



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



**RANCANG BANGUN SISTEM *MONITORING* DAN
NOTIFIKASI PEBERSIH KACA JENDELA
LABORATORIUM BERBASIS *INTERNET OF THINGS* (IoT)
TERINTEGRASI APLIKASI *ANDROID***

**“RANCANG BANGUN SISTEM MIKROKONTROLER DAN
PROTOTYPE PEBERSIH KACA JENDELA LABORATORIUM”**

**POLITEKNIK
NEGERI
JAKARTA**

SYIFA EKA MULIANDARI

1803332013

**PROGRAM STUDI TEKNIK TELEKOMUNIKASI
JURUSAN TEKNIK ELEKTRO
POLITEKNIK NEGERI JAKARTA
JULI 2021**



HALAMAN PERNYATAAN ORISINALITAS

Tugas Akhir ini adalah hasil karya saya sendiri dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.

Nama : Syifa Eka Muliandari

NIM : 1803332013

Tanda Tangan : *Syifa*

Tanggal : 17 Juli 2021

POLITEKNIK
NEGERI
JAKARTA

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



LEMBAR PENGESAHAN TUGAS AKHIR

Tugas Akhir diajukan oleh :

Nama : Syifa Eka Muliandari

NIM : 1803332013

Program Studi : Telekomunikasi

Judul Tugas Akhir : Rancang Bangun Sistem *Monitoring* dan Notifikasi Pembersih Kaca Jendela Laboratorium Berbasis *Internet of Things* (IoT) Terintegrasi Aplikasi *Android*

Telah diuji oleh tim penguji dalam Sidang Tugas Akhir pada Tanggal 29 Juli 2021 dan dinyatakan **LULUS**.

Pembimbing : Yenniwarti Rafsyam, SST., M.T.

NIP. 19680627 199303 2 002

()

Depok, 23 Agustus 2021

Disahkan oleh

Ketua Jurusan Teknik Elektro

Ir. Sri Danaryani, M.T.

NIP. 196305031991032001

- Hak Cipta :**
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
 2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummikan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

KATA PENGANTAR

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, penulis dapat menyelesaikan Tugas Akhir ini. Penulisan Tugas Akhir ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Diploma Tiga Politeknik.

Tugas Akhir ini yang berjudul “Rancang Bangun Sistem *Monitoring* dan Notifikasi Pembersih Kaca Jendela Laboratorium Berbasis *Internet of Things* (IoT) Terintegrasi Aplikasi *Android*” dapat membantu kegiatan pembersihan serta pengawasan kebersihan jendela pada laboratorium.

Penulis menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan tugas akhir ini, sangatlah sulit bagi penulis untuk menyelesaikan tugas akhir ini. Oleh karena itu, penulis mengucapkan terima kasih kepada :

1. Ibu Yenniwati Rafsyam, SST., M.T. selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan penulis dalam penyusunan Tugas Akhir ini;
2. Seluruh Staf Pengajar dan Karyawan Jurusan Teknik Elektro Politeknik Negeri Jakarta, khususnya Program Studi Telekomunikasi
3. Orang tua dan keluarga penulis yang telah memberikan bantuan dukungan material dan moral;
4. Dian Chairunnisa selaku rekan Tugas Akhir saya yang sangat baik dan telah banyak membantu saya selama pengerjaan Tugas Akhir ini;
5. Sahabat yang telah banyak membantu penulis dalam menyelesaikan Tugas Akhir ini.

Akhir kata, penulis berharap Tuhan Yang Maha Esa berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga laporan PKL ini membawa manfaat bagi pengembangan ilmu.

Depok, 17 Juli 2021

Penulis



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumikan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Rancang Bangun Sistem *Monitoring* dan Notifikasi Pembersih Kaca Jendela Laboratorium Berbasis *Internet of Things* (IoT) Terintegrasi Aplikasi *Android*

Abstrak

Kebersihan merupakan salah satu faktor utama yang memberikan dampak bagi kesehatan manusia. Namun sampai saat ini masih banyak kelalaian manusia dalam menjaga kebersihan dan seringkali lalai dalam pengawasannya terutama diruangan tertutup, salah satu contohnya adalah pada laboratorium. Pada tugas akhir ini dirancang sistem monitoring dan notifikasi pembersih kaca jendela laboratorium berbasis *Internet of Things* (IoT) terintegrasi aplikasi *Android*. Sistem ini bertujuan untuk membantu petugas kebersihan dalam memantau serta membersihkan kaca jendela laboratororium. Perancangan sistem menggunakan *optical dust sensor*, modul relay, driver motor L298N, dan motor DC. Sumber tegangan berasal dari catu daya yang menggunakan transformator sebesar 3A dengan tegangan masukan sebesar 220 VAC dan tegangan keluaran catu daya sebesar 12 VDC dan 9VDC. Pembersih kaca jendela laboratorium ini dapat diimplementasikan pada kaca jendela dengan tinggi 77 cm dan lebar 20 cm. Rel sliding digunakan sebagai jalur wiper untuk bergerak keatas dan kebawah dalam membersihkan kaca jendela. Hasil pengujian menunjukkan, saat intensitas debu berada diatas $1,5 \text{ mg/m}^3$ secara otomatis sistem bekerja membersihkan kaca jendela dari bawah menuju ke atas dan kondisi kebersihan jendela dipantau setiap 1 jam sekali, dimana kamera mengirimkan gambar ke Aplikasi *Android*. Sistem yang dibangun berfungsi dengan baik sebagai alat monitoring pembersih kaca jendela laboratorium.

Kata kunci : *Arduino Mega, ESP32, ESP32-Cam, Pembersih Kaca, Optical Dust Sensor, Relay, Driver Motor L298N, Motor DC.*



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Design and Build a Monitoring and Notification System of Window Glass Cleaning Laboratory Based on Internet of Things (IoT) Integrated Android Application

Abstract

Cleanliness is one of the main factors that have an impact on human health. However, until now there are still many human negligence in maintaining cleanliness and often negligent in its supervision, especially in a closed room, one example is in the laboratory. In this final project, a monitoring and notification system for window cleaners based on Internet of Things (IoT) is designed with an Android application integrated. This system aims to assist janitors in monitoring and cleaning laboratory windows. The system design uses optical dust sensor, relay module, L298N motor driver, and DC motor. The voltage source comes from a power supply that uses a transformer of 3A with an input voltage of 220 VAC and an output voltage of 12 VDC and 9VDC. This laboratory window cleaner can be implemented on window panes with a height of 77 cm and a width of 20 cm. The sliding rail is used as a wiper path to move up and down in cleaning the window glass. The test results show, when the dust intensity is above 1.5 mg/m³ the system automatically works to clean the window glass from the bottom to the top and the condition of the window cleanliness is monitored every 1 hour, where the camera sends an image to the Android application. The system that was built functions well as a monitoring tool for laboratory window cleaning.

Key words : *Arduino Mega, ESP32, ESP32-Cam, Glass Cleaner, Optical Dust Sensor,, Relay, L298N Motor Driver, DC Motor.*



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

DAFTAR ISI

	Halaman
HALAMAN SAMBUT.....	i
HALAMAN JUDUL	ii
HALAMAN PERNYATAAN ORISINALITAS	iii
LEMBAR PENGESAHAN TUGAS AKHIR	iv
KATA PENGANTAR.....	v
ABSTRAK	vi
ABSTRACT	vii
DAFTAR ISI.....	viii
DAFTAR GAMBAR.....	x
DAFTAR TABEL	xii
DAFTAR LAMPIRAN	xiii
BAB I PENDAHULUAN	2
1.1 Latar Belakang.....	2
1.2 Rumusan Masalah	4
1.3 Tujuan.....	4
1.4 Luaran.....	4
BAB II TINJAUAN PUSTAKA	3
2.1 <i>Internet of Things</i> (IoT).....	3
2.2 Arduino Mega.....	3
2.3 ESP32-Cam	5
2.4 ESP32	5
2.5 Motor DC.....	6
2.6 <i>Driver Motor L298N</i>	6
2.7 <i>Optical Dust Sensor</i> (GP2Y1010AU0F)	7
2.7 <i>Relay</i>	7
2.8 Arduino IDE	8
2.9 EAGLE.....	9
BAB III PERANCANGAN DAN REALISASI	10
3.1 Rancangan Alat.....	10
3.1.1 Deskripsi Alat.....	10
3.1.2 Cara Kerja Alat.....	11
3.2 Perancangan dan Pemrograman Sistem.....	14



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

3.2.1. Perancangan Sistem <i>Monitoring</i> Pembersih Kaca Jendela Laboratorium Menggunakan Arduino, ESP32, dan ESP32-Cam.....	14
3.2.2. Perancangan Catu Daya (<i>Power Supply</i>).....	18
3.2.3. Perancangan Modul <i>Power Bank</i>	19
3.2.4. Pemrograman Arduino Mega	20
3.2.5. Pemrograman ESP32	29
3.3 Realisasi Alat.....	35
3.3.1 Realisasi Catu Daya.....	35
3.3.2 Realisasi Modul <i>Power Bank</i>	36
3.3.3 Realisasi Program	37
BAB IV PEMBAHASAN	39
4.1 Pengujian	39
4.1.1 Pengujian Catu Daya (<i>Power Supply</i>)	39
4.1.2 Pengujian Modul <i>Power Bank</i>	44
4.1.3 Pengujian Program Arduino IDE.....	47
4.1.3.1 Pengujian <i>Optical Dust Sensor</i>	48
4.1.3.2 Pengujian Kamera ESP32-Cam	53
BAB V PENUTUP	57
5.1 Simpulan.....	57
5.2 Saran	57
DAFTAR PUSTAKA	58
DAFTAR RIWAYAT HIDUP	60
LAMPIRAN	xiv

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Arduino Mega	4
Gambar 2.2 ESP32-Cam	5
Gambar 2.3 ESP32	5
Gambar 2.4 Motor DC	6
Gambar 2.5 Driver Motor L298N	7
Gambar 2.6 Optical Dust Sensor	7
Gambar 2.7 Relay.....	8
Gambar 2.8 Arduino IDE.....	8
Gambar 2.9 EAGLE.....	9
Gambar 3.1 Ilustrasi Kerja Sistem Monitoring dan Notifikasi Pembersih Kaca Jendela Laboratorium Berbasis IoT Terintegrasi Aplikasi Android ..	11
Gambar 3.2 Diagram Blok Sistem Monitoring dan Notifikasi Pembersih Kaca Jendela Laboratorium Berbasis IoT Terintegrasi Aplikasi Android ..	11
Gambar 3.3 Flowchart Sistem Monitoring dan Notifikasi Pembersih Kaca Jendela Laboratorium.....	12
Gambar 3.4 Rancangan Optical Dust Sensor dengan Arduino Mega.....	16
Gambar 3.5 Rancangan Modul Relay dengan Arduino Mega	17
Gambar 3.6 Rancangan Driver Motor L298N Pada Sistem	18
Gambar 3.7 Rangkaian Skematik Catu Daya.....	18
Gambar 3.8 Rangkaian Skematik Modul Power Bank	20
Gambar 3.9 Layout Rangkaian Catu Daya	36
Gambar 3.10 Rangkaian Catu Daya yang Dibuat.....	36
Gambar 3.11 Layout Rangkaian Modul Power Bank	37
Gambar 3.12 Rangkaian Modul Power Bank yang Dibuat.....	37
Gambar 3.13 Realisasi Program.....	38
Gambar 4.1 Set Up Rangkaian Catu Daya.....	40
Gambar 4.2 Hasil Pengukuran Tegangan Keluaran Catu Daya 12,05V	41
Gambar 4.3 Hasil Pengukuran Tegangan Keluaran Catu Daya 9,10V	41
Gambar 4.4 Tegangan Keluaran Trafo.....	42
Gambar 4.5 Tegangan Keluaran Dioda Bridge.....	42
Gambar 4.6 Tegangan Keluaran IC Regulator 7812.....	43
Gambar 4.7 Tegangan Keluaran IC Regulator 7809.....	43
Gambar 4.8 Set Up Rangkaian Modul Power Bank	45
Gambar 4.9 Tegangan Keluaran Baterai	46
Gambar 4.10 Tegangan Keluaran Modul Power Bank	46
Gambar 4.11 Proses Upload Program Arduino IDE Berhasil.....	48
Gambar 4.12 Takaran Bubuk 1/4 sdt Untuk Optical Dust Sensor.....	49
Gambar 4.13 Pengujian Optical Dust Sensor Pertama	49
Gambar 4.14 Takaran Bubuk 1/2 sdt Untuk Optical Dust Sensor.....	50
Gambar 4.15 Pengujian Optical Dust Sensor Kedua	50
Gambar 4.16 Takaran Bubuk 1 sdt Untuk Optical Dust Sensor.....	50
Gambar 4.17 Pengujian Optical Dust Sensor Ketiga.....	50



Gambar 4.18 Hasil Pembacaan <i>Optical Dust Sensor</i> yang Mengoperasikan Pompa Air Otomatis	51
Gambar 4.19 Hasil Pembacaan <i>Optical Dust Sensor</i> yang Mengoperasikan Wiper Otomatis	51
Gambar 4.20 <i>Serial Monitor</i> Pembacaan <i>Optical Dust Sensor</i> Kaca Bersih	52
Gambar 4.21 <i>Serial Monitor</i> Pembacaan <i>Optical Dust Sensor</i> Kaca Kotor	52
Gambar 4.22 <i>Serial Monitor</i> Kamera Menangkap Gambar Kaca Jendela	53
Gambar 4.23 Pengujian ESP32-Cam Pertama	53
Gambar 4.24 Pengujian ESP32-Cam Kedua	54
Gambar 4.25 Pengujian ESP32-Cam Ketiga	54



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummikan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummikan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

DAFTAR TABEL

	Halaman
Tabel 3.1 Penggunaan <i>Pin</i> Arduino Mega	15
Tabel 3.2 Penggunaan <i>Pin</i> ESP32.....	15
Tabel 4.1 Hasil Pengukuran Catu Daya	43
Tabel 4.2 Hasil Pengukuran Modul <i>Power Bank</i>	46
Tabel 4.3 Rentang Nilai <i>Optical Dust Sensor</i> Taburan Bubuk.....	48
Tabel 4.4 Tabel Hasil Pengujian <i>Optical Dust Sensor</i>	48





Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kebersihan merupakan salah satu hal yang wajib diperhatikan dan dijaga, karena kebersihan dapat memberikan dampak bagi kesehatan manusia. Tetapi sampai saat ini masih banyak dijumpai ruangan tertutup yang masih kurang dalam pengawasan kebersihannya, salah satunya pada laboratorium. Walaupun sudah terdapat beberapa petugas yang memiliki tanggung jawab untuk membersihkan laboratorium secara berkala, namun masih ada petugas yang lalai dalam melakukan tugasnya. Padahal laboratorium sendiri merupakan tempat yang hampir setiap hari didatangi banyak orang yang akan melakukan kegiatan praktikum.

Kualitas kebersihan laboratorium tentunya dapat membawa dampak yang kurang baik bagi setiap orang yang mendatangi laboratorium tersebut, misalnya dengan keadaan kaca jendela laboratorium yang berdebu dan banyak sawang. Debu dapat mengakibatkan beberapa dampak negatif bagi manusia, antara lain alergi, membuat mata gatal, batuk, pilek, dan lain – lain.

Berdasarkan hal tersebut, pengusul membuat sistem monitoring pembersih kaca jendela laboratorium berbasis *Internet of Things* yang terintegrasi aplikasi android. Sistem ini memudahkan petugas yang ingin membersihkan jendela laboratorium dengan adanya *wiper* yang dapat bekerja secara otomatis ketika intensitas debu sudah tinggi dan juga petugas dapat memberikan perintah melalui aplikasi *android*. Foto kondisi kaca juga akan diterima petugas secara otomatis melalui aplikasi *android*. Maka dari itu, pada tugas akhir ini akan dibuat Rancang Bangun Sistem *Monitoring* dan Notifikasi Pembersih Kaca Jendela Laboratorium Berbasis *Internet of Things* (IoT) Terintegrasi Aplikasi *Android*.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

1.2 Rumusan Masalah

Berdasarkan latar belakang yang diuraikan di atas, maka permasalahan yang akan dibahas dalam tugas akhir ini adalah sebagai berikut :

1. Bagaimana merancang sistem mikrokontroler dan *prototype* pembersih kaca jendela laboratorium?
2. Bagaimana membangun sistem pembersih kaca jendela laboratorium?
3. Bagaimana mengaplikasikan sistem untuk pembersih kaca jendela laboratorium?

1.3 Tujuan

Tujuan yang ingin dicapai dalam tugas akhir ini adalah :

1. Mampu merancang sistem mikrokontroler dan *prototype* pembersih kaca jendela laboratorium.
2. Mampu membangun sistem pembersih kaca jendela laboratorium.
3. Mampu mengaplikasikan sistem untuk pembersih kaca jendela laboratorium.

1.4 Luaran

Luaran yang ingin dicapai dari tugas akhir ini adalah menghasilkan perancangan dari sistem mikrokontroler dan *prototype* pembersih kaca jendela laboratorium yang dapat digunakan oleh petugas kebersihan laboratorium. Selain itu juga adanya jurnal mengenai judul Tugas Akhir yang diangkat oleh penulis.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

BAB V

PENUTUP

5.1 Simpulan

Berdasarkan pembahasan mengenai Rancang Bangun Sistem *Monitoring* dan Notifikasi Pembersih Kaca Jendela Laboratorium Berbasis Internet of Things (IoT) Terintegrasi Aplikasi *Android* dengan sub-judul “Rancang Bangun Sistem Mikrokontroler dan *Prototype* Pembersih Kaca Jendela Laboratorium” diperoleh beberapa simpulan :

1. Pembersih kaca jendela laboratorium dirancang menggunakan mikrokontroler Arduino Mega yang terhubung dengan *optical dust sensor*, *relay*, dan *driver motor L298N*, kemudian terhubung dengan ESP32 untuk pertukaran data yang akan dikirimkan ke internet, dan juga menggunakan *ESP32-Cam*.
2. Sistem pembersih kaca jendela laboratorium menggunakan *optical dust sensor* yang harus bernilai diatas $1,5 \text{ mg/m}^3$ untuk dapat mengoperasikan pembersih kaca jendela laboratorium secara otomatis dan menggunakan *ESP32-Cam* untuk menangkap gambar keadaan kaca setiap 1 jam sekali.
3. Pembersih kaca jendela laboratorium bersifat *portable* dan dapat diimplementasikan secara baik pada kaca jendela dengan tinggi 77 cm x 20 cm.

5.2 Saran

Diharapkan Tugas Akhir yang berjudul Rancang Bangun Sistem Monitoring dan Notifikasi Pembersih Kaca Jendela Laboratorium Berbasis *Internet of Things* (IoT) Terintegrasi Aplikasi *Android* ini dapat bermanfaat dan bisa ditambahkan dengan sistem lainnya yang dapat memaksimalkan kinerja pembersih kaca jendela laboratorium.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

DAFTAR PUSTAKA

- Azro, Isnaini. 2015. *Alat Pembersih Kaca Otomatis Pada Gedung Bertingkat Berbasis Mikrokontroler ATMEGA 8535*. Jurnal Penelitian Ilmu dan Teknologi Komputer : Vol. 7, No. 2.
- Gunawan, Indra. 2020. *Prototipe Penerapan Internet Of Things (Iot) Pada Monitoring Level Air Tandon Menggunakan Nodemcu Esp8266*. Jurnal Informatika dan Teknologi : Vol. 3 No. 1, Januari 2020, hal 1 – 7.
- Immersa-lab. 2018. PENGERTIAN RELAY, FUNGSI, DAN CARA KERJA RELAY. <https://www.immersa-lab.com/pengertian-relay-fungsi-dan-cara-kerja-relay.htm>. [diakses pada 30 Januari 2021]
- Labelektronika. 2017. ARDUINO MEGA 2560 MIKROKONTROLER ATmega2560. <http://www.labelektronika.com/2017/02/arduino-mega-2560-mikrokontroler.html>. [diakses pada 30 Januari 2021]
- Kho, Dickson. Tanpa Tahun. Pengertian Motor DC dan Prinsip Kerjanya. <https://teknikelektronika.com/pengertian-motor-dc-prinsip-kerja-dc-motor/>. [diakses pada 28 Januari 2021]
- Kho, Dickson. Tanpa Tahun. Pengertian Relay dan Fungsinya. <https://teknikelektronika.com/pengertian-multivibrator-monostabil-monostable-multivibrator-cara-kerjanya/>. [diakses pada 28 Januari 2021]
- Kusumah, Hendra., Pradana, Restu Adi. 2019. *PENERAPAN TRAINER INTERFACING MIKROKONTROLER DAN INTERNET OF THINGS BERBASIS ESP32 PADA MATA KULIAH INTERFACING*. Jurnal Sistem Komputer : Vol. 5 No. 2.
- Nyabarilmu. 2017. Tutorial Arduino mengakses driver motor L298N. <https://www.nyabarilmu.com/tutorial-arduino-mengakses-driver-motor-l298n/>. [diakses pada 9 Agustus 2021]

- Oneguyoneblog. 2019. ESP32-CAM: ESP32 WITH CAMERA AND SD CARD SLOT. <https://oneguyoneblog.com/2019/09/09/esp32-cam-esp32-camera-sd-card-slot/>. [diakses pada 23 Januari 2021]
- Razor, Aldy. Tanpa Tahun. Software Arduino IDE: Cara Download, Instal, dan Fungsinya. <https://www.aldyrazor.com/2020/05/software-arduino-ide.html>. [diakses pada 22 Juli 2021]
- Wisnulaksito, Febri Aji. Sari, Marti Widya., Tentua, Meilany Nonsi. 2017. *RANCANG BANGUN SISTEM PENDETEKSI DEBU BERBASIS ARDUINO*. Jurnal Dinamika Informatika : Vol. 6, No. 2.



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

LAMPIRAN



Proses Pemotongan Plat Untuk *Casing*



Proses Pengeboran PCB Untuk Catu Daya



Proses *Wiring* Alat

```

#include <ArduinoJson.h>

#define measurePin A4
#define measurePin2 A3
#define ledPower 12
#define ledPower2 13

unsigned int samplingTime = 280;
unsigned int deltaTime = 40;
unsigned int sleepTime = 9680;

float voMeasured = 0;
float calcVoltage = 0;
float dustDensity = 0;
float voMeasured2 = 0;
float calcVoltage2 = 0;
float dustDensity2 = 0;
float dustTotal = 0;
//kalkulasi awal 0

// Motor A connections
#define enA 9
#define in1 8
#define in2 7

#define relayPompa 6

String kacaStat;

void setup() {
  Serial.begin(9600);

```

© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

Serial1.begin(115200);
pinMode(ledPower, OUTPUT);
pinMode(ledPower2, OUTPUT);
pinMode(enA, OUTPUT);
pinMode(in1, OUTPUT);
pinMode(in2, OUTPUT);
digitalWrite(in1, LOW);
digitalWrite(in2, LOW);

pinMode(relayPompa, OUTPUT);
digitalWrite(relayPompa, HIGH);

kacaStat = "bersih";
}

void loop() {
  static uint8_t flag = 0, dusttotstat = 0, dust1stat = 0, dust2stat = 0;
  static uint32_t millisDust, timeDust = 1000;

  // check if data is available
  if (Serial1.available() > 0 && flag == 0) {
    // read the incoming string:
    String incomingString = Serial1.readStringUntil('\n');

    // prints the received data
    //Serial.print("Arduino received: ");
    //Serial.println(incomingString);

    if (incomingString == "*w0p0#") {
      Serial.println("BUTTON WIPER OFF, BUTTON POMPA OFF");
    }
  }
}

```



```

delayMicroseconds(deltaTime);

digitalWrite(ledPower, HIGH); // turn the LED off
digitalWrite(ledPower2, HIGH); // turn the LED off
delayMicroseconds(sleepTime);

calcVoltage = voMeasured * (5.0 / 1024.0);
calcVoltage2 = voMeasured2 * (5.0 / 1024.0);

dustDensity = 1.7 * calcVoltage - 0.1;
dustDensity2 = 1.7 * calcVoltage2 - 0.1;
dustTotal = dustDensity + dustDensity2;

Serial.print("DUST DENSITY SENSOR KE-1: ");
Serial.println(dustDensity); // unit: ug/m3
Serial.print("DUST DENSITY SENSOR KE-2: ");
Serial.println(dustDensity2); // unit: ug/m3

if (dustDensity <= 1.5) {
  //Serial.println("flag = 0");
  //kacaStat = "bersih";
  dust1stat = 0;
  //flag = 0;
} else {
  //Serial.println("flag = 1");
  //kacaStat = "kotor";
  dust1stat = 1;
  //flag = 1;
}

//delay(10);

if (dustDensity2 <= 1.5) {

```

© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

//Serial.println("flag = 0");
//kacaStat = "bersih";
dust2stat = 0;
//flag = 0;
} else {
//Serial.println("flag = 1");
//kacaStat = "kotor";
dust2stat = 1;
//flag = 1;
}
//delay(10);

dusttotstat = dust1stat + dust2stat;
if (dusttotstat == 0) {
kacaStat = "bersih";
flag = 0;
} else {
kacaStat = "kotor";
flag = 1;
}

/*if (dustTotal >= 100.0) {
//Serial.println("flag = 0");
kacaStat = "bersih";
flag = 0;
} else {
//Serial.println("flag = 1");
kacaStat = "kotor";
flag = 1;
}
delay(10);*/

```

© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

// Create the JSON document
JsonObject doc;
doc["kacaVal"] = dustDensity;
doc["kacaVal"] = dustTotal;//value dust total
doc["kacaStat"] = kacaStat;//value kacastat

// Send the JSON document over the "link" serial port
serializeJson(doc, Serial1);
serializeJson(doc, Serial);
Serial.println();
delay(10);

//delay(500);
}
}

if (flag == 1) {
// Create the JSON document
JsonObject doc;
doc["kacaVal"] = dustDensity;
doc["kacaVal"] = dustTotal;
doc["kacaStat"] = kacaStat;

// Send the JSON document over the "link" serial port
serializeJson(doc, Serial1);
serializeJson(doc, Serial);
Serial.println();
delay(100);

Serial.println("KACA AKAN DIBERSIHKAN");

```

```

Serial.println("1. POMPA MENYALA 2 DETIK");
digitalWrite(relayPompa, LOW);
delay(1000);
Serial.println("2. WIPER MENYALA 7 DETIK CLOCKWISE");
digitalWrite(relayPompa, HIGH);
analogWrite(enA, 90);
digitalWrite(in1, HIGH);
digitalWrite(in2, LOW);
delay(5000);
Serial.println("3. WIPER MATI 2 DETIK");
analogWrite(enA, 0);
digitalWrite(in1, LOW);
digitalWrite(in2, LOW);
delay(3000);
Serial.println("4. WIPER MENYALA 7 DETIK ANTI-CLOCKWISE");
analogWrite(enA, 90);
digitalWrite(in1, LOW);
digitalWrite(in2, HIGH);
delay(5000);
Serial.println("5. WIPER MATI");
analogWrite(enA, 0);
digitalWrite(in1, LOW);
digitalWrite(in2, LOW);
delay(3000);
Serial.println("6. WIPER MENYALA 7 DETIK CLOCKWISE");
analogWrite(enA, 90);
digitalWrite(in1, HIGH);
digitalWrite(in2, LOW);
delay(5000);
Serial.println("7. WIPER MATI");
analogWrite(enA, 0);

```

© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta


```

digitalWrite(in1, LOW);
digitalWrite(in2, LOW);
delay(3000);
Serial.println("8. WIPER MENYALA 7 DETIK ANTI-CLOCKWISE");
analogWrite(enA, 90);
digitalWrite(in1, LOW);
digitalWrite(in2, HIGH);
delay(5000);
Serial.println("9. WIPER MATI");
Serial.println("KACA SELESAI DIBERSIHKAN");
Serial.println();
delay(10);
flag = 0;
delay(10);
}

if (flag == 2) {
    // Create the JSON document
    StaticJsonDocument<200> doc;
    //doc["kacaVal"] = dustDensity;
    doc["kacaVal"] = dustTotal;
    doc["kacaStat"] = kacaStat;

    // Send the JSON document over the "link" serial port
    serializeJson(doc, Serial1);
    serializeJson(doc, Serial);
    Serial.println();
    delay(100);

    Serial.println("1. POMPA MENYALA 2 DETIK");
    digitalWrite(relayPompa, LOW);

```

© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

delay(1000);
Serial.println("2. POMPA MATI");
Serial.println();
flag = 0;
delay(10);
}

if (flag == 3) {
    // Create the JSON document
    StaticJsonDocument<200> doc;
    doc["kacaVal"] = dustDensity;
    //doc["kacaVal"] = dustTotal;
    doc["kacaStat"] = kacaStat;

    // Send the JSON document over the "link" serial port
    serializeJson(doc, Serial1);
    serializeJson(doc, Serial);
    Serial.println();
    delay(100);

    Serial.println("KACA AKAN DIBERSIHKAN");
    Serial.println("1. POMPA TIDAK MENYALA 2 DETIK");
    digitalWrite(relayPompa, HIGH);
    delay(1000);
    Serial.println("2. WIPER MENYALA 7 DETIK CLOCKWISE");
    digitalWrite(relayPompa, HIGH);
    analogWrite(enA, 90);
    digitalWrite(in1, HIGH);
    digitalWrite(in2, LOW);
    delay(5000);
    Serial.println("3. WIPER MATI 2 DETIK");
}

```

© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



```

analogWrite(enA, 0);
digitalWrite(in1, LOW);
digitalWrite(in2, LOW);
delay(3000);
Serial.println("4. WIPER MENYALA 7 DETIK ANTI-CLOCKWISE");
analogWrite(enA, 90);
digitalWrite(in1, LOW);
digitalWrite(in2, HIGH);
delay(5000);
Serial.println("5. WIPER MATI");
analogWrite(enA, 0);
digitalWrite(in1, LOW);
digitalWrite(in2, LOW);
delay(3000);
Serial.println("6. WIPER MENYALA 7 DETIK CLOCKWISE");
analogWrite(enA, 90);
digitalWrite(in1, HIGH);
digitalWrite(in2, LOW);
delay(5000);
Serial.println("7. WIPER MATI");
analogWrite(enA, 0);
digitalWrite(in1, LOW);
digitalWrite(in2, LOW);
delay(3000);
Serial.println("8. WIPER MENYALA 7 DETIK ANTI-CLOCKWISE");
analogWrite(enA, 90);
digitalWrite(in1, LOW);
digitalWrite(in2, HIGH);
delay(5000);
Serial.println("9. WIPER MATI");
Serial.println("KACA SELESAI DIBERSIHKAN");

```

© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



```
Serial.println();  
delay(10);  
flag = 0;  
  
delay(10);  
}  
}
```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



```

#include <ArduinoJson.h>

#if defined(ESP32)

#include <WiFi.h>

#include <FirebaseESP32.h>

#elif defined(ESP8266)

#include <ESP8266WiFi.h>

#include <FirebaseESP8266.h>

#endif

//Provide the token generation process info.
#include "addons/TokenHelper.h"
//Provide the RTDB payload printing info and other helper functions.
#include "addons/RTDBHelper.h"

/* 1. Define the WiFi credentials */
#define WIFI_SSID "WIFI TETANGGA"
#define WIFI_PASSWORD "Tanyanenek"

/* 2. Define the API Key */
#define API_KEY "AIzaSyC8x3rGaQxpRCi5ybM1ONrhIM7WEhJV0gw"

/* 3. Define the RTDB URL */
#define DATABASE_URL "tasyifadian-default-rtdb.asia-southeast1.firebaseio.com" //<databaseName>.firebaseio.com or <databaseName>.<region>.firebasedatabase.app

/* 4. Define the user Email and password that already registered or added in your project */
#define USER_EMAIL "syifaekam@gmail.com"
#define USER_PASSWORD "SyifaDian?!"

//Define Firebase Data object

```

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

FirebaseData fbdo; //wakilin library firebase

FirebaseAuth auth;
FirebaseConfig config;

unsigned long sendDataPrevMillis = 0;

int count = 0;

String outStr1, outStr2;

void setup() {
  Serial.begin(115200);

  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("Connecting to Wi-Fi");
  while (WiFi.status() != WL_CONNECTED)
  {
    Serial.print(".");
    delay(300);
  }
  Serial.println();
  Serial.print("Connected with IP: ");
  Serial.println(WiFi.localIP());
  Serial.println();

  Serial.printf("Firebase Client v%s\n\n", FIREBASE_CLIENT_VERSION);

  /* Assign the api key (required) */
  config.api_key = API_KEY;

```

© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

/* Assign the user sign in credentials */
auth.user.email = USER_EMAIL;
auth.user.password = USER_PASSWORD;

/* Assign the RTDB URL (required) */
config.database_url = DATABASE_URL;

/* Assign the callback function for the long running token generation
task */
config.token_status_callback = tokenStatusCallback; //see
addons/TokenHelper.h

Firebase.begin(&config, &auth);

//Or use legacy authenticate method
//Firebase.begin(DATABASE_URL, "<database secret>");
}

void loop() {
  static uint8_t flag = 0, wiper, pompa;
  static uint32_t millisButton, timeButton = 250;

  if (millis() - millisButton > timeButton) {
    if (Firebase.ready()) {

      if (Firebase.getInt(fbdo, "/wiper")) {
        if (fbdo.dataType() == "int") {
          wiper = fbdo.intData();
          //Serial.print("wiper = ");
          //Serial.println(wiper);
          if (wiper == 1) {
            outStr1 = String("*w1"); kalo 1, dia outstr1 nyimpen

```

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

//Serial.print(outStr1);
} else if (wiper == 0) {
    outStr1 = String("*w0");
    //Serial.print(outStr1);
}
delay(10);
}
}

if (Firebase.getInt(fbdo, "/pompa")) {
    if (fbdo.dataType() == "int") {
        pompa = fbdo.intData();
        //Serial.print("pompa = ");
        //Serial.println(pompa);
        if (pompa == 1) {
            outStr2 = outStr1 + String("p1#") + String('\n');
            Serial.print(outStr2);
        } else if (pompa == 0) {
            outStr2 = outStr1 + String("p0#") + String('\n');
            Serial.print(outStr2);
        }
        delay(10);
    }
}

else {
    Serial.println(fbdo.errorReason());
}
}

millisButton = millis();
}

```



```

while (Serial.available()) {
    StaticJsonDocument<200> doc;
    DeserializationError err = deserializeJson(doc, Serial);
    if (err == DeserializationError::Ok)
    {
        float kacaVal = doc["kacaVal"];
        const char* kacaStat = doc["kacaStat"]; //const char sebenarnya
string, json gabisa baca string
        if (Firebase.ready())
        {
            Serial.printf("Set kacaVal... %s\n", Firebase.setFloat(fbdo,
"/kacaVal", kacaVal) ? "ok" : fbdo.errorReason().c_str());//difolder
kacaval
            Serial.printf("Set kacaStat... %s\n", Firebase.setString(fbdo,
"/kacaStat", kacaStat) ? "ok" : fbdo.errorReason().c_str());
        }
    }
else
{
    Serial.print("deserializeJson() returned ");
    Serial.println(err.c_str());
}
while (Serial.available() > 0)
    Serial.read(); //baca ulang lagi
}
}
}

```

© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



```

const char* ssid = "WIFI TETANGGA";
const char* password = "Tanyanenek";

//https://console.firebase.google.com/project/xxxxxxxxx/settings/serviceaccounts/databasesecrets

String FIREBASE_HOST = "tasyifadian-default-rtdb.asia-southeast1.firebaseio.com";
String FIREBASE_AUTH = "UBE4bFBMW8D2Vo7Po0iUe3xkQcnk1fBzewPOoN37";

#include "FirebaseESP32.h"
FirebaseData firebaseData;

#include <WiFi.h>
#include "soc/soc.h"
#include "soc/rtc_cntl_reg.h"
#include "Base64.h"
#include "esp_camera.h"

#include "FS.h"
#include "SD_MMC.h"
#include <Preferences.h>
Preferences preferences;

// WARNING!!! Make sure that you have either selected ESP32 Wrover
Module,

// or another board which has PSRAM enabled

//CAMERA_MODEL_AI_THINKER

#define PWDN_GPIO_NUM 32
#define RESET_GPIO_NUM -1
#define XCLK_GPIO_NUM 0

```

© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

#define SIOD_GPIO_NUM 26
#define SIOC_GPIO_NUM 27
#define Y9_GPIO_NUM 35
#define Y8_GPIO_NUM 34
#define Y7_GPIO_NUM 39
#define Y6_GPIO_NUM 36
#define Y5_GPIO_NUM 21
#define Y4_GPIO_NUM 19
#define Y3_GPIO_NUM 18
#define Y2_GPIO_NUM 5
#define VSYNC_GPIO_NUM 25
#define HREF_GPIO_NUM 23
#define PCLK_GPIO_NUM 22

void setup() {
WRITE_PERI_REG(RTC_CNTL_BROWN_OUT_REG, 0);
Serial.begin(115200);
Serial.setDebugOutput(true);
Serial.println();
Serial.println("ssid: " + (String)ssid);
Serial.println("password: " + (String)password);
WiFi.begin(ssid, password);
long int StartTime=millis();
while (WiFi.status() != WL_CONNECTED) {
delay(500);

if ((StartTime+10000) < millis()) break;
}

```

```

if (WiFi.status() == WL_CONNECTED) {
  char* apssid = "WIFI TETANGGA";
  char* appassword = "Tanyanenek"; //AP password require at least 8
  characters.

  Serial.println("");
  Serial.print("Camera Ready! Use 'http://");
  Serial.print(WiFi.localIP());
  Serial.println("' to connect");
  WiFi.softAP((WiFi.localIP().toString()+"_"+(String)apssid).c_str(),
  appassword);
} else {
  Serial.println("Connection failed");
  return;
}

camera_config_t config;

config.ledc_channel = LEDC_CHANNEL_0;
config.ledc_timer = LEDC_TIMER_0;
config.pin_d0 = Y2_GPIO_NUM;
config.pin_d1 = Y3_GPIO_NUM;
config.pin_d2 = Y4_GPIO_NUM;
config.pin_d3 = Y5_GPIO_NUM;
config.pin_d4 = Y6_GPIO_NUM;
config.pin_d5 = Y7_GPIO_NUM;
config.pin_d6 = Y8_GPIO_NUM;
config.pin_d7 = Y9_GPIO_NUM;
config.pin_xclk = XCLK_GPIO_NUM;
config.pin_pclk = PCLK_GPIO_NUM;
config.pin_vsync = VSYNC_GPIO_NUM;

```

© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

config.pin_href = HREF_GPIO_NUM;
config.pin_sscb_sda = SIOD_GPIO_NUM;
config.pin_sscb_scl = SIOC_GPIO_NUM;
config.pin_pwdn = PWDN_GPIO_NUM;
config.pin_reset = RESET_GPIO_NUM;
config.xclk_freq_hz = 20000000;
config.pixel_format = PIXFORMAT_JPEG;
//init with high specs to pre-allocate larger buffers
if(psramFound()){
config.frame_size = FRAMESIZE_UXGA;
config.jpeg_quality = 10; //0-63 lower number means higher quality
config.fb_count = 2;
} else {
config.frame_size = FRAMESIZE_SVGA;
config.jpeg_quality = 12; //0-63 lower number means higher quality
config.fb_count = 1;
}

// camera init
esp_err_t err = esp_camera_init(&config);
if (err != ESP_OK) {
Serial.printf("Camera init failed with error 0x%x", err);
delay(1000);
ESP.restart();
}

//drop down frame size for higher initial frame rate
sensor_t * s = esp_camera_sensor_get();
s->set_framesize(s, FRAMESIZE_CIF); // VGA|CIF|QVGA|HQVGA|QQVGA ( UXGA?
SXGA? XGA? SVGA? )

```

© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);

Firebase.reconnectWiFi(true);

Firebase.setMaxRetry(firebaseData, 3);

Firebase.setMaxErrorQueue(firebaseData, 30);

Firebase.enableClassicRequest(firebaseData, true);
}

void loop() {
    String jsonData = "{\"photo\":\\"" + Photo2Base64() + "\"}";
    String photoPath = "/esp32-cam";
    Serial.println(Photo2Base64());
    if (Firebase.pushJSON(firebaseData, photoPath, jsonData)) {
        Serial.println(firebaseData.dataPath());
        Serial.println(firebaseData.pushName());
        Serial.println(firebaseData.dataPath() + "/" + firebaseData.pushName());
    } else {
        Serial.println(firebaseData.errorReason());
    }
    Firebase.setString(firebaseData, "/photo", Photo2Base64());
    delay(300000);
}

String Photo2Base64() {
    camera_fb_t * fb = NULL;

    fb = esp_camera_fb_get();

    if(!fb) {
        Serial.println("Camera capture failed");
        return "";
    }
}

```

```

}

String imageFile = "data:image/jpeg;base64,";

char *input = (char *)fb->buf;

char output[base64_enc_len(3)];

for (int i=0;i<fb->len;i++) {

base64_encode(output, (input++), 3);

if (i%3==0) imageFile += urlencode(String(output));

}

esp_camera_fb_return(fb);

return imageFile;

}

//https://github.com/zenmanenergy/ESP8266-Arduino-Examples/
String urlencode(String str)
{
String encodedString="";
char c;
char code0;
char code1;
char code2;
for (int i =0; i < str.length(); i++){
c=str.charAt(i);
if (c == ' '){
encodedString+= '+';

} else if (isalnum(c)){
encodedString+=c;

} else{
code1=(c & 0xf)+'0';
if ((c & 0xf) >9){

```

© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



```

code1=(c & 0xf) - 10 + 'A';
}
c=(c>>4)&0xf;
code0=c+'0';
if (c > 9){
code0=c - 10 + 'A';
}
code2='\0';
encodedString+='%';
encodedString+=code0;
encodedString+=code1;
//encodedString+=code2;
}
yield();
}
return encodedString;
}

```

© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



POLITEKNIK
NEGERI
JAKARTA



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Schematic: [arduino-mega2560-schematic.pdf](#)

Summary

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 14 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

Power

The Arduino Mega can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The Mega2560 differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.



- **I²C: 20 (SDA) and 21 (SCL).** Support I²C (TWI) communication using the [Wire library](#) (documentation on the Wiring website). Note that these pins are not in the same location as the I²C pins on the Duemilanove or Diecimila.

The Mega2560 has 16 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though it is possible to change the upper end of their range using the `AREF` pin and `analogReference()` function.

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with `analogReference()`.
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

Communication

The Arduino Mega2560 has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega2560 provides four hardware UARTs for TTL (5V) serial communication. An ATmega8U2 on the board channels one of these over USB and provides a virtual com port to software on the computer (Windows machines will need a .inf file, but OSX and Linux machines will recognize the board as a COM port automatically). The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the ATmega8U2 chip and USB connection to the computer (but not for serial communication on pins 0 and 1). A [SoftwareSerial library](#) allows for serial communication on any of the Mega2560's digital pins.

The ATmega2560 also supports I²C (TWI) and SPI communication. The Arduino software includes a [Wire library](#) to simplify use of the I²C bus; see the [documentation on the Wiring website](#) for details. For SPI communication, use the [SPI library](#).

Programming

The Arduino Mega can be programmed with the Arduino software ([download](#)). For details, see the [reference](#) and [tutorials](#). The ATmega2560 on the Arduino Mega comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It



The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

Memory

The ATmega2560 has 256 KB of flash memory for storing code (of which 8 KB is used for the bootloader), 8 KB of SRAM and 4 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

Input and Output

Each of the 54 digital pins on the Mega can be used as an input or output, using `pinMode()`, `digitalWrite()`, and `digitalRead()` functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX); Serial 1: 19 (RX) and 18 (TX); Serial 2: 17 (RX) and 16 (TX); Serial 3: 15 (RX) and 14 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. Pins 0 and 1 are also connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts: 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), and 21 (interrupt 2).** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- **PWM: 0 to 13.** Provide 8-bit PWM output with the [analogWrite\(\)](#) function.
- **SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS).** These pins support SPI communication using the [SPI library](#). The SPI pins are also broken out on the ICSP header, which is physically compatible with the Uno, Duemilanove and Diecimila.
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH

communicates using the original STK500 protocol ([reference](#), [C header files](#)). You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.

Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Mega2560 is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2 is connected to the reset line of the ATmega2560 via a 100 nanofarad capacitor. When this line is asserted (set low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload. This setup has other implications. When the Mega2560 is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Mega2560. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

The Mega2560 contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see [this forum thread](#) for details.

USB Overcurrent Protection

The Arduino Mega2560 has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

Physical Characteristics and Shield Compatibility

The maximum length and width of the Mega2560 PCB are 4 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Three screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.

The Mega2560 is designed to be compatible with most shields designed for the Uno, Diecimila or Duemilanove. Digital pins 0 to 13 (and the adjacent AREF and GND pins), analog inputs 0 to 5, the power header, and ICSP header are all in equivalent locations. Further the main UART (serial port) is located on the same pins (0 and 1), as are external interrupts 0 and 1 (pins 2 and 3 respectively). SPI is available through the ICSP header on both the Mega2560 and Duemilanove / Diecimila. Please note that I²C is not located on the same pins on the Mega (20 and 21) as the Duemilanove / Diecimila (analog inputs 4 and 5).



Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengummikan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

1 Overview

ESP32-WROOM-32 is a powerful, generic Wi-Fi+BT+BLE MCU module that targets a wide variety of applications, ranging from low-power sensor networks to the most demanding tasks, such as voice encoding, music streaming and MP3 decoding.

At the core of this module is the ESP32-D0WD06 chip*. The chip embedded is designed to be scalable and adaptive. There are two CPU cores that can be individually controlled, and the CPU clock frequency is adjustable from 80 MHz to 240 MHz. The chip also has a low-power co-processor that can be used instead of the CPU to save power while performing tasks that do not require much computing power, such as monitoring of peripherals. ESP32 integrates a rich set of peripherals, ranging from capacitive touch sensors, Hall sensors, SD card interface, Ethernet, high-speed SPI, UART, PS and PC.

Note:
* For details on the part numbers of the ESP32 family of chips, please refer to the document [ESP32 Pinouts](#).

The integration of Bluetooth®, Bluetooth LE and Wi-Fi ensures that a wide range of applications can be targeted, and that the module is all-around: using Wi-Fi allows a large physical range and direct connection to the Internet through a Wi-Fi router, while using Bluetooth allows the user to conveniently connect to the phone or broadcast low energy beacons for its detection. The sleep current of the ESP32 chip is less than 5 µA, making it suitable for battery powered and wearable electronics applications. The module supports a data rate of up to 150 Mbps, and 20 dBm output power at the antenna to ensure the widest physical range. As such the module does offer industry-leading specifications and the best performance for electronic integration, range, power consumption, and connectivity.

The operating system chosen for ESP32 is freeRTOS with LwIP, TLS 1.2 with hardware acceleration is built in as well. Secure (encrypted) over the air (OTA) upgrade is also supported, so that users can upgrade their products even after their release, at minimum cost and effort.

Table 1 provides the specifications of ESP32-WROOM-32.

Table 1: ESP32-WROOM-32 Specifications

Categories	Items	Specifications
Certification	RF certification	FCC/CE-RED/IC/TELEC/KCC/SRRC/NCC
	Wi-Fi certification	Wi-Fi Alliance
	Bluetooth certification	BBB
Test	Green certification	RoHS/REACH
	Reliability	HTCL/HTSL/HAHAST/TCT/ESD
Wi-Fi	Protocols	802.11 b/g/n (802.11n up to 150 Mbps) A-MPDU and A-MSDU aggregation and 0.4 µs guard interval support
	Frequency range	2.4 GHz ~ 2.5 GHz
	Protocols	Bluetooth v4.2 BR/EDR and BLE specification NZIF receiver with -97 dBm sensitivity Class-1, class-2 and class-3 transmitter AFH
Bluetooth	Radio	Class-1, class-2 and class-3 transmitter AFH
	Audio	CVSD and SBC

Not Recommended For New Designs (NRND)
ESP32-WROOM-32 Datasheet v3.2

2 Pin Definitions

2.1 Pin Layout

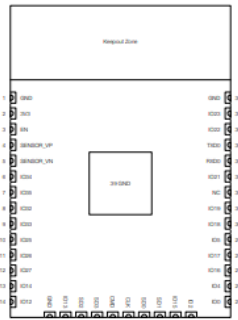


Figure 1: ESP32-WROOM-32 Pin Layout (Top View)

2.2 Pin Description

ESP32-WROOM-32 has 38 pins. See pin definitions in Table 2.

Table 2: Pin Definitions

Name	No.	Type	Function
GND	1	P	Ground
V33	2	P	Power supply
EN	3	I	Module-enable signal. Active high.
SENSOR_VP	4	I	GPIO36, ADC1_CH0, RTC_GPIO0
SENSOR_VN	5	I	GPIO39, ADC1_CH3, RTC_GPIO3
IO34	6	I	GPIO34, ADC1_CH6, RTC_GPIO4
IO35	7	I	GPIO35, ADC1_CH7, RTC_GPIO5
IO32	8	I/O	GPIO32, XTAL_32K_P (32.768 kHz crystal oscillator input), ADC1_CH4, TOUCH8, RTC_GPIO8
IO33	9	I/O	GPIO33, XTAL_32K_N (32.768 kHz crystal oscillator output), ADC1_CH5, TOUCH8, RTC_GPIO8

Not Recommended For New Designs (NRND)
ESP32-WROOM-32 Datasheet v3.2

Categories	Items	Specifications
Hardware	Module interfaces	SD card, UART, SPI, SDIO, PC, LED PWM, Motor PWM, I ² S, IR, pulse counter, GPIO, capacitive touch sensor, ADC, DAC, Two-Wire Automotive Interface (TWAI®), compatible with ISO11898-1 (CAN Specification 2.0)
	On-chip sensor	Hall sensor
	Integrated crystal	40 MHz crystal
	Integrated SPI flash	4 MB
	Operating voltage/Power supply	3.0 V ~ 3.6 V
	Operating current	Average: 80 mA
	Minimum current delivered by power supply	500 mA
	Recommended operating temperature range	-40 °C ~ +85 °C
	Package size	(18.00±0.10) mm × (25.50±0.10) mm × (3.10±0.10) mm
	Moisture sensitivity level (MSL)	Level 3



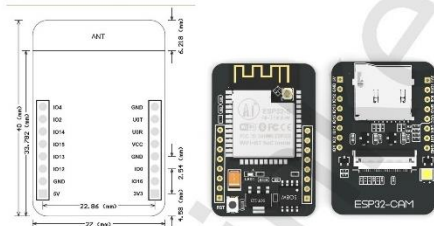
Name	No.	Type	Function
IO25	10	I/O	GPIO25, DAC_1, ADC2_CH6, RTC_GPIO6, EMAC_RXD0
IO26	11	I/O	GPIO26, DAC_2, ADC2_CH9, RTC_GPIO7, EMAC_RXD1
IO27	12	I/O	GPIO27, ADC2_CH7, TOUCH7, RTC_GPIO17, EMAC_RX_DV
IO14	13	I/O	GPIO14, ADC2_CH6, TOUCH6, RTC_GPIO16, MTMS, HSPICLK, HS2_CLK, SD_CLK, EMAC_TXD2
IO12	14	I/O	GPIO12, ADC2_CH6, TOUCH5, RTC_GPIO15, MTDI, HSPICLK, HS2_DATA2, SD_DATA2, EMAC_TXD3
GND	15	P	Ground
IO13	16	I/O	GPIO13, ADC2_CH4, TOUCH4, RTC_GPIO14, MTCK, HSPID, HS2_DATA3, SD_DATA3, EMAC_RX_ER
SHD/SD2*	17	I/O	GPIO9, SD_DATA2, SPIHD, HS1_DATA2, U1RXD
SWP/SD3*	18	I/O	GPIO10, SD_DATA3, SPIWP, HS1_DATA3, U1TXD
SCS/CMD*	19	I/O	GPIO11, SD_CMD, SPICSD, HS1_CMD, U1RTS
SCW/CLK*	20	I/O	GPIO6, SD_CLK, SPICLK, HS1_CLK, U1CTS
SD0/SD0*	21	I/O	GPIO7, SD_DATA0, SPIQ, HS1_DATA0, U2RTS
SD1/SD1*	22	I/O	GPIO8, SD_DATA1, SPD, HS1_DATA1, U2CTS
IO15	23	I/O	GPIO15, ADC2_CH3, TOUCH3, MTD0, HSPICSD0, RTC_GPIO13, HS2_CMD, SD_CMD, EMAC_RXD3
IO2	24	I/O	GPIO2, ADC2_CH2, TOUCH2, RTC_GPIO12, HSPWP, HS2_DATA0, SD_DATA0
IO0	25	I/O	GPIO0, ADC2_CH1, TOUCH1, RTC_GPIO11, CLK_OUT1, EMAC_TX_CLK
IO4	26	I/O	GPIO4, ADC2_CH0, TOUCH0, RTC_GPIO10, HSPHD, HS2_DATA1, SD_DATA1, EMAC_TX_ER
IO16	27	I/O	GPIO16, HS1_DATA4, U2RXD, EMAC_CLK_OUT
IO17	28	I/O	GPIO17, HS1_DATA5, U2TXD, EMAC_CLK_OUT_180
IO5	29	I/O	GPIO5, VSPICSD0, HS1_DATA5, EMAC_RX_CLK
IO18	30	I/O	GPIO18, VSPICLK, HS1_DATA7
IO19	31	I/O	GPIO19, VSPICQ, U0CTS, EMAC_TXD0
NC	32	-	-
IO21	33	I/O	GPIO21, VSPHD, EMAC_TX_EN
IO20	34	I/O	GPIO3, U0RXD, CLK_OUT2
TXD0	35	I/O	GPIO1, U0TXD, CLK_OUT3, EMAC_RXD2
IO22	36	I/O	GPIO22, VSPWP, U0RTS, EMAC_TXD1
IO23	37	I/O	GPIO23, VSPID, HS1_STROBE
GND	38	P	Ground

Notice:
* Pins SD0/CLK, SD0/SD1, SD1/SD1, SHD/SD2, SWP/SD3 and SCS/CMD, namely, GPIO5 to GPIO11 are connected to the integrated SPI flash integrated on the module and are not recommended for other uses.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummikan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

ESP32-CAM Module



Features

- The smallest 802.11b/g/n Wi-Fi SoC Module
- Low power 32-bit CPU, can also serve the application processor
- Up to 160MHz clock speed, Summary computing power up to 600 DMIPS
- Built-in 520 KB SRAM, external 4MPSRAM
- Supports UART/SPI/I2C/PWM/ADC/DAC
- Support OV2640 and OV7670 cameras, Built-in Flash lamp.
- Support image WiFi upload
- Support TF card
- Supports multiple sleep modes.
- Embedded Lwip and FreeRTOS
- Supports STA/AP/STA+AP operation mode
- Support Smart Config/AirKiss technology
- Support for serial port local and remote firmwares (OTA)

Overview

The ESP32-CAM has a very competitive small-size camera module that can operate independently as a minimum system with a footprint of only 27*40.5*4.5mm and a deep sleep current of up to 6mA.

ESP-32CAM can be widely used in various IoT applications. It is suitable for home smart devices, industrial wireless control, wireless monitoring, QR wireless identification, wireless positioning system signals and other IoT applications. It is an ideal solution for IoT applications.

ESP-32CAM adopts DIP package and can be directly inserted into the backplane to realize rapid production of products, providing customers with high-reliability connection mode, which is convenient for application in various IoT hardware terminals.

Product Specifications

Module Model	ESP32-CAM
Package	DIP-16
Size	27*40.5*4.5 (L0.2) mm
SPI Flash	Default 32Mbit
RAM	520KB SRAM +4M PSRAM
Bluetooth	Bluetooth 4.2 BR/EDR and BLE standards
Wi-Fi	802.11 b/g/n/
Support interface	UART, SPI, I2C, PWM
Support TF card	Maximum support 4G
IO port	9
UART Baudrate	Default 115200 bps
Image Output Format	JPEG(OV2640 support only), BMP, GRAYSCALE
Spectrum Range	2412 ~2484MHz
Antenna	Onboard PCB antenna, gain 2dBi
Transmit Power	802.11b: 17.42 dBm (@11Mbps) 802.11g: 14.42 dBm (@54Mbps) 802.11n: 13+2 dBm (@MCS7)
Receiving Sensitivity	CCK, 1 Mbps: -90dBm CCK, 11 Mbps: -85dBm 6 Mbps (1/2 BPSK): -88dBm 54 Mbps (3/4 64-QAM): -70dBm MCS7 (6.5 Mbps, 7.2 Mbps): -67dBm
Power Dissipation	Turn off the flash lamp: 180mA@5V Turn on the flash lamp and turn on the brightness to the maximum: 310mA@5V Deep-sleep: Minimum power consumption can be achieved 6mA@5V Modern-sleep: Minimum up to 20mA@5V Light-sleep: Minimum up to 6.7mA@5V
Security	WPA/WPA2/WPA3-Enterprise/WPS
Power Supply Range	5V
Operating Temperature	-20 °C ~ 85 °C
Storage Environment	-40 °C ~ 90 °C, <90%RH

Weight	10g
--------	-----

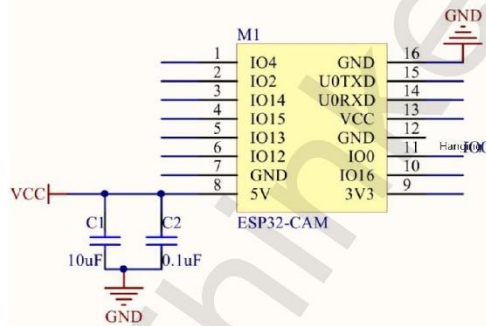
ESP32-CAM module picture output format rate

Format Size	QQVGA	QVGA	VGA	SVGA
JPEG	6	7	7	8
BMP	9	9	-	-
GRAYSCALE	9	8	-	-

Internal Pin Connect

CAM	ESP32	SD	ESP32
D0	PIN5	CLK	PIN14
D1	PIN18	CMD	PIN15
D2	PIN19	DATA0	PIN2
D3	PIN21	DATA1/Flash lamp	PIN4
D4	PIN36	DATA2	PIN12
D5	PIN39	DATA3	PIN13
D6	PIN34		
D7	PIN35		
XCLK	PIN0		
PCLK	PIN22		
VSYNC	PIN25		
HREF	PIN23		
SDA	PIN26		
SCL	PIN27		
POWER PIN	PIN32		

Minimum system diagram



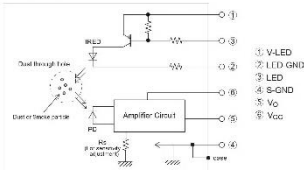
Contact US

Shenzhen Ai-Thinker Technology Co., Ltd
 Address: 7/F, Fengze Building B, Huaifeng Industrial Park 2th, Huangkoung street, Xixiang Road, Baoan, Shenzhen
 China
 Website: www.ai-thinker.com Tel: 0755-29162996 E-mail: support@aitinker.com

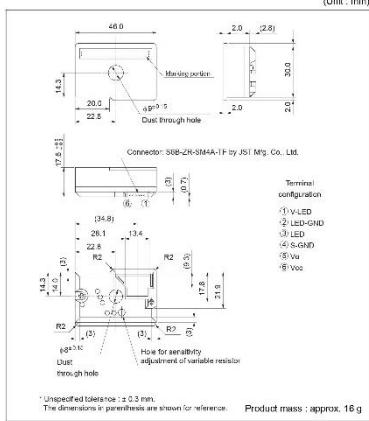
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummikan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Internal schematic



Outline Dimensions



Marking information



Date code (2 digit)

1st digit		2nd digit	
Year of production	Month of production	Year of production	Month of production
A.D.	Mark	Year	Mark
2000	0	1	1
2001	1	2	2
2002	2	3	3
2003	3	4	4
2004	4	5	5
2005	5	6	6
2006	6	7	7
2007	7	8	8
2008	8	9	9
2009	9	10	X
2010	0	11	Y
:	:	12	Z

repeats in a 10 year cycle

Absolute Maximum Ratings

(T _a =25°C)			
Parameter	Symbol	Rating	Unit
Supply voltage	V _{CC}	-0.3 to +7	V
Input terminal voltage	V _{I,LED}	-0.3 to V _{CC}	V
Operating temperature	T _{OP}	-10 to +65	°C
Soldering temperature	T _{sold}	-20 to +80	°C

Electro-optical Characteristics

(T _a =25°C, V _{CC} =5V)						
Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Sensitivity	K	*1)*2)*3)	0.35	0.5	0.65	V/(0.1mg/m ³)
Output voltage at no dust	V _{OC}	*2)*3)	0	0.9	1.5	V
Output voltage range	V _{OH}	*1)*3) R _L =4.7kΩ	3.4	—	—	V
LED terminal current	I _{I,LED}	*2)LED terminal voltage=0	—	10	20	mA
Consumption current	I _{CC}	*2) R _L =∞	—	11	20	mA

*1) Sensitivity is specified by the amount of output voltage change when dust density changes by 0.1 mg/m³.
 And the dust density (in detection) is a value of the density of signpost (MIL-D-8789B) made measured by the digital dust monitor (P-3L2, manufactured by SEIRATA SCIENTIFIC TECHNOLOGY LTD.).
 *2) Test condition is shown in Fig. 1.
 *3) Output sampling timing is shown in Fig. 2.

Recommended input condition for LED input terminal

Parameter	Symbol	Value	Unit
Pulse Cycle	T	10 ± 1	ms
Pulse Width	PW	0.32 ± 0.02	ms
Operating Supply voltage	V _{CC}	5 ± 0.5	V

Fig. 1 Input Condition for LED Input Terminal

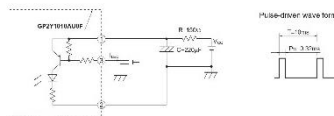


Fig. 2 Sampling Timing of Output Pulse

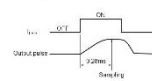
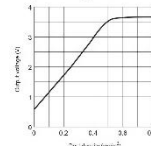


Fig. 3 Output Voltage vs. Dust Density



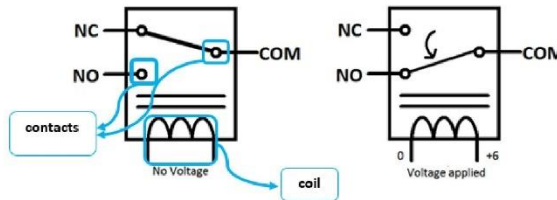
Remarks : Please be aware that all data in the graph are just for reference and are not for guarantee.



RELAY MODULES

RELAY WORKING IDEA

Relays consist of three pins normally open pin , normally closed pin, common pin and coil. When coil powered on magnetic field is generated the contacts connected to each other.

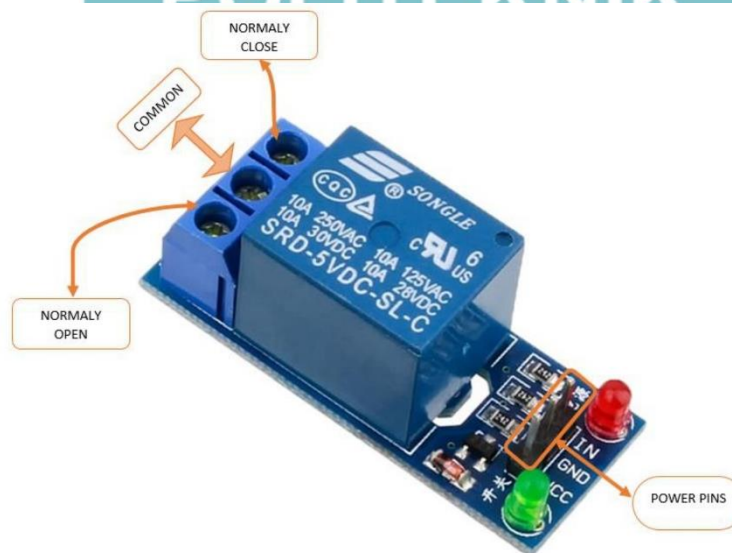


Relay modules 1-channel features

- Contact current 10A and 250V AC or 30V DC.
- Each channel has indication LED.
- Coil voltage 12V per channel.
- Kit operating voltage 5-12 V
- Input signal 3-5 V for each channel.
- Three pins for normally open and closed for each channel.

How to connect relay module with Arduino

As shown in relay working idea it depends on magnetic field generated from the coil so there is power isolation between the coil and the switching pins so coils can be easily powered from Arduino by connecting VCC and GND pins from Arduino kit to the relay module kit after that we choose Arduino output pins depending on the number of relays needed in project designed and set these pins to output and make it out high (5 V) to control the coil that allow controlling of switching process.



NOTE : whatever was the relay channels number the pinconfiguration is the same for every channel except the power pins (VCC and GND) are for the board itself. The input signal (IN) pin for every relay.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan Laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta





Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta


2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

HT Handson Technology

L298N Dual H-Bridge Motor Driver

User Guide

This dual bidirectional motor driver, is based on the very popular L298 Dual H-Bridge Motor Driver Integrated Circuit. The circuit will allow you to easily and independently control two motors of up to 2A each in both directions. It is ideal for robotic applications and well suited for connection to a microcontroller requiring just a couple of control lines per motor. It can also be interfaced with simple manual switches, TTL logic gates, relays, etc. This board equipped with power LED indicators, on-board +5V regulator and protection diodes.

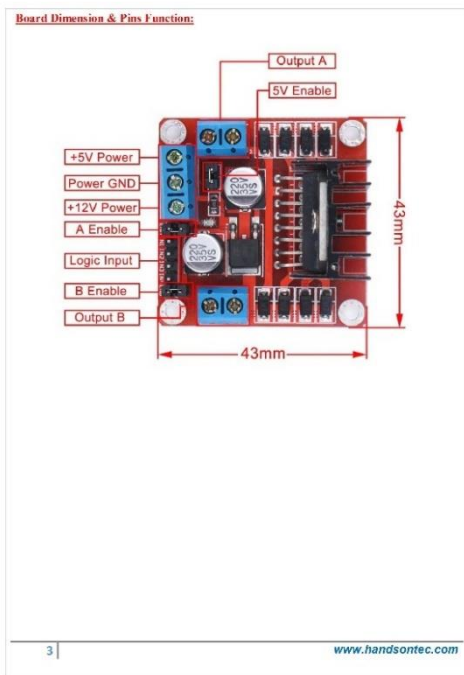
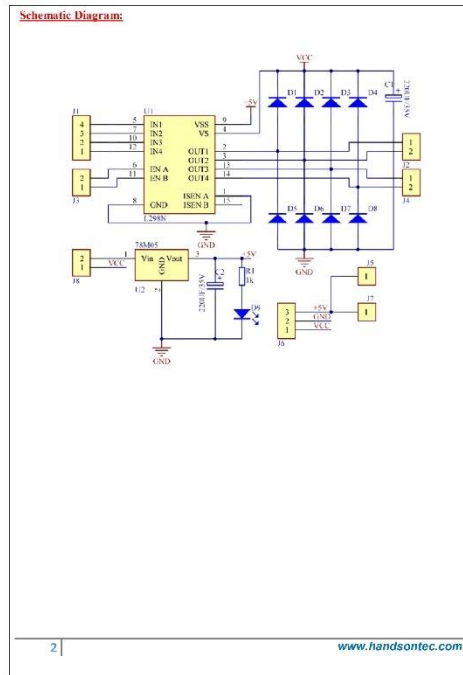


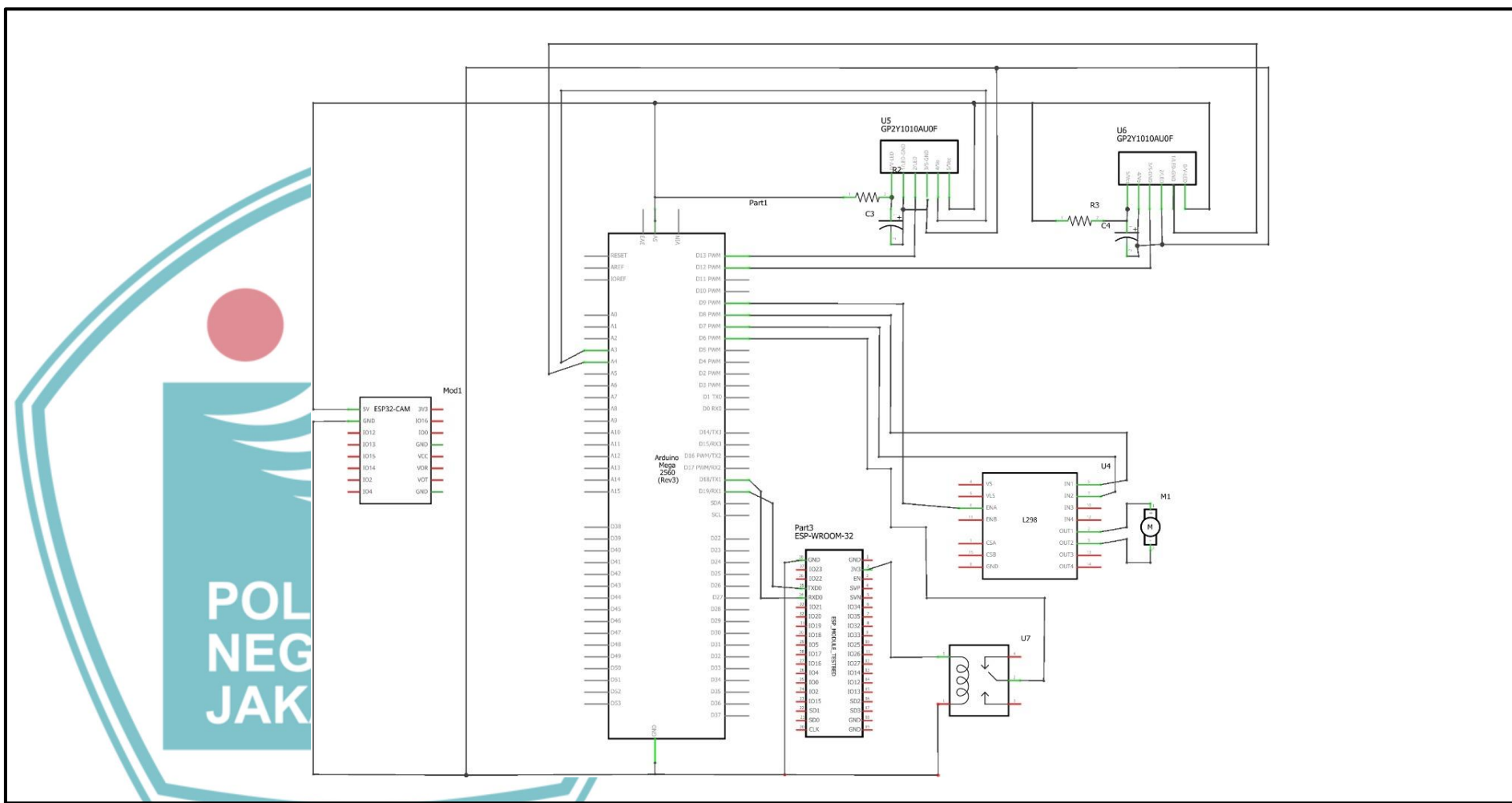
SKU: MDU-1049

Brief Data:

- Input Voltage: 3.3V-10Vdc.
- Driver: L298N Dual H Bridge DC Motor Driver
- Power Supply: DC 5V - 35V
- Peak current: 2 Amp
- Operating current range: 0 - 36mA
- Control signal input voltage range :
- Low: $-0.3V \leq V_{in} \leq 1.5V$.
- High: $2.3V \leq V_{in} \leq V_{cc}$.
- Enable signal input voltage range :
 - o Low: $-0.3 \leq V_{in} \leq 1.5V$ (control signal is invalid).
 - o High: $2.3V \leq V_{in} \leq V_{cc}$ (control signal active).
- Maximum power consumption: 20W (when the temperature $T = 75 \text{ }^\circ\text{C}$).
- Storage temperature: $-25 \text{ }^\circ\text{C} - 110 \text{ }^\circ\text{C}$.
- On-board +5V regulated Output supply (supply to controller board i.e. Arduino).
- Size: 3.4cm x 4.3cm x 2.7cm.

1 | www.handsontec.com



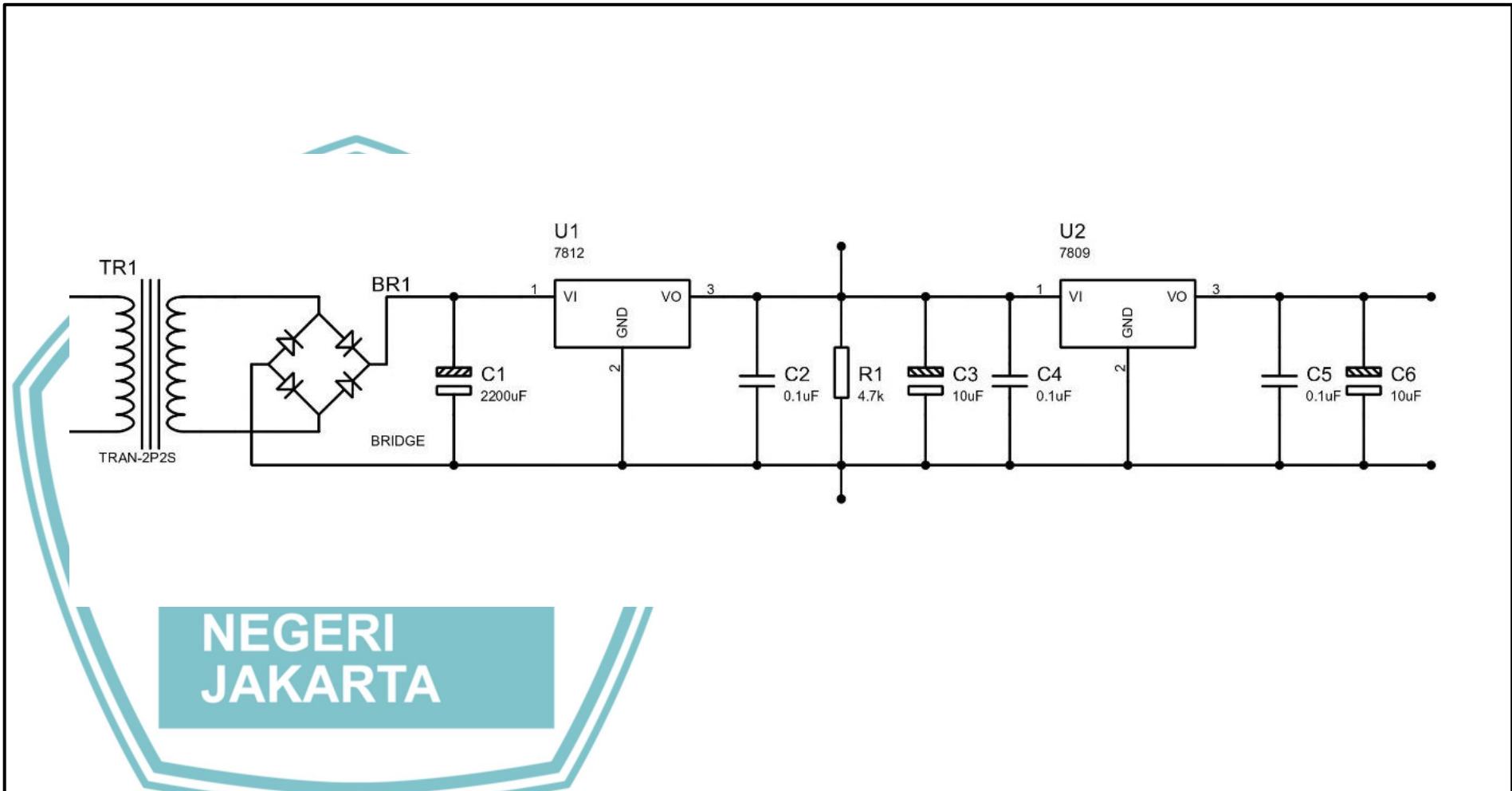


01	SKEMATIK RANGKAIAN SISTEM
PROGRAM STUDI TELEKOMUNIKASI	
JURUSAN TEKNIK ELEKTRO – POLITEKNIK NEGERI JAKARTA	
<i>Digambar</i>	: Syifa Eka Muliandari
<i>Diperiksa</i>	: Yenniwarti Rafsyam, SST., M.T.
<i>Tanggal</i>	: Juli 2021



Hak Cipta :

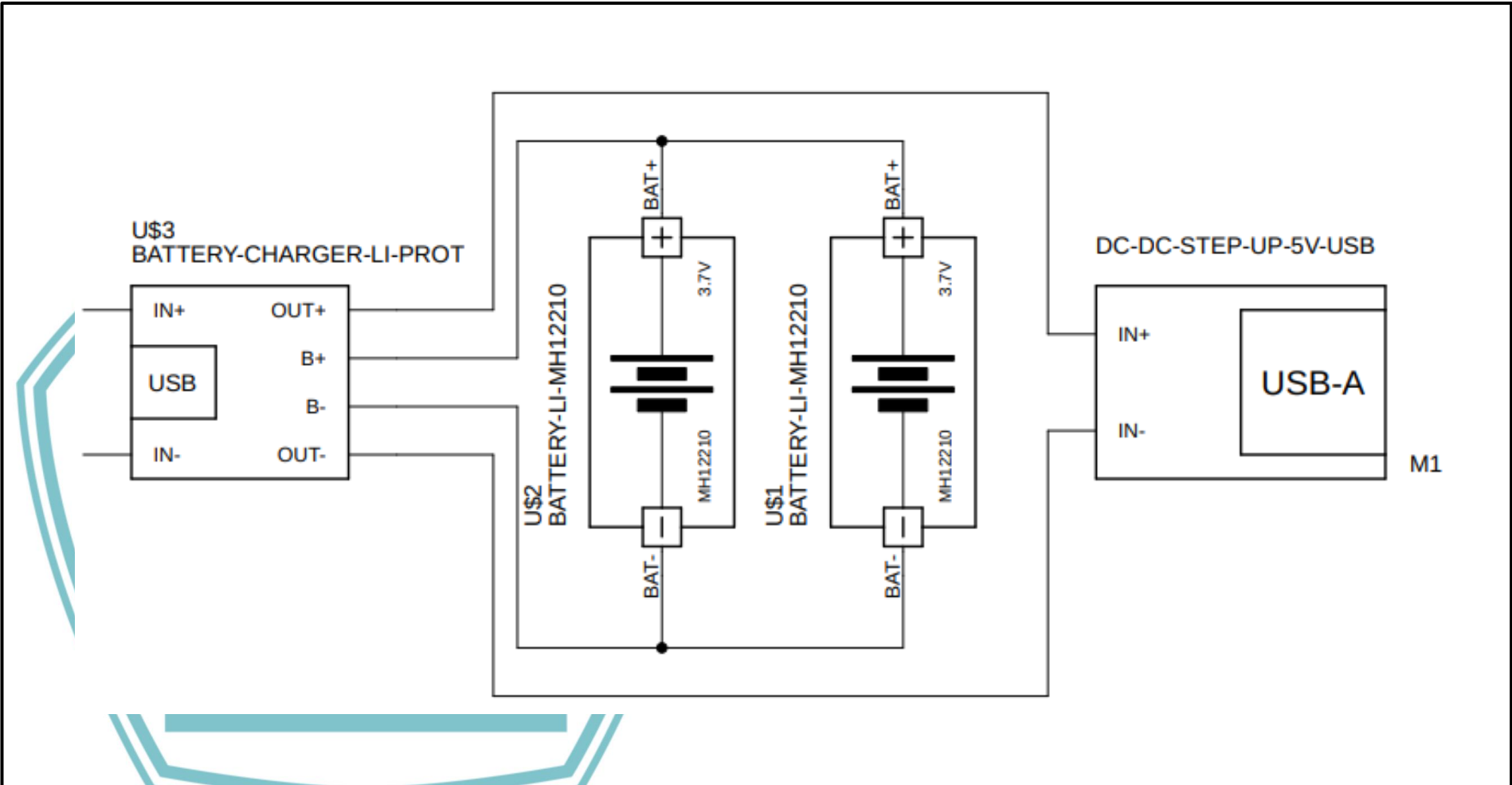
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



NEGERI
JAKARTA

02	SKEMATIK RANGKAIAN CATU DAYA	
 <p>PROGRAM STUDI TELEKOMUNIKASI</p> <p>JURUSAN TEKNIK ELEKTRO – POLITEKNIK NEGERI JAKARTA</p>	Digambar	: Syifa Eka Muliandari
	Diperiksa	: Yenniwati Rafsyam, SST., M.T.
	Tanggal	: Juli 2021





03

SKEMATIK RANGKAIAN MODUL POWER BANK



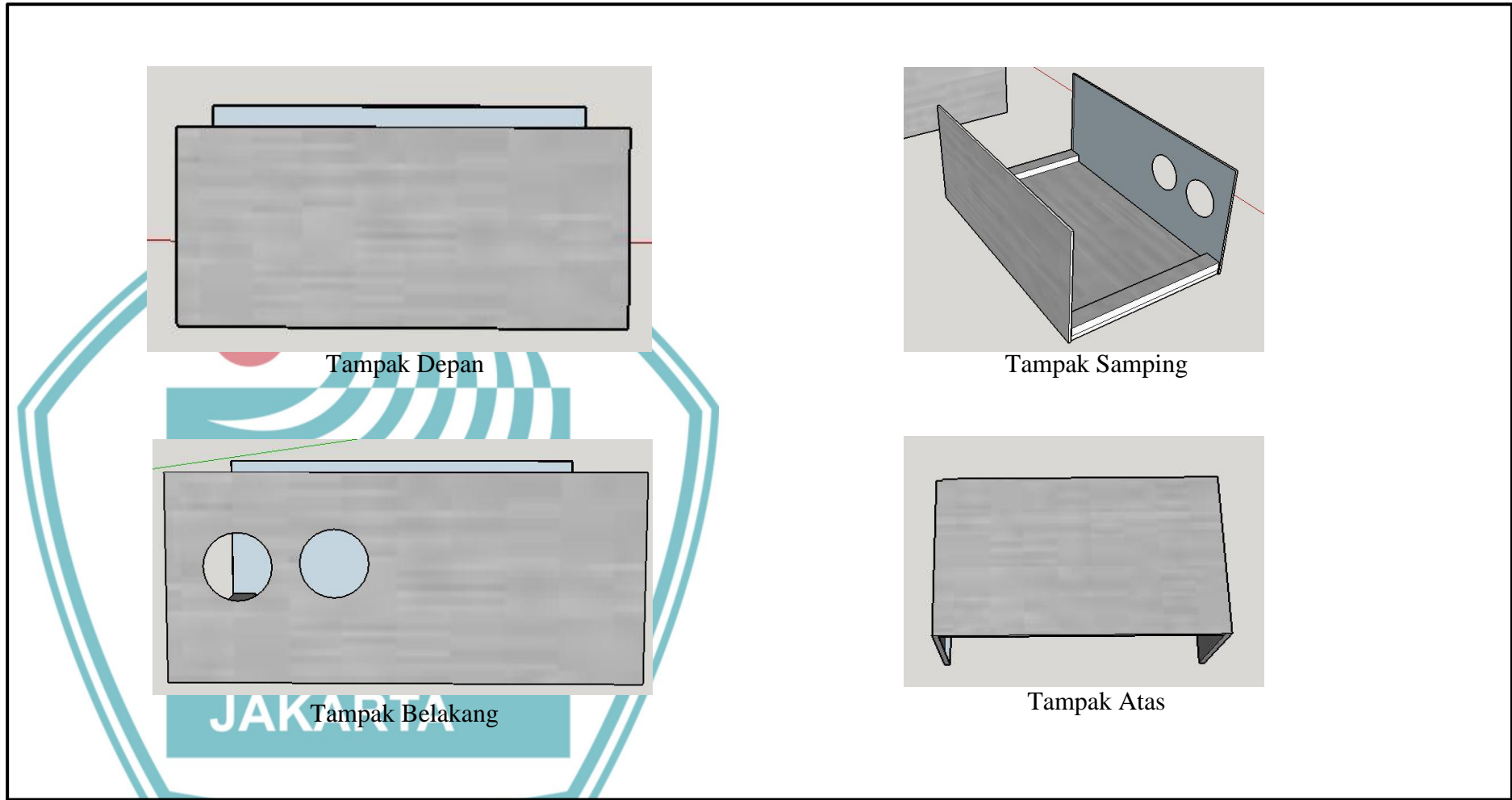
PROGRAM STUDI TELEKOMUNIKASI

JURUSAN TEKNIK ELEKTRO – POLITEKNIK NEGERI JAKARTA

Digambar	: Syifa Eka Muliandari
Diperiksa	: Yenniwarti Rafsyam, SST., M.T.
Tanggal	: Juli 2021

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



04	DESAIN CASING
-----------	----------------------

	PROGRAM STUDI TELEKOMUNIKASI	<i>Digambar</i>	<i>: Syifa Eka Muliandari</i>
	JURUSAN TEKNIK ELEKTRO – POLITEKNIK NEGERI	<i>Diperiksa</i>	<i>: Yenniwarti Rafsyam, SST., M.T.</i>
	JAKARTA	<i>Tanggal</i>	<i>: Juli 2021</i>

