

BAB IV

PERANCANGAN DAN PEMBUATAN ALAT

Perancangan dan pembuatan alat dalam skripsi ini bertujuan untuk merancang beberapa perangkat maupun alat secara keseluruhan. Dalam bab ini membahas mengenai perancangan dan pembuatan sistem pengontrolan kecepatan putaran *propeller* pada *wind tunnel* menggunakan Kontrol Logika *Fuzzy* (KLF). Perancangan perangkat tersebut meliputi perancangan perangkat keras (*Hardware*) maupun perancangan perangkat lunak (*software*). Sedangkan pembuatan alat bertujuan untuk menghasilkan semua perangkat pendukung maupun alat secara keseluruhan.

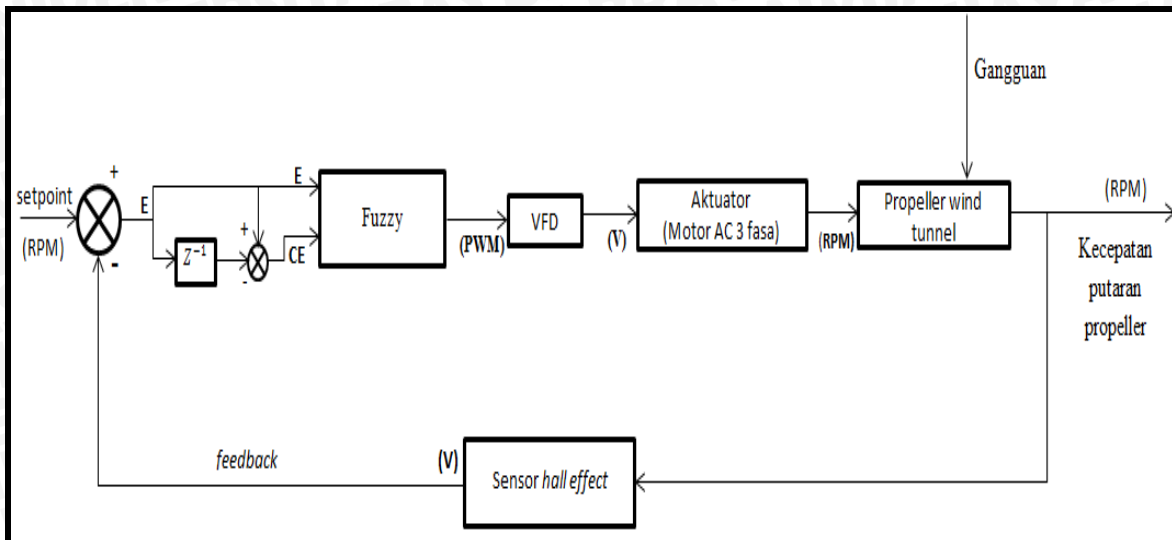
4.1 Perancangan Sistem

Pada perancangan alat ini untuk memudahkan penganalisan yang dilakukan secara bertahap dalam bentuk diagram blok. Penganalisan dilakukan pada setiap blok dan pada keseluruhan sistem. Perancangan ini terdiri atas:

1. Perancangan perangkat keras baling-baling (*propeller*), sensor *Hall effect*, variabel frekuensi drive sebagai inverter, dan motor induksi 3 fasa sebagai penggerak aktuator.
2. Perancangan perangkat lunak (perancangan algoritma Kontrol Logika *Fuzzy* pada *software* Arduino Mega 2560).

4.2 Diagram Blok Sistem

Perancangan sistem keseluruhan disajikan secara diagram blok, diperlihatkan dalam gambar 4.1



Gambar 4. 1 Diagram Blok Sistem

Diagram blok dalam Gambar 4.1 dijelaskan sebagai berikut:

- Masukan / *setpoint* diberikan melalui program pada Arduino Mega berupa kecepatan putaran (RPM).
- Kontroler mengolah data dari *setpoint* dan *feedback* untuk memperoleh besaran error dan change error. KLF mengolah nilai *crisp Error* dan *Change Error* mulai dari proses *fuzzifikasi*, *inferensi* dan *defuzzifikasi*. Hasil *defuzzifikasi* berupa nilai 0-255 dikeluarkan berupa PWM.
- VFD menerima nilai PWM selanjutnya digunakan sebagai acuan untuk menghasilkan catu daya motor induksi 3 fasa.
- Tegangan keluaran VFD selanjutnya akan menjalankan motor 3 fasa.
- Kecepatan putaran dari propeller selanjutnya dibaca oleh sensor *hall effect* dengan keluaran tegangan analog, selanjutnya diubah menjadi digital oleh *converter ADC*.
- Dari pembacaan sensor (*output*) kemudian nilai *setpoint* dikurangkan dengan nilai pembacaan sensor sehingga mikrokontroler mampu mengkompensasi *Error* yang terjadi.

4.3 Perancangan Perangkat Keras

Berdasarkan diagram blok perancangan alat yang telah disusun, perangkat keras meliputi mekanik *propeller wind tunnel*, sensor *Hall Effect*, Motor Induksi 3 Fasa, *Variable Frequency Drive*, Modul Arduino Mega 2560. Di bawah ini adalah penjelasan masing-masing rangkaian penyusun keseluruhan alat.

4.3.1 *Propeller*

Propeller yang digunakan adalah baling-baling berbahan plat besi berdiameter 147 cm. Penentuan ini didasarkan pada ukuran *wind tunnel* yaitu 150x150 cm. *Propeller* berfungsi mengubah energi mekanis putaran poros motor penggerak menjadi energi *fluida* dalam hal ini dapat berupa kecepatan aliran udara. Fungsi dari *Propeller* adalah untuk membangkitkan aliran udara pada *wind tunnel*, Dengan adanya Perubahan aliran udara pada *wind tunnel* maka didapatkan parameter-parameter yang akan diukur dalam pengujian ini. Model *propeller* dapat dilihat dalam Gambar 4.2.



Gambar 4.2 *Propeller*

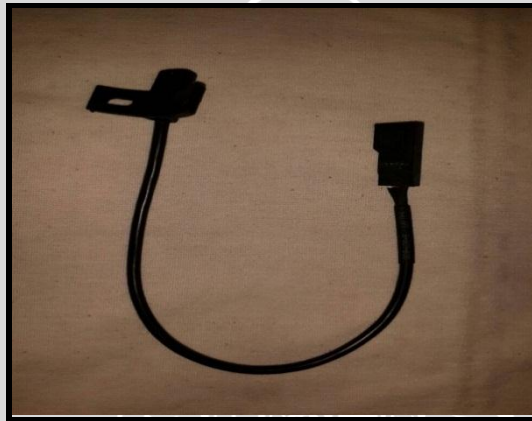
Dan untuk model mekanik alat uji *wind tunnel* dapat dilihat pada Gambar 4.3.



Gambar 4.3 Model Alat *Wind Tunnel*

4.3.2 Sensor *Hall effect*

Sensor *Hall Effect* yang digunakan pada perancangan alat ini terpasang menjadi satu pada *propeller*, bekerja sebagai pengolah sinyal yang dihasilkan dari magnet/ reluktor yang berputar. Sensor *Hall Effect* memberikan output berupa tegangan, pada saat tanpa ada pengaruh dari medan magnet maka beda potensial antar kedua elektroda tersebut 0 Volt karena arus listrik mengalir ditengah kedua elektroda. Ketika terdapat medan magnet mempengaruhi sensor ini maka arus yang mengalir akan berbelok mendekati/menjauhi sisi yang dipengaruhi oleh medan magnet. Hal ini menghasilkan beda potensial diantara kedua elektroda dari sensor *Hall Effect*, dimana beda potensial tersebut sebanding dengan kuat medan magnet yang diterima oleh sensor *Hall Effect* ini. Sinyal dari sensor *Hall Effect* menghasilkan keluaran berupa sinyal analog. Sensor yang digunakan dalam pengujian ini dapat dilihat dalam Gambar 4.4.



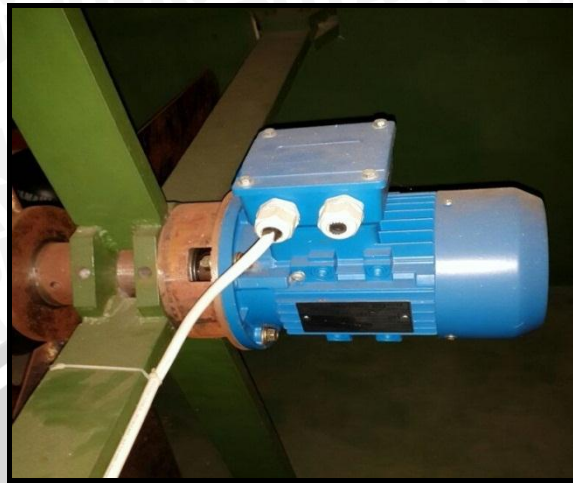
Gambar 4.4 Sensor *Hall Effect*

4.3.3 Motor Induksi 3 Fasa

Motor induksi 3 fasa yang digunakan dalam perancangan ini berguna sebagai penggerak *propeller*. Perputaran motor pada mesin arus bolak – balik ditimbulkan oleh adanya medan putar (fluks yang berputar) yang dihasilkan dalam kumparan statornya. Motor induksi 3 fasa ini juga dapat langsung terhubung ke Arduino mega tanpa menggunakan driver karena bekerja pada maksimum tegangan masukan 4,8 V dan dengan putaran yang konstan.

Parameter yang dibutuhkan dari motor induksi adalah pengaturan kecepatan dan torsi motor. Untuk itu dibutuhkan pengaturan yang fleksibel dengan cara mengubah frekuensi inputannya dari 50 Hz (Standar PLN) menjadi frekuensi yang diinginkan agar

motor dapat berputar pada kecepatan yang diinginkan. Gambar motor induksi 3 fasa diperlihatkan pada Gambar 4.5.



Gambar 4.5 Motor Induksi 3 Fasa

4.3.4 Variable Frequency Drive

Variable Frequency Drive pada perancangan ini digunakan sebagai inverter. *Variable Frequency Drive* (VFD) sering yang disebut dengan inverter adalah solusi aplikasi yang membutuhkan kemampuan pengaturan putaran motor sesuai bebannya atau sesuai dengan nilai yang diinginkan. Inverter ini pada peralatan elektronika daya berfungsi untuk mengubah listrik DC menjadi AC.

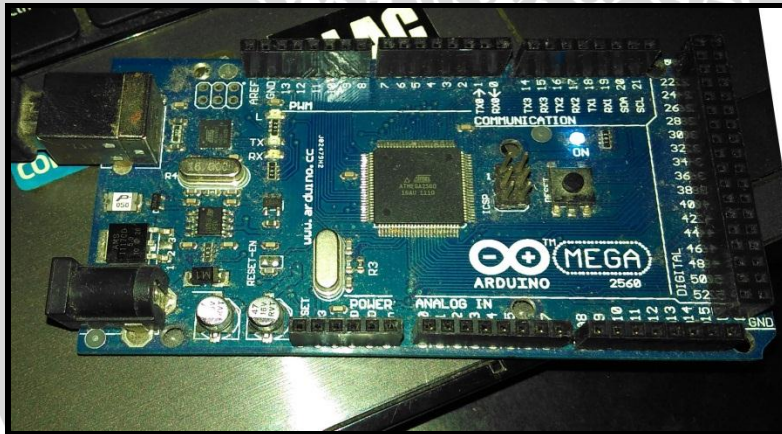
Prinsip kerja dari VFD pada dasarnya adalah mengubah tegangan bolak balik (AC) dari *input* menjadi tegangan searah (DC) oleh AC/DC *converter/ rectifier*, kemudian difilter untuk menghaluskan hasil penyearah tegangan tersebut. Setelah didapatkan listrik DC yang murni, langkah berikutnya adalah mengubah Listrik DC menjadi listrik AC dengan rangkaian inverter. Inverter sebenarnya berisi rangkaian *flip-flop* yang melakukan pensaklaran secara bergantian terhadap listrik DC sehingga menghasilkan listrik AC. Bentuk gelombang yang dihasilkan dengan rangkaian *inverter* bisa gelombang kotak atau gelombang sinus. Untuk menghasilkan Listrik AC dari *output* rangkaian *inverter* dengan gelombang sinus diperlukan rangkaian PWM (*Pulse Width Modulator*). Rangkaian ini yang akan mencacah listrik DC menjadi listrik AC dalam bentuk gelombang mendekati sinus dengan frekuensi tertentu sesuai keinginan. Gambar alat *Variable Frequency Drive* dapat dilihat pada Gambar 4.6.



Gambar 4.6 Variable Frequency Drive

4.3.5 Modul Arduino Mega 2560

Pada perancangan ini Arduino Mega 2560 digunakan sebagai pusat pengolah utama dalam melakukan proses pengaturan kecepatan putaran motor induksi 3 fasa untuk menggerakkan baling-baling (*propeller*). Gambar Arduino mega 2560 dapat dilihat pada Gambar 4.7.



Gambar 4.7 Arduino Mega 2560

Berikut tabel fungsi masing - masing pin Arduino Mega 2560 dapat dilihat pada Tabel 4.1

Tabel 4.1 Fungsi Pin Arduino Mega 2560

No	Pin	Fungsi
1	A0	Input analog
2	8	Input dari motor
3	GND	Jalur masukan <i>ground</i>
4	Vin	Jalur masukan tegangan

4.4 Perancangan Kontrol Logika Fuzzy (KLF)

Kontrol Logika Fuzzy yang digunakan dalam perancangan ini adalah KLF metode mamdani dengan *output singleton*.

4.4.1 Variabel Masukan dan Keluaran

Kontrol Logika Fuzzy dalam perancangan ini mempunyai dua *crisp input* yaitu berupa *Error* kecepatan putaran (rpm), *Change Error*, dan *crisp output*. Definisi *Error* dan *Change Error* dapat dilihat pada Persamaan 4-1 dan 4-2.

$$Error = SP - PV(t) \quad (4-1)$$

Di mana:

SP = *Setpoint* (nilai yang diinginkan)
 PV(t) = *Present Value* pada waktu t (nilai aktual)

$$Change Error = Error(t) - Error(t-1) \quad (4-2)$$

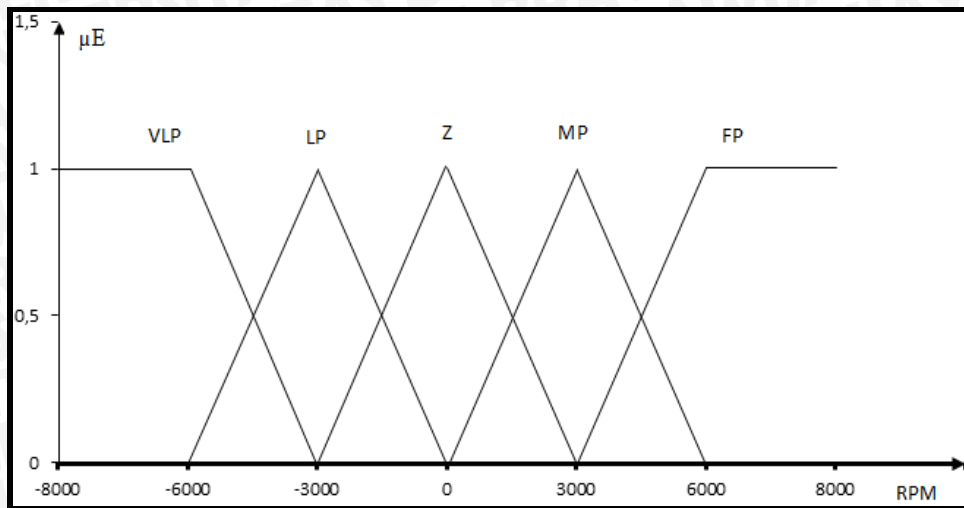
dimana:

Error (t) = *Error* pada waktu t
Error (t-1) = *Error* pada waktu (t-1)

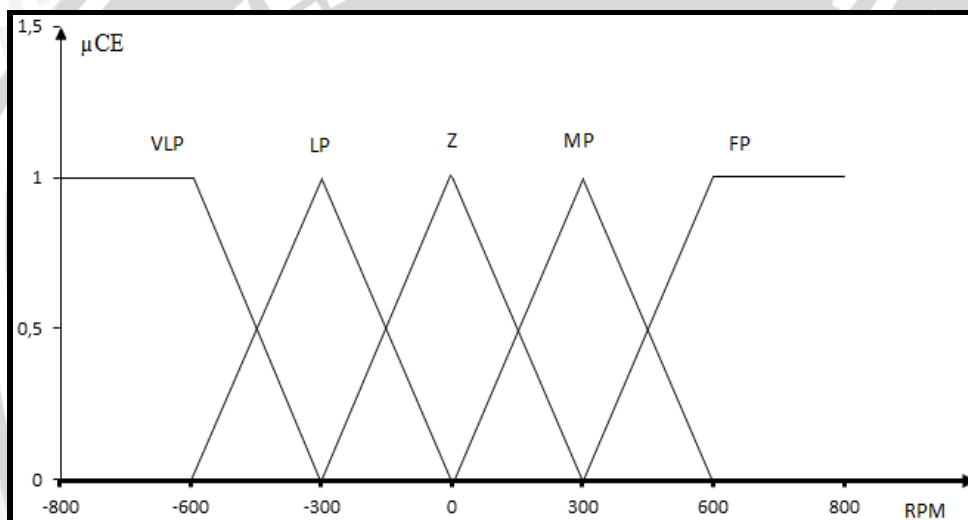
4.4.2 Fungsi Keanggotaan

Fungsi keanggotaan dari *Error* dan *Change Error* terdiri atas lima label, yaitu *Very Low Power* (VLP), *Low Power* (LP), *Zero* (Z), *More Power* (MP), dan *Full Power* (FP). *Error* adalah nilai *setpoint* dikurangi dengan nilai sebenarnya atau nilai dari *feedback*, sedangkan *change Error* didapat dari nilai *Error* rpm sekarang dikurangi dengan nilai *Error* rpm sebelumnya. Data masukan dari fungsi keanggotaan didapat dari nilai rpm kecepatan putaran *propeller* yang dikeluarkan oleh sensor *Hall Effect*.

Label masing-masing fungsi keanggotaan berjumlah 5 dengan tujuan agar *rule* (aturan) yang dihasilkan semakin banyak. Semakin banyak *rule* maka semakin banyak kemungkinan kondisi yang tercipta dan semakin banyak pula aksi kontrol yang akan dipilih. Gambar 4.8 dan 4.9 menunjukkan fungsi keanggotaan masukan *Error* dan *Change Error* yang digunakan.



Gambar 4. 8 Fungsi Keanggotaan Masukan *Error*



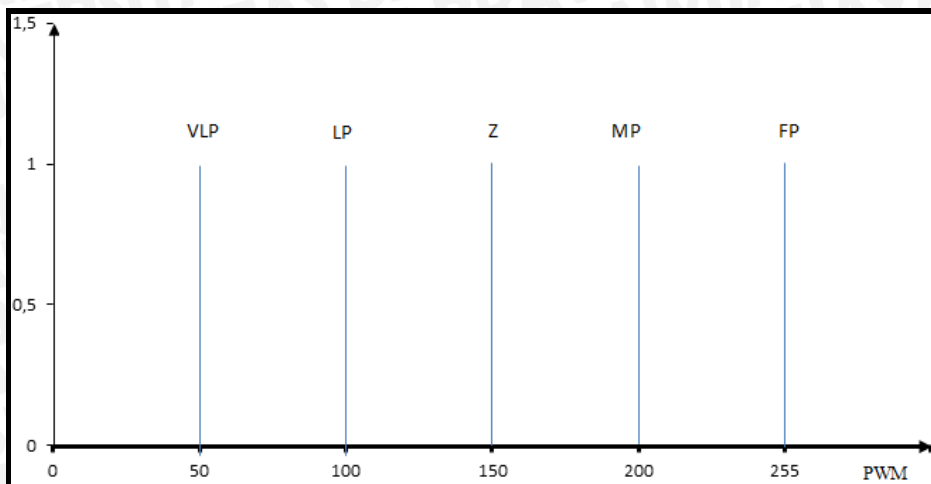
Gambar 4. 9 Fungsi Keanggotaan Masukan *Change Error*

4.4.3 Fungsi Keanggotaan Keluaran

Fungsi keanggotaan keluaran menunjukkan representasi catu daya motor *induksi 3 fasa* dan besarnya nilai PWM yang dikeluarkan oleh Arduino mega 2560 berupa hasil perhitungan menggunakan metode Defuzzifikasi *Weighted Average (WA)* yang telah dibulatkan dengan fungsi *integer* dalam pemrograman.

Dalam pemrograman dipilih frekuensi 290Hz untuk mencapai nilai kecepatan putaran *Propeller* yang diinginkan. Nilai frekuensi tersebut dipilih karena sesuai dengan keluaran kecepatan putaran *propeller* yang diinginkan yaitu dengan *setpoint* 8000 rpm.

Perancangan dalam *software* Arduino Mega 2560 disesuaikan dengan fungsi keanggotaan keluaran tersebut yaitu besar frekuensi yang diberikan. Fungsi keanggotaan keluaran ditunjukkan dalam Gambar 4.10.



Gambar 4. 10 Fungsi Keanggotaan Keluaran (PWM)

4.4.4 Perancangan Aturan Fuzzy

Aturan fuzzy (rule) digunakan sebagai penentu keluaran dari fuzzifikasi yang akan diolah dalam proses defuzzifikasi. Rule didapatkan dari 5 buah label dalam fungsi keanggotaan masukan Error dan change Error sehingga menghasilkan 25 macam rule. Tabel 4.2 merupakan aturan fuzzy (rule) yang digunakan.

Tabel 4. 2 Aturan Fuzzy (Rule)

CE \ E	VLP	LP	Z	MP	FP
VLP	FP	FP	FP	MP	Z
LP	FP	FP	MP	Z	LP
Z	FP	MP	Z	LP	VLP
MP	MP	Z	LP	VLP	VLP
FP	Z	LP	VLP	VLP	VLP

4.4.5 Metode Inferensi Max-Min

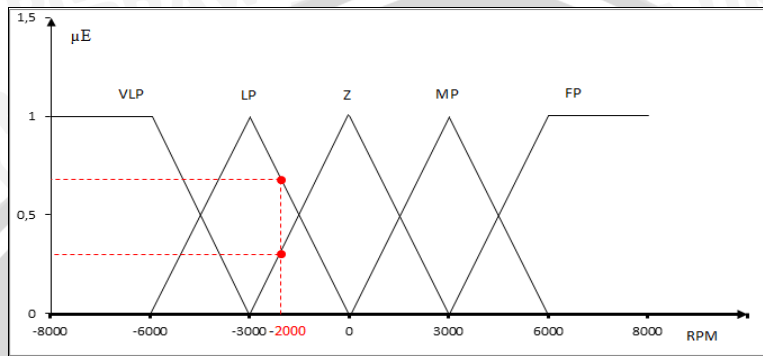
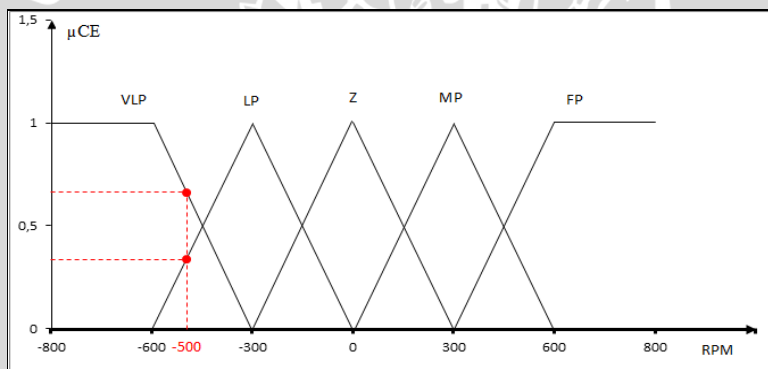
Sebagai pembuktian metode ini pada sistem perlu dilakukan beberapa perhitungan. Perhitungan hanya dilakukan pada hasil keluaran nilai crips Error dan Change Error tertentu dengan setpoint 8000 rpm sebagai perwakilan dari keseluruhan sistem. Berikut merupakan hasil perhitungan berdasarkan data dalam Tabel 4.3.

Tabel 4. 3 Tabel Hasil Keluaran

Sampel ke-	<i>Crisp E (rpm)</i>	<i>Crips CE (rpm)</i>
1	-2000	-500
2	1200	150

Perhitungan Fuzzyfikasi pada sistem:

- Perhitungan fuzzyfikasi pada sampel ke-1

**Gambar 4. 11 Fungsi Keanggotaan Masukan Pada Error (-2000)****Gambar 4. 12 Fungsi Keanggotaan Masukan Pada Change Error(-500)**

$$E/CE = -2000/-500$$

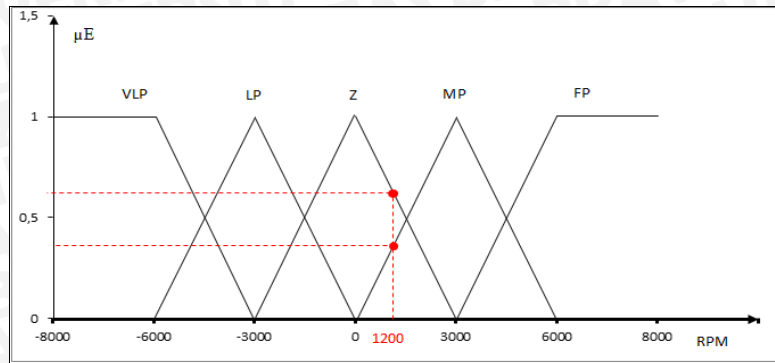
$$\mu_E (LP = -2000) = \{0 - (-2000)\} : 3000 = 0,67$$

$$\mu_E (Z = -2000) = \{-2000 - (-3000)\} : 3000 = 0,33$$

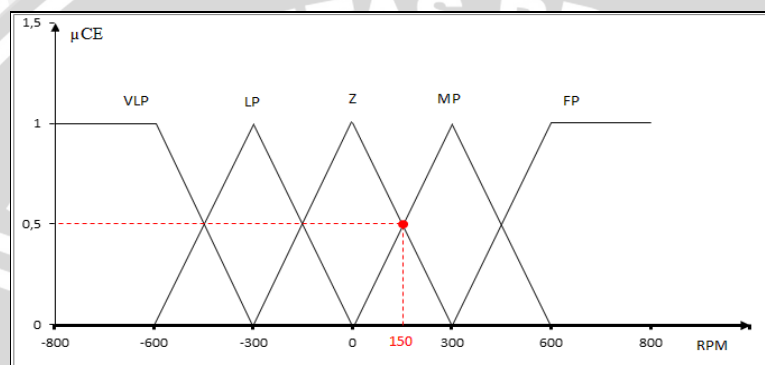
$$\mu_{CE} (VLP = -500) = \{-300 - (-500)\} : 300 = 0,67$$

$$\mu_{CE} (LP = -500) = \{-500 - (-600)\} : 300 = 0,33$$

- Perhitungan fuzzyfikasi pada sampel ke-2



Gambar 4. 13 Fungsi Keanggotaan Masukan Pada *Error* (1200)



Gambar 4. 14 Fungsi Keanggotaan Masukan Pada *Change Error*(150)

$$E/CE = 1200/150$$

$$\mu_E (Z=1200) = (3000-1200) : 3000 = 0,6$$

$$\mu_E (MP=1200) = (1200-0) : 3000 = 0,4$$

$$\mu_{CE} (Z=150) = (300-150) : 300 = 0,5$$

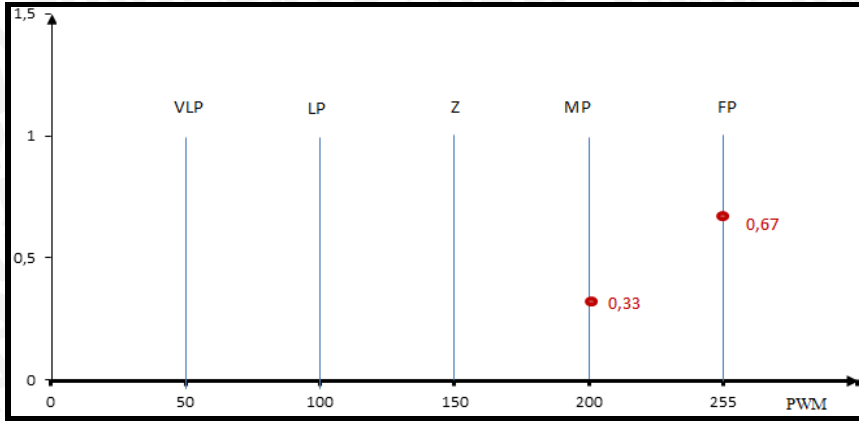
$$\mu_{CE} (MP=150) = (150-0) : 300 = 0,5$$

Perhitungan Inferensi pada sistem:

Perhitungan inferensi menggunakan *rule* yang telah ditentukan pada Tabel 4.2.

- Perhitungan inferensi pada sampel ke-1

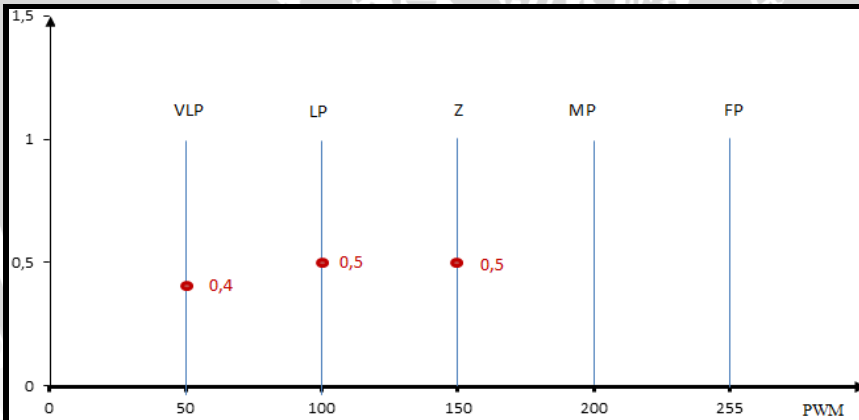
FP =0	FP =0,67	FP =0,33	MP =0	Z =0
FP =0	FP =0,33	MP =0,33	Z =0	LP =0
FP =0	MP=0	Z =0	LP =0	VLP=0
MP=0	Z =0	LP =0	VLP=0	VLP=0
Z =0	LP =0	VLP=0	VLP=0	VLP=0



Gambar 4. 15 Hasil Inferensi Pada Sampel ke-1

- Perhitungan inferensi pada sampel ke-2

FP =0	FP =0	FP =0	MP =0	Z =0
FP =0	FP =0	MP =0	Z =0	LP =0
FP =0	MP =0	Z =0,5	LP =0,4	VLP =0
MP =0	Z =0	LP =0,5	VLP =0,4	VLP =0
Z =0	LP =0	VLP =0	VLP =0	VLP =0



Gambar 4. 16 Hasil Inferensi Pada Sampel ke-2

4.4.6 Defuzzifikasi

Defuzzifikasi adalah proses untuk mendapatkan nilai crisp dari suatu himpunan fuzzy yang dihasilkan dari proses inferensi. Hasil dari proses defuzzifikasi ini yang akan digunakan untuk mengatur besar nilai PWM sebagai acuan dalam menggerakkan motor induksi 3 fasa. Metode defuzzifikasi yang digunakan adalah *Weighted Average (WA)*

- Defuzzifikasi untuk inferensi sampel ke-1

$$\begin{aligned}
 z^* &= \frac{\sum_{i=1}^n \mu_c(\bar{z}) \cdot \bar{z}}{\sum_{i=1}^n \mu_c(\bar{z})} \\
 &= \frac{0,33 [MP] + 0,67 [FP]}{0,33 + 0,67} \\
 &= \frac{0,33 [200] + 0,67 [255]}{0,33 + 0,67} \\
 &= 236,85
 \end{aligned}$$

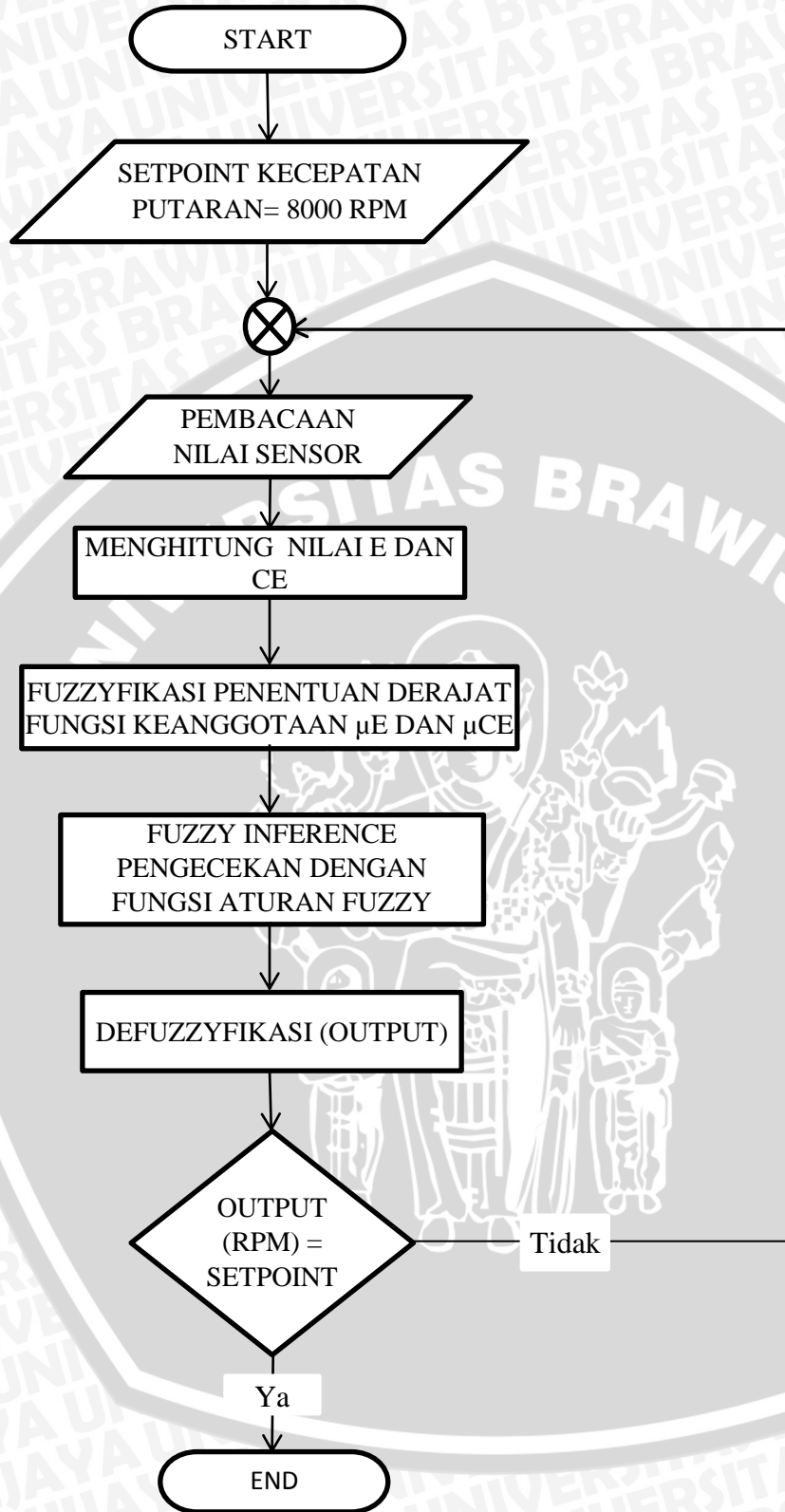
- Defuzzifikasi untuk inferensi sampel ke-2

$$\begin{aligned}
 z^* &= \frac{\sum_{i=1}^n \mu_c(\bar{z}) \cdot \bar{z}}{\sum_{i=1}^n \mu_c(\bar{z})} \\
 &= \frac{0,4 [VLP] + 0,5 [LP] + 0,5 [Z]}{0,4 + 0,5 + 0,5} \\
 &= \frac{0,4 [50] + 0,5 [100] + 0,5 [150]}{0,4 + 0,5 + 0,5} \\
 &= 103,57
 \end{aligned}$$

Berdasarkan hasil perhitungan defuzzifikasi di atas, dapat disimpulkan bahwa sistem sudah mampu menjalankan logika *fuzzy*. Hasil defuzzifikasi dalam perhitungan bukan merupakan bilangan bulat karena masih dalam perhitungan manual. Didalam program, keluaran yang diperintahkan ke *Variable frequency Drive* sudah dalam bentuk bilangan bulat karena perhitungan defuzzifikasi didalam program sudah dibulatkan oleh fungsi integer.

4.5 Perancangan Perangkat Lunak

Perancangan perangkat lunak pada pengontrolan ini menggunakan *software* Arduino Mega 2560. Untuk membuat program yang sesuai dengan tujuan penelitian, maka terlebih dahulu dibuat rancangan program dalam suatu diagram alir (*flowchart*) dari program yang dibuat agar alat bekerja berdasarkan program yang diinginkan. *Flowchart* perancangan perangkat lunak ditunjukkan pada Gambar 4.17.



Gambar 4.17 Flowchart Perangkat Lunak