

PERANCANGAN JAM WAKTU SHOLAT *REALTIME* BERDASARKAN GPS *RECEIVER* BERBASIS MIKROKONTROLER ARDUINO

Muhammad Nufail¹, Cahya Bagus Sanjaya²

^{1,2}Program Studi Teknik Informatika, Universitas Yudharta Pasuruan
Pasuruan, Indonesia

Email:m.nufail@gmail.com¹,cbsanjaya@yudharta.ac.id²

Received : September, 2018

Accepted : Nopember, 2018

Published : Desember, 2018

ABSTRACT

Prayer is an obligation for every Muslim that must be carried out at a specified time. The design of the realtime prayer time clock based on the Arduino microcontroller aims to make it easier to find out the prayer time in real terms based on the clock and the accurate coordinates of the location with the Hashb Auqot as-Shalâh formula. This prayer time clock uses several modules, namely: 1) RTC as a timer, 2) GPS receiver to receive coordinate data and UTC time, 3) Bluetooth as a communication medium with remote settings that uses an Android Smart Phone to manage text information, ihtiyâth, display and murottal, 4) micro-SD card adapter as murottal file reader stored in micro-SD, and 5) Dot Matrix Display (DMD) P10 as the output media displaying running text.

Keywords : prayer times, Arduino, GPS.

ABSTRAK

Sholat adalah kewajiban bagi setiap Muslim yang harus dilaksanakan pada waktu yang ditentukan. Desain jam waktu sholat real-time berdasarkan mikrokontroler Arduino bertujuan untuk membuatnya lebih mudah untuk mengetahui waktu sholat secara riil berdasarkan jam dan koordinat akurat lokasi dengan formula Hashb Auqot as-Shalâh. Jam waktu sholat ini menggunakan beberapa modul, yaitu: 1) RTC sebagai penghitung waktu, 2) Penerima GPS untuk menerima data koordinat dan waktu UTC, 3) Bluetooth sebagai media komunikasi dengan pengaturan jarak jauh yang menggunakan Ponsel Pintar Android untuk mengelola informasi teks, ihtiyth, display, dan murottal, 4) adaptor kartu micro-SD sebagai pembaca file murottal yang disimpan dalam micro-SD, dan 5) Dot Matrix Display (DMD) P10 sebagai media output menampilkan teks yang sedang berjalan.

Kata kunci : Waktu Sholat, Arduino, GPS.

1. PENDAHULUAN

1.1 Latar Belakang

Ibadah sholat merupakan kewajiban bagi setiap orang islam. Pada zaman Rasulullah ilmu perhitungan waktu masih belum ditemukan, untuk menentukan waktu sholat masih menggunakan gejala alam dengan melihat langsung ke matahari atau menggunakan bayangan benda yang terkena sinar matahari sesuai dengan petunjuk Nabi. Pada masa keemasan Islam perkembangan ilmu semakin pesat yang mana dalam menentukan masuk dan berakhirnya waktu shalat di suatu daerah bisa dihitung menggunakan formula trigonometri dengan data titik kordinat bujur dan lintang tempat beserta data ephemeris matahari yang hasil perhitungannya kemudian dituliskan dalam bentuk tabel jadwal waktu sholat dan beredar sampai sekarang.

Dewasa ini seiring dengan berkembangnya teknologi yang dapat digunakan untuk berbagai macam keperluan, manusia dapat membuat berbagai alat bantu dalam menjalankan kegiatan, salah satunya membuat jadwal waktu sholat yang tentunya bukan hanya printout berbentuk tabel pada kertas saja, namun juga menggunakan perangkat elektronika seperti jam digital waktu sholat berbasis mikrokontroler seperti halnya pada kebanyakan masjid saat ini dimana sebelum masuk waktu sholat biasa diputar murottal al-Qur'an dan juga menampilkan waktu sholat dan *running text* yang berisi informasi nama dan lokasi masjid dengan tujuan mempermudah umat muslim dalam melakukan ibadah sholat.

Pada penelitian [1] yang berfokus pada pembuatan jadwal waktu sholat menggunakan mikrokontroler atmega 32 dengan *output display* yang berupa *sevent segment*, dengan hasil yang diperoleh dapat mempermudah dalam mengetahui masuknya waktu sholat. Penelitian sejenis dilakukan oleh Rajasali Muhammad [2] dengan menggunakan data waktu sholat yang dipakai dalam penelitian ini masih mengambil dari tabel jadwal terbatas pada kota Semarang yang diambil dari laman web RHI. Hasil yang diambil dari tabel jadwal kemudian dimasukkan ke memori EEPROM pada mikrokontroler ini tidak bisa digunakan untuk kota-kota selain Semarang.

Untuk membuat jadwal waktu sholat berbasis digital dibutuhkan formulasi perhitungan waktu yang biasa disebut hisab/perhitungan falak/astronomi dimana datanya mengacu pada koordinat lintang dan bujur tempat *musholli* (orang yang

melaksanakan sholat) berada juga ketinggian lokasi dan data matahari yang meliputi ketinggian, deklinasi, semidiameter matahari digunakan untuk mencari selisih waktu istiwa' dengan waktu setempat (WIB/WITA/WIT) karena output hisab waktu sholat adalah menggunakan waktu istiwa' dengan permulaan ketika matahari berada di titik kulminasi. Dalam hal ini telah dilakukan penelitian yang dilakukan oleh Ery Safryanti dkk.[3] hasil dari penelitian ini menunjukkan jadwal waktu sholat secara realtime menggunakan formula perhitungan waktu sholat, namun waktu yang ditunjukkan pada jam digital ini masih bisa berubah karena bersumber dari RTC (*Realtime Time Clock*) yang membutuhkan supply dari baterai sehingga ketika baterai melemah akan mempengaruhi akurasi waktu yang dihasilkan.

Untuk mengatasi kurang akurasi jam, titik koordinat dan data waktu sholat pada penelitian sebelumnya, dalam penelitian ini penulis mencoba menggunakan *realtime* Global Positioning System (GPS) untuk pengambilan data waktu (detik, menit, jam, tanggal, bulan dan tahun). GPS ini berfungsi sebagai sinkronasi waktu pada modul RTC sebagaimana pada penelitian [4] yang melakukan sinkronasi transmisi menggunakan GPS juga sesuai dengan yang dijelaskan oleh Yu-Ling Hsueh [5] bahwasanya Titik GPS adalah sepasang koordinat garis bujur dan lintang dengan stempel waktu yang dikumpulkan oleh perangkat GPS tersebut.

Sebelumnya telah dilakukan penelitian[6] perancangan jadwal waktu sholat berbasis Android yang memanfaatkan GPS sebagai sumber data koordinat dan Signal dari BTS provider sebagai sumber waktu berjalan, hal serupa juga banyak ditemukan aplikasi Android yang bisa didapatkan di PlayStore.

1.2 Rumusan Masalah

Berdasarkan dari uraian latar belakang di atas maka dapat dirumuskan suatu permasalahan, antara lain :

1. Bagaimana formulasi perhitungan waktu sholat metode *Hisab Tahqiqi bit Tahqiq Irsyadul Murid* pada masing-masing daerah berdasarkan koordinat?
2. Bagaimana perancangan dan membuat sistem jadwal sistem jadwal waktu sholat berbasis mikrokontroler Arduino Mega dengan integrasi data *realtime* dari modul GPS (*Global Positioning System*) sebagai acuan waktu.?
3. Bagaimana cara membuat alarm murottal/tarhim berdasarkan waktu sholat menggunakan mikrokontroler Arduino Mega?
4. Bagaimana cara mengintegrasikan modul GPS, Bluetooth, RTC dan DMD dengan mikrokontroler Arduino Mega 2560?

1.3 Tujuan Penelitian

Berdasarkan dari uraian latar belakang di atas maka dapat dirumuskan suatu permasalahan, antara lain :

1. Bagaimana formulasi perhitungan waktu sholat metode *Hisab Tahqiqi bit Tahqiq Irsyadul Murid* pada masing-masing daerah berdasarkan koordinat?
2. Bagaimana perancangan dan membuat sistem jadwal sistem jadwal waktu sholat berbasis mikrokontroler Arduino Mega dengan integrasi data *realtime* dari modul GPS (*Global Positioning System*) sebagai acuan waktu.?
3. Bagaimana cara membuat alarm murottal/tarhim berdasarkan waktu sholat menggunakan mikrokontroler Arduino Mega?
4. Bagaimana cara mengintegrasikan modul GPS, Bluetooth, RTC dan DMD dengan mikrokontroler Arduino Mega 2560?

1.4 Manfaat Penelitian

Hasil dari penelitian ini diharapkan dapat bermanfaat untuk :

1. Membantu kaum muslimin agar lebih mudah mengetahui masuknya waktu sholat pada umumnya.
2. Membantu muadzin atau takmir masjid ketika lupa tidak menyalakan tarhim penanda menjelang masuknya waktu sholat.
3. Masyarakat umum bisa mengetahui waktu secara riil berdasarkan Global Positioning System (GPS).
4. Masyarakat bisa mengetahui kalender Masehi maupun Hijriyah sesuai dengan ketetapan pemerintah dengan akurasi yang lebih baik.
5. Tambahan reverensi akademik bagi universitas atau umum.
6. Mengaplikasikan ilmu yang didapat penulis selama berada di perkuliahan.

2. TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Penelitian yang dilakukan pada Tugas Akhir ini mengacu pada beberapa jurnal yang membahas tema yang sama dengan penambahan GPS sebagai penyempurnaan dari sistem, beberapa penelitian sebelumnya yaitu sebagaimana berikut ini :

Judul	Jurnal
Alat pengingat waktu sholat di masjid berbasis raspberry pi	Prosiding SNATIF Ke-4, 2017
Perancangan pengingat waktu sholat menggunakan dot matriks berbasis mikrokontroler	Pelita Informatika Budi Darma, Volume : VII, Nomor: 2,

at89s52	Agustus 2014
Mobile Prayer Times and Qiblat Direction Using GPS	APACE 2007
Perancangan Perangkat Keras Sistem Pengingat Waktu Sholat Elektronik	Sekolah Tinggi Teknik Harapan Medan
Rancang Bangun Indikator Jam Sholat Abadi Menggunakan Atmel 89S52	Petrokimia Indonesia 2008
<i>Time Synchronization for Transmission Substations Using GPS and IEEE 1588</i>	CSEE Journal of Power and Energy Systems, Vol. 2, No. 3, Sept 2016
<i>Map Matching for Low-Sampling-Rate GPS Trajectories by Exploring Real-time Moving Directions</i>	Information Sciences, 2017

2.2 Definisi Perancangan

Perancangan sistem menurut Zakiyudin [7] adalah penentuan proses dan data yang diperlukan oleh sistem baru, jika sistem itu berbasis komputer, maka rancangan dapat menyertakan spesifikasi jenis peralatan yang digunakan.

Berdasarkan definisi di atas menurut hemat penulis perancangan merupakan suatu kegiatan yang dilakukan untuk mengatasi masalah yang dihadapi baik oleh individu maupun organisasi setelah melakukan analisis.

2.3 Waktu Sholat

Menurut kesepakatan ulama fiqih berdasarkan al-Qur'an dan hadits, sholat yang diwajibkan ada 5 yaitu Dhuhur, Ashar, Maghrib, Isya' dan Shubuh, namun waktu yang dihitung juga termasuk thulu' dan dluha yang dilakukan pada waktu yang telah ditentukan berdasarkan ketinggian matahari.

Untuk menghitung masing-masing waktu sholat seperti yang ditunjukkan pada Gambar 2.3 dibutuhkan beberapa data antara lain [8]:

- Lintang tempat (ϕ)
- Bujur tempat (λ)
- Tinggi Tempat (TT)
- Time Zone / Zona waktu daerah (TZ).
- Tinggi matahari masing-masing waktu sholat (h)
- Deklinasi matahari menurut tanggal, bulan dan tahun Masehi (δ)
- Semi diameter matahari menurut tanggal, bulan dan tahun Masehi (sd)
- Equation of Time/perata waktu menurut tanggal, bulan dan tahun Masehi (λ).

Data-data tersebut di atas didapat dari module GPS, module Bluetooth dan perhitungan astronomis. Untuk perhitungan astronomis adalah data matahari meliputi Deklinasi Matahari, Semi Diameter Matahari dan Equation of Time, berikut ini rumus perhitungan Data Matahari berdasarkan kalender Masehi.

D = Tanggal Masehi.

M = Apabila Bulan 1 atau 2 maka Bulan Masehi ditambah 1 apabila tidak maka tetap.

Y = Apabila Bulan 1 atau 2 maka Tahun Masehi ditambah 1 apabila tidak maka tetap.

B = $-(\text{Int}(Y / 100)) + \text{Int}(\text{Int}(Y / 100) / 4) + 2$

H = Jam

JD = $1720994.5 + \text{Int}(365.25 * Y) + \text{Int}(30.6001 * (M + 1)) + B + D + (H / 24) - TZ / 24$

Julian Date (JD) ini adalah sistem kalender setelah tanggal 15 Oktober 1582, dari Julian Date bisa didapatkan data-data matahari dan hari pasaran jawa.

$T = (JD - 2451545) / 36525$

$S = \text{Mod}(280.46645 + 36000.76983 * T), 360$

$m = \text{Mod}(357.5291 + 35999.0503 * T), 360$

$N = \text{Mod}(125.04 - 1934.136 * T), 360$

$K' = (17.264 / 3600) * \text{Sin}(N) + (0.206 / 3600) * \text{Sin}(2 * N)$

$K'' = (-1.264 / 3600) * \text{Sin}(2 * S)$

$R' = (9.23 / 3600) * \text{Cos}(N) - (0.09 / 3600) * \text{Cos}(2 * N)$

$R'' = (0.548 / 3600) * \text{Cos}(2 * S)$

$Q' = 23.43929111 + R' + R'' - (46.815 / 3600) * T$

$$E = (6898.06 / 3600) \times \sin(m) + (72.095 / 3600) \times \sin(2 \times m) + (0.966 / 3600) \times \sin(3 \times m)$$

$$S' = \text{Mod}(S+E+K'+K'' - (20.47 / 3600), 360)$$

$$\delta = \sin^{-1}(\sin S' \times \sin Q')$$

$$e = (-1.915 \times \sin(m) - 0.02 \times \sin(2 \times m) + 2.466 \times \sin(2 \times S') - 0.053 \times \sin(4 \times S')) / 15$$

$$s.d = 0.267 / (1 - 0.017 \times \cos(m))$$

Dimana variabel :

- 1) D : *day* atau tanggal yang dihitung.
- 2) M : *month* atau bulan yang dihitung
- 3) Y : *year* atau bulan yang dihitung.
- 4) B : koreksi hari antara *Gregorian Calendar* (sebelum tanggal 15 Oktober 1582) dengan *Julian Calendar* (setelah 15 Oktober 1582).
- 5) JD : *Julian Date* yang berupa angka float yang sistem perhitungannya dilakukan setelah Kaisar Julius berkuasa di Romawi (Italia).
- 6) T : *Juz'ul Ashl al-Miladiy*.
- 7) S : *Wasath as-Syamsi* adalah *Thul as-Syamsi* yang belum dikoreksi.
- 8) m : *Khôsshoh as-Syamsi* jarak titik pusat matahari dari titik *Cancer* (Saronon) sepanjang lingkaran ekliptika.
- 9) N : *'Uqdatus Syamsi* adalah titik simpul matahari atau titik perpotongan antara lintasan matahari dan ekliptika.
- 10) K', K'', R' dan R'' : Koreksi pertama, kedua dan ketiga dari N.
- 11) Q' : *Mail as-Syamsi al-Kulliy* yaitu nilai maksimum deklinasi matahari.
- 12) E : *Ta'dil as-Syamsi* yaitu koreksi matahari.
- 13) S' : *Thul as-Syamsi* atau *Longitude of Sun* yaitu jarak titik pusat matahari dari titik Vernal Equinok (titik *haml*) sepanjang lingkaran Ekliptika.
- 14) δ : Lambang delta ini adalah *Mail as-Syamsi* atau deklinasi matahari.
- 15) e : *Equation of Time* atau perata waktu yaitu selisih antara kulminasi matahari hakiki dengan waktu kulminasi matahari rata-rata (jam 12:00).
- 16) s.d: *Nishf Quthris Syamsi* atau semi diameter matahari yaitu jarak titik pusat sampai dengan piringan luar matahari.

Setelah diketahui deklinasi matahari, perata waktu dan semi diameter matahari selanjutnya melakukan perhitungan waktu sholat yang berdasarkan waktu istiwa' dan rumusnya adalah sebagai berikut :

$$Wis = \cos^{-1}(-\tan \varphi \cdot \tan \delta + \sin h / \cos \varphi / \cos \delta) / 15$$

Dimana :

φ = Lintang daerah.

δ = Deklinasi matahari.

h = Ketinggian matahari.

Ketinggian matahari masing-masing waktu sholat adalah sebagai berikut :

$$\text{Thulu' / Maghrib : } h = -(sd + (34.5/60) + \text{Dip}) - 0.0024$$

$$\text{Isya' : } -18$$

$$\text{Shubuh : } -20$$

$$\text{Dluha : } 4.5$$

$$\text{Ashar : } \tan h = \frac{1}{\tan(\varphi - \delta + 1)}$$

Untuk mengkonversi Waktu Istiwa' ke dalam Waktu Daerah (misal : WIB) membutuhkan Koreksi Waktu Daerah, yaitu dengan rumus :

$$\text{KWD} = \frac{TZ \times 15 - \lambda}{15} - e$$

$$\text{WD} = \text{Wis} + \text{KWD}$$

Rumus waktu sholat pada dasarnya menggunakan waktu istiwa' yang mengacu pada saat kulminasi (titik tertinggi matahari) matahari yang terbagi menjadi 2 bagian yaitu :

- 1) Posisi matahari pasca kulminasi berada di barat yaitu waktu Ashar, Maghrib dan Isya'

$$\text{Rumus : } 12 + W$$

- 2) Posisi matahari pra kulminasi berada di timur yaitu waktu Shubuh, Thulu' dan Dluha.

$$\text{Rumus : } 12 - W$$

2.4 Rumus Waktu dengan Bayangan Sinar Matahari

Rumus untuk mengetahui waktu atau jam dengan bantuan bayangan benda yang terkena sinar matahari ini digunakan untuk pengujian akurasi jam pada RTC (*Real Time Clock*) yang didapatkan dari GPS.

Rumus mengetahui "h" :

$\tan h$ = tinggi benda dibagi (-:-) bayangannya.

Contoh :

Tinggi benda = 14.8 cm, bayangan = 11 cm.

$$\tan^{-1}(14.8 / 11) = 53.37867251 = 53^{\circ} 22' 43.22''$$

Selanjutnya adalah menghitung waktu dengan menggunakan data tinggi matahari 'h' yang sudah diketahui sama dengan menghitung waktu sholat.

2.5 Mikrokontroler Arduino

2.5.1 Definisi

Mikrokontroler adalah sebuah alat yang berfungsi sebagai pengendali (*controller*) yang dikemas dalam bentuk chip atau IC (*integrated circuit*) yang bisa diprogram menggunakan komputer. Tujuan menanamkan program pada mikrokontroler adalah agar rangkaian elektronik dapat membaca input, memproses input tersebut dan kemudian menghasilkan output sesuai yang diinginkan. Jadi mikrokontroler bertugas sebagai 'otak' yang mengendalikan input, proses dan output sebuah rangkaian elektronik.

Arduino adalah kit elektronik atau papan rangkaian elektronik open source yang di dalamnya terdapat komponen utama yaitu sebuah chip mikrokontroler dengan jenis AVR dari perusahaan Atmel [9].

2.5.2 Arduino Mega

Papan Arduino Mega ini menjadi pilihan penulis pada penelitian ini karena memiliki pin I/O yang cukup banyak, sejumlah 54 buah digital I/O pin dimana 15 pin diantaranya adalah PWM, 16 pin analog input, 4 pin UART (serial port hardware). Arduino Mega 2560 dilengkapi dengan sebuah oscillator 16 Mhz, sebuah port USB, power jack DC, ICSP header, dan tombol reset. Board ini sudah sangat lengkap, sudah memiliki segala sesuatu yang dibutuhkan untuk sebuah mikrokontroler. Dengan penggunaan yang cukup sederhana, anda tinggal menghubungkan power dari USB ke PC anda atau melalui adaptor AC/DC ke jack DC.

2.6 Hardware Module

2.6.1 GPS

1) Pengertian GPS

Menurut Jochen Schille dan Agnes Voisar [10] GPS adalah singkatan dari *Global Positioning System* yaitu sistem navigasi yang menggunakan satelit yang didesain agar dapat menyediakan posisi secara instan, kecepatan dan informasi waktu di hampir semua tempat di muka bumi, setiap saat dan dalam kondisi cuaca apapun.

Sedangkan alat untuk menerima sinyal satelit yang dapat digunakan oleh pengguna secara umum dinamakan GPS Tracker atau GPS Tracking, dengan menggunakan alat ini maka dimungkinkan user dapat melacak lokasi secara *realtime*.

2) Cara Kerja GPS

Bagian yang paling penting dalam sistem navigasi GPS adalah beberapa satelit yang berada di orbit bumi atau yang sering kita sebut di ruang angkasa. Satelit GPS saat ini berjumlah 24 unit yang semuanya dapat memancarkan sinyal ke bumi yang lalu dapat ditangkap oleh alat penerima sinyal tersebut atau GPS Tracker. Selain satelit terdapat 2 sistem lain yang saling berhubungan, sehingga jadilah 3 bagian penting dalam sistem GPS. Ketiga bagian tersebut terdiri dari *GPS Control Segment* (Bagian Kontrol), *GPS Space Segment* (bagian angkasa), dan *GPS User Segment* (bagian pengguna).

2.6.2 Bluetooth

Bluetooth adalah peralatan yang digunakan untuk menghubungkan perangkat satu dengan perangkat lainnya tanpa menggunakan media kabel, misalnya *smartphone* dengan *smartphone* dan perangkat lain yang terpasang bluetooth. Bluetooth beroperasi pada frekuensi 2,4 GHz dengan menggunakan frekuensi hopping tranceiver yang mampu menyediakan layanan komunikasi tanpa kabel dengan jarak yang terbatas [11]. Salah satu hasil contoh modul *Bluetooth* yang paling banyak digunakan adalah tipe HC-05. modul *Bluetooth* HC-05 merupakan salah satu modul *Bluetooth* yang dapat ditemukan dipasaran dengan harga yang relatif murah. Modul *Bluetooth* HC-05 terdiri dari 6 pin konektor, yang setiap pin konektor memiliki fungsi yang berbeda - beda. Modul Bluetooth HC-05 ini memiliki 6 Pin yang masing-masing memiliki fungsi sebagai berikut :

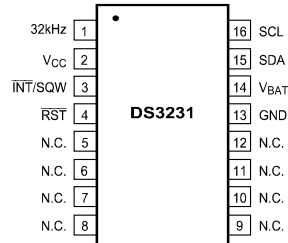
Tabel 2.1 Pin Mapping Modul Bluetooth HC-05

Pin	Fungsi
VCC	Sumber tegangan 5V
GND	Ground
TXD	Mengirim data
RXD	Menerima Data
KEY	Pin ini hanya dipakai pada module Bluetooth yang bisa menjadi Master
State	Mengaktifkan indikator LED ketika ada koneksi atau tidak.

Sumber: Datasheet HC-05 Bluetooth

2.6.3 RTC (Real Time Clock)

Real Time Clock merupakan suatu chip atau IC (*Integrated Circuit*) yang memiliki fungsi sebagai penyimpan waktu dan tanggal. RTC memiliki banyak tipe, namun dalam penelitian ini penulis menggunakan RTC DS3231 mengingat *timing precision* yang lebih akurat dibanding dengan RTC yang lain [12].



Gambar 2.1 IC RTC DS3231

(Sumber: Datasheet DS3231 Module)

Modul ini berupa *Integrated Circuit* (IC) yang memiliki 6 buah *pin* yaitu ; X1 dan X2 yang dihubungkan ke sebuah Kristal, Vbat yang dihubungkan ke sebuah baterai CMOS bertegangan 3V, GND yang dihubungkan ke ground, SDA dan SCL yang dihubungkan ke kontroler dan VCC yang dihubungkan ke tegangan +5v.

2.6.4 Mini DF-Player

Modul ini berfungsi untuk membaca file yang berformat mp3 pada sebuah *sd-card* yang dikirim melalui pin TX dan diterima oleh mikro kontroler melalui pin RX dan untuk menerima perintah dari mikro kontroler melalui pin TX untuk memutar, *pause* dan *stop* sebuah mp3 melalui pin RX.

2.6.5 DMD (Dot Matrix Display)

DMD merupakan salah satu modul display yang populer di kalangan umum karena kemudahan dalam menggunakan modul ini dengan adanya *library* DMD yang telah dikembangkan oleh FreeStonic.

Modul ini membutuhkan konektor 16 pin, dengan konfigurasi :

Tabel 2.2 Tabel Konfigurasi Pin DMD

Con	Pin Mapped	Con	Pin Mapped
1	PD 46 (OE)	9	Ground
2	PD 47 (A)	10	PD 49 (Store)
3	Ground	11	Ground
4	PD 48 (B)	12	PD 51 Data In (MOSI)
5	Ground	13	Ground
6	Not Use	14	-
7	Ground	15	Ground
8	PD 52 (Clock)	16	-

Sumber : Datasheet DMD

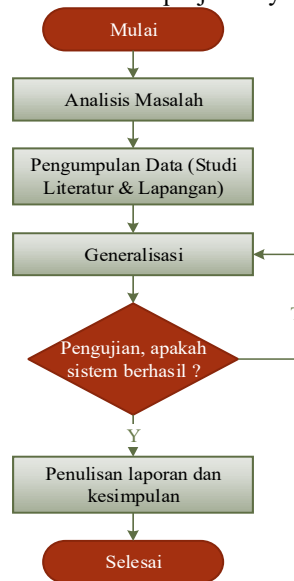
3. METODE PENELITIAN

3.1 Tahapan Penelitian

Tahapan proses penelitian merupakan serangkaian kegiatan yang ditempuh peneliti menurut prosedur dan proses yang benar serta akurat, sehingga hasil yang dicapai diyakini benar, dapat dipercaya dan berdaya guna serta diakui secara ilmiah.

Menurut Nachmias [13] menyatakan bahwa :

“Proses penelitian itu dimulai dari masalah dan diakhiri dengan generalisasi. Apabila kegiatan itu telah berakhir, maka akan dilanjutkan "*cyclus*" (merupakan kegiatan berulang) dan "*self-correcting*"; yang dimaksud dengan *self-correcting* adalah generalisasi tentatif diuji secara logika dan empiris. Apabila ditolak, maka diformulasikan lagi dan diuji lagi. Dalam setiap reformulasi itu semua pelaksanaan penelitian dinilai kembali, sehingga sesuatu yang tidak sah diperbaiki atau disempurnakan.”

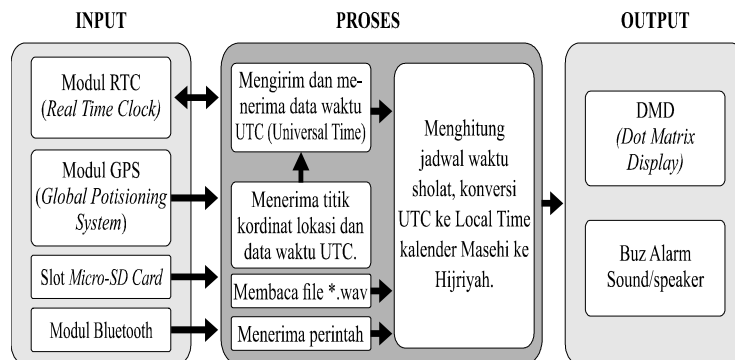


Gambar 3.1 Flowchart Tahapan Penelitian

3.2 Rancangan Alur Sistem

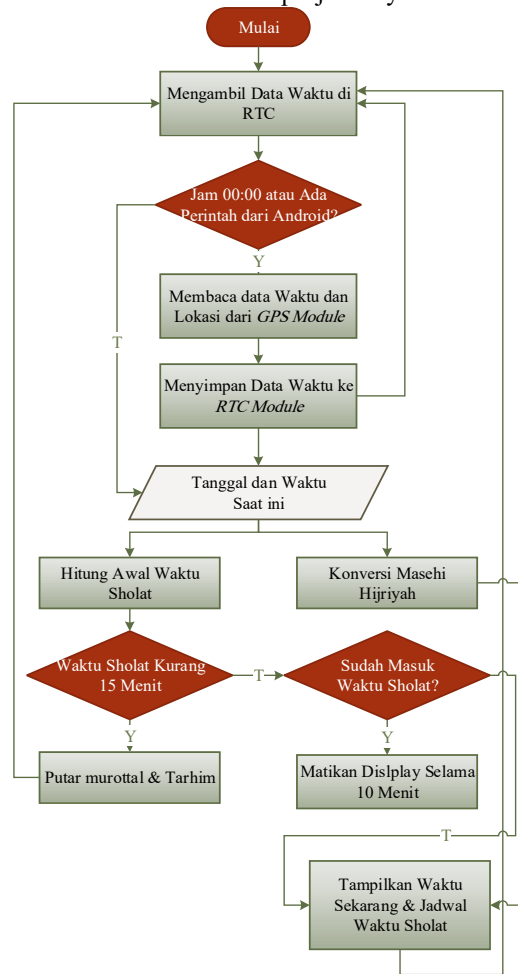
Penelitian yang dilakukan dalam tugas akhir ini menitikberatkan pada proses pengambilan data waktu secara real yang meliputi: detik, menit, jam, tanggal, bulan dan tahun dari sebuah chip GPS yang nantinya akan dikirim ke dalam *Real Time Clock* kemudian digunakan sebagai variabel untuk proses perhitungan waktu sholat.

Penelitian yang dilakukan dapat dijelaskan dengan secara rinci melalui blok diagram seperti yang terlihat pada gambar 3.2.



Gambar 3.2 Blok Diagram Sistem

Dari blok diagram pada Gambar 3.2 menunjukkan alur sistem yang terbagi menjadi 3 kelompok bagian, yaitu bagian *input*, proses dan *output*. Dalam tugas akhir ini, penulis menggunakan modul RTC, GPS dan Mini DF-Player sebagai *input*-nya, papan Arduino Mega atau atmega 3208P sebagai kontrolernya, *dot matrix display*, Buzz Alarm sound/speaker sebagai outputnya. Untuk lebih jelasnya bisa dilihat pada flowchart yang ditunjukkan pada gambar 3.3 di bawah ini :

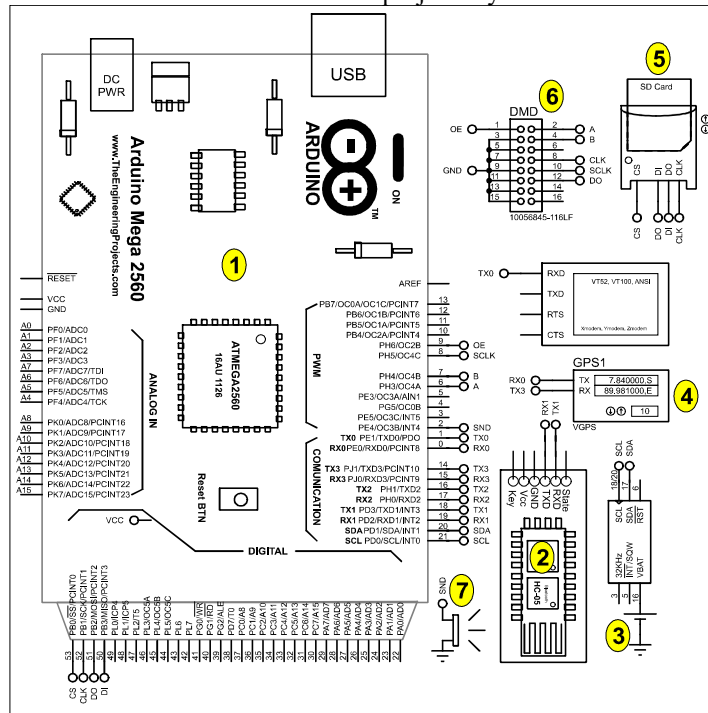


Gambar 3.3 Flowchart Kerja Sistem

Rancangan alur kerja sistem yang ditunjukkan pada Gambar 3.3 dimulai dari pengambilan detak waktu yang ada pada modul RTC, kemudian dilihat apakah jam menunjukkan pukul 00:00 waktu setempat atau apakah ada perintah dari pengguna melalui *Smart Phone Android* melalui Bluetooth, kalau terdapat perintah atau waktu menunjukkan pukul 00:00 maka detak jam pada RTC akan diperbarui yang diambil dari modul GPS begitu pula dengan titik koordinat lokasi. Setelah didapat data waktu dan lokasi proses utamanya menghitung waktu sholat dan konversi kalender yang hasilnya ditampilkan pada *display LED matrix* p10 dengan ukuran 16x32 sebanyak 3 *panel*, apabila 15 menit menjelang waktu sholat masuk, maka alarm yang berupa murottal dan tarhim akan dijalankan, setelah masuk waktu sholat maka *display* akan dimatikan untuk kekhusyukan beribadah.

3.3 Rancangan Project

Beberapa *hardware* pada perancangan *project* jam waktu sholat yang diperlukan di dalam penelitian ini adalah *power supply*, *Input/Output Device* dan kontroler untuk lebih jelasnya bisa dilihat pada gambar 3.4:



Gambar 3.4 Rancangan Project Jam Waktu Sholat

Dalam pembuatan *project* jam waktu sholat pada penelitian ini membutuhkan 7 module hardware seperti yang ditunjukkan pada Gambar 3.4, yaitu sebagaimana pada tabel 3.1 berikut ini:

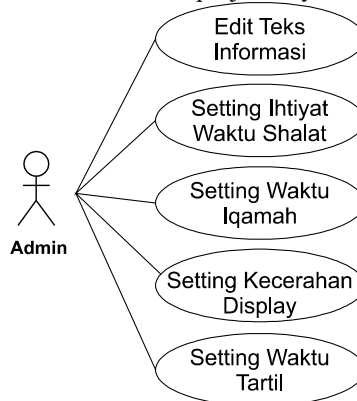
Tabel 3.1 Alat dan Bahan yang akan digunakan dalam Penelitian

Alat / Bahan	Spesifikasi
Mikrokontroler	Arduino Mega2560
Display	Dot Matrix Display
IC RTC	Modul RTC DS3231
Amplifier	DF-Player mini mp3 modul
Modul GPS	GPS SiRF Star III
Bluetooth module	Bluetooth HC-05
Power Supply	PSU 5 Volt 20 Ampere
Breadboard	Breadboard 400 points
Kabel Jumper	Male to Male
Pin Konektor	20 x 2 pin
Kabel listrik	Kabel halus 2 meter
Multi meter	Avometer analog
Timah	Timah standar
Solder	Solder 40 W 220 V
Kabel data	Kabel data standar android dan kabel printer.

Pada dasarnya masing-masing module yang tertulis pada tabel 3.1 tersebut memiliki fungsi yang terbagi menjadi 3 golongan yaitu *input*, kontroler dan *output*.

3.4 Use Case Diagram

Use Case Diagram merupakan salah satu hal yang penting dilakukan untuk memetakan pekerjaan yang bisa dilakukan oleh user pada suatu sistem. *Use Case Diagram* pada penelitian ini bisa dilihat pada Gambar 3.7 di bawah ini :



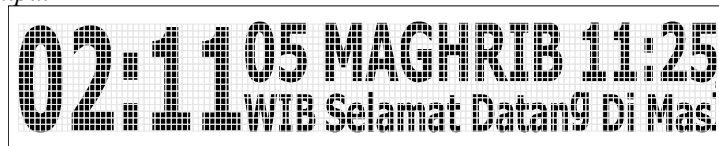
Gambar 3.5 Use Case Diagram

Untuk mempermudah *user*, pada project jam waktu shalat ini dilengkapi dengan pengaturan seperti yang ditunjukkan pada Gambar 3.7 yang bisa dilakukan oleh *user* dengan mudah melalui aplikasi di *Smart Phone* Android.

3.5 Rancangan Hasil

Rancangan dari hasil yang dicapai dalam penelitian ini didapat dari 2 arah, yaitu display jam waktu sholat dan aplikasi *remote control* di *Smartphone* Android yang berfungsi untuk mengirim text informasi, memerintahkan mikrokontroler untuk meng-*update* lokasi dan waktu berdasarkan GPS, mengatur *ihthyat* dalam jadwal sholat dan mengatur tampilan sesuai dengan opsi yang diberikan.

1) Rancangan Display Output

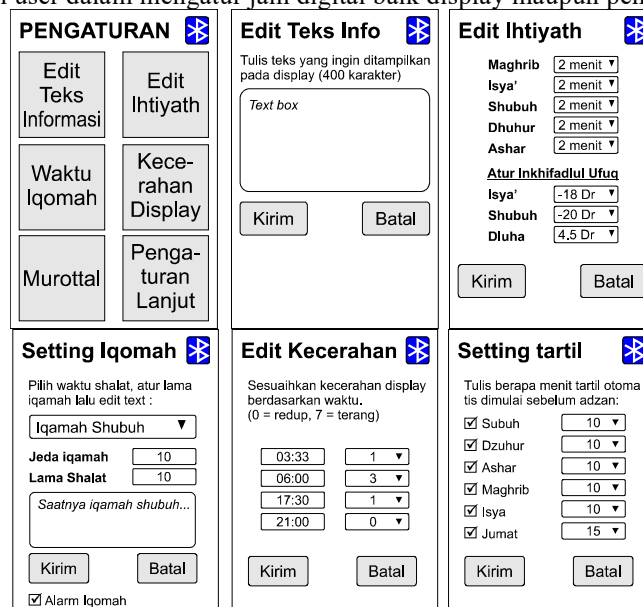


Gambar 3.6 Rancangan Display Output

Rancangan Display output yang dihasilkan nanti berupa jam WIB di sebelah kiri, informasi jadwal waktu sholat *blink* di sebelah kanan atas dan teks informasi berjalan di bawahnya yang terdapat hasil konversi kalender di dalam teks informasi tersebut.

2) Rancangan Remote Control via Smartphone Android

Rancangan ini dirancang menggunakan google appinventor yang berguna sebagai alat kontrol dengan tujuan untuk mempermudah penggunaan bagi user dalam mengatur jam digital baik display maupun pengaturan lain.



Gambar 3.7 Rancangan Remote via Smart Phone

3.6 Pengujian

Pada penelitian ini terdapat beberapa pengujian yang perlu dilakukan untuk hasil yang lebih baik yaitu akurasi GPS,

sensifitas bluetooth, akurasi alarm, akurasi perhitungan dan akurasi RTC.

4. HASIL DAN PEMBAHASAN

4.1 Perakitan Komponen *Hardware*

Proses perakitan yang penulis lakukan tidak menggunakan papan PCB melainkan menggunakan *Breadboard* yang membutuhkan kabel *jumper* untuk koneksi antar pin pada *hardware*.



Gambar 4.1 Interfacing Module

Pada gambar 4.1 di atas terdapat komponen-komponen yang dibutuhkan sesuai dengan perencanaan pada perancangan yaitu 1) *Power Supply 5v 20A*, 2) *Arduino Mega 2560*, 3) *Modul RTC*, 4) *Modul GPS*, 5) *Modul Bluetooth*, 6) *Mini DF-Player* dan 7) *Mini Sound*

Ketujuh komponen tersebut dimasukkan ke dalam *box-frame DMD (Dot Matrix Display)*, kemudian pin konektor *male DMD* yang tidak terpasang pada Gambar 4.1 tersebut dipasang pada pin konektor *female* pada modul *DMD*. Untuk lebih jelasnya *interfacing* masing-masing komponen bisa dilihat pada Tabel 4.1 di bawah ini :

Tabel 4.1 Interfacing Module

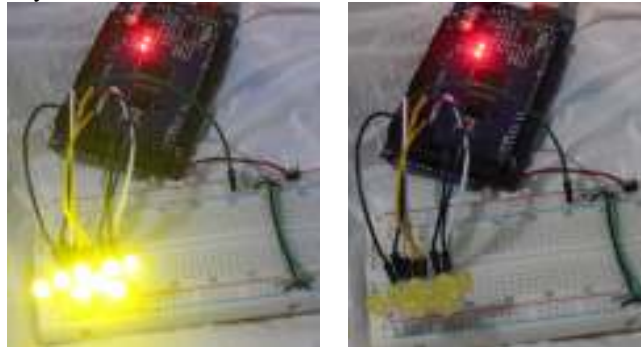
No	Komponen	Pin Module	Pin Arduino
1	Power Supply 5V	V+	Vin
		V-	GND
2	RTC (<i>Real Time Clock</i>)	VCC	5V
		GND	GND
		SCL	SCL
		SDA	SDA
3	Bluetooth	VCC	5V
		GND	GND
		TX	RX1
4	GPS (<i>Global Positioning System</i>)	VCC	5V
		GND	GND
		TX	RX2
5	Mini DF-Player	VCC	V5
		GND	GND
		TX	RX3
		RX	TX3
6	DMD (<i>Dot Matrix Display</i>)	OE	PD46
		A	PD47
		B	PD48
		SCLK	PD49
		MOSI	PD51
	CLK	PD 52	

Setelah melakukan perakitan komponen-komponen penulis memastikan koneksi antar komponen menggunakan *multi meter*.

4.2 Pengujian dan Analisa *Hardware*

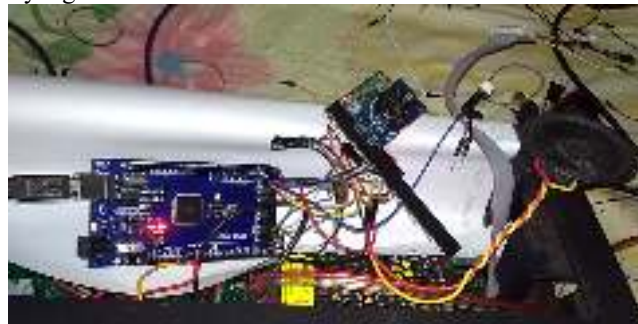
4.2.1 Pengujian Mikrokontroler Arduino Mega 2560

Pengujian mikrokontroler Arduino Mega2560 ini penulis bagi menjadi dua tahap. Tahap pertama yaitu dengan menggunakan multi tester dan lampu led dimana kaki led anoda atau pen multi tester positif disambungkan ke pin-pin yang akan digunakan dan kaki led katoda atau pen multi tester negatif disambungkan ke pin GND, kemudian memberikan perintah secara bergantian 500ms mode High dan 500ms mode Low pada semua pin yang akan digunakan secara bergantian. Selain menguji pin-pin yang akan disambungkan ke modul, pengujian dengan multimeter juga dilakukan untuk memastikan besar tegangan yang dihasilkan oleh pin 5V, 3V dan GND. Tahap kedua pengujian dilakukan langsung dikoneksikan dengan komponen-komponen lain yang digunakan pada alat yang akan dibuat menjadi jam digital jadwal waktu sholat seperti RTC, bluetooth, modul GPS, mini DF-Player dan modul DMD.



Gambar 4.2 Pengujian Arduino Tahap 1

Hasil dari pengujian tahap pertama ini berhasil semua dengan indikator semua led menyala dengan durasi 500 ms atau ½ detik kemudian mati berdurasi 500 ms secara bergantian seperti yang ada pada Gambar 4.2, begitu juga dengan multi tester juga sudah menunjukkan tegangan yang sesuai.



Gambar 4.3 Pengujian Arduino Tahap 2

Pengujian tahap kedua ini berhasil dilakukan dengan indikator semua komponen berjalan dengan baik sesuai dengan program/sketch yang diberikan ke dalam mikrokontroler Arduino Mega2560.

4.2.2 Akurasi GPS

Pengujian akurasi GPS ini penulis bagi menjadi dua bagian, yang pertama menguji sensitivitas GPS dengan cara melihat durasi penerimaan data yang diterima oleh modul GPS pada 5 lokasi yang berbeda dan yang kedua adalah membandingkan data yang didapat oleh modul GPS dengan GPS yang ada pada *smart phone* Android.

1. Uji Sensitivitas

Pengujian tingkat sensitivitas modul GPS yang dipakai dilakukan dengan cara melihat durasi penguncian lokasi, semakin cepat *locking location* atau penerimaan data maka tingkat akurasi GPS *receiver* semakin baik. *Locking location* ini ditandai dengan berkedipnya lampu indikator pada modul GPS.

Tabel 4.2 Uji Sensivitas GPS

No.	Nama Tempat	Durasi <i>Locking Location</i>
1	Masjid Al-Anwar Pasuruan	4626 ms
2	Stadion Gajayana Malang	3231 ms
3	TPQ-Madin Al-Falah Karangrejo Purwosari	1150 ms
4	Masjid Baitul Muttaqin Kertosari	10328 ms
5	Masjid Al-Ma'ruf Sromo Barat	9726 ms

	Pacarkeling Kejayan	
6	Masjid Jami' Al-Mukhlisin Sukorejo	5782 ms
7	Masjid 10 November Sengonagung	3019 ms
8	Masjid Chengho Pandaan	6719 ms

2. Uji Akurasi GPS

Uji akurasi GPS ini dilakukan dengan membandingkan data lintang, bujur dan tinggi tempat pada 6 daerah yang berbeda. Sebagaimana yang ada pada Gambar 4.4 dimana yang sebelah kiri adalah GPS modul menggunakan terminal android dan yang sebelah kanan adalah aplikasi GPS tester yang bisa didapatkan di playstore.



Gambar 4.4 Pengujian Akurasi GPS di Salah Satu Masjid

Dari contoh pengujian seperti pada Gambar 4.4, didapat data-data sebagaimana pada tabel 2.6 di bawah ini.

Tabel 4.3 Pengujian Akurasi GPS

No.	Nama Tempat	GPS Modul	GPS Android
1	Masjid Al-Anwar Pasuruan	La: -7°38'29.32" Lo: 112°54'21.7" Al: 62 m	La: -7°38'30.04" Lo: 112°54'21.6" Al: 22.4 m
2	Stadion Gajayana Malang	La: -7°58'31.1" Lo: 112°37'27.3" Al: 47 m	La: -7°58'30.97" Lo: 112°37'27.8" Al: 21.4 m
3	TPQ-Madin Al-Falah Karangrejo Purwosari	La: -7°44'34.94" Lo: 112°42'12.4" Al: 425 m	La: -7°44'35.04" Lo: 112°42'12.31" Al: 403 m
4	Masjid Baitul Muttaqin Kertosari	La: -7°47'25.7" Lo: 112°44'22.0" Al: 316 m	La: -7°47'25.59" Lo: 112°44'22.5" Al: 296 m
5	Masjid Al-Ma'ruf Sromo Barat Pacarkeling Kejayan	La: -7°42'16.7" Lo: 112°49'45.3" Al: 40 m	La: -7°42'16.77" Lo: 112°44'45.1" Al: 49 m
6	Masjid Jami' Al-Mukhlisin Sukorejo	La: -7°43'9.63" Lo: 112°43'8.52" Al: 257 m	La: -7°43'10.06" Lo: 112°43'8.8" Al: 241 m
7	Masjid 10 November Sengonagung	La: -7°39'39.1" Lo: 112°43'43" Al: 268 m	La: -7°44'39.44" Lo: 112°43'43.2" Al: 275 m
8	Masjid Chengho	La: -7°39'5.67" Lo: 112°43'43" Al: 268 m	La: -7°39'6.1" Lo: 112°43'43.2" Al: 275 m

Pandaan	Lo: 112°41'14.6" Al: 194 m	Lo: 112°41'15" Al: 205 m
---------	-------------------------------	-----------------------------

Hasil dari pengujian modul GPS baik dari segi sensitivitas maupun akurasi yang telah dilakukan didapatkan perbedaan data yang tidak terlalu signifikan untuk perhitungan waktu sholat.

4.2.3 Sensifitas Bluetooth

Pengujian sensifitas bluetooth ini dilakukan dengan cara mengukur jarak remote/HP dengan modul dan durasi penerimaan modul bluetooth yang ditandai dengan perubahan pada alat jam digital dan jadwal waktu sholat sesuai dengan perintah yang diberikan.

Tabel 4.4 Pengujian Sensifitas Bluetooth

No.	Jarak	Durasi Penerimaan Data
1	1 meter	Kurang dari 1 detik
2	2 meter	Kurang dari 1 detik
3	3 meter	Kurang dari 1 detik
4	4 meter	Kurang dari 1 detik
5	5 meter	Kurang dari 1 detik
6	6 meter	Kurang dari 1 detik

4.2.4 Akurasi Jam / Waktu pada RTC

Pengujian akurasi jam atau waktu pada RTC ini pada dasarnya juga menguji akurasi jam yang didapat dari GPS receiver, penulis melakukan pengujian dengan cara membandingkan dengan hasil perhitungan waktu menggunakan bayangan benda yang terkena sinar matahari sebagaimana rumus yang telah dijelaskan pada bab 2. Pengujian yang penulis lakukan menggunakan bayangan kertas folio dengan tinggi 33 cm.

Langkah pertama adalah menghitung berapa tinggi matahari jika pengukuran dilakukan dengan bayangan benda 50 cm.
 $h = \tan^{-1}(33/33) = 45$

Pengukuran dilakukan di halaman musholla Al-Falah Karangrejo Purwosari Pasuruan dengan koordinat :

Lintang tempat (LT) : -7°44'35.49"
 Bujur tempat (BT) : 112°42'11.68"
 Zona Waktu (TZ) : 7

Menggunakan data matahari pada hari Ahad 1 Juli 2018, yang didapat dari perhitungan :

$$\begin{aligned}
 D &= 1 \\
 M &= 7 \\
 Y &= 2018 \\
 B &= -(\text{Int}(Y / 100)) + \text{Int}(\text{Int}(Y / 100) / 4) + 2 = -13 \\
 H &= 12 \\
 JD &= 1720994.5 + \text{Int}(365.25 * Y) + \text{Int}(30.6001 * (M + 1)) + B + D + (H / 24) - TZ / 24 \\
 &= 2458300.70833333 \\
 T &= (JD - 2451545) / 36525 = 0.184961214 \\
 S &= \text{Mod}((280.46645 + 36000.76983 * T), 360) \\
 &= 99.21253479 \\
 m &= \text{Mod}((357.5291 + 35999.0503 * T), 360) \\
 &= 175.9571384 \\
 N &= \text{Mod}((125.04 - 1934.136 * T), 360) \\
 &= 127.2998578 \\
 K' &= (17.264 / 3600) * \text{Sin}(N) + (0.206 / 3600) * \text{Sin}(2 * N) \\
 &= 0.003759577 \\
 K'' &= (-1.264 / 3600) * \text{Sin}(2 * S) = 0.000110974 \\
 R' &= (9.23 / 3600) * \text{Cos}(N) - (0.09 / 3600) * \text{Cos}(2 * N) \\
 &= -0.001547043 \\
 R'' &= (0.548 / 3600) * \text{Cos}(2 * S) = -0.000144419 \\
 Q' &= 23.43929111 + R' + R'' - (46.815 / 3600) * T \\
 &= 23.43519438 \\
 E &= (6898.06 / 3600) * \text{Sin}(m) + (72.095 / 3600) * \text{Sin}(2 * m) + (0.966 / 3600) * \text{Sin}(3 * m) \\
 &= 0.132331772 \\
 S' &= \text{Mod}(S + E + K' + K'' - (20.47 / 3600), 360) \\
 &= 99.343051 \\
 \delta &= \text{Sin}^{-1}(\text{Sin } S' * \text{Sin } Q') = 23.1061295
 \end{aligned}$$

$$e = (-1.915 \times \sin(m) - 0.02 \times \sin(2 \times m) + 2.466 \times \sin(2 \times S') - 0.053 \times \sin(4 \times S')) / 15$$

$$= -0.063579426$$

Deklinasi matahari (δ) : $23^{\circ}6'22.07''$

Equation of Time (e) : $-0^{\circ}3'48.89''$

Istiwa' Sore = $\cos^{-1}(-\tan LT \times \tan \delta + \sin h / \cos LT / \cos \delta) / 15$
 = $\cos^{-1}(-\tan -7^{\circ}44'35.49'' \times \tan 23^{\circ}6'22.07'' + \sin 45 / \cos -7^{\circ}44'35.49'' / \cos 23^{\circ}6'22.07'') / 15$
 = $2^{\circ}14'0.46''$

Istiwa' Pagi = $24 - \text{Istiwa' Sore}$
 = $24 - 2^{\circ}14'0.46'' = 21^{\circ}45'59.54''$

Kemudian mengkonversi istiwa' ke WIB :

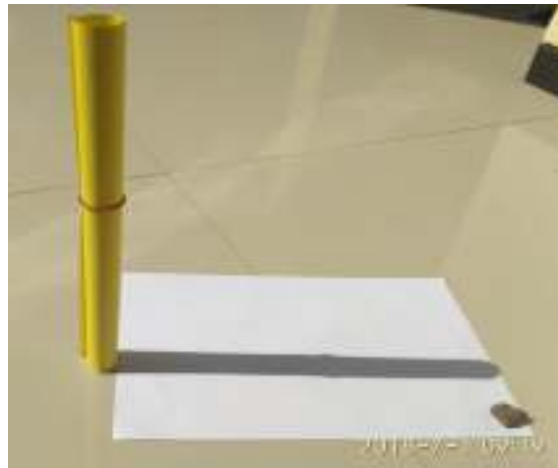
KWD = $\frac{TZ \times 15 - \lambda}{15} - e$
 = $\frac{7 \times 15 - 112^{\circ}42'11.68''}{15} - -0^{\circ}3'48.89''$
 = $-0^{\circ}34'37.66''$

WIB = Wis + KWD

WIB Pagi = $21^{\circ}45'59.54'' + -0^{\circ}34'37.66'' - 12 = 9^{\circ}18'57.76''$

WIB Sore = $2^{\circ}14'0.46'' + -0^{\circ}34'37.66'' + 12 = 13^{\circ}46'58.68''$

Jadi pada hari Ahad 01 Juli 2018 benda setinggi 33 cm dengan bayangan yang sama 33 cm menunjukkan pukul $9^{\circ}18'57.76''$ WIB untuk pengukuran pagi hari atau $13^{\circ}46'58.68''$ WIB jika pengukuran dilakukan sore hari sebagaimana pada Gambar 4.5 dan Gambar 4.6.



Gambar 4.5 Pengujian Akurasi Jam pada Pagi Hari



Gambar 4.6 Pengujian Akurasi Jam pada Sore Hari

4.2.5 Akurasi Alarm

Akurasi alarm ini dihasilkan berdasarkan *speed clock* yang ada pada mikrokontroler arduino. *Speed clock* tersebut menghasilkan *micro second* atau *milli second*, hal ini yang membuat sebuah alarm tidak bisa berbunyi karena kondisi yang dimasukkan pada saat pengkodean tidak terpenuhi. Untuk mengatasi hal ini penulis tidak menggunakan *microsecond* namun

menggunakan *second* dan pengujiannya adalah pemutaran file mp3 setiap menjelang masuknya waktu sholat.

Tabel 4.5 Uji Akurasi Alarm

Alarm	Durasi	Bunyi
Tarhim Duhur	6 menit	Bunyi
Tarhim Ashar	6 menit	Bunyi
Tarhim Maghrib	6 menit	Bunyi
Tarhim Isya'	6 menit	Bunyi
Tarhim Shubuh	6 menit	Bunyi

4.2.6 Akurasi Hisab

Arduino Mega2560 merupakan mikrokontroler yang menggunakan sistem 8 bit dimana digit yang dihasilkan tentunya berbeda dengan sistem *processor* 32 bit maupun 64 bit, hal ini menyebabkan perhitungan waktu sholat yang dihasilkan juga berbeda, oleh karena itu pengujian akurasi hisab ini perlu dilakukan dengan cara membandingkan hasil perhitungan menggunakan Arduino (8 bit) dan laptop (64 bit).

Pada Tabel 4.5 penulis menggunakan perhitungan data-data matahari yang diperlukan dengan variabel dasar yang diambil saat maghrib tanggal 2 Juli 2018 dengan koordinat bujur 112.7032444, lintang -7.743191667, Zona Waktu 7 dan ketinggian 404.1648 meter dari permukaan laut.

Tabel 4.6 Perbandingan Hasil Hisab menggunakan Sistem 8 Bit dan 64 Bit

Variabel	Sistem 64 Bit	Sistem 8 Bit
JD	2458301.5	2458301.5
T	0.184982888	0.18498288
S	99.99283895	99.99218750
M	176.7374053	176.73730468
N	127.2579361	127.2579651
K'	0.003761724	0.00376172
K'	0.000120005	0.00012000
R'	-0.001545515	-0.00154552
R'	-0.000143055	-0.00014306
Q'	23.43519699	23.43519783
E	0.106820963	0.10682423
S'	100.0978555	100.09721374
δ	23.05103793	23.05108451
PT	100.9845071	100.98381805
e	-0.066111213	-0.06610870
SDM	0.262543986	0.26254396
Dip	0.589712944	0.58971290
h	1.429656931	1.42965688
Istiwa'	5.885688345	5.88349533
WIB	17.43824993 17°26'17.7"	17.43605422 17°26'9.8"

Dari perbandingan perhitungan waktu maghrib tanggal 2 Juli 2018 di Musholla Al-Falah Karangrejo pada tabel 4.5 di atas terdapat perbedaan yang tidak terlalu signifikan yaitu sebesar 7,9 detik.

4.3 Pemrograman Arduino

Setelah melakukan perakitan dan pengujian seluruh komponen, selanjutnya adalah pemberian program pada Arduino menggunakan platform Arduino IDE. Dalam pemrograman ini penulis menggunakan library yang sudah disediakan Arduino yang bisa didownload di www.github.com antara lain:

1. DMD.h
2. EEPROM.h
3. DFPlayer_Mini_Mp3.h
4. TimerOne.h
5. RTCLib.h
6. SPI.h
7. TinyGPS.h
8. math.h.

Selain ke delapan library di atas untuk menampilkan tulisan-tulisan ke display membutuhkan library font yang support

dengan modul DMD, library font ini juga bisa dikembangkan sendiri melalui aplikasi GLCD Font Creator. Untuk *full source code* bisa dilihat pada lampiran 2.

4.4 Pengujian Aplikasi Remote pada Smart Phone

Pengujian ini dilakukan pada aplikasi remote yang telah dibangun menggunakan App Inventor yang berupa perintah perubahan *display text*, setting iqomah, tartil otomatis, pengaturan kecerahan, penyesuaian jadwal waktu sholat dan pengaturan lain (pengiriman data lokasi menggunakan GPS Android). Tombol kirim yang ada pada aplikasi remote ini berfungsi untuk mengirimkan sebuah teks via Bluetooth yang didahului dengan kode 3 karakter yang diikuti dengan teks masing-masing pilihan, kode-kode tersebut adalah:

1. Kode icon clock : JM
2. Kode menu Edit Teks : TXT
3. Kode menu Iqomah : IQM
4. Kode menu Tartil Otomatis : MRT
5. Kode menu Kecerahan : CRH
6. Kode menu Penyesuaian Jadwal : IHT
7. Kode menu Pengaturan Lain : GPS



Gambar 4.7 Tampilan Aplikasi Remote Control via Android

Pada masing-masing menu dan tombol yang terdapat pada aplikasi seperti Gambar 4.5 diuji apakah sudah berjalan dengan baik yang ditandai dengan perubahan pada jam digital waktu sholat yang telah dirakit dan diberi perintah melalui program Arduino.

Tabel 4.7 Pengujian Aplikasi Remote Control Android

No.	Menu	Tombol Kirim	Tombol Batal
1	Edit Teks	Berfungsi	Berfungsi
2	Iqomah	Berfungsi	Berfungsi
3	Tartil Otomatis	Berfungsi	Berfungsi
4	Kecerahan	Berfungsi	Berfungsi
5	Penyesuaian Jadwal	Berfungsi	Berfungsi
6	Pengaturan Lain	Berfungsi	Berfungsi

5. PENUTUP

5.1 Kesimpulan

Setelah menyelesaikan tahap perakitan dan pengujian dan pemrograman/pengkodean jam digital jadwal waktu sholat ini penulis menarik kesimpulan sebagai berikut :

1. Penggunaan modul GPS pada sebuah jam waktu sholat lebih efektif daripada menggunakan GPS Android yang

disinkronkan dan dikirim melalui aplikasi Android.

2. Penggunaan modul RTC (*Real Time Clock*) tanpa sinkronasi dengan GPS lebih rentan akan berubahnya waktu yang tersimpan ketika daya baterai yang dibutuhkan melemah.
3. Modul bluetooth dapat membantu dalam proses pengaturan teks informasi pada display, pengaturan lokasi maupun pengaturan jam ketika terjadi error pada modul GPS dengan bantuan aplikasi remote pada android.
4. Hasil perhitungan menggunakan mikrokontroler dengan sistem 8 bit tidak jauh berbeda dengan perhitungan menggunakan laptop dengan sistem 64 bit, perbedaannya hanya 7,9 detik.
5. Data waktu yang didapatkan dari satelit GPS meliputi : tahun, bulan, tanggal, jam, menit dan detik sudah sesuai dengan waktu yang didapatkan menggunakan bayangan benda yang terkena sinar matahari.
6. Dari hasil pengujian secara global, jam digital dan waktu sholat ini berjalan sesuai dengan hasil rancangan yang telah dilakukan.

5.2 Saran

Setelah melakukan analisa dari project jam digital waktu sholat ini, saran penulis adalah sebagai berikut :

1. Bagi peneliti selanjutnya hendaknya tidak menggunakan *breadboard* sebagai penghubung antar komponen seperti yang penulis lakukan melainkan menggunakan papan PCB dengan desain minimalis.
2. Bagi para pengembang jam digital berbasis mikrokontroler untuk keperluan apapun atau menggunakan display apapun agar menggunakan chip GPS untukantisipasi jika baterai pada RTC sudah melemah.
3. Untuk pengembangan selanjutnya hendaknya menambahkan kondisi alarm atau murottal otomatis ketika hari jum'at.
4. Bagi para pembaca pada umumnya yang akan melaksanakan sholat agar tidak hanya melihat dan menggunakan jam digital yang ada di masjid-masjid sebagai patokan, namun juga memperhatikan penunjuk waktu lain untuk memastikan masuknya waktu sholat.

DAFTAR PUSTAKA

- [1] Darmawan, Sudjadi Darjat. *Rancang Bangun Jam Digital Waktu Shalat Berbasis Mikrokontroler AT89S52*. UNDIP Tembalang, Year 2008.
- [2] Siregar, Rajasali Muhammad, Abdul Jabar, and Ari Usman. *Perancangan Perangkat Keras Sistem Pengingat Waktu Sholat Elektronik*. Sekolah Tinggi Teknik Harapan Medan, , p.1. Year 2016..
- [3] Safryanti, Ery, Feranita, and Romi Ardiles. *Rancang Bangun Indikator Jam Sholat Abadi Menggunakan Atmel 89S52*. Seminar Nasional Teknik Kimia Oleo & Petrokimia Indonesia, , p.1. Year 2008.
- [4] Peter A Crossley, Hao Guo, Zhao Ma. *Time Synchronization for Transmission Substations Using GPS and IEEE 1588*. CSEE JOURNAL OF POWER AND ENERGY SYSTEMS, Vol.2, , p.91. Year 2016.
- [5] Yu-Ling Hsueh, Ho-Chian Chen. *Map Matching for Low-Sampling-Rate GPS Trajectories by Exploring Real-time Moving Directions*. Journal of Information Sciences, Vol. II , pp.1-27. Year 2017.
- [6] Isa, Azmi Awang Md., Zahriladha Zakaria, Fauzi Mohd. Johar, and Ahmad Faisal Othman. *Mobile Prayer Times and Qiblat Direction Using GPS*. Apace, , pp.1-5. Year 2007.
- [7] Zakiyudin, Ais. *Sistem Informasi Manajemen*. Bogor: Mitrawacanamedia, 2012.
- [8] Fathulloh, Ahmad Ghozali Muhammad. *Irsyadul Murid*. Sampang: Lafal, 2015.
- [9] Efendi, Ilham. (2014). *Pengertian dan Kelebihan Arduino*. Retrieved 4 2, 2018, from IT-Jurnal.com: <https://www.it-jurnal.com/pengertian-dan-kelebihan-arduino/>
- [10] Schille, J., & Voisar, A. (2004). *Location-Based Services*. San Fracisco: Morgan gKaufmann.
- [11] Mujiono. (2015, 10). *Pengertian dan Fungsi Bluetooth*. Retrieved 3 28, 2018, from Teori Komputer: <http://www.teorikomputer.com/2015/10/pengertian-dan-fungsi-bluetooth.html>
- [12] Yusuf, R. M. (2017, November 10). *Realtime Clock*. Retrieved March 29, 2018, from Scribd, Read Anywhere Anytime: <https://www.scribd.com/doc/22362718/Real-Time-Clock>
- [13] Yusuf, R. M. (2017, November 10). *Realtime Clock*. Retrieved March 29, 2018, from Scribd, Read Anywhere Anytime: <https://www.scribd.com/doc/22362718/Real-Time-Clock>