



**RANCANGAN *RUNNING TEXT* DENGAN *UPDATE*  
DATA MELALUI SMS BERBASIS ARDUINO**

Disusun dan Diajukan untuk Memenuhi Persyaratan Ujian Akhir Memperoleh  
Gelar Sarjana Komputer pada Fakultas Sains dan Teknologi  
Universitas Pembangunan Panca Budi Medan

**SKRIPSI**

**OLEH**

**NAMA : MED ZUARDI RINANDA NST**  
**NPM : 1724370999**  
**PROGRAM STUDI : SISTEM KOMPUTER**

**FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS PEMBANGUNAN PANCA BUDI  
MEDAN**

**2020**

## **ABSTRAK**

**MHD. ZUARDI RINANDA NST**

### **RANCANGAN RUNNING TEXT DENGAN UPDATE DATA MELALUI SMS BERBASIS ARDUINO**

**2019**

Running text atau yang biasa dikenal dengan text berjalan merupakan media penyampaian informasi yang paling sering dijumpai sebagai sarana advertising yang bersifat informatif. Display Running text pada umumnya berupa led-led yang disambung dan dirangkai menjadi deretan led ataupun dapat berupa dot matrix. Penyampaian informasi dengan menampilkan running text tersebut belum efektif dan efisien dalam hal penggunaan. Masih terdapat kekurangan yang memungkinkan manusia mengalami kesalahan penglihatan dan terlalu lama menunggu text yang akan ditampilkan. Maka dari itu dibutuhkan alat penyampaian informasi yang dapat bekerja secara maksimal untuk meminimalisir kekurangan dengan merancang alat pengontrolan running text menggunakan via SMS melalui telpon genggam. Penulisan teks melalui SMS dapat mendukung penulisan yang lebih efisien terutama untuk menyebarkan informasi yang sama pada beberapa lokasi running text, sehingga penulisan teks tidak lagi harus berada di dekat running text melainkan dapat dilakukan dipusat informasi yang jauh dari letak running text.

**Kata Kunci :** Arduino Uno R3, Display P10, Mikrokontroler ATmega328, Mini LM2596, SIM800L

# DAFTAR ISI

	Halaman
KATA PENGANTAR.....	i
DAFTAR ISI.....	ii
DAFTAR GAMBAR.....	iv
DAFTAR TABEL.....	v
DAFTAR LAMPIRAN.....	vi
<b>BAB I PENDAHULUAN</b>	
1.1 Latar Belakang Masalah.....	1
1.2 Perumusan Masalah.....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	3
<b>BAB II LANDASAN TEORI</b>	
2.1 Running Text.....	4
2.2 Mikrokontroller.....	7
2.2.1 Fitur AVR ATmega328.....	11
2.2.2 Deskripsi Pin Mikrokontroler ATmega 328.....	12
2.2.3 Arsitektur CPU ATMEGA 328.....	16
2.3 Mikrokontroler Arduino Uno R3.....	19
2.3.1 Power.....	21
2.3.2 Memori.....	22
2.3.3 Input & Output.....	22
2.3.4 Komunikasi.....	23
2.3.5 Software Arduino.....	24
2.4 Bahasa Pemrograman Arduino.....	24
2.4.1 Reset Otomatis (Software) .....	27
2.4.2 Proteksi Arus lebih USB.....	29
2.4.3 Karakteristik Fisik.....	29
2.5 Sim800L.....	30
2.5.1 Spesifikasi Modul SIM800L.....	31
2.6 Mini LM2596.....	32
2.7 Power Supply.....	34
2.7.1 Klasifikasi Umum Power Supply.....	34
2.8 Android.....	36
2.8.1 Sejarah Android.....	36
2.8.2 Kelebihan atau Keunggulan Android.....	39
2.8.3 Kelemahan atau Kekurangan Android.....	40
2.8.4 Fitur dan Arsitektur Android.....	41

### **BAB III PERANCANGAN DAN PEMBUATAN**

3.1	Tahapan Penelitian.....	43
3.2	Metode Pengumpulan Data.....	44
3.3	Analisis Sistem Sedang Berjalan.....	45
3.3.1	Identifikasi Kebutuhan.....	45
3.3.2	Desain Perangkat Keras.....	46
3.4	Alat dan Bahan.....	47
3.5	Spesifikasi Sistem.....	47
3.6	Rangkaian Keseluruhan Sistem.....	49
3.7	Software Pemrograman.....	49
3.7.1	Software Arduino IDE.....	49
3.7.2	Program Running Text.....	50
3.7.3	Cara Menggunakan Software Arduino IDE.....	56

### **BAB IV HASIL PENELITIAN DAN PEMBAHASAN**

4.1	Pengujian Power Supply 5V 2.....	60
4.2	Pengujian Tegangan Output Step Down LM2596 Mini.....	61
4.3	Percobaan.....	62
4.3.1	Gambar Tampilan Hasil Pengujian Alat.....	62

### **BAB V PENUTUP**

5.1	Kesimpulan.....	65
5.2	Saran.....	65

### **DAFTAR PUSTAKA**

### **BIOGRAFI PENULIS**

### **LAMPIRAN-LAMPIRAN**

## DAFTAR GAMBAR

	Halaman
Gambar 2.1 Running Text.....	7
Gambar 2.2 Deskripsi Pin Mikrokontroler ATmega328.....	13
Gambar 2.3 Arduino Uno R3.....	29
Gambar 2.4 Konfigurasi Pin Arduino .....	30
Gambar 2.5 Sim800L.....	30
Gambar 2.6 Mini LM2596.....	33
Gambar 2.7 Power Supply.....	34
Gambar 2.8 Aplikasi Android.....	37
Gambar 2.9 Fitur dan Arsitektur Android.....	42
Gambar 3.1 Diagram Alur Penelitian.....	43
Gambar 3.2 Diagram Blok Sistem.....	47
Gambar 3.3 Flow Chart Sistem.....	48
Gambar 3.4 Rangkaian Sistem.....	49
Gambar 3.5 Tampilan Jendela Arduino IDE.....	56
Gambar 3.6 Tampilan Arduino IDE Menu Tools.....	57
Gambar 3.7 Tampilan Sketch dan Source Code.....	59
Gambar 4.1 Pengujian Power Supply.....	60
Gambar 4.2 Pengujian Tegangan Output LM2596 Mini.....	61
Gambar 4.3 Tampilan Hasil Display P10.....	62
Gambar 4.4 Tampilan Hasil Display P10.....	63
Gambar 4.5 Tampilan Hasil Display P10.....	64

## DAFTAR TABEL

	Halaman
Tabel 2.1 Konfigurasi Port B.....	14
Tabel 2.2 Konfigurasi Port C.....	15
Tabel 2.3 Konfigurasi Port C.....	16
Tabel 2.4 Deskripsi Arduino.....	20

## DAFTAR LAMPIRAN

	Halaman
Lampiran 2. Lembar pengesahan Skripsi.....	L-2
Lampiran 3. Abstrak .....	L-3
Lampiran 4. Kata Pengantar.....	L-4
Lampiran 5. Daftar Isi .....	L-5
Lampiran 6. Daftar Gambar.....	L-6
Lampiran 7. Daftar Tabel .....	L-7

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Perkembangan teknologi saat ini kian pesat. Seiring berkembangnya teknologi tersebut, cara dan alat berkomunikasi pun semakin berkembang. Perkembangan cara berkomunikasi maupun alat berkomunikasi ini mempengaruhi cara penyampaian informasi yang dilakukan sehari-hari. Penyampaian informasi pada jarak yang jauh sekalipun saat ini sudah bisa terlaksana, ini semua didukung oleh pesatnya perkembangan teknologi khususnya perkembangan pada alat informasi. Layanan internet memungkinkan penyampaian informasi dengan jarak yang jauh akan bisa diselesaikan dalam hitungan menit bahkan detik. *Running text* merupakan salah satu media publikasi digital yang terdiri atas *Light Emitting Diode* (LED) yang disusun rapat dengan pola yang teratur dan terdapat titik koordinat di tiap LED nya, sehingga dapat dibuat pola pemunculan cahaya yang membentuk tulisan maupun gambar tertentu. Penggunaan LED menjadi media publikasi bukanlah hal yang baru, banyak tempat-tempat yang menggunakan LED ini sebagai sarana penyampaian informasi maupun promosi seperti di toko-toko, sekolah, rumah sakit, serta tempat-tempat umum lainnya.

Pengaturan *running text* yang sudah ada biasanya dengan menggunakan koneksi pc, kelemahannya adalah terlalu repot ketika ingin mengganti tulisan atau text yang ada dalam *running text*, sehingga perlu solusi untuk mempermudah pemrograman text pada *running text*. Tentu saja cara tersebut memiliki kekurangan



dalam melakukan pengupdate-an, karena pengupdate-an hanya bisa dilakukan disatu tempat saja, dengan kata lain pembaharuan informasi pada *Running Text* sangat statis. Akan tetapi, hal tersebut tidak akan terjadi bila pembaharuan informasinya dilakukan menggunakan media lain. Salah satu solusi adalah dengan memanfaatkan salah satu fitur dari telepon selular yaitu melalui via SMS secara manual. Sehingga diharapkan dapat menghemat waktu bagi pusat informasi untuk menyebarkan informasi ke lokasi *running text*.

*Running text* sebagai media informasi dengan komunikasi melalui SMS (*short message service*) dari ponsel, kemudian pesan diterima oleh modul GSM (*global system for mobile communications*) sim800L yang diteruskan ke *mikrokontroller* Arduino untuk ditampilkan pada *display running text* yang dapat berupa huruf, angka dan tanda baca.

## **1.2 Perumusan Masalah**

Pada kesempatan ini penulis ingin memberikan suatu rumusan dalam menganalisa perancangan alat ini. Beberapa permasalahan yang mendasar untuk melakukan perancangan dan implementasi sistem ini adalah :

1. Bagaimana alat ini dapat mengirimkan informasi?
2. Apakah Fungsi dari alat ini?

## **1.3 Batasan Masalah**

Pembahasan perancangan dalam skripsi ini telah penulis batasi agar tidak terlalu luas dan menyimpang dari topik yang telah ditentukan, batasan permasalahan sebagai berikut:

1. Pembahasan tentang *mikrokontroler* Arduino Uno R3 hanya sebatas yang berkaitan dengan perancangan ini.
2. Pembahasan mengenai komponen dukung yang meliputi : dan komponen lainnya hanya sebatas teori umum dan yang berkaitan dengan rancangan *running text*.
3. *Running text* atau teks berjalan yang digunakan berupa *display* p10 yang dirancang sedemikian rupa.

#### **1.4 Tujuan Penelitian**

Berikut ini beberapa tujuan penelitian yang akan dibahas dalam penelitian ini yaitu sebagai berikut :

1. Dapat menyampaikan informasi melalui SMS yang ditampilkan pada *Running Text* dan dapat dibaca.
2. Memudahkan komunikasi dalam menyampaikan informasi.
3. Dapat mengirimkan informasi berupa pengumuman dari SMS.

#### **1.5 Manfaat Penulisan**

Berikut ini beberapa manfaat penelitian yang akan dibahas dalam penelitian ini yaitu sebagai berikut :

1. Rancangan *Running Text* dengan Update Data melalui SMS berbasis Arduino ini dapat mempermudah publik atau target informasi mendapatkan informasi dengan cepat dan akurat
2. Pengurangan terhadap segala prosedur pengiriman informasi manual.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Running Text**

*Running Text* adalah salah satu media elektronik yang sangat berguna untuk menyampaikan pesan dan informasi yang dapat juga digunakan sebagai Sarana Iklan. *Running Text* juga dikenal dengan sebutan *Moving Sign*. Dalam pengembangannya, *Display Running text* kini hadir tidak hanya menampilkan rangkaian tulisan berjalan, tapi juga bisa untuk menampilkan gambar atau logo. *Running Text* pada umumnya digunakan untuk memperkenalkan produk atau jasa, dikarenakan display yang beragam serta efisiensi penyampaian informasi. Kita tahu bahwa mata manusia sangat cepat mendeteksi suatu hal yang berbeda dari biasanya, khususnya pada hal atau benda yang bersinar, memiliki warna , mencolok, dan berbeda dari sekitarnya. Inilah yang menjadi dasar perangkaian warna dari *display running text* mempengaruhi orang di sekeliling untuk menatapnya.

Pesatnya perkembangan teknologi pada saat ini, mempengaruhi perkembangan berbagai macam tampilan digital maupun elektronik, hal ini menyebabkan inovasi inovasi bermunculan dalam menciptakan tampilan tampilan baru dari *display running text*. Beberapa diantaranya munculnya berbagai bentuk *display LED*. Mulai dari ukuran yang dapat disesuaikan dengan keinginan sampai warna yang bisa diatur dengan kebutuhan. Pada tahun 2014 produk *Full Color* sudah dapat ditemui pada toko toko elektronik.

Pada mulanya, *Running text* hanya dipakai untuk menampilkan tulisan, dengan jumlah huruf yang terbatas dan karakter tertentu, dengan perkembangan teknologi saat ini, kita dapat menemukan berbagai macam *running text* yang telah dimodifikasi untuk menampilkan logo bahkan gambar, model perubahan maupun pergerakan tulisannya pun semakin menarik.

Para ahli elektronik menemukan peluang dalam menciptakan dan membangun sebuah teknologi yang bisa berguna dan dapat digunakan maupun dimanfaatkan untuk menjadi keuntungan. *Led running text* ini memiliki fitur komunikasi secara seri yang dapat mendistribusikan tulisan berjalan maupun bergerak yang dapat dimanipulasi melalui komputer, laptop dan juga remot. Pada *running text* ini kita juga dapat mengatur waktu dengan menggunakan *Real Time Clock* (RTC) sehingga dapat menampilkan detik, menit, jam, hari, tanggal dan tahun. *Led running text* ini tidak akan berubah walaupun dimatikan artinya data akan tersimpan di dalam memori.

Ada satu lagi *running text* yang dipasarkan sebagai media informasi ini yaitu *moving sign*. *Moving sign* ini adalah sebuah tampilan elektronik yang menggunakan led dan menghasilkan bermacam cahaya yang sangat menarik dan berbeda, tergantung dengan keinginan pengguna. *Moving sign* ini adalah sebuah media elektronik yang berguna untuk dijadikan sebagai media pemasaran, pemberitahuan informasi serta sebagai media untuk memperindah dekorasi kantor. *Moving led* ataupun *led running text* ini tidak bisa kita samakan dengan banner ataupun spanduk, karena banner dan spanduk memiliki masa kadaluarsa bahan yang tidak

dapat bertahan lama dan cepat terjadi kerusakan. Sedangkan *moving sign* dan *led running text* memiliki sifat permanen dan tahan lama yang isinya bisa diubah sesuai dengan kebutuhan atau sesuai dengan selera kita. Untuk desainnya, tersedia *desain available* yang berarti telah dirancang sedemikian rupa agar saat dibeli dapat langsung dibawa, tapi ada juga *desain custom* yang berarti kita bisa mengajukan permintaan dahulu untuk bentuk dan desainnya sesuai keinginan dan sesuai dengan kebutuhan. Sehingga membuat isi dari *running text* tersebut lebih inovatif dan menarik.

*Running text* yang ada yaitu berupa led-led yang dirangkai dan disatukan menjadi deretan led ataupun dapat berupa dot matrix. Dot matrix adalah deretan led yang membentuk array dengan jumlah kolom dan baris tertentu, sehingga titik-titik yang menyala dapat membentuk suatu karakter angka, huruf, tanda baca dan sebagainya. Program dot matrix 5 x 7 menggunakan *shift register 74hc595* untuk mengendalikan nyala *array led*, dan *input teks*. Jika dot matrix tidak menggunakan *shift register*, maka led bisa menyala bersamaan satu kolom atau satu baris.

(Imelda U.V. Simanjuntak dan Asep Suhendar, 2018)



**Gambar 2.1** Running Text

## **2.2 Mikrokontroler**

Mikrokontroler adalah sebuah alat yang bertugas sebagai pengontrol rangkaian elektronik dan umumnya dapat menyimpan program didalamnya. Mikrokontroler tersusun dalam satu chip dimana prosesor, memori dan i/o terintegrasi menjadi satu kesatuan kontrol sistem yang dapat bekerja secara inovatif. Mikrokontroler juga bisa didefinisikan sebagai alat elektronika digital yang memiliki masukan dan keluaran dan juga kontrol dengan program yang dapat ditulis dan dihapus dengan cara tertentu. Sebagai analogi, ilustrasikan kemampuan kita pada waktu belajar membaca dan menulis, saat kita telah mampu melakukan hal tersebut kita dapat membaca tulisan apapun baik dari media tulis, cetak dan media digital sekalipun, dan kitapun bisa pula menulis hal-hal sebaliknya. Dan sebaliknya, apabila kita telah mampu membaca dan menulis data maka kita mampu pula membangun program untuk merancang sebuah sistem dengan pengaturan otomatis menggunakan mikrokontroler sesuai keinginan kita. (Sumarsono, Dwiatmi Saptaningtyas, 2018)

Pada umumnya perintah dan tindakan yang digunakan dalam mikro controller adalah tindakan-tindakan yang bersifat perulangan dan berjalan otomatis. Sehingga mikrocontroller banyak ditemukan di sistem kontrol mesin, peralatan rumah tangga, alat berat, mainan dan sebagainya. Dengan penggunaan mikrokontroler ini maka :

1. Sistem elektronik akan menjadi lebih ringkas.
2. Rancang bangun sistem elektronik akan lebih cepat karena sebagian besar dari sistem adalah perangkat lunak yang mudah dimodifikasi.
3. Pencarian gangguan lebih mudah ditelusuri karena sistemnya yang kompak.

Hadirnya mikrocontroller dalam alat-alat elektronika membuat pekerjaan akan menjadi efisien dan efektif. Sebab sistem elektronik akan menjadi semakin ringkas dan perancangan sistem elektronik menjadi lebih efektif karena tidak lagi perlu merancang “otak” inti. Ada banyak jenis mikrocontroller yang masing-masing memiliki keluarga atau series sendiri, sehingga diperlukan pemahaman yang cukup untuk menggunakan dan memilih mikrocontroller. Hingga saat ini mikrocontroller telah berkembang begitu pesat dan memiliki kemampuan olah data besar untuk sebuah komputer mini yang ringkas. Bahkan sebuah mikrocontroller saat ini telah mampu mengolah bahasa pemrograman untuk menjalankan fungsi-fungsi yang sangat kreatif karena umumnya telah dilengkapi lampu *Light Emitting Diode* (LED), sensor, monitor dan sebagainya. (Bahrin, 2017)

Mikrokontroler pada golongan yang serupa akan mempunyai kemiripan dalam hal arsitektur dan kompatibilitas pemrograman, yang membedakan hanya dalam kemasan fisik, banyaknya pin, dan fitur-fitur yang dimiliki dari mikrokontroler tersebut. Beberapa contoh keluarga mikrokontroler antara lain:

1. Keluarga MCS-48 (Intel)
2. Keluarga MC68H08 (Motorola)
3. Keluarga AT89S (Intel)
4. Keluarga AT90, ATtiny, ATmega (Atmel)
5. Keluarga MC68HC05 (Motorola)
6. Keluarga PIC 8 (Mikrochip)
7. Keluarga MC68H11 (Motorola)
8. Keluarga Z80 (Zilog)
9. Keluarga MCS-51 (Intel)

Perbedaan mendasar dari mikrokontroler yang sering dijumpai di pasaran adalah berdasarkan arsitekturnya. Ada dua tipe arsitektur yang dapat kita temukan yaitu arsitektur *Complex Instruction Set Computing* (CISC) dan arsitektur *Reduced Instruction Set Computing* (RISC). Arsitektur CISC yaitu mikrokontroler yang mempunyai bahasa pemrograman lebih luas dan lebih banyak, tetapi memiliki fasilitas yang terbatas.

Keluarga AT89S adalah salah satu keluarga mikrokontroler pendahulu produksi Atmel jenis MCS yang lebih dikenal dengan mikrokontroler MCS-51 dengan arsitektur CISC. Arsitektur RISC yaitu mikrokontroler yang mempunyai



perintah yang terbatas, tetapi memiliki fasilitas yang banyak. Keluarga ATmega adalah salah satu perkembangan teknologi mikrokontroler buatan Atmel tipe AVR atau biasa disebut mikrokontroler AVR dengan arsitektur RISC. Perkembangan teknologi mikrokontroler berarsitektur CISC ke arsitektur RISC bertujuan untuk mempercepat mikrokontroler dalam mengeksekusi suatu instruksi. Mikrokontroler jenis MCS memiliki kecepatan frekuensi kerja 1/12 kali frekuensi osilator yang digunakan, sedangkan pada kecepatan frekuensi kinerja mikrokontroler jenis AVR sama dengan kecepatan frekuensi osilator yang digunakan. Apabila mikrokontroler jenis MCS dan mikrokontroler AVR menggunakan frekuensi osilator yang sama, maka mikrokontroler jenis AVR mempunyai kecepatan kinerja 12 kali lebih cepat dibandingkan dengan mikrokontroler jenis MCS. Agar sebuah mikrokontroler dapat bekerja, maka mikrokontroler tersebut memerlukan komponen tambahan yang kemudian disebut dengan sistem minimum. Untuk merancang sistem minimal paling tidak dibutuhkan sistem clock dan reset, walaupun pada beberapa mikrokontroler sudah menyediakan fitur sistem clock bawaan, sehingga tanpa rangkaian tambahan pun mikrokontroler dapat beroperasi. Yang dimaksud dengan sistem minimal adalah sebuah rangkaian mikrokontroler yang sudah bisa dipakai untuk menjalankan sebuah aplikasi. Seperangkat IC mikrokontroler tidak akan berfungsi apabila hanya berdiri sendiri. Pada dasarnya sebuah sistem minimal mikrokontroler AVR memiliki prinsip yang sama. Pada pembahasan ini Mikrokontroler yang dipakai adalah AVR Atmega328.

### 2.2.1 Fitur AVR ATmega328

ATmega328 adalah mikrokontroler produksi setelah atmel yang memiliki disain arsitektur RISC yang mana pada tahap pelaksanaannya data lebih cepat dari pada arsitektur CISC. Mikrokontroler ini memiliki beberapa fitur antara lain :

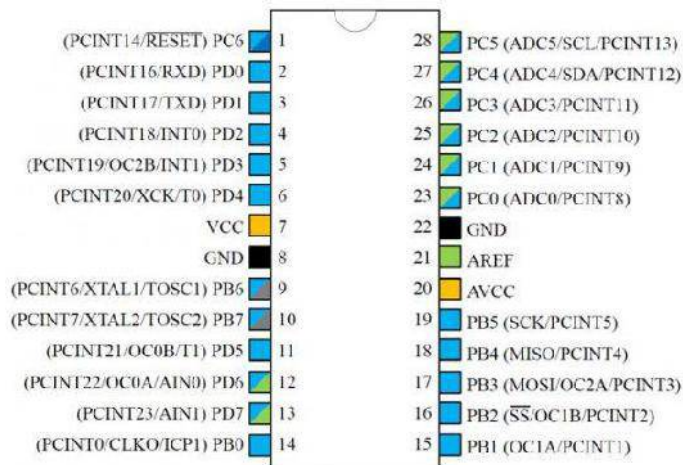
1. Sebanyak 130 jenis instruksi yang hampir keseluruhannya dijalankan dalam satu siklus clock.
2. Ada 32 x 8-bit register serba guna.
3. Kecepatan hingga 16 MIPS dengan clock 16 MHz.
4. 32 KB Flash memori dan pada arduino mempunyai bootloader yang memakai 2 KB dari flash memori sebagai *bootloader*.
5. Mempunyai *Electrically Erasable Programmable Read Only Memory* (EEPROM) sebesar 1KB yang berfungsi untuk wadah penyimpanan data semi permanent, karena EEPROM tetap dapat menyimpan data meskipun catu daya dimatikan.
6. Memiliki *Static Random Access Memory* (SRAM) sebesar 2KB.
7. Mempunyai pin I/O digital sebanyak 14 pin 6 salah satunya *Pulse Width Modulation* (PWM) output.
8. Master / Slave SPI Serial interface.

Mikrokontroler ATmega 328 mempunyai arsitektur Harvard, yaitu arsitektur untuk memisahkan memori untuk pengkodean program dan memori untuk data sehingga dapat meningkatkan kinerja dan parallelism. Perintah-perintah dalam memori program dijalankan dalam satu alur tunggal, dimana pada saat satu

perintah dikerjakan perintah berikutnya sudah diambil dari memori program. Konsep ini yang memungkinkan instruksi – instruksi dapat dijalankan dalam setiap satu siklus clock. 32 x 8-bit register serba guna dipakai untuk mendukung operasi pada Arithmetic Logic unit (ALU) yang bisa dilaksanakan dalam satu tahap. Enam dari register serbaguna ini bisa dipakai sebagai 3 buah register pointer 16-bit pada mode pengalamatan tidak langsung untuk mengambil data pada ruang memori data. Ketiga register pointer 16-bit ini disebut dengan register X (gabungan R26 dan R27), register Y (gabungan R28 dan R29), dan register Z (gabungan R30 dan R31). Hampir semua instruksi AVR memiliki format 16-bit. Setiap alamat memori program terdiri dari instruksi 16-bit atau 32-bit. Selain register serba guna tersebut, terdapat register lain yang terpetakan dengan teknik memory mapped I/O selebar 64 *byte*. Beberapa register ini digunakan untuk fungsi khusus antara lain sebagai *register control Timer/ Counter, Interupsi, ADC, USART, SPI, EEPROM*, dan fungsi I/O lainnya. Register – register ini menempati memori pada alamat 0x20h – 0x5Fh. (Ahmad Fatoni, dkk 2015)

### **2.2.2 Deskripsi Pin Mikrokontroler ATmega 328**

ATmega328 mempunyai 3 buah PORT utama yaitu PORTB, PORTC, dan PORTD dengan jumlah keseluruhan pin input/output sebanyak 23 pin. PORT ini dapat digunakan untuk input/output digital atau digunakan sebagai periperhal lainnya dapat dilihat pada gambar 2.2.



**Gambar 2.2** Deskripsi Pin Mikrokontroler ATmega328

Adapun nama dan fungsi dari setiap pin pada Mikrokontroler ATmega328:

1. Port B merupakan jalur data 8 bit yang bisa digunakan untuk input/output. Selain itu PORTB juga bisa difungsikan sebagai alternatif seperti di bawah ini.
  - a. OC1A (PB1), OC1B (PB2) dan OC2 (PB3) dapat difungsikan sebagai keluaran PWM (Pulse Width Modulation).
  - b. MOSI (PB3), MISO (PB4), SCK (PB5), SS (PB2) merupakan jalur komunikasi SPI.
  - c. ICP1 (PB0), berfungsi sebagai Timer Counter 1 input capture pin.
  - d. TOSC1 (PB6) dan TOSC2 (PB7) dapat difungsikan sebagai sumber clock external untuk timer.
  - e. XTAL1 (PB6) dan XTAL2 (PB7) merupakan sumber clock utama mikrokontroler.

- f. Selain itu pin ini juga berfungsi sebagai jalur pemrograman serial (ISP).

**Tabel 2.1** Konfigurasi Port B

	Alternate Function
PB7	XTAL2 (Chip Clock Oscillator pin 2) TOSC2 (Timer Oscillator pin 2) PCINT7 (Pin Change Interrupt 7)
PB6	XTAL1 (Chip Clock Oscillator pin 1 or External Clock input) TOSC21 (Timer Oscillator pin 1) PCINT6 (Pin Change Interrupt 6)
PB5	SCK (SPI Bus Master Clock Input) PCINT5 (Pin Change Interrupt 5)
PB4	MISO (SPI Bus Master Output/Slave Output) PCINT4 (Pin Change Interrupt 4)
PB3	MOSI (SPI Bus Master Output/Slave Input) OC2A (Timer/Counter2 Output Compare Match A Output) PCINT3 (Pin Change Interrupt 3)
PB2	SS (SPI Bus Master Slave Select) OC1B (Timer/Counter1 Output Compare Match B Output) PCINT2 (Pin Change Interrupt 2)
PB1	OC1A (Timer/Counter1 Output Compare Match A Output) PCINT1 (Pin Change Interrupt 1)
PB0	ICP1 (Timer/Counter1 Input Capture Input) CLKO (Divided System Clock Output) PCINT0 (Pin Change Interrupt 0)

2. Port C Port C merupakan jalur data 7 bit yang bisa digunakan sebagai Input/output digital. Fungsi alternatif PORTC adalah sebagai berikut.
- a. ADC6 channel (PC0, PC1, PC2, PC3, PC4, PC5) dengan resolusi sebesar 10 bit. ADC bisa kita gunakan untuk mengubah input yang berupa tegangan analog menjadi data digital.
  - b. I2C (SDA dan SDL) merupakan salah satu fitur yang terdapat pada PORTC. I2C digunakan untuk komunikasi dengan sensor atau device lain yang memiliki komunikasi data tipe I2C seperti sensor kompas, accelerometer nunchuck.

**Tabel 2.2** Konfigurasi Port C

	Alternate Function
PC6	XTAL1 (Chip Clock Oscillator pin 1 or External Clock input) TOSC21 (Timer Oscillator pin 1) PCINT6 (Pin Change Interrupt 6)
PC5	SCK (SPI Bus Master Clock Input) PCINT5 (Pin Change Interrupt 5)
PC4	MISO (SPI Bus Master Output/Slave Output) PCINT4 (Pin Change Interrupt 4)
PC3	MOSI (SPI Bus Master Output/Slave Input) OC2A (Timer/Counter2 Output Compare Match A Output) PCINT3 (Pin Change Interrupt 3)
PC2	SS (SPI Bus Master Slave Select) OC1B (Timer/Counter1 Output Compare Match B Output) PCINT2 (Pin Change Interrupt 2)
PC1	OC1A (Timer/Counter1 Output Compare Match A Output) PCINT1 (Pin Change Interrupt 1)
PC0	ICP1 (Timer/Counter1 Input Capture Input) CLKO (Divided System Clock Output) PCINT0 (Pin Change Interrupt 0)

3. Port D Port D merupakan jalur data 8 bit yang masing-masing pin-nya juga dapat difungsikan sebagai input/output. Sama seperti Port B dan Port C, Port D juga memiliki fungsi alternatif dibawah ini.
  - a. USART (Universal Synchronous and Asynchronous serial Receiver and Transmitter), merupakan salah satu mode komunikasi yang dimiliki oleh Mikrokontroler ATmega8535. USART memiliki 2 pin (RxD dan TxD) untuk *Asynchronous* dan 3 bit TxD, RxD, xCK untuk *Synchronous*.
  - b. Interrupt (INT0 dan INT1) merupakan pin dengan fungsi khusus sebagai interupsi hardware. Interupsi biasanya digunakan sebagai selaan dari program, misalkan pada saat program berjalan kemudian terjadi interupsi *hardware/software* maka program utama akan berhenti dan akan menjalankan program interupsi.

- c. XCK dapat difungsikan sebagai sumber clock external untuk USART, namun kita juga dapat memanfaatkan clock dari CPU, sehingga tidak perlu membutuhkan external clock.
- d. T0 dan T1 berfungsi sebagai masukan counter external untuk timer 1 dan timer 0.
- e. AIN0 dan AIN1 keduanya merupakan masukan input untuk analog comparator.

**Tabel 2.3** Konfigurasi Port C

	<b>Alternate Function</b>
PD7	AIN1 (Analog Comparator Negative Input) PCINT23 (Pin Change Interrupt 23)
PD6	AIN0 (Analog Comparator Positive Input) OC0A (Timer/Counter() Output Compare Match A Output) PCINT22 (Pin Change Interrupt 22)
PD5	T1 (Timer/Counter 1 External Counter Input) OC0B (Timer/Counter0 Output Compare Match B Output) PCINT21 (Pin Change Interrupt 21)
PD4	XCK (USART External Clock Input/Output) T0 (Timer/Counter 0 External Counter Input) PCINT20 (Pin Change Interrupt 20)
PD3	INT 1 (External Interrupt 1 Input) OC2B (Timer/Counter2 Output Compare Match B Output) PCINT19 (Pin Change Interrupt 19)
PD2	INT 0 (External Interrupt 0 Input) PCINT18 (Pin Change Interrupt 18)
PD1	TXD (USART Output Pin) PCINT17 (Pin Change Interrupt 17)
PD0	RXD (USART Output Pin) PCINT16 (Pin Change Interrupt 16)

### 2.2.3 Arsitektur CPU ATMEGA 328

Fungsi utama CPU adalah memastikan pengekseskuan instruksi dilaksanakan dengan baik. CPU harus bisa mengakses memori, melakukan perhitungan, mengatur peripheral, dan menangani interupsi. Ada 32 General purpose register yang membantu ALU berjalan. Untuk operasi aritmatika dan logika, operand berasal dari dua buah register umum dan hasil operasi dituangkan kembali ke register. Status dan control berfungsi untuk menyimpan perintah aritmatika yang baru saja dijalankan. Informasi ini berguna mengubah alur program

saat mengeksekusi operasi. Setiap byte flash memori memiliki alamat masing-masing. Alamat perintah yang akan dijalankan senantiasa disimpan program Counter. Ketika terjadi intrupsi atau pemanggilan rutin biasa, alamat diprogram rutin counter disimpan terlebih dahulu di stack. Alamat intrupsi atau rutin kemudian ditulis kedalam program Counter, instruksi kemudian di jemput dan dieksekusi. Ketika CPU telah selesai mengeksekusi rutin intrupsi atau rutin biasa, alamat yang ada distack dibaca dan ditulis kembali ke program counter. (Desmira dan Didik Ariwibowo, 2016)

1. Program Memori ATMEGA328 mempunyai 32 KiloByte flash memori untuk menyimpan program. Disebabkan ukuran instruksi 16 bit atau 32 bit. Maka flash memori dibuat berukuran 16K x 6. Artinya ada 16K alamat di flash memori yang dapat digunakan yang diawali dari alamat 0 heksa hingga alamat 3FFF heksa dan setiap alamatnya menyimpan 16 bit perintah.
2. SRAM Data Memori ATMEGA328 memiliki 2 KiloByte SRAM. Memori ini digunakan untuk menyimpan variabel. Tempat khusus di SRAM yang selalu ditunjukkan register SP disebut stack. Stack memiliki fungsi untuk menyimpan nilai yang dipush.
3. EEPROM Data Memori ATMEGA328 mempunyai 1 KiloByte data EEPROM. Walaupun catuan daya ke sistem data di EEPROM tidak akan hilang. Batasan sistem yang penting disimpan di EEPROM. Saat



sistem pertama kali menyala parameter tersebut akan terbaca oleh sistem dan diinisialisasi sesuai dengan nilai batasan tersebut.

4. Interupsi Sumber Interupsi ATmega328 ada 21 buah. Setiap interupsi, selalu memiliki Interrupt Service Routine (ISR), atau disebut juga Interrupt Handler. Yaitu rutin-rutin yang khusus dieksekusi sebagai fitur dari sebuah interupsi. Saat interupsi terjadi, CPU akan mulai menjalankan rutin ISR ini. Setiap Interupsi selalu mempunyai lokasi tetap dalam memory program yang disebut Interrupt Vector Table. Pada waktu interupsi diaktifkan dan interupsi terjadi maka CPU menunda interupsi sekarang dan melompat ke alamat rutin interupsi yang terjadi. Setelah selesai menjalankan perintah-perintah yang ada di alamat rutin interupsi CPU kembali melanjutkan interupsi yang sempat kembali tertunda.
5. I/O Port ATMEGA328 Memiliki pin I/O digital sebanyak 14 pin 6 diantaranya PWM (Pulse Width Modulation) output. Melalui pin I/O inilah ATMEGA328 berkomunikasi dengan sistem lain. Masing masing pin I/O dapat dikonfigurasi tanpa mempengaruhi fungsi pin I/O yang lain. Setiap pin I/O memiliki tiga register yakni: DDxn, PORTxn, dan PINxn. Kombinasi nilai DDxn dan PORTxn menentukan arah pin I/O.

6. *Clear Timer on Compare Match (CTC)* CTC adalah salah satu mode Timer/Counter. Pada CTC mode maka nilai TCNT1 menjadi 0 jika nilai TCNT1 telah sama dengan OCR1A atau ICR1.
7. *USART* Selain untuk general I/O, pin PD1 dan PD0 ATMEGA328 berfungsi untuk mengirim dan menerima bit secara serial. Perubahan fungsi ini dibuat dengan mengubah nilai beberapa register serial. Untuk menekankan fungsi ini, pin PD1 disebut TxD dan pin PD0 disebut RxD.

### **2.3 Mikrokontroler Arduino Uno R3**

Arduino adalah sebuah papan mikrokontroler berbasis ATmega328. Arduino memiliki 14 pin input/output dimana 6 pin dapat dipakai untuk output PWM, 6 analog input, crystal osilator 16 MHz, koneksi USB, jack power, kepala ICSP, dan tombol reset. Arduino dapat *men-support* mikrokontroler; dapat dikoneksikan dengan komputer menggunakan kabel USB. (Prio Handoko, 2017)

Arduino mempunyai kelebihan dibanding papan mikrokontroler yang lain selain bersifat terbuka, arduino juga mempunyai bahasa pemrogramannya sendiri yaitu bahasa C. Selain itu pada perangkat arduino ini sendiri telah tersedia *loader* yang berbentuk USB sehingga mempermudah kita saat memprogram mikrokontroler didalam arduino. Sedangkan pada kebanyakan papan mikrokontroler yang lain, masih memerlukan rangkaian *loader* yang berbeda atau terpisah untuk “menyuntikkan” program saat akan memprogram mikrokontroler. Port USB tersebut selain untuk *loader* ketika memprogram, dapat juga difungsikan sebagai port komunikasi serial. Pada arduino terdapat 20 pin I/O, yang terdiri dari

sebagai port komunikasi serial. Pada arduino terdapat 20 pin I/O, yang terdiri dari 6 pin input analog dan 14 pin digital input/output. Untuk 6 pin analog tersebut dapat juga difungsikan untuk output digital jika diperlukan output digital tambahan selain 14 pin yang telah tersedia. Kita cukup merubah konfigurasi pin pada program apabila ingin merubah pin analog menjadi digital. Dalam papan kita dapat melihat pin digital diberi penjelasan 0-13, lalu untuk menggunakan pin analog menjadi output digital, pin analog yang pada keterangan board 0-5 kita ubah menjadi pin 14-19 dengan kata lain pin analog 0-5 juga berfungsi sebagai pin output digital 14-16. Sifat *open source* arduino juga banyak memberikan manfaat tersendiri untuk kita dalam menggunakan perangkat ini, karena dengan sifat open source komponen yang kita gunakan bukan hanya tergantung pada satu *brand* tertentu, tapi juga memungkinkan bagi kita untuk dapat menggunakan komponen-komponen yang ada dipasaran. Bahasa pemrograman arduino adalah bahasa C yang sudah dipermudah syntax bahasa pemrogramannya sehingga mempermudah kita dalam mempelajari dan mendalami mikrokontroller. Deskripsi Arduio UNO :

**Tabel 2.4** Deskripsi Arduino

Mikrokontroller	ATmega 328
Tegangan Pengoperasian	5 V
Tegangan Input yang disarankan	7 – 12 V
Batas Tegangan Input	6 – 20 V
Jumlah Pin I/O Digital	14 Pin Digital (6 diantaranya menyediakan kekurangan PWM)
Jumlah Pin Input Analog	6 Pin
Arus DC tiap Pin I/O	40mA

Arus DC untuk Pin 3,3 V	50mA
<i>Memory Flash</i>	32 KB (ATmega 328) sekitar 0,5 KB digunakan oleh bootloader
SRAM	2 KB (ATmega 328)
EPROM	1 KB (ATmega 328)
<i>Clock Speed</i>	16 MHz

### 2.3.1 Power

Komponen arduino bisa diberikan tenaga melalui sambungan USB maupun *power supply*. *Power supply* juga bisa menggunakan adaptor DC atau baterai. Adaptor dapat dihubungkan dengan menggunakan *jack* adaptor pada port input *supply*. Arduino dapat dijalankan menggunakan *supply* dari luar sebesar 6 - 20 volt. Jika penyediaan tenaga tegangan kurang dari 7V, terkadang pin 5V akan menyuplai kurang dari 5 volt dan papan arduino ini bisa menjadi tidak stabil. Jika menggunakan lebih dari 12 V, tegangan di regulator dapat menjadi sangat panas dan menyebabkan kerusakan pada *board*. Rekomendasi tegangan ada pada 7 sampai 12 volt.

Penjelasan pada pin power adalah sebagai berikut :

a. Vin

Tegangan input ke papan arduino saat menggunakan tegangan dari luar (seperti yang disebutkan 5 volt dari koneksi USB atau tegangan yang diregulasikan). Pengguna dapat memberikan tegangan melalui vin ini, atau jika penyedia tegangan menggunakan *power jack*, maka jalurnya menggunakan pin ini.

b. 5V

Regulasi power *supply* digunakan untuk power mikrokontroller dan komponen lainnya pada papan. Tegangan 5V bisa melalui Vin menggunakan regulator pada perangkat arduino, atau disediakan oleh USB atau *supply* regulasi 5V lainnya.

c. 3,3 V

Suplai 3.3 volt berasal dari FTDI chip yang ada pada papan. Limit tekanan arusnya adalah 50mA

d. Pin Ground

Berfungsi sebagai jalur ground pada arduino.

### 2.3.2 Memori

Arduino Uno R3 ini memiliki 32 KB flash memori untuk penyimpanan kode, juga 2 KB yang digunakan untuk *bootloader*. Arduino mempunyai kapasitas 2 KB untuk RAM dan 1 KB untuk EEPROM.

### 2.3.3 Input & Output

Setiap 14 pin digital yang terdapat di arduino bisa dipakai untuk input atau output, menggunakan fungsi `pinMode()`, `digitalWrite()`, dan `digitalRead()`. Input/output dioperasikan pada tegangan 5 volt. Batas dari masing-masing pin untuk bisa menghasilkan atau menampung 40 mA dan memiliki internal pull-up resistor (disconnected oleh default) 20-50K Ohm.

Beberapa pin memiliki fungsi sebagai berikut :

- a. Serial : 0 (RX) dan 1 (TX). Berfungsi untuk menerima (RX) dan mengirim (TX) TTL data serial. Pin ini terhubung pada pin yang koresponding dari USB ke TTL chip serial.
- b. Interupt eksternal : 2 dan 3. Pin ini bisa diatur sedemikian rupa sebagai pemacu sebuah interup pada nilai rendah, menaikkan atau menurunkan nilai, ataupun perubahan nilai.
- c. PWM : 3, 5, 6, 9, 10, dan 11. Mendukung 8-bit output PWM dengan fungsi `analogWrite()`.
- d. SPI : 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Pin ini mendukung komunikasi SPI, yang mana masih mendukung hardware, yang tidak termasuk pada bahasa arduino.
- e. LED : 13. Ini dibuat sebagai penghubung LED ke digital pin 13. Ketika pin bernilai HIGH, LED hidup, ketika pin LOW, LED mati.

#### **2.3.4 Komunikasi**

Pada arduino uno ini terdapat sejumlah fitur sebagai tempat untuk berkomunikasi dengan komputer, Arduino lain, atau mikrokontroler lain. Perangkat ini juga menyediakan UART TTL (5V) komunikasi serial, yang terdapat pada pin digital 0 (RX) dan 1 (TX). *Firmware* Arduino menggunakan USB *driver* standar COM, dan tidak ada *driver* eksternal yang dibutuhkan. Namun, pada sistem operasi Windows, file Ini diperlukan. Perangkat lunak Arduino termasuk monitor serial yang memungkinkan data sederhana yang akan dikirim ke *papan* Arduino.

### 2.3.5 Software Arduino

Perangkat Arduino Uno ini bisa diprogram dengan menggunakan *Software* Arduino. Pada ATmega328 di Arduino terdapat *bootloader* yang memungkinkan kita untuk memasukkan kode baru tanpa menggunakan *programmer hardware eksternal*. IDE Arduino adalah *software* yang dibuat dengan sangat canggih dengan bahasa Java. IDE Arduino terdiri dari:

1. *Compiler*, sebuah modul yang mengubah kode program (bahasa Processing) menjadi kode biner. Bagaimanapun sebuah mikrokontroler tidak akan bisa memahami bahasa *Processing*. Yang bisa dipahami oleh mikrokontroler adalah kode biner. Hal tersebut mengapa *compiler* diperlukan.
2. Editor program, sebuah window yang memungkinkan pengguna menulis dan mengedit program dalam bahasa *Processing*.
3. *Uploader*, modul yang mengisi kode biner dari komputer ke dalam memori ke dalam papan Arduino. Sebuah kode program Arduino umumnya disebut dengan istilah *sketch*. Kata "*sketch*" digunakan secara bergantian dengan "kode program" dimana keduanya mempunyai arti yang sama.

### 2.4 Bahasa Pemrograman Arduino

Seperti yang telah dijabarkan diatas, program Arduino sendiri menggunakan bahasa C. walaupun terdapat banyak bahasa pemrograman tingkat tinggi (*high level language*) seperti pascal, basic, cobol, dan lainnya. Meskipun demikian, sebagian

besar dari para Bahasa C programmer profesional masih tetap memilih bahasa C sebagai bahasa yang lebih unggul, berikut alasan-alasannya :

Bahasa C adalah Bahasa yang kuat dan *fleksibel* yang sudah terbukti mampu menyelesaikan program program besar seperti pembuatan sistem operasi, pengolahan gambar (seperti pembuatan game) dan juga pembuatan kompilator bahasa pemrograman baru. Bahasa C merupakan bahasa yang *portabel* sehingga dapat dijalankan di beberapa system operasi yang berbeda. Sebagai contoh program yang kita tulis dalam sistem operasi windows dapat kita kompilasi didalam sistem operasi linux dengan sedikit ataupun tanpa perubahan sama sekali. (Rachmat Adi Purnama dan Adi Tri Laksono Putra, 2018)

Bahasa C merupakan bahasa yang sangat populer dan banyak digunakan oleh programmer berpengalaman sehingga kemungkinan besar *library* pemrograman telah banyak disediakan oleh pihak luar/lain dan dapat diperoleh dengan mudah.

Bahasa C juga merupakan bahasa yang bersifat modular, yaitu terdiri dari rutin rutin tertentu yang disebut dengan fungsi (*function*) dan fungsi-fungsi itu dapat digunakan kembali untuk pembuatan program-program yang lain tanpa harus menulis ulang implementasinya. Meskipun termasuk *general-purpose programming language*, yakni bahasa pemrograman yang bisa membuat berbagai aplikasi, bahasa pemrograman C paling cocok merancang aplikasi yang berhubungan langsung dengan Sistem Operasi dan hardware. Ini tidak terlepas dari tujuan awal bahasa C dikembangkan.



Struktur penulisan program dalam bahasa C harus memiliki fungsi utama, yang bernama `main()`. Fungsi inilah yang akan dipanggil pertama kali pada saat proses eksekusi program. Oleh karena itu bahasa C merupakan bahasa prosedural yang menerapkan konsep runtutan (program dijalankan per baris dari atas ke bawah secara berurutan), maka apabila kita menuliskan fungsi-fungsi lain tersebut dibawah fungsi utama, maka kita harus menuliskan bagian prototipe (*prototype*), aturan ini ditujukan untuk mengenalkan terlebih dahulu kepada kompiler daftar fungsi yang akan digunakan di dalam program. Tetapi apabila kita menuliskan fungsi-fungsi lain tersebut diatas atau sebelum fungsi utama, maka kita tidak perlu lagi untuk menuliskan bagian prototype diatas. (Eka W Fridayanthie dan Jimmy Charter, 2016)

*File header* adalah *file library* dengan ekstensi *h* (*\*.h*) yang berisi deklarasi fungsi C dan definisi makro yang akan digunakan di dalam sebuah program. Ada 2 jenis *file header* yang ada pada bahasa c, yaitu *file header* yang telah disediakan oleh *compiler* dan *file header* yang kita buat sendiri. Jika kita menggunakan *file header* yang sudah disediakan oleh *compiler*, maka dalam penulisannya harus di dalam tanda `<>` (contoh `<stdio.h>`).

Sebagai catatan penting bahwa apabila kita menggunakan *file header* yang telah disediakan oleh kompilator, maka kita harus menuliskannya didalam tanda „<“ dan „>“ (misalnya `<stdio.h>`). Namun apabila menggunakan *file header* yang kita buat sendiri, maka file tersebut ditulis diantara tanda “ dan ” (misalnya “*cobaheader.h*”). perbedaan antara keduanya terletak pada saat pencarian file

tersebut. Apabila kita menggunakan tanda  $\langle \rangle$ , maka file tersebut dianggap berada pada direktori default yang telah ditentukan oleh kompilator. Sedangkan apabila kita menggunakan tanda `""`, maka *file header* dapat kita tentukan sendiri lokasinya.

*File header* yang akan kita gunakan harus kita daftarkan dengan menggunakan directive `#include`. Directive `#include` ini berfungsi untuk memandu kompilator bahwa program yang telah kita buat akan menggunakan file-file yang didaftarkan. Berikut ini contoh penggunaan directive `#include`.

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

```
#include<stdlib.h>
```

```
#include"myheader.h"
```

Apabila kita akan menggunakan fungsi tertentu yang akan disimpan dalam sebuah *file header*, maka kita juga harus mendaftarkan *file headernya* dengan menggunakan directive `#include`. Sebagai contoh, apabila kita menggunakan fungsi `getch()` dalam program, maka kita harus mendaftarkan *file header*`<conio.h>`.

#### **2.4.1 Reset Otomatis (Software)**

Terlepas dari konsentrasi terhadap penggunaan fisik dari tombol reset sebelum melakukan penguploadan, perangkat ini didesain dengan cara yang memungkinkannya untuk direset menggunakan software yang sedang berjalan pada komputer ataupun yang sedang terhubung. Salah satu garis kontrol aliran hardware (DTR) dari ATmega8U2/16U2 dihubungkan pada garis reset dari ATmega328 melalui sebuah kapasitor 100 nanofarad. Jadi ketika saluran ini

dipaksakan, garis reset jatuh cukup panjang untuk mereset chip.

Software Arduino menggunakan *tools* ini untuk memungkinkan kita mengupload kode dengan mudah dengan menekan tombol upload di software Arduino. Ini berarti bahwa *bootloader* bisa mempunyai sebuah rentang waktu yang lebih singkat, sebagai penurunan dari DTR yang dapat menjadi koordinasi yang baik dengan memulai penguploadan. Pengaturan ini tentunya mempunyai dampak tertentu. Pada saat perangkat Arduino Uno dikoneksikan pada sebuah komputer lain yang sedang berjalan dengan menggunakan sistem operasi seperti Mac X atau Linux, Arduino Uno akan mengatur ulang setiap kali sebuah koneksi yang dibuat dari software (melalui USB). Lalu selanjutnya, setengah-detik atau lebih, *bootloader* akan berjalan pada perangkat Arduino UNO. Saat Arduino UNO diprogram untuk mengabaikan data yang cacat/salah (contohnya apa saja selain sebuah penguploadan kode baru) untuk menahan beberapa bit pertama dari data yang dikirim ke board setelah sebuah koneksi dibuka.

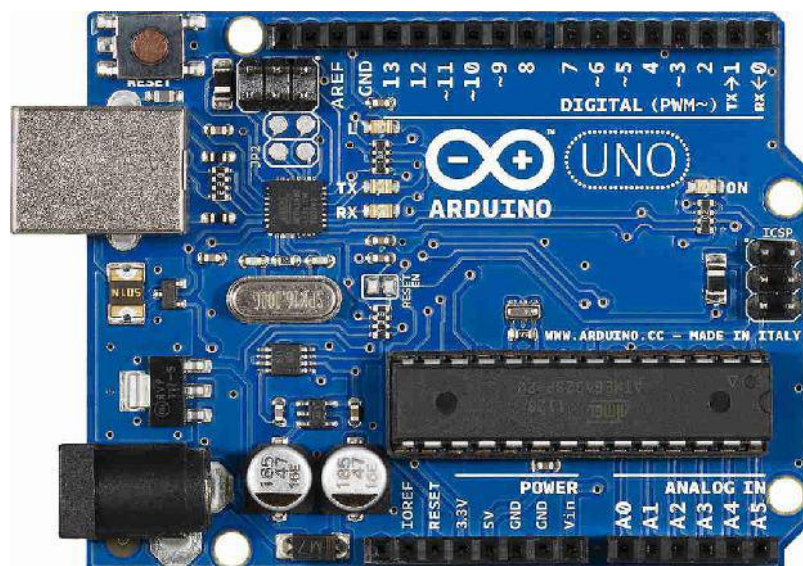
Apabila sebuah sketch sedang berjalan pada perangkat ini dan menerima satu kali konfigurasi atau data lain ketika sketch pertama mulai, maka perangkat akan memastikan bahwa software yang berkomunikasi menunggu satu detik setelah membuka koneksi dan sebelum mengirim data ini. Arduino Uno berisikan sebuah “*history*” yang dapat dihapus untuk mencegah reset otomatis. Pada salah satu sisi dari jejak dapat disolder bersama untuk mengaktifkan kembali. Pad itu diberi label “RESET-RN”. Kita juga bisa menonaktifkan reset otomatis dengan cara menghubungkan sebuah resistor 110 ohm dari tegangan 5V ke garis reset.

## 2.4.2 Proteksi Arus lebih USB

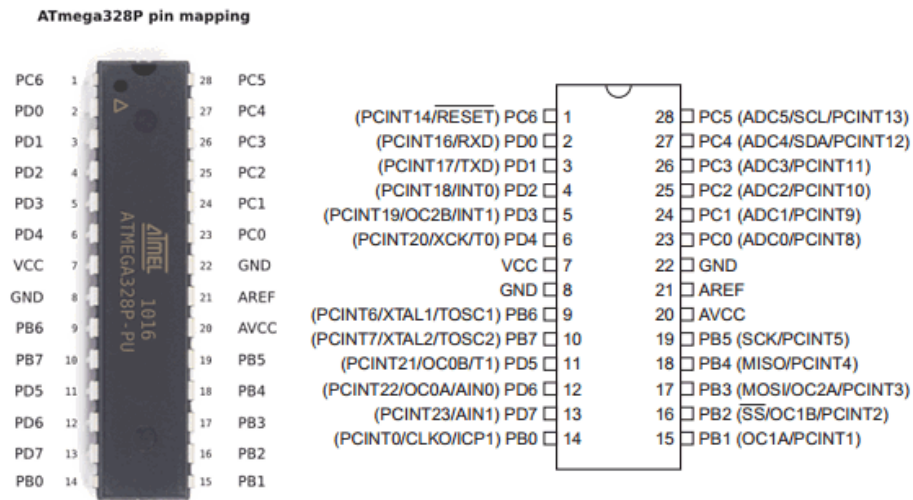
Arduino Uno memiliki sebuah sekering reset yang akan melindungi port USB komputer dari hubungan pendek ataupun arus yang berlebihan. Meskipun sebagian besar komputer menyediakan proteksi internal sendiri, sekering menyediakan sebuah pengamanan tambahan. Jika lebih dari 500 mA diterima port USB, sekering secara otomatis akan memutuskan koneksi sampai hubungan pendek atau kelebihan beban hilang.

## 2.4.3 Karakteristik Fisik

Panjang dan lebar maksimum dari PCB Arduino UNO masing-masingnya adalah 2.7 dan 2.1 inci, dengan konektor USB dan power jack yang memperluas dimensinya. Empat lubang sekrup memungkinkan board untuk dipasangkan ke sebuah permukaan atau kotak. Sebagai catatan, bahwa jarak antara pin digital 7 dan 8 adalah 160 mil.(0.16"), bukan sebuah kelipatan genap dari jarak 100 mil dari pin lainnya.



**Gambar 2.3** Arduino Uno R3



**Gambar 2.4** Konfigurasi Pin Arduino (ATmega 328p)

Sumber : ( Rian Affrilianto, dkk 2017)

## 2.5 SIM800L

Module SIM800L ini merupakan jenis module GSM/GPRS Serial yang paling banyak digunakan oleh para pegiat elektronika, maupun profesional elektronika yang diimplementasikan dalam berbagai aplikasi pengendalian jarak jauh via Telepon dengan simcard jenis Micro sim. Saat ini, terdapat beberapa jenis dari Breakout Board, namun yang paling banyak didistribusikan di Indonesia ialah versi mini dengan kartu GSM jenis Micro SIM. ( Rian Affrilianto, dkk 2017)



**Gambar 2.5** SIM800L

#### Keterangan PinOut :

1. VCC : tegangan masukan 3.7 – 4.2Vdc
2. RST : Reset
3. RX : Rx Data Serial
4. TX : Tx Data Serial
5. ANT : Antena
6. Micro Sim (Kartu GSM)
7. RING : ketika ada telp masuk
8. DTR
9. GND : Ground
10. MIC + : ke microphone kutub +
11. MIC – : ke microphone kutub –
12. Speaker + : ke speaker atau amplifier kutub +
13. Speaker – : ke speaker atau amplifier kutub –

#### **2.5.1 Spesifikasi Modul SIM800L**

1. Ic Chip : SIM800
2. Tegangan ke VCC : antara 3.7 – 4.2Vdc (namun pada datasheet = 3.4 – 4.4V), dan dianjurkan menggunakan 3.7 Vdc supaya tidak terdapat notifikasi.
3. Menangkap frequency jaringan GSM yaitu QuadBand (850/900/1800/1900Mhz)

4. Konektifitas class 1 (1W) pada DCS 1800 dan PCS 1900GPRS, sedangkan pada class 4 (2W) pada GSM 850 dan EGSM 900
5. GPRS multi-slot class 1~12 (option) tetapi default pada class 12
6. Suhu pengoperasian normal : 40°C ~ +85°C
7. Menggunakan port TTL serial port, sehingga dapat langsung diakses menggunakan microcontroller tanpa perlu memerlukan MAX232
8. Transmitting power
9. Power module automatically boot, homing network
10. Tersedia Led pada modul yang berfungsi sebagai indikator. Apabila pada module terhubung dengan jaringan GSM maka LED akan berkedip perlahan, akan tetapi apabila tidak ada sinyal maka LED akan berkedip cepat.
11. Ukuran module : 2.5cm x 2.3cm

## **2.6 Mini LM2596**

Modul Step-Down Voltage Regulator / DC Buck Converter Module ini adalah modul yang sangat mudah untuk dipakai dalam mengkonversi / menurunkan tegangan dari power supply menjadi tegangan keluaran yang lebih rendah. Modul elektronika ini menggunakan *Integrated Circuit* / IC LM2596, 3A Step-Down Voltage Regulator. Chip LM2596 bekerja pada *switching frequency* 150 kHz, memungkinkan komponen penyangkutan berukuran lebih kecil dibanding komponen penyangkutan yang biasa dibutuhkan oleh switching regulator berfrekuensi rendah. Produsen IC ini menjamin toleransi perbedaan tegangan keluaran hanya  $\pm 4\%$  pada

tegangan masukan dan kondisi beban keluaran sesuai spesifikasi, dan  $\pm 15\%$  toleransi pada frekuensi osilator. IC ini dapat distatiskan secara eksternal, dengan konsumsi daya hanya sebesar  $80\mu\text{A}$  pada moda siaga. Fitur proteksi termasuk pembatas arus pengurang frekuensi dua tahap (*two stage frequency reducing current limit*) untuk *output switch* dan fitur mematikan chip secara otomatis pada kondisi kelebihan panas (*over temperature*). (Marco Failla and Gian Paolo Vitale, 2019)



**Gambar 2.6** Mini LM2596

Pada modul ini telah dilengkapi dengan *voltmeter* (pengukur tegangan) pada pin keluaran, sehingga pengaturan tegangan keluaran yang dibutuhkan menjadi mudah dan akurat tanpa harus diukur oleh pengukur tegangan eksternal. Nilai tegangan keluaran ditampilkan pada LED numerik (tinggi tampilan numerik 0,36") yang terpasang di papan modul ini. Besar tegangan yang diinginkan bisa diatur dengan memutar *mutli-turn potentiometer* (komponen potensiometer yang berwarna biru dengan sekrup kuning) di papan *electronic module* ini.



## 2.7 Power Supply

Power Supply atau dalam bahasa Indonesia disebut dengan Catu Daya adalah suatu komponen listrik yang berfungsi sebagai penyedia energi listrik untuk diteruskan pada perangkat listrik ataupun elektronika lainnya. Power Supply atau catu daya ini sebenarnya memerlukan sumber energi listrik, yang nantinya akan diteruskan menjadi energi listrik yang dibutuhkan oleh perangkat elektronika lainnya. Oleh sebab itu, Power Supply bisa disebut juga dengan istilah Electric Power Converter.



**Gambar 2.7** Power Supply

### 2.7.1 Klasifikasi Umum Power Supply

Pada umumnya Power Supply dapat diklasifikasikan menjadi 3 kelompok besar, yakni berdasarkan Fungsinya, berdasarkan Bentuk Mekanikalnya dan juga berdasarkan Metode Konversinya. Berikut ini merupakan penjelasan singkat mengenai ketiga kelompok tersebut :

## 1. Power Supply Berdasarkan Fungsi (Functional)

Berdasarkan kegunaannya, Power supply dapat dibedakan menjadi regulated Power Supply, Unregulated Power Supply dan Adjustable Power Supply.

- a. **Regulated Power Supply** adalah catu daya yang dapat menjaga kestabilan tegangan dan arus listrik walaupun terdapat perubahan atau dinamika pada beban atau sumber listrik (Tegangan dan Arus Input).
- b. **Unregulated Power Supply** adalah Power Supply yang tegangan ataupun arusnya dapat berubah saat beban berubah atau sumber listriknya mengalami perubahan.
- c. **Adjustable Power Supply** adalah Power Supply yang tegangan atau Arusnya dapat dikontrol sesuai kebutuhan dengan menggunakan Knob Mekanik. Terdapat 2 jenis Adjustable Power Supply yaitu Regulated Adjustable Power Supply dan Unregulated Adjustable Power Supply.

## 2. Power Supply Berdasarkan Bentuknya

Pada peralatan Elektronika pada umumnya seperti TV, Monitor, Komputer Desktop maupun DVD Player, Power Supply biasanya ditanamkan di dalam atau menyatu bersama perangkat-perangkat tersebut sehingga kita sebagai konsumen tidak dapat melihatnya secara langsung. Jadi hanya sebuah kabel listrik yang dapat kita lihat dari luar.

Power Supply ini disebut dengan Power Supply Internal (Built in). Namun ada juga Power Supply yang berdiri sendiri (stand alone) dan berada diluar perangkat elektronika yang kita gunakan seperti Charger Handphone dan Adaptor Laptop. Ada juga Power Supply stand alone yang bentuknya besar dan dapat disetel tegangannya sesuai dengan kebutuhan kita.

### 3. Power Supply Berdasarkan Metode Konversinya

Berdasarkan Metode Konversinya, Power supply dapat dibedakan menjadi Power Supply Linier yang mengkonversi tegangan listrik secara langsung dari Inputnya dan Power Supply Switching yang harus mengkonversi tegangan input ke pulsa AC atau DC terlebih dahulu.

(Suwitno, 2016)

## **2.8 Android**

### **2.8.1 Sejarah Android**

Android adalah sistem operasi yang berbasis Linux dan dibuat untuk perangkat seluler layar sentuh seperti smartphone serta computer tablet. Android pada awalnya dikembangkan oleh perusahaan bernama Android, Inc., dengan dukungan keuangan yang berasal dari perusahaan Google, yang kemudian Google pun pada akhirnya membelinya pada tahun 2005. Sistem operasi android ini secara resmi dirilis pada tahun 2007, bersamaan dengan dibangunnya sebuah perusahaan Open Handset Alliance, penggabungan dari beberapa perusahaan-perusahaan

perangkat keras, perangkat lunak, serta telekomunikasi yang memiliki tujuan untuk memajukan standar terbuka dari perangkat seluler. Ponsel yang berbasis sistem operasi Android pertama dijual pada bulan Oktober 2008.

Sejarah android pada mulanya berasal dari perusahaan bernama Android, Inc. didirikan tempatnya di Palo Alto, California, pada Oktober tahun 2003 oleh Andy Rubin (pendiri Danger), Rich Miner seorang pendiri Wildfire Communications, Inc., Nick Sears seorang mantan VP T-Mobile, dan Chris White seorang kepala desain dan pengembangan interface WebTV untuk mengembangkan sebuah "perangkat seluler pintar yang lebih sadar tentang lokasi dan preferensi penggunanya".



**Gambar 2.8** Aplikasi Android

Awalnya maksud dan tujuan dari perkembangan itu ialah di khususnya untuk kamera digital, tetapi mereka menyadari bahwa pasar dari kamera digital tidak terlalu memenuhi keinginan pasar internasional dan dengan banyak

pertimbangan bahwa perkembangan tersebut tidak besar potensinya, pada akhirnya pengembangan Android ini lalu dialihkan pada pasar telepon pintar atau smartphone untuk menjadi kompetitor dari perusahaan Symbian serta Windows Mobile (iPhone Apple pada saat itu belum dirilis). Kumpulan dari pengembang Android tersebut adalah para pakar teknologi yang telah lama berkecimpung dalam usaha teknologi, namun mereka tetap menjalankan Android Inc ini secara tertutup, dan mereka hanya memberitahu kepada media bahwa mereka sebagai pengembang hanya sedang berusaha menciptakan sebuah perangkat lunak untuk telepon genggam. Ditahun yang sama pada saat pengembangan berjalan, Andy Rubin kehabisan uang. Steve Perlman yang sejatinya adalah teman dekat Andy Rubin datang dan meminjaminya \$10.000 tunai dan menolak tawaran saham di perusahaan.

Pada tahun 2005, Google membeli perusahaan Android Inc. dan menjadikan perusahaan Android Inc tersebut sebagai anak perusahaan yang dimiliki oleh Google. Pendiri Android Inc. yaitu Rubin, Miner, serta White tetap bekerja pada perusahaan tersebut setelah diakuisisi oleh Google mengingat merekalah pelopor dari perusahaan tersebut. Saat perusahaan tersebut telah menjadi anak dari perusahaan Google, tim yang dipimpin oleh pendiri Android, Andy Rubin menggunakan Kernel Linux sebagai sarana untuk mulai bekerja mengembangkan sebuah platform perangkat seluler. Lalu di mulai pada tahun 2008, Android mulai secara bertahap melakukan sejumlah pembaruan atau update untuk meningkatkan kinerja dari system operasi tersebut dengan menambahkan fitur baru, memperbaiki

bug pada versi android yang sebelumnya. Mereka memiliki ide bagaimana memberikan nama pada setiap hasil rilis dengan nama sebuah makanan pencuci mulut yang diurutkan secara alfabetis agar lebih mudah diingat dan lebih menarik penggunaanya kelak, seperti cupcake, donut, dll.

### **2.8.2 Kelebihan atau Keunggulan Android**

Sebagai pengguna Android, penulis yakin bahwa hampir semua pengguna Android ini mengetahui apa saja keunggulan dari Android, tapi tidak ada salahnya jika saya bahas disini.

1. Mudah untuk digunakan – Seperti pada sistem operasi Windows yang memiliki fitur fitur yang sangat mudah untuk digunakan, Android juga di desain agar pengguna dapat dengan mudah mengoperasikan atau menjalankan aplikasi ini, hanya dalam hitungan hari atau bahkan hitungan jam, pengguna dapat memahami pengoperasian sistem pada Android yang berjalan pada Smartphone.
2. Pemberitahuan – Kita sebagai pengguna dapat mengatur atau mengontrol pemberitahuan pada E-mail, SMS, telephon, pembaharuan perangkat dan lain sebagainya pada smartphone untuk mendapatkan pemberitahuan dengan sangat mudah dan cepat.
3. Interface – Android masih bisa bersaing dengan iOS yang dimiliki oleh Apple dari segi tampilan, karena Android memang dari awal mengusung teknologi dari iOS, Android berusaha membangun

interface yang tidak kalah bahkan menyaingi iOS namun dengan versi yang lebih murah.

4. Open Source – Dikarenakan berbasis Kernel Linux, maka sistem operasi ini memang sengaja dibuat open source oleh penciptanya. Jadi kita akan menemui ada banyak sekali Custom Rom yang telah dibuat oleh ahli-ahli modifikasi sistem di luar sana untuk masing-masing perangkat android.
5. Aplikasi – Kita akan menemui juga jutaan aplikasi yang menarik dan bisa kita gunakan sesuai keinginan dan kebutuhan kita. Mulai dari aplikasi yang gratis hingga berbayar, dan itu semua dapat kita cari dan download di aplikasi GooglePlay yang memang wadah dari seluruh pilihan aplikasi yang disuguhkan kepada pengguna, yang tentunya milik dari perusahaan Google.
6. Dan masih banyak lagi kelebihan dan fitur yang dimiliki Android, dan anda bisa mengembangkannya sesuai dengan keinginan anda, karena Sistem operasi ini Open Source dan User Friendly.

### **2.8.3 Kelemahan atau Kekurangan Android**

Segala sesuatunya tidak ada yang sempurna, bahkan pada aplikasi Android ini pun masih terdapat beberapa kelemahan ataupun kekurangan. Apa saja itu?? Beberapa hal yang paling sering kita alami saat mengoperasikan android.

1. Grafis dan animasi – Hardware acceleration adalah proses dimana smartphone bisa berganti antara GPU (grafis) dan CPU tanpa

menggunakan memori smartphone dalam jumlah besar. Hal ini sudah ada pada perangkat Android sejak versi awal, namun terbatas hanya pada beberapa fitur, sedangkan iPhone sudah menggunakan hardware acceleration sejak awal, sehingga iPhone menciptakan *user experience* yang lebih menyenangkan.

2. **Batterai Cepat Habis** – Ini sering terjadi jika anda menyalakan paket data dan menggunakan widget serta aplikasi yang berjalan secara berlebihan, untuk mengatasinya anda harus mengurangi aktivitas aplikasi pada smartphone anda.
3. **Malware/virus** – Karena Android adalah open source dan pengguna dapat menginstall aplikasi yang bukan berasal dari Google Play, perangkat Android lebih rentan terkena malware dan aplikasi bajakan/palsu.

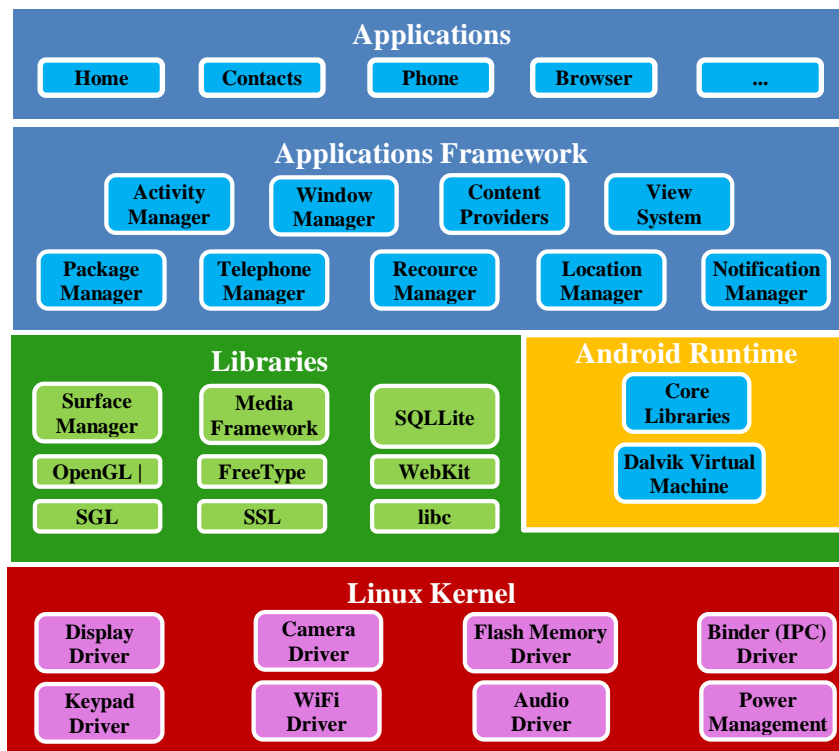
#### **2.8.4 Fitur dan Arsitektur Android**

Fitur yang tersedia di Android adalah:

1. **Kerangka aplikasi:** memungkinkan penggunaan dan pemindahan komponen yang tersedia.
2. **Dalvik mesin virtual:** mesin virtual dimaksimalkan untuk perangkat telepon.
3. **Grafik:** grafik di 2D dan grafis 3D berdasarkan pustaka OpenGL.
4. **SQLite:** untuk penyimpanan data.



5. Mendukung media: audio, video, dan berbagai format gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF)
6. GSM, Bluetooth, EDGE, 3G, dan WiFi (hardware dependent)
7. Kamera, Global Positioning System (GPS), kompas, dan accelerometer (tergantung hardware)
8. Lingkungan pengembangan yang kaya termasuk Emulator, peralatan debugging, dan plugin untuk Eclipse IDE.



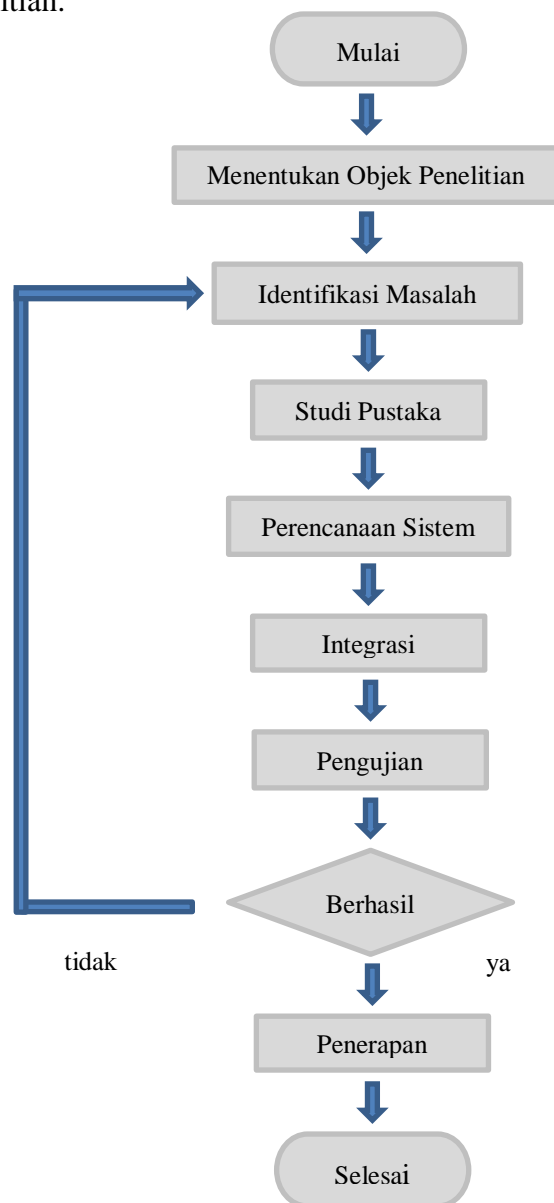
**Gambar 2.9** Fitur dan Arsitektur Android

## BAB III

### PERANCANGAN DAN PEMBUATAN

#### 3.1 Tahapan Penelitian

Berikut ini adalah diagram alur yang digunakan sebagai acuan dalam pelaksanaan penelitian.



**Gambar 3.1** Diagram Alur Penelitian

### 3.2 Metode Pengumpulan Data

Pada tahap pengumpulan data, penulis melakukan pengumpulan data yang diperlukan dalam merancang *hardware* maupun *software* pembuatan Rancangan Running Text dengan Update Data melalui SMS berbasis Arduino. Adapun metode pengumpulan data yang akan dilakukan adalah sebagai berikut :

1. Studi Literatur

Studi literatur dilakukan dengan cara menganalisis, mengutip dan membuat penjelasan yang bersumber pada bahan-bahan pustaka yang mendukung dan berkaitan dengan *running text*. Selanjutnya dengan cara mempelajari dan memahami sistem yang berhubungan dengan masalah yang akan dibahas dalam penulisan ini.

2. Penelitian Lapangan

Penelitian lapangan adalah teknik yang dilakukan penulis guna mengumpulkan dan menggabungkan data agar dapat lebih objektif dalam melakukan perancangan alat. Dalam penelitian ini penulis menggunakan beberapa metode dalam pengumpulan data sebagai berikut:

- a. Metode observasi

Pengumpulan data yang dilakukan secara langsung dengan melihat secara fisik alat maupun komponen yang digunakan dan dibutuhkan pada *running text*.

b. Wawancara

Melakukan interview secara langsung dengan penjual komponen, pembuat alat dan pengguna alat untuk memperoleh data tentang *running text*.

### 3.3 Analisis Sistem Sedang Berjalan

Dengan pesatnya perkembangan teknologi saat ini, maka kebutuhan pasar berorientasi pada produk yang berkualitas, hemat energi, ekonomis dan menarik, produk seperti inilah yang bisa dimanfaatkan untuk kebutuhan perusahaan. Maka *Running Text* dengan *Update Data* Melalui SMS berbasis Arduino ini adalah produk yang tepat untuk digunakan sebagai penyampaian informasi kepada publik.

Dalam bidang teknik, pembuatan atau perancangan produk harus dilengkapi dengan penjelasan mengenai bahan-bahan yang digunakan untuk setiap komponen pada produk tersebut, baik dari ukuran, alat yang digunakan untuk pengerjaan, serta prosedur kerja.

#### 3.3.1 Identifikasi Kebutuhan

Rancangan *Running text* dengan *update* data melalui SMS berbasis Arduino ini terdiri dari 2 bagian yaitu :

1. *Hardware* (Perangkat Keras)
  - a. Sistem mikrokontroler Arduino Uno sebagai Sistem pengolah *input/ output*.
  - b. SIM800L berfungsi untuk mengirim sms, menerima sms, transfer data melalui gprs dan fungsi dtmf.

- c. Mini LM 2596 atau *stepdown* berfungsi untuk mengubah tingkatan tegangan (*voltage level*) arus.
- d. Panel Matrik atau Display P10 berfungsi untuk menampilkan text counter.
- e. Power supply atau Catu daya berfungsi untuk mengubah tegangan arus ac menjadi dc.

## 2. *Software* (Perangkat Lunak)

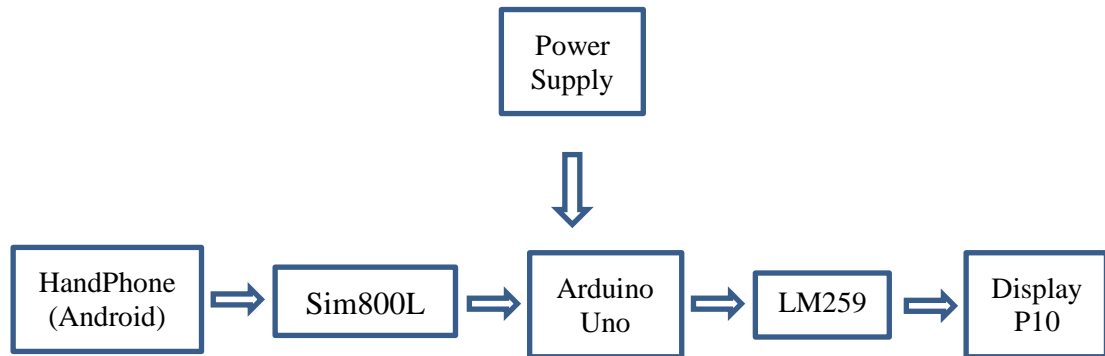
### a. *Software* Arduino Uno

*Software* Arduino Uno yang digunakan adalah IDE Arduino (*Integrated Development Environment*) yang berfungsi untuk menulis program, meng *compile* menjadi kode biner dan meng *upload* kedalam *memory microcontroller*.

### 3.3.2 Desain Perangkat Keras

Pada proses pembuatan ini, komponen utama yang digunakan untuk mengontrol maupun mengatur komponen lainnya seperti: Sim800l, LM2596, display p10, dan power supply adalah Mikro Kontroler Arduino Uno. Mikrokontroler Arduino dirancang agar dapat menerima masukan dari Sim800l, kemudian menentukan pergerakan LM2596 hingga fungsi display p10 untuk menampilkan hasil dari *running text*.

Alat dan bahan yang digunakan dalam penelitian, secara umum didesain seperti blok diagram dibawah ini :



**Gambar 3.2** Diagram Blok Sistem

### 3.4 Alat dan Bahan

#### A. Alat yang digunakan dalam penelitian ini

1. Tool Box dan perlengkapan lain yang mendukung.

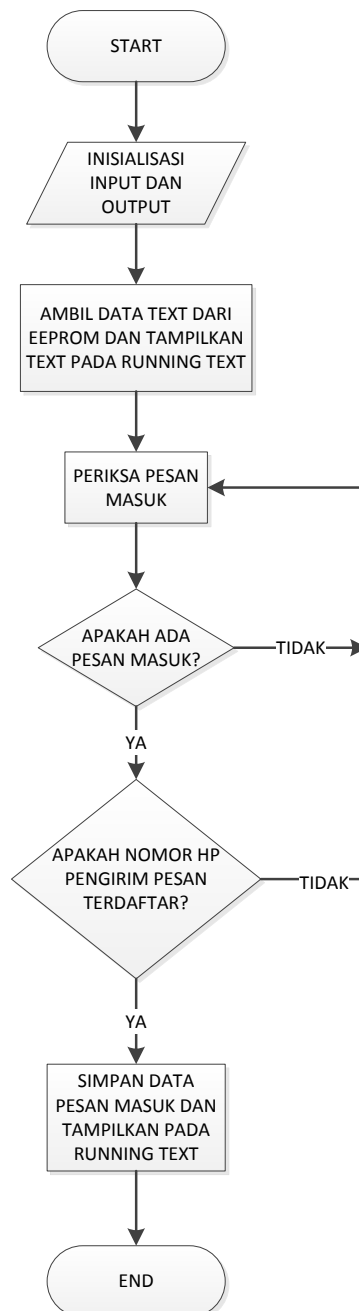
#### B. Bahan yang Digunakan

1. Arduino Uno berfungsi sebagai mikrokontroler dari display p10.
2. SIM800L berfungsi untuk mengirim sms, menerima transfer data melalui gprs dan fungsi dtmf.
3. Mini LM 2596 berfungsi untuk mengubah tingkatan tegangan (*voltage level*) arus.
4. Display P10 berfungsi untuk menampilkan text.
5. Power supply berfungsi untuk menyuplai arus dan mengubah tegangan arus ac menjadi dc.

### 3.5 Spesifikasi Sistem

Dalam perancangan running text mikrokontroler Arduino Uno digunakan sebagai komponen utama yang mengatur komponen lainnya seperti: sim 800l, lm

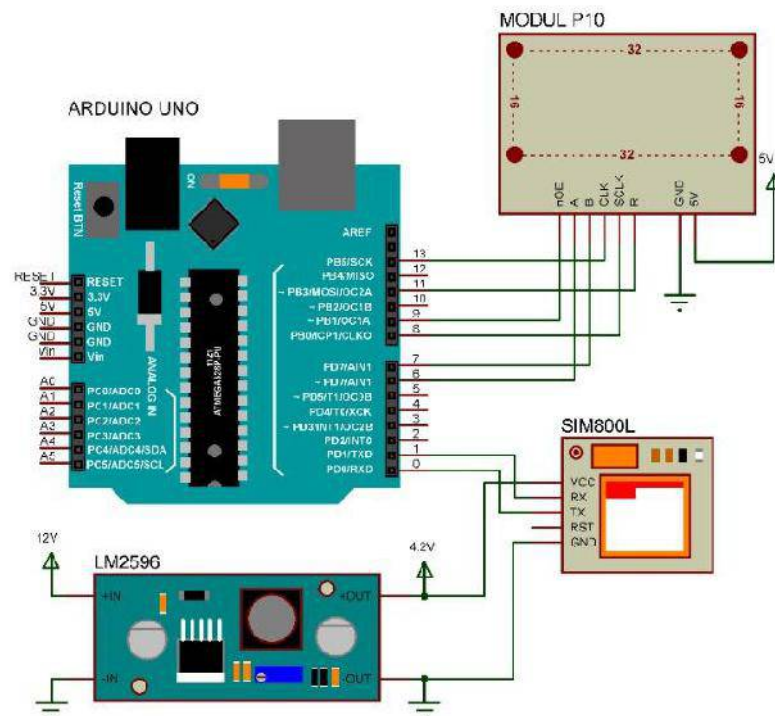
2596, display p10, dan power supply dirancang agar mikrokontroler arduino dapat menerima masukan dari sim 8001, kemudian menentukan pergerakan lm 2596 hingga fungsi display p10 untuk menampilkan hasil dari running text.



**Gambar 3.3** Flow Chart Sistem

### 3.6 Rangkaian Keseluruhan Sistem

Rangkaian keseluruhan sistem dari running text dibagi menjadi 4 bagian yaitu : sim800l, lm2596, Arduino Uno dan power supply.



Gambar 3.4 Rangkaian Sistem

### 3.7 Software Pemograman

#### 3.7.1 Software Arduino IDE

Integrated Development Environment (IDE), adalah aplikasi lintas platform (untuk windows, macOS, Linux) yang ditulis dalam bahasa C dan C++. Aplikasi ini digunakan untuk menulis dan mengunggah program pada papan Arduino yang kompatibel, tapi juga dengan bantuan partisi ke tiga , maupun pengembang mikro kontroler yang lain. Arduino IDE memuat daftar pustaka perangkat lunak dari mulai



proyek pengkabelan yang menyediakan banyak sekali prosedur penginputan dan output yang umum. Kode yang digunakan pengguna hanya memerlukan dua fungsi dasar yaitu untuk memulai sketsa dan pengulangan program utama yang akan dikompilasi dan dihubungkan dengan program sub main() kedalam program eksekutif yang dapat dieksekusi dengan GNU. Bahasa pemrograman Arduino (Sketch) sudah dilakukan perubahan untuk memudahkan pemula dalam melakukan pemrograman dari bahasa aslinya. IC mikrokontroler Arduino telah ditanamkan suatu program bernama Bootlader yang berfungsi sebagai penengah antara compiler Arduino dengan mikrokontroler sebelum dijual ke pasaran.

Program yang telah ditulis menggunakan Arduino Software (IDE) disebut sebagai *sketch*. Sketch ditulis dalam suatu lembar kerja editor dan disimpan dalam file dengan ekstensi *.ino*. Fitur-fitur yang ada pada Teks editor Arduino Software adalah *cutting* atau paste dan *searching* atau *replacing*. Hali ini guna memudahkan dalam menulis kode program.

Pada Software Arduino IDE, tersedia *message box* berwarna hitam yang berguna menampilkan status, seperti pesan *error*, *compile*, dan *upload* program. Lalu pada bagian bawah paling kanan Software Arduino IDE, menampilkan papan yang telah terkonfigurasi beserta COM Ports yang sedang digunakan.

### **3.7.2 Program Running Text**

Persiapan pertama sebelum memasukkan program adalah mengkoneksikan mikrokontroller arduino dengan PC menggunakan USB port. Pastikan lampu pada mikrokontroler menyala atau berkedip menandakan bahwa sudah terjadi hubungan

elektronik. Selanjutnya adalah membuka software arduino , kemudian penulisan program pada software , berikut ini adalah program yang di tuliskan pada software.

```
#include <SPI.h>

#include <DMD.h>

#include <TimerOne.h>

#include "Arial_black_16.h"

#include <EEPROM.h>

const String LockNumber = "81288727***";

int SPEED = 15;

String TEKS = "Universitas Pembangunan Panca Budi Medan";

boolean stringComplete = false;

String inputString = "";

String psnSMS;

bool dataValid;

String dtRunningTeks;

#define DISPLAYS_ACROSS 3

#define DISPLAYS_DOWN 1

DMD dmd(DISPLAYS_ACROSS, DISPLAYS_DOWN);

void ScanDMD()

{

    dmd.scanDisplayBySPI();

}
```

```

void setup() {
    Serial.begin(9600);

    delay(10000); // tunggu 10 detik

    Serial.println("AT\r\n");

    delay(1000);

    Serial.write("AT+CNMI=3,2,0,0,0\r\n");

    delay(1000);

    Serial.write("AT+CMGF=1\r\n");

    delay(1000);

    while (Serial.available()) {
        char inChar = (char)Serial.read();
    }

    Timer1.initialize( 1000 );

    Timer1.attachInterrupt( ScanDMD );

    delay(1000);

    dmd.clearScreen( true );

    //eepSaveString(0,TEKS);

    TEKS = eepLoadString(0,100);

}

void loop() {

    char dtTeks[TEKS.length()+1];

    TEKS.toCharArray(dtTeks,TEKS.length()+1);

```

```

dmd.clearScreen( true );

dmd.selectFont(Arial_black_16);

dmd.drawMarquee(dtTeks,TEKS.length()+1,(32*DISPLAYS_ACROSS)-1,0);

long start=millis();

long timer=start;

boolean ret=false;

while(!ret){

    if ((timer+SPEED) < millis()) {

        ret=dmd.stepMarquee(-1,0);

        timer=millis();

    }

    serialSMS();

}

}

void serialSMS() {

    while (Serial.available()) {

        char inChar = (char)Serial.read();

        if (inChar == '*') {

            dataValid = true;

            inputString += inChar;

        }else if (inChar == '\n') {

        }else if (inChar == '#') {

```

```

if(inputString.indexOf(LockNumber) != -1){

    int pos1 = inputString.indexOf("*");

    dtRunningTeks = inputString.substring(pos1+1,inputString.length());

    if(dtRunningTeks != ""){

        TEKS = dtRunningTeks;

        eepSaveString(0,TEKS);

    }

}

inputString = "";

}else{

    if(inChar > 30) inputString += inChar;

}

}

}

// simpan data string ke eeprom

void eepSaveString(int addr,String dtSave){

    int dtLen = dtSave.length()+1;

    char dtBuff[dtLen];

    int addrS = addr;

    dtSave.toCharArray(dtBuff,dtLen);

    dtBuff[dtLen-1]='#'; // beri kode eof

```

```

for(int j=0;j<dtLen;j++){
    EEPROM.write(addr, dtBuff[j]); delay(10);
    addr++;
}
}

// baca data string dari eeprom
String eepLoadString(int addr, int jlhBaca){
    String rtn;
    int addr = addr;
    for(int j=0;j<100;j++){
        char dt = EEPROM.read(addr); delay(10);
        if(dt == '#'){ // cek eof
            break;
        }else{
            rtn += dt;
        }
        addr++;
    }
    if(addr >= addr + jlhBaca){ // kalo gak nemu eof maka data kosong
        rtn = "";
    }
}

```

### 3.7.3 Cara Menggunakan Software Arduino IDE

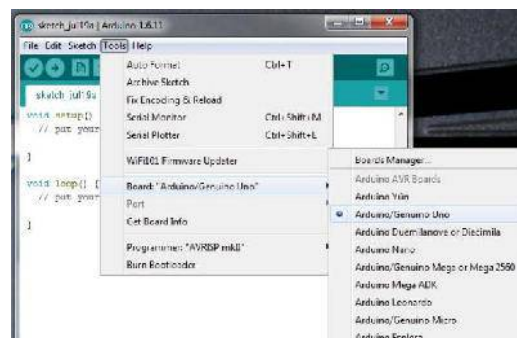
Arduino selain memiliki hardware juga memiliki software, untuk hardware sudah dijelaskan mengenai jenis-jenisnya, karakteristik dan konfigurasi pin Arduino yang tersedia, untuk software, Arduino menggunakan software sendiri dan bahasa pemrograman sendiri yang dinamakan Arduino IDE, bahasa pemrograman Arduino yang dipakai yaitu C / C++.

#### 1. Cara Setting Arduino IDE

Perlu diketahui bahwa pada bagian ini sangat menentukan sekali dalam kelancaran saat upload program dilakukan, jadi jika salah setting akan menyebabkan program tidak jalan dan juga ada tampilan error pada editor, maka dari itu perhatikanlah setting awal berikut ini untuk Arduino Uno dan sejenisnya.

##### a. Langkah 1

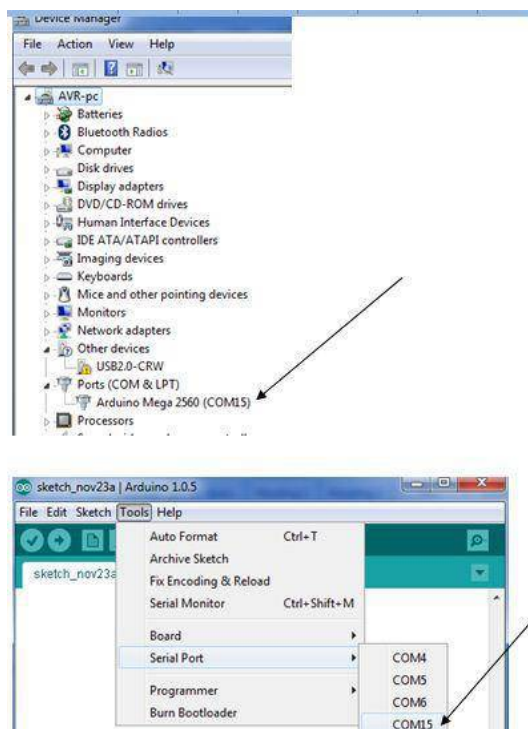
Hubungkan kabel USB dari komputer ke Arduino. Tentukan jenis board Arduino yang dipakai, pada contoh menggunakan Arduino UNO, Pilih menu Tools -> Board -> Arduino UNO (Tergantung jenis Arduino yang dipakai)



**Gambar 3.5** Tampilan jendela Arduino IDE menu Tools

b. Langkah 2

Tentukan serial port yang dipakai, ini digunakan untuk memprogram device Arduino, jika belum tahu serial port berapa yang dipakai dapat dilihat di device manager -> Ports, lalu cek nomer COM-nya, pada contoh tertulis COM15 artinya Port komunikasi yang digunakan adalah COM15, ini bisa berubah-ubah, maka dari itu pastikan dahulu COM yang dipakai sesuai dengan setting Arduino, Setelah tahu COM yang digunakan, kembali ke Arduino IDE, Pilih Tools -> Serial Port -> COM15. Jika berhasil maka pada bagian editor bawah akan tampak COM yang dipakai.



**Gambar 3.6** Tampilan Arduino IDE menu tools Serial Port



## 2. Cara Upload Program ke Arduino

Setelah proses setting, hal selanjutnya yaitu mengetahui cara mengupload program yang sudah kita buat, pada contoh ini menggunakan program kosong lalu di upload ke Arduino, jika berhasil maka akan ada keterangan berhasil, namun jika error maka di bawah editor akan tertulis error. Langkah langkahnya seperti berikut.

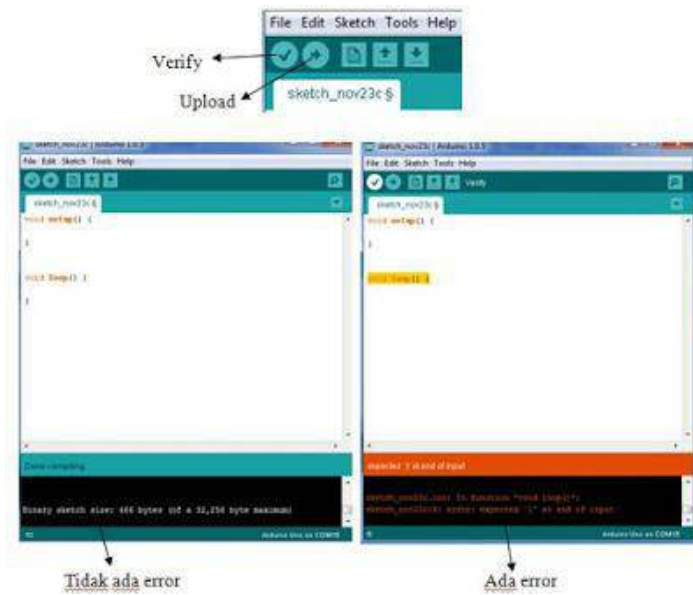
### a. Langkah 1

Tuliskan sebuah program pada editor yang mana program tersebut akan dieksekusi oleh Arduino, dicontohkan seperti dibawah ini :

```
voidsetup(){  
  
}  
  
voidloop(){  
  
}
```

### b. Langkah 2

Setelah menulis program kemudian masuk ke proses upload, caranya yaitu tekan tombol verify, kemudian jika tidak ada error, tekan tombol upload, tunggu sampai proses selesai.



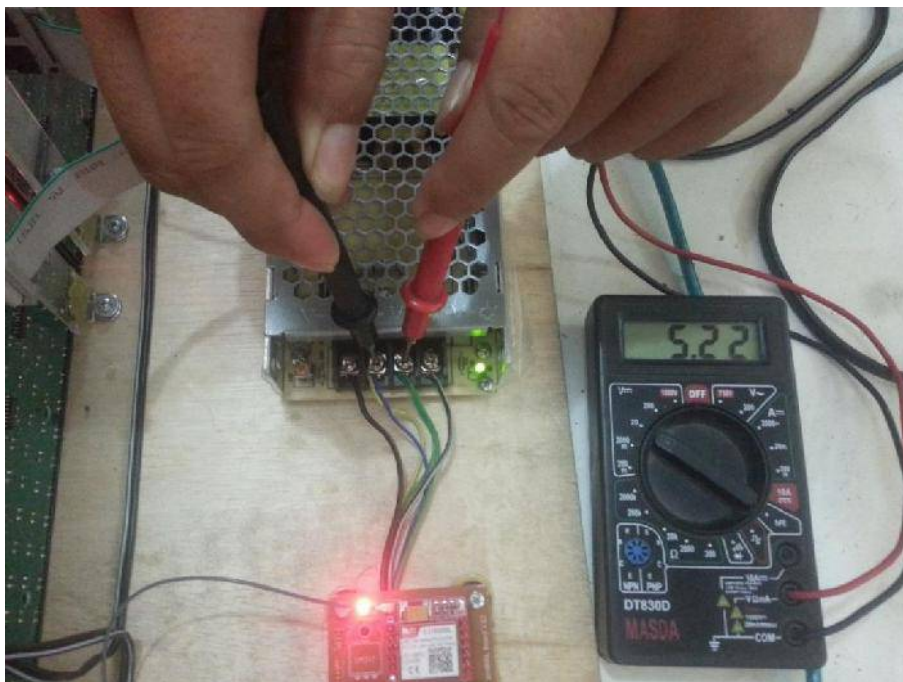
**Gambar 3.7** Tampilan Sketch dan Source Code pada Arduino

## BAB IV

### HASIL PENELITIAN DAN PEMBAHASAN

#### 4.1 Pengujian Power Supply 5V 2A

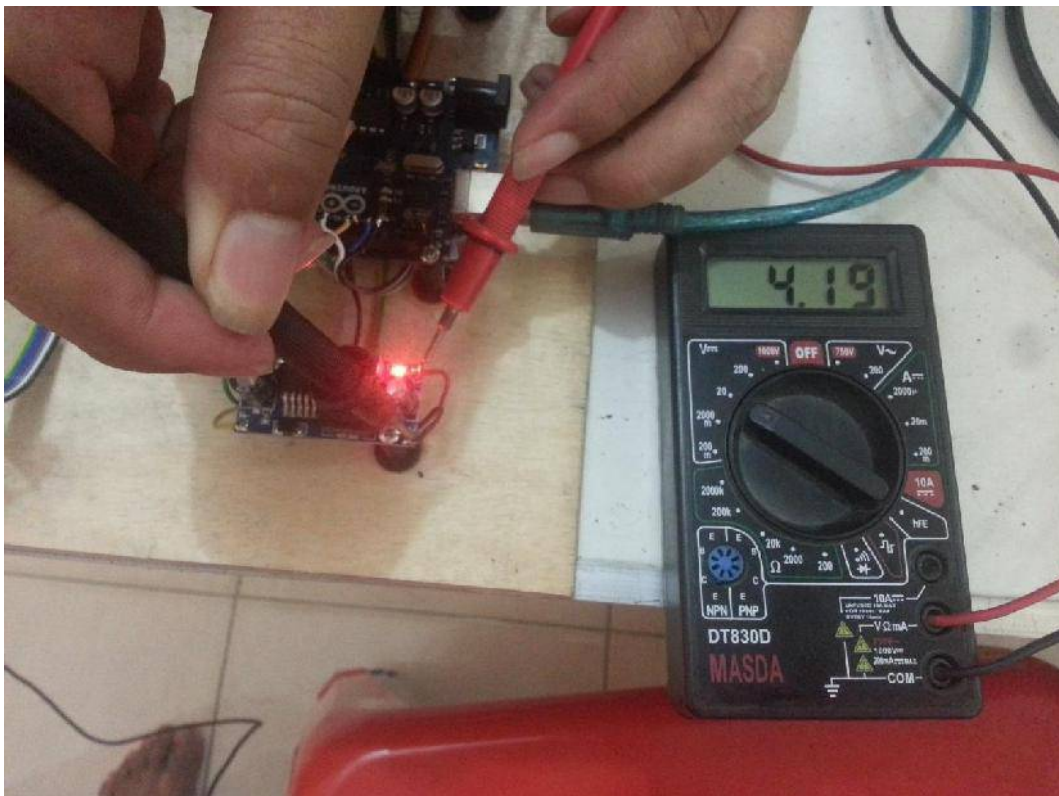
Pengujian ini dilakukan tujuannya adalah untuk mengecek kondisi dari Power Supply dapat bekerja dengan baik atau tidak. Pengukuran ini dilakukan dengan menggunakan multimeter sebagai alat untuk mengukur tegangan output power supply. Kabel positif multimeter dihubungkan ke kutub positif Power Supply dan kabel negative multimeter dihubungkan ke kutub negatif Power Supply juga, seperti terlihat oleh gambar 4.1 dibawah ini. Dari hasil pengukuran di dapatkan hasil yaitu sebesar 5.22 Volt dan ini membuktikan bahwa Power Supply dalam keadaan atau kondisi baik dan bagus.



**Gambar 4.1** Pengujian Power Supply

#### 4.2 Pengujian Tegangan Output Step Down LM2596 Mini

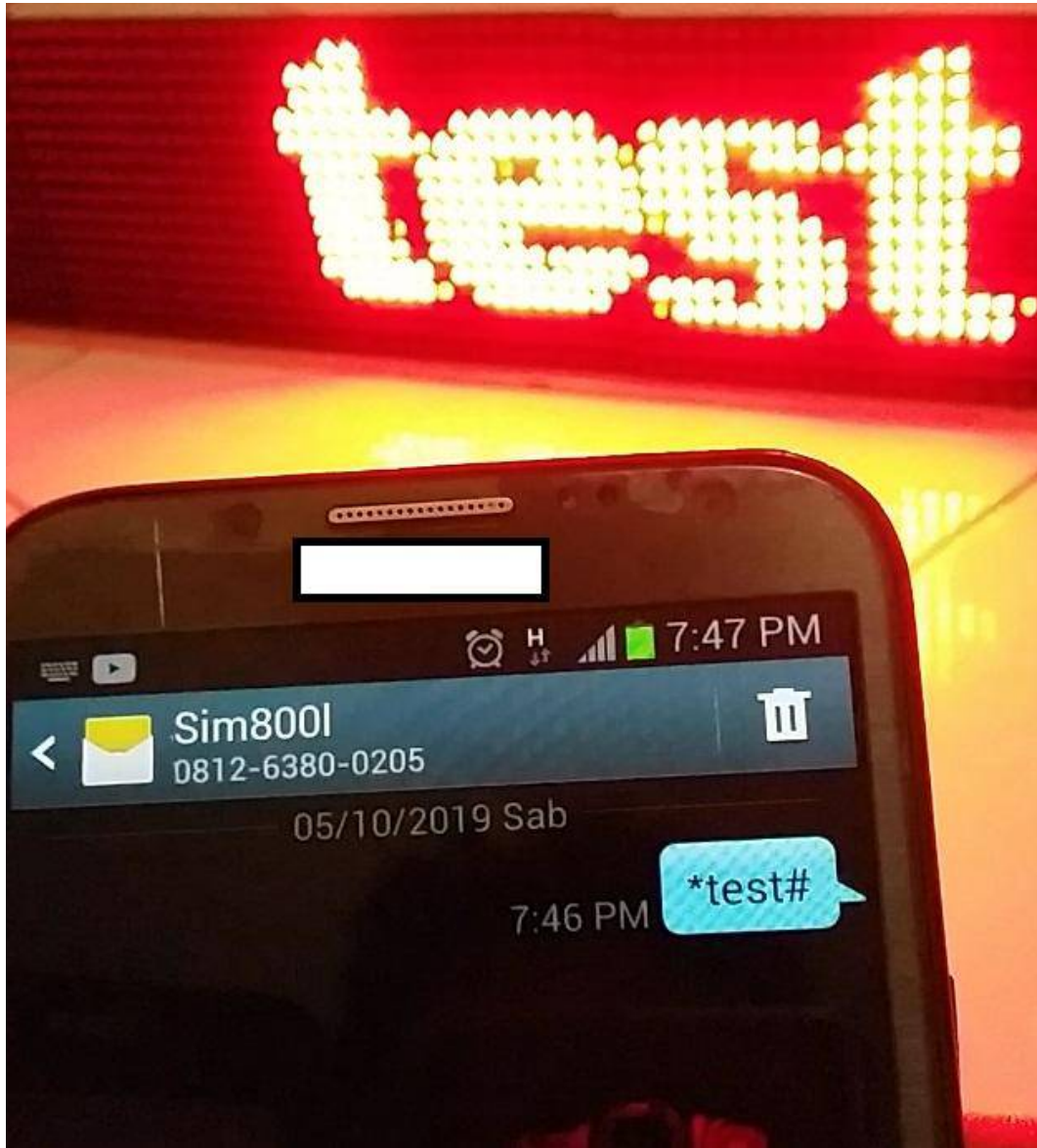
Pengukuran atau pengujian ini dilakukan untuk membuktikan bahwa LM2596 Mini masih dalam kondisi baik. Seperti terlihat pada gambar 4.3 dibawah ini. Dari hasil pengujian di dapatkan hasil yaitu sebesar 4.19 Volt. Pengujian ini dilakukan dengan cara menghubungkan kutub positive LM2596 pada negative multimeter dan kutub negative LM2596 pada positive multimeter.



**Gambar 4.2** Tegangan Output Step Down LM2596 Mini

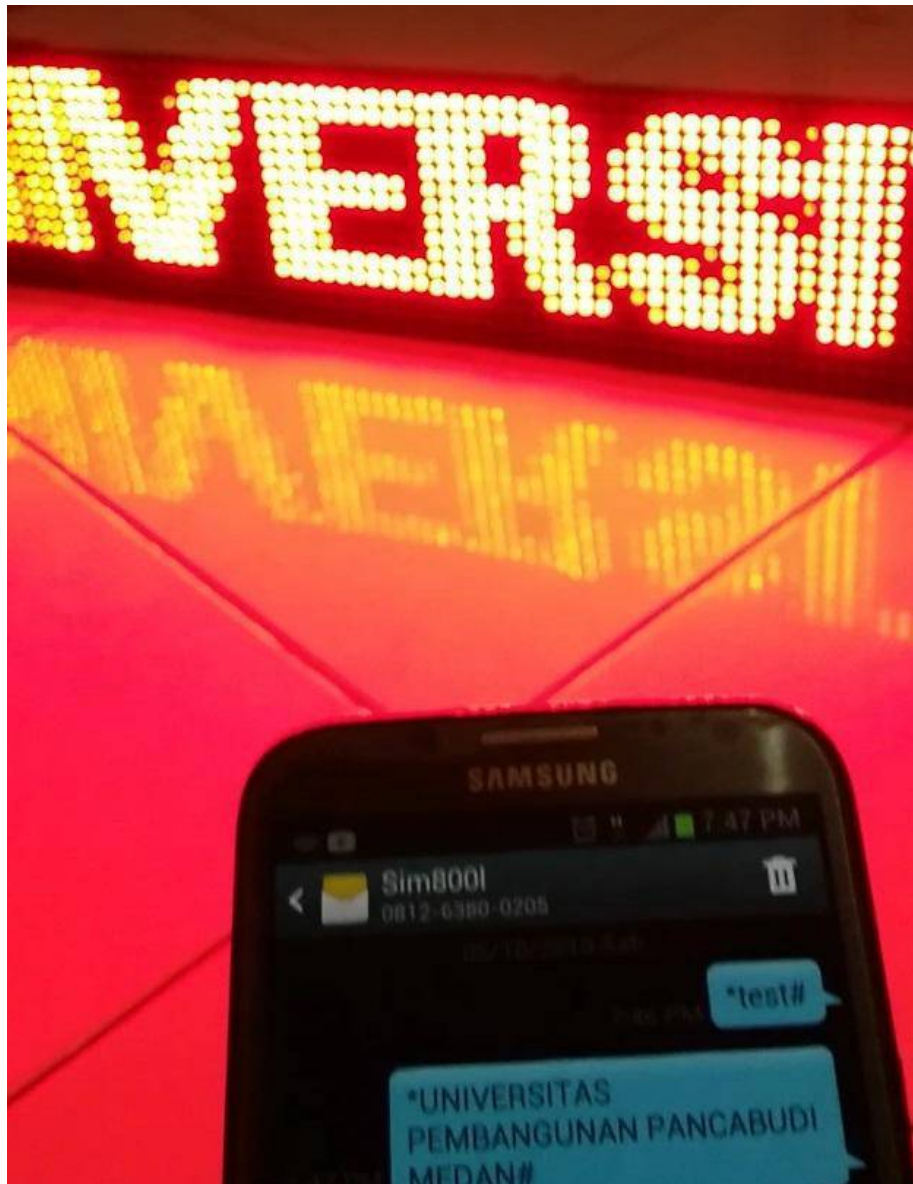
### 4.3 Gambar Percobaan

#### 4.3.1 Gambar Tampilan Hasil Pengujian Alat



**Gambar 4.3** Tampilan Hasil Display P10 Saat Pengujian





**Gambar 4.4** Tampilan Hasil Display P10 Saat Pengujian



**Gambar 4.5** Tampilan Hasil Display P10 Saat Pengujian

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Hasil dari perancangan dan pengujian *Running Text* ada beberapa kesimpulan yang dapat penulis ambil sebagai berikut:

1. Dengan adanya komponen Sim800L yang berfungsi untuk menerima pesan dari perangkat lain, maka telpon genggam yang telah ditunjuk sebagai operator dapat digunakan untuk mengedit text pada Display P10 *running text*.
2. Perubahan text berguna untuk menampilkan informasi sesuai keinginan dan kebutuhan.
3. Input yang dilakukan pada alat ini sudah berjalan dengan baik, sesuai dengan fungsinya yaitu dapat menghasilkan *output text* pada panel display P10.

#### **5.2 Saran**

Dari hasil skripsi ini masih banyak terdapat beberapa kekurangan dan dimungkinkan untuk pengembangan lebih lanjut. Oleh karenanya penulis merasa perlu memberi beberapa saran sebagai berikut:

1. Arduino Uno sebagai Chip kontroler diharapkan dapat membantu dalam pengembangan dan kemajuan khususnya dunia teknologi. Serta



contoh implementasi nyata yang masih banyak lagi yang dapat dikembangkan semaksimal mungkin dari sebuah mikrokontroler.

2. Agar tampilan tidak hanya menampilkan text(huruf), maka diperlukan penyempurnaan program sehingga nantinya display dapat menampilkan berupa animasi.

## DAFTAR PUSTAKA

- Ahmad Fatoni, dkk (2015). Rancang Bangun Alat Pembelajaran *Microcontroller* Berbasis Atmega 328 di Universitas Serang Raya Vol : 2, No 1 ISSN : 2406-7733
- Bahrin, (2017). Sistem Kontrol Penerangan Menggunakan Arduino Uno Pada Universitas Ichsan Gorontalo
- Desmira dan Didik Ariwibowo, (2016). Perancangan Smarthome Dengan Rasberry Berbasis Wireless Menggunakan Mikrokontoller AVR Atmega328 dan Fuzzy Logic Vol : 7 No 2, 2016 ISSN : 2252-4983
- Eka W Fridayanthie dan Jimmy Charter, (2016). Rancang Bangun Sistem Informasi Simpan Pinjam Karyawan Menggunakan Metode Object Oriented Programming Vol XIII No 2 ISSN : 1978-2136
- Fachri, barany, agus perdana windarto, and ikhsan parinduri. "penerapan backpropagation dan analisis sensitivitas pada prediksi indikator terpenting perusahaan listrik." jepin (jurnal edukasi dan penelitian informatika) 5.2 (2019): 202-208.
- Fachri, b., windarto, a. P., & parinduri, i. (2019). Penerapan backpropagation dan analisis sensitivitas pada prediksi indikator terpenting perusahaan listrik. Jepin (jurnal edukasi dan penelitian informatika), 5(2), 202-208.
- Fachri, barany; windarto, agus perdana; parinduri, ikhsan. Penerapan backpropagation dan analisis sensitivitas pada prediksi indikator terpenting perusahaan listrik. Jepin (jurnal edukasi dan penelitian informatika), 2019, 5.2: 202-208
- Hamdi, nurul. "model penyiraman otomatis pada tanaman cabe rawit berbasis programmable logic control." jurnal ilmiah core it: community research information technology 7.2 (2019)
- Imelda U.V. Simanjuntak, Asep Suhendar, (2018). Rancang Bangun Running Text P10 16x32 Berbasis Arduino Uno dengan Komunikasi SMS. *Jurnal Ilmiah Teknologi Informasi Terapan* Volume IV, No 2, 30 April 2018 ISSN : 2407-3911
- Marco Failla and Gian Paolo Vitale, (2019). A Double Stage DC/DC Converter for LED Lighting Automotive Systems Volume No 17, ISSN : 2172-038X
- Permana, aminuddin indra. "kombinasi algoritma kriptografi one time pad dengan generate random keys dan vigenere cipher dengan kunci em2b." (2019).

- Prio Handoko, (2017). Sistem Kendali Kerangka Elektronik onolitik berbasis arduino uno r3 ISSN : 2407-1846
- Putra, randi rian. "sistem informasi web pariwisata hutan mangrove di kelurahan belawan sicanang kecamatan medan belawan sebagai media promosi." jurnal ilmiah core it: community research information technology 7.2 (2019).
- Putra, randi rian, et al. "decision support system in selecting additional employees using multi-factor evaluation process method." (2019).
- Putra, randi rian. "implementasi metode backpropagation jaringan saraf tiruan dalam memprediksi pola pengunjung terhadap transaksi." jurti (jurnal teknologi informasi) 3.1 (2019): 16-20.
- Rachmat Adi Purnama dan Adi Tri Laksono Putra, (2018). Aplikasi Web Server Berbasis Bahasa C Sharp Vol 4 No 1 ISSN : 2442-2436
- Rian Affrilianto, dkk (2017). Rancang Bangun Sistem Pelacak Kendaraan Bermotor Menggunakan GPS Dengan Antarmuka *website* Vol 5 No 3, ISSN : 2338-493X
- Saputra, muhammad juanda, and nurul hamdi. "rancang bangun aplikasi sejarah kebudayaan aceh berbasis android studi kasus dinas kebudayaan dan pariwisata aceh." journal of informatics and computer science 5.2 (2019): 147-157
- Sidik, a. P., efendi, s., & suherman, s. (2019, june). Improving one-time pad algorithm on shamir's three-pass protocol scheme by using rsa and elgamal algorithms. In journal of physics: conference series (vol. 1235, no. 1, p. 012007). Iop publishing.
- Sitepu, n. B., zarlis, m., efendi, s., & dhany, h. W. (2019, august). Analysis of decision tree and smooth support vector machine methods on data mining. In journal of physics: conference series (vol. 1255, no. 1, p. 012067). Iop publishing
- Sumarsono, Dwiatmi Saptaningtyas, (2018). Pengembangan Mikrokontroler Sebagai *Remote Control* Berbasis Android Vol : 11 No. 1 ISSN : 1979-9160
- Suwitno, (2016). Mendisain Rangkaian Power Supply pada Rancang Bangun Miniatur Pintu Garasi Otomatis Vol 1, No 1 ISSN : 2502-3624
- Tasril, v., wijaya, r. F., & widya, r. (2019). Aplikasi pintar belajar bimbingan dan konseling untuk siswa sma berbasis macromedia flash. Jurnal informasi komputer logika, 1(3).