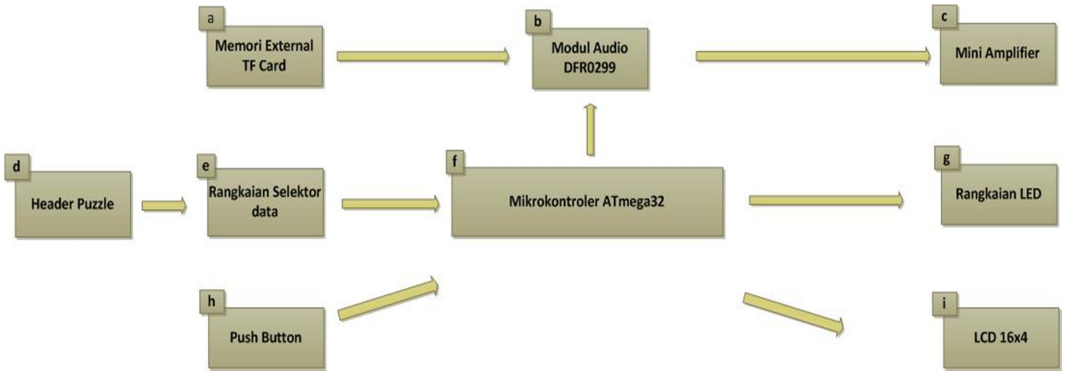
BAB III
PERANCANGAN SISTEM

Bab ini membahas semua perancangan yang dibuat untuk memenuhi maksud dan tujuan tugas akhir. Perancangan sistem terdiri dari perancangan perangkat keras, perancangan perangkat lunak dan perancangan mekanik. Gambar 3.1 adalah diagram blok dari keseluruhan sistem.



Gambar 3.1 Diagram blok keseluruhan sistem

Penjelasan mengenai diagram blok keseluruhan sistem adalah sebagai berikut :

a. **Memori Eksternal TF *Card*** : Berfungsi sebagai penyimpan informasi data *audio digital*. *Data audio digital* tersebut dibaca oleh modul *audio* DFR0299 untuk dilakukan pemrosesan selanjutnya.
b. **Modul *Audio* DFR0299** : Berfungsi untuk membaca informasi data *audio digital* pada TF *Card* dan merubahnya menjadi data *audio analog*. Data *audio analog* tersebut dikeluarkan dalam bentuk suara melalui *speaker*.
c. ***Mini Amplifier*** : Berfungsi untuk mengolah data *audio analog* dari modul *audio* DFR0299 untuk dikeluarkan dalam bentuk suara melalui *speaker*.
d. ***Header Puzzle*** : Berfungsi untuk menghasilkan data setiap potongan *puzzle*. Data potongan *puzzle* terdiri dari 6 bit data yang dihasilkan dari rangkaian listrik pada *header puzzle*. Untuk menghasilkan data biner 1 saluran data dihubungkan dengan tegangan 5 volt sedangkan untuk

menghasilkan data biner 0 saluran data dihubungkan dengan tegangan 0 volt.

e. **Rangkaian Selektor Data** : Berfungsi sebagai sebagai penyeleksi data dari setiap potongan *puzzle* yang dipasang untuk dikirimkan kepada mikrokontroler. Rangkaian selektor data membaca data dari saluran *header puzzle* dan mengirimkannya kepada mikrokontroler.

f. **Mikrokontroler ATmega32** : Berfungsi sebagai pengolah data dari setiap potongan *puzzle* yang dipasang untuk dibandingkan dengan *database* yang berada di mikrokontroler. Mikrokontroler ATmega32 memperoleh data setiap potongan *puzzle* dari rangkaian selektor data. Setelah data setiap potongan *puzzle* diperoleh, selanjutnya proses perbandingan data untuk menentukan benar atau tidak setiap potongan *puzzle* yang dipasang. Selain itu mikrokontroler melakukan tugas tugas lainnya yaitu menampilkan tulisan di layar LCD, menyalakan LED petunjuk kesalahan dan memberikan instruksi kepada blok modul *audio* untuk memutar data *audio*.

g. **Rangkaian LED Petunjuk Kesalahan** : Berfungsi sebagai petunjuk kesalahan pemasangan *puzzle*. Mikrokontroler dapat mengatur penyalaan setiap LED untuk memberikan informasi jumlah dan posisi setiap potongan *puzzle* yang dipasang pada tempat yang salah.

h. ***Push Button*** : Berfungsi untuk memberikan perintah kepada mikrokontroler untuk melakukan tugas tugas tertentu. Tugas-tugas tersebut diantaranya memilih surat al-Qur'an, memilih mode permainan dan memberikan perintah cek *puzzle*.

i. **LCD 16x4** : Berfungsi untuk menampilkan informasi berupa tulisan yang diperlukan oleh pengguna. Tulisan yang ditampilkan adalah berupa informasi daftar surat al-Qur'an, daftar mode permainan, waktu dan *skor.*

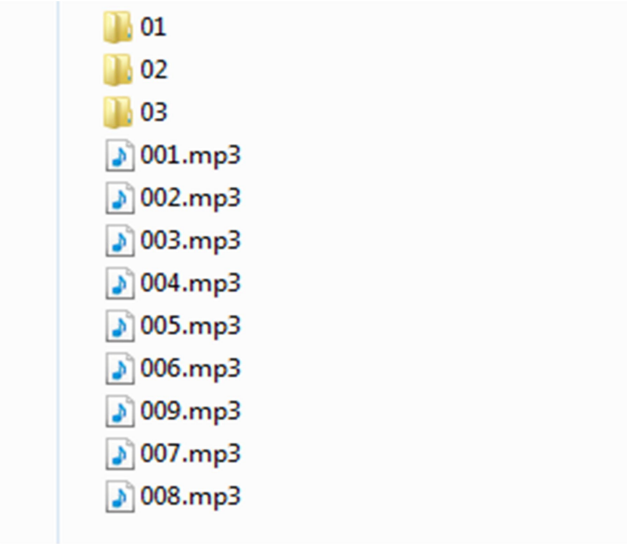
3.1 Perancangan Perangkat Keras

Perancangan perangkat keras terdiri dari penjelasan setiap perangkat keras, konfigurasi perangkat keras dan fungsi perangkat keras yang digunakan pada tugas akhir ini. Adapun perangkat keras yang dipakai pada tugas akhir ini adalah sebagai berikut :

3.1.1 Memori Eksternal TF *Card*

Memori eksternal TF *Card* berfungsi untuk menyimpan rekaman suara yang disimpan dalam bentuk data *audio digital.* Jumlah data yang dapat disimpan pada TF *Card* tergantung dari nilai *byte* dari data *audio digital* yang disimpan dan kapasitas memori TF *Card*. Cara membaca data dari TF *Card* dengan cara menghubungkan TF *Card* dengan *socket* modul *audio* DFR0299.

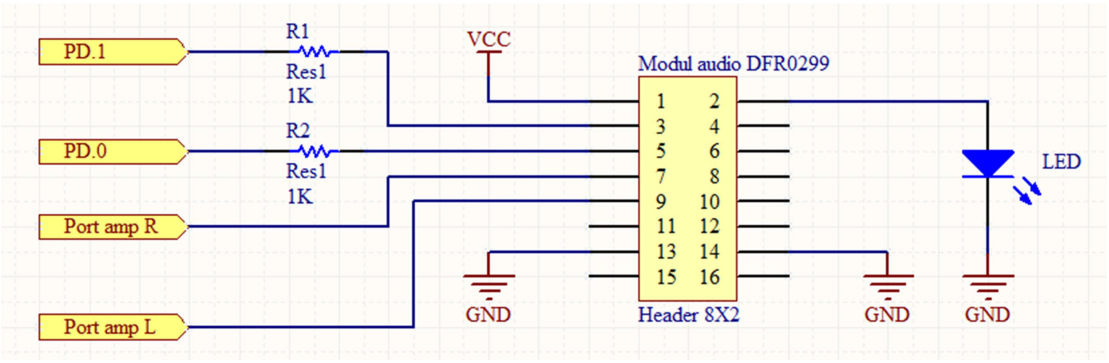
Data rekaman suara yang disimpan pada TF *Card* berupa data *audio digital* berformat mp3. Agar dapat dibaca oleh modul *audio* DFR0299, sistem penamaan folder dan data mp3 harus mengikuti sistem penamaan modul yang dimengerti *audio* DFR0299. Gambar 3.2 adalah contoh penamaan data pada TF *Card*.



Gambar 3.2 Penamaan data pada TF Card

3.1.2 Modul *Audio* DFR0299

Modul *audio* DFR0299 berfungsi untuk mengambil data *audio digital* berformat mp3 pada memori TF *Card* dan mengirimkannya ke modul *amplifier* untuk dikeluarkan menjadi data berbentuk suara. Agar modul *audio* dapat menjalankan fungsinya maka diperlukan perintah berupa kode program. Kode program yang diterima oleh modul *audio* dikirim oleh mikrokontroler melalui pengiriman data secara *serial*. Gambar 3.3 adalah skematik rangkaian modul *audio* DFR0299



Gambar 3.3 Skematik rangkaian modul audio DFR0299

Pin yang digunakan untuk komunikasi data secara *serial* adalah pin 3 sebagai pin penerima (RX) dan pin 5 sebagai pin pengirim (TX). Terdapat penambahan *resistor* 1 kilo ohm yang dihubungkan secara seri pada pin 3 dan pin 5 untuk menangani perbedaan level tegangan antara modul *audio* DFR0299 dengan mikrokontroler. Pin 7 dan pin 9 adalah pin keluaran yang dihubungkan dengan port pada *speaker*. Pin 2 dapat dihubungkan dengan LED sebagai informasi proses yang sedang terjadi pada modul *audio* DFR0299.

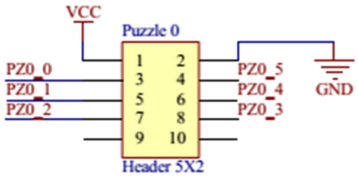
Contoh kode program untuk modul *audio* DFR0299 adalah sebagai berikut :

Contoh program :

i) Memutar data *audio* yang ke 100, 7E FF 06 03 00 00 64 FF E6 EF
ii) Menentukan nilai volume, 7E FF 06 06 00 00 0F FF D5 EF
iii) Memutar 001.mp3 pada folder 01, 7E FF 06 0F 00 01 01 EF

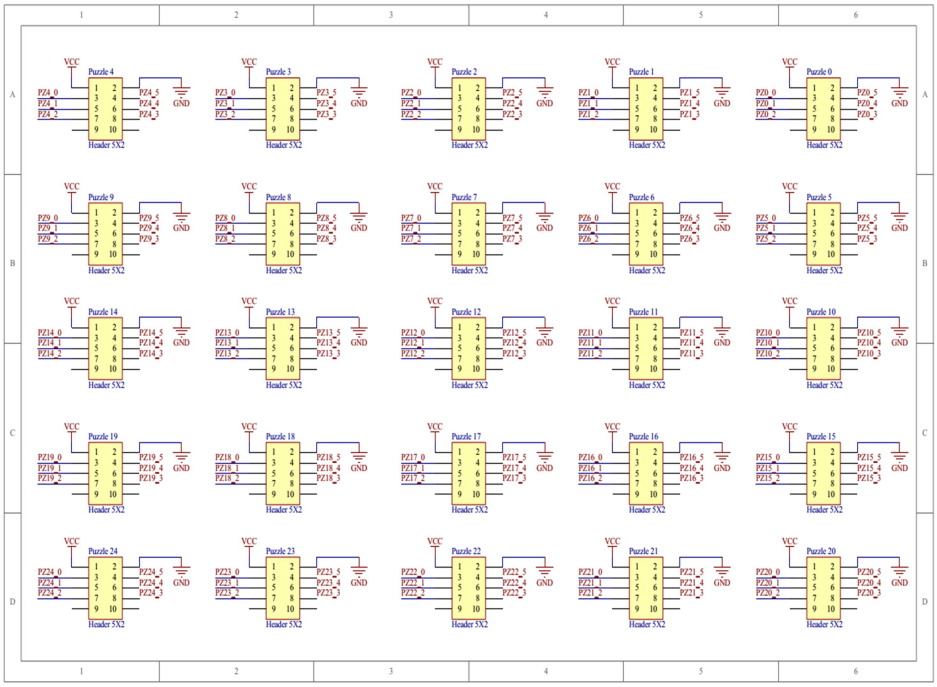
3.1.3 *Header Puzzle*

Header puzzle terdiri dari rangkaian listrik yang berfungsi untuk menghasilkan data potongan *puzzle*. *Header puzzle* menghasilkan 6 bit data yang berbeda-beda untuk setiap potongan *puzzle*. Setiap *header puzzle* terdiri dari 8 pin yaitu 2 pin catu daya dan 6 pin masukan data untuk dikirimkan kepada rangkaian selektor data. 2 pin catu daya terdiri dari catu daya 5 volt dan catu daya 0 volt *(ground)*. Fungsi dari pin catu daya adalah untuk menghasilkan 6 bit data yang berbeda-beda. Ketika pin masukan data dihubungkan dengan catu daya 5 volt maka itu menghasilkan data biner 1. Sebaliknya, ketika pin masukan data dihubungkan dengan catu daya 0 volt maka itu menghasilkan data biner 0. Gambar 3.4 adalah skematik dari rangkaian satu *header puzzle*



Gambar 3.4 Satu potong header puzzle

Gambar 3.5 adalah skematik keseluruhan dari rangkaian *header puzzle*

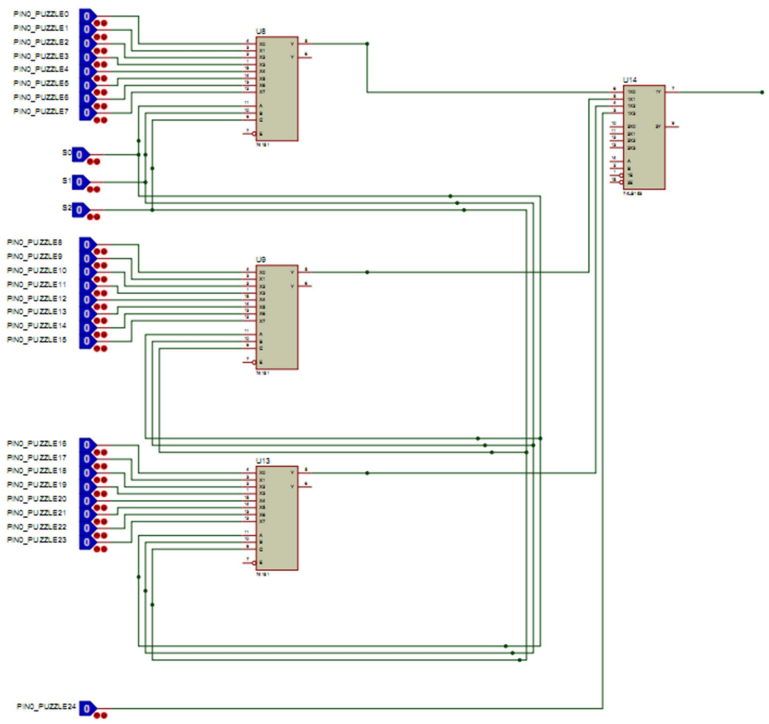


Gambar 3.5 Skematik keseluruhan header puzzle

3.1.4 Rangkaian Selektor Data

Rangkaian selektor data berfungsi sebagai media yang menjembatani antara rangkaian *header puzzle* dengan mikrokontroler. Melalui rangkaian selektor data, 150 saluran data dari *header puzzle* dapat disederhanakan menjadi 6 saluran data untuk dikirimkan kepada mikrokontroler. Rangkaian selektor data terdiri dari 18 IC *multiplexer* 8 ke 1, 3 IC *dual multiplexer* 4 ke 1 dan 1 IC *decoder* yang dibentuk dalam satu rangkaian. Rangkaian selektor data memiliki 5 pin masukan dan 6 pin keluaran. Pin masukan digunakan oleh mikrokontroler untuk memilih posisi potongan *puzzle* tertentu untuk dilakukan pengambilan data. Pin keluaran digunakan untuk mengirimkan data dari setiap potongan *puzzle* kepada mikrokontroler.

Rangkaian selektor data terdiri dari 6 rangkaian *multiplexer* 25 ke 1. Untuk menguji kevalidan dari rangkaian selektor data, dilakukan beberapa simulasi dengan perangkat lunak proteus. Gambar 3.6 adalah skematik rangkaian *multiplexer* 25 ke 1.



Gambar 3.6 Rangkaian multiplexer 25 ke 1

Secara keseluruhan rangkaian selektor data terdiri dari 6 rangkaian *multiplexer* 25 ke 1. Nilai 25 menunjukkan jumlah keseluruhan potongan *puzzle*. Setiap rangkaian *multiplexer* 25 ke 1 berfungsi untuk mengalirkan 1 pin data dari setiap potongan *puzzle*. Sesuai dengan jumlah 6 pin pada setiap potongan *puzzle* maka diperlukan 6 rangkaian *multiplexer* 25 ke 1. Pembacaan data dari potongan *puzzle* dilakukan secara berurutan dengan cara memberikan cacahan naik nilai biner pada pin masukan selektor data. Tabel 3.1 adalah tabel untuk menunjukkan nilai biner masukan selektor data dan potongan *puzzle* yang dipilih untuk diambil datanya.

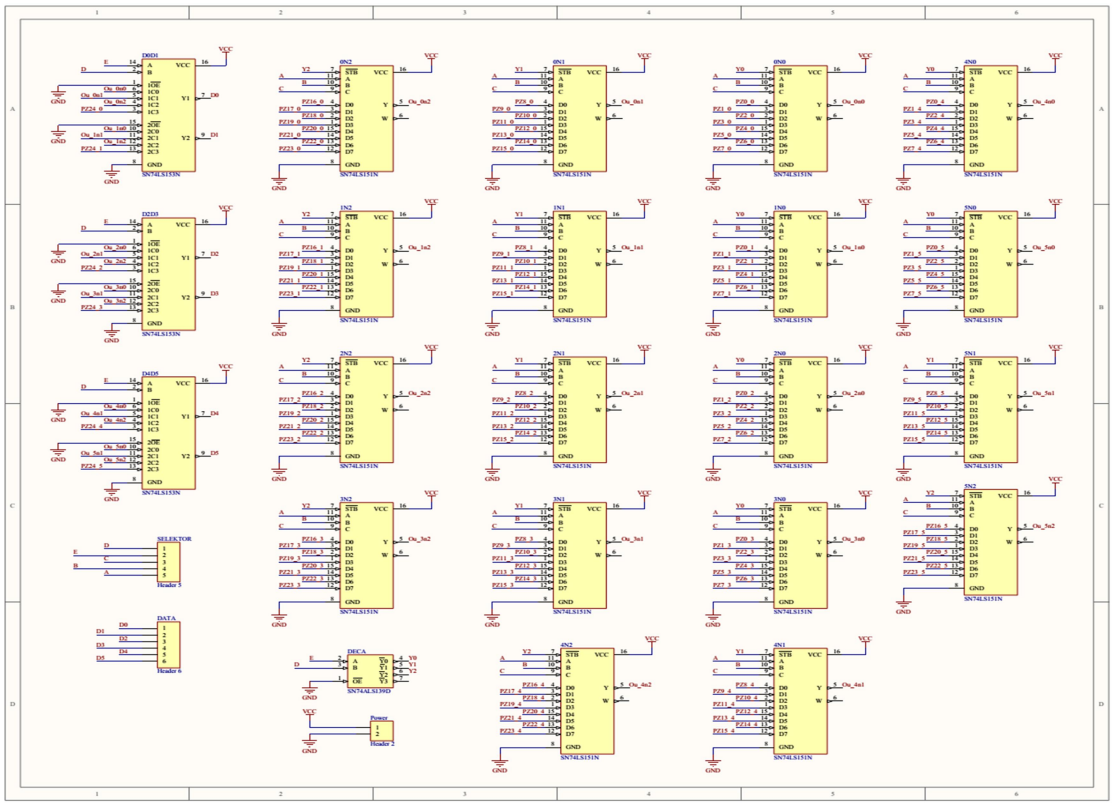
Tabel 3.1 Nilai biner selektor data untuk memilih potongan puzzle

No	Pin Selektor					Potongan *Puzzle*
	0	1	2	3	4	
1	0	0	0	0	0	*Puzzle 0*
2	0	0	0	0	1	*Puzzle 1*
3	0	0	0	1	0	*Puzzle 2*
4	0	0	0	1	1	*Puzzle 3*
5	0	0	1	0	0	*Puzzle 4*
6	0	0	1	0	1	*Puzzle 5*
7	0	0	1	1	0	*Puzzle 6*
8	0	0	1	1	1	*Puzzle 7*
9	0	1	0	0	0	*Puzzle 8*
10	0	1	0	0	1	*Puzzle 9*
11	0	1	0	1	0	*Puzzle 10*
12	0	1	0	1	1	*Puzzle 11*
13	0	1	1	0	0	*Puzzle 12*
14	0	1	1	0	1	*Puzzle 13*
15	0	1	1	1	0	*Puzzle 14*
16	0	1	1	1	1	*Puzzle 15*
17	1	0	0	0	0	*Puzzle 16*
18	1	0	0	0	1	*Puzzle 17*
19	1	0	0	1	0	*Puzzle 18*
20	1	0	0	1	1	*Puzzle 19*
21	1	0	1	0	0	*Puzzle 20*

Tabel lanjutan

22	1	0	1	0	1	*Puzzle 21*
23	1	0	1	1	0	*Puzzle 22*
24	1	0	1	1	1	*Puzzle 23*
25	1	1	0	0	0	*Puzzle 24*

Gambar 3.7 adalah skematik keseluruhan dari rangkaian selektor data



Gambar 3.7 Skematik keseluruhan dari rangkaian selektor data

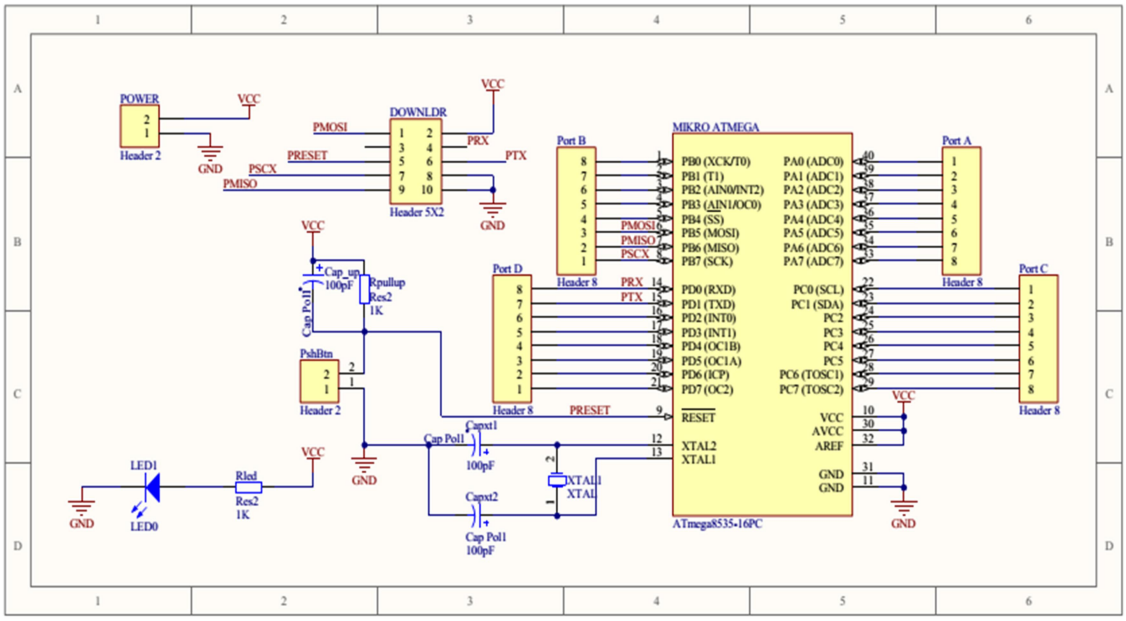
3.1.5 Mikrokontroler ATmega32

Mikrokontroler ATmega32 adalah unit untuk pemrosesan data. Pada tugas akhir ini, mikrokontroler memiliki beberapa tugas khusus, diantaranya ;

1. Mengatur tampilan tulisan pada layar LCD
2. Mengatur pewaktuan dalam satuan menit, satuan detik dan satuan milidetik.
3. Membaca penekanan tombol *push button*.

4. Memberikan instruksi kepada rangkaian selektor data.
5. Membaca data setiap potongan *puzzle*.
6. Menentukan benar atau tidak setiap potongan *puzzle* yang dipasang.
7. Mengatur penyalaan LED petunjuk kesalahan.
8. Memberikan instruksi kepada modul *audio* untuk memutar data *audio*.

Gambar 3.8 adalah skematik dari rangkaian mikrokontroler ATmega32



Gambar 3.8 Skematik sistem minimum mikrokontroler ATmega32

Tabel 3.2 adalah tabel alokasi setiap pin mikrokontroler ATmega32

Tabel 3.2 Alokasi port mikrokontroler ATmega32

PORT	PIN	TIPE	FUNGSI
A	0	*Output*	Memberikan data ke 0 kepada rangkaian selektor data
	1	*Output*	Memberikan data ke 1 kepada rangkaian selektor data
	2	*Output*	Memberikan data ke 2 kepada rangkaian selektor data
	3	*Output*	Memberikan data ke 3 kepada rangkaian selektor data
	4	*Output*	Memberikan data ke 4 kepada rangkaian selektor data
	5	*Input*	Menerima data ke 0 dari rangkaian selektor data
	6	*Input*	Menerima data ke 1 dari rangkaian selektor data
	7	*Input*	Menerima data ke 2 dari rangkaian selektor data

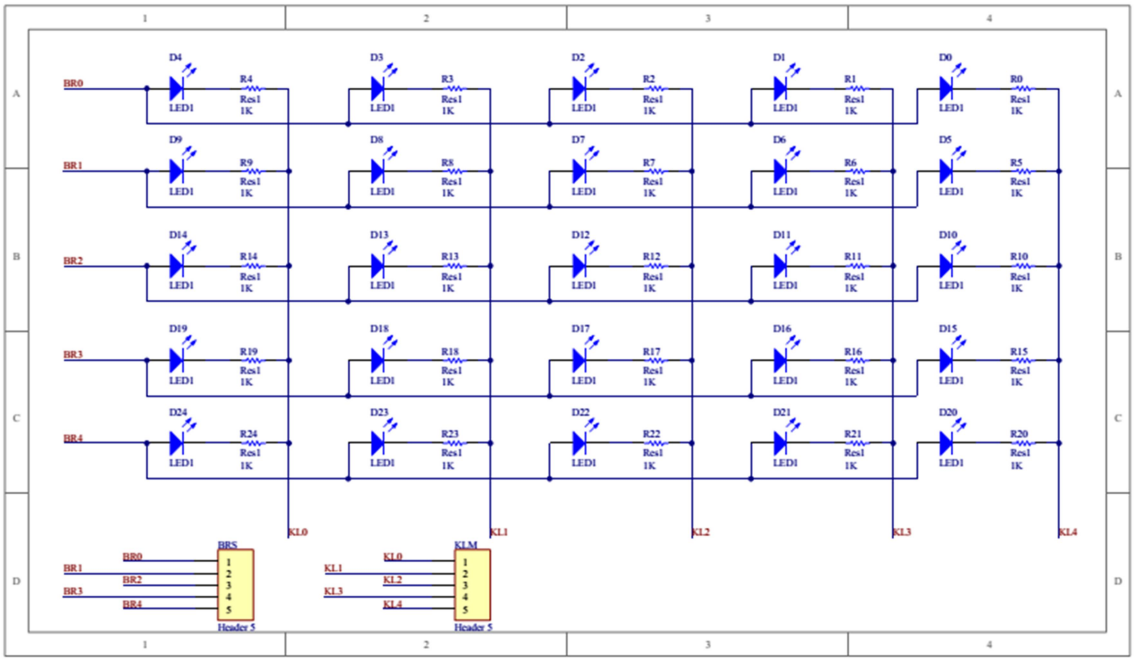
Tabel lanjutan

B	0	*Output*	Data untuk kolom ke 0 pada rangkaian LED
	1	*Output*	Data untuk kolom ke 1 pada rangkaian LED
	2	*Output*	Data untuk kolom ke 2 pada rangkaian LED
	3	*Output*	Data untuk kolom ke 3 pada rangkaian LED
	4	*Output*	Data untuk kolom ke 4 pada rangkaian LED
	5	*Output*	Data untuk baris ke 0 pada rangkaian LED
	6	*Output*	Data untuk baris ke 1 pada rangkaian LED
	7	*Output*	Data untuk baris ke 2 pada rangkaian LED
C	0	*Output*	D5 LCD
	1	*Output*	D6 LCD
	2	*Output*	D7 LCD
	3	*Input*	Tombol *push button* 1
	4	*Input*	Tombol *push button* 2
	5	*Input*	Menerima data 3 dari rangkaian selektor data
	6	*Input*	Menerima data 4 dari rangkaian selektor data
	7	*Input*	Menerima data 5 dari rangkaian selektor data
D	0	*Input*	RX
	1	*Output*	TX
	2	*Not connected*	
	3	*Output*	Data untuk baris ke 3 pada rangkaian LED
	4	*Output*	Data untuk baris ke 4 pada rangkaian LED
	5	*Output*	RS LCD
	6	*Output*	EN LCD
	7	*Output*	D4 LCD

3.1.6 Rangkaian LED Petunjuk Kesalahan

Rangkaian LED petunjuk kesalahan berfungsi untuk menunjukkan benar atau tidak setiap potongan *puzzle* yang dipasang. Ketika LED dalam keadaan menyala, itu menunjukkan bahwa potongan *puzzle* dipasang pada tempat yang salah. Sebaliknya ketika LED dalam keadaan padam, itu menunjukkan bahwa potongan *puzzle* dipasang pada tempat yang benar.

Jumlah dan posisi LED disesuaikan dengan jumlah dan posisi potongan *puzzle*, sehingga dengan melihat LED menyala maka jumlah dan posisi potongan *puzzle* yang dipasang pada tempat yang salah dapat diketahui. Gambar 3.9 adalah skematik dari rangkaian LED petunjuk kesalahan.



Gambar 3.9 Rangkaian LED petunjuk kesalahan

Jumlah LED yang digunakan pada rangkaian LED petunjuk kesalahan berjumlah 25. LED disusun sehingga membentuk susunan kolom dan baris. Pada LED petunjuk kesalahan ini terdiri dari 5 kolom dan 5 baris dimana setiap baris dan kolom terdiri dari 5 LED. Penyalaan LED diatur oleh mikrokontroler melalui port mikrokontroler yang dihubungkan dengan kolom dan baris pada rangkaian LED petunjuk kesalahan. Untuk mengatur penyalaan LED, mikrokontroler menyediakan 10 saluran melalui Port B dan Port D.

Pada Tabel 3.3 adalah tabel untuk menunjukkan alokasi port mikrokontroler yang digunakan untuk penyalaan LED petunjuk kesalahan.

Tabel 3.3 Alokasi port mikrokontroler untuk rangkaian LED petunjuk kesalahan

No	Port Mikrokontroler	PIN LED
1	PB0	KOLOM 0
2	PB1	KOLOM 1
3	PB2	KOLOM 2
4	PB3	KOLOM 3
5	PB4	KOLOM 4
6	PB5	BARIS 0
7	PB6	BARIS 1
8	PB7	BARIS 2
9	PD3	BARIS 3
10	PD4	BARIS 4

Terdapat dua perbedaan mendasar mengenai cara untuk mengatur penyalaan LED. Perbedaan terjadi karena pada mode pandu dan mode lomba membutuhkan sistem penyalaan LED yang berbeda.

3.1.6.1 Sistem penyalaan LED mode lomba

1. Pada kondisi awal semua LED diatur dalam kondisi padam
2. Ketika semua *puzzle* selesai terpasang dan dilakukan pengecekkan, maka LED mulai dinyalakan.
3. Ketika lebih dari satu LED yang harus dinyalakan, maka LED dinyalakan satu persatu secara bergantian dalam waktu yang cepat sehingga terlihat dinyalakan bersamaan. Jumlah LED yang dinyalakan disesuaikan dengan jumlah potongan *puzzle* yang dipasang pada tempat yang salah.

Terdapat perbedaan kondisi pada port mikrokontroler ketika mikrokontroler menyalakan LED yang berbeda. Tabel 3.4 adalah tabel untuk menunjukkan perbedaan kondisi port dalam penyalaan LED di mode lomba.

Tabel 3.4 Kondisi port mikrokontroler dalam penyalaan LED mode lomba

NO	LED NYALA	PORTB								PORTD	
		PB0	PB1	PB2	PB3	PB4	PB5	PB6	PB7	PD3	PD4
1	0	1	1	0	1	0	0	0	0	1	1
2	1	1	1	0	0	1	0	0	0	1	1
3	2	1	1	0	0	0	1	0	0	1	1
4	3	1	1	0	0	0	0	1	0	1	1
5	4	1	1	0	0	0	0	0	1	1	1
6	5	1	0	1	1	1	0	1	1	1	1
7	6	1	0	1	1	1	0	1	1	1	1
8	7	1	0	1	1	1	0	1	1	1	1
9	8	1	0	1	1	1	0	1	1	1	1
10	9	1	0	1	1	1	0	1	1	1	1
11	10	0	1	1	1	1	0	1	1	1	1
12	11	0	1	1	1	1	0	1	1	1	1
13	12	0	1	1	1	1	0	1	1	1	1
14	13	0	1	1	1	1	0	1	1	1	1
15	14	0	1	1	1	1	0	1	1	1	1
16	15	1	1	1	0	0	0	0	0	1	0
17	16	1	1	1	0	0	0	0	0	1	0
18	17	1	1	1	0	0	0	0	0	1	0
19	18	1	1	1	0	0	0	0	0	1	0
20	19	1	1	1	0	0	0	0	0	1	0
21	20	1	1	1	0	0	0	0	0	0	1
22	21	1	1	1	0	0	0	0	0	0	1
23	22	1	1	1	0	0	0	0	0	0	1
24	23	1	1	1	0	0	0	0	0	0	1
25	24	1	1	1	0	0	0	0	0	0	1

3.1.6.2 Sistem penyalaan LED mode pandu

1. Pada kondisi awal semua LED diatur dalam kondisi nyala
2. Pada proses selanjutnya, LED dipadamkan satu persatu secara berurutan mengikuti pemasangan potongan *puzzle* pada posisi benar. Jumlah LED yang dipadamkan bertambah sesuai dengan bertambahnya jumlah potongan *puzzle* yang dipasang.

Terdapat perbedaan kondisi pada port mikrokontroler ketika mikrokontroler memadamkan LED. Dalam memadamkan LED, port mikrokontroler mengalami dua proses. Pertama mikrokontroler memberikan bit data tertentu kepada port yang dihubungkan dengan saluran kolom kemudian memberikan bit data tertentu lainnya kepada port yang dihubungkan dengan saluran baris. Tabel 3.5 adalah tabel untuk menunjukkan perbedaan kondisi port dalam penyalaan LED di mode pandu.

Tabel 3.5 Kondisi port mikrokontroler dalam penyalaan LED mode pandu

NO	LED PADAM	PORTB									PORTD		
		Proses	PB0	PB0	PB0	PB0	PB0	PB0	PB0	PB0	Proses	PD3	PD4
1	0	P1	0	0	0	0	1	1	1	1	P1	0	0
		P2	0	0	1	1	1	1	1	1	P2	0	0
2	0,1	P1	0	0	0	0	0	1	1	1	P1	0	0
		P2	0	0	1	1	1	1	1	1	P2	0	0
3	0,1,2,	P1	0	0	0	0	0	0	1	1	P1	0	0
		P2	0	0	1	1	1	1	1	1	P2	0	0
4	0,1,2,3	P1	0	0	0	0	0	0	1	1	P1	0	0
		P2	0	0	1	1	1	1	1	1	P2	0	0
5	0,1,2,3,4,	P1	0	0	0	0	0	0	0	1	P1	0	0
		P2	0	0	1	1	1	1	1	1	P2	0	0
6	0,1,2,3,4,5	P1	0	0	1	0	1	1	1	1	P1	0	0
		P2	0	1	1	1	1	1	1	1	P2	0	0
7	0,1,2,3,4,5,6,	P1	0	0	1	0	0	1	1	1	P1	0	0
		P2	0	1	1	1	1	1	1	1	P2	0	0
8	0,1,2,3,4,5,6,7,	P1	0	0	1	0	0	0	1	1	P1	0	0
		P2	0	1	1	1	1	1	1	1	P2	0	0

Tabel lanjutan

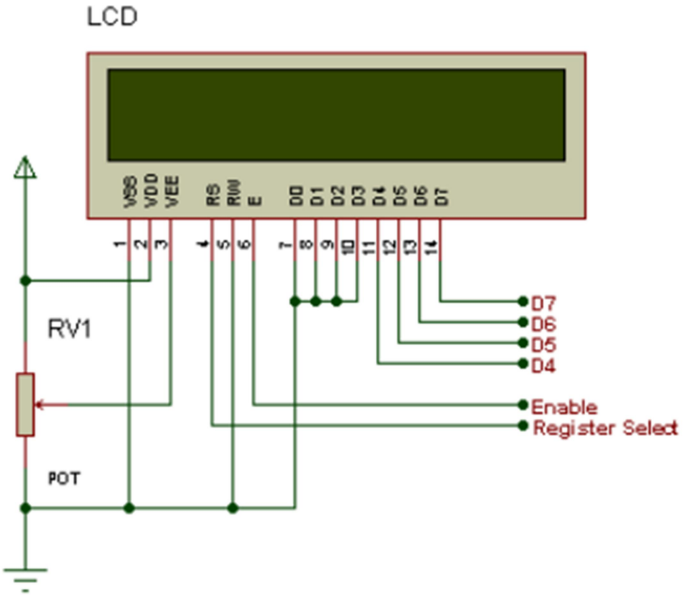
9	0,1,2,3,4,5,6,7,8,	P1	0	0	1	0	0	0	1	1	P1	0	0
		P2	0	1	1	1	1	1	1	1	P2	0	0
10	0,1,2,3,4,5,6,7,8,9,	P1	0	0	1	0	0	0	0	1	P1	0	0
		P2	0	1	1	1	1	1	1	1	P2	0	0
11	0,1,2,3,4,5,6,7,8,9,10	P1	0	1	1	0	1	1	1	1	P1	0	0
		P2	1	1	1	1	1	1	1	1	P2	0	0
12	0,1,2,3,4,5,6,7,8,9,10,11,	P1	0	1	1	0	0	1	1	1	P1	0	0
		P2	1	1	1	1	1	1	1	1	P2	0	0
13	0,1,2,3,4,5,6,7,8,9,10,11, 12	P1	0	1	1	0	0	0	1	1	P1	0	0
		P2	1	1	1	1	1	1	1	1	P2	0	0
14	0,1,2,3,4,5,6,7,8,9,10,11, 12,13,	P1	0	1	1	0	0	0	1	1	P1	0	0
		P2	1	1	1	1	1	1	1	1	P2	0	0
15	0,1,2,3,4,5,6,7,8,9,10,11, 12,13,14	P1	1	1	1	0	0	0	0	1	P1	0	0
		P2	1	1	1	1	1	1	1	1	P2	0	0
16	0,1,2,3,4,5,6,7,8,9,10,11, 12,13,14,15,	P1	1	1	1	0	1	1	1	1	P1	0	0
		P2	1	1	1	1	1	1	1	1	P2	0	1
17	0,1,2,3,4,5,6,7,8,9,10,11, 12,13,14,15,16,	P1	1	1	1	0	0	1	1	1	P1	0	0
		P2	1	1	1	1	1	1	1	1	P2	0	1
18	0,1,2,3,4,5,6,7,8,9,10,11, 12,13,14,15,16,17	P1	1	1	1	0	0	0	1	1	P1	0	0
		P2	1	1	1	1	1	1	1	1	P2	0	1
19	0,1,2,3,4,5,6,7,8,9,10,11, 12,13,14,15,16,17,18,	P1	1	1	1	0	0	0	1	1	P1	0	0
		P2	1	1	1	1	1	1	1	1	P2	0	1
20	0,1,2,3,4,5,6,7,8,9,10,11, 12,13,14,15,16,17,18,19	P1	1	1	1	0	0	0	0	1	P1	0	0
		P2	1	1	1	1	1	1	1	1	P2	0	1
21	0,1,2,3,4,5,6,7,8,9,10,11, 12,13,14,15,16,17,18,19, 20	P1	1	1	1	0	1	1	1	1	P1	0	1
		P2	1	1	1	1	1	1	1	1	P2	1	1
22	0,1,2,3,4,5,6,7,8,9,10,11, 12,13,14,15,16,17,18,19, 20,21,	P1	1	1	1	0	0	1	1	1	P1	0	1
		P2	1	1	1	1	1	1	1	1	P2	1	1
23	0,1,2,3,4,5,6,7,8,9,10,11, 12,13,14,15,16,17,18,19, 20,21,22	P1	1	1	1	0	0	0	1	1	P1	0	1
		P2	1	1	1	1	1	1	1	1	P2	1	1
24	0,1,2,3,4,5,6,7,8,9,10,11, 12,13,14,15,16,17,18,19, 20,21,22,23	P1	1	1	1	0	0	0	1	1	P1	0	1
		P2	1	1	1	1	1	1	1	1	P2	1	1

Tabel lanjutan

25	0,1,2,3,4,5,6,7,8,9,10,11, 12,13,14,15,16,17,18,19, 20,21,22,23,24	P1	1	1	1	0	0	0	0	1	P1	0	1
		P2	1	1	1	1	1	1	1	1	P2	1	1

3.1.7 LCD 16x4

LCD 16x4 berfungsi untuk menampilkan informasi berupa tulisan. Melalui tulisan pada layar LCD pengguna dapat mengetahui daftar surat al-Qur'an, mode permainan, pewaktuan dan skor. LCD 16x4 memiliki kemampuan untuk menampilkan karakter huruf sebanyak 64 karakter. 64 karakter huruf ini terbagi kedalam 4 baris, sehingga masing-masing baris mampu menampilkan 16 karakter. Gambar 3.10 adalah gambar rangkaian LCD dan konfigurasi pinnya



Gambar 3.10 LCD dan konfigurasi pinnya

Berikut daftar tulisan yang ditampilkan pada LCD

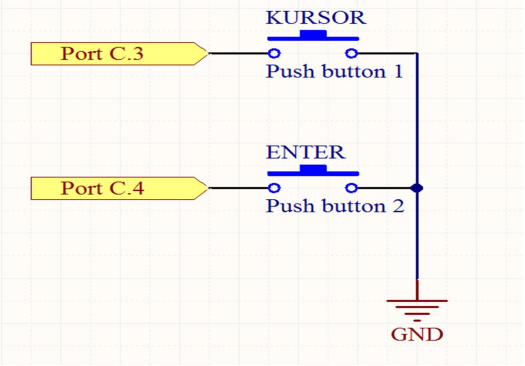
1. Selamat datang di *Puzzle* Al-Qur'an
   Mari Mencoba !
2. Silahkan pilih surat !
   a. An-Naas
   b. Al-Falaq
   c. Al-Ikhlas

3. Silahkan pilih mode !
   a. Mode Lomba
   b. Mode Pandu
4. Waktu : xx : xx : xx
5. Skor : xxxx
   Jumlah *puzzle* yang salah : xx
6. Terimakasih Adik

3.1.8 Tombol *Push Button*

Tombol *push button* berfungsi untuk memberikan perintah kepada mikrokontroler untuk menjalankan proses tertentu. Melalui tombol *push button* pengguna dapat memilih surat, memilih mode permainan dan memberikan perintah pengecekkan *puzzle* kepada mikrokontroler. Pada tugas akhir ini, digunakan dua tombol *push button*. Tombol *push button* 1 berfungsi sebagai eksekutor atau pengambil keputusan. Sedangkan tombol *push button* 2 berfungsi untuk memindahkan kursor ketika dilakukan pemilihan menu baik pemilihan surat al-Qur'an maupun pemilihan mode permainan.

Pada port C khusus pin 3 dan 4 digunakan untuk tombol *push button.* Gambar 3.11 adalah skematik untuk menunjukkan bagaimana tombol *push button* terhubung dengan pin mikrokontroler.



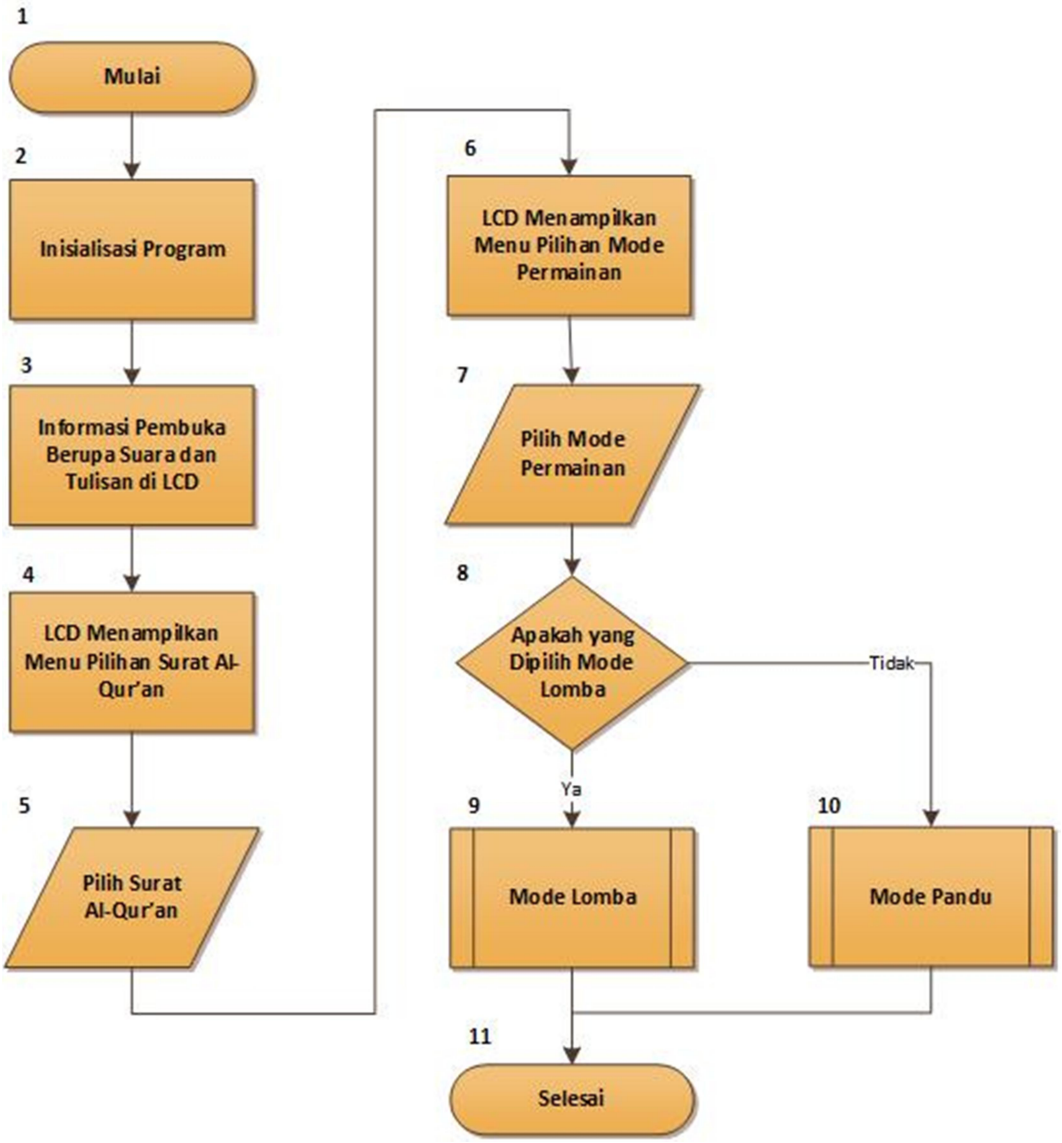
Gambar 3.11 Alokasi port mikrokontroler untuk push button

3.2 Perancangan Perangkat Lunak

Perancangan perangkat lunak terdiri dari beberapa diagram alir untuk menggambarkan bagaimana program berjalan pada mikrokontroler. Berikut adalah penjelasan mengenai semua diagram alir yang digunakan pada tugas akhir ini.

3.2.1 Diagram Alir Program Utama

Diagram alir program utama adalah diagram untuk menjelaskan secara menyeluruh mengenai proses yang terjadi di mikrokontroler. Gambar 3.12 adalah skema dari diagram alir program utama.



Gambar 3.12 Diagram alir program utama

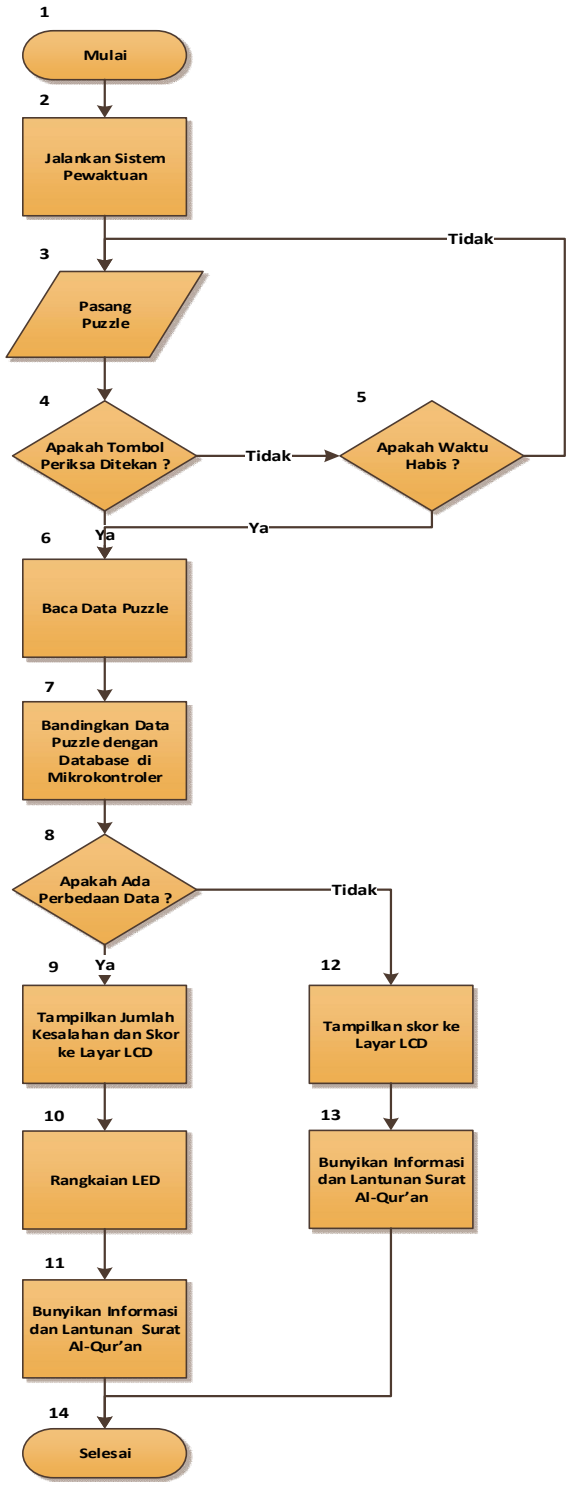
Tabel 3.6 adalah tabel untuk menjelaskan setiap proses yang terjadi pada program utama.

Tabel 3.6 Penjelasan diagram alir pada program utama

No	Keterangan
1	Awal program
2	Inisialisasi program
3	Perintah kepada modul *audio* untuk memutar *audio* pembuka dan perintah kepada LCD untuk menampilkan informasi pembuka
4	Perintah kepada LCD untuk menampilkan tulisan pilihan surat al-Qur'an
5	Pengguna memilih surat al-Qur'an
6	Perintah kepada LCD untuk menampilkan tulisan pilihan mode permainan
7	Pengguna memilih mode permainan
8	Membaca pilihan mode permainan untuk menentukan eksekusi program selanjutnya
9	Prosedur pada mode lomba
10	Prosedur pada mode pandu
11	Akhir program

3.2.2 Diagram Alir pada Mode Lomba

Diagram alir pada mode lomba terdiri dari beberapa proses seperti pewaktuan selama 10 menit, pembacaan data *puzzle*, perbandingan data *puzzle*, penyalaan LED dan pemutaran data suara oleh modul *audio*. Gambar 3.13 adalah skema dari diagram alir pada mode lomba.



Gambar 3.13 Diagram alir pada mode lomba

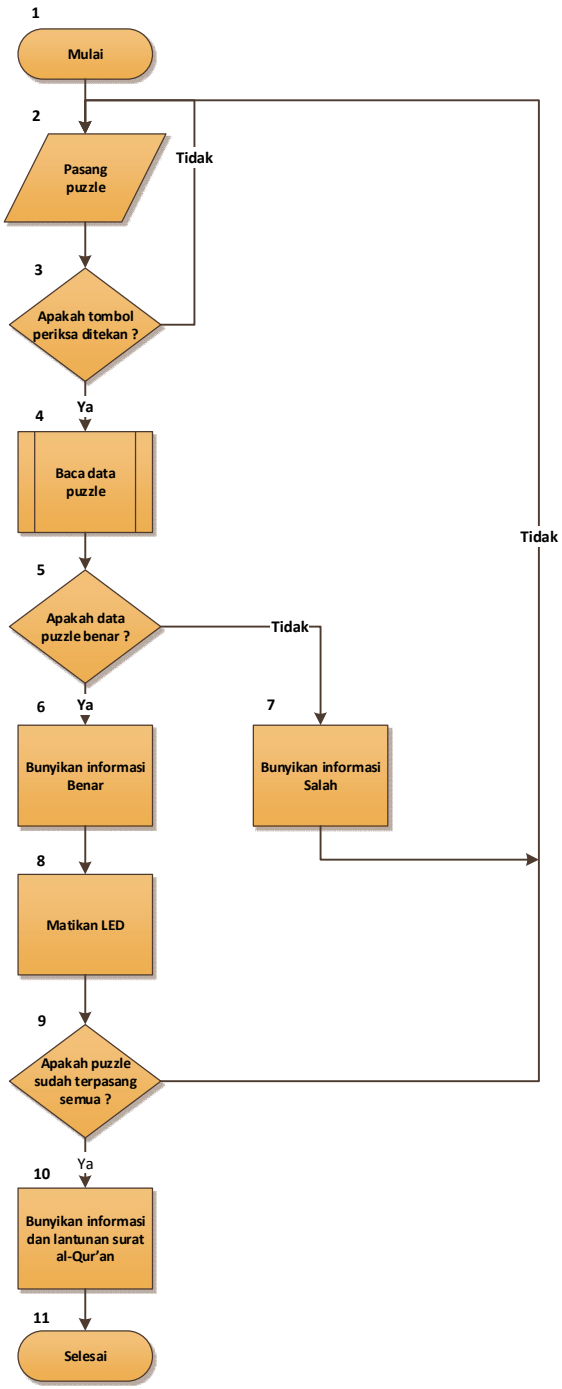
Tabel 3.7 adalah tabel untuk menjelaskan setiap proses yang terjadi pada program di mode lomba.

Tabel 3.7 Penjelasan diagram alir pada mode lomba

No	Keterangan
1	Awal program
2	Jalankan sistem pewaktuan
3	Pengguna memasang *puzzle*
4	Memeriksa penekanan tombol periksa data *puzzle*
5	Memeriksa status waktu untuk menentukan eksekusi program selanjutnya
6	Membaca data *puzzle* yang telah dipasang
7	Membandingkan data *puzzle* dengan data pada mikrokontroler
8	Memeriksa perbedaan data *puzzle* untuk menentukan eksekusi program selanjutnya
9	Perintah kepada LCD untuk menampilkan tulisan jumlah potongan *puzzle* yang salah dan skor yang didapatkan
10	Mengatur penyalaan LED petunjuk kesalahan
11	Perintah kepada modul *audio* untuk memutar *audio* informasi dan surat al-Qur'an
12	Perintah kepada LCD untuk menampilkan tulisan skor yang didapatkan
13	Perintah kepada modul *audio* untuk memutar *audio* informasi dan surat al-Qur'an
14	Akhir program

3.2.3 Diagram Alir pada Mode Pandu

Diagram alir pada mode pandu terdiri dari beberapa proses seperti pembacaan data *puzzle*, pengecekkan potongan *puzzle* yang terpasang, penyalaan LED dan pemutaran data suara oleh modul *audio*. Gambar 3.14 adalah skema dari diagram alir pada mode pandu.



Gambar 3.14 Diagram alir pada mode pandu

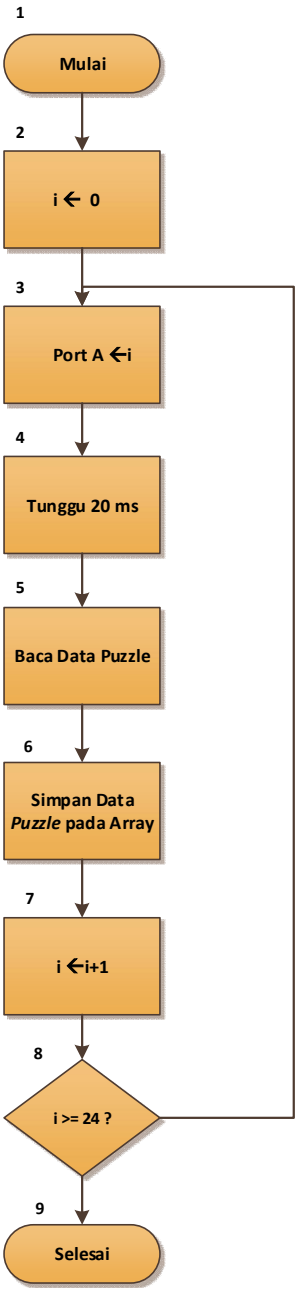
Tabel 3.8 adalah tabel untuk menjelaskan setiap proses yang terjadi pada program di mode pandu.

Tabel 3.8 Penjelasan diagram alir pada mode pandu

No	Keterangan
1	Awal program
2	Pengguna memasang potongan *puzzle*
3	Memeriksa penekanan tombol cek data potongan *puzzle*
4	Membaca data potongan *puzzle* yang telah dipasang
5	Memeriksa status benar atau tidak dari potongan *puzzle* yang dipasang
6	Perintah kepada modul *audio* untuk memutar informasi benar
7	Perintah kepada modul *audio* untuk memutar informasi salah
8	Mengatur penyalaan LED petunjuk kesalahan
9	Memeriksa apakah potongan *puzzle* sudah terpasang seluruhnya
10	Perintah kepada modul *audio* untuk memutar *audio* informasi dan surat al-Qur'an
11	Akhir program

3.2.4 Prosedur Baca Data *Puzzle* pada Mode Lomba

Diagram alir baca data *puzzle* pada mode lomba terdiri dari beberapa proses seperti pengisian nilai pada port A, pemberian *delay*, baca data potongan *puzzle* dan penyimpanan data *puzzle* di array pada mikrokontroler. Gambar 3.15 adalah skema dari diagram alir baca data *puzzle* pada mode lomba.



Gambar 3.15 Diagram alir baca data puzzle pada mode lomba

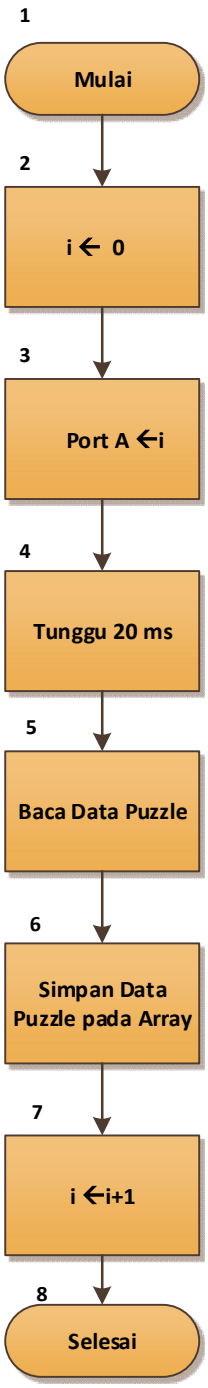
Tabel 3.9 adalah tabel untuk menjelaskan setiap proses yang terjadi pada program baca data *puzzle* di mode lomba.

Tabel 3.9 Penjelasan diagram alir baca data pada mode lomba

No	Keterangan
1	Awal program
2	Isi nilai awal variabel i dengan 0, variable i digunakan sebagai pencacah naik dan indeks array
3	Kirimkan nilai variabel i pada port A sebagai pencacah naik untuk mengatur rangkaian selektor data
4	Tahan proses program selama 20 milidetik
5	Baca data *puzzle* yang dikirimkan oleh rangkaian selektor data
6	Simpan data *puzzle* pada array
7	Berikan tambahan nilai pada variabel i untuk mencacah naik proses berikutnya
8	Memeriksa nilai i, proses berhenti ketika terjadi 25 pembacaan data *puzzle*
9	Akhir program

3.2.5 Prosedur Baca Data *Puzzle* pada Mode Pandu

Diagram alir baca data *puzzle* pada mode pandu terdiri dari beberapa proses seperti pengisian nilai pada port A, pemberian *delay*, baca data potongan *puzzle* dan penyimpanan data *puzzle* di array pada mikrokontroler. Gambar 3.16 adalah diagram alir setiap proses pada prosedur baca data *puzzle* di mode pandu.



Gambar 3.16 Diagram alir baca data puzzle pada mode pandu

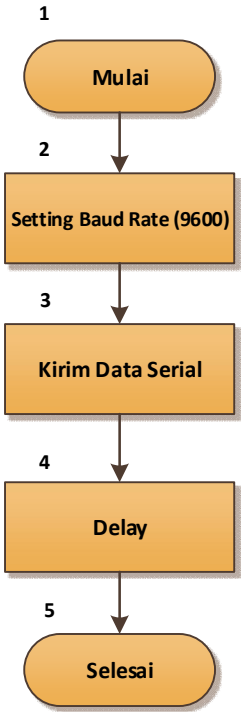
Tabel 3.10 adalah tabel untuk menjelaskan setiap proses yang terjadi pada program baca data *puzzle* di mode pandu.

Tabel 3.10 Penjelasan diagram alir baca data pada mode pandu

No	Keterangan
1	Awal program
2	Isi nilai awal variabel i dengan 0, variable i digunakan sebagai pencacah dan indeks pada array
3	Kirimkan nilai variabel i pada port A sebagai pencacah naik untuk mengatur rangkaian selektor data
4	Tahan proses program selama 20 milidetik
5	Baca data *puzzle* yang dikirimkan oleh rangkaian selektor data
6	Simpan data *puzzle* pada array
7	Berikan tambahan nilai pada variabel i untuk mencacah naik proses berikutnya
8	Akhir program

3.2.6 Prosedur Kirim Data ke Modul *Audio*

Diagram alir kirim data ke modul *audio* terdiri dari beberapa proses seperti inisialisasi nilai *baudrate*, kirim kode program secara *serial* dan pemberian *delay*. Gambar 3.17 adalah skema dari diagram alir kirim data ke modul *audio*.



Gambar 3.17 Diagram alir kirim data ke modul audio

Tabel 3.11 adalah tabel untuk menjelaskan setiap proses yang terjadi pada program kirim data ke modul *audio*.

Tabel 3.11 Penjelasan diagram alir kirim data ke modul audio

No	Keterangan
1	Awal program
2	Tentukan nilai baudrate menjadi 9600 bps
3	Kirimkan kode program secara serial kepada modul *audio*
4	Tunggu hingga data *audio* selesai diputar
5	Akhir program

3.3 Perancangan Mekanik

Perancangan mekanik terdiri dari perancangan bentuk *puzzle* dan kemasan alat secara keseluruhan.

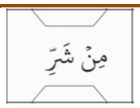
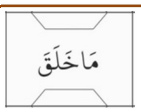
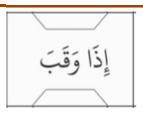
3.3.1 Perancangan *Puzzle*

Pada tugas akhir ini, setiap potongan *puzzle* dirancang untuk memiliki data biner yang berbeda. Hal ini dilakukan agar mikrokontroler dapat membedakan setiap potongan *puzzle* yang dipasang. Jumlah perbedaan nilai biner potongan *puzzle* keseluruhan surat adalah 39 nilai biner. Setiap potongan *puzzle* memiliki nilai biner yang dipresentasikan kedalam 6 bit data. Tabel 3.12 adalah tabel untuk menjelaskan potongan *puzzle* beserta nilai binernya.

Tabel 3.12 Potongan puzzle beserta nilai binernya

No	*Puzzle*	Nilai Biner	No	*Puzzle*	Nilai Biner	No	*Puzzle*	Nilai Biner
1	قُلْ	000001	14	فِي	001110	27	يُوَسْوِسُ	011011
2	أَعُوذُ	000010	15	الْعُقَدِ	001111	28	صُدُورِ	011100
3	بِرَبِّ	000011	16	٤	010000	29	مِنَ الْجِنَّةِ	011101
4	الْفَلَقِ	000100	17	حَاسِدٍ	010001	30	٦	011110
5	١	000101	18	إِذَا حَسَدَ	010010	31	هُوَ	011111

Tabel lanjutan

6		000110	19		010011	32		100000
7		000111	20		010100	33		100001
8		001000	21		010101	34		100010
9		001001	22		010110	35		100011
10		001010	23		010111	36		100100
11		001011	24		011000	37		100101
12		001100	25		011001	38		100110
13		001101	26		011010	39		100111

Pada tugas akhir ini surat al-Qur'an yang dapat dipilih terdiri dari 3 surat yaitu surat an-Naas, al-Falaq dan al-Ikhlas. Setiap surat dipecah kedalam 25 bagian sehingga jumlah keseluruhan potongan *puzzle* dalam 1 surat adalah 25 potongan *puzzle*. Gambar 3.18 adalah tampilan susunan nilai biner untuk *puzzle* surat an-Naas.

000101	010101	000011	000010	000001
010101	010111	001000	010101	010110
010000	011001	011000	000110	001100
010101	011100	001110	011011	011010
011110	010101	001001	011101	010011

Gambar 3.18 Susunan nilai biner surat an-Naas

Gambar 3.19 adalah gambar potongan *puzzle* secara lengkap untuk surat an-Naas.

١	النَّاسِ	بِرَبِّ	أَعُوْذُ	قُلْ
النَّاسِ	إِلَٰهِ	٢	النَّاسِ	مَلِكِ
٤	الْخَنَّاسِ	الْوَسْوَاسِ	مِنْ شَرِّ	٣
النَّاسِ	صُدُوْرِ	فِيْ	يُوَسْوِسُ	الَّذِ يْ
٦	النَّاسِ	وَ	مِنَ الْجِنَّةِ	٥

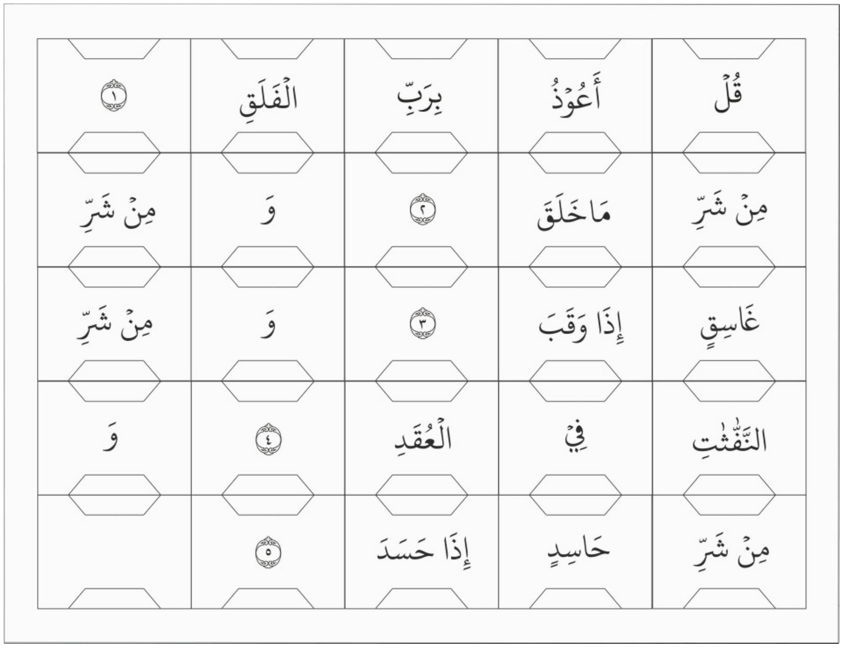
Gambar 3.19 Puzzle surat an-Naas

Gambar 3.20 adalah tampilan susunan nilai biner untuk *puzzle* surat al-Falaq.



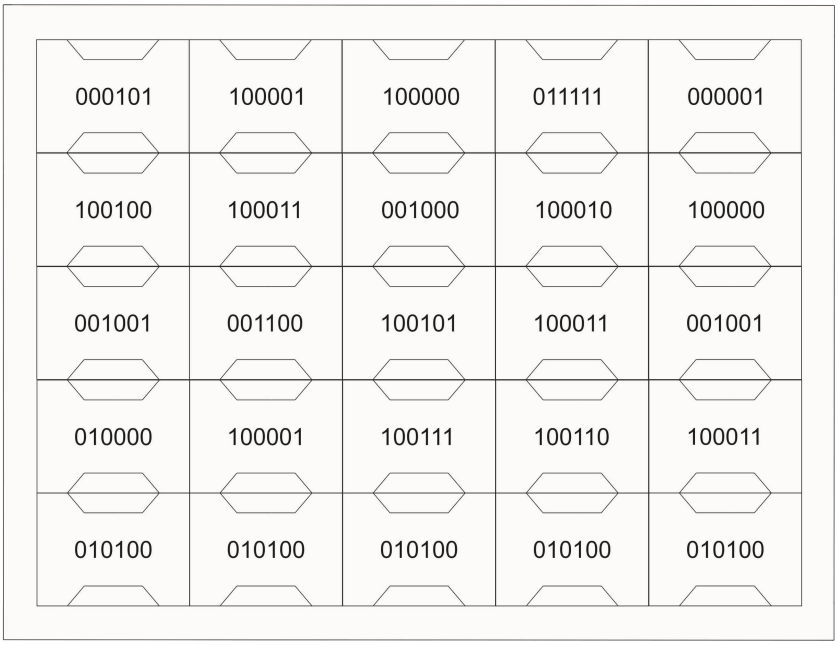
Gambar 3.20 Susunan nilai biner surat al-Falaq

Gambar 3.21 adalah gambar potongan *puzzle* secara lengkap untuk surat al-Falaq.



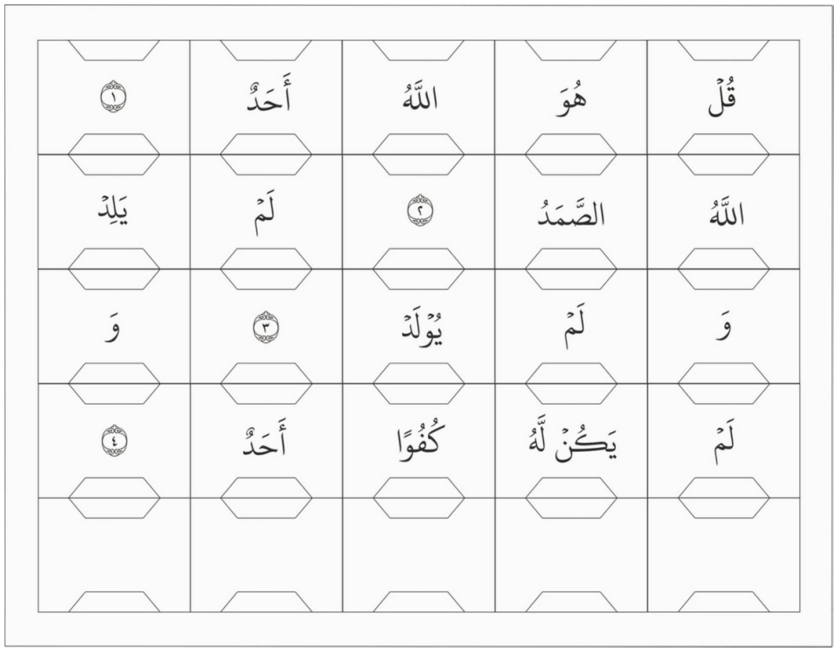
Gambar 3.21 Puzzle surat al-Falaq

Gambar 3.22 adalah tampilan susunan nilai biner untuk *puzzle* surat al-Ikhlas.



Gambar 3.22 Susunan nilai biner surat al-Ikhlas

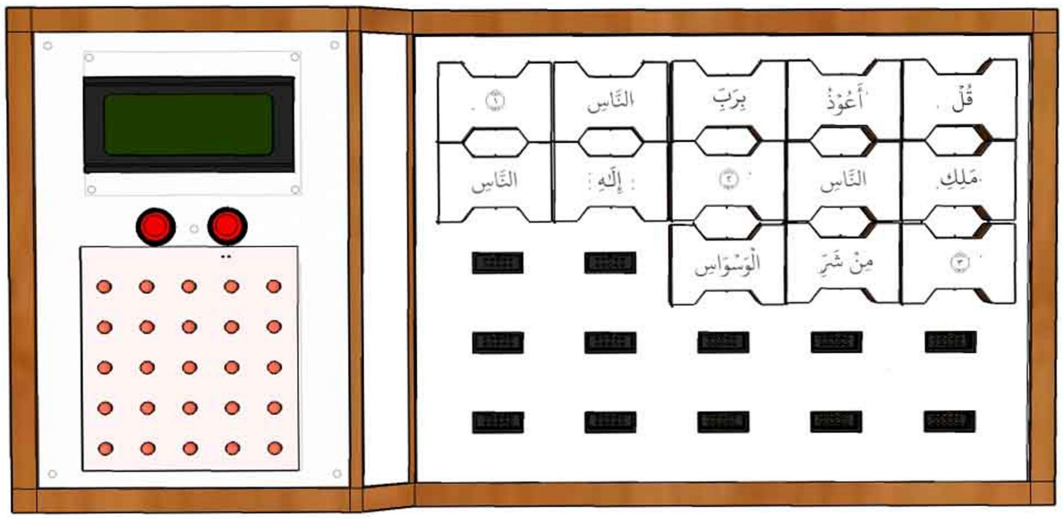
Gambar 3.23 adalah gambar potongan *puzzle* secara lengkap untuk surat al-Ikhlas.



Gambar 3.23 Puzzle surat al-Ikhlas

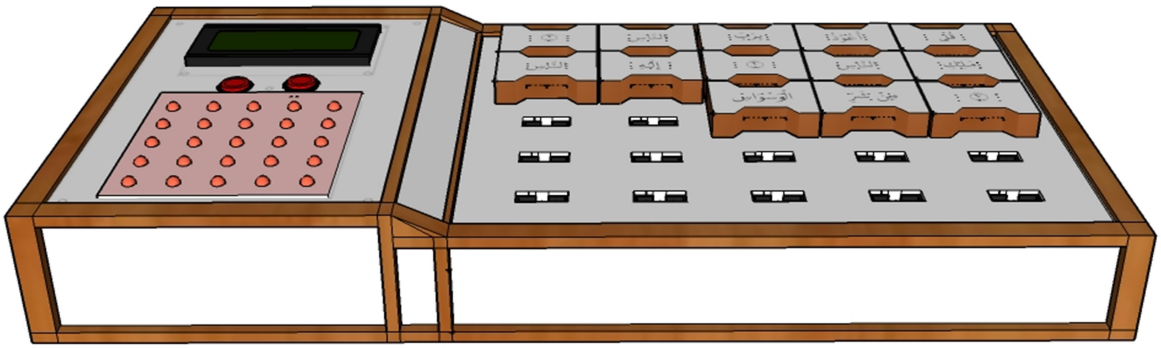
3.3.2 Perancangan Kemasan

Dalam proses pendesainan kemasan alat dilakukan beberapa pertimbangan sebagai landasan untuk menentukan bahan, ukuran dan bentuk kemasan. Ukuran dan bentuk kemasan didesain agar dapat menutupi perangkat keras yang dipakai serta memiliki nilai estetika agar dapat menarik minat pengguna untuk menggunakan alat ini. Bahan dasar pada pembuatan alat ini adalah kayu pinus. Alasan pemilihan kayu pinus sebagai bahan dasar dikarenakan kayu pinus memiliki tekstur yang mudah dibentuk, kokoh dan memiliki nilai estetika. Gambar 3.24 adalah gambar kemasan *puzzle* dilihat dari sisi atas.



Gambar 3.24 Kemasan puzzle dilihat dari atas

Gambar 3.25 adalah gambar kemasan *puzzle* dilihat dari sisi depan



Gambar 3.25 Kemasan puzzle dilihat dari depan