

**PENGONTROLAN INTENSITAS CAHAYA LAMPU DAN
SUHU RUANGAN OTOMATIS MENGGUNAKAN METODE
PID BERBASIS MIKROKONTROLER**

PROYEK AKHIR

Laporan akhir ini dibuat dan diajukan untuk memenuhi salah satu syarat kelulusan
Diploma III Politeknik Manufaktur Negeri Bangka Belitung



Disusun Oleh:

Raihan Afiruru

NIRM 0032053

Sephia Oevi

NIRM 0032056

**POLITEKNIK MANUFAKTUR NEGERI
BANGKA BELITUNG
TAHUN 2023**

LEMBAR PENGESAHAN

JUDUL PROYEK AKHIR

PENGONTROLAN INTENSITAS CAHAYA LAMPU DAN SUHU RUANGAN OTOMATIS MENGGUNAKAN METODE PID BERBASIS MIKROKONTROLER

Oleh:

Raihan Afiruru

/NIRM 0032053

Sephia Oevi

/NIRM 0032056

Laporan akhir ini telah disetujui dan disahkan sebagai salah satu syarat kelulusan Diploma III Politeknik Manufaktur Negeri Bangka Belitung

Menyetujui,

Pembimbing 1



(Ocsirendi, M.T.)

Pembimbing 2



(Zanu Saputra, M.Tr.T.)

Penguji 1



(Yudhi, M.T.)

Penguji 2



(Dr. Parulian Silalahi, M.Pd.)

PERNYATAAN BUKAN PLAGIAT

Yang bertanda tangan dibawah ini:

Nama Mahasiswa 1: Raihan Afiruru

NIRM: 0032053

Nama Mahasiswa 2: Sephia Ocvi

NIRM: 0032056

Dengan Judu : Pengontrolan Intensitas Cahaya Lampu dan Suhu Ruangan

Otomatis Menggunakan Metode PID Berbasis Mikrokontroler

Menyatakan bahwa laporan akhir ini adalah hasil kerja kami sendiri dan bukan merupakan plagiat. Pernyataan ini kami buat dengan sebenarnya dan bila ternyata dikemudian hari ternyata melanggar pernyataan ini, kami bersedia menerima sanksi yang berlaku.

Sungailiat, 17 Juli 2023

Nama Mahasiswa

Tanda Tangan

1. Raihan Afiruru



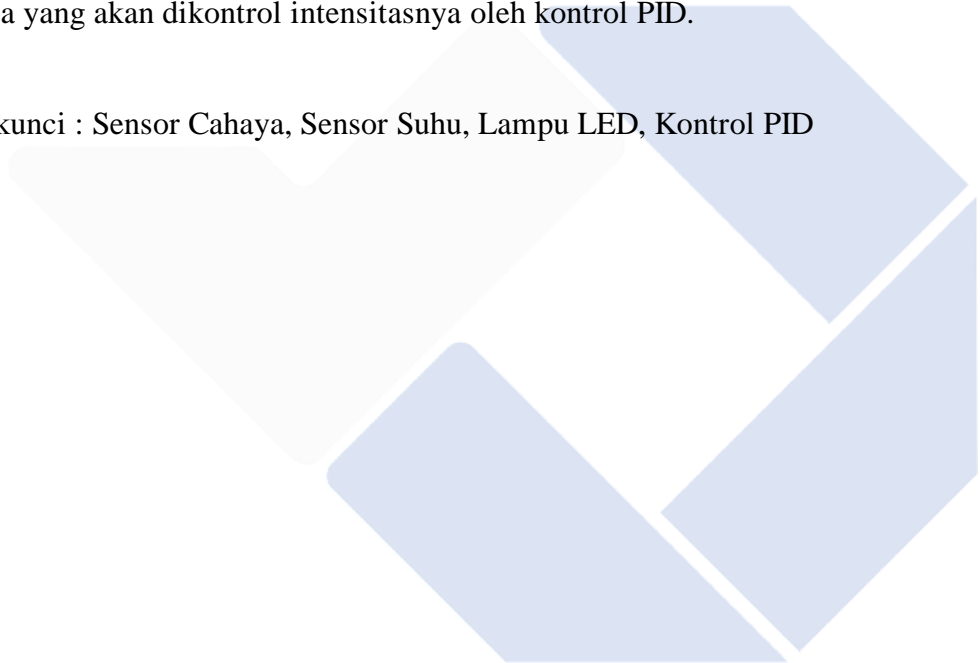
2. Sephia Ocvi



ABSTRAK

Pengendalian intensitas cahaya lampu menggunakan sensor cahaya dan kontrol PID telah menjadi perhatian dalam bidang otomasi. Dalam penelitian ini, kami mengkaji penggunaan sensor cahaya dan sensor suhu, serta kontrol PID untuk mengatur intensitas cahaya pada lampu LED. Sistem yang dirancang terdiri dari sensor cahaya dan sensor suhu, lampu LED, dan kontrol PID. Sensor cahaya digunakan untuk mendeteksi intensitas cahaya sekitar, sedangkan sensor suhu digunakan untuk mendeteksi suhu ruangan. Lampu LED digunakan sebagai sumber cahaya yang akan dikontrol intensitasnya oleh kontrol PID.

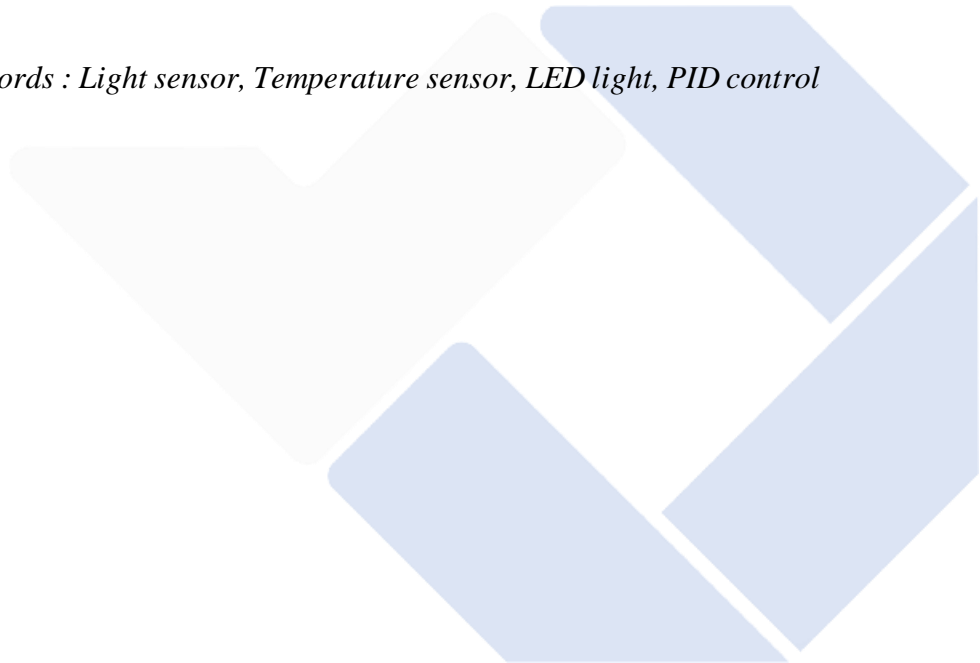
Kata kunci : Sensor Cahaya, Sensor Suhu, Lampu LED, Kontrol PID



ABSTRACT

In the realm of automation, controlling light intensity with light sensors and PID control has become a problem. In this study, we investigate the use of PID controllers, temperature sensors, and light sensors to modify the brightness of LED bulbs. The system is built using temperature and light sensors, LED lights, and PID controllers. The temperature sensor determines the room's temperature, while the light sensor measures the amount of ambient light. The PID control will regulate the brightness of the LED bulb, which is employed as the light source.

Keywords : Light sensor, Temperature sensor, LED light, PID control



KATA PENGANTAR

Segala puji bagi Allah SWT atas nikmat yang sudah diberikan sehingga penulis dapat menyelesaikan laporan serta proyek akhir dengan baik dan tepat pada waktunya sesuai dengan jadwal yang sudah ditetapkan oleh pihak Institusi Politeknik Manufaktur Negeri Bangka Belitung. Shalawat serta salam dicurahkan kepada baginda tercinta Nabi Muhammad SAW agar kita dapat mendapat syafaat di akhirat nanti.

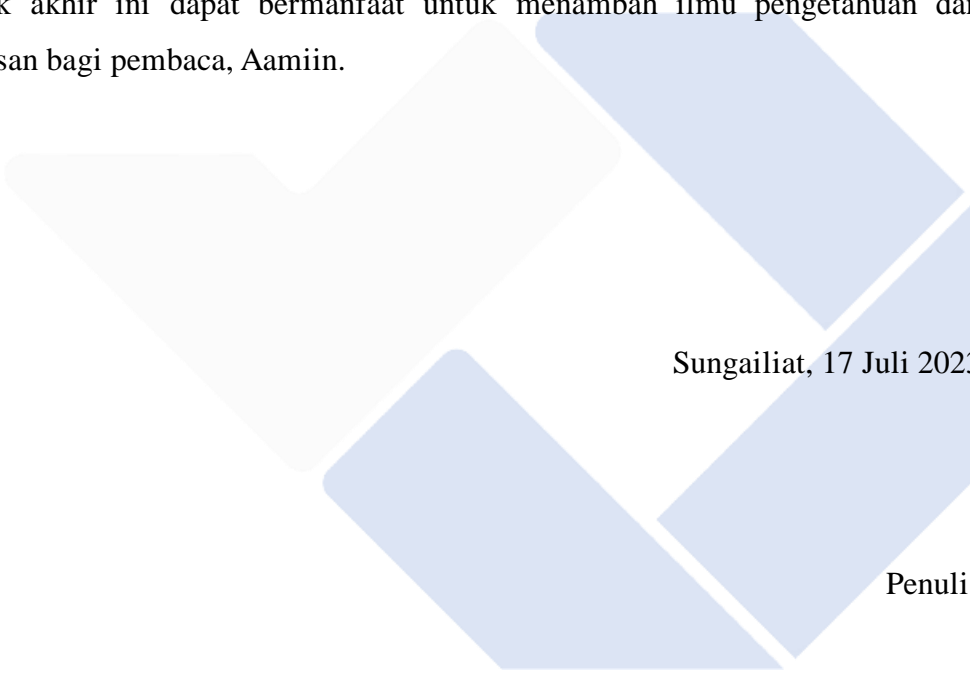
Adapun judul proyek akhir ini adalah “Pengontrolan Intensitas Cahaya Lampu dan Suhu Ruangan Otomatis Menggunakan Metode PID Berbasis Mikrokontroler”. Tujuan dari pembuatan laporan proyek akhir ini yaitu sebagai salah satu persyaratan dan kewajiban mahasiswa untuk menyelesaikan kurikulum program Pendidikan Diploma III di Politeknik Manufaktur Negeri Bangka Belitung.

Dalam pelaksanaan pembuatan proyek akhir ini, penulis ingin menyampaikan rasa terimakasih yang sebesar-besarnya kepada;

1. Orangtua dan keluarga penulis yang selalu memberikan kasih sayang, untaian doa, dukungan, serta moril maupun materil dan semangat.
2. Bapak I Made Andik Setiawan, M.Eng, Ph.D, selaku Direktur Politeknik Manufaktur Negeri Bangka Belitung.
3. Bapak Zanu Saputra, M.Tr.T, selaku Ketua Jurusan Teknik Elektro dan Informatika di Politeknik Manufaktur Negeri Bangka Belitung, Dosen Wali, serta Dosen Pembimbing II yang telah membantu memberi arahan dan bimbingan dalam pembuatan alat dan penulisan laporan proyek akhir di Politeknik Manufaktur Negeri Bangka Belitung.
4. Bapak Ocsirendi, M.T, selaku Kepala Program Studi Diploma III Teknik Elektronika serta Dosen Pembimbing I yang telah membantu memberikan arahan dan bimbingan dalam penulisan serta pembuatan alat pada penyelesaian proyek akhir di Politeknik Manufaktur Negeri Bangka Belitung.

5. Seluruh Dosen, Instruktur dan Staff pengajar Jurusan Teknik Elektro dan Informatika di Politeknik Manufaktur Negeri Bangka Belitung yang telah mrngajarkan banyak ilmu pengetahuan.
6. Pihak-pihak lain yang telah memberikan bantuan secara langsung maupun tidak Langsung dalam pembuatan proyek akhir.

Dalam penyusunan proyek akhir, penulis menyadari bahwa laporan proyek akhir ini masih jauh dari kata sempurna dan juga terdapat banyak kesalahan serta kekurangan di dalamnya karena keterbatasan pengetahuan dan ilmu. Untuk itu, penulis mengharapkan kritik serta saran yang membangun dari pembaca, agar dapat menunjang pengembangan dan perbaikan penulis selanjutnya. Akhir kata, semoga proyek akhir ini dapat bermanfaat untuk menambah ilmu pengetahuan dan wawasan bagi pembaca, Aamiin.



Sungailiat, 17 Juli 2023

Penulis

DAFTAR ISI

LEMBAR PENGESAHAN	ii
PERNYATAAN BUKAN PLAGIAT	iii
ABSTRAK	iv
<i>ABSTRACT</i>	v
KATA PENGANTAR	vi
DAFTAR ISI	viii
DAFTAR TABEL	x
DAFTAR GAMBAR	xi
DAFTAR LAMPIRAN	xii
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah.....	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Proyek Akhir	3
BAB II DASAR TEORI	4
2.1 Proses Kontrol	4
2.2 Kendali PID	4
2.3 Intensitas Cahaya.....	5
2.4 Pengaturan Set Point	6
2.5 Pengaruh suhu ruangan terhadap kipas.....	6
BAB III METODE PELAKSANAAN	7
3.1 Perancangan Model Konstruksi Maket Miniatur	7
3.2 Rancang Hardware	7
3.2.1 Skema dan Pembuatan Prototype Sistem Kendali	7
3.2.2 Blok Diagram Sistem Kendali	8
3.3 Blok Diagram PID	9
3.4 Perancangan Sistem Kontrol.....	9
3.5 Analisis dan Pembuatan Laporan Akhir.....	12
BAB IV PEMBAHASAN.....	13
4.1 Model Konstruksi Maket Miniatur	13
4.1.1 Desain Prototype Maket Miniatur	13
4.2 Pembuatan Hardware	14

4.3 Pengujian Keseluruhan Alat.....	15
4.3.1 Pengujian Alat yang Mengatur Intensitas Cahaya Lampu Otomatis..	15
4.3.2 Pengujian Potensiometer (SV).....	15
4.3.3 Pengujian MOSFET (driver) dan Lampu (Aktuator).....	17
4.3.4 Pengujian Sensor LDR (Feedback).....	21
4.3.5 Pengujian pengaruh beban terhadap sensor	22
4.4 Pengujian Sensor Suhu pada Kipas Angin.....	24
4.4.1 Pengujian Suhu Terhadap Ruang	24
4.4.2 Pengujian Sensor LM35	26
BAB V KESIMPULAN	29
5.1 Kesimpulan.....	29
5.2 Saran	29



DAFTAR TABEL

Tabel 4. 1 Program Pengujian Potensiometer	16
Tabel 4. 2 Pengujian Potensiometer	17
Tabel 4. 3 Program Pengujian MOSFET.....	17
Tabel 4. 4 Hasil pengujian MOSFET	18
Tabel 4. 5 Program Pengujian Sensor LDR.....	22
Tabel 4. 6 Nilai Maksimum Ketika diberi Objek.....	23
Tabel 4. 7 Nilai Minimum Ketika diberi Objek.....	23
Tabel 4. 8 Program Pengujian pada Kipas Angin.....	24
Tabel 4. 9 Pengujian sensor LM35	26
Tabel 4. 10 Hasil dari Pengujian Kipas Angin	28

DAFTAR GAMBAR

Gambar 2. 1 Diagram Blok Kontrol PID Idea Bentuk Dependent	5
Gambar 2. 2 Blok Diagram Kontrol PID Idea Bentuk Independent	5
Gambar 3. 1 Rangkaian Skematik Lampu	8
Gambar 3. 2 Blok Diagram Sistem Kendali Intensitas Cahaya Lampu	8
Gambar 3. 3 Blok Diagram PID	9
Gambar 3. 4 Flowchart Kerja Sistem.....	10
Gambar 3. 5 Blok Diagram Sistem Kendala Suhu Ruangan.....	11
Gambar 3. 6 Blok Diagram PID pada suhu ruangan kipas DC	12
Gambar 4. 1 Gambar Design Prototype	14
Gambar 4. 2 Tahapan Pembuatan Hardware.....	14
Gambar 4. 3 Pengujian Potensiometer (SV).....	15
Gambar 4. 4 Hasil Plotter dari Potensiometer.....	16
Gambar 4. 5 Pengujian MOSFET (driver) dan lampu (aktuator).....	18
Gambar 4. 6 Pengujian dari Hasil MOSFET dan Lampu Menyala Terang (1V)	19
Gambar 4. 7 Pengujian dari Hasil MOSFET dan Lampu Menyala Redup (2V).....	19
Gambar 4. 8 Pengujian Hasil MOSFET Menyala Redup Sedang (3V).....	20
Gambar 4. 9 Pengujian dari Hasil MOSFET dan Lampu Menyala Redup (4V).....	20
Gambar 4. 10 Pengujian dari Hasil MOSFET dan Lampu (5V).....	20
Gambar 4. 11 Pengujian Sensor LDR.....	21
Gambar 4. 12 Hasil Plotter Pengujian Sensor LDR.....	21
Gambar 4. 13 Tampilan LCD SV dan PV	22
Gambar 4. 14 Pengujian Suhu pada Kipas Angin.....	24
Gambar 4. 15 Pengujian Sensor LM35	26
Gambar 4. 16 Tampilan Sensor Suhu LM35 Mendeteksi Panas dari Korek Api	28

DAFTAR LAMPIRAN

Lampiran 1: Daftar Riwayat Hidup

Lampiran 2: Program Pengujian Keseluruhan





BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Salah satu komponen penting dalam kehidupan adalah pencahayaan. Dengan teknologi yang berkembang lebih cepat dari generasi ke generasi, banyak kemudahan yang telah diciptakan. Salah satu contoh teknologi adalah lampu rumah yang dapat diatur dan dapat dinyalakan atau dimatikan dan suhu ruangan secara otomatis [1]. Pengendalian secara otomatis tersebut akan lebih efisiensi dan efektivitas kerja suatu sistem.

Umumnya, pada perangkat listrik seperti lampu sebuah ruangan masih dikendalikan secara manual oleh seseorang yang harus menyalakan dan mematikan sakelar yang terhubung langsung ke lampu. Terkadang ada beberapa lampu yang dijumpai masih hidup ketika tidak digunakan, hal ini dapat disebabkan oleh kelalaian pengguna yang tidak mematikan lampu. Jika jumlah lampu yang berada di dalam rumah cukup banyak, maka sangat tidak efektif dan tidak nyaman untuk menyalakan dan mematikan lampu secara manual. Pengendalian secara manual yang dikendalikan oleh manusia yang bertindak sebagai operator, sementara pengendalian otomatis oleh alat yang telah diatur dapat beroperasi secara otomatis tetapi biasanya masih dibawah pengawasan manusia [2].

Menurut Badan Standarisasi Nasional SNI 03-6575-2001, tingkat pencahayaan untuk rumah tinggal terutama pada ruang kerja, ruang tamu, ruang makan dan kamar tidur berkisar antara 120-250 lux [3]. Manusia melakukan berbagai tugas sehari-hari dalam suatu ruangan, masing-masing membutuhkan tingkat intensitas yang berbeda. Oleh karena itu, diperlukan suatu sistem control intensitas cahaya untuk menyediakan intensitas cahaya ruangan sesuai kebutuhan, sesuai dengan pedoman SNI dan dapat menjaga intensitas pencahayaan sesuai dengan standar meskipun intervensi cahaya dari luar bervariasi [3]. Pencahayaan di bawah standar berkontribusi pada penurunan produktivitas dan tumbuhan yang

membutuhkan pencahayaan khusus. Kondisi tersebut dapat diminimalisir dengan mengatur intensitas cahaya secara otomatis sesuai kebutuhan [4].

Metode kontrol PID (*Proportional Integral Derivative*) merupakan salah satu jenis sistem loop tertutup dengan kemampuan menggunakan umpan balik untuk menilai tingkat akurasi sistem instrumentasi. PID terdiri dari tiga sistem kontrol kendali yaitu Proporsional (P), Integral (I), dan Derivatif (D). Dengan memodifikasi masing-masing parameter P, I, dan D agar sesuai dengan sinyal atau rencana input yang diinginkan dan *setpoint* yang dicapai. Oleh karena itu, PID dapat memberikan respon dengan tingkat kestabilan yang tinggi, mempercepat reaksi sistem seperti menghilangkan *offset* dan membatasi *overshoot* [5].

Dengan menggunakan kontrol PID dan bantuan mikrokontroler, khususnya Arduino Uno, memungkinkan untuk meningkatkan kinerja sistem kontrol intensitas cahaya. Para peneliti juga sering menggunakan papan mikrokontroler Arduino Uno yang didasarkan pada Atmega328 untuk membangun perangkat listrik dengan cepat dan handal yang beragam mulai dari yang sederhana hingga yang rumit [6].

Dari latar belakang diatas, sistem pengendalian intensitas cahaya dimana pada proyek ini akan menggunakan sensor LDR (*Light Dependent Resistor*) dan sensor suhu LM35 sebagai feedback yang nantinya menggunakan metode PID. Fungsi LDR itu sendiri sebagai penerima cahaya yang akan dikontrol melalui mikrokontroler dan akan distabilkan menggunakan metode PID. Sistem kendali intensitas mengatur cahaya yang dipancarkan oleh lampu dengan sensor cahaya yang menyesuaikan dengan kondisi dalam atau luar ruangan. Berkaitan dengan penjabaran diatas, maka akan melakukan penelitian mengenai “Pengontrolan Intensitas Cahaya Lampu dan Sensor Suhu Otomatis Menggunakan Metode PID Berbasis Mikrokontroler” dengan judul ini, penulis berharap semoga laporan proyek akhir bisa memberikan manfaat bagi para pembaca.

1.1 Rumusan Masalah

Pada proses penyusunan proyek akhir akan ada beberapa permasalahan yang harus diselesaikan. Berikut rumusan masalah yang diangkat berdasarkan latar belakang proyek akhir diantaranya:

1. Bagaimana merancang sistem pengontrolan intensitas cahaya lampu otomatis menggunakan metode PID berbasis mikrokontroler?
2. Bagaimana menguji sistem pengontrolan intensitas cahaya lampu dan suhu ruangan otomatis untuk memastikan kinerja yang baik dan kestabilan sistem?
3. Bagaimana mengoptimalkan sistem pengontrolan intensitas cahaya lampu dan suhu ruangan otomatis agar sesuai dengan kebutuhan dan preferensi pengguna?

1.2 Batasan Masalah

Adapun batasan masalah yang mengacu pada latar belakang proyek akhir diantaranya:

1. Pembuatan sistem pengontrolan intensitas cahaya lampu dan suhu ruangan otomatis pada proyek akhir ini diterapkan dalam bentuk maket miniatur rumah.
2. Sistem pengontrolan ini menggunakan sensor LDR dan sensor LM35.
3. Sistem pengontrolan intensitas cahaya lampu otomatis bekerja pada 2 perangkat elektronik, yaitu 1 buah lampu dan 1 buah kipas angin mini.

1.3 Tujuan Proyek Akhir

Tujuan yang akan disampaikan melalui proyek akhir ini yaitu:

1. Merancang alat pengontrolan intensitas cahaya lampu dan suhu ruangan otomatis pada kebutuhan rumah tangga.
2. Menjelaskan hasil pengujian sistem pengontrolan intensitas cahaya lampu dan suhu ruangan otomatis pada perangkat elektronik.
3. Menjelaskan optimalisasi sistem pengontrolan intensitas cahaya lampu dan suhu ruangan otomatis.

BAB II

DASAR TEORI

2.1 Proses Kontrol

Sistem kontrol merupakan suatu sistem yang dapat mengukur nilai variabel dan menerapkan variabel yang di modifikasi ke dalam sistem untuk mengurangi penyimpangan dari nilai yang ditentukan. Sistem kontrol berfungsi sebagai kontroler atau pengontrol untuk suatu proses yang memungkinkannya terjadi secara manual atau otomatis tergantung ada situasinya. Komponen-komponen dari sistem otomatis adalah *controler, final control element plant, feedback* dan *range* [7].

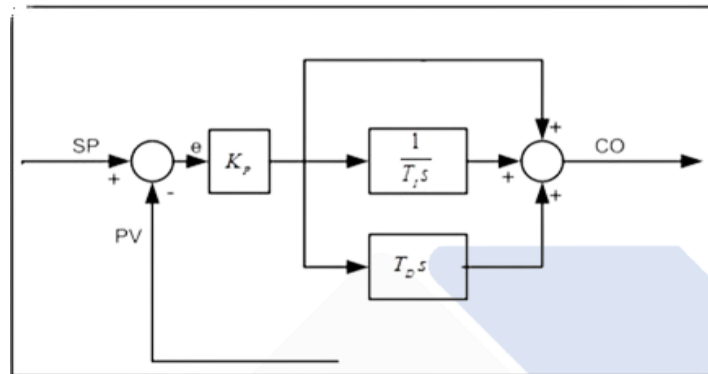
Model maket miniatur digunakan untuk membuat *prototype* kontrol intensitas cahaya otomatis yang dalam bentuk maket miniatur tersebut ada dua perangkat elektronik yaitu 1 buah lampu dan 1 buah kipas angin mini. Perangkat ini diprogram untuk digunakan sebagai perangkat yang dikontrol oleh pengontrol intensitas utama. Sistem kontrol intensitas cahaya ini menggunakan mikrokontroler Arduino Uno. Penulis berharap dapat mendorong individu untuk mengadopsi gaya hidup yang dapat memfasilitasi dan mengatur semua aktivitas dengan bantuan upaya terakhir ini.

2.2 Kendali PID

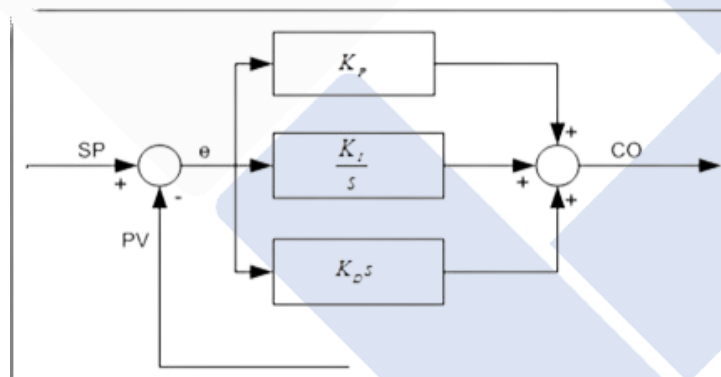
Sistem kendali *on* atau *off* dan sistem kendali otomatis adalah dua jenis kategori dalam suatu sistem kendali. Dalam tiga tindakan kendali dapat membentuk sistem kendal otomatis yaitu ; pengontrol Proporsional (P), pengontrol Integral (I), dan pengontrol Derivatif (D). Tindakan kontrol PID dapat menggabungkan kontrol P, I, dan D dalam upaya untuk memaksimalkan manfaat dari ketiga komponen dasar PID :

1. Kontrol Proporsional : berfungsi untuk mempercepat reaksi,
2. Kontrol Integral : berfungsi untuk menghilangkan kesalahan tetap (*error steady*)
3. Kontrol Derivatif : berfungsi untuk meningkatkan dan mempercepat reaksi transien.

PID ideal dibagi menjadi PID dependent (Gambar 2.1) dan independen (Gambar 2.2). Bentuk dependen tidak bergantung pada nilai konstanta proporsional (K_p), sedangkan bentuk independen pada nilai (K_p). Sehingga pada saat melakukan perubahan K_p tidak mempengaruhi konstanta parameter lainnya.



Gambar 2. 1 Diagram Blok Kontrol PID Idea Bentuk Dependent



Gambar 2. 2 Blok Diagram Kontrol PID Idea Bentuk Independent

Keluaran pengontrol PID merupakan penjumlahan dari keluaran pengontrol proporsional, integral, dan derivatif.

2.3 Intensitas Cahaya

Kuantitas fluks cahaya yang memancar dari sudut ruang dikenal sebagai intensitas pencahayaan (cd). Sedangkan besaran fluks cahaya yang mengenai

satuan luas permukaan yang terkena sinar cahaya dimaksud dengan istilah “penerapan” atau “iluminasi” lux yang kuat [8].

2.4 Pengaturan Set Point

Nilai set point merupakan target yang akan dikejar atau nilai kontrol yang diinginkan. Jika output yang terdeteksi oleh alat ukur tidak sama dengan nilai set point maka akan terjadi error, seperti yang dihasilkan dari ketidaksesuaian antara sinyal umpan dan set point [9].

2.5 Pengaruh suhu ruangan terhadap kipas

Suhu merupakan parameter yang mengukur tingkat panas atau dinginnya suatu objek atau sistem. Secara umum, suhu dapat didefinisikan sebagai besaran fisik yang mencerminkan tingkat energi termal yang dimiliki oleh benda atau sistem. Suhu dapat diukur dan dibandingkan antara dua objek atau lebih yang berada dalam kesetimbangan termal [10].

BAB III

METODE PELAKSANAAN

Metode pelaksanaan merupakan bagian dari suatu tahapan yang dilakukan untuk melakukan proses pelaksanaan proyek akhir yang berjudul “Pengontrolan Intensitas Cahaya Lampu dan Suhu Ruang Otomatis Menggunakan Metode PID Berbasis Mikrokontroler” yang digunakan untuk pengerjaan proyek akhir dan dapat mempermudah dalam proses pengerjaan proyek akhir.

3.1 Perancangan Model Konstruksi Maket Miniatur

Langkah pertama dalam mengatur pembuatan alat yang akan dibuat adalah merancang model bangunan maket kecil untuk menyesuaikan intensitas cahaya. Aplikasi yang digunakan dalam perancangan model konstruksi dari maket miniatur dibuat dengan menggunakan aplikasi Sketch Up. Rancangan konstruksi proyek akhir ini dibuat sebagai prototype yang dilengkapi dengan perangkat untuk rangkaian listrik berupa perangkat lampu dan kipas angin.

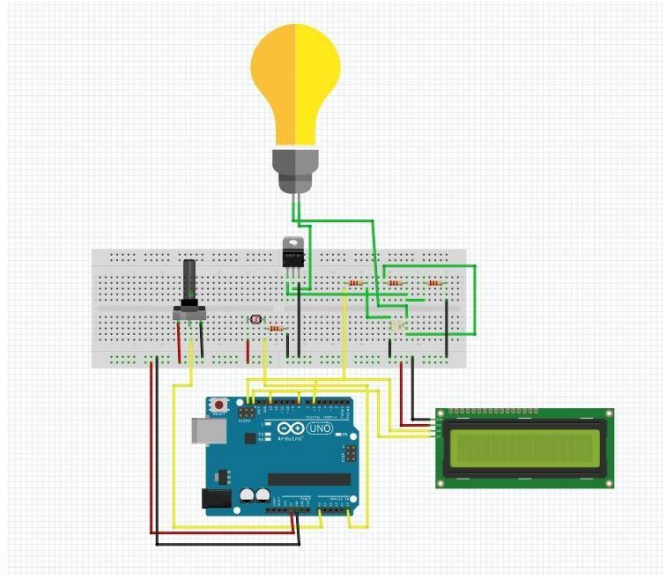
3.2 Rancang Hardware

Rancang hardware merupakan sistem kendali perangkat lampu yang dilakukan untuk memaparkan konsep hardware perangkat lampu yang memiliki bentuk fisik berupa prototype sistem penerangan yang mengatur intensitas cahaya lampu yang dikendalikan secara otomatis.

3.2.1 Skema dan Pembuatan Prototype Sistem Kendali

Berikut skema rangkaian sistem kendali secara keseluruhan dan komponen-komponen yang digunakan dalam proyek akhir ditunjukkan pada gambar 3.1 di

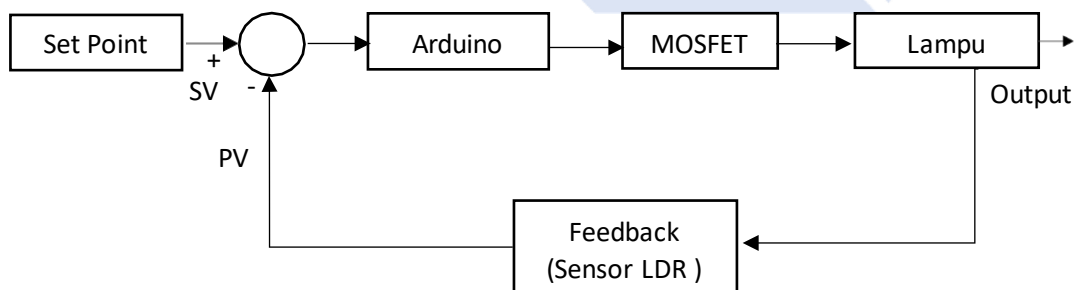
bawah ini akan menampilkan diagram sirkuit yang dihasilkan oleh aplikasi Fritzing pada judul proyek akhir ini.



Gambar 3. 1 Rangkaian Skematik Lampu

3.2.2 Blok Diagram Sistem Kendali Cahaya Lampu

Sistem kendali yang mengatur intensitas cahaya dengan PID berbasis Arduino, MOSFET, dan sensor LDR merupakan alat yang memiliki kemampuan untuk mengendalikan intensitas cahaya menggunakan MOSFET sebagai driver dan sensor LDR sebagai sensor feedback. Berikut adalah gambar 3.2 blok diagram dari sistem kendali yang dibuat.



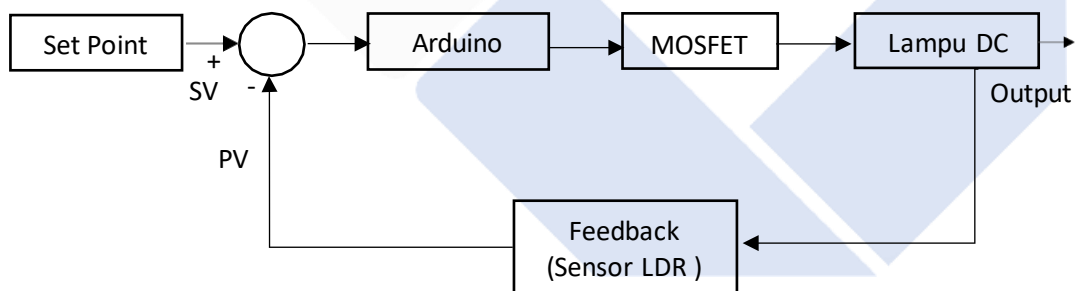
Gambar 3. 2 Blok Diagram Sistem Kendali Intensitas Cahaya Lampu

Set point digambarkan dalam diagram blok grafik diatas menunjukkan sebagai ambang batas sistem yang biasa disebut dengan Set Value (SV). Kemudian,

kontroller Arduino berfungsi sebagai pengolah sinyal untuk memastikan bahwa keluaran sinyal harus sesuai dengan yang diharapkan. Setelah keluar dari Arduino, sinyal tersebut beralih ke plant driver (MOSFET) dan lampu DC. Lalu lampu akan terus bertambah atau bahkan bisa berkurang intensitas cahaya sesuai dengan sinyal yang akan keluar dari arduino. Dan gangguan yang diberikan pada aktuator berupa penggelapan atau penerangan tambahan pada sistem plant. Selanjutnya umpan balik dari sensor LDR memilih intensitas cahaya yang dihasilkan agar sesuai dengan input yang diinginkan disebut sebagai Present Value (PV).

3.3 Blok Diagram PID pada Cahaya Lampu

Sumber 220V yang di gunakan untuk sumber pada MOSFET untuk menghidupkan lampu DC kemudian input LDR di gunakan untuk mengukur resistansi dari cahaya matahari yang dimana nantinya akan di kontrol melalui modul mikrokontroler Arduino dengan menggunakan metode PID untuk menurunkan atau mengatur intensitas cahaya pada lampu. Dimana cahaya lampu akan di tentukan dari hasil resistansi LDR. Berikut adalah gambar 3.3 blok diagram PID dibawah ini.

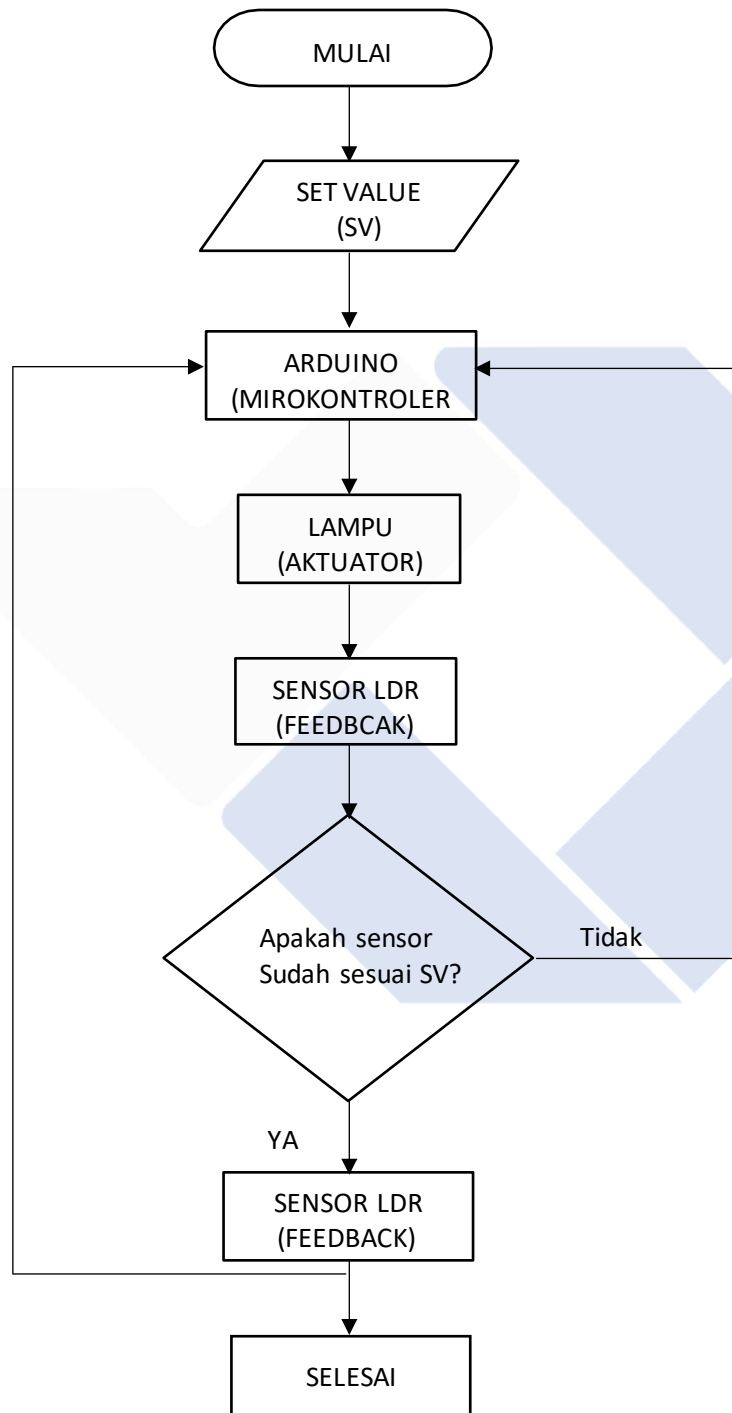


Gambar 3. 3 Blok Diagram PID

3.4 Perancangan Sistem Kontrol pada Lampu

Pada tahap ini, pengerjaan program yang dibuat pada aplikasi Arduino Uno IDE yang memiliki fungsi untuk mengembangkan sistem kerja proyek akhir. Program diunggah dengan bertindak sebagai kode basis data dan instruksi untuk sirkuit sistem yang kemudian dapat memberi sinyal untuk mengontrol intensitas cahaya lampu yang ada pada maket rumah tersebut.

Penulis juga telah membuat flowchart perancangan software, gunanya untuk mempermudah para pembaca. Berikut gambar 3.4 flowchart yang telah dibuat oleh penulis di bawah ini.

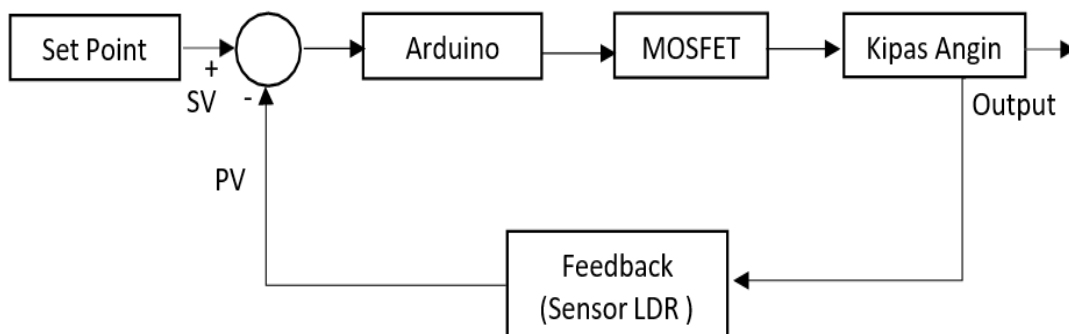


Gambar 3. 4 Flowchart Kerja Sistem

Berdasarkan flowchart diatas pada gambar 3.4 menunjukkan alur kerja sistem pada proyek akhir ini, yang diawali dengan set value (sv) agar dapat berfungsi sebagaimana mestinya, kemudian bisa melanjutkan untuk mengontrol intensitas cahaya lampu sesuai dengan situasi dan kondisi.

3.5 Blok Diagram Sistem Kendala Suhu Ruangan

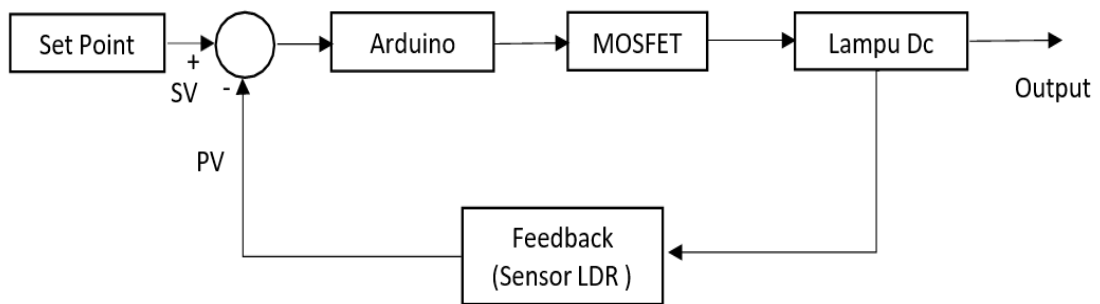
Sumber 12V yang digunakan untuk MOSFET untuk menyalakan kipas DC, kemudian input sensor LM35 untuk mengukur suhu sekitar atau suhu ruangan yang dikontrol menggunakan arduino sebagai kepala perangkat mikrokontroler dengan metode PID sebagai untuk menstabilkan rangkaian sehingga dapat sesuai dengan sistem kendali yang dibuat. Berikut adalah gambar 3.5 blok diagram dari sistem kendali suhu ruangan.



Gambar 3.5 Blok Diagram Sistem Kendala Suhu Ruangan

Gambar 3.5 diagram diatas menunjukkan sistem kerja semestinya, dengan dimulai mengatur set point (nilai yang diinginkan) kemudian dapat menghasilkan keluaran yang diolah oleh arduino, lalu sinyal akan distabilkan oleh MOSFET dan berakhir menuju kipas DC. Gangguan dari luar berupa suhu panas atau dingin menjadi tambahan planning kerja sistem. Kemudian *feedback* dari suhu LM35 akan menyesuaikan nilai tangkapan suhu yang kemudian menghasilkan *present value* (PV).

Dengan adanya umpan balik dari sensor suhu LM35, sistem akan mampu mengatasi fluktuasi suhu eksternal dengan lebih efektif. Ini akan menghasilkan pengaturan yang lebih akurat dan kinerja yang lebih konsisten dalam mencapai tujuan yang telah ditentukan. Setelah selesai mengontrol suhu ruangan pada kipas DC, Berikut gambar 3.6 blok diagram PID dibawah ini.



Gambar 3. 6 Blok Diagram PID pada suhu ruangan kipas DC

3.6 Analisis dan Pembuatan Laporan Akhir

Pada tahap ini semua hasil yang telah didapatkan dari tahapan berikutnya diolah dan dilakukan analisa. Analisa untuk data yang digunakan pada pembuatan proyek akhir dengan judul “Pengontrolan Intensitas Cahaya Lampu dan Suhu Ruangan Otomatis Menggunakan Metode PID Berbasis Mikrokontroler” dengan pengontrol utamanya menggunakan sensor LDR yang kemudian mengontrol tingkat kecerahan lampu ruangan sesuai situasi dan kondisi. Dengan adanya analisa ini bertujuan untuk melihat data hasil pembuatan proyek dan melihat kekurangan yang ada pada alat yang dibuat.

BAB IV

PEMBAHASAN

Pada bab ini akan membahas tentang proses pembuatan dan pengujian proyek akhir yang berjudul “Pengontrolan Intensitas Cahaya Lampu dan Suhu Ruangan Otomatis Menggunakan Metode PID Berbasis Mikrokontroler”.

4.1 Model Konstruksi Maket Miniatur

Model pengontrolan intensitas cahaya adalah representasi skala kecil dari sebuah maket rumah yang terbuat dari triplek yang dindingnya dicat dengan warna kuning keemasan. Di dalam maket rumah terdapat box berbentuk kotak persegi panjang yang digunakan sebagai kotak mikrokontroler atau tempat menyimpan seperti Arduino, Potensiometer, Mosfet, serta sensor LDR. Maket rumah ini dibuat menjadi dua ruang, untuk ruangan yang pertama untuk penempatan pengontrolan intensitas cahaya lampu otomatis dan ruangan kedua untuk penempatan pengontrolan suhu dari kipas angin. Ruangan pertama maket rumah berukuran 50 cm x 50 cm dan ruangan kedua berukuran 30 cm x 30 cm.

Pengontrolan intensitas cahaya lampu otomatis ini menggunakan perangkat yang mempermudah pengguna dalam mengontrol pemakaian lampu dan kipas angin. Untuk alat kontrol utama yang digunakan pada lampu adalah sensor LDR dan kontrol utama pada kipas angin adalah sensor suhu LM35.

4.1.1 Desain Prototype Maket Miniatur

Desain alat merupakan suatu proses awal dalam melakukan perencanaan pembuatan alat yang akan dibuat. Bersamaan dengan perancangan alat, desain alat pada proyek akhir ini di desain oleh penulis menggunakan aplikasi *Sketch Up*,

berikut gambar 4.1 desain alat pada proyek akhir yang dapat dilihat pada gambar dibawah ini.



Gambar 4. 1 Gambar Design Prototype

4.2 Pembuatan Hardware

Setelah alat beroperasi dengan benar, langkah terakhir adalah merakit semua komponen menjadi satu entitas fisik yang berbentuk maket miniatur. Berikut adalah gambar 4.2 yang menunjukkan tahapan pembuatan hardware di bawah ini.



Gambar 4. 2 Tahapan Pembuatan Hardware

4.3 Pengujian Keseluruhan Alat

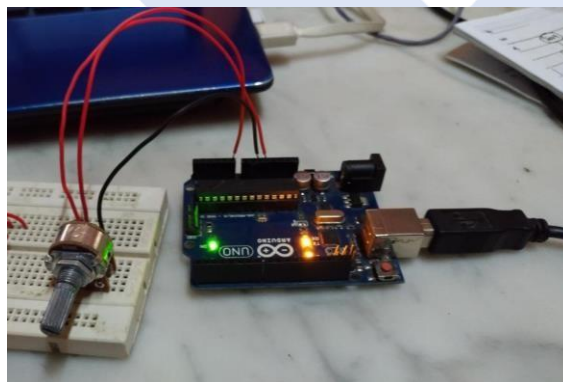
Pengujian yang dimaksudkan adalah melakukan pengujian untuk mengetahui dengan seksama kerja dari keseluruhan alat tersebut dapat mengontrol intensitas cahaya tersebut dalam situasi dan kondisi tertentu dengan hasil nilai yang sesuai.

4.3.1 Pengujian Alat yang Mengatur Intensitas Cahaya Lampu Otomatis

Dalam setiap proses pengujian dilakukan percobaan terhadap suatu komponen yang digunakan sebagai alat kendali, seperti pengujian set point (SV) yang dapat mengubah keadaan keluarannya tegangan. Kemudian dilakukan pengujian MOSFET (driver) dan lampu (aktuator). Hasil dari setiap pengujian yang telah dilakukan selanjutnya akan digunakan sebagai dasar analisis pada bagian selanjutnya.

4.3.2 Pengujian Potensiometer (SV)

Pengujian set point menggunakan potensiometer yang dilakukan dengan menghubungkan kaki tengah potensiometer ke pin A0 arduino dan untuk salah satu kakinya dihubungkan ke ground dan kaki lainnya ke 5V. Berikut gambar 4.3 program saat pengujian pada potensiometer (SV).



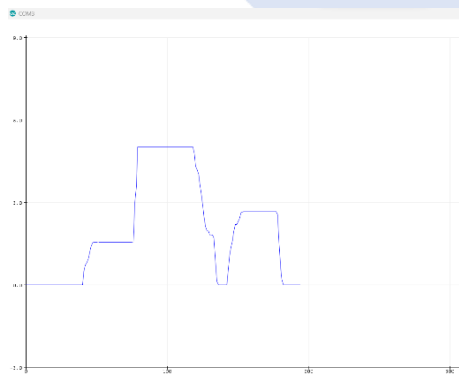
Gambar 4. 3 Pengujian Potensiometer (SV)

Dibawah ini menunjukkan gambar 4.1 program pengujian pada potensiometer

Tabel 4.1 Program Pengujian Potensiometer

Program Pengujian Potensiometer
<pre>float a; void setup () { Serial.begin (9600) ; } void loop () a=analogRead(0)*0.0049 Serial.print (“ ”); Serial.println (a); Delay (100) ; }</pre>

Setelah dilakukan pengujian pada potensiometer, berikut adalah gambar 4.4 yang menunjukkan hasil plotter dari potensiometer.



Gambar 4. 4 Hasil Plotter dari Potensiometer

Pengujian potensiometer sebagai SV (set point) dilakukan dengan membuat rangkaian seperti gambar diatas. Dengan menggunakan program untuk

mengkonversi atau mengubah nilai pembacaan potensiometer menjadi nilai tegangan. Hasil dari pengujian komponen ini dapat dilihat pada tabel 4.2 dibawah ini.

Tabel 4.2 Pengujian Potensiometer

No	Nilai Tegangan (V)	Nilai Tegangan yang Terlihat (V)	Posisi Potensiometer
1.	0V	0V	
2.	1V	1,85V	Menyala
3.	2V	2,15V	Redup
4.	3V	3,45V	Redup
5.	4V	4,68V	Redup
6.	5V	4,97VG	Gelap

Potensiometer ini dapat secara efektif mengatur tegangan yang keluar dari arduino dengan baik. Oleh karena itu, potensiometer dapat digunakan sebagai set point (SV) pada sistem kendali yang mengatur intensitas cahaya lampu dengan PID.

4.3.3 Pengujian MOSFET (driver) dan Lampu (Aktuator)

Dalam proyek akhir ini menggunakan MOSFET IRFZ44N sebagai driver dan lampu DC 12V sebagai aktuator. Pengujian dilakukan untuk memahami dan menentukan apakah driver dapat digunakan untuk menggerakkan lampu yang bertegangan 12 volt sebagai 5 volt. Berikut tabel 4.3 merupakan program pengujian pada MOSFET.

Tabel 4.3 Program Pengujian MOSFET

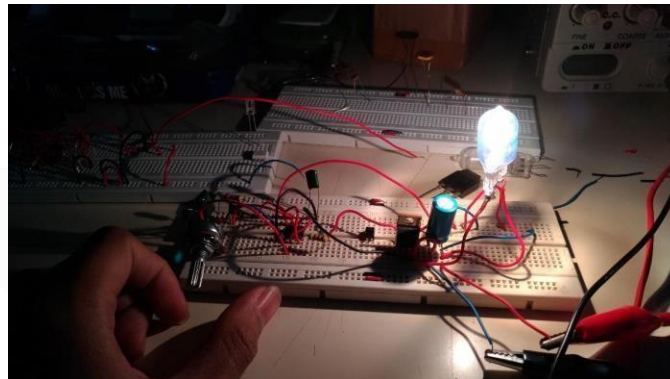
Program Pengujian MOSFET
<pre>float a; void setup () { Serial.begin (9600) ; }</pre>

```

void loop ()
  a=analogRead(0)*0.0049
  Serial.print (“ ”);
  Serial.println (a);
  Delay (100);
}

```

Setelah dilakukan pengujian program pada MOSFET, di bawah ini adalah gambar 4.5 hasil dari pengujian MOSFET (driver) dan lampu (aktuator).



Gambar 4. 5 Pengujian MOSFET (driver) dan lampu (aktuator)

Hasil daripada pengujian yang dilakukan pada MOSFET dan lampu akan ditampilkan di bawah ini pada tabel 4.4.

Tabel 4. 4 Hasil pengujian MOSFET

No	Nilai Tegangan (V)	Nilai Tegangan yang Terlihat (V)	Posisi MOSFET
1.	0V	0V	
2.	1V	1,85V	Menyala
3.	2V	2,15V	Redup
4.	3V	3,45V	Redup
5.	4V	4,68V	Redup
6.	5V	4,97V	Gelap

Dalam pengujian dari MOSFET dan Lampu ini menghasilkan kondisi lampu berbeda-beda sesuai dengan nilai tegangan yang dibutuhkan, seperti gambar 4.6 kondisi lampu pada nilai tegangan yang dibutuhkan 1V dalam keadaan menyala.



Gambar 4. 6 Pengujian dari Hasil MOSFET dan Lampu Menyala Terang (1V)

Kemudian kondisi lampu pada gambar 4.7 menunjukkan hasil lampu pada nilai tegangan yang dibutuhkan 2V dalam keadaan menyala juga.



Gambar 4. 7 Pengujian dari Hasil MOSFET dan Lampu Menyala Redup Sedang (2V)

Lalu pada pengujian di nilai tegangan yang dibutuhkan yaitu 3V menunjukkan lampu menyala redup sedang, seperti gambar 4.8 di bawah ini.



Gambar 4. 8 Pengujian Hasil MOSFET Menyala Redup Sedang (3V)

Gambar 4.9 juga menunjukkan kondisi lampu yang dilakukan pengujian dengan nilai tegangan yang dibutuhkan 4V.



Gambar 4. 9 Pengujian dari Hasil MOSFET dan Lampu Menyala Redup (4V)

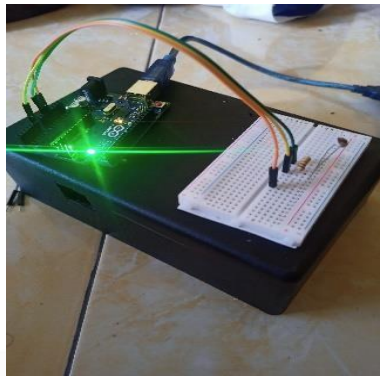
Gambar 4.10 menunjukkan kondisi lampu yang dilakukan pengujian dengan nilai tegangan yang dibutuhkan yaitu 5V.



Gambar 4. 10 Pengujian dari Hasil MOSFET dan Lampu (5V)

4.3.4 Pengujian Sensor LDR (Feedback)

Pengujian sensor dilakukan dengan memberikan cahaya pada sensor LDR. Kemudian sensor harus dikalibrasi agar output pada plotter yang awalnya disetel ke volt diubah menjadi lux (satuan cahaya), seperti gambar 4.11 yang menunjukkan pengujian sensor LDR.



Gambar 4. 11 Pengujian Sensor LDR

Setelah selesai melakukan pengujian sensor LDR, berikut gambar 4.12 yang menunjukkan hasil dari plotter pengujian sensor LDR.



Gambar 4. 12 Hasil Plotter Pengujian Sensor LDR

Di bawah ini menunjukkan tabel 4.5 yang merupakan program dari pengujian sensor LDR.

Tabel 4.5 Program Pengujian Sensor LDR

Program Pengujian Potensiometer
<pre>float a; void setup () { Serial.begin (9600) ; } void loop () a=analogRead(0)*0.004887; Serial.print (a) ; Serial.println (“”); Delay (100) ; }</pre>

4.3.5 Pengujian pengaruh beban terhadap sensor

Pengujian ini dilakukan dengan cara memberikan cahaya dari luar untuk dapat melihat reaksi dari sensor LDR tersebut. Berikut adalah gambar 4.13 hasil respon saat diberi gangguan dari cahaya luar.



Gambar 4. 13 Tampilan LCD SV dan PV

Di bawah ini juga ditampilkan tabel 4.6 sebagai data antar sensor untuk nilai maksimum SV dan PV ketika diberi objek.

Tabel 4. 6 Nilai Maksimum Ketika diberi Objek

No.	Jarak Sensor	Data SV dan PV	Kondisi Lampu
1.	2 cm	SV: 9998.80 LUX PV: 12257.17UX	Redup
2.	4 cm	SV: 9998.80 LUX PV: 12033.17UX	Redup
3.	6 cm	SV: 9998.80 LUX PV: 10857.94UX	Redup
4.	8 cm	SV: 9998.80 LUX PV: 10048.65UX	Redup
5.	10 cm	SV: 9998.80 LUX PV: 9998UX	Redup

Di bawah ini juga ditampilkan tabel 4.7 sebagai data antar sensor untuk nilai minimum SV dan PV ketika diberi objek.

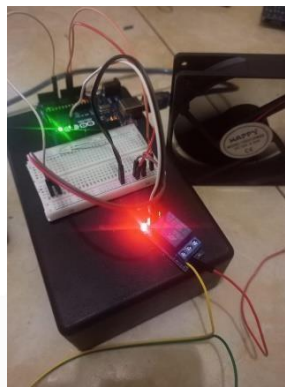
Tabel 4. 7 Nilai Minimum Ketika diberi Objek

No.	Jarak Sensor	Data SV dan PV	Kondisi Lampu
1.	2 cm	SV: 0.00 LUX PV: 7301.65UX	Terang
2.	4 cm	SV: 0.00 LUX PV: 5597.97UX	Redup
3.	6 cm	SV: 0.00 LUX PV: 5243.61UX	Redup
4.	8 cm	SV: 0.00 LUX PV: 5428.97UX	Redup
5.	10 cm	SV: 0.00 LUX PV: 4029.58UX	Redup

Saat cahaya dari luar diterima, maka sensor akan membaca cahaya lebih teliti dari set point sehingga menyebabkan lampu menjadi redup dan saat cahaya dari luar hilang, lampu akan membutuhkan waktu untuk menyala kembali agar dapat mendapatkan respon seperti pada gambar 4.13.

4.4 Pengujian Sensor Suhu pada Kipas Angin

Pengujian sensor suhu pada kipas angin juga dilakukan menggunakan sensor LM35, seperti pada gambar 4.14 dibawah ini.



Gambar 4.14 Pengujian Suhu pada Kipas Angin

Pengujian dilakukan menggunakan program yang menjalankan suhu pada kipas, berikut tabel 4.8 di bawah ini.

Tabel 4.8 Program Pengujian pada Kipas Angin

Program Pengujian Sensor Suhu Pada Kipas Angin
<pre>// Pin yang digunakan untuk mengendalikan kipas DC const int fanPin = 9; // Pin yang digunakan untuk membaca sensor suhu LM35 const int lm35Pin = A0; // Variabel untuk menyimpan nilai suhu float temperature = 0;</pre>

```
void setup() {
  // Mengatur pin fanPin sebagai OUTPUT
  pinMode(fanPin, OUTPUT);

  // Memulai komunikasi serial
  Serial.begin(9600);
}

void loop() {
  // Membaca nilai suhu dari sensor LM35
  int sensorValue = analogRead(lm35Pin);

  // Mengonversi nilai analog menjadi suhu dalam derajat Celsius
  temperature = (sensorValue * 5.0 / 1024.0) * 100;

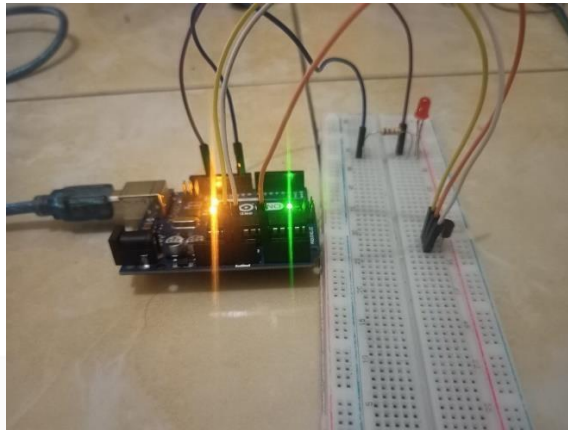
  // Menampilkan nilai suhu pada Serial Monitor
  Serial.print("Suhu: ");
  Serial.print(temperature);
  Serial.println(" °C");

  if (temperature <=30) {
    digitalWrite(fanPin, HIGH); // Menghidupkan kipas menggunakan relay
  } else {
    digitalWrite(fanPin, LOW); // Mematikan kipas menggunakan relay
  }

  delay(100);
}
```

4.4.1 Pengujian Sensor LM35

Pada pengujian ini menggunakan korek api sebagai penghantar panas, sensor suhu LM35 akan mendeteksi panas dan memberikan hasil ke mikrokontroler untuk dieksekusi sehingga akan memberikan perintah untuk menampilkan karakter dan motor DC berputar, seperti gambar 4.15 di bawah ini.



Gambar 4. 15 Pengujian Sensor LM35

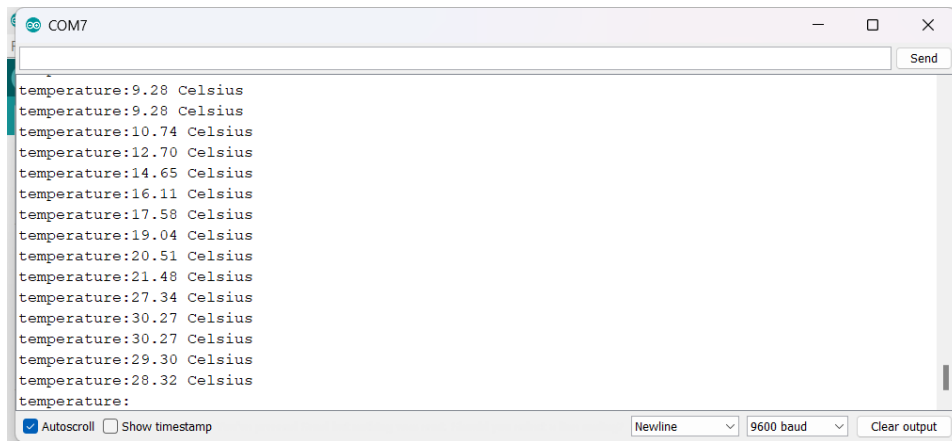
Tabel 4.9 menunjukkan program pengujian pada sensor LM35.

Tabel 4. 9 Pengujian sensor LM35

Program Pengujian Sensor LM35
<pre>// Konstanta pin const int sensor = A0; // Pin analog untuk sensor LM35 const int LED = 9; // Pin untuk mengendalikan LEDfloat suhu ; // Variabel untuk menyimpan suhu void setup() { // Mengatur pin sebagai OUTPUT pinMode(LED, OUTPUT); // Memulai komunikasi serial Serial.begin(9600); }</pre>

```
void loop() {  
  // Membaca suhu dari sensor LM35  
  int nilaiDigital = analogRead(sensor);  
  suhu = (5.0 * nilaiDigital *100.0) / 1024.0; // Konversi nilai analog ke suhu  
  dalam derajat Celsius  
  
  // Menampilkan suhu pada Serial Monitor  
  Serial.print("temperature:");  
  Serial.print(suhu);  
  Serial.println(" Celsius");  
  delay(300);  
  
  // Mengendalikan LED berdasarkan suhu  
  if (suhu > 30) {  
    digitalWrite(LED, HIGH); // Menghidupkan LED  
  } else {  
    digitalWrite(LED, LOW); // Mematikan LED  
  }  
  
  delay(100); // Delay 1 detik sebelum membaca suhu berikutnya  
}
```


Berikut gambar tampilan 4.16 sensor suhu LM35 yang mendeteksi panas dari korek api.



Gambar 4. 16 Tampilan Sensor Suhu LM35 Mendeteksi Panas dari Korek Api

Dan pembacaan data yang diambil disesuaikan dengan suhu yang dicapai, berikut tabel 4.10 menunjukkan hasil data pembacaan sensor LM35 pada kipas angin yang mendeteksi panas dari korek api.

Tabel 4.10 Hasil dari Pengujian Kipas Angin

NO	Suhu yang Dicapai	Data Pembacaan Sensor	Kondisi Kipas
1	25°	20°	OFF
2	25°	21°	OFF
3	25°	22°	OFF
4	25°	23°	OFF
5	25°	24°	OFF
6	25°	25°	ON
7	25°	26°	ON
8	25°	27°	ON
9	25°	28°	ON
10	25°	29°	ON
11	25°	30°	ON

BAB V

KESIMPULAN

5.1 Kesimpulan

Berdasarkan hasil perancangan dan penelitian prototype “Pengontrolan Intensitas Cahaya Lampu dan Sensor Suhu Otomatis Menggunakan Metode PID Berbasis Mikrokontroler”, maka dapat disimpulkan sebagai berikut:

1. Berhasil dibangun sistem pengontrol intensitas cahaya lampu otomatis dengan ukuran 50 cm x 50 cm dan jarak antara sensor dengan lampu adalah 10 cm. Sistem ini mampu membaca intensitas cahaya dalam rentang 0 - 120 lux dengan tingkat akurasi yang sesuai.
2. Dalam pengembangan sistem kontrol kipas angin otomatis, digunakan mikrokontroler sebagai pengendali utama. Kipas angin otomatis ini menggunakan sensor suhu LM35 untuk mendeteksi suhu ruangan. Sensor suhu LM35 akan memberikan informasi suhu yang akurat dan dapat diandalkan untuk mengendalikan operasi kipas angin secara otomatis sesuai dengan suhu ruangan yang terdeteksi.
3. Berdasarkan penelitian ini, dapat disimpulkan bahwa penggunaan metode PID dalam pengendalian intensitas cahaya dan penggunaan sensor suhu LM35 untuk mengendalikan kipas angin secara otomatis merupakan solusi yang efektif dan efisien. Sistem yang dikembangkan mampu mengoptimalkan penggunaan lampu dan kipas angin berdasarkan kondisi cahaya dan suhu yang ada, sehingga memberikan kenyamanan dan efisiensi energi yang lebih baik.

5.2 Saran

Apabila nanti ada pengembangan alat untuk kedepannya, ada beberapa saran dari penulis antara lain:

1. Penggunaan sensor yang lebih baik mengganti sensor cahaya dan suhu dengan sensor yang memiliki akurasi dan respons yang lebih tinggi.

Misalnya, menggunakan sensor cahaya dengan resolusi yang lebih tinggi atau sensor suhu dengan tingkat presisi yang lebih baik.

2. Melakukan evaluasi dan analisis performa sistem secara mendalam. Mengukur respons sistem terhadap perubahan intensitas cahaya dan suhu, serta mengidentifikasi waktu respons, stabilitas, *overshoot*, dan *error* yang terjadi. Evaluasi ini akan memberikan pemahaman yang lebih baik tentang kinerja sistem dan membantu dalam mengidentifikasi area perbaikan yang mungkin diperlukan.



DAFTAR PUSTAKA

- [1] I. Setiawan, “Kontrol PID untuk proses industri” Surabaya: PT. Elex Media Komputindo; 2008.
- [2] Taufiqullah. (2023, January 21). Diambil kembali dari tneutron.net: <https://www.neutron.net/industri/sistem-kendali-manual-dan-otomatis.com>
- [3] Konservasi Energi pada Sistem Pencahayaan. (t.thn.). Diambil kembali dari PanduanTeknikPenerangan:<https://pupr.tebingtinggikota.go.id/wpcontent/upload/2020/11/>
- [4] Arindya, R. (2014). Instrumentasi dan kontrol proses. *Graha Ilmu*.
- [5] Arindya R., “Instrumentasi dan Kontrol Proses” Yogyakarta: Graha Ilmu; 2014
- [6] Budiharto W. Aneka Proyek Mikrokontroler, “Panduan Utama untuk riset atau tugas akhir” Yogyakarta; 2011
- [7] Al-Islami, M. A. (t.thn.). Diambil kembali dari scribd.com: <https://www.scribd.com/doc/307202983/Bab-1-Pengantar-Sistem-kendali>
- [8] Tanza, Nendya Artika. Sumariyah, “Rancang Bangun Sistem Kendali PID untuk Intensitas Cahaya Lampu DC Menggunakan Mikrokontroler Arduino Uno” Vol. 22, No. 1, Januari 2019, Hal 32-40
- [9] Kadir A(2013). Panduan praktis mempelajari pengaturan set point.
- [10] Akbar. Khoirul, “Simulasi Kipas Angin Otomatis Deteksi Suhu Ruangan Dengan LM35 Berbasis Arduino”

DAFTAR LAMPIRAN

Lampiran 1 Daftar Riwayat Hidup

DAFTAR RIWAYAT HIDUP

1. Data Pribadi

Nama Lengkap : Raihan Afiruru
Tempat Tanggal Lahir : Temanggung, 19 Februari 2003
Alamat Rumah : Jalan Sudirman, Dusun Kalmoa
Desa Mekar Jaya, Kec. Manggar
Jenis Kelamin : Laki-laki
Agama : Islam
Telp : -
Email : mynameisrehan19@gmail.com



2. Riwayat Pendidikan

SD Negeri 1 Manggar	2008-2014
SMP Negeri 4 Manggar	2014-2017
SMA Negeri 1 Manggar	2017-2020
Politeknik Manufaktur Negeri Bangka Belitung	2020-2023

3. Pengalaman Kerja

Praktik Kerja Lapangan di PT. Global Sahabat Otomasi Tahun 2022.

Sungailiat, 17 Juli 2023

(Raihan Afiruru)

DAFTAR RIWAYAT HIDUP

1. Data Pribadi

Nama Lengkap : Sephia Ocvi
Tempat Tanggal Lahir : Deniang, 03 Oktober 2002
Alamat Rumah : Jalan Gurami 1, Gabek Permai,
Kota Pangkalpinang
Jenis Kelamin : Perempuan
Agama : Islam
Telp : -
Email : sephiaocvi@gmail.com



2. Riwayat Pendidikan

SD Negeri 62 Pangkalpinang	2008-2014
SMP Negeri 7 Pangkalpinang	2014-2017
SMA Negeri 2 Pangkalpinang	2017-2020
Politeknik Manufaktur Negeri Bangka Belitung	2020-2023

3. Pengalaman Kerja

Praktik Kerja Lapangan di PT. Veritech Perdana Tahun 2022.

Sungailiat, 17 Juli 2023

(Sephia Ocvi)

Lampiran 2 Program

Program Pengontrolan Intensitas Cahaya Lampu Otomatis

```
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27,16,2);

float PID,ut,ut_1;
float et,et_1,et_2; //et=serror sekarang, et_1=errorsebelumnya
float Kp, Ti, Td, K1,K2,K3,K4;
float SV, PV;
int MV;

unsigned long t;
double t_1, Ts;

float interval_elapsed;
float interval_limit;

int start;

void setup() {
  lcd.init();
  lcd.backlight();
  Kp = 4.62345733;
  Ti = 31990;
  Td = 7997.5;
  et_1 = 0;
  et_2 = 0;
  ut_1 = 0;

  interval_limit = 5;
```

```

interval_elapsed=0;

lcd.setCursor(0,0);
lcd.print("SV:");
lcd.setCursor(10,0);
lcd.print("LUX:");
lcd.setCursor(0,1);
lcd.print("PV");
lcd.setCursor(10,1);
lcd.print("LUX");
t=millis();
delay (100);
pinMode(13,OUTPUT);
pinMode(8,INPUT);
pinMode(6,OUTPUT);
digitalWrite(13,HIGH);
Serial.begin(9600);
}

void loop() {
  // put your main code here, to run repeatedly:
  start=digitalRead(8);

  while (start == 1){

    SV=analogRead(A2)*0.004887;
    SV=SV*2000;//tergantung batas SV yang diinginkan disini 0-10.000

    PV=analogRead(A3)*0.004887;
    PV=PV*3450;//disesuaikan dengan LM35 1 c=10mV
  }
}

```



```

t=millis();
Ts=0.01;//(t - t_1)/1000;
et=SV-PV;
if (Ti<=0){
    K1=0;
}
else{
    K1=(Kp*Ts)/Ti;
}
K2=(Kp*Td)/Ts;
K3=Kp+K2+K1;
K4=Kp+2*K2;

ut=ut_1 + K3*et- K4*et_1 + K2*et_2;
PID = ut;

if (PID>9){
    PID=9;
}
else if (PID < 0){
    PID=0;
}
else {
    PID=PID;
}
PID= PID/2;
MV=PID*51;
analogWrite(6,MV); //mengirim sinyal kendali ke PWM 6

```

```

interval_elapsed = interval_elapsed + Ts;

if (interval_elapsed >= interval_limit){

    Serial.print(SV); //batas bawah grafik
    Serial.print(" ");
    Serial.print(PV);
    Serial.print(" ");
    Serial.print(0);
    Serial.print(" ");
    Serial.println(10);

    lcd.setCursor(3,0);
    lcd.print(SV);
    lcd.setCursor(3,1);
    lcd.print(PV);
    interval_elapsed=0;
}
else {
    interval_elapsed=interval_elapsed;
}

et_1=et;
et_2=et_1;
ut_1=ut;
t_1=t;

start=digitalRead(8);
}
analogWrite(6,0);

interval_elapsed=0;

```

```
et_1=0;
et_2=et_1;
ut_1=0;

t=millis();
SV=analogRead(A2)*0.004887;
PV=analogRead(A3)*0.004887;

lcd.print(SV);
lcd.print("");
lcd.print(PV);
lcd.print("");
lcd.print(0);
lcd.print("");
lcd.print(10);
lcd.setCursor(3,0);
lcd.print(SV);
lcd.setCursor(3,1);
lcd.print(PV);

}
```

Program Pengontrolan Kipas

```
#include "Wire.h"
#include <LiquidCrystal.h>
LiquidCrystal lcd(7, 6, 5, 4, 3, 2);

int relay = 10;
int kipas = 9;

const int btset = A2;
const int btup = A3;
const int btdown = A2;
const int btok = A5;

int btsetx = 0;
int btupx = 0;
int btdownx = 0;
int btokx = 0;
float sv = 35;
float hys = 1;
float onrelay = 37;

int adcdat;

float simpansv = 35;
float simpanhys = 1;
float simpanonrelay = 37;

float kp = 1.85;
float ki = 1.67;
float kd = 1.15;
```

```
float batasatas;
float batasbawah;

float p,i,d,suhu,pid;
float error,errorx,sumerr;

void setup() {
  lcd.begin(16,2);
  lcd.clear();
  lcd.noCursor();
  pinMode(relay,OUTPUT);
  pinMode(A1,OUTPUT);
  pinMode(kipas,OUTPUT);
  pinMode(btset,INPUT);
  pinMode(btup,INPUT);
  pinMode(btdown,INPUT);
  pinMode(btok,INPUT);
  digitalWrite(btset,HIGH);
  digitalWrite(btup,HIGH);
  digitalWrite(btdown,HIGH);
  digitalWrite(btok,HIGH);
  digitalWrite(A1,LOW);
}

void loop() {

  btsetx = digitalRead(btset);

  if(btsetx == 0){
    delay(200);
    lcd.clear();
```



```
lcd.setCursor(0, 0);
lcd.print("LOADING...");
delay(3000);
lcd.clear();
setting();
lcd.setCursor(0, 0);
lcd.print("LOADING...");
delay(3000);
lcd.clear();
settinghy();
lcd.setCursor(0, 0);
lcd.print("LOADING...");
delay(3000);
lcd.clear();
settingrelay();
lcd.clear();
}

error = batasatas - suhu;
p = error * kp;
sumerr = error + errorx;
i = ki * sumerr;
d = error - errorx;
pid = p + i + d;

if(pid < 1){
pid = 0;
}

adcdat = analogRead(0);
```

```
suhu = (adcdat * (5.0 / 1023.0)) * 100;

analogWrite(kipas,pid);

lcd.setCursor(0, 0);
lcd.print("PV=");
lcd.print(suhu,1);
lcd.setCursor(0, 1);
lcd.print("SV=");
lcd.print(simpansv,1);
lcd.setCursor(8, 0);
lcd.print("HYS=");
lcd.print(simpanhys,1);
lcd.setCursor(8, 1);
lcd.print("PWM=");
//lcd.print(simpanonrelay,1);
lcd.print(pid,1);
delay(200);

batasatas = simpansv + simpanhys;
batasbawah = simpansv - simpanhys;

if(suhu >= simpanonrelay){
digitalWrite(relay,HIGH);
}
if(suhu < simpanonrelay){
digitalWrite(relay,LOW);
}

errorx = error;
```

```
}

//=====
void setting(){

    lcd.setCursor(0, 0);
    lcd.print("SETTING SV");
    lcd.setCursor(0, 1);
    lcd.print(sv);

    btupx = digitalRead(btup);
    btkeydown = digitalRead(btkeydown);
    btokx = digitalRead(btok);

    if(btupx == 0){
        delay(200);
        sv = sv + 0.5;
    }

    if(btkeydown == 0){
        delay(200);
        sv = sv - 0.5;
    }

    if(btokx == 0){
        delay(200);
        lcd.clear();
        simpansv = sv;
        return;
    }
}
```



```
setting();  
}  
  
void settinghy(){  
  
  lcd.setCursor(0, 0);  
  lcd.print("SETTING HYS");  
  lcd.setCursor(0, 1);  
  lcd.print(hys);  
  
  btupx = digitalRead(btup);  
  btkeydown = digitalRead(btkeydown);  
  btokx = digitalRead(btok);  
  
  if(btupx == 0){  
    delay(200);  
    hys = hys + 0.5;  
  }  
  
  if(btkeydown == 0){  
    delay(200);  
    hys = hys - 0.5;  
  }  
  
  if(btokx == 0){  
    delay(200);  
    lcd.clear();  
    simpanhys = hys;  
  }  
}
```

```
    return;
}

settinghy();
}

void settingrelay(){

    lcd.setCursor(0, 0);
    lcd.print("SETTING ON RELAY");
    lcd.setCursor(0, 1);
    lcd.print(onrelay);

    btupx = digitalRead(btup);
    btkeydown = digitalRead(btkeydown);
    btokx = digitalRead(btok);

    if(btupx == 0){
        delay(200);
        onrelay = onrelay + 0.5;
    }

    if(btkeydown == 0){
        delay(200);
        onrelay = onrelay - 0.5;
    }

    if(btokx == 0){
```

```
    delay(200);  
    lcd.clear();  
    simpanonrelay = onrelay;  
    return;  
}  
  
settingrelay();  
  
}
```

