

# **BAB I**

## **PENDAHULUAN**

### **1.1. Latar Belakang**

Setiap hari manusia melaksanakan segala aktivitasnya baik di dalam ruangan (*indoor*) maupun di luar ruangan (*outdoor*). Saat ini, manusia lebih banyak beraktivitas di dalam sebuah ruangan, baik itu sebagai rumah, tempat bekerja, sekolah, dan lain sebagainya. Agar dapat melakukan kegiatannya secara nyaman, suatu penerangan cahaya pada ruangan diperlukan oleh manusia untuk dapat mengenali objek secara visual. Sebuah ruangan harus dapat memberikan kenyamanan dan kesehatan bagi orang yang tinggal di dalamnya agar nyaman dan sehat, di mana harus didukung oleh sistem penunjang yang baik, termasuk sistem sirkulasi cahaya. Sirkulasi cahaya yang sehat adalah sirkulasi yang dapat memberikan cahaya yang cukup untuk masuk dan keluar pada sebuah ruangan, sehingga cahaya yang dibutuhkan di ruangan tersebut menjadi tercukupi dan baik untuk kesehatan mata bagi penghuninya. Contohnya, cahaya yang dihasilkan oleh sinar matahari mampu untuk mengusir kelembaban sebuah ruangan. Umumnya, untuk pengaturan penerangan sebuah ruangan yang berjendela pada saat pagi hingga sore hari menggunakan prinsip buka tutup tirai secara manual yang apabila terlalu terang maka akan membuat silau dan juga panas, sehingga membuat mata menjadi tidak nyaman dan gerah dalam beraktivitas. Apabila ruangan terlalu gelap, maka hal tersebut akan membuat penglihatan menjadi buram yang dapat berdampak bagi kesehatan mata.

Membuka dan menutup tirai secara manual dengan tangan adalah hal yang biasa dilakukan dan dilihat setiap hari. Namun dengan kesibukan manusia yang bermacam-macam, terkadang membuat lupa bahkan malas untuk melakukan hal kecil tersebut. Meski terdengar sepele, namun hal ini terkadang menjadi penting untuk dilakukan, semisal ketika sudah nyaman duduk mengerjakan laporan dengan laptop tiba-tiba cahaya dari luar terang sekali maka hal yang dilakukan adalah beranjak dari tempat duduk dan menutup secara manual tirai tersebut.

Melihat kondisi dan kenyataan di sekitar yang dapat dilihat setiap hari, oleh karena itu penulis melakukan penelitian guna membantu dan mempermudah bagi

masyarakat untuk melaksanakan aktivitasnya yang banyak dengan nyaman, dengan mengambil judul skripsi "Sistem Pengendalian Intensitas Cahaya Menggunakan Tirai *Blind* dengan Kontroler Proporsional" dengan cara mengatur bilah pada tirai *blind* sebagai tempat masuknya cahaya yang digerakkan oleh motor arus searah (motor DC) dan pergerakan motor DC tersebut diatur oleh mikrokontroler. Perangkat keras yang dirancang bekerja berdasarkan pengaturan *loop* tertutup (*close loop*). Sistem pengaturan *close loop* berfungsi untuk membandingkan nilai yang keluar dari *plant* dengan keluaran yang diinginkan.

## 1.2. Rumusan Masalah

Mengacu pada permasalahan yang diuraikan pada latar belakang, maka rumusan masalah dapat ditekankan pada:

1. Bagaimana mengatur intensitas cahaya pada ruangan agar cahaya yang masuk sesuai dengan yang diinginkan dengan tirai *blind* ?
2. Bagaimana merancang sistem pengaturan intensitas cahaya yang masuk pada miniatur ruangan dengan menggunakan Kontroler Proporsional (P) ?
3. Bagaimana hasil perancangan kontroler P sebagai pengendali intensitas cahaya pada *plant* ?

## 1.3. Batasan Masalah

Untuk menekankan pada objek pembahasan yang ada, maka pada penelitian ini diberikan batasan masalah sebagai berikut:

1. Perangkat keras hanya digunakan untuk mengatur intensitas cahaya yang nilainya dipertahankan sesuai dengan yang diinginkan.
2. Sumber cahaya yang digunakan untuk pengaturan intensitas cahaya menggunakan cahaya lampu bohlam sebagai pengganti cahaya matahari dengan intensitas cahaya yang bisa berubah.

3. Gangguan yang berasal dari lampu bohlam yang dihalangi kertas atau sebagainya sehingga terjadi perbedaan naik turun pembacaan sensor.
4. Analisa dititik beratkan pada respon yang dihasilkan dari proses pengolahan mikrokontroler.
5. Perancangan *hardware* yang di gunakan adalah desain sendiri.
6. Pengaturan intensitas cahaya hanya pada siang hari, tidak membahas pada malam hari.
7. *Setpoint* ditentukan sebesar 140 lux dan 150 lux.

#### **1.4. Tujuan**

Tujuan dari penelitian ini adalah untuk merancang sistem pengendalian intensitas cahaya di miniatur ruang sehingga didapatkan kondisi pencahayaan miniatur ruang yang stabil jika cahaya luar berubah-ubah

## **1.5. Sistematika Penulisan**

Agar penyusunan laporan skripsi ini dapat mencapai sasaran dan tidak menyimpang dari judul yang telah ditentukan, maka diperlukan sistematika pembahasan yang jelas. Pembahasan dalam skripsi ini secara garis besar adalah sebagai berikut:

### **Bab I Pendahuluan**

Membahas latar belakang, rumusan masalah, batasan masalah, tujuan, dan sistematika penulisan.

### **Bab II Tinjauan Pustaka**

Membahas teori-teori yang mendukung dalam perancangan dan pembuatan alat.

### **Bab III Metode Penelitian**

Membahas metode penelitian dan perencanaan alat.

### **Bab IV Perancangan dan Pembuatan Alat**

Membahas perancangan alat yang meliputi spesifikasi, perencanaan blok diagram, prinsip kerja dan pembuatan alat, serta bagaimana penerapannya dalam sistem secara keseluruhan.

### **Bab V Pengujian dan Analisis**

Menjelaskan tentang pengujian alat dan analisa yang meliputi pengujian bagian blok sistem dan pengujian sistem secara keseluruhan.

### **Bab VI Penutup**

Membahas kesimpulan perancangan dan saran-saran yang diperlukan untuk pengembangan selanjutnya.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1. Tirai *Blind***

Tirai *blind* adalah tirai/penutup jendela yang berfungsi sebagai pengatur cahaya, melengkapi dekorasi dan keindahan berbagai macam tipe ruangan. dengan bahan pilihan tirai *blind* menjadi solusi untuk mereduksi efek negatif cahaya atau panas matahari. Tampak pada Gambar 2.1 tirai horisontal *blind*.



Gambar 2.1 Tirai *Horisontal Blind*  
Sumber: <http://iriswc.com>

#### **2.2. Ruang**

Suatu tempat yang di kelilingi dinding dan beratap yang berada di suatu gedung atau rumah. Dimana ruangan biasanya berpintu dan berjendela dengan fungsi masuknya aliran udara, cahaya dan akses masuk keruangan tersebut. Tampak pada Gambar 2.2 ilustrasi dari ruangan tersebut.



Gambar 2.2 Ruang Berjendela

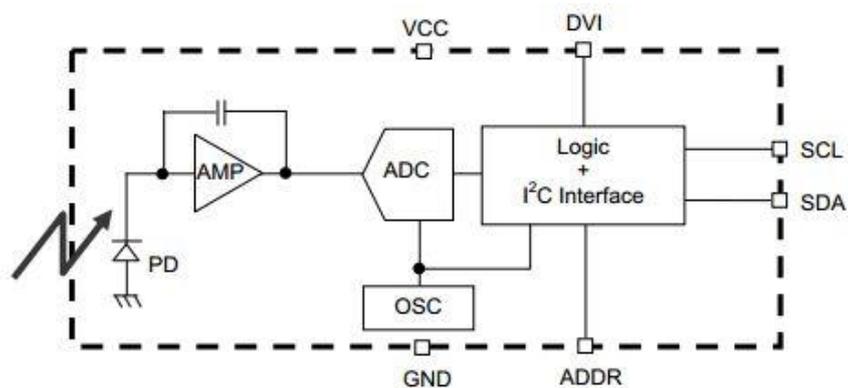
### 2.3. Sensor Intensitas Cahaya BH1750

Sensor intensitas cahaya modul dengan 16 bit *analog digital converter* (ADC) built-in yang output sinyal sudah dalam lux , tidak perlu perhitung tegangan seperti Light Dependent Resistor (LDR). Sensor BH1750 ini lebih akurat dan lebih mudah untuk menggunakan karena dikalibrasi secara digital, dari pada menggunakan versi foto dioda, atau LDR sederhana yang perlu hitungan untuk mendapatkan data intensitas. Dapat dilihat Gambar 2.3 modul sensor BH1750.



Gambar 2.3 Sensor Intensitas Cahaya BH1750

Dan berikut adalah blok diagram pada Gambar 2.4 sensor intensitas cahaya BH1750



Gambar 2.4 Blok Diagram Sensor BH1750  
Sumber: alldatasheet.com

Menurut data sheet Didalamnya terdapat *photodiode* dengan respon mendekati kemampuan mata manusia lalu terdapat *Integration- Operasional amplifier* (OPAMP) untuk mengubah arus dari *photodiode* menjadi tegangan kemudian di konversi ke digital oleh ADC lalu di kalkulasi dengan *logic plus I2c ( inter-integrated circuit) interface* berikut Tabel 2.1 Karakteristik Sensor Intensitas cahaya.

Tabel 2.1 Karakteristik Sensor

<b>Karakteristik Sensor Intensitas Cahaya BH1750</b>	
Range cahaya	0 lux hingga 65535 lux
Variasi pengukuran	+/- 20%
power supply voltage:	2.4 sampai 3.6Vdc
Suhu Operasi	-40 sampai 85 °C

Sumber : <http://www.Alldatasheet.com>

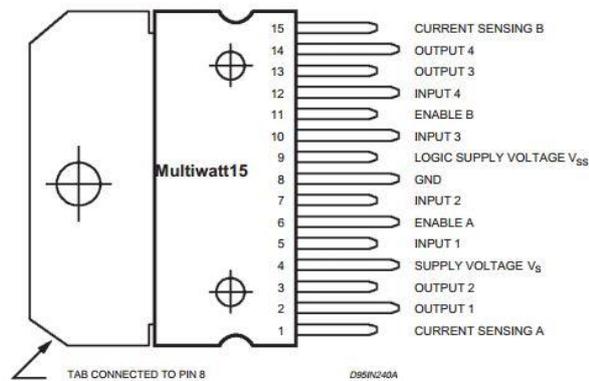
## 2.4. Motor DC

Motor DC adalah suatu mesin yang berfungsi mengubah tenaga listrik arus searah menjadi tenaga gerak atau tenaga mekanik berupa putaran rotor. Prinsip dasar motor arus searah adalah jika sebuah kawat penghantar dialiri arus listrik diletakkan dalam medan magnet, maka pada kawat itu akan bekerja suatu gaya yang mempunyai arah seperti pada kaidah tangan kiri.

## 2.5. Driver motor L298N

Biasa digunakan untuk mengendalikan motor agar motor dapat dikendalikan melalui mikrokontroler, sehingga motor dapat berputar, kecepatan motor bisa diatur termasuk arah putar.

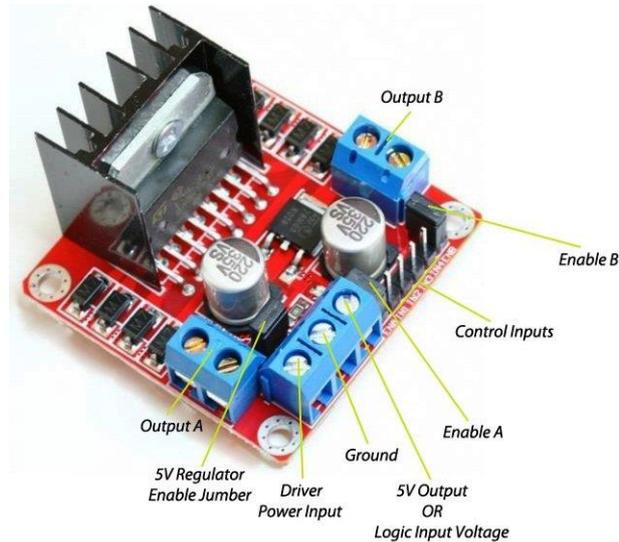
*Driver* Motor DC berfungsi sebagai penguat arus dan tegangan, sehingga motor mendapatkan suplay arus yang sesuai, L298N adalah *driver* motor dc, H-Bridge yang paling sederhana dan mudah untuk dipergunakan, untuk rangkaian pun menjadi lebih simple dan sederhana. dapat dilihat pada Gambar 2.5.



Gambar 2.5 Konstruksi *Pin Driver Motor* DC IC L298N

Sumber: Datasheet l289n

Dan berikut pada Gambar 2.6 modul *driver* motor dc tersebut



Gambar 2.6 Modul *Driver* L289N

Sumber: [electrosome.com](http://electrosome.com)

Fungsi pin pada modul:

1. *Driver power input*: input sumber tegangan untuk modul *driver* dan motor dc
2. *Ground*: untuk *ground*
3. *Enable A* dan *B*: pengaturan PWM untuk motor A atau motor B
4. *Control input*: pengaturan arah putaran motor DC
5. *Output A* dan *B*: *output* untuk disambung ke motor DC
6. *5v output*: jika *driver* input diberi 12v maka modul *driver output* menghasilkan 5v

## 2.6. Mikro Kontroler Arduino Uno

Mikrokontroler yang berbasis pada ATmega328. *Board* ini punya 14 digital *input / output* pin (dimana 6 pin dapat digunakan sebagai *output* PWM), 6 input *analog*, 16 MHz *osilator* kristal, koneksi USB, jack listrik tombol *reset*. Untuk mendukung mikrokontroler agar dapat digunakan, cukup hanya menghubungkan *Board* Arduino Uno ke komputer dengan menggunakan kabel USB atau sumber tegangan bisa didapat dari adaptor arus bolak balik (AC) ke arus searah (DC) atau

menggunakan baterai untuk menjalankannya. *Board* Arduino Uno memiliki fitur-fitur baru sebagai berikut:

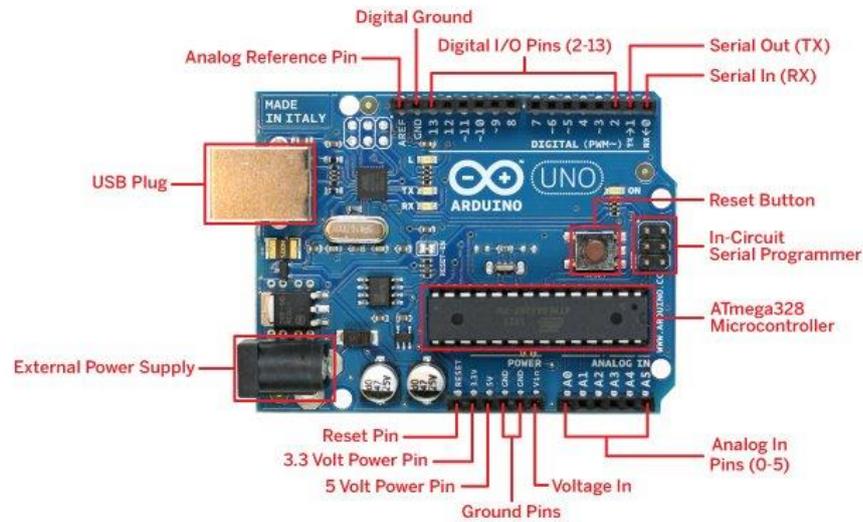
- Pin out 1.0: ditambah *serial data* (SDA) dan *serial clock* (SCL) pin yang dekat dengan pin *analog reference* (AREF) dan dua pin baru lainnya yang diletakkan dekat dengan pin *RESET*, *Input Output Reference* (IOREF) yang memungkinkan *shield-shield* untuk menyesuaikan tegangan yang disediakan dari *board*. Untuk ke depannya, *shield* akan dijadikan kompatibel atau cocok dengan *board* yang beroperasi dengan tegangan 5 volt dan dengan Arduino ide yang beroperasi dengan tegangan 3,3 volt. Yang ke dua merupakan sebuah *pin* yang tak terhubung, yang disediakan untuk tujuan kedepannya.
- Sirkuit *RESET* yang lebih kuat
- Atmega 16U2 menggantikan 8U2

Secara umum spesifikasi Arduino uno dalam Tabel 2.1

Tabel 2.2 Deskripsi Arduino Uno

Microcontroller	Atmega328
Operasi <i>Voltage</i>	5V
Input Voltage	7-12 V (Rekomendasi)
Input Voltage	6-20 V ( <i>limits</i> )
I/O	14 pin (6 pin untuk PWM)
Arus	50 mA
Flash Memory	32KB
Bootloader	SRAM 2 KB
EEPROM	1 KB
Kecepatan	16 Mhz

Dan berikut Gambar 2.7 bentuk Arduino Uno.



Gambar 2.7 Arduino Uno  
Sumber : robomart.com

### 2.6.1 Input dan Output

Masing-masing dari 14 pin digital pada Uno dapat digunakan sebagai *input* atau *output*, menggunakan fungsi *pinMode()*, *digitalWrite()*, dan *digitalRead()*. Fungsi-fungsi tersebut beroperasi di 5 volt. Setiap pin dapat memberikan atau menerima maksimum 40 mA dan memiliki resistor *pull-up* internal dari 20-50 K. Selain itu, beberapa pin memiliki fungsi khusus:

- Serial: 0 *receiver* (RX) dan 1 *transmit* (TX). Digunakan untuk menerima (RX) dan mengirimkan (TX) data TTL (*transistor-transistor logic*) *serial*. Pin ini terhubung ke pin yang sesuai dari chip ATmega8U2 USB-to-Serial TTL (*transistor-transistor logic*).
- Eksternal Interupsi: 2 dan 3. Pin ini dapat dikonfigurasi untuk memicu interupsi pada nilai yang rendah, tepi naik atau jatuh, atau perubahan nilai. Lihat *attach Interrupt ()* fungsi untuk rincian.

- PWM: 3, 5, 6, 9, 10, dan 11. Menyediakan 8-bit *output* PWM dengan *analog Write ()* fungsi.
- SPI: 10 *slave select* (SS), 11 *Master In Slave Out* (mosi), 12 *Master Out Slave In* (MISO), 13 *serial clock* (SCK). Pin ini mendukung komunikasi Serial Peripheral Interface (SPI) menggunakan perpustakaan SPI.
- LED: 13. Ada *built-in light emitting diode* (LED) terhubung ke pin digital 13. Ketika pin adalah nilai TINGGI, LED menyala, ketika pin adalah RENDAH, itu *off*.

### 2.6.2 Komunikasi

Arduino Uno memiliki sejumlah fasilitas untuk berkomunikasi dengan komputer, Arduino lain, atau mikrokontroler lain. ATmega328 ini menyediakan *universal asynchronous receiver/transmitter* (UART) TTL (5V) komunikasi serial, yang tersedia pada pin digital 0 (RX) dan 1 (TX). Sebuah ATmega16U2 pada saluran *board* ini komunikasi serial melalui USB dan muncul sebagai *com port virtual* untuk perangkat lunak pada komputer. *Firmware* Arduino menggunakan USB *driver* standar, dan tidak ada *driver* eksternal yang dibutuhkan. Namun, pada Windows, file.inf diperlukan. Perangkat lunak Arduino termasuk monitor serial yang memungkinkan data sederhana yang akan dikirim ke *board* Arduino. RX dan TX LED di *board* akan berkedip ketika data sedang dikirim melalui *chip* USB-to-serial dan koneksi USB ke komputer. ATmega328 ini juga mendukung komunikasi Inter-Integrated Circuit (I2C) dan SPI. Fungsi ini digunakan untuk melakukan komunikasi *inteface* pada sistem.

### 2.6.3 Catu Daya

Arduino Uno dapat diaktifkan melalui koneksi USB atau dengan catu daya eksternal. Sumber daya dipilih secara otomatis. Eksternal (non-USB) tegangan dapat diperoleh dari sebuah adaptor AC ke DC atau baterai. Adaptor dapat dihubungkan dengan mencolokkan sebuah *center-positive plug* yang panjangnya 2.1mm ke *power jack* dari *board*. Kabel *lead* dari baterai dapat dimasukkan ke dalam *header* pin *Ground* (Gnd) dan *Vin* dari konektor POWER.

*Board* Arduino Uno dapat beroperasi pada sebuah suplai eksternal dari 6 sampai 20 volt. Jika disuplai kurang dari 7 volt, contohnya ketika pin 5V mendapat suplai

kurang dari 5 volt dapat berakibat *board* arduino menjadi tidak stabil. Jika menggunakan lebih dari 12V, regulator tegangan bisa panas dan merusak *board*. Rentang yang dianjurkan adalah 7 - 12 volt.

Pin catu daya adalah sebagai berikut:

- VIN (*volt input*). Tegangan input ke *board* Arduino ketika menggunakan sumber daya eksternal (sebagai lawan dari 5 volt dari koneksi USB atau sumber daya lainnya diatur). Anda dapat menyediakan tegangan melalui pin ini, atau, jika memasok tegangan melalui colokan listrik, mengaksesnya melalui pin ini.
- 5V. Catu daya diatur digunakan untuk daya mikrokontroler dan komponen lainnya di *board*. Hal ini dapat terjadi baik dari VIN melalui regulator *on board*, atau diberikan oleh USB.
- 3,3 volt pasokan yang dihasilkan oleh regulator *on-board*. Menarik arus maksimum adalah 50 mA.

#### 2.6.4 Memori

ATmega328 memiliki 32 KB dengan 0,5 KB digunakan untuk *loading file* dan juga memiliki 2 KB dari SRAM (*static random access memory*) dan 1 KB dari EEPROM (Electrically Erasable Programmable Read-Only Memory).

#### 2.6.5 Programming

Arduino Uno dapat diprogram dengan perangkat lunak Arduino. Pilih Arduino Uno dari *Tool* lalu sesuaikan dengan mikrokontroler yang digunakan. ATmega328 pada Arduino Uno memiliki *bootloader* yang memungkinkan anda untuk meng-*upload* program baru tanpa menggunakan *programmer hardware* eksternal. Cara berkomunikasi menggunakan protokol dari bahasa C. Sistem ini dapat menggunakan perangkat lunak Atmel (Windows) untuk memuat *firmware* baru. Atau anda dapat menggunakan *header* dengan programmer eksternal.

#### 2.6.6 Perangkat Lunak (Arduino Ide)

Lingkungan *open-source* Arduino memudahkan untuk menulis kode dan meng-*upload* ke *board* Arduino. Ini berjalan pada Windows, Mac OS X, dan Linux. Berdasarkan Pengolahan dan perangkat lunak sumber terbuka lainnya.

### 2.6.7 Otomatis Software Reset

Tombol *reset* Arduino Uno dirancang untuk menjalankan program yang tersimpan didalam mikrokontroler dari awal. Tombol *reset* terhubung ke ATmega328 melalui kapasitor 100nf. Setelah tombol *reset* ditekan cukup lama untuk me-*reset chip*, *software IDE* Arduino dapat juga berfungsi untuk meng-*upload* program dengan hanya menekan tombol *upload* di *software IDE* Arduino.

### 2.7. Lcd

*Liquid Crystal Display* (LCD) adalah suatu jenis media tampil yang menggunakan kristal cair sebagai penampil utama. LCD sudah digunakan di berbagai bidang misalnya alat-alat elektronik seperti televisi, kalkulator ataupun layar komputer. Tipe LCD dot matrik dengan jumlah karakter 16\*2.

LCD sangat berfungsi sebagai penampil yang nantinya akan digunakan untuk menampilkan status kerja alat. Bentuk LCD 16\*2 dapat dilihat dalam Gambar 2.8

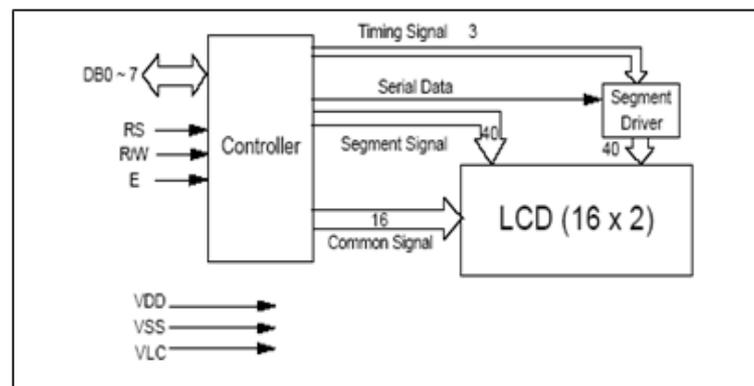


Gambar 2.8 LCD 1602  
Sumber dfrobot.com

Spesifikasi dari LCD ini adalah sebagai berikut:

1. Menampilkan 16 karakter pada tiap baris LCD dengan  $5 \times 7$  dot matrik.
2. Pembangkit karakter ROM (*Read-only Memory*) untuk 192 jenis karakter.
3. Pembangkit karakter RAM (*Random access memory*) untuk 8 jenis karakter.
4. 80 x 8 bit data RAM.
5. Otomatis *reset* pada saat dihidupkan.

Masukan yang diperlukan untuk mengendalikan modul LCD ini berupa bus data yang masih termultipleks dengan bus alamat serta 3 bit sinyal kontrol, yaitu RS (*register select*), R/W (*read/write*) dan E (*enable*). Sementara pengendali *dot matrix* LCD dilakukan secara internal oleh kontroler yang sudah terpasang pada modul LCD. Berikut ini adalah diagram blok dari LCD 1602A yang ditunjukkan dalam Gambar 2.9. (Anonymous, 2010).



Gambar 2.9. Diagram Blok LCD 1602A  
Sumber : Anonymous, 2001:3

Fungsi pin dari LCD 1602A ditunjukkan dalam Tabel 2.3.

Tabel 2.3 Data sheet lcd 1602.

No Pin	Nama Pin	Fungsi
16	V – BL	Sebagai ground dari backlight
15	V + BL	Sebagai kutub positif dari <i>backlight</i>
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 = register instruksi (tulis) 1 = register 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>

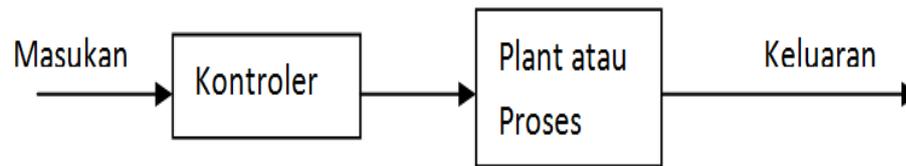
Sumber : *Datasheet* LCD 1602A

## 2.8. Loop Pengendalian

Secara umum bentuk *loop* sistem pengendalian dibagi menjadi dua macam, berdasarkan *loop* nya yaitu:

### 1. Sistem Pengendalian *Loop* Terbuka (*Open Loop Control System*)

Sistem pengendalian *loop* terbuka adalah sistem pengendalian yang keluarannya tidak dapat mempengaruhi aksi dari pengendaliannya. Jadi, pada sistem ini keluaran dari kontrolernya tidak diukur atau diumpan balikkan untuk membandingkan dengan masukannya (K. Ogata, 2014). Tampak Gambar 2.10 blok diagram *open loop*.



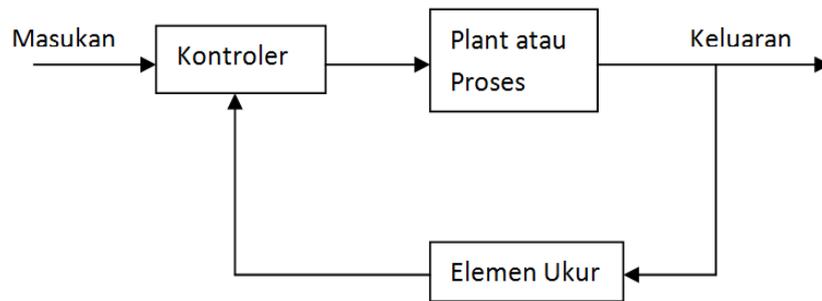
Gambar 2.10 Blok Diagram *Open Loop Control System*

Salah satu contoh sistem *loop* terbuka adalah sistem pengaturan temperatur ruangan. Untuk mendapatkan temperatur yang diinginkan, operator menggunakan pengalamannya untuk mengeset daya yang dibutuhkan sistem agar keluaran sistem yang berupa temperatur ruangan sesuai dengan temperatur ruangan yang diinginkan.

## 2. Sistem Pengendalian *Loop* Tertutup (*Close Loop Control System*)

Sistem pengendalian *loop* tertutup adalah sistem pengendalian yang sinyal keluarannya mempunyai pengaruh langsung pada aksi pengendaliannya (K. Ogata, 1985). Sinyal kesalahan yang bekerja yaitu antara sinyal masukan dan sinyal umpan balik yang disajikan ke kontroler disajikan sedemikian rupa untuk mengurangi kesalahan dan membawa keluaran sistem ke nilai yang dikehendaki.

Contohnya *thermostat* pada ruangan untuk menjalankan atau mematikan alat pemanas atau pendingin agar suhu tetap nyaman. Tampak Gambar 2.11 blok diagram *close loop*.



Gambar 2.11 Blok Diagram *Close Loop Control System*

Biasanya berupa pengukur tidak langsung seperti pengukur *level* dengan radar dan ultrasonik. Di dalam pengendalian *loop* tertutup terdapat tiga macam metode pengendalian, yaitu:

*a. Feed Back Control System*

*Feed back control system* adalah sistem pengendalian di mana besaran proses yang diatur dan diukur (PV) dibandingkan dengan nilai yang dikehendaki (SV) dan perbedaannya digunakan sebagai dasar untuk mengeliminir perbedaan yang ada (membuka dan menutup *control valve*).

*b. Feed Forward Control System*

Pada metode ini beban proses pengaturan diukur kemudian dibandingkan dengan beban normal dan bila ada perbedaan, maka perbedaan tersebut digunakan sebagai dasar untuk melakukan aksi antisipasi agar tidak terjadi penyimpangan pada *primary* proses variabel yang diatur.

### c. Cascade Control System

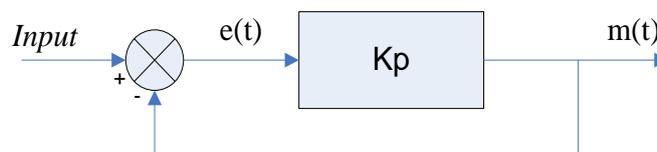
Kontrol *cascade* adalah kontrol yang melibatkan penggunaan dua buah pengontrol dengan keluaran dari pengontrol pertama merupakan titik pengaturan bagi pengontrol kedua. *Loop* umpan balik untuk salah satu pengontrol berada di dalam *loop* umpan balik untuk pengontrol yang lain. (W. Bolton, 2006: 290).

Kontrol kaskade meliputi penggabungan dua kontroler menjadi satu yang difungsikan secara bersamaan dan dikontrol secara bersama pula. (Khairul Ramadhan, 2014)

## 2.9. Kontroler Proporsional

Kontroler proporsional memiliki keluaran yang sebanding/proporsional dengan besarnya sinyal kesalahan (selisih antara besaran yang diinginkan dengan harga aktualnya). Secara lebih sederhana dapat dikatakan, bahwa keluaran kontroler proporsional merupakan perkalian antara konstanta proporsional dengan masukannya. Perubahan pada sinyal masukan akan segera menyebabkan sistem secara langsung mengubah keluarannya sebesar konstanta pengalinya.

Pada Gambar 2.12 menunjukkan blok diagram yang menggambarkan hubungan antara *input* (besaran referensi yang diinginkan), besaran aktual dengan besaran keluaran kontroler proporsional, dan besaran kesalahan (*error*). Sinyal kesalahan (*error*) merupakan selisih antara besaran *setting* dengan besaran aktualnya.



Gambar 2.12 Diagram Blok Kontroler Proporsional  
( Sumber: Ogata, 1995: 157)

Pada pengendali proporsional hubungan antara keluaran kontroler  $m(t)$  dan sinyal kesalahan  $e(t)$  adalah sebagai berikut :

$$m(t) = K_p e(t)$$

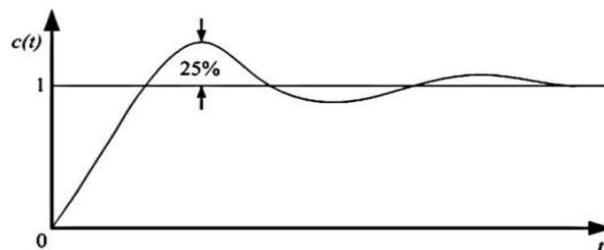
Dengan  $K_p$  adalah penguatan proporsional. Keluaran  $m(t)$  hanya tergantung pada  $K_p$  dan *error*, semakin besar *error* maka semakin besar koreksi yang dilakukan. Penambahan  $K_p$  akan menaikkan penguatan sistem sehingga dapat digunakan untuk memperbesar kecepatan respon dan mengurangi kesalahan keadaan mantap.

## 2.10. Perancangan PID Menggunakan Metode Ziegler- Nichols

Ziegler dan Nichols mengemukakan aturan-aturan untuk menentukan nilai dari gain proporsional  $K_p$ , waktu integral  $T_i$ , dan waktu derivatif  $T_d$  berdasarkan karakteristik respon transien dari *plant* yang diberikan. Penentuan parameter kontroler PID (*Proportional-Integral-Derivative*) atau penalaan kontroler PID tersebut dapat dilakukan dengan bereksperimen dengan *plant* (Ogata K., 1997).

Terdapat dua metode yang disebut dengan aturan penalaan Ziegler-Nichols, pada kedua metode tersebut memiliki tujuan yang sama yaitu untuk mencapai 25% *maximum overshoot* pada respon unit step.

Kurva respon unit step yang menunjukkan 25% *maximum overshoot* terlihat dalam Gambar 2.13

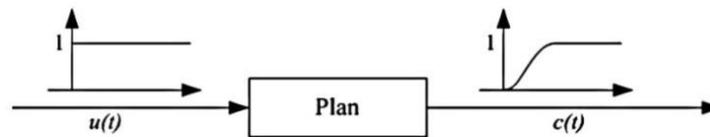


Gambar 2.13 Kurva Respon *Unit Step* Menunjukkan 25% *Maximum Overshoot*

(Sumber: Ogata K., 1997)

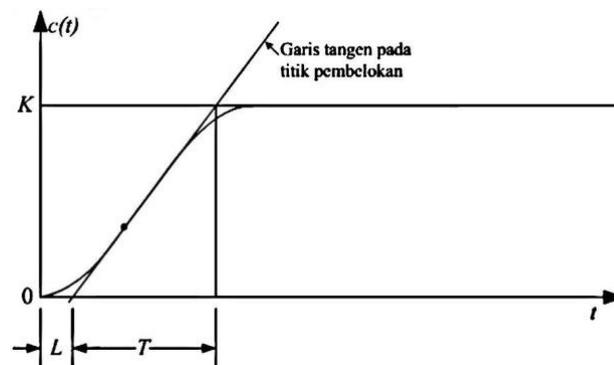
## a) Metode Pertama

Metode pertama atau sering disebut metode kurva reaksi, respon dari *plant* dapat diperoleh secara eksperimental dengan masukan berupa *unit step*, seperti yang ditunjukkan dalam Gambar 2.14.



Gambar 2.14 Respon *Plant* Terhadap Masukan Berupa *Unit Step*  
(Ogata K., 1997)

Jika dalam *plant* tersebut terdapat integrator atau *dominan complex-conjugate poles*, maka kurva respon unit step berbentuk seperti huruf S, seperti dalam Gambar 2.11, jika respon tidak memberikan bentuk kurva S, maka metode ini tidak berlaku.(Ogata K., 1997). Tampak Gambar 2.15 kurva berbentuk S.



Gambar 2.15 Kurva yang Berbentuk S

Sumber: Ogata K., 1997

Kurva berbentuk S tersebut dapat dikarakteristikan menjadi dua konstanta yaitu waktu tunda  $L$  dan konstanta waktu  $T$ . Waktu tunda dan konstanta waktu ditentukan dengan menggambar sebuah garis tangen pada titik pembelokan dari kurva S, dan menentukan perpotongan antara garis tangen dengan sumbu waktu  $t$  dan sumbu  $c(t) = K$ .

Fungsi alih  $C(s)/U(s)$  dapat dilakukan pendekatan dengan sistem orde satu dengan persamaan sebagai berikut:

$$\frac{C(s)}{U(s)} = \frac{Ke^{-Ls}}{Ts + 1}$$

Ziegler dan Nichols menyarankan untuk menentukan nilai-nilai dari  $K_p$ ,  $T_i$  dan  $T_d$  berdasarkan pada formula yang ditunjukkan dalam Tabel 2.4

Tabel 2.4 Aturan Penalaran Ziegler-Nichols Berdasarkan Respon Unit Step dari *Plant*

<b>Tipe Kontroler</b>	$K_p$	$T_i$	$T_d$
P	$\frac{T}{L}$	$\infty$	0
PI	$0.9 \frac{T}{L}$	$\frac{L}{0.3}$	0
PID	$1.2 \frac{T}{L}$	2L	0.5L

Aturan untuk metode pertama dengan persamaan sebagai berikut:

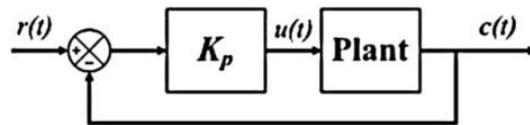
$$\begin{aligned} G_c(s) &= K_p \left( 1 + \frac{1}{T_i s} + T_d s \right) \\ &= 1.2 \frac{T}{L} \left( 1 + \frac{1}{2Ls} + 0.5Ls \right) \end{aligned}$$

$G_c(s)$  merupakan penguatan sistem yang memiliki nilai gain proporsional  $K_p$ , waktu integral  $T_i$ , dan waktu derivatif  $T_d$ .

b) Metode Kedua

Dalam metode kedua Ziegler-Nichols, mula-mula yang dilakukan adalah membuat  $T_i = \infty$  dan  $T_d = 0$ . Kemudian hanya dengan menggunakan tindakan kontrol proporsional, harga ditingkatkan dari nol ke suatu nilai kritis  $K_{cr}$ , disini mula-mula keluaran memiliki osilasi yang berkesinambungan (Jika keluaran tidak memiliki osilasi berkesinambungan untuk nilai  $K_p$  manapun yang telah diambil, maka metode ini tidak berlaku). Dari keluaran yang berosilasi secara berkesinambungan, penguatan kritis  $K_{cr}$  dan periode  $P_{cr}$  dapat ditentukan.

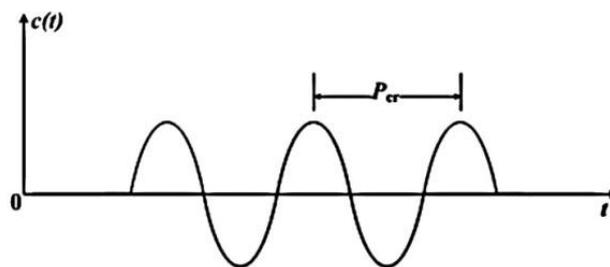
Diagram blok sistem *loop* tertutup dengan kontroler proporsional dapat dilihat dalam Gambar 2.16.



Gambar 2.16 Sistem *Loop* Tertutup dengan Kontroler Proporsional

(Sumber: Ogata K., 1997)

Osilasi berkesinambungan dengan periode  $P_{cr}$  dapat dilihat dalam Gambar 2.17.



Gambar 2.17 Osilasi Berkesinambungan dengan Periode  $P_{cr}$

(Sumber: Ogata K., 1997)

Ziegler dan Nichols menyarankan penyetelan nilai parameter  $K_p$ ,  $T_i$ , dan  $T_d$  berdasarkan rumus yang diperlihatkan dalam Tabel 2.5 (Ogata K., 1997).

Tabel 2.5 Aturan Dasar Ziegler-Nichols Berdasarkan *Critical Gain*  $K_{cr}$  dan *Critical Period*  $P_{cr}$

<b>Tipe Kontroler</b>	$K_p$	$T_i$	$T_d$
P	0.5 $K_{cr}$	$\infty$	0
PI	0.45 $K_{cr}$	$\frac{1}{1.2} P_{cr}$	0
PID	0.6 $K_{cr}$	0.5 $P_{cr}$	0.125 $P_{cr}$

Aturan untuk metode kedua dengan persamaan sebagai berikut:

$$G_c(s) = K_p \left( 1 + \frac{1}{T_i s} + T_d s \right)$$

$$= 0.6 K_{cr} \left( 1 + \frac{1}{0.5 P_{cr} s} + 0.125 P_{cr} s \right)$$

$G_c(s)$  merupakan penguatan sistem yang memiliki nilai gain proporsional  $K_p$ , waktu integral  $T_i$ , dan waktu derivatif  $T_d$ .

Tetapi dalam skripsi kali ini kontrol hanya menggunakan kontrol Proporsional sehingga hanya mencari  $K_p$  saja dan menggunakan metode pertama Ziegler- Nichol

## **BAB III**

### **METODOLOGI**

Dalam menyusun skripsi ini berdasarkan pada masalah yang bersifat aplikatif yang diwujudkan dalam bentuk prototipe yaitu perencanaan dan perealisasiian alat agar terbentuk unit kerja sesuai dengan yang di rencanakan dengan acuan rumusan masalah

Hal-hal yang dilakukan untuk merealisasikan alat yang akan dibuat secara umum adalah sebagai berikut:

- Studi literatur.
- Perancangan alat.
- Realisasi pembuatan alat.
- Pengujian alat dan analisa data.
- Pengambilan kesimpulan.

#### **3.1. Studi Literatur**

Studi literatur skripsi ini dengan acuan pada prinsip kerja alat yang dirancang meliputi sistem pengendalian tirai *blind* tersebut dengan mikrokontroler Arduino Uno, karakteristik dari komponen yang digunakan serta studi tentang pengendalinya yaitu mikrokontroler Aduino Uno. Hal hal yang berhubungan dengan studi literatur tersebut adalah:

- a. Studi tentang teori dasar
  - Teori tentang arduino uno.
  - Spesifikasi suatu komponen.
- b. Studi tentang rangkaian sensor
  - Karakteristik spesifikasi sensor BH1750 dan rangkaian elektronik.
  - Penerapan sensor BH1750.
  - Karakteristik komponen elektronik yang digunakan diantaranya meliputi LCD 16\*2, Mikrokontroler Arduino Uno dan motor dc.

### 3.2. Perancangan alat

Perancangan sistem dilakukan sebagai langkah awal sebelum terbentuknya suatu sistem beserta rangkaian elektronik pendukungnya, hal ini dimaksudkan agar sistem pengendalian pada mesin pengering gabah tersebut dapat berjalan sesuai dengan deskripsi awal yang telah direncanakan. Perancangan sistem yang dilakukan meliputi:

1. penentuan spesifikasi alat.
2. pembuatan diagram blok sistem keseluruhan.
3. perancangan mekanik.
4. perancangan perangkat elektrik yang terdiri dari mikrokontroler dan rangkaian sensor.
5. Perancangan perangkat lunak menggunakan IDE arduino.

#### 3.2.1 Perancangan perangkat keras

1. Penentuan spesifikasi sistem yang akan dibuat, meliputi:
  - a. Penentuan deskripsi kerja sistem secara keseluruhan.
  - b. Arduino UNO dan rangkaian elektronik pendukung.
2. Penentuan rangkaian yang digunakan.
3. Pemrograman arduino yang dapat digunakan sebagai pengontrol tirai *blind* sehingga intensitas cahaya yang masuk pada miniatur ruangan sesuai dengan yang diinginkan.

#### 3.2.2 Perancangan perangkat lunak

Setelah mengetahui perangkat keras yang dirancang, maka dibutuhkan perangkat lunak untuk mengendalikan dan mengatur kerja dari alat ini. Parameter yang diperoleh dari hasil perhitungan kemudian diterapkan kedalam Arduino Uno dengan menggunakan bahasa pemrograman C.

### 3.3. Pengujian alat

Setelah semua komponen pada alat sudah terhubung sesuai dengan diagram blok sistem yang telah dirancang dan perangkat lunak untuk mendukung sistem telah dibuat, maka diadakan pengujian dan analisa alat. Metode pengujian alat adalah sebagai berikut:

- Menguji pada tiap-tiap blok rangkaian.
- Kalibrasi pada sensor yang digunakan.
- Menggabungkan beberapa blok rangkaian menjadi sebuah sistem.
- Mengadakan pengujian sistem secara keseluruhan.
- Mengevaluasi hasil pengujian sistem secara keseluruhan.

### 3.4. Pengembalian kesimpulan dan saran

Pengambilan kesimpulan dan saran dilakukan setelah didapatkan hasil dari pengujian. Jika hasil yang diperoleh telah sesuai dengan spesifikasi yang direncanakan maka alat tersebut telah memenuhi harapan dan memerlukan pengembangan untuk penyempurnaan. Perancangan sistem

Perencanaan sistem dilakukan sebagai langkah awal sebelum terbentuknya suatu sistem beserta rangkaian elektronik pendukungnya, hal ini dimaksudkan agar sistem pengidentifikasi dapat berjalan sesuai dengan yang telah direncanakan.

Perancangan sistem yang dilakukan meliputi:

- Penentuan motor DC yang digunakan sebagai penggerak tirai *blind*.
- Penentuan spesifikasi sistem yang akan dibuat, meliputi:
  - a) Penentuan deskripsi kerja sistem secara keseluruhan.
  - b) Kontrol P dan rangkaian elektronik pendukung.
    - Perancangan program pada mikrokontroler ATmega sebagai pusat pengontrol dengan kontrol P.

### 3.5. Perancangan Perangkat Keras

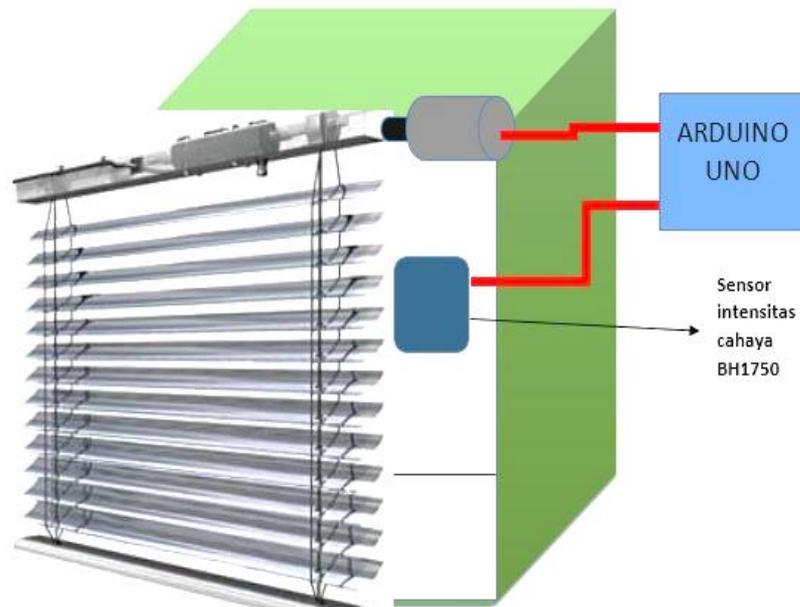
Perancangan perangkat keras dilakukan sebagai langkah awal sebelum terbentuknya suatu sistem beserta rangkaian elektronik pendukungnya, hal ini dimaksudkan agar sistem tersebut dapat berjalan sesuai dengan deskripsi awal yang telah direncanakan. Perancangan alat yang dilakukan meliputi:

- a. Penentuan deskripsi kerja sistem secara keseluruhan.
- b. Mikrokontroler ATmega dan rangkaian elektronik pendukung.

Penentuan rangkaian yang digunakan.

- a. Pemrograman Arduino Uno sehingga dapat digunakan sebagai pengontrol tirai *blind* sehingga intensitas cahaya yang masuk pada miniatur ruangan sesuai set point yang diinginkan.

Dan berikut pada Gambar 3.1 skema rancangan alat.



Gambar 3.1 Sketsa Perancangan Alat.

### **3.6. Pengujian Alat**

Setelah semua komponen pada alat sudah terhubung sesuai dengan blok diagram sistem yang telah dirancang dan perangkat lunak untuk mendukung sistem telah dibuat, maka diadakan pengujian dan analisa alat. Metode pengujian alat adalah sebagai berikut:

- a. Menguji pada tiap-tiap blok rangkaian.
- b. Kalibrasi pada sensor yang digunakan.
- c. Menggabungkan beberapa blok rangkaian menjadi sebuah sistem.
- d. Mengadakan pengujian sistem secara keseluruhan.
- e. Mengevaluasi hasil pengujian sistem secara keseluruhan.

### **3.7. Pengambilan Kesimpulan Dan Saran**

Kesimpulan diambil berdasarkan data yang diperoleh dari pengujian sistem secara keseluruhan. Jika hasil yang didapatkan telah sesuai dengan yang direncanakan sebelumnya, maka sistem kendali tersebut telah berhasil memenuhi harapan dan tentunya memerlukan pengembangan lebih lanjut untuk penyempurnaan.



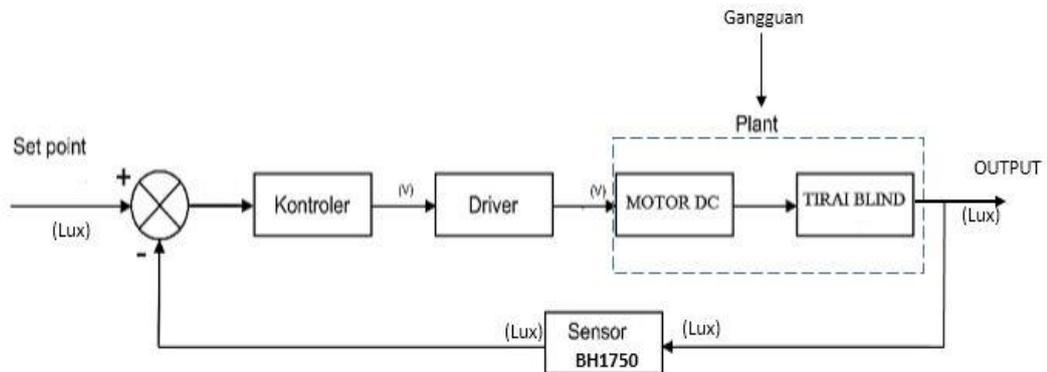
## BAB IV PERANCANGAN DAN PEMBUATAN ALAT

### 4.1. Perancangan Sistem

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

1. Perancangan perangkat keras (*hardware*) terdiri atas perancangan sensor cahaya dengan modul sensor BH1750 , motor DC , rangkaian *driver* motor.
2. Pembuatan dan penyelesaian program perangkat lunak (*software*) Arduino untuk sensor cahaya modul sensor BH1750 dan sinyal PWM untuk pengendalian *driver* motor.

### 4.2. Diagram Blok Sistem



Gambar 4.1 diagram blok sistem

Keterangan dari blok diagram sistem tersebut adalah sebagai berikut:

1. *Setpoint* nya berupa kuat cahaya (lux) .
2. Pusat pengendalian sistem menggunakan arduino UNO 328p yang memberikan sinyal keluaran berupa *pulse width modulation* (PWM) ke *driver* motor L293N.
3. Motor DC sebagai penggerak tirai *blind* buka menutup
4. Sensor BH1750 digunakan sebagai *feedback*.

#### 4.3. Spesifikasi Alat

Pengendalian *plant* dengan Arduino Uno dengan mikrokontroler ATmega328 dengan spesifikasi *plant* sebagai berikut:

1. Ukuran miniatur ruangan, panjang 40 cm, tinggi 33 cm dan lebar 30 cm.
2. Sensor yang digunakan adalah sensor intensitas cahaya *BH1750* dengan *output* sensor berupa lux.
3. Aktuatornya menggunakan tirai dengan motor DC yang mendapat sinyal masukan dari *output* PWM modul *driver* motor IC L293N.

#### 4.4. Prinsip Kerja Sistem

Cara kerja sistem sebagai berikut:

1. Catu daya dari adaptor 12V yang digunakan untuk catu daya motor dc yang tersambung ke modul *driver* motor dan juga sebagai catu daya Arduino Uno dan Arduino tersebut menyuplai tegangan untuk LCD dan pin enable motor dc untuk pengaturan PWM dan arah motor dc.
2. Komunikasi antara Arduino uno dengan PC menggunakan usb type B yg terhubung di Arduino Uno.
3. sensor intensitas cahaya *BH1750* yang digunakan sinyal keluaran berupa sinyal digital yang berupa intensitas cahaya lux.
4. Sinyal digital lux tersebut masuk ke mikrokontroler kemudian diproses menjadi sinyal kontrol berupa PWM.
5. Sinyal kontrol dari mikrokontroler masuk ke *driver* motor IC L293. *Driver* berfungsi menguatkan sinyal yang dihasilkan mikrokontroler dari 0-5 Volt menjadi 0-12 Volt.
6. Dengan PWM yang diatur maka motor DC akan menggerakkan tirai *blind* bergerak menutup atau membuka agar cahaya yang masuk ke dalam rumah kaca sesuai dengan set point yang ditetapkan.
7. Motor DC akan berhenti bergerak saat kuat cahaya mencapai *setpoint* yang diinginkan.

#### 4.5. Perancangan Perangkat Keras (Hardware)

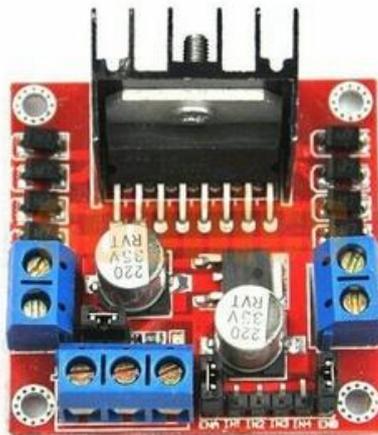
Berdasarkan diagram balok perancangan alat yang telah disusun, maka meliputi perancangan rangkaian sensor cahaya *light dependent resistor*, minimum sistem ATmega328, rangkaian catu daya dan rangkaian *driver* pengendali *PWM*. Di bawah ini adalah penjelasan masing-masing rangkaian penyusun keseluruhan alat.

##### 4.5.1 Modul Sensor Cahaya BH1750

Sensor cahaya pada perancangan ini berfungsi sebagai sensor pengukur atau pendeteksi kuat cahaya. Keluaran dari sensor cahaya ini digunakan sebagai masukan untuk perhitungan algoritma P yang dirancang. Keluaran sensor sudah dalam lux.

##### 4.5.2 *Driver* Motor L298N Dual H-Bridge

Modul pengendali motor dc digunakan untuk mengendalikan arah dan putaran motor DC yang menjadi penggerak plat. Rangkaian ini dihubungkan dengan mikrokontroler ATmega 328(Mikrokontroler utama). *Driver* pengendali pada perancangan ini menggunakan *driver* L298N yaitu sebuah perangkat keras untuk menggerakkan motor DC. Modul *driver* L298N ditunjukkan dalam gambar 4.2.



Gambar 4.2 Modul *driver* L298N

### 4.5.3 Mikrokontroler 328P

Pada sistem pengendalian *plant* ini digunakan Arduino Uno sebagai pengolah data. 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 baterai untuk menjalankannya. Fungsi setiap pin ditunjukkan dalam Tabel 4.1.

Tabel 4.1 Fungsi pin Arduino

NO	Pin	Fungsi
1	Vin	Jalur masukan 5V
2	GND	Jalur masukan GND
3	A5	Masukkan modul sensor BH1750
4	A4	Masukkan modul sensor BH1750
5	7	RS LCD
6	6	Enable pada LCD
7	5	D1 LCD
8	4	D2 LCD
9	3	D3 LCD
10	2	D4 LCD
11	12	Input1 <i>driver</i> motor
12	13	Input2 <i>driver</i> motor
13	11	Enable A, PWM untuk motor

#### 4.5.4 Rangkaian Display Lcd

LCD ditemukan pada alat elektronika yang memerlukan penampilan data atau fungsi, sehingga pemakai dapat mengerti dengan informasi yang ditampilkan oleh alat. LCD menggunakan 8 bit data (*bidirectional bus*) dan 3 buah sinyal kontrol yaitu RS, R/W dan E. Ketiga sinyal kontrol tersebut mempunyai fungsi sebagai berikut:

1. RS digunakan untuk memilih register yaitu register IR (instruction register) atau DR (data register).
2. R/W digunakan untuk memilih fungsi membaca atau menulis pada kedua register IR dan DR.
3. E digunakan untuk memberikan sinyal pada bahwa data akan ditulis atau dibaca ke register.

Mode pengiriman data antara mikrokontroler dengan LCD dapat dilakukan dengan mode transfer data 4 bit. Konfigurasi terminal I/O pada sebuah LCD biasanya akan tampak seperti dalam Tabel 4.2.

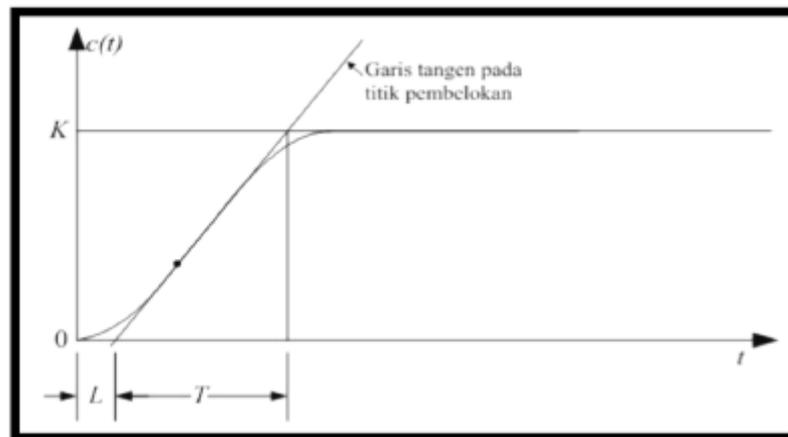
Tabel 4.2 Fungsi penyemat LCD M1632

<b>Penyemat</b>	<b>Simbol</b>	<b>Logika</b>	<b>Keterangan</b>
1	Vss	-	Catu Daya 0 Volt (Ground)
2	Vcc	-	Catu Daya 5 Volt
3	Vee	-	Catu daya untuk LCD
4	RS	H/L	H:Masukan Data, L:Masukan Instruksi
5	R/W	H/L	H: Baca (Read), L: Tulis (Write)
6	E	H/L (L)	Enable Signal
7	DB0	H/L	Data Bit 0
8	DB1	H/L	Data Bit 1
9	DB2	H/L	Data Bit 2
10	DB3	H/L	Data Bit 3
11	DB4	H/L	Data Bit 4
12	DB5	H/L	Data Bit 5
13	DB6	H/L	Data Bit 6
14	DB7	H/L	Data Bit 7
15	A	-	Backlight 4-4,2 Volt ; 50-200 mA
16	K	-	Backlight 0 Volt (ground)

#### 4.6. Perancangan Kontroler

Pada sub bab bagian ini membahas tentang perancangan sistem kontroler yang akan digunakan. Pada *plant* ini menggunakan kontrol proporsional saja dengan bertujuan memperkecil kepekaan sistem terhadap gangguan sehingga kontrol proporsional saja sudah cukup efektif dalam *plant* ini

Ziegler dan Nichols mengemukakan aturan-aturan untuk menentukan nilai dari gain proporsional  $K_p$ , waktu integral  $T_i$ , dan waktu derivative  $T_d$  berdasarkan karakteristik respon transien dari *plant* yang diberikan. Penentuan parameter kontroler PID atau penalaan kontroler PID tersebut dapat dilakukan dengan bereksperimen dengan *plant* (Ogata, K., 1997). Pada pengerjaan skripsi ini digunakanlah metode *Ziegler Nichols* yang pertama sehingga mendapatkan sinyal kurve S seperti pada Gambar 4.3.



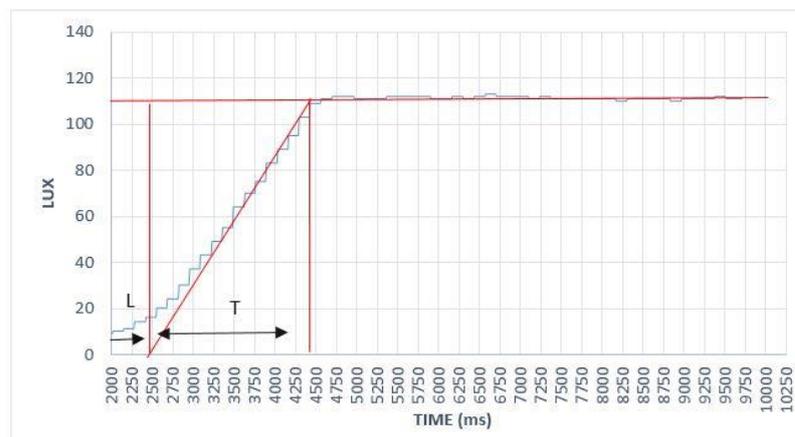
Gambar 4.3 Kurva Respon yang Berbentuk S (Ogata, K., 1997)

Setelah mendapatkan nilai  $L$  dan  $T$  kemudian menentukan nilai  $K_p$ ,  $T_i$  dan  $T_d$  sesuai ketentuan pada Gambar 4.4.

Tipe Kontrol	$K_p$	$T_i$	$T_d$
P	$\frac{T}{L}$	$\infty$	0
PI	$0,9 \frac{T}{L}$	$\frac{L}{0,3}$	0
PID	$1,2 \frac{T}{L}$	$2L$	$0,5 L$

Gambar 4.4 Aturan Metode 1 Ziegler-Nichols (Ogata, K., 1997)

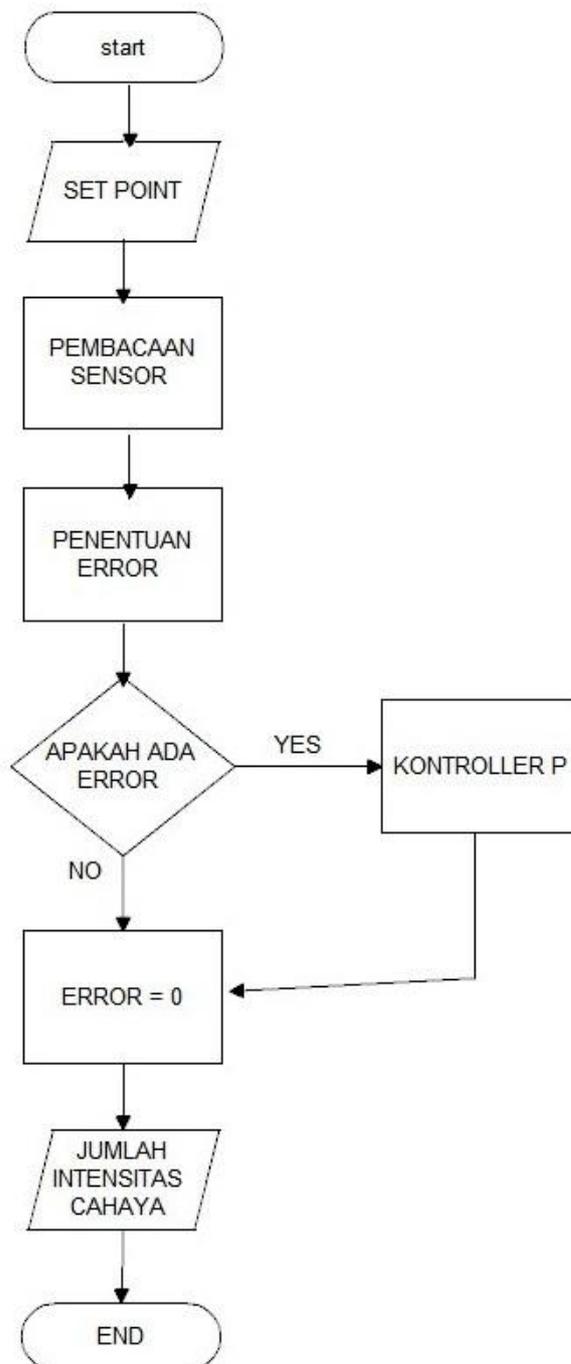
Dan berikut grafik pada *plant* ini seperti terlihat pada Gambar 4.5.



Gambar 4.5. Respons Intensitas Cahaya

Dimana di dapat dilihat dalam grafik bahwa didapatkan  $L=2,5$  detik dan  $T=1,8$  detik lalu nilai mengacu pada tabel metode pertama Ziegler Nichols dimasukkan dalam ketentuan kontroler P sehingga di dapatkan  $K_p = 0,7$ . Nilai  $K_p$  yang didapatkan akan digunakan dalam pengujian untuk sistem secara keseluruhan. Penggunaan kontroler P saja bertujuan agar mudah di aplikasikan, mudah di *tuning*, dan *offset* yang dapat diterima untuk rancangan *plant* sebesar 10%.

#### 4.7. Flowchart Perancangan Perangkat Lunak



Gambar 4.6 *Flowchart* Keseluruhan Sistem



## **BAB V**

### **PENGUJIAN DAN ANALISIS**

Tujuan pengujian dan analisis ini adalah untuk menentukan apakah alat yang telah dibuat berfungsi dengan baik dan sesuai dengan apa yang diinginkan. 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 BH1750 dengan perbandingan luxmeter.
2. Pengujian *driver* motor DC.
3. Pengujian kecepatan motor DC.
4. Pengujian LCD.
5. Pengujian seluruh sistem tanpa gangguan .
6. Pengujian sistem dengan gangguan.

#### **5.1. Pengujian Sensor Cahaya BH1750 dengan Luxmeter**

Dalam pengujian sensor cahaya BH1750 terdapat beberapa aspek yang harus diperhatikan, antara lain:

##### **5.1.1 Tujuan**

Mengetahui intensitas cahaya yang terbaca pada sensor intensitas Sensor Cahaya BH1750 telah sesuai dengan yang *setpoint*.

##### **5.1.2 Peralatan yang digunakan**

1. Sensor Intensitas Cahaya BH1750.
2. Luxmeter Krisbow KW06-288.
3. Arduino UNO 328p.

### 5.1.3 Langkah pengujian

1. *Photodetector* pada luxmeter di letakkan samping sensor cahaya BH1750 sehingga bersebelahan.
2. Program pada Arduino IDE pembacaan sensor lux dimana output sensor sudah dalam lux.
3. Sumber cahaya yang bisa di terangkan dan di redupkan diatur sedemikian rupa untuk mencapai lux yang diinginkan.
4. Kemudian dicatat pembacaan sensor nya dan pembacaan pada Luxmeter Krisbow KW06-288.

### 5.1.4 Hasil Pengujian

Setelah melakukan prosedur pengujian, didapatkan data hasil pengujian sensor BH1750. Data hasil pengujian ditunjukkan pada Tabel 5.1.

Tabel 5.1 Hasil Pengukuran Intensitas Cahaya Antara Luxmeter Dan Sensor BH1750.

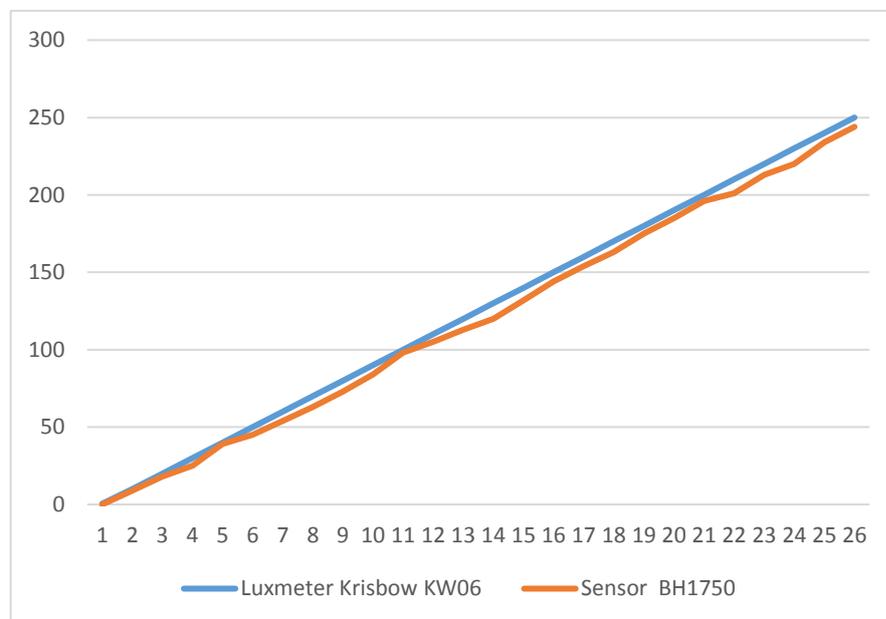
<b>NOMOR</b>	<b>LUXMETER</b>	<b>BH1750</b>
1	0	0
2	10	9
3	20	18
4	30	25
5	40	39
6	50	45
7	60	54
8	70	63
9	80	73
10	90	84
11	100	98
12	110	105
13	120	113
14	130	120
15	140	132
16	150	144
17	160	154
18	170	163
19	180	175
20	190	185
21	200	196
22	210	201
23	220	213
24	230	220
25	240	234
26	250	244

Perhitungan kesalahan pembacaan sensor:

$$\% \text{ kesalahan} = \frac{(\sum \text{Hasil Pengukuran} - \sum \text{Referensi})}{\sum \text{Referensi}} \times 100\%$$

$$\% \text{ kesalahan} = \frac{(3107 - 3250)}{3250} \times 100\% = 4,4\%$$

Tabel 5.1 memperlihatkan bahwa nilai pembacaan yang dihasilkan oleh sensor intensitas cahaya BH1750 sudah baik dan sangat mendekati nilai pada luxmeter Krisbow dengan nilai persen kesalahan sebesar 4,4% . Berikut Pembacaan lux meter dengan sensor dalam bentuk grafik pada Gambar 5.1.



Gambar 5.1 Hasil Pengukuran Intensitas Cahaya

## 5.2. Pengujian *Driver* Motor DC

Dalam pengujian *driver* motor dc terdapat beberapa hal yang harus dilakukan, antara lain:

### 5.2.1 Tujuan

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

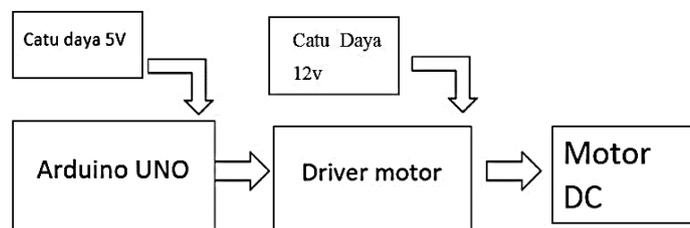
### 5.2.2 Peralatan yang digunakan

1. Arduino UNO.
2. *Driver* motor L293n.
3. Motor dc.
4. Catu daya adaptor 12v.

### 5.2.3 Langkah pengujian

1. Merangkai peralatan.
2. Mengisi mikrokontroler arduino UNO dengan program yang mengeluarkan output PWM di pin digital.
3. Mengaktifkan catu daya 9v untuk *driver* dan output 5v untuk daya Arduino UNO.
4. Mencatat pergerakan motor DC.

Berikut pada gambar 5.2 blok penempatan komponennya.



Gambar 5.2 Blok Penempatan

### 5.2.4 Hasil pengujian

Tabel 5.2 Hasil Pengujian *Driver* Motor DC

No	Input			Output
	In3	In4	Enable B	Arah Putaran Motor DC
1	X	X	0	Pengereman atau Diam
2	0	0	1	Pengereman atau Diam
3	0	1	1	Motor Berputar Searah Jarum Jam
4	1	0	1	Motor Berputar Berlawanan Arah Jarum Jam
5	1	1	1	Pengereman atau Diam

Berdasarkan data hasil pengujian pada Tabel 5.2 diketahui bahwa motor dc akan berputar searah jarum jam (*clockwise*) ketika pin In3 *low* dan pin in4 *high*, dan apabila pin In3 *high* dan pin in4 *low* maka motor akan berputar berlawanan arah jarum jam (*counter clockwise*). 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:

#### **5.3.1 Tujuan**

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

#### **5.3.2 Peralatan yang digunakan**

1. Arduino UNO 328p.
2. Multitester.
3. Tachometer.
4. Arduino IDE.

#### **5.3.3 Langkah pengujian**

Mengukur kecepatan motor dengan nilai *Pulse Width Modulation* (PWM) yang berbeda dengan menggunakan tachometer, sehingga didapatkan kecepatan yang diinginkan lalu mengukur besar tegangan yang keluar dari *driver*

#### **5.3.4 Hasil pengujian**

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

Tabel 5.3 Hasil Pengujian *Driver*

PWM	Kecepatan putar (rpm) pengujian ke-			Rata- Rata Rpm	<i>Vout Driver</i>
	1	2	3		
0	0	0	0	0	0
25	4,6	4,5	4,3	4,6	1,9
50	5	5,2	5,1	5,1	3,09
75	6,1	6,3	6	6,1	4,26
100	7	7	7,2	7	5,43
125	8,1	8,4	8	8,1	6,61
150	9,2	8,9	9,2	9,1	7,78
175	10	10	10,1	10	8,95
200	11,2	11	11	11	10,11
225	12	12,1	12,3	12,2	11,2
250	13,1	13,2	13	13,1	11,95

Dengan begitu seperti terlihat pada Tabel 5.3 motor dc bekerja dengan baik

#### 5.4. Pengujian *LCD*

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

##### 5.4.1 Tujuan

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

#### 5.4.2 Peralatan yang digunakan

1. Arduino Uno 328p.
2. LCD.
3. Catu daya 5V.

#### 5.4.3 Langkah Pengujian

1. Membuat program dengan menggunakan Arduino Ide untuk menampilkan data karakter nama.
2. Menghubungkan pin LCD dengan mikrokontroler.
3. Meng-*upload* program ke dalam Arduino UNO.

#### 5.4.4 Hasil Pengujian

Hasil pengujian LCD dapat dilihat pada Gambar 5.3.



Gambar 5.3 Hasil Pengujian LCD

## **5.5. Pengujian seluruh Sistem**

Dalam pengujian seluruh sistem ada hal-hal yang perlu diperhatikan yaitu:

### **5.5.1 Tujuan**

Dilakukan untuk mengetahui hasil respon sistem dari perangkat keras dan perangkat lunak dengan menggunakan kontroler proporsional.

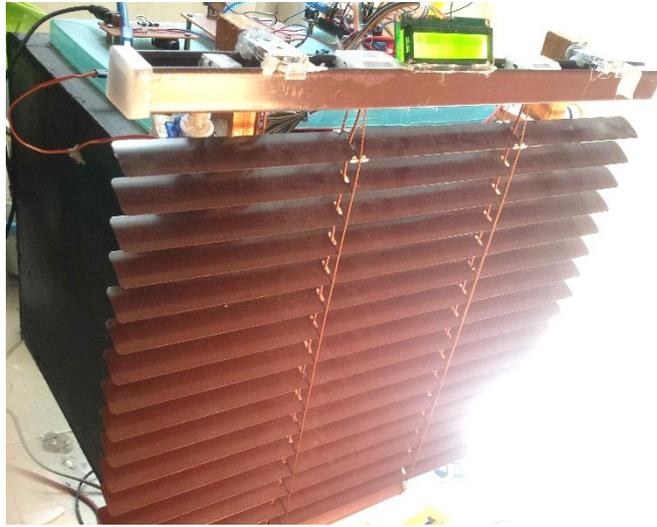
### **5.5.2 Peralatan yang digunakan**

1. Miniatur Ruangan .
2. Sensor Intensitas Cahaya BH1750.
3. *Driver* Motor DC .
4. Sumber Tegangan DC.
5. Arduino UNO 328p.
6. Laptop.

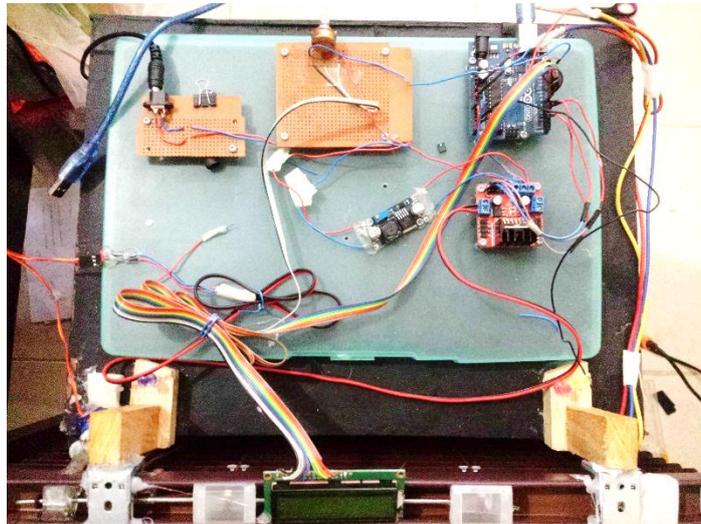
### **5.5.3 Langkah pengujian**

1. Membuat rangkaian Komponen elektrik seperti Gambar 5.5
2. Mengunduh program dengan kontroler P sesuai dengan parameter yang telah didapat dengan Arduino IDE.
3. Memberi *setpoint* dan melihat respon sistemnya.
4. Memberikan gangguan berupa penghalang cahaya yang berasal dari kertas putih dan diberi cahaya tambahan berupa mendekatkan sumber cahaya.
5. Data yang disampling dicatat dan dibuat grafik.

Berikut Gambar 5.4 dan 5.5 seluruh sistem dan *plant* yang akan di uji.



Gambar 5.4 *Plant* Tirai Blind

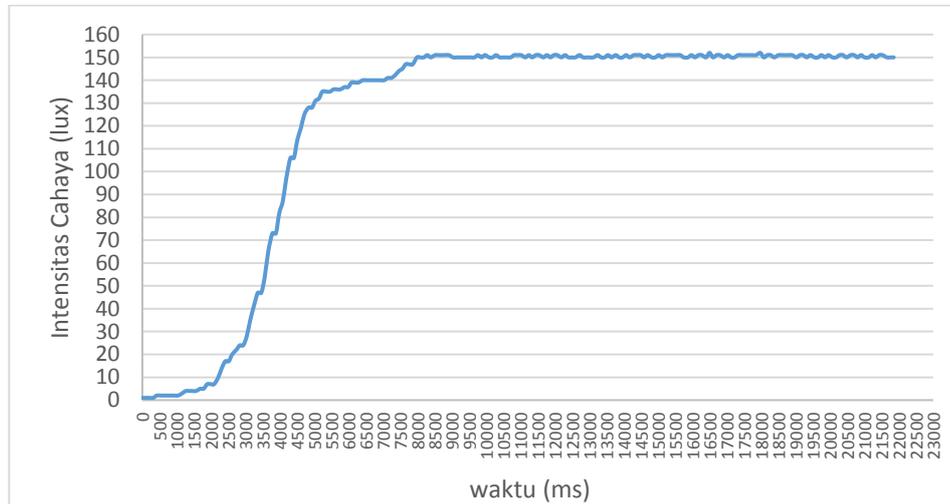


Gambar 5.5 Rangkain Komponen Pada *Plant*

### 5.5.4 Hasil Pengujian

- Pengujian Sistem Tanpa Adanya Gangguan dengan *setpoint* 150 lux.

Didapatkan hasil respon keluaran dengan memberikan *setpoint* 150 lux. Grafik hasil pengujian bisa dilihat pada Gambar 5.6.



Gambar 5.6 Grafik Respon Sistem Keseluruhan dengan *Set Point* 150 Lux.

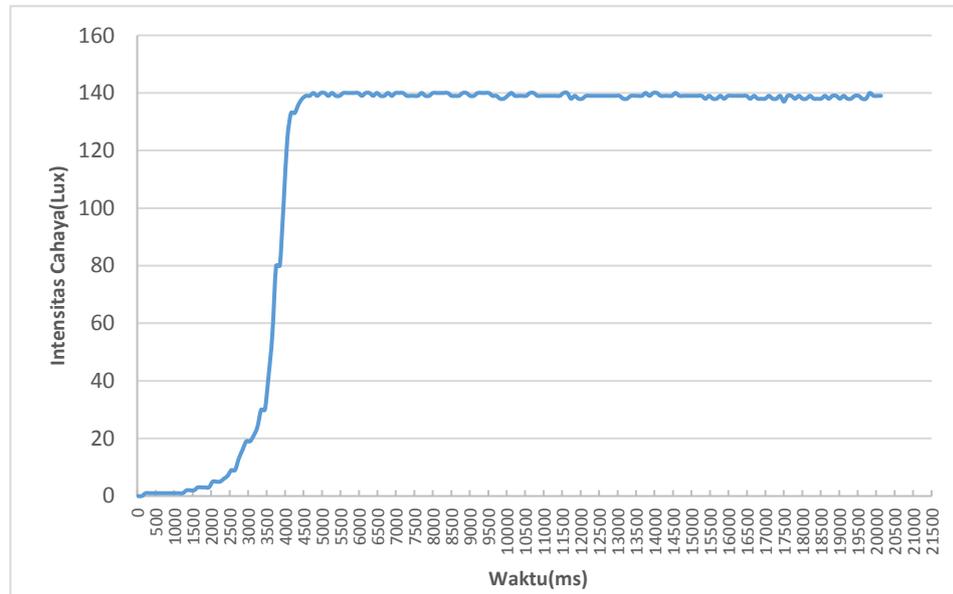
Berdasarkan hasil dan grafik pengujian keseluruhan dengan tanpa gangguan, diperoleh kinerja sistem antara lain:

Kontroler proporsional tidak terdapat *overshoot* dan *settling time* nya 8185 ms (8,1 detik). Mencapai *Settling time* ketika intensitas cahaya bernilai 139 lux. Dari grafik hasil pengujian bisa di hitung *error steady state* nya sebagai berikut:

$$\begin{aligned}
 e_{ss} (\%) &= \left| \frac{(T \text{ tengah steady state}) - (T \text{ setpoint})}{(T \text{ setpoint}) - (T \text{ mula-mula})} \right| \times 100\% \\
 &= \left| \frac{149 - 150}{150 - 1} \right| \times 100\% \\
 &= 0,67\%
 \end{aligned}$$

- Pengujian Sistem Tanpa Adanya Gangguan dengan *setpoint* 140 lux

Setelah melakukan langkah percobaan, didapatkan hasil respon keluaran dengan memberikan *setpoint* 140 lux. Grafik hasil pengujian bisa dilihat pada Gambar 5.7



Gambar 5.7 Grafik Respon Sistem Keseluruhan dengan *Set Point* 140 Lux.

Berdasarkan hasil dan grafik pengujian keseluruhan dengan tanpa gangguan, diperoleh kinerja sistem antara lain:

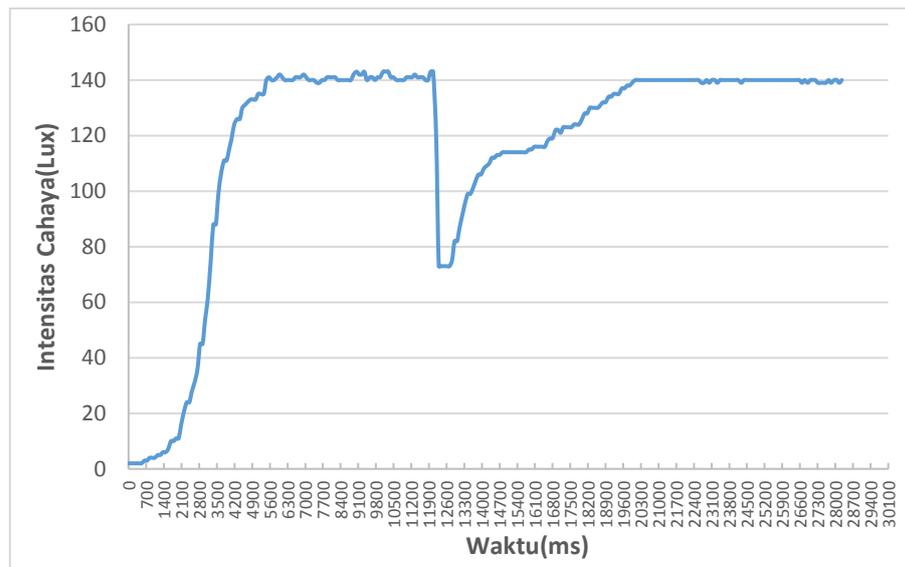
Kontroler proporsional tidak terdapat *overshoot* dan *settling time* nya 4567 ms (4,5 detik). Mencapai *Settling time* ketika intensitas cahaya telah 139 lux. Dari grafik hasil pengujian bisa di hitung *error steady state* nya sebagai berikut:

$$\begin{aligned}
 e_{ss} (\%) &= \left| \frac{(T \text{ tengah steady state}) - (T \text{ setpoint})}{(T \text{ setpoint}) - (T \text{ mula-mula})} \right| \times 100\% \\
 &= \left| \frac{139 - 140}{140 - 1} \right| \times 100\% \\
 &= 0,71\%
 \end{aligned}$$

Dari hasil penghitungan didapatkan *error steady state* sebesar 0.71% dan sistem dikatakan baik karena *error* yang didapatkan masih dalam toleransi 5%.

- Pengujian Sistem Dengan Gangguan

Pengujian ini bertujuan untuk mengetahui bagaimana daya tahan sistem *plant* terhadap gangguan luar. Gangguan yang diberikan berupa gangguan dimana antar sumber cahaya dan tirai *blind* diberi kertas putih dan intensitas cahaya tambahan dengan sumber cahaya di didekatkan ke tirai *blind* tersebut. Berikut pada Gambar 5.8 yaitu grafik gangguan dengan halangan kertas putih dengan *setpoint* 140 lux.

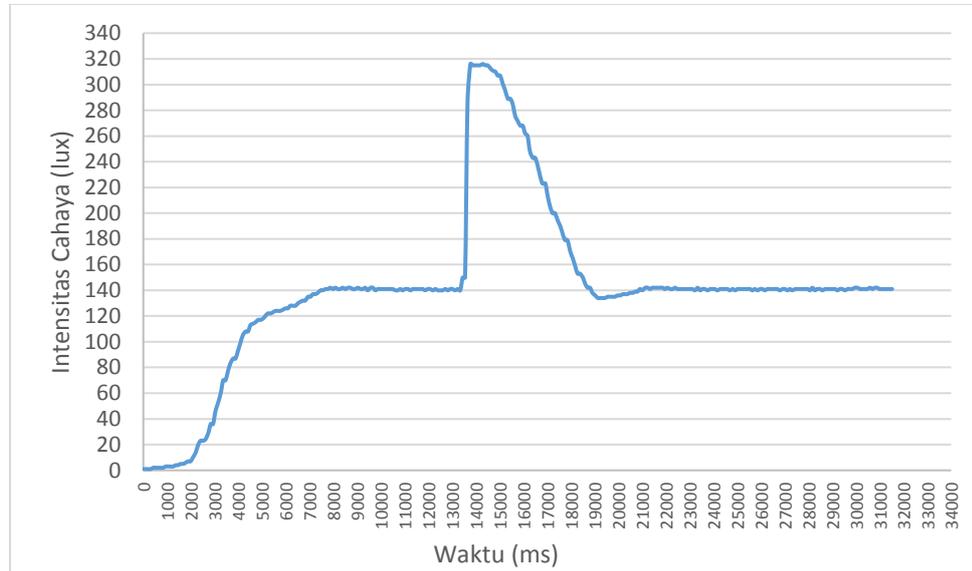


Gambar 5.8 Grafik Respon Sistem dengan Gangguan Kertas Putih

Dari Gambar 5.8 sistem *plant* tidak mengalami *overshoot* dengan *settling time* 5661 ms (5,6 detik). Lalu diberikan gangguan dihalangi kertas putih pada detik ke 11,8 dengan intensitas cahaya turun hingga menyentuh 73 lux dan membutuhkan *recovery time* selama 7.4 detik untuk mencapai *setpoint* lagi.

$$\begin{aligned}
 e_{ss} (\%) &= \left| \frac{(T \text{ tengah steady state}) - (T \text{ setpoint})}{(T \text{ setpoint}) - (T \text{ mula-mula})} \right| \times 100\% \\
 &= \left| \frac{141 - 140}{140 - 73} \right| \times 100\% \\
 &= 1,49\%
 \end{aligned}$$

Berikut pada Gambar 5.9 merupakan grafik respon sistem dengan gangguan tambahan intensitas cahaya pada *set point* 140 lux.



Gambar 5.9 Grafik Respon Sistem dengan Gangguan Tambahan Cahaya

Dari Gambar 5.9 kontroler P tidak ada *overshoot* dengan *settling time* 7236 ms (7.23 detik). Kontroler P saat diberikan gangguan tambahan cahaya pada detik ke 13,3 dengan intensitas cahaya bertambah mencapai 315 lux dan *recovery time* nya selama 6,7 detik. Kemudian intensitas cahaya kembali ke *setpoint*.

$$\begin{aligned}
 e_{ss} (\%) &= \left| \frac{(T \text{ tengah steady state}) - (T \text{ setpoint})}{(T \text{ setpoint}) - (T \text{ mula-mula})} \right| \times 100\% \\
 &= \left| \frac{141 - 140}{140 - 315} \right| \times 100\% \\
 &= 0,57\%
 \end{aligned}$$



## BAB VI PENUTUP

### 6.1. Kesimpulan

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

1. Pengaturan intensitas cahaya menggunakan tirai *blind* dengan motor DC yang tersambung dengan *driver* motor sebagai aktuator, Arduino UNO sebagai kontroler yang berisi logika kontrol proporsional, dan sensor intensitas cahaya BH1750 sebagai *feedback* untuk pembacaan dan masukkan *error*, sehingga didapatkan sistem pengendalian intensitas cahaya otomatis dengan tirai *blind*.
2. Perancangan kontroler proporsional terdiri atas perancangan perangkat keras yaitu merancang modul mikrokontroler beserta *driver* dan modul sensor. Kemudian melakukan perancangan perangkat lunak yang berupa pembuatan program pada software mikrokontroler untuk pembacaan sensor dan aksi *driver* motor dc.
3. Berdasarkan hasil kontroler P metode Ziegler Nichols 1, maka di peroleh:
  - nilai parameter  $K_p=0.7$ .
  - Dari penentuan parameter P dengan *setpoint* 120 lux yang didapatkan *settling time* 8,7 detik menghasilkan respon sistem berada dalam daerah *setpoint* dan mampu diaplikasikan pada sistem pengendalian intensitas cahaya karena *error steady state* yang didapatkan sebesar 0.84%.
  - Pada *setpoint* 150 lux didapatkan *settling time* 8,1 detik dengan *error steady state* yang didapatkan sebesar 0.67%.
  - Pada *setpoint* 140 lux didapatkan *settling time* 4,5 detik dengan *error steady state* yang didapatkan sebesar 0,71%.
  - Uji pengendali P dengan penambahan gangguan berupa kertas putih dan cahaya tambahan. *Recovery time* masing-masing gangguan kertas putih dan cahaya tambahan adalah sebesar 7.4 detik dan 6.7 detik. Kontroler P memiliki respon kerja yang baik dibuktikan dengan respon sistem dapat kembali menuju keadaan *steady state*.

## 6.2. Saran

Dalam Perancangan dan pembuatan alat ini masih terdapat kekurangan. Untuk memperbaiki kinerja alat dan pengembangan lebih lanjut disarankan :

1. Memakai sensor cahaya lebih dari satu dan penempatan sensor nya berada di luar tirai sebagai pembanding.
2. Menggunakan selain motor DC seperti motor stepper atau servo.
3. Menggunakan metode kontrol selain kontrol proporsional.
4. Dibuat agar ada batasan membuka atau menutup nya bilah tirai *Blind* tersebut.

## DAFTAR PUSTAKA

- Anonim. *Data sheet ATmega328*
- Anonim. *Datasheet LCD M1632*
- Anonim. *Datasheet driver l298n*
- Alldatasheet.com. *Datasheet BH1750*
- Alfido.2015.*Sistem Pengaturan Intensitas Cahaya Dengan Perencanaan Kondisi Lingkungan Pada Rumah Kaca*. Skripsi, Fakultas Teknik Universitas Brawijaya,Malang
- Coughlin, Robert F. 1982. *Penguat Operasional dan Rangkaian Terpadu Linear*, Jakarta: Erlangga.
- Johar S. M., Cornelius.2015. *Sistem Pengendali Intensitas Cahaya Pada Tanaman Anggrek Menggunakan Kontroler PID Berbasis Arduino Mega 2560*. Skripsi, Fakultas Teknik Universitas Brawijaya, Malang.
- Kilian, Christopher T, *Modern Control Technology*, (West Publishing Co:1996)
- Malvino, AP. 1987. *Prinsip-prinsip Elektronika, Jilid 2, Cetakan ketiga*, terjemahan Prof. M. Barmawi, Ph.D, M.O Tjia, Ph.D. Jakarta : Erlangga.
- Ogata, K. 1997. *Teknik Kontrol Automatik*. Jakarta: Erlangga.
- Tooley, Mike. 2002. *Rangkaian Elektronik Prinsip dan Aplikasi*. Jakarta. Penerbit Erlangga.



## **LAMPIRAN**

