

BAB II

TINJAUAN PUSTAKA

II.1. Pengertian Perancangan

Desain atau perancangan dalam pengembangan perangkat lunak merupakan upaya untuk mengkonstruksi sebuah sistem yang memberikan kepuasan (mungkin informal) akan spesifikasi kebutuhan fungsional memenuhi target, memenuhi kebutuhan secara implisit dan eksplisit dari segi performansi maupun penggunaan sumber daya, kepuasan batasan pada proses desain dari segi biaya, waktu, dan perangkat. (Rosa A.S, M. Shalahuddin ; 2011: 21).

Desain atau perancangan adalah mentransformasikan kebutuhan detail menjadi kebutuhan yang sudah lengkap, dokumen desain sistem *fokus* pada bagaimana dapat memenuhi fungsi-fungsi yang dibutuhkan. (Rosa A.S, M. Shalahuddin ; 2011: 25).

Desain atau perancangan perangkat lunak adalah proses langkah yang *fokus* pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean. tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan perangkat lunak ke representasi desain agar dapat di implementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu di dokumentasikan. (Rosa A.S, M. Shalahuddin ; 2011: 27).

Perancangan adalah proses menuangkan ide dan gagasan berdasarkan teori-teori dasar yang mendukung. Proses perancangan dapat dilakukan dengan cara pemilihan komponen yang akan digunakan, mempelajari karakteristik dan data fisiknya, membuat rangkaian skematik dengan melihat fungsi-fungsi komponen yang dipelajari, sehingga dapat dibuat alat yang sesuai dengan spesifikasi yang diharapkan. (<http://elib.unikom.ac.id/download.php?id=143047>: dikunjungi pada tanggal 28 Juli 2016).

II.2. Pengertian Dasar Sistem Kendali

Sistem kendali merupakan suatu sistem yang keluarannya dikendalikan pada suatu nilai tertentu atau untuk mengubah beberapa ketentuan yang telah ditetapkan dari masukan ke sistem. Untuk merancang suatu sistem yang dapat merespon suatu perubahan tegangan dan mengeksekusi perintah berdasarkan situasi yang terjadi, maka diperlukan pemahaman tentang sistem kendali (*control system*). Sistem kendali merupakan suatu kondisi dimana sebuah perangkat (*device*) dapat dikendalikan sesuai dengan perubahan situasi.

Kendali (pengendalian) adalah sebuah proses atau upaya untuk mencapai tujuan. Dengan demikian, sistem kendali adalah kombinasi dari beberapa komponen (*subsistem*) yang bekerja secara sinergi dan terpadu untuk memperoleh hasil yang diinginkan (tujuan). Sebagai contoh sederhana dan akrab dengan aktivitas sehari-hari dari konsep kendali adalah saat mengendarai kendaraan. Tujuan yang diinginkan dari proses tersebut adalah berjalannya kendaraan pada lintasan (*track*) yang diinginkan. Ada beberapa komponen yang terlibat di dalamnya, misalnya pedal gas, *speedometer*, mesin (penggerak), rem, dan pengendara. Sistem kendali berkendara berarti kombinasi dari komponen-komponen tersebut yang menghasilkan berjalannya kendaraan pada lintasan yang diinginkan. Ketika jalan lengang dan aturan memperbolehkan, pengendara mempercepat laju kendaraan dengan membuka pedal gas. Demikian pula, jika ada kendaraan lain di depan atau lampu penyeberangan berwarna merah maka pengendara menginjak rem dan menurunkan kecepatannya. Semua upaya itu dilakukan untuk mempertahankan kendaraan pada lintasan yang diinginkan.

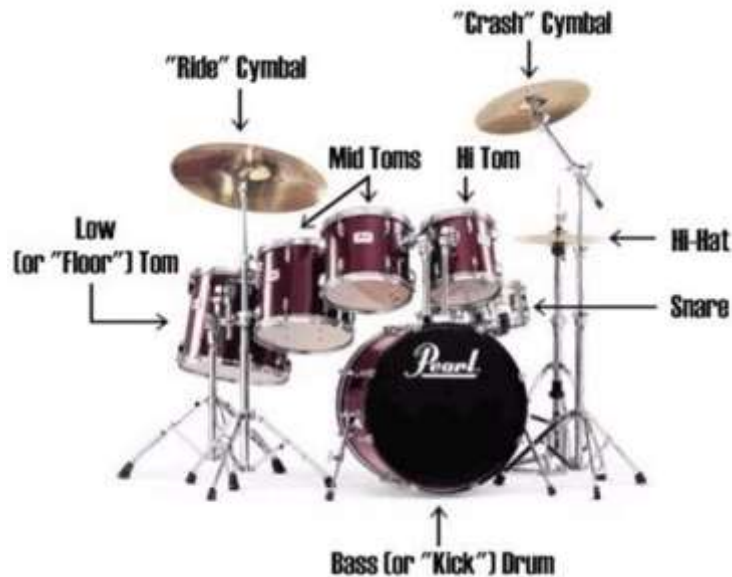
Contoh lain dapat disebutkan berupa proses memindahkan barang oleh tangan kita. Pada proses tersebut, tujuannya adalah posisi atau letak barang yang diinginkan. Komponennya

berupa tangan (dalam hal ini tentunya dengan otot tangan), mata, dan otak sebagai pengendali. Pada saat tangan bergerak untuk memindahkan barang, mata akan menangkap informasi tentang posisi pada saat itu. Informasi tersebut diproses oleh otak untuk disimpulkan apakah posisinya sudah benar atau tidak. Selanjutnya, apabila posisinya masih belum tercapai maka otak akan memerintahkan otot tangan untuk bergerak memindahkan barang ke posisi yang diinginkan. Secara umum dapat dikatakan semua proses yang terjadi di alam pada hakikatnya adalah sebuah sistem kendali. (Asep Najmurrokhman, 2010 : 1)

II.3. Alat Musik Drum

Alat musik drum termasuk kedalam golongan alat musik perkusi yaitu alat musik yang dimainkan dengan cara dipukul dengan tangan atau dengan bantuan alat yang dinamakan dengan stik. Drum yang banyak digunakan sekarang ini adalah berupa drum set yang terdiri dari berbagai macam komponen seperti snare, tom 1, tom 2, bass drum, cymbal dan hi-hat. (jurnal : “Rancang Bangun Alat Musik Drum Air Berbasis Mikrokontroler”, Sistem Komputer Fakultas Teknologi Informasi Universitas Andalas, diakses tanggal 30 juli 2016)

Bentuk dari alat musik drum ditunjukkan pada gambar II.1 berikut.



Gambar II.1. Alat Musik Drum

(Sumber : <http://www.clasemusica.com/>, diakses tanggal 15 September 2016)

II.4. Pengertian Sensor

Dalam rangkaian elektronika untuk keperluan pengukuran atau deteksi, diperlukan suatu bagian yang disebut sensor. Sensor berfungsi untuk menubah besaran yang bersifat fisis atau suhu, tekanan, berat, atau intensitas cahaya menjadi besaran listrik (tegangan atau arus listrik).

Sensor adalah alat untuk mendeteksi atau mengukur sesuatu yang digunakan untuk mengubah variasi mekanis, *magnetic*, panas, dan kimia menjadi tegangan dan arus listrik. Dalam lingkungan sistem pengendali dan robotika, sensor memberikan kesamaan yang menyerupai mata, pendengaran, hidung, lidah yang kemudian akan diolah oleh kontroler sebagai otaknya. (Petruzella : 2010)

Sensor memiliki suatu ukuran yang disebut sensitivitas. Sensitivitas menunjukkan seberapa besar pengaruh perubahan nilai besaran fisis yang diukur oleh sensor terhadap keluaran dari sensor tersebut. Misalnya, sebuah sensor suhu yang tegangan keluarannya berubah 0,1 V

jika terjadi perubahan suhu sebesar 1°C . Maka sensor suhu tersebut dapat dikatakan memiliki sensitivitas sebesar $0,1\text{V}/1^{\circ}\text{C}$. (Endra Pitowarno ; 2006 : 56)

Sensor yang baik memiliki ciri-ciri sebagai berikut:

1. Peka terhadap besaran yang akan diukur.
2. Tidak peka terhadap besaran lain yang tidak akan diukur.
3. Keberadaan sensor tidak mempengaruhi besaran yang akan diukur.

II.5. Mikrokontroler Arduino

Arduino dikatakan sebagai sebuah platform dari physical computing yang bersifat *open source*. Pertama-tama perlu dipahami bahwa kata “*platform*” disini adalah sebuah pilihan kata yang tepat. Arduino tidak hanya sekedar sebuah alat pengembangan, tetapi ia adalah kombinasi dari hardware, bahasa pemrograman dan *Integrated Development Environment* (IDE) yang canggih. IDE adalah sebuah *software* yang sangat berperan untuk menulis program, meng-*compile* menjadi kode biner dan meng-*upload* ke dalam *memory microcontroller*. Ada banyak projek dan alat – alat yang dikembangkan oleh akademisi dan profesional dengan menggunakan Arduino, selain itu juga ada banyak modul-modul pendukung (sensor, tampilan, penggerak dan sebagainya) yang dibuat oleh pihak lain untuk bisa disambungkan dengan Arduino. Arduino berevolusi menjadi sebuah platform karena ia menjadi pilihan dan acuan bagi banyak praktisi.

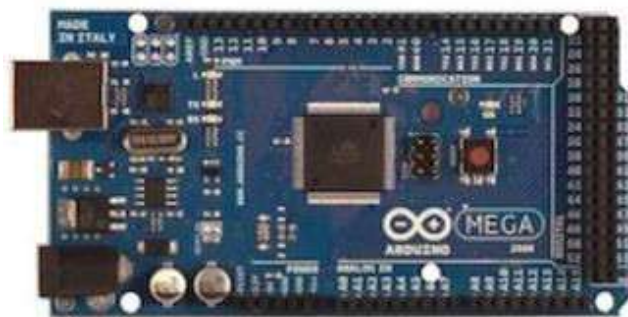
Salah satu yang membuat arduino memikat hati banyak orang adalah karena sifatnya *open source*, baik untuk *hardware* maupun *software*-nya. Komponen utama didalam papan Arduino adalah sebuah microcontroller 8 bit dengan merk Atmega yang dibuat oleh perusahaan Atmel Corporation. Berbagai papan Arduino menggunakan tipe Atmega yang

berbeda-beda tergantung dari spesifikasinya, sebagai contoh Arduino Uno menggunakan Atmega328 sedangkan Arduino Mega 2560 yang lebih canggih menggunakan Atmega2560 (Feri Djuandi,2011: 2).

II.6. Arduino Mega 2560

II.6.1. Pengenalan Arduino Mega 2560

Arduino Mega 2560 adalah sebuah papan mikrokontroler berbasis Atmega 2560 (*datasheet*). Mempunyai 54 pin digital *input/output* (dimana 14 pun dapat digunakan sebagai keluaran PWM), 16 pin input analog, 2 UARTs (*Hardware serial ports*), sebuah *crystal oscillator* 16 MHz, sebuah penghubung USB, sebuah colokan listrik, ICSP *header*, dan tombol kembali. Setiap isi dari Arduino Mega 2560 membutuhkan dukungan mikrokontroler; koneksi mudah antara Arduino mega 2560 ke komputer dengan sebuah kabel USB atau daya dengan AC to DC adaptor atau baterai untuk memulai. Arduino mega cocok sebagai rancangan pelindung untuk Arduino *Deumilanove* atau *Diecimila*.



Gambar II.2. Arduino Mega 2560

(Sumber : arduino.cc)

II.6.2. Arsitektur Arduino Mega 2560

Arduino Mega 2560 terbentuk dari processor yang dikenal dengan Mikrokontroler ATmega 2560. Mikrokontroler ATmega 2560 memiliki beberapa fitur / spesifikasi yang menjadikannya sebagai solusi pengendali yang efektif untuk berbagai keperluan.



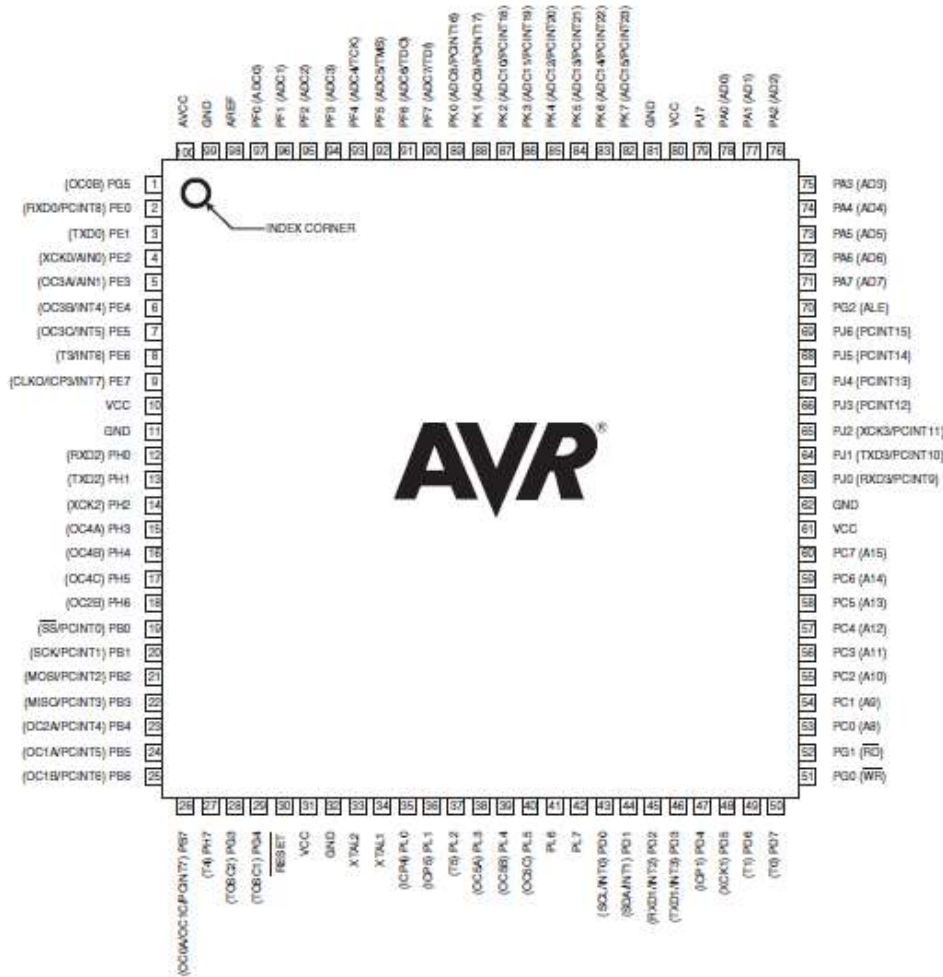
Gambar II.3. ATmega 2560 pada Arduino Mega 2560

(Sumber :Atmel Corporation. 2014)

Fitur-fitur tersebut antara lain :

1. Tegangan Operasi sebesar 5 V
2. Tegangan input sebesar 6 – 20 V tetapi yang direkomendasikan untuk ATmega 2560 sebesar 7 – 12 V.
3. Pin digital I/O sebanyak 54 pin dimana 14 pin merupakan keluaran dari PWM.
4. Pin input analog sebanyak 16 pin
5. Arus DC pin I/O sebesar 40 mA sedangkan Arus DC untuk pin 3.3V sebesar 50 mA
6. *Flash memory* 156 Kb yang mana 8 Kb digunakan oleh bootloader.
7. SRAM 8 Kbyte
8. EEPROM 4 Kbyte
9. Serta mempunyai 2 Port UARTs untuk komunikasi serial.

II.6.4. Konfigurasi Pin Arduino Mega



Gambar II.4. Konfigurasi Pin Atmega 2560

(Sumber :Atmel Corporation. 2014)

1. VCC adalah tegangan catu digital
2. GND adalah *Ground*
3. Port A (PA7..PA0)

Port A adalah sebuah port I/O 8 bit dua arah dengan *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port A memiliki karakter penggerak karakteristik dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port A eksternal *pulled low* sumber arus jika resistor pull-up aktif. Pin port A dinyatakan tri ketika sebuah kondisi reset menjadi aktif, bahkan jika waktu

tidak berjalan. Port A juga menyajikan fungsi dari berbagai fitur spesial dari Atmega640/1280/1281/2560/2561.

4. Port B (PB7..PB0)

Port B adalah sebuah port I/O 8 bit dua arah dengan *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port B memiliki karakter penggerak karakteristik dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port A eksternal *pulled low* sumber arus jika resistor pull-up aktif. Pin port A dinyatakan tri ketika sebuah kondisi reset menjadi aktif, bahkan jika waktu tidak berjalan. Port B mempunyai kemampuan bergerak lebih baik daripada port lainnya.

5. Port C (PC7..PC0)

Port C adalah sebuah port I/O 8 bit dua arah dengan *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port C memiliki karakter penggerak karakteristik dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port C eksternal *pulled low* sumber arus jika resistor pull-up aktif. Pin port C dinyatakan tri ketika sebuah kondisi reset menjadi aktif, bahkan jika waktu tidak berjalan.

6. Port D (PD7..PD0)

Port D adalah sebuah port I/O 8 bit dua arah dengan *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port D memiliki karakter penggerak karakteristik dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port D eksternal *pulled low* sumber arus jika resistor *pull-up* aktif. Pin

port D dinyatakan tri ketika sebuah kondisi reset menjadi aktif, bahkan jika waktu tidak berjalan.

7. Port E (PE7..PE0)

Port E adalah sebuah port I/O 8 bit dua arah dengan *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port E memiliki karakter penggerak karakteristik dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port E eksternal *pulled low* sumber arus jika *resistor pull-up* aktif. Pin port E dinyatakan tri ketika sebuah kondisi reset menjadi aktif, bahkan jika waktu tidak berjalan.

8. Port F (PF7..PF0)

Port F disajikan sebagai masukan analog ke *A/D converter*. Port F juga menyajikan sebuah port I/O 8 bit dua arah, jika *A/D Converter* tidak digunakan. Pin port dapat menyediakan *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port F memiliki karakter penggerak karakteristik dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port F *eksternal pulled low* sumber arus jika resistor pull-up aktif. Pin port F dinyatakan tri ketika sebuah kondisi reset menjadi aktif, bahkan jika waktu tidak berjalan. Jika antarmuka JTAG mengizinkan, *pull-up resistor* pada pin PF7(TDI), PF5(TMS), dan PF4(TCK) akan iaktifkan bahkan jika terjadi reset. Port F juga menyajikan fungsi dari antarmuka JTAG.

9. Port G (PG7..PG0)

Port G adalah sebuah port I/O 6 bit dua arah *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port G memiliki karakter penggerak dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port G

eksternal *pulled low* sumber arus jika resistor *pull-up* aktif. Pin port G dinyatakan tri ketika sebuah kondisi reset menjadi aktif, bahkan jika waktu tidak berjalan.

10. Port H (PH7..PH0)

Port H adalah sebuah port I/O 8 bit dua arah *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port H memiliki karakter penggerak dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port H eksternal *pulled low* sumber arus jika *resistor pull-up* aktif. Pin port H dinyatakan tri ketika sebuah kondisi reset menjadi aktif, bahkan jika waktu tidak berjalan.

11. Port J (PJ7..PJ0)

Port J adalah sebuah port I/O 8 bit dua arah *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port J memiliki karakter penggerak dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port J eksternal *pulled low* sumber arus jika *resistor pull-up* aktif. Pin port J dinyatakan tri ketika sebuah kondisi reset menjadi aktif, bahkan jika waktu tidak berjalan.

12. Port K (PK7..PK0)

Port K disajikan sebagai masukan analog ke *A/D converter*. Port K adalah sebuah port I/O 8 bit dua arah *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port K memiliki karakter penggerak karakteristik dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port K eksternal *pulled low* sumber arus jika *resistor pull-up* aktif. Pin port K dinyatakan tri ketika kondisi reset menjadi aktif, bahkan jika waktu tidak berjalan.

13. Port L (PL7..PL0)

Port L adalah sebuah port I/O 8 bit dua arah dengan *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port L memiliki karakter penggerak karakteristik dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port L eksternal *pulled low* sumber arus jika *resistor pull-up* aktif. Pin port L dinyatakan tri ketika sebuah kondisi reset menjadi aktif, bahkan jika waktu tidak berjalan.

14. Reset

Input reset. Sebuah level rendah pada pin ini untuk lebih panjang dari pada panjang minimum pulsa akan menghasilkan sebuah reset, bahkan jika waktu tidak berjalan. Panjang minimum pulsa dijelaskan pada “Sistem dan karakter reset” pada halaman 360. Pulsa terpendek tidak dijamin menghasilkan sebuah reset .

15. XTAL1

Input ke *inverting amplifier oscillator* dan input ke internal jalur operasi waktu.

16. XTAL2

Keluaran dari *inverting oscillator amplifier*

17. AVCC

AVCC merupakan pin tegangan catu untuk port F dan *A/D Converter*. AVCC dapat terhubung secara eksternal ke VCC, bahkan jika ADC tidak digunakan jika ADC digunakan, ADC akan terhubung ke VCC melalui sebuah *low pass filter*.

18. AREF

AREF adalah pin referensi analog untuk *A/D Converter* (Atmel Corporation.2014).

II.7. Infrared

LED adalah salah satu jenis diode, hanya saja LED akan mengeluarkan cahaya apabila arus mengalir maju. LED digunakan sebagai lampu indicator pada beberapa aplikasi elektronika. LED memiliki sifat dan konsumsi tegangan yang rendah, usia pemakaiannya panjang, dan kecepatan penyakelarnya cepat. LED hampir sama dengan diode biasa. Perbedaannya, jika energy pada diode biasa akan dikeluarkan dalam bentuk panas (disipasi daya), maka energy pada LED akan dikeluarkan dalam bentuk pancaran cahaya. Dalam memilih LED, selain warna, kita juga perlu memperhatikan tegangan kerja, arus maksimum, dan disipasi dayanya. (Suyadhi Taupiq Dwi Septian, 2010 : 31)

LED merupakan komponen yang dapat mengeluarkan emisi cahaya. LED merupakan produk temuan lain setelah dioda. Strukturnya sama dengan dioda, tetapi belakangan ditemukan bahwa elektron yang menerjang sambungan p-n juga melepaskan energi panas dan energi cahaya. Bedanya jika dioda membuang energi kedalam bentuk panas, sedangkan LED membuang energi dalam bentuk cahaya. Bentuk fisik LED *infrared* dapat dilihat pada gambar II.5.



Gambar II.5. Bentuk Fisik LED Infrared

(Sumber : Afrie Setiawan ; 2011 : 11)

LED dapat memancarkan sejumlah kecil dari cahaya ketika arus mengalir pada bias maju. LED dapat dirancang untuk memancarkan cahaya merah, biru, kuning, hijau dan cahaya

inframerah. Cahaya *infrared*/inframerah merupakan cahaya yang tidak tampak. Dengan menggunakan spektroskop cahaya, radiasi inframerah akan tampak pada spektrum elektromagnet dengan panjang gelombang diatas panjang gelombang cahaya inframerah. Dengan panjang gelombang ini, maka cahaya infra merah ini akan tidak tampak oleh mata namun radiasi panas yang ditimbulkan masih terasa atau dideteksi. LED inframerah merupakan salah satu LED yang paling efisien sebagai pembangkit cahaya. Dalam aplikasinya, sensor ini ideal sebagai pendeteksi keamanan. Berkas cahaya dari LED inframerah ini nantinya akan ditangkap oleh *photodiode*. (Afrie Setiawan. 2011 : 11).

II.8. Photodiode

Photodiode dibuat dari semikonduktor dengan bahan yang populer adalah *silicon* (Si) atau *galium arsenida* (GaAs), dan yang lain meliputi InSb, InAs, PbSe. Material ini menyerap cahaya dengan karakteristik panjang gelombang mencakup: 2500 Å - 11000 Å untuk *silicon*, 8000 Å - 20,000 Å untuk GaAs. Ketika sebuah *photon* (satu satuan energi dalam cahaya) dari sumber cahaya diserap, hal tersebut membangkitkan suatu elektron dan menghasilkan sepasang pembawa muatan tunggal, sebuah elektron dan sebuah *hole*, di mana suatu *hole* adalah bagian dari kisi-kisi semikonduktor yang kehilangan elektron. Arah Arus yang melalui sebuah semikonduktor adalah kebalikan dengan gerak muatan pembawa. cara tersebut didalam sebuah *photodiode* digunakan untuk mengumpulkan *photon* menyebabkan pembawa muatan (seperti arus atau tegangan) mengalir/terbentuk di bagian-bagian elektroda.

Photodiode digunakan sebagai penangkap gelombang cahaya yang dipancarkan oleh inframerah. Besarnya tegangan atau arus listrik yang dihasilkan oleh *photodiode* tergantung

besar kecilnya radiasi yang dipancarkan oleh inframerah. Bentuk fisik *photodiode* dapat dilihat pada gambar II.6.



Gambar II.6. Bentuk Fisik Photodiode
(Sumber : Afrie Setiawan ; 2011 : 12)

II.9. Piezoelektrik

Piezoelektrik terjadi jika medan listrik terbentuk ketika material dikenai tekanan mekanik. Pada saat medan listrik melewati material, molekul yang terpolarisasi akan menyesuaikan dengan medan listrik, dihasilkan dipole yang terinduksi dengan molekul atau struktur kristal material. Penyesuaian molekul akan mengakibatkan material berubah dimensi. Fenomena tersebut dikenal dengan electrostriction Efek Piezoelektrik.



Gambar II.7. Bentuk Fisik Piezoelektrik
Sumber : <http://www.kaskus.co.id>

II.10. Bluetooth

Bluetooth sebuah teknologi komunikasi *wireless* (tanpa kabel) yang beroperasi dalam pita frekuensi 2,4 GHz *unlicensed ISM (Industrial, Scientific and Medical)* dengan menggunakan sebuah *frequency hopping tranceiver* yang mampu menyediakan layanan komunikasi data dan suara secara *real-time* antara *host to host bluetooth* dengan jarak jangkauan layanan yang terbatas.^[5]

Bluetooth sendiri dapat berupa *card* yang bentuk dan fungsinya hampir sama dengan card yang digunakan untuk *Wireless Local Area Network (WLAN)* dimana menggunakan frekuensi radio standar IEEE802.11, hanya saja pada *bluetooth* mempunyai jangkauan jarak layanan yang lebih pendek dan kemampuan *transfer* data yang lebih rendah.

Sistem *Bluetooth* terdiri dari sebuah *radio transceiver, baseband link Management dan Control, Baseband (processor core, SRAM, UART, PCM USB Interface), flash* dan *voice code*. sebuah *link manager*. *Baseband link controller* menghubungkan perangkat keras radio ke *baseband processing* dan *layer* protokol fisik. *Link manager* melakukan aktivitas-aktivitas protokol tingkat tinggi seperti melakukan *link setup*, autentikasi dan konfigurasi. Bentuk fisik modul *Bluetooth* HC-06 dapat dilihat pada gambar II.8 di bawah ini :

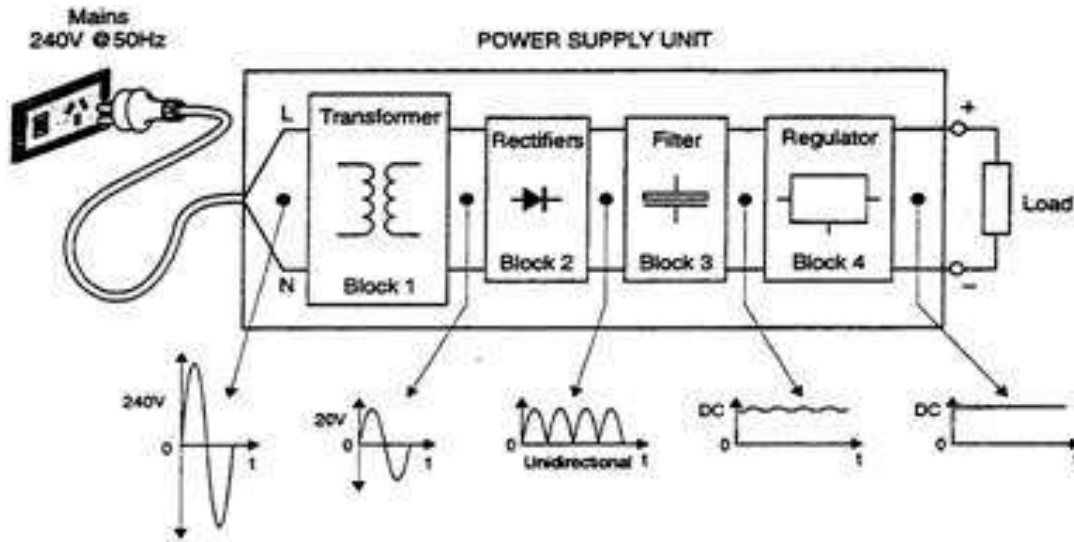


Gambar II.8. Bentuk Fisik Modul Bluetooth HC-06

Sumber : <http://www.botscience.net>

II.11. Catu Daya dan Regulator Tegangan

Perangkat elektronika seharusnya dicatu oleh sumber listrik searah DC (*direct current*) yang stabil agar dapat bekerja dengan baik sesuai dengan kegunaan dan perancangannya. Baterai atau *accu* adalah sumber catu daya DC yang paling baik. Namun apabila digunakan untuk aplikasi yang membutuhkan catu daya lebih besar atau bermacam, sumber dari baterai atau *accu* tidak akan cukup. Sumber catu daya yang lain adalah sumber listrik bolak-balik AC (*alternating current*) dari pembangkit tenaga listrik. Diagram proses catu daya dapat dilihat pada gambar II.9.

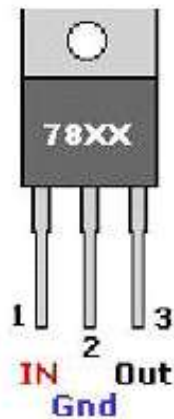


Gambar II.9. Diagram Proses Catu Daya

(Sumber : www.scribd.com di akses pada tanggal 28 Juli 2016)

Transformator diperlukan sebagai komponen yang berfungsi untuk menurunkan tegangan AC dari jala-jala listrik pada kumparan primernya menjadi tegangan AC yang lebih kecil pada kumparan sekundernya. Keluaran transformator yang masih AC kemudian disearahkan oleh untai penyearah (*rectifier*) (Fredy Indra Oktaviansyah 2011).

Regulator seri 7805 adalah *regulator* untuk mendapatkan tegangan keluaran sebesar +5 volt, sedangkan *regulator* seri 7812 adalah untuk mendapatkan tegangan keluaran sebesar +12 volt. Agar rangkaian *regulator* dengan *IC* tersebut dapat bekerja dengan baik, tegangan *input* harus lebih besar dari tegangan *output regulator*-nya. Bentuk Fisik dari *regulator* 78xx dapat dilihat pada gambar II.10. (Fredy Indra Oktaviansyah: 2011).



Gambar II.10. Bentuk Fisik dari Regulator 78xx
(Sumber : Datasheet LM 78xx)

II.12. Bahasa Pemrograman

II.12.1. Bahasa Pemrograman C

Struktur penulisan bahasa C secara umum terdiri atas empat blok, yaitu:

1. *Header*;
2. Deklarasi konstanta global dan atau variabel,
3. Fungsi dan atau prosedur (bisa di bawah program utama),
4. Program utama.

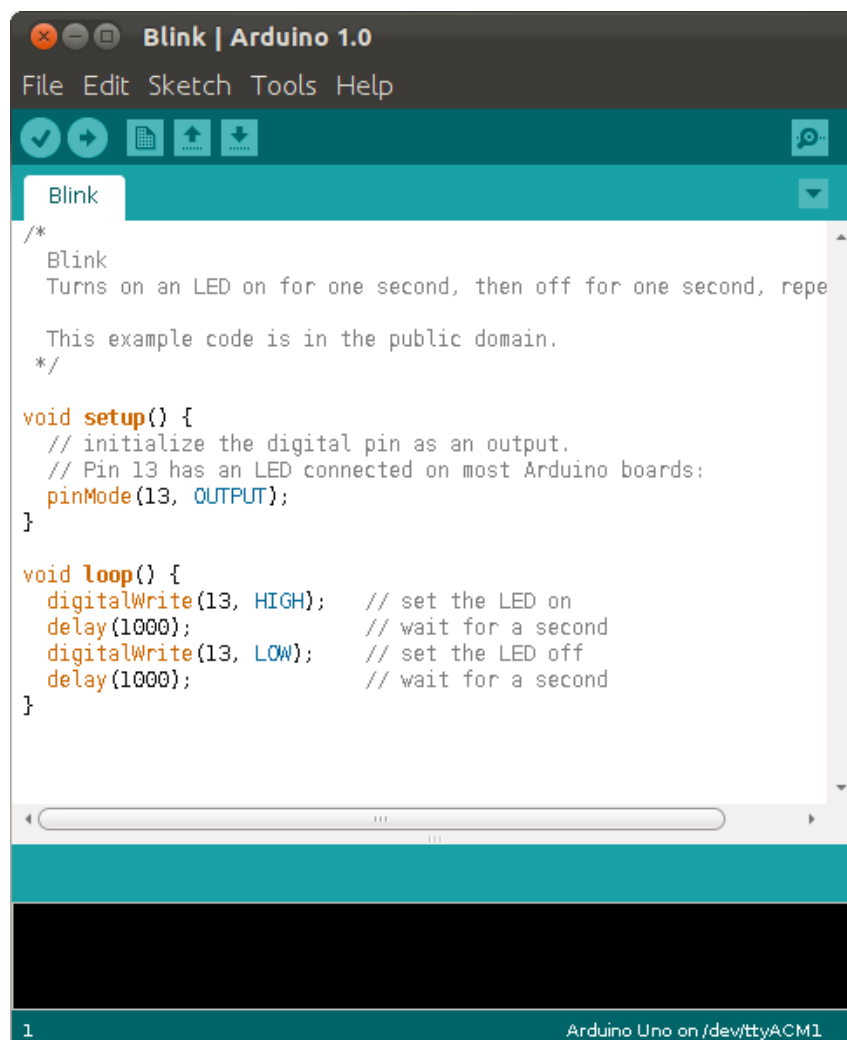
Secara umum, pemrograman C paling sederhana dilakukan dengan hanya menuliskan program utamanya saja, yaitu:

```
/* fungsi utama */
void main()
{
    Statemen-statemen;
}
/* fungsi-fungsi lain yang ditulis oleh pemrogram komputer */
Fungsi_fungsi_lain()
{
    Statemen-statemen;
}
```

(M. Ary Heryanto, ST dan Ir. Wisnu Adi P. (2008: 8))

II.12.2. Pemrograman *IDE* Arduino)

Arduino memiliki open-source yang memudahkan untuk menulis kode dan meng-upload board ke arduino. Arduino *IDE* (*Integrated Development Enviroment*) ini merupakan media yang digunakan untuk memberikan informasi kepada arduino sehingga dapat memberikan output sesuai dengan apa yang diinginkan. Aplikasi arduino *IDE* ini dapat dijalankan di windows, Mac OS X, dan linux (Moh. Kamalul Wafi, 2014: 2). berikut merupakan gambaran tampilan arduino *IDE* :

The image shows a screenshot of the Arduino IDE window titled "Blink | Arduino 1.0". The window has a menu bar with "File", "Edit", "Sketch", "Tools", and "Help". Below the menu bar is a toolbar with icons for opening files, saving, and uploading. The main area displays the code for a "Blink" sketch. The code includes a comment block, a setup function that initializes pin 13 as an output, and a loop function that toggles the LED on and off with 1000ms delays. At the bottom, there is a status bar showing "1" and "Arduino Uno on /dev/ttyACM1".

```
/*
 * Blink
 * Turns on an LED on for one second, then off for one second, repeatedly.
 *
 * This example code is in the public domain.
 */

void setup() {
  // initialize the digital pin as an output.
  // Pin 13 has an LED connected on most Arduino boards:
  pinMode(13, OUTPUT);
}

void loop() {
  digitalWrite(13, HIGH); // set the LED on
  delay(1000);           // wait for a second
  digitalWrite(13, LOW); // set the LED off
  delay(1000);           // wait for a second
}
```

Gambar II.11. Tampilan Arduino IDE

(Sumber: arduino.stackexchange.com dikunjungi pada tanggal 29 Juli 2016)

Dalam arduino terhubung dengan arduino *IDE* ini dengan hanya menekan tombol RESET. tombol ini dirancang untuk menjalankan program yang telah di upload ke arduino *board*, tombol ini juga terhubung dengan ATMEga 328 melalui kapasitor 100nf. *IDE (Integrated Development Enviroment)* arduino merupakan pemograman dengan menggunakan bahasa C. Setiap program *IDE* arduino (biasa disebut *sketch*) mempunyai dua buah fungsi yang harus ada, yaitu :

a. `void setup() { }`

Semua kode didalam kurung kurawal akan dijalankan hanya satu kali ketika program *IDE* Arduino dijalankan untuk pertama kalinya.

b. `void loop() { }`

Fungsi ini akan dijalankan setelah setup (fungsi *void setup*) selesai. Setelah dijalankan satu kali fungsi ini akan dijalankan lagi, dan lagi secara terus menerus sampai catu daya (power) dilepaskan.

Compiler merupakan modul yang berfungsi mengubah bahasa *processing* (kode pemograman) kedalam kode biner karena kode biner adalah satu-satunya bahasa program yang dipahami oleh mikrokontroler. Sedangkan *upload* program adalah modul yang berfungsi memasukkan kode biner kedalam mikrokontroler. Pada *software* Arduino *IDE* memiliki fitur *compiler* sedangkan untuk *upload* program menggunakan *USBisp* yang dihubungkan ke port *ISP* pada papan rangkaian mikrokontroler. Pada proses ini akan merubah bahasa pemograman dari digital ke bahasa analog yang dapat dipahami mikrokontroler (Anandya Bagus Venesa dan Wibowo Basuki Dwi, 2004 : 5).

II.12.3. Android

Android adalah sistem operasi untuk perangkat selular yang berbasis Linux yang mencakup sistem operasi, *middleware* dan aplikasi.

Android menyediakan *platform* terbuka bagi para pengembang buat menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak. Awalnya, Google Inc. membeli Android Inc.pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan Android, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia. Pada saat perilisan perdana Android, 5 November 2007, Android bersama *Open Handset Alliance* menyatakan mendukung pengembangan standar terbuka pada perangkat seluler. Dilain pihak, Google merilis kode-kode Android di bawah lisensi *Apache*, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler. Di dunia ini terdapat dua jenis distributor sistem operasi Android. Pertama yang mendapat dukungan penuh dari Google atau *Google Mail Services* (GMS) dan kedua adalah yang benar-benar bebas distribusinya tanpa dukungan langsung Google atau dikenal sebagai *Open Handset Distribution* (OHD).

II.12.3.1. Fitur – fitur Android

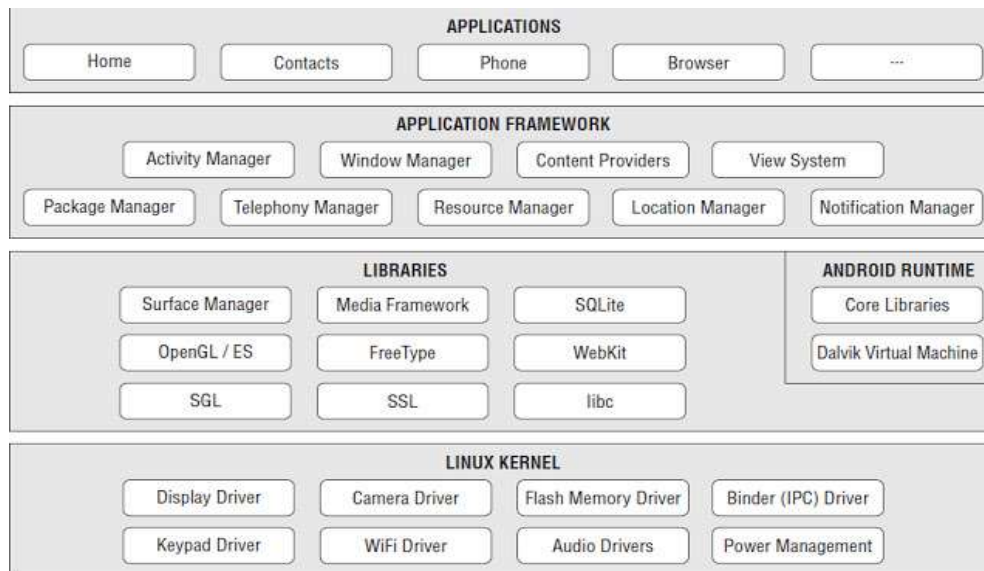
Android tersedia secara open source bagi manufaktur perangkat keras untuk modifikasinya sesuai kebutuhan. Meskipun konfigurasi perangkat Android tidak sama antara satu perangkat dengan perangkat lain. Fitur yang tersedia Android adalah :

1. Penyimpanan (*Storage*) : menggunakan SQLite yang merupakan database *relational* yang ringan untuk menyimpan data.
2. Koneksi (*Connectivity*) : mendukung *GSM/EDGE*, *IDEN*, *CDMA*, *EV-DO*, *UMTS*, *Bluetooth* (termasuk *A2DP* dan *AVRCP*), *Wifi*, *LTE*, dan *WiMax*.

3. Pesan (*Messaging*) : mendukung SMS dan MMS.
4. Mendukung Media : audio, video, dan berbagai format gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF).
5. *Web Browser* : menggunakan *open source WebKit* termasuk di dalamnya *engine Chrome V8 JavaScript*.
6. *Hardware* : terdapat *Accelerometer Sensor*, *Camera*, *Digital Compass*, *Proximity Sensor* dan *GPS*.
7. *Multi touch* : mendukung layar *multi touch*.
8. *Multi tasking* : mendukung aplikasi *multi tasking*.

II.12.3.2. Arsitektur Android

Agar lebih mudah memahami bagaimana Android bekerja, berikut ini bagan tingkatan – tingkatan sistem operasi Android :



Gambar II.12. Arsitektur Sistem Operasi Android

Sumber : (*Pemrograman Aplikasi Android*)

Secara garis besar system operasi Android terbagi menjadi lima tingkatan :

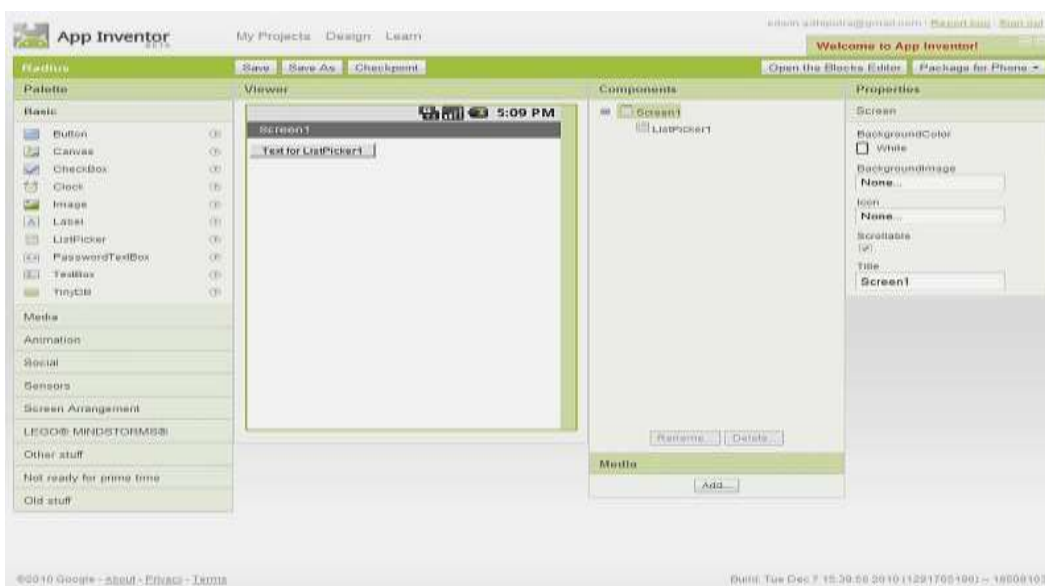
1. **Applications** pada tingkat inilah kita akan bekerja, contoh aplikasi ini banyak ditemui, seperti : *Phone, Contact, Browse* dan lain – lain. Seperti aplikasi Android pada umumnya yang dapat di *download* dan di *install* dari Market Android. Semua aplikasi yang anda buat terletak pada tingkat *Applications*.
2. **Application Framework** adalah semacam kumpulan class built-in yang tertanam dalam sistem operasi Android sehingga pengembang dapat memanfaatkannya untuk aplikasi yang sedang dibangun.
3. **Libraries** berisi semua kode program yang menyediakan layanan – layanan utama sistem operasi Android. Sebagai contoh library SQLite yang menyediakan dukungan database sehingga aplikasi Android dapat menggunakannya untuk menyimpan data. Library WebKit yang menyediakan fungsi – fungsi *browsing web*, dan lain – lain.
4. **Android Runtime** kedudukannya setingkat dengan libraries, *Android Runtime* menyediakan kumpulan pustaka inti yang dapat diaktifkan oleh pengembang untuk menulis kode aplikasi Android dengan bahasa pemrograman Java. *Dalvik Virtual Machine* setiap kali aplikasi Android berproses (aplikasi Android dikompilasi menjadi *Dalvik executable*). *Dalvik* adalah mesin semu yang dirancang khusus untuk Android yang dapat mengoptimalkan daya *battery* perangkat bergerak dengan memori dan *CPU* terbatas.
5. **Linux kernel** adalah kernel dasar Android. Tingkat ini berisi semua *driver* perangkat tingkat rendah untuk komponen – komponen *hardware* perangkat Android.

II.12.4. App Inventor

App Inventor for Android adalah aplikasi yang pada dasarnya disediakan oleh Google dan sekarang di-*maintenance* oleh Massachusetts Institute of Technology (MIT). App Inventor memungkinkan semua orang untuk membuat *software* aplikasi untuk sistem operasi android. Pengguna dapat menggunakan tampilan grafis GUI dan fitur drag and drop visual objek untuk membuat sebuah aplikasi dapat berjalan pada sistem operasi Android.

App Inventor adalah sebuah pemrograman visual yang digunakan untuk mengembangkan aplikasi berbasis Android dengan dukungan fitur berupa drag and drop tool. Anda dapat mendesain user *interface* dari sebuah aplikasi dengan menggunakan web GUI (*Graphical User Interface*) *builder*, kemudian anda dapat menspesifikasikan behavior aplikasi dengan memasang *block* yang sesuai dengan kebutuhan anda.

App Inventor menggunakan *Kawa Language Framework* dan *Kawa's dialect* yang dikembangkan oleh Per Botner. Kedua aplikasi tersebut didistribusikan sebagai bagian dari GNU Operating System oleh *Free Software Foundation*. Kedua aplikasi tersebut dijadikan sebagai compiler dan menerjemahkan *Visual Block Programming* untuk di implementasikan pada *platform* Android.









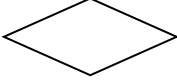
Gambar II.13. Tampilan Awal App Inventor



(Sumber : Wahana Komputer : 2013 : 3)

II.13. Flowchart

Prinsip kerja dari pemodelan di atas dapat digambarkan melalui *flowchart*. *Flowchart* adalah gambar atau bagan yang memperlihatkan urutan dan hubungan antar proses beserta pernyataannya. Gambaran ini dinyatakan dengan simbol dan dengan demikian setiap simbol menggambarkan proses tertentu. Hubungan antar proses digambarkan dengan garis penghubung (Zarlis et al, 2007). Flowchart disusun dengan simbol-simbol. Simbol ini dipakai sebagai alat bantu menggambarkan proses di dalam program. Adapun tabel simbol *flowchart* pada Tabel II.1:

Tabel II.1. Tabel Simbol Flowchart

SIMBOL	NAMA	FUNGSI
	TERMINATOR	Permulaan/akhir program
	GARIS ALIR (FLOW LINE)	Arah aliran program
	PREPARATION	Proses inialisasi/pemberian harga awal
	PROSES	Proses perhitungan/proses pengolahan data
	INPUT/OUTPUT DATA	Proses input/output data, parameter, informasi
	PREDEFINED PROCESS (SUB PROGRAM)	Permulaan sub program/proses menjalankan sub program
	DECISION	Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya

	<p align="center">ON PAGE CONNECTOR</p>	<p align="center">Penghubung bagian-bagian flowchart yang berada pada satu halaman</p>
	<p align="center">OFF PAGE CONNECTOR</p>	<p align="center">Penghubung bagian-bagian flowchart yang berada pada halaman berbeda</p>

(Sumber : Dr. Suarga, M. sc., M. Math., Ph D., 2012.)