



BALAI BESAR PENGEMBANGAN
PENJAMINAN MUTU PENDIDIKAN VOKASI
BIDANG OTOMOTIF DAN ELEKTRONIKA

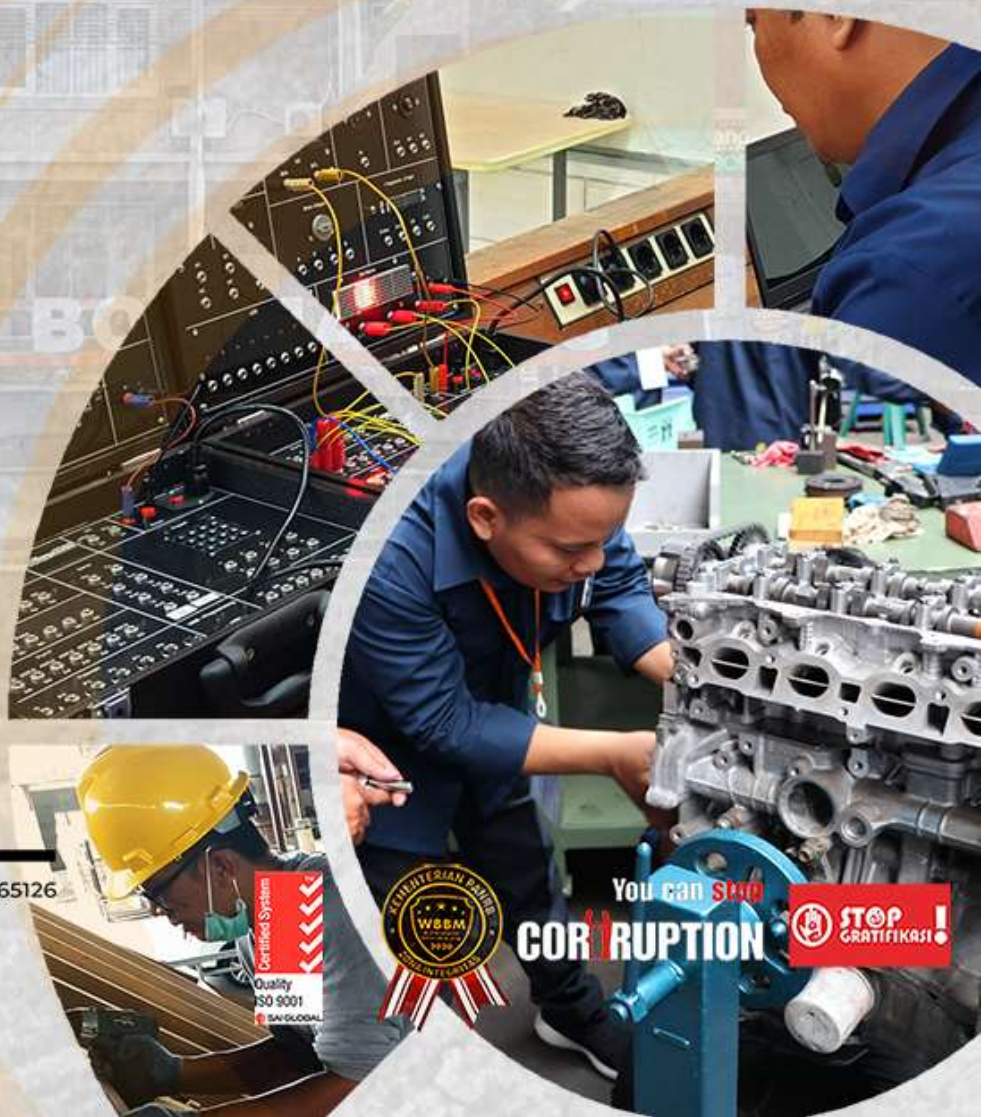
**MERDEKA
BELAJAR**

Pemrograman MicroPython Dalam Sistem Komputasi Physical

#1. Raspberry Pi Pico

Penyusun:

Dr. Eko Subiyantoro, S.Pd, S.ST, MT



Jl. Teluk Mandar, Arjosari Tromol Pos 5 Malang 65126
Telp. (0341) 491239, 495849 Fax. (0341) 491342
Instagram : bbppmpvboe.kemdikbud
Website : bbppmpvboe.kemdikbud.go.id/
Facebook : Bbppmpvboe.kemdikbud
Youtube : bbppmpv boe
Tiktok : bbppmpvboemlg



You can **STOP**
CORRUPTION



PRAKTIKUM 1

KOMPUTASI PHYSICAL-RAPSBERRY PI PICO

2.1. Tujuan Praktikum

Peserta pelatihan mampu:

1. Memahami perangkat rapsberry pico pi
2. Menerapkan perangkat elektronik dalam rangkaian
3. Melakukan instalasi micro python pada rapsberry pico pi
4. Melakukan kontrol perangkat elektronik dengan rapsberry pico pi menggunakan MicroPython
5. Membuat mini proyek menggunakan MicroPython dengan fungsi GPIO input dan output.

2.2. Pendahuluan

2.2.1 Rapsberry Pi Pico

Raspberry Pi adalah sebuah komputer papan tunggal (single-board computer) atau SBC seukuran kartu kredit yang dapat digunakan untuk menjalankan program perkantoran, permainan komputer, dan sebagai pemutar media hingga video beresolusi tinggi. Raspberry Pi dikembangkan oleh yayasan nirlaba, Rasberry Pi Foundation dengan tujuan untuk belajar pemrograman.



Gambar 2.1 Rapsberry Pi 4 Model B

Raspberry Pi pertama kali dikembangkan di laboratorium Komputer Universitas Cambridge oleh Eben Upton, Rob Mullins, Jack Lang, dan Alan Mycrof. Mereka kemudian mendirikan yayasan Raspberry Pi bersama dengan Pete Lomas dan David Braben pada tahun 2009. Pada tahun 2012, Raspberry Pi Model B memasuki produksi massal. Dalam peluncuran pertamanya pada akhir Febuari 2012 dalam beberapa jam saja sudah terjual 100.000 unit. Pada bulan Februari 2016, Raspberry Pi Foundation mengumumkan bahwa mereka telah menjual 8 juta perangkat Raspi, sehingga menjadikannya sebagai perangkat paling laris di Inggris.

Nama Raspberry Pi diambil dari nama buah, yaitu buah Raspberry, sedangkan Pi diambil dari kata Python, yaitu nama dari sebuah bahasa pemrograman. Python dijadikan bahasa pemrograman utama dari Raspberry Pi, namun tidak tertutup kemungkinan untuk menggunakan bahasa pemrograman lain pada Raspberry Pi. Keunggulan python dibanding dengan bahasa pemrograman yang lain adalah kode kode lebih mudah ditulis dan dibaca, dan juga banyak terdapat modul modul yang beragam. Adapun kekurangannya adalah tidak realtime, sehingga untuk akan kesusahan untuk melakukan pekerjaan yang mempunyai delay, akibatnya tingkat presisi juga tidak tinggi.

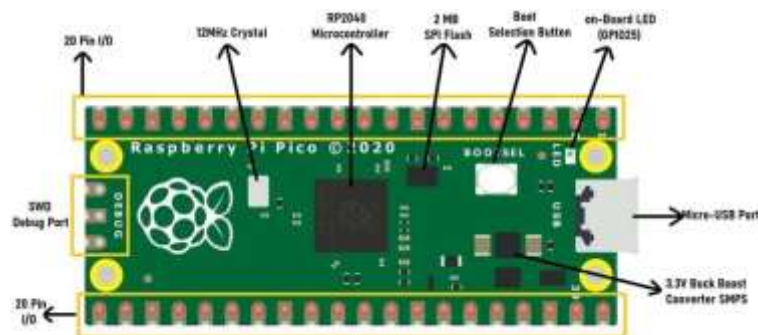
Selama bertahun-tahun Raspberry Pi telah menjadi alat wajib bagi siswa, penghobi, atau Industrialis. Namun dalam hal biaya, Papan Raspberry Pi dikalahkan oleh Arduino, ESP32, STM32, atau Mikrokontroler AVR, ARM, PIC lainnya. Komputer Raspberry Pi harganya sekitar \$35-40 sedangkan mikrokontroler lainnya hanya berharga \$2-5 saja. Ini adalah alasan mengapa Raspberry Pi Foundation merilis Raspberry Pi Pico Board berdaya rendah dan kompetitif dengan RP2040, Mikrokontroler Dual Core ARM Cortex-M0+.

Raspberry Pi Pico adalah papan mikrokontroler pertama menggunakan RP2040. Ini sangat mirip dengan papan mikrokontroler lainnya dengan MCU di tengah, konektor micro-USB di satu ujung, dan deretan kontak di setiap sisi. Konektor debug 3-pin tersedia di ujung lain papan.



Gambar 2.2 Rapsberry Pi Pico

Raspberry Pi Pico berukuran 51 kali 21 mm, yang ukurannya sama persis dengan Kit Pico ESP32 & sedikit lebih besar dari Arduino Nano atau Micro. Pico hadir dengan memori Flash QSPI 2 MB dan 25 dari 30 pin GPIO RP2040.

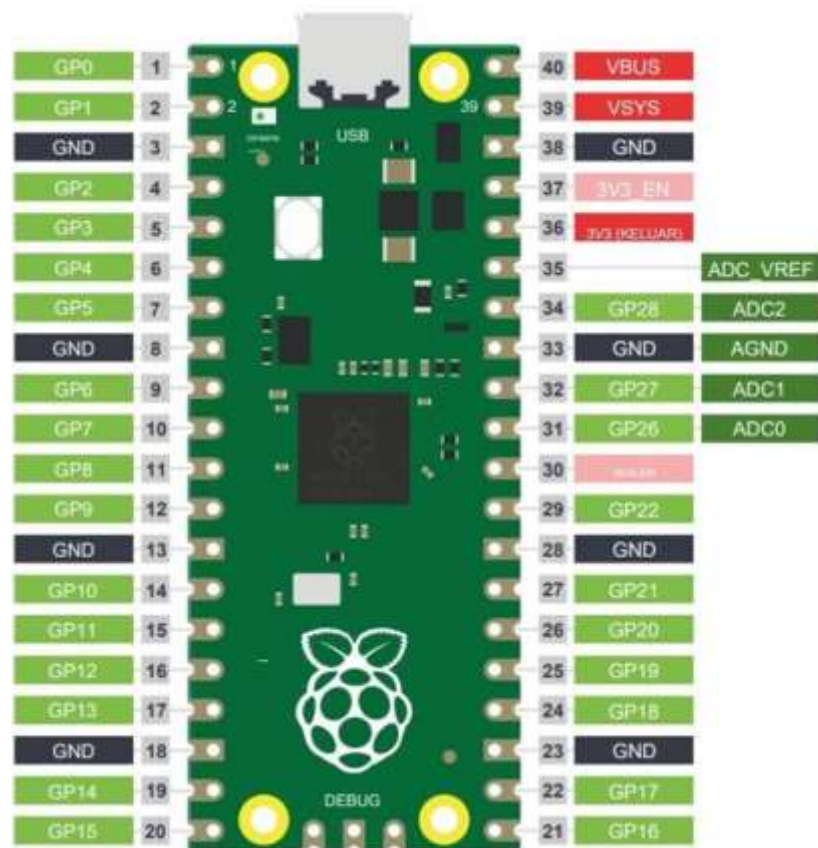


Gambar 2.2 Rapsberry Pi Pico

2.2.2 Rapsberry Pi Pico Pinout

Pico dapat berkomunikasi dengan perangkat keras melalui serangkaian pin di kedua sisinya. Sebagian besar pin tersebut berfungsi sebagai General Purpose Input Output (GPIO). Artinya, pin-pin tersebut dapat diprogram untuk berperan sebagai input atau output. Beberapa pin memiliki fitur tambahan dan mode alternatif untuk berinteraksi dengan perangkat keras yang lebih kompleks, sementara beberapa pin lainnya memiliki tujuan tetap, seperti menyediakan koneksi untuk daya.

Pada Raspberry Pi Pico dengan 40 pin, penomoran pin tertera di bagian bawah papan, dengan tiga pin di atasnya diberi label dengan nomor: Pin 1, Pin 2, dan Pin 39. Label ini membantu memahami sistem penomoran: Pin 1 berada di sudut kiri atas saat papan dilihat dari atas dengan port USB mikro di bagian atas. Pin 19 berada di sudut kiri bawah, Pin 20 berada di sudut kanan bawah, dan Pin 39 berada di sudut kanan atas, di bawahnya terdapat Pin 40 tanpa label.



Gambar 2.3 Pinout Rapsberry Pi Pico

Daripada menggunakan nomor pin fisik, lebih umum merujuk pada pin berdasarkan fungsi masing-masing dapat dilihat pada Gambar 2.4. Ada beberapa kategori jenis pin yang memiliki fungsi spesifik masing-masing:

3v3	listrik 3,3 volt	Sumber daya 3,3 V, voltase yang sama dengan Pico Anda secara internal, dihasilkan dari input VSYS. Catu daya ini dapat dinyalakan dan dimatikan menggunakan pin 3V3_EN di atasnya, yang juga mematikan Pico Anda.
VSYS	~ Daya 2-5 volt	Pin yang terhubung langsung ke catu daya internal Pico Anda, yang tidak dapat dimatikan tanpa juga mematikan Pico.
VBUS	listrik 5 volt	Sumber daya 5 V diambil dari port micro USB Pico Anda, dan digunakan untuk menyalakan perangkat keras yang membutuhkan lebih dari 3,3 V.
GND	tanah 0 volt	Koneksi ground, digunakan untuk melengkapi sirkuit yang terhubung ke sumber listrik. Beberapa pin ini bertitik di sekitar Pico Anda untuk mempermudah pengkabelan
GPxx	Nomor pin input/output tujuan umum 'xx'	Pin GPIO yang tersedia untuk program Anda, berlabel 'GP0' hingga 'GP28'.
GPxx_ADCx	nomor pin input/output umum 'xx', dengan nomor input analog 'x'	Pin GPIO yang diakhiri dengan 'ADC' dan angka dapat digunakan sebagai input analog juga input atau output digital – tetapi tidak keduanya pada saat itu waktu yang sama.
ADC_VREF	Konverter analog-ke-digital (ADC) tegangan referensi	Pin input khusus yang menetapkan <i>referensi tegangan</i> untuk setiap input analog.
AGND	Konverter analog-ke-digital (ADC) 0 volt tanah	Koneksi ground khusus untuk digunakan dengan pin ADC_VREF.
BERLARI	Mengaktifkan atau menonaktifkan Pico Anda	Header RUN digunakan untuk memulai dan menghentikan Anda Pico dari mikrokontroler lain.

Gambar 2.4 Fungsi Pinout Raspberry Pi Pico



PIN GPIO

Seperti menghitung dengan Python, pin GPIO Pico Anda dimulai dari angka 0 bukan angka 1. Diberi label di bagian bawah papan, mereka berubah dari 0 menjadi 29, meskipun beberapa diantaranya tidak dikelompokkan pin fisik.



PIN HILANG

Pin input/output tujuan umum pada Pico di beri nomor berdasarkan pin chip yang menggerakkannya, sebuah RP2040 mikrokontroler. Tidak semua pin yang tersedia di RP2040 dibawa keluar ke pin Pico Anda, namun - itulah sebabnya ada celah di dalamnya penomoran antara pin GP22 tujuan umum dasar terakhir dan pin berkemampuan analog pertama GP26_ADC0.



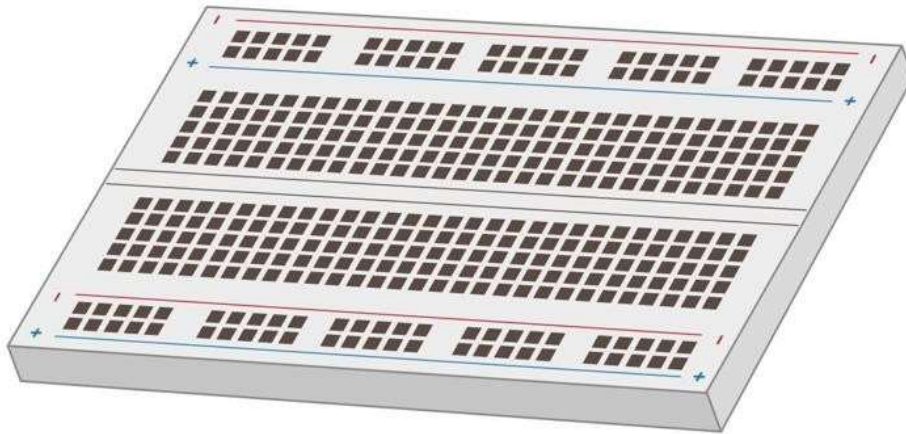
PERINGATAN !

Pin Pico Anda dirancang untuk menjadi cara yang menyenangkan dan aman bereksperimen dengan komputasi fisik, tetapi harus selalu dirawat dengan hati-hati. Berhati-hatilah untuk tidak membengkokkan pin, terutama saat Anda melakukannya. Memasukkan Pico Anda ke papan tempat memotong roti. Jangan pernah menghubungkan dua pin langsung bersama, sengaja atau tidak sengaja, kecuali disuruh lakukan dalam petunjuk proyek ini dikenal sebagai korsleting dan, bergantung pada pinnya, dapat merusak Pico Anda secara permanen.

2.2.3 Komponen Elektronik

Raspberry Pi Pico hanyalah salah satu bagian dari apa yang diperlukan untuk memulai komputasi fisik; separuh sisanya terdiri dari komponen listrik, yaitu perangkat yang akan dikendalikan melalui pin GPIO Pico. Terdapat ribuan komponen yang berbeda yang tersedia, namun sebagian besar proyek komputasi fisik menggunakan komponen umum berikut.

1. Breadboard Simulator



Gambar 2.5 Breadboard Simulator

Breadboard Simulator atau solderless breadboard, mempermudah proyek komputasi fisik dengan menghubungkan komponen melalui jalur logam tersembunyi di bawah permukaannya. Banyak papan ini juga dilengkapi dengan bagian distribusi daya, memudahkan pembuatan sirkuit. Meskipun tidak diperlukan, penggunaan papan tempat memotong roti membuat semuanya lebih sederhana dan stabil dengan menghubungkan komponen ke pin GPIO Pico menggunakan kabel khusus.

2. Kabel Jumper (LED Jumper)



Gambar 2.6 Kabel Jumper

Kabel jumper digunakan untuk menghubungkan komponen ke Pico, baik secara langsung maupun antar komponen jika tidak menggunakan Breadboard Simulator. Terdapat tiga jenis kabel jumper: male-to-female (M2F), female-to-female (F2F), dan male-to-male (M2M). Kabel M2F digunakan untuk menghubungkan komponen ke Pico jika tidak menggunakan breadboard. Kabel M2M digunakan untuk menghubungkan bagian-bagian pada papan tempat memotong roti. Pilihan kabel jumper yang digunakan tergantung pada proyek yang sedang dikerjakan. Jika menggunakan breadboard, umumnya hanya diperlukan kabel jumper M2F dan M2M.

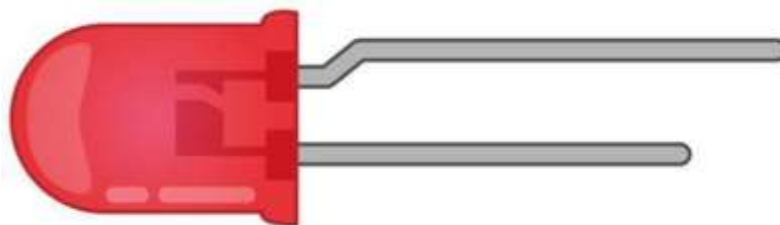
3. Push Button Switch (Sakelar Tombol Tekan)



Gambar 2.7 Kabel Jumper

Push Button Switch (Sakelar Tombol Tekan), juga dikenal sebagai sakelar momen, adalah jenis sakelar yang digunakan untuk mengendalikan konsol game. Biasanya memiliki dua atau empat kaki - salah satu jenis akan kompatibel dengan Pico. Tombol tekan merupakan perangkat input yang memungkinkan Anda memberitahu program untuk merespons saat ditekan dan menjalankan tugas tertentu. Jenis sakelar umum lainnya adalah sakelar pengunci; berbeda dengan tombol tekan yang hanya aktif saat ditekan dan dilepaskan, sakelar pengunci - seperti yang ditemui pada sakelar lampu - akan tetap aktif setelah diaktifkan dan tetap aktif hingga diaktifkan kembali.

4. Light Emitting Diode (LED)

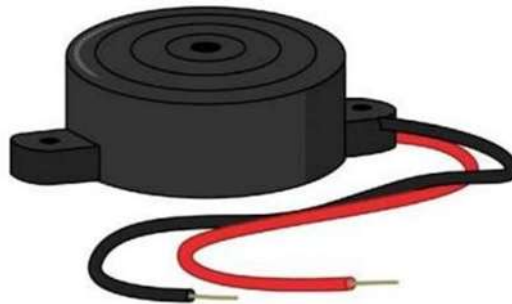


Gambar 2.8 Light Emitting Diode

Light-emitting diode (LED) / Dioda Pemancar Cahaya merupakan perangkat keluaran yang dikendalikan langsung dari program. LED akan menyala saat diaktifkan dan umum ditemukan di rumah, mulai dari yang kecil sebagai indikator mesin cuci hingga yang besar untuk pencahayaan

ruangan. LED hadir dalam berbagai bentuk, warna, dan ukuran, namun perlu diingat bahwa tidak semua cocok untuk digunakan dengan Pico, hindari LED yang dirancang untuk catu daya 5V atau 12V.

5. Piezoelectric Buzzer



Gambar 2.9 Piezoelectric Buzzer

Piezoelectric buzzer yang biasa disebut buzzer atau sounder, adalah perangkat keluaran yang menghasilkan suara. Berbeda dengan LED yang menghasilkan cahaya, buzzer menghasilkan suara berdengung. Di dalam rumah plastik buzzer terdapat sepasang pelat logam yang bergetar satu sama lain saat dinyalakan, menghasilkan suara. Terdapat dua jenis buzzer, yaitu buzzer aktif dan buzzer pasif. Pastikan untuk memilih buzzer aktif, karena ini paling sederhana untuk digunakan.

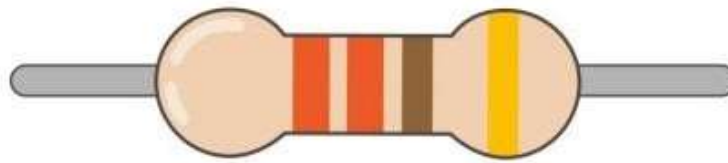
6. I2C display



Gambar 2.10 LCD I2C

Layar I2C adalah perangkat tampilan yang berkomunikasi dengan Pico melalui bus I2C. Ada berbagai jenis layar, termasuk SerLCD dari SparkFun dengan antarmuka I2C dan SPI. Beberapa layar menggunakan bus SPI dan perlu penyesuaian program. Komponen kelistrikan lainnya meliputi motor, sensor arus, unit pengukuran inersia (IMU), dan resistor bergantung cahaya (LDR) sebagai perangkat input.

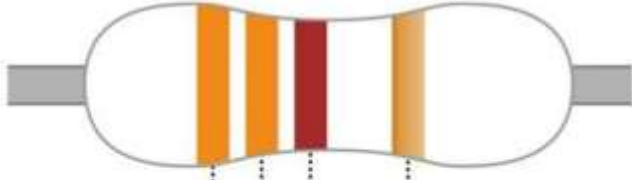
7. Resistor



Gambar 2.11 Resistor

Resistor adalah komponen yang mengatur aliran arus listrik dan memiliki nilai yang diukur dalam satuan ohm (Ω). Semakin tinggi nilai ohm, semakin besar resistansinya. Dalam proyek komputasi fisik dengan Pico, resistor umumnya digunakan untuk melindungi LED agar tidak mengalirkan terlalu banyak arus yang dapat merusak LED atau Pico Anda. Untuk itu, resistor dengan nilai sekitar 330Ω sangat direkomendasikan. Namun, tersedia juga paket resistor dengan nilai-nilai umum yang berbeda untuk memberikan fleksibilitas lebih kepada pengguna.

Resistor memiliki berbagai nilai, mulai dari yang resistansinya nol yang sebenarnya hanya potongan kawat, hingga yang seukuran kaki Anda yang digunakan di pembangkit listrik. Hanya sedikit resistor yang memiliki nilai yang tercetak dalam angka. Sebaliknya, resistor menggunakan kode khusus yang ditampilkan sebagai garis atau pita berwarna di sekitar badannya.



	Band 1/2	Pengali	Toleransi
Hitam	0	$\times 100$	-
Cokelat	1	$\times 10^1$	$\pm 1\%$
Merah	2	$\times 10^2$	$\pm 2\%$
Oranye	3	$\times 10^3$	-
Kuning	4	$\times 10^4$	-
Hijau	5	$\times 10^5$	$\pm 0,5\%$
Biru	6	$\times 10^6$	$\pm 0,25\%$
Ungu	7	$\times 10^7$	$\pm 0,1\%$
Abu-abu	8	$\times 10^8$	$\pm 0,05\%$
Putih	9	$\times 10^9$	-
Emas	-	$\times 10^{-1}$	$\pm 5\%$
Perak	-	$\times 10^{-2}$	$\pm 10\%$
Tidak ada	-	-	$\pm 20\%$

Gambar 2.11 Cara Membaca Nilai Resistor

2.3. Integrasi MicroPython Dengan Perangkat Elektronik

2.3.1 Instalasi MicroPython Pada Raspberry Pi Pico

MicroPython awalnya dikembangkan oleh Damien George dan dirilis pertama kali pada tahun 2014, adalah bahasa pemrograman yang kompatibel dengan Python yang dikhususkan untuk mikrokontroler. MicroPython mencakup banyak fitur Python utama dan menambahkan fitur baru yang dirancang khusus untuk memanfaatkan fasilitas yang ada di Raspberry Pi Pico dan papan mikrokontroler lainnya. Jika telah menguasai pemrograman dengan Python sebelumnya, akan merasa familiar dengan MicroPython.

Menginstal MicroPython ke dalamnya Pi Pico diawali dengan mencolokkan kabel micro USB ke port micro USB di Pico, pastikan itu benar sebelum mendorongnya pulang dengan lembut dan selanjutnya sambungan ke USB laptop atau komputer.



PERINGATAN

Untuk menginstal MicroPython ke Pico, harus mengunduhnya dari internet. Jika Anda tidak memiliki koneksi internet di ponsel atau Raspberry Pi, harus menghubungkan Pico ke sistem yang dapat melakukannya dan memiliki koneksi internet untuk menyelesaikan penyiapannya. Anda hanya akan melakukan ini sekali: setelah MicroPython diinstal, itu akan tetap ada di Pico kecuali Anda memutuskan untuk menggantinya dengan sesuatu yang lain di masa mendatang.

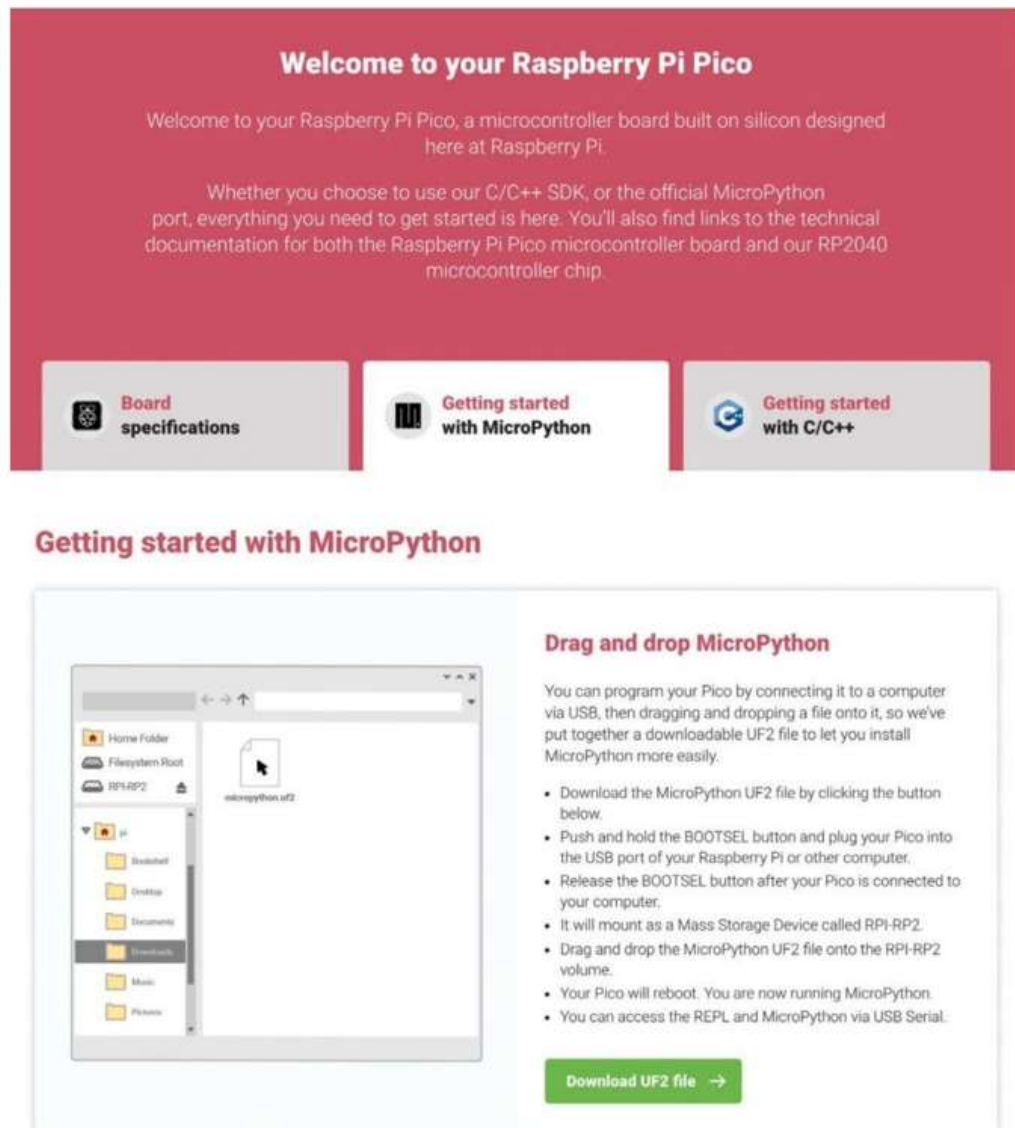
Tahan tombol 'BOOTSEL' di bagian atas Pico; kemudian sambil tetap menahannya, sambungkan ujung kabel micro USB yang lain ke salah satu port USB di Raspberry Pi atau komputer lain. Hitung sampai tiga, lalu lepaskan tombol 'BOOTSEL'.

Setelah beberapa detik, Anda akan melihat Pico Anda muncul sebagai drive yang dapat dilepas – seolah-olah Anda telah menghubungkan flash drive USB atau hard drive eksternal. Di Raspberry Pi atau laptop akan melihat pop-up menanyakan apakah ingin membuka drive di File Manager.

Di jendela File Manager, akan terdapat dua file di Pico yaitu dua file INDEX.HTM dan INFO_UF2.TXT. File kedua menyimpan informasi tentang Pico, seperti versi bootloader yang sedang berjalan. File pertama, INDEX.HTM, adalah file yang diinginkan: arahkan mouse kursor padanya dan klik dua kali untuk membukanya di browser.

Saat browser terbuka, akan terlihat halaman selamat datang yang memberi tahu tentang semua tentang Pico Anda. Baca informasi di halaman, lalu klik tab MicroPython untuk memuat file khusus MicroPython bagian halaman.

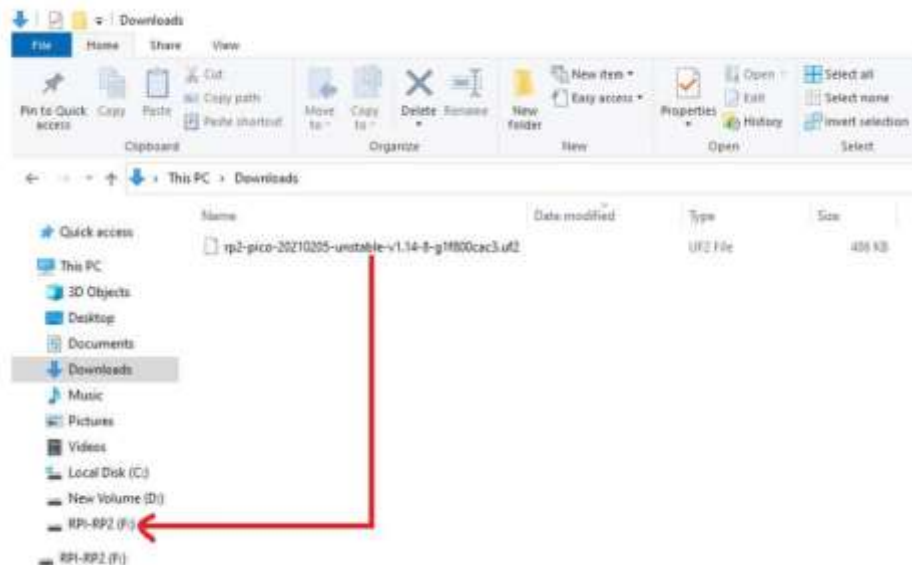
Klik tombol 'Unduh file UF2' disajikan pada Gambar 2.12 , untuk mengunduh MicroPython firmware – file kecil yang berisi MicroPython untuk Pico Pi.



Gambar 2.12 Site untuk mengunduh firmware Micropyhton

File ini sangat kecil, Setelah diunduh, tutup jendela browser dengan mengklik ikon silang di pojok kanan atas. Buka jendela File Manager baru dengan mengklik menu ikon raspberry, pergi ke Aksesoris, dan mengklik File Manager. Temukan folder unduhan dengan menggunakan daftar folder di sebelah kiri jendela File Manager. Buka folder unduhan dan temukan file yang baru saja yang diunduh itu akan disebut 'micropython' diikuti dengan tanggal dan ekstensi 'uf2'.

Klik dan tahan tombol mouse pada file UF2 lalu seret ke jendela File Manager lainnya yang membuka drive penyimpanan Pico yang dapat dilepas. Arahkan ke atas jendela dan lepaskan tombol mouse untuk menempatkan file ke Pico Gambar 2.13.



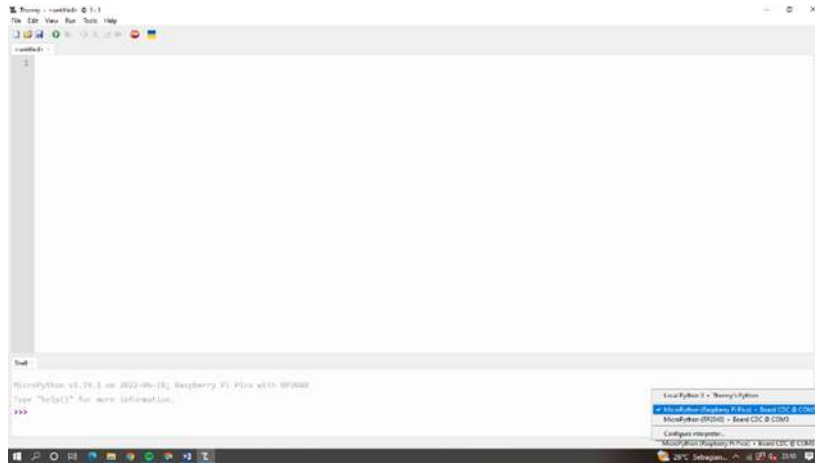
Gambar 2.13 Drag dan Drop firmware micropython ke Pi Pico

Setelah beberapa detik, file pada Pi Pico akan menghilang dari File Manager, dan mungkin juga melihat peringatan bahwa drive telah dihapus tanpa dikeluarkan. Pada saat menyeret file firmware MicroPython ke Pico, sebenarnya adalah proses untuk mem-flash firmware ke penyimpanan internalnya. Tombol 'BOOTSEL' digubakan untuk mem-flash firmware baru, dan kemudian memuatnya – artinya Pico Pi sekarang sudah bisa menjalankan MicroPython.

2.3.2 Menghubungkan IDE Thonny ke Rapsberry Pi Pico

Sebelum memulai pemrograman MicroPython dengan Pi Pico, langkah pertama adalah mengatur lingkungan pengembangan terintegrasi (IDE). Thonny adalah IDE populer untuk Python dan MicroPython yang telah diinstal sebelumnya di OS Raspberry Pi. Jika Anda menggunakan Pico dengan distribusi Linux yang berbeda, Windows, atau macOS, Anda perlu mengunduh dan menginstal Thonny dan Python untuk sistem operasi. Thonny adalah lingkungan pengembangan terintegrasi yang menggabungkan semua alat yang dibutuhkan untuk menulis perangkat lunak dalam antarmuka pengguna tunggal.

Pastikan telah menghubungkan Pico ke Raspberry Pi atau komputer atau laptop melalui kabel micro USB. Setelah Thonny terbuka, klik kata "Python" diikuti oleh nomor versi di kanan bawah jendela Thonny. Ini menunjukkan interpreter Python yang digunakan saat ini. Anda perlu mengubah interpreter untuk menjalankan program di MicroPython di Pico Pi. Pilih "MicroPython (Raspberry Pi Pico)" dari daftar yang muncul. Jika tidak melihat opsi tersebut, periksa koneksi Pico Pi dan pastikan kabel micro USB terpasang dengan benar ke Raspberry Pi atau komputer.



Gambar 2.14 Memilih interpreter micropython



TIDAK ADA OPSI RASPBERRY PI PICO?

Interpreter Raspberry Pi Pico hanya tersedia di versi terbaru Thonny. Jika Anda menjalankan versi yang lebih lama dan tidak dapat memperbaruinya, cari 'MicroPython (generik)' sebagai gantinya. Jika versi Thonny Anda masih lebih lama dan tidak memiliki opsi juru bahasa di kanan bawah jendela dan Anda tidak dapat memperbaruinya, klik 'Beralih ke mode biasa' di kanan atas, mulai ulang Thonny, klik menu Run, dan klik 'Pilih juru bahasa.' Klik panah tarik-turun di sebelah 'Penerjemah yang sama yang menjalankan Thonny (default)'; klik 'MicroPython (generik)' dalam daftar, lalu klik panah tarik-turun di sebelah 'Port' dan klik 'Papan dalam mode FS' dalam daftar itu sebelum mengklik OK untuk mengonfirmasi perubahan Anda.

Setelah instalasi berhasil, versi MicroPython dan Raspberry Pi Pico akan muncul di Python Shell. Untuk mengujinya, kita dapat menulis fungsi print untuk menampilkan "Hello World". Tekan Enter untuk menjalankan kode, selanjutnya akan ditampilkan Hello World pada Shell yang ditunjukkan dalam Gambar 2.15.



Gambar 2.15 MicroPython mencetak pesan 'Hello, World!' di area Shell

2.3.3 Membuat Program LED Berkedip Pada Pico Onboard

1. LED on-board pada Raspberry Pi Pico terhubung ke GPIO25. Salin kode berikut pada editor IDE Thonny.

```
1 from machine import Pin
2 import utime
3
4 led = Pin(25, Pin.OUT)
5 led.low()
6
7 while True:
8     led.value(1)
9     utime.sleep(1)
10
11     led.value(0)
12     utime.sleep(1)
```

2. Simpan program tersebut di komputer dan beri nama yang unik seperti blink.py.



3. Program akan berjalan dan menunjukkan toggling LED pada Pi Pico

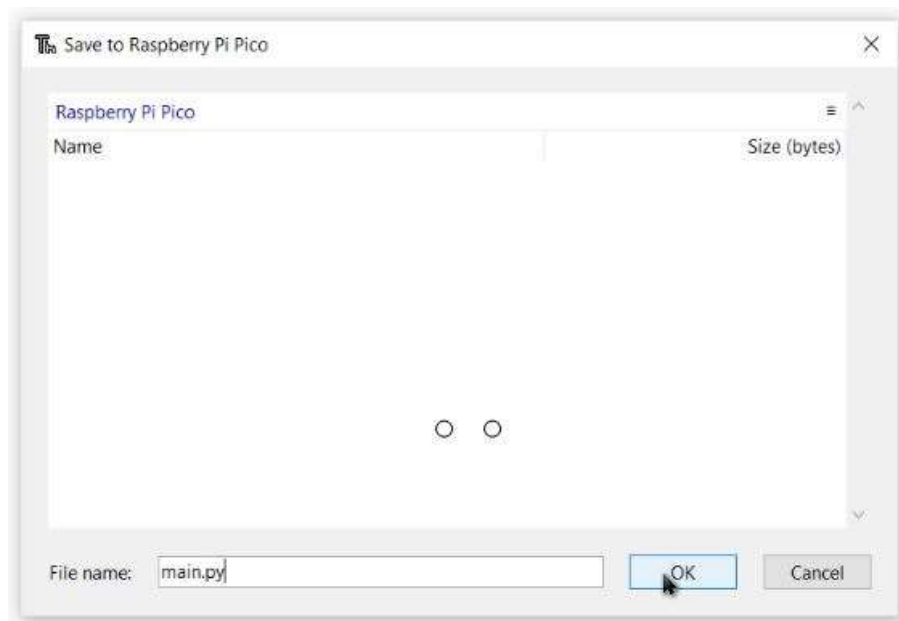


Sekarang lepas USB dari Raspberry Pi Pico dan pasang kembali, yang terjadi adalah LED tidak berkedip. Ini karena program disimpan di komputer, bukan di Raspberry Pi Pico Board.

Agar sistem berfungsi bahkan setelah mencolokkan dan mencabut kabel USB, File harus disimpan ke dalam Pico Board. Untuk melakukan itu buka tab baru dan buka program yang sama dan klik simpan. Kali ini simpan di Pico Board.



Simpan dan beri nama main.py seperti yang ditunjukkan pada gambar di bawah ini.

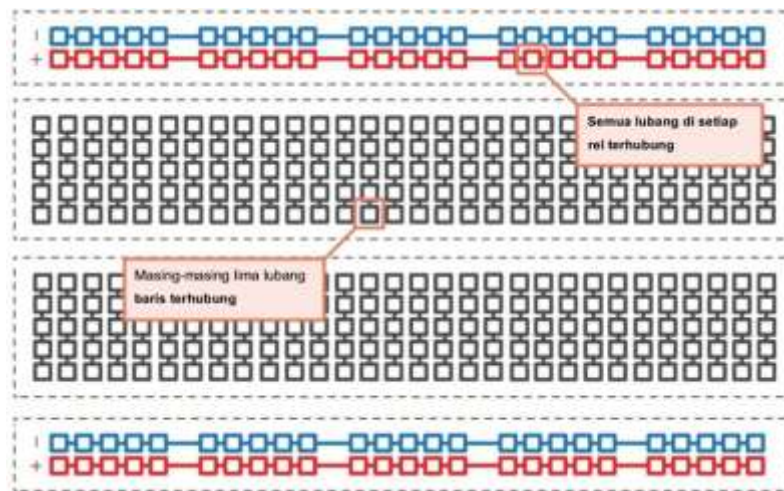


Perhatikan LED akan berkedip bahkan setelah mencolokkan dan melepas kabel daya. Ini karena program disimpan di Pico Pi.

2.3.4 Membuat Program LED Berkedip Dengan LED Eksternal

Projek-projek berikutnya akan lebih mudah diselesaikan jika menggunakan papan tempat memotong breadboard untuk meletakkan komponen dan membuat sambungan listrik. Breadboard ditutupi dengan lubang – diberi jarak, agar sesuai dengan komponen, dengan jarak 2,54 mm. Di

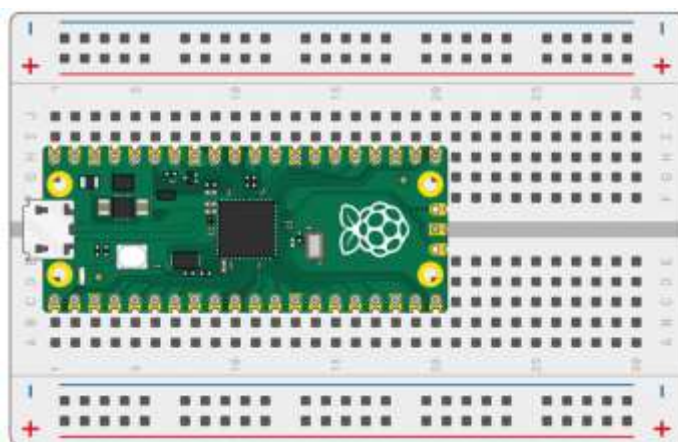
bawah lubang ini adalah strip logam yang berfungsi seperti kabel jumper. Ini berjalan dalam barisan di seluruh breadboard, dengan sebagian besar papan memiliki celah di tengah untuk membaginya dua bagian.



Gambar 2.16 Sambungan Internal Breadboard

Banyak tipe breadboard juga memiliki huruf di bagian atas dan angka di bagian samping. Ini memungkinkan menemukan lubang tertentu: A1 adalah sudut kiri atas, B1 adalah lubang di sebelah kanan, sementara B2 berjarak satu lubang dari sana. A1 terhubung ke B1 dengan strip logam tersembunyi, tapi tidak lubang bertanda 1 selalu terhubung ke lubang mana pun yang bertanda 2.

Dorong Pico ke breadboard sehingga diantara celah tengah dan port micro USB di bagian paling atas papan pada Gambar 2.17. Pin kiri atas, Pin 0, harus ada di breadboard baris yang ditandai dengan 1, jika breadboard diberi nomor. Sebelum mendorong Pico ke bawah, lakukan dan pastikan semua pin tajuk berada pada posisi yang benar, jika pin bengkok akan sulit untuk luruskan lagi dan kemungkinan akan patah.



Gambar 2.17. Cara Penempatan Rapsberry Pi Pico Pada Breadboard

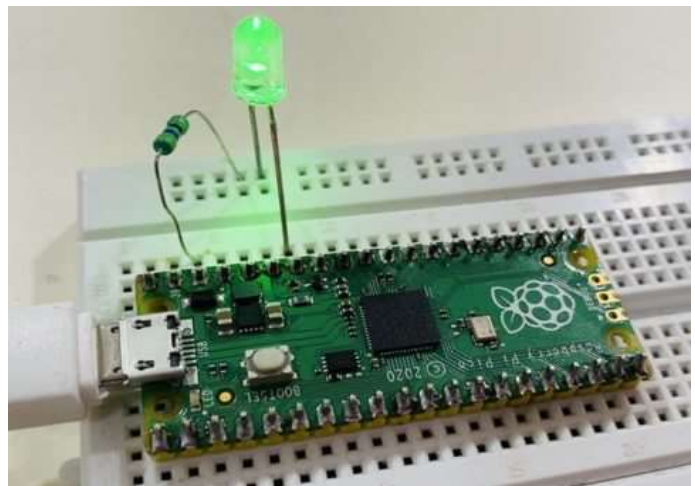
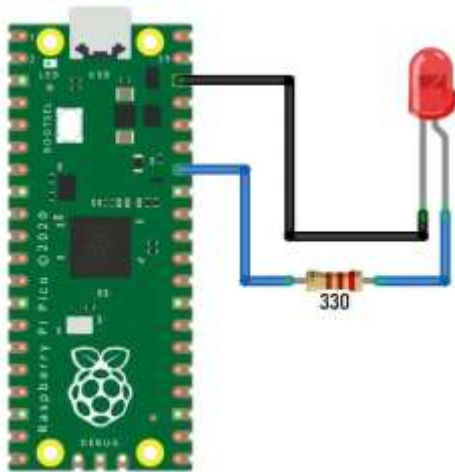
Sekarang mari kita tulis kode Blink LED Raspberry Pi Pico untuk Blink LED eksternal yang terhubung ke Raspberry Pi Pico GPIO Pin28. Skema diberikan di bawah ini. Hubungkan LED ke GPIO 28 melalui resistor 330 ohm.

Projek 1 : LED Blink

A. Alat dan Bahan

- ☐ 1 Buah LED 5mm atau LED 3mm
- ☐ 1 Buah Resistor 100Ω - 330 Ω
- ☐ 2 Buah Kabel Jumper
- ☐ 1 Buah Project board
- ☐ 1 Buah Raspberry Pi Pico

B. Rangkaian Kerja



C. Kode Program

```

1  from machine import Pin
2  import utime
3
4  led = Pin(28, Pin.OUT)
5  led.low()
6
7  while True:
8      led.value(1)
9      utime.sleep(1)
10
11     led.value(0)
12     utime.sleep(1)

```

D. Analisa Program

- ✓ Baris ke-1 module machine yang diperlukan untuk mengenali PIN IO dari Pico Pi
- ✓ Baris ke-2 module utime untuk fungsi tunda waktu
- ✓ Baris ke-4 fungsi membuat variabel led dan seting GPIO 28 sebagai output.

- ✓ Baris ke-5 fungsi variabel led memanggil fungsi low() untuk memastikan LED di-off-kan
- ✓ Baris ke-7 while(True) mengulang perintah ke 8 sampai ke 12 selama kondisinya True.
 - Baris ke-8 GPIO-28 diberi logika True atau 1 sehingga LED menyala
 - Baris ke-9 fungsi sleep(1) untuk mengatur tunda waktu dalam satu detik
 - Baris ke-11 GPIO-28 diberi logika False atau 0 sehingga LED mati
 - Baris ke-12 fungsi sleep(1) untuk mengatur tunda waktu dalam satu detik, selanjutnya program akan mengulang proses mulai dari baris ke-7 sampai baris ke-12 terus menerus selama kondisinya True.

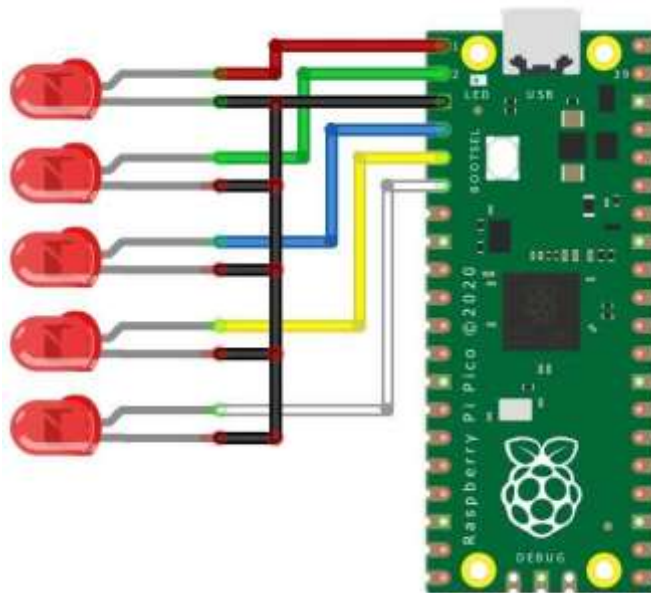
Tugas 1 : Lampu Berjalan

A. Deskripsi

Buatlah program lampu berjalan dengan 5 buah LED dengan menggunakan MicroPython dengan ketentuan sebagai berikut:

- Lampu berjalan ke kanan secara berurutan dengan tunda waktu yang sama
- Lampu berjalan dari kanan dan kiri
- Lampu menyala dari tengah kemudian menyebar ke kanan dan kiri

B. Rangkaian Kerja



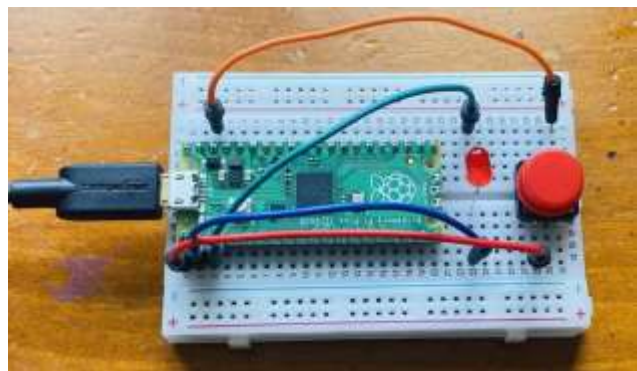
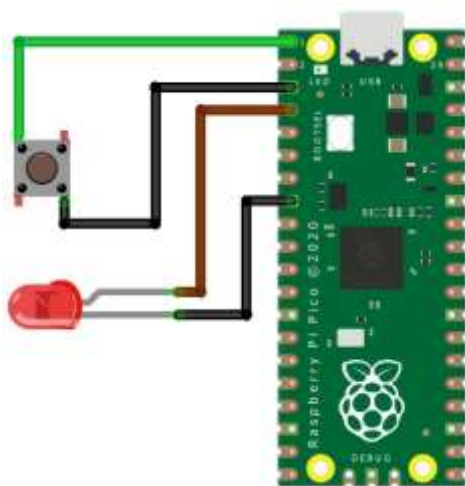
Raspberry Pi Pico	LED
GP0	Led1_VCC (+)
GP1	Led2_VCC (+)
GP2	Led3_VCC (+)
GP3	Led4_VCC (+)
GP4	Led5_VCC (+)
GND	GND (-)

Projek 2 : Push Button Switch

A. Alat dan Bahan

- ☐ 1 Buah LED 5mm atau LED 3mm
- ☐ 1 Buah Resistor 100Ω - 330 Ω
- ☐ 1 Buah Push Button Switch
- ☐ 4 Buah Kabel Jumper
- ☐ 1 Buah Project board
- ☐ 1 Buah Raspberry Pi Pico

B. Rangkaian Kerja



C. Kode Program

```

1  from machine import Pin
2  import utime
3
4  button = Pin(0, Pin.IN)
5  led = Pin(1, Pin.OUT)
6  led.low()
7
8  while True:
9      if button.value() == 0:
10         led.value(1)
11         utime.sleep(2)
12     else:
13         led.value(0)

```

D. Analisa Program

- ✓ Baris ke-1 module machine yang diperlukan untuk mengenali PIN IO dari Pico Pi
- ✓ Baris ke-2 module utime untuk fungsi tunda waktu
- ✓ Baris ke-4 fungsi membuat variabel button dan seting GPIO 0 sebagai input.
- ✓ Baris ke-5 fungsi membuat variabel led dan seting GPIO 1 sebagai output.
- ✓ Baris ke-6 fungsi variabel led memanggil fungsi low() untuk memastikan LED di-off-kan

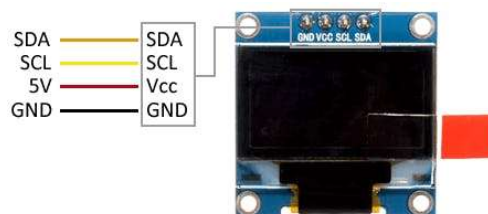
- ✓ Baris ke-8 while(True) mengulang perintah ke 8 sampai ke 13 selama kondisinya True.
 - Baris ke-9 Jika nilai button=0 maka:
 - Baris ke-10 GPIO-1 diberi logika True atau 1 sehingga LED menyala
 - Baris ke-11 fungsi sleep(2) untuk mengatur tunda waktu dalam dua detik
 - Baris ke-12 Jika nilai button=1 maka:
 - Baris ke-10 GPIO-1 diberi logika False atau 0 sehingga LED padam
 - Selanjutnya program akan mengulang proses mulai dari baris ke-8 sampai baris ke-13 terus menerus selama kondisinya True.

Projek 1 : OLED Display (ssd1306)

A. Teori Dasar

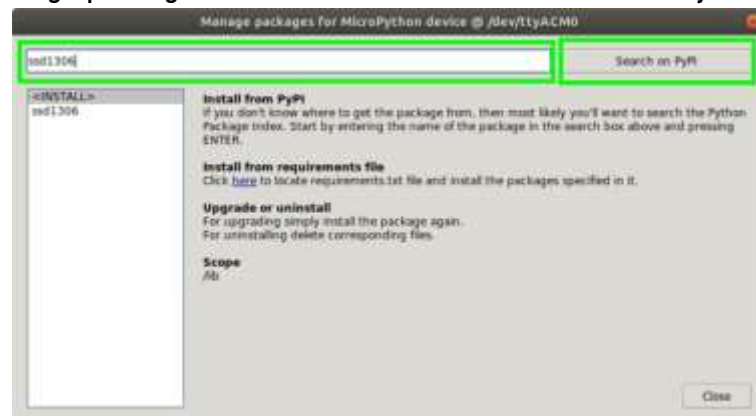
OLED (Organic Light-Emitting Diode) adalah perangkat output konsumsi daya rendah yang mampu menampilkan teks dan gambar dengan mengontrol setiap piksel.

Ukuran yang paling umum adalah 128x64 piksel (0,96 inci), tetapi ukuran 128x32 piksel (0,91 inci) juga tersedia di toko elektronik. Layar mereka biasanya satu warna atau dua warna. Yang satu warna terbuat dari piksel biru atau putih, sedangkan model dua warna biasanya memiliki bagian atas berwarna kuning dan bagian bawah berwarna biru.

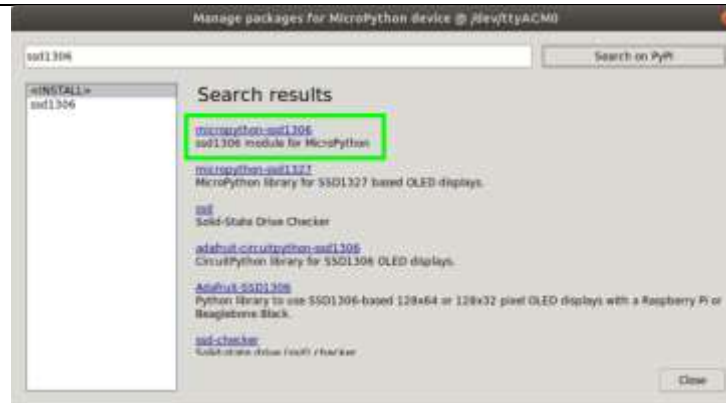


Cara menambah library OLED ke IDE Thonny:

1. Klik Tools > Manage packages > Ketik "ssd1306" > Klik "Search on PyPI"



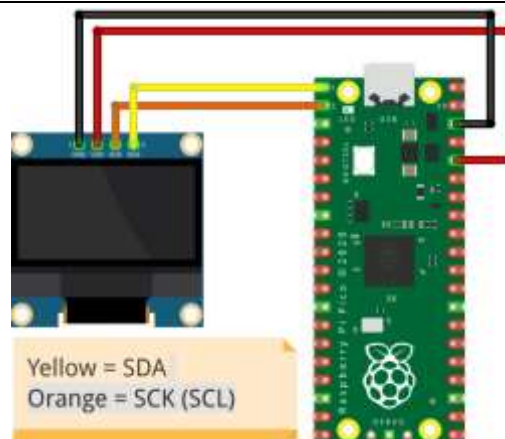
2. Klik "micropython-ssd1306" kemudian klik Instal. Ini akan menyalin library ke folder, lib di Pico.



B. Alat dan Bahan

- ☐ 1 Buah LED 5mm atau LED 3mm
- ☐ 1 Buah OLED
- ☐ 2 Buah Kabel Jumper
- ☐ 1 Buah Project board
- ☐ 1 Buah Raspberry Pi Pico

C. Rangkaian Kerja



D. Kode Program

```

1 from machine import Pin, I2C
2 from ssd1306 import SSD1306_I2C
3
4 WIDTH =128
5 HEIGHT= 64
6
7 i2c=I2C(0,scl=Pin(1),sda=Pin(0),freq=200000)
8 oled = SSD1306_I2C(WIDTH,HEIGHT,i2c)
9
10 while True:
11     oled.fill(0)
12     oled.text("PROJEK IPAS-TI", 0, 0)
13     oled.text("VEDC Malang", 0, 40)
14     oled.show()

```

E. Analisa Program

- ✓ Baris ke-1 module machine yang diperlukan untuk mengenali PIN dan I2C
- ✓ Baris ke-2 module ssd1306 untuk menampilkan dan komunikasi OLED
- ✓ Baris ke-4 variabel WIDTH diset 128 sebagai lebar layar

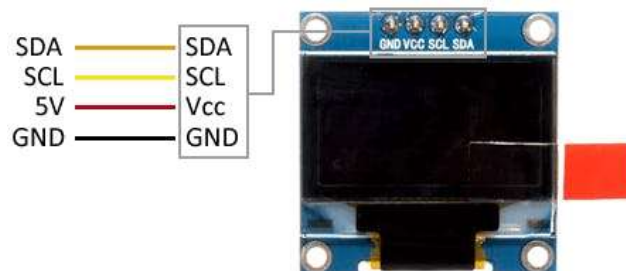
- ✓ Baris ke-5 variabel HEIGHT diset 64 sebagai tinggi layar
- ✓ Baris ke-7 inialisasi komunikasi I2C
- ✓ Baris ke-8 inialisai variabel oled
- ✓ Baris ke-10 while(True) mengulang perintah ke 11 sampai ke 14 selama kondisinya True.
 - Baris ke-11 seting tampilan layar
 - Baris ke-12 mencetak text PROJEK IPAS-TI pada posisi x = 0, y = 0
 - Baris ke-13 mencetak text VEDC Malang pada posisi x = 0, y = 40
 - Baris ke-14 memanggil fungsi show() untuk menampilkan semua text pada layar OLED, selanjutnya program akan mengulang proses mulai dari baris ke-11 sampai baris ke-14 terus menerus selama kondisinya True.

Projek 1 : OLED Display (SH1106)

A. Teori Dasar

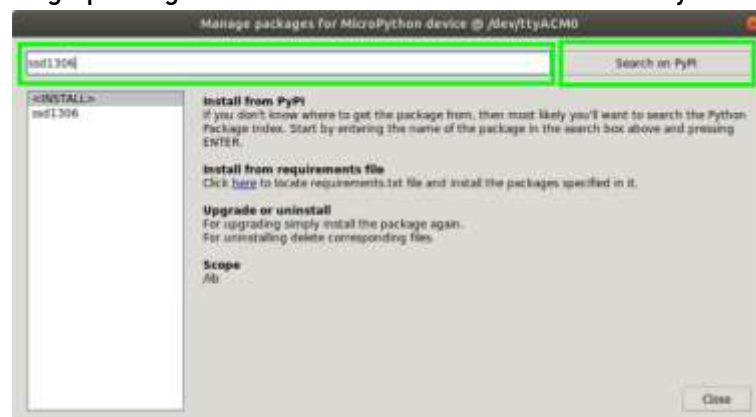
OLED (Organic Light-Emitting Diode) adalah perangkat output konsumsi daya rendah yang mampu menampilkan teks dan gambar dengan mengontrol setiap piksel.

Ukuran yang paling umum adalah 128x64 piksel (0,96 inci), tetapi ukuran 128x32 piksel (0,91 inci) juga tersedia di toko elektronik. Layar mereka biasanya satu warna atau dua warna. Yang satu warna terbuat dari piksel biru atau putih, sedangkan model dua warna biasanya memiliki bagian atas berwarna kuning dan bagian bawah berwarna biru.

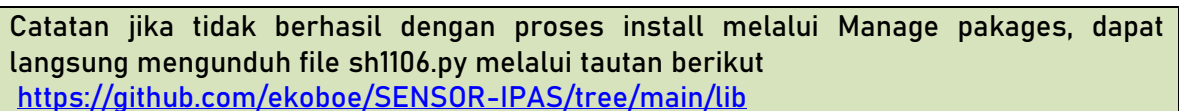


Cara menambah library OLED ke IDE Thonny:

1. Klik Tools > Manage packages > Ketik "sh1106" > Klik "Search on PyPI"



3. Klik "micropython-ssd1306" kemudian klik Instal. Ini akan menyalin library ke folder, lib di Pico.



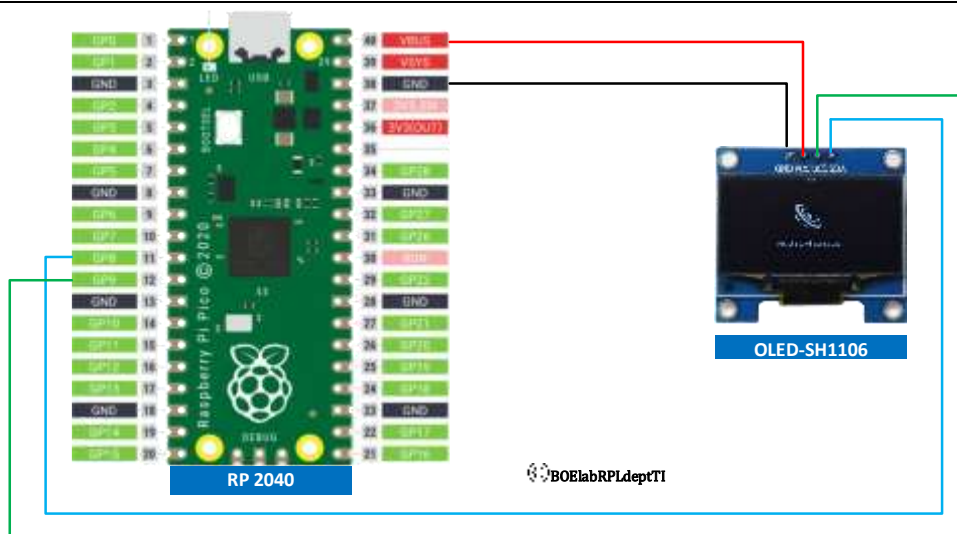
The screenshot shows the Thon IDE interface. On the left, the file explorer displays a directory structure for a Raspberry Pi Pico, including files like `time_module.py`, `example1.c`, `gpio.py`, `uart.py`, and `uSBC2.py`. The main editor window shows a Python script with the following content:

```
MicroPython v1.21.0 on 2023-10-06; Raspberry Pi Pico w with RP2040
Type "help()" for more information.
>>>
```

A context menu is open over the file explorer, showing options like "Focus into", "Open in default external app", "Hide hidden files", "Download to D:\Tahun 2024\DLAA3\RPAS\CodeProgram", "New file...", "New directory...", "Delete", and "Properties". A red arrow points from the "New directory..." option to a "New directory" dialog box that is open on the right. The dialog box prompts the user to "Enter name for new directory under /lib" and has input fields for "lib" and "OK" and "Cancel" buttons.

- ☐ 1 Buah LED 5mm atau LED 3mm
- ☐ 1 Buah OLED SH1106
- ☐ 2 Buah Kabel Jumper
- ☐ 1 Buah Project board
- ☐ 1 Buah Raspberry Pi Pico

C. Rangkaian Kerja



D. Kode Program

```

1 from machine import Pin, I2C
2 from sh1106 import SH1106_I2C
3 from utime import sleep
4
5 i2c=I2C(0, sda=Pin(8), scl=Pin(9), freq=400000)
6
7 display = SH1106_I2C(128, 64, i2c, Pin(4), 0x3c)
8 display.sleep(False)
9
10 while True:
11     display.fill(0)
12     display.text('IPAS - BOE ',23, 3, 1)
13     display.hline(0, 63, 127, 1)
14     display.hline(0, 0, 127, 1)
15     display.hline(0, 14, 127, 1)
16     display.vline(0, 0, 63, 1)
17     display.vline(127, 0, 63, 1)
18     display.show()
19     sleep(1)

```

E. Analisa Program

- ✓ Baris ke-1 module machine yang diperlukan untuk mengenali PIN dan I2C
- ✓ Baris ke-2 module sh1106 untuk menampilkan dan komunikasi OLED
- ✓ Baris ke-3 module utime untuk fungsi tunda waktu (sleep)
- ✓ Baris ke-5 inisialisasi komunikasi I2C(0, sda=Pin(8), scl=Pin(9), freq=400000)
- ✓ Baris ke-7 seting fungsi SH1106_I2C(128, 64, i2c, Pin(4), 0x3c)
- ✓ Baris ke-8 display diseting di offkan
- ✓ Baris ke-10 while(True) mengulang perintah ke 11 sampai ke 19 selama kondisinya True.
 - Baris ke-11 seting tampilan layar clear
 - Baris ke-12 mencetak text IPAS-BOE pada posisi x = 23, y = 3, dengan mode 1 ON
 - Baris ke-13 mencetak garis horisontal pada titik awal x= 0 , y = 63 sepanjang 127 dengan mode 1 ON
 - Baris ke-14 mencetak garis horisontal pada titik awal x= 0 , y = 0 sepanjang 127 dengan mode 1 ON

- Baris ke-15 mencetak garis horisontal pada titik awal $x = 0$, $y = 14$ sepanjang 127 dengan mode 1 ON
- Baris ke-16 mencetak garis horisontal pada titik awal $x = 0$, $y = 0$ dengan tinggi 63 dengan mode 1 ON
- Baris ke-17 mencetak garis horisontal pada titik awal $x = 127$, $y = 0$ dengan tinggi 63 dengan mode 1 ON
- Baris ke-18 memanggil fungsi `show()` untuk menampilkan semua text pada layar OLED, selanjutnya program akan mengulang proses mulai dari baris ke-11 sampai baris ke-19 terus menerus selama kondisinya True.

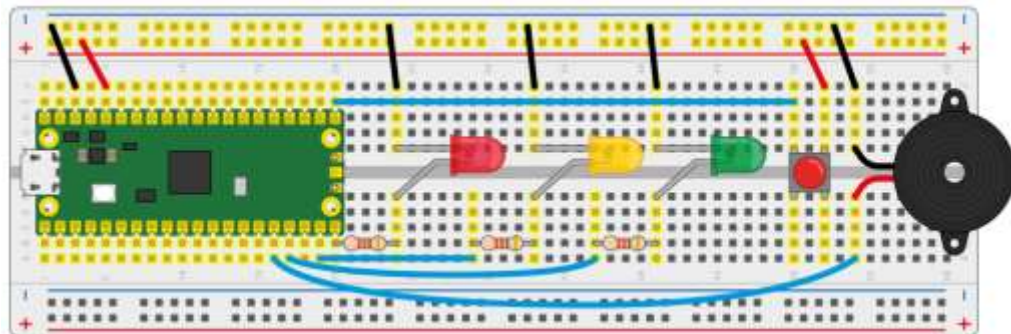
Tugas 2 : Sistem Traffic Light Penyeberangan

A. Deskripsi

Buatlah program sistem traffic light dengan penyeberangan dengan menggunakan MicroPython dengan ketentuan sebagai berikut:

- Perhatikan LED: pertama LED merah akan menyala, memberi tahu lalu lintas untuk berhenti; selanjutnya, LED kuning akan menyala untuk memperingatkan pengemudi bahwa lampu akan segera berubah; selanjutnya kedua LED mati dan LED hijau menyala untuk memberi tahu lalu lintas bahwa itu bisa lewat; kemudian LED hijau padam dan yang kuning menyala untuk memperingatkan pengemudi bahwa lampu akan berubah lagi; akhirnya, LED kuning padam – dan loop dimulai kembali dari awal, dengan LED merah menyala.
- Pola akan berulang hingga tombol Stop ditekan, dengan looping terus menerus. Ini didasarkan pada pola lampu lalu lintas yang digunakan dalam sistem kontrol lalu lintas dan memberi mobil hanya lima detik untuk melewati lampu tidak akan membiarkan lalu lintas mengalir dengan sangat bebas!
- Namun, lampu lalu lintas yang sebenarnya tidak hanya ada untuk kendaraan jalan: namun juga ada untuk melindungi pejalan kaki, memberi mereka kesempatan untuk menyeberang jalan yang sibuk dengan aman. Jenis yang paling umum dari lampu ini dikenal sebagai penyeberangan cerdas yang ramah pengguna yang dioperasikan pejalan kaki atau penyeberangan.
 - ✓ Untuk mengubah lampu lalu lintas Anda menjadi penyeberangan puffin, Anda memerlukan dua hal: saklar tombol tekan, sehingga pejalan kaki dapat meminta lampu untuk membiarkan mereka menyeberang jalan; dan bel, sehingga pejalan kaki tahu kapan giliran mereka untuk menyeberang.

B. Rangkaian Kerja



Raspberry Pi Pico	Keterangan
GP16	Button
GP12	Buzzer
GP15	LED Merah
GP14	LED Kuning
GP13	LED Hijau

DAFTAR PUSTAKA

- [1] Gareth Halfacree, Ben Everard (2021), Get started with MicroPython on Raspberry Pi Pico, Raspberry Pi Trading Ltd, Maurice Wilkes Building, St. John's Innovation Park, Cowley Road, Cambridge, CB4
- [2] Richard Blum (2014), Python Programming for Raspberry Pi in 24 Hours, Sams Teach Yourself.
- [3] <https://www.raspberrypi.org/documentation/usage/python/>
- [4] <https://www.programiz.com/python-programming>
- [5] <https://how2electronics.com/>

PENULIS




Eko Subiyantoro adalah widyaiswara di BBPPMPV BOE-VEDC Malang di Departemen Teknologi Informasi. Lahir di Banyuwangi pada tahun 1975. Pendidikan SD, SMP, dan STM (sekarang SMK) diselesaikan di kota kelahirannya. Selanjutnya ia menyelesaikan S1 di Program Studi Pendidikan Teknik Elektronika FPTK IKIP Yogyakarta (sekarang UNY) pada tahun 1998. Sarjana Sains Terapan (SST) diperolehnya melalui program beasiswa Dikmenjur (sekarang Direktorat Pembinaan Sekolah Menengah Kejuruan) di PENS ITS Surabaya pada tahun 2002 pada Prodi Teknologi Informasi dengan predikat cum laude. Magister Teknik diperolehnya melalui program beasiswa unggulan BPKLN di ITS Surabaya Program Studi Teknik Elektro konsentrasi Jaringan Cerdas

Multimedia pada tahun 2009 juga dengan predikat cum laude. Doktor diperolehnya juga melalui beasiswa unggulan bagi pegawai Kemdikbud di Prodi S3 Ilmu Komputer FMIPA Universitas Gadjah Mada.

Pengalaman luar negeri diperolehnya melalui program Inwent dari pemerintah Jerman pada tahun 2003 dalam program Advance Automation Training. Selain itu, pada tahun 2006 mengikuti Expert Program Maintenance and Repair di Abha College Kingdom of Saudi Arabia (KSA). Selanjutnya pada tahun 2013 melalui IGI kerja sama Indonesia-Jerman diikutkan dalam program Modern Management School di Kassel Jerman.

Penulis memiliki pengalaman kerja di PPPPTK BOE/VEDC Malang sebagai Kepala Departemen Teknologi Informasi selama 2 periode 2010–2013 dan 2013–2015. Selain itu juga sebagai tim pengembang Java Education Network Indonesia (JENI), sebuah sistem yang diinisiasi BPKLN Kemdikbud dalam kurun waktu tahun 2007–2009. Sertifikasi yang diperoleh dalam bidang pemrograman adalah Sun Certified Java Programmer (SCJP) pada tahun 2008.

Eko Subiyantoro dapat dihubungi melalui alamat e-mail ekovedc@gmail.com

	Bidang Keahlian: Teknologi Informasi	Laboratorium: RPL-239	Tanggal: Maret 2024	70	01	02	VT	Hal 28 - 28
	Program Keahlian: Pengembangan PL dan Gim	Komputasi Physical Rapsberry Pi Pico	Rev.Tanggal: 09/03/2024	Dibuat oleh: <i>Eko Subiyantoro</i>				