

BAB II

KAJIAN PUSTAKA

2.1 Tinjauan Pustaka

Istilah *kinetic art* sudah tak asing lagi di dunia seni rupa. *Kinetic art* mempunyai arti dalam bahasa Inggris yaitu seni gerak. *Kinetic* sendiri secara etimologi berasal dari kata “*kinesis*”. *Kinesis* mempunyai arti dalam bahasa Yunani yaitu gerakan, atau “*kinetikos*” yang artinya bergerak. Para seniman sebenarnya sudah sejak lama tertarik untuk memvisualkan karya *kinetic art*. Akan tetapi *kinetic art* merupakan gaya seni yang muncul sekitar abad kedua puluh, sedangkan di Indonesia baru saja bermunculan dimasa *milenial* ini. Membuat karya *kinetic art* tidak sesederhana itu, karena para seniman yang sudah mencoba membuat karya *kinetic art* justru lebih tertarik dengan gerak itu sendiri dari pada obyek atau karakter yang mereka telah buat.



Gambar 2.1 Karya Cyril Barret

Menurut Cyril Barret pada tahun 1981 dalam Nikos Stangos (*Concepts of Modern Art*). *Kinetic art* dapat memunculkan efek tidak hanya dari karya tersebut, melainkan dari penonton disekitar karya, sentuhan penonton, dan perlakuan oleh penonton. Penggambaran yang membedakan *kinetic art* dengan bentuk seni lainnya tidak hanya dengan membedakan antara gerak dan gerak sebenarnya. Dan tidak semua karya yang memunculkan gerak dapat disebut *kinetic art*. Suatu karya dapat disebut karya *kinetic art* jika karya tersebut mempunyai kualitas khusus.

Edwin Rahardjo pemilik Edwin's Gallery salah satu galeri seni di Jakarta menuturkan bahwa kata kinetik berarti karya yang berhubungan dengan gerak. *Kinetic art* menurut Edwin Rahardjo adalah yang bergantung pada gerak dan mempunyai efek. Penggabungan gerakan kedalam seni telah digabungkan oleh para seniman sejak awal abad kedua puluh. Eksplorasi yang dibuat memunculkan kemungkinan-kemungkinan gerakan, memperkenalkan unsur waktu, mencerminkan pentingnya mesin dan teknologi di dunia *modern*, dan mengeksplorasi sifat visi. Gerakan dimunculkan secara mekanis oleh motor atau dengan mengeksploitasi gerakan secara alami pada udara yang terdapat disuatu ruang. Edwin menunjuk salah seorang seniman *kinetic art*, dia adalah pelopor *kinetic art* yaitu Naum Gabo pada tahun 1919 dan Alexander Calder dari sekitar 1930.



Gambar 2.2 Karya Naum Gabo, Standing Wave, 1919



Gambar 2.3 Karya Alexander Calder, Untitled, 1937

Seniman Asia memiliki pendekatan yang berbeda dalam menggunakan motor mekanis pada karya mereka. *Kinetic art* bukan yang diutamakan pada karya-karya mereka, *kinetic art* adalah elemen pelengkap untuk karya mereka. dan *kinetic art* digunakan untuk mendukung tema pada suatu karya yang mereka buat. Bahkan istilah *kinetic art* jarang digunakan untuk menggambarkan suatu karya yang memiliki gerakan sebagai bagian dari karya yang membuat karya itu berdiri sendiri. Seni instalasi lebih populer sebagai istilah dari pada *kinetic art*. Tidak ada penelitian khusus yang dilakukan atau diskusi tentang *kinetic art* di Asia. Semua seniman Asia yang tangguh memiliki berbagai macam pendekatan dan tema unik dalam konsep pada karya mereka yang menggunakan gerak dan gerakan. Seni kinetik di Asia memiliki potensi yang sangat besar untuk terus berkembang menjadi bentuk seni baru. Salah satu seniman Asia yang telah bekerja untuk mengembangkan *kinetic art* melalui eksperimen dan eksplorasi yang intensif.



Gambar 2.4 Karya Heri Dono, Angels Falls from The Sky, 1996

Seniman Indonesia Heri Dono, lahir pada 12 Juni 1960 di Jakarta. Karakter seni Heri Dono adalah manusia dalam suatu bentuk tetapi mengandung komponen-komponen mekanis. Heri Dono menggunakan boneka wayang Indonesia, suatu bentuk seni yang secara tradisional menggabungkan unsur-unsur sosial. Sumber lain untuk tokoh-tokohnya termasuk gaya Barat dan robot, yang mempertanyakan status dan tindakan manusia.

Dalam karya-karyanya, Heri Dono sering menggunakan matriks atau struktur seperti grid. Struktur ini memiliki signifikansi estetika dalam seni *modern* Eropa serta memiliki makna spiritual dalam budaya Indonesia. Heri Dono juga mengambil inspirasi dari budaya populer Barat, dengan *Flying Angels* 1995 berdasarkan kartun *Flash Gordon* animasi yang ia cintai sejak kecil.

2.2 Dasar Teori

2.2.1 Motor Servo

Motor servo adalah sebuah motor dengan sistem *closed feedback*, posisi motor servo akan menginformasikan kembali ke bagian kontrol yang berada dalam motor servo. Motor servo ini terdiri dari beberapa rangkaian yaitu motor, gear, potensiometer, dan kontrol. Motor servo mampu bekerja dengan dua arah yaitu CW dan CCW yang mana arah dan sudut pergerakan tersebut pada rotor hanya dapat dikendalikan dengan memberi pengaturan PWM pada bagian pin kontrol. Gambar motor servo dapat dilihat pada gambar 2.5.



Gambar 2.5 Motor Servo Metal Gear

Dalam motor servo memiliki sebuah motor DC yang mempunyai rangkaian kontrol elektronik dan internal gear untuk mengendalikan gerak dan sudut angular. Sistem mekanik motor servo dapat dilihat pada gambar 2.6.



Gambar 2.6 Mekanik Motor Servo

2.2.1.1 Aplikasi Motor Servo

Motor servo merupakan motor yang berputar dengan lambat, akan tetapi motor servo mempunyai *torsi* yang kuat karena pada motor servo terdapat *internal gear*.

Sebuah motor servo memiliki rangkaian berupa :

- a. Tiga jalur kabel power, ground, dan kontrol.
- b. Sinyal kontrol untuk mengendalikan posisi.
- c. Motor servo mempunyai operasional yang dikendalikan oleh sebuah pulsa 20ms.
- d. Motor servo mempunyai konstruksi berupa *gear*, potensiometer, dan *feedback* kontrol.

2.2.1.2 Jenis-Jenis Motor Servo

a. Motor Servo 180°

Motor servo jenis ini hanya mampu bergerak dua arah CW dan CCW dengan masing *defleksi* sudut mencapai 90° sehingga total *defleksi* sudut dari kanan, tengah, kiri adalah 180°.

b. Motor Servo 360°

Motor servo ini mampu bergerak dua arah CW dan CCW tanpa mempunyai batasan *defleksi* sudut putar tertentu. Motor servo tersebut dapat berputar secara kontinyu.

2.2.1.3 Kegunaan Motor Servo

Kebanyakan motor servo digunakan sebagai :

- a. *Manipulators*.
- b. *Moving camera's*.
- c. Robot arms.

2.2.2 Mikrokontroler

Mikrokontroler merupakan sebuah sistem yang menyerupai sistem komputer dalam satu *chip*. Sebuah mikrokontroler mempunyai beberapa macam fitur di dalamnya seperti:

1. ROM (*Read-Only Memory*)
2. RAM (*Read-Write Memory*)
3. Beberapa Port I/O (*Port Input dan Output*)
4. Beberapa *peripheral* seperti ADC (*Analog to Digital Converter*) dan DAC (*Digital to Analog Converter*)
5. *Timer* dan *counter*
6. Komunikasi *serial* dengan menggunakan *port transmitter* dan *receiver*

Dengan kata lain mikrokontroler bukan hanya sebuah mikro processor saja karena bisa melakukan yang lebih dari itu.

2.2.2.1 Arduino

Arduino adalah salah satu dari sekian pengendali mikrokontroler yang bersifat *open-source* yang dirancang khusus agar pengguna yang sedang melakukan perancangan elektronik lebih mudah dalam pembuatannya di berbagai bidang. Arduino memiliki *hardware* dengan prosesor Atmel AVR dan juga untuk programnya Arduino memiliki *software* yang bahasa programnya pun memiliki bahasa pemrograman sendiri. Arduino sendiri banyak diminati di seluruh dunia.

Kebanyakan orang-orang yang masih pemula yang mulai belajar elektronika, robotika dan pemrograman mikrokontroler memulai dengan mengenal Arduino karena Arduino cukup mudah untuk dipelajari. Selain orang baru belajar robotika, seseorang yang senang robotika dan para profesional pun banyak yang membuat sesuatu *prototype* dan mengembangkan *prototype* tersebut dengan menggunakan Arduino.

Arduino tidak memakai bahasa *assembler* yang rumit, bahasa yang digunakan adalah bahasa C yang telah disederhanakan sehingga

lebih mudah dengan menggunakan bantuan pustaka (*Libaries*) khusus Arduino. Selain itu, Arduino juga dapat bekerja dengan mikrokontroler dan menyederhanakan proses.

Arduino memiliki beberapa kelebihan, adapun kelebihan yang ditawarkan antara lain:

- a. Harga perangkat keras Arduino yang dijual di pasaran biasanya relatif murah (berada di kisaran 125 ribuan hingga 400 ribuan) dibandingkan dengan perangkat keras mikrokontroler pro yang lainnya. Selain itu, kita juga bisa membuat perangkat keras Arduino sendiri karena semua sumber daya yang dibutuhkan tersedia di *website* resmi Arduino dan komunitas-komunitas lainnya. Arduino juga cocok bukan hanya untuk Windows. OS Linux juga cocok untuk Arduino.
- b. Ruang lingkup pemrograman Arduino sangat sederhana dan sangat mudah untuk melakukan pemrogramannya. Bagi pemula, Arduino merupakan alternatif pembelajaran yang bisa dipahami dengan cepat. Selain itu, Arduino juga cukup fleksibel untuk digunakan sebagai pengembangan perangkat elektronik. Arduino berbasis pada lingkungan pemrograman *processing*. Dengan begitu, seseorang yang sudah terbiasa menggunakan *processing* tentu saja akan jauh lebih mudah ketika menggunakan Arduino.
- c. Perangkat lunak yang digunakan Arduino bernama Arduino IDE. perangkat lunak tersebut tersedia bagi para pemrogram baik yang pemula maupun yang sudah berpengalaman. Pemakaiannya sendiri tidak sulit.
- d. Perangkat keras yang terdapat pada Arduino menggunakan berbagai macam mikrokontroler. Pada umumnya tipe yang

dipakai adalah tipe ATmega, seperti ATmega 1280, ATmega 168, ATmega 328, dan ATmega 8 (yang terbaru adalah ATmega 2560).

- e. Arduino tidak memerlukan *chip programmer* karena di dalam perangkatnya sudah terdapat *bootloader* yang menangani *upload* programnya dari PC. Arduino juga memiliki sarana komunikasi USB.
- f. *Input/output digital* atau *digital* pin adalah pin yang berfungsi menghubungkan perangkat Arduino dengan beberapa komponen atau rangkaian *digital* sebagai *input* atau *output* Arduino.
- g. Pin catu daya adalah pin yang berfungsi untuk memberikan tegangan yang dibutuhkan untuk komponen atau rangkaian yang nantinya dihubungkan dengan Arduino. Pada bagian catu daya ini terdapat pin *Vin* dan pin *Reset*.

2.2.2.2 Arduino Uno

Arduino UNO merupakan sebuah *board* mikrokontroler yang menggunakan ATmega328 sebagai *datasheet*. Arduino ini memiliki 14 pin *digital input* dan *output*, 6 di antaranya digunakan sebagai *output* pin PWM) dan 6 lainnya adalah *input analog*. Arduino UNO juga memiliki osilator kristal 16 MHz, koneksi USB, *power jack*, ICSP *header*, dan tombol *reset*. Arduino UNO mudah menghubungkannya ke dalam sebuah komputer dengan sebuah kabel USB atau mensuplainya dengan sebuah adaptor dan dapat juga menggunakan baterai untuk mensuplai *power*.

Arduino Uno ini sangat berbeda dari semua *board* Arduino sebelumnya. Arduino UNO tidak menggunakan *chip driver* FTDI USB to *serial*. Dari *board* Arduino UNO mempunyai fitur sebagai berikut:

- a. Pin Out 1.0 yang ditambahkan dengan pin SDA dan SCL dan dekat dengan pin AREF. Pin tersebut diletakkan dekat dengan pin RESET dan IOREF yang memungkinkan *shield-shield* untuk menyesuaikan tegangan.
- b. Sirkuit RESET yang lebih kuat.
- c. Atmega 16U2 menggantikan 8U2.

Tabel 2.1 Spesifikasi Arduino UNO

Mokrokontroler	ATmega328P
Operasi Tegangan	5V
<i>Input</i> Tegangan (yang direkomendasikan)	7-12V
<i>Input</i> Tegangan (yang dibatasi)	6-20V
<i>Digital</i> I/O Pin	14 (6 diantaranya menyediakan PWM)
PWM <i>Digital</i> I/O Pin	6
<i>Analog Input</i> Pin	6
Arus DC per I/O Pin	20mA
Arus DC untuk 3.3V Pin	50mA
<i>Flash Memory</i>	32 KB (ATmega328P)
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
<i>Clock Speed</i>	16 MHz
Panjang	68,6mm
Lebar	53,4mm
Berat	25g

Catatan: Desain Arduino UNO ini dapat menggunakan Atmega8, Atmega168, dan Atmega328. Untuk model saat ini menggunakan Atmega328, tetapi Atmega8 ditampilkan pada skema arduino sebagai referensi. Konfigurasi pin tersebut identik dengan semua prosesor.

2.2.2.2.1 Daya (*Power*) Arduino UNO

Daya pada Arduino UNO dapat disuplai melalui koneksi USB dan dengan *power supply* eksternal. *Power supply* eksternal (non-USB) artinya dapat diperoleh dari adaptor AC ke DC atau battery. Adaptor dapat dihubungkan dengan memasukkan colokan dalam *header* pin ground (Gnd) pada pin Vin dari konektor *power*.

Board Arduino UNO dapat beroperasi pada *power supply* eksternal mulai dari 6V sampai 20V. Jika *power supply* diberikan lebih kecil dari 7V, maka pin 5V hanya mensuplai lebih kecil dari 5V dan *board* Arduino UNO menjadi tidak stabil. Jika menggunakan *power supply* lebih dari besar 12V, maka *voltage regulator* dapat menjadi kelebihan panas dan membahayakan *board* Arduino UNO. Pin-pin daya yang terdapat pada *board* Arduino UNO adalah sebagai berikut:

- a. Vin yaitu tegangan *input* ke Arduino *board* ketika *board* sedang menggunakan sumber *power supply* eksternal misalnya tegangan 5V dari koneksi USB.
- b. 5V adalah pin *output* yang merupakan tegangan bernilai 5V yang diatur dari regulator pada *board* Arduino UNO. *Board* pada Arduino UNO dapat disuplai dengan salah satu *power supply* dari DC *power jack* (7-12V), USB *connector* (5V), atau pin Vin dari *board* (7-12V). Sedangkan memberi *supply* tegangan melalui pin 5V atau 3,3V sama saja dengan *membypass* regulator, dan dapat membahayakan *board* Arduino UNO.
- c. 3,3V. Sebuah *supply* 3,3V dihasilkan oleh regulator pada *board*. Arus maksimum yang dapat dilalui adalah 50mA.
- d. GND adalah Pin untuk *ground*.

- e. IOREF. Pin ini terdapat pada *board* arduino UNO untuk memberikan tegangan pada saat mikrokontroler beroperasi. Sebuah perisai dikonfigurasi yang benar dapat membaca pin tegangan IOREF dan memilih sumber daya yang tepat untuk mengaktifkan penerjemah tegangan pada output untuk bekerja dengan pin 5V atau pin 3,3V. Pada ATmega328 mempunyai kapasitas 32 KB dan 0,5 KB digunakan untuk *bootloader*. ATmega 328 mempunyai 2 KB SRAM dan 1 KB EEPROM yang dapat dibaca dan ditulis pada *RW/read* dan *written* dengan EEPROM *library*.

2.2.2.2.2 Input dan Output Arduino UNO

Pada setiap 14 pin *digital* arduino UNO dapat digunakan sebagai *input* dan *output* menggunakan fungsi pin *Mode()*, *digital Write()*, dan *digital Read()*. Fungsi-fungsi tersebut dapat beroperasi pada tegangan 5V. Selain itu, beberapa pin mempunyai fungsi-fungsi spesial yaitu:

- a. Pin *Serial* yaitu pin yang terdapat pada 0 (RX) dan 1 (TX). Pin ini dapat digunakan untuk menerima (RX) dan memancarkan (TX) *serial* data TTL (Transistor-Transistor Logic). Kedua pin tersebut dihubungkan ke pin-pin yang sesuai dari *chip serial* Atmega8U2 USB ke TTL.
- b. Pin *External Interrupts* 2 dan 3. Pin ini dikonfigurasi untuk dipicu sebuah gangguan (*interrupt*) pada nilai rendah. Kenaikan atau penurunan yang besar atau suatu perubahan nilainya dapat dilihat fungsi *attachInterrupt()* untuk lebih jelasnya.

- c. Pin PWM. Pin ini terdapat pada pin 3, pin 5, pin 6, pin 9, pin 10, dan pin 11. Cara memberikan 8-bit PWM pada *output* yaitu dengan fungsi *analogWrite()*.
- d. Pin SPI. Pin-pin ini adalah pin yang *mensupport* komunikasi SPI menggunakan *SPI library*.
- e. LED. Ada sebuah LED yang terpasang pada Arduino UNO yang terhubung ke pin *digital* 13. Ketika pin bernilai *high* maka LED akan menyala, dan ketika pin bernilai *low* maka LED akan mati.

Board Arduino UNO mempunyai 6 *input analog* dan diberi label A0 sampai A5, setiap pin tersebut memberikan 10bit resolusi. Secara *default*, 6 *input analog* tersebut dapat mengukur dari *ground* sampai tegangan 5V. Dengan pin tersebut mengganti batas atas dari *range* dengan menggunakan pin AREF dan fungsi *analog reference*. 6 pin mempunyai beberapa fungsi spesial yaitu:

- a. TWI. Pin yang *mensupport* komunikasi TWI dengan menggunakan *wire library* adalah Pin A4 atau SDA dan pin A5 atau SCL.
- b. AREF. Adalah referensi tegangan untuk *input analog* yang digunakan dengan *analogReference()*.
- c. *Reset*. Digunakan untuk menyalurkan *low* untuk *mereset* mikrokontroler.

2.2.2.2.3 Komunikasi Arduino UNO

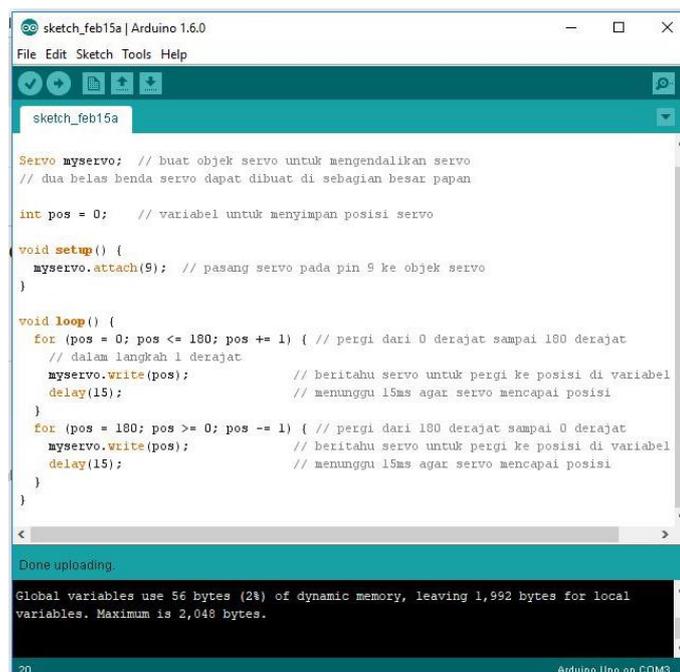
Arduino UNO menyediakan sebuah *software serial library* yang dapat memungkinkan untuk komunikasi *serial* pada beberapa pin *digital*.

Pada Atmega328 juga dapat mensupport komunikasi I2C (TWI) dan SPI. *Software* pada Arduino UNO mencakup sebuah *wire library* untuk memudahkan menggunakan bus I2C, dan komunikasi SPI menggunakan *SPI library*.

2.2.2.2.4 Programming Arduino UNO

Arduino UNO dapat diprogram dengan menggunakan *software* Arduino. Pilih “Arduino Uno dari menu *Tools > Board* (termasuk mikrokontroler pada *board*). Untuk lebih jelas, lihat referensi dan tutorial.

ATmega328 pada Arduino UNO hadir dengan sebuah *bootloader* yang memungkinkan untuk mengupload kode baru ke ATmega328 tanpa menggunakan pemrogram *hardware* eksternal. ATmega328 berkomunikasi menggunakan protokol STK500 asli (referensi, *file C header*).



```

sketch_feb15a | Arduino 1.6.0
File Edit Sketch Tools Help

sketch_feb15a

Servo myservo; // buat objek servo untuk mengendalikan servo
// dua belas benda servo dapat dibuat di sebagian besar papan

int pos = 0; // variabel untuk menyimpan posisi servo

void setup() {
  myservo.attach(9); // pasang servo pada pin 9 ke objek servo
}

void loop() {
  for (pos = 0; pos <= 180; pos += 1) { // pergi dari 0 derajat sampai 180 derajat
    // dalam langkah 1 derajat
    myservo.write(pos); // beritahu servo untuk pergi ke posisi di variabel
    delay(15); // menunggu 15ms agar servo mencapai posisi
  }
  for (pos = 180; pos >= 0; pos -= 1) { // pergi dari 180 derajat sampai 0 derajat
    myservo.write(pos); // beritahu servo untuk pergi ke posisi di variabel
    delay(15); // menunggu 15ms agar servo mencapai posisi
  }
}

Done uploading.

Global variables use 56 bytes (2%) of dynamic memory, leaving 1,992 bytes for local
variables. Maximum is 2,048 bytes.

20 Arduino Uno on COM3

```

Gambar 2.7 Program Arduino IDE

Membypass *bootloader* dan program mikrokontroler melalui kepala (*header*) ICSP (*In-Circuit Serial Programming*), lihat instruksi untuk lebih jelas sumber kode *firmware* ATmega16U2 (atau 8U2 pada *board* revisi 1 dan revisi 2) tersedia. ATmega16U2/8U2 di*load* dengan sebuah *bootloader* DFU, yang dapat diaktifkan dengan:

- a. Pada *board* Revisi 1. Dengan menghubungkan *jumper* solder pada belakang *board* (dekat peta Italy) dan kemudian *mereset* 8U2.
- b. Pada *board* Revisi 2 atau setelahnya. Ada sebuah resistor yang menarik garis HWB 8U2/16U2 ke *ground*, dengan itu dapat lebih mudah untuk meletakkan ke dalam *mode* DFU. Menggunakan atau pemrogram DFU (Mac OS X dan Linux) untuk *meload* sebuah *firmware* baru. Atau dapat menggunakan *header* ISP dengan sebuah pemrogram eksternal (mengoverwrite *bootloader* DFU). Lihat tutorial *user-contributed* ini untuk informasi selengkapnya.

2.2.2.2.5 Reset Otomatis Software Arduino UNO

Dari pada mengharuskan sebuah penekanan fisik dari tombol *reset* sebelum sebuah *penguploadan*. Arduino Uno didesain pada sebuah cara yang memungkinkannya untuk *direset* dengan *software* yang sedang berjalan pada pada komputer yang sedang terhubung. Salah satu garis kontrol aliran *hardware* (DTR) dari ATmega8U2/16U2 sihubungkan ke garis *reset* dari ATmega328 melalui sebuah kapasitor 100 nanofarad.

Ketika saluran ini dipaksakan (diambil rendah), garis *reset* jatuh cukup panjang untuk *mereset chip*. *Software* Arduino menggunakan kemampuan ini untuk memungkinkan untuk *mengupload* kode dengan mudah menekan tombol *upload* di *software*

arduino. Ini berarti bahwa *bootloader* dapat mempunyai sebuah batas waktu yang lebih singkat, sebagai penurunan dari DTR yang dapat menjadi koordinasi yang baik dengan memulai penguploadan.

Pengaturan ini mempunyai implikasi. Ketika Arduino UNO dihubungkan ke sebuah komputer lain yang sedang *running* menggunakan OS Mac X atau Linux, arduino UNO *mereset* setiap kali sebuah koneksi dibuat dari *software* (melalui USB). Untuk berikutnya, setengah detik atau lebih, *bootloader* sedang berjalan pada Arduino UNO. Ketika Arduino UNO diprogram untuk mengabaikan data yang cacat/salah (contohnya apa saja selain sebuah penguploadan kode baru) untuk menahan beberapa bit pertama dari data yang dikirim ke *board* setelah sebuah koneksi dibuka. Jika sebuah *sketch* sedang berjalan pada *board* menerima satu kali konfigurasi atau data lain ketika *sketch* pertama mulai, memastikan bahwa *software* yang berkomunikasi menunggu satu detik setelah membuka koneksi dan sebelum mengirim data ini.

Arduino Uno berisikan sebuah jejak yang dapat dihapus untuk mencegah *reset* otomatis. *Pad* pada salah satu sisi dari jejak dapat disolder bersama untuk mengaktifkan kembali. *Pad* itu -RN dan *reset* dapat menotomatiskan dengan menghubungkan sebuah resistor 110 ohm dari tegangan 5V ke garis *reset*, lihat *thread forum* ini untuk lebih jelasnya.

2.2.2.2.6 Proteksi Arus USB Arduino UNO

Arduino UNO mempunyai sebuah *sekring reset* yang memproteksi *port* USB komputer dari hubungan pendek dan arus lebih. Walaupun sebagian besar komputer menyediakan proteksi internal sendiri, *sekring* menyediakan sebuah proteksi tambahan. Jika lebih dari 500 mA diterima *port* USB, *sekring* secara otomatis akan

memutuskan koneksi sampai hubungan pendek atau kelebihan beban hilang.

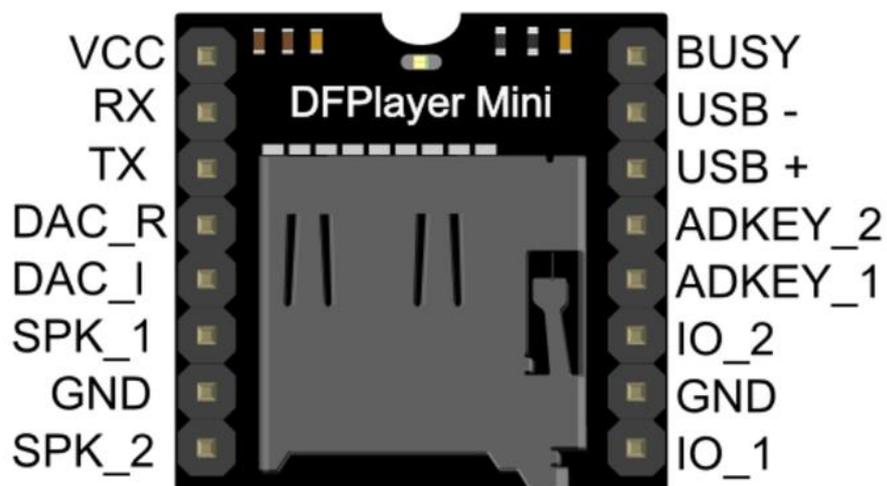
2.2.2.2.7 Karakteristik Fisik Arduino UNO

Panjang dan lebar maksimum Arduino UNO adalah 2,7 dan 2,1 inci, dengan konektor USB dan *power jack* yang memperluas dimensinya. Empat lubang sekrup memungkinkan *board* untuk dipasangkan ke sebuah permukaan atau kotak. Sebagai catatan, bahwa jarak antara pin *digital* 7 dan 8 adalah 160 mil. (0.16"), bukan sebuah kelipatan genap dari jarak 100 mil dari pin lainnya.

2.2.3 Mp3 Player Module

DFPlayer *mini* adalah modul *sound player* yang dapat mendukung beberapa *file* salah satunya adalah *file* mp3 yang umumnya digunakan sebagai format *sound file*.

DFPlayer *mini* ini mempunyai 16 pin *interface* yaitu berupa pin standar DIP dan pin *header* pada kedua sisinya. Berikut adalah gambar DFPlayer *mini* pada gambar 2.8.



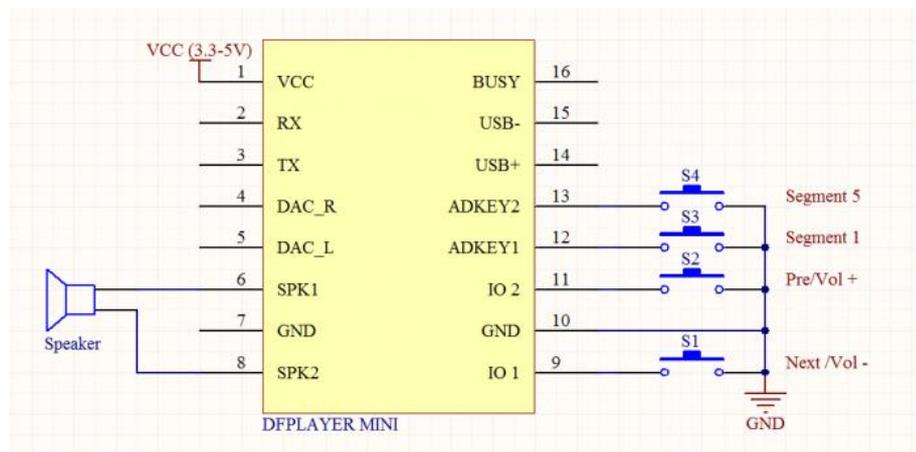
Gambar 2.8 Modul Mp3

Tabel 2.2 Spesifikasi Modul Mp3

Nama	Deskripsi	Catatan
VCC	<i>Input</i> Tegangan	DC 3,2-5.0V
RX	UART <i>input serial</i>	
TX	UART <i>output serial</i>	
DAC_R	<i>Output audio</i> saluran kanan	<i>Earphone drive</i> dan amplifier
DAC_L	<i>Output audio</i> saluran kiri	
SPK2	Speaker	Speaker <i>power</i> (<3W)
GND	<i>Ground</i>	<i>Power ground</i>
SPK1	Speaker	Speaker <i>power</i> (<3w)
I0 1	<i>Trigger port 1</i>	Tekan sebentar untuk memainkan lagu berikutnya (tahan lama untuk mengurangi volume)
GND	<i>Ground</i>	<i>Power ground</i>
I02	<i>Trigger port 2</i>	Tekan lama untuk memainkan lagu berikutnya (tahan lama untuk meningkatkan volume)
ADKEY1	<i>AD port 1</i>	Memicu memainkan segmen pertama
ADKEY2	<i>AD port 2</i>	Memicu memainkan segmen kelima
UBS +	USB + DP	<i>Port</i> USB
USB -	USB - DM	<i>Port</i> USB
<i>Busy</i>	Memainkan status	Rendah Memainkan musik Tinggi tidak memainkan musik

DFPlayer *mini* tersebut dapat bekerja sendiri secara *stand-alone* ataupun bekerja bersama dengan mikrokontroler melalui koneksi *serial*.

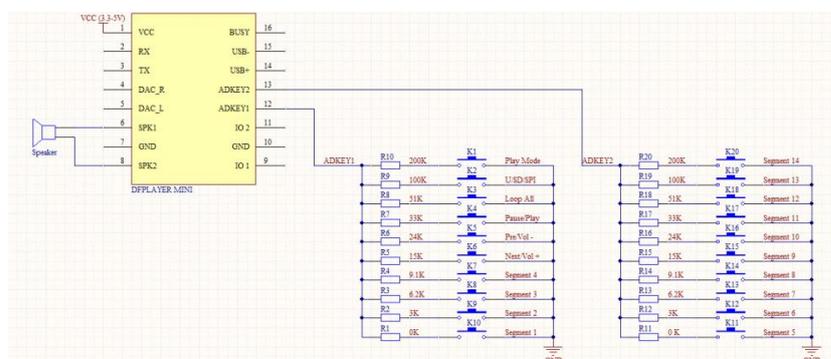
2.2.3.1 I/O Mode DFPlayer Mini



Gambar 2.9 Rangkaian I/O Mode

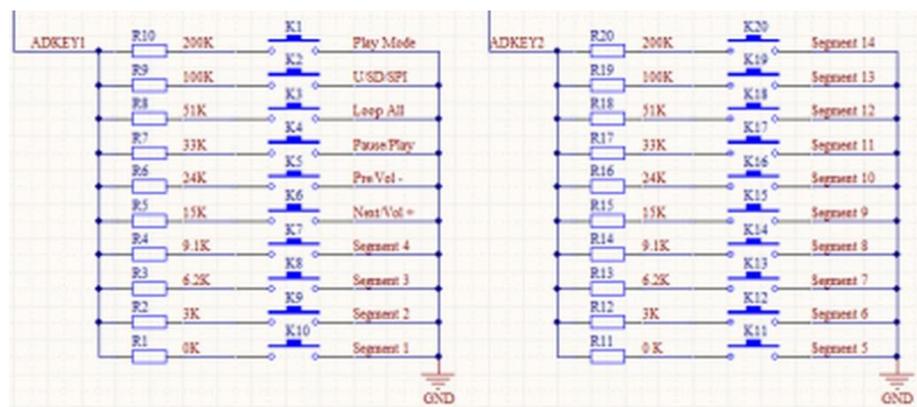
Berikut adalah *wiring* yang sangat sederhana dari penggunaan modul DFPlayer *mini*, bahkan dapat mengabaikan *push button* S3 dan S4 yang terhubung di pin ADKey. Hal ini hanya memerlukan 2 buah *push button* dan 1 *mini speaker* yaitu menekan S1 dan S2 dengan cepat untuk *next* atau *previous* dan tekan S1 dan S2 secara ditahan untuk atur *volume*.

2.2.3.2 AD Key (Analog ke Digital) Mode DFPlayer Mini



Gambar 2.10 Rangkaian AD Key

Modul DFPlayer *mini* mempunyai 2 pin ADC (*analog ke digital converter*) pada pin 12 dan pin 13 yang dapat digunakan sebagai metode *input* untuk memberikan *trigger* kepada internal MCU DFPlayer *mini* tersebut. Dan untuk mengartikan beberapa perintah tombol.



Gambar 2.11 Rangkaian ADKEY 1 dan ADEKEY 2

Dengan cara membuat *button array* seperti yang ada pada LCD *button* modul, lalu dapat membuat 20 *push button* dengan 20 fungsi berbeda.

Modul DFPlayer *mini* tersebut sudah memiliki *builtin* amplifier (*mini*) dan sudah dapat menjalankan *mini* speaker sebagai *output* suaranya. Namun daya *power* amplifier yang dihasilkan masih kecil sehingga modul ini cepat panas saat digunakan untuk menjalankan speaker 4 Ohm sampai 8 Ohm.

Jika dengan menggunakan eksternal amplifier maka dapat diambil pin DAC_R dan DAC_L serta *common* GND. Eksternal amplifier yang dapat digunakan adalah seri PAM ataupun TDA.

2.2.3.3 DFPlayer *Mini* Arduino UNO

DFPlayer *mini* memiliki koneksi serial yang dapat menerima instruksi dari kontroler lain seperti halnya Arduino UNO. Berikut adalah daftar perintah yang dapat digunakan untuk kontrol modul DFPlayer *mini* melalui koneksi *serial* Arduino UNO:

a. *Mode Serial*

Dukungan untuk *mode serial asynchronous* melalui perintah pengiriman *serial* PC deskripsi instruksi.

Tabel 2.3 *Serial Mode*

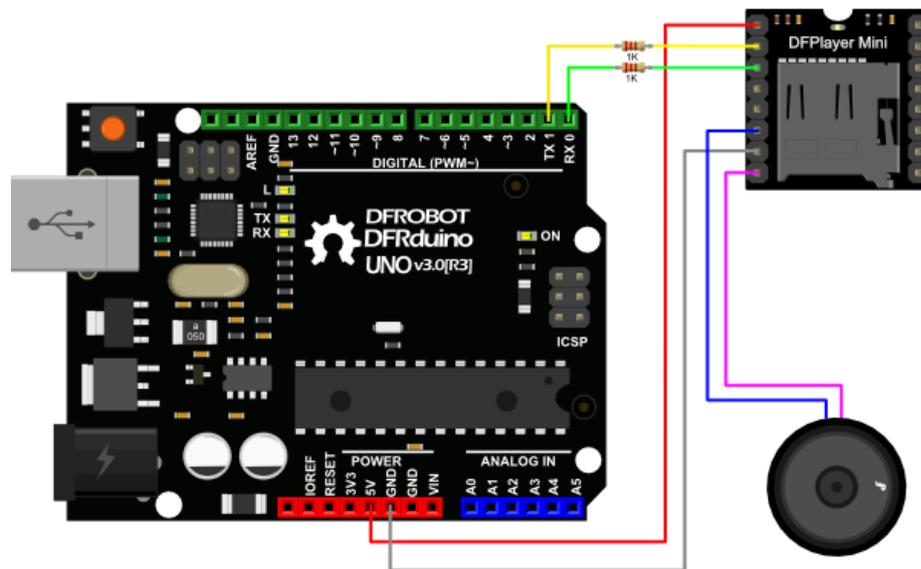
Perintah	Deskripsi Fungsi	Parameter (16bit)
0 x 01	Berikutnya	
0 x 02	Sebelumnya	
0 x 03	Pelacakan khusus (NUM)	0-2999
0 x 04	Meningkatkan suara	
0 x 05	Mengurangi suara	
0 x 06	Menentukan suara	0-30
0 x 07	Menentukan equalizer 0/1/2/3/4/5	Normal/Pop/Rock/ Jazz/Classic/Bass
0 x 08	Menentukan <i>mode</i> pemutaran (0/1/2/3)	<i>Repeat</i> /folder, <i>repeat/single</i> , dan <i>repeat/random</i>
0 x 09	Menentukan sumber <i>mode</i> (1/2/3/4)	U/TF/AUX/SLEEP/F LASH
0 x 0A	Masuk bertahan ke daya rendah	
0 x 0B	Bekerja normal	
0 x 0C	<i>Reset module</i>	
0 x 0D	Pemutaran	
0 x 0E	Jeda	
0 x 0F	Menentukan folder untuk diputar	1-10 (perlu diatur oleh pengguna)
0 x 10	Setting pengaturan <i>volume</i>	[DH=1; buka suara menyesuaikan] [DL; mengatur suara 0-31]
0 x 11	Ulangi bermain	[1;ulangi] [0;berhenti bermain]

Lanjutan Tabel 2.3 *Serial Mode*

Perintah	Deskripsi Fungsi	Parameter (16bit)
0 x 12	Menentukan folder lagu MP3	0-9999
0 x 13	Iklan	0-9999
0 x 14	Mendukung 15 folder	Lihat uraian detail dibawah ini
0 x 15	Berhenti memutar, putar latang belakang	
0 x 16	Berhenti memutar	

Tabel 2.4 Permintaan *Serial Cmd*

Perintah	Fungsi Deskripsi
0x3C	Tinggal
0x3D	Tinggal
0x3E	Tinggal
0x3F	Kirim parameter inisialisasi
0x40	Mengembalikan kesalahan, meminta pengiriman ulang
0x41	Ulangan
0x42	Pertanyaan status saat ini
0x43	Pertanyaan <i>volume</i> saat ini
0x44	Pertanyaan status equalizer saat ini
0x45	Pertanyaan <i>mode</i> pemutaran saat ini
0x46	Pertanyaan versi perangkat lunak saat ini
0x47	Pertanyaan jumlah total <i>file</i> kartu TF
0x48	Pertanyaan jumlah total <i>file</i> U-disk
0x49	Pertanyaan jumlah total <i>file</i> kartu <i>Flash</i>
0x4A	Terus
0x4B	Pertanyaan melacak kartu TF saat ini
0x4C	Pertanyaan melacak U-disk saat ini
0x4D	Pertanyaan melacak <i>Flash</i> saat ini



Gambar 2.12 Koneksi *Serial* Modul Mp3 dengan Arduino UNO

Dengan koneksi tersebut dapat mengontrol DFPlayer *mini* melalui *serial command*. Lalu mengkoneksikan *serial* DFPlayer *mini* ke Arduino UNO menggunakan *library* DFPLayer yang terdapat pada *library* V.2.0 dan *library software serial*.