



PROYEK AKHIR - VE 180626

**RANCANG BANGUN SISTEM KONTROL SUHU
PADA RUANG PANEL INVERTER MENGGUNAKAN
MIKRO KONTROLER**

Andy Rahma Septianto
NRP 10311600010003

Dosen Pembimbing
Ir. Joko Susila, MT
Dr. Eng. Imam Wahyudi Farid, S.T., M.T.

DEPARTEMEN TEKNIK ELEKTRO OTOMASI
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember
Surabaya 2020



FINAL PROJECT - VE 180626

DESIGN OF TEMPERATURE CONTROL SYSTEM ON INVERTER PANEL ROOM USING MICRO CONTROLLER

Andy Rahma Septianto
NRP 10311600010003

Advisor
Ir. Joko Susila, MT
Dr. Eng. Imam Wahyudi Farid, S.T., M.T.

DEPARTEMENT OF ELECTRICAL ENGINEERING AUTOMATION
Faculty of Vokasi
Sepuluh Nopember Insitute of Technology
Surabaya 2020

PERNYATAAN KEASLIAN

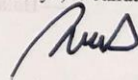
PROYEK AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Proyek Akhir saya dengan judul "**Rancang Bangun Sistem Kontrol Suhu pada Ruang Panel Inverter Menggunakan Mikrokontroler**" adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, 20 Januari 2020



Andy Rahma Septianto
NRP 10311600010003

-----Halaman ini sengaja dikosongkan-----

**RANCANG BANGUN SISTEM KONTROL SUHU PADA RUANG
PANEL INVERTER MENGGUNAKAN MIKROKONTROLER**

PROYEK AKHIR

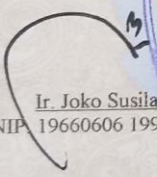
Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Ahli Madya
Pada

Program Studi Komputer Kontrol
Departemen Teknik Elektro Otomasi
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember

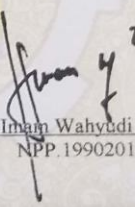
Menyetujui:

Dosen Pembimbing 1,

Dosen Pembimbing 2,


Ir. Joko Susila, MT
NIP. 19660606 199102 1 001


DEPARTEMEN
TEKNIK ELEKTRO OTOMASI


Dr. Eng. Imam Wahyudi Farid, S.T., M.T.
NPP. 199020191186

**SURABAYA
JANUARI, 2020**

-----Halaman ini sengaja dikosongkan-----

RANCANG BANGUN SISTEM KONTROL SUHU PADA RUANG PANEL INVERTER MENGGUNAKAN MIKRO KONTROLER

Pembimbing I : Ir. Joko Susila, MT

Pembimbing II : Dr. Eng. Imam Wahyudi Farid, S.T., M.T.

ABSTRAK

Proyek Akhir ini berfokus pada perancangan otomatis pendeteksi suhu pada ruang panel inverter dan perancangan otomatisasi pengontrol suhu menggunakan kipas. Parameter yang digunakan dalam alat ini yaitu 40°C-45°C. Alat ini menggunakan Mikrokontroler Arduino Uno.

Cara kerja alat ini yaitu dengan menentukan setpoint minimal dan maksimal suhu (40°C-45°C) yang di deteksi oleh sensor suhu. Jika sensor mendeteksi suhu kurang dari suhu minimal 40°C maka hanya kipas 1 yang menyala, tetapi jika sensor suhu mendeteksi suhu lebih dari suhu minimal 40°C maka kipas 2 akan menyala sampai suhu dalam panel menurun ke suhu minimal 40°C.

Hasil dari alat ini, dapat mengontrol suhu pada panel inverter yang mana menggunakan kipas yang menyala otomatis sebagai penurun suhu pada ruang panel. Dari data yang telah diuji kesimpulan bahwa metode memasukkan udara ke kipas lebih cepat untuk menurunkan suhu panas dengan rata rata waktu yang di butuhkan kurang lebih 1 sampai 2 menit dari suhu 80°C sampai 45°C. Sedangkan metode mengeluarkan udara dari dalam panel membutuhkan waktu lebih lama untuk menurunkan suhu panas dengan rata rata waktu yang di butuhkan kurang lebih 5 – 6 menit dari suhu 80°C sampai 45°C.

Kata Kunci: Kipas, Suhu Ruang

-----Halaman ini sengaja dikosongkan-----

DESIGN OF TEMPERATURE CONTROL SYSTEM ON INVERTER PANEL ROOM USING MICRO

Advisor I : Ir. Joko Susila, MT

Advisor II : Dr. Eng. Imam Wahyudi Farid, S.T., M.T.

ABSTRACT

This Final Project focuses on the design of automatic temperature detection in the inverter panel chamber and the design of a temperature controller automation using a fan. The parameters used in this tool are 40°C-45°C. This tool uses Arduino Uno Microcontroller.

The way this tool works is by determining the minimum and maximum setpoints of temperature (40°C-45°C) detected by the temperature sensor. If the sensor detects a temperature less than a minimum temperature of 40°C then only fan 1 lights up, but if the temperature sensor detects a temperature more than a minimum temperature of 40°C then fan 2 will turn on until the temperature in the panel decreases to a temperature of at least 40°C.

The results of this tool, can control the temperature on the inverter panel which uses a fan that turns on automatically as a temperature reducer in the panel chamber. From the data that has been tested the conclusion that the method of entering air into the fan is faster to lower the temperature of the heat with an average time required of approximately 1 to 2 minutes from a temperature of 80°C to 45°C. While the method of removing air from the panel takes longer to reduce hot temperature with an average time needed approximately 5-6 minutes from 80°C to 45°C.

Keywords: *Fan, Room Temperature*

-----Halaman ini sengaja dikosongkan-----

KATA PENGANTAR

Puji syukur kami panjatkan kehadiran Allah SWT, karena berkat rahmat dan hidayah-Nya, penulis dapat menyelesaikan pengerjaan hingga menyusun buku Proyek Akhir dengan judul “Rancang Bangun Sistem Kontrol Suhu pada Ruang Panel Inverter Menggunakan Mikrokontroler”. Pada kesempatan kali ini, penulis ingin menyampaikan rasa terimakasih kepada kedua orang tua serta keluarga yang telah memberi dukungan dari awal hingga saat ini. Kepada Bapak Ir. Joko Susila, MT selaku Dosen Pembimbing pertama, Bapak Dr. Eng. Imam Wahyudi Farid, S.T., M.T. selaku dosen pembimbing kedua. Kepada kepala Departemen dan segenap civitas akademik Teknik Elektro Otomasi yang telah mendidik mulai dari awal hingga akhir perkuliahan.

Penulis menyadari bahwa dalam penyusunan laporan ini tidak lepas dari kekurangan. Oleh karena itu, kritik dan saran yang bersifat membangun sangat diperlukan demi kebaikan dan kesempurnaan penyusunan laporan dimasa yang akan datang. Semoga laporan ini dapat memberikan manfaat bagi penulis khususnya dan pembaca umumnya.

Surabaya, 20 Januari 2020

Andy Rahma Septianto

-----Halaman ini sengaja dikosongkan-----

DAFTAR ISI

	HALAMAN
ALAMAN JUDUL	I
HALAMAN JUDUL	III
PERNYATAAN KEASLIAN.....	ERROR! BOOKMARK NOT DEFINED.
PROYEK AKHIR.....	ERROR! BOOKMARK NOT DEFINED.
ABSTRAK.....	IX
ABSTRACT.....	XI
KATA PENGANTAR.....	XIII
DAFTAR ISI.....	XV
DAFTAR GAMBAR.....	XVII
DAFTAR TABEL	XIX
BAB I.....	1
PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Permasalahan.....	1
1.3 Batasan Masalah.....	1
1.4 Tujuan	2
1.5 Sistematika Laporan.....	2
1.6 Relevansi	3
BAB II.....	5
TEORI PENUNJANG	5
2.1 Panel inverter.....	5
2.2 Suhu	6
2.3 Arduino Uno.....	7
2.3.1 Bagian <i>Software</i>	12
2.3.2 Setting Serial	13
2.4 Relay	14
2.5 Thermistor NTC 3950	15
2.6 Dimmer Ac light 220v	17
2.7 Power Supply	19
2.8 Kipas AC 220V	23
2.9 LCD 16X2	25
BAB III	29
PERANCANGAN SISTEM.....	29

3.1 Diagram Sistem Kerja Alat.....	30
3.2 Perancangan Perangkat Elektrik	31
3.2.1 <i>Setting Port</i> Mikrokontroler	31
3.2.2 Perancangan Rangkaian Keseluruhan Alat	31
3.2.3 Perancangan Rangkaian Sensor.....	33
3.2.4 Perancangan Rangkaian Kipas 1 Disambungkan pada <i>Dimmer</i>	33
3.2.5 Perancangan Rangkaian Rangkaian Kipas 2 Disambungkan pada <i>relay</i>	34
3.2.6 Rangkaian LCD	34
3.2.7 Perancangan Rangkaian Keseluruhan Alat	35
BAB IV.....	37
PENGUJIAN DAN ANALISA	37
4.1 Pengujian Sensor NTC 3950.....	37
4.2 Pengujian relay dan kipas 2	38
4.3 Pengujian <i>Dimmer</i> dan kipas 1	39
4.4 Pengujian <i>Liquid Crystal Display</i> (LCD).....	40
4.5 Pengujian Alat Keseluruhan	41
BAB V	45
PENUTUP	45
5.1. Kesimpulan	45
5.2. Saran	45
DAFTAR PUSTAKA	47
LAMPIRAN A	A-1
LAMPIRAN B	B-1
LAMPIRAN C	C-1
RIWAYAT PENULIS	C-3

DAFTAR GAMBAR

Gambar 2. 1	Panel Inverter	5
Gambar 2. 2	Hardware Arduino Uno	8
Gambar 2. 3	Fungsi Port Arduino.....	9
Gambar 2. 4	Software Arduino.....	13
Gambar 2. 5	Setting Serial Port	14
Gambar 2. 6	Bagian dan Prinsip Kerja dari Relay	15
Gambar 2. 7	Sensor Thermistor NTC 3950.....	16
Gambar 2. 8	Ilustrasi Perubahan Resistansi PTC dan NTC	16
Gambar 2. 10	Diagram Blok Power Supply	20
Gambar 2. 11	Power supply 12VDC	20
Gambar 3. 1	Perancangan Perangkat Mekanik Tampak Samping	29
Gambar 3. 2	Perancangan Perangkat Mekanik Tampak Dalam.....	29
Gambar 3. 3	Perancangan Perangkat Elektrik	30
Gambar 3. 4	Diagram Sistem Kerja sensor NTC	30
Gambar 3. 5	Flowchart Pembacaan Sensor NTC	32
Gambar 3. 6	Rangkaian Rangkaian sensor	33
Gambar 3. 7	Rangkaian Kipas 1 Disambungkan pada dimmer	33
Gambar 3. 8	Rangkaian Rangkaian Kipas 2 Disambungkan pada Relay	34
Gambar 3. 9	Rangkaian LCD	34
Gambar 3. 10	Perancangan Rangkaian Keseluruhan Alat.....	35
Gambar 4. 2	Metode Pengujian sensor NTC3950.....	38
Gambar 4. 3	Pengujian relay dan kipas 2	38
Gambar 4. 4	Pengujian Dimmer dan kipas 1	39
Gambar 4. 5	Rangkaian Pengujian LCD	40
Gambar 4. 6	Tampilan Hasil Pengujian LCD.....	41

-----Halaman ini sengaja dikosongkan-----

DAFTAR TABEL

Tabel 2. 1 Spesifikasi Arduino Uno.....	8
Tabel 2. 2 Spesifikasi LCD 16X 2	26
Tabel 3. 1 Setting Port Mikrokontroler.....	31
Tabel 3. 2 Setting Port pada Rangkaian Keseluruhan Alat	35
Tabel 4. 1 Pengujian relay dan kipas 2	39
Tabel 4. 2 Pengujian Dimmer dan kipas 1.....	40
Tabel 4. 3 Pengujian Menurunkan Suhu Panas Dengan Kondisi Kipas Mengeluarkan Udara dari Dalam Panel.	41
Tabel 4. 4 Pengujian Menurunkan Suhu Panas Dengan Kondisi Kipas Memasukkan Udara ke Dalam Panel	42

-----Halaman ini sengaja dikosongkan-----

BAB I

PENDAHULUAN

1.1 Latar Belakang

Di dalam sebuah panel inverter terdapat banyak sekali komponen yang fungsinya masing-masing berguna untuk kinerja inverter. Baik atau tidaknya kinerja sebuah inverter tentunya terpengaruh dari komponen yang ada di dalamnya. Jika komponen dalam keadaan yang bagus atau baik, maka inverter pun akan memiliki kinerja yang baik pula, dan begitu juga dengan sebaliknya jika komponen yang terdapat di dalamnya rusak atau tidak baik, maka inverter tidak akan dapat bekerja dengan baik pula. Salah satu permasalahan yang dialami oleh satu komponen tersebut adalah temperatur suhu yang terlampaui panas sehingga membuat satu komponen tersebut menjadi mengalami penurunan kualitas bahkan sampai terjadi kerusakan pada komponen. Untuk mengatasi suhu yang sangat panas pada panel inverter ditambahkan komponen tambahan berupa kipas yang dapat berputar secara otomatis sesuai dengan suhu yang diinginkan.

Pada rancang bangun kontrol Proyek Akhir menggunakan sensor thermistor, pada kondisi tertentu sensor akan mengirimkan parameter suhu yang akan diterima oleh mikrokontroler. Output dari sensor thermistor adalah relay yang digunakan untuk menaktifkan kipas 1 dan kipas 2. Jika suhu kurang dari 40°C maka arduino akan mengontrol dimmer untuk menggerakkan kipas 1 dengan kecepatan, Jika Suhu melebihi 40°C maka arduino akan mengontrol relay untuk menyalakan kipas 2.

1.2 Permasalahan

Permasalahan yang diambil pada Proyek Akhir ini adalah pada panel inverter yang tidak terdapat sensor untuk mendeteksi adanya kelebihan suhu sehingga menimbulkan panas yang berlebih. Hal ini dirasa masih kurang efektif karena akan merusak komponen yang ada dalam panel inverter.

1.3 Batasan Masalah

Untuk menghindari pembahasan yang meluas maka ditentukan batasan masalah sebagai berikut :

1. Proyek Akhir berfokus pada perancangan otomatis pendeteksi suhu pada panel.
2. Parameter yang digunakan pada proyek akhir yaitu 40°C - 45°C .
3. Suhu maksimal yang di gunakan pada box adalah 80°C .
4. Sesuai dengan datasheet, sensor hanya dapat membaca suhu -40°C - 270°C .
5. Mikrokontroler menggunakan Arduino Uno.

1.4 Tujuan

Tujuan dari penelitian ini, yaitu terbentuknya Rancang Bangun Sistem Kontrol Suhu pada Ruang Panel Inverter Menggunakan Mikrokontroler:

Mengimplementasikan peralatan tersebut agar dapat digunakan untuk membantu mengontrol suhu pada panel, serta memperpanjang usia komponen.

1.5 Sistematika Laporan

Pembahasan Proyek Akhir ini akan dibagi menjadi lima Bab dengan sistematika sebagai berikut:

Bab I PENDAHULUAN

Bab ini meliputi latar belakang, permasalahan, tujuan penelitian, metodologi penelitian, sistematika laporan, dan relevansi.

Bab II TEORI DASAR

Bab ini menjelaskan tentang tinjauan pustaka, konsep dari alat komponen penunjang, serta spesifikasi sensor.

Bab III PERANCANGAN SISTEM

Bab ini membahas tentang perancangan sistem kerja alat, perancangan perangkat mekanik, dan perancangan perangkat elektrik.

Bab IV PENGUJIAN DAN ANALISA

Bab ini berisi tentang pemaparan dan analisis hasil pengujian sistem pada keadaan sebenarnya. Seperti kalibrasi sensor dan pengujian sensor menggunakan cairan yang dicampurkan pada air serta menganalisa kondisi hasil yang didapatkan oleh sensor.

Bab V PENUTUP

Bab ini berisi kesimpulan dan saran dari hasil pembahasan yang telah diperoleh.

1.6 Relevansi

Manfaat atau relevansi dari proyek akhir ini dapat diuraikan yaitu alat ini dapat digunakan untuk membantu memperpanjang masa komponen ketika digunakan terus-menerus. Dari alat ini diharapkan dapat beroperasi dengan baik, yaitu tentang cara menurunkan suhu dalam panel yang bekerja secara otomatis

-----Halaman ini sengaja dikosongkan-----

BAB II

TEORI PENUNJANG

Beberapa teori penunjang yang dipaparkan dalam buku Proyek Akhir ini adalah teori dasar mengenai penurunan suhu, sensor yang digunakan serta beberapa komponen penunjang yang terdapat pada otomatisasi pada panel inverter.

2.1 Panel inverter



Gambar 2. 1 Panel Inverter

Sumber: <http://www.automationid.com/2016/instalasi-sederhana-variable-speed-drive-abb-ac355.html>

Gambar 2.1 merupakan panel Inverter dimana ketika PLN Menyala Maka Listrik Akan Dialirkan Ke Rumah Melalui Output 1 dan Output 2, dan Sebagian Energinya Akan Disimpan Kedalam Batrai. Ketika PLN Padam, Maka Inverter Akan Dinyalakan, Lalu Listrik Akan DiAlirkan Ke Output 1 saja, sedangkan Output 2 tetap Padam. Fungsinya Pembagiannya agar beban - beban Berat Dapat Di pisahkan Ke Output Ke 2 (MIS : Kulkas, dll). sedangkan Perangkat Ringan Dapat Disambungkan Ke Output 1. (Mis : Lampu penerangan, Dan beberapa colokan dirumah yang bebas dari beban berat. hal ini berfungsi untuk menjaga daya tahan batrai pada saat dipakai ketika PLN padam dan Juga agar Menghindari Kerusakan Inverter. Setelah PLN Menyala Kembali, Maka Otomatis Inverter Akan Off, Listrik Akan Dialirkan

Kembali ke Ouput 1 dan 2 serta Charge Mengisi Batrai kembali. Begitu Seterusnya.

2.2 Suhu

Dalam kehidupan sehari-hari, suhu merupakan ukuran mengenai panas atau dinginnya suatu benda. Oven yang panas dikatakan bersuhu tinggi, sedangkan es yang membeku dikatakan memiliki suhu rendah.

Suhu dapat mengubah sifat zat, contohnya sebagian besar zat akan memuai ketika dipanaskan. Sebatang besi lebih panjang ketika dipanaskan daripada dalam keadaan dingin. Jalan dan trotoar beton memuai dan menyusut terhadap perubahan suhu. Hambatan listrik dan materi zat juga berubah terhadap suhu. Hambatan listrik dan materi zat juga berubah terhadap suhu. Demikian juga warna yang dipancarkan benda, paling tidak pada suhu tinggi. Kalau kita perhatikan, elemen pemanas kompor listrik memancarkan warna merah ketika panas. Pada suhu yang lebih tinggi, zat padat seperti besi, bersinar jingga atau bahkan putih. Cahaya putih dari bola lampu pijar berasal dari kawat tungsten yang sangat panas.

Dengan demikian, suhu atau temperatur didefinisikan sebagai derajat panas dinginnya suatu benda. Alat untuk mengukur suhu yaitu termometer. Termometer memiliki sifat termometrik zat, yaitu akan berubah jika dipanaskan. Jenis dan paparan pada termometer berbeda-beda, namun pada prinsipnya semua termometer mempunyai acuan yang sama dalam menetapkan skala. Titik lebur es murni dipakai sebagai titik tetap bawah, sedangkan suhu uap di atas air yang sedang mendidih pada tekanan (atm sebagai titik tetap atas). Dua titik acuan (patokan) yaitu titik tetap bawah dan titik tetap atas itu digunakan untuk penentuan skala suhu, yakni kepekaan suatu besaran terhadap perubahan suhu yang disebut sifat termometrik.

Ketika tangan disentuhkan pada air dingin, kita katakan suhu air tersebut dingin. Sementara ketika tangan disentuhkan pada air hangat, kita katakan suhu air tersebut panas. Namun hal itu, tidak dapat dijadikan acuan. Mengapa demikian? Untuk mengetahuinya, coba kalian tentukan suhu air biasa di atas, apakah suhunya dingin atau panas? Langkah awal, sentuhkan tangan kananmu pada air hangat. Sementara itu, sentuhkan juga tangan kirimu pada air dingin. Selanjutnya, masukkan kedua tanganmu secara bersamaan pada wadah yang berisi air biasa.

Cermatilah pengukuran yang dilakukan kedua tanganmu. Pada saat menyentuh air biasa, tangan kananmu akan terasa dingin karena

melepas kalor sedangkan tangan kirimu akan terasa panas karena menerima kalor. Mengapa kedua tanganmu tidak merasakan hal yang sama meskipun yang disentuh adalah air yang sama, yaitu air biasa. Karena tangan tidak dapat digunakan sebagai alat ukur suhu, digunakanlah thermometer.

Ketika mengukur temperatur dengan menggunakan thermometer, terdapat beberapa skala yang digunakan, di antaranya skala Celsius, skala Reamur, skala Fahrenheit, dan skala Kelvin. Keempat skala tersebut memiliki perbedaan dalam pengukuran suhunya. Berikut rentang temperatur yang dimiliki setiap skala.

2.3 Arduino Uno

Arduino merupakan mikrokontroler yang memang dirancang untuk bisa digunakan dengan mudah oleh para teknisi dengan demikian, tanpa mengetahui bahasa pemrograman, Arduino bisa digunakan untuk menghasilkan karya yang canggih. Hal ini seperti yang diungkapkan oleh Mike Schmidt. Arduino merupakan sebuah platform hardware open source yang mempunyai input/output (I/O) yang sederhana. Menggunakan Arduino sangatlah membantu dalam membuat suatu prototyping ataupun untuk melakukan pembuatan proyek. Arduino memberikan I/O yang sudah lengkap dan bisa digunakan dengan mudah. Arduino dapat digabungkan dengan modul elektro yang lain sehingga proses perakitan jauh lebih efisien.

2.3.1 Bagian Hardware

Arduini Uno R3 adalah papan pengembangan (*development board*) mikrokontroler yang berbasis chip ATmega328P. Disebut sebagai papan pengembangan karena *board* ini memang berfungsi sebagai arena *prototyping* sirkuit mikrokontroler. Dengan menggunakan papan pengembangan, anda akan lebih mudah merangkai rangkaian elektronika mikrokontroler dibanding jika anda memulai merakit ATmega328 dari awal di *breadboard*. Tabel 2.1 merupakan spesifikasi arduino UNO REV3.



Gambar 2. 2 Hardware Arduino Uno

Sumber: <https://www.kitronik.co.uk/4622-arduino-uno-main-board-retail.html>

Gambar 2.2 merupakan tampilan fisik dari arduino uno. Untuk mengetahui spesifikasi lebih lanjut mengenai Arduino Uno R3 dapat dilihat pada Tabel 2.1

Tabel 2. 1 Spesifikasi Arduino Uno

Mikrokontroler	ATmega 328P
<i>Operating Voltage</i>	5 V
<i>Input Voltage (recommended)</i>	7 – 12 V
<i>Input Voltage (limit)</i>	6 – 20 V
<i>Digital I/O Pins</i>	16 (6 diantaranya menyediakan PWM)
<i>Analog Input Pins</i>	6
<i>DC Current per I/O Pin</i>	20 mA
<i>DC Current for 3,3V Pin</i>	50 mA
<i>Flash Memory</i>	32 KB (ATmega328P) (0.5 KB sebagai <i>bootloader</i>)
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
<i>USB Host Chip</i>	MAX3421E
<i>Length</i>	68,6 mm
<i>Width</i>	53,4 mm
<i>Weight</i>	25 g

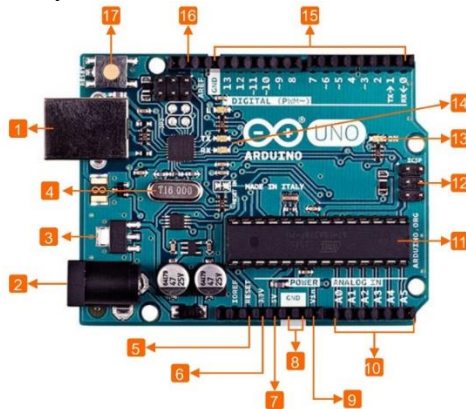
Untuk memprogram arduino juga harus dilakukan beberapa tahapan sebagai berikut :

1. *Setting Board Arduino*

Dalam pemrograman *software* arduino harus di *setting* terlebih dahulu *board* arduino agar penggunaan arduino cocok. Dalam purwarupa kali ini arduino menggunakan arduino UNO REV3. Untuk *setting board* arduino bisa masuk ke *tools – board* – setelah itu pilihlah *board* arduino yang sesuai.

2. *Setting Serial*

Serial ini merupakan kabel arduino yang dihubungkan kepada komputer atau laptop. *Serial* ini mempunyai dua fungsi yang bisa digunakan. Pertama *serial port* digunakan untuk men-*download* program dari arduino yang kedua *serial* digunakan sebagai komunikasi *serial* pada arduino dengan komputer.



Gambar 2. 3 Fungsi Port Arduino

Sumber: <http://fungkynotes.blogspot.com/2018/04/bagian-dan-fungsi-pada-board-arduino.html>

Berdasarkan pada Gambar 2.3, akan diurutkan bagian-bagian dari board Arduino beserta fungsi-fungsinya yaitu sebagai berikut:

1. USB Soket/Power USB

USB Soket/Power USB digunakan untuk memberikan catu daya ke Papan Arduino menggunakan kabel USB dari komputer. Selain menjadi port catu daya, USB juga memiliki berfungsi untuk:

- Memuat program dari komputer ke dalam board Arduin
- Komunikasi serial antara papan Arduino dan komputer begitu juga sebaliknya.

Pada versi lebih lama Arduino terdapat sambungan SV1 Sambungan atau jumper untuk memilih sumber daya yang digunakan, apakah dari sumber eksternal atau menggunakan USB. Sambungan ini tidak diperlukan lagi pada papan Arduino versi terakhir karena pemilihan sumber daya eksternal atau USB dilakukan secara otomatis.

2. Power (*Barrel Jack*)

Papan Arduino dapat juga diberikan colokan catu daya secara langsung dari sumber daya AC dengan menghubungkannya ke Barrel Jack yang tersedia. Tegangan maksimal yang dapat diberikan kepada Arduino maksimal 12volt dengan range arus maksimal 2A (Agar regulator tidak panas).

3. Voltage Regulator

Fungsi dari voltage regulator adalah untuk mengendalikan atau menurunkan tegangan yang diberikan ke papan Arduino dan menstabilkan tegangan DC yang digunakan oleh prosesor dan elemen-elemen lain.

4. Crystal Oscillator

Kristal (*quartz crystal oscillator*), jika mikrokontroler dianggap sebagai sebuah otak, maka kristal adalah jantung-nya karena komponen ini menghasilkan detak-detak yang dikirim kepada mikrokontroler agar melakukan sebuah operasi untuk setiap detak nya. Kristal ini dipilih yang berdetak 16 juta kali per detik (16MHz). Crystal oscillator membantu Arduino dalam hal yang berhubungan dengan waktu. Bagaimana Arduino menghitung waktu? Jawabannya adalah, dengan menggunakan crystal oscillator. Angka yang tertulis pada bagian atas crystal 16.000H9H berarti bahwa frekuensi dari oscillator tersebut adalah 16.000.000 Hertz atau 16 MHz.

5. 17 Arduino Reset

Kita dapat mereset papan arduino, misalnya memulai program dari awal. Terdapat dua cara untuk mereset Arduino Uno. Pertama, dengan menggunakan reset button (17) pada papan arduino. Kedua, dengan menambahkan reset eksternal ke pin Arduino yang

- berlabel RESET (5). Perhatikan bahwa tombol reset ini bukan untuk menghapus program atau mengosongkan mikrokontroler.
6. 3.3V (6) – Supply 3.3 output volt
 7. 5V (7) – Supply 5 output volt
- Sebagian besar komponen yang digunakan papan Arduino bekerja dengan baik pada tegangan 3.3volt dan 5 volt.
8. GND (8) (Ground) – Ada beberapa pin GND pada Arduino, salah satunya dapat digunakan untuk menghubungkan ground rangkaian.
 9. Vin (9) – Pin ini juga dapat digunakan untuk memberi daya ke papan Arduino dari sumber daya eksternal, seperti sumber daya AC.
 10. Analog pins
- Papan Arduino Uno memiliki enam pin input analog A0 sampai A5. Pin-pin ini dapat membaca tegangan dan sinyal yang dihasilkan oleh sensor analog seperti sensor kelembaban atau temperatur dan mengubahnya menjadi nilai digital yang dapat dibaca oleh mikroprosesor. Program dapat membaca nilai sebuah pin input antara 0 – 1023, dimana hal itu mewakili nilai tegangan 0 – 5V.
11. Main microcontroller
- Setiap papan Arduino memiliki Mikrokontroler (11). Kita dapat menganggapnya sebagai otak dari papan Arduino. IC (integrated circuit) utama pada Arduino sedikit berbeda antara papan arduino yang satu dengan yang lainnya. Mikrokontroler yang sering digunakan adalah ATMEL. Kita harus mengetahui IC apa yang dimiliki oleh suatu papan Arduino sebelum memulai memprogram arduino melalui Arduino IDE. Informasi tentang IC terdapat pada bagian atas IC. Untuk mengetahui konstruksi detail dari suatu IC, kita dapat melihat lembar data dari IC yang bersangkutan.
12. 12 ICSP pin
- Kebanyakan, ICSP (12) adalah AVR, suatu programming header kecil untuk Arduino yang berisi MOSI, MISO, SCK, RESET, VCC, dan GND. Hal ini sering dirujuk sebagai SPI (Serial Peripheral Interface), yang dapat dipertimbangkan sebagai “expansion” dari output. Sebenarnya, kita memasang perangkat output ke master bus SPI.
- In-Circuit Serial Programming (ICSP)Port ICSP memungkinkan pengguna untuk memprogram microcontroller secara langsung, tanpa melalui bootloader. Umumnya pengguna Arduino tidak

melakukan ini sehingga ICSP tidak terlalu dipakai walaupun disediakan.

Power LED indicator

LED ini harus menyala jika menghubungkan Arduino ke sumber daya. Jika LED tidak menyala, maka terdapat sesuatu yang salah dengan sambungannya.

13. TX dan RX LEDs

Pada papan Arduino, kita akan menemukan label: TX (transmit) dan RX (receive). TX dan RX muncul di dua tempat pada papan Arduino Uni. Pertama, di pin digital 0 dan 1, Untuk menunjukkan pin yang bertanggung jawab untuk komunikasi serial. Kedua, TX dan RX led (13). TX led akan berkedip dengan kecepatan yang berbeda saat mengirim data serial. Kecepatan kedip tergantung pada baud rate yang digunakan oleh papan arduino. RX berkedip selama menerima proses.

14. Digital I/O

Papan Arduino Uno memiliki 14 pin I/O digital (15), 6 pin output menyediakan PWM (Pulse Width Modulation). Pin-pin ini dapat dikonfigurasi sebagai pin digital input untuk membaca nilai logika (0 atau 1) atau sebagai pin digital output untuk mengendalikan modul-modul seperti LED, relay, dan lain-lain. Pin yang berlabel “~” dapat digunakan untuk membangkitkan PWM.

15. AREF

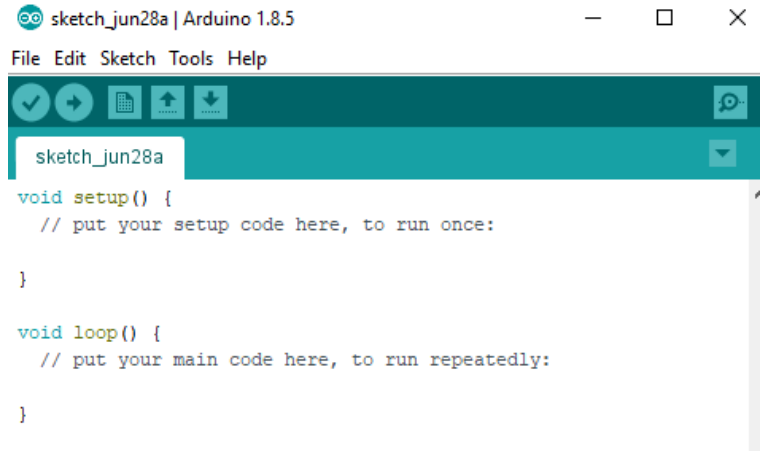
AREF merupakan singkatan dari Analog Reference. AREF kadang-kadang digunakan untuk mengatur tegangan referensi eksternal (antar 0 dan 5 Volts) sebagai batas atas untuk pin input analog input.

2.3.2 Bagian Software

Arduino IDE adalah perangkat lunak yang digunakan untuk memprogram, monitoring dan *debugging* mikrokontroler Arduino. IDE itu merupakan kependekan dari *Integrated Development Environment*, atau secara bahasa mudahnya merupakan lingkungan terintegrasi yang digunakan untuk melakukan pengembangan. Disebut sebagai lingkungan karena melalui *software* inilah Arduino dilakukan pemrograman untuk melakukan fungsi-fungsi yang dinamakan melalui sintaks pemrograman. Arduino menggunakan bahasa pemrograman sendiri yang menyerupai bahasa C.

Bahasa pemrograman Arduino (*Sketch*) sudah dilakukan perubahan untuk memudahkan pemula dalam melakukan pemrograman

dari bahasa aslinya. Sebelum dijual ke pasaran, IC mikrokontroler Arduino telah ditanamkan suatu program bernama *Bootlader* yang berfungsi sebagai penengah antara *compiler* Arduino dengan mikrokontroler

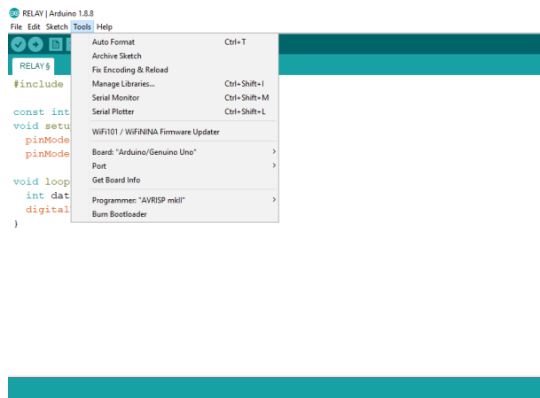


Gambar 2. 4 *Software* Arduino

Pada Gambar 2.3 merupakan tampilan awal untuk membuat program pada *software* Arduino IDE.

2.3.3 Setting Serial

Setting serial bisa masuk *tools – serial* - lali pilih COM yang sesuai dengan arduino yang terpasang. Untuk lebih jelasnya dapat dilihat pada Gambar 2.4

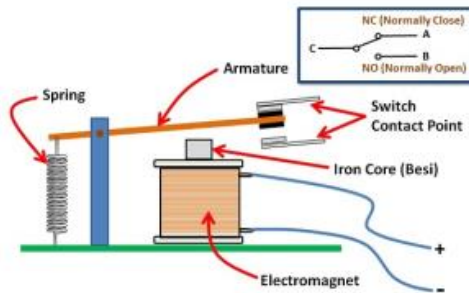


Gambar 2. 5 Setting Serial Port

Apabila *program* tidak dapat di *download* karena *serial port*, maka cek terlebih dahulu *serial* yang benar pada *device manager*. Lalu dalam *software* arduino untuk memilih *serial port*nya samakan dengan *serial port* untuk arduino dalam *device manager* tersebut. Untuk masuk ke *device manager* dapat masuk start windows – lalu ketika *device manager* klik dua kali dan masuk ke dan COM.

2.4 Relay

Relay yang ada pada Gambar 2.5 adalah Saklar (*switch*) yang dioperasikan secara listrik dan merupakan komponen *electromechanical* (elektromekanikal) yang terdiri dari 2 bagian utama yakni elektromagnet (*coil*) dan mekanikal (seperangkat kontak saklar/*switch*). Relay menggunakan prinsip elektromagnetik untuk menggerakkan kontak saklar sehingga dengan arus listrik yang kecil (*low power*) dapat menghantarkan listrik yang bertegangan lebih tinggi. Relay pada Proyek Akhir ini digunakan sebagai penggerak atau kontak yang diperintah oleh arduino yang nantinya akan menggerakkan *valve* atau pompa pada kondisi tertentu. Relay menggunakan prinsip elektromagnetik untuk menggerakkan kontak saklar sehingga dengan arus listrik yang kecil (*low power*) dapat menghantarkan listrik yang bertegangan lebih tinggi.



Gambar 2. 6 Bagian dan Prinsip Kerja dari Relay

Sumber: <http://www.immersa-lab.com/pengertian-relay-fungsi-dan-cara-kerja-relay.htm>

Seperti pada Gambar 2.6 relay memiliki fungsi sebagai saklar elektrik, namun jika di aplikasikan ke dalam rangkaian elektronika, relay memiliki beberapa fungsi yang cukup unik. Berikut beberapa fungsi saat di aplikasikan ke dalam sebuah rangkaian elektronika. Mengendalikan sirkuit tegangan tinggi dengan menggunakan bantuan signal tegangan rendah. Menjalankan logic function atau fungsi logika. Memberikan time delay function atau fungsi penundaan waktu. Melindungi motor atau komponen lainnya dari korsleting atau kelebihan tegangan. Cara Kerja Rela Setelah mengetahui pengertian serta fungsi dari relay, anda juga harus mengetahui cara kerja atau prinsip kerja dari relay. Namun sebelumnya anda perlu mengetahui bahwa pada sebuah relay terdapat 4 bagian penting yaitu electromagnet (coil), Armature, Switch Contact Point (saklar) dan spring. Untuk lebih jelasnya silahkan lihat gambar di bawah ini

2.5 Thermistor NTC 3950

Thermistor adalah sebuah sensor suhu yang memiliki nilai resistansi yang bergantung pada suhu. *Thermistor* sensidri merupakan singkatan dari *Thermal Resistor*. *Thermistor* terdiri dari dua jenis yaitu PTC(*Positive Temperature Coefficient*) dan NTC(*Negative Temperature Coefficient*). *Thermistor* jenis PTC memiliki nilai resistansi yang berbanding lurus dengan kenaikan suhu. Artinya nilai resistansi dari PTC akan naik jika suhu yang diukur naik. *Thermistor* jenis NTC memiliki nilai resistensi yang berbanding terbalik dengan

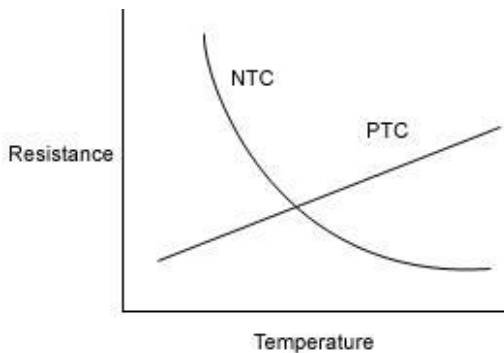
kenaikan suhu. Artinya nilai resistansi dari NTC akan turun jika suhu yang diukur naik. Pada proyek akhir ini menggunakan sensor thermistor NTC 3950 seperti pada Gambar 2.6 [4]



Gambar 2. 7 Sensor Thermistor NTC 3950

Sumber : <https://circuit.rocks/thermistor-ntc-3950-100k.html>

Kemampuan pengukuran suhu oleh *thermistor* berbeda-beda sesuai dengan produsen dari *thermistor*. Namun pada umumnya kemampuan *thermistor* berkisar antara -90°C sampai 130°C . Penggunaan *thermistor* dalam kehidupan sehari-hari banyak dijumpai di detektor kebakaran, sensor suhu untuk kulkas, sensor suhu untuk *charging* baterai, sensor suhu untuk inkubator, dan masih banyak lagi yang berkaitan dengan pengukuran suhu.



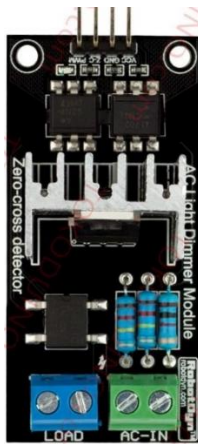
Gambar 2. 8 Ilustrasi Perubahan Resistansi PTC dan NTC

2.6 Dimmer Ac light 220v

Pada prinsipnya *dimmer* mereduksi arus cahaya yang dikeluarkan oleh lampu dengan mengtur daya untuk nyala kipas, prinsip pengaturan yang di gunakan pada rangkaian *dimmer* biasanya berupa pengaturan tegangan, pengaturan arus, pengaturan pulse width modulation PWM 8 bit dengan nilai keluaran 0-255 yang mewakili dutycycle 0-100%. Dimmer awalnya hanya digunakan untuk mengatur pencahayaan lampu. Namun saat ini dimmer sudah memiliki banyak kegunaan lain seperti mengatur kecepatan kipas angin, exhaust fan, ceiling fan, blower, motor listrik, dinamo mesin jahit, dinamo mesin cuci, gerinda tangan, gerinda duduk, bor listrik (drill), mixer, blender, mengatur pencahayaan lampu pijar, lampu edison, lampu led khusus yg dimmable, mengontrol panas heater, dan kontrol input daya equipment lain.

Dimmer AC ini sangat aman digunakan. jadi tidak perlu khawatir alat elektronik anda akan rusak ketika menggunakan dimmer AC ini. Dimmer ini dilengkapi dengan heatsink yang besar, lebih kuat, sehingga dimmer menjadi lebih awet. Mampu digunakan terus menerus tanpa dimatikan. Pada alat ini *dimmer* digunakan sebagai alat untuk mengatur kecepatan kipas pada kondisi 25°C (kecepatan 50%), jika melebihi 39°C maka kecepatan akan penuh. Komponen pada *dimmer* ini menggunakan SCR BT 16, kapasitor 100nF 400 V, dua resistor 1.00.000 Ω dengan toleransi 5 %. Pada alat ini menggunakan 3 *dimmer* untuk mengtur Kipas A. pin yang digunakan menyambungkan *dimmer* dengan *ArduinoUno*

Pada proyek akhir ini menggunakan dimmer untuk mengatur kecepatan kipas seperti pada Gambar 2.8



Gambar 2. 9 Dimmer Ac light 220v

Sumber : https://sea.banggood.com/RobotDyn-AC-Light-Dimmer-Module-For-PWM-Controller-1-Channel-3_3V5V-Logic-AC-50hz-60hz-220V-110V-p-1244351.html?cur_warehouse=CN

Dimmer adalah rangkaian elektronik yang memodifikasi bentuk sinyal ac murni menjadi sinyal terpotong-potong sehingga daya keluaran bisa diatur. Pemotongan sinyal ac ini berguna sebagai peredup lampu, memperlambat motor, mengatur pemanasan dan lainnya. Dimmer yang lebih kompleks menggunakan PWM sebagai pengendalinya. PWM bisa dihasilkan oleh rangkaian SCR, chip/IC PWM atau mikrokontroler. Dimmer PWM ini mampu menghasilkan tingkatan daya yang kecil, sehingga pengontrolan menjadi lebih presisi. Dimmer PWM bisa dikategorikan menjadi dua macam yaitu penyalan berdasarkan titik nol, Penyalan bebas. Penyalan berdasarkan titik nol. Waktu penyalan bergantung pada saat sinyal menyentuh nilai nol. Maka dibutuhkan mekanisme untuk mendeteksi waktu sinyal tersebut bernilai 0. Komponen SCR memiliki sifat forward blocking, forward conduction, dan reverse blocking, maka komponen ini cocok digunakan sebagai dimmer elektronik. Rangkaian dimmer lampu ac yang memanfaatkan sifat SCR. Pada aplikasi dimmer digital, perlintasan titik nol harus dideteksi terlebih dahulu sebelum melakukan penyalan, Pendeteksian nilai nol bisa dilakukan dengan rangkaian zero crossing detector / ZCD berikut. Perancangan dimmer PWM mengikuti kaidah berikut:

1. Waktu penyalan, sinyal bolak-balik (AC) senantiasa bergerak naik dan turun, maka sinyal pengontrol PWM haruslah dimulai saat sinyal AC meninggalkan nilai nol
2. Frekuensi AC, Sinyal pengontrol PWM juga harus memiliki frekuensi tepat dengan frekuensi sinyal AC (listrik)

Kendala membuat dimmer metode zero cross adalah:

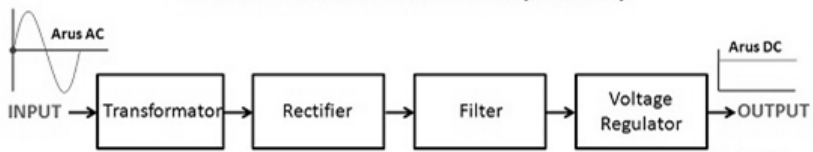
1. frekuensi ac 220v dari penyedia listrik seperti PLN bisa saja berubah-ubah. pada contoh dimmer lampu 220v, intensitas cahaya menjadi tidak konsisten. Untuk mengantisipasi ini sebaiknya dilakukan pengukuran frekuensi secara berkala.
2. Sulit untuk sinkronisasi frekuensi jala listrik dengan frekuensi pengontrol, karena keterbatasan perhitungan digital, misalnya faktor pergeseran frekuensi akibat pembagian bilangan yang tidak sempurna. Sebagai contoh perangkat digital akan sulit mencapai frekuensi 50,00019 Hz.
3. Memakai resources mikrokontroller seperti arduino untuk mendeteksi ZCD terus menerus.
4. Jika ada kesalahan setting waktu penyalan (program diinterupsi) maka beban seperti lampu akan berkedip.
5. Pada duty cycle rendah, untuk dimmer lampu akan terlihat flicker karena perbandingan waktu on sangat kecil dibandingkan waktu off.

2.7 Power Supply

Power Supply atau pencatu daya merupakan rangkaian elektronika yang dapat menghasilkan energi listrik atau sebagai sumber energi untuk rangkaian elektronika lainnya. Sumber arus dari *power supply* adalah arus bolak – balik (AC) dari pembangkit listrik yang kemudian diubah menjadi arus searah (DC). Untuk dapat melakukan hal tersebut *power supply* memerlukan perangkat yang bisa mengubah arus AC menjadi DC. Sebuah DC *power supply* pada dasarnya memiliki 4 bagian utama agar dapat menghasilkan arus DC yang stabil. Keempat bagian utama tersebut diantaranya adalah Transformator, *rectifier*, *filter* dan *voltage regulator* [5].

Berikut adalah diagram blok *power supply* seperti pada Gambar

2.9



Gambar 2. 10 Diagram Blok Power Supply

Sumber: <https://teknikelektronika.com/prinsip-kerja-dc-power-supply-adaptor/>

Gambar 2.10 merupakan *power supply* yang digunakan dalam pembuatan Proyek akhir ini.



Gambar 2. 11 Power supply 12VDC

Sumber: <https://www.indiamart.com/proddetail/multipurpose-12v-5a-60w-sm-ps-dc-power-supply-for-cctv-led-14903412888.html>

Pada Gambar 2.11 Fungsi dari power supply adalah memberikan daya arus listrik ke berbagai komponen atau hardware (baca: pengertian hardware) yang terdapat di dalam casing komputer. Sumber energi listrik yang berasal dari luar masih berbentuk alternating current (AC). Ketika energi listrik masuk ke power supply, maka energi listrik akan dikonversi menjadi bentuk direct current (DC). Daya DC inilah yang kemudian disalurkan ke semua komponen yang ada di dalam casing komputer agar dapat bekerja. Salah satu sisi power supply umumnya tersedia kipas yang berguna untuk membuang udara panas dari dalam casing komputer. Selain itu, pada power supply juga terdapat sebuah port male jenis IEC 60320 C14 yang berfungsi sebagai konektor antara sumber energi listrik dan power supply. Berdasarkan fungsinya, ada dua jenis power supply yang umum digunakan pada komputer dan keduanya memiliki perbedaan yang cukup signifikan. Adapun jenis-jenis power supply adalah sebagai berikut:

1. **Power Supply AT**
Bisa dibilang ini merupakan jenis power supply yang pertama kali digunakan pada jenis-jenis komputer jaman dulu seperti pada tipe Pentium II dan Pentium III pada tahun '97. Sehingga untuk saat ini penggunaan Power Supply AT sudah tidak banyak ditemukan pada jenis perangkat komputer terbaru. Power Supply AT Power Supply AT Ciri-ciri power supply AT: Memiliki 8-12 terminal output yang terhubung pada motherboard computer Umumnya memiliki daya di bawah 250 watt Karena tombol power On dan Off terhubung dengan chasing komputer, maka untuk mematikannya harus dilakukan secara manual pada saat shut down komputer.
2. **Power Supply ATX**
Power Supply ATX merupakan pembaruan dari jenis Power Supply AT. Power supply ini memiliki desain yang lebih kompleks dengan sumber pasokan energi listrik yang lebih efisien. Power Supply
saat ini banyak ditemukan pada jenis komputer generasi terbaru dimana pengoperasiannya bisa dikontrol dengan software yang sudah terinstal dalam komputer. Sehingga bisa diatur dalam beberapa mode seperti mode tidur, mode siaga dan mode saat komputer dimatikan. Ciri-ciri Power Suplly ATX Memiliki 20 sampai 24 terminal output yang berhubungan dengan motherboard Memakai daya yang lebih besar Ketika PC di-shuttdown maka power supply ini akan mati sepenuhnya
Komponen Power Supply Mengacu pada pengertian power supply, perangkat keras ini berfungsi mengubah arus AC menjadi arus DC dan menyalurkannya ke berbagai komponen komputer di dalam chasing. Untuk membentuk tegangan maka dibutuhkan beberapa komponen, adapun komponen power supply adalah sebagai berikut:
 - a. **Transformator**
Ini merupakan komponen di dalam pada Power Supply yang digunakan untuk memindahkan tenaga listrik antar dua rangkaian listrik atau lebih melalui induksi elektromagnetik.
 - b. **Dioda**
Ini adalah gabungan dari dua kata elektroda, yaitu anoda dan katoda. Sifat dari dioda yaitu menghantarkan arus pada tegangan maju dan menghambat arus pada aliran tegangan balik.

- c. Kapasitor
Kapasitor berfungsi sebagai penyempurna penyerahan dari tegangan arus AC ke tegangan arus DC.
- d. Resistor
Resistor adalah perangkat yang membantu Power Supply dalam menurunkan tegangan, membagi tegangan, dan membatasi arus listrik yang masuk, sehingga akan dapat mengontrol perangkat-perangkat keras yang ada pada motherboard.
- e. IC regulator
IC Regulator berfungsi untuk mengatur tegangan pada rangkaian elektronika selalu tetap stabil.
- f. LED
LED pada Power Supply adalah komponen sejenis diode semikonduktor yang memiliki keistimewaan.

Cara Kerja Power Supply Ketika pengguna menyalakan power pada komputer, maka power supply akan melakukan pemeriksaan dan tes sebelum menjalankan sistem komputer. Jika tes berjalan dengan baik maka power supply akan mengirim sinyal (power good) ke mainboard sebagai pertanda bahwa sistem komputer siap untuk beroperasi.

Selanjutnya, power supply atau catu daya akan membagi daya sesuai dengan kapasitas yang diperlukan masing-masing komponen komputer. Selain menyalurkan daya listrik ke komponen komputer, power supply juga menjaga stabilitas arus listrik pada berbagai komponen tersebut.

Dari penjelasan pengertian power supply dan fungsinya di atas, maka komponen ini sama pentingnya seperti CPU pada komputer yang seringkali dianggap sebagai otak komputer. Jika terjadi gangguan pada power supply, maka akan menyebabkan gangguan aliran daya pada komponen-komponen komputer.

2.8 Kipas AC 220V

Kipas pendingin adalah komponen dalam elektronika yang memiliki prinsip kerjanya yaitu mengeluarkan udara panas dan mengganti dengan udara baru kedalam sistem. Sistem pendingin adalah suatu sistem yang akan membantu mempercepat proses pendinginan mesin, karena disaat mesin melakukan beroperasi maka akan menghasilkan kalor yang semakin besar. Jika tidak ada proses pendinginan maka resikonya akan mesin akan cepat overheat. Kipas angin dipergunakan untuk menghasilkan angin. Fungsi yang umum adalah untuk pendingin udara, penyegar udara, ventilasi (exhaust fan), pengering (umumnya memakai komponen penghasil panas). Kipas angin juga ditemukan di mesin penyedot debu dan berbagai ornamen untuk dekorasi ruangan.

Kipas angin secara umum dibedakan atas kipas angin tradisional antara lain kipas angin tangan dan kipas angin listrik yang digerakkan menggunakan tenaga listrik.

Perkembangan kipas angin semakin bervariasi baik dari segi ukuran, penempatan posisi, serta fungsi. Ukuran kipas angin mulai kipas angin mini (Kipas angin listrik yang dipegang tangan menggunakan energi baterai), kipas angin digunakan juga di dalam unit CPU komputer seperti kipas angin untuk mendinginkan processor, kartu grafis, power supply dan casing. Kipas angin tersebut berfungsi untuk menjaga suhu udara agar tidak melewati batas suhu yang ditetapkan. Kipas angin juga dipasang pada alas atau tatakan Laptop untuk menghantarkan udara dan membantu kipas laptop dalam mendinginkan suhu laptop tersebut.

Kipas angin dapat dikontrol kecepatan hembusan dengan 3 cara yaitu menggunakan pemutar, tali penarik serta remote control. Perputaran baling-baling kipas angin dibagi dua yaitu centrifugal (Angin mengalir searah dengan poros kipas) dan Axial (Angin mengalir secara paralel dengan poros kipas)

Kipas angin telah ditemukan beribu tahun silam dan telah digunakan oleh beberapa negara didunia seperti bangsa Mesir, Yunani, Roma, dan China. Pada masa itu kipas angin memiliki berbagai macam dan fungsi. Kipas angin pertama ditemukan sekitar 4000 tahun silam pada sebuah makam raja di Mesir yaitu Raja Tutankhamen yang digali pada tahun 1922.

Fungsi dari kipas angin yang ditemukan di Mesir pada masa itu sebagai alat upacara keagamaan, sehingga sebuah kipas angin merupakan benda yang sakral. Kipas angin juga merupakan lambang kekuatan raja. Ada dua buah kipas angin yang ditemukan di makam raja. Salah satu kipas angin tersebut gagangnya dilapisi oleh emas dan terbuat dari bulu burung unta, sedangkan yang satu lagi dilapisi eboni dengan emas dan batu-batu berharga.

Perkembangan kipas angin juga terdapat di Eropa. Negara pertama di Eropa yang memproduksi kipas angin adalah Italia. Italia memproduksi kipas angin pada tahun 1500. Pada masa itu kipas angin merupakan sebuah komoditi perdagangan yang eksotik dan sangat stylish. Kipas angin sendiri dipandang sebagai simbol kemakmuran dan kelas sosial seseorang.

Perkembangan kipas angin sebagai komoditas fashion sangat populer pada abad ke-16 sampai abad ke-18. Terjadi pergeseran fungsi kipas angin pada awal abad ke 20. Pada masa itu kipas angin sudah tidak lagi sebagai sebuah aksesoris fashion namun menjadi alat periklanan. Sedangkan di Spanyol sendiri, kipas angin menjadi alat untuk mendinginkan udara karena di Spanyol memiliki iklim yang panas.

Kipas angin listrik pertama ditemukan oleh Schuyler Skaats Wheeler pada tahun 1882. Wheeler pertama kali memperkenalkan kipas angin listrik dengan dua buah baling-baling, tanpa ada pelindung apapun dan digerakkan dengan tenaga motor listrik. Perkembangan kipas angin listrik lebih lanjut dikembangkan oleh Philip H. Diehl yang dipatenkan pada tahun 1887. Diehl memperkenalkan kipas angin yang menempel di langit-langit rumah.

Diehl terus mengembangkan temuannya. Pada tahun 1904 Diehl menambahkan sendi split-ball pada kipas angin listriknya. Tiga tahun kemudian, ide ini menjadi dasar penemuan kipas angin yang dapat bergerak ke sana-kemari. Sedangkan pada tahun 1902 Willis Carrier menemukan air conditioning.



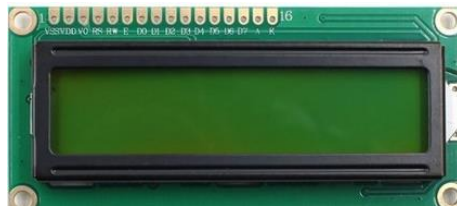
Gambar 2. 12 Kipas AC 220V

Sumber: <https://moedah.com/cooling-fan-ac-220v-merk-kitani/>

Pada proyek akhir ini emnggunakan 2 buah kipas seperti pada Gambar 2.11 yang berfungsi untuk menurunkan suhu dalam panel.

2.9 LCD 16X2

Liquid Crystal Display (LCD) adalah komponen yang dapat menampilkan tulisan. Salah satu jenisnya memiliki dua baris dengan setiap baris terdiri atas enam belas karakter. LCD seperti itu biasa disebut LCD 16x2. Seperti Gambar 2.12



Tabel 2. 2 Spesifikasi LCD 16X 2

No	Nama pin	Diskripsi	Port
1	VCC	+5V	VCC
2	GND	0V	GND
3	VWW	Tegangan kontras LCD	
4	RS	Register select 0	PD7
5	R/W	1= read; 0=write	PD5
6	E	Enable clock	PD6
7	D4	Data Bus 4	PC4
8	D5	Data Bus 5	PC5
9	D6	Data Bus 6	PC6
10	D7	Data Bus 7	PC7
11	anode	Tegangan positif backlight	
12	katode	Tegangan negative backlight	

Pada aplikasinya tidak semua pin pada LCD 20x4 terpakai, melainkan hanya pin-pin yang dihubungkan dengan mikrokontroler yang digunakan saja. Pada aplikasi umumnya RW diberi logika rendah “0”. Bus data terdiri dari 4-bit atau 8-bit. Jika jalur data 4-bit maka yang digunakan ialah DB4 sampai dengan DB7. *Interface* LCD merupakan sebuah paralel bus, dimana hal ini sangat memudahkan dan sangat cepat dalam pembacaan dan penulisan data dari atau ke LCD. Kode ASCII yang ditampilkan sepanjang 8-bit dikirim ke LCD secara 4-bit atau 8 bit pada satu waktu. Jika mode 4-bit yang digunakan, maka 2 nibble data dikirim untuk membuat sepenuhnya 8-bit (pertama dikirim 4-bit MSB lalu 4-bit LSB dengan pulsa *clock* EN setiap nibble-nya). Jalur kontrol EN digunakan untuk memberitahu LCD bahwa mikrokontroler mengirimkan data ke LCD. Untuk mengirim data ke LCD program harus men-set EN ke kondisi *high* “1” dan kemudian mengatur dua jalur kontrol lainnya (RS dan R/W) atau juga mengirimkan data ke jalur data bus.

Saat jalur lainnya sudah siap, EN harus diatur ke “0” dan tunggu beberapa saat (tergantung pada *datasheet* LCD), dan set EN kembali ke *high* “1”. Ketika jalur RS berada dalam kondisi *low* “0”, data yang dikirimkan ke LCD dianggap sebagai sebuah perintah atau instruksi khusus (seperti bersihkan layar, posisi kursor dll). Ketika RS dalam kondisi *high* atau “1”, data yang dikirimkan adalah data ASCII yang akan ditampilkan di layar. Misal, untuk menampilkan huruf “A” pada layar maka RS harus di-set ke “1”. Jalur kontrol R/W harus berada dalam

kondisi *low* (0) saat informasi pada data bus akan dituliskan ke LCD. Apabila R/W berada dalam kondisi *high* “1”, maka program akan melakukan *query* (pembacaan) data dari LCD. Instruksi pembacaan hanya satu, yaitu Get LCD status (membaca status LCD), lainnya merupakan instruksi penulisan. Jadi hampir setiap aplikasi yang menggunakan LCD, R/W selalu di-set ke “0”. Jalur data dapat terdiri 4 atau 8 jalur (tergantung mode yang dipilih pengguna), DB0, DB1, DB2, DB3, DB4, DB5, DB6 dan DB7. Mengirim data secara paralel baik 4-bit atau 8-bit merupakan 2 mode operasi primer.

Mode 8-bit sangat baik digunakan ketika kecepatan menjadi keutamaan dalam sebuah aplikasi dan setidaknya minimal tersedia 11 pin I/O (3 pin untuk kontrol, 8 pin untuk data). Sedangkan mode 4 bit minimal hanya membutuhkan 7-bit (3 pin untuk kontrol, 4 pin untuk data). Bit RS digunakan untuk memilih apakah data atau instruksi yang akan ditransfer antara mikrokontroler dan LCD. Jika bit ini di-set ($RS = 1$), maka byte pada posisi kursor LCD saat itu dapat dibaca atau ditulis. Jika bit ini di-reset ($RS = 0$), merupakan instruksi yang dikirim ke LCD atau status eksekusi dari instruksi terakhir yang dibaca

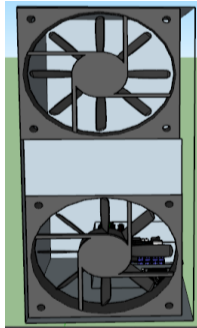
-----Halaman ini sengaja dikosongkan-----

BAB III

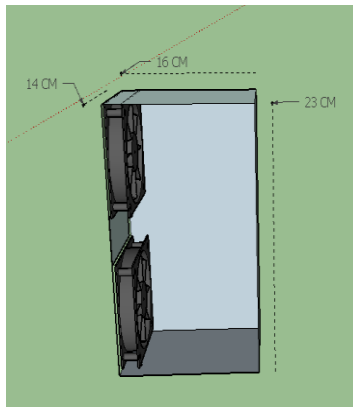
PERANCANGAN SISTEM

Pada bab ini berisi tahapan yang dilakukan dalam perencanaan dan pembuatan alat “Rancang Bangun Sistem Kontrol Suhu pada Ruang Panel Inverter Menggunakan Mikro Kontroler” berfokus pada perancangan sistem otomatisasi dimana secara otomatis pengontrolan suhu ruang panel dan dapat dilihat pada Gambar 3.1 dan Gambar 3.2 merupakan perancangan perangkat mekanik yang terdiri dari *box* panel.

1. Perancangan Perangkat Mekanik

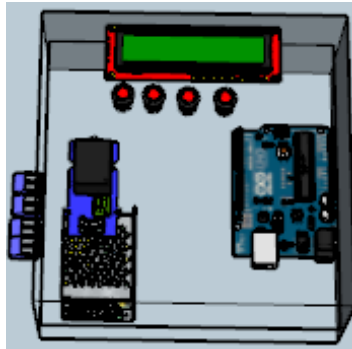


Gambar 3. 1 Perancangan Perangkat Mekanik Tampak Samping



Gambar 3. 2 Perancangan Perangkat Mekanik Tampak Dalam

2. Perancangan Perangkat Elektrik

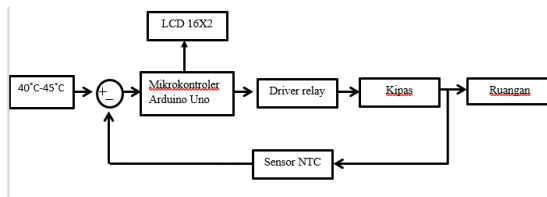


Gambar 3. 3 Perancangan Perangkat Elektrik

Perancangan elektrik seperti pada Gambar 3.3 terdapat power supply, sensor NTC, Ac light Dimmer, Arduino uno, relay, LCD 16 X 2, dan kipas.

3.1 Diagram Sistem Kerja Alat

Dalam pembuatan proyek akhir ini, berikut merupakan diagram fungsional ditunjukkan pada Gambar 3.4 merupakan diagram fungsional sistem kerja NTC.



Gambar 3. 4 Diagram Sistem Kerja sensor NTC

Keterangan Diagram Fungsional sebagai berikut:

1. Arduino Uno, merupakan mikrokontroler yang berfungsi sebagai *interface* dari perangkat elektronik dan dapat menyimpan program di dalamnya.
2. Sensor NTC berfungsi sebagai alat pendeteksi suhu ruang.
3. Driver Relay berfungsi sebagai saklar untuk menyalakan dan mematikan kipas ketika sensor mendeteksi suhu yang ada pada

panel dan mengeluarkan perintah kecepatan kipas sesuai dengan *set poin*.

4. Ruang panel merupakan media untuk sensor bekerja dengan set poin yang diinginkan agar dapat mengurangi suhu ruangan agar komponen tidak terlalu panas.

3.2 Perancangan Perangkat Elektrik

Pada perancangan perangkat elektrik (*Hardware*) pada Proyek Akhir ini meliputi :

1. *Setting Port* Mikrokontroler
2. Perancangan Rangkaian Keseluruhan Alat
3. Rangkaian LCD

3.2.1 *Setting Port* Mikrokontroler

Mikrokontroler pada Proyek Akhir ini digunakan sebagai pengendali utama sistem sistem otomatisasi, mikrokontroler yang digunakan adalah Arduino Uno. Arduino Uno adalah sebuah board mikrokontroler didasarkan pada ATmega 328. Memiliki 14 pin digital diantaranya pin 22 -53 merupakan pin digital input output dan 6 pin merupakan pin PWM dan terdapat 6 pin analog input A0-A5, dengan *clock* sebesar 16 MH osilator kristal, koneksi USB, *jack power*, ICSP header dan tombol *reset*. Adapun *setting port* mikrokontroler yang digunakan untuk Proyek Akhir ini ditunjukkan pada Tabel 3.1

Tabel 3. 1 *Setting Port* Mikrokontroler

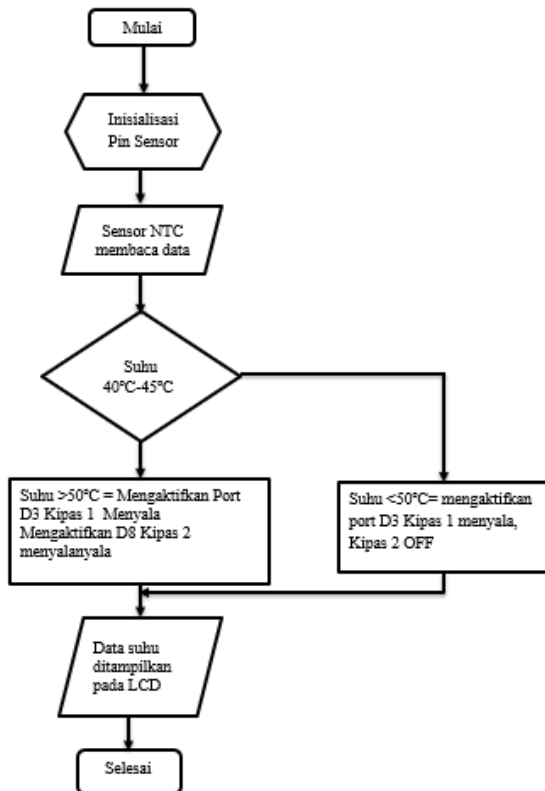
No	Pin Arduino	Keterangan
1	Pin A0	<i>Input Sensor Suhu</i>
2	Pin D3	Input dimer
2	Pin D8	<i>Output sensor kipas 2</i>
11	Pin GND, VCC, SDA, SCL	LCD

3.2.2 Perancangan Rangkaian Keseluruhan Alat

Pada proses perancangan rancangan program sistem kontrol, pemrograman akan dibuat meliputi perancangan perangkat lunak untuk sensor NTC, sebelum melakukan perancangan program terlebih dahulu membuat *flowchart* atau diagram aliran sistem Sensor NTC secara keseluruhan pada Gambar 3.5 menunjukkan algoritma pemrograman utama dengan penjelasan sebagai berikut:

1. *Start* adalah kondisi ketika program dimulai.
2. *Inisialisasi port* Analog A0.

3. Pembacaan sensor ketika terjadi ketika suhu $\leq 40^{\circ}\text{C}$ maka arduino akan mengontrol dimmer untuk menggerakkan kipas 1 dengan kecepatan 50%.
4. Jika Suhu $>40^{\circ}\text{C}$ maka arduino akan mengontrol relay untuk menyalakan kipas 2. jika dengan kondisi diatas suhu panas tetap bertambah hingga melbihi 45°C maka arduino akan mengontrol dimmer untuk menggerakkan kipas 1 dengan kecepatan 100%
5. Kondisi sistem akan terbaca terus terus sehingga terdapat rangkaian *loop* pada flowchart pada Gambar 3.5
6. Hasil pembacaan sensor akan ditampilkan pada layar LCD.



Gambar 3. 5 Flowchart Pembacaan Sensor NTC

Setelah merancang *Flowchart* seperti Gambar 3.5 selanjutnya merancang rangkaian dimana menggunakan sensor NTC yang di dalamnya terdapat *relay*, *dimmer*, kipas dan *power supply* dan dibagi menjadi beberapa bagian perancangan rangkaian, sampai menjadi rangkaian keseluruhan alat.

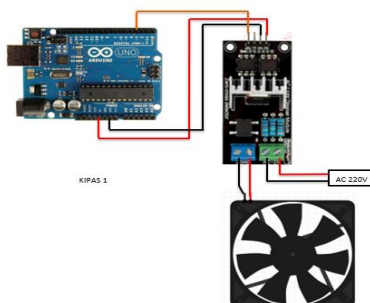
3.2.3 Perancangan Rangkaian Sensor



Gambar 3. 6 Rangkaian Rangkaian sensor

Rancangan rangkaian sensor yang terdapat pada Gambar 3.6 merupakan rancangan yang terdiri dari sensor NTC, Arduino UNO dan resistor yang digabungkan. Pada rancangan ini, sensor berguna untuk mengetahui suhu yang nantinya akan diperintahkan ke Arduino.

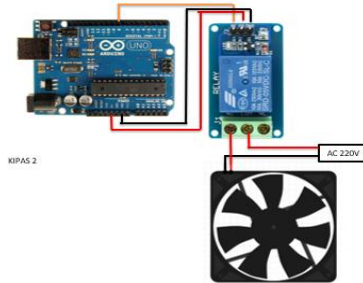
3.2.4 Perancangan Rangkaian Kipas 1 Disambungkan pada Dimmer



Gambar 3. 7 Rangkaian Kipas 1 Disambungkan pada *dimmer*

Gambar 3.7 merupakan perancangan rangkaian kipas 1 disambungkan pada *dimmer* digunakan untuk menyalakan kipas menyala pada suhu $<40^{\circ}\text{C}$ arduino akan mengontrol *dimmer* untuk menggerakkan kipas 1 dengan kecepatan 50%.

3.2.5 Perancangan Rangkaian Rangkaian Kipas 2 Disambungkan pada *relay*

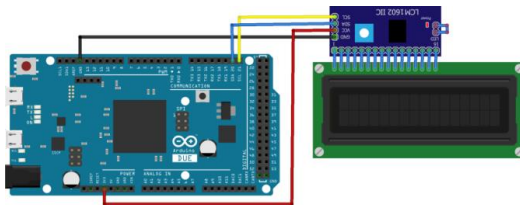


Gambar 3. 8 Rangkaian Rangkaian Kipas 2 Disambungkan pada Relay

Gambar 3.8 merupakan perancangan rangkaian kipas 2 disambungkan pada *relay*, cara kerja sistem yaitu jika Suhu $>40^{\circ}\text{C}$ maka arduino akan mengontrol relay untuk menyalakan kipas 2.

3.2.6 Rangkaian LCD

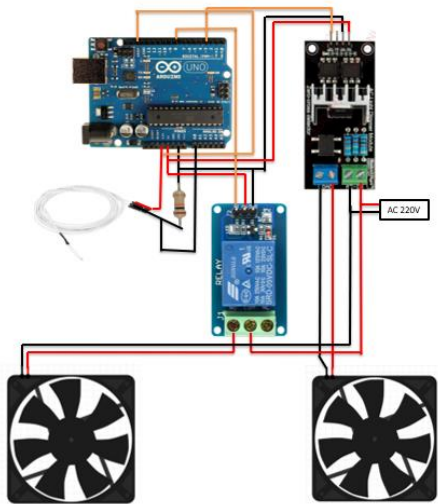
Rangkaian LCD pada Gambar 3.9 berfungsi sebagai media yang menggunakan kristal cair sebagai tampilan suatu data. I2C merupakan standar komunikasi serial dua arah menggunakan dua saluran yang didisain khusus untuk mengirim maupun menerima data. Sistem I2C terdiri dari saluran SCL (*Serial Clock*) dan SDA (*Serial Data*) yang membawa informasi data antara I2C dengan pengontrolnya, serta pin GND disambungkan pada *Ground* Arduino Uno dan VCC pada 5V.



Gambar 3. 9 Rangkaian LCD

3.2.7 Perancangan Rangkaian Keseluruhan Alat

Perancangan keseluruhan alat ini meliputi semua komponen yang digunakan dan digabungkan menjadi satu, antara penggabungan sistem kerja kipas 1 dan sistem kerja kipas 2, yang terdapat pada Gambar 3.10



Gambar 3. 10 Perancangan Rangkaian Keseluruhan Alat

Gambar 3.10 Perancangan Relay pada Rangkaian Keseluruhan Alat Adapun setting port yang digunakan dalam rangkaian Gambar 3.9 dapat dilihat pada Tabel 3.2

Tabel 3. 2 Setting Port pada Rangkaian Keseluruhan Alat

No	Pin Arduino	Keterangan
1.	Pin A0	Input Sensor NTC
2.	Pin D3	Input dimmer
4.	Pin D8	Input relay
5.	Ground	GND
6.	VCC	5V

-----Halaman ini sengaja dikosongkan-----

BAB IV

PENGUJIAN DAN ANALISA

Pada bab ini membahas tentang pengukuran dan analisa sistem yang telah dibuat. Pengujian sistem yang dilakukan merupakan pengujian terhadap perangkat keras dan perangkat lunak dari sistem secara keseluruhan yang telah selesai dibuat untuk memastikan agar komponen komponen sistem yang akan digunakan dapat berfungsi dengan baik, sehingga akan bekerja secara optimal. Pengujian dan analisa pada “Rancang Bangun Sistem Kontrol Suhu pada Ruang Panel Inverter Menggunakan Mikro Kontroler” meliputi

1. Pengujian sensor NTC 3950
2. Pengujian relay dan kipas
3. Pengujian dimmer dan kipas
4. Pengujian *Liquid Crystal Display* (LCD)
5. Pengujian alat keseluruhan



Gambar 4. 1 Hasil keseluruhan alat

4.1 Pengujian Sensor NTC 3950

Pengujian sensor NTC 3950 ini dengan cara memasukkan beberapa air yang digunakan sebagai sample untuk pengujian sesuai dengan suhu masing- masing air.



Gambar 4. 2 Metode Pengujian sensor NTC3950

Pengujian sensor pada Gambar 4.2 *Probe* merah pada AVO diletakkan pada A0 sensor, sedangkan *probe* hitam pada AVO diletakkan pada *gorund* sensor. Tujuan dari pengujian sensor ini untuk mengetahui bahwa sensor dapat bekerja sesuai dengan perintah yang diberikan

4.2 Pengujian relay dan kipas 2

Pengujian relay dan kipas 2 ini berfungsi sebagai pengetesan pada saat sensor mendeteksi suhu lebih dari set poin minum sampai mendekati set poin maksimum (40°C-45°C), dimana dalam pengujian ini terdapat satu buah relay yang disambungkan pada pin D8 arduino.



Gambar 4. 3 Pengujian relay dan kipas 2

Dari Gambar 4.5 yang menunjukkan pengujian relay dan kipas 2 dapat diambil data seperti pada Tabel 4.2

Tabel 4. 1 Pengujian relay dan kipas 2

No	Suhu (°C)	Relay	Kipas 2
1	45	Mati	Mati
2	48	Mati	Mati
3	54	Nyala	Nyala
4	58	Nyala	Nyala
5	60	Nyala	Nyala
6	62	Nyala	Nyala
7	66	Nyala	Nyala
8	70	Nyala	Nyala
9	75	Nyala	Nyala

Dari hasil data pengujian pada Tabel 4.2 dapat dilihat bahwa relay dan kipas menyala sesuai dengan kondisi yang di tentukan, yaitu pada saat suhu lebih dari 45°C.

4.3 Pengujian Dimmer dan kipas 1



Gambar 4. 4 Pengujian Dimmer dan kipas 1

Dari Gambar 4.7 pengujian dilakukan dengan cara menyambungkan input dimmer ke port arduino D3, serta penyambungan vcc dan *ground* serta menghubungkan kipas 1 pada *dimmer*, dari rangkaian tersebut maka didapatkan data seperti pada Tabel 4.3

Tabel 4. 2 Pengujian Dimmer dan kipas 1

No.	Suhu(°C)	Kecepatan(%)	Kipas 1
1.	40	50	Nyala
2.	42	50	Nyala
3.	45	50	Nyala
4.	50	50	Nyala
5.	52	50	Nyala
6.	55	50	Nyala
7.	58	50	Nyala
8.	60	100	Nyala
9.	65	100	Nyala

Data pada Tabel 4.3 merupakan data pengujian dimmer dan kipas 1, kecepatan kipas pada dimmer sudah sesuai dengan kondisi yang dibutuhkan, yaitu jika kurang dari sama dengan 50°C maka kipas akan berputar dengan kecepatan 50%.

4.4 Pengujian *Liquid Crystal Display* (LCD)

Pengujian LCD ini bertujuan untuk mengetahui *Liquid Crystal Display* (LCD) yang digunakan bisa berfungsi dengan baik yang ditunjukkan pada Gambar 4.12. Pada Proyek Akhir ini menggunakan I2C yang dihubungkan dengan LCD. Pin SDA dihubungkan ke pin SDA pada Arduino Uno, sedangkan pin SCL pada dihubungkan dengan pin SCL pada Arduino Uno dan pin Vcc, Gnd dihubungkan pada Volt Arduino dan Gnd Arduino.



Gambar 4. 5 Rangkaian Pengujian LCD

Dapat dilihat pada Gambar 4.12 cara kerja dari I2C adalah sebagai media *transfer* data serial, atau *device* atau komponen yang

mengirim data disebut *transmitter*, sedangkan *device* yang menerimanya disebut *receiver*. *Device* yang mengendalikan operasi *transfer* data disebut master, sedangkan *device* lainnya yang dikendalikan oleh master disebut *slave*. Master *device* harus menghasilkan serial *clock* melalui pin SCL, mengendalikan akses ke BUS serial dan menghasilkan sinyal kendali *START* dan *STOP*, dengan menampilkan kalimat sesuai pada baris dan kolom yang telah diatur pada program sesuai pada Gambar 4.13 dapat dipastikan bahwa rangkaian LCD dan progam pada *Arduino* berhasil dan dapat digunakan dengan baik pada Proyek Akhir ini.



Gambar 4. 6 Tampilan Hasil Pengujian LCD

4.5 Pengujian Alat Keseluruhan

Pengujian keseluruhan dilakukan untuk mengetahui brapa lama waktu yang dibutuhkan untuk menurunkan suhu panas dengan kondisi kipas memasukkan dan mengeluarkan uadara ke dalam panel.

4.5.1 Pengujian penurunan Suhu Panas Dengan Posisi Kipas Mengeluarkan Udara dari Dalam Panel

Proses penurunan suhu dengan posisi kipas mengeluarkan udara dari dalam panel dapat dilihat pada Tabel 4.3 dan proses penurunan suhu dengan posisi kipas memasukkan udara kedalam panel Tabel 4.4 menggunakan *box* besi dipanaskan diatas kompor agar mengetahui pergerakan kenikan suhu pada sensor. Set poin yang digunakan pada percobaan ini yaitu antara 40°C sampai 45°C.

Tabel 4. 3 Pengujian Menurunkan Suhu Panas Dengan Kondisi Kipas Mengeluarkan Udara dari Dalam Panel.

Suhu Minimal (°C)	Suhu Ruang (°C)	Kipas 1	Kipas 2	Waktu (jam;menit;detik)
45	80	Nyala	Nyala	00;06;12
45	75	Nyala	Nyala	00;05;45
45	70	Nyala	Nyala	00;05;38
45	65	Nyala	Nyala	00;05;18
45	60	Nyala	Nyala	00;02;49

45	55	Nyala	Nyala	00;02;38
45	50	Nyala	Nyala	00;01;32
40	45	Nyala	Nyala	00;01;25

Dari Tabel 4.3 merupakan pengujian penurunan suhu panas dengan posisi kipas mengeluarkan udara dari dalam panel suhu yang digunakan yaitu 45°C merupakan suhu normal dalam sebuah panel. Suhu panas 80°C merupakan suhu yang digunakan pada saat simulasi yang dihasilkan sebuah panel yang mengalami kenaikan suhu. Untuk posisi kipas “mengeluarkan udara” dapat diartikan bahwa kipas mengeluarkan udara dari panel untuk mengontrol suhu dengan menggunakan suhu yang keluar dari panel. Dari hasil pengujian diatas dapat di jelaskan bahwa proses penurunan suhu panas maksimal 80°C ke suhu minimal 40°C membutuhkan waktu kurang lebih 6 menit. Sedangkan penurunan suhu panas minimal 45°C ke suhu minimal 40°C membutuhkan waktu 1 menit. Dan dengan pengujian *range* suhu panas 45°C-80°C membutuhkan waktu rata rata 2 – 6 menit.

Tabel 4. 4 Pengujian Menurunkan Suhu Panas Dengan Kondisi Kipas Memasukkan Udara ke Dalam Panel

Suhu Minimal (°C)	Suhu Ruang (°C)	Kipas 1	Kipas 2	Waktu (jam;menit;detik)
45	80	Nyala	Nyala	00;01;58
45	75	Nyala	Nyala	00;01;48
45	70	Nyala	Nyala	00;01;48
45	65	Nyala	Nyala	00;01;48
45	60	Nyala	Nyala	00;01;34
45	55	Nyala	Nyala	00;01;14
45	50	Nyala	Nyala	00;01;21
40	45	Nyala	Nyala	00;00;52

Dari Tabel 4.4 suhu minimal 45°C adalah suhu yang di gunakan sebagai Batasan suhu normal dalam sebuah panel. Suhu panas 80°C merupakan suhu yang digunakan pada saat simulasi yang dihasilkna sebuah panel yang mengalami kenaikan suhu. Untuk posisi kipas “memasukkan udara” dapat diartikan bahwa kipas memasukkan udara dari panel untuk mengontrol suhu dengan menggunakan suhu yang berada di dalam panel. Dari hasil pengujian diatas dapat di jelaskan bahwa proses penurunan suhu panas maksimal 80°C ke suhu minimal 45°C membutuhkan waktu kurang lebih 1menit. Sedangkan penurunan suhu panas minimal 45°C ke suhu miniman 40°C membutuhkan waktu kurang lebih 50 detik. Dan dengan pengujian *range* suhu panas 45°C-80°C membutuhkan waktu rata rata 1 menit.

-----Halaman ini sengaja dikosongkan-----

BAB V

PENUTUP

5.1. Kesimpulan

Cepat lambatnya sensor mendeteksi adanya perubahan suhu dipengaruhi oleh waktu, karena proses pendeteksi sensor cepat yakni rata-rata sekitar 4 detik untuk pembacaan suhu, perubahan suhu dipengaruhi karena adanya perubahan panas yang terjadi akibat pemanasan box besi diatas kompor.

Dari hasil 2 pengujian menggunakan metode mengeluarkan udara dari dalam panel Tabel 4.4 dan memasukkan udara kedalam panel dapat di ambil kesimpulan bahwa metode memasukkan udara ke kipas lebih cepat untuk menurunkan suhu panas dengan rata rata waktu yang di butuhkan kurang lebih 1 sampai 2 menit dari suhu 80°C sampai 45°C.

Sedangkan metode mengeluarkan udara dari dalam panel Tabel 4.3 membutuhkan waktu lebih lama untuk menurunkan suhu panas dengan rata rata waktu yang di butuhkan kurang lebih 5 – 6 menit dari suhu 80°C sampai 45°C.

5.2. Saran

Jika ingin mengembangkan proyek akhir ini maka dapat menambahkan sistem berbasis android dan untuk pemanasnya menggunakan inverter atau pemanas listrik.

-----Halaman ini sengaja dikosongkan-----

DAFTAR PUSTAKA

- [1]. Izah,Aqidatul,"Rancang Bangun Dan Analisis Inverter Full Bridge 1 Fasa Dengan Berbagai Variasi Input Menggunakan Spwm (Sinusoidal Pulse Width Modulation)", **Tugas Akhir**, Jurusan instrumentasi, fakultas vokasi, Institut Teknologi Sepuluh Nopember, Surabaya, 2017.
- [2]. Harmansyah, Agung, "Rancang Bangun Sistem Pengendalian Suhu Pada Tangki Reaktor Plant Fluidized Bed Reactor", **Tugas Akhir**, Jurusan Instrumentasi, Fakultas Vokasi, Institut Teknologi Sepuluh Nopember, Surabaya, 2017.
- [3]. Abdul Kadir, “Panduan Praktis Mempelajari Aplikasi Mikrokontroler dan Pemeogramannya menggunakan Arduino”, Penerbit Andi, Yogyakarta, 2013.
- [4]. Christofer, Gerry., Sujaini, Herry., Irwansyah, M.Azhar., “Rancang Bangun Aplikasi Early Warning Dengan Pemanfaa-tan Pengukuran Suhu Ruangan Berbasiskan Arduino Mega 2560”, **Tugas Akhir**, Program Studi Informatika, Fakultas Teknologi Informasi, Universitas Tanjungpura, Tanjungpura, 2014.
- [5]. Santi, Karina Eka., Ramadhani, Erwin., “AMC (Automatic Money Changer)”, **Tugas Akhir**, Jurusan D3 Teknik Elektro, Fakultas Vokasi, Institut Teknologi Sepuluh Nopember, Surabaya, 2016.
- [6]. Syam, Rafiuddin, “Dasar Dasar Teknik Sensor”, Fakultas Teknik Universitas Hasanuddin, Makassar, bab 2, 2013.
- [7]. Latino, Nico., Hamzah, Amir., “Perbaikan Faktor Daya dengan Implementasi TRIAC Berbasis Mikrokontroler”, Jurnal Teknik Elektro, Jurusan Teknik Elektro, Fakultas Teknik, Uni-versitas Riau, Riau, 2016.
- [8]. Novi,Ridho., “Pendeteksi Tanah Longsor Menggunakan Sensor Cahaya“ Jurnal Tim Darmajaya vol. 01 no.02 Ilmu Komputer, Informatika dan Bisnis Institute Darmajaya Bandar Lampung, 2015.

-----Halaman ini sengaja dikosongkan-----

LAMPIRAN A

A-1. Program Keseluruhan

```
#include <EEPROMex.h>
#include <EEPROMVar.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <SmoothThermistor.h>
#include <Dimmer.h>

LiquidCrystal_I2C lcd(0x3F, 16, 2);
Dimmer dimmer(3, DIMMER_NORMAL);
SmoothThermistor sensor(A0,           // the analog pin to read
from
                                ADC_SIZE_10_BIT, // the ADC size
                                100000,         // the nominal resistance
                                10000,          // the series resistance
                                3950,           // the beta coefficient of the
thermistor
                                25,              // the temperature for nominal
resistance
                                10);            // the number of samples to take for
each measurement

#define upButton 12
#define downButton 11
#define selectButton 10
#define displayButton 9
#define pinFan 8
#define pinLain 7
int nilaiSuhu;
byte menu, kondisiTombol, Min, Max, nilaiToleransi;
boolean konMenuUtama, konMenu, konSubmenu, konMin,
konMax, konToleransi;
String kondisiFan;

void setup() {
    //serial.begin(9600);
```

```

lcd.init();
lcd.backlight();
dimmer.begin();
pinMode(upButton, INPUT_PULLUP);
pinMode(downButton, INPUT_PULLUP);
pinMode(selectButton, INPUT_PULLUP);
pinMode(displayButton, INPUT_PULLUP);
pinMode(pinFan, OUTPUT);
pinMode(pinLain, OUTPUT);
digitalWrite(pinLain, HIGH);
sensor.useAREF(true);
startUpMenu();
konMenuUtama=true;
konMenu=true;
konSubmenu=false;
konMin=false;
konMax=false;
kondisiTombol=1;
//Min=50;
//Max=100;
//nilaiToleransi=12;
//ambil eeprom
Min=EEPROM.readByte(0);
Max=EEPROM.readByte(1);
nilaiToleransi=EEPROM.readByte(2);
menu=1;
}

void menuUtama(){
cekSuhu();
if (konMenuUtama==true){
    nilaiSuhu=sensor.temperature()-nilaiToleransi;
    if (kondisiTombol==1){lcd.clear(); kondisiTombol=0;}
    lcd.setCursor(0,0);
    lcd.print("Suhu: ");
    lcd.setCursor(5,0);
    lcd.print(" ");
    lcd.setCursor(5,0);
    lcd.print(nilaiSuhu);
    lcd.setCursor(8,0);

```

```

        lcd.print((char)223);
        lcd.setCursor(9,0);
        lcd.print("C");
        lcd.setCursor(0,1);
        lcd.print("Fan : ");
        lcd.setCursor(5,1);
        lcd.print(kondisiFan);
        menu=1;
    }
}
//-----Display Menu-----
void startUpMenu(){
    lcd.clear();
    lcd.setCursor(2,0);
    lcd.print(" BISMILLAH");
    lcd.setCursor(2,1);
    lcd.print("LULUS");
    delay(5000);
}
void updateMenu() {
    switch (menu) {
        case 0:
            menu = 3;
            updateMenu();
            break;
        case 1:
            lcd.clear();
            lcd.print(">Set Point");
            lcd.setCursor(0, 1);
            lcd.print(" Toleransi");
            break;
        case 2:
            lcd.clear();
            lcd.print(" Set Point");
            lcd.setCursor(0, 1);
            lcd.print(">Toleransi");
            break;
        case 3:
            lcd.clear();

```

```

        lcd.print(">SaveToEEPROM!");
        lcd.setCursor(0, 1);
        lcd.print(" Set Point");
        break;
case 4:
    menu=1;
    updateMenu();
    break;

case 100:
    menu = 102;
    updateMenu();
    break;
case 101:
    lcd.clear();
    lcd.print(">Max: ");
    lcd.setCursor(13,0);
    lcd.print(Max);
    lcd.setCursor(0,1);
    lcd.print(" Min: ");
    lcd.setCursor(13,1);
    lcd.print(Min);
    break;
case 102:
    lcd.clear();
    lcd.print(" Max: ");
    lcd.setCursor(13,0);
    lcd.print(Max);
    lcd.setCursor(0,1);
    lcd.print(">Min: ");
    lcd.setCursor(13,1);
    lcd.print(Min);
    break;
case 103:
    menu = 101;
    updateMenu();
    break;
    }
}

```

//-----sub aksi-----

```
void executeAction() {
    switch (menu) {
        case 1:
            action1();
            break;
        case 2:
            toleransi();
            break;
        case 3:
            save();
            break;
        case 101:
            aksiMax();
            break;
        case 102:
            aksiMin();
            break;
    }
}

void action1() {
    menu=101;
    konSubmenu=true;
    updateMenu();
}

void toleransi() {
    lcd.clear();
    konToleransi=true;
    lcd.print("Set Toleransi:");
    lcd.setCursor(0,1);
    lcd.print(">");
    lcd.setCursor(1,1);
    lcd.print(nilaiToleransi);
}

void aksiMin() {
    lcd.clear();
    konMin=true;
```

```

    kondisiTombol=1;
    lcd.print("Set Suhu Min: ");
    lcd.setCursor(0,1);
    lcd.print(">");
    lcd.setCursor(1,1);
    lcd.print(Min);
}
void aksiMax() {
    lcd.clear();
    konMax=true;
    kondisiTombol=1;
    lcd.print("Set Suhu Max: ");
    lcd.setCursor(0,1);
    lcd.print(">");
    lcd.setCursor(1,1);
    lcd.print(Max);
}
void save(){
    EEPROM.writeByte(0,Min);
    EEPROM.writeByte(1,Max);
    EEPROM.writeByte(2,nilaiToleransi);
    lcd.clear();
    lcd.setCursor(3,0);
    lcd.print("Data Telah");
    lcd.setCursor(3,1);

    lcd.print("Tersimpan!");
    delay(2000);
}
//-----void baca suhu-----
void cekSuhu(){
    if (nilaiSuhu>=Max){
        digitalWrite(pinFan, HIGH);
        kondisiFan="ON ";
    }
    if (nilaiSuhu>=(Max+5)){
        dimmer.set(100);
        kondisiFan="ON 2";
    }
    if (nilaiSuhu<=Min){

```

```

digitalWrite(pinFan, LOW);
dimmer.set(20);

    kondisiFan="OFF ";
}
}
//-----batas void bacaSuhu-----
void loop() {
    menuUtama();
    if (!digitalRead(downButton)){
        if (konMenuUtama==true){
            konMenuUtama=false;
            kondisiTombol=0;
        }
        if (konMenu==true){
            menu++;
            updateMenu();
        }
        if (konMin==true){
            Min--;
            aksiMin();
        }
        if (konMax==true){
            Max--;
            aksiMax();
        }
        if (konToleransi==true){
            nilaiToleransi--;
            toleransi();
        }
        delay(100);
        while (!digitalRead(downButton));
    }
    if (!digitalRead(upButton)){
        if (konMenuUtama==true){
            konMenuUtama=false;
            kondisiTombol=0;
        }
        if (konMenu==true){

```

```

    menu--;
    updateMenu();
}
if (konMin==true){
    konMenu=false;
    Min++;
    aksiMin();
}
if (konMax==true){
    konMax=false;
    Max++;
    aksiMax();
}
if (konToleransi==true){
    nilaiToleransi++;
    toleransi();
}
delay(100);
while(!digitalRead(upButton));
}
if (!digitalRead(selectButton)){
    if (konMenuUtama==false){
        executeAction();
    }

    if (konMenu==true){
        updateMenu();
        konMenuUtama=false;
        kondisiTombol=0;
    }
    if (konMin==true){
        menu=1;
        konMenu=false;
        konMin=false;
        aksiMin();
    }
    if (konMax==true){
        menu=1;
        konMenu=false;
        aksiMax();
    }
}

```

```

    }
    if (konToleransi==true){
        menu=1;
        konMenu=false;
        toleransi();
    }
    delay(100);
    while (!digitalRead(selectButton));
    }
    if (!digitalRead(displayButton)){
        menu=1;
        konMenuUtama=true;
        konMenu=true;
        konSubmenu=false;
        konMin=false;
        konMax=false;
        konToleransi=false;
        kondisiTombol=1;
        menuUtama();
        delay(100);
    }
}

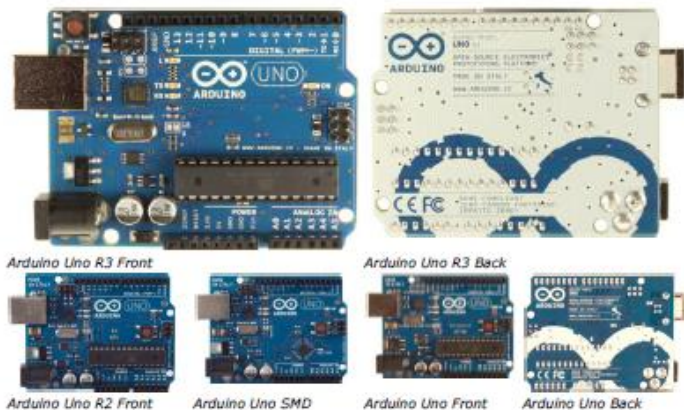
```

-----Halaman ini sengaja dikosongkan-----

LAMPIRAN B

B-1. Datasheet Arduino UNO

Arduino Uno



Overview

The Arduino Uno is a microcontroller board based on the ATmega328 ([datasheet](#)). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

[Revision 2](#) of the Uno board has a resistor pulling the 8U2 HWB line to ground, making it easier to put into [DFU mode](#).

[Revision 3](#) of the board has the following new features:

- 1.0 pinout: added SDA and SCL pins that are near to the AREF pin and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board. In future, shields will be compatible both with the board that use the AVR, which operate with 5V and with the Arduino Due that operate with 3.3V. The second one is a not connected pin, that is reserved for future purposes.
- Stronger RESET circuit.
- Atmega 16U2 replace the 8U2.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the [index of Arduino boards](#).

Summary

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V

Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

Schematic & Reference Design

EAGLE files: [arduino-uno-Rev3-reference-design.zip](#) (NOTE: works with Eagle 6.0 and newer)

Schematic: [arduino-uno-Rev3-schematic.pdf](#)

Note: The Arduino reference design can use an Atmega8, 168, or 328. Current models use an ATmega328, but an Atmega8 is shown in the schematic for reference. The pin configuration is identical on all three processors.

Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

Memory

The ATmega328 has 32 KB (with 0.5 KB used for the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- **PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with the [analogWrite\(\)](#) function.

- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication using the [SPI library](#).
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though it is possible to change the upper end of their range using the AREF pin and the [analogReference\(\)](#) function. Additionally, some pins have specialized functionality:

- **TWI: A4 or SDA pin and A5 or SCL pin.** Support TWI communication using the [Wire library](#).

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with [analogReference\(\)](#).
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

See also the [mapping between Arduino pins and ATmega328 ports](#). The mapping for the ATmega8, 168, and 328 is identical.

Communication

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, [on Windows, a .inf file is required](#). The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Uno's digital pins.

The ATmega328 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the [documentation](#) for details. For SPI communication, use the [SPI library](#).

Programming

The Arduino Uno can be programmed with the Arduino software ([download](#)). Select "Arduino Uno" from the **Tools > Board** menu (according to the microcontroller on your board). For details, see the [reference](#) and [tutorials](#).

The ATmega328 on the Arduino Uno comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference](#), [C header files](#)).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.

The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available. The ATmega16U2/8U2 is loaded with a DFU bootloader, which can be activated by:

- On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2.
- On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode.

You can then use [Atmel's FLIP software](#) (Windows) or the [DFU programmer](#) (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader). See [this user-contributed tutorial](#) for more information.

-----Halaman ini sengaja dikosongkan-----

B-2. Datasheet Sensor NTC

Reprap Hotend Thermistor NTC 3950 100K with 1M Cable



- Standard resistance: R25(25): 1.5K, 2K, 5K, 10K, 20K, 30K, 47K, 50K, 100K, 200K, 500K, etc
- Resistance tolerance: $\pm 1\%$, $\pm 2\%$, $\pm 3\%$
- B-value R25/50: 3435K, 3600K, 3950K, 3990K, 4100K, 4200K, etc
- B-value accuracy: $\pm 1\%$
- Operating temperature: $-40 \sim +300^\circ\text{C}$
- Power dissipation coefficient: $\geq 5 \text{ mW}/^\circ\text{C}$ (in static air)
- Max. rated power: 45mW
- Thermal time constant: ≤ 75 (in static air)
- Temperature coefficient of resistance: $-2 \sim -5\%/^\circ\text{C}$
- It is recommended to use: R25°C = 100K, B25/50 = 3950K $\pm 1\%$

Reprap Hotend Thermistor NTC 3950 100K with 1M Cable

R—T TABLE

R25℃=100K $\Omega \pm 1\%$

B25℃/50℃=3950 $\pm 1\%$

T	Rmax	Rnom	Rmin	T	Rmax	Rnom	Rmin
-30	1005.1971	1733.2000	1663.6657	10	203.3838	199.9900	196.6338
-29	1693.5063	1630.4080	1569.3019	11	193.7916	190.5578	187.3599
-28	1593.9380	1534.4770	1476.9663	12	184.7142	181.6319	178.5938
-27	1500.0930	1444.9030	1390.7495	13	176.1211	173.1822	170.2759
-26	1413.9673	1361.2200	1310.2028	14	167.9835	165.1804	162.4033
-25	1329.1367	1283.0000	1238.3016	15	159.8962	157.6000	155.3211
-24	1252.8144	1209.3270	1167.1953	16	152.6170	150.4253	148.2502
-23	1181.4336	1140.4240	1100.6928	17	145.7160	143.6234	141.5466
-22	1114.6401	1075.9490	1038.4640	18	139.1712	137.1726	135.1891
-21	1052.1085	1015.5890	980.2059	19	132.9622	131.0528	129.1578
-20	990.8617	959.0500	928.1695	20	126.7793	125.2450	123.7170
-19	936.0641	906.0117	876.8390	21	121.1240	119.6582	118.1904
-18	884.6914	856.2883	828.7167	22	115.7568	114.3559	112.9608
-17	836.5067	809.6506	783.5807	23	110.6613	109.3221	107.9884
-16	791.2910	765.8865	741.2257	24	105.6340	104.3415	103.0506
-15	746.7035	724.8000	703.4662	25	101.0000	100.0000	99.0000
-14	706.3711	685.6507	665.4693	26	96.8195	95.8191	94.8195
-13	668.5024	648.8929	629.7934	27	92.8384	91.8392	90.8415
-12	632.9310	614.3649	596.2817	28	89.0074	88.0494	87.0928
-11	599.5024	581.9169	564.7888	29	85.3582	84.4395	83.5222
-10	566.5131	551.4100	536.6581	30	81.9074	81.0000	80.1067
-9	537.0073	522.6908	508.7073	31	78.5700	77.6238	76.6815
-8	509.2437	495.6674	482.4068	32	75.3161	74.4091	73.5058
-7	483.1082	470.2286	457.6486	33	72.2170	71.3472	70.4812
-6	458.4948	446.2714	434.3223	34	69.2643	68.4301	67.5994
-5	434.1442	422.7000	413.4672	35	66.5901	65.6500	64.7167
-4	411.9667	402.0560	392.3459	36	63.8857	62.9838	62.0884
-3	391.0739	381.6658	372.4482	37	61.3075	60.4420	59.5827
-2	371.2832	362.4488	353.6953	38	58.8489	58.0181	57.1923
-1	352.6178	344.3301	336.0142	39	56.5027	55.7060	54.9141
0	334.4393	327.2400	320.1638	40	54.2774	53.5000	52.6316
1	317.8826	311.0397	304.3138	41	52.2133	51.3708	50.5270
2	302.2571	295.7506	289.3553	42	50.1482	49.3291	48.5382
3	287.5046	281.3157	275.2325	43	48.1771	47.3990	46.6304
4	273.5710	267.6820	261.8936	44	46.2952	45.5483	44.8090
5	259.7533	254.8000	249.9157	45	44.5064	43.7800	42.9826
6	247.2985	242.5827	237.9326	46	42.8302	42.0555	41.2905
7	235.5234	231.0321	226.6034	47	41.1535	40.4092	39.6741
8	224.3869	220.1080	215.8887	48	39.5522	38.8269	38.1304

B-3. Datasheet Dimmer

Arduino Controlled Light Dimmer

 by diy_bloke

WAIT!! before you decide to build this, it is good to know that a similar dimmer is available at Aliexpress at cost that is hard to beat (currently 2.70 euro)

WARNING: *Some people try to build this with an optocoupler with zerocrossing coz 'that is better' right? Some are even told in electronics shops it is better to use such an optocoupler. WRONG. This will only work with a random fire optocoupler: NOT igniting at zerocrossing is the principle of this dimmer.*

Switching an AC load with an Arduino is rather simpel: either a mechanical relay or a solid state relay with an optically isolated Triac. (I say Arduino, but if you use an 8051 or PIC16F877A microcontroller, there is stuff for you too here.)

It becomes a bit more tricky if one wants to dim a mains AC lamp with an arduino: just limiting the current through e.g. a transistor is not really possible due to the large power the transistor then will need to dissipate, resulting in much heat and it is also not efficient from an energy use point of view.

Phase cutting

One way of doing it is through phase control with a Triac: the Triac then is fully opened, but only during a part of the sinus AC wave. This is called leading edge cutting.

One could let an Arduino just open the Triac for a number of microseconds, but that has the problem that it is unpredictable during what part of the sinus wave the triac opens and therefore the dimming level is unpredictable. One needs a reference point in the

sinus wave.

For that a zero crossing detector is necessary. This is a circuit that tells the Arduino (or another micro controller) when the sinus-wave goes through zero and therefore gives a defined point on that sinus wave.

Opening the Triac after a number of microseconds delay starting from the zero crossing therefore gives a predictable level of dimming.

Pulse Skip Modulation

Another way of doing this is by Pulse Skip Modulation. With PSM, one or more full cycles (sinuswaves) are transferred to the load and then one or more cycles are not. Though effective, it is not a good way to dim lights as there is a chance for flickering. Though it might be tempting, in PSM one should always allow a full sinuswave to be passed to the load, not a half sinus as in that case the load will be fed factually from DC which is not a good thing for most AC loads. The difference between leading edge cutting and PSM is mainly in the software: in both cases one will need a circuit that detects the zero crossing and that can control a triac.

A circuit that can do this is easy to build: The zero crossing is directly derived from the rectified mains AC lines – via an optocoupler of course- and gives a signal every time the wave goes through zero. Because the sine wave first goes through double phased rectification, the zero-crossing signal is given regardless whether the sinus wave goes up through zero or down through zero. This signal then can be used to trigger an interrupt in the Arduino.

PWM dimming

PWM dimming, as in LEDs is not done frequently

with AC loads for a number of reasons. It is possible though. Check [this instructable to see how](#).

It goes without saying that there needs to be a galvanic separation between the Arduino side of things and anything connected to the mains. For those who do not understand 'galvanic separation' it means 'no metal connections' thus --> opto-couplers. BUT, if you do not understand 'galvanic separation', maybe you should not build this.

The circuit pictured here does just that. The mains 220V voltage is led through two 30k resistors to a bridge rectifier that gives a double phased rectified signal to a 4N25 opto-coupler. The LED in this opto-coupler thus goes low with a frequency of 100Hz and the signal on the collector is going high with a frequency of 100Hz, in line with the sinusoid wave on the mains net. The signal of the 4N25 is fed to an interrupt pin in the Arduino (or other microprocessor). The interrupt routine feeds a signal of a specific length to one of the I/O pins. The I/O pin signal goes back to our circuit and opens the LED and a MOC3021, that triggers the Opto-Thyristor briefly. The LED in series with the MOC3021 indicates if there is any current going through the MOC3021. Mind you though that in dimming operation that light will not be very visible because it is very short lasting. Should you chose to use the triac switch for continuous use, the LED will light up clearly.

Mind you that only regular incandescent lamps are truly suitable for dimming. It will work with a halogen lamp as well, but it will shorten the life span of the halogen lamp. It will not work with any cfl lamps, unless they are specifically stated to be suited for a dimmer. The same goes for LED lamps

NOTE! It is possible that depending on the LED that is used, the steering signal just does not cut it and you may end up with a lamp that just flickers rather than being smoothly regulated. Replacing the LED with a wire bridge will cure that. The LED is not really necessary. increase the 220 ohm resistor to 470 then

coming even close to the PCB. The cooling plate of the Triac is attached to the mains. Do not touch it while in operation. Put it in a proper enclosure/container.

WAIT: Let me just add a stronger warning here: This circuit is safe if it is built and implemented only by people who know what they are doing. If you have no clue or if you are doubting about what you do, chances are you are going to be DEAD! **DO NOT TOUCH WHEN IT IS CONNECTED TO THE GRID**

Materials

Zero-crossing

4N25 €0.25 or H11AA1 or IL250, IL251, IL252, LTV814 (see text in the next step)

Resistor 10k €0.10

bridge rectifier 400 Volt €0.30

2x 30 k resistor 1/2 Watt (resistors will probably dissipate 400mW max each €0.30

1 connector €0.20

5.1 Volt zenerdiode (optional)

Lamp driver

LED (Note: you can replace the LED with a wire bridge as the LED may sometimes cause the lamp to flicker rather than to regulate smoothly)

MOC3021 If you chose another type, **make sure it has NO zero-crossing detection**, I can't stress this enough DO NOT use e.g. a MOC3042

Resistor 220 Ohm €0.10 (I actually used a 330 Ohm and that worked fine)

Resistor 470 Ohm-1k (I ended up using a 560 Ohm and that worked well)

TRIAC TIC206 €1.20 or BR136 €0.50

1 connector €0.20

Other

Piece of PCB 6x3cm
electric wiring

That is about €3 in parts

STOP: This circuit is attached to a 110-220 Voltage. Do not build this if you are not confident about what you are doing. Unplug it before

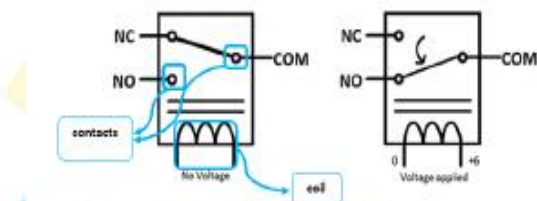
B-4. Datasheet Relay



RELAY MODULES

RELAY WORKING IDEA

Relays consist of three pins normally open pin , normally closed pin, common pin and coil. When coil power on magntic field is generated the contacts connected to each other.

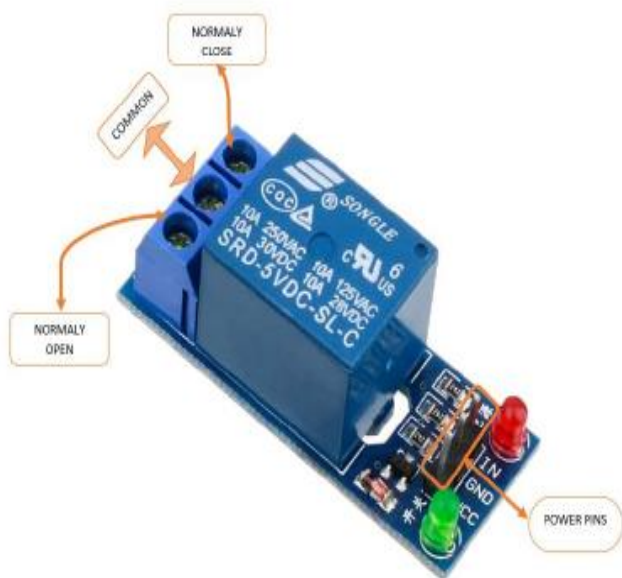


Relay modules 1-channel features

- Contact current 10A and 250V AC or 30V DC.
- Each channel has indication LED.
- Coil voltage 12V per channel.
- Kit operating voltage 5-12 V
- Input signal 3-5 V for each channel.
- Three pins for normally open and closed for each channel.

How to connect relay module with Arduino

As shown in relay working idea it depends on magnetic field generated from the coil so there is power isolation between the coil and the switching pins so coils can be easily powered from Arduino by connecting VCC and GND pins from Arduino kit to the relay module kit after that we choose Arduino output pins depending on the number of relays needed in project designed and set these pins to output and make it out high (5 V) to control the coil that allow controlling of switching process.

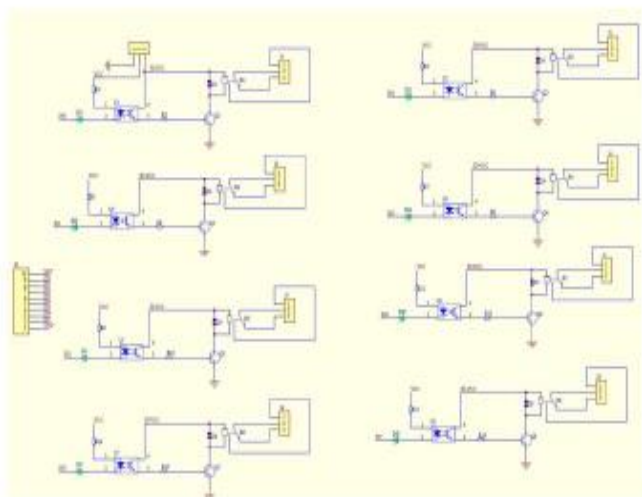


NOTE : whatever was the relay channels number the pinconfiguration is the same for every channel except the power pins (VCC and GND) are for the board itself. The input signal (IN) pin for every relay.

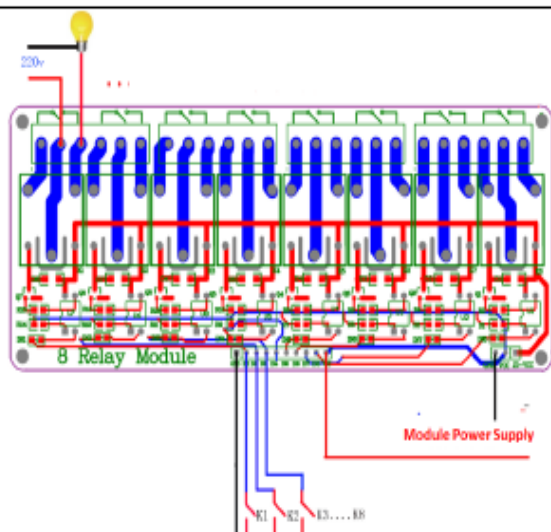
VCC and RY-VCC are also the power supply of the relay module. When you need to drive a large power load, you can take the jumper cap off and connect an extra power to RY-VCC to supply the relay, connect VCC to 5V of the MCU board to supply input signals.

NOTES: If you want complete optical isolation, connect "Vcc" to Arduino +5 volts but do NOT connect Arduino Ground. Remove the Vcc to JD-Vcc jumper. Connect a separate +5 supply to "JD-Vcc" and board Gnd. This will supply power to the transistor drivers and relay coils.

If relay isolation is enough for your application, connect Arduino +5 and Gnd, and leave Vcc to JD-Vcc jumper in place.



3 Channel Relay Module Schematic



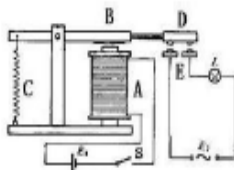
It is sometimes possible to use this relay boards with 3.3V signals, if the JD-VCC (Relay Power) is provided from a +5V supply and the VCC to JD-VCC jumper is removed. That 5V relay supply could be totally isolated from the 3.3V device, or have a common ground if opto-isolation is not needed. If used with isolated 3.3V signals, VCC (To the input of the opto-isolator, next to the IN pins) should be connected to the 3.3V device's +3.3V supply.

NOTE: Some Raspberry-Pi users have found that some relays are reliable and others do not actuate sometimes. It may be necessary to change the value of R1 from 1000 ohms to something like 220 ohms, or supply +5V to the VCC connection.

NOTE: The digital inputs from Arduino are Active LOW: The relay actuates and LED lights when the input pin is LOW, and turns off on HIGH.

Operating Principle:

See the picture below: A is an electromagnet, B armature, C spring, D moving contact, and E fixed contacts. There are two fixed contacts, a normally closed one and a normally open one. When the coil is not energized, the normally open contact is the one that is off, while the normally closed one is the other that is on.



Supply voltage to the coil and some currents will pass through the coil thus generating the electromagnetic effect. So the armature overcomes the tension of the spring and is attracted to the core, thus closing the moving contact of the armature and the normally open (NO) contact or you may say releasing the former and the normally closed (NC) contact. After the coil is de-energized, the electromagnetic force disappears and the armature moves back to the original position, releasing the moving contact and normally closed contact. The closing and releasing of the contacts results in power on and off of the circuit.

Input:

VCC : Connected to positive supply voltage (supply power according to relay voltage)

GND : Connected to supply ground.

IN1: Signal triggering terminal 1 of relay module

IN2: Signal triggering terminal 2 of relay module

IN3: Signal triggering terminal 3 of relay module

IN4: Signal triggering terminal 4 of relay module

IN5: Signal triggering terminal 5 of relay module

IN6: Signal triggering terminal 6 of relay module

IN7: Signal triggering terminal 7 of relay module

IN8: Signal triggering terminal 8 of relay module

Output:

Each module of the relay has one NC (normally close), one NO (normally open) and one COM (Common) terminal. So there are 8 NC, 8 NO and 8 COM of the channel relay in total. NC stands for the normal close port contact and the state without power. NO stands for the normal open port contact and the state with power. COM means the common port. You can choose NC port or NO port according to whether power or not.

Testing Setup:

When a low level is supplied to signal terminal of the 8-channel relay, the LED at the output terminal will light up. Otherwise, it will turn off. If a periodic high and low level is supplied to the signal terminal, you can see the LED will cycle between on and off.

Arduino Application Examples for 4-Channel Relay Board:

Step 1:

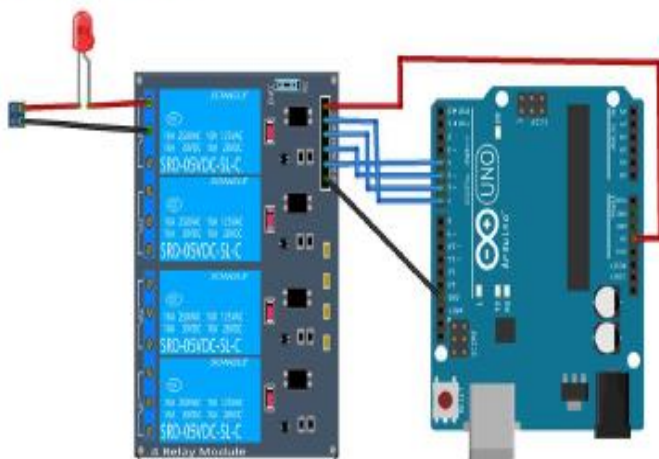
Connect the signal terminal DN1, DN2, DN3 & DN4 of 4-channel relay to digital pin 4, 5, 6, 7 of the Arduino Uno or ATmega2560 board, and connect an LED at the output terminal.

DN1> 4; DN2> 5; DN3>6; DN4>7

Step 2:

Upload the sketch "4 Channel Relay Demo " to the Arduino Uno or ATmega2560 board. Then you can see the LED cycle between on and off

The actual figure is shown below:



Arduino Sketch: 4 Channel Relay Demo

```
/*
*****
Name: 4 channel_relay
Description: control the 4 channel relay module to ON or OFF
Website: www.handsontec.com
Email: techsupport@handsontec.com
*****
//the relays connect to

int RelayControl1 = 4; // Digital Arduino Pin used to control the motor
int RelayControl2 = 5;
int RelayControl3 = 6;
int RelayControl4 = 7;

void setup()
{
  Serial.begin(9600);
  pinMode(RelayControl1, OUTPUT);
  pinMode(RelayControl2, OUTPUT);
  pinMode(RelayControl3, OUTPUT);
  pinMode(RelayControl4, OUTPUT);
}
```

```

}

void loop()
{
    digitalWrite(RelayControl1,HIGH);// NOL and COM1 Connected (LED on)
    delay(1000);
    digitalWrite(RelayControl1,LOW);// NOL and COM1 disconnected (LED off)
    delay(1000);
    digitalWrite(RelayControl2,HIGH);
    delay(1000);
    digitalWrite(RelayControl2,LOW);
    delay(1000);
    digitalWrite(RelayControl3,HIGH);
    delay(1000);
    digitalWrite(RelayControl3,LOW);
    delay(1000);
    digitalWrite(RelayControl4,HIGH);
    delay(1000);
    digitalWrite(RelayControl4,LOW);
    delay(1000);
}

```



Handsontec.com

We have the parts for your ideas

HandsOn Technology provides a multimedia and interactive platform for everyone interested in electronics. From beginner to diehard, from student to lecturer. Information, education, inspiration and entertainment. Analog and digital, practical and theoretical; software and hardware.



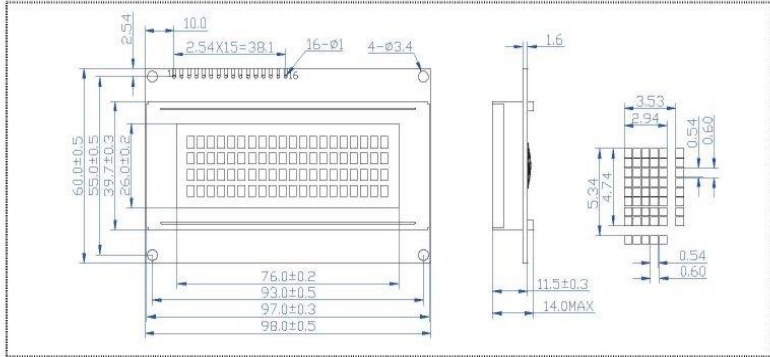
HandsOn Technology support Open Source Hardware (OSHW) Development Platform.

Learn : Design : Share

www.handsontec.com



B-5. Datasheet LCD



MECHANICAL DATA						PIN CONNECTIONS								
Item	Nominal Dimensions			Unit		PIN	Symbol	Level	Function					
Module Size(W×H×T)	98.0×60.0×14.0			mm		1	VSS	—	GND(0V)					
Viewing Area(W×H)	76.0×26.0			mm		2	VDD	—	Supply Voltage for Logic(+5V)					
Character Size(W×H)	2.94×4.74			mm		3	V0	—	Power supply for LCD					
Dot Size(W×H)	0.54×0.54			mm		4	RS	H/L	H: Data; L: Instruction Code					
						5	R/W	H/L	H: Read; L: Write					
						6	E	H/L	Enable Signal					
						7	DB0	H/L	Data Bus Line					
						8	DB1	H/L						
						9	DB2	H/L						
						10	DB3	H/L						
						11	DB4	H/L						
						12	DB5	H/L						
						13	DB6	H/L						
						14	DB7	H/L						
						15	A	—	Backlight Power(+5V)					
						16	K	—	Backlight Power(0V)					
ELECTRICAL CHARACTERISTICS														
Item	Symbol	Min	Type	Max	Unit									
Input Hight Voltage	V _{IH}	2.2	—	VDD	V									
Input Low Voltage	V _{IL}	0	—	0.6 V	V									
Output Hight Voltage	V _{OH}	2.4	—	VDD	V									
Output Low Voltage	V _{OL}	0	—	0.4 V	V									
BLOCK DIAGRAM						POWER SUPPLY								

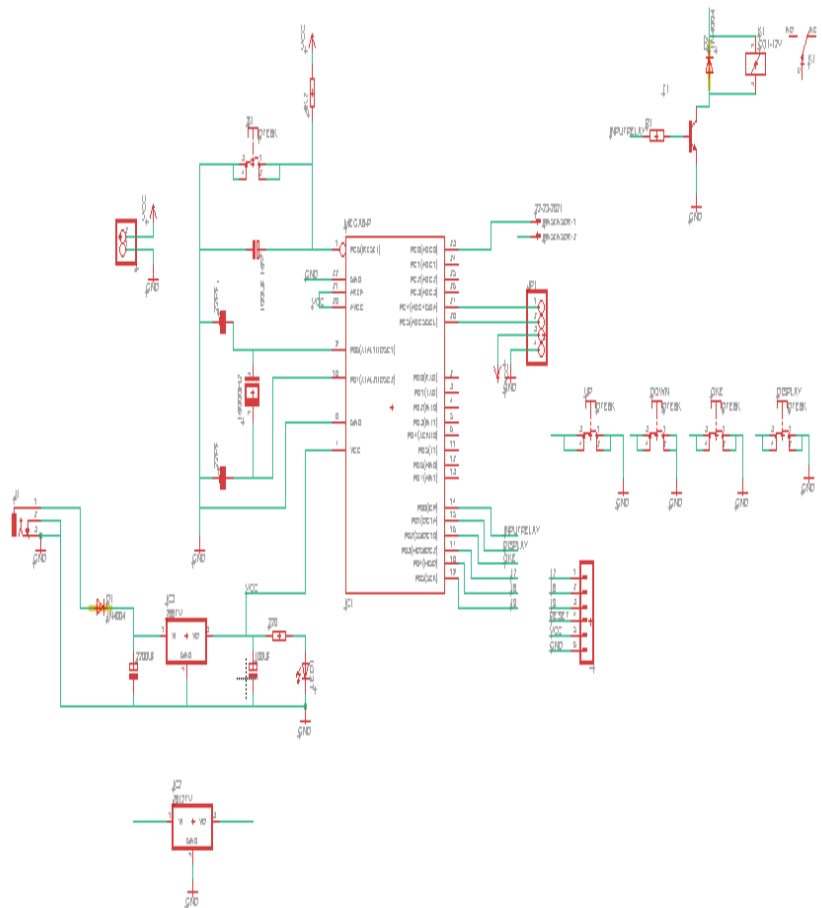
-----Halaman ini sengaja dikosongkan-----

B-6. Pengujian Menggunakan Stopwatch



-----Halaman ini sengaja dikosongkan-----

C-1. Skematik Eagle Rangkaian Keseluruhan



-----Halaman ini sengaja dikosongkan-----

RIWAYAT PENULIS



Nama : Andy Rahma Septianto
TTL : Malang, 20 September 1997
Jenis Kelamin : Laki - laki
Agama : Islam
Alamat : PERUM ITS JL. Teknik
Komputer Blok U No 89
Telp/HP : 0818526867832
E-mail : *andyrasma71@gmail.com*

RIWAYAT PENDIDIKAN:

1. 2006-2012 : SDN KALISARI I 242
2. 2012-2014 : MTsN Surabaya 1
3. 2014-2016 : SMK PGRI 4 Surabaya
4. 2016-sekarang : D3 teknik elektro kerjasama disnaker, Departemen Teknik Elektro Otomasi, Fakultas Vokasi, Institut Teknologi Sepuluh Nopember, Surabaya.

PENGALAMAN KERJA

1. Kerja Praktek TTS (Teknik Tadakara Sumber Karya Surabaya) (Juli-Agustus 2017)
2. UPT. Balai Yasa Surabaya Gubeng (Juli – Agustus 2018)
3. PT. Kereta Api Indonesia (September 2018- Januari 2019)

PENGALAMAN ORGANISASI

1. ITS Badminton Comunity 2017-2018
2. Karang Taruna Blok U (2013-2016)
3. ISGA-SGS (2014-2016)

-----Halaman ini sengaja dikosongkan-----

