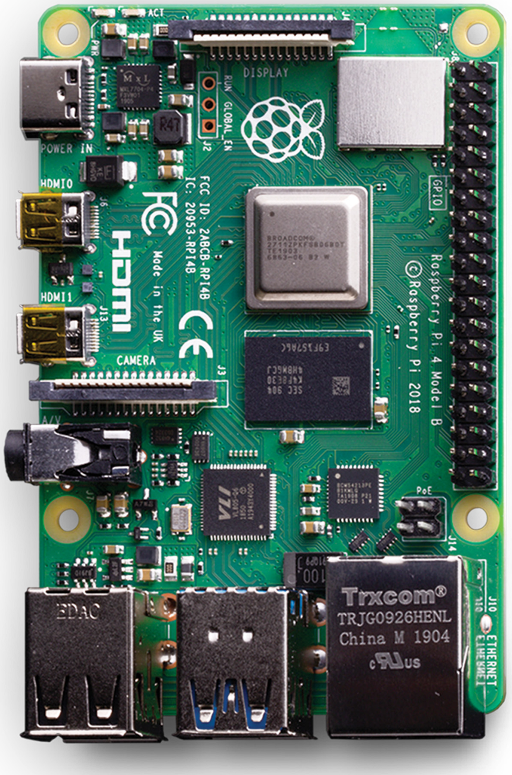
element14
AN AVNET COMMUNITY

March 2020



ESSENTIAL RASPBERRY PI TIPS

TABLE OF CONTENTS

ESSENTIAL RASPBERRY PI TIPS

February of 2020 will mark the eighth birthday of the initial release of the Raspberry Pi 1 Model B. While the Pi is far from the first single board computer, none have enjoyed the same widespread success. Alternatives to the Pi may have faster CPU speeds, more IO, or specialized on-board peripherals, and they have their place, but none of them have that "just right" combination of price, support, power, and features. According to the **Raspberry Pi blog**, over 19 million Pis have been sold.

This writer has been using and following the evolution of the Raspberry Pi since its release. I watched as it transitioned from a tool for students and hobbyists to a viable option for certain types of engineering projects. If you're considering the Raspberry Pi because you are curious about electronics and coding, or even if you are a practicing engineer, just go for it ! The size of the community and the sheer number of publically available forums, tutorials, and projects means that you will be in good company as you explore and experiment.

As "just right" as the Raspberry Pi may be, unfortunately popularity doesn't always equal perfection. The Pi has its eccentricities and idiosyncrasies like anything else. But that's okay; the following tips are going to help you overcome some of the vulnerabilities and limitations of the Pi.

- **Cabe Atwell**, element14 Contributor

element14 Community Team

CHAPTER - 1

A WORD ON POWER SUPPLIES

Raspberry Pis do not come with on-board power supplies; you will have to provide one. As a rule of thumb, consider a higher mA rating than necessary. The additional capacity may keep your Pi from resetting when you start adding accessories and external circuitry. Even if that giant breadboarded circuit you cobbled together doesn't cause a reset or a lock up condition, you may inadvertently hurt your Pi's performance. You might even corrupt the SD card. If the supply voltage dips, the GPU in the Pi will throttle the CPU speed down. Incidentally, overheating may also cause the GPU unit to throttle down the CPU speed.

While the Raspberry Pi 4 requires a minimum of 2.5A, the official power supply has a 3A output. The extra 500mA is good insurance against throttling, resets, and lock ups.

CHAPTER - 2

STORAGE

You may have noticed that Raspberry Pis have no provisions for a hard drive; this limitation is actually pretty easy to overcome. You can connect to an NAS unit via the Ethernet port or WiFi, or you can connect an external drive via one of the USB ports. And, by the way, connecting external storage isn't just a novelty; it's insurance. As you experiment with the Pi, inevitably you will corrupt your SD card. Get into the practice of storing anything important absolutely anywhere but the SD card.

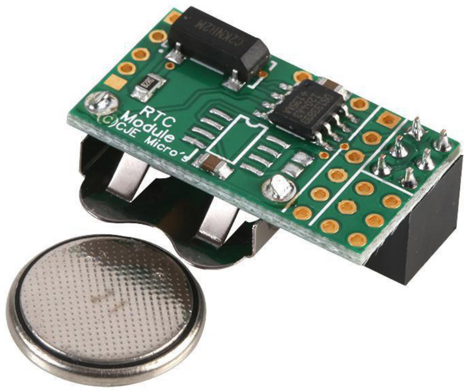
A USB-to-SATA adaptor is a fast and cheap way to give your Raspberry Pi a hard drive.

CHAPTER - 3

RASPBERRY PIS CANNOT TELL TIME BY THEMSELVES, AND OTHER MATTERS

There are no iterations of the Raspberry Pi that come with a real-time clock (RTC) or analog-to-digital converter (ADC). Also, the GPIO pins on a Pi operate at 3.3VDC, whereas most accessories you might want to use operate at 5VDC. Fortunately, RTC, ADC, and 3.3V to 5V converter modules are inexpensive and easily obtained devices.

Also, if your time telling requirements are not terribly stringent, then you can simply set up the Pi to check the time via a WiFi, Ethernet, or Bluetooth connection.

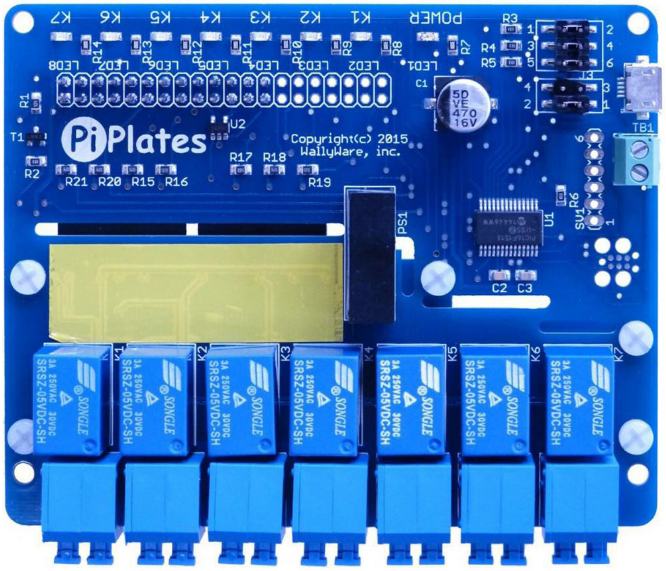


An RTC module like this 4D Systems RPI-RTC fits directly onto the Raspberry Pi header.

CHAPTER - 4

SWITCHING INDUCTIVE LOADS

Raspberry Pis (and digital electronics in general) are susceptible to the electromagnetic interference (EMI) and back electromotive force (EMF) that come from switching inductive loads. Examples of these interference culprits include solenoid valves, relays, contactors, motors, and more. This author has personally witnessed many a Pi reset or lock up just by being situated too close to a solenoid valve or a beefy relay. To avoid these side effects, and the possibly permanent damage that comes with them, always isolate your Pi from inductive loads with a Pi HAT (Hardware Attached on Top, a hardware specification made for Raspberry Pi) designed for this purpose, or an opto-isolator. Also, as a general rule, avoid powering inductive loads from a Pi's power supply. Roughly speaking, the amount of trouble an inductor will give you will be proportional to its physical size. In some cases, you may find that you will need quite a bit of physical separation between Pi and inductor. In cases where this separation isn't an option, consider employing a snubbing circuit of some sort, like a diode in parallel with the relay coil or an RC snubber on a DC motor.



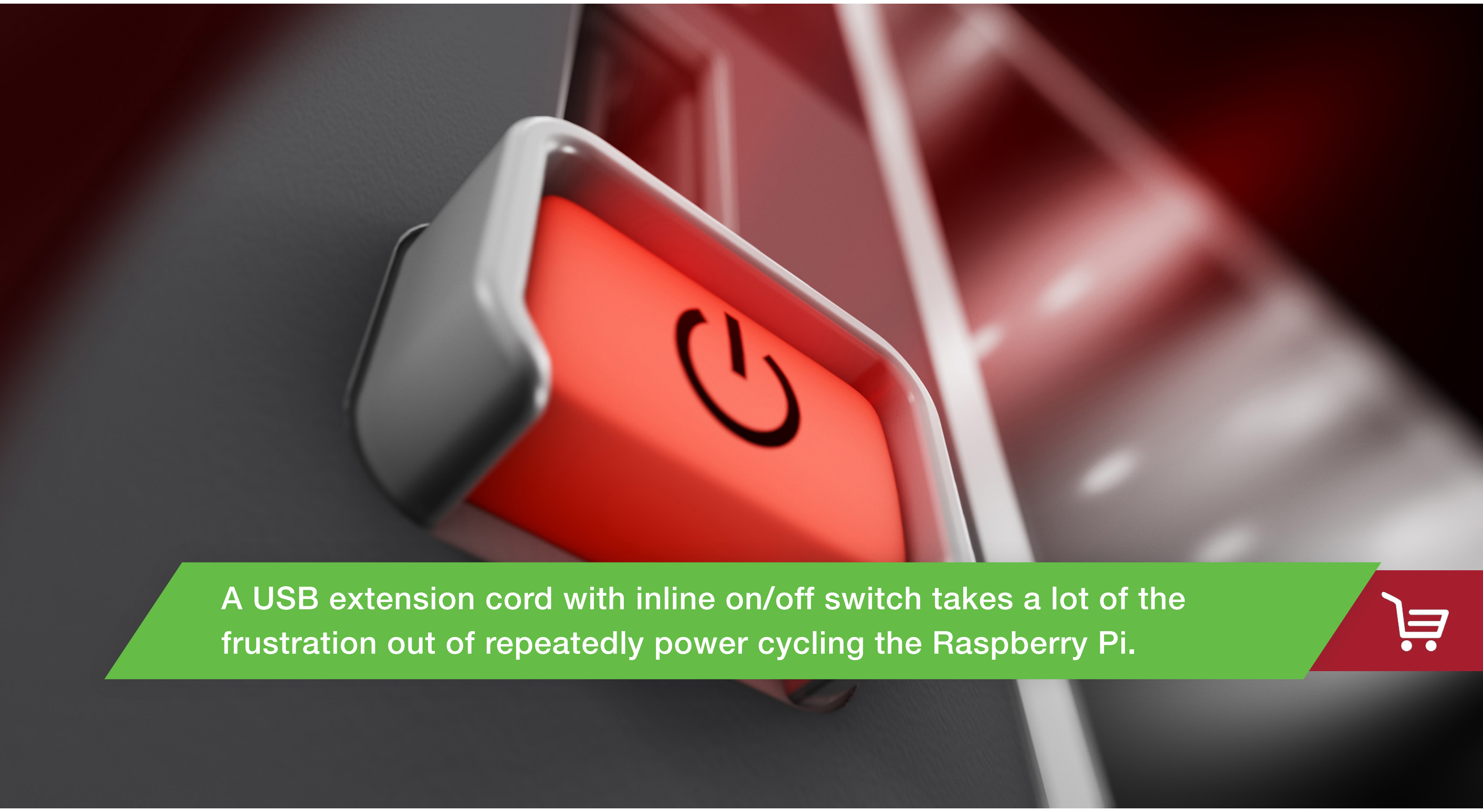
Judicious use of a Pi HAT like this "Relay Plate" will prevent damage to your Raspberry Pi.

CHAPTER - 5

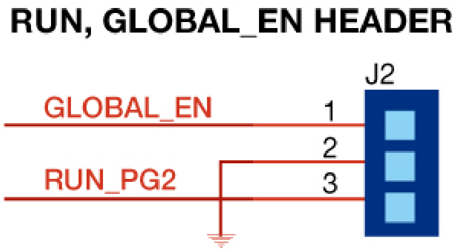
RASPBERRY PIS DO NOT HAVE A RESET BUTTON

Unlike a machine running Windows, when something goes awry, a Pi user will not have the option of Ctrl+Alt+Delete to bring it out of a halted state. Inevitably, during the course of experimentation, you will lock up your Pi. Usually this isn't a big deal, and you have a few ways of dealing with it. For one, you can power cycle your Pi by unplugging the power supply from it. This can get pretty tedious, though!

A better option is to get a power cable with an integrated on/off switch, or make or purchase a USB pass-thru board with a switch on it. If you have a Pi 4, another option is to reset the Pi's SoC by connecting the Global_EN pin to a ground.



A USB extension cord with inline on/off switch takes a lot of the frustration out of repeatedly power cycling the Raspberry Pi.



NOTE: RUN IS 3v3, GLOBAL_EN IS 5V

The Global_EN pin can be found on header J2.

MORE INFO

CHAPTER - 6

THE HUMBLE LED

At a glance, this tip is going to seem so simple that it isn't worth mentioning. However, the path to success is traversed with a series of steps, and if you choose to apply this tip you will be taking a deceptively large step toward your goal. This tip may have been ever so slightly overhyped, so let's just get to it!

When writing code, specifically code that controls the hardware on the Pi, start by blinking an LED. It doesn't matter if you are attempting to control the GPIO pins on the Pi itself, or some aspect of a Pi HAT or accessory board. When you can blink an LED at exactly the rate you want, then you have begun to exert a measure of control over that hardware. Blinking an LED is the "Hello World" of hardware programming.

LEDs can also be used as a kind of physical breakpoint in your code, by setting one up to blink or activate when a program hits certain points or thresholds. This isn't always possible due to the nature of a project, but when circumstances allow it, an LED is a dead simple diagnostic tool.

CHAPTER - 7

CREATIVE GRAPHICS

Python is the preferred programming language for the Raspberry Pi, and shortly after you start writing Python code you're going to want to make your own custom GUIs. TkInter is the de-facto GUI package for Python, as stated right at the top of its Wiki page. Unfortunately, Tkinter isn't documented very well in any one place. To really get to know it, you will have to spend a lot of time combing the internet and pooling information from multiple sources.

TkInter can be a little overwhelming/frustrating for newer users. If you want to give yourself a break, or you find you don't need many of the features of TkInter, take a look at Guizero, a library for Python 3.

It is perhaps the fastest and easiest way to make a GUI in Python: **About - Guizero**.

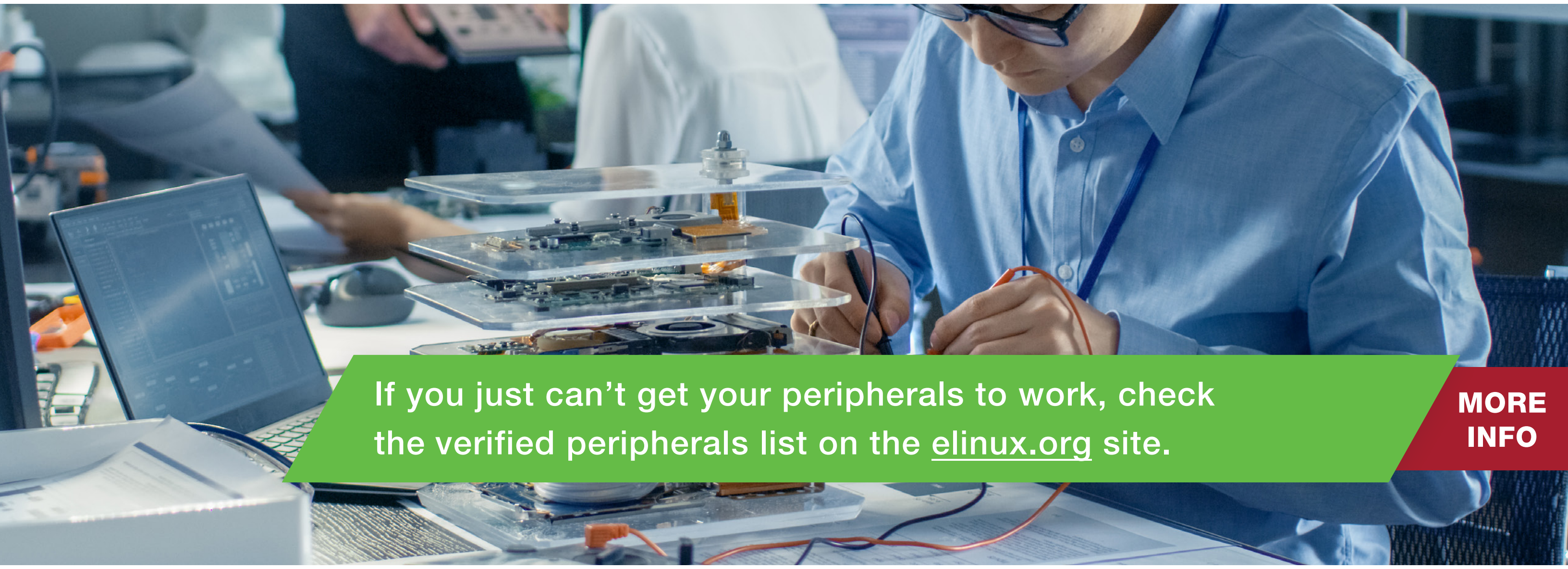
One way to accelerate this process is to purchase a copy of Jan Bodnar's Tkinter programming e-book. In the book, Jan has already done much of the research for you.

MORE INFO

CHAPTER - 8

TROUBLESHOOTING

- If you are having boot or software issues, check the green light on the Pi. If it's blinking, then the Pi is working. If the LED is off or on without blinking, check your SD card, as it may not be seated correctly in its slot. If that isn't the issue, then the SD card data may be corrupted and you may have to remount the OS.
- Make sure the mouse and keyboard are plugged into the Pi before you turn it on. If you don't, the Pi will not detect them on startup. This is also true for HDMI displays. Try to make sure that the display is powered on before powering up the Pi. At one point, I witnessed a monitor that initialized slower than the Pi. When I applied power to the Pi and this particular monitor at the same time, the Pi never detected it; the monitor had to be **ON** before the Pi started up. If you just can't get your peripherals to work, check the verified peripherals list on the elinux.org site.
- Speaking of peripherals, if the characters showing up on screen do not match with the key you've pressed, then you probably are not using the default UK layout keyboard. You can change this from the Raspbian desktop by opening the main menu, going to **Preferences**, then clicking **Raspberry Pi Configuration**. Click the **Localization** tab, and then click the **Set keyboard** option.



If you just can't get your peripherals to work, check the verified peripherals list on the elinux.org site.

MORE INFO

- Alternatively, you can make the change from the command line. If you are running Raspbian Lite, this is the only way to change the keyboard layout. Run the command sudo raspi-config, select **Localisation options**, then **Change keyboard layout**, and finally select the model of keyboard you are using. Typically this will be **Generic 105-key PC (intl.)**. You will then be prompted to select your keyboard layout.

- If you are unable to gain access to the Pi via an SSH (Secure Shell) session, then you probably have to activate SSH communication. On the Raspbian desktop, click the **Preferences** option and then the **Configure Raspberry Pi** option. You will be presented with a Raspberry Pi Configuration window. Click the **Interfaces** tab, and then choose the **Enabled** option to the right of the SSH: line item. If your Raspberry Pi is not connected to a monitor, then you will have to use a separate computer to create a file named SSH (with no file extension) with a text editor to the boot partition on the SD card. The addition of this file will allow you to SSH into the Pi.

Looking for more tips on working with the Pi, information about useful add-ons, or do you just want to see cool Pi projects? Check out our Raspberry Pi space on the element14 community!

element14 is a Community of over 700,000 makers, professional engineers, electronics enthusiasts, and everyone in between. Since our beginnings in 2009, we've provided a place to discuss electronics, get help with your designs and projects, show off your skills by building a new prototype, and much more. We also offer online learning courses such as our Essentials series, video tutorials from element14 Presents, and electronics competitions with our Design Challenges. We hope you have found this guide useful and enlightening.

element14
AN AVNET COMMUNITY

300 S. Riverside Plaza, Suite 2200
Chicago, IL 60606

www.element14.com/community

Facebook.com/e14Community
Twitter.com/e14Community