



Akurasi Sistem Penjadwalan Sholat Digital Menggunakan Arduino Sebagai Pengendali

Emil Naf'an^{1✉}

¹Fakultas Ilmu Komputer, Universitas Putra Indonesia YPTK Padang

emilnafan@upiyptk.ac.id

Abstract

This study aims to test the accuracy of the Digital Prayer scheduling system using the Arduino as a controller. The place of research is the scheduling of Digital Prayers at the Putra Putra University Mosque (UPI) of the Padang College of Computer Foundation (YPTK). Prayer schedule data is taken from the website of the Indonesian Ministry of Religion and scheduling routine lecturer activities every morning. Data is stored on a MicroSD card with a different data file name. Data is processed using the Arduino module. The data tested is the display of prayer schedules on Seven Segments and lecturer activity schedule displays on Running Text. The prayer times automatically work with the date and time display (hours, minutes, seconds) starting from the Fajr prayer to the Isha prayer at Seven Segments. Display the activity of lecturer routines automatically by displaying the names of lecturers with the specifications of their assignments, such as reading the Qur'an, reciting Asmaul Husna and praying. Testing is done by activating the equipment for a week (7 days) non stops. The system is tested in terms of the durability of components in operation, the timeliness in activating Buzzer when prayer time has entered and scheduling lecturer activities very well, precisely and accurate. Thus the system can be used and produced on a larger scale.

Keywords: Prayer Times, Micro SD, Arduino, Seven Segment, Running Text.

Abstrak

Penelitian ini bertujuan untuk menguji akurasi sistem penjadwalan Sholat Digital menggunakan Arduino sebagai pengendali. Tempat penelitian adalah penjadwalan Sholat Digital pada Mesjid Universitas Putra Indonesia (UPI) Yayasan Perguruan Tinggi Komputer (YPTK) Padang. Data jadwal sholat diambil dari website Kementerian Agama RI dan penjadwalan rutinitas kegiatan dosen setiap pagi. Data disimpan dalam kartu MicroSD dengan nama file data yang berbeda. Data diolah menggunakan modul Arduino. Data yang diuji adalah tampilan jadwal sholat pada *Seven Segment* dan tampilan jadwal kegiatan dosen pada *Running Text*. Waktu sholat secara otomatis bekerja dengan tampilan tanggal dan waktu (jam, menit, detik) yang dimulai dari sholat Shubuh sampai sholat Isya pada *Seven Segment*. Tampilan kegiatan rutinitas dosen juga secara otomatis dengan menampilkan nama-nama dosen dengan spesifikasi tugasnya, seperti membaca Al-Qur'an, berdzikir Asmaul Husna dan berdo'a. Pengujian dilakukan dengan mengaktifkan peralatan selama seminggu (7 hari) tidak berhenti. Sistem diuji dari segi ketahanan komponen dalam beroperasi, ketepatan waktu dalam mengaktifkan *Buzzer* saat waktu sholat telah masuk dan penjadwalan kegiatan dosen dengan sangat baik, tepat dan akurat. Dengan demikian sistem dapat dipergunakan dan diproduksi dalam skala yang lebih besar.

Kata kunci: Jadwal Sholat, *micro SD*, Arduino, *Seven Segment*, *Running Text*.

© 2019 JSisfotek

1. Pendahuluan

Universitas Putra Indonesia (UPI) Yayasan Perguruan Tinggi Komputer (YPTK) Padang merupakan salah satu Perguruan Tinggi swasta terkemuka di Wilayah LLDIKTI Wilayah X (Sumatera Barat, Riau, Jambi dan Kepulauan Riau). Pada awal berdirinya pada tahun 1985, masih berbentuk Sekolah Tinggi yang terdiri dari STMIK, AMIK, STIE dan AAMPK [1]. Universitas ini sangat menekankan pendidikan karakter bagi civitas akademika. Setiap akan dimulai perkuliahan, semua mahasiswa berdzikir Asmaul Husna, mengikrarkan 12 Prinsip Dasar UPI YPTK. Begitu juga halnya dengan dosen, sebelum dimulai aktivitas kerja semua dosen berkumpul di Masjid Rahmatan Lil 'Alamin UPI YPTK

Padang. Di masjid tersebut semua dosen membaca Al-Qur'an, kemudian berdzikir Asmaul Husna, berdo'a dan mengikrarkan 12 Prinsip Dasar UPI YPTK Padang. Kegiatan ini dilakukan setiap hari kerja, dipimpin oleh salah seorang dosen dan karyawan yang bergantian sesuai dengan jadwal yang telah ditetapkan. Jadwal ini masih dibuat secara manual menggunakan selembar kertas yang ditempelkan di dinding depan Masjid. Hal ini memiliki beberapa kelemahan antara lain ; sulit melihat jadwal dosen dan karyawan yang bertugas pada saat kegiatan sudah dimulai. Terkadang ada dosen dan karyawan yang tidak ingat ataupun ragu terhadap jadwal tugasnya, sehingga harus dipanggil dulu untuk memimpin kegiatan tersebut. Tentu saja ini membuat kegiatan tersebut menjadi sedikit terganggu.

Berdasarkan permasalahan tersebut timbul inisiatif dari pimpinan (rektor dan wakil rektor) untuk membuat *Running Text* yang secara otomatis menampilkan jadwal dosen dan karyawan yang bertugas pada kegiatan di Masjid setiap hari kerja. Kegiatan tersebut antara lain, membaca Al-Qur'an, berdzikir Asmaul Husna dan berdo'a. Pimpinan meminta penulis untuk merancang dan membuat *Running Text* tersebut. Penulis menyanggapi dan menyarankan *Running Text* tersebut dibuat sebagai sarana pelengkap pada Jadwal Sholat Digital [2]. Hal ini disebabkan jadwal sholat di masjid Rahmatan Lil 'Alamin UPI YPTK Padang pada saat itu masih dibuat secara manual menggunakan kertas yang ditempelkan di dinding bagian depan Masjid. Alhamdulillah pihak Yayasan dan pimpinan menyetujui sekaligus mendanai pembuatan alat tersebut.

Jadwal sholat Digital sudah banyak diteliti dan diterapkan oleh berbagai penulis, antara lain ; Husain dan Ahyuna dari STMIK Dipanegara Makassar [3]. Pada sistem ini pengingat waktu sholat (adzan) dikirim melalui SMS (*Short Message Service*) dari server ke masjid-masjid yang telah dipasang alat penerima SMS tersebut (*client*). Berdasarkan waktu tersebut maka *Buzzer* pada masjid akan berbunyi. Dari hasil penelitian terdapat perbedaan waktu (jeda) pengiriman SMS sebesar maksimum 2 detik, untuk jarak 0 – 10 Km. Sistem ini sangat tergantung kepada jaringan telekomunikasi. Jika jaringan bermasalah, maka waktu sholat pada masing-masing masjid tidak akan bekerja. Akibatnya bisa saja jadwal sholat di masjid-masjid tersebut terganggu.

Penelitian yang lain, dilakukan oleh Risal dkk [4]. Pada sistem ini digunakan komputer sebagai server, Arduino [5][6] dengan *Ethernet Shield* [7] terkoneksi dengan internet untuk mengontrol perangkat elektronik masjid. Sistem ini juga memiliki kelemahan yang hampir sama dengan penelitian sebelumnya yaitu terdapat jeda waktu pengiriman dan terkadang pengiriman bisa saja tidak berhasil dilakukan. Perbedaan dengan penelitian sebelumnya adalah pengiriman informasi jadwal sholat dikirimkan melalui jaringan internet.

Dalam penelitian ini digunakan MicroSD [8] sebagai media penyimpanan data jadwal sholat dan data jadwal kegiatan dosen setiap harinya. Hal ini disebabkan MicroSD merupakan media penyimpanan yang banyak digunakan, baik pada smartphone, data logger dan lain-lain. Disamping itu ukuran fisiknya yang relatif kecil dan kapasitas penyimpanan MicroSD tersebut cocok untuk menyimpan data jadwal sholat [9] dan data jadwal kegiatan dosen setiap harinya selama lebih dari 10 tahun. Jadwal sholat ditampilkan menggunakan Seven Segment [10] – [13], sedangkan data kegiatan dosen ditampilkan pada *Running Text*. Semua peralatan dikendalikan oleh modul Arduino berdasarkan program yang tersimpan di dalam Memory Program Arduino tersebut.

2. Metodologi Penelitian

Penelitian ini bertujuan untuk merancang dan membuat suatu sistem berupa alat Jadwal Sholat Digital menggunakan Arduino sebagai pengendali. Metode yang dilakukan dalam penelitian ini yaitu : Perancangan Sistem, Implementasi Sistem dan Pengujian Sistem. Dari hasil pembahasan terhadap pengujian yang dilakukan, diperoleh kesimpulan dari sistem yang dirancang.

1. Perancangan Sistem

Dalam melakukan perancangan sistem terdapat dua hal yang perlu dilakukan yaitu perancangan baik dari segi perangkat keras (*hardware*) maupun perangkat lunak (*software*). Perancangan tersebut didasari atas ketentuan antara lain :

- a. Sistem harus mampu menampilkan waktu (jam, menit, detik) dan waktu tunda untuk jeda iqomah pada *Seven Segment* waktu berdasarkan data waktu pada RTC DS1307.
- b. Sistem juga harus mampu menampilkan tanggal, bulan, tahun dan nama hari pada *Seven Segment* tanggal/jeda iqomah secara bergantian.
- c. Pada saat waktu sholat masuk, sistem harus mampu mengaktifkan *Buzzer* sebagai tanda bagi jamaah bahwa waktu sholat sudah masuk.
- d. Sistem harus mampu menampilkan waktu jeda iqomah yang menghitung mundur, sehingga jamaah bisa memperkirakan waktu untuk sholat sunat sebelum sholat wajib (fardhu) dilaksanakan.
- e. Seiring dengan waktu jeda iqomah tampil di *Seven Segment*, *Running Text* menampilkan pesan bahwa waktu sholat segera dimulai, kemudian pesan untuk meluruskan dan merapatkan shaf saat sholat berjamaah dilakukan.
- f. Pada saat waktu jeda iqomah sudah mencapai nilai 0, maka *Buzzer* aktif selama 500ms, kemudian semua *Seven Segment* dan *Running Text* padam yang tampil hanya *Seven Segment* untuk menampilkan kata 'Shalat' ditambah dengan nama waktu sholat saat itu.
- g. Sistem harus mampu menampilkan kembali waktu, tanggal dan nama hari serta data waktu sholat secara otomatis setelah 12 menit. Nilai ini diambil dari rata-rata lama waktu sholat berjamaah.
- h. Sistem juga harus mampu menampilkan kembali pesan teks pada *Running Text* setelah selesai waktu sholat.
- i. Pada jam 06.53 WIB, *Running Text* secara otomatis menampilkan nama-nama dosen yang bertugas dalam membaca Al-Qur'an, berdzikir Asmaul Husna dan berdo'a.
- j. Sistem harus bisa mengaktifkan *Buzzer* selama 500ms di setiap hari kerja sebagai tanda

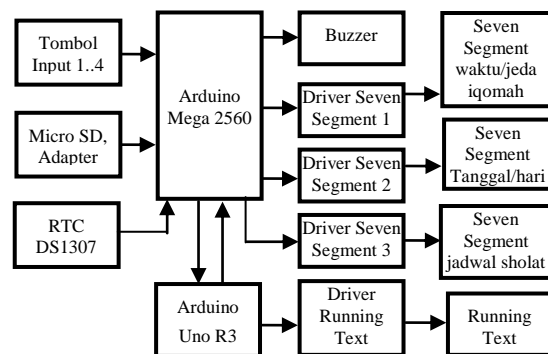
dimulainya kegiatan dosen di masjid. Adapun jadwal pengaktifan *Buzzer* sebagai berikut :

Hari Senin s/d Kamis dan Sabtu : 07.15 WIB.

Hari Jum'at : 07.00 WIB.

- k. *Running Text* kembali menampilkan pesan berupa terjemahan ayat Al-Qur'an ataupun Hadits setelah pukul 7.30 WIB, karena setelah waktu ini biasanya kegiatan dosen hampir selesai dilakukan.
- l. Pada saat pergantian hari pada pukul 00:00 WIB, sistem secara otomatis mengambil data jadwal sholat yang tersimpan pada Micro SD dan menampilkan pada *Seven Segment* jadwal sholat.

Adapun blok *diagram* dari sistem yang dirancang dapat dilihat pada gambar 1.



Gambar 1. Blok Diagram Jadwal Sholat Digital

Pada blok diagram tersebut terlihat bahwa sistem yang dirancang memiliki 2 (dua) buah modul Arduino, yaitu modul Arduino Mega2560 dan Arduino Uno R3. Dalam hal ini kendali sistem dipegang oleh Arduino Mega2560, sedangkan Arduino Uno R3 berfungsi untuk menerima data dari Arduino Mega2560 berupa pesan teks yang akan ditampilkan pada *Running Text*. Arduino Uno R3 mengolah data tersebut berdasarkan program yang tersimpan di dalamnya, kemudian mengirimkan ke *Driver Running Text*. Pesan teks tersebut ditampilkan pada *Running Text* dengan ukuran dot matrix 8 x 96. Tombol input berfungsi sebagai input dalam proses *setting* jadwal sholat. Tombol ini terdiri dari 4 buah tombol, yaitu tombol *setting*, tombol *pindah_posisi*, tombol *tambah_nilai* dan tombol *simpan*. MicroSD berfungsi menyimpan data jadwal sholat dan data jadwal kegiatan dosen. Data waktu sholat di-*update* setiap pergantian hari, pada pukul 00:00:01 dan ditampilkan pada *Seven Segment* jadwal sholat. Untuk menjaga agar data waktu dan tanggal tetap tepat walaupun sumber catudaya dimatikan, digunakan modul *realtime clock* DS1307 yang mampu beroperasi sampai tahun 2099. Data waktu ini diolah oleh Arduino Mega2560 kemudian ditampilkan pada *Seven Segment* waktu, berupa jam, menit dan detik. Sebagai tanda bahwa waktu

sholat masuk dalam hal ini digunakan *Buzzer*. Begitu juga sebagai tanda bahwa kegiatan dosen dimulai digunakan *Buzzer* yang berbunyi selama setengah detik (500ms) sebanyak 3 kali berturut-turut. Tegangan sumber catudaya peralatan berasal dari jala-jala PLN 220VAC. Keseluruhan sistem dikendalikan oleh Arduino Mega2560, dengan dibantu oleh Arduino Uno R3 berdasarkan data yang tersimpan dalam *memory* program modul tersebut.

2. Implementasi Sistem

Pada tahap ini dilakukan implementasi dari sistem yang telah dirancang. Implementasi sistem dilakukan dengan membangun perangkat (alat) yang telah dirancang sebelumnya baik dari segi perangkat keras (*hardware*) maupun perangkat lunak (*software*). Dari segi perangkat keras (*hardware*), perangkat (alat) dibangun mulai dari pembuatan rangkaian elektroniknya sampai ke penempelan stiker *cover* masjid dan pembuatan rangka aluminium yang terletak di belakang bingkai tersebut. Rangka aluminium ini berguna untuk memperkuat bingkai sekaligus sebagai peredam panas (*heat sink*) IC *Regulator* rangkaian catudaya, seperti terlihat pada gambar 2. Adapun bingkai peralatan dipakai yang sudah jadi yang dibeli dari toko pembuat bingkai foto dengan ukuran 100cm x 200cm.

IC *Regulator* yang ditempelkan pada bingkai aluminium sebagai peredam panas (*heat sink*)



Gambar 2. Bingkai Aluminium Sebagai Peredam Panas (*Heat Sink*)

Sedangkan dari perangkat lunak (*software*), implementasi sistem dibangun untuk mengendalikan peralatan yang dirancang. Adapun bahasa yang digunakan adalah bahasa C untuk pemrograman Arduino [14][15]. Setelah program selesai dibuat, program di-*upload* ke dalam modul Arduino. Selanjutnya peralatan diaktifkan dan dilakukan pengujian baik dari segi perangkat keras (*hardware*) maupun dari segi perangkat lunak (*software*).

3. Hasil dan Pembahasan

Setelah peralatan dibuat, selanjutnya peralatan diuji untuk mengetahui sejauh mana kinerja dari peralatan yang dirancang. Adapun gambar keseluruhan Jadwal Sholat Digital tersebut dapat dilihat pada gambar 3 di bawah ini.



Gambar 3. Tampilan Fisik Jadwal Sholat Digital

Pengujian Rangkaian Tombol Input 1..4

Pengujian dilakukan dengan mengukur tegangan saat tombol ditekan maupun saat tombol tidak ditekan. Hasil pengujian dapat dilihat pada tabel 1.

No.	Kondisi Tombol	Tegangan Output (VDC)
1	Ditekan	0,00 V
2	Tidak Ditekan	4,98 V

Dari hasil pengujian dapat dilihat bahwa pada saat tombol ditekan, tegangan output yang dihasilkan adalah 0,00VDC. Tegangan ini pada sistem digital dikategorikan sebagai logika '0' (*Low*). Hal ini disebabkan output rangkaian tombol langsung terhubung ke *ground*, sedangkan pada saat tombol tidak ditekan, tegangan output rangkaian sebesar 4,98VDC. Hal ini disebabkan pada saat tombol tidak ditekan, output rangkaian langsung terhubung ke sumber tegangan melewati resistor 4,7Kohm. Tegangan ini dalam sistem digital dikategorikan sebagai logika '1' (*High*). Berdasarkan hasil pengujian tersebut, rangkaian tombol input 1..4 dapat dikatakan bekerja dengan baik, karena memenuhi syarat untuk dibaca secara digital oleh modul Arduino.

Pengujian MicroSD

Pengujian dilakukan dengan melihat langsung pada tampilan *Seven Segment* dan *Running Text*. Setelah dilakukan pengujian diperoleh hasil bahwa tampilan pada *Seven Segment* dan *Running Text* sama dengan data yang ada pada MicroSD.

Pengujian Realtime Clock DS1307

Pengujian *realtime clock* DS1307 dilakukan dengan melihat langsung pada *Seven Segment* waktu (jam, menit, detik) dan membandingkan dengan nilai detik yang ada pada sistem jam digital laptop (PC). Dari hasil pengujian diperoleh perbedaan waktu antara RTC DS1307 dengan waktu pada laptop untuk setiap harinya sebanyak ± 3 detik lebih cepat. Solusinya adalah memperlambat nilai pada RTC DS1307 tersebut setiap hari menggunakan program yang disimpan pada

Arduino Mega2560. Algoritma kalibrasi tersebut dapat dilihat pada Gambar 4.

Algoritma Kalibrasi

```
//Program kalibrasi waktu saat jam 00:05 WIB
DateTime now = RTC.now();
int vtahun,vbulan,vtgl,vjam,vmenit;
if ((now.hour()==0) && (now.minute()==0) &&
(now.second() == 5))
{
vtahun = now.year();
vbulan = now.month();
vtgl = now.day();
vjam = now.hour();
vmenit = now.minute();
delay(3000); //tunda 3 detik untuk
// kalibrasi
RTC.adjust(DateTime(vtahun,vbulan,
vtgl, vjam, vmenit, 6));
}
```

Gambar 4. Algoritma Kalibrasi Realtime DS1307

Pengujian Seven Segment Waktu (jam, menit, detik)

Pengujian dilakukan dengan melihat langsung kemampuan *Seven Segment* waktu (jam, menit, detik) dalam menampilkan angka 0 sampai dengan 9. Setelah dilakukan pengujian ternyata *Seven Segment* mampu menampilkan angka 0 sampai dengan 9 dengan baik.



Gambar 5. Pengujian Tampilan *Seven Segment* Waktu (jam, menit, detik)

Sebagian dari pengujian dapat dilihat pada gambar 5. Dengan demikian dapat dikatakan bahwa rangkaian *Seven Segment* waktu (jam, menit, detik) dapat bekerja dengan baik.

Pengujian Seven Segment Tanggal / Hari

Pengujian dilakukan dengan melihat langsung pada *Seven Segment* tanggal / hari dan membandingkannya dengan data tanggal pada laptop (PC). Hasil pengujian dapat dilihat pada gambar 6.



(a)



(b)



(c)

Gambar 6. Pengujian Tampilan *Seven Segment* Tanggal / Hari

- (a) Tampilan Hari
- (b) Tampilan Tanggal

(c) Tampilan *Setting* waktu Pada Laptop (PC)

Pada gambar 6 terlihat bahwa antara hari dan tanggal tampil secara bergantian dengan *interval* waktu sebesar 3 detik. *Interval* ini dipilih berdasarkan kemampuan rata-rata orang dalam membaca tanggal dan hari, paling lama ± 3 detik. Setelah nilai tanggal dan hari dibandingkan dengan tampilan *setting* waktu pada laptop (PC), diperoleh hasil yang akurat, yaitu hari Sabtu adalah merupakan tanggal 07 September 2019. Dari hasil yang diperoleh dapat dikatakan bahwa rangkaian *Seven Segment* waktu (jam, menit, detik) dapat bekerja dengan baik.

Pengujian *Seven Segment* Jadwal Sholat

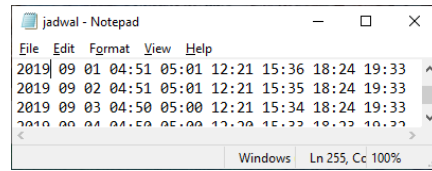
Pengujian dilakukan dengan melihat langsung pada *Seven Segment* jadwal sholat dan membandingkan nilai yang tampil pada *Seven Segment* masing-masing waktu sholat dengan data jadwal sholat yang tersimpan dalam MicroSD. Hasil pengujian dapat dilihat pada gambar 7.



(a)



(b)



(c)

Gambar 7. Pengujian Tampilan *Seven Segment* Jadwal Sholat

- (a) Tampilan Jadwal Sholat Pada Alat
- (b) Tampilan Jadwal Sholat Pada Kalender
- (c) Tampilan Jadwal Sholat Pada MicroSD

Pada gambar 7 terlihat bahwa nilai jadwal sholat tanggal 02 September 2019 yang tampil pada alat dengan nilai jadwal sholat pada kalender serta pada MicroSD adalah akurat. Nilai tersebut secara detail dapat dilihat pada tabel 2.

Tabel 2. Pengujian Data Jadwal Sholat

No.	Nama	Alat	Kalender	MicroSD	Hasil
1	Imsak	04:51	04:51	04:51	Akurat
2	Shubuh	05:01	05:01	05:01	Akurat
3	Dzuhur	12:21	12:21	12:21	Akurat
4	Ashar	15:35	15:35	15:35	Akurat
5	Maghrib	18:24	18:24	18:24	Akurat
6	Isya	19:33	19:33	19:33	Akurat

Dengan demikian dari hasil yang diperoleh dapat dikatakan bahwa rangkaian *Seven Segment* jadwal sholat dapat bekerja dengan baik.

Pengujian Rangkaian *Buzzer*

Pengujian dilakukan dengan memberikan tegangan pada input rangkaian *Buzzer* berupa logika '0' dan logika '1', kemudian mendengarkan bunyi yang dihasilkan oleh *Buzzer*. Hasil pengujian dapat dilihat pada tabel 3.

Tabel 3. Pengujian *Buzzer*

No.	Logika Input	Kondisi <i>Buzzer</i>
1	'0'	Tidak aktif
2	'1'	Aktif/Mengeluarkan Suara/bunyi yang bisa didengar dalam radius ± 30 meter

Dari hasil pengujian dapat dilihat bahwa pada saat logika '0' diberikan, *Buzzer* tidak aktif (tidak mengeluarkan suara). Sedangkan pada saat logika '1' diberikan, *Buzzer* aktif mengeluarkan suara/bunyi yang bisa didengar dalam jarak ± 30 meter. Berdasarkan hasil pengujian tersebut, rangkaian *Buzzer* dapat dikatakan bekerja dengan baik, karena memenuhi syarat untuk dijadikan sebagai tanda masuk waktu sholat.

Pengujian *Running Text*

Pengujian dilakukan dengan melihat langsung pada tampilan *Running Text* dan *Seven Segment* tanggal / hari. Selanjutnya tampilan ini dibandingkan dengan data yang ada pada MicroSD. Hasil pengujian dapat dilihat pada gambar 8.



(a)



(b)



(b)



(c)



(c)



(d)



(d)

Gambar 9. Pengujian Tampilan *Running Text*, pesan yang tampil pada gambar 9.a sampai dengan 9.d : “Asmaul Husna : Emil Naf'an, S.Kom,M.Kom” pada hari Sabtu, tanggal 07-09-2019 jam 06:53:54 sampai dengan jam 06:54:01 WIB



(e)

Gambar 8. Pengujian Tampilan *Running Text*, pesan yang tampil pada gambar 8.a sampai dengan 8.e : “Insha Allah yang mengaji dan tilawah : Boy Oktovery” pada hari Sabtu, tanggal 07-09-2019 jam 06:53:32 sampai dengan jam 06:53:49 WIB

Pada gambar 8 terlihat bahwa pesan pada *Running Text* yang ditampilkan untuk yang bertugas mengaji yang ditampilkan pada hari Sabtu tanggal 07 September 2019. Pesan ini terus berjalan (*running*) mulai dari jam 06:53:32 WIB sampai dengan jam 06:53:49 WIB. Selanjutnya untuk tampilan *Running Text* dosen yang mengumandakan Asmaul Husna dapat dilihat gambar 9.

Pada gambar 9 terlihat bahwa pesan pada *Running Text* yang ditampilkan untuk yang bertugas mengumandangkan Asmaul Husna yang ditampilkan pada hari Sabtu tanggal 07 September 2019. Pesan ini terus berjalan (*running*) mulai dari jam 06:53:54 WIB sampai dengan jam 06:54:01 WIB. Selanjutnya untuk tampilan *Running Text* dosen yang berdo'a dapat dilihat gambar 10.



(a)



(a)



(b)



(c)

Gambar 10. Pengujian Tampilan *Running Text*, pesan yang tampil pada gambar 10.a sampai dengan 10.c : “Do’a : Irsan, ST,MM” pada hari Sabtu, tanggal 07-09-2019 jam 06:54:05 sampai dengan jam 06:54:10 WIB

Pada gambar 10 terlihat bahwa pesan pada *Running Text* yang ditampilkan untuk yang bertugas untuk berdo’a yang ditampilkan pada hari Sabtu tanggal 07 September 2019. Pesan ini terus berjalan (*running*) mulai dari jam 06:54:05 WIB sampai dengan jam 06:54:10 WIB. Selanjutnya pesan yang tampil ini dibandingkan dengan data jadwal kegiatan dosen yang ada pada MicroSD yang dapat dilihat pada gambar 11.



Gambar 11. Tampilan Pesan Jadwal Kegiatan Dosen pada MicroSD

Hasil perbandingan secara detail dapat dilihat pada tabel 3.

Tabel 3. Pengujian Data Pesan Pada *Running Text*

No.	Gabungan Pesan pada gambar 8,9 dan 10)	Data pada MicroSD	Hasil
1	Insya Allah, yang Mengaji dan Tilawah: BOY OKTOVERY Asmaul Husna: EMIL NAF'AN,S.Kom,M.Kom Do'a: IRSAN, ST, MM	2019 09 07 Insya Allah, yang Mengaji dan Tilawah: BOY OKTOVERY Asmaul Husna: EMIL NAF'AN, S.Kom,M.Kom Do'a: IRSAN,ST,MM	Akurat

Pada tabel 3 tersebut terlihat bahwa data MicroSD dan pesan pada *Running Text* adalah akurat, namun pada MicroSD terdapat penambahan tanggal. Hal ini berguna untuk menandakan bahwa pesan tersebut ditampilkan pada tanggal tersebut, yaitu pada tanggal 2019 09 07 (*format yyyy mm dd*) atau pada tanggal 07-09-2019. Pada gambar 8, 9 dan 10 dapat dilihat pada *Seven Segment* tanggal / hari, menampilkan tanggal 07-09-2019. Dengan demikian dari hasil pengujian yang telah dilakukan dapat dikatakan bahwa rangkaian *Running Text* untuk menampilkan jadwal kegiatan dosen dapat bekerja dengan baik.

Pengujian Ketahanan Sistem

Pengujian dilakukan dengan menempelkan jari telunjuk pada bingkai aluminium (*heat sink*) yang berdekatan

dengan IC Regulator rangkaian catudaya. Waktu pengujian dilakukan secara acak (*random*) selama selama ± 7 hari. Dalam masa pengujian ini, peralatan diaktifkan tanpa henti (*non stop*). Hasil pengujian dapat dilihat pada tabel 4.

Tabel 4. Pengujian Ketahanan Sistem

No.	Hari ke-	Waktu	Kondisi IC Regulator (<i>Heat Sink</i>)	Tampilan <i>seven segment</i> dan <i>running text</i>
1	1	08:00	Normal	Akurat
2	1	08:03	Normal	Akurat
3	1	08:06	Normal	Akurat
4	1	08:10	Normal	Akurat
5	1	08:15	Normal	Akurat
6	1	09:30	Normal	Akurat
7	1	10:00	Normal	Akurat
8	1	11:00	Normal	Akurat
9	1	14:00	Normal	Akurat
10	1	18:00	Normal	Akurat
11	1	22:00	Normal	Akurat
12	2	08:00	Normal	Akurat
13	2	11:00	Normal	Akurat
14	2	17:00	Normal	Akurat
15	2	22:00	Normal	Akurat
16	3	08:00	Normal	Akurat
17	3	14:00	Normal	Akurat
18	3	22:00	Normal	Akurat
19	4	08:00	Normal	Akurat
20	4	14:00	Normal	Akurat
21	4	22:00	Normal	Akurat
22	5	08:00	Normal	Akurat
23	5	14:00	Normal	Akurat
24	5	22:00	Normal	Akurat
25	6	08:00	Normal	Akurat
26	6	14:00	Normal	Akurat
27	6	22:00	Normal	Akurat
28	7	08:00	Normal	Akurat
29	7	14:00	Normal	Akurat
30	7	22:00	Normal	Akurat

Pada tabel 4 terlihat bahwa interval pengujian pada hari pertama relatif pendek. Hal ini disebabkan pada hari pertama harus dipastikan bahwa IC Regulator (*heat sink*) yang dipakai bekerja normal (tidak panas). Kisaran suhu normal berada pada antara 27°C s/d 33°C. Pada tabel 4 terlihat bahwa selama 7 hari pengujian kondisi IC Regulator selalu normal. Demikian juga dengan tampilan pada *seven segment* dan *running text* selalu akurat. Dengan demikian dapat dikatakan bahwa sistem yang dibuat dapat bekerja normal dan akurat serta tahan dipakai tanpa henti (*non stop*).

4. Kesimpulan

Berdasarkan penelitian yang telah dilakukan dapat ditarik kesimpulan bahwa peralatan yang dirancang dan dibuat telah dapat bekerja dengan baik layaknya fungsi jadwal sholat yang dijual di pasaran. Sistem yang dirancang juga mempunyai kelebihan fungsi dalam menampilkan jadwal kegiatan dosen secara otomatis. Pengujian pada jadwal sholat diperoleh hasil yang akurat. Begitu juga dengan pengujian tampilan pada *running text* juga diperoleh hasil yang akurat. Pada pengujian jadwal sholat, terdapat perbedaan waktu ± 3 detik, namun ini bisa diatasi dengan penambahan

program kalibrasi setiap hari pada jam 00:05 WIB. Dari segi ketahanan, sistem telah diuji coba selama ± 7 hari diaktifkan tanpa henti (*non stop*). Dengan penggunaan casing aluminium sebagai peredam panas (*heat sink*) catudaya menjadi lebih stabil dalam mensupply arus ke rangkaian. Sampai saat laporan penelitian ini ditulis, jadwal sholat digital dan jadwal kegiatan dosen yang dirancang dan dibuat ini alhamdulillah berfungsi dengan baik dan sudah digunakan pada masjid Rahmatan Lil'alamin Universitas Putra Indonesia "YPTK" Padang selama lebih dari 2 tahun.

Daftar Rujukan

- [1] Sejarah UPI YPTK Padang. [Online] Available at : <http://upiptk.ac.id/profile/sejarah> [Accessed 02nd March 2019].
- [2] Rosad, S., Yudhayana, A. & Fadlil, A. (2019) Jadwal Sholat Digital Menggunakan Metode Ephemeris Berdasarkan Titik Koordinat Smartphone. *Information Technology Journal Research and Development*, 3(2) 30–43. [https://doi.org/10.25299/itjrd.2019.vol3\(2\).2285](https://doi.org/10.25299/itjrd.2019.vol3(2).2285)
- [3] Husain, T. & Ayuna (2018) Perancangan Aplikasi Pengingat Waktu Adzan Berbasis Mikrokontroler Dengan Trigger SMS Gateway Design Applications Reminder Time Adzan Based Microcontroller With Trigger SMS Gateway. *Jurnal VOI*, 7(2), 1–12.
- [4] Risal, M., Munandar, A. H., & Wali, A., R. (2018) Prototype Pengontrolan Alat Elektronik Masjid Berbasis Arduino. *J. INSTEK (Informatika Sains dan Teknol*, 3(1), 81–90. <https://doi.org/10.24252/instek.v3i1.4822>
- [5] Arduino Uno Rev3. [Online] Available at: <https://store.arduino.cc/usa/arduino-uno-rev3> [Accessed 21st August 2019]
- [6] Arduino Mega 2560 Rev3. [Online] Available at: <https://store.arduino.cc/usa/arduino-uno-rev3> [Accessed 21st August 2019]
- [7] Arduino Ethernet Shield 2. Available at: [http://https://store.arduino.cc/usa/arduino-uno-rev3](https://store.arduino.cc/usa/arduino-uno-rev3) [Accessed 21st August 2019]
- [8] Micro SD Card Micro SDHC Mini TF Card Adapter Reader Module for Arduino. [Online] Available at: <http://datalogger.pbworks.com/w/file/fetch/89507207/Datalogger%20-%20SD%20Memory%20Reader%20Datasheet.pdf> [Accessed 15th November 2019]
- [9] Ilmu Falak Praktik. [Online] (Updated 1st October 2013) Available at : <http://simbi.kemenag.go.id/pustaka/images/materibuku/ilmu%20falak%20paraktik-2013.pdf> [Accessed 15th November 2019]
- [10] Nafan, E. & Gushelmi (2017) *Sistem Kendali Scoring Board Futsal Berbasis Mikrokontroler Atmega8535 dengan Komunikasi Rs-485 dan Bluetooth*. Seminar Nasional: Peranan Ipteks Menuju Industri Masa Depan (PIMIMD), 128–38
- [11] 2.3 inch (58.42 mm) 8X8 Dot Matrix Led Display. Available at : http://pdf.datasheetcatalog.com/datasheets/134/300343_DS.pdf [Accessed 24th October 2019]
- [12] Real Time Clock RTC Module Using DS1307 [Online] (Updated 27th August 2016) Available at : <https://www.twovolt.com/tag/real-time-clock-rtc-module-using-ds1703/> [Accessed 16th November 2019]
- [13] Use a Buzzer Module (Piezo Speaker) using Arduino [Online] (Updated 05th June 2018) Available at : <https://create.arduino.cc/projecthub/SURYATEJA/use-a-buzzer-module-piezo-speaker-using-arduino-uno-89df45> [Accessed 15th November 2019]
- [14] Kadir, A. (2013) *Panduan Praktis Mempelajari Aplikasi Mikrokontroler dan Pemrograman menggunakan Arduino*. Yogyakarta: Andi Offset.
- [15] Setiawan, A. (2011). *20 Aplikasi Mikrokontroler ATmega 8535 dan ATmega16 Menggunakan BASCOM-AVR*. Yogyakarta: Andi Offset.