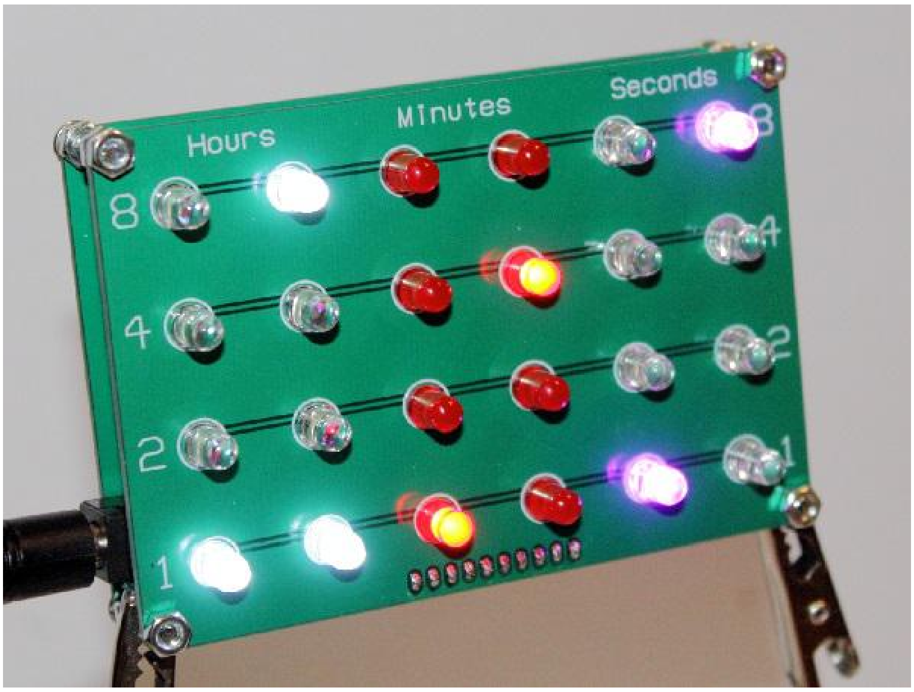
Raspberry-Pi Shield: Binary-Coded-Decimal Clock



ASSEMBLY INSTRUCTIONS

What is it?

This kit builds a binary-coded-decimal clock, driven by a Raspberry-Pi (which should be mounted on the back). This is a great kit if you want to learn to read binary at speed, love unusual clocks … or just want to impress/infuriate your co-workers!

The shield uses the time maintained on the Raspberry-Pi … so if you configure it to set itself from the Internet (usually with NTP, the “Net Time Protocol”) then this clock becomes self-setting and is highly accurate. It's then up to you whether you use a Wifi-enabled Pi to connect to your home Wifi network, or use a cat-5 Pi to connect via a LAN cable.

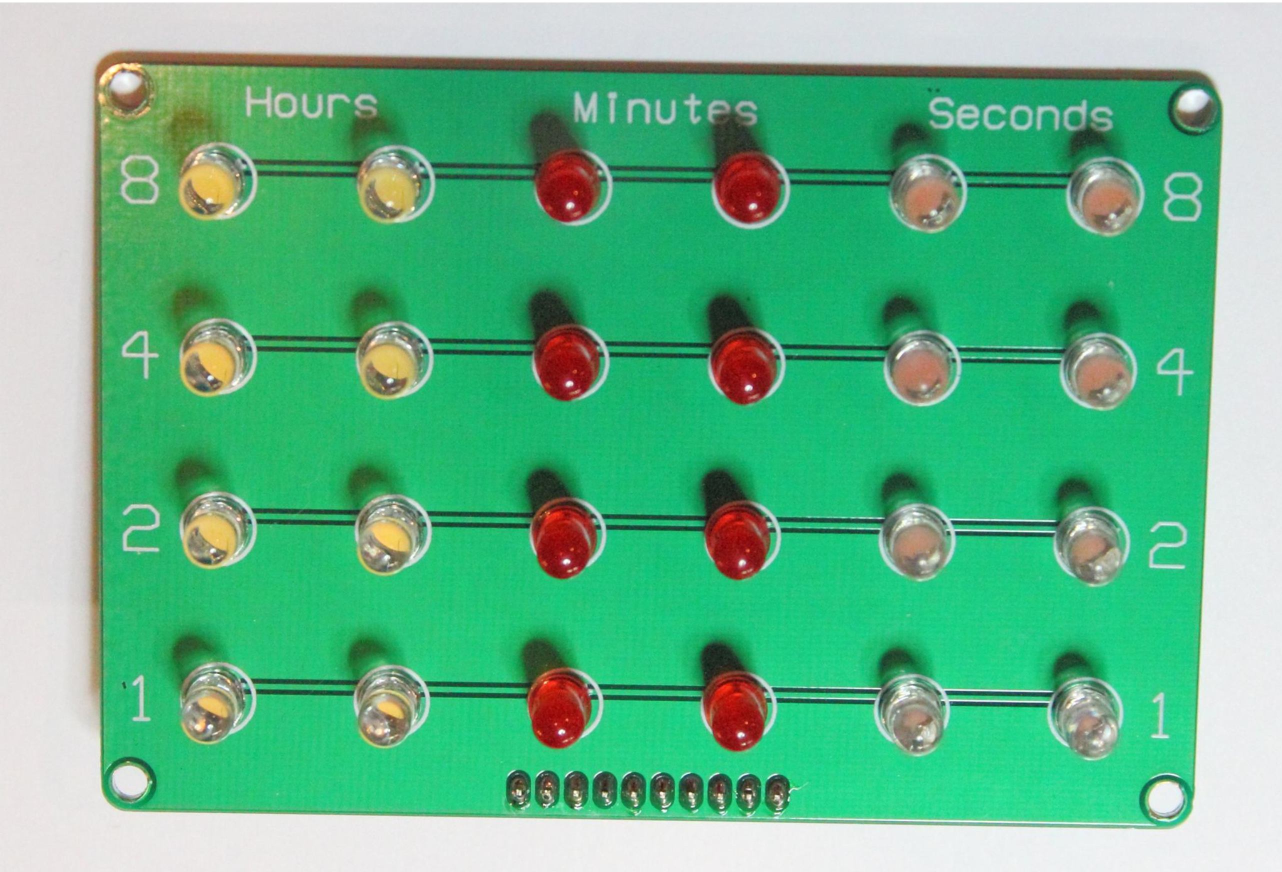
Assembly overview

The clock is made up of two PCBs. The front PCB is the display, with a 6x4 matrix of LEDs. The rear PCB has the driving electronics, and is where the Raspberry-Pi is mounted.

The PCBs are fitted back-to-back, connected via a ten-way connector (pins on one PCB, sockets on the other). The PCBs are held together with a brass standoff at each corner; each standoff is held in place with two M3 bolts.

Assembling the front PCB

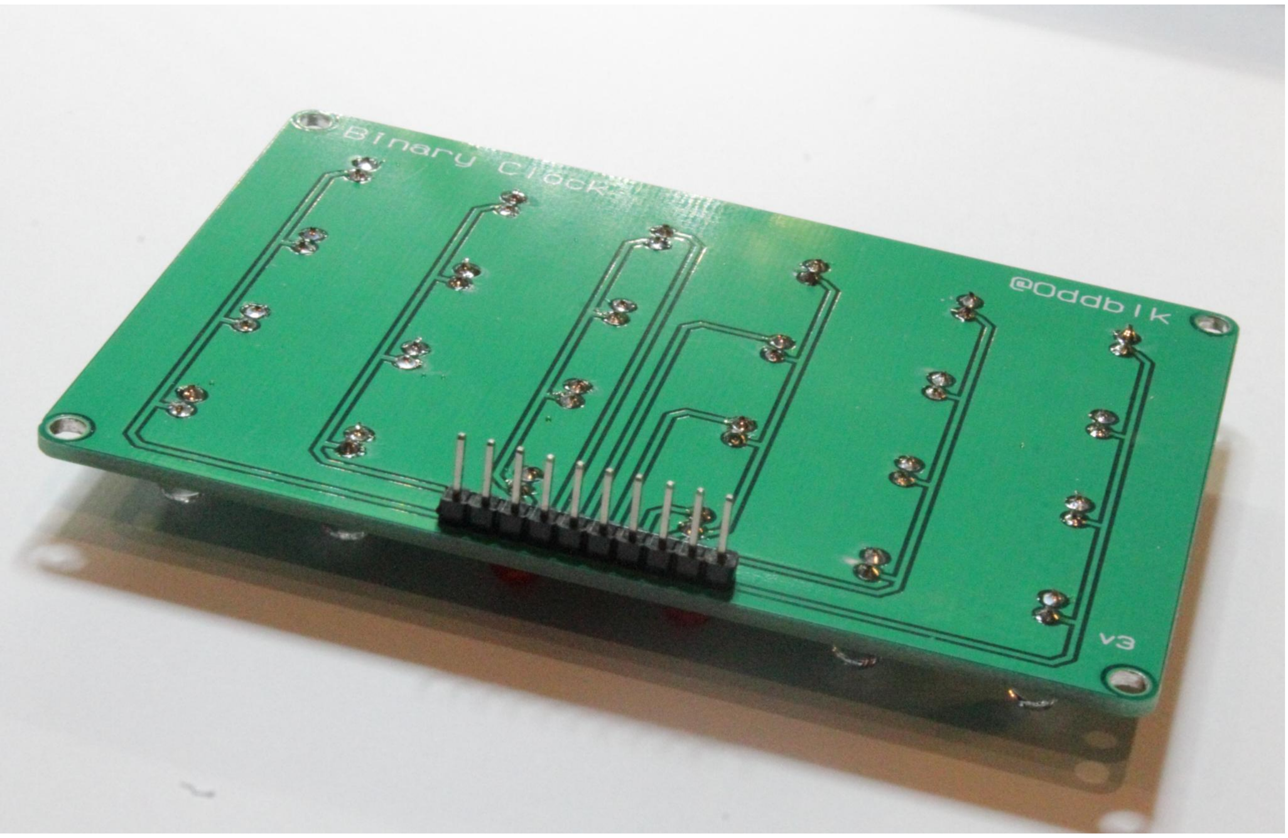
Spend a bit of time thinking about your colour-scheme. I normally choose a colour for each pair of columns: it helps "group" columns together and makes it a bit easier to speed-read. I've also found that just using a single colour works well ... but it's entirely up to you!



Note that LEDs must be soldered the correct way around. If you look at the silk screen (the white markings on the PCB) you'll see that each LED has the positive pad marked with a "+" symbol. The **longer** of the LEDs two legs should go in here.

Push the LEDs in from the front, solder on the back. After you've soldered each LED, trim the excess legs off with a pair of flush-cutters.

The ten way header-pin should obviously point the opposite way to the LEDs: pushed in from the back, soldered from the front.



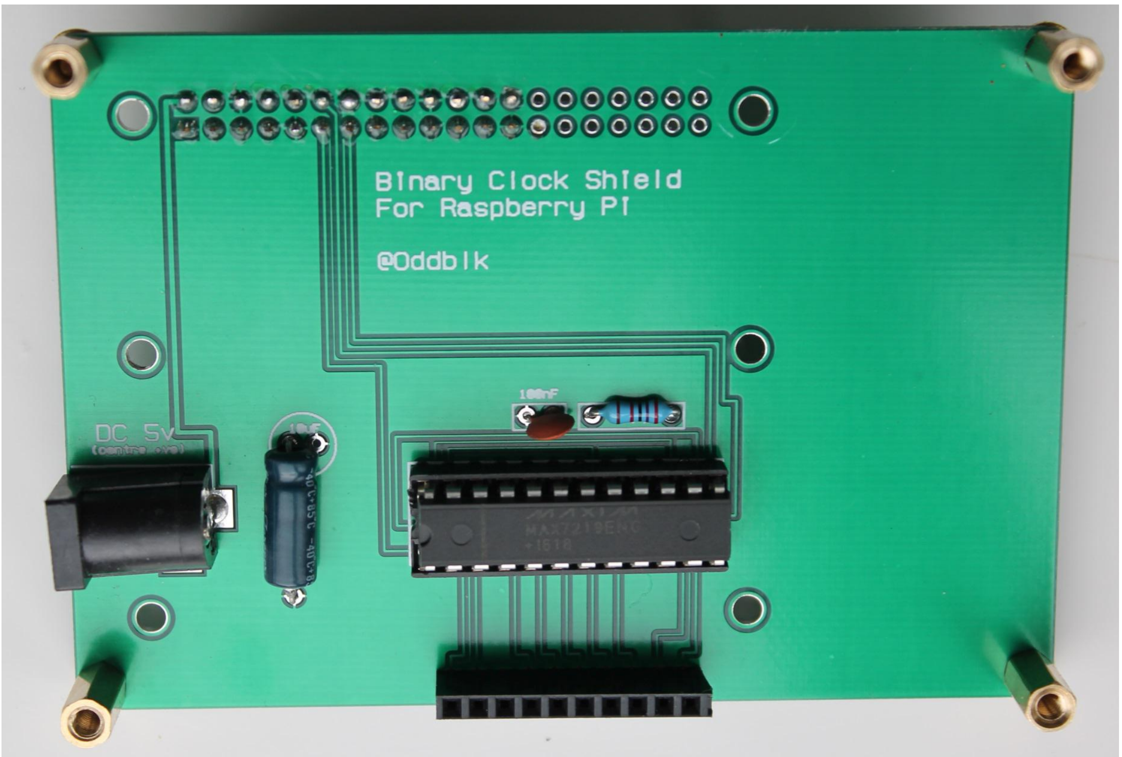
The front PCB, shown from the rear.

Assembling the rear PCB

I've designed the rear PCB so that most of the components go between the two PCBs, leaving the other side just for the Raspberry-Pi. To make it clearer, the silk-screen markings denote which side the components should go.

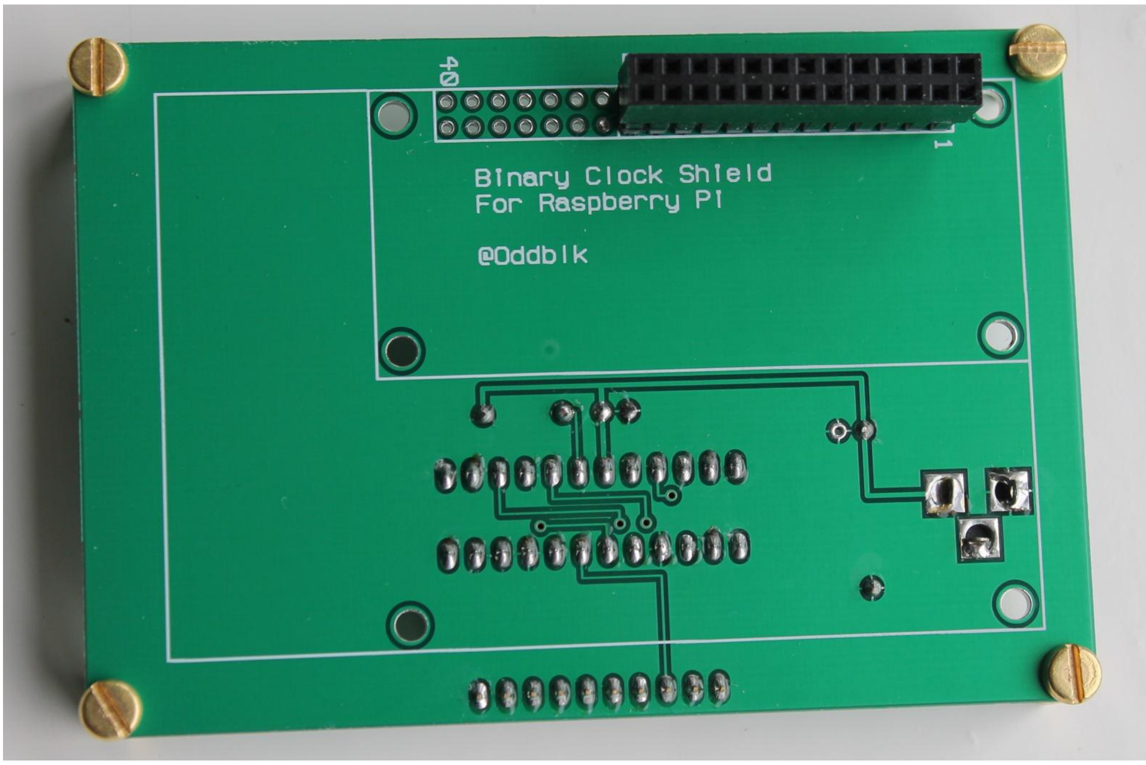
The circuit is nice and simple … but then I *would* say that, wouldn't I? There's a barrel connector, a 10uF capacitor, a 100nF capacitor, a 27K-ohm resistor, and the IC socket for the MAX7219.

Note that I've made provision for **both** types of 10uF capacitor: axial (one leg at each end) and radial (both legs at one end). I try my hardest to provide an axial capacitor with this kit, because that way the capacitor lies flat along the board (there's lots of room). If your kit comes with a radial capacitor, then you'll need to lie it flat against the board.



Note the standoffs at the corner and the 10-way connector at the bottom. The front PCB sits on top of this.

And on the other side of the rear PCB:



The Raspberry-Pi is mounted on this side. I supply a 40-way connector and 11mm standoffs for people who are going to use a Raspberry-Pi **Zero**, or a 26-way connector and 16mm standoffs for any of the other models of Raspberry-Pi. Mounting holes are on the board to secure your Raspberry-Pi … but note that the very early models of Raspberry-Pi didn't have mounting holes designed onto them!

Connect the PCBs together

Use a brass standoff at each corner, with M3 bolts to hold them in place.

This completes the assembly! See separate document (or my blog) about how to install the software onto a Raspberry-Pi Linux distro.

Power requirements

Raspberry Pis can be quite choosy about the minimum voltage they require to run **reliably**. It should be a minimum of 5v, but preferably 5.1 or 5.2v. Also, the current requirements vary between models of Pi (some of the older models might need over 500mA, but the Zero only needs about 150mA). For this reason, I stronly recommend you find a power supply that has been recommended for use with Raspberry Pis.

For example: my development PC seems to supply around 4.9v to the USB sockets, and this caused my clock to suffer frequent clashes. But since I moved it to a better quality power supply, the performance has been faultless.

Mounting the screen, cases, cradles ...

That's all entirely up to you! I've bought an adjustable mobile phone cradle that clips to my desk. Or, if you're the creative sort you could make yourself a wood or acrylic case, or 3D-print a pair of comedy clown shoes to stand it upright, or hold it with one of those crocodile-clip "helping hands" - whatever works for you! Please send me pictures of whatever you make!

A PDF with the measurements of the front PCB (including LED spacing and mount points) is available from my website in case you find that helpful. Please also visit my website for schematics, information about how to read the clock, and for any firmware updates.

Have fun!

For more information …

Please visit my blog page about it all, here:

`http://danceswithferrets.org/geekblog/?page_id=1034`