

BAB IV

PERANCANGAN DAN PEMBUATAN ALAT

Dalam bab ini membahas mengenai perancangan dan pembuatan sistem pengendali kecepatan putaran *propeller* pada *wind tunnel* dengan *feedback* kecepatan aliran udara menggunakan kontroler PID. Perancangan perangkat tersebut meliputi perancangan perangkat keras maupun perancangan perangkat lunak. Sedangkan pembuatan bertujuan untuk menghasilkan semua perangkat pendukung maupun alat secara keseluruhan.

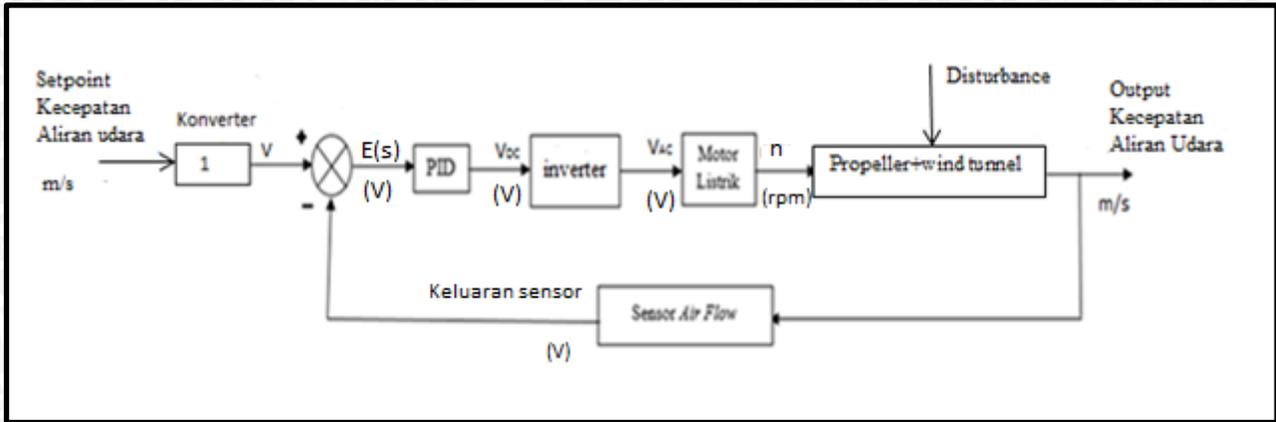
4.1 Perancangan Sistem

Perancangan alat ini dilakukan bertahap dalam bentuk diagram blok sehingga memudahkan dalam analisis pada setiap bloknnya maupun secara keseluruhan sistem. Perancangan ini terdiri atas:

1. Perancangan perangkat keras *propeller* (baling-baling) dan motor induksi 3 fasa sebagai penggerak aktuator.
2. Perancangan perangkat lunak (perancangan algoritma kontrol PID pada software arduino mega).

4.2 Diagram Blok Sistem

Dalam skripsi ini dibuat diagram blok agar dalam pengerjaan dapat dilakukan sesuai dengan rancangan sistem. Adapun diagram blok tersebut dapat dilihat pada Gambar 4.1:



Gambar 4. 1 Diagram Blok Sistem

Keterangan dari diagram blok dalam Gambar 4.1:

- Masukan / *setpoint* berupa kecepatan putaran diberikan melalui *program* pada Arduino Mega.
- Kemudian input diolah dan menghasilkan sinyal kontrol berupa PWM yang kemudian akan menjadi masukan untuk menggerakkan motor induksi 3 fasa.
- Sinyal dari motor induksi 3 fasa tadi kemudian menggerakkan *propeller* pada *wind tunnel* sehingga mengatur putaran sesuai *setpoint*.
- Keluaran putaran berupa aliran udara yang dihasilkan oleh putaran *propeller* tadi dibaca oleh sensor *air flow* yang kemudian diproses hingga mendapatkan keluaran dalam bentuk digital.
- Keluaran dari mekanik berupa aliran udara kemudian dibaca oleh sensor *air flow*. Keluaran dari sensor yang berupa analog kemudian di ubah menjadi digital oleh *converter* ADC.
- Hasil akhir pembacaan sensor kemudian di kurangkan dengan *input / setpoint* sehingga mikrokontroler mampu mengkompensasi *error* yang terjadi.

4.3 Perancangan Perangkat Keras

4.3.1 Propeller

Propeller yang digunakan adalah *propeller* dengan diameter 147 cm. Pemilihan ini didasarkan pada ukuran *wind tunnel* yaitu 150x150 cm. *Propeller* digunakan untuk menghasilkan aliran udara pada alat uji ini sehingga didapatkan parameter-parameter yang

akan diukur. Gambar *propeller* diperlihatkan dalam gambar 4.2. Sedangkan model mekanik alat ditunjukkan dalam gambar 4.3



Gambar 4.2 Propeller



Gambar 4.3 Model Alat

4.3.2 Sensor Air Flow

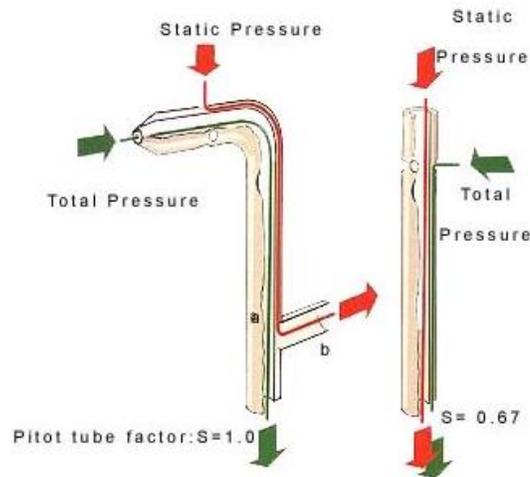
Air flow sensor adalah perangkat yang mengukur aliran udara, yaitu berapa banyak udara mengalir melalui tabung. Ini tidak mengukur volume udara yang lewat melalui

tabung, mengukur kecepatan yang sebenarnya dari udara yang mengalir melalui perangkat dalam segmen waktu yang ditetapkan. Dengan demikian sensor aliran udara ini hanya sebuah aplikasi pengukuran aliran pada suatu media khusus.

Tekanan total diukur dengan menggunakan pipa bagian dalam dari tabung *pitot* dan tekanan statis diukur dengan menggunakan pipa luar dari tabung *pitot*. Ujung tabung luar dan dalam disambungkan ke mikrokontroler. Tekanan kecepatan (yaitu perbedaan antara tekanan total dan tekanan statis), dikonversi menjadi tegangan masukan bagi sensor. Pipa yang mengukur tekanan statis terletak secara radial pada batang yang dihubungkan ke mikrokontroler (pstat). Tekanan pada ujung pipa di mana fluida masuk merupakan tekanan stagnasi (p_0). Gambar diagram aliran udara yang masuk pada pipa ditunjukkan dalam Gambar 4.4.

$$p_o = p_{stat} + \frac{1}{2} \rho V^2 \quad 4-1$$

$$V = \frac{\sqrt{2(p_o - p_{stat})}}{\rho} \quad 4-2$$



Gambar 4.4 Diagram aliran udara pada tabung pitot

Sumber: <https://yefrichan.wordpress.com/2010/08/02/cara-menghitung-daya-blowerfan/>

Sensor yang digunakan dalam pengujian ini ditunjukkan dalam gambar 4.5.



Gambar 4.5 Sensor Air Flow

4.3.3 Motor Induksi 3 Fasa

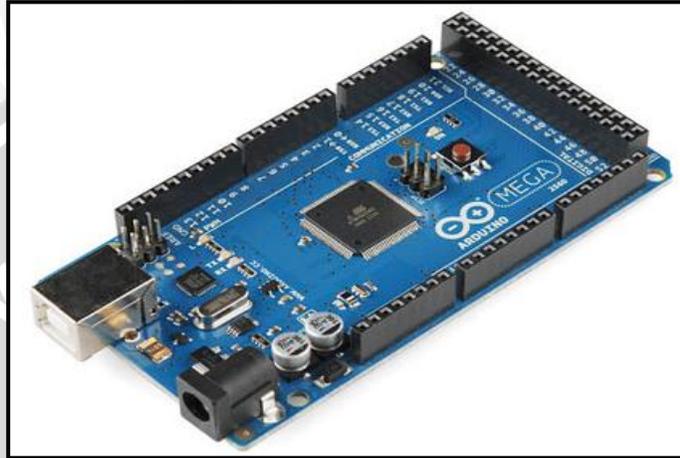
Motor induksi 3 fasa yang digunakan dalam perancangan kali ini berguna sebagai penggerak baling – baling (*propeller*). Motor induksi 3 fasa ini juga dapat langsung terhubung ke Arduino mega tanpa menggunakan driver karena bekerja pada maksimum tegangan masukan 4,8 V dan dengan putaran yang konstan. Gambar motor induksi 3 fasa diperlihatkan pada gambar 4.6.



Gambar 4.6 Motor Induksi 3 Fasa

4.3.4 Modul Arduino Mega 2560

Pada alat ini digunakan Arduino Mega 2560 sebagai pengolah dalam proses pengaturan putaran motor induksi 3 fasa dalam menggerakkan *propeller*. Tampak depan Arduino mega 2560 ditunjukkan dalam Gambar 4.7.



Gambar 4.7 Tampak depan Arduino mega 2560

Arduino Mega 2560 adalah merupakan *board* mikrokontroler berbasis ATmega 2560. Modul ini memiliki 54 digital *input/output* di mana 14 digunakan untuk *output* PWM dan 16 digunakan sebagai analog *input*, 4 untuk UART, 16 MHz osilator kristal, koneksi USB, *power jack*, *ICSP Header*, dan tombol reset.

Modul ini memiliki segalanya yang dibutuhkan untuk memprogram mikrokontroler seperti kabel USB dan sumber daya melalui Adaptor ataupun baterai. Pin masukan dan keluaran Arduino Mega 2560 pada perancangan ini akan difungsikan sesuai Tabel 4.1

Tabel 4.1 Fungsi Pin Arduino Mega 2560

No	Pin	Fungsi
1	A0	Masukan <i>Pre-Amplifier</i>
2	8	Masukan motor
3	GND	Jalur masukan GND seluruh sistem
4	Vin	Jalur masukan 5V seluruh sistem

4.3.5 Variable Frequency Drive

Dalam penelitian ini, *Variable Frequency Drive* digunakan sebagai *inverter*. *Inverter* adalah suatu peralatan elektronika daya yang berfungsi untuk mengubah listrik DC menjadi AC. *Inverter* seringkali disebut sebagai *Variable Frequency Drive* (VFD).

Variable Frequency Drive mengubah input motor (listrik AC) menjadi DC dan kemudian dijadikan AC lagi dengan frekuensi yang dikehendaki sehingga motor dapat dikontrol sesuai dengan kecepatan yang diinginkan. Untuk mengubah tegangan AC menjadi DC dibutuhkan penyearah (*converter AC-DC*) dan biasanya menggunakan penyearah tidak terkendali (*rectifier dioda*) namun juga ada yang menggunakan penyearah terkendali (*thyristor rectifier*). Setelah tegangan sudah diubah menjadi DC maka diperlukan perbaikan kualitas tegangan DC dengan menggunakan tandon kapasitor sebagai perata tegangan. Kemudian tegangan DC diubah menjadi tegangan AC kembali oleh *inverter* dengan teknik PWM (*Pulse Width Modulation*). Setelah ini bisa didapatkan amplitudo dan frekuensi keluaran yang diinginkan. Gambar alat *Variable Frequency Drive* diperlihatkan pada gambar 4.8.



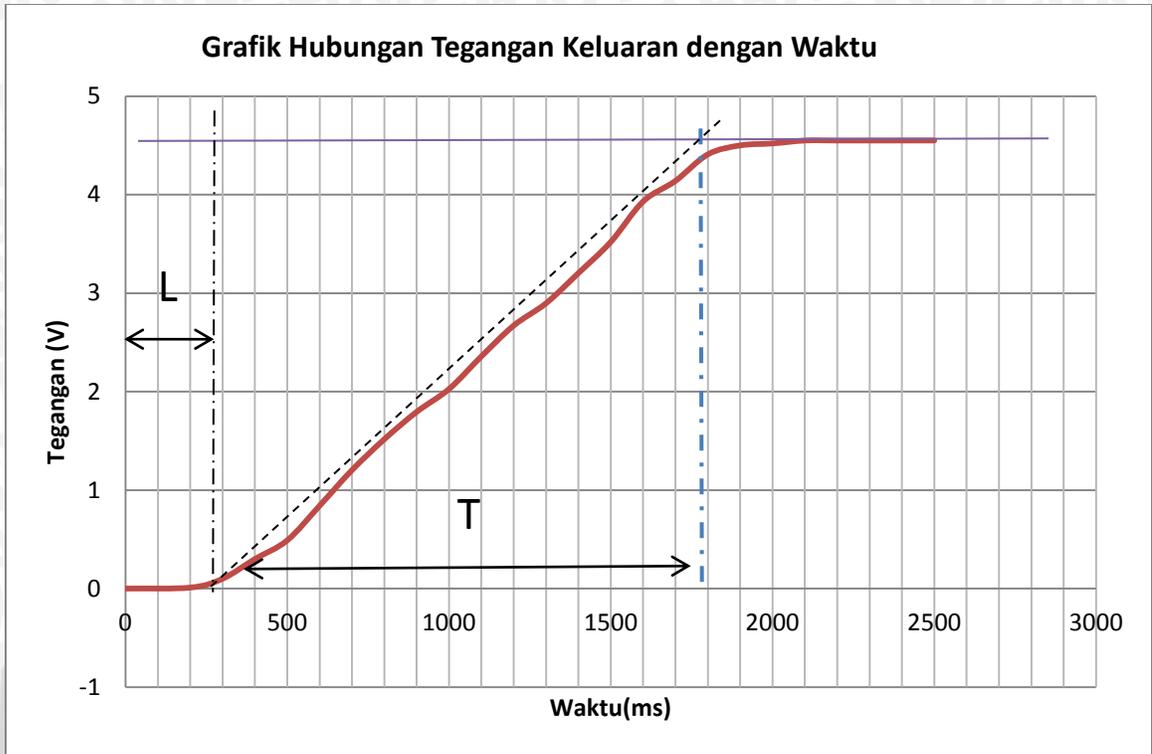
Gambar 4.8 Bentuk fisik *Variable Frequency Drive*

4.4 Perancangan Kontrol PID

Pengambilan data dilakukan dengan pembacaan sensor yang masuk dalam serial monitor Arduino mega. *Tuning* kontroler PID adalah dengan menggunakan metode *Ziegler-Nichols 1* yang telah dimasukkan pada Arduino Mega yang digunakan. Metode ini dipilih karena respon *plant* yang menghasilkan kurva berbentuk S. Kurva tanggapan *plant* digunakan untuk mencari waktu tunda L dan konstanta waktu T yang diperlihatkan dalam Gambar 4.9 sehingga diperoleh nilai dari tiga buah parameter yang terdapat pada kontroler PID yaitu konstanta proporsional (K_p), konstanta integral (K_i) dan konstanta diferensial (K_d).

Dari hasil pengujian *open loop* yang diperlihatkan dalam Gambar 5.3 pada bab selanjutnya berupa kurva S akan digunakan untuk menentukan parameter *tuning* PID dengan metode 1 *Ziegler-Nichols*. Adapun langkah-langkah yang dilakukan sebagai berikut:

1. Menarik garis tangen pada titik infleksi grafik karakteristik *open loop* seperti yang diperlihatkan dalam Gambar 4.9
2. Menentukan perpotongan garis tangen terhadap sumbu waktu t untuk mendapatkan L
3. Menentukan perpotongan garis tangen terhadap sumbu *steady* untuk mendapatkan nilai T
4. Nilai L dan T digunakan untuk menentukan nilai K_p K_i dan K_d sesuai dengan Tabel 4.2.



Gambar 4.9 Metode 1 Ziegler-Nichols (hasil pengujian)

Tabel 4.2 Aturan Metode 1 Ziegler-Nichols (Ogata K., 1997)

Tipe Kontrol	K_p	τ_i	τ_d
P	$\frac{\tau}{L}$	∞	0
PI	$0,9 \frac{\tau}{L}$	$\frac{L}{0,3}$	0
PID	$1,2 \frac{\tau}{L}$	$2L$	$0,5L$

Dalam Gambar 4.9, besarnya *steady state* ketika keluaran sensor adalah maksimal yaitu sebesar 4,5 Volt yang ekivalen dengan 8000 RPM. sehingga diperoleh besarnya $L = 300\text{ms}$ dan $T = 1800 - 300 = 1500\text{ms}$.

Dengan demikian parameter kontroler diperoleh sebagai berikut:

$$K_p = 1,2 \frac{\tau}{L} = 1,2 \times \frac{1,400\text{S}}{0,300\text{S}} = 4,67$$

$$\tau_i = 2L = 2 \times 0,300S = 0,6$$

$$\tau_d = 0,5L = 0,5 \times 0,300S = 0,15$$

Selanjutnya akan diperoleh K_i dan K_d sebagai berikut:

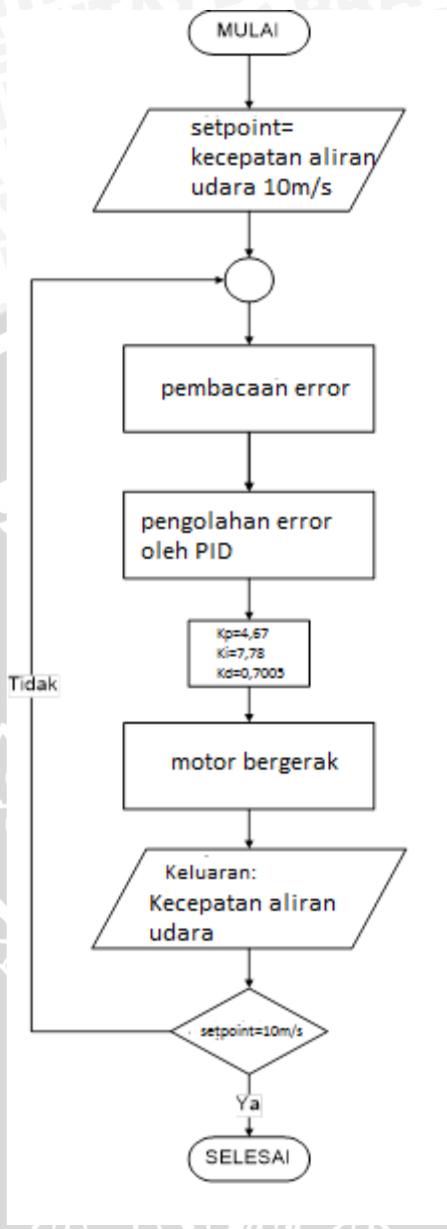
$$K_i = \frac{Kp}{\tau_i} = \frac{4,67}{0,6} = 7,78$$

$$K_d = Kp \times \tau_d = 4,67 \times 0,2 = 0,7005$$

4.5 Perancangan Perangkat Lunak

Perancangan perangkat lunak pada pengendalian ini menggunakan bahasa pemrograman C++ dengan menggunakan *software* Arduino ERW 1.0.5. *Flowchart* perancangan perangkat lunak dapat dilihat pada Gambar 4.10. Penjelasan *flowchart* dalam Gambar 4.10 sebagai berikut:

- *Setpoint* berupa kecepatan aliran udara sebesar 10 m/s
- Keluaran dari sensor dibandingkan dengan *setpoint* menghasilkan error dimana *error* adalah selisih antara *setpoint* dengan *output*
- Kemudian oleh PID, *error* dikurangi dengan memasukkan nilai-nilai parameter PID
- Keluaran dari PID menghasilkan masukan bagi VFD sehingga motor bergerak pada kecepatan tertentu
- Motor menggerakkan *propeller* sehingga menghasilkan aliran udara dengan kecepatan tertentu
- Jika kecepatan aliran udara belum sesuai dengan *setpoint* maka sistem kembali ke pembacaan *error* hingga diperoleh *error* terkecil, jika sudah diperoleh *error* terkecil maka sistem sudah berjalan sesuai dengan *setpoint* dengan baik.



Gambar 4.10 Flowchart Perangkat Lunak

