Part I

Raspberry Pi Basics

In This Part

CHAPTER

1

Raspberry Pi Hardware

In this chapter, you are introduced to the Raspberry Pi (RPi) platform hardware. The chapter focuses on recently released Raspberry Pi models and describes the various subsystems and physical inputs/outputs of the boards. In addition, the chapter lists accessories that can prove helpful in developing your own Raspberry Pi–based projects. By the end of this chapter, you should have an appreciation of the power and complexity of this physical-computing platform. You should also be aware of the first steps to take to protect your board from physical damage.

Introduction to the Platform

The RPi models are capable general-purpose computing devices, and for that reason they have found favor for introducing learners to general computing and computer programming. The RPi models, some of which are illustrated in Figure 1-1, are also capable *physical computing* devices that can be used for embedded systems applications—and for Internet-attached embedded applications in particular.

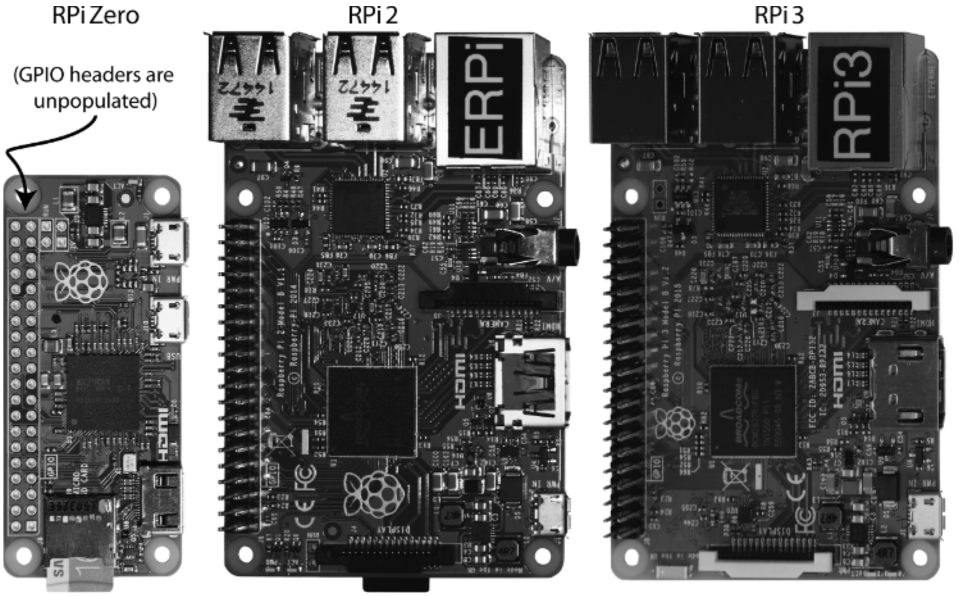


Figure 1-1: Raspberry Pi platform board examples (to relative scale)

Some general characteristics of RPi devices include the following:

- They are low cost, available for as little as $5–$35.
- They are powerful computing devices. For example, the RPi 3 contains a 1.2 GHz ARM Cortex-A53 processor that can perform more than 700 million Whetstone instructions per second (MWIPS).[1]
- They are available in a range of models that are suitable for different applications (e.g., the larger-format RPi 3 for prototyping and the tiny-format RPi Zero or Compute Module for deployment).
- They support many standard interfaces for electronic devices.
- They use little power, running at between approximately 0.5 W (RPi Zero when idle) and approximately 5.5 W (RPi 3 under load).
- They are expandable through the use of Hardware Attached on Top (HAT) daughter boards and USB devices.
- They are supported by a huge community of innovators and enthusiasts, who generously give of their time to help the RPi Foundation with their educational mission.

The RPi platform can run the Linux operating system, which means that you can use many open source software libraries and applications directly

[1] www.roylongbottom.org.uk/Raspberry%20Pi%20Benchmarks.htm

with it. Open source software driver availability also enables you to interface devices such as USB cameras, keyboards, and Wi-Fi adapters with your project, without having to source proprietary alternatives. Therefore, you have access to comprehensive libraries of code that have been built by a talented open source community; however, it is important to remember that the code typically comes without any type of warranty or guarantee. If there are problems, you have to rely on the good nature of the community to resolve them. Of course, you could also fix the problems yourself and make the solutions publicly available.

One impressive feature of recent RPi models is that their functionality can be extended with daughter boards, called *HATs* (*Hardware Attached on Top*), that connect to the GPIO header (the 40-pin double-pin connector row on the boards in Figure 1-1). You can design your own HATs and attach them securely to your RPi using this header. In addition, many HATs are available for purchase that can be used to expand the functionality of your RPi platform. Some examples of these are described toward the end of this chapter.

Who Should Use the RPi

Anybody who wants to transform an engineering concept into a real interactive electronics project, prototype, or work of art should consider using the RPi. That said, integrating high-level software and low-level electronics is not an easy task. However, the difficulty involved in an implementation depends on the level of sophistication that the project demands. The RPi community is working hard to ensure that the platform is accessible by everyone who is interested in integrating it into their projects, whether they are students, makers, artists, or hobbyists. For example, the availability of the Scratch visual programming tool on the RPi (`tiny.cc/erpi101`) is an excellent way to engage children with both computer programming and the RPi.

For more advanced users with electronics or computing knowledge, the RPi platform enables additional development and customization to meet specific project needs. Again, such customization is not trivial: You may be an electronics expert, but high-level software programming and/or the Linux operating system might cause you difficulty. Or you may be a programming guru but you have never wired an LED! This book aims to cater to all types of users who are interested in interfacing with the RPi, providing each type of reader with enough Linux, electronics, and software exposure to ensure that you can be productive, regardless of your previous experience level.

When to Use the RPi

The RPi is perfectly placed for the integration of high-level software and low-level electronics in any type of project. Whether you are planning to build an automated home management system, robot, multimedia display, Internet of

Things (IoT) application, vending machine, or Internet-connected work of interactive art, the RPi has the processing power to do whatever you can imagine of an embedded device.

The major advantage the RPi and other embedded Linux devices have over more traditional embedded systems, such as the Arduino, PIC, and AVR microcontrollers, is apparent when you leverage the Linux OS for your projects. For example, if you build a home automation system using the RPi and you then decide that you want to make certain information available on the Internet, you can simply install the Nginx web server. You could then use server-side scripting or your favorite programming language to interface with your home automation system to capture and share information. Alternatively, your project might require secure remote shell access. In that case, you could install a Secure Shell (SSH) server simply by using the Linux command `sudo apt install sshd` (as covered in Chapter 2). This could potentially save you weeks of development work. In addition, you have the comfort of knowing that the same software is running securely on millions of machines around the world.

Linux also provides you with device driver support for many USB peripherals and adapters, making it possible for you to connect cameras, Wi-Fi adapters, and other low-cost consumer peripherals directly to your platform without the need for complex/expensive software driver development.

The RPi is also an excellent device for playing high-definition video. The RPi has this capability because its Broadcom BCM2835/6/7 processor was designed for multimedia applications, and it has a hardware implementation of H.264/MPG-4 and MPG-2/VC-1 (via additional license) decoders and encoders. The RPi has found popular use for multimedia applications such as running the Kodi home media center[2] (`www.kodi.tv`) for playing full-HD video content.

When to Not Use the RPi

The Linux OS was not designed for real-time or predictable processing. This would be problematic if, for example, you want to sample a sensor precisely every one millionth of a second. If the precise time arises to take a sample and the kernel is busy with a different task, it cannot be easily interrupted. Therefore, in its default state, the RPi is not an ideal platform for real-time systems applications. Real-time versions of Linux are available, but they are currently targeted at very experienced Linux developers, and there are limits to their real-time capabilities. However, the RPi can be combined with real-time service processors, and the RPi can be used as the "central intelligence." You can interconnect such real-time microcontrollers to the RPi via electrical buses (e.g., I^2C, UART) and

[2] Formerly known as XBMC.

Ethernet, and have the RPi act as the central processor for a distributed control system. This concept is described in Chapters 11, 12, and 13.

The RPi platform is not ideal for project developments that are likely to be commercialized. The Raspberry Pi platform largely utilizes open source software (there are some closed-source blobs used with the GPU), but it is not open source hardware. Schematics are available for RPi boards (e.g., `tiny.cc/erpi102`), but there is a lack of documentation on the hardware used. In addition, the Broadcom bootloader license[3] explicitly states that its redistribution in binary form is only permitted if it will "*… only be used for the purposes of developing for, running or using a Raspberry Pi device.*" It is unlikely that such a license would transfer to a product of your own design.

As described earlier in this chapter, the focus of the RPi Foundation is on education, and product commercialization is far from that brief. If you are planning to build an embedded Linux project that is to be commercialized, you should examine the BeagleBone platform, which is entirely open source and is supported by strong Texas Instruments documentation. In addition, you should of course purchase my book *Exploring BeagleBone* from the same Wiley mini-series.

RPi Documentation

This book integrates my experiences in developing with the RPi platform along with supporting background materials on embedded Linux, software development, and general electronics, to create an in-depth guide to building with this platform. However, it is simply not possible to cover everything in just one book, so I have avoided restating information that is listed in the key documents and websites described in this section. The first starting point for supporting documentation is the following website:

- **The Raspberry Pi Foundation website:** This provides the main support for the RPi platform, with blogs, software guides, community links, and downloads to support your development. See `www.raspberrypi.org`.

A huge amount of documentation is available on the RPi platform, but the most important documents for this book are as follows:

- **The Raspberry Pi Documentation:** This is the official documentation for the RPi that is written by the Raspberry Pi Foundation. It includes guides on getting started, configuration, guides to Linux distributions, and more. See `www.raspberrypi.org/documentation/`.

[3] `github.com/raspberrypi/firmware/blob/master/boot/LICENCE.broadcom`

- **Broadcom BCM2835 ARM Peripherals Datasheet:** This is the core document that describes the processor on most RPi models (except the RPi 2/3). It is 200 pages long and provides a technical description of the functionality and capabilities of the processor on the RPi. See `tiny.cc/erpi103`. There is also an important errata document at `tiny.cc/erpi104`.
- **The BCM2836 Document:** This document describes features of the processor on the RPi 2, and related features on the RPi 3. It should be read in association with the previous Broadcom document for the BCM2835. See `tiny.cc/erpi105`.

Key websites are also available to support your learning on this platform, with combinations of tutorials, discussion forums, sample code libraries, Linux distributions, and project ideas to stimulate your creative side. Here is a selection of important websites:

- **The website for this book:** `www.exploringrpi.com`
- **My personal blog site:** `www.derekmolloy.ie`
- **The eLinux.org website:** `www.elinux.org`

Getting started with the RPi platform software is described in Chapter 2. The remainder of this chapter discusses the RPi hardware platform, explaining the functionality that is available, summarizing the technical specifications, and providing some examples of the types of peripherals and HATs that you might like to connect to the RPi.

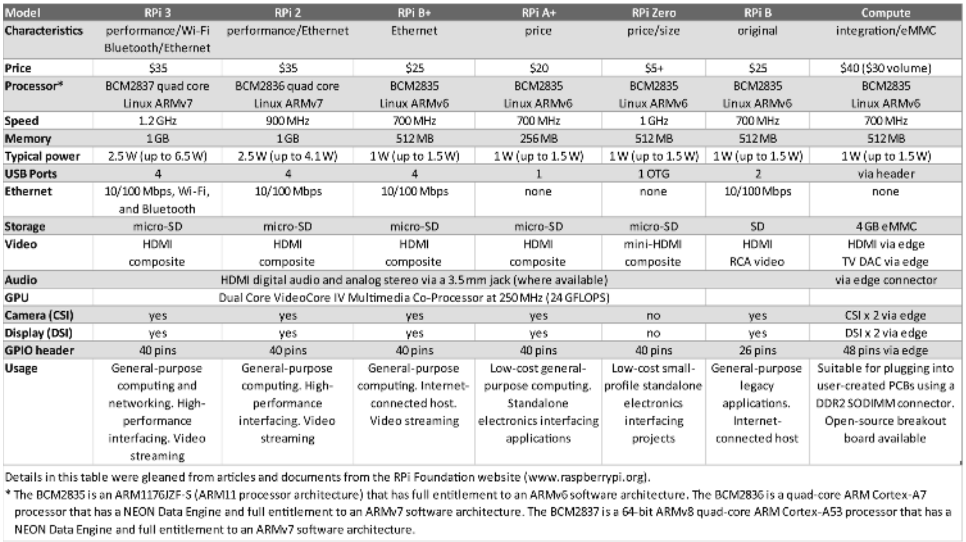
The RPi Hardware

At their heart, the RPi boards use the Broadcom BCM2835, BCM2836, and BCM2837 system on a chip (SoC). Several different RPi models are currently available, and the content in this book is perfectly applicable to all of them. However, the book focuses on more recent versions of the RPi that have a 40-pin GPIO header (for example, the RPi A+, B+, 2, 3, and Zero). If you have yet to purchase an RPi model, it is recommended that you purchase the RPi 3. It supports wired and wireless networking, and has a multicore processor, which means that it supports the superset of all the concepts described in this book. The RPi A+ and Zero do not have a wired network interface, and the RPi B+ does not have a multicore processor, but the majority of examples in this book also work perfectly well with them. If you are to use the RPi A+ or RPi Zero, it is recommended that you skip forward to the beginning of Chapter 13 so that you can read about configuring a USB wireless network adapter.

Raspberry Pi Versions

Figure 1-2 provides a summary feature comparison of the different RPi models that are presently available. Here is a quick summary of this table:

- If you need an RPi for general-purpose computing, consider the RPi 3. The 1 GB of memory and 1.2 GHz quad-core processor provide the best performance out of all the boards.
- For applications that interface electronics circuits to the Internet on a wired network, consider the RPi 3, RPi 2, or RPi B+, with cost being the deciding factor.
- If you need a small-footprint device with wireless connectivity, consider the RPi Zero. The RPi A+ could be used to develop the initial prototype.
- If you want to design your own PCB that uses the RPi (or multiple RPi boards), investigate the Compute module.



Model	RPi 3	RPi 2	RPi B+	RPi A+	RPi Zero	RPi B	Compute
Characteristics	performance/Wi-Fi Bluetooth/Ethernet	performance/Ethernet	Ethernet	price	price/size	original	integration/eMMC
Price	$35	$35	$25	$20	$5+	$25	$40 ($30 volume)
Processor*	BCM2837 quad core Linux ARMv7	BCM2836 quad core Linux ARMv7	BCM2835 Linux ARMv6	BCM2835 Linux ARMv6	BCM2835 Linux ARMv6	BCM2835 Linux ARMv6	BCM2835 Linux ARMv6
Speed	1.2 GHz	900 MHz	700 MHz	700 MHz	1 GHz	700 MHz	700 MHz
Memory	1 GB	1 GB	512 MB	256 MB	512 MB	512 MB	512 MB
Typical power	2.5 W (up to 6.5 W)	2.5 W (up to 4.1 W)	1 W (up to 1.5 W)	1 W (up to 1.5 W)	1 W (up to 1.5 W)	1 W (up to 1.5 W)	1 W (up to 1.5 W)
USB Ports	4	4	4	1	1 OTG	2	via header
Ethernet	10/100 Mbps, Wi-Fi, and Bluetooth	10/100 Mbps	10/100 Mbps	none	none	10/100 Mbps	none
Storage	micro-SD	micro-SD	micro-SD	micro-SD	micro-SD	SD	4 GB eMMC
Video	HDMI composite	HDMI composite	HDMI composite	HDMI composite	mini-HDMI composite	HDMI RCA video	HDMI via edge TV DAC via edge
Audio	HDMI digital audio and analog stereo via a 3.5 mm jack (where available)						via edge connector
GPU	Dual Core VideoCore IV Multimedia Co-Processor at 250 MHz (24 GFLOPS)						
Camera (CSI)	yes	yes	yes	yes	no	yes	CSI x 2 via edge
Display (DSI)	yes	yes	yes	yes	no	yes	DSI x 2 via edge
GPIO header	40 pins	40 pins	40 pins	40 pins	40 pins	26 pins	48 pins via edge
Usage	General-purpose computing and networking. High-performance interfacing. Video streaming	General-purpose computing. High-performance interfacing. Video streaming	General-purpose computing. Internet-connected host. Video streaming	Low-cost general-purpose computing. Standalone electronics interfacing applications	Low-cost small-profile standalone electronics interfacing projects	General-purpose legacy applications. Internet-connected host	Suitable for plugging into user-created PCBs using a DDR2 SODIMM connector. Open-source breakout board available

Details in this table were gleaned from articles and documents from the RPi Foundation website (www.raspberrypi.org).

* The BCM2835 is an ARM1176JZF-S (ARM11 processor architecture) that has full entitlement to an ARMv6 software architecture. The BCM2836 is a quad-core ARM Cortex-A7 processor that has a NEON Data Engine and full entitlement to an ARMv7 software architecture. The BCM2837 is a 64-bit ARMv8 quad-core ARM Cortex-A53 processor that has a NEON Data Engine and full entitlement to an ARMv7 software architecture.

Figure 1-2: A summary comparison of commonly available RPi models

The Raspberry Pi Hardware

Figure 1-3 and Figure 1-4 detail the core systems of typical RPi models. Figure 1-3(a) illustrates the RPi Zero, and the key systems identified by the callouts 1–11 are described in more detail in Figure 1-4. Similarly, Figure 1-3(b) illustrates the equivalent key systems on the RPi 3, and the callouts 1–15 are described in more detail in Figure 1-4.

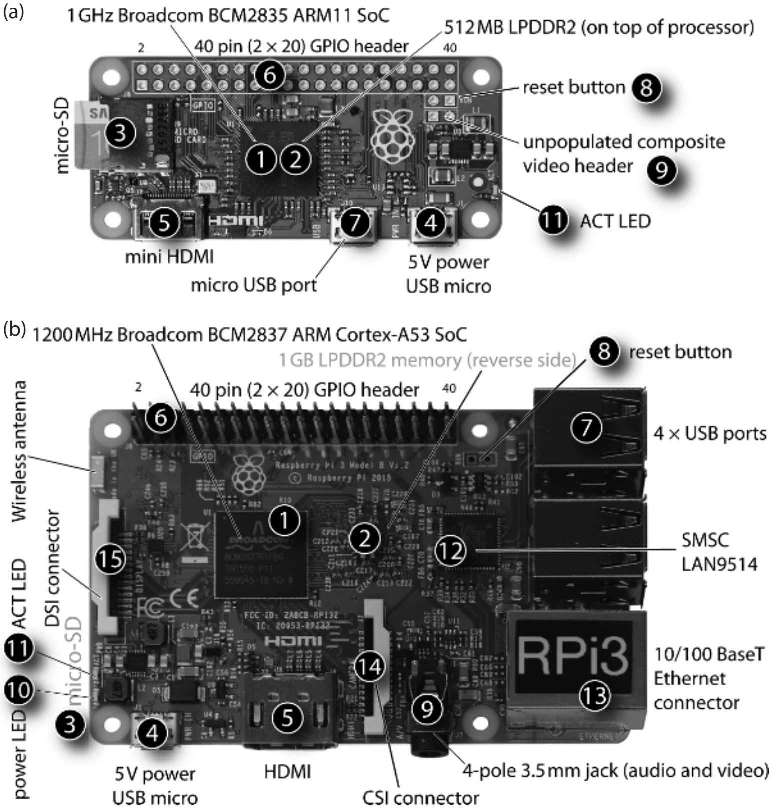


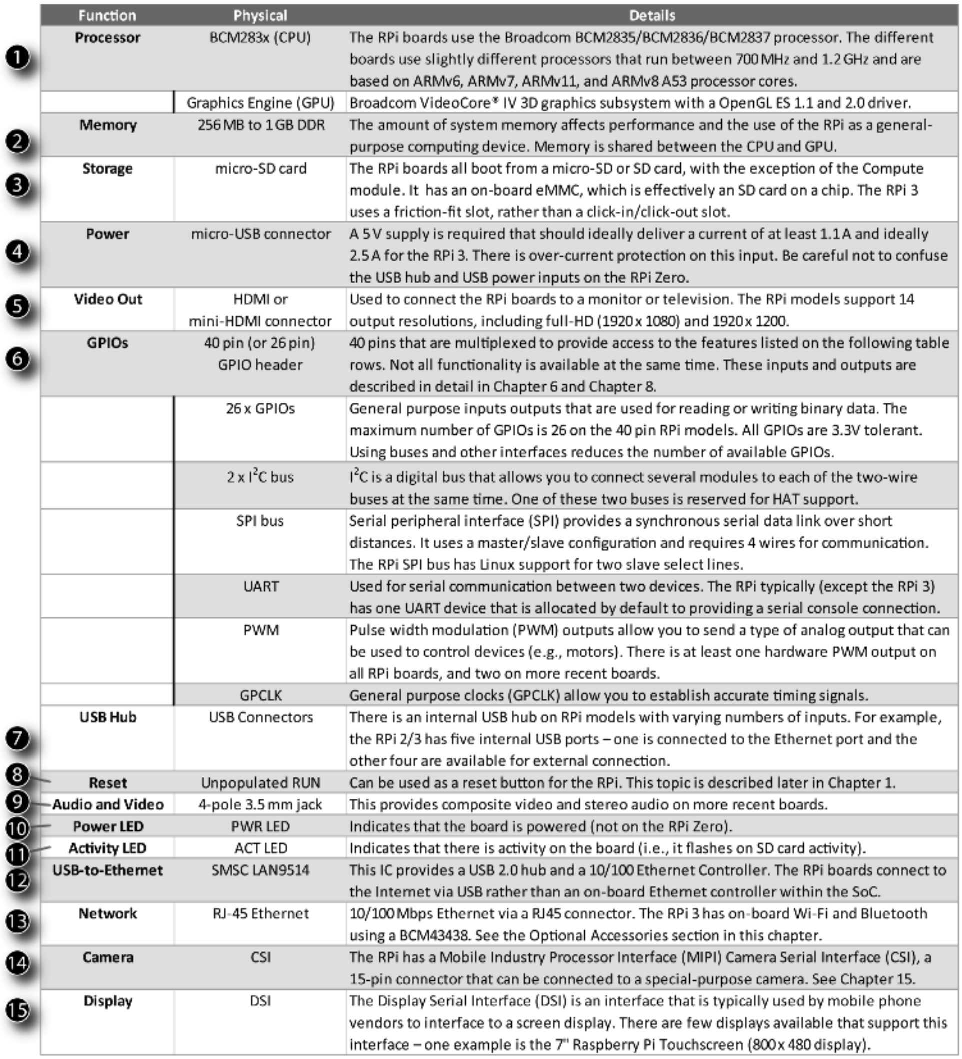
Figure 1-3: The inputs/outputs and subsystems on two RPi models (to relative scale): (a) The RPi Zero; and (b) The RPi 3

Figure 1-4 details the various inputs and outputs that are available on the GPIO header. On recent RPi models (A+, B+, 2, 3, and Zero), there are 40 pins in total on this header (2 × 20); however, not all are available for general-purpose input/outputs (GPIOs). Several of the connections have a fixed configuration:

- 8 pins are connected to ground.

- 4 pins are allocated to voltage supplies: 3.3 V (up to 50 mA) and to 5 V (up to 300 mA).
- 2 pins are reserved for HATs (discussed later in this chapter) but they can be re-tasked (see Chapter 8).

The remaining 26 connectors are available to be multiplexed to many different functions, several of which are listed in Figure 1-4 (under the GPIOs heading). The function of each of these input/output types is described in detail in Chapter 6 and Chapter 8.



	Function	Physical	Details
1	**Processor**	BCM283x (CPU)	The RPi boards use the Broadcom BCM2835/BCM2836/BCM2837 processor. The different boards use slightly different processors that run between 700 MHz and 1.2 GHz and are based on ARMv6, ARMv7, ARMv11, and ARMv8 A53 processor cores.
		Graphics Engine (GPU)	Broadcom VideoCore® IV 3D graphics subsystem with a OpenGL ES 1.1 and 2.0 driver.
2	**Memory**	256 MB to 1 GB DDR	The amount of system memory affects performance and the use of the RPi as a general-purpose computing device. Memory is shared between the CPU and GPU.
3	**Storage**	micro-SD card	The RPi boards all boot from a micro-SD or SD card, with the exception of the Compute module. It has an on-board eMMC, which is effectively an SD card on a chip. The RPi 3 uses a friction-fit slot, rather than a click-in/click-out slot.
4	**Power**	micro-USB connector	A 5 V supply is required that should ideally deliver a current of at least 1.1 A and ideally 2.5 A for the RPi 3. There is over-current protection on this input. Be careful not to confuse the USB hub and USB power inputs on the RPi Zero.
5	**Video Out**	HDMI or mini-HDMI connector	Used to connect the RPi boards to a monitor or television. The RPi models support 14 output resolutions, including full-HD (1920 x 1080) and 1920 x 1200.
6	**GPIOs**	40 pin (or 26 pin) GPIO header	40 pins that are multiplexed to provide access to the features listed on the following table rows. Not all functionality is available at the same time. These inputs and outputs are described in detail in Chapter 6 and Chapter 8.
		26 x GPIOs	General purpose inputs outputs that are used for reading or writing binary data. The maximum number of GPIOs is 26 on the 40 pin RPi models. All GPIOs are 3.3V tolerant. Using buses and other interfaces reduces the number of available GPIOs.
		2 x I^2C bus	I^2C is a digital bus that allows you to connect several modules to each of the two-wire buses at the same time. One of these two buses is reserved for HAT support.
		SPI bus	Serial peripheral interface (SPI) provides a synchronous serial data link over short distances. It uses a master/slave configuration and requires 4 wires for communication. The RPi SPI bus has Linux support for two slave select lines.
		UART	Used for serial communication between two devices. The RPi typically (except the RPi 3) has one UART device that is allocated by default to providing a serial console connection.
		PWM	Pulse width modulation (PWM) outputs allow you to send a type of analog output that can be used to control devices (e.g., motors). There is at least one hardware PWM output on all RPi boards, and two on more recent boards.
		GPCLK	General purpose clocks (GPCLK) allow you to establish accurate timing signals.
7	**USB Hub**	USB Connectors	There is an internal USB hub on RPi models with varying numbers of inputs. For example, the RPi 2/3 has five internal USB ports – one is connected to the Ethernet port and the other four are available for external connection.
8	**Reset**	Unpopulated RUN	Can be used as a reset button for the RPi. This topic is described later in Chapter 1.
9	**Audio and Video**	4-pole 3.5 mm jack	This provides composite video and stereo audio on more recent boards.
10	**Power LED**	PWR LED	Indicates that the board is powered (not on the RPi Zero).
11	**Activity LED**	ACT LED	Indicates that there is activity on the board (i.e., it flashes on SD card activity).
12	**USB-to-Ethernet**	SMSC LAN9514	This IC provides a USB 2.0 hub and a 10/100 Ethernet Controller. The RPi boards connect to the Internet via USB rather than an on-board Ethernet controller within the SoC.
13	**Network**	RJ-45 Ethernet	10/100 Mbps Ethernet via a RJ45 connector. The RPi 3 has on-board Wi-Fi and Bluetooth using a BCM43438. See the Optional Accessories section in this chapter.
14	**Camera**	CSI	The RPi has a Mobile Industry Processor Interface (MIPI) Camera Serial Interface (CSI), a 15-pin connector that can be connected to a special-purpose camera. See Chapter 15.
15	**Display**	DSI	The Display Serial Interface (DSI) is an interface that is typically used by mobile phone vendors to interface to a screen display. There are few displays available that support this interface – one example is the 7" Raspberry Pi Touchscreen (800 x 480 display).

Figure 1-4: Table of general RPi subsystems and connectors

A RESET BUTTON FOR THE RASPBERRY Pi

The RPi does not have a power or a reset button, which means that if a system lock-up occurs you must unplug and replug the micro-USB power supply. This task can be awkward and can lead to physical damage of the RPi. (On older models, a common issue is that a large 220 µF capacitor is often used for physical leverage when unplugging the USB power input and it tends to fall off!) A low-cost leaded PC power/reset switch, such as that in Figure 1-5(a), can be used to provide a solution. A two-pin male header can be soldered to the unpopulated RUN header on RPi models, as illustrated in Figure 1-5(b), and the switch attached as in Figure 1-5(c). One advantage of the leaded switch is that it can be attached to the outside of a case that contains the RPi.

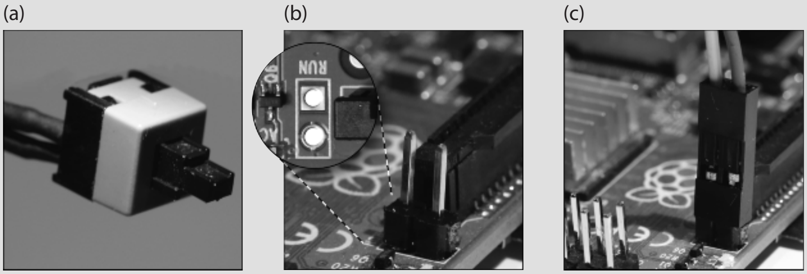


Figure 1-5: A power/reset button for the RPi: (a) A PC power/reset button; (b) A two-pin male header that is soldered to the board; and (c) Attachment of the PC power/reset button

Should you attach such a button to the RPi, it should not be used to routinely reset the RPi; rather, software commands should be issued, as described in Chapter 2.

Raspberry Pi Accessories

The RPi has minimal external requirements to use the board, typically as follows:

- A USB 2.0 cable (usually a micro-USB plug to USB-A plug) that is used to connect the RPi to a power supply, such as a desktop computer or USB mains supply (e.g., a cell phone charger)
- A micro-SD card that is used to contain the operating system, which is used to boot the board
- A CAT 5 network patch cable to connect your RPi to the network using its RJ-45 10/100 Ethernet connector

The RPi can be connected to a display using a HDMI cable (a mini-HDMI cable for the RPi Zero), but most of the examples in this book assume that the RPi is used in *headless mode*—that is, not connected directly to a display; rather, the RPi is used as a networked device that interfaces to electronic circuits, USB modules, and wireless sensors.

Important Accessories

The following accessories are important for purchase along with your RPi board.

External 5 V Power Supply (for Powering the RPi)

The RPi is best powered using a micro-USB cable that is connected to a good-quality 5 V power supply (±5%) that is rated to deliver a current of at least 1.1 A (1,100 mA) for older boards, and 2.5 A (2,500 mA) for the RPi 3. RPi boards typically require 500 mA–700 mA, but some USB peripherals (e.g., Wi-Fi adapters and webcams) have significant power requirements. The micro-USB input on the RPi boards has a Polyfuse, which limits current input to approximately 1,100 mA (with 700 mA hold current; see Chapter 4) on most RPi models, and 2,500 mA on the RPi 3. You can connect a USB power supply that is capable of supplying current of greater than 2,500 mA, but do not connect one that supplies voltage outside the range 4.75 V–5.25 V (i.e., 5 V ± 5%).

If you are having stability problems such as random reboots, random crashes, or keyboard problems, the power supply is the likely culprit. The power supply may not be able to deliver adequate current or it (or the connecting USB cable) may be of a poor quality and operating outside of tolerance. For example, some poor-quality "generic" 5 V power supplies may be advertised by vendors as suitable for a 1 A current supply (possibly referring to a short-circuit current limit), but their output voltage level may drop to unacceptable levels as the current drawn increases. Should you suspect such a problem, you should measure the voltage level on the RPi. On newer models, you use PP1 or PP2 and GND (or any of the metal shielded components), as illustrated in Figure 1-6(a). On older models use TP1 and TP2.

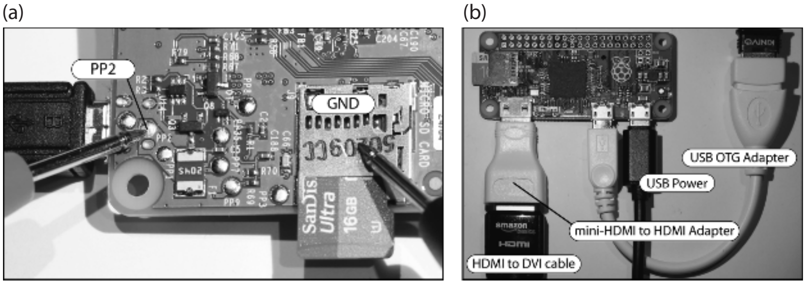


Figure 1-6: (a) Testing that the RPi supply voltage level is in the range 4.75 V to 5.25 V (i.e., 5 V ± 5%); (b) The RPi Zero and its associated connectors

Micro-SD Card (for Booting an Operating System)

Purchase a genuine, branded micro-SD card of at least 8 GB capacity. You may also require a micro-SD-to-SD adapter so that it can be used in your computer's card reader. Older RPi boards (e.g., A, B) require full-size SD cards, and such an adapter can be used with them. Many micro-SD cards are bundled with an adapter, which is a cheaper option than purchasing them separately.

The micro-SD card should be of Class 10 or greater, because the faster read/write speed will save you time in writing images in particular. Ideally, you should use an 8 GB to 32 GB micro-SD card with wear-leveling functionality because it will extend the lifespan of the card, particularly if you format but do not consume the full capacity. Larger micro-SD cards also work, but they may be cost prohibitive. (Alternative approaches to increasing the storage capacity of the RPi using USB storage devices are shortly discussed.)

Ethernet Cable (for Network Connection)

The RPi B/B+/2/3 can be connected to the Internet using a wired network connection. The RPi A/A+/Zero can be connected to the Internet using a USB wireless adapter. If you are connecting an RPi to your wired network, don't forget to purchase a CAT 5 network patch cable to connect your RPi to the network using its RJ-45 10/100 Ethernet connector. If you plan to use more than one RPi simultaneously, you could invest in a low-cost four-port switch, which can be placed close to your desktop computer.

Recommended Accessories

The following accessories are recommended for purchase along with your RPi board. If you are planning to carry out development work with the RPi, you should probably have all of them.

HDMI Cable (for Connection to Monitors/Televisions)

The RPi can be easily connected to a monitor or television that has a HDMI or DVI connector. The majority of RPi models have a full-size HDMI connector. However, the RPi Zero has a mini-HDMI socket (HDMI-C), so be careful to match that to your monitor/television type (usually HDMI-A or DVI-D). The cable that you are likely to need for the RPi Zero is an HDMI-Mini-C plug to HDMI-A male plug. A 1.8 M (6 ft.) cable should cost no more than $10. Be careful with your purchase; an HDMI-D (micro-HDMI) connector will *not* fit the RPi Zero.

Alternatively, you can purchase a low-cost ($3) mini-HDMI (HDMI-C) plug to regular HDMI (HDMI-A) socket adapter or mini-HDMI (HDMI-C) plug to DVI-D socket adapter cable. These enable you to use regular-size HDMI-A or to connect to DVI-D devices, respectively (see Figure 1-6(b)).

RPi ZERO USB ON-THE-GO (OTG)

The RPi Zero uses USB On-The-Go (OTG) to connect to USB peripherals. USB OTG is often used for devices that switch between the roles of USB client and host. For example, USB OTG connectors are often used to allow cell phones or tablet computers to connect to external USB storage devices. The USB OTG connector allows the RPi host to connect to a slave device such as a Wi-Fi or Bluetooth adapter, as illustrated in Figure 1-6(b).

USB to Serial UART TTL 3.3 V (for Finding Problems)

The USB-to-TTL UART serial cable, as illustrated in Figure 1-7(a), is one accessory that proves really useful when there are problems with the Linux installation on your board. It can provide you with a console interface to the RPi without the need for connection to an external display and keyboard.

Ensure that you purchase the *3.3 V level* version and ideally purchase a version with 0.1″ female headers pre-attached. This cable contains a chipset and requires that you install drivers on your desktop computer, creating a new COM port. The FTDI TTL-232R-3V3 cable works well and provides a very stable connection (~$20). See `tiny.cc/erpi106` for the datasheet and the VCP link to the software drivers for this adapter cable.

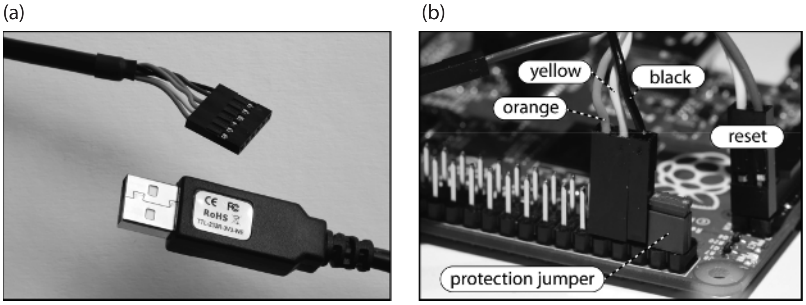


Figure 1-7: (a) The USB-to-TTL 3.3 V serial cable and, (b) its connection to the RPi

The cable connects to the serial UART on the RPi, which is available via the GPIO header. With the RPi powered using a regular USB 5 V supply, connect the cable to the RPi in the following way (as illustrated in Figure 1-7(b)):

- The black ground (GND) wire to Pin 6 on the GPIO header, which is the RPi GND pin
- The yellow receive (RXD) wire to Pin 8 (GPIO14) on the GPIO header, which is the UART transmit pin (TXD0)
- The orange transmit (TXD) wire to Pin 10 (GPIO15) on the GPIO header, which is the UART receive pin (RXD0)

Note that the 40-pin GPIO header is described in detail in Chapter 6. The exact use of this cable is described in Chapters 2, 3, and 8.

This cable is also used to test the UART connection on the RPi in Chapter 8 and to program the Arduino Pro devices in Chapter 11.

WARNING **The RPi is 3.3 V tolerant but makes a 5 V supply available on the GPIO header pins 2 and 4. The easiest way to destroy the RPi is to accidentally connect these pins to a circuit that requires 3.3 V logic levels, or to accidentally short these pins with other pins on the GPIO header. To help prevent accidental contact, you can bridge these pins with an insulated jumper connector, as illustrated in Figure 1-7(b). The plastic cover insulates the pins from contact and prevents you from mistakenly connecting a 5 V supply to your circuit.**

Optional Accessories

The following sections describe optional accessories that you may need, depending on the applications that you are developing.

USB Hub (to Connect to Many USB Devices)

Most RPi models have a built-in USB hub that allows several devices to be connected to the RPi simultaneously. If you plan to connect many devices to the RPi, you will need an external USB hub. USB hubs are either bus powered or externally powered. Externally powered hubs are more expensive; however, if you are using several power-hungry adapters (Wi-Fi in particular), you may need a powered hub.

One issue that you have to be aware of with powered USB hubs is that many are back feeding. *Back feeding* (*back powering*) is where a USB hub connected to the RPi hub (not the micro-USB power) supplies power back into the RPi through the RPi hub. It can cause difficulties if you have two separate power supplies competing to power the RPi. In addition, there is no protection on the RPi hub to prevent excessive current from being drawn.

This is not an issue on more-recent RPi models (e.g., the RPi 2/3) because circuitry is present to prevent back powering. However, it can also be useful to use a single power supply for your project. The easy way to do this is to attach a cable from the powered USB hub to the RPi micro-USB power input.

Micro-HDMI to VGA adapters (for VGA Video and Sound)

Several low-cost HDMI-to-VGA adapters are for sale for converting the HDMI output to a VGA output. As well as providing for VGA video output, many of these connectors provide a separate 3.5 mm audio line out, which can be used if you want to play audio using your RPi. There are also USB audio adapters available that can provide high-quality playback and recording functionality. These adapters and their usage is described in Chapter 15. Many RPi models also make composite video and stereo audio available via a four-pole 3.5 mm connector. A standard 3.5 mm four-pole headphone jack (with microphone) can be used for this task. The tip of the jack is connected to the left audio channel, followed by the right audio channel, ground connection, and then the video channel.

Wi-Fi Adapters (for Wireless Networking)

The RPi 3 has on-board Wi-Fi, but this capability can also be added to other RPi models using the many different Wi-Fi adapters that are available, such as those in Figure 1-8(a); however, not all adapters will work on the RPi. The Linux distribution and the chipset inside the adapter will determine the likelihood of success. Wi-Fi configuration and applications are discussed in detail in Chapter 13, which tests a range of different low-cost adapters that are widely available. Be aware that manufacturers can change chipsets within the same product and that buying an adapter from the list in Chapter 13 does not guarantee that it will work. You are more likely to succeed if you can confirm the chipset in the adapter you are planning to purchase and evaluate that against the list. You can use a small low-cost USB current meter, such as the one illustrated in Figure 1-8(c) ($3), which enables you to gain some insight into the power utilization of the RPi and the impact of connecting a Wi-Fi adapter.

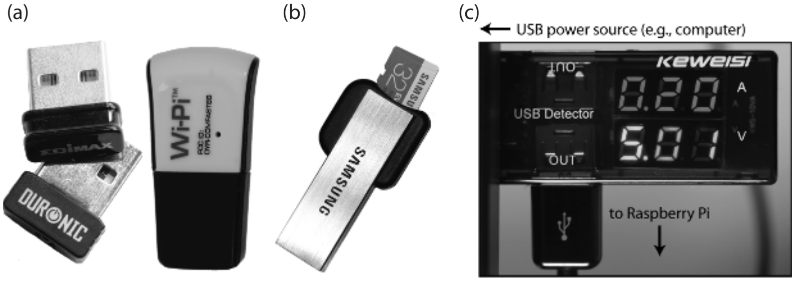


Figure 1-8: USB adapters: (a) Wi-Fi adapters; (b) Memory card reader/writer; and (c) A low-cost USB current and voltage monitor

USB Storage Devices (for Additional Storage)

USB flash drives, USB hard disks, and USB SD card reader/writers can be attached to the RPi for additional storage. The device can be prepared with a Linux file system and mounted under the RPi file system (see Chapter 3). One such device that is particularly useful is a USB card reader/writer, as illustrated in Figure 1-8(b). These devices have similar prices to USB flash drives, and they support "hot swapping" of the micro-SD card. In addition, they prove particularly useful if you need to mount the root file system of one RPi on another RPi for file interchange or to correct a configuration error on the card that is preventing the other RPi from booting (see Chapter 3). In addition, such a device can be utilized on a desktop machine to write a new Linux image to a micro-SD card.

USB Webcam (for Capturing Images and Streaming Video)

Attaching an RPi camera, as illustrated in Figure 1-9(a) and Figure 1-9(b), or a USB webcam, as illustrated in Figure 1-9(c), can be a low-cost way to integrate image and video capture into your RPi projects. In addition, utilizing Linux libraries such as Video 4 Linux and Open Source Computer Vision (OpenCV) enables you to build "seeing" applications. This topic is discussed in detail in Chapter 15.

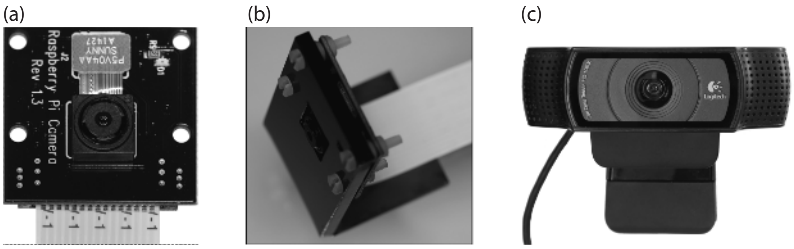


Figure 1-9: (a) RPi NoIR Camera, (b) RPi Camera bracket, and (c) Logitech C920 USB webcam

USB Keyboard and Mouse (for General-Purpose Computing)

It is possible to connect a USB keyboard and mouse to the RPi or to use a 2.4 GHz wireless keyboard and mouse combination. Very small wireless handheld combinations are available, such as the Rii 174 Mini, Rii i10, and ESYNiC mini, all of which include a handheld keyboard with integrated touchpad. A USB Bluetooth adapter is also useful for connecting peripherals to the RPi. A similar Bluetooth keyboard/touchpad is utilized in Chapter 14.

Cases (for Protecting the RPi)

Many different cases are available for protecting your RPi, including the one illustrated in Figure 1-10(a) ($6). Cases are useful for protecting the RPi from accidental short circuits (e.g., placing the RPi on a metal surface), but they do have an impact on the temperature that the RPi operates at (see Chapter 12). Try to ensure that you purchase a case with adequate ventilation, but avoid noisy active-ventilation solutions or ridiculous water-cooled solutions!

HATs

HATs (Hardware Attached on Top) are daughter boards that can be attached to the GPIO expansion header on the RPi. Add-on boards were available for the 26-pin GPIO header on older RPi models, but the RPi had no formal mechanism for identifying which daughter board was attached. HATs were introduced in conjunction with the release of the RPi B+. Some pins on the expanded 40-pin GPIO header (ID_SD and ID_SC) of newer RPi models are utilized to automatically identify which HAT is attached to the RPi. This allows the Linux OS to automatically configure pins on the GPIO header and to load drivers that make working with the HATs very easy.

Figure 1-10(b) illustrates the RPi Sense HAT ($35). It contains an: 8 × 8 LED matrix display, accelerometer, gyroscope, magnetometer, air pressure sensor, temperature sensor, humidity sensor, and a small joystick. Figure 1-10(d) illustrates a low-cost blank prototyping HAT that can be used to design your own HAT, which includes space on the bottom right for a surface-mounted EEPROM that can be used to identify the HAT to the RPi.

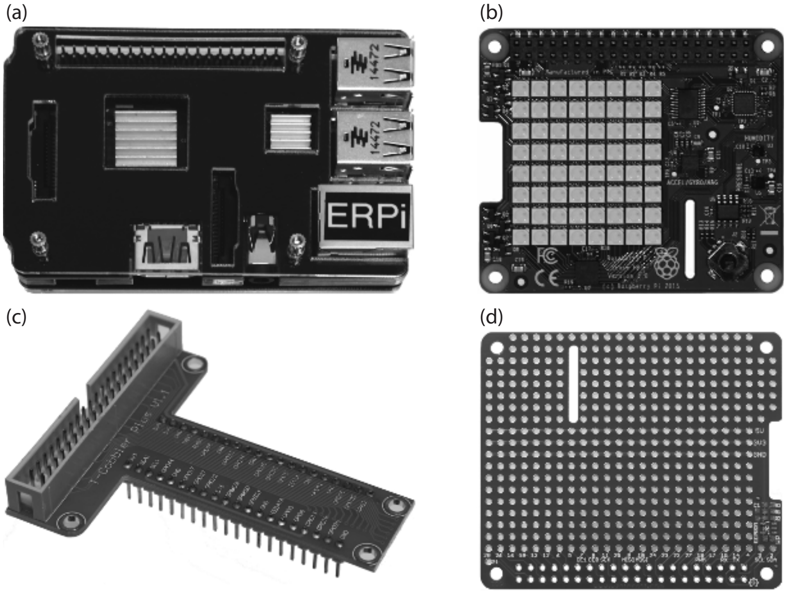


Figure 1-10: RPi Accessories: (a) An example case; (b) The Sense HAT; (c) The T-Cobbler board; and (d) A prototyping HAT

An alternative to designing your own HAT is to use the T-Cobbler board as illustrated in Figure 1-10(c) to break out the RPi GPIO header to a breadboard using a 40-pin ribbon cable, which is available with the T-Cobbler. This sits neatly into a prototyping breadboard (see Chapter 4), providing clear pin labels for all of the RPi GPIO pins.

How to Destroy Your RPi!

RPi boards are complex and delicate devices that are very easily damaged if you do not show due care. If you are moving up from boards like the Arduino to the RPi platform, you have to be especially careful when connecting circuits that you built for that platform to the RPi. Unlike the Arduino Uno, the microprocessor on the RPi cannot be replaced. If you damage the microprocessor SoC, you will have to buy a new board!

Here are some things that you should *never* do:

- Do not shut the RPi down by pulling out the USB power supply. You should shut down the board correctly using a software shutdown procedure (see Chapter 2).
- Do not place a powered RPi on metal surfaces (e.g., aluminum-finish computers) or on worktops with stray/cut-off wire segments, resistors, etc. If you short the pins underneath the GPIO header, you can easily destroy your board. You can buy a case such as that in Figure 1-10(a). Alternatively, you can attach small self-adhesive rubber feet to the bottom of the RPi.
- Do not connect circuits that source/sink other than very low currents from/to the GPIO header. The maximum current that you can source or sink from many of these header pins is approximately 2 mA to 3 mA. The power rail and ground pins can source and sink larger currents. For comparison, some Arduino models allow currents of 40 mA on each input/output. This issue is covered in detail in Chapter 4 and Chapter 6.
- The GPIO pins are 3.3 V tolerant. Do not connect a circuit that is powered at 5 V; otherwise, you will destroy the board. This is discussed in Chapter 4, Chapter 6, and Chapter 8.
- Do not connect circuits that apply power to the GPIO header while the RPi is not powered on. Make sure that all self-powered interfacing circuits are gated by the 3.3 V supply line or using optocouplers. This is covered in Chapter 6.

You should *always* do the following:

- Carefully check the pin numbers that you are using. There are 40 pins on the GPIO header, and it is very easy to plug into header connector 21 instead of 19. The T-Cobbler board in Figure 1-10(c) is very useful for interconnecting the RPi to a breadboard, and it is highly recommended for prototyping work.

Summary

After completing this chapter, you should be able to do the following:

- Describe the capability of the Raspberry Pi (RPi) and its suitability for different project types.
- Describe the major hardware systems and subsystems on the RPi boards.
- Identify important accessories that you can buy to enhance the capability of your RPi.
- Have an appreciation of the power and complexity of the RPi as a physical computing platform.
- Be aware of the first steps to take in protecting your board from physical damage.

Support

The key sources of additional support documentation are listed earlier in this chapter. If you are having difficulty with the RPi platform and the issues are not described in the documentation, visit the Raspberry Pi Community Forums at `www.raspberrypi.org/forums/`. Please remember that the people on these forums are community members who volunteer their time to respond to questions.