde1 =
delError[PB];
      hide1 = PB;
    }
  }
}

void firing_strength()
{
  fs1 = min(dke1, dkde1);
  fs2 = min(dke1, dkde2);
  fs3 = min(dke2, dkde1);
  fs4 = min(dke2, dkde2);

  if (fs1 > 0)
  {
    if (hie1 == NB)
    {
      if (hide1 == NB)
      {
        hop1 = NB;
        hoi1 = NB;
      }
      else if (hide1 ==
NK)
      {
        hop1 = NK;
        hoi1 = NB;
      }
      else if (hide1 ==
ZE)
      {
        hop1 = NK;
        hoi1 = NB;
      }
      else if (hide1 ==
PK)
      {
        hop1 = NK;
        hoi1 = NK;
      }
      else if (hide1 ==
PB)
      {
        hop1 = ZE;
        hoi1 = ZE;
      }
    }
    else if (hie1 == NK)
    {
      if (hide1 == NB)
      {
        hop1 = NB;
        hoi1 = NB;
      }
```

```
   else if (hide1 ==
NK)
   {
    hop1 = NK;
    hoi1 = NB;
   }
   else if (hide1 ==
ZE)
   {
    hop1 = NK;
    hoi1 = NK;
   }
   else if (hide1 ==
PK)
   {
    hop1 = ZE;
    hoi1 = ZE;
   }
   else if (hide1 ==
PB)
   {
    hop1 = PK;
    hoi1 = PK;
   }
  }
  else if (hie1 == ZE)
  {
   if (hide1 == NB)
   {
    hop1 = NB;
    hoi1 = NB;
   }
   else if (hide1 ==
NK)
   {
    hop1 = NK;
    hoi1 = NK;
   }
   else if (hide1 ==
ZE)
   {
    hop1 = ZE;
    hoi1 = ZE;
   }
   else if (hide1 ==
PK)
   {
    hop1 = PK;
    hoi1 = PK;
   }
   else if (hide1 ==
PB)
   {
    hop1 = PB;
    hoi1 = PB;
   }
  }
  else if (hie1 == PK)
  {
   if (hide1 == NB)
   {
    hop1 = NK;
    hoi1 = NK;
   }
   else if (hide1 ==
NK)
   {
    hop1 = ZE;
    hoi1 = ZE;
   }
```

```
   else if (hide1 ==
ZE)
   {
    hop1 = PK;
    hoi1 = PK;
   }
   else if (hide1 ==
PK)
   {
    hop1 = PK;
    hoi1 = PB;
   }
   else if (hide1 ==
PB)
   {
    hop1 = PB;
    hoi1 = PB;
   }
  }
  else if (hie1 == PB)
  {
   if (hide1 == NB)
   {
    hop1 = ZE;
    hoi1 = ZE;
   }
   else if (hide1 ==
NK)
   {
    hop1 = PK;
    hoi1 = PK;
   }
   else if (hide1 ==
ZE)
   {
    hop1 = PK;
    hoi1 = PB;
   }
   else if (hide1 ==
PK)
   {
    hop1 = PK;
    hoi1 = PB;
   }
   else if (hide1 ==
PB)
   {
    hop1 = PB;
    hoi1 = PB;
   }
  }
 }

 if (fs2 > 0)
 {
  if (hie1 == NB)
  {
   if (hide2 == NB)
   {
    hop2 = NB;
    hoi2 = NB;
   }
   else if (hide2 ==
NK)
   {
    hop2 = NK;
    hoi2 = NB;
   }
   else if (hide2 ==
ZE)
   {
```

```
    hop2 = NK;
    hoi2 = NB;
   }
   else if (hide2 ==
PK)
   {
    hop2 = NK;
    hoi2 = NK;
   }
   else if (hide2 ==
PB)
   {
    hop2 = ZE;
    hoi2 = ZE;
   }
  }
  else if (hie1 == NK)
  {
   if (hide2 == NB)
   {
    hop2 = NB;
    hoi2 = NB;
   }
   else if (hide2 ==
NK)
   {
    hop2 = NK;
    hoi2 = NB;
   }
   else if (hide2 ==
ZE)
   {
    hop2 = NK;
    hoi2 = NK;
   }
   else if (hide2 ==
PK)
   {
    hop2 = ZE;
    hoi2 = ZE;
   }
   else if (hide2 ==
PB)
   {
    hop2 = PK;
    hoi2 = PK;
   }
  }
  else if (hie1 == ZE)
  {
   if (hide2 == NB)
   {
    hop2 = NB;
    hoi2 = NB;
   }
   else if (hide2 ==
NK)
   {
    hop2 = NK;
    hoi2 = NK;
   }
   else if (hide2 ==
ZE)
   {
    hop2 = ZE;
    hoi2 = ZE;
   }
   else if (hide2 ==
PK)
   {
    hop2 = PK;
    hoi2 = PK;
   }
   else if (hide2 ==
PB)
   {
    hop2 = PB;
    hoi2 = PB;
   }
  }
  else if (hie1 == PK)
  {
   if (hide2 == NB)
   {
    hop2 = NK;
    hoi2 = NK;
   }
   else if (hide2 ==
NK)
   {
    hop2 = ZE;
    hoi2 = ZE;
   }
   else if (hide2 ==
ZE)
   {
    hop2 = PK;
    hoi2 = PK;
   }
   else if (hide2 ==
PK)
   {
    hop2 = PK;
    hoi2 = PB;
   }
   else if (hide2 ==
PB)
   {
    hop2 = PB;
    hoi2 = PB;
   }
  }
  else if (hie1 == PB)
  {
   if (hide2 == NB)
   {
    hop2 = ZE;
    hoi2 = ZE;
   }
   else if (hide2 ==
NK)
   {
    hop2 = PK;
    hoi2 = PK;
   }
   else if (hide2 ==
ZE)
   {
    hop2 = PK;
    hoi2 = PB;
   }
   else if (hide2 ==
PK)
   {
    hop2 = PK;
    hoi2 = PB;
   }
   else if (hide2 ==
PB)
   {
    hop2 = PB;
    hoi2 = PB;
```

```
   }
  }
 }
 if (fs3 > 0)
 {
  if (hie2 == NB)
  {
   if (hide1 == NB)
   {
    hop3 = NB;
    hoi3 = NB;
   }
   else if (hide1 ==
NK)
   {
    hop3 = NK;
    hoi3 = NB;
   }
   else if (hide1 ==
ZE)
   {
    hop3 = NK;
    hoi3 = NB;
   }
   else if (hide1 ==
PK)
   {
    hop3 = NK;
    hoi3 = NK;
   }
   else if (hide1 ==
PB)
   {
    hop3 = ZE;
    hoi3 = ZE;
   }
  }
  else if (hie2 == NK)
  {
   if (hide1 == NB)
   {
    hop3 = NB;
    hoi3 = NB;
   }
   else if (hide1 ==
NK)
   {
    hop3 = NK;
    hoi3 = NB;
   }
   else if (hide1 ==
ZE)
   {
    hop3 = NK;
    hoi3 = NK;
   }
   else if (hide1 ==
PK)
   {
    hop3 = ZE;
    hoi3 = ZE;
   }
   else if (hide1 ==
PB)
   {
    hop3 = PK;
    hoi3 = PK;
   }
  }
  else if (hie2 == ZE)
```

```
  {
   if (hide1 == NB)
   {
    hop3 = NB;
    hoi3 = NB;
   }
   else if (hide1 ==
NK)
   {
    hop3 = NK;
    hoi3 = NK;
   }
   else if (hide1 ==
ZE)
   {
    hop3 = ZE;
    hoi3 = ZE;
   }
   else if (hide1 ==
PK)
   {
    hop3 = PK;
    hoi3 = PK;
   }
   else if (hide1 ==
PB)
   {
    hop3 = PB;
    hoi3 = PB;
   }
  }
  else if (hie2 == PK)
  {
   if (hide1 == NB)
   {
    hop3 = NK;
    hoi3 = NK;
   }
   else if (hide1 ==
NK)
   {
    hop3 = ZE;
    hoi3 = ZE;
   }
   else if (hide1 ==
ZE)
   {
    hop3 = PK;
    hoi3 = PK;
   }
   else if (hide1 ==
PK)
   {
    hop3 = PK;
    hoi3 = PB;
   }
   else if (hide1 ==
PB)
   {
    hop3 = PB;
    hoi3 = PB;
   }
  }
  else if (hie2 == PB)
  {
   if (hide1 == NB)
   {
    hop3 = ZE;
    hoi3 = ZE;
   }
```

```
      else if (hide1 ==
NK)
      {
        hop3 = PK;
        hoi3 = PK;
      }
      else if (hide1 ==
ZE)
      {
        hop3 = PK;
        hoi3 = PB;
      }
      else if (hide1 ==
PK)
      {
        hop3 = PK;
        hoi3 = PB;
      }
      else if (hide1 ==
PB)
      {
        hop3 = PB;
        hoi3 = PB;
      }
    }
  }
  if (fs4 > 0)
  {
    if (hie2 == NB)
    {
      if (hide2 == NB)
      {
        hop4 = NB;
        hoi4 = NB;
      }
      else if (hide2 ==
NK)
      {
        hop4 = NK;
        hoi4 = NB;
      }
      else if (hide2 ==
ZE)
      {
        hop4 = NK;
        hoi4 = NB;
      }
      else if (hide2 ==
PK)
      {
        hop4 = NK;
        hoi4 = NK;
      }
      else if (hide2 ==
PB)
      {
        hop4 = ZE;
        hoi4 = ZE;
      }
    }
    else if (hie2 == NK)
    {
      if (hide2 == NB)
      {
        hop4 = NB;
        hoi4 = NB;
      }
      else if (hide2 ==
NK)
      {
        hop4 = NK;
```

```
        hoi4 = NB;
      }
      else if (hide2 ==
ZE)
      {
        hop4 = NK;
        hoi4 = NK;
      }
      else if (hide2 ==
PK)
      {
        hop4 = ZE;
        hoi4 = ZE;
      }
      else if (hide2 ==
PB)
      {
        hop4 = PK;
        hoi4 = PK;
      }
    }
    else if (hie2 == ZE)
    {
      if (hide2 == NB)
      {
        hop4 = NB;
        hoi4 = NB;
      }
      else if (hide2 ==
NK)
      {
        hop4 = NK;
        hoi4 = NK;
      }
      else if (hide2 ==
ZE)
      {
        hop4 = ZE;
        hoi4 = ZE;
      }
      else if (hide2 ==
PK)
      {
        hop4 = PK;
        hoi4 = PK;
      }
      else if (hide2 ==
PB)
      {
        hop4 = PB;
        hoi4 = PB;
      }
    }
    else if (hie2 == PK)
    {
      if (hide2 == NB)
      {
        hop4 = NK;
        hoi4 = NK;
      }
      else if (hide2 ==
NK)
      {
        hop4 = ZE;
        hoi4 = ZE;
      }
      else if (hide2 ==
ZE)
      {
        hop4 = PK;
        hoi4 = PK;
```

```
      }
      else if (hide2 ==
PK)
      {
        hop4 = PK;
        hoi4 = PB;
      }
      else if (hide2 ==
PB)
      {
        hop4 = PB;
        hoi4 = PB;
      }
    }
    else if (hie2 == PB)
    {
      if (hide2 == NB)
      {
        hop4 = ZE;
        hoi4 = ZE;
      }
      else if (hide2 ==
NK)
      {
        hop4 = PK;
        hoi4 = PK;
      }
      else if (hide2 ==
ZE)
      {
        hop4 = PK;
        hoi4 = PB;
      }
      else if (hide2 ==
PK)
      {
        hop4 = PK;
        hoi4 = PB;
      }
      else if (hide2 ==
PB)
      {
        hop4 = PB;
        hoi4 = PB;
      }
    }
  }
}

void weighted_average()
{
  if (fs1 > 0)
  {
    if (hop1 == 0)
    {
      a = 6.47 - fs1 *
3.237;
      tn = ((a * a + a +
1) / (3 * (a + 1))) -
9.704;
      wp = wp + tn * fs1;
    }
    else if (hop1 == 1)
    {
      wp = wp + (-3.234)
* fs1;
    }
    else if (hop1 == 2)
    {
      wp = wp + 0 * fs1;
    }
    else if (hop1 == 3)
    {
      wp = wp + 3.234 *
fs1;
    }
    else if (hop1 == 4)
    {
      b = 6.47 - fs1 *
3.237;
      tp = 9.704 - ((b *
b + b + 1) / (3 * (b +
1)));
      wp = wp + tp * fs1;
    }

    if (hoi1 == 0)
    {
      a = 18.58 - fs1 *
12.37;
      tn = ((a * a + a +
1) / (3 * (a + 1))) -
30.98;
      wi = wi + tn * fs1;
    }
    else if (hoi1 == 1)
    {
      wi = wi + (-12.37)
* fs1;
    }
    else if (hoi1 == 2)
    {
      wi = wi + 0 * fs1;
    }
    else if (hoi1 == 3)
    {
      wi = wi + 12.37 *
fs1;
    }
    else if (hoi1 == 4)
    {
      b = 18.58 - fs1 *
12.37;
      tp = 30.98 - ((b *
b + b + 1) / (3 * (b +
1)));
      wi = wi + tp * fs1;
    }
  }

  if (fs2 > 0)
  {
    if (hop2 == 0)
    {
      a = 6.47 - fs2 *
3.237;
      tn = ((a * a + a +
1) / (3 * (a + 1))) -
9.704;
      wp = wp + tn * fs2;
    }
    else if (hop2 == 1)
    {
      wp = wp + (-3.234)
* fs2;
    }
    else if (hop2 == 2)
    {
      wp = wp + 0 * fs2;
    }
    else if (hop2 == 3)
    {
      wp = wp + 3.234 *
fs2;
    }
    else if (hop2 == 4)
    {
      b = 6.47 - fs2 *
3.237;
      tp = 9.704 - ((b *
b + b + 1) / (3 * (b +
1)));
      wp = wp + tp * fs2;
    }

    if (hoi2 == 0)
    {
      a = 18.58 - fs2 *
12.37;
      tn = ((a * a + a +
1) / (3 * (a + 1))) -
30.98;
      wi = wi + tn * fs2;
    }
    else if (hoi2 == 1)
    {
      wi = wi + (-12.37)
* fs2;
    }
    else if (hoi2 == 2)
    {
      wi = wi + 0 * fs2;
    }
    else if (hoi2 == 3)
    {
      wi = wi + 12.37 *
fs2;
    }
    else if (hoi2 == 4)
    {
      b = 18.58 - fs2 *
12.37;
      tp = 30.98 - ((b *
b + b + 1) / (3 * (b +
1)));
      wi = wi + tp * fs2;
    }
  }

  if (fs3 > 0)
  {
    if (hop3 == 0)
    {
      a = 6.47 - fs3 *
3.237;
      tn = ((a * a + a +
1) / (3 * (a + 1))) -
9.704;
      wp = wp + tn * fs3;
    }
    else if (hop3 == 1)
    {
      wp = wp + (-3.234)
* fs3;
    }
    else if (hop3 == 2)
    {
      wp = wp + 0 * fs3;
    }
    else if (hop3 == 3)
    {
      wp = wp + 3.234 *
fs3;
    }
    else if (hop3 == 4)
    {
      b = 6.47 - fs3 *
3.237;
      tp = 9.704 - ((b *
b + b + 1) / (3 * (b +
1)));
      wp = wp + tp * fs3;
    }

    if (hoi3 == 0)
    {
      a = 18.58 - fs3 *
12.37;
      tn = ((a * a + a +
1) / (3 * (a + 1))) -
30.98;
      wi = wi + tn * fs3;
    }
    else if (hoi3 == 1)
    {
      wi = wi + (-12.37)
* fs3;
    }
    else if (hoi3 == 2)
    {
      wi = wi + 0 * fs3;
    }
    else if (hoi3 == 3)
    {
      wi = wi + 12.37 *
fs3;
    }
    else if (hoi3 == 4)
    {
      b = 18.58 - fs3 *
12.37;
      tp = 30.98 - ((b *
b + b + 1) / (3 * (b +
1)));
      wi = wi + tp * fs3;
    }
  }

  if (fs4 > 0)
  {
    if (hop4 == 0)
    {
      a = 6.47 - fs4 *
3.237;
      tn = ((a * a + a +
1) / (3 * (a + 1))) -
9.704;
      wp = wp + tn * fs4;
    }
    else if (hop4 == 1)
    {
      wp = wp + (-3.234)
* fs4;
    }
    else if (hop4 == 2)
    {
      wp = wp + 0 * fs4;
    }
    else if (hop4 == 3)
    {
      wp = wp + 3.234 *
fs4;
    }
    else if (hop4 == 4)
    {
      b = 6.47 - fs4 *
3.237;
      tp = 9.704 - ((b *
b + b + 1) / (3 * (b +
1)));
      wp = wp + tp * fs4;
    }

    if (hoi4 == 0)
    {
      a = 18.58 - fs4 *
12.37;
      tn = ((a * a + a +
1) / (3 * (a + 1))) -
30.98;
      wi = wi + tn * fs4;
    }
    else if (hoi4 == 1)
    {
      wi = wi + (-12.37)
* fs4;
    }
    else if (hoi4 == 2)
    {
      wi = wi + 0 * fs4;
    }
    else if (hoi4 == 3)
    {
      wi = wi + 12.37 *
fs4;
    }
    else if (hoi4 == 4)
    {
      b = 18.58 - fs4 *
12.37;
      tp = 30.98 - ((b *
b + b + 1) / (3 * (b +
1)));
      wi = wi + tp * fs4;
    }
  }

  wap = wp / (fs1 + fs2 +
fs3 + fs4);
  wai = wi / (fs1 + fs2 +
fs3 + fs4);
}

void PI_controller()
{
  prop = Kp * data_Error;
```

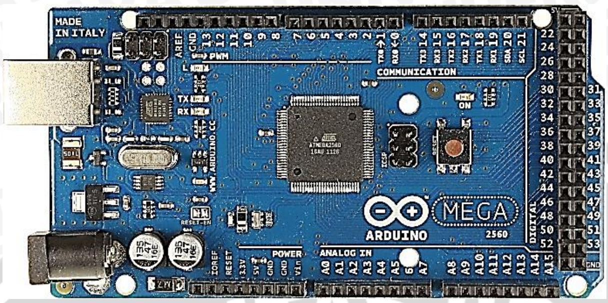
```
  intg = last1_integrator
+ Ki * Ts * data_Error;
  MV = prop + intg;

  last1_integrator =
intg;

  if (MV > 160) pwm =
160;
  else if (MV < 80) pwm =
80;
  else pwm = MV;
  analogWrite(3, pwm);
}

void reset_value()
{
  dke1 = 0;
  dke2 = 0;
  dkde1 = 0;
  dkde2 = 0;
  a = 0;
  b = 0;
  wp = 0;
  wi = 0;
}

void timer()
{
  TCCR1A = 0;
  TCCR1B = 0;
  TCNT1 = 0;

  OCR1A = 6250;
  TCCR1B |= (1 << WGM12);
  TCCR1B |= (1 << CS12);
  TIMSK1 |= (1 <<
OCIE1A);
  TCCR3B = TCCR3B &
B11111000 | B00000001;
}

ISR(TIMER1_COMPA_vect)
{
  detachInterrupt(0);
  rpm = (pulsa * 600) /
36;
  y = rpm / 2.807 + 65;
  pulsa = 0;
  attachInterrupt(0,
rpm_fan, FALLING);
```

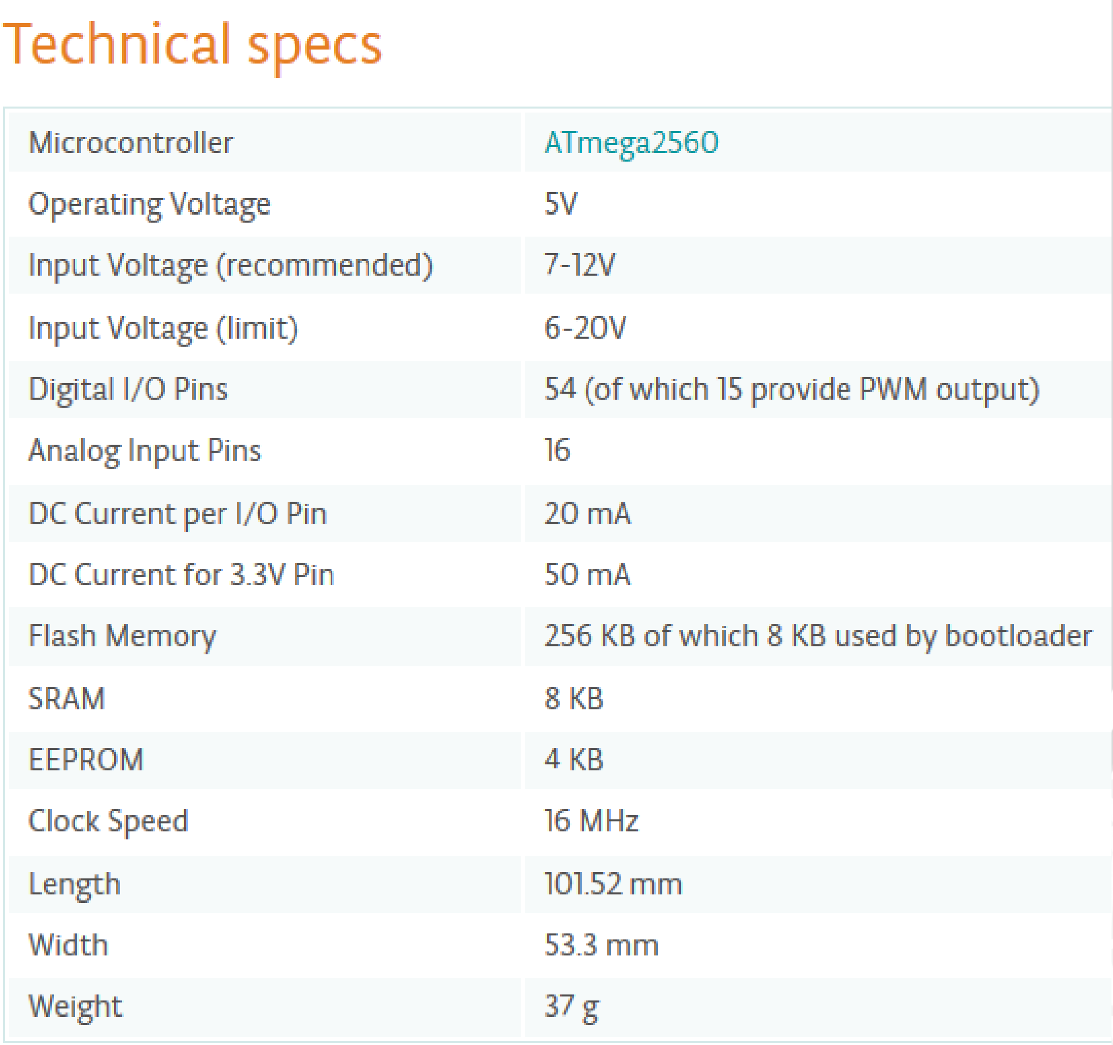
```
  data_Error = setpoint -
y;

  reset_value();
  fuzzy();
  derajat_keanggotaan();
  firing_strength();
  weighted_average();
  PI_controller();

  data_delError =
data_Error;

  Serial.print(setpoint);
  Serial.print("\t");
  Serial.print(MV);
  Serial.print("\t");
  Serial.print(wap);
  Serial.print("\t");
  Serial.print(wai);
  Serial.print("\t");
  Serial.print(pwm);
  Serial.print("\t");
  Serial.println(rpm);
}

void setup()
{
  setpoint = 105;
  timer();
  Serial.begin(9600);
  pinMode (3, OUTPUT);
  data_Error = 0;
  float integrator = 0;
  float last1_integrator
= 0;
  attachInterrupt(0,
rpm_fan, FALLING);
}

void loop()
{
}

void rpm_fan()
{
  pulsa++;
}
```

LAMPIRAN III

DATASHEET

Mikrokontroler Arduino Mega 2560





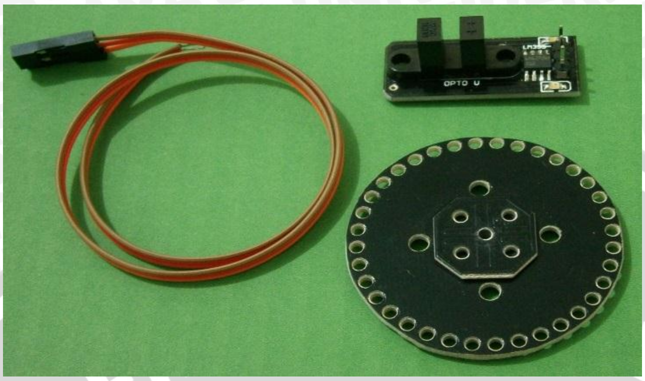
Technical specs

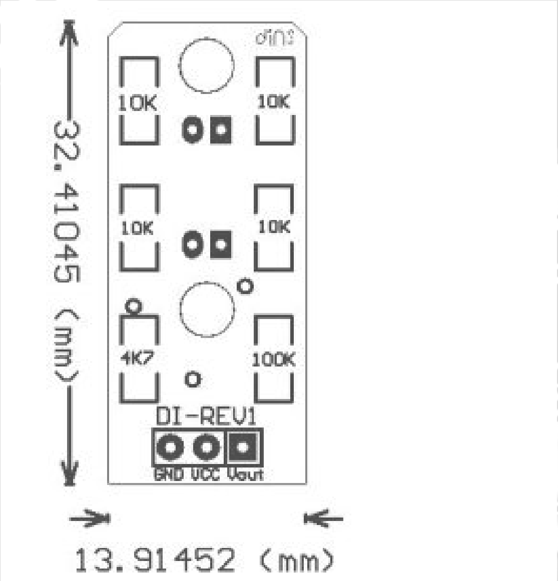
Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz
Length	101.52 mm
Width	53.3 mm
Weight	37 g

Beberapa pin memiliki fungsi khusus:

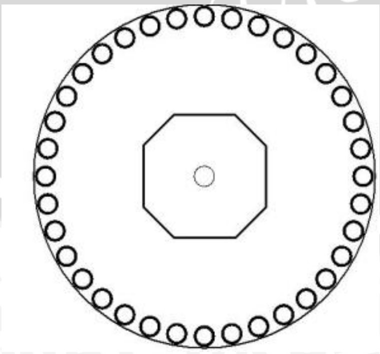
- *Serial*: 0 (RX) dan 1 (TX); *Serial* 1: 19 (RX) dan 18 (TX); *Serial* 2: 17 (RX) dan 16 (TX); *Serial* 3: 15 (RX) dan 14 (TX). Digunakan untuk menerima (RX) dan mengirimkan (TX) TTL data serial. Pin ini dihubungkan ke pin yang berkaitan dengan chip Serial ATmega8U2 USB-to-TTL.
- *Eksternal interrupts*: 2 (*interrupt* 0), 3 (*interrupt* 1*)*, 18 (*interrupt* 5), 19 (*interrupt* 4), 20 (*interrupt* 3), dan 2 (*interrupt* 2). Pin ini dapat dikonfigurasi untuk memicu *interrupt* pada nilai yang rendah, dengan batasan tepi naik atau turun, atau perubahan nilai.
- PWM: 0 - 13. Menyediakan output PWM 8-bit dengan fungsi *analogWrite ()*.
- *SPI*: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS). Pin ini mendukung komunikasi SPI menggunakan *SPI library*.
- *LED*: 13. Ada *built-in* LED terhubung ke pin digital 13. Ketika pin bernilai nilai *high* LED menyala dan ketika pin bernilai *low* LED mati.
- I^{2}C: 20 (SDA) dan 21 (SCL). Dukungan I^{2}C (TWI) komunikasi menggunakan *wire*.

Sensor *Rotary Encoder* DRN136

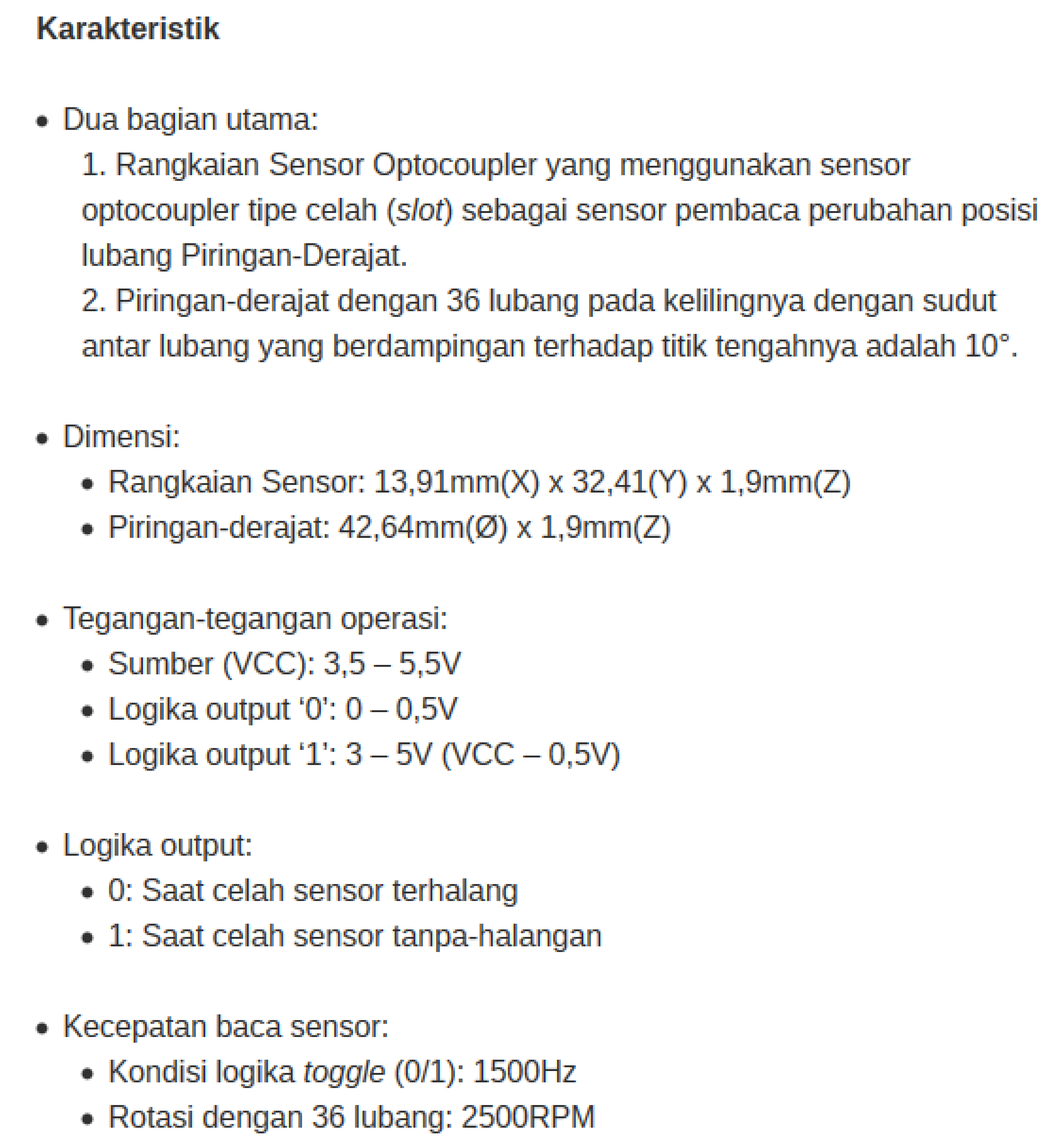




Rangkaian Sensor Optocoupler DRN136



Piringan Derajat DRN136



Karakteristik

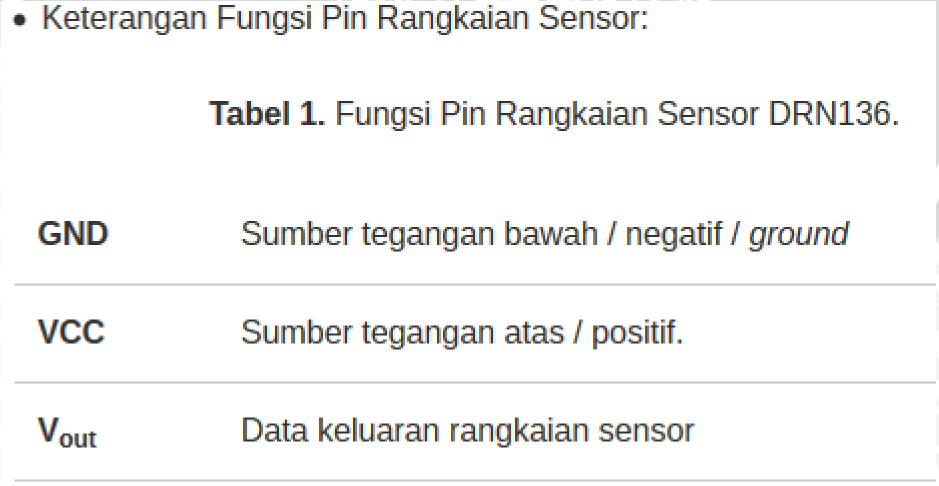
- Dua bagian utama:
  1. Rangkaian Sensor Optocoupler yang menggunakan sensor optocoupler tipe celah (*slot*) sebagai sensor pembaca perubahan posisi lubang Piringan-Derajat.
  2. Piringan-derajat dengan 36 lubang pada kelilingnya dengan sudut antar lubang yang berdampingan terhadap titik tengahnya adalah 10°.

- Dimensi:
  - Rangkaian Sensor: 13,91mm(X) x 32,41(Y) x 1,9mm(Z)
  - Piringan-derajat: 42,64mm(Ø) x 1,9mm(Z)

- Tegangan-tegangan operasi:
  - Sumber (VCC): 3,5 – 5,5V
  - Logika output ‘0’: 0 – 0,5V
  - Logika output ‘1’: 3 – 5V (VCC – 0,5V)

- Logika output:
  - 0: Saat celah sensor terhalang
  - 1: Saat celah sensor tanpa-halangan

- Kecepatan baca sensor:
  - Kondisi logika *toggle* (0/1): 1500Hz
  - Rotasi dengan 36 lubang: 2500RPM



- Keterangan Fungsi Pin Rangkaian Sensor:

Tabel 1. Fungsi Pin Rangkaian Sensor DRN136.

GND	Sumber tegangan bawah / negatif / *ground*
VCC	Sumber tegangan atas / positif.
V_{out}	Data keluaran rangkaian sensor