

TUGAS AKHIR MUH DEVANGGA HAMDAN MAULANA (19120042

by Nia Nur

Submission date: 04-Oct-2023 12:03AM (UTC-0700)

Submission ID: 2185242812

File name: TUGAS_AKHIR_MUH_DEVANGGA_HAMDAN_MAU_LANA_19120042.pdf (1.76M)

Word count: 10903

Character count: 68741

¹
TUGAS AKHIR

**RANCANG BANGUN SISTEM SORTIR UANG PADA
KOTAK PENYIMPANAN UANG (MONEYBOX
PLUS) BERBASIS MIKROKONTROLER ARDUINO**



MUH. DEWANGGA HAMDAN MAULANA
NPM :19120042

¹
DOSEN PEMBIMBING
Firman Hadi Sukma Pratama, ST., MT.

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS WIJAYA KUSUMA SURABAYA
SURABAYA
2023**

Tugas Akhir disusun untuk memenuhi salah satu
syarat memperoleh gelar
Sarjana Komputer (S.Kom)

di
Universitas Wijaya Kusuma Surabaya

Oleh :

MUH. DEVANGGA HAMDAN MAULANA

NPM : 19120042

Hari/Tanggal Sidang : 12 Juli 2023

Pembimbing

Firman Hadi Sukma Pratama, ST., MT.

NIK : 15734-ET

Ketua Program Studi
Informatika

Nonot Wisnu Karyanto, ST., M.Kom.

NIK : 11563-ET

Dekan
Fakultas Teknik

Johan Paing Heru Waskito, ST., MT.

NIK : 196903102005011002

1
LEMBAR PENGESAHAN
TUGAS AKHIR

Judul : Rancang Bangun Sistem Sortir Uang Pada Kotak Penyimpanan Uang (*MoneyBox Plus*) Berbasis Mikrokontroler Arduino

Oleh : Muh. Devangga Hamdan Maulana
NPM : 19120042

Telah diuji Pada :

Hari : Rabu

Tanggal : 12 Juli 2023

Tempat : Ruangan F302, Univ. Wijaya Kusuma Surabaya

Disetujui:

Dosen Penguji :

Dosen Pembimbing :

1. Nonot Wisnu Karyanto,
ST., M.Kom.
NIK : 11563-ET

1. Firman Hadi Sukma
Pratama, ST., MT.
NIK : 15734-ET

2. Lestari Retnawati,
S.Kom., M.MT.
NIK : 16762A-ET

RANCANG BANGUN SISTEM SORTIR UANG PADA KOTAK PENYIMPANAN UANG (*MONEYBOX PLUS*) BERBASIS MIKROKONTROLER ARDUINO

1 Muh. Devangga Hamdan Maulana
Program Studi Informatika Fakultas Teknik
Universitas Wijaya Kusuma Surabaya
devangga19@gmail.com

ABSTRAK

Meski sudah banyak media penyimpanan uang yang praktis, masih ada beberapa orang lebih memilih menyimpan uangnya secara pribadi, yang dikarenakan tidak semua Bank memiliki banyak cabang atau mesin setor tunai yang banyak, serta adanya ketidakseimbangan dalam jarak tempuh menuju alat setor tunai yang jauh. Perancangan alat *MoneyBox Plus* dengan sistem sortir uang yang bisa memisahkan uang kertas dengan nominal Rp 50.000,00 dan Rp 100.000,00. Ini adalah penerapan mata *Internet Of Things* dalam program studi Informatika, yang dikembangkan dengan menggunakan metode *prototype* yang dimana *MoneyBox Plus* ini menggunakan beberapa komponen seperti *Motor Servo* dan mikrokontroler arduino. Hasil yang dituju dalam penelitian ini adalah bisa mensortir uang nominal pecahan uang Rp.50.000 dan Rp.100.000 pada tempat yang telah disediakan didalam *MoneyBox Plus*.

Kata Kunci : Sistem Sortir Uang, *Internet Of Things*, *Motor Servo*, Mikrokontroler Arduino, *MoneyBox Plus*

KATA PENGANTAR

Alhamdulillah, puji dan syukur kami ucapkan kepada ALLAH SWT yang telah memberikan kesehatan dan kesempatan sehingga kami mampu menyelesaikan Tugas Akhir. Tugas Akhir ini berjudul “Rancang Bangun Sistem Sortir Uang Pada Kotak Penyimpanan Uang (*MoneyBox Plus*) Berbasis Mikrokontroler Arduino”.

Tugas Akhir ini dilaksanakan dengan baik di kampus Universitas Wijaya Kusuma Surabaya. Laporan Tugas Akhir ini merupakan tugas yang harus diselesaikan oleh mahasiswa jurusan teknik informatika program S1 di Universitas Wijaya Kusuma Surabaya. Tujuan utama dari pembuatan Tugas Akhir ini adalah untuk memantapkan teori dan praktek yang telah dipelajari di kampus dan dapat diselesaikan dengan serta diaplikasikan di lapangan. Dalam proses pembuatan laporan ini tak lupa kami menghaturkan terima kasih kepada :

1. Bapak Johan Paing H.W, ST, MT sebagai Dekan Fakultas Teknik
2. Bapak Nonot Wisnu Karyanto, ST., M.Kom selaku ketua prodi Informatika.
3. Bapak Firman Hadi Sukma Pratama, ST., MT. selaku dosen pembimbing.
4. Segenap Dosen Program Studi Informatika Universitas Wijaya Kusuma Surabaya yang telah memberikan ilmunya kepada penulis selama masa perkuliahan.
5. Orang tua kami yang telah banyak memberikan dorongan semangat dari awal hingga selesainya laporan ini.

6. Teman-teman yang telah memberikan dorongan moral dan material serta informasi.

Penulis menyadari bahwa laporan ini masih jauh dari kesempurnaan dengan segala kekurangannya. Untuk itu penulis mengharapkan adanya kritik dan saran dari semua pihak demi kesempurnaan dari laporan tugas akhir ini. Akhir kata penulis berharap, semoga laporan ini dapat bermanfaat bagi rekan-rekan mahasiswa-mahasiswi dan pembaca sekaligus demi menambah pengetahuan tentang Tugas Akhir.

Surabaya, 3 Oktober 2022

Muh. Devangga Hamdan M.

DAFTAR ISI

TUGAS AKHIR	1
LEMBAR PENGESAHAN.....	i
ABSTRAK	ii
KATA PENGANTAR.....	iii
DAFTAR ISI	v
DAFTAR TABEL	viii
DAFTAR GAMBAR	ix
BAB 1	1
1.1 Latar belakang	1
1.2 Rumusan Masalah.....	1
1.3 Batasan Masalah	2
1.4 Tujuan	2
1.5 Manfaat	3
1.6 Sistematika Penulisan	3
BAB II	6
2.1 Penelitian Terdahulu	6
2.2 Teori Penunjang Penelitian	10
2.2.1 Sistem Sortir Uang Pada <i>MoneyBox Plus</i>	10
2.2.2 <i>Prototype</i>	11
2.2.3 <i>Motor Servo</i>	12
2.2.3.1 Penggunaan <i>Motor Servo</i> pada Arduino	13

2.2.4 Mikrokontroler Arduino	14
2.2.4.1 Arduino Mega 2560	16
2.2.4.2 Arduino IDE.....	18
2.2.5 <i>Flowchart</i>	20
2.2.6 TFT LCD <i>DISPLAY</i>	22
2.2.7 Pemanfaatan LCD pada Mikrokontroler Arduino ..	23
2.2.8 Sensor.....	25
2.2.9 <i>Internet Of Things</i>	27
2.2.10 <i>Modeling Quick Desain</i>	28
2.2.11 Diagram Alur	28
2.2.12 Uang	30
2.2.12.1 Jenis Uang Terhadap Pembuatannya	30
2.2.13 EPROME	31
1 BAB III	33
3.1 Tahapan Penelitian	33
3.1.1 Alur Penelitian	33
3.1.2 Identifikasi Masalah	34
3.1.3 Analisis Kebutuhan	34
3.1.3.1 Analisis Kebutuhan Fungsional	35
3.1.3.2 Analisis Kebutuhan Non Fungsional	35
3.1.4 <i>Flowchart</i>	37
3.1.5 <i>Modeling Quick Design</i>	38
3.1.6 <i>User Interface Design</i>	38

3.1.6.1 Design UI Menu Utama	39
3.1.6.2 Desain UI Menu Cek Jumlah	39
3.1.7 <i>Sketching Proyototype</i>	42
3.1.8 Implementasi	42
3.1.9 Pengujian.....	43
3.1.10 Pembuatan Laporan	43
BAB IV	44
4.1 Pengujian Sistem	44
4.1.1 Menu utama	44
4.1.2 Menu Cek Jumlah	45
4.1.3 Penerapan Cara Kerja Sistem Sortir	47
4.2 Hasil Penelitian	49
4.2.1 Layout Kode Menu Utama.....	49
4.2.2 Layout Kode Menu Cek Jumlah	51
4.2.3 Layout Kode <i>Motor Servo</i>	57
4.3 Hasil Implementasi	66
4.3.1 Penggunaan Layar LCD.....	67
4.3.2 Penggunaan <i>Motor Servo</i>	67
BAB V	70
5.1 Kesimpulan	70
5.2 Saran	70
DAFTAR PUSTAKA	72
LAMPIRAN	76

DAFTAR TABEL

Tabel 2. 1 Penelitian Terdahulu	6
Tabel 2.2 Spesifikasi Arduino Mega 2560.....	17

DAFTAR GAMBAR

Gambar 2. 1 <i>Motor Servo</i>	13
Gambar 2. 2 <i>Motor Servo</i> pada Arduino sebelum ke-trigger ..	14
Gambar 2. 3 <i>Motor Servo</i> pada Arduino setelah ke-trigger	14
Gambar 2. 4 Mikrokontroler Arduino	15
Gambar 2. 5 Arduino Mega 2560.....	17
Gambar 2. 6 Software Arduino IDE.....	19
Gambar 2. 7 Simbol <i>Flowchart</i> 1	21
Gambar 2. 8 Simbol <i>Flowchart</i> 2.....	21
Gambar 2. 9 TFT LCD <i>Display Shield</i>	23
Gambar 2. 10 Komponen Modul LCD.....	25
Gambar 2. 11 Simbol Diagram Alur	29
Gambar 3. 1 Diagram Alur.....	33
Gambar 3. 2 <i>Flowchart</i>	37
Gambar 3. 3 <i>Modeling Quick Design</i>	38
Gambar 3. 4 Rancangan UI utama	39
Gambar 3. 5 Rancangan UI Masukan PIN	40
Gambar 3. 6 Rancangan UI Cek Jumlah Uang	41
Gambar 3. 7 <i>Sketching Prototype</i>	42
Gambar 4. 1 Menu Utama	44
Gambar 4. 2 UI Menu Cek Jumlah	45
Gambar 4. 3 UI menampilkan jumlah uang	46
Gambar 4. 4 Akses Cek Jumlah Ditolak	47
Gambar 4. 5 <i>Source Code</i> Menu Utama	49
Gambar 4. 6 <i>Source Code</i> Menu Cek Jumlah Fungsi 1	51
Gambar 4. 7 <i>Source Code</i> Menu Cek Jumlah Fungsi 2	52
Gambar 4. 8 <i>Source Code</i> Menu Cek Jumlah Fungsi 3	53
Gambar 4. 9 <i>Source Code</i> Menu Cek Jumlah Fungsi 4	54
Gambar 4. 10 <i>Source Code</i> Menu Cek Jumlah Fungsi 5	55

Gambar 4. 11 <i>Source Code</i> Menu Cek Jumlah fungsi 6	56
Gambar 4. 12 <i>Source Code Motor Servo</i> fungsi 1	57
Gambar 4. 13 <i>Source Code Motor Servo</i> fungsi 2	58
Gambar 4. 14 <i>Source Code Motor Servo</i> fungsi 3	59
Gambar 4. 15 <i>Source Code Motor Servo</i> fungsi 4	61
Gambar 4. 16 <i>Source Code Motor Servo</i> fungsi 5	63
Gambar 4. 17 <i>Source Code Motor Servo</i> fungsi 6	65
Gambar 4. 18 Penggunaan Layar LCD	67
Gambar 4. 19 Kotak pecahan uang RP. 100.000 terbuka	68
Gambar 4. 20 Kotak pecahan uang RP. 50.000 terbuka	69

BAB 1

PENDAHULUAN

1.1 Latar belakang

Pada saat ini alat setor tunai terhitung sedikit daripada alat pengambil uang yang disediakan oleh Bank, dan kurangnya minat ketika hanya menyetorkan uang yang nominalnya sedikit dikarenakan tidak seimbang dengan jarak yang harus ditempuh menuju tempat yang memiliki mesin atau setor tunai sehingga lebih memilih untuk menyimpan uangnya secara pribadi dirumah.

Dari permasalahan tersebut maka dibuatnya alat yang bisa membantu dalam penyimpanan pribadi, alat ini dinamai *MoneyBox Plus* dan alat ini menggunakan *mikrokontroler Arduino* yang berfungsi untuk memproses *output* sesuai dengan yang di-*input*-kan.

Pembuatan alat *MoneyBox Plus* dan alat ini sangat memerlukan sistem sortir uang pada kotak penyimpanan uang (*Money Box Plus*) berdasarkan inputan uang yang masuk agar dapat memisahkan uang pecahan Rp 50.000,00 dan Rp 100.000,00 di dalam *MoneyBox Plus* dan agar mempermudah untuk pengambilan uang pada *MoneyBox Plus*.

1.2 Rumusan Masalah

1. Bagaimana cara kerja sistem sortir uang pada kotak penyimpanan uang (*Money Box Plus*)
2. Bagaimana cara kerja sistem sortir uang pada kotak penyimpanan uang (*Money Box Plus*) dapat menghitung keseluruhan uang yang tersimpan di *Money Box Plus*.

3. Bagaimana cara agar alat *MoneyBox Plus* dapat menampilkan jumlah keseluruhan uang di LCD.

1.3 Batasan Masalah

Untuk membatasi supaya sistem sesuai dengan yang diharapkan maka disusunlah batasan dari masalah sebagai berikut:

1. Sistem ini hanya dapat mensortir pecahan uang Rp.50.000,00 dan Rp.100.000,00
2. Sistem ini tidak dapat mensortir pecahan uang Rp. 50.000,00 dan Rp. 100.000,00 dalam 1 waktu yang bersamaan.
3. Sistem ini tidak dapat menghitung jumlah uang yang sudah keluar

1.4 Tujuan

1 Dari rumusan masalah yang telah didapat, maka tujuan dari Rancang Bangun Sistem Sortir Uang Pada Kotak Penyimpanan Uang (*MoneyBox Plus*) Berbasis Mikrokontroler Arduino ini sebagai berikut:

1. Merancang dan membangun sistem yang dapat mensortir Uang pecahan yang masuk pada *Money Box Plus*.
2. Merancang dan membangun sistem yang dapat memudahkan untuk pengambilan uang pada *Money Box Plus*.

1.5 Manfaat

Dari rumusan masalah yang telah didapat, maka dapat diambil tujuan dari pembuatan sistem sortir penyimpanan uang pada kotak *MoneyBox Plus* sebagai berikut:

1. Bagi pengguna *MoneyBox Plus* bisa menyimpan uang dan dapat memisahkan uang pecahan Rp 50.000,00 dan Rp 100.000,00 dengan adanya sistem sortir pada *MoneyBox Plus*.
2. Bagi penulis, sebagai implementasi atau penerapan ilmu dan teori yang telah didapat selama perkuliahan nyata.
3. Bagi pembaca sebagai referensi atau bacaan mengenai penelitian rancang bangun sistem penyimpanan uang pribadi.

2

1.6 Sistematika Penulisan

Sistematika penulisan dalam penelitian ini dibagi menjadi 3 bagian sebagai berikut :

BAB 1 PENDAHULUAN

Pada bab pendahuluan menjelaskan garis besar dari penelitian ini, yaitu tentang uraian latar belakang yang berisi mengenai apa itu sensor dan keresahan yang dialami generasi muda, kemudian rumusan masalah yang membahas mengenai bagaimana merancang dan membangun sistem, batasan masalah mengenai hal hal yang membatasi dalam hal pembuatan maupun penggunaan sistem, tujuan penelitian berisi tentang tujuan pembuatan sistem, serta manfaat penelitian.

2

BAB 2 LANDASAN TEORI

Bab ini berisi tentang teori – teori yang relevan berdasarkan hasil temuan atau hasil penelitian oleh orang – orang yang sebelumnya masih berhubungan dengan penelitian yang dilakukan. Teori teori yang dibahas mengenai rancang bangun sistem pemanfaatan sesnsor maupun arduino.

2

BAB 3 METODE PENELITIAN

Bab ini berisi penjelasan tentang metode yang digunakan dalam mengumpulkan dan menganalisa data serta langka-langka yang akan dilakukan pada penelitian ini. Adapun metode yang akan digunakan yaitu metode *prototype* yang terdiri dari beberapa tahapan dalam penggunaanya seperti identifikasi masalah, analisa kebutuhan, desain, pengkodean atau implementasi dan pengujian.

BAB 4 HASIL DAN PEMBAHASAN

Bab ini akan membahas hasil penelitian yang diperoleh dari penggunaan metode *prototype* yang telah dijelaskan pada Bab 3. Hasil penelitian akan dianalisis dan dibahas untuk memberikan pemahaman yang lebih mendalam tentang topik penelitian yang dilakukan.

BAB 5 PENUTUP

Bab ini merupakan bagian penutup yang berisi kesimpulan dan saran dari penelitian yang telah dilakukan. Penutup ini akan merangkum temuan-temuan penting yang telah dijelaskan dalam penelitian serta memberikan rekomendasi untuk pengembangan lebih lanjut.

2

DAFTAR PUSTAKA

Merupakan daftar referensi atau rujukan dasar-dasar teori yang digunakan peneliti untuk menyelesaikan penelitian.

BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Tabel 2. 1 Penelitian Terdahulu

Judul	Tujuan	Kelebihan	Tahun
Penyortir Uang Koin Berbasis Arduino Uno.	Penelitian ini dilakukan agar dapat membuat system penyortiran uang koin serta menghitung jumlah uang koin yang sudah di sortir.	Hasil dari percobaan ini menunjukkan bahwa penyortir koin dapat bekerja dengan baik.	2018
Rancang Bangun Alat Sortir Jeruk Nipis Berbasis Mikrokontroler	Berdasarkan rumusan masalah diatas maka adapun tujuan dari penelitian ini adalah merancang dan membangun suatu alat yang berfungsi untuk melakukan penyortiran jeruk nipis	Metode penelitian yang digunakan adalah metode eksperimental, dengan melakukan eksperimen terhadap variabel-variabel kontrol (<i>input</i>) untuk	2019

6

	secara otomatis menggunakan mikrokontroler arduino uno, Sehingga dapat membantu petani dalam melakukan penyortiran.	menganalisis output yang dihasilkan.	
Kotak Penyimpanan Uang Berbasis Mikrokontroler Arduino Uno.	Bertujuan mengembangkan teknologi sensor berbasis mikrokontroler dan membantu orang-orang yang masih ingin menyimpan uang di kotak penyimpanan uang.	Alat ini dapat menyimpan uang seperti halnya menabung sehingga tidak mudah diambil begitu saja.	2018
Sistem Sortir Barang Otomatis Berbasis Arduino dengan Sensor Warna dan Monitoring via Android	Penelitian ini bertujuan untuk meningkatkan kinerja produksi suatu perusahaan dan untuk sistem	Sensor mampu untuk mendeteksi perbedaan warna benda, sehingga prosesor dapat mengambil	2020

	pendukung produksinya.	keputusan, apakah benda tersebut akan dikumpulkan bersama benda lain yang memiliki kategori warna yang sama.	
Rancang Bangun Sistem Sortir Berdasarkan Warna Permen	Tujuan dari penelitian ini ialah smengetahui dan mengukur sejauh mana efektifitas dari perancangan dalam hal penyortiran permen serta membahas kinerja alat dan komponen pada sistem yang terlibat.	Alat ini dapat mensortir permen berdasarkan warna permen dengan baik	2019
Perancangan Sistem Sortir Buah Kopi Berdasarkan	Penelitian ini menawarkan sistem otomasi dapat membantu	Alat ini dapat mensortir otomatis yang dapat	2018

Warna Dengan Teknik Citra Digital Berbasis Mikrokontroler Atmega 328p	para petani dalam proses penyortiran sehingga dapat mengklasifikasi buah kopi sesuai dengan tingkat kematangan.	membantu para petani untuk mensortir buah kopi yang sudah matang dengan akurat dan efektif	
Rancang Bangun Media Pembelajaran Penyortiran Benda Berbasis Mikrokontroler	Tujuan penelitian ini adalah sebagai media pembelajaran penyortiran benda berbasis mikrokontroler dengan menggunakan konveyor merupakan alat sortir sekaligus pemindah barang yang biasanya digunakan dalam dunia perindustrian.	Dengan adanya alat tersebut maka sebagai Penggunaan konveyor dapat menghemat biaya produksi yang tinggi serta meningkatkan laju produksi dengan kecepatan yang signifikan dan stabil.	2020
Penyortir Bola Berwarna	Tujuan dari penelitian ini	Hasil dari penelitian ini	2022

Berbasis Arduino Menggunakan Metode <i>Fuzzy</i>	adalah untuk memanfaatkan fungsi arduino yang dapat digunakan untuk meningkatkan produktifitas dalam proses produksi	menunjukkan bahwa penyortir bola berwarna berbasis Arduino berjalan dengan baik	
--	--	---	--

1

2.2 Teori Penunjang Penelitian

2.2.1 Sistem Sortir Uang Pada *MoneyBox Plus*

Menurut (Gudang, 2022) Secara umum, sortir barang merujuk pada proses memisahkan barang sesuai dengan tujuan atau keperluannya. Dalam kehidupan sehari-hari, konsep sortir lebih dikenal dengan istilah decluttering.

Pemisahan barang sesuai dengan fungsinya memiliki manfaat dalam menjaga kebersihan dan kerapihan. Sortir barang dalam konteks ini berarti memisahkan barang-barang yang tidak terpakai untuk memberikan ruang di area tertentu, seperti ruang tamu dan kamar tidur.

Sistem sortir adalah suatu sistem yang digunakan untuk mengelompokkan barang-barang berdasarkan tujuannya. Dalam hal ini, sistem sortir berfungsi sebagai satu kesatuan untuk mencapai tujuan tertentu dengan cara yang efektif dan efisien.

Sistem sortir uang pada *MoneyBox Plus* adalah alat yang dapat memisahkan uang pecahan Rp 50.000,00 dan Rp 100.000,00 di dalam *MoneyBox Plus*. Hal ini bertujuan

untuk memudahkan pengambilan uang dari *MoneyBox Plus*.

2.2.2 Prototype

Menurut (Purnomo, 2017) *Prototyping* adalah sebuah metode pengembangan perangkat lunak yang melibatkan pembuatan model fisik yang berfungsi sebagai versi awal dari sistem yang sedang dikembangkan. Tujuan utama dari metode *Prototyping* ini adalah memungkinkan interaksi antara pengembang dan pengguna dalam proses pengembangan sistem informasi.

Proses pembuatan *prototype* yang sukses dapat dicapai dengan mendefinisikan aturan-aturan pada tahap awal. Baik pengembang maupun pengguna harus memiliki pemahaman yang sama bahwa *prototype* dibuat untuk mendefinisikan kebutuhan awal. Selama proses pengembangan, bagian-bagian dari *prototype* dapat ditambahkan atau dihilangkan sesuai dengan perencanaan dan analisis yang dilakukan oleh pengembang. Ujicoba *prototype* dilakukan secara simultan seiring dengan proses pengembangan untuk memastikan kesesuaian dengan kebutuhan yang diinginkan.

Dengan demikian, *Prototyping* menjadi sebuah pendekatan yang efektif dalam membangun sistem informasi, karena memungkinkan interaksi aktif antara pengembang dan pengguna serta memungkinkan perubahan dan perbaikan yang fleksibel selama proses pengembangan

Ada 4 metodologi *Prototyping* yang paling utama yaitu :

1. *Illustrative*, menghasilkan contoh laporan dan tampilan layar.
2. *Simulated*, mensimulasikan beberapa alur kerja sistem tetapi tidak menggunakan data *real*.
3. *Functional*, mensimulasikan beberapa alur sistem yang sebenarnya dan menggunakan data *real*.
4. *Evolutionary*, menghasilkan model yang menjadi bagian dari operasional sistem.

2.2.3 Motor Servo

Menurut (Dewi Lestari, 2019) *Motor Servo* adalah perangkat atau motor yang dirancang dengan sistem kontrol umpan balik loop tertutup (*servo*) untuk mengatur dan memastikan posisi sudut poros output motor. *Motor Servo* terdiri dari motor DC, rangkaian gigi, rangkaian kontrol, dan potensiometer.

Dalam konteks pemilihan objek benda, *Motor Servo* jenis standar dapat digunakan. *Motor Servo* standar adalah alat yang dapat mengendalikan, membungkuk, dan menjaga posisi berdasarkan sinyal elektronik yang diterima. Hal ini terjadi karena *Motor Servo* DC mengubah energi listrik menjadi energi gerak. *Motor Servo* standar memiliki kemampuan putaran hingga 180°. Jenis *Motor Servo* yang digunakan dalam hal ini adalah TowerPro Mg90s.



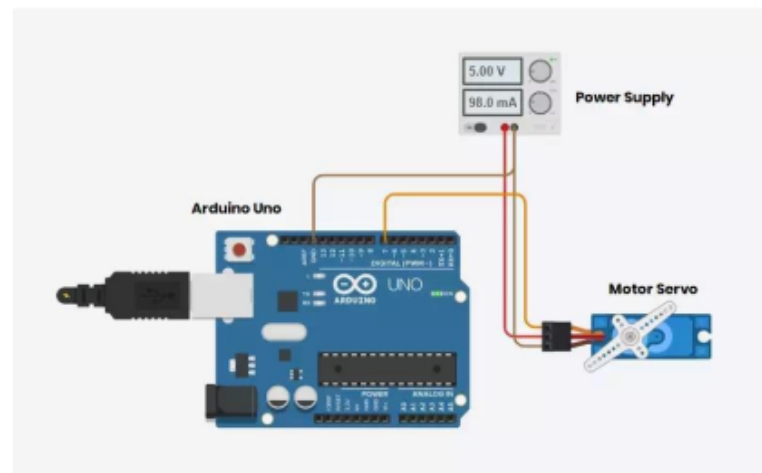
Gambar 2. 1 *Motor Servo*

2.2.3.1 Penggunaan *Motor Servo* pada *Arduino*

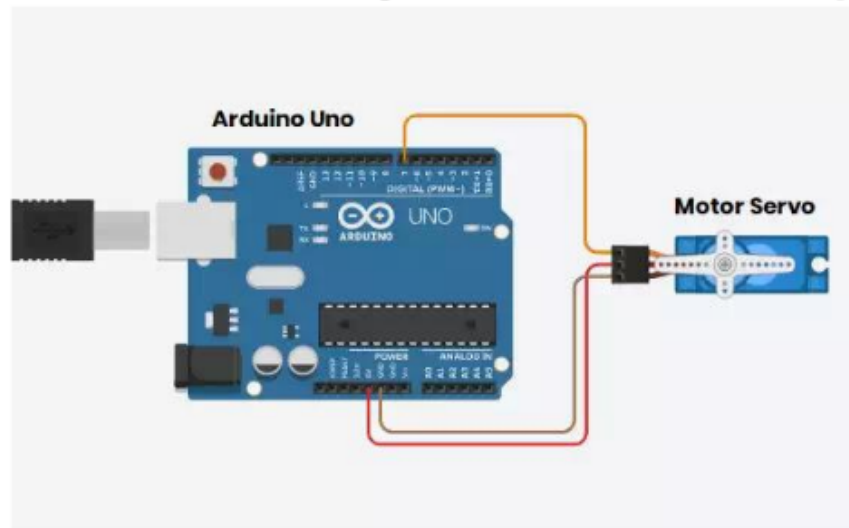
Mengontrol *Motor Servo* menggunakan *Arduino* sangat mudah karena *Arduino* memiliki perpustakaan sendiri untuk menggunakan servo. Kali ini kita akan mencoba untuk mengontrol servo agar *servo* dapat berputar ke posisi yang kita ingin kan, misalnya mengatur *Motor Servo* berputar ke sudut 90° .

Berikut adalah cara menghubungkan kabel pada *Motor Servo* ke *Arduino*:

1. Kabel hitam/cokelat terhubung ke *ground*.
2. Kabel kuning/oranye terhubung ke port 7 *Arduino*.
3. Kabel merah terhubung ke sumber tegangan (sebaiknya mengambil dari sumber tegangan eksternal).



Gambar 2. 2 *Motor Servo* pada Arduino sebelum ke-trigger



Gambar 2. 3 *Motor Servo* pada Arduino setelah ke-trigger

2.2.4 Mikrokontroller Arduino

Menurut (Aguskhumaidi, 2019) ⁴ Arduino adalah pengendali *mikro single-board* yang bersifat *open-source*, diturunkan dari *Wiring platform*, dirancang untuk memudahkan penggunaan elektronik dalam berbagai bidang. *Hardware*nya memiliki prosesor Atmel AVR dan *software*nya memiliki bahasa pemrograman sendiri. Saat ini Arduino sangat populer di seluruh dunia. Banyak pemula

yang belajar mengenal robotika dan elektronika lewat Arduino karena mudah dipelajari. Tapi tidak hanya pemula, para *hobbyist* atau profesional pun ikut senang mengembangkan aplikasi elektronik menggunakan Arduino. Bahasa yang dipakai dalam Arduino bukan *assembler* yang relatif sulit, tetapi bahasa C yang disederhanakan dengan bantuan pustaka-pustaka (*libraries*) Arduino.

AVR *Microcontroller* adalah *microcontroller* yang paling penting penggunaannya dikarenakan semua sistem pengendalian proses, pengolahan data, dan pengaturan dipusatkan. AVR memiliki arsitektur RISC (*Reduced Instruction Set Computing*) 8bit. Yang dimana semua perintah di export kedalam kode 16bit dan sebagian lainnya dijalankan dalam 1 siklus clock.



Gambar 2. 4 Mikrokontroler Arduino

2.2.4.1 Arduino Mega 2560

Menurut (Simatupang, Prasetyo, Galina, & Suhartomo, 2022) Arduino Mega 2560 merupakan board mikrokontroler berbasis Arduino Mega 2560 yang memiliki 54 pin digital *input* atau *output*. 15 pin diantaranya berfungsi untuk output PWM, 16 pin untuk *input* analog, 4 pin untuk UART (*port serial hardware*), 16 MHz kristal osilator, koneksi USB, *jack power*, *header ICSP*, dan tombol *reset*. Pin-pin tersebut berfungsi sebagai pendukung kinerja dari mikrokontroler. Untuk mengaktifkan Arduino Mega 2560 bisa dengan cara menghubungkan kabel USB ke komputer atau menggunakan power adaptor AC-DC atau bisa juga menggunakan baterai.

Arduino Mega 2560 memiliki banyak koneksi pin, sehingga memudahkan pengguna untuk mengatur seberapa efektif dari pin-pin tersebut. Selain itu, sebagian besar *shield* yang telah dibuat untuk Arduino Duemiliamove atau Arduino Diecimilia kompatibel dengan Arduino Mega 2560.

Arduino Mega 2560 merupakan versi terbaru, dan telah menggantikan dari versi sebelumnya yaitu Arduino Mega.

Spesifikasi teknis dari Arduino Mega 2560 adalah sebagai berikut :

Tabel 2.2 Spesifikasi Arduino Mega 2560

Mikrokontroler	ATmega2560
Tegangan Operasi	5V
Tegangan Input	7-12V
Pin I/O Digital	54
Pin Analog	16
Arus DC tiap pin I/O	20 mA
Arus DC untuk pin 3.3V	50 mA
<i>Flash Memory</i>	256 KB
SRAM	8 KB
EEPROM	4 KB
<i>Clock Speed</i>	16 MHz



Gambar 2. 5 Arduino Mega 2560

2.2.4.2 Arduino IDE

Menurut (Ikhsan, Yahya, & Fiolana, 2018) IDE merupakan singkatan dari *Integrated Development Environment* (Lingkungan Pengembangan Terintegrasi). Arduino IDE adalah salah satu contoh dari IDE yang digunakan untuk mengembangkan program pada *platform* Arduino.

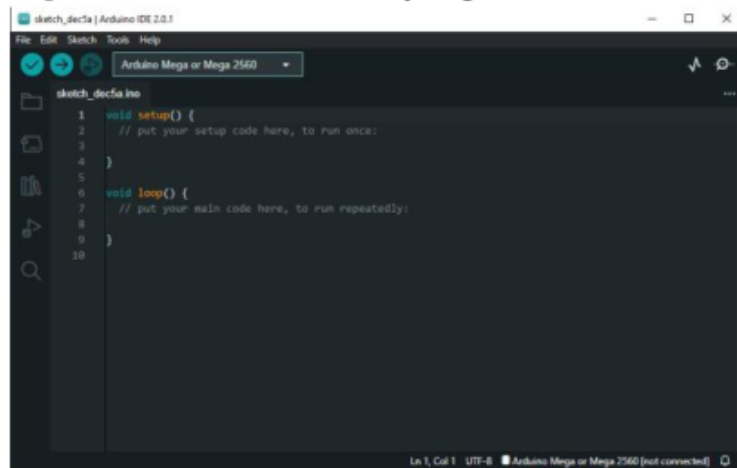
Arduino IDE menggunakan bahasa pemrograman yang serupa dengan bahasa C dan C++, sehingga pemrogram dapat membuat program yang sesuai dengan kebutuhan mereka menggunakan *software* ini. IDE ini menyediakan antarmuka yang intuitif dan sederhana, memungkinkan pemula untuk dengan mudah memulai pemrograman dengan Arduino.

Salah satu fitur penting pada Arduino adalah adanya bootloader yang telah terprogram di dalam mikrokontroler. *Bootloader* berfungsi sebagai perantara antara kompilator (*compiler*) dan mikrokontroler, memungkinkan pemrogram untuk mengunggah program ke papan Arduino tanpa perlu menggunakan perangkat pemrograman eksternal yang rumit.

Arduino IDE memang dirancang dengan tujuan untuk menjadi *user-friendly*, sehingga bahkan pemula dalam pemrograman dapat dengan mudah menggunakannya. Selain itu, IDE ini juga sering mendapatkan pembaruan dan pengembangan baru, sehingga terus meningkatkan kemudahan penggunaannya dan memberikan fitur-fitur yang lebih baik kepada pengguna.

Beberapa fitur pada *Software* Arduino IDE adalah sebagai berikut :

1. *Verify*, digunakan untuk meng-*compile* atau mengecek *coding* yang ditulis apakah masih ada kesalahan atau tidak.
2. *Upload*, digunakan untuk mengirimkan atau memasukan program yang dibuat ke dalam *board* yang telah ditentukan.
3. *New*, digunakan unuk membuat halaman sketch yang baru.
4. *Open*, digunakan untuk membuka *project* yang pernah dibuat sebelumnya.
5. *Save*, digunakan untuk menyimpan *sketch* atau program yang telah dibuat.
6. Serial Monitor, digunakan untuk menampilkan data yang telah dibuat setelah *sketch* tersebut di-*upload* ke dalam *board* yang telah ditentukan.



Gambar 2. 6 *Software* Arduino IDE







2.2.5 Flowchart

Menurut (Setiawan, 2021) *Flowchart* atau bagan alur adalah diagram yang digunakan untuk menggambarkan langkah-langkah dan keputusan dalam suatu proses pemrograman. Setiap langkah diwakili oleh simbol-simbol grafis dan dihubungkan dengan panah atau garis.







Flowchart memiliki peran penting dalam menentukan fase atau fungsi dari proyek perangkat lunak yang melibatkan tim pengembang yang bekerja bersama-sama. Dengan menggunakan *Flowchart*, alur program dapat dijelaskan dengan lebih jelas, ringkas, dan mengurangi kemungkinan terjadinya kesalahan penafsiran. Penggunaan *Flowchart* dalam pemrograman juga merupakan cara yang baik untuk menghubungkan kebutuhan teknis dan non-teknis.

Menggunakan *Flowchart* membantu pengembang perangkat lunak dalam memvisualisasikan alur program secara sistematis. Ini memungkinkan mereka untuk melihat langkah-langkah yang diperlukan, pengambilan keputusan, dan aliran data atau informasi yang terlibat dalam proses tersebut. Dengan adanya *Flowchart*, tim pengembang dapat bekerja secara efisien dan saling memahami dalam melaksanakan proyek perangkat lunak.

Berikut adalah contoh macam-macam simbol dari *Flowchart* Sebagai berikut :

	<p>Flow</p> <p>Simbol yang digunakan untuk menggabungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga dengan Connecting Line.</p>
	<p>On-Page Reference</p> <p>Simbol untuk keluar - masuk atau penyambungan proses dalam lembar kerja yang sama.</p>
	<p>Off-Page Reference</p> <p>Simbol untuk keluar - masuk atau penyambungan proses dalam lembar kerja yang berbeda.</p>
	<p>Terminator</p> <p>Simbol yang menyatakan awal atau akhir suatu program.</p>
	<p>Process</p> <p>Simbol yang menyatakan suatu proses yang dilakukan komputer.</p>
	<p>Decision</p> <p>Simbol yang menunjukkan kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, yaitu ya dan tidak.</p>

Gambar 2. 7 Simbol *Flowchart 1*

	<p>Input/output</p> <p>Simbol yang menyatakan proses input atau output tanpa tergantung peralatan.</p>
	<p>Manual Operation</p> <p>Simbol yang menyatakan suatu proses yang tidak dilakukan oleh komputer.</p>
	<p>Document</p> <p>Simbol yang menyatakan bahwa input berasal dari dokumen dalam bentuk fisik, atau output yang perlu dicetak.</p>
	<p>Predefine Proses</p> <p>Simbol untuk pelaksanaan suatu bagian (sub-program) atau prosedur.</p>
	<p>Display</p> <p>Simbol yang menyatakan peralatan output yang digunakan.</p>
	<p>Preparation</p> <p>Simbol yang menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberikan nilai awal.</p>

Gambar 2. 8 Simbol *Flowchart 2*

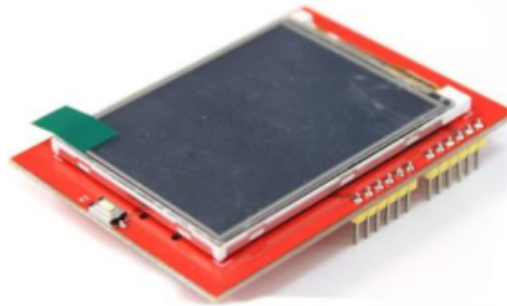
2.2.6 TFT LCD DISPLAY

Menurut (Rijali & Khana, 2020) TFT LCD (*Thin-Film Transistor Liquid Crystal Display*) adalah modul layar yang menggunakan empat LED putih sebagai *backlight*. Modul ini memiliki ukuran piksel 240 x 320 dan menggunakan skema warna dasar RGB (*Red Green Blue*). Selain itu, TFT LCD juga dilengkapi dengan fitur layar sentuh (*touchscreen*).

Modul TFT LCD ini dapat digunakan dengan board Arduino Mega 2560. Dalam penelitian ini, TFT LCD berfungsi sebagai input kode pengaman dalam bentuk PIN. Layar TFT LCD memiliki ukuran 2.4 inci dan dapat dioperasikan dengan menggunakan jari tangan atau stylus pen.

Dengan adanya modul TFT LCD ini, pengguna dapat memasukkan kode pengaman melalui layar sentuh, sehingga memberikan kemudahan dan kenyamanan dalam mengoperasikan sistem. Ukuran layar yang relatif kecil memungkinkan pengguna untuk melakukan *input* dengan presisi yang baik, baik menggunakan jari tangan maupun *stylus pen*.

Modul TFT LCD pada Arduino Mega 2560 memberikan fleksibilitas dalam pengembangan sistem dengan tampilan grafis yang menarik dan kemampuan interaksi langsung melalui layar sentuh.



Gambar 2. 9 TFT LCD *Display Shield*

2.2.7 Pemanfaatan LCD pada Mikrokontroler Arduino

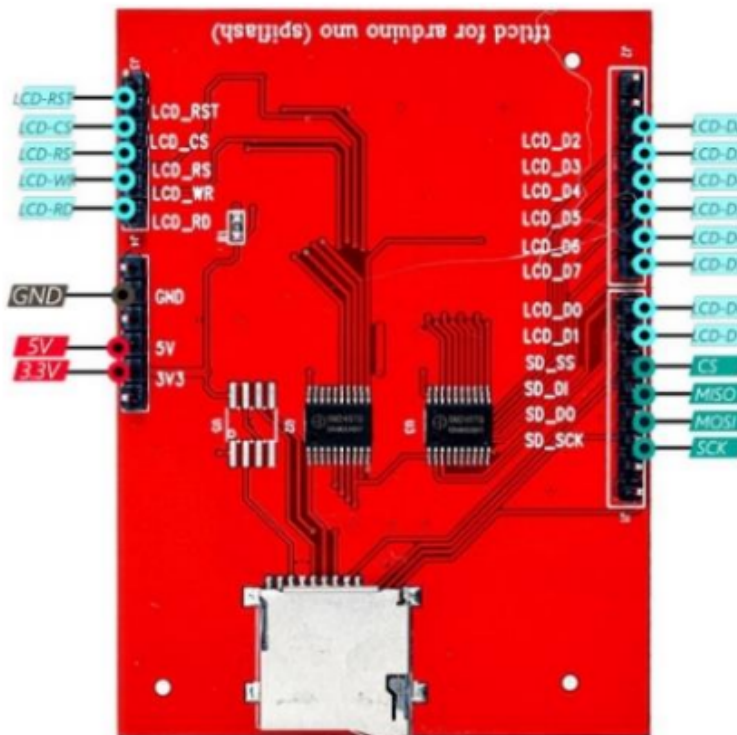
(ditempel, 2021) Pada modul LCD, beberapa pin memiliki fungsi dan koneksi yang spesifik. Berikut adalah penjelasan tentang fungsi dan koneksi pin-pi tersebut:

1. VSS (*Ground*): Pin VSS berfungsi sebagai *ground* atau koneksi tanah. Pin ini dapat dihubungkan ke pin GND pada Arduino untuk menyediakan jalur koneksi tanah yang diperlukan.
2. VDD/VCC: Pin VDD atau VCC digunakan untuk menyediakan sumber daya listrik pada modul LCD. Biasanya disambungkan ke sumber listrik 5 volt, dan dapat dihubungkan ke pin 5V pada Arduino.
3. V0 (*Contrast*): Pin V0 digunakan untuk mengatur kecerahan atau kontras layar LCD. Pin ini terhubung ke potensiometer, dengan menggunakan potensiometer 10K ohm untuk mengatur kecerahan yang diinginkan. Jika tidak menggunakan potensiometer, layar LCD dapat terlalu terang sehingga karakter tidak terlihat dengan jelas.
4. RS (*Register Select*): Fungsi pin RS adalah untuk mengirimkan perintah atau data ke layar LCD. Ketika pin RS LOW (0), perintah dikirimkan ke LCD seperti

membersihkan layar, mematikan layar, mengatur posisi kursor, dan lain-lain. Ketika pin RS *HIGH* (1), itu berarti mengirimkan data, seperti mengirimkan karakter untuk ditampilkan di layar.

5. RW (*Read/Write*): Pin RW digunakan untuk membaca atau menulis ke layar LCD. Dalam kebanyakan kasus, kita tidak membutuhkan fungsi pin ini, sehingga dapat dihubungkan ke *ground* untuk menetapkan mode penulisan (*write*) ke LCD.
6. E (*Enable*): Pin E (*Enable*) digunakan untuk mengaktifkan penulisan atau pembacaan fungsi ke register layar LCD. Ketika sinyal E berubah dari *LOW* (0) ke *HIGH* (1), maka data di register LCD akan diakses.
7. D0-D7 (*Data*): Pin D0 hingga D7 digunakan untuk mengirimkan data ke layar LCD. Pada modul LCD yang menggunakan mode 4-bit, hanya pin D4, D5, D6, dan D7 yang digunakan. Pin-pin ini akan mengirimkan 4 bit data pertama, kemudian diikuti oleh 4 bit data kedua untuk membentuk data 8 bit yang akan ditampilkan di layar.

Selain pin-pi di atas, terdapat juga pin untuk koneksi *power* dan *ground*, serta pin untuk mengatur kecerahan dengan menggunakan potensiometer. Dalam koneksi ke Arduino, hanya pin RS, E, D4, D5, D6, dan D7 yang terhubung ke pin Arduino sesuai dengan kebutuhan pengiriman perintah dan data ke layar LCD.



Gambar 2. 10 Komponen Modul LCD

2.2.8 Sensor

Menurut (Ningsih, 2022) sensor adalah perangkat yang digunakan untuk menerima input fisik dari lingkungan sekitarnya. Sensor ini dapat mengukur berbagai jenis parameter seperti suhu, tekanan, cahaya, suara, gerakan, dan banyak lagi. Setelah mengukur input tersebut, sensor mengubahnya menjadi data yang dapat diinterpretasikan oleh manusia atau mesin.

Sebagian besar sensor menggunakan prinsip elektronik untuk mengubah input fisik menjadi data elektronik yang dapat diproses lebih lanjut. Contohnya adalah sensor suhu yang menggunakan termokopel atau

sensor suhu berbasis semikonduktor untuk menghasilkan sinyal listrik yang berkorelasi dengan suhu.

Namun, ada juga sensor yang lebih sederhana seperti termometer raksa (termometer kaca) yang menggunakan ekspansi termal dari merkuri untuk mengukur suhu. Meskipun cara kerjanya berbeda, tujuan dari sensor tetap sama, yaitu mengukur dan menghasilkan data yang merepresentasikan *input* fisik yang diterima.

Penting untuk dicatat bahwa sensor bekerja berdasarkan pemicu atau trigger dari luar. Mereka tidak membutuhkan perintah atau instruksi dari sistem atau manusia untuk berfungsi. Ketika sensor menerima stimulus atau perubahan di lingkungan sekitarnya, mereka akan merespons secara otomatis dengan menghasilkan keluaran yang sesuai.

Sensor memiliki peran yang penting dalam berbagai aplikasi, termasuk di bidang industri, otomotif, kesehatan, lingkungan, dan banyak lagi. Data yang dikumpulkan oleh sensor dapat digunakan untuk pengendalian sistem, pemantauan kondisi, pengambilan keputusan, dan pengembangan solusi yang lebih cerdas dan efisien.

Sensor yang baik adalah yang memiliki tiga prinsip sebagai berikut :

1. Sensitif terhadap elemen yang akan diukur.
2. Tidak terpengaruh oleh faktor selain yang akan diukur.
3. Jangan mengubah status item pengukuran.

2.2.9 Internet Of Things

.*Internet of Things* (IoT) adalah konsep di mana objek atau benda sehari-hari dilengkapi dengan teknologi seperti sensor, perangkat lunak, dan koneksi internet untuk berkomunikasi dan bertukar informasi dengan perangkat lain. Dengan adanya IoT, objek-objek tersebut dapat saling terhubung dan berinteraksi secara otomatis tanpa intervensi manusia.

Konsep M2M (*Machine-to-Machine*) juga terkait erat dengan IoT. M2M mengacu pada komunikasi antara perangkat elektronik, di mana perangkat tersebut dapat saling berinteraksi dan bertukar informasi tanpa keterlibatan manusia. Setiap perangkat yang memiliki kemampuan komunikasi M2M sering disebut sebagai perangkat pintar atau *smart device*.

Dalam konteks IoT, perangkat pintar ini dirancang untuk membantu manusia dalam berbagai tugas sehari-hari. Mereka dapat mengumpulkan data melalui sensor-sensor yang terpasang, mengirimkan informasi ke server atau platform, dan menerima instruksi atau perintah untuk mengontrol atau berinteraksi dengan perangkat lain. Contohnya, lampu pintar yang dapat diatur melalui *smartphone*, pengunci pintu otomatis yang dapat dikendalikan jarak jauh, atau perangkat pengukur kelembaban tanah yang memberikan data secara *real-time* kepada petani.

Melalui konektivitas dan kemampuan berkomunikasi, IoT memberikan potensi untuk menghubungkan dan mengintegrasikan berbagai aspek kehidupan sehari-hari secara lebih cerdas dan efisien. Ini membuka peluang untuk

meningkatkan produktivitas, efisiensi energi, kenyamanan, keamanan, dan kualitas hidup secara keseluruhan.












2.2.10 Modeling Quick Desain

Quick Plan and Modelling Quick Design adalah tahapan ini akan melanjutkan proses yang ada dalam tahap communication, yaitu membuat perencanaan dan permodelan secara cepat tersebut lebih difokuskan dalam mempresentasikan aspek-aspek yang akan ditampilkan atau dilihat oleh pelanggan.

2.2.11 Diagram Alur

Diagram yang mewakili algoritme, alur kerja, atau proses, menunjukkan langkah-langkah dalam bentuk simbol grafik, urutannya dihubungkan dengan panah. Diagram ini menunjukkan contoh atau deskripsi solusi untuk masalah tersebut. Bagan alir digunakan untuk menganalisis, merancang, mendokumentasikan, atau mengelola suatu proses atau program di berbagai bidang.

Diagram Alur digunakan untuk merancang dan mendokumentasikan proses atau program sederhana. Seperti jenis diagram lainnya, mereka membantu menggambarkan apa yang terjadi dan dengan demikian membantu memahami prosesnya, dan dimungkinkan untuk menemukan celah fungsional atau ambiguitas dalam proses tersebut.

Gambar	Nama	Keterangan
	Garis Alir	Menunjukkan arah aliran algoritma, dari satu proses ke proses berikutnya.
	Terminal	Menunjukkan awal atau akhir sebuah proses.
	Proses / Langkah	Menyatakan kegiatan yang akan terjadi dalam diagram alir.
	Titik Keputusan	Proses / langkah di mana perlu adanya keputusan atau adanya kondisi tertentu. Di titik ini selalu ada dua keluaran untuk melanjutkan aliran kondisi yang berbeda.
	Masukan / Keluaran	Digunakan untuk mewakili data masuk, atau data keluar. Hanya bisa dimulai dari masukan menuju keluaran, bukan sebaliknya.
	Anotasi	Melambungkan komentar tentang suatu atau beberapa bagian dari diagram alir. Tentu saja, komentar tidak memiliki dampak apapun terhadap proses yang berlangsung.
	Predefined Process	Digunakan untuk menunjukkan suatu proses yang begitu kompleks, sehingga tidak bisa dijelaskan di diagram alir ini dan merujuk pada diagram alir yang terpisah.
	Persiapan / Inisialisasi	Menunjukkan operasi yang tidak memiliki efek khusus selain mempersiapkan sebuah nilai untuk langkah / proses berikutnya. Lambang ini juga digunakan untuk menggantikan titik keputusan yang biasanya berbentuk ketupat jika ingin menggunakan pengulangan pada kondisi tertentu.
	Konektor Dalam Halaman	Biasanya digunakan dalam pengulangan. Digunakan untuk menghubungkan satu proses ke proses lainnya, sama halnya seperti tanda panah. Boleh saja lebih dari satu proses yang mengarah kepadanya, namun hanya bisa menghasilkan satu keluaran. Sehingga diagram alir terlihat lebih rapi karena mengurangi tanda panah yang lalu lalang di dalam diagram alir.
	Konektor Luar Halaman	Terkadang, diagram alir tidak muat dalam satu halaman saja. Oleh karena itu, lambang ini berfungsi untuk menghubungkan satu proses ke proses lainnya, sama halnya seperti tanda panah, hanya saja untuk merujuk ke halaman yang berbeda.
	Kontrol / Inspeksi	Menunjukkan proses / langkah di mana ada inspeksi atau pengontrolan.

Gambar 2. 11 Simbol Diagram Alur

2.2.12 Uang

Dalam ekonomi modern, uang adalah sesuatu yang tersedia dan diterima secara umum sebagai alat pembayaran untuk pembelian barang dan jasa serta aset berharga lainnya, dan untuk pembayaran utang. Beberapa ahli juga menyebutkan menghasilkan uang sebagai cara untuk menunda pembayaran. Singkatnya, uang adalah objek yang biasa diterima oleh masyarakat untuk mengukur nilai, pertukaran dan pembayaran untuk pembelian barang dan jasa, selain itu juga berfungsi sebagai alat untuk mengumpulkan kekayaan. Dilihat dari bahan pembuatannya, jenis uang dibedakan menjadi dua berdasarkan bahannya.

Undang-Undang Nomor 13 Pasal 26 Ayat 1 Tahun 1968 mencabut hak pemerintah untuk mencetak uang. Pemerintah kemudian membentuk bank sentral, Bank Indonesia, sebagai satu-satunya lembaga yang berhak menciptakan mata uang. Hak untuk menciptakan uang disebut hak octroi. Gramedia bisa mengetahui lebih dalam tentang keuangan Indonesia di kancah politik dalam buku *Power of Money* karya Burhanuddin Muhtadi. (Gramedia, 2021)

2.2.12.1 Jenis Uang Terhadap Pembuatannya

Jenis uang terhadap pembuatannya dibagi menjadi 2 yaitu :

1. Uang Logam : Koin terbuat dari logam; biasanya terbuat dari emas atau perak karena kedua logam ini cenderung bernilai tinggi dan stabil, bentuknya mudah dikenali, tidak mudah

hancur, tahan lama dan dapat dipecah menjadi satuan yang lebih kecil tanpa kerusakan.

2. Uang Kertas : Uang kertas adalah uang kertas dengan gambar dan stempel tertentu, dan merupakan alat pembayaran yang sah. Undang-Undang Bank Indonesia No. 23 Tahun 1999 Uang kertas adalah uang dalam bentuk lembaran kertas atau (Chairul Gunawan, 2021) (Anantajaya, 2021) bahan lain (sejenis kertas).

2.2.13 EPROME

EPROM (*Erasable Programmable Read-Only Memory*) adalah jenis chip memori yang dapat menyimpan data bahkan ketika catu daya dimatikan. Dalam konteks memori komputer, EPROM dikategorikan sebagai jenis memori *non-volatile*, yang berarti data yang tersimpan dalam chip dapat dipertahankan bahkan setelah catu daya dimatikan dan dihidupkan kembali.

EPROM terdiri dari serangkaian transistor dengan gate yang dapat diprogram ulang. Setiap transistor dalam chip mewakili satu bit data. Untuk memprogram EPROM, perangkat elektronik yang memasok tegangan yang lebih tinggi dari yang biasanya digunakan dalam rangkaian digital digunakan untuk memprogram transistor-transistor tersebut.

Satu fitur khas dari EPROM adalah kemampuannya untuk dihapus. Proses penghapusan EPROM melibatkan paparan chip ke sumber cahaya ultraviolet yang kuat,

seperti sinar ultraviolet dari lampu uap merkuri. EPROM mudah dikenali karena memiliki jendela transparan yang terbuat dari leburan kuarsa di atas paket chip. Jendela ini memungkinkan sinar ultraviolet untuk mencapai chip dan menghapus data yang telah diprogram sebelumnya.

Proses penghapusan EPROM memungkinkan chip untuk diprogram ulang dengan data yang berbeda setelah dihapus. Hal ini membuat EPROM sangat berguna dalam situasi di mana perubahan data yang disimpan dalam chip diperlukan secara periodik. Namun, perlu dicatat bahwa proses penghapusan EPROM melibatkan paparan yang kuat terhadap sinar ultraviolet, dan oleh karena itu perlu dilakukan dengan hati-hati.

EPROM digunakan dalam berbagai aplikasi di mana data perlu disimpan secara permanen dan dapat diubah setelah diprogram. Contohnya termasuk BIOS komputer, firmware perangkat keras, dan aplikasi lain yang memerlukan penyimpanan data yang dapat diprogram ulang.

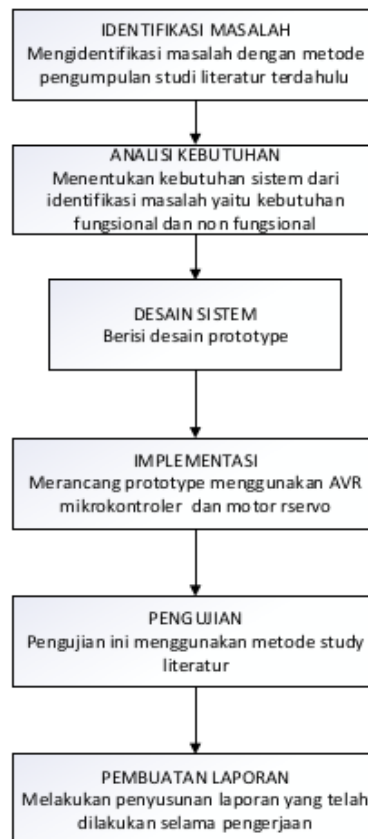
BAB III

METODE PENELITIAN

3.1 Tahapan Penelitian

3.1.1 Alur Penelitian

Penelitian Sistem Sortir Uang Pada Kotak Penyimpanan Uang (*MoneyBox Plus*) Berbasis Mikrokontroler Arduino dikerjakan menggunakan metode *prototype*. Alur penelitian ini digunakan sebagai acuan atau pedoman dalam proses pengerjaan penelitian yang agar dapat berjalan secara sistematis, efektif, dan diselesaikan tepat pada waktunya. Adapun tahapan tahapan sebagai berikut:



Gambar 3. 1 Diagram Alur

3.1.2 Identifikasi Masalah

Pada penelitian ini dibutuhkan cara dan Langkah dalam penyelesaian masalah merancang sistem sortir uang pada kotak penyimpanan uang (*MoneyBox Plus*) berbasis mikrokontroler Arduino. Penelitian ini mengimplementasikan Bahasa Pemrograman C sebagai dasar penelitian serta penelitian ini akan mendapatkan data yang akurat dari beberapa metode, diantara lain :

1. Studi Literatur

Studi literatur merupakan upaya mencari dan mempelajari berbagai sumber tulisan seperti jurnal, buku, laporan penelitian, situs internet dan berbagai artikel yang memiliki tema yang sama atau serupa dari sistem sortir uang pada kotak penyimpanan uang (*MoneyBox Plus*) berbasis mikrokontroler arduino. Studi literatur ini menjadi bagian penting untuk memperbanyak teori penelitian yang akan di uji

2. Percobaan Langsung

Percobaan pada sistem sortir uang pada kotak penyimpanan uang (*MoneyBox Plus*) berbasis mikrokontroler arduino digunakan untuk mengetahui apakah ada kendala dan masalah dalam perancangan *prototype* sehingga ada langkah perbaikan agar sistem berfungsi sesuai dengan tujuan yang diharapkan.

3.1.3 Analisis Kebutuhan

Setelah melakukan identifikasi masalah didapatkan data dan informasi yang berhubungan dengan sistem sortir uang pada penyimpanan uang (*MoneyBox Plus*) berbasis mikrokontroler Arduino dan akan dibangun sebuah

prototypenya. Maka analisis kebutuhan akan dipecah menjadi 2 bagian yaitu Analisis Kebutuhan Fungsional dan Non Fungsional.

3.1.3.1 Analisis Kebutuhan Fungsional

Analisa kebutuhan fungsional adalah menjelaskan mengenai layanan atau fungsi apa saja yang disediakan oleh sistem untuk memudahkan user dalam menggunakannya. Di dalam kebutuhan fungsional ini terdiri dari *input*, proses, *output*, *database*, dan kebutuhan user. Kebutuhan fungsional dapat diartikan sebagai fitur apa saja yang dimiliki sistem atau kebutuhan yang berisi tentang proses-proses apa saja yang nantinya dilakukan oleh sistem. Adapun proses yang ada pada sistem ini nanti yaitu:

1. Pada perancangan sistem sortir uang pada kotak penyimpanan uang (*MoneyBox Plus*) berbasis mikrokontroler Arduino ini memiliki fitur yang bisa mensortir uang pada pecahan Rp.50.000 dan Rp.100.000 ke kotak yang sudah disediakan.

3.1.3.2 Analisis Kebutuhan Non Fungsional

Kebutuhan non-fungsional merupakan kebutuhan diluar kebutuhan fungsional yaitu meliputi kebutuhan akan spesifikasi *hardware* dan kebutuhan *software* yang sangat mempengaruhi beroperasinya mesin. Adapun kebutuhan non-fungsional tersebut meliputi:

1. *Hardware:*

Spesifikasi *hardware* minimum yang dibutuhkan pada perancangan sistem sortir uang pada kotak penyimpanan uang (*MoneyBox Plus*) berbasis mikrokontroler arduino adalah

a. PC dengan spesifikasi minimal :

- Processor : 1.5 GHz Dual Core
- Memory (RAM) : 2 GB DDR3
- Harddisk : 120 GB
- Monitor : Resolusi Layar (1366 x 768)
- Keyboard : QWERT (86 keys)

b. *Motor Servo*

c. LCD

d. Kabel

e. Uang Rp.50.000 dan Rp.100.000

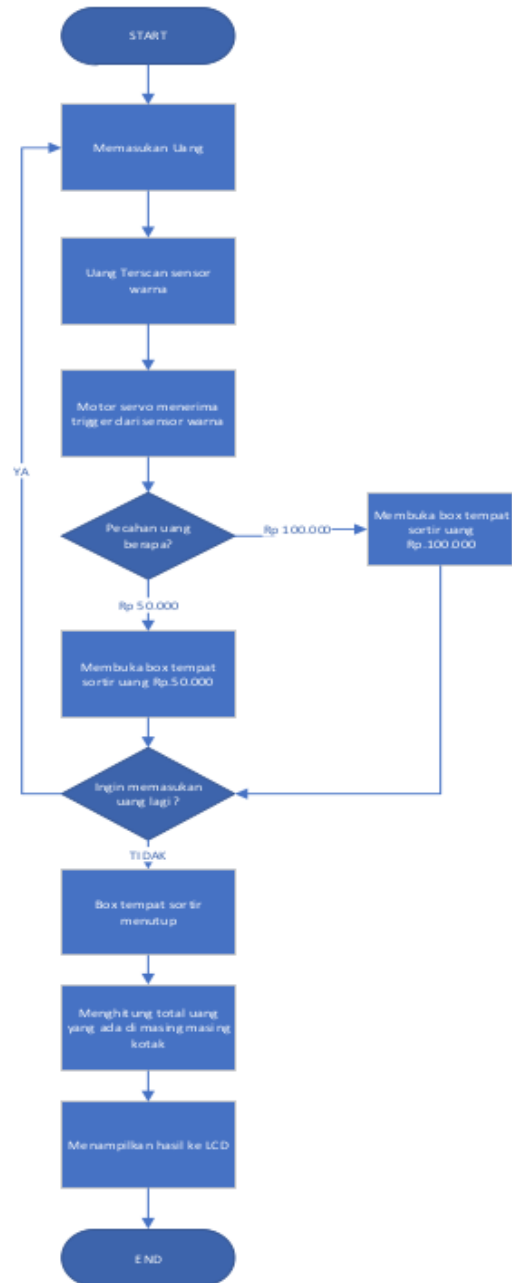
2. *Software:*

Spesifikasi minimal software minimum yang dibutuhkan pada perancangan sensor deteksi warna pada kotak penyimpanan uang (*MoneyBox Plus*) berbasis mikrokontroler arduino adalah:

1. Sistem Operasi : Windows 7
2. Aplikasi : Arduino IDE
3. Bahasa Pemrograman : Bahasa C

3.1.4 Flowchart

Adapun *Flowchart* yang digunakan pada sistem ini nanti yaitu:



Gambar 3.2 *Flowchart*

3.1.5 Modeling Quick Design

Tahapan ini akan menggambarkan desain rancangan rangkaian sistem sortir uang pada kotak penyimpanan uang (*MoneyBox Plus*) secara kasar atau gambaran umum, adapun Modeling desain cepat yang akan digunakan sebagai berikut:



Gambar 3. 3 *Modeling Quick Design*

3.1.6 *User Interface Design*

Pada tahap *user interface design* atau *UI Design* ini akan menggambarkan desain rancangan dari antarmuka pengguna yang ditampilkan pada *TFT LCD Display Shield* dan digunakan untuk memilih Menu yang terdapat di *Sistem Keamanan Pada Kotak Penyimpanan Uang (MoneyBox Plus) Berbasis Mikrokontroler Arduino*. Adapun desain rancangan dari antarmuka pengguna yang akan digunakan adalah sebagai berikut.

3.1.6.1 Design UI Menu Utama

Rancangan Desain UI Menu Utama ini digunakan sebagai tampilan awal pada Kotak Penyimpanan Uang (*MoneyBox Plus*), dan di tampilan ini pengguna dapat memilih beberapa Menu yang tersedia pada sistem. Diantaranya yang termasuk ke dalam Sistem Sortir ini adalah Menu Akses Masuk, Menu Ubah PIN, dan Menu Lupa PIN.



Gambar 3. 4 Rancangan UI utama

3.1.6.2 Desain UI Menu Cek Jumlah

Rancangan Desain UI Menu cek jumlah ini digunakan sebagai tampilan atau *layout* yang terdapat instruksi “Masukkan PIN”, selain itu juga berisi angka untuk memasukkan kode pengaman dan “OK” untuk konfirmasi kode pengaman, beserta “x” untuk

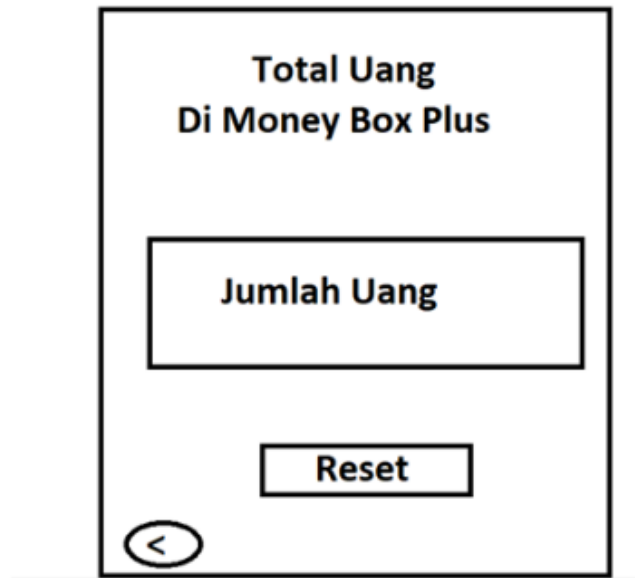
cancel atau kembali ke Menu tampilan awal. Pada Menu Akses Masuk untuk inputan kode pengaman disembunyikan.

Masukan Pin		
1	2	3
4	5	6
7	8	9

1	2	3
4	5	6
7	8	9

Gambar 3. 5 Rancangan UI Masukan PIN

Saat melakukan *input* kode pengaman pada Menu Akses Masuk dan kode pengaman tersebut cocok atau benar dengan PIN yang sudah tersimpan ke dalam sistem, maka tampilan akan berubah menjadi seperti gambar berikut :



Gambar 3. 6 Rancangan UI Cek Jumlah Uang

3.1.7 Sketching Proyotype

Setelah membuat tahapan perencanaan adalah tahap mengubah modeling *quick design* ke *sketching prototype* untuk mempermudah pembuatan *prototype*, berikut adalah hasil dari *sketching prototype* :



Gambar 3. 7 Sketching Prototype

3.1.8 Implementasi

Pada tahapan pengimplementasian ini akan menerapkan materi pada mata kuliah *Internet of Things* di jurusan Informatika dan pemngimplementasian koding sistem sortir uang pada kotak penyimpanan uang (*MoneyBox Plus*) memanfaatkan aplikasi Arduino IDE dan penyusunan alat sensor ini tinggal menyusun yang sudah ditetapkan di kebutuhan non-fungsional seperti LCD, kabel, dan *Motor Servo*.

3.1.9 Pengujian

Pada tahapan selanjutnya adalah pengujian atau proses eksekusi sistem sortir uang pada kotak penyimpanan uang (*MoneyBox Plus*) menentukan apakah alat ini berjalan sesuai dengan apa yang diinginkan. Pada tahapan pengujian kotak penyimpanan uang (*MoneyBox Plus*) ini menggunakan metode pengujian langsung.

3.1.10 Pembuatan Laporan

Pada tahapan pembuatan laporan, melakukan penyusunan laporan dari sistem sortir uang pada kotak penyimpanan uang (*MoneyBox Plus*) yang telah dibangun dan memaparkan hasil sistem yang telah selesai.

BAB IV HASIL DAN PEMBAHASAN

4.1 Pengujian Sistem

Pengujian sistem disini berdasarkan rancangan desain yang telah dibuat dan untuk mengetahui apakah sistem yang telah dibuat dapat berfungsi dengan sesuai yang diharapkan atau tidak. Berikut adalah uji coba sistem dari pengerjaan sistem berdasarkan rancangan desain yang telah dibuat. Adapun implementasinya pada Kotak Penyimpanan Uang (*MoneyBox Plus*) sebagai berikut :

4.1.1 Menu utama

Uji coba pada menu utama ini merupakan akses yang akan digunakan sebagai tampilan awal yang nantinya bisa menjadi jembatan kepada fitur scan uang yang ada pada Kotak Penyimpanan Uang (*MoneyBox Plus*).



Gambar 4. 1 Menu Utama

4.1.2 Menu Cek Jumlah

Menu Cek Jumlah merupakan akses yang di gunakan untuk melihat jumlah uang pada Kotak Penyimpanan Uang (*MoneyBox Plus*) dan hanya orang berhak saja yang memiliki akses. Adapun Pengujian Menu Akses Masuk ini adalah sebagai berikut.

1. Setelah masuk ke Menu Akses Masuk, masukkan PIN atau kode pengaman yang sesuai dengan PIN atau kode pengaman pada sistem untuk masuk ke dalam akses Kotak Penyimpanan Uang (*MoneyBox Plus*).



Gambar 4. 2 UI Menu Cek Jumlah

2. Jika PIN atau kode pengaman yang dimasukkan sesuai dengan PIN atau kode pengaman pada sistem, maka akan muncul jumlah uang yang telah tersimpan pada Kotak Penyimpanan Uang (*MoneyBox Plus*).



Gambar 4. 3 UI menampilkan jumlah uang

3. Sedangkan jika PIN atau kode pengaman yang dimasukkan tidak sesuai dengan PIN atau kode pengaman pada sistem, maka akses ditolak dan Kotak Penyimpanan Uang (*MoneyBox Plus*) tetap terkunci.



Gambar 4. 4 Akses Cek Jumlah Ditolak

4.1.3 Penerapan Cara Kerja Sistem Sortir

Penerapan cara kerja deteksi warna pada Kotak Penyimpanan Uang (*MoneyBox Plus*) sebagai berikut :

1. Sambungkan alat *MoneyBox Plus* pada *Power Supply*, yang dimana alat yang digunakan sebagai *power supply* ini menggunakan kabel data dari USB.
2. Setelah alat terhubung dengan *power supply* maka arduino yang bekerja sebagai otak mulai bekerja untuk menjalankan mesin pada Kotak Penyimpanan Uang (*MoneyBox Plus*).

3. Selanjutnya, menu utama dari Kotak Penyimpanan Uang ini akan menyala.
4. Tekan tombol pada layar dibagian Scan Uang, dan nanti akan pindah ke bagian Scan Uang.
5. Masukkan uang pecahan Rp.50.000 atau Rp.100.000 yang mau ditaruh ke Kotak Penyimpanan Uang (*MoneyBox Plus*), batas memasukkan uang dalam satu fase adalah 5 kali memasukkan uang setelah itu Kotak Penyimpanan Uang (*MoneyBox Plus*) akan menampilkan total uang yang sudah discan melalui scan warna.
6. Setelah memasukan uang pecahan Rp. 50.000 atau Rp.100.000 *Motor Servo* yang ada pada Kotak Penyimpanan uang (*MoneyBox Plus*) akan ke-*Trigger*.
7. Jika *Motor Servo* sudah ke-*Trigger* maka akan bergerak untuk membuka kotak pemisah yang ada di dalam Kotak Penyimpanan uang (*MoneyBox Plus*).
8. Setelah itu akan terbuka masing-masing kotak yang ada pada Kotak Penyimpanan uang (*MoneyBox Plus*) yang dimana akan mensortir uang pecahan Rp.50.000 atau Rp.100.000 sesuai kotak yang telah tersedia di dalam Kotak Penyimpanan uang (*MoneyBox Plus*).

4.2 Hasil Penelitian

Berdasarkan pengujian sistem yang dilakukan saat penelitian terhadap kotak Penyimpanan Uang (*MoneyBos Plus*) memperoleh hasil sebagai berikut :

4.2.1 *Layout* Kode Menu Utama

```
324 void Menu_display()
325 {
326     tft.setTextSize(3);
327     tft.setTextColor(BLACK);
328     tft.setCursor(40, 37);
329     tft.print("MoneyBox");
330
331     tft.setTextSize(3);
332     tft.setTextColor(BLACK);
333     tft.setCursor(185, 22);
334     tft.print("+");
335
336     DrawButtonScanUang();
337     DrawButtonCekJumlah();
338     DrawButtonAksesMasuk();
339     DrawButtonResetPIN();
340     DrawButtonLupaPIN();
341     DrawButtonClear();
342 }
```

Gambar 4. 5 *Source Code* Menu Utama

Pada gambar diatas, penjelasan mengenai kode dalam fungsi "Menu *Display*" pada program Arduino adalah sebagai berikut:

1. ``tft.setTextSize(3);``: Mengatur ukuran teks yang akan ditampilkan pada layar menjadi 3 kali ukuran default.
2. ``tft.setTextColor(BLACK);``: Mengatur warna teks menjadi hitam.
3. ``tft.setCursor(40, 37);``: Mengatur posisi kursor teks pada layar dengan koordinat (40, 37). Koordinat ini menentukan letak awal teks yang akan ditampilkan.
4. ``tft.print("MoneyBox");``: Mencetak teks "MoneyBox" pada layar pada posisi yang telah ditentukan sebelumnya.
5. ``tft.setTextSize(3);``: Mengatur ukuran teks kembali menjadi 3 kali ukuran default.
6. ``tft.setTextColor(BLACK);``: Mengatur warna teks kembali menjadi hitam.
7. ``tft.setCursor(185, 22);``: Mengatur posisi kursor teks pada layar dengan koordinat (185, 22).
8. ``tft.print("+");``: Mencetak tanda "+" pada layar pada posisi yang telah ditentukan sebelumnya.
9. Selanjutnya, terdapat pemanggilan beberapa fungsi lain, seperti ``DrawButtonScanUang()``, ``DrawButtonCekJumlah()``, ``DrawButtonAksesMasuk()``, ``DrawButtonResetPIN()``, ``DrawButtonLupaPIN()``, dan ``DrawButtonClear()``. Fungsi-fungsi ini bertanggung jawab untuk menggambar dan menampilkan tombol-tombol pada layar.

Dengan menjalankan fungsi "*Menu Display*", layar akan menampilkan teks "*MoneyBox*" dan beberapa tombol yang tergambar sesuai dengan pemanggilan fungsi-fungsi yang ada di dalamnya. Fungsi ini bertanggung jawab untuk menampilkan tampilan menu utama pada layar TFT (*Thin-Film Transistor*) display pada Arduino.

4.2.2 Layout Kode Menu Cek Jumlah

```
void DrawButtonCekJumlah()  
{  
  tft.fillRoundRect(123, 95, 95, 70, 10, BLACKM);  
  tft.fillRoundRect(125, 97, 91, 66, 10, BLUE);  
  tft.setTextSize(2);  
  tft.setTextColor(WHITE);  
  tft.setCursor(153, 110);  
  tft.print("CEK");  
  tft.setCursor(135, 135);  
  tft.print("JUMLAH");  
}
```

Gambar 4. 6 Source Code Menu Cek Jumlah Fungsi 1

Pada gambar diatas merupakan fungsi pertama, `DrawButtonCekJumlah()`, digunakan untuk menggambar tombol dengan teks "CEK JUMLAH". Berikut adalah langkah-langkah yang diambil dalam fungsi tersebut :

1. `tft.fillRoundRect(123, 95, 95, 70, 10, BLACKM);``:
Fungsi ini menggambar sebuah persegi panjang dengan sudut yang dibulatkan pada posisi (123, 95) dengan lebar 95 piksel dan tinggi 70 piksel. Parameter terakhir, ``BLACKM``, kemungkinan merujuk pada warna hitam dalam format yang digunakan oleh library yang digunakan dalam proyek ini.

2. `tft.fillRoundRect(125, 97, 91, 66, 10, BLUE);``: Fungsi ini menggambar persegi panjang dengan sudut yang dibulatkan pada posisi (125, 97) dengan lebar 91 piksel dan tinggi 66 piksel. Perbedaannya adalah warna yang digunakan kali ini adalah biru (BLUE).
3. `tft.setTextSize(2);``: Fungsi ini mengatur ukuran teks pada layar menjadi 2.
4. `tft.setTextColor(WHITE);``: Fungsi ini mengatur warna teks menjadi putih (WHITE).
5. `tft.setCursor(153, 110);``: Fungsi ini menentukan posisi kursor untuk penulisan teks berikutnya. Dalam hal ini, kursor ditempatkan pada posisi (153, 110).
6. `tft.print("CEK");``: Fungsi ini mencetak teks "CEK" pada posisi kursor yang telah ditentukan sebelumnya.
7. `tft.setCursor(135, 135);``: Fungsi ini menentukan posisi kursor untuk penulisan teks selanjutnya. Dalam hal ini, kursor ditempatkan pada posisi (135, 135).
8. `tft.print("JUMLAH");``: Fungsi ini mencetak teks "JUMLAH" pada posisi kursor yang baru ditentukan.

```

373 void DrawButtonCekJumlahPress()
374 {
375     tft.fillRoundRect(123, 95, 95, 70, 10, BLACKM);
376 }

```

Gambar 4. 7 *Source Code* Menu Cek Jumlah Fungsi 2

Pada gambar di atas merupakan Fungsi `DrawButtonCekJumlahPress()`. Berikut adalah langkah-langkah yang diambil dalam fungsi tersebut :

1. `tft.fillRoundRect(123, 95, 95, 70, 10, BLACKM);`: Fungsi ini menggambar persegi panjang dengan sudut yang dibulatkan pada posisi (123, 95) dengan lebar 95 piksel dan tinggi 70 piksel. Perbedaannya adalah kali ini persegi panjang diisi dengan warna hitam (BLACKM), menciptakan efek tekanan pada tombol.

```
1451 void writeIntIntoEEPROM(int address, int number)
1452 {
1453     EEPROM.write(address, number >> 8);
1454     EEPROM.write(address + 1, number & 0xFF);
1455 }
```

Gambar 4. 8 *Source Code* Menu Cek Jumlah Fungsi 3

Pada gambar diatas merupakan Fungsi `writeIntIntoEEPROM(int address, int number)` Fungsi ini digunakan untuk menulis sebuah bilangan bulat (*integer*) ke dalam EEPROM (*Electrically Erasable Programmable Read-Only Memory*). EEPROM adalah jenis memori yang dapat menyimpan data bahkan ketika daya listrik terputus. Berikut adalah langkah-langkah yang diambil dalam fungsi tersebut :

1. `EEPROM.write(address, number >> 8);`: Pada baris ini, kita menggunakan fungsi `EEPROM.write()` untuk menulis data ke alamat yang ditentukan dalam EEPROM. Parameter pertama, `address`, adalah alamat memori EEPROM di mana kita ingin menyimpan bilangan tersebut. Parameter kedua, `number >> 8`, adalah bilangan yang akan ditulis ke EEPROM setelah digeser 8 bit ke kanan. Hal ini dilakukan untuk mendapatkan 8 bit paling signifikan (*most significant bits/MSB*) dari bilangan `number`.

2. `EEPROM.write(address + 1, number & 0xFF);` Pada baris ini, kita menggunakan fungsi `EEPROM.write()` sekali lagi untuk menulis data ke alamat yang berikutnya dalam EEPROM. Alamat yang digunakan adalah `address + 1`, untuk menulis 8 bit paling tidak signifikan (least significant bits/LSB) dari bilangan `number`. `number & 0xFF` mengambil 8 bit terakhir dari `number` dengan menggunakan operasi bitwise AND dengan `0xFF`.

```
1457 int readIntFromEEPROM(int address)
1458 {
1459     return (EEPROM.read(address) << 8) + EEPROM.read(address + 1);
1460 }
1461
```

Gambar 4. 9 *Source Code* Menu Cek Jumlah Fungsi 4

Pada gambar diatas merupakan fungsi `readIntFromEEPROM(int address)` Fungsi ini digunakan untuk membaca sebuah bilangan bulat (integer) dari EEPROM. Berikut adalah langkah-langkah yang diambil dalam fungsi tersebut :

1. `(EEPROM.read(address)<< 8) + EEPROM.read(address + 1);` Baris ini membaca data dari EEPROM dengan menggunakan fungsi `EEPROM.read()`. Kita membaca byte pertama dari alamat yang ditentukan, `EEPROM.read(address)`, kemudian menggeser (shift) nilai byte tersebut ke kiri (left shift) sebanyak 8 bit menggunakan operator bitwise shift (`<< 8`).
2. Setelah itu, kita menambahkan hasil dari pergeseran bit tersebut dengan byte kedua yang dibaca dari alamat berikutnya, `EEPROM.read(address + 1)`. Hal ini

dilakukan untuk menggabungkan nilai byte pertama (MSB) dengan byte kedua (LSB) dan menghasilkan bilangan bulat yang akan dikembalikan oleh fungsi.

```
1462 void SimpanTotalUang()  
1463 {  
1464     jumlahUang = readIntFromEEPROM(23);  
1465     jumlahUang = jumlahUang + totalSementara;  
1466     writeIntIntoEEPROM(23, jumlahUang);  
1467 }
```

Gambar 4. 10 *Source Code* Menu Cek Jumlah Fungsi 5

Pada gambar diatas merupakan fungsi `SimpanTotalUang()` Fungsi ini digunakan untuk menyimpan total uang ke dalam EEPROM. Berikut adalah langkah-langkah yang diambil dalam fungsi tersebut :

1. `jumlahUang = readIntFromEEPROM(23);`: Pada baris ini, kita menggunakan fungsi `readIntFromEEPROM()` untuk membaca nilai total uang yang telah disimpan sebelumnya dari alamat EEPROM 23. Nilai yang dibaca kemudian disimpan dalam variabel `jumlahUang`.
2. `jumlahUang = jumlahUang + totalSementara;`: Setelah itu, kita menambahkan nilai `totalSementara` ke dalam `jumlahUang`. Ini bertujuan untuk mengupdate nilai total uang dengan menambahkan nilai sementara yang baru.
3. `writeIntIntoEEPROM(23, jumlahUang);`: Pada baris ini, kita menggunakan fungsi `writeIntIntoEEPROM()` untuk menulis nilai yang telah diubah ke dalam alamat EEPROM 23.

```

1469 void TampilanResetTotalUang()
1470 {
1471     Serial.println("Total Uang Telah Direset");
1472     tft.fillScreen(WHITE);
1473     tft.setTextSize(3);
1474     tft.setTextColor(JADE);
1475     tft.setCursor(28, 124);
1476     tft.println("Total Uang");
1477     tft.setCursor(3, 164);
1478     tft.println("Telah Direset");
1479 }

```

Gambar 4. 11 *Source Code* Menu Cek Jumlah fungsi 6

Pada gambar diatas merupakan fungsi `TampilanResetTotalUang()`. Fungsi ini bertujuan untuk menampilkan pesan bahwa total uang telah direset dan melakukan beberapa pengaturan tampilan pada layar TFT (Thin-Film Transistor). Berikut adalah langkah-langkah yang diambil dalam fungsi tersebut :

1. `Serial.println("Total Uang Telah Direset");`: Pada baris ini, pesan "Total Uang Telah Direset" akan dicetak ke output Serial Monitor. Ini berguna jika Anda menggunakan koneksi serial untuk melihat pesan ini dalam lingkungan pemrograman Arduino atau sejenisnya.
2. `tft.fillScreen(WHITE);`: Baris ini akan mengisi seluruh layar TFT dengan warna putih. `tft` adalah objek yang digunakan untuk mengendalikan tampilan pada layar TFT.
3. `tft.setTextSize(3);`: Pada baris ini, ukuran teks pada layar TFT diatur menjadi 3. Ini menentukan ukuran teks yang akan digunakan untuk teks selanjutnya yang ditampilkan pada layar.

4. `tft.setTextColor(JADE);`: Baris ini mengatur warna teks pada layar TFT menjadi warna "JADE". "JADE" mungkin merupakan konstanta yang didefinisikan sebelumnya dalam program atau kode Anda.
5. `tft.setCursor(28, 124);`: Pada baris ini, posisi kursor teks pada layar TFT diatur ke koordinat (28, 124). Ini menentukan posisi awal di mana teks akan ditampilkan pada layar.
6. `tft.println("Total Uang");`: Baris ini akan mencetak teks "Total Uang" pada layar TFT sesuai dengan posisi kursor yang telah diatur sebelumnya.
7. `tft.setCursor(3, 164);`: Pada baris ini, posisi kursor teks pada layar TFT diatur ke koordinat (3, 164). Ini menentukan posisi awal di mana teks selanjutnya akan ditampilkan pada layar.
8. `tft.println("Telah Direset");`: Baris ini akan mencetak teks "Telah Direset" pada layar TFT sesuai dengan posisi kursor yang telah diatur sebelumnya.

Dengan demikian, fungsi `TampilanResetTotalUang()` akan menampilkan pesan "Total Uang Telah Direset" pada output Serial Monitor (jika ada) dan melakukan pengaturan tampilan pada layar TFT untuk menampilkan teks "Total Uang" dan "Telah Direset".

4.2.3 Layout Kode *Motor Servo*

```
83 int position1=100, position2=140;
```

Gambar 4. 12 *Source Code Motor Servo* fungsi 1

Pada gambar diatas merupakan Source Code yang berfungsi untuk mendefinisikan dua variabel bertipe int, yaitu position1 dan position2, dan memberikan nilai awal masing-masing variabel tersebut. Berikut adalah langkah-langkah yang diambil dalam fungsi tersebut :

1. `int position1 = 100;`: Baris ini mendefinisikan variabel `position1` dengan tipe data int (integer) dan memberikan nilai awal 100 ke variabel tersebut. Variabel `position1` kemungkinan digunakan untuk menyimpan posisi yang akan diberikan ke servo yang terhubung ke servo_28.
2. `int position2 = 140;`: Baris ini mendefinisikan variabel `position2` dengan tipe data int (integer) dan memberikan nilai awal 140 ke variabel tersebut. Variabel `position2` kemungkinan digunakan untuk menyimpan posisi yang akan diberikan ke servo yang terhubung ke servo_30.

Dengan memberikan nilai awal ini dapat diatur posisi awal dari servo motor ketika program dijalankan, dan dapat mengubah nilai `position1` dan `position2` sesuai dengan kebutuhan untuk mengatur posisi servo motor yang diinginkan pada awal program.

```
76 Servo servo;  
77 Servo servo_28;  
78 Servo servo_30;  
79
```

Gambar 4. 13 *Source Code Motor Servo* fungsi 2

Pada gambar diatas mendefinisikan tiga objek Servo dengan nama `servo`, `servo_28`, dan `servo_30`. Objek-objek ini digunakan untuk mengendalikan servo motor pada mikrokontroler atau papan Arduino. Berikut adalah langkah-langkah yang diambil dalam fungsi tersebut :

1. `Servo servo;`: Baris ini mendefinisikan objek servo dengan tipe data `Servo`. Objek ini digunakan untuk mengendalikan sebuah servo motor.
2. `Servo servo_28;`: Baris ini mendefinisikan objek `servo_28` dengan tipe data `Servo`. Objek ini juga digunakan untuk mengendalikan sebuah servo motor.
3. `Servo servo_30;`: Baris ini mendefinisikan objek `servo_30` dengan tipe data `Servo`. Objek ini juga digunakan untuk mengendalikan sebuah servo motor.

Dengan mendefinisikan objek-objek `Servo` ini dapat menggunakan objek-objek tersebut untuk menghubungkan servo motor ke pin tertentu pada mikrokontroler atau papan Arduino, serta menggerakkan servo ke posisi yang diinginkan menggunakan fungsi-fungsi yang disediakan oleh library `Servo`. Misalnya, Anda dapat menggunakan fungsi `attach()` untuk menghubungkan servo ke pin tertentu, dan fungsi `write()` untuk menggerakkan servo ke posisi tertentu.

```
124     servo_28.attach(28);
125     servo_30.attach(30);
126
127     servo_28.write(position1);
128     servo_30.write(position2);
```

Gambar 4. 14 *Source Code Motor Servo* fungsi 3

Pada gambar diatas merupakan Source Code yang berfungsi untuk mengontrol dua servo motor. Mari kita

jelaskan setiap barisnya. Berikut adalah langkah-langkah yang diambil dalam fungsi tersebut :

1. `servo_28.attach(28);`: Pada baris ini, servo dengan nama `servo_28` dihubungkan ke pin digital 28 pada mikrokontroler atau papan Arduino yang digunakan. Fungsi `attach()` digunakan untuk menghubungkan objek servo ke pin yang ditentukan.
2. `servo_30.attach(30);`: Pada baris ini, servo dengan nama `servo_30` dihubungkan ke pin digital 30 pada mikrokontroler atau papan Arduino yang digunakan. Fungsi `attach()` digunakan untuk menghubungkan objek servo ke pin yang ditentukan.
3. `servo_28.write(position1);`: Baris ini mengatur posisi servo `servo_28` sesuai dengan nilai yang ada dalam variabel `position1`. Fungsi `write()` digunakan untuk menggerakkan servo ke posisi tertentu yang diberikan dalam parameter.
4. `servo_30.write(position2);`: Baris ini mengatur posisi servo `servo_30` sesuai dengan nilai yang ada dalam variabel `position2`. Fungsi `write()` digunakan untuk menggerakkan servo ke posisi tertentu yang diberikan dalam parameter.

Dalam kode tersebut, servo motor dihubungkan ke pin digital 28 dan 30, dan kemudian digerakkan ke posisi yang telah ditentukan oleh variabel `position1` dan `position2`. Anda perlu memastikan bahwa pin-pin yang digunakan sesuai dengan koneksi fisik pada mikrokontroler atau papan Arduino yang Anda gunakan, dan juga memastikan bahwa variabel `position1` dan `position2` berisi nilai posisi yang *valid* untuk servo motor yang digunakan.

```

if (r>48 && r<70 && g>102 && g<117 && b>86 && b<110)
{
  tft.fillRoundRect(15, 100, 210, 80, 5, BLACKM);
  tft.fillRoundRect(17, 102, 206, 76, 5, LIMAPULUH);

  tft.setCursor(40, 129);
  tft.setTextSize(3);
  tft.setTextColor(WHITE);
  tft.println("Rp.50.000");
  limapuluh = 50;

  servo_28.write(65);
  delay(3000);
  servo_28.write(position1);
}

```

Gambar 4. 15 Source Code *Motor Servo* fungsi 4

Pada gambar diatas merupakan *Source Code* adalah sebuah blok if yang melakukan pengecekan kondisi pada variabel r, g, dan b. Jika kondisi tersebut terpenuhi, maka blok kode di dalamnya akan dieksekusi. Berikut adalah langkah-langkah yang diambil dalam fungsi tersebut:

1. if (r > 48 && r < 70 && g > 102 && g < 117 && b > 86 && b < 110): Baris ini adalah kondisi yang harus terpenuhi agar blok kode di dalamnya dieksekusi. Kondisi ini memeriksa apakah nilai variabel r berada di antara 48 dan 70, nilai variabel g berada di antara 102 dan 117, serta nilai variabel b berada di antara 86 dan 110.
2. {: Kurung kurawal membuka blok kode yang akan dieksekusi jika kondisi di atas terpenuhi.
3. tft.fillRoundRect(15, 100, 210, 80, 5, BLACKM);: Baris ini mengisi suatu area persegi panjang dengan sudut

- melengkung pada layar TFT menggunakan warna BLACKM. Area yang diisi dimulai dari koordinat (15, 100) dengan lebar 210 piksel dan tinggi 80 piksel.
4. `tft.fillRoundRect(17, 102, 206, 76, 5, LIMAPULUH);`: Baris ini mengisi suatu area persegi panjang dengan sudut melengkung pada layar TFT menggunakan warna LIMAPULUH. Area yang diisi dimulai dari koordinat (17, 102) dengan lebar 206 piksel dan tinggi 76 piksel.
 5. `tft.setCursor(40, 129);`: Baris ini mengatur posisi kursor pada layar TFT ke koordinat (40, 129). Kursor ini akan digunakan untuk menampilkan teks selanjutnya pada layar.
 6. `tft.setTextSize(3);`: Baris ini mengatur ukuran teks pada layar TFT menjadi 3.
 7. `tft.setTextColor(WHITE);`: Baris ini mengatur warna teks pada layar TFT menjadi WHITE.
 8. `tft.println("Rp.50.000");`: Baris ini mencetak teks "Rp.50.000" pada layar TFT sesuai dengan posisi kursor yang telah diatur sebelumnya.
 9. `limapuluh = 50;`: Baris ini mengassign nilai 50 ke variabel limapuluh.
 10. `servo_28.write(65);`: Baris ini menggerakkan servo yang terhubung ke servo_28 ke posisi 65. Fungsi `write()` digunakan untuk menggerakkan servo ke posisi tertentu yang diberikan dalam parameter.
 11. `delay(3000);`: Baris ini menyebabkan program menghentikan eksekusi selama 3000 milidetik (3 detik).
 12. `servo_28.write(position1);`: Baris ini menggerakkan servo yang terhubung ke servo_28 ke posisi yang ditentukan oleh variabel `position1`.

Dengan demikian, jika kondisi pada baris pertama terpenuhi, maka blok kode di dalamnya akan di eksekusi dan *Motor Servo* akan bergerak.

```
1123 else if (r1>150 && r1<175 && g1>166 && g1<190 && b1>135 && b1<177)
1124 {
1125     tft.fillRoundRect(15, 100, 210, 80, 5, BLACKM);
1126     tft.fillRoundRect(17, 102, 206, 76, 5, SERATUS);
1127
1128     tft.setCursor(31, 129);
1129     tft.setTextSize(3);
1130     tft.setTextColor(WHITE);
1131     tft.println("Rp.100.000");
1132     seratus = 100;
1133
1134
1135     servo_30.write(65);
1136     delay(3000);
1137     servo_30.write(position2);
1138 }
```

Gambar 4. 16 Source Code *Motor Servo* fungsi 5

Pada gambar diatas merupakan Source Code yang terdapat blok else if yang memeriksa kondisi lainnya. Berikut adalah langkah-langkah yang diambil dalam fungsi tersebut:

1. Kondisi else if: Ini adalah kondisi kedua yang dievaluasi jika kondisi pertama tidak terpenuhi. Dalam kondisi ini, kita memeriksa apakah nilai r1, g1, dan b1 berada dalam rentang tertentu.
2. Blok kode di dalam kondisi else if akan dieksekusi jika kondisi tersebut terpenuhi.
3. tft.fillRoundRect(): Fungsi ini mengisi suatu area persegi panjang dengan sudut melengkung pada layar TFT. Pada baris ini, kita mengisi area persegi panjang dengan sudut melengkung menggunakan warna BLACKM.

4. `tft.setCursor()`: Fungsi ini mengatur posisi kursor pada layar TFT menggunakan koordinat x dan y.
5. `tft.setTextSize()`: Fungsi ini mengatur ukuran teks pada layar TFT.
6. `tft.setTextColor()`: Fungsi ini mengatur warna teks pada layar TFT.
7. `tft.println()`: Fungsi ini mencetak teks pada layar TFT diikuti dengan karakter baru.
8. `seratus = 100;`: Nilai variabel `seratus` diatur menjadi 100.
9. `servo_30.write()`: Fungsi ini menggerakkan servo yang terhubung ke pin 30 ke posisi 65.
10. `delay()`: Fungsi ini menunda eksekusi program selama 3000 milidetik (3 detik).
11. `servo_30.write(position2);`: Setelah jeda 3 detik, servo yang terhubung ke pin 30 digerakkan ke posisi yang ditentukan oleh variabel `position2`.

Dengan demikian, jika kondisi pada baris kedua terpenuhi, maka blok kode di dalamnya akan di eksekusi dan *Motor Servo* akan bergerak.

```

1140     else
1141     {
1142         tft.fillRoundRect(15, 100, 210, 80, 5, BLACKM);
1143         tft.fillRoundRect(17, 102, 206, 76, 5, WHITE);
1144
1145         tft.setCursor(85, 129);
1146         tft.setTextSize(3);
1147         tft.setTextColor(BLACK);
1148         tft.println("Rp.0");
1149         nol = 0;
1150
1151
1152         servo.write(0);
1153     }
1154

```

Gambar 4. 17 *Source Code Motor Servo* fungsi 6

Pada gambar diatas merupakan *Source Code* terdapat blok `else` yang akan dieksekusi jika kedua kondisi sebelumnya tidak terpenuhi. Berikut adalah langkah-langkah yang diambil dalam fungsi tersebut:

1. `else`: Ini adalah blok yang akan dieksekusi jika kedua kondisi sebelumnya tidak terpenuhi.
2. Blok kode di dalam `else` akan dieksekusi jika kondisi tersebut tidak terpenuhi.
3. `tft.fillRoundRect()`: Fungsi ini mengisi suatu area persegi panjang dengan sudut melengkung pada layar TFT. Pada baris ini, kita mengisi area persegi panjang dengan sudut melengkung menggunakan warna BLACKM.
4. `tft.setCursor()`: Fungsi ini mengatur posisi kursor pada layar TFT menggunakan koordinat x dan y.
5. `tft.setTextSize()`: Fungsi ini mengatur ukuran teks pada layar TFT.
6. `tft.setTextColor()`: Fungsi ini mengatur warna teks pada layar TFT.

7. `tft.println()`: Fungsi ini mencetak teks pada layar TFT diikuti dengan karakter baru.
8. `nol = 0;`: Nilai variabel `nol` diatur menjadi 0.
9. `servo.write(0);`: Fungsi ini menggerakkan servo ke posisi 0.

Dengan demikian, jika kondisi pada baris ketiga terpenuhi, maka blok kode di dalamnya akan eksekusi dan *Motor Servo* tidak akan bergerak.

4.3 Hasil Implementasi

Hasil implementasi disini merupakan hasil dari *prototype* atau perancangan Kotak Penyimpanan Uang (*MoneyBox Plus*) dengan berbagai sistem. Salah satunya sistem sortir ini didapatkan hasil yang di harapkan. *Motor Servo* mampu menggerakkan kotak yang terdapat pada *MoneyBox Plus* dengan akurat dan memberikan respons sesuai kondisi yang telah di tentukan untuk pecahan uang Rp.50.000 dan Rp.100.000. Melalui penggunaan kondisional *if-else*, sistem dapat mengidentifikasi dengan tepat pecahan uang yang dimasukkan ke dalam *MoneyBox Plus*.

Selain itu, hasil uji coba juga memvalidasi kinerja layar TFT yang digunakan untuk menampilkan informasi mengenai pecahan uang yang terdeteksi. Informasi yang ditampilkan dengan jelas dan akurat, memudahkan pengguna dalam memonitor jumlah uang yang telah dimasukkan ke dalam kotak penyimpanan.

Dengan hasil uji coba yang positif ini, dapat disimpulkan bahwa implementasi sensor deteksi warna pada *MoneyBox Plus* telah berhasil dan sesuai dengan tujuan penelitian. Hasil ini memberikan bukti bahwa sistem dapat digunakan sebagai alat bantu penyimpanan uang yang efektif dan efisien

4.3.1 Penggunaan Layar LCD

Pada Kotak Penyimpanan Uang (*MoneyBox Plus*) ini, Modul TFT LCD *Display Shield* digunakan untuk menampilkan informasi yang tersedia pada sistem, dan juga sebagai kontrol utama pada sistem. Modul ini dapat dihubungkan secara langsung pada board Arduino Mega 2560.



Gambar 4. 18 Penggunaan Layar LCD

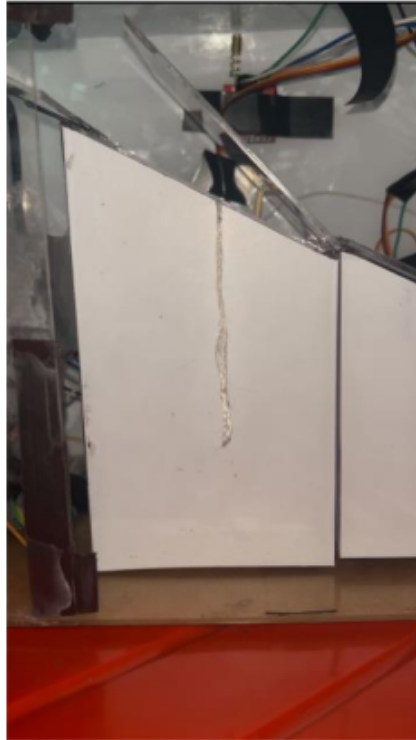
4.3.2 Penggunaan *Motor Servo*

Sistem sortir pada Kotak Penyimpanan Uang (*MoneyBox Plus*) ini menggunakan *Motor Servo* yang berfungsi sebagai penggerak kotak yang terdapat didalam Kotak Penyimpanan Uang (*MoneyBox Plus*). *Motor Servo* ini dapat bergerak secara otomatis untuk dapat memisahkan uang pecahan Rp. 50.000 dan Rp. 100.000 .



Gambar 4. 19 Kotak pecahan uang RP. 100.000 terbuka

Pada gambar diatas memperlihatkan ketika uang pecahan RP. 100.000 terdeteksi maka kotak untuk uang pecahan RP. 100.000 terbuka secara otomatis.



Gambar 4. 20 Kotak pecahan uang RP. 50.000 terbuka

Pada gambar diatas memperlihatkan ketika uang pecahan RP. 50.000 terdeteksi maka kotak untuk uang pecahan RP. 50.000 terbuka secara otomatis.

BAB V

PENUTUP

5.1 Kesimpulan

1

Dari hasil pengujian Rancang Bangun Sistem Keamanan Pada Kotak Penyimpanan Uang (*MoneyBox Plus*) Berbasis Mikrokontroler Arduino dapat disimpulkan :

1. Sistem sortir pada Kotak Penyimpanan Uang (*MoneyBox Plus*) berfungsi dengan baik dan sesuai dengan kondisi *Motor Servo* saat menerima proses dari sistem.
2. *Layout* pada menu cek jumlah uang dapat bekerja dengan baik dan dapat menampilkan jumlah uang yang telah masuk dengan akurat.
3. Sistem sortir pada Kotak Penyimpanan Uang (*MoneyBox Plus*) dapat memudahkan untuk pengambilan uang pada *Money Box Plus* dikarenakan adanya kotak pemisah untuk uang pecahan Rp. 50.000 dan Rp. 100.000

5.2 Saran

Dalam laporan tugas akhir yang telah dibuat oleh penulis, maka penulis mencoba memberikan saran untuk pengembangan alat atau sistem ini kedepannya :

1. Pengembangan Fitur Tambahan: *MoneyBox Plus* dapat terus dikembangkan dengan menambahkan fitur-fitur tambahan yang dapat meningkatkan fungsionalitas dan kenyamanan pengguna. Misalnya, integrasi dengan sistem keamanan tambahan seperti pengenalan wajah atau sidik jari untuk mengamankan akses ke kotak penyimpanan.
2. Peningkatan Akurasi dan Kecepatan: Untuk meningkatkan performa *MoneyBox Plus*, dapat dilakukan penelitian lebih lanjut dalam mengoptimalkan algoritma pengolahan citra digital dan pemrosesan

sensor warna. Hal ini dapat menghasilkan tingkat akurasi yang lebih tinggi dan waktu respons yang lebih cepat dalam mendeteksi dan mengenali pecahan uang.

3. Pengembangan Aplikasi *Mobile*: Mengembangkan aplikasi mobile yang terhubung dengan *MoneyBox Plus* dapat memberikan kemudahan bagi pengguna dalam memantau jumlah uang yang disimpan, melacak riwayat transaksi, atau mengatur pengaturan khusus. Aplikasi ini dapat memanfaatkan konektivitas Bluetooth atau Wi-Fi untuk berkomunikasi dengan *MoneyBox Plus*.
4. Skalabilitas: *MoneyBox Plus* dapat dirancang dengan kemampuan skalabilitas, sehingga pengguna dapat menambahkan modul atau komponen tambahan sesuai dengan kebutuhan. Misalnya, menambahkan modul sensor tambahan untuk mendeteksi dan memisahkan koin, atau memperluas kapasitas penyimpanan uang dengan modul tambahan.
5. Pengembangan Model Bisnis: *MoneyBox Plus* dapat dijadikan sebagai produk yang dapat dipasarkan secara komersial. Dalam bab ini, dapat diberikan saran tentang strategi pemasaran, analisis pasar, dan potensi kerjasama dengan lembaga keuangan atau toko retail untuk memperluas distribusi dan meningkatkan adopsi produk.

Saran-saran tersebut diharapkan dapat membantu dalam mengembangkan *MoneyBox Plus* menjadi produk yang lebih baik, lebih inovatif, dan dapat memberikan manfaat yang lebih besar bagi pengguna.

DAFTAR PUSTAKA

- Aguskhumaidi. (2019, september 5). *Mikrokontroler Arduino*. Retrieved from Agus Khumaidi, S.ST.,M.T.: <https://lecturer.ppns.ac.id/aguskhumaidi/2019/09/05/mikrokontroler-arduino/>
- Aji, H. (2018, Desember 16). *Pengertian, Perbedaan Power Supply AT dan ATX*. Retrieved from IlmuOnline.net: <https://ilmuonline.net/pengertian-perbedaan-power-supply-at-dan-atx/>
- Anantajaya, I. M. (2021). REVIEW APLIKASI SENSOR PADA SISTEM . *Jurnal SPEKTRUM*, vol 8.
- ARIANSYAH, R. (2019). RANCANG BANGUN ALAT SORTIR JERUK NIPIS.
- Budiarso, Z. (2022). Penyortir Bola Berwarna Berbasis Arduino Menggunakan Metode Fuzzy. *Jurnal Teknik Komputer AMIK BSI*.
- Chairul Gunawan, F. N. (2021). Prototipe Light Meter Fotografi Studio Menggunakan Mikrokontroler . *JURNAL MEDIA INFORMATIKA BUDIDARMA*, 769-778.
- Dewi Lestari, M. H. (2019). RANCANG BANGUN SISTEM SORTIR BERDASARKAN WARNA . *jurutera*, 15.
- DIMAS K, R. D. (2018). PENYORTIR UANG KOIN BERBASIS ARDUINO UNO. *Universitas Gajah Mada*.
- ditempel. (2021, maret 19). *Menggunakan Modul LCD 16 x 2*. Retrieved from ditempel.com: <https://www.ditempel.com/2021/03/menggunakan-modul-lcd-16-x-2.html>

- Februariyant, H. (2022). Penyortir Bola Berwarna Berbasis Arduino Menggunakan Metode Fuzzy. *Jurnal Teknik Komputer AMIK BSI*.
- Gramedia, B. (2021, Agustus 21). *Pengertian Uang: Fungsi, Ragam, dan Teori Nilai Uang*. Retrieved from Pengertian Apa Itu Uang: https://www.gramedia.com/literasi/uang/#Pengertian_Uang_-_Apa_itu_Uang
- Gudang, M. (2022, 06 19). *Pengertian dan Cara Sortir Barang dengan Mudah di Gudang*. Retrieved from Sortir Barang adalah: <https://crewdible.com/edukasi/topik/manajemen-gudang/sortir-barang-adalah#:~:text=Sortir%20barang%20adalah%20kegiatan%20pemisahan,seperti%20pemisahan%20sesuai%20alamat%20tujuan>.
- Hanafi, M. H. (2019). Rancang Bangun Sistem Sortir Berdasarkan Warna Permen. *jurutera*.
- Ikhsan, M. A., Yahya, M., & Fiolana, F. A. (2018). Pendeteksi Kekeruhan Air Di Tandon Rumah Berbasis Arduino Uno. *Jurnal Qua Teknika, Vol. 8 No. 2*, 17-29.
- Irvawansyah, I. (2020). Rancang Bangun Media Pembelajaran Penyortiran Benda Berbasis Mikrokontroler. *Joule (Jurnal of Elictrical Engineering)*.
- Jufriyanto, J. (2020). Rancang Bangun Media Pembelajaran Penyortiran Benda Berbasis Mikrokontroler. *Joule (Journal of Electrical Engineering)*.
- Lestari, D. (2019). Rancang Bangun Sistem Sortir Berdasarkan Warna Permen. *jurutera*.

- Listiyono, H. (2022). Penyortir Bola Berwarna Berbasis Arduino Menggunakan Metode Fuzzy. *Jurnal Teknik Komputer AMIK BSI*.
- Mustafa, S. (2020). Rancang Bangun Media Pembelajaran Penyortiran Benda Berbasis Mikrokontroler. *joule (Jurnal of Elictrical Engineering)*.
- Nanda, T. R. (2018). Perancangan Sistem Sortir Buah Kopi Berdasarkan Warna Dengan Teknik Citra Digital Berbasis Mikrokontroler Atmega 328p. *KITEKTRO*.
- Ningsih. (2022). *Apa Yang Dimaksud Dengan Sensor? : Pengertian, Fungsi, Jenis-Jenis, Faktor dan Contoh Penerapan*. IDMETSFORA.
- Purnomo. (2017). Model Prototyping Pada Pengembangan Sistem Informasi. *Jurnal Informatika Merdeka pasuruan*, 55.
- Rijali, M., & Khana, J. R. (2020). Bangun Sistem Pemantauan Dan Pengendalian Cairan Infus Melalui Display Kontrol Dan Aplikasi Mobile Di Masa Pandemic Covid-19. *Ejournal Kajian Teknik Elektro Vol.5 No.1*, 1-21.
- Saputro, R. A. (2022). Penyortir Bola Berwarna Berbasis Arduino Menggunakan Metode Fuzzy. *Jurnal Teknik Komputer AMIK BSI*.
- Setiawan, R. (2021, Agustus 4). *Flowchart Adalah: Fungsi, Jenis, Simbol, dan Contohnya*. Retrieved from Dicoding: <https://www.dicoding.com/blog/flowchart-adalah/>
- Simatupang, J. W., Prasetyo, B., Galina, M., & Suhartomo, A. (2022). Prototipe Mesin Penjual Air Mineral Otomatis berbasis Arduino Mega 2560 dan RFID-RC522. *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik*

Telekomunikasi, & Teknik Elektronika Vol. 10 No. 2, 484-499.

- Syaryadh, M. (2018). Perancangan Sistem Sortir Buah Kopi Berdasarkan Warna Dengan Teknik Citra Digital Berbasis Mikrokontroler Atmega 328p. *KITEKTRO*.
- T. W. Wisjhnuadji, A. N. (2020). Sistem Sortir Barang Otomatis Berbasis Arduino dengan Sensor . *Faktor Exacta*, 106.
- Yohanes C Saghoa, S. R. (2018). Kotak Penyimpanan Uang Berbasis . *Jurnal Teknik Elektro dan Komputer* , 167.
- Zulhelmi, Z. (2018). Perancangan Sistem Sortir Buah Kopi Berdasarkan Warna Dengan Teknik Citra Digital Berbasis Mikrokontroler Atmega 328p. *KITEKTRO*.
- Zulkarnain, M. (2020). Rancang Bangun Media Pembelajaran Penyortiran Benda Berbasis Mikrokontroler. *joule (Jurnal of Elictrical Engineering)*.

LAMPIRAN

TUGAS AKHIR MUH DEVANGGA HAMDAN MAULANA (19120042)

ORIGINALITY REPORT

13%

SIMILARITY INDEX

13%

INTERNET SOURCES

3%

PUBLICATIONS

8%

STUDENT PAPERS

PRIMARY SOURCES

1	erepository.uwks.ac.id Internet Source	6%
2	Submitted to Universitas Wijaya Kusuma Surabaya Student Paper	2%
3	text-id.123dok.com Internet Source	1%
4	docplayer.info Internet Source	1%
5	www.mahirelektro.com Internet Source	1%
6	123dok.com Internet Source	1%
7	ojs.trigunadharma.ac.id Internet Source	1%
8	id.wikipedia.org Internet Source	1%
9	smart.stmikplk.ac.id Internet Source	1%

10

repository.president.ac.id
Internet Source

1 %

Exclude quotes On

Exclude matches < 1%

Exclude bibliography On

TUGAS AKHIR MUH DEVANGGA HAMDAN MAULANA (19120042)

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13

PAGE 14

PAGE 15

PAGE 16

PAGE 17

PAGE 18

PAGE 19

PAGE 20

PAGE 21

PAGE 22

PAGE 23

PAGE 24

PAGE 25

PAGE 26

PAGE 27

PAGE 28

PAGE 29

PAGE 30

PAGE 31

PAGE 32

PAGE 33

PAGE 34

PAGE 35

PAGE 36

PAGE 37

PAGE 38

PAGE 39

PAGE 40

PAGE 41

PAGE 42

PAGE 43

PAGE 44

PAGE 45

PAGE 46

PAGE 47

PAGE 48

PAGE 49

PAGE 50

PAGE 51

PAGE 52

PAGE 53

PAGE 54

PAGE 55

PAGE 56

PAGE 57

PAGE 58

PAGE 59

PAGE 60

PAGE 61

PAGE 62

PAGE 63

PAGE 64

PAGE 65

PAGE 66

PAGE 67

PAGE 68

PAGE 69

PAGE 70

PAGE 71

PAGE 72

PAGE 73

PAGE 74

PAGE 75

PAGE 76

PAGE 77

PAGE 78

PAGE 79

PAGE 80

PAGE 81

PAGE 82

PAGE 83

PAGE 84

PAGE 85

PAGE 86

PAGE 87

PAGE 88
