

Piggyaxe Binary Timer Kit



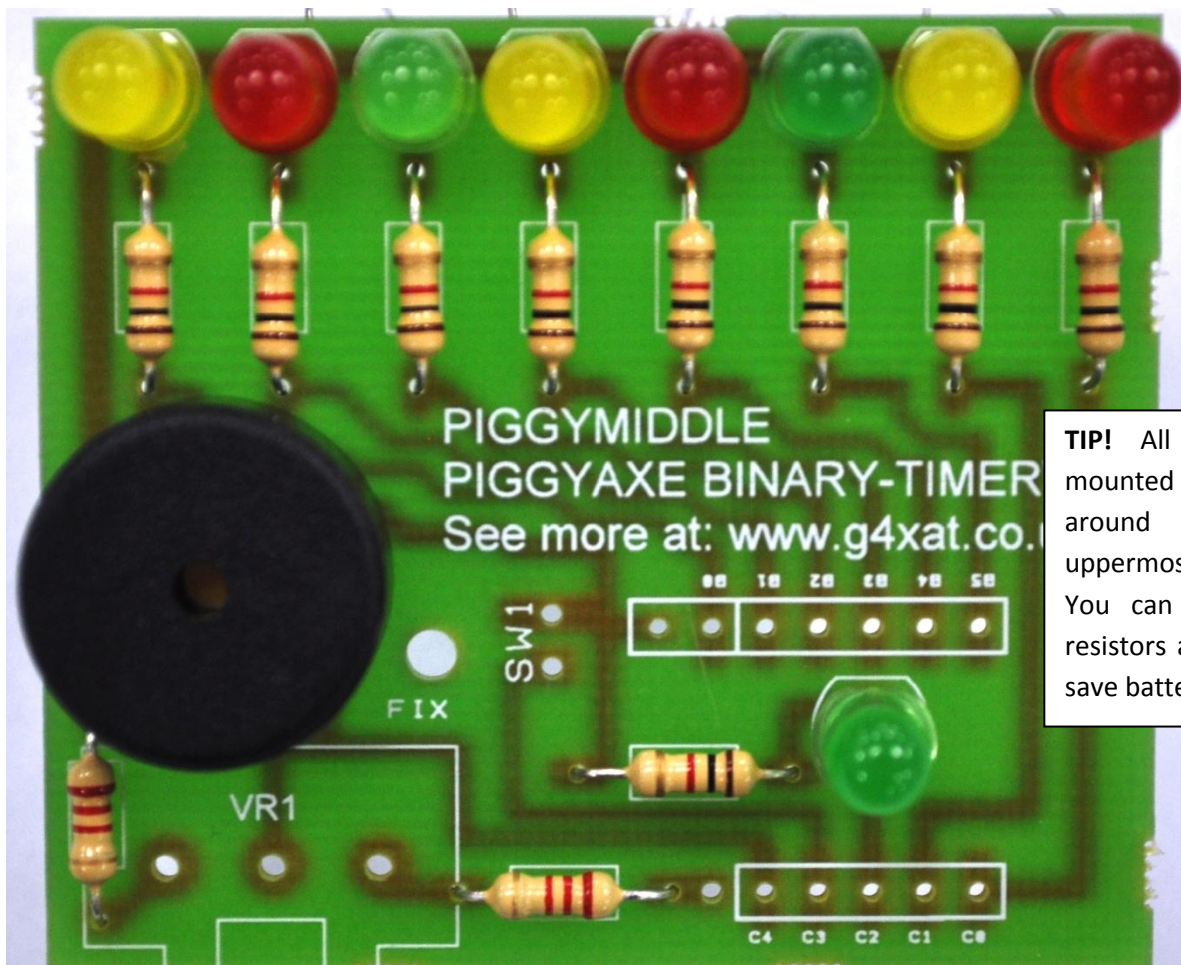
Electronics is FUN – so let's build a PIGGYAXE Binary Timer!

Start by collecting the following parts (Available as a Rapid Electronics kit, order code 70-1078):

Binary Timer PCB, R1 to 9: 1k Ω resistor (brown-black-red-gold), R10 and 11: 2.2k Ω resistor, 9 LEDs (kit contains red), Piezo sounder, 470k Ω linear potentiometer and matching knob, push-to-make switch.

You will also need: a soldering iron with a stand and a wet sponge, a PCB holder of some sort, some solder and a pair of side cutters. Remember: soldering irons can burn you. You should always wash your hands after handling solder.

Take great care to fit the components exactly where they are supposed to go, otherwise your circuit may not work as expected. Use the photograph below to help you place the components correctly.



TIP! All the LEDs are mounted the same way around – flat side uppermost in this picture. You can also make the resistors a higher value to save battery power.

Put a tick ✓ in each box as you solder in each part, or if you prefer, get someone to check your placement before you solder it. **CHECK TWICE – SOLDER ONCE!**

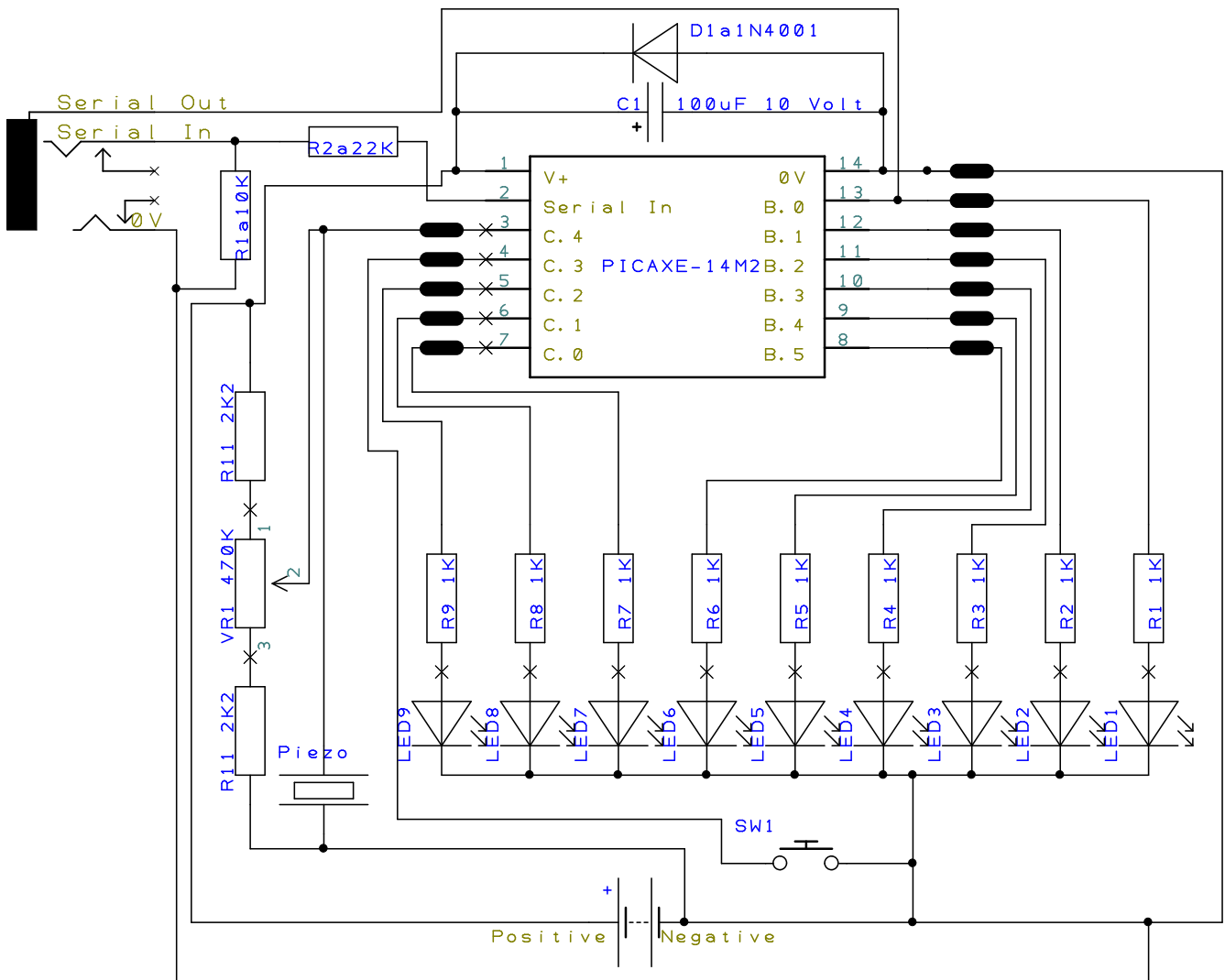
Resistors R1 to R8 R10 to 11 Piezo sounder C2 LEDs 1 to 8 LED 9

Next fit the header pins, 1 row of 6, one row of 7 and 18 individual ones

Depending on how you envisage the final product, the push-to-make switch can be mounted on wires and taken through the 'fix' hole to provide some stress relief. The potentiometer can also be mounted in a variety of ways to increase flexibility of product design.

Finally fit the control potentiometer to the PCB and solder carefully

Now carefully check your soldering for errors (missed joints or solder splashes). If it looks OK, the board may be piggybacked on to a PIGGYMIDDLE motherboard. The basic product can be used as a timer for 1 minute to 4 1/4 hours.



The diagram above shows the entire binary timer circuitry. This includes the 14M2 motherboard. Hence some components are numbered as R1a, etc.

Note that the Piezo element is driven by the same pin that is used to read the potentiometer used to set the time initially. The two 2.2kΩ resistors are there to 'pad' the potentiometer – no point in timing '0' minutes after all!! It also stops the potentiometer 'shorting out' the audio signal when the time is up if it is left at one or other extremes of travel. The relatively high resistance 470kΩ works fine and as it's wired across the power supply all the time when the unit is on, it helps reduce standing current. Equally, high value current limiting resistors for the LEDs (at 1kΩ) help reduce current draw. Simple as it is, it makes the basis of a great little project with times from 1 minute to 4 hrs 15 minutes possible and of course as many tunes as you want to write in and then select from. A basic functioning program gets you started. The only clever bit is how to output an 8-bit binary number to the non 8-bit port. Thanks to the wonderful Picaxe forum for the code on how to do that, and therefore making the whole project possible.