

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang Masalah**

Bagi umat Islam, sangat penting untuk menghafal Al-Qur'an karena memiliki banyak keutamaan. Salah satu keutamaannya yaitu Al-Qur'an menjadi syafaat bagi penghafalnya. Memiliki anak yang masih di bangku Sekolah Dasar menjadi seorang penghafal Al-Qur'an merupakan impian besar bagi orang tua. Kebanyakan orang tua mendapati kendala seperti banyaknya ayat yang dihafal sehingga membuat anak malas dan hanya ingin bermain. Kondisi ini seringkali membuat anak merasa terbebani dan kurang termotivasi untuk menghafal ayat-ayat Al-Qur'an, karena mereka menganggapnya sebagai tugas yang membosankan.

Kendala lain adalah anak merasa bosan karena metodenya hanya membaca kitab. Metode menghafal Al-Qur'an bagi siswa Sekolah Dasar melalui pendekatan hafalan atau mengulang bacaan. Contoh menghafal Al-Qur'an dengan metode talaqqi yaitu ustad atau guru membacakan ayat dan para murid memperhatikan. Guru dan murid kemudian membaca ayat bersama. Jika murid salah dalam membaca atau belum hafal maka diulangi dari awal. Metode ini memiliki kekurangan seperti menguras tenaga guru lebih banyak dan tidak bisa menargetkan hafalan dengan cepat.

Kebosanan ini juga berdampak negatif pada motivasi dan semangat menghafal anak. Ketika anak merasa bosan, mereka cenderung tidak fokus dan kurang berpartisipasi aktif dalam proses menghafal. Hal ini tentunya menjadi

tantangan besar bagi orang tua. Mengatasi masalah ini, diperlukan pendekatan yang lebih kreatif dan interaktif dalam proses menghafal. Metode menghafal yang melibatkan berbagai media, seperti permainan edukatif, dapat membantu menarik minat anak-anak dan membuat mereka lebih antusias dalam menghafal.

Terdapat 2 penelitian terdahulu yang memiliki kesamaan dengan tema dan objek yang berbeda. Penelitian oleh Jauhari dkk. (2016) dengan judul “Perancangan Murottal Otomatis Menggunakan Mikrokontroller Arduino Mega 2560”. Penelitian tersebut berfokus pada kemudahan bagi penjaga masjid yang sedang tidak ada di tempat sehingga dapat melantunkan ayat Al-Qur’an ketika waktu sholat hampir tiba. Penelitian sebelumnya oleh Alifiyan dan Khizanatul (2023) dengan judul “Alat Pembelajaran Bahasa Arab Inovatif Berbasis Arduino untuk Pendidikan Usia Dini”. Penelitian tersebut berfokus pada desain media pembelajaran menggunakan Arduino Mega 2560 untuk membantu siswa mempelajari kosakata bahasa Arab.

Berdasarkan latar belakang masalah yang telah dijelaskan di atas, maka dibutuhkan metode untuk meningkatkan minat menghafal ayat surah Al-Baqarah bagi siswa Sekolah Dasar. Peneliti memilih Surah Al-Baqarah karena surah ini merupakan surah dengan ayat terbanyak dalam Al-Qur’an. Selain itu, Surah Al-Baqarah juga merupakan surah yang banyak berisi pelajaran moral, hukum, dan petunjuk bagi kehidupan sehari-hari umat manusia. Oleh karena itu, peneliti tertarik untuk merancang dan membuat **“Game tebak Ayat Surah Al-Baqarah Sebagai Media Menghafal Al-Qur’an bagi Siswa Sekolah Dasar Berbasis Arduino Mega 2560”**.

## 1.2 Perumusan Masalah

Berdasarkan latar belakang masalah yang telah peneliti jelaskan di atas, dalam melakukan penelitian ini didapat beberapa rumusan masalah yaitu sebagai berikut:

1. Bagaimana baterai Lithium-Ion dan *switch* dapat digunakan sebagai sumber arus listrik dan menyalurkannya ke seluruh komponen pada sistem?
2. Bagaimana *push button* dapat digunakan sebagai *input* berupa angka dan memilih dengar ayat atau tebak ayat?
3. Bagaimana *push button* *Ya* dan *Tidak* digunakan sebagai konfirmasi untuk ayat yang dipilih dan didengar?
4. Bagaimana LCD 4x20 digunakan sebagai tampilan informasi untuk ayat Surah Al-Baqarah?
5. Bagaimana DFPlayer Mini, *speaker* dan LM386 dapat mengeluarkan *output* berupa lantunan ayat Surah Al-Baqarah dan dapat menguatkan suara?
6. Bagaimana *buzzer* dan LED dapat digunakan sebagai indikator suara dan status sistem sedang berjalan?

## 1.3 Batasan Masalah

Penelitian ini memiliki batasan masalah agar penelitian tidak menyimpang dari tujuan penelitian, maka penelitian ini difokuskan pada masalah yang berkaitan dengan hal-hal sebagai berikut:

1. Baterai Lithium-Ion dan *switch* hanya digunakan sebagai sumber arus dan menyalakan sistem.

2. *Push button* dapat digunakan *user* untuk memilih mendengar ayat atau menebak ayat.
3. LCD 4x20 digunakan sebagai tampilan *output* layar *display* seperti dapat menampilkan tampilan awal, dengar ayat dan tebak ayat serta hasil jawaban *user*.
4. DFPlayer Mini, *speaker* dan LM386 dapat melantunkan ayat Surah Al-Baqarah serta dapat menguatkan atau merendahkan suara lantunan ayat.
5. *Buzzer* dan LED hanya digunakan sebagai tampilan *output* berupa suara dan cahaya sebagai status sistem sedang berjalan.

#### 1.4 Hipotesa

Menurut KBBI (Kamus Besar Bahasa Indonesia), Hipotesis merupakan sesuatu yang dianggap benar untuk alasan atau pengutaraan pendapat (teori, proposisi, dan sebagainya) meskipun kebenarannya masih harus dibuktikan. Berdasarkan perumusan masalah di atas, penelitian ini mengambil hipotesis sebagai berikut:

1. Diharapkan alat ini dapat berkontribusi dalam pengembangan metode menghafal Al-Qur'an terutama bagi siswa Sekolah Dasar.
2. Diharapkan LCD 4x20 dapat menampilkan tampilan awal, pilihan mendengar ayat dan menebak ayat serta hasil jawaban *user*.
3. Diharapkan DFPlayer Mini, *speaker* dan LM386 dapat melantunkan serta dapat menguatkan atau merendahkan suara lantunan ayat Surah Al-Baqarah.

4. Diharapkan *buzzer* dan LED dapat digunakan sebagai tampilan *output* berupa suara dan cahaya.

### 1.5 Tujuan Penelitian

Bersadarkan rumusan masalah yang telah penulis jelaskan sebelumnya, maka penelitian ini dilakukan untuk mendapatkan tujuan penelitian tentang hal-hal yang dibahas sebagai berikut:

1. Merancang baterai Lithium-Ion dan *switch* dapat digunakan sebagai sumber arus listrik dan menyalurkannya ke seluruh komponen pada sistem.
2. Mendesain rangkaian *push button* dapat digunakan sebagai *input* berupa angka ayat dan memilih dengar ayat atau tebak ayat.
3. Merancang penggunaan LCD 4x20 agar dapat digunakan sebagai tampilan *output* layar *display*.
4. Mendesain rangkaian bagaimana DFPlayer Mini, *speaker* dan LM386 agar dapat mengeluarkan *output* berupa lantunan ayat Surah Al-Baqarah serta dapat mengatur tinggi rendah suara.
5. Merancang *buzzer* dan LED agar dapat digunakan sebagai indikator sistem sedang berjalan.

### 1.6 Manfaat Penelitian

Manfaat penelitian pada Game Tebak Ayat Surah Al-Baqarah Sebagai Media Menghafal Al-Qur'an Bagi Siswa Sekolah Dasar Berbasis Arduino Mega 2560 sebagai berikut:

A. Bagi Penulis

1. Sebagai sarana untuk menerapkan ilmu dan mengembangkan potensi diri dalam menambah pengetahuan terutama di bidang elektronika dan kontroler.
2. Sebagai syarat untuk mendapatkan gelar di jenjang Pendidikan Strata 1 (S1).
3. Dapat mengetahui cara kerja komponen yang digunakan pada game yang diproses mikrokontroler.

B. Bagi Program Studi

1. Penelitian ini dapat menambah referensi untuk mengembangkan pemanfaatan ilmu pengetahuan di bidang komputer dalam permainan game menggunakan Arduino.
2. Penelitian ini dapat menambah *literature* bagi mahasiswa yang berhubungan dengan Arduino.

C. Bagi Masyarakat

1. Dapat membantu siswa Sekolah Dasar untuk menghafal ayat Surah Al-Baqarah.
2. Dapat meningkatkan minat siswa Sekolah Dasar untuk menghafal Al-Qur'an.
3. Menjadi media inovatif belajar menghafal Al-Qur'an bagi siswa Sekolah Dasar.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Konsep Dasar Sistem**

Menurut Kamus Besar Bahasa Indonesia (KBBI), sistem adalah perangkat unsur yang secara teratur saling berkaitan sehingga membentuk suatu totalitas. Definisi lain dari sistem adalah susunan yang teratur dari pandangan, teori, asas, dan sebagainya. Menurut Risah (2024), suatu sistem terdiri dari kumpulan unsur-unsur yang saling berhubungan sebagai suatu kesatuan. Sedangkan menurut Achmad dan Indra (2020), sistem adalah merupakan satu kesatuan data yang terhubung dan terorganisir secara prosedural. Dari pengertian tersebut, definisi sistem adalah sekumpulan komponen *input*, proses, dan komponen *output* yang saling berhubungan sebagai suatu kesatuan sehingga membentuk suatu totalitas serta mencapai suatu tujuan.

##### **2.1.1 Karakteristik Sistem**

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu mempunyai komponen-komponen, batasan sistem, lingkungan luar sistem, penghubung, masukan, keluaran pengolah dan sasaran atau tujuan. Menurut Risah (2024), karakteristik sistem yaitu sebagai berikut:

1. Komponen (*Components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk suatu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem.

2. Batasan sistem (*Boundary*)

Batasan sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan dan menunjukkan ruang lingkup dari sistem tersebut.

3. Lingkungan luar sistem (*Environment*)

Lingkungan luar sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan juga merugikan.

4. Penghubung sistem (*Interface*)

Penghubung sistem merupakan media yang menghubungkan antara suatu subsistem dengan yang lainnya. Melalui penghubung ini kemungkinan sumber-sumber daya mengalir dari suatu subsistem ke subsistem yang lainnya.

5. Masukan sistem (*Input*)

Masukan sistem adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan dan masukan sinyal maintenance input adalah energi yang dimasukkan supaya sistem tersebut dapat berjalan.

#### 6. Keluaran sistem (*Output*)

Keluaran sistem adalah energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran dapat merupakan masukan untuk subsistem yang lain.

#### 7. Pengolahan sistem (*Process*)

Pengolahan sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya. Pengolah yang akan merubah masukan menjadi keluaran.

#### 8. Sasaran sistem (*Objective*)

Suatu sasaran sistem mempunyai tujuan atau sasaran, kalau sistem tidak mempunyai sasaran maka sistem tidak akan ada. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya. Sasaran sangat berpengaruh pada masukan dan keluaran yang dihasilkan.

### 2.1.2 Siklus Pengembangan Sistem

Sistem memiliki siklus pengembangan yang dikenal sebagai Siklus Hidup Pengembangan Sistem atau *System Development Life Cycle* (SDLC). Menurut Nur dan Syahid (2022), SDLC adalah sebuah metode yang digunakan untuk mengembangkan sebuah sistem atau sebuah proses logika yang digunakan oleh seorang *system analist* untuk mengembangkan sebuah sistem informasi. Langkah-langkah siklus hidup pengembangan sistem adalah sebagai berikut.

### 1. *Requirements Analysis and Definition*

Tahapan ini muncul karena adanya permintaan terhadap sistem baru. Tahapan ini dimulai dari mengumpulkan kebutuhan *user*, kemudian menganalisis dan mengidentifikasi kebutuhan yang harus dipenuhi oleh sistem yang akan dibuat. Dalam mengumpulkan analisis kebutuhan, pembuat sistem dapat melakukannya dengan berbagai cara seperti melakukan wawancara, riset, dan membuat prototipenya.

### 2. *System and Software Design*

Membuat desain adalah tahapan membuat sistem dalam bentuk visual sebelum mengubahnya ke dalam bentuk asli. Desain harus dapat mengimplementasikan kebutuhan yang telah disebutkan pada tahapan sebelumnya.

### 3. *Implementation and Unit Testing*

Tahapan ini adalah tahap menguji setiap unit yang ada pada sistem dengan menggunakan bahasa pemrograman yang ditentukan dan mengubahnya menjadi kode-kode yang dimengerti oleh komputer.

### 4. *Integration and System Testing*

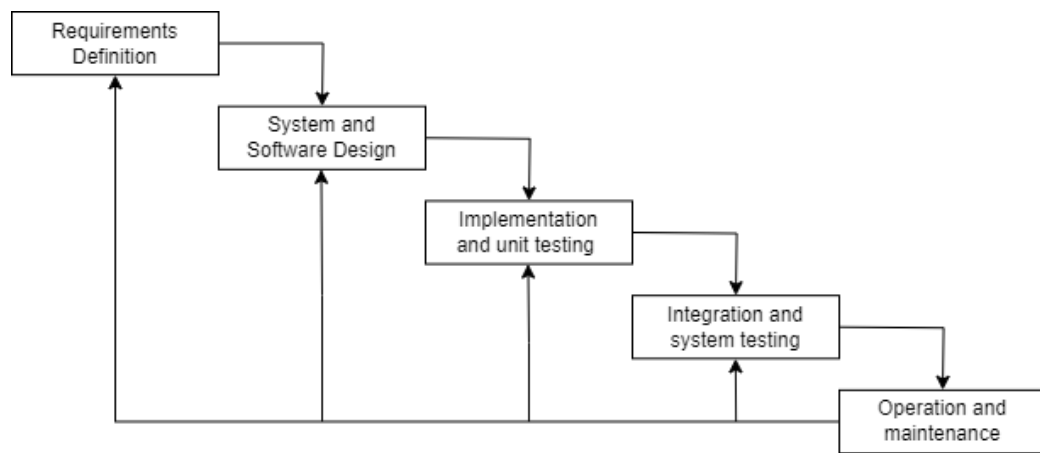
Pada tahapan ini dilakukan penggabungan setiap unit menjadi satu sistem yang utuh dan terintegrasi.

### 5. *Operation and Maintenance*

Tahapan berikutnya adalah pengoperasian sistem, yaitu melakukan uji coba pada sistem dan hasil yang didapatkan haruslah benar sesuai dengan kebutuhan

yang sudah didefinisikan sebelumnya. Pada tahap ini juga terjadi pengembangan atau *maintenance* agar sistem dapat berjalan selamanya.

Berdasarkan penjelasan di atas, tahapan dari siklus hidup pengembangan sistem atau disebut juga model *waterfall* ini dapat dilihat pada gambar 2.1 berikut di bawah ini.



Sumber: Vira Adi Kurniyanti dkk, 2022

**Gambar 2.1 Model Waterfall**

## 2.2 Sistem Kontrol

Menurut Kamus Besar Bahasa Indonesia, kontrol adalah pengawasan, pemeriksaan, atau pengendalian. Sistem kontrol atau *control system* adalah serangkaian komponen *input*, proses, dan komponen *output* yang saling berkaitan untuk mencapai suatu tujuan yaitu mengendalikan sistem itu sendiri. Berdasarkan prosesnya, sistem kontrol terdiri dari dua jenis, yaitu sistem *loop* tertutup dan *loop* terbuka. Pada sub-judul ini, sistem *loop* terbuka dan sistem *loop* tertutup akan dijelaskan sebagai berikut.

### 2.2.1 Sistem Loop Terbuka

Sistem kontrol memiliki karakteristik yaitu dimana nilai *output* yang dihasilkan tidak berpengaruh terhadap aksi kontrol yang disebut sebagai sistem kontrol *loop* terbuka (*Open Loop*). Sistem kontrol *loop* terbuka (*Open Loop*) lebih sederhana, akan tetapi memiliki kekurangan yaitu lebih memiliki tingkat kesalahan yang lebih besar apabila diberikan gangguan dari luar. Secara umum sistem kontrol *loop* terbuka dapat dilihat pada gambar 2.2 berikut:

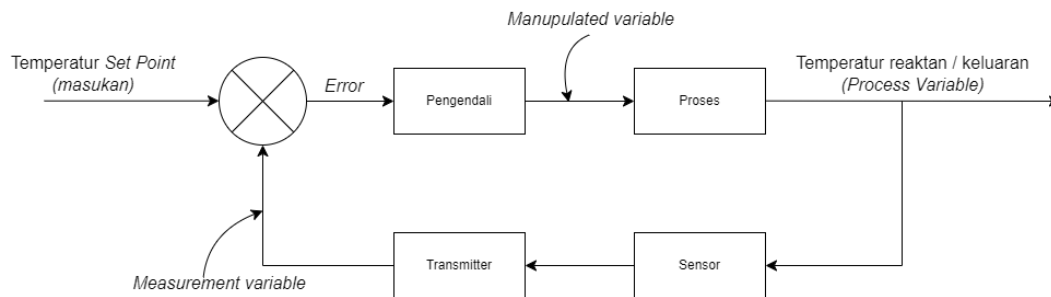


Sumber: Kiagus Ahmad Roni dkk, 2020

**Gambar 2.2 Sistem Loop Terbuka**

### 2.2.2 Sistem Loop Tertutup

Sistem kontrol *loop* tertutup (*Close Loop*) identik dengan sistem kontrol yang memiliki umpan balik (*feedback*), atau dapat dikatakan nilai *output* akan mempengaruhi pada aksi kontrolnya. Sistem kontrol *loop* tertutup (*Close Loop*) dapat dibandingkan dengan sistem kontrol *loop* terbuka, pada sistem *loop* tertutup lebih rumit. Akan tetapi tingkat kesetabilan pada sistem *loop* tertutup lebih relatif konstan serta nilai kesalahan lebih kecil dibandingkan sistem kontrol *loop* terbuka. Secara umum sistem kontrol *loop* tertutup dapat dilihat pada gambar 2.3 berikut.



Sumber: Kiagus Ahmad Roni dkk, 2020

**Gambar 2.3 Sistem Loop Tertutup**

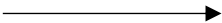
## 2.3 Alat Bantu Perancangan Sistem

Proses penganalisaan terhadap suatu sistem dilakukan dengan aturan dasar yang mendefinisikan secara menyeluruh sistem yang akan dirancang. Hal ini mengandung arti bahwa harus ada gambaran yang jelas mengenai ruang lingkup tentang sistem yang dibahas. Media yang digunakan untuk menggambarkan sistem tersebut adalah *Context Diagram*, *Data Flow Diagram (DFD)*, dan *Flowchart Program*. Pada sub-judul ini, *Context Diagram*, *Data Flow Diagram (DFD)*, dan *Flowchart Program* akan dijelaskan sebagai berikut:

### 2.3.1 Context Diagram

*Context Diagram* merupakan alat pemodelan data yang menggambarkan sistem sebagai suatu jaringan dari fungsi-fungsi atau proses-proses dari sistem yang saling berhubungan satu sama lain dengan aliran data yang digambarkan dengan anak panah. Simbol yang digunakan pada *context diagram* dapat pada tabel 2.1 berikut.

**Tabel 2.1 Context Diagram**

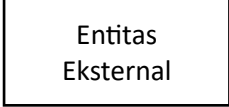
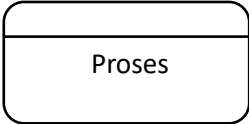
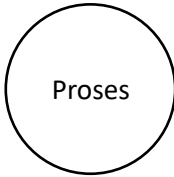
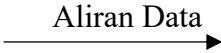
| No. | Simbol                                                                            | Nama        | Keterangan                                                          |
|-----|-----------------------------------------------------------------------------------|-------------|---------------------------------------------------------------------|
| 1.  |  | Terminator  | Menggambarkan asal data dan tujuan                                  |
| 2.  |  | Proses      | Simbol ini digunakan untuk proses pengolahan atau transformasi data |
| 3.  |  | Aliran data | Menggambarkan perpindahan data                                      |

*Sumber: Yanes Robert Yeska Kakihary dkk, 2021*

### 2.3.2 Data Flow Diagram

*Data Diagram Flow* (DFD) adalah suatu model logika sata atau proses yang dibuat untuk menggambarkan dari mana asal data dan ke mana tujuan data keluar dari sistem, di mana data tersimpan, proses apa yang menghasilkan data tersebut, dan interaksi antara data tersimpan dan proses yang dikenakan pada data tersebut. DFD menggambarkan suatu sistem yang menggunakan sejumlah bentuk-bentuk simbol untuk menggambarkan bagaimana arus data melalui proses yang saling berkaitan. DFD hanya terdiri dari empat simbol. Simbol-simbol itu digunakan untuk elemen-elemen lingkungan yang berhubungan dengan sistem, proses, arus data, serta penyimpanan data. Simbol yang digunakan untuk DFD dapat dilihat pada Tabel 2.2 berikut:

**Tabel 2.2 Data Flow Diagram**

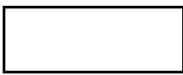
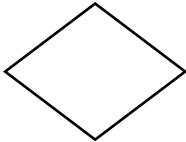
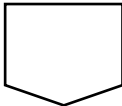
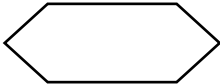
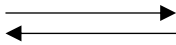
| No. | Gane/Sarson                                                                        | Yourdon/ De Marvo                                                                   | Keterangan                                             |
|-----|------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|--------------------------------------------------------|
| 1.  |   |   | Berupa orang/unit yang berinteraksi dengan sistem.     |
| 2.  |   |   | Orang/unit yang melakukan transformasi pada data.      |
| 3.  |   |   | Arah dari aliran data dari sumber ke tujuan.           |
| 4.  |  |  | Penyimpanan data atau tempat data dilihat dari proses. |

Sumber: Yanes Robert Yeska Kakihary dkk, 2021

### 2.3.3 Flowchart Program

Diagram alir atau *flowchart* dapat diartikan sebagai langkah-langkah penyelesaian masalah yang dituliskan dalam suatu simbol-simbol tertentu. Diagram alir ini akan menunjukkan alur di dalam program secara logika. Untuk menggambarkan langkah atau pemecahan masalah secara sederhana, dapat dimengerti, rapi dan tidak ambigu dengan menggunakan beberapa simbol-simbol yang bisa dibilang standart merupakan tujuan dari *flowchart*. Simbol dan kegunaan suatu algoritma dalam bentuk diagram alir dapat dilihat pada tabel 2.3 berikut:

**Tabel 2.3 Flowchart Program**

| No. | Simbol                                                                              | Nama                      | Fungsi                                                                                  |
|-----|-------------------------------------------------------------------------------------|---------------------------|-----------------------------------------------------------------------------------------|
| 1.  |    | Terminal                  | Awal atau akhir suatu program (prosedur).                                               |
| 2.  |    | <i>Input/Output</i>       | Proses input atau output terlepas dari jenis perangkat.                                 |
| 3.  |    | <i>Process</i>            | Proses operasional komputer.                                                            |
| 4.  |    | <i>Decision</i>           | Untuk menunjukkan bahwa suatu kondisi tertentu mengarah pada dua kemungkinan, ya/tidak. |
| 5.  |   | <i>Connector</i>          | Koneksi penghubung proses ke proses lain pada halaman yang sama.                        |
| 6.  |  | <i>Offline connector</i>  | Koneksi penghubung dari suatu proses ke proses lain di halaman lain.                    |
| 7.  |  | <i>Predefined Process</i> | Mewakili ketentuan penyimpanan untuk diproses untuk memberikan awal harga.              |
| 8.  |  | <i>Flow</i>               | Menyatakan jalannya arus suatu proses.                                                  |

Sumber: Nurhaliza Khesya, 2021

## 2.4 Arduino

Menurut Zulfikar (2020), Arduino merupakan suatu papan elektronik atau minimum sistem, di dalamnya terdapat mikrokontroler sederhana yang dapat

dimanfaatkan untuk mewujudkan jalannya sebuah rangkaian elektronik dari yang sederhana hingga kompleks. Mikrokontroler merupakan keluarga mikroprosesor yaitu sebuah chip yang dapat melakukan pemrosesan data secara digital sesuai dengan perintah Bahasa assembly yang diberikan (Saleilei, Hendri, & Wirawan, 2023). Arduino dirancang untuk memudahkan penggunaan elektronik dalam berbagai bidang yang dirilis oleh Atmel dan memiliki banyak kelebihan dibandingkan dengan mikrokontroler lainnya. Kelebihan diantaranya yaitu memiliki pustaka kode program (*code library*) dan banyak modul yang *support* dengan Arduino. Berikut adalah penjelasan komponen utama dan komponen pendukung.

#### **2.4.1 Komponen Utama**

Arduino sebagai mikrokontroler membutuhkan komponen utama dan komponen pendukung. Pada perancangan sistem ini, komponen-komponen yang digunakan akan dijelaskan sebagai berikut.

##### **2.4.1.1 Arduino Mega 2560**

Arduino Mega 2560 adalah papan Mikrokontroler berbasis Atmega 2560. Arduino Mega 2560 memiliki 54 pin digital *input/output*, dimana 15 pin dapat digunakan sebagai output PWM, 16 pin sebagai *input* analog, dan 4 pin sebagai UART (*port serial hardware*), 16 MHz kristal osilator, koneksi USB, *jack power*. Menurut Johanes dkk (2024), Arduino Mega 2560 dilengkapi dengan *header ICSP (In-Circuit Serial Programming)* yang mampu membuat

pemrograman mikrokontroler dapat mengembangkan *firmware* yang lebih canggih. Kelebihan dari Arduino Mega ialah kapasitas memori yang besar. Selain itu Arduino Mega juga memiliki tombol reset yang dapat membantu pengguna untuk dengan lebih cepat memulai ulang program tanpa perlu memutuskan sambungan daya ke Arduino Mega. Bentuk Arduino Mega 2560 dapat dilihat pada gambar 2.4.



*Sumber: M Nurul Hilal Lubudi, 2020*

**Gambar 2.4 Arduino Mega 2560**

Arduino Mega 2560 berbeda dari papan sebelumnya, karena versi terbaru sudah tidak menggunakan *chip driver* FTDI USB-to-serial. Tapi, menggunakan *chip* Atmega 16U2 (Atmega 8U2 pada papan Revisi 1 dan Revisi 2) yang diprogram sebagai konverter USB-to-serial. Arduino Mega 2560 Revisi 2 memiliki resistor penarik jalur HWB 8U2 ke *Ground*, sehingga lebih mudah untuk dimasukkan ke dalam mode DFU. Tabel spesifikasi Arduino Mega 2560 dapat dilihat pada tabel 2.4 berikut:

**Tabel 2.4 Spesifikasi Arduino Mega 2560**

| No. | Keterangan                   | Nilai (Parameter)                |
|-----|------------------------------|----------------------------------|
| 1.  | Mikrokontroler               | ATmega2560                       |
| 2.  | Tegangan Operasional         | 5V                               |
| 3.  | Tegangan Input (rekomendasi) | 7-12V                            |
| 4.  | Tegangan Input (limit)       | 6-20V                            |
| 5.  | Pin Digital I/O 54           | (of which 15 provide PWM output) |
| 6.  | Pin Analog Input             | 16                               |
| 7.  | Arus DC per Pin I            | 20 mA                            |

|     |                         |                              |
|-----|-------------------------|------------------------------|
| 8.  | Arud DC untuk Pin 3.3 V | 50 mA                        |
| 9.  | Memori Flash            | 250 KB of which 8 KB used by |
| 10. | SRAM                    | 8 KB                         |
| 11. | EEPROM                  | 4 KB                         |
| 12. | Clock Speed             | 16 MHz                       |
| 13. | LED_BUILTIN             | 13                           |
| 14. | Panjang                 | 101.52 mm                    |
| 15. | Lebar                   | 53.3 mm                      |
| 16. | Berat                   | 37 g                         |

*Sumber: Datasheet Arduino Mega 2560*

Pin pada Arduino Mega 2560 mempunyai fungsi yang berbeda sehingga tidak semua pin bisa digunakan dengan fungsi yang sama. Fungsi Arduino Mega 2560 dapat dilihat pada tabel 2.5 berikut:

**Tabel 2.5 Pinmap Arduino Mega 2560**

| No. | Kategori Pin         | Nama Pin            | Fungsi                                                                                |
|-----|----------------------|---------------------|---------------------------------------------------------------------------------------|
| 1.  | Input/Output Digital | 0 – 53              | Menerima atau mengirim sinyal digital                                                 |
| 2.  | Input Analog         | A0 – A15            | Mengubah sinyal analog menjadi sinyal digital                                         |
| 3.  | Serial 0             | 0 (RX) dan 1 (TX)   | Pin RX digunakan untuk menerima data serial dan pin TX untuk mengirim data serial TTL |
| 4.  | Serial 1             | 19 (RX) dan 18 (TX) |                                                                                       |
| 5.  | Serial 2             | 17 (RX) dan 16 (TX) |                                                                                       |
| 6.  | Serial 3             | 15 (RX) dan 14 (TX) |                                                                                       |
| 7.  | Interupsi Eksternal  | 2 (Interrupt 0)     | Memicu interupsi pada nilai yang rendah, meningkat, menurun, atau perubahan nilai     |
|     |                      | 3 (Interrupt 1)     |                                                                                       |
|     |                      | 21 (Interrupt 2)    |                                                                                       |
|     |                      | 20 (Interrupt 3)    |                                                                                       |
|     |                      | 19 (Interrupt 4)    |                                                                                       |
|     |                      | 18 (Interrupt 5)    |                                                                                       |
| 8.  | PWM                  | 2 – 13 dan 44 – 46  | Menghasilkan sinyal analog dari perangkat digital                                     |
| 9.  | SPI                  | 50 (MISO)           | Sebagai komunikasi SPI                                                                |
|     |                      | 51 (MOSI)           |                                                                                       |
|     |                      | 52 (SCK)            |                                                                                       |
|     |                      | 53 (SS)             |                                                                                       |
| 10. | I2C                  | 20 (SDA)            | Sebagai komunikasi I2C atau TWI                                                       |
|     |                      | 21 (SCL)            |                                                                                       |

|     |         |       |                                                      |
|-----|---------|-------|------------------------------------------------------|
| 11. | LED     | 13    | Menyalakan LED bawaan papan yang terhubung di pin 13 |
| 12. | Daya    | VIN   | Sebagai pin masukan daya eksternal                   |
|     |         | 5 V   | Sebagai pin keluaran daya 5 volt                     |
|     |         | 3,3 V | Sebagai pin keluaran daya 3,3 volt                   |
|     |         | GND   | Meniadakan beda potensial                            |
| 13. | Lainnya | AREF  | Tegangan referensi untuk input analog                |
|     |         | Reset | Untuk menambahkan tombol reset                       |

Sumber: Datasheet Arduino Mega 2560

#### 2.4.1.2 Push Button

Menurut Hervan (2023), *push button* adalah saklar tekan yang berfungsi untuk menghubungkan atau memisahkan bagian-bagian dari suatu instalasi listrik maupun peralatan (suatu sistem saklar tekan *push button* terdiri dari saklar tekan start. Stop reset dan saklar tekan untuk *emergency*. *Push button* memiliki kontak NC (*normally close*) dan NO (*normally open*). Bentuk fisik *push button* dapat dilihat pada gambar 2.5 berikut.

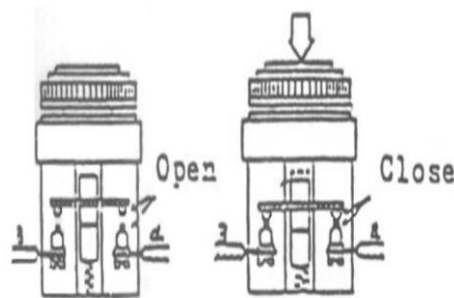


Sumber: Yul Antonisfia dkk, 2021

**Gambar 2.5 Push Button**

Saklar tekan (*push button*) menurut kedudukan kontak-kontaknya saklar tekan dibagi menjadi dua, yaitu saklar tekan NO dan NC:

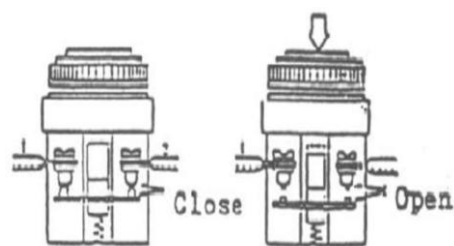
- a. *Push button* tipe NO atau *normally open* adalah saklar tekan yang apabila dalam keadaan normal maka kontakannya terbuka, dan bila ditekan maka akan tertutup, tetapi kontak akan kembali dalam keadaan normal jika tekanan dilepas.



Sumber: Hervan Fernando Sitorus, 2023

**Gambar 2.6 Normally Open**

- b. *Push button* tipe NC atau *normally closed* adalah saklar tekan yang apabila dalam keadaan normal maka kontakannya akan tertutup, dan bila ditekan maka akan terbuka, tetapi kontak akan kembali dalam keadaan normal jika tekanan dilepas.



Sumber: Hervan Fernando Sitorus, 2023

**Gambar 2.7 Normally Closed**

### 2.4.1.3 Baterai Lithium-Ion

Baterai Li-ion atau Lithium-Ion merupakan salah satu jenis baterai yang sekarang banyak digunakan karena baterai Lithium-Ion bila dibandingkan dengan baterai lainnya seperti Lead-Acid, NiCd dan Ni-MH, baterai Lithium-Ion memiliki fitur energi dan kerapatan daya yang tinggi, durasi hidup yang tahan lama dan ramah lingkungan, dan juga sudah banyak diaplikasikan pada peralatan elektronika yang beredar di pasaran. Baterai Lithium-Ion harus dioperasikan pada area aman dan handal, yang dimana akan berefek pada *charge rate*, suhu, dan rentang tegangan. Jika melebihi dari rentang tersebut akan mengarah pada melemahnya kinerja baterai dan akan menghasilkan masalah pada keamanan seperti terjadinya ledakan pada baterai. Untuk memastikan operasi yang handal dari baterai Lithium-Ion dan memprediksikan usia baterai yang tersisa selama masa pemakaian. Bentuk fisik baterai Lithium-Ion seperti yang ditunjukkan pada gambar 2.8 berikut.



Sumber: M Nurul Hilal Lubudi, 2020

**Gambar 2.8 Baterai Lithium-Ion**

### 2.4.1.4 Switch

*Switch* yang digunakan pada penelitian ini adalah *rocker switch*. *Rocker switch* atau sakelar *rocker* adalah perantara untuk memutuskan atau menyambungkan tegangan pada sebuah rangkaian, sehingga rangkaian akan mulai bekerja bila sakelar dalam posisi ON, sedangkan ketika OFF maka rangkaian akan

berhenti berfungsi. *Rocker Switch* ini merupakan sakelar DPST (*Double Pole Single Throw Switch*), merupakan sakelar yang terdiri dari dua kutub dengan satu arah. Jadi hanya dapat menghubungkan dan memutus saja. Untuk jenis konstruksi putar jenis sakelar ini banyak dijumpai pada kotak sekering instalasi rumah (panel hubung bagi yang paling sederhana). Bentuk fisik *rocker switch* dapat dilihat pada gambar 2.9 berikut.



*Sumber: Muhammad Fajar Ikanov, 2021*

**Gambar 2.9 Switch**

#### **2.4.1.5 LCD 20x4**

LCD atau *Liquid Crystal Display* adalah perangkat yang berfungsi sebagai media penampil dengan memanfaatkan kristal cair sebagai objek penampil utama. Menurut Slamet dan Fajar (2022), LCD merupakan salah satu jenis *display* elektronik yang di buat dengan teknologi *complementary metal oxide semiconductor* (CMOS) *logic* yang bekerja mentransmisikan cahaya dari *back-lit*. LCD berfungsi sebagai penampil data baik dalam bentuk karakter huruf angka ataupun grafik. Menurut I Made Agus Rianta (2022), LCD terdiri dari dua bagian utama yaitu *backlight* (lampu latar belakang) dan bagian *liquid crystal*. Cahaya *backlight* tersebut pada umumnya adalah berwarna putih.

Sedangkan kristal cair (*liquid crystal*) sendiri adalah cairan organik yang berada diantara dua lembar kaca yang memiliki permukaan. Peneliti ini menggunakan LCD I2C. I2C LCD adalah modul LCD yang dikendalikan secara serial sinkron dengan protokol I2C/IIC (*Inter Integrated Circuit*) atau TWI (*Two Wire Interface*). Modul LCD pada normalnya dikendalikan secara paralel baik untuk jalur data maupun kontrolnya. Namun jalur paralel akan memakan banyak pin di sisi kontroller (misal Arduino, komputer, dan lain-lain). Setidaknya akan membutuhkan 6 atau 7 pin untuk mengendalikan sebuah modul LCD.

Dengan demikian untuk sebuah *controller* yang harus mengendalikan banyak I/O, menggunakan jalur paralel adalah solusi yang kurang tepat. Modul I2C converter ini menggunakan *chip* ICPCF8574 produk dari NXP sebagai *controller*-nya. IC ini adalah sebuah 8 bit I/O *expander for* I2C bus yang pada dasarnya adalah sebuah *shift* register. Adapun bentuk fisik dari LCD 20x4 pada gambar 2.10 berikut.



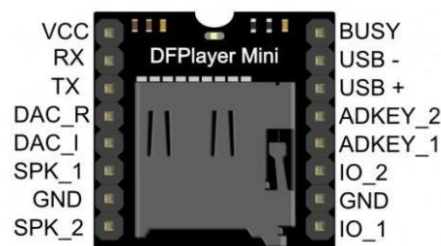
*Sumber: Kureve dkk, 2021*

**Gambar 2.10 LCD 20x4**

#### **2.4.1.6 DFPlayer Mini**

DFPlayer Mini adalah modul MP3 yang *output*-nya sederhana, dapat langsung diaplikasikan pada pengeras suara speaker. DFPlayer mini dapat digunakan dengan cara berdiri tunggal menggunakan baterai, *speaker*, dan *push*

*button*, juga dapat digunakan pada Arduino Uno ataupun dengan perangkat lain yang memiliki kemampuan *receiver/transmitter*. DFPlayer Mini adalah sebuah modul *microSD* dimana modul tersebut dapat mengases dan menyimpan data pada *microSD*. Modul DFPlayer Mini adalah sebuah modul MP3 serial yang menyiakan kesempurnaan integrasi MP3, WMV *hardware decoding*. Sedangkan *software*-nya mendukung *driver TF card*, mendukung sistem *file* FAT16, FAT32. Melalui perintah-perintah serial sederhana untuk menentukan memutar musik, serta bagaimana cara memutar musik dan fungsi lainnya, tidak melalui operasi yang rumit, mudah digunakan, stabil dan dapat diandalkan adalah fitur-fitur yang paling penting dari modul ini. Modul ini dapat digunakan sebagai modul yang berdiri sendiri dengan menggunakan baterai. Untuk memperjelas keterangan diatas gambar DFPlayer Mini dapat dilihat gambar pada gambar 2.11 berikut.



*Sumber: Danang Sindhu Prasetyo, 2022*

**Gambar 2.11 DFPlayer Mini**

#### **2.4.1.7 Speaker**

*Speaker* merupakan sebuah perangkat keras *output* yang berfungsi untuk mengeluarkan hasil dari proses audio maupun suara. *Speaker* juga bisa disebut sebagai alat bantu untuk mengeluarkan suara yang lebih maksimal pada perangkat

musik maupun lainnya. *Speaker* ini bentuknya sangat beragam, selain bentuk fitur maupun ukurannya juga bisa disesuaikan dengan kebutuhan serta keinginan. *Speaker* yang digunakan dalam penelitian ini adalah *speaker* 5W. Bentuk fisik *speaker* 5W dapat dilihat pada gambar 2.12 berikut.

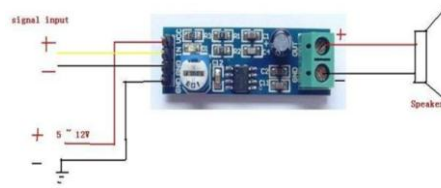


Sumber: Danang Sindhu Prasetyo, 2022

**Gambar 2.12 Speaker**

#### **2.4.1.8 LM386**

Modul amplifier LM386 atau disebut LM386 digunakan sebagai penguat yang dapat beroperasi dalam tegangan rendah, kelebihan modul amplifier LM386 lainnya yaitu gain telah diatur ke 20 internal, dan kenaikan lebih besar dapat dilakukan secara eksternal. Sinyal yang berhasil ditransmisikan melalui gelombang ultrasonik, amplitudo sinyal tersebut menjadi bernilai kecil dikarenakan terdapat loss dan redaman akibat proses pentransmisian sinyal. Begitu pula dengan kondisi derau yang terdapat pada sinyal semakin bertambah akibat filter yang tidak bekerja secara baik. Oleh karena itu digunakan rangkaian penguat untuk membuat amplitudo kembali meningkat dan menghilangkan derau yang terkandung dalam sinyal. Bentuk fisik modul LM386 dapat dilihat pada gambar 2.13 berikut.

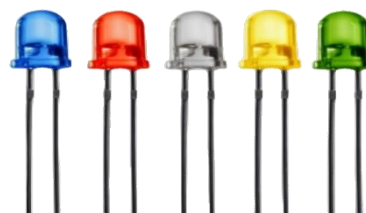


*Sumber: M. Imran Hossain dkk, 2021*

**Gambar 2.13 Modul LM386**

#### 2.4.1.9 LED

Menurut Agung (2020), LED atau *Light Emitting Diode* adalah komponen elektronika yang bisa memancarkan cahaya monokromatik ketika diberikan suatu tegangan maju. LED masih termasuk dalam keluarga dioda yang terdiri dari sebuah *chip* dari bahan semikonduktor yang diisi penuh dengan ketidakmurnian untuk menciptakan sebuah struktur. Warna Cahaya yang dipancarkan LED tergantung dari jenis bahan semikonduktor yang digunakannya. LED juga mampu memancarkan sebuah sinar *inframerah* yang tidak dapat dilihat oleh mata. Bentuk fisik LED dapat dilihat pada gambar 2.14 berikut.



*Sumber: Agung Rizki Wiguna, 2020*

**Gambar 2.14 LED**

#### 2.4.1.10 Buzzer

*Buzzer* adalah sebuah komponen elektronika yang berfungsi untuk mengubah getaran listrik menjadi getaran suara. Pada dasarnya prinsip kerja *buzzer* hampir sama dengan *loud speaker*, jadi *buzzer* juga terdiri dari kumparan yang terpasang pada diafragma dan kemudian kumparan tersebut dialiri arus sehingga menjadi elektromagnet, kumparan tadi akan tertarik ke dalam atau keluar, tergantung dari arah arus dan polaritas magnetnya, karena kumparan dipasang pada diafragma maka setiap gerakan kumparan akan menggerakkan diafragma secara bolak-balik sehingga membuat udara bergetar yang akan menghasilkan suara. *Buzzer* biasa digunakan sebagai indikator bahwa proses telah selesai atau terjadi suatu kesalahan pada sebuah alat (alarm). Bentuk fisik *buzzer* dapat dilihat pada gambar 2.15 berikut.



*Sumber: Handri Al Fani dkk, 2020*

**Gambar 2.15 Buzzer**

#### 2.4.2 Komponen Pendukung

Selain komponen utama, terdapat komponen pendukung sebagai pelengkap sistem agar berjalan lebih maksimal. Komponen pendukung yang digunakan akan dijelaskan sebagai berikut.

#### 2.4.2.1 Stepdown LM2596

Modul *stepdown* LM2596 adalah modul penurun tegangan yang outputnya dapat diatur melalui *multiturn* potensiometer. Keunggulan modul *stepdown* LM2596 adalah besar tegangan *output* tidak berubah (stabil) walaupun tegangan *input* naik turun. *Stepdown* LM2596 DC-DC merupakan konverter penurun tegangan yang mengkonversikan tegangan masukan DC menjadi tegangan DC. Modul ini sangat efisien dan sering digunakan dalam berbagai aplikasi elektronik untuk memastikan komponen menerima tegangan yang tepat. Bentuk fisik modul *stepdown* LM2596 dapat dilihat pada gambar 2.16 berikut.



*Sumber: M Nurul Hilal Lubudi, 2020*

**Gambar 2.16 Stepdown LM2596**

#### 2.4.2.2 Kartu Memori

Kartu memori adalah perangkat penyimpanan praktis sebuah file, pada penelitian ini kartu memori digunakan sebagai tempat menyimpan file .mp3, yang nantinya dioperasikan oleh modul MP3. Dengan kapasitas penyimpanan yang besar dan ukuran yang kecil, kartu memori sangat cocok untuk keperluan penelitian ini. Selain itu, kecepatan akses yang tinggi memastikan file .mp3 dapat diputar tanpa hambatan, menjadikannya solusi efisien untuk berbagai aplikasi audio. Gambar kartu memori dapat dilihat pada gambar 2.17 berikut.



*Sumber: Danang Sindhu Prasetyo, 2022*

**Gambar 2.17 Kartu memori**

#### **2.4.2.3 Resistor**

Resistor adalah komponen dasar elektronika yang digunakan untuk membatasi jumlah arus yang mengalir dalam suatu rangkaian. Sesuai namanya resistor bersifat resistif dan umumnya terbuat dari bahan karbon. Dari hukum Ohm diketahui, resistansi berbanding terbalik dengan jumlah arus yang mengalir melaluinya, satuan resistansi dari suatu resistor disebut Ohm atau dilambangkan dengan simbol *omega*. Suatu resistor dikatakan memiliki hambatan satu Ohm apabila resistor tersebut menjembatani beda tegangan sebesar satu volt dan arus listrik yang timbul akibat tegangan tersebut adalah sebanyak satu Ampere, atau sama dengan sebanyak  $6.241506 \times 10^{18}$  elektron per detik mengalir menghadap arah berlawanan dari arus. Bentuk resistor dapat dilihat pada gambar 2.18 berikut.



*Sumber: Endang dkk, 2024*

**Gambar 2.18 Resistor**

Pada badan resistor terdapat lingkaran membentuk gelang kode warna yang digunakan untuk memudahkan pemakai dalam mengenali besar resistansi resistor tersebut. Warna gelang untuk nilai resistor dapat dilihat pada tabel 2.6 berikut.

**Tabel 2.6 Warna Gelang Resistor**

| No. | Kode Warna | Pita ke-1 | Pita ke-2 | Pita ke-3 | Pita ke-4 | Pita ke-5 |
|-----|------------|-----------|-----------|-----------|-----------|-----------|
| 1.  | Hitam      | 0         | 0         | 0         | $10^0$    | -         |
| 2.  | Coklat     | 1         | 1         | 1         | $10^1$    | 1%        |
| 3.  | Merah      | 2         | 2         | 2         | $10^2$    | 2%        |
| 4.  | Orange     | 3         | 3         | 3         | $10^3$    | -         |
| 5.  | Kuning     | 4         | 4         | 4         | $10^4$    | -         |
| 6.  | Hijau      | 5         | 5         | 5         | $10^5$    | 0,5%      |
| 7.  | Biru       | 6         | 6         | 6         | $10^6$    | 0,25%     |
| 8.  | Ungu       | 7         | 7         | 7         | $10^7$    | 0,1%      |
| 9.  | Abu-abu    | 8         | 8         | 8         | -         | -         |
| 10. | Putih      | 9         | 9         | 9         | -         | -         |
| 11. | Emas       | -         | -         | -         | $10^{-1}$ | 5%        |
| 12. | Perak      | -         | -         | -         | $10^{-2}$ | 10%       |

*Sumber: Zulfikar A. Salam, 2020*

#### 2.4.2.4 Potensiometer

Menurut Bayu dkk. (2024), potensiometer adalah salah satu jenis resistor yang nilai resistansinya dapat diatur sesuai dengan kebutuhan rangkaian elektronika ataupun kebutuhan pemakainya. Potensiometer terdiri dari elemen resistif berbentuk pita atau kumparan, serta terminal yang terhubung ke pita atau kumparan tersebut. Terminal tengah pada potensiometer biasanya digunakan sebagai titik tengah yang dapat digerakkan untuk mengubah nilai resistansi. Nilai resistansi pada potensiometer sendiri terdapat beberapa macam nilai resistansi, seperti 1K, 10K,

100K, dan lain-lain. Potensiometer ini nilai resistansinya yang dapat diatur sesuai dengan kebutuhan rangkaian elektronika atau kebutuhan pemakaiannya.

Pada penelitian ini, peneliti menggunakan potensiometer 10k yang memiliki resistansi 10v. Menurut Andi (2020), potensiometer tidak perlu menggunakan metode pembagi tegangan karena secara intrinsik potensiometer sudah melakukan pembagian tegangan pada bagian *wiper*, dibuktikan dengan adanya 3 pin, di mana pin yang berada di tengah menjadi titik referensi sebagai tegangan keluaran dari sensor, dan kedua pin lainnya sebagai sumber tegangan dan *ground* dan terdapat resistor di antara keduanya. Potensiometer yang digunakan adalah potensiometer *rotary* yang diatur dengan cara memutar *wiper*-nya menggunakan ibu jari. Menurut Isra' dan Niken (2022), potensiometer merupakan jenis variabel resistor. Pada potensiometer terdiri dari 3 kaki yang terhubung pada poros. Bentuk fisik potensiometer dapat dilihat pada gambar 2.19 berikut.



*Sumber: Andi Falih, 2020*

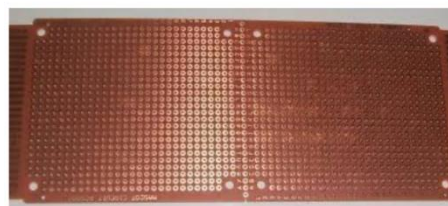
**Gambar 2.19 Potensiometer**

#### **2.4.2.5 PCB**

PCB merupakan singkatan dari *Printed Circuit Board*, yang jika dalam bahasa Indonesia banyak disebut dengan istilah papan sirkuit cetak atau papan rangkaian cetak. PCB ini secara fisik merupakan alat yang digunakan untuk

menghubungkan komponen elektronik dalam komputer dengan lapisan jalur konduktornya. Gambar PCB dapat dilihat pada gambar 2.20. Berikut beberapa fungsi PCB yaitu:

1. Tempat menyusun komponen-komponen elektronik sehingga terpasang lebih rapi dan terorganisir.
2. Menghubungkan kaki komponen satu sama lain baik kaki komponen aktif maupun pasif.
3. Pengganti kabel untuk menyambung berbagai komponen, sehingga membutuhkan tempat yang lebih efisien.
4. Membuat tampilan suatu rangkaian elektronik menjadi lebih rapi dan tertata.



*Sumber: Elfitra Desifatma, 2020*

**Gambar 2.20 PCB**

#### **2.4.2.6 Kabel Jumper**

Kabel *jumper* atau *jumper wire* adalah kabel yang biasanya digunakan untuk penghubung antara perangkat sensor ataupun *breadbord* dengan mikrokontroler dan media transmisi penghantar listrik maupun sinyal-sinyal dari sensor, kemudian diterjemahkan oleh mikrokontroler itu sendiri. Secara umum *jumper wire* terdiri dari 3 jenis, yaitu: *male – male*, *male – female* dan *female – female*. Bentuk fisik kabel jumper dapat dilihat pada gambar 2.21 berikut.



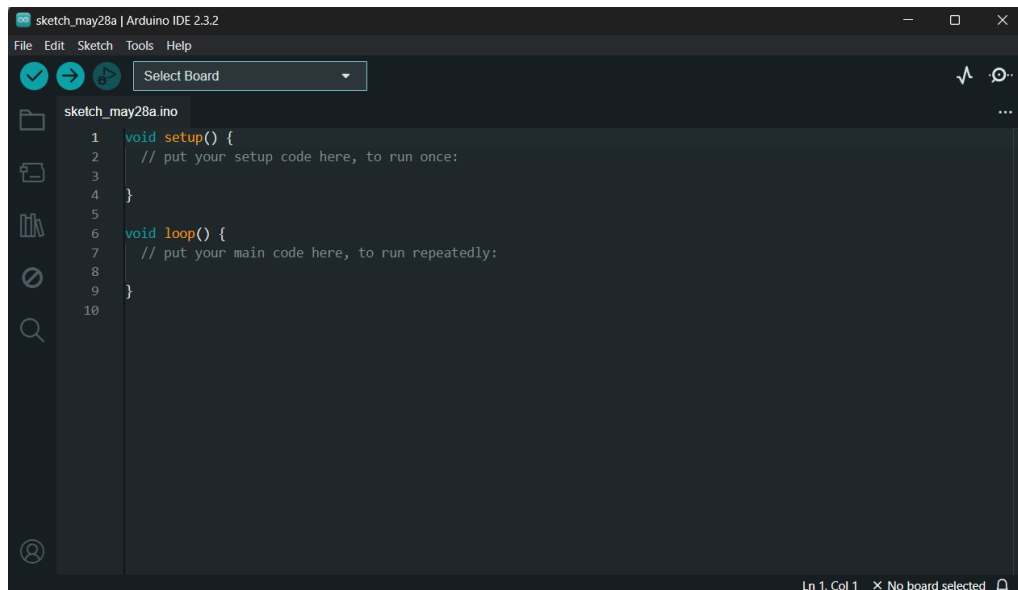
*Sumber: Muhammad Yogi Fariska, 2021*

**Gambar 2.21 Kabel Jumper**

## 2.5 Arduino IDE

Menurut Johaness dkk (2024), Arduino menggunakan Arduino *software* berupa IDE. IDE adalah akronim dari *Integrated Development Environment* dan adalah esensi atau dasar dari pengembangan perangkat lunak, tidak terkecuali dengan Arduino. IDE berperan sebagai lingkungan terintegrasi yang memungkinkan para *developer* untuk menulis, menguji, dan men-*debug* dalam satu lingkungan terpadu. Lingkungan atau *environment* ini dalam konteks Arduino ialah Arduino IDE. Arduino IDE dibangun menggunakan bahasa pemrograman *Wiring* yang merupakan penggunaan bahasa pemrograman Java dengan memanfaatkan perpustakaan (*library*) C/C++.

Pada tanggal 20 Februari 2024, bersumber dari website resminya, Arduino merilis versi terbaru Arduino IDE yaitu versi 2.3.2. Arduino IDE versi 2.3.2 lebih cepat dan lebih bertenaga. Selain itu, keunggulan versi 2.3.2 ini lebih modern dengan antarmuka yang lebih responsif, juga memiliki fitur pelengkapan otomatis, navigasi kode, dan bahkan *debugger* langsung. Arduino IDE memiliki empat tema yang berbeda, yaitu *Light*, *Dark*, *Light High Contrast*, *Dark High Contrast*. Tampilan *dark* Arduino IDE 2.3.2 dapat dilihat pada gambar 2.22 berikut:



*Sumber: Arduino Software*

**Gambar 2.22 Arduino IDE**

## 2.6 Pemrograman Arduino

Bahasa Pemrograman Arduino adalah bahasa pemrograman utama yang digunakan untuk membuat program untuk Arduino *board*. Bahasa pemrograman Arduino menggunakan bahasa pemrograman C sebagai dasarnya. Akar bahasa C adalah bahasa BCPL yang dikembangkan oleh Martin Richards pada tahun 1967. Karena menggunakan bahasa pemrograman C sebagai dasarnya, bahasa pemrograman Arduino memiliki banyak sekali kemiripan, walaupun beberapa hal telah berubah. Sistem arduino merupakan sebuah sistem yang *open source* baik secara *hardware* maupun *software*.

Perkembangan sistem *software* arduino disesuaikan dengan perkembangan *hardware*-nya. Program Arduino dapat dibagi menjadi tiga bagian

utama: Struktur, Nilai (variable dan konstanta), dan Fungsi. Pada sub-judul ini, dasar pemrograman Arduino akan dijelaskan sebagai berikut.

### 2.6.1 Struktur

Dalam proyek Arduino, sebuah program disebut sebagai “*sketch*”. *Sketch* adalah tempat menulis program dengan memiliki ekstensi .ino. Menurut Eko (2022) program Arduino memiliki dua fungsi utama:

1. Fungsi *Setup()*

Fungsi *setup()* hanya dipanggil satu kali saja saat program mulai berjalan. Fungsi *setup()* berfungsi untuk melakukan inisialisasi mode pin atau untuk memulai komunikasi serial. Fungsi *setup()* ini harus ada meskipun tidak ada program yang akan dieksekusi.

2. Fungsi *Loop()*

Setelah menyiapkan inisialisasi pada *setup()*, berikutnya membuat fungsi *loop()*. Sesuai dengan namanya, fungsi akan mengulang program secara terus menerus, sehingga program akan berubah dan merespon sesuai *input* yang diberikan. Fungsi *loop()* ini akan secara aktif mengontrol *board* Arduino.

### 2.6.2 Tipe Data

Pada umumnya, tipe data dibedakan menjadi dua yaitu variabel dan konstanta. Variabel berfungsi untuk menampung nilai angka dan memberikan nama sesuai dengan kebutuhan membuat program. Dengan menggunakan variabel, maka

nilai yang ada dapat diubah. Berikut ini tipe data yang dapat digunakan dalam program Arduino, ditunjukkan dalam tabel 2.7 berikut.

**Tabel 2.7 Tipe Data Arduino**

| No. | Tipe Data            | Ukuran (bit) | Jangkauan Nilai                                            |
|-----|----------------------|--------------|------------------------------------------------------------|
| 1.  | <i>boolean</i>       | 1            | 0, 1 (tipe data bit hanya digunakan untuk variable global) |
| 2.  | <i>char</i>          | 8            | -128 to 127                                                |
| 3.  | <i>unsigned char</i> | 8            | 0 to 255                                                   |
| 4.  | <i>byte</i>          | 8            | 0 to 255                                                   |
| 5.  | <i>int</i>           | 16           | -32768 to 32767                                            |
| 6.  | <i>unsigned int</i>  | 16           | 0 to 65535                                                 |
| 7.  | <i>word</i>          | 16           | 0 to 65535                                                 |
| 8.  | <i>long</i>          | 32           | -2147483648 to 2147483647                                  |
| 9.  | <i>unsigned long</i> | 32           | 0 to 4294967295                                            |
| 10. | <i>short</i>         | 16           | -32768 to 32767                                            |
| 11. | <i>float</i>         | 32           | $\pm 1,175e-38$ to $\pm 3,402e38$                          |
| 12. | <i>double</i>        | 32           | $\pm 1,175e-38$ to $\pm 3,402e38$                          |

Sumber: Eko Mardianto, 2022

### 2.6.3 Operator

Pada Arduino, operator merupakan simbol yang biasa dilibatkan dalam program untuk melakukan suatu operasi yang terdiri dari dua jenis yaitu operator aritmatika dan operator pembandingan. Operator aritmatika digunakan untuk operasi matematika. Operator pembandingan digunakan untuk membandingkan dua nilai. Operator logika digunakan untuk membentuk suatu logika atas dua buah kondisi atau lebih.

### 2.6.3.1 Operator Aritmatika

Operator aritmatika merupakan operator yang digunakan untuk memanipulasi angka seperti penambahan, pengurangan, perkalian dan pembagian. Operator aritmatika yang terdapat pada Arduino dapat dilihat pada tabel 2.8 berikut:

**Tabel 2.8 Operator Aritmatika**

| No. | Operator Aritmatika | Keterangan                       |
|-----|---------------------|----------------------------------|
| 1.  | +                   | Operasi penjumlahan              |
| 2.  | -                   | Operasi pengurangan              |
| 3.  | *                   | Operasi perkalian                |
| 4.  | /                   | Operasi pembagian                |
| 5.  | %                   | Operasi sisa pembagian (modulus) |

*Sumber: Eko Mardianto, 2022*

### 2.6.3.2 Operator Pembandingan

Operator pembandingan merupakan operator yang digunakan untuk operasi logika perbandingan antara dua nilai atau antara dua variabel. Operator pembandingan yang terdapat pada Arduino dapat dilihat pada tabel 2.9 berikut.

**Tabel 2.9 Operator Pembandingan**

| No. | Operator Kondisi | Contoh   | Keterangan                                                             |
|-----|------------------|----------|------------------------------------------------------------------------|
| 1.  | ==               | $x == y$ | Bernilai benar bila kedua data sama dan bernilai salah bila sebaliknya |
| 2.  | !=               | $x != y$ | Tidak sama dengan                                                      |
| 3.  | >                | $x > y$  | Lebih besar dari                                                       |
| 4.  | <                | $x < y$  | Lebih kecil dari                                                       |
| 5.  | >=               | $x >= y$ | Lebih besar atau sama dengan                                           |
| 6.  | <=               | $x <= y$ | Lebih kecil atau sama dengan                                           |

*Sumber: Eko Mardianto, 2022*

### 2.6.3.3 Operator Logika

Operator logika merupakan operator yang digunakan untuk membentuk suatu logika atas dua buah kondisi atau lebih. Operator logika yang terdapat pada Arduino dapat dilihat pada tabel 2.10 berikut:

**Tabel 2. 10 Operator Logika**

| No. | Operator Logika | Keterangan                       |
|-----|-----------------|----------------------------------|
| 1.  | &&              | Operator untuk logika <i>AND</i> |
| 2.  |                 | Operator untuk logika <i>OR</i>  |
| 3.  | !               | Operator untuk logika <i>NOT</i> |

Sumber: Eko Mardianto, 2022

### 2.6.3.4 Operator Bitwise

Operator *bitwise* merupakan operasi khusus untuk digunakan dalam operasi logika bilangan biner atau hanya bekerja pada operasi logika 0 dan 1 saja.

Operator bitwise dapat dilihat pada tabel 2.11 berikut:

**Tabel 2.11 Operator Bitwise**

| No. | Nama Operator               | Operator Sederhana | Deskripsi                                                                                                      | Contoh                      |
|-----|-----------------------------|--------------------|----------------------------------------------------------------------------------------------------------------|-----------------------------|
| 1.  | <i>Increment</i>            | ++                 | Operator <i>increment</i> , menaikkan nilai integer sebanyak satu langkah                                      | a++                         |
| 2.  | <i>Decrement</i>            | --                 | Operator <i>decrement</i> , menurunkan nilai integer sebanyak satu langkah                                     | x--                         |
| 3.  | <i>Compound Addition</i>    | +=                 | Menambahkan <i>operand</i> sisi kanan dengan <i>operand</i> sisi kiri dan hasilnya ke <i>operand</i> sisi kiri | B+=A ekuivalen dengan B=B+A |
| 4.  | <i>Compound Subtraction</i> | -=                 | Mengurangi <i>operand</i> sisi kanan dengan <i>operand</i> sisi kiri dan                                       | B-=A ekuivalen dengan B=B-A |

|    |                              |         |                                                                                                            |                                          |
|----|------------------------------|---------|------------------------------------------------------------------------------------------------------------|------------------------------------------|
|    |                              |         | hasilnya ke <i>operand</i> sisi kiri                                                                       |                                          |
| 5. | <i>Compound Addition</i>     | $\ast=$ | Mengali <i>operand</i> sisi kanan dengan <i>operand</i> sisi kiri dan hasilnya ke <i>operand</i> sisi kiri | $B\ast=A$ ekuivalen dengan $B=B\ast A$   |
| 6. | <i>Compound Substraction</i> | $/=$    | Membagi <i>operand</i> sisi kanan dengan <i>operand</i> sisi kiri dan hasilnya ke <i>operand</i> sisi kiri | $B/=A$ ekuivalen dengan $B=B/A$          |
| 7. | <i>Compound Modulo</i>       | $\% =$  | Menetapkan bentuk modulus menggunakan dua <i>operand</i> dan menetapkan hasilnya ke <i>operand</i> kiri    | $B \% = A$ ekuivalen dengan $B = B \% A$ |
| 8. | <i>Compound Bitwise And</i>  | $\&=$   | <i>Bitwise</i> dan operator penugasan                                                                      | $A \& = 2$ ekuivalen dengan $A = A \& 2$ |
| 9. | <i>Compound Bitwise Or</i>   | $ =$    | <i>Bitwise inclusive OR</i> dan operator penugasan                                                         | $A  = 2$ ekuivalen dengan $A = A   2$    |

Sumber: Eko Mardianto, 2022

#### 2.6.4 Kondisi

Struktur kondisi biasa digunakan untuk menyeleksi kondisi yang akan dieksekusi selanjutnya oleh program berdasarkan kondisi yang terpenuhi. Struktur ini umumnya dipakai untuk memutuskan satu pilihan pada program dari banyaknya pilihan kondisi. Beberapa struktur kondisi dijelaskan sebagai berikut.

#### 2.6.4.1 Kondisi If

Struktur kondisi *if* digunakan untuk operasi bersyarat pada suatu kondisi Tunggal. Bila proses yang diseleksi terpenuhi atau bernilai benar maka pernyataan yang ada di dalam blok *if* akan diproses dan dikerjakan. Bentuk umum struktur kondisi *if* adalah sebagai berikut:

```
if (kondisi > 50) {  
    // pernyataan yang ada di dalam blok  
}
```

#### 2.6.4.2 Kondisi If-Else

Dalam struktur kondisi *if...else* minimal terdapat dua pernyataan. Jika kondisi yang diperiksa bernilai benar atau terpenuhi maka pernyataan pertama yang dilaksanakan dan jika kondisi yang diperiksa bernilai salah maka pernyataan yang kedua yang akan dilaksanakan. Bentuk umumnya adalah sebagai berikut:

```
if (kondisi) {  
    pernyataan1;  
} else {  
    pernyataan2;  
}
```

#### 2.6.4.3 Kondisi Switch Case

Struktur kondisi *switch...case...default* digunakan untuk penyeleksian kondisi dengan kemungkinan yang terjadi cukup banyak. Struktur ini akan melaksanakan salah satu dari beberapa pernyataan '*case*' tergantung nilai kondisi yang ada di dalam *switch*. Selanjutnya proses diteruskan hingga ditemukan pernyataan '*break*'. Jika tidak ada nilai pada *case* yang sesuai dengan nilai kondisi

maka proses akan diteruskan kepada pernyataan yang ada di bawah '*default*'.

Bentuk umum dari struktur kondisi ini adalah

```
Select Case(kondisi) {
case 1: pernyataan-1; break;
case 1: pernyataan-1; break;
...
...
case n: pernyataan-n; break;
Default: pernyataan-m
}
```

### 2.6.5 Perulangan

Perulangan digunakan untuk mengulang satu pernyataan atau beberapa pernyataan yang terdapat di dalam blok perulangan selama kondisi yang diberikan masih terpenuhi. Beberapa struktur perulangan dijelaskan sebagai berikut.

#### 2.6.5.1 Perulangan For

Pernyataan *for* digunakan untuk mengulang sebuah blok *statement* yang akan terus berputar. Biasanya, pernyataan *for* digunakan untuk mengombinasikan sebuah *array* dalam menyatakan sebuah pin atau data. Berikut *header* untuk perulangan *for*:

```
for (initialization; condition; expression) {
doSomething;
}
```

#### 2.6.5.2 Perulangan While

*While* akan melakukan perulangan secara kontinu dan tidak terbatas sampai ekspresi tersebut menemukan Kembali parenthesisnya, () menjadi *false*.

Perulangan ini banyak digunakan apabila jumlah perulangannya belum diketahui. Biasanya digunakan untuk mengetes sebuah sensor karena perulangan ini tak akan berakhir sampai adanya kondisi eksternal yang menutupnya.

```
while (someVariable?? value) {
doSomething;
}
```

### 2.6.5.3 Perulangan Do...While

Perulangan *do* bekerja seperti cara *while* melakukan perulangan, namun bedanya perulangan *do* hanya mengerjakan sekali di akhir pernyataan. Berikut adalah bentuk umumnya:

```
do {
doSomething;
} while (test condition);
```

*Sumber: Fahmizal dkk, 2022*

## 2.7 Game Tebak Ayat dan Surah Al-Baqarah

Berdasarkan judul penelitian yang telah ditentukan, maka pada sub-judul ini akan dijelaskan mengenai *game* tebak ayat dan surah Al-Baqarah sebagai berikut.

### 2.7.1 Game Tebak Ayat

Kata *Game* pada Bahasa Inggris diterjemahkan ke Bahasa Indonesia adalah permainan. Sesuatu yang digunakan untuk bermain; barang atau sesuatu yang dimainkan adalah pengertian permainan menurut Kamus Besar Bahasa

Indonesia (KBBI). *Game* tebak ayat merupakan *game* edukasi yang berfokus pada hafalan dengan menebak ayat. Menurut Arsy dkk (2022), *game* edukasi merupakan suatu permainan yang telah dirancang khusus untuk mengajar orang tentang suatu subjek tertentu, memperluas konsep, memperkuat pembangunan, memahami sebuah peristiwa historis atau budaya, atau membantu mereka dalam mempelajari keterampilan dalam bermain. Menurut M. Riyan dkk (2023), *game* edukasi adalah permainan yang dibuat dengan tujuan pembelajaran yang bukan hanya bermaksud menghibur sehingga diharapkan bisa menambah wawasan pengetahuan, *game* edukasi merupakan permainan yang dikemas untuk merangsang daya pikir dan termasuk salah satu cara untuk melatih meningkatkan konsentrasi penggunaannya, pengembangan *game* edukasi bertujuan agar siswa mampu mencapai tujuan pembelajaran.

Menurut Kamus Besar Bahasa Indonesia (KBBI), menebak berasal dari kata dasar tebak memiliki arti menerka; menduga; mengira-ngira(kan). Penelitian ini menggunakan ayat-ayat dalam surah Al-Baqarah sebagai tebakan atau sesuatu yang ditebak. Penelitian *game* tebak ayat ini berfokus pada metode memahami hafalan dengan menebak ayat yang didengar. Menurut Salman dkk (2023), untuk memahami Al-Qur'an, Anda perlu banyak pengetahuan. Dalam *ulumul qur'an*, dibahas tentang syarat-syarat *mufasssir*. Ini termasuk memahami berbagai cabang ilmu bahasa Arab, ilmu *Qira'at*, ilmu tauhid, dasar-dasar tafsir, sebab turunnya Al-Qur'an, dan lain-lain.

### 2.7.2 Surah Al-Baqarah

Surah Al-Baqarah merupakan surah kedua setelah Al-Fatihah, pada susunan mushaf Al-Qur'an. Surah Al-Baqarah diturunkan pada urutan ke-87, sesudah surat Al-Mutaffifin dan sebelum surah Al-Anfal. Surah ini merupakan surah terpanjang dari antara 114 surah dalam Al-Qur'an, yaitu terdiri atas 286 ayat. Surah ini turun di Madinah sehingga termasuk surah Madaniyyah, dan sebagian besar diturunkan pada permulaan Hijriah. Surah ini adalah surah kedua dalam Al-Qur'an yang terdiri dari juz 1 (ayat 1-141), juz 2 (ayat 142-252), dan juz 3 (ayat 252-286).

Menurut M. Yusub (2021), Khalid ibnu Madan mengatakan, “Surah Al-Baqarah adalah *fustat* (perhiasan) Al-Quran.” Sebagian ulama mengatakan bahwa surah Al-Baqarah mengandung 1.000 kalimat berita, 1.000 kalimat perintah dan 1.000 kalimat larangan. Di dalamnya terdapat 286 ayat, 6.221 kalimat, dan hurufnya berjumlah 25.500. Surah Al-Baqarah ini dinamai *As-sinam* yang berarti puncak, karena tiada lagi puncak petunjuk setelah kitab suci ini, dan tiada puncak setelah kepercayaan kepada Allah Yang Maha Esa dan keniscayaan hari kiamat. Surah Al-Baqarah ini juga dinamai *Az-zahra'* yakni terang benderang, karena kandungan surah ini menerangi jalan dengan benderang menuju kebahagiaan dunia dan akhirat, serta menjadi penyebab bersinar terangnya wajah siapa yang mengikuti petunjuk-petunjuk surah ini kelak di kemudian hari.

Surah Al-Baqarah mengandung beberapa topik utama, berikut adalah penjelasan tentang surah Al-Baqarah ayat 1 sampai ayat 286:

1. Ayat 1 sampai ayat 20 membahas tentang Al-Qur'an sebagai petunjuk bagi orang-orang yang bertakwa, sifat-sifat orang mukmin, orang kafir, dan orang munafik.
2. Ayat 21 sampai ayat 39 membahas tentang seruan kepada manusia untuk beribadah hanya kepada Allah *subhanahu wa ta'ala*, penciptaan manusia, serta kisah Nabi Adam *alayhi salam*, termasuk perintah Allah *subhanahu wa ta'ala* untuk tidak mendekati pohon terlarang.
3. Ayat 40 sampai ayat 123 membahas tentang peringatan kepada Bani Israil atas penyimpangan mereka dari ajaran Allah *subhanahu wa ta'ala*, serta kisah-kisah Nabi Musa *alayhi salam*, keingkaran kaum Yahudi, dan berbagai nikmat yang Allah berikan kepada mereka.
4. Ayat 124 sampai ayat 141 membahas tentang kisah Nabi Ibrahim *alayhi salam* dan keturunannya, pentingnya mengikuti *millah* (ajaran) Ibrahim *alayhi salam*, serta penyebutan tentang Ka'bah sebagai kiblat.
5. Ayat 142 sampai ayat 152 membahas tentang Perintah untuk menghadap ke Ka'bah sebagai kiblat dalam shalat, serta pesan tentang pentingnya mengikuti petunjuk Allah *subhanahu wa ta'ala*.
6. Ayat 153 sampai ayat 177 membahas tentang perintah sabar, kisah tentang kaum yang dibunuh karena agama mereka, perintah untuk bersedekah, berpuasa, dan menjalankan berbagai ibadah sebagai tanda ketakwaan.
7. Ayat 178 sampai ayat 203 membahas tentang Hukum *qisas* (balasan yang setimpal), wasiat, puasa di bulan Ramadhan, hukum-hukum terkait pernikahan, dan ibadah haji.

8. Ayat 204 sampai ayat 242 membahas tentang hukum-hukum sosial, termasuk tentang perdagangan, hutang, pernikahan, perceraian, *iddah* (masa tunggu setelah perceraian), dan hak-hak perempuan.
9. Ayat 243 sampai ayat 253 membahas tentang kisah-kisah para nabi dan umat terdahulu yang diuji oleh Allah *subhanahu wa ta'ala*, termasuk kisah Nabi Musa *alayhi salam*, Talut, dan Jalut.
10. Ayat 254 sampai ayat 283 membahas tentang penekanan pada pentingnya bersedekah, peringatan tentang riba, keutamaan sedekah, serta aturan terkait hutang piutang.
11. Ayat 284 sampai ayat 286 membahas tentang ajakan untuk bertakwa, berdoa, dan berserah diri kepada Allah *subhanahu wa ta'ala*, serta permohonan ampun atas kelemahan dan kekhilafan manusia.

## 2.8 Penelitian Terdahulu

Bersasarkan sub-sub-judul yang telah dijelaskan, maka pada sub-judul ini akan dijelaskan dua penelitian terdahulu yang berkaitan dengan *game* tebak ayat dan surah Al-Baqarah sebagai berikut.

### 2.8.1 Penelitian Oleh Alimin dkk

Permasalahan yang dihadapi oleh mahasiswa pengurus atau Dewan Kemakmuran Masjid Universitas Teknologi Sumbawa (DKM-UTS) Masjid Al-Kahfi adalah keterbatasan dalam sistem yang dapat mempermudah mereka dalam pemutaran audio. Sistem yang digunakan saat ini dirasa sangat kurang efisien dan

sering mengalami gangguan saat pemutaran, seperti kualitas suara yang kurang konsisten dan sulitnya pengoperasian perangkat yang sudah ada bagi pengguna selain mahasiswa DKM-UTS tidak familiar dengan teknologi tersebut. Penelitian oleh Alimin dkk (2024), Implementasi Modul Dfplayer Mini Mp3 Untuk Mempermudah Pemutaran Audio Surah Al-Quran di Masjid Al-Kahfi Universitas Teknologi Sumbawa, merupakan upaya dalam membantu Dewan Kemakmuran Masjid (DKM). Kegiatan ini bertujuan untuk mengimplementasikan modul DFPlayer Mini MP3 sebagai solusi teknologi yang sederhana dan mudah dioperasikan. Metode yang diterapkan dalam kegiatan pengabdian ini meliputi pelatihan tata cara penggunaan DFPlayer kepada DKM.

Menurut survei yang dilakukan oleh tim pengabdian, sistem suara Masjid Al-Kahfi saat ini menggunakan *amplifier* yang sudah tua dan tidak efisien. Selain itu, masjid memiliki satu *mixer* audio yang dapat digunakan oleh pengurus untuk memutar musik melalui porta USB. Dalam praktiknya, pengurus harus mengganti *flashdisk* setiap kali mereka ingin memutar *file* MP3. Ini menjadi tidak praktis dan memakan waktu. Desain alat ini bertujuan untuk memudahkan pengurus masjid dalam mengganti audio tanpa harus sering mengganti *flashdisk*, sehingga operasional pemutaran audio di masjid menjadi lebih efisien dan praktis.

SD *card* sebagai media penyimpanan audio yang akan diputar, harus diisi dengan *file* audio berformat .MP3. Untuk mempermudah proses pembacaan *file* audio oleh DFPlayer, maka nama dari *file* ditulis dengan urutan nomor 001, 002, 003 dan seterusnya. Alat pemutar audio untuk Masjid Al-Kahfi Universitas Teknologi Sumbawa yang dibuat berfungsi dengan baik, sehingga kegiatan

pengabdian ini berhasil diselesaikan sepenuhnya. Meskipun ada beberapa kendala yang menyebabkan penundaan karena kerusakan DFPlayer, alat ini mempermudah pengurus masjid untuk memutar musik saat diperlukan. Alat ini dapat dikembangkan lebih lanjut untuk pengabdian selanjutnya dengan menambah *case*, memperbaiki PCB, dan menambahkan layar untuk menampilkan daftar dan status audio yang sedang diputar.

### 2.8.2 Penelitian Oleh Alifiyah dkk

Bahasa berperan penting dalam perkembangan intelektual, sosial, dan emosional manusia sejak usia dini. Bahasa memungkinkan seseorang untuk mempelajari hampir semua bidang. Dalam konteks pembelajaran, belajar bahasa asing juga dapat berdampak pada kehidupan seseorang. Observasi di berbagai sekolah anak usia dini di Kecamatan Sidoarjo menunjukkan bahwa meskipun anak-anak mudah menghafal Al-Qur'an, mereka sering kali tidak memahami artinya. bahasa Arab bukan komponen wajib dalam kurikulum PAUD namun, pengajarannya tetap penting karena masa *golden age* atau dalam bahasa Arab *masyhur* adalah periode terpenting dalam perkembangan otak dan rasa ingin tahu anak. Dengan perkembangan zaman yang semakin modern, anak usia dini dituntut untuk tidak hanya mahir dalam bahasa lokal, tetapi juga perlu diperkenalkan dengan bahasa asing seperti Bahasa Arab untuk memperluas kemampuan mereka di berbagai aspek kehidupan.

Observasi yang dilakukan di sekolah POS PAUD Buah Hati Sidoarjo pada 24 Maret 2022, ditemukan bahwa pembelajaran kosakata bahasa Arab kurang

inovatif. Hal ini terungkap melalui wawancara dengan guru kelas dan observasi langsung di kelas, di mana penggunaan metode bernyanyi yang berulang menyebabkan siswa kurang fokus. Untuk mengatasi masalah ini, dirancang media pembelajaran baru berbasis Arduino Mega 2560. Media ini didesain dengan gambar kartun yang membawa papan bergambar kosakata Bahasa Arab, sebagai solusi untuk meningkatkan perhatian dan minat siswa. Alat ini diharapkan memberikan inovasi baru dalam media pembelajaran bahasa Arab yang akan membantu siswa menguasai kosakata bahasa Arab dengan cara yang mudah dan menarik.

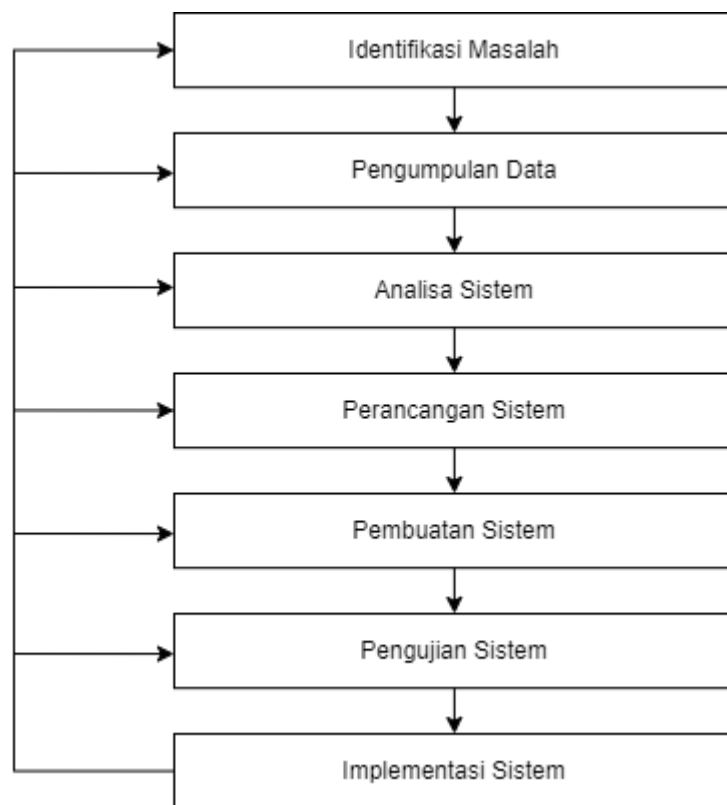
Arduino Mega 2560 otomatis mengeluarkan output berdasarkan perintah pembuatan pada sistem sebelumnya. Perancangan sistem terbaru, Arduino Mega berfungsi sebagai pengontrol suara pada Desain Pembelajaran Bahasa Arab Berbasis Arduino Mega di POS PAUD Buah Hati dengan menekan tombol. Setiap tombol memiliki posisi yang disesuaikan dengan desain karakternya. Penelitian Alifiyah dan Khizanatul (2024), Alat Pembelajaran Bahasa Arab Inovatif Berbasis Arduino untuk Pendidikan Anak Usia Dini, mengembangkan Desain Media Pembelajaran Bahasa Arab Berbasis Arduino Mega di POS PAUD Buah Hati, dengan pendekatan Research and Development (R&D). Peneliti merancang alat media pembelajaran kosakata bahasa Arab berbasis Arduino Mega, menggunakan model R&D ADDIE sebagai acuan.

## BAB III

### METODOLOGI PENELITIAN

#### 3.1 Kerangka Kerja Penelitian

Kerangka kerja penelitian merupakan tahapan awal dari sebuah penelitian yang dilakukan untuk mempermudah langkah-langkah dari pembuatan alat yang akan dibuat. Kerangka kerja penelitian yang akan dilakukan diuraikan pada gambar 3.1 berikut.



**Gambar 3.1 Kerangka Kerja Penelitian**

### **3.2 Uraian Kerangka Kerja Penelitian**

Berdasarkan kerangka kerja penelitian yang telah digambarkan di atas, maka dapat diuraikan pembahasan masing-masing tahapan dalam menyelesaikan penelitian ini sebagai berikut.

#### **3.2.1 Identifikasi Masalah**

Identifikasi masalah dilakukan dengan pendekatan terhadap objek penelitian. Tujuan dari tahap ini adalah untuk mengetahui permasalahan yang terjadi secara tepat sehingga diharapkan penelitian ini dapat memberikan solusi yang paling optimal terhadap pemecahan masalah tersebut. Permasalahan yang berhasil diidentifikasi adalah menurunnya motivasi siswa Sekolah Dasar karena metode yang hanya membaca Al-Qur'an tanpa variasi. Hal ini berdampak pada proses hafalan menjadi terhambat dan konsentrasi anak dalam menghafal Al-Qur'an menurun. Permasalahan lain yaitu efektivitas metode hafalan dan target hafalan yang memerlukan banyak tenaga guru dan waktu untuk memastikan semua anak hafal dengan benar.

#### **3.2.2 Pengumpulan Data**

Penelitian ini melakukan pengumpulan data untuk pembuatan alat dan pengerjaan laporan yang akurat dengan menggunakan metodologi penelitian yang dijelaskan sebagai berikut.

### 3.2.2.1 Waktu Penelitian

Penelitian dilakukan dengan memproses data-data yang berkaitan dengan alat yang akan dirancang dan dibuat. Waktu pengumpulan data mulai dilakukan dari bulan Mei 2024 hingga September 2024. Jadwal penelitian tersebut dapat dilihat pada tabel 3.1 berikut.

**Tabel 3.1 Jadwal Penelitian**

| No. | Kegiatan            | 2024 |   |   |   |      |   |   |   |      |   |   |   |         |   |   |   |           |   |   |   |
|-----|---------------------|------|---|---|---|------|---|---|---|------|---|---|---|---------|---|---|---|-----------|---|---|---|
|     |                     | Mei  |   |   |   | Juni |   |   |   | Juli |   |   |   | Agustus |   |   |   | September |   |   |   |
|     |                     | 1    | 2 | 3 | 4 | 1    | 2 | 3 | 4 | 1    | 2 | 3 | 4 | 1       | 2 | 3 | 4 | 1         | 2 | 3 | 4 |
| 1.  | Potensi dan Masalah |      |   |   |   |      |   |   |   |      |   |   |   |         |   |   |   |           |   |   |   |
| 2.  | Pengumpulan Data    |      |   |   |   |      |   |   |   |      |   |   |   |         |   |   |   |           |   |   |   |
| 3.  | Desain Sistem       |      |   |   |   |      |   |   |   |      |   |   |   |         |   |   |   |           |   |   |   |
| 4.  | Validasi Sistem     |      |   |   |   |      |   |   |   |      |   |   |   |         |   |   |   |           |   |   |   |
| 5.  | Perancangan Sistem  |      |   |   |   |      |   |   |   |      |   |   |   |         |   |   |   |           |   |   |   |
| 6.  | Pembuatan Sistem    |      |   |   |   |      |   |   |   |      |   |   |   |         |   |   |   |           |   |   |   |
| 7.  | Uji Coba Tahap 1    |      |   |   |   |      |   |   |   |      |   |   |   |         |   |   |   |           |   |   |   |
| 8.  | Uji Coba Tahap 2    |      |   |   |   |      |   |   |   |      |   |   |   |         |   |   |   |           |   |   |   |
| 9.  | Implementasi Sistem |      |   |   |   |      |   |   |   |      |   |   |   |         |   |   |   |           |   |   |   |
| 10. | Pembuatan Laporan   |      |   |   |   |      |   |   |   |      |   |   |   |         |   |   |   |           |   |   |   |

### 3.2.2.2 Metode Penelitian

Dalam melakukan penelitian agar mendapat hasil seperti yang diharapkan, maka sekiranya diperlukan suatu metodologi penelitian yang umum dilakukan, yaitu:

### 1. Penelitian Perpustakaan (*Library Research*)

Tujuan penelitian perpustakaan adalah untuk melengkapi kaidah, konsep, teori dan lain-lain sehingga menjadikan penelitian ini mempunyai landasan dan keilmuan yang tervalidasi. Penelitian ini dilakukan dengan membaca buku, jurnal, referensi dan lain-lain menyangkut masalah yang berhubungan dengan penelitian.

### 2. Penelitian Lapangan (*Field Research*)

Tujuan penelitian lapangan adalah untuk mengungkapkan makna yang diberikan oleh anggota masyarakat pada perilakunya dan kenyataan sekitar. Metode *field research* dapat diposisikan sebagai pembuka jalan kepada metode survey dan eksperimen.

### 3. Penelitian Laboratorium (*Laboratory Research*)

Tujuan penelitian laboratorium adalah untuk melakukan pengujian terhadap modul dan komponen terhadap sistem yang dibuat. Adapun perangkat lunak (*software*) dan perangkat keras (*hardware*) yang akan digunakan dalam penelitian ini dijelaskan pada tabel 3.2 berikut.

**Tabel 3.2 Hardware dan Software**

| Hardware                          | Software                         |
|-----------------------------------|----------------------------------|
| Satu unit laptop ASUS Memory 8 GB | Sistem Operasi Windows 11 64 bit |
| RAM SSD 256 GB                    | Microsoft Office 2019            |
| Arduino Mega 2560                 | Arduino IDE                      |
| Kabel downloader                  | Draw.io                          |
| Solder                            | Fritzing                         |
| Multitester                       | Photoshop                        |
| Printer 3D laser                  | Tinkercad                        |

### 3.2.3 Analisa Sistem

Analisa sistem merupakan tahap menjelaskan *input*, proses dan *output* yang terdapat pada sistem. *Input* adalah suatu masukan yang akan diproses dan menghasilkan sebuah keluaran atau *output*. Input yang digunakan pada penelitian ini adalah *push button* sebagai masukan untuk memulai *game*. Arduino Mega akan memproses *input* lalu *output* yang dihasilkan berupa teks akan ditampilkan pada LCD dan *output* yang berupa suara akan dilantunkan melalui speaker.

### 3.2.4 Perancangan Sistem

Perancangan sistem merupakan gambaran dan bentuk awal dari alat secara keseluruhan. Dalam memudahkan penelitian, merancang alat ini dibutuhkan beberapa panduan sebelum membuat alat, yaitu:

1. *Context Diagram*

*Context Diagram* menggambarkan rancangan keseluruhan. *Entity* eksternal harus digambarkan sedemikian rupa sehingga terlihat data yang mengalir pada *input-proses-output*.

2. *Data Flow Diagram (DFD)*

*Data Flow Diagram (DFD)* merupakan suatu diagram yang menggambarkan alir data dalam suatu entitas ke sistem atau sebaliknya. DFD juga dapat diartikan sebagai teknik grafis yang menggambarkan alir data dan transformasi yang digunakan sebagai perjalanan data dari masukan atau *input* menuju *output* atau keluaran.

### 3. *Blok Diagram*

*Blok diagram* dibuat untuk memetakan proses kerja pada suatu alat, hal ini bertujuan untuk memudahkan seseorang dalam mengenal komponen-komponen elektronika pada alat dan memahami alur kerja di dalamnya.

### 4. *Flowchart*

Bagan alur atau *flowchart* adalah diagram yang menampilkan langkah-langkah tahapan proses yang dilakukan oleh alat, dimulai dari awal hingga akhir proses suatu alat tersebut selesai.

### 5. *Software Design*

Desain dari alat yang akan dibuat berupa gambaran dan bentuk awal dari alat secara keseluruhan. Dengan adanya desain ini, maka prinsip kerja dari alat alat serta komponen-komponen dari sistem yang digunakan akan dapat dirancang dengan jelas. *Software* yang digunakan untuk perancangan alat adalah SketchUp, Tinkercad dan Photoshop.

### 6. Perakitan *Hardware*

Setelah desain dibuat dengan menggunakan *software*, tahap selanjutnya adalah menghubungkan komponen-komponen yang ada.

### 7. *Coding*

Pengkodean merupakan tahapan yang dilakukan setelah semua komponen dihubungkan. Arduino Mega 2560 yang digunakan akan diprogram sehingga dapat mengontrol *input* dan *output* dari sistem yang dibuat.

### 3.2.5 Pembuatan Sistem

Pembuatan sistem didasari oleh beberapa. Pada tahap ini, semua data yang telah dikumpulkan akan diolah dan semua komponen utama maupun komponen pendukung akan disatukan berdasarkan pertimbangan-pertimbangan pada perancangan sistem sebelumnya seperti *context diagram*, *data flow diagram*, *blok diagram*, *flowchart*, dan *design*.

### 3.2.6 Pengujian Sistem

Pengujian sistem merupakan tahap dimana alat sudah dibuat. Pada tahap ini, alat yang telah dibuat diuji atau melakukan *testing* untuk mengetahui kesalahan dalam sistem. Adapun tahap pengujian sistem adalah sebagai berikut.

1. Menyiapkan alat dan komponen-komponen pendukung yang akan digunakan seperti multimeter.
2. Memastikan baterai memiliki tegangan.
3. Menyalakan alat dengan menekan *switch*.
4. LED dan *buzzer* menyala menandakan alat aktif berjalan (*running*).
5. LCD menampilkan kalimat selamat datang, dan dua pilihan mode *game* yaitu mode mendengar ayat dan menebak ayat.
6. Memilih mode *game* yang terdapat pada LCD dengan menekan antara *push button* Y dan *push button* N.
7. *Push button* Y menjadikan alat sebagai mode *game* mendengar ayat kepada pengguna. LCD akan menampilkan “Ketik Ayat (Y) Dengarkan (N) Hapus”,

pengguna meng-*input* ayat yang ingin didengar dengan menekan *push button*. *Speaker* akan melantunkan ayat yang di-*input*. Pengguna dapat menaikkan atau menurunkan volume suara dengan memutar *wiper* pada potensiometer.

8. *Push button* N menjadikan alat sebagai mode *game* menebak ayat kepada pengguna. LCD akan menampilkan “Quiz Tebak Ayat Kesempatan: 3x Jumlah Soal 5 Selamat Menjawab” setelah beberapa detik LCD menampilkan “Soal ke- 1 dari 5 Kesempatan: 3 Masukkan Nomor Ayat”, dan *speaker* melantunkan ayat secara acak sampai selesai. Pengguna menjawab dengan menekan *push button*. Jika benar maka LCD menampilkan “Jawaban Benar”, LCD menampilkan sisa soal untuk menebak dan *speaker* akan menyala. Jika jawabanmu salah maka LCD akan menampilkan “Jawaban Salah. Jawaban Benar: ” dan *speaker* akan menyala.
9. Kesempatan akan berkurang jika jawaban salah.

### 3.2.7 Implementasi Sistem

Pada tahap ini, alat yang telah dibangun siap untuk dioperasikan sesuai dengan yang diinginkan peneliti. Implementasi sistem merupakan hasil dari perwujudan sistem yang dibangun bertumpu pada kerja sistem tersebut sudah mampu diterapkan. Tujuan implementasi sistem adalah untuk melihat seberapa tepat alat yang dibuat sesuai dengan kebutuhan dan mengidentifikasi masalah yang terjadi pada sistem. Implementasi alat pada penelitian ini yaitu pada siswa Sekolah Dasar yang merasa bosan namun ingin menghafal Al-Qur'an dengan pendekatan melalui teknologi berupa *game*.

## **BAB IV**

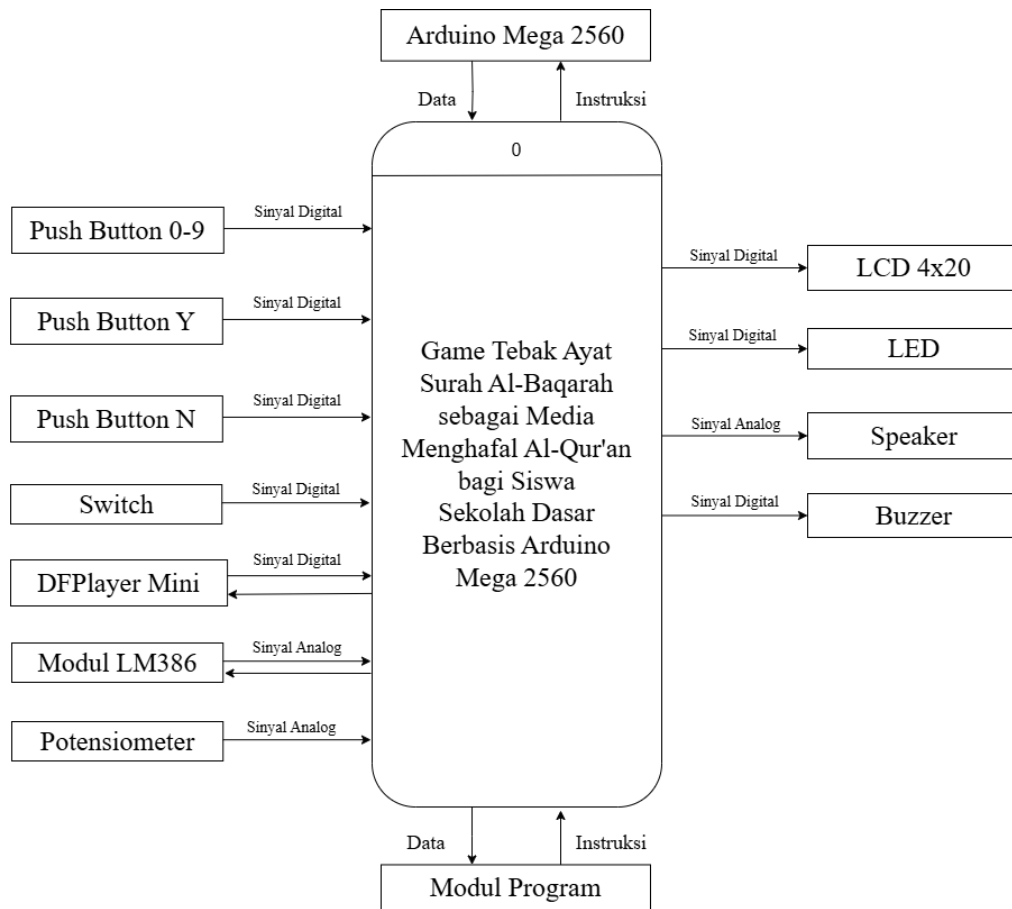
### **ANALISA DAN HASIL**

#### **4.1 Desain Sistem Secara Umum**

Desain dari sistem yang dibuat merupakan gambaran sistem secara keseluruhan. Prinsip kerja dari sistem dan setiap komponen yang digunakan akan terlihat jelas dengan adanya desain sistem secara umum.

##### **4.1.1 Context Diagram**

*Context Diagram* merupakan pendefinisian terhadap sistem yang akan dirancang dan bersifat secara menyeluruh. *Context Diagram* memperlihatkan adanya batasan dan interaksi antara *entity* eksternal dengan sistem. Hal ini digunakan untuk mempermudah dalam proses penganalisaan sistem yang dirancang secara keseluruhan. Berikut adalah gambaran *Context Diagram* dari sistem yang dapat dilihat pada gambar 4.1 berikut.



**Gambar 4.1 Context Diagram**

Sistem ini berkomunikasi dengan beberapa *entity*, yaitu Arduino Mega 2560, *push button*, LCD 20x4, baterai Lithium-Ion, *switch*, modul LM386, DFPlayer Mini, *speaker*, *buzzer*, LED, potensiometer dan Stepdown LM2596. *Entity* tersebut akan dijelaskan fungsinya sebagai berikut:

1. Arduino Mega 2560

Arduino Mega 2560 adalah mikrokontroler yang mengontrol seluruh sistem. Arduino Mega 2560 akan menjalankan program yang berfungsi sebagai game tebak ayat dan mengatur interaksi antara semua komponen.

2. *Push Button Y*

*Push button Y* digunakan oleh pengguna (siswa) untuk mode dengar ayat.

Siswa dapat menekan *push button Y* untuk konfirmasi “Yes” dengar ayat atau konfirmasi “Yes” tebak ayat.

3. *Push Button N*

*Push button N* digunakan oleh pengguna (siswa) untuk mode tebak ayat.

Siswa dapat menekan *push button N* untuk menghapus ayat yang ingin didengar atau ayat yang akan ditebak.

4. *Push Button 0-9*

*Push button 0-9* (*push button* angka nol sampai *push button* angka sembilan) digunakan oleh pengguna (siswa) untuk memberikan *input* ayat seberapa yang ingin didengar atau ayat yang akan ditebak.

5. LCD 20x4

LCD 20x4 adalah layar yang menampilkan informasi dan instruksi kepada pengguna. LCD 20x4 ini menampilkan teks pertanyaan tebak ayat, informasi dengar ayat, tampilan awal dan hasil permainan tebak ayat.

6. *Switch*

*Switch* berfungsi sebagai saklar utama untuk menghidupkan atau mematikan sistem. *Switch* memungkinkan pengguna untuk menghemat daya dengan mematikan perangkat saat tidak digunakan.

7. Modul LM386

Modul LM386 adalah modul penguat audio yang digunakan untuk memperkuat sinyal suara dari DFPlayer Mini sehingga suara yang dihasilkan oleh *speaker* lebih jelas dan keras.

8. Potensiometer

Potensiometer: Alat yang digunakan untuk mengatur volume suara yang keluar dari *speaker*. Potensiometer memungkinkan pengguna untuk menyesuaikan tingkat suara sesuai kebutuhan.

9. DFPlayer Mini

DFPlayer mini adalah modul pemutar audio MP3 yang dapat menyimpan dan memutar file audio dari kartu memori. Dalam konteks ini, DFPlayer Mini bisa digunakan untuk memutar rekaman ayat-ayat Al-Qur'an yang akan ditebak oleh siswa.

10. *Speaker*

*Speaker* menghasilkan *output* suara yang diperkuat oleh modul LM386. *Speaker* akan memutar audio dari DFPlayer mini seperti bacaan ayat yang harus ditebak oleh siswa.

11. LED

LED digunakan sebagai indikator visual yang memberikan umpan balik kepada pengguna bahwa sistem sedang berjalan. LED ini menyala saat sistem aktif dan berfungsi untuk menunjukkan bahwa perangkat sedang dalam keadaan operasional.

#### 12. *Buzzer*

*Buzzer* merupakan komponen *output* yang menghasilkan bunyi sederhana untuk memberikan umpan balik kepada pengguna. *Buzzer* digunakan untuk menandakan sistem berjalan.

#### 13. Stepdown LM2596

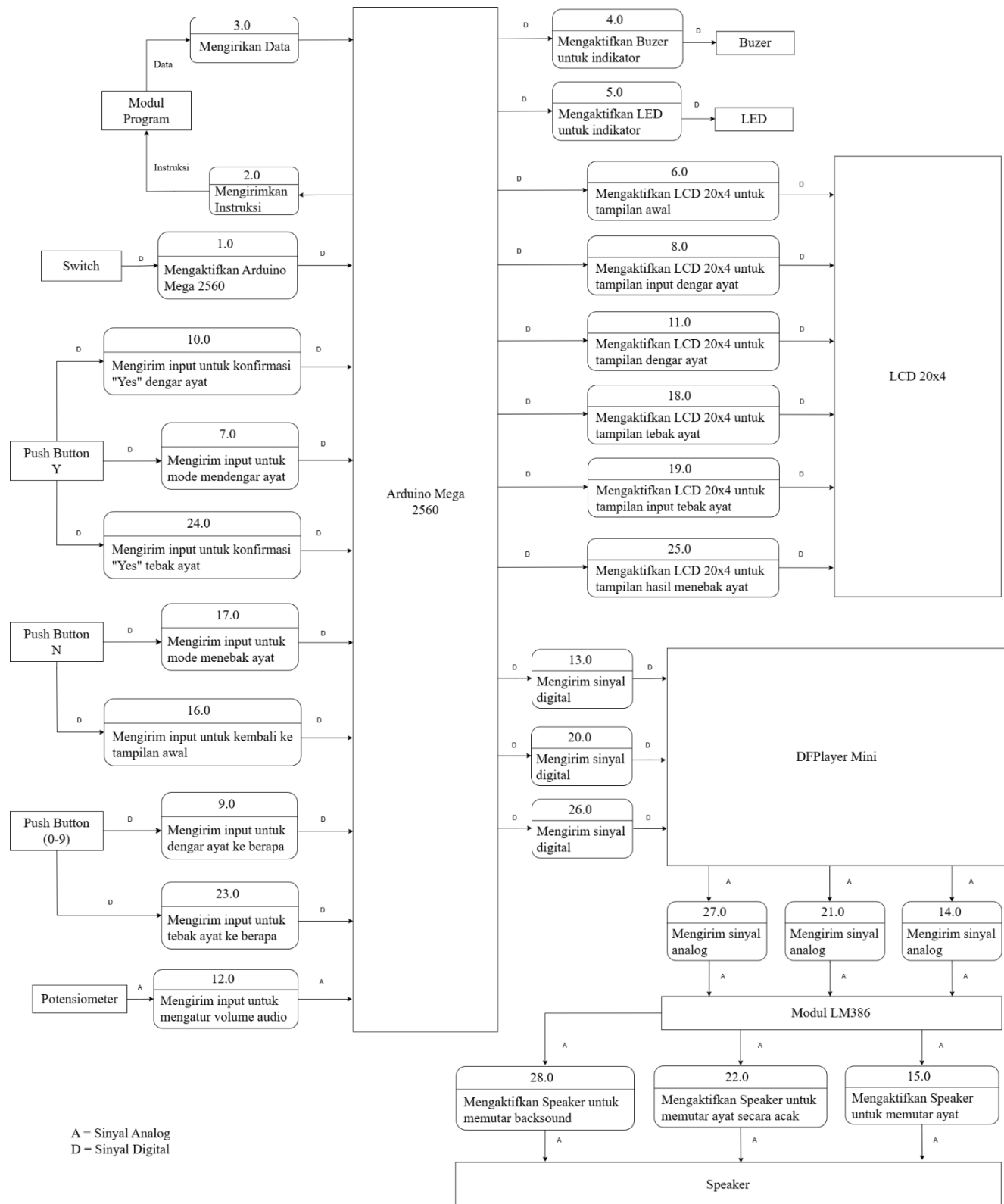
Stepdown LM2596 adalah modul konverter tegangan yang menurunkan tegangan dari baterai Lithium-Ion ke tegangan yang sesuai untuk komponen lainnya. Ini memastikan bahwa semua komponen menerima tegangan yang stabil dan aman.

#### 14. Modul Program

Modul program mengacu pada perangkat lunak atau kode yang ditulis dan diunggah ke Arduino Mega 2560. Program ini mengatur logika permainan, mengendalikan interaksi antar komponen, dan mengelola *input* dari pengguna serta *output* ke pengguna.

### 4.1.2 Data Flow Diagram

*Data Flow Diagram* adalah gambaran yang lebih rinci dari alat yang dirancang. *Data Flow Diagram* diuraikan berdasarkan *Context Diagram* yang telah dijelaskan sebelumnya. Gambar *Data Flow Diagram* dapat dilihat pada gambar 4.2 berikut.



Gambar 4.2 Data Flow Diagram

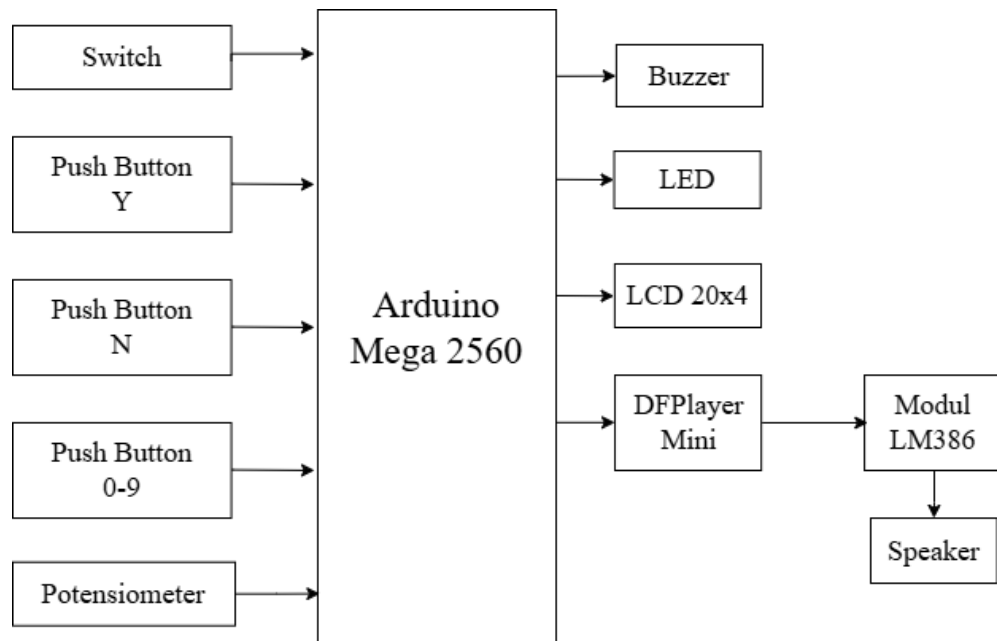
Berdasarkan gambar 4.2 di atas dapat dijelaskan bahwa *game* tebak ayat Surah Al-Baqarah memiliki beberapa urutan proses instruksi yaitu:

1. Switch mengaktifkan Arduino Mega 2560 dengan menghubungkan sumber daya dari baterai ke sistem (1.0).
2. Arduino Mega 2560 mengirim instruksi ke modul program untuk memproses data (2.0), hasil proses dari modul program dikirim kembali ke Arduino Mega 2560 (3.0).
3. Hasil proses diteruskan oleh Arduino Mega 2560 untuk mengaktifkan *buzzer* sebagai indikator sistem sedang berjalan (4.0) dan mengaktifkan LED (5.0).
4. Arduino Mega 2560 mengaktifkan LCD 20x4 untuk menampilkan tampilan awal (6.0).
5. *Push button Y* ditekan akan mengirimkan *input* untuk mode dengar ayat ke Arduino Mega 2560 (7.0). Arduino Mega 2560 mengaktifkan LCD 20x4 untuk menampilkan tampilan *input* dengar ayat (8.0). *Push button (0-9)* ditekan akan mengirimkan *input* ayat ke berapa yang ingin didengar ke Arduino Mega 2560 (9.0). *Push button Y* ditekan akan mengirimkan *input* konfirmasi “Yes” dengar ayat ke Arduino Mega 2560 (10.0).
6. Arduino Mega 2560 mengaktifkan LCD 20x4 untuk menampilkan tampilan ayat yang sedang didengar (11.0).
7. Potensiometer mengirim *input* untuk mengatur volume audio ke Arduino Mega 2560 (12.0) lalu ke DFPlayer Mini (13.0). DFPlayer Mini mengirim sinyal analog ke modul LM386 (14.0) untuk memutar ayat melalui *speaker* (15.0).

8. *Push button N* ditekan akan mengirimkan *input* untuk kemapi ke tampilan awal (16.0). *Push button N* ditekan akan mengirimkan *input* untuk mode tebak ayat ke Arduino Mega 2560 (17.0). Arduino Mega 2560 mengaktifkan LCD 20x4 untuk menampilkan tampilan tebak ayat (18.0). Arduino Mega 2560 mengaktifkan LCD 20x4 untuk menampilkan tampilan *input* tebak ayat (19.0).
9. Arduino Mega 2560 mengirim sinyal digital ke DFPlayer Mini (20.0). DFPlayer Mini dapat mengirim sinyal analog ke modul LM386 (21.0) dan mengaktifkan *speaker* untuk memutar ayat secara acak (22.0).
10. *Push button (0-9)* ditekan akan mengirimkan *input* ke Arduino Mega 2560 untuk menebak ayat (23.0). *Push button Y* ditekan untuk mengirimkan *input* konfirmasi “Yes” tebak ayat ke Arduino Mega 2560 (24.0). Arduino Mega 2560 mengaktifkan LCD 20x4 untuk menampilkan tampilan hasil menebak ayat (25.0).
11. Arduino Mega 2560 mengirim sinyal digital ke DFPlayer Mini (26.0) agar DFPlayer Mini dapat mengirim sinyal analog ke modul LM386 (27.0) dan mengaktifkan *speaker* sebagai media suara untuk memutar *backsound* (28.0).

#### 4.1.3 Blok Diagram

Dengan mengacu pada *Data Flow Diagram* di atas, penggunaan komponen-komponen pada alat dapat dilihat dalam *blok diagram* seperti gambar 4.3 berikut ini.



**Gambar 4.3 Blok Diagram**

Berdasarkan *blok diagram* di atas dapat dilihat bahwa sistem terdiri dari *input*, *process*, dan *output*. Mikrokontroler Arduino Mega 2560 berperan sebagai pusat pemrosesan dan pengontrolan. *Push button 0-9*, *push button Y*, *push button N*, *switch*, dan *potensiometer* sebagai *input*. *Output* dari sistem adalah LCD 20x4, LED, DFPlayer mini, modul LM386, *speaker*, dan *buzzer*.

## 4.2 Prinsip Kerja Sistem

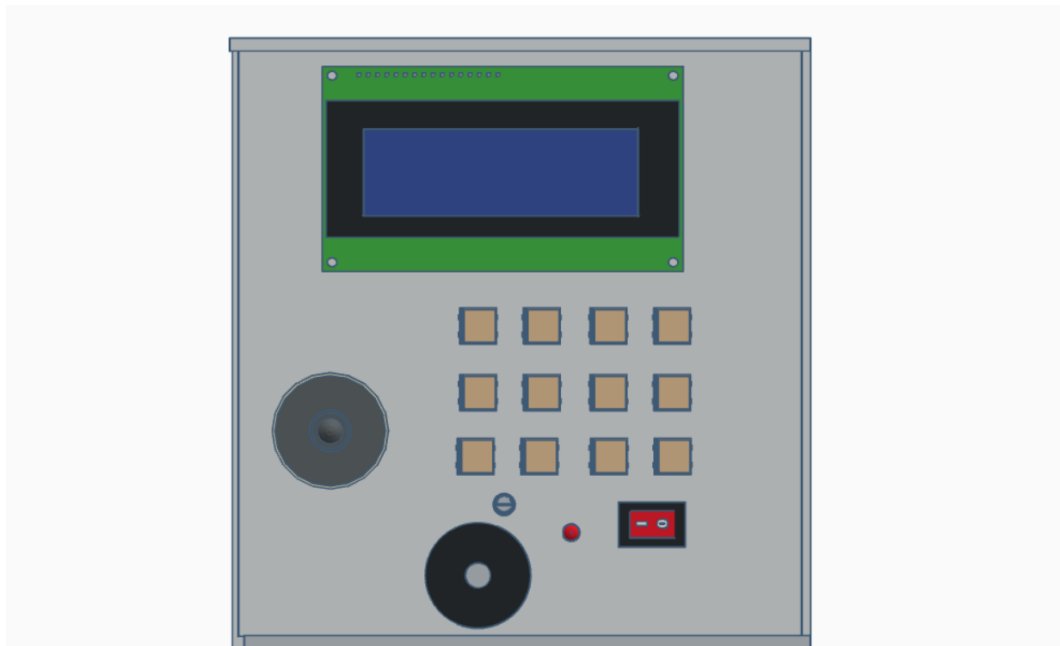
Adapun prinsip dan sistem kerja dari *game* tebak ayat Surah Al-Baqarah ini adalah sebagai berikut:

1. Alat ini menggunakan *switch* untuk mengaktifkan rangkaian dari sistem.
2. Arduino Mega 2560 sebagai kontrol dan pengendalian sistem alat.
3. LED dan *buzzer* akan menyala sebagai indikator alat sedang *running*.

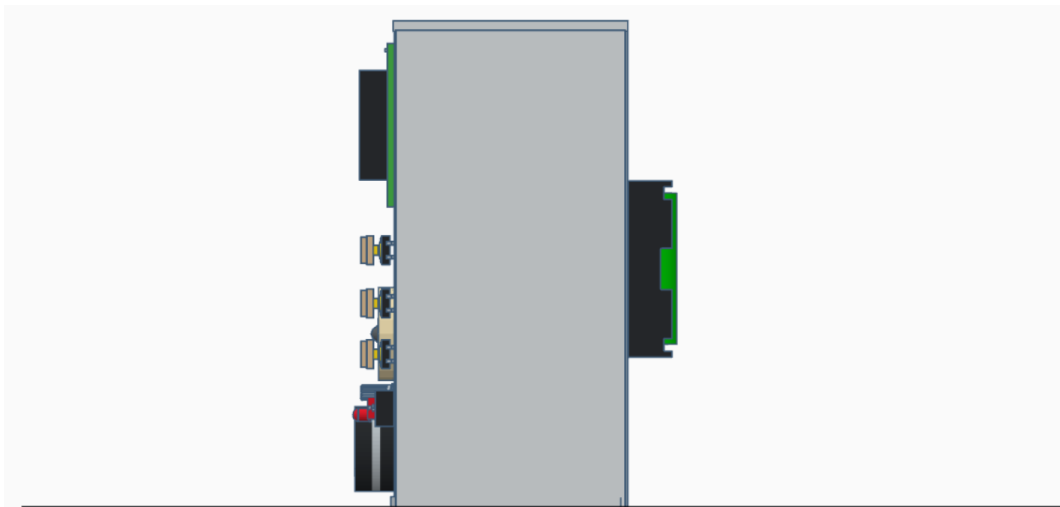
4. LCD 20x4 akan menampilkan pesan awal untuk mulai apakah pengguna ingin mendengar ayat atau menebak ayat.
5. Jika yang ditekan adalah *push button Y*, maka LCD 20x4 akan menampilkan pesan untuk menekan ayat yang ingin didengar.
6. Jika yang ditekan adalah *push button N*, maka LCD 20x4 akan menampilkan pesan tebak ayat.
7. Knop pada *potensiometer* dapat diputar untuk mengatur suara apabila suara lantunan ayat kurang jelas.
8. LCD 20x4 akan menampilkan hasil dan *speaker* akan menyala jika jawaban salah atau jawaban benar.

### 4.3 Rancangan Fisik Sistem

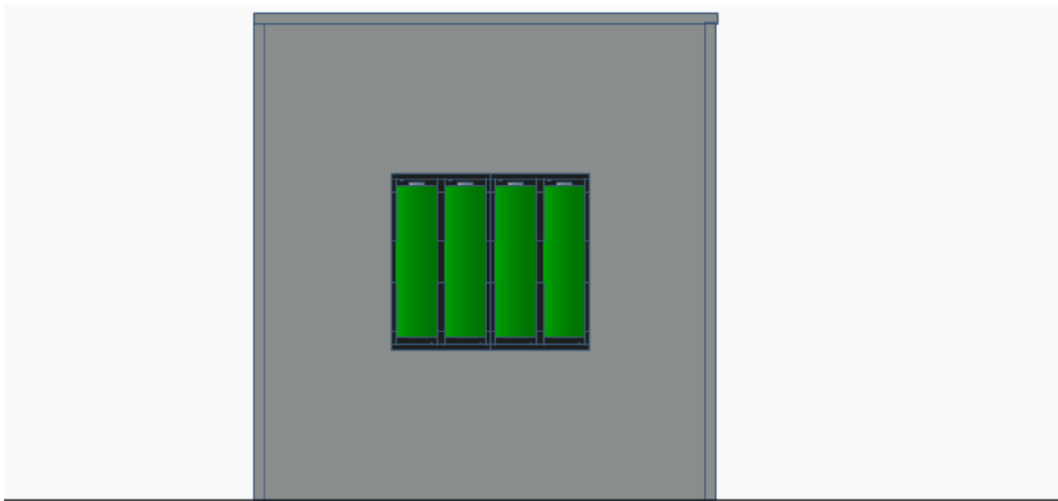
Perancangan alat ini merupakan tahap awal dari pemasangan dan menganalisa permasalahan yang dihadapi berdasarkan literatur. Hal ini dilakukan untuk memberikan gambaran jelas tentang alat yang akan dibuat sehingga tidak akan terjadi masalah ketika membuatnya. Rancangan fisik dari alat yang digambarkan menggunakan *software* Tinkercad yang dapat dilihat pada gambar berikut.



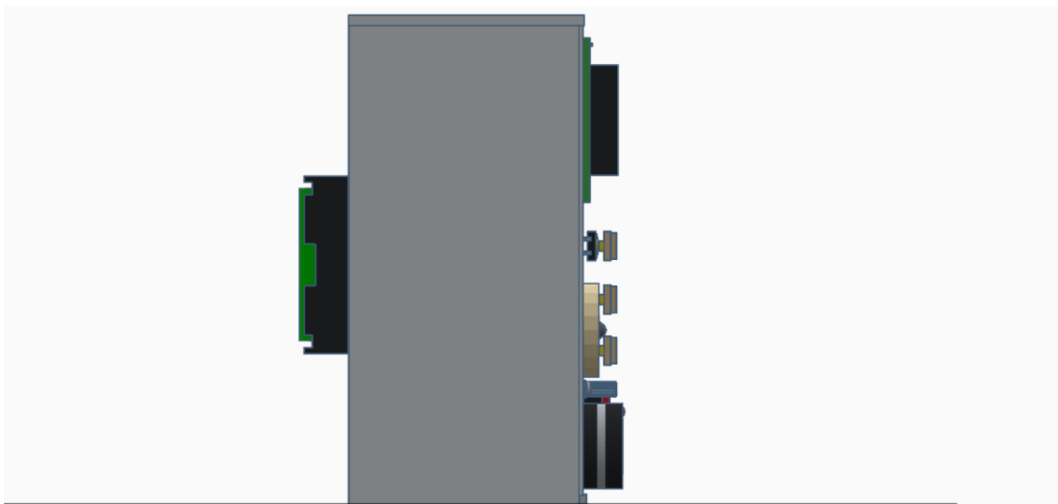
**Gambar 4.4 Rancangan Alat Tampak Depan**



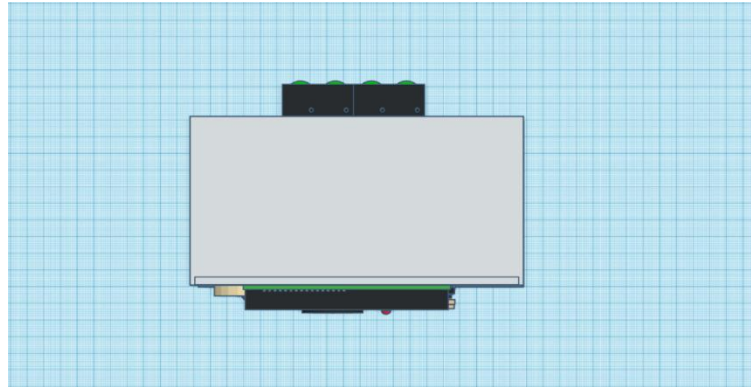
**Gambar 4.5 Rancangan Alat Tampak Kanan**



**Gambar 4.6 Rancangan Alat Tampak Belakang**



**Gambar 4.7 Rancangan Alat Tampak Kiri**



**Gambar 4.8 Rancangan Alat Tampak Atas**

Perancangan fisik alat memiliki peranan yang mendukung sistem, beberapa di antaranya terlihat pada gambar di atas. Rancangan fisik tersebut memperlihatkan komponen seperti LCD 20x4, *push button*, *speaker*, *switch*, *buzzer*, LED, dan potensiometer.

#### **4.4 Desain Sistem Terperinci**

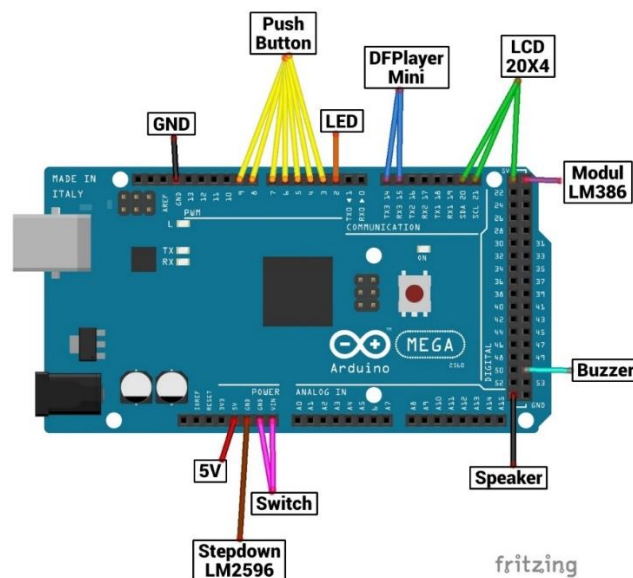
Desain dari sistem yang dibuat merupakan gambaran dari sistem secara keseluruhan. Dengan adanya desain ini, maka prinsip kerja dari sistem serta komponen-komponen dari sistem yang digunakan akan dapat dilihat dengan jelas.

##### **4.4.1 Rangkaian Mikrokontroler Arduino Mega 2560**

Pada sistem ini, Arduino Mega 2560 digunakan sebagai *entity* yang berfungsi sebagai pengendali. Arduino Mega 2560 adalah pengembangan mikrokontroler yang berbasis Arduino dengan menggunakan chip ATmega2560. Board ini memiliki pin I/O yang cukup banyak dengan 54 buah digital I/O pin (15

pin diantaranya adalah PWM), 16 pin analog *input*, dan 4 pin UART (serial port *hardware*). Arduino Mega 2560 dilengkapi dengan sebuah *oscillator* 16 Mhz, sebuah port USB, power jack DC, ICSP header, dan tombol reset. Board ini sudah lengkap dan memiliki segala yang dibutuhkan untuk sebuah mikrokontroler. Dengan penggunaan yang cukup sederhana, yaitu menghubungkan power dari USB ke laptop atau melalui adaptor AC/DC ke *jack* DC.

Blok rangkaian dari modul Arduino Mega 2560 dapat dilihat pada gambar 4.9 berikut.



**Gambar 4.9 Rangkaian Arduino Mega 2560**

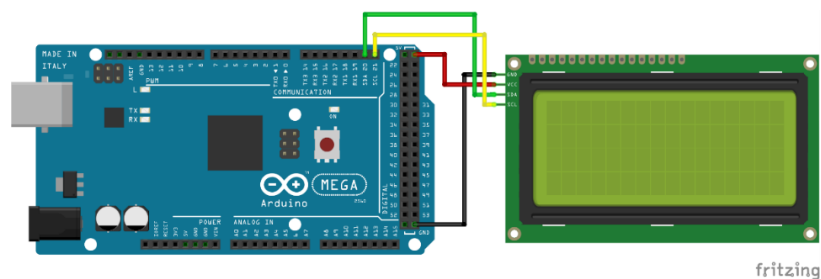
Berdasarkan gambar 4.9 di atas, konfigurasi pin yang digunakan pada modul Arduino Mega 2560 adalah sebagai berikut:

1. Pin 9, 8, 7, 6, 5, 4, dan 3 pada Arduino Mega 2560 terhubung pada *push button matrix* 4x3.
2. Pin 2 pada Arduino Mega 2560 terhubung pada LED.

3. Pin RX3 dan TX3 pada Arduino Mega 2560 terhubung pada DFPlayer Mini.
4. Pin SDA dan SCL pada Arduino Mega 2560 terhubung pada LCD 20x4.
5. Pin 5V pada Arduino Mega 2560 terhubung pada modul LM386.
6. Pin 51 pada Arduino Mega 2560 terhubung pada *buzzer*.
7. Pin GND pada Arduino Mega 2560 terhubung pada katoda *speaker*.
8. Pin VIN dan GND pada Arduino Mega 2560 terhubung pada *switch*.
9. Pin GND pada Arduino Mega 2560 terhubung pada Step-down LM2596.

#### 4.4.2 Rangkaian LCD 20x4

Perancangan alat ini menggunakan LCD 20x4 untuk menampilkan teks berupa pesan dan hasil jawaban. Rangkaian LCD 20x4 dapat dilihat pada gambar 4.10 berikut.



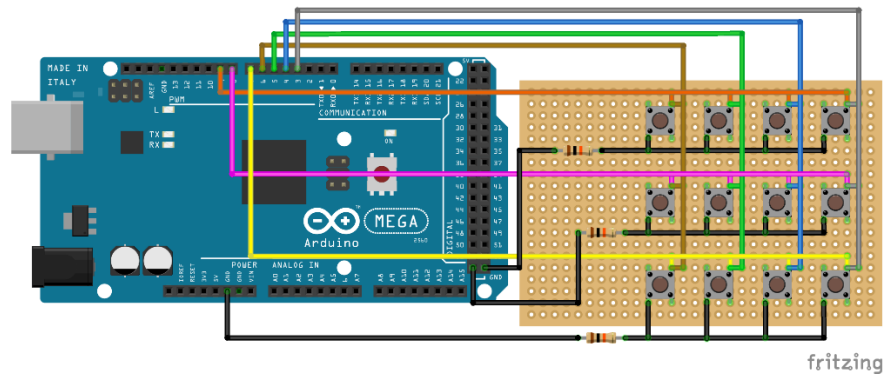
**Gambar 4.10 Rangkaian LCD 20x4**

Penjelasan pin yang digunakan pada LCD 20x4 yaitu:

1. Pin GND ke pin GND Arduino Mega 2560.
2. Pin Vcc ke pin 5V Arduino Mega 2560.
3. Pin SDA ke pin 20 Arduino Mega 2560.
4. Pin SCL ke pin 21 Arduino Mega 2560.

#### 4.4.3 Rangkaian Push Button

Perancangan alat ini menggunakan rangkaian *push button* sebagai *input* untuk memilih mode permainan antara mendengar ayat atau menebak ayat. Rangkaian *push button* dapat dilihat pada gambar 4.11 berikut.



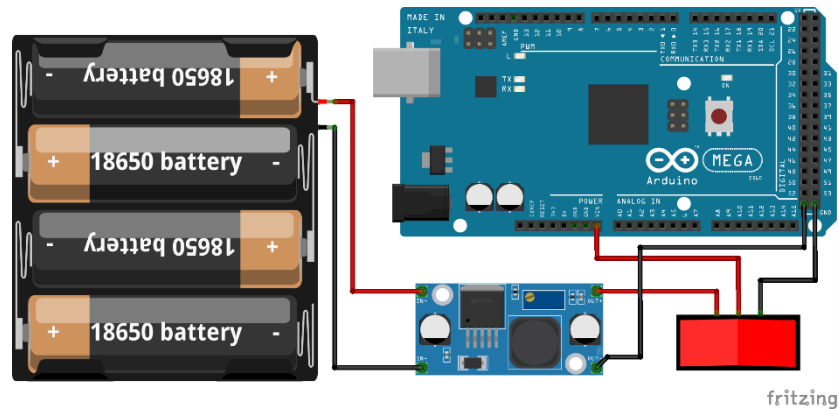
**Gambar 4.11 Rangkaian Push Button**

Penjelasan pin yang digunakan pada *push button* yaitu:

1. Pin baris pertama dan kolom pertama ke pin 9 Arduino Mega 2560.
2. Pin baris kedua dan kolom pertama ke pin 8 Arduino Mega 2560.
3. Pin baris ketiga dan kolom pertama ke pin 7 Arduino Mega 2560.
4. Pin baris pertama dan kolom kedua ke pin 6 Arduino Mega 2560.
5. Pin baris pertama dan kolom kedua ke pin 5 Arduino Mega 2560.
6. Pin baris pertama dan kolom kedua ke pin 4 Arduino Mega 2560.
7. Pin baris pertama dan kolom kedua ke pin 3 Arduino Mega 2560.
8. Pin baris pertama dengan 10K $\Omega$  resistor ke pin GND Arduino Mega 2560.
9. Pin baris kedua dengan 10K $\Omega$  resistor ke pin GND Arduino Mega 2560.
10. Pin baris ketiga dengan 10K $\Omega$  resistor ke pin GND Arduino Mega 2560.

#### 4.4.4 Rangkaian Switch

Perancangan alat ini menggunakan *switch* sebagai penyambung dan pemutus tegangan, Stepdown LM2596 untuk menstabilkan tegangan dan baterai sebagai sumber tegangan. Rangkaian *switch* dapat dilihat pada gambar 4.12 berikut.



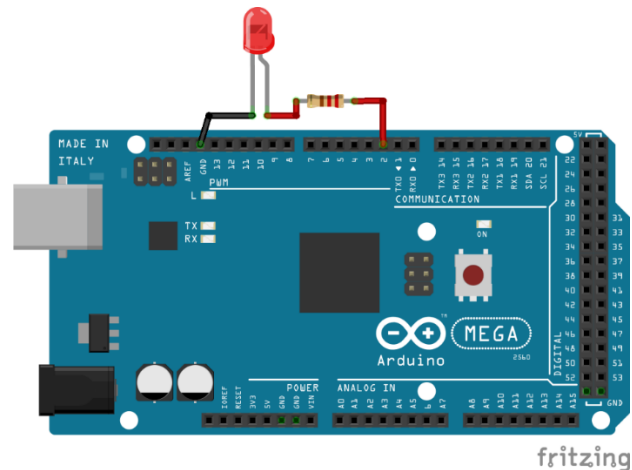
**Gambar 4.12 Rangkaian Switch**

Penjelasan pin yang digunakan pada *switch* yaitu:

1. Pin 1 *switch* ke pin OUT+ Stepdown LM2596.
2. Pin 2 *switch* ke pin VIN Arduino Mega 2560.
3. Pin 3 *switch* ke pin GND Arduino Mega 2560.
4. Anoda pada baterai ke pin IN+ Stepdown LM2596.
5. Katoda pada baterai ke pin IN- Stepdown LM2596.
6. Pin OUT- Stepdown LM2596 ke pin GND Arduino Mega 2560.

#### 4.4.5 Rangkaian LED

Perancangan alat ini menggunakan LED sebagai indikator alat sedang berjalan. LED yang digunakan adalah berwarna merah. Rangkaian LED dapat dilihat pada gambar 4.13 berikut.



**Gambar 4.13 Rangkaian LED**

Penjelasan pin yang digunakan pada LED yaitu:

1. Anoda pada LED dan resistor 220 $\Omega$  ke pin 2 Arduino Mega 2560.
2. Katoda pada LED ke pin GND Arduino Mega 2560.

#### 4.4.6 Rangkaian Speaker

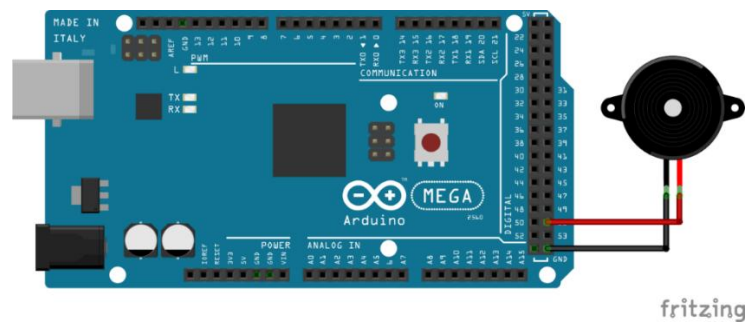
Perancangan alat ini menggunakan *speaker* sebagai *output* audio suara saat *user* memilih mode mendengar ayat atau saat mode menebak ayat dilantunkan. Konfigurasi lebih jelasnya dapat dilihat pada gambar 4.14 berikut.



13. Pin 1 GAIN LM386 ke katoda kapasitor.
14. Pin 8 GAIN LM386 ke anoda kapasitor.
15. Pin 7 BYPASS LM386 dengan kapasitor ke pin 2 Inverting Input LM386.

#### 4.4.7 Rangkaian Buzzer

Perancangan alat ini menggunakan *buzzer* sebagai *output* jika jawaban *user* saat menebak ayat salah. Konfigurasi lebih jelasnya dapat dilihat pada gambar 4.15 berikut.



**Gambar 4.15 Rangkaian Buzzer**

Penjelasan pin yang digunakan pada *buzzer* yaitu:

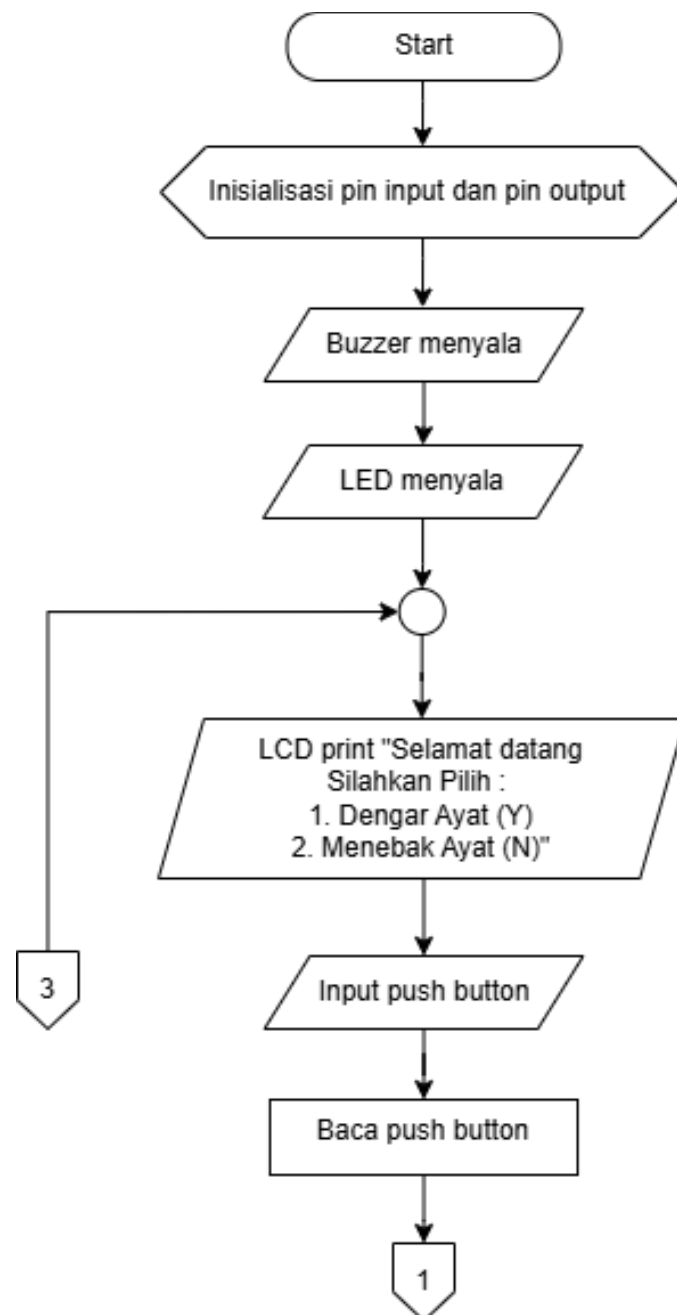
1. Anoda pada *buzzer* ke pin 51 Arduino Mega 2560.
2. Katoda pada *buzzer* ke pin GND Arduino Mega 2560.

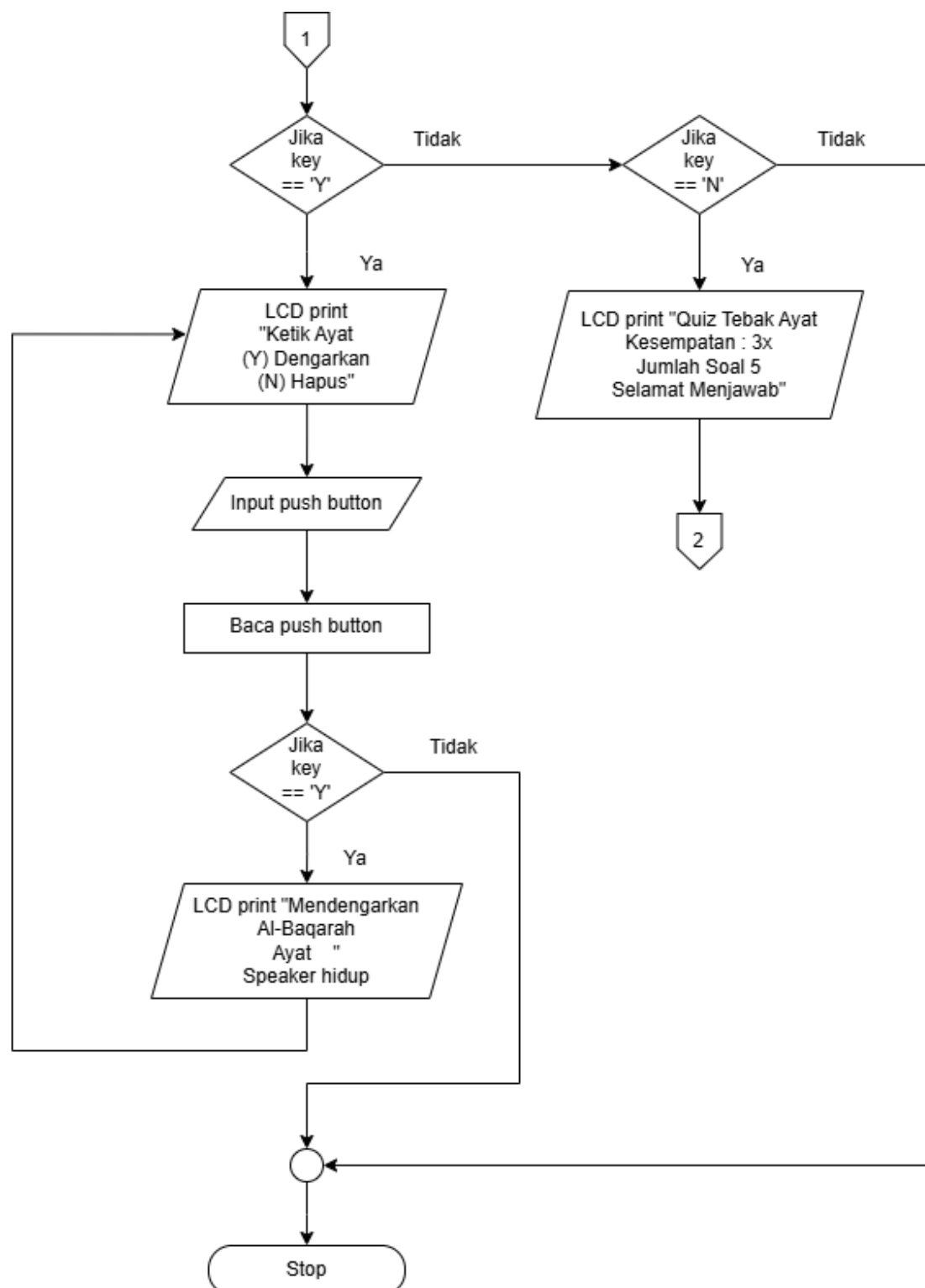
#### 4.5 Rancangan Modul Program

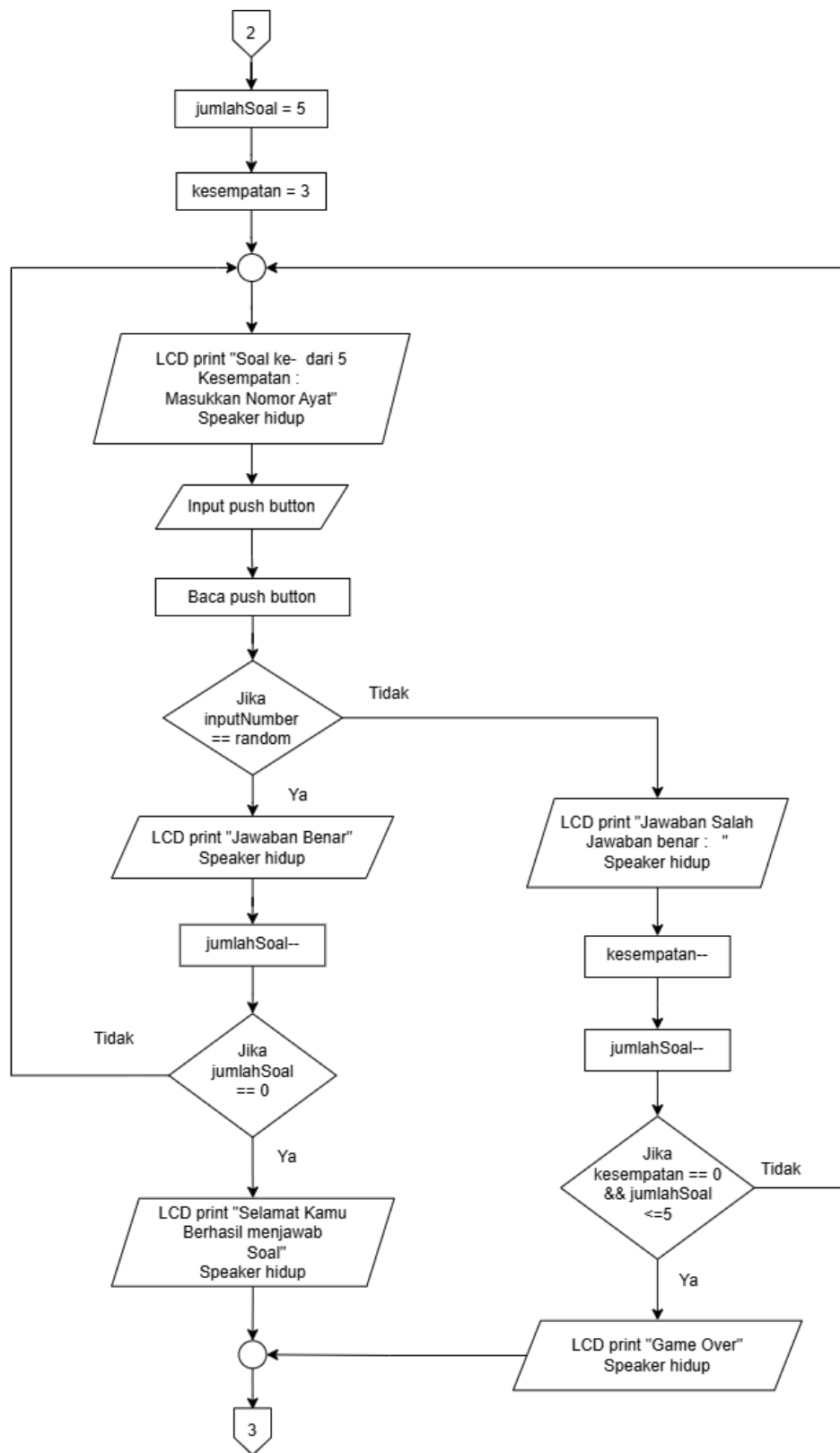
Pada sub bab ini menjelaskan tentang modul program yang digunakan pada sistem yang dirancang. Rancangan modul dapat menjadi dua bagian, yaitu *flowchart* dan *listing program*.

#### 4.5.1 Flowchart

Modul program dirancang menggunakan struktur dengan kualitas yang baik dan mudah dimengerti, maka sebelum pembuatan *listing program* perlu diawali dengan penentuan logika program. Logika dasar gambaran pada penulisan ini adalah dengan menggunakan *flowchart* seperti pada gambar 4.16 berikut ini.







Gambar 4.16 Rangkaian Flowchart Alat

### 4.5.2 Listing Program

Pada sub bab ini diuraikan mengenai modul program yang digunakan untuk membuat gim tebak ayat menggunakan *software* Arduino IDE 2.3.2 pada Arduino Mega 2560. Beberapa modul program yang digunakan adalah sebagai berikut.

#### 4.5.2.1 Program LCD 20x4

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 20, 4);
void setup() {
  lcd.begin();
  lcd.backlight();
  lcd.setCursor(0,0);
  lcd.print("Selamat Datang");
  lcd.setCursor(0,1);
  lcd.print("Silahkan Pilih :");
  lcd.setCursor(0,2);
  lcd.print("1. Dengar Ayat (Y)");
  lcd.setCursor(0,3);
  lcd.print("2. Menebak Ayat (N)");
}
```

#### 4.5.2.2 Program Push Button

```
#include <Keypad.h>
const byte ROWS = 3;
const byte COLS = 4;
char keys[ROWS][COLS] = {
  {'Y','N','0','1'},
  {'2','3','4','5'},
  {'6','7','8','9'}
};
byte rowPins[ROWS] = {9, 8, 7};
byte colPins[COLS] = {6, 5, 4, 3};
Keypad customKeypad = Keypad(makeKeymap(hexaKeys), rowPins,
colPins, ROWS, COLS);
```

```

void loop() {
  char customKey = customKeypad.getKey();
  if (customKey) {
    switch (customKey) {
      case 'Y':
        break;
      case 'N':
        break;
      default:
        break;
    }
  }
}

```

#### 4.5.2.3 Program Speaker

```

#include <DFRobotDFPlayerMini.h>
SoftwareSerial mySerial(10, 11);
#define potensio A1
int volumeAwal = 30;
void setup() {
  mySerial.begin (9600);
  mp3_set_serial (mySerial);
  delay(1);
  mp3_set_volume (30);
}

```

#### 4.5.2.4 Program LED

```

const int ledPin = 2;
void setup() {
  pinMode(led, OUTPUT);
}

```

#### 4.5.2.5 Program Buzzer

```

#define buzzer A0
void setup() {
  pinMode(buzzer, OUTPUT);
}

```

