Getting Started with Raspberry Pi Pico

Using C/C++ SDK

bit.ly/piday-absen

whoami

Alwin Arrasyid

IoT Software Lead, DycodeX

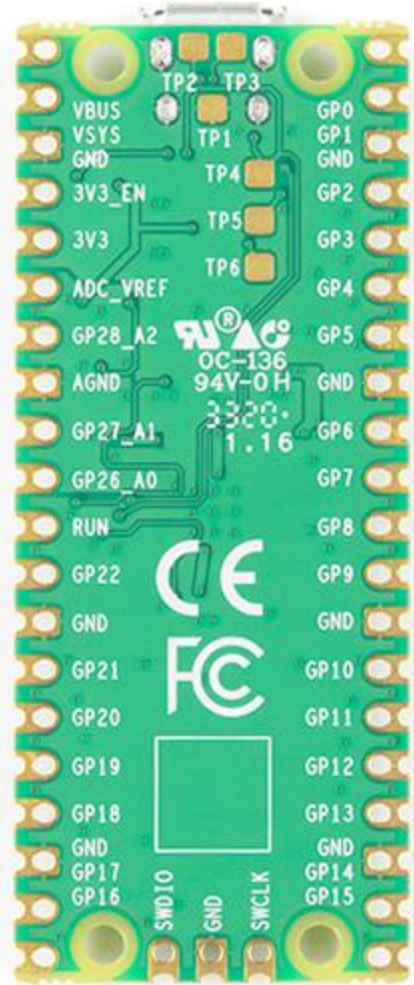
winter@dycode.com

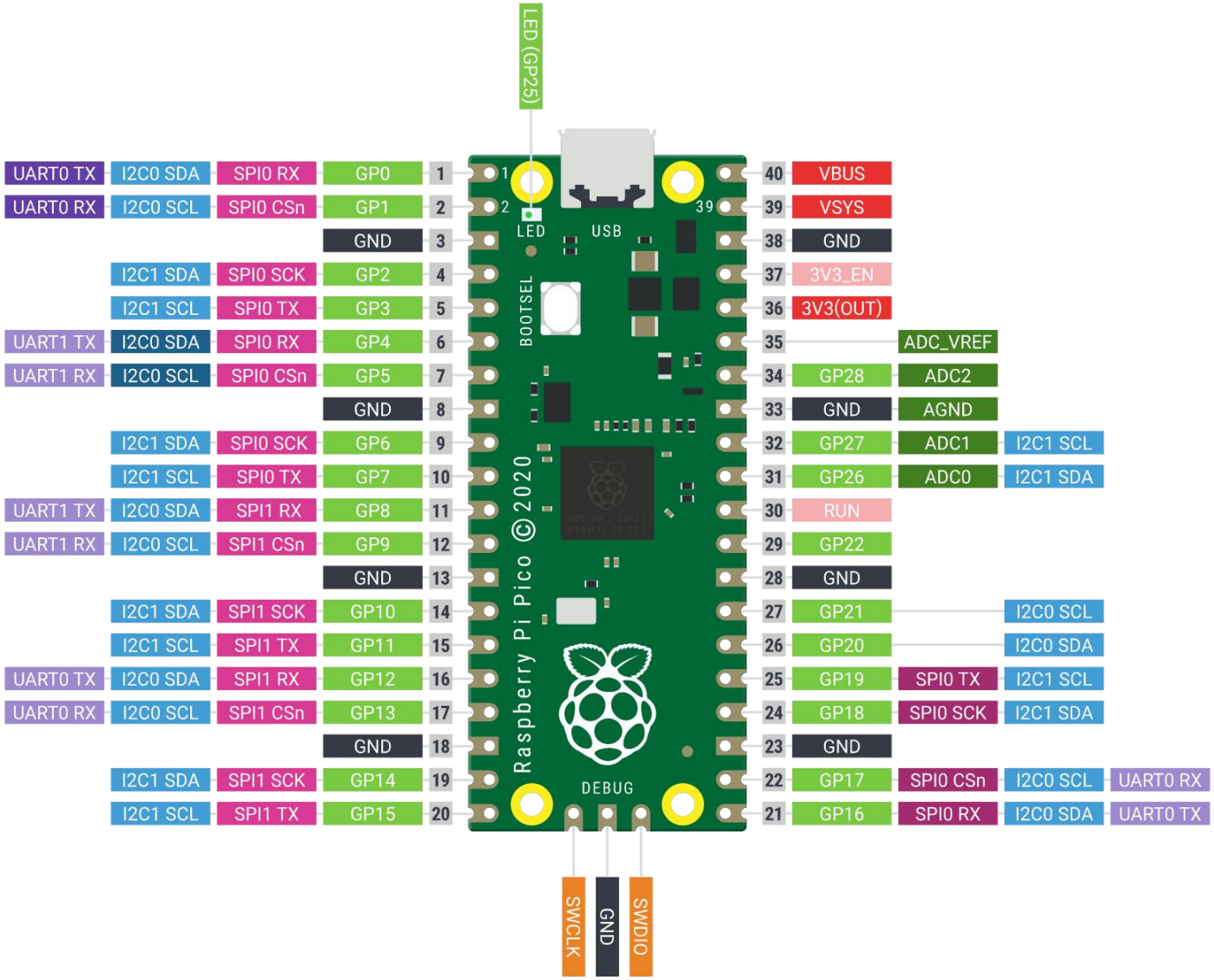
alwint3r.github.io

Raspberry Pi Pico

- Powered by RP2040, Dual ARM Cortex-M0+ @ 133 MHz
- 264 KB SRAM and 2 MB on-board flash memory
- 26 multi-function GPIO
- 3 pins for ADC
- Serial Wire Debug interface
- 8 programmable I/O
- Drag & drop programming
- Castellated module
- 4 USD


1
2
39
LED
USB
BOOTSEL
© 2020
Raspberry Pi Pico
DEBUG


TP2
TP3
TP1
TP4
TP5
TP6
VBUS
VSYS
GND
3V3_EN
3V3
ADC_VREF
GP28_A2
AGND
GP27_A1
GP26_A0
RUN
GP22
GND
GP21
GP20
GP19
GP18
GND
GP17
GP16
GP0
GP1
GND
GP2
GP3
GP4
GP5
GND
GP6
GP7
GP8
GP9
GND
GP10
GP11
GP12
GP13
GND
GP14
GP15
SWDIO
GND
SWCLK
OC-136
94V-0 H
3320.
1.16


LED (GP25)
LED
USB
BOOTSEL
DEBUG
Raspberry Pi Pico ©2020
SWCLK
GND
SWDIO
UART0 TX I2C0 SDA SPI0 RX GP0 1
UART0 RX I2C0 SCL SPI0 CSn GP1 2
GND 3
I2C1 SDA SPI0 SCK GP2 4
I2C1 SCL SPI0 TX GP3 5
UART1 TX I2C0 SDA SPI0 RX GP4 6
UART1 RX I2C0 SCL SPI0 CSn GP5 7
GND 8
I2C1 SDA SPI0 SCK GP6 9
I2C1 SCL SPI0 TX GP7 10
UART1 TX I2C0 SDA SPI1 RX GP8 11
UART1 RX I2C0 SCL SPI1 CSn GP9 12
GND 13
I2C1 SDA SPI1 SCK GP10 14
I2C1 SCL SPI1 TX GP11 15
UART0 TX I2C0 SDA SPI1 RX GP12 16
UART0 RX I2C0 SCL SPI1 CSn GP13 17
GND 18
I2C1 SDA SPI1 SCK GP14 19
I2C1 SCL SPI1 TX GP15 20
40 VBUS
39 VSYS
38 GND
37 3V3_EN
36 3V3(OUT)
35 ADC_VREF
34 GP28 ADC2
33 GND AGND
32 GP27 ADC1 I2C1 SCL
31 GP26 ADC0 I2C1 SDA
30 RUN
29 GP22
28 GND
27 GP21 I2C0 SCL
26 GP20 I2C0 SDA
25 GP19 SPI0 TX I2C1 SCL
24 GP18 SPI0 SCK I2C1 SDA
23 GND
22 GP17 SPI0 CSn I2C0 SCL UART0 RX
21 GP16 SPI0 RX I2C0 SDA UART0 TX
Power
Ground
UART / UART (default)
GPIO, PIO, and PWM
ADC
SPI / SPI (default)
I2C / I2C (default)
System Control
Debugging

Agenda

- Install Pico C/C++ SDK on Raspberry Pi OS / Debian
- Running example project
- Coding using Visual Studio Code
- Interfacing with peripherals

Installing the C/C++ SDK

Installation Methods

- Using setup script
- Manually install every required software by hand

Manual Installation

- Required softwares:
  - git
  - cmake
  - gcc-arm-none-eabi
  - libnewlib-arm-none-eabi
  - build-essentials

Manual Installation

- Clone pico-sdk repository from https://github.com/raspberrypi/pico-sdk
- Clone pico-examples repository from https://github.com/raspberrypi/pico-examples

Manual Installation

- Set PICO_SDK_PATH environment variable to the pico-sdk path.
- You should set it in ~/.bashrc file so each time you open a terminal the environment variable is already set for you

Demo

Coding using Visual Studio Code

Required Extensions

- Cortex-Debug
- CMake Tools
- C/C++

Demo

Using Pico Project Generator

Using Pico Project Generator

- Less time setting up project
- No need to code the CMakeLists.txt manually from scratch
- Support for Visual Studio Code as IDE

Required Software

- Python 3.6.4 or newer
  - Debian 9 users will need to install the Python 3.6.4 from source
- tcl/tk
  - Should be available by default on every system after Python installation is complete

Demo

Materials

- Getting Started Guide
  - https://datasheets.raspberrypi.org/pico/getting-started-with-pico.pdf
- RP2040 Datasheet
  - https://datasheets.raspberrypi.org/rp2040/rp2040-datasheet.pdf
- Raspberry Pi Pico Datasheet
  - https://datasheets.raspberrypi.org/pico/pico-datasheet.pdf

Interfacing with Peripherals

Demo

Materials

- Getting Started Guide
  - https://www.raspberrypi.org/documentation/rp2040/getting-started/
  - https://datasheets.raspberrypi.org/pico/getting-started-with-pico.pdf
- RP2040 Datasheet
  - https://datasheets.raspberrypi.org/rp2040/rp2040-datasheet.pdf
- Raspberry Pi Pico Datasheet
  - https://datasheets.raspberrypi.org/pico/pico-datasheet.pdf

Demo Code

- github.com/raspberrypi/pico-examples
- github.com/alwint3r

Thank You!