

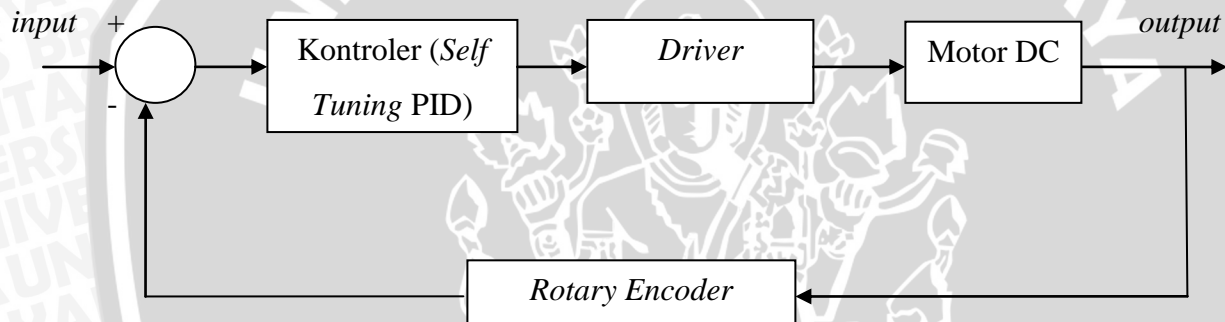
BAB IV

PERANCANGAN DAN PEMBUATAN ALAT

Pada bab ini akan dijelaskan mengenai perancangan dan pembuatan alat yang terdiri dari dua bagian, yaitu perancangan dan pembuatan perangkat keras serta perancangan dan pembuatan perangkat lunak. Perancangan dan pembuatan alat dilakukan secara bertahap untuk memudahkan analisis sistem.

4.1 Diagram Blok Sistem

Pada perancangan alat diperlukan perancangan blok diagram sistem yang dapat menjelaskan sistem secara garis besar dan diharapkan alat dapat bekerja sesuai dengan rencana. Blok diagram tersebut dapat dilihat dari Gambar 4.1 berikut.



Gambar 4.1 Blok Diagram Sistem

Sumber: Perancangan

Keterangan dari blok diagram diatas adalah sebagai berikut :

- Sensor yang digunakan berupa *rotary encoder* yang berfungsi mendeteksi kecepatan motor pada saat motor bergerak dan akan memberikan masukan berupa pulsa ke Arduino.
- Pusat pengendalian sistem menggunakan Arduino UNO yang memberikan keluaran berupa *duty cycle* PWM kepada *driver* motor.
- Motor yang digunakan adalah motor DC yang berfungsi sebagai aktuator.
- *Driver* motor menggunakan modul EMS 5A H-Bridge.

4.2 Prinsip Kerja Alat

Cara kerja alat adalah sebagai berikut :

- Menggunakan catu daya sebesar 12 volt untuk motor DC dan 5 volt untuk sensor *rotary encoder*.

- *Rotary Encoder* sebagai sensor kecepatan motor DC. *Rotary encoder* akan memberikan keluaran berupa pulsa yang berubah-ubah sesuai dengan perubahan kecepatan.
- Tahap pertama yaitu ketika sistem diaktifkan motor akan berputar, sensor akan mendeteksi kecepatan yang terbaca kemudian memberikan masukan ke arduino berupa pulsa.
- Tahap berikutnya yaitu pulsa dari sensor akan diolah di Arduino yang akan memberikan sinyal kontrol berupa *Pulse Width Modulation (PWM)* ke *driver* motor agar motor berputar sesuai dengan besarnya PWM yang diberikan.

4.3 Perancangan Perangkat Keras (Hardware)

4.3.1 Spesifikasi Alat

Spesifikasi alat yang dirancang adalah sebagai berikut:

1. Motor DC dengan catu daya 12 volt.
2. Sensor yang digunakan untuk mendeteksi kecepatan adalah *rotary encoder* dengan resolusi 48.
3. Kontroler yang digunakan adalah kontroler *self tuning* PID.
4. Menggunakan satu buah mikrokontroler Arduino Uno Rev3.
5. Software yang digunakan sebagai pemrograman yaitu Arduino ERW 1.0.5 dan MATLAB 2013a.

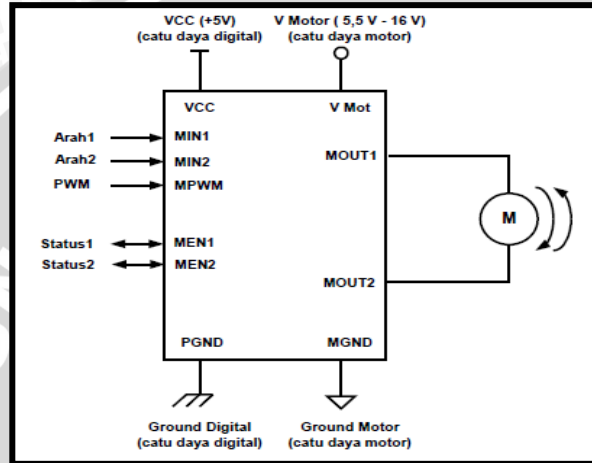
Perancangan perangkat keras dapat dilihat pada Gambar 4.2



Gambar 4.2 Rancangan Plant Motor DC

4.3.2 Perancangan *Driver Motor DC*

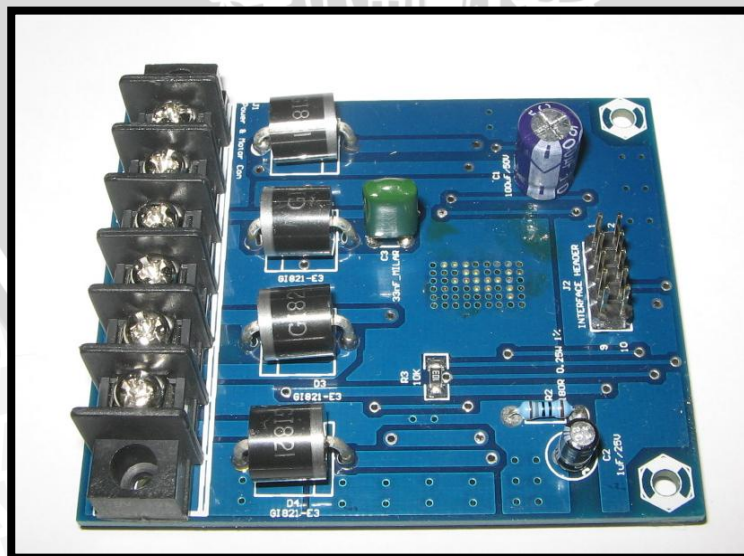
Modul pengendali motor DC yang digunakan adalah modul EMS 5A *H-Bridge*. Secara garis besar, fungsi modul pengendali motor ini adalah untuk mengendalikan arah dan kecepatan putaran motor DC sesuai instruksi kendali dari Arduino Uno ERW 1.0.5. Gambar koneksi modul pengendali motor DC *EMS 5A H-Bridge* ditunjukkan dalam Gambar 4.3



Gambar 4.3 Koneksi Modul EMS 5A H-Bridge

Sumber: *EMS 5A H-Bridge datasheet*

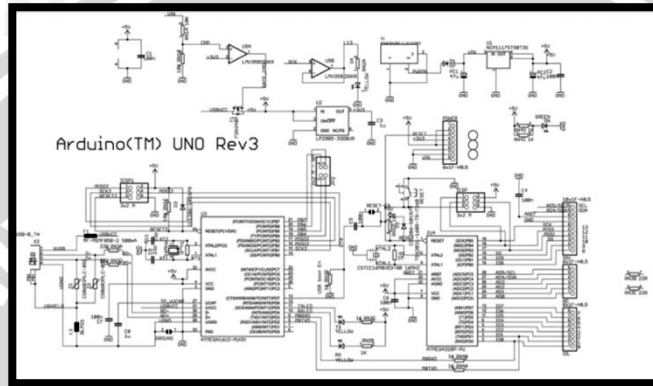
Rangkaian *driver* ini sanggup bekerja dengan tegangan maksimal 40 volt, serta kapasitas arus maksimum yang dapat dilewatkan pada modul ini sebesar 5 ampere. Gambar modul rangkaian driver EMS 5A *H-Bridge* dapat dilihat pada Gambar 4.4



Gambar 4.4 Modul Rangkaian *Driver* EMS 5A *H-Bridge*

4.3.3 Modul Arduino Uno Rev.3

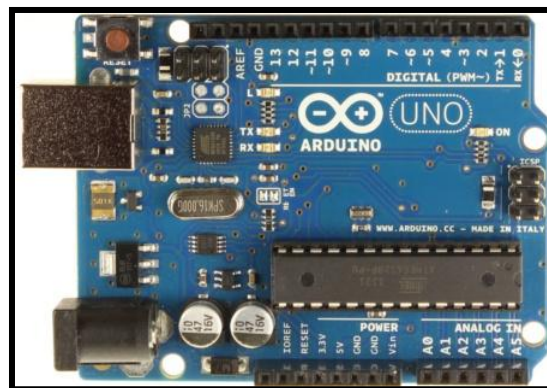
Pada sistem pengaturan kecepatan putaran alat penyiraman tanaman ini digunakan Arduino Uno Rev.3 sebagai pengolah data dalam proses pengaturan kecepatan terdapat pada Driver motor *H-Bridge* untuk menggerakkan motor DC. Konfigurasi kaki I/O dari Arduino Uno Rev3 ditunjukkan dalam Gambar 4.5.



Gambar 4.5 Desain Sistem Arduino Uno Rev3

Sumber : Datasheet Arduino Rev3

Arduino Uno adalah *board* mikrokontroler berbasis ATmega328. Memiliki 14 pin *input* dari *output* digital dimana 6 pin *input* tersebut dapat digunakan sebagai *output* PWM dan 6 pin *input* analog, 16 MHz osilator kristal, koneksi USB, *jack power*, ICSP *header*, dan tombol *reset*. Untuk mendukung mikrokontroler agar dapat digunakan, cukup hanya menghubungkan *Board* Arduino Uno ke komputer dengan menggunakan kabel USB atau listrik dengan AC ke adaptor DC atau baterai untuk menjalankannya. Modul arduino uno ditunjukkan pada Gambar 4.6.



Gambar 4.6 Modul Arduino Uno

Tabel 4.1 Fungsi Pin Arduino Uno Rev3

No	Pin	Fungsi
1	2	Jalur masukan data dari sensor
2	5	Jalur Masukan PWM
3	7	Pin input untuk MOUT1
4	8	Pin input untuk MOUT2
5	9	Pin enable untuk MOUT1
6	5V	Jalur keluaran 5V
7	GND	Jalur keluaran ground

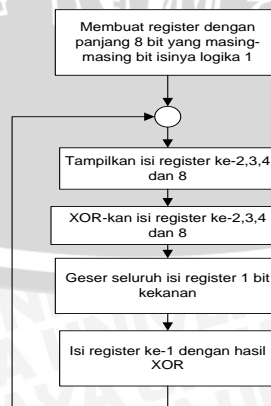
Tabel 4.1 menjelaskan pin-pin yang digunakan pada Arduino Uno Rev3.

4.4 Perancangan Perangkat Lunak

4.4.1 Perancangan Identifikasi *Plant*

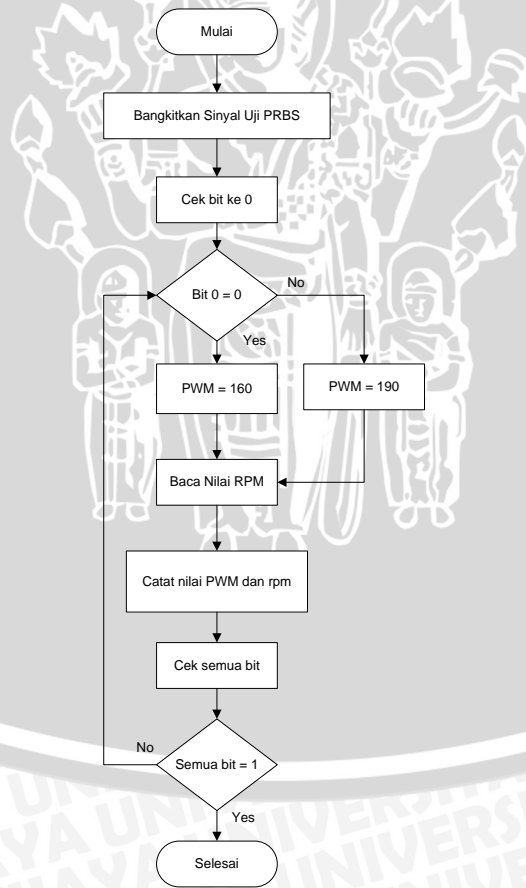
Pada skripsi ini sinyal uji yang digunakan adalah jenis *Pseudo Random Binary Sequence* (PRBS) dengan jumlah bit 8. Sinyal uji ini akan dibangkitkan oleh mikrokontroler. Karena digunakan panjang register 8 bit maka panjang sekuensial yang akan dihasilkan adalah 255 bit untuk 1 kali proses generasi sinyal uji PRBS.

Proses pembentukan sinyal PRBS ditunjukkan oleh Gambar 4.7



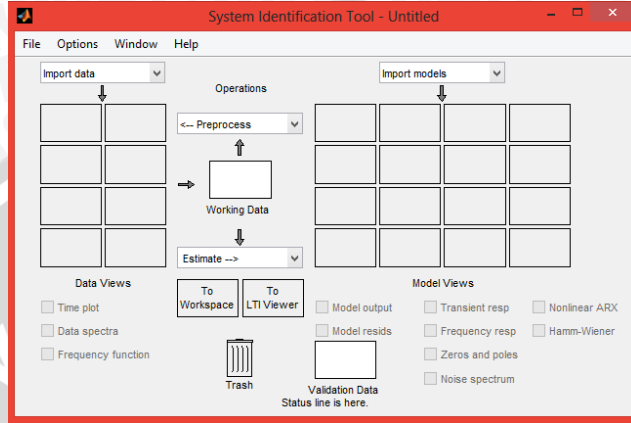
Gambar 4.7 Diagram alir pembentukan sinyal PRBS

Perangkat lunak pengambilan data *input-output* ini bertugas membangkitkan sinyal uji PRBS, mengambil data respon plant, dan mengirimkan balik ke komputer. Agar dapat melaksanakan tugas-tugas tersebut maka perlu diatur waktu kerja dari mikrokontroler (MK). Pertama yang dilakukan MK adalah menyiapkan register sebanyak 8 bit untuk proses pembangkitan sinyal PRBS. Setelah 8 bit register siap maka MK akan memulai membangkitkan sinyal PRBS, namun keseluruhan sekuensial sinyal PRBS tidak langsung terbentuk sekali proses selesai melainkan secara bertahap tiap bit. Setelah isi logika bit ke-8 dari register dikeluarkan disalahsatu pin MK selanjutnya MK akan melakukan proses PRBS selanjutnya, namun di antara proses itu MK diberi tugas mengambil data respon *plant* dari ADC dan mengirimkan ke komputer (proses ambil data dan kirim data dilakukan beberapa kali sesuai *delay* waktu yang diinginkan). Setelah rutin ambil data dan kirim data selesai MK akan melanjutkan proses PRBS. Proses ini akan berulang terus. *Flowchart*nya ditunjukkan oleh Gambar 4.8



Gambar 4.8 Diagram Alir Pengambilan Data Input-Output

Setelah mendapat pasangan data input dan output maka langkah selanjutnya adalah pemilihan struktur model, estimasi parameter dan validasi model akan diprogramkan pada software MATLAB menggunakan *System Identification Toolbox* (Gambar 4.9)



Gambar 4.9 *System Identification Toolbox* pada MATLAB

4.4.2 Perancangan *Self Tuning* PID

Karena dibutuhkan parameter a_1 , a_2 , b_1 , b_2 sebagai *input*, maka tahap pertama yang dilakukan identifikasi sistem, struktur model yang dipilih adalah *Auto Regresive with Exogenous input* (ARX) derajat 2. Secara umum, persamaan ARX dapat dituliskan sebagai berikut:

$$A(q)y(k) = B(q)u(k - nk) + e(k) \tag{4.1}$$

dengan: $A(q) = 1 + a_1q^{-1} + a_2q^{-2} \dots + a_nq^{-na}$

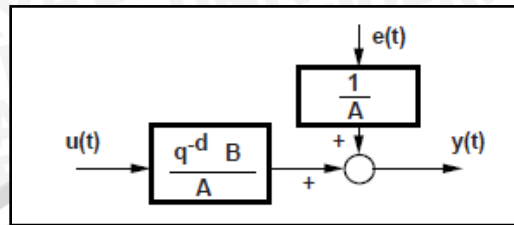
$$B(q) = b_1q^{-1} + b_2q^{-2} \dots + b_nq^{-nb}$$

$y(k)$ = keluaran

$u(k)$ = masukan

$e(k)$ = gangguan

atau dalam bentuk diagram seperti pada Gambar 4.10



Gambar 4.10 Struktur Model ARX

Sumber: Landau, 2006

Setelah mendapatkan nilai a_1 , a_2 , b_1 , b_2 , langkah selanjutnya adalah menghitung nilai K_{pu} dan T_u untuk mendapatkan nilai K_p , T_I , T_D . Pencarian nilai K_{pu} dan T_u didapat melalui perhitungan sebagai berikut:

$$K_c = \frac{1-a_2}{b_2} \tag{4.2}$$

$$\alpha = \frac{a_1 + K_c \cdot b_1}{-2} \tag{4.3}$$

Jika $\alpha < -1$ atau $\alpha > 1$, maka nilai $K_c = 0$.

$$K_r = \frac{1-a_1+a_2}{b_1-b_2} \tag{4.4}$$

$$f = -(a_1 + K_r \cdot b_1 - 1) \tag{4.5}$$

Jika $f < -1$ atau $f > 1$, maka nilai $K_r = 0$.

Setelah nilai K_c , α , K_r , f didapat, maka nilai K_{pu} dan T_u didapat dengan persyaratan sebagai berikut.

- Jika $K_c > 0$ atau $K_c < 0$ dan $K_r \leq 0$, maka nilai K_{pu} dan T_u adalah

$$K_{pu} = K_c \tag{4.6}$$

$$T_u = \frac{T_0 \cdot 2 \cdot \pi}{\text{acos}(\alpha)} \tag{4.7}$$

- Jika $K_r \neq 0$, maka nilai K_{pu} dan T_u adalah

$$K_{pu} = K_r \quad (4.8)$$

$$T_u = T_0 \cdot 2 \quad (4.9)$$

- Jika kedua persyaratan di atas tidak terpenuhi, maka nilai $K_{pu} = 1$ sedangkan nilai $T_u = 2 \cdot T_0$

Dari hasil nilai K_{pu} dan T_u yang telah diperoleh, maka nilai K_p , T_I , T_D adalah sebagai berikut:

$$K_p = 0,6 \cdot K_{pu} \quad (4.10)$$

$$T_I = \frac{T_u}{2} \quad (4.11)$$

$$T_D = \frac{T_u}{8} \quad (4.12)$$

Setelah didapat nilai K_p , T_I , T_D , untuk mendapatkan nilai fungsi alih kontroler maka perlu didapatkan nilai q_0 , q_1 , q_2 , p_1 , p_2 , dengan cara sebagai berikut:

$$q_0 = K_p \left(1 + \frac{T_0}{T_I} + \frac{T_D}{T_0}\right) \quad (4.13)$$

$$q_1 = -K_p \left(1 + 2 \frac{T_D}{T_0}\right) \quad (4.14)$$

$$q_2 = K_p \frac{T_D}{T_0} \quad (4.15)$$

$$p_1 = -1 \quad (4.16)$$

$$p_0 = 0 \quad (4.17)$$

fungsi alih kontroler dalam domain z dituliskan sebagai:

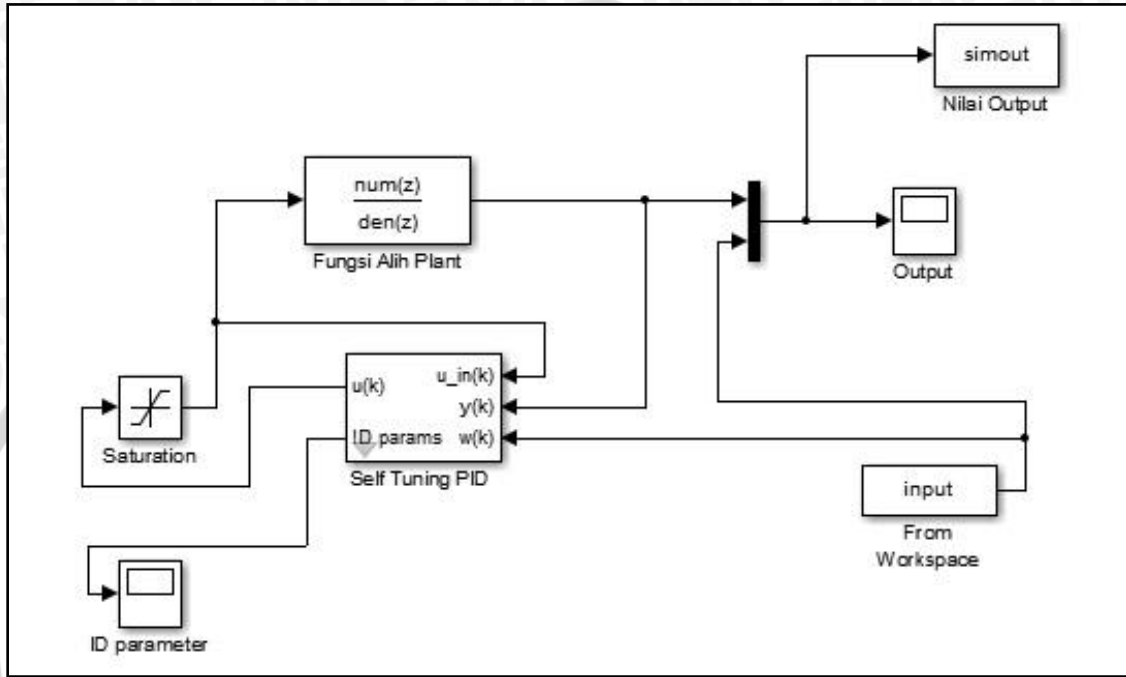
$$u(z) = \frac{q_0 + q_1 z^{-1} + q_2 z^{-2}}{1 + p_1 z^{-1} + p_2 z^{-2}} \quad (4.18)$$

atau dapat ditulis menjadi:

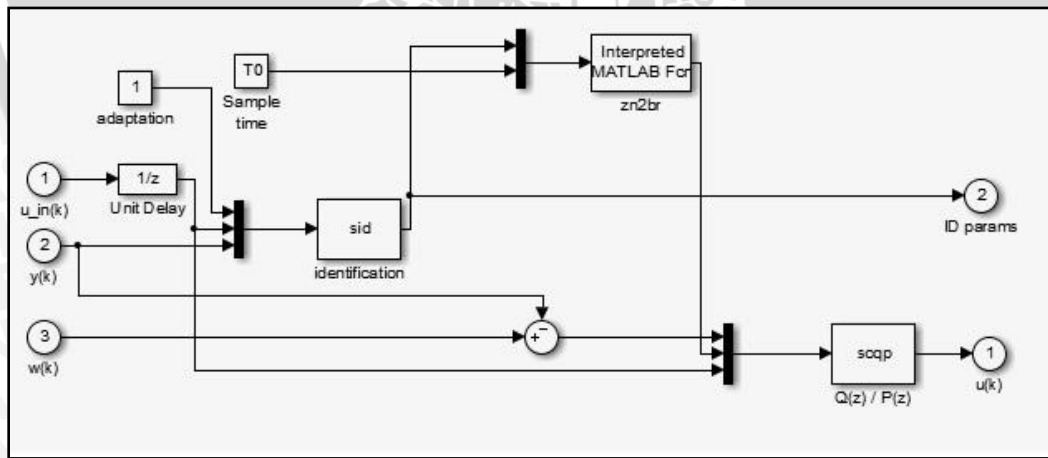
$$u_k = q_0 e_k + q_1 e_{(k-1)} + q_2 e_{(k-2)} + p_1 u_{(k-1)} + p_2 u_{(k-2)} \quad (4.19)$$

4.4.3 Perancangan Simulasi sistem

Untuk perancangan simulasi sistem, akan menggunakan program Simulink pada MATLAB. Perancangan blok diagram akan ditunjukkan seperti pada Gambar 4.11 dan Gambar 4.12



Gambar 4.11 Perancangan Blok Diagram Simulink pada MATLAB



Gambar 4.12 Blok Skema *Self Tuning* PID Kontroler Metode *Backward Rectangular*