

BAB I PENDAHULUAN

1.1 Latar Belakang

Pencahayaan merupakan salah satu faktor penting dalam perancangan ruang. Ruangan yang telah dirancang tidak dapat memenuhi fungsinya dengan baik apabila tidak disediakan akses pencahayaan. Pencahayaan di dalam ruangan memungkinkan orang yang menempatnya dapat melihat dengan baik. Tanpa dapat melihat dengan jelas maka aktivitas di dalam ruang akan terganggu. Sebaliknya, cahaya yang terlalu terang juga dapat mengganggu penglihatan. Permasalahan lainnya juga ada pada penghematan energi, dimana energi merupakan hal yang sangat penting bagi kehidupan manusia saat ini.

Salah satu contoh penyebabnya adalah pemborosan penggunaan energi pada sistem penerangan. Oleh karena itu dibuatlah suatu alat yang dapat mengurangi pemborosan energi pada sistem penerangan atau pencahayaan pada ruangan. Alat tersebut diberi nama *solatube*, yang berfungsi untuk memberikan pencahayaan langsung dari cahaya matahari tanpa menggunakan energi listrik, sehingga pada siang hari kita tidak memerlukan pencahayaan dari lampu. Oleh karena itu energi listrik akan dapat dihemat. Namun, *solatube* ini hanya dapat merefleksikan cahaya matahari ke dalam ruangan tanpa bisa diatur intensitas kuat cahaya yang masuk ke dalam ruangan tersebut. Sehingga ada kalanya cahaya yang masuk ke dalam ruangan terlalu kuat atau terlalu terang.

Maka pada penelitian ini saya akan memberikan suatu solusi yakni dengan membuat *solatube* otomatis yang dapat mengatur intensitas kuat cahaya, sehingga cahaya yang masuk sesuai dengan yang dibutuhkan atau yang diinginkan. Sistem pada penelitian ini berbasis Mikrokontroler. Kontrol PID digunakan pada sistem kontrol pengendalian motor dalam mengatur plat sebagai alat untuk mengatur intensitas kuat cahaya yang masuk ke dalam ruangan. Diharapkan dengan menggunakan Kontrol PID berbasis Mikrokontroler ini, pergerakan motor yang mengatur plat dapat dikendalikan.

1.2 Rumusan Masalah

Mengacu pada permasalahan yang telah diuraikan pada latar belakang, maka rumusan masalah dapat ditekankan pada beberapa poin berikut:

1. Bagaimana merancang dan membuat perangkat keras (*hardware*) sistem pengaturan kuat cahaya pada *solatube* otomatis?
2. Bagaimana membuat suatu program perangkat lunak (*software*) untuk pengaturan kuat cahaya pada *solatube* otomatis dengan menggunakan Kontrol PID yang dapat menghasilkan respon yang sesuai?

1.3 Batasan Masalah

Batasan masalah yang digunakan pada penelitian kali ini meliputi:

1. *Solatube* yang digunakan untuk penelitian merupakan rancangan yang dibuat dengan desain sendiri.
2. Sensor yang digunakan adalah sensor cahaya LDR.
3. Kisaran kuat cahaya yang digunakan dalam penelitian kali ini yakni sebesar 150 lux.
4. Pembahasan ditekankan pada pengendalian motor untuk menggerakkan plat pada sistem dengan menggunakan Kontrol PID berbasis Mikrokontroler.

1.4 Tujuan

Merancang sistem pengaturan atau pengendali kuat cahaya pada *solatube* dengan kontrol PID dengan menggunakan sensor cahaya berbasis Mikrokontroler, yang dapat menggerakkan motor untuk mengatur gerak plat agar sesuai dengan kuat cahaya yang diinginkan untuk penerangan suatu ruangan.

1.5 Sistematika Pembahasan

Sistematika penulisan yang digunakan dalam skripsi ini sebagai berikut:

BAB I Pendahuluan

Memuat latar belakang, rumusan masalah, tujuan, batasan masalah dan sistematika pembahasan.

BAB II Tinjauan Pustaka

Memuat pembahasan teori-teori yang mendukung dalam perancangan dan pembuatan alat.

BAB III Metodologi Penelitian

Berisi tentang metode penelitian dan perencanaan alat serta pengujian.

BAB IV Perancangan dan Pembuatan Alat

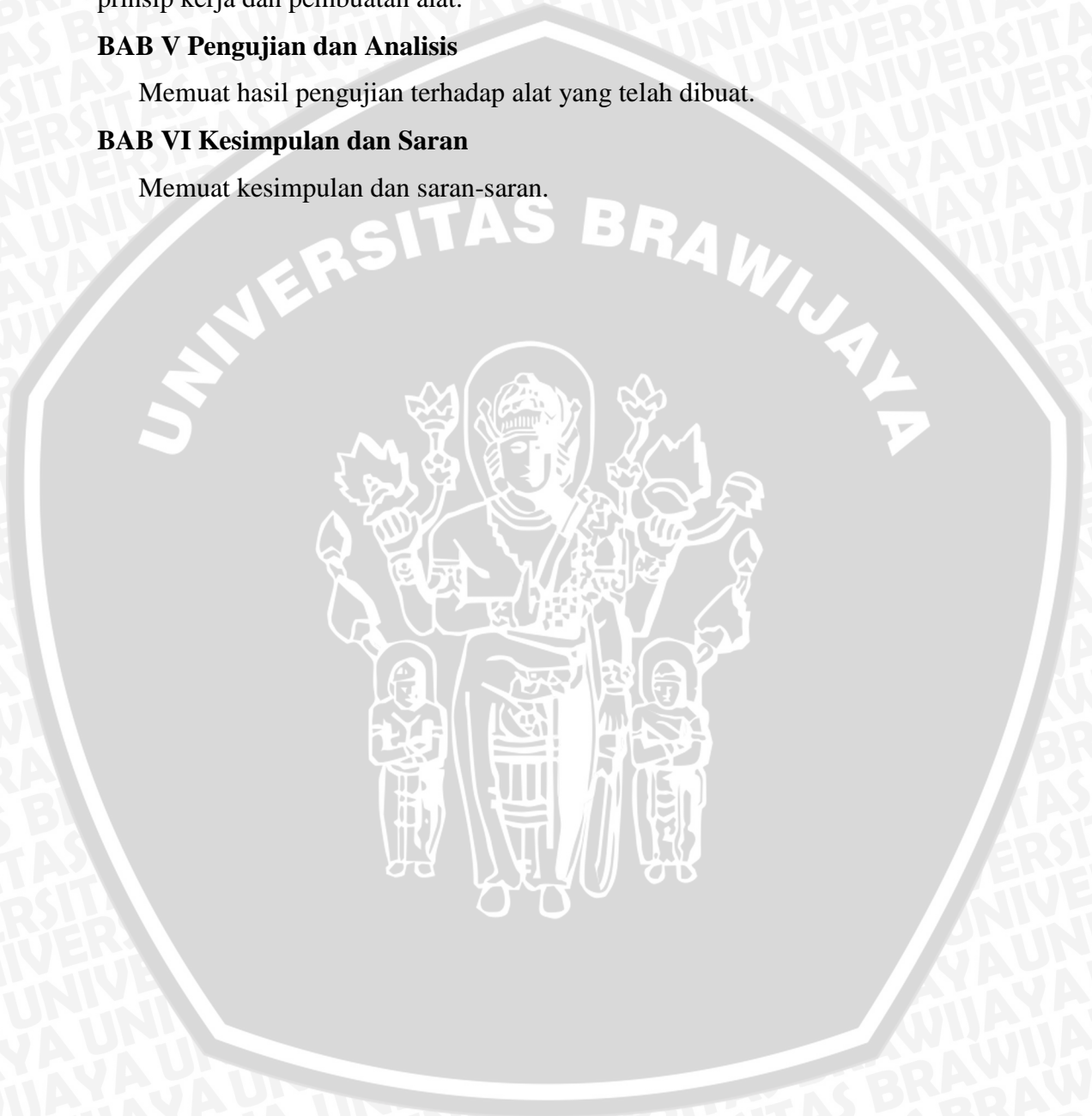
Perancangan alat yang merupakan spesifikasi, perencanaan blok diagram, prinsip kerja dan pembuatan alat.

BAB V Pengujian dan Analisis

Memuat hasil pengujian terhadap alat yang telah dibuat.

BAB VI Kesimpulan dan Saran

Memuat kesimpulan dan saran-saran.



BAB II TINJAUAN PUSTAKA

Dalam bab ini akan dijelaskan mengenai teori-teori yang akan menunjang perancangan pengendalian plat pada sistem *solatube* menggunakan kontroler PID berbasis Mikrokontroler.

2.1 *Solatube*

Solatube merupakan suatu alat yang berasal dari Amerika Serikat yang teknologinya telah ada selama hampir 20 tahun. Hanya di Indonesia baru berjalan kurang lebih sekitar 2 tahun. Konsep kerja dari *solatube* ini sangat sederhana, dari namanya pun dapat diketahui “sola” yang berarti matahari dan “tube” yang berarti tabung. Jadi sistem kerja dari *solatube* ialah memasukkan cahaya matahari ke dalam tabung reflektor, dipantulkan dan masuk ke dalam ruangan. Cahaya yang masuk ke dalam ruangan benar-benar dari cahaya matahari bukan cahaya listrik buatan. Berbeda dengan *solarcell* yang memanfaatkan energi matahari menjadi listrik dalam baterai untuk menyalakan lampu listrik. Sebaliknya *solatube* benar-benar hanya memanfaatkan cahaya matahari.

Keunggulan dari *solatube* ini ialah tidak adanya panas yang ikut terhantar. Dibandingkan dengan lampu listrik sejenis LED yang masih terdapat panas yang terhantar. Cahaya matahari yang mengandung panas adalah sinar infra merah, sinar ini dipantulkan di doom, sehingga yang masuk hanya spektrum cahayanya saja. Berbeda dengan lampu listrik yang harus diganti dalam rentang waktu beberapa minggu atau bulan. Keunggulan lain dari *solatube* ini adalah bebas perawatan, bebas biaya dan tanpa energi listrik.

Cahaya yang dikeluarkan oleh *solatube* berbeda dengan cahaya lampu yang konstan, karena *solatube* berkonsep reflektif, maka cahaya yang dihasilkan tergantung pada cahaya di luar. Namun, dalam keadaan mendung sekalipun, cahaya yang direfleksikan di dalam ruangan dapat mengakomodasi aktivitas dalam ruangan. Poin utama dari *solatube* ini adalah fakta bahwa cahaya matahari memberikan efek positif bagi kehidupan manusia dari segi psikologis. Ruangan yang diaplikasikan *solatube* terbukti memiliki aktivitas dan produktivitas yang tinggi. Dari segi ramah lingkungan, *solatube* merupakan langkah konkret hemat

energi karena pengoperasiannya tanpa menggunakan energi listrik dan memanfaatkan cahaya matahari yang berlimpah. Pada Gambar 2.1 merupakan bentuk daripada *solatube* pada umumnya.



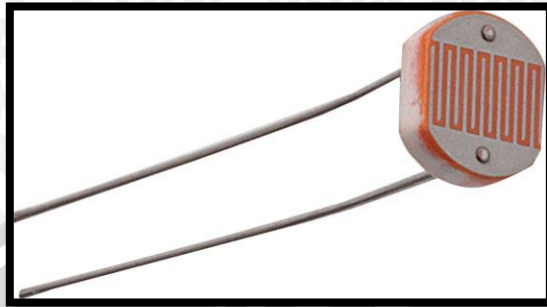
Gambar 2.1 Solatube
Sumber: Solatube.com

2.2 Sensor Cahaya (*Light Dependent Resistor*)

Sensor cahaya adalah salah satu alat yang digunakan dalam bidang elektronika, alat ini berfungsi untuk mengubah besaran cahaya menjadi besaran listrik. Alat ini memungkinkan seseorang untuk melakukan pendeteksian cahaya yang kemudian dapat dilakukan perubahan menjadi sinyal listrik dan dipakai dalam sebuah rangkaian yang memakai cahaya sebagai pemicunya. Cara kerja dari alat ini adalah mengubah energi dari foton menjadi elektron, umumnya satu foton dapat membangkitkan satu elektron. Alat ini mempunyai kegunaan yang sangat luas salah satu yang paling populer adalah pada kamera digital. Beberapa komponen yang biasanya digunakan dalam rangkaian sensor cahaya adalah LDR, *Photodiode*, dan *Photo Transistor*.

Sensor Cahaya LDR adalah salah satu jenis resistor yang dapat mengalami perubahan resistansinya apabila mengalami perubahan penerimaan cahaya. Besarnya nilai hambatan pada sensor cahaya LDR tergantung pada besar kecilnya cahaya yang diterima oleh LDR itu sendiri. LDR sering disebut dengan alat atau sensor yang berupa resistor yang peka terhadap cahaya. Biasanya LDR terbuat dari *cadmium sulfida* yaitu merupakan bahan semikonduktor yang resistansinya berubah-ubah menurut banyaknya cahaya (sinar) yang mengenainya. Resistansi LDR pada tempat yang gelap biasanya mencapai 10 M Ω , dan ditempat terang LDR mempunyai resistansi yang turun menjadi 150 Ω . Seperti halnya resistor

konvensional, pemasangan sensor LDR dalam suatu rangkaian sama persis seperti pemasangan resistor biasa. Bentuk dari sensor LDR ditunjukkan dalam Gambar 2.2.



Gambar 2.2 *Light Dependent Resistor*

Sumber: encepnurdinbogor.wordpress.com

2.3 Mikrokontroler

Mikrokontroler populer yang pertama dibuat oleh Intel pada tahun 1976, yaitu mikrokontroler 8-bit Intel 8748. Mikrokontroler tersebut adalah bagian dari keluarga mikrokontroler MCS-48. Sebelumnya, Texas instruments telah memasarkan mikrokontroler 4-bit pertama yaitu TMS 1000 pada tahun 1974. TMS 1000 yang mulai dibuat sejak 1971 adalah mikrokomputer dalam sebuah *chip*, lengkap dengan RAM dan ROM.

Pengendali mikro (*microcontroller*) adalah sistem mikroprosesor lengkap yang terkandung di dalam sebuah *chip*. Mikrokontroler berbeda dari mikroprosesor serba guna yang digunakan dalam sebuah PC, karena sebuah mikrokontroler umumnya telah berisi komponen pendukung sistem minimal mikroprosesor, yakni memori dan antarmuka I/O.

Berbeda dengan CPU serba-guna, mikrokontroler tidak selalu memerlukan memori eksternal, sehingga mikrokontroler dapat dibuat lebih murah dalam kemasan yang lebih kecil dengan jumlah *pin* yang lebih sedikit.

Sebuah *chip* mikrokontroler umumnya memiliki fitur:

- a) *central processing unit* - mulai dari prosesor 4-bit yang sederhana hingga prosesor kinerja tinggi 64-bit.
- b) *input/output* antarmuka jaringan seperti *port serial* (UART)
- c) antarmuka komunikasi serial lain seperti I²C, *Serial Peripheral Interface* and *Controller Area Network* untuk sambungan sistem

- d) periferan seperti *timer* dan *watchdog*
- e) RAM untuk penyimpanan data
- f) ROM, EPROM, EEPROM atau *Flash memory* untuk menyimpan program komputer
- g) pembangkit *clock* biasanya berupa resonator rangkaian RC
- h) pengubah analog ke digital

2.4 Mikrokontroler ATmega32

AVR ATmega32 merupakan sebuah mikrokontroler low power CMOS 8 bit berdasarkan arsitektur AVR RISC. Mikrokontroler ini memiliki karakteristik sebagai berikut:

1. Menggunakan arsitektur AVR RISC
 - a. 131 perintah dengan satu *clock cycle*
 - b. 32x8 register umum
2. Data dan program memori
 - a. 32 Kb *In-System Programmable Flash*
 - b. 2 Kb SRAM
 - c. 1 Kb *In-System EEPROM*
3. 8 *channel* 10-bit ADC
4. Two Wire Interface
5. USART *Serial Communication*
6. *Master/Slave SPI Serial Interface*
7. *On-Chip Oscillator*
8. *Watch-dog Timer*
9. 32 *Bi-directional I/O*
10. Tegangan operasi 2,7-5,5 V

Arsitektur AVR ini menggabungkan perintah secara efektif dengan 32 register umum. Semua register tersebut langsung terhubung dengan *Arithmetic Logic Unit* (ALU) yang memungkinkan 2 register terpisah diproses dengan satu perintah tunggal dalam satu *clock cycle*. Hal ini menghasilkan kode yang efektif dan kecepatan prosesnya 10 kali lebih cepat dari pada mikrokontroler CICS biasa. Pada Gambar 2.3 adalah konfigurasi pin Mikrokontroler AVR ATmega32

(XCK/T0) PB0	1	40	PA0 (ADC0)
(T1) PB1	2	39	PA1 (ADC1)
(INT2/AIN0) PB2	3	38	PA2 (ADC2)
(OC0/AIN1) PB3	4	37	PA3 (ADC3)
(SS) PB4	5	36	PA4 (ADC4)
(MOSI) PB5	6	35	PA5 (ADC5)
(MISO) PB6	7	34	PA6 (ADC6)
(SCK) PB7	8	33	PA7 (ADC7)
RESET	9	32	AREF
VCC	10	31	GND
GND	11	30	AVCC
XTAL2	12	29	PC7 (TOSC2)
XTAL1	13	28	PC6 (TOSC1)
(RXD) PD0	14	27	PC5 (TDI)
(TXD) PD1	15	26	PC4 (TDO)
(INT0) PD2	16	25	PC3 (TMS)
(INT1) PD3	17	24	PC2 (TCK)
(OC1B) PD4	18	23	PC1 (SDA)
(OC1A) PD5	19	22	PC0 (SCL)
(ICP1) PD6	20	21	PD7 (OC2)

Gambar 2.3 : Konfigurasi Pin ATmega 32 (Datasheet ATmega 32)

2.5 Motor Direct Current (DC)

Motor dc merupakan sebuah perangkat elektromagnetis yang mengubah energi listrik menjadi energi mekanik. Energi mekanik ini digunakan untuk, misalnya, memutar *impeller* pompa, *fan* atau *blower*, menggerakkan kompresor, mengangkat bahan, dll. Motor listrik digunakan juga di rumah (*mixer*, bor listrik, *fan* angin) dan di industry. Contoh bentuk dari motor dc ditunjukkan pada Gambar

2.4.



Gambar 2.4 Motor DC

Sumber: techcing.com

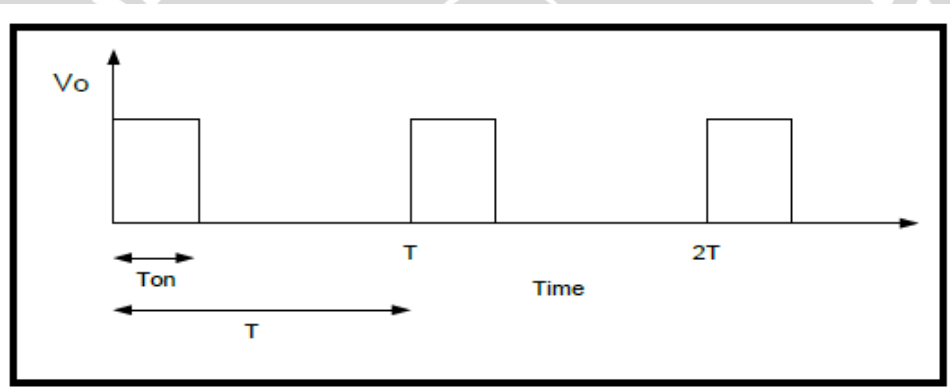
Prinsip kerja motor dc sesuai dengan hukum kemagnetan Lorenz, yaitu membangkitkan fluksi magnet pada suatu konduktor berarus dalam medan magnet sehingga timbul ggl induksi. Setiap arus yang mengalir melalui sebuah konduktor

akan menimbulkan medan magnet (Fathurohim, M. 2010). Metode *Pulse Width Modulation* (PWM) adalah salah satu metode yang dapat digunakan untuk mengatur kecepatan dari motor dc.

Kecepatan motor dc tergantung pada besarnya *duty cycle* yang diberikan pada motor DC tersebut. Pada sinyal PWM, frekuensi sinyal konstan sedangkan *duty cycle* bervariasi dari 0%-100%. Dengan mengatur *duty cycle* akan diperoleh keluaran yang diinginkan. (Alfarisi.R.,2011). *Duty cycle* adalah besarnya sinyal kontrol yang diberikan pada motor. Persamaan untuk perhitungan *duty cycle* ditunjukkan dalam persamaan 2.1 dengan T_{on} adalah periode logika tinggi, dan T adalah periode keseluruhan.

$$Duty\ cycle = \frac{T_{on}}{T} \times 100\% \quad (2.1)$$

Sinyal PWM secara umum ditunjukkan dalam Gambar 2.5



Gambar 2.5 Sinyal PWM (Ardyani, F. 2013)

Keterangan:

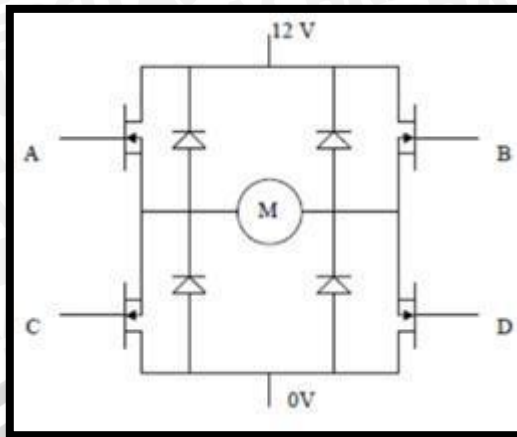
T_{on} = Periode logika tinggi

T = Periode keseluruhan

V_o = Amplitudo

2.6 Driver Motor Tipe H-Bridge MOSFET

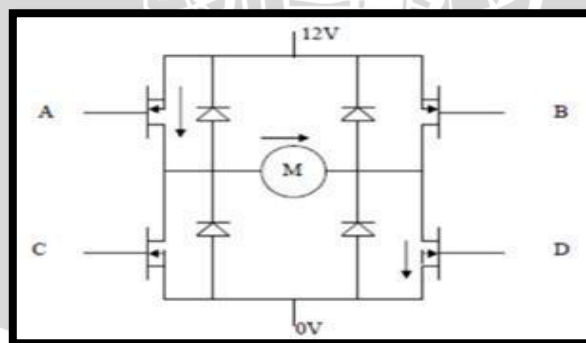
H-bridge adalah sebuah perangkat keras berupa rangkaian yang berfungsi untuk menggerakkan motor. Rangkaian ini diberi nama *H-bridge* karena bentuk rangkaiannya yang menyerupai huruf H seperti pada Gambar 2.6 berikut.



Gambar 2.6 H-Bridge MOSFET

Sumber: fahmizaleeits.wordpress.com

Rangkaian ini terdiri dari dua buah MOSFET kanal P dan dua buah MOSFET kanal N. Prinsip kerja rangkaian ini adalah dengan mengatur mati-hidupnya ke empat MOSFET tersebut. Huruf M pada gambar adalah motor dc yang akan dikendalikan. Bagian atas rangkaian akan dihubungkan dengan sumber daya kutub positif, sedangkan bagian bawah rangkaian akan dihubungkan dengan sumber daya kutub negatif. Pada saat MOSFET A dan MOSFET D *on* sedangkan MOSFET B dan MOSFET C *off*, maka sisi kiri dari gambar motor akan terhubung dengan kutub positif dari catu daya, sedangkan sisi sebelah kanan motor akan terhubung dengan kutub negatif dari catu daya sehingga motor akan bergerak searah jarum jam yang ditunjukkan pada Gambar 2.7.

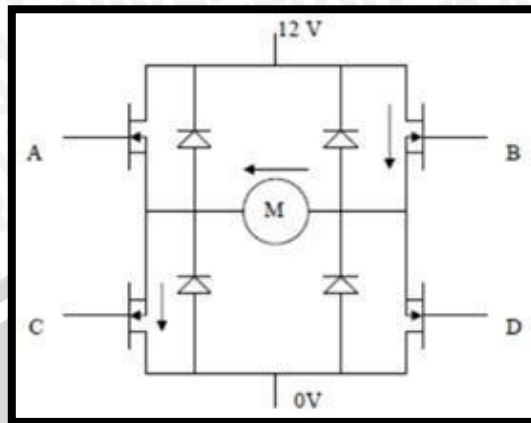


Gambar 2.7 Motor Bergerak Searah Jarum Jam

Sumber: fahmizaleeits.wordpress.com

Sebaliknya, jika MOSFET B dan MOSFET C *on* sedangkan MOSFET A dan MOSFET D *off*, maka sisi kanan motor akan terhubung dengan kutub positif dari catu daya sedangkan sisi kiri motor akan terhubung dengan kutub negatif dari

catu daya. Maka motor akan bergerak berlawanan arah jarum jam yang ditunjukkan pada Gambar 2.8.



Gambar 2.8 Motor Bergerak Berlawanan Arah Jarum Jam
Sumber: fahmizaleeits.wordpress.com

Konfigurasi lainnya adalah apabila MOSFET A dan MOSFET B sedangkan MOSFET C dan MOSFET D *off*. Konfigurasi ini akan menyebabkan sisi kiri dan kanan motor terhubung pada kutub yang sama yaitu kutub positif sehingga tidak ada perbedaan tegangan diantara dua buah polaritas motor, sehingga motor akan diam. Konfigurasi seperti ini disebut dengan konfigurasi *break*. Begitu pula jika MOSFET C dan MOSFET D saklar *on*, sedangkan MOSFET A dan MOSFET B *off*, kedua polaritas motor akan terhubung pada kutub negatif dari catu daya. Maka tidak ada perbedaan tegangan pada kedua polaritas motor, dan motor akan diam. Konfigurasi yang harus dihindari adalah pada saat MOSFET A dan MOSFET C *on* secara bersamaan atau MOSFET B dan MOSFET D *on* secara bersamaan. Pada konfigurasi ini akan terjadi hubungan arus singkat antara kutub positif catu daya dengan kutub negatif catu daya.

2.7 Kontroler

Sistem pengendalian dirancang untuk melakukan dan menyelesaikan tugas tertentu. Syarat utama sistem pengendalian adalah harus stabil. Di samping kestabilan mutlak, maka sistem harus memiliki kestabilan secara relatif, yakni tolak ukur kualitas kestabilan sistem dengan menganalisis sampai sejauh mana batas-batas kestabilan sistem tersebut jika dikenai gangguan (Ogata K.,1997). Selain itu analisis juga dilakukan untuk mengetahui bagaimana kecepatan sistem

dalam merespons *input*, dan bagaimana peredaman terhadap adanya lonjakan (*over shoot*).

Suatu sistem dikatakan stabil jika diberi gangguan maka sistem tersebut akan kembali ke keadaan *steady state* di mana *output* berada dalam keadaan tetap seperti tidak ada gangguan. Sistem dikatakan tidak stabil jika *output*-nya beresilasi terus menerus ketika dikenai suatu gangguan. Karena suatu sistem pengendalian biasanya melibatkan penyimpanan energi maka *output* sistem ketika diberi suatu *input*, tidak dapat mengikuti *input* secara serentak, tapi menunjukkan respon transien berupa suatu osilasi teredam sebelum mencapai *steady state*.

2.8 Kontroler Proporsional (P)

Kontroler proporsional adalah sebuah kontroler yang memiliki karakteristik mempercepat respon. Hubungan antara keluaran kontroler $m(t)$ dan sinyal kesalahan penggerak $e(t)$ ditunjukkan pada persamaan 2.2:

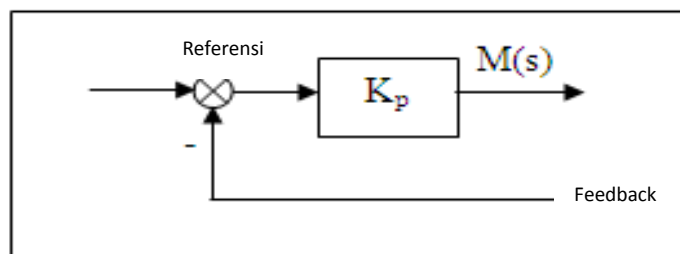
$$m(t) = K_p e(t) \quad (2.2)$$

atau, dalam besaran transformasi Laplace ditunjukkan pada persamaan 2.3

$$\frac{M(s)}{E(s)} = K_p \quad (2.3)$$

Di mana K_p adalah kepekaan proporsional atau penguatan.

Apapun wujud mekanisme yang sebenarnya dan apapun bentuk daya penggeraknya, kontroler proporsional pada dasarnya merupakan penguat dengan penguatan yang dapat diatur (Ogata K.,1997). Diagram blok kontroler proporsional ditunjukkan pada Gambar 2.9.



Gambar 2.9 Diagram Blok Kontroler Proporsional (OgataK., 1997)

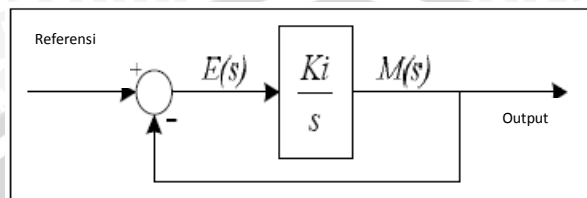
2.9 Kontroler Integral (I)

Kontroler integral memiliki kemampuan untuk mengurangi *offset* yang ditinggalkan oleh kontroler proporsional. Harga keluaran kontroler $m(t)$ diubah

dengan laju yang sebanding dengan sinyal kesalahan penggerak $e(t)$. Persamaan kontroler Ki ditunjukkan dalam persamaan 2.4.(Ogata K.,1997).

$$\frac{dm(t)}{dt} = Kie(t) \quad (2.4)$$

Ki adalah penguatan integral. Gambar 2.10 menunjukkan diagram blok kontroler integral.



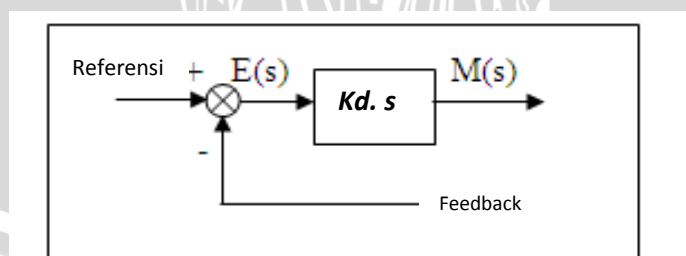
Gambar 2.10 Diagram Blok Kontroler Integral (OgataK., 1997)

2.10 Kontroler Differensial (D)

Kontroler ini digunakan untuk memperbaiki atau mempercepat respons transien sebuah sistem kontrol dengan cara memperbesar *phase lead* terhadap penguatan kontrol dan mengurangi *phase lag* penguatan tersebut (Ogata K.,1997). Kontroler differensial tidak dapat mengeluarkan *output* bila tidak ada perubahan *input*, selain itu kontroler differensial tidak dapat digunakan untuk proses yang mengandung *noise*. Hubungan antara keluaran kontroler $m(t)$ dan sinyal kesalahan penggerak $e(t)$ dinyatakan pada persamaan 2.5 :

$$\frac{M(s)}{E(s)} = Kd.s \quad (2.5)$$

Kd adalah penguatan differensial. Gambar 2.11 menunjukkan diagram blok kontroler differensial.



Gambar 2.11 Diagram Blok Kontroler Differensial (OgataK., 1997)

2.11 Kontroler Proporsional Integral Differensial (PID)

Gabungan aksi kontrol proporsional, integral, dan differensial mempunyai keunggulan dibandingkan dengan masing-masing dari tiga aksi kontrol tersebut. Masing – masing kontroler P, I, maupun D berfungsi untuk mempercepat reaksi

sistem, menghilangkan *offset*, dan mendapatkan energi ekstra ketika terjadi perubahan *load*.

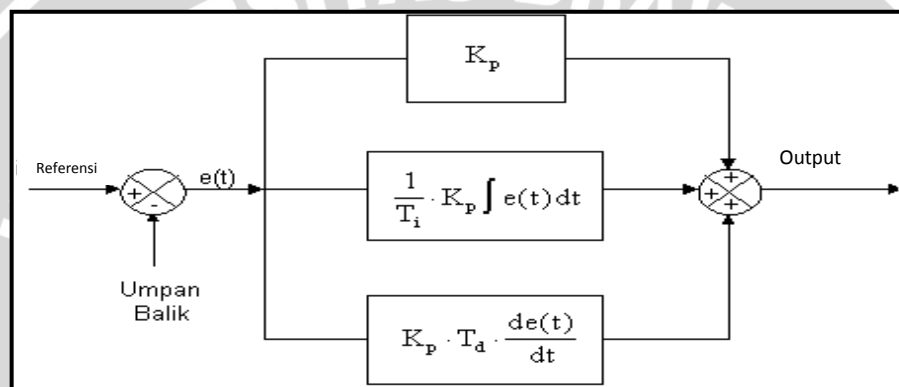
Persamaan kontroler PID ini dapat dinyatakan dalam persamaan 2.6 :

$$m(t) = K_p \cdot e(t) + \frac{K_p}{T_i} \cdot e(t)dt + K_p \cdot T_d \frac{de(t)}{dt} \quad (2.6)$$

Dalam transformasi Laplace dinyatakan dalam persamaan 2.7 :

$$\frac{M(s)}{E(s)} = K_p \left(1 + \frac{1}{T_i \cdot s} + T_d \cdot s \right) \quad (2.7)$$

T_i adalah waktu integral dan T_d adalah waktu derivatif. Gambar 2.12 menunjukkan diagram blok kontroler PID.



Gambar 2.12 Diagram Blok Kontroler PID (OgataK., 1997)

2.12 Metode Perancangan Kontroler Proporsional Integral Diferensial (PID) Menggunakan *Hand Tuning*.

Kontroler PID dapat di *tuning* dalam beberapa cara, antara lain *ziegler-nichols tuning*, *loop shaping*, metode analitis, optimasi, *pole placement*, *auto tuning* dan *hand tuning* (Smith,1997; Astrom & Hagglund,2004).

2.13 *Liquid Crystal Display (LCD)*

LCD yang digunakan berfungsi untuk menampilkan tegangan keluaran sensor LDR. LCD yang akan digunakan bertipe JHD608-202C Series. Spesifikasi dari LCD ini adalah sebagai berikut:

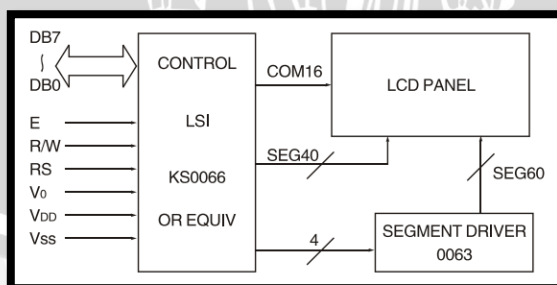
1. Menampilkan 20 karakter pada masing masing baris LCD sebanyak dua baris
2. Tegangan catu 5 volt dan temperatur operasi 0 – 50⁰ C

Masukan yang diperlukan untuk mengendalikan modul LCD ini berupa *bus data* yang masih termultipleks dengan *bus* alamat serta 3 bit sinyal kontrol, yaitu

RS, R/W dan E. Sementara pengendali *dot matrix* LCD dilakukan secara *internal* oleh kontroler yang sudah terpasang pada modul LCD. Berikut ini adalah diagram blok dari LCD JHD608-202C Series ditunjukkan dalam Gambar 2.13 serta Fungsi pin dari LCD JHD608-202C Series yang diperlihatkan pada Tabel 2.1.

Tabel 2.1 Fungsi Pin LCD JHD608-202C Series (Datasheet LCD JHD608-202C Series)

No Pin	Nama Pin	Fungsi
16	LEDA	Anoda (+)
15	LEDK	Katoda (-)
7 – 14	DB0– DB7	Merupakan saluran data, berisi perintah dan data yang akan ditampilkan.
6	E	Sinyal operasi awal, sinyal ini mengaktifkan data tulis atau baca
5	R/W	Sinyal seleksi tulis atau baca 0 = tulis 1 = baca
4	RS	Sinyal pemilih register 0 = <i>register</i> instruksi (tulis) 1 = <i>register</i> data (tulis dan baca)
3	Vlc	Untuk mengendalikan kecerahan LCD dengan mengubah Vlc
2	Vcc	Tegangan catu + 5 volt
1	Vss	Terminal <i>ground</i>



Gambar 2.13 Diagram Blok LCD JHD608-202C Series (Datasheet LCD JHD608-202C Series)

BAB III METODOLOGI PENELITIAN

Dalam pencapaian tujuan skripsi ini dibutuhkan metodologi penelitian dalam pelaksanaannya, berikut ini adalah metodologi yang digunakan dalam penelitian:

3.1 Studi Literatur

Studi literatur digunakan untuk mempelajari teori dan ilmu pengetahuan yang berhubungan dengan permasalahan yang akan diteliti. Sumber literatur diperoleh dari buku, jurnal, serta studi terhadap penelitian terdahulu yang terkait dengan topik utama dalam penelitian.

3.2 Perancangan Blok Diagram Sistem

Perancangan blok diagram sistem dibuat berdasarkan sejumlah komponen yang berperan dalam sistem kontrol, yang memuat fungsi operasi matematis dan digambarkan dalam bentuk blok - blok terhadap aliran proses. Yang memiliki tujuan untuk mempermudah peneliti dan pembaca dalam memahami proses serta langkah - langkah sistem pada suatu perancangan.

3.3 Perancangan Sistem

Perancangan Sistem dilakukan sebagai langkah awal sebelum terbentuknya suatu sistem beserta rangkaian elektronik serta pemrogramannya, hal ini dimaksudkan agar sistem pengendalian plat pada *solatube* dapat berjalan sesuai deskripsi awal yang telah direncanakan.

Perancangan sistem yang dilakukan meliputi :

1. Penentuan motor dc yang digunakan pada alat.
2. Penentuan rangkaian elektronik yang digunakan meliputi :
 - a. Rangkaian Mikrokontroler
 - b. Rangkaian *driver* motor.
 - c. Rangkaian catu daya.
 - d. Rangkaian sensor cahaya.
 - e. Rangkaian LCD.

3. Penentuan spesifikasi sistem, meliputi :
 - a. Penentuan deskripsi kerja sistem keseluruhan.
 - b. Kontroler PID serta rangkaian elektronik pendukung.
4. Perancangan program pada *software* CVAVR sebagai pusat pengendali dengan metode PID.

3.4 Realisasi Pembuatan Sistem

Untuk merealisasikan alat yang telah dirancang sebelumnya, tahapan - tahapan yang dilakukan adalah sebagai berikut :

1. Penyempurnaan konstruksi *solatube*.
2. Pembuatan rangkaian elektronik meliputi rangkaian pengondisi sinyal, rangkaian sensor cahaya, rangkaian catu daya, rangkaian *driver* motor, rangkaian LCD.
3. Pembuatan dan penyelesaian program dengan *software* CVAVR.

3.5 Pengujian dan Analisa Data

Setelah semua komponen pada alat sudah terhubung sesuai dengan diagram blok sistem yang telah dirancang dan program *software* sudah dibuat, maka diadakan pengujian dan analisa alat sebagai berikut :

1. Pengujian setiap blok rangkaian.
2. Kalibrasi pada sensor yang digunakan.
3. Penggabungan semua blok rangkaian menjadi sebuah sistem.
4. Pengujian alat secara keseluruhan.
5. Evaluasi dan analisa pengujian sistem yang didapat.

3.6 Pengambilan Keputusan

Kesimpulan diambil berdasarkan data yang didapat dari hasil pengujian sistem secara keseluruhan. Apabila hasil yang didapatkan sesuai dengan yang direncanakan sebelumnya, maka sistem kendali tersebut telah berhasil memenuhi harapan dan dapat dikembangkan untuk penelitian selanjutnya untuk disempurnakan.

BAB IV PERANCANGAN DAN PEMBUATAN ALAT

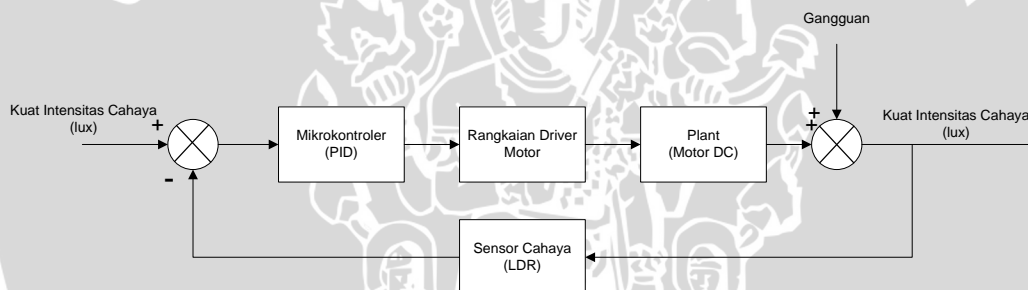
4.1 Perancangan Sistem

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

1. Perancangan perangkat keras terdiri atas perancangan *solatube* ber-plat, sistem minimum ATmega32, rangkaian catu daya, rangkaian *driver* motor dan sensor cahaya LDR.
2. Perancangan perangkat lunak dan perancangan program menggunakan *software* CVAVR.

4.2 Diagram Blok Sistem

Diagram blok sistem yang dirancang ditunjukkan dalam Gambar 4.1.



Gambar 4.1 Diagram Balok Sistem (Perancangan)

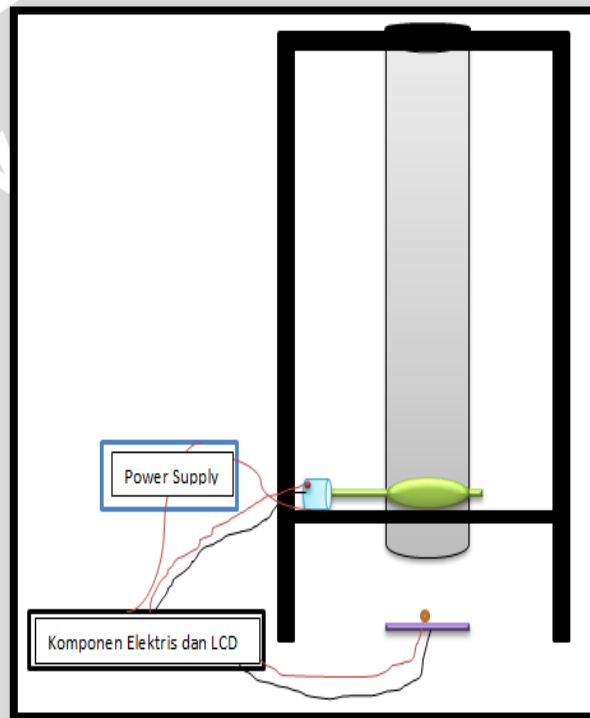
Keterangan dari blok diagram diatas adalah sebagai berikut :

1. *Setpoint* sistem berupa kuat cahaya (lux) sebesar 150 lux yang dikonversi dalam bentuk tegangan sebesar 3,68 Volt.
2. Pusat pengendalian sistem menggunakan ATmega32 yang memberikan sinyal keluaran berupa *pulse width modulation* (PWM) ke *driver* H-Bridge MOSFET.
3. Motor sebagai penggerak plat pada *solatube*.
4. Sebagai *feed back* digunakan sensor LDR.

4.3 Spesifikasi Alat

Spesifikasi sistem pengendalian motor menggunakan ATmega32 adalah sebagai berikut :

1. *Setpoint* yang sudah ditentukan yakni kuat cahaya sebesar 150 lux.
2. Ukuran *solatube*, tinggi 150cm dan diameter 20cm.
3. Sensor yang digunakan adalah sensor LDR dengan *output* sensor berupa tegangan analog 0-5 volt dc.
4. Aktuator menggunakan plat dengan motor dc yang mendapat sinyal masukan dari *output* PWM *driver* H-Bridge MOSFET.
5. Skema perancangan sistem dapat dilihat dalam Gambar 4.2.



Gambar 4.2 Skema Keseluruhan Sistem (Perancangan)

4.4 Prinsip Kerja Sistem

Cara kerja sistem adalah sebagai berikut :

1. Catu daya sebesar +/- 5 Volt digunakan sebagai catu rangkaian sistem minimum, dan *enable driver* motor. Catu daya 12 Volt digunakan sebagai catu motor dc pada plat.
2. Port komunikasi antara mikrokontroler dan PC menggunakan perantara kabel USBASP.

3. Menggunakan sensor LDR, sinyal-sinyal keluaran dari sensor yang masih berupa sinyal analog diproses oleh ADC agar dapat ditampilkan pada LCD.
4. Sinyal analog yang masuk ke mikrokontroler dikonversi ke dalam bentuk digital dengan resolusi 10 bit untuk kemudian diproses menjadi sinyal kontrol berupa PWM .
5. Sinyal kontrol dari mikrokontroler masuk ke *driver* H-Bridge MOSFET. *Driver* berfungsi menguatkan sinyal yang dihasilkan mikrokontroler dari 0-5 Volt menjadi 0-12 Volt.
6. Motor dc yang menggerakkan plat akan bergerak menutup ketika pembacaan sensor menunjukkan bahwa kuat cahaya yang terdeteksi melebihi dari *setpoint* yaitu 150 lux.
7. Motor dc yang menggerakkan plat akan bergerak membuka ketika pembacaan sensor menunjukkan bahwa kuat cahaya yang terdeteksi kurang dari *setpoint* yaitu 150 lux.
8. Motor dc yang menggerakkan plat akan berhenti bergerak saat kuat cahaya mencapai *setpoint* yang diinginkan.

4.5 Perancangan Perangkat Keras (*Hardware*)

Berdasarkan diagram blok perancangan alat yang telah disusun, perancangan perangkat keras meliputi perancangan rangkaian sensor cahaya LDR , minimum sistem ATmega32, rangkaian catu daya dan rangkaian *driver* motor. Di bawah ini adalah penjelasan masing-masing rangkaian penyusun keseluruhan alat.

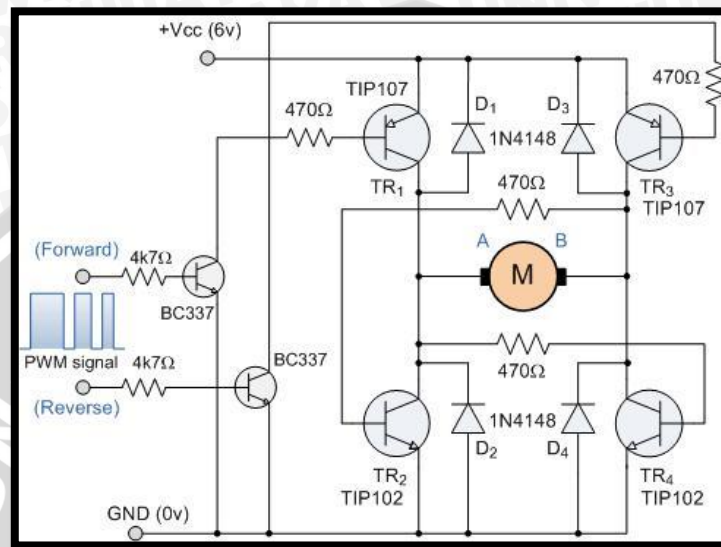
4.5.1 Sensor Cahaya *Light Dependent Resistor*

Sensor cahaya pada perancangan ini berfungsi sebagai sensor pengukur atau pendeteksi kuat cahaya. Keluaran dari sensor cahaya ini digunakan sebagai masukan untuk perhitungan algoritma PID yang dirancang. Sensor LDR mampu melakukan pembacaan pada rentang 0-5 volt.

4.5.2 *Driver* H-Bridge MOSFET

Modul pengendali motor dc digunakan untuk mengendalikan putaran motor dc yang menjadi penggerak plat. Rangkaian ini dihubungkan dengan

mikrokontroler. *Driver* pengendali pada perancangan ini menggunakan *driver* H-Bridge MOSFET yaitu sebuah perangkat keras berupa rangkaian yang berfungsi untuk menggerakkan motor DC. Rangkaian skematik *driver* motor H-Bridge MOSFET ditunjukkan pada Gambar 4.3.

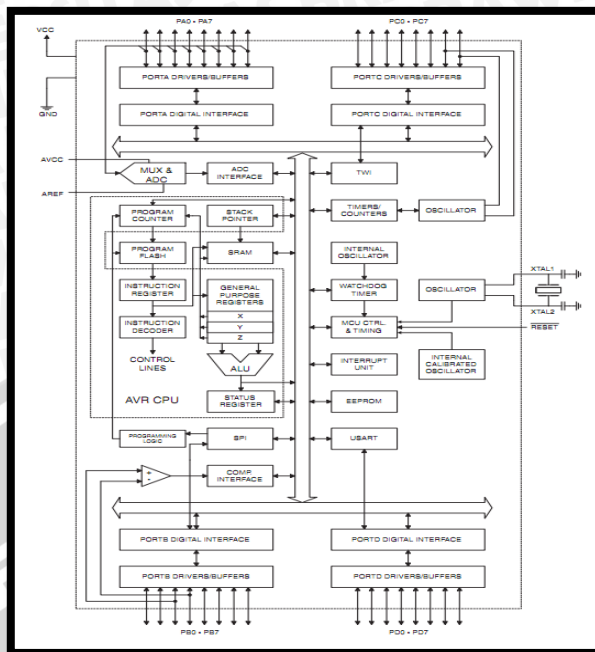


Gambar 4.3 Skematik Rangkaian Driver H-Bridge MOSFET

Sumber: fahmizaleeits.wordpress.com

4.5.3 Konfigurasi Mikrokontroler ATmega32

Pada alat ini digunakan ATmega32 sebagai pusat pengolah utama dalam melakukan proses pengendalian. Diagram blok ATmega32 dapat dilihat pada gambar 4.4.



Gambar 4.4 Diagram Blok ATmega32 (Datasheet ATmega 32)

Secara fungsional konfigurasi pin ATmega32 adalah sebagai berikut:

- Vcc (Tegangan sumber)
- GND (Ground)
- Port A (PA7-PA0)
- Port B (PB7-PB0)
- Port C (PC7-PC0)
- Port D (PD7-PD0)

Tabel 4.1 dan Tabel 4.2 merupakan fungsi pin dari Mikrokontroler ATmega32.

Tabel 4.1 Fungsi Pin ATmega32

PORT	FUNGSI
PA7	ADC7 (ADC input channel 7)
PA6	ADC6 (ADC input channel 6)
PA5	ADC5 (ADC input channel 5)
PA4	ADC4 (ADC input channel 4)
PA3	ADC3 (ADC input channel 3)
PA2	ADC2 (ADC input channel 2)
PA1	ADC1 (ADC input channel 1)
PA0	ADC0 (ADC input channel 0)



Tabel 4.2 Fungsi Pin ATmega32 (Lanjutan)

PORT	FUNGSI
PB7	SCK (<i>SPI Bus Serial Clock</i>)
PB6	MISO (<i>SPI Bus Master Input/Slave Output</i>)
PB5	MOSI (<i>SPI Bus Master Output/Slave Input</i>)
PB4	SS (<i>SPI Slave Select Input</i>)
PB3	AIN1 (<i>Analog Comparator Negative Input</i>), OCO (<i>Timer/Counter0</i>)
PB2	AIN0 (<i>Analog Comparator Positive Input</i>), INT2 (<i>External Interrupt</i>)
PB1	T1(<i>Timer/Counter1</i>)
PB0	T0(<i>Timer/Counter</i>), XCK (<i>USART External Clock I/O</i>)
PC7	TOSC2 (<i>Timer Oscillator Pin 2</i>)
PC6	TOSC1 (<i>Timer Oscillator Pin 1</i>)
PC5	TD1 (<i>JTAG Test Data In</i>)
PC4	TD0 (<i>JTAG Test Data Out</i>)
PC3	TMS (<i>JTAG Test Mode Select</i>)
PC2	TCK (<i>JTAG Test Clock</i>)
PC1	SDA (<i>Two-wire Serial Bus Data I/O Line</i>)
PC0	SCL (<i>Two-wire Serial Bus Clock Line</i>)
PD7	OC2 (<i>Timer/Counter2 Output</i>)
PD6	ICP1 (<i>Timer/Counter1 Input</i>)
PD5	OC1B (<i>Timer/Counter1 Output</i>)
PD4	TD0 (<i>JTAG Test Data Out</i>)
PD3	INT1 (<i>External Interrupt 1 Input</i>)
PD2	INT0 (<i>External Interrupt 0 Input</i>)
PD1	TXD (<i>USART Output Pin</i>)
PD0	RXD (<i>USART Input Pin</i>)

4.6 Perancangan Kontroler

4.6.1 Hand Tuning Kontroler PID

Kontroler PID dapat di tuning dalam beberapa cara, antara lain *ziegler-nichols tuning*, *loop shaping*, metode analitis, optimasi, *pole placement*, *auto tuning* dan *hand tuning* (Smith,1997; Astrom & Hagglund,2004). Pada penelitian kali ini digunakan cara *hand tuning* untuk menentukan besar Kp, Ki dan Kd. Ini dilakukan karena ada kendala untuk melakukan cara lain yang disebutkan di atas.

repository.ub.ac.id

Yakni tidak adanya model matematis, sehingga membuat cara analitis sulit untuk dilakukan. Prosedur untuk melakukan tuning adalah sebagai berikut:

1. Melepaskan kontroler integral dan diferensial dengan memberikan nilai $K_i = 0$ dan $K_d = 0$.
2. Mengatur nilai K_p hingga didapatkan respon yang diinginkan, dengan mengabaikan *offset*.
3. Dengan terus menaikkan nilai K_p , nilai dari K_d dinaikkan untuk mengurangi *overshoot* yang terjadi.
4. Naikkan nilai K_i untuk mengurangi *offset*.

Dengan menggunakan metode *hand tuning* nilai parameter PID perlu diubah-ubah secara *trial* dan *error* agar respon yang diperoleh sesuai dengan yang diinginkan.

4.6.2 Tuning Eksperimen

Dalam menentukan nilai PID dilakukan pengujian pada sistem dengan melihat respon motor sebelum dipasang pada plat *solatube*, pengujian dilakukan dengan langkah-langkah sebagai berikut:

1. Persiapkan minimum sistem yang telah tersambung dengan sensor LDR, motor dan catu daya.
2. Berkas cahaya yang digunakan berasal dari lampu.
3. Menentukan *set point* sistem yakni sebesar 3,68 volt (150 lux).
4. Lakukan pengujian dengan cara menyinari sensor LDR dengan cahaya lampu dari sebelum diberi kontroler sampai dengan telah diberi kontroler dan lihat respon motor yang terjadi kemudian catat.
5. Hasil dari proses pengujian dapat dilihat pada Tabel 4.3.

Tabel 4.3 Hasil Pengujian Respon Motor Tanpa Kontroler

Pengujian Ke-	Kp	Ki	Kd	Tegangan Sensor (Vout)	Respon Motor
1	0	0	0	0,00 - 3,68	Motor Bergerak Ke Kiri
				3,69 - 3,78	Motor Bergerak Ke Kanan
				3,79	Motor Berhenti
				3,80-4,98	Motor Bergerak Ke Kanan

Pada Tabel 4.3 dapat dilihat bahwa sebelum diberi nilai PID, motor berhenti saat sensor LDR mendeteksi kuat cahaya sebesar 3,79 volt. Hal tersebut menunjukkan bahwa masih terdapat kesalahan pada sistem. Oleh karena itu dibutuhkan suatu pengontrolan agar motor dapat berhenti pada *setpoint* yang diinginkan.

Tabel 4.4 Hasil Pengujian Respon Motor Dengan Menambahkan Kontroler Kp

Pengujian Ke-	Kp	Ki	Kd	Tegangan Sensor (Vout)	Respon Motor
1	1	0	0	0,00 - 4,98	Motor Berhenti
2	10	0	0	0,00 - 3,39	Motor Bergerak Ke Kiri
				3,40 - 3,77	Motor Berhenti
				3,78 - 4,98	Motor Bergerak Ke Kiri
3	50	0	0	0,00 - 3,62	Motor Bergerak Ke Kiri
				3,63 - 3,69	Motor Berhenti
				3,70 - 4,98	Motor Bergerak Ke Kanan
4	80	0	0	0,00 - 3,65	Motor Bergerak Ke Kiri
				3,66 - 3,69	Motor Berhenti
				3,70 - 4,98	Motor Bergerak Ke Kanan
5	100	0	0	0,00 - 3,66	Motor Bergerak Ke Kiri
				3,67 - 3,69	Motor Berhenti
				3,70 - 4,98	Motor Bergerak Ke Kanan

Pada Tabel 4.4 dapat dilihat bahwa dari beberapa pengujian dengan menambahkan nilai Kp, respon motor yang paling baik dan mendekati *setpoint* ialah dengan pemberian nilai Kp sebesar 100. Namun motor masih belum berhenti pada *setpoint* yang diinginkan, oleh karena itu ditambahkan nilai parameter Ki agar respon motor dapat berhenti pada *setpoint* yang diinginkan. Hasil pengujian dengan menambahkan nilai Ki dapat dilihat pada Tabel 4.5

Tabel 4.5 Hasil Pengujian Respon Motor Dengan Menambahkan Kontroler Kp dan Ki

Pengujian Ke-	Kp	Ki	Kd	Tegangan Sensor (Vout)	Respon Motor
1	100	0,5	0	0,00 - 3,37	Motor Bergerak Ke Kiri
				3,68 - 3,69	Motor Berhenti
				3,70 - 4,98	Motor Bergerak Ke Kiri

Hasil pengujian dengan menambahkan nilai Ki dapat dilihat bahwa, motor dapat berhenti pada *setpoint* yang diinginkan. Namun masih terdapat kesalahan, di mana motor juga berhenti melebihi dari *setpoint*, sehingga perlu untuk ditambahkan nilai Kd agar motor tepat berhenti pada *setpoint* yang diinginkan.

Tabel 4.6 Hasil Pengujian Respon Motor Dengan Menambahkan Kontroler Kp, Ki dan Kd

Pengujian Ke-	Kp	Ki	Kd	Tegangan Sensor (Vout)	Respon Motor
1	100	0,5	0,2	0,00 - 3,37	Motor Bergerak Ke Kiri
				3,68	Motor Berhenti
				3,69 - 4,98	Motor Bergerak Ke Kiri

Pada Tabel 4.6 dapat dilihat bahwa setelah ditambahkan nilai Kd, motor dapat berhenti pada *setpoint* yang diinginkan. Oleh karena itu nilai-nilai parameter yang digunakan pada sistem yaitu Kp = 100, Ki = 0,5 dan Kd = 0,2.

4.6.3 Penerapan Kontroler PID Pada Program

Pada Bab II telah dijelaskan bahwa persamaan dari kontrol PID ialah:

$$m(t) = Kp e(t) + Ki \int_0^t e(t)dt + Kd \frac{de(t)}{dt} \quad (4.1)$$

Atau

$$m(t) = Kp e(t) + \frac{Kp}{Ti} e(t)dt + KpTd \frac{de(t)}{dt} \quad (4.2)$$

Di mana m(t) merupakan sinyal *output* pengendali PID dan e(t) adalah sinyal *error*. Sinyal *error* yang dimaksud ialah nilai referensi (*setpoint*) dikurangi nilai sensor yang terbaca. Untuk persamaan PID no. 4.1 merupakan PID bentuk *independent* dan persamaan no. 4.2 merupakan PID bentuk *dependent*. Pada persamaan no 4.1 dan 4.2 merupakan persamaan dalam kawasan waktu *continuous* (analog). Sedangkan agar persamaan tersebut dapat direalisasikan

dalam bentuk pemrograman, maka persamaan tersebut harus didiskretisasi terlebih dahulu (*digitalization*). Untuk kontroler PID diskrit dapat dilihat pada persamaan 4.3 dan 4.4.

$$m(k) = Kp e_k + Ki T \sum_{i=0}^k e_k + \frac{1}{T} Kd(e_k - e_{k-1}) \quad (4.3)$$

$$m(k) = Kp e_k + Ki T [e_0 + e_1 + \dots + e_{k-1} + e_k] + \frac{1}{T} Kd(e_k - e_{k-1}) \quad (4.4)$$

Setelah didapatkan bentuk diskrit dari PID maka dapat kita terapkan pada bahasa pemrograman dengan rumus seperti pada persamaan 4.5.

$$Output = Kp.error + Ki (error + last_error)T_s + \frac{1}{T_s} Kd (error - last_error) \quad (4.5)$$

T_s merupakan waktu *sampling* yakni lamanya waktu yang digunakan untuk mencuplik nilai dari sensor. Nilai dari sensor ini berguna untuk mendapatkan sinyal *error* ($error = setpoint - \text{nilai sensor}$). Di mana waktu *sampling* ini sangat berpengaruh pada kesensitifan sistem yang akan dikontrol.

Sehingga persamaan PID yang diterapkan pada sistem ini ialah:

Error = tegangan referensi – tegangan sensor LDR; //Tegangan referensi 3,68 volt

P = 100*error; //Nilai tuning Kp sebesar 100

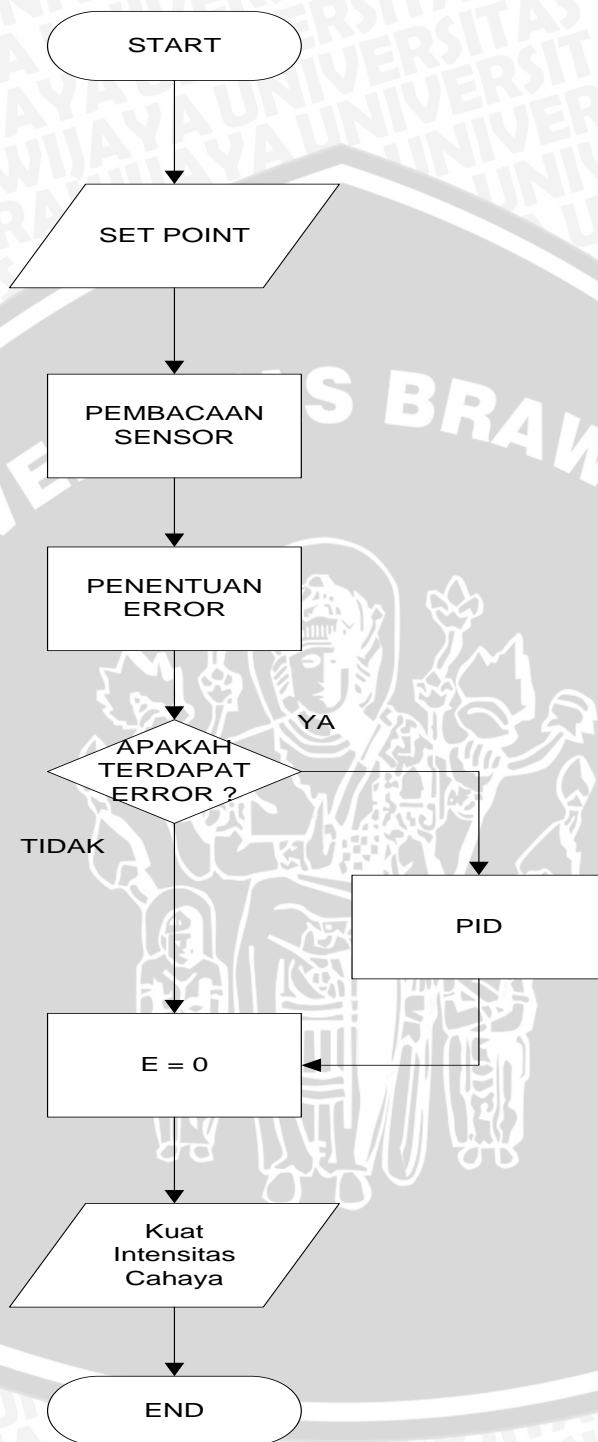
I = (0.5*(error + last_err))*0.01; // Nilai tuning Ki sebesar 0,5 dan Ts 0,01 detik

D = (0.2*(error - last_err))/0.01; // Nilai tuning Kd sebesar 0,2 dan Ts 0,01 detik

$$Output = 100.error + 0,5 (error + last_error)0,01 + \frac{1}{0,01} 0,2(error - last_error)$$

4.7 Perancangan Perangkat Lunak

Flowchart keseluruhan sistem ditunjukkan pada Gambar 4.5.



Gambar 4.5 Flowchart Keseluruhan Sistem (Perancangan)

BAB V PENGUJIAN DAN ANALISIS

Tujuan pengujian sistem ini adalah untuk menentukan apakah alat yang telah dibuat berfungsi dengan baik dan sesuai dengan perancangan. Pengujian pada sistem ini meliputi pengujian setiap blok maupun pengujian secara keseluruhan. Pengujian setiap blok ini dilakukan untuk menemukan letak kesalahan dan mempermudah analisis pada sistem apabila alat tidak bekerja sesuai dengan perancangan. Pengujian dibagi menjadi beberapa bagian, yaitu :

1. Pengujian sensor cahaya LDR
2. Pengujian *driver* motor dc
3. Pengujian kecepatan motor dc
4. Pengujian LCD
5. Pengujian tanpa kontroler
6. Pengujian keseluruhan sistem

5.1. Pengujian Sensor *Light Dependent Resistor*

Dalam pengujian Sensor LDR terdapat beberapa aspek yang harus diperhatikan, antara lain:

a. Tujuan

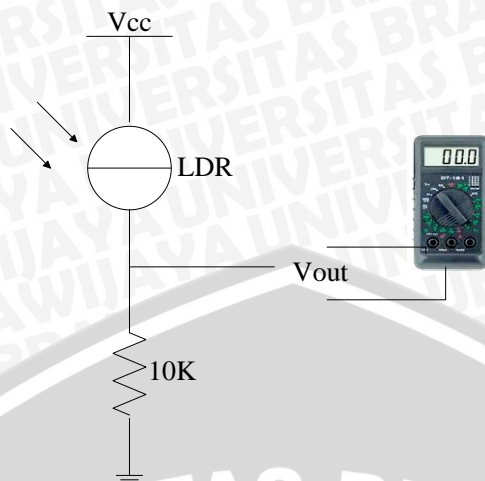
Mengetahui kemampuan atau kinerja dari sensor LDR dan grafik dari perbandingan antara tegangan keluaran dengan resistansi dari sensor LDR.

b. Peralatan yang digunakan

1. Sensor LDR.
2. Resistor 10K.
3. Multimeter.
4. Luxmeter Krisbow KW06-288.
5. Catu daya 5 volt.

c. Langkah pengujian

1. Merangkai peralatan seperti pada Gambar 5.1
2. Luxmeter sebagai alat ukur untuk mendeteksi besar cahaya dari 0 – 100 lux
3. Mencatat data tegangan keluaran tiap 5 lux dari V_{out} .



Gambar 5.1 Pengujian *Light Dependent Resistor*

d. Hasil Pengujian

Tegangan keluaran dari rangkaian sensor ditulis dalam persamaan

$$V_{output} = \frac{10K}{10K + R_{ldr}} \times V_{input} \dots\dots\dots(5.1)$$

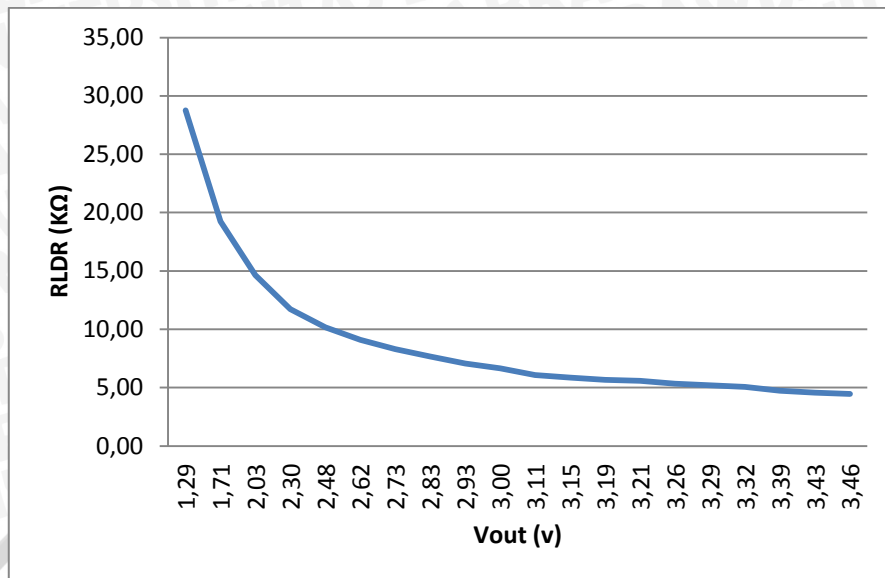
data hasil pengujian sensor LDR ditunjukkan dalam Tabel 5.1 berikut

Tabel 5.1 Data Hasil Pengujian LDR

No	Luxmeter (lux)	Vout (v)	Rldr (KΩ)
1	0	0,00	~
2	5	1,29	28,76
3	10	1,71	19,24
4	15	2,03	14,63
5	20	2,30	11,74
6	25	2,48	10,16
7	30	2,62	9,08
8	35	2,73	8,32
9	40	2,83	7,67
10	45	2,93	7,06
11	50	3,00	6,67
12	55	3,11	6,08
13	60	3,15	5,87
14	65	3,19	5,67
15	70	3,21	5,58
16	75	3,26	5,34
17	80	3,29	5,20
18	85	3,32	5,06
19	90	3,39	4,75
20	95	3,43	4,58
21	100	3,46	4,45

Data dalam Tabel 5.1 merupakan hasil pengujian pada sensor LDR, apabila dibuat dalam bentuk grafik maka hasilnya dapat dilihat pada Gambar 5.2.





Gambar 5.2 Grafik Hubungan Antara Tegangan Keluaran dengan Resistansi LDR.

Pada Gambar 5.2 dapat dilihat bahwa hubungan antara tegangan keluaran sensor dengan resistansinya berbanding terbalik. Oleh karena itu sensor LDR yang digunakan dalam kondisi baik.

5.2. Pengujian Driver Motor DC

Dalam pengujian driver motor dc terdapat beberapa aspek yang harus diperhatikan, antara lain:

a. Tujuan

Mengetahui *output driver* motor apabila diberi *input* yang berbeda-beda.

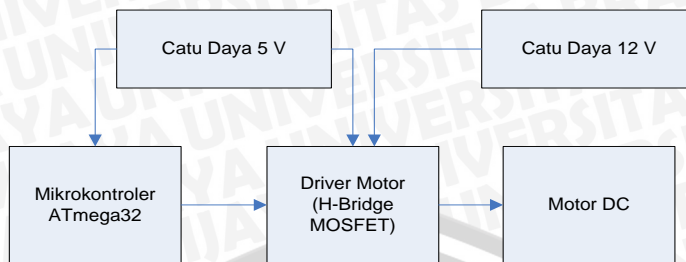
b. Peralatan yang digunakan

1. ATmega32
2. Driver motor
3. Motor dc
4. Catu daya 5V dan 12V

c. Langkah pengujian

1. Merangkai peralatan seperti dalam Gambar 5.3
2. Mengisi mikrokontroler dengan program yang mengeluarkan data keluaran biner.
3. Mengaktifkan catu daya 5 volt dan 12 volt

4. Mencatat pergerakan motor dc



Gambar 5.3 Diagram Blok Pengujian Driver Motor DC

d. Hasil pengujian

Tabel 5.2 Hasil Pengujian Driver Motor DC

No	A	B	C	D	Arah Putaran Motor DC
1	1	0	0	1	Motor Berputar Searah Jarum Jam
2	0	1	1	0	Motor Berputar Berlawanan Arah Jarum Jam
3	0	0	0	0	Diam
4	0	0	1	1	Pengereman/Diam
5	1	1	0	0	Pengereman/Diam

Berdasarkan hasil pengujian pada Tabel 5.2 dapat diketahui bahwa motor dc akan berputar searah jarum jam ketika MOSFET A dan MOSFET D *on*, dan apabila MOSFET B dan MOSFET C *on* maka motor akan berputar berlawanan arah jarum jam. Dengan demikian dapat disimpulkan bahwa rangkaian *driver* motor dc ini dapat berjalan dengan baik pada sistem yang direncanakan.

5.3. Pengujian Kecepatan Motor DC

Dalam pengujian kecepatan motor dc terdapat beberapa aspek yang harus diperhatikan, antara lain:

a. Tujuan

Untuk mengetahui apakah motor dc dapat berjalan sesuai dengan yang diinginkan.

b. Peralatan yang digunakan

1. ATmega32
2. Optocoupler
3. Tachometer



c. Langkah pengujian

Mengukur kecepatan motor dengan nilai *Pulse Width Modulation* (PWM) yang berbeda dengan menggunakan tachometer, sehingga didapatkan kecepatan yang diinginkan.

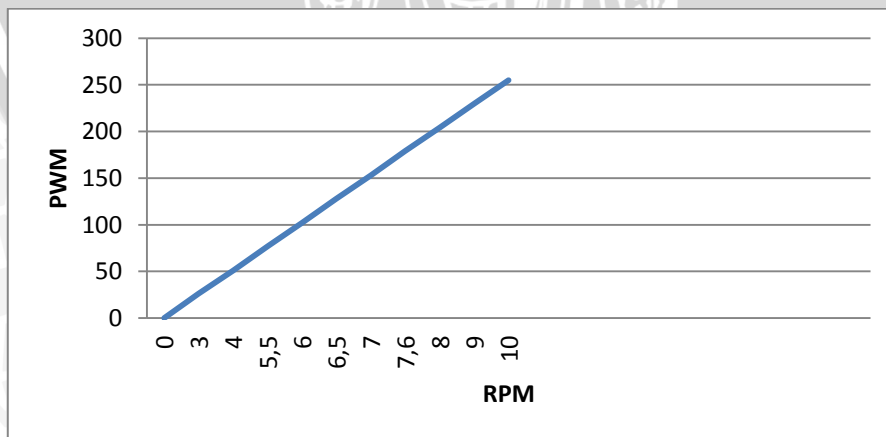
d. Hasil pengujian

Hasil pengujian kecepatan motor dengan nilai *Pulse Width Modulation* (PWM) yang ditentukan ditunjukkan pada Tabel 5.3

Tabel 5.3 Hasil Pengujian *Pulse Width Modulation* (PWM)

No	Dalam Persen (%)	PWM	Vout Driver (v)	RPM (Tachometer)
1	0	0	0	0
2	10	26	6,9	3
3	20	51	7,8	4
4	30	77	8	5,5
5	40	102	8,5	6
6	50	128	9	6,5
7	60	153	9,4	7
8	70	179	9,8	7,6
9	80	204	10	8
10	90	230	11	9
11	100	255	12,5	10

Data dalam Tabel 5.3 merupakan hasil pengujian pada motor dc, apabila dibuat dalam bentuk grafik maka hasilnya dapat dilihat pada Gambar 5.4.



Gambar 5.4 Grafik Hubungan PWM dengan RPM Motor DC

Pada Gambar 5.4 dapat dilihat bahwa hubungan antara PWM dengan RPM motor sebanding. Sehingga dapat diartikan bahwa kondisi motor dc yang digunakan dalam penelitian dapat bekerja dengan baik.

5.4. Pengujian LCD

Dalam pengujian LCD terdapat beberapa aspek yang harus diperhatikan, antara lain:

a. Tujuan

Untuk mengetahui bahwa masing-masing pin pada *port C* mikrokontroler dapat mengirim data menuju LCD sesuai dengan program yang dibuat dan LCD dapat menampilkan data karakter dengan benar.

b. Peralatan yang digunakan

1. ATmega32
2. LCD

c. Langkah Pengujian

1. Membuat program dengan menggunakan *software* CVAVR untuk menampilkan data karakter nama.
2. Menghubungkan pin LCD dengan Mikrokontroler

Tabel 5.4 Hubungan Pin LCD Dengan Mikrokontroler

LCD Pin	Mikrokontroler
GND	GND
VCC	+5 V
VLC	VR 10K
RS	Port C7
R/W	Port C1
E	Port C5
DB4	Port C4
DB5	Port C2
DB6	Port C6
DB7	Port C0

3. Meng-*upload* program ke dalam mikrokontroler.

d. Hasil Pengujian

Hasil pengujian LCD dapat dilihat pada Gambar 5.5



Gambar 5.5 Tampilan Data Karakter Pada LCD

5.5. Pengujian Sistem Keseluruhan Tanpa Kontroler

Dalam pengujian sistem tanpa kontroler ini terdapat beberapa aspek yang harus diperhatikan, antara lain:

a. Tujuan

Pengujian ini bertujuan untuk mengetahui bagaimana respon sistem terhadap *setpoint* ketika tanpa menggunakan kontroler, sehingga dapat ditentukan bahwa sistem memerlukan sebuah sistem pengontrolan.

b. Peralatan yang digunakan

1. *Solatube* ber-plat
2. Mikrokontroler ATmega32
3. *Driver* Motor
4. Motor DC
5. Optocoupler
6. Komputer
7. Program dan software CVAVR.

c. Langkah Pengujian

1. Merangkai peralatan seperti Gambar 5.6.
2. Menentukan *setpoint* sistem sebesar 3,68 v (150 lux)
3. Pengambilan data dilakukan secara *sampling* tiap 500 ms terhadap tegangan yang dihasilkan sensor.
4. Pengambilan data awal dilakukan saat plat *solatube* menutup atau saat tegangan sensor bernilai 0 v.
5. Data yang di *sampling* ditampilkan pada LCD dan dicatat.



Gambar 5.6 Pengujian Rangkaian Sistem

d. Hasil Pengujian

Setelah melakukan prosedur pengujian, didapatkan data hasil pengujian sistem keseluruhan tanpa kontroler. Hasil dapat dilihat pada Tabel 5.5 dan 5.6.

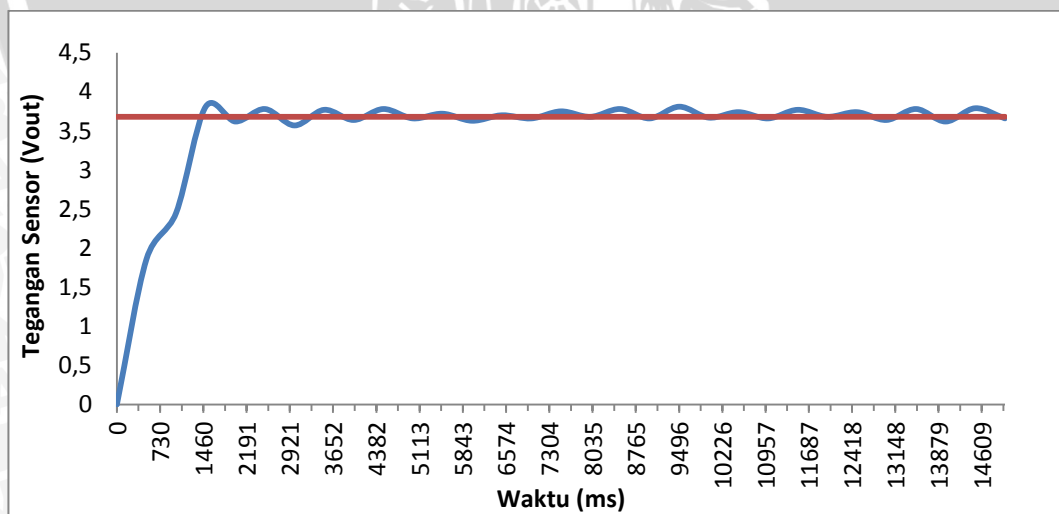
Tabel 5.5 Data Hasil Pengujian Tanpa Kontroler

No.	Set Point (v)	Tegangan Sensor (Vout)	Waktu (ms)	Respon Motor
1	3,68	0	0	Motor Bergerak Ke Kiri
2	3,68	1,86	500	Motor Bergerak Ke Kiri
3	3,68	2,45	1000	Motor Bergerak Ke Kiri
4	3,68	3,81	1500	Motor Bergerak Ke Kanan
5	3,68	3,62	2000	Motor Bergerak Ke Kiri
6	3,68	3,78	2500	Motor Bergerak Ke Kanan
7	3,68	3,57	3000	Motor Bergerak Ke Kiri
8	3,68	3,77	3500	Motor Bergerak Ke Kanan
9	3,68	3,64	4000	Motor Bergerak Ke Kiri
10	3,68	3,78	4500	Motor Bergerak Ke Kanan
11	3,68	3,66	5000	Motor Bergerak Ke Kiri
12	3,68	3,72	5500	Motor Bergerak Ke Kanan
13	3,68	3,63	6000	Motor Bergerak Ke Kiri
14	3,68	3,7	6500	Motor Bergerak Ke Kanan
15	3,68	3,66	7000	Motor Bergerak Ke Kiri
16	3,68	3,75	7500	Motor Bergerak Ke Kanan
17	3,68	3,68	8000	Motor Bergerak Ke Kiri

Tabel 5.6 Data Hasil Pengujian Tanpa Kontroler (lanjutan)

No.	Set Point (v)	Tegangan Sensor (Vout)	Waktu (ms)	Respon Motor (plat)
18	3,68	3,78	8500	Motor Bergerak Ke Kanan
19	3,68	3,66	9000	Motor Bergerak Ke Kiri
20	3,68	3,81	9500	Motor Bergerak Ke Kanan
21	3,68	3,67	10000	Motor Bergerak Ke Kiri
22	3,68	3,74	10500	Motor Bergerak Ke Kanan
23	3,68	3,66	11000	Motor Bergerak Ke Kiri
24	3,68	3,77	11500	Motor Bergerak Ke Kanan
25	3,68	3,68	12000	Motor Bergerak Ke Kiri
26	3,68	3,74	12500	Motor Bergerak Ke Kanan
27	3,68	3,64	13000	Motor Bergerak Ke Kiri
28	3,68	3,78	13500	Motor Bergerak Ke Kanan
29	3,68	3,62	14000	Motor Bergerak Ke Kiri
30	3,68	3,79	14500	Motor Bergerak Ke Kanan
31	3,68	3,66	15000	Motor Bergerak Ke Kiri

Pada Tabel 5.5 dan Tabel 5.6 dapat dilihat bahwa hasil *sampling* menunjukkan jika plat *solatube* bergerak ke kanan dan ke kiri secara terus-menerus tanpa bisa berhenti meskipun sudah mencapai *setpoint*. Sehingga diperlukan suatu pengontrolan agar sistem dapat bekerja sesuai dengan yang diinginkan. Grafik hasil pengujian ditunjukkan pada Gambar 5.7 di bawah.



Gambar 5.7 Grafik Hasil Pengujian Tanpa Kontroler

Pada Gambar 5.7 dapat dilihat jika sistem sudah berada didaerah *setpoint* meskipun masih terdapat *error* dengan %Ess sebesar 0,68 %.

$$\begin{aligned}\%Ess &= \frac{|Average\ speed\ steady-setpoint|}{setpoint} \times 100\% \\ &= \frac{|3,705-3,68|}{3,68} \times 100\% \\ &= 0,68\%\end{aligned}$$

5.6. Pengujian Sistem Secara Keseluruhan Dengan Kontroler

Dalam pengujian sistem dengan kontroler ini terdapat beberapa aspek yang harus diperhatikan, antara lain:

a. Tujuan

Pengujian dilakukan untuk mengetahui bagaimana respon keseluruhan sistem *solatube* dapat bekerja dengan baik sesuai dengan *setpoint* yang diharapkan dengan parameter PID yang dicari menggunakan *hand tuning*, dimana nilai $K_p = 100$, $K_i = 0,5$ dan $K_d = 0,2$.

b. Peralatan Pengujian

1. *Solatube* ber-plat
2. Mikrokontroler ATmega32
3. Driver Motor
4. Motor DC
5. Optocoupler
6. Komputer
7. Program dan software CVAVR.

c. Langkah Pengujian

1. Merangkai peralatan seperti Gambar 5.6.
2. Menentukan *setpoint* sistem sebesar 3,68 v (150 lux)
3. Pengambilan data dilakukan secara *sampling* tiap 500 ms terhadap tegangan yang dihasilkan sensor.
4. Pengambilan data awal dilakukan saat plat *solatube* menutup atau saat tegangan sensor bernilai 0 v.
5. Data yang di *sampling* ditampilkan pada LCD dan dicatat.

d. Hasil Pengujian

Setelah melakukan prosedur pengujian, didapatkan data hasil pengujian.

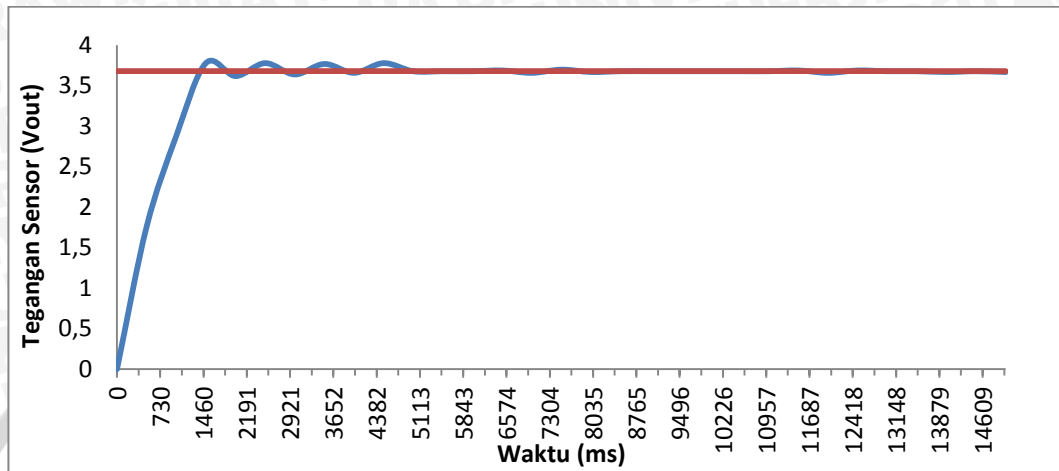
Hasil dapat dilihat pada Tabel 5.7.

Tabel 5.7 Data Hasil Pengujian Dengan Kontroler

No.	Set Point (v)	Tegangan Sensor (Vout)	Waktu (ms)	Respon Motor
1	3,68	0	0	Motor Bergerak Ke Kiri
2	3,68	1,76	500	Motor Bergerak Ke Kiri
3	3,68	2,88	1000	Motor Bergerak Ke Kiri
4	3,68	3,78	1500	Motor Bergerak Ke Kanan
5	3,68	3,62	2000	Motor Bergerak Ke Kiri
6	3,68	3,78	2500	Motor Bergerak Ke Kanan
7	3,68	3,64	3000	Motor Bergerak Ke Kiri
8	3,68	3,77	3500	Motor Bergerak Ke Kanan
9	3,68	3,66	4000	Motor Bergerak Ke Kiri
10	3,68	3,78	4500	Motor Bergerak Ke Kanan
11	3,68	3,68	5000	Motor Berhenti
12	3,68	3,68	5500	Motor Berhenti
13	3,68	3,68	6000	Motor Berhenti
14	3,68	3,69	6500	Motor Bergerak Ke Kanan
15	3,68	3,66	7000	Motor Bergerak Ke Kiri
16	3,68	3,7	7500	Motor Bergerak Ke Kanan
17	3,68	3,67	8000	Motor Bergerak Ke Kiri
18	3,68	3,68	8500	Motor Berhenti
19	3,68	3,68	9000	Motor Berhenti
20	3,68	3,68	9500	Motor Berhenti
21	3,68	3,68	10000	Motor Berhenti
22	3,68	3,68	10500	Motor Berhenti
23	3,68	3,68	11000	Motor Berhenti
24	3,68	3,69	11500	Motor Bergerak Ke Kanan
25	3,68	3,66	12000	Motor Bergerak Ke Kiri
26	3,68	3,69	12500	Motor Bergerak Ke Kanan
27	3,68	3,68	13000	Motor Berhenti
28	3,68	3,68	13500	Motor Berhenti
29	3,68	3,67	14000	Motor Bergerak Ke Kiri
30	3,68	3,68	14500	Motor Berhenti
31	3,68	3,67	15000	Motor Bergerak Ke Kiri

Pada Tabel 5.7 dapat dilihat bahwa hasil *sampling* menunjukkan jika plat *solatube* bergerak ke kanan dan ke kiri. Namun saat sensor mendeteksi nilai *setpoint*, plat *solatube* dapat berhenti meskipun tidak terlalu lama. Itu dikarenakan

sensor cahaya sangat sensitif terhadap perubahan cahaya yang terjadi. Sehingga akan mengakibatkan motor kembali bergerak ke kanan atau ke kiri. Grafik hasil pengujian ditunjukkan pada Gambar 5.8.

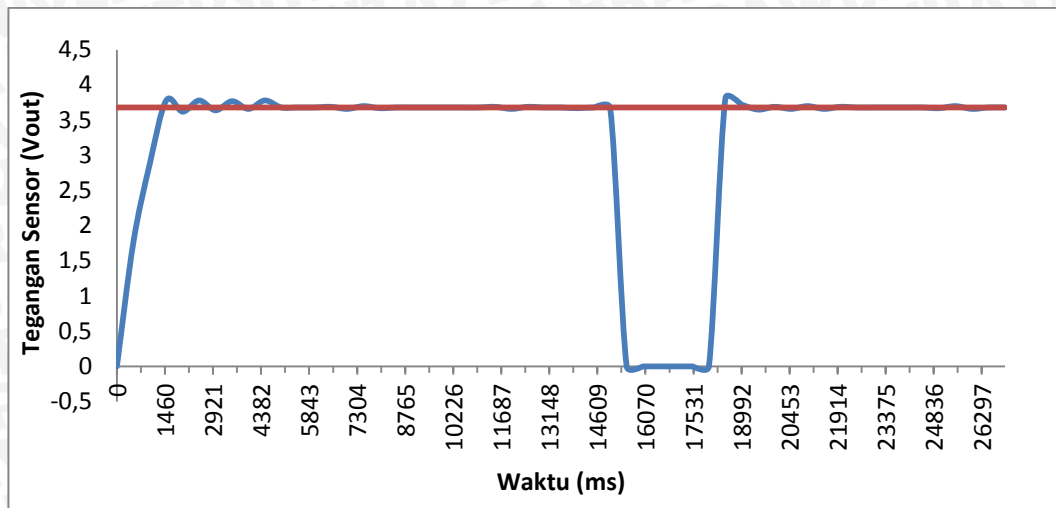


Gambar 5.8 Grafik Hasil Pengujian Dengan Kontroler

Pada Gambar 5.8 dapat dilihat jika sistem berada didaerah *setpoint* meskipun masih terdapat *error*. Namun demikian plat dapat berhenti saat sensor mendeteksi *setpoint*, dengan %Ess sebesar 0,03 %

$$\begin{aligned}
 \%Ess &= \frac{|Average\ speed\ steady-setpoint|}{setpoint} \times 100\% \\
 &= \frac{|3,679-3,68|}{3,68} \times 100\% \\
 &= 0,03\% \text{ dengan } settling\ time \text{ sebesar } 5 \text{ detik}
 \end{aligned}$$

Pada Gambar 5.9 merupakan grafik respon sistem saat diberikan gangguan, dengan cara menutup tabung atas *solatube* selama 3 detik sehingga cahaya yang masuk menjadi berkurang.



Gambar 5.9 Grafik Hasil Pengujian Dengan Kontroler Saat Diberikan Gangguan

Pada Gambar 5.8 dapat dilihat jika sistem dapat kembali pada daerah *setpoint* dan plat *solatube* juga dapat berhenti saat sensor mendeteksi cahaya sebesar 150 lux, dengan *settling time* sebesar 4 detik. Pada Tabel 5.8 dan Tabel 5.9 merupakan data hasil pengujian saat sistem diberikan suatu gangguan.

Tabel 5.8 Data Hasil Pengujian Dengan Kontroler Disertai Gangguan

No.	Set Point	Tegangan Sensor (Vout)	Waktu (ms)	Respon Motor
1	3,68	0	0	Motor Bergerak Ke Kiri
2	3,68	1,76	500	Motor Bergerak Ke Kiri
3	3,68	2,88	1000	Motor Bergerak Ke Kiri
4	3,68	3,78	1500	Motor Bergerak Ke Kanan
5	3,68	3,62	2000	Motor Bergerak Ke Kiri
6	3,68	3,78	2500	Motor Bergerak Ke Kanan
7	3,68	3,64	3000	Motor Bergerak Ke Kiri
8	3,68	3,77	3500	Motor Bergerak Ke Kanan
9	3,68	3,66	4000	Motor Bergerak Ke Kiri
10	3,68	3,78	4500	Motor Bergerak Ke Kanan
11	3,68	3,68	5000	Motor Berhenti
12	3,68	3,68	5500	Motor Berhenti
13	3,68	3,68	6000	Motor Berhenti
14	3,68	3,69	6500	Motor Bergerak Ke Kanan
15	3,68	3,66	7000	Motor Bergerak Ke Kiri
16	3,68	3,7	7500	Motor Bergerak Ke Kanan
17	3,68	3,67	8000	Motor Bergerak Ke Kiri
18	3,68	3,68	8500	Motor Berhenti
19	3,68	3,68	9000	Motor Berhenti

Tabel 5.9 Data Hasil Pengujian Dengan Kontroler Disertai Gangguan (lanjutan)

No.	Set Point	Tegangan Sensor (Vout)	Waktu (ms)	Respon Motor
20	3,68	3,68	9500	Motor Berhenti
21	3,68	3,68	10000	Motor Berhenti
22	3,68	3,68	10500	Motor Berhenti
23	3,68	3,68	11000	Motor Berhenti
24	3,68	3,69	11500	Motor Bergerak Ke Kanan
25	3,68	3,66	12000	Motor Bergerak Ke Kiri
26	3,68	3,69	12500	Motor Bergerak Ke Kanan
27	3,68	3,68	13000	Motor Berhenti
28	3,68	3,68	13500	Motor Berhenti
29	3,68	3,67	14000	Motor Bergerak Ke Kiri
30	3,68	3,68	14500	Motor Berhenti
31	3,68	3,67	15000	Motor Bergerak Ke Kiri
32	3,68	0	15500	Motor Bergerak Ke Kiri
33	3,68	0	16000	Motor Bergerak Ke Kiri
34	3,68	0	16500	Motor Bergerak Ke Kiri
35	3,68	0	17000	Motor Bergerak Ke Kiri
36	3,68	0	17500	Motor Bergerak Ke Kiri
37	3,68	0	18000	Motor Bergerak Ke Kiri
38	3,68	3,81	18500	Motor Bergerak Ke Kanan
39	3,68	3,72	19000	Motor Bergerak Ke Kanan
40	3,68	3,65	19500	Motor Bergerak Ke Kiri
41	3,68	3,69	20000	Motor Bergerak Ke Kanan
42	3,68	3,66	20500	Motor Bergerak Ke Kiri
43	3,68	3,7	21000	Motor Bergerak Ke Kanan
44	3,68	3,66	21500	Motor Bergerak Ke Kiri
45	3,68	3,69	22000	Motor Bergerak Ke Kanan
46	3,68	3,68	22500	Motor Berhenti
47	3,68	3,68	23000	Motor Berhenti
48	3,68	3,68	23500	Motor Berhenti
49	3,68	3,68	24000	Motor Berhenti
50	3,68	3,68	24500	Motor Berhenti
51	3,68	3,67	25000	Motor Bergerak Ke Kiri
52	3,68	3,7	25500	Motor Bergerak Ke Kanan
53	3,68	3,66	26000	Motor Bergerak Ke Kiri
54	3,68	3,68	26500	Motor Berhenti
55	3,68	3,68	27000	Motor Berhenti

BAB VI KESIMPULAN DAN SARAN

6.1. Kesimpulan

Dari hasil perancangan dan pengujian alat yang telah dilakukan, maka didapatkan:

1. Alat yang dirancang sudah sesuai dengan yang diharapkan, di mana plat *solatube* dapat bergerak membuka dan menutup serta berhenti saat sensor cahaya mendeteksi kuat cahaya sebesar 150 lux.
2. Dengan menggunakan metode *hand tuning* untuk menentukan nilai parameter kontroler PID, maka didapatkan parameter yang terbaik yaitu $K_p = 100$, $K_i = 0,5$, $K_d = 0,2$.
3. Setelah diimplementasikan, sistem dapat berjalan sesuai dengan yang diharapkan. Dengan *settling time* sebesar 5 detik dan *Ess* sebesar 0,03 %.

6.2 Saran

Disarankan untuk menggunakan sensor cahaya yang memiliki kualitas yang lebih baik dan mekanik yang lebih sempurna, serta dalam mengembangkan skripsi ini dapat ditambahkan cahaya dari lampu *led* apabila cahaya kurang dari yang diinginkan.