

HALAMAN JUDUL

**MEDIA PEMBELAJARAN PENGONTROLAN BERBASIS IoT
(*INTERNET Of THINGS*)**

PROYEK AKHIR

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



Disusun oleh:

Dimas Pradipta NIRM : 0031636

Muhammad Ega Wahyudi NIRM : 0031646

**POLITEKNIK MANUFAKTUR NEGERI
BANGKA BELITUNG
TAHUN 2019**

LEMBAR PENGESAHAN

MEDIA PEMBELAJARAN PENGONTROLAN BERBASIS IoT (INTERNET Of THINGS)

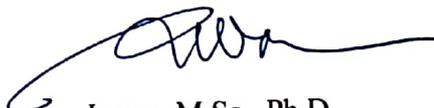
Oleh:

Dimas Pradipta NIRM : 0031636
Muhammad Ega Wahyudi NIRM : 0031646

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

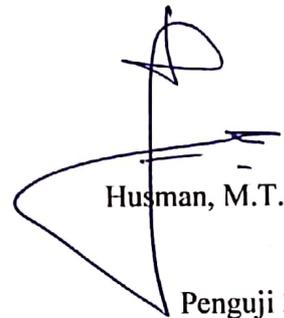
Menyetujui,

Pembimbing 1



Irwan, M.Sc., Ph.D.

Pembimbing 2



Husman, M.T.

Penguji 1



Indra Dwisaputra, M.T.

Penguji 2



Ocsirendi, M.T.

Penguji 3



Ahmat Josi, M.Kom.

PERNYATAAN BUKAN PLAGIAT

Yang bertanda tangan dibawah ini:

Nama Mahasiswa 1 : Dimas Pradipta NIRM : 0031636
Nama Mahasiswa 2 : Muhammad Ega Wahyudi NIRM : 0031646

Dengan Judul : Media Pembelajaran Pengontrolan Berbasis IoT (*Internet Of Things*).

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

Sungailiat, 9 September 2019

Nama mahasiswa

Tanda Tangan

1. Dimas Pradipta



2. Muhammad Ega Wahyudi



ABSTRAK

Di era globalisasi saat ini, perkembangan ilmu pengetahuan dan teknologi sangat pesat, khususnya yaitu perkembangan internet. Oleh karena itu dunia pendidikan pun tidak lepas dari perkembangan internet. Untuk itu dalam pembelajaran perlu adanya media belajar yaitu internet. Dengan berkembangnya Internet of Things (IoT), maka internet pun bisa dimanfaatkan untuk keperluan lain yang mendukung pembelajaran, diantaranya yaitu dengan memanfaatkan internet tersebut untuk kegiatan pembelajaran teori maupun praktikum. Salah satu contohnya yaitu pemanfaatan internet untuk kegiatan pembelajaran studi elektronika. Internet tersebut digunakan sebagai sarana untuk sistem kontrol otomatis dengan jarak jauh menggunakan arduino dan mikrokontroler. Penerapan dari internet of things (IoT) di POLMAN BABEL adalah berupa pengendalian perangkat elektronik berupa lampu LED menggunakan mikrokontroler arduino uno dengan memanfaatkan internet. Penerapan berupa tugas sistem kontrol lampu yang dirancang menggunakan perangkat keras (lampu led dan pin arduino), serta menggunakan perangkat lunak berupa bahasa pemrograman AVR dan pemrograman Arduino. Hasil dari perancangan adalah sebuah trainer kit dan modul praktikum untuk materi piranti elektronika, elektronika digital, mikrokontroler serta IoT dengan sistem kontrol lampu LED yang dapat dikendalikan melalui internet dan menampilkan data sensor pada aplikasi Blynk.

Kata kunci: *Internet of Things, Mikrokontroler, Sistem Kontrol.*

ABSTRACT

ABSTRACT In the era of globalization, The internet technology is developed very rapidly over the world. Consequently the IT could be applied to support learning, including by making use of the internet for theoretical and practical learning activities. One example is the use of the internet for learning activities. Mikrokontroller applied subjects in the program study electrical engineering. The internet is used as a means for automated control system by remotely using a mikrokontroller. Implementation of the internet of things (IOT) in POLMAN BABEL is a form of control of electronic devices such as LED lights using mikrokontroller arduino uno by utilizing the internet. Implementation of such a task light control system that is designed to use the hardware (the LED lights and pin arduino), as well as using software such as AVR programming language and Arduino programming. The results of the design are the manufacture of trainer kits and practicum modules for electronic devices, digital electronics, microcontrollers and IoT materials with LED lighting control systems that can be controlled via the internet and display sensor data in the Blynk application.

Keywords: *Internet of Things, Mikrokontroller, Control System.*

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Allah SWT. karena berkat Rahmat dan Hidayah-Nya, sehingga dapat menyelesaikan proyek akhir ini dengan baik.

Proyek akhir ini dibuat guna sebagai persyaratan dan kewajiban mahasiswa untuk menyelesaikan kurikulum program pendidikan Diploma III di Politeknik Manufaktur Negeri Bangka Belitung.

Selama proses pelaksanaan proyek akhir, pengambilan data dan penulisan laporan ini kami telah banyak mendapatkan bantuan dari berbagai pihak maka dari itu pada kesempatan ini penulis mengucapkan banyak terima kasih kepada :

1. ALLAH SWT atas segala berkat dan rahmat-Nya hingga proyek akhir ini selesai.
2. Keluarga tercinta, yang selalu sabar membimbing, mendoakan, dan memberikan motivasi dalam menyelesaikan proyek akhir ini. Terima kasih untuk setiap peluh keringat serta kasih sayang yang tidak ternilai.
3. Bapak Sugeng Haryono M.eng, Ph D. selaku Direktur Politeknik Manufaktur Negeri Bangka Belitung.
4. Bapak Irwan, Ph. D. Selaku pembimbing I dan Bapak Husman, M.T selaku pembimbing II yang telah meluangkan banyak waktu, tenaga, pikiran di dalam memberikan pengarahan serta bimbingan dalam pembuatan alat dan penulisan laporan proyek akhir ini.
5. Seluruh staf pengajar di Politeknik Manufaktur Negeri Bangka Belitung.
6. Rekan-rekan mahasiswa Politeknik Manufaktur Negeri Bangka Belitung yang telah banyak membantu selama menyelesaikan proyek akhir.
7. Rekan-rekan sepermainan di tempat saya tinggal, yang selalu memberikan semangat mental dan dukungannya.
8. Pihak-pihak lain yang telah memberikan bantuan secara langsung maupun tidak langsung dalam pembuatan proyek akhir ini yang tidak dapat disebutkan satu per satu.

Proyek akhir ini diharapkan dapat bermanfaat bagi penulis khususnya dan bagi pembaca pada umumnya. Penulis menyadari dalam penyusunan proyek akhir ini masih banyak terdapat kekurangan, baik itu dari segi materi, maupun dalam penyampaian materi. Oleh sebab itu, penulis mengharapkan kritik dan saran yang membangun dari semua pihak guna memperbaiki kedepannya.

Akhirnya, semoga laporan ini bisa bermanfaat dan berguna sebagaimana yang diharapkan dan semoga Allah S.W.T. meridhoi-Nya. Aamiin.

Sungailiat, 9 September 2019

Penulis

DAFTAR ISI

HALAMAN JUDUL.....	i
LEMBAR PENGESAHAN	ii
PERNYATAAN BUKAN PLAGIAT	iii
<i>ABSTRAK</i>	iv
KATA PENGANTAR	vi
DAFTAR ISI.....	viii
DAFTAR GAMBAR	x
DAFTAR TABEL.....	xi
DAFTAR LAMPIRAN.....	xii
1.1 Latar Belakang Masalah	1
1.2 Rumusan dan Batasan Masalah	2
1.2.1 Rumusan Masalah	2
1.2.2 Batasan Masalah.....	2
1.3 Tujuan Proyek Akhir.....	3
BAB II DASAR TEORI	4
2.1 Piranti Elektronika	4
2.2 Elektronika Digital.....	5
2.3 Mikrokontroler.....	6
2.4 IoT (<i>Internet Of Things</i>).....	7
BAB III METODE PELAKSANAAN	9
3.1 Studi Pustaka.....	10
3.2 Pengujian Komponen.....	10
3.3 Desain Modul Praktikum	10
3.4 Pembuatan Modul Praktikum	10
3.5 Uji Coba Alat	11
BAB IV PEMBAHASAN.....	12
4.1 Pengujian komponen.....	12
4.1.1 Komponen Piranti Elektronika.....	12

4.1.2	Komponen Elektronika Digital	16
4.2	Mendesain <i>layout</i> PCB	17
4.3	Modul praktikum	22
4.4	<i>Trainer Kit</i>	25
4.4.1.	Piranti Elektronika	25
4.4.2.	Elektronika Digital	27
4.4.3.	Mikrokontroler	29
4.4.4.	IoT (<i>Internet OfThings</i>).....	30
4.5	<i>Box Trainer Kit</i>	30
4.6	Pengujian <i>Hardware</i>	30
4.6.1	Pengujian Rangkaian Piranti Elektronika	31
4.6.2	Pengujian Rangkaian Elektronika Digital	32
4.6.3	Pengujian Output Mikrokontroler	33
4.6.4	Pengujian IoT	34
BAB V	KESIMPULAN DAN SARAN	46
5.1	Kesimpulan	46
5.2	Saran	46
DAFTAR PUSTAKA	47

DAFTAR GAMBAR

Gambar 2.1 Contoh komponen elektronika. [4].....	5
Gambar 3.1 Flow chart pengerjaan proyek akhir.....	9
Gambar 4.1 Cara mengukur resistor. [11].....	13
Gambar 4.2 Cara mengukur dioda. [14].....	14
Gambar 4.3 Layout rangkaian piranti elektronika untuk bagian bawah	18
Gambar 4.4 Layout rangkaian piranti elektronika untuk bagian atas	18
Gambar 4.5 Layout rangkaian elektronika digital untuk bagian bawah	19
Gambar 4.6 Layout rangkaian elektronika digital untuk bagian atas.....	19
Gambar 4.7 Konstruksi seven-segment. [20].....	20
Gambar 4.8 Layout rangkaian output Mikrokontroler tampak bawah.....	21
Gambar 4.9 Layout rangkaian output Mikrokontroler tampak atas.....	21
Gambar 4.10 Blok Digram Konfigurasi Alat.....	25
Gambar 4.11 Modul piranti elektronika.....	26
Gambar 4.12 Modul Elektronika Digital	28
Gambar 4.13 Trainer Kit mikrokontroler.....	29
Gambar 4.14 Desain Box modul IoT	30
Gambar 4.15 Rangkaian dioda Forward Bias	31
Gambar 4.16 Rangkaian gerbang Logika AND.....	32
Gambar 4.17 Output Mikrokontroler LCD 2 x 16	34
Gambar 4.18 Cara menampilkan serial monitor	36
Gambar 4.19 Blok diagram IoT	37
Gambar 4.20 Blok diagram hubungan Arduino-NodeMCU-Android.	37
Gambar 4.21 Pengujian sensor di LCD.....	39
Gambar 4.22 Tampilan grafik sensor Ultrasonik.....	39
Gambar 4.23 Pengujian sensor suhu	41
Gambar 4.24 Tampilan hasil pengujian sensor suhu di Blynk.....	42
Gambar 4.25 LED dalam kondisi off.....	44
Gambar 4.26 LED dalam kondisi on	45

DAFTAR TABEL

Tabel 4.1 Hasil pengukuran praktikum	27
Tabel 4.2 Tabel Kebenaran	29
Tabel 4.3 Tabel hasil pengukuran	31
Tabel 4.4 Tabel kebenaran gerbang logika AND.....	32
Tabel 4.5 Tabel pengujian sensor ultrasonik	40
Tabel 4.6 Tabel pengujian sensor suhu	42
Tabel 4.7 Tabel pengujian On/Off LED	45

DAFTAR LAMPIRAN

Lampiran 1 : Daftar Riwayat Hidup

Lampiran 2 : Modul Piranti Elektronika

Lampiran 3 : Modul Elektronika Digital

Lampiran 4 : Modul Mikrokontroler

Lampiran 5 : Tutorial aplikasi Blynk & Modul IoT

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Saat ini perkembangan teknologi sudah semakin berkembang dengan membuat segala sesuatunya dapat terhubung melalui jaringan internet atau yang biasanya disebut IoT (*Internet of Things*). Perangkat IoT yang murah meriah dan canggih sudah banyak dan akan terus bermunculan. Berbagai layanan platform IoT pun semakin berkembang dan saling berkompetisi. Ekosistem yang semakin meluas ini berdampak pada membludaknya jumlah penggiat IoT untuk membuat berbagai kreasi. *Vision Mobile* dalam salah satu report-nya menyebutkan bahwa pada hingga penghujung tahun 2015 ada sekitar 4,5 juta individu *developer* aktif di seluruh dunia yang mengembangkan perangkat IoT. [1]

Teknologi IoT memungkinkan konektivitas antara sejumlah benda atau objek seperti tag, sensor, aktuator, perangkat *mobile* yang mampu bekerja sama untuk mencapai tujuan bersama. *Internet of Things* membuat sejumlah benda atau objek untuk saling berinteraksi melalui koneksi kabel atau nirkabel, sehingga dapat menciptakan suatu aplikasi atau layanan yang sangat bermanfaat. *Smart classroom* adalah konsep kelas cerdas yang mengintegrasikan teknologi informasi dengan metode pengajaran, pembelajaran yang baru. Konsep *smart classroom* merupakan *evolusi* pembelajaran berbasis internet seperti *e-learning*, *m-learning*, *u-learning* didukung dengan berbagai perangkat yang berinteraksi melalui internet. Menurut ITU-*International Telecommunication Union*, IoT memungkinkan konektivitas untuk siapa saja dan apapun dari manapun kapan saja. IoT dapat mendukung konsep *smart classroom* agar lebih menarik dan mudah untuk di implementasikan. [2]

IoT merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus. Aplikasi penggunaan IoT banyak sekali digunakan di dunia industri dan juga di dunia

pendidikan, misalnya pada proses pengontrolan lampu, pengontrolan suhu dan pengontrolan jarak. Pada dasarnya untuk memahami dengan baik *Internet Of Things* sangat dibutuhkan pemahaman yang jelas tentang karakteristik dari IoT itu sendiri. Penggunaan IoT bertujuan untuk meningkatkan mutu pendidikan, dan efisiensi misalnya dalam hal *E-Learning*, peminjaman buku, absensi siswa dan lain-lain. Pembelajaran tentang gabungan teknik elektronik dan teknologi informasi khususnya di prodi Teknik Rekayasa Perangkat Lunak Politeknik Manufaktur Negeri Bangka Belitung sangat diperlukan mengingat prodi ini mengharapkan lulusannya memiliki kompetensi gabungan antara perangkat lunak, elektronika, dan sistem pengontrolan. Untuk mencapai kompetensi tersebut diperlukanlah media pembelajaran yang sesuai dengan kebutuhan yang diinginkan.

1.2 Rumusan dan Batasan Masalah

1.2.1 Rumusan Masalah

Rumusan masalah yang terdapat pada proyek akhir ini yakni:

1. Bagaimana mendesain dan membuat modul pembelajaran elektronika yang dapat menunjang pembelajaran IoT?
2. Bagaimana mendesain dan membuat *trainer kit* pembelajaran IoT?

1.2.2 Batasan Masalah

Proyek Akhir ini dibatasi oleh beberapa ruang lingkup sebagai berikut:

1. Batas nilai maksimum tegangan *input* setiap praktikum 12Volt
2. Modul praktikum piranti elektronika mencakup:
 - a. Dioda *Forward Bias*
 - b. Dioda *Reverse Bias*
 - c. Hukum Ohm
 - d. Resistor Seri dan Paralel
 - e. Karakteristik Transistor NPN
 - f. Karakteristik Transistor PNP
3. Modul praktikum elektronika digital mencakup:
 - a. Gerbang logika AND

- b. Gerbang logika OR
 - c. Gerbang logika NOT
 - d. Gerbang logika NAND
 - e. Gerbang logika NOR
 - f. Gerbang logika X-OR
4. Modul praktikum mikrokontroler mencakup:
- a. Aplikasi *Port* sebagai *Output* menggunakan LED
 - b. *Input Switch* dan *Output* menggunakan LED
 - c. Aplikasi *Port* sebagai *Output* menggunakan *7-Segment*
 - d. *Interfacing* dan Pemrograman LCD 2 x 16
5. Modul praktikum IoT mencakup:
- a. *On / Off* dari android dengan *Output* LED
 - b. *On / Off* dari android dengan *Output* lampu AC
 - c. Pembacaan nilai sensor jarak
 - d. Pembacaan nilai suhu

1.3 Tujuan Proyek Akhir

Adapun tujuan yang ingin dicapai pembuatan Proyek Akhir ini adalah

1. Membuat modul dan *trainer kit* yang berbasis IoT yang merupakan media/alat bantu ajar untuk pengajar dan mahasiswa.
2. Membuat modul praktikum Piranti Elektronika sehingga hasil pengujian antara perhitungan dengan pengukurannya sama.
3. Membuat modul Elektronika Digital sehingga hasil pengujiannya sesuai dengan tabel kebenaran.
4. Membuat modul mikrokontroler.

BAB II

DASAR TEORI

Untuk memahami prinsip kerja dari IoT maka perlu memahami beberapa tahapan pembelajaran yaitu piranti elektronika, elektronika digital, mikrokontroler dan IoT. Pada bab ini akan menjelaskan tentang teori-teori dasar yang akan digunakan dalam proses pembuatan proyek akhir, yaitu:

2.1 Piranti Elektronika

Didalam perangkat arduino terdapat komponen-komponen piranti elektronika, jadi setiap pemula yang akan mempelajari teknologi berbasis arduino harus terlebih dahulu mengenal komponen piranti elektronika karena ini merupakan satu hal terpenting bagi para peminat baru di bidang elektronika. Bidang elektronika memang sangat dibutuhkan di dalam kehidupan manusia sehari – hari karena sebagian besar kebutuhan hidup mereka dipenuhi oleh benda – benda elektronik. Sayangnya, saat ini banyak sekali orang yang belajar ilmu elektronika dengan mengacuhkan dasar – dasar ilmu elektronika karena dianggap buang – buang waktu dan mereka langsung mempelajari ilmu praktis dari elektronika itu sendiri.

Mengetahui dasar–dasar elektronika adalah suatu kewajiban untuk memberikan mental yang benar untuk para penggiat elektronika dalam pekerjaan mereka. Banyak para penggiat “instan” elektronika yang mengacuhkan peraturan – peraturan dasar elektronika dan hanya fokus kepada aspek keuntungan yang mereka dapatkan saja sehingga hasil kerja mereka kebanyakan asal – asalan dan merugikan banyak orang. Untuk dapat lebih mudah memahami apa itu *Internet Of Things* dengan benar, kita harus mengetahui tahapan proses belajar seperti definisi dari elektronika.

Elektronika adalah ilmu yang mempelajari alat listrik arus lemah yang dioperasikan dengan cara mengontrol, aliran elektron atau partikel bermuatan listrik dalam suatu peralatan elektronika. Ilmu yang mempelajari alat-alat seperti ini merupakan cabang dari ilmu fisika. Sementara bentuk desain dan pembuatan sirkuit

elektroniknya adalah bagian dari teknik elektro, teknik komputer, dan ilmu/teknik elektronika dan instrumentasi. [3]

Ilmu elektronika selalu dan pasti berhubungan dengan komponen – komponen elektronika yang akan sering kita temui dalam proses pembelajarannya. Komponen – komponen tersebut dibagi menjadi dua kelompok yaitu komponen aktif dan pasif. Contoh dari komponen yang termasuk dalam kelompok aktif adalah dioda, transistor, SCR (*Silicon Controlled Rectifier*), FET (*Field Effect Transistor*), dan IC (*Integrated Circuit*). Sedangkan komponen dari kelompok pasif adalah resistor atau tahanan, kapasitor atau kondensator, induktor atau kumparan, dan transformator, berikut gambar 2.1 contoh komponen elektronika.



Gambar 2. 1 Contoh komponen elektronika. [4]

Selanjutnya saat mempelajari elektronika, pembelajar harus mengetahui apa saja konsep elektronika yang akan kita hadapi pada proses pembelajaran berikutnya. Ada dua konsep elektronika yang harus kita kuasai sebelum melangkah ke jenjang pembelajaran praktis elektronika.

2.2 Elektronika Digital

Sebelum ke tahap yang lebih tinggi yaitu IoT maka tahap selanjutnya akan mempelajari Elektronika Digital. Elektronika digital adalah sistem elektronik yang menggunakan *signal* digital. *Signal* digital didasarkan pada signal yang bersifat terputus-putus. Biasanya dilambangkan dengan notasi aljabar 1 dan 0. Notasi 1 melambangkan terjadinya hubungan dan notasi 0 melambangkan tidak terjadinya

hubungan. Contoh yang paling gampang untuk memahami pengertian ini adalah saklar lampu. Ketika kalian tekan *ON* berarti terjadi hubungan sehingga dinotasikan 1. Ketika kalian tekan *OFF* maka akan berlaku sebaliknya sehingga dinotasikan 0. Elektronik digital merupakan aplikasi dari aljabar *boolean* dan digunakan pada berbagai bidang seperti komputer, telpon selular dan berbagai perangkat lain. Hal ini karena elektronik digital mempunyai beberapa keuntungan, antara lain: sistem digital mempunyai antar muka yang mudah dikendalikan dengan komputer dan perangkat lunak, penyimpanan informasi jauh lebih mudah dilakukan dalam sistem digital dibandingkan dengan analog. Namun sistem digital juga memiliki beberapa kelemahan, yaitu: pada beberapa kasus sistem digital membutuhkan lebih banyak energi, lebih mahal dan rapuh. [5]

Pembelajaran Elektronika Digital ini bertujuan memberikan bekal pengetahuan dan keterampilan yang mendasar dan cukup agar mahasiswa dapat merancang dan mengimplementasikan rangkaian digital yang tingkatannya relatif masih sederhana maupun tingkatan rumit seperti IoT.

2.3 Mikrokontroler

Tahapan selanjutnya untuk memahami IoT yaitu mempelajari tentang Mikrokontroler. Menurut Malik (2009:2), “mikrokontroler merupakan alat elektronika digital yang mempunyai masukan dan keluaran serta kendali dengan program yang bisa ditulis dan dihapus dengan cara khusus, cara kerja mikrokontroler sebenarnya membaca dan menulis data” [6]. Mikrokontroler dapat dianalogikan sebagai sebuah sistem komputer yang dikemas dalam sebuah chip, artinya di dalam sebuah IC mikrokontroler sebetulnya sudah terdapat kebutuhan minimal agar mikroprosesor dapat bekerja, yaitu meliputi mikroprosesor, ROM (*Read Only Memory*), RAM (*Random access memory*), I/O dan *clock* seperti halnya yang dimiliki oleh sebuah PC (*Personal Computer*). Mengingat kemasannya yang berupa sebuah chip dengan ukuran yang relatif lebih kecil, tentu saja spesifikasi dan kemampuan yang dimiliki oleh mikrokontroler akan menjadi lebih rendah bila dibandingkan dengan sistem komputer seperti PC baik dilihat dari segi kecepatannya. Mikrokontroler hanya bisa digunakan untuk satu aplikasi tertentu

saja. Meskipun dari sebuah kemampuan lebih rendah tetapi mikrokontroller memiliki kelebihan yang tidak bisa diperoleh pada sistem komputer yaitu, dengan kemasannya yang kecil dan kompak membuat mikrokontroller menjadi lebih *fleksibel* dan praktis digunakan dalam aplikasi pemrograman pengontrolan peralatan *input* atau *output*.

2.4 IoT (*Internet Of Things*)

Perkembangan penggunaan internet pada era modern ini berlangsung cepat dan menyentuh berbagai aspek dalam kehidupan masyarakat. Salah satu konsep penggunaan internet yang perkembangan penggunaan internet pada era modern ini berlangsung cepat dan menyentuh berbagai aspek dalam kehidupan masyarakat. Salah satu konsep penggunaan internet yang membantu orang-orang dalam melakukan suatu tugas atau pekerjaan. Dalam implementasinya, IoT membutuhkan suatu protokol dalam proses peredaran datanya. Pada penelitian “Sistem *Monitoring* Suhu Jarak Jauh Berbasis *Internet of Things* menggunakan Protokol MQTT” yang dilakukan oleh Budioko (2016), konsep *Internet of Things* diterapkan pada suatu sistem *monitoring* dengan sensor suhu menggunakan MQTT sebagai protokol. Sistem *monitoring* adalah suatu sistem yang melakukan proses pemantauan secara terus menerus (Mudjahidin et al., 2010). Sistem *monitoring* dibutuhkan dalam proses pemantauan keadaan suatu objek yang diamati guna mendapatkan informasi yang tepat waktu. Berbagai contoh penerapan sistem *monitoring* yaitu Sistem *Monitoring* dan Pengendalian Suhu dan Kelembaban Ruang pada Rumah Walet Berbasis Android, Web dan SMA oleh Atmoko (2013), Implementasi *Wireless Monitoring* Energi Listrik Berbasis Web *Database* oleh Dinata (2015) dan Sistem Pendeteksi Banjir Berbasis sensor ultrasonik dan Mikrokomputer dengan Media Komunikasi SMS *Gateway* oleh Sulistyowati et al. (2015). [7]

Perkembangan dalam bidang teknologi telah mengubah peradaban dunia. Teknologi memberikan banyak kemudahan dalam kehidupan sehari-hari melalui berbagai peralatan elektronik sampai alat komunikasi. Berbagai macam perangkat (*devices*) elektronik yang sebelumnya berukuran besar, boros *energy*, dan

mahal saat sudah digantikan dengan perangkat-perangkat yang berukuran jauh lebih kecil, lebih canggih, hemat energi (*low-power*), hemat *bandwidth*, dan jauh lebih murah. IoT menjadi topik hangat untuk dibicarakan akhir-akhir ini. IoT tidak hanya menjadi suatu konsep yang mempengaruhi hidup manusia tetapi bagaimana IoT bisa membantu memudahkan kehidupan manusia. [8]

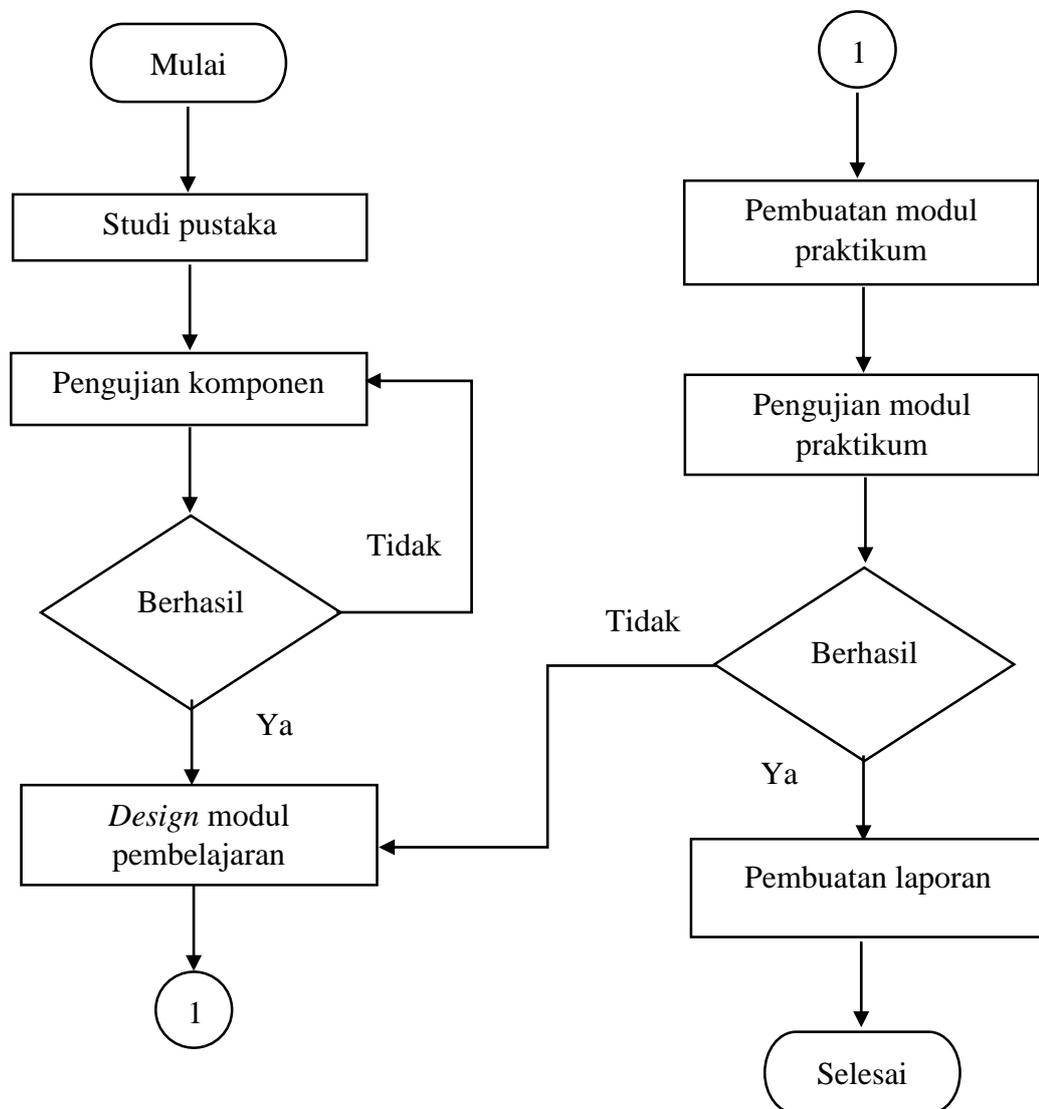
Internet of Things merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus. Adapun kemampuan seperti berbagi data, *remote control*, dan sebagainya, termasuk juga pada benda di dunia nyata. Contohnya bahan pangan, elektronik, koleksi, peralatan apa saja, termasuk benda hidup yang semuanya tersambung ke jaringan lokal dan global melalui sensor yang tertanam dan selalu aktif. Pada dasarnya, *Internet of Things* mengacu pada benda yang dapat diidentifikasi secara unik sebagai representasi *virtual* dalam struktur berbasis Internet. Istilah *Internet of Things* awalnya disarankan oleh Kevin Ashton pada tahun 1999 dan mulai terkenal melalui *Auto-ID Center* di MIT. [9]

Seluruh aktivitas dalam IoT dapat dikendalikan oleh suatu perangkat elektronik yang merupakan pengertian sederhana bagaimana IoT bekerja. Pada dasarnya, jika kita memiliki peralatan yang mempunyai saklar *on* dan *off*, peralatan tersebut mempunyai kesempatan yang besar untuk digunakan dalam ranah IoT. Jadi IoT adalah sebuah konsep dimana suatu objek yang memiliki kemampuan untuk mentransfer data melalui jaringan tanpa memerlukan interaksi manusia ke manusia atau manusia ke komputer. [8]

BAB III

METODE PELAKSANAAN

Pada gambar 3.1 adalah *flow chart* proses pembuatan proyek akhir. Beberapa tahapan yang dilakukan dalam pengerjaan proyek akhir ini, antara lain:



Gambar 3. 1 Flow chart pengerjaan proyek akhir.

3.1 Studi Pustaka

Studi pustaka pada proyek akhir ini dilakukan dengan menelusuri referensi-referensi baik yang tersedia secara *online* maupun *offline*. Penelusuran diawali dengan studi *literatur* tentang IoT itu sendiri, dilanjutkan dengan komponen-komponen atau perangkat pendukung IoT.

Setelah perangkat pendukung IoT dipahami, dilanjutkan dengan penentuan materi-materi yang akan dimasukkan ke dalam materi-materi yang akan dipelajari melalui *trainer kit* ini.

3.2 Pengujian Komponen

Tahapan pengujian komponen pada proyek akhir ini merupakan suatu tahapan untuk menguji komponen-komponen yang digunakan pada tiap-tiap modul pembelajaran. Misalkan pengujian terhadap komponen dioda untuk pembelajaran materi piranti elektronika. Pada tahapan ini dioda diuji untuk memastikan bahwa dioda bekerja sebagaimana mestinya. Demikian juga dengan komponen-komponen pendukung lainnya.

3.3 Desain Modul Praktikum

Desain modul praktikum merupakan tahapan untuk mendesain tata letak dari komponen yang akan dipasang pada papan PCB pada tiap-tiap modul praktikum. *Software* yang digunakan yaitu *software* PCB Wizard dan *software* Circuit Wizard.

3.4 Pembuatan Modul Praktikum

Pembuatan modul praktikum dilakukan setelah desain pada papan PCB selesai dilakukan. Langkah selanjutnya mencetak desain menggunakan kertas foto pada printer *Laser Jet*. Ini bertujuan karena printer tersebut hasil cetakannya sangat bagus sehingga pada saat pencetakan di papan PCB hasilnya bagus pula. Modul praktikum dibuat berdasarkan bagian-bagian dari proses pembelajaran, seperti piranti elektronika, elektronika digital, mikrokontroler, sensor dan IoT.

3.5 Uji Coba Alat

Sebelum melakukan uji coba *trainer kit*, alangkah lebih baiknya jika mempersiapkan peralatan dengan sebaik mungkin supaya *trainer kit* bisa bekerja dengan semaksimal mungkin. Uji coba media pembelajaran ini meliputi beberapa tahapan pengujian seperti uji coba terhadap rangkaian *trainer kit* Piranti Elektronika yang meliputi pengujian terhadap rangkaian dioda *Forward Bias*, dioda *Reverse Bias*, karakteristik transistor NPN dan transistor PNP, hukum Ohm, serta resistor seri dan paralel. Untuk pengujian rangkaian tersebut dilakukan dengan melakukan pengambilan data berupa nilai tegangan dan arus pada masing-masing percobaan.

Uji coba *trainer kit* Piranti Elektronika dianggap berhasil apabila karakteristik komponen sesuai dengan penjelasan teori mengenai karakteristik komponen serta nilai tegangan dan arus sama dengan nilai analisa perhitungan.

Sedangkan untuk uji coba *trainer kit* Elektronika Digital meliputi uji coba rangkaian Gerbang logika AND, Gerbang logika OR, Gerbang logika NOT, Gerbang logika NAND, Gerbang logika NOR, Gerbang logika EX OR, Gerbang logika EX NOR dilakukan dengan melakukan pengambilan data berupa data biner untuk mengisi tabel kebenaran pada setiap percobaan. Jika hasil pengujian sesuai dengan analisa tabel kebenaran gerbang logika maka uji coba terhadap *trainer kit* Elektronika Digital dianggap berhasil.

Pengujian mikrokontroler berhasil jika *output* yang digunakan berfungsi sebagai mana mestinya. Sedangkan untuk pengujian IoT dilihat dari segi pengiriman data antara android dengan arduino apabila berfungsi perintah yang diberikan dari android maka itu dianggap berhasil.

BAB IV

PEMBAHASAN

Pembuatan proyek akhir ini menggunakan beberapa objek yang terdiri dari komponen elektronika, *software* pemrograman, mikrokontroler ATmega8535, Arduino Uno, serta komponen *input* dan *output*.

Pada proyek akhir ini tujuan utamanya yaitu dapat mengontrol peralatan elektronika menggunakan android sehingga dapat dikontrol dari jarak jauh menggunakan software Arduino Uno dan aplikasi Blynk, tetapi sebelum melangkah ke pengontrolan menggunakan android ada beberapa tahapan-tahapan yang harus dipelajari yaitu memahami piranti elektronika, elektronika digital dan mikrokontroler.

Proses pembuatan modul media pembelajaran pengontrolan berbasis IoT akan di jelaskan di bagian bab ini sehingga modul yang dibuat sesuai apa yang diinginkan dan siap untuk digunakan. Dalam pembuatan modul pembelajaran ini ada beberapa tahapan proses:

1. Pengujian komponen
2. Mendesain *layout* modul praktikum
3. Pembuatan modul praktikum
4. Pengujian modul praktikum

4.1 Pengujian komponen

Setiap komponen harus melalui tahap uji coba untuk memastikan apakah komponen tersebut masih berfungsi atau tidak, berikut ini cara pengujiannya:

4.1.1 Komponen Piranti Elektronika

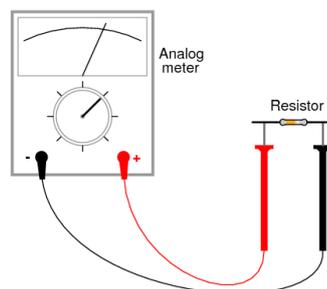
Komponen piranti elektronika sebenarnya banyak jenis tetapi dalam pembuatan modul ini ada beberapa komponen yang dipakai dalam pembuatan modul piranti elektronika yaitu:

a. Resistor

Resistor merupakan salah satu komponen pasif yang paling sering ditemukan dan digunakan dalam Rangkaian Elektronika. Hampir setiap peralatan Elektronika menggunakannya. Pada dasarnya Resistor adalah komponen Elektronika pasif yang memiliki nilai resistansi atau hambatan tertentu yang berfungsi untuk membatasi dan mengatur arus listrik dalam suatu rangkaian Elektronika. Resistor atau dalam bahasa Indonesia sering disebut dengan Hambatan atau Tahanan dan biasanya disingkat dengan Huruf “R”. Satuan Hambatan atau Resistansi Resistor adalah OHM (Ω). Sebutan “OHM” ini diambil dari nama penemunya yaitu Georg Simon Ohm yang juga merupakan seorang Fisikawan Jerman. [10]

Untuk menguji apakah resistor masih bagus atau tidak dan mengetahui berapa nilai pada suatu resistor tersebut yakni:

1. Pastikan Multimeter sudah terkalibrasi, untuk multimeter digital tidak perlu kalibrasi.
2. Siapkan resistor yang akan diukur misalkan 470 Ohm
3. Putarlah selektor multimeter pada posisi $\times 1 \text{ K}\Omega$ atau sesuai keperluan dan tidak boleh selektor berada dibawah nilai resistor yang akan diukur.
4. Hubungkan *probe* merah (+) dan *probe* hitam (-) pada masing-masing ujung resistor yang akan diukur. Berikut pada gambar 4.1 merupakan cara pengukuran resistor.



Gambar 4. 1 Cara megukur resistor. [11]

5. Setelah kedua *probe* dihubungkan, maka seketika jarum pada multimeter akan bergerak pada nilai skala tertentu. Kemudian lihat dan amati nilai

yang ditunjukkan pada jarum multimeter tersebut. Untuk multimeter digital nilainya secara otomatis akan muncul dilayar *display*.

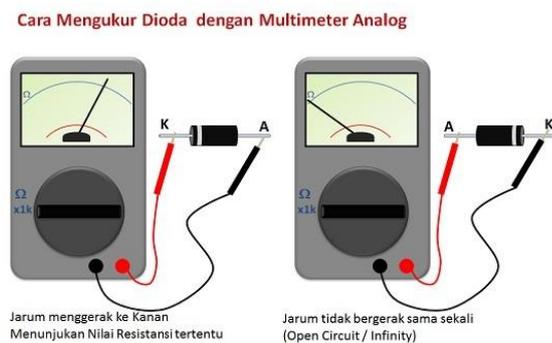
6. Jika jarum tidak bergerak atau nilai tidak muncul maka dipastikan resistor tersebut rusak. [12]

b. Dioda

Dioda adalah komponen elektronika yang terdiri dari dua kutub dan berfungsi menyearahkan arus. Komponen ini terdiri dari penggabungan dua semikonduktor yang masing-masing diberi *doping* (penambahan material) yang berbeda, dan tambahan material konduktor untuk mengalirkan listrik. [13]

ada beberapa langkah dalam mengukur dioda sebagai berikut:

1. Aturkan Posisi Saklar pada Posisi OHM (Ω) x1k atau x100
2. Hubungkan *Probe* Merah pada Terminal *Katoda* (tanda gelang)
3. Hubungkan *Probe* Hitam pada Terminal *Anoda*.
4. Baca hasil Pengukuran di *Display* Multimeter
5. Jarum pada *Display* Multimeter harus bergerak ke kanan
6. Balikan *Probe* Merah ke Terminal *Anoda* dan *Probe* Hitam pada Terminal *Katoda* (tanda gelang).
7. Baca hasil Pengukuran di *Display* Multimeter
8. Jarum harus tidak bergerak. Jika Jarum bergerak, maka Dioda tersebut berkemungkinan sudah rusak [14]. Cara pengukuran dioda bisa dilihat pada gambar 4.2.



Gambar 4. 2 Cara mengukur dioda. [14]

Catatan: untuk mengetahui pin *anoda* dan *katoda* caranya dengan melihat cincin atau gelang berwarna putih, itu menandakan bahwa pin *katoda*.

c. LED

LED (*Light Emitting Diode*) adalah sejenis dioda, yang dapat memancarkan cahaya apabila mendapat arus maju sekitar 5-30 mA. pada umumnya LED terbuat dari *gallium pospat* dan *arsenit pospat*. didalam aplikasinya, LED sering digunakan sebagai alat indikasi status/kondisi tertentu, misalnya sebagai tampilan “*Seven-segment*”. [15]

d. Transistor

Transistor adalah komponen semikonduktor yang dipakai sebagai penguat, pemotong (*switching*), stabilisasi tegangan, modulasi sinyal atau fungsi lainnya. Transistor dapat berfungsi semacam kran listrik, dimana berdasarkan arus *input*-nya BJT (*Bipolar Junction Transistor*) atau tegangan inputnya FET (*Field Effect Transistor*), memungkinkan pengaliran listrik yang sangat akurat dari sirkuit sumber listriknya. [16]

e. Kapasitor

Kapasitor adalah komponen listrik umumnya yang secara fisis terdiri dari dua konduktor yang dipisahkan bahan isolator atau *dielektrikum*. Kapasitansi ialah ukuran kemampuan kapasitor menyimpan energi dalam medan listrik. Kapasitansi dinyatakan dalam bentuk *farad*. 1 *farad* adalah kemampuan kapasitor untuk menyimpan muatan listrik 1 *coulomb* apabila diberi tegangan 1 volt. [17]

f. Saklar *Toggle*

Saklar *Toggle* ini menghubungkan atau memutuskan arus dengan cara menggerakkan *toggle*/tuas yang ada secara mekanis. Cara kerja saklar *toggle* adalah pada saat saklar ditekan maka saklar tersebut akan mengalirkan arus listrik dan pada saat dilepas maka saklar tersebut masih dapat mengalirkan arus listrik dan baru saat saklar ditekan kembali maka saklar akan memutuskan arus listrik, intinya jika ingin melakukan perubahan kondisi *on/off* pada saklar maka harus dengan melakukan penekanan pada saklar. [18]

4.1.2 Komponen Elektronika Digital

Gerbang Logika (*Logic Gate*) sebuah pemrosesan dasar yang berguna dalam memproses *input-input* berupa bilangan biner. Dengan kata lain gerbang logika beroperasi atau bekerja berdasarkan sistem bilangan biner, secara singkat sistem bilangan biner dapat diartikan sebagai jenis bilangan yang terdiri dari 2 kode angka yaitu "0" dan "1".

Didalam pembuatan modul praktikum elektronika digital ada beberapa komponen yang dipakai dalam pembuatan modul ini yaitu:

a. IC 7408

Gerbang AND menggunakan IC TTL tipe SN 7408. Pada gerbang AND apabila kedua *input* berada pada logika 1 maka *output* akan berada pada logika 1, namun apabila semua atau salah satu *input*-nya berada pada logika 0 maka *output*-nya juga berada pada logika 0. [19]

b. IC 7432

Gerbang OR menggunakan IC TTL tipe SN 7432. Pada gerbang OR apabila salah satu atau kedua *input* berada pada logika 1 maka *output* akan berada pada logika 1. Namun jika kedua *input* berlogika 0 maka *output* akan berlogika 0. [19]

c. IC 7404

Gerbang NOT menggunakan IC TTL tipe SN 7404. Gerbang NOT hanya mempunyai satu sinyal *input* dan satu sinyal *output*. Dalam gerbang NOT, untuk menghasilkan *output* berlogika 1 maka *input* harus berada pada logika 0. Sebaliknya untuk menghasilkan *output* logika 0 maka *input* harus berlogika 1. [19]

d. IC 7400

Gerbang NAND menggunakan IC TTL tipe SN 7400. Gerbang NAND merupakan kebalikan dari gerbang AND. Apabila kedua *input* berada pada logika 1 maka *output* akan berlogika 0. Namun apabila semua *input* atau salah satu *input* berada pada logika 0 maka *output* akan berlogika 1. [19]

e. IC 7402

Gerbang NOR menggunakan IC TTL tipe SN 7402. Gerbang NOR merupakan kebalikan dari gerbang logika OR. Apabila kedua *input* berada

pada logika 0 maka *output* berlogika 1. Sebaliknya jika salah satu/kedua *input* berlogika 1 maka *output* akan berlogika 0. [19]

Gerbang logika ini merupakan kombinasi antara gerbang logika OR dan gerbang logika NOT, jadi hasil keluaran gerbang logika OR itu di NOT kan.

f. IC 7486

Gerbang EX-OR menggunakan IC TTL tipe SN 7486. Gerbang EX-OR berfungsi untuk mendeteksi keadaan-keadaan logika yang berbeda diantara kedua *input* nya. Jika kedua *input* mempunyai logika yang berbeda maka *output* berlogika 1. Tetapi jika kedua *input* mempunyai logika yang sama maka *output* berlogika 0. [19]

4.2 Mendesain *layout* PCB

Untuk mendesain rangkaian *layout*, menggunakan software PCB Wizard dan Circuit Wizard karena *software* ini penggunaannya terbilang mudah dan komponennya pun cukup lengkap. Pembuatan desain *layout* PCB ada beberapa macam:

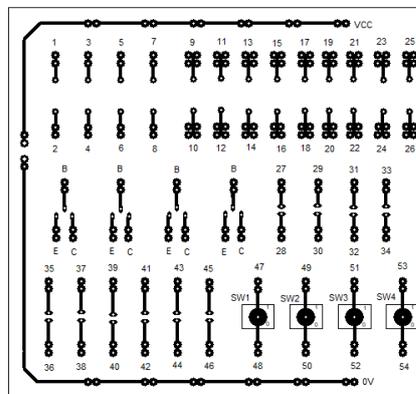
a. Desain *layout* PCB untuk rangkaian piranti elektronika

Elektronika adalah ilmu yang mempelajari tentang komponen-komponen listrik yang biasanya digunakan untuk mengontrol aliran atau partikel bermuatan listrik di dalam suatu rangkaian atau dalam suatu alat. Ilmu yang mempelajari alat-alat seperti ini merupakan cabang dari ilmu fisika, sementara bentuk desain dan pembuatan sirkuit elektroniknya adalah bagian dari Teknik Elektro, Teknik Komputer dan Instrumentasi. Peralatan yang menggunakan dasar kerja elektronika ini biasanya disebut sebagai peralatan elektronik. Didalam rangkaian piranti elektronika penulis menggunakan beberapa komponen elektronika seperti: resistor, dioda, LED, transistor, kapasitor dan saklar togel.

Dalam hal mendesain tata letak komponen perlu memperhitungkan beberapa hal seperti, ukuran PCB, jumlah komponen yang digunakan dan kerapian tata letak komponen. karena tujuan dari pembuatan PCB adalah memperoleh papan (board) untuk merakit komponen-komponen yang dibutuhkan dalam membuat suatu sistem elektronika. Hal yang terpenting disini adalah PCB yang kita buat adalah PCB yang

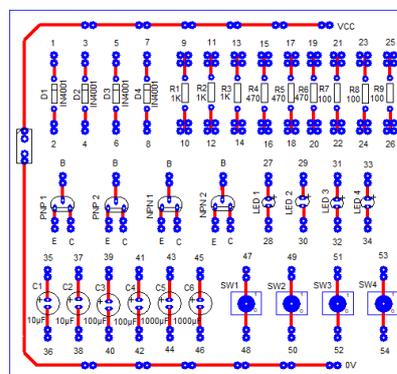
sesuai dengan harapan. Artinya ketika komponen telah dipasang pada PCB, maka rangkaian elektronika yang kita rancang dapat bekerja sesuai dengan harapan kita. Kedua, dengan cara yang sederhana ini kita tidak mengeluarkan banyak biaya untuk membuat PCB.

Berikut ini gambar *layout* rangkaian praktikum piranti elektronika yang akan digunakan sebagai tampilan bagian bawah papan PCB. Berikut ini adalah gambar *layout* rangkaian piranti elektronika yang di tampilkan pada gambar 4.3.



Gambar 4. 3 Layout rangkaian piranti elektronika untuk bagian bawah

Sedangkan pada gambar 4.4 adalah *layout* rangkaian praktikum piranti elektronika yang akan digunakan sebagai tampilan di papan PCB untuk tampak atas. Berikut ini adalah gambar *layout* rangkaian piranti elektronika.

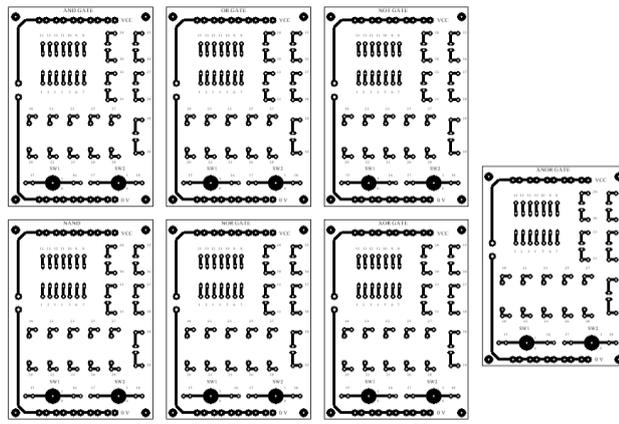


Gambar 4. 4 *Layout* rangkaian piranti elektronika untuk bagian atas

b. Desain *layout* PCB untuk rangkaian elektronika digital

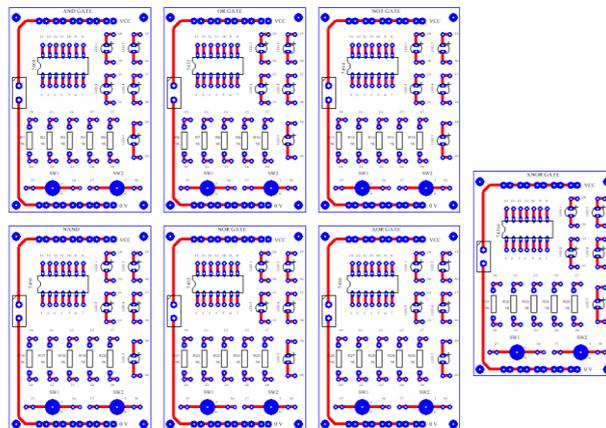
Didalam rangkaian Elektronika Digital penulis menggunakan beberapa IC (*Integrated Circuit*) Gerbang Logika seperti: IC 7408, IC 7432, IC 7404, IC 7400, IC 7402, IC 7486.

Berikut ini gambar 4.5 *layout* rangkaian praktikum elektronika digital untuk bagian bawah pada papan PCB.



Gambar 4. 5 *Layout* rangkaian elektronika digital untuk bagian bawah

Sedangkan untuk gambar 4.6 merupakan gambar *layout* rangkaian praktikum elektronika digital untuk bagian atas menggunakan papan PCB.



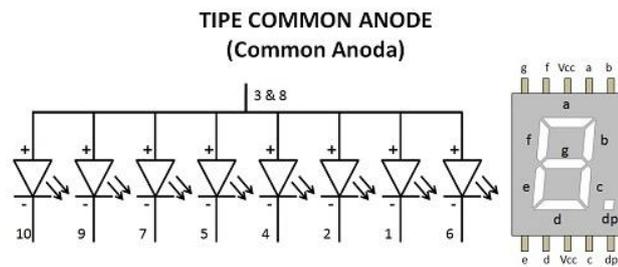
Gambar 4. 6 *Layout* rangkaian elektronika digital untuk bagian atas.

c. Desain *layout* PCB untuk *output* mikrokontroler

Ada beberapa komponen yang digunakan untuk *output* mikrokontroler seperti: LED, 7-*segment*, dan LCD.

a. *Seven-Segment*

Pada LED 7 Segmen jenis *Common Anode (Anoda)*, Kaki *Anoda* pada semua *segment* LED adalah terhubung menjadi 1 Pin, sedangkan kaki *Katoda* akan menjadi *input* untuk masing-masing *Segment* LED. Kaki *Anoda* yang terhubung menjadi 1 Pin ini akan diberikan tegangan Positif (+) dan *Signal* Kendali (*control signal*) akan diberikan kepada masing-masing Kaki *Katoda Segment* LED [20]. Berikut bentuk konstruksi *seven-segment* pada gambar 4.7



Gambar 4. 7 Konstruksi *seven-segment*. [20]

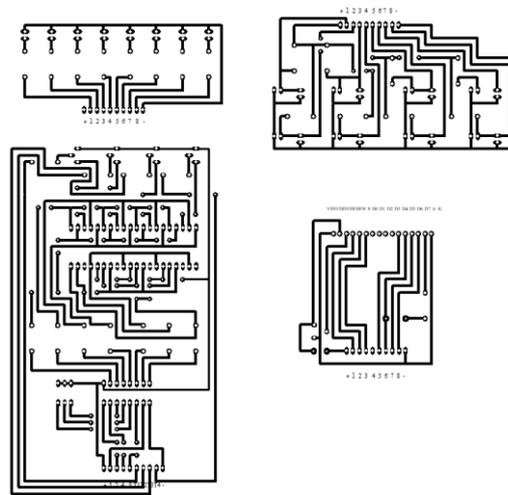
b. *Liquid Crystal Display (LCD)*

LCD adalah suatu jenis media *display* (tampilan) yang menggunakan kristal cair (*liquid crystal*) untuk menghasilkan gambar yang terlihat. Teknologi LCD atau Penampil Kristal Cair sudah banyak digunakan pada produk-produk seperti layar Laptop, layar Ponsel, layar Kalkulator, layar Jam Digital, layar Multimeter, Monitor Komputer, Televisi, layar *Game portabel*, layar *Thermometer* Digital dan produk-produk elektronik lainnya. [21]

Dari beberapa komponen diatas merupakan komponen yang akan digunakan untuk bahan mendesain *layout output* mikrokontroller. Didalam

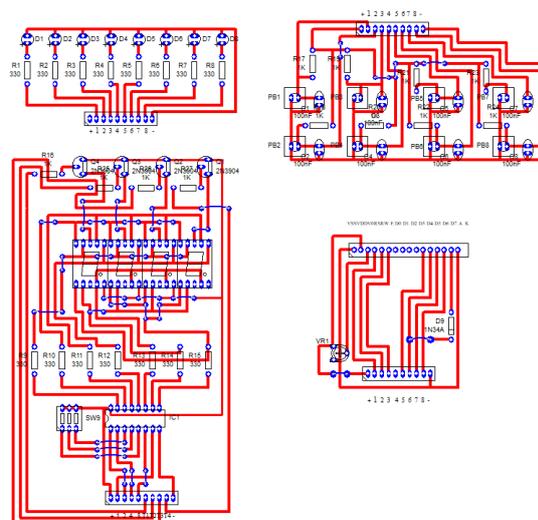
pembuatan proyek akhir ini tidak menggunakan *output* lain seperti *Dot matrix* dan *key pad*.

Berikut ini gambar 4.8 *layout* rangkaian *output* mikrokontroler yang digunakan untuk rangkaian bagian bawah papan PCB.



Gambar 4. 8 *Layout* rangkaian *output* Mikrokontroler tampak bawah

Sedangkan untuk tampilan atas PCB modul mikrokontroler dapat dilihat pada gambar 4.9 berikut ini:



Gambar 4. 9 *Layout* rangkaian *output* Mikrokontroler tampak atas

Gambar 4.9 tersebut digunakan untuk tampilan pada bagian atas sehingga dapat mengetahui rangkaian pada output mikrokontroler

4.3 Modul praktikum

Pembuatan modul praktikum pada proyek akhir ini meliputi pembuatan modul praktikum itu sendiri yang berisi dengan langkah-langkah dan petunjuk-petunjuk praktikum, serta *trainer kit* pembelajaran.

Materi pembelajaran piranti elektronika mencakup materi tentang:

a. Dioda *Forward Bias*

Materi ini bertujuan agar mahasiswa dapat memahami karakteristik dioda *forward bias* berdasarkan hasil pengukuran arus dan tegangan pada dioda *forward bias*, mengetahui simbol dioda serta mahasiswa dapat membuat aplikasi penerapan dioda.

b. Dioda *Reverse Bias*

Tujuan mempelajari materi ini yaitu mahasiswa dapat memahami karakteristik dioda *reverse bias* berdasarkan hasil pengukuran arus dan tegangan pada dioda *reverse bias*, dapat mengetahui simbol dioda serta mahasiswa dapat membuat aplikasi penerapan dioda.

c. Hukum Ohm

Materi ini bertujuan agar mahasiswa dapat mempelajari hubungan arus listrik, tegangan listrik, dan hambatan listrik yang lebih dikenal dengan hukum ohm.

d. Resistor Seri dan Paralel

Materi rangkaian seri paralel bertujuan mahasiswa dapat memahami dan mempelajari rangkaian seri dan paralel pada resistor.

e. Karakteristik Transistor NPN

Mahasiswa dapat memahami karakteristik Transistor NPN berdasarkan percobaan yang sudah dilakukan, mengetahui simbol Transistor NPN, dan dapat membuat aplikasi penerapan Transistor NPN. Contoh aplikasi penerapan transistor NPN seperti transistor sebagai saklar, transistor sebagai penguat dan lain-lain.

f. Karakteristik Transistor PNP

Mahasiswa dapat memahami karakteristik Transistor PNP berdasarkan percobaan yang sudah dilakukan, mengetahui simbol Transistor PNP, dan dapat membuat aplikasi penerapan Transistor PNP.

Materi pembelajaran elektronika digital terdiri dari:

a. Gerbang logika AND

Tujuan materi ini adalah mahasiswa dapat memahami prinsip dasar gerbang logika AND serta dapat menganalisa tabel kebenaran gerbang logika AND.

b. Gerbang logika OR

Tujuan materi ini adalah Mahasiswa dapat memahami prinsip dasar gerbang logika OR, dapat menganalisa tabel kebenaran gerbang logika OR serta dapat mengetahui simbol dari gerbang logika OR.

c. Gerbang logika NOT

Tujuan materi ini adalah mahasiswa dapat memahami prinsip dasar gerbang logika NOT, dapat menganalisa tabel kebenaran gerbang logika NOT, dapat mengetahui simbol dari komponen gerbang logika NOT.

d. Gerbang logika NAND

Tujuan materi ini adalah mahasiswa dapat memahami prinsip dasar gerbang logika NAND, dapat menganalisa tabel kebenaran gerbang logika NAND.

e. Gerbang logika NOR

Tujuan materi ini adalah mahasiswa dapat memahami prinsip dasar gerbang logika NOR dan dapat menganalisa tabel kebenaran gerbang logika NOR.

f. Gerbang logika X-OR

Tujuan materi ini adalah mahasiswa dapat memahami prinsip dasar gerbang logika XOR, dapat menganalisa tabel kebenaran gerbang logika XOR.

Sedangkan materi mikrokontroler mencakup:

a. Aplikasi *Port* sebagai Output menggunakan LED

Tujuan materi ini adalah mahasiswa bisa menggunakan *software* CodeVisionAVR berbasis bahasa pemrograman C dengan prosedur yang benar dan mahasiswa bisa mengendalikan nyala LED sebagai *port output* mikrokontroler dengan teknik dan kombinasi yang diinginkan secara baik dan benar.

b. *Input Switch* dan *Output* menggunakan LED

Tujuan materi ini adalah mahasiswa bisa menggunakan *software* CodeVisionAVR berbasis bahasa pemrograman C dengan prosedur yang benar, mahasiswa mengerti teknik pembacaan *input* pada pin mikrokontroller menggunakan *switch* baik secara *hardware* maupun pemrogramannya.

c. Aplikasi *Port* sebagai *Output* menggunakan *7-Segment*

Tujuan materi ini adalah mahasiswa bisa menggunakan *software* CodeVisionAVR berbasis bahasa pemrograman C dengan prosedur yang benar, mahasiswa bisa mengendalikan nyala *7-segment* sebagai *port* keluaran mikrokontroler sesuai dengan karakter *interfacing*.

d. *Interfacing* dan Pemrograman LCD 2 x 16

Tujuan materi ini adalah mahasiswa bisa menggunakan *software* CodeVisionAVR berbasis bahasa pemrograman C dengan prosedur yang benar, mahasiswa mengerti teknik *interfacing* LCD 2x16 untuk mode 4 bit dan 8 bit, dan mahasiswa bisa memprogram LCD 2x16 untuk mode 4 bit dan 8 bit dengan baik dan benar.

Untuk IoT, materi pembelajaran terdiri dari:

a. *On/Off* dari android dengan *output* LED

Materi ini bertujuan mahasiswa dapat memahami proses sistem kerja IoT sehingga dapat mengontrol lampu LED dari jarak jauh.

b. *On/Off* dari android dengan *output* lampu AC

Materi ini bertujuan mahasiswa dapat memahami proses sistem kerja IoT sehingga dapat mengontrol lampu AC dari jarak jauh.

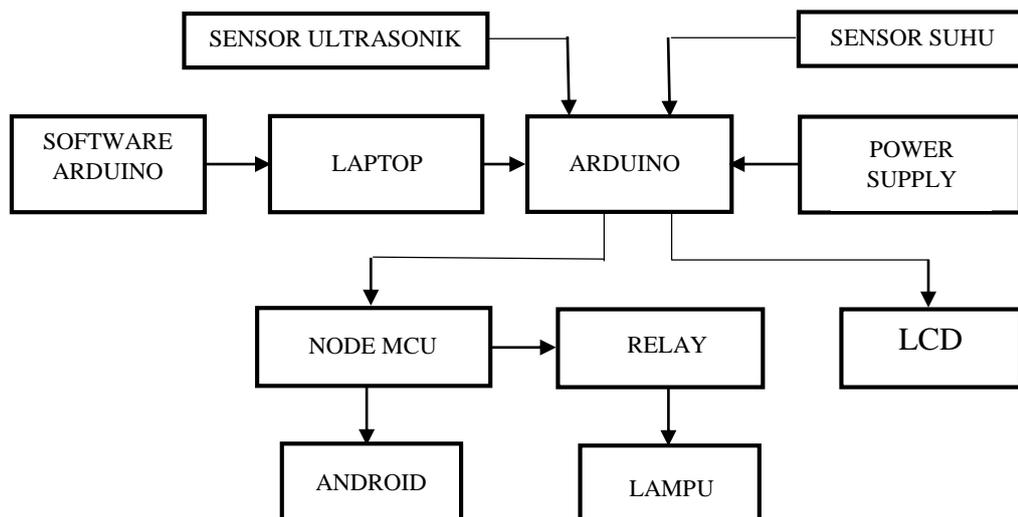
c. Pembacaan nilai sensor jarak

Tujuan materi ini adalah mahasiswa dapat mengontrol jarak dan mengetahui karakteristik sensor ultrasonik. Sensor ultrasonik HC-SR04 adalah sebuah sensor yang berfungsi untuk mengubah besaran fisis (bunyi) menjadi besaran listrik dan sebaliknya. Cara kerja sensor ini didasarkan pada prinsip dari pantulan suatu gelombang suara sehingga dapat dipakai untuk menafsirkan eksistensi (jarak) suatu benda dengan frekuensi tertentu. Disebut sebagai sensor ultrasonik karena sensor ini menggunakan gelombang ultrasonik (bunyi ultrasonik). [22]

d. Pembacaan nilai suhu

Tujuan materi ini adalah mahasiswa dapat mengontrol *temperature* dan mengetahui karakteristik sensor suhu (DHT11). **Sensor DHT11** adalah *module* sensor yang berfungsi untuk *sensing* objek suhu dan kelembaban yang memiliki *output* tegangan analog yang dapat diolah lebih lanjut menggunakan mikrokontroler. *Module* sensor ini tergolong kedalam elemen resistif seperti perangkat pengukur suhu seperti contohnya yaitu NTC. [23]

Secara umum blok diagram Konfigurasi proyek akhir ini terdiri dari laptop, software Arduino, Node MCU Esp 8266, Android, Arduino Uno, relay, *input* dan *output*. Pada gambar 4.10 merupakan blok diagram konfigurasi proyek akhir.



Gambar 4. 10 Blok Digram Konfigurasi Alat

4.4 Trainer Kit

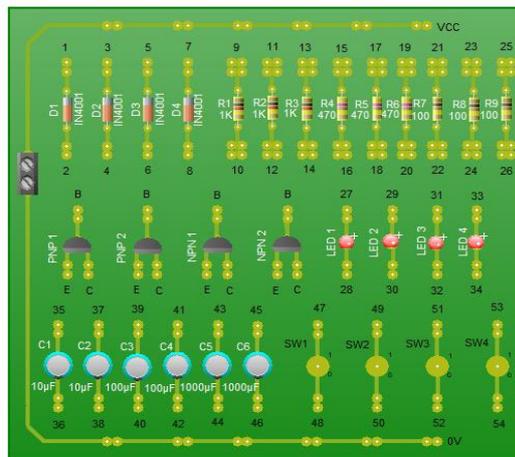
Trainer kit pada proyek akhir ini merupakan media praktikum untuk mengimplementasikan instruksi-instruksi yang diberikan pada modul-modul praktikum yang dibuat. *Trainer kit* ini terdiri dari *trainer kit*:

4.4.1. Piranti Elektronika

Trainer kit piranti elektronika ini ditujukan agar pembelajaran tentang komponen-komponen elektronika dapat dipahami langsung oleh mahasiswa

dengan mempraktikkan langsung petunjuk-petunjuk yang ada di modul praktikum. Praktikum ini merupakan materi dasar yang perlu dipelajari sebelum ke tahapan yang lebih tinggi karena di praktikum inilah mahasiswa mengenal berbagai macam rangkaian serta karakteristik dari beberapa komponen piranti elektronika.

Berikut ini gambar 4.11 menunjukkan tampak *trainer kit* piranti elektronika pada proyek akhir ini.



Gambar 4. 11 Modul piranti elektronika

Sebagai contoh, pada praktikum tentang “Dioda Forward Bias”, mahasiswa diinstruksikan untuk mengikuti prosedur-prosedur berikut:

- Matikan terlebih dahulu catu dayanya.
- Buatlah rangkaian seperti gambar percobaan 1. Hubungkan setiap titik dari masing-masing komponen menggunakan kabel jumper.
- Hubungkan *input* saklar di titik 47 dengan *power supply* (vcc).
- Hubungkan titik 48 saklar 1 (SW 1) ke alat ukur ampermeter (+) untuk mengukur arus pada dioda (I_d). Kemudian ampermeter (-) itu juga dihubungkan dengan titik 1 pada dioda.
- Hubungkan titik 2 dioda dengan resistor di titik 9 sebagai beban, karena menggunakan resistor $1K \Omega$.
- Hubungkan titik 10 resistor dengan *power supply* sebagai *ground*.

- Atur *power supply* dari tegangan 0V sampai 12V. Untuk mengukur tegangan dioda (V_d) hubungkan alat ukur voltmeter di titik 1 dan 2. Sedangkan untuk mengukur tegangan keluaran (V_{out}) hubungkan voltmeter di titik 9 dan 10. Kemudian catat hasil pengukuran tegangan dioda (V_d), arus dioda (I_d) dan tegangan keluaran (V_{out}) pada data hasil percobaan.

Kemudian mahasiswa diminta untuk mencatat hasil pengukuran pada tabel 4.1 berikut:

Tabel 4. 1 Hasil pengukuran praktikum

No	Vcc (Volt)	I_d (mA)	V_d (Volt)	V_{out} (Volt)
1.	0V			
2.	1V			
.
13.	12 V			

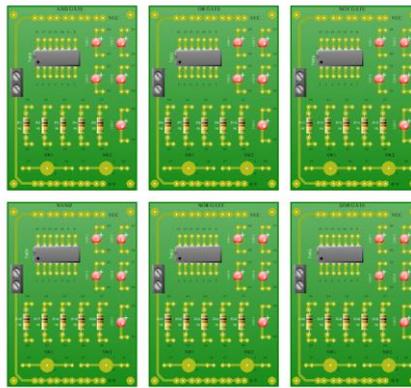
4.4.2. Elektronika Digital

Materi ini membahas dasar rangkaian elektronika digital. Topik-topik yang dibahas meliputi cara kerja dan sifat dari gerbang-gerbang kombinasional dasar (AND, OR, NOT, NAND, NOR, EXOR). *Trainer kit* elektronika digital ini ditujukan agar pembelajaran tentang gerbang logika dasar yang dapat dipahami langsung oleh mahasiswa dengan mempraktikkan langsung petunjuk-petunjuk yang ada di modul praktikum. Mata kuliah ini merupakan lanjutan dari mata kuliah Elektronika Digital 1, dimana rangkaian digital dikontrol berdasarkan waktu menjadi rangkaian *sekuensial*. Dengan mengetahui prinsip dari gerbang logika maka akan mempermudah dalam memahami prinsip kerja dari sebuah rangkaian yang mengandung unsur logika.

Gerbang logika adalah suatu entitas dalam elektronika dan matematika *boolean* yang mengubah satu atau beberapa masukan logik menjadi sebuah sinyal keluaran logik.

Gerbang logika terutama diimplementasikan secara elektronis menggunakan diode atau transistor, akan tetapi dapat pula dibangun menggunakan susunan komponen-komponen yang memanfaatkan sifat-sifat elektromagnetik (*relay*), cairan, optik dan bahkan mekanik." [15]

Gambar 4.12 berikut menunjukkan gambar *trainer kit* elektronika digital pada proyek akhir ini.



Gambar 4. 12 Modul Elektronika Digital

Sebagai contoh, pada praktikum tentang “Gerbang Logika AND”, mahasiswa diinstruksikan untuk mengikuti prosedur-prosedur berikut:

- a. Matikan terlebih dahulu catu dayanya.
- b. Rangkailah sesuai dengan prosedur-prosedur yang ada di praktikum
- c. Ujilah rangkaian yang sesuai dengan petunjuk tabel yang ada pada gerbang AND dan buatlah tabel kebenaran dari tegangan masukan terhadap tegangan keluarannya.
- d. kemudian amati hasil dari percobaan satu persatu.
- e. Analisa rangkaian gerbang logika AND.
- f. Buatlah kesimpulan berdasarkan hasil pengujian rangkaian gerbang logika AND. Catatan: untuk praktikum diperlukan ketelitian dalam melakukan praktikum agar tidak salah pada saat pengambilan data.

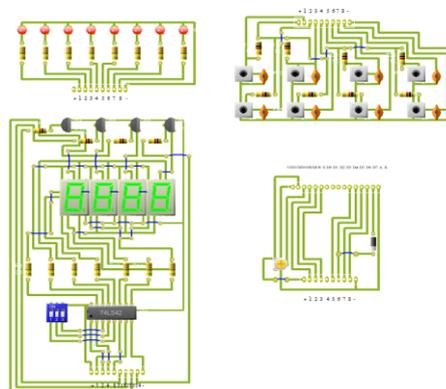
Kemudian setelah melakukan praktikum mahasiswa diminta untuk menganalisa dari praktikum tersebut dan mencatat hasil praktikum pada tabel 4.2 sebagai berikut:

Tabel 4. 2 Tabel Kebenaran

V INPUT		Output (V)
SW1	SW2	
0V	0V	
0V	5V	
5V	0V	
5V	5V	

4.4.3. Mikrokontroler

Modul praktikum mikrokontroler ini ditujukan agar pembelajaran tentang mikrokontroler dasar dapat dipahami langsung oleh mahasiswa dengan mempraktikkan langsung petunjuk-petunjuk yang ada di modul praktikum. Tujuan mempelajari materi mikrokontroler ini adalah untuk membangun kompetensi mahasiswa sehingga penerapan mikrokontroler dapat berguna untuk kebutuhan sehari-hari seperti kontrol motor kontinyu, peralatan kendali *display* (LCD, 7-Segment, LED). Pada gambar 4.13 berikut menunjukkan gambar *trainer kit* mikrokontroler pada proyek akhir ini.



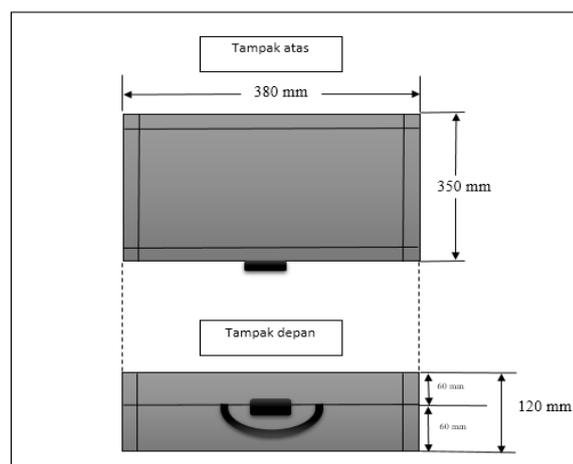
Gambar 4. 13 *Trainer Kit* mikrokontroler

4.4.4. IoT (*Internet Of Things*)

Modul praktikum IoT ini ditujukan agar pembelajaran tentang IoT dasar dapat dipahami langsung oleh mahasiswa dengan mempraktikkan langsung petunjuk-petunjuk yang ada di modul praktikum. Tujuan mempelajari materi IoT ini adalah untuk menambah wawasan tentang pemanfaatan teknologi jaringan internet sehingga mempermudah pengendalian, pengawasan dan pengontrolan terhadap objek elektronika.

4.5 *Box Trainer Kit*

Tempat untuk meletakkan rangkaian modul praktikum yaitu berbentuk koper dengan menggunakan triplek sebagai bahan pembuatan koper yang di lapiasi kain berbahan dasar kulit. Dimensi dari koper yang digunakan sesuai pada gambar 4.14



Gambar 4. 14 *Desain Box* modul IoT

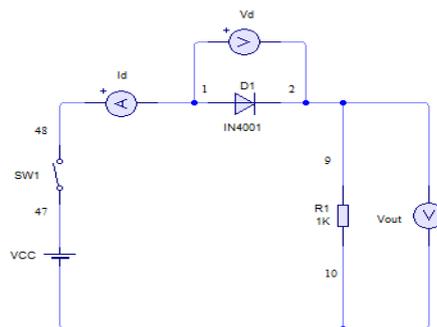
4.6 Pengujian *Hardware*

Hardware yang telah dibuat kemudian dilakukanlah pengujian, setelah itu ada beberapa aspek pengujian yang berguna untuk memastikan apakah *hardware* tersebut berfungsi dengan baik. Hal-hal tersebut yang akan diuji adalah sebagai berikut.

4.6.1 Pengujian Rangkaian Piranti Elektronika

Langkah selanjutnya adalah pengujian modul praktikum untuk materi piranti elektronika ini mengambil salah satu praktikum yang ada pada modul ajar, tetapi untuk percobaan lainnya terdapat di lampiran. Pengujian ini bertujuan untuk melihat apakah modul berfungsi sesuai keinginan atau tidak. Pada gambar 4.15 rangkaian percobaan “DIODA FORWARD BIAS”, rangkaian ini salah satu contoh dari praktikum piranti elektronika.

Berikut ini pengujian salah satu rangkaian percobaan yang terdapat pada modul piranti elektronika, untuk percobaan lainnya terdapat pada lampiran 2. Pada kali ini pengujiannya menggunakan rangkaian percobaan Dioda *Forward Bias* dilihat pada gambar 4.15



Gambar 4. 15 Rangkaian dioda *Forward Bias*

Setelah selesai merangkai maka instruksi selanjutnya lakukan pengukuran dan catat hasil pada tabel 4.3

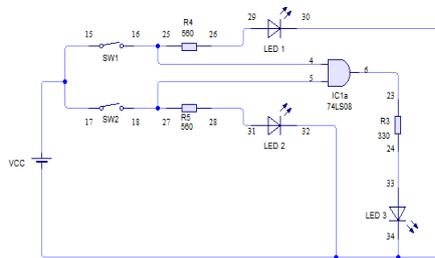
Tabel 4. 3 Tabel hasil pengukuran

No	Vcc (Volt)	Id (mA)	Vd (Volt)	Vout (Volt)
1.	0V	0	0	0
2.	1V	0,3	0,7	0,3
.
13.	12 V	11	0,7	11,3

Dari praktikum yang dilakukan bahwa hasil pengukuran dan hasil perhitungan teradapat selisih yaitu 0,1 – 0,2 mA untuk hasil pada Id, sedangkan pada Vd selisih nya yaitu 0,1 volt dan Vout hasilnya menyesuaikan pada hasil Vd.

4.6.2 Pengujian Rangkaian Elektronika Digital

Langkah selanjutnya adalah pengujian modul praktikum untuk materi elektronika digital ini bertujuan untuk melihat apakah modul berfungsi sesuai keinginan atau tidak. Modul praktikum yang telah selesai maka dilakukan pengujian dengan cara mempraktikkan dan menganalisa prinsip dari rangkaian tersebut. Pada pengujian ini menggunakan salah satu rangkaian dari percobaan yang ada pada modul elektronika digital yaitu menggunakan rangkaian gerbang logika AND dan percobaan lainnya terdapat pada lampiran 2. Berikut gambar 4.16 rangkaian percobaan praktikum elektronika digital.



Gambar 4. 16 Rangkaian gerbang Logika AND

Dari praktikum diatas maka hasil pengujian diisi kedalam tabel kebenaran 4.4 berikut ini:

Tabel 4. 4 Tabel kebenaran gerbang logika AND

V INPUT		Output (V)
SW1	SW2	
0V	0V	0
0V	5V	0
5V	0V	0
5V	5V	5

Dari hasil pengujian yang dilakukan hasil *output* yang dikeluarkan oleh gerbang logika AND sesuai dengan tabel kebenaran.

4.6.3 Pengujian Output Mikrokontroler

Pengujian output mikrokontroler hanya menggunakan salah satu *output* mikrokontroler yaitu LCD. Langkah-langkah memprogram pada mikrokontroler dapat dilihat pada lampiran 4. Berikut contoh pengujian program instruksi pengujian LCD:

```
#include <mega8535.h>
// Alphanumeric LCD functions
#include <alcd.h>
// Declare your global variables here
void main(void)
{
    DDRA =0xFF;
    PORTA=0x00;
    lcd_init(16);
while (1)
    {
        lcd_gotoxy(0,0);
        lcd_putsf("Dimas Pradipta");
        lcd_gotoxy(0,1);
        lcd_putsf("M. Ega Wahyudi");
    }
}
```

Setelah selesai memprogram maka program tersebut di *download* ke IC mikrokontroler yaitu IC ATmega 8535. Pada gambar 4.17 merupakan hasil *output* pengujian mikrokontroler yang merupakan salah satu dari praktikum pada modul mikrokontroler yang berjudul “Menampilkan Karakter pada LCD” untuk percobaannya lainnya bisa dilihat pada lampiran 4.

Pada dasarnya pengujian yang dilakukan ini bertujuan untuk memahami bagaimana cara menampilkan karakter yang diinginkan ke sebuah *display* berupa LCD. Tampilan tersebut dapat berupa tulisan atau angka sesuai keinginan pemrogram, tampilan LCD yang digunakan yaitu 2x16 artinya mempunyai 2 baris dan 16 kolom.

Hasil pengujian yang telah dilakukan dapat dilihat pada gambar 4.17 yang menampilkan karakter pada baris 1 menampilkan “Dimas Pradipta” sedangkan pada baris 2 menampilkan “M. Ega Wahyudi”.



Gambar 4. 17 *Output* Mikrokontroler LCD 2 x 16

Dari hasil pengujian dapat disimpulkan bahwa *trainer kit* yang dibuat telah bekerja sesuai tujuan yang diinginkan.

4.6.4 Pengujian IoT

Arduino dapat dioperasikan dengan cara menginstal terlebih dahulu *software* atau aplikasi pendukung untuk memprogram mikrokontroler arduino berupa program sederhana, dengan *programming environment* turunan dari bahasa pemrograman C yang mudah dimengerti.

Setelah proses penginstalan *software* arduino berhasil, maka untuk dasar tutorial pemrograman arduino dapat dilihat pada lampiran 6 dan berikut ini penjelasan beberapa *function* yang biasa digunakan pada program arduino, sebagai berikut:

Void setup() dimana Semua kode program yang berada dalamnya akan dibaca sekali oleh Arduino. Biasanya isinya berupa kode perintah untuk menentukan fungsi pada sebuah pin.

Void loop() dimana semua kode program yang berada didalamnya akan dibaca setelah *void setup* dan akan dibaca secara berulang terus-menerus oleh arduino, hingga Arduino dimatikan atau di reset. Isinya berupa kode-kode perintah kepada pin *INPUT* dan *OUTPUT* pada Arduino.

pinMode() digunakan untuk melakukan konfigurasi secara spesifik fungsi dari sebuah pin, apakah digunakan sebagai input atau sebagai *output*. Contoh penggunaan *function pinMode()* ialah sebagai berikut.

- ***pinMode(0, INPUT);*** //konfigurasi pin 0 Arduino sebagai pin *input*
- ***pinMode(13, OUTPUT);*** //konfigurasi pin 13 Arduino sebagai pin *output*

digitalRead() digunakan untuk membaca nilai pin digital yang spesifik, apakah bernilai *HIGH* atau *LOW*. Contoh penggunaan *function digitalRead()* seperti berikut ini :

- ***digitalRead(0);*** //membaca nilai digital dari pin 0 Arduino

digitalWrite() digunakan untuk membaca nilai digital, selain itu *function* ini juga untuk menuliskan atau memberikan nilai pada suatu pin digital secara spesifik. *Function digitalWrite()* memberikan nilai pin digital yang spesifik apakah bernilai *HIGH* atau *LOW*, yang dapat dilakukan. Contoh penggunaan *function digitalWrite()* seperti berikut ini:

- ***digitalWrite(13, HIGH);*** //memberikan nilai digital *HIGH* pada pin 13 Arduino
- ***digitalWrite(13, LOW);*** //memberikan nilai digital *LOW* pada pin 13 Arduino

delay() dimana *function delay()* ini digunakan untuk memberikan waktu tundaan (dalam satuan *millisecond*) untuk mengerjakan satu baris program ke baris selanjutnya. Contoh penggunaan *function delay()* seperti berikut ini :

- ***delay(500);*** /*memberikan waktu tundaan 500 *millisecond*, atau setara dengan 0.5 detik sebelum melanjutkan mengerjakan perintah baris program selanjutnya, jika diinginkan waktu tunda 1 detik maka ditulis *delay(1000)* dan seterusnya.

analogRead() digunakan untuk membaca nilai analog. *Function analogRead()*, digunakan untuk membaca nilai analog melalui pin analog. Untuk board Arduino Uno memiliki 6 channel analog, Arduino Mini dan Nano 8 channel, sedangkan Arduino Mega 10 channel, dengan resolusi 10 bit analog to digital *converter*. Dengan resolusi 10 bit memungkinkan pemetaan tegangan antara 0 volt hingga 5 volt dalam nilai integer dari 0 hingga 1023. Sehingga resolusi pembacaan nilai analog ialah 5 volt dibagi 1024 unit, atau sekitar 4,9 mV per unit. Dibutuhkan sekitar 100 *microsecond* untuk membaca suatu *input* analog, dengan kata lain

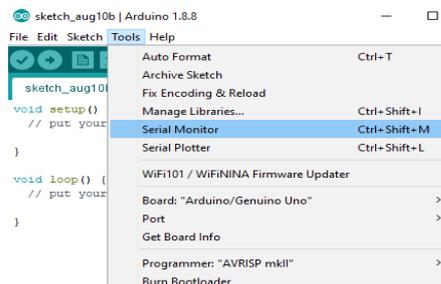
tingkat pembacaan *maximum* nilai analog ialah 10000 kali dalam satu detik. Contoh dari *function analogread()* seperti berikut ini :

- ***analogRead(A0);*** //membaca nilai analog dari pin A0 Arduino

Serial.print() digunakann untuk menampilkan teks dan juga dapat menampilkan nilai pada sebuah sensor di Serial Monitor, contohnya penggunaan *function Serial.print()* seperti berikut ini :

- ***Serial.print ("Hello World");*** //Untuk menampilkan kata *Hello World*
- ***Serial.print (nama sensor);*** //Untuk menampilkan nilai dari sensor yang telah di kenalkan. [24]

Serial Monitor pada Arduino sendiri dapat dibuka dengan dengan memilih menu *Tools* kemudian pilih Serial Monitor. Ataupun dengan menekan kombinasi CTRL+SHIFT+M di keyboard. Atau bisa juga dengan meng-klik ikon Kaca Pembesar di Arduino, seperti ditunjukkan pada gambar 4.18 berikut ini:

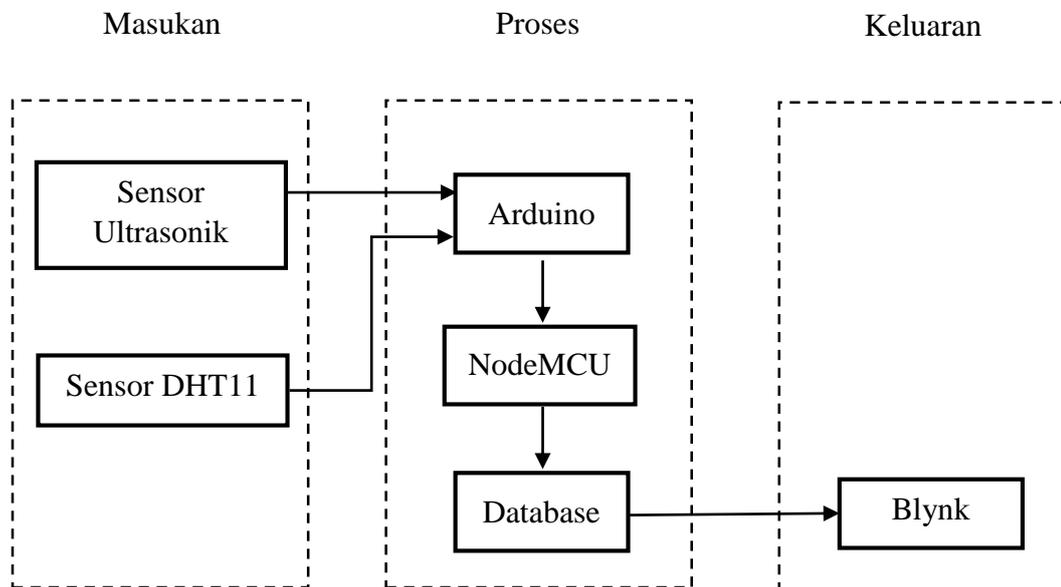


Gambar 4. 18 Cara menampilkan serial *monitor*

Pengiriman, pemrosesan data koneksi internet antara NodeMCU dan *software* Blynk dilakukan dengan mengirimkan data sensor (*input*) sehingga data tersebut diproses oleh arduino kemudian akan dikirim ke NodeMCU lalu dengan bantuan jaringan internet nama NodeMCU akan mengirimkan data tersebut ke android yang ditampilkan melalui aplikasi Bynk.

Pengiriman data oleh arduino ke NodeMCU menggunakan pin D1 dan D2 ini berfungsi sebagai jalur penerima dan pengirim data. Ketika melakukan *uploading* ke NodeMCU maka pin tersebut tidak boleh terhubung ke arduino karena akan menyebabkan *error* saat proses *uploading*.

Proses pemrograman untuk pengujian modul IoT terdapat dua program utama yaitu program arduino dan program NodeMCU. Berikut blok diagram IoT.



Gambar 4. 19 Blok diagram IoT

Sedangkan untuk blok diagram hubungan Arduino - NodeMCU - Android dapat dilihat pada gambar 4.20



Gambar 4. 20 Blok diagram hubungan Arduino-NodeMCU-Android.

Pada NodeMCU menggunakan data serial dari arduino untuk mendapatkan data. Kemudian setelah NodeMCU terkoneksi dengan internet maka NodeMCU mengirimkan data ke android untuk ditampilkan pada *software* yang telah di desain.

Pemrograman utama terdapat pada arduino yaitu menangkap nilai sensor. Pembacaan nilai sensor ini digunakan sebagai input data yang akan dikirim. Selain itu terdapat program fungsi-fungsi lain seperti pengiriman data. Selanjutnya pemrograman untuk mengirim data serial arduino ke NodeMCU, pemrograman ini berfungsi untuk mengirimkan data-data yang akan dibaca oleh NodeMCU untuk

dikirim ke android. Terdapat banyak *coding* didalam program tersebut contoh instruksi `ArduinoSerial.print();` adalah perintah untuk mengirimkan data ke NodeMCU.

Data yang akan dikirim oleh arduino akan diterima oleh NodeMCU selanjutnya akan di proses oleh NodeMCU untuk dikirimkan ke aplikasi Blynk pada android berfungsi untuk melihat data yang tampil sehingga mudah di kontrol oleh user. Data yang diterima akan dikirimkan ke android pengguna. Ketika data diterima oleh NodeMCU maka data itu akan menjadi acuan dalam menentukan data yang akan dikirim ke android.

Pada instruksi `char auth[] = "0f6f5336487045689be94fab9c858ad";` ini merupakan perintah untuk meletakkan token akun Blynk sehingga terhubung pada *project* yang dibuat. Selain token pemrograman dilengkapi ID dan *password* WiFi yang akan digunakan sebagai jaringan internet. [25]

Aplikasi Blynk bisa di-*download* di *play store* pada android kemudian daftarkan menggunakan email atau akun lainnya. Setelah mendaftar tunggu beberapa saat, maka token akan dikirim ke alamat email tersebut. Kemudian pada *setting* komponen *monitor gauge* atau komponen lainnya dapat mengisi identitas *tittle* dan *input*. Pada pengujian modul IoT, disini data yang di *monitor* menggunakan *SuperChart* untuk mengontrol grafik jarak. Proses *setting*-an pada aplikasi Blynk secara jelas dapat dilihat dilampiran 5. Jika telah di *setting* maka selanjutnya pengujian. Ini dilakukan bertujuan untuk melihat apakah modul berfungsi sesuai tujuan. Untuk pengujian modul IoT maka digunakan salah satu yaitu sensor ultrasonik digunakan untuk mengetahui seberapa akurasi jarak yang di ukur manual menggunakan penggaris. Kemudian *example* program untuk sensor jarak yang telah disediakan bisa di-*download* dalam *library* arduino itu sendiri.

Dibawah ini merupakan pengujian modul IoT menggunakan sensor ultrasonik, pemrograman bisa dilihat pada lampiran 6.

Program untuk mengirim nilai sensor dari arduino ke NodeMCU yaitu :

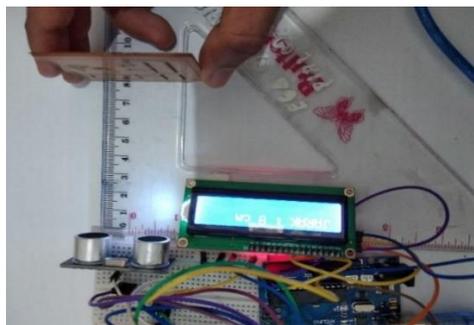
```
void loop()
{
  // dht();
  ultrasonik();
}
```

```

mySerial.print (cm); //menampilkan nilai jarak yg akan dikirim
mySerial.print ('j'); //kode sensor ultrasonik
lcd.setCursor(0,0);
lcd.print("JARAK : ");
lcd.print(cm);
lcd.print(" cm    ");
delay(200);}

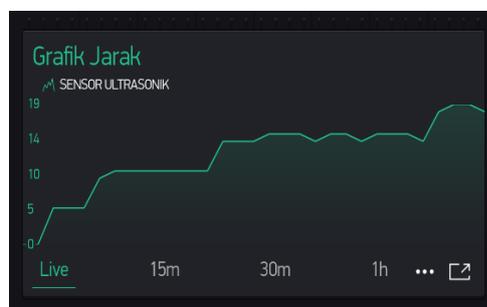
```

Setelah selesai memprogram maka *upload* program ke arduino, kemudian bandingkan dengan alat ukur baik berupa penggaris atau pun alat ukur lainnya. Dibawah ini pengujian menggunakan alat ukur berupa penggaris dapat dilihat pada gambar 4.21



Gambar 4. 21 Pengujian sensor di LCD

Dibawah ini tampilan hasil di *monitor* android dapat dilihat pada gambar 4.22 dibawah ini.



Gambar 4. 22 Tampilan grafik sensor Ultrasonik

Hasil dari pengujian diatas bahwa tampilan yang terdapat di *monitor* android sesuai dengan input. Pada tampilan hasil *monitor* android jika hasilnya belum tampil maka itu disebabkan gangguan sinyal yang terjadi pada saat NodeMCU mengirimkan data ke android, jadi untuk meminimal hal tersebut di anjurkan menggunakan jaringan yang kualitas bagus.

Berikut persentase *error* dari hasil praktikum sensor ultrasonik pada modul *trainer* IoT dapat dilihat pada tabel 4.5

Tabel 4. 5 Tabel pengujian sensor ultrasonik

No	Pengukuran Sensor (Center)	Pengukuran Sensor (Kemiringan)	Pengukuran Penggaris	Persentase Error (center)	Persentase Error (kemiringan)
1	5 cm	6 cm	5 cm	0%	1%
2	10 cm	11 cm	10 cm	0%	1%
3	14 cm	15 cm	14 cm	0%	1%
4	19 cm	20 cm	19 cm	0%	1%

$$\text{Persentase error sensor} = \frac{20-19}{100} \times 100\%$$

$$\text{Persentase error sensor} = 0,01 \times 100\%$$

$$\text{Persentase error sensor} = 1\%$$

Dari hasil pengujian diatas dapat disimpulkan bahwa sensor ultrasonik tidak presisi ketika membaca nilai yang non desimal karena pada saat membaca nilai non desimal hasil pada monitor selalu berubah-ubah sehingga untuk pengambilan data menggunakan nilai desimal. Apabila posisi objek miring maka hasil pengukuran pada sensor ultrasonik + 1 maka persentase *error* yaitu 1%.

Kemudian pengujian kedua pada modul IoT menggunakan sensor suhu, pemrograman detailnya bisa dilihat pada lampiran 6.

Program untuk mengirim nilai sensor dari arduino ke NodeMCU yaitu :

```
void loop()
{
  dhtt();
  mySerial.print (suhu); //menampilkan nilai jarak yg akan dikirim
}
```

```

void dhdt()
{
  // read without samples.
  byte temperature = 0;
  byte humidity = 0;
  float err = SimpleDHTerrSuccess;
  if ((err = dht11.read(&temperature, &humidity, NULL)) !=
SimpleDHTerrSuccess) {
Serial.print("Read DHT11 failed, err=");
Serial.println(err);delay(1000);
  return;
}
Serial.print("Sample OK: ");
Serial.print((float)temperature); Serial.print(" *C, ");
Serial.print((float)humidity); Serial.println(" H");
lcd.setCursor(0,0);
lcd.print("SUHU : ");lcd.print((float)temperature); lcd.print("
drajat, ");
  suhu=(float)temperature;
  // DHT11 sampling rate is 1HZ.
  delay(500);
}

```

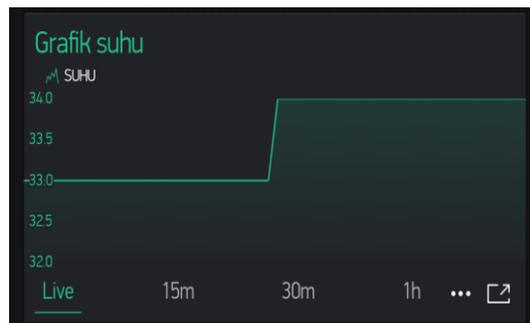
Setelah memprogram maka *upload* program ke arduino lalu bandingkan dengan thermomter. Dibawah ini pengujian menggunakan alat ukur berupa thermometer dapat dilihat pada gambar 4.23



Gambar 4. 23 Pengujian sensor suhu

Sensor suhu yang digunakan ialah sensor DHT11, sumber panas yang digunakan adalah panas dari solder. Dalam pengujian terdapat selisih antara sensor suhu dengan alat ukur dengan selisih 0,4 derajat *celcius*.

Berikut tampilan hasil pengujian sensor suhu pada layar *monitor* android di aplikasi Blynk dapat dilihat pada gambar 4.24



Gambar 4. 24 Tampilan hasil pengujian sensor suhu di Blynk

Sedangkan untuk hasil selisih perbandingan antara sensor suhu dengan thermometer dapat dilihat pada tabel 4.6 yang merupakan hasil pengujian praktikum sensor suhu pada modul IoT.

Tabel 4. 6 Tabel pengujian sensor suhu

No	Pengukuran Sensor	Pengukuran Thermometer	Persentase Error
1	28 C	28,4 C	0,4%
2	29 C	29,4 C	0,4%
3	33 C	33,6 C	0,6%
4	34 C	34,4 C	0,4%

$$\text{Persentase error sensor} = \frac{34,4 - 34}{100} \times 100\%$$

$$\text{Persentase error sensor} = 0,004 \times 100\%$$

$$\text{Persentase error sensor} = 0,4\%$$

Dari hasil pengujian diatas dapat disimpulkan bahwa sensor suhu terdapat perbandingan dengan alat ukur thermometer dan sensor suhu tersebut tidak presisi

ketika membaca nilai yang non desimal karena pada saat membaca nilai non desimal hasil pada monitor selalu berubah-ubah sehingga untuk pengambilan data menggunakan nilai desimal. Dari hasil pengukuran tersebut maka rata-rata *error* nya adalah 0,4%.

Dengan menguji dari tiap-tiap bagian rangkaian untuk mendapatkan hasil apakah alat yang telah dirancang sesuai dengan yang diharapkan. Pengujian ini bertujuan untuk memastikan apakah *trainer kit* berfungsi dengan baik. langkah pertama yaitu meemprogram untuk instruksi *on/off* LED, setelah selesai pembuatan program untuk hardware arduino, cobalah lakukan koneksi antara android dan arduino untuk menghidupkan LED secara *ON/OFF*.

Membuka project aplikasi blynk yang telah dibuat, kemudian coba lakukan dengan menekan tombol “Play” untuk melakukan koneksi antara hp smartphone anda dengan arduino. Tekan tombol *ON* untuk mengaktifkan LED dan tombol *OFF* untuk mematikan LED. Pada tabel 4.6 merupakan pengujian untuk praktikum *On/Off* LED pada modul IoT.

Kemudian pengujian ketiga pada modul IoT *on/off* lampu, pemrograman detailnya bisa dilihat pada lampiran 5.

Program untuk mengirim data dari arduino ke NodeMCU yaitu :

```
const int PIN_2 = 2;
//const int PIN_13 = 11;
void setup() {
  pinMode (2, INPUT);
  pinMode (11, OUTPUT);
}
void loop() {
  // int kondisiPB = digitalRead (PIN_2);
  if (digitalRead (2) == HIGH)
  {
    digitalWrite(11, HIGH);
  }
  if (digitalRead (2) == LOW)
  {
    digitalWrite(11, LOW);
  }
}
```

```
// delay (1000);  
}
```

Setelah selesai pembuatan program untuk hardware arduino, cobalah lakukan koneksi antara android dan arduino untuk menghidupkan LED secara *ON/OFF*. Buka *project* aplikasi Blynk yang telah dibuat, kemudian coba lakukan dengan menekan tombol “Play” untuk melakukan koneksi antara hp *smartphone* anda dengan arduino. Tekan tombol *ON* untuk mengaktifkan LED dan tombol *OFF* untuk mematikan LED

Pada gambar 4.25 adalah push button pada kondisi *off* maka hasilnya lampu LED juga *off*.



Gambar 4. 25 LED dalam kondisi *off*

Pada hasil pengujian diatas hubungan antara *receiver* dan *transmitter* berfungsi sesuai dengan tujuan yang dicapai. Karena instruksi yang diberikan dari android bias diterima dan diolah dengan baik oleh NodeMCU dan arduino uno. Saat android memberikan instruksi *off* maka led tidak menyala atau *off* dan tampilan pada software BLYNK juga *off*. Instruksi ini berfungsi untuk memberikan *notifikasi* bahwa saat *push button* dalam kondisi *off* maka led pada software BLYNK tidak menyala.

Kemudian pada gambar 4.26 LED pada kondisi on maka LED menyala sesuai dengan input pada *monitor* Blynk.



Gambar 4. 26 LED dalam kondisi *on*

Maka kesimpulannya adalah koneksi antara *monitor* dengan arduino berfungsi dengan baik karena LED dapat bekerja sesuai data yang di *input*. Berikut hasil pengujian LED pada tabel 4.7

Tabel 4. 7 Tabel pengujian *On/Off* LED

No	Kondisi	Hasil
1	Off	Off
2	On	On

Dari tutorial ini kita dapat mengetahui bahwa aplikasi yang dibuat diharapkan mempermudah aktifitas pengguna dalam kehidupan sehari-hari, Dimana lebih efisien jika dibandingkan kita mengontrol LED secara manual (*switching*). Kode program yang di *upload* ke Arduino akan dijalankan dengan memeriksa data yang diterima dan membandingkannya. Jika data yang diterima adalah 1 LED akan menyala dan jika data yang diterima adalah 0 maka LED akan padam. [26]

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari hasil pengujian dan analisa yang telah dilakukan terhadap proyek akhir ini yang berjudul “Media Pembelajaran Pengontrolan Berbasis IoT” maka kesimpulan yang dapat ditarik sebagai berikut:

Pada modul praktikum Piranti Elektronika hasil pengujian antara perhitungan dengan pengukuran terdapat selisih hasil pengukuran dan hasil perhitungan terdapat selisih yaitu 0,1 – 0,2 mA untuk hasil pada I_d , sedangkan pada V_d selisih nya yaitu 0,1 volt.

Modul Elektronika Digital hasil pengujiannya sesuai dengan tabel kebenaran. Pada pengujian sensor ultrasonik jarak minimum yang terdeteksi yaitu 3 cm dan maksimal 30cm.

Trainer kit sudah mencakup empat materi dasar yang dibutuhkan yaitu, Piranti Elektronika, Elektronika Digital, Mikrokontroler dan IoT.

5.2 Saran

Dari hasil pengerjaan proyek akhir ini terdapat beberapa saran kedepannya sehingga bisa dikembangkan modul ini sebagai media pembelajaran yang lebih efektif, antara lain :

Untuk pengembangan lebih lanjut modul pembelajaran pengontrolan berbasis IoT ini dapat ditambah menggunakan server.

Modul dan *trainer kit* yang dibuat sebaiknya dilakukan pengujian ke mahasiswa untuk melihat apakah efektif atau tidak.

DAFTAR PUSTAKA

- [1] T. Haryanto, “Belajar IoT - Menyalakan dan Mematikan Lampu Via Internet Menggunakan AgnosThings dan ESP8266,” 4 Februari 2019. [Online]. Available: <https://www.codepolitan.com/belajar-iot-menyalakan-dan-mematikan-lampu-via-internet-menggunakan-agnosthings-dan-wemos-esp8266>. [Diakses 1 September 2019].
- [2] R. Habibi, “PERENCANAAN SISTEM SMART ACADEMIC DENGAN SMART CLASSROOM DAN TEKNOLOGI INTERNET OF THING PADA STMIK BINA PATRIA,” *TRANSFORMASI Jurnal Informasi & Pengembangan Iptek*, vol. 13, no. STMIK BINA PATRIA, pp. 1-2, 2017.
- [3] R. Elektronika, “PENTINGNYA BELAJAR ELEKTRONIKA UNTUK PEMULA,” Rangkaian Elektronika.Org, 18 September 2015. [Online]. Available: <http://www.rangkaianelektronika.org/pentingnya-belajar-elektronika-untuk-pemula.htm>. [Diakses 24 Maret 2019].
- [4] D. Kho, “Jenis-jenis Komponen Elektronika beserta Fungsi dan Simbolnya,” Teknik Elektronika, [Online]. Available: <https://teknikelektronika.com/jenis-jenis-komponen-elektronika-beserta-fungsi-dan-simbolnya/>. [Diakses 3 September 2019].
- [5] B. Herlandi, “ELEKTRONIKA DIGITAL,” 3 April 2009. [Online]. Available: <https://bambangherlandi.web.id/elektronika-digital/>. [Diakses 31 Juli 2019].
- [6] B. S. Yoni Mochtiarsa, “Rancangan Kendali Lampu Menggunakan Mikrokontroler ATmega328 Berbasis Sensor Getar,” *Jurnal Informatika SIMANTIK*, vol. 1, no. STMIK Cikarang, p. 41, 2016.
- [7] d. Chrisyantar Hasiholan, “Implementasi Konsep Internet of Things pada Sistem Monitoring,” *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. II, no. 2, pp. 1-2, 2018.
- [8] G. Anggaranie, “Manfaat Internet of Things dan Potensinya dalam Sektor Logistik dan Transportasi,” *Supply Chain Indonesia*, 3 November 2017. [Online]. Available: <http://supplychainindonesia.com/new/manfaat-internet-of-things-dan-potensinya-dalam-sektor-logistik-dan-transportasi/>. [Diakses 17 Agustus 2019].

- [9] F. A. R. R. W. Eka Larasati Choerunisa, "makalah internet of things," 16 Desember 2015. [Online]. Available: <http://amalliafr.blog.st3telkom.ac.id/2015/12/16/makalah-internet-of-things-2/>. [Diakses 06 Agustus 2019].
- [10] D. Kho, "Pengertian Resistor dan Jenis-jenisnya," Teknik elektronika, [Online]. Available: <https://teknikelektronika.com/pengertian-resistor-jenis-jenis-resistor/>. [Diakses 27 juni 2019].
- [11] R. Angga, "Rangkaian Power Supplay Variabel LM317," Skemaku.com, 2014. [Online]. Available: <https://skemaku.com/rangkaian-power-supply-variabel-lm317/>. [Diakses 31 Juli 2019].
- [12] R. Angga, "Cara mengukur resistor dengan multimeter," 2016. [Online]. Available: <https://skemaku.com/cara-mengukur-resistor-dengan-multimeter/>. [Diakses 6 Agustus 2019].
- [13] S. Adi Nugroho, "Dioda," Studio belajar, [Online]. Available: <https://www.studiobelajar.com/dioda/>. [Diakses 27 Juni 2019].
- [14] D. Kho, "Fungsi Dioda dan Cara Mengukurnya," Teknik lektronika, 15 Juli 2017. [Online]. Available: <https://teknikelektronika.com/fungsi-dioda-cara-mengukur-dioda/>. [Diakses 6 Agusutus 2019].
- [15] D. F. Ramadhani, "LAPORAN PRAKTIKUM IIIDAN IVKARAKTERISTIK DIODA DAN TRANSFORMATOR," UNIVERSITAS NEGERI MALANG FAKULTAS TEKNIK JURUSAN TEKNIK ELEKTRO, 28 May 2017. [Online]. Available: https://caridokumen.com/download/laporan-praktikum-3-dan-4-elektronika-karakteristik-dioda-amp-transformator-_5a466c51b7d7bc7b7a082d6f_pdf. [Diakses 16 Juli 2019].
- [16] A. Nahyudin, "Laporan Praktikum Elektronika Dasar Transistor," FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM, 7 Januari 2013. [Online]. Available: http://ahmadnahyudin.blogspot.com/2013/01/laporan-praktikum-elektronika-dasar_7.html. [Diakses 16 Juli 2019].
- [17] A. R. Witjaksono, "Laporan Praktikum Charge-Discharge," Laboratorium Fisika Dasar UPP IPD Universitas Indonesia, 24 Juni 2017. [Online]. Available: https://caridokumen.com/download/laporan-praktikum-charge-discharge-_5a452f28b7d7bc7b7aab1fdb_pdf. [Diakses 16 Juli 2019].

- [18] B. Kiswoyo, “pengertian dan jenis-jenis saklar pada rangkaian elektronika,” Blog Elektronika, 6 Mei 2017. [Online]. Available: <https://www.jalankatak.com/id/saklar/>. [Diakses 17 Juli 2017].
- [19] I. D. Rachma, “Laporan Praktikum Teknik Digital Lengkap,” TEKNIK ELEKTRONIKA FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA 2016, 3 Juni 2017. [Online]. Available: https://caridokumen.com/download/laporan-praktikum-teknik-digital-lengkap-_5a460ce4b7d7bc7b7aeabe9c_pdf. [Diakses 17 Juli 2019].
- [20] D. Kho, “Pengertian Seven Segment Display (Layar Tujuh Segmen),” Teknik elektronika, [Online]. Available: <https://teknikelektronika.com/pengertian-seven-segment-display-layar-tujuh-segmen/>. [Diakses 7 Agustus 2019].
- [21] D. Kho, “Teknik Elektronika,” [Online]. Available: <https://teknikelektronika.com/pengertian-lcd-liquid-crystal-display-prinsip-kerja-lcd/>. [Diakses 8 Agustus 2019].
- [22] E. Sakti, “Cara Kerja Sensor Ultrasonik, Rangkaian, & Aplikasinya,” Mei 2015. [Online]. Available: <https://www.elangsakti.com/2015/05/sensor-ultrasonik.html>. [Diakses 19 Agustus 2019].
- [23] A. Faudin, “Cara mengakses sensor DHT11 menggunakan Arduino,” Nyebarilmu.com, 10 Agustus 2017. [Online]. Available: <https://www.nyebarilmu.com/cara-mengakses-sensor-dht11/>. [Diakses 19 Agustus 2019].
- [24] R. Fachri, “Belajar Dasar Arduino dan Tutorial Pemograman arduino,” Electricity of Dream, 6 September 2016. [Online]. Available: <http://electricityofdream.blogspot.com/2016/09/belajar-dasar-arduino-dan-tutorial.html>. [Diakses 13 Agustus 2019].
- [25] R. Rafi Rahmadi, “Monitor dan Kontrol Penyiraman Tanaman Otomatis Berbasis IoT,” vol. 1, no. Polman Babel, p. 33, 2018.
- [26] T. Com, “Interaksi Blynk dengan Arduino untuk Kontrol Arduino Menggunakan Android,” 8 Desember 2016. [Online]. Available: <https://alvintelkom.blogspot.com/2016/12/interaksi-blynk-dengan-arduino-untuk.html>. [Diakses 4 September 2019].

LAMPIRAN

LAMPIRAN I
DAFTAR RIWAYAT HIDUP

DAFTAR RIWAYAT HIDUP



1. Data Pribadi

Nama lengkap : Muhammad Ega Wahyudi
Tempat dan tanggal lahir : Zed, 30 Juli 1998
Alamat rumah : Jln. Mentok km 25, RT 10 RW 04 Desa Zed,
Kecamatan Mendo Barat, Kab. Bangka, Prov.
Bangka Belitung.
Telp : -
Hp : 0831-7542-8203
Email : egaw925@gmail.com
Jenis kelamin : Laki-laki
Agama : Islam

2. Riwayat Pendidikan

Sekolah Dasar Negeri 09 Desa Zed 2004 – 2010
Madrasah Tsanawiyah Negeri Petaling 2010 – 2013
Sekolah Menengah Kejuruan Negeri 2 Pangkalpinang 2013 – 2016

3. Pendidikan Non Formal

-

Sungailiat, 15 Agustus 2019


Muhammad Ega Wahyudi

DAFTAR RIWAYAT HIDUP



4. Data Pribadi

Nama lengkap : Dimas Pradipta
Tempat dan tanggal lahir : Jakarta, 20 April 1998
Alamat rumah : Jl. Serandang , RT 11 RW 00, Desa Baturusa,
Kecamatan Merawang, Kab. Bangka, Prov.
Kepulauan Bangka Belitung.
Telp : -
Hp : 0857-0971-4881
Email : dimas83868@gmail.com
pradiptadimas23@gmail.com
Jenis kelamin : Laki-laki
Agama : Islam

5. Riwayat Pendidikan

Sekolah Dasar Negeri 1 Desa Baturusa 2004 – 2010
Sekolah Menengah Pertama Negeri 2 Sungailiat 2010 – 2013
Sekolah Menengah Kejuruan Negeri 2 Pangkalpinang 2013 – 2016

6. Pendidikan Non Formal

-

Sungailiat, 15 Agustus 2019

Dimas Pradipta

LAMPIRAN 2
MODUL PIRANTI ELEKTRONIKA

LAMPIRAN 3
MODUL ELEKTRONIKA DIGITAL

LAMPIRAN 4
MODUL MIKROKONTROLER

LAMPIRAN 5
TUTORIAL APLIKASI BLYNK &
MODUL IoT



MODUL PRAKTIKUM PIRANTI ELEKTRONIKA



PERCOBAAN 1 HUKUM OHM

A. Tujuan

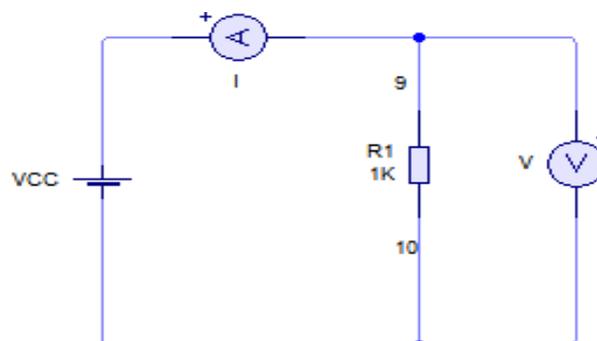
Mahasiswa dapat mempelajari hubungan arus listrik, tegangan listrik, dan hambatan listrik yang lebih dikenal dengan hukum ohm.

B. Komponen Dan Peralatan Yang Digunakan

- Komponen
 - Resistor

- Peralatan
 - Power supply
 - Ampermeter
 - Voltmeter
 - Kabel jumper

C. Gambar Rangkaian Percobaan



Gambar 1. Rangkaian Hukum Ohm

D. Prosedur Percobaan

- Matikan terlebih dahulu catu dayanya.
- Buatlah rangkaian seperti gambar 1. Hubungkan setiap titik dari masing-masing komponen menggunakan kabel jumper.
- Hubungkan alat ukur ampermeter (+) ke power supply (vcc) untuk mengukur arus pada resistor. Kemudian alat ukur ampermeter (-) tersebut dihubungkan juga dengan titik 9 resistor 1.
- Hubungkan titik 10 resistor 1 ke power supply sebagai ground (0V).
- Untuk mengukur tegangan hubungkan alat ukur voltmeter dititik 9 dan 10. Atur power supply dari tegangan 0V sampai 12V. Catat hasil pengukuran tegangan (V) dan arus (I).



E. Data Hasil Percobaan

No	Vcc (Volt)	I (mA)	V (Volt)
1.	0V	0	0
2.	1V	1,0	1
3.	2V	2,1	2
4.	3V	2,9	3
5.	4V	3,9	4
6.	5V	4,8	5
7.	6 V	5,8	6
8.	7 V	6,9	7
9.	8 V	8,1	8
10.	9 V	9	9
11.	10 V	10	10
12.	11 V	11	11
13.	12 V	11,9	12

F. Analisa

- a. Buatlah analisa dan kesimpulan dari hasil percobaan yang sudah dilakukan, bandingkan hasil pengukuran dengan hasil perhitungannya.
- b. Buatlah grafik Tegangan (V) terhadap Arus (I) .



PERCOBAAN 2 RESISTOR SERI DAN PARAREL

A. Tujuan

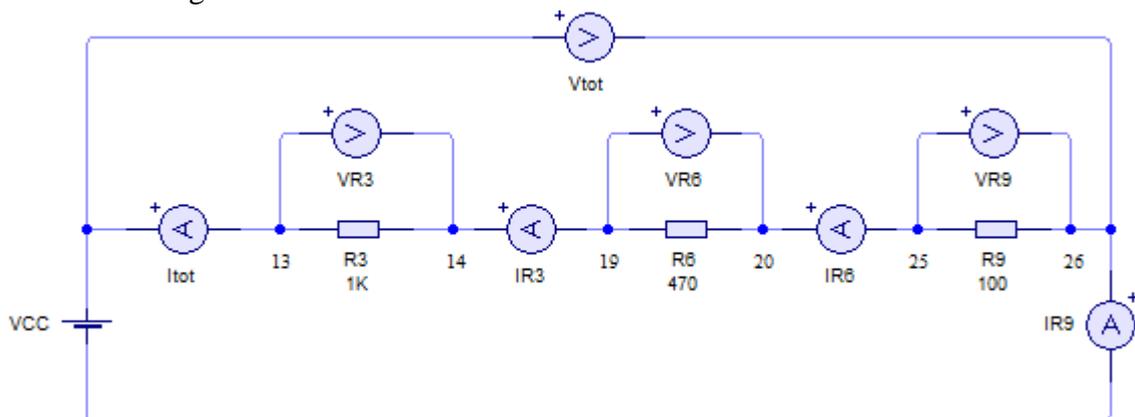
Mahasiswa dapat memahami dan mempelajari rangkaian seri dan paralel pada resistor.

B. Komponen Dan Peralatan Yang Digunakan

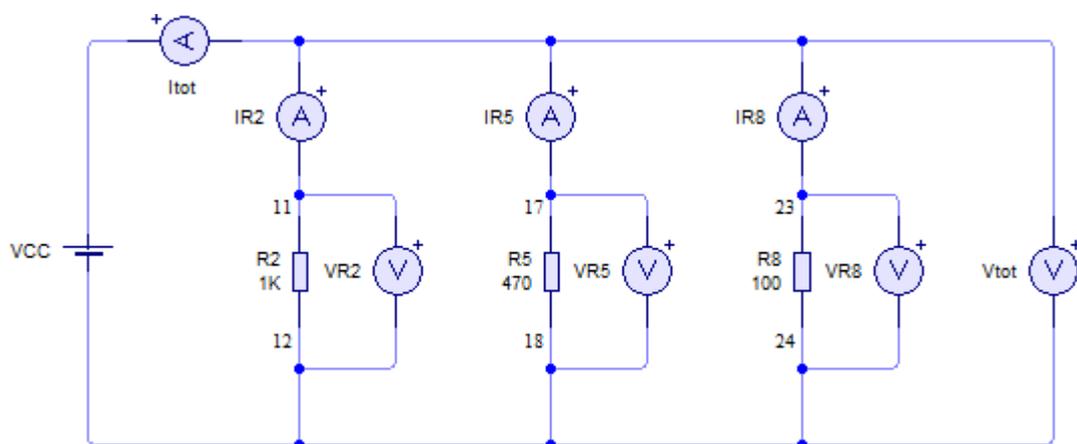
- Komponen
 - Resistor

- Peralatan
 - Power supply
 - Ampermeter
 - Voltmeter
 - Kabel jumper

C. Gambar Rangkaian Percobaan



Gambar 2. Resistor Hubungan Seri



Gambar 3. Resistor Hubungan Paralel



D. Prosedur Percobaan

- Resistor hubungan seri
 - Matikan terlebih dahulu catu dayanya.
 - Buatlah rangkaian seperti gambar 2. Hubungkan setiap titik dari masing-masing komponen menggunakan kabel jumper.
 - Hubungkan alat ukur ampermeter (+) ke VCC untuk mengukur arus total pada rangkaian resistor. Kemudian alat ukur ampermeter (-) tersebut dihubungkan juga dengan titik 13 resistor 3.
 - Hubungkan titik 14 resistor 3 ke alat ukur ampermeter (+) untuk mengukur arus pada resistor 3 (IR3). Kemudian alat ukur ampermeter (-) dihubungkan dengan titik 19 resistor 6. Kemudian titik 20 resistor 6 dihubungkan dengan alat ukur ampermeter (+) untuk mengukur arus pada resistor 6 (IR6). Kemudian alat ukur ampermeter (-) juga dihubungkan dengan titik 25 resistor 9.
 - Hubungkan titik 26 resistor 9 dengan alat ukur ampermeter (+) untuk mengukur arus pada resistor 9 (IR9). Kemudian alat ukur ampermeter (-) juga dihubungkan dengan power supply (0V).
 - Untuk mengukur tegangan di resistor 3 (VR3) hubungkan voltmeter dititik 13 dan 14, tegangan resistor 6 (VR6) hubungkan voltmeter dititik 19 dan 20, dan tegangan di resistor 9 (VR9) hubungkan voltmeter dititik 25 dan 26. Sedangkan untuk mengukur tegangan total (VTot) resistor hubungkan alat ukur voltmeter dititik 13 dan 26. Catat hasil pengukuran tegangan total (V) dan arus rangkaian seri resistor (I).

- Resistor hubungan paralel
 - Matikan terlebih dahulu catu dayanya.
 - Buatlah rangkaian seperti gambar 3. Hubungkan setiap titik dari masing-masing komponen menggunakan kabel jumper.
 - Hubungkan alat ukur ampermeter (+) ke VCC untuk mengukur arus total (Itot) pada rangkaian resistor. Kemudian alat ukur ampermeter (-) tersebut dihubungkan juga ke ampermeter (+) pada IR2, ampermeter (+) pada IR5, dan juga ampermeter (+) pada IR8. Ampermeter IR2 digunakan untuk mengukur arus yang mengalir pada resistor 2, ampermeter IR5 digunakan untuk mengukur arus yang mengalir pada resistor 5, dan ampermeter IR8 digunakan untuk mengukur arus yang mengalir pada resistor 8.
 - Hubungkan titik 11 resistor 2 dengan ampermeter (-) IR2. Kemudian titik 11 juga dihubungkan dengan voltmeter (+) VR2 untuk mengukur tegangan di resistor 2. Titik 12 resistor 2 dihubungkan dengan voltmeter (-) VR2 dan juga dihubungkan dengan VCC (0V).
 - Hubungkan titik 17 resistor 5 dengan ampermeter (-) IR5. Kemudian titik 17 juga dihubungkan dengan voltmeter (+) VR5 untuk mengukur tegangan di resistor 5. Titik 18 resistor 5 dihubungkan dengan voltmeter (-) VR5 dan juga dihubungkan dengan VCC (0V).
 - Hubungkan titik 23 resistor 8 dengan alat ukur ampermeter (-) IR8. Kemudian titik 23 juga dihubungkan dengan voltmeter (+) VR8 untuk mengukur tegangan di resistor 8. Titik 24 resistor 8 dihubungkan dengan voltmeter (-) VR8 dan juga dihubungkan dengan VCC (0V).
 - Untuk mengukur tegangan di resistor 2 (VR2) hubungkan voltmeter dititik 11 dan 12, tegangan resistor 5 (VR5) hubungkan voltmeter dititik 17 dan 18, dan tegangan



di resistor 8 hubungkan voltmeter dititik 23 dan 24. Sedangkan untuk mengukur tegangan total resistor (V_{tot}) hubungkan alat ukur voltmeter dititik 11 dan 24. Atur power supply dari tegangan 0V sampai 12V. Catat hasil pengukuran tegangan dan arus pada table data hasil percobaan.

E. Data Hasil Percobaan

➤ Resistor Hubungan Seri

No	Vin (Volt)	ITOT (mA)	IR3 (mA)	IR6 (mA)	IR9 (mA)	VTOT (Volt)	VR3 (Volt)	VR6 (Volt)	VR9 (Volt)
1.	0V	0	0	0	0	0	0	0	0
2.	1V	0,6	0,6	0,6	0,6	1	0,63	0,29	0,06
3.	2V	1,2	1,2	1,2	1,2	2	1,28	0,6	0,13
4.	3V	1,9	1,9	1,9	1,9	3	1,80	0,99	0,2
5.	4V	2,5	2,5	2,5	2,5	4	2,45	1,2	0,27
6.	5 V	3,1	3,1	3,1	3,1	5	3,2	1,5	0,33
7.	6 V	3,8	3,8	3,8	3,8	6	3,7	1,7	0,38
8.	7 V	4,5	4,5	4,5	4,5	7	2,45	1,2	0,33
9.	8 V	5,0	5,0	5,0	5,0	8	3,7	1,7	0,38
10.	9 V	5,7	5,7	5,7	5,7	9	5,6	2,6	0,55
11.	10 V	6,3	6,3	6,3	6,3	10	6,4	2,9	0,63
12.	11 V	7,0	7,0	7,0	7,0	11	7,0	3,3	0,7
13.	12 V	7,6	7,6	7,6	7,6	12	7,6	3,6	7,6



➤ Resistor Hubungan Paralel

No	Vin (Volt)	ITOT (mA)	IR2 (mA)	IR5 (mA)	IR8 (mA)	VTOT (Volt)	VR2 (Volt)	VR5 (Volt)	VR8 (Volt)
1.	0V	0	0	0	0	0	0	0	0
2.	1V	13	0,9	2,12	9,9	1	1	1	1
3.	2V	26	1,9	4,25	20	2	2	2	2
4.	3 V	39	3,0	6,38	29	3	3	3	3
5.	4 V	52	3,9	8,5	39,9	4	4	4	4
6.	5 V	65,6	4,9	10,6	49,9	5	5	5	5
7.	6 V	78,7	6,0	12,7	59,9	6	6	6	6
8.	7 V	91,9	6,9	14,8	69,9	7	7	7	7
9.	8 V	105	7,9	17,0	79,9	8	8	8	8
10.	9 V	118	8,9	19,1	89,9	9	9	9	9
11.	10 V	131,2	9,9	21,2	99,9	10	10	10	10
12.	11 V	144,4	10,9	23,4	110	11	11	11	11
13.	12 V	157,3	12	25,5	120	12	12	12	12

F. Analisa

- a. Buatlah analisa perhitungan dan kesimpulan dari hasil percobaan yang sudah dilakukan, bandingkan hasil pengukuran dengan hasil perhitungannya.
- b. Berdasarkan data percobaan jelaskan perbedaan rangkaian seri dan paralel pada resistor.



PERCOBAAN 3 DIODA FORWARD BIAS

A. Tujuan

- Mahasiswa dapat memahami karakteristik dioda forward bias berdasarkan hasil pengukuran arus dan tegangan pada dioda forward bias.
- Mahasiswa dapat mengetahui simbol dioda.
- Mahasiswa dapat membuat aplikasi penerapan dioda.

B. Komponen Dan Peralatan Yang Digunakan

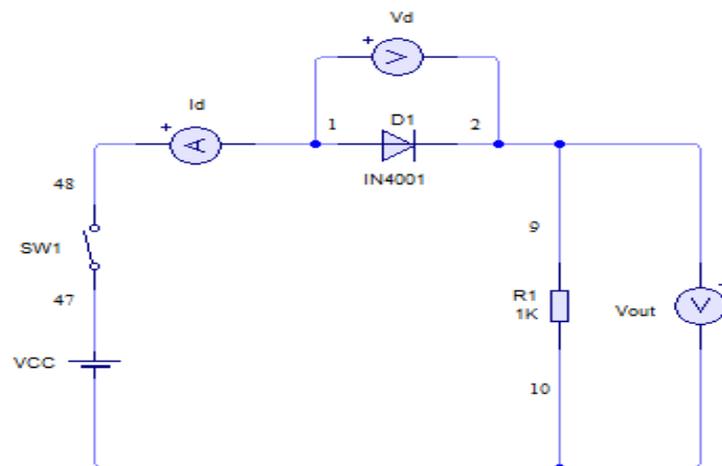
➤ Komponen

- Dioda Silikon IN4001
- Resistor
- Saklar

➤ Peralatan

- Power supply
- Ampermeter
- Voltmeter
- Kabel jumper

C. Gambar Rangkaian Percobaan



Gambar 4. Rangkaian dioda forward bias

D. Prosedur Percobaan

- Matikan terlebih dahulu catu dayanya.
- Buatlah rangkaian seperti gambar percobaan 3. Hubungkan setiap titik dari masing-masing komponen menggunakan kabel jumper.
- Hubungkan titik 47 SW1 dengan power supply (vcc).



- Titik 48 SW1 dihubungkan ke alat ukur amperemeter (+) untuk mengukur arus pada dioda (I_d). Kemudian amperemeter (-) itu juga dihubungkan dengan titik 1 pada dioda.
- Hubungkan titik 2 dioda ke resistor 1 dititik 9 sebagai beban, karena menggunakan resistor $1K \Omega$.
- Hubungkan titik 10 resistor 1 dengan power supply sebagai ground.
- Untuk mengukur tegangan dioda (V_d) hubungkan alat ukur voltmeter dititik 1 dan 2. Sedangkan untuk mengukur tegangan keluaran (V_{out}) hubungkan voltmeter dititik 9 dan 10. Atur power supply dari tegangan 0V sampai 12V. Kemudian catat hasil pengukuran tegangan dioda (V_d), arus dioda (I_d) dan tegangan keluaran (V_{out}) pada data hasil percobaan.

E. Data Hasil Percobaan

No	Vcc (Volt)	I_d (mA)	V_d (Volt)	V_{out} (Volt)
1.	0V	0	0	0
2.	1V	0,29	0,7	0,3
3.	2V	1,4	0,7	1,3
4.	3V	2,5	0,6	2,4
5.	4V	3,2	0,6	3,4
6.	5V	4,3	0,6	4,2
7.	6 V	5,1	0,6	5,2
8.	7 V	6,0	0,69	6,3
9.	8 V	7,0	0,7	7,3
10.	9 V	8,1	0,7	8,4
11.	10 V	9,1	0,7	9,4
12.	11 V	10,3	0,7	10,1



No	Vcc (Volt)	Id (mA)	Vd (Volt)	Vout (Volt)
13.	12 V	11,4	0,7	11,2

F. Analisa

- a. Buatlah analisa dan kesimpulan dari hasil percobaan yang sudah dilakukan, bandingkan hasil pengukuran dengan hasil perhitungan.
- b. Buatlah grafik tegangan dioda (V_d) terhadap arus dioda (I_d).



PERCOBAAN 4 DIODA REVERSE BIAS

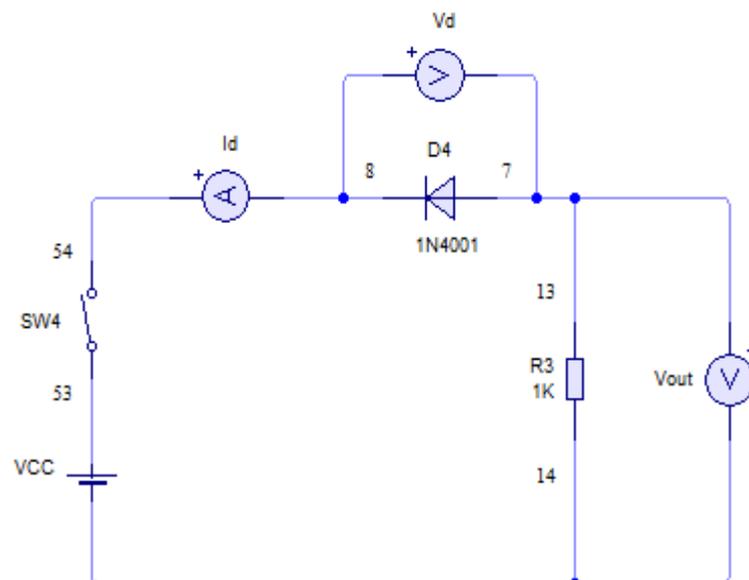
A. Tujuan

- Mahasiswa dapat memahami karakteristik dioda reverse bias berdasarkan hasil pengukuran arus dan tegangan pada dioda reverse bias.
- Mahasiswa dapat mengetahui simbol dioda.
- Mahasiswa dapat membuat aplikasi penerapan dioda.

B. Komponen Dan Peralatan Yang Digunakan

- Komponen
 - Dioda Silikon IN4001
 - Resistor
 - Saklar
- Peralatan
 - Power supply
 - Ampermeter
 - Voltmeter
 - Kabel jumper

C. Gambar Rangkaian Percobaan



Gambar 5. Rangkaian dioda reverse bias



D. Prosedur Percobaan

- Matikan terlebih dahulu catu dayanya.
- Buatlah rangkaian seperti gambar percobaan 4. Hubungkan setiap titik dari masing-masing komponen menggunakan kabel jumper.
- Hubungkan titik 53 SW4 ke power supply (vcc).
- Kemudian titik 54 SW4 dihubungkan ke alat ukur ampermeter (+) untuk mengukur arus pada dioda (I_d). Kemudian ampermeter (-) dihubungkan dengan titik 8 pada dioda 4.
- Titik 7 pada dioda 4 dihubungkan ke titik 13 resistor 3 sebagai beban, karena menggunakan resistor $1K\Omega$.
- Titik 14 resistor 4 dihubungkan ke power supply sebagai ground (0V).
- Untuk mengukur tegangan dioda (V_d) hubungkan alat ukur voltmeter dititik 8 dan 7. Sedangkan untuk mengukur tegangan keluaran (V_{out}) hubungkan voltmeter dititik 13 dan 14. Atur power supply dari tegangan 0V sampai 12V. Kemudian catat hasil pengukuran tegangan dioda (V_d), arus dioda (I_d) dan tegangan keluaran (V_{out}) pada data hasil percobaan.

E. Data Hasil Percobaan

No	Vcc (Volt)	I_d (mA)	V_d (Volt)	V_{out} (Volt)
1.	0V	0	0	0
2.	1V	0	0,9	0
3.	2V	0	1,9	0
4.	3V	0	2,9	0
5.	4V	0	3,9	0
6.	5V	0	4,9	0
7.	6 V	0	5,9	0
8.	7 V	0	7,9	0
9.	8 V	0	7,9	0
10.	9 V	0	8,9	0



No	Vcc (Vout)	Id (mA)	Vd (Volt)	Vout (Volt)
11.	10 V	0	9,9	0
12.	11 V	0	10,9	0
13.	12 V	0	11,9	0

F. Analisa

- a. Buatlah analisa dan kesimpulan dari hasil percobaan yang sudah dilakukan, bandingkan hasil pengukuran dengan hasil perhitungan.
- b. Buatlah grafik tegangan dioda (V_d) terhadap arus dioda (I_d).

PERCOBAAN 5 KARAKTERISTIK TRANSISTOR NPN

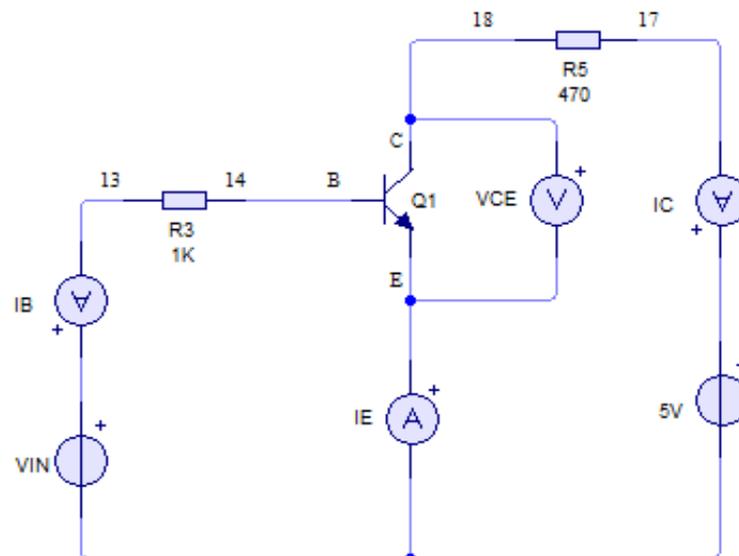
A. Tujuan

- Mahasiswa dapat memahami karakteristik Transistor NPN berdasarkan percobaan yang sudah dilakukan.
- Mahasiswa dapat mengetahui simbol Transistor NPN.
- Mahasiswa dapat membuat aplikasi penerapan Transistor NPN.

B. Komponen Dan Peralatan Yang Digunakan

- Komponen
 - Transistor NPN
 - Resistor 1K
- Peralatan
 - Power supply
 - Ampermeter
 - Voltmeter
 - Kabel jumper

C. Gambar Rangkaian Percobaan



Gambar 6. Karakteristik Transistor NPN



D. Prosuder Percobaan

- Matikan terlebih dahulu catu dayanya.
- Buatlah rangkaian seperti gambar 6. Hubungkan setiap titik dari masing-masing komponen menggunakan kabel jumper.
- Hubungkan alat ukur ampermeter (+) ke power supply (VIN) untuk mengukur arus pada pin basis transistor (IB). Kemudian alat ukur ampermeter (-) tersebut dihubungkan juga ke titik 13 resistor 3.
- Hubungkan titik 14 resistor 3 ke pin Basis Transistor NPN 1.
- Hubungkan ampermeter (+) ke power supply (5V) untuk mengukur arus pada pin Kolektor Transistor (IC). Kemudian ampermeter (-) tersebut dihubungkan juga dengan titik 17 resistor 5.
- Kemudian titik 18 resistor 5 dihubungkan ke pin Kolektor Transistor NPN 1.
- Pin Emittor Transistor NPN 1 dihubungkan dengan alat ukur ampermeter (+) untuk mengukur arus pada pin Emittor Transistor (IE)\
- Kemudian ampermeter (-) tersebut dihubungkan ke 0V.
- Untuk mengukur tegangan Kolektor – Emittor hubungkan alat ukur voltmeter di pin Kolektor dan Emittor Transistor NPN1. Atur power supply dari tegangan 0V sampai 12V. Catat hasil pengukuran di tabel data hasil percobaan.

E. Data Hasil Percobaan

No	Vin (Volt)	IB(mA)	IC(mA)	IE (mA)	VCE (V)
1.	0V	0	0	0	0
2.	1V	0,005	0,09	0,09	0,8
3.	2V	0,005	0,09	0,09	1,8
4.	3V	0,005	0,1	0,1	2,6
5.	4V	0,005	0,1	0,1	3,5
6.	5 V	0,005	0,1	0,1	4,6
7.	6 V	0,005	0,1	0,1	5,6
8.	7 V	0,005	0,1	0,1	6,8
9.	8 V	0,005	0,1	0,1	7,7



No.	Vin (Volt)	IB (mA)	IC (mA)	IE (mA)	VCE (V)
10.	9 V	0,005	0,1	0,1	8,7
11.	10 V	0,005	0,12	0,12	9,8
12.	11 V	0,005	0,12	0,12	10,2
13.	12 V	0,005	0,12	0,12	10,5

F. Analisa

- a. Buatlah analisa perhitungan dan kesimpulan dari hasil percobaan yang sudah dilakukan, bandingkan hasil pengukuran dengan hasil perhitungannya.
- b. Jelaskan prinsip kerja dari piranti Transistor NPN.



PERCOBAAN 6 KARAKTERISTIK TRANSISTOR PNP

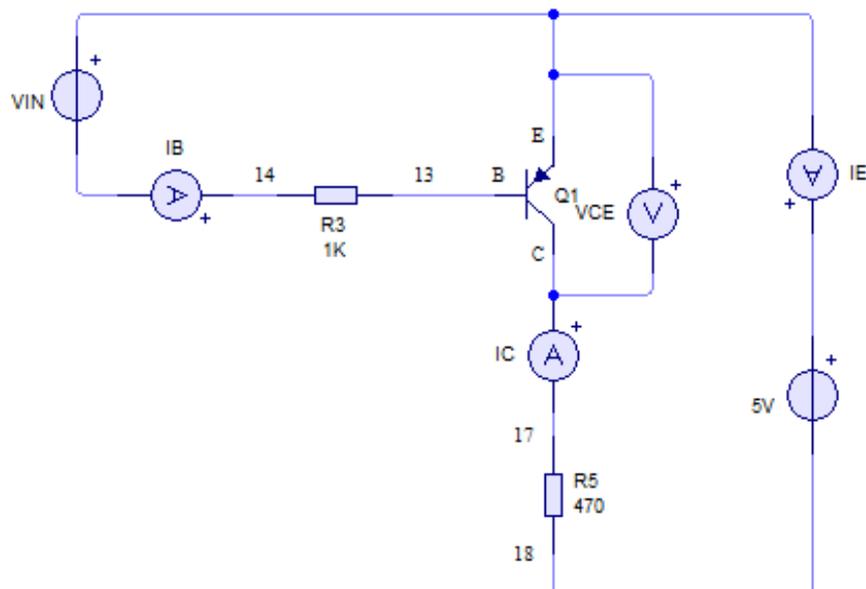
A. Tujuan

- Mahasiswa dapat memahami karakteristik Transistor PNP berdasarkan percobaan yang sudah dilakukan.
- Mahasiswa dapat mengetahui simbol Transistor PNP.
- Mahasiswa dapat membuat aplikasi penerapan Transistor PNP.

B. Komponen Dan Peralatan Yang Digunakan

- Komponen
 - Transistor PNP
 - Resistor 1K
- Peralatan
 - Power supply
 - Ampermeter
 - Voltmeter
 - Kabel jumper

C. Gambar Rangkaian Percobaan



Gambar 7. Karakteristik Transistor PNP



D. Prosuder Percobaan

- Matikan terlebih dahulu catu dayanya.
- Buatlah rangkaian seperti gambar 7. Hubungkan setiap titik dari masing-masing komponen menggunakan kabel jumper.
- Pin Basis Transistor PNP 1 dihubungkan ke titik 13 resistor 3. Titik 14 resistor 3 dihubungkan dengan ampermeter (+). Sedangkan untuk ampermeter (-) dihubungkan dengan sumber tegangan 0V (-). Ampermeter ini digunakan untuk mengukur arus pada pin Basis (IB) Transistor PNP 1.
- Hubungkan sumber tegangan VIN (VCC) ke pin Emittor Transistor PNP.
- Pin Colector Transistor PNP dihubungkan dengan alat ukur ampermeter (+) untuk mengukur arus pada pin Colector (IC) Transistor PNP. Ampermeter (-) dihubungkan ke titik 17 resistor 5. Titik 18 pada resistor 5 dihubungkan ke sumber tegangan 0V.
- Sumber tegangan 5V dihubungkan ke ampermeter (+), ampermeter (-) dihubungkan ke pin Emittor untuk mengukur arus pada pin Emittor (IE).
- Untuk mengukur tegangan Kolektor – Emittor hubungkan alat ukur voltmeter di pin Kolektor dan Emittor Transistor NPN1. Atur power supply dari tegangan 0V sampai 12V. Catat hasil pengukuran di tabel data hasil percobaan.

E. Data Hasil Percobaan

No	Vin (Volt)	IB(mA)	IC(mA)	IE (V)	VCE (V)
1.	0V	0	0	0	0
2.	1V	0,005	0,09	0,09	0,8
3.	2V	0,005	0,09	0,09	1,8
4.	3V	0,005	0,1	0,1	2,6
5.	4V	0,005	0,1	0,1	3,5
6.	5 V	0,005	0,1	0,1	4,6
7.	6 V	0,005	0,1	0,1	5,6
8.	7 V	0,005	0,1	0,1	6,8
9.	8 V	0,005	0,1	0,1	7,7



No	Vin (Volt)	IB (mA)	IC (mA)	IE (mA)	VCE (V)
10.	9 V	0,005	0,1	0,1	8,7
11.	10 V	0,005	0,12	0,12	9,8
12.	11 V	0,005	0,12	0,12	10,2
13.	12 V	0,005	0,12	0,12	10,5

F. Analisa

- Buatlah analisa perhitungan dan kesimpulan dari hasil percobaan yang sudah dilakukan, bandingkan hasil pengukuran dengan hasil perhitungannya.
- Jelaskan prinsip kerja dari piranti Transistor PNP.
- Jelaskan perbedaan antara Transistor tipe NPN dan Transistor tipe PNP.



MODUL PRAKTIKUM ELEKTRONIKA DIGITAL

PERCOBAAN 1 GERBANG LOGIKA AND

A. Tujuan

- Mahasiswa dapat memahami prinsip dasar gerbang logika AND.
- Dapat menganalisa tabel kebenaran gerbang logika AND.
- Dapat mengetahui simbol dari gerbang logika OR.

B. Komponen Dan Peralatan Yang Digunakan

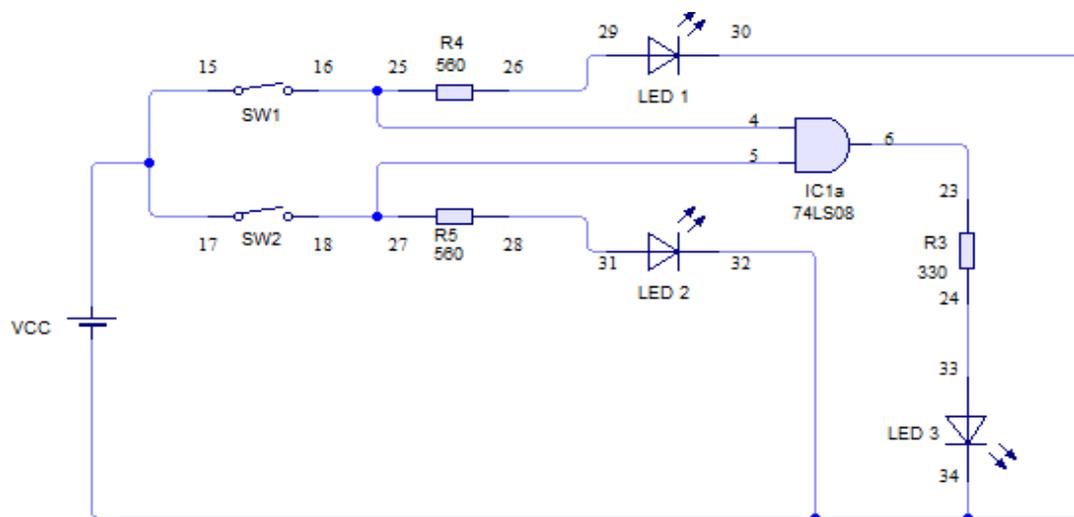
➤ Komponen

- Resistor
- Led
- Switch
- ICTTL tipe 74LS08 (AND GATE)

➤ Peralatan

- Power supply
- Kabel jumper

C. Gambar Rangkaian Percobaan



Gambar 1. Rangkaian gerbang Logika AND

D. Prosedur Percobaan

- Matikan terlebih dahulu catu dayanya.
- Hubungkan setiap titik berdasarkan nomor pin yang ada pada setiap komponen menggunakan kabel jumper berdasarkan gambar 1.
- Hubungkan pin 14 ke VCC sebagai sumber tegangan (+).
- Hubungkan pin 7 dengan 0 V sebagai ground.



- Ujilah rangkaian yang ada pada Gambar 1. dan buatlah tabel kebenaran dari tegangan masukan terhadap tegangan keluarannya.
- Analisa rangkaian gerbang logika AND.
- Buatlah kesimpulan berdasarkan hasil pengujian rangkaian gerbang logika AND

E. Data Hasil Percobaan

Tabel Kebenaran

VINPUT		Output
SW1	SW2	
0V	0V	0
0V	5V	0
5V	0V	0
5V	5V	1

F. Analisa

Buatlah analisa dan kesimpulan dari hasil percobaan yang sudah dilakukan.

PERCOBAAN 2 GERBANG LOGIKA OR

A. Tujuan

- Mahasiswa dapat memahami prinsip dasar gerbang logika OR.
- Dapat menganalisa tabel kebenaran gerbang logika OR.
- Dapat mengetahui simbol dari gerbang logika OR.

B. Komponen Dan Peralatan Yang Digunakan

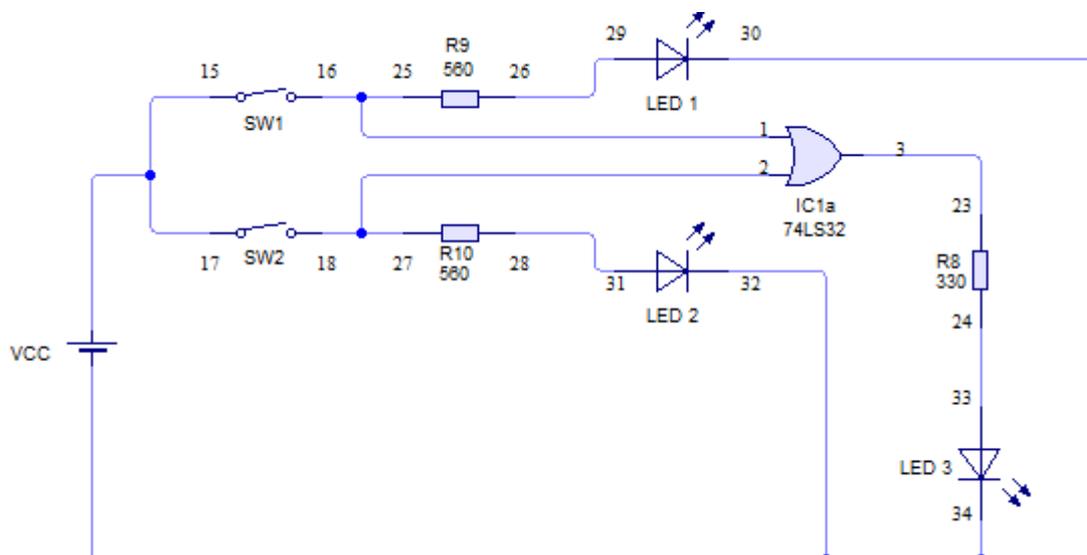
➤ Komponen

- Resistor
- Led
- Switch
- IC TTL tipe 74LS32 (OR GATE)

➤ Peralatan

- Power supply
- Kabel jumper

C. Gambar Rangkaian Percobaan



Gambar 2. Rangkaian gerbang Logika OR

D. Prosedur Percobaan

- Matikan terlebih dahulu catu dayanya.
- Hubungkan setiap titik berdasarkan nomor pin yang ada pada setiap komponen menggunakan kabel jumper berdasarkan gambar 2.
- Hubungkan pin 14 dengan VCC sebagai sumber tegangan (+).



- Hubungkan pin 7 dengan 0 V sebagai ground.
- Ujilah rangkaian yang ada pada Gambar 2. dan buatlah tabel kebenaran dari tegangan masukan terhadap tegangan keluarannya.
- Analisa rangkaian gerbang logika OR.
- Buatlah kesimpulan berdasarkan hasil pengujian rangkaian gerbang logika OR.

E. Data Hasil Percobaan

Tabel Kebenaran

VINPUT		Output
SW1	SW2	
0V	0V	0
0V	5V	1
5V	0V	1
5V	5V	1

F. Analisa

Buatlah analisa dan kesimpulan dari hasil percobaan yang sudah dilakukan.



PERCOBAAN 3 GERBANG LOGIKA NOT

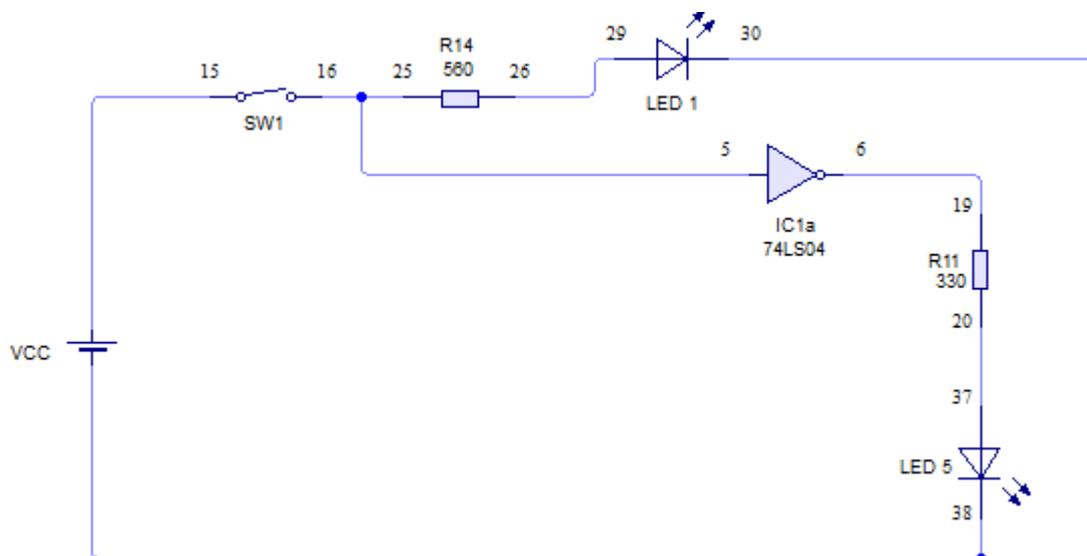
A. Tujuan

- Mahasiswa dapat memahami prinsip dasar gerbang logika NOT.
- Dapat menganalisa tabel kebenaran gerbang logika NOT.
- Dapat mengetahui simbol dari gerbang logika NOT.

B. Komponen Dan Peralatan Yang Digunakan

- Komponen
 - Resistor
 - Led
 - Switch
 - IC TTL tipe 74LS04 (NOT GATE)
- Peralatan
 - Power supply
 - Kabel jumper

C. Gambar Rangkaian Percobaan



Gambar 3. Rangkaian gerbang Logika NOT

D. Prosedur Percobaan

- Matikan terlebih dahulu catu dayanya.
- Hubungkan setiap titik berdasarkan nomor pin yang ada pada setiap komponen menggunakan kabel jumper berdasarkan gambar 3.
- Hubungkan pin 14 dengan VCC sebagai sumber tegangan (+).
- Hubungkan pin 7 dengan 0 V sebagai ground.



- Ujilah rangkaian yang ada pada Gambar 3. dan buatlah tabel kebenaran dari tegangan masukan terhadap tegangan keluarannya.
- Analisa rangkaian gerbang logika NOT.
- Buatlah kesimpulan berdasarkan hasil pengujian rangkaian gerbang logika NOT.

E. Data Hasil Percobaan

Tabel Kebenaran

SW1	OUTPUT
0V	0
5V	1

F. Analisa

Buatlah analisa dan kesimpulan dari tabel kebenaran yang sudah dilakukan.

PERCOBAAN 4 GERBANG LOGIKA NAND

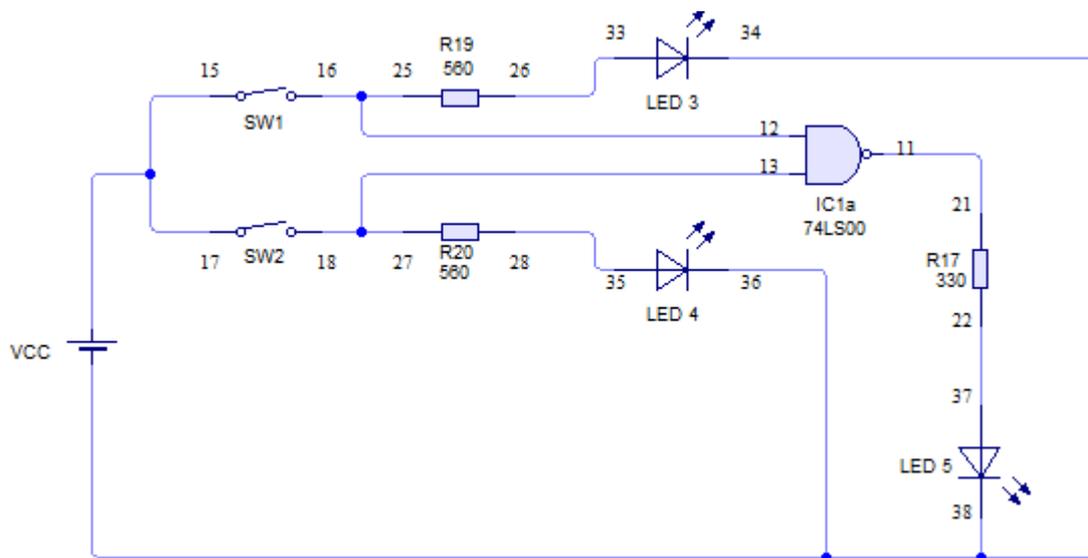
A. Tujuan

- Mahasiswa dapat memahami prinsip dasar gerbang logika NAND.
- Dapat menganalisa tabel kebenaran gerbang logika NAND.

B. Komponen Dan Peralatan Yang Digunakan

- Komponen
 - Resistor
 - Led
 - Switch
 - IC TTL tipe 74LS00 (NAND GATE)
- Peralatan
 - Power supply
 - Kabel jumper

C. Gambar Rangkaian Percobaan



Gambar 4. Rangkaian gerbang logika NAND

D. Prosedur Percobaan

- Matikan terlebih dahulu catu dayanya.
- Hubungkan setiap titik berdasarkan nomor pin yang ada pada setiap komponen menggunakan kabel jumper berdasarkan gambar 4.
- Hubungkan pin 14 dengan VCC sebagai sumber tegangan (+).
- Hubungkan pin 7 dengan 0 V sebagai ground.
- Ujilah rangkaian yang ada pada Gambar 4. dan buatlah tabel kebenaran dari tegangan masukan terhadap tegangan keluarannya.



- Analisa rangkaian gerbang logika NAND.
- Buatlah kesimpulan berdasarkan hasil pengujian rangkaian gerbang logika NAND.

E. Data Hasil Percobaan

Tabel Kebenaran

VINPUT		Output
SW1	SW2	
0V	0V	1
0V	5V	1
5V	0V	1
5V	5V	0

F. Analisa

Buatlah analisa dan kesimpulan dari hasil percobaan yang sudah dilakukan.

PERCOBAAN 5 GERBANG LOGIKA NOR

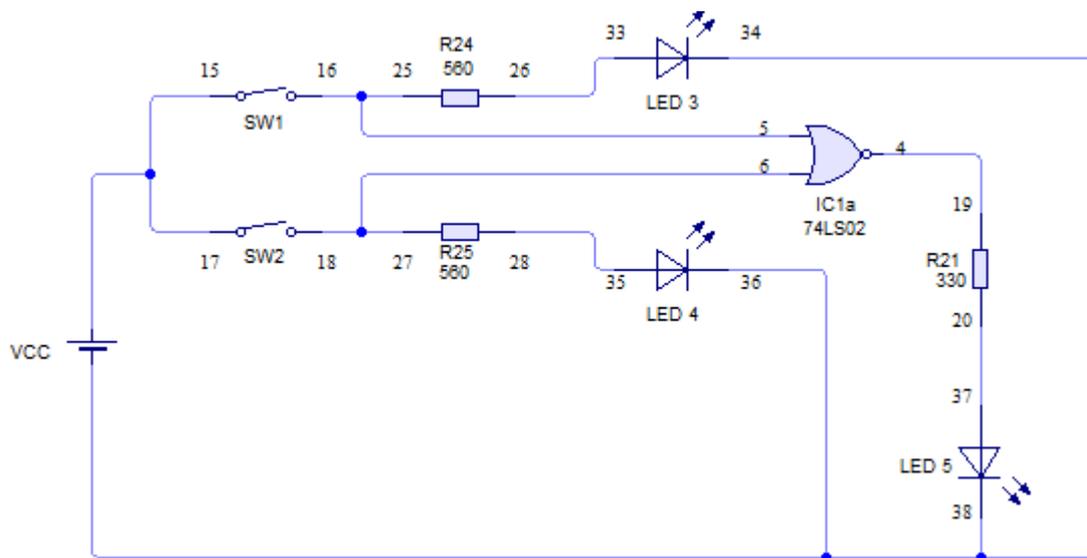
A. Tujuan

- Mahasiswa dapat memahami prinsip dasar gerbang logika NOR.
- Dapat menganalisa tabel kebenaran gerbang logika NOR.

B. Komponen Dan Peralatan Yang Digunakan

- Komponen
 - Resistor
 - Led
 - Switch
 - IC TTL tipe 74LS02 (NOR GATE)
- Peralatan
 - Power supply
 - Kabel jumper

C. Gambar Rangkaian Percobaan



Gambar 5. Rangkaian gerbang logika NOR

D. Prosedur Percobaan

- Matikan terlebih dahulu catu dayanya.
- Hubungkan setiap titik berdasarkan nomor pin yang ada pada setiap komponen menggunakan kabel jumper berdasarkan gambar 5.
- Hubungkan pin 14 dengan VCC sebagai sumber tegangan (+).
- Hubungkan pin 7 dengan 0 V sebagai ground.
- Ujilah rangkaian yang ada pada Gambar 5. dan buatlah tabel kebenaran dari tegangan masukan terhadap tegangan keluarannya.



- Analisa rangkaian gerbang logika NOR.
- Buatlah kesimpulan berdasarkan hasil pengujian rangkaian gerbang logika NOR.

E. Data Hasil Percobaan

Tabel Kebenaran

V INPUT		Output
SW1	SW2	
0V	0V	1
0V	5V	0
5V	0V	0
5V	5V	0

F. Analisa

Buatlah analisa dan kesimpulan dari hasil percobaan yang sudah dilakukan.

PERCOBAAN 6 GERBANG LOGIKA XOR

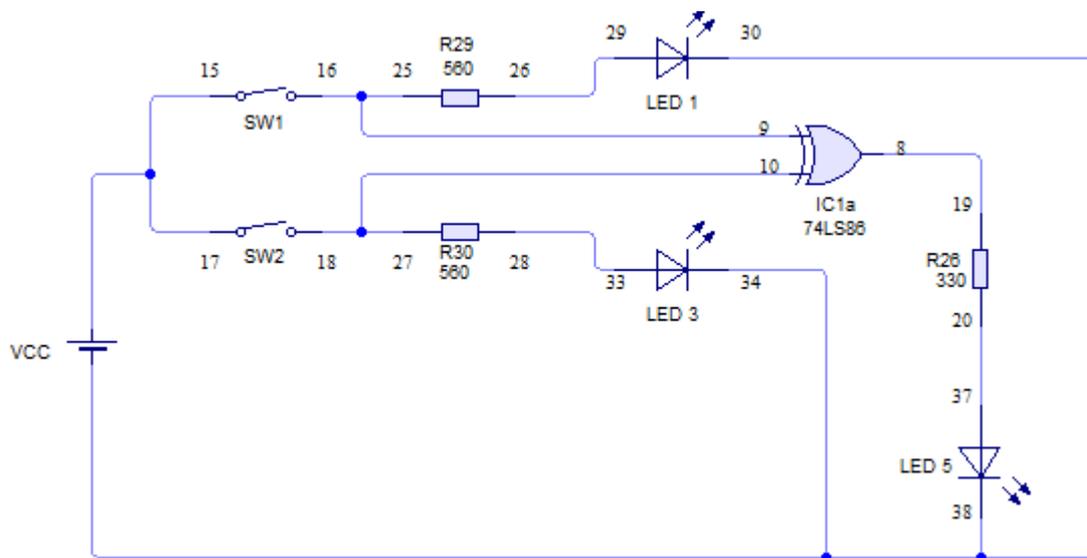
A. Tujuan

- Mahasiswa dapat memahami prinsip dasar gerbang logika XOR.
- Dapat menganalisa tabel kebenaran gerbang logika XOR.

B. Komponen Dan Peralatan Yang Digunakan

- Komponen
 - Resistor
 - Led
 - Switch
 - IC TTL tipe 74LS86 (XOR GATE)
- Peralatan
 - Power supply
 - Kabel jumper

C. Gambar Rangkaian Percobaan



Gambar 6. Rangkaian gerbang logika XOR

D. Prosedur Percobaan

- Matikan terlebih dahulu catu dayanya.
- Hubungkan setiap titik berdasarkan nomor pin yang ada pada setiap komponen menggunakan kabel jumper berdasarkan gambar 6.
- Hubungkan pin 14 dengan VCC sebagai sumber tegangan (+).
- Hubungkan pin 7 dengan 0 V sebagai ground.
- Ujilah rangkaian yang ada pada Gambar 6. dan buatlah tabel kebenaran dari tegangan masukan terhadap tegangan keluarannya.



- Analisa rangkaian gerbang logika XOR.
- Buatlah kesimpulan berdasarkan hasil pengujian rangkaian gerbang logika XOR.

E. Data Hasil Percobaan

Tabel Kebenaran

V INPUT		Output
SW1	SW2	
0V	0V	0
0V	5V	1
5V	0V	1
5V	5V	0

F. Analisa

Buatlah analisa dan kesimpulan dari hasil percobaan yang sudah dilakukan.



MODUL PRAKTRIKUM MIKROKONTROLLER



PERCOBAAN 1 APLIKASI PORT SEBAGAI OUTPUT MENGGUNAKAN LED

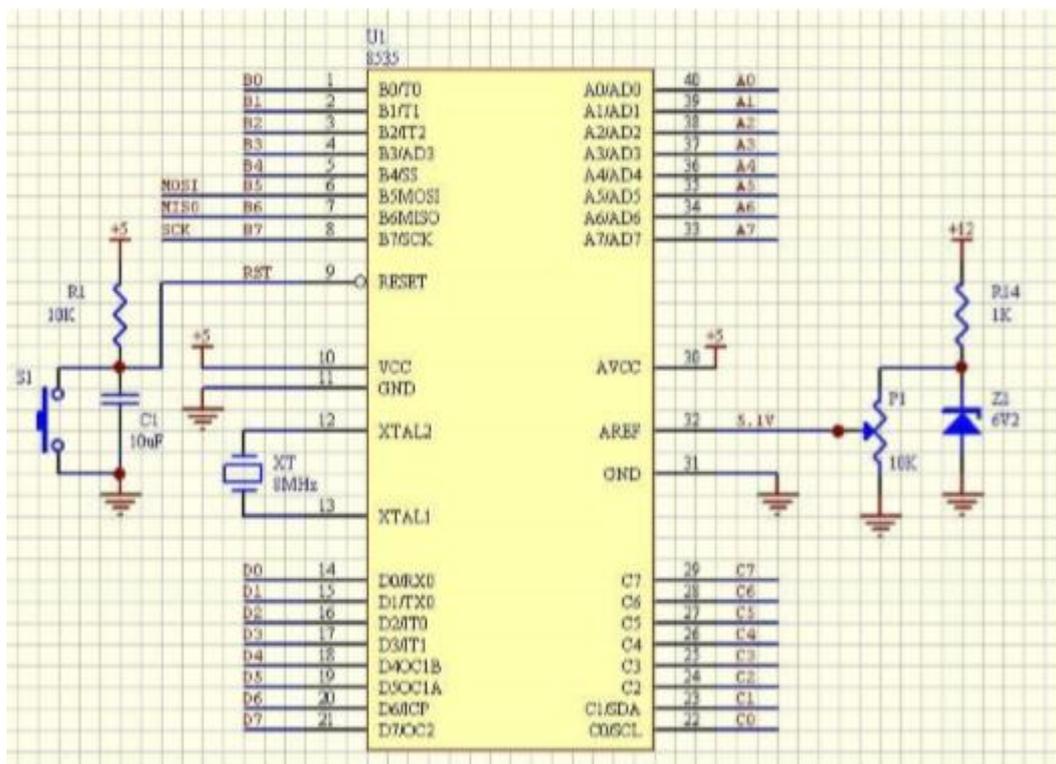
A. Tujuan

- Mahasiswa bisa menggunakan software CodeVisionAVR berbasis bahasa pemrograman C dengan prosedur yang benar.
- Mahasiswa bisa mengendalikan nyala LED sebagai port output mikrokontroler dengan teknik dan kombinasi yang diinginkan secara baik dan benar.

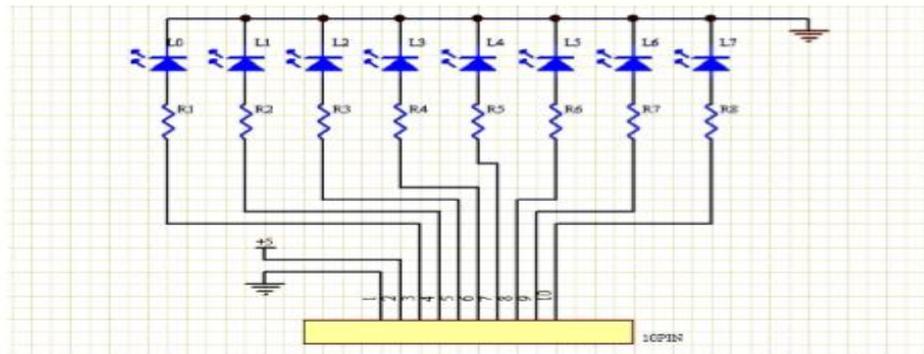
B. Komponen Dan Peralatan Yang Digunakan

- Modul mikrokontroler AVR
- Komputer

C. Gambar Rangkaian Percobaan



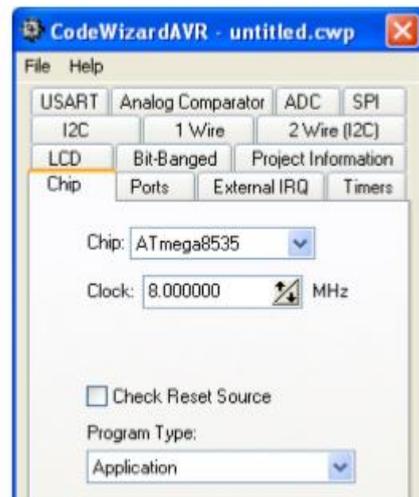
Gambar 1.1 IC ATmega 8535



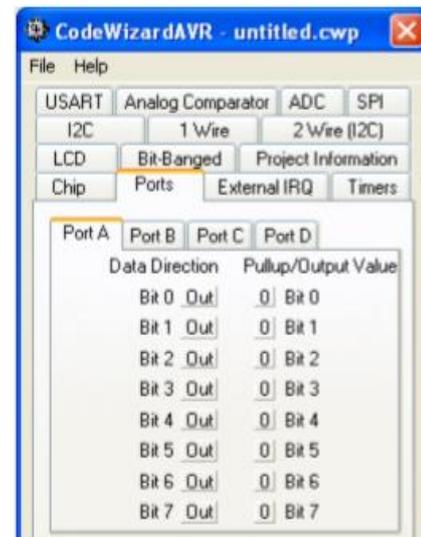
Gambar 1.2 Output LED

D. Prosedur Percobaan

- Buka program CodeVisionAVR dengan menekan klik ganda icon  pada dekstop komputer. Hubungkan setiap titik menggunakan kabel jumper.
- Pilih File – New – Project - Yes.
- Selanjutnya pilih AVR8 (ATtiny,Atmega,AT90) – Ok, jika menggunakan IC ATmega.
- Kemudian Pilih chip ATmega8535 dan clock 8MHz, dengan mengikuti isi sesuai gambar 1.3.
- Lalu setting port-A sebagai output dengan cara memilih Ports, lalu pada Port A ubah In menjadi Out mulai dari Bit 0 sampai dengan Bit 7.



Gambar 1.3 Wizard Chip

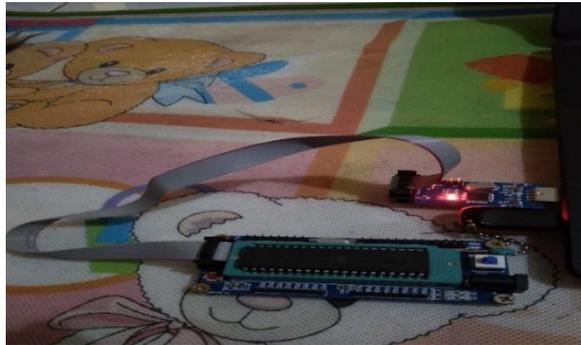


Gambar 1.4 Wizard Port

- Jika selesai, pilih File – Generate, Save and Exit.
- Arahkan direktori ke dalam folder yang anda buat sebelumnya, lalu ketik nama file dengan Percobaan1. Tekan Save. Lakukan sampai tiga kali dengan nama folder yang sama.

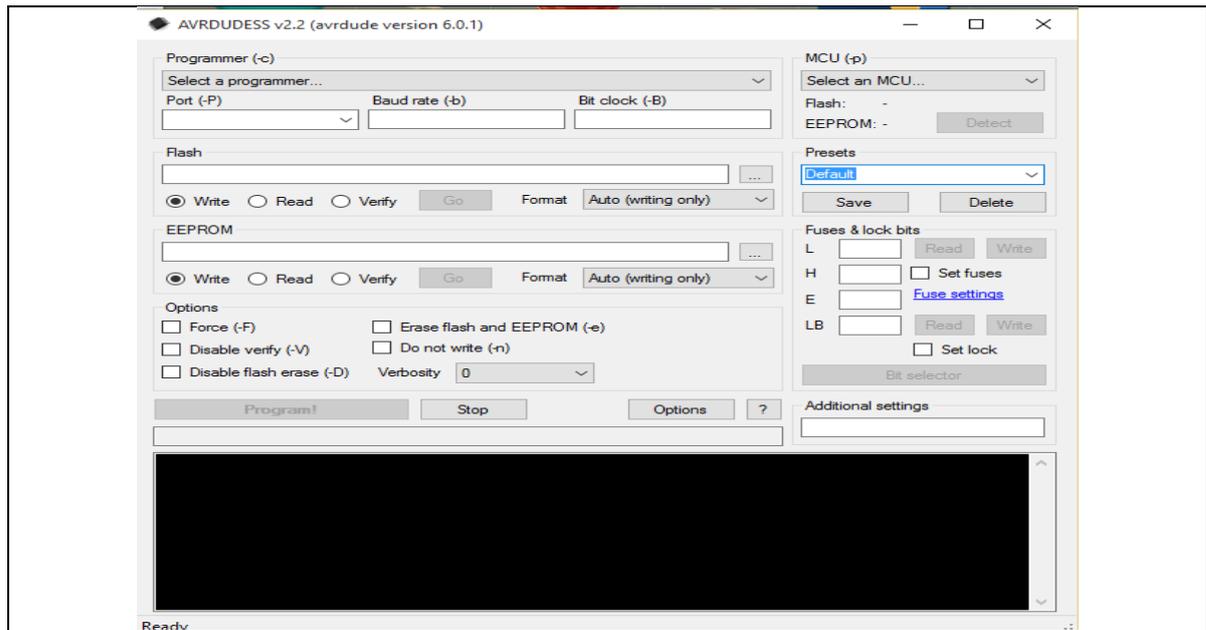


- Ketikkan program yang diinginkan atau sesuai contoh pada daftar program percobaan 1.
- Jika sudah selesai mengetik program, Lakukan compile dengan cara menekan gambar  atau tombol F9. Jika tidak ada error tekan OK. Bila ada error, maka periksa program yang anda ketik. Lalu ulangi lagi.
- Langkah selanjutnya yaitu mendownload program yang sudah dibuat ke IC Atmega 8535
- Hubungkan kabel USBASP downloader dari terminal USB komputer ke downloader pada trainer praktikum. Hubungkan PORTA ke modul LED dengan menggunakan kabel jumper. Hubungkan kabel jumper dari PORTA0 ke pin 0, PORTA1 ke pin 1, PORTA2 ke pin 2 secara berurutan sampai PORTA7 ke pin 7. Hubungkan + ke VCC dan – ke Ground.



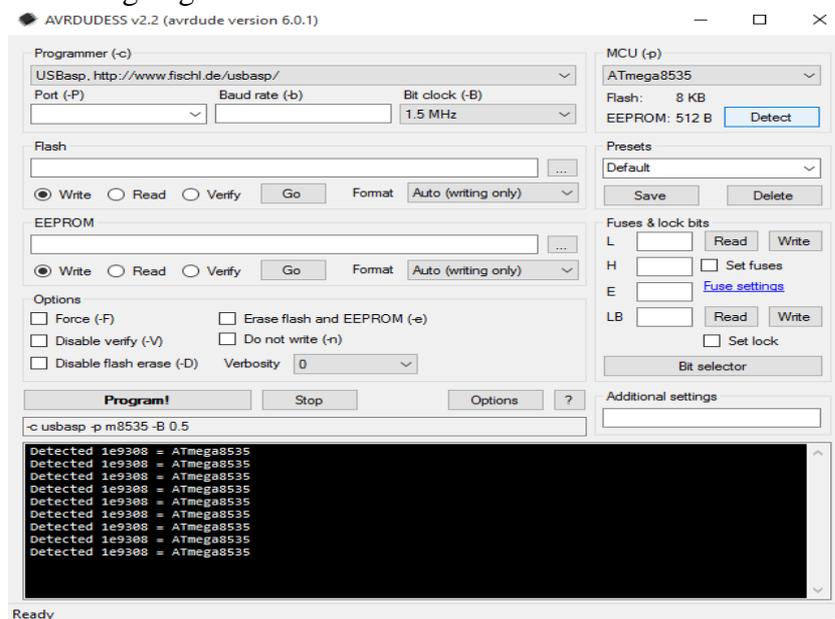
Gambar 1.5 Downloader Mikrokontroler

- Untuk mengisi program yang telah dibuat ke dalam chip, maka terlebih dahulu buka software AVRDUDES dengan menekan klik ganda icon  pada desktop Komputer.
- Selanjutnya akan muncul seperti gambar 1.5 Tampilan awal software AVRDUDES.



Gambar 1.6 Tampilan Awal Software AVRDUDES

- o Kemudian pilih IC yang digunakan di menu Select an MCU. Karena menggunakan IC ATmega8535 jadi pilih IC ATmega8535. Setelah itu pilih menu Select a Programmer, pilih USBasp. Sebagai port downloader yang digunakan.
- o Setelah selesai, klik Detect untuk melihat IC ATmega8535 terhubung atau tidak ke komputer. Jika terhubung maka di kolom bawah warna hitam akan ada tulisan "Detected 1e9308 = ATmega8535". Jika tidak terhubung periksa kembali kabel usb. Lalu ulangi lagi.

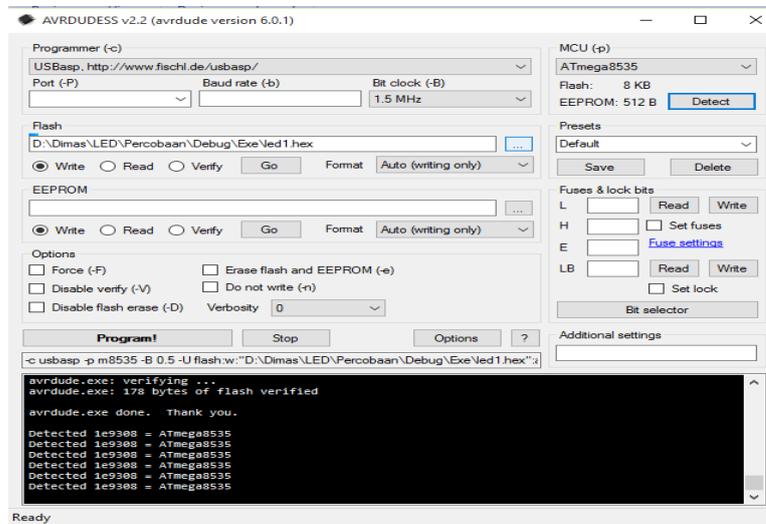


Gambar 1.7 konfigurasi IC ATmega

- o Jika sudah terhubung langkah selanjutnya yaitu mengupload program yang sudah dibuat kedalam IC ATmega8535, dengan memilih flash – pilih tanda  cari file program yang sudah disimpan. Kemudian di dalam file tersebut pilih debug – exe-



percobaan1. Jika sudah baru klik “Program” di software AVRDUDES, tunggu sampai proses selesai uploading selesai.



Gambar 1.8 Proses Uploading Program

- Hubungkan konektor port-A dengan konektor LED. Amati yang terjadi dan dianalisa!
- Lakukan untuk semua contoh program percobaan. Lalu selesaikan tugas yang diberikan dengan menunjukkan hasilnya kepada dosen yang mengampuh!

Daftar program percobaan 1:

1. Membuat LED menyala semua

```
#include <mega8535.h>
void main(void)
{
    PORTA=0x00;
    DDRA=0xFF;
    while (1)
    {
        PORTA=0xFF;
    };
}
```

2. Membuat LED menyala berkedip

```
#include <mega8535.h>
#include <delay.h>
void main(void)
{
    PORTA=0x00;
    DDRA=0xFF;
    while (1)
    {
```



```
        PORTA=0xFF;
        delay_ms(200);
        PORTA=0x00;
        delay_ms(200);
    }
}
```

3. Membuat LED menyala berjalan

```
#include <mega8535.h>
void main(void)
{
    PORTA=0x00;
    DDRA=0xFF;
    while (1)
    {
        PORTA=0b00000001;
        delay_ms(200);
        PORTA=0b00000010;
        delay_ms(200);
        PORTA=0b00000100;
        delay_ms(200);
        PORTA=0b00001000;
        delay_ms(200);
        PORTA=0b00010000;
        delay_ms(200);
        PORTA=0b00100000;
        delay_ms(200);
        PORTA=0b01000000;
        delay_ms(200);
        PORTA=0b10000000;
        delay_ms(200);
    }
}
```

4. Membuat LED menyala berjalan (Metode Matematis)

```
#include <mega8535.h>
#include <delay.h>
#include <math.h>

void main(void)
{
    int i,a;
    PORTA=0xff;
    DDRA=0xff;
    while (1)
    {
        for (i=0;i<=7;i++)
        {
            a=pow(2,i);
            PORTA=~a;
            delay_ms(500);
        }
    }
}
```



```
    }  
  }  
}
```

5. Membuat LED menyala berjalan (metode geser bit)

```
#include <mega8535.h>  
void delay(int n)  
{  
  int i;  
  for(i=0;i<n;i++)  
  {  
    int b;  
    for(b=0;b<=150;i++);  
  }  
}  
  
void kiri()  
{  
  unsigned char a=0x7f;  
  int i;  for(i=0;i<8;i++)  
  {  
    a=((a<<1)|(a>>7)); // instruksi geser bit  
    PORTA=a;  
    delay(200);  
  }  
}  
  
void main(void)  
{  
  PORTA=0x00;  
  DDRA=0xFF;  
  while (1)  
  {  
    kiri();  
  }  
}
```

6. Membuat LED menyala berjalan (metode lookup table)

```
#include <mega8535.h>  
#include <delay.h>  
  
void main(void)  
{  
  unsigned char led[8] =  
  {0x01,0x02,0x04,0x08,0x10,0x20,0x40,0x80};  
  int a;  
  PORTA=0x00;  
  DDRA=0xFF;  
  while (1)  
  {  
    for (a=0;a<=7;a++)
```



```
        {
            PORTA = led[a];
            delay_ms(200);
        }
    }
}
```

Tugas !

a. Buatlah program LED menyala berselingan

```
#include <mega8535.h>

void main(void)
{
    DDRA =0xFF;
    PORTA=0x00;

    while (1)
    {
        PORTA=0b01010101;
    }
}
```

b. Buatlah program LED menyala flip-flop (empat mati dan hidup bergantian) sebanyak 5kali kemudian LED padam semua

```
#include <mega8535.h>
#include <delay.h>

void main(void)
{
    int i,a;

    DDRA =0xFF;
    PORTA=0x00;

    for(i=0;i<=4;i++)
    {
        a=0b00001111;
        PORTA=a;
        delay_ms(100);
        PORTA=~a;
        delay_ms(100);
    }

    if(i==5)
    {
        PORTA=0b00000000;
    }
}
```



```
}
```

c. Buatlah program LED menyala geser kiri-ke-kanan sebanyak 5kali kemudian geser dari kanan-ke-kiri sebanyak 5kali, terus – menerus

```
#include <mega8535.h>
#include <delay.h>
#include <math.h>

void main(void)
{
    int a,b,c;
    DDRA=0xFF;
    PORTA=0x00;

    while (1)
    {
        for(a=0;a<=4;a++)
        {
            for(b=0;b<=7;b++)
            {
                c=pow(2,b);
                PORTA=c;
                delay_ms(50);
            }
        }
        for(a=0;a<=4;a++)
        {
            for(b=7;b>=0;b--)
            {
                c=pow(2,b);
                PORTA=c;
                delay_ms(50);
            }
        }
    }
}
```

d. Buatlah program LED menyala mencacah(counter) 8bit naik kemudian turun

```
#include <mega8535.h>
#include <delay.h>

void main(void)
{
    int a;
    DDRA=0xFF;
    PORTA=0x00;

    while (1)
    {
        for(a=0;a<=255;a++)
        {
```



```
        PORTA=a;
        delay_ms(200);
    }

    for(a=0;a<255;a++)
    {
        PORTA=~a;
        delay_ms(200);
    }
}
```



PERCOBAAN 2 INPUT SWITCH DAN OUTPUT MENGGUNAKAN LED

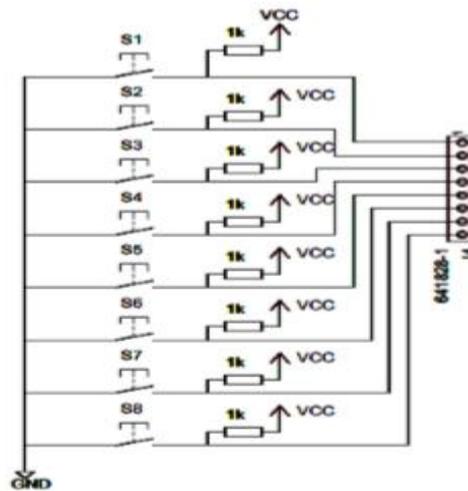
A. Tujuan

- Mahasiswa bisa menggunakan software CodeVisionAVR berbasis bahasa pemrograman C dengan prosedur yang benar.
- Mahasiswa mengerti teknik pembacaan input pada pin mikrokontroler menggunakan push button baik secara hardware maupun pemrogramannya.

B. Komponen Yang Digunakan

- Modul mikrokontroler AVR
- Komputer

C. Rangkaian Aplikasi Switch



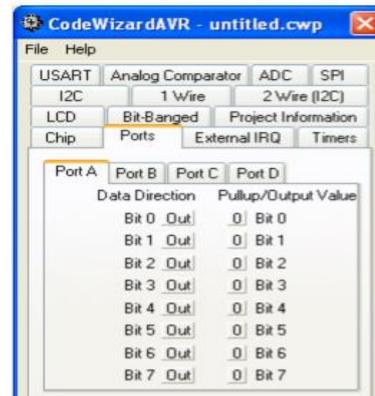
Gambar 2.1 Input Push Button

D. Prosedur Percobaan

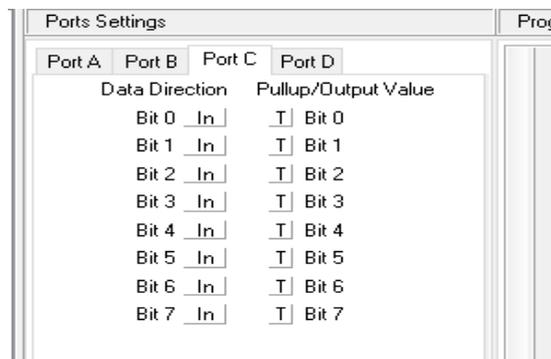
- Buka program CodeVisionAVR dengan menekan klik ganda icon  pada dekstop komputer.
- Pilih File – New – Project - Ok.
- Selanjutnya pilih Yes untuk menggunakan CodeWizardAVR.
- Pilih chip ATMega8535 dan clock 8MHz, dengan cara mengikuti isi sesuai gambar 2.2 & gambar 2.3
- Lalu setting port-A sebagai output dengan cara memilih Ports, lalu pada Port A ubah In menjadi Out mulai dari Bit 0 sampai dengan Bit 7.
- Kemudian setting port-C sebagai input dengan cara memilih Ports, biarkan saja Port C dalam kondisi In dari Bit 0 sampai Bit 7.



Gambar 2.2 Wizard Chip



Gambar 2.3 Wizard Port A



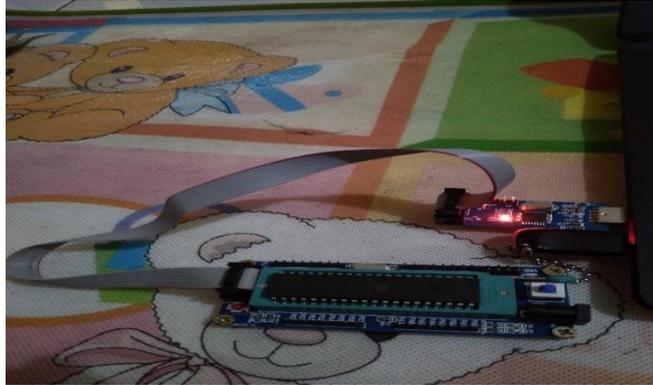
Gambar 2.4 Wizard Port C

- Jika selesai, pilih File – Generate, Save and Exit.
- Arahkan direktori ke dalam folder yang anda buat sebelumnya, lalu ketik nama file dengan Percobaan2. Tekan Save. Lakukan sampai tiga kali dengan nama folder yang sama.
- Ketikkan program yang diinginkan atau sesuai contoh pada daftar program percobaan2.
- Jika sudah selesai mengetik program, Lakukan compile dengan cara menekan gambar  atau tombol F9. Jika tidak ada error tekan OK. Bila ada error, maka periksa program yang anda ketik. Lalu ulangi lagi.
- Hubungkan kabel USBASP downloader dari terminal USB komputer ke downloader pada trainer praktikum sesuai gambar 2.5. Hubungkan PORTA ke modul LED dengan menggunakan kabel jumper. Hubungkan kabel jumper dari PORTA0 ke pin 0, PORTA1 ke pin 1, PORTA2 ke pin 2 secara berurutan sampai PORTA7 ke pin 7. Hubungkan + ke VCC dan – ke Ground.
- Hubungkan PORTC ke modul Push Button dengan menggunakan kabel jumper. Hubungkan kabel jumper dari PORTC0 ke pin 0, PORTC1 ke pin 1, PORTC2 ke pin 2 secara berurutan sampai PORTC7 ke pin 7. Hubungkan + ke VCC dan – ke Ground.



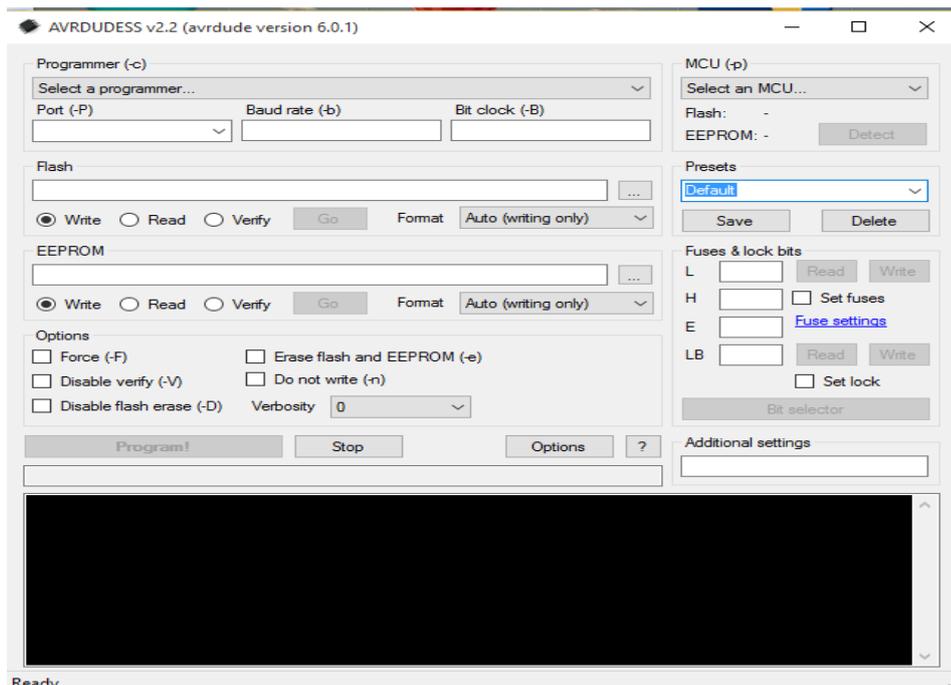
- Untuk mengisi program yang telah dibuat ke dalam chip, maka terlebih dahulu

buka software AVRDUDES dengan menekan klik ganda icon  pada desktop Komputer.



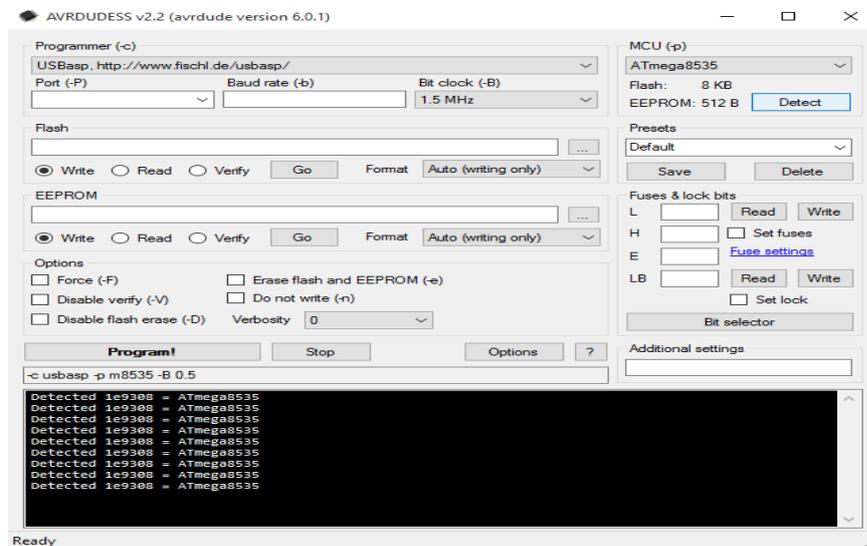
Gambar 2.5 Downloader Mikrokontroler

- Selanjutnya akan muncul seperti gambar 2.6 Tampilan awal software AVRDUDES.



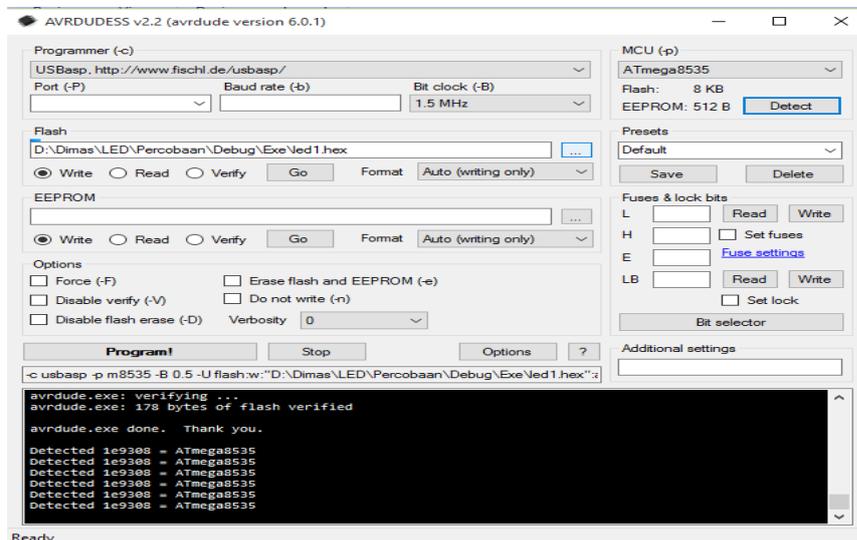
Gambar 2.6 Tampilan Awal Software AVRDUDES

- Kemudian pilih IC yang digunakan di menu Select an MCU. Karena menggunakan IC ATmega8535 jadi pilih IC ATmega8535. Setelah itu pilih menu Select a Programmer, pilih USBasp. Sebagai port downloader yang digunakan.
- Setelah selesai klik Detect untuk melihat IC ATmega8535 terhubung atau tidak ke komputer. Jika terhubung maka di kolom bawah warna hitam akan ada tulisan "Detected 1e9308 = ATmega8535". Jika tidak terhubung periksa kembali kabel usb. Lalu ulangi lagi.



Gambar 2.7 konfigurasi IC ATMega

- Jika sudah terhubung langkah selanjutnya yaitu mengupload program yang sudah dibuat kedalam IC ATMega8535, dengan mencari file yang sudah disimpan. Kemudian di dalam file tersebut pilih debug – exe- percobaan2. Jika sudah baru klik “Program” di software AVRDUDES, tunggu sampai proses selesai uploading selesai.



Gambar 2.8 Proses Uploading Program

- Hubungkan konektor port-A dengan konektor LED, port C dihubungkan ke Push Button. Amati yang terjadi dan dianalisa!
- Lakukan untuk semua contoh program percobaan. Lalu selesaikan tugas yang diberikan dengan menunjukkan hasilnya kepada dosen yang mengampuh

**Daftar program percobaan 2:****1. Membuat LED menyala sesuai dengan kondisi switch**

```
#include <mega8535.h>
void main(void)
{
    PORTA=0x00;
    DDRA=0xFF;
    DDRC=0x00;
    while (1)
    {
        PORTA=PINC;
    };
}
```

2. LED menyala semua atau padam semua bergantung kondisi S1

```
#include <mega8535.h>
void main(void)
{
    DDRA=0xFF;
    PORTA=0x00;
    DDRC=0x00;
    PORTC=0x00;

    while (1)
    {
        if(PINC.0==1)
            PORTA=0;
        else PORTA=255;
    };
}
```

3. Membuat LED menyala berjalan bergantung kondisi S2

```
#include <mega8535.h>
#include <delay.h>
#include <math.h>

void main(void)
{
    int i,a;
    DDRA=0xFF;
    PORTA=0x00;
    DDRC=0x00;
    PORTC=0xFF;
    while(1)
    {
        while (PINC.1==0)
        {
            for (i=0;i<=7;i++)
```



```

        {
            a=pow(2,i);
            PORTA=a;
            delay_ms(100);
        }
    };
};
}

```

Tugas !

- 1. Buatlah program dengan syarat jika S1 ATAU S2= ON, DAN yang lain OFF, maka LED menyala berjala. Buatlah program LED menyala flip-flop (empat mati dan hidup bergantian) sebanyak 5kali kemudian LED padam semua**

```

#include <mega8535.h>
#include <delay.h>
#include <math.h>

void main(void)
{
    int a,d;
    DDRA=0xFF;
    PORTA=0x00;
    DDRC=0x00;
    PORTC=0xFF;

    while (1)
    {
        if((PINC.0==0 ||PINC.1==0) && (PINC.2==1 && PINC.3==1
        && PINC.4==1 && PINC.5==1 && PINC.6==1 && PINC.7==1))
        {
            for(a=0;a<=7;a++)
            {
                d=pow(2,a);
                PORTA=d;
                delay_ms(50);
            }
        }
    }
}

```

- 2. Buatlah program dengan syarat jika S1 = ON maka LED menyala berjalan dari kiri ke kanan. Namun, bila S1 di-OFF-kan maka LED langsung berubah arah dari kanan ke kiri. Kondisi awal seluruh switch = OFF dan LED padam.**

```

#include <mega8535.h>
#include <delay.h>
#include <math.h>

void main(void)
{

```



```
int d=0;
int c;
int b;

DDRA=0xFF;
PORTA=0x00;

DDRC=0x00;
PORTC=0xFF;

while (1)
{
    if(PINC.0==0)
    {
        d++;
        delay_ms(100);
    }

    if(d==1)
    {
        for(b=0;b<=7;b++)
        {
            c=pow(2,b);
            PORTA=c;
            delay_ms(50);
        }
    }
    if(d==2)
    {
        for(b=7;b>=0;b--)
        {
            c=pow(2,b);
            PORTA=c;
            delay_ms(50);
        }
    }
    if(d==3)
    {
        d=1;
    }
}
}
```

PERCOBAAN 3 APLIKASI PORT SEBAGAI OUTPUT MENGGUNAKAN 7-SEGMENT

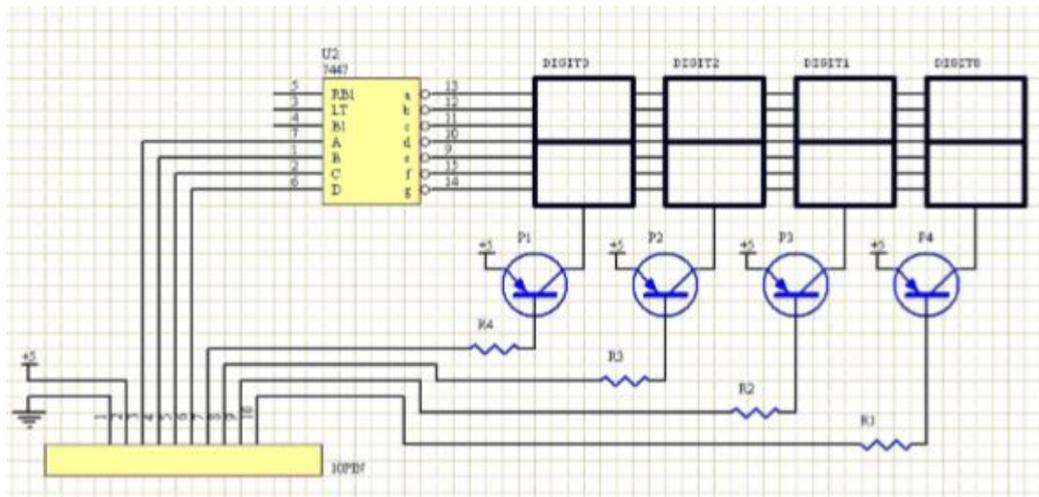
A. Tujuan

- Mahasiswa bisa menggunakan software CodeVisionAVR berbasis bahasa pemrograman C dengan prosedur yang benar.
- Mahasiswa bisa mengendalikan nyala 7segment sebagai port keluaran mikrokontroler sesuai dengan karakter interfacing.

B. Komponen Yang Digunakan

- Modul mikrokontroler AVR
- Komputer

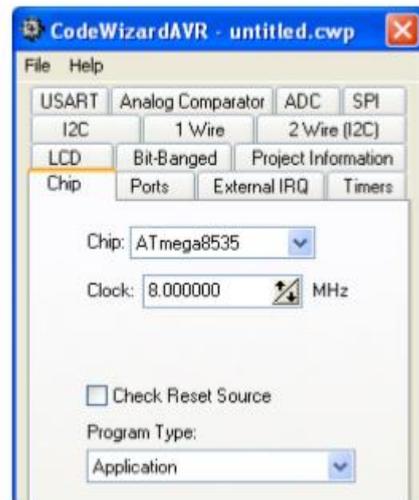
C. Rangkaian Percobaan



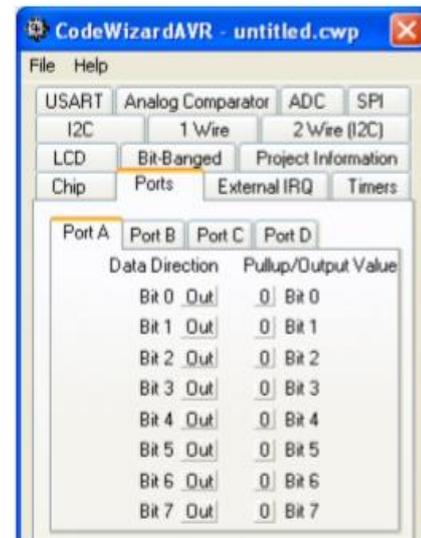
Gambar 3.1 Output Seven Segment

D. Prosedur Percobaan

- Buka program CodeVisionAVR dengan menekan klik ganda icon  pada dekstop komputer. Hubungkan setiap titik menggunakan kabel jumper.
- Pilih File – New – Project - Yes.
- Selanjutnya pilih AVR8 (ATtiny,Atmega,AT90) – Ok, jika menggunakan IC ATmega.
- Kemudian Pilih chip ATmega8535 dan clock 8MHz, dengan mengikuti isi sesuai gambar 3.2.
- Lalu setting port-A sebagai output dengan cara memilih Ports, lalu pada Port A ubah In menjadi Out mulai dari Bit 0 sampai dengan Bit 7.

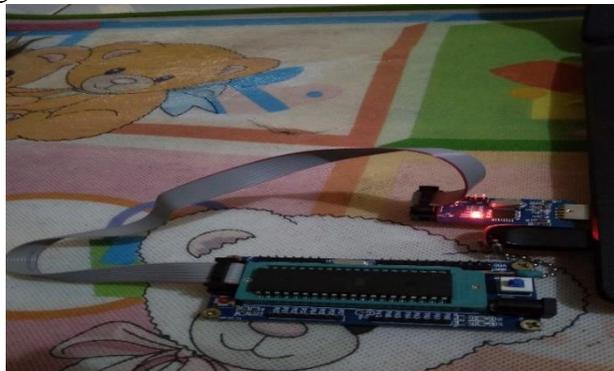


Gambar 3.2 Wizard Chip



Gambar 3.3 Wizard Port

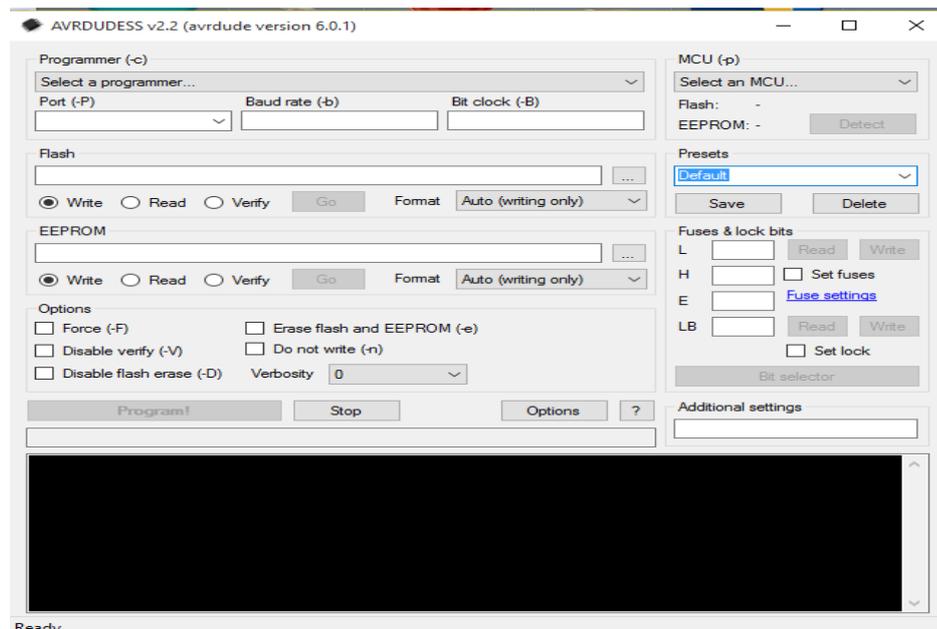
- Jika selesai, pilih File – Generate, Save and Exit.
- Arahkan direktori ke dalam folder yang anda buat sebelumnya, lalu ketik nama file dengan Percobaan3. Tekan Save. Lakukan sampai tiga kali dengan nama folder yang sama.
- Ketikkan program yang diinginkan atau sesuai contoh pada daftar program percobaan3.
- Jika sudah selesai mengetik program, Lakukan compile dengan cara menekan gambar  atau tombol F9. Jika tidak ada error tekan OK. Bila ada error, maka periksa program yang anda ketik. Lalu ulangi lagi.
- Langkah selanjutnya yaitu mendownload program yang sudah dibuat ke IC Atmega 8535
- Hubungkan kabel USBASP downloader dari terminal USB komputer ke downloader pada trainer praktikum. Hubungkan PORTA ke modul Seven Segment dengan menggunakan kabel jumper. Hubungkan kabel jumper dari PORTA0 ke pin 0, PORTA1 ke pin 1, PORTA2 ke pin 2 secara berurutan sampai PORTA7 ke pin 7. Hubungkan + ke VCC dan – ke Ground.



Gambar 3.4 Downloader Mikrokontroler

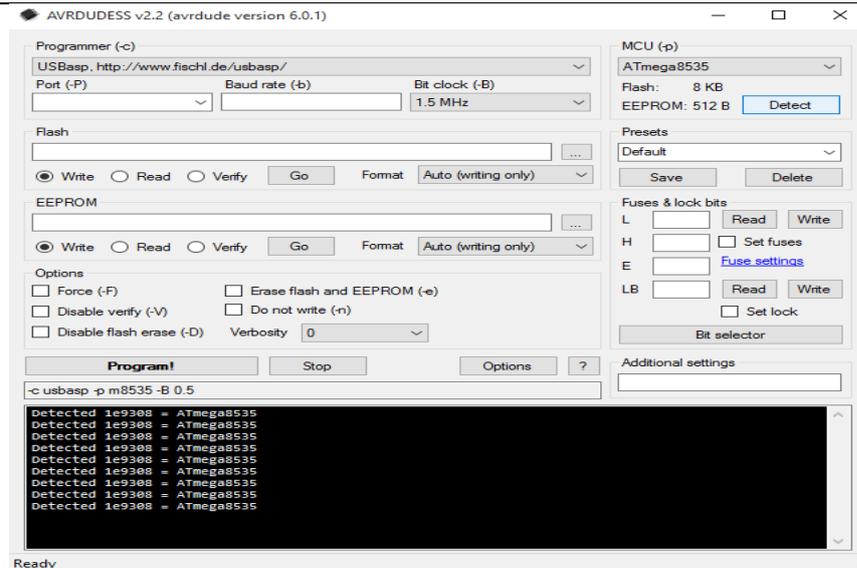


- Untuk mengisi program yang telah dibuat ke dalam chip, maka terlebih dahulu buka software AVRDUDES dengan menekan klik ganda icon  pada desktop Komputer.
- Selanjutnya akan muncul seperti gambar 3.5 Tampilan awal software AVRDUDES.



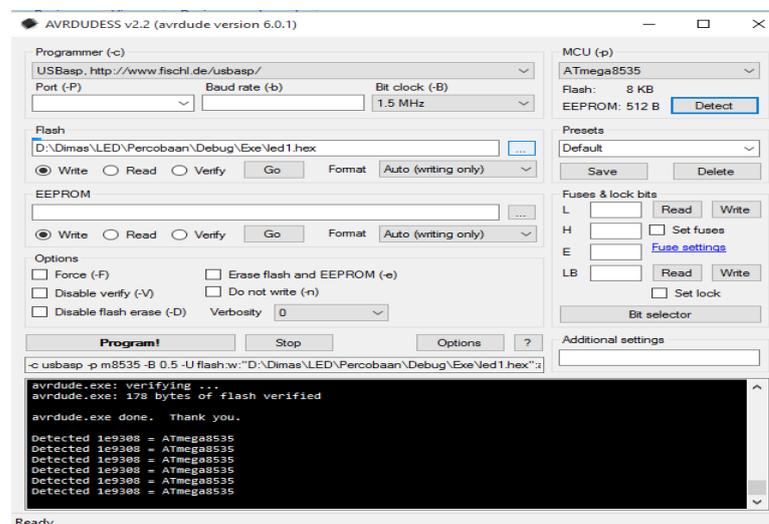
Gambar 3.5 Tampilan Awal Software AVRDUDES

- Kemudian pilih IC yang digunakan di menu Select an MCU. Karena menggunakan IC ATmega8535 jadi pilih IC ATmega8535. Setelah itu pilih menu Select a Programmer, pilih USBasp. Sebagai port downloader yang digunakan.
- Setelah selesai, klik Detect untuk melihat IC ATmega8535 terhubung atau tidak ke komputer. Jika terhubung maka di kolom bawah warna hitam akan ada tulisan "Detected 1e9308 = ATmega8535". Jika tidak terhubung periksa kembali kabel usb. Lalu ulangi lagi.



Gambar 3.6 konfigurasi IC ATmega

- Jika sudah terhubung langkah selanjutnya yaitu mengupload program yang sudah dibuat kedalam IC ATmega8535, dengan memilih flash – pilih tanda  cari file program yang sudah disimpan. Kemudian di dalam file tersebut pilih debug – executable3. Jika sudah baru klik “Program” di software AVRDUDES, tunggu sampai proses selesai uploading selesai.



Gambar 3.7 Proses Uploading Program

- Hubungkan konektor port-A ke modul Seven Segment. Amati yang terjadi dan dianalisa!
- Lakukan untuk semua contoh program percobaan. Lalu selesaikan tugas yang diberikan dengan menunjukkan hasilnya kepada dosen yang mengampuh!

Daftar program percobaan 3:

1. Menampilkan angka "0" pada semua 7Segment



```
// D7  D6  D5  D4  D3 D2 D1 D0
// SEG3 SEG2 SEG1 SEG0 D  C  B  A
// 1   1   1   1   0  0  0  0 = F0H
```

```
#include <mega8535.h>
```

```
void main(void)
```

```
{
    DDRA =0xFF;
    PORTA=0x00;
```

```
while (1)
```

```
{
    {
        PORTA=0xF0;
    }
}
```

2. Menampilkan angka "0" pada satu buah 7Segment saja

```
#include <mega8535.h>
```

```
void main(void)
```

```
{
    DDRA=0xFF;
    PORTA=0x00;
```

```
while (1)
```

```
{
    {
        PORTA=0x40;
    }
}
```

3. Menampilkan angka "3" pada satu buah 7Segment saja

```
#include <mega8535.h>
```

```
void main(void)
```

```
{
    DDRA=0xFF;
    PORTA=0x00;
```

```
while (1)
```

```
{
    {
        PORTA=0x83;
    }
}
```

4. Menampilkan angka "0123"

```
#include <mega8535.h>
```

```
#include <delay.h>
```



```
void main(void)
{
    DDRA = 0xFF;
    PORTA= 0x00;
    while(1)
    {
        PORTA= 0x80;
        delay_ms(5);

        PORTA= 0x41;
        delay_ms(5);

        PORTA= 0x22;
        delay_ms(5);

        PORTA= 0x13;
        delay_ms(5);
    }
}
```

5. Menampilkan angka "0123"

```
#include <mega8535.h>
#include <delay.h>

void main(void)
{
    int rib, rat, pul, sat;
    DDRA = 0xFF;
    while(1)
    {
        sat = 3;
        pul = 2;
        rat = 1;
        rib = 0;
        PORTA= 0x80 | rib;
        delay_ms(1);
        PORTA= 0x41 | rat;
        delay_ms(1);
        PORTA= 0x22 | pul;
        delay_ms(1);
        PORTA= 0x13 | sat;
        delay_ms(1);
    }
}
```

Tugas !

1. Buatlah program konter up (0 s.d 1000)

```
#include <mega8535.h>
#include <delay.h>
```



```
#include <math.h>

void main(void)
{
    int sat,pul, rat, rib, data, i;
    DDRA=0xFF;
    PORTA=0x00;

    for(i=0;i<=1000;i++)
    {
        data =i;
        sat = data % 10;
        pul = data / 10;
        pul = pul % 10;
        rat = data / 100;
        rat = rat % 10;
        rib = data / 1000;

        PORTA= 0x80 | sat;
        delay_ms(1000);
        PORTA= 0x40 | pul;
        delay_ms(1000);
        PORTA= 0x20 | rat;
        delay_ms(1000);
        PORTA= 0x10 | rib;
        delay_ms(1000);
    }

    while (1)
    {
        PORTA= 0x80;
        delay_ms(1000);
        PORTA= 0x40;
        delay_ms(1000);
        PORTA= 0x20;
        delay_ms(1000);
        PORTA= 0x10;
        delay_ms(1000);
    }
}
```

2. Buatlah program konter barang dengan asusmi S1 sebagai sensornya. Jika S1 naik 1kali (naik – turun), maka 7segment terjadi increment. S2 untuk clear data

```
#include <mega8535.h>
#include <delay.h>
#include <math.h>

void main(void)
{
    int sat,pul, rat, rib, data, i=0;
    DDRA=0xFF;
```



```
PORTA=0x00;
DDRC=0x00;
PORTC=0xFF;

while (1)
{
    if(PINC.0==0)
    {
        i++;
        while(PINC.0==0)
        {
            i=i;
        }
    }
    if(PINC.1==0)
    {
        i=0;
    }
    data =i;
    sat = data % 10;
    pul = data / 10;
    pul = pul % 10;
    rat = data / 100;
    rat = rat % 10;
    rib = data / 1000;

    PORTA= 0x80 | sat;
    delay_ms(1);
    PORTA= 0x40 | pul;
    delay_ms(1);
    PORTA= 0x20 | rat;
    delay_ms(1);
    PORTA= 0x10 | rib;
    delay_ms(1);
}
}
```



PERCOBAAN 4 INTERFACING DAN PEMROGRAMAN LCD 2 X 16

A. Tujuan

- Mahasiswa bisa menggunakan software CodeVisionAVR berbasis bahasa pemrograman C dengan prosedur yang benar.
- Mahasiswa mengerti teknik interfacing LCD 2x16 untuk mode 4 bit dan 8 bit.
- Mahasiswa bisa memprogram LCD 2x16 untuk mode 4 bit dan 8 bit dengan baik dan benar

B. Komponen Yang Digunakan

- a. Modul mikrokontroler AVR
- b. Komputer

C. Teori Dasar Karakter LCD 2x16

Modul LCD Character dapat dengan mudah dihubungkan dengan mikrokontroler. LCD yang akan kita praktikumkan ini berbasis ASCII dan mempunyai lebar display 2 baris 16 kolom atau biasa disebut sebagai LCD Character 2x16, dengan 16 pin konektor.



Gambar 4.1 Tampilan LCD

Display karakter pada LCD diatur oleh pin EN, RS dan RW:

Jalur EN dinamakan Enable. Jalur ini digunakan untuk memberitahu LCD bahwa anda sedang mengirimkan sebuah data. Untuk mengirimkan data ke LCD, maka melalui program EN harus dibuat logika LOW "0" dan set pada dua jalur kontrol yang lain RS dan RW. Ketika dua jalur yang lain telah siap, set EN dengan logika HIGH "1" dan tunggu untuk sejumlah waktu tertentu (sesuai dengan datasheet dari LCD tersebut) dan berikutnya set EN ke logika LOW "0" lagi.

Jalur RS adalah jalur Register Select. Ketika RS berlogika LOW "0", data akan dianggap sebagai sebuah perintah atau instruksi khusus (seperti clear screen, posisi kursor dll). Ketika RS berlogika HIGH "1", data yang dikirim adalah data text yang akan ditampilkan pada display LCD. Sebagai contoh, untuk menampilkan huruf "T" pada layar LCD maka RS harus diset logika HIGH "1".

Jalur RW adalah jalur kontrol Read/ Write. Ketika RW berlogika LOW "0", maka informasi pada bus data akan dituliskan pada layar LCD. Ketika RW berlogika HIGH "1", maka program akan melakukan pembacaan memori dari LCD. Sedangkan pada aplikasi umum pin RW selalu diberi logika LOW "0".



Bus data terdiri dari 4 atau 8 jalur (bergantung pada mode operasi yang dipilih oleh user). Pada mode 8 bit, maka data bus yang digunakan 8 jalur yang diacukan sebagai DB0 s.d DB7 sedangkan pada mode 4 bit, maka data bus yang digunakan adalah 4 jalur yaitu DB4 s.d DB7.

PIN	Nama	Fungsi
1	VSS	Ground voltage
2	VCC	+5V
3	VEE	Contrast voltage
4	RS	Register Select 0 = Instruction Register 1 = Data Register
5	R/W	Read/ Write, to choose write or read mode 0 = write mode 1 = read mode
6	E	Enable 0 = start to lacht data to LCD character 1= disable
7	DB0	Data Bus 0
8	DB1	Data Bus 1
9	DB2	Data Bus 2
10	DB3	Data Bus 3
11	DB4	Data Bus 4
12	DB5	Data Bus 5
13	DB6	Data Bus 6
14	DB7	Data Bus 7
15	BPL	Back Plane Light / Anode
16	GND	Ground voltage / Katode

Gambar 4.2 Konfigurasi Pin Pada LCD

Beberapa perintah dasar yang harus dipahami adalah inisialisasi LCD Character:

- **Function Set**

Mengatur interface lebar data, jumlah dari baris dan ukuran font karakter

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	1	DL	N	F	X	X

X : Don't care

DL: Mengatur lebar data

DL=1, Lebar data interface 8 bit (DB7 s/d DB0)

DL=0, Lebar data interface 4 bit (DB7 s/d DB4)

Ketika menggunakan lebar data 4 bit, data harus dikirimkan dua kali

N: Pengaktifan baris

N=0, 1 baris

N=1, 2 baris

F: Penentuan ukuran font karakter



F=0, 5x7

F=1, 5x8

○ **Entry Mode Set**

Mengatur increment/ decrement dan mode geser

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	1	I/D	S

I/D : Increment/ decrement dari alamat DDRAM dengan 1 ketika kode karakter dituliskan ke DDRAM.

I/D = "0", decrement

I/D = "1", increment

S: Geser keseluruhan display kekanan dan kekiri

S=1, geser kekiri atau kekanan bergantung pada I/D

S=0, display tidak bergeser

○ **Display On / Off Cursor**

Mengatur status display ON atau OFF, cursor ON/ OFF dan fungsi Cursor Blink

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	1	D	C	B

D : Display, Mengatur display

D = 1, Display is ON

D = 0, Display is OFF

Pada kasus ini data display masih tetap berada di DDRAM, dan dapat ditampilkan kembali secara langsung dengan mengatur D=1.

C : Cursor, Menampilkan kursor

C = 1, kursor ditampilkan

C = 0, kursor tidak ditampilkan

B : Blink, Karakter ditunjukkan dengan kursor yang berkedip

B=1, kursor blink

○ **Clear Display**

Perintah ini untuk menghapus layar

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	0	0	1

○ **Geser Kursor dan Display**



Geser posisi kursor atau display ke kanan atau kekiri tanpa menulis atau baca data display. Fungsi ini digunakan untuk koreksi atau pencarian display.

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	1	S/C	R/L	X	X

x = don't care

S/C	R/L	Keterangan
0	0	Shift cursor position to the left
0	1	Shift cursor position to the right
1	0	Shift the entire display to the left
1	1	Shift the entire display to the right

o Posisi Kursor

Modul LCD terdiri dari sejumlah memory yang digunakan untuk display. Semua teks yang kita tuliskan ke modul LCD adalah disimpan didalam memory ini, dan modul LCD secara berturut-turut membaca memory ini untuk menampilkan teks ke modul LCD itu sendiri.

Display	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16						
Line 1	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	...
Line 2	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	51	52	53	54	55	...

Pada peta memori tersebut, 00 s.d. 0F dan 40 s.d. 4F adalah display yang tampak. Sebagaimana yang anda lihat, jumlahnya sebanyak 16 karakter per baris dengan dua baris. Angka pada setiap kotak adalah alamat memori yang bersesuaian dengan posisi dari layar. Demikianlah karakter pertama di sudut kiri atas adalah menempati alamat 00h. Posisi karakter berikutnya adalah alamat 01h dan seterusnya. Akan tetapi, karakter pertama dari baris 2 sebagaimana yang ditunjukkan pada peta memori adalah pada alamat 40h. Demikianlah kita perlu untuk mengirim sebuah perintah ke LCD untuk mengatur letak posisi kursor pada baris dan kolom tertentu.

Instruksi Set Posisi Kursor adalah 80h. Untuk ini kita perlu menambahkan alamat lokasi dimana kita berharap untuk menempatkan kursor. Sebagai contoh, kita ingin menampilkan kata "World" pada baris ke dua pada posisi kolom ke sepuluh. Sesuai peta memori, posisi karakter pada kolom 11 dari baris ke dua, mempunyai alamat 4Ah, sehingga sebelum kita tulis kata "World" pada LCD, kita harus mengirim instruksi set posisi kursor, dan perintah untuk instruksi ini adalah 80h ditambah dengan alamat $80h + 4Ah = 0Cah$. Sehingga dengan mengirim perintah Cah ke LCD, akan menempatkan kursor pada baris kedua dan kolom ke 11 dari DDRAM.

Pemrograman LCD pada CodeVisionAVR

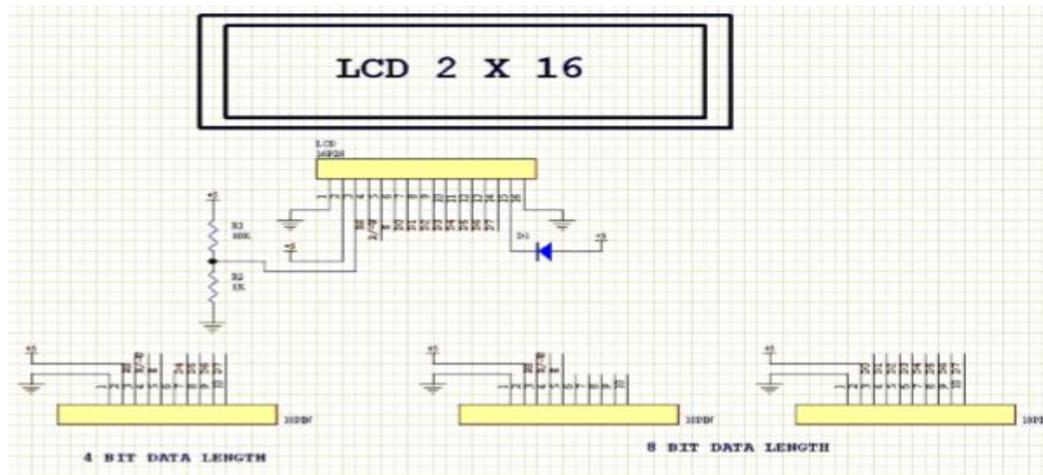
Pada CodeVisionAVR sudah tersedia fasilitas pemrograman LCD 2x16 untuk mode 4 bit, sehingga koneksi interface dan instruksi pemrograman tinggal disesuaikan saja. Namun, untuk mode 8 bit kita harus melakukan inisialisasi sendiri dan membangun fungsi sendiri pula

Beberapa fungsi LCD yang ada di CodeVisionAVR untuk mode 4 bit :



1. `lcd_init();` // inialisasi lcd
2. `lcd_putchar(char c);` // menulis karakter c pada LCD
3. `lcd_puts(char *str);` // menulis string yang disimpan pada SRAM pada LCD
4. `lcd_putsf(" ");` // menulis string ke lcd
5. `lcd_gotoxy(unsigned char x, unsigned char y);` // posisi baris dan kolom
6. `lcd_clear();` // menghapus layer pada LCD

D. Rangkaian Percobaan



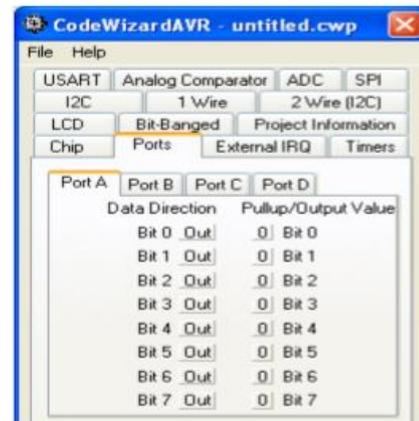
Gambar 4.3 Output LCD

Prosedur Percobaan Mode 4 bit

- Buka program CodeVisionAVR dengan menekan klik ganda icon  pada dekstop komputer. Hubungkan setiap titik menggunakan kabel jumper.
- Pilih File – New – Project - Yes.
- Selanjutnya pilih AVR8 (ATtiny,Atmega,AT90) – Ok, jika menggunakan IC ATMega.
- Kemudian Pilih chip ATMega8535 dan clock 8MHz, dengan mengikuti isi sesuai gambar 4.4.
- Lalu setting port-A sebagai output dengan cara memilih Ports, lalu pada Port A ubah In menjadi Out mulai dari Bit 0 sampai dengan Bit 7.

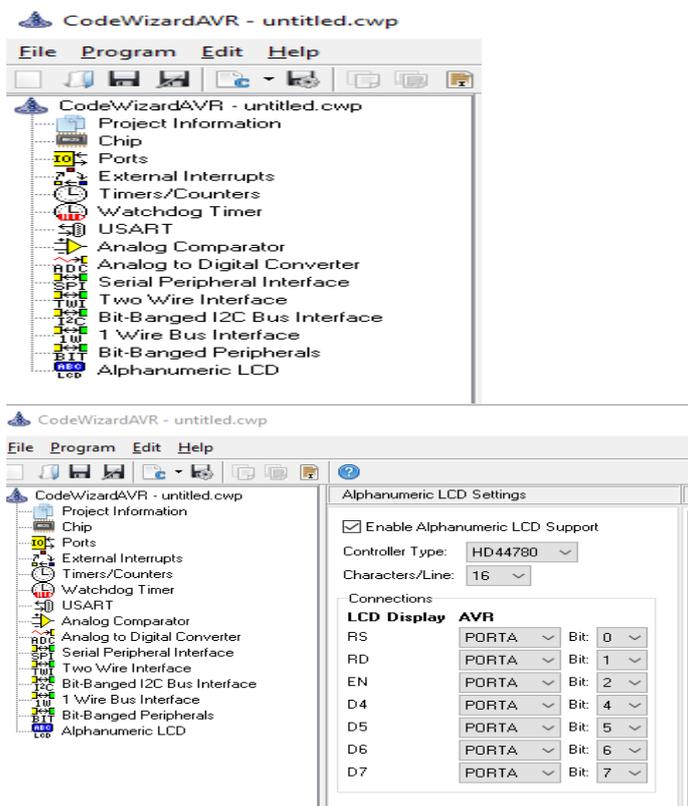


Gambar 4.4 Wizard Chip



Gambar 4.5 Wizard Port

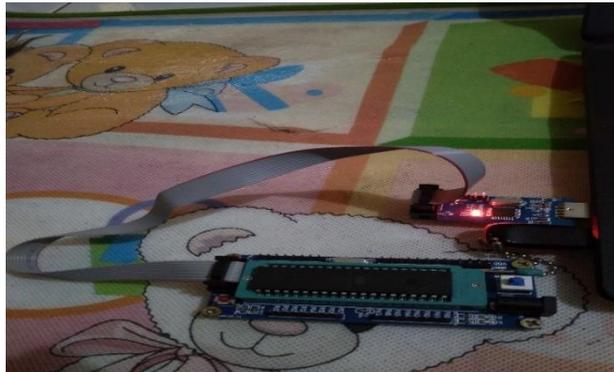
- o Kemudian dilanjutkan settingan pada LCD dengan memilih AlphanumericLCD - centang pada Enable Alphanumeric LCD Support.
- o Atur Controller Type HD44780, Characters/Line 16. Pastikan bahwa settingan pada LCD benar berdasarkan gambar 4.6.



Gambar 4.6 Setting Port LCD

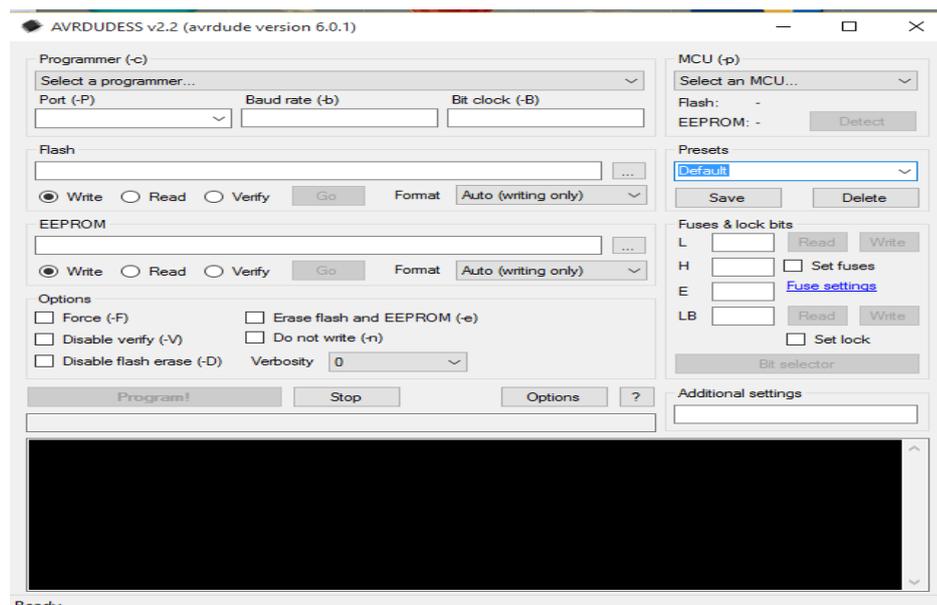
- o Jika selesai, pilih File – Generate, Save and Exit.
- o Arahkan direktori ke dalam folder yang anda buat sebelumnya, lalu ketik nama file dengan Percobaan4. Tekan Save. Lakukan sampai tiga kali dengan nama folder yang sama.
- o Ketikkan program yang diinginkan atau sesuai contoh pada daftar program percobaan4.

- Jika sudah selesai menetik program, Lakukan compile dengan cara menekan gambar  atau tombol F9. Jika tidak ada error tekan OK. Bila ada error, maka periksa program yang anda ketik. Lalu ulangi lagi.
- Langkah selanjutnya yaitu mendownload program yang sudah dibuat ke IC Atmega 8535
- Hubungkan kabel USBASP downloader dari terminal USB komputer ke downloader pada trainer praktikum. Hubungkan PORTA ke modul LED dengan menggunakan kabel jumper. Hubungkan kabel jumper dari PORTA0 ke pin 0, PORTA1 ke pin 1, PORTA2 ke pin 2 secara berurutan sampai PORTA7 ke pin 7. Hubungkan + ke VCC dan – ke Ground.



Gambar 4.7 Downloader Mikrokontroler

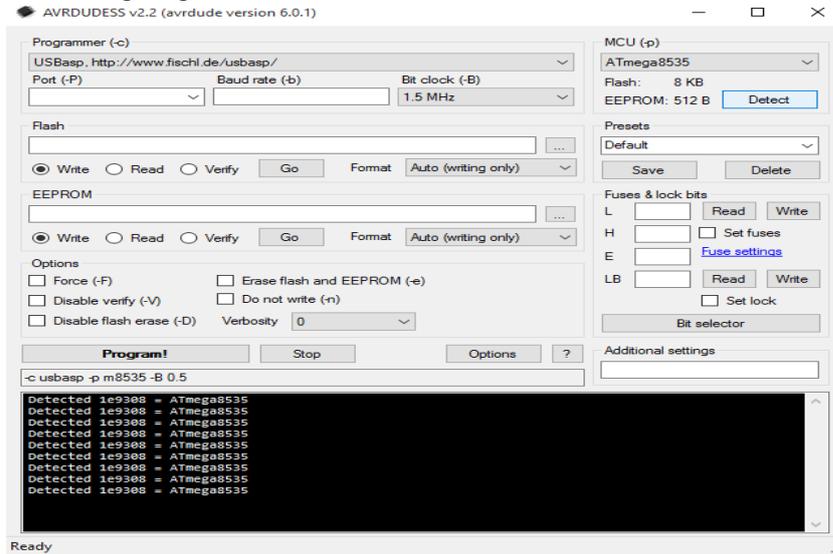
- Untuk mengisi program yang telah dibuat ke dalam chip, maka terlebih dahulu buka software AVRDUDES dengan menekan klik ganda icon  pada desktop Komputer.
- Selanjutnya akan muncul seperti gambar 4.8 Tampilan awal software AVRDUDES.



Gambar 4.8 Tampilan Awal Software AVRDUDES

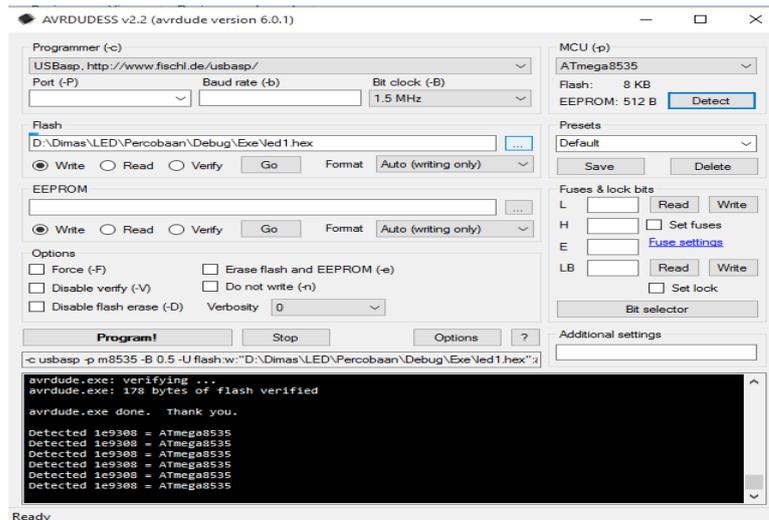


- Kemudian pilih IC yang digunakan di menu Select an MCU. Karena menggunakan IC ATmega8535 jadi pilih IC ATmega8535. Setelah itu pilih menu Select a Programmer, pilih USBasp. Sebagai port downloader yang digunakan.
- Setelah selesai, klik Detect untuk melihat IC ATmega8535 terhubung atau tidak ke komputer. Jika terhubung maka di kolom bawah warna hitam akan ada tulisan “Detected 1e9308 = ATmega8535”. Jika tidak terhubung periksa kembali kabel usb. Lalu ulangi lagi.



Gambar 4.9 konfigurasi IC ATmega

- Jika sudah terhubung langkah selanjutnya yaitu mengupload program yang sudah dibuat kedalam IC ATmega8535, dengan memilih flash – pilih tanda  cari file program yang sudah disimpan. Kemudian di dalam file tersebut pilih debug – execution. Jika sudah baru klik “Program” di software AVRDUDES, tunggu sampai proses selesai uploading selesai.



Gambar 4.10 Proses Uploading Program

- Hubungkan konektor port-A dengan konektor LCD. Amati yang terjadi dan dianalisa!



- o Lakukan untuk semua contoh program percobaan. Lalu selesaikan tugas yang diberikan dengan menunjukkan hasilnya kepada dosen yang mengampuh!

Daftar program percobaan mode 4 bit:

1. Menampilkan karakter "A" pada baris ke-0 dan kolom ke-0

```
#include <mega8535.h>
// Alphanumeric LCD Module functions
#asm
.equ __lcd_port=0x15 ;PORTC
#endasm
#include <lcd.h>

void main(void)
{
    PORTA=0x00;
    DDRA=0x00;
    PORTB=0x00;
    DDRB=0x00;
    PORTC=0x00;
    DDRC=0xFF;
    PORTD=0x00;
    DDRD=0x00;

    while(1)
    {
        lcd_gotoxy(0,0);
        lcd_putchar('A');
    }
}
```

2. Menulis karakter di baris 1 dan baris 2

```
#include <mega8535.h>
#asm
.equ __lcd_port=0x15 ;PORTA
#endasm
#include <lcd.h>

void main (void)
{
    lcd_gotoxy(0,0);
    lcd_putsf("    BARIS 1    ");
    lcd_gotoxy(0,1);
    lcd_putsf("    BARIS 2    ");
    while(1);
}
```



3. Menampilkan angka "0" dengan nilai ASCII

```
#include <mega8535.h>
#include <alcd.h>
void main(void)
{
    DDRA=0xFF;
    PORTA=0x00;

    lcd_init(16);
    while (1)
    {
        lcd_gotoxy(0,0);
        lcd_putchar(48);
    }
}
```

4. Menampilkan tulisan dan angka dengan library standard I/O bahasa C

```
#include <mega8535.h>
#include <stdio.h>

#asm
.equ __lcd_port=0x15 ;PORTA
#endasm
#include <alcd.h>
unsigned char n,kata[16];

void main(void)
{
    DDRA=0xFF;
    PORTA=0x00;

    lcd_init(16);
    n=6;
    while (1)
    {
        sprintf(kata,"Percobaan %d",n);
        lcd_gotoxy(0,0);
        lcd_puts(kata);
    }
}
```

Tugas !

1. Buatlah program konter up (0 s.d 10) secara kontinyu pada baris 1 kolom 1

```
#include <mega8535.h>
#include <stdio.h>
```



```
#include <delay.h>
#include <alcd.h>

unsigned char x;
char buffer[16];
void main(void)
{
    DDRA=0xFF;
    PORTA=0x00;

    lcd_init(16);
    while (1)
    {
        for(x=0;x<=10;x++)
        {
            sprintf(buffer,"%d",x);
            lcd_gotoxy(0,0);
            lcd_puts(buffer);
            delay_ms(100);
        }
        lcd_clear();
        delay_ms(100);
        lcd_clear();
    }
}
```

- 2. Buatlah program led menyala geser kiri-kanan sebanyak 10 kali kemudian pindah kanan-kiri sebanyak 10 kali secara kontinyu, dengan display pada LCD tulisan "geser kiri" ketika terjadi geser kanan-kiri dan sebaliknya "geser kanan" serta tampilkan juga nilai konternya.**

```
#include <mega8535.h>
#include <stdio.h>
#include <alcd.h>
#include <delay.h>
#include <math.h>

void main(void)
{
    int c=0,a;
    unsigned char b;
    char buffer[16];

    DDRA =0xFF;
    PORTA=0x00;
    DDRC =0xFF;
    PORTC=0x00;

    lcd_init(16);
    while (1)
    {
        c=0;
        lcd_gotoxy(0,1);
```

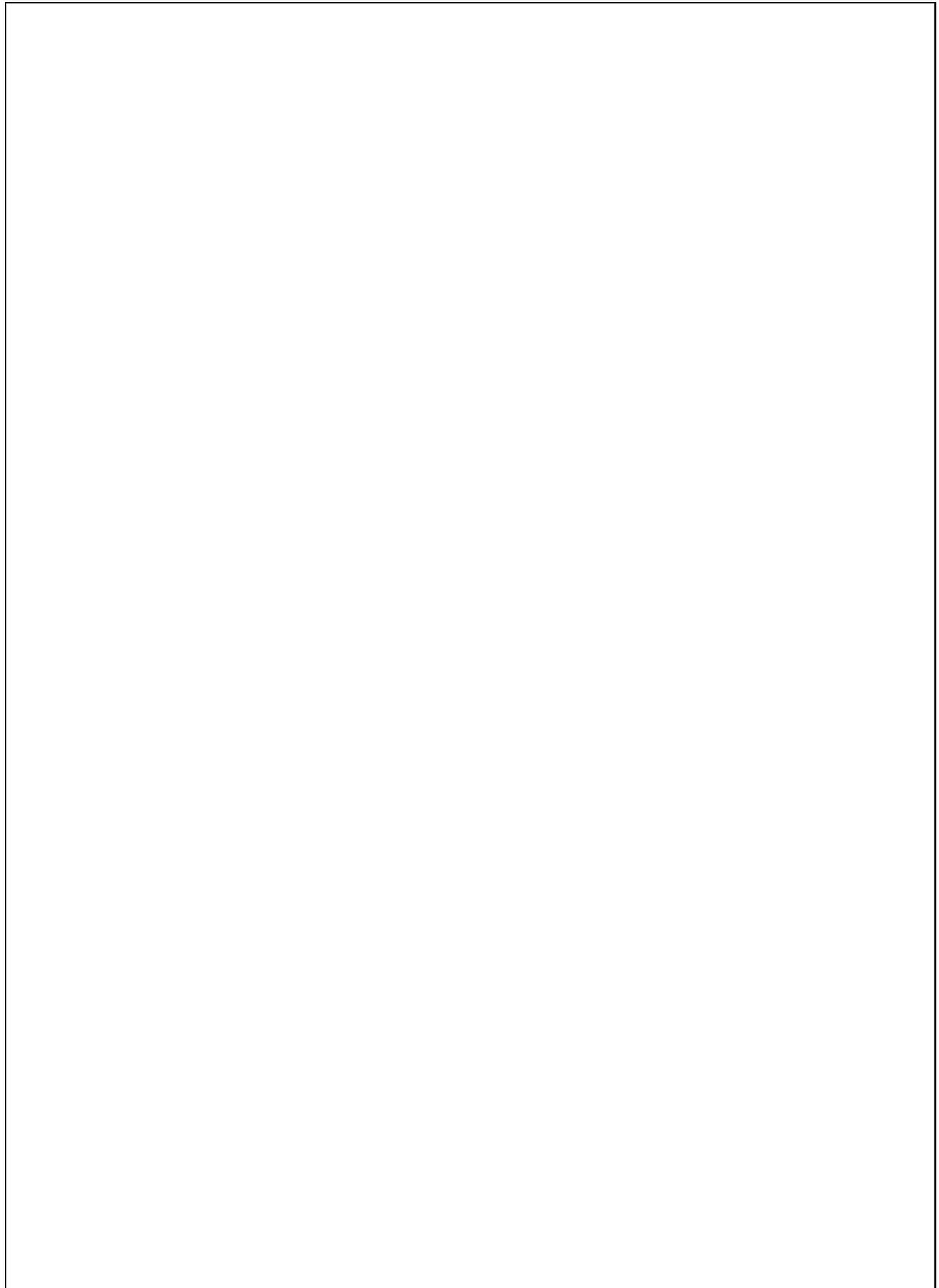


```
    sprintf(buffer, "%d", c);
    lcd_puts(buffer);
    delay_ms(1);
    for(c=1; c<=10; c++)
    {
        for(a=0; a<=7; a++)
        {
            b=pow(2, a);
            PORTC=b;
            lcd_gotoxy(0, 0);
            lcd_puts("Led Geser Kanan");
            delay_ms(50);
        }

        lcd_gotoxy(0, 1);
        sprintf(buffer, "%d", c);
        lcd_puts(buffer);
        delay_ms(1);
    }
    delay_ms(200);
    lcd_clear();

    c=0;
    lcd_gotoxy(0, 1);
    sprintf(buffer, "%d", c);
    lcd_puts(buffer);
    delay_ms(1);
    for(c=1; c<=10; c++)
    {
        for(a=7; a>=0; a--)
        {
            b=pow(2, a);
            PORTC=b;
            lcd_gotoxy(0, 0);
            lcd_puts("Led Geser Kiri");
            delay_ms(50);
        }

        lcd_gotoxy(0, 1);
        sprintf(buffer, "%d", c);
        lcd_puts(buffer);
        delay_ms(1);
    }
    delay_ms(200);
    lcd_clear();
}
```





MODUL PRAKTIKUM
IOT (INTERNET OF THINGS)

PERCOBAAN 1 MENAMPILKAN NILAI SENSOR ULTRASONIK DI ANDROID

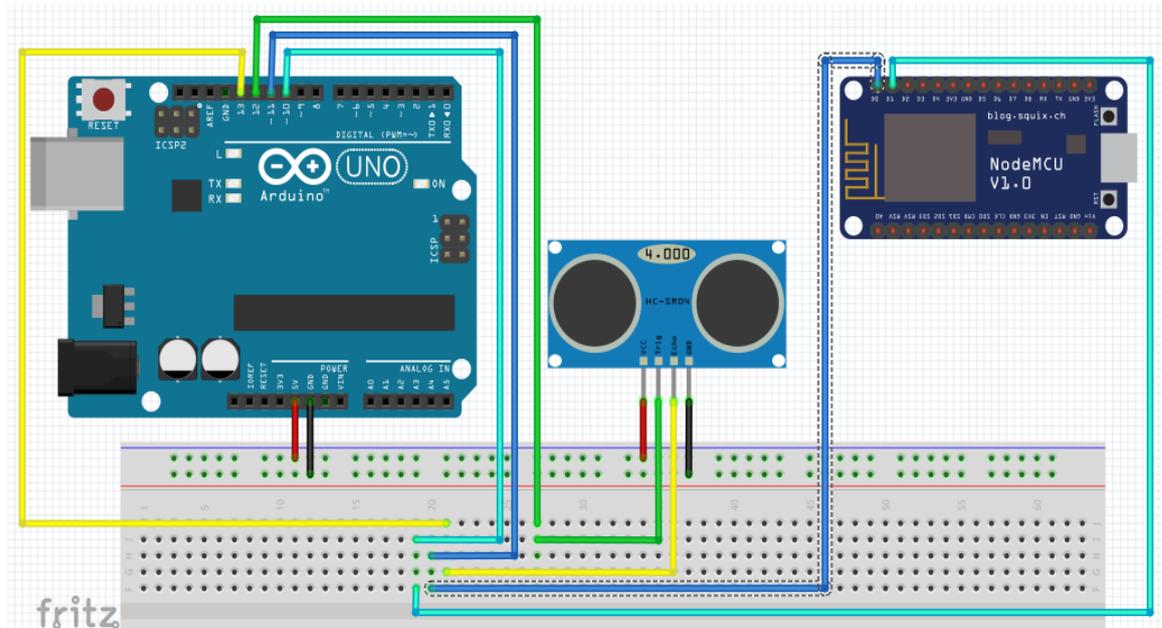
A. Tujuan

1. Mahasiswa dapat memahami program Arduino.
2. Mahasiswa bisa menggunakan software Arduino dengan prosedur yang benar.
3. Mahasiswa bisa membaca nilai sensor dari android.
4. Mahasiswa dapat memahami prinsip kerja sensor ultrasonik.

B. Komponen Dan Peralatan Yang Digunakan

- a. Modul Iot (*Internet of Things*)
- b. Komputer/Laptop

C. Gambar Rangkaian Percobaan



Gambar 1.1 Rangkaian Sensor Ultrasonik

D. Prosedur Percobaan

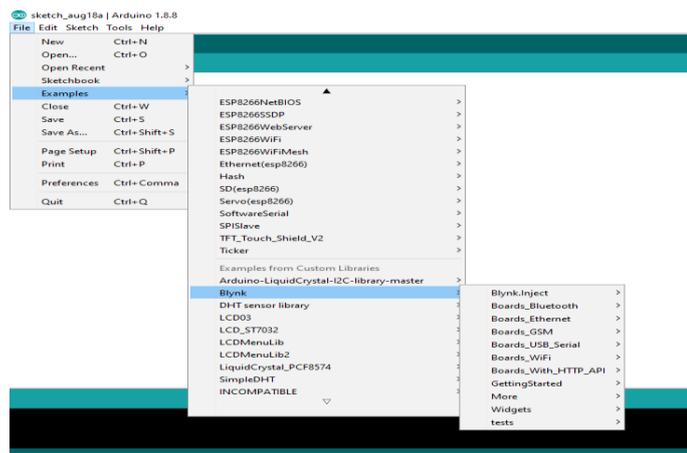
1. Buka program Arduino uno dengan menekan klik ganda icon  pada dekstop komputer.
2. Setelah itu akan muncul tampilan seperti gambar 1.2.



```
File Edit Sketch Tools Help  
sketch_aug18a  
void setup() {  
  // put your setup code here, to run once:  
  
}  
  
void loop() {  
  // put your main code here, to run repeatedly:  
  
}
```

Gambar 1.2 Tampilan Awal Software Arduino

3. Untuk melihat contoh program klik File - klik Example, lalu pilih contoh program yang dibutuhkan.



Gambar 1.3 Contoh Program di Arduino

4. Setelah memilih program yang diinginkan maka klik program tersebut, maka akan muncul *coding* seperti gambar 1.4. Selanjutnya program lah sesuai keinginan.



```
ESP8266_Standalone | Arduino 1.8.8
File Edit Sketch Tools Help
ESP8266_Standalone
*****
/* Comment this out to disable prints and save space */
#define BLYNK_PRINT Serial

#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>

// You should get Auth Token in the Blynk App.
// Go to the Project Settings (nut icon).
char auth[] = "yourAuthToken";

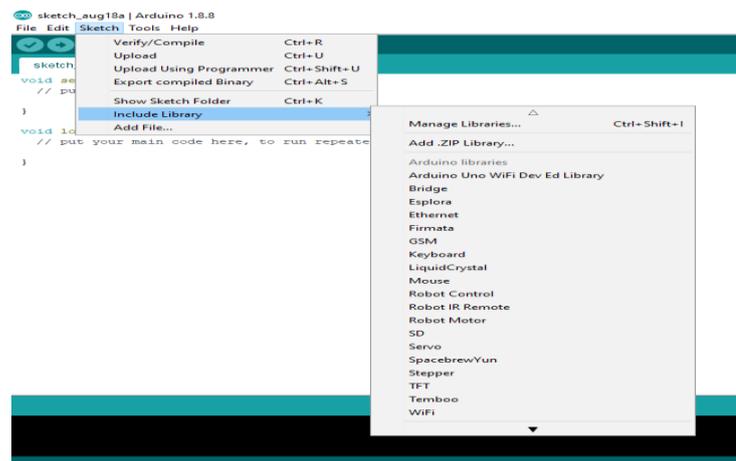
// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "YourNetworkName";
char pass[] = "YourPassword";

void setup()
{
  // Debug console
  Serial.begin(9600);
  Blynk.begin(auth, ssid, pass);
}

void loop()
{
  Blynk.run();
}
```

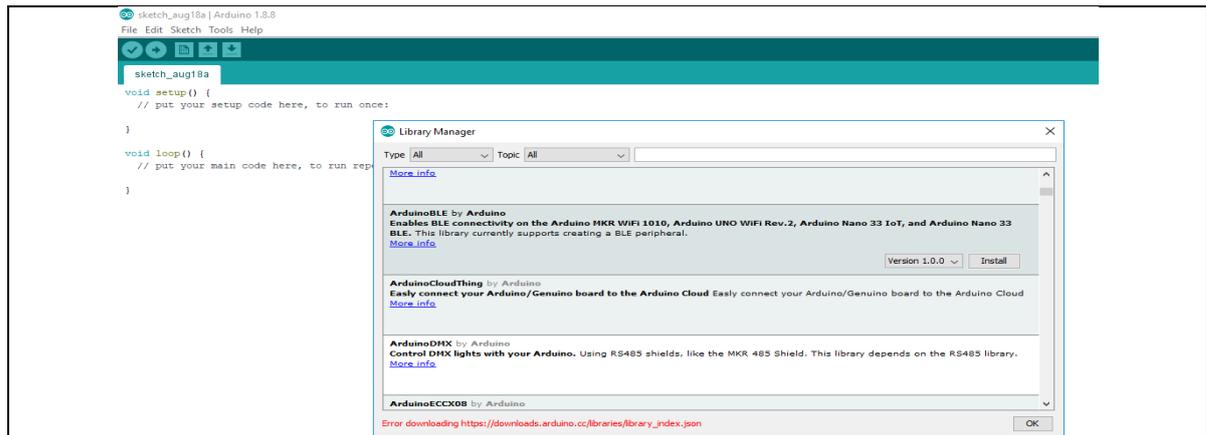
Gambar 1.4 Contoh Program Arduino Yang Sudah Di ketik

5. Jika ingin menambahkan Library maka download terlebih dahulu dengan cara klik Skecth – Include library – Manage libraries atau bisa menekan Ctrl+Shift+I



Gambar 1.5 Cara Menambahkan Library Pada Software Arduino

6. Kemudian akan muncul tampilan seperti gambar 1.6, lalu pilih library yang dibutuhkan setelah itu klik instal. Tunggu hingga selesai mendownload, setelah selesai maka secara otomatis akan tersimpan di software arduino.



Gambar 1.6 Library Arduino Yang Sudah Ditambahkan

7. Jika Library yang diinginkan telah di download maka langkah selanjutnya memprogram sesuai yang petunjuk yang berlaku.

Contoh program:

```
#include <SoftwareSerial.h> //software komunikasi serial
#include <LiquidCrystal_I2C.h>
#include <Wire.h>

LiquidCrystal_I2C lcd (0x27, 16,2);
SoftwareSerial mySerial(5, 6); // RX, TX //pin yg dipakai
komunikasi serial
int trigPin = 12; //Trig - blue Jumper
int echoPin = 13; //Echo - yellow Jumper
int duration, cm;
int batas;
void setup() {
  //lcd.init();
  Serial.begin(9600);
  while (!Serial) {
    ; // wait for serial port to connect. Needed for native
    USB port only
  }
  lcd.init();

  lcd.backlight();
  Serial.println("Goodnight moon!");
  // set the data rate for the SoftwareSerial port
  mySerial.begin(9600);
  mySerial.println("Hello, world?");
  //Define inputs and outputs
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
```



```
}
void ultrasonik()
{
// Sensor dipicu oleh pulsa HIGH dari 10us atau lebih.
// Berikan pulsa LOW pendek terlebih dahulu untuk memastikan
pulsa HIGH bersih:
digitalWrite(trigPin, LOW);
delayMicroseconds(5);
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
// Baca sinyal dari sensor: pulsa HIGH yang
// duration adalah waktu (dalam mikrodetik) dari pengirim
// Dari ping untuk penerimaan Echo off dari sebuah objek.
pinMode(echoPin, INPUT);
duration = pulseIn(echoPin, HIGH);
// convert jarak kedalam cm dan meter
batas = (duration/2) / 29.1;
//meter = (duration/2) / 100;
//Serial.print(meter);
//Serial.print("m, ");
if(batas>=31){
    cm=0;
}else{
    cm=batas;
}
Serial.print(cm);
Serial.print("cm");
Serial.println();
delay(250);
}
void loop()
{
// dht();
ultrasonik();
mySerial.print (cm); //menampilkan nilai jarak yg akan
dikirim
mySerial.print ('j'); //kode sensor ultrasonik
lcd.setCursor(0,0);
lcd.print("JARAK : ");
lcd.print(cm);
lcd.print(" cm ");
delay(200);
}
```

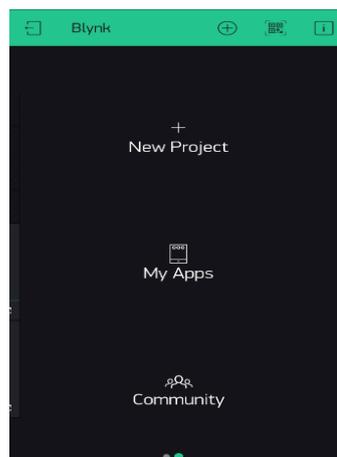
8. Jika sudah selesai mengetik program maka dilanjutkan dengan melakukan verify dengan memilih tombol centang (✓) kolom kiri atas software arduino seperti gambar 2.7. Verify ini fungsinya untuk mengecek program yang sudah diketik eror ataupun tidak eror. Kemudian tunggu sampai loading selesai.



```
arduino_ultrasonik
File Edit Sketch Tools Help
arduino_ultrasonik
digitalWrite(trigPin, LOW);
// Baca sinyal dari sensor pulsa HIGH yang
// duration adalah waktu (dalam mikrodetik) dari pingtim
// Dari ping untuk penerimaan Echo off dari sebuah objek.
pinMode(echoPin, INPUT);
duration = pulseIn(echoPin, HIGH);
// convert jarak kedalam cm dan meter
batas = (duration/2) / 29.1;
//meter = (duration/2) / 100;
//Serial.print(meter);
//Serial.print("m, ");
if(batas>31){
  cm=;
  j=1;
  cm=batas;
}
Serial.print(cm);
Serial.print("cm");
Serial.println();
delay(250);
}
void loop()
{
  // dbci();
  ultrasonik();
  mySerial.print (cm); //menampilkan nilai jarak yg akan dikirim
  mySerial.print (" "); //kode sensor ultrasonik
  led_becUnsur(0,0);
  led_print("DMSM" + " ");
  led_print(cm);
  led_print(" cm ");
  .....
}
Computing sketch
Arduino@Genuino Uno on COM4
```

Gambar 1.7 Verify program Arduino

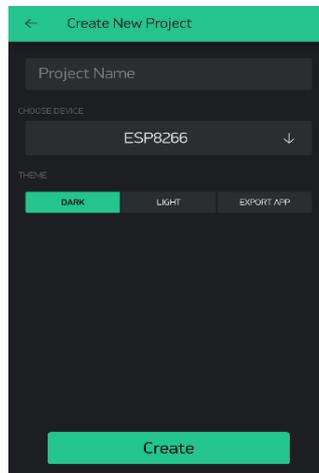
9. Langkah selanjutnya, hubungkan sumber tegangan VCC dan Gnd sensor ultrasonik ke VCC dan GND pada arduino. Kabel merah sebagai VCC dan kabel hitam sebagai Ground. Kemudian pin 13 arduino dihubungkan ke pin Echo pada Sensor Ultrasonik ditandai kabel warna kuning. Pin 12 Arduino dihubungkan ke pin Trig (Trigger) pada Sensor Ultrasonik ditandai dengan kabel warna hijau.
10. Selanjutnya pin 11 Arduino dihubungkan ke pin D0 NodeMCU ESP8266 ditandai dengan kabel biru tua. Pin 10 dihubungkan ke pin D1 NodeMCU ESP8266 ditandai dengan kabel warna biru muda. Jika sudah selesai dilanjutkan dengan melakukan pengaturan tampilan di aplikasi BLYNK.
11. Download terlebih dahulu aplikasi Blynk di PlayStore, lalu instal aplikasi tersebut. Setelah menginstal buka aplikasi Blynk kemudian akan muncul tampilan seperti gambar 1.8 klik New Project.



Gambar 1.8 Tampilan Awal Aplikasi BLYNK



12. Kemudian akan tampil seperti gambar 1.9, selanjutnya isi Project Name contoh Grafik Jarak, klik Create.

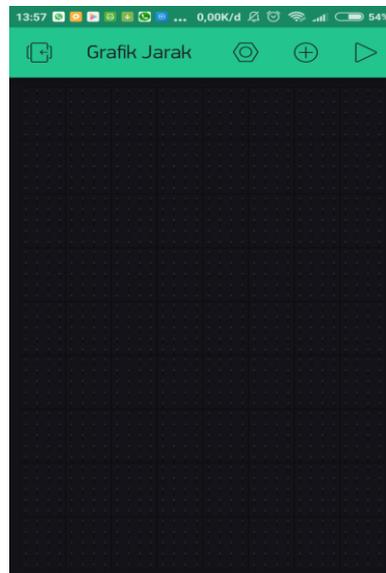


Gambar 1.9 Pengisian Nama Project

13. Kemudian BLYNK akan mengirim token ke email pengguna.

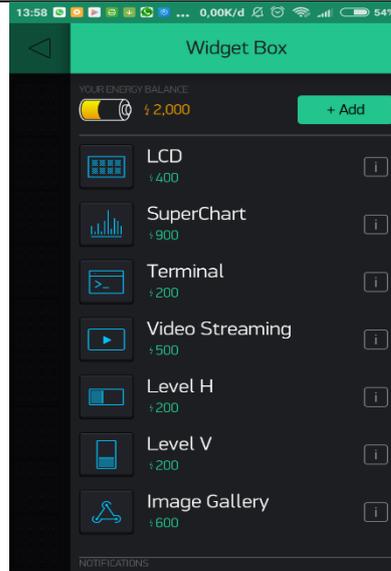
Catatan: jika token tidak terkirim maka periksalah email yang digunakan untuk login ke aplikasi Blynk dan pastikan bahwa sinyal dalam kondisi bagus.

14. Maka akan muncul tampilan project seperti gambar 1.10, untuk menambah library klik simbol 



Gambar 1.10 Memilih Tampilan Library

15. Kemudian pilih SuperChart jika ingin tampilan nilai sensor ultrasonik berbentuk grafik.



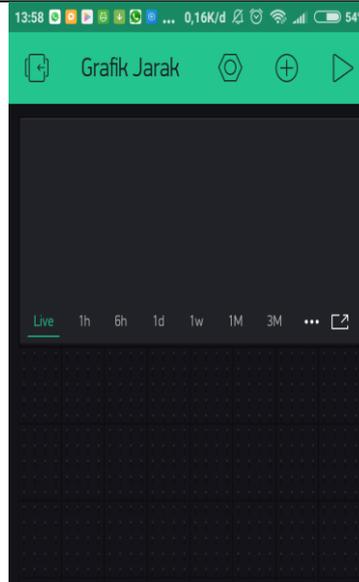
Gambar 1.11 Library Yang Ada di Aplikasi BLYNK

16. Jika sudah dilanjutkan dengan pengaturan pada SuperChart. Pilih Standart Resolution, kemudian klik Live dan 15m. Live menandakan bahwa saat pembacaan nilai sensor ultrasonik secara langsung, nanti akan tampil hasil dari pembacaan sensor tersebut. Sedangkan untuk 15m yaitu pembacaan nilai sensor setiap 15 menit sekali.



Gambar 1.12 Pengaturan Pada SuperChart

17. Jika sudah klik tombol ← dan tampilan BLYNK akan berubah seperti gambar 1.13 berikut ini.



Gambar 1.13 Tampilan Aplikasi BLYNK

18. Jika sudah selesai pengaturan pada aplikasi BLYNK, dilanjutkan dengan mengetik program di Arduino yang nantinya akan di download di modul Wifi NodeMCU ESP8266. Program ini yang nanti akan berfungsi supaya android bisa berkomunikasi dengan peralatan input/output. Ketiklah program NodeMCU sesuai contoh yang ada. Jika sudah diketik kemudian di download. di arduino
19. Jika pengguna ingin menggunakan jaringan pribadi untuk ESP8266 maka di dalam program NodeMCU untuk char ssid dan char pass menggunakan jaringan hospot dari android itu sendiri, tetapi jika ingin menggunakan jaringan dari hospot yang lain maka char ssid dan char pass juga menyesuaikan pada android itu sendiri.

Contoh Program NodeMCU ESP8266:

```
/* Comment this out to disable prints and save space */  
  
#define BLYNK_PRINT Serial  
  
#include <ESP8266WiFi.h>  
  
#include <BlynkSimpleEsp8266.h>  
  
#include <SoftwareSerial.h>  
  
SoftwareSerial myesp(D1, D2); // RX, TX  
  
// You should get Auth Token in the Blynk App.  
// Go to the Project Settings (nut icon).  
char auth[] = "0f6f5336487045689be94fab9c858ad";  
  
// Your WiFi credentials.  
// Set password to "" for open networks.  
char ssid[] = "one phasa";
```



```
char pass[] = "ilang1234";

float ultrasonik, suhu, ult, sh;

void setup()

{

  // Debug console

  Serial.begin(9600);

  while (!Serial) {

    ; // wait for serial port to connect. Needed for native USB port
    only

  }

  Serial.println("Goodnight moon!");

  // set the data rate for the SoftwareSerial port

  myesp.begin(9600);

  myesp.println("Hello, world?");

  Blynk.begin(auth, ssid, pass);

  // You can also specify server:

  //Blynk.begin(auth, ssid, pass, "blynk-cloud.com", 80);

  //Blynk.begin(auth, ssid, pass, IPAddress(192,168,1,100), 8080);

}

void loop()

{

  Blynk.run();

  while (myesp.available()>0)

  {

    ult = myesp.parseFloat();

    if(myesp.read() == 'j');

    {

      ultrasonik=ult;

      Serial.println(ultrasonik);

    }

    sh = myesp.parseFloat();

  }

}
```



```

    if(myesp.read() == 'g');

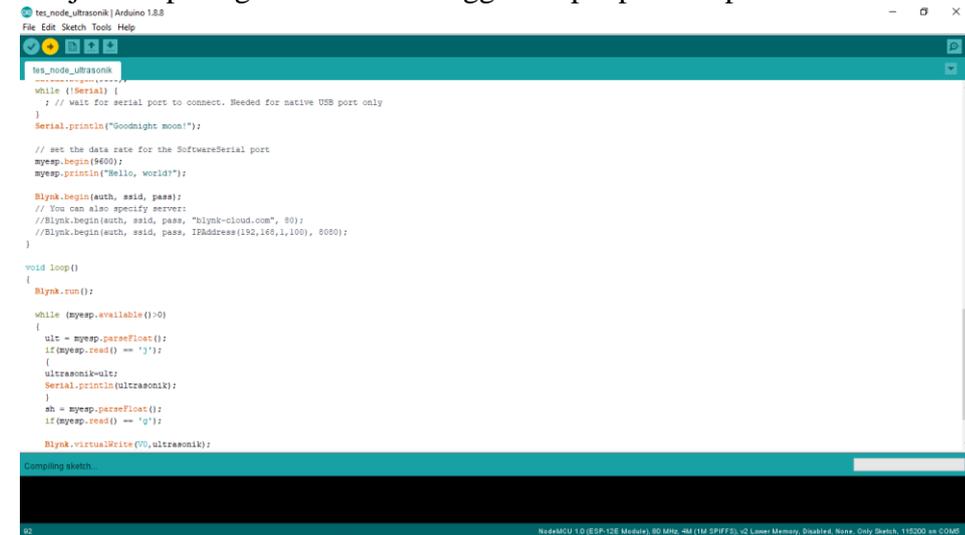
    Blynk.virtualWrite(V0,ultrasonik);

}

}

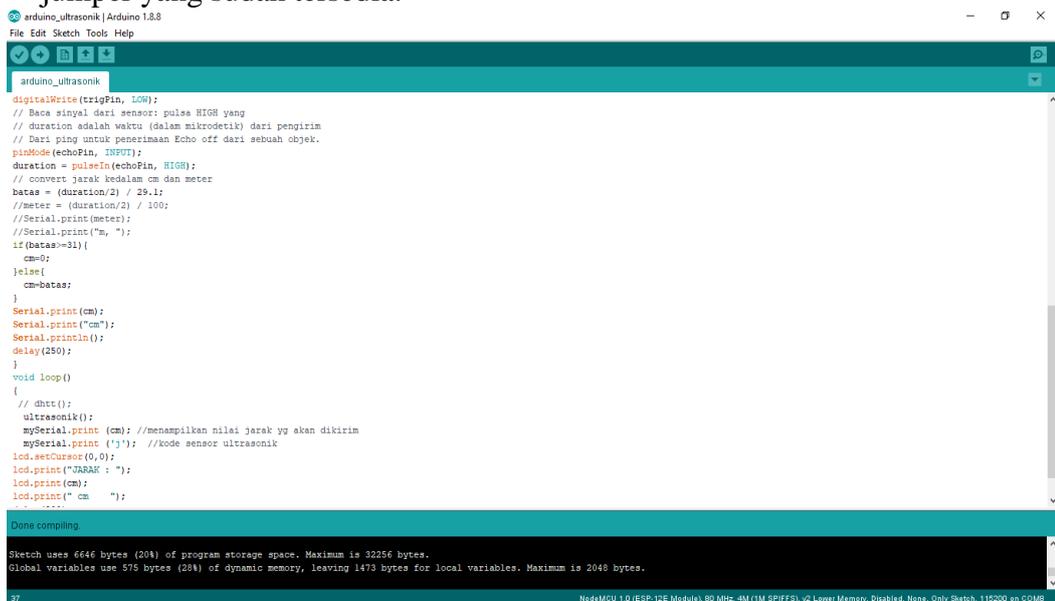
```

20. Jika sudah dilanjutkan dengan mengupload program arduino ke NodeMCU ESP8266 dengan memilih tombol compile di sebelah verify. seperti yang ditunjukkan pada gambar 1.13. Tunggu sampai proses uplod selesai.



Gambar 1.14 Proses Upload Program Arduino.

21. Jika sudah selesai maka hubungkan setiap pin komponen menggunakan kabel jumper yang sudah tersedia.



Gambar 1.15 Proses Uploading Selesai



Tabel hasil pengujian sensor ultrasonik

Pin	Pengukuran Sensor (Senter)	Pengukuran Sensor (Kemiringan)	Pengukuran Penggaris	Persentase Error
12 & 13	5 cm	6 cm	5 cm	1%
12 & 13	10 cm	11 cm	10 cm	1%
12 & 13	14 cm	15 cm	14 cm	1%
12 & 13	19 cm	20 cm	19 cm	1%

Analisa: Dari hasil pengujian diatas dapat disimpulkan bahwa sensor ultrasonik tidak presisi ketika membaca nilai yang non desimal karena pada saat membaca nilai non desimal hasil pada monitor selalu berubah-ubah sehingga untuk pengambilan data menggunakan nilai desimal. Apabila posisi objek miring maka hasil pengukuran pada sensor ultrasonik + 1 maka persentase *error* yaitu 1%.

PERCOBAAN 2 MENAMPILKAN NILAI SENSOR SUHU DI ANDROID

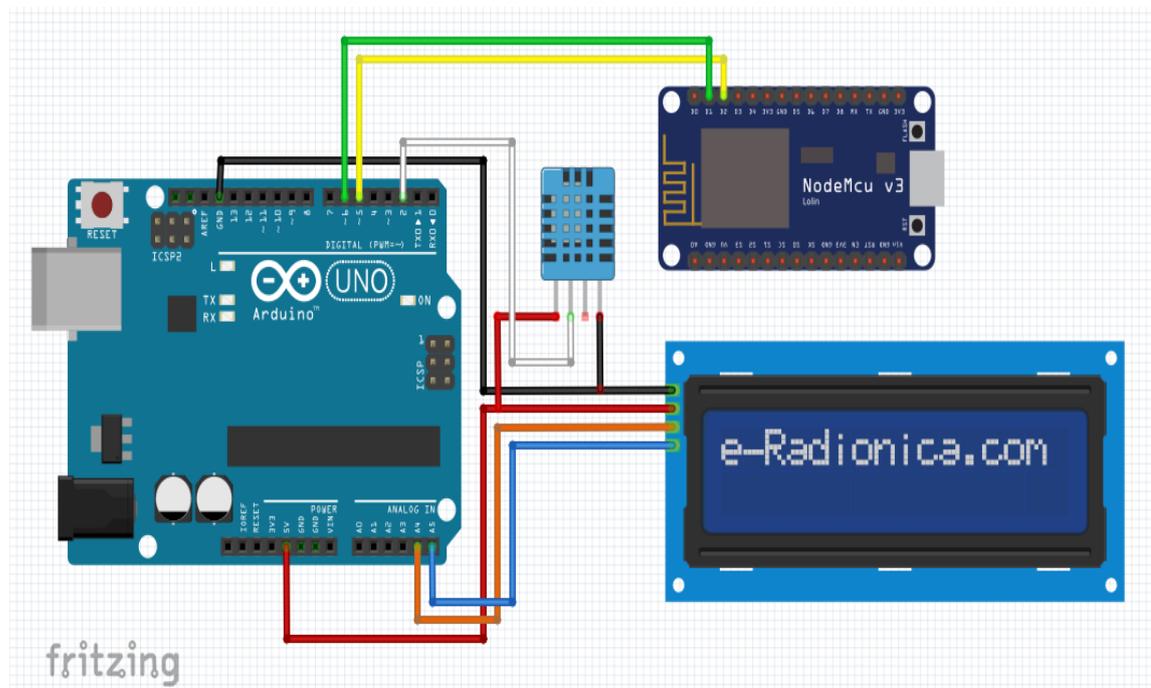
A. Tujuan

1. Mahasiswa dapat memahami program Arduino.
2. Mahasiswa bisa menggunakan software Arduino dengan prosedur yang benar.
3. Mahasiswa bisa membaca nilai sensor dari android.
4. Mahasiswa dapat memahami prinsip kerja dari sensor DHT 11.

B. Komponen Dan Peralatan Yang Digunakan

- a. Modul Iot (*Internet of Things*)
- b. Komputer/Laptop

C. Gambar Rangkaian Percobaan



Gambar 2.1. Rangkaian Sensor Suhu DHT 11

D. Prosedur Percobaan

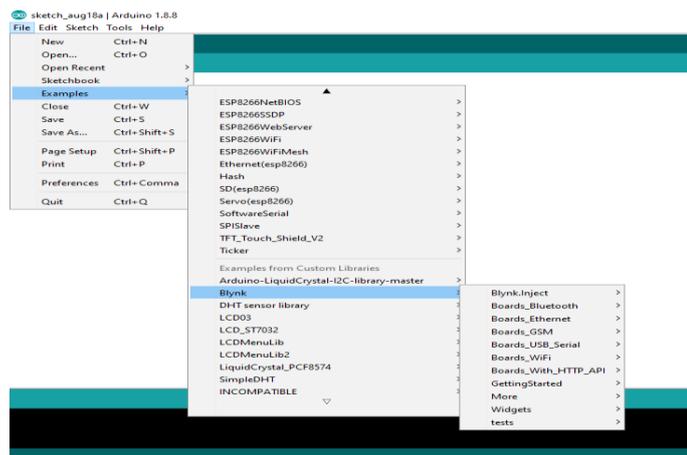
1. Buka program Arduino uno dengan menekan klik ganda icon  pada dekstop komputer.
2. Setelah itu akan muncul tampilan seperti gambar 2.2.



```
File Edit Sketch Tools Help  
sketch_aug18a  
void setup() {  
  // put your setup code here, to run once:  
  
}  
  
void loop() {  
  // put your main code here, to run repeatedly:  
  
}
```

Gambar 2.2 Tampilan Awal Software Arduino

3. Untuk melihat contoh program klik File - klik Example, lalu pilih contoh program yang dibutuhkan.



Gambar 2.3 Contoh Program di Arduino

4. Setelah memilih program yang diinginkan maka klik program tersebut, maka akan muncul *coding* seperti gambar 2.4. Selanjutnya program lah sesuai keinginan.



```
ESP8266_Standalone | Arduino 1.8.8
File Edit Sketch Tools Help
ESP8266_Standalone
*****
/* Comment this out to disable prints and save space */
#define BLYNK_PRINT Serial

#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>

// You should get Auth Token in the Blynk App.
// Go to the Project Settings (nut icon).
char auth[] = "yourAuthToken";

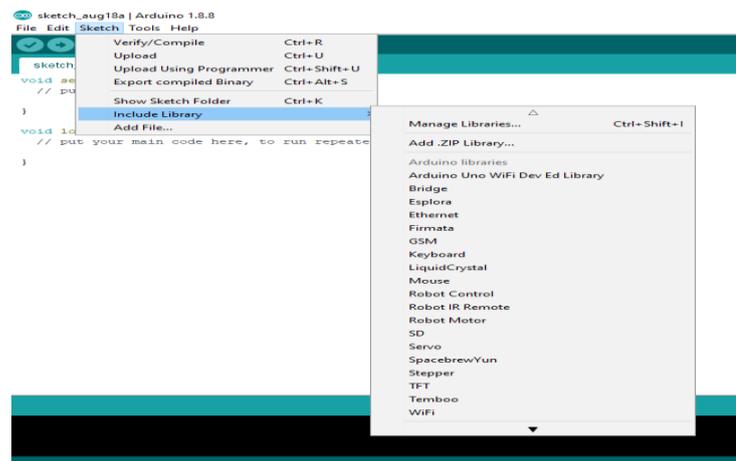
// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "YourNetworkName";
char pass[] = "YourPassword";

void setup()
{
  // Debug console
  Serial.begin(9600);
  Blynk.begin(auth, ssid, pass);
}

void loop()
{
  Blynk.run();
}
```

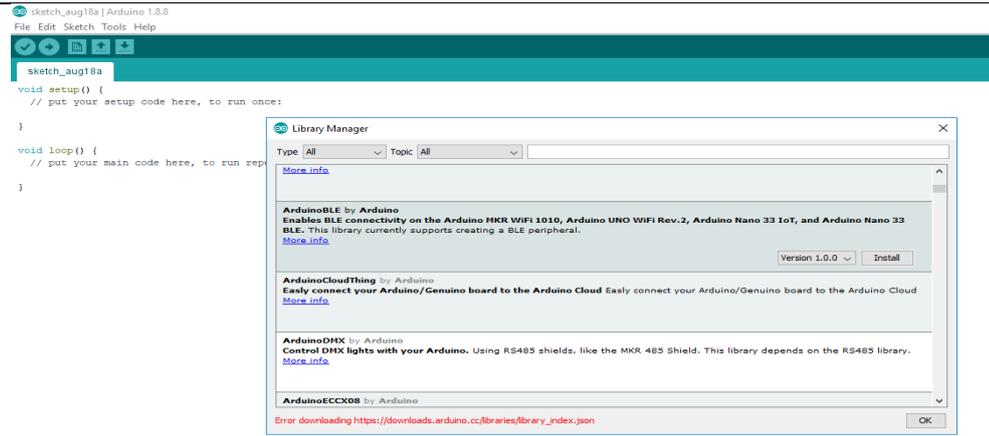
Gambar 2.4 Contoh Program Arduino Yang Sudah Di ketik

5. Jika ingin menambahkan Library maka download terlebih dahulu dengan cara klik Skecth – Include library – Manage libraries atau bisa menekan Ctrl+Shift+I



Gambar 2.5 Cara Menambahkan Library Pada Software Arduino

6. Kemudian akan muncul tampilan seperti gambar 2.6, lalu pilih library yang dibutuhkan setelah itu klik instal. Tunggu hingga selesai mendownload, setelah selesai maka secara otomatis akan tersimpan di software arduino.

Gambar 2.6 Library Arduino Yang Sudah Ditambahkan

7. Jika Library yang diinginkan telah di download maka langkah selanjutnya memprogram sesuai yang petunjuk yang berlaku.

Contoh program:

```
#include <SoftwareSerial.h> //software komunikasi serial
#include <LiquidCrystal_I2C.h>
#include <Wire.h>
#include <SimpleDHT.h>
LiquidCrystal_I2C lcd (0x27, 16,2);

SoftwareSerial mySerial(5, 6); // RX, TX //pin yg dipakai
komunikasi serial

int pinDHT11 = 2;
SimpleDHT11 dht11(pinDHT11);
float cnt = 0;
float err_cnt = 0;
float suhu;
void setup() {
  //lcd.init();
  Serial.begin(9600);
  while (!Serial) {
```



```
    ; // wait for serial port to connect. Needed for native USB
port only

}

lcd.init();

lcd.backlight();

Serial.println("Goodnight moon!");

// set the data rate for the SoftwareSerial port
mySerial.begin(9600);

mySerial.println("Hello, world?");

}

void loop()

{

    dhtt();

    mySerial.print (suhu); //menampilkan nilai jarak yg akan dikirim
mySerial.print ('g'); //kode sensor ultrasonik

}

void dhtt()

{

    // read without samples.

    byte temperature = 0;

    byte humidity = 0;

    float err = SimpleDHTErrSuccess;

    if ((err = dht11.read(&temperature, &humidity, NULL)) !=
SimpleDHTErrSuccess) {

        Serial.print("Read DHT11 failed, err=");
Serial.println(err);delay(1000);

        return;

    }

    Serial.print("Sample OK: ");

    Serial.print((float)temperature); Serial.print(" *C, ");

    Serial.print((float)humidity); Serial.println(" H");

}
```



```
lcd.setCursor(0,0);

lcd.print("SUHU : ");lcd.print((float)temperature); lcd.print("
drajat, ");

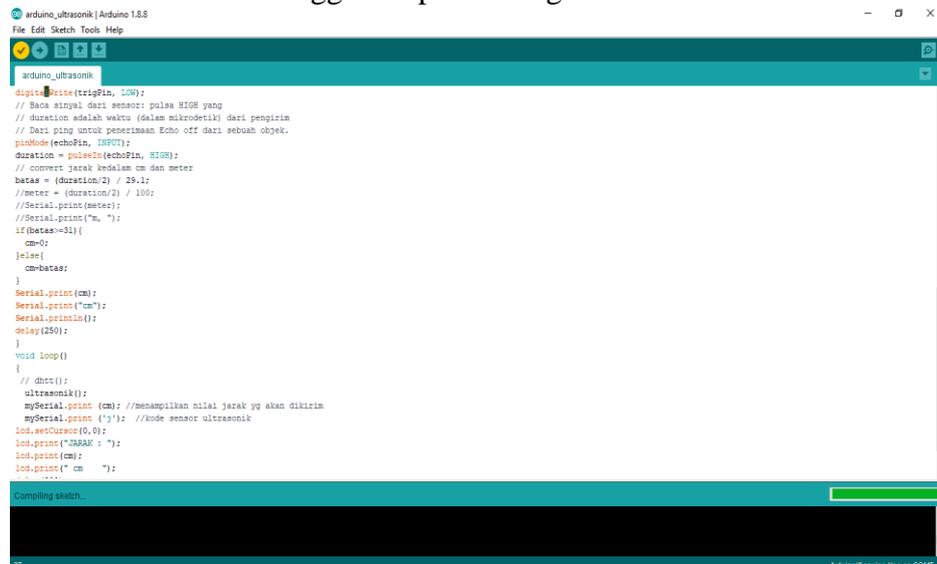
suhu=(float)temperature;

// DHT11 sampling rate is 1HZ.

delay(500);

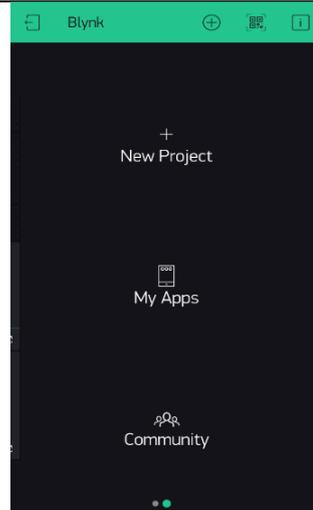
}
```

8. Jika sudah selesai mengetik program maka dilanjutkan dengan melakukan verify dengan memilih tombol centang (✓) kolom kiri atas software arduino seperti gambar 2.7. Verify ini fungsinya untuk mengecek program yang sudah diketik eror ataupun tidak eror. Kemudian tunggu sampai loading selesai.



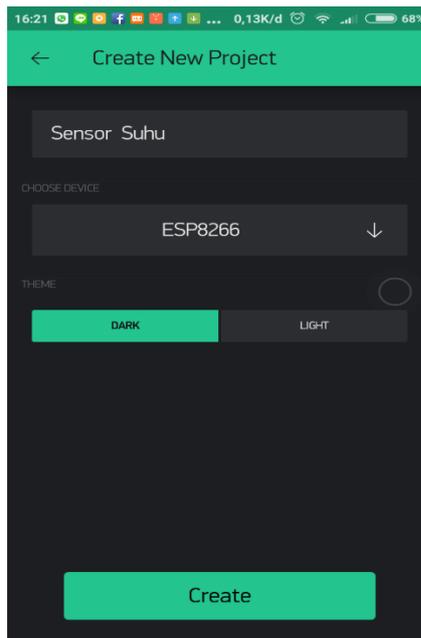
Gambar 2.7 Verify program Arduino

9. Download terlebih dahulu aplikasi Blynk di PlayStore, lalu instal aplikasi tersebut. Setelah menginstal buka aplikasi Blynk kemudian akan muncul tampilan seperti gambar 2.8 klik New Project.



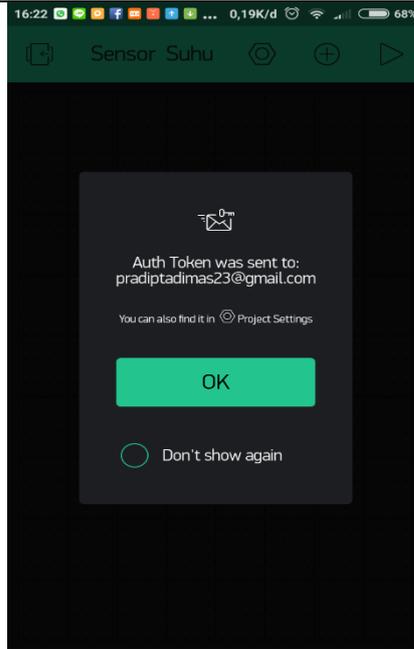
Gambar 2.8 Tampilan Awal Aplikasi BLYNK

10. Kemudian akan tampil seperti gambar 2.9, selanjutnya isi Project Name contoh Sensor Suhu – pilih ESP8266, klik Create.



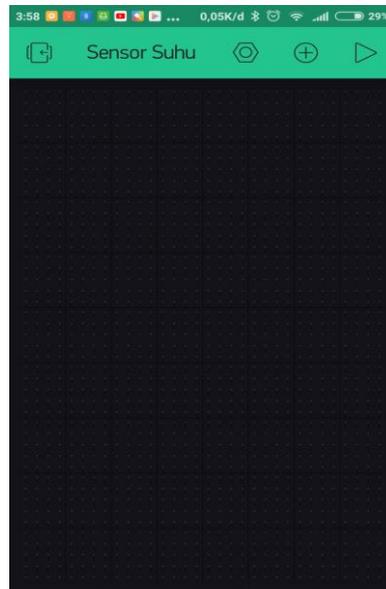
Gambar 2.9 Pengisian Nama Project

11. Kemudian BLYNK akan mengirim token ke email pengguna seperti gambar 2.10
Catatan: jika token tidak terkirim maka periksalah email yang digunakan untuk login ke aplikasi Blynk dan pastikan bahwa sinyal dalam kondisi bagus.



Gambar 2.10 Token Yang dikirim BLYNK

12. Maka akan muncul tampilan project seperti gambar 2.11, untuk menambah library klik symbol 

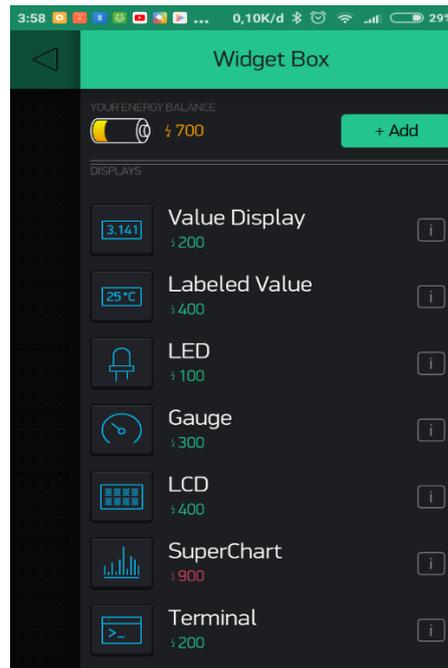


Gambar 2.11

Tampilan Awal Aplikasi BLYNK

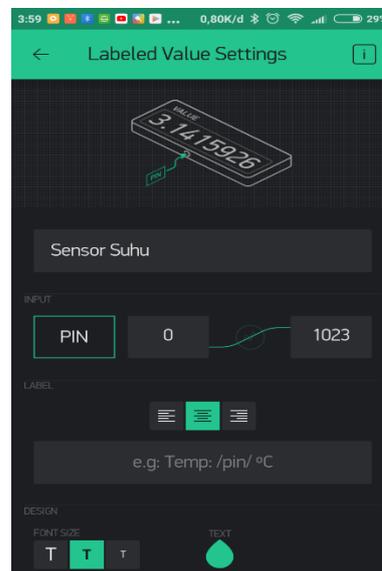


13. Maka akan tampil seperti gambar 2.12, pilih Labeled Value



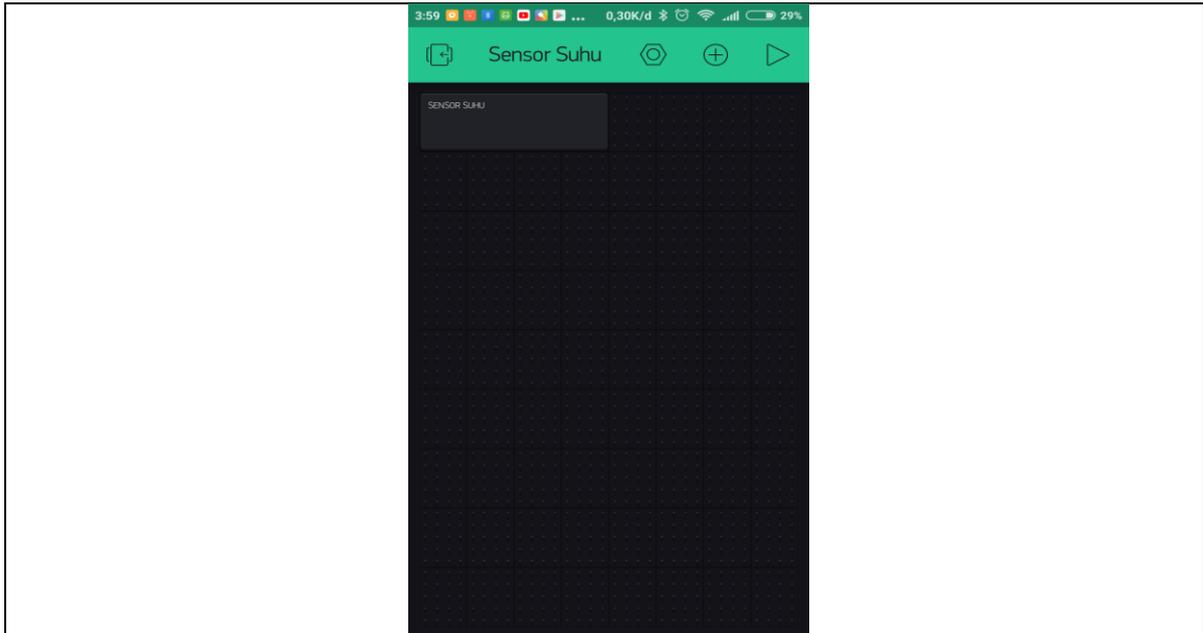
Gambar 2.11 Library Yang Ada di Aplikasi BLYNK

14. Jika sudah dilanjutkan dengan pengaturan pada Labeled Value.



Gambar 2.12 Pengaturan Pada Labeled Value

15. Jika sudah klik tombol ← dan tampilan BLYNK akan berubah seperti gambar 2.13 berikut ini.



Gambar 2.13 Tampilan Aplikasi BLYNK

16. Jika sudah selesai pengaturan pada aplikasi BLYNK, dilanjutkan dengan menyetik program di Arduino yang nantinya akan di download di modul Wifi NodeMCU ESP8266. Program ini yang nanti akan berfungsi supaya android bisa berkomunikasi dengan peralatan input/output. Ketiklah program NodeMCU sesuai contoh yang ada.
17. Jika pengguna ingin menggunakan jaringan pribadi untuk ESP8266 maka di dalam program NodeMCU untuk char ssid dan char pass menggunakan jaringan hotspot dari android itu sendiri, tetapi jika ingin menggunakan jaringan dari hotspot yang lain maka char ssid dan char pass juga menyesuaikan pada android itu sendiri.

```
/* Comment this out to disable prints and save space */  
  
#define BLYNK_PRINT Serial  
  
#include <ESP8266WiFi.h>  
  
#include <BlynkSimpleEsp8266.h>  
  
#include <SoftwareSerial.h>  
  
SoftwareSerial myesp(D1, D2); // RX, TX  
  
// You should get Auth Token in the Blynk App.  
// Go to the Project Settings (nut icon).  
  
char auth[] = "0f6f5336487045689be94fab9c858ad";  
  
// Your WiFi credentials.  
// Set password to "" for open networks.
```



```
char ssid[] = "one phasa";
char pass[] = "ilang1234";
float ultrasonik, suhu, ult, sh;
void setup()
{
  // Debug console
  Serial.begin(9600);
  while (!Serial) {
    ; // wait for serial port to connect. Needed for native USB port
    only
  }
  Serial.println("Goodnight moon!");
  // set the data rate for the SoftwareSerial port
  myesp.begin(9600);
  myesp.println("Hello, world?");
  Blynk.begin(auth, ssid, pass);
  // You can also specify server:
  //Blynk.begin(auth, ssid, pass, "blynk-cloud.com", 80);
  //Blynk.begin(auth, ssid, pass, IPAddress(192,168,1,100), 8080);
}
void loop()
{
  Blynk.run();
  while (mysp.available()>0)
  {
    sh = myesp.parseFloat();
    if(mysp.read() == 'g');
    {
      suhu=sh;
    }
  }
}
```



```

Serial.println(suhu);

}

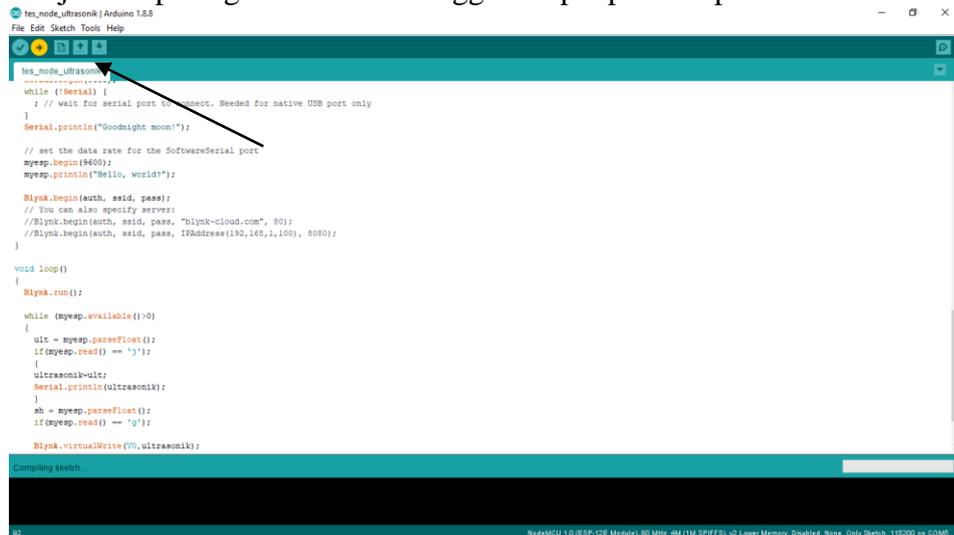
Blynk.virtualWrite(V1, suhu);

}

}

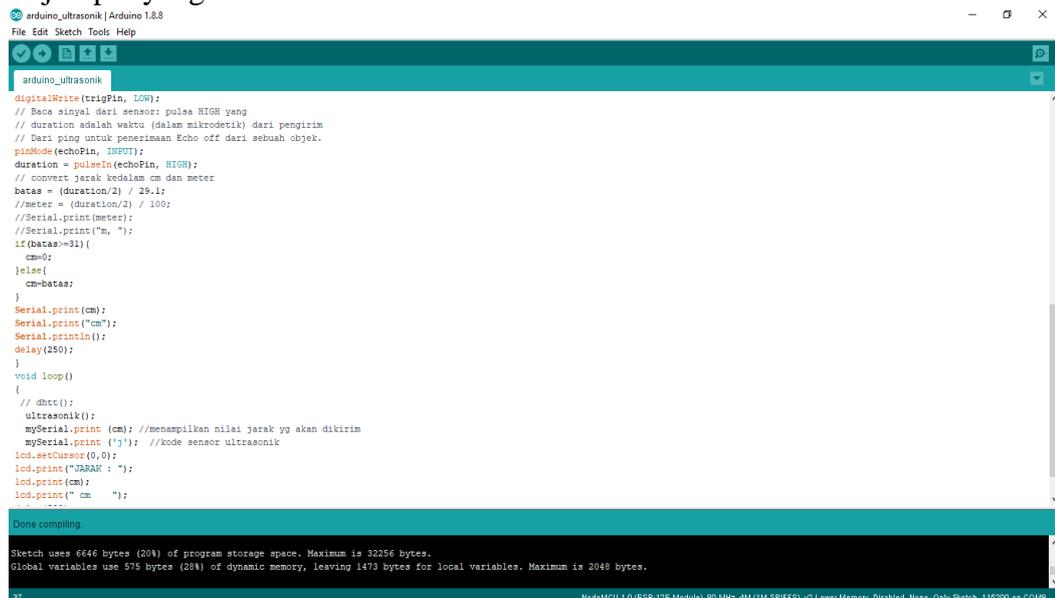
```

18. Jika sudah dilanjutkan dengan mengupload program arduino ke NodeMCU ESP8266 dengan memilih tombol compile di sebelah verify. seperti yang ditunjukkan pada gambar 1.13. Tunggu sampai proses upload selesai.



Gambar 2.14 Proses Upload Program Arduino.

19. Jika sudah selesai maka hubungkan setiap pin komponen menggunakan kabel jumper yang sudah tersedia.



Gambar 2.15 Proses Uploading Selesai.



Tabel 2.1 hasil pengukuran sensor suhu

Pin	Pengukuran Sensor	Pengukuran Thermometer	Persentase Error
12 & 13	28 C	28,1 C	0,1
12 & 13	29 C	29,1 C	0,1
12 & 13	30 C	30,1 C	0,1
12 & 13	32 C	32,1 C	0,1

Analisa: Dari percobaan yang telah dilakukan persentase hasil praktikum yaitu persentase error 0,1%.

PERCOBAAN 3 ON/OFF LAMPU

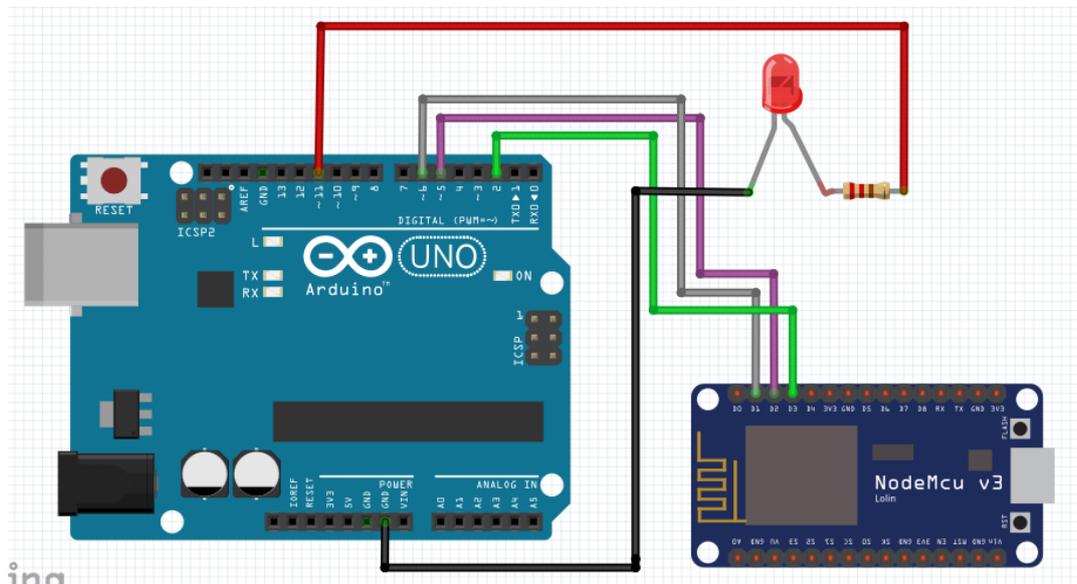
A. Tujuan

1. Mahasiswa dapat memahami program Arduino.
2. Mahasiswa bisa menggunakan software Arduino dengan prosedur yang benar.
3. Mahasiswa dapat memprogram on/off lampu.

B. Komponen Dan Peralatan Yang Digunakan

1. Modul Iot (Internet of Things)
2. Komputer/Laptop

C. Gambar Rangkaian Percobaan



Gambar 3.1 Rangkaian LED

D. Prosedur Percobaan

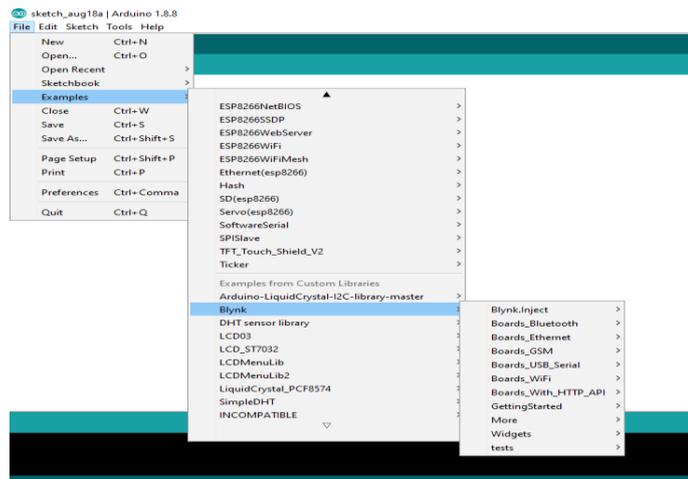
1. Buka program Arduino uno dengan menekan klik ganda icon  pada dekstop komputer.
2. Setelah itu akan muncul tampilan seperti gambar 3.2.



```
File Edit Sketch Tools Help  
sketch_aug18a  
void setup() {  
  // put your setup code here, to run once:  
  
}  
  
void loop() {  
  // put your main code here, to run repeatedly:  
  
}
```

Gambar 3.2 Tampilan Awal Software Arduino

3. Untuk melihat contoh program klik File - klik Example, lalu pilih contoh program yang dibutuhkan.



Gambar 3.3 Contoh Program di Arduino

4. Setelah memilih program yang diinginkan maka klik program tersebut, maka akan muncul *coding* seperti gambar 3.4. Selanjutnya program lah sesuai keinginan.



```
ESP8266_Standalone | Arduino 1.8.8
File Edit Sketch Tools Help
ESP8266_Standalone
*****
/* Comment this out to disable prints and save space */
#define BLYNK_PRINT Serial

#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>

// You should get Auth Token in the Blynk App.
// Go to the Project Settings (nut icon).
char auth[] = "yourAuthToken";

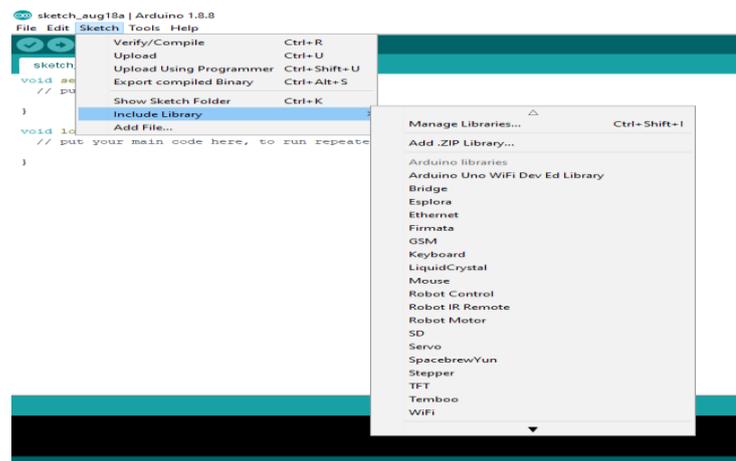
// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "YourNetworkName";
char pass[] = "YourPassword";

void setup()
{
  // Debug console
  Serial.begin(9600);
  Blynk.begin(auth, ssid, pass);
}

void loop()
{
  Blynk.run();
}
```

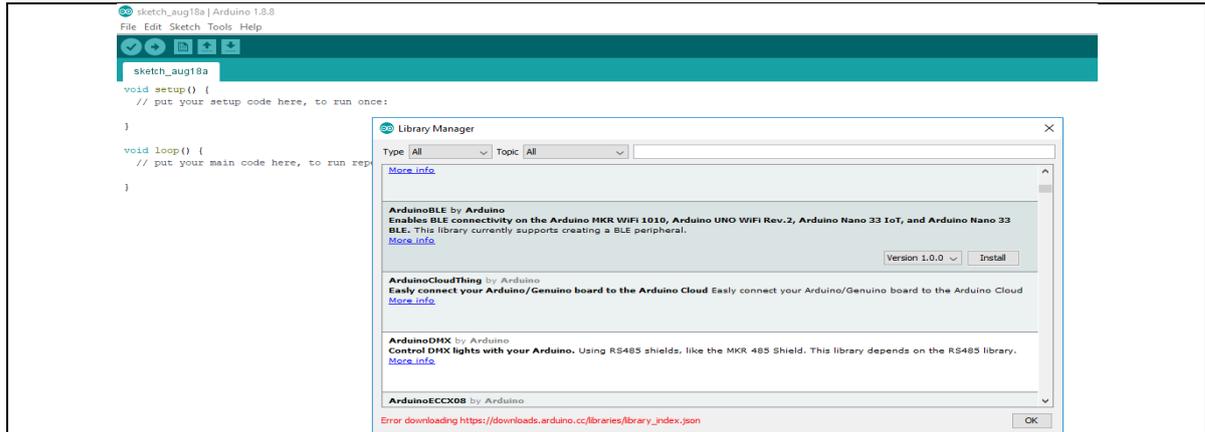
Gambar 3.4 Contoh Program Arduino Yang Sudah Di ketik

5. Jika ingin menambahkan Library maka download terlebih dahulu dengan cara klik Skecth – Include library – Manage libraries atau bisa menekan Ctrl+Shift+I



Gambar 3.5 Cara Menambahkan Library Pada Software Arduino

6. Kemudian akan muncul tampilan seperti gambar 3.6, lalu pilih library yang dibutuhkan setelah itu klik instal. Tunggu hingga selesai mendownload, setelah selesai maka secara otomatis akan tersimpan di software arduino.



Gambar 3.6 Library Arduino Yang Sudah Ditambahkan

7. Jika Library yang diinginkan telah di download maka langkah selanjutnya memprogram sesuai yang petunjuk yang berlaku.

```
//const int PIN_2 = 2;
//const int PIN_13 = 11;

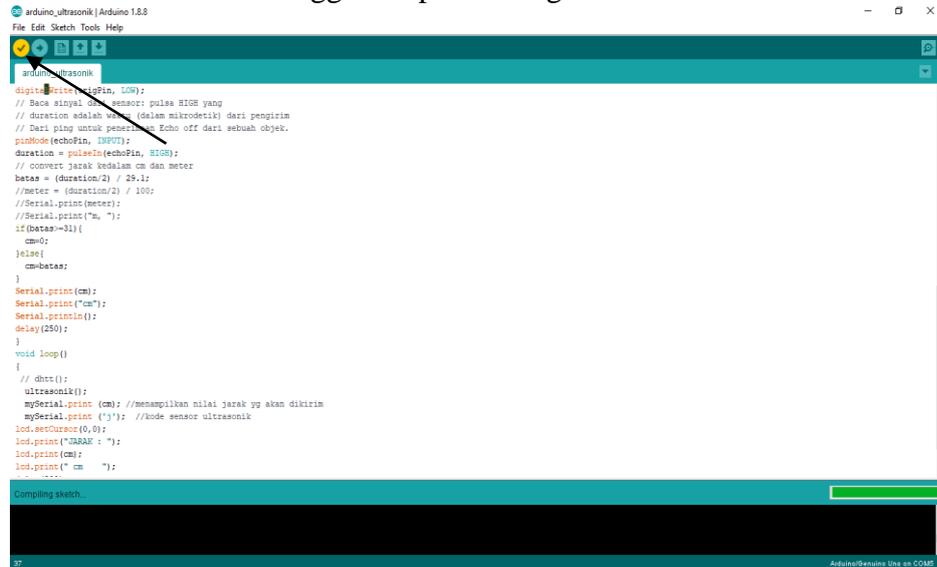
void setup() {
  pinMode (2, INPUT);
  pinMode (11, OUTPUT);
}

void loop() {
  // int kondisiPB = digitalRead (PIN_2);
  if (digitalRead (2) == HIGH)
  {
    digitalWrite(11, HIGH);
  }
  if (digitalRead (2) == LOW)
  {
    digitalWrite(11, LOW);
  }
  // delay (1000);
}
```

8. Jika sudah selesai mengetik program maka dilanjutkan dengan melakukan verify dengan memilih tombol centang (✓) kolom kiri atas software arduino seperti gambar

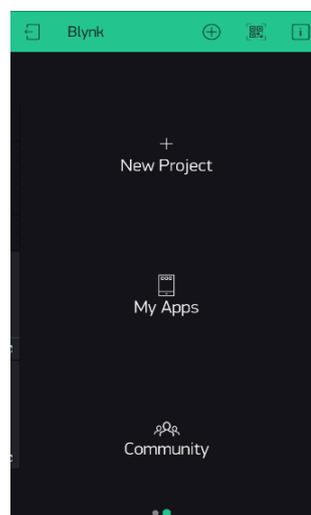


3.7. Verify ini fungsinya untuk mengecek program yang sudah diketik error ataupun tidak error. Kemudian tunggu sampai loading selesai.



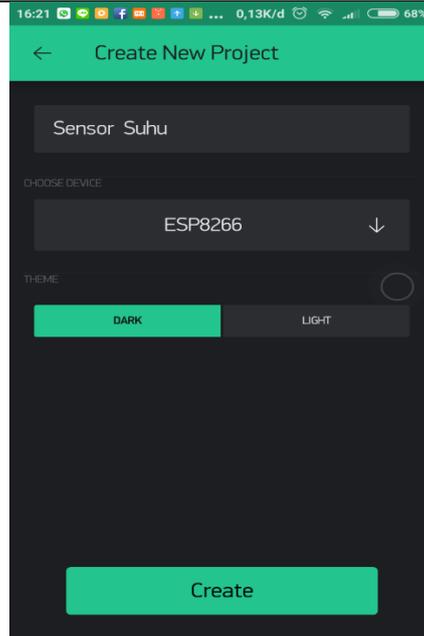
Gambar 3.7 Verify program Arduino

9. Download terlebih dahulu aplikasi Blynk di PlayStore, lalu instal aplikasi tersebut. Setelah menginstal buka aplikasi Blynk kemudian akan muncul tampilan seperti gambar 3.8 klik New Project.



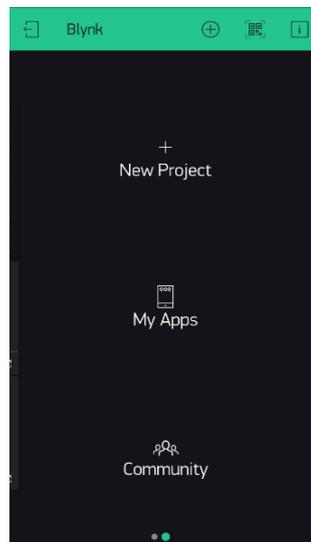
Gambar 3.8 Tampilan Awal Aplikasi BLYNK

10. Kemudian akan tampil seperti gambar 3.9, selanjutnya isi Project Name contoh Sensor Suhu – pilih ESP8266, klik Create.



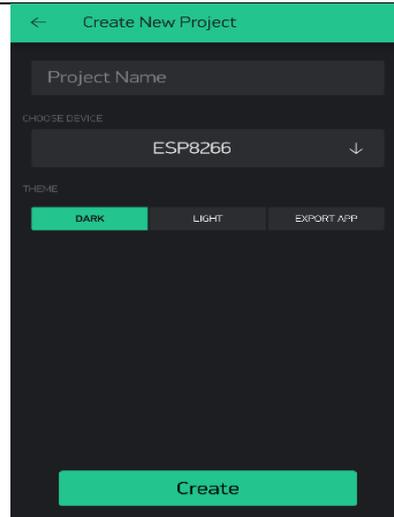
Gambar 3.9 Pengisian Nama Project

11. Download terlebih dahulu aplikasi Blynk di PlayStore, lalu instal aplikasi tersebut. Setelah menginstal buka aplikasi Blynk kemudian akan muncul tampilan seperti gambar 4.0 klik New Project.



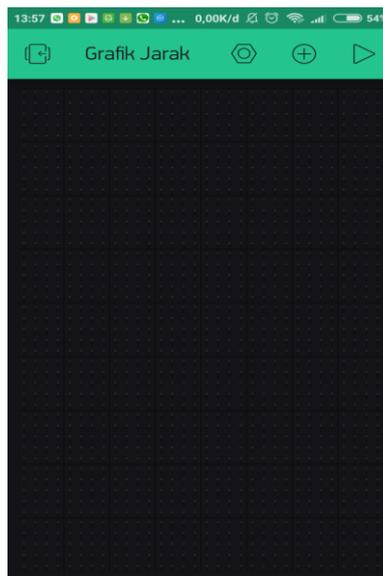
Gambar 4.0 Tampilan Awal Aplikasi BLYNK

12. Kemudian akan tampil seperti gambar 4.1, selanjutnya isi Project Name contoh Sensor Suhu – pilih ESP826, klik Create.



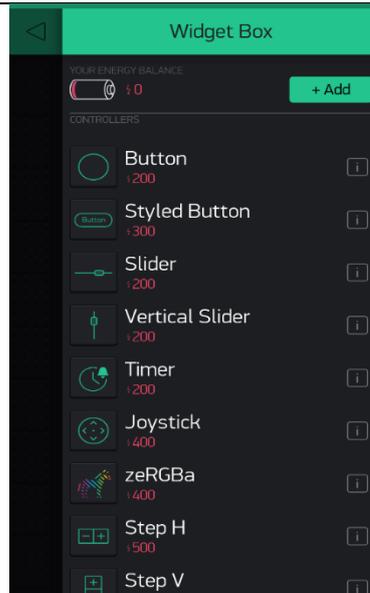
Gambar 4.1 Pengisian Nama Project

13. Kemudian BLYNK akan mengirim token ke email pengguna.
Catatan: jika token tidak terkirim maka periksalah email yang digunakan untuk login ke aplikasi Blynk dan pastikan bahwa sinyal dalam kondisi bagus.
14. Maka akan muncul tampilan project seperti gambar dibawah ini, untuk menambah library klik symbol 



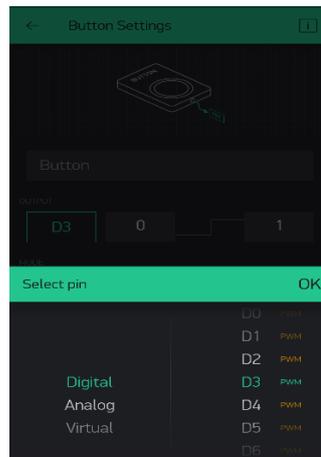
Gambar 4.2 Tampilan pengaturan library

15. Kemudian pilih Button



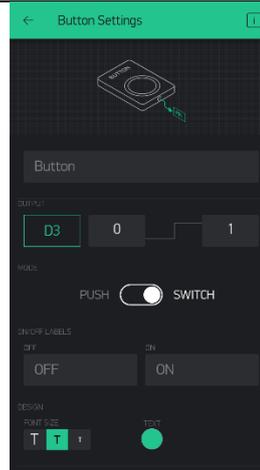
Gambar 4.3 Library Yang Ada di Aplikasi BLYNK

16. Setelah itu isi nama pada kolom button misal “Tombol lampu” dan atur output menjadi Digital dan pilih pin berapa yang akan digunakan pada NodeMCU. Pengaturan pin pada mode Digital berguna untuk mengkoneksi dari NodeMCU ke output lampu.



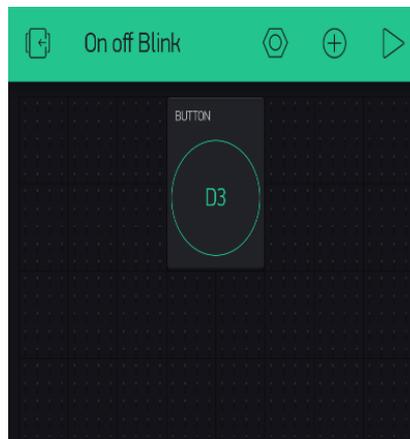
Gambar 4.4 Pengaturan Mode pin pada Blynk

17. Setelah itu setting mode push menjadi switch. Setelah selesai maka klik back.



Gambar 4.5 Pengaturan mode pada tombol button

18. Maka tampilan akan seperti gambar dibawah ini dan siap di operasikan.



Gambar 4.6 Tampilan project yang telah dibuat

19. Jika pengguna ingin menggunakan jaringan pribadi untuk ESP8266 maka di dalam program NodeMCU untuk char ssid dan char pass menggunakan jaringan hospot dari android itu sendiri, tetapi jika ingin menggunakan jaringan dari hospot yang lain maka char ssid dan char pass juga menyesuaikan pada android itu sendiri.

```
/* Comment this out to disable prints and save space */
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <SoftwareSerial.h>
SoftwareSerial myesp(D1, D2); // RX, TX
// You should get Auth Token in the Blynk App.
```



```
// Go to the Project Settings (nut icon).
char auth[] = "0f6f5336487045689be94fab9c858ad";

// Your WiFi credentials.

// Set password to "" for open networks.

char ssid[] = "one phasa";
char pass[] = "ilang1234";
float ultrasonik, suhu, ult, sh;

void setup()
{
  // Debug console
  Serial.begin(9600);
  while (!Serial) {
    ; // wait for serial port to connect. Needed for native USB port
    only
  }
  Serial.println("Goodnight moon!");

  // set the data rate for the SoftwareSerial port
  myesp.begin(9600);
  myesp.println("Hello, world?");
  Blynk.begin(auth, ssid, pass);
  // You can also specify server:
  //Blynk.begin(auth, ssid, pass, "blynk-cloud.com", 80);
  //Blynk.begin(auth, ssid, pass, IPAddress(192,168,1,100), 8080);
}

void loop()
{
  Blynk.run();
  while (myesp.available()>0)
  {
```



```
sh = myesp.parseFloat();  
  
if(myesp.read() == 'g');  
  
{  
  
suhu=sh;  
  
Serial.println(suhu);  
  
}  
  
Blynk.virtualWrite(V1,suhu);  
  
}  
  
}
```

No	Kondisi	Hasil
1	Off	Off
2	On	On

Analisa: Dari hasil percobaan yang telah dilakukan dapat di simpulkan bahwa lampu akan menyesuaikan dari instruksi yang di berikan melalui android.