

## SISTEM SIRKULASI UDARA DAN PENCAHAYAAN OTOMATIS DI DALAM RUMAH

<sup>[1]</sup>Hirzen Hasfani, <sup>[2]</sup>Dedi Triyanto, <sup>[3]</sup>Fatma Agus Setyaningsih  
<sup>[1][2][3]</sup>Jurusan Sistem Komputer, Fakultas MIPA Universitas Tanjungpura  
Jl. Ahmad Yani, Pontianak  
Telp./Fax.: (0561) 577963  
e-mail:  
<sup>[1]</sup>hirzen.hasfani@gmail.com, <sup>[2]</sup>dedi.triyanto@siskom.untan.ac.id,  
<sup>[3]</sup>fatmasetyaningsih@gmail.com

### ABSTRAK

*Sirkulasi udara dan pencahayaan merupakan salah satu aspek yang berpengaruh pada tingkat kenyamanan rumah. Sistem pengoperasian perangkat elektronik maupun non elektronik dalam rumah seperti lampu, kipas dan tirai kurang efisien karena harus dioperasikan secara manual. Oleh karena itu dalam penelitian ini menerapkan sistem sirkulasi udara dan pencahayaan otomatis di dalam rumah dengan Arduino sebagai unit pemroses sistem yang akan mengeluarkan logika kepada mikrokontroler ATmega16. Sensor cahaya digunakan sebagai pendeteksi intensitas cahaya, sensor suhu sebagai pendeteksi suhu, dan RTC sebagai parameter waktu. Hasil penelitian berupa sistem yang dapat mengatur sirkulasi udara dan pencahayaan di dalam rumah secara otomatis berdasarkan data waktu yang telah ditentukan pengguna melalui aplikasi antarmuka dan sensor. Pada sistem lampu, lampu akan hidup jika nilai ADC lebih dari 100 di siang hari. Sedangkan pada malam hari nyala lampu diatur melalui aplikasi antarmuka. Pada sistem kipas, kipas akan mati jika suhu dibawah 31 °C. Pada suhu diantara 30 °C sampai 33 °C, maka kipas akan hidup dengan kecepatan lambat sedangkan pada suhu diatas 33 °C, kipas akan hidup dengan kecepatan tinggi. Pada sistem tirai, penjadwalan buka tutup tirai dapat diatur melalui aplikasi antarmuka*

**Kata kunci:** Sistem Pencahayaan Otomatis, Sirkulasi Udara, Arduino, Mikrokontroler, ATmega16

### 1. PENDAHULUAN

Pada kehidupan sehari-hari masih banyak perangkat elektronik maupun non elektronik yang harus dioperasikan secara manual dalam rumah seperti lampu dan tirai yang harus dioperasikan secara manual. Namun kesibukan manusia membuat mereka terlambat bahkan lupa untuk mengoperasikan lampu dan tirai jendela di dalam rumah. Selain itu sebuah rumah yang nyaman juga perlu memenuhi tingkat kenyamanan tertentu pada berbagai aspek yang salah satunya adalah sirkulasi udara. Apabila di dalam rumah terasa sesak dan panas, manusia biasanya akan mengoperasikan pendingin ruangan, kipas angin atau sejenisnya.

Penelitian ini menerapkan sistem otomatisasi untuk mengatasi masalah tersebut. Otomatisasi adalah penggunaan peralatan mekanik dan atau elektronik yang

menggantikan peranan manusia. Otomatisasi berguna antara lain untuk tugas yang monoton, berulang atau berbahaya. Sistem otomatisasi dapat dikendalikan menggunakan *chip* mikrokontroler yang ditanamkan program logika baik yang sederhana maupun kompleks [1].

Pada penelitian ini akan dikembangkan suatu alat yang dapat dioperasikan secara otomatis menggunakan aplikasi antarmuka. Dalam penelitian ini akan dibuat suatu sistem sirkulasi udara dan pencahayaan otomatis di dalam rumah.

### 2. LANDASAN TEORI

#### 2.1 Pengantar Rumah

Kriteria rumah menurut Winslow dalam buku "Pengantar Kesehatan Lingkungan" adalah rumah yang dapat memenuhi kebutuhan fisiologis dan juga dapat memenuhi kebutuhan psikologis [2].

## 2.2 Arduino Uno

Arduino adalah *platform* pembuatan prototipe elektronik yang bersifat *open-source hardware* yang berdasarkan pada perangkat keras dan perangkat lunak yang fleksibel dan mudah digunakan. Bahasa yang dipakai dalam Arduino adalah bahasa C yang disederhanakan dengan bantuan pustaka-pustaka Arduino. Arduino juga menyederhanakan proses bekerja dengan mikrokontroler, sekaligus menawarkan berbagai macam kelebihan [3]. Gambar 1 merupakan bentuk fisik dari Arduino Uno.



Gambar 1. Arduino Uno [3]

## 2.3 Mikrokontroler ATmega16

Mikrokontroler AVR (*Alf and Vegard's Risc processor*) ATmega16 standar memiliki arsitektur 8 bit, dimana semua dikemas dalam kode 16 bit, dan sebagian besar instruksi dieksekusi dalam 1 (satu) siklus *clock*. AVR berteknologi RISC (*Reduced Instruction Set Computing*), sedangkan seri MCS51 berteknologi CISC (*Complex Instruction Set Computing*). AVR dapat dikelompokkan menjadi empat kelas, yaitu keluarga Attiny, keluarga AT90Sxx, keluarga ATmega, AT86RFxx. Pada dasarnya, yang membedakan masing-masing kelas adalah memori, peripheral, dan fungsinya [4].

## 2.4 Sensor Suhu LM35

Sensor suhu LM35 adalah komponen elektronika yang memiliki fungsi untuk mengubah besaran suhu menjadi besaran listrik dalam bentuk tegangan. LM35 memiliki keakuratan tinggi dan kemudahan perancangan jika dibandingkan dengan sensor suhu yang lain, LM35 juga mempunyai keluaran impedansi yang rendah dan linearitas yang tinggi sehingga dapat dengan mudah dihubungkan dengan rangkaian kendali khusus serta tidak memerlukan penyetelan lanjutan [5].

## 2.5 Sensor Cahaya LDR

*Light Dependent Resistors* (LDR) adalah resistor yang terbuat dari cadmium sulfida yaitu merupakan bahan semikonduktor yang resistansinya berubah-ubah menurut banyaknya cahaya (sinar) yang mengenainya [6].

Multimeter dapat digunakan untuk menentukan tahanan dalam keadaan gelap dan terang, ini adalah standar LDR : [7].

- Dalam keadaan gelap tahanan maksimum, sekitar 1 M $\Omega$ .
- Dalam keadaan terang tahanan minimum, sekitar 1 K $\Omega$ .

## 2.6 Real Time Clock (RTC)

*Real Time Clock* merupakan suatu IC yang memiliki fungsi sebagai penyimpan waktu dan tanggal. *RTC* adalah jam elektronik dengan chip yang dapat menghitung waktu dan menyimpan data waktu tersebut secara real time. *RTC* cukup akurat sebagai pewaktu karena menggunakan osilator kristal. Salah satu IC *RTC* yang akan digunakan adalah DS 1307 [5].

## 2.7 Visual Basic

*Visual Basic* adalah salah satu bahasa pemrograman berorientasi objek (*Object Oriented Programming*) yang mudah dipelajari. Visual Basic telah menyediakan kontrol MSComm yang digunakan untuk mengakses *port* serial secara langsung. Kontrol MSComm menyediakan fasilitas komunikasi antara program aplikasi yang akan dibuat dengan *port* serial untuk mengirim atau menerima data melalui data melalui *port* serial. Setiap MSComm hanya menangani satu *port* serial sehingga jika ingin menggunakan lebih dari satu *port* serial harus menggunakan MSComm lain [8].

## 3. METODOLOGI PENELITIAN

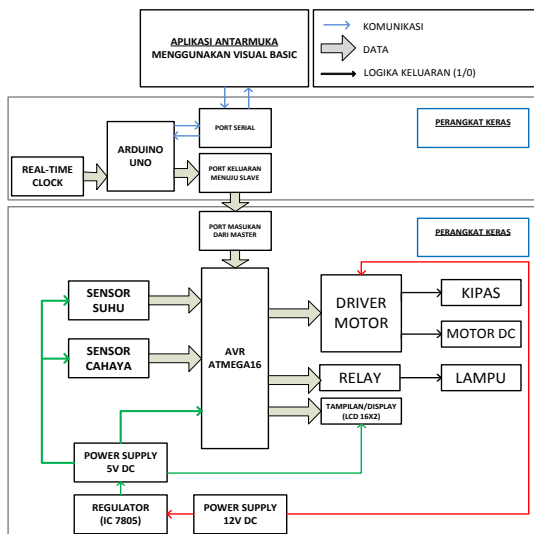
Penelitian ini menggunakan metodologi penelitian yang dimulai dari studi pustaka tentang referensi yang telah ada dan komponen-komponen yang dibutuhkan dalam pembuatan *prototype* sistem sirkulasi udara dan pencahayaan otomatis di dalam rumah. Tahap selanjutnya adalah perancangan yang diawali dengan analisis kebutuhan sistem kendali alat dari perangkat lunak maupun perangkat keras. Tahap selanjutnya adalah melakukan

pengintergrasian dari perangkat keras dan perangkat lunak menggunakan aplikasi antarmuka. Tahap selanjutnya adalah melakukan pengujian pada perangkat keras dan perangkat lunak sehingga sistem dapat bekerja secara otomatis. Tahap terakhir adalah tahap analisa dan penerapan yang dilakukan setelah alat bekerja dengan baik.

#### 4. PERANCANGAN DAN IMPLEMENTASI

##### 4.1. Diagram Blok Perancangan Sistem

Dari gambar 2, dapat diketahui bahwa aplikasi antarmuka akan mengambil data pada Arduino kemudian mengirim data jadwal ke Arduino melalui port komunikasi serial. Kemudian Arduino yang berperan sebagai rangkaian master akan mengirimkan logika kepada mikrokontroler melalui port input/output untuk menghidupkan atau mematikan lampu dan menggerakkan motor DC pada tirai dan Kipas DC.



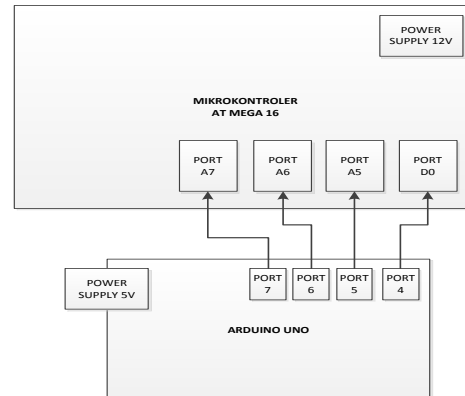
Gambar 2. Diagram Blok Sistem

#### 4.2. Perancangan Perangkat Keras

##### 4.2.1. Perancangan Arduino dan Mikrokontroler ATmega16

Penelitian mengenai sistem sirkulasi udara dan pencahayaan otomatis ini menggunakan dua buah perangkat mikrokontroler yaitu mikrokontroler ATmega 16 dan Arduino. Sumber tegangan yang digunakan sebesar 12 VDC untuk mikrokontroler ATmega16 dan Arduino menggunakan sumber tegangan sebesar 5 VDC.

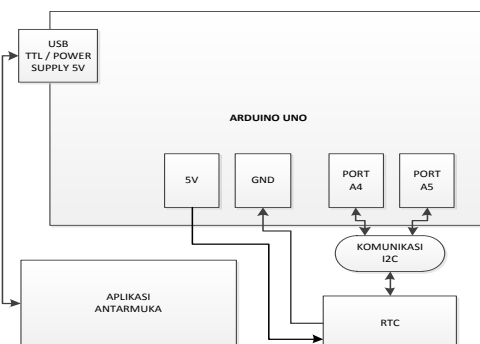
Port mikrokontroler ATmega16 berlaku sebagai masukan dan port Arduino mengeluarkan logika 1 atau 0 sebagai keluaran. Port A7 ATmega 16 dihubungkan pada port 7 Arduino. Sedangkan port A6 pada mikrokontroler terhubung dengan port 6 Arduino, port A5 pada mikrokontroler dihubungkan dengan port 5 Arduino, serta port D0 pada mikrokontroler dihubungkan ke port 4 pada Arduino.



Gambar 3. Arduino dan Mikrokontroler ATmega16

##### 4.2.2. Perancangan Arduino dan RTC

RTC berperan sebagai sumber data waktu pada Arduino, sehingga sistem dapat bekerja secara otomatis. Dalam perancangan ini, RTC terhubung dengan port analog 4 dan port analog 5. Proses pengambilan data pada RTC menggunakan komunikasi I2C.

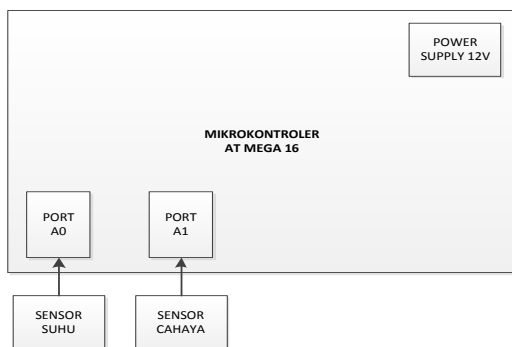


Gambar 4. Arduino dan RTC

##### 4.2.3. Perancangan Mikrokontroler dan Sensor

Pada penelitian ini digunakan sumber tegangan sebesar 12 VDC kemudian menggunakan IC Regulator untuk menurunkan tegangan mikrokontroler menjadi 5 VDC.

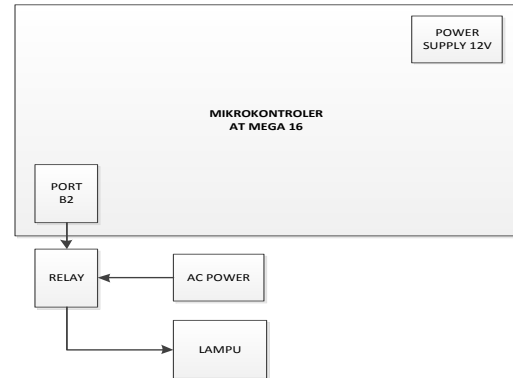
Sensor yang digunakan yaitu sensor suhu LM35 dan sensor cahaya LDR. Sensor suhu digunakan untuk mengetahui besaran suhu yang akan dijadikan acuan untuk mengendalikan nyala kipas. Sensor cahaya digunakan untuk mengetahui besaran perubahan penerimaan cahaya berdasarkan dari perubahan resistansinya dan dijadikan acuan untuk mengendalikan nyala lampu. Sensor tersebut dihubungkan pada port ADC pada mikrokontroler. Mikrokontroler mendapatkan nilai pembacaan dari masing-masing sensor tersebut dan diteruskan ke masing-masing keluaran yang telah ditentukan. Sensor suhu dihubungkan dengan port A0 dan sensor cahaya dihubungkan dengan port A5 pada mikrokontroler.



Gambar 5. Mikrokontroler dan Sensor

#### 4.2.4. Perancangan Mikrokontroler dan Relay

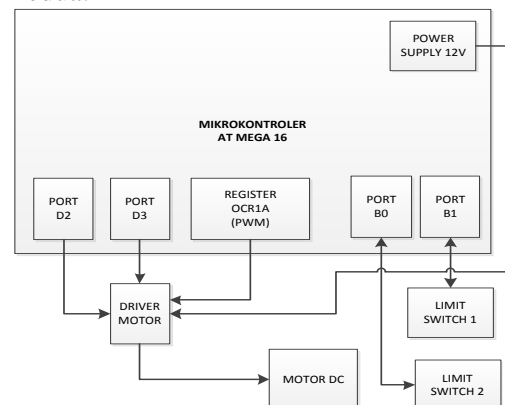
Pada penelitian ini digunakan sumber tegangan sebesar 12 VDC untuk mikrokontroler ATmega16, sedangkan relay digunakan sumber tegangan 220 VAC / Listrik PLN. Relay digunakan sebagai saklar elektromagnetik yang memutus atau menghubungkan arus listrik AC. Relay menerima data atau sinyal yang berupa logika 1 atau 0 dari mikrokontroler ATmega16. Dalam penelitian ini hanya digunakan 1 relay yang fungsinya untuk menghubungkan atau memutus arus listrik pada lampu. Relay dihubungkan dengan port B2 pada mikrokontroler yang berlaku sebagai keluaran. Jika relay menerima logika 1, maka relay tersebut akan aktif dan lampu akan menyala. Jika relay menerima logika 0, maka relay tersebut akan mati dan lampu akan padam.



Gambar 6. Mikrokontroler dan Relay

#### 4.2.5. Perancangan Mikrokontroler dan Motor DC

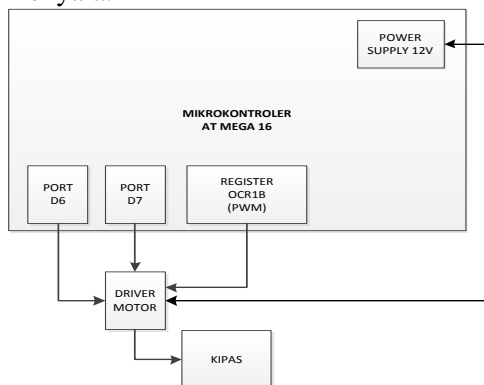
Pada penelitian ini digunakan sumber tegangan sebesar 12 VDC untuk mikrokontroler ATmega16 dan driver motor. Driver motor digunakan untuk mengendalikan arah putaran dan kecepatan pada motor DC yang akan menggerakkan tirai baik itu terbuka maupun tertutup. Pada rangkaian driver motor dihubungkan ke port D2 dan port D3 serta register OCR1A yang berfungsi sebagai *port Pulse Width Modulation (PWM)*. Kemudian terdapat dua limit switch yang akan digunakan untuk menghentikan putaran motor. *Limit switch* pertama dihubungkan ke port B1 dan limit switch kedua dihubungkan dengan ke port B0 pada mikrokontroler. Ketika tirai terbuka, maka motor akan aktif ke arah berlawanan jarum jam dan berhenti ketika tirai mencapai limit switch pertama. Kondisi saat tirai tertutup adalah motor akan berputar searah jarum jam dan berhenti ketika tirai mencapai *limit switch* kedua.



Gambar 7. Mikrokontroler dan Motor DC.

#### 4.2.6. Perancangan Mikrokontroler dan Kipas DC

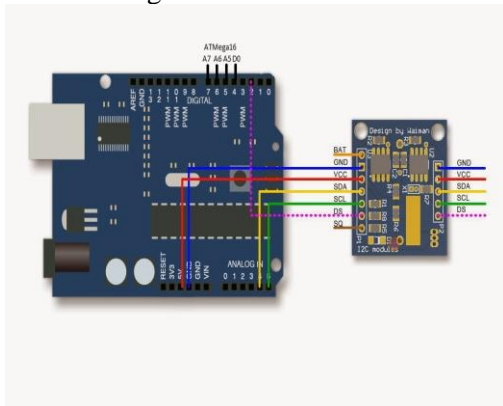
Pada penelitian ini digunakan tegangan sumber sebesar 12 VDC untuk mikrokontroler ATmega16 dan driver motor. Driver motor digunakan untuk mengendalikan arah putaran dan kecepatan pada kipas DC. Pada rangkaian driver motor dihubungkan ke port D6 dan port D7 serta register OCR1B yang berfungsi sebagai port PWM. Ketika port D6 menerima data logika 1 dan port D7 menerima data logika 0 dari mikrokontroler, maka kipas tersebut akan menyala.



Gambar 8. Mikrokontroler dan Kipas DC

#### 4.2.7 Perancangan Keseluruhan Perangkat Keras

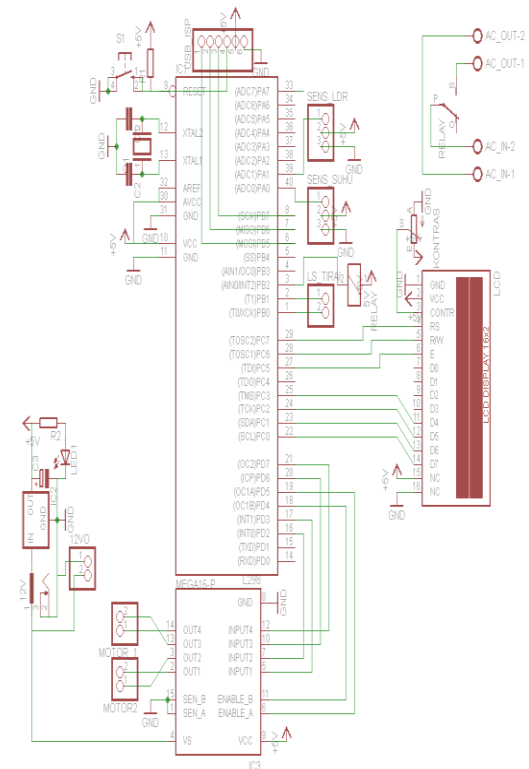
Gambar 9 merupakan gambar dari skematik rangkaian arduino.



Gambar 9. Skematik Rangkaian Arduino

Setelah dilakukan perancangan terhadap perangkat keras, maka dilanjutkan dengan tahap perancangan untuk keseluruhan perangkat keras. Perangkat keras seperti RTC dan Mikrokontroler ATmega16 dihubungkan dengan Arduino, sedangkan sensor LM35, sensor LDR,

LCD, relay, dan limit switch dihubungkan dengan mikrokontroler ATmega16 melalui port-port yang telah ditentukan. Pada tahap ini keseluruhan perangkat keras diwujudkan menjadi sebuah prototype rumah.



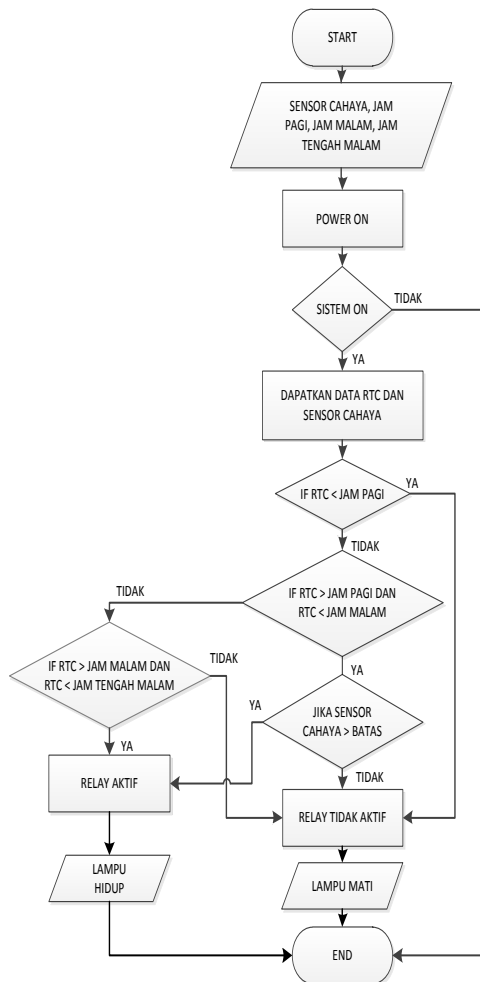
Gambar 10. Skematik Rangkaian Mikrokontroler ATmega 16

### 4.3. Perancangan Perangkat Lunak

#### 4.3.1. Diagram Alir Sistem Pada Lampu

Program dimulai dengan inisialisasi sensor cahaya, jam pagi, jam malam, dan jam tengah malam. Pada saat sistem dinyalakan, sistem akan membaca data waktu dan sensor dengan mengambil data dari RTC dan sensor cahaya. Pada kondisi pertama, jika data waktu yang didapat kurang dari jam pagi maka lampu akan mati. Jika tidak maka selanjutnya dilakukan proses pengecekan kondisi kedua. Jika data waktu yang didapat lebih dari jam pagi dan kurang dari jam malam, maka sensor akan bekerja sesuai fungsinya. Jika cahaya melewati batas yang telah ditentukan (gelap), maka lampu akan hidup. Begitu juga sebaliknya, jika cahaya belum melewati batas yang telah ditentukan (terang), maka lampu akan mati. Jika tidak

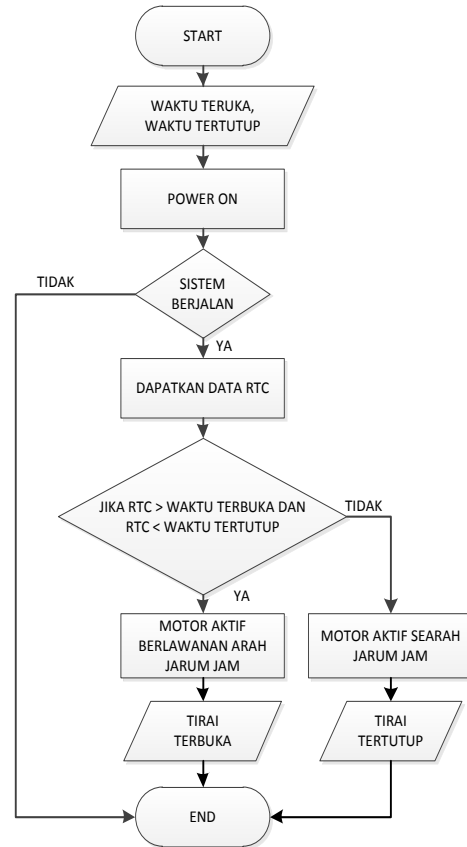
terpenuhi kondisi kedua, maka akan dilakukan pengecekan kondisi ketiga. Jika data waktu yang didapat lebih dari jam malam dan kurang dari jam tengah malam, maka lampu akan hidup, tetapi jika tidak lampu akan mati.



**Gambar 11.** Diagram Alir Pada Lampu

4.3.2. Diagram Alir Sistem Pada Tirai

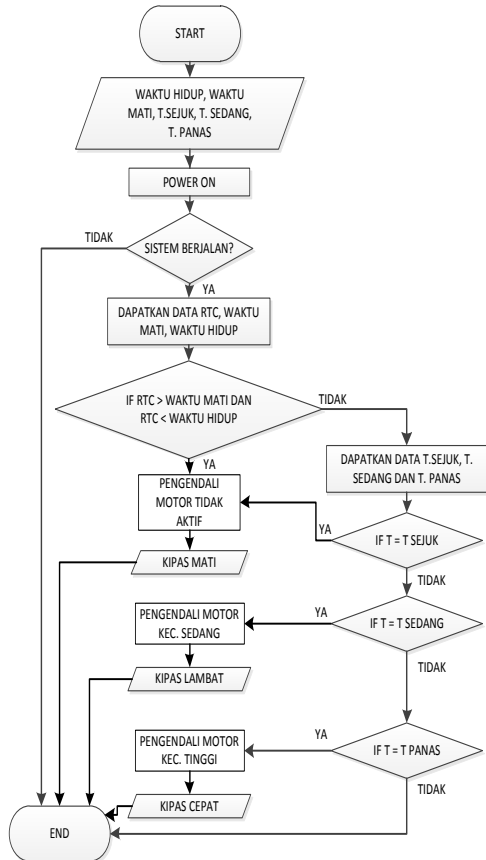
Program dimulai dari inisialisasi waktu terbuka dan waktu tertutup. Pada sistem saat dinyalakan, sistem akan membaca data waktu dengan mengambil data dari RTC. Jika data waktu yang didapat lebih dari waktu terbuka dan kurang dari waktu tertutup, maka tirai akan terbuka tetapi jika tidak maka tirai akan tertutup.



**Gambar 12.** Diagram Alir Pada Tirai

4.3.3. Diagram Alir Sistem Pada Kipas

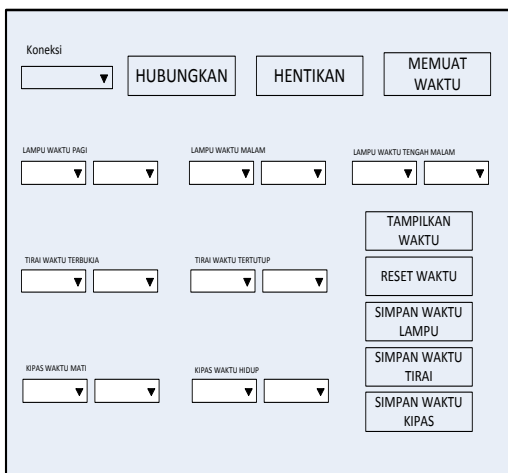
Program dimulai dengan inisialisasi waktu hidup, waktu mati, suhu sejuk, suhu sedang, dan suhu panas. Pada saat sistem dinyalakan, sistem akan membaca data waktu dan sensor suhu dengan mengambil data dari RTC dan sensor suhu. Jika data waktu yang didapat lebih dari waktu mati dan kurang dari waktu hidup, maka kipas akan mati. Jika tidak maka sensor suhu akan bekerja sesuai fungsinya. Setelah mengambil data suhu sejuk, sedang dan panas dari sensor suhu, maka dilakukan pengecekan pada kondisi pertama. Jika suhu sejuk, maka kipas akan mati. Jika suhu sedang, maka kipas akan hidup dengan kecepatan lambat. Jika suhu panas, maka kipas juga akan hidup dengan kecepatan cepat.



Gambar 13. Diagram Alir Pada Kipas

#### 4.3.4. Perancangan Antarmuka

Gambar 14 merupakan tampilan dari perancangan antarmuka.



Gambar 14. Perancangan Antarmuka

Penjelasan tombol-tombol dari aplikasi antarmuka :

1. Tombol Hubungkan digunakan untuk menghubungkan koneksi aplikasi antarmuka ke perangkat keras.

2. Tombol Hentikan digunakan untuk memutuskan koneksi aplikasi antarmuka ke perangkat keras.
3. Tombol Memuat Data untuk mengambil data waktu pada perangkat keras.
4. Tombol Tampilkan Waktu digunakan untuk menampilkan data waktu setelah data diambil dari perangkat keras.
5. Tombol Reset Waktu untuk mengatur ulang data waktu menjadi nol.
6. Tombol Simpan Waktu Lampu untuk menyimpan data kondisi waktu lampu yang sudah ditentukan pengguna.
7. Tombol Simpan Waktu Tirai digunakan untuk menyimpan data kondisi waktu tirai yang sudah ditentukan pengguna.
8. Tombol Simpan Waktu Kipas untuk menyimpan data kondisi waktu kipas yang sudah ditentukan pengguna.

## 5. PENGUJIAN SISTEM DAN IMPLEMENTASI

### 5.1. Pengujian Terhadap Sistem Sirkulasi Udara dan Pencahayaan Otomatis

#### 5.1.1. Pengujian Sensor Suhu

Mikrokontroler akan membaca nilai suhu ruangan menggunakan sensor suhu LM35. Sensor tersebut menggunakan prinsip pembacaan suhu secara digital dengan kenaikan sebesar 10mv/°C dan akurasi ±1/4 °C pada suhu ruangan. Pada Perangkat keras akan dilakukan pengujian sensor suhu pada kondisi ruangan. Nilai suhu ruangan akan ditampilkan pada LCD. Sensor suhu LM35 digunakan termometer digital untuk membandingkan nilai suhu yang didapat dari termometer dan sensor suhu LM35. Hasil pengukuran sensor suhu LM35 terhadap termometer dapat dilihat pada Tabel 1.

Tabel 1. Pengujian Sensor Suhu

V masukan (V)	Pembacaan ADC	Suhu Termometer (°C)	Suhu Tampilan LCD (°C)	Error (%)
0,24	50	24,3	24	1,23
0,25	51	25,5	25	1,96
0,26	53	26,4	26	1,51
0,27	55	26,9	27	0,37
0,28	57	27,6	28	1,44
0,29	59	28,7	29	1,04
0,31	63	31,2	31	0,64
0,34	70	33,8	34	0,59

Berdasarkan Tabel 1 maka dapat disimpulkan bahwa pembacaan nilai suhu pada termometer dan tampilan LCD memiliki persentase error di bawah 5 %. Perhitungan persentase error untuk nilai ADC pada sensor cahaya dapat dihitung dengan rumus dibawah ini.

$$\%error = ( A - B ) / A * 100\%$$

%error = nilai persentase kesalahan pengukuran

A = nilai alat yang jadi patokan pengukuran atau perhitungan

B = nilai bacaan dari sistem

Pada data pengamatan pertama, persentase error dari suhu yang didapat adalah

$$\%error = ( 24,3 - 24 ) / 24,3 * 100\%$$

$$\%error = 1,23 \%$$

Nilai persentase error pada pengamatan yang berbeda juga dapat dihitung dengan menggunakan persamaan yang sama. Hal ini menunjukkan bahwa sensor suhu bekerja sesuai dengan fungsinya. Besar nilai temperatur yang diperoleh dari sensor suhu terlihat pada tampilan LCD dengan nilai yang mendekati hasil pengukuran menggunakan termometer.

#### 5.1.2. Pengujian Sensor Cahaya

Mikrokontroler akan membaca nilai ADC dari sensor cahaya LDR yang kemudian akan ditampilkan pada LCD. Pengujian sensor cahaya menggunakan nilai ADC untuk menentukan status nyala lampu. Hasil pengujian dari sensor cahaya LDR dapat dilihat pada Tabel 2.

Tabel 2. Pengujian Sensor Cahaya

Waktu Pengukuran	V Masukan (V)	Tampilan Nilai ADC pada LCD	Nilai Hitung ADC	Error (%)
06.00	0,30	64	61,38	4,26
07.00	0,09	19	18,41	3,20
08.00	0,07	15	14,32	4,74
09.00	0,03	6	6,13	2,12
12.00	0,02	4	4,09	2,20
15.00	0,04	8	8,18	2,20
17.30	0,59	127	120,71	5,21

Pada tabel 2 menunjukkan hasil pengujian terhadap sensor cahaya LDR. Pengukuran tegangan masukan pada sensor cahaya didapatkan dari hasil pengukuran menggunakan multimeter. Perhitungan nilai ADC pada sensor cahaya dapat dilihat pada rumus dibawah ini.

Pada pukul 06.00 :

$$ADC = \frac{V_{input}}{V_{ref}} \times 1023$$

$$ADC = \frac{0,30}{5 \text{ v}} \times 1023$$

$$ADC = 61,38 \approx 61$$

Perhitungan persentase *error* untuk nilai ADC pada sensor cahaya dapat lihat di bawah ini.

Pada pukul 06.00 :

$$\%error = ( 61,38 - 64 ) / 61,38 * 100\%$$

$$\%error = 4,26 \%$$

Nilai tegangan masukan dan persentase error pada waktu yang berbeda dapat dihitung dengan menggunakan masing-masing persamaan. Hasil perhitungan nilai ADC dan tampilan nilai ADC di LCD memiliki persentase error sebesar di bawah 10 %. Hasil pengujian terhadap sensor cahaya dapat disimpulkan bahwa besarnya intensitas cahaya dapat dilihat dari besarnya nilai ADC yang ditampilkan pada LCD. Semakin besar intensitas cahaya maka nilai ADC semakin kecil. Begitu juga sebaliknya, semakin kecil intensitas cahaya maka nilai ADC semakin besar. Dapat dilihat bahwa sensor cahaya LDR cukup sensitif dalam membaca kondisi pencahayaan. Perubahan nilai ADC dari pukul 06.00 sampai 12.00 mengalami penurunan nilai ADC yang menunjukkan intensitas cahaya semakin besar. Kemudian dari pukul 15.00 sampai 17.30 terjadi peningkatan nilai ADC yang menunjukkan intensitas cahaya semakin kecil. Hal ini menunjukkan bahwa sensor cahaya LDR cukup sensitif dan mampu merespon perubahan intensitas cahaya baik terang maupun gelap.

#### 5.1.3. Pengujian Terhadap Sistem Lampu

Sistem akan bekerja otomatis dengan menggunakan waktu yang berasal dari RTC dan sensor. Pengujian terhadap Sistem Lampu dilakukan selama 3 hari dengan sampel waktu 06.00, 09.00, 12.00, 15.00 dan 17.30 untuk mengetahui sistem berjalan sesuai dengan kondisi yang ditentukan. Sistem nyala lampu diatur berdasarkan jadwal waktu yang ditentukan pada antarmuka dan dipengaruhi oleh intensitas cahaya dari sensor cahaya LDR. Hasil dari



pengujian terhadap sistem lampu dapat dilihat pada Tabel 3 dan Tabel 4.

Tabel 3. Tabel Pengujian Sistem Lampu Berdasarkan Intensitas Cahaya

Hari	Waktu	Nilai ADC	Status Lampu
Hari 1	06.00	61	Mati
	09.00	8	Mati
	12.00	3	Mati
	15.00	10	Mati
	17.30	132	Hidup
Hari 2	06.00	59	Mati
	09.00	7	Mati
	12.00	4	Mati
	15.00	8	Mati
	17.30	121	Hidup
Hari 3	06.00	64	Mati
	09.00	9	Mati
	12.00	3	Mati
	15.00	7	Mati
	17.30	134	Hidup

Berdasarkan tabel 3 dapat dilihat bahwa pada hari pertama, kedua, dan ketiga pengujian terjadi perubahan intensitas cahaya. Pada pukul 06.00 sampai pukul 12.00 terjadi penurunan nilai ADC yang menunjukkan intensitas cahaya semakin besar. Pada pukul 15.00 sampai pukul 17.30 terjadi peningkatan nilai ADC secara terus-menerus yang menunjukkan intensitas cahaya semakin kecil. Pada pukul 17.30 mengalami peningkatan nilai ADC sehingga membuat status lampu menjadi hidup. Hal ini menunjukkan alat sudah bekerja dengan baik karena telah sesuai dengan kondisi yang diinginkan

Tabel 4. Tabel Pengujian Sistem Lampu Berdasarkan Waktu

Hari	Waktu	Status Lampu yang Diinginkan	Status Lampu Saat Pengujian
Hari 1	18.00 - 23.00	Hidup	Hidup
	23.00 - 06.00	Mati	Mati
Hari 2	18.00 - 23.00	Hidup	Hidup
	23.00 - 06.00	Mati	Mati
Hari 3	18.00 - 23.00	Hidup	Hidup
	23.00 - 06.00	Mati	Mati

Berdasarkan tabel 4 dapat dilihat bahwa hasil pengujian dari hari pertama, kedua, dan ketiga yaitu pada pukul 18.00 sampai 23.00 dan 23.00 sampai 06.00 sudah sesuai dengan status lampu yang diinginkan. Hal ini menunjukkan bahwa alat sudah bekerja dengan baik karena telah sesuai dengan waktu yang ditentukan.

#### 5.1.4. Pengujian Terhadap Sistem Tirai

Pengujian terhadap sistem tirai otomatis dilakukan selama 3 hari untuk mengetahui sistem berjalan sesuai dengan perintah yang diinginkan. Sistem buka tutup tirai diatur berdasarkan jadwal waktu pada RTC. Hasil pengujian sistem tirai otomatis dapat dilihat pada Tabel 5.

Tabel 5. Tabel Pengujian Sistem Tirai Berdasarkan Waktu

Hari	Waktu	Status Tirai yang Diinginkan	Status Tirai Saat Pengujian
Hari 1	06.00 – 18.00	Terbuka	Terbuka
	18.00 - 06.00	Tertutup	Tertutup
Hari 2	06.00 – 18.00	Terbuka	Terbuka
	18.00 – 06.00	Tertutup	Tertutup
Hari 3	06.00 – 18.00	Terbuka	Terbuka
	18.00 – 06.00	Tertutup	Tertutup

Berdasarkan tabel 5 dapat dilihat bahwa hasil pengujian alat pada hari pertama, kedua dan ketiga yaitu pada pukul 06.00–18.00 dan 18.00–06.00 sudah sesuai dengan status tirai yang diinginkan. Hal ini menunjukkan bahwa alat sudah bekerja dengan baik karena alat tersebut berjalan sesuai dengan waktu yang di-tentukan.

#### 5.1.5. Pengujian Terhadap Sistem Kipas

Pengujian terhadap sistem kipas otomatis dilakukan selama 3 hari untuk mengetahui sistem berjalan sesuai dengan perintah yang diinginkan. Sistem nyala kipas diatur berdasarkan jadwal waktu pada RTC dan suhu yang didapatkan pada sensor suhu LM35. Hasil pengujian sistem kipas otomatis dapat dilihat pada Tabel 6 dan Tabel 7.

Tabel 6. Tabel Pengujian Sistem Kipas Berdasarkan Suhu

Hari	Waktu	Tampilan Suhu LCD (°C)	Status Kipas
Hari 1	12.00	33	Hidup, Kecepatan Tinggi
	16.00	31	Hidup, Kecepatan Lambat
	20.00	29	Mati
	00.00	28	Mati
	04.00	27	Mati
Hari 2	12.00	31	Hidup, Kecepatan Lambat
	16.00	30	Mati
	20.00	29	Mati
	00.00	28	Mati
	04.00	27	Mati
Hari 3	12.00	32	Hidup, Kecepatan Lambat
	16.00	29	Mati
	20.00	28	Mati
	00.00	27	Mati
	04.00	26	Mati

Berdasarkan tabel 6 dapat dilihat bahwa hasil pengujian dari sistem kipas otomatis terjadi perubahan suhu sehingga dapat disimpulkan bahwa semakin besar suhu maka kipas akan hidup dan terjadi peningkatan kecepatan kipas. Pada suhu dibawah 31 °C maka kipas akan mati. Ketika siang hari pada saat suhu lebih dari 30 °C dan kurang dari 33 °C maka kipas akan hidup dengan kecepatan lambat dan pada saat suhu lebih dari 32 °C maka kipas akan hidup dengan kecepatan tinggi. Kecepatan putaran motor kipas akan dikendalikan oleh driver motor yang dipengaruhi oleh Pulse With Modulation (PWM) dan pada mikrokontroler berdasarkan register OCR1B, jika kondisi kecepatan lambat maka register OCR1B akan bernilai 100 dan jika kondisi kecepatan cepat maka register OCR1B akan bernilai 255. Kecepatan motor kipas dapat menyesuaikan suhu sehingga menunjukkan sistem kipas otomatis berjalan sesuai dengan fungsi dan kondisi yang diinginkan

Berdasarkan tabel 7 dapat dilihat bahwa hasil pengujian pada hari pertama, kedua dan ketiga yaitu pada pukul 08.00 sampai 12.00 telah sesuai dengan status kipas yang diinginkan. Hal ini menunjukkan bahwa alat sudah bekerja

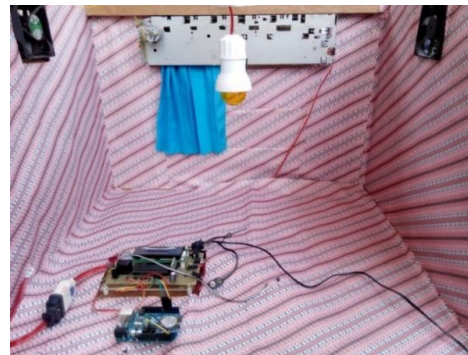
dengan baik karena telah sesuai dengan waktu yang ditentukan.

Tabel 7. Tabel Pengujian Sistem Kipas Berdasarkan Waktu

Hari	Waktu	Status Kipas yang Diinginkan	Status Kipas Saat Pengujian
Hari 1	08.00 - 12.00	Mati	Mati
Hari 2	08.00 - 12.00	Mati	Mati
Hari 3	08.00 - 12.00	Mati	Mati

## 5.2. Implementasi Perancangan Perangkat Keras

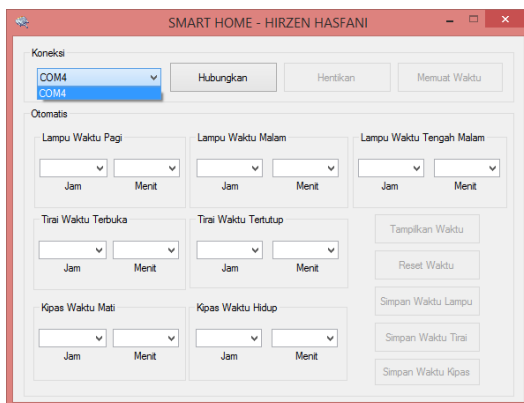
Pada penelitian ini digunakan perangkat keras yang terdiri dari Arduino, Mikrokontroler ATmega16, Kipas DC, Motor DC yang menggerakkan tirai, lampu, sensor suhu, dan sensor cahaya. Hasil dari perancangan perangkat keras pada sistem sirkulasi udara dan pencahayaan otomatis dapat dilihat pada Gambar 15 yang merupakan dari tampilan *prototype* rumah secara keseluruhan. Dalam *prototype* rumah tersebut terdapat dua buah kipas, satu buah motor DC yang menggerakkan tirai, satu buah lampu dan rangkaian Arduino serta mikrokontroler.



Gambar 15. Tampilan *Prototype* Rumah

## 5.3. Implementasi Perancangan Antarmuka

Hasil dari perancangan antarmuka pada sistem sirkulasi udara dan pencahayaan otomatis dapat dilihat pada Gambar 16. Aplikasi antarmuka ini dibuat dengan *visual basic*. Terdapat beberapa menu pada aplikasi antarmuka ini yaitu menu port koneksi, hentikan, memuat data, tampilkan waktu, simpan waktu lampu, simpan waktu tirai, simpan waktu kipas dan reset waktu.



Gambar 16. Tampilan Antarmuka

## 6. KESIMPULAN DAN SARAN

### 6.1 Kesimpulan

Berdasarkan proses perancangan sistem sirkulasi udara dan pencahayaan otomatis di dalam rumah dan telah dilakukan implementasi beserta pengujian, maka diperoleh kesimpulan antara lain:

1. Alat ini dapat bekerja untuk mengendalikan nyala lampu, kipas dan buka tutup tirai secara otomatis melalui aplikasi antarmuka.
2. Hasil pengujian terhadap sensor suhu LM35 dapat bekerja sesuai dengan fungsinya dengan persentase error di bawah 5 %.
3. Hasil pengujian terhadap sensor cahaya LDR dapat bekerja sesuai dengan fungsinya. Hasil perhitungan nilai ADC dan tampilan nilai ADC di LCD memiliki persentase error di bawah 10 %. Hasil pengujian terhadap sensor cahaya dapat disimpulkan bahwa semakin kecil nilai ADC maka intensitas cahaya semakin besar. Begitu juga sebaliknya semakin besar nilai ADC maka intensitas cahaya semakin kecil.
4. Terdapat kendala dalam distribusi arus listrik sehingga terkadang alat tidak stabil saat dioperasikan.

### 6.2 Saran

Berdasarkan penelitian yang telah dilakukan pada sistem sirkulasi udara dan pencahayaan otomatis di dalam rumah, maka diperoleh saran untuk penelitian lebih lanjut antara lain:

1. Perlu nya ditambahkan fitur untuk mengendalikan kinerja sistem kendali

dari jarak jauh seperti menggunakan *smartphone* android.

2. Diharapkan menggunakan sumber listrik yang lebih stabil agar dapat mengatasi kendala yang disebabkan arus listrik yang kurang stabil.

## DAFTAR PUSTAKA

- [1] Herjanto, E. (2009). *Manajemen Operasi Edisi Tiga*. Jakarta: Grasindo.
- [2] Chandra, B. (2005). *Pengantar Kesehatan Lingkungan*. Jakarta: Buku Kedokteran EGC
- [3] Kadir, A. (2013). *Panduan Praktis Mempelajari Aplikasi Mikrokontroler dan Pemogramannya Menggunakan Arduino*. Yogyakarta: Andi
- [4] Budiharto, W. (2008). *Panduan Praktikum Mikrokontroler AVR Atmega16*. Jakarta: Elex Media Komputindo.
- [5] Andrianto, H. (2013). *Pemrograman Mikrokontroler AVR Atmega16 Menggunakan Bahasa C (Code VisionAVR)*. Bandung: Informatika
- [6] Brindley, K. (2005). *Starting Electronics Construction*. Loughborough : Co-publications
- [7] Bishop, O. (2004). *Dasar-Dasar Elektronika*. Jakarta: Erlangga
- [8] Utami, Ema dan Sukrisno. (2005). *Konsep Dasar Pengolahan dan Pemrograman Database dengan SQL Server, Ms. Access, dan Ms. Visual Basic*. Yogyakarta: Andi