

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Pemanfaatan teknologi *Smartphone* berbasis *Android* sebagai alat komunikasi dan telepon cerdas telah banyak mengalami perkembangan, salah satunya adalah sebagai *remote control* yang dapat mengendalikan perangkat elektronik dari jarak jauh. Misalnya mengendalikan lampu rumah dengan *WiFi*, mengendalikan AC (*Air Conditioner*) dengan *WiFi*, mengendalikan robot mobil dengan menggunakan *WiFi*, monitoring suhu ruangan, dan lain – lain. *Remote control* secara tidak langsung dapat mengurangi tugas *user* dalam menangani sebuah kendali peralatan elektronik.

Pengembangan fasilitas atau perangkat berbasis *wireless* dimaksudkan agar semua fasilitas atau perangkat dapat terintegrasi dalam suatu sistem, yang merujuk pada perkembangan IoT (*Internet of Things*). Di dalam salah satu ruangan memiliki 4 buah lampu yang dikendalikan secara manual. Permasalahannya ketika pengguna lupa untuk memadamkan lampu pada saat keluar ruangan atau pergi, maka akan terjadi pemborosan listrik. Untuk itu diperlukan sebuah alat yang dapat mengendalikan lampu dari jarak jauh.

Teknologi sistem kendali ini diperlukan untuk efisiensi dan waktu jam kerja pengguna serta dari segi penghematan energi listrik yang digunakan. Karena dalam proses *on/off* lampu di ruangan masih dilakukan secara manual, jadi pengguna untuk *on/off* lampu harus berjalan menuju saklar lampu. (Iswanto dan Gandi,2018)

Pengguna juga dapat memantau dan mengendalikan perangkat- perangkat listrik di dalam rumah dari jarak jauh melalui suatu saluran komunikasi seperti melalui jaringan internet, *WiFi* atau *bluetooth*. Salah satu pengendali itu menggunakan mikrokontroler.

Mikrokontroler adalah suatu perangkat yang digunakan untuk mengontrol suatu proses atau aspek-aspek dari lingkungan. Mikrokontroler itu sendiri adalah *chip* atau IC (*Integrated Circuit*) yang bisa diprogram menggunakan komputer. Tujuan menanamkan program pada mikrokontroler adalah agar rangkaian elektronik dapat membaca input memproses *input* dan kemudian menghasilkan output sesuai yang diinginkan (Setiawan, M. M. , 2019)

Salah satu pengendali yang berbasis *open source* adalah Arduino Uno. Arduino adalah mikrokontroler serbaguna yang memungkinkan untuk diprogram. Program di Arduino biasanya dinamakan dengan *Sketch*. Dengan menuliskan *sketch*, maka dapat memberikan beberapa intruksi yang akan membuat Arduino melaksanakan tugas sesuai dengan instruksi-instruksi yang diberikan. Selain itu, *sketch* juga dapat diubah sesuai dengan kebutuhan dan waktu yang diinginkan. Arduino Uno menggunakan mikrokontroler ATmega328 yang dibuat oleh Atmel Corporation dengan 8-bit kecepatan transfer data. Arduino Uno memiliki SRAM (*Static Random Access Memory*) sebesar 2KB yang digunakan sebagai memori kerja selama *sketch* dijalankan. Arduino Uno juga memiliki EEPROM (*Electrically Erasable Programmable Read-Only Memory*) sebesar 1KB yaitu memori yang dapat digunakan untuk menyimpan data secara permanen dan *Flash Memory* yang digunakan untuk menyimpan *sketch* (Program) (Hudori, M., & Paisal, Y. 2019)

Koneksi Wifi ESP8266 yang bersifat SOC (*System on Chip*), sehingga kita bisa melakukan programming langsung ke ESP8266 tanpa memerlukan mikrocontroller tambahan. Kelebihan lainnya, ESP8266 ini dapat menjalankan peran sebagai adhoc akses poin maupun klien sekaligus. (Sinuardino, 2016). Dengan memanfaatkan kelebihan *Wifi* ESP8266 memungkinkan untuk menghubungkan Arduino Uno terhadap perangkat *Android*. Dari penjelasan untuk mengatasi masalah diatas perlu dilakukan penelitian pembuatan alat pengaturan lampu jarak jauh berbasis IoT (*Internet of Things*), maka penulis mengadakan penelitian tugas akhir dengan judul “ **SISTEM KENDALI LAMPU BERBASIS IOT ( *INTERNET OF THINGS* ) ANDROID** “.

## 1.2 Rumusan Masalah

1. Bagaimana sistem kendali *on/off* lampu menggunakan *Smartphone* android dengan memanfaatkan fasilitas *wifi* ESP8266 ?
2. Bagaimana program pengendali *on/off* dibuat sebagai dasar program kerja untuk menjalankan atau mengendalikan penggunaan lampu dengan menggunakan *smartphone* android memanfaatkan fasilitas *wifi* ESP8266 ?

## 1.3 Batasan Masalah

Agar tidak meluas maksud dan tujuan penelitian ini, maka permasalahannya hanya dibatasi sebagai berikut :

1. Penulis hanya membahas tentang sistem kendali *on/off* lampu menggunakan *smartphone android* dengan memanfaatkan fasilitas *wifi* ESP8266.
2. Program pengendali dibuat dengan bahasa pemrograman C Arduino

IDE dan aplikasi *Android* dibuat dengan *software tool* visual MIT App Inventor.

3. Penelitian menggunakan arduino ATmega328S
4. Penelitian menggunakan hostpot *handphone*

#### **1.4 Tujuan penelitian**

1. Memahami sistem pengendalian *on/off* lampu dengan menggunakan *smartphone android* melalui koneksi *wifi* yang saling terhubung dalam jangkauan koneksi *wifi* ESP8266.
2. Memahami program dasar sistem kerja pada aplikasi *android* yang dapat mengendalikan atau sebagai dasar pengendalian fungsi *on/off* lampu melalui koneksi *wifi* ESP8266.

#### **1.5 Manfaat penelitian**

Manfaat penelitian ini :

1. Bagi Universitas

Universitas Islam Sumatera Utara, yaitu dapat menjadi referensi bagi mahasiswa untuk penelitian selanjutnya

2. Bagi pengguna

Bagi pengguna yaitu dapat memperoleh cara baru dalam mengendalikan perangkat lampu melalui koneksi *wifi* ESP8266

3. Bagi peneliti

Dapat menerapkan ilmu dan mengembangkan kemampuan yang diperoleh selama perkuliahan serta mengaplikasikan kemampuan dalam pembuatan sistem kendali lampu berbasis *iot (internet of things) android*

## **1.6 Sistematika penulisan**

Agar penulisan dapat terarah dan terhindar dari pembahasan yang berulang-ulang serta memudahkan pembaca dan memahami, maka diperlukan sistematika penulisan, yaitu sebagai berikut:

### **BAB I : PENDAHULUAN**

Pada bab ini dibahas mengenai latar belakang, tujuan dan manfaat penulisan, rumusan masalah, batasan masalah, dan sistematika penulisan.

### **BAB II : TINJAUAN PUSTAKA**

Tinjauan pustaka memuat penjelasan tentang konsep dan prinsip dasar yang diperlukan untuk memecahkan masalah penelitian dan membahas tentang teori-teori yang menjelaskan beberapa pengertian, konsep dasar serta beberapa hal yang berhubungan dengan judul yang diangkat penulis.

### **BAB III : METODE PENELITIAN**

Bab ini berisi perencanaan kebutuhan, tahap analisis dan tahap desain beserta aksi yang diperlukan setiap tahap.

### **BAB IV : HASIL DAN PEMBAHASAN**

Dalam bab ini berisi hasil-hasil yang diperoleh dalam penelitian dan pembahasan terhadap hasil yang telah dicapai maupun masalah-masalah yang ditemui selama penelitian, uji coba, termasuk kelemahan dan kelebihan sistem yang dibuat.

### **BAB V : KESIMPULAN DAN SARAN**

Berisikan kesimpulan dan saran dari penulis

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Sistem Kendali**

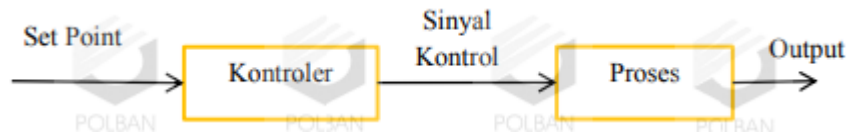
Perkembangan kendali otomatis dicapai secara signifikan setelah ditemukannya governor sentrifugal untuk mesin uap oleh James Watt pada abad 18. Penemuan-penemuan yang memicu perkembangan sistem kendali diantaranya adalah penemuan kendali kapal otomatis oleh Minorsky yang dapat menunjukkan kestabilan dari persamaan diferensial sebuah sistem. Hingga Nyquist mengembangkan prosedur sederhana untuk kendali *loop* tertutup dari respon *loop* terbuka sebuah sistem.

Sistem kendali adalah kumpulan peralatan yang bekerja untuk melakukan aksi kendali terhadap kondisi lingkungan agar sesuai dengan kondisi yang diharapkan. Sistem kendali terdiri dari *input* dan *set point* menyatakan kondisi terkini dari lingkungan atau respon sistem, *output* yang merupakan keluaran atau respon sistem aktual, dan plant yaitu objek yang dikendalikan atau dikontrol. Secara umum ada dua teknik dasar pengontrolan yaitu *loop* terbuka dan *loop* tertutup.

##### **2.1.1 Sistem Kendali Loop Terbuka (*OpenLoop*)**

Pada sistem kendali *loop* terbuka, respon sistem tidak mempengaruhi aksi pengontrolan. Tindakan pengendalian tidak dipengaruhi hasil keluaran sistem sehingga koreksi sistem tidak dapat dilakukan apabila terjadi gangguan. Dalam kendali ini, untuk memperoleh hasil yang baik juga sangat bergantung pada kalibrasi. Sistem kontrol *loop* terbuka dapat digunakan hanya jika hubungan antara masukan dan keluaran diketahui dan tidak

terdapat gangguan internal maupun eksternal. Dapat kita lihat pada gambar 2.1.



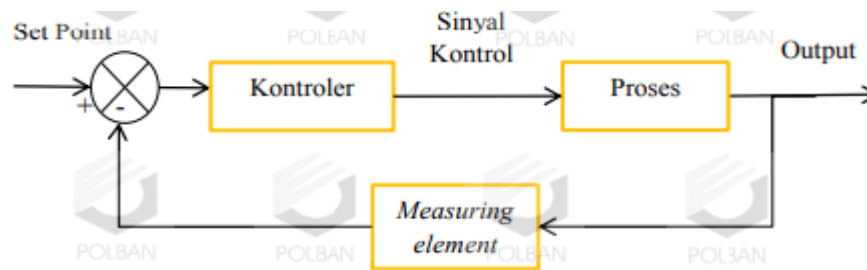
Gambar 2.1 Sistem Kontrol Loop Terbuka

*Sumber : (Zamrodah, Y. 2016).*

Dengan adanya gangguan, system kontrol *loop* terbuka tidak dapat bekerja seperti yang diinginkan. Kontrol *loop* terbuka dapat digunakan dalam praktek hanya jika hubungan antara masukan dan keluaran diketahui dan jika tidak terdapat gangguan internal maupun eksternal. Sebagai contoh pengontrolan lalu lintas dengan sinyal yang dioperasikan pada basis waktu adalah contoh lain dari Kontrol *loop* terbuka.

### 2.1.2 Sistem Kendali Loop Tertutup (*CloseLoop*)

Pada sistem kendali *loop* tertutup, keluaran atau respon sistem mempengaruhi aksi kendali. Tindakan pengendalian sangat tergantung pada *feedback* yang diambil dari hasil keluaran sistem. Sistem dapat mengatasi kelemahan yang ada karena dapat memberikan koreksi jika terjadi gangguan. Untuk menghasilkan tindakan pengendalian yang tepat diperlukan metode pengendalian untuk menghindari terjadinya *over correction* sehingga menyebabkan sistem tidak stabil. Dibandingkan dengan sistem loop terbuka, sistem loop tertutup lebih kompleks dan mahal, karena komponen lebih banyak.



Gambar 2.2 Sistem Kontrol Loop Tertutup

*Sumber* : (Zamrodah, Y. 2016).

## 2.2 Lampu

Menurut kamus bahasa Indonesia, arti kata lampu adalah alat untuk menerangi. Perkembangan lampu berawal dari sebuah lampu pijar yang selalu dicari inovasi kumparan sumber cahaya yang paling efisien. Pada tahun 1870-an, Thomas Alva Edison dari Menlo Park, Negara bagian New Jersey, Amerika Serikat, mendapatkan paten pertamanya pada bulan April 1879 untuk lampu pijar. Tahun 1933 filamen karbon diganti dengan filamen tungsten atau Wolfram (Wo) yang dibuat membentuk lilitan kumparan sehinggal dapat meningkatkan Efficacy lampu menjadi + 20 Lumen/W. sistem pembangkitan cahaya buatan ini disebut sistem pemijaran (Incondesence). Revolusi teknologi perlampuan berkembang dengan pesatnya. Pada tahun 1910 pertama kali digunakan lampu pendar (discharge) tegangan tinggi. Prinsip kerja lampu ini menggunakan sistem emisi elektron yang bergerak dari katoda menuju anoda pada tabung lampu akan menumbuk atom-atom media gas yang ada didalam tabung tersebut, akibat tumbukan akan menjadi pelepasan energi dalam bentuk cahaya. Sistem pembangkitan cahaya buatan ini disebut Luminscence (berpendarnya energy cahaya luar tabung). Media gas yang digunakan dapat berbagai macam, tahun 1932 ditemukan dilampu pendar dengan gas sodium tekanan rendah, dan tahun



1935 dikembangkan lampu pendar merkuri, dan kemudian tahun 1939 berhasil dikembangkan lampu Fluorescen, yang biasa dikenal dengan lampu neon. Selanjutnya lampu xenon tahun 1959, khusus lampu sorot dengan warna yang lebih baik telah dikembangkan gas metalhalide (halogen yang dicampur dengan iodine pada tahun 1964, pada sampai akhirnya lampu sodium tegangan tinggi tahun 1965. Prinsip emisi electron ini yang dapat meningkatkan efikasi lampu diatas 50 Lumen/W. jauh lebih tinggi dibanding dengan prinsip pemijaran.

### **2.3 Internet of Things (IoT)**

*Internet of Things* adalah suatu konsep yang bertujuan untuk memanfaatkan teknologi internet yang terus berkembang agar dapat diimplementasikan ke dalam benda fisik sehingga manusia dapat berinteraksi langsung dengan benda tersebut seperti mengirim data dan melakukan kendali jarak jauh secara *real-time*. Makna lain serupa, *Internet of Things* (IoT) adalah sebuah konsep di mana suatu objek yang memiliki kemampuan untuk mentransfer data melalui jaringan internet tanpa melakukan interaksi manusia ke manusia atau manusia ke komputer. Teknologi perangkat keras IoT yang digunakan pada umumnya adalah teknologi *Radio Frequency Identification* (RFID), *Wireless Sensor Network* (WSN), dan nano teknologi. Perangkat keras umum seperti kamera dan sensor api, sensor asap, sensor gas atau sensor suhu digunakan untuk IoT. Beberapa teknologi perangkat lunak adalah pemrosesan informasi dan teknologi keamanan. IoT memiliki arsitektur yang terdiri atas *perception layer*, *network layer*, dan *application layer*.

*Perception layer* adalah lapisan yang terdiri atas sensor dan perangkat yang digunakan untuk menerima data dari lingkungan yang diubah menjadi bentuk digital dan kemudian akan disalurkan ke *network layer*. Sensor yang dapat

digunakan contohnya dapat berupa RFID *chip*, perangkat yang dapat menerima data dari lingkungan, maupun gateway yang diakses oleh suatu perangkat. Kamera pada *smartphone* juga dapat digunakan sebagai sensor. *Network layer* adalah lapisan jaringan merupakan lapisan kedua yang berfungsi untuk menghubungkan lapisan sensor dengan lapisan aplikasi. Pada lapisan ini ditentukan informasi yang akan disalurkan pada lapisan aplikasi. Selain itu, pemrosesan data dilakukan pada lapisan ini. Kemampuan jaringan dan bagaimana data dikirim ditentukan pada lapisan ini. *Application layer* adalah lapisan terakhir pada arsitektur IoT yang digunakan adalah *application layer*. Lapisan ini merupakan antarmuka yang mudah digunakan oleh pengguna yang terhubung dengan lapisan jaringan. Pengguna dapat berkomunikasi dengan lapisan sensor untuk mendapatkan data yang sesuai dengan kebutuhan.

### **2.3.1 Sejarah Singkat *Internet of Things***

Pada tahun 1969 Departemen Pertahanan Amerika, U.S. DARPA (*Defense Advanced Research Projects Agency*) melakukan riset penelitian tentang bagaimana cara menghubungkan beberapa komputer agar saling terhubung dalam satu jaringan. Program riset tersebut dikenal dengan nama ARPANET. Pada tahun 1970, sepuluh komputer berhasil terhubung antara satu komputer dengan komputer lainnya sehingga dapat berkomunikasi dan membentuk sebuah jaringan. Memasuki tahun 1972 permintaan untuk mengembangkan internet semakin banyak yang mengakibatkan ARPANET terbagi menjadi dua, yaitu ARPANET dan MILNET. ARPANET lebih fokus menangani non-militer sedangkan MILNET fokus menangani keperluan militer pada saat itu. Dari keduanya bergabung dan sekarang

dikenal sebagai internet.

Pada tahun 1997 Paul Saffo memberikan penjelasan singkat pertama tentang sensor gas dan masa depan. Kevin Ashton yang pada saat itu menjabat sebagai direktur eksekutif Auto IDCentre, MIT berhasil menciptakan *The Internet of Things* yang pertama. Pada tahun yang sama mereka juga berhasil menemukan peralatan berbasis RFID (*Radio Frequency Identification*) global. Penemuan ini disebut sebagai sebuah lompatan besar dalam komersialisasi IoT. Pada tahun 2000 perkembangan IoT semakin pesat dimanfaatkan perusahaan LG dan mengumumkan untuk menciptakan kulkas pintar yang akan menentukan sendiri apakah bisa atau tidak makanan yang tersimpan di dalamnya diisi ulang.

Sistem RFID mulai diperkenalkan pada tahun 2003 dan sudah digunakan di tingkat yang lebih besar seperti pada militer Amerika Serikat di Program Savi mereka. Di tahun yang sama, raksasa ritel Walmart mulai menyebarkan RFID di toko-toko seluruh dunia untuk penggunaan komersial. Pada tahun 2005 arus publikasi utama seperti The Guardian, Amerika Ilmiah dan Boston Globe mengutip banyak artikel tentang IoT. Pada tahun 2008 IPSO Alliance diluncurkan oleh kelompok perusahaan yang digunakan untuk mempromosikan penggunaan Internet Protocol (IP) dalam suatu jaringan dari "*Smart object*" dan untuk mengaktifkan Internet of Things. Tahun 2008 FCC sudah menyetujui penggunaan "*white space spectrum*". dan beuruju peluncuran IPv6 di tahun 2011 yang memicu pertumbuhan besar di bidang *Internet of Things*, dalam teknologi tersebut ada beberapa perusahaan raksasa yang mendukung seperti Cisco, IBM,

Ericson untuk ikut dari bidang pendidikan dan komersial menggunakan teknologi IoT. Perkembangan *Internet of Things* yang semakin pesat hingga saat ini membuat semua peralatan yang biasa digunakan dalam kehidupan kita sehari-hari dapat dikendalikan dan dipantau menggunakan teknologi IoT. Mayoritas proses dilakukan menggunakan bantuan sensor gas IoT. Sensor gas dikerahkan pada setiap bagian perangkat yang dibutuhkan dan akan menghasilkan data mentah dan mengkonversikannya menjadi sinyal digital untuk dikirimkan ke pusat kontrol. Dengan cara ini pengguna dapat memonitor perubahan lingkungan jarak jauh dari setiap bagian dari dunia melalui internet. Arsitektur sistem ini akan didasarkan pada konteks operasi dan proses dalam skenario *real-time*.

*Smart City* sangat erat kaitannya dengan IoT salah satu contoh penerapannya adalah notifikasi gempa dan tsunami (*Early Warning System*) sehingga warga di daerah rawan bencana dengan mudah mendapatkan informasi secara langsung dalam hitungan detik. Selain itu, penggunaan IoT juga dapat ditemui pada lingkup yang lebih kecil seperti home automation atau rumah otomatis. Dalam penerapannya, *Home Automation* membuat pemilik rumah dapat dengan mudah mengontrol peralatan rumah tangga dengan memanfaatkan jaringan internet. (Maros, H., & Juniar, S. 2016.)

#### **2.4. Mikrokontroler**

Mikrokontroler adalah sebuah komputer kecil ("*special purpose computers*") di dalam satu IC yang berisi CPU, memori, *timer*, saluran komunikasi serial dan paralel, *Port input/output*, ADC. Mikrokontroler digunakan untuk suatu tugas dan menjalankan suatu program. Mikrokontroler ini adalah ilmu

terapan yang pengaplikasiannya dapat kita temui di kehidupan sehari-hari seperti jam digital, televisi, sistem keamanan rumah, dll. Mikrokontroler juga sangat banyak digunakan dalam penelitian dan pengembangan yang dilakukan oleh peneliti, dosen, guru, bahkan sekarang banyak mahasiswa yang mengangkat judul tesis/sekripsi/tugas akhir dengan berbasiskan mikrokontroller. Mikrokontroler adalah komponen yang sangat umum dalam sistem elektronika modern. Penggunaannya sangat luas, dalam kehidupan kita sehari-hari baik dirumah, kantor, rumah sakit, bank, sekolah, industri, dll. Mikrokontroler digunakan dalam sejumlah besar sistem elektronika seperti : sistem manajemen mesin mobil, *keyboard* komputer, alat ukur elektronik ( multimeter digital, *synthesizer* frekuensi, dan osiloskop ), televisi, radio, telepon digital, *mobile phone*, *microwave oven*, *printer*, *scanner*, kulkas, pendingin ruangan, CD/DVD *player*, kamera, mesin cuci, PLC (*programmable logic controller* ), robot, sistem otomasi, sistem akuisisi data, sistem keamanan, sistem EDC ( *Electronic Data Capture* ), mesin ATM, modem, router dll. Mikrokontroler dapat kita gunakan untuk berbagai aplikasi misalnya untuk pengendalian, otomasi industri, akuisisi data, telekomunikasi, dan lain-lain. Keuntungan menggunakan mikrokontroler yaitu harganya murah, dapat diprogram berulang kali, dan dapat kita program sesuai dengan keinginan kita. Saat ini keluarga mikrokontroler yang ada dipasaran yaitu intel 8048 dan 8051 (MCS51), Motorola 68HC11, microchip PI, hitachi H8, dan atmel AVR.

Mikrokontroler adalah Suatu kontroler digunakan untuk mengontrol suatu proses atau aspek-aspek dari lingkungan. Satu contoh aplikasi dari mikrokontroler adalah untuk memonitor rumah kita. Ketika suhu naik kontroler membuka jendela

dan sebaliknya. Pada masanya, kontroler dibangun dari komponen-komponen logika secara keseluruhan, sehingga menjadikannya besar dan berat. Setelah itu barulah dipergunakan mikroprosesor sehingga keseluruhan kontroler masuk kedalam PCB yang cukup kecil. Hingga saat ini masih sering kita lihat kontroler yang dikendalikan oleh mikroprosesor. Proses pengecilan komponen terus berlangsung, semua komponen yang diperlukan guna membangun suatu kontroler yang dapat dikemas dalam satu keping. Maka lahirlah komputer keping tunggal (*one chip microcomputer*) atau disebut juga mikrokontroler dapat dilihat pada Gambar 2.3.



Gambar 2.3. Mikroprosesor

*Sumber:* (Setiawan, M. M. , 2019).

Mikrokontroler adalah suatu dimana semua bagian yang diperlukan untuk suatu kontroler sudah dikemas dalam satu keping, biasanya terdiri dari:

1. CPU (*Central Processing Unit*)
2. RAM (*Random Access Memory*)
3. EEPROM/EPROM/PROM/ROM
4. I/O, Serial & Parallel
5. *Timer*

## 6. *Interrupt Controller*

Rata-rata mikrokontroler memiliki instruksi manipulasi bit, akses ke I/O secara langsung dan mudah, dan proses *interrupt* yang cepat dan efisien. Dengan kata lain mikrokontroler adalah " Solusi satu *Chip*" yang secara drastis mengurangi jumlah komponen dan biaya disain (harga relatif rendah).

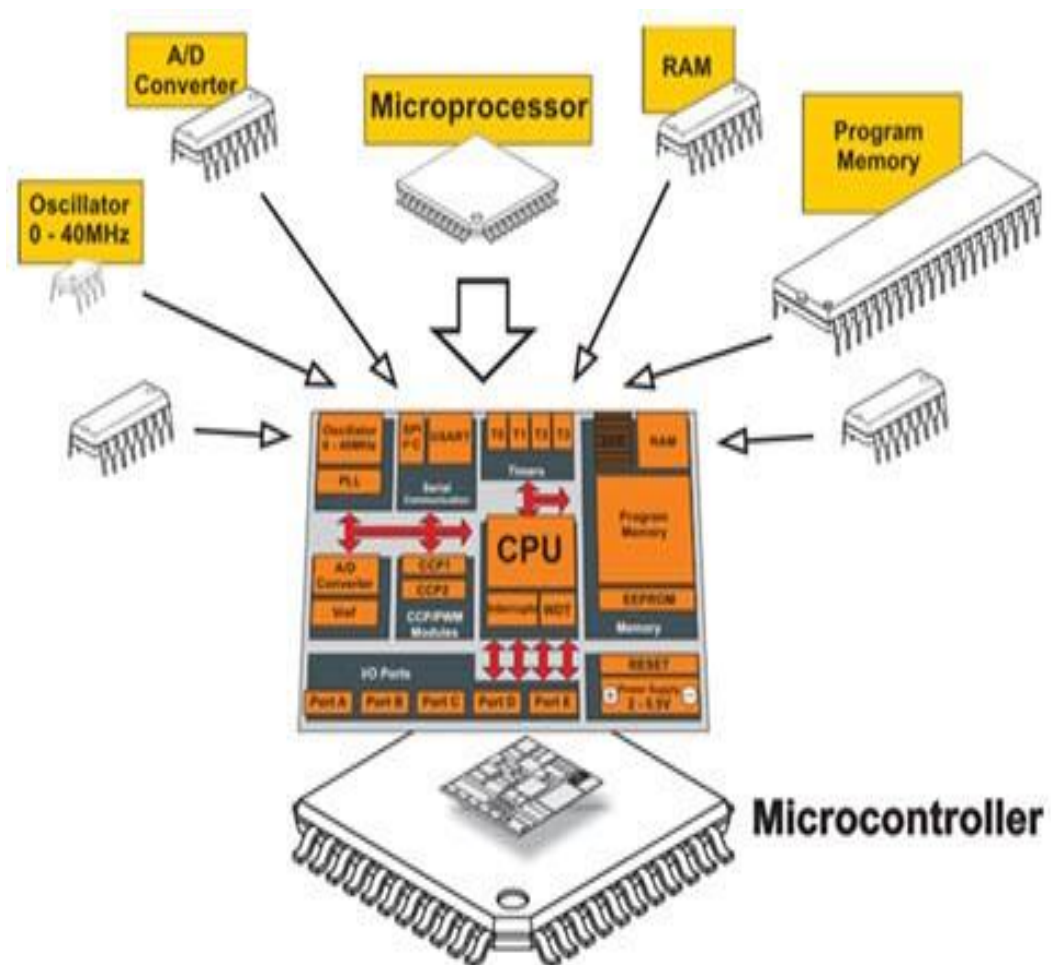
Selain sebagai sistem monitor rumah seperti diatas, mikrokontroler sering dijumpai pada peralatan rumah tangga (*microwave oven*, TV, stereo set dll), komputer dan perlengkapannya, mobil dan lain sebagainya. Pada beberapa penggunaan bisa ditemukan lebih dari satu prosesor didalamnya. Mikrokontroler biasanya digunakan untuk peralatan yang tidak terlalu membutuhkan kecepatan pemrosesan yang tinggi. Sifat spesial dari mikrokontroler adalah kecil dalam ukuran, hemat daya listrik serta *flexibilitasnya* menyebabkan mikrokontroler sangat cocok untuk dipakai sebagai pencatat/perekam data pada aplikasi yang tidak memerlukan kehadiran operator (Setiawan, M. M. , 2019).

### **2.4.1 Mikroprosesor, Mikro-Komputer dan Mikro-kontroler**

Mikroprosesor adalah *Central Professing Unit (CPU)* didalam single chip. Komponen CPU : *Arithmetic and Logic Unit (ALU)*, *instruction decoder*, *register*, *bus control circuit*, dll. Mikro-Komputer adalah mikroprosesor yang dihubungkan dengan rangkaian pendukung.

Komponen I/O dan memori (program & data) ditempatkan bersama untuk membentuk komputer kecil khususnya untuk akuisisi data dan aplikasi kontrol.

Jika komponen yang menyusun sebuah mikrokomputer diletakkan bersama didalam *single chip silicon* maka disebut mikrokontroler berisi CPU, memori, *timer*, *port* serial dan paralel, *port input / output*, ADC. (Setiawan, M. M. , 2019).



Gambar 2.4: Perbedaan Mikroprocessor dan Mikrokontroler

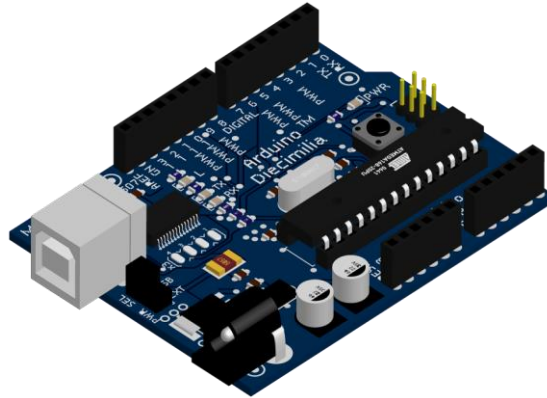
Sumber: (Setiawan, M. M. , 2019).

### 2.4.2 Board Arduino

Arduino merupakan papan rangkaian sistem minimum mikrokontroler yang memang dirancang untuk bisa digunakan dengan mudah oleh para seniman dan desainer (yang memang bukan orang



teknik). Dengan demikian, tanpa mengetahui bahasa pemrograman, Arduino bisa digunakan untuk menghasilkan karya yang canggih.



Gambar 2.5: Arduino Board

Sumber: (Risal. A , 2017)

Menggunakan Arduino sangatlah membantu dalam membuat suatu prototyping ataupun untuk melakukan pembuatan proyek. Arduino memberikan I/O yang sudah fix dan bisa digunakan dengan mudah. Arduino dapat digabungkan dengan modul elektro yang lain sehingga proses perakitan jauh lebih efisien. Para desainer hanya tinggal membuat *software* untuk mendayagunakan rancangan H/D yang ada. *Software* jauh lebih mudah untuk dimodifikasi tanpa memindahkan kabel.

Saat ini arduino sangat mudah dijumpai dan ada beberapa perusahaan yang mengembangkan sistem H/D *open source* ini. Pengembangan-pengembangan tersebut antara lain:

- a) Arduino : <http://www.arduino.cc/>
- b) I-CubeX : <http://www.infusionsystems.com/>
- c) Arie Robot Project Junior :

<http://www.arobotineveryhome.com/>

- d) Dwengo : <http://www.dwenfo.org/>
- e) EmbeddedLab : <http://www.embedded.arch.ethz.ch/>
- f) GP3 : <http://www.awce.com/gp3.htm/>

Di antara pengembang-pengembang yang ada, Arduino merupakan salah satu pengembang yang banyak digunakan. Pada buku ini, kita akan menggunakan produk pengembang dari Arduino. Keistimewaan Arduino adalah *hardware* yang *open source*. Hal ini sangatlah memberi keleluasan bagi pengguna untuk bereksperiment secara bebas dan gratis.

Arduino adalah pengendali mikro *single board* yang bersifat *opensource*, diturunkan dari *Wiring platform*, dirancang untuk memudahkan penggunaan elektronik dalam berbagai bidang. *Hardware*nya memiliki prosesor Atmel AVR dan *software*nya memiliki bahasa pemrograman sendiri.

Saat ini Arduino sangat populer di seluruh dunia. Banyak pemula yang belajar mengenal robotika dan elektronika lewat Arduino karena mudah dipelajari. Tapi tidak hanya pemula, para *hobbyist* atau profesional pun ikut senang mengembangkan aplikasi elektronik menggunakan Arduino. Bahasa yang dipakai dalam Arduino bukan *assembler* yang relatif sulit, tetapi bahasa C yang disederhanakan dengan bantuan pustaka-pustaka (*libraries*) Arduino. Dan seperti Mikrokontroler yang banyak jenisnya, Arduino lahir dan

berkembang, kemudian muncul dengan berbagai jenis. Diantaranya adalah:

A. **Arduino Uno**, Jenis yang ini adalah yang paling banyak digunakan.

Terutama untuk pemula sangat disarankan untuk menggunakan Arduino Uno. Dan banyak sekali referensi yang membahas Arduino Uno.

B. Versi yang terakhir adalah **Arduino Uno R3 (Revisi 3)**,

menggunakan ATMEGA328 sebagai *Microcontrollernya*, memiliki 14 pin I/O digital dan 6 pin input analog. Untuk pemograman cukup menggunakan koneksi USB type A to To type B. Sama seperti yang digunakan pada USB printer.

C. **Arduino Due**, Berbeda dengan saudaranya, Arduino Due tidak meng

gunakan ATMEGA, melainkan dengan *chip* yang lebih tinggi ARM Cortex CPU. Memiliki 54 I/O pin digital dan 12 pin *input* analog. Untuk pemogramannya menggunakan Micro USB, terdapat pada beberapa *handphone*.

D. **Arduino Mega**, Arduino Mega Mirip dengan Arduino Uno, sama-

sama menggunakan USB type A to B untuk pemogramannya. Tetapi Arduino Mega, menggunakan *Chip* yang lebih tinggi ATMEGA2560. Dan tentu saja untuk Pin I/O Digital dan pin *input* Analognya lebih banyak dari Uno.

E. **Arduino Leonardo**, Bisa dibilang Leonardo adalah saudara kembar

dari Uno. Dari mulai jumlah pin I/O digital dan pin input Analognya sama. Hanya pada Leonardo menggunakan Micro USB

untuk pemrogramannya.

F. **Arduino Fio**, Bentuknya lebih unik, terutama untuk socketnya.

Walau jumlah pin I/O digital dan input analognya sama dengan uno dan leonardo, tapi Fio memiliki Socket XBee. XBee membuat Fio dapat dipakai untuk keperluan proyek yang berhubungan dengan *wireless*.

G. **Arduino Lilypad**, Bentuknya yang melingkar membuat Lilypad

dapat dipakai untuk membuat proyek unik. Seperti membuat amor iron man misalkan. Hanya versi lamanya menggunakan ATMEGA168, tapi masih cukup untuk membuat satu proyek keren.

Dengan 14 pin I/O digital, dan 6 pin input analognya.

H. **Arduino Nano**, Sepertinya namanya, Nano yang berukuran kecil

dan sangat sederhana ini, menyimpan banyak fasilitas. Sudah dilengkapi dengan FTDI untuk pemrograman lewat Micro USB. 14 Pin I/O Digital, dan 8 Pin input Analog (lebih banyak dari Uno).

Dan ada yang menggunakan ATMEGA168, atau ATMEGA328.

I. **Arduino Mini**, Fasilitasnya sama dengan yang dimiliki Nano.

Hanya tidak dilengkapi dengan Micro USB untuk pemrograman. Dan ukurannya hanya 30 mm x 18 mm saja.

J. **Arduino Micro**, Ukurannya lebih panjang dari Nano dan Mini.

Karena memang fasilitasnya lebih banyak yaitu; memiliki 20 pin I/O digital dan 12 pin input analog.

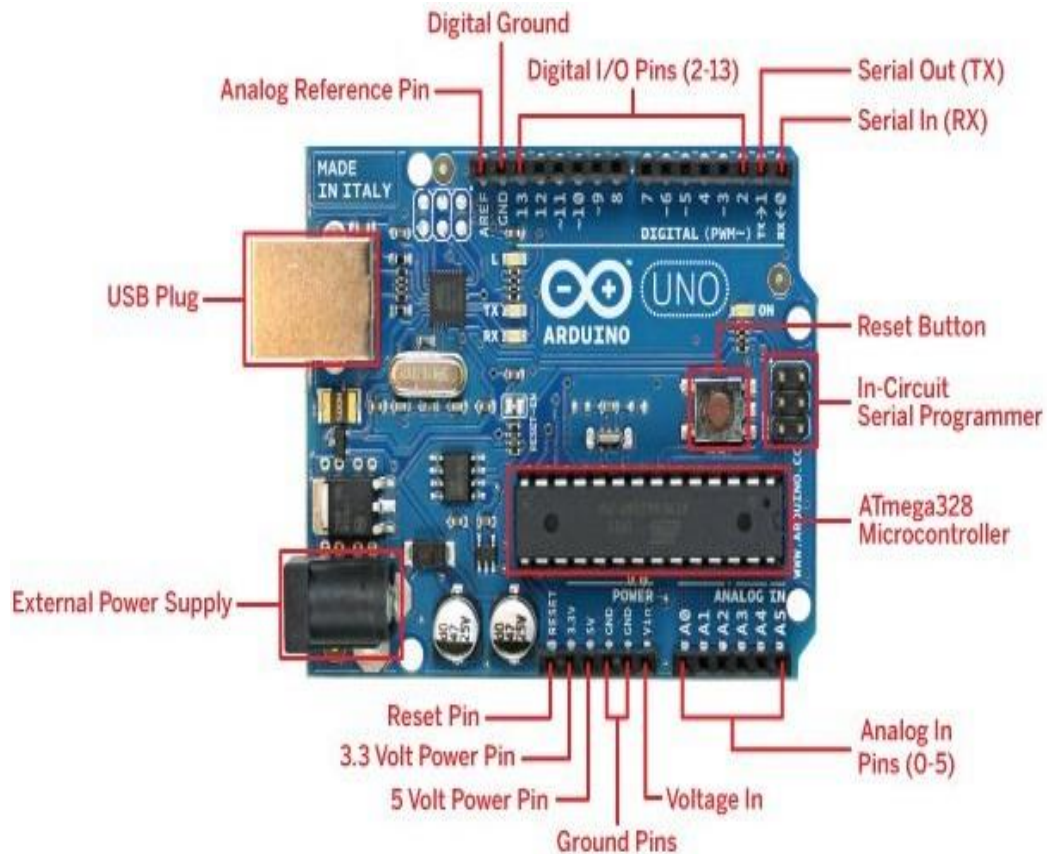
K. **Arduino Ethernet**, Ini arduino yang sudah dilengkapi dengan

fasilitas ethernet. Membuat Arduino kamu dapat berhubungan melalui jaringan LAN pada komputer. Untuk fasilitas pada Pin I/O Digital dan Input Analognya sama dengan Uno.

L. **Arduino Esplora**, Rekomendasi bagi kamu yang mau membuat *gadget* seperti *Smartphone*, karena sudah dilengkapi dengan *Joystick*, *button*, dan sebagainya. Kamu hanya perlu tambahkan LCD, untuk lebih mempercantik Esplora.

M. **Arduino Robot**, Ini adalah paket komplet dari Arduino yang sudah berbentuk robot. Sudah dilengkapi dengan LCD, *Speaker*, Roda, *Sensor Infrared*, dan semua yang kamu butuhkan untuk robot sudah ada pada Arduino ini.

Pembahasan pada penelitian ini menggunakan arduino jenis Arduino UNO. Arduino UNO adalah jenis arduino yang paling populer digunakan dikalangan pelajar maupun kalangan umum. Berikut ini adalah perangkat keras arduino dengan fungsi dari masing-masing bagiannya.



Gambar 2.6: Arduino UNO

*Sumber:* (Risal. A , 2017)

Arduino uno sebenarnya adalah salah satu kit mikrokontroler yang berbasis pada ATmega28. Modul ini sudah dilengkapi dengan berbagai hal yang dibutuhkan untuk mendukung mikrokontroler untuk bekerja, tinggal colokkan ke *power supply* atau sambungkan melalui kabel USB ke PC. *board* Arduino Uno ini sudah siap digunakan.

Kabel berwarna biru Gambar 2.7 adalah konektor USB tipe A ke tipe B. Tipe B biasanya adalah USB yang dipakai pada *printer inkjet*. Board pada Gambar 2.6 mempunyai sebuah port USB , sebuah colokan DC *power jack*, penghubung ISP, pin analog, pin *digital* dan tombol reset.

Port USB ini digunakan untuk koneksi data ke PC, DC power jack

untuk catu daya eksternal. Meski bisa bekerja dari tegangan 3V – 20V, tapi disarankan untuk memakai tegangan antara 5V – 12V saja, karena jika catu kurang dari 5V maka *output high* pada digital *output* tidak akan bernilai “1” secara penuh, dan jika catu melebihi 12V maka *output low* akan lebih dari 0V sehingga tidak akan bernilai “0” sempurna.



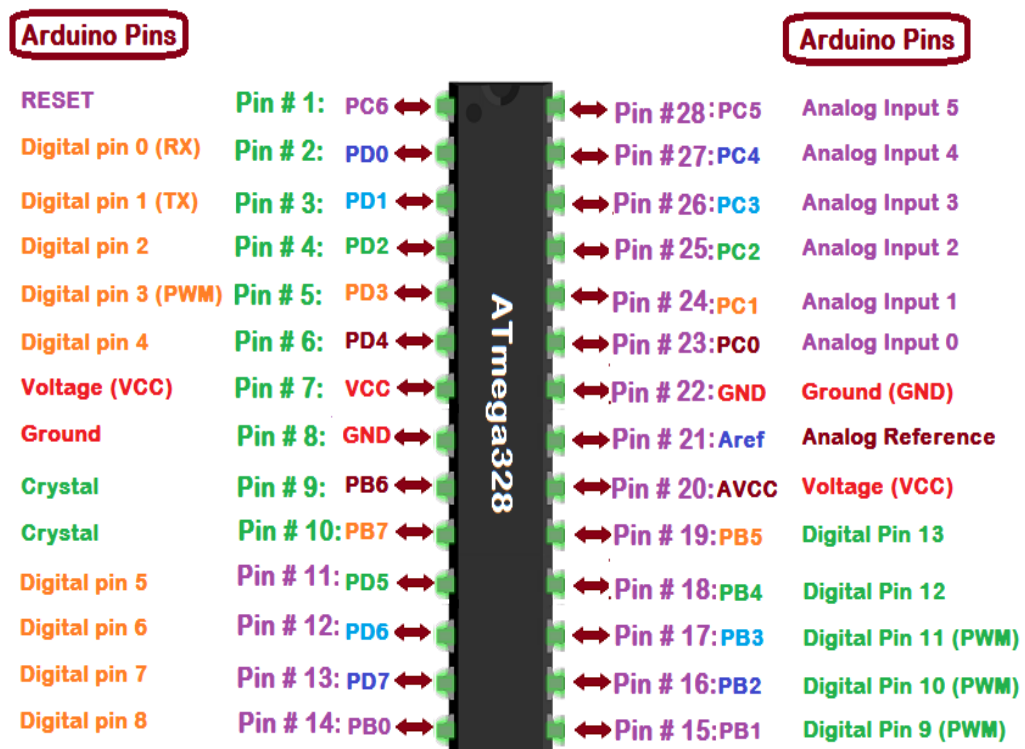
Gambar 2.7. Kabel USB .

Sumber : <https://duniaarduino.wordpress.com/2015/08/04/mengenal-arduino/>

Tombol reset pada Gambar 2.6 adalah tombol yang difungsikan untuk mengatur ulang (reset) Arduino ke program terakhir apabila terjadi keanehan pada eksekusi yang dilakukan. Pin analog digunakan untuk menerima keluaran sensor analog. Pada pin-pin ini terdapat ADC (*Analog Digital Converter*) yang berfungsi mengubah nilai analog menjadi digital. Pin analog ini diperlukan karena proses yang terjadi dalam Arduino adalah proses digital. Pin digital digunakan untuk menerima keluaran dari sensor digital. Pada pin digital terdapat dua kondisi yaitu (*high*) dan 0 (*low*).

Arduino Uno sendiri menggunakan *chip* mikrokontroler ATmega 328 sebagai 'otak' pusat pengendaliannya *chip* ini dipasangkan ke *header socket* yang ada pada board Arduino Uno sehingga memungkinkan untuk dilepas dan dipasang ulang. Kelebihannya adalah apabila terjadi kerusakan pada chip yang diakibatkan oleh kegagalan ataupun 'cedera fisik' (jatuh atau terbanting) maka

anda dapat melakukan penggantian *chip* yang rusak tersebut dengan chip yang baru. Jadi anda tidak perlu mengganti *board* Arduino Uno anda secara keseluruhan sehingga dapat menghemat biaya kerusakan. Anda dapat melihat ilustrasi chip ATmega 328 dan konfigurasi pada Gambar 2.8.

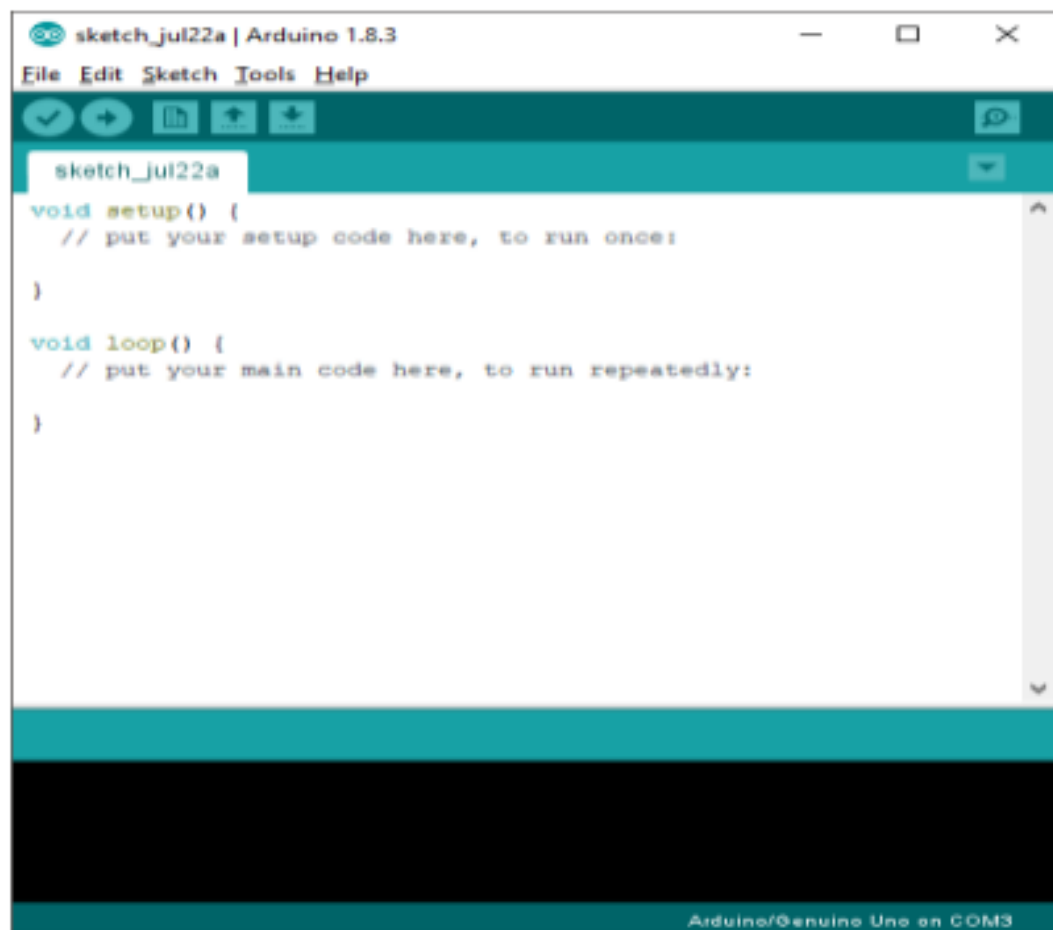


Gambar 2.8. ATmega 328

Sumber : <https://32933-toi.blogspot.com/2018/09/blog-post.html>.

Pemrograman Arduino dapat dilakukan dengan Arduino *software Integrated Development Environment*(IDE).IDE berfungsi untuk menulis dan mengunggah kode dari computer ke Arduino.Adapun bahasa pemrograman yang dipakai oleh Arduino IDE ini adalah bahasa pemrograman C dan C++. Berikut tampilan *software IDE* dari arduino dapat dilihat pada Gambar 2.9. (Risal. A, 2017)





Gambar 2.9. Tampilan IDE

Sumber: (Effenberger, F., & Kiefer, G)

Arduino IDE digunakan sebagai tempat membuat perintah (program) atau yang dikenal dengan istilah *source code*, melakukan pengecekan kesalahan, kompilasi, *upload* program ke Arduino, dan menguji hasil kerja Arduino melalui monitor Komputer (Effenberger, F., & Kiefer, G).

## 2.5. ATmega 328

ATmega 328 adalah mikrokontroler keluaran dari atmel yang mempunyai arsitektur RISC (*Reduce Instruction Set Computer*) yang dimana setiap proses eksekusi data lebih cepat dari pada arsitektur CISC (*Completed Instruction Set Computer*).

Mikrokontroler ATmega 328 memiliki arsitektur Harvard, yaitu memisahkan memori untuk kode program dan memori untuk data sehingga dapat memaksimalkan kerja. Instruksi – instruksi dalam memori program dieksekusi dalam satu alur tunggal, dimana pada saat satu instruksi dikerjakan instruksi berikutnya sudah diambil dari memori program. Konsep inilah yang memungkinkan instruksi – instruksi dapat dieksekusi dalam setiap satu siklus *clock*. 32 x 8-bit register serba guna digunakan untuk mendukung operasi pada ALU ( *Arithmetic Logic unit* ) yang dapat dilakukan dalam satu siklus. 6 dari register serbaguna ini dapat digunakan sebagai 3 buah register pointer 16-bit pada mode pengalamatan tak 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 ).

ATMega328 Secara umum, AVR dapat dikelompokkan menjadi tiga kelas, yaitu TinyAVR, AT90Sxx, dan ATmega. Pada dasarnya yang membedakan masing-masing kelas adalah memori, peripheral, dan fungsinya. Dari segi arsitektur dan instruksi yang digunakan, mereka bisa dikatakan hampir sama.

Tabel 2.1 Spesifikasi keluarga AVR(*Automatic Voltage Regulator*).

Mikrokontroler AVR		Memori		
Jenis	Jumlah Pin	Flash (Byte)	EEPROM (Byte)	SRAM (Byte)
TinyAVR	8-32	1-2K	64-128	0-128
AT90Sxx	20-44	1-8K	128-512	0-1K
ATMega	32-64	8-128K	512-4K	512-4K

Adapun Fitur AVR dari ATmega 328 yaitu sebagai berikut dibawah ini:

1. 130 macam instruksi yang hampir semuanya dieksekusi dalam satu siklus clock.
2. 32 x 8-bit register serba guna.
3. Kecepatan mencapai 16 MIPS dengan clock 16 MHz.
4. 32 KB *Flash* memory dan pada arduino memiliki *bootloader* yang menggunakan 2 KB dari *flash* memori sebagai *bootloader*.
5. Memiliki EEPROM (*Electrically Erasable Programmable Read Only Memory*) sebesar 1KB sebagai tempat penyimpanan data semi permanen karena EEPROM tetap dapat menyimpan data meskipun catu daya dimatikan.
6. Memiliki SRAM (*Static Random Access Memory*) sebesar 2KB.
7. Memiliki pin I/O digital sebanyak 14 pin 6 diantaranya PWM (*Pulse Width Modulation*) output.
8. Master / *Slave SPI Serial interface*.

Selain register serba guna di atas, 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.

### 2.5.1 Spesifikasi ATmega328

Tabel 2.2. spesifikasi ATmega 328

Jumlah Pin	28
CPU	RISC 8-Bit AVR
Tegangan Operasi	1.8 to 5.5 V
Memori Program	32KB
Jenis Memori Program	<i>Flash</i>
SRAM	2048 Bytes
EEPROM	1024 Bytes
ADC	10-Bit
Jumlah Saluran ADC	8
Pin PWM	6
Pembanding	1
Paket (4)	8-pin PDIP 32-lead TQFP 28-pad QFN/MLF 32-pad QFN/MLF
Osilator	<i>up to 20 MHz</i>
<i>Timer</i> (3)	8-Bit x 2 & 16-Bit x 1
Penyetelan ulang pengaktifan yang ditingkatkan	Ya
Nyalakan <i>Timer</i>	Ya
Pin I/O	14
Pabrikan	Microchip
SPI	Ya
I2C	Ya
<i>Timer</i> Pengawas	Ya

Deteksi Brownout (BOD)	Ya
Mengatur ulang	Ya
USI ( <i>Universal Serial Interface</i> )	Ya
Suhu Operasional Minimum	-40 C to +85 C

### 2.5.2. Mikrokontroler ATmega328

Mikrokontroler ATmega328 memiliki 14 *input* digital *output* pin/(6 *output* PWM), 6 *input* analog, 16 MHz osilator kristal, koneksi serial, ICSP *header*, dan tombol reset. Ini berisi semua fitur yang diperlukan untuk mendukung mikrokontroler, cukup hubungkan ke komputer dengan kabel USB to Serial atau listrik AC yang ke adaptor DC/baterai untuk memulai.

### 2.5.3. Konstruksi mikrokontroler ATmega 328

Mikrokontroler ATmega328 memiliki 3 jenis memori, yaitu memori program, memori data dan memori EEPROM. Ketiganya memiliki ruang sendiri dan terpisah.

#### 1. Memori program

ATmega328 memiliki kapasitas memori program sebesar 8K byte yang terpetakan dari alamat 0x0000 – 0x3FFF dimana masing-masing alamat memiliki lebar data 32 bit. Memori program ini terbagi menjadi 2 bagian yaitu bagian program *boot* dan bagian program aplikasi.

#### 2. Memori data

Memori data ATmega328 terbagi menjadi 3 bagian yaitu register

serba guna, register I/O dan SRAM. ATmega328 memiliki 32 register serba guna, 64 register I/O yang dapat diakses sebagai bagian dari memori RAM (menggunakan instruksi LD atau ST) atau dapat juga diakses sebagai I/O (menggunakan instruksi IN atau OUT), dan 2048 byte memori data SRAM.

### 3. Memori EEPROM

ATmega328 memiliki memori EEPROM sebesar 1K byte yang terpisah dari memori program maupun memori data. Memori EEPROM ini hanya dapat diakses dengan menggunakan register-register I/O yaitu register EEPROM Address, register EEPROM Data, dan register EEPROM Control. Untuk mengakses memori EEPROM ini diperlakukan seperti mengakses data eksternal, sehingga waktu eksekusinya relatif lebih lama bila dibandingkan dengan mengakses data dari SRAM.

#### 2.5.4. *Input & Output*

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

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

atau perubahan nilai. Lihat *attachInterrupt* () fungsi untuk rincian.

3. PWM: 3, 5, 6, 9, 10, dan 11. Menyediakan 8-bit *output* PWM dengan *analogWrite* () fungsi.
4. SPI: 10 (SS), 11 (mosi), 12 (MISO), 13 (SCK). Pin ini mendukung komunikasi SPI menggunakan perpustakaan SPI
5. LED: 13. Ada *built-in* LED terhubung ke pin digital 13. Ketika pin adalah nilai TINGGI, LED menyala, ketika pin adalah RENDAH, itu *off*.

Uno memiliki 6 input analog, diberi label A0 melalui A5, masing-masing menyediakan 10 bit resolusi yaitu 1024 nilai yang berbeda. Secara default sistem mengukur dari tanah sampai 5 volt.

1. TWI: A4 atau SDA pin dan A5 atau SCL pin. Mendukung komunikasi TWI.
2. Aref. Referensi tegangan untuk input analog. Digunakan dengan *analogReference* ().
3. Reset.

Lihat juga pemetaan antara pin Arduino dan ATmega328 port. Pemetaan untuk ATmega8, 168 dan 328 adalah identik (Nasir, S. Z., 2017)

Deskripsi pin ATmega328

1. Fungsi yang terkait dengan pin harus diketahui agar dapat menggunakan perangkat yang tepat.
2. Pin ATmega-328 dibagi menjadi port yang berbeda yang diberikan

secara rinci dibawah ini

3. **VCC** adalah suplai tegangan digital
4. **AVCC** adalah pin tegangan suplai untuk konverter analog ke digital.
5. **GND** menunjukan Ground dan memiliki 0V

#### 6. Port B

Port B adalah jalur data 8 bit dan difungsikan sebagai *input / output*,

Selain itu PORTB dapat memiliki fungsi alternatif seperti di bawah ini :

- a. ICP1 (PB0), sebagai *Timer Counter 1 input capture* pin.
- b. OC1A (PB1), OC1B (PB2) dan OC2 (PB3), sebagai keluaran PWM (Pulse Width Modulation).
- c. MOSI (PB3), MISO (PB4), SCK (PB5), SS (PB2), sebagai jalur komunikasi SPI.
- d. Selain itu pin ini juga berfungsi sebagai jalur pemograman serial (ISP).
- e. TOSC1 (PB6) dan TOSC2 (PB7), sebagai sumber *clock external* untuk timer.
- f. XTAL1(PB6) dan XTAL2 (PB7), sebagai sumber *clock* utama mikrokontroler.

#### 7. Port C

Port C adalah jalur data 7 bit dan difungsikan sebagai input / output digital, Fungsi alternatif PORTC antara lain sebagai berikut :

- a. ADC6 channel (PC0,PC1,PC2,PC3,PC4,PC5) dengan resolusi sebesar 10 bit. ADC dapat digunakan untuk mengubah *input* yang berupa tegangan analog menjadi data digital.



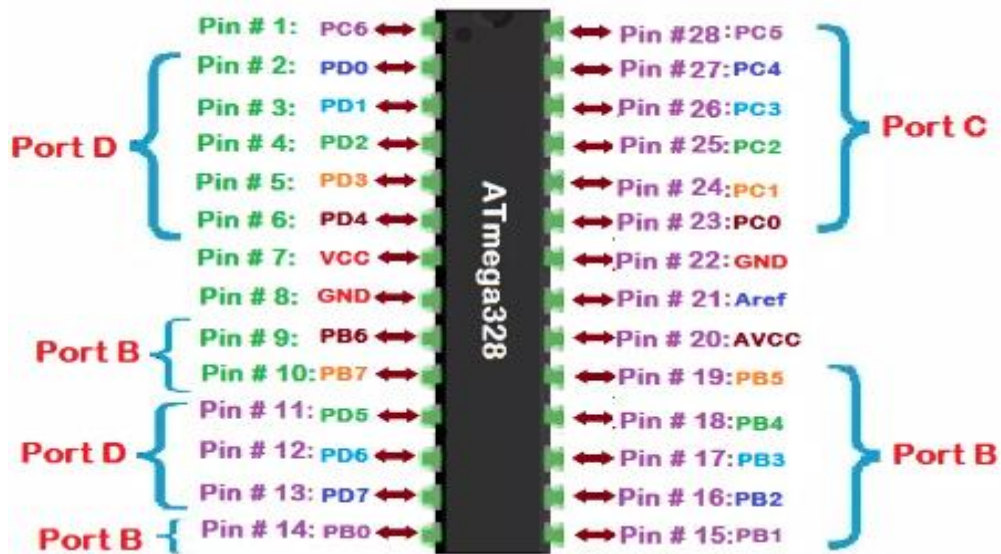
- b. I2C (SDA dan SDL) adalah salah satu fitur terdapat pada PORTC. I2C digunakan untuk komunikasi dengan sensor atau *device* lain yang memiliki komunikasi data tipe I2C seperti sensor kompas, *accelerometer nunchuck*.

## 8. Port D

Port D adalah jalur data 8 bit masing-masing pin dapat difungsikan sebagai *input / output*. Sama dengan Port B, Port C, Port D juga memiliki fungsi alternatif dibawah ini yaitu :

- a. USART (TXD dan RXD) merupakan jalur data komunikasi serial dengan level sinyal TTL. Pin TXD berfungsi untuk mengirimkan data serial, sedangkan RXD kebalikannya yaitu sebagai pin yang berfungsi untuk menerima data serial.
- b. Interrupt (INT0 dan INT1) adalah pin dengan fungsi khusus sebagai interupsi *hardware*. Interupsi ini 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 difungsikan sebagai sumber *clock external* untuk USART, namun dapat memanfaatkan *clock* dari CPU, sehingga tidak perlu membutuhkan *external clock*.
- d. T0 & T1 masukkan counter external *timer 1 & timer 0*.
- e. AIN0 & AIN1 keduanya merupakan masukkan *input* untuk analog comparator.

9. AREF adalah pin referensi analog untuk konverter analog ke digital.



Gambar 2.10. ATmega 328 ports

Sumber : <https://www.theengineeringprojects.com/2017/08/introduction-to-atmega328.html>

### 2.5.5. Kelebihan Arduino

Arduino menyederhanakan proses bekerja dengan mikrokontroler, sekaligus menawarkan berbagai macam kelebihan antara lain :

1. *Murah.* Papan (perangkat keras) Arduino biasanya dijual relatif murah dibandingkan dengan *platform* mikrokontroler pro lainnya.
2. *Sederhana dan mudah pemrogramannya.* Perlu diketahui bahwa lingkungan pemrograman Arduino mudah digunakan untuk pemula, dan cukup fleksibel bagi mereka yang sudah tingkat lanjut.
3. *Perangkat lunaknya Open Source.* Perangkat lunak Arduino IDE dipublikasikan sebagai *open source*, tersedia bagi para pemrograman berpengalaman untuk pengembangan lebih lanjut. Bahasanya bisa dikembangkan lebih lanjut melalui pustaka-pustaka C++ yang berbasis pada bahasa C untuk AVR

4. *Perangkat kerasnya Open Source.* Perangkat keras Arduino berbasis mikrokontroler ATMEGA8, ATMEGA168, ATMEGA328 dan ATMEGA1280.
5. *Tidak perlu perangkat chip programmer.* Karena di dalamnya sudah ada *bootloader* yang akan mengenai *upload* program dari computer.
6. Sudah memiliki sarana komunikasi USB. Sehingga pengguna laptop yang tidak memiliki *port serial/RS323* bisa menggunakannya.
7. Bahasa pemrograman reatif mudah. Karena *software* Arduino dilengkapikumpulan *library* yang cukup lengkap.
8. Memiliki modul siap pakai yang bisa ditancapkan pada *board* Arduino. Misalnya *shield* GPS, Ethernet, SD Card, dan lain-lain.( Nasir, S. Z. (2017)).

## 2.6. Modul ESP8266

ESP8266 adalah sebuah modul *WiFi* yang akhir-akhir ini semakin digemari para *hardware developer* dapat dilihat pada Gambar 2.11. Selain karena harganya yang sangat terjangkau, modul *WiFi* serbaguna ini sudah bersifat SOC (*System on Chip*), sehingga kita bisa melakukan *programming* langsung ke ESP8266 tanpa memerlukan mikrokontroler tambahan. Kelebihan lainnya, ESP8266 ini dapat menjalankan peran sebagai *ad hoc* akses poin maupun klien sekaligus. (Sinuardino,2016).



Gambar 2.11. Modul ESP8266

Sumber : <https://auftechnique.com/cara-setup-esp8266-pada-arduino-ide/>

ESP8266 dikembangkan oleh pengembang asal china yang bernama “espressif”. Pada ESP8266 sendiri sudah memiliki GPIO ( *General Purpose Input Output*), yang artinya ESP8266 ini bisa melakukan fungsi *input* atau *output*. Salah satu kelebihan ESP8266 adalah memiliki *DEEP SLEEP MODE*, sehingga penggunaan akan lebih efisien dalam hal sumber daya. ESP-01 memiliki tiga jenis mode operasi yang perlu diketahui, yaitu *Station*, *Access Point* dan gabungan mode keduanya. jika yang dipilih adalah mode AP ( *Access Point*), berarti ESP-01 difungsikan sebagai akses point *wifi* (memiliki SSID sendiri), sehingga perangkat lain bisa terhubung dengan ESP-01. Mode ini mirip dengan *wifi* tathering yang dimiliki oleh smartphone. Namun, jika mode STA yang dipilih, ESP-01 dapat terhubung dengan jaringan *wifi* yang tersedia oleh akses poin dari router, ataupun modem *Mifi* (seperti yang disediakan oleh *provider bolt* atau *andromax*, misalnya). Sehingga ESP-01 otomatis terhubung dengan jaringan internet, tentu jika ada internet aktif di jaringan tersebut. Sedangkan mode yang ketiga adalah gabungan dari AP dan STA. Konfigurasi ESP8266 sebagai *client* dan *access point* dilakukan dalam mode *AT Command*. Pengklasifikasian *AT Command* dapat dilihat pada Gambar 2.12 (Sibagariang, M. E., 2018).



Gambar 2.12. Pengklasifikasian *AT command*  
 Sumber: (Sibagariang, M. E., 2018)

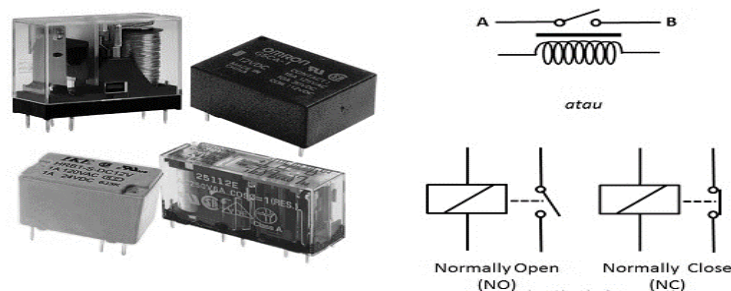
*AT Command* digunakan untuk berkomunikasi dengan terminal melalui port pada Komputer, dan penggunaan *AT command* pada ESP8266 dapat

memberi kemudahan untuk mengetahui:

- a. Mengetahui versi *Firmware* (AT+GMR).
- b. Menampilkan *List Access Point* (AT+CWLAP).
- c. Mengubungkan dengan *Access Point* (AT+CWJAP="SSID", "PASSWORD").
- d. Memutuskan hubungan dengan *Access point* (AT+CWQAP).
- e. Mendapatkan dan mengetahui IP Address (AT+CIFSR).

## 2.7. Relay

*Relay* adalah Saklar (*Switch*) yang dioperasikan secara listrik dan merupakan komponen Electromechanical (Elektromekanikal) yang terdiri dari 2 bagian utama yakni Elektromagnet (*Coil*) dan Mekanikal (seperangkat Kontak Saklar/*Switch*). *Relay* menggunakan Prinsip Elektromagnetik untuk menggerakkan Kontak Saklar sehingga dengan arus listrik yang kecil (*low power*) dapat menghantarkan listrik yang bertegangan lebih tinggi. Sebagai contoh, dengan *Relay* yang menggunakan Elektromagnet 5V dan 50 mA mampu menggerakkan *Armature Relay* (yang berfungsi sebagai saklarnya) untuk menghantarkan listrik 220V 2A.



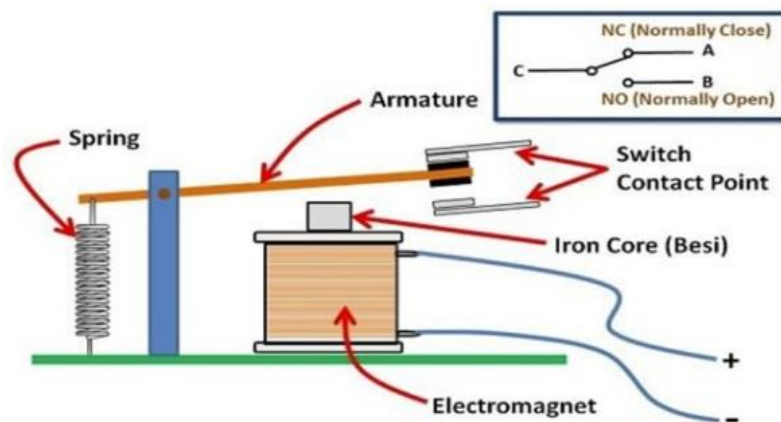
Gambar 2.13. Gambar Dan Simbol *Relay*  
 Sumber: (Baharuddin & Abd rahman, 2019)

Kontak point pada *relay* ada dua yaitu *Normally open* yaitu keadaan dimana

kondisi awal sebelum diaktifkan *open* (terbuka). dan *Normally close* yaitu kondisi awal sebelum diaktifkan berada pada posisi *close* (tertutup).

Pada dasarnya, *Relay* terdiri dari 4 komponen dasar yaitu :

1. *Electromagnet* (Coil)
2. *Armature*
3. *Switch Contact Point* (Saklar)
4. *Spring*.



Gambar 2.14. Struktur Sederhana Relay

Sumber : <https://balagia.blogspot.com/2021/01/relay-struktur-dan-simbol-relay.html>

Berdasarkan Gambar 2.14 diatas, sebuah Besi (*Iron Core*) yang dililit oleh sebuah kumparan *Coil* yang berfungsi untuk mengendalikan Besi tersebut. Apabila Kumparan *Coil* diberikan arus listrik, maka akan timbul gaya Elektromagnet yang kemudian menarik *Armature* untuk berpindah dari Posisi sebelumnya (*NC*) ke posisi baru (*NO*) sehingga menjadi Saklar yang dapat menghantarkan arus listrik di posisi barunya (*NO*). Posisi dimana *Armature* tersebut berada sebelumnya (*NC*) akan menjadi *OPEN* atau tidak terhubung. Pada saat tidak dialiri arus listrik, *Armature* akan kembali lagi ke posisi Awal

(NC). *Coil* yang digunakan oleh Relay untuk menarik *Contact Poin* ke Posisi *Close* pada umumnya hanya membutuhkan arus listrik yang relatif kecil. (Baharuddin & Abd rahman, 2019)

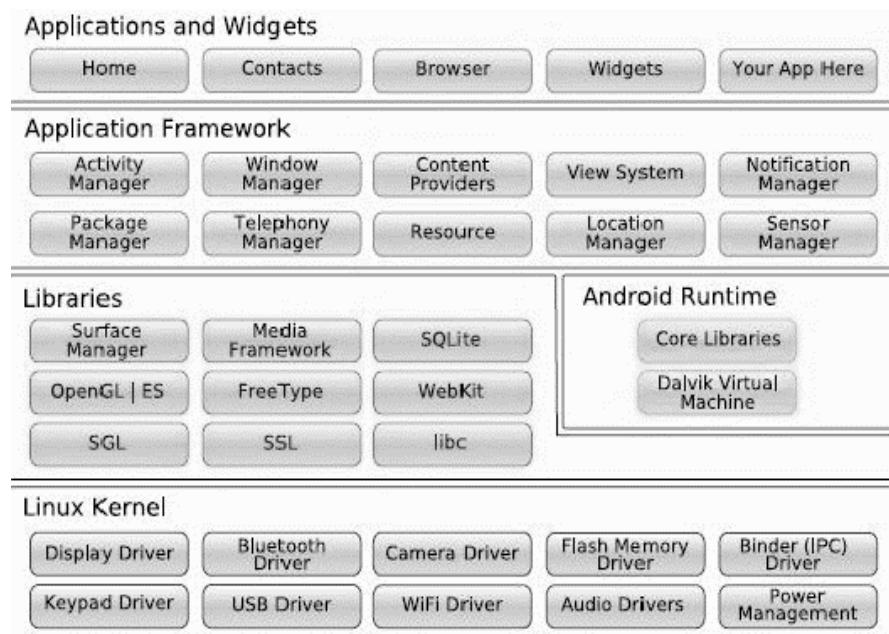
## **2.8. Android**

*Android* adalah kumpulan perangkat lunak yang ditujukan bagi perangkat bergerak mencakup sistem operasi, *middleware*, dan aplikasi kunci. *Android Standart Development Kit* (SDK) menyediakan perlengkapan dan *Application Programming Interface* (API) yang diperlukan untuk mengembangkan aplikasi pada *platform Android* menggunakan bahasa pemrograman Java.

*Android* dikembangkan oleh Google bersama *Open Handset Alliance* (OHA) yaitu aliansi perangkat selular terbuka yang terdiri dari 47 perusahaan *Hardware, Software* dan perusahaan telekomunikasi ditujukan untuk mengembangkan standar terbuka bagi perangkat selular.

Pada mulanya terdapat berbagai macam sistem operasi pada perangkat selular, diantaranya sistem operasi *Symbian, Microsoft Windsos Mobile, Mobile Linux, iPhone*, dan sistem operasi lainnya. Namun diantara sistem operasi yang ada belum mendukung standar dan penerbitan API yang dapat dimanfaatkan secara keseluruhan dan dengan biaya yang murah. Kemudian Google ikut berkecimpung didalamnya dengan *platform Android*, yang menjanjikan keterbukaan, keterjangkauan, *open source*, dan *framework* berkualitas. Dalam paket sistem operasi *Android* terdiri dari beberapa unsur seperti tampak pada Gambar 2.15. Secara sederhana arsitektur *Android* merupakan sebuah kernel Linux dan sekumpulan pustaka C / C++ dalam suatu *framework* yang menyediakan dan

mengatur alur proses aplikasi. 7



Gambar 2.15. Arsitektur *Android7* (Sibagariang & Morris Egmon, 2018)

Di lapisan terbawah Arsitektur *Android* terdapat Linux Kernel, lapisan ini merupakan jantung dari seluruh sistem yang ada di *Android* karena lapisan ini memberikan fungsi menyediakan *Driver* Layar, Kamera, *Keypad*, *WiFi*, *Flash Memory*, *Audio* Dll. Diatas Linux Kernel ada *Library* yang berfungsi membawa sekumpulan instruksi untuk mengarahkan perangkat *Android* kita dalam menangani berbagai tipe data. Contohnya, perekam dari berbagai macam format Video dan Audio ditangani oleh *Media Framework Library*. Di lapisan ini terdapat *Android Runtime* yang memiliki *Core Libraries* yang menyediakan sebagian besar fungsi *android*. Inilah yang membedakan sistem operasi lain yang juga meng implementasikan Linux, karena terdapat sekumpulan *Library* Java yang dikhususkan untuk *Android*. Dalam lapisan *Android Runtime* juga terdapat *Dalvik VM (Virtual Machine)* yang merupakan sejenis *Java Virtual Machine* yang didesain khusus dan Dioptimalkan untuk *android*. *Dalvik VM*



menggunakan fitur linux untuk membuat setiap Aplikasi *Android* dapat berjalan dengan prosesnya sendiri. Lapisan berikutnya adalah *Application Framework* yang berhubungan langsung dengan aplikasi.program-program di atas manajemen fungsi dasar dari perangkat seperti Manajemen *Resource*, Manajemen Panggilan, Manajemen Window dll. Sebagai seorang *Developer*, lapisan ini dapat dilihat sebagai alat dasar yang digunakan untuk mengembangkan aplikasi. *Application Layer* berada pada lapisan terluar dari Arsitektur Android, pengguna Android akan berinteraksi di Lapisan ini. Memiliki fungsi umum seperti menelepon, mengakses *website*, da (Sibagariang & Morris Egmon, 2018)

## **2.9. Bahasa Pemrograman C**

Menurut Iswanto (2011) bahasa C adalah bahasa pemrograman yang dapat dikatakan berada di bahasa beraras tinggi. Bahasa beraras rendah artinya bahasa yang berorientasi pada mesin, sedangkan beruas tinggi berorientasi pada manusia. Bahasa beraras rendah, misalnya bahasa assembler. ditulis dengan sandi yang hanya dimengerti oleh mesin sehingga hanya digunakan bagi yang memrogram mikrokontroler. Bahasa beruas rendah merupakan bahasa yang membutuhkan kecermatan tinggi bagi pemrogram karena perintahnya harus rinci, ditambah lagi masingmasing pabrik mempunyai sandi perintah sendiri. Bahasa tinggi relatif mudah digunakan karena ditulis dengan bahasa manusia sehingga mudah. dimengerti dan tidak tergantung mesinnya. Bahasa beraras tinggi umumnya digunakan pada komputer.

Pencipta bahasa C adalah Brian Wr Kemighan dan Denis M. Ritchi sekitar 1972. Penulisan program dalam bahasa C dilakukan dengan membagi dalam

blok-blok sehingga bahasa C disebut bahasa terstruktur. Bahwa C dapat di berbagai mesin dengan mudah, mulai dari PC sampai mainframe, serta menggunakan berbagai sistem operasi misalnya DOS, UNIX, VMS, dan lain-lain.

### 2.9.1 Penulisan Bahasa C Arduino

Penulisan program bahasa C di Arduino:



```

sketch_nov03a$
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}

```

Gambar 2.16 : Contoh Penulisan Program Pada C .

*Sumber:* (Risal .A, 2017)

Arduino Penjelasan tentang fungsi setiap sintak pada Arduino hampir sama dengan AVR. Beberapa bagian yang membedakannya adalah sebagai berikut:

1. "void main(void)" menjadi "void setup()": penanda atau sebagai pe- nunjuk isi dari program dalam pemrograman

AVR, isi dari program dimulai dari tanda kurung kurawal ({}  
dan diakhiri pula dengan tandatutup kurung kurawal ({})

2. perintah "*while(1)*" sendiri juga dapat digunakan di arduino,  
tapi seca-ra khusus aplikasi Arduino sudah menyediakan tempat  
untuk menulis program yang memiliki karakteristik berulang  
(looping) yang tertulis "*void loop()*"

Secara Prinsip, *sketch* selalu melibatkan dua fungsi, yaitu *setup()* dan *loop()*.

Kode Sketch minimal dapat dilihat pada gambar 2.16. Baris yang  
mengandung *void* dan nama fungsi adalah judul fungsi, sedangkan bagian { }  
dinamakan tubuh fungsi. Semua definisi fungsi melibatkan judul fungsi dan  
tubuh fungsi. Menurut Kadir fungsi sendiri adalah deretan instruksi yang  
diberi suatu anam, umumnya fungsi memberikan nilai ketika dipanggil. Nilai  
yang diberikan tersebut dinamakan nilai balik.( A. Risal , 2017)