



**PEMANFAATAN *ARTIFICIAL INTELLIGENCE* UNTUK
PEMANTAUAN PERTUMBUHAN TANAMAN HIDROPONIK
BERBASIS *INTERNET OF THINGS* (IOT)**

**“RANCANG BANGUN SISTEM PEMANTAUAN
PERTUMBUHAN TANAMAN HIDROPONIK”**

TUGAS AKHIR

Diajukan sebagai salah satu syarat untuk memperoleh gelar Diploma Tiga

FERRY NUR PRATAMA

2003332009

PROGRAM STUDI TELEKOMUNIKASI

JURUSAN TEKNIK ELEKTRO

POLITEKNIK NEGERI JAKARTA

2023



© 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



**PEMANFAATAN *ARTIFICIAL INTELLIGENCE* UNTUK
PEMANTAUAN PERTUMBUHAN TANAMAN HIDROPONIK
BERBASIS *INTERNET OF THINGS* (IOT)**

**“RANCANG BANGUN SISTEM PEMANTAUAN
PERTUMBUHAN TANAMAN HIDROPONIK”**

TUGAS AKHIR

Diajukan sebagai salah satu syarat untuk memperoleh gelar Diploma Tiga

FERRY NUR PRATAMA

2003332009

**PROGRAM STUDI TELEKOMUNIKASI
JURUSAN TEKNIK ELEKTRO
POLITEKNIK NEGERI JAKARTA
2023**



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

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 : Ferry Nur Pratama

NIM : 2003332009

Tanda Tangan :

Tanggal : 2023



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

HALAMAN PENGESAHAN TUGAS AKHIR

Tugas Akhir ini diajukan oleh :

Nama : Ferry Nur Pratama
NIM : 2003332009
Program Studi : Telekomunikasi
Judul Tugas Akhir : Pemanfaatan *Artificial Intelligence* untuk Pemantauan Pertumbuhan Tanaman Hidroponik berbasis *Internet of Things* (IoT)

Telah diuji oleh tim penguji dalam Sidang Tugas Akhir pada tanggal...9 Agustus...
2023 dan dinyatakan **LULUS**.

Pembimbing : Benny Nixon, S.T., M.T.
NIP. 196811072000031001

Depok, 23 Agustus 2023

Disahkan oleh
Ketua Jurusan Teknik Elektro




Rika Novia Wardhani, S.T., M.T.
NIP. 197011142008122001



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

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 berjudul “Pemanfaatan *Artificial Intelligence* untuk Pemantauan Pertumbuhan Tanaman Hidroponik berbasis *Internet of Things (IoT)*”. Penulis menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan tugas akhir ini, akan sulit bagi penulis untuk menyelesaikan tugas akhir ini. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Benny Nixon, S.T., 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. Shaffanah Ghaniyah Miranda selaku rekan dalam mengerjakan tugas akhir dan teman-teman di Program Studi Telekomunikasi Angkatan 2020 yang telah mendukung serta bekerja sama untuk menyelesaikan tugas akhir ini;
5. Bapak Zulfikar dan rekan-rekan Green Puspa Hidroponik yang telah memberikan bantuan dukungan dan material dalam menyelesaikan penelitian terkait tugas akhir penulis.

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

Depok, Juli 2023

Penulis



PEMANFAATAN ARTIFICIAL INTELLIGENCE UNTUK PEMANTAUAN PERTUMBUHAN TANAMAN HIDROPONIK BERBASIS *INTERNET OF THINGS* (IoT)

“Rancang Bangun Sistem Pemantauan Pertumbuhan Tanaman Hidroponik”

ABSTRAK

Tanaman hidroponik telah menjadi pilihan utama dalam mencapai produksi tanaman yang efisien dan berkelanjutan. Namun, tantangan dalam mengoptimalkan tanaman hidroponik seringkali memerlukan pemantauan dan pengendalian yang cermat dan efisien. Artificial Intelligence menjadi salah satu teknologi yang dapat mendukung solusi tersebut. Untuk itu dalam penelitian ini dibuat sistem yang memanfaatkan Raspberry Pi sebagai mikrokontroler, dan dapat memonitoring pertumbuhan tanaman hidroponik melalui aplikasi android. Kondisi tanaman hidroponik akan dipantau dengan LCD yang menampilkan nilai dari sensor suhu DS18B20, sensor TDS, dan sensor pH. Selain itu digunakan juga Relay 4 channel dengan Pompa DC 12 Volt sebagai alat untuk memompa cairan nutrisi dan cairan pH dan tanaman akan dipantau melalui web camera yang telah diprogram sehingga dapat mengukur ketinggian tanaman. Hasil pemantauan tanaman hidroponik dilakukan dengan melihat pada aplikasi android dan data akan tersimpan dalam database firebase melalui jaringan internet. Pada penelitian ini dilakukan pengujian output nilai dari sensor suhu, pH, TDS dengan membandingkan nilai sensor dengan alat ukur yaitu mendapatkan nilai rata rata akurasi >95%, pengujian ketinggian tanaman dilakukan dengan membandingkan nilai pada tampilan di histori aplikasi dengan ketinggian asli pada tanaman yaitu mendapatkan nilai rata rata akurasi >95%. Hal tersebut menunjukkan pengujian berjalan lancar dengan kategori baik.

Kata kunci : Artificial Intelligence, Hidroponik, pH, Raspberry Pi, Suhu, TDS.

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



UTILIZATION OF ARTIFICIAL INTELLIGENCE FOR GROWTH MONITORING OF HYDROPONIC PLANTS BASED INTERNET OF THINGS (IoT)

"Design and Development of Hydroponic Plant Growth Monitoring System"

ABSTRACT

Hydroponic plants have become the preferred choice for achieving efficient and sustainable crop production. However, optimizing hydroponic plants often requires careful and efficient monitoring and control. Artificial Intelligence has emerged as a technology that can support these solutions. Therefore, this research aims to develop a system that utilizes Raspberry Pi as a microcontroller and can monitor the growth of hydroponic plants through an Android application. The hydroponic plant's conditions will be monitored using an LCD that displays values from the DS18B20 temperature sensor, TDS sensor, and pH sensor. Additionally, a 4-channel Relay with a 12-Volt DC Pump will be used to pump nutrient and pH solutions, and plant growth will be monitored through a programmed web camera capable of measuring plant height. The hydroponic plant monitoring results will be accessible through an Android application, and the data will be stored in a Firebase database via the internet. In this study, testing will be conducted to validate the output values of the temperature, pH, and TDS sensors by comparing sensor values with measurement devices. The aim is to achieve an average accuracy >95%. Furthermore, plant height testing will involve comparing values displayed in the application's history with the actual plant height. The objective is to attain an average accuracy >95%. These tests demonstrate the smooth execution of the study with a good performance rating.

Key Words: Artificial Intelligence, Hydroponics, pH, Raspberry Pi, Temperature, TDS.

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



DAFTAR ISI

HALAMAN PERNYATAAN ORISINALITAS	ii
HALAMAN PENGESAHAN TUGAS AKHIR	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	v
ABSTRACT	vi
DAFTAR ISI	vii
DAFTAR GAMBAR.....	ix
DAFTAR TABEL	x
DAFTAR LAMPIRAN.....	xi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	2
1.4 Luaran	2
BAB II TINJAUAN PUSTAKA.....	3
2.1 Tanaman Hidroponik	3
2.2 Artificial Intelligence	3
2.3 Raspberry Pi	4
2.4 Python	6
2.5 Analog to Digital Converter Raspberry Pi	8
2.6 Sensor Suhu DS18B20	9
2.7 Sensor pH SEN0161	9
2.8 Sensor TDS SEN0244.....	9
2.9 Modul Relay 4 Channel	10
2.10 Pompa DC (R385 DC Pump).....	10
2.11 Modul LCD I2C 20x4	11
2.12 VNC Viewer.....	11
2.13 Webcam Logitech C310.....	12
2.14 Computer Vision	12
2.14.1 <i>OpenCV</i>	13
2.14.2 <i>Color Image Processing</i>	13
2.14.3 <i>HSV</i>	13
2.14.4 <i>Contours</i>	13
2.14.5 <i>Bounding Box</i>	14
2.15 Thonny Python IDE	14
BAB III RANCANGAN DAN REALISASI	15
3.1 Rancangan Alat	15
3.1.1 Deskripsi Alat	15
3.1.2 Cara Kerja Sistem	16
3.2 Spesifikasi Sistem	18
3.3 Diagram Blok	19
3.4 Rancangan Sistem	19
3.4.1 Realisasi Sistem <i>monitoring</i> pertumbuhan tanaman hidroponik	19
3.4.2 Realisasi Sensor Suhu DS18B20	20
3.4.3 Realisasi Sensor pH SEN0161	21
3.4.4 Realisasi Sensor TDS SEN0244	21

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

3.4.5	Realisasi <i>Relay 4 Channel</i>	22
3.4.6	Realisasi <i>Output Relay</i> dengan Pompa DC.....	23
3.4.7	Realisasi <i>WebCam</i>	24
3.4.8	Realisasi Pemrograman.....	25
3.4.9	Realisasi <i>Catu Daya (Power Supply)</i>	43
BAB IV PEMBAHASAN.....		45
4.1	Pengujian alat	45
4.2	Pengujian <i>Catu Daya</i>	45
4.2.1	Prosedur Pengujian <i>Catu Daya</i>	45
4.2.2	Data Hasil Pengujian <i>Catu Daya</i>	47
4.2.3	Analisis Data / Evaluasi	47
4.3	Pengujian Program <i>Thonny Python IDE</i>	47
4.3.1	Deskripsi Pengujian	47
4.3.2	Prosedur Pengujian	47
4.3.3	Pengujian Sensor Suhu DS18B20.....	48
4.3.4	Pengujian Sensor pH SEN0161	49
4.3.5	Pengujian Sensor TDS SEN0244.....	51
4.3.6	Pengujian <i>Relay 4 Channel</i> dengan Pompa DC.....	52
4.3.7	Pengujian LCD.....	54
4.3.8	Pengujian <i>Color Image Processing</i>	55
BAB V PENUTUP.....		57
5.1	Simpulan	57
5.2	Saran.....	57
DAFTAR PUSTAKA		58
DAFTAR RIWAYAT HIDUP PENULIS		59
LAMPIRAN.....		60

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

DAFTAR GAMBAR

Gambar 2. 1 Tanaman Hidroponik	3
Gambar 2. 2 <i>Smart</i> Hidroponik.....	4
Gambar 2. 3 Raspbery Pi 3 Model B+	4
Gambar 2. 4 GPIO Pin Raspbery Pi 3	5
Gambar 2. 5 Logo Pemrograman <i>Python</i>	7
Gambar 2. 6 <i>Analog to Digital Converter</i> Raspbery Pi.....	8
Gambar 2. 7 Sensor Suhu DS18B20	9
Gambar 2. 8 Sensor pH Air SEN0161	9
Gambar 2. 9 Sensor TDS SEN0244.....	10
Gambar 2. 10 Modul <i>Relay 4 Channel</i>	10
Gambar 2. 11 R385 DC <i>Pump</i>	10
Gambar 2. 12 LCD 20x4 I2C.....	11
Gambar 2. 13 Logo VNC <i>Viewer</i>	11
Gambar 2. 14 <i>Webcam</i> Logitech C310	12
Gambar 2. 15 HSV Model	13
Gambar 2. 16 <i>Bounding Box</i>	14
Gambar 2. 17 Logo <i>Thonny Python IDE</i>	14
Gambar 3. 1 Ilustrasi kerja sistem secara keseluruhan	16
Gambar 3.2 Diagram alir kerja sistem	17
Gambar 3.3 Diagram blok <i>monitoring</i> pertumbuhan tanaman hidroponik.....	19
Gambar 3. 4 Sistem <i>monitoring</i> pertumbuhan tanaman hidroponik.....	20
Gambar 3. 5 Realisasi Sensor Suhu DS18B20	20
Gambar 3. 6 Realisasi Sensor pH SEN0161	21
Gambar 3. 7 Realisasi Sensor TDS SEN0244	22
Gambar 3. 8 Realisasi Modul <i>Relay 4 Channel</i>	22
Gambar 3. 9 Realisasi <i>Pin Output Relay 4 Channel</i> dengan Pompa DC.....	23
Gambar 3. 10 Rangkaian skematik kamera <i>webcam</i> pada Raspbery Pi	25
Gambar 3. 11 Diagram Alir Pemrograman.....	25
Gambar 3. 12 <i>Install Library</i> Pemrograman <i>Python</i>	26
Gambar 3. 13 Diagram Skematik Catu Daya.....	43
Gambar 4. 1 Pengukuran tegangan listrik PLN	46
Gambar 4. 2 Pengukuran tegangan keluaran catu daya	46
Gambar 4. 3 <i>Run</i> program pada <i>software Thonny Python IDE</i>	48
Gambar 4. 4 Hasil pembacaan sensor suhu air	48
Gambar 4. 5 Hasil pembacaan sensor pH pada air detergen.....	49
Gambar 4. 6 Hasil pembacaan sensor pH pada air Nutrisi AB Mix	51
Gambar 4. 7 Indikator <i>relay 4 channel</i> untuk 4 pompa DC.....	52
Gambar 4. 8 Tampilan Karakter LCD	54
Gambar 4. 9 Tampilan <i>Image Processing</i>	55



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

DAFTAR TABEL

Tabel 2. 1 Tipe Data.....	7
Tabel 3.1 Spesifikasi alat sistem.....	18
Tabel 3.2 Hubungan Pin Sensor Suhu DS18B20 dengan Raspberry Pi	21
Tabel 3.3 Hubungan Pin Sensor pH SEN0161 dengan Raspberry Pi.....	21
Tabel 3.4 Hubungan Pin Sensor TDS SEN0244 dengan Raspberry Pi	22
Tabel 3.5 Hubungan Pin <i>Input Relay 4 Channel</i> dengan Raspberry Pi	23
Tabel 3.6 Hubungan Pin <i>Output Relay 4 Channel</i> dengan Pompa DC dan konektor Jack DC <i>Female</i>	24
Tabel 3. 7 Hubungan <i>Port USB WebCam</i> dengan <i>Port USB</i> Raspberry Pi	25
Tabel 4. 1 Data Hasil Pengujian Catu Daya.....	47
Tabel 4. 2 Pengujian sensor suhu DS18B20.....	49
Tabel 4. 3 Pengujian sensor pH SEN0161	50
Tabel 4. 4 Pengujian sensor TDS SEN0244	51
Tabel 4. 5 Pengujian <i>Relay 4 Channel</i> dengan Pompa DC.....	53
Tabel 4. 6 Pengujian LCD.....	54
Tabel 4. 7 Pengujian <i>color image processing</i>	56

**POLITEKNIK
NEGERI
JAKARTA**



DAFTAR LAMPIRAN

L- 1 Rangkaian Skematik Catu Daya.....	61
L- 2 Rangkaian Skematik Sistem.....	62
L- 3 Ilustrasi <i>Casing</i> Sistem.....	63
L- 4 Ilustrasi Keseluruhan Sistem.....	64
L- 5 <i>Datasheet</i> Raspberry Pi 3B+.....	65
L- 6 <i>Datasheet</i> Sensor Suhu DS18B20.....	66
L- 7 <i>Datasheet</i> Sensor pH air SEN0161.....	67
L- 8 <i>Datasheet</i> Sensor TDS SEN0244.....	68
L- 9 <i>Datasheet</i> ADC DS1115.....	69
L- 10 <i>Datasheet</i> WebCam Logitech C310.....	70
L- 11 <i>Datasheet</i> Relay 4 Channel.....	71
L- 12 <i>Datasheet</i> R385 DC Pump.....	72
L- 13 <i>Datasheet</i> LCD 2004 I2C.....	74
L- 14 Kode Program.....	75

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

BAB I PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi pertanian menjadi semakin penting di Indonesia seiring dengan pertumbuhan populasi yang signifikan. Saat ini, sektor pertanian masih menjadi salah satu kontributor utama terhadap perekonomian Indonesia, dengan menyediakan mata pencaharian bagi sebagian besar penduduk di pedesaan. Namun, lahan pertanian yang tersedia terus mengalami penyusutan akibat urbanisasi dan pengembangan infrastruktur. Dalam beberapa dekade terakhir, lahan pertanian di Indonesia telah mengalami penyusutan yang cukup signifikan. Menurut data Badan Pusat Statistik (BPS), pada tahun 1990, lahan pertanian di Indonesia mencapai sekitar 17 juta hektar. Pada tahun 2023, lahan pertanian menjadi sekitar 10,61 juta hektar yang mengalami peningkatan sebanyak 194,71 ribu hektar atau 1,87 % dibandingkan tahun 2021 sebesar 10,41 Juta Hektar. Sementara itu, pertumbuhan populasi yang terus meningkat menambah tekanan pada ketersediaan pangan dan kebutuhan pertanian. Dalam menghadapi tantangan ini, banyak petani di Indonesia beralih ke metode pertanian alternatif, seperti hidroponik, untuk mengoptimalkan penggunaan lahan yang terbatas. Metode hidroponik memungkinkan tanaman tumbuh tanpa perlu menggunakan lahan pertanian tradisional.

Berdasarkan pada permasalahan diatas, maka pada topik Tugas Akhir ini penulis mengusulkan Pemantauan Tanaman Hidroponik berbasis *Internet of Things* (IoT). Perancangan sistem tersebut berfungsi untuk memantau pertumbuhan tanaman hidroponik. Sistem dibangun dengan menggunakan Raspberry Pi sebagai mikrokontroler, yang diintegrasikan menggunakan *Web Camera* dengan memanfaatkan *Artificial Intelligence* untuk monitoring ketinggian Tanaman Hidroponik. Selain itu untuk pemantauan tanaman digunakan alat sensor yang menampilkan nilai data yang berasal dari Sensor Suhu, Sensor pH, dan Sensor TDS, dan menggunakan Aplikasi Android yang terintegrasikan secara *realtime* dengan *database* pada *Firebase*.

Atas dasar uraian diatas, penulis merancang alat tugas akhir yang berjudul “Pemanfaatan *Artificial Intelligence* untuk Pemantauan Pertumbuhan Tanaman



© 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 mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Hidroponik berbasis *Internet of Things* (IoT)”. Dengan demikian, adanya alat tersebut dapat membantu meningkatkan produktivitas pertanian, mengatasi keterbatasan lahan, dan berkontribusi pada ketahanan pangan di masa depan, sesuai dengan pertumbuhan populasi yang terus meningkat di Indonesia.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah di uraikan di atas, maka rumusan masalah yang akan dibahas dalam tugas akhir ini adalah sebagai berikut :

1. Bagaimana cara merancang dan membuat sistem pemantau Suhu, pH, dan Larutan Nutrisi pada Tanaman Hidroponik?
2. Bagaimana cara merancang dan membuat sistem pemantau tinggi Tanaman Hidroponik dengan *Color Image Processing*?
3. Bagaimana cara mengirimkan respon untuk sistem pemantauan pertumbuhan Tanaman Hidroponik berbasis *Internet of Things* pada aplikasi?

1.3 Tujuan

Adapun tujuan yang ingin dicapai dari tugas akhir ini adalah :

1. Mampu merancang dan membuat sistem pemantau suhu, pH, dan kandungan Nutrisi yang terhubung oleh LCD pada Tanaman Hidroponik
2. Mampu merancang dan membuat sistem pengukur ketinggian Tanaman Hidroponik menggunakan *Color Image Processing*
3. Dapat mengirimkan respon untuk sistem *monitoring* pertumbuhan Tanaman Hidroponik berbasis *Internet of Things* pada aplikasi.

1.4 Luaran

Sistem Pemantauan Pertumbuhan Tanaman Hidroponik berbasis *Internet of Things* (IoT) diharapkan dapat membantu para petani. Dengan begitu dapat meningkatkan produktivitas UMKM Tanaman Hidroponik dan terpenuhinya produksi pangan untuk masyarakat. Adapun luaran yang hendak dicapai dalam tugas akhir ini adalah:

1. Alat untuk Sistem Pemantauan Pertumbuhan Tanaman Hidroponik berbasis *Internet of Things* (IoT).
2. Laporan tugas akhir.
3. Artikel ilmiah.

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

BAB V PENUTUP

5.1 Simpulan

Berdasarkan perancangan dan hasil pengujian dari alat tugas akhir yang telah dibuat, maka dapat disimpulkan bahwa :

1. Sistem pemantauan suhu, pH, dan kandungan nutrisi serta LCD pada Tanaman Hidroponik telah berhasil dirancang dan dibuat dengan baik. Selama pengujian, sistem ini mampu mengukur dan memantau melalui sensor tersebut. Pengujian Sensor Suhu mendapatkan nilai rata – rata akurasi 99.38%, Sensor pH diuji menggunakan air nutrisi AB Mix yang memiliki sifat asam dengan tingkat akurasi pengujian 97.33% sedangkan air sabun memiliki sifat basa dengan tingkat akurasi pengujian 96.71% dan Sensor TDS diuji menggunakan air nutrisi AB Mix dengan tingkat akurasi 99.96%, lalu air mineral memiliki tingkat akurasi 99.6%. LCD dapat menampilkan nilai dari sensor suhu, pH, dan TDS, dan untuk pengujian *electrical* dari masing-masing sensor mendapatkan nilai 4.75 V.
2. Sistem pengukur ketinggian tanaman hidroponik yang menggunakan teknologi *Artificial Intelligence* (AI) dengan metode *Color Image Processing* telah berhasil dibuat dan diimplementasikan. Hasil pengujian menunjukkan bahwa pengukuran ketinggian tanaman dengan *Artificial Intelligence* (AI) memiliki tingkat akurasi yang masih dalam kategori baik, yakni mendapatkan nilai rata – rata akurasi 97.06 %.
3. Sistem monitoring pertumbuhan tanaman hidroponik yang berbasis *Internet of Things* (IoT) berhasil diimplementasikan dan diuji. Selama pengujian, sistem ini mampu mengirimkan data secara cepat dan konsisten melalui *database* pada *firebase* yang terhubung dengan aplikasi android.

5.2 Saran

Diharapkan ide dan gagasan baru yang tertuang dalam Tugas Akhir Pemanfaatan *Artificial Intelligence* untuk pertumbuhan tanaman hidroponik berbasis *Internet of Things* (IoT) ini dapat diaplikasikan pada petani khususnya petani kota secara masif dan dapat membuat tanaman hidroponik lebih mudah dikontrol dan dipantau dan sepenuhnya otomatis.



DAFTAR PUSTAKA

- Firdaus, M. R., & Susanto, A. D. (2021). Analisis Pengolahan Citra Digital untuk Estimasi Ketinggian Tanaman. *Jurnal Informatika*.
- Hakim, L., & Fathoni, M. I. (2019). Sistem Otomatisasi Greenhouse Tanaman Hidroponik Menggunakan Raspberry Pi. *Jurnal Teknik Elektro*.
- Istiqomah, N. A., & Muslim, M. A. (2019). Sistem Monitoring pH Tanaman Hidroponik Berbasis IoT. *Jurnal Teknik Elektro dan Informatika*.
- Kurniawan, F. R., & Nasution, D. S. (2018). Pengenalan Citra Tanaman Hidroponik Berbasis Raspberry Pi. *Jurnal Nasional Teknik Elektro dan Teknologi Informasi*.
- Mardiyanto, A. W., & Nurhayati, S. (2020). Rancang Bangun Sistem Monitoring dan Kontrol Budidaya Tanaman Hidroponik Berbasis Raspberry Pi. *Jurnal Teknik Elektro dan Komputer*.
- Rosyadi, D. A., & Apriyanto, E. (2020). Sistem Monitoring dan Kontrol Hidroponik dengan Raspberry Pi Berbasis Internet of Things. *Jurnal Teknik Elektro*.
- Saputra, D., & Iskandar, B. P. (2017). Rancang Bangun Sistem Monitoring Suhu, Kelembaban dan Nutrisi Hidroponik Berbasis Mikrokontroler. *Jurnal Teknik Elektro dan Komputer*.
- Sugiyanto, A. I., & Nurhadirah, I. (2020). Pengolahan Citra untuk Deteksi Warna Daun Pada Sistem Pengenalan Tanaman Berbasis Mikrokontroler. *Jurnal Ilmiah Teknologi Informasi Terapan*.
- Yulianto, T., & Susanto, A. D. (2021). Rancang Bangun Sistem Pemantauan pH Air pada Sistem Budidaya Hidroponik Menggunakan Mikrokontroler Arduino. *Jurnal Nasional Teknik Elektro*.
- Zainuddin, R., Nuzula, N. A., & Nugraha, R. P. (2020). Pengembangan Sistem Kontrol Nutrisi Hidroponik Otomatis dengan Sensor TDS (Total Dissolved Solids). *Jurnal Teknologi Elektro*.

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



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 RIWAYAT HIDUP PENULIS



Ferry Nur Pratama

Lulus dari SDN Kapuk 09 Pagi tahun 2014, SMPN 201 Jakarta Barat tahun tahun 2017, dan SMK Telkom Sandhy Putra Jakarta tahun 2020, Gelar Diploma Tiga (D3) diperoleh pada tahun 2023 dari jurusan Teknik Elektro, Program Studi Telekomunikasi, Politeknik Negeri Jakarta



LAMPIRAN

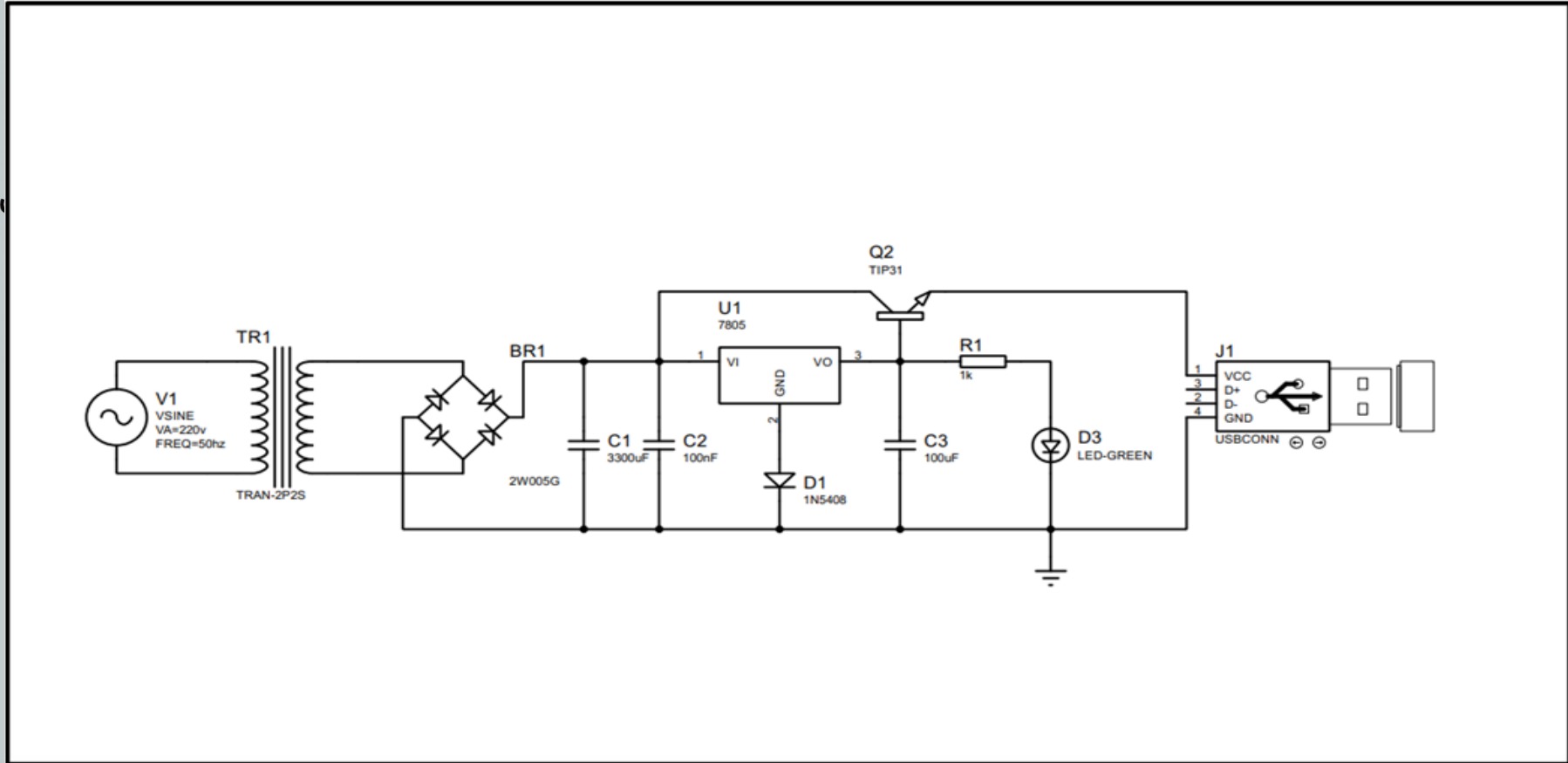


© Hak Cipta milik Politeknik Negeri Jakarta

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 satu 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

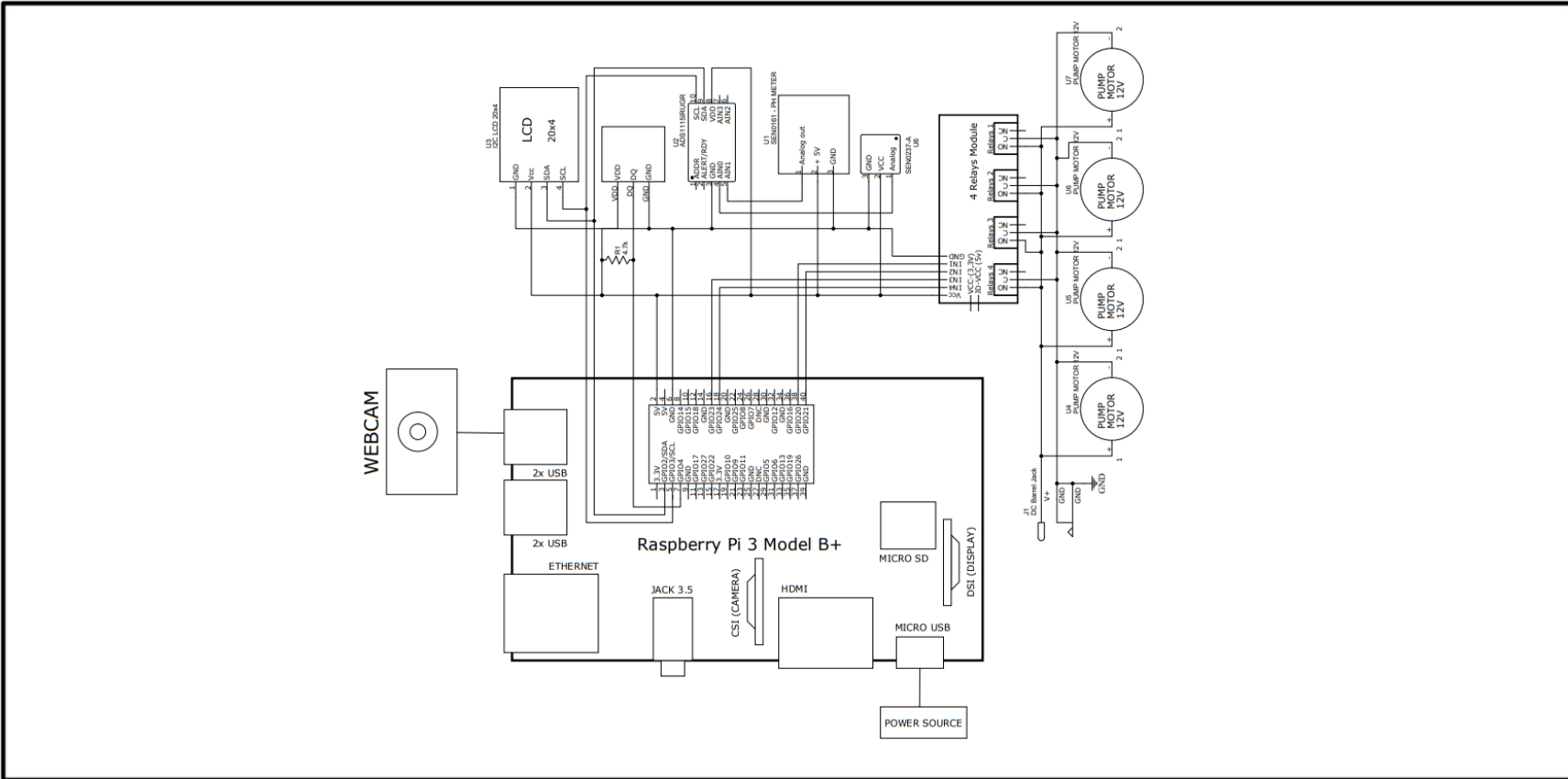




01	Rangkaian Skematik Catu Daya
----	------------------------------

	PROGRAM STUDI TELEKOMUNIKASI	<i>Digambar</i>	Ferry Nur Pratama
	JURUSAN TEKNIK ELEKTRO – POLITEKNIK NEGERI JAKARTA	<i>Diperiksa</i>	Benny Nixon S.T, M.T
		<i>Tanggal</i>	26 Juli 2023

Diarangi 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 ma
 b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
 Dilarang mengumumkkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun
 tanpa izin Politeknik Negeri Jakarta



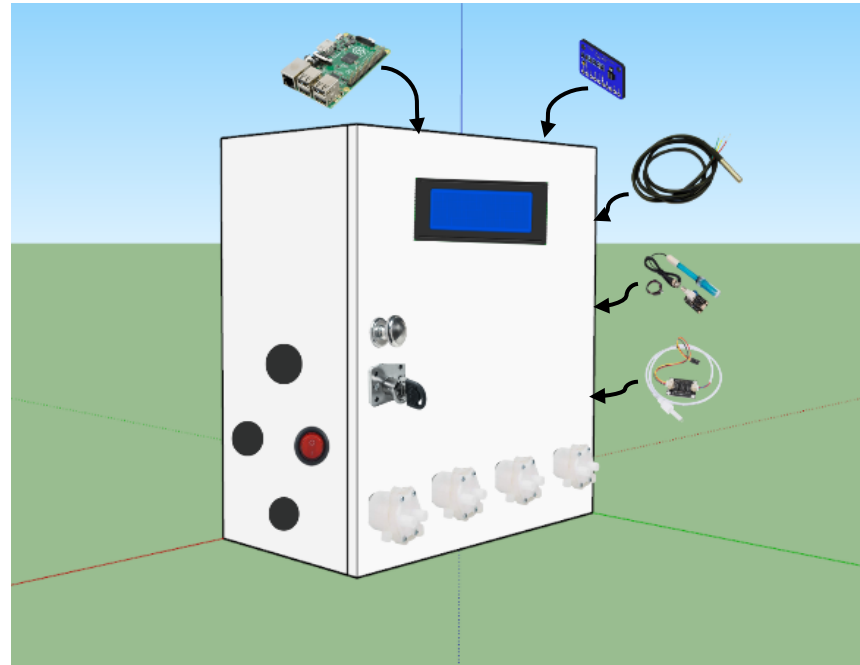
02

Rangkaian Skematik Sistem Monitoring Tanaman Hidroponik



PROGRAM STUDI TELEKOMUNIKASI
 JURUSAN TEKNIK ELEKTRO – POLITEKNIK NEGERI JAKARTA

Digambar	Ferry Nur Pratama
Diperiksa	Benny Nixon S.T., M.T.
Tanggal	26 Juli 2023



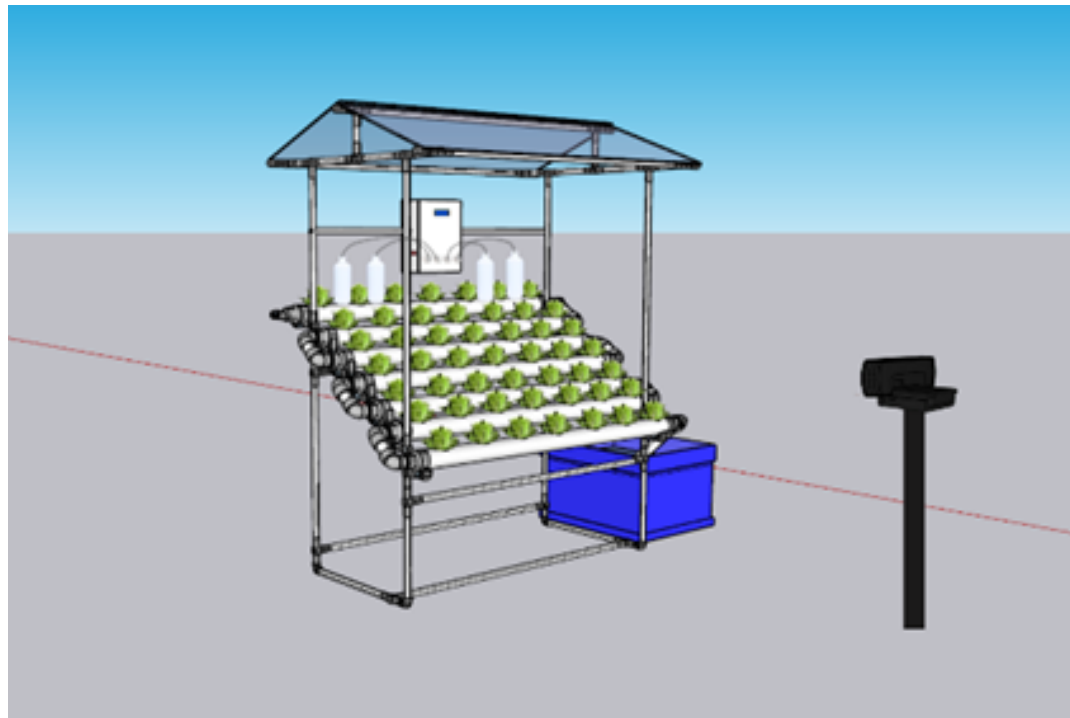
03

ILUSTRASI CASING SISTEM



PROGRAM STUDI TELEKOMUNIKASI
JURUSAN TEKNIK ELEKTRO – POLITEKNIK NEGERI JAKARTA

<i>Digambar</i>	Ferry Nur Pratama
<i>Diperiksa</i>	Benny Nixon S.T., M.T.
<i>Tanggal</i>	26 Juli 2023



04

ILUSTRASI KESELURUHAN



**PROGRAM STUDI TELEKOMUNIKASI
JURUSAN TEKNIK ELEKTRO – POLITEKNIK NEGERI JAKARTA**

<i>Digambar</i>	Ferry Nur Pratama
<i>Diperiksa</i>	Benny Nixon S.T., M.T.
<i>Tanggal</i>	26 Juli 2023

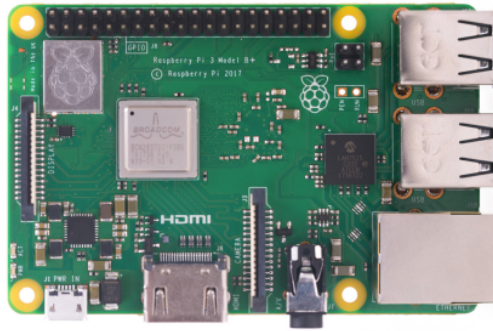
milik Polit
 Cipta :
 larang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah
 Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
 larang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun
 tanpa izin Politeknik Negeri Jakarta



Datasheet Raspberry Pi 3 Model B+

- 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

Overview



The Raspberry Pi 3 Model B+ is the latest product in the Raspberry Pi 3 range, boasting a 64-bit quad core processor running at 1.4GHz, dual-band 2.4GHz and 5GHz wireless LAN, Bluetooth 4.2/BLE, faster Ethernet, and PoE capability via a separate PoE HAT

The dual-band wireless LAN comes with modular compliance certification, allowing the board to be designed into end products with significantly reduced wireless LAN compliance testing, improving both cost and time to market.

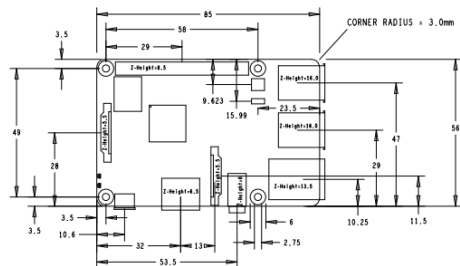
The Raspberry Pi 3 Model B+ maintains the same mechanical footprint as both the Raspberry Pi 2 Model B and the Raspberry Pi 3 Model B.

Specifications

Processor:	Broadcom BCM2837B0, Cortex-A53 64-bit SoC @ 1.4GHz
Memory:	1GB LPDDR2 SDRAM
Connectivity:	<ul style="list-style-type: none"> ■ 2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE ■ Gigabit Ethernet over USB 2.0 (maximum throughput 300Mbps) ■ 4 x USB 2.0 ports
Access:	Extended 40-pin GPIO header
Video & sound:	<ul style="list-style-type: none"> ■ 1 x full size HDMI ■ MIPI DSI display port ■ MIPI CSI camera port ■ 4 pole stereo output and composite video port
Multimedia:	H.264, MPEG-4 decode (1080p30); H.264 encode (1080p30); OpenGL ES 1.1, 2.0 graphics
SD card support:	Micro SD format for loading operating system and data storage
Input power:	<ul style="list-style-type: none"> ■ 5V/2.5A DC via micro USB connector ■ 5V DC via GPIO header ■ Power over Ethernet (PoE)-enabled (requires separate PoE HAT)
Environment:	Operating temperature, 0–50 °C
Compliance:	For a full list of local and regional product approvals, please visit www.raspberrypi.org/products/raspberry-pi-3-model-b+
Production lifetime:	The Raspberry Pi 3 Model B+ will remain in production until at least January 2023.



Physical specifications



Warnings

- This product should only be connected to an external power supply rated at 5V/2.5A DC. Any external power supply used with the Raspberry Pi 3 Model B+ shall comply with relevant regulations and standards applicable in the country of intended use.
- This product should be operated in a well-ventilated environment and, if used inside a case, the case should not be covered.
- Whilst in use, this product should be placed on a stable, flat, non-conductive surface and should not be contacted by conductive items.
- The connection of incompatible devices to the GPIO connection may affect compliance, result in damage to the unit, and invalidate the warranty.
- All peripherals used with this product should comply with relevant standards for the country of use and be marked accordingly to ensure that safety and performance requirements are met. These articles include but are not limited to keyboards, monitors, and mice when used in conjunction with the Raspberry Pi.
- The cables and connectors of all peripherals used with this product must have adequate insulation so that relevant safety requirements are met.

Safety instructions

- To avoid malfunction or damage to this product, please observe the following:
- Do not expose to water or moisture, or place on a conductive surface whilst in operation.
 - Do not expose to heat from any source; the Raspberry Pi 3 Model B+ is designed for reliable operation at normal ambient temperatures.
 - Take care whilst handling to avoid mechanical or electrical damage to the printed circuit board and connectors.
 - Whilst it is powered, avoid handling the printed circuit board, or only handle it by the edges to minimise the risk of electrostatic discharge damage.



Datasheet Sensor DS18B20

- 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

PRELIMINARY

DS18B20
Programmable Resolution
1-Wire® Digital Thermometer

www.dallassemi.com

FEATURES

- Unique 1-Wire interface requires only one port pin for communication
- Multidrop capability simplifies distributed temperature sensing applications
- Requires no external components
- Can be powered from data line. Power supply range is 3.0V to 5.5V
- Zero standby power required
- Measures temperatures from -55°C to +125°C. Fahrenheit equivalent is -67°F to +257°F
- ±0.5°C accuracy from -10°C to +85°C
- Thermometer resolution is programmable from 9 to 12 bits
- Converts 12-bit temperature to digital word in 750 ms (max.)
- User-definable, nonvolatile temperature alarm settings
- Alarm search command identifies and addresses devices whose temperature is outside of programmed limits (temperature alarm condition)
- Applications include thermostatic controls, industrial systems, consumer products, thermometers, or any thermally sensitive system

DESCRIPTION

The DS18B20 Digital Thermometer provides 9 to 12-bit (configurable) temperature readings which indicate the temperature of the device.

Information is sent to/from the DS18B20 over a 1-Wire interface, so that only one wire (and ground) needs to be connected from a central microprocessor to a DS18B20. Power for reading, writing, and performing temperature conversions can be derived from the data line itself with no need for an external power source.

Because each DS18B20 contains a unique silicon serial number, multiple DS18B20s can exist on the same 1-Wire bus. This allows for placing temperature sensors in many different places. Applications where this feature is useful include HVAC environmental controls, sensing temperatures inside buildings, equipment or machinery, and process monitoring and control.

PIN ASSIGNMENT

PIN DESCRIPTION

GND - Ground
DQ - Data In/Out
V_{CC} - Power Supply Voltage
NC - No Connect

1 of 27 050400

DS18B20

DETAILED PIN DESCRIPTION Table 1

PIN	PIN	SYMBOL	DESCRIPTION
5	1	GND	Ground.
4	2	DQ	Data Input/Output pin. For 1-Wire operation: Open drain. (See "Parasite Power" section.)
3	3	V _{DD}	Optional V_{DD} pin. See "Parasite Power" section for details of connection. V _{DD} must be grounded for operation in parasite power mode.

DS18B20Z (8-pin SOIC). All pins not specified in this table are not to be connected.

OVERVIEW

The block diagram of Figure 1 shows the major components of the DS18B20. The DS18B20 has four main data components: 1) 64-bit lasered ROM, 2) temperature sensor, 3) nonvolatile temperature alarm triggers TH and TL, and 4) a configuration register. The device derives its power from the 1-Wire communication line by storing energy on an internal capacitor during periods of time when the signal line is high and continues to operate off this power source during the low times of the 1-Wire line until it returns high to replenish the parasitic (capacitor) supply. As an alternative, the DS18B20 may also be powered from an external 3 volt - 5.5 volt supply.

Communication to the DS18B20 is via a 1-Wire port. With the 1-Wire port, the memory and control functions will not be available before the ROM function protocol has been established. The master must first provide one of five ROM function commands: 1) Read ROM, 2) Match ROM, 3) Search ROM, 4) Skip ROM, or 5) Alarm Search. These commands operate on the 64-bit lasered ROM portion of each device and can single out a specific device if many are present on the 1-Wire line as well as indicate to the bus master how many and what types of devices are present. After a ROM function sequence has been successfully executed, the memory and control functions are accessible and the master may then provide any one of the six memory and control function commands.

One control function command instructs the DS18B20 to perform a temperature measurement. The result of this measurement will be placed in the DS18B20's scratch-pad memory, and may be read by issuing a memory function command which reads the contents of the scratchpad memory. The temperature alarm triggers TH and TL consist of 1 byte EEPROM each. If the alarm search command is not applied to the DS18B20, these registers may be used as general purpose user memory. The scratchpad also contains a configuration byte to set the desired resolution of the temperature to digital conversion. Writing TH, TL, and the configuration byte is done using a memory function command. Read access to these registers is through the scratchpad. All data is read and written least significant bit first.

DS18B20

For situations where the bus master does not know whether the DS18B20s on the bus are parasite powered or supplied with external V_{DD}, a provision is made in the DS18B20 to signal the power supply scheme used. The bus master can determine if any DS18B20s are on the bus which require the strong pullup by sending a Skip ROM protocol, then issuing the read power supply command. After this command is issued, the master then issues read time slots. The DS18B20 will send back "0" on the 1-Wire bus if it is parasite powered; it will send back a "1" if it is powered from the V_{DD} pin. If the master receives a "0," it knows that it must supply the strong pullup on the DQ line during temperature conversions. See "Memory Command Functions" section for more detail on this command protocol.

STRONG PULLUP FOR SUPPLYING DS18B20 DURING TEMPERATURE CONVERSION Figure 2

USING V_{DD} TO SUPPLY TEMPERATURE CONVERSION CURRENT Figure 3

4 of 27

DS18B20

DS18B20 BLOCK DIAGRAM Figure 1

PARASITE POWER

The block diagram (Figure 1) shows the parasite-powered circuitry. This circuitry "steals" power whenever the DQ or V_{DD} pins are high. DQ will provide sufficient power as long as the specified timing and voltage requirements are met (see the section titled "1-Wire Bus System"). The advantages of parasite power are twofold: 1) by parasitizing off this pin, no local power source is needed for remote sensing of temperature, and 2) the ROM may be read in absence of normal power.

In order for the DS18B20 to be able to perform accurate temperature conversions, sufficient power must be provided over the DQ line when a temperature conversion is taking place. Since the operating current of the DS18B20 is up to 1.5 mA, the DQ line will not have sufficient drive due to the 5k pullup resistor. This problem is particularly acute if several DS18B20s are on the same DQ and attempting to convert simultaneously.

There are two ways to assure that the DS18B20 has sufficient supply current during its active conversion cycle. The first is to provide a strong pullup on the DQ line whenever temperature conversions or copies to the E² memory are taking place. This may be accomplished by using a MOSFET to pull the DQ line directly to the power supply as shown in Figure 2. The DQ line must be switched over to the strong pull-up within 10 µs maximum after issuing any protocol that involves copying to the E² memory or initiates temperature conversions. When using the parasite power mode, the V_{DD} pin must be tied to ground.

Another method of supplying current to the DS18B20 is through the use of an external power supply tied to the V_{DD} pin, as shown in Figure 3. The advantage to this is that the strong pullup is not required on the DQ line, and the bus master need not be tied up holding that line high during temperature conversions. This allows other data traffic on the 1-Wire bus during the conversion time. In addition, any number of DS18B20s may be placed on the 1-Wire bus, and if they all use external power, they may all simultaneously perform temperature conversions by issuing the Skip ROM command and then issuing the Convert T command. Note that as long as the external power supply is active, the GND pin may not be floating.

The use of parasite power is not recommended above 100°C, since it may not be able to sustain communications given the higher leakage currents the DS18B20 exhibits at these temperatures. For applications in which such temperatures are likely, it is strongly recommended that V_{DD} be applied to the DS18B20.

3 of 27

DS18B20

For situations where the bus master does not know whether the DS18B20s on the bus are parasite powered or supplied with external V_{DD}, a provision is made in the DS18B20 to signal the power supply scheme used. The bus master can determine if any DS18B20s are on the bus which require the strong pullup by sending a Skip ROM protocol, then issuing the read power supply command. After this command is issued, the master then issues read time slots. The DS18B20 will send back "0" on the 1-Wire bus if it is parasite powered; it will send back a "1" if it is powered from the V_{DD} pin. If the master receives a "0," it knows that it must supply the strong pullup on the DQ line during temperature conversions. See "Memory Command Functions" section for more detail on this command protocol.

STRONG PULLUP FOR SUPPLYING DS18B20 DURING TEMPERATURE CONVERSION Figure 2

USING V_{DD} TO SUPPLY TEMPERATURE CONVERSION CURRENT Figure 3

4 of 27



Datasheet Sensor Ph air SEN0161



PH meter(SKU: SEN0161)



Analog pH Meter Kit SKU: SEN0161



Analog pH Meter Kit SKU: SEN0169

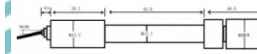
Contents

- 1 Introduction
- 2 Specification
- 3 Precautions
- 4 pH Electrode Characteristics
- 5 Usage
 - 5.1 Connecting Diagram
 - 5.2 Method 1. Software Calibration
 - 5.3 Method 2. Hardware Calibration through potentiometer
- 6 FAQ

Introduction

Need to measure water quality and other parameters but haven't got any low cost pH meter? Find it difficult to use with Arduino? Here comes an analog pH meter, specially designed for Arduino controllers and has built-in simple, convenient and practical connection and features. It has an LED which works as the Power Indicator, a BNC connector and PH2.0 sensor interface. You can just connect the pH sensor with BNC connector, and plug the PH2.0 interface into any analog input on Arduino controller to read pH value easily.

Specification



SEN0161 dimension

Module Power: 5.00V
 Circuit Board Size: 43mm*32mm
 pH Measuring Range: 0-14
 Measuring Temperature: 0-60 °C
 Accuracy: ± 0.1pH (25 °C)
 Response Time: ≤ 1min
 pH Sensor with BNC Connector
 PH2.0 interface (3 foot patch)
 Gain Adjustment Potentiometer
 Power Indicator LED

Precautions

Before and after use of the pH electrode every time, you need to use (pure)water to clean it.

The electrode plug should be kept clean and dry in case of short circuit.

Preservation. Electrode reference preservation solution is the 3N KCL solution.

Measurement should be avoided staggered pollution between solutions, so as not to affect the accuracy of measurement.

Electrode blub or sand core is defiled which will make PTS decline, slow response. So, it should be based on the characteristics of the pollutant, adapted to the cleaning solution, the electrode performance recovery.

Electrode when in use, the ceramic sand core and liquid outlet rubber ring should be removed, in order to make salt bridge solution to maintain a certain velocity.

NOTE: Differences between the probes, SEN0161 and SEN0169

Their usages/ specifications are almost the same. The differences locates at

Long-firing Operation: SEN0169 supports, while SEN0161 NOT, i.e. you can not immerse SEN0161 in water for Continuous Testing.

Life Span: In 25 °C, pure water, do Continuous Testing with them both, SEN0169 can work two years, while SEN0161 can only last for 6 months. And just for reference, if put them in turbid, strongly acid and alkali solution, 25°C, the life span would drop to one year (SEN0169), 1 month(or shorter, SEN0161).
 Temperature, pH, turbidity of the water effect the probe life span a lot.

Waterproof: You can immerse the whole probe SEN0169 into the water, while you can only immerse the front part of the probe SEN0161, the electrode glass bulb, into water, the rear part, from the white shell to the cable, MUST NOT be under water.

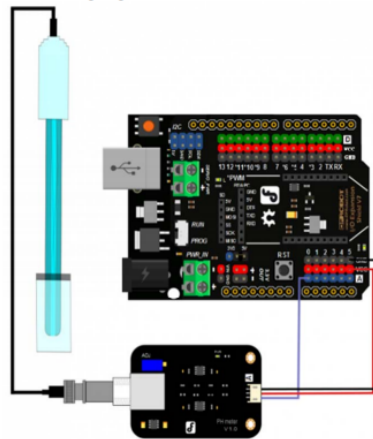
Strongly Acid and Alkali: SEN0169 are preferred for strongly acid and alkali test. And if your testing range is usually within pH6-8, then SEN0161 is capable for that.

pH Electrode Characteristics

The output of pH electrode is Millivolts, and the pH value of the relationship is shown as follows (25 °C):

VOLTAGE (mV)	pH value	VOLTAGE (mV)	pH value
414.12	0.00	-414.12	14.00
354.96	1.00	-354.96	13.00
295.80	2.00	-295.80	12.00
236.64	3.00	-236.64	11.00
177.48	4.00	-177.48	10.00
118.32	5.00	-118.32	9.00
59.16	6.00	-59.16	8.00
0.00	7.00	0.00	7.00

Usage Connecting Diagram



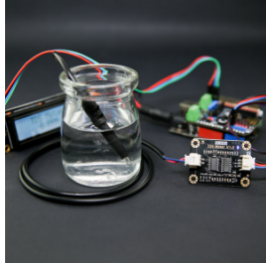
NOTE:

NOTE: It is normal that if your reading is much different with the table since you are not reading from the electrode directly but from the voltage adapter, it has converted the original voltage (-5V ~ +5V) to Arduino compatible voltage, i.e. 0 ~ 5V. See the discussion on Forum

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



Datasheet Sensor TDS SEN0244



Gravity: Analog TDS Sensor / Meter For Arduino
SKU: SEN0244

TDS (Total Dissolved Solids) indicates that how many milligrams of soluble solids dissolved in one liter of water. In general, the higher the TDS value, the more soluble solids dissolved in water, and the less clean the water is. Therefore, the TDS value can be used as one of the references for reflecting the cleanliness of water.

TDS pen is a widely used equipment to measure TDS value. The price is affordable, and it is easy to use, but it is not able to transmit data to the control system for online monitoring to do some water quality analysis. The professional instrument has high accuracy and can send data to the control system, but the price is expensive for the ordinary people. To this end, we have launched an analog TDS sensor kit which is compatible with Arduino, plug and play, easy to use. Matching with Arduino controller, you can build a TDS detector easily to measure the TDS value of liquid.

This product supports 3.3 ~ 5.5V wide voltage input, and 0 ~ 2.3V analog voltage output, which makes it compatible with 5V or 3.3V control system or board. The excitation source is AC signal, which can effectively prevent the probe from polarization and prolong the life of the probe, meanwhile, increase the stability of the output signal. The TDS probe is waterproof, it can be immersed in water for long time measurement.

This product can be used in water quality application, such as domestic water, hydroponics. With this product, you can easily DIY a TDS detector to reflect the cleanliness of water to protect your health.



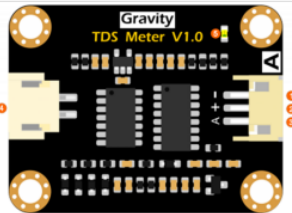
Attention:

1. The probe can not be used in water above 55 degrees centigrade.
2. The probe can not be left too close to the edge of the container, otherwise it will affect the reading.
3. The head and the cable of the probe are waterproof, but the connector and the signal transmitter board are not waterproof. Please be careful.

Specification

- **Signal Transmitter Board**
Input Voltage: 3.3 ~ 5.5V
Output Voltage: 0 ~ 2.3V
Working Current: 3 ~ 6mA
TDS Measurement Range: 0 ~ 1000ppm
TDS Measurement Accuracy: ± 10% F.S. (25 °C)
Module Size: 42 * 32mm
Module Interface: PH2.0-3P
Electrode Interface: XH2.54-2P
- **TDS probe**
Number of Needle: 2
Total Length: 83cm
Connection Interface: XH2.54-2P
Colour: Black
Other: Waterproof Probe

Board Overview



Analog TDS Sensor / Meter For Arduino

Num	Label	Description
1	-	Power GND(0V)
2	+	Power VCC(3.3 ~ 5.5V)
3	A	Analog Signal Output(0 ~ 2.3V)
4	TDS	TDS Probe Connector
5	LED	Power Indicator

Tutorial

This tutorial will show you how to measure the TDS value of the water. Please read this tutorial carefully, and pay attention to the steps and details.



The probe can not be used in water above 55 degrees centigrade. The probe can not be too close to the edge of the container, otherwise it will affect the reading. The head and the cable of the probe are waterproof, but the connector and the signal transmitter board are not waterproof. Please pay attention to use.

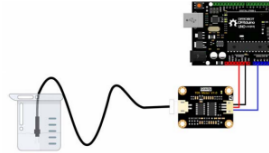
Requirements

- **Hardware**
DFRduino UNO R3 (or similar) x 1
Analog TDS Sensor / Meter Module x 1
TDS Probe x1
Jumper Wires x3
tested liquid x1
- **Software**

Arduino IDE (Version requirements: V1.0.x or V1.8.x). Click to Download Arduino IDE from Arduino®

<https://www.arduino.cc/en/Main/Software%7C70>

Connection Diagram



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



Datasheet Analog to Digital Converter ADS1115

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

DS18B20

Programmable Resolution
1-Wire Digital Thermometer

General Description

The DS18B20 digital thermometer provides 9-bit to 12-bit Celsius temperature measurements and has an alarm function with nonvolatile user-programmable upper and lower trigger points. The DS18B20 communicates over a 1-Wire bus that by definition requires only one data line (and ground) for communication with a central microprocessor. In addition, the DS18B20 can derive power directly from the data line ("parasite power"), eliminating the need for an external power supply. Each DS18B20 has a unique 64-bit serial code, which allows multiple DS18B20s to function on the same 1-Wire bus. Thus, it is simple to use one microprocessor to control many DS18B20s distributed over a large area. Applications that can benefit from this feature include HVAC environmental controls, temperature monitoring systems inside buildings, equipment, or machinery, and process monitoring and control systems.

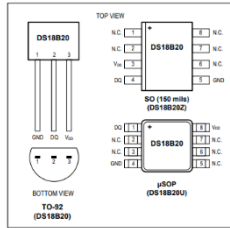
Applications

- Thermistic Controls
- Industrial Systems
- Consumer Products
- Thermometers
- Thermally Sensitive Systems

Benefits and Features

- Unique 1-Wire® Interface Requires Only One Port Pin for Communication
- Reduce Component Count with Integrated Temperature Sensor and EEPROM
 - Measures Temperatures from -55°C to +125°C (-67°F to +257°F)
 - ±0.5°C Accuracy from -10°C to +85°C
 - Programmable Resolution from 9 Bits to 12 Bits
 - No External Components Required
- Parasitic Power Mode Requires Only 2 Pins for Operation (DQ and GND)
- Simplifies Distributed Temperature-Sensing Applications with Multidrop Capability
 - Each Device Has a Unique 64-Bit Serial Code Stored in On-Board ROM
- Flexible User-Definable Nonvolatile (NV) Alarm Settings with Alarm Search Command Identifies Devices with Temperatures Outside Programmed Limits
- Available in 8-Pin SO (150 mils), 8-Pin µSOP, and 3-Pin TO-92 Packages

Pin Configurations



Ordering Information appears at end of data sheet.

1-Wire is a registered trademark of Maxim Integrated Products, Inc.

Absolute Maximum Ratings

Voltage Range on Any Pin Relative to Ground.....-0.5V to +6.0V
Storage Temperature Range.....-55°C to +125°C
Operating Temperature Range.....-55°C to +125°C
Solder Temperature.....Refer to the IPC/JEDEC J-STD-020 Specification.

DC Electrical Characteristics

(-55°C to +125°C; V_{DD} = 3.0V to 5.5V)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Supply Voltage	V _{DD}	Local power (Note 1)	+3.0	+5.5		V
		Parasite power	+3.0	+5.5		V
Pullup Supply Voltage	V _{PU}	Local power (Notes 1, 2)	+3.0	V _{DD}		V
		Thermometer Error	-10°C to +85°C		±0.5	
Thermometer Error	t _{ERR}	(Note 3)	-30°C to +100°C		±1	
			-55°C to +125°C		±2	
			Input Logic-Low	V _{IL}	(Notes 1, 4, 5)	-0.3
Input Logic-High	V _{IH}	Local power	+2.2	The lower of 5.5 or V _{DD} + 0.3		V
		Parasite power	+3.0			
Sink Current	I _L	V _{IO} = 0.4V	4.0			mA
Standby Current	I _{DDS}	(Notes 7, 8)		750	1000	nA
Active Current	I _{DD}	V _{DD} = 5V (Note 9)		1	1.5	mA
DQ Input Current	I _{DQ}	(Note 10)		5		µA
Drift		(Note 11)		±0.2		°C

- Notes:
Note 1: All voltages are referenced to ground.
Note 2: The Pullup Supply Voltage specification assumes that the pullup device is ideal, and therefore the high level of the pullup is equal to V_{PU}. In order to meet the V_{IH} spec of the DS18B20, the actual supply rail for the strong pullup transistor must include margin for the voltage drop across the transistor when it is turned on, thus: V_{PU_ACTUAL} = V_{PU_DEAL} + V_{TRANSISTOR}.
Note 3: See typical performance curve in Figure 1. Thermometer Error limits are 3-sigma values.
Note 4: Logic-low voltages are specified at a sink current of 4mA.
Note 5: To guarantee a presence pulse under low voltage parasite power conditions, V_{IHMAX} may have to be reduced to as low as 0.5V.
Note 6: Logic-high voltages are specified at a source current of 1mA.
Note 7: Standby current specified up to +70°C. Standby current typically is 3µA at +125°C.
Note 8: To minimize I_{DD}, DQ should be within the following ranges: GND ≤ DQ ≤ GND + 0.3V or V_{DD} - 0.3V ≤ DQ ≤ V_{DD}.
Note 9: Active current refers to supply current during active temperature conversions or EEPROM writes.
Note 10: DQ line is high ("high-Z" state).
Note 11: Drift data is based on a 1000-hour stress test at +125°C with V_{DD} = 5.5V.

DS18B20

Programmable Resolution
1-Wire Digital Thermometer

AC Electrical Characteristics—NV Memory

(-55°C to +125°C; V_{DD} = 3.0V to 5.5V)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
NV Write Cycle Time	t _{WR}			2	10	ms
EEPROM Writes	N _{EEWR}	-55°C to +55°C	50k			writes
EEPROM Data Retention	t _{EEDR}	-55°C to +55°C	10			years

AC Electrical Characteristics

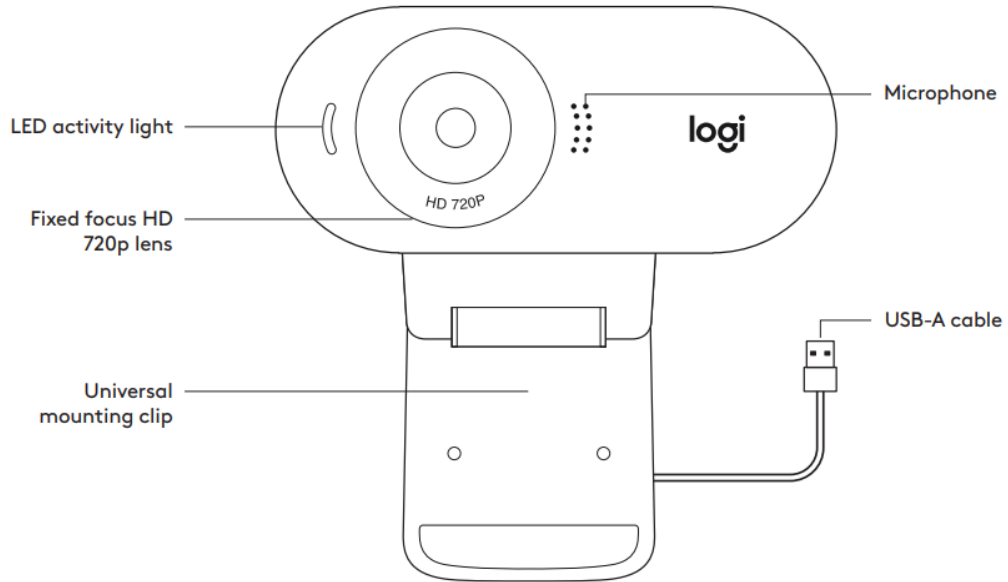
(-55°C to +125°C; V_{DD} = 3.0V to 5.5V)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Temperature Conversion Time	t _{CONV}	9-bit resolution	(Note 12)		93.75	ms
		10-bit resolution		187.5		
		11-bit resolution		375		
		12-bit resolution		750		
Time to Strong Pullup On	t _{SPON}	Start convert T command issued			10	µs
Time Slot	t _{SLOT}	(Note 12)	60		120	µs
Recovery Time	t _{REC}	(Note 12)	1			µs
Write 0 Low Time	t _{LOW0}	(Note 12)	60		120	µs
Write 1 Low Time	t _{LOW1}	(Note 12)	1		15	µs
Read Data Valid	t _{RDV}	(Note 12)			15	µs
Reset Time High	t _{RSTH}	(Note 12)	480			µs
Reset Time Low	t _{RSTL}	(Notes 12, 13)	480			µs
Presence-Detect High	t _{PDHIGH}	(Note 12)	15		60	µs
Presence-Detect Low	t _{PDLOW}	(Note 12)	60		240	µs
Capacitance	C _{IN/OUT}				25	pF

Note 12: See the timing diagrams in Figure 2.

Note 13: Under parasite power, if t_{RSTL} > 960µs, a power-on reset can occur.

Datasheet WebCam Logitech C310



DIMENSIONS

INCLUDING FIXED MOUNTING CLIP:

Height x Width x Depth:
2.80 in (71.15 mm) x 1.23 in (31.3 mm) x 1.02 in (25.95 mm)

Cable Length: 5 ft (1.5 m)

Weight: 2.51 oz (71.2 g), including clip and cable

- 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 satu 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



Datasheet Relay 4 Channel



User Guide

4 Channel 5V Optical Isolated Relay Module

This is a LOW Level 5V 4-channel relay interface board, and each channel needs a 15-20mA driver current. It can be used to control various appliances and equipment with large current. It is equipped with high-current relays that work under AC250V 10A or DC30V 10A. It has a standard interface that can be controlled directly by microcontroller. This module is optically isolated from high voltage side for safety requirement and also prevent ground loop when interface to microcontroller.



Brief Data:

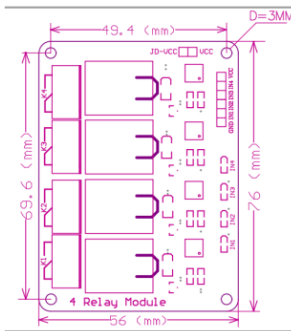
- Relay Maximum output: DC 30V/10A, AC 250V/10A.
- 4 Channel Relay Module with Opto-coupler. LOW Level Trigger expansion board, which is compatible with Arduino control board.
- Standard interface that can be controlled directly by microcontroller (8051, AVR, *PIC, DSP, ARM, ARM, MSP430, TTL logic).
- Relay of high quality low noise relays SPDT. A common terminal, a normally open, one normally closed terminal.
- Opto-Coupler isolation, for high voltage safety and prevent ground loop with microcontroller.

It is sometimes possible to use this relay boards with 3.3V signals, if the JD-VCC (Relay Power) is provided from a +5V supply and the VCC to JD-VCC jumper is removed. That 5V relay supply could be totally isolated from the 3.3V device, or have a common ground if opto-isolation is not needed. If used with isolated 3.3V signals, VCC (To the input of the opto-isolator, next to the IN pins) should be connected to the 3.3V device's +3.3V supply.

NOTE: Some Raspberry-Pi users have found that some relays are reliable and others do not actuate sometimes. It may be necessary to change the value of R1 from 1000 ohms to something like 220 ohms, or supply +5V to the VCC connection.

NOTE: The digital inputs from Arduino are Active LOW: The relay actuates and LED lights when the input pin is LOW, and turns off on HIGH.

Module Layout:



Operating Principle:

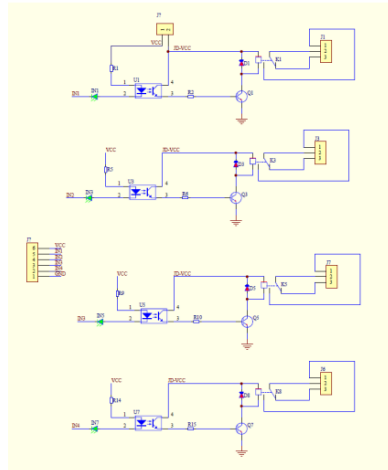
See the picture below: A is an electromagnet, B armature, C spring, D moving contact, and E fixed contacts. There are two fixed contacts, a normally closed one and a normally open one. When the coil is not energized, the normally open contact is the one that is off, while the normally closed one is the other that is on.

Schematic:

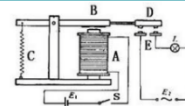
VCC and RY-VCC are also the power supply of the relay module. When you need to drive a large power load, you can take the jumper cap off and connect an extra power to RY-VCC to supply the relay; connect VCC to 5V of the MCU board to supply input signals.

NOTES: If you want complete optical isolation, connect "Vcc" to Arduino +5 volts but do NOT connect Arduino Ground. Remove the Vcc to JD-Vcc jumper. Connect a separate +5 supply to "JD-Vcc" and board Gnd. This will supply power to the transistor drivers and relay coils.

If relay isolation is enough for your application, connect Arduino +5 and Gnd, and leave Vcc to JD-Vcc jumper in place.



4 Channel Relay Module Schematic



Supply voltage to the coil and some currents will pass through the coil thus generating the electromagnetic effect. So the armature overcomes the tension of the spring and is attracted to the core, thus closing the moving contact of the armature and the normally open (NO) contact or you may say releasing the former and the normally closed (NC) contact. After the coil is de-energized, the electromagnetic force disappears and the armature moves back to the original position, releasing the moving contact and normally closed contact. The closing and releasing of the contacts results in power on and off of the circuit.

Input:

- VCC : Connected to positive supply voltage (supply power according to relay voltage)
- GND : Connected to supply ground.
- IN1: Signal triggering terminal 1 of relay module
- IN2: Signal triggering terminal 2 of relay module
- IN3: Signal triggering terminal 3 of relay module
- IN4: Signal triggering terminal 4 of relay module

Output:

Each module of the relay has one NC (normally close), one NO (normally open) and one COM (Common) terminal. So there are 4 NC, 4 NO and 4 COM of the channel relay in total. NC stands for the normal close port contact and the state without power. NO stands for the normal open port contact and the state with power. COM means the common port. You can choose NC port or NO port according to whether power or not.

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



Datasheet R385 DC Pump (Pompa DC)

DC Diaphragm Water Pump Micro Motor Model: R385

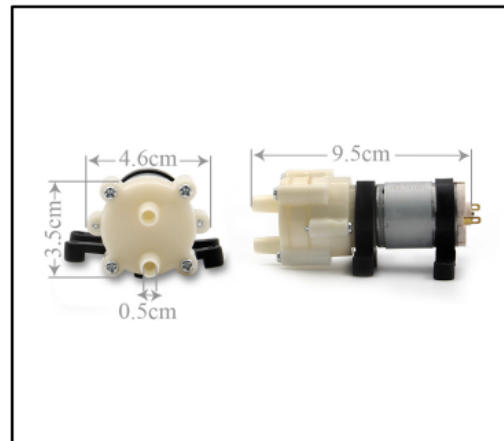


Description:

This is a R385 DC diaphragm pump. It can be used in all kinds of watering devices and fish tank pumps. Large flow, low noise.

Specifications:

- Voltage: 6-12v DC
- Current: 0.25A
- Power: 3W
- Flow rate: 1.8L ±0.1L/min
- Outlet diameter: Inner 6mm, Outer 9mm
- Head: up to 5M
- Continuous working time: 120Hrs
- Life span: up to 2500Hrs
- Working water pressure: 0.3Mpa
- Applicable water temperature: +5°C~+45°C



- 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 milik Politeknik Negeri Jakarta

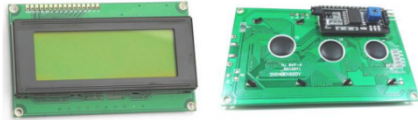
Datasheet LCD 2004 I2C



User Guide

I2C Serial Interface 20x4 LCD Module

This is I2C interface 20x4 LCD display module, a new high-quality 4 line 20 character LCD module with on-board contrast control adjustment, backlight and I2C communication interface. For Arduino beginners, no more cumbersome and complex LCD driver circuit connection. The real significance advantages of this I2C Serial LCD module will simplify the circuit connection, save some I/O pins on Arduino board, simplified firmware development with widely available Arduino library.



SKU: DSP-1165

Brief Data:

- Compatible with Arduino Board or other controller board with I2C bus.
- Display Type: Black on yellow green backlight.
- I2C Address: 0x38-0x3F (0x3F default)
- Supply voltage: 5V
- Interface: I2C to 4bits LCD data and control lines.
- Contrast Adjustment : built-in Potentiometer.
- Backlight Control : Firmware or jumper wire.
- Board Size: 98x60 mm.

Hitachi's HD44780 based character LCD are very cheap and widely available, and is an essential part for any project that displays information. Using the LCD piggy-back board, desired data can be displayed on the LCD through the I2C bus. In principle, such backpacks are built around PCF8574 (from NXP) which is a general purpose bidirectional 8 bit I/O port expander that uses the I2C protocol. The PCF8574 is a silicon CMOS circuit provides general purpose remote I/O expansion (an 8-bit quasi-bidirectional) for most microcontroller families via the two-line bidirectional bus (I2C-bus). Note that most piggy-back modules are centered around PCF8574T (SO16 package of PCF8574 in DIP16 package) with a default slave address of 0x27. If your piggy-back board holds a PCF8574AT chip, then the default slave address will change to 0x3F. In short, if the piggy-back board is based on PCF8574T and the address connections (A0-A1-A2) are not bridged with solder it will have the slave address 0x27.



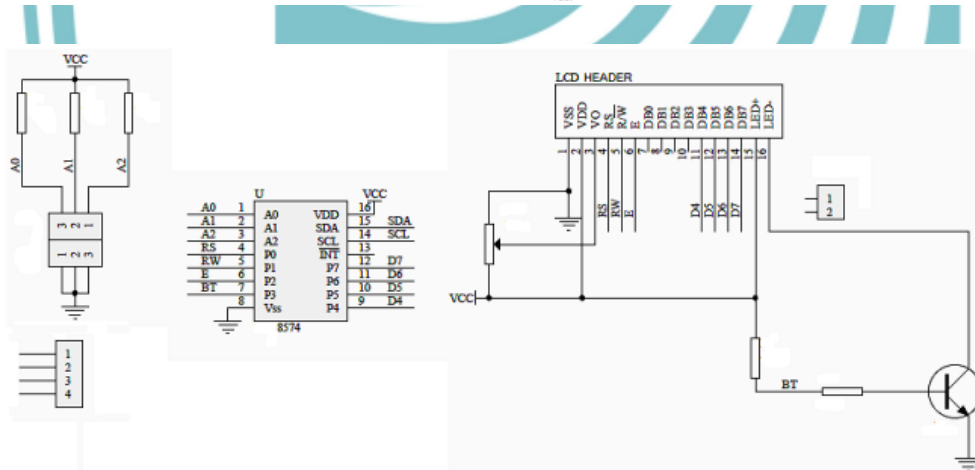
Address selection pads in the I2C-to-LCD piggy-back board.

Table 5. PCF8574A address map

Pin connectivity			Address of PCF8574A				Address byte value		7-bit hexadecimal address without RW				
A2	A1	A0	A5	A4	A3	A2	A1	A0		R/W	Write	Read	
Vss	Vss	Vss	0	1	1	1	0	0	0	-	70h	71h	38h
Vss	Vss	Vcc	0	1	1	1	0	0	1	-	72h	73h	39h
Vss	Vcc	Vss	0	1	1	1	0	1	0	-	74h	75h	3Ah
Vss	Vcc	Vcc	0	1	1	1	0	1	1	-	76h	77h	3Bh
Vcc	Vss	Vss	0	1	1	1	1	0	0	-	78h	79h	3Ch
Vcc	Vss	Vcc	0	1	1	1	1	0	1	-	7Ah	7Bh	3Dh
Vcc	Vcc	Vss	0	1	1	1	1	1	0	-	7Ch	7Dh	3Eh
Vcc	Vcc	Vcc	0	1	1	1	1	1	1	-	7Eh	7Fh	3Fh

Address Setting of PCF8574A (extract from PCF8574A data sheet).

Note: When the pad A0-A2 is open, the pin is pull up to VDD. When the pin is solder shorted, it is pull down to VSS.



Reference circuit diagram of the I2C-to-LCD piggy-back board.

I2C LCD Display.

At first you need to solder the I2C-to-LCD piggy-back board to the 16-pins LCD module. Ensure that the I2C-to-LCD piggy-back board pins are straight and fit in the LCD module, then solder in the first pin while keeping the I2C-to-LCD piggy-back board in the same plane with the LCD module. Once you have finished the soldering work, get four jumper wires and connect the LCD module to your Arduino as per the instruction given below.



- 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



Program Thonny

Python IDE

Sensor Suhu

```
import os
import glob
import time
import pyrebase
import schedule

os.system('modprobe wl-gpio')
os.system('modprobe wl-therm')

device_folder = '/sys/bus/wl/devices/'
device_file = glob.glob(device_folder + '28*')[0] + '/wl_slave'

config = {
    "apiKey": "rQNQCf00uqGHKnX8yBXcYvxsxGsHKnSrGwnShl4C",
    "authDomain": "monitoring-hydroponics.appspot.com",
    "databaseURL": "https://monitoring-hydroponics-default-rtdb.firebaseio.com/",
    "storageBucket": "YOUR_STORAGE_BUCKET"
}

firebase = pyrebase.initialize_app(config)
db = firebase.database()

def read_temperature():
    with open(device_file, 'r') as f:
        lines = f.readlines()

    while lines[0].strip()[-3:] != 'YES':
        time.sleep(0.2)
        lines = read_temperature()

    equals_pos = lines[1].find('t=')
    if equals_pos != -1:
        temp_string = lines[1][equals_pos + 2:]
        temperature = float(temp_string) / 1000.0
```

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 mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
        temperature += 0.2
        return temperature

def send_data_to_history():
    temperature = read_temperature()
    print('Suhu Air (History): %.2f °C' % temperature)

    if temperature >= 26 and temperature <= 32:
        status = "Keadaan normal"
    elif temperature < 26:
        status = "Terlalu rendah"
    else:
        status = "Terlalu tinggi"

    current_time_unix = int(time.time())

    data = {
        "unixTime": current_time_unix,
        "value": temperature,
        "status": status,
    }

    db.child("history").child("waterTemp").child(str(current_time_unix)
    ).update(data)

def send_data_to_transaction():
    temperature = read_temperature()
    print('Suhu Air (Transaction): %.2f °C' % temperature)

    db.child("transaction").child("waterTemp").set(temperature)

schedule.every(1).minutes.do(send_data_to_history)

while True:
    send_data_to_transaction()
    time.sleep(1)

    schedule.run_pending()
```

Sensor TDS



© 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

```
import time
import Adafruit_ADS1x15
import pyrebase
import schedule
import RPi.GPIO as GPIO

GPIO.setwarnings(False)

adc = Adafruit_ADS1x15.ADS1115()

GAIN = 1

tds_factor = 0.0645

relay_1_pin = 20
relay_2_pin = 21

GPIO.setmode(GPIO.BCM)

GPIO.setup(relay_1_pin, GPIO.OUT)
GPIO.setup(relay_2_pin, GPIO.OUT)

def read_tds():
    value = adc.read_adc(0, gain=GAIN)
    tds_value = value * tds_factor
    return tds_value

def control_relay_and_pump():
    tds_value = read_tds()

    if tds_value >1000:
        GPIO.output(relay_1_pin, GPIO.HIGH)
        GPIO.output(relay_2_pin, GPIO.HIGH)
        print("Relay and Pumps OFF (TDS: {:.2f}
ppm)".format(tds_value))
    else:

        GPIO.output(relay_1_pin, GPIO.LOW)
```



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

```
GPIO.output(relay_2_pin, GPIO.LOW)
print("Relay and Pumps ON (TDS: {:.2f}
ppm)".format(tds_value))

config = {
    "apiKey": "rQNQCf00uqGHKnX8yBXcYvxSxGsHKnSrGwnShl4C",
    "authDomain": "monitoring-hydroponics.appspot.com",
    "databaseURL": "https://monitoring-hydroponics-default-
rtdb.firebaseio.com/",
    "storageBucket": "YOUR_STORAGE_BUCKET"
}

firebase = pyrebase.initialize_app(config)
db = firebase.database()

def send_data_to_history():
    tds_value = read_tds()

    if tds_value >= 1000 and tds_value <= 1400:
        status = "Normal"
        status_msg = "Kondisi normal."
    elif tds_value < 1000:
        status = "Kurang Nutrisi"
        status_msg = "Kondisi Kurang nutrisi."
    else:
        status = "Tinggi Nutrisi"
        status_msg = "Kondisi nutrisi tinggi."

    print("status:", status)
    print(status_msg)

current_time_unix = int(time.time())
data = {
    "unixTime": current_time_unix,
    "value": tds_value,
    "status": status,
}
```




© 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

```
db.child("history").child("plantNutrition").child(str(current_time_unix)).update(data)
```

```
def send_data_to_transaction():  
    tds_value = read_tds()  
    print('Nilai TDS (Transaction): ', tds_value)  
    db.child("transaction").child("plantNutrition").set(tds_value)
```

```
control_relay_and_pump()
```

```
schedule.every(1).minutes.do(send_data_to_history)
```

```
while True:  
    send_data_to_transaction()  
    time.sleep(1)  
  
    schedule.run_pending()
```

Sensor pH

```
import time  
import Adafruit_ADS1x15  
import pyrebase  
import schedule  
import RPi.GPIO as GPIO
```

```
GPIO.setwarnings(False)
```

```
adc = Adafruit_ADS1x15.ADS1115()
```

```
GAIN = 1
```

```
ADC_CHANNEL_PH = 1
```

```
MIN_PH = 0.0
```

```
MAX_PH = 14.0
```

```
relay_3_pin = 23
```



© 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

```
relay_4_pin = 24

GPIO.setmode(GPIO.BCM)

GPIO.setup(relay_3_pin, GPIO.OUT)
GPIO.setup(relay_4_pin, GPIO.OUT)

config = {
    "apiKey": "rQNQCfOOuqGHKnX8yBXcYvxsxGsHKnSrGwnShl4C",
    "authDomain": "monitoring-hydroponics.appspot.com",
    "databaseURL": "https://monitoring-hydroponics-default-
rtdb.firebaseio.com/",
    "storageBucket": "YOUR_STORAGE_BUCKET"
}

firebase = pyrebase.initialize_app(config)

db = firebase.database()

def read_ph():
    raw_value = adc.read_adc(ADC_CHANNEL_PH, gain=GAIN)

    ph_value = map_range(raw_value, 0, 29757, MIN_PH, MAX_PH)

    return ph_value

def map_range(value, in_min, in_max, out_min, out_max):
    return (value - in_min) * (out_max - out_min) / (in_max -
in_min) + out_min

def control_relay_and_pump():
    ph_value = read_ph()

    if ph_value < 6.5:
        GPIO.output(relay_3_pin, GPIO.LOW)
```



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

```
        print("Relay and Pumps 3 ON (pH:
{:.2f})".format(ph_value))

    elif ph_value > 7:

        GPIO.output(relay_4_pin, GPIO.LOW)

        print("Relay and Pumps 4 ON (pH:
{:.2f})".format(ph_value))

    else:

        GPIO.output(relay_3_pin, GPIO.HIGH)

        GPIO.output(relay_4_pin, GPIO.HIGH)

        print("Relay and Pumps 3 and 4 OFF (pH:
{:.2f})".format(ph_value))

def send_data_to_history():
    ph_value = read_ph()

    print('Nilai pH (History): ', ph_value)

    if ph_value >= 6.5 and ph_value <= 7:

        status = "Keadaan normal"

    elif ph_value < 6.5:

        status = "Terlalu asam"

    else:

        status = "Terlalu basa"

    current_time_unix = int(time.time())

    data = {

        "unixTime": current_time_unix,

        "value": ph_value,

        "status": status,

    }

db.child("history").child("phLevel").child(str(current_time_unix))
.update(data)

def send_data_to_transaction():

    ph_value = read_ph()
```




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

```
print('Nilai pH (Transaction): ', ph_value)

db.child("transaction").child("phLevel").set(ph_value)

control_relay_and_pump()

schedule.every(1).minutes.do(send_data_to_history)

while True:
    send_data_to_transaction()
    time.sleep(1)
    schedule.run_pending()

Modul LCD

import os
import glob
import time
from smbus import SMBus
from RPLCD.i2c import CharLCD
import Adafruit_ADS1x15

os.system('modprobe w1-gpio')
os.system('modprobe w1-therm')

device_folder = '/sys/bus/w1/devices/'
device_file = glob.glob(device_folder + '28*')[0] + '/w1_slave'

adc = Adafruit_ADS1x15.ADS1115()

GAIN = 1
ADC_CHANNEL_PH = 1
ADC_CHANNEL_TDS = 0

MIN_PH = 0.0
MAX_PH = 14.0

tds_factor = 0.0470
```



© 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

```
lcd = CharLCD(i2c_expander='PCF8574', address=0x27, port=1,
cols=20, rows=4)

def read_temperature():
    with open(device_file, 'r') as f:
        lines = f.readlines()
    while lines[0].strip()[-3:] != 'YES':
        time.sleep(0.2)
        lines = read_temperature()
    equals_pos = lines[1].find('t=')
    if equals_pos != -1:
        temp_string = lines[1][equals_pos + 2:]
        temperature = float(temp_string) / 1000.0
    return temperature

def read_ph():
    raw_value = adc.read_adc(ADC_CHANNEL_PH, gain=GAIN)
    ph_value = map_range(raw_value, 0, 32767, MIN_PH, MAX_PH)
    return ph_value

def read_tds():
    tds_value = adc.read_adc(ADC_CHANNEL_TDS, gain=GAIN) *
tds_factor
    return tds_value

def map_range(value, in_min, in_max, out_min, out_max):
    return (value - in_min) * (out_max - out_min) / (in_max -
in_min) + out_min

lcd.clear()
lcd.write_string(' ' * 1 + 'Sistem' + ' ' * 2 + 'Pemantauan' + ' '
* 1)

lcd.write_string(' ' * 5 + 'Pertumbuhan' + ' ' * 3)
lcd.write_string(' ' * 2 + 'Tanaman' + ' ' * 2 + 'Hidroponik')
lcd.write_string(' ' * 3 + 'Berbasis' + ' ' * 2 + 'IoT' + ' ' * 1)
time.sleep(5)

while True:
    temp = read_temperature()
```



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

```
ph_value = read_ph()
tds = read_tds()

lcd.clear()
lcd.write_string('Deteksi Nilai Sensor')
lcd.crlf()
lcd.write_string('Suhu : %.2f °C' % temp)
lcd.crlf()
lcd.write_string('TDS : %.2f ppm' % tds)
lcd.crlf()
lcd.write_string('pH : %.2f' % ph_value)
lcd.crlf()
time.sleep(1)
```

Sensor Color Image Processing

```
import cv2
import numpy as np
import time
import pyrebase

config = {
    "apiKey": "rQNQCf00uqGHKnx8yBXcYvxsxGsHKnSrGwnShl4C",
    "authDomain": "monitoring-hydroponics.appspot.com",
    "databaseURL": "https://monitoring-hydroponics-default-
rtdb.firebaseio.com/",
    "storageBucket": "monitoring-hydroponics.appspot.com"
}

firebase = pyrebase.initialize_app(config)
storage = firebase.storage()

def detect_plant_height(frame, distance_to_plant):
    hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)

    lower_green = np.array([30, 50, 50])
    upper_green = np.array([90, 255, 255])

    mask = cv2.inRange(hsv, lower_green, upper_green)
```




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

```
kernel = np.ones((5, 5), np.uint8)
opening = cv2.morphologyEx(mask, cv2.MORPH_OPEN, kernel)
closing = cv2.morphologyEx(opening, cv2.MORPH_CLOSE, kernel)

contours, _ = cv2.findContours(closing, cv2.RETR_EXTERNAL,
cv2.CHAIN_APPROX_SIMPLE)

plant_heights = []

for contour in contours:
    if cv2.contourArea(contour) > 200:
        x, y, w, h = cv2.boundingRect(contour)
        cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 255,
0), 2)

        plant_height = 0.6 * h * distance_to_plant /
frame.shape[0]
        plant_heights.append(plant_height)

        text = "{:.2f} cm".format(plant_height)
        text_width, text_height = cv2.getTextSize(text,
cv2.FONT_HERSHEY_SIMPLEX, 0.7, 2)[0]
        cv2.putText(frame, text, (x + int(w/2) -
int(text_width/2), y - 10), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0,
255, 0), 2, cv2.LINE_AA)

    return plant_heights

distance_to_plant = 140

cap = cv2.VideoCapture(0)

last_height_display_time = time.time()

last_capture_time = time.time()

capture_interval = 30

while True:
```



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

```
ret, frame = cap.read()

plant_heights = detect_plant_height(frame, distance_to_plant)

cv2.imshow('Plant Height Detection', frame)

if plant_heights and time.time() - last_height_display_time >=
2:
    for i, plant_height in enumerate(plant_heights):
        print("Tinggi Tanaman", i+1, " (cm):", plant_height)
        last_height_display_time = time.time()

    if time.time() - last_capture_time >= capture_interval:
        image_path = "/home/pi/Pictures/"
        image_name =
f"gambar_tanaman_{time.strftime('%Y%m%d%H%M%S')}.jpg"
        image_file = image_path + image_name
        cv2.imwrite(image_file, frame)
        print("Mengambil gambar:", image_name)
        last_capture_time = time.time()

        storage.child("images/" + image_name).put(image_file)

        image_url = storage.child("images/" +
image_name).get_url(None)
        print("URL Gambar:", image_url)

        db = firebase.database()
        data = {
            "unixTime": int(time.time()),
            "value": image_url
        }

        db.child("transaction").child("img").set(image_url)

db.child("history").child("img").child(str(int(time.time()))).update(data)

if cv2.waitKey(1) & 0xFF == ord('q'):
```

```
break
```

```
cap.release()
```

```
cv2.destroyAllWindows()
```

Kode Program Gabungan

```
import multiprocessing  
import subprocess
```

```
def run_file(file_name):  
    subprocess.run(["python", file_name])
```

```
if __name__ == "__main__":  
    file_names = ["suhu.py", "tds.py", "ph.py", "camera.py",  
                "lcd20x4.py"]
```

```
    with multiprocessing.Pool() as pool:  
        pool.map(run_file, file_names)
```



POLITEKNIK
NEGERI
JAKARTA

© 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

