

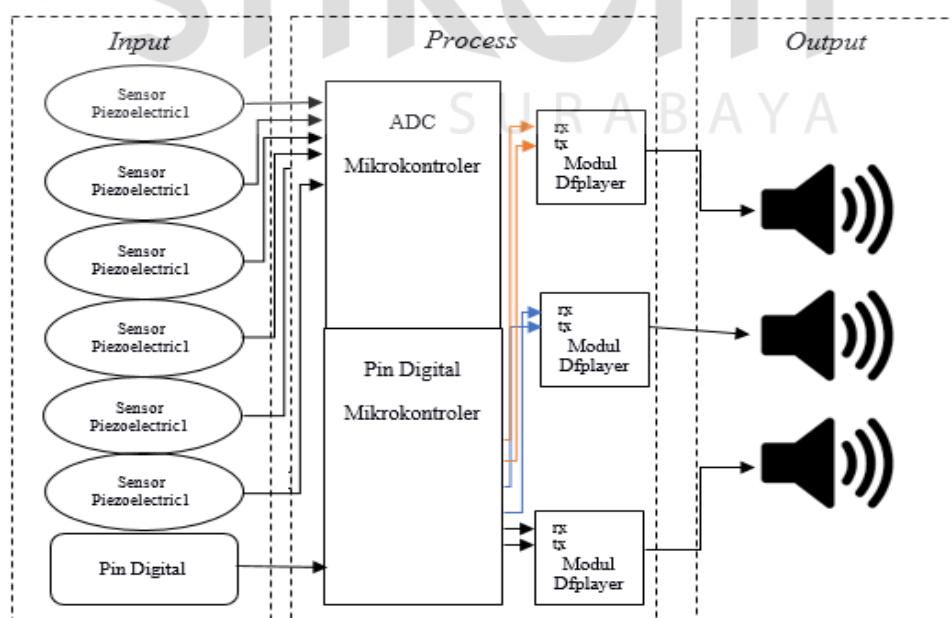
BAB III

METODE PENELITIAN

3.1 Metode Penelitian

Metode penelitian yang digunakan pada tugas akhir ini melalui beberapa tahapan penelitian dan mencari informasi tentang data yang dibutuhkan dalam mengerjakan tugas akhir ini. Tahap pertama adalah pengembangan konsep penelitian berdasarkan daftar pustaka. Studi kepustakaan dilakukan untuk mencari teori atau informasi dari buku, jurnal, dan artikel-artikel yang berkaitan dengan permasalahan. Dari data-data yang diperoleh, selanjutnya dilakukan perencanaan penelitian meliputi perancangan sistem perangkat keras dan perangkat lunak.

Untuk mendapatkan hasil yang dikehendaki dibutuhkan suatu rancangan agar dapat mempermudah dalam memahami sistem yang akan dibuat, oleh karena itu akan dibuat seperti gambar 3.1.



Gambar 3.1 Blok Diagram

1) INPUT

Sensor *piezoelectric* yang terpasang pada setiap drum pad, berguna untuk mendekksi besar getaran yang dihasilkan dari pukulan yang dilakukan terhadap drum pad, hasil keluaran dari sensor ini berupa data *analog*. *Output* sensor ini terhubung ke *port ADC* pada *microcontroller*.

2) PROSES

Microcontroller bertugas untuk mengolah data dari *input* sensor – sensor pada drum *pad*. *Output* sensor *piezoelectric* yang berupa data *analog* akan diubah menjadi data *digital* dengan menguunakan fitur *ADC* pada *microcontroller*, yang kemudian diteruskan untuk dikirim menuju modul *DFplayer*. *DFplayer* bertugas menerima data digital yang dikirim oleh arduino, data digital yang diterima akan diproses untuk menentukan suara yang akan diputar berdasarkan kondisi yang dikirimkan oleh mikrokontroler.

3) OUTPUT

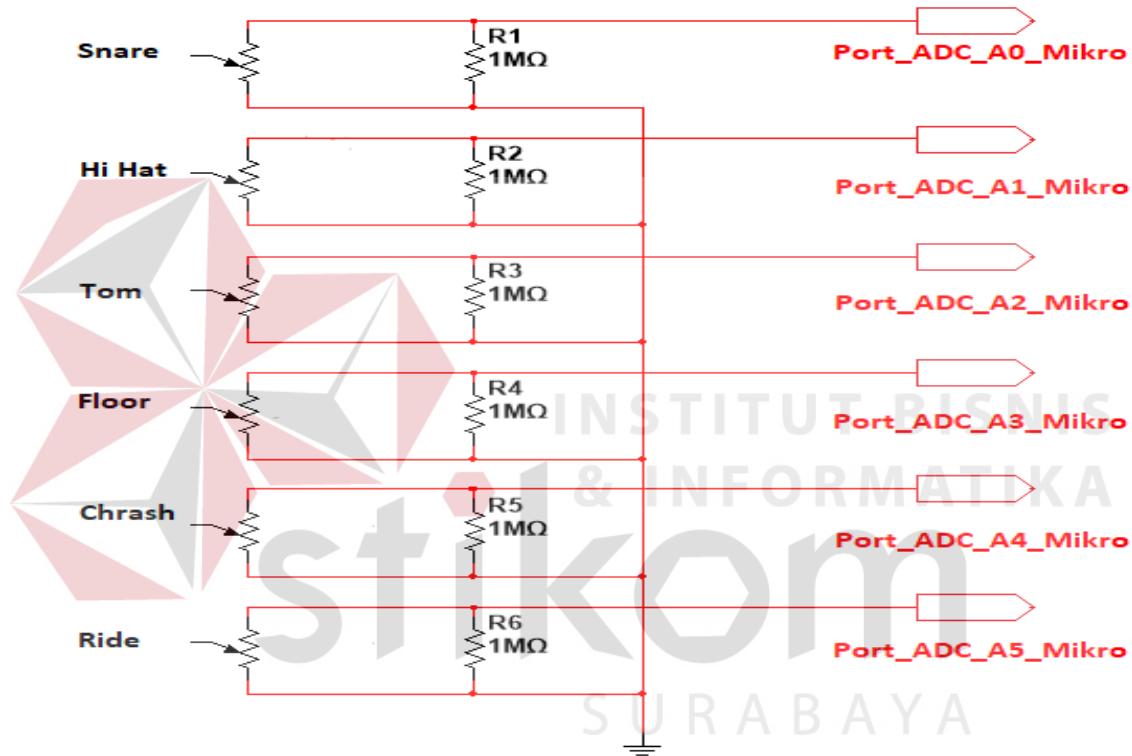
Speaker atau pengeras suara yang terhubung langsung ke *DFplayer* bertugas untuk mengeluarkan suara yang telah ditentukan oleh *DFplayer* untuk dimainkan.

3.2 Perancangan Perangkat Keras (*Hardware*)

Perancangan perangkat keras akan dibahas bagaimana komponen-komponen elektronika yang terhubung pada *hardware* dengan mikrokontroler agar elektronika pendukung dapat bekerja sesuai dengan sistem yang diharapkan.

3.2.1 Perancangan Sensor Piezoelectric

Mendeteksi pukulan pada *drum pad* secara elektronik dapat dilakukan dengan menggunakan alat yang dinamakan sensor. Sensor yang digunakan pada penelitian ini adalah sensor *piezoelectric*. Pada penelitian ini menggunakan 6 buah sensor *piezoelectric* yang ditempatkan pada 6 *pad-pad* pada *drum* yang telah dibuat, adapun perancangan rangkaian sensor *piezoelectric* ditunjukkan pada gambar 3.2.



Gambar 3.2 Rangkaian Sensor *Piezoelectric*

Sensor *piezoelectric* memiliki *output* berupa tegangan ketika sensor mendapat tekanan atau getaran. Agar data tegangan keluaran dari sensor *piezoelectric* dapat diolah oleh mikrokontroler maka sensor *piezoelectric* dihubungkan ke resistor. Fungsi resistor dalam rangkaian sensor ini adalah untuk membatasi tegangan dan arus yang dihasilkan oleh sensor *piezoelectric* dan untuk melindungi input analog.

Pada rangkaian diatas, dilakukan *trial and error* untuk mencari nilai *threshold* pada sensor *piezoelectric*, dimana nilai *threshold* ini diambil dari nilai keluaran tertinggi yang dikeluarkan oleh sensor *piezoelectric* saat sensor dalam keadaan diam atau dalam keadaan tidak diberi masukan. Pencarian *threshold* dilakukan dengan cara menghubungkan sensor dengan arduino kemudian mengeluarkan nilai sensor pada *serial monitor* pada aplikasi arduino IDE, dan melihat nilai keluaran tertinggi yang dikeluarkan oleh sensor dalam keadaan diam. Hal ini dilakukan karena sensor *piezoelectric* sangat sensitif dalam mendekripsi getaran-getaran yang ada disekitar. Penentuan nilai *threshold* berguna agar mikrokontroller arduino hanya memproses data yang berasal dari pemberian *input* tekanan terhadap sensor.

Melalui *trial and error* pencarian *threshold*, didapati bahwa nilai keluaran tertinggi sensor adalah sebesar 180. Sehingga rentang nilai *adc* yang dikeluarkan oleh rangkaian ini pada masing-masing sensor adalah 180 -1023 .

3.2.2 Perancangan Sensor *Switch*

Pada penelitian ini sensor *switch* digunakan untuk mendekripsi pijakan pada pedal *bass drum* yang dipasangkan pada pedal yang telah dibuat. Keluaran dari sensor *switch* menghasilkan kondisi *high* jika tombol pada sensor ditekan dan akan menghasilkan kondisi *low* jika tombol pada sensor tidak ditekan. Sehingga keluaran dari sensor *switch* ini dihubungkan pada *port* digital mikrokontroler arduino uno sebagai *input*. Pada pembacaan sensor *switch* menggunakan fitur *pullup* internal pada mikrokontroler agar nilai yang didapat pada saat pembacaan sensor menjadi

akurat yaitu menghasilkan output *high* atau *low* saja. Adapun perancangan rangkaian sensor *switch* ditujukan pada gambar 3.3 dibawah.



Gambar 3.3 Rangkaian Sensor *Switch*.

3.2.3 *Drum Pad*

Pada tugas akhir ini *pad* yang dibuat sebanyak 6 buah yang didalamnya terdapat sensor *piezoelectric*, yaitu:

- 1) *Ride pad*
 - 2) *Middle tom pad*
 - 3) *Simbal crash pad*
 - 4) *Simbal hi-hat pad*
 - 5) *Snare pad*
 - 6) *Floor tom pad*

Drum pad yang telah dibuat dapat dilihat pada gambar 3.4 dibawah ini.



Gambar 3.4 *Drum Pad*

3.2.4 Pedal Drum

Pada tugas akhir ini pedal yang di buat sebanyak 1 buah yang didalamnya terdapat sensor *switch*, pedal yang dibuat yaitu:

- 1) Pedal *bass*

Pedal *drum* yang telah dibuat dapat dilihat pada gambar 3.5.

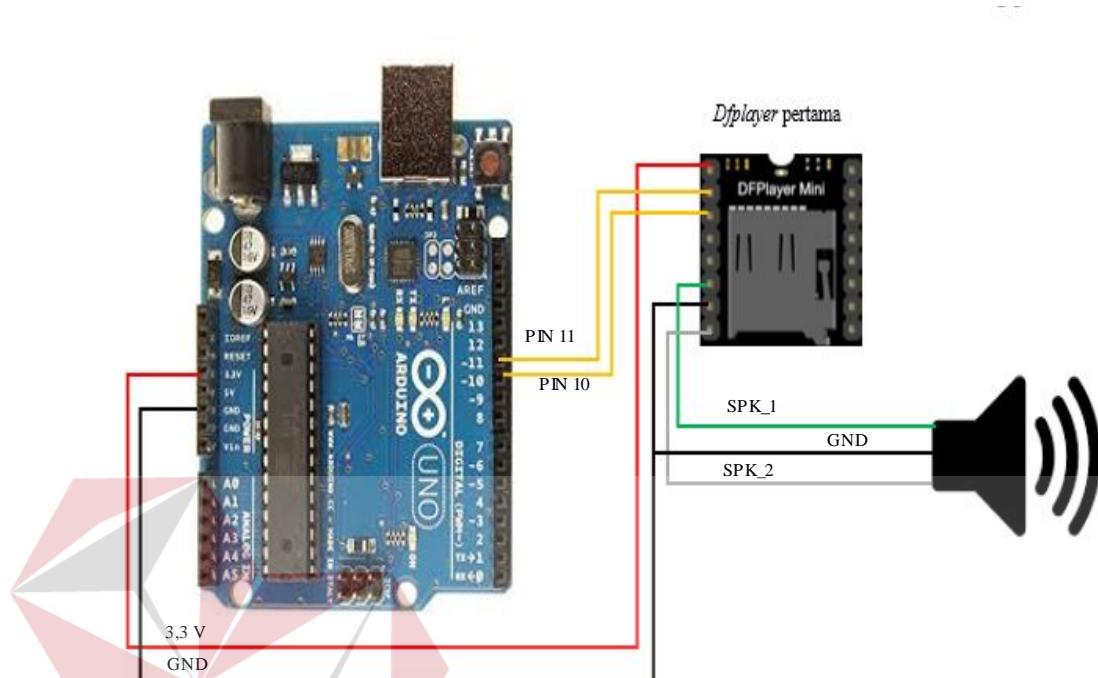


Pedal *bass drum* dirancang menggunakan sensor switch dan dihubungkan pada pin digital arduino, nilai yang dikeluarkan oleh pedal switch ini adalah 1 & 0

3.2.5 Perancangan *DFplayer*

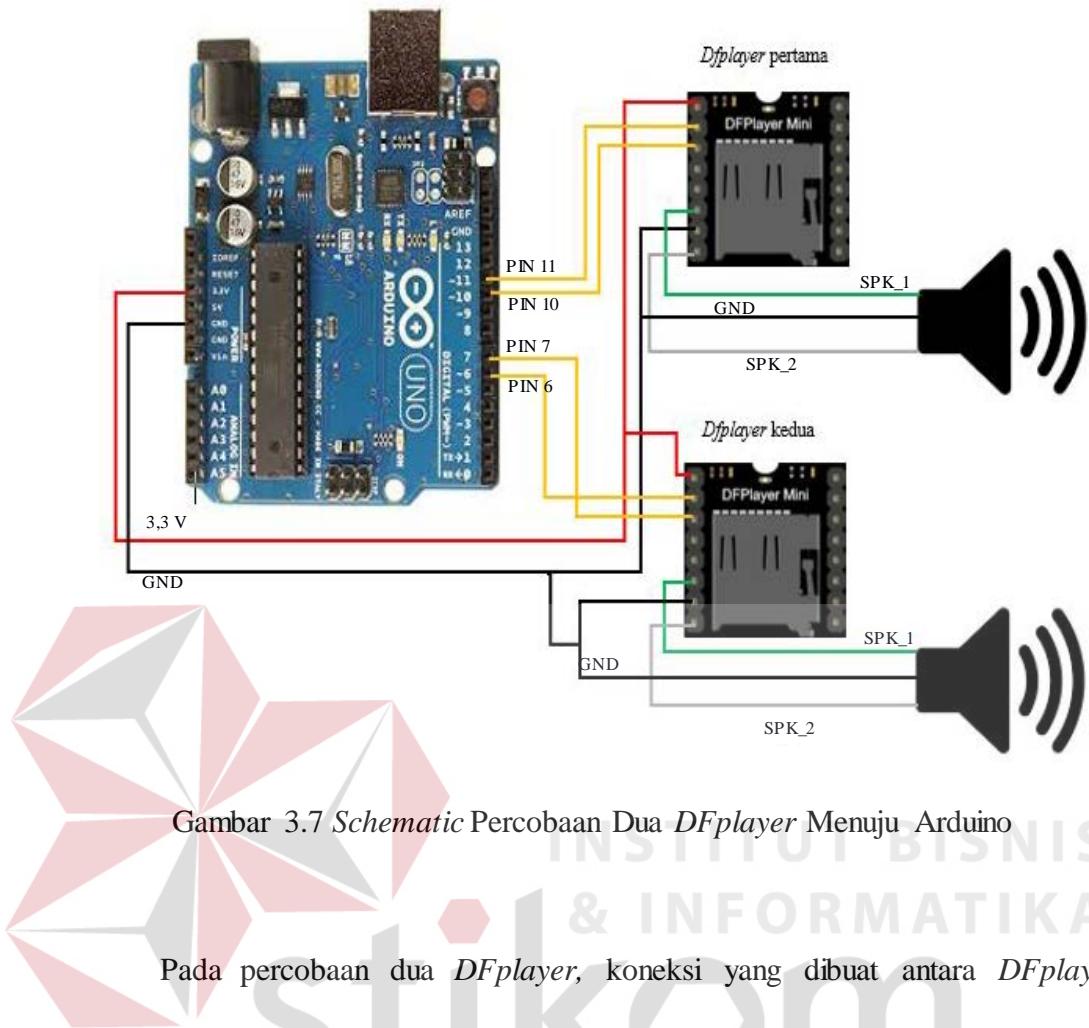
DFplayer digunakan sebagai perangkat yang akan memutar suara *drum* yang tersimpan pada micro sd *DFplayer*. Pada penelitian ini dilakukan tiga kali percobaan yaitu menggunakan satu *DFplayer* pada sistem, dua *DFplayer* pada sistem, dan tiga *DFplayer* pada sistem. *Schematic DFplayer* menuju arduino dapat

dilihat pada gambar 3.6 untuk percobaan satu *DFplayer*, pada gambar 3.7 untuk percobaan dua *DFplayer*



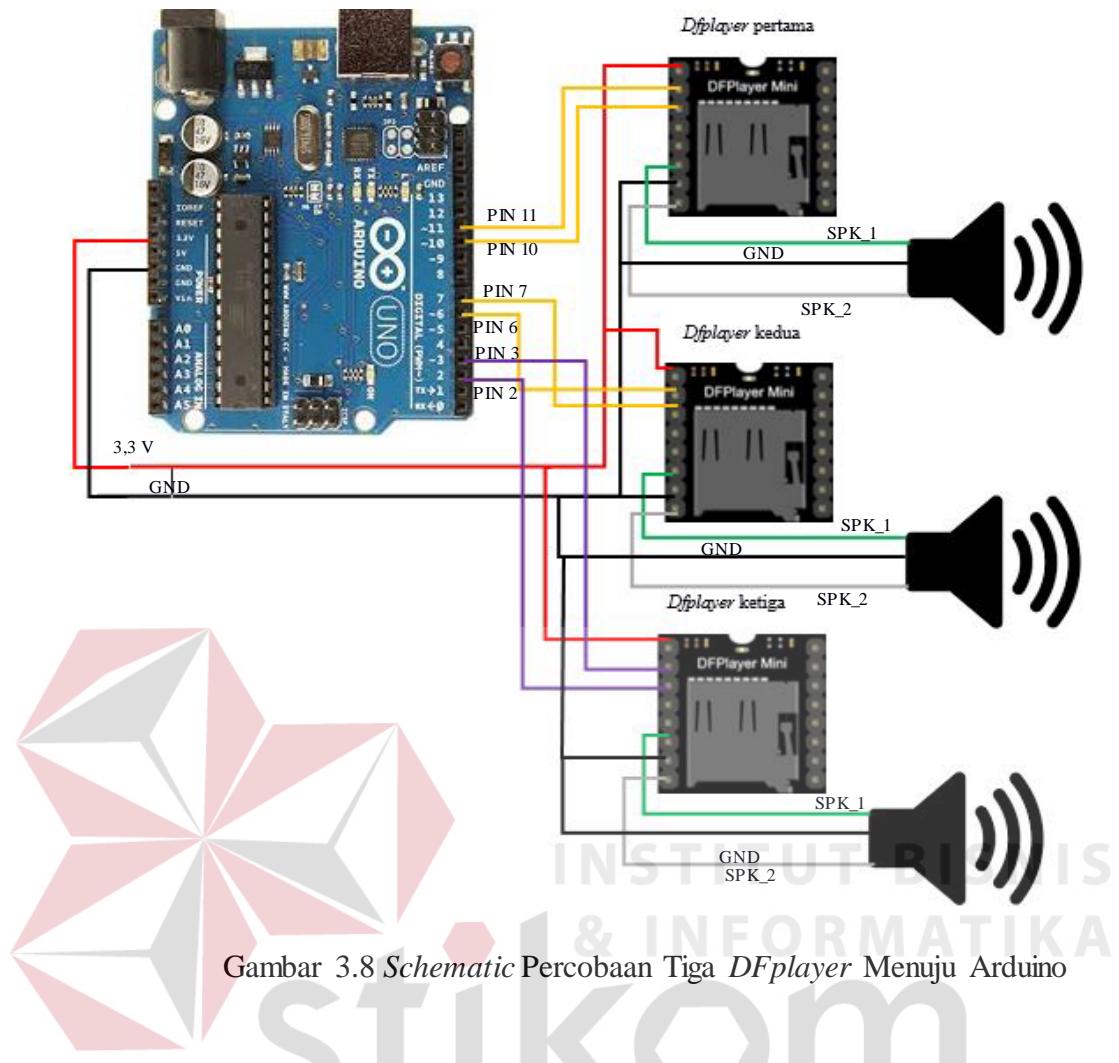
Gambar 3.6 Schematic Percobaan Satu *DFplayer* Menuju Arduino

Pada percobaan satu *DFplayer*, koneksi yang dibuat antara *DFplayer* pertama dan arduino diantaranya adalah vcc modul *DFplayer*, dihubungkan dengan pin 3,3 volt arduino, rx pada *DFplayer* dihubungkan dengan pin 10 arduino, tx *DFplayer* dihubungkan dengan pin 11 arduino. Spk1 dan spk2 pada *DFplayer* langsung dihubungkan ke pengeras suara



Gambar 3.7 Schematic Percobaan Dua *DFplayer* Menuju Arduino

Pada percobaan dua *DFplayer*, koneksi yang dibuat antara *DFplayer* menuju arduino tetap sama seperti percobaan satu *DFplayer*, namun pada percobaan dua *DFplayer*, mendapat penambahan satu buah *DFplayer* yang akan dimasukan ke dalam *schematic* perancangan. Koneksi *DFplayer* kedua menuju arduino adalah vcc modul *DFplayer*, dihubungkan dengan pin 3,3 volt arduino, rx pada *DFplayer* dihubungkan dengan pin 6 arduino, tx *DFplayer* dihubungkan dengan pin 7 arduino. Spk1 dan spk2 pada *DFplayer* langsung dihubungkan ke pengeras suara



Gambar 3.8 Schematic Percobaan Tiga *DFplayer* Menuju Arduino

Perancangan pada percobaan ketiga ini tidak jauh berbeda pada kedua percobaan sebelumnya. Pada percobaan ini akan ditambahkan satu buah *DFplayer* yang dikoneksikan dengan arduino. Koneksi *DFplayer* ketiga kedua menuju arduino adalah vcc modul *DFplayer*, dihubungkan dengan pin 3,3 volt arduino, rx pada *DFplayer* dihubungkan dengan pin 2 arduino, tx *DFplayer* dihubungkan dengan pin 3 arduino. Spk1 dan spk2 pada *DFplayer* langsung dihubungkan ke pengeras suara.

3.2.6 Koneksi *Port* pada Modul Mikrokontroler Arduino Uno

Pada penelitian ini, modul mikrokontroler yang digunakan adalah modul mikrokontroler arduino uno. koneksi *port* pada arduino uno dapat dilihat pada tabel 3.1.

Tabel 3.1 Koneksi *Port* Arduino Uno.

No	Port Arduino Uno	Keterangan
1	A0	Terhubung dengan sensor 1 sebagai <i>input Snare</i>
2	A1	Terhubung dengan sensor 2 sebagai <i>input Hi-hat</i>
3	A2	Terhubung dengan sensor 3 sebagai <i>input Tom</i>
4	A3	Terhubung dengan sensor 4 sebagai <i>input Floor</i>
5	A4	Terhubung dengan sensor 5 sebagai <i>input Crash</i>
6	A5	Terhubung dengan sensor 6 sebagai <i>input Ride</i>
7	D8	Terhubung dengan sensor <i>switch</i> 1 sebagai <i>input Bass</i>
8	D10	Terhubung dengan RX pada modul <i>DFplayer</i> pertama
9	D11	Terhubung dengan TX pada modul <i>DFplayer</i> pertama
10	D6	Terhubung dengan RX pada modul <i>DFplayer</i> kedua
11	D7	Terhubung dengan TX pada modul <i>DFplayer</i> kedua
12	D2	Terhubung dengan RX pada modul <i>DFplayer</i> ketiga
13	D3	Terhubung dengan TX pada modul <i>DFplayer</i> ketiga
14	5V	Terhubung dengan modul sensor sebagai <i>supply</i>
15	GND	Terhubung dengan <i>ground</i> pada modul sensor

3.3 Perancangan Perangkat Lunak (*Software*)

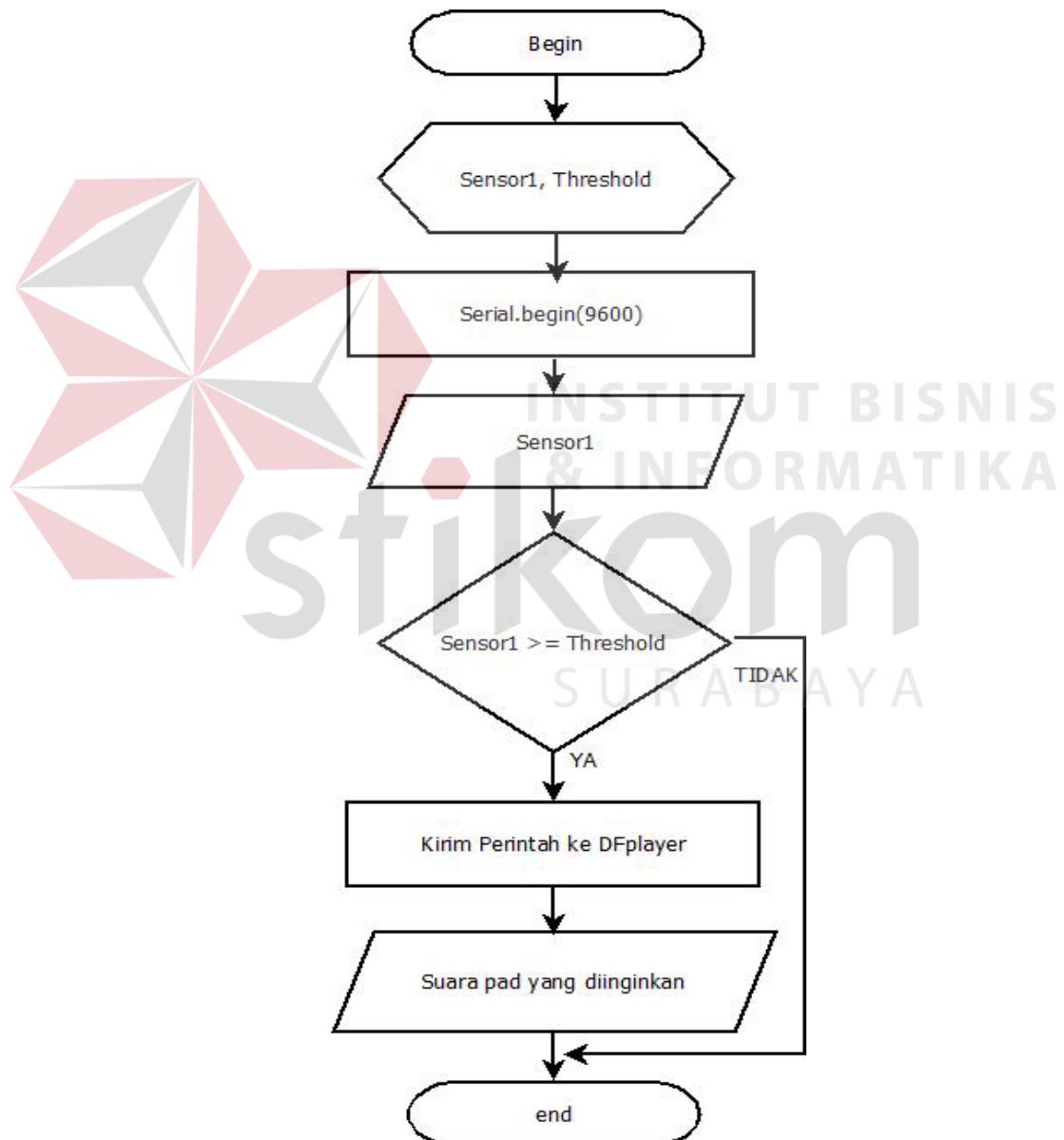
Selain perancangan berupa *hardware*, juga dibutuhkan perancangan berupa *software* yang berguna untuk menjalankan perancangan *hardware* yang telah dibuat. Perancangan *software* terdiri dari beberapa algoritma perancangan dari sistem yang ditangani oleh pengontrol.

3.3.1 Perancangan Program pada Modul Mikrokontroler Arduino

Software yang digunakan untuk memprogram arduino tersebut ialah *software* Arduino IDE. Mikrokontroler arduino uno bertugas untuk mengolah data

dari *output* sensor-sensor pada *drum pad*, dan mengirimkan data hasil pembacaan sensor pada android *smartphone*.

Pada Tugas Akhir ini sebelum membuat program pada arduino uno, dibuat *flowchart* terlebih dahulu yang bertujuan untuk membantu agar nantinya memudahkan dalam membuat program pada arduino uno. Adapun perancangan *flowchart* pada arduino uno dapat dilihat pada Gambar 3.9.

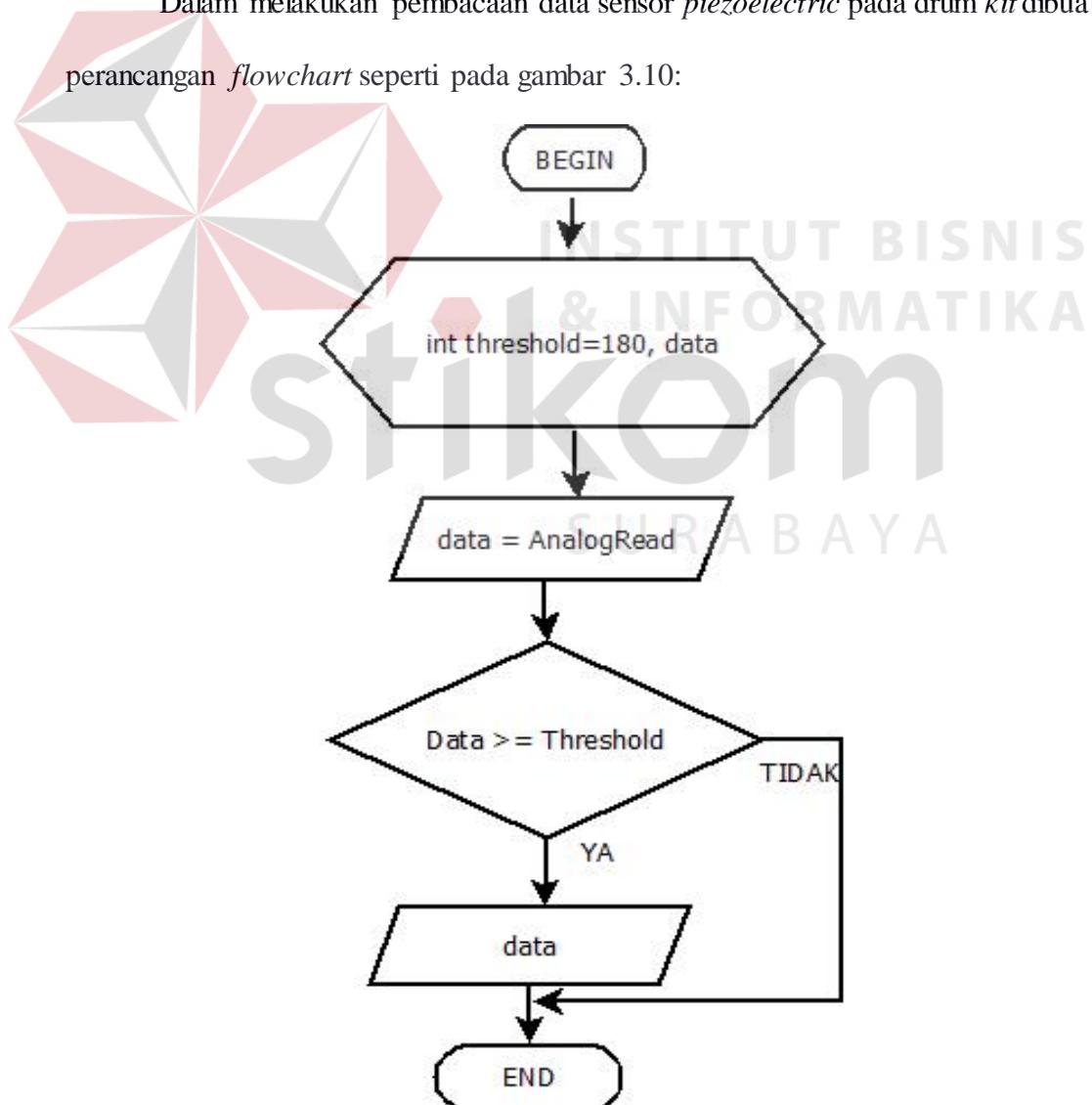


Gambar 3.9 *Flowchart* pada Arduino

Pada saat arduino dijalankan pertama kali yang dilakukan adalah proses menginisialisasi *variabel-variabel* yang dibutuhkan, setelah itu arduino akan melakukan pengecekan terhadap serial-serial yang terhubung modul *DFplayer* Jika ada, maka akan melakukan proses pembacaan data dari sensor. Setelah proses pembacaan data dari sensor dilakukan maka akan melanjutkan proses selanjutnya yaitu proses pengiriman data menuju modul *DFplayer* melalui komunikasi serial.

3.3.2 Pembacaan Data Sensor *Piezoelectric*

Dalam melakukan pembacaan data sensor *piezoelectric* pada drum *kit* dibuat perancangan *flowchart* seperti pada gambar 3.10:



Gambar 3.10 *Flowchart* Pembacaan Data Sensor *Piezoelectric* Arduino Uno

Pada saat pembacaan data sensor *piezoelectric* dilakukan pengelompokan nilai pada setiap masukan sensor yang masuk dari data yang telah dibaca oleh mikrokontroler arduino, pengelompokan nilai rata-rata sensor mulai dilakukan ketika mikrokontroler membaca nilai sensor melebihi nilai *threshold* yang sudah ditentukan.

Nilai *threshold* adalah acuan nilai terbesar dari sensor ketika tidak ada inputan dari *user*. Penentuan nilai *threshold* sensor *piezoelectric* dilakukan dengan cara mencari nilai awal sensor yang terbaca ketika sensor tidak diberikan *inputan* berupa tekanan.

Pengelompokan nilai pada masukan dilakukan untuk menentukan besar volume suara yang nantinya akan dikeluarkan oleh modul *DFplayer*.

3.3.3 Perancangan Produksi Suara pada *DFplayer*

Agar dalam penelitian ini mendapatkan hasil keluaran suara yang sesuai dengan yang diharapkan, maka dilakukan suatu perancangan suara berdasarkan tiga percobaan yang dilakukan oleh penelitian ini. Yaitu perancangan produksi suara ketika menggunakan satu buah *DFplayer*, perancangan produksi suara ketika menggunakan dua buah *DFplayer*, dan perancangan produksi suara ketika menggunakan tiga buah *DFplayer*. Perancangan produksi suara pada penelitian ini adalah mengelompokkan penyimpanan masing-masing suara. Pada tabel 3.1, tabel 3.2, dan tabel 3.4 dijelaskan pengelompokan suara yang dilakukan pada setiap percobaan.

Tabel 3.2 Perancangan Produksi Suara pada Percobaan Satu *DFplayer*

Penyimpanan suara	Suara Drum
<i>DFplayer</i> pertama	<i>Snare</i>
	<i>Hi-hat</i>
	<i>Tom</i>
	<i>Bass Drum</i>
	<i>Floor</i>
	<i>Crash</i>
	<i>Ride</i>

Tabel 3.3 Perancangan Produksi Suara pada Percobaan Dua *DFplayer*

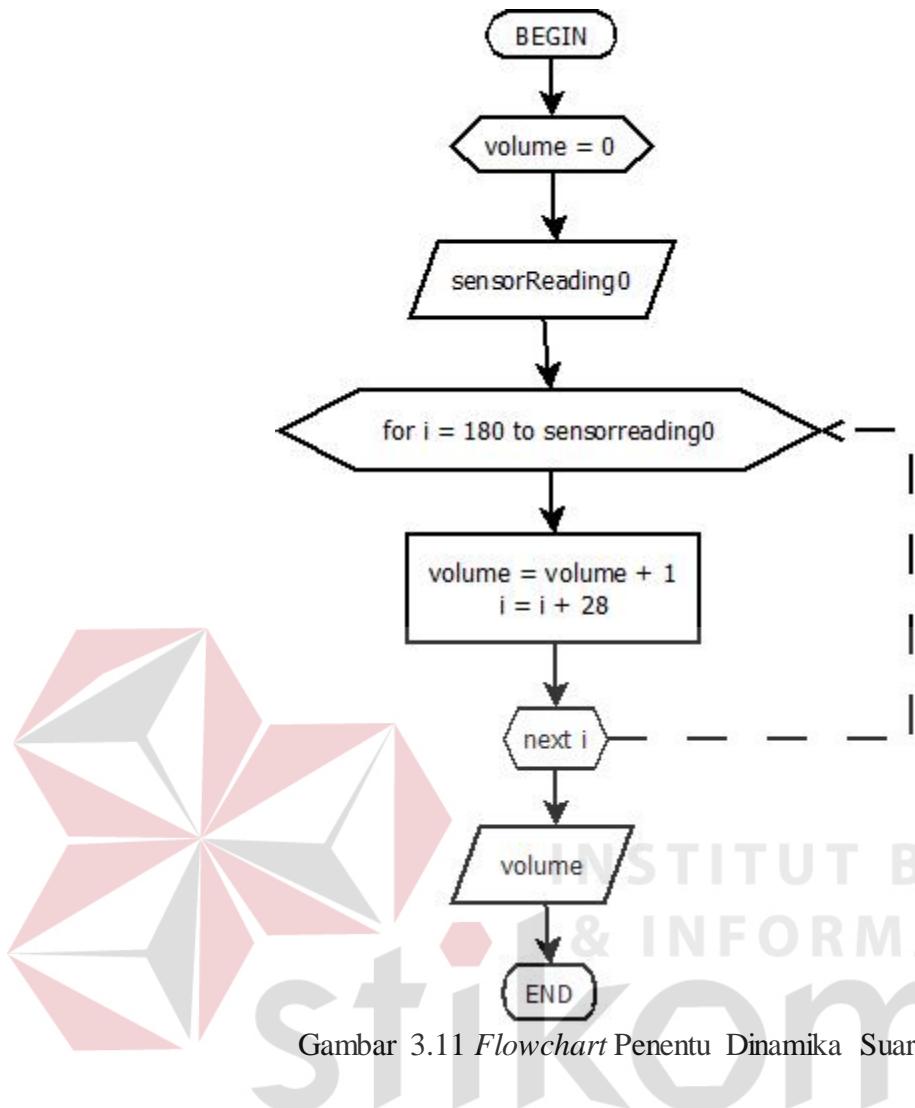
Penyimpanan suara	Suara Drum
<i>DFplayer</i> pertama	<i>Snare</i>
	<i>Hi-hat</i>
	<i>Tom</i>
<i>DFplayer</i> kedua	<i>Bass Drum</i>
	<i>Floor</i>
	<i>Crash</i>
	<i>Ride</i>

Tabel 3.4 Perancangan Produksi Suara pada Percobaan Tiga *DFplayer*

Penyimpanan suara	Suara Drum
<i>DFplayer</i> pertama	<i>Snare</i>
	<i>Tom</i>
	<i>Hi-hat</i>
<i>DFplayer</i> kedua	<i>Crash</i>
	<i>Ride</i>
	<i>Bass Drum</i>
<i>DFplayer</i> ketiga	<i>Floor tom</i>

3.3.4 Perancangan Produksi Dinamika pada *DFplayer*

Untuk menghasilkan keluaran suara yang dinamis, dalam hal ini yaitu besar, kecilnya suara yang dikeluarkan oleh *DFplayer* berdasarkan *input* dari sensor *piezoelectric*. Maka dilakukan suatu perancangan produksi dinamika pada *DFplayer*. Pada Gambar 3.11.



Gambar 3.11 Flowchart Penentu Dinamika Suara

Saat sensor *piezoelectric* diberi masukan, arduino memilih nilai masukan dari sensor *piezoelectric* sebagai acuan untuk memilih pada tingkatan dinamika mana *DFplayer* diputar. Nilai threshold pada masing-masing pad adalah 180 digunakan sebagai titik awal dalam menentukan volume. Volume pada *DFplayer* berada pada rentang 0 - 30. Besar masukan pada sensor adalah 180-1023. Pemilihan dilakukan dengan rumus sebagai berikut.

$$1023 - 180 = 843 \Rightarrow \text{Rentang jarak masukan sensor reading}$$

$$\frac{\text{Rentang jarak masukan sensor reading}}{\text{Jumlah volume pada } DFplayer} = \frac{843}{30} = 28$$

$$\text{maka dihasilkan rumus : } \frac{\text{sensorReading} - 180}{28} = \text{volume}$$

Melalui perumusan diatas, volume *DFplayer* dalam mewakili setiap nilai keluaran sensor piezoelectric dapat dimuat seperti pada tabel 3.5.

Tabel 3.5 Volume *DFplayer* Terhadap Masukan Sensor

Nilai keluaran Sensor	Volume <i>DFplayer</i>
180 – 208	1
209 – 236	2
237 - 264	3
265 – 292	4
293 - 320	5
321 - 348	6
349 - 376	7
377 - 404	8
405 - 432	9
433 - 460	10
461 - 488	11
489 - 516	12
517 - 544	13
545 - 572	14
573 - 600	15
601 - 628	16
629 - 656	17
657 - 684	18
685 - 712	19
713 - 740	20
741 - 768	21
769 - 796	22
797 - 824	23
825 - 852	24
853 - 880	25
881 - 908	26
909 - 936	27
937 - 964	28
965 - 992	29
992 - 1023	30

3.4 Perancangan Keseluruhan Sistem

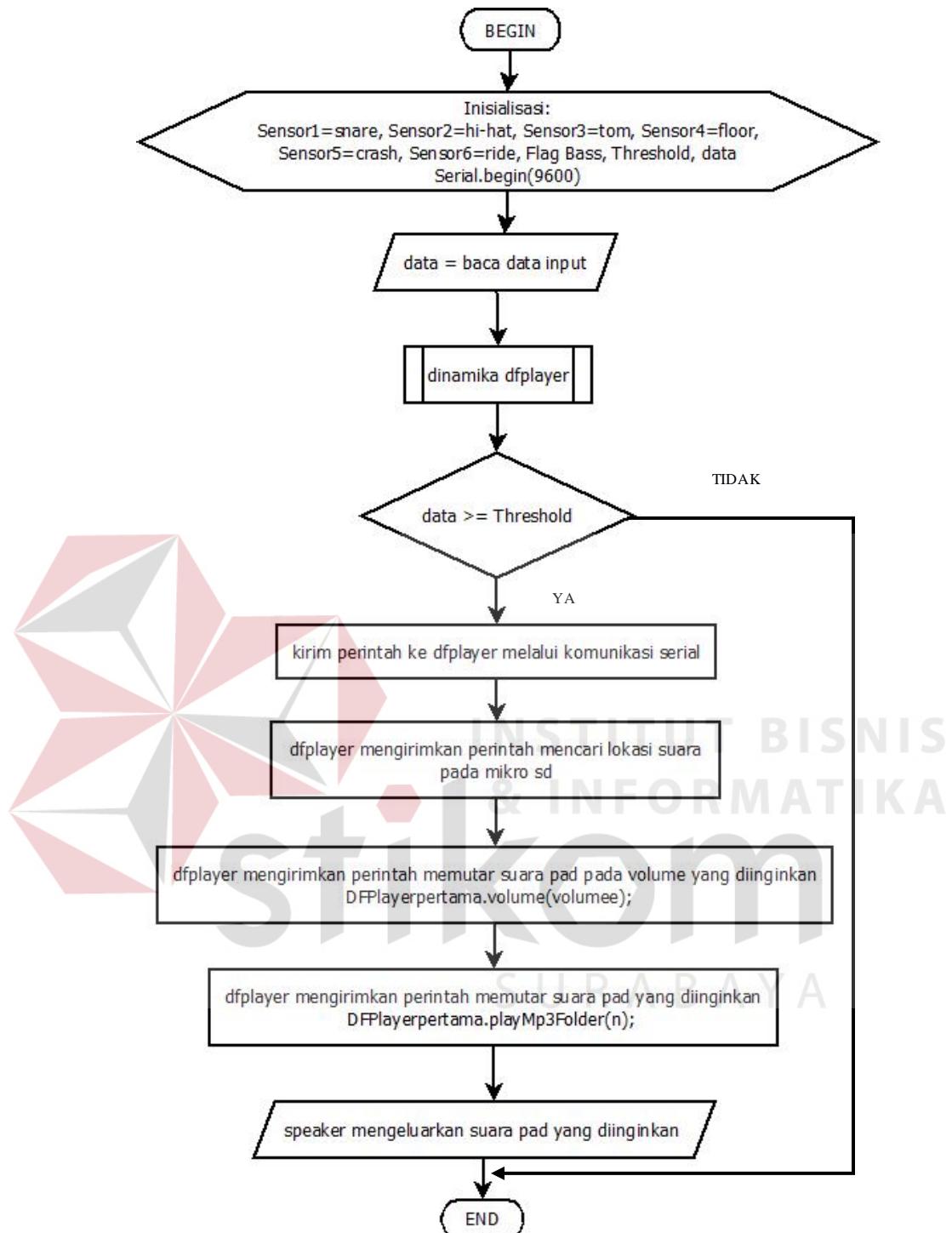
Untuk dapat mengamati ketiga percobaan yang dilakukan dalam penelitian ini, maka perancangan sistem memiliki tiga jenis sistem yang diterapkan dalam

penelitian ini, sistem menggunakan satu *DFplayer*, sistem menggunakan dua *DFplayer*, & sistem menggunakan tiga *DFplayer*.

3.4.1 Sistem Menggunakan Satu *DFplayer*

Dalam sistem menggunakan satu *DFplayer* seluruh suara *drum* disimpan dan dieksekusi oleh satu buah *DFplayer*. Flowchart pada Gambar 3.12.





Gambar 3.12 Flowchart Sistem Menggunakan Satu *DFplayer*

Ketika sistem menggunakan satu *DFplayer* seluruh suara disimpan pada *sd card* pada *DFplayer* pertama. Proses dimulai dengan inisialisasi masing-masing masukan pada pin digital arduino, ketika ada masukan pada salah satu pin digital,

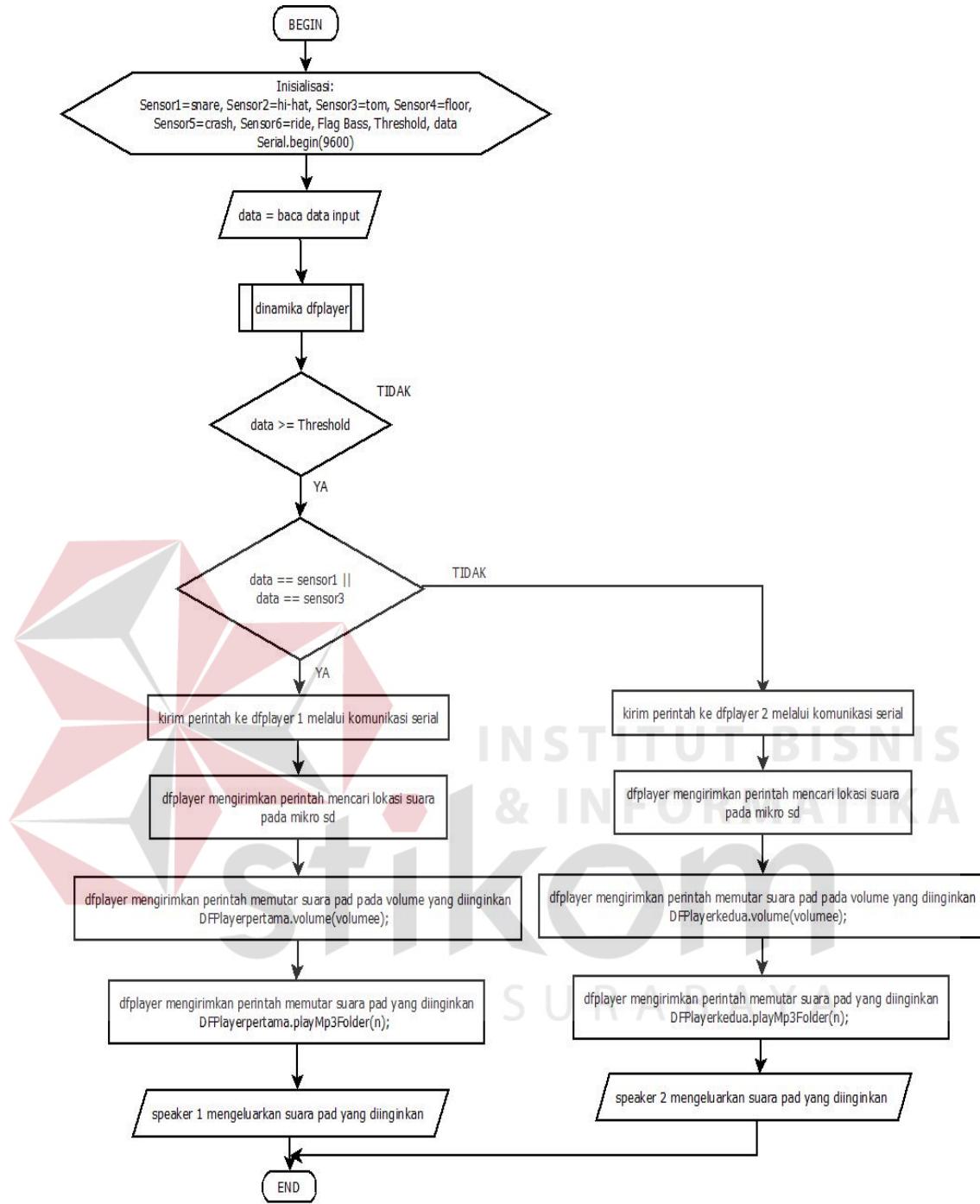
arduino akan memproses masukan tersebut dengan menangkap besar masukan, kemudian diproses untuk menentukan dinamika volume kemudian mengirimkan perintah ke *DFplayer* melalui komunikasi serial, untuk memutar suara dan besar volume yang dikeluarkan *DFplayer*

3.4.2 Sistem Menggunakan Dua *DFplayer*

Dalam sistem menggunakan dua *DFplayer* suara drum dikelompokkan menjadi dua bagian dan setiap kelompok suara disimpan di salah satu *DFplayer*. Suara drum dieksekusi oleh *DFplayer* tempat penyimpanannya. Flowchart pada

Gambar 3.13.



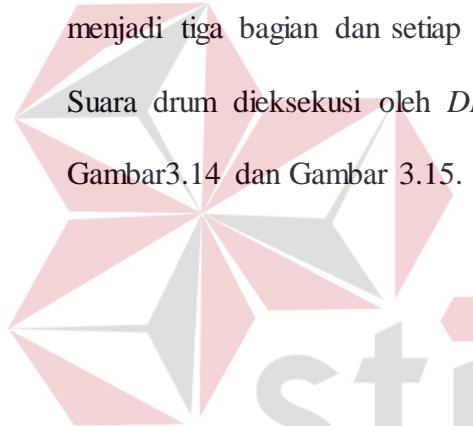
Gambar 3.13 Flowchart Sistem Menggunakan Dua *DFplayer*

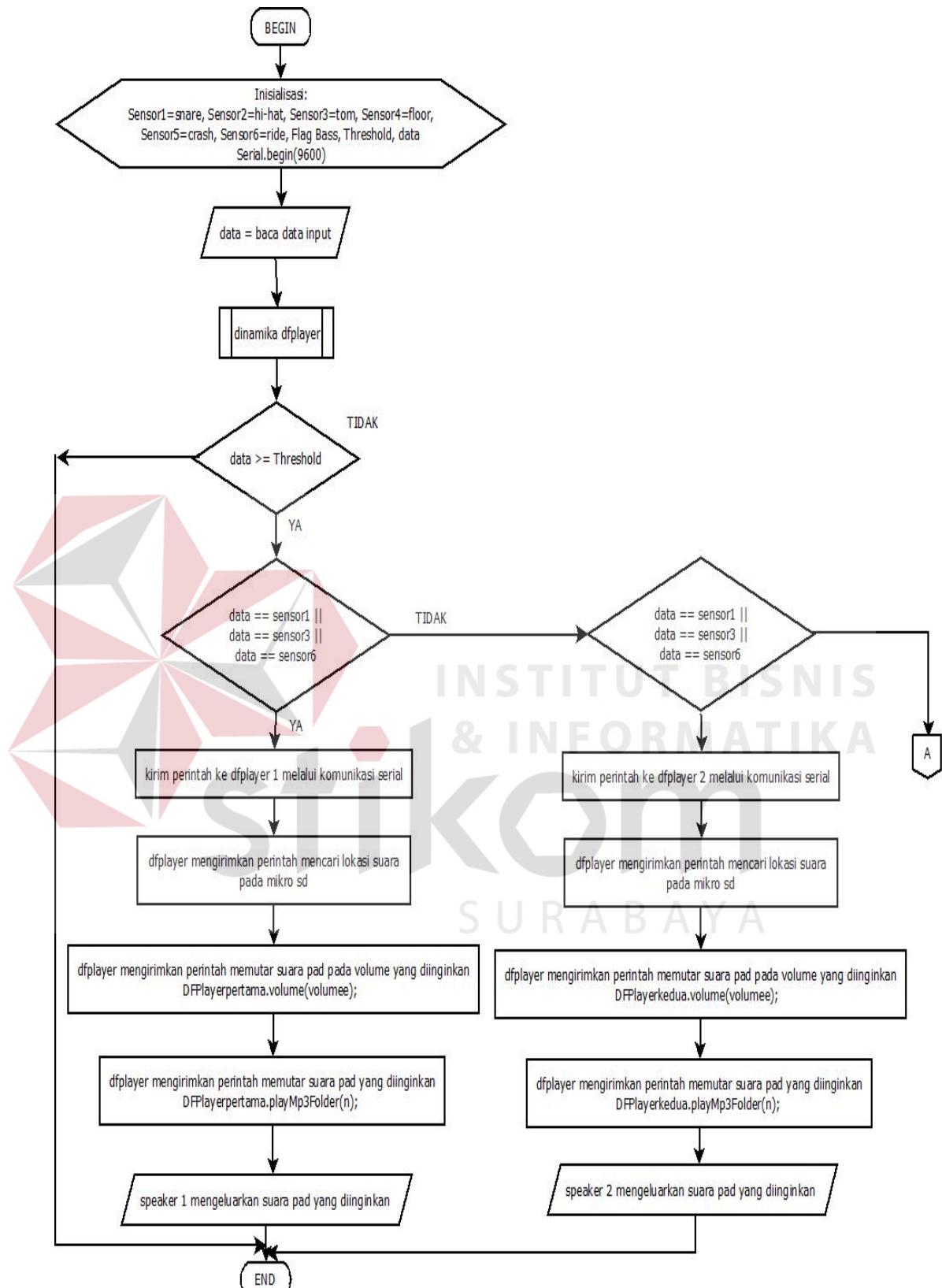
Ketika sistem menggunakan dua *DFplayer* suara dikelompokkan menjadi dua bagian. Sebagian disimpan pada *sd card* pada *DFplayer* pertama, sebagian lagi disimpan pada *sd card* pada *DFplayer* kedua (lihat Tabel3.3). Proses dimulai dengan inisialisasi masing-masing masukan pada pin digital arduino, ketika ada

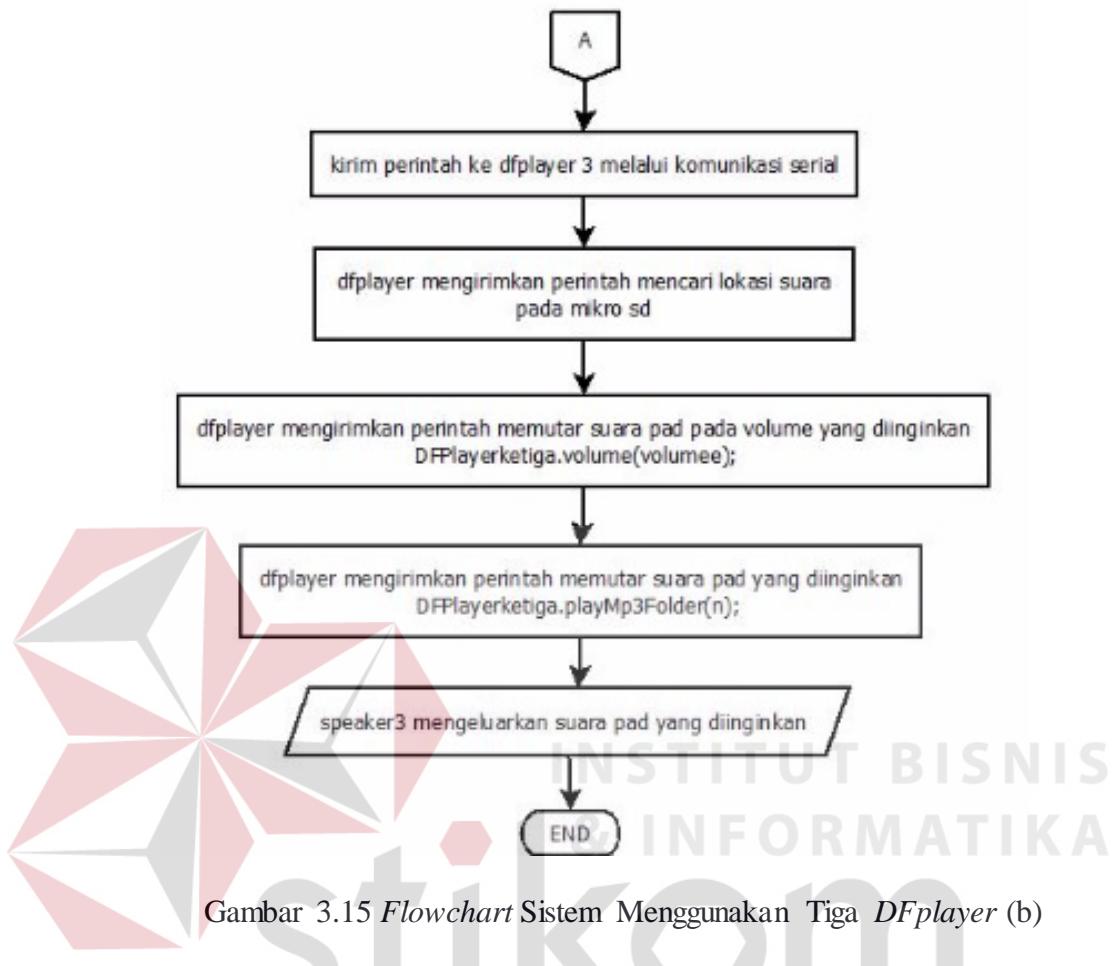
masukan pada salah satu pin digital, arduino akan memproses masukan tersebut dengan menangkap besar masukan, kemudian diproses untuk menentukan dinamika volume kemudian mengirimkan perintah ke *DFplayer* tempat suara disimpan melalui komunikasi serial, untuk memutar suara dan besar volume yang dikeluarkan *DFplayer*.

3.4.3 Sistem Menggunakan Tiga *DFplayer*

Dalam sistem menggunakan tiga *DFplayer* suara drum dikelompokkan menjadi tiga bagian dan setiap kelompok suara disimpan di salah satu *DFplayer*. Suara drum dieksekusi oleh *DFplayer* tempat penyimpanannya. Flowchart pada Gambar 3.14 dan Gambar 3.15.



Gambar 3.14 Flowchart sistem menggunakan tiga *DFplayer* (a)



Ketika sistem menggunakan tiga *DFplayer* suara dikelompokkan menjadi 3 bagian. Bagian pertama disimpan pada *sd card* pada *DFplayer* pertama, bagian kedua disimpan pada *sd card* pada *DFplayer* kedua, bagian ketiga disimpan pada *sd card* pada *DFplayer* ketiga (lihat Tabel3.4). Proses dimulai dengan inisialisasi masing-masing masukan pada pin digital arduino, ketika ada masukan pada salah satu pin digital, arduino akan memproses masukan tersebut dengan menangkap besar masukan, kemudian diproses untuk menentukan dinamika volume kemudian mengirimkan perintah ke *DFplayer* tempat suara disimpan melalui komunikasi serial, untuk memutar suara dan besar volume yang dikeluarkan *DFplayer*.

3.5 Metode Pengujian dan Evaluasi Sistem

Untuk mengetahui apakah sistem yang dibuat dapat berjalan sesuai yang diharapkan, maka akan dilakukan tiga kali percobaan pengujian. Yaitu pengujian sistem menggunakan satu buah *DFplayer*, pengujian sistem menggunakan dua buah *DFplayer*, dan pengujian menggunakan tiga buah *DFplayer*. Untuk memastikan apakah sistem dapat mengeksekusi masukan secara pararel, memproses, dan mengeluarkannya.

Evaluasi sistem untuk setiap tahapan-tahapan dalam pembuatan sistem. Dimulai dari pembacaan data dari sensor dengan memukul masing-masing pad, pengiriman data dari mikrokontroler arduino, dan program pada mikrokontroler arduino.

