



**RANCANG BANGUN SISTEM PENGINGAT JADWAL DAN
PENGUMUMAN OTOMATIS DI SD PANCA BUDI MEDAN**

Disusun dan Diajukan untuk Memenuhi Persyaratan Ujian Akhir Memperoleh Gelar Sarjana

Komputer pada Fakultas Sains dan Teknologi

Universitas Pembangunan Panca Budi

Medan

SKRIPSI

OLEH:

NAMA : CUT YUNI AULIANI ERSA

NPM : 1824370191

PROGRAM STUDI : SISTEM KOMPUTER

**PROGRAM STUDI SISTEM KOMPUTER
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS PEMBANGUNAN PANCA BUDI MEDAN
2020**

ABSTRAK

**CUT YUNI AULIANI ERS
RANCANG BANGUN SISTEM PENGINGAT JADWAL DAN
PENGUMUMAN OTOMATIS DI SD PANCA BUDI MEDAN
2020**

Untuk saat ini masih banyak sekolah yang masih menggunakan cara manual untuk mengingatkan jadwal kegiatan dan memberi pengumuman dengan cara guru piket memberitahukan didepan microfon. Semakin seringnya tamu atau orang tua murid yang datang kesekolah dan kurangnya guru piket, sehingga mengakibatkan keterlambatan dalam memberikan pengumuman. Tujuan dari Tugas Akhir ini adalah membuat pengingat jadwal dan pengumuman otomatis yang sehingga guru piket dapat melakukan pekerjaannya yang lain dengan lebih fokus. Berdasarkan permasalahan diatas dengan demikian penulis membuat Tugas Akhir dengan judul "RANCANG BANGUN SISTEM PENGINGAT JADWAL DAN PENGUMUMAN OTOMATIS DI SD PANCA BUDI MEDAN. Proses kerja dari alat ini adalah dengan melakukan setting jam waktu sekarang dengan menggunakan RTC DS3231 kemudian memasukkan daftar jadwal kegiatan yang ada di SD Panca Budi Medan. Setelah itu mikrokontroler Arduino Uno akan membaca data daftar bel hari ini. Aplikasi ini dilengkapi dengan modul DFPlayer mini yang mampu memainkan format file suara *MP3* atau *.WAV*, sehingga informasi yang disampaikan melalui bel dapat berupa suara.

Kata Kunci : Pengingat jadwal dan Pengumuman Otomatis, Mikrokontroler Arduino Uno, RTC DS3231,DFPlayer Mini

DAFTAR ISI

	Halaman
KATA PENGANTAR	i
DAFTAR ISI	iii
DAFTAR GAMBAR.....	vi
DAFTAR TABEL.....	vii
DAFTAR LAMPIRAN.....	viii
DAFTAR ISTILAH.....	ix

BAB 1 PENDAHULUAN

1.1	Latar Belakang Masalah	1
1.2	Perumusan Masalah.....	2
1.3	Batasan Masalah	2
1.4	Tujuan Penelitian	3
1.5	Manfaat Penelitian.....	3

BAB 2 LANDASAN TEORI

2.1	Definisi Mikrokontroller.....	5
2.1.1	Bagian-bagian Mikrokontroller	6
2.2	Arduino Uno	8
2.2.1	Karakteristik Fisik.....	10
2.2.2	Software Arduino.....	10
2.2.3	Bahasa Pemograman Arduino Berbasis Bahasa C	12
2.3	<i>LCD (Liquid Crystal Display)</i>	14
2.3.1	Jalur Input LCD.....	15
2.3.2	<i>I2C (Inter Integred Circuit)</i>	16
2.4	Adaptor.....	17
2.5	Modul RTC DS3231	18
2.6	DFPlayer Mini.....	21
2.7	Speaker	23
2.7.1	Jenis-jenis Speaker	24
2.7.2	Pengertian Speaker Aktif dan Pasif	25
2.8	SD Modul.....	26
2.8.1	SD Card.....	27
2.9	Flow Chart	28

BAB 3 Metode Penelitian

3.1	Tempat dan Jadwal Penelitian	30
3.1.1	Tempat Penelitian	30
3.1.2	Jadwal Penelitian	31
3.2	Jenis dan Sumber Data	31
3.3	Metode Pengumpulan Data	32
3.4	Analisa Sistem Yang Sedang Berjalan	34
3.5	Prinsip Kerja Sistem	34
3.6	Perancang Sistem <i>Hardware</i> dan <i>Software</i>	35
3.7	Perancangan Hardware	35
3.7.1	Diagram Blok Rangkaian	35
3.7.2	Kebutuhan Perangkat Keras	36
3.7.3	<i>Flowchart</i> Rancangan Alat	37
3.7.4	Perancangan Perangkat Keras	39
3.7.5	Rangkaian LCDX12C	40
3.7.6	Rangkaian Modul RTC DS3231	41
3.7.7	Rangkaian DFPlayer Mini	42
3.8	Perancangan <i>Software</i>	43
3.8.1	Kebutuhan Perangkat Lunak (<i>Software</i>)	43
3.8.2	Rancangan Program	44
3.8.3	Proses Input Jadwal Kegiatan Sekolah	44
3.8.4	Proses Upload Jadwal kedalam Mikrokontroler Arduino	45

BAB 4 HASIL DAN PEMBAHASAN

4.1	Pengujian Rangkaian LCD dan I2C	48
4.2	Pengujian <i>Real Time Clock</i> (RTC)	50
4.3	Pengujian <i>DF Mini Player</i> (Pengujian Keseluruhan)	57
4.3.1	Pengingat Upacara	58
4.3.2	Pengingat Literasi Quran	62
4.3.3	Pengingat Literasi Buku	66
4.3.4	Pengingat Senam	70
4.3.5	Pengingat Salat Dhuha	73
4.3.6	Pengingat Salat dzuhur	77
4.3.7	Pengingat Salat Jumat	80
4.3.8	Pengingat Kegiatan Bersih-bersih	84
4.3.9	Pengingat Kegiatan Ekskul	88
4.3.10	Pengingat Kegiatan Mengaji	92

BAB 5 PENUTUP

5.1	Simpulan.....	97
5.2	Saran.....	98

DAFTAR PUSTAKA

BIOGRAFI PENULIS

SURAT PERNYATAAN

KETERANGAN PLAGIAT CHECKER DARI LPMU

SURAT BEBAS PRAKTIKUM

FORM PENGAJUAN JUDUL

FORM PERMOHONAN MEJA HIJAU EKSISTENSI BIMBINGAN

DOPING 1 DAN 2

BAB I

PENDAHULUAN

1.1 Latar Belakang

Berkembangnya ilmu pengetahuan dan teknologi saat ini semakin banyak memberikan kemudahan dalam kehidupan sehari-hari. Dimana segala hal yang banyak diterapkan ilmu pengetahuan dan teknologi dengan mesin ataupun elektronika, sehingga pekerjaan manusia dapat dikerjakan dengan mudah tanpa harus membuang tenaga dan dapat mempersingkat waktu. Berbagai alat rumah tangga hingga alat kerja menggunakan alat elektronik sehingga pekerjaan manusia jauh lebih ringan dan mudah.

Sebagai salah satu contoh pemanfaatan mikrokontroler Arduino Uno yang dimanfaatkan sebagai pengontrol pengingat jadwal dan pengumuman otomatis di sekolah. Dengan adanya sistem ini dapat memudahkan dalam hal mengingatkan dan memberikan pengumuman pada saat waktu belajar dimulai dan waktu pulang sekolah. Sedangkan bagi guru yang bertugas piket untuk memberikan pengingat jadwal pengumuman akan merasa kerepotan untuk selalu berbicara di depan Mikrofon yang dirasa kurang efisien waktu dan tenaga. Permasalahn tersebut dapat diatasi dengan pengontrol pengingat jadwal dan pengumuman otomatis di sekolah secara otomatis menggunakan Mikrokontroler Arduino Uno.

Salah satu kemudahan yang dimiliki oleh pengingat jadwal dan pengumuman otomatis ini adalah dengan menginputkan beberapa kegiatan ke Mikrokontroler Arduino Uno, maka secara otomatis pengingat jadwal dan pengumuman di SD Panca Budi Medan akan aktif secara otomatis sesuai inputan pengingat jadwal dan pengumuman yang telah diprogram, sehingga guru yang bertugas piket untuk memberikan pengingat jadwal pengumuman tidak perlu lagi berbicara di depan Mikrofon.

1.1 Perumusan Masalah

Sehubungan dengan latar belakang tersebut di atas maka timbul permasalahan sebagai berikut:

- a. Bagaimana merancang sistem pengingat jadwal dan pengumuman sekolah yang dapat aktif secara otomatis dengan menggunakan Mikrokontroler Arduino Uno?
- b. Bagaimana proses *Implementasi* pengingat jadwal dan pengumuman sekolah otomatis?

1.2 Batasan Masalah

Dalam perancangan dan pembuatan alat ini, terdapat beberapa pembatasan masalah, antara lain:

- a. Sistem ini menggunakan Mikrokontroler Arduino Uno sebagai pengelola perintah.

- b. Penelitian dilakukan hanya di SD Panca Budi Medan.
- c. Pengingat jadwal dan pengumuman sekolah otomatis hanya dapat bekerja sesuai dengan jadwal settingan kegiatan sekolah SD Panca Budi Medan.
- d. Sistem ini memakai catu daya dari PLN sehingga bila ada pemadaman listrik, maka sistem dan rangkain tidak akan berfungsi.

1.3 Tujuan Penelitian

Tujuan dari perancangan alat pada tugas akhir ini adalah:

- a. Untuk merancang dan mengaplikasikan sistem kontrol pengingat jadwal dan pengumuman sekolah menggunakan Arduino dengan alat pendukung Modul RTC DS3231.
- b. Untuk menciptakan alat pengingat jadwal dan pengumuman sekolah maka penerapan yang digunakan dengan cara observasi ke sekolah SD Panca Budi Medan.

1.4 Manfaat Penelitian

Manfaat yang diharapkan dari perancangan alat ini jika dalam uji cobanya adalah sebagai berikut:

- a. Menciptakan alat tepat guna.

- b. Membantu dalam hal mengaktifkan pengingat jadwal dan pengumuman sekolah.
- c. Dapat menciptakan inovasi baru yang bermanfaat bagi penulis dan masyarakat umum.
- d. Dapat meringankan pekerjaan guru yang bertugas piket untuk memberikan pengumuman di sekolah.

BAB 2

LANDASAN TEORI

2.1 Definisi Mikrokontroler

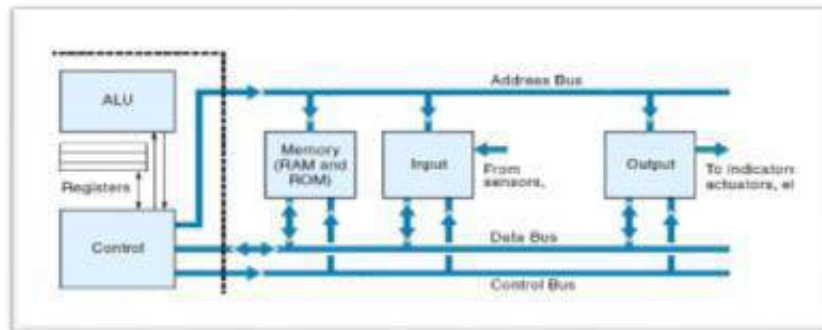
Mikrokontroler adalah sistem mikroprosesor lengkap yang terkandung di dalam sebuah chip. Mikrokontroler berbeda dari mikroprosesor serbaguna yang digunakan dalam sebuah PC karena didalam sebuah mikrokontroler umumnya juga telah berisi komponen pendukung sistem minimal mikroprosesor, yakni memori dan antar muka I/O sedangkan didalam mikroprosesor umumnya hanya berisi CPU saja **(Putra dan Harris, 2017)**.

Kelebihan utama dari mikrokontroler ialah tersedianya RAM dan peralatan I/O pendukung sehingga ukuran board mikrokontroler menjadi sangat ringkas. Mikrokontroler adalah sebuah chip yang berfungsi sebagai pengontrol rangkaian elektronik dan umumnya dapat menyimpan di MCS51 ialah mikrokomputer CMOS 8 bit dengan 4 KB Flash PEROM (*Programmable and Erasable Only Memory*) yang dapat dihapus dan ditulisi sebanyak 1000 kali. Mikrokontroler adalah sebuah chip yang berfungsi sebagai pengontrol rangkaian elektronik dan umumnya dapat menyimpan program didalamnya **(Putra dan Harris, 2017)**.

2.1.1 Bagian-bagian Mikrokontroler

a. Mikroprosesor/CPU

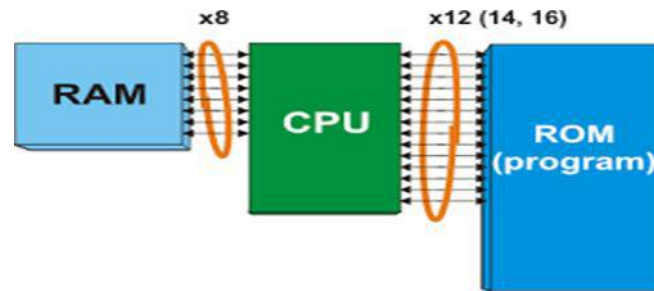
Mikroprosesor adalah sebuah chip yang memiliki fungsi untuk memproses data biner secara digital dan komponennya terdiri dari ALU (*Arithmetic Logic Unit*), instruksi decoder, register, bus control circuit, *control* dan *timing unit*.



Gambar 2.1 Mikroprosesor/CPU
(Sumber: Putra dan Harris, 2017)
di akses pada tanggal 15 november 2019

b. Bus

Bus adalah jalur jalur fisik yang menghubungkan CPU dengan memori dan unit lain dalam mikrokontroler.

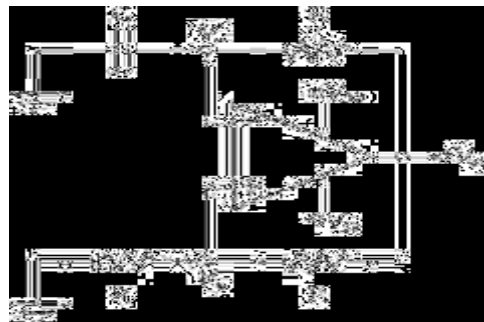


Gambar2.2 Bus

(Sumber: Putra dan Harris, 2017)
di akses pada tanggal 15 november 2019

a. Osilator

Osilator adalah suatu rangkaian yang menghasilkan keluaran yang amplitudonya berubah-ubah secara periodic dengan waktu.

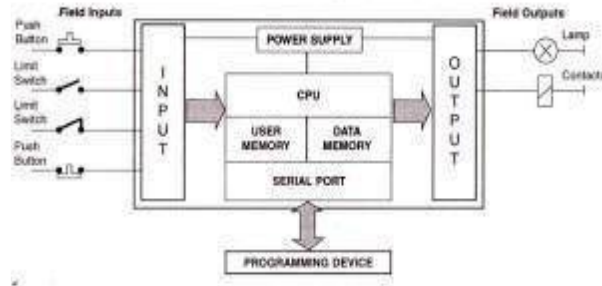


Gambar 2.3 Osilator

(Sumber: Putra dan Harris, 2017)
di akses pada tanggal 16 november 2019

b. Input/Output (I/O)

Setiap sistem komputer memerlukan sistem input dan output yang merupakan media keluar masuka data dari dan ke komputer. Contoh peralatan I/O yang umum yang terhubung dengan sebuah komputer seperti *keyboard, mouse, monitor, sensor, printer, LED*, dan lain-lain.

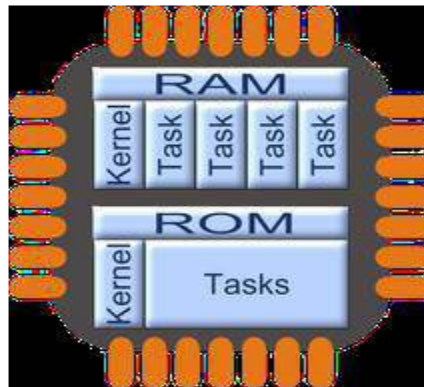


Gambar 2.4 Input/Output

Sumber: (Putra dan Harris,2017) di akses pada tanggal 16 november 2019

c. Memori

Memori adalah bagian mikrokontroller yang berfungsi untuk menyimpan data, terdiri dari RAM dan ROM.



Gambar 2.5 Memori

(Sumber: Putra dan Harris, 2017) di akses pada tanggal 15 november 2019

2.2 Arduino Uno

Arduino adalah sebuah board mikrokontroller yang berbasis ATmega328, Arduino memiliki 14 pin input/output yang mana 6 pin dapat digunakan sebagai output PWM, 6 analog input, crystal osilator 16 MHz, koneksi USB, jack power, kepala ICSP, dan tombol reset. Arduino mampu mensupport mikrokontroller, dapat dikoneksikan dengan komputer menggunakan kabel USB.

Arduino memiliki kelebihan tersendiri dibandingi board mikrokontroller yang lain bersifat open source, arduino juga mempunyai bahasa perogramnya sendiri yang berupa bahasa C. Selain itu dalam board arduino sendiri sudah terdapat loader yang berupa USB sehingga memudahkan kita ketika kita memprogram mikrokontroller didalam arduino, port USB tersebut selain untuk loader ketika memprogram, bisa juga difungsikan sebagai port komunikasi serial.

(Batu Bara, 2017)

Tabel 2.1 Tabel Deskripsi Arduino

Mikrokontroller	ATmega 328
Tegangan pengoprasian	5V
Tegangan Input yang disarankan	7 – 12 V
Batas Tegangan Input	6 – 20 V
Jumlah pin I/O Digital	14 Pin digital (6 diantaranya menyediakan keluaran PWM)
Jumlah pin input Analog	6 pin
Arus Dc tiap pin I/O	40mA
Arus Dc untuk pin 3,3 V	50mA
<i>Memoir Flash</i>	32 KB (ATmega 328) sekita0,5 KD digunakan oleh bootloader
SRAM	2 KB (ATmega 328)
EPROM	1 KB (ATmega 328)
<i>Clock Speed</i>	16 MHz

Sumber : Batu Bara,2017

2.2.1 Karakteristik Fisik

Panjang dan lebar maksimum dari PCB Arduino UNO masing-masingnya adalah 2.7 dan 2.1 inci, dengan konektor USB dan power jack yang memperluas dimensinya empat lubang sekrup memungkinkan board untuk dipasangkan ke sebuah permukaan atau kotak. Sebagai catatan, bahwa jarak antara pin digital 7 dan 8 adalah 160 mil.(0.16”), bukan sebuah kelipatan genap dari jarak 100 mil dari pin lainnya (**Batu Bara, 2017**).



Gambar 2.6 Arduino Uno
(Sumber :Batu Bara, 2017)
Di akses pada tanggal 15 november 2019

2.2.2 Software Arduino

Arduino Uno dapat diprogram dengan perangkat lunak Arduino. Pada ATmega328 di Arduino terdapat bootloader yang memungkinkan anda untuk meng-upload kode baru untuk itu tanpa menggunakan programmer hardware eksternal (Batu Bara, 2017).

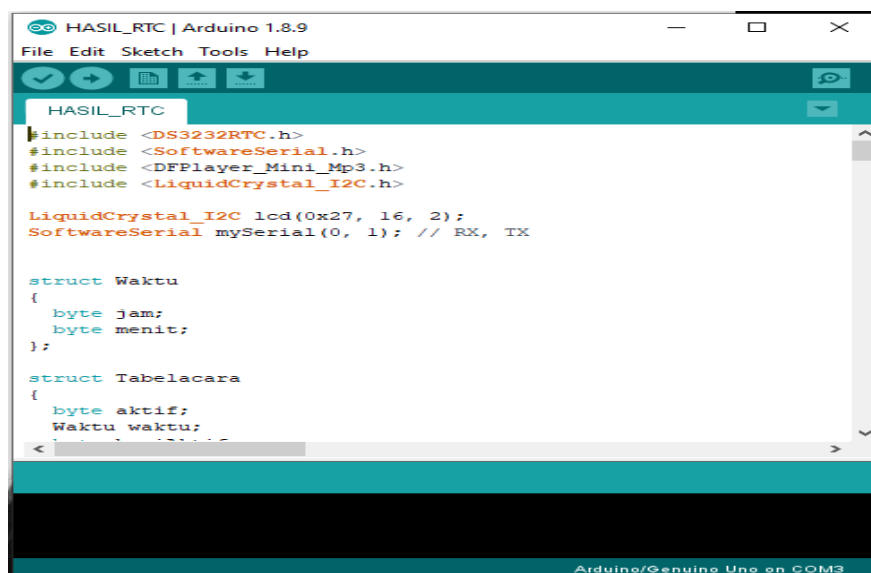
IDE (*Integrated Development Environment*) Arduino adalah software yang sangat canggih ditulis dengan menggunakan Java. IDE Arduino terdiri dari:

1. Editor program, sebuah window yang memungkinkan pengguna menulis dan mengedit program dalam bahasa Processing.
2. Compiler, sebuah modul yang mengubah kode program (bahasa Processing) menjadi kode biner. Bagaimanapun sebuah mikrokontroler tidak akan bisa memahami bahasa Processing. Yang bisa dipahami oleh mikrokontroler adalah kode biner, itulah sebabnya compiler diperlukan dalam hal ini.
3. Uploader, sebuah modul yang memuat kode biner dari komputer ke dalam memory di dalam papan Arduino.

Sebuah kode program Arduino umumnya disebut dengan istilah *sketch*.

Kata “*sketch*” digunakan secara bergantian dengan “kode program” dimana menggunakan bahasa pemrograman C keduanya memiliki arti yang sama.

(Batu Bara, 2017).



```

HASIL_RTC | Arduino 1.8.9
File Edit Sketch Tools Help
HASIL_RTC
#include <DS3232RTC.h>
#include <SoftwareSerial.h>
#include <DFPlayer_Mini_Mp3.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 16, 2);
SoftwareSerial mySerial(0, 1); // RX, TX

struct Waktu
{
  byte jam;
  byte menit;
};

struct Tabelacara
{
  byte aktif;
  Waktu waktu;
}

```

Gambar 2.7 Tampilan IDE Arduino dengan sebuah *sketch*

Sumber. Penulis, 2019 diakses pada tanggal 20 desember 2019

2.2.3 Bahasa Pemrograman Arduino Berbasis Bahasa C

Seperti yang telah dijelaskan diatas program Arduino sendiri menggunakan bahasa C, walaupun banyak sekali terdapat bahasa pemrograman tingkat tinggi (*high level language*) seperti *pascal*, *basic*, *cobol*, dan lainnya. Walaupun demikian, sebagian besar dari programmer professional masih tetap memilih bahasa C sebagai bahasa yang lebih unggul, berikut alasan-alasannya:

- a. Bahasa C merupakan bahasa yang *powerful* dan *fleksibel* yang telah terbukti dapat menyelesaikan program-program besar seperti pembuatan sistem operasi, pengolah, gambar (seperti pembuatan game) dan juga pembuatan kompilator bahasa pemrogram baru.
- b. Bahasa C merupakan bahasa yang sangat populer dan banyak digunakan oleh programmer sehingga kemungkinan besar *library* pemrogram telah banyak disediakan oleh pihak luar dan dapat diperoleh dengan pihak mudah.
- c. Bahasa C merupakan bahasa yang bersifat modular, yaitu terusun atas rutin-rutin tertentu yang dinamakan dengan fungsi (*function*) dan fungsi-fungsi tersebut dapat digunakan kembali untuk pembuatan program-program lainnya tanpa harus menulis ulang implementasinya.
- d. Bahasa C merupakan bahasa tingkat menengah (*middle level language*) sehingga mudah untuk melakukan interface (pembuatan programantar muka) ke perangkat keras.

- e. Struktur penulisan program dalam bahasa C harus memiliki fungsi utama, yang bernama `main()`. Fungsi inilah yang akan dipanggil pertama kali pada saat proses eksekusi program. Artinya apabila kita mempunyai fungsi lain selain fungsi utama, maka fungsi lain tersebut baru akan dipanggil pada saat digunakan (Daulay,2018).

Selain itu juga dalam bahasa C kita akan mengenal *file header*, biasa ditulis dengan ekstensi `h(*.h)`, adalah file bantuan yang digunakan untuk menyimpan daftar-daftar fungsi yang akan digunakan dalam program. Bagi anda yang sebelumnya pernah mempelajari bahasa pascal, *file header* ini serupa dengan unit. Dalam bahasa C, file header standar yang untuk proses *input/output* adalah `<stdio.h>`.

Perlu sekali untuk diperhatikan bahwa apabila kita menggunakan *file header* yang telah disediakan oleh kompilator, maka kita harus menuliskannya didalam tanda ‘<’ dan ‘>’ (mialnya `<stdio.h>`). Namun apabila menggunakan *file header* yang kita buat sendiri, maka file tersebut ditulis diantara tanda “ dan ” (misalnya “*cobaheader.h*”). Perbedaan antara keduanya terletak pada saat pencarian file tersebut. Apabila kita menggunakan `<>`, maka file tersebut dianggap berada pada direktori *default* yang telah ditentukan oleh kompilator. Sedangkan apabila kita menggunakan tanda “ ”, maka *file header* dapat kita dapat tentukan sendiri lokasinya.

File header yang akan kita gunakan harus kita daftarkan dengan menggunakan directive `#include`. Directive `#include` ini berfungsi untuk memberi tahu kepada

kompilator bahwa program yang kita buat akan menggunakan file-file yang didaftarkan. Berikut ini contoh penggunaan directive *#include*.

```
#include<stdio.h> #include<stdlib.h> #include "myheader.h"
```

Setiap kita akan menggunakan fungsi tertentu yang disimpan dalam sebuah *file header*, maka kita juga harus mendaftarkan *file header* nya dengan menggunakan directive *#include*. Sebagai contoh, kita akan menggunakan fungsi *getch()* dalam program, maka kita harus mendaftarkan *file header* <conio.h> (Daulay,2018).

2.3 LCD (Liquid Crystal Display)

LCD (*Liquid Crystal Display*) adalah suatu jenis media tampil yang menggunakan Kristal cair sebagai penampil utama. LCD sudah digunakan diberbagai bidang misalnya alat-alat elektronik seperti televisi, kalkulator, ataupun layar komputer. Pada bab ini aplikasi LCD yang digunakan ialah LCD dot matrik dengan dengan jumlah 2 x 16. LCD sangat berfungsi sebagai penampil yang nantinya akan digunakan untuk menampilkan status kerja alat (Batu Bara, 2017). Adapun fitur yang disajikan dalam LCD ini adalah :

- Terdiri dari 16 karakter dan 2 baris.
- Mempunyai 192 karakter tersimpan.
- Terdapat karakter generator terprogram.
- Dapat dialamati dengan mode4-bit dan 8-bit.
- Dilengkapi dengan back light.



Gambar 2.8 LCD (*Liquid Crystal Display*)
 (Sumber : Batu Bara, 2017) diakses pada tanggal 15 november 2019

s

Tabel 2.2 Deskripsi Pin LCD 2X16

PIN	Deskripsi
1	Gnd
2	VCC
3	Pengantur Kontras
4	RS
5	RW
6	Enable
7-14	Data I/O Pins
15	VCC
16	GBD

Sumber : Batu Bara ,2017

2.3.1 Jalur Input LCD

Pin, kaki atau jalur input dari control dalam suatu LCD (*Liquid Crystal Display*) diantaranya adalah :

- a. Pin data adalah jalur memberikan data karakter yang ingin ditampilkan menggunakan LCD (*Liquid Crystal Display*) dapat dihubungkan dengan bus data dari rangkaian lain seperti mikrokontroler dengan lebar data 8 bit.
- b. Pin RS (*Register Select*) berfungsi sebagai indikator atau yang menentukan jenis data yang masuk, apakah data atau perintah. Logika low menunjukkan yang masuk adalah perintah, sedangkan logika high menunjukkan data.
- c. Pin R/W (*Read Write*) berfungsi sebagai intruksi pada modul jika low tulis data, sedangkan high baca data.
- d. Pin E (*Enable*) digunakan untuk memegang data baik masuk atau keluar.

Pin VLCD berfungsi mengatur kecerahan tampilan (kontras) dimana pin ini dihubungkan dengan trimpot 5 Kohm, jika tidak digunakan dihubungkan ke ground, sedangkan tegangan catu daya ke LCD sebesar 5 volt (**Daulay,2018**).

2.3.2 I2C (*Inter Integrated Circuit*)

Inter Integrated Circuit atau sering disebut I2C adalah standar komunikasi serial dua arah menggunakan dua saluran yang di desain khusus untuk mengirim maupun menerima data. I2C merupakan bus standar yang didesain oleh Philips pada awal tahun 1980-an untuk memudahkan komunikasi antar komponen pada suatu rangkaian. I2C merupakan singkatan dari Inter IC atau komunikasi antar IC, sering disebut juga IIC atau I2C. Pada awalnya kecepatan komunikasi maksimumnya atur pada 100kbps karena pada awalnya kecepatan tinggi belum dibutuhkan pada transmisi data. Untuk yang membutuhkan kecepatan tinggi, ada

mode 400kbps dan sejak 1998 ada mode kecepatan tinggi 3,4 Mbps. I2C tidak hanya digunakan pada komponen yang terletak pada satu *board*, tetapi juga digunakan untuk menghubungkan komponen yang terhubung melalui kabel (Endaryono, Harianto, dan Wibowo, 2014).



Gambar 2.9 12C I2C (*Inter Integrated Circuit*)
Sumber. Saptaji.com diakses pada tanggal 15 desember 2019

2.4 Adaptor

Adaptor adalah sebuah perangkat berupa rangkaian elektronika untuk mengubah tegangan listrik yang besar menjadi tegangan listrik yang besar menjadi tegangan listrik lebih kecil, atau rangkaian untuk mengubah arus bolak-balik (arus AC) menjadi arus searah (arus DC). Adaptor / *power supplay* merupakan tegangan AC 22 Volt menjadi kecil antara 3 volt sampai 12 volt sesuai kebutuhan alat elektronika. Terdapat 2 jenis adaptor berdasarkan sistem kerjanya, adaptor sistem trafo *step down* dan adaptor sistem *switching*.

Dalam prinsip kerjanya kedua sistem adaptor tersebut berbeda, adaptor *step-down* menggunakan teknin industry medan magnet, komponen utamanya adalah

kawat email yang di lilit pada teras besi, terdapat dua lilitan yaitu lilitan primer dan lilitan skunder, ketika listrik masuk kelilitan primer maka akan terjadi induksi pada kawat email sehingga akan terjadi gaya medan magnet pada teras besi kemudian akan menginduksi lilitan skunder.

Sedangkan sistem *switching* menggunakan teknik transistor maupun IC *switching*, adaptor ini lebih baik dari pada adaptor teknik induksi, tegangan yang di keluarkan lebih stabil dan komponennya suhunya tidak terlalu panas sehingga mengurangi tingkat resiko kerusakan suhu berlebih, biasanya regulator ini digunakan pada peralatan elektronik digital (**Damayanti, 2017**)



Gambar 2.10 Adaptor

(Sumber : Damayanti, 2017) diakses pada tanggal 15 desember 2019

2.5 Modul RTC DS3231

Real Time Clock atau sering disebut juga RTC merupakan salah satu komponen elektronika aktif yang dapat menyimpan data tanggal dan waktu di

dalamnya. Data waktu ini sering kali digunakan untuk membuat sebuah alat penjadwalan terpadu atau hanya sekedar jam digital (**Daulay,2018**). Pada DS3231 Operasi jam bisa diformat dalam 24 jam atau 12 jam (AM/PM). Untuk tatp muka dengan suatu mikroprosesor dapat disederhanakan dengan menggunakan sinkronisasi komunikasi serial I2C dengan kecepatan clock 400Khz. Hanya membutuhkan 2 saluran untuk komunikasi dengan clock/RAM: SCL (serial clock), SDA (Serial I/O data), dan juga dilengkapi dengan keluaran *SQW/OUT* yang dapat di program untuk mengetahui perubahan data waktu pada RTC dan pin RST. DS3231 didesain untuk mengoperasi pada power yang sangat rendah dan mempertahankan data dan informasi ± 1 microwatt (**Assegaf,2017**)



Gambar 2.11 *Module RTC DS 3231*

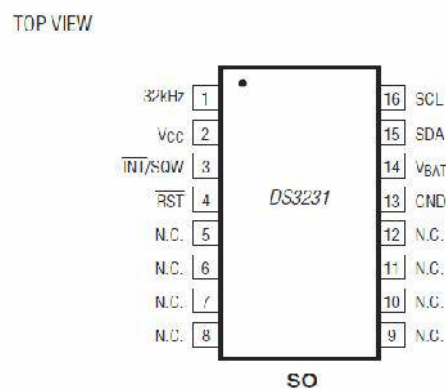
Sumber. <https://www.ebay.com/itm/Module-Rtc-DS3231-6-Pin-Welded-Battery-Arduino-/202811860417> diakses pada tanggal 12 desember 2019

Adapun karakteristik dari RTC tipe 3231 yaitu:

- a. RTC menghitung detik, menit, jam, tanggal, bulan, hari, setiap minggu dan tahun dengann benar sampai tahun 2100

- b. Serial I2C untuk pin minimum proses komunikasi RTC
- c. 2.0 – 5.5 Volt full operation
- d. Mempunyai kemasan 16 pin SOICs
- e. 3 simple wire interface (I2C dan SQW/Out)
- f. Square wave output yang dapat deprogram
- g. Mempunyai sensor temperature dengan akurasi $\pm 3^\circ$ Celcius.

Adapun konfigurasi pin RTC DS3231 ditunjukkan



Gambar 2.12 Bagian-bagian RTC DS 3231

Sumber. <http://www.rinkydinkelectronics.com/library.php?id=74>
diakses pada tanggal 20 desember 2019

Adapun penjelasan dari pin DS3231 ditunjukkan pada tabel 2.3:

Tabel 2.3 Penjelasan pin-pin pada RTC

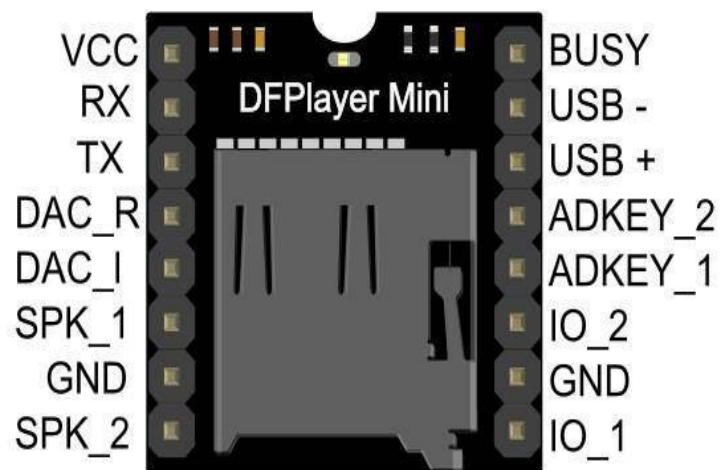
Pin	Fungsi
VCC1, VBAT	sebagai power supply. Jika $V_{CC2} > V_{CC1} (+0.2V)$ maka V_{CC2} menjadi power DS3231, begitu juga sebaliknya
SCL	untuk sinkronisasi data pada serial interface (clock)
SDA	pin data bidireksional (input/output)

INT/SQW _{OUT}	Output interupsi dari RTC yang dapat deprogram sebagai pemberi informasi perubahan waktu
32Khz	Output gelombang kotak yang dapat diprogram
RST	Pin Resetyang RTC
Clock/ kalender	memuat data dalam bentuk BCD dan memiliki 7 register write/ read.
AM-PM/12-24	7 bit register ditetapkan sebagai mode 12 atau 24 jam.
Write protect bit	pada 7 bit pertama (bit 0.....6) berlogika 0 sampai pada proses <i>read</i> , bit 7 harus berlogika 0 sebelum ada operasi penulisan untuk clock atau RAM
Clock/ calendar Burst Mode	bagian dari operasi burst mode, secara teratur dapat dibaca atau ditulis mulai dengan bit 0 pada alamat 0

Sumber: Assegaf 2017

2.6 DFPlayer Mini

Modul DFPlayer Mini adalah sebuah modul MP3 serial yang menyiapkan kesempurnaan integrasi MP3, WMV hardware decoding. Sedangkan softwarena mendukung sistem file FAT16, FAT32. Melalui perintah-perintah serial sederhana untuk menentukan memutar music, serta bagaimana cara memutar musik dan fungsi lainnya, tidak melalui operasi yang rumit, mudah digunakan, stabil dan dapat diandalkan adalah fitur-fitur yang paling penting dari modul ini (**Wijayanto, Hadiyoso, dan Hariyani 2015**).



Gambar 2.13 DFPlayer Mini

(Sumber: (Wijayanto, Hadiyoso, dan Hariyani 2015) diakses pada tanggal 23 Januari 2020

Tabel 2.4 Keterangan Port DFPlayer mini

Nomor	Nama	Deskripsi	Keterangan
1	VCC	Input Voltage	DC 3.2-5.0V; Typical: DC4.2
2	RX	UART serial input	
3	TX	UART serial output	
4	DAC_R	Audio output right channel	Drive earphone dan amplifier
5	DAC_L	Audio output left channel	Drive earphone dan amplifier
6	SPK2	Speaker	Drive speaker less than 3W
7	GND	Ground	Power Ground
8	SPK1	Speaker	Drive speaker less than 3W
9	IO1	Trigger port1	Short pree to play previous (long press to decrease volume)
10	GND	Ground	Power Ground
11	IO2	Trigger port1	Short pree to play next (long press to decrease volume)
12	ADKEY1	AD port 1	Trigger play first segment
13	ADKEY2	AD port 2	Trigger play fifth segment

14	USB+	USB+DP	USB PORT
15	USB-	USB-DM	USB PORT
16	Busy	Playing Status	Low means playing/High means no

Sumber <https://www.nyebarilmu.com/tutorial-mengakses-module-mp3-dfplayer-mini/>

2.7 Speaker

Pengeras suara (bahasa inggris: *loudspeaker* atau *speaker*) adalah transduser yang mengubah sinyal elektrik ke frekuensi audio (suara) dengan cara menggetarkan komponennya yang berbentuk membran untuk menggetarkan udara sehingga terjadilah gelombang suara sampai di gendang telinga kita dan dapat kita dengar sebagai suara (**Daulay,2018**).

Dalam setiap sistem penghasil suara (*loud speaker*), pengeras suara merupakan juga menentukan kualitas suara di samping juga peralatan pengolah suara sebelumnya yang masih berbentuk listrik dalam rangkaian penguat amplifier. Sistem pada pengeras suara adlah suatu komponen ynag mengubah kode sinyal elektronik terkahir menjadi gerakan mekanik. Dalam penyimpan suara pada kepingan CD, pita magnetic tape, dan kepingan DVD, dapat direproduksi oleh pengeras suara *loud speaker* yang dapat kita dengar. Pengeras suara adalah sebuah teknologi yang memberikan dampak yang sangat besar terhadap budaya kita.



Gambar 2.14 Speaker

Sumber : <https://www.amazon.in/Logitech-980-000802-Multimedia-Speakers-Z150/dp/B00EZ9XLEY> diakses pada tanggal 15 desember 2019

Frekuensi Audio adalah suatu getaran frekuensi yang dapat didengar oleh manusia dengan standar getaran antar 20 – 20.000 Hertz. Sesuai dengan kemampuan mendengarnya sistem pendengaran manusia dibagi menjadi tiga macam, yakni infrasonik, audible, dan ultrasonik. Frekuensi infrasonik adalah frekuensi dengan rentang gelombang antara 0 – 2 Hz. Frekuensi audible adalah frekuensi dengan rentang gelombang antara 0 – 20.000 Hz. Sedangkan Frekuensi ultrasonik adalah frekuensi dengan rentang gelombang di atas 20.000 Hz (**Daulay,2018**).

2.7.1 Jenis-jenis Speaker

Berdasarkan Frekuensi yang dihasilkan, *Speaker* dapat dibagi menjadi:

- a. *Speaker Tweeter* yaitu *speaker* yang menghasilkan Frekuensi Tinggi (sekitar 2kHz – 20kHz).
- b. *Speaker Mid-range*, yaitu *speaker* yang menghasilkan Frekuensi menengah (sekitar 300Hz – 5kHz).

- d. *Speaker Woofer*, yaitu *speaker* yang menghasilkan Frekuensi rendah (sekitar 40Hz – 1kHz).
- e. *Speaker Sub-woofer*, yaitu *speaker* yang menghasilkan Frekuensi sangat rendah yaitu sekitar 20Hz – 200Hz.
- f. *Speaker Full Range*, yaitu *speaker* yang dapat menghasilkan Frekuensi rendah hingga Frekuensi tinggi.

Berdasarkan Fungsi dan bentuknya, *speaker* juga dapat dibedakan menjadi:

- a. *Speaker Corong*.
- b. *Speaker Hi-fi*
- c. *Speaker Headphone*
- d. *Headphone*
- e. *Earphone*
- f. *Speaker Televisi*
- g. *Speaker Sound System (home theater)*
- h. *Speaker Laptop*

2.7.2 Pengertian Speaker Aktif dan Speaker Pasif

Speaker yang digunakan untuk dalam kehidupan sehari-hari pada umumnya dapat dibedakan menjadi 2 kategori, yaitu Speaker Aktif dan Speaker Pasif. Berikut ini adalah penjelasan singkat mengenai kedua jenis speaker ini.

1. Speaker Aktif

Speaker Aktif adalah *Speaker* yang memiliki *Amplifier* (penguat suara) di dalamnya. *Speaker* Aktif memerlukan kabel listrik tambahan untuk menghidupkan *Amplifier* yang terdapat di dalamnya.

2. Speaker Pasif

Speaker pasif adalah *speaker* yang di dalamnya tidak mempunyai sebuah *Amplifier* atau penguat suara, untuk menggunakan *speaker* pasif membutuhkan tambahan *Amplifier* dan untuk menggerakkan *speaker* pasif harus menguatkan sinyal terlebih dahulu.

2.8 SD Modul

SD Card Shield atau *SD Modul* merupakan solusi untuk mengirim data ke *SD card*.



Gambar 2.15 SD Modul

Sumber. Hartono, 2013 diakses pada tanggal 20 desember 2019

Pin Out dari *SD Card Shield* dapat dihubungkan ke Arduino maupun mikrokontroler lainnya, sehingga bermanfaat untuk menambah kapasitas tempat penyimpanan data dan pencatatan data. *SD Card Shield* ini dapat langsung dipasang pada Arduino dan terdapat switch untuk memilih flash card slot.

(Hartono, 2013).

2.8.1 SD Card

SD Card adalah kartu memori non-volatile yang dikembangkan oleh *SD Card Association* yang digunakan dalam perangkat portable. Saat ini, teknologi microSD sudah digunakan lebih dari 400 merek produk serta dianggap sebagai standar.



Gambar 2.16 SD Card

Sumber. Rudi Hartono, 2013 diakses pada tanggal 20 desember 2019

Keluarga SD Card yang lain terbagi menjadi SDSC yang kapasitas maksimum resminya sekitar 2G, meskipun beberapa ada yang sampai 4GB. SDHC (*High Capacity*) memiliki kapasitas dari 4GB sampai 32GB dan SDXC (*Extended Capacity*) kapasitasnya diatas 32 GB hingga maksimum 2TB. Keberagaman kapasitas seringkali membuat kebingungan karena masing-masing protokol komunikasi sedikit berbeda. Dari sudut pandang perangkat, semua kartu ini


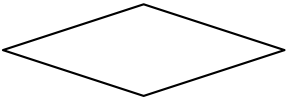
termasuk kedalam keluarga SD. SD adapter memungkinkan konversi fisik kartu SD yang lebih kecil untuk bekerja di slot fisik yang lebih besar dan pada dasarnya ini adalah alat pasif yang menghubungkan pin dari SD Card yang kecil ke pin adaptor SD card yang lebih besar (**Hartono, 2013**).

2.9 Flow Chart

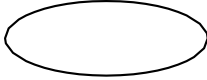
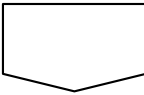





Flowchart adalah representasi secara simbolik dari suatu algoritma atau prosedur untuk menyelesaikan suatu masalah, dengan menggunakan *flowchart* akan memudahkan pengguna melakukan pengecekan bagian-bagian yang terlupakan dalam analisis masalah, disamping itu *flowchart* juga berguna sebagai fasilitas untuk berkomunikasi antara pemrogram yang bekerja dalam tim suatu proyek (**Santoso, Nurmalina, 2017**).

Flowchart membantu memahami urutan-urutan logika yang rumit dan panjang. *Flowchart* membantu mengkomunikasikan jalannya program ke orang lain (bukan pemrogram) akan lebih mudah.

Tabel 2.5 Simbol-simbol *flowchart*

No.	Simbol	Keterangan
1.		Permulaan sub program
2.		Perbandingan, pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya

Tabel 2.5 Simbol-simbol *flowchart* (lanjutan)

No.	Simbol	Keterangan
3.		<i>Penghubung bagian-bagian flowchart yang berada pada satu halaman</i>
4.		<i>Penghubung bagian-bagian flowchart yang berada pada halaman berbeda</i>
5.		<i>Permulaan/akhir program</i>
6.		<i>Arah aliran program</i>
7.		<i>Proses inisialisasi/pemberian harga awal</i>
8.		<i>Proses penghitung/proses pengolahan data</i>
9.		<i>Proses input/output data</i>

Sumber (Santoso, Nurmalina, 2017).

BAB 3

METODE PENELITIAN

3.1 Tempat dan Jadwal Penelitian

3.1.1 Tempat Penelitian

Penelitian ini dilakukan di Sekolah Dasar (SD) Panca Budi Medan yang berlokasi di Jalan Jend. Gatot Subroto km 4,5 Kel.Simpang Tanjung Kec. Medan Sunggal (2,58km) Kota Medan.



Gambar 3.1 Map SD Panca Budi Medan
Sumber: Google 2020

Kegiatan pembelajaran dilaksanakan pagi hingga siang hari dimulai dari jam 7.30 sampai dengan jam 13.30 dan di lanjutkan dengan kegiatan ekstrakurikuler dan kelas mengaji sampai jam 16.00. Setiap harinya dalam kegiatan pembelajaran dilaksanakan secara teori dan praktek, hingga saat ini jumlah tenaga pendidik/guru

/guru di SD Panca Budi Medan berjumlah 36 orang dan jumlah tenaga kependidikan/pegawai 6 orang dan untuk jumlah siswa/siswi terdiri dari 580 orang. Penelitian ini dilakukan guna memperoleh informasi kegiatan-kegiatan apa saja yang dilakukan di SD Panca Budi Medan.

3.1.2 Jadwal Penelitian

Jadwal penelitian pada skripsi ini adalah sebagai berikut :

NO	Kegiatan	2019			2020			
		Okt	Nov	Des	Jan	Feb	Mar	Apr
1	Pengajuan Judul	■						
2	Seminar Proposal		■					
3	Pengumpulan Data		■	■				
4	Melakukan Penelitian	■						
5	Menyiapkan Peralatan			■				
6	Perancangan Alat			■	■			
7	Pembuatan Sistem/Program			■	■	■		
8	Uji Coba Alat dan Program				■	■		
9	Implementasi Alat dan Program						■	
10	Dokumentasi						■	
11	Seminar Hasil							
12	Sidang Skripsi							

Gambar 3.2 Jadwal Penelitian Kualitatif

Sumber: Penulis

3.2 Jenis dan Sumber Data

Dalam penyusunan ini, penulis memperoleh data berupa data primer, yaitu data yang secara langsung diperoleh dari sekolah dalam bentuk wawancara dan data tersebut penulis olah sebagai informasi dari penelitian.

Sumber data berasal dari internal dari objek penelitian yaitu data yang diambil dari dalam pihak sekolah. Salah satunya adalah dengan melakukan wawancara kepada

beberapa pihak dari Sekolah SD Panca Budi Medan tentang penelitian yang dilaksanakan.

3.3 Metode Pengumpulan Data

Adapun metode yang digunakan dalam penelitian pada perusahaan adalah sebagai berikut :

a. Observasi

Studi lapangan (observasi) merupakan teknik pengumpulan data dengan langsung terjun ke lapangan untuk mengamati permasalahan yang terjadi secara langsung di tempat kejadian secara sistematis kejadian-kejadian, perilaku, objek-objek yang dilihat dan hal-hal yang diperlukan dalam mendukung penelitian yang sedang berlangsung. Dalam penulisan ini, penulis melakukan pengamatan langsung atau peninjauan terhadap kegiatan yang menjadi bahan sampel dalam penelitian. Adapun jadwal siswa/siswi di SD Panca Budi Medan sudah penulis observasi ke sekolah tersebut, dimana jadwal siswa/siswi di SD Panca Budi Medan masuk setiap minggu nya dan memiliki kegiatan dari hari Senin-Sabtu yang akan menjadi acuan pengumuman berbunyi setiap harinya.

Berikut beberapa kegiatan yang dilakukan siswa-siswi SD Panca Budi setiap minggu nya yang didapat dari observasi ke tata usaha SD Panca Budi Medan terdapat 10 kegiatan yaitu :

1. Upacara

2. Literasi Quran
3. Literasi Buku
4. Senam
5. Shalat Dhuha
6. Shalat Dzuhur
7. Shalat Jumat
8. Kegiatan bersih-bersih
9. Kegiatan Ekskul
10. Kegiatan Mengaji

- **Wawancara**

Wawancara merupakan teknik pengumpulan data yang dilakukan melalui tatap muka dan tanya jawab langsung antara penulis dengan beberapa orang dari pihak sekolah. Dalam hal wawancara tersebut, penulis mencoba memperoleh keterangan atau informasi untuk mengetahui sistem yang berjalan.

- **Studi Literatur**

Studi Literatur adalah salah satu metode pengumpulan data dengan cara membaca buku-buku dan jurnal sesuai dengan data yang dibutuhkan. Pada penelitian ini penulis memilih studi literature untuk mengumpulakn

referensi dari buku-buku mengenai mikrokontroler serta jurnal yang membahas tentang mikrokontroler arduino.

3.4 Analisa Sistem Yang Sedang Berjalan

Analisis sistem ini merupakan gambaran tentang sistem yang saat ini sedang berjalan. Sistem yang digunakan masih menggunakan cara manual, yaitu memberikan pengumuman jadwal dengan memakai tenaga manusia. Analisis ini bertujuan untuk membuat sistem yang baru agar terkomputerisasi sehingga dapat lebih efektif dan efisien.

3.5 Prinsip Kerja Sistem

Pada sistem ini jadwal kegiatan di inputkan melalui program, arduino membaca data jadwal kegiatan kemudian membandingkan dengan waktu jam yang telah di setting dari RTS DS3231. Apabila data jadwal sama dengan waktu sekarang, maka arduino akan menjalankan perintah untuk membunyikan suara Mp3 sesuai yang diprogramkan sehingga modul Mp3 berfungsi sebagai *output* suara diteruskan ke *Loudspeaker*, dan LCD menampilkan informasi hari, jam, menit dan nama kegiatan sesuai dengan waktu bel berdering. Selanjutnya untuk memasukkan input-an jadwal kegiatan sekolah yang baru, dilakukan dengan menuliskan algoritma jadwal kegiatan sekolah yang berupa *coding* ke dalam software Arduino begitu juga seterusnya setiap hari, jam, menit dan jadwal kegiatan yang telah di

setting dalam suatu program akan terus berperan sebagai kunci dari sistem pengingat jadwal dan pengumuman otomatis berbasis Arduino bekerja.

3.6 Perancangan Sistem *Hardware* dan *Software*

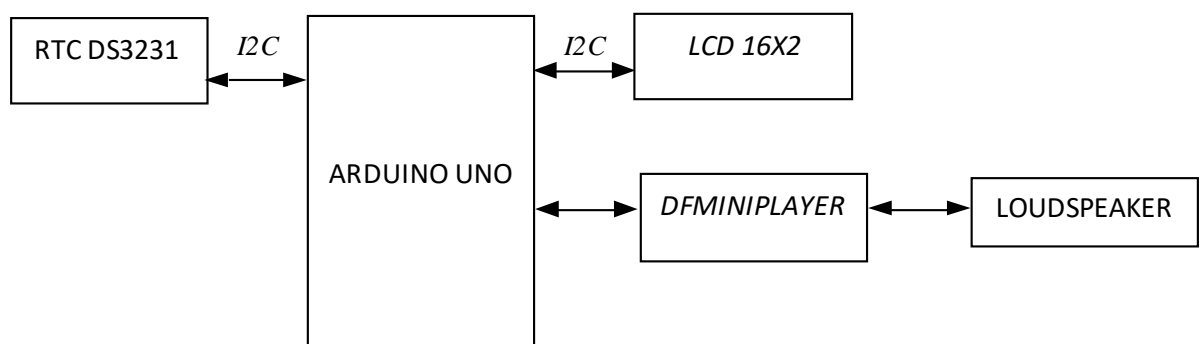
Rancang Bangun Sistem Pengingat Jadwal dan Pengumuman Otomatis di SD Panca Budi Medan terbagi atas dua bagian, yaitu perancangan *hardware* dan perancangan *software*. Perancangan *hardware* terbagi atas perancangan sistem kontrol, perancangan unit masukan dan perancangan unit keluaran. Sedangkan perancangan *software* terdiri dari perancangan pemrograman bahasa C.

3.7 Perancangan *Hardware*

Perancangan *Hardware* untuk alat perancang jadwal bunyi ini dapat diawali dengan membuat diagram blok. Dimana tiap-tiap blok saling berhubungan antara yang satu dengan yang lainnya. Diagram blok memiliki beberapa fungsi yakni: Menjelaskan cara suatu rancangan secara sederhana, menganalisa cara kerja rangkaian, mempermudah memeriksa kesalahan suatu rancangan yang dibangun.

3.7.1 Diagram Blok Rangkaian

Berikut Diagram blok rangkaian sistem yang dirancang adalah seperti berikut:



Gambar 3.3 Diagram Blok Rangkaian

Sumber. Penulis 2019

Pada sistem penanda kegiatan yang akan dibuat mempunyai input berupa keypad untuk mengatur ulang jadwal jika ada perubahan, dan input nilai waktu atau tanggal dari RTC dengan output berupa LCD dan suara dari pemutar audio yang dikendalikan oleh Arduino Uno. Pada gambar3.1 dapat dilihat input dari sistem yaitu berupa keypad dan RTC. Kemudian input ini akan diproses oleh mikrokontroler yaitu papan Arduino yang selanjutnya mengendalikan output berupa tampilan LCD dan suara dari pemutar audio. Proses alat bekerja yaitu saat pertama dihidupkan *DFMiniPlayer* berfungsi membaca settingan program yang sesuai untuk memutar suara yang di inginkan. *Speaker* berfungsi sebagai pengeras yang tersambung ke modul Mp3.

3.7.2 Kebutuhan Perangkat Keras (*Hardware*)

Untuk merancang alat pengaturan jadwal bunyi dan pengumuman otomatis dibutuhkan perangkat keras (*hardware*) Yang berfungsi sebagai media untuk mengolah dan memproses program agar dapat bekerja.

Perangkat keras yang digunakan untuk mengembangkan dan menguji coba terbagi menjadi beberapa bagian antara lain:

Tabel 3. 1Bahan dan Alat membuat rangkaian

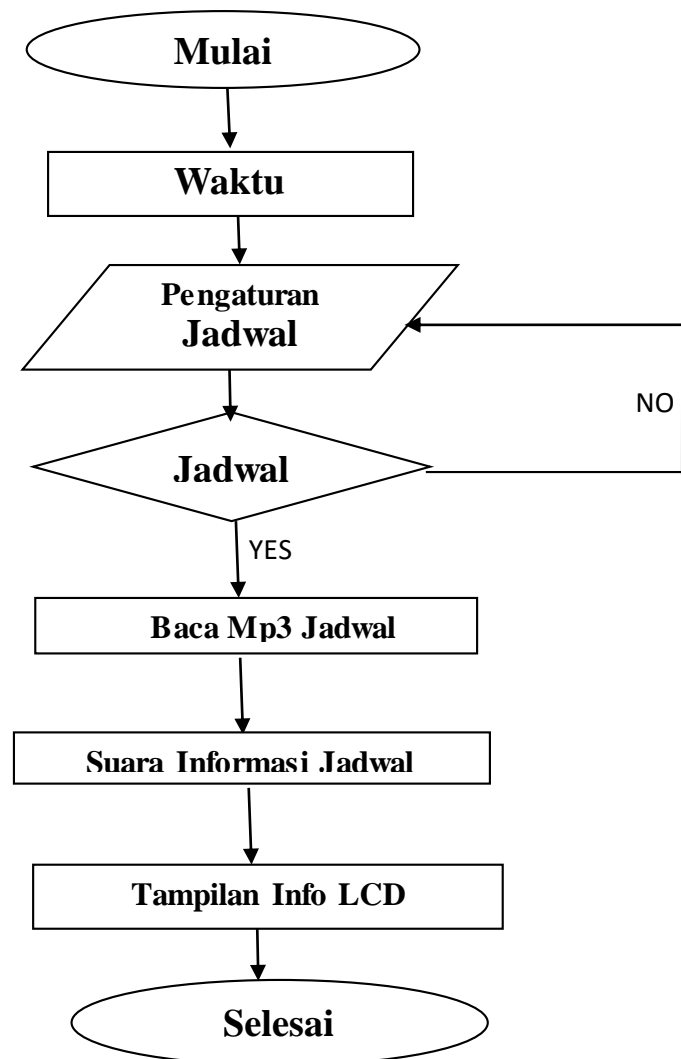
BAHAN	ALAT
Mikrokontroler Arduino	Obeng
LCD 16X2 + I2C	Solder
<i>Module</i> RTC DS3231	Multimeter
DFMini Player	
Speaker	
Saklar	
Micro SD	
Kabel Jumper	
Adaptor	
Laptop Asus Intel Core i3	

Sumber : Penulis

3.7.3 *Flowchart* Rancangan Alat

Ketika peratama kali alat di nyalakan sistem melakukan inisialisasi pin-pin Arduino sebagai input dan output, kemudian membaca jadwal kegiatan sesuai waktu yang telah dilakukan, apabila tanggal, jam, menit sesuai dengan waktu RTC DS 3231 maka Arduino membaca Mp3 sesuai jadwal. Arduino memrintahkan modul Mp3 untuk memutar Mp3 yang sesuai, lalu LCD menampilkan informasi nama kegiatan lalu berhenti.

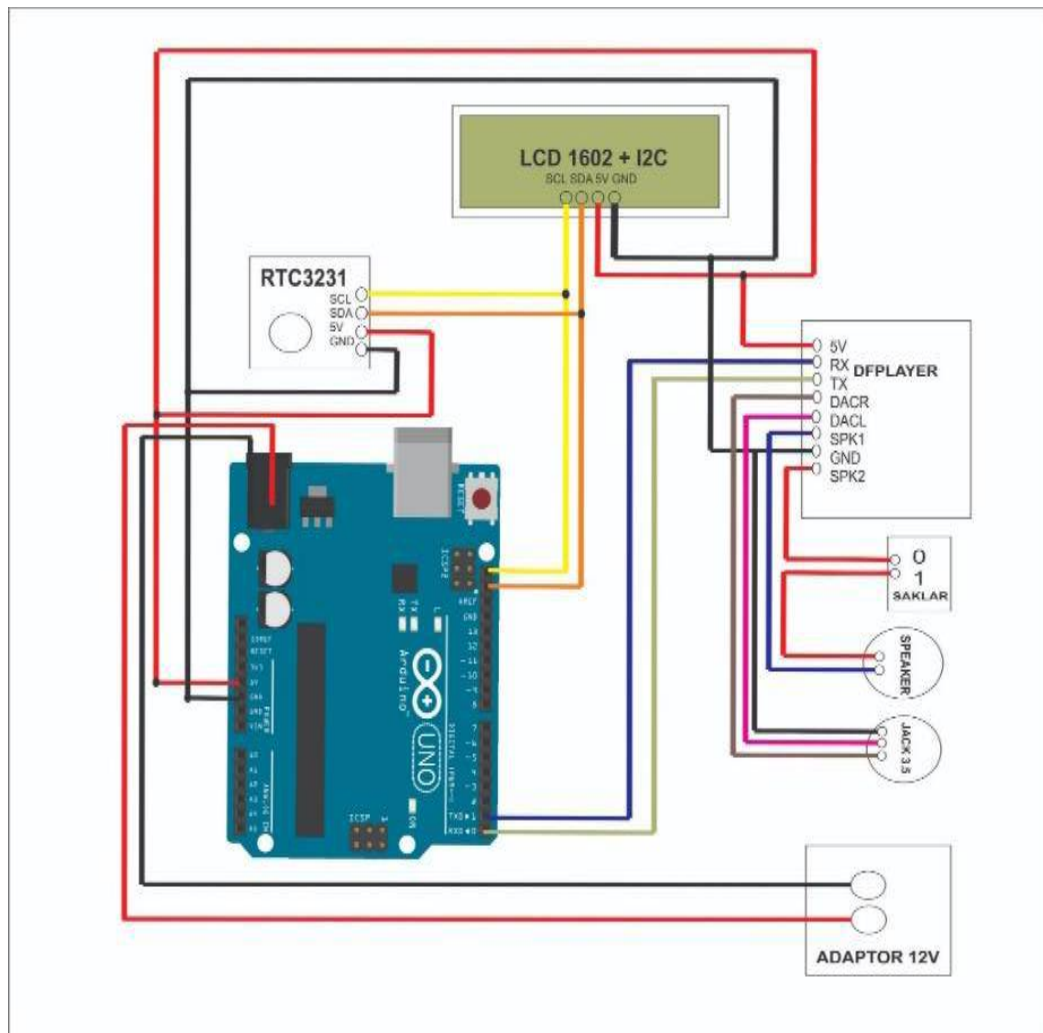
Flowchart untuk perancangan alat kendali pada pengingat jadwal dapat dilihat pada gambar berikut :



Gambar 3.4 *Flowchart* Proses Kendali
Sumber. Penulis 2019

3.7.4 Perancangan Perangkat Keras

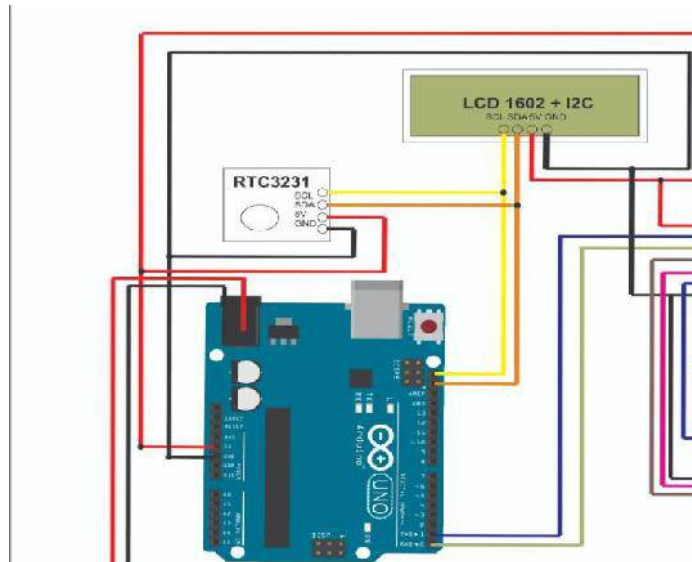
Pada perancangan pengingat otomatis berbasis arduino bagian rancangan perangkat keras terdiri dari rangkaian Modul RTC DS3231, *DFMiniPlayer*, *LCD*, berikut rangkaian :



Gambar 3.5 Rangkaian Keseluruhan
Sumber. Penulis 2019

3.7.5 Rangkaian LCDx12C

Rancangan Display LCD 2x16 berfungsi untuk menampilkan output hari/tanggal/jam dan jadwal kegiatan pada saat kegiatan berbunyi otomatis, LCD yang digunakan pada alat ini mempunyai lebar display 2 baris 16 kolom.



Gambar 3.6 Rangkaian LCDxI2C
Sumber. Penulis 2019

Pada alat indikator jadwal kegiatan otomatis ini digunakan modul I2C untuk mengendalikan LCD, penggunaan modul I2C ini dipilih guna menghemat penggunaan pin pada Arduino, sebab dengan menggunakan modul I2C maka pengiriman perintah LCD membutuhkan 4 pin yaitu SDA,SCL,VCC dan GND. Sedangkan untuk mengendalikan LCD dengan tanpa modul I2C dibutuhkan 6 pin digital Arduino

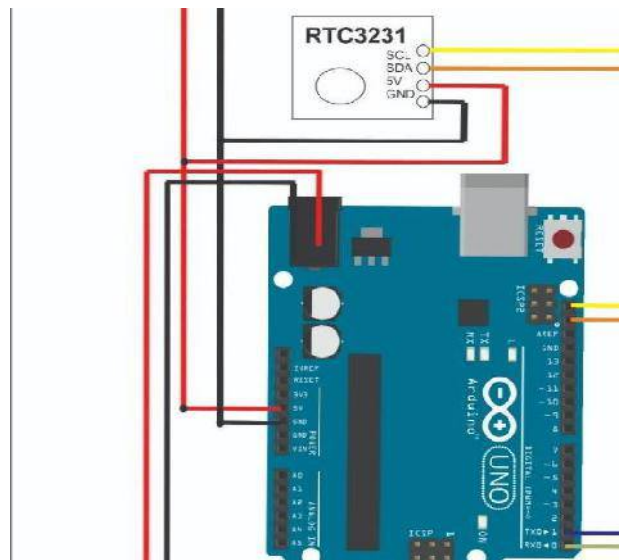
Tabel 3.2 Konfigurasi pin Arduino dengan I2C LCD

ARDUINO	I2C+LCD
Pin A4	SCL
Pin A5	SDA
Pin 5V	VCC
GND	GND

Sumber: Penulis, 2019

3.7.6 Rangkaian Modul RTC DS3231

Rangkaian *RTC* pada alat ini berfungsi untuk menghitung waktu secara terus-menerus mulai dari hari,tanggal,bulan,tahun,jam,menit dan detik IC yang digunakan sebagai pemberi informasi sehingga pengingat otomatis berfungsi adalah DS3231.

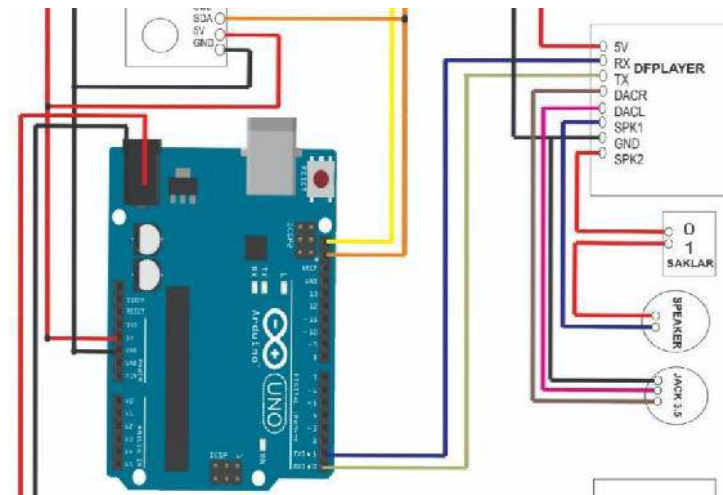


Gambar 3.7 Rangkaian RTC DS3231
Sumber Penulis 2019

3.7.7 Rangkain DFPlayer Mini

DFPlayer Mini adalah modul mp3 yang dapat langsung dihubungkan ke *speaker* dan dapat juga di kontrol melalui port serial, modul *Arduino Uno* atau mikrokontroler seri. Modul itu sendiri sempurna terintegrasi *hardware decode* MP3,WAV,WMA.

Rangkaian *DFPlayer Mini* berfungsi untuk menghasilkan suara nada yang diinginkan ketika waktu yang diperoleh dari RTC sudah sesuai dengan jadwal yang tersimpan dan yang telah di programkan. Nada atau suara yang dihasilkan berasal dari file Mp3 yang disimpan didalam kartu memori *microsd* yang dipasangkan pada modul *DFPlayer Mini*. Modul ini memiliki 16 pin yaitu : VCC,RX,TX,DAC_R,DAC_L,SPK2,GND,SPK1,IO1,GND,IO2,ADKEY1,ADKEY2,USB+,USB-,BUSY.



Gambar 3.8 Rangkaian *DFPLAYER MINI*
Sumber. Penulis 2019

3.8 Perancangan *Software*

Perancangan *software* pada proses kendali bel otomatis dimulai dengan menginput jadwal bel melalui *software* Arduino IDE yang berfungsi sebagai interface bagi operator untuk menyesuaikan waktu dan jadwal terhadap mikrokontroler arduino. Setelah proses pengaturan waktu dan tanggal serta input jadwal kegiatan dilakukan, maka pengingat otomatis siap digunakan.

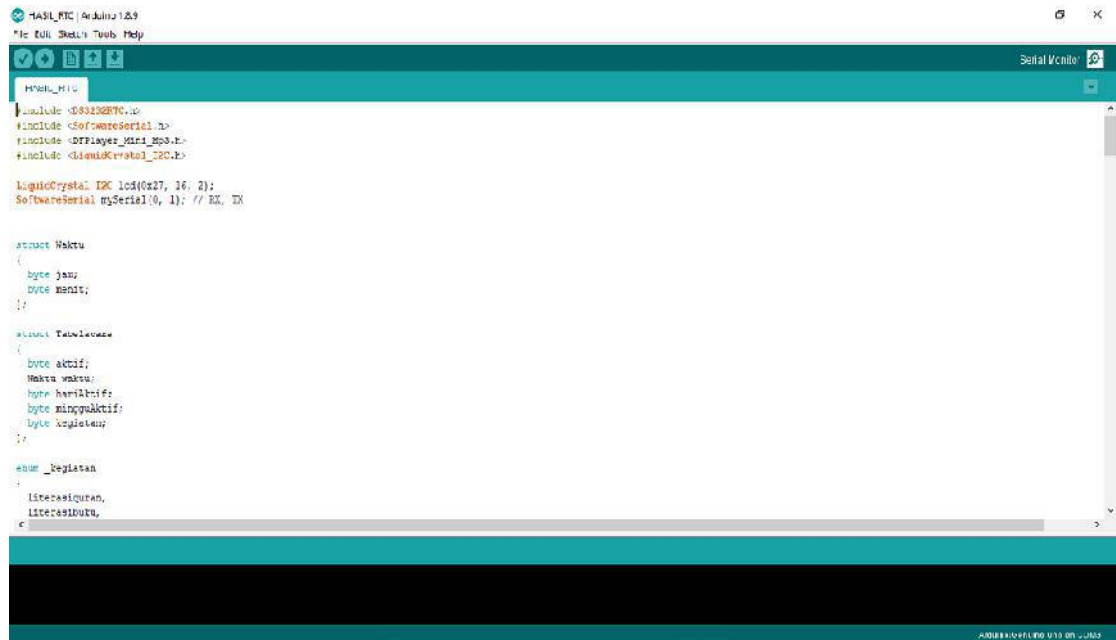
3.8.1 Ke butuhan Perangkat Lunak (*Software*)

Untuk merancang alat kendali pada bel ini dibutuhkan perangkat lunak (*software*) sebagai berikut :

1. Arduino IDE

3.8.2 Rancangan Program

Agar mikrokontroller dapat bekerja maka pada mikrokontroller harus dimasukkan program atau deprogram. Program yang digunakan dalam rangkaian ini yakni bahasa programan C.



```

HWAHU_M10
#include <DS1302RTC.h>
#include <SoftwareSerial.h>
#include <MP3Player_Mini_MP3.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C LCD(0x27, 16, 2);
SoftwareSerial mySerial(0, 1); // RX, TX

struct waktu
{
  byte jam;
  byte menit;
};

enum Tahap
{
  byte aktif;
  waktu waktu;
  byte hariaktif;
  byte mingguaktif;
  byte kegiatan;
};

enum kegiatan
{
  istirahat,
  belajar,
};

```

Gambar 3.9 Bahasa Programan C

Sumber : Penulis 2019

3.8.3 Proses Input Jadwal Kegiatan Sekolah

Pada aplikasi Arduino IDE proses input jadwal bel sekolah dapat dilakukan dengan mudah, adapun tampilan dari software Arduino IDE adalah sebagai berikut :

```

HASIL_RTC | Arduino 1.8.9
File Edit Sketch Tools Help

HASIL_RTC

//Senin
jadwalBelajar[i++] = (Aktif, _waktu( 7, 30 ), hariAktifSenin,          _mingguAktif(0,1,0,1,0), upacara);
jadwalBelajar[i++] = (Aktif, _waktu( 7, 30 ), hariAktifSenin,          _mingguAktif(1,0,0,0,0), literasiquran);
jadwalBelajar[i++] = (Aktif, _waktu( 7, 30 ), hariAktifSenin,          _mingguAktif(0,0,1,0,0), literasibuku);

//Rabu
jadwalBelajar[i++] = (Aktif, _waktu( 7, 40 ), hariAktifRabu,           _mingguAktif(0,1,0,1,0), senam);
jadwalBelajar[i++] = (Aktif, _waktu( 7, 30 ), hariAktifRabu,           _mingguAktif(1,0,0,0,0), literasiquran);
jadwalBelajar[i++] = (Aktif, _waktu( 7, 30 ), hariAktifRabu,           _mingguAktif(0,0,1,0,0), literasibuku);

//-----
jadwalBelajar[i++] = (Aktif, _waktu( 9, 45 ),          _hariAktif(1,0,1,0,0,0,0),          _mingguAktif(1,1,1,1,1), selatduha);
jadwalBelajar[i++] = (Aktif, _waktu( 8, 40 ),          _hariAktif(0,1,0,0,0,0,0),          _mingguAktif(1,1,1,1,1), selatduha);
jadwalBelajar[i++] = (Aktif, _waktu( 7, 40 ),          _hariAktif(0,1,0,1,1,1,0),          _mingguAktif(1,1,1,1,1), selatduha);
jadwalBelajar[i++] = (Aktif, _waktu( 12, 15 ),          _hariAktif(1,1,1,1,0,0,0),          _mingguAktif(1,1,1,1,1), selatduha);
jadwalBelajar[i++] = (Aktif, _waktu( 12, 0 ),          _hariAktif(0,0,0,1,0,0,0),          _mingguAktif(1,1,1,1,1), salatjumat);
jadwalBelajar[i++] = (Aktif, _waktu( 13, 13 ),          _hariAktif(1,1,1,1,0,0,0),          _mingguAktif(1,1,1,1,1), bersih);
jadwalBelajar[i++] = (Aktif, _waktu( 13, 0 ),          _hariAktif(0,0,0,1,0,0,0),          _mingguAktif(1,1,1,1,1), bersih);
jadwalBelajar[i++] = (Aktif, _waktu( 10, 0 ),          _hariAktif(0,0,0,0,1,0,0),          _mingguAktif(1,1,1,1,1), bersih);
jadwalBelajar[i++] = (Aktif, _waktu( 13, 30 ),          _hariAktif(1,1,1,1,0,0,0),          _mingguAktif(1,1,1,1,1), ekskul);
jadwalBelajar[i++] = (Aktif, _waktu( 13, 15 ),          _hariAktif(0,0,0,1,0,0,0),          _mingguAktif(1,1,1,1,1), ekskul);
jadwalBelajar[i++] = (Aktif, _waktu( 10, 30 ),          _hariAktif(0,0,0,0,1,0,0),          _mingguAktif(1,1,1,1,1), ekskul);
jadwalBelajar[i++] = (Aktif, _waktu( 14, 45 ),          _hariAktif(1,1,1,1,0,0,0),          _mingguAktif(1,1,1,1,1), mengeja);

//-----

```

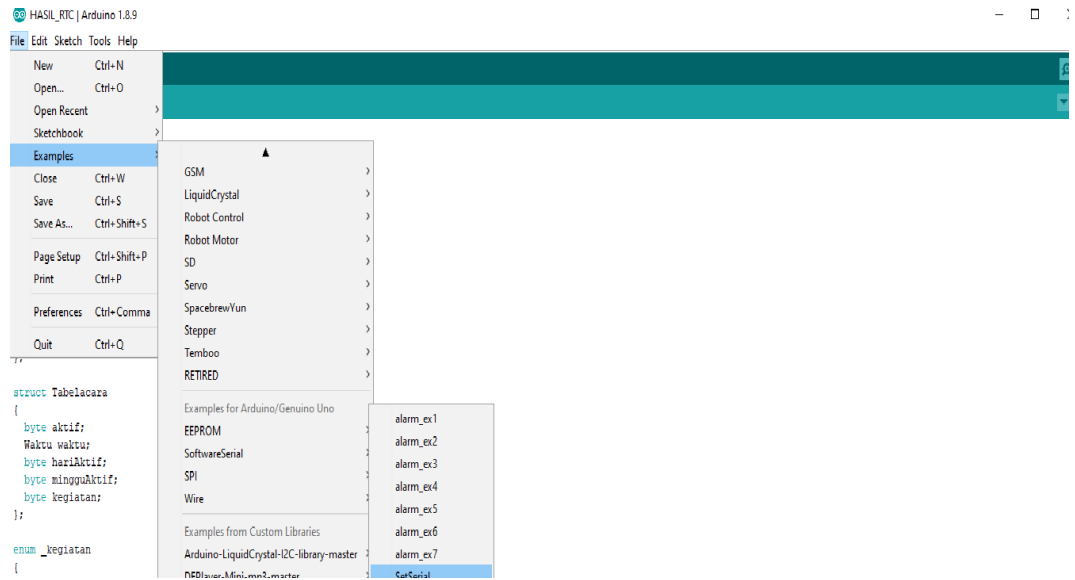
Gambar 3.10 Jadwal Kegiatan
Sumber : Penulis 2019

3.8.4 Proses Upload Jadwal kedalam Mikrokontroler Arduino

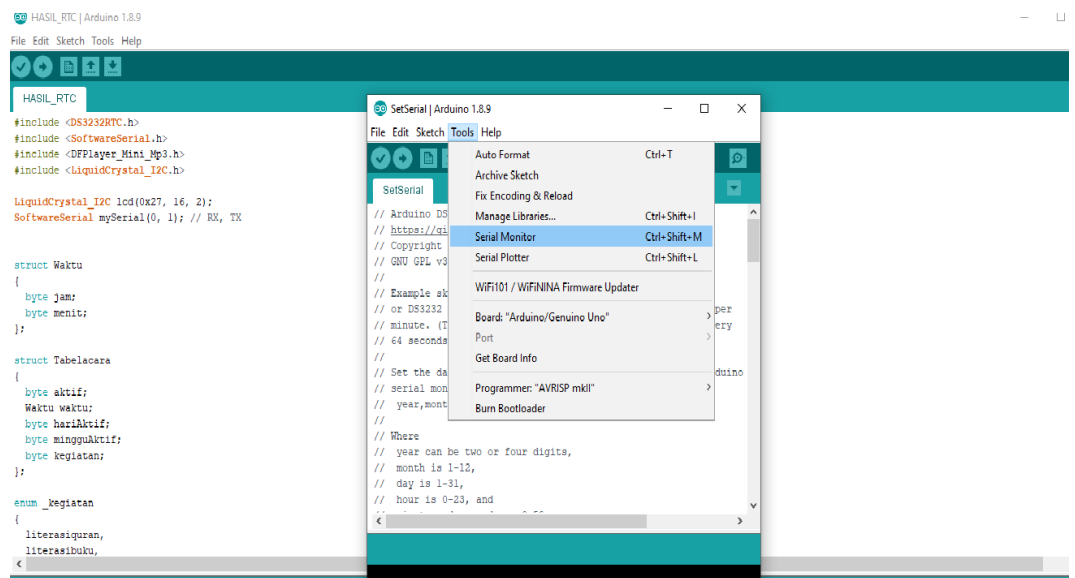
Setelah seluruh proses input data jadwal bel sekolah selesai dilakukan maka langkah selanjutnya adalah mentransfer jadwal yang tersimpan di PC/Laptop kedalam mikrokontroler Arduino. Adapun upload data jadwal dapat dilakukan dengan cara :

1. Hubungkan mikrokontroler Arduino melalui port USB ke PC/ Laptop kemudian
2. Buka *software* Arduino IDE
3. Pada bagian menu utama klik File pilih examples – DS3232RTC – Set serial- upload – Tools – Serial Monitor – enter- lalu Upload program.

Berikut Gambar alur upload program



Gambar 3.11 Alur upload program
Sumber : Penulis 2019



Gambar 3.12 Alur upload program (finish)
Sumber : Penulis 2019

BAB 4

HASIL DAN PEMBAHASAN

Dalam bab ini membahas pengujian alat, apakah alat yang telah direncanakan bekerja dengan baik atau tidak. Pengujian sistem dilakukan oleh penulis adalah :

1. Pengujian LCD dan I2C
2. Pengujian *Module* RTC DS3231
3. Pengujian *Module DF Mini Player*

Pengujian ini dilakukan berdasarkan pada masing-masing rangkaian pendukung secara keseluruhan .



Gambar 4.1 Tampilan Alat pengingat Otomatis Berbasis Arduino
Sumber : Penulis

4.1 Pengujian Rangkaian LCD dan I2C

Pada tahap ini dilakukan percobaan untuk menguji *display* LCD yang akan menampilkan bebrapa karakter pada LCD yang diinstruksikan, dimulai dari melakukan inisialisasi setiap set instruks.

```
lcd.setCursor(0,0);
```

```
lcd.print("  ");  
  
print2digits(tm.Hour);  
  
lcd.print(':');  
  
print2digits(tm.Minute);  
  
lcd.setCursor(0,1);  
  
lcd.print(namaHari[tm.Wday- 1]);  
  
lcd.print(' ');  
  
lcd.print(tm.Day);  
  
lcd.print('/');  
  
lcd.print(tm.Month);  
  
lcd.print('/');  
  
lcd.print(tmYearToCalendar(tm.Year));  
  
}  
  
void print2digits(int number) {  
  
  if (number >= 0 && number < 10) {  
  
    Serial.write('0');  
  
  }  
  
  lcd.print(number);  
  
}
```

Program diatas setelah di upload akan menampilkan jumlah jam dan menit yang terus bertambah di baris pertama dan menampilkan tanggal, hari, bulan, tahun

pada baris kedua pada LCD 2X16. Dengan demikian dapat dikatakan bahwa rangkaian LCD dan programnya telah berjalan sesuai rancangan.



Gambar 4.2 Uji Coba LCD+I2C
Sumber : Penulis

4.2 Pengujian *Real Time Clock* (RTC)

Pengujian RTC dalam alat hanya menggunakan beberapa alamat yang dipakai dalam alat. Dalam RTC sudah tersedia banyak alamat yang untuk mempermudah dalam mengaksesnya. Dalam alat ini memakai jam, menit, tanggal, bulan dan tahun. Blok untuk pengujian seperti gambar dibawah:

```
RTC.read(tm);

if(detikSebelumnya != tm.Second)
{
    detikSebelumnya = tm.Second;
    if(tm.Second == 0)
    {
```

```
TampilkanWaktu();/***
```

```
byte hariKeDiTanggall = (((tm.Wday + 8) - (tm.Day % 7)) % 7);
```

```
byte SeninKe = ((tm.Day + 7 - hariKeDiTanggall) / 7);
```

```
byte mingguKe = ((tm.Day + 8 - hariKeDiTanggall) / 7) + 1;
```

```
for(byte i=0; i<indexacara ;i++)
```

```
{
```

```
if(jadwalBelajar[i].aktif)
```

```
{
```

```
if((jadwalBelajar[i].waktu.jam == tm.Hour) &&
```

```
(jadwalBelajar[i].waktu.menit == tm.Minute) &&
```

```
(jadwalBelajar[i].hariAktif & (1<<(8-tm.Wday))) &&
```

```
(jadwalBelajar[i].mingguAktif & (1<<(8-SeninKe))))
```

```
{
```

```
TampilkanWaktu();
```

```
//Bisa ditambahkan sistem output
```

```
switch(jadwalBelajar[i].kegiatan)
```

```
{
```

```
case literasiquran:
```

```
    lcd.clear();
```

```
    lcd.setCursor(0,0);
```



```
lcd.print
("LiterasiQURAN ");
mp3_play (14);
delay(3500);
mp3_play (9);
delay(17500);
mp3_play (9);
delay(17500);
mp3_play (14);
break;
case literasibuku:
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print
("Literasibuku ");
mp3_play (14);
delay(3500);
mp3_play (3);
delay(19500);
mp3_play (3);
delay(19500);
mp3_play (14);
```

```
break;

case upacara:

    lcd.clear();

    lcd.setCursor(0,0);

    lcd.print

    ("Waktunya upacara ");

    mp3_play (14);

    delay(3500);

    mp3_play (12);

    delay(17500);

    mp3_play (12);

    delay(17500);

    mp3_play (14);

    break;

case senam:

    lcd.clear();

    lcd.setCursor(0,0);

    lcd.print

    ("Waktunya senam ");

    mp3_play (14);

    delay(3500);

    mp3_play (4);
```

```
    delay(19000);  
  
    mp3_play (4);  
  
    delay(19000);  
  
    mp3_play (14);  
  
    break;  
  
case salatdhuha:  
  
    lcd.clear();  
  
    lcd.setCursor(0,0);  
  
    lcd.print  
    ("Jam salatdhuha ");  
  
    mp3_play (14);  
  
    delay(3500);  
  
    mp3_play (2);  
  
    delay(19000);  
  
    mp3_play (2);  
  
    delay(19000);  
  
    mp3_play (14);  
  
    break;  
  
case salatdzuhur:  
  
    lcd.clear();  
  
    lcd.setCursor(0,0);  
  
    lcd.print
```

```
("Jam salat dzuhur ");  
  
mp3_play (14);  
  
delay(3500);  
  
mp3_play (7);  
  
delay(20000);  
  
mp3_play (7);  
  
delay(20000);  
  
mp3_play (14);  
  
break;  
  
case bersih:  
  
  lcd.clear();  
  
  lcd.setCursor(0,0);  
  
  lcd.print  
  
  ("Waktunya bersih bersih ");  
  
  mp3_play (14);  
  
  delay(3500);  
  
  mp3_play (5);  
  
  delay(18000);  
  
  mp3_play (5);  
  
  delay(18000);  
  
  mp3_play (14);  
  
  break;
```

```
case ekskul:

    lcd.clear();

    lcd.setCursor(0,0);

    lcd.print

    ("Waktunya ekskul ");

    mp3_play (14);

    delay(3500);

    mp3_play (6);

    delay(20000);

    mp3_play (6);

    delay(20000);

    mp3_play (14);

    break;

case mengaji:

    lcd.clear();

    lcd.setCursor(0,0);

    lcd.print

    ("Waktunya Mengaji ");

    mp3_play (14);

    delay(3500);

    mp3_play (10);
```

```
    delay(17500);  
  
    mp3_play (10);  
  
    delay(17500);  
  
    mp3_play (14);  
  
    break;  
  
case salatjumat:  
  
    lcd.clear();  
  
    lcd.setCursor(0,0);  
  
    lcd.print  
    ("Shalat JUMAT ");  
  
    mp3_play (14);  
  
    delay(3500);  
  
    mp3_play (8);  
  
    delay(18000);  
  
    mp3_play (8);  
  
    delay(18000);  
  
    mp3_play (14);  
  
    break;
```

4.3 Pengujian *DF Mini Player* (Pengujian Keseluruhan)

Pada pengujian ini merupakan pengujian pengujian pengingat otomatis sesuai dengan jadwal yang telah di inputkan pada program ada beberapa perbedaan

setiap minggu pada jadwal kegiatan di SD Panca Budi Medan. Berikut Kondisi pengingat otomatis di SD Panca Budi Medan yang di dapat dari observasi ke tata usaha SD Panca Budi Medan terdapat 10 kegiatan yaitu :

1. Upacara
2. Literasiquran
3. Literasibuku
4. Senam
5. Salat dhuha
6. Salat dzuhur
7. Salat jumat
8. Bersih
9. Ekskul
10. Mengaji

4.3.1 Pengingat Upacara

Pengingat pertama kali berbunyi yaitu pada jam 07.30 WIB, sekolah SD Panca Budi Medan setiap minggunya masuk pada hari senin-sabtu. Pola pengingat upacara hanya bunyi pada minggu kedua dan keempat. Adapun program menguji pengingat otomatis di SD Panca Budi Medan yaitu:

Dimisalkan tahun 2020, tanggal 10 Februari, bulan 2, jam 7 : menit 30, sesuai dengan inputan program dibawah yaitu RTC memanggil waktu yang diinputkan

pada program sehingga bel berbunyi sesuai dengan file audio yang diinputkan pada jam 07.30 WIB.

```
#include <DS3232RTC.h>
```

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

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

```
#include <LiquidCrystal_I2C.h>
```

```
LiquidCrystal_I2C lcd(0x27, 16, 2);
```

```
SoftwareSerial mySerial(0, 1); // RX, TX
```

```
struct Waktu
```

```
{
```

```
    byte jam;
```

```
    byte menit;
```

```
};
```

```
struct Tabelacara
```

```
{
```

```
    byte aktif;
```

```
    Waktu waktu;
```

```
    byte hariAktif;
```



```
byte mingguAktif;

byte kegiatan;

};

enum _kegiatan
{
    literasiquran,
    literasibuku,
    upacara,
    senam,
    salatdhuha,
    salatdzuhur,
    bersih,
    ekskul,
    mengaji,
    salatjumat,

};

volatile bool interupsiDetik;

byte indexacara;

byte detikSebelumnya = 60;
```

```

#define hariAktifSenin 1<<6

#define hariAktifRabu 1<<4

StringnamaHari[]={ "Minggu", "Senin", "Selasa", "Rabu", "Kamis", "Jum'at", "Sabtu;

#define _hariAktif(Sen, Sel, Rab, Kam, Jum, Sab, Mgu)

(Mgu<<7)|(Sen<<6)|(Sel<<5)|(Rab<<4)|(Kam<<3)|(Jum<<2)|(Sab<<1)

#define _mingguAktif(Mgu1, Mgu2, Mgu3, Mgu4, Mgu5)

(Mgu1<<7)|(Mgu2<<6)|(Mgu3<<5)|(Mgu4<<4)|(Mgu5<<3)

#define _waktu(Jam, Menit) {Jam, Menit}

#define Aktif 1

#define TidakAktif 0

Tabelacara jadwalBelajar[35];

Waktu waktu;

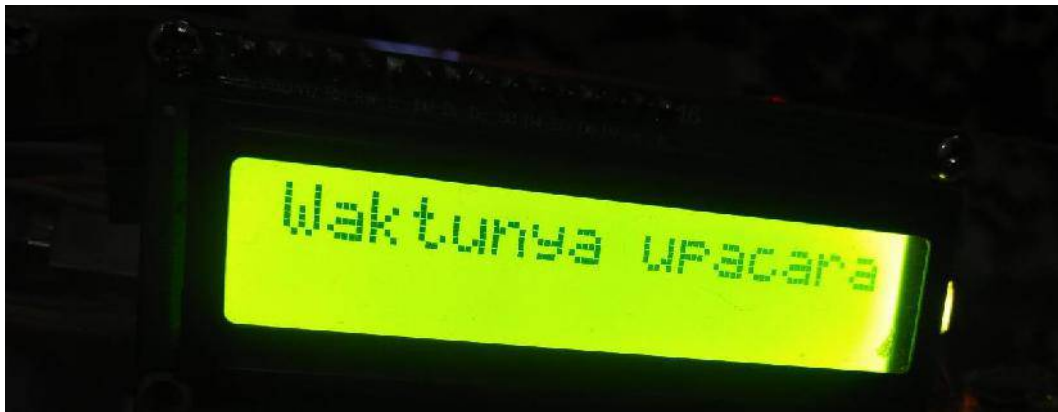
tmElements_t tm;

void setup() {
  byte i = 0;
  //-----

  //Senin

```

```
jadwalBelajar[i++] = {Aktif, _waktu( 7, 30 ), hariAktifSenin,
_mingguAktif(0,1,0,1,0), upacara};
```



Gambar 4.3 Uji Coba Kegiatan Upacara
Sumber: Penulis

4.3.2 Pengingat Literasi Quran

Pada kondisi ini pengingat otomatis berbunyi ketika masuk waktunya untuk siswas/siswi SD Panca Budi Medan Literasi Quran yang terjadi hanya pada minggu pertama dihari senin dan rabu. Dimisalkan tahun 2020, tanggal 3, bulan 2, jam 7 : menit 30 sesuai dengan inputan program dibawah yaitu RTC memanggil waktu yang diinputkan pada program sehingga bel berbunyi sesuai dengan file audio yang diinputkan pada jam 07.30 WIB.

```
#include <DS3232RTC.h>
```

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

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

```
#include <LiquidCrystal_I2C.h>
```

```
LiquidCrystal_I2C lcd(0x27, 16, 2);  
  
SoftwareSerial mySerial(0, 1); // RX, TX  
  
struct Waktu  
{  
    byte jam;  
    byte menit;  
};  
  
struct Tabelacara  
{  
    byte aktif;  
    Waktu waktu;  
    byte hariAktif;  
    byte mingguAktif;  
    byte kegiatan;  
};  
  
enum _kegiatan  
{  
    literasiquran,  
    literasibuku,  
    upacara,
```

```

senam,

salatdhuha,

salatdzuhur,

bersih,

ekskul,

mengaji,

salatjumat,

};

volatile bool interupsiDetik;

byte indexacara;

byte detikSebelumnya = 60;

#define hariAktifSenin 1<<6

#define hariAktifRabu 1<<4

StringnamaHari[]={ "Minggu", "Senin", "Selasa", "Rabu", "Kamis", "Jum'at", "Sabtu;

#define _hariAktif(Sen,Se1,Rab,Kam,Jum,Sab,Mgu)

(Mgu<<7)|(Sen<<6)|(Se1<<5)|(Rab<<4)|(Kam<<3)|(Jum<<2)|(Sab<<1)

#define _mingguAktif(Mgu1,Mgu2,Mgu3,Mgu4,Mgu5)

(Mgu1<<7)|(Mgu2<<6)|(Mgu3<<5)|(Mgu4<<4)|(Mgu5<<3)

```

```

#define _waktu(Jam, Menit) {Jam, Menit}

#define Aktif 1

#define TidakAktif 0

Tabelacara jadwalBelajar[35];

Waktu waktu;

tmElements_t tm;

void setup() {

    byte i = 0;

    //-----

    //Senin

    jadwalBelajar[i++] = {Aktif, _waktu( 7, 30 ), hariAktifSenin,
    _mingguAktif(1,0,0,0,0), literasiquran};

    //Rabu

    jadwalBelajar[i++] = {Aktif, _waktu( 7, 30 ), hariAktifRabu,
    _mingguAktif(1,0,0,0,0), literasiquran};

```



Gambar 4.4 Uji Coba Literasi Quran
Sumber : Penulis

4.3.3 Pengingat Literasi Buku

Pada kondisi ini pengingat otomatis berbunyi ketika masuk waktunya untuk siswa/siswi SD Panca Budi Medan Literasi Buku yang terjadi hanya pada minggu ke tiga di hari senin dan rabu. Dimisalkan tahun 2020, tanggal 17, bulan 2, jam 7: menit 30 sesuai dengan inputan program dibawah yaitu RTC memanggil waktu yang diinputkan pada program sehingga bel berbunyi sesuai dengan file audio yang diinputkan pada jam 07.30 WIB.

```
#include <DS3232RTC.h>
```

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

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

```
#include <LiquidCrystal_I2C.h>
```

```
LiquidCrystal_I2C lcd(0x27, 16, 2);
```

```
SoftwareSerial mySerial(0, 1); // RX, TX
```

```
struct Waktu
```

```
{  
    byte jam;  
    byte menit;  
};
```

```
struct Tabelacara
```

```
{  
    byte aktif;  
    Waktu waktu;  
    byte hariAktif;  
    byte mingguAktif;  
    byte kegiatan;  
};
```

```
enum _kegiatan
```

```
{  
    literasiquran,  
    literasibuku,  
    upacara,  
    senam,  
    salatdhuha,
```



```

    salatdzuhur,

    bersih,

    ekskul,

    mengaji,

    salatjumat,

};

volatile bool interupsiDetik;

byte indexacara;

byte detikSebelumnya = 60;

#define hariAktifSenin 1<<6

#define hariAktifRabu 1<<4

StringnamaHari[]={ "Minggu", "Senin", "Selasa", "Rabu", "Kamis", "Jum'at", "Sabtu;

#define _hariAktif(Sen, Sel, Rab, Kam, Jum, Sab, Mgu)

(Mgu<<7)|(Sen<<6)|(Sel<<5)|(Rab<<4)|(Kam<<3)|(Jum<<2)|(Sab<<1)

#define _mingguAktif(Mgu1, Mgu2, Mgu3, Mgu4, Mgu5)

(Mgu1<<7)|(Mgu2<<6)|(Mgu3<<5)|(Mgu4<<4)|(Mgu5<<3)

#define _waktu(Jam, Menit) {Jam, Menit}

#define Aktif 1

#define TidakAktif 0

```

```

Tabelacara jadwalBelajar[35];

Waktu waktu;

tmElements_t tm;

void setup() {
  byte i = 0;

  //-----

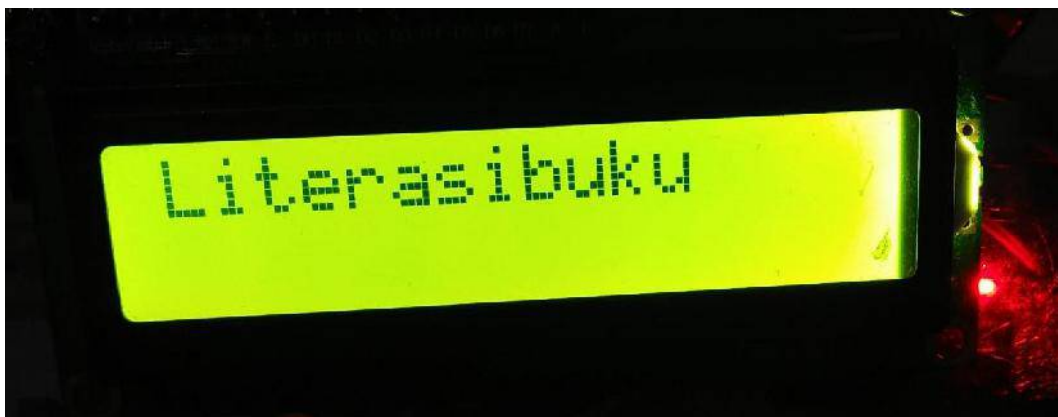
  //Senin

  jadwalBelajar[i++] = {Aktif, _waktu( 7, 30 ), hariAktifSenin,
    _mingguAktif(0,0,1,0,0), literasibuku};

  //Rabu

  jadwalBelajar[i++] = {Aktif, _waktu( 7, 30 ), hariAktifRabu,
    _mingguAktif(0,0,1,0,0), literasibuku};

```



Gambar 4.5 Uji Coba Literasi Buku
Sumber : Penulis

4.3.4 Pengingat Senam

Pada kondisi ini pengingat otomatis berbunyi ketika masuk waktu nya untuk siswa/siswi SD Panca Budi Medan Senam yang terjadi hanya pada minggu ke dua dan ke empat di hari rabu. Dimisalkan tahun 2020, tanggal 17, bulan 2, jam 7: menit 30 sesuai dengan inputan program dibawah yaitu RTC memanggil waktu yang diinputkan pada program sehingga bel berbunyi sesuai dengan file audio yang diinputkan pada jam 07.40 WIB.

```
#include <DS3232RTC.h>
```

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

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

```
#include <LiquidCrystal_I2C.h>
```

```
LiquidCrystal_I2C lcd(0x27, 16, 2);
```

```
SoftwareSerial mySerial(0, 1); // RX, TX
```

```
struct Waktu
```

```
{
```

```
    byte jam;
```

```
    byte menit;
```

```
};
```

```
struct Tabelacara
```

```
{
```

```
byte aktif;  
  
Waktu waktu;  
  
byte hariAktif;  
  
byte mingguAktif;  
  
byte kegiatan;  
  
};
```

```
enum _kegiatan  
{  
  
    literasiquran,  
  
    literasibuku,  
  
    upacara,  
  
    senam,  
  
    salatdhuha,  
  
    salatdzuhur,  
  
    bersih,  
  
    ekskul,  
  
    mengaji,  
  
    salatjumat,  
  
};
```

```

volatile bool interupsiDetik;

byte indexacara;

byte detikSebelumnya = 60;

#define hariAktifSenin 1<<6

#define hariAktifRabu 1<<4

StringnamaHari[]={ "Minggu", "Senin", "Selasa", "Rabu", "Kamis", "Jum'at", "Sabtu;

#define _hariAktif(Sen,Se1,Rab,Kam,Jum,Sab,Mgu)

(Mgu<<7)|(Sen<<6)|(Se1<<5)|(Rab<<4)|(Kam<<3)|(Jum<<2)|(Sab<<1)

#define _mingguAktif(Mgu1,Mgu2,Mgu3,Mgu4,Mgu5)

(Mgu1<<7)|(Mgu2<<6)|(Mgu3<<5)|(Mgu4<<4)|(Mgu5<<3)

#define _waktu(Jam, Menit) {Jam, Menit}

#define Aktif 1

#define TidakAktif 0

Tabelacara jadwalBelajar[35];

Waktu waktu;

tmElements_t tm;

void setup() {

    byte i = 0;

```

```
//-----
//Rabu

jadwalBelajar[i++] = {Aktif, _waktu( 7, 40 ), hariAktifRabu,
_mingguAktif(0,1,0,1,0), senam};
```



Gambar 4.6 Uji Coba Kegiatan Senam
Sumber : Penulis

4.3.5 Pengingat Salat Dhuha

Pada kondisi ini pengingat otomatis berbunyi ketika masuk waktu nya untuk siswa/siswi SD Panca Budi Medan melaksanakan Salat dhuha yang terjadi di setiap hari senin-sabtu. Namun ada perbedaan waktu diantara hari-hari tersebut dikarenakan ada jadwal lain nya namun penulis sudah menginput jadwal sesuai dengan data yang diterima, seperti hari senin dan rabu jam 09:45, sedangkan di hari selasa, kamis, jumat dan sabtu jam 07.40. Dimisalkan tahun 2020, tanggal 11, bulan 2, jam 7 : menit 40 sesuai dengan inputan program dibawah yaitu RTC memanggil waktu yang diinputkan pada program sehingga bel berbunyi sesuai dengan file audio yang diinputkan pada jam 07.40 WIB.

```
#include <DS3232RTC.h>
```

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

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

```
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 16, 2);
SoftwareSerial mySerial(0, 1); // RX, TX

struct Waktu
{
    byte jam;
    byte menit;
};

struct Tabelacara
{
    byte aktif;
    Waktu waktu;
    byte hariAktif;
    byte mingguAktif;
    byte kegiatan;
};

enum _kegiatan
{
    literasiquran,
```

```

literasibuku,
upacara,
senam,
salatdhuha,
salatdzuhur,
bersih,
ekskul,
mengaji,
salatjumat,

};

volatile bool interupsiDetik;
byte indexacara;
byte detikSebelumnya = 60;

#define hariAktifSenin 1<<6
#define hariAktifRabu 1<<4

StringnamaHari[]={ "Minggu", "Senin", "Selasa", "Rabu", "Kamis", "Jum'at", "Sabtu;
#define _hariAktif(Sen, Sel, Rab, Kam, Jum, Sab, Mgu)
(Mgu<<7)|(Sen<<6)|(Sel<<5)|(Rab<<4)|(Kam<<3)|(Jum<<2)|(Sab<<1)

```



```

#define _mingguAktif(Mgu1,Mgu2,Mgu3,Mgu4,Mgu5)
(Mgu1<<7)|(Mgu2<<6)|(Mgu3<<5)|(Mgu4<<4)|(Mgu5<<3)

#define _waktu(Jam, Menit) {Jam, Menit}

#define Aktif 1

#define TidakAktif 0

Tabelacara jadwalBelajar[35];

Waktu waktu;

tmElements_t tm;

void setup() {
    byte i = 0;

    //-----
    jadwalBelajar[i++]={Aktif,_waktu(9,45 ), _hariAktif(1,0,1,0,0,0,0),
_mingguAktif(1,1,1,1,1),salatdhuha};
    jadwalBelajar[i++]={Aktif,_waktu(8,40), _hariAktif(0,1,0,0,0,0,0),
_mingguAktif(1,1,1,1,1),salatdhuha};
    jadwalBelajar[i++]={Aktif,_waktu(7,40), _hariAktif(0,1,0,1,1,1,0),
_mingguAktif(1,1,1,1,1),salatdhuha};

```



Gambar 4.7 Uji Coba Shalat Dhuha
Sumber : Penulis

4.3.6 Pengingat Salat dzuhur

Pada kondisi ini pengingat otomatis berbunyi ketika masuk waktu nya untuk siswa/siswi SD Panca Budi Medan melaksanakan Salat dhuha yang terjadi di setiap hari senin-kamis. Dimisalkan tahun 2020, tanggal 11, bulan 2, jam 12 : menit 15, sesuai dengan inputan program dibawah yaitu RTC memanggil waktu yang diinputkan pada program sehingga bel berbunyi sesuai dengan file audio yang diinputkan pada jam 12.15 WIB.

```
#include <DS3232RTC.h>
```

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

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

```
#include <LiquidCrystal_I2C.h>
```

```
LiquidCrystal_I2C lcd(0x27, 16, 2);
```

```
SoftwareSerial mySerial(0, 1); // RX, TX
```

```
struct Waktu
```

```
{  
    byte jam;  
    byte menit;  
};  
  
struct Tabelacara  
{  
    byte aktif;  
    Waktu waktu;  
    byte hariAktif;  
    byte mingguAktif;  
    byte kegiatan;  
};  
enum _kegiatan  
{  
    literasiquran,  
    literasibuku,  
    upacara,  
    senam,  
    salatdhuha,  
    salatdzuhur,  
    bersih,
```

```

    ekskul,

    mengaji,

    salatjumat,

};

volatile bool interupsiDetik;

byte indexacara;

byte detikSebelumnya = 60;

#define hariAktifSenin 1<<6

#define hariAktifRabu 1<<4

StringnamaHari[]={ "Minggu", "Senin", "Selasa", "Rabu", "Kamis", "Jum'at", "Sabtu;

#define _hariAktif(Sen, Sel, Rab, Kam, Jum, Sab, Mgu)

(Mgu<<7)|(Sen<<6)|(Sel<<5)|(Rab<<4)|(Kam<<3)|(Jum<<2)|(Sab<<1)

#define _mingguAktif(Mgu1, Mgu2, Mgu3, Mgu4, Mgu5)

(Mgu1<<7)|(Mgu2<<6)|(Mgu3<<5)|(Mgu4<<4)|(Mgu5<<3)

#define _waktu(Jam, Menit) {Jam, Menit}

#define Aktif 1

#define TidakAktif 0

Tabelacara jadwalBelajar[35];

```

```

Waktu waktu;

tmElements_t tm;

void setup() {

  byte i = 0;

  //-----

  jadwalBelajar[i++]={Aktif,_waktu(12,15 ),           _hariAktif(1,1,1,1,0,0,0),

  _mingguAktif(1,1,1,1,1), salatdzuhur};

```



Gambar 4.8 Uji Coba Kegiatan Shalat Dzuhur
Sumber : Penulis

4.3.7 Peningat Salat Jumat

Pada kondisi ini pengingat otomatis berbunyi ketika masuk waktu nya untuk siswa/siswi SD Panca Budi Medan melaksanakan Salat Jumat yang di laksanakan setiap hari Jumat. Dimisalkan tahun 2020, tanggal 14, bulan 2, jam 12 : menit 00, sesuai dengan inputan program dibawah yaitu RTC memanggil waktu yang

diinputkan pada program sehingga bel berbunyi sesuai dengan file audio yang diinputkan pada jam 12.00 WIB.

```
#include <DS3232RTC.h>
```

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

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

```
#include <LiquidCrystal_I2C.h>
```

```
LiquidCrystal_I2C lcd(0x27, 16, 2);
```

```
SoftwareSerial mySerial(0, 1); // RX, TX
```

```
struct Waktu
```

```
{
```

```
  byte jam;
```

```
  byte menit;
```

```
};
```

```
struct Tabelacara
```

```
{
```

```
  byte aktif;
```

```
  Waktu waktu;
```

```
  byte hariAktif;
```

```
  byte mingguAktif;
```

```
  byte kegiatan;
```

```
};
```

```
enum _kegiatan
```

```
{
```

```
    literasiquran,
```

```
    literasibuku,
```

```
    upacara,
```

```
    senam,
```

```
    salatdhuha,
```

```
    salatdzuhur,
```

```
    bersih,
```

```
    ekskul,
```

```
    mengaji,
```

```
    salatjumat,
```

```
};
```

```
volatile bool interupsiDetik;
```

```
byte indexacara;
```

```
byte detikSebelumnya = 60;
```

```
#define hariAktifSenin 1<<6
```

```

#define hariAktifRabu 1<<4

StringnamaHari[]={ "Minggu", "Senin", "Selasa", "Rabu", "Kamis", "Jum'at", "Sabtu;

#define _hariAktif(Sen,Se1,Rab,Kam,Jum,Sab,Mgu)

(Mgu<<7)|(Sen<<6)|(Se1<<5)|(Rab<<4)|(Kam<<3)|(Jum<<2)|(Sab<<1)

#define _mingguAktif(Mgu1,Mgu2,Mgu3,Mgu4,Mgu5)

(Mgu1<<7)|(Mgu2<<6)|(Mgu3<<5)|(Mgu4<<4)|(Mgu5<<3)

#define _waktu(Jam, Menit) {Jam, Menit}

#define Aktif 1

#define TidakAktif 0

Tabelacara jadwalBelajar[35];

Waktu waktu;

tmElements_t tm;

void setup() {

    byte i = 0;

    //-----

    jadwalBelajar[i++]={Aktif,_waktu(12,0),_hariAktif(0,0,0,0,1,0,0),
_mingguAktif(1,1,1,1,1),salatjumat};

```




Gambar 4.9 Uji Coba Kegiatan Shalat Jumat
Sumber : Penulis

4.3.8 Pengingat Kegiatan Bersih-bersih

Pada kondisi ini pengingat otomatis berbunyi ketika masuk waktu nya untuk siswa/siswi SD Panca Budi Medan melaksanakan kegiatan bersih-bersih sebelum pulang sekolah. Namun ada perbedaan waktu diantara hari-hari tersebut dikarenakan ada jadwal pulang sekolah yang berbeda pada hari nya senin-kamis jam 13:13 hari jumat jam 13:00, dan sabtu jam 10.00. Dimisalkan tahun 2020, tanggal 15, bulan 2, jam 10 : menit 00 sesuai dengan inputan program dibawah yaitu RTC memanggil waktu yang diinputkan pada program sehingga bel berbunyi sesuai dengan file audio yang diinputkan pada jam 10.00 WIB.

```
#include <DS3232RTC.h>
```

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

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

```
#include <LiquidCrystal_I2C.h>
```

```
LiquidCrystal_I2C lcd(0x27, 16, 2);  
  
SoftwareSerial mySerial(0, 1); // RX, TX  
  
struct Waktu  
{  
    byte jam;  
    byte menit;  
};  
  
struct Tabelacara  
{  
    byte aktif;  
    Waktu waktu;  
    byte hariAktif;  
    byte mingguAktif;  
    byte kegiatan;  
};  
  
enum _kegiatan  
{  
    literasiquran,  
    literasibuku,  
    upacara,
```

```

senam,

salatdhuha,

salatdzuhur,

bersih,

ekskul,

mengaji,

salatjumat,

};

volatile bool interupsiDetik;

byte indexacara;

byte detikSebelumnya = 60;

#define hariAktifSenin 1<<6

#define hariAktifRabu 1<<4

StringnamaHari[]={ "Minggu", "Senin", "Selasa", "Rabu", "Kamis", "Jum'at", "Sabtu;

#define _hariAktif(Sen,Se1,Rab,Kam,Jum,Sab,Mgu)

(Mgu<<7)|(Sen<<6)|(Se1<<5)|(Rab<<4)|(Kam<<3)|(Jum<<2)|(Sab<<1)

#define _mingguAktif(Mgu1,Mgu2,Mgu3,Mgu4,Mgu5)

(Mgu1<<7)|(Mgu2<<6)|(Mgu3<<5)|(Mgu4<<4)|(Mgu5<<3)

```

```

#define _waktu(Jam, Menit) {Jam, Menit}

#define Aktif 1

#define TidakAktif 0

Tabelacara jadwalBelajar[35];

Waktu waktu;

tmElements_t tm;

void setup() {
    byte i = 0;

    //-----

    jadwalBelajar[i++]={Aktif,_waktu(13,13), _hariAktif(1,1,1,1,0,0,0),
_mingguAktif(1,1,1,1,1), bersih};

    jadwalBelajar[i++]={Aktif,_waktu(13,0), _hariAktif(0,0,0,0,1,0,0),
_mingguAktif(1,1,1,1,1), bersih};

    jadwalBelajar[i++]={Aktif,_waktu(10,0), _hariAktif(0,0,0,0,0,1,0),
_mingguAktif(1,1,1,1,1), bersih};

```



Gambar 4.10 Uji Coba Kegiatan Bersih-bersih
Sumber : Penulis

4.3.9 Pengingat Kegiatan Ekskul

Pada kondisi ini pengingat otomatis berbunyi ketika masuk waktu nya untuk siswa/siswi SD Panca Budi Medan melaksanakan kegiatan ekstrakurikuler sesudah pulang sekolah. Namun ada perbedaan waktu diantara hari-hari tersebut dikarenakan ada jadwal pulang sekolah yang berbeda pada hari nya senin-kamis jam 13:30 hari jumat jam 13:15, dan sabtu jam 10.30. Dimisalkan tahun 2020, tanggal 15, bulan 2, jam 10 : menit 30 sesuai dengan inputan program dibawah yaitu RTC memanggil waktu yang diinputkan pada program sehingga bel berbunyi sesuai dengan file audio yang diinputkan pada jam 10.30 WIB.

```
#include <DS3232RTC.h>
```

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

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

```
#include <LiquidCrystal_I2C.h>
```

```
LiquidCrystal_I2C lcd(0x27, 16, 2);  
  
SoftwareSerial mySerial(0, 1); // RX, TX  
  
struct Waktu  
{  
    byte jam;  
    byte menit;  
};  
  
struct Tabelacara  
{  
    byte aktif;  
    Waktu waktu;  
    byte hariAktif;  
    byte mingguAktif;  
    byte kegiatan;  
};  
  
enum _kegiatan  
{  
    literasiquran,  
    literasibuku,  
    upacara,
```

```

senam,

salatdhuha,

salatdzuhur,

bersih,

ekskul,

mengaji,

salatjumat,

};

volatile bool interupsiDetik;

byte indexacara;

byte detikSebelumnya = 60;

#define hariAktifSenin 1<<6

#define hariAktifRabu 1<<4

StringnamaHari[]={ "Minggu", "Senin", "Selasa", "Rabu", "Kamis", "Jum'at", "Sabtu;

#define _hariAktif(Sen,Se1,Rab,Kam,Jum,Sab,Mgu)

(Mgu<<7)|(Sen<<6)|(Se1<<5)|(Rab<<4)|(Kam<<3)|(Jum<<2)|(Sab<<1)

#define _mingguAktif(Mgu1,Mgu2,Mgu3,Mgu4,Mgu5)

(Mgu1<<7)|(Mgu2<<6)|(Mgu3<<5)|(Mgu4<<4)|(Mgu5<<3)

```

```

#define _waktu(Jam, Menit) {Jam, Menit}

#define Aktif 1

#define TidakAktif 0

Tabelacara jadwalBelajar[35];

Waktu waktu;

tmElements_t tm;

void setup() {
    byte i = 0;

    //-----

    jadwalBelajar[i++]={Aktif,_waktu(13,30)_hariAktif(1,1,1,1,0,0,0),
_mingguAktif(1,1,1,1,1), ekskul};

    jadwalBelajar[i++]={Aktif,_waktu(13,15),_hariAktif(0,0,0,0,1,0,0),
_mingguAktif(1,1,1,1,1), ekskul};

    jadwalBelajar[i++]={Aktif,_waktu(10,30),_hariAktif(0,0,0,0,0,1,0),
_mingguAktif(1,1,1,1,1), ekskul};

```




Gambar 4.11 Uji Coba Kegiatan Ekskul
Sumber ; Penulis

4.3.10 Pengingat Kegiatan Mengaji

Pada kondisi ini pengingat otomatis berbunyi ketika masuk waktu nya untuk siswa/siswi SD Panca Budi Medan melaksanakan kegiatan Mengaji sesudah pulang kegiatan ekskul yang dilaksanakan setiap hari senin-kamis jam 14:45. Dimisalkan tahun 2020, tanggal 10, bulan 2, jam 14 : menit 45 sesuai dengan inputan program dibawah yaitu RTC memanggil waktu yang diinputkan pada program sehingga bel berbunyi sesuai dengan file audio yang diinputkan pada jam 14.45 WIB.

```
#include <DS3232RTC.h>
```

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

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

```
#include <LiquidCrystal_I2C.h>
```

```
LiquidCrystal_I2C lcd(0x27, 16, 2);
```

```
SoftwareSerial mySerial(0, 1); // RX, TX
```

```
struct Waktu
```

```
{  
    byte jam;  
    byte menit;  
};
```

```
struct Tabelacara
```

```
{  
    byte aktif;  
    Waktu waktu;  
    byte hariAktif;  
    byte mingguAktif;  
    byte kegiatan;  
};
```

```
enum _kegiatan
```

```
{  
    literasiquran,  
    literasibuku,  
    upacara,  
    senam,  
    salatdhuha,
```

```

    salatdzuhur,

    bersih,

    ekskul,

    mengaji,

    salatjumat,

};

volatile bool interupsiDetik;

byte indexacara;

byte detikSebelumnya = 60;

#define hariAktifSenin 1<<6

#define hariAktifRabu 1<<4

StringnamaHari[]={ "Minggu", "Senin", "Selasa", "Rabu", "Kamis", "Jum'at", "Sabtu;

#define_hariAktif(Sen,Se1,Rab,Kam,Jum,Sab,Mgu)

(Mgu<<7)|(Sen<<6)|(Se1<<5)|(Rab<<4)|(Kam<<3)|(Jum<<2)|(Sab<<1)

#define_mingguAktif(Mgu1,Mgu2,Mgu3,Mgu4,Mgu5)

(Mgu1<<7)|(Mgu2<<6)|(Mgu3<<5)|(Mgu4<<4)|(Mgu5<<3)

#define _waktu(Jam, Menit) {Jam, Menit}

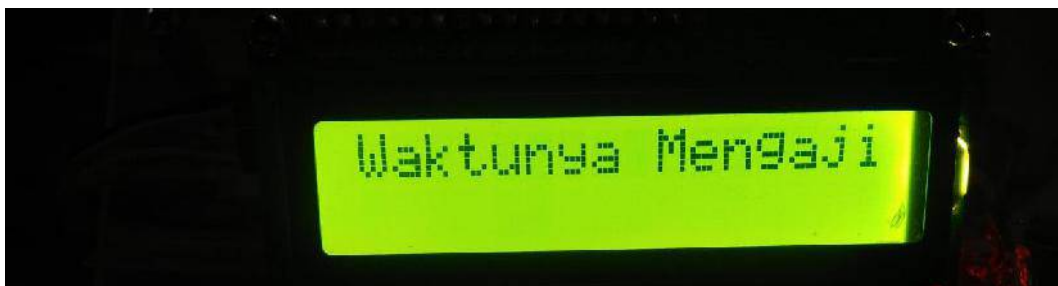
#define Aktif 1

```

```
#define TidakAktif 0

Tabelacara jadwalBelajar[35];
Waktu waktu;
tmElements_t tm;

void setup() {
    byte i = 0;
    //-----
    jadwalBelajar[i++]={Aktif,_waktu(14,45),_hariAktif(1,1,1,1,0,0),
_mingguAktif(1,1,1,1,1), mengaji};
}
```



Gambar 4.12 Uji Coba Kegiatan Mengaji
Sumber : Penulis

Hari	Waktu	Keterangan	<i>Output</i> Suara	Lama Alarm	Sesuai	Tidak Sesuai
Senin	7.30	Literasi Qur'an	Bunyi Suara Tanda Literasi Qur'an	Sesuai dengan durasi file MP3	YA	-
Senin	9.45	Shalat Dhuha	Bunyi Suara Shalat Dhuha	Sesuai dengan durasi file MP3	YA	-
Senin	12.15	Shalat Dzuhur	Bunyi Suara Shalat Dzuhur	Sesuai dengan durasi file MP3	YA	-
Senin	13.13	Budaya Bersih-bersih	Bunyi Suara Budaya Bersih-bersih	Sesuai dengan durasi file MP3	YA	-
Senin	13.30	Kegiatan Ekskul	Bunyi Suara Kegiatan Ekskul	Sesuai dengan durasi file MP3	YA	-
Senin	14.45	Kelas Mengaji	Bunyi Suara Kelas Mengaji	Sesuai dengan durasi file MP3	YA	-

Tabel 4.1 Hasil Pengujian Sistem Secara Keseluruhan Jadwal Senin Minggu Pertama

BAB 5

PENUTUP

5.1 Kesimpulan

Dari hasil rancangan dan pengujian diperoleh kesimpulan sebagai berikut :

1. Dengan adanya alat pengingat sekolah otomatis ini, maka guru yang bertugas membunyikan/mengaktifkan pengumuman sekolah tidak lagi memberikan pengumuman secara manual.
2. Kemudian file suara yang telah disimpan kedalam Micro SD Card dapat dimainkan dengan baik dengan jenis format MP3 melalui modul DFMiniPlayer.
3. Pengumuman dan pengingat berbunyi otomatis di SD Panca Budi Medan pada saat 10 kondisi dengan durasi masing-masing 1 menit yaitu: Upacara, Literasiquran, Literasibuku, Senam, Shlat Dhuha, Shalat Dzuhur, Shalat Jumat, Bersih-bersih, Ekskul, Mengaji.

5.2 Saran

Dari hasil Pengujian ini masih terdapat kekurangan dan dimungkinkan untuk pengembangan lebih lanjut. Oleh karenanya penulis merasa perlu untuk memberi saran sebagai berikut :

1. Output suara yang terdengar kecil, sehingga dapat menggunakan *speaker* yang lebih bagus dan canggih sehingga suara yang keluar lebih besar dan jernih.
2. Tampilan interface pengingat otomatis sekolah yang dirasa kurang lebar, karena hanya memakai lcd 16x2 dapat diperbarui dengan menggunakan layar oled yang hasil tampilannya akan lebih lebar dan lebih halus dan tajam sebagai interface atau tampilan jam digital pengingat otomatis.
3. Jika listrik padam maka dapat digantikan dengan solar panel yang terhubung dengan *speaker* dan mikrokontroler arduino, untuk itu SD Panca Budi harus menyediakan perangkat solar panel.

DAFTAR PUSTAKA

- Assegaf, FA. (2017). Sistem Penyiraman Tanaman Otomatis dan Monitoring Kelembaban Tanah Jarak Jauh Menggunakan ATMEGA8535 Berbasis Web Server . Univeristas Muhammadiyah Malang
- Batu Bara, YM. (2017). Rancang Bangun Alat Kendali Lampu Menggunakan Sensor Tepuk Berbasisi Arduino. Universitas Sumatera Utara. Medan
- Daulay, RS. (2018). Rancang Bangun Bel Sekolah Otomatis Berbasis Arduino di SMK PANCA BUDI 1 MEDAN. Univeristas Pembangunan Panca Budi Medan
- Damayanti, UC. (2017). Rancang Bangun Sistem Pengunci Loker Otomatis dengan Kendali Akses Menggunakan RFID. Politeknik Negeri Sriwijaya.Palembang
- Endaryano, PJ, Harianto, Wibowo, MC. (2014). Rancang Bangun Sistem Pembayaran Mandiri Pada Wahana Permainan. *Journal of Countrol and Network Systems.3 (1).70-77.* Diakses dari <https://jurnal.dinamika.ac.id/index.php/jcone/article/view/301/258>
- Fachri, barany, agus perdana windarto, and ikhsan parinduri. "penerapan backpropagation dan analisis sensitivitas pada prediksi indikator terpenting perusahaan listrik." jepin (jurnal edukasi dan penelitian informatika) 5.2 (2019): 202-208.
- Fachri, b., windarto, a. P., & parinduri, i. (2019). Penerapan backpropagation dan analisis sensitivitas pada prediksi indikator terpenting perusahaan listrik. Jepin (jurnal edukasi dan penelitian informatika), 5(2), 202-208.
- Fachri, barany; windarto, agus perdana; parinduri, ikhsan. Penerapan backpropagation dan analisis sensitivitas pada prediksi indikator terpenting perusahaan listrik. Jepin (jurnal edukasi dan penelitian informatika), 2019, 5.2: 202-208
- Hamdi, nurul. "model penyiraman otomatis pada tanaman cabe rawit berbasis programmable logic control." jurnal ilmiah core it: community research information technology 7.2 (2019).
- Hartono, R. (2018). Perancangan Sistem data logger Temperature Baterai Berbasis Arduino DueMilanove. Universitas Jember.

- Permana, aminuddin indra. "kombinasi algoritma kriptografi one time pad dengan generate random keys dan vigenere cipher dengan kunci em2b." (2019).
- Putra, randi rian. "sistem informasi web pariwisata hutan mangrove di kelurahan belawan sicanang kecamatan medan belawan sebagai media promosi." jurnal ilmiah core it: community research information technology 7.2 (2019).
- Putra, randi rian, et al. "decision support system in selecting additional employees using multi-factor evaluation process method." (2019).
- Putra, randi rian. "implementasi metode backpropagation jaringan saraf tiruan dalam memprediksi pola pengunjung terhadap transaksi." jurti (jurnal teknologi informasi) 3.1 (2019): 16-20
- Putra, AA, Haris MY. (2017). Perancangan Sistem Kontrol Lampu Berbasis Mikrokontroler Arduino Uno R3 Denga Sensor Suara. Univeristas Muhammadiyah Makassar.
- Santoso, Nurmalina S. (2017). Perencanaa dan Pengembangan Aplikasi Absensi Mahasiswa Menggunakan Smart Card Guna Pengembangan Kampus Cerdas (Studi Kasus Politeknik Negeri Tanah Laut). Jurnal Integrasi. 9 (1).2548-9828. Diakses dari <https://jurnal.polibatam.ac.id/index.php/JI/article/view/288>
- Saputra, muhammad juanda, and nurul hamdi. "rancang bangun aplikasi sejarah kebudayaan aceh berbasis android studi kasus dinas kebudayaan dan pariwisata aceh." journal of informatics and computer science 5.2 (2019): 147-157
- Sidik, a. P., efendi, s., & suherman, s. (2019, june). Improving one-time pad algorithm on shamir's three-pass protocol scheme by using rsa and elgamal algorithms. In journal of physics: conference series (vol. 1235, no. 1, p. 012007). Iop publishing.
- Sitepu, n. B., zarlis, m., efendi, s., & dhany, h. W. (2019, august). Analysis of decision tree and smooth support vector machine methods on data mining. In journal of physics: conference series (vol. 1255, no. 1, p. 012067). Iop publishing
- Tasril, v., wijaya, r. F., & widya, r. (2019). Aplikasi pintar belajar bimbingan dan konseling untuk siswa sma berbasis macromedia flash. Jurnal informasi komputer logika, 1(3)
- Univeristas Telkom.1 (1). 847. Diakses dari https://openlibrary.telkomuniversity.ac.id/pustaka/files/100504/jurnal_eproc/implementasi-sistem-pemanggil-antrian-dengan-tampilan-seven-segment-berbasis-mikrokontroler-pada-pt-pln-sukoharjo.pdf

Wijayanto, D, Hadiyoso, S, Hariyani YS. (2015). Implementasi Sistem Pemanggil Antrian dengan Tampilan *Seven Segment* Berbasisi Mikrokontroler Pada PT.PLN Sukoharjo . Jurnal Teknik Telekomunikasi Fakultas Ilmu Terapan