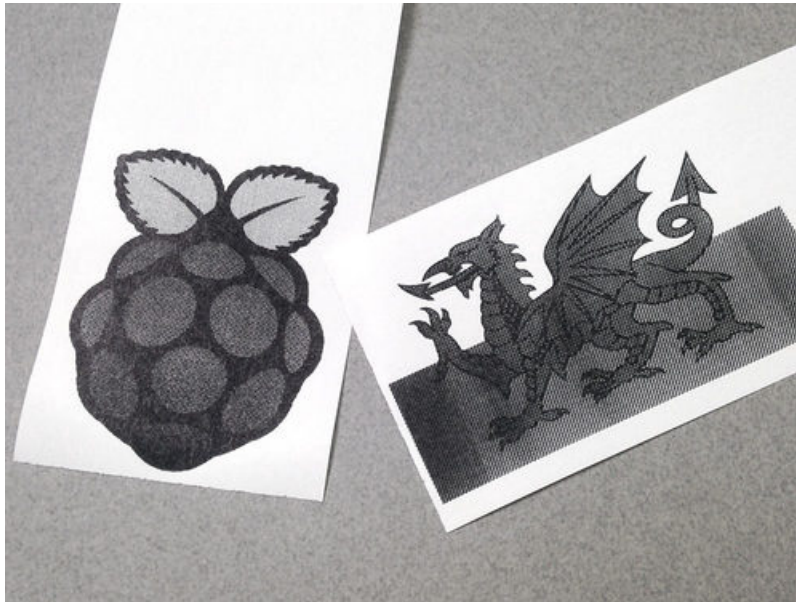


## Networked Thermal Printer using Raspberry Pi and CUPS

Created by Phillip Burgess



Last updated on 2018-08-22 03:53:24 PM UTC

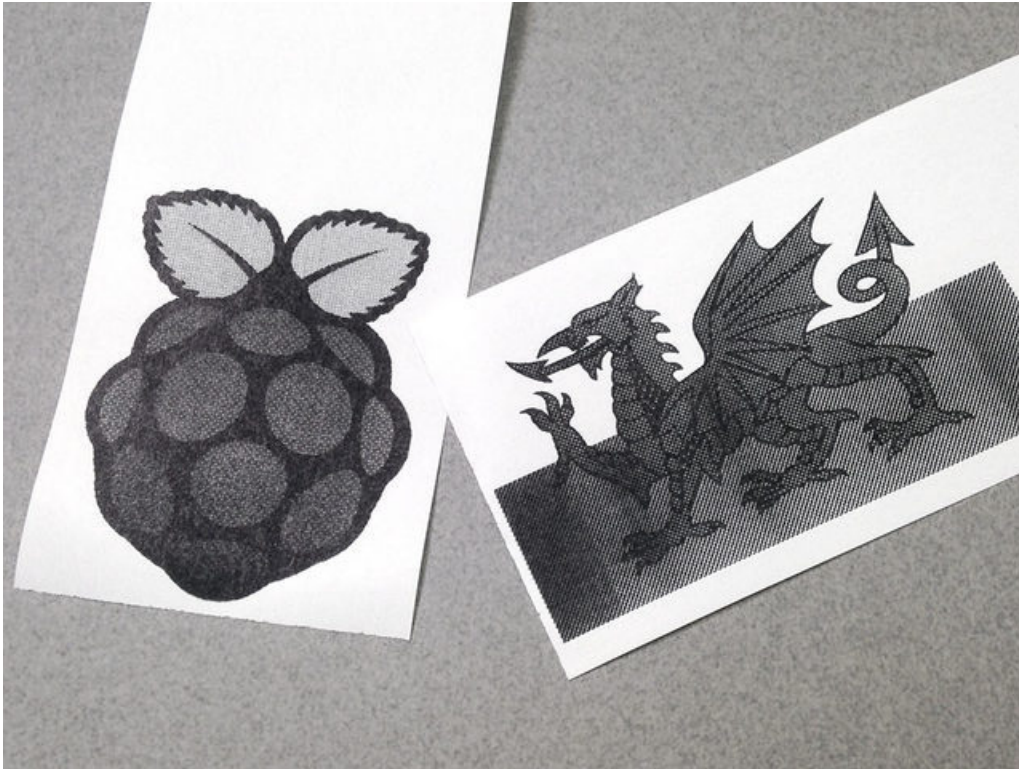
## Guide Contents

Guide Contents	2
Overview	3
First-Time System Setup	5
Connect and Configure Printer	7
Commence Major Installation!	9
Network Printing	17

## Overview

**CUPS** — the *Common Unix Printing System* — is an open source print spooling and scheduling system. One of the interesting aspects of CUPS is its *filtering* system, which can reconstitute print job data between formats. For example, rasterizing PDF graphics for output on non-PostScript printers.

Turns out **there's a CUPS filter for thermal printers** similar to those Adafruit offers. With a little setup, it's possible to have these producing razor-sharp graphics unlike anything I've seen from a thermal receipt printer before. *Much* nicer than our improvised Python library!



This guide is based on a [tutorial by Stewart Russell \(https://adafru.it/mUE\)](https://adafru.it/mUE), with some changes and additions to work with Adafruit's various thermal printers.

### Required items:

- Raspberry Pi computer (any model or version)
- Adafruit Mini Thermal Receipt Printer (any model, though the “Tiny” model with USB is easiest to interface) — **the Raspberry Pi 3 currently requires this USB model, no TTL support yet** — Pi 2 and earlier can use either type
- Thermal paper roll(s) — the smaller printer models require special smaller rolls — and you may want at least one spare to run through while experimenting with settings
- 5V power supply (2A or larger)
- DC jack adapter and perhaps some breadboard jumper wires
- Latest version of [Raspbian Jessie \(https://adafru.it/fQi\)](https://adafru.it/fQi) from Raspberry Pi web site (*not* Jessie Lite)
- 8 GB or larger SD/microSD card as appropriate
- Monitor, keyboard and mouse
- Network connection — Ethernet or WiFi

**Some prior Raspberry Pi experience is assumed** — downloading the OS, writing an SD card, basic system and network configuration, etc. You can search the Adafruit Learning System for related guides if any of this is unfamiliar.

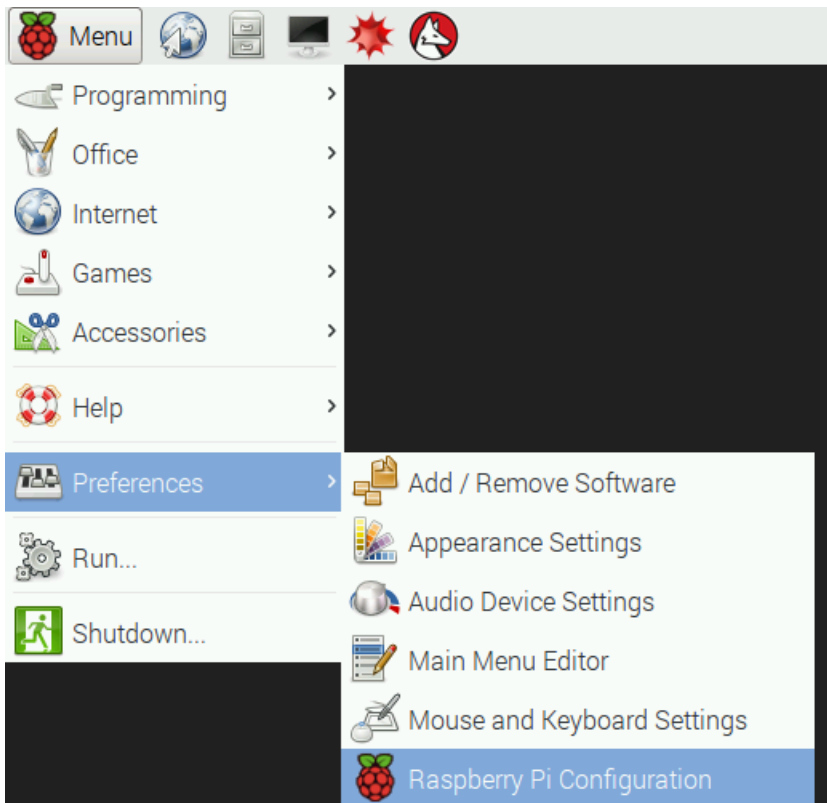


## First-Time System Setup

If you're using a TTL printer (not USB), DO NOT connect it to the Raspberry Pi yet! It will spit paper like mad. Some system configuration is required first...

Download the latest version of [Raspbian \(https://adafru.it/fQi\)](https://adafru.it/fQi) if you haven't already. This guide *requires* the **full** version of Jessie — not the “Lite” edition — as some of the setup requires the graphical user interface. You'll also need a keyboard, mouse and monitor attached. Write the OS to an 8GB or larger SD card, insert in the Raspberry Pi and power it up. After a minute or two you'll see the Raspbian desktop.

From the “Pi” menu at the top left, select Preferences→Raspberry Pi Configuration. Or Linux guru types might be more comfortable opening a terminal window and running **raspi-config** from the command line.

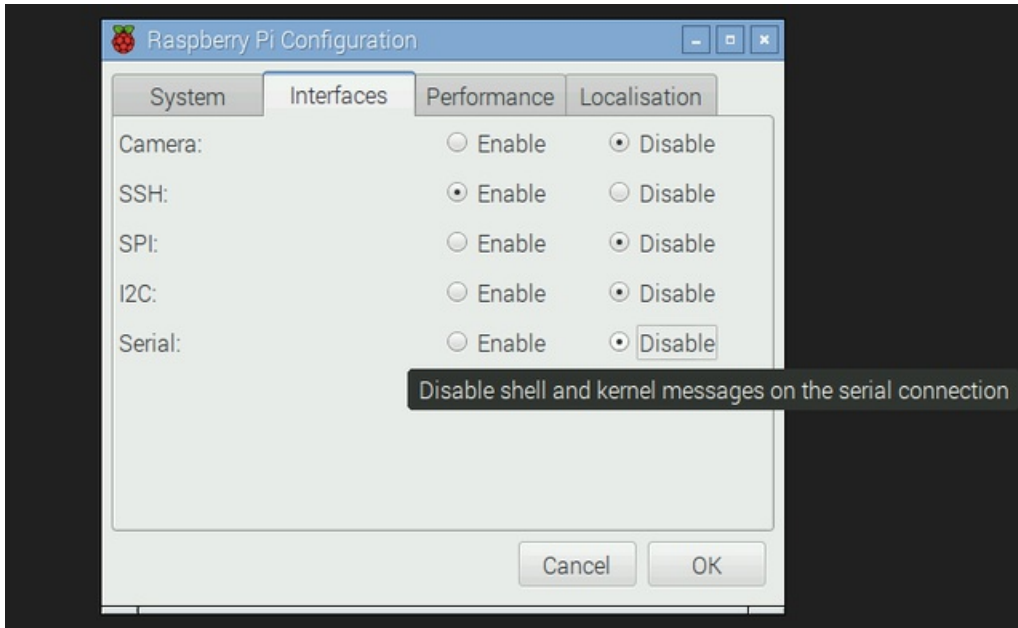


The following settings are **required**:

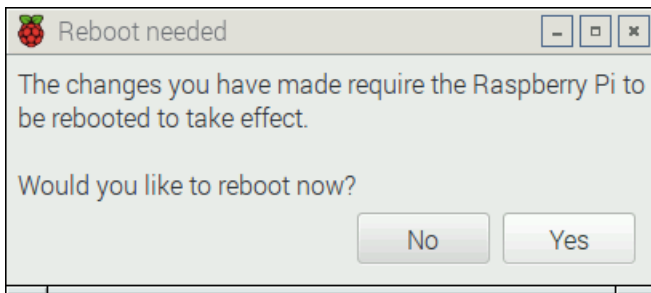
- Expand Filesystem
- If using a TTL (not USB) printer: **Under “Interfacing Options,” select “Serial.” Turn OFF the login shell over serial, and ENABLE the hardware serial port. NO and YES, respectively. This is vital!**

The following are **optional** but **recommended**:

- Change hostname to distinguish it from other Raspberry Pi systems on the network
- Change password (everyone knows the default)
- Disable overscan
- On "Localization" tab, set up Locale, Timezone, Keyboard, etc.



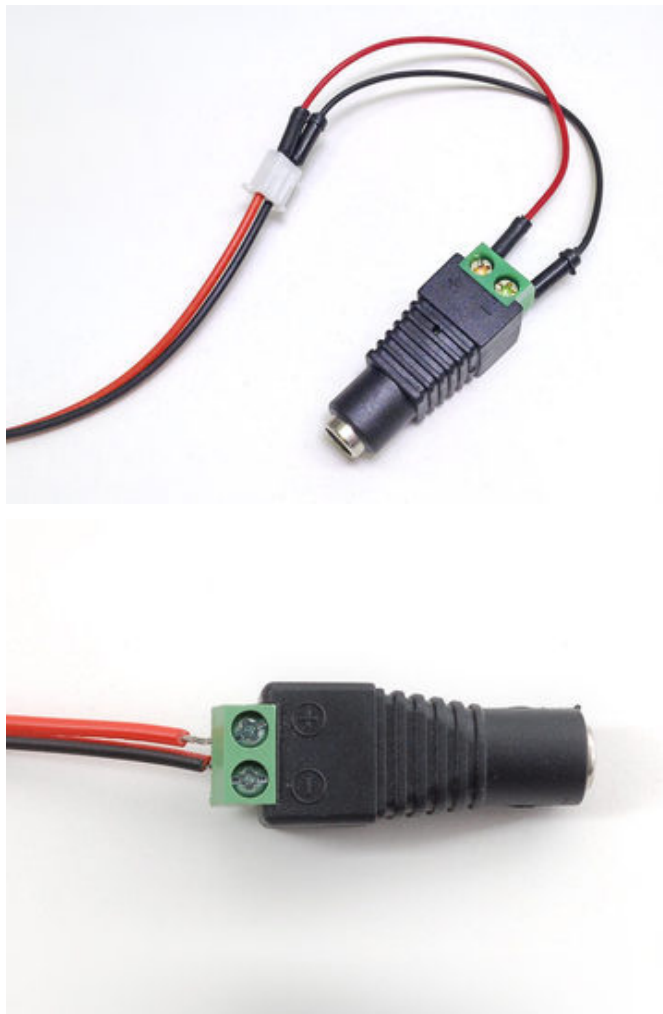
Set up any other options to your liking, click “OK” and then reboot when prompted.



After rebooting, you can set up networking. Ethernet is usually a simple matter of plugging in a cable. Wireless networking is more involved and beyond the scope of this guide...you can search for other guides on this topic, or many networks are easily accessed by clicking the network icon in the status tray at the top-right of the screen.

## Connect and Configure Printer

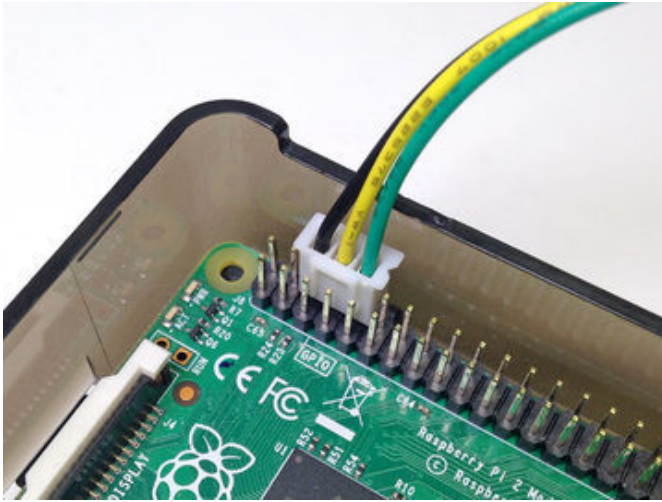
---



All the Adafruit thermal printer varieties are bare units; they don't have a DC barrel jack for power. Use a [Female DC Power Adapter \(http://adafru.it/368\)](http://adafru.it/368) to connect to a [5V 2A power supply \(http://adafru.it/276\)](http://adafru.it/276) (our [5V 4A \(http://adafru.it/1466\)](http://adafru.it/1466) and [10A \(http://adafru.it/658\)](http://adafru.it/658) supplies can also work, if you already have one around, but *do not* use a 12V adapter).

During initial testing, you can press a couple of breadboard jumper wires into the printer's JST power plug; the red wire is +, black wire is -. This is not very durable though...once you're confident everything is working, you can remove the wires, clip off the JST plug, strip the ends and screw them directly to the adapter's terminals.

3.3V	□	□	5V	
GPIO2	□	□	5V	
GPIO3	□	□	GND	← To GND on printer
GPIO4	□	□	TXD	← To RX on printer
GND	□	□	RXD	← To TX on printer
GPIO17	□	□	GPIO18	
GPIO27	□	□	GND	
GPIO22	□	□	GPIO23	
3.3V	□	□	GPIO24	
GPIO10	□	□	GND	
GPIO9	□	□	GPIO25	
GPIO11	□	□	GPIO8	
GND	□	□	GPIO7	
DNC	□	□	DNC	
GPIO5	□	□	GND	
GPIO6	□	□	GPIO12	
GPIO13	□	□	GND	
GPIO19	□	□	GPIO16	
GPIO26	□	□	GPIO20	
GND	□	□	GPIO21	



**For TTL printers (not USB):** connect GND, RX and TX to the Raspberry Pi GPIO header in the positions shown. The transmit/receive lines will cross: the TX pin on the printer connects to RXD on the Pi, RX on the printer to TXD on the Pi.

If using the “Mini” Thermal Receipt Printer, the data plug from the printer can fit directly over these pins. Handy!

The “Tiny” and “Nano” printers (and “Printer Guts”) have a single 5-pin connector for power and data. Use something like [F-M jumper wires \(http://adafru.it/825\)](http://adafru.it/825) or a [Pi Cobbler \(http://adafru.it/2029\)](http://adafru.it/2029) and breadboard to make these connections. You’ll also need to split GND from the power supply to both the printer and the Pi GND pin.

The “Tiny” printer also features **USB** — that’s much easier if you have a port available!

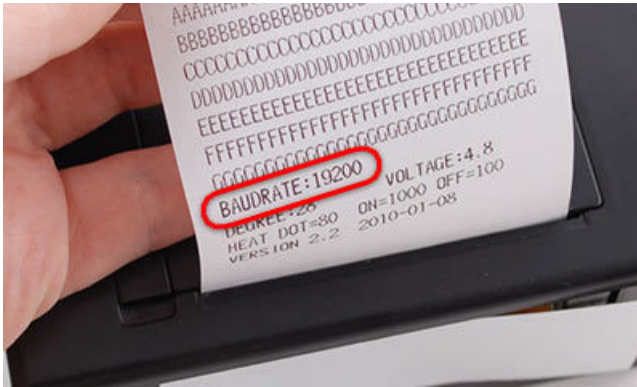
It’s perfectly safe to connect the TX/RX directly to the Raspberry Pi GPIO pins. Though these printers are sometimes called “5V TTL,” in reality the controller logic all runs at 3.3 Volts...the inputs are simply 5 Volt *tolerant* so they can work with Arduino and similar.

If using a **TTL** printer and it starts printing a bunch of junk, disconnect power immediately! Return to the rasi-config utility and **disable the serial console**, then reboot.

Your thermal printer may have arrived with a **test page** in the box or the paper bay. If not, or if you threw that away, you can generate a new one by installing a roll of paper and holding the feed button (on printers that have one) while connecting power, or tapping the button on the back of the “Nano” printer or the “Printer Guts.”

Look for the **baud rate** that’s printed near the bottom of the page. This is typically either **9600** or **19200** baud. This is important...you’ll need to know the correct value for your printer later.





(Don't fret if your printer is "only" 9600 baud. This has *no impact* on the speed of printing! The paper feed rate is really the bottleneck...the baud rate just determines how we'll get the Pi and printer communicating.)

Once your printer's connected, powered up and not spouting junk, launch a Terminal window and we'll enter a few commands...

For a **TTL** (not USB) printer, type the following (**substitute the correct baud rate** for your printer):

```
stty -F /dev/serial0 19200
echo -e "This is a test.\n\n\n" > /dev/serial0
```

(In older versions of Raspbian we'd use `/dev/ttyAMA0` instead. If the above commands present a problem, check what's in the `/dev` directory and use `serial0` or `AMA0` as necessary.)

For a **USB** printer, type:

```
echo -e "This is a test.\n\n\n" > /dev/ttyUSB0
```

**If your printer outputs only gibberish:** check the baud rate and try again. Substitute 9600 or 19200 as needed.

Sometimes there's a *little bit* of gibberish at the start of the first line printed; some residue in the serial buffer. This won't happen on subsequent lines or after we're done fully configuring the printer.

**If nothing happens**, check the following:

- Is the printer connected to a 5V power source, 2A or larger? You should see an occasional LED blink from the printer.
- Is thermal paper installed in the bay?
- For TTL printers: are the TX/RX pins connected between Pi and printer, and do they cross? (i.e. TX on printer should go to RXD on Pi.) Is GND connected between printer and Pi?

## Commence Major Installation!

Once the printer passes the above basic test, it's time to move on to software.

Make sure the package database is up-to-date, then install several packages...

```
sudo apt-get update
sudo apt-get install libcups2-dev libcupsimage2-dev git build-essential cups system-config-printer
```

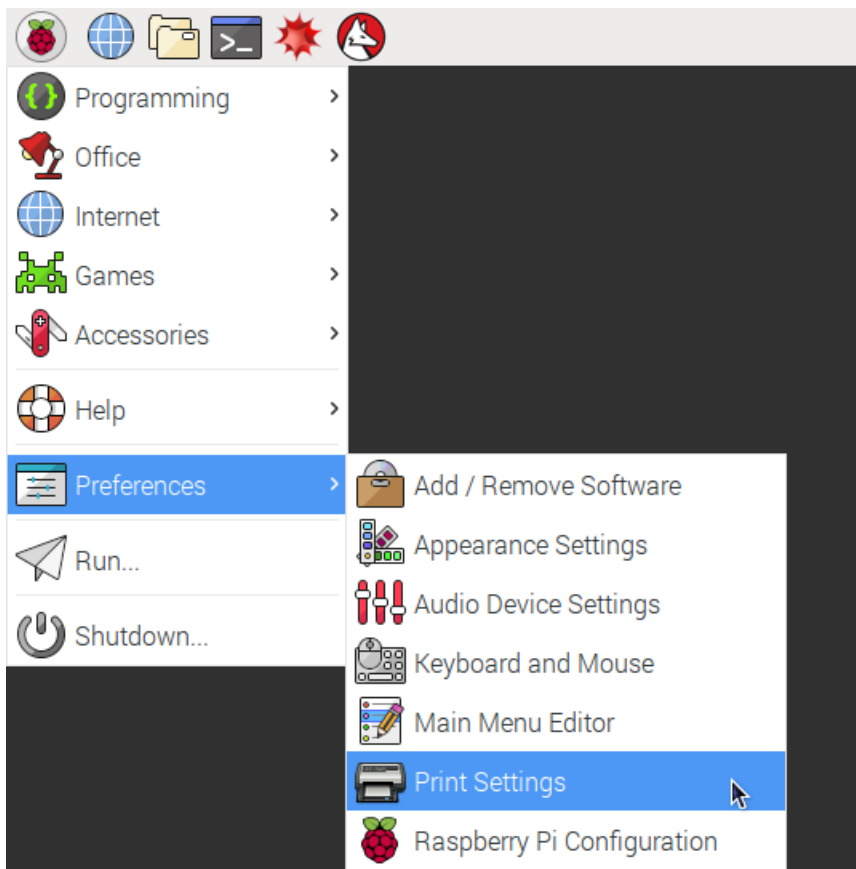
Though not unusually large (about 75 MB total), there are many small pieces to process. This could take about 30 minutes to install, so you might want to go mow the lawn or something.

Once that's done, download and install this CUPS filter for the thermal printer:

```
git clone https://github.com/adafruit/zj-58
cd zj-58
make
sudo ./install
```

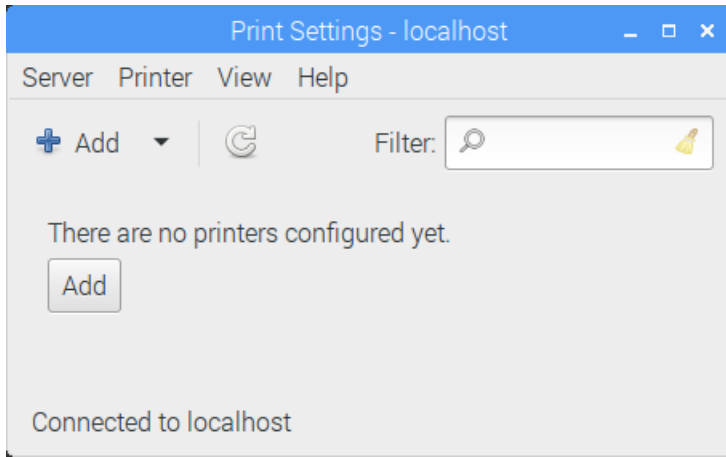
Now we'll switch over to using the GUI for the rest.

From the Pi menu, select Preferences→Print Settings.

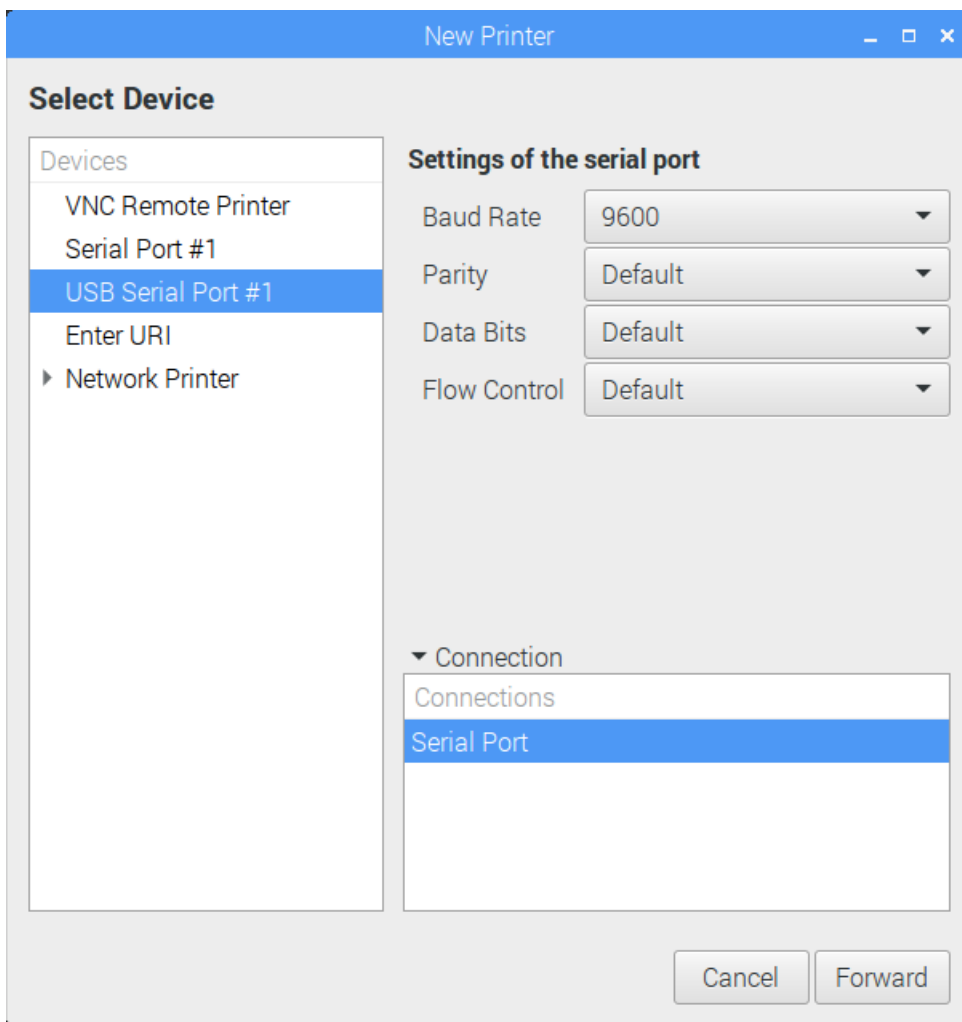


If the “Print Settings” menu option is not present, the ZJ-58 CUPS filter didn’t install. Check the above steps and try again.

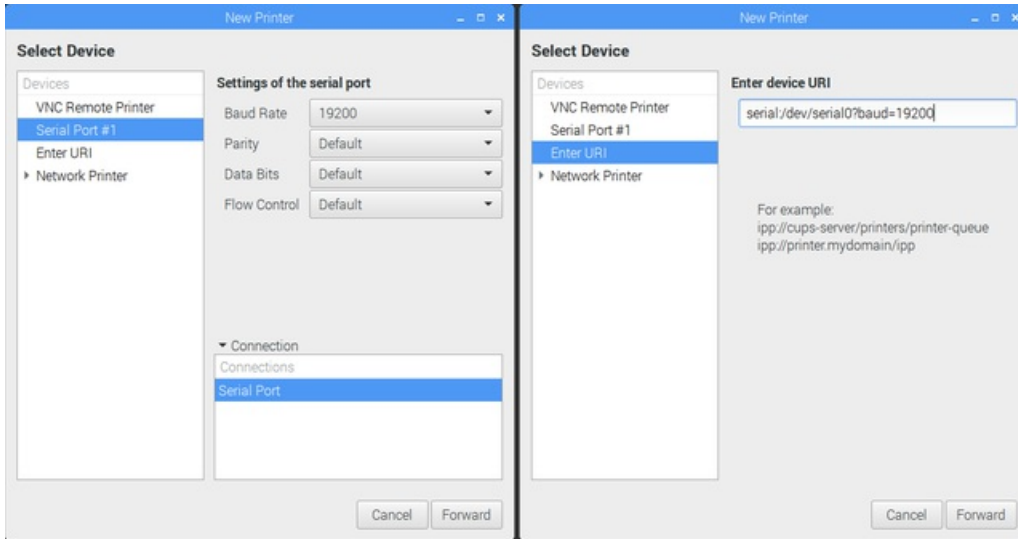
No printers have been set up yet, so click the “Add” button:



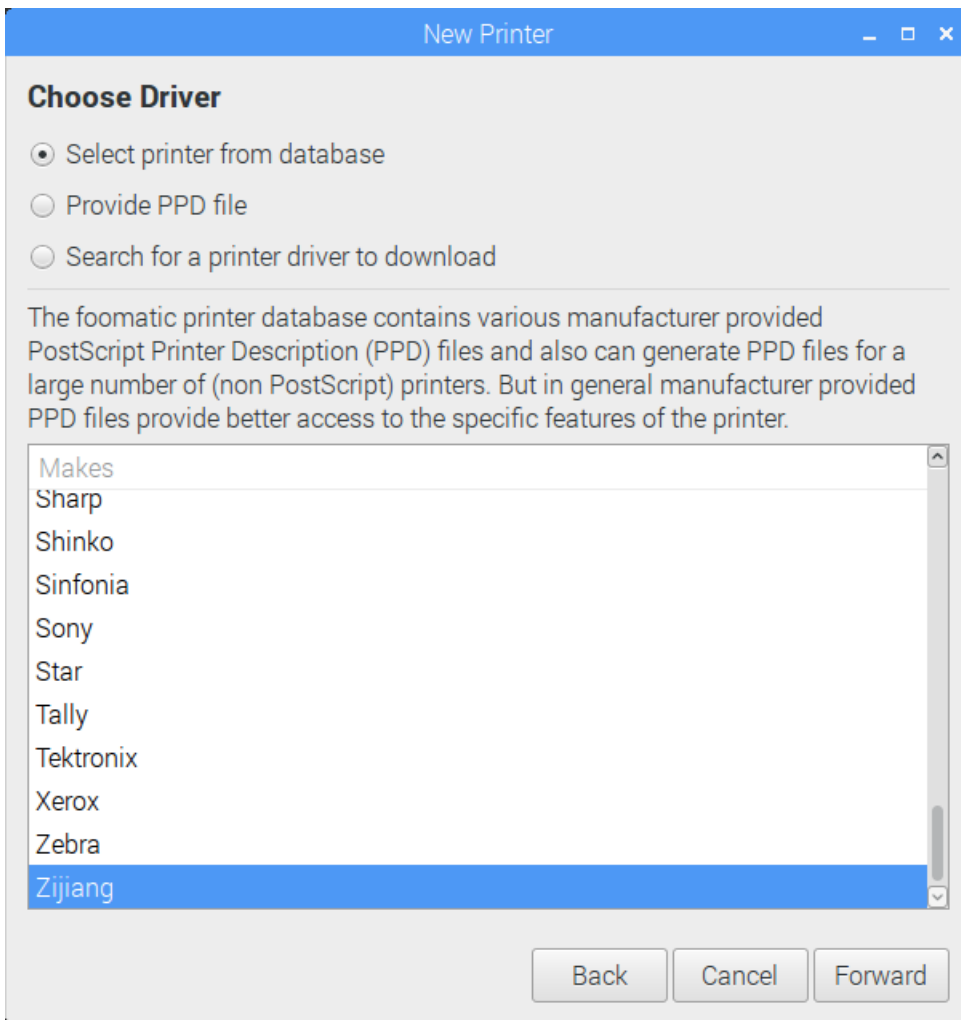
For a **USB printer**: you should see “USB Serial Port #1” in the Devices list. Highlight that item (*not* Serial Port #1, but “USB Serial Port #1” specifically), select the correct baud rate, then click the “Forward” button.



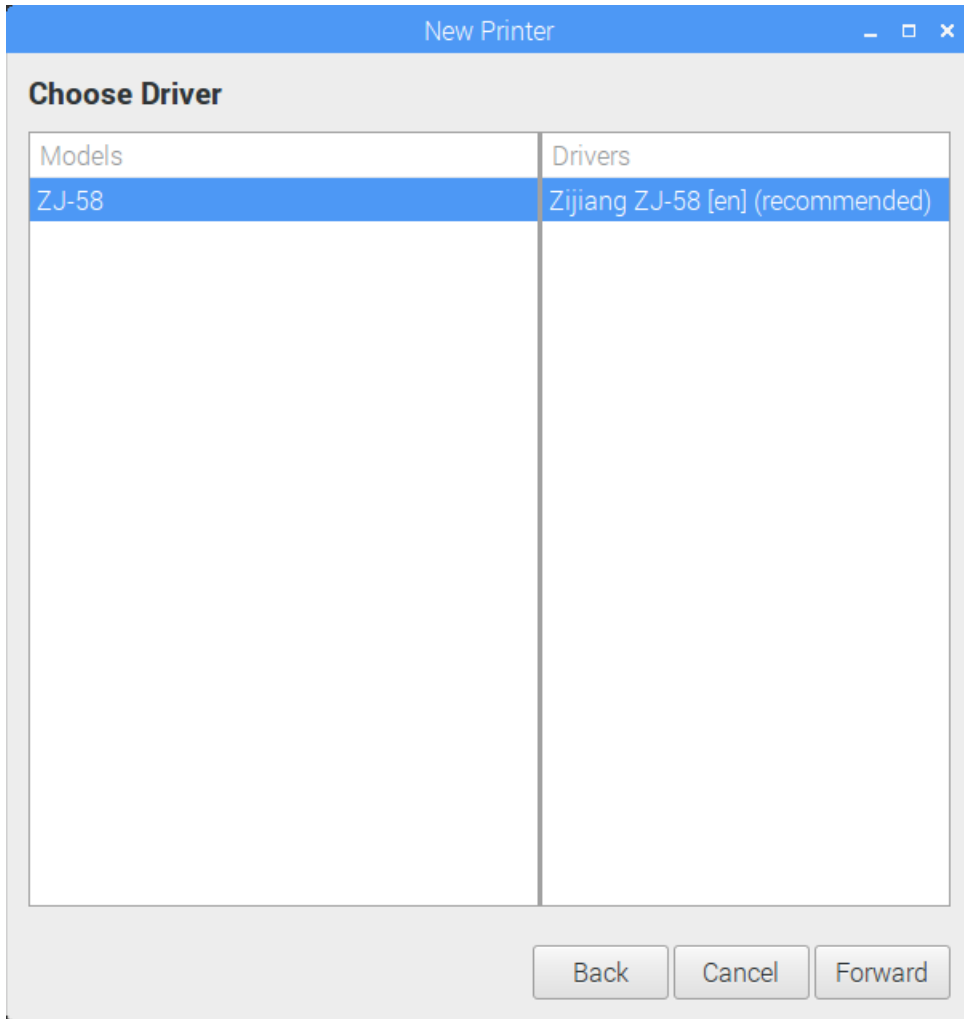
For **TTL printers**: If “Serial Port #1” is present, select that and set the baud rate. Otherwise (slightly older Raspbian OS), highlight “Enter URI” in the Devices list. Then in the device URI field, type “serial:/dev/serial0?baud=19200” (or 9600) (and might be “ttyAMA0” instead of “serial0”, depending on commands tested earlier). The device name is case-sensitive, so get this exactly right!



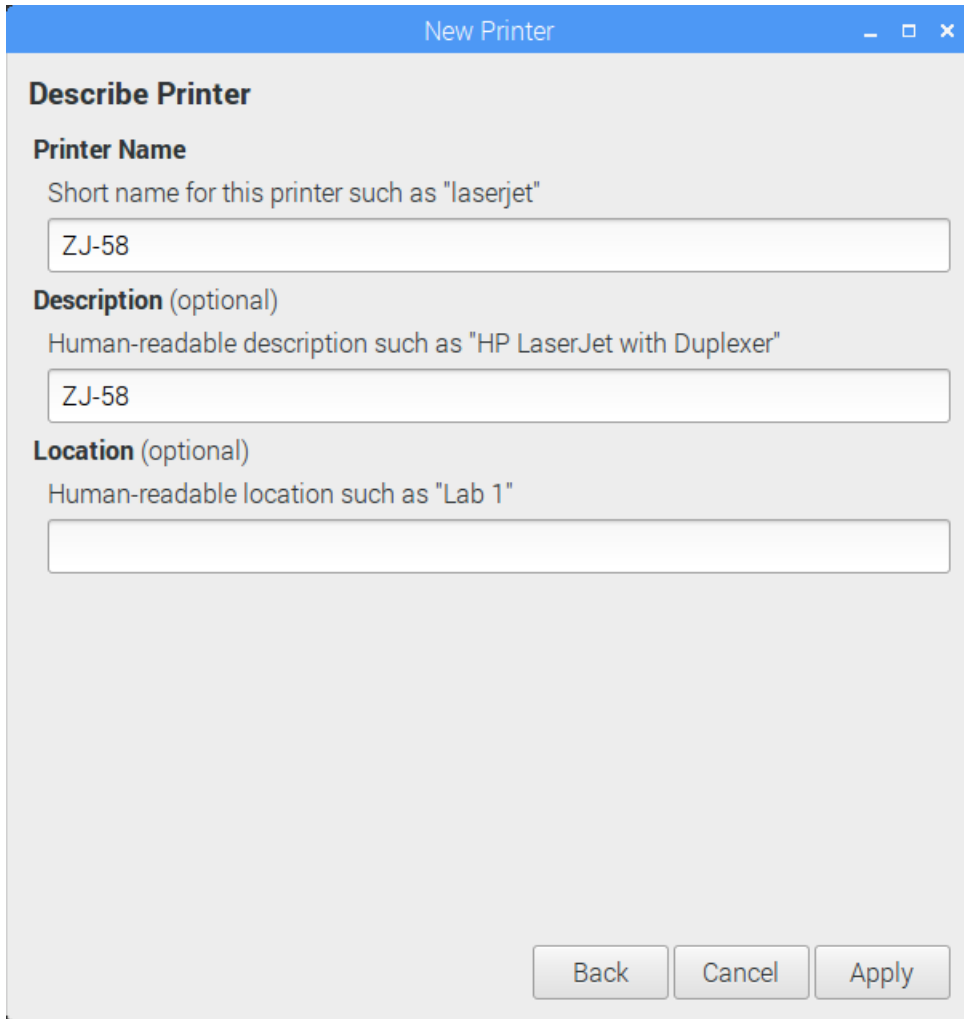
Click the “Forward” button. On the next form, click “Select printer from database,” then scroll to the very bottom of the list and select “Zijiang” — this is a brand of thermal printer that happens to use the same command set.



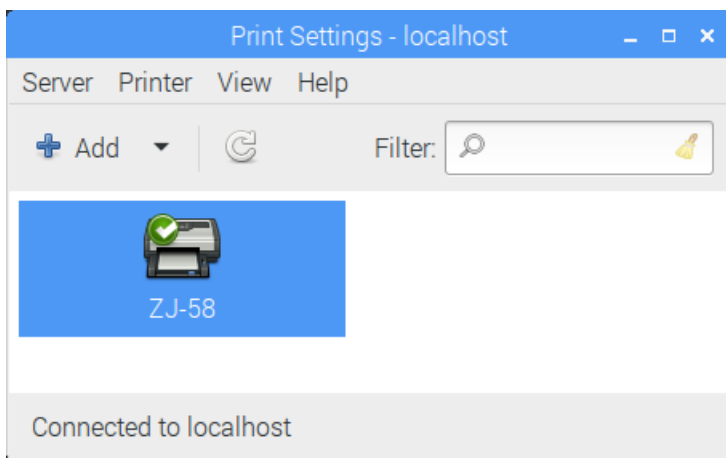
Click “Forward” again, and on the next form select the “Zijiang ZJ-58” driver:



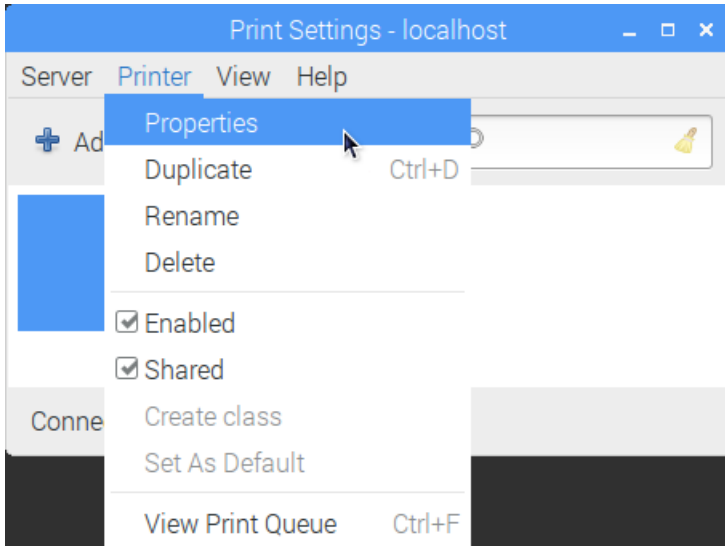
“Forward” once more, then you’ll have the option of assigning your printer a name...or just stick with the default ZJ-58:



Finally, click “Apply” and you’ll now see the ZJ-58 printer in the Print Settings list:

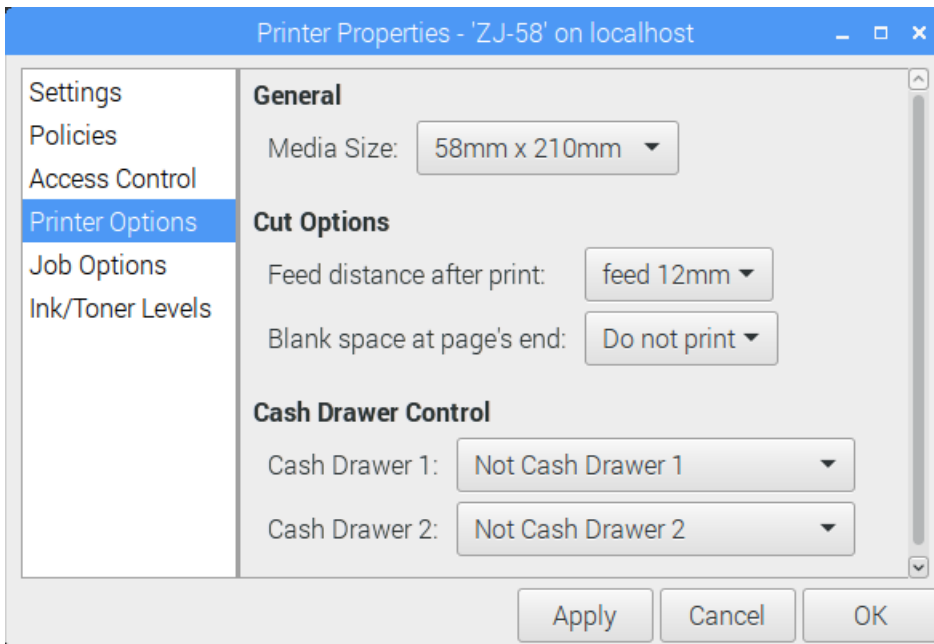


Click the ZJ-58 icon, then from the “Printer” menu select “Properties”:



Set “Feed distance after print” to “feed 12mm,” then click OK.

Your mileage may vary; if you want more or less paper feed after each print, this is where you’ll find it later.



After clicking “OK,” the printer should now fully work with the Raspberry Pi via the “lp” or “lpr” services.

Let’s give it a try! Back to the Terminal window...

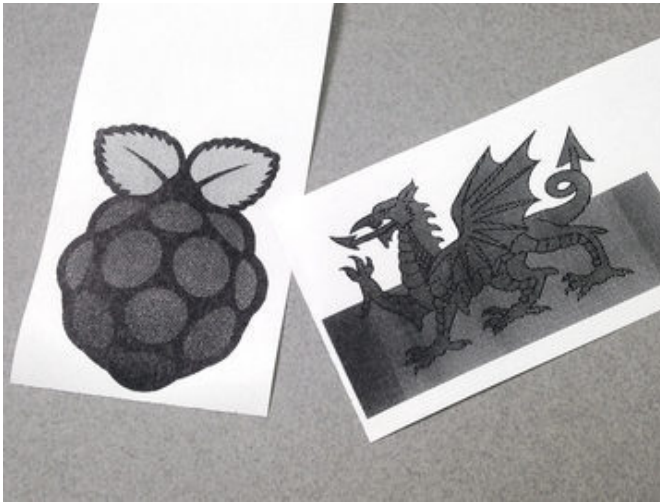
```
echo "This is a test." | lpr
```

This should print the “This is a test.” string again...but unlike the first test, you’ll notice this one isn’t using the printer’s internal font...it’s actually rasterizing the text and we can tweak settings such as size.

You can try a printing bitmap as well:

```
lpr -o fit-to-page /usr/share/raspberrypi-artwork/raspberry-pi-logo.png
```

“-o fit-to-page” is important — most graphics, if printed “actual-size,” would print just a narrow strip cropped from the left side.



Most raster formats are supported, and for vector there’s PDF.

**SVG is not directly supported.** If you try to print an SVG file, you’ll see a bunch of text instead! You can use the LibreOffice Draw application included with Raspbian Jessie to open SVG files and save them as PDF for printing.

The Raspberry Pi logo shown is from a raster PNG file, you can see it’s a bit “dithery.” The dragon was an SVG file converted to PDF...this uses better halftoning and is amazingly sharp.

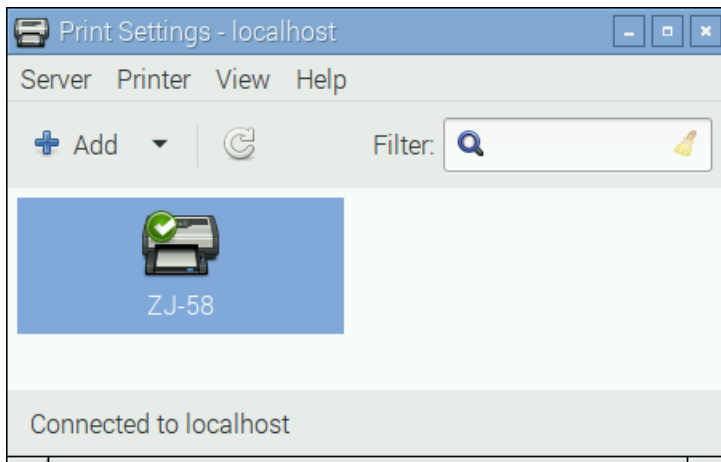
CUPS is a complex system way beyond the scope of this guide (there are entire books on the subject), but [the CUPS web site](https://adafru.it/mVb) explains some command-line options to get you started (<https://adafru.it/mVb>).



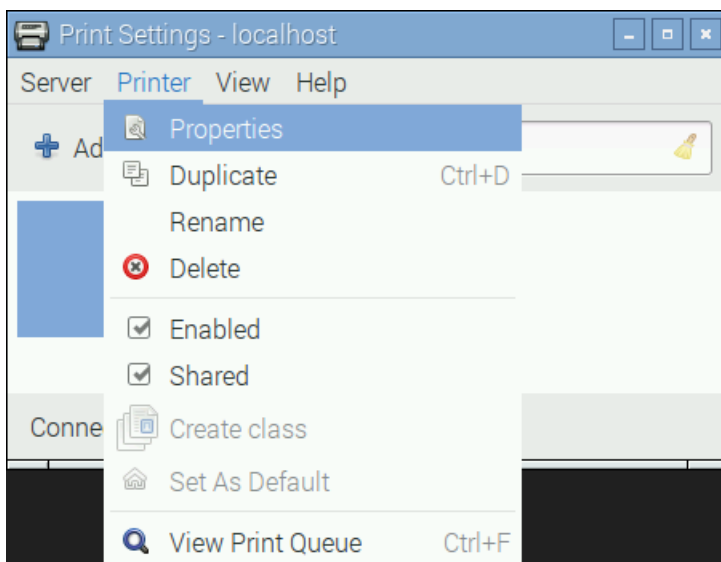
## Network Printing

Accessing the printer from other systems on the same network requires just a few extra clicks...

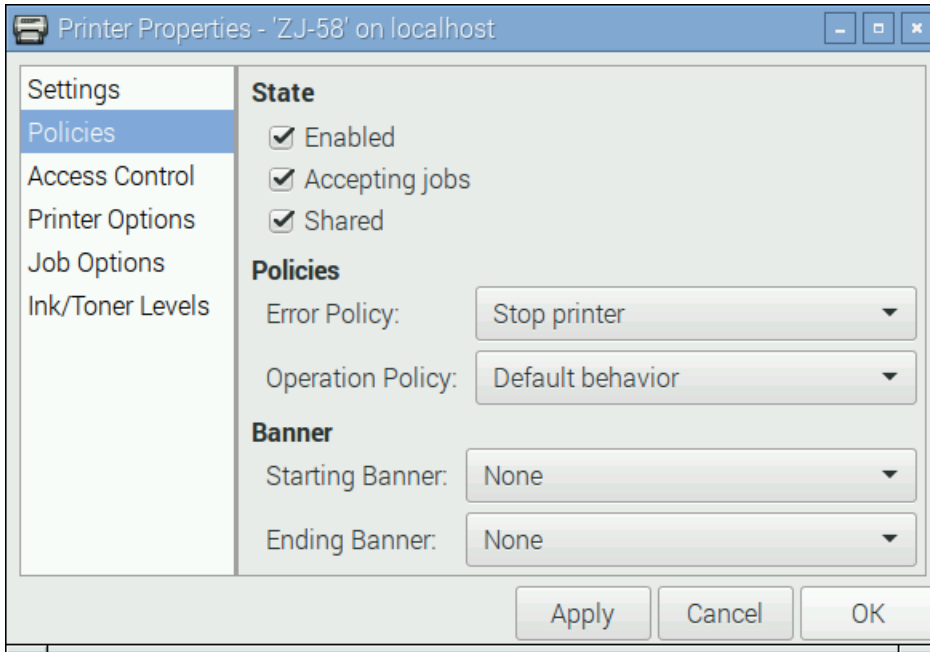
From the Pi menu, select Preferences→Print Settings.



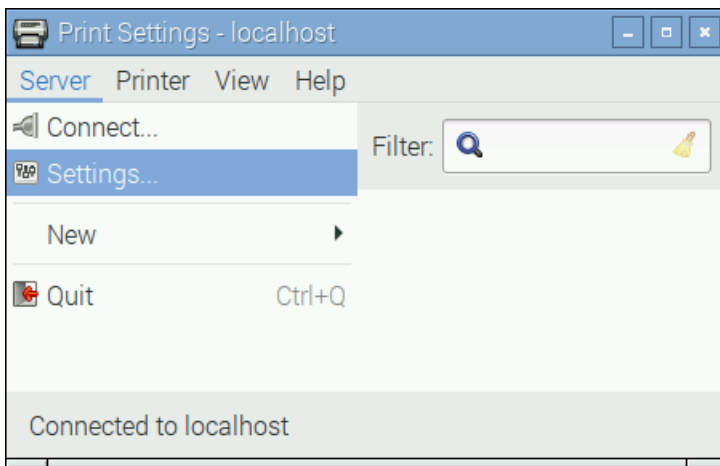
Double-click the ZJ-58 printer, or highlight it and select Printer→Properties:



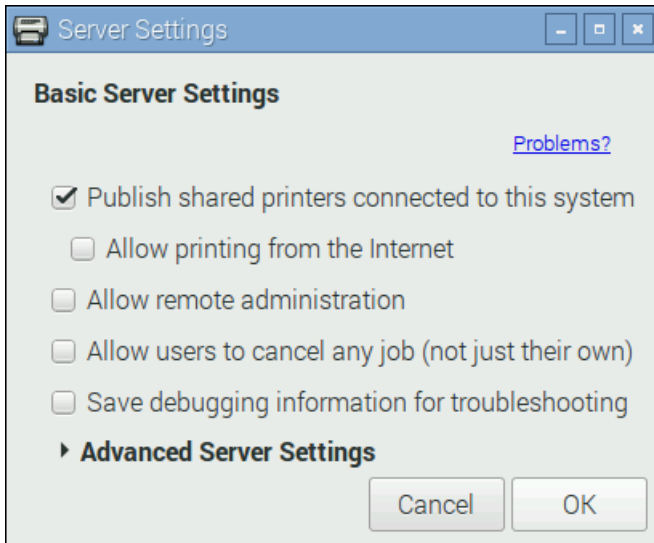
In the “Policies” section, make sure “Shared” is checked, then click OK.



Returning to the main Print Settings form, now select Server→Settings...

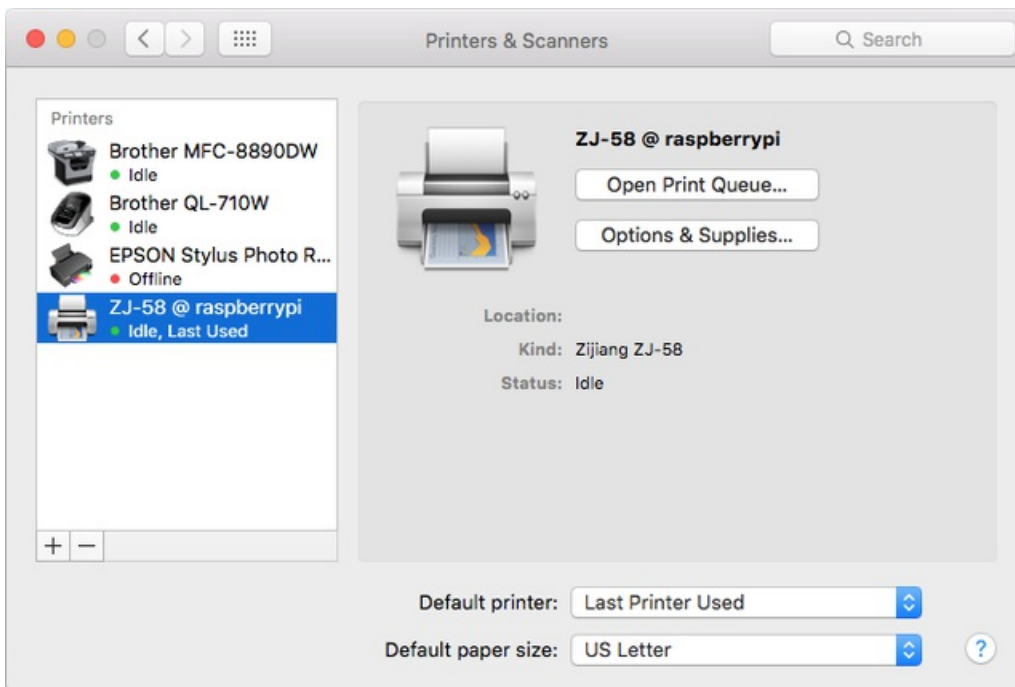


Check the “Publish shared printers connected to this system” box, then click OK.



The printer is now accessible to other systems...even non-Linux computers running Windows or Mac OS X.

For example, on a Mac, go to the “Printers and Scanners” system preference form, click the add (+) icon, and you’ll see the networked printer available there:



It’s then possible to print to the thermal printer from any application.

Due to the peculiar media size, this will require some trial and error and experimentation. Though the media size is described as “58mm” wide, the actual imageable area is more like 48mm. If you try to print any “normal size” artwork, or if Media Size is set to “Letter,” you’ll just get a narrow strip from the left edge of the page.

