

BAB II

DASAR TEORI

2.1. JARINGAN LOKAL AKSES FIBER (JARLOKAF)

Jaringan Lokal Akses Fiber adalah jaringan lokal yang menggunakan kabel serat optik untuk menghubungkan antara sentral lokal dengan terminal pelanggan. Kabel serat optik dijadikan sebagai media transmisi menggantikan kabel tembaga karena mempunyai beberapa karakteristik, seperti *bandwidth* lebih lebar, redaman lebih rendah, kapasitas kanal yang disediakan jauh lebih besar, tahan terhadap interferensi elektromagnetik, dan mampu menyalurkan informasi dengan kecepatan yang lebih tinggi. Terdapat beberapa teknologi yang mendukung penggunaan *fiber optic* dalam jaringan lokal antara lain :

1. *Digital Loop Carrier* (DLC)

Digital Loop Carrier (DLC) atau biasa disebut dengan *Optic Network Unit* (ONU) adalah salah satu elemen perangkat suatu jaringan telekomunikasi telepon dimana masing-masing pelanggan akan disambungkan pada suatu sentral telepon yang disebut dengan *local loop*.

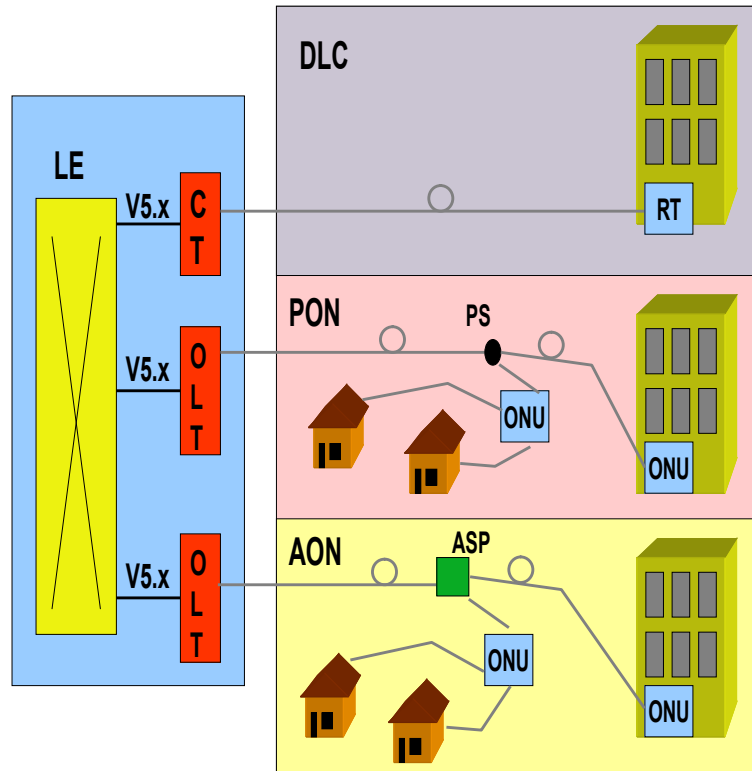
2. *Optical Network Unit* (ONU)

Optical Network Unit (ONU) merupakan perangkat yang tersebar dan berada di dekat lokasi pelanggan, ke arah saluran pelanggan. ONU berfungsi mengubah kembali sinyal optik menjadi sinyal elektrik untuk disalurkan ke pelanggan.

3. *Optical Access Network* (OAN)

Optical Access Network (OAN) merupakan sistem yang membagi *optical/electrical* pada perangkat Jarlokaf diantara beberapa pelanggan dengan membawa fiber sedekat mungkin dengan pelanggan. OAN menggunakan arsitektur *multiple star* atau *point to multipoint* dengan titik pertama berada di sentral dan titik berikutnya berada di daerah antara sentral dengan pelanggan. Teknologi OAN dibagi ke dalam dua kelompok yaitu *Passive Optical Network* (PON) dan *Active Optical Network* (AON). *Passive Optical Network* merupakan sistem yang membagi intensitas

sinyal optik melalui *Passive Splitter* (PS) sedangkan *Active Optical Network* (AON) merupakan sistem yang menggunakan perangkat aktif untuk membagi dan mengkombinasikan sinyal optik.[1]



Gambar 2.1 Beberapa Tipe Jaringan Lokal Akses Fiber[1]

2.2. TEKNOLOGI *MULTI SERVICE ACCESS NODE* (MSAN)

Multi Service Access Network merupakan sebuah jaringan akses yang memberikan layanan *mutiservice* sejalan dengan perkembangan *Next Generation Network* (NGN). MSAN dapat memberikan dan menyediakan layanan berupa *broadband* akses *multiplexer* sebagai IP DSLAM berdasarkan pada teknologi IP, ATM, dan TDM serta layanan *narrowband* dalam jaringan PSTN melalui kabel tembaga atau fiber optic. MSAN juga diharapkan mampu untuk menyediakan sebuah solusi layanan berbasis jaringan lokal akses fiber atau tembaga dengan *cost-effective* pada suatu layer jaringan yang konvergen dimana untuk layanan PSTN, NGN, dan jaringan *broadband* berada pada daerah yang sama.[1]

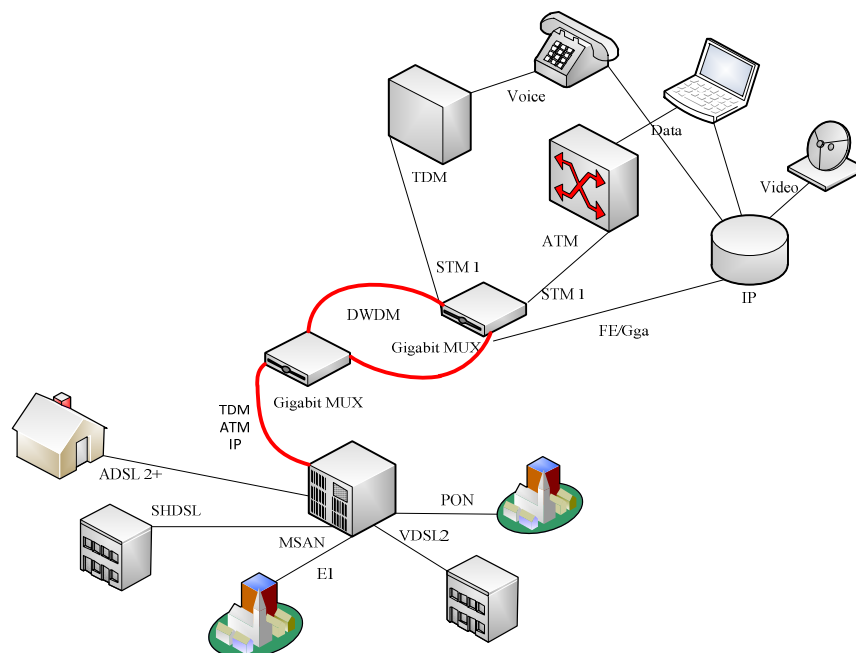
Topologi yang digunakan MSAN yaitu topologi *stacking* (bertingkat) atau *master slave architecture* yang artinya *node slave* digunakan untuk

perpanjangan tangan dari master. Bentuk konfigurasi MSAN diilustrasikan seperti pada gambar 2.7. *Multi Service Access Network* (MSAN) merupakan generasi ketiga dari teknologi OAN, sehingga mampu memberikan berbagai macam jenis layanan diantaranya yaitu:[1]

- a. Melalui fungsi *Access Gateway* yang bisa langsung terhubung ke *Softswitch* yang disebut juga *Voice H.248/Megaco*.
- b. Melakukan fungsi layanan *Broadband Access Multiplexer* melalui ADSL, ADSL2/2+,G.SHDSL, *Annex M*.
- c. Memberikan layanan *Triple Play*, layanan *High Speed Internet Access* (HSIA), layanan *Voice packet* dan layanan IPTV secara bersamaan melalui infrastruktur yang sama. Layanan *Triple Play* MSAN antara lain *voice* (layanan VoIP gateway), data (melayani data internet kecepatan tinggi), dan video (jasa televisi berbasis IP Telkom).

Fungsi MSAN antara lain :

- a. Sebagai sistem akses *broadband*.
- b. Sebagai akses gateway dalam *Next Generation Network* (NGN).
- c. Sebagai jaringan akses tradisional PSTN.
- d. Menyediakan layanan berbasis jaringan lokal akses fiber dan jaringan lokal akses tembaga.



Gambar 2.2 Konfigurasi MSAN[1]

Konfigurasi Jaringan MSAN terdiri dari.[1]

- a. *Asymmetric Digital Subscriber Line 2+ (ADSL2+)* adalah teknologi data berkecepatan tinggi yang dikirim melalui kabel telepon dan memungkinkan untuk menerima data sampai kecepatan 24 Mbps untuk *downstream* dan kecepatan 3 Mbps untuk *upstream*.
- b. *Symmetric High Data Rate Digital Subscriber Line (SHDSL)* merupakan perkembangan dari HDSL dan menggunakan teknologi xDSL. Dalam aplikasinya menggunakan 1 *pair* atau 2 *pair* tembaga yang mempunyai kecepatan *upstream* dan *downstream* sebesar 2.320 Mbps. Teknologi SHDSL ini mulai diluncurkan sekitar tahun 2001.
- c. *Very High Speed Digital Subscriber Line 2 (VSDL2)* adalah jaringan yang mempunyai kecepatan super cepat dengan kecepatan mencapai 300 Mbps untuk data transfer dengan jangkauan 100 Mbps untuk jarak 400 meter. Teknologi ini diluncurkan pada tahun 2005.
- d. *Passive Optical Network (PON)* adalah jaringan *point to multipoint* yang menggunakan kabel fiber optik yang mempunyai elemen untuk membagi optik (*optical splitter*) melalui *passive spliter* sebagai penyalur data untuk beberapa tujuan.

Dense Wavelength Division Multiplexing (DWDM) adalah teknologi dengan memanfaatkan sistem *Synchoronous Digital Hierarchy (SDH)* yang sudah ada dengan memultiplekskan sumber – sumber sinyal yang ada.[1]

2.3. STANDAR **BLUETOOTH**

Teknologi *bluetooth* dibuat pada tahun 1994, teknologi *Bluetooth* disebut sebagai alternatif nirkabel untuk kabel data dengan bertukar data menggunakan transmisi radio. Nama *Bluetooth* berasal dari abad kesepuluh Raja Denmark, Harald Blatand atau, dalam bahasa Inggris, Harold Bluetooth. Seperti ceritanya, Raja Blatand membantu menyatukan faksi di bagian Norwegia, Swedia dan Denmark. Demikian pula, teknologi Bluetooth diciptakan sebagai standar terbuka untuk memungkinkan konektivitas dan kolaborasi antara produk yang berbeda dan industri.[2]

Pengembangan standar industri *Bluetooth* dimulai pada akhir musim dingin 1998 ketika *Ericsson*, *IBM*, *Intel*, *Nokia*, dan *Toshiba* membentuk *Bluetooth Special Industry Group* (SIG) untuk mengembangkan dan mempromosikan solusi global untuk jarak pendek nirkabel komunikasi yang beroperasi pada pita frekuensi 2,4 GHz ISM (industrial, ilmiah, medis).

Tabel 2.1 Kronologi *Bluetooth*[3]

No.	Waktu Rilis	Perkembangan
1	Mid '90s	Early seeds of the Bluetooth radio technology.
2	Feb'98	The Bluetooth SIG is formed by the five promoter companies.
3	March'98	First face-to-face technical meeting of the software task force. The "hardware" task force, called the air group, has been meeting prior to the formation of the SIG.
4	May'98	The SIG announced publicly its existence and its objectives.
5	Oct'98	The first Bluetooth Developers Conference takes place in Atlanta, GA, USA.
6	July'99	The Bluetooth spec. ver. 1.0A is release to the general public.
7	July'99	The Bluetooth SIG submits a proposal to the newly formed IEEE 802.15 working group on wireless personal area networks.
8	Dec'99	The promoter group increases to nine.
9	Dec'99	The Bluetooth spec. ver. 1.0B is release to the general public.
10	Jan'00	New working groups start to be formed to develop additional Bluetooth specifications
11	Feb'01	The Bluetooth SIG is incorporated in the Bluetooth SIG, Inc.
12	Feb'01	The Bluetooth spec. ver. 1.1 is release to the general public.

Spesifikasi *Bluetooth* dikembangkan oleh lima promotor (pendiri) anggota dengan kontribusi besar dari dua perusahaan tambahan, *3Com* dan *Motorola*. Pada keluaran pertama teknologi *bluetooth* memiliki spesifikasi ver. 1.0 A yang terdiri dari dua bagian berikut :

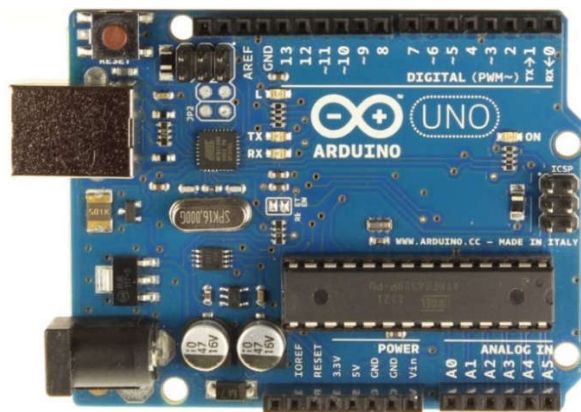
- Inti spesifikasi mendefinisikan karakteristik radio dan komunikasi protokol untuk pertukaran data antar perangkat melalui link radio Bluetooth.
- Spesifikasi profil yang mendefinisikan bagaimana protokol Bluetooth akan digunakan untuk menyadari sejumlah aplikasi yang dipilih.

Pada bulan Desember 1999, kelompok promotor meningkat dari 5 sampai dengan 9 dengan penambahan berupa *3Com, Lucent, Microsoft, dan Motorola*. Pada awal tahun 2001, Agere, sebuah *Lucent spin-off* terdiri dari divisi mikroelektronika, telah mengambil tempat *Lucent* di kelompok promotor.[2]

2.4. PERANGKAT PENYUSUN *HARDWARE*

2.4.1 Arduino UNO

Arduino Uno adalah sebuah *board* mikrokontroler berdasarkan ATmega328. Uno memiliki 14 pin masukan/keluaran digital (yang mana 6 dari 14 tersebut dapat digunakan sebagai keluaran PWM), 6 masukan analog, sebuah resonator keramik 16MHz, sebuah koneksi USB, sebuah power jack, sebuah header ICSP, dan sebuah tombol reset.



Gambar 2.3 *Board* Arduino UNO[4]

Arduino UNO memiliki semua yang dibutuhkan untuk mendukung mikrokontroler, dengan mudah menghubungkan Uno ke sebuah komputer dengan sebuah kabel USB atau memberi Uno tenaga dengan sebuah adapter *AC-to-DC* atau baterai untuk memulai. Uno berarti “satu” dalam bahasa Itali dan dinamakan untuk menandakan

peluncuran mendatang dari Arduino 1.0. Uno dan Versi 1.0, menjadi referensi versi-versi dari Arduino ke depannya. Uno merupakan seri terakhir dari *board* USB Arduino dan referensi model untuk platform Arduino.[4]

Arduino Uno berbeda dari semua board Arduino sebelumnya, Arduino UNO tidak menggunakan *chip driver* FTDI USB-to-serial. Sebaliknya, fitur-fitur Atmega16U2 (Atmega8U2 sampai ke versi R2) diprogram sebagai sebuah pengubah USB ke serial. Revisi 2 dari board Arduino Uno mempunyai sebuah resistor yang menarik garis 8U2 HWB ke ground, yang membuatnya lebih mudah untuk diletakkan ke dalam DFU mode.

Tabel 2.2. Spesifikasi Arduino UNO[4]

No.	Bagian Penyusun Arduino	Keterangan Spesifikasi
1.	Mikrokontroler	ATmega328
2.	Tegangan pengoperasian	5V
3.	Tegangan input yang disarankan	7-12V
4.	Batas tegangan input	6-20V
5.	Jumlah pin I/O digital	14 (6 di antaranya menyediakan keluaran PWM)
6.	Jumlah pin input analog	6
7.	Arus DC tiap pin I/O	40 mA
9.	Arus DC untuk pin 3.3V	50 mA
10.	Memori Flash	32 KB (ATmega328), sekitar 0.5 KB digunakan oleh <i>bootloader</i>
11.	SRAM	2 KB (ATmega328)
12.	EEPROM	1 KB (ATmega328)
13.	<i>Clock Speed</i>	16 MHz

Revisi 3 dari board Arduino UNO memiliki fitur-fitur baru sebagai berikut:

1. Pinout 1.0: ditambah pin SDA dan SCL yang dekat dengan pin AREF dan dua pin baru lainnya yang diletakkan dekat dengan pin RESET, IOREF yang memungkinkan *shield-shield* untuk menyesuaikan tegangan yang disediakan dari board. Untuk ke

depannya, *shield* akan dijadikan kompatibel/cocok dengan board yang menggunakan AVR yang beroperasi dengan tegangan 5V dan dengan Arduino Due yang beroperasi dengan tegangan 3.3V. Yang ke-dua ini merupakan sebuah pin yang tak terhubung, yang disediakan untuk tujuan kedepannya.

2. Sirkuit *RESET* yang lebih kuat.
3. Atmega 16U2 menggantikan 8U2.

Arduino UNO dapat disuplai melalui koneksi USB atau dengan sebuah power suplai eksternal. Sumber daya dipilih secara otomatis. Suplai eksternal (non-USB) dapat diperoleh dari sebuah adaptor AC ke DC atau battery. Adaptor dapat dihubungkan dengan mencolokkan sebuah center-positive plug yang panjangnya 2,1 mm ke power jack dari board. Kabel lead dari sebuah *battery* dapat dimasukkan dalam *header/* kepala pin Ground (Gnd) dan pin Vin dari konektor *POWER*.

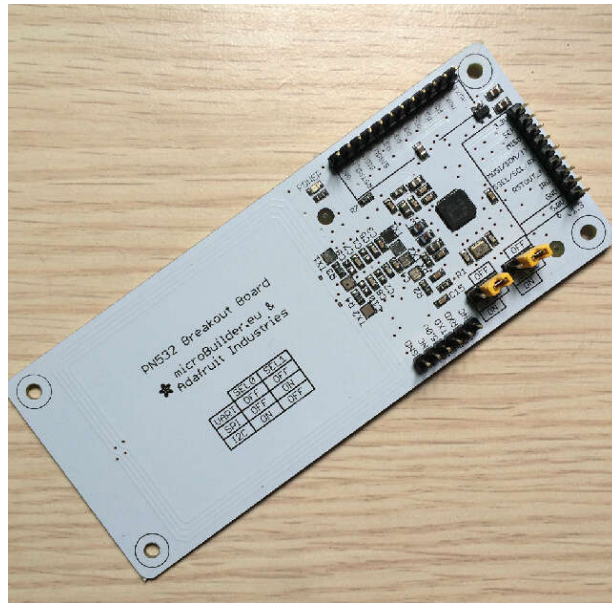
Board Arduino UNO dapat beroperasi pada sebuah suplai eksternal 6 sampai 20 Volt. Jika disuplai dengan yang lebih kecil dari 7 V, kiranya pin 5 Volt mungkin mensuplai kecil dari 5 Volt dan board Arduino UNO bisa menjadi tidak stabil. Jika menggunakan suplai yang lebih dari besar 12 Volt, voltage regulator bisa kelebihan panas dan membahayakan board Arduino UNO. Rentang yang direkomendasikan adalah 7 sampai 12 Volt.[4]

2.4.2 NFC *Shield*

NFC *Shield* merupakan komponen yang berbasis teknologi RFID pada 13.56 MHz, dengan jarak operasi biasanya hingga 10 cm serta memiliki kecepatan pertukaran data hingga 42 kilobits/s dalam waktu yang sangat singkat, berkisar antara 100-150 milisekon. NFC *Shield* menggunakan *chipset* PN532 (chip NFC yang paling populer di pasaran) dan juga tertanam cukup banyak di dalam setiap telepon atau perangkat yang melakukan NFC. *Chipset* ini sangat kuat dan cukup banyak dapat melakukan semuanya, seperti membaca dan menulis ke

tag dan kartu, berkomunikasi dengan telepon (untuk proses pembayaran), dan bertindak seperti sebuah tag NFC.[5]

NFC yang beroperasi pada frekuensi 13.56MHz, didasarkan di sekitar "inisiator" dan "target" model di mana inisiator menghasilkan medan magnet kecil dengan kekuatan target, yang berarti bahwa target tidak memerlukan sumber daya.



Gambar 2.4 Board NFC Shield [6]

Ini sarana komunikasi disebut sebagai Pasif Komunikasi, dan digunakan untuk membaca dan menulis dalam frekuensi kecil, seperti tag NFC/RFID 13.56MHz berdasarkan standar ISO14443A. Pada NFC *Shield* komunikasi aktif (peer-to-peer) juga dimungkinkan ketika kedua perangkat bertenaga, di mana masing-masing perangkat bergantian menciptakan lapangan magnetik sendiri, dengan perangkat sekunder sebagai target dan sebaliknya di rotasi terus menerus.[6]

2.4.3 NFC Tag

Ada lima jenis *tag* yang didefinisikan oleh NFC forum. Untuk Jenis 1, 2, dan 4 berdasarkan pada ISO-14443A, dan tipe 3 berdasarkan pada ISO-18092. Rincian tipe NFC *tag* adalah sebagai berikut :

1. Tipe 1
 - a) Berdasarkan spesifikasi ISO-14443A.

- b) Dapat berupa *read-only*, atau mampu sebagai *read/write*.
- c) Kapasitas memori 96 bytes hingga 2 kilobytes.
- d) Kecepatan komunikasi 106Kb.
- e) Tidak ada keamanan *collision data*.
- f) Contoh : Innovision Topaz, Broadcom BCM20203.

2. Tipe 2

- a) Mirip dengan jenis *tag* tipe 1, *tag* tipe 2 didasarkan pada NXP Philips Mifare Ultralight dengan spesifikasi *tag* (ISO-14443A).
- b) Dapat berupa *read-only*, atau mampu sebagai *read/write*.
- c) Kapasitas memori 96 byte untuk 2 kilobyte.
- d) Kecepatan komunikasi 106Kb.
- e) Mendukung *anti-collision*.
- f) Contoh: NXP Mifare Ultralight.

3. Tipe 3

- a) *Tag* ini didasarkan pada *tag* Sony FeliCa (ISO-18092 dan JIS-X 6319-4), tanpa enkripsi dan dukungan otentikasi yang mampu diberikan oleh FeliCa.
- b) Dikonfigurasi oleh pabrik sebagai *read-only*, atau mampu sebagai *read/write*.
- c) Kapasitas memori bervariasi, hingga 1 MB per pertukaran data.
- d) Memiliki dua kecepatan komunikasi, 212 atau 424Kbps.
- e) Mendukung *anti-collision*.
- f) Contoh: Sony FeliCa.

4. Tipe 4

- a) Mirip dengan jenis *tag* tipe 1, *tag* tipe 4 didasarkan pada spesifikasi *tag* (ISO-14443A) NXP DESFire.
- b) Dikonfigurasi oleh pabrik sebagai *read-only*, atau mampu sebagai *read/write*.
- c) Kapasitas memori 2, 4 atau 8KB.
- d) Memori bervariasi, sampai dengan 32KB per pertukaran data.
- e) Memiliki tiga komunikasi kecepatan: 106, 212 atau 424Kbps.
- f) Mendukung *anti-collision*.

g) Contoh: NXP DESFire, Smart MX-JCOP.

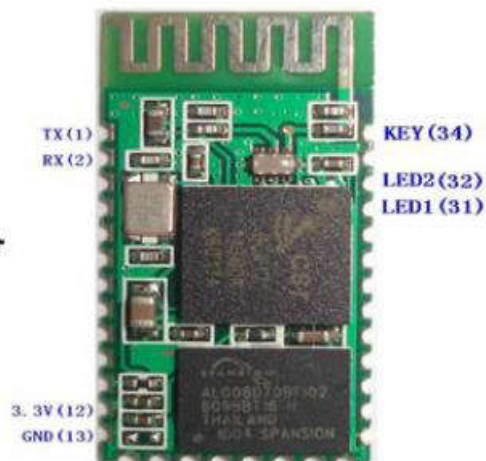
5. Tipe 5

Khususnya untuk NXP Semiconductor, merupakan NFC yang paling umum digunakan saat ini:

- a) MIFARE Classic *tag* (ISO-14443A).
- b) Memori pilihan: 192, 768 atau 3,584 byte.
- c) Kecepatan komunikasi 106Kbps.
- d) Mendukung *anti-collision*.
- e) Contoh: Classic NXP Mifare 1K, Mifare klasik 4K, Mifare Classic Mini.[7]

2.4.4 Modul *Bluetooth* HC-05

Bluetooth adalah protokol komunikasi *wireless* yang bekerja pada frekuensi radio 2,4 GHz. Protokol komunikasi ini mampu melakukan pertukaran data pada perangkat bergerak seperti PDA, laptop, HP, dan lain-lain. *Bluetooth* merupakan teknologi berbasis RF (*Radio Frequency*) pendek dengan jarak maksimal 20 meter. Oleh karena itu *Bluetooth* masuk dalam kategori jaringan komputer PAN (*Personal Area Network*) karena jaringan yang digunakan personal dan memiliki jarak pendek.



Gambar 2.5 Modul *Bluetooth* HC-05[8]

Modul *bluetooth* HC-05 dengan Serial HC 05 adalah Modul *Bluetooth* yang dapat di set sebagai *Master* ataupun sebagai *Slave*. Penamaan *master* dan *slave* pada module *bluetooth* seperti konsep *client* dan *server* pada komunikasi serial biasanya. *Bluetooth* Serial HC 05 adalah versi pengembangan dari Modul *Bluetooth* to Serial HC06. Modul *Bluetooth* to Serial HC05 ini dapat di *setting* sebagai *Master* ataupun di *setting* sebagai *Slave*, berbeda dengan Modul HC06 yang hanya dapat di gunakan sebagai *Slave*. [8]

2.4.5 Motor Servo

Servo umumnya bekerja dengan memindah posisi semula ke posisi lainnya yang secara akurat mengontrol gerakan fisik bukannya terus berputar. Servo ideal untuk membuat sesuatu yang berputar di atas kisaran 0 sampai 180 derajat. servos mudah untuk menghubungkan dan melakukan kontrol.

Servo berisi motor kecil yang terhubung melalui roda gigi ke poros. output poros drive lengan servo, dan juga terhubung ke potensiometer untuk memberikan umpan balik posisi ke rangkaian kontrol internal. Servo mudah digunakan karena tidak perlu motor driver karena motor driver sudah berada di dalam servo. [9]



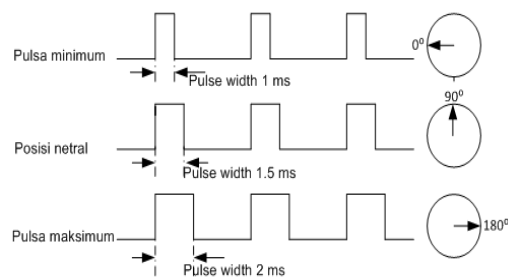
Gambar 2.6 Motor Servo Standar 180° [10]

Motor servo adalah motor yang mampu bekerja dua arah (*Clock Wise* dan *Counter Clock Wise*) dimana arah dan sudut pergerakan rotornya dapat dikendalikan hanya dengan memberikan pengaturan

duty cycle sinyal PWM pada bagian pin kontrolnya. *Duty cycle* adalah sebuah siklus tugas yang diberikan pada komponen, perangkat atau sistem untuk bekerja pada proporsi waktu tertentu. Contoh untuk motor servo memiliki *duty cycle* untuk beroperasi atau bergerak 90° searah jarum jam selama 1 detik, lalu bergerak 90° lagi ke arah sebaliknya selama 1 detik dan begitu seterusnya.

Ada dua jenis motor servo, motor servo standar 180° dan motor servo *continuous*. Untuk motor servo standar 180° seperti pada gambar 2.6 hanya mampu bergerak dua arah dengan defleksi masing-masing sudut mencapai 90° sehingga total defleksi sudut dari kanan-tengah-kiri adalah 180° . Untuk motor servo *continuous* mampu bergerak dua arah tanpa batasan defleksi sudut putar (dapat berputar secara kontinyu).

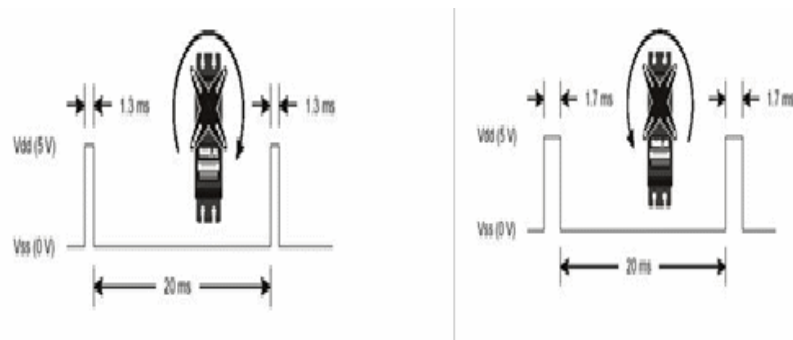
Perbedaan yang bisa dilihat antara motor servo standar dengan motor servo *continuous* selain dari sudut gerak adalah pada salah satu komponen penyusunnya yaitu *gear*. Pada gambar 2.7 adalah bentuk *gear* dari motor servo standar. Perbedaannya adalah adanya semacam pembatas gerak *gear* yang disebut *plastic tab* pada salah satu sisi *gear* di motor servo standar. Selain pada komponen fisik, perbedaan motor servo standar dengan motor *continuous* terdapat juga pada lebar pulsa sinyal PWM (*Pulse Width Modulation*).



Gambar 2.7 Lebar Pulsa Motor Servo Standar 180° [10]

Lebar pulsa untuk motor servo standar dan *continuous* adalah 20ms, seperti pada gambar 2.8 dimana untuk motor servo standar memiliki lebar pulsa antara 0,5 ms dan 2 ms untuk akhir dari rentang sudut maksimum dan sudut 0° pada pulsa ke 1,5ms. Sedangkan untuk

motor servo *continuous* lebar pulsa yang digunakan adalah seperti pada gambar 2.9.[9]

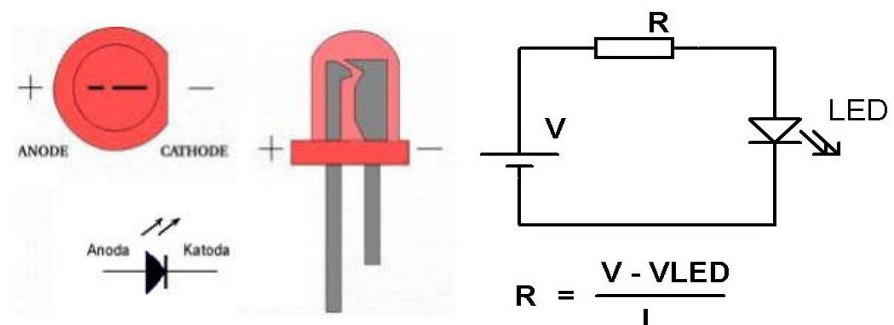


Gambar 2.8 Lebar Pulsa Motor Servo *Continuous*[9]

Motor servo *continuous* memiliki lebar pulsa antara 1,3 ms dan 1,7 ms untuk bergerak ke kanan dan ke kiri dengan sudut 0° atau untuk perintah berhenti pada pulsa ke 1,5 ms.[10]

2.4.6 Light Emitting Diode (LED)

LED atau singkatan dari *Light Emitting Diode* adalah salah satu komponen elektronik yang tidak asing lagi di kehidupan manusia saat ini. LED merupakan dioda yang dapat memancarkan cahaya pada saat mendapatkan arus maju (*forward bias*). Untuk mendapatkna emisi cahaya pada semikonduktor, doping yang pakai adalah *galium*, *arsenic* dan *phosporus*. Jenis doping yang berbeda menghasilkan warna cahaya yang berbeda pula.



Gambar 2.9 *Light Emitting Diode* (LED). [11]

Dari gambar diatas dapat diketahui bahwa LED memiliki 2 buah kaki seperti dioda yaitu kaki anoda (+) dan kaki katoda (-). Pada gambar diatas kaki anoda memiliki ciri fisik lebih panjang dari kaki katoda. Pemasangan LED agar dapat menyala adalah dengan

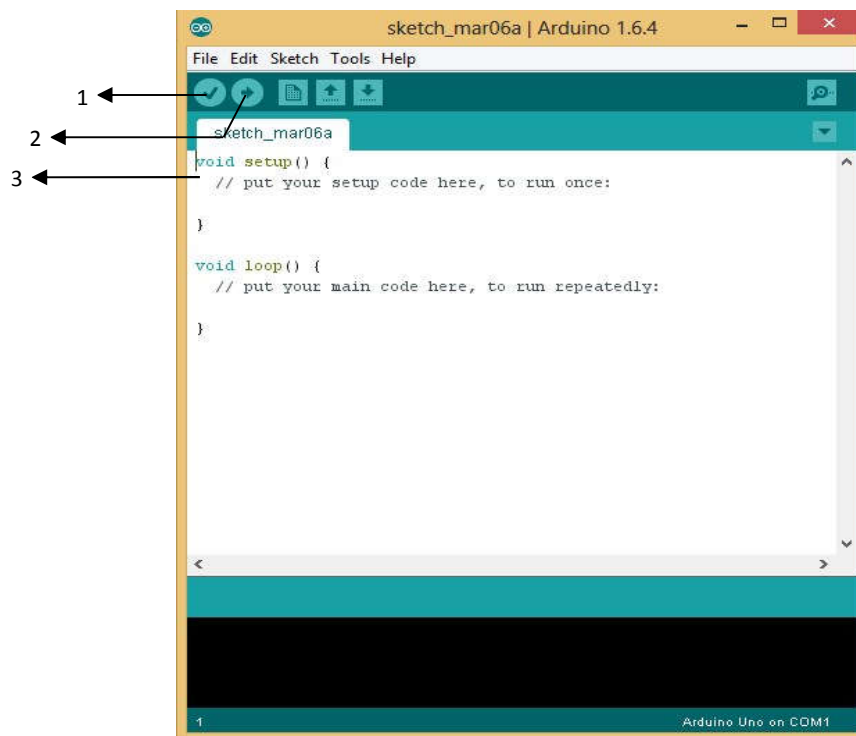
memberikan tegangan bias maju yaitu dengan memberikan tegangan positif ke kaki anoda dan tegangan negatif ke kaki katoda. Konsep pembatas arus pada dioda adalah dengan memasang resistor secara seri pada salah satu kaki LED.

Kemampuan mengalirkan arus pada LED cukup rendah maksimal 20 mA apabila LED dialiri arus lebih besar dari 20 mA maka LED akan rusak, sehingga LED dipasang sebuah resistor sebagai pembatas arus.[11]

2.5. PERANGKAT LUNAK

2.5.1 IDE Arduino

Pada Atmega328 yang terpasang pada *board* Arduino UNO terdapat *bootloader* yang memungkinkan untuk *meng-upload* program ke perangkat Arduino. Arduino UNO dapat diprogram dengan *software* Arduino bernama IDE Arduino.



Gambar 2.10 Tampilan awal IDE Arduino.[4]

IDE adalah bagian dari *software* yang bersifat *open source* yang memungkinkan untuk membuat program sehingga dapat dimengerti oleh Arduino.

Bahasa yang digunakan adalah bahasa pemrograman Arduino, IDE dapat menulis program di komputer secara *step-by-step* kemudian setelah selesai instruksi dari program tersebut akan di *upload*.

Program IDE Arduino terdiri dari :

1. *Compaile*, sebuah modul yang mengubah kode program menjadi kode biner.
2. *Uploader*, sebuah modul yang memuat kode biner dari komputer ke dalam *momory* di dalam papan Arduino.
3. *Editor program*, sebuah windows yang memungkinkan pengguna menulis dan mengedit program.

Tampilan pada gambar 2.6 adalah tampilan awal program IDE Arduino.

Sebuah kode program Arduino umumnya disebut dengan istilah *skect*. Kata “*sketch*” digunakan dengan “kode kode program” dimana keduanya memiliki arti yang sama. Di dalam menu pada IDE tersebut terdapat menu pendukung antara lain pada *tools*, didalam *tools* terdiri dari beberapa fungsi antara lain adalah *verivy* yang berfungsi untuk meng-*compile* program, *stop* berfungsi untuk menghentikan program yang sedang berjalan, *new* untuk membuat *sketch* baru, *open* untuk membuka *sketch* yang ada, dan *save* untuk menyimpan *sketch*. [4]

2.5.2 Strukur Bahasa Pemrograman Arduino

Bahasa pemrograman Arduino memiliki struktur dasar yang terdiri dari dua bagian utama atau fungsi, blok tertutup (dengan pasangan kurung kurawal), pernyataan dan komentar.

```
1 // B// Baris komentar
2
3 /*
4 Blok komentar
5 */
6
7 void setup()
8 {
9 pernyataan;
10 }
11
12 void loop()
13 {
14 pernyataan;
15 }aris komentar
```

Gambar 2.11 Instruksi Pemrograman Dasar [12]

Pada Gambar 2.10, Penjelsan untuk masing-masing bagian dari struktur program Arduino seperti diatas adalah:

a) Setup()

Fungsi setup() dipanggil sekali pada waktu program pertama kali berjalan. Fungsi ini biasanya digunakan sebagai inisialisasi atau tahap persiapan berupa pengaturan cara kerja atau nilai-nilai awal. Contoh:

```
1 void setup()
2 {
3 // Mengatur pin sebagai keluaran
4 pinMode(pin, OUTPUT);
5 }
```

Gambar 2.12 Fungsi Setup()[12]

b) Loop

Fungsi loop() akan menjalankan pernyataan di dalamnya berulang-ulang kali tanpa henti. Fungsi ini berisi program utama yang akan menyebabkan perubahan keadaan pada program, menerima respon dari papan Arduino dan mengendalikannya. Contoh:

```
1 void loop()
2 {
3 digitalWrite(pin, HIGH); /*Mengaktifkan port */
4 delay(1000); // Jeda satu detik
5 digitalWrite(pin, LOW); /*Mematikan
6 port*/
7 delay(1000); // Jeda satu detik
8 }
```

Gambar 2.13 Fungsi Loop()[12]

c) Komentar

Komentar adalah *text* yang akan diabaikan oleh compiler karena tidak akan dikompilasi. Komentar tidak wajib disertakan didalam sebuah program, namun berguna untuk memberikan penjelasan mengenai suatu fungsi atau cara kerja dari program tersebut. Seperti C/C++, bahasa pemograman Arduino mengenali dua jenis komentar yaitu dalam bentuk baris komentar dan blok komentar.

d) Blok Kurung Kurawal

Kurung kurawal menandai bagian awal dan akhir dari sebuah blok fungsi atau blok pernyataan. Kurung kurawal pembuka ({}) harus

diikuti dengan kurung kurawal penutup(`)`) dan IDE Arduino telah menyediakan fasilitas untuk memeriksa penggunaan kurung kurawal tersebut. Pilih kurung kurawal pembuka atau spasi kosong sesudahnya lalu Arduino akan memandu untuk memasang kurung kurawal penutup. Contoh:

```
1 // Kurung kurawal sebagai penanda fungsi
2 void pesan()
3 {
4 /* Kurung kurawal sebagai penanda blok
5 pernyataan */
6 x = 5;
7 while(x > 0)
8 {
9 Serial.print('a');
10 X++;
11 }
12 }
```

Gambar 2.14 Fungsi Kurung Kurawal Awal dan Akhir Sebuah Blok[12]

e) Pernyataan

Pernyataan adalah instruksi pemrograman untuk melakukan sesuatu. Setiap pernyataan dalam Bahasa Arduino harus di akhiri dengan tanda titik koma(`:`). Tanda tersebut juga berfungsi untuk memisahkan satu elemen dengan elemen lainnya dalam suatu pernyataan.

f) Fungsi

Fungsi adalah suatu blok kode program yang berisi satu atau sekumpulan pernyataan untuk melakukan sesuatu hal tertentu. Fungsi dapat memiliki argumen masukan dan menghasilkan suatu nilai keluaran dengan tipe data-data tertentu. Dua jenis fungsi yang wajib adalah `setup()` dan `loop()` dengan penjelasan seperti di atas. Struktur dari sebuah fungsi adalah :

```
1 jenis_data nama_fungsi(argumen)
2 {
3 pernyataan;
4 }
```

Gambar 2.15 Instruksi Fungsi-fungsi[12]

Jika suatu fungsi tidak membutuhkan masukan, maka argumen dapat dikosongkan atau diisi dengan kata kunci `void`. Tipe data

keluaran dari suatu fungsi didefinisikan dalam jenis_data atau bila fungsi tidak memberikan nilai keluaran maka diisi dengan void.[12]

2.5.3 Sistem Operasi Android

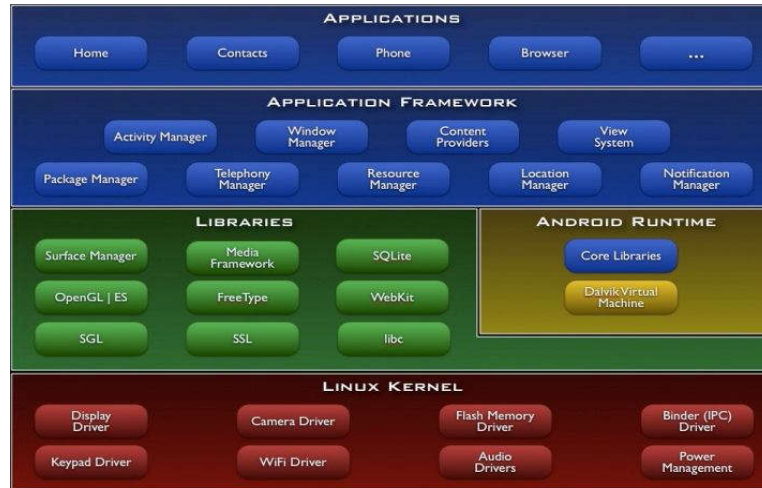
Dalam menjalankan sistem kendali pintu pada perangkat MSAN menggunakan sistem operasi Android, yang akan menjalankan aplikasi dari sistem pengendali pintu. Android merupakan sebuah *operating system* bergerak yang mengadopsi sistem operasi LINUX, namun sudah termodifikasi. Android menjadi kepemilikan *Google* pada tahun 2005. Android bersifat *open-source* sehingga setiap kode program Android bersifat terbuka dan gratis.

Setiap tahun Android mengalami perkembangan sejak diluncurkan pertama kali. Yang membedakan perkembangan di setiap tahun nya yaitu dari segi penamaannya yang berasal dari makanan penutup dari abjad A hingga Z.

Android tersedia secara *Open Source* bagi manufaktur perangkat keras untuk memodifikasi kebutuhan. Meskipun konfigurasi perangkat Android tidak sama antara satu perangkat dengan perangkat lainnya, namun Android sendiri mendukung *fitur-fitur* berikut :

- Penyimpanan (*Storage*) – menggunakan SQLite yang merupakan *database relational* yang ringan untuk menyimpan data.
- Koneksi (*Concetivity*) – mendukung GSM/EDGE, CDMA, EV-DO, UMTS, *Bluetooth*, *WiFi*, LTE, dan *WiMAX*.
- Pesan (*Messaging*) – mendukung SMS dan MMS.
- Web Browser
- Media – media yang didukung antara lain (MPEG-4, MP3, MIDI, dan media lainnya.
- *Hardware* – terdapat Accelerometer, Sensor, Camera, Digital Compass, Proximity Sensor dan GPS.
- *Multi-touch* – mendukung layar *multi-touch*.
- *Multi-tasking* – mendukung aplikasi multi-tasking
- Dukung Flash – Android 2.4 mendukung Flash 10.1.[13]

Sistem operasi Android mempunyai lapisan - lapisan yang berisi dari beberapa program yang mendukung jalannya fungsi dari sistem operasi. Lapisan-lapisan tersebut dapat dilihat pada gambar 2.8.[14]



Gambar 2.16 Arsitektur Android[14]

1. *Applications dan Widgets*

Applications dan *Widgets* adalah *layer* di mana berhubungan dengan aplikasi saja yang mana sering diakses oleh pengguna, pada *layer* ini pengguna biasanya mendownload aplikasi kemudian melakukan instalasi dan jalankan aplikasi tersebut. Di *layer* ini pengguna menemukan aplikasi inti termasuk klien email, program SMS, kalender, peta, *browser*, kontak.

2. *Applications Frameworks*

Android adalah "*Open Development Platform*" yaitu Android memberikan kesempatan kepada pengembang untuk membangun aplikasi yang bagus, inovatif serta lebih kompleks. *Application framework* merupakan sekumpulan peralatan dasar seperti alokasi *resource smartphone*, aplikasi telepon, pergantian antar-proses atau program, serta pelacakan lokasi fisik telepon. Pengembang memiliki akses penuh menuju *Android Protocol Interface (API) framework* seperti yang dilakukan oleh aplikasi yang kategori inti. Arsitektur aplikasi dirancang agar pengguna dengan mudah dapat menggunakan

kembali komponen yang sudah digunakan (*reuse*). Dengan demikian, aplikasi *framework* merupakan *layer* di mana bagi para pembuat aplikasi dapat mengembangkan/membuat aplikasi yang akan dijalankan di sistem operasi Android. Hal ini dikarenakan pada *layer* ini aplikasi dapat dirancang dan dibuat.

3. *Libraries*

Libraries adalah *layer* di mana fitur - fitur Android berada, biasanya para pembuat aplikasi mengakses *libraries* untuk menjalankan aplikasinya. Pada *layer* ini berisi *library - library* dalam bahasa /C++ yang digunakan oleh komponen - komponen yang berada pada Android. Beberapa *core library* seperti *libraries media* untuk pemutaran media audio dan video, *libraries surface manager* untuk mengatur tampilan, *libraries Graphics* mencakup SGL untuk grafis 2D dan OpenGL untuk grafis 3D, *libraries SQLite* untuk dukungan *database*, *libraries SSL* dan *Webkit* terintegrasi dengan *web browser* dan *security*, *libraries LiveWebcore* mendukung modern *web browser* meupun *engine embeded web view*.

4. *Android Run Time*

Pada *layer* ini yang menjadikan aplikasi Android dapat berjalan, dimana dalam prosesnya menggunakan implementasi Linux. Di dalam *Android Run Time* dibagi menjadi dua bagian yaitu:

Aplikasi Android yang dibangun menggunakan bahasa java, sementara Dalvik sebagai virtual mesinnya bukan *Virtual Machine Java*. Dengan demikian, *core libraries* merupakan *library* yang berfungsi menterjemahkan bahasa java.

- a) *Core Libraries* merupakan virtual mesin berbasis register yang dioptimalkan untuk menjalankan fungsi - fungsi secara efisien.
- b) *Dalvik Virtual Machine* adalah mesin yang membuat kerangka dasar pada aplikasi Android.

5. *Linux Kernel*

Linux kernel merupakan *layer* yang berisi inti dari sistem operasi Android tersebut berada. Pada *layer* ini terdapat *file -file system* yang mengatur sistem *processing*, *memory*, *resources*, *drivers*, dan sistem-sistem operasi Android lainnya. Google menggunakan *linux kernel* versi 2.6 untuk membangun sistem Android.[14]

Android menggunakan bahasa pemrograman java. Kode java dan *file resource* yang dibutuhkan oleh suatu aplikasi dikompilasi secara bersama yang kemudian dijadikan dalam satu paket oleh *tool* ke dalam paket Android yang kemudian menghasilkan *file* berformat .apk. Berikut enam jenis komponen pada aplikasi Android yaitu:

1. *Activities*

Suatu *activity* menampilkan *user interface* (UI) kepada pengguna. Satu *activity* bertindak sebagai *user interface* dan menunjukkan aplikasi satu layar tersebut kepada pengguna. Apabila pindah dari satu *activity* ke *activity* yang lain dapat dilakukan dengan satu langkah, misalnya dengan mengklik tombol, memilih opsi atau menggunakan *triggers* tertentu.

2. *Service*

Service tidak memiliki *Graphic User Interface* (GUI), melainkan berjalan secara *background*. *Service* berjalan pada *thread* utama dari proses aplikasi. Sehingga, pengguna dapat melakukan aktifitas ganda contohnya pada saat pengguna sedang memutar musik pengguna juga dapat mengetik pesan atau menggunakan aplikasi lain.

3. *Broadcast Receiver*

Broadcast receiver digunakan untuk menerima dan menyampaikan pemberitahuan atau informasi. Contoh *broadcast* seperti notifikasi zona waktu berubah, ketika baterai sudah dalam keadaan akan habis, ketika gambar sudah selesai diambil oleh *camera*, atau yang lainnya. *Broadcast receiver* tidak mempunyai

user interface (UI), namun memiliki *activity* yang dapat memberikan respon informasi yang diterima, atau dapat menggunakan *Notification Manager* untuk memberitahu kepada pengguna, seperti lampu latar menyala atau terdapat getaran.

4. *Content Provider*

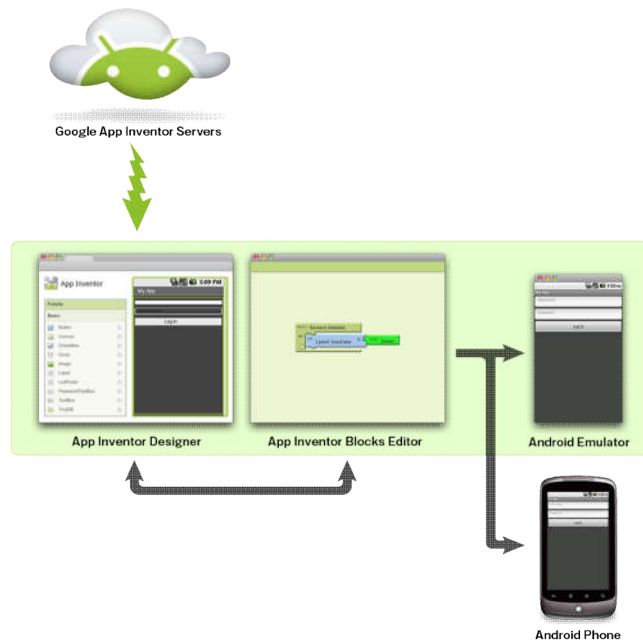
Content provider membuat kumpulan aplikasi data secara spesifik sehingga bisa digunakan oleh aplikasi lain. Data disimpan dalam *file* sistem seperti *database SQLite*. *Content provider* menyediakan cara untuk mengakses data yang dibutuhkan oleh suatu *activity*, misalnya ketika kita menggunakan aplikasi yang membutuhkan peta atau aplikasi yang membutuhkan untuk mengakses data kontak dan navigasi, maka disinilah fungsi *content provider*.^[14]

2.5.4 *App Inventor*

App Inventor adalah program yang dibuat oleh Google dan sekarang dikembangkan oleh MIT. Program ini dapat digunakan untuk membuat dan mendesain aplikasi Android yang berbasis *Web page* dan *Java interface*.^[15]

App inventor merupakan sistem perangkat lunak untuk membuat aplikasi pada Android. App Inventor memiliki perbedaan dengan sistem pengembangan aplikasi pada biasanya melainkan dengan interaksi visual berbasis grafis. App Inventor disebut sebagai sistem terpadu untuk pengembangan aplikasi berbasis blok-blok grafis (*block language*). Pembuatan aplikasi di App Inventor dapat dijalankan pada Android minimal versi 2.3 Gingerbread keatas.^[12]

Interaksi programmer dengan App Inventor menggunakan antar muka visual dengan operasi *drag and drop*. Secara sistem, App Inventor terdiri dari dua komponen, yaitu *server* dan *client*. Komponen App Inventor dapat dilihat pada Gambar 2.10. Server App inventor berfungsi menyimpan semua aset program dan memberikan layanan lainnya yang terkait dengan manajemen berkas aplikasi (*project*).



Gambar 2.17 Komponen App Inventor 2[16]

Sedangkan sisi *client* adalah aplikasi yang berhubungan langsung dengan *programmer* (permbuat aplikasi). Sisi *client* terdiri dari tiga bagian berikut :

- App Inventor ***Designer***, merupakan aplikasi web untuk mendesain tampilan *visual GUI (Graphical User Interface)* dan memilih komponen yang diperlukan;
- App Inventor ***Block Editor***, merupakan aplikasi Java untuk mendefinisikan *behaviour* (perilaku) dari tiap komponen dengan menyusun blok-blok program secara *visual* (tampilan berbentuk potongan *puzzle*).
- ***Emulator*** merupakan perangkat lunak yang berjalan diatas komputer dan dapat mensimulasikan seperti perangkat sesungguhnya. Emulator ini dapat digunakan untuk beberapa layanan seperti telepon, sms, atau layanan lokasi.[16]