

**ALAT PENUNJUK ARAH KIBLAT DAN INFORMASI
WAKTU SHOLAT BAGI PENDERITA TUNA NETRA**

SKRIPSI

**Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Teknik**



Disusun oleh:

MUKHAMAD DENICA ELSA

NIM. 0210630089

DEPARTEMEN PENDIDIKAN NASIONAL

UNIVERSITAS BRAWIJAYA

FAKULTAS TEKNIK

MALANG

2007

repository.ub.ac.id

**ALAT PENUNJUK ARAH KIBLAT DAN INFORMASI
WAKTU SHOLAT BAGI PENDERITA TUNA NETRA**

SKRIPSI

**Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Teknik**



Disusun oleh:

MUKHAMAD DENICA ELSA

NIM. 0210630089

Telah diperiksa dan disetujui oleh
DOSEN PEMBIMBING:

Ir. M. Julius St., MS.
NIP. 131 124 655

Ir. Nurussa'adah
NIP. 131 994 339



repository.ub.ac.id

ALAT PENUNJUK ARAH KIBLAT DAN INFORMASI WAKTU SHOLAT BAGI PENDERITA TUNA NETRA

Disusun oleh:

MUKHAMAD DENICA ELSA

NIM. 0210630089

Skripsi ini telah diuji dan dinyatakan lulus pada
Tanggal 14 Juni 2007

DOSEN PENGUJI

Ir. Bambang Siswojo
NIP. 131 759 588

Arief Andy Subroto, ST., M.Kom.
NIP. 132 231 567

M. Rif'an, ST., MT.
NIP. 132 283 659

Ir. Ponco Siwindarto, MS.
NIP. 131 837 966

Mengetahui
Ketua Jurusan Teknik Elektro

Ir. Heru Nurwarsito, M.Kom
NIP. 131 879 033

PENGANTAR

Segala puji hanya bagi Allah SWT yang telah melimpahkan rahmat-Nya kepada penulis sehingga penyusunan skripsi ini dapat terselesaikan. Hanya karena pertolongan-Nya semata penulis mampu melewati segala kendala yang ada selama penyusunan skripsi ini. Semoga sholawat dan salam senantiasa tercurahkan kepada Nabi Muhammad SAW, beserta keluarga, sahabat dan para pengikut beliau sampai hari akhir.

Skripsi berjudul “Alat Penunjuk Arah Kiblat dan Informasi Waktu Sholat Bagi Penderita Tuna Netra” ini disusun sebagai salah satu syarat untuk mendapatkan gelar Sarjana Teknik di Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya Malang.

Penulis menyadari bahwa penyusunan skripsi ini tidak terlepas dari bantuan berbagai pihak. Oleh karena itu, dengan segala kerendahan hati penulis menyampaikan terima kasih kepada:

- Ayah dan Ibunda-ku yang aku cintai karena-Nya, jazaakumullah khairan katsir. Semoga Allah SWT membalas kasih sayang dan kebaikan kalian selama ini dengan kebaikan yang jauh lebih banyak,
- Ir. Heru Nurwarsito, M.Kom selaku Ketua Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya Malang,
- Rudy Yuwono, ST., MS selaku Sekretaris Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya Malang,
- Ir. Ponco Siwindarto, MS selaku KKDK Elektronika Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya Malang,
- Ir. M. Julius St., MS selaku dosen pembimbing I yang telah banyak memberikan ide, bimbingan, motivasi, pengarahan serta saran dalam penyusunan skripsi ini,
- Ir. Nurussa’adah selaku dosen pembimbing II yang telah banyak memberikan ide, bimbingan, motivasi, pengarahan serta saran dalam penyusunan skripsi ini,
- Adikku yang menyayangi aku dan semua keluargaku, jazakumullah khairan katsir atas dukungannya selama ini,
- *All Elkamania Crews*, Mas Mul selaku laboran, terima kasih atas dukungan dan kesabarannya. Jazakumullah khairan katsir, saya mendapatkan banyak ilmu dan pengalaman berharga di Laboratorium Elektronika ini,

- *All Kersent 27 Crews* (Aditya Budiman, Imam Sumarsono, Ahmad Zulfi Ardian, Andhik Wahyu Widodo, Erick Elsafan S, Boby Marjaya H), terima kasih atas dukungan dan kebersamaan yang telah kalian berikan selama saya di Malang ini,
- Teman-teman seperjuangan di Teknik Elektro Fakultas Teknik Universitas Brawijaya Malang yang tidak memungkinkan untuk kutulis semua namanya disini, terima kasih atas bantuan dan dukungannya.

Penyusun menyadari bahwa yang tersusun dalam skripsi ini masih banyak kekurangan-kekurangan dan jauh dari sempurna, hal ini tidak lain karena keterbatasan materi dan pengetahuan yang dimiliki penulis. Karena itu kritik dan saran yang membangun sangat diharapkan demi kesempurnaan penyusunan skripsi ini.

Akhir kata, penulis mengharapkan semoga yang terdapat dalam skripsi ini, dapat bermanfaat bagi rekan-rekan mahasiswa khususnya dan bagi seluruh pembaca pada umumnya.



Malang, Mei 2007

Penulis



DAFTAR ISI

PENGANTAR	i
DAFTAR ISI	iii
DAFTAR TABEL	vii
DAFTAR GAMBAR	viii
ABSTRAK	x
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Ruang Lingkup	2
1.4 Tujuan.....	2
1.5 Sistematika Penulisan.....	3
BAB II TINJAUAN PUSTAKA	4
2.1 Dasar Penentuan Waktu Sholat	4
2.2 Dasar Penentuan Arah Kiblat	7
2.3 Sensor Magnet CMPS03	8
2.4 RTC (<i>Real Time Clock</i>).....	10
2.5 IC Suara ISD2500	12
2.6 EEPROM (<i>Electrically Erasable Programmable Memory</i>)	16
2.7 Mikrokontroler ATMEGA8535	18
2.7.1 Struktur dan Operasi Port	19
2.7.2 Sistem Interrupt.....	20
2.7.3 Antarmuka Serial Dua Kabel (<i>Two-Wire Serial Interface</i>)	21
2.8 Tombol Tekan (Push Button).....	21
BAB III METODOLOGI PENELITIAN	22
3.1 Studi Literatur	22
3.2 Perancangan Alat.....	22
3.3 Perealisasiian Alat	23
3.4 Pengujian Alat	23

3.5	Penarikan Kesimpulan.....	24
BAB IV PERANCANGAN DAN PEMBUATAN ALAT.....		25
4.1	Spesifikasi Alat	25
4.2	Diagram Blok Rangkaian	25
4.3	Perancangan Perangkat Keras	27
4.3.1	Sensor Magnet CMPS03.....	27
4.3.2	Real Time Clock (RTC).....	28
4.3.3	EEPROM AT24C64	28
4.3.4	Rangkaian Mikrokontroler ATMEGA8535.....	29
4.3.4.1	Rangkaian Minimum Sistem Mikrokontroler ATMEGA8535	29
4.3.4.2	Rangkaian Reset.....	31
4.3.5	IC Suara ISD2590.....	33
4.3.6	Penguat Audio.....	36
4.4	Perancangan Perangkat Lunak	37
BAB V PENGUJIAN DAN ANALISIS		43
5.1	Pengujian Sensor Magnet CMPS03	43
5.1.1	Tujuan	43
5.1.2	Peralatan.....	43
5.1.3	Prosedur Pengujian	44
5.1.4	Hasil.....	45
5.1.5	Analisis dan Kesimpulan	45
5.2	Pengujian RTC DS1302.....	45
5.2.1	Tujuan	45
5.2.2	Peralatan.....	45
5.2.3	Prosedur Pengujian	46
5.2.4	Hasil.....	46
5.2.5	Analisis dan Kesimpulan	49
5.3	Pengujian EEPROM AT24C64.....	49
5.3.1	Tujuan	49
5.3.2	Peralatan.....	49
5.3.3	Prosedur Pengujian	50
5.3.4	Hasil.....	51
5.3.5	Analisis dan Kesimpulan	52

5.4	Pengujian IC Suara ISD2590	52
5.4.1	Tujuan	52
5.4.2	Peralatan	52
5.4.3	Prosedur Pengujian	53
5.4.4	Hasil	53
5.4.5	Analisis dan Kesimpulan	54
5.5	Pengujian Rangkaian Penguat Audio	55
5.5.1	Tujuan	55
5.5.2	Peralatan	55
5.5.3	Prosedur Pengujian	55
5.5.4	Hasil	56
5.5.5	Analisis dan Kesimpulan	57
5.6	Pengujian Perangkat Lunak Untuk Mengetahui Waktu Sholat	57
5.6.1	Tujuan	57
5.6.2	Peralatan	57
5.6.3	Prosedur Pengujian	57
5.6.4	Hasil	58
5.6.5	Analisis dan Kesimpulan	59
5.7	Pengujian Perangkat Lunak Untuk Perhitungan Arah Kiblat	59
5.7.1	Tujuan	59
5.7.2	Peralatan	59
5.7.3	Prosedur Pengujian	59
5.7.4	Hasil	61
5.7.5	Analisis dan Kesimpulan	61
5.8	Pengujian Sistem Secara Keseluruhan	61
5.8.1	Tujuan	61
5.8.2	Peralatan	61
5.8.3	Prosedur Pengujian	62
5.8.4	Hasil	62
5.8.5	Analisis dan Kesimpulan	63
BAB IV PENUTUP		64
6.1	Kesimpulan	64
6.2	Saran	64

DAFTAR PUSTAKA..... 65

LAMPIRAN 66



DAFTAR TABEL

Tabel 2.1 Keterangan kaki-kaki pin sensor magnet CMPS03	9
Tabel 2.2 Register sensor magnet CMPS03.....	10
Tabel 2.3 Fungsi dari masing-masing pin.....	11
Tabel 2.4 Alamat register RTC	12
Tabel 2.5 Contoh Susunan A0..A2.....	17
Tabel 2.6 Fungsi Tambahan Port B.....	19
Tabel 2.7 Fungsi Tambahan Port C.....	19
Tabel 2.8 Fungsi Tambahan Port D	20
Tabel 2.9 Alamat Vektor <i>Interrupt</i> ATMEGA8535	20
Tabel 4.1 Pemilihan alamat untuk merekam suara lokasi ke dalam ISD2590.....	34
Tabel 4.2 Pemilihan alamat untuk merekam suara angka ke dalam ISD2590.....	35
Tabel 4.3 Pemilihan alamat untuk merekam suara pilih-derajat ke dalam ISD2590....	35
Tabel 4.4 Pemilihan alamat untuk merekam suara arah ke dalam ISD2590	36
Tabel 4.5 Pemilihan alamat untuk merekam suara adzan ke dalam ISD2590	36
Tabel 4.6 Pemilihan alamat untuk merekam suara pukul-bulan ke dalam ISD2590	36
Tabel 5.1 Hasil Pengujian RTC DS1302 Untuk Data Detik.....	47
Tabel 5.2 Hasil Pengujian RTC DS1302 Untuk Data Menit	48
Tabel 5.3 Hasil Pengujian RTC DS1302 Untuk Data Jam	48
Tabel 5.4 Hasil Pengujian RTC DS1302 Untuk Data Hari.....	49
Tabel 5.5 Hasil Pengujian RTC DS1302 Untuk Data Bulan	49
Tabel 5.6 Hasil Pengujian RTC DS1302 Untuk Data Tahun.....	49
Tabel 5.7 Hasil pengujian pembacaan data EEPROM AT24C64.....	52
Tabel 5.8 Hasil Pengujian <i>Record</i> dan <i>Play Back</i> ISD2590	53
Tabel 5.9 Data Hasil Pengujian Rangkaian Penguat Audio Dengan Nilai Frekuensi Berbeda	56
Tabel 5.10 Data Hasil Pengujian Rangkaian Penguat Audio Dengan Nilai Frekuensi Tetap	56
Tabel 5.11 Data Hasil Pengujian Perangkat Lunak Untuk Mengetahui Waktu Sholat .	59
Tabel 5.12 Data Hasil Pengujian Perangkat Lunak Perhitungan Arah Kiblat	61

DAFTAR GAMBAR

Gambar 2.1	Waktu-waktu shalat.....	5
Gambar 2.2	Segitiga Bola	5
Gambar 2.3	Sudut acuan untuk menentukan arah kiblat.....	8
Gambar 2.4	Sensor magnet CMPS03.....	9
Gambar 2.5	Konfigurasi pin DS1302.....	11
Gambar 2.6	Blok diagram IC DS1302	12
Gambar 2.7	Blok Diagram Seri ISD 2500	13
Gambar 2.8	Konfigurasi Pin ISD2500	14
Gambar 2.9	(a) Susunan kaki IC AT24Cxx (b) Keluarga I2C Serial EEPROM IC AT24Cxx.....	17
Gambar 2.10	Konfigurasi Pin ATMEGA8535	18
Gambar 4.1	Blok diagram penunjuk arah kiblat dan informasi waktu shalat bagi penderit tuna netra	26
Gambar 4.2	Rangkaian Sensor Magnet.....	27
Gambar 4.3	Rangkaian <i>Real Time Clock</i> DS1302	28
Gambar 4.4	Rangkaian EEPROM AT24C64.....	29
Gambar 4.5	Rangkaian Mikrokontroler ATMEGA8535	30
Gambar 4.6	Timing Diagram Sinyal Reset	31
Gambar 4.7	Rangkaian Reset	31
Gambar 4.8	Rangkaian IC Suara ISD2590	33
Gambar 4.9	Rangkaian penguat audio	36
Gambar 4.10	Diagram alir program utama	39
Gambar 4.11	Diagram alir program interupsi eksternal lokasi	40
Gambar 4.12	Diagram alir program interupsi eksternal kiblat.....	41
Gambar 4.13	Diagram alir program arah gerak	42
Gambar 5.1	Diagram Blok Pengujian Sensor Magnet CPMS03	44
Gambar 5.2	Diagram Alir Pengujian Sensor Magnet CMPS03.....	44
Gambar 5.3	Hasil Pengujian Sensor Magnet CMPS03.....	45
Gambar 5.4	Diagram Blok Pengujian RTC DS1302	46

Gambar 5.5	Diagram Alir Program Pengujian RTC DS1302.....	46
Gambar 5.6	Diagram Blok Penulisan Data ke EEPROM AT24C64	50
Gambar 5.7	Diagram Blok Pembacaan Data dari EEPROM AT24C64	50
Gambar 5.8	Diagram Alir Pembacaan Data dari EEPROM AT24C64	51
Gambar 5.9	Hasil pengujian penulisan data ke EEPROM AT24C64.....	51
Gambar 5.10	Rangkaian Pengujian Rangkaian Penguat Audio.....	55
Gambar 5.11	Tampilan <i>Oscilloscop</i> Pengujian Penguat Audio.....	56
Gambar 5.12	Diagram Blok Pengujian Perangkat Lunak Untuk Mengetahui Waktu Sholat	58
Gambar 5.13	Diagram Alir Pengujian Perangkat Lunak Untuk Mengetahui Waktu Sholat	58
Gambar 5.14	Diagram Blok Pengujian Perangkat Lunak Perhitungan Arah Kiblat..	60
Gambar 5.15	Diagram Alir Pengujian Perangkat Lunak Perhitungan Arah Kiblat...	60
Gambar 5.16	Diagram Blok Pengujian Sistem Keseluruhan	62



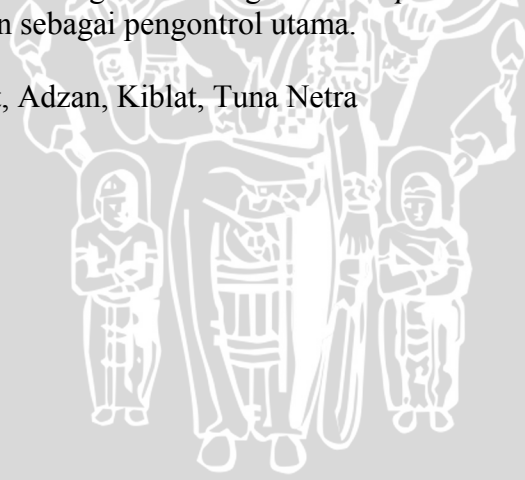
ABSTRAK

Mukhamad Denica Elsa, Jurusan Teknik Elektro, Fakultas Teknik, Universitas Brawijaya, 2007, *Alat Penunjuk Arah Kiblat dan Informasi Waktu Sholat Bagi Penderita Tuna Netra*, Dosen Pembimbing: Ir. Julius St., MS dan Ir. Nurussa'adah.

Kewajiban ibadah sholat lima waktu setiap harinya dan dengan aturan-aturan yang telah ditetapkan dalam Al-Qur'an khususnya mengenai arah kiblat ke Masjidil Haram di Mekah, menyebabkan umat Islam untuk selalu memperhatikan waktu dan aturan-aturan yang telah ditetapkan tersebut. Tidak terkecuali dengan umat Islam yang menderita tuna netra.

Dengan memanfaatkan sensor magnet CMPS03 dan RTC DS1302, dapat dibuat alat untuk membantu umat Islam khususnya bagi penderita tuna netra untuk dapat menunjukkan waktu sholat yang tiba dan dilengkapi dengan penunjuk arah kiblat dengan keluaran berupa suara melalui *speaker*. Suara adzan dengan dua mode yaitu mode adzan ketika waktu sholat Subuh dan mode adzan ketika waktu sholat Duhur, Ashar, Maghrib, dan Isya yang digunakan sebagai penanda datangnya waktu sholat serta panduan pergerakan arah digunakan sebagai panduan arah kiblat. Perekaman dan pemutaran ulang suara dilakukan dengan memanfaatkan IC suara ISD2590 yang kemudian dikuatkan menggunakan penguat audio yang memanfaatkan LM386. Data-data mengenai waktu sholat dan arah kiblat disimpan dalam EEPROM AT24C64. Sebagai tombol masukan digunakan tiga buah *push button*. Mikrokontroler ATMEGA8535 digunakan sebagai pengontrol utama.

Kata kunci: Waktu Sholat, Adzan, Kiblat, Tuna Netra



BAB I

PENDAHULUAN

1.1 Latar Belakang

Islam merupakan salah satu agama samawi yang dianut oleh umat manusia. Salah satu kewajiban yang harus dijalankan oleh umat Islam adalah ibadah sholat yang dilaksanakan sebanyak lima kali setiap harinya. Dan untuk menentukan waktu pelaksanaannya digunakan matahari sebagai acuannya. Dengan acuan tersebut maka dapat disusun suatu perumusan tentang jadwal waktu sholat di suatu daerah dalam jangka waktu tertentu. Dalam Al-Qur'an Allah berfirman "Sesungguhnya sholat itu diwajibkan atas orang-orang yang beriman menurut waktu-waktu yang tertentu" (Q.S. An-Nisa : 103)

Tata cara pelaksanaan sholat telah ditetapkan dan salah satunya adalah mengenai arah kiblat ketika melaksanakan sholat. Dalam Al-Quran Allah berfirman "Dan dari mana saja kamu keluar (datang), maka palingkanlah wajahmu ke arah Masjidil Haram, sesungguhnya ketentuan itu benar-benar sesuatu yang hak dari Tuhanmu. Dan Allah sekali-kali tidak lengah dari apa yang kamu kerjakan. Dan dari mana saja kamu (keluar), maka palingkanlah wajahmu ke arah Masjidil Haram. Dan dimana saja kamu (sekalian) berada, maka palingkanlah wajahmu ke arahnya, agar tidak ada hujjah bagi manusia atas kamu, kecuali orang-orang yang zalim diantara mereka. Maka janganlah kamu takut kepada mereka dan takutlah kepada-Ku (saja). Dan agar Ku-sempurnakan nikmat-Ku atasmu, dan supaya kamu mendapat petunjuk." (Q.S. Al-Baqoroh : 149-150), sehingga semua orang yang melaksanakan sholat diwajibkan untuk menghadap Ka'bah di Masjidil Haram, Mekah ketika sholatnya. Tidak terkecuali bagi penderita tuna netra, sehingga diperlukan suatu perhatian yang cukup penting untuk masalah ini.

Skripsi dengan judul "Perencanaan dan pembuatan alat penunjuk waktu sholat dengan menggunakan mikrokontroler AT89C51" karya dari Wahid Arief Kurniawan, merupakan perancangan sebuah alat yang dapat menunjukkan waktu sholat berdasarkan perumusan dengan acuan posisi matahari. Alat tersebut dibuat dengan menggunakan mikrokontroler AT89C51, dengan tampilan LCD, menggunakan IC suara sebagai penanda waktu sholat, dan diprogram untuk 5 kota besar di Indonesia.

Dalam pembuatan alat tersebut ditemukan beberapa kelemahan diantaranya hanya 5 kota besar di Indonesia saja yang bisa diatur waktu sholatnya. Penanda waktu sholat hanya berupa satu jenis adzan, sehingga adzan pada waktu Subuh tidak sesuai dengan yang seharusnya. Serta tidak adanya penunjuk arah kiblat.

Dari skripsi dengan judul “Perencanaan dan pembuatan alat penunjuk waktu sholat dengan menggunakan mikrokontroler AT89C51” tersebut memungkinkan untuk dikembangkan dengan menggunakan sensor magnet sebagai penunjuk arah kiblatnya.

1.2 Rumusan Masalah

Dari latar belakang masalah di atas dapat dirumuskan permasalahan sebagai berikut:

1. Bagaimana cara membuat alat yang dapat memberikan informasi waktu sholat.
2. Bagaimana cara membuat alat yang dapat menunjukkan arah kiblat.
3. Bagaimana merancang dan membuat antarmuka sensor dengan rangkaian.
4. Bagaimana cara menulis dan membaca isi IC suara.
5. Bagaimana cara membuat perangkat lunak yang akan dipakai untuk menunjukkan arah kiblat dan informasi waktu sholat.

1.3 Ruang Lingkup

Dalam perencanaan dan pembuatan skripsi ini perlu dilakukan pembatasan masalah. Pembatasan masalah yang diajukan dalam skripsi ini antara lain:

1. Hanya dibatasi untuk kota-kota besar di Indonesia.
2. Perhitungan waktu sholat berdasarkan hisab.
3. Perhitungan arah kiblat berdasarkan lokasi.
4. Sistem memberikan keluaran hanya dari *speaker*.

1.4 Tujuan

Tujuan penyusunan skripsi ini adalah merealisasikan sebuah alat yang dapat memberikan panduan arah kiblat dan informasi waktu sholat bagi penderita tuna netra.

1.5 Sistematika Penulisan

Sistematika penyusunan skripsi ini dibagi menjadi beberapa bagian utama, yaitu:

- **Bagian Awal**

Bagian awal ini terdiri dari halaman judul, lembar persetujuan, kata pengantar, daftar isi, daftar tabel, daftar gambar dan daftar lampiran.

- **Bagian Inti**

Bagian inti terbagi menjadi 5 bab yang terdiri dari:

BAB I PENDAHULUAN

Memuat tentang latar belakang permasalahan, pembatasan masalah, tujuan penulisan, metodologi, serta sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Membahas teori dasar, cara kerja rangkaian serta hubungan dengan perangkat yang digunakan.

BAB III METODOLOGI PENELITIAN

Membahas perencanaan setiap rangkaian elektronika dan perangkat lunak yang dipergunakan dalam alat penunjuk arah kiblat dan informasi waktu sholat bagi penderita tuna netra.

BAB IV PERANCANGAN DAN PEMBUATAN ALAT

Membahas perancangan dan proses pembuatan alat yang meliputi perencanaan blok diagram, prinsip kerja dan realisasi alat.

BAB V PENGUJIAN DAN ANALISIS

Membahas hasil pengujian terhadap alat yang telah direalisasikan

- **Bagian Akhir**

BAB VI PENUTUP

Memuat kesimpulan dan saran-saran untuk pengembangan lebih lanjut dari alat yang telah dibuat.

BAB II

TINJAUAN PUSTAKA

Dalam merencanakan dan merealisasikan sistem Penunjuk Arah Kiblat dan Informasi Waktu Sholat Bagi Penderita Tuna Netra dibutuhkan pemahaman mengenai teori-teori penunjang yang mendukung. Pemahaman ini akan bermanfaat dalam perencanaan dan perealisasi perangkat keras dan perangkat lunak sistem. Teori-teori penunjang tersebut meliputi dasar penentuan waktu sholat dan arah kiblat, sensor magnet CMPS03, *Real Time Clock* (RTC), IC suara ISD2500, EEPROM, mikrokontroler ATMEGA8535, dan tombol tekan (*push button*).

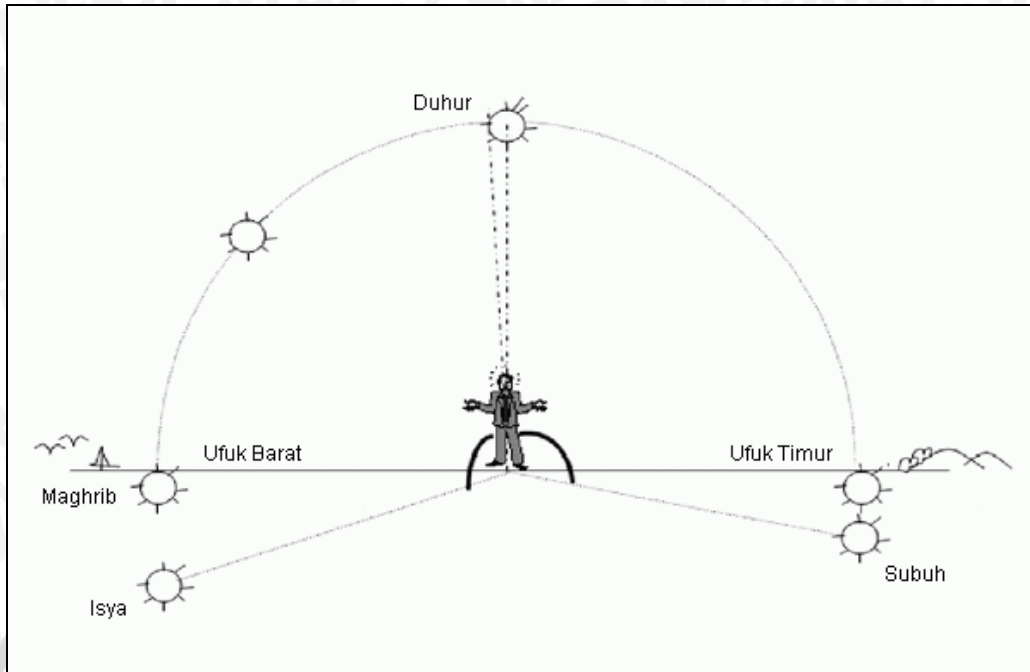
2.1 Dasar Penentuan Waktu Sholat

Penentuan awal dan akhir waktu sholat berdasarkan pada posisi matahari dilihat dari suatu tempat di permukaan bumi. Awal dan akhir waktu sholat dapat dirinci sebagai berikut:

- Waktu sholat Duhur dimulai saat matahari tergelincir kearah barat, sampai panjang bayang-bayang suatu benda sama dengan panjang bendanya.
- Waktu sholat Ashar dimulai saat panjang bayang-bayang suatu benda lebih panjang dari panjang bendanya hingga matahari tenggelam.
- Waktu sholat Maghrib dimulai sejak matahari tenggelam sampai awan merah di ufuk barak telah hilang.
- Waktu sholat Isya dimulai ketika awan merah di ufuk barat telah hilang sampai terbit fajar sidiq di ufuk timur.
- Waktu sholat Subuh dimulai sejak terbit fajar sidiq di ufuk timur sampai terbitnya matahari.

(<http://rukyatulhلال.tripod.com/jadwalshalat>, 2006)

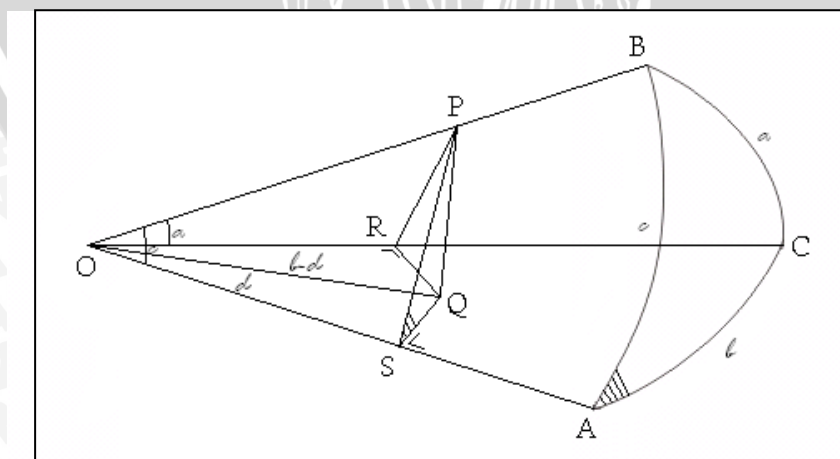
Untuk lebih jelasnya, waktu-waktu sholat tersebut dapat dilihat dalam Gambar 2.1.



Gambar 2.1 Waktu-waktu sholat

Sumber: <http://rukyaatulhلال.tripod.com/jadwalshalat>

Untuk menghisab awal waktu sholat yang cukup teliti adalah dengan menggunakan ilmu ukur segitiga bola atau *spherical trigonometri*. Pada dasarnya bila tiga buah lingkaran besar pada permukaan sebuah bola saling berpotongan, terjadilah sebuah segitiga bola. Ketiga titik potong merupakan titik sudut A, B dan C; besar masing-masing sudut segitiga bola itupun dinamakan A, B dan C. Sisi-sisinya dinamakan berturut-turut dengan a, b dan c, yaitu yang berhadapan dengan sudut A, B dan C. Untuk mencegah keraguan-raguan, sisi itu biasanya diambil, yang kurang dari seperdua lingkaran. Gambar dari segitiga bola dapat dilihat dalam Gambar 2.2.



Gambar 2.2 Segitiga Bola

Sumber: Abdur Rachim, 1983:64

Dengan menggunakan hukum cosinus, diperoleh:

$$\cos A = \frac{\cos a - \cos b \cos c}{\sin b \sin c} = \frac{\cos a}{\sin b \sin c} - \cot b \cot c \quad (1)$$

Karena Persamaan (1) tidak dapat diselesaikan dengan persamaan logaritma maka bentuknya diubah menjadi:

$$\begin{aligned} \sin^2 \frac{1}{2}A &= \frac{1 - \cos A}{2} \\ &= \frac{1}{2} - \frac{\cos a - \cos b \cos c}{2 \sin b \sin c} \\ &= \frac{\sin b \sin c + \cos b \cos c - \cos a}{2 \sin b \sin c} \\ &= \frac{\cos(b - c) - \cos a}{2 \sin b \sin c} \\ &= \frac{\sin \frac{1}{2}(a - b + c) \sin \frac{1}{2}(a + b - c)}{\sin b \sin c} \end{aligned} \quad (2)$$

Jika dimasukkan dalam Persamaan (2) harga $a + b + c = 2s$, maka akan diperoleh:

$$\sin^2 \frac{1}{2}A = \frac{\sin(s - b) \sin(s - c)}{\sin b \sin c} \quad (3)$$

Bila A sudah diketahui, maka kedua sudut yang lain dapat dicari dengan rumus sinus misalnya:

$$\sin B = \frac{\sin b \sin A}{\sin a} \quad (4)$$

Bila dalam Persamaan (3) dilakukan penggantian tanda $A = t$; $a = 90^\circ - h$; $b = 90^\circ - \delta$; $c = 90^\circ - \phi$, maka diperoleh persamaan:

$$\begin{aligned} \cos t &= \frac{\sin(h) - \sin \delta \sin \phi}{\cos \delta \cos \phi} \\ &= \frac{\sin(h)}{\cos \delta \cos \phi} - \tan \delta \tan \phi \\ &= -\tan \delta \tan \phi + \sec \delta \sec \phi \sec(h) \end{aligned} \quad (5)$$

Dengan menggunakan Persamaan (5), maka akan dihasilkan sudut waktu matahari yang diberi tanda 't' yaitu jarak matahari dari titik kulminasi diukur sepanjang lintasan harian. Sudut waktu diberi tanda positif (+) jika diukur dari titik kulminasi ke arah barat dan diberi tanda negative (-) jika diukur dari titik kulminasi ke arah timur. (Abdur Rachim, 1983)

Oleh karena menghisab waktu sholat adalah menghitung kapan matahari akan menempati posisinya. Maka hasil dari Persamaan (5) belum dapat menentukan awal waktu sholat yang dikehendaki dan masih diperlukan penambahan rumus yaitu merubah satuan derajat sudut matahari (t) menjadi satuan jam yang kemudian ditambahkan dengan saat matahari berkulminasi, yaitu titik tertinggi yang dicapai matahari dalam perjalanan hariannya. Hasil ini merupakan awal waktu sholat dengan satuan waktu pertengahan setempat atau Mean Local Time (LMT). Dan digunakan persamaan:

$$\text{waktu daerah} = \text{waktu setempat} + (\lambda_{dh} - \lambda_{tpt}) \quad (6)$$

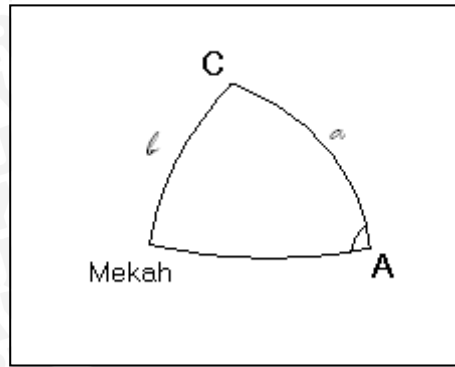
Dengan: λ_{dh} = bujur daerah

λ_{tpt} = bujur tempat

2.2 Dasar Penentuan Arah Kiblat

Kiblat merupakan perkataan Arab yang mengacu pada arah apabila seseorang muslim sholat. Asalnya mengarah ke Baitulmuqaddis atau kini Jerusalem, namun pada tahun 624 M, semasa Nabi Muhammad SAW hijrah ke Madinah, arah Kiblat ditukar ke arah Ka'bah dan tetap sampai sekarang ini. Kiblat merupakan titik acuan di bumi ke arah Ka'bah. Dalam amalan agama Islam, sholat harus menghadap ke arah Ka'bah. (www.wikipedia.com, 2005)

Kiblat menunjukkan arah terpendek dari Ka'bah. Letak geografis Ka'bah dapat ditentukan secara geometri untuk menentukan arah Kiblat dari semua tempat di bumi ini. Lokasi Ka'bah adalah di 21°25' Lintang Utara dan 39°50' Bujur Timur. (Almanak Hisab Rukyat, Badan Hisab & Rukyat Dep. Agama)



Gambar 2.3 Sudut acuan untuk menentukan arah kiblat
Sumber: Anonymous, 1981

Untuk melakukan perhitungan arah kiblat dapat dilakukan dengan menggunakan rumus-rumus:

$$1. \quad \cot A = \frac{\cot b \cdot \sin a}{\sin C} - \cos a \cdot \cot C \quad (7)$$

$$2. \quad \cot A = \frac{\cos \phi \cdot \tan 21^{\circ}25'}{\sin(\lambda - 39^{\circ}50')} - \frac{\sin \phi}{\tan(\lambda - 39^{\circ}50')} \quad (8)$$

Dengan:

A = arah kiblat suatu tempat (yaitu sudut yang dibentuk antara busur Mekah–tempat sholat, dengan busur Kutub Utara–tempat sholat)

C = sudut Kutub Utara (yaitu sudut yang dibentuk antara busur Mekah–Kutub Utara, dengan busur tempat sholat–Kutub Utara)

a = 90° – lintang tempat (yaitu panjang busur antara titik Kutub Utara dengan tempat yang akan dicari arah kiblatnya)

b = 90° – lintang Ka’bah (yaitu panjang busur antara titik Kutub Utara dengan Ka’bah)

ϕ = lintang tempat/kota yang dicari arah kiblatnya

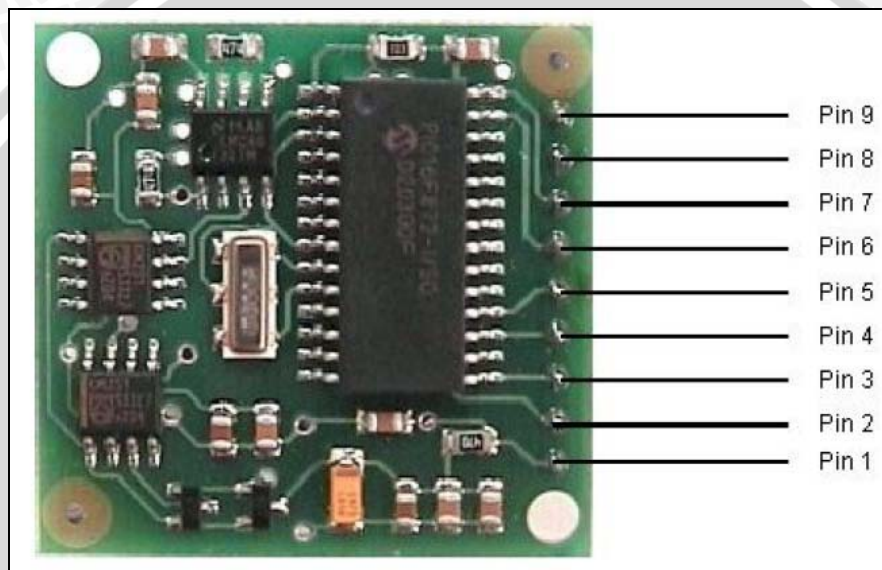
λ = bujur tempat/kota yang dicari arah kiblatnya

2.3 Sensor Magnet CMPS03

Untuk mendeteksi arah tentunya diperlukan suatu sensor yang dapat mendeteksi medan magnet bumi. Agar pendeteksian arah tersebut akurat, maka diperlukan suatu sensor yang memiliki sensitivitas yang tinggi, sehingga untuk medan magnet yang sangat lemah sekalipun masih dapat diukur. Selain itu juga diperlukan sensor yang

tahan terhadap kondisi-kondisi tertentu yang dapat mengganggu pendeteksian ini. Misalnya saja daya tahan terhadap suhu, tekanan ataupun yang lainnya. (www.phillips.com, 2000)

Untuk itu dapat digunakan sensor magnetoresistif KMZ51 yang memiliki sensitivitas tinggi dalam pendeteksian medan magnet, memiliki daerah operasi temperatur yang lebar, offset stabil yang rendah dan sensitivitas yang rendah terhadap tekanan mekanik. Dan sensor magnetoresistif KMZ51 ini digunakan dalam sensor magnet CMPS03. Gambar sensor CMPS03 dapat dilihat dalam Gambar 2.4.



Gambar 2.4 Sensor magnet CMPS03

Sumber: www.superdroidrobots.com

Keterangan dari kaki-kaki pin pada sensor magnet CMPS03 ditunjukkan dalam Tabel 2.1.

Tabel 2.1 Keterangan kaki-kaki pin sensor magnet CMPS03

Kaki Pin	Keterangan
Pin 1	VCC (+5V)
Pin 2	SCL
Pin 3	SDA
Pin 4	PWM
Pin 5	N/C
Pin 6	Kalibrasi
Pin 7	50/60 Hz
Pin 8	N/C
Pin 9	GND

Sumber: www.superdroidrobots.com

Sensor magnet CMPS03 memiliki resolusi 0,1°, sehingga hasil keluaran yang dihasilkan akan menjadi lebih presisi. Sebelum digunakan sensor magnet CMPS03 ini perlu dikalibrasi terlebih dahulu dengan mencocokkan sesuai arah-arah mata angin bumi yaitu utara sebagai referensi utama yang selanjutnya diikuti timur, selatan, dan barat. Hasil dari kalibrasi ini akan disimpan dalam EEPROM internal dari sensor magnet CMPS03, oleh karena itu proses kalibrasi cukup dilakukan satu kali saja. Untuk komunikasinya, sensor ini memiliki dua cara yaitu menggunakan komunikasi I2C dan komunikasi PWM. Sehingga dapat dipilih salah satu cara komunikasi untuk mendapatkan hasil keluaran dari sensor magnet CMPS03 ini. (www.superdroidrobots.com).

Sensor magnet CMPS03 ini memiliki 16 buah register yang masing-masingnya memiliki kapasitas sebesar 1 *byte*. Fungsi register-register sensor magnet CMPS03 dapat dilihat dalam Tabel 2.2.

Tabel 2.2 Register sensor magnet CMPS03

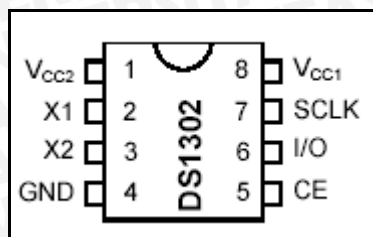
Register	Fungsi
0	Nomor revisi perangkat lunak
1	<i>Compass bearing</i> dalam <i>byte</i> , 0-255 untuk satu lingkaran penuh
2,3	<i>Compass bearing</i> dalam <i>word</i> , 0-3599 untuk satu lingkaran penuh (menunjukkan 0-359,9 derajat)
4,5	Tes internal, sinyal diferensial sensor 1 – 16 bit <i>signed word</i>
6,7	Tes internal, sinyal diferensial sensor 2 – 16 bit <i>signed word</i>
8,9	Tes internal, nilai kalibrasi 1 – 16 bit <i>signed word</i>
10,11	Tes internal, nilai kalibrasi 2 – 16 bit <i>signed word</i>
12	Tidak digunakan
13	Tidak digunakan
14	Tidak digunakan
15	Perintah kalibrasi

Sumber: <http://www.robot-electronics.co.uk/htm/cmeps3doc.shtml>

2.4 RTC (*Real Time Clock*)

RTC merupakan rangkaian jam dan kalender dalam sebuah IC lengkap dengan battery back up-nya, bahkan rangkaian jam dalam IC tersebut dapat tetap bekerja walaupun power supply sudah dimatikan. RTC yang digunakan adalah tipe DS1302. IC ini dapat menghitung detik, menit, jam, tanggal, bulan, hari, dan tahun. Memiliki 31 x 8 RAM dan dalam transfer datanya baik proses baca atau tulis dapat dilakukan dengan cara *single-byte* maupun *multiple-byte*. (www.maxim-ic.com)

DS1302 memiliki delapan buah pin, dimana konfigurasi pinnya dapat dilihat dalam Gambar 2.5.



Gambar 2.5 Konfigurasi pin DS1302

Sumber: Datasheet DS1302, 2005:1

Adapun fungsi dari masing-masing pin dijelaskan dalam Tabel 2.2.

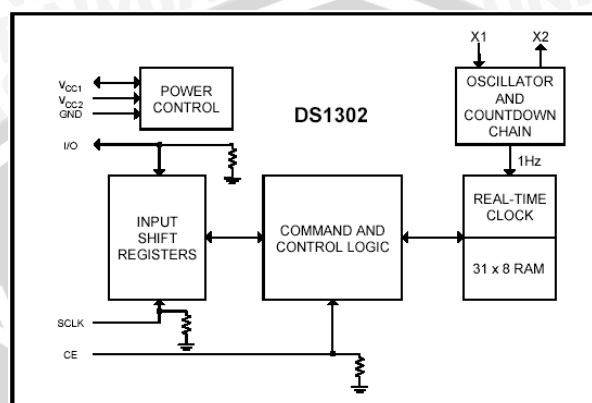
Tabel 2.3 Fungsi dari masing-masing pin

PIN	NAMA	FUNGSI
1	V _{CC2}	Pin power supply utama dalam konfigurasi supply dual. V _{CC1} dihubungkan pada sumber backup dan memelihara kerja dari DS1302 jika power supply utama tidak mampu untuk mencatu. Jika V _{CC2} lebih besar dari V _{CC1} + 0,2 volt, maka DS1302 akan dicatu oleh V _{CC2} . Jika V _{CC2} kurang dari V _{CC1} maka V _{CC1} yang akan mencatu DS1302.
2	X1	Koneksi untuk 32,768kHz Quartz Crystal Standard. Osilator internal dirancang untuk beroperasi dengan crystal yang mempunyai beban kapasitansi sebesar 6pF. Dapat juga digunakan osilator eksternal 32,768kHz dimana pin X1 dihubungkan ke osilator eksternal dan pin X2 diambangkan.
3	X2	
4	GND	Ground
5	CE	Input. Sinyal CE harus dinyatakan tinggi selama proses baca atau tulis. Pin ini mempunyai 40kΩ resistor pulldown internal yang terhubung ke ground.
6	I/O	Input/Push-Pull Output. Merupakan jalur data bidirectional untuk 3-wire interface dan mempunyai 40kΩ resistor pulldown internal yang terhubung ke ground.
7	SCLK	Input. Digunakan untuk perpindahan data sinkron pada antarmuka serial. Pin ini mempunyai 40kΩ resistor pulldown internal yang terhubung ke ground.

8	V_{CC1}	Digunakan sebagai <i>charger</i> daya dari DS1302 dan dihubungkan ke baterai.
---	-----------	---

Sumber: Datasheet DS1302,2004:4

IC DS1302 mempunyai elemen-elemen utama yaitu *shift register*, *control logic*, *oscillator*, *real-time clock*, dan RAM. Blok diagram dari DS1302 ditunjukkan dalam Gambar 2.6.



Gambar 2.6 Blok diagram IC DS1302

Sumber: Datasheet DS1302,2004:3

Informasi penanggalan dan waktu diperoleh dengan pembacaan byte register yang sesuai. Tabel 2.3 menggambarkan register-register RTC. Penanggalan dan waktu di-set dengan penulisan byte register yang sesuai. Isi dari register penanggalan dan waktu adalah dalam format binary-coded desimal (BCD).

Tabel 2.4 Alamat register RTC

READ	WRITE	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	RANGE
81h	80h	CH	10 Seconds			Seconds				00–59
83h	82h		10 Minutes			Minutes				00–59
85h	84h	12/24	0	10 AM/PM	Hour	Hour				1–12/0–23
87h	86h	0	0	10 Date		Date				1–31
89h	88h	0	0	0	10 Month	Month				1–12
8Bh	8Ah	0	0	0	0	0	Day			1–7
8Dh	8Ch	10 Year			Year					00–99
8Fh	8Eh	WP	0	0	0	0	0	0	0	—
91h	90h	TCS	TCS	TCS	TCS	DS	DS	RS	RS	—

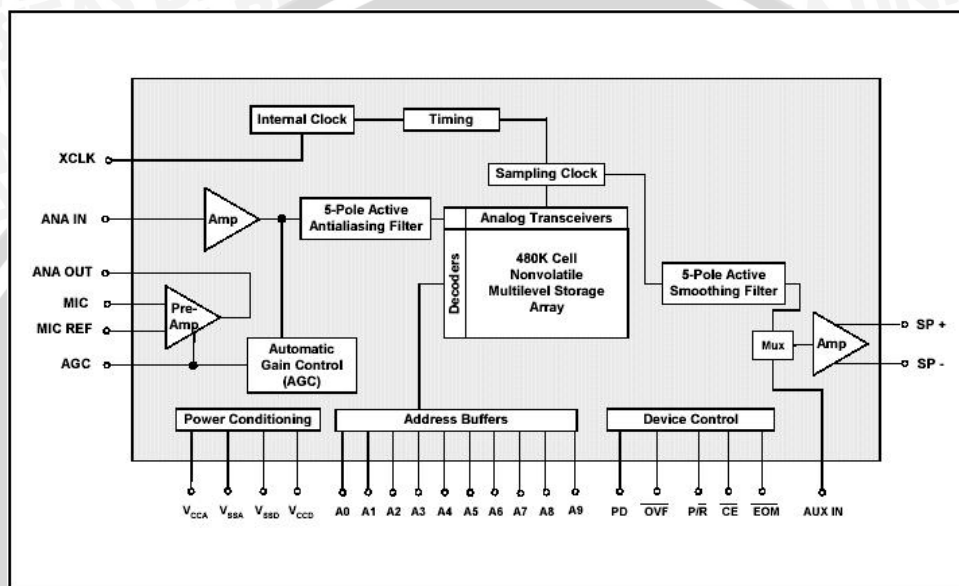
Sumber: Datasheet DS1302,2004:9

2.5 IC Suara ISD2500

Winbond ISD2500 ChipCorder menyediakan kemampuan penyimpanan pesan 60 sampai dengan 120 detik. Di dalam piranti CMOS ini tersedia *oscillator*, *microphone amplifier*, *automatic gain control*, *antialiasing filter*, *smoothing filter*, *speaker amplifier* dan *high density multi level storage array*.

Winbond ISD2500 menyediakan frekuensi sampling pada 4.0, 5.3, 6.4, dan 8.0 kHz, yang memungkinkan pengguna untuk memilih kualitas suara. Semakin bertambah durasi penyimpanan semakin berkurang frekuensi sampling dan *bandwidth* yang akan mengakibatkan perubahan kualitas suara. Sampel suara disimpan secara langsung ke dalam sebuah chip memori nonvolatil tanpa digitalisasi dan kompresi seperti solusi lainnya. (www.winbond-usa.com, 2003)

Adapun blok diagram seri ISD2500 dapat dilihat dalam Gambar 2.7.

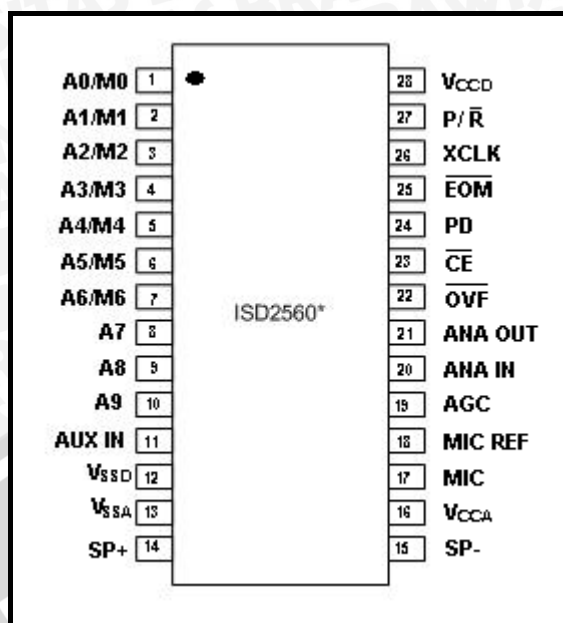


Gambar 2.7 Blok Diagram Seri ISD 2500

Sumber: Datasheet ISD2500,2003:3

Susunan pin dari ISD2500 ditunjukkan dalam Gambar 2.8 dan penjelasan dari masing – masing pin adalah sebagai berikut:

- *Voltage Input* (V_{CCA}, V_{CCD}), untuk mengurangi noise.
- *Ground Input* (V_{SSA}, V_{SSD}), pin ini harus dihubungkan dalam *power supply ground* dengan impedansi rendah.



Gambar 2.8 Konfigurasi Pin ISD2500

Sumber: Datasheet ISD2500, 2003:5

- *Power Down Input (PD)*, ketika sedang tidak digunakan untuk melakukan operasi *record* atau *playback*, pin PD harus diberi logika tinggi. Ketika pulsa *overflow* (\overline{OVF}) rendah, PD harus berlogika tinggi untuk mereset address pointer kembali ke awal *record* atau *playback*.
- *Chip Enable Input* (\overline{CE}) berfungsi untuk mengaktifkan semua operasi *record* dan *playback*.
- *Playback/Record Input* (P/\bar{R}), P/\bar{R} input ditahan dengan adanya transisi turun dari pin \overline{CE} . Logika tinggi akan memilih *playback cycle* dan logika rendah untuk memilih *record cycle*.
- *End-Of-Message Run Output* (\overline{EOM}), sebuah alamat akan dimasukkan secara otomatis di akhir masing-masing perekaman. Pulsa output \overline{EOM} akan rendah untuk setiap periode dari T_{EOM} di akhir masing-masing perekaman.
- *Microphone Input (MIC)*, digunakan untuk mentransfer sinyal suara ke on-chip *preamplifier*.
- *Microphone Reference Input (MIC REF)*, merupakan input pembalik ke *microphone preamplifier* yang memberikan *noise-cancelling* atau *common-mode rejection* input ke IC ketika dihubungkan ke sebuah *microphone* diferensial.

- *Overflow Output* (\overline{OVF}), sinyal pulsa rendah pada akhir tempat memori, mengindikasikan bahwa IC ini telah terpenuhi dan pesan telah melebihi kapasitas. Keluaran (\overline{OVF}) kemudian diikuti masukan \overline{CE} sampai pulsa PD telah mereset. Pin ini juga berfungsi untuk menambah beberapa IC ISD2500 untuk menambah durasi *record/playback*.
- *Automatic Gain Control Input* (AGC), secara dinamik mengubah penguatan dari *preamplifier* untuk mengimbangi dari lebar jarak dari level *microphone* input. AGC memberikan jarak secara penuh dari suara rendah ke tinggi untuk direkam dengan distorsi minimal.
- *Analog Output* (ANA OUT), pin ini memberikan *preamplifier* output ke pengguna.
- *Analog Input* (ANA IN), pin ini akan mentransfer sinyal ke dalam chip untuk perekaman.
- *External Clock Input* (XCLK), untuk ISD2500 mempunyai sebuah internal pulldown. Frekuensi clock sampling internal kurang lebih 1% dari spesifikasi. Frekuensi ini bervariasi dari $\pm 2,25\%$ berada pada suhu kamar dan dalam rane tegangan operasi. Internal clock mempunyai toleransi $\pm 5\%$ pada temperatur dan tegangan kerja. Jika pin XCLK tidak digunakan, pin ini harus dihubungkan ke ground.
- *Speaker Output* (SP+/SP-), ISD2500 telah mempunyai sebuah *driver on-chip diferensial speaker* yang sanggup memikul beban 50mW dalam 16Ω dari AUX IN. Speaker output berada pada level V_{SSA} selama proses *record* dan *power down*.
- *Auxiliary Input* (AUX IN), dihubungkan langsung ke kaki keluaran *amplifier* dan keluaran *speaker* ketika $\overline{CE}, P/\overline{R}$ berada pada logika tinggi dan *playback* tidak aktif.
- *Address Mode Input* (AX/MX), mempunyai dua fungsi tergantung pada level dari dua *Most Significant Bit* (MSB) dari alamat tersebut (A8 dan A9). Jika salah satu atau keduanya dari MSB berlogika rendah, semua input dianggap sebagai bit alamat dan digunakan sebagai awal alamat untuk proses *record* atau *playback cycle* terbaru. Alamat input ditahan transisi turun dari \overline{CE} . Jika kedua MSB berlogika tinggi, address/mode input dianggap sebagai bit mode.

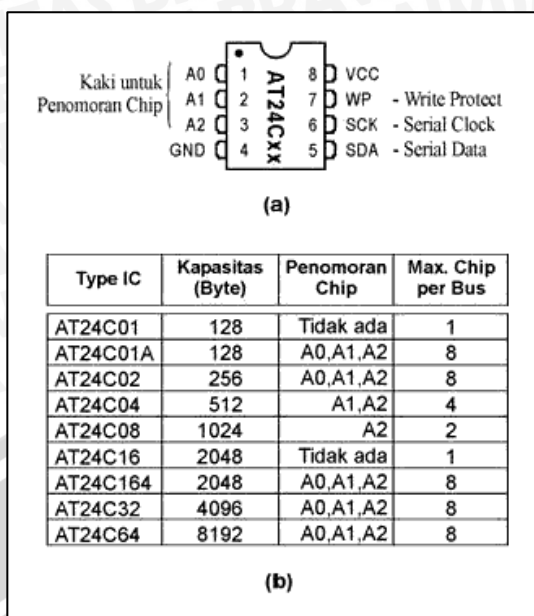
2.6 EEPROM (*Electrically Erasable Programmable Memory*)

Secara garis besar dikenal ada 2 jenis komponen memori yaitu RAM (*Random Access Memory*) dan ROM (*Read Only Memory*). Pada saat ini komponen yang sering digunakan ialah EEPROM (*Electrically Erasable Programmable Memory*). EEPROM merupakan memori yang bersifat tetap tertahan (*non-volatile*) walaupun catu daya dimatikan, selain itu EEPROM memiliki kelebihan dibandingkan dengan generasi-generasi sebelumnya seperti EPROM yang penghapusan memorinya harus dengan menggunakan sinar UV dengan tenggang waktu tertentu. EEPROM digunakan sebagai penyimpan data. (www.atmel.com, 2002)

Atmel memproduksi Serial EEPROM jenis I2C dengan kode AT24Cxx, AT merupakan kode pabrik Atmel, 24 menandakan bahwa IC tersebut adalah Serial EEPROM, sedangkan xx merupakan angka yang mengindikasikan kapasitas Serial EEPROM itu dalam satuan KiloBit, sebagai contoh AT24C08 merupakan IC EEPROM I2C berkapasitas 8 KiloBit (1 KiloByte).

Keluarga AT24Cxx terdiri dari 9 macam IC seperti terlihat di Gambar 2.9(b), kesembilan IC itu berbeda kapasitas, tapi mempunyai susunan kaki IC yang sama, seperti terlihat dalam Gambar 2.9(a).

Kaki SDA (kaki nomor 5) dan kaki SCK (kaki nomor 6) merupakan kaki baku IC jenis I2C, kedua kaki inilah yang membentuk I2C Bus. Kaki nomor 7 (WP – *Write Protect*) merupakan kaki yang dipakai untuk melindungi isi yang disimpan di dalam IC Serial EEPROM, jika kaki ini diberi tegangan '1' maka IC dalam keadaan ter-proteksi, isinya tidak dapat diganti. Agar bisa menuliskan informasi ke dalam IC ini, kaki ini harus diberi tegangan '0'.



Gambar 2.9 (a) Susunan kaki IC AT24Cxx (b) Keluarga I2C Serial EEPROM IC AT24Cxx.

Sumber : Atmel, 2002

Kaki nomor 1 sampai dengan nomor 3 (A0, A1 dan A2) merupakan fasilitas untuk penomoran chip, hal ini diperlukan kalau dalam satu rangkaian dipakai lebih dari satu IC EEPROM sejenis. Misalnya dalam satu rangkaian dipakai 3 chip AT24C02, SDA dan SCK ketiga IC ini masing-masing dihubungkan jadi satu membentuk I2C Bus, agar ketiga IC ini bisa dipakai secara terpisah kaki A0..A2 (kaki nomor 1 sampai nomor 3) masing-masing AT24C02 diberi level tegangan seperti terlihat dalam Tabel 2.4.

Tabel 2.5 Contoh Susunan A0..A2.

	A2	A1	A0
AT24C02 No. 1	0	0	0
AT24C02 No. 2	0	0	1
AT24C02 No. 3	0	1	0

Sumber: Atmel, 1997

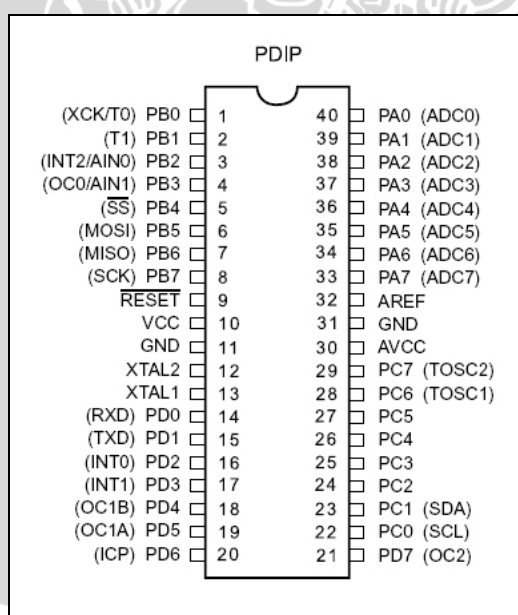
Meskipun demikian A0, A1, dan A2 tidak selalu ada pada semua IC anggota AT24Cxx, akibatnya jumlah IC yang boleh dipasang pada I2C Bus tidak sama, Gambar 2.15(b) memperlihatkan distribusi A0,A1, dan A2 pada masing-masing IC dan jumlah IC maksimal yang dapat dipakai bersama. AT24C01 sampai AT24C16 memakai metode pengalamatan 11 bit, sedangkan AT24C164 sampai AT24C64 memakai metode pengalamatan 16 bit.

2.7 Mikrokontroler ATMEGA8535

Mikrokontroler ATMEGA8535 yang diproduksi oleh ATMEL Company Amerika Serikat merupakan salah satu anggota keluarga dari jenis AVR. IC jenis ini berorientasi pada kontrol yang dapat diprogram ulang. Mikrokontroler ATMEGA8535 mempunyai karakteristik utama sebagai berikut:

- CPU dengan lebar data 8 bit
- Empat buah saluran I/O 8 bit
- Ruang memori program sebesar 8 Kbyte
- Ruang memori data sebesar 512 byte
- EEPROM sebesar 512 byte yang dapat diprogram saat operasi
- Tiga buah *Timer/Counter* dengan kemampuan perbandingan
- Unit interupsi internal dan eksternal
- *Oscillator* internal terdapat dalam chip

Konfigurasi pin mikrokontroler ATMEGA8535 dapat dilihat dalam Gambar 2.10.



Gambar 2.10 Konfigurasi Pin ATMEGA8535

Sumber: Datasheet ATMEGA8535, 2002:2

2.7.1 Struktur dan Operasi Port

Mikrokontroler ATMEGA8535 memiliki 4 buah port yang masing-masing memiliki 8 buah jalur I/O yang bersifat *bidirectional*. Beberapa karakteristik port mikrokontroler ATMEGA8535 dijelaskan secara singkat berikut ini:

1. Unit I/O dapat dialamati per jalur atau per port
2. Setiap jalur I/O memiliki *buffer*, penahan (*latch*), pengarah input dan output
3. Port A merupakan port I/O 8 bit dua arah dengan *pull up* internal. Fungsi tambahan dari port A adalah sebagai port masukan ADC internal ADC0-ADC7
4. Port B merupakan port I/O 8 bit dua arah dengan *pull up* internal. Fungsi tambahan dari port B dapat dilihat dalam Tabel 2.5

Tabel 2.6 Fungsi Tambahan Port B

Port pin	Fungsi tambahan
PB7	SCK (SPI Bus Serial Clock)
PB6	MISO (SPI Bus Master Input/Slave Output)
PB5	MOSI (SPI Bus Master Output/Slave Input)
PB4	SS (SPI Slave Select Input)
PB3	AIN1 (Analog Comparator Negative Input) OC0 (Timer/Counter0 Output Compare Match Output)
PB2	AIN0 (Analog Comparator Positive Input) INT2 (External Interrupt 2 Input)
PB1	T1 (Timer/Counter1 External Counter Input)
PB0	T0 (Timer/Counter0 External Counter Input) XCK (USART External Clock Input/Output)

Sumber: Datasheet ATMEGA8535, 2002

5. Port C merupakan port I/O 8 bit dua arah dengan *pull up* internal. Fungsi tambahan dari port C dapat dilihat dalam Tabel 2.6.

Tabel 2.7 Fungsi Tambahan Port C

Port pin	Fungsi tambahan
PC7	TOSC2 (Timer Oscillator Pin 2)
PC6	TOSC1 (Timer Oscillator Pin 1)
PC1	SDA (Two-wire Serial Bus Data Input/Output Line)
PC0	SCL (Two-wire Serial Bus Clock Line)

Sumber: Datasheet ATMEGA8535, 2002

6. Port D merupakan port I/O 8 bit dua arah dengan pull up internal. Fungsi tambahan dari port C dapat dilihat dalam Tabel 2.7.

Tabel 2.8 Fungsi Tambahan Port D

Port pin	Fungsi tambahan
PD7	OC2 (<i>Timer/Counter2 Output Compare Match Output</i>)
PD6	ICP (<i>Timer/Counter1 Input Capture Pin</i>)
PD5	OC1A (<i>Timer/Counter1 Output Compare A Match Output</i>)
PD4	OC1B (<i>Timer/Counter1 Output Compare B Match Output</i>)
PD3	INT1 (<i>External Interrupt1 Input</i>)
PD2	INT0 (<i>External Interrupt0 Input</i>)
PD1	TXD (<i>USART Output Pin</i>)
PD0	RXD (<i>USART Input Pin</i>)

Sumber: Datasheet ATMEGA8535, 2002

2.7.2 Sistem Interrupt

Mikrokontroler ATMEGA8535 memiliki 21 alamat vektor *interrupt* dimana nomor urut dari vektor *interrupt* tadi menyatakan prioritas dari *interrupt* tersebut. Alamat vektor *interrupt* dari mikrokontroler ATMEGA8535 dapat dilihat dalam Tabel 2.8.

Tabel 2.9 Alamat Vektor *Interrupt* ATMEGA8535

Vector No.	Program Address ⁽⁹⁾	Source	Interrupt Definition
1	0x000 ⁽¹⁾	RESET	External Pin, Power-on Reset, Brown-out Reset and Watchdog Reset
2	0x001	INT0	External Interrupt Request 0
3	0x002	INT1	External Interrupt Request 1
4	0x003	TIMER2 COMP	Timer/Counter2 Compare Match
5	0x004	TIMER2 OVF	Timer/Counter2 Overflow
6	0x005	TIMER1 CAPT	Timer/Counter1 Capture Event
7	0x006	TIMER1 COMPA	Timer/Counter1 Compare Match A
8	0x007	TIMER1 COMPB	Timer/Counter1 Compare Match B
9	0x008	TIMER1 OVF	Timer/Counter1 Overflow
10	0x009	TIMER0 OVF	Timer/Counter0 Overflow
11	0x00A	SPI, STC	Serial Transfer Complete
12	0x00B	USART, RXC	USART, Rx Complete
13	0x00C	USART, UDRE	USART Data Register Empty
14	0x00D	USART, TXC	USART, Tx Complete
15	0x00E	ADC	ADC Conversion Complete
16	0x00F	EE_RDY	EEPROM Ready
17	0x010	ANA_COMP	Analog Comparator
18	0x011	TWI	Two-wire Serial Interface
19	0x012	INT2	External Interrupt Request 2
20	0x013	TIMER0 COMP	Timer/Counter0 Compare Match
21	0x014	SPM_RDY	Store Program Memory Ready

Sumber: Datasheet ATMEGA8535, 2002

2.7.3 Antarmuka Serial Dua Kabel (*Two-Wire Serial Interface*)

Mikrokontroler ATMEGA8535 dilengkapi antarmuka serial dua kabel yang memiliki beberapa fitur sebagai berikut:

- Antarmuka komunikasi yang sederhana tetapi *powerful* dan fleksibel, serta hanya memerlukan dua jalur bus
- Dapat beroperasi sebagai pengirim dan penerima
- 7 bit alamat yang memungkinkan untuk 128 alamat *slave* yang berbeda
- Mendukung *multi-master arbitration*
- Kecepatan transfer data sampai 400 kHz
- Alamat *slave* yang dapat diprogram dengan dukungan *general call*

2.8 Tombol Tekan (*Push Button*)

Tombol tekan pada prinsipnya sama dengan saklar. Biasanya digunakan untuk pengendalian atau pengaturan peralatan, untuk memberi tanda-tanda hubung, putus dan sebagainya. Tombol tekan mempunyai konstruksi sederhana terdiri atas dua bagian utama, yaitu bagian mekanis yang berupa tombol dan bagian listrik yang berupa bagian kontak.

Ada dua macam kondisi pada tombol tekan yaitu kontak NO (*normally open*) dan kontak NC (*normally close*). Kondisi kontak NO dalam keadaan normalnya terbuka sedangkan untuk kontak NC dalam keadaan normalnya terhubung. Dalam pengoperasiannya tombol tekan dapat dibedakan menjadi dua macam, yaitu tombol tekan sesaat dan tombol tekan pengunci. Tombol tekan sesaat bekerja bila tombolnya ditekan, dan akan kembali bila tombolnya dilepas. Sedangkan untuk tombol tekan pengunci bila tombolnya ditekan akan berkerja dan selama itu akan dipertahankan, walaupun tombolnya dilepaskan. Hal ini disebabkan konstruksi jenis ini dilengkapi dengan kunci mekanis, untuk kembali ke posisi semula harus ditekan sekali lagi. Tombol tekan yang akan dipakai dalam tugas akhir ini adalah tombol tekan sesaat.

BAB III

METODOLOGI PENELITIAN

Penyusunan skripsi ini didasarkan pada masalah yang bersifat aplikatif, yaitu perencanaan dan perealisasiian alat agar dapat menampilkan unjuk kerja sesuai dengan yang direncanakan dengan mengacu pada rumusan masalah. Data dan spesifikasi komponen yang digunakan dalam perencanaan merupakan data sekunder yang diambil dari buku data komponen elektronika. Pemilihan komponen berdasarkan perencanaan dan disesuaikan dengan komponen yang ada di pasaran. Langkah-langkah yang perlu dilakukan untuk merealisasikan alat penunjuk arah kiblat dan informasi waktu sholat adalah:

1. Studi Literatur
2. Perancangan Alat
3. Perealisasiian Alat
4. Pengujian Alat
5. Penarikan Kesimpulan

3.1 Studi Literatur

Studi literatur digunakan untuk mempelajari teori penunjang sistem yang dibutuhkan dalam perancangan dan perealisasiian alat. Teori yang diperlukan meliputi sensor magnet CMPS03, mikrokontroler ATMEGA8535, RTC DS1032, EEPROM AT24C64, dan IC ISD2590.

3.2 Perancangan Alat

Perancangan alat penunjuk arah kiblat dan informasi waktu sholat meliputi:

1. Penentuan spesifikasi alat yang direncanakan adalah sebagai berikut:
 - a. Menggunakan sensor magnet CMPS03 untuk mengetahui posisi suatu tempat
 - b. Menggunakan mikrokontroler ATMEGA8535 sebagai pengontrol utama
 - c. *Real Time Clock* (RTC) digunakan untuk memperoleh informasi waktu
 - d. Sebagai perekam dan penampil suara digunakan ISD2590
 - e. Digunakan EEPROM AT24C64 untuk menyimpan *database* lokasi tempat dan waktu sholat
 - f. Alat dapat menampilkan suara mengenai informasi waktu sholat dan arah kiblat

2. Pemilihan komponen elektronika penyusun sistem sesuai dengan yang tersedia di pasaran dan mempelajari karakteristik tiap-tiap komponen elektronika tersebut
3. Perencanaan perangkat keras sensor magnet, mikrokontroler, RTC, EEPROM dan perekam/pemutar suara
4. Pembuatan diagram alir perangkat lunak mikrokontroler untuk menangani kebutuhan sistem yang direncanakan

3.3 Perealisasian Alat

Perealisasian alat penunjuk arah kiblat dan informasi waktu sholat meliputi:

1. Pembuatan perangkat keras sistem yang meliputi pembuatan *Printed Circuit Board* (PCB), pengujian komponen, pengeboran, perakitan, dan penyolderan komponen pada PCB
2. Pembuatan perangkat lunak yang meliputi penulisan kode, pengujian (*debugging*), dan kompilasi program menggunakan *software* Code Vision AVR C Compiler versi 1.24.8, AVR Studio 4 dan Ponyprog *downloader*

3.4 Pengujian Alat

Pengujian alat ini bertujuan untuk mengetahui apakah alat hasil perancangan dapat bekerja sesuai dengan spesifikasi yang direncanakan. Adapun bentuk pengujian yang dilakukan adalah sebagai berikut:

1. Pengujian Perangkat Keras

Untuk mengetahui unjuk kerja perangkat keras alat dilakukan pengujian pada masing-masing blok dan sistem secara keseluruhan. Pengujian antara lain berupa:

- a. Pengujian Sensor Magnet CMPS03

Pengujian ini dilakukan untuk mengetahui respon sensor terhadap medan magnet bumi.

- b. Pengujian RTC DS1302

Pengujian RTC dilakukan dengan simulasi menggunakan *Light Emitting Diode* (LED) untuk mengetahui apakah keluaran RTC DS1302 sesuai dengan yang diharapkan.

- c. Pengujian EEPROM AT24C64

Tujuan pengujian ini adalah untuk mengetahui apakah data-data yang dimasukkan ke dalam EEPROM AT24C64 dapat disimpan dan diakses kembali.

d. Pengujian ISD2590

Pengujian ISD2590 ini dilakukan untuk mengetahui kemampuan ISD2590 dalam merekam dan menampilkan suara.

e. Pengujian mikrokontroler ATMEGA8535

Tujuan pengujian ini adalah untuk mengetahui apakah program yang dibuat sesuai dengan kebutuhan sistem.

2. Pengujian Perangkat Lunak

Untuk pengujian perangkat lunak digunakan fasilitas simulasi yang terdapat pada *software* AVR Studio 4 dan juga diterapkan langsung pada rangkaian perangkat keras yang telah dirancang sebelumnya. Tujuan pengujian ini adalah untuk mengetahui kemampuan perangkat lunak untuk menangani perangkat keras yang ada.

3. Pengujian Keseluruhan Sistem

Pengujian keseluruhan sistem dilakukan dengan menyambungkan blok perangkat keras dan selanjutnya mengoperasikan sistem sehingga dapat diketahui apakah alat ini bekerja sesuai dengan spesifikasi yang diharapkan.

3.5 Penarikan Kesimpulan

Penarikan kesimpulan ini didasarkan pada kesesuaian antara perancangan yang direncanakan sebelumnya dengan hasil pengujian alat yang telah dibuat baik itu perangkat keras maupun perangkat lunak.

BAB IV

PERANCANGAN DAN PEMBUATAN ALAT

Bab ini membahas mengenai perancangan dan pembuatan alat penunjuk arah kiblat dan informasi waktu sholat bagi penderita tuna netra. Perancangan dan pembuatan alat ini terdiri dari dua bagian yaitu perancangan dan pembuatan perangkat keras serta perencanaan dan pembuatan perangkat lunak. Perancangan dan pembuatan alat dilakukan secara bertahap untuk memudahkan analisis sistem setiap bagian ataupun sistem secara keseluruhan. Beberapa aspek lain yang perlu dijelaskan dalam pembahasan bab ini adalah penentuan spesifikasi dari alat yang dirancang, blok diagram dan prinsip kerja sistem.

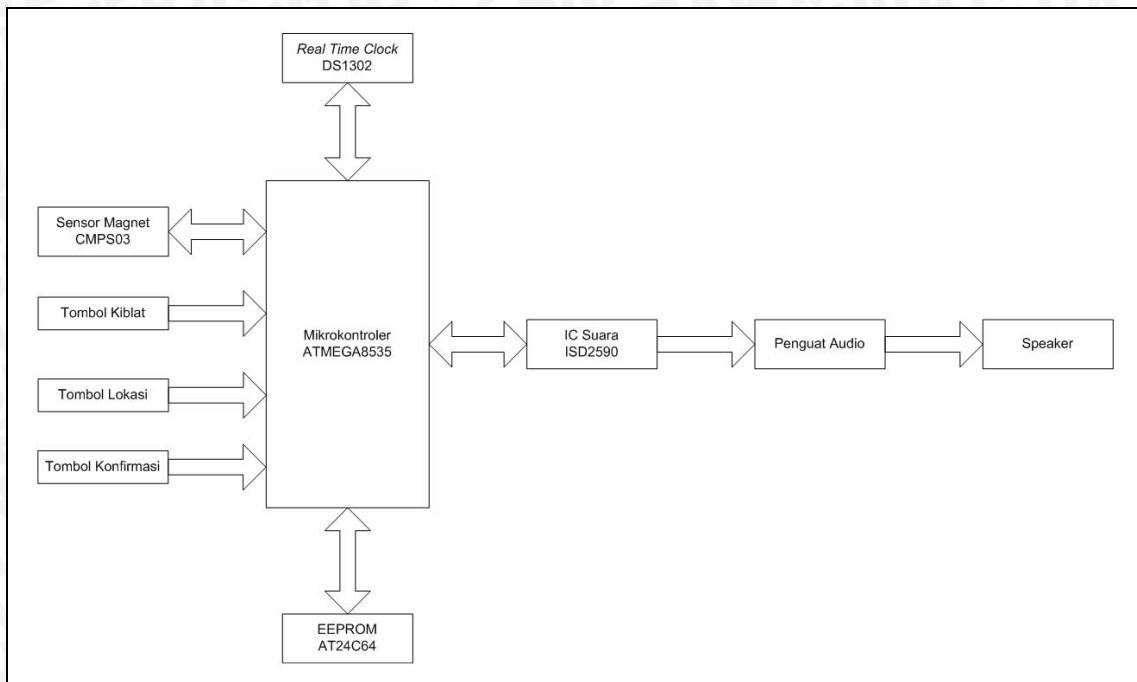
4.1 Spesifikasi Alat

Spesifikasi alat secara global ditetapkan terlebih dahulu sebagai acuan dalam perancangan selanjutnya. Spesifikasi alat yang direncanakan adalah sebagai berikut:

- Menggunakan sensor magnet untuk mengetahui posisi suatu tempat
- Menggunakan mikrokontroler ATMEGA8535 sebagai pengontrol utama
- Real Time Clock* (RTC) DS1302 digunakan untuk memperoleh informasi waktu
- Sebagai perekam dan penampil suara digunakan IC Suara ISD2590
- Digunakan EEPROM AT24C64 untuk menyimpan *database* lokasi tempat dan waktu sholat
- Alat dapat menampilkan suara mengenai informasi waktu sholat dan arah kiblat

4.2 Diagram Blok Rangkaian

Diagram blok alat penunjuk arah kiblat dan informasi waktu sholat bagi penderita tuna netra ditunjukkan dalam Gambar 4.1.



Gambar 4.1 Blok diagram penunjuk arah kiblat dan informasi waktu sholat bagi penderita tuna netra

Keterangan diagram blok yang ditunjukkan dalam Gambar 4.1 adalah sebagai berikut:

- Tombol lokasi, tombol konfirmasi dan tombol kiblat sebagai tombol masukan
- Sensor magnet CMPS03 yang keluarannya dihubungkan dengan mikrokontroler ATMEGA8535
- RTC DS1302 berfungsi sebagai penyedia informasi mengenai detik, menit, jam, tanggal, bulan, hari, dan tahun
- Mikrokontroler ATMEGA8535 sebagai pengendali utama
- EEPROM AT24C64 digunakan untuk media penyimpan data perhitungan arah kiblat dan perhitungan waktu sholat
- Hasil proses mikrokontroler dikirimkan ke IC suara ISD2500
- Speaker dihubungkan dengan keluaran dari IC suara ISD2500 yang sebelumnya telah dikuatkan oleh penguat audio

Prinsip kerja sistem yang direncanakan adalah sebagai berikut:

Ketika alat pertama kali dinyalakan, sensor magnet akan mulai mendeteksi posisi alat terhadap kutub-kutub bumi. Untuk memilih lokasi kota, pengguna harus menekan tombol lokasi dan diikuti oleh tombol konfirmasi untuk memastikan bahwa lokasi yang dipilih sudah benar. Setelah itu, jika pengguna menginginkan informasi mengenai arah kiblat maka pengguna harus menekan tombol kiblat dan selanjutnya data yang terbaca

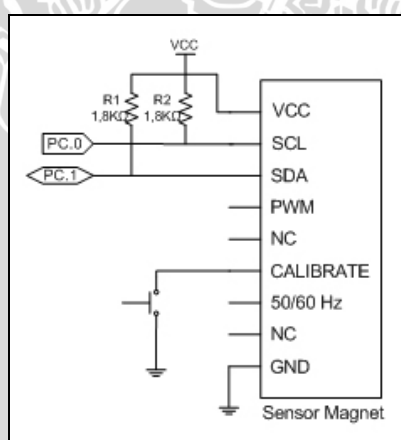
oleh sensor magnet dibaca kemudian diolah dalam mikrokontroler ATMEGA8535. Setelah itu mikrokontroler ATMEGA8535 akan mengirimkan alamat suara yang sesuai dengan hasil proses pengolahan data ke IC suara ISD2500. Dan selanjutnya IC suara ISD2500 diaktifkan. Suara akan dikeluarkan melalui speaker dengan terlebih dahulu melalui penguat audio agar suara terdengar lebih keras.

Alat ini juga akan mengeluarkan suara adzan ketika waktu dari sholat wajib tiba dan tombol lokasi telah ditekan. Ada dua jenis adzan yang berbeda untuk membedakan waktu sholat subuh dan sholat yang lainnya. Informasi waktu yang didapatkan untuk mengetahui waktu sholat ini digunakan RTC. Sebagai media penyimpan data mengenai waktu sholat dan posisi lokasi kota digunakan EEPROM.

4.3 Perancangan Perangkat Keras

4.3.1 Sensor Magnet CMPS03

Sensor magnet CMPS03 digunakan sebagai pendeteksi posisi alat terhadap kutub-kutub bumi. Rangkaian dari sensor magnet CMPS03 ditunjukkan dalam Gambar 4.2.



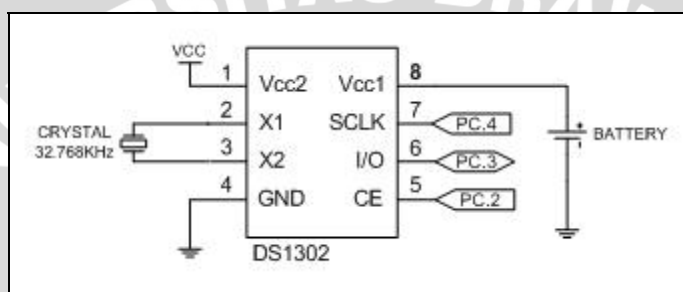
Gambar 4.2 Rangkaian Sensor Magnet

Agar sensor lebih akurat dalam mendeteksi arah dari kutub-kutub bumi, maka diperlukan kalibrasi. Dan metode yang digunakan untuk mengkalibrasi sensor adalah metode manual karena proses pengkalibrasian akan lebih cepat dan efisien. Metode manual ini dilakukan dengan menekan switch pada pin *calibrate* ketika sensor diarahkan ke utara. Setelah *switch* dilepaskan, sensor diarahkan secara perlahan-lahan ke timur kemudian *switch* ditekan kembali. Selanjutnya ke arah selatan dan barat. Proses kalibrasi ini cukup dilakukan sekali saja karena hasil dari proses kalibrasi tersebut disimpan dalam EEPROM yang terdapat pada sensor magnet CMPS03 ini.

Komunikasi yang digunakan untuk memperoleh data keluaran dari pendeteksian sensor adalah komunikasi I²C. Data hasil pendeteksian sensor akan disimpan pada register 2 dan 3 dalam modul sensor magnet CMPS03 ini. Untuk penggunaan komunikasi I2C maka pin SCL dan SDA dihubungkan dengan PC.0 dan PC.1 dari mikrokontroler.

4.3.2 Real Time Clock (RTC)

Pada alat ini RTC digunakan untuk memberikan informasi mengenai tanggal, bulan, jam, ataupun menit yang diinginkan. RTC yang digunakan adalah jenis DS1302. Rangkaian RTC dapat dilihat dalam Gambar 4.3.



Gambar 4.3 Rangkaian Real Time Clock DS1302

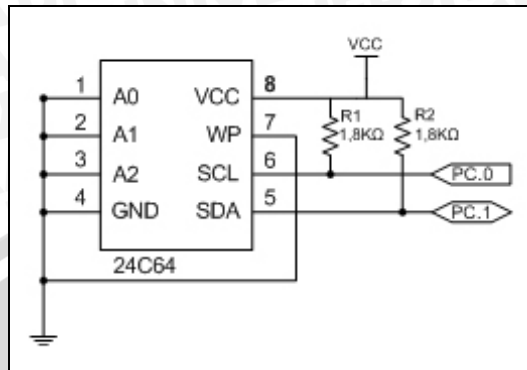
RTC jenis DS1302 merupakan IC yang memiliki dua buah catu daya. V_{CC2} dihubungkan ke catu 5 volt dan V_{CC1} dihubungkan dengan baterai *backup*. V_{CC1} berfungsi untuk memelihara kerja dari DS1302 jika catu utama tidak mampu untuk mencatu. Sesuai dengan *datasheet*, pin 2 dan 3 dihubungkan dengan *crystal* 32,768 kHz untuk memberikan frekuensi yang sesuai (Dallas Semiconductor, 2004:3).

Pin SCLK dihubungkan pada pin PC.4, pin I/O dihubungkan pada pin PC.3, dan pin CE dihubungkan pada pin PC.2 dari mikrokontroler.

4.3.3 EEPROM AT24C64

EEPROM AT24C64 pada alat ini digunakan untuk menyimpan *database* lokasi, perhitungan arah kiblat dan waktu sholat. Rangkaian antarmuka EEPROM AT24C64 ditunjukkan dalam Gambar 4.4. Pin A0-A2 merupakan pin yang digunakan untuk fasilitas penomoran *chip*, hal ini diperlukan apabila dalam satu rangkaian digunakan lebih dari satu IC EEPROM sejenis. Karena dalam perencanaan alat ini hanya memerlukan satu IC EEPROM, maka pin A0-A2 diberikan logika rendah. Pin WP (*Write Protect*) berfungsi untuk melindungi isi yang disimpan di dalam IC EEPROM.

Agar dapat mengganti isi dari IC ini maka pin WP diberikan logika rendah. Pin SCL dan SDA dihubungkan ke PC.0 dan PC.1 dari mikrokontroler dengan melalui resistor *pull-up*.



Gambar 4.4 Rangkaian EEPROM AT24C64

Dari *datasheet* mikrokontroler ATMEGA8535 diketahui bahwa frekuensi SCL dapat diketahui dengan menggunakan persamaan:

$$\begin{aligned} f_{SCL} &= \frac{f_{CK}}{16 + 2(TWBR).4^{TWPS}} \\ &= \frac{8MHz}{16 + 2(10).4^1} \\ &= 83,333 \text{ kHz} \end{aligned}$$

Dengan $f_{SCL} = 83,333 \text{ kHz}$, maka untuk mendapatkan nilai $R_{pull-up}$ minimum digunakan persamaan:

$$\begin{aligned} R_{pull-up \min} &= \frac{V_{cc} - 0,4}{3mA} \\ &= \frac{5 - 0,4}{3mA} = 1533,33\Omega = 1,533K\Omega \end{aligned}$$

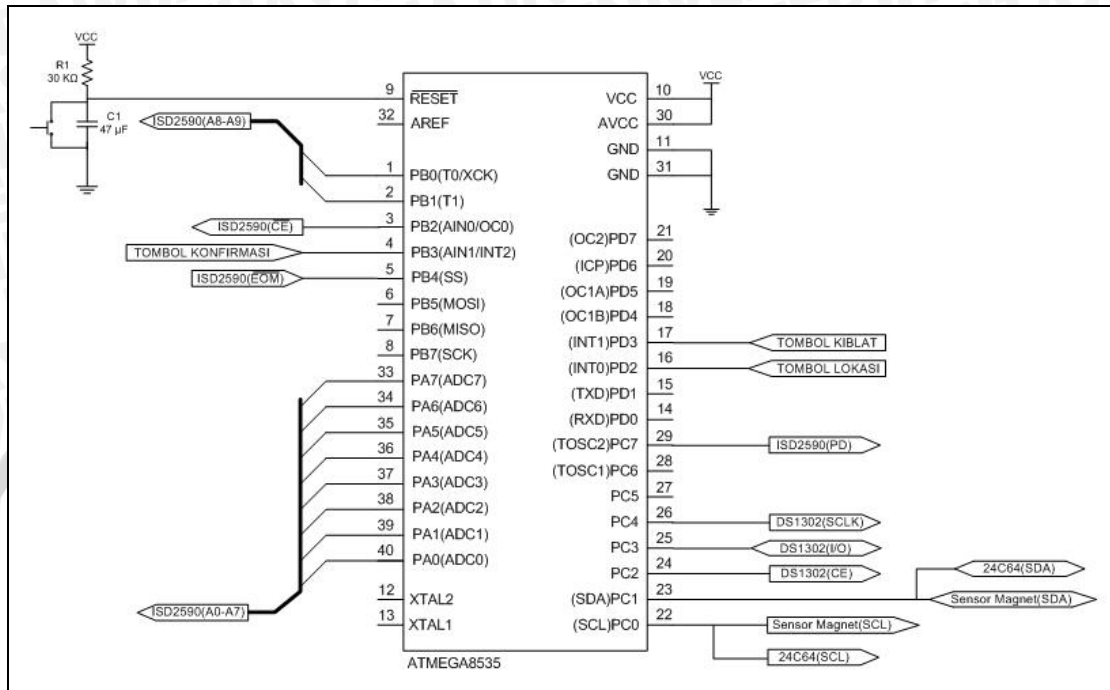
Dalam perancangan ini digunakan resistor 1,8 k Ω .

4.3.4 Rangkaian Mikrokontroler ATMEGA8535

4.3.4.1 Rangkaian Minimum Sistem Mikrokontroler ATMEGA8535

Sistem mikrokontroler ATMEGA8535 berfungsi sebagai pengendali utama dalam sistem keseluruhan seperti pengambilan data keluaran dari sensor magnet CMPS03, pengaksesan EEPROM, RTC, dan IC suara ISD. Selain itu proses perhitungan aritmetika juga dilakukan oleh mikrokontroler ini. Mikrokontroler jenis ATMEGA8535 ini dipilih karena memiliki kecepatan instruksi per MHz yang tinggi

dan pin I/O yang mencukupi untuk sistem keseluruhan serta memiliki kemampuan untuk melakukan komunikasi secara I²C. Sebagai pengendali utama dari pengolahan data dan pengontrolan sistem, pin-pin ATMEGA8535 dihubungkan dengan rangkaian pendukung membentuk suatu sistem minimum seperti dalam Gambar 4.5.



Gambar 4.5 Rangkaian Mikrokontroler ATMEGA8535

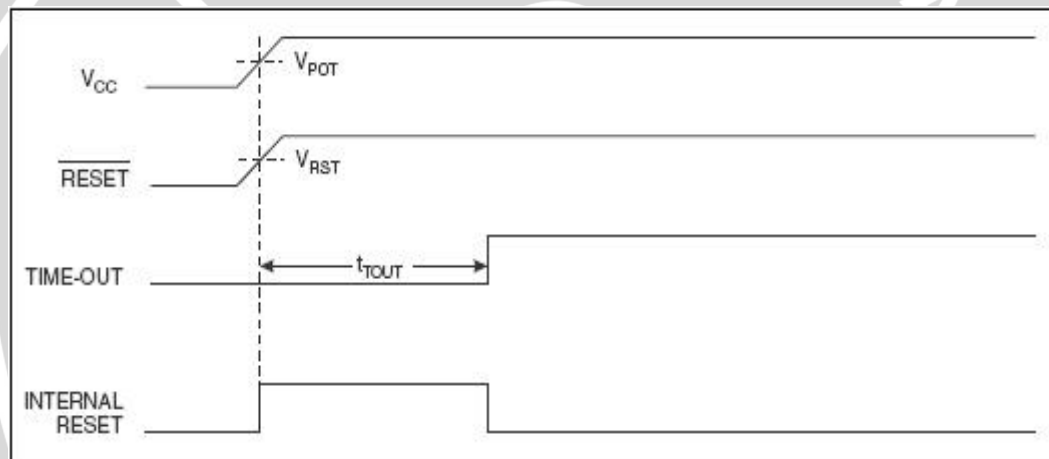
Mikrokontroler ATMEGA8535 mempunyai 4 port yaitu port A, port B, port C dan port D, 32 jalur yang dapat diprogram menjadi masukan atau keluaran. Pada perancangan ini pin-pin dari mikrokontroler yang digunakan adalah:

- PA.0 – PA.7 : digunakan sebagai jalur data/alamat dari ISD2590
- PB.0 – PB.1 : digunakan sebagai jalur data/alamat dari ISD2590
- PB.2 : digunakan sebagai sinyal kontrol dari ISD2590 yaitu pin CE
- PB.3 : digunakan sebagai tombol konfirmasi
- PB.4 : digunakan sebagai sinyal kontrol dari ISD2590 yaitu pin EOM
- PC.0 – PC.1 : digunakan untuk komunikasi dengan EEPROM AT24C64 dan sensor magnet CMPS03 yaitu pin SDA dan SCL dari masing-masing komponen
- PC.2 – PC.4 : digunakan sebagai jalur komunikasi dengan RTC DS1302 yaitu pin CE, I/O, dan SCLK
- PC.7 : digunakan sebagai sinyal kontrol dari ISD2590 yaitu pin PD
- PD.1 : digunakan sebagai tombol konfirmasi

- PD.2 : digunakan sebagai tombol lokasi
 PD.3 : digunakan sebagai tombol kiblat

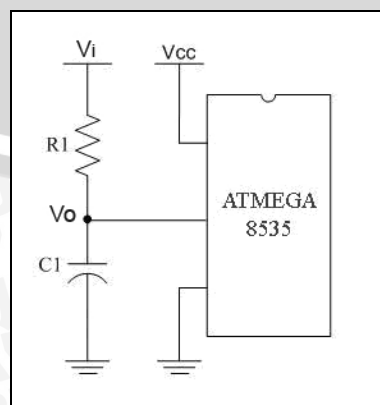
4.3.4.2 Rangkaian Reset

Rangkaian reset diperlukan untuk mereset mikrokontroler secara otomatis setiap kali catu daya dinyalakan, hal ini akan mereset program counter sehingga perintah program dieksekusi dimulai pada alamat awal. Ketika catu daya diaktifkan, rangkaian reset menahan logika rendah pada pin reset dengan jangka waktu yang ditentukan oleh lamanya pengisian muatan C yaitu membutuhkan waktu t_{TOUT} pada saat tegangan pada V_{cc} mencapai tegangan V_{POT} (*Voltage Power On Threshold*). Tegangan logika rendah pada pin reset antara $-0,5 - 0,2V_{cc}$. Sedangkan resistor *pull-up* yang direkomendasikan sebesar $30\text{ k}\Omega - 60\text{ k}\Omega$. Timing diagram sinyal reset ditunjukkan dalam Gambar 4.6.



Gambar 4.6 Timing Diagram Sinyal Reset
 Sumber : ATMEGA8535, 2004

Rangkaian reset dibentuk dengan rangkaian RC yang dapat dilihat dalam Gambar 4.7.



Gambar 4.7 Rangkaian Reset

Dari rangkaian dan data timing diagram tersebut dapat diperoleh :

$$V_o = \frac{1}{sC_1} V_i$$

$$R_1 + \frac{1}{sC_1}$$

$$V_o = \frac{1}{sR_1C_1 + 1} V_i$$

Dengan tegangan V_i adalah V_{POT} (Voltage Power On Threshold) sebesar 2,3 V dalam fungsi Laplace menjadi $\frac{2,3}{s}$ sehingga :

$$V_o = \frac{2,3}{s} \cdot \frac{1}{sR_1C_1 + 1}$$

$$V_o = 2,3 \frac{1}{s(sR_1C_1 + 1)}$$

$$\mathcal{L}^{-1}[V_o] = 2,3 \mathcal{L}^{-1} \left[\frac{1}{s(sR_1C_1 + 1)} \right]$$

$$\frac{1}{s(sR_1C_1 + 1)} = \frac{X_1}{s} + \frac{X_2}{sR_1C_1 + 1}$$

$$X_1 = \left[\frac{1}{s(sR_1C_1 + 1)} \cdot s \right]_{s=0} = 1$$

$$X_2 = \left[\frac{1}{s(sR_1C_1 + 1)} \cdot (sR_1C_1 + 1) \right]_{s = \frac{1}{R_1C_1}} = -R_1C_1$$

$$\mathcal{L}^{-1}[V_o] = 2,3 \mathcal{L}^{-1} \left[\frac{1}{s} - \frac{R_1C_1}{sR_1C_1 + 1} \right]$$

$$\mathcal{L}^{-1}[V_o] = 2,3 \mathcal{L}^{-1} \left[\frac{1}{s} - \frac{1}{s + \frac{1}{R_1C_1}} \right]$$

$$V_o = 2,3 (1 - e^{-\frac{t}{R_1C_1}})$$

$$\frac{V_o}{2,3} = 1 - e^{-\frac{t}{R_1C_1}} \Leftrightarrow 1 - \frac{V_o}{2,3} = e^{-\frac{t}{R_1C_1}}$$

$$\ln \left(1 - \frac{V_o}{2,3} \right) = \frac{-t}{R_1C_1}$$

$$t = -R_1 C_1 \ln \left(1 - \frac{V_o}{2,3} \right)$$

Dimana V_o adalah tegangan logika rendah pada pin reset maksimal sebesar $0,2V_{cc}$ maka $V_o = 0,2 \times 5 = 1 \text{ V}$.

$$t = -R_1 C_1 \ln \left(1 - \frac{1}{2,3} \right)$$

$$t = 0,571 R_1 C_1$$

Dengan menggunakan R_1 sebesar $30 \text{ k}\Omega$ dan t sebesar 65 ms maka :

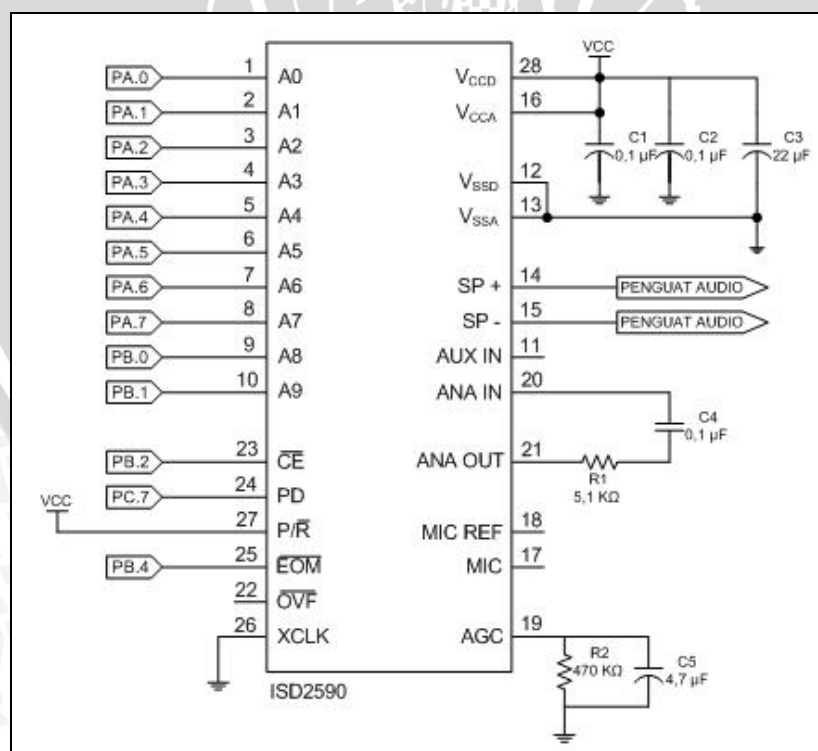
$$65 \times 10^{-3} = 0,571 \times 30 \cdot 10^3 \times C_1$$

$$C_1 = 3,795 \mu\text{F}$$

Berdasarkan nilai yang terdapat di pasaran maka C_1 dipilih dengan menggunakan kapasitor dengan nilai $4,7 \mu\text{F}$.

4.3.5 IC Suara ISD2590

Dalam perancangan alat ini IC Suara ISD2590 digunakan untuk merekam dan memutar kembali suara yang telah direkam. Suara yang direkam dan diputar kembali diantaranya adalah suara adzan, lokasi tempat, dan suara untuk menunjukkan arah. Rangkaian dari IC Suara ISD2590 dapat dilihat dalam Gambar 4.8.



Gambar 4.8 Rangkaian IC Suara ISD2590

IC Suara ISD2590 mempunyai 10 jalur alamat dan 3 buah pin kontrol. Pada perancangan ini pin-pin yang digunakan adalah:

- A0-A9 : Alamat ISD2590 yang dihubungkan ke mikrokontroler ATMEGA8535 port A, port B.0 dan port B.1
- CE : digunakan untuk mengaktifkan ISD2590 dan dihubungkan ke port B.2
- PD : digunakan untuk merekam ke ISD2590 dan dihubungkan ke port C.7
- EOM : digunakan sebagai penanda akhir dari sebuah rekaman pada ISD2590 dan dihubungkan ke port B.4

IC Suara ISD2590 memiliki durasi 90 detik yang terbagi menjadi 600 ruang alamat (alamat 000h sampai dengan alamat 257h). Sehingga setiap ruang alamat memiliki durasi selama 0,15 detik. Dalam perancangan ini, pemilihan alamat untuk merekam suara dalam ISD2590 ditunjukkan dalam Tabel 4.1, 4.2, 4.3, 4.4 dan 4.5.

Tabel 4.1 Pemilihan alamat untuk merekam suara lokasi ke dalam ISD2590

Suara yang direkam	Alamat (heksadesimal)	Durasi (detik)
Banda Aceh	000	0,7
Medan	006	0,3
Pekanbaru	00A	0,7
Padang	00F	0,3
Bengkulu	013	0,4
Jambi	253	0,3
Palembang	017	0,5
Tanjungkarang	01C	0,7
Serang	022	0,3
Jakarta	026	0,5
Bandung	02B	0,4
Cirebon	02F	0,3
Yogyakarta	033	0,6
Semarang	039	0,4
Surakarta	03D	0,6
Malang	043	0,3
Surabaya	048	0,7
Pontianak	050	0,6
Palangkaraya	057	0,7
Balikpapan	05F	0,7
Samarinda	066	0,7
Denpasar	06D	0,6
Lombok	074	0,3
Palu	079	0,3

Manado	07E	0,5
Makassar	084	0,5
Gorontalo	089	0,6
Kupang	091	0,4
Kendari	096	0,4
Ambon	09C	0,4
Jayapura	0A1	0,6
Total durasi		15,3

Tabel 4.2 Pemilihan alamat untuk merekam suara angka ke dalam ISD2590

Suara yang direkam	Alamat (heksadesimal)	Durasi (detik)
Nol	0A8	0,3
Satu	0AD	0,3
Dua	0B2	0,3
Tiga	0B7	0,3
Empat	0BC	0,3
Lima	0C1	0,3
Enam	0C6	0,3
Tujuh	0CA	0,3
Delapan	0CF	0,4
Sembilan	0D4	0,4
Se	0DB	0,2
Puluh	0DF	0,3
Belas	0E4	0,3
Ratus	0E8	0,4
Total durasi		4,4

Tabel 4.3 Pemilihan alamat untuk merekam suara pilih-derajat ke dalam ISD2590

Suara yang direkam	Alamat (heksadesimal)	Durasi (detik)
Pilih	0EF	0,3
Lokasi	0F3	0,5
Dipilih	0F9	0,5
Derajat	123	0,4
Kiblat	105	0,3
Sudah	10A	0,4
Tepat	10F	0,4
Total durasi		2,8

Tabel 4.4 Pemilihan alamat untuk merekam suara arah ke dalam ISD2590

Suara yang direkam	Alamat (heksadesimal)	Durasi (detik)
Arah	100	0,4
Kanan	119	0,3
Kiri	11E	0,3
Ke	114	0,2
Total durasi		1,2

Tabel 4.5 Pemilihan alamat untuk merekam suara adzan ke dalam ISD2590

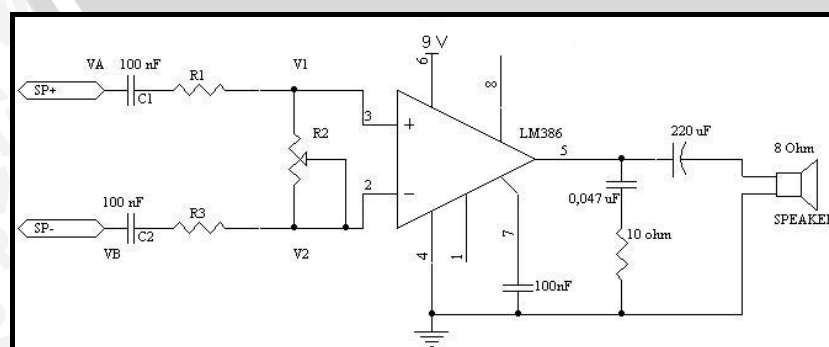
Suara yang direkam	Alamat (heksadesimal)	Durasi (detik)
Adzan	13D	27,9
Mode 1	1FD	6,2
Mode 2	22B	6
Total durasi		40,1

Tabel 4.6 Pemilihan alamat untuk merekam suara pukul-bulan ke dalam ISD2590

Suara yang direkam	Alamat (heksadesimal)	Durasi (detik)
Pukul	128	0,5
Menit	12E	0,3
Tanggal	133	0,3
Bulan	138	0,3
Total durasi		1,4

4.3.6 Penguat Audio

Penguat Audio digunakan untuk menguatkan tegangan keluaran dari ISD2590 sehingga suara yang dihasilkan menjadi lebih keras. Komponen utama dari rangkaian penguat audio ini adalah IC LM386. Adapun rangkaian dari penguat audio ini dapat ditunjukkan dalam Gambar 4.9.

**Gambar 4.9** Rangkaian penguat audio

Dari gambar di atas maka dapat diketahui besarnya tegangan pada V_1 dan V_2 yaitu sebesar:

$$V_1 = \frac{Z_2 + R_2}{Z_1 + Z_2 + R_2} (V_A - V_B) + V_B$$

$$V_2 = \frac{Z_2}{Z_1 + Z_2 + R_2} (V_A - V_B) + V_B$$

Dengan:

$$Z_1 = R_1 + \frac{1}{sC_1}$$

$$Z_2 = R_3 + \frac{1}{sC_2}$$

Untuk mendapatkan ΔV digunakan persamaan:

$$\Delta V = V_1 - V_2$$

$$V_1 - V_2 = \frac{R_2}{Z_1 + Z_2 + R_2} (V_A - V_B)$$

Sehingga:

$$V_o = A_v (V_1 - V_2) = A_v \left(\frac{R_2}{Z_1 + Z_2 + R_2} (V_A - V_B) \right)$$

Karena beda tegangan keluaran pin SP+ dan SP- dari ISD2590 adalah 2,5 Vpp maka nilai dari $(V_A - V_B)$ adalah 2,5 Vpp. Untuk rangkaian di atas maka nilai A_v dari IC LM386 adalah 12 kali. Jika diinginkan tegangan keluaran dari LM386 (V_o) = 5Vpp dan $R_1 = R_3 = 560 \text{ K}\Omega$, $C_1 = C_2 = 100\text{nF}$. Range frekuensi suara manusia pada saat berbicara umumnya 200 Hz – 5000 Hz, jika $f = 1000 \text{ Hz}$, maka nilai R_2 adalah:

$$5V_{pp} = 12 \frac{R_2}{562K + R_2 + 562K} 2,5V_{pp}$$

$$R_2 = 224800 \approx 200K\Omega$$

Dalam rangkaian penguat audio ini digunakan C_1 dan C_2 sebagai kapasitor kopling yang berfungsi untuk mengubah sinyal DC tidak murni menjadi sinyal AC.

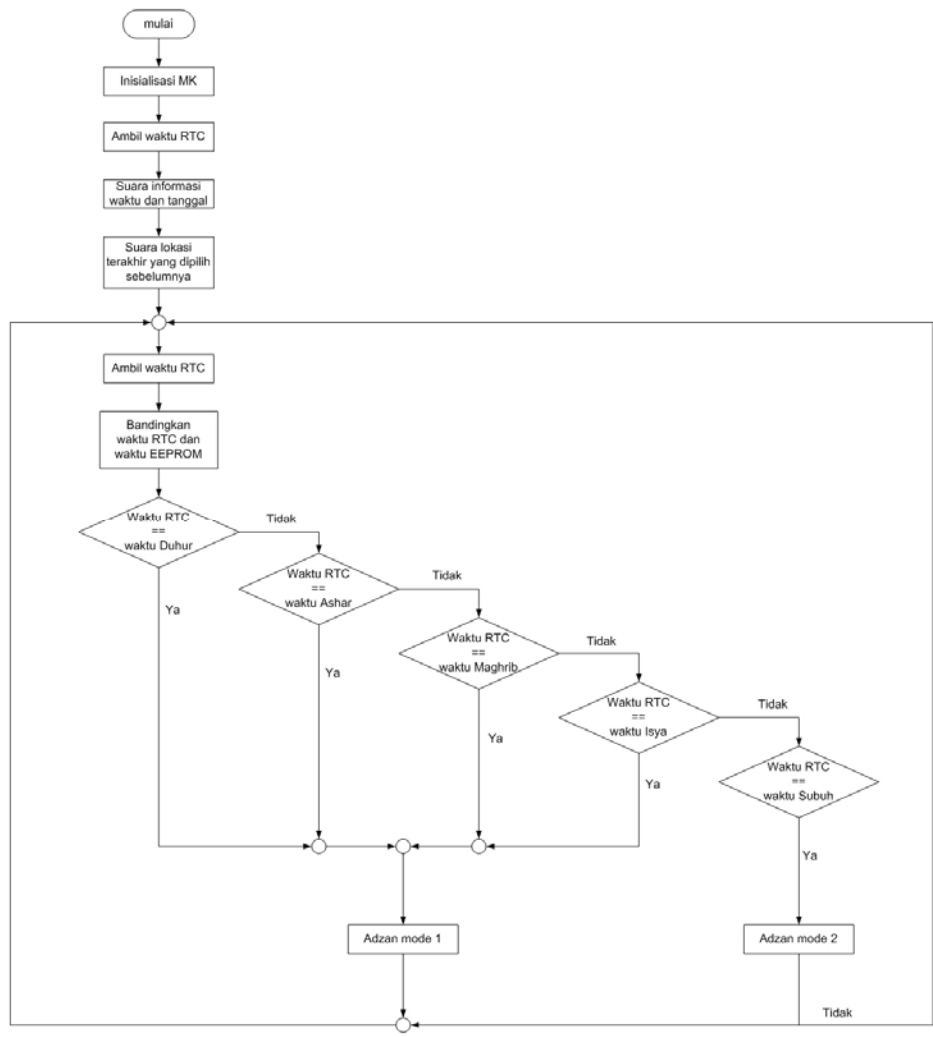
4.4 Perancangan Perangkat Lunak

Perangkat lunak yang dirancang untuk mikrokontroler harus mampu menangani pengambilan data dari sensor magnet, mampu mengambil informasi waktu dari RTC, mampu membaca adanya penekanan tombol lokasi ataupun tombol start, mampu

membandingkan informasi waktu yang didapatkan dari RTC dengan data waktu yang tersimpan dalam EEPROM sehingga dapat diketahui waktu sholat, mampu mengolah data yang didapatkan dari sensor magnet sehingga arah kiblat dari suatu tempat dapat diketahui, dan juga mampu mengambil data suara dari IC suara ISD2590 sehingga suara tersebut dapat dikeluarkan melalui *speaker*.

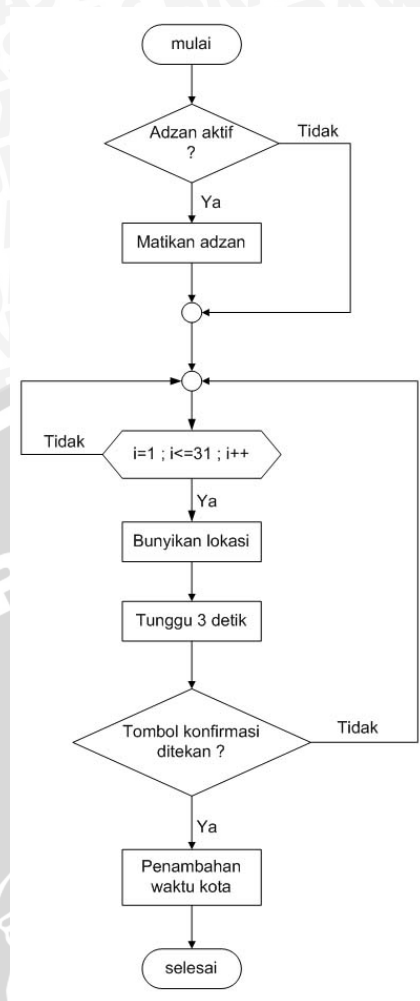
Sistem akan mengambil data waktu dari *real time clock*, dan suara adzan akan dikeluarkan jika sudah masuk waktu sholat. Apabila tombol lokasi ditekan, maka akan dikeluarkan suara pilihan lokasi kota. Pengguna dapat memilih lokasi kota tersebut dengan menekan tombol konfirmasi. Jika tombol konfirmasi tidak ditekan sampai dengan suara pilihan lokasi kota dikeluarkan semua, maka lokasi kota yang digunakan adalah pilihan lokasi kota yang sebelumnya atau jika belum ada penekanan tombol lokasi sama sekali maka lokasi kota yang digunakan adalah Malang sebagai lokasi kota *default*. Untuk mengetahui arah kiblat, pengguna dapat menekan tombol kiblat. Kemudian program akan mengolah data yang didapatkan dari sensor magnet, dan hasilnya akan dikeluarkan sebagai suara panduan arah kiblat. Diagram alir dari program utama ditunjukkan dalam Gambar 4.10.

Penjelasan diagram alir dari program utama adalah sebagai berikut: Inisialisasi mikrokontroler dilakukan pertama kali ketika program dijalankan dan dilakukan reset. Kemudian data waktu dari RTC diambil yaitu berupa data detik, menit, jam, hari, dan bulan. Data yang telah dibaca dikeluarkan melalui speaker berupa informasi jam, menit, tanggal dan bulan. Setelah itu dilakukan pengambilan waktu dari RTC kembali yang hasil pembacaannya dibandingkan dengan data waktu yang terdapat di EEPROM. Kemudian dilakukan pengecekan apakah data waktu di RTC sesuai dengan waktu Duhur, jika ya maka suara adzan mode 1 diaktifkan. Jika tidak maka selanjutnya dilakukan pengecekan waktu Ashar. Begitu seterusnya sampai dengan pengecekan waktu Subuh. Perbedaannya adalah untuk waktu Duhur, Ashar, Maghrib dan Isya suara adzan yang diaktifkan adalah suara adzan mode 1. Sedangkan untuk waktu Subuh suara adzan yang diaktifkan adalah suara adzan mode 2. Jika data waktu RTC tidak ada yang sesuai dengan waktu-waktu sholat maka program akan berulang lagi dari mulai pengambilan data waktu dari RTC. Ini juga dilakukan jika suara adzan telah dikeluarkan.



Gambar 4.10 Diagram alir program utama

Untuk pemilihan lokasi kota dan penekanan arah kiblat, digunakan fasilitas interupsi eksternal dari mikrokontroler. Diagram alir program interupsi eksternal lokasi ditunjukkan dalam Gambar 4.11. Penjelasan dari program interupsi eksternal lokasi adalah: dilakukan pengecekan apakah suara adzan sedang dijalankan, jika ya maka suara adzan akan dimatikan. Kemudian dilakukan perulangan sebanyak 31 kali yang merupakan jumlah dari lokasi kota yang disimpan dalam EEPROM. Dalam perulangan ini, dikeluarkan suara pilihan lokasi kota dan disediakan waktu selama 3 detik bagi pengguna untuk memilih tombol konfirmasi atau tidak. Jika ya, maka dilakukan proses penambahan waktu sholat sesuai dengan lokasi kota yang diinginkan. Jika tidak, suara pilihan lokasi kota akan dilanjutkan sampai suara pilihan kota yang terakhir.



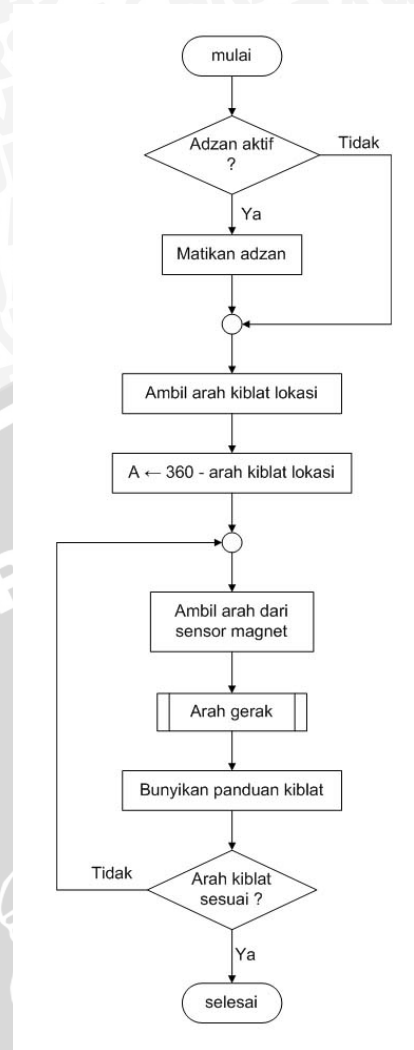
Gambar 4.11 Diagram alir program interupsi eksternal lokasi

Diagram alir untuk program interupsi eksternal kiblat ditunjukkan dalam Gambar 4.12. Penjelasan untuk program interupsi eksternal kiblat adalah: pengecekan suara adzan apakah aktif, jika ya maka suara adzan akan dimatikan. Kemudian arah kiblat lokasi yang disimpan dalam EEPROM diambil. Untuk memudahkan perhitungan, maka dilakukan proses:

$$A = 360 - \text{arah kiblat lokasi}$$

Dengan: A = variabel bantu untuk menyimpan hasil perhitungan

Selanjutnya dilakukan pengambilan data arah dari hasil pendeteksian sensor magnet. Kemudian dilakukan program arah gerak, yang diagram alirnya ditunjukkan dalam Gambar 4.13. Setelah program arah gerak dijalankan, maka suara panduan arah kiblat akan dikeluarkan sampai arah kiblat sesuai.



Gambar 4.12 Diagram alir program interupsi eksternal kiblat

Penjelasan untuk program arah gerak adalah sebagai berikut: dilakukan proses perhitungan yang merupakan kelanjutan dari perhitungan pada program interupsi eksternal kiblat. Untuk itu digunakan perhitungan:

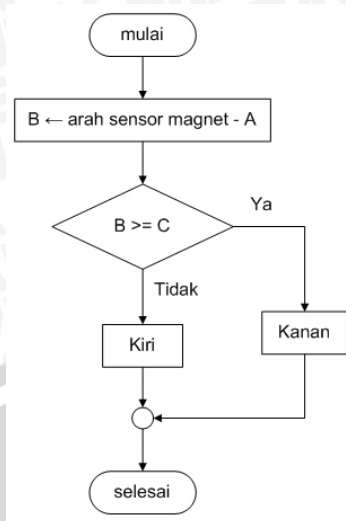
$$|B| = \text{data arah dari sensor magnet} - A$$

Dengan: A = hasil perhitungan pada program interupsi eksternal kiblat

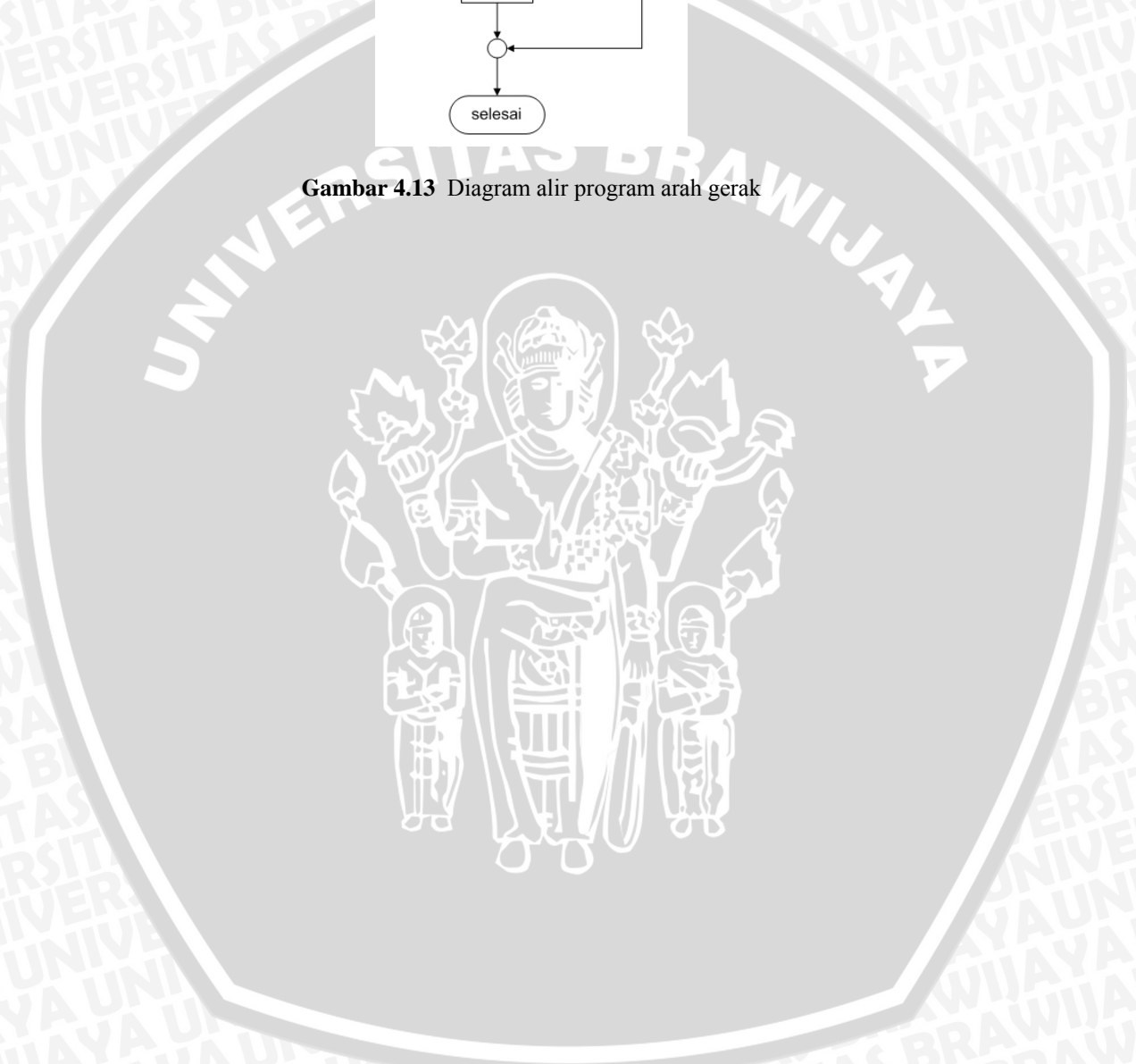
B = variabel bantu untuk menyimpan hasil perhitungan

Selanjutnya dilakukan pengecekan apakah $B \geq C$, jika ya maka akan dikeluarkan suara kanan. Sedangkan jika tidak, maka akan dikeluarkan suara kiri.

Dengan: C = arah kiblat lokasi – 180°



Gambar 4.13 Diagram alir program arah gerak



BAB V

PENGUJIAN DAN ANALISIS

Pengujian dan analisis dilakukan untuk mengetahui apakah sistem yang telah dirancang dan dibuat bekerja sesuai dengan yang diharapkan. Pengujian ini dibagi dua yaitu pengujian perangkat keras dan perangkat lunak. Dalam pengujian perangkat keras, dilakukan pengujian tiap-tiap blok untuk memudahkan analisis. Setelah dilakukan pengujian perangkat keras, maka selanjutnya dilakukan pengujian perangkat lunak. Jika pengujian perangkat keras dan perangkat lunak telah selesai dilakukan, maka dilakukan pengujian sistem secara keseluruhan.

Pengujian yang dilakukan adalah sebagai berikut:

1. Pengujian sensor magnet CMPS03
2. Pengujian RTC DS1302
3. Pengujian EEPROM AT24C64
4. Pengujian IC suara ISD2590
5. Pengujian rangkaian penguat audio
6. Pengujian perangkat lunak untuk mengetahui waktu sholat
7. Pengujian perangkat lunak untuk perhitungan arah kiblat
8. Pengujian sistem secara keseluruhan

5.1 Pengujian Sensor Magnet CMPS03

5.1.1 Tujuan

Tujuan pengujian ini adalah untuk mengetahui apakah sensor magnet CMPS03 dapat bekerja dengan baik dalam mendeteksi arah-arah mata angin (utara, timur, selatan, dan barat) dan menghasilkan keluaran berupa nilai-nilai derajat.

5.1.2 Peralatan

Peralatan yang digunakan dalam pengujian sensor magnet CMPS03 adalah:

1. Minimum sistem mikrokontroler ATMEGA8535
2. Modul sensor magnet CMPS03
3. Modul LCD
4. Kompas



5.1.3 Prosedur Pengujian

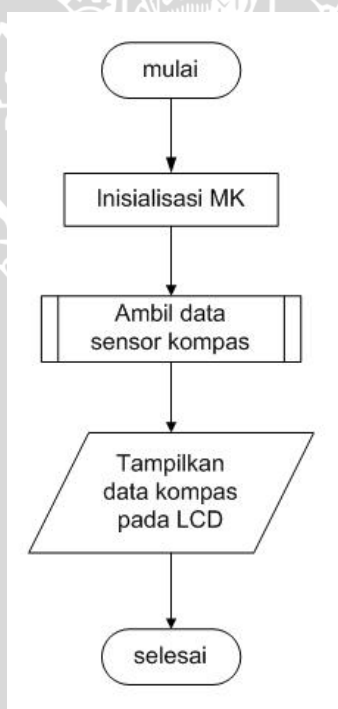
Prosedur pengujian sensor magnet CMPS03 yang dilakukan adalah:

1. Menyusun rangkaian seperti dalam Gambar 5.1
2. Memprogram mikrokontroler ATMEGA8535 untuk dapat membaca keluaran dari sensor magnet CMPS03 dan dikeluarkan melalui port D
3. Mengamati hasil keluaran pada LCD



Gambar 5.1 Diagram Blok Pengujian Sensor Magnet CPMS03

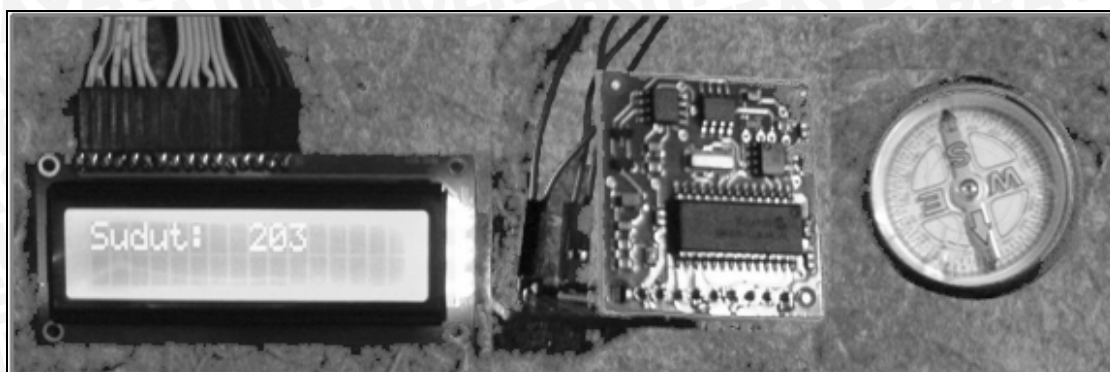
Diagram alir program pengujian sensor magnet CMPS03 ditunjukkan dalam Gambar 5.2.



Gambar 5.2 Diagram Alir Pengujian Sensor Magnet CMPS03

5.1.4 Hasil

Data hasil pengujian sensor magnet CMPS03 dapat dilihat dalam Gambar 5.3.



Gambar 5.3 Hasil Pengujian Sensor Magnet CMPS03

5.1.5 Analisis dan Kesimpulan

Dari hasil pengujian sensor magnet CMPS03 didapatkan keluaran berupa nilai-nilai derajat dengan arah utara sebagai acuan titik 0° dan data keluaran sensor magnet CMPS03 sesuai dengan nilai-nilai derajat dalam kompas. Dengan demikian dapat disimpulkan bahwa sensor magnet CMPS03 dapat berfungsi dengan baik sesuai dengan perancangan.

5.2 Pengujian RTC DS1302

5.2.1 Tujuan

Tujuan pengujian ini adalah untuk mengetahui apakah RTC DS1302 dapat bekerja dengan baik dan dapat memberikan informasi yang diinginkan berupa informasi jam, menit, detik, tanggal, bulan maupun tahun.

5.2.2 Peralatan

Peralatan yang digunakan dalam pengujian RTC DS1302 adalah:

1. Minimum sistem mikrokontroler ATMEGA8535
2. Rangkaian RTC DS1302
3. Delapan buah LED
4. *Stopwatch*

5.2.3 Prosedur Pengujian

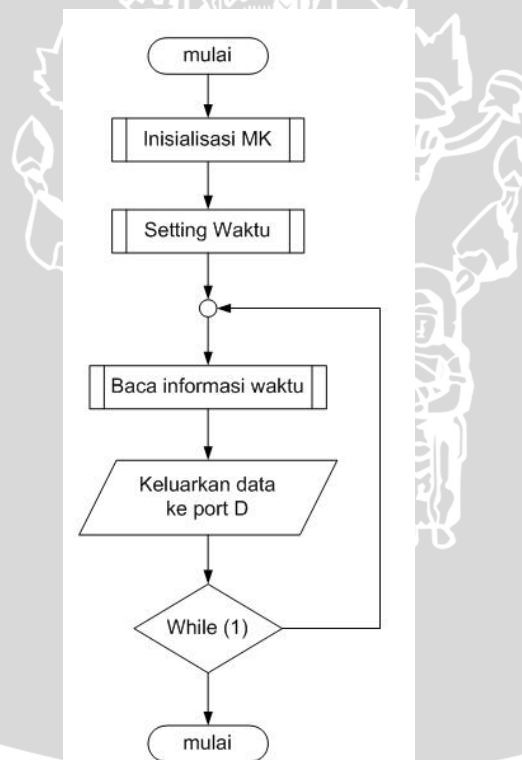
Prosedur pengujian RTC DS1302 yang dilakukan adalah:

1. Menyusun rangkaian pengujian seperti dalam Gambar 5.4
2. Memprogram mikrokontroler ATMEGA8535 sesuai dengan program pengujian untuk mengeluarkan informasi detik, menit, jam, hari, tanggal, dan tahun ke port D
3. Mengamati perubahan tiap waktu melalui nyala LED



Gambar 5.4 Diagram Blok Pengujian RTC DS1302

Diagram alir program pengujian RTC DS1302 ditunjukkan dalam Gambar 5.5.



Gambar 5.5 Diagram Alir Program Pengujian RTC DS1302

5.2.4 Hasil

Data hasil pengujian RTC DS1302 dapat dilihat dalam Tabel 5.1, 5.2, 5.3, 5.4, 5.5 dan 5.6.

Tabel 5.1 Hasil Pengujian RTC DS1302 Untuk Data Detik

Langkah	LED								Data (Detik)
	8	7	6	5	4	3	2	1	
1	OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF	0
2	OFF	OFF	OFF	OFF	OFF	OFF	OFF	ON	1
3	OFF	OFF	OFF	OFF	OFF	OFF	ON	OFF	2
4	OFF	OFF	OFF	OFF	OFF	OFF	ON	ON	3
5	OFF	OFF	OFF	OFF	OFF	ON	OFF	OFF	4
6	OFF	OFF	OFF	OFF	OFF	ON	OFF	ON	5
7	OFF	OFF	OFF	OFF	OFF	ON	ON	OFF	6
8	OFF	OFF	OFF	OFF	OFF	ON	ON	ON	7
9	OFF	OFF	OFF	OFF	ON	OFF	OFF	OFF	8
10	OFF	OFF	OFF	OFF	ON	OFF	OFF	ON	9
11	OFF	OFF	OFF	OFF	ON	OFF	ON	OFF	10
12	OFF	OFF	OFF	OFF	ON	OFF	ON	ON	11
13	OFF	OFF	OFF	OFF	ON	ON	OFF	OFF	12
14	OFF	OFF	OFF	OFF	ON	ON	OFF	ON	13
15	OFF	OFF	OFF	OFF	ON	ON	ON	OFF	14
16	OFF	OFF	OFF	OFF	ON	ON	ON	ON	15
17	OFF	OFF	OFF	ON	OFF	OFF	OFF	OFF	16
18	OFF	OFF	OFF	ON	OFF	OFF	OFF	ON	17
19	OFF	OFF	OFF	ON	OFF	OFF	ON	OFF	18
20	OFF	OFF	OFF	ON	OFF	OFF	ON	ON	19
21	OFF	OFF	OFF	ON	OFF	ON	OFF	OFF	20
22	OFF	OFF	OFF	ON	OFF	ON	OFF	ON	21
23	OFF	OFF	OFF	ON	OFF	ON	ON	OFF	22
24	OFF	OFF	OFF	ON	OFF	ON	ON	ON	23
25	OFF	OFF	OFF	ON	ON	OFF	OFF	OFF	24
26	OFF	OFF	OFF	ON	ON	OFF	OFF	ON	25
27	OFF	OFF	OFF	ON	ON	OFF	ON	OFF	26
28	OFF	OFF	OFF	ON	ON	OFF	ON	ON	27
29	OFF	OFF	OFF	ON	ON	ON	OFF	OFF	28
30	OFF	OFF	OFF	ON	ON	ON	OFF	ON	29
31	OFF	OFF	OFF	ON	ON	ON	ON	OFF	30
32	OFF	OFF	OFF	ON	ON	ON	ON	ON	31
33	OFF	OFF	ON	OFF	OFF	OFF	OFF	OFF	32
34	OFF	OFF	ON	OFF	OFF	OFF	OFF	ON	33
35	OFF	OFF	ON	OFF	OFF	OFF	ON	OFF	34
36	OFF	OFF	ON	OFF	OFF	OFF	ON	ON	35
37	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF	36
38	OFF	OFF	ON	OFF	OFF	ON	OFF	ON	37
39	OFF	OFF	ON	OFF	OFF	ON	ON	OFF	38

40	OFF	OFF	ON	OFF	OFF	ON	ON	ON	39
41	OFF	OFF	ON	OFF	ON	OFF	OFF	OFF	40
42	OFF	OFF	ON	OFF	ON	OFF	OFF	ON	41
43	OFF	OFF	ON	OFF	ON	OFF	ON	OFF	42
44	OFF	OFF	ON	OFF	ON	OFF	ON	ON	43
45	OFF	OFF	ON	OFF	ON	ON	OFF	OFF	44
46	OFF	OFF	ON	OFF	ON	ON	OFF	ON	45
47	OFF	OFF	ON	OFF	ON	ON	ON	OFF	46
48	OFF	OFF	ON	OFF	ON	ON	ON	ON	47
49	OFF	OFF	ON	ON	OFF	OFF	OFF	OFF	48
50	OFF	OFF	ON	ON	OFF	OFF	OFF	ON	49
51	OFF	OFF	ON	ON	OFF	OFF	ON	OFF	50
52	OFF	OFF	ON	ON	OFF	OFF	ON	ON	51
53	OFF	OFF	ON	ON	OFF	ON	OFF	OFF	52
54	OFF	OFF	ON	ON	OFF	ON	OFF	ON	53
55	OFF	OFF	ON	ON	OFF	ON	ON	OFF	54
56	OFF	OFF	ON	ON	OFF	ON	ON	ON	55
57	OFF	OFF	ON	ON	ON	OFF	OFF	OFF	56
58	OFF	OFF	ON	ON	ON	OFF	OFF	ON	57
59	OFF	OFF	ON	ON	ON	OFF	ON	OFF	58
60	OFF	OFF	ON	ON	ON	OFF	ON	ON	59
61	OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF	0

Tabel 5.2 Hasil Pengujian RTC DS1302 Untuk Data Menit

Langkah	LED								Data (Menit)
	8	7	6	5	4	3	2	1	
1	OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF	0
2	OFF	OFF	OFF	OFF	OFF	OFF	OFF	ON	1
3	OFF	OFF	OFF	OFF	OFF	OFF	ON	OFF	2
4	OFF	OFF	OFF	OFF	OFF	OFF	ON	ON	3
5	OFF	OFF	OFF	OFF	OFF	ON	OFF	OFF	4

Tabel 5.3 Hasil Pengujian RTC DS1302 Untuk Data Jam

Langkah	LED								Data (Jam)
	8	7	6	5	4	3	2	1	
1	OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF	0
2	OFF	OFF	OFF	OFF	OFF	OFF	OFF	ON	1
3	OFF	OFF	OFF	OFF	OFF	OFF	ON	OFF	2

Tabel 5.4 Hasil Pengujian RTC DS1302 Untuk Data Hari

Langkah	LED								Data (Hari)
	8	7	6	5	4	3	2	1	
1	OFF	OFF	OFF	ON	ON	ON	ON	OFF	30
2	OFF	OFF	OFF	ON	ON	ON	ON	ON	31

Tabel 5.5 Hasil Pengujian RTC DS1302 Untuk Data Bulan

Langkah	LED								Data (Bulan)
	8	7	6	5	4	3	2	1	
1	OFF	OFF	OFF	OFF	OFF	OFF	OFF	ON	1
2	OFF	OFF	OFF	OFF	OFF	OFF	ON	OFF	2

Tabel 5.6 Hasil Pengujian RTC DS1302 Untuk Data Tahun

Langkah	LED								Data (Tahun)
	8	7	6	5	4	3	2	1	
1	OFF	OFF	OFF	OFF	OFF	OFF	OFF	ON	1

Dimana kondisi ON = 4,61 volt dan kondisi OFF = 0,02 volt

5.2.5 Analisis dan Kesimpulan

Dari hasil pengujian, diperoleh kesimpulan bahwa rangkaian RTC DS1302 dapat berfungsi dengan baik. Data detik, menit, jam, hari, bulan dan tahun dapat ditampilkan pada LED sesuai dengan waktu awal yang ditetapkan terlebih dahulu dalam program pengujian. Perubahan data detik dan menit sesuai dengan perubahan detik dan menit pada *stopwatch*.

5.3 Pengujian EEPROM AT24C64

5.3.1 Tujuan

Tujuan pengujian ini adalah untuk mengetahui apakah EEPROM AT24C64 dapat ditulis dan dibaca kembali dengan baik.

5.3.2 Peralatan

Peralatan yang digunakan dalam pengujian EEPROM AT24C64 adalah:

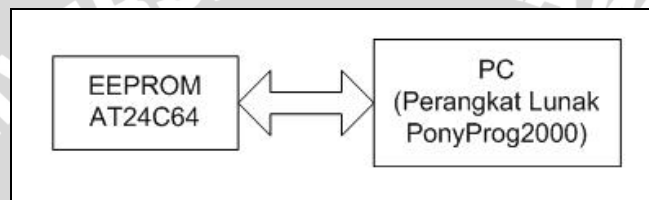
1. Minimum sistem mikrokontroler ATMEGA8535
2. Rangkaian EEPROM AT24C64
3. Delapan buah LED

4. Perangkat lunak PonyProg2000

5.3.3 Prosedur Pengujian

Prosedur pengujian EEPROM AT24C64 yang dilakukan adalah:

1. Menulis data ke EEPROM AT24C64
 - a. Menyusun rangkaian pengujian seperti dalam Gambar 5.6
 - b. Melakukan proses penulisan data menggunakan perangkat lunak PonyProg2000
 - c. Melakukan proses baca dari EEPROM AT24C64 menggunakan program PonyProg2000



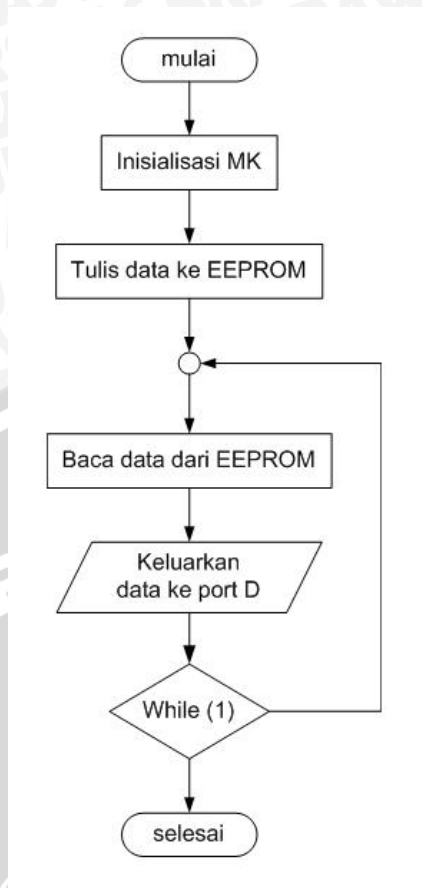
Gambar 5.6 Diagram Blok Penulisan Data ke EEPROM AT24C64

2. Membaca data dari EEPROM AT24C64
 - a. Menyusun rangkaian pengujian seperti dalam Gambar 5.7
 - b. Memprogram mikrokontroler ATMEGA8535 untuk dapat membaca data dari EEPROM AT24C64 dan dikeluarkan melalui port D
 - c. Mengamati keluaran data melalui nyala LED



Gambar 5.7 Diagram Blok Pembacaan Data dari EEPROM AT24C64

Diagram alir proses pembacaan data dari EEPROM AT24C64 dapat dilihat dalam Gambar 5.8.



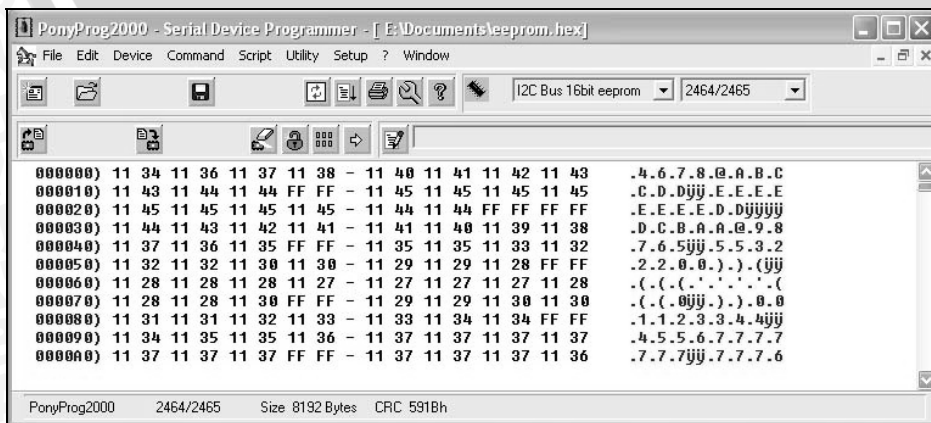
Gambar 5.8 Diagram Alir Pembacaan Data dari EEPROM AT24C64

5.3.4 Hasil

Data hasil pengujian EEPROM AT24C64 adalah:

1. Penulisan data EEPROM AT24C64

Penulisan data dilakukan menggunakan *software* PonyProg2000. Gambar 5.9 menunjukkan hasil pengujian penulisan data ke EEPROM AT24C64.



Gambar 5.9 Hasil pengujian penulisan data ke EEPROM AT24C64



2. Pembacaan data EEPROM AT24C64

Hasil pengujian pembacaan data EEPROM AT24C64 ditampilkan melalui LED. Hasil pengujian ini ditunjukkan dalam Tabel 5.7.

Tabel 5.7 Hasil pengujian pembacaan data EEPROM AT24C64

Data yang ditulis	Alamat (heksadesimal)	Data hasil pembacaan
11	00	11
34	01	34
43	11	43
45	1F	45
11	20	11
45	27	45
32	51	32
29	59	29
11	60	11
14	1B0	14
00	1BF	00
52	241	52
19	390	19
51	469	51

5.3.5 Analisis dan Kesimpulan

Dari hasil pengujian dapat diambil kesimpulan bahwa EEPROM AT24C64 dapat ditulis dan dibaca kembali sesuai dengan perancangan.

5.4 Pengujian IC Suara ISD2590

5.4.1 Tujuan

Tujuan pengujian ini adalah untuk mengetahui apakah ISD2590 dapat melakukan proses perekaman dan memutar ulang hasil perekaman sesuai dengan perancangan.

5.4.2 Peralatan

Peralatan yang digunakan dalam pengujian ISD2590 adalah:

1. ISD2590
2. *Writer* ISD25xx

5.4.3 Prosedur Pengujian

Prosedur pengujian ISD2590 yang dilakukan adalah:

1. Melakukan proses *record* dan *play back* menggunakan *writer* ISD25xx
2. Mendengarkan hasil rekaman melalui *speaker*

5.4.4 Hasil

Dari hasil pengujian *record* dan *play back* menggunakan *writer* ISD25xx diperoleh hasil seperti dalam Tabel 5.8.

Tabel 5.8 Hasil Pengujian *Record* dan *Play Back* ISD2590

Suara yang direkam	Alamat (heksadesimal)	Suara keluaran <i>speaker</i>	Durasi
Banda Aceh	000	Banda Aceh	0,7
Medan	006	Medan	0,3
Pekanbaru	00A	Pekanbaru	0,7
Padang	00F	Padang	0,3
Bengkulu	013	Bengkulu	0,4
Jambi	253	Jambi	0,3
Palembang	017	Palembang	0,5
Tanjungkarang	01C	Tanjungkarang	0,7
Serang	022	Serang	0,3
Jakarta	026	Jakarta	0,5
Bandung	02B	Bandung	0,4
Cirebon	02F	Cirebon	0,3
Yogyakarta	033	Yogyakarta	0,6
Semarang	039	Semarang	0,4
Surakarta	03D	Surakarta	0,6
Malang	043	Malang	0,3
Surabaya	048	Surabaya	0,7
Pontianak	050	Pontianak	0,6
Palangkaraya	057	Palangkaraya	0,7
Balikpapan	05F	Balikpapan	0,7
Samarinda	066	Samarinda	0,7
Denpasar	06D	Denpasar	0,6
Lombok	074	Lombok	0,3
Palu	079	Palu	0,3
Manado	07E	Manado	0,5
Makassar	084	Makassar	0,5
Gorontalo	089	Gorontalo	0,6
Kupang	091	Kupang	0,4

Kendari	096	Kendari	0,4
Ambon	09C	Ambon	0,4
Jayapura	0A1	Jayapura	0,6
Nol	0A8	Nol	0,3
Satu	0AD	Satu	0,3
Dua	0B2	Dua	0,3
Tiga	0B7	Tiga	0,3
Empat	0BC	Empat	0,3
Lima	0C1	Lima	0,3
Enam	0C6	Enam	0,3
Tujuh	0CA	Tujuh	0,3
Delapan	0CF	Delapan	0,4
Sembilan	0D4	Sembilan	0,4
Se	0DB	Se	0,2
Puluh	0DF	Puluh	0,3
Belas	0E4	Belas	0,3
Ratus	0E8	Ratus	0,4
Pilih	0EF	Pilih	0,3
Lokasi	1F3	Lokasi	0,5
Dipilih	0F9	Dipilih	0,5
Derajat	123	Derajat	0,4
Kiblat	105	Kiblat	0,3
Sudah	10A	Sudah	0,4
Tepat	10F	Tepat	0,4
Arah	100	Arah	0,4
Kanan	119	Kanan	0,3
Kiri	11E	Kiri	0,3
Ke	114	Ke	0,2
Adzan	13D	Adzan	27,9
Mode 1	1FD	Mode 1	6,2
Mode 2	22B	Mode 2	6
Pukul	128	Pukul	0,5
Menit	12E	Menit	0,3
Tanggal	133	Tanggal	0,3
Bulan	138	Bulan	0,3
Total Durasi			65,2

5.4.5 Analisis dan Kesimpulan

Dari hasil pengujian *record* dan *play back* ISD2590 menggunakan *writer* ISD25xx dapat disimpulkan bahwa ISD2590 dapat merekam dan memutar ulang hasil perekaman sesuai dengan yang direncanakan.

5.5 Pengujian Rangkaian Penguat Audio

5.5.1 Tujuan

Pengujian rangkaian penguat audio dilakukan untuk mengetahui apakah rangkaian penguat audio dapat menguatkan sinyal keluaran IC suara ISD2590 sesuai dengan yang direncanakan. Dalam pengujian ini, sinyal yang digunakan sebagai masukan rangkaian penguat audio berasal dari *function generator*.

5.5.2 Peralatan

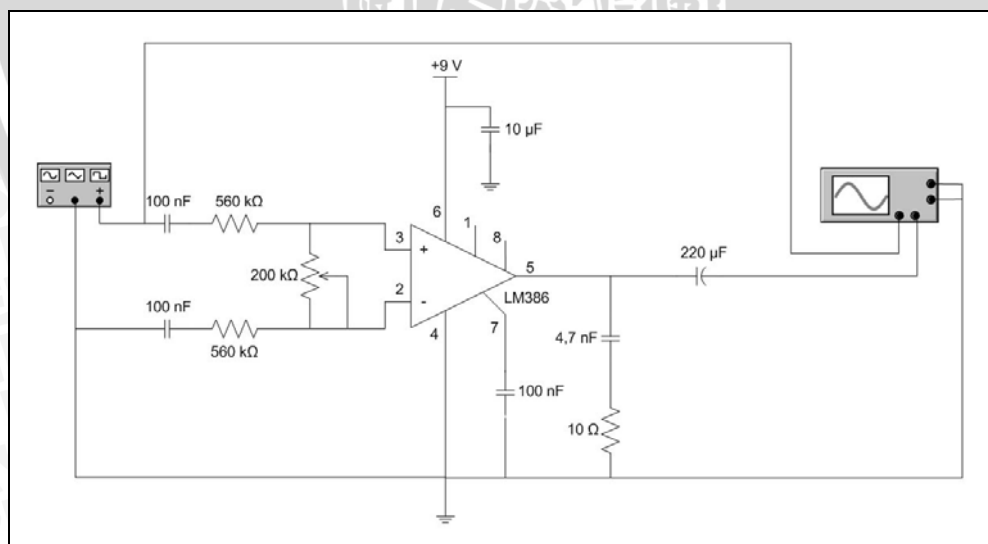
Peralatan yang digunakan dalam pengujian rangkaian penguat audio adalah:

1. *Function generator*
2. Rangkaian penguat audio
3. *Oscilloscop*

5.5.3 Prosedur Pengujian

Prosedur pengujian rangkaian penguat audio yang dilakukan adalah:

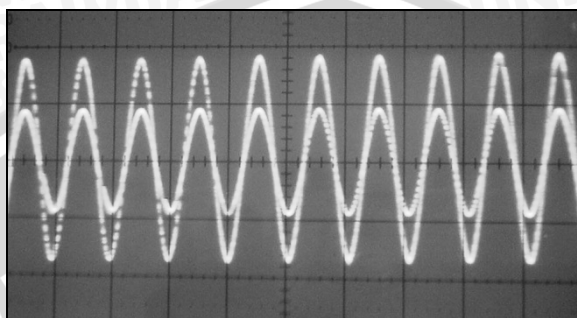
1. Menyusun rangkaian seperti dalam Gambar 5.10
2. Mengamati keluaran dari rangkaian penguat audio melalui *oscilloscop* dan mencatat tegangan keluaran dari *oscilloscop* dalam V_{p-p} .



Gambar 5.10 Rangkaian Pengujian Rangkaian Penguat Audio

5.5.4 Hasil

Dalam pengujian rangkaian penguat audio, *channel 1* dan *channel 2 osciloscop* diatur pada 0,5 mV/div dan time/div-nya diatur pada 1 ms/div. Hasil dari pengujian dapat dilihat dalam Tabel 5.9 dan Tabel 5.10. Tampilan pada *osciloscop* untuk pengujian rangkaian penguat audio dapat dilihat dalam Gambar 5.11.



Gambar 5.11 Tampilan *Osciloscop* Pengujian Penguat Audio

Tabel 5.9 Data Hasil Pengujian Rangkaian Penguat Audio Dengan Nilai Frekuensi Berbeda

V_{in} (V_{p-p})	Frekuensi (Hz)	V_{out} Teori (V_{p-p})	V_{out} Praktek (V_{p-p})	Persen Kesalahan (%)
1	500	1,809455	1,8	0,5225330
1	1000	1,813807	1,8	0,7612166
1	1500	1,815264	1,8	0,8408694
1	2000	1,815992	1,8	0,8806206
1	2500	1,816429	1,8	0,9044669
1	3000	1,816720	1,8	0,9203399
1	3500	1,816929	1,8	0,9317370
1	4000	1,817086	1,8	0,9402967
1	4500	1,817207	1,8	0,4689268
Persen Kesalahan Rata-Rata (%)				0,8498859

Tabel 5.10 Data Hasil Pengujian Rangkaian Penguat Audio Dengan Nilai Frekuensi Tetap

V_{in} (V_{p-p})	Frekuensi (Hz)	V_{out} Teori (V_{p-p})	V_{out} Praktek (V_{p-p})	Persen Kesalahan (%)
0,5	1000	0,906903	0,9	0,7611619
0,75	1000	1,360355	1,4	2,9143128
1	1000	1,813807	1,8	0,7612166
1,25	1000	2,267258	2,3	1,4441233
1,5	1000	2,720710	2,7	0,7611984
1,75	1000	3,174162	3,2	0,8140101
2	1000	3,627613	3,6	0,7611892
2,25	1000	4,081065	4	1,9863687
2,5	1000	4,534517	4,5	0,7612057
Persen Kesalahan Rata-Rata (%)				1,2183096

5.5.5 Analisis dan Kesimpulan

Dari data hasil pengujian dalam Tabel 5.9 dan 5.10 dapat diketahui bahwa persen kesalahan rata-rata untuk pengujian rangkaian penguat audio dengan frekuensi tetap adalah 0,85% dan untuk pengujian rangkaian penguat audio dengan frekuensi tetap adalah 1,21%. Dengan demikian dapat disimpulkan bahwa rangkaian penguat audio sudah dapat berfungsi sesuai dengan perancangan karena kesalahan yang terjadi tidak berpengaruh besar pada sistem.

5.6 Pengujian Perangkat Lunak Untuk Mengetahui Waktu Sholat

5.6.1 Tujuan

Tujuan pengujian ini adalah untuk mengetahui apakah program yang dibuat sudah dapat bekerja sesuai dengan yang diharapkan.

5.6.2 Peralatan

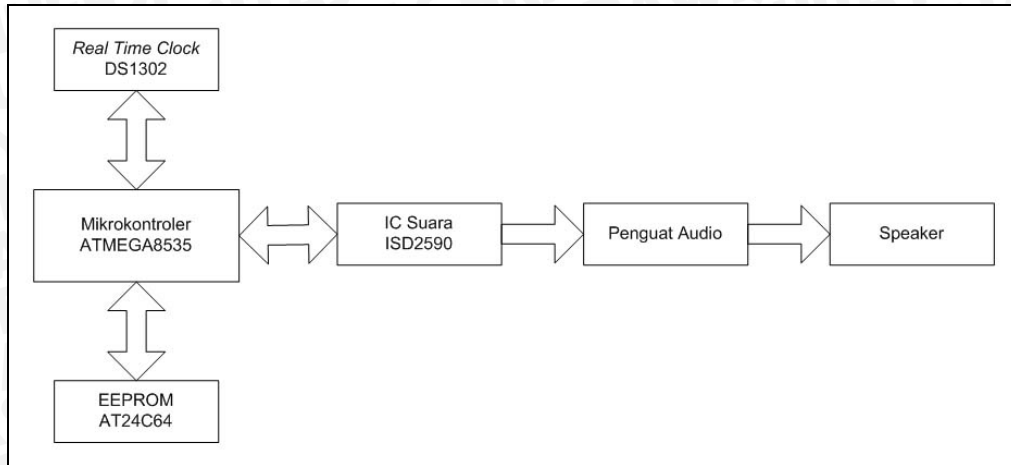
Peralatan yang digunakan untuk pengujian perangkat lunak ini adalah:

1. Minimum sistem mikrokontroler ATMEGA8535
2. Rangkaian RTC DS1302
3. Rangkaian EEPROM AT24C64
4. Rangkaian IC suara ISD2590
5. Rangkaian penguat audio
6. Speaker
7. *Software* CodeVisionAVR dan PonyProg2000

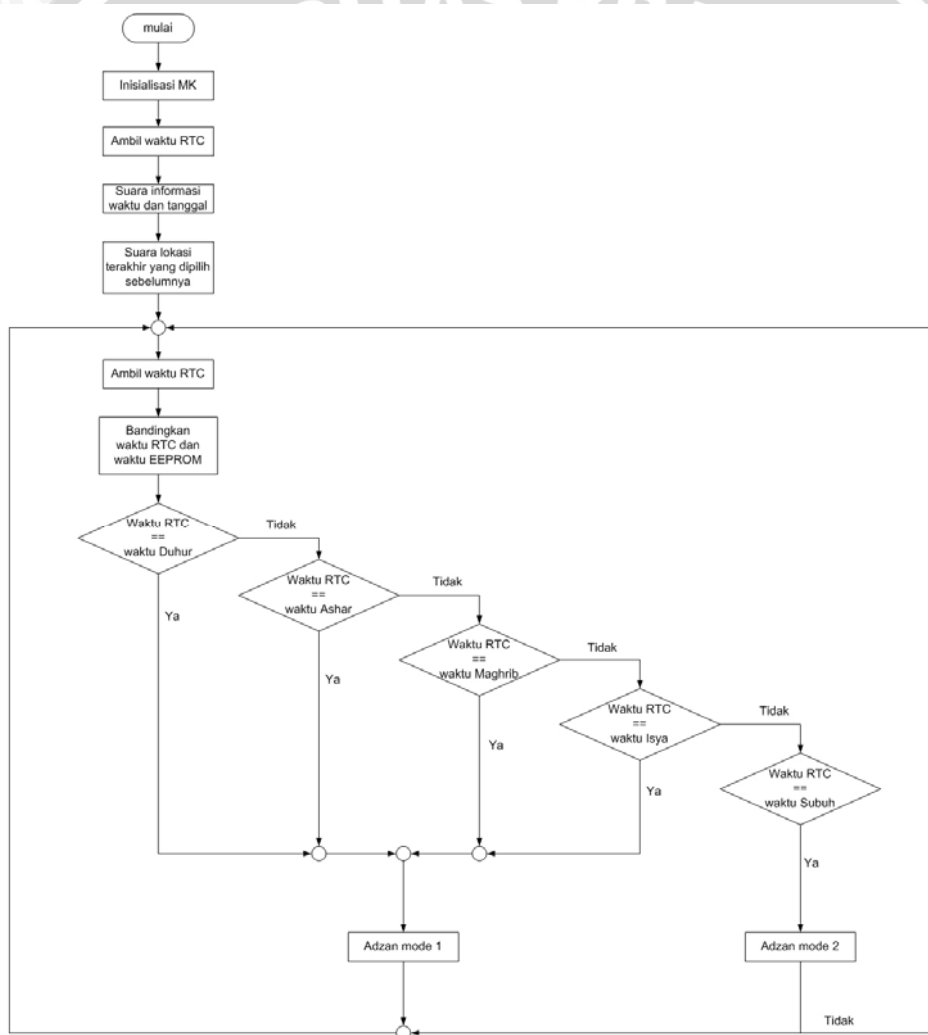
5.6.3 Prosedur Pengujian

Prosedur pengujian yang dilakukan adalah:

1. Menyusun rangkaian seperti dalam Gambar 5.12
2. Memprogram mikrokontroler ATMEGA8535 sesuai dengan program pengujian perangkat lunak untuk mengetahui waktu sholat. Diagram alir program pengujian ini ditunjukkan dalam Gambar 5.13
3. Mengamati waktu sholat yang ditunjukkan dan keluaran suara melalui *speaker*.



Gambar 5.12 Diagram Blok Pengujian Perangkat Lunak Untuk Mengetahui Waktu Sholat



Gambar 5.13 Diagram Alir Pengujian Perangkat Lunak Untuk Mengetahui Waktu Sholat

5.6.4 Hasil

Dari pengujian perangkat lunak untuk mengetahui waktu sholat didapatkan hasil yang ditunjukkan dalam Tabel 5.11.

Tabel 5.11 Data Hasil Pengujian Perangkat Lunak Untuk Mengetahui Waktu Sholat

Tanggal Pengujian	Waktu Pengujian	Waktu Sholat	Suara Keluaran <i>Speaker</i>
5 Maret	11.43	Duhur	Adzan mode 1
7 Maret	14.47	Ashar	Adzan mode 1
9 Maret	17.48	Maghrib	Adzan mode 1
12 Maret	18.56	Isya	Adzan mode 1
15 Maret	04.19	Subuh	Adzan mode 2

5.6.5 Analisis dan Kesimpulan

Data hasil pengujian dalam Tabel menunjukkan bahwa perangkat lunak untuk mengetahui waktu sholat dapat berfungsi dengan baik sesuai dengan yang direncanakan. Suara keluaran speaker ketika waktu menunjukkan waktu sholat Duhur, Ashar, Maghrib dan Isya adalah suara adzan mode1. Sedangkan ketika waktu menunjukkan waktu Subuh, suara keluaran speaker adalah suara adzan mode2.

5.7 Pengujian Perangkat Lunak Untuk Perhitungan Arah Kiblat

5.7.1 Tujuan

Tujuan pengujian ini adalah untuk mengetahui apakah program yang dibuat sudah dapat melakukan perhitungan arah kiblat sesuai dengan yang diharapkan.

5.7.2 Peralatan

Peralatan yang digunakan untuk pengujian perangkat lunak ini adalah:

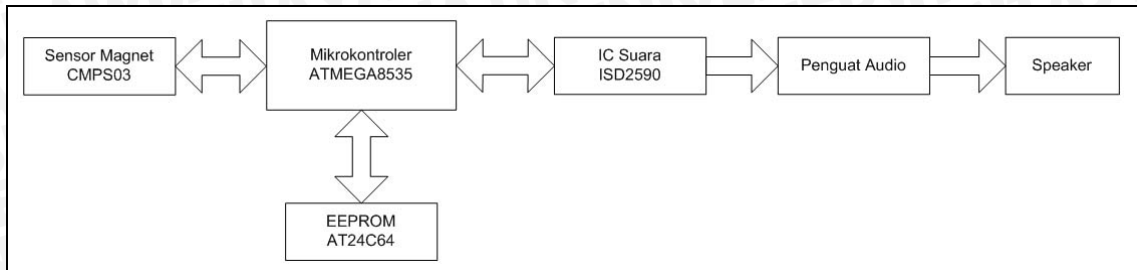
1. Minimum sistem mikrokontroler ATMEGA8535
2. Rangkaian sensor magnet CMPS03
3. Rangkaian EEPROM AT24C64
4. Rangkaian IC suara ISD2590
5. Rangkaian penguat audio
6. Speaker
7. *Software CodeVisionAVR dan PonyProg2000*

5.7.3 Prosedur Pengujian

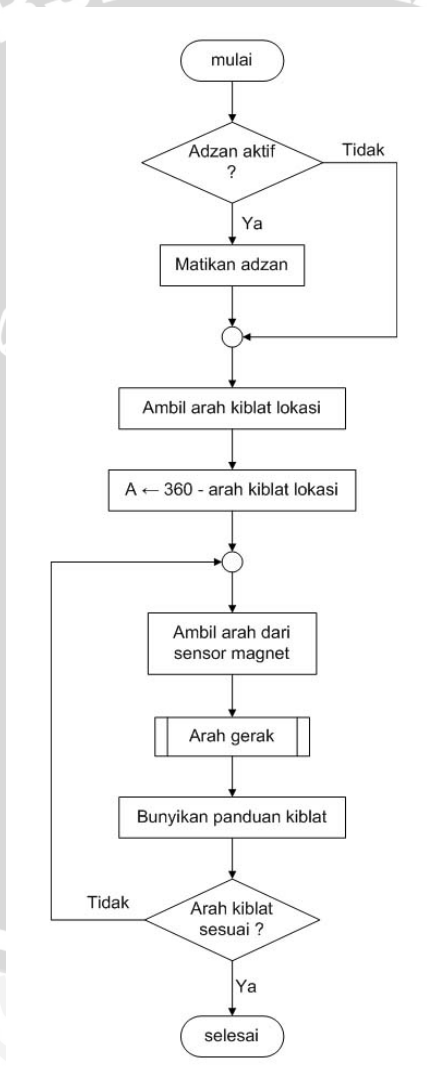
Prosedur pengujian yang dilakukan adalah:

1. Menyusun rangkaian seperti dalam Gambar 5.14.

2. Memprogram mikrokontroler ATMEGA8535 sesuai dengan program pengujian perangkat lunak untuk perhitungan arah kiblat. Diagram alir program pengujian ini dapat dilihat dalam Gambar 5.15.
3. Mengamati keluaran suara melalui *speaker*.



Gambar 5.14 Diagram Blok Pengujian Perangkat Lunak Perhitungan Arah Kiblat



Gambar 5.15 Diagram Alir Pengujian Perangkat Lunak Perhitungan Arah Kiblat

5.7.4 Hasil

Data hasil pengujian perangkat lunak perhitungan arah kiblat ditunjukkan dalam Tabel 5.12.

Tabel 5.12 Data Hasil Pengujian Perangkat Lunak Perhitungan Arah Kiblat

Arah Kiblat (Utara sebagai acuan = 0°)	Pendeteksian Sensor Magnet (Utara sebagai acuan = 0°)	Suara Keluaran <i>Speaker</i>
68°	200°	Se/ratus/tiga/puluh/dua/derajat/ke/kiri
78°	20°	Lima/puluh/delapan/derajat/ke/kanan
86°	130°	Empat/puluh/empat/derajat/ke/kiri
67°	35°	Tiga/puluh/dua/derajat/ke/kanan
68°	180°	Se/ratus/dua/belas/derajat/ke/kiri
69°	60°	Sembilan/derajat/ke/kanan
108°	250°	Se/ratus/empat/puluh/dua/ke/kiri

5.7.5 Analisis dan Kesimpulan

Dari data dalam Tabel menunjukkan bahwa perangkat lunak perhitungan arah kiblat dapat berfungsi dengan baik sesuai dengan yang direncanakan. Suara keluaran speaker sudah dapat menunjukkan dengan baik nilai selisih dari pendeteksian sensor magnet dan arah kiblat serta dapat menunjukkan arah ke kiri ataupun ke kanan.

5.8 Pengujian Sistem Secara Keseluruhan

5.8.1 Tujuan

Tujuan pengujian ini adalah untuk mengetahui kinerja sistem secara keseluruhan apakah sudah sesuai dengan perancangan yang telah dibuat.

5.8.2 Peralatan

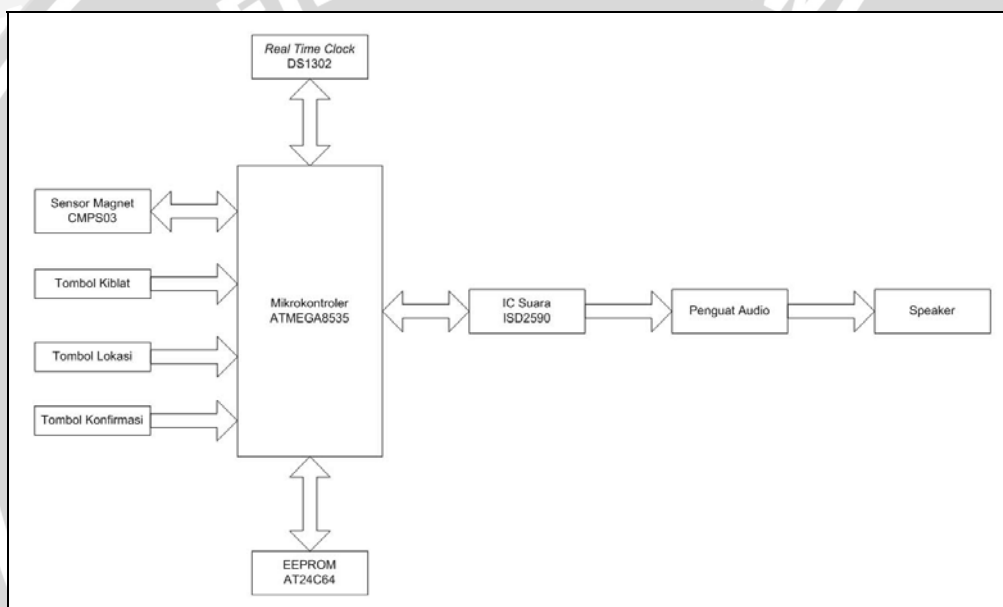
Peralatan yang digunakan untuk pengujian perangkat lunak ini adalah:

1. Minimum sistem mikrokontroler ATMEGA8535
2. Rangkaian RTC DS1302
3. Rangkaian sensor magnet CMPS03
4. Rangkaian EEPROM AT24C64
5. Rangkaian IC suara ISD2590
6. Rangkaian penguat audio
7. Tiga buah tombol
8. Speaker
9. *Software* CodeVisionAVR dan PonyProg2000

5.8.3 Prosedur Pengujian

Prosedur pengujian yang dilakukan adalah:

1. Menyusun rangkaian sesuai dengan diagram blok keseluruhan seperti dalam Gambar 5.16.
2. Memprogram mikrokontroler ATMEGA8535 sesuai dengan program pengujian sistem keseluruhan.
3. Melakukan penekanan tombol lokasi, tombol konfirmasi untuk memilih lokasi.
4. Melakukan penekanan tombol kiblat untuk mengetahui arah kiblat.
5. Mengamati keluaran suara melalui speaker.



Gambar 5.16 Diagram Blok Pengujian Sistem Keseluruhan

5.8.4 Hasil

Dari pengujian sistem keseluruhan didapatkan data sebagai berikut:

1. Jika sistem dinyalakan, program yang dijalankan adalah program utama yaitu program untuk mengetahui waktu sholat. Ketika waktu sholat tiba (selain sholat Subuh), suara adzan mode 1 dikeluarkan melalui *speaker* dan ketika waktu sholat Subuh tiba maka suara adzan mode 2 yang dikeluarkan speaker. Lokasi yang digunakan untuk mengetahui waktu sholat ini adalah Malang.

2. Ketika tombol lokasi ditekan, keluaran speaker adalah suara pilihan lokasi kota.
3. Penekanan tombol konfirmasi setelah suara pilihan lokasi sesuai dengan yang diinginkan menyebabkan penambahan waktu terhadap data waktu sholat.
4. Ketika tombol kiblat ditekan, program mulai melakukan perhitungan arah kiblat dan hasilnya dikeluarkan melalui speaker sebagai panduan arah kiblat.

5.8.5 Analisis dan Kesimpulan

Dari hasil pengujian, dapat disimpulkan bahwa sistem keseluruhan dapat berfungsi untuk memberikan informasi waktu sholat dan sebagai panduan untuk mengetahui arah kiblat sesuai dengan yang direncanakan.



BAB IV PENUTUP

6.1 Kesimpulan

Dari hasil perancangan, pembuatan dan pengujian alat penunjuk arah kiblat dan informasi waktu sholat bagi penderita tuna netra diperoleh kesimpulan sebagai berikut:

1. Alat dapat memberikan informasi waktu sholat dengan baik, yaitu dengan memberikan keluaran suara adzan mode 1 ketika waktu sholat Duhur, Ashar, Maghrib dan Isya, serta suara adzan mode 2 ketika waktu sholat Subuh melalui *speaker*.
2. Arah kiblat dapat ditunjukkan oleh alat dengan memberikan keluaran suara panduan arah kiblat berupa selisih antara nilai arah yang dideteksi sensor magnet dan nilai arah kiblat yang tersimpan dalam EEPROM. Dan juga dengan memberikan keluaran suara 'ke kanan' dan 'ke kiri' untuk memudahkan pengguna.
3. Antarmuka komunikasi antara sensor magnet CMPS03 dan mikrokontroler yaitu I²C yang dapat melakukan komunikasi secara dua arah.
4. Berdasarkan hasil pengujian, IC suara ISD2590 dapat melakukan perekaman suara dan memutar ulang hasil rekaman dengan durasi maksimal 90 detik.
5. Dari hasil pengujian, perangkat lunak untuk mengetahui waktu sholat dan perhitungan arah kiblat yang direncanakan sudah dapat berfungsi dengan baik.

6.2 Saran

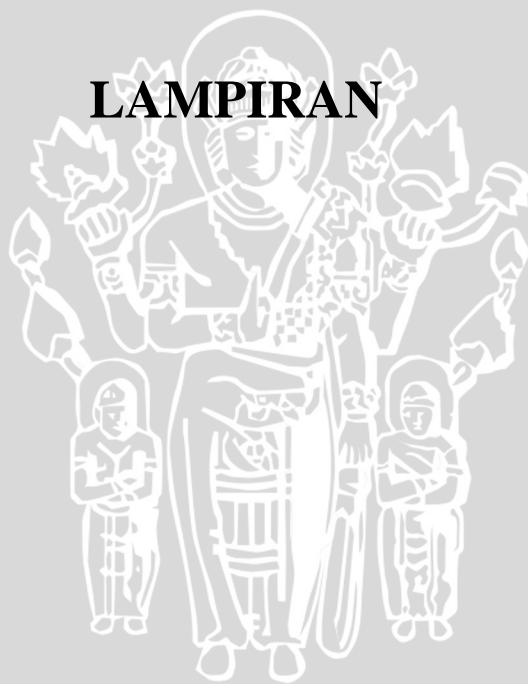
Untuk pengembangan lebih lanjut, ada beberapa hal yang perlu ditambahkan dalam alat ini, diantaranya adalah:

1. Penggunaan GPS (*Global Positioning System*) akan dapat memberikan hasil perhitungan arah kiblat yang lebih akurat.

DAFTAR PUSTAKA

- Atmel. 2001. 2 – *Wire Serial EEPROM*. www.atmel.com/literatur. Diakses tanggal 1 Januari 2007
- Atmel. 2003. *ATMEGA8535/ATMEGA8535L, 8-bit AVR with 8 Kbytes in System Programable Flash*. www.atmel.com/literatur. Diakses tanggal 1 Januari 2007.
- Departemen Agama RI. 1981. *Almanak Hisab dan Rukyat*. Badan Hisab dan Rukyat
- Departemen Agama RI. 2005. *Al Quran dan Terjemahnya*. Bandung : PT. Syamiil Cipta Media
- Electronics Compass Support. 2006. www.superdroidrobots.com. Diakses tanggal 14 Desember 2006
- Hayt, William H. 1981. *Engineering Electromagnetic, 4th editions*. Purdue : McGraw – Hill
- <http://www.robot-electronics.co.uk/htm/cmeps3doc.shtml>. Diakses tanggal 14 Desember 2006
- <http://rukayatulhilar.tripod.com/jadwalshalat>. Diakses tanggal 5 Januari 2007
- Iskander, Magdy F. 1992. *Electromagnetic field and Waves*. New Jersey : Prentice Hall
- Kurniawan, Wahid Arief. 2003. *Perencanaan dan Pembuatan Alat Penunjuk Waktu Sholat dengan Menggunakan Mikrokontroller AT89C51*. Skripsi Tidak Diterbitkan. Malang: Jurusan Teknik Elektro FT Unibraw.
- National Semiconductor, 2000, *LM386 Low Voltage Audio Power Ampilfier*. www.datasheetcatalog.com. Diakses tanggal 17 Januari 2007
- Philips. 2000. *General Magnetoresistive Sensors for Magnetic Field Measurement*. www.semiconductors.philips.com. Diakses tanggal 14 Desember 2006
- Ponyprog. 2005. *Serial device programmer*. www.lancos.com. Diakses tanggal 1 Januari 2007
- Rachim, Abdur. 1983. *Ilmu Falak*. Yogyakarta : Liberty
- STMicroelectronic. 2005. *64Kbit and 32Kbit Serial I²C Bus EEPROM*. www.st.com. Diakses tanggal 17 Januari 2007
- Winbond. 2003. *ISD2560/75/90/120 Single chip, multiple message, voice record/playback device 60-, 75-, 90- and 120- second duration*. www.winbond-usa.com. Diakses tanggal 17 Januari 2007

UNIVERSITAS BRAWIJAYA



LAMPIRAN

LAMPIRAN 1

Tabel Jadwal Waktu Sholat

JANUARI						
Tanggal	Dhuhur	Ashar	Maghrib	Isya	Subuh	
1	2	11.34	15.02	17.52	19.07	03.54
3	5	11.36	15.04	17.53	19.09	03.56
6	8	11.37	15.04	17.54	19.10	03.58
9	11	11.38	15.05	17.55	19.10	03.59
12	14	11.40	15.06	17.56	19.11	04.01
15	17	11.41	15.06	17.57	19.11	04.03
18	20	11.42	15.07	17.58	19.12	04.05
21	23	11.43	15.07	17.58	19.12	04.06
24	26	11.43	15.07	17.58	19.12	04.07
27	29	11.44	15.06	17.59	19.11	04.10
30	31	11.44	15.05	17.59	19.11	04.11

FEBRUARI						
Tanggal	Dhuhur	Ashar	Maghrib	Isya	Subuh	
1	2	11.45	15.05	17.59	19.11	04.11
3	5	11.45	15.04	17.58	19.10	04.13
6	8	11.45	15.02	17.58	19.10	04.14
9	11	11.45	15.02	17.58	19.09	04.15
12	14	11.45	15.00	17.57	19.08	04.16
15	17	11.45	14.58	17.56	19.07	04.17
18	20	11.45	14.57	17.56	19.06	04.17
21	23	11.45	14.54	17.55	19.05	04.18
24	26	11.44	14.52	17.54	19.04	04.19
27	29	11.44	14.49	17.53	19.03	04.19

MARET						
Tanggal	Dhuhur	Ashar	Maghrib	Isya	Subuh	
1	2	11.44	14.48	17.52	19.02	04.19
3	5	11.43	14.46	17.51	19.00	04.19
6	8	11.42	14.47	17.50	18.59	04.19
9	11	11.41	14.49	17.48	18.57	04.19
12	14	11.41	14.49	17.47	18.56	04.19
15	17	11.40	14.50	17.46	18.54	04.19
18	20	11.39	14.51	17.44	18.53	04.19
21	23	11.38	14.51	17.43	18.51	04.19
24	26	11.37	14.52	17.41	18.50	04.19
27	29	11.36	14.52	17.40	18.48	04.18
30	31	11.35	14.52	17.38	18.47	04.18

APRIL						
Tanggal	Dhuhur	Ashar	Maghrib	Isya	Subuh	
1	2	11.35	14.52	17.38	18.46	04.18
3	5	11.35	14.52	17.36	18.45	04.17
6	8	11.33	14.52	17.35	18.44	04.17
9	11	11.32	14.52	17.34	18.43	04.16
12	14	11.32	14.52	17.32	18.41	04.16
15	17	11.32	14.52	17.31	18.40	04.15
18	20	11.30	14.52	17.30	18.39	04.15
21	23	11.30	14.51	17.29	18.38	04.14
24	26	11.29	14.51	17.28	18.38	04.14
27	29	11.29	14.51	17.27	18.37	04.14
30		11.28	14.51	17.26	18.36	04.14

MEI						
Tanggal	Dhuhur	Ashar	Maghrib	Isya	Subuh	
1	2	11.28	14.51	17.26	18.36	04.14
3	5	11.28	14.50	17.25	18.36	04.13
6	8	11.28	14.50	17.24	18.36	04.13
9	11	11.27	14.50	17.24	18.35	04.13
12	14	11.27	14.50	17.23	18.35	04.13
15	17	11.27	14.50	17.23	18.35	04.13
18	20	11.27	14.50	17.23	18.35	04.13
21	23	11.28	14.50	17.22	18.35	04.13
24	26	11.28	14.51	17.22	18.35	04.14
27	29	11.28	14.51	17.22	18.36	04.14
30	31	11.30	14.51	17.23	18.36	04.14

JUNI						
Tanggal	Dhuhur	Ashar	Maghrib	Isya	Subuh	
1	2	11.29	14.51	17.23	18.36	04.14
3	5	11.29	14.52	17.23	18.37	04.15
6	8	11.30	14.52	17.23	18.37	04.15
9	11	11.30	14.53	17.24	18.38	04.16
12	14	11.31	14.53	17.24	18.38	04.17
15	17	11.31	14.54	17.25	18.39	04.17
18	20	11.32	14.54	17.25	18.40	04.18
21	23	11.33	14.55	17.26	18.40	04.18
24	26	11.33	14.56	17.27	18.41	04.19
27	29	11.34	14.57	17.27	18.42	04.20
30		11.34	14.57	17.27	18.42	04.20

JULI						
Tanggal	Dhuhur	Ashar	Maghrib	Isya	Subuh	
1	2	11.34	14.57	17.27	18.42	04.20
3	5	11.35	14.57	17.28	18.42	04.20
6	8	11.35	14.58	17.29	18.43	04.21
9	11	11.36	14.59	17.30	18.44	04.22
12	14	11.37	14.59	17.31	18.44	04.22
15	17	11.37	15.00	17.31	18.44	04.23
18	20	11.37	15.00	17.32	18.45	04.23
21	23	11.37	15.00	17.32	18.45	04.23
24	26	11.37	15.00	17.33	18.45	04.23
27	29	11.37	15.00	17.33	18.45	04.23
30	31	11.37	15.00	17.33	18.45	04.23

AGUSTUS						
Tanggal	Dhuhur	Ashar	Maghrib	Isya	Subuh	
1	2	11.37	15.00	17.33	18.45	04.23
3	5	11.37	15.00	17.34	18.45	04.23
6	8	11.37	15.00	17.34	18.45	04.22
9	11	11.36	14.59	17.34	18.45	04.22
12	14	11.36	14.58	17.34	18.44	04.21
15	17	11.35	14.58	17.34	18.44	04.21
18	20	11.35	14.57	17.33	18.43	04.20
21	23	11.34	14.56	17.33	18.43	04.19
24	26	11.33	14.55	17.33	18.42	04.18
27	29	11.32	14.53	17.33	18.42	04.17
30	31	11.31	14.52	17.32	18.41	04.16

SEPTEMBER						
Tanggal	Dhuhur	Ashar	Maghrib	Isya	Subuh	
1	2	11.31	14.51	17.32	18.41	04.15
3	5	11.30	14.50	17.32	18.41	04.14
6	8	11.29	14.48	17.31	18.40	04.13
9	11	11.28	14.46	17.31	18.39	04.11
12	14	11.28	14.44	17.30	18.39	04.10
15	17	11.26	14.42	17.30	18.38	04.08
18	20	11.25	14.40	17.29	18.38	04.06
21	23	11.24	14.37	17.29	18.37	04.05
24	26	11.23	14.35	17.28	18.37	04.03
27	29	11.22	14.32	17.28	18.36	04.01
30		11.22	14.30	17.27	18.36	04.00

OKTOBER						
Tanggal	Dhuhur	Ashar	Maghrib	Isya	Subuh	
1	2	11.22	14.30	17.27	18.36	04.00
3	5	11.20	14.27	17.27	18.36	03.58
6	8	11.19	14.24	17.27	18.36	03.58
9	11	11.19	14.22	17.26	18.36	03.55
12	14	11.17	14.21	17.26	18.36	03.53
15	17	11.17	14.23	17.26	18.36	03.52
18	20	11.16	14.25	17.26	18.36	03.50
21	23	11.16	14.26	17.26	18.36	03.49
24	26	11.15	14.28	17.26	18.37	03.47
27	29	11.15	14.29	17.27	18.37	03.46
30	31	11.15	14.30	17.27	18.38	03.45

NOVEMBER						
Tanggal	Dhuhur	Ashar	Maghrib	Isya	Subuh	
1	2	11.15	14.31	17.27	18.38	03.45
3	5	11.15	14.33	17.28	18.39	03.44
6	8	11.15	14.34	17.28	18.40	03.43
9	11	11.15	14.35	17.29	18.41	03.42
12	14	11.15	14.37	17.30	18.42	03.42
15	17	11.16	14.39	17.31	18.44	03.41
18	20	11.16	14.40	17.32	18.45	03.41
21	23	11.17	14.42	17.33	18.47	03.41
24	26	11.19	14.43	17.34	18.48	03.41
27	29	11.20	14.45	17.35	18.50	03.41
30		11.20	14.46	17.37	18.51	03.41

DESEMBER						
Tanggal	Dhuhur	Ashar	Maghrib	Isya	Subuh	
1	2	11.20	14.46	17.37	18.51	03.41
3	5	11.21	14.48	17.38	18.53	03.42
6	8	11.23	14.50	17.39	18.55	03.43
9	11	11.24	14.51	17.41	18.57	03.44
12	14	11.25	14.53	17.43	18.58	03.45
15	17	11.26	14.55	17.44	19.00	03.46
18	20	11.29	14.56	17.46	19.02	03.47
21	23	11.30	14.58	17.47	19.03	03.49
24	26	11.31	14.59	17.49	19.04	03.50
27	29	11.32	15.01	17.50	19.06	03.52
30	31	11.35	15.02	17.51	19.27	03.53

Sumber: Syamil Cipta Media (yang bersumber pada Departemen Agama RI)



Tabel Perhitungan Arah Kiblat

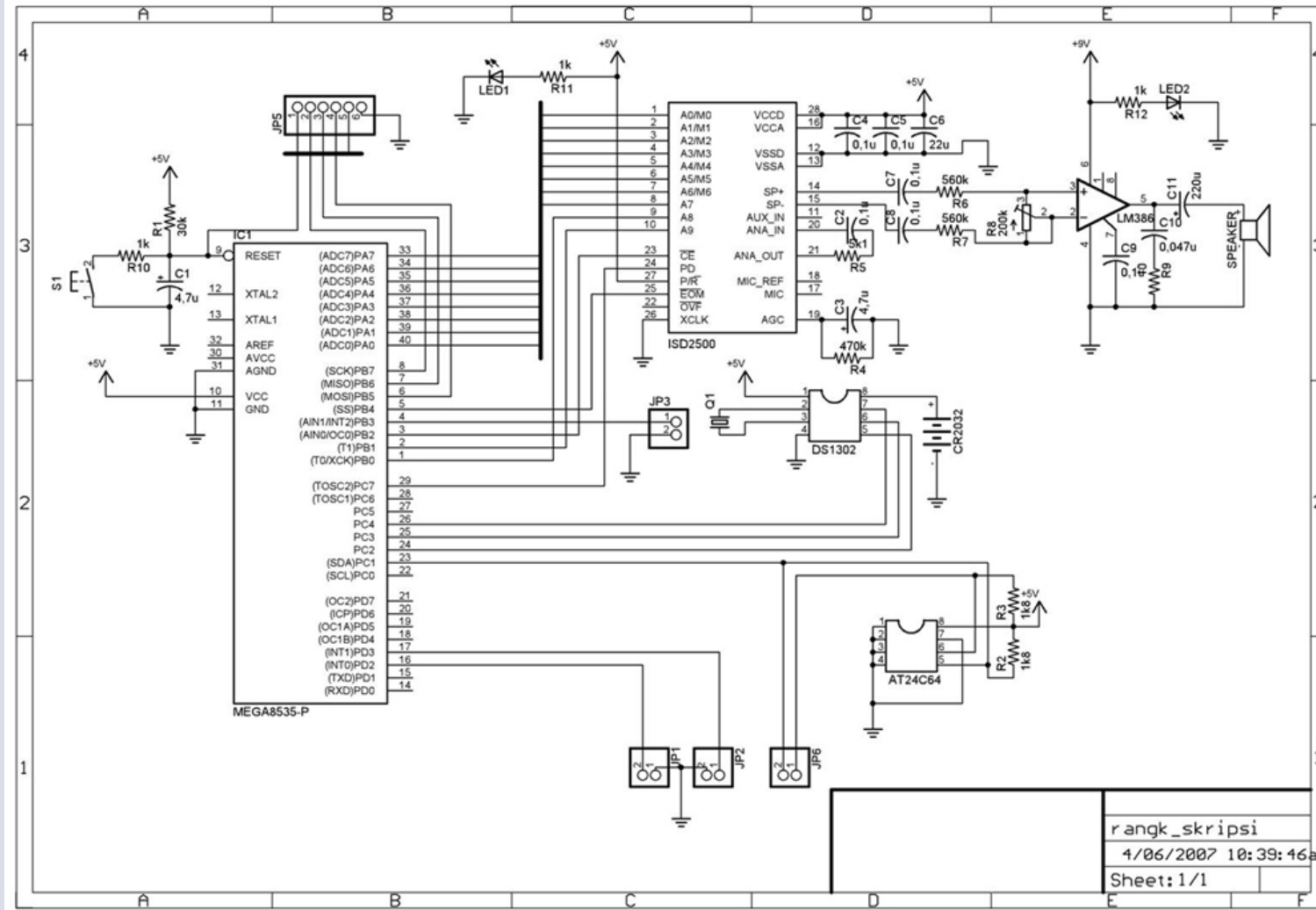
No	Kota	Lintang		Bujur		cot A	tg A	A	Pembulatan A
1	Banda Aceh	5°32'	U	95°19'	T	0.407500746	2.453983238	67.82906771	67.8
2	Medan	3°35'	U	98°40'	T	0.419695316	2.382680868	67.23243532	67.2
3	Pekanbaru	0°33'	U	101°26'	T	0.44068468	2.269196197	66.21764745	66.2
4	Padang	0°57'	S	100°21'	T	0.459894345	2.174412473	65.30256633	65.3
5	Bengkulu	3°47'	S	102°14'	T	0.476128173	2.100274793	64.53956488	64.5
6	Jambi	1°35'	S	103°36'	T	0.450718569	2.218679391	65.73802608	65.7
7	Palembang	2°58'	S	104°45'	T	0.456718592	2.189532061	65.45293861	65.5
8	Tanjungkarang	5°26'	S	105°15'	T	0.47270819	2.115470011	64.69951504	64.7
9	Serang	6°07'	S	106°08'	T	0.472692898	2.115538448	64.70023118	64.7
10	Jakarta	6°10'	S	106°49'	T	0.46932567	2.130716609	64.85812941	64.9
11	Bandung	6°54'	S	107°36'	T	0.46977533	2.128677127	64.83701996	64.8
12	Cirebon	6°42'	S	108°33'	T	0.463514978	2.157427586	65.13157073	65.1
13	Yogyakarta	7°47'	S	110°21'	T	0.460135257	2.173274021	65.29117388	65.3
14	Semarang	6°58'	S	110°25'	T	0.455567077	2.195066436	65.5075523	65.5
15	Surakarta	7°34'	S	110°49'	T	0.456644743	2.189886154	65.45643968	65.5
16	Malang	7°58'	S	112°37'	T	0.449614792	2.224126114	65.79061142	65.8
17	Surabaya	7°14'	S	112°44'	T	0.445841848	2.242947818	65.97068704	66.0
18	Pontianak	0°01'	S	109°20'	T	0.418858567	2.387440723	67.27320965	67.3
19	Palangkaraya	2°12'	S	113°55'	T	0.418514841	2.389401527	67.28996629	67.3
20	Balikpapan	1°15'	S	116°49'	T	0.407522909	2.453849779	67.82797871	67.8
21	Samarinda	0°29'	S	117°08'	T	0.403954847	2.475524201	68.00351588	68.0
22	Denpasar	8°39'	S	115°13'	T	0.439961998	2.272923581	66.25232972	66.3
23	Lombok	8°30'	S	116°39'	T	0.433046352	2.309221623	66.58515199	66.6
24	Palu	0°53'	S	119°51'	T	0.400928091	2.494212859	68.15276407	68.1
25	Manado	1°29'	U	124°50'	T	0.391332891	2.555369158	68.6279595	68.6

26	Makassar	5°09'	S	119°24'	T	0.413744036	2.416953268	67.522966	67.5
27	Gorontalo	0°33'	U	123°03'	T	0.393836329	2.539125841	68.50367774	68.5
28	Kupang	10°10'	S	123°34'	T	0.407776663	2.45232278	67.81551146	67.8
29	Kendari	3°57'	S	122°35'	T	0.403216656	2.48005628	68.03988694	68.0
30	Ambon	3°42'	S	128°09'	T	0.393479231	2.541430198	68.52139272	68.5
31	Jayapura	2°32'	S	140°42'	T	0.390517643	2.560703768	68.66847768	68.7



LAMPIRAN 2

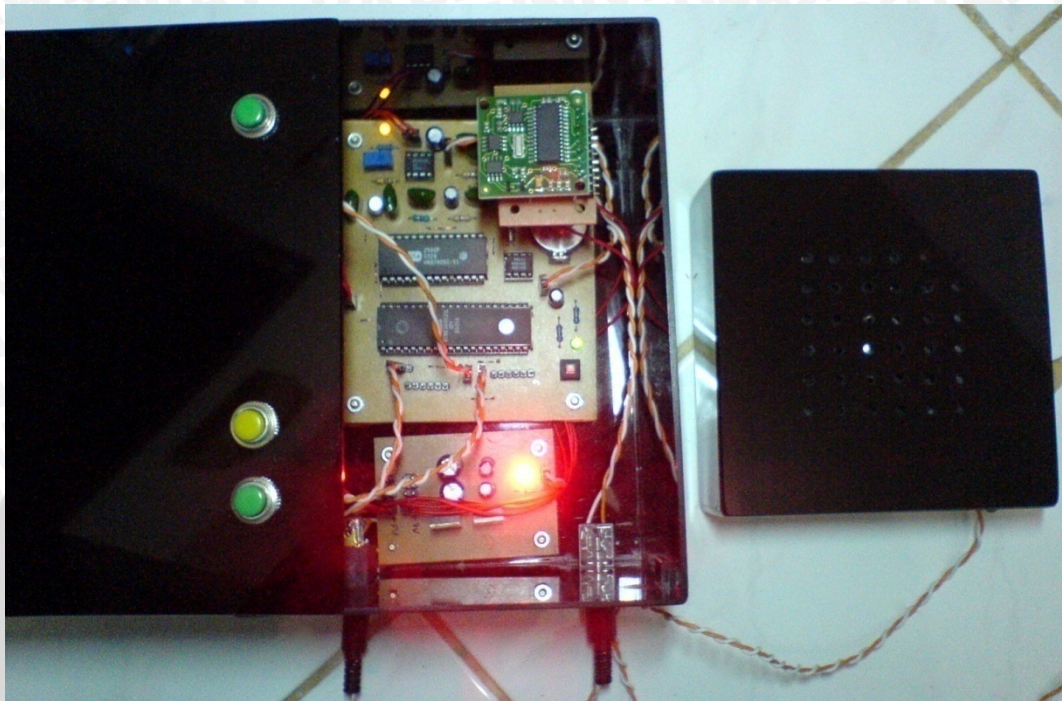
Gambar Rangkaian Lengkap



rangk_skripsi
4/06/2007 10:39:46a
Sheet: 1/1

LAMPIRAN 3

Foto Alat



LAMPIRAN 4

Spesifikasi Alat

Spesifikasi alat penunjuk arah kiblat dan informasi waktu sholat bagi penderita tuna netra adalah sebagai berikut:

1. Alat ini menggunakan sensor magnet CMPS03.
2. Untuk informasi waktu diperoleh dari *Real Time Clock* DS1302 yang menggunakan baterai tipe CR2032 sebagai catu cadangan dengan tegangan sebesar 3 volt.
3. Untuk catu daya utama digunakan sumber tegangan DC 12 volt.

Petunjuk Pengoperasian Alat

Alat ini memiliki bentuk yang sederhana dan tidak diperlukan keahlian khusus dalam mengoperasikannya. Adapun cara mengoperasikan alat ini adalah sebagai berikut:

1. Alat dapat disimpan ataupun dipegang dengan posisi tombol lebih dekat dengan tubuh pengguna, bisa dilakukan dengan cara diraba.
2. Tekan tombol sebelah kiri pertama untuk pemilihan lokasi.
3. Tekan tombol sebelah kiri kedua untuk konfirmasi lokasi.
4. Tekan tombol sebelah kanan untuk mengetahui informasi arah kiblat.

LAMPIRAN 5

Listing Program

```

#include <mega8535.h>
#include <delay.h>
#include <math.h>

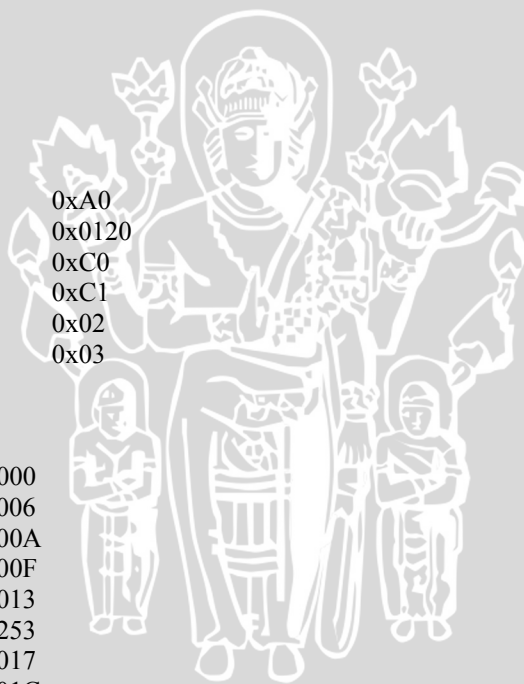
// I2C Bus functions
#asm
.equ __i2c_port=0x15 ;PORTC
.equ __sda_bit=1
.equ __scl_bit=0
#endasm
#include <i2c.h>

// DS1302 Real Time Clock functions
#asm
.equ __ds1302_port=0x15 ;PORTC
.equ __ds1302_io=3
.equ __ds1302_sclk=4
.equ __ds1302_rst=2
#endasm
#include <ds1302.h>

#define alamat_eeprom      0xA0
#define alamat_tambahan    0x0120
#define alamat1_kompas     0xC0
#define alamat2_kompas     0xC1
#define register1_kompas   0x02
#define register2_kompas   0x03

// Alamat Suara Pada ISD
#define Banda_Aceh         0x0000
#define Medan              0x0006
#define Pekanbaru         0x000A
#define Padang             0x000F
#define Bengkulu          0x0013
#define Jambi              0x0253
#define Palembang         0x0017
#define Tanjungkarang     0x001C
#define Serang            0x0022
#define Jakarta           0x0026
#define Bandung           0x002B
#define Cirebon           0x002F
#define Yogyakarta        0x0033
#define Semarang          0x0039
#define Surakarta         0x003D
#define Malang            0x0043
#define Surabaya          0x0048
#define Pontianak         0x0050
#define Palangkaraya      0x0057
#define Balikpapan       0x005F
#define Samarinda         0x0066
#define Denpasar         0x006D
#define Lombok           0x0074
#define Palu              0x0079

```



```
#define Manado      0x007E
#define Makassar    0x0084
#define Gorontalo   0x0089
#define Kupang      0x0091
#define Kendari     0x0096
#define Ambon       0x009C
#define Jayapura    0x00A1
```

```
#define Nol         0x00A8
#define Satu        0x00AD
#define Dua         0x00B2
#define Tiga       0x00B7
#define Empat      0x00BC
#define Lima       0x00C1
#define Enam       0x00C6
#define Tujuh      0x00CA
#define Delapan    0x00CF
#define Sembilan   0x00D4
#define Se         0x00DB
#define Puluh      0x00DF
#define Belas      0x00E4
#define Ratus      0x00E8
```

```
#define Pilih      0x00EF
#define Lokasi     0x00F3
#define Dipilih    0x00F9
```

```
#define Arah       0x0100
#define Kiblat     0x0105
#define Sudah      0x010A
#define Tepat      0x010F
```

```
#define Ke         0x0114
#define Kanan      0x0119
#define Kiri       0x011E
#define Derajat    0x0123
```

```
#define Pukul     0x0128
#define Menit      0x012E
#define Tanggal    0x0133
#define Bulan      0x0138
```

```
#define adzan     0x013D
#define mode1      0x01FD
#define mode2      0x022B
```

```
//Variabel Global
```

```
//Variabel RTC
unsigned char tanggal,bulan,tahun;
unsigned char jam,menit,detik;
```

```
//Variabel EEPROM
unsigned char jam_subuh,menit_subuh,jam_duhur,menit_duhur,jam_ashar,menit_ashar,
jam_maghrib,menit_maghrib,jam_isya,menit_isya;
unsigned int alamat_data_eeprom = 0;
unsigned char eeprom_1,eeprom_h,data_eeprom;
unsigned int arah_kiblat;
unsigned char eeprom_data_1,eeprom_data_h;
```



```

//Variabel ISD
unsigned char isdh,isdl;
unsigned int selisih,selisih_temp,bunyi_angka,temp_int,lokasi;
bit tanda_subuh,tanda_duhur,tanda_ashar,tanda_maghrib,tanda_isya;
unsigned char ratusan,puluhan,satuan,temp,a,b,c;
unsigned char angka,angka_temp;

//Variabel CMPS03
unsigned char data_h,data_l,hasil_konversi,data_baca_kompas;
unsigned int data_kompas,kompas_h,cermin;
unsigned char kompas_l;

void ambil_data_kompas(void);
void cek_bulan(void);
void cek_tanggal(void);
void cek_waktu(void);
void cek_selisih(void);
void decoder_suara(void);
void cek_tombol_konfirmasi(void);
void konversi_waktu_tambah(void);
void konversi_waktu_kurang(void);
void konversi_waktu_kurang_khusus(void);
void suara_angka(void);
void suara_adzan(void);
void isd_set(void);
void inisialisasi(void);
void baca_waktu(void);
void isd_play(unsigned int alamat_isd);
unsigned char baca_eeprom(unsigned int address);
unsigned char konversi_data(unsigned char data_konversi);
void tulis_eeprom(unsigned int address, unsigned int data_eeprom_tulis);
unsigned char baca_kompas(unsigned char register_kompas);

void main(void)
{
    inisialisasi();
    while (1)
    {
        cek_bulan();
    }
}

void baca_waktu(void)
{
    jam_duhur = konversi_data(baca_eeprom(alamat_data_eeprom));
    menit_duhur = konversi_data(baca_eeprom(alamat_data_eeprom+1));
    jam_ashar = konversi_data(baca_eeprom(alamat_data_eeprom+alamat_tambahan));
    menit_ashar = konversi_data(baca_eeprom(alamat_data_eeprom+alamat_tambahan+1));
    jam_maghrib = konversi_data(baca_eeprom(alamat_data_eeprom+(2*alamat_tambahan)));
    menit_maghrib = konversi_data(baca_eeprom(alamat_data_eeprom+(2*alamat_tambahan+1)));
    jam_isya = konversi_data(baca_eeprom(alamat_data_eeprom+(3*alamat_tambahan)));
    menit_isya = konversi_data(baca_eeprom(alamat_data_eeprom+(3*alamat_tambahan+1)));
    jam_subuh = konversi_data(baca_eeprom(alamat_data_eeprom+(4*alamat_tambahan)));
    menit_subuh = konversi_data(baca_eeprom(alamat_data_eeprom+(4*alamat_tambahan+1)));
}

unsigned char baca_eeprom(unsigned int address)
{
    eeprom_l = (unsigned char) address;
    address = address >> 8;
}

```



```

eeprom_h = (unsigned char) address;
i2c_start();
i2c_write(alamat_eeprom);
i2c_write(eeprom_h);
i2c_write(eeprom_l);
i2c_start();
i2c_write(alamat_eeprom | 1);
data_eeprom = i2c_read(0);
i2c_stop();
return data_eeprom;
}

unsigned char konversi_data(unsigned char data_konversi)
{
    data_l = data_konversi & 0x0f;
    data_h = 10 * ((data_konversi >> 4) & 0x0f);
    hasil_konversi = data_h + data_l;
    return hasil_konversi;
}

void tulis_eeprom(unsigned int address, unsigned int data_eeprom_tulis)
{
    eeprom_l = (unsigned char) address;
    address = address >> 8;
    eeprom_h = (unsigned char) address;
    i2c_start();
    i2c_write(alamat_eeprom);
    i2c_write(eeprom_h);
    i2c_write(eeprom_l);
    eeprom_data_l = (unsigned char) data_eeprom_tulis;
    data_eeprom_tulis = data_eeprom_tulis >> 8;
    eeprom_data_h = (unsigned char) data_eeprom_tulis;
    i2c_write(eeprom_data_h);
    i2c_stop();
    delay_ms(10);

    i2c_start();
    i2c_write(alamat_eeprom);
    i2c_write(eeprom_h);
    i2c_write(eeprom_l+1);
    i2c_write(eeprom_data_l);
    i2c_stop();
    delay_ms(10);
}

void inisialisasi(void)
{
    PORTA=0xff;
    DDRA=0xff;
    PORTB=0x1f;
    DDRB=0x07;
    PORTC=0x80;
    DDRC=0x80;
    PORTD=0x0C;
    DDRD=0x00;

    // External Interrupt(s) initialization
    // INT0: On
    // INT0 Mode: Low level
    // INT1: On

```

```

// INT1 Mode: Low level
// INT2: Off
GICR|=0xC0;
MCUCR=0x00;
MCUCSR=0x00;
GIFR=0xC0;

// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;

// DS1302 Real Time Clock initialization
// Trickle charger: Off
rtc_init(0,0,0);

rtc_get_time(&jam,&menit,&detik);
rtc_get_date(&tanggal,&bulan,&tahun);

isd_play(Pukul);
angka = jam;
suara_angka();
delay_ms(500);
angka = menit;
suara_angka();
delay_ms(500);
isd_play(Menit);

isd_play(Tanggal);
angka = tanggal;
suara_angka();
delay_ms(500);
isd_play(Bulan);
angka = bulan;
suara_angka();
delay_ms(500);

//Baca Arah Kiblat dan Lokasi Terakhir Yang Dipilih
lokasi = baca_eeprom(0x05E0);
lokasi <=<= 8;
lokasi = lokasi | baca_eeprom(0x05E1);
isd_play(lokasi);

arah_kiblat = baca_eeprom(0x05E2);
arah_kiblat <=<= 8;
arah_kiblat = arah_kiblat | baca_eeprom(0x05E3);

// Global Interrupt Enable
#asm("sei")
}

// External Interrupt 0 service routine
interrupt [EXT_INT0] void ext_int0_isr(void)
{
isd_play(Pilih);isd_play(Lokasi);
delay_ms(1000);

```

```
lokasi = 0x00FF;
```

```
isd_play(Banda_Aceh);
temp_int = Banda_Aceh;
cek_tombol_konfirmasi();
if (lokasi == Banda_Aceh)
{
    baca_waktu();
    jam_duhur+=1;jam_ashar+=1;jam_maghrib+=1;jam_isya+=1;jam_subuh+=1;
    menit_duhur+=9;menit_ashar+=9;menit_maghrib+=9;menit_isya+=9;menit_subuh+=9;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05C0);
    goto finish;
}
```

```
isd_play(Medan);
temp_int = Medan;
cek_tombol_konfirmasi();
if (lokasi == Medan)
{
    baca_waktu();
    menit_duhur+=55;menit_ashar+=55;menit_maghrib+=55;menit_isya+=55;menit_subuh+=55;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05C1);
    goto finish;
}
```

```
isd_play(Pekanbaru);
temp_int = Pekanbaru;
cek_tombol_konfirmasi();
if (lokasi == Pekanbaru)
{
    baca_waktu();
    menit_duhur+=49;menit_ashar+=49;menit_maghrib+=49;menit_isya+=49;menit_subuh+=49;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05C2);
    goto finish;
}
```

```
isd_play(Padang);
temp_int = Padang;
cek_tombol_konfirmasi();
if (lokasi == Padang)
{
    baca_waktu();
    menit_duhur+=49;menit_ashar+=49;menit_maghrib+=49;menit_isya+=49;menit_subuh+=49;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05C3);
    goto finish;
}
```

```
isd_play(Bengkulu);
temp_int = Bengkulu;
cek_tombol_konfirmasi();
if (lokasi == Bengkulu)
{
    baca_waktu();
    menit_duhur+=41;menit_ashar+=41;menit_maghrib+=41;menit_isya+=41;menit_subuh+=41;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05C4);
}
```



```

    goto finish;
}

isd_play(Jambi);
temp_int = Jambi;
cek_tombol_konfirmasi();
if (lokasi == Jambi)
{
    baca_waktu();
    menit_duhur+=36;menit_ashar+=36;menit_maghrib+=36;menit_isya+=36;menit_subuh+=36;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05C5);
    goto finish;
}

isd_play(Palembang);
temp_int = Palembang;
cek_tombol_konfirmasi();
if (lokasi == Palembang)
{
    baca_waktu();
    menit_duhur+=31;menit_ashar+=31;menit_maghrib+=31;menit_isya+=31;menit_subuh+=31;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05C6);
    goto finish;
}

isd_play(Tanjungkarang);
temp_int = Tanjungkarang;
cek_tombol_konfirmasi();
if (lokasi == Tanjungkarang)
{
    baca_waktu();
    menit_duhur+=29;menit_ashar+=29;menit_maghrib+=29;menit_isya+=29;menit_subuh+=29;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05C7);
    goto finish;
}

isd_play(Serang);
temp_int = Serang;
cek_tombol_konfirmasi();
if (lokasi == Serang)
{
    baca_waktu();
    menit_duhur+=26;menit_ashar+=26;menit_maghrib+=26;menit_isya+=26;menit_subuh+=26;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05C8);
    goto finish;
}

isd_play(Jakarta);
temp_int = Jakarta;
cek_tombol_konfirmasi();
if (lokasi == Jakarta)
{
    baca_waktu();
    menit_duhur+=23;menit_ashar+=23;menit_maghrib+=23;menit_isya+=23;menit_subuh+=23;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05C9);
}

```

```

    goto finish;
}

isd_play(Bandung);
temp_int = Bandung;
cek_tombol_konfirmasi();
if (lokasi == Bandung)
{
    baca_waktu();
    menit_duhur+=20;menit_ashar+=20;menit_maghrib+=20;menit_isya+=20;menit_subuh+=20;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05CA);
    goto finish;
}

isd_play(Cirebon);
temp_int = Cirebon;
cek_tombol_konfirmasi();
if (lokasi == Cirebon)
{
    baca_waktu();
    menit_duhur+=16;menit_ashar+=16;menit_maghrib+=16;menit_isya+=16;menit_subuh+=16;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05CB);
    goto finish;
}

isd_play(Yogyakarta);
temp_int = Yogyakarta;
cek_tombol_konfirmasi();
if (lokasi == Yogyakarta)
{
    baca_waktu();
    menit_duhur+=9;menit_ashar+=9;menit_maghrib+=9;menit_isya+=9;menit_subuh+=9;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05CC);
    goto finish;
}

isd_play(Semarang);
temp_int = Semarang;
cek_tombol_konfirmasi();
if (lokasi == Semarang)
{
    baca_waktu();
    menit_duhur+=8;menit_ashar+=8;menit_maghrib+=8;menit_isya+=8;menit_subuh+=8;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05CD);
    goto finish;
}

isd_play(Surakarta);
temp_int = Surakarta;
cek_tombol_konfirmasi();
if (lokasi == Surakarta)
{
    baca_waktu();
    menit_duhur+=7;menit_ashar+=7;menit_maghrib+=7;menit_isya+=7;menit_subuh+=7;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05CE);
}

```

```

    goto finish;
}

isd_play(Malang);
temp_int = Malang;
cek_tombol_konfirmasi();
if (lokasi == Malang)
{
    baca_waktu();
    arah_kiblat = 360 - baca_eeprom(0x05CF);
    goto finish;
}

isd_play(Surabaya);
temp_int = Surabaya;
cek_tombol_konfirmasi();
if (lokasi == Surabaya)
{
    baca_waktu();
    konversi_waktu_kurang();
    menit_duhur-=1;menit_ashar-=1;menit_maghrib-=1;menit_isya-=1;menit_subuh-=1;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05D0);
    goto finish;
}

isd_play(Pontianak);
temp_int = Pontianak;
cek_tombol_konfirmasi();
if (lokasi == Pontianak)
{
    baca_waktu();
    menit_duhur+=29;menit_ashar+=29;menit_maghrib+=29;menit_isya+=29;menit_subuh+=29;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05D1);
    goto finish;
}

isd_play(Palangkaraya);
temp_int = Palangkaraya;
cek_tombol_konfirmasi();
if (lokasi == Palangkaraya)
{
    baca_waktu();
    menit_duhur+=6;menit_ashar+=6;menit_maghrib+=6;menit_isya+=6;menit_subuh+=6;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05D2);
    goto finish;
}

isd_play(Balikpapan);
temp_int = Balikpapan;
cek_tombol_konfirmasi();
if (lokasi == Balikpapan)
{
    baca_waktu();
    konversi_waktu_kurang();
    menit_duhur-=17;menit_ashar-=17;menit_maghrib-=17;menit_isya-=17;menit_subuh-=17;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05D3);
}

```



```
    goto finish;
}

isd_play(Samarinda);
temp_int = Samarinda;
cek_tombol_konfirmasi();
if (lokasi == Samarinda)
{
    baca_waktu();
    konversi_waktu_kurang();
    menit_duhur==19;menit_ashar==19;menit_maghrib==19;menit_isya==19;menit_subuh==19;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05D4);
    goto finish;
}

isd_play(Denpasar);
temp_int = Denpasar;
cek_tombol_konfirmasi();
if (lokasi == Denpasar)
{
    baca_waktu();
    konversi_waktu_kurang();
    menit_duhur==11;menit_ashar==11;menit_maghrib==11;menit_isya==11;menit_subuh==11;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05D5);
    goto finish;
}

isd_play(Lombok);
temp_int = Lombok;
cek_tombol_konfirmasi();
if (lokasi == Lombok)
{
    baca_waktu();
    konversi_waktu_kurang();
    menit_duhur==16;menit_ashar==16;menit_maghrib==16;menit_isya==16;menit_subuh==16;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05D6);
    goto finish;
}

isd_play(Palu);
temp_int = Palu;
cek_tombol_konfirmasi();
if (lokasi == Palu)
{
    baca_waktu();
    konversi_waktu_kurang();
    menit_duhur==30;menit_ashar==30;menit_maghrib==30;menit_isya==30;menit_subuh==30;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05D7);
    goto finish;
}

isd_play(Manado);
temp_int = Manado;
cek_tombol_konfirmasi();
if (lokasi == Manado)
```

```

{
    baca_waktu();
    konversi_waktu_kurang();
    menit_duhur==31;menit_ashar==31;menit_maghrib==31;menit_isya==31;menit_subuh==31;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05D8);
    goto finish;
}

isd_play(Makassar);
temp_int = Makassar;
cek_tombol_konfirmasi();
if (lokasi == Makassar)
{
    baca_waktu();
    konversi_waktu_kurang();
    menit_duhur==38;menit_ashar==38;menit_maghrib==38;menit_isya==38;menit_subuh==38;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05D9);
    goto finish;
}

isd_play(Gorontalo);
temp_int = Gorontalo;
cek_tombol_konfirmasi();
if (lokasi == Gorontalo)
{
    baca_waktu();
    konversi_waktu_kurang();
    menit_duhur==42;menit_ashar==42;menit_maghrib==42;menit_isya==42;menit_subuh==42;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05DA);
    goto finish;
}

isd_play(Kupang);
temp_int = Kupang;
cek_tombol_konfirmasi();
if (lokasi == Kupang)
{
    baca_waktu();
    konversi_waktu_kurang();
    menit_duhur==44;menit_ashar==44;menit_maghrib==44;menit_isya==44;menit_subuh==44;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05DB);
    goto finish;
}

isd_play(Kendari);
temp_int = Kendari;
cek_tombol_konfirmasi();
if (lokasi == Kendari)
{
    baca_waktu();
    konversi_waktu_kurang();
    menit_duhur==48;menit_ashar==48;menit_maghrib==48;menit_isya==48;menit_subuh==48;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05DC);
    goto finish;
}

```

```

isd_play(Ambon);
temp_int = Ambon;
cek_tombol_konfirmasi();
if (lokasi == Ambon)
{
    baca_waktu();
    konversi_waktu_kurang_khusus();
    menit_duhur==63;menit_ashar==63;menit_maghrib==63;menit_isya==63;menit_subuh==63;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05DD);
    goto finish;
}

isd_play(Jayapura);
temp_int = Jayapura;
cek_tombol_konfirmasi();
if (lokasi == Jayapura)
{
    baca_waktu();
    konversi_waktu_kurang_khusus();
    menit_duhur==113;menit_ashar==113;menit_maghrib==113;menit_isya==113;menit_subuh==113;
    konversi_waktu_tambah();
    arah_kiblat = 360 - baca_eeprom(0x05DE);
    goto finish;
}

finish:
isd_play(lokasi);isd_play(Dipilih);
tulis_eeprom(0x05E0,lokasi);
tulis_eeprom(0x05E2,arah_kiblat);
}

// External Interrupt 1 service routine
interrupt [EXT_INT1] void ext_int1_isr(void)
{
    isd_set();
    ambil_data_kompas();
    cermin = arah_kiblat - 1800;

// Simpangan
if (data_kompas >= cermin && data_kompas <= arah_kiblat)
{
    selisih = arah_kiblat - data_kompas;
    selisih = selisih / 10;
    cek_selisih();
}
if (data_kompas >= cermin && data_kompas > arah_kiblat)
{
    selisih = data_kompas - arah_kiblat;
    selisih = selisih / 10;
    cek_selisih();
}

if (data_kompas < cermin)
{
    selisih = 360 - arah_kiblat + data_kompas;
    selisih = selisih / 10;
    cek_selisih();
}

```



```
// Arah
if (data_kompas < cermin || data_kompas > arah_kiblat)
  {isd_play(Ke);isd_play(Kiri);}
if (data_kompas >= cermin && data_kompas < arah_kiblat)
  {isd_play(Ke);isd_play(Kanan);}
}
```

```
void cek_bulan(void)
{
```

```
rtc_get_date(&tanggal,&bulan,&tahun);
```

```
if (bulan == 1)
```

```
{
  alamat_data_eeprom = 0x0000;
  cek_tanggal();
}
```

```
if (bulan == 2)
```

```
{
  alamat_data_eeprom = 0x0018;
  cek_tanggal();
}
```

```
if (bulan == 3)
```

```
{
  alamat_data_eeprom = 0x0030;
  cek_tanggal();
}
```

```
if (bulan == 4)
```

```
{
  alamat_data_eeprom = 0x0048;
  cek_tanggal();
}
```

```
if (bulan == 5)
```

```
{
  alamat_data_eeprom = 0x0060;
  cek_tanggal();
}
```

```
if (bulan == 6)
```

```
{
  alamat_data_eeprom = 0x0078;
  cek_tanggal();
}
```

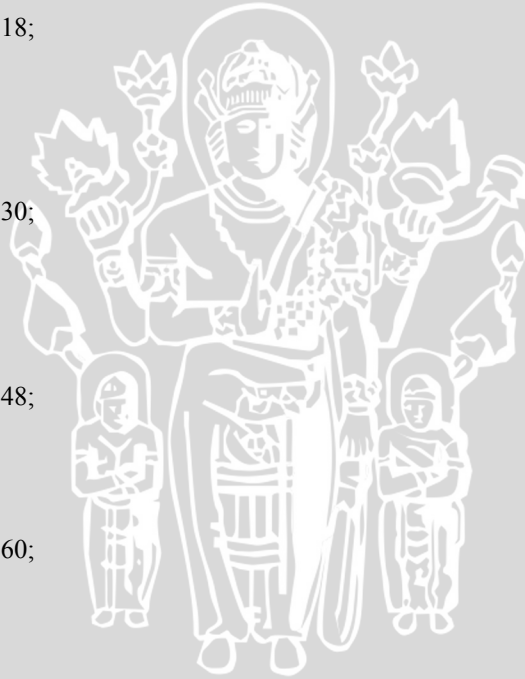
```
if (bulan == 7)
```

```
{
  alamat_data_eeprom = 0x0090;
  cek_tanggal();
}
```

```
if (bulan == 8)
```

```
{
  alamat_data_eeprom = 0x00A8;
  cek_tanggal();
}
```

```
if (bulan == 9)
```



```
{
    alamat_data_eeprom = 0x00C0;
    cek_tanggal();
}
if (bulan == 10)
{
    alamat_data_eeprom = 0x00D8;
    cek_tanggal();
}
if (bulan == 11)
{
    alamat_data_eeprom = 0x00F0;
    cek_tanggal();
}
if (bulan == 12)
{
    alamat_data_eeprom = 0x0108;
    cek_tanggal();
}
}

void cek_tanggal(void)
{
    rtc_get_date(&tanggal,&bulan,&tahun);
    if ((tanggal == 1)|| (tanggal == 2))
    {
        baca_waktu();
        cek_waktu();
    }
    if ((tanggal==3)|| (tanggal==4)|| (tanggal==5))
    {
        alamat_data_eeprom += 2;
        baca_waktu();
        cek_waktu();
    }
    if ((tanggal==6)|| (tanggal==7)|| (tanggal==8))
    {
        alamat_data_eeprom += 4;
        baca_waktu();
        cek_waktu();
    }
    if ((tanggal==9)|| (tanggal==10)|| (tanggal==11))
    {
        alamat_data_eeprom += 6;
        baca_waktu();
        cek_waktu();
    }
    if ((tanggal==12)|| (tanggal==13)|| (tanggal==14))
    {
        alamat_data_eeprom += 8;
        baca_waktu();
        cek_waktu();
    }
    if ((tanggal==15)|| (tanggal==16)|| (tanggal==17))
    {
        alamat_data_eeprom += 10;
        baca_waktu();
        cek_waktu();
    }
}
```

```
if ((tanggal==18)||(tanggal==19)||(tanggal==20))
```

```
{
    alamat_data_eeprom += 12;
    baca_waktu();
    cek_waktu();
}
```

```
if ((tanggal==21)||(tanggal==22)||(tanggal==23))
```

```
{
    alamat_data_eeprom += 14;
    baca_waktu();
    cek_waktu();
}
```

```
if ((tanggal==24)||(tanggal==25)||(tanggal==26))
```

```
{
    alamat_data_eeprom += 16;
    baca_waktu();
    cek_waktu();
}
```

```
if (tanggal==27||tanggal==28||tanggal==29)
```

```
{
    alamat_data_eeprom += 18;
    baca_waktu();
    cek_waktu();
}
```

```
if (tanggal==30||tanggal==31)
```

```
{
    alamat_data_eeprom += 20;
    baca_waktu();
    cek_waktu();
}
```

```
void cek_waktu(void)
```

```
{
    rtc_get_time(&jam,&menit,&detik);
```

```
if (jam == jam_duhur)
```

```
{
    if (menit == menit_duhur)
```

```
{
    if (tanda_duhur == 0)
```

```
{
        tanda_duhur=1;
        tanda_ashar=0;
        tanda_maghrib=0;
        tanda_isya=0;
        tanda_subuh=0;
        suara_adzan();
    }
}
```

```
}
}
```

```
if (jam == jam_ashar)
```

```
{
    if(menit == menit_ashar)
```

```
{
    if (tanda_ashar == 0)
```

```
{
        tanda_duhur=0;
        tanda_ashar=1;
    }
}
```




```

if ((ratusan == 0)&&(puluhan == 0)&&(satuan == 0))
{
    isd_play(Arah);isd_play(Kiblat);isd_play(Sudah);isd_play(Tepat);
    return;
}

```

```

if (ratusan == 0)goto puluhan_;
if (ratusan == 1){ isd_play(Se); }
else
{
    temp = ratusan;
    decoder_suara();
    isd_play(bunyi_angka);
}
isd_play(Ratus);

```

```

puluhan_:
if (puluhan == 0) goto satuan_;
if (puluhan == 1)
{
    if (satuan == 0)
    {
        isd_play(Se);
        isd_play(Puluh);
        goto selesai;
    }
    else if (satuan == 1)
    {
        isd_play(Se);
        isd_play(Belas);
        goto selesai;
    }
    else
    {
        temp = satuan;
        decoder_suara();
        isd_play(bunyi_angka);
        isd_play(Belas);
        goto selesai;
    }
}
}
else
{
    temp = puluhan;
    decoder_suara();
    isd_play(bunyi_angka);
    isd_play(Puluh);
}
}

```

```

satuan_:
if (satuan == 0)goto selesai;
temp = satuan;
decoder_suara();
isd_play(bunyi_angka);

```

```

selesai:
isd_play(Derajat);
}

```



```

void suara_angka(void)
{
angka_temp = angka;
satuan = angka_temp % 10;
puluhan = (angka_temp / 10) % 10;

```

```

if (puluhan == 0) goto satuan_;
if (puluhan == 1)

```

```

{
    if (satuan == 0)
    {
        isd_play(Se);
        isd_play(Puluh);
        return;
    }

```

```

    else if (satuan == 1)
    {
        isd_play(Se);
        isd_play(Belas);
        return;
    }

```

```

    else
    {
        temp = satuan;
        decoder_suara();
        isd_play(bunyi_angka);
        isd_play(Belas);
        return;
    }
}

```

```

else
{
temp = puluhan;
decoder_suara();
isd_play(bunyi_angka);
isd_play(Puluh);
}

```

```

satuan_ :
if (satuan == 0 && puluhan == 0)

```

```

{
    isd_play(Nol);
    return;
}

```

```

temp = satuan;
decoder_suara();
isd_play(bunyi_angka);
}

```

```

void decoder_suara(void)

```

```

{
    if (temp==1)bunyi_angka = 0x00AD;
    if (temp==2)bunyi_angka = 0x00B2;
    if (temp==3)bunyi_angka = 0x00B7;
    if (temp==4)bunyi_angka = 0x00BC;
    if (temp==5)bunyi_angka = 0x00C1;
    if (temp==6)bunyi_angka = 0x00C6;
    if (temp==7)bunyi_angka = 0x00CA;
    if (temp==8)bunyi_angka = 0x00CF;
    if (temp==9)bunyi_angka = 0x00D4;
}

```




```

void cek_tombol_konfirmasi(void)
{
for (a = 0; a < 3; a++)
{
for (b = 0; b < 200; b++)
{
for (c = 0; c < 200; c++)
{ PORTB.3 = 1;
if (PINB.3 == 0)
{
lokasi = temp_int;
a = b = c = 255;
}
}
}
}
}
}
}

```

```

void konversi_waktu_tambah(void)
{
if (menit_duhur >= 60)
{menit_duhur-=60;jam_duhur +=1;}
if (menit_ashar >= 60)
{menit_ashar-=60;jam_ashar +=1;}
if (menit_maghrib >= 60)
{menit_maghrib-=60;jam_maghrib +=1;}
if (menit_isya >= 60)
{menit_isya-=60;jam_isya +=1;}
if (menit_subuh >= 60)
{menit_subuh-=60;jam_subuh +=1;}
}

```

```

void konversi_waktu_kurang(void)
{
jam_duhur-=1;jam_ashar-=1;jam_maghrib-=1;jam_isya-=1;jam_subuh-=1;
menit_duhur+=60;menit_ashar+=60;menit_maghrib+=60;menit_isya+=60;menit_subuh+=60;
}

```

```

void konversi_waktu_kurang_khusus(void)
{
jam_duhur-=2;jam_ashar-=2;jam_maghrib-=2;jam_isya-=2;jam_subuh-=2;
menit_duhur+=120;menit_ashar+=120;menit_maghrib+=120;menit_isya+=120;menit_subuh+=120;
}

```

```

void isd_play(unsigned int alamat_isd)
{
isd_set();
isdl = (unsigned char)alamat_isd;
alamat_isd >>= 8;
isdh = (unsigned char)alamat_isd;
isdh = isdh & 0x03;
PORTB = ((PORTB & 0xFC) | isdh);
PORTA = isdl;
delay_ms(100);
PORTC.7 = 0;
delay_ms(100);
PORTB.2 = 0;
delay_ms(100);
PORTB.2 = 1;
}

```

```
while (PINB.4 == 1);
return;
}

void suara_adzan(void)
{
    isd_set();
    isd_play(adzan);
    if (tanda_subuh == 1)
    {
        isd_play(mode2);
        isd_play(mode1);
    }
    else
    { isd_play(mode1); }
}

void isd_set(void)
{
    PORTC.7=1;
    PORTB.2=1;
}

unsigned char baca_kompas(unsigned char register_kompas)
{
    i2c_start();
    i2c_write(alamat1_kompas);
    i2c_write(register_kompas);
    i2c_start();
    i2c_write(alamat2_kompas);
    data_baca_kompas=i2c_read(0);
    i2c_stop();
    return data_baca_kompas;
}

void ambil_data_kompas(void)
{
    kompas_h = baca_kompas(register1_kompas); //MSB
    kompas_l = baca_kompas(register2_kompas); //LSB
    kompas_h = kompas_h << 8;
    data_kompas = kompas_h | kompas_l;
    data_kompas = data_kompas;
}
```

LAMPIRAN 6

Datasheet Komponen

1. *Datasheet* Sensor Magnet CMPS03
2. *Datasheet* RTC DS1302
3. *Datasheet* EEPROM AT24C64
4. *Datasheet* Mikrokontroler ATMEGA8535
5. *Datasheet* ISD2590
6. *Datasheet* LM386

