

BAB II

TINJAUAN PUSTAKA

2.1 Definisi Robot

Istilah robot berasal dari bahasa Ceko Slowakia. Kata robot berasal dari kosakata “Robota” yang berarti “kerja cepat”. Istilah ini muncul pada tahun 1921 oleh seorang pengarang sandiwara bernama Karel Capek. Karyanya pada saat itu berjudul “Rossum’s Universal Robots” yang artinya Robot Dunia milik Rossum.

Robotik adalah ilmu yang mematerikan kecerdasan atau itelegensia terhadap energi, artinya pengendalian secara cerdas terhadap gerakan yang terkoordinasi secara nyata. Kata Robotics juga berasal dari novel fiksi sains “Runaround” yang ditulis oleh Isaac Asimov pada tahun 1942. Sedangkan pengertian robot secara tepat adalah sistem atau alat yang dapat berperilaku atau meniru perilaku manusia dengan tujuan untuk menggantikan dan mempermudah kerja/aktifitas manusia. Untuk dapat diklasifikasikan sebagai robot, mesin harus memiliki dua macam kemampuan yaitu bisa mendapatkan informasi dari sekelilingnya dan bisa melakukan sesuatu secara fisik seperti bergerak atau memanipulasi objek (Christian, 2009).

- Menurut Japanese Industrial Robot Association (JIRA) robot diklasifikasikan menjadi 5 (lima) bagian (Suwito, 2009) yaitu :
 1. Play-back Robot, adalah robot yang menjalankan fungsi-fungsi berulang yang telah direkamkan kepadanya, biasanya memiliki sistem kontrol open loop.
 2. Robots controlled by sensors, adalah robot yang memiliki loopback yang diakibatkan oleh gerakan dan membuat keputusan berdasarkan data yang diperoleh dari sensor.
 3. Robot Vision adalah robot yang informasi diperoleh dari sistem vision, dimana robot dapat memanipulasi objek yang ditangkap.
 4. Robot controlled adaptably adalah robot yang dapat secara otomatis memprogram aksinya sendiri berdasarkan data yang diperoleh dari sensor.

5. Intelligent Robot, yaitu robot yang menggunakan teknik artificial intelligence untuk membuat keputusan dan mampu memecahkan permasalahannya sendiri.

2.2 Arduino

Arduino adalah sistem punarupa elektronika (*electronic prototyping platform*) berbasis *open-source* yang fleksibel dan mudah digunakan baik dari sisi perangkat keras/*hardware* maupun perangkat lunak/*software*. Di luar itu, kekuatan utama arduino adalah jumlah pemakai yang sangat banyak sehingga tersedia pustaka kode program (*code library*) maupun modul pendukung (*hardware support modules*) dalam jumlah yang sangat banyak. Hal ini memudahkan para pemula untuk mengenal dunia mikrokontroler.

Arduino didefinisikan sebagai sebuah *platform* elektronik yang *open source*, berbasis pada *software* dan *hardware* yang fleksibel dan mudah digunakan, yang ditujukan untuk seniman, desainer, *hobbies* dan setiap orang yang tertarik dalam membuat sebuah objek atau lingkungan yang interaktif (Sumber: Artanto,2012:1).

Arduino sebagai sebuah *platform* komputasi fisik (*Physical Computing*) yang *open source* pada *board input ouput* sederhana, yang dimaksud dengan *platform* komputasi fisik disini adalah sebuah sistem fisik yang interaktif dengan penggunaan *software* dan *hardware* yang dapat mendeteksi dan merespon situasi dan kondisi.

Kelebihan arduino dari *platform hardware* mikrokontroller lain adalah:

1. IDE Arduino merupakan *multiplatform*, yang dapat dijalankan di berbagai sistem operasi, seperti *Windows*, *Macintosh* dan *Linux*.
2. IDE Arduino dibuat berdasarkan pada IDE *Processing* sederhana sehingga mudah digunakan.
3. Pemrograman Arduino menggunakan kabel yang terhubung dengan *port* USB bukan *port* serial. Fitur ini berguna karena banyak komputer sekarang ini tidak memiliki *port* serial.

4. Arduino adalah *hardware* dan *software open source*, pembaca bisa mendownload *software* dan gambar rangkaian arduino tanpa harus membayar ke pembuat arduino.
5. Biaya *hardware* cukup murah, sehingga tidak terlalu menakutkan untuk membuat kesalahan.
6. Proyek arduino ini dikembangkan dalam lingkungan pendidikan sehingga bagi pemula akan lebih cepat dan mudah mempelajarinya.
7. Memiliki begitu banyak pengguna dan komunitas di internet dapat membantu setiap kesulitan yang dihadapi terutama oleh programmer pemula. (Sumber: Artanto, 2012:2)

2.2.1 Sejarah Arduino

Modul *hardware* Arduino diciptakan pertama kali di Ivrea, Italia pada tahun 2005 oleh Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino, David A. Mellis, dan Nicholas Zambetti (Arduino, 2011 dan Banzi 2008).

Bahasa Arduino merupakan *fork* (turunan) bahasa *Wiring Platform* dan bahasa *Processing*. *Wiring Platform* diciptakan oleh Hernando Barragan pada tahun 2003 (Wiring.org.co, 2011) dan *Processing* dibuat oleh Casey Reas dan Benjamin Fry pada tahun 2011 (Processing.org, 2012).

Arduino dikembangkan dari thesis Hernando Barragan di desain interaksi Institute Ivrea. Arduino dapat menerima masukan dari berbagai macam sensor dan juga dapat mengontrol lampu, motor, dan aktuator lainnya. Mikrokontroler pada *board* arduino di program menggunakan bahasa pemrograman arduino (*based on wiring*) dan IDE arduino (*based on processing*). Proyek arduino dapat berjalan sendiri atau juga bisa berkomunikasi dengan *software* yang berjalan pada komputer.

Arduino memakai standar lisensi *open source*, mencakup *hardware* (skema rangkaian, desain PCB), *firmware bootloader*, dokumen, serta perangkat lunak IDE (*Integrated Development Environment*) sebagai aplikasi programmer *board* Arduino. Setiap modul arduino menggunakan seri mikrokontroler yang berbeda

seperti misalnya arduino leonardo yang menggunakan mikrokontroler ATmega328/32U4.

Menggunakan Arduino sangatlah membantu dalam membuat suatu *prototyping* ataupun untuk melakukan pembuatan proyek. Arduino memberikan input output (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 rancang hardware yang ada. Software jauh lebih mudah untuk dimodifikasi tanpa harus memindahkan kabel. Saat ini arduino sangat mudah dijumpai dan ada beberapa perusahaan yang mengembangkan sistem hardware *open source* ini. (Sumber: Djuandi, 2011:4).

2.2.2 Arduino Mega 2560

Arduino mega 2560 adalah papan mikrokontroler Atmega 2560 berdasarkan (*datasheet*) memiliki 54 digital pin *input* atau *output* (dimana 15 pin dapat digunakan sebagai *output* PWM atau *Pulse Width Modulation*), 16 *analog input*, 4 UART (*Universal Asynchronous Receiver/Transmitter*), *osilator* kristal 16 MHz, koneksi USB, *jack* listrik, *header ICSP (In-Circuit Serial Proqraming)*, dan tombol *reset*. Semuanya diperlukan untuk mendukung mikrokontroler, hanya menghubungkannya ke komputer dengan kabel USB atau *power* dengan adaptor AC (*Alternating Current*) – DC (*Direct Current*) atau baterai. Gambar 2.4 menunjukkan bentuk fisik Arduino Mega 2560. Adapun spesifikasi singkat mengenai Arduino Mega 2560 adalah sebagai berikut :

- | | |
|-------------------------------------|--------------|
| 1. Mikrokontroler | : ATmega2560 |
| 2. Tegangan Operasional | : 5V |
| 3. Tegangan Masukan (direkomendasi) | : 7-12V |
| 4. Tegangan Masukan (batas) | : 6-20V |
| 5. Pin Digital I/O | : 54 |
| (14 pin untuk keluaran PWM) | |

6. Analog Input Pins	: 16
7. Arus DC per I/O Pin	: 40 mA
8. Arus DC for 3.3V Pin	: 50 mA
9. Memori Flash	: 256 KB
(8 KB digunakan untuk <i>bootloader</i>)	
10. SRAM	: 8 KB
11. EEPROM	: 4 KB
12. Clock Speed	: 16 MHz



Gambar 2.1 Arduino Mega 2560

(Sumber : <http://www.arduino.cc>,2015)

Arduino Mega 2560 berbeda dari semua papan sebelumnya dalam hal itu tidak menggunakan FTDI *chip driver* USB-to-serial. Sebaliknya, fitur Atmega 16U2 (Atmega 8U2 dalam *board* revisi 1 dan revisi 2) diprogram sebagai *converter* USB-to-serial. Revisi 2 dari Arduino Mega 2560 memiliki resistor menarik garis 8U2 HWB ke *ground*, sehingga lebih mudah untuk dimasukkan ke dalam mode DFU (*Device Firmware Update*). Revisi 3 dari arduino mega 2560 memiliki fitur-fitur baru sebagai berikut:

- a. 1,0 *pinout* tambah SDA (*Serial Data*) dan SCL (*Serial Clock*) pin yang dekat dengan pin AREF (*ADC Reference*) dan dua pin baru lainnya ditempatkan dekat dengan pin *reset*.
- b. Sirkuit *reset* lebih kuat.
- c. Atmega 16U2 menggantikan 8U2.

2.2.3 Power

Arduino mega 2560 dapat diaktifkan melalui koneksi USB atau dengan catu daya eksternal. Sumber daya dipilih secara manual.

Daya eksternal (non-USB) dapat berasal baik dari adaptor AC (*Alternating Current*)-DC (*Direct Current*) atau baterai. Adaptor ini dapat dihubungkan dengan steker 2.1mm pusat-positif ke sumber listrik.

Papan arduino mega 2560 dapat beroperasi pada tegangan 6 sampai 20 VDC. Jika menggunakan lebih dari 12V, regulator tegangan bisa panas dan merusak *board arduino*. Kisaran yang dianjurkan adalah 7 sampai 12 volt. Pin listrik *board arduino mega 2560* adalah sebagai berikut:

a. VIN

Tegangan *input* ke papan arduino menggunakan sumber daya eksternal (sebagai lawan 5 volt dari koneksi USB atau sumber daya yang diatur lainnya). Jika ingin menyediakan tegangan eksternal dapat digunakan pin ini.

b. 5V

Pin *output* 5V ini diatur dari regulator di *board* dapat diaktifkan dengan daya baik dari colokan listrik DC (7 - 12V), konektor USB (5V), atau pin VIN dari *board* (7-12V).

c. 3.3V

Sebuah pasokan 3,3 volt yang dihasilkan oleh *regulator on-board*. Arus maksimum adalah 50 mA.

d. *Ground*

Pin *ground*.

e. AREF

Pin ini menyediakan tegangan referensi untuk operasi mikrokontroler. Sebuah *shield* dikonfigurasi dengan benar agar dapat membaca tegangan pada pin AREF dan memilih sumber daya yang tepat atau mengaktifkan *voltage translator* pada *output* untuk bekerja dengan 5V atau 3.3V.

2.2.4 *Input dan Output*

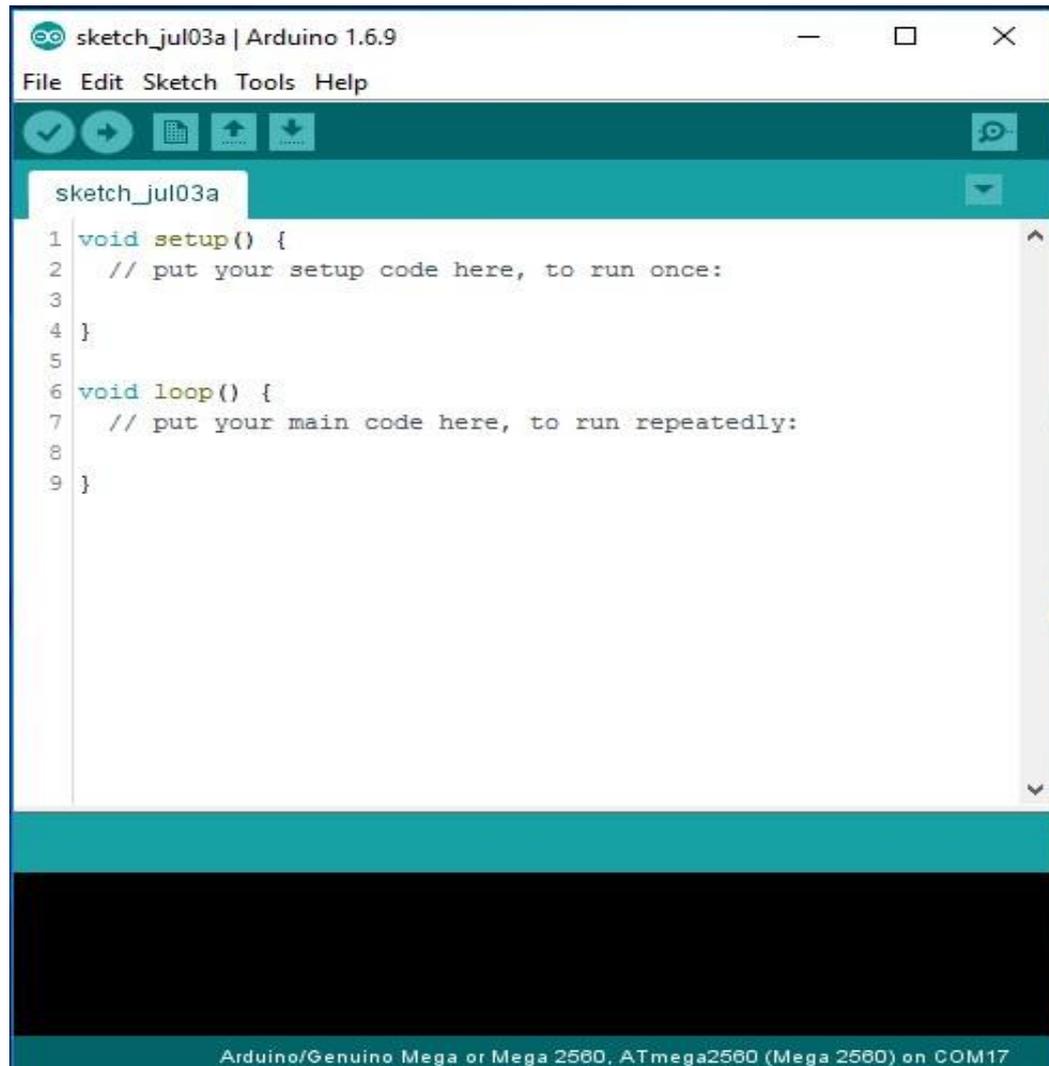
Masing-masing dari 54 pin digital pada arduino mega 2560 dapat digunakan sebagai *input* atau *output*, menggunakan fungsi `pinMode ()`, `digitalWrite ()`, dan `digitalRead`. Selain itu, beberapa pin memiliki fungsi khusus sebagai berikut:

1. Serial: 0 (RX) dan 1 (TX); Serial 1: 19 (RX) dan 18 (TX); Serial 2: 17 (RX) dan 16 (TX); Serial 3: 15 (RX) dan 14 (TX). Digunakan untuk menerima (RX) dan mengirimkan (TX) TTL data serial. Pin 0 dan 1 juga terhubung ke pin yang sesuai dari Atmega 16U2 USB- to-Serial TTL.
2. *External Interrupts*: 2 (*interrupt 0*), 3 (*interrupt 1*), 18 (*interrupt 5*), 19 (*interrupt 4*), 20 (*interrupt 3*), dan 21 (*interrupt 2*). Pin ini dapat dikonfigurasi untuk memicu interupsi pada nilai yang rendah, yang naik atau jatuh tepi, atau perubahan nilai. Lihat fungsi `attachInterrupt ()` untuk lebih rinci.
3. PWM: 2-13 dan 44 sampai 46. Memberikan 8-bit PWM output dengan fungsi `analogWrite ()`.
4. SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS). Pin ini mendukung komunikasi SPI (*Serial Peripheral Interface*) menggunakan *library* SPI. Pin SPI juga terpisah dari *header* ICSP, yang secara fisik kompatibel dengan Uno, Duemilanove dan Diecimila.
5. LED: 13. Ada *built-in* LED (*Light Emiting Diode*) terhubung ke pin digital 13. Ketika logika pin bernilai nilai tinggi atau *high*, LED akan menyala, ketika logika pin rendah atau *low*, maka LED akan mati atau off.

Arduino mega 2560 memiliki 16 *input* analog, yang masing-masing menyediakan 10 bit resolusi (yaitu 1024 nilai yang berbeda). Secara *default* nilai tersebut dari 0 sampai 5 volt, meskipun adalah mungkin untuk mengubah nilai jangkauan atas (5V) dengan menggunakan pin AREF dan fungsi `analogReference ()` fungsi.

2.2.5. Aplikasi Program IDE (Integrated Development Environment)

Ketika kita membuka program Arduino IDE (*Integrated Development Environment*), akan terlihat serupa dengan tampilan gambar 2.6 di bawah ini. Jika kita menggunakan Windows atau Linux, akan terlihat perbedaan, tetapi pada dasarnya IDE akan sama, tidak peduli Operasi Sistem apa yang digunakan.

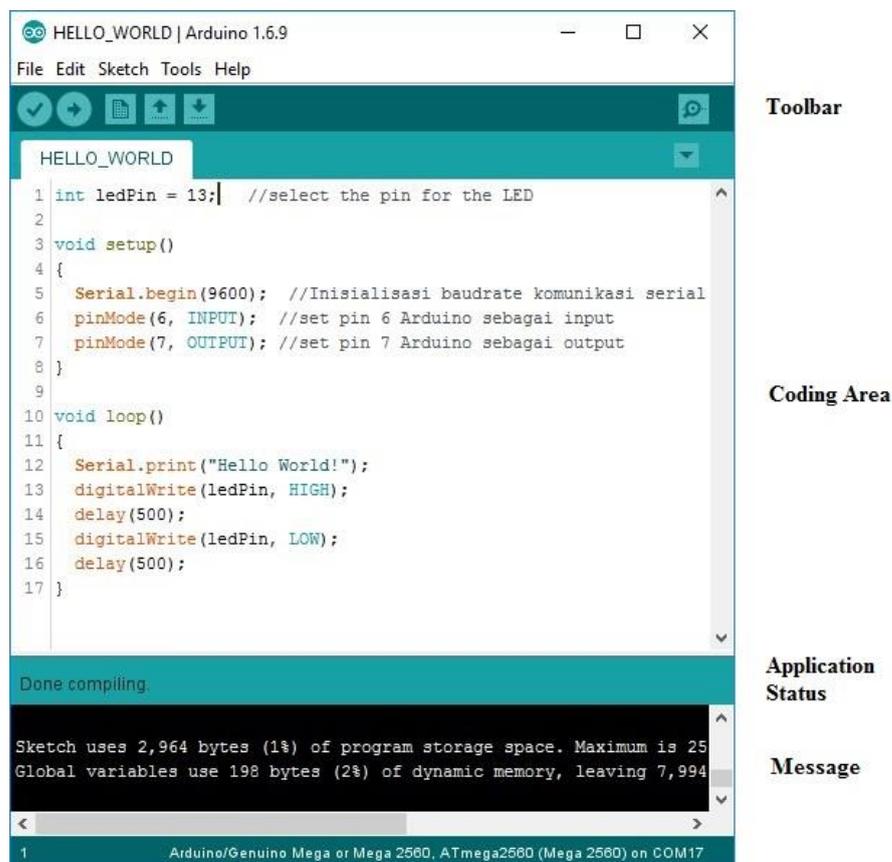


Gambar 2.2 Tampilan Program IDE

(Sumber : <http://www.arduino.cc> , 2015)

2.2.6 Arduino Programming Tool

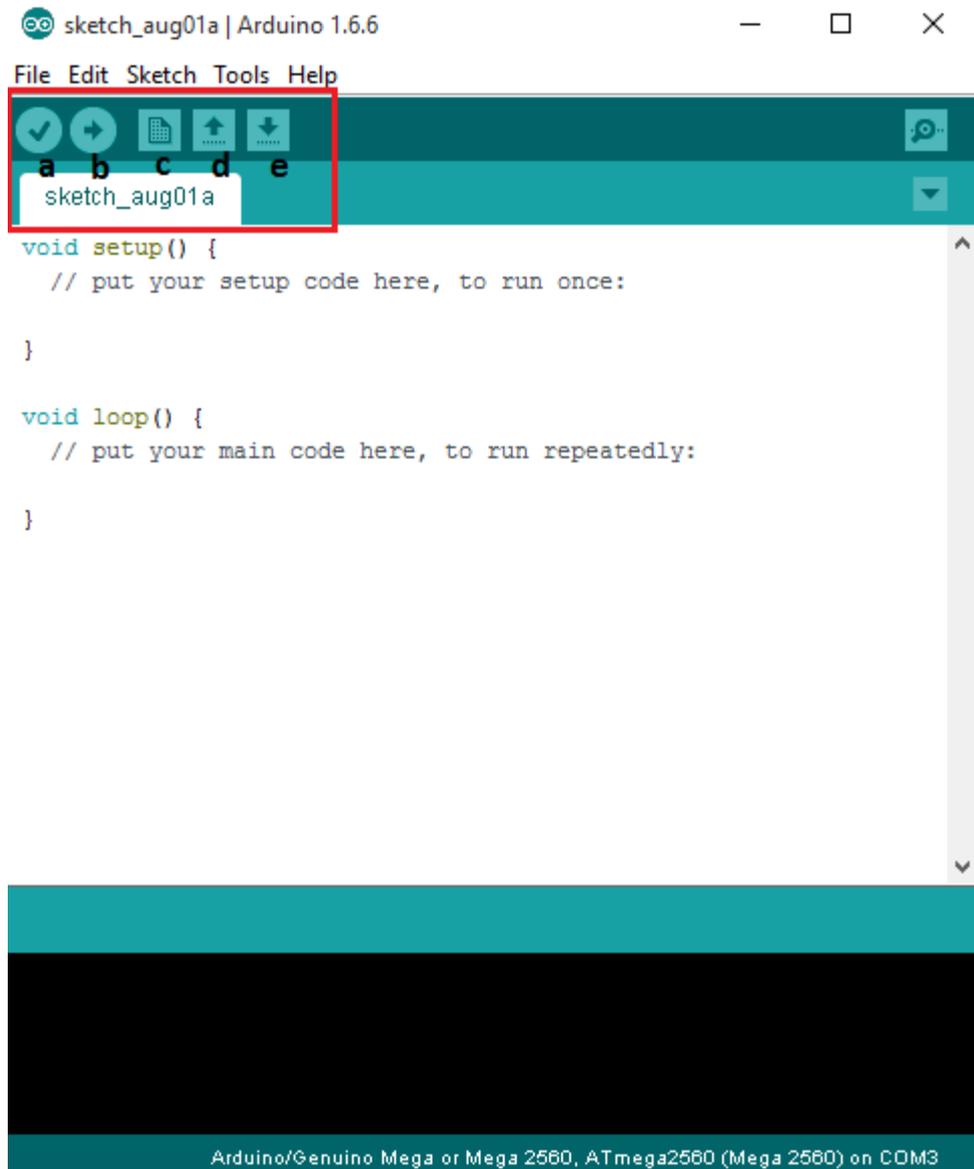
Arduino merupakan perangkat pemrograman mikrokontroler jenis AVR yang tersedia secara bebas (*open source*) untuk membuat prototip elektronika yang dapat berinteraksi dengan keadaan sekitarnya. Arduino dapat menerima input dari berbagai jenis sensor dan mengendalikan sensor, servo, dan actuator lainnya.



Gambar 2.3 Tampilan Utama Aplikasi Arduino
(Sumber : <http://www.arduino.cc> , 2015)

1. *Toolbar*

Tombol-tombol toolbar memungkinkan Anda untuk memverifikasi dan meng-upload program, membuat, membuka, dan menyimpan sketsa, juga membuka monitor serial.



Gambar 2.4 Toolbar Pada Aplikasi Arduino
(Sumber : <http://www.arduino.cc> , 2015)

a. *Verify*

Tombol ini digunakan untuk meng-*compile* program yang telah dibuat. *Compile* berguna untuk mengetahui apakah program yang telah

dibuat benar atau masih memiliki kesalahan. Apabila ada kesalahan yang terjadi, bagian *message* akan menampilkan letak kesalahan tersebut.

b. Upload

Tombol ini digunakan untuk mengirim *coding* yang sudah dikerjakan ke mikrokontroler.

c. New

Tombol ini digunakan untuk membuat *coding* pada layar baru

d. Open

Tombol ini digunakan untuk membuka *coding* yang sudah disimpan sebelumnya.

e. Save

Tombol ini digunakan untuk menyimpan *coding* yang sedang dikerjakan.

f. Serial Monitor

Tombol ini digunakan untuk melihat aktivitas komunikasi serial dari mikrokontroler baik yang dikirim oleh *user* ke mikrokontroler maupun sebaliknya.

2. Coding Area

Bagian ini merupakan tempat penulisan *coding* dengan menggunakan bahasa pemrograman C. *Coding* di dalam Arduino memiliki dua bagian utama, yaitu :

a. void setup ()

Bagian ini merupakan inisialisasi yang diperlukan sebelum program utama dijalankan, contoh :

```
1.void setup ( ){
2.Serial.begin (9600) ; // Inisialisasi baudrate komunikasi
3.serial
4.pinMode (6, INPUT) ; // set pin 6 Arduino sebagai input
5.pinMode (7, OUTPUT) ; // set pin 7 Arduino sebagai output }
```

b. `void loop ()`

Bagian ini merupakan fungsi utama yang dijalankan terus menerus selama modul Arduino terhubung dengan power *supply*. Contoh :

```

1.voidloop ( )
2.{
3.digitalWrite (6, HIGH) ;
4.delay (1000) ; // menunda selama 1 detik
5.digitalWrite (6, LOW) ;
6.delay (2000) ; // menunda selama 2 detik
7.}

```

c. *Application Status*

Bagian ini memberikan informasi kepada pengguna mengenai tugas yang sedang dijalankan oleh aplikasi Arduino.

d. *Message*

Bagian ini memberikan informasi kepada pengguna mengenai besarnya ukuran file dari *coding* yang dibuat dan letak kesalahan yang terjadi pada *coding*.

2.2.7 Tipe-Tipe data dalam Arduino

Setiap bagian dari data yang anda simpan dalam program arduino memiliki tipe datanya masing-masing. Tergantung pada kebutuhan anda, anda dapat memilih dari tipe-tipe data berikut ini :

- a. Tipe data boolean mengambil satu byte memori dan dapat bernilai benar atau salah.
- b. Tipe data *char* mengambil satu byte nomor memori dan menyimpan dari -128 sampai 127. Angka-angka ini biasanya mewakili karakter yang dikodekan dalam ASCII.

- c. Tipe data *int* (*integer*) membutuhkan dua *byte* memori. Anda dapat menggunakannya untuk menyimpan angka dari -32.768 ke 32.767. *unsigned int* juga menghabiskan dua *byte* memori tetapi menyimpan angka dari 0 sampai 65.535.
- d. Untuk angka yang lebih besar, digunakan tipe data *long*. Mengonsumsi empat *byte* memori dan menyimpan nilai dari -214783648 ke 2147483647. *Unsigned long* juga perlu empat *byte* tetapi menyimpan rentang nilai dari 0 sampai 4.294.967.295.
- e. Tipe data *float* dan *double* adalah tipe data yang sama. Anda dapat menggunakan jenis tipe ini untuk menyimpan angka floating-point. Keduanya menggunakan empat byte memori dan mampu menyimpan nilai-nilai dari -3.4028235E+38 untuk 3.4028235E+38.
- f. Tipe data *void* hanya untuk deklarasi fungsi. Ini menunjukkan bahwa fungsi tersebut tidak mengembalikan nilai.
- g. *Array* menyimpan nilai yang memiliki tipe data yang sama.
- h. Sebuah *string* adalah sebuah *array* nilai *char*. Arduino IDE mendukung penciptaan *string* dengan beberapa sintaksis gula semua ini deklarasi membuat *string* dengan isi yang sama.

2.2.8 Komplikasi dan Program *Uploading*

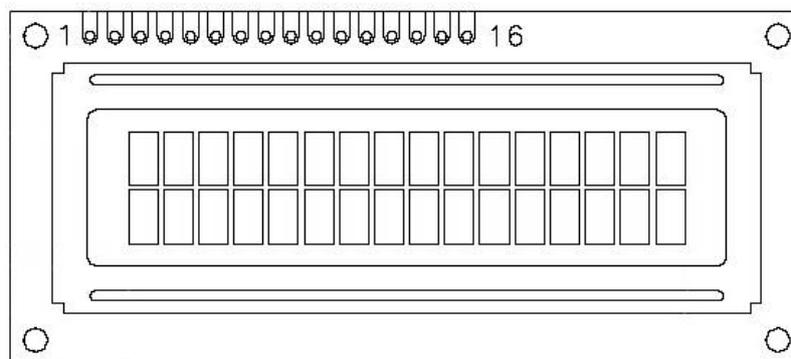
Sebelum anda mengkompilasi dan meng-*upload* program ke arduino, anda harus mengkonfigurasi dua hal dalam IDE : jenis Arduino anda menggunakan dan port serial arduino anda terhubung. Ketika anda telah mengidentifikasi dengan tepat jenis arduino anda. Memilih dari menu *tools>board*. Sekarang anda harus memilih *port* serial arduino anda terhubung untuk dari >menu *serial port tools*. Pada sistem *windows*, *Device Manager*, dan mencari USB Serial *Port* dibawah *ports* (COM dan LPT) entri menu. Biasanya *port* bernama COM1, COM2, atau sesuatu yang serupa. Setelah anda telah memilih *port* serial kanan, klik tombol *verify* dan anda akan melihat *output* berikut di daerah pesan IDE (yang arduino IDE menyebut program sketsa)

Selama proses upload, TX dan RX LED akan berkedip selama beberapa

detik. Ini adalah normal itu terjadi setiap kali Arduino dan komputer anda berkomunikasi melalui port serial. Ketika arduino mengirimkan informasi ternyata pada TX LED. Ketika mendapat beberapa bit, ternyata pada RX LED. (M Bangun Agung. 2014).

2.3. LCD LMB162A Sebagai Output

LMB162A adalah piranti output berupa LCD untuk sistem komputer atau mikrokontroler yang dapat menampilkan karakter standar yang telah tersimpan dalam ROM pada LCD tersebut. Karakter yang dapat ditampilkan sebanyak 32 karakter yang tersusun dalam 16 kolom dan dua baris, sehingga sering disebut LCD 16x2. LCD ini memiliki 16 pin dan memerlukan tegangan 5V DC. Sebelum menggunakan modul LCD ini, power supply 5V DC harus diberikan sebagai sumber arusnya.

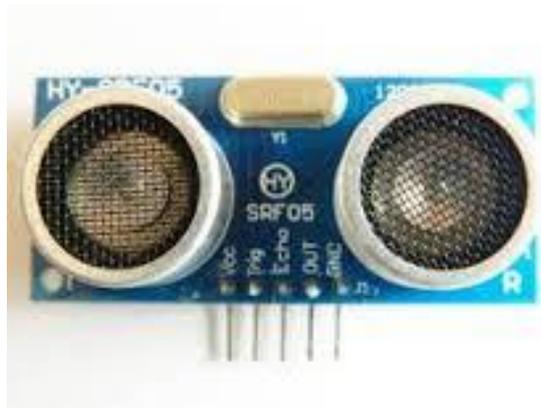


Gambar 2.5 Susunan Pin LMB162A

(Resipotary USU, 2010)

2.4. Sensor Ultrasonik SRF05

Sensor SRF05 adalah sensor pengukur jarak berbasis gelombang ultrasonik. Prinsip kerja sensor ini mirip dengan radar ultrasonik. Gelombang ultrasonik di pancarkan kemudian di terima balik oleh receiver ultrasonik. Jarak antara waktu pancar dan waktu terima adalah representasi dari jarak objek. Sensor ini cocok untuk aplikasi elektronik yang memerlukan deteksi jarak termasuk untuk sensor pada robot.



Gambar 2.6 Sensor Ultrasonik SRF05

Santoso Hari, 2015

2.5. Motor Servo

Motor servo adalah sebuah motor dengan sistem closed feedback di mana posisi dari motor akan diinformasikan kembali ke rangkaian kontrol yang ada di dalam motor servo. Motor ini terdiri dari motor dc, rangkaian gear , potensio meter dan rangkaian kontrol.



Gambar 2.7 Servo

(Sumber : <http://www.mikropik.com>, 2010)

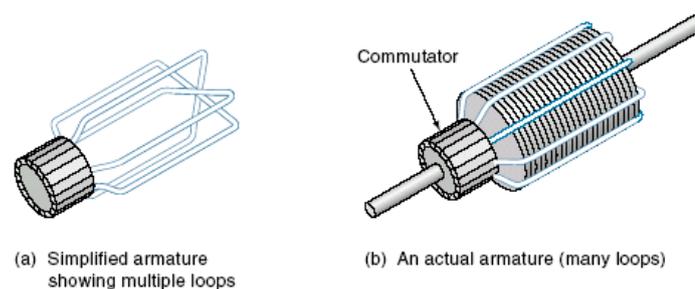
Untuk mengoperasikannya yaitu dengan memberikan Pulsa digital tertentu pada motor ini. secara teori, gambar lebar pulsa di bawah ini merupakan pulsa pengatur sudut servo standard.

Pada dasarnya penggunaan servo itu menggunakan cara yang sama (yaitu dengan memberikan lebar pulsa tertentu). hanya salah satu perbedaannya yaitu pada sudut putarnya. untuk servo standard, sudut putarnya adalah 180 derajat yang dapat dioperasikan dalam dua arah (clock wise / counter clock wise). Gambar diatas adalah lebar pulsa yang dibutuhkan untuk mengoperasikan motor servo standard. pulsa diatas harus diberikan secara terus menerus, agar motor servo mempertahankan posisinya sesuai dengan pulsa yang diberikan.

Sedangkan untuk jenis servo continous putaran yang dapat dilakukan adalah 360 derajat. untuk mengatur arah putarannya yaitu dengan membedakan lebar pulsa saat kondisi ON (logic "1") seperti gambar di bawah ini.

2.6. Motor DC

Salah satu komponen yang tidak dapat dilupakan dalam sistem pengaturan adalah aktuator. Aktuator adalah komponen yang selalu bergerak mengubah energi listrik menjadi pergerakan mekanik. Salah satu aktuator adalah motor listrik. motor DC baik kecepatan, laju dan arah putarnya dapat diatur dengan mudah sesuai dengan keinginan. Motor DC yang kecil bahkan dapat digerakkan dengan tegangan DC yang kecil misalnya motor pada disk drive yang digerakkan dengan tegangan 12 Volt.



Gambar 2.8 Amatur Motor DC

Mulyanti Budi, 2010

Motor DC dapat bekerja sebaliknya yaitu mengubah dari energi mekanik gerak menjadi energi listrik yang disebut dengan generator. Saat ada gaya putar luar memutar comutator, motor DC akan menimbulkan tegangan yang disebut

electromotive force (EMF). Tapi saat motor DC digerakkan oleh daya listrik terdapat juga tegangan balik yang arahnya berlawanan dengan arus armatur yang disebut dengan counter-electromotive force (CEMF). CEMF ini mengurangi tegangan armatur (VA). CEMF ini akan meningkat dengan meningkatnya laju putar motor dan sebaliknya akan berkurang saat laju motor berkurang.

Motor DC memiliki kemampuan untuk mempertahankan lajunya saat dihubungkan dengan beban yang disebut dengan Self-regulation speed. Saat beban meningkat laju berkurang sekaligus menurunkan CEMF. Saat CEMF menurun tegangan armatur akan naik dan menyebabkan laju motor meningkat kembali. (*E-book Universitas Indonesia, 2010*).

2.7. Baterai LiPo 12 V

Baterai Lithium Polimer atau biasa disebut dengan LiPo merupakan salah satu jenis baterai yang sering digunakan dalam dunia Robot. Ada tiga kelebihan utama yang ditawarkan oleh baterai berjenis LiPo ketimbang baterai jenis lain yaitu :

1. Baterai LiPo memiliki bobot yang ringan dan tersedia dalam berbagai macam bentuk dan ukuran
2. Baterai LiPo memiliki kapasitas penyimpanan energi listrik yang besar
3. Baterai LiPo memiliki tingkat discharge rate energi yang tinggi, dimana hal ini sangat berguna sekali dalam bidang RC.

Apabila kapasitas baterai sudah habis, dapat di charge sehingga kapasitas baterai terisi kembali dan dapat digunakan lagi.