

BAB II

TINJAUAN PUSTAKA

2.1 Definisi Robot

Robot klasik sudah ada sejak zaman Yunani kuno. Hingga kini robot terus dikembangkan sehingga keberadaannya sangat membantu manusia dalam mengerjakan pekerjaan rutin dan berat, atau bahkan sebagai penghibur. Elektronika merupakan bidang yang menarik untuk dipelajari oleh pelajar dan penghobi. Hal ini karena kita dapat berkreasi apa saja sesuai keinginan kita. Secara umum robot dapat didefinisikan sebagai sebuah piranti mekanik yang mampu melakukan pekerjaan manusia atau berperilaku seperti manusia. (McComb, 2001).

Ada banyak hal menarik jika anda bermain dengan elektronika, diantaranya adalah membuat robot. Jika anda pernah atau memiliki hobi merakit mobil Tamiya, maka anda sudah memiliki bekal dasar untuk membuat robot karena prinsip yang digunakan pada mobil Tamiya juga banyak digunakan pada robot, yaitu dasar mekanik mesin, roda, dan sumber catu daya. Kata robot sendiri diperkenalkan ke public oleh Karel pada saat memainkan *RUR* (Rossum's Universal Robots). Namun awal munculnya robot dapat diketahui dari bangsa Yunani kuno yang membuat patung yang dapat dipindah-pindahkan (Budiharto, 2009).

Robot adalah sebuah alat mekanik yang dapat melakukan tugas-tugas fisik, baik menggunakan pengawasan dan control manusia, ataupun menggunakan program yang telah dimasukkan terlebih dulu. Dari definisi tersebut ada 4 sifat yang dimiliki robot yaitu (Artanto, 2012) :

1. Robot hanyalah alat mekanik atau mesin.
2. Robot harus deprogram.
3. Robot dapat dibuat responsif sesuai dengan kondisi lingkungan.
4. Robot dapat dibuat otomatis tanpa kendali manusia.

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 (Artanto, 2012:1).

Arduino sebagai sebuah *platform* komputasi fisik (*Physical Computing*) yang *open source* pada *board input output* 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* mikrokontroler 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. (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.

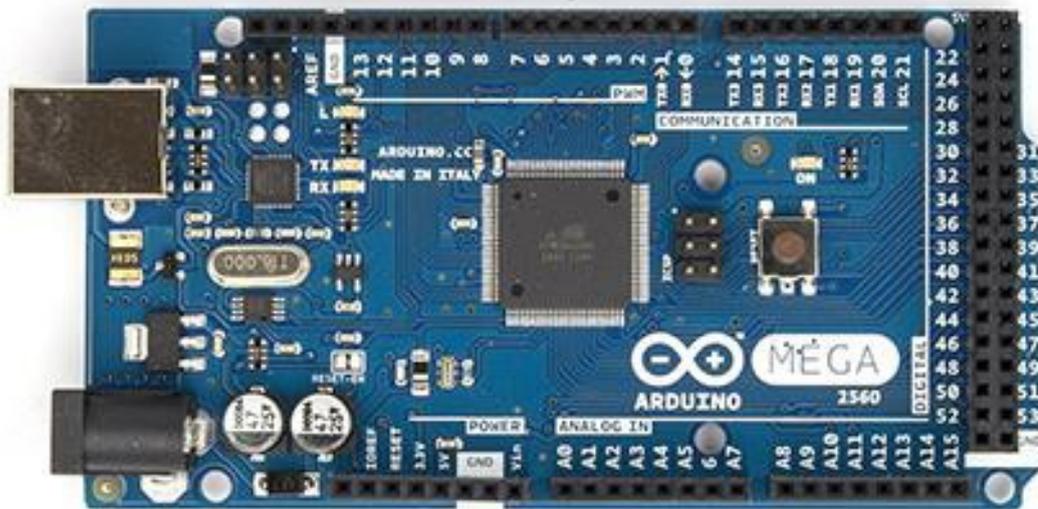
Para desainer hanya tinggal membuat software untuk mendayagunakan rancang hardware yang ada. Software jauh lebih mudah untuk dimodifikasi tanpa harus memindahkan kabel. Saat ini arduino sangat mudah dijumpai dan ada beberapa perusahaan yang mengembangkan sistem hardware *open source* ini. (Djuandi, 2011:4).

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 Programing)*, 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

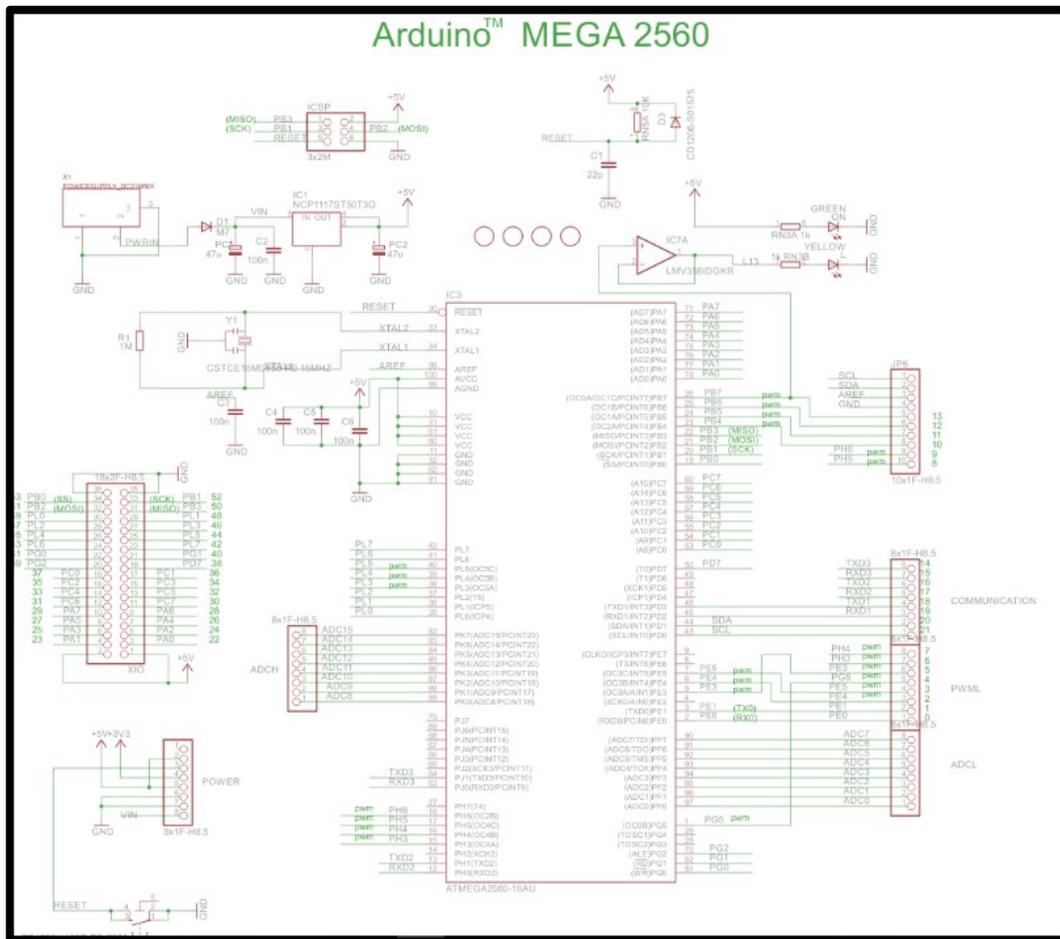
(Sumber : Arduino, 2011)

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.

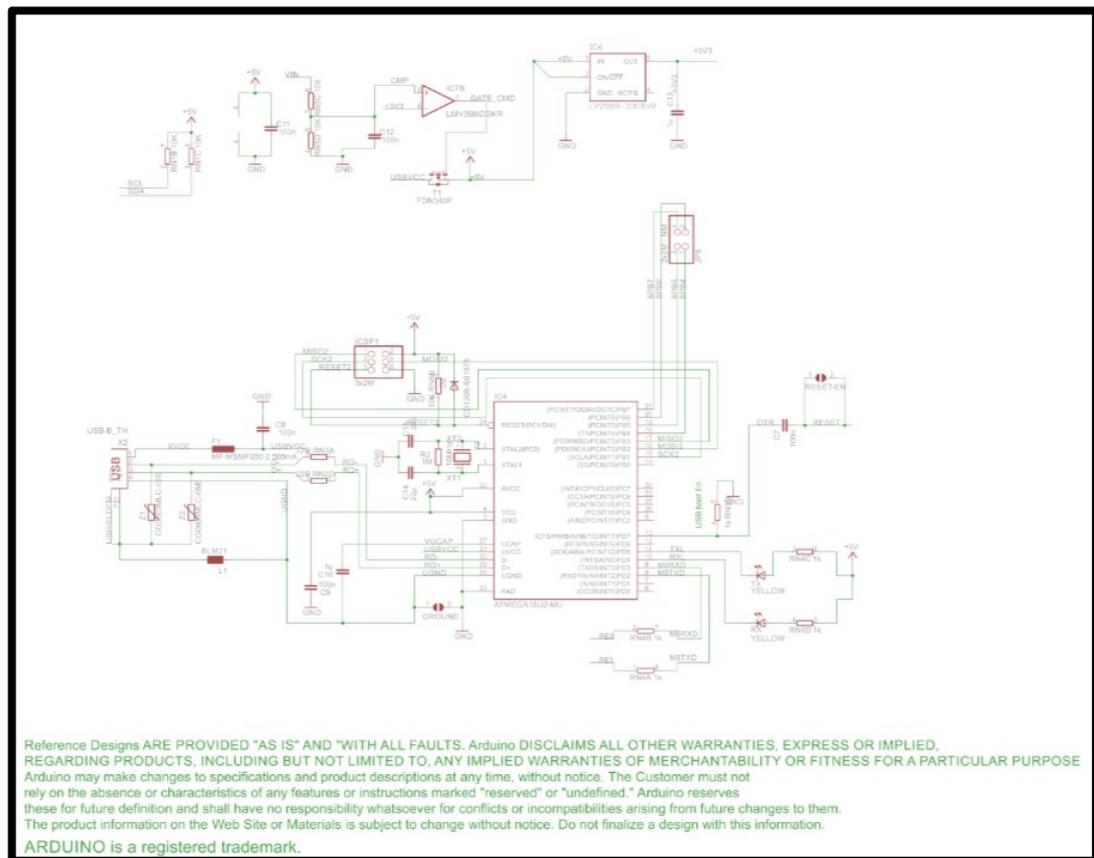
2.2.3 Schematic Arduino Mega 2560

Adapun gambar *schematic* dari rangkaian arduino mega 2560, dapat dilihat pada Gambar 2.2 dibawah ini :



Gambar 2.2.a Schematic Arduino Mega 2560 bagian pertama

(Sumber : Arduino, 2011)



Gambar 2.2.b Schematic Arduino Mega 2560 bagian kedua
(Sumber : Arduino, 2011)

2.2.4 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.

2.2.5 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:

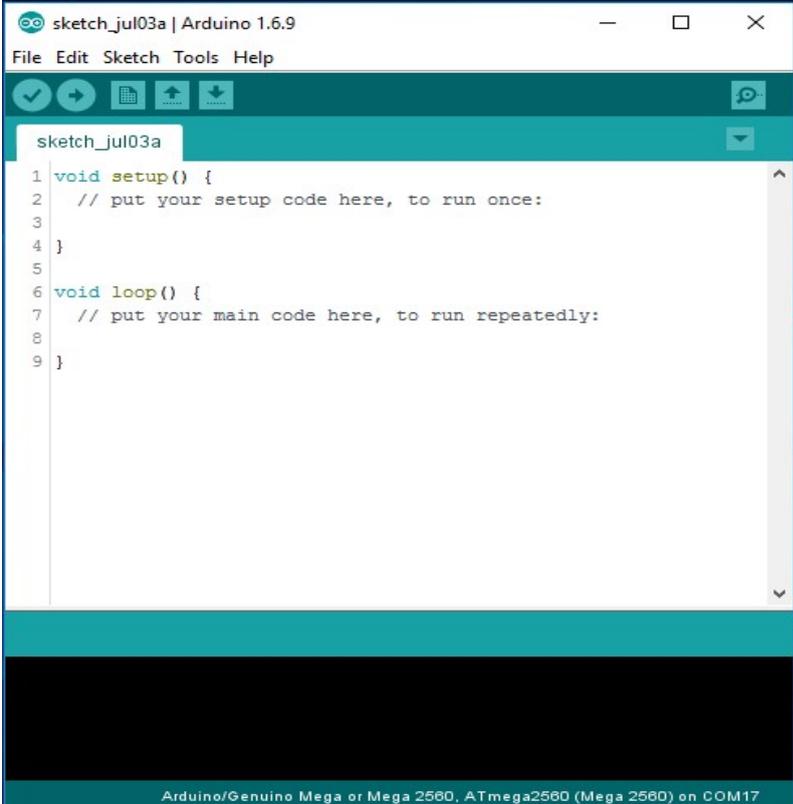
1. Serial: 0 (RX) dan 1 (TX); Serial 1: 19 (RX) dan 18 (TX); Serial 2: 17 (RX) dan 16 (TX); Serial 3: 15 (RX) dan 14 (TX). Digunakan untuk menerima (RX) dan mengirimkan (TX) TTL data serial. Pin 0 dan 1 juga terhubung ke pin yang sesuai dari Atmega 16U2 USB- to-Serial TTL.

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 Emitting 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.

2.2.6 Aplikasi Program IDE (Integrated Development Environment)

Ketika kita membuka program Arduino IDE (*Integrated Development Environment*), akan terlihat serupa dengan tampilan gambar 2.3 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.



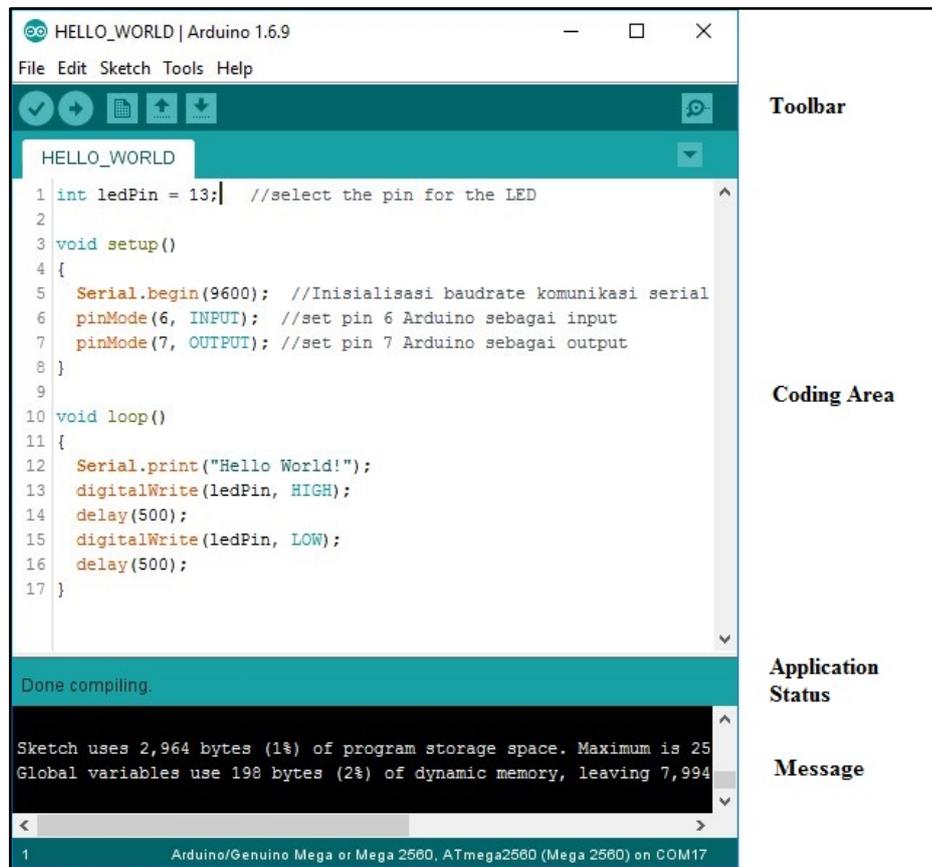
```
sketch_jul03a | Arduino 1.6.9
File Edit Sketch Tools Help
sketch_jul03a
1 void setup() {
2   // put your setup code here, to run once:
3
4 }
5
6 void loop() {
7   // put your main code here, to run repeatedly:
8
9 }
Arduino/Genuino Mega or Mega 2560, ATmega2560 (Mega 2560) on COM17
```

Gambar 2.3 Tampilan Program IDE

(Sumber : Arduino, 2011)

2.2.7 *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 2.4 Tampilan Utama Aplikasi Arduino

(Sumber : Arduino, 2011)

1. *Toolbar*

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



Gambar 2.5 Toolbar Pada Aplikasi Arduino

(Sumber : Arduino, 2011)

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

```

Gambar 2.6 Contoh *Coding* Inisialisasi

(Sumber: Arduino, 20110)

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) ; // memberikan logic HIGH pada pin 6
4   delay (1000) ; // menunda selama 1 detik
5   digitalWrite (6, LOW) ; // memberikan logic LOW pada pin 6
6   delay (2000) ; // menunda selama 2 detik
7 }

```

Gambar 2.7 Contoh *Coding* Fungsi Utama

(Sumber: Arduino, 2011)

3. Application Status

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

4. *Message*

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

2.2.8 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.

2.2.9 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* berukut 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 Joystick Wireless PS2

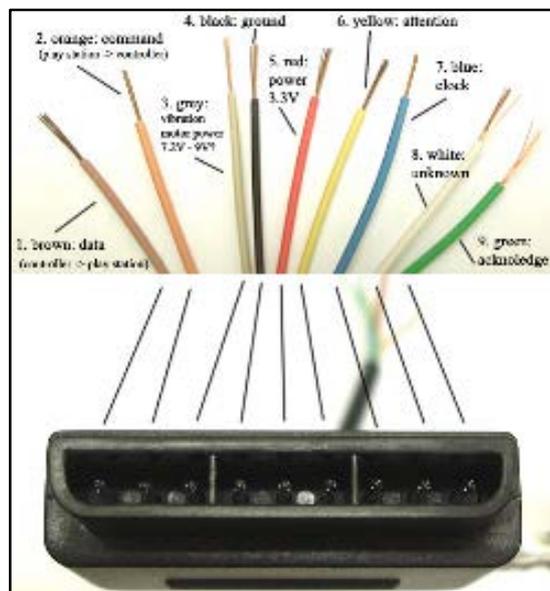
Joystick Wireless PS2 terdiri dari dua modul, yaitu modul *transmitter* dan modul *receiver*. Modul *transmitter* berfungsi sebagai data *input* dan mengirim data *input* tersebut ke modul *receiver*. Sedangkan modul *receiver* berfungsi sebagai penerima data yang dikirim dari modul *transmitter*. Pada setiap Stik PS (*joystick Playstation*) terdapat kontroler yang bertugas untuk berkomunikasi dengan *console playstation*. Komunikasi yang digunakan adalah serial sinkron, yaitu data dikirim satu per satu melalui jalur data. Untuk mengkoordinasikan antara pengirim dan penerima terdapat satu jalur *clock*.



Gambar 2.8 *Joystik PS2 Wireless*
(Sumber : Sony, 2008)

2.3.1 Pin Konfigurasi dan Komunikasi

Konfigurasi pin *wiring Connector Joystik Wireless PS2*, berikut gambarnya:



Gambar 2.9 *Connector PS2 Controller*
(Sumber: Hamdi, 2010)

Joystick Wireless PS2 menggunakan komunikasi SPI (*Serial Peripheral Interface*) atau biasa orang sebut sebagai *3 wire interface*. Adapun penjelasan setiap pin dari *connector PS2 controller* sebagai berikut:

1. Pin 1 (data) : Dihubungkan pada pin 7 pada arduino mega 2560
2. Pin 2 (command) : Dihubungkan pada pin 6 pada arduino mega 2560
3. Pin 3 (motor vibrate) : Vcc
4. Pin 4 (Gnd) : Ground
5. Pin 5 (Vcc) : +5v
6. Pin 6 (attention) : Dihubungkan pada pin 5 pada arduino mega 2560
7. Pin 7 (clock) : Dihubungkan pada pin 4 pada arduino mega 2560
8. Pin 8 (NC) : Tidak dihubungkan
9. Pin 9 (ACK) : Tidak dihubungkan

2.3.2 Data Protokol

Untuk berkomunikasi dengan kontroler pada *joystick wireless PS2* diperlukan beberapa proses pengiriman ID. Mikrokontroler mengirim data &H01 (*start up*), setelah itu mikrokontroler mengirim data &H42 (*read data*). Kemudian disaat yang sama mikrokontroler akan menerima data tipe *joystick* yang digunakan. &H41 = Konsul Digital dan &H73 = Konsul Analog. Setelah itu mikrokontroler akan menerima data &H5. Data *byte* pertama akan diterima kemudian *byte* kedua, setelah itu data analog 1 dan analog 2.

2.3.3 Frame Data

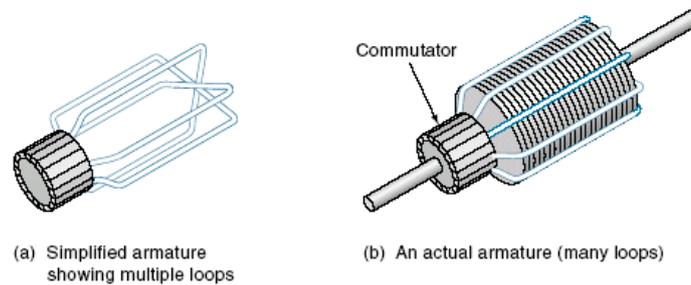
Tabel 2.1 *Frame Data PS2 Controller*

Byte	PsxCmd	PsxData	Keterangan							
1	0x01	-								
2	0x42	0x73								
3	-	0x5A	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
4	-	data digital 1	←	↓	→	↑	Start	Joy-R	Joy-L	Select
5	-	data digital 2	□	X	O	Δ	R1	L1	R2	L2
6	-	data analog 1	Joystick Analog Kanan – Sumbu X 0x00 – kiri, 0xff – kanan							
7	-	data analog 2	Joystick Analog Kanan – Sumbu Y 0x00 – atas, 0xff – bawah							
8	-	data analog 3	Joystick Analog Kiri – Sumbu X 0x00 – kiri, 0xff – kanan							
9	-	data analog 4	Joystick Analog Kiri – Sumbu Y 0x00 – atas, 0xff – bawah							

(Sumber : Hamdi, 2010)

2.4 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

(Sumber: Universitas Indonesia, 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)

2.5 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 2.9 Motor Servo

(Sumber: Universitas Indonesia, 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.

2.7 LCD 16x2

LCD 16x2 merupakan LCD yang memiliki 2 baris dimana setiap barisnya dapat memuat 16 karakter. LCD inilah yang sering digunakan sebagai display data sederhana untuk data yang tidak panjang (tidak banyak jumlahnya), seperti yang ditunjukkan pada Gambar 2.10. LCD ini mudah dihubungkan dengan mikrokontroler.



Gambar 2.10 Tampilan LCD 16x2

(Sumber: Datasheet JHD 162A)

LCD 16x2 memiliki 16 pin konektor yang didefinisikan seperti yang ditunjukkan pada Tabel 2.2 berikut:

Tabel 2.2 Fungsi pin LCD 16x2

PIN	NAMA PIN	FUNGSI
1	VSS	Ground voltage
2	VCC	+5 volt
3	VEE	Contrast voltage
4	RS	Register Select 0 = Instruction Register 1 = Data Register
5	R/W	Enable 0 = Write Mode 1 = Read Mode
6	E	Enable 0 = Start to latch data to LCD character 1 = Disable
	DB0	Data bit ke - 0 (LSB)
8	DB1	Data bit ke - 1
9	DB2	Data bit ke - 2
10	DB3	Data bit ke - 3
11	DB4	Data bit ke - 4
12	DB5	Data bit ke - 5
13	DB6	Data bit ke - 6
14	DB7	Data bit ke - 7
15	BPL	Ground voltage
16	GND	Ground voltage

(Sumber: Datasheet JHD 162A)

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.