

**SURAT KETERANGAN**  
**MELAKUKAN KEGIATAN PENGABDIAN KEPADA MASYARAKAT**  
**INSTITUT TEKNOLOGI NASIONAL**  
**No. 277/C.02.01/LP2M/VI/2020**

Yang bertanda tangan di bawah ini,

Nama : Iwan Juwana, S.T., M.EM., Ph.D.  
Jabatan : Kepala  
Unit Kerja : LP2M-Itenas  
JL. P.K.H. Mustafa No.23 Bandung

Menerangkan bahwa,

No	Nama	NPP	Jabatan
1	Hendi Handian Rachmat, S.T., M.T., Ph.D.	20010401	Ketua Tim
2	M. Irfan Fariz, S.T.	-	Anggota Tim

Telah melakukan kegiatan Pengabdian kepada Masyarakat sebagai berikut:

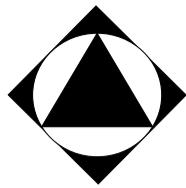
Nama Kegiatan : Pengembangan Sistem Jeda Iqomah Terkontrol Masjid Jamie Arrohmah  
Tempat : Masjid Jamie Arrohmah, Desa Cihanjuang, Kecamatan Parongpong, Kabupaten Bandung Barat  
Waktu : Januari - Juni 2020  
Sumber Dana : Pribadi

Demikian surat keterangan ini dibuat untuk dapat dipergunakan sebagaimana mestinya.

Bandung, 15 Juni 2020

Lembaga Penelitian dan Pengabdian  
kepada Masyarakat (LP2M) Itenas  
Kepala,

Iwan Juwana, S.T., M.EM., Ph.D.  
NPP. 20010601



**YAYASAN PENDIDIKAN DAYANG SUMBI**  
**INSTITUT TEKNOLOGI NASIONAL**  
**UNIT PELAKSANA TEKNIS PERPUSTAKAAN**

Jl. PKH. Hasan Mustapa No.23 Bandung 40124 Indonesia, Telepon: +62-22-7272215, Fax: +62-22-7202892  
Website: <http://lib.itenas.ac.id>, e-mail: library@itenas.ac.id

**Tanda Terima**

UPT Perpustakaan Itenas telah menerima Laporan Akhir Kegiatan Pengabdian Kepada Masyarakat:

<b>Tim Pelaksana</b>	<b>Judul Penelitian</b>
<b>1. Hendi Handian Rachmat</b>	<b>Pengembangan Sistem Jeda Iqomah Terkontrol di</b>
<b>2. M. Irfan Fariz</b>	<b>Masjid Jamie Arrohmah Desa Cihanjuang Kecamatan Parongpong Kabupaten Bandung Barat</b>

Kami ucapkan terima kasih, semoga laporan yang kami terima menjadi khazanah pengetahuan yang dapat dimanfaatkan oleh seluruh civitas akademika.

Demikian surat keterangan ini dibuat untuk digunakan sebagaimana mestinya.

Bandung, 17 Juni 2020

UPT Perpustakaan Itenas



Agus Wardana, S.Sos

**LAPORAN AKHIR KEGIATAN  
PENGABDIAN KEPADA MASYARAKAT**

**PENGEMBANGAN SISTEM JEDA IQAMAH TERKONTROL  
DI MASJID JAMIE ARROHMAH  
DESA CIHANJUANG KECAMATAN PARONGPONG  
KABUPATEN BANDUNG BARAT**

**Tim Pelaksana:  
Hendi Handian Rachmat  
M. Irfan Fariz**



**Laboratorium Elektronika – Jurusan Teknik Elektro  
Institut Teknologi Nasional  
Bandung  
2020**

**HALAMAN PENGESAHAN  
PENGABDIAN KEPADA MASYARAKAT (PKM)**

---

<b>Judul PKM</b>	:	PENGEMBANGAN SISTEM JEDA IQAMAH TERKONTROL
<b>Lokasi PKM</b>	:	MASJID JAMIE ARROHMAH DESA CIHANJUANG KECAMATAN PARONGPONG KABUPATEN BANDUNG BARAT
<b>Ketua PKM</b>	:	
a. Nama Lengkap	:	Hendi Handian Rachmat, S.T., M.T., PhD.
b. NIDN	:	0417087701
c. Jabatan Fungsional	:	Lektor Kepala
d. Program Studi	:	Teknik Elektro
e. Nomor HP	:	+62 82 11545 7393
f. Alamat surel (e-mail)	:	hendi.hr@gmail.com
g. Perguruan Tinggi	:	Institut Teknologi Nasional (ITENAS) Bandung
<b>Anggota (1)</b>	:	
a. Nama Lengkap	:	M. Irfan Fariz, S.T.
b. NIDN	:	-
c. Perguruan Tinggi	:	Teknik Elektro Institut Teknologi Nasional (ITENAS) Bandung
Lama PKM Keseluruhan	:	6 (enam) Bulan
Biaya PKM Keseluruhan	:	± Rp. 1.000.000,- (Satu Juta Rupiah)
Biaya Luaran Tambahan	:	Rp. 0,-

Mengetahui,  
Dekan FTI ITENAS



(Jono Suhartono, ST., MT., PhD. )  
NIP/NIK: 0406017801

Bandung, 10 Juni 2020  
Ketua PKM,



( Hendi H Rachmat, ST., MT., PhD. )  
NIDN: 0417087701

Menyetujui,  
Ketua lembaga penelitian



(Iwan Juwana, ST., M.EM, PhD.)  
NIDN: 0403017701

## ABSTRAK

Dalam kegiatan pengabdian kepada masyarakat saat ini dilakukan pengembangan sistem jeda iqamah yang dapat di-*setting* serta memiliki keluaran suara selain bunyi alarm atau *buzzer*. Pelaksanaan kegiatan ini berdasarkan permohonan bantuan dari pihak **Dewan Kemakmuran Masjid (DKM) Masjid Jamie ARROHMAH di Desa Cihanjuang Kecamatan Parongpong Kabupaten Bandung Barat**, di mana menghendaki adanya suatu alat untuk menjeda antara adzan dengan iqamah yang berfungsi sebagai alat bantu tertibnya ibadah di masjid tersebut. Pengaturan waktu jeda iqamah ingin dapat diatur dengan mudah dari posisi yang mudah dijangkau tanpa harus melepas jam jeda iqamah tersebut terlebih dahulu dari dinding yang berada di posisi atas dinding.. Hal ini dilakukan dikarenakan sistem jeda iqamah yang tersedia di pasaran tidak ada spesifikasi yang dikehendaki.

Sistem jeda iqamah ini terdiri dari tombol pengatur waktu jeda iqamah, kontoler berbasis Arduino Nano, *display* LED matriks, modul suara berbasis MP3 *Player* dan modul eksternal catu daya.. Sistem jeda iqamah ini dapat diatur waktu jedanya dalam satuan menit yaitu dari satu menit sampai dengan 60 menit. Pengaturan waktu dan pemberhentian pencacahan waktu jeda dapat di-*setting* melalui tombol *push button* yang diletakkan di dekat mimbar masjid guna mempermudah pengaturan pencacahan jeda waktu iqamah. Proses pengaturan dan pencacahan waktu diatur oleh mikrokontroler berbasis Arduino Nano. Proses pencacahan ditampilkan pada modul LED matrik DMD P10 yang menampilkan dua digit waktu menit dan dua digit waktu detik. Setelah pencacahan waktu jeda iqamah (pencacahan mundur) selesai dilakukan, maka mikrokontroler mengaktifkan modul MP3 *Player* untuk menjalankan file suara “Aqimiş-şalāta” pada loud speaker pada sistem jeda iqamah ini. Sistem ini menggunakan catu daya eksternal 8 VDC yang dhubungkan pada pin VIN modul Arduino Nano.

Dari hasil kegiatan ini, sistem jeda iqamah telah diserahkan dan digunakan di Masjid Jamie ARROHMAH. Proses evaluasi kerja sistem ini dilakukan secara parallel ketika sistem ini digunakan oleh pihak DKM Masjid.

## DAFTAR ISI

ABSTRAK .....	i
DAFTAR ISI .....	ii
DAFTAR GAMBAR .....	iv
DAFTAR TABEL .....	v
BAB I. PENDAHULUAN .....	1
1.1. Latar Belakang Masalah .....	1
1.2. Tujuan dan Manfaat .....	1
1.3. Luaran Yang Diharapkan .....	2
BAB II. MODUL-MODUL SISTEM JEDA WAKTU IQAMAH .....	3
2.1. Modul Arduino Nano .....	3
2.2. DMD ( <i>Dot Matrix Display</i> ) .....	6
2.3. DF <i>Player</i> Mini .....	12
2.4. <i>Audio Amplifier</i> Eksternal .....	16
BAB III. PERANCANGAN DAN IMPLEMENTASI SISTEM .....	18
3.1. Diagram Blok dan Cara Kerja Sistem .....	18
3.2. Perancangan dan Implementasi Sistem .....	19
3.2.1. Modul <i>Keypad</i> .....	20
3.2.2. Modul Kontroler .....	21
3.2.3. Modul <i>Display</i> .....	22
3.2.4. Modul MP3 <i>Player</i> .....	23
3.2.5. Modul <i>Audio Amplifier</i> .....	25
3.2.6. Modul Catu Daya .....	25
3.2.7. Perangkat Lunak Sistem .....	27
BAB IV. HASIL KEGIATAN .....	40
BAB V. KESIMPULAN DAN SARAN .....	44
5.1. Kesimpulan Hasil Kegiatan .....	44
5.2. Saran Hasil Kegiatan .....	44
DAFTAR PUSTAKA .....	45

## DAFTAR GAMBAR

Gambar 2.1.	Bentuk fisik modul Arduino Nano	3
Gambar 2.2.	<i>Pinout</i> Arduino Nano	5
Gambar 2.3.	Bentuk fisik bagian depan dan bagian belakang DMD P105	7
Gambar 2.4.	Bentuk fisik konektor DMD P10	7
Gambar 2.5.	Koneksi pin-pin Arduino dengan pin-pin DMD P10	8
Gambar 2.6.	Prosedur koneksi antara modul Arduino, konektor DMD dan modul DMD	9
Gambar 2.7.	Skematik diagram modul DMD P10	10
Gambar 2.8.	<i>Listing</i> program pengujian modul DMD P10	11
Gambar 2.9.	Contoh tampilan hasil pengujian modul DMD P10	11
Gambar 2.10.	Letak dan deskripsi setiap <i>pinout</i> modul DF Player Mini	13
Gambar 2.11.	Skematik diagram hubungan antara modul DF <i>Player</i> Mini dengan modul kontroler	14
Gambar 2.12	Contoh <i>listing</i> program untuk menjalankan modul DF <i>Player</i> Mini.	15
Gambar 2.13.	<i>Pinout</i> IC PAM8403, modul PAM8403 dan deskripsi pin-pin IC	16
Gambar 2.14.	Rangkaian skematik modul <i>audio amplifier</i> berbasis IC PAM8403	17
Gambar 3.1.	Diagram blok sistem sistem jeda iqamah	18
Gambar 3.2.	Skematik diagram modul <i>keypad</i>	20
Gambar 3.3.	Skematik diagram modul kontroler	22
Gambar 3.4.	Skematik diagram soket modul <i>display</i> DMD P10	23
Gambar 3.5.	Skematik diagram modul MP3 <i>Player</i> dengan modul Arduino Nano	24
Gambar 3.6.	Skematik diagram modul <i>audio amplifier</i>	25
Gambar 3.7.	Diagram skematik sistem jeda iqamah	26
Gambar 3.8.	Diagram alir program utama sistem waktu jeda iqamah	28
Gambar 3.9.	Potongan <i>listing</i> program inisialisasi program	29

Gambar 3.10. Potongan <i>listing</i> awal program prosedur <b>setup()</b>	30
Gambar 3.11. Potongan <i>listing</i> program pengecekan tombol <b>mode</b> dan tombol <b>start</b>	31
Gambar 3.12. Potongan <i>listing</i> program pengurangan waktu setiap detik	32
Gambar 3.13. Potongan <i>listing</i> program pengecekan tombol <b>stop</b>	33
Gambar 3.14. Diagram alir sub program <b>tampil()</b>	34
Gambar 3.15. Potongan <i>listing</i> program sub program <b>tampil()</b>	35
Gambar 3.16. Diagram alir sub program <b>current()</b>	36
Gambar 3.17. Potongan <i>listing</i> program sub program <b>current()</b>	37
Gambar 3.18. Diagram alir sub program <b>waktu()</b>	38
Gambar 3.19. Potongan <i>listing</i> program sub program <b>waktu()</b>	39
Gambar 4.1. Hasil implementasi sistem jeda waktu iqamah	40
Gambar 4.2. Tampak dalam <i>casing display</i>	41
Gambar 4.3. Tampak depan <i>casing</i> tombol	41
Gambar 4.4. Sistem jeda waktu iqamah dengan speaker eksternal	42
Gambar 4.5. Dokumentasi serah terima sistem jeda waktu iqamah kepada pihak DKM Masjid Jamie ARROHMAH	43

## **DAFTAR TABEL**

Tabel 2.1. Tabel 2.1. Spesifikasi Arduino Nano

4

## **BAB I**

### **PENDAHULUAN**

#### **1.1. Latar Belakang Masalah**

Berdasarkan pada surat permohonan (terlampir) tertanggal 19 Januari 2020 yang diajukan kepada Laboratorium Elektronika ITENAS bahwa pihak **Dewan Kemakmuran Masjid (DKM) Masjid Jamie ARROHMAH di Desa Cihanjuang Kecamatan Parongpong Kabupaten Bandung Barat** memohon bantuan penyediaan alat untuk menjeda antara adzan dengan iqamah. Alat ini akan difungsikan sebagai alat bantu untuk mentertibkan pelaksanaan ibadah shalat berjamaah di Masjid Jamie ARROHMAH.

Pada saat ini, alat jeda iqamah ini belum tersedia di masjid tersebut, sehingga untuk memulai shalat berjamaah seringkali tidak tertib karena harus menunggu jamaah untuk melaksanakan shalat sunnah terlebih dahulu. Dengan adanya alat jeda iqamah ini, pihak DKM masjid mengharapkan agar waktu memulainya pelaksanaan shalat berjamaah lebih tertib, di mana para jamaah yang hadir dalam masjid dapat memperkirakan waktu yang tersedia berdasarkan alat jeda iqamah tersebut untuk melakukan shalat sunnah.

Pihak DKM memohon juga agar *setting* waktu jeda iqamah dapat di-*setting* dengan mudah. Hal ini disebabkan karena dikehendaki bahwa lamanya jeda iqamah ini data disesuaikan waktunya berdasarkan jadwal shalat waktu lima waktu yang akan dilaksanakan. Sebagai contoh, jeda waktu iqamah shalat shubuh akan di-*setting* lebih lama dari waktu jeda untuk shalat-shalat lainnya. Selain itu, pihak DKM menginginkan bahwa *output* penanda selesainya waktu jeda ditandai dengan suara “aqimis salata”, bukan berupa suara *buzzer*.

#### **1.2. Tujuan dan Manfaat**

Perancangan dan implementasi sistem jeda iqamah ini bertujuan untuk meningkatkan ketertiban pelaksanaan ibadah shalat berjamaah di Masjid Jamie ARROHMAH.

### **1.3. Luaran Yang Diharapkan**

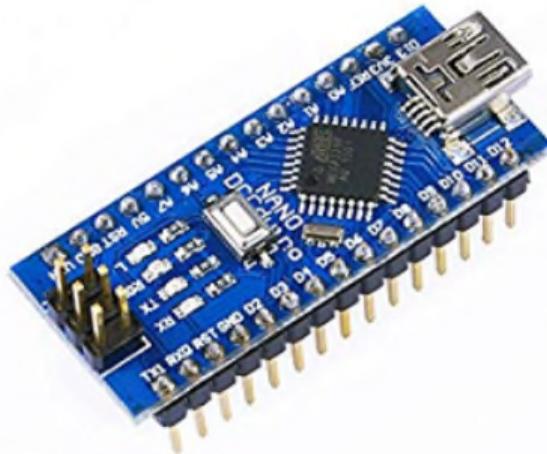
Luaran utama yang diharapkan dari kegiatan ini adalah tersedianya alat jeda iqamah di Masjid Jamie ARROHMAH yang dapat di-*setting* waktu jedanya dengan mudah dan memiliki *output* suara “aqimis salata”, selain display berupa angka pencacahan waktu mundur.

## **BAB II**

### **MODUL-MODUL SISTEM JEDA WAKTU IQAMAH**

#### **2.1. Modul Arduino Nano ([www.theengineeringprojects.com](http://www.theengineeringprojects.com))**

Arduino Nano adalah sebuah *board* mikrokontroler yang memiliki spesifikasi dimensi yang kecil, kompatibel, fleksibel, *breadboard-friendly*. Modul Arduino Nano menggunakan ATmega328p (Arduino Nano V3.x) atau ATmega 168 (Arduino Nano V3.x) sebagai kontroler utamanya. Arduino Nano memiliki fungsionalitas dan jumlah pin yang sama dengan Arduino UNO tetapi dimensinya kecil. Mikrokontroler ini beroperasi pada tegangan 5V yang disuplai melalui port USB. Namun, tegangan *inputnya* dapat juga menggunakan tegangan 7V sampai 12V melalui pin VIN. bentuk fisik Arduino Nano ditunjukkan pada Gambar 2.1.



Gambar 2.1. Bentuk fisik modul Arduino Nano ([www.theengineeringprojects.com](http://www.theengineeringprojects.com))

Arduino Nano memiliki sejumlah *pinout* yaitu 14 pin digital, 8 pin analog, 2 pin *reset* dan 6 pin *power*. Masing-masing pin digital dan pin analog sebetulnya memiliki beberapa fungsi, namun fungsi utamanya dapat dikonfigurasikan sebagai pin *input* atau pin *output*. Pin-pin berfungsi sebagai *input* ketika dihubungkan dengan sensor, sedangkan berfungsi sebagai *output* ketika dihubungkan dengan

beban. Adapun pin analog memiliki karakteristik resolusi 10-bit yang mengukur variasi nilai tegangan dari 0 sampai dengan 5V.

Arduino Nano memiliki osilator kristal dengan frekuensi 16 MHz yang digunakan untuk menghasilkan *clock*. Kapasitas *flash memory* pada Arduino Nano adalah sebesar 16 kB atau 32 kB. Hal ini tergantung dari *board* ATmega, di mana ATmega 168 berkapasitas 16 kB dan ATmega328 berkapasitas 32 kB. *Flash memory* digunakan untuk penyimpanan kode-kode program (ROM *flash*). Dari total *flash memory* yang dimiliki, kapasitas sebesar 2 kB digunakan untuk *bootloader*.

Arduino Nano memiliki kapasitas memori yang bervariasi tergantung dari mikrokontroler yang digunakan, yaitu SRAM dari 1 kB hingga 2 kB dan EEPROM dari 512 byte hingga 1 kB untuk masing-masing ATmega168 dan ATmega328. Pada Tabel 2.1. ditunjukkan spesifikasi dari Arduino Nano.

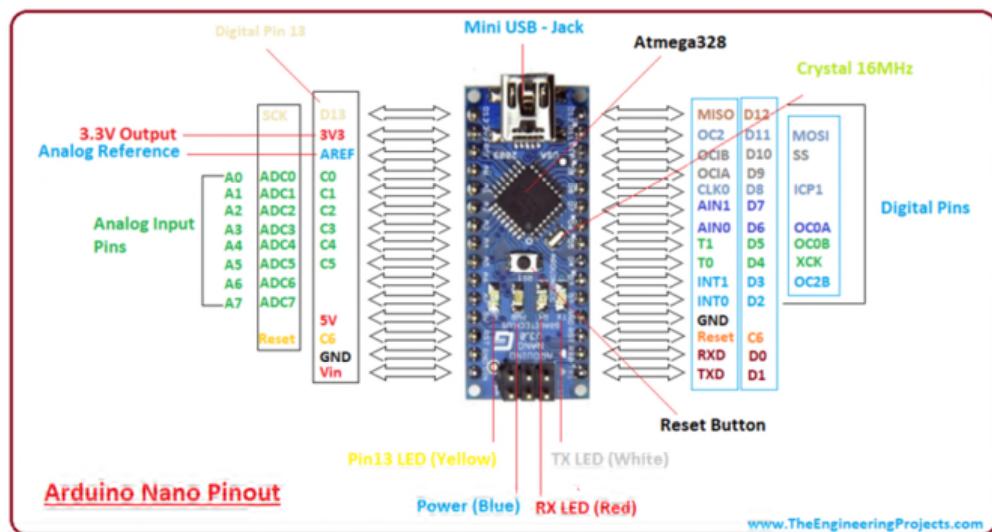
Tabel 2.1. Spesifikasi Arduino Nano ([www.theengineeringprojects.com](http://www.theengineeringprojects.com))

<b>Microcontroller</b>	<b>ATmega 168P/ATmega328P</b>
<b>Operating Voltage</b>	5V
<b>Input Voltage</b>	7 – 12 V
<b>Digital I/O Pins</b>	14
<b>PWM</b>	6 out of 14 digital pins
<b>Max. Current Rating</b>	40mA
<b>USB</b>	Mini
<b>Analog Pins</b>	8
<b>Flash Memory</b>	16KB or 32KB
<b>SRAM</b>	1KB or 2KB
<b>Crystal Oscillator</b>	16 MHz
<b>EEPROM</b>	512bytes or 1KB
<b>USART</b>	Yes

Arduino Nano diprogram dengan menggunakan Arduino IDE (*Integrated Development Environment*) yang bekerja secara *online* dan *offline*. Pengaturan awal tidak diperlukan untuk menjalankan *board* ini. Untuk menjalankan Arduino

Nano dibutuhkan beberapa *hardware* dan *software* yaitu *board* Arduino Nano, kabel USB mini (untuk menyalurkan program dari komputer ke Arduino Nano), dan *software* Arduino IDE yang sudah terinstalasi pada komputer..

*Pinout* Arduino Nano ditunjukkan pada Gambar 2.2. Setiap pin-pin Arduino Nano memiliki fungsi spesifik, sebagai contoh pin-pin analog A4 dan A5 digunakan sebagai konverter analog ke digital di mana kedua pin tersebut juga dapat digunakan untuk komunikasi data I2C. Sama halnya dengan pin digital yang berjumlah 14, enam buah pin diantaranya digunakan untuk menghasilkan PWM (*Pulse Width Modulation*).



Gambar 2.2. *Pinout* Arduino Nano ([www.theengineeringprojects.com](http://www.theengineeringprojects.com))

Deskripsi pin-pin Arduino Nano terdapat di bawah ini:

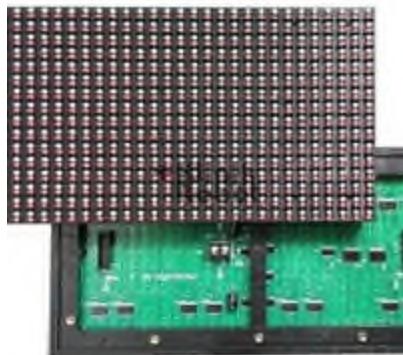
- VIN berfungsi sebagai pin *input* tegangan catu daya ke *board* ketika menggunakan sumber daya eksternal sebesar 7 V sampai 12 V.
- 5V merupakan pin *output* tegangan 5V yang digunakan untuk memberi daya pada komponen lain yang dihubungkan dengan modul Arduino.
- 3.3V adalah pin *output* tegangan 3,3V yang dihasilkan oleh pengatur tegangan pada *board*.
- GND adalah pin *ground*. Terdapat beberapa pin *ground* yang dapat dihubungkan ketika dibutuhkan lebih dari satu pin *ground*.

- e. *Reset* berfungsi untuk melakukan *reset* pada *board*. Nilai **LOW** ke pin *reset* akan mengatur ulang *controller*.
- f. Pin analog berjumlah 8 pin, yaitu dari pin A0 sampai A7. Pin-pin ini digunakan untuk mengukur tegangan analog dengan rentang tegangan analog dari 0 sampai 5 V.
- g. Rx dan Tx digunakan untuk komunikasi serial di mana Tx berfungsi untuk transmisi data dan Rx sebagai penerima data.
- h. AREF digunakan sebagai tegangan referensi untuk tegangan *input*.
- i. PWM (*Pulse Width Modulation*) terdiri dari 6 pin yaitu pin 3, pin 5, pin 6, pin 9, pin 10, dan pin 11. Fungsi keenam pin tersebut untuk menyediakan 8-bit *output* PWM. Metode ini digunakan untuk mendapat hasil analog dari sumber digital.
- j. SPI (*Serial Peripheral Interface*) terdiri dari 4 pin yaitu pin 10 (SS), pin 11 (MOSI), pin 12 (MISO), pin 13 (SCK). SPI digunakan untuk transfer data antara mikrokontroler dan periferal lainnya seperti sensor, register, dan SD *card*.
- k. *External Interrupts* terdiri dari pin 2 dan pin 3 yang digunakan ketika keadaan darurat di mana program utama harus dihentikan dan instruksi penting harus dipanggil saat itu juga. Program utama akan berjalan kembali setelah instruksi *interrupt* telah dipanggil dan dieksekusi.
- l. I2C (*Inter-Integrated Circuit*) menggunakan pin A4 dan pin A5 dimana A4 sebagai jalur data serial (SDA) yang membawa data dan A5 sebagai jalur *clock* serial (SCL) yang merupakan sinyal *clock*. I2C dihasilkan oleh divais *master* yang digunakan untuk sinkronisasi data antar divais pada bus I2C.

## 2.2. DMD (*Dot Matrix Display*) P10

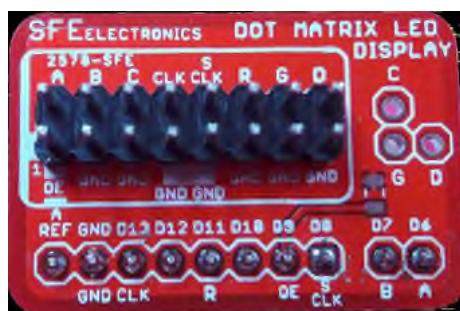
*Dot Matrix Display* (DMD) P10 merupakan sebuah *display* LED yang biasa digunakan untuk menampilkan suatu karakter dengan metoda running text. Divais ini biasanya digunakan untuk pedagang di toko-toko, waktu jadwal adzan di masjid, nomor antrian di perusahaan dan lain-lain. *Display* ini sangat praktis

dan fleksibel untuk digunakan, karena jumlah *display*-nya dapat diperbesar dengan cara menghubungkan secara *cascade* beberapa *display*. Salah satu tipe dari DMD ini adalah P10 dengan dimensi yaitu 16 LED x 32 LED, seperti ditunjukkan pada Gambar 2.3.



Gambar 2.3. Bentuk fisik bagian depan dan bagian belakang DMD P10  
(<https://www.tokopedia.com/klinikrobot/kr04110-p10-outdoor-waterproof-dual-color-led-module-320x160mm>)

Untuk mengontrol aktifasi dan jenis tampilan *display* ini pun relatif mudah dengan menggunakan kontroler Arduino. Hal ini dikarenakan sudah terdapat sejumlah *library* yang telah dibuat oleh para pembuat-pembuat program. Untuk menghubungkan antara kontroler Arduino dengan DMD P10, diperlukan konektor (soket) DMD yang berfungsi sebagai *interface* pin konektor antara pin-pin Arduino dengan pin-pin DMD P10. Bentuk fisik dari konektor DMD ditunjukkan pada Gambar 2.4.



Gambar 2.4. Bentuk fisik konektor DMD P10  
(<https://www.instructables.com/id/Display-Text-at-P10-LED-Display-Using-Arduino/>)

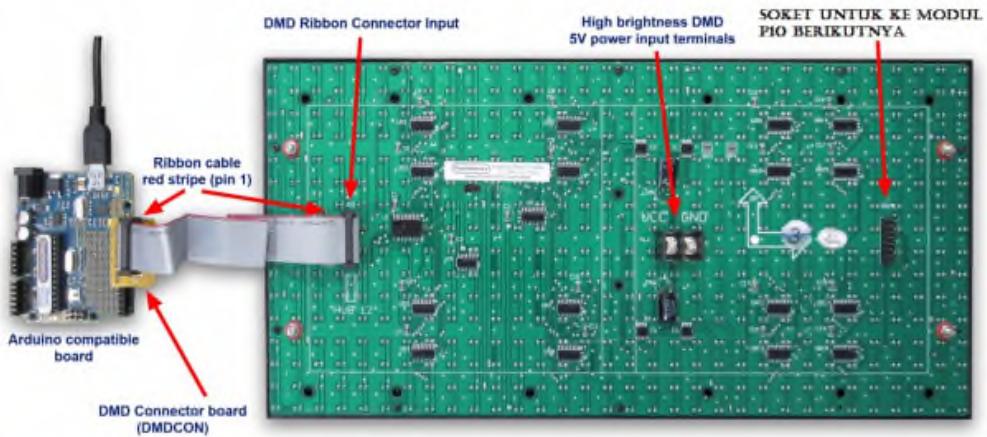
Jumlah pin digital yang diperlukan untuk dihubungkan dengan modul DMD P10 adalah sejumlah enam pin digital ditambah dengan pin *ground*. Hubungan antara pin-pin Arduino dan pin-pin DMD P10 ditunjukkan pada Gambar 2.5.

soket Kabel P10	Arduino
Pin 1	Pin Digital 9 ( <b>D9</b> )
Pin 2	Pin Digital 6 ( <b>D6</b> )
Pin 3	GND
Pin 4	Pin Digital 7 ( <b>D7</b> )
Pin 8	Pin Digital 13 ( <b>D13</b> )
Pin 10	Pin Digital 8 ( <b>D8</b> )
Pin 12	Pin Digital 11 ( <b>D11</b> )

Gambar 2.5. Koneksi pin-pin Arduino dengan pin-pin DMD P10  
(<https://pecontrol.wordpress.com/2016/04/30/pengetahuan-dasar-pemrograman-modul-leddot-matrik-display-dmd-p10-dengan-arduino/>)

Pin digital Arduino D9 dihubungkan dengan pin OE (*output enable*) untuk *on/off* semua LED. Pin digital Arduino D6 dan D7 dihubungkan secara berturut-turut dengan pin A dan pin B dari modul DMD P10. Kedua pin ini berfungsi untuk memilih kolom yang aktif. Selanjutnya pin digital Arduino D8 dihubungkan dengan pin SCLK yang merupakan suatu sinyal *serial clock* yang dikeluarkan ketika *master device* akan melakukan transmisi data (bit). Adapun pin CLK sebagai pin untuk membangkitkan sinyal *clock* yang dihubungkan dengan pin digital Arduino D13. Data yang dikirimkan oleh modul Arduino melalui pin digital D11 diterima oleh DMD P10 pada pin DATA. Pin GND (*ground*) dari modul Arduino harus dihubungkan juga dengan pin GND dari modul DMD P10.

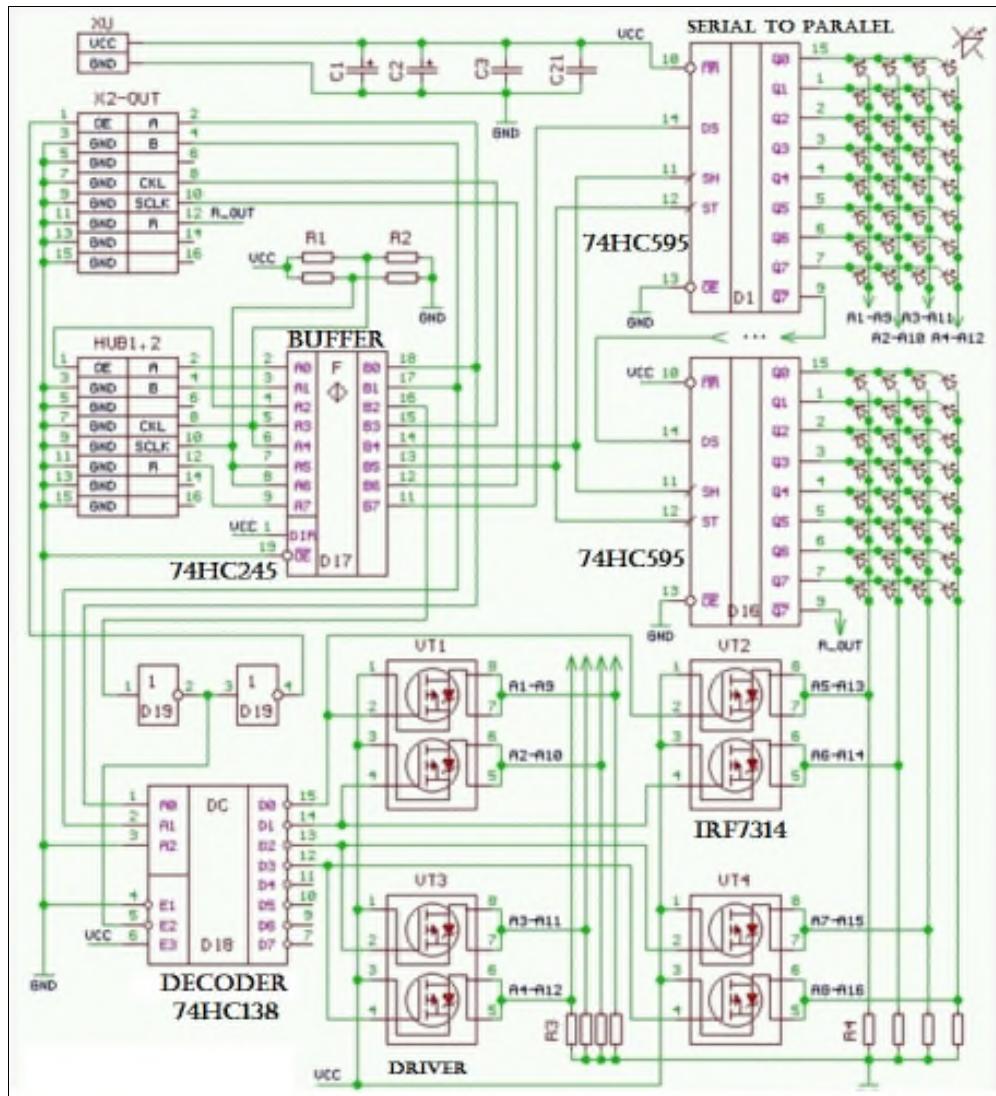
Pin-pin konektor DMD ini dirancang memang bersesuaian dengan soket pin-pin digital modul Arduino, sehingga konektor DMD ini dapat dikatakan sebagai modul *shield*. Adapun untuk menghubungkan konektor DMD dengan modul DMD P10 digunakan kabel **Ribbon** 16 jalur. Koneksi antar modul ini ditunjukkan pada Gambar 2.6.



Gambar 2.6. Prosedur koneksi antara modul Arduino, konektor DMD dan modul DMD (<https://pecontrol.wordpress.com/2016/04/30/pengetahuan-dasar-pemrograman-modul-leddot-matrik-display-dmd-p10-dengan-arduino/>)

Pada Gambar 2.7. ditunjukkan skematik diagram sebuah modul DMD P10. Dari gambar skematik tersebut dapat dianalisis bahwa data yang dikirimkan dari Arduino secara serial melalui konektor **HUB1.2**. Selanjutnya data tersebut diterima oleh pin data IC *serial to parallel 74HC595* setelah melalui sebuah IC *Buffer 74HC245*. Prosedur *cascade* dengan DMD P10 yang berikutnya dilakukan melalui soket P10 (**X2-OUT**) guna memperbesar dimensi *display* sesuai dengan kebutuhan.

Untuk keperluan suplai arus modul ini menggunakan empat buah IC *driver* penguat daya VT1, VT2, VT3 dan VT4 berupa IC dual MOSFET **IRF7314**. Suplai arus ini diberikan kepada setiap baris jalur LED. Pengaturan kerja MOSFET ini dilakukan oleh empat buah jalur output IC *Decoder 74HC138* yang diatur oleh dua buah pin yaitu pin A (LSB) dan pin B. Adapun pengaturan penyalaan 32 kolom LED modul DMD P10 dilakukan oleh dua buah IC *serial to parallel 74HC595*. Jika modul ini dianggap kurang terang penyalaannya, maka dapat memberikan suplai tegangan eksternal 5 Volt melalui pin **XV** (*5V power input terminal*).



Gambar 2.7. Skematik diagram modul DMD P10  
[\(https://pccontrol.wordpress.com/2016/04/30/pengetahuan-dasar-pemrograman-modul-leddot-matrik-display-dmd-p10-dengan-arduino/\)](https://pccontrol.wordpress.com/2016/04/30/pengetahuan-dasar-pemrograman-modul-leddot-matrik-display-dmd-p10-dengan-arduino/)

Setelah dilakukan prosedur instalasi antar modul, prosedur selanjutnya adalah proses pemrograman pada Arduino. Ada beberapa *library* yang diperlukan untuk menjalankan modul DMD P10 dengan menggunakan modul Arduino yaitu *library <SPI.h>* untuk komunikasi serial, *library <DMD2.h>* untuk aktifasi modul DMD P10 dan *library <fonts/....h>* untuk jenis huruf yang akan digunakan. Salah satu contoh program yang dapat digunakan untuk menguji fungsi kerja display DMD ditunjukkan pada Gambar 2.8.

```

// Insert file Library
#include <SPI.h>
#include <DMD2.h>
#include <fonts/Arial_Black_16.h>

SoftDMD dmd(1,1);           // Number of P10 panels used X, Y
DMD_TextBox box(dmd, 2, 1, 32, 16); // Set Box (dmd, x, y, Height, Width)

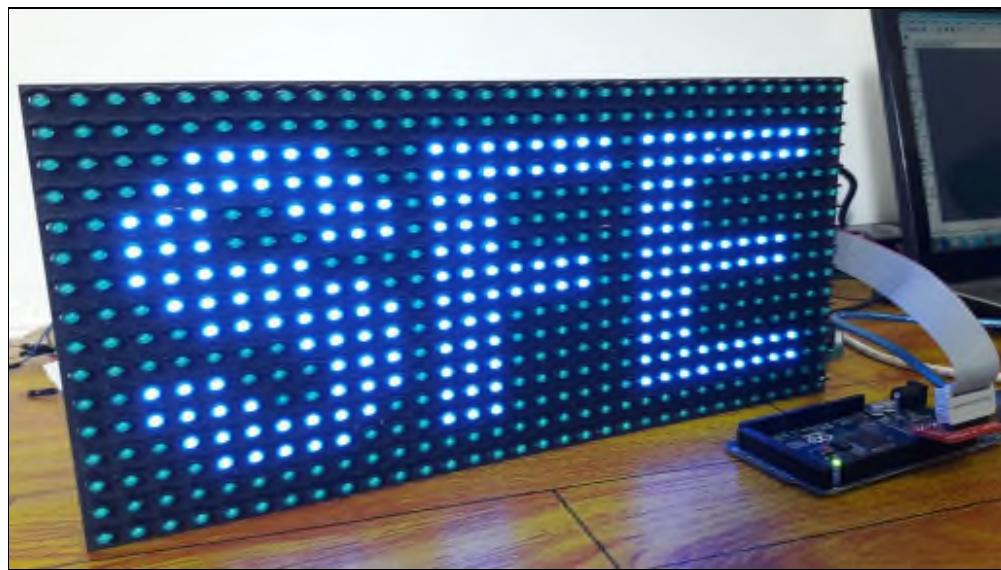
void setup()
{
    dmd.setBrightness(10);           // Set brightness 0 - 255
    dmd.selectFont(Arial_Black_16); // Font used
    dmd.begin();                   // Start DMD
    box.print("SFE");             // Display TEXT SFE
}

void loop() {
}

```

Gambar 2.8. Listing program pengujian modul DMD P10  
[\(https://www.instructables.com/id/Display-Text-at-P10-LED-Display-Using-Arduino/\)](https://www.instructables.com/id/Display-Text-at-P10-LED-Display-Using-Arduino/)

Jika hasil instalasi *hardware* dan *software* telah benar, maka seharusnya *display* DMD P10 akan menampilkan tulisan seperti yang ditunjukkan pada Gambar 2.9.



Gambar 2.9. Contoh tampilan hasil pengujian modul DMD P10  
[\(https://www.instructables.com/id/Display-Text-at-P10-LED-Display-Using-Arduino/\)](https://www.instructables.com/id/Display-Text-at-P10-LED-Display-Using-Arduino/)

### 2.3. DF Player Mini (<http://www.picaxe.com/docs/spe033.pdf>)

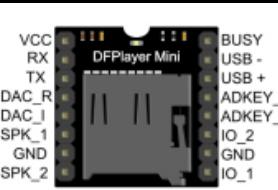
DF Player Mini merupakan salah satu jenis modul MP3 Player yang relatif murah dan berdimensi kecil (2 cm x 2 cm). Modul ini dapat diaktifkan secara portabel dengan hanya menghubungkan ke baterai, *loud speaker* dan *push button*. Namun, jika ingin dikontrol dengan sebuah perangkat berbasis sistem mikroprosesor seperti modul Arduino, sistem ini dihubungkan melalui komunikasi serial UART (pin Tx dan pin Rx).

Sejumlah aplikasi yang membutuhkan *output* berupa pemutaran suara atau lagu dapat diimplementasikan dengan modul ini. Spesifikasi modul DF Player Mini ini adalah sebagai berikut:

- Mendukung tipe file MP3 dan WMV
- Mendukung *sampling rate* (kHz) : 8/11.025/12/16/22.05/24/32/44.1/48
- *Output* DAC 24 bit, mendukung untuk jangkauan dinamik 90 dB dan SNR 85 dB
- Mendukung sistem file FAT16 dan FAT 32 dengan kapasitas penyimpanan maksimum 32 GB untuk TF *card*, 32 GB untuk U *disk*, 64 MB untuk NORFLASH
- Mendukung sejumlah mode kontrol: mode kontrol I/O, mode serial, mode kontrol AD button.
- Memiliki fungsi tungu suara iklan, di mana music dapat ditangguhkan. Ketika iklan telah selesai, musik dapat dimainkan kembali.
- Dikemas dengan audio amplifier internal dengan daya 3 W
- Data audio disusun dalam bentuk *folder*, di mana sistem dapat mendukung sampai dengan 100 *folder*. Setiap *folder* dapat menyimpan hingga maksimal 255 lagu.
- Volume dapat diatur hingga level 30, serta Equalizer hingga level 6.
- Terdapat komunikasi port UART untuk komunikasi serial standar dengan *baud rate* yang dapat diatur (standar baud rate = 9600)
- Tegangan operasi modul ini adalah dari 3,2 VDC sampai dengan 5,0 VDC
- Arus *standby* sebesar 20 mA

- Suhu kerja modul berkisar dari  $-40^{\circ}\text{C}$  sampai dengan  $+70^{\circ}\text{C}$ .
- Kelembaban kerja modul berada di 5% sampai dengan 95%

Adapun letak dan deskripsi setiap *pinout* dari modul ini ditunjukkan pada Gambar 2.10.



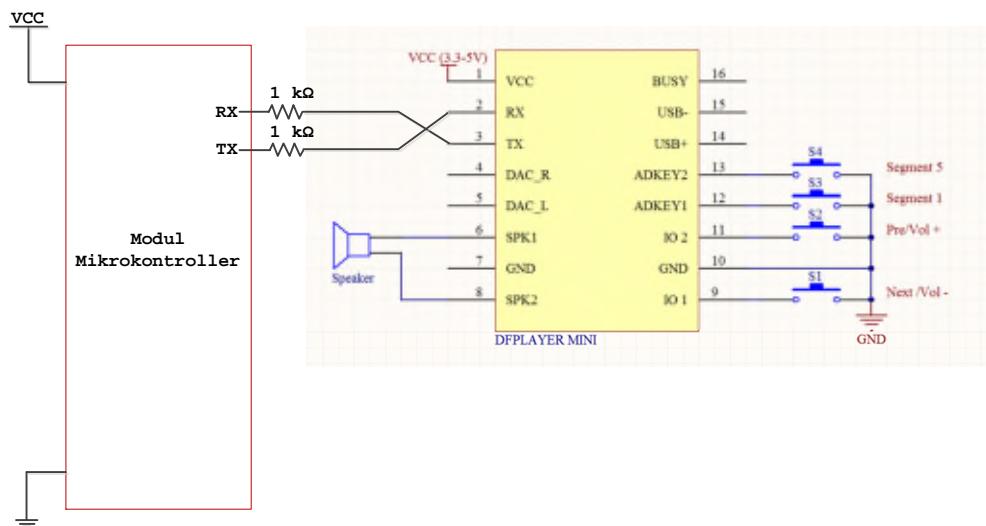
Pin	Description	Note
VCC	Input Voltage	DC3.2~5.0V; Type: DC4.2V
RX	UART serial input	
TX	UART serial output	
DAC_R	Audio output right channel	Drive earphone and amplifier
DAC_L	Audio output left channel	Drive earphone and amplifier
SPK2	Speaker-	Drive speaker less than 3W
GND	Ground	Power GND
SPK1	Speaker+	Drive speaker less than 3W
IO1	Trigger port 1	Short press to play previous (long press to decrease volume)
GND	Ground	Power GND
IO2	Trigger port 2	Short press to play next (long press to increase volume)
ADKEY1	AD Port 1	Trigger play first segment
ADKEY2	AD Port 2	Trigger play fifth segment
USB+	USB+ DP	USB Port
USB-	USB- DM	USB Port
BUSY	Playing Status	Low means playing \ High means no

Gambar 2.10. Letak dan deskripsi setiap *pinout* modul DF Player Mini  
[\(http://www.picaxe.com/docs/spe033.pdf\)](http://www.picaxe.com/docs/spe033.pdf)

Seperti telah dijelaskan pada bagian spesifikasi sistem, terdapat beberapa mode operasi untuk mengaktifkan modul ini yaitu: mode kontrol I/O, mode serial, mode kontrol AD button. Adapun untuk sistem jeda waktu iqamah memilih mode kontrol I/O. Hal dikarenakan pengaturan aktifasi pemutaran file MP3 pada modul DF Player Mini dilakukan oleh modul kontoler Arduino.

Agar dapat modul DF Player Mini ini diatur dengan menggunakan modul Arduino maka terdapat delapan pin yang digunakan dalam modul DF Player Mini yaitu pin VCC, pin RX, pin TX, pin DAC\_R, pin DAC\_L, pin SPK\_1, pin SPK\_2 dan pin GND. Pin VCC dan pin GND dihubungkan dengan catudaya DC 3,3 volt sampai dengan 5 volt. Pin TX modul DF Player Mini dihubungkan dengan pin yang dijadikan RX pada modul Arduino dan pin RX modul DF Player Mini dihubungkan dengan pin yang dijadikan TX modul Arduino. Untuk *output* suara dari modul ini berasal dari pin SPK\_1 dan pin SPK yang dihubungkan dengan

*loud speaker*. Adapun jika suara kurang besar, maka pin DAC\_R dan pin DAC\_L dihubungkan ke audio input *amplifier* eksternal. Pada Gambar 2.11 ditunjukkan contoh skematik diagram hubungan antara modul DF *Player* Mini dengan modul kontroler.



Gambar 2.11. Skematik diagram hubungan antara modul DF *Player* Mini dengan modul kontroler (<http://www.picaxe.com/docs/spe033.pdf>).

Jika ingin ditambahkan pengaturan volume suara dan perpindahan pemutaran file MP3 secara manual, maka dapat ditambahkan beberapa *push button* pada pin IO1 dan pin IO2 di mana pin-pin ini dapat difungsikan ganda yaitu sebagai pengaturan volume dan pemutaran file MP3. Fungsi ganda pin-pin ini ditentukan oleh durasi penekanan tombol yaitu ketika penekanan tombolnya sebentar (cepat) maka pin akan berfungsi untuk mengatur pemutaran file MP3, sedangkan jika penekanan tombol lama maka pin akan berfungsi untuk mengatur volume suara (lihat deskripsi pada Gambar 2.10). Untuk pin ADKEY1 dan pin ADKEY2 adalah pin yang secara berturut-turut digunakan untuk mengatur penyalaan file MP3 ke segmen 1 dan ke segmen 5.

Apabila modul mikrokontroler yang digunakan adalah modul Arduino, maka pin RX dan pin TX dapat menggunakan dua metoda yaitu menggunakan pin RX (pin 0) dan pin TX (pin 1) atau menggunakan dua pin digital lain yang di-*setting* menjadi pin RX dan pin TX. Pemograman DF *Player* Mini dengan

menggunakan modul Arduino memerlukan dua buah *library* yaitu <**SoftwareSerial.h**> untuk mengatur pin-pin digital sebagai pin RX dan pin TX, serta <**DFPlayer\_Mini\_Mp3.h**> untuk pengaturan *setting* fungsi-fungsi modul DF *Player* Mini yang lain. Pada Gambar 2.12. ditunjukkan salah satu contoh *listing* program untuk mengaktifkan modul DF *Player* Mini dengan menggunakan modul Arduino (<http://www.belajarduino.com/2016/10/arduinop.html>).

```
#include <SoftwareSerial.h>
#include <DFPlayer_Mini_Mp3.h>

mySerial(2, 3);           // setting pin 2 sbg RX & pin 3 sbg TX
const int=4;

void setup ()
{
    pinMode (busyPin, INPUT);
    mySerial.begin (9600);

    mp3_set_serial (mySerial);      //set softwareSerial DFPlayer
    delay(10);

    mp3_reset();                  //soft-Reset modul DFPlayer
    delay(10);                   //tunggu 1ms untuk respon perintah

    mp3_set_volume (15);          //set Volume module DFPlayer
    delay(1000);
}

void loop ()
{
    mp3_play (1);                //Play mp3 0001.mp3 dalam folder mp3 di SD Card

    //memainkan file berikutnya jika lagu sebelumnya sudah selesai
    if( busyPin == HIGH )
    {
        mp3_next ();
    }

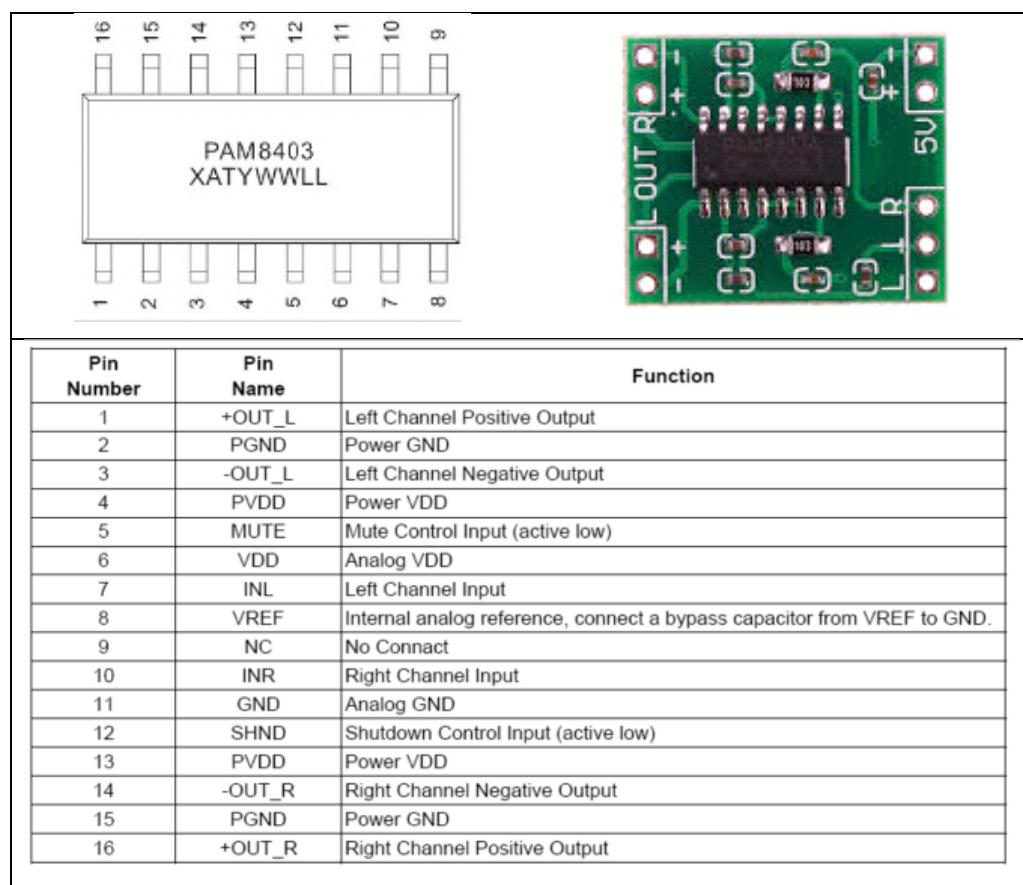
    //memainkan file 0001.mp3 s/d 0010.mp3 selama 2detik/file secara berurutan
    for (int i=1; i<=10; i++)
    {
        mp3_play (i);
        delay(2000);
    }
}
```

Gambar 2.12 Contoh *listing* program untuk menjalankan modul DF *Player* Mini.

## 2.4. *Audio Amplifier* Eksternal

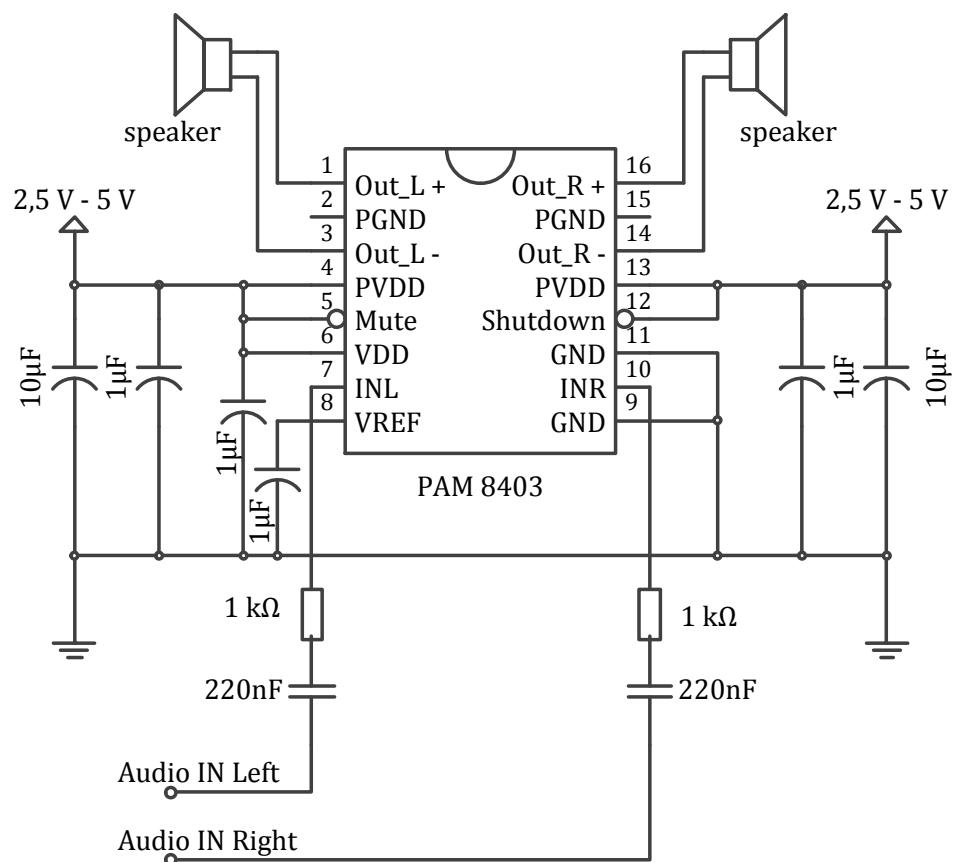
Modul *audio amplifier* eksternal pada sistem ini berfungsi untuk sebagai fasilitas opsional guna meningkatkan volume suara rekaman sistem jeda waktu iqamah ketika dirasakan tidak cukup keras. Modul *audio amplifier* yang dipergunakan pada sistem ini berbasis rangkaian terintegrasi (IC) PAM8403. IC PAM8403 merupakan *audio amplifier* kelas D dengan daya keluaran 3 W.

Suara yang dihasilkan oleh IC ini memiliki kualitas yang relatif tinggi. Keluaran IC ini dapat langsung dihubungkan dengan *loud speaker* secara langsung, tanpa perlu menggunakan sebuah filter frekuensi rendah tambahan pada sisi *output*. Hal ini membuat IC ini lebih ekonomis dan lebih kecil dimensinya. Pada Gambar 2.13. ditunjukkan *pinout* IC beserta dengan deskripsi setiap pin IC tersebut.



Gambar 2.13. *Pinout* IC PAM8403, modul PAM8403 dan deskripsi pin-pin IC.  
(<https://www.diodes.com/assets/Datasheets/PAM8403.pdf>)

Untuk menfungsikan IC PAM8403 menjadi sebuah *audio amplifier*, diperlukan beberapa komponen tambahan yaitu kapasitor dan resistor. *Audio amplifier* yang dihasilkan oleh IC ini merupakan *audio amplifier* stereo. Catu daya yang disuplaikan ke dalam rangkaian ini mulai dari 2,5 volt sampai dengan 5 volt. Pada Gambar 2.14. ditunjukkan rangkaian skematik dari modul *audio amplifier* berbasis IC PAM8403. Pin Audio IN dari modul ini dihubungkan dengan *output* dari modul MP3 *Player*.



Gambar 2.14. Rangkaian skematik modul *audio amplifier* berbasis IC PAM8403.

## BAB III

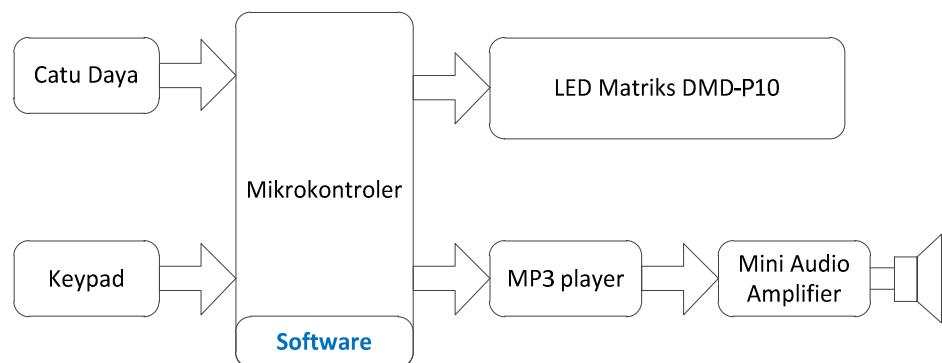
### PERANCANGAN DAN IMPLEMENTASI SISTEM

#### 3.1. Diagram Blok dan Cara Kerja Sistem

Sebelum melakukan perancangan dan implementasi sistem jeda iqamah, dilakukan wawancara dengan pihak DKM Masjid Jamie ARROHMAH dan studi literatur mengenai sistem jeda iqamah yang diperlukan. Dari hasil wawancara diperoleh bahwa sistem jeda iqamah yang diperlukan memiliki spesifikasi sebagai berikut:

- a. Sistem jeda iqamah dapat diatur melalui *keypad* dengan mudah
- b. Sistem jeda iqamah dapat di-set dalam satuan menit dari 1 menit sampai dengan 15 menit.
- c. Sistem jeda iqamah dapat diberhentikan sebelum pencacahan selesai melalui *keypad*.
- d. Sistem jeda iqamah memiliki *output* berupa *display* untuk menampilkan pencacahan dan berupa suara “aqimis salata”

Pada Gambar 3.1. ditunjukkan diagram blok sistem jeda iqamah berbasis Arduino. Sistem jeda iqamah ini terdiri dari beberapa modul perangkat keras yaitu modul *keypad*, modul kontroler, modul *display*, modul MP3 Player, modul *audio amplifier* dan modul catu daya. Keseluruhan sistem kerja sistem ini diatur menggunakan perangkat lunak yang ditempatkan pada modul kontroler.



Gambar 3.1. Diagram blok sistem jeda iqamah

Pada sistem ini *keypad* berfungsi untuk melakukan *setting* waktu jeda iqamah dalam satuan menit, memberhentikan pencacahan dan melakukan *reset* sistem. Sistem ini memiliki *setting* awal waktu jeda iqamah selama 5 menit (05:00). Setelah waktu jeda iqamah di-*setting* maka mikrokontroler akan menunggu penekan tombol **START** dari modul *keypad*.

Ketika tombol **START** ditekan, maka pencacahan dimulai. Proses pencacahan waktu mundur ditampilkan pada *display* LED matriks tipe DMD P-10. Pencacahan terus dilakukan sampai waktu menunjukkan 00:00. Proses pencacahan juga dapat dihentikan sebelum mencapai waktu 00:00 dengan menekan tombol **STOP** pada modul *keypad*.

Setelah pencacahan mencapai waktu 00:00 maka modul MP3 *Player* diaktifkan oleh mikrokontroler untuk menjalankan file suara “aqimis salata” sebagai tanda bahwa muadzin akan mengumandangkan iqamah untuk memulai shalat berjamaah. Sistem ini dilengkapi dengan modul *audio amplifier* untuk memperkeras suara *output* modul MP3 *Player*.

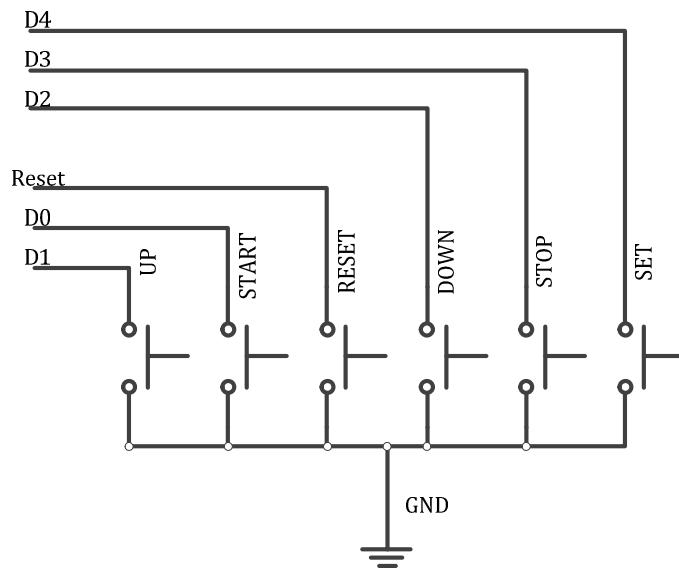
Catu daya sistem ini menggunakan sumber tegangan eksternal 8 VDC yang disuplai ke modul kontroler. Modul-modul lain disuplai catu dayanya dari modul kontroler setelah dilakukan pengubahan tegangan menjadi 5 VDC oleh modul kontroler.

### 3.2. Perancangan dan Implementasi Sistem

Seperti ditunjukkan pada Gambar 3.1., sistem jeda iqamah ini terdiri dari perangkat keras dan perangkat lunak. Modul perangkat keras yang diimplementasikan terdiri dari modul *keypad*, modul kontroler, modul *display*, modul MP3 *Player*, modul *audio amplifier* dan modul catu daya. Adapun perangkat lunak yang diimplementasikan difungsikan untuk mengatur sistem kerja seluruh modul yang mencakup: pembacaan *input* pengaturan pencacahan dari modul *keypad*, penampilan proses pencacahan pada modul *display* serta pengaktifan modul MP3 *Player*. Pada sub bab ini dijelaskan perancangan dan implementasi dari setiap modul sistem jeda iqamah.

### 3.2.1. Modul Keypad

Secara umum modul ini memiliki beberapa fungsi yaitu untuk melakukan *setting* waktu jeda iqamah dalam satuan menit, memberhentikan pencacahan dan melakukan *reset* sistem. Modul ini diimplementasikan dengan menggunakan enam buah *push button*, di mana setiap *push button* dihubungkan dengan pin-pin Arduino Nano. Skematik diagram rangkaian modul *keypad* ini ditunjukkan pada Gambar.3.2.



Gambar 3.2. Skematik diagram modul *keypad*

Rangkaian ini akan menghasilkan logika *low* jika salah satu *push button* ditekan. Modul ini diletakkan terpisah dari bagian *display* dikarenakan pihak DKM Masjid Jamie ARROHMAH menginginkan bahwa *setting* waktu iqamah dapat di-set dengan mudah, sehingga antara modul *keypad* dengan modul-modul lainnya dihubungkan dengan sebuah kabel.

Keenam tombol yang terdapat pada modul ini antara lain:

- Tombol **SET** berfungsi untuk memulai dan mengakhiri proses *setting* waktu jeda iqamah yang diinginkan.
- Tombol **UP** berfungsi untuk menaikkan pencacahan waktu jeda iqamah
- Tombol **DOWN** berfungsi untuk menurunkan pencacahan waktu jeda iqamah

- Tombol **START** berfungsi untuk memulai pencacahan mundur waktu jeda iqamah sesuai dengan *setting* waktu yang diberikan
- Tombol **STOP** berfungsi untuk memberhentikan pencacahan waktu jeda iqamah agar pencacahan langsung menuju ke akhir pencacahan yaitu 00:00
- Tombol **RESET** berfungsi untuk melakukan *reset* program ke *setting* awal.

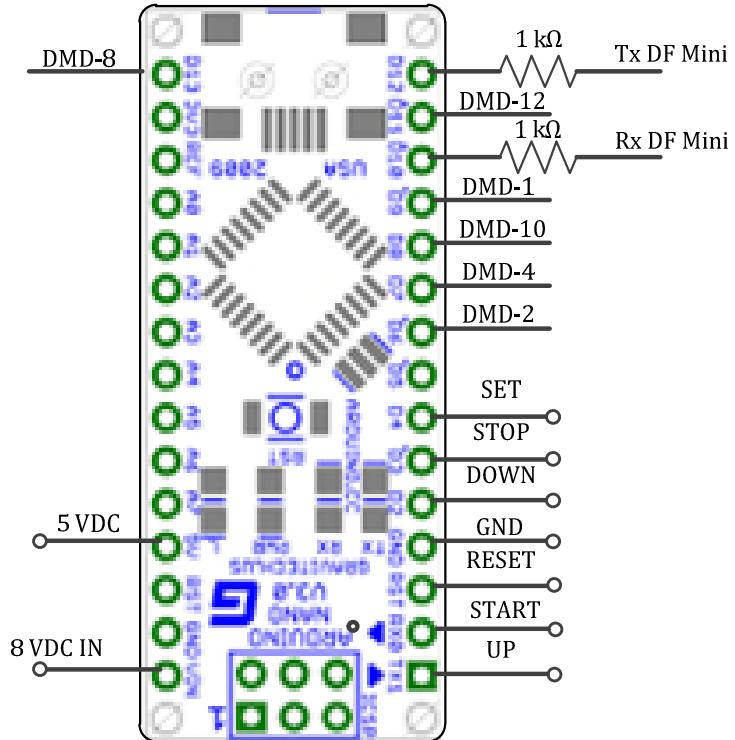
### 3.2.2. Modul Kontroler

Pemrosesan data pada sistem jeda iqamah ini menggunakan Arduino Nano. Selain kebutuhan pin-pin *input* dan *output* yang dapat terpenuhi, penggunaan Arduino Nano juga disebabkan karena dimensi dari modul ini yang relatif kecil. Modul kontroler dalam sistem ini melakukan beberapa fungsi yaitu:

- Membaca data *setting* waktu dan kontrol pencacahan dari modul *keypad*.
- Mengatur proses penampilan pencacahan mundur dari waktu jeda iqamah pada modul *display*.
- Mengatur proses aktifasi modul MP3 *player* ketika pencacahan waktu jeda iqamah telah selesai yaitu ketika pencacahan telah mencapai angka 00:00.

Modul kontroler ini dihubungkan dengan modul *keypad*, modul *display* dan modul MP3 *player*. Skematik diagram dari modul kontroler ini ditunjukkan pada Gambar 3.3. Modul ini disuplai dengan catu daya eksternal 8 VDC yang dihubungkan dengan pin VIN. Tegangan ini kemudian dikonversikan sehingga dapat menghasilkan tegangan 5 VDC yang digunakan untuk mensuplai tegangan modul *display*, modul MP3 *Player* dan modul audio amplifier. Adapun modul *keypad* hanya mengambil pin *ground* dari modul kontroler ini.

Adapun perangkat lunak pada modul ini menggunakan bahasa pemrograman *Arduino Software* (IDE). Dengan menggunakan perangkat lunak ini, maka memungkinkan untuk membuat Arduino menjadi sebuah *in system programming*. Proses pembuatan sintak program hingga *upload* program ke dalam memori mikrokontroler dapat dilakukan dalam satu perangkat lunak ini



Gambar 3.3. Skematik diagram modul kontroler

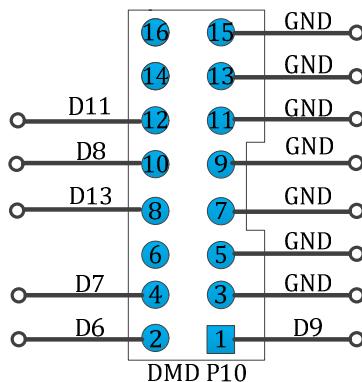
### 3.2.3. Modul *Display*

Fungsi utama modul *display* pada sistem ini adalah untuk menampilkan proses pencacahan mundur waktu jeda iqamah dari *setting* waktu diberikan. Di samping itu, modul ini juga berfungsi untuk menampilkan *setting* waktu yang diinginkan melalui modul *keypad*. Penyalaan modul *display* ini diatur oleh modul kontoler Arduino Nano.

Implementasi modul *display* pada sistem ini menggunakan LED Matriks DMD P10. Penggunaan tipe *display* ini dikarenakan kemudahan dalam melakukan *interfacing* dan pemrograman modul ini dengan menggunakan Arduino. *Library* untuk pemrograman modul ini dengan menggunakan Arduino telah banyak tersedia. Adapun alasan teknik lainnya adalah masalah catu daya digunakan untuk modul *display* tipe ini adalah sama dengan kebutuhan catu daya modul-modul

lainnya yaitu 5 Volt DC, serta masalah kebutuhan jumlah pin yang digunakan modul *display* ini hanya menggunakan 6 pin digital Arduino.

Tipe dimensi *display* DMD P10 yang digunakan pada sistem ini adalah 16 LED x 32 LED. Untuk menghubungkan modul *display* dengan modul kontroler dilakukan melalui soket 16 pin modul *display*. Adapun skematik diagram hubungan soket modul *display* dengan modul kontroler Arduino Nano ditunjukkan pada Gambar 3.4.



Gambar 3.4. Skematik diagram soket modul *display* DMD P10

Pin 1 (*Output Enable*) modul ini dihubungkan dengan pin digital 9 dari modul Arduino Nano. Pin 2 dan Pin 4 sebagai modul pemilih kolom A dan B modul *display* dihubungkan secara berturut-turut dengan pin digital 6 dan pin digital 7 modul Arduino Nano. Pin 8 sebagai *SPI clock* modul *display* DMD P10 dihubungkan dengan pin digital 13 modul Arduino Nano and Pin 10 sebagai *Latch data register* modul *display* DMD P10 dihubungkan dengan pin digital 8 modul Arduino Nano. Adapun pin data modul *display* (pin 12) dihubungkan dengan pin digital 11 Arduino Nano. Tujuh pin dari modul *display* yaitu pin 3, pin 5, pin 7, pin 9, pin 11, pin 13, pin 15 dihubungkan dengan pin *Ground* modul Arduino Nano.

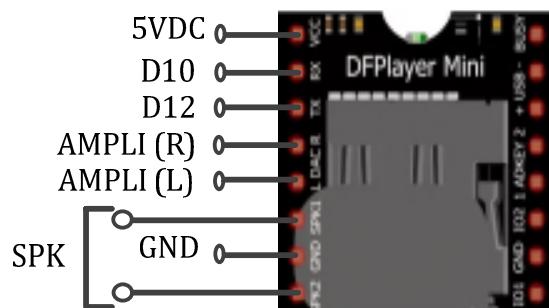
### 3.2.4. Modul MP3 Player

Dikarenakan sistem jeda iqamah yang diimplementasikan ini memiliki spesifikasi *output* suara berupa seruan kalimat “aqimis salata” ketika pencacahan jeda iqamah selesai, maka sistem ini dihubungkan dengan sebuah modul MP3

*Player* untuk memainkan rekaman suara yang diinginkan. Suara tersebut direkam dalam bentuk format MP3 dan disimpan dalam *SD Card* dengan penamaan file tertentu.

Modul MP3 *Player* yang digunakan pada sistem ini adalah tipe **DFPlayer Mini**. Penggunaan modul tipe ini dikarenakan beberapa hal yaitu: dimensi yang relatif kecil, pemrograman yang relatif mudah dan memiliki spesifikasi kebutuhan catu daya seperti modul-modul pendukungan sistem jeda iqamah lainnya (5 VDC).

Untuk memainkan file tersebut, modul ini dikontrol oleh modul Arduino Nano. Pin yang diperlukan oleh modul ini hanya dua buah pin yaitu pin RX dihubungkan dengan pin digital 10 modul Arduino Nano dan pin TX dihubungkan dengan pin digital 12 modul Arduino Nano. Hubungan skematik diagram modul MP3 *Player* dengan modul Arduino ditunjukkan pada Gambar 3.5.



Gambar 3.5. Skematic diagram modul MP3 *Player* dengan modul Arduino Nano

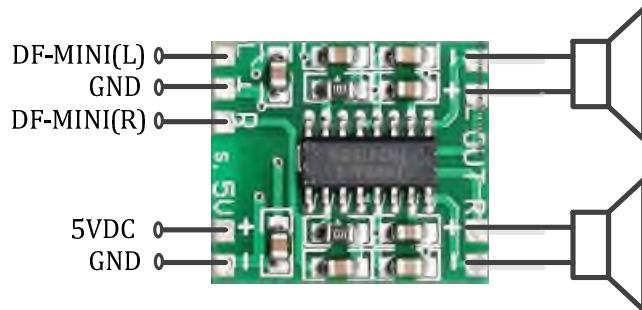
File suara rekaman yang tersimpan pada *SD Card* akan dijalankan ketika pencacahan menunjukkan angka 00:00 (terlihat pada modul *display*). Modul Arduino diprogram untuk memainkan file tersebut sesuai dengan nama file suara yang diberikan. Lamanya pemutaran file tersebut diukur secara manual berdasarkan panjang rekaman tersebut. Lamanya waktu pemutaran suara tersebut akan diprogram berupa *delay* pada modul Arduino Nano.

Suara yang dihasilkan oleh modul MP3 *Player* ini sebenarnya dapat langsung didengarkan melalui *loud speaker* yang dihubungkan pada pin SPK\_1 dan pin SPK\_2. Namun, jika dirasa suara yang dihasilkan kurus begitu keras maka

dapat dilakukan dengan mengatur besarnya volume melalui sintak program modul Arduino Nano atau dengan menambahkan modul *audio amplifier* yang dihubungkan dengan pin DAC\_R dan pin DAC\_L.

### 3.2.5. Modul *Audio Amplifier*

Modul *audio amplifier* dalam sistem ini bertujuan untuk memperbesar suara yang dihasilkan oleh modul MP3 *Player*, di mana jika dirasakan suara yang dihasilkan oleh modul MP3 *Player* tidak terlalu keras. Pada sistem ini digunakan mini *audio amplifier* tipe **PAM8403**. Modul ini bersifat *amplifier* stereo, di mana dua buah saluran *input* modul ini (L dan R) dihubungkan masing-masing pada dua buah saluran *output* modul MP3 *Player* (DAC\_L dan DAC\_R). Hubungan skematik antara modul MP3 *Player* dengan modul *audio amplifier* ditunjukkan pada Gambar 3.6. Pertimbangan penggunaan modul ini pada sistem jeda iqamah yaitu dikarenakan dimensi modul ini yang relatif kecil dan suplai tegangan yang sama dengan modul-modul lainnya yaitu tegangan 5 VDC.

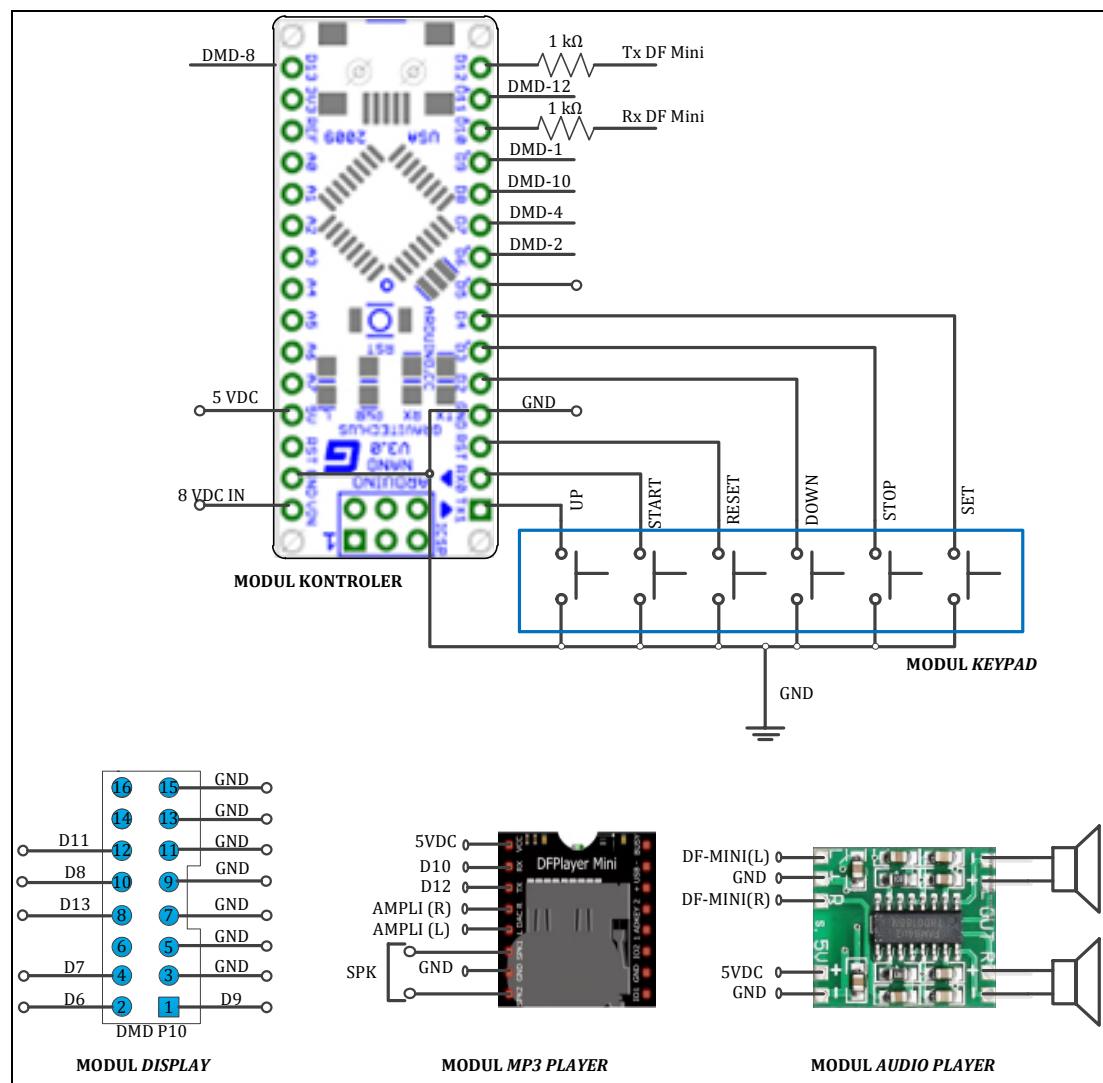


Gambar 3.6. Skematic diagram modul *audio amplifier*

### 3.2.6. Modul Catu Daya

Suplai catu daya yang digunakan pada sistem ini menggunakan adaptor *switching* 8 VDC 2 ampere yang telah tersedia di pasaran. Suplai tegangan 8VDC ini dihubungkan dengan modul kontroler Arduino Nano melalui pin VIN. Selain untuk mencatu daya modul kontroler, sumber tegangan ini dikonversi menjadi tegangan 5 VDC untuk mensuplai modul-modul lainnya.

Pada Gambar 3.7. ditunjukkan diagram skematik sistem jeda iqamah yang terdiri dari modul *keypad*, modul kontroler, modul *display*, modul *MP3 Player*, dan modul *audio amplifier*. Modul catu daya tidak digambarkan pada diagram skematik ini. Saluran catu daya dihubungkan pada pin 8 VDC IN dari modul kontroler.



Gambar 3.7. Diagram skematik sistem jeda iqamah

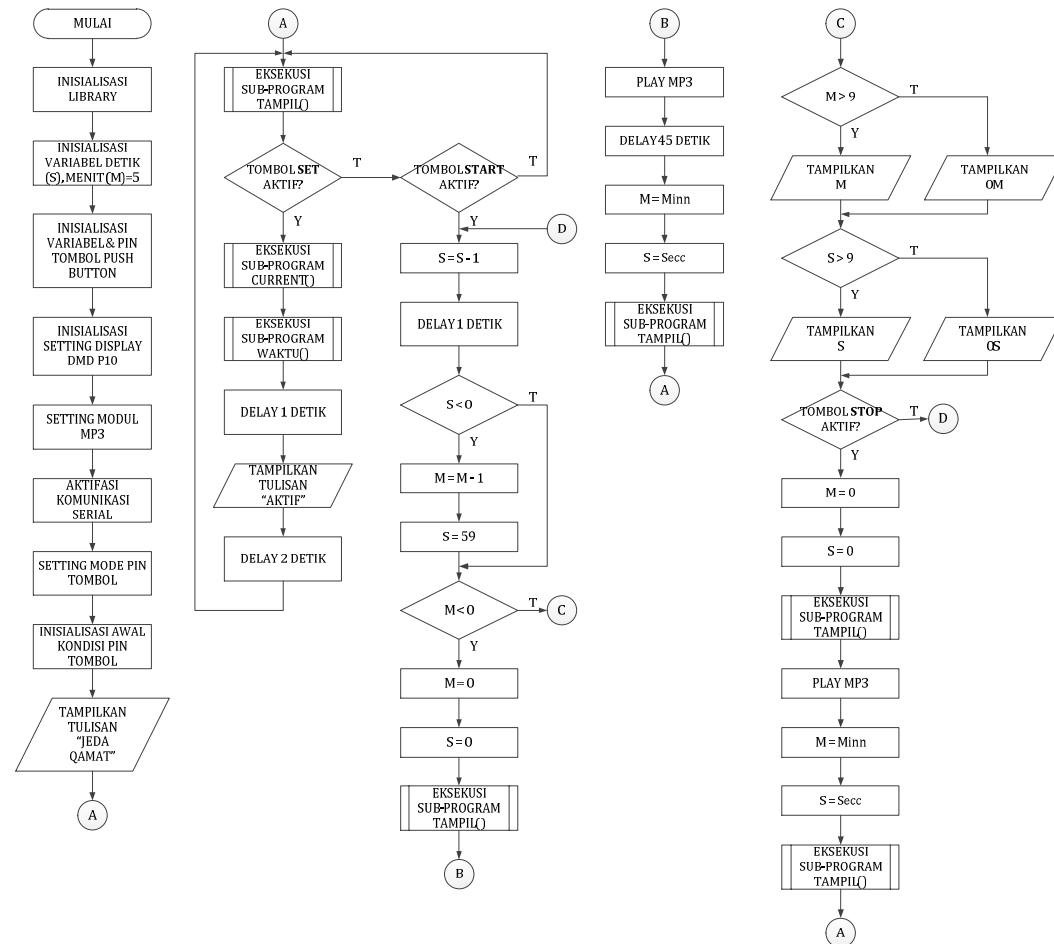
### 3.2.7. Perangkat Lunak Sistem

Terdapat beberapa fungsi yang dikerjakan oleh perangkat lunak pada sistem ini yaitu membaca *input* pengaturan pencacahan waktu dari modul *keypad*, mengatur dan menampilkan proses pencacahan waktu pada modul *display* serta mengaktifkan modul *MP3 Player*. Perangkat lunak pada sistem ini diimplementasikan dengan bahasa pemrograman C untuk sistem minimum Arduino. Perangkat lunak yang diimplementasikan pada sistem ini terdiri dari satu program utama dan tiga buah sub program yaitu sub program *tampil()*, sub program *current()* dan sub program *waktu()*. Keempat program ini akan dijelaskan masing-masing dengan menggunakan diagram alir program.

#### a. Program Utama

Diagram alir program utama sistem ini ditunjukkan pada Gambar 3.8. Program dimulai dengan melakukan inisialisasi sejumlah *library* yang digunakan pada sistem ini yaitu *library* untuk pengatutan EEPROM <EEPROM.h>, *library* untuk pengaturan MP3 *Player* yang terdiri dari: <SoftwareSerial.h> dan <DFPlayer\_Mini\_Mp3.h> serta *library* untuk pengaturan *display* DMD P-10 yang terdiri dari <SPI.h>, <DMD2.h>, <fonts/SystemFont5x7.h>, <fonts/Arial14.h> dan <fonts/Droid\_Sans\_12.h>. Kemudian dilanjutkan dengan melakukan inisialisasi variabel untuk menyimpan pencacahan waktu detik (S) dan pencacahan waktu menit (M). Secara *default*, sistem ini di-setting untuk melakukan pencacahan 5 (lima) menit sebagai jeda waktu iqamah.

Selanjutnya dilakukan inisialisasi variabel dan pin-pin yang digunakan untuk modul *keypad*. Terdapat enam tombol *push button* yang masing-masing difungsikan untuk memulai pencacahan waktu jeda iqamah (tombol **start** di pin 0), menaikkan *setting* waktu menit pencacahan jeda iqamah (tombol **up** di pin 1), menurunkan *setting* waktu menit pencacahan jeda iqamah (tombol **down** di pin 2), menghentikan pencacahan waktu jeda iqamah (tombol **stop** di pin 3), dan mengaktifkan fungsi *setting* waktu jeda iqamah (tombol **set** di pin 4). Untuk tombol **reset**, tidak dilakukan pemograman tetapi langsung mengambil fungsi dari modul kontroler Arduino Nano.



Gambar 3.8. Diagram alir program utama sistem waktu jeda iqamah

Inisialisasi dilakukan juga untuk setting *display* DMD P10 yang terdiri dari *setting* dimensi *display* dan *setting* jenis-jenis huruf. Kemudian dilanjutkan pada *setting* komunikasi serial yang digunakan untuk modul MP3 di mana pin 12 digunakan sebagai *receiver* dan pin 10 digunakan sebagai *transmitter*. Selanjutnya dilakukan *setting* beberapa variabel yang digunakan dalam program. Potongan *listing* program untuk inisialisasi ini ditunjukkan pada Gambar 3.9.

```
-----inisialisasi library-----
#include<EEPROM.h>
#include <SoftwareSerial.h>
#include <DFPlayer_Mini_Mp3.h>
#include <SPI.h>
#include <DMD2.h>
#include <fonts/SystemFont5x7.h>
#include <fonts/Arial14.h>
#include <fonts/Droid_Sans_12.h>
-----inisialisasi variabel detik dan menit -----
int S = 0;                                // count seconds
int M = 5;                                 // count minutes (default 5 minutes)
-----inisialisasi variabel dan pin tombol push button -----
int start = 0;
int up = 1;                                //increment minute
int down = 2;                               //decrement minute
int stop = 3;                                //stop counting down
int set = 4;                                //Mode setting
-----inisialisasi setting display DMD P10 -----
const int WIDTH = 1;
SoftDMD dmd(WIDTH,1);                      // DMD controls the entire display
DMD_TextBox box(dmd, 2, 4, 32, 16); // Set Box (dmd, x, y, Height, Width)
const uint8_t *FONT2 = Arial14;
const uint8_t *FONT1 = SystemFont5x7;
const uint8_t *FONT = Droid_Sans_12;

SoftwareSerial mySerial(12,10);      ///(Rx, Tx)

int off = 0;
int Minn, Secc;
int tmp,mIn,add=11;
const char *MESSAGE = "Jeda Iqamah.....C.2.I.....";
```

Gambar 3.9. Potongan *listing* program inisialisasi program

Pada sub program *setup( )*, dimulai dengan melakukan aktifasi komunikasi serial, melakukan pengaturan mode pin-pin tombol *keypad* dan melakukan pengaturan kondisi awal pin-pin tombol *keypad*. Kemudian program melakukan

*setting* untuk *display* DMD P10 dan menampilkan tulisan yang telah di-*setting* pada karakter MESSAGE yaitu "Jeda Iqamah.....C.2.I.....". Setelah tulisan tersebut selesai ditampilkan maka akan diberi waktu delay selama 2 detik kemudian *display* di non-aktifkan sementara sampai melakukan eksekusi perintah program selanjutnya. Potongan awal program *setup()* ini ditunjukkan pada Gambar 3.10.

```
void setup()
{
  //-----setting komunikasi serial MP3
  mySerial.begin (9600);
  mp3_set_serial (mySerial);
  delay(10);
  //-----setting mode pin
  pinMode(start, INPUT);
  pinMode(up, INPUT);
  pinMode(down, INPUT);
  pinMode(stop, INPUT);
  pinMode(set, INPUT);

  //-----inisialisasi awal push button
  digitalWrite(start, HIGH);
  digitalWrite(up, HIGH);
  digitalWrite(down, HIGH);
  digitalWrite(stop, HIGH);
  digitalWrite(set, HIGH);
  //-----setting display P10
  dmd.setBrightness(255);
  dmd.selectFont(FONT);
  dmd.begin();
  const char *next = MESSAGE;
  while(*next) {
    box.print(*next);
    delay(500);
    next++;
  }
  delay(2000);
  dmd.drawFilledBox(0,0,31,15, GRAPHICS_OFF);
```

Gambar 3.10. Potongan *listing* awal program prosedur *setup()*

Sub program *setup()* kemudian dilanjutkan pada label **cek** sebagai awal untuk melakukan eksekusi sub program *tampil()*. Fungsi dari sub program *tampil()* ini secara umum adalah untuk menampilkan *setting* awal menit dan detik sebelum dilakukan

perintah aktifasi memulai pencacahan atau *setting* pencacahan waktu jeda iqamah oleh pengguna. Secara detail diagram alir sub program akan dijelaskan secara terpisah.

Sebelum pencacahan dimulai, program melakukan pengecekan tombol **set** dan tombol **start**. Potongan program pengecekan tombol **set** dan tombol **start** ini ditunjukkan pada Gambar 3.11. Jika tombol **set** ditekan maka program akan melakukan *setting* waktu jeda iqamah yang baru dengan mengaktifkan sub program *current()* dan sub program *waktu()*. Fungsi sub program *current()* hampir sama seperti fungsi sub program *tampil()*. Adapun sub program *waktu()* berfungsi untuk mengatur penambahan dan pengurangan waktu menit. *Setting* waktu menit yang baru disimpan pada variabel **Minn**. Selama melakukan pengaturan waktu yang baru, *display* menampilkan tulisan “**Aktif**”. Prosedur ini terus dieksukusi sampai dengan ada penekan tombol **set** kembali. Jika tombol **start** ditekan pada saat awal pengecekan tombol atau setelah prosedur *setting* waktu selesai dilakukan, maka program memulai proses pencacahan waktu jeda iqamah dengan mengeksekusi program yang diberi label **mulai**.

cek:

```
dmd.selectFont(FONT);
tampil();
if(digitalRead(set) == 0)
{
    current();
    waktu();
    delay(1000);
    dmd.selectFont(FONT1);
    dmd.drawString(3,0,String("Aktif"));
    delay(2000);
    dmd.drawFilledBox(0,0,31,15, GRAPHICS_OFF);
    goto cek;
}
else
{
    if(digitalRead(start) == 0)
    {
        goto mulai;
    }
    else
    {
        goto cek;
    }
}
```

Gambar 3.11. Potongan *listing* program pengecekan tombol **mode** dan tombol **start**

Program yang berada pada label **mulai** ini berfungsi untuk memulai pencacahan mundur dengan mengurangi setiap detik (1000 ms) dari *setting* waktu menit dan detik yang telah ditetapkan. Jika waktu *setting* telah menunjukkan nilai/angka 0 pada nilai menit dan detik, maka rekaman suara “aqimih salata” pada modul MP3 *Player* akan diaktifkan. Aktifasi modul MP3 *Player* ini diatur dengan perintah delay 45 detik sesuai dengan durasi file MP3. Setelah rekaman suara dijalankan, program akan menampilkan *setting* menit dan detik ke *setting* awal sebelum pencacahan, kemudian program kembali label **cek**. Potongan program pengurangan waktu setiap detik ini ditunjukkan pada Gambar 3.12.

```
mulai:  
{  
    dmd.selectFont(FONT);  
    dmd.drawFilledBox(0,0,31,15, GRAPHICS_OFF);  
  
    putar:  
        dmd.drawString(16,3,(":"));  
  
        S--;  
        delay(1000);  
  
        if(S<0)  
        {  
            M--;  
            S=59;  
        }  
        if(M<0)  
        {  
            M=0;  
            S=0;  
            tampil();  
            mp3_set_volume (25);  
            delay(10);  
            mp3_play ();  
            delay(10);  
            mp3_play (1);  
            delay(45000);  
            dmd.drawFilledBox(0,0,31,15, GRAPHICS_OFF);  
            M=Minn;  
            S=Secc;  
            tampil();  
            goto cek;  
        }  
}
```

Gambar 3.12. Potongan *listing* program pengurangan waktu setiap detik

Akan tetapi jika *setting* waktu menit belum menunjukkan angka 0, maka akan dilakukan pengurangan terus sambil dilakukan pengecekan tombol **stop**. Potongan program pengecekan tombol **stop** ini ditunjukkan pada Gambar 3.13.

```
if(M>9)
{
    dmd.drawString(2,4,String(M));
}
else
{
    dmd.drawString(2,4,String(0));
    dmd.drawString(9,4,String(M));
}
if(S>9)
{
    dmd.drawString(19,4,String(S));
}
else
{
    dmd.drawString(19,4,String(0));
    dmd.drawString(26,4,String(S));
}

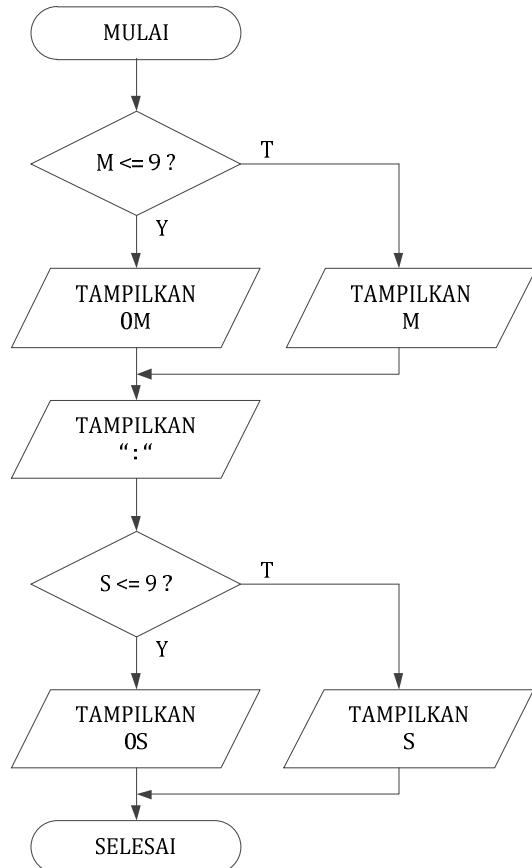
if(digitalRead(stop) == 0)
{
    M=0;
    S=0;
    tampil();
    mp3_set_volume (25);
    delay(10);
    mp3_play ();
    delay(10);
    mp3_play (1);
    delay(45000);
    dmd.drawFilledBox(0,0,31,15, GRAPHICS_OFF);
    M=Minn;
    S=Secc;
    tampil();
    goto cek;
}
else
{
    goto putar;
}
} //kurawal tutup untuk mulai:
} //kurawal tutup untuk setup()
```

Gambar 3.13. Potongan *listing* program pengecekan tombol **stop**

Tombol **stop** ini merupakan fungsi untuk menghentikan pencacahan ketika pencacahan detik dan menit belum mencapai angka 0. Jika tombol **stop** ini ditekan maka *display* akan menampilkan angka 0 dan menjalankan file MP3. Selanjutnya program kembali menampilkan *setting* menit dan detik ke *setting* awal sebelum pencacahan, kemudian program kembali label **cek**. Akan tetapi jika tombol **stop** ini tidak ditekan maka program akan mengurangi nilai waktu detik seperti yang dilakukan pada potongan program pada Gambar 3.12 (label **putar**).

#### b. Sub Program Tampil

Pada Gambar 3.14. ditunjukkan diagram alir sub program *tampil()*. Sub program ini berfungsi untuk mengatur tampilan menit (M) dan detik (S) pada *setting* awal pada *display* DMD P10.



Gambar 3.14. Diagram alir sub program *tampil()*

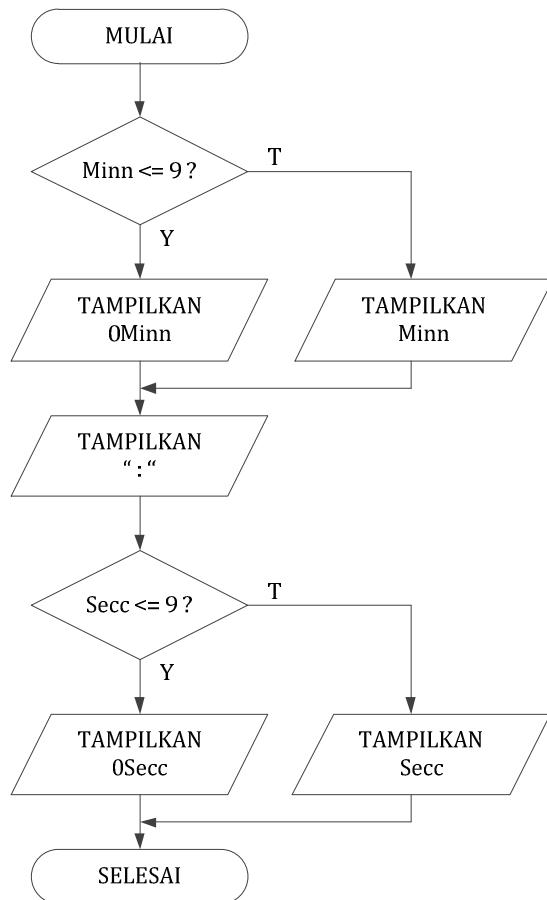
Pada program ini, tampilan menit dan detik akan ditambahkan angka 0 (nol) pada digit depan bilangan jika nilai menit atau detik kurang dari atau sama dengan 9 (sembilan). Akan tetapi jika nilai menit atau detik lebih dari 9 (sembilan), maka tampilannya tidak ditambahkan angka 0 (nol) pada digit depan bilangan. Dengan adanya sub-program ini maka tampilan detik dan menit selalu ditampilkan dalam bentuk dua digit. Potongan program sub program **tampil( )** ditunjukkan pada Gambar 3.15.

```
void tampil()
{
    if(M<=9)
    {
        dmd.drawString(2,4,String(0));
        dmd.drawString(9,4,String(M));
    }
    else
    {
        dmd.drawString(2,4,String(M));
    }
    dmd.drawString(16,3,(":"));
    if(S<=9)
    {
        dmd.drawString(19,4,String(0));
        dmd.drawString(26,4,String(S));
    }
    else
    {
        dmd.drawString(19,4,String(S));
    }
}
```

Gambar 3.15. Potongan *listing* program sub program **tampil( )**

### c. Sub Program Current

Diagram alir sub program **current( )** ditunjukkan pada Gambar 3.16. Sub program ini fungsinya sama seperti sub program **tampil( )** yaitu untuk mengatur tampilan menit dan detik pada *display* DMD P10. Namun yang membedakan adalah variabel menit dan detik yang ditampilkan merupakan nilai menit (**Minn**) dan detik (**Secc**) dari hasil *setting* waktu melalui tombol **set**.



Gambar 3.16. Diagram alir sub program **current()**

Pada program ini juga sama yaitu untuk menambahkan angka 0 (nol) di depan digit bilangan menit dan detik jika nilai menit dan detik tersebut bernilai kurang dari atau sama dengan 9 (sembilan). Hal ini untuk membuat tampilan detik dan menit selalu ditampilkan dalam bentuk dua digit. Potongan program sub program **current()** ditunjukkan pada Gambar 3.17.

```

void current()
{
    if(Minn<=9)
    {
        dmd.selectFont(FONT1);
        dmd.drawString(2,8,String(0));
        dmd.drawString(8,8,String(Minn));
    }
    else
    {
        dmd.selectFont(FONT1);
        dmd.drawString(2,8,String(Minn));
    }
    dmd.selectFont(FONT1);
    dmd.drawString(14,8,:");
}

if(Secc<=9)
{
    dmd.selectFont(FONT1);
    dmd.drawString(20,8,String(0));
    dmd.drawString(26,8,String(Secc));
}
else
{
    dmd.selectFont(FONT1);
    dmd.drawString(20,8,String(Secc));
}

```

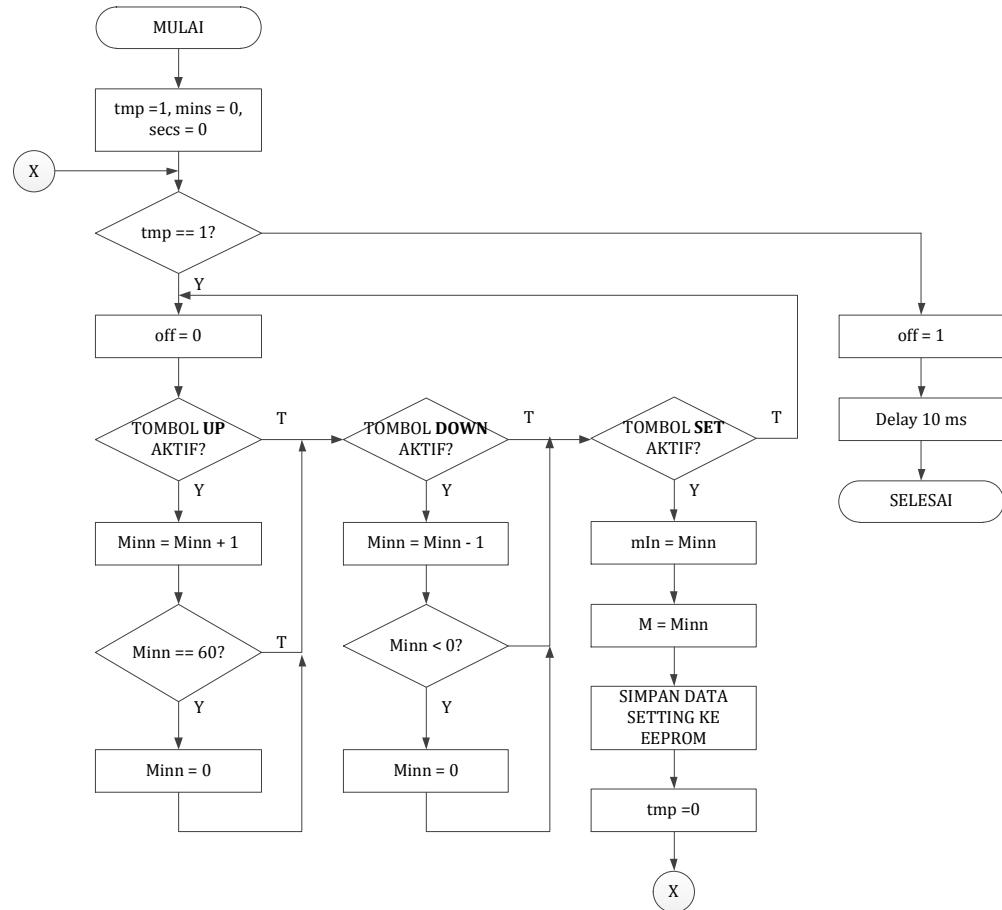
Gambar 3.17. Potongan *listing* program sub program *current()*

#### d. Sub Program Waktu

Sub program *waktu()* berfungsi untuk melakukan pengaturan *setting* baru waktu menit sesuai dengan yang diinginkan pengguna. Sub program ini dijalankan ketika tombol **set** ditekan. Jadi ketika tombol **set** ditekan, maka program selanjutnya memeriksa kondisi tiga tombol lainnya yaitu tombol **up**, tombol **down** dan tombol **set**. Pada Gambar 3.18 ditunjukkan diagram alir sub program *waktu()*.

Jika tombol **up** aktif, maka variabel menit (**Minn**) akan ditambah sampai dengan maksimum 60 menit. Adapun jika tombol **down** aktif, maka variabel menit (**Minn**) akan dikurangi sampai dengan minimum 0 menit. *Setting* waktu menit diakhiri dengan menekan tombol **set**. Setelah tombol **set** ditekan maka data *setting* waktu menit pada variabel **Minn** akan disimpan pada variabel menit (**M**)

dan disimpan juga pada EEPROM. Potongan sub program *waktu( )* ditunjukkan pada Gambar 3.19.



Gambar 3.18. Diagram alir sub program *waktu( )*

```

void waktu()
{
    int tmp=1,mins=0,hors=0,secs=0;
    while(tmp==1)
    {
        off=0;
        if(digitalRead(naik)==0)
        {
            Minn++;
            if(Minn==60)
            {
                Minn=0;
            }
        }
        if(digitalRead(turun)==0)
        {
            Minn--;
            if(Minn<0)
            {
                Minn=0;
            }
        }
        dmd.drawFilledBox(0,0,31,15, GRAPHICS_OFF);
        dmd.selectFont(FONT1);
        dmd.drawString(8,0,String("Set"));
        current();
        delay(200);

        if(digitalRead(mode)==0)
        {
            mIn=Minn;
            M=Minn;
            EEPROM.write(add++,mIn);
            tmp=0;
            while(digitalRead(mode)==0);
        }
    } //kurawal tutup utk While

    off=1;
    delay(10);
} //kurawal tutup utk waktu()

```

Gambar 3.19. Potongan *listing* program sub program *waktu()*

## BAB IV

### HASIL KEGIATAN

Setelah perancangan sistem selesai diimplementasikan, selanjutnya sistem dikemas ke dalam *frame*. Pengemasan sistem dibagi menjadi dua yaitu *casing display* dan *casing tombol*. Kedua *casing* ini dihubungkan dengan sebuah kabel dengan konektor jenis DB9. Kedua *casing* tersebut ditunjukkan pada Gambar 4.1.



Gambar 4.1. Hasil implementasi sistem jeda waktu iqamah

Pada *casing display* terdapat modul-modul sebagai berikut: modul kontroler, modul *display*, modul *MP3 Player*, modul *audio amplifier* (termasuk *loud speaker*) dan modul catu daya. Pada Gambar 4.2 ditunjukkan bagian dalam dari *casing display*. Adapun modul *keypad* diletakkan pada bagian *casing tombol* secara terpisah, seperti ditunjukkan pada Gambar 4.3. Tombol Reset diletakkan terpisah pada bagian atas agar terhindar dari kesalahan penekanan ketika dilakukan *setting* atau ketika aktifasi sistem pencacahan jeda waktu iqamah.



Gambar 4.2.Tampak dalam *casing display*



Gambar 4.3.Tampak depan *casing tombol*

Setelah dikemas ke dalam *casing*, sistem kemudian diuji dengan melakukan beberapa fungsi kerja sistem yang mencakup fungsi setiap tombol, fungsi *setting* sistem, fungsi *start* dan *stop* sistem, fungsi *reset* sistem dan fungsi MP3 *Player*. Dari hasil pengujian ini diperoleh bahwa dikhawatirkan bahwa suara MP3 *Player* dengan loud speaker yang telah terpasang terlalu kecil untuk ukuran **Masjid Jamie ARROHMAH**. Untuk mengatasi hal tersebut, maka ditambahkan speaker aktif yang sifatnya opsional. Speaker aktif ini ditempatkan di luar *casing displays* seperti ditunjukkan pada Gambar 4.4.



Gambar 4.4.Sistem jeda waktu iqamah dengan speaker eksternal

Setelah sistem diuji, kemudian sistem diserahkan kepada pihak DKM **Masjid Jamie ARROHMAH**. Prosesi serah terima sistem ini ditunjukkan pada Gambar 4.5. Pemasangan sistem ini di dinding masjid akan dilakukan oleh pihak DKM. Proses evaluasi kinerja sistem akan dilakukan juga oleh pihak DKM dan akan dilaporkan kembali jika terdapat kekurangan atau kerusakan sistem.



Gambar 4.5. Dokumentasi serah terima sistem jeda waktu iqamah kepada pihak DKM  
Masjid Jamie ARROHMAH

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **4.1. Kesimpulan Hasil Kegiatan**

Dari hasil perancangan dan implementasi sistem jeda waktu iqamah ini dapat disimpulkan beberapa hal yaitu:

1. Program untuk *looping* pada sistem ini dapat bekerja dengan baik walaupun tanpa menggunakan sub program *loop( )* pada pemrograman Arduino.
2. Catu daya eksternal dengan arus 2 Ampere dapat mencukupi kebutuhan daya seluruh sistem.
3. Fungsi tombol-tombol pada modul *keypad* dapat berfungsi baik dengan penambahan kabel lebih kurang 2 meter.

#### **4.2. Saran Hasil Kegiatan**

Dari hasil kegiatan ini, terdapat beberapa hal yang dikembangkan terhadap sistem sebagai langkah pengembangan lebih lanjut yaitu diantaranya:

- a. Pengaturan waktu dapat menggunakan sistem *wireless* berbasis sistem komunikasi data melalui *infrared* atau komunikasi radio melalui sebuah tombol-tombol pada sistem *remote control*.
- b. Sistem dapat ditambahkan modul RTC (*Real Time Clock*) untuk menampilkan jam digital jika sedang tidak digunakan sebagai sistem jeda waktu iqamah.
- c. Penambahan suara rekaman adzan dapat pula ditambahkan pada sistem ini yang dapat dijalankan sesuai *database* waktu adzan yang tersimpan pada memori Arduino sebagai *look up table* atau menambahkan modul memori eksternal.

## **DAFTAR PUSTAKA**

- [1] [www.theengineeringprojects.com](http://www.theengineeringprojects.com)
- [2] <https://www.tokopedia.com/klinikrobot/kr04110-p10-outdoor-waterproof-dual-color-led-module-320x160mm>
- [3] <https://www.instructables.com/id/Display-Text-at-P10-LED-Display-Using-Arduino/>
- [4] <https://pcccontrol.wordpress.com/2016/04/30/pengetahuan-dasar-pemrograman-modul-leddot-matrik-display-dmd-p10-dengan-arduino/>
- [5] <http://www.picaxe.com/docs/spe033.pdf>
- [6] <http://www.belajarduino.com/2016/10/arduinop.html>
- [7] <https://www.diodes.com/assets/Datasheets/PAM8403.pdf>

**LAMPIRAN A  
SURAT PERMOHONAN**



**MASJID JAMIE ARROHMAH**  
**MAJELIS YA ARHAMARROHIMIN**  
Jl. Cihanjuang H. Masduki RT 02/03 Ds. Cihanjuang Kec.  
Parongpong Kab. Bandung Barat 40559  
Telp. 081321797841/081573094648



Nomor : 01/MJAR-PR/2020  
Lamp : -  
Hal : Permohonan

Kepada Yth.  
Kepala Laboratorium Elektronika ITENAS

*Assalamu'alaikum Wr.Wb.*

Salam berkah berlimpah kami sampaikan semoga segala aktifitas kita senantiasa berada di dalam lindungan Allah SWT. Aamiin.

Dengan ini, kami atas nama pengurus Dewan Kemakmuran Masjid (DKM) Masjid Jamie ARROHMAH mengajukan permohonan pembuatan, 1 unit alat untuk menjeda antara adzan dengan iqomah yang akan kami fungsikan sebagai alat bantu tertibnya ibadah di Mesjid Jamie ARROHMAH.

Demikian surat permohonan ini kami buat, atas perhatiannya saya ucapkan terima kasih.

*Jazakumuloh Ahsanal Jaza.*

*Wassalamu'alaikum Wr.Wb.*  
Karangsari, 19 Januari 2020

Ketua DKM  
  
H. Tatang Sutisna

Ketua Majelis  
  
M. Tajul Arifin, S.Sos.I