Alat Penyaji Sarapan Pagi Berbasis Mikrokontroler Dan Android

M. A. Imron[1], dan M. F. Wicaksono[2]
[1,2] Program Studi Sistem Komputer, Fakultas Teknik dan Ilmu Komputer, Universitas Komputer Indonesia, Bandung.
Email : imroonfive@email.unikom.ac.id

ABSTRAK

Sarapan pagi mempunyai peranan penting bagi setiap manusia. Sarapan merupakan aspek penting bagi kesehatan. Manfaat dari sarapan adalah dapat mempertahankan daya tahan tubuh dan meningkatkan kinerja otak. Saat ini banyak kendala untuk membuat sarapan pagi, yaitu kesibukan dan aktivitas yang padat dipagi hari yang membuat seseorang tidak dapat menyajikan sarapan. Berdasarkan permasalahan tersebut, maka perlu dibuat alat penyaji sarapan. Makanan yang dapat dibuat oleh alat adalah bubur dan roti kukus. Minuman yang dapat dibuat oleh alat adalah berbahan dasar kopi, susu dan gula. Hasil akhir dari penelitian ini diharapkan dapat membantu dalam pembuatan sarapan pagi, yang dapat dilakukan secara otomatis.

Kata kunci: Android, Arduino Mega, Penyaji Sarapan, Real-Time Clock, Servo.

ABSTRACT

Breakfast has an important role for every human being. Breakfast is an important aspect for health. The benefits of breakfast are able to maintain endurance and improve brain performance. Today many problems to make breakfast, which is busy and busy activity in the morning that makes people can not serve breakfast. Based on this, it is necessary to make a breakfast publisher. Foods that can be made by tools are porridge and steamed buns. Drinks that can be made by the tool is made from coffee, milk and sugar. The end result of this research is expected to assist in making breakfast, which can be done automatically.

Keywords: Android, Arduino Mega, Breakfast Serving, Real-Time Clock, Servo

I. PENDAHULUAN

Sarapan pagi merupakan kebutuhan manusia yang paling mendasar, tanpa sarapan pagi akan berakibat menurunkan konsentrasi belajar, tubuh menjadi lebih lemas, mudah lelah dan emosi menjadi tidak stabil. Saat ini dalam pembuatan sarapan pagi masih mempunyai banyak kendala, salah satunya yaitu kesibukan dan aktivitas yang padat dipagi hari yang membuat seseorang tidak dapat menyajikan sarapan pagi.

Permasalahan diatas mendorong penulis untuk membuat "Alat Penyaji Sarapan Pagi Berbasis Mikrokontroler ATMEGA 2560 Dan Android" yang mampu membuat makanan dan minuman tertentu. Dengan cara memilih menu makanan dan minuman yang dikontrol oleh aplikasi android melalui bluetooth. Sehingga memudahkan pengguna *(user)* dalam pembuatan sarapan pagi dengan sentuhan android.

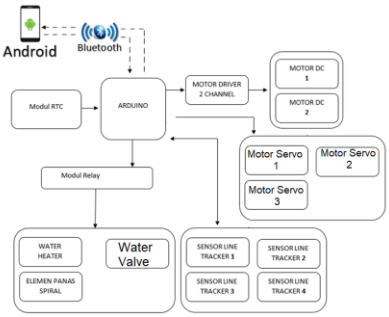
Diharapkan alat ini dapat membantu pekerjaan seseorang dalam penyediaan sarapan pagi. Sehingga mengurangi masalah seseorang untuk membuat sarapan pagi.

II. PERANCANGAN SISTEM

Perancangan yang dilakukan terdiri dari Diagram Blok, Perancangan Perangkat Keras, Perancangan Perangkat Lunak, dan Perancangan Mekanik.

2.1 Diagram Blok

Mengenai komponen-komponen apa saja yang digunakan dalam pembuatan alat penyaji sarapan pagi, baik dari segi peracangan alat, perangkat keras dan serta alasan mengenai tipe dan jenis komponen yang digunakan. Secara garis besar peracangan sistem kontrol dapat dilihat pada diagram blok pada gambar 2.1.



Gambar 2.1 Diagram blok Alat penyaji sarapan pagi.

2.2 Cara Kerja Alat

Cara kerja dari alat penyaji sarapan pagi adalah pertama user memilih menu makanan dan minuman yang ada pada aplikasi android. Pada menu makanan terdapat 3 menu yang mempunyai waktu proses yaitu nasi (60 menit), bubur (40 menit) dan roti kukus (20 menit). Kemudian dari ketiga menu tersebut masing-masing terdapat 3 pilihan waktu penyajian yang dapat dipilih. Waktu penyajian terdiri dari pukul 5, 6 dan 7. Setelah memilih menu makanan dan waktu yang telah ditentukan, Arduino akan menentukan kondisi dimana relay1 yang terhubung dengan rice cooker akan menyala ketika (Waktu RTC = Waktu Penyajian – Waktu Proses) dan relay1 akan mati ketika (Waktu RTC = Waktu Penyajian).

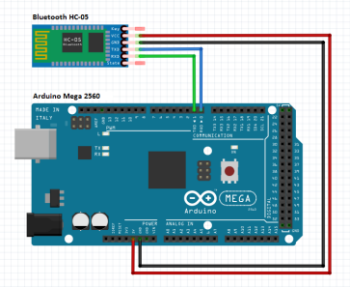
Pada bagian minuman terdapat 6 pilihan menu yang berbahan dasar gula, susu dan kopi. Menu yang tersedia adalah air mineral, kopi manis, kopi susu, kopi hitam, susu manis dan susu tawar. Terdapat 3 buah motor servo sebagai katup bahan minuman dan 1 selenoid valve sebagai katup air mineral. Kemudian gelas diletakan diatas papan putar yang digerakan oleh roda motor dc gearbox, dimana papan putar tersebut diberikan marker yang akan berhenti pada 4 posisi yang masing-masing posisi tersebut diberikan sensor line tracker. Papan putar akan berhenti ketika sensor yang dibaca ‘LOW’ dan akan menyala ketika sensor yang dibaca ‘HIGH’. Posisi pertama dari papan putar merupakan posisi awal gelas yaitu posisi air. Lalu posisi kedua adalah posisi gula, ketiga adalah posisi susu dan keempat adalah posisi kopi.

Ketika pengguna sedang melakukan pengiriman data menu minuman, sebagai contoh user memilih susu manis. Kemudian arduino akan mengkondisikan beberapa langkah untuk membuat minuman susu manis. Langkah pertama, papan putar berjalan menuju posisi 2, kemudian motor servo1 (gula) akan menuangkan gula. Langkah kedua, papan putar berjalan menuju posisi 3, kemudian motor servo2 (susu) akan menuangkan susu. Langkah terakhir, papan putar berjalan menuju posisi 1, kemudian solenoid valve akan menuangkan air kedalam gelas. Dan ketika gelas berada pada posisi awal, maka minuman siap disajikan.

2.3 Perancangan Perangkat Keras

Pada bagian ini akan dibahas perancangan perangkat keras terkait dengan alat penyaji sarapan pagi.

2.3.1 Rangkaian Bluetooth dengan Arduino Mega 2560



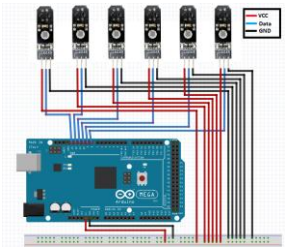
Gambar 2.2.1 Rangkaian antara Bluetooth dan Arduino Mega 2560

Berdasarkan gambar diatas yaitu *Rangkaian Bluetooth dengan Arduino Mega 2560.* Dibawah ini merupakan tabel penempatan pin yang berhubungan dengan gambar di atas:

Tabel 2.2.1 Penempatan Pin antara Bluetooth dan Arduino Mega 2560

Bluetooth HC-05	Arduino Mega 2560
Pin RXD	Pin 1 (TX0)
Pin TXD	Pin 0 (RX0)
VCC	VCC (5V)
GND	GND

2.3.2 Rangkaian modul Sensor TCRT5000 dengan Arduino Mega 2560



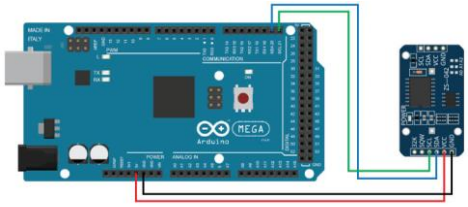
Gambar 2.2.2 Rangkaian antara Modul Sensor TCRT5000 dan Arduino Mega 2560

Berdasarkan gambar diatas yaitu *Rangkaian Modul TCRT5000 dengan Arduino Mega 2560.* Dibawah ini merupakan tabel penempatan pin yang berhubungan dengan gambar di atas:

Tabel 2.2.2 Penempatan Pin antara Bluetooth dan Arduino Mega 2560

Modul Sensor TCRT5000	Arduino Mega 2560
Input Sensor 1	Pin 7
Input Sensor 2	Pin 6
Input Sensor 3	Pin 5
Input Sensor 4	Pin 4
VCC (Sensor 1 ~ 4)	VCC (5V)
GND (Sensor 1 ~ 4)	GND

2.3.3 Rangkaian Modul RTC dengan Arduino Mega 2560



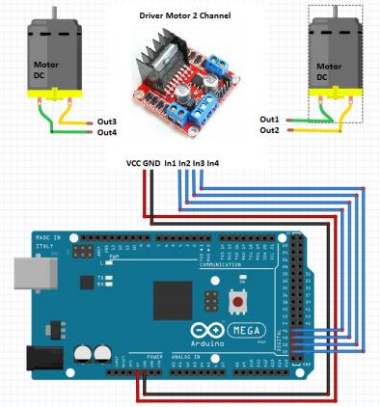
Gambar 2.2.2 Rangkaian antara Modul RTC dan Arduino Mega 2560

Berdasarkan gambar diatas yaitu *Rangkaian Modul RTC dengan Arduino Mega 2560.* Dibawah ini merupakan tabel penempatan pin yang berhubungan dengan gambar di atas:

Tabel 2.2.3 Penempatan Pin antara Modul RTC dan Arduino Mega 2560

Modul RTC	Arduino Mega 2560
Pin SDA	Pin 20 (SDA)
Pin SCL	Pin 21 (SCL)
VCC	VCC (5V)
GND	GND

2.3.4 Rangkaian Arduino Mega 2560 dengan Motor Driver 2 Channel



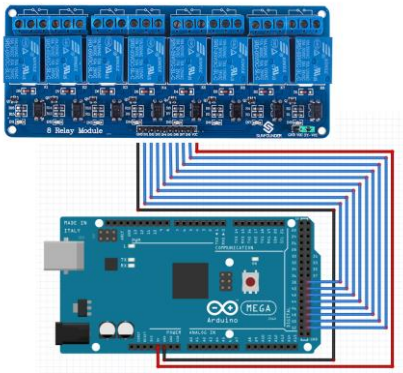
Gambar 2.2.2 Rangkaian antara Motor Driver dan Arduino Mega 2560

Berdasarkan gambar diatas yaitu *Rangkaian Arduino Mega 2560 dengan Motor Driver 2 Channel.* Dibawah ini merupakan tabel penempatan pin yang berhubungan dengan gambar di atas:

Tabel 2.2.4 Penempatan Pin antara Arduino Mega 2560 dan Motor Driver 2 Channel

Motor Driver 2 Channel	Arduino Mega 2560
Input 1	Pin 52
Input 2	Pin 50
Input 3	Pin 48
Input 4	Pin 46
VCC (5V)	VCC
GND	GND

2.3.5 Arduino Mega 2560 dengan Relay 8 Channel



Gambar 2.2.5 Rangkaian antara Arduino Mega 2560 dan Modul Relay 8 Channel

Berdasarkan gambar diatas yaitu *Rangkaian Arduino Mega 2560 dengan Modul Relay 8 Channel.* Dibawah ini merupakan tabel penempatan pin yang berhubungan dengan gambar di atas:

Tabel 2.2.5 Penempatan Pin antara Arduino Mega 2560 dan Modul Relay 8 Channel

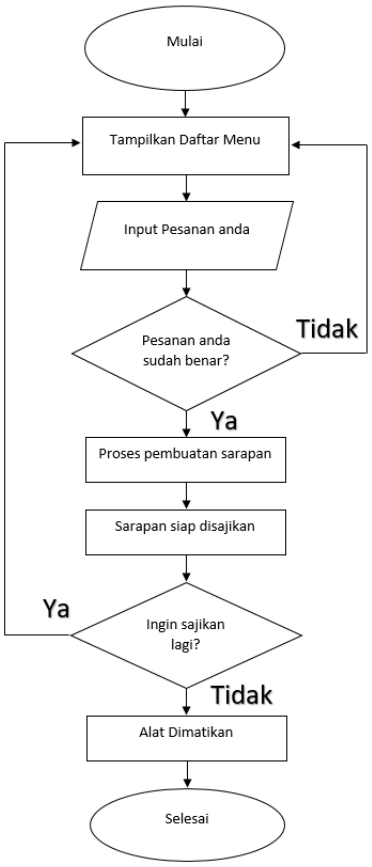
Relay 8 Channel	Arduino Mega 2560
Input 1	**Pin 39**
Input 2	**Pin 41**
Input 3	**Pin 43**
VCC	**VCC (5V)**
GND	**GND**

2.4 Perancangan Perangkat Lunak

Pada bagian ini akan dibahas perancangan perangkat lunak terkait dengan alat penyaji sarapan pagi.

2.4.1 Diagram Alir

Berikut ini adalah flowchart dan penjelasan dari alat penyaji sarapan pagi.



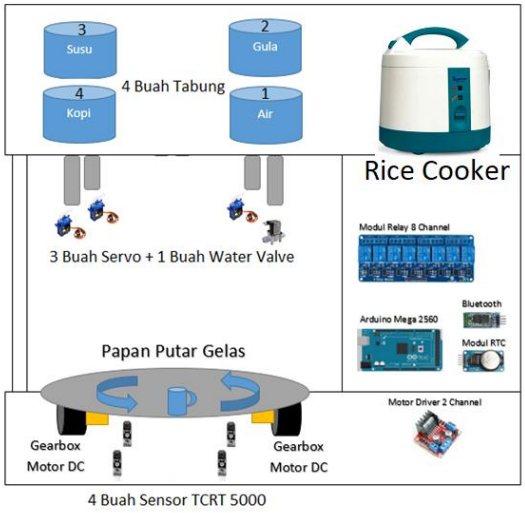
Gambar 2.4 Flowchart

Berikut ini adalah penjelasan dari flowchart diatas:

- ✓ Aplikasi android menampilkan daftar menu makanan dan minuman
- ✓ User menginput pesanan dan mengirimkan data berupa character ke Arduino.
- ✓ Arduino mengkondisikan dan menjalankan komponen sesuai dengan data yang dikirimkan untuk membuat makanan atau minuman.
- ✓ Ketika selesai, sarapan siap disajikan.
- ✓ Notifikasi alat ingin melakukan penyajian, jika ya akan kembali ke daftar menu.
- ✓ Jika tidak, alat dimatikan.

2.5 Perancangan Mekanik

Berikut ini merupakan tampilan perancangan mekanik dari alat penyaji sarapan pagi:



Gambar 2.4 Perancangan Mekanik Alat Penyaji Sarapan Pagi

III. HASIL PENGUJIAN

Pengujian perangkat Alat Penyaji Sarapan Pagi telah dilakukan pengujian alat dan pengiriman data sehingga dapat menghasilkan data yang tepat dari 5 percobaan.

3.1 Pengujian Relay 1 (Rice Cooker)

Pada pengujian ini dilakukan pengiriman data yang dikirimkan dari arduino terhadap relay1 yang akan mempengaruhi keluaran Rice Cooker. Data yang dikirimkan berupa variable 'LOW' dan 'HIGH'. Kemudian jika variable relay1 bernilai 'LOW' maka Rice Cooker akan mati dan ketika variable relay1 bernilai 'HIGH' maka Rice Cooker akan menyala. Pada Tabel 3.1 merupakan hasil pengujian pengiriman data dari Arduino terhadap relay1.

Tabel 3.1 Pengujian pengiriman data pada Relay1 (Rice Cooker)

NO	Kirim Variable Data	Hasil Pengujian (Relay 1) Ke-					Percentage Of Error (%)	Keterangan
		1	2	3	4	5		
1	LOW	**LOW**	**LOW**	**LOW**	**LOW**	**LOW**	**0**	**Berhasil**
2	HIGH	**HIGH**	**HIGH**	**HIGH**	**HIGH**	**HIGH**	**0**	**Berhasil**

Keterangan :
LOW = Arduino mengirimkan logika data '0'.
HIGH = Arduino mengirimkan logika data '1'.

Rumus :

$$\%Error = \frac{|Excact\ value - Approximate\ value|}{Excact\ value} \times 100$$

Error = 0 %

Dari hasil pengujian diatas pengiriman data Arduino ke relay1 dapat dinyatakan 100% berhasil dan dalam percobaan tersebut *percentage of error* adalah 0%. Maka relay1 dapat berfungsi dengan baik.

3.2 Pengujian Relay 2 (Water Heater)

Pada pengujian ini dilakukan pengiriman data yang dikirimkan dari arduino terhadap relay2 yang akan mempengaruhi keluaran Water Heater. Data yang dikirimkan berupa variable 'LOW' dan 'HIGH'. Kemudian jika variable relay2 bernilai 'LOW' maka Water Heater akan mati dan ketika variable relay2 bernilai 'HIGH' maka Water Heater akan menyala. Pada Tabel 3.2 merupakan hasil pengujian pengiriman data dari Arduino terhadap relay2.

Tabel 3.2 Pengujian pengiriman data pada Relay2 (Water Heater)

NO	Kirim Variable Data	Hasil Pengujian (Relay 2) Ke-					Percentage Of Error (%)	Keterangan
		1	2	3	4	5		
1	LOW	**LOW**	**LOW**	**LOW**	**LOW**	**LOW**	**0**	**Berhasil**
2	HIGH	**HIGH**	**HIGH**	**HIGH**	**HIGH**	**HIGH**	**0**	**Berhasil**

Keterangan :
LOW = Arduino mengirimkan logika data '0'.
HIGH = Arduino mengirimkan logika data '1'.
Rumus :

$$\%Error = \frac{|Excact\ value - Approximate\ value|}{Excact\ value} \times 100$$

Error = 0 %

Dari hasil pengujian diatas pengiriman data Arduino ke relay2 dapat dinyatakan 100% berhasil dan dalam percobaan tersebut *percentage of error* adalah 0%. Maka relay2 dapat berfungsi dengan baik.

3.3 Pengujian Relay 3 (Selenoid Valve)

Pada pengujian ini dilakukan pengiriman data yang dikirimkan dari arduino terhadap relay3 yang akan mempengaruhi keluaran Selenoid Valve. Data yang dikirimkan berupa variable 'LOW' dan 'HIGH'. Kemudian jika variable relay3 bernilai 'LOW' maka Selenoid Valve akan mati dan ketika variable relay3 bernilai 'HIGH' maka Selenoid Valve akan menyala. Pada Tabel 3.3 merupakan hasil pengujian pengiriman data dari Arduino terhadap relay3.

Tabel 3.3 Pengujian pengiriman data pada Relay3 (Selenoid Valve)

NO	Kirim Variable Data	Hasil Pengujian (Relay 3) Ke-					Percentage Of Error (%)	Keterangan
		1	2	3	4	5		
1	LOW	**LOW**	**LOW**	**LOW**	**LOW**	**LOW**	**0**	**Berhasil**
2	HIGH	**HIGH**	**HIGH**	**HIGH**	**HIGH**	**HIGH**	**0**	**Berhasil**

Keterangan :
LOW = Arduino mengirimkan logika data '0'.
HIGH = Arduino mengirimkan logika data '1'.
Rumus :

$$\%Error = \frac{|Excact\ value - Approximate\ value|}{Excact\ value} \times 100$$

Error = 0 %

Dari hasil pengujian diatas pengiriman data Arduino ke relay3 dapat dinyatakan 100% berhasil dan dalam percobaan tersebut *percentage of error* adalah 0%. Maka relay3 dapat berfungsi sebagai mana semestinya.

3.4 Pengujian Bluetooth

Pada pengujian ini dilakukan pengiriman data dari bluetooth ke arduino. Pada Tabel 3.4 merupakan hasil pengujian pengiriman data bluetooth.

Tabel 3.4 Pengujian pengiriman data pada Bluetooth

NO	Kirim Data	Data yang diterima					Percentage Of Error (%)	Keterangan
		1	2	3	4	5		
1	1	**1**	**1**	**1**	**1**	**1**	**0**	**Berhasil**
2	2	**2**	**2**	**2**	**2**	**2**	**0**	**Berhasil**
3	3	**3**	**3**	**3**	**3**	**3**	**0**	**Berhasil**
4	A	**A**	**A**	**A**	**A**	**A**	**0**	**Berhasil**
5	B	**B**	**B**	**B**	**B**	**B**	**0**	**Berhasil**

Keterangan :
Kirim Data = Bluetooth.
Data yang diterima = Arduino.
Data = Char.
Rumus :

$$\%Error = \frac{|Excact\ value - Approximate\ value|}{Excact\ value} \times 100$$

Error = 0 %

Dari hasil pengujian diatas pengiriman data bluetooth ke arduino dapat dinyatakan 100% berhasil dan dalam percobaan tersebut *percentage of error*

adalah 0%. Maka bluetooth dapat berfungsi dengan baik.

3.5 Pengujian Servo 1 (Gula)

Pada pengujian ini dilakukan kondisi servo1 dengan sudut 0° dan 60°. Dimana pada sudut 0° katup gula akan tertutup dan pada sudut 60° katup gula akan terbuka. Pada Tabel 3.5 merupakan hasil pengujian sudut servo1.

Tabel 3.5 Pengujian sudut Servo 1 (Gula)

Percobaan Ke-	Hasil Pengujian Servo1 Sudut-		Percentage Of Error (%)	Keterangan
	0°	60°		
1	✓	✓	**0**	**Berhasil**
2	✓	✓	**0**	**Berhasil**
3	✓	✓	**0**	**Berhasil**
4	✓	✓	**0**	**Berhasil**
5	✓	✓	**0**	**Berhasil**

Keterangan :
✓ = Sesuai.
✗ = Tidak Sesuai.
Rumus :

$$\%Error = \frac{|Excact\ value - Approximate\ value|}{Excact\ value} \times 100$$

Error = 0 %

Dari hasil pengujian diatas pengujian servo1 dapat dinyatakan 100% sesuai dan dalam percobaan tersebut *percentage of error* adalah 0%. Maka servo1 dapat berfungsi dengan baik.

3.6 Pengujian Servo 2 (Susu)

Pada pengujian ini dilakukan kondisi servo2 dengan sudut 0° dan 60°. Dimana pada sudut 0° katup susu akan tertutup dan pada sudut 60° katup susu akan terbuka. Pada Tabel 3.6 merupakan hasil pengujian sudut servo2.

Tabel 3.6 Pengujian sudut Servo 2 (Susu)

Percobaan Ke-	Hasil Pengujian Servo2 Sudut-		Percentage Of Error (%)	Keterangan
	0°	60°		
1	✓	✓	**0**	**Berhasil**
2	✓	✓	**0**	**Berhasil**
3	✓	✓	**0**	**Berhasil**
4	✓	✓	**0**	**Berhasil**
5	✓	✓	**0**	**Berhasil**

Keterangan :
✓ = Sesuai.
✗ = Tidak Sesuai.
Rumus :

$$\%Error = \frac{|Excact\ value - Approximate\ value|}{Excact\ value} \times 100$$

Error = 0 %

Dari hasil pengujian diatas pengujian servo2 dapat dinyatakan 100% sesuai dan dalam percobaan tersebut *percentage of error* adalah 0%. Maka servo2 dapat berfungsi dengan baik.

3.7 Pengujian Servo 3 (Kopi)

Pada pengujian ini dilakukan kondisi servo3 dengan sudut 0° dan 60°. Dimana pada sudut 0° katup kopi akan tertutup dan pada sudut 60° katup kopi akan terbuka. Pada Tabel 3.7 merupakan hasil pengujian sudut servo3.

Tabel 3.7 Pengujian sudut Servo 3 (Kopi)

Percobaan Ke-	Hasil Pengujian Servo3 Sudut-		Percentage Of Error (%)	Keterangan
	0°	60°		
1	✓	✓	**0**	**Berhasil**
2	✓	✓	**0**	**Berhasil**
3	✓	✓	**0**	**Berhasil**
4	✓	✓	**0**	**Berhasil**
5	✓	✓	**0**	**Berhasil**

Keterangan :
✓ = Sesuai.
✗ = Tidak Sesuai.
Rumus :

$$\%Error = \frac{|Excact\ value - Approximate\ value|}{Excact\ value} \times 100$$

Error = 0 %

Dari hasil pengujian diatas pengujian servo3 dapat dinyatakan 100% sesuai dan dalam percobaan tersebut *percentage of error* adalah 0%. Maka servo3 dapat berfungsi dengan baik.

3.8 Pengujian Sensor Line Tracker (TCRT5000)

Pada pengujian ini dilakukan pengukuran jarak baca sensor 5mm sampai 25mm. Pada Tabel 3.8 merupakan hasil pengujian jarak sensor.

Tabel 3.8 Pengujian jarak Sensor TCRT5000

NO	Jarak Uji (mm)	Hasil Pengujian Sensor-				Percentage Of Error (%)	Keterangan
		1	2	3	4		
1	5	✓	✓	✓	✓	0	Berhasil
2	10	✓	✓	✓	✓	0	Berhasil
3	15	✓	✓	✓	✓	0	Berhasil
4	20	✓	✓	✓	✓	0	Berhasil
5	25	✓	✓	✓	✓	0	Berhasil

Keterangan :
✓ = Kebaca.
✗ = Tidak Kebaca.
Rumus :

$$\%Error = \frac{|Excact\ value - Approximate\ value|}{Excact\ value} \times 100$$

Error = 0 %

Dari hasil pengujian diatas pengujian sensor dapat dinyatakan 100% sesuai dan dalam percobaan tersebut *percentage of error* adalah 0%. Maka sensor dapat berfungsi dengan baik.

IV. KESIMPULAN DAN SARAN

4.1 KESIMPULAN

Berdasarkan perancangan sistem hingga pengujian yang telah dilakukan serta hasil pengujian dan analisa perangkat keras yang telah dibangun dapat disimpulkan bahwa:

Alat yang di buat telah berhasil membuat minuman berbahan dasar (Kopi, Susu dan Gula) dengan cara melakukan pengujian sensor tcrt5000, motor servo, relay2 dan relay3, pengujian pengukuran telah berhasil dan berjalan sesuai prosedur sehingga masing-masing pengujian mendapatkan *percentage of error* sebesar 0%.

Alat yang di buat telah berhasil membuat makanan (Nasi, Bubur dan Roti Kukus), dengan cara melakukan pengujian terhadap relay1 sebagai saklar rice cooker dan telah berhasil dan berjalan sesuai prosedur sehungga mendapatkan *percentage of error* sebesar 0%.

Android sebagai user interface mampu berfungsi sebagai kontrol, dengan cara melakukan pengujian pengiriman data Bluetooth dan mendapatkan *percentage of error* sebesar 0%.

4.2 SARAN

Saran yang dianjurkan penulis agar dapat menjadikan masukan dalam pengembangan alat ini adalah:

Untuk pengembangan selanjutnya dalam pembuatan minuman, alat ini dapat menggunakan conveyor untuk menggantikan papan putar gelas dalam menentukan posisi gelas.

Untuk pengembangan selanjutnya dalam pembuatan makanan, alat ini dapat ditambahkan mekanik supaya makanan dapat memasukan bahan.

Untuk pengembangan selanjutnya dalam interface android, tampilan android dapat diperbaharui lagi dan menambah fitur notifikasi didalamnya.

DAFTAR PUSTAKA

[1] M. F. Wicaksono. *Mudah Belajar Mikrokontroler*, Bandung: Informatika, 2017.

[2] R. Sigit, *Robotika, Sensor dan Aktuator*,Graha Ilmu, 2014.

[3] Modul Sensor Line Tracker, *Datasheet Line Tracker.* Available: https://static.shop033.com/resources/35/1000245/Other/Instructions/Viper/Line-Follower-Datasheet.pdf. Diakses 4 Januari 2018.

[4] Motor Driver. *Push-Pull Four Channel Driver With Diodes.* Available: https://www.arduino.cc/documents/datasheets/H-bridge_motor_driver.PDF. Diakses 4 Januari 2018.

[5] Modul Bluetooth HC-05, *HC-05 Bluetooth to Serial Port Module.* Available: *www.thaieasyelec.com/downloads/EFDV390/EFDV390_**Datasheet**.pdf.* Diakses 4 Januari 2018.

[6] Realtime Clock. *Real-Time Clock (RTC) (Rev.R) – Texas Instruments.* Available: *www.ti.com/lit/ds/symlink/bq32000.pdf.* Diakses 4 Januari 2018.

[7] Selenoid Valve, *Selenoid Valve Datasheet.* Available: http://category.alldatasheet.com/index.jsp?sSearchword=SOLENOID%20VALVE. Diakses 4 Januari 2018.

[8] Water Heater. *Datasheet(CR280SST4) – Hubbell Water Heaters.* Available: www.hubbellheaters.com/downloads/datasheets/cr280sst4.pdf. Diakses 4 Januari 2018.

[9] Relay. *Relay Datasheet.* Available: https://www.hobbytronics.co.uk/datasheets/songle-12v-relay.pdf. Diakses 4 Januari 2018.

[10] Motor DC dan Servo. *Servo Motor SG90 Datasheet.* Available: www.ee.ic.ac.uk/pcheung/teaching/DE1_EE/stores/sg90_**datasheet**.pdf. Diakses 4 Januari 2018.