# Chapter 5. Flash Programming with SWD

Serial Wire Debug (SWD) is a standard interface on Cortex-M-based microcontrollers, which the machine you are using to develop your code (commonly called the *host*) can use to reset the board, load code into flash, and set the code running. Raspberry Pi Pico exposes the RP2040 SWD interface on three pins at the bottom edge of the board. The host can use the SWD port to access RP2040 internals at any time, so there is no need to manually reset the board or hold the BOOTSEL button.

Figure 6. The SWD port is labelled at the bottom of this Pico pinout diagram. The ground (GND) connection is required to maintain good signal integrity between the host and the Pico. The SWDIO pin carries debug traffic in both directions, between RP2040 and the host. The SWCLK pin keeps the connection well-synchronised. These pins connect to a dedicated SWD interface on RP2040, so you don't need to sacrifice any GPIOs to use the SWD port.
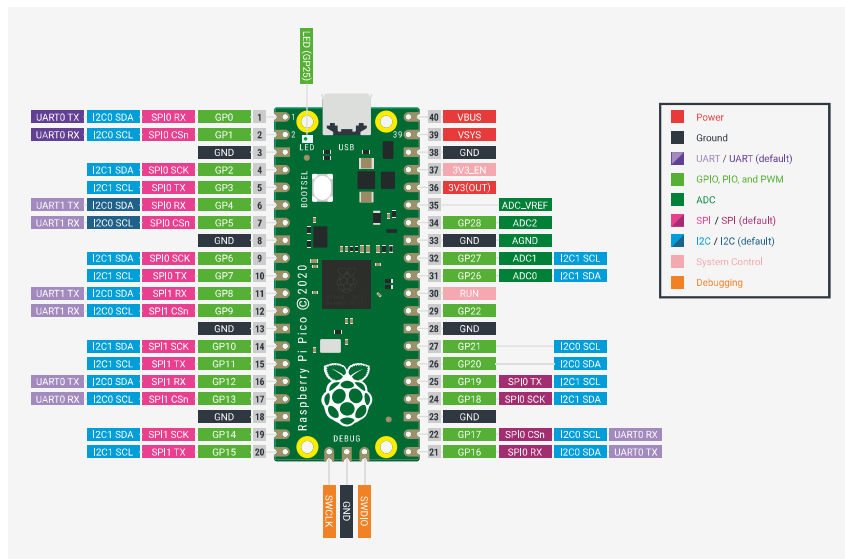
On a Raspberry Pi, you can connect the Pi GPIOs directly to Pico's SWD port, and load code from there. On other machines you will need an extra piece of hardware — a *debug probe* — to bridge a connection on your host machine (like a USB port) to the SWD pins on the Pico. One of the cheapest ways to do this is to use *another* Pico as the debug probe, and this is covered in Appendix A.

This chapter covers how you can connect your machine to Raspberry Pi Pico's SWD port, and use this to write programs into flash and run them.

💡 **TIP**

If you use an IDE like Visual Studio Code (Chapter 7), this can be configured to use SWD automatically behind the scenes, so you click the play button and the code runs, as though you were running native code on your own machine.

ℹ️ **NOTE**

You can also use SWD for interactive debugging techniques like setting breakpoints, stepping through code execution line-by-line, or even peeking and poking IO registers directly from your machine without writing any RP2040 software. This is covered in Chapter 6.

5.1. Installing OpenOCD

To access the SWD port on a microcontroller, you need a program on your host machine called a *debug translator*, which understands the SWD protocol, and knows how to control the processor (two Cortex-M0+s in the case of