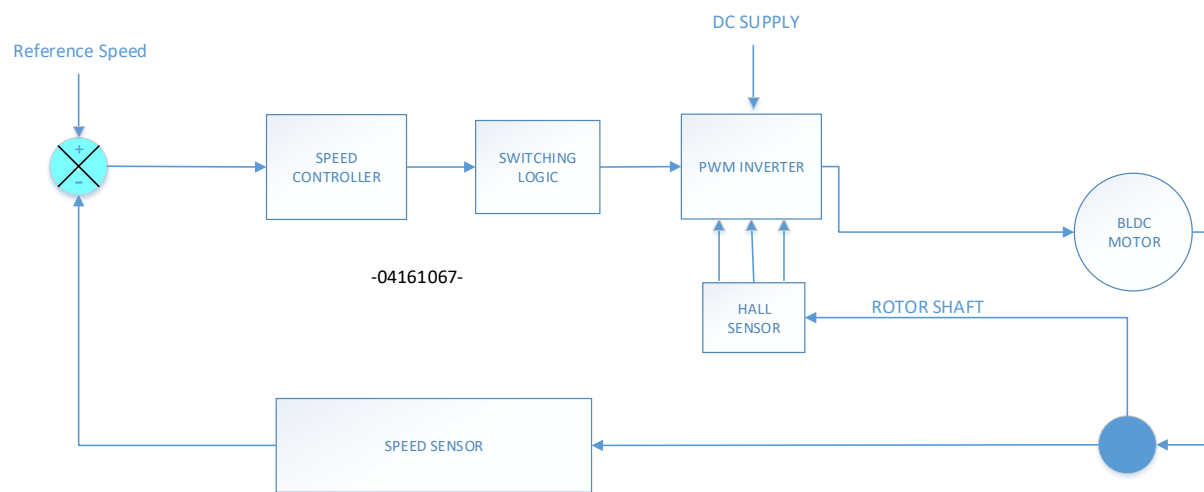


NAMA : RIZKY WULANDARI
NIM : 04161067
TUGAS : SISTEM MIKROPROSESOR

Sistem Pengendalian Kecepatan BLDC Motor



Gambar 1. Sistem Closed Loop Pengendalian Kecepatan BLDC Motor

A. SPEED CONTROLLER

Speed control merupakan salah satu unit dari controller Motor BLDC. Pada system ini digunakan Arduino Uno sebagai Speed Controller. Speed controller ini berguna agar system terus melakukan *loop* hingga mencapai *Reference Speed* / kecepatan yang diinginkan. Berikut source code untuk pengaturan kecepatan motor BLDC dengan PWM :

```
Kode motor BLDC with PWM
const int motordc = 11;
void setup() {
  // mengatur pin 11 sebagai output.
  pinMode(motordc,OUTPUT);
  // mengatur agar motordc berhenti terlebih dahulu
  analogWrite(motord,0);
```

```

}

void loop() {
  // membuat variabel i sebagai nilai pengatur kecepatan.
  int i = 0
  // melakukan looping untuk meningkatkan kecepatan.
  for (i=0;i<240;i=i+40){
    analogWrite(motordc,i);
    delay(300);
  }
  // melakukan looping untuk menurunkan kecepatan
  for(i=250;i>0;i=i-40){
    analogWrite(motordc,i);
    delay(300);
  }
}

```

B. PWM INVERTER

Inverter merupakan salah satu modul yang terdapat didalam controller motor BLDC yang mana berfungsi untuk mengubah arus listrik dari baterai yang berupa arus DC menjadi arus AC, dikarenakan motor BLDC memiliki karakteristik multipole. Didalam inverter terdapat komponen kelistrikan utama yaitu berupa diode. Terdapat beberapa jenis diode yang sering digunakan yaitu IGBT Diode, GTO Diode dan MOSFET Diode. Distribusi voltage diatur oleh diode yang terhubung dengan PWM, dimana signal yang telah di distribusi di dalam modul PWM dialirkan kedalam diode melalui gate yang dimiliki oleh komponen ini, yang kemudian dialirkan diteruskan ke masing-masing fasa motor baik v_a , v_b , maupun v_c sesuai dengan kebutuhan.

Pulse Width Modulation (PWM) secara umum adalah sebuah cara memanipulasi lebar sinyal yang dinyatakan dengan pulsa dalam suatu perioda, untuk mendapatkan tegangan rata-rata yang berbeda. Beberapa contoh aplikasi PWM adalah pemodulasian data untuk telekomunikasi, pengontrolan daya atau tegangan yang masuk ke beban, regulator tegangan, audio effect dan penguatan, serta aplikasi-aplikasi lainnya. Aplikasi PWM berbasis mikrokontroler biasanya berupa pengendalian kecepatan motor DC, pengendalian motor servo, pengaturan nyala terang LED dan lain sebagainya.

Sinyal PWM pada umumnya memiliki amplitudo dan frekuensi dasar yang tetap, namun memiliki lebar pulsa yang bervariasi. Lebar Pulsa PWM berbanding lurus dengan amplitudo sinyal asli yang belum termodulasi. Artinya, Sinyal PWM memiliki frekuensi gelombang yang tetap namun duty cycle bervariasi (antara 0% hingga 100%). Pulse Width Modulation (PWM) merupakan salah satu teknik untuk mendapatkan signal analog dari sebuah piranti digital.

Sebenarnya Sinyal PWM dapat dibangkitkan dengan banyak cara, dapat menggunakan metode analog dengan menggunakan rangkaian op-amp atau dengan menggunakan metode digital. Dengan metode analog setiap perubahan PWMnya sangat halus, sedangkan menggunakan metode digital setiap perubahan PWM dipengaruhi oleh resolusi dari PWM itu sendiri. Resolusi adalah jumlah variasi perubahan nilai dalam PWM tersebut. Misalkan suatu PWM memiliki resolusi 8 bit berarti PWM ini memiliki variasi perubahan nilai sebanyak $2^8 = 256$ variasi mulai dari 0 – 255 perubahan nilai yang mewakili duty cycle 0 – 100% dari keluaran PWM tersebut. Pwm dengan duty cycle 25% dengan periode katakanlah 100ms maka berarti sinyal logika 1 pada pwm tersebut adalah 25% dari keseluruhan periode sinyal atau 25% dari 100ms = 25ms, sedangkan untuk sinyal logika 0 pada pwm tersebut adalah sisanya atau 75 ms.

C. BLDC MOTOR

Brushless DC Motor termasuk kedalam jenis motor sinkron. Artinya medan magnet yang dihasilkan oleh stator dan medan magnet yang dihasilkan oleh rotor berputar pada frekuensi yang sama. Motor BLDC tidak mengalami *slip* seperti yang terjadi pada motor induksi biasa. Motor jenis ini mempunyai magnet permanen pada bagian rotor dan elektromagnet pada bagian stator. Setelah itu, dengan menggunakan sebuah rangkaian sederhana (*simple computer system*), maka kita dapat merubah arus di elektromagnet ketika bagian rotornya berputar.

D. ROTOR

Rotor adalah bagian pada motor yang berputar karena adanya gaya elektromagnetik dari stator, dimana pada motor DC brushless bagian rotornya berbeda dengan rotor pada motor DC konvensional yang hanya tersusun dari satu buah elektromagnet yang berada diantara brushes (sikat) yang terhubung pada dua buah motor hingga delapan pasang kutub magnet permanen berbentuk persegi panjang yang saling direkatkan menggunakan semacam “epoxy” dan tidak ada brushes-nya. Rotor dibuat dari magnet tetap dan dapat desain dari dua sampai delapan kutub Magnet Utara(N) atau Selatan(S). Material magnetis yang bagus sangat diperlukan untuk mendapatkan kerapatan medan magnet yang bagus pula. Biasanya magnet ferrit yang dipakai untuk membuat magnet tetap. Tetapi dewasa ini dengan kemajuan teknologi, campuran logam sudah kurang populer untuk digunakan. Benar sekali magnet Ferrit lebih murah, tetapi material ini mempunyai kekurangan yaitu flux density yang rendah untuk ukuran volume material yang diperlukan untuk membentuk rotor.

E. HALL SENSOR

Tidak seperti motor DC brushed komutasi dari motor DC brushless diatur secara elektronik agar motor dapat berputar, stator harus di-energize secara berurutan dan teratur. Sensor hall inilah yang berperan dalam mendeteksi pada bagian rotor mana yang ter-energize oleh fluks magnet sehingga proses komutasi yang berbeda (enam step komutasi) dapat dilakukan oleh

stator dengan tepat karena sensor hall ini dipasang menempel pada stator. Berikut source code untuk mengatur Hall Sensor dengan Arduino Uno :

```
/Analog Hall Sensor
/*****
* Analog Hall Sensor      Uno R3
* A0                      A0
* D0                      7
* VCC                     5V
* GND                     GND
*****/

const int ledPin = 13; //led dihubungkan ke pin13
int sensorPin = A0; // input dari potensiometer
int digitalPin=7; //D0  dihubungkan ke pin7

int sensorValue = 0; // nilai variable sensor dari pin A0
boolean digitalValue=0; // nilai variable sensor dari pin pin7

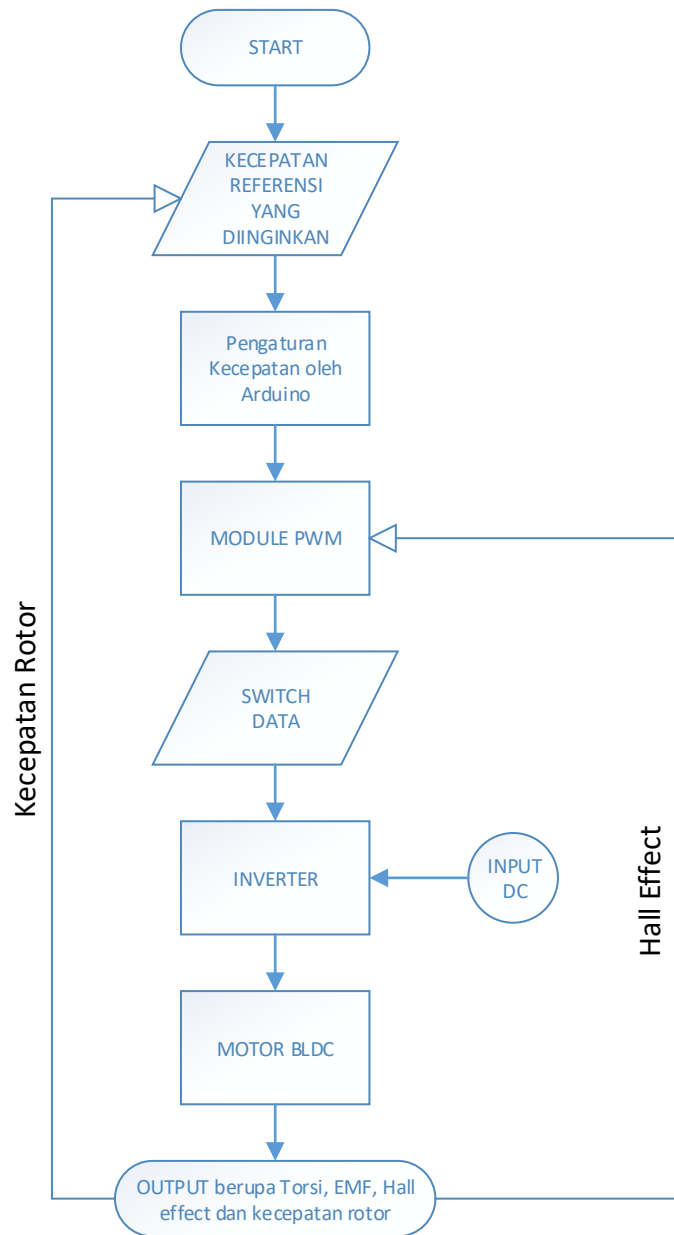
void setup()
{
  pinMode(digitalPin,INPUT); // mengatur D0 sebagai INPUT
  pinMode(ledPin,OUTPUT); //mengatur pin13 sebagai OUTPUT
  Serial.begin(9600); // inisialisasi komunikasi pada kecepatan 9600
  bps
```

F. SPEED SENSOR

Sensor kecepatan atau velocity sensor merupakan suatu sensor yang digunakan untuk mendeteksi kecepatan gerak benda untuk selanjutnya diubah kedalam bentuk sinyal elektrik. Proses penginderaan sensor kecepatan merupakan proses kebalikan dari suatu motor, dimana suatu poros/object yang berputar pada suatu generator akan menghasilkan suatu tegangan yang sebanding dengan kecepatan putaran object. Kecepatan putar sering pula diukur dengan menggunakan sensor yang mengindera pulsa magnetis (induksi) yang timbul saat medan magnetis terjadi. Lalu tegangan ini di kirim ke speed controller (Arduino).

G. FLOWCHART SISTEM

Flowchart dari sistem pengendalian kecepatan dengan motor BLDC dilampirkan sebagai berikut :



Gambar 2. Flowchart Sistem Closed Loop Pengaturan Kecepatan Motor BLDC

H. CONTOH PERHITUNGAN

Misal motor BLDC yang digunakan memiliki rating seperti tabel berikut :

No	input	nilai
1	Voltage	48 Volt
2	Frekuensi	60 Hz
3	Rotor speed	500
4	Daya	800 Watt
5	inductance	0.51×10^{-3} H
6	Arus motor	15 Ampere

Karena nilai input yang diambil dari data spesifikasi belum cukup untuk digunakan untuk simulasi, maka selanjutnya dilakukan perhitungan dengan menggunakan rumus empiris untuk mencari nilai input yang diperlukan dan disesuaikan dengan data berdasarkan spesifikasi dari motor.

- Penentuan nilai torsi Nilai torsi pada motor DC yang kita miliki ditentukan melalui perhitungan secara numeric. Seperti kita ketahui motor yang kita miliki memiliki daya sebesar 800 watt dan putaran sebesar 500 RPM, sehingga kita harus mengubah putaran menjadi kecepatan sudut. Adapun penghitungannya adalah sebagai berikut:

$$\omega = \frac{2\pi \text{ rad}}{1 \text{ revolution}} \times \frac{1 \text{ min}}{60 \text{ sec}} \times 500 \frac{\text{revolution}}{\text{min}} = 52,36 \text{ rad/s}$$

Maka penghitungan Torsi yang dilakukan adalah :

$$P = T \times \omega \quad 800 = T \times 52,36 \text{ rad/s} \quad T = 15,28 \text{ Nm}$$

Berdasarkan perhitungan diatas diperoleh nilai output dari torsi motor sebesar 15.25 Nm

- Penentuan nilai pole pairs Setelah memperoleh nilai dari torsi, kemudian mencari nilai dari pole pairs. Pole pairs merupakan pasangan pole (kutub) yang terdapat pada motor DC. Sebenarnya parameter pole pairs terdapat pada bagian awal namun penentuannya harus berdasarkan nilai RPM sehingga penentuannya dilakukan pada bagian akhir. Adapun penghitungannya adalah sebagai berikut :

$$\omega = \frac{120 \text{ Fz}}{p}$$

$$500 = \frac{120 \cdot 60}{p}$$

$$P = 14,7$$

- Mencari nilai motor voltage Untuk mencari nilai Motor voltage dapat diperoleh dari hubungan daya dan arus pada motor, maka nilai dari motor voltage akan diperoleh sebagai berikut;

$$P = V \times i$$

$$800 = V \times 15 \text{ ampere}$$

$$V = 53.3 \text{ volt}$$