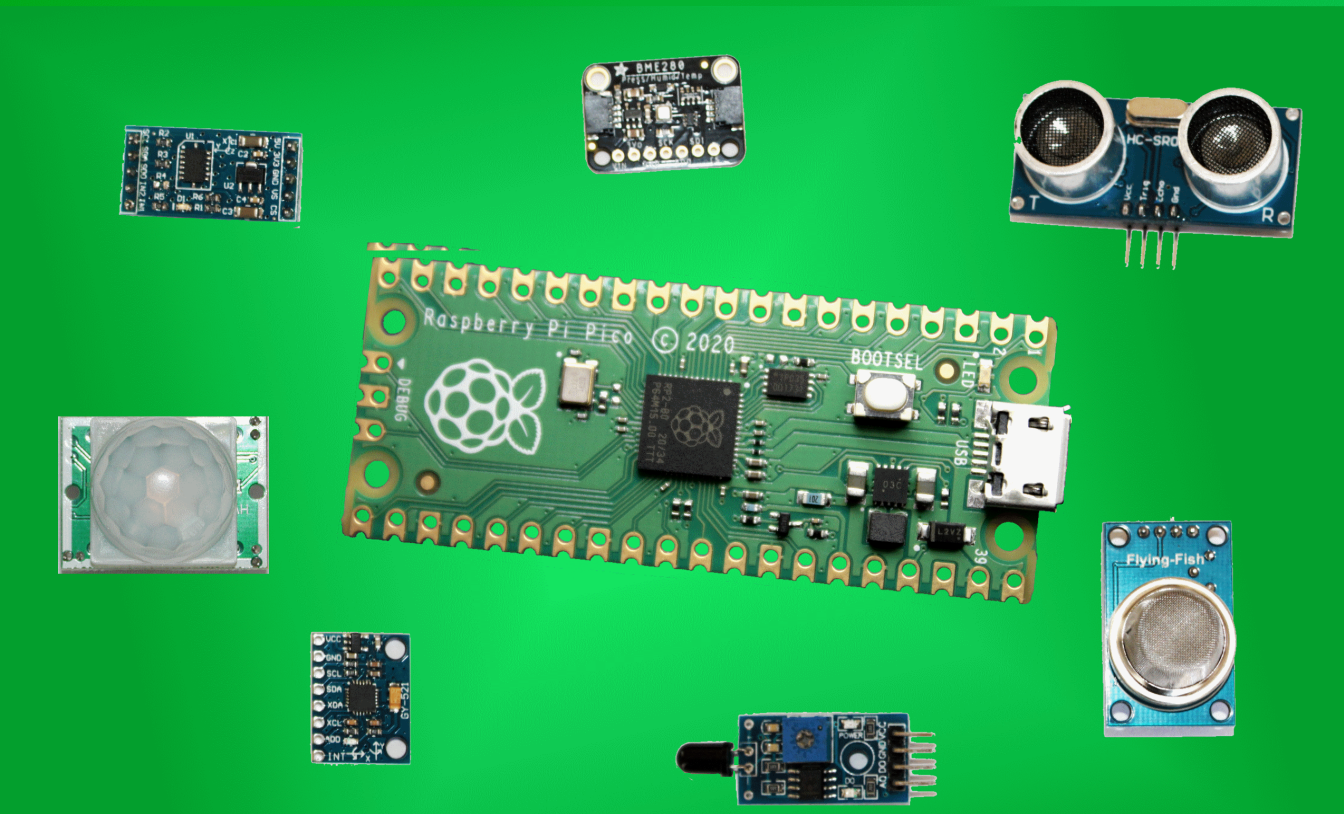
Using Sensors with the Raspberry Pi Pico



Daniel Quadros

Using Sensors with the Raspberry Pi Pico

Daniel Quadros

This book is for sale at http://leanpub.com/picosensors

This version was published on 2023-09-08



This is a Leanpub book. Leanpub empowers authors and publishers with the Lean Publishing process. Lean Publishing is the act of publishing an in-progress ebook using lightweight tools and many iterations to get reader feedback, pivot until you have the right book and build traction once you do.

© 2022 - 2023 Daniel Quadros

Contents

Introduction **1**
The Raspberry Pi Pico 2
Getting the Example Code 4
Using the Examples 4
Organization of This Book 5
Acknowledgments 6
How to Send Feedback 6

Using Sensors **7**
Measuring What We Really Want 7
A Sensor Can Affect What You Are Measuring 7
Accuracy and Resolution 8
ADC Errors 8
Averaging Readings 8
Calibration 8
Readings Validation 8

Programming the Raspberry Pi Pico **9**
Libraries 9
The Official C/C++ SDK 9
The Arduino IDE 9
MicroPython 9
CircuitPython 9
Thonny IDE 10
Seeing Program Output 10
Which Environment Should I Use? 11

Interfaces and Protocols **12**
A Little Bit of Electronics 12
A General View of Interfaces 12

The ADC - Analog to Digital Converter . . . 13
Connecting Digital Signals from Sensors to the Pico . . . 13
GPIO - General Purpose (Digital) Input and Output . . . 14
UART - Universal Asynchronous Receiver and Transceiver . . . 17
SPI - Serial Peripheral Interface . . . 18
I^2C . . . 19

Basic Digital Sensors . . . 22
Read This Before Trying the Examples . . . 22
Buttons . . . 22
Reed Switch . . . 23
Keypads . . . 23
Vibration Sensor (SW-420) . . . 24
Presence Sensor (PIR) . . . 24
Flame Sensor . . . 25
Digital Sound Sensor . . . 26
MQ Gas Sensors . . . 27
Digital Hall Effect Sensors . . . 27

Analog Sensors . . . 29
Potentiometers . . . 29
Analog Joysticks . . . 30
Light-Dependent Resistors (LDR) . . . 30
Phototransistor . . . 31
Using LEDs as a Light Sensor . . . 32
Gas Sensor . . . 33
Analog Hall Effect Sensor . . . 33

Temperature Sensors . . . 34
Thermistor . . . 34
LM35D and TMP36 . . . 35
DS18B20 . . . 43
DHT11 and DHT22 . . . 44
LM75A . . . 44
HDC1080 . . . 45
MCP9808 . . . 46
AHT10 . . . 47
Sensors Comparison Table . . . 48

Atmospheric Pressure Sensors . . . 49
BMP085 and BMP180 . . . 50

BMP280 and BME280 . . . 54
BMP390 . . . 54
Sensors Comparison Table . . . 55

Electronic Compass, Accelerometers, and Gyroscopes . . . 56
HMC5883L, HMC5983, and QMC5883L 3-Axis Magnetic Sensor . . . 56
ADXL345 3-Axis Accelerometer . . . 57
MMA8452 3-Axis Accelerometer . . . 58
MPU6050 3-Axis Accelerometer and Gyroscope . . . 58
Sensors Comparison Table . . . 59

Miscellaneous Sensors . . . 60
HC-SR04 Ultrasonic Sensor . . . 60
Rotary Encoder . . . 61
Load Cell (Strain Gauge) . . . 61
iButton . . . 62

Fingerprint Sensors . . . 64
Fingerprints Basics . . . 64
FPM10A Sensor Protocol . . . 64
Sensor Configuration . . . 64
Enrolling a Fingerprint . . . 64
Identifying a Fingerprint . . . 64
Transferring the Image and the Template . . . 65
Fingerprint Sensor Example . . . 65

RFID . . . 66
RFID 125kHz . . . 66
MIFARE and NFC . . . 67

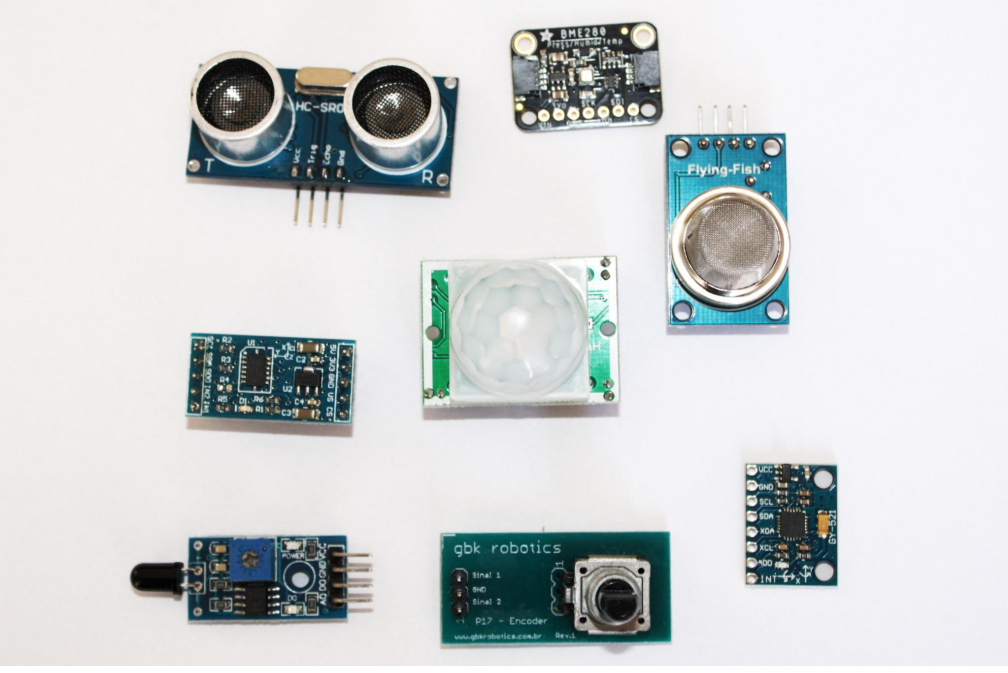
Conclusion . . . 69
Choosing Sensors and Using Them Correctly . . . 69
Using a New Sensor . . . 69
Writing Code and Using Libraries . . . 69
What's Next? . . . 69

Appendix A - Other Boards Based on the RP2040 Microcontroller . . . 70
Adafruit Feather RP2040 . . . 70
SeeedStudio XIAO RP2040 . . . 70
Arduino Nano RP2040 Connect . . . 70
Raspberry Pi Pico W . . . 70

Appendix B - Non-Original Sensors **71**
Non-original DS18B20 Temperature Sensors 71
Conterfeit MPU6050 Accelerometer 71

Introduction

Sensors allow an embedded system to get a glimpse of what is happening in the physical world. In a typical system, data from sensors are processed in a microcontroller and used to decide what actions should be taken.



A Few Sensors

In this book, you are going to learn about sensors in general and the many ways they can send data to a microcontroller. You will also see some of the typical strategies to get reliable data in a timely way. On a more specific view, you will learn about the resources available in the **Raspberry Pi Pico** board to connect sensors and how to use them when programming in **C/C++**, **MicroPython**, and **CircuitPython**. And then we are going to look deeply into quite a few popular sensors.

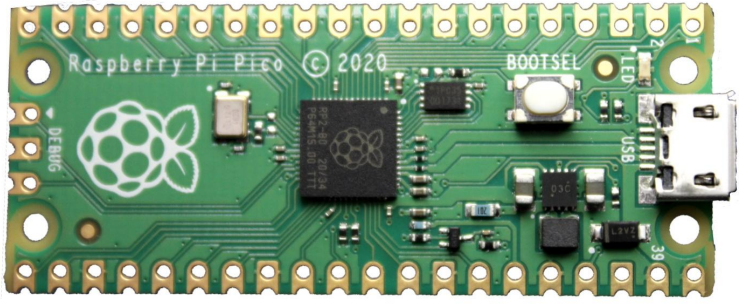
I tried to cover a good number of the more common sensors. Of course, new sensors enter the market daily (and some old ones get harder to find). That is one of the reasons I explain in detail

the workings of the sensors, with this knowledge you will have a head start when using a new sensor.

I will talk about how to use the Pi Pico interfaces in your code and you are going to see many code examples, but teaching programming languages is **not** the object of this book. You should have a basic knowledge of C or Python to better understand the example code and be able to write your own programs.

The Raspberry Pi Pico

The **Raspberry Pi Pico** was selected for this book for many reasons: it is an easily found, cheap, and powerful board. By powerful I mean not only that it has a good number of interfaces for connecting sensors and some serious processing power, but also that this processing power and the memory available allows us to use a sophisticated language such as **MicroPython** and **CircuitPython**.



The Raspberry Pi Pico

The Pi Pico uses the RP2040 microcontroller. It has two ARM Cortex M0+ cores, 264K of RAM, and a good number of peripherals (GPIO, Timers, PWM, UART, I2C, SPI, ADC, and USB).

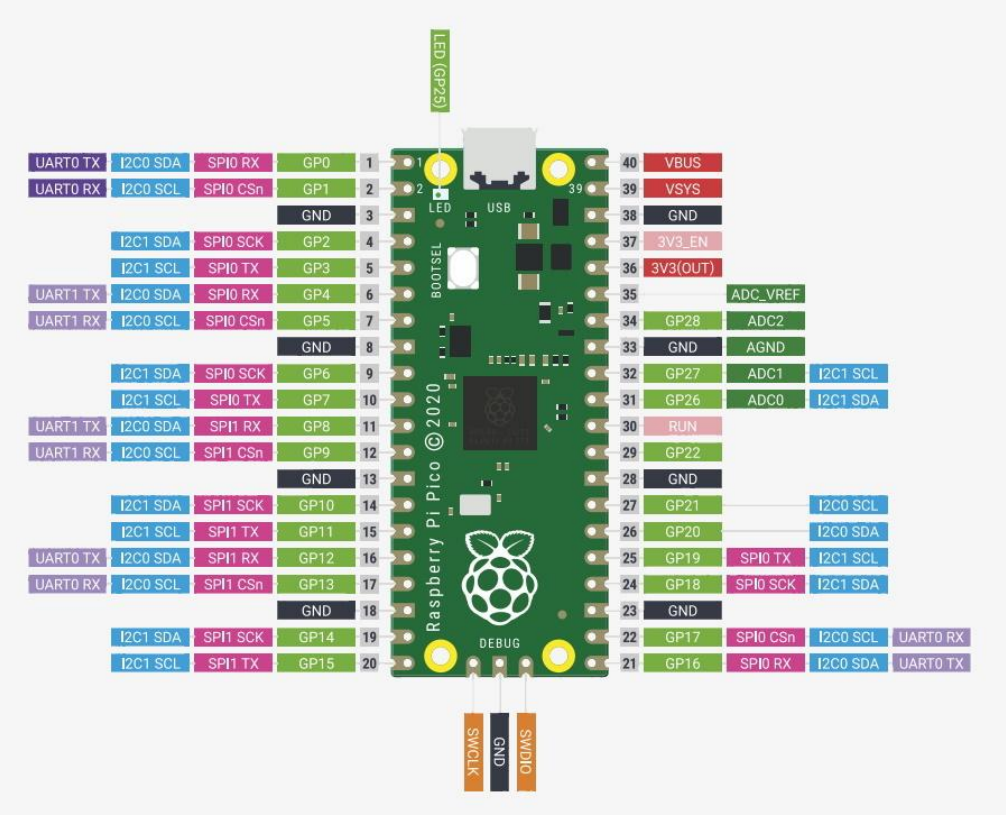
One important feature of the RP2040 is the **PIO** (*Programmable I/O*), which allows simple hardware protocols to be implemented as short code routines that run parallel to the main ARM cores. I will not go into details of how to program the PIO in this book. If you are interested, take a look at my book Knowing the RP2040[1].

While the figures and text refer to the Pi Pico, most of the information and techniques presented can be used with other boards that use the RP2040 microcontroller (and there are many of them, take a look at some in Appendix A).

[1] https://leanpub.com/rp2040

C/C++ has long established itself as the language of choice for low-end embedded systems, giving performance and predictability (at the cost of writing more lines of code). **MicroPython** brings to microcontrollers the ease of use and power of the Python language. Both have direct support from the people behind the Pi Pico. **CircuitPython** is a "fork" (a derivation) of MicroPython. While it does not have direct support from the Raspberry Foundation for the Pi Pico, it has I/O modules that make it easier to move code between different boards and a huge amount of drivers and sample code from Adafruit.

The image below (taken from the official documentation) shows the Raspberry Pi Pico and the signals at its pins. You will notice that some functions (like UART0 TX) appear in multiple pins (1, 16, and 21 in this case). As part of initializing some of the peripherals, we will select the pin that will be used.



Raspberry Pi Pico Pinout

The BOOTSEL button places the Pico in a special mode if you power (or reset) the Pico with it

pressed.

For the examples in this book, we are going to connect the Pico to a PC through USB. This connection will be used to:

- Power the board (and the sensors connected to it)
- Load programs in the board
- Interact with the program running on the Pico

We will talk more about all this in Chapter 3.

Getting the Example Code

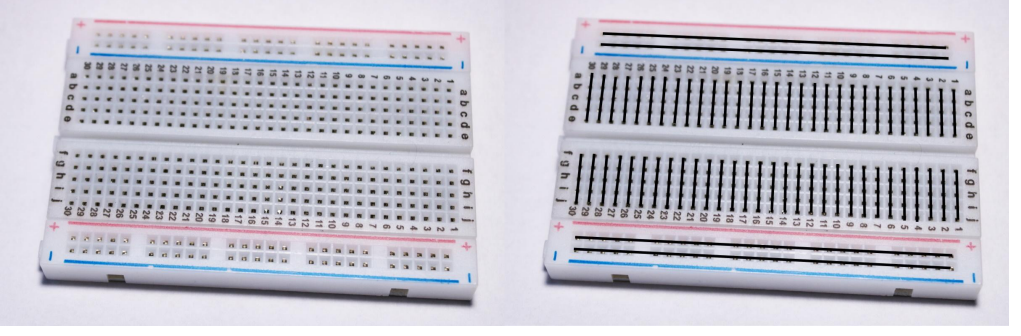
In this book, I will list only the more important parts of the code. The full code can be downloaded from

https://github.com/dquadros/PicoSensors

Using the Examples

The easiest way to play with the examples is by assembling the circuits on a breadboard.

A breadboard (or protoboard) is a board with holes where components can be inserted and internal connections between these holes. Wires (*jumpers*) are used to make the connections that cannot be made with the internal ones.



Breadboard

Once you understand what holes are connected internally, it's easy to change the position of the components.

To use the Pico and the sensors with a breadboard, they need to have pin connectors (*headers*) soldered to their pins. In some cases, instead of inserting a sensor directly into the breadboard, it's better to use male-to-female jumpers.

I have included pictures (made with Fritzing[2]) showing the connections for all the examples.

TIP: There is no reset button on the Pico, but you can connect one between pin 30 (RUN) and GND. Some small buttons have the exact spacing between pins 30 and 28, you can use one in a breadboard or solder it directly to the board.

Organization of This Book

This introduction (Chapter 1) gives a general idea of what are sensors and what you will find in this book.

Chapter 2 talks (in a general way) about why sensors will sometimes not give the right value and how to get better results from them.

Chapter 3 is about how we can write programs for the Pi Pico. We are going to use four options in this book: the official C/C++ SDK, the Arduino IDE, MicroPython, and CircuitPython.

In Chapter 4 we see the various ways a sensor can send data to a microcontroller and how they are accessed in the four programming environments.

The next chapters will talk about specific sensors. We can organize sensors in many ways, including how they are connected and the property they measure. I have adopted a mixed way:

Chapter 5 describes sensors that have a single digital output to give a yes/no or on/off result. A good example is a button that will signal if it is pressed or released.

Chapter 6 is about sensors that output an analog signal, like a light detector that informs how light or dark it is.

In Chapter 7 we change to the "what is measured" classification and talk about temperature sensors.

Chapter 8 is about pressure sensors and Chapter 9 is about accelerometers and compasses.

In Chapter 10 I talk about a few other sensors that did not fit in the previous chapters.

Chapter 11 is about fingerprint sensors and in Chapter 12 you will learn about two types of RFID.

Chapter 13 concludes the text with a quick recap and some reminders about using sensors in a project.

Appendix A shows a few other boards based on the RP2040 microprocessor.

[2] https://fritzing.org/

Appendix B talks about a problem I hope you don't have to face: non-original sensors (clones and fakes). There I share a couple of bad experiences I had while testing the examples for this book.

The book is now finished, but there may be updates to fix errors.

Acknowledgments

Looking back, there are too many people that, one way or another, have helped me come to the point where I could write this book. This is where I mention and thank a few of them.

First, my mother and father nurtured my curiosity and addicted me to reading.

There were many teachers who not only gave important lessons but also encouraged me to learn more.

In my professional life, I am grateful for all that believed I could deliver and those that helped me do so.

A special mention to the late Alberto Fabiano, who introduced me to the wild community of hackerspaces. Fabio Souza and Tiago Lima at Embarcados were very supportive of my writing. Also Mauricio Aniche, my son-in-law was a big encourager.

Of course, this book would not exist if not for the patience of my wife Cecilia while I spent days in front of a PC and playing with all those "little boards".

How to Send Feedback

If you find an error or have a suggestion or comment, you can reach me by:

- Clicking on the "Email the Author" button on the book page in leanpub.com
- Sending an email to dqsoft.blogspot@gmail.com
- Sending me a message on Twitter (@DQSoft)

Using Sensors

Many years ago, I got this new magnetic sensor that should return its position regards the magnetic North (like a compass). I studied the documentation and wrote some code to get a reading. The sensor returned a number and, again following the docs, I converted this number into an angle. I made some crude tests and the measurement changed when I rotated the sensor. All fine, right?

Sometime later I decided to build a compass using this sensor, showing the North direction with a 12 LED ring. To my dismay, I could not get it to work accurately. The reading changed a lot with the sensor in a fixed position, there was no linear relation between readings and the rotation and nearby objects would change the results!

Getting good results from sensors require a lot more than simply getting a number from them. In this chapter, I will talk about some of the challenges and some ways to get through them.

Measuring What We Really Want

Inside the RP2040 there is a temperature sensor. With a few lines of code, we can get a reading. But what are we measuring? The sensor is inside the chip so it will give a better idea of the internal temperature than of the ambient temperature. Indeed, most of the time you will get a temperature high above the ambient temperature. The reading will increase when the ARM cores are processing.

Sensor placement is fundamental to measuring what you want. In many cases, the property you want to measure varies with the position. It is up to you to figure out what is the important value for your application. Some testing may be needed to learn how it varies and what is the best placement. In some cases, you may need to use multiple sensors and use an average or have a more complicated logic using all the values,

A Sensor Can Affect What You Are Measuring

While this is not that common, you should look out for cases where the sensor can affect the property you are measuring. One example is when you are measuring electrical properties, like current and voltage. In many cases the sensor is wired in series or parallel with the circuit and its resistance and capacitance may affect what you are measuring.

Accuracy and Resolution

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

ADC Errors

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Averaging Readings

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Calibration

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Readings Validation

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Programming the Raspberry Pi Pico

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Libraries

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

The Official C/C++ SDK

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

The Arduino IDE

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Thonny IDE

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Setting Up Thonny

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Accessing the REPL

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Entering, Editing, and Saving Programs

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Running a Program

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Moving Files Between the Pi Pico and the PC

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Seeing Program Output

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython and CircuitPython

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Which Environment Should I Use?

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Interfaces and Protocols

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

A Little Bit of Electronics

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

A General View of Interfaces

First, we can divide the ways to connect a sensor into two large groups: analog (signals can assume any value in a range) and digital (signals can only assume a HIGH or LOW value).

Analog interface is very straightforward: the sensor outputs an analog signal (normally a voltage) that can be converted to the measurement. We use the Analog to Digital Converter (**ADC**) of the RP2040 to convert this signal into a number.

Digital interface can be a lot more complicated, involving one or more digital signals.

A simple sensor, like a switch, can output a single digital signal. More complicated sensors may need to receive and transmit multiple words of commands and data. In this case, it's common to use some kind of serial communication. In a **serial interface**, words are transferred one bit at a time through a single wire. While this is more complex than sending all bits at the same time through individual signals (a *parallel interface*), it requires fewer pins of the microcontroller and fewer connections to the sensor. The standard serial interfaces we are going to see in this chapter are the asynchronous (**UART**), **SPI**, and **I²C**. All three have hardware support in the RP2040.

The standard serial interfaces will transfer words (normally 8-bit bytes) between the microcontroller and the sensor. What these words mean varies from one sensor to another. In some cases, there is some kind of *protocol* - a precise sequence of commands and responses that must be followed. Complex sensors may use many words of configuration, status, and readings. A common strategy in these cases is to organize these words as a collection of *registers*, each with an individual address (like a small memory).

Some sensors use some special digital signaling. This is normally implemented by manually controlling and checking the digital signals. The RP2040 has a feature called **PIO** (*Programmable*

I/O) that allows us to defer this to a part of the hardware that will run concurrently with the ARM CPUs.

The ADC - Analog to Digital Converter

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

The Internal Temperature Sensor

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Using the ADC with the C/C++ SDK

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Using the ADC in the Arduino Environment

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Using the ADC in MicroPython

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Using the ADC in CircuitPython

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Connecting Digital Signals from Sensors to the Pico

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

GPIO - General Purpose (Digital) Input and Output

GPIO is the basic form of digital input and output. In digital input, the voltage in a pin is read as 0 or 1. In digital output, you write a 0 or 1 to set the voltage in a pin to a LOW or HIGH level.

By default, the I/O pins in the Raspberry Pi Pico are set to GPIO input. They need to be configured to be used for other functions (like ADC or SPI). Depending on the programming environment, this can be done automatically or you may need to do it explicitly.

If you use digital input in an unconnected input, the result is undetermined. You can change this by enabling the *pull-up* or *pull-down* resistors. These are resistors (with values somewhere between 50k and 80k Ohms) internal to the microcontroller that can connect a pin to the ground (pull-down) or the power source (pull-up).

> *The GPIO in the Pico has many features we will not talk about here. We will stick with the basics you will need to interface with sensors.*

Using GPIO with the C/C++ SDK

The SDK functions numbers the GPIO pins from 0 to 29. Some functions use a *mask*, a 32-bit value where bit `n` is associated with pin `n`.

The functions below are in the library `hardware_gpio`.

```
void gpio_init (uint gpio)
```

Initializes a pin for GPIO use. The pin is configured for input.

```
void gpio_init_mask (uint gpio_mask)
```

Initializes the pins select by `mask` for GPIO use. The pins are configured for input.

```
static void gpio_set_dir (uint gpio, bool out)
```

Sets the direction (`out` = true for output, false for input) of a GPIO pin.

```
static void gpio_set_dir_masked (uint32_t mask, uint32_t value)
```

Sets the direction of the pins selected by `mask`. Each bit in `value` correspond to a pin (0 for input, 1 for output).

```
static void gpio_pull_up (uint gpio)
```

Connects the pull-up resistor.

```
static void gpio_pull_down (uint gpio)
```

Connects the pull-down resistor.

`void gpio_set_pulls (uint gpio, bool up, bool down)`

Control both pull resistors, true means connect, false disconnect.

`static bool gpio_get (uint gpio)`

Get the current state of a pin (0 for low, 1 for high).

`static uint32_t gpio_get_all (void)`

Get the current state of all pins. Each bit in the result corresponds to a pin (0 for low, 1 for high).

`static void gpio_put (uint gpio, bool value)`

Changes the state (`value` = true for high, false for low) of a GPIO pin.

`static void gpio_put_masked (uint32_t mask, uint32_t value)`

Changes the state of the pins selected by `mask`. Each bit in `value` correspond to a pin (0 for low, 1 for high).

Using GPIO the Arduino Environment

Arduino numbers the digital pins from 0 to 29, using the same numbers as the C/C++ SDK.

There are three functions related to basic digital I/O in Arduino:

`pinMode(int pin, int mode)`

Sets the mode of a pin to `INPUT`, `OUTPUT`, or `INPUT_PULLUP`.

`int digitalRead(int pin)`

Reads a pin, returning HIGH or LOW.

`digitalWrite(int pin, int value)`

Sets the level of a pin, the `value` should be HIGH or LOW.

Another function that is useful in Arduino is

`unsigned long pulseIn(int pin, int value, unsigned long timeout)`

This function will measure a pulse on a pin. `pin` is the pin number, `value` is the value (HIGH or LOW) during the pulse, and `timeout` is the maximum time (in microseconds) to wait for the measurement to complete. `timeout` can be omitted (the function assumes 1 second).

This function works by:

- Waiting for the pin to be at the "not value" level. If this is not detected in the timeout, 0 is returned

- Waiting for the pin to change from “not value” to “value”. If this change is not detected in the timeout, 0 is returned
- Starting the timing
- Wait for the pin to change from “value” to “not value”. If this change is not detected in the timeout, 0 is returned
- Return the time the pin stayed in the “value” level.

Using GPIO in MicroPython

MicroPython numbers the digital pins from 0 to 29, using the same numbers as the C/C++ SDK. Digital pins are accessed through the class `Pin` in the `machine` module.

```
from machine import Pin

p0 = Pin(0, Pin.OUT)    # create output pin on GPIO0
p0.on()                 # set pin to "on" (high) level
p0.off()                # set pin to "off" (low) level
p0.value(1)             # set pin to on/high

p2 = Pin(2, Pin.IN)     # create input pin on GPIO2
print(p2.value())       # get value, 0 or 1

p4 = Pin(4, Pin.IN, Pin.PULL_UP) # enable internal pull-up resistor
p5 = Pin(5, Pin.OUT, value=1) # set pin high on creation
```

Using GPIO in CircuitPython

In CircuitPython digital pins are referred to by names defined in the `board` object. For the Pi Pico, they are `board.GP0` through `board.GP29`. Digital pins are accessed through the class `DigitalInOut` in the `digitalio` module.

```
import digitalio
import board

 # Read pin GP3
pin3 = digitalio.DigitalInOut(board.GP3)
pin3.pull = digitalio.Pull.UP
print(pin3.value)

 # Change level of GP10 to HIGH
pin10 = digitalio.DigitalInOut(board.GP10)
pin10.direction = digitalio.Direction.OUTPUT
pin10.value = True
```

UART - Universal Asynchronous Receiver and Transceiver

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Framing

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Control Signals and Hardware Flow Control

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Pins Options

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Using the UART with the C/C++ SDK

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Using the UART in the Arduino Environment

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Using the UART in MicroPython

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Using the UART in CircuitPython

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

SPI - Serial Peripheral Interface

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

SPI Signals

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

SPI Modes

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

SPI in the Raspberry Pi Pico

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Pins Options

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Using the SPI with the C/C++ SDK

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Using the SPI in the Arduino Environment

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Using the SPI in MicroPython

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Using the SPI in CircuitPython

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

I^2C

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

I^2C Topology

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

I^2C Electrical Interface

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Read and Write Operations in I^2C

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Start and Stop Conditions

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Read Operation

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Write Operation

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Combined Write/Read Operation

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

I²C in the Raspberry Pi Pico

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Using I²C With the C/C++ SDK

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Using I²C in the Arduino Environment

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Using I²C in MicroPython

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Using I²C in CircuitPython

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Finding the I²C Address of a Device

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Basic Digital Sensors

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Read This Before Trying the Examples

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Buttons

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Button Use Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Reed Switch

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Keypads

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Keypad Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Vibration Sensor (SW-420)

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Vibration Sensor Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Presence Sensor (PIR)

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Presence (PIR) Sensor Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Flame Sensor

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Flame Sensor Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Digital Sound Sensor

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Digital Sound Sensor Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MQ Gas Sensors

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MQ Gas Sensor Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Digital Hall Effect Sensors

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

KY-003 Module

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

US1881 Hall Effect Sensor

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Hall Effect Sensor Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Analog Sensors

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Potentiometers

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Potentiometer Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Analog Joysticks

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Analog Joystick Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Light-Dependent Resistors (LDR)

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

LDR Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Phototransistor

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Phototransistor Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Using LEDs as a Light Sensor

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

LED as a Light Sensor Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Gas Sensor

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Analog Gas Sensor Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Analog Hall Effect Sensor

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Analog Hall Effect Sensor Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Temperature Sensors

In this chapter, we are looking at sensors that will measure **temperature**. The sensors examined have very different interfaces and performance characteristics. *There is no perfect sensor, you have to balance precision, ease of use, and price for your particular application.*

> Note: In the next chapter, we will look at barometric pressure sensors that will also measure ambient temperature.

As we saw in Chapter 2, you have to figure out where to position a temperature sensor so you can measure what you want.

The sensors in the examples will return temperature in Celsius degrees. In case you don't remember, you can convert from Celsius to Fahrenheit with this simple formula:

$$T_F = 9 * \frac{T_C}{5} + 32$$

Some of the sensors we will examine also measure *Relative Humidity*. This indicates a present state of absolute humidity relative to a maximum humidity given the same temperature and is expressed as a percentage (0 to 100%).

Thermistor

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Thermistor Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

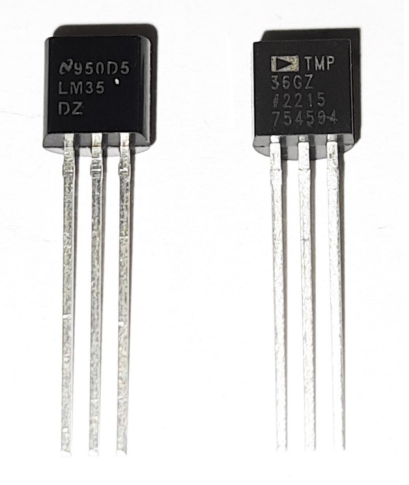
This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

LM35D and TMP36

These temperature sensors use a very direct approach: they output a voltage that is linearly proportional to the temperature, at the rate of 10mV per Celsius degree.



LM35 (left) and TMP36 (right) Sensors

The LM35D works from 0 to 100°C and needs a 4V to 20V power supply. Its output is the temperature (in Celsius) multiplied by 0.01V. For example, an output of 0,23V corresponds to a temperature of 23°C. The sensor is designed for a ±2°C accuracy, the typical value is ±0.9°C.

$$T = V/0.01$$

There are other models in the LM35 line, the plain LM35, LM35A, LM35C, and LM35CA can measure negative temperatures but require additional components for that (you will find the circuits in the datasheet). These other models also have better precision.

The TMP36 works from –40°C to +125°C and needs a 2.7V to 5.5V power supply. It outputs 0.75V at 25°C, so the temperature can be calculated from the output voltage by the following formula:

$$T = 25 + (V - 0.75)/0.01$$

The TMP36 is available in two grades ("F" and "G") based on the accuracy:

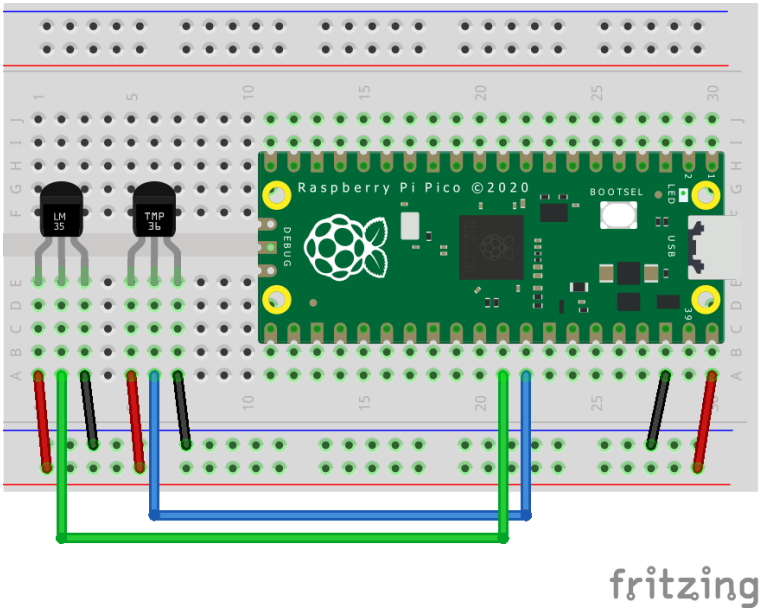
Parameter[1]	Symbol	Test Conditions/Comments	Min	Typ	Max	Unit
ACCURACY						
TMP35/TMP36/TMP37 (F Grade)		T_A = 25°C		±1	±2	°C
TMP35/TMP36/TMP37 (G Grade)		T_A = 25°C		±1	±3	°C
TMP35/TMP36/TMP37 (F Grade)		Over rated temperature		±2	±3	°C
TMP35/TMP36/TMP37 (G Grade)		Over rated temperature		±2	±4	°C

TMP76 Accuracy

These specifications were taken from the datasheets from National Semiconductor / Texas Instrument (LM35) and Analog Devices (TMP36)

LM35D and TMP36 Example

In this example, I am using both sensors. If you want, you can mount only one and change the pin corresponding to the other one to -1 in the code.



Connecting LM35 and TMP36 Sensors to the Pi Pico

Notice that, while I am powering the sensors with +5V, their output will not exceed 1V (LM35 @ 100°C) and 2V (TMP36 @ 125°C) and can be connected directly to the Pico's ADC.

The code is pretty straightforward. For better accuracy, an average of 10 readings is used. The output voltage is calculated from the ADC reading (V = adc*3.3/(max+1)) and then converted to the temperature by the previous equations. A key point is that the max adc reading varies with the programming environment (1023 for the Arduino, 4095 for the SDK, and 65535 for MicroPython and CircuitPython).

C/C++ SDK Code

C/C++ LM35D and TMP36 Example

```c
/**
 * @file lm35_tmp36_sdk.c
 * @author Daniel Quadros (dqsoft.blogspot@gmail.com)
 * @brief LM35 and TMP36 Sensors Example
 * @version 1.0
 * @date 2023-05-18
 *
 * @copyright Copyright (c) 2023, Daniel Quadros
 *
 */

#include <stdio.h>
#include <stdlib.h>
#include "pico/stdlib.h"
#include "hardware/adc.h"

// Pins  (-1 if not used)
#define LM35_PIN 26
#define TMP36_PIN 27

// returns the voltage in an ADC pin
// averages 10 readings
#define N_READINGS 10
float readSensor(int pin) {
  if (pin == -1) {
    return 0;
  }

  // pin 26 -> channel 0, pin 27 -> channel 1
  // pin 28 -> channel 2, pin 29 -> channel 3
  adc_select_input(pin-26);

  uint32_t sum = 0;
  for (int i = 0; i < N_READINGS; i++) {
    sum += adc_read();
  }
  return (sum*3.3 / N_READINGS)/4096.0;;
}
```

```
39
40
41  // Main Program
42  int main() {
43    // Init stdio
44    stdio_init_all();
45    #ifdef LIB_PICO_STDIO_USB
46    while (!stdio_usb_connected()) {
47        sleep_ms(100);
48    }
49    #endif
50
51    // Init sensor ADCs
52    adc_init();
53    if (LM35_PIN != -1) {
54      adc_gpio_init(LM35_PIN);
55    }
56    if (TMP36_PIN != -1) {
57      adc_gpio_init(TMP36_PIN);
58    }
59
60    // Main loop
61    while(1) {
62      float tempLM35 = readSensor(LM35_PIN)/0.01;
63      float tempTMP36 = 25.0 + (readSensor(TMP36_PIN)-0.75)/0.01;
64      printf("LM35 = %.2fC  TMP36 = %.2fC\n", tempLM35, tempTMP36);
65      sleep_ms(2000);
66    }
67  }
```

Arduino Code

Arduino LM35D and TMP36 Example

```
// LM35D and TMP36 Temperature Sensor Example

// Pins (-1 if not used)
#define LM35_PIN A0
#define TMP36_PIN A1

//returns the voltage in an ADC pin
// averages 10 readings
#define N_READINGS 10
float readSensor(int pin) {
  uint32_t sum = 0;

  if (pin == -1) {
    return 0;
  }

  for (int i = 0; i < N_READINGS; i++) {
    sum += analogRead(pin);
  }
  return (sum*3.3 / N_READINGS)/1024.0;
}

// initialization
void setup() {
  Serial.begin(115200);
}

// Main loop
void loop() {
  float tempLM35 = readSensor(LM35_PIN)/0.01;
  float tempTMP36 = 25.0 + (readSensor(TMP36_PIN)-0.75)/0.01;
  Serial.print("LM35 = ");
  Serial.print(tempLM35,1);
  Serial.print("C  TMP36 = ");
  Serial.print(tempTMP36,1);
  Serial.println("C");
  delay(2000);
}
```

MicroPython Code

MicroPython LM35D and TMP36 Example

```
# LM35D and TMP36 Temperature Sensor Example
from machine import ADC
from time import sleep

# Connections (-1 if not used)
pinLM35 = 26
pinTMP36 = 27

if pinLM35 != -1:
    adcLM35 = ADC(pinLM35)
if pinTMP36 != -1:
    adcTMP36 = ADC(pinTMP36)

# returns the voltage in an ADC pin
# averages 10 readings
def readADC(adc):
    sum = 0
    for i in range(10):
        sum = sum+adc.read_u16()
    return (sum*0.33)/65536

# Main Loop
while True:
    tempLM35 = 0
    if pinLM35 != -1:
        vLM35 = readADC(adcLM35)
        tempLM35 = vLM35/0.01
    tempTMP36 = 0
    if pinTMP36 != -1:
        vTMP36 = readADC(adcTMP36)
        tempTMP36 = 25.0 + (vTMP36-0.75)/0.01
    print("LM35 = {:.1f}C  TMP36 = {:.1f}C".format(tempLM35, tempTMP36))
    sleep(2)
```

CircuitPython Code

CircuitPython LM35D and TMP36 Example

```
# LM35D and TMP36 Temperature Sensor Example
import analogio
import board
from time import sleep

# Connections (-1 if not used)
pinLM35 = board.A0
pinTMP36 = board.A1

if pinLM35 != -1:
    adcLM35 = analogio.AnalogIn(pinLM35)
if pinTMP36 != -1:
    adcTMP36 = analogio.AnalogIn(pinTMP36)

# returns the voltage in an ADC pin
# averages 10 readings
def readADC(adc):
    sum = 0
    for i in range(10):
        sum = sum+adc.value
    return (sum*0.33)/65536

# Main Loop
while True:
    tempLM35 = 0
    if pinLM35 != -1:
        vLM35 = readADC(adcLM35)
        tempLM35 = vLM35/0.01
    tempTMP36 = 0
    if pinTMP36 != -1:
        vTMP36 = readADC(adcTMP36)
        tempTMP36 = 25.0 + (vTMP36-0.75)/0.01
    print("LM35 = {:.1f}C  TMP36 = {:.1f}C".format(tempLM35, tempTMP36))
    sleep(2)
```

DS18B20

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

1-Wire Protocol

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

ROM Commands

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

DS18B20 Function Commands

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

DS18B20 Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

DHT11 and DHT22

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

DHT11 and DHT22 Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

LM75A

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

LM75A Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

HDC1080

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

HDC1080 Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MCP9808

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Alert Function

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MCP9808 Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

AHT10

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

AHT10 Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Sensors Comparison Table

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Atmospheric Pressure Sensors

Atmospheric pressure (also called *barometric pressure*) is the pressure from the Earth's atmosphere. In most circumstances, it's closely approximated by the pressure from the weight of the air above the measurement point. Atmospheric pressure decreases with increasing elevation. At low altitudes above sea level, the pressure decreases by about 1.2 kPa[3] (12 hPa) for every 100 meters elevation.

Atmospheric pressure is also an indicator of weather. When a low-pressure system moves into an area, it usually leads to cloudiness, wind, and precipitation. High-pressure systems usually lead to fair, calm weather.

So, there are two uses for a pressure reading: finding the altitude and (crudely) predicting the weather.

For finding the altitude, given the measure pressure p and that the pressure at sea level, p_0 is 1013.25 hPa, we can use the simple formula (for low altitudes)

$$altitude = 100 * (p - p_0)/12$$

or the more precise

$$altitude = 44330 * (1 - (\frac{p}{p_0})^{\frac{1}{5.255}})$$

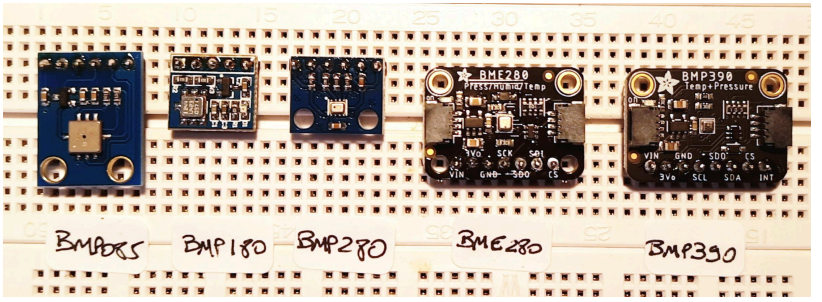
These formulas don't take into account variations in pressure from the weather.

For a simple weather forecast, if we know the altitude, we can compare the measured pressure with the expected pressure based on the altitude. We should have good weather if the measured pressure is 2.5hPa or more above the expected pressure. We should have bad weather if it's 2.5hPa or move below.

In a practical situation, if we know the sensor is not changing altitude, we can record the pressure over time and use the average to find the altitude and use the changes from it to predict changes in the weather.

[3] It is also possible to construct a bridge with one or two sensors and fixed resistors, but the result is not as good.

The atmospheric pressure sensors we are going to look at in this chapter are all from Bosch Sensortec GmbH. Some of the older models have been discontinued, but still can be found. All measure pressure and temperature; the BME280 model adds Humidity.



A Few Bosch Atmospheric Pressure Sensors Modules

In the boards above, the sensors are the small metal-encased components in the center.

Each sensor is factory calibrated, the calibration coefficients are saved in an EEProm in the sensor. The datasheets provide the formulas to calculate the temperature and pressure using these coefficients.

All the sensors can operate from -40 to +85°C but only have full accuracy from 0 to +65°C.

These components are pretty small and are available only as surface-mounted devices (SMD), so if you are a hobbyist (or just want to make some tests) you will probably use a module (like the above) with it. Pay attention to what other components are in the module, typically you will find:

- Voltage regulator and converters to allow connection to a 5V device.
- Pull-up or pull-down resistors to select the interface (I^2C or SPI) or address.
- I^2C bus pull-ups.

The examples will send to a PC the readings from the sensors. You will need to connect the Pico to a PC to see the results.

In the C/C++ SDK examples, I will write the full code. For the Arduino, MicroPython, and CircuitPython examples, I will use already available libraries.

BMP085 and BMP180

The **BMP085** is the older of the sensors in this chapter. The **BMP180** is function compatible with it, but it is physically smaller and consumes less energy. Both are discontinued (but still easily found).

They have an I²C interface, supporting standard, fast, and high-speed modes (clock up to 3.4Mbits/s). The address is fixed at **0x77**.

There are eleven signed 16-bit coefficients. They are read by issuing an I²C write transaction with the 8-bit register address (0xAA to 0xBE) followed by an I²C read transaction to get the two bytes (most significant first). Since they don't change, you can read them at the start of your program and save them for later use.

Param	MSB	LSB
AC1	0xAA	0xAB
AC2	0xAC	0xAD
AC3	0xAE	0xAF
AC4	0xB0	0xB1
AC5	0xB2	0xB3
AC6	0xB4	0xB5
B1	0xB6	0xB7
B2	0xB8	0xB9
MB	0xBA	0xBB
MC	0xBC	0xBD
MD	0xBE	0xBF

BMP085/BMP180 Calibration Coefficients

You can do independent pressure and temperature readings. To do a temperature reading, you write a command (0x2E) to the control register (address 0xF4), wait 4.5 ms, and then read the unsigned 16-bit raw data from register 0xF6. The raw data is converted to the temperature by the formula in the datasheet (you can also see it in the C/C++ SDK example).

Reading pressure is similar, but the command written to the control register (0x34) allows an *oversampling* mode in the upper two bits:

Bits	Mode	Samples	Time	Current
00	ultra low power	1	4.5 ms	3 µA
01	standard	2	7.5 ms	5 µA
10	high resolution	4	13.5 ms	7 µA
11	ultra high resolution	8	25.5 ms	12 µA

Oversampling works by doing multiple readings to get more bits of precision. As you can see in the table, the time for the measurement (and the power consumption) depends on the mode.

The raw pressure data has 19 to 16 bits, depending on the oversampling mode. You must read three bytes starting at register 0xF6 (most significant first), shift it right 3, 2, 1, or 0 bits, depending on the mode, and use the resulting value in the pressure formula (see C/C++ SDK example).

Name	Address	R/W	Content								Default
			7	6	5	4	3	2	1	0	
Calibration	0xAA - 0xBF	R									-
ID	0xD0	R	0	1	0	1	0	1	0	1	0x55
Reset	0xE0	W									0x00
ctrl_meas	0xF4	R/W	oss		sco	control					0x00
	0xF6	R	adc_out_msb								0x80
	0xF7	R	adc_out_lsb								0x00
	0xF8	R	adc_out_xlsb					0	0	0	0x00

BMP085/BMP180 Registers

Note that temperature and pressure data share the same registers, so you must start one conversion, wait, read the result, start the other conversion, wait, and read the result.

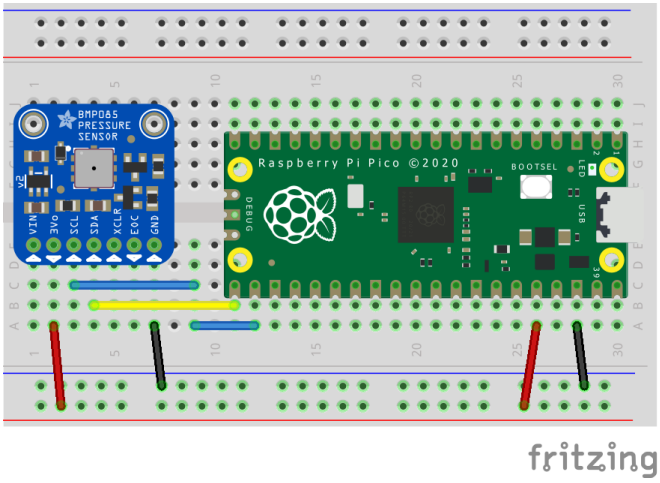
The ID register returns 0x55 for both models. Writing 0xB6 to the Reset register will reset the sensor to the power-on (default) condition.

The BMP085 has two additional connections to the microcontroller. The EOC output signals the end of a conversion and the XCLR input resets the sensor when at a LOW level.

BMP085 and BMP180 Example

We will ignore the EOC and XCLR pins in the BMP085 and just connect the power and the I²C signals.

Check the pinout of your breakout board!



Connecting a BMP085 (or BMP180) Sensor to the Pi Pico

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

BMP280 and BME280

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

BMP280 and BME280 Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

BMP390

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

BMP390 Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Sensors Comparison Table

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Electronic Compass, Accelerometers, and Gyroscopes

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

HMC5883L, HMC5983, and QMC5883L 3-Axis Magnetic Sensor

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

HMC5883L and HMC5983 Registers and Operation

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

QMC5883L Registers and Operation

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

HMC5883L, HMC5983, and QMC5883L Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

ADXL345 3-Axis Accelerometer

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

ADXL345 Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MMA8452 3-Axis Accelerometer

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MMA8452 Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MPU6050 3-Axis Accelerometer and Gyroscope

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MPU6050 Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Sensors Comparison Table

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Miscellaneous Sensors

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

HC-SR04 Ultrasonic Sensor

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

HY-SRF05

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

HC-SR04 Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Rotary Encoder

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Rotary Encoder Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Load Cell (Strain Gauge)

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

The HX711 ADC

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Load Cell Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

iButton

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

iButton Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Fingerprint Sensors

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Fingerprints Basics

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

FPM10A Sensor Protocol

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Sensor Configuration

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Enrolling a Fingerprint

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Identifying a Fingerprint

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Transferring the Image and the Template

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Fingerprint Sensor Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

C/C++ SDK Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

CircuitPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

RFID

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

RFID 125kHz

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

RFID 125kHz Example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MicroPython Code

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Code for SDK C/C++

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Code for the Arduino Environment

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Code for CircuitPython

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MIFARE and NFC

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Memory Addressing

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

The Configuration Block

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Ticketing

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MFRC522 Reader

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

PN532 Reader

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

MIFARE Examples

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Reading and Writing Data in MicroPython with the MFRC522 Reader

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Ticketing with MFRC522 Reader in the Arduino Environment

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Reading and Writing Data in CircuitPython with the PN532 Reader

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Conclusion

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Choosing Sensors and Using Them Correctly

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Using a New Sensor

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Writing Code and Using Libraries

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

What's Next?

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Appendix A - Other Boards Based on the RP2040 Microcontroller

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Adafruit Feather RP2040

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

SeeedStudio XIAO RP2040

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Arduino Nano RP2040 Connect

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Raspberry Pi Pico W

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Appendix B - Non-Original Sensors

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Non-original DS18B20 Temperature Sensors

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.

Conterfeit MPU6050 Accelerometer

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/picosensors.