Introduction to the Pico Pins   
TEACHER GUIDE

In this guide there is a simple overview of the pins which is definitely enough to get you started. This is followed by a more detailed overview of the pins covering pins and protocol information.

**Understanding GPIO Pins on Raspberry Pi Pico**

General-Purpose Input/Output pins allow you to connect input and output devices to Raspberry Pi Pico and obtain or send signals between them.

The diagram below identifies the physical pin numbers and the GPO pin numbers of the Pins on the Pico. Note how the physical pin numbers differ from the GP numbers for example physical pin 1 is GP pin 0. Also note how some physical pins don’t have GP numbers for example pin’s 3, 8 & 12 are ground pins so can’t be used as GPIO pins.

A computer chip with many different colored labels

Description automatically generated

A computer chip with many different colored labels

Description automatically generated

In this second diagram you will see that some GPIO pins can be used for specific purposes such as pin 31 & 32 which are analogue to digital pins.

**Setting Up GPIO Pins**

Before you can use a GPIO pin, you need to set it up. Setting up a pin involves configuring it for a specific purpose, such as input or output. Here's a basic input and output sample.

**Input Mode**

A white background with text

Description automatically generated

**Output Mode**

A screenshot of a computer program

Description automatically generated

**Digital Input and Output**

* Digital Input is used to read binary signals (0 or 1) from external devices like buttons or switches.
* Digital Output Is used to send binary signals (0 or 1) to external devices like LEDs or relays.

**Analog Input**

* Analog Input (ADC) Used to read continuous analog signals (range of values) from sensors like potentiometers or light sensors.
* PWM (Pulse Width Modulation)
* PWM Output Used to control the intensity of devices like LEDs or motors by varying the width of pulses.
* I2C, SPI, UART

**Communication Pins (I2C, SPI, UART)** Used for communication with other devices, enabling your Pico to talk to sensors, displays, or other microcontrollers.

**Raspberry Pi Pico Detailed Pin Out Diagram and Explanation**

A computer chip with many different colored labels

Description automatically generated

**1. Power Pins (VCC and GND): Think of these as the Pico's energy** source. VCC draws or provides power to external devices, similar to plugging into an electrical outlet.

GND acts like its grounding, Raspberry Pi Pico, ground pins (GND) provide a connection to the common ground. Ensuring that the Pico shares the same ground reference as other devices in a circuit is crucial for proper functioning.

**2. Input/Output Pins (GPIO):** These are the Pico's versatile connectors, akin to its hands and fingers. They can either send signals, like activating a light or spinning a motor, or receive signals, such as detecting a button press or reading data from a sensor.

**3. Communication Pins (I2C, SPI, UART):** Picture these as the Pico's multilingual skills. It can communicate using different "languages" to interact with various devices like sensors, displays, or even other Picos.

**4. Analog Pins:** These serve as the Pico's analogue eyes. Unlike a simple on/off switch, these pins can measure and interpret different levels, such as the brightness of light or the temperature of a room.

**Breakdown of the I2C, SPI, and UART, which are different communication protocols that the Raspberry Pi Pico can use:**

**1. I2C (InterIntegrated Circuit):**

**Purpose**: I2C is like a private chat between devices. It allows multiple devices (like sensors, displays, or other microcontrollers) to communicate with each other using just two wires: one for data (SDA) and one for a clock signal (SCL).

**Usage**: Imagine devices passing notes back and forth during class. Each note (data) has an address, so the devices know who the message is for. The Pico has two I2c (pronounced 1 squared c) channels each channel can accommodate several devices

**2. SPI (Serial Peripheral Interface):**

**Purpose**: SPI is a bit like a group discussion. It's a faster communication protocol suitable for connecting devices like displays, memory chips, or other microcontrollers.

**Components**: It typically involves four lines Master Out, Slave In (MOSI); Master In, Slave Out (MISO); Serial Clock (SCK); and Chip Select (CS).

**Usage**: It's like passing a talking stick around in a circle. Only the device with the stick (Chip Select) talks, and others listen.

**3. UART (Universal Asynchronous Receiver/Transmitter):**

**Purpose**: UART is like having a phone conversation. It's a simple way for two devices to talk to each other using two wires: one for transmitting data (TX) and one for receiving data (RX).

**Usage**: One device sends information (TX), and the other listens and receives that information (RX). It's like a continuous, two way conversation.