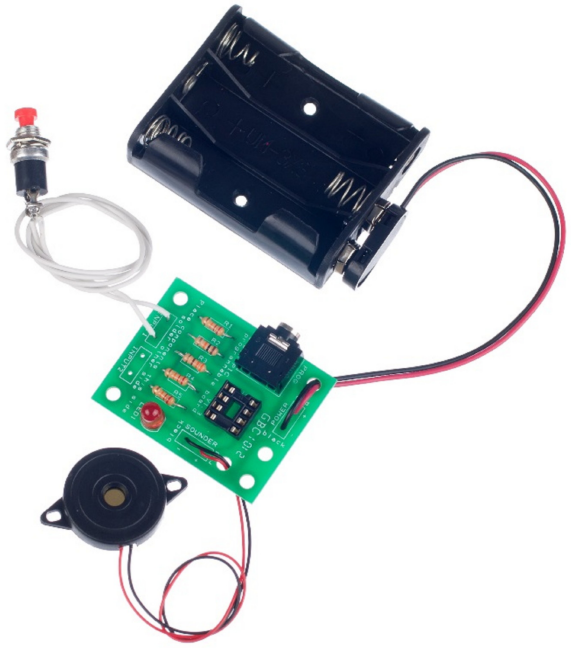
Kitronik

ESSENTIAL INFORMATION

BUILD INSTRUCTIONS
CHECKING YOUR PCB & FAULT-FINDING
MECHANICAL DETAILS
HOW THE KIT WORKS

CREATE YOUR OWN UNQUIE TIMER WITH THIS

PROGRAMMABLE TIMER KIT



Version 2.0

Programmable Timer Essentials

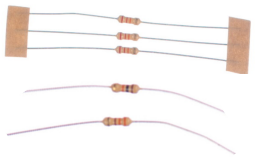
www.kitronik.co.uk/2121

Build Instructions

Before you start, take a look at the Printed Circuit Board (PCB). The components go in the side with the writing on and the solder goes on the side with the tracks and silver pads.

1 Place the resistors

Start with the five resistors:
The text on the PCB shows where R1, R2 etc go.
Ensure that you put the resistors in the right place.



PCB Ref	Value	Colour Bands
R1, R4 & R5	22kΩ	Red, red, orange
R2	10kΩ	Brown, black, orange
R3	330Ω	Orange, orange, brown

2 Place the IC holder

Solder the Integrated Circuit (IC) holder in to IC1. When putting this into the board, be sure to get it the right way around. The notch on the IC holder should line up with the notch on the lines marked on the PCB.

3 Place the programming connector

Solder the programming connector into the board where it is labeled 'PROG'.

4 Place the LED

Solder the Light Emitting Diode (LED) in to LED1. The timer won't work if it doesn't go in the right way around. If you look carefully one side of the LED has a flat edge, which must line up with the flat edge on the lines on the PCB.

5 Place the battery clip lead

The battery clip should be soldered into the 'POWER' terminal. First start by feeding the wire through the strain relief hole (feed from the solder side). The red wire must go to the '+' terminal (also marked 'red') and the black wire must go to the '–' terminal (also marked 'black').

6 Place the buzzer

The buzzer should be soldered into the 'SOUNDER' terminal. First start by feeding the wire through the strain relief hole (feed from the solder side). The red wire must go to the '+' terminal (also marked 'red') and the black wire must go to the '–' terminal (also marked 'black').

Build Instructions continued

7 Place the switch

Cut and strip two pieces of wire to the required length for connecting to the timer start switch. Solder one end of each wire to each of the terminals on the switch and the other end to the terminals labeled ‘INPUT1’. It does not matter which wire goes to which terminal.

8 Place the IC

The IC can be put into the holder ensuring the notch on the chip lines up with the notch on the holder.

Checking Your Programmable Timer PCB

Check the following before you insert the batteries:

Check the bottom of the board to ensure that:

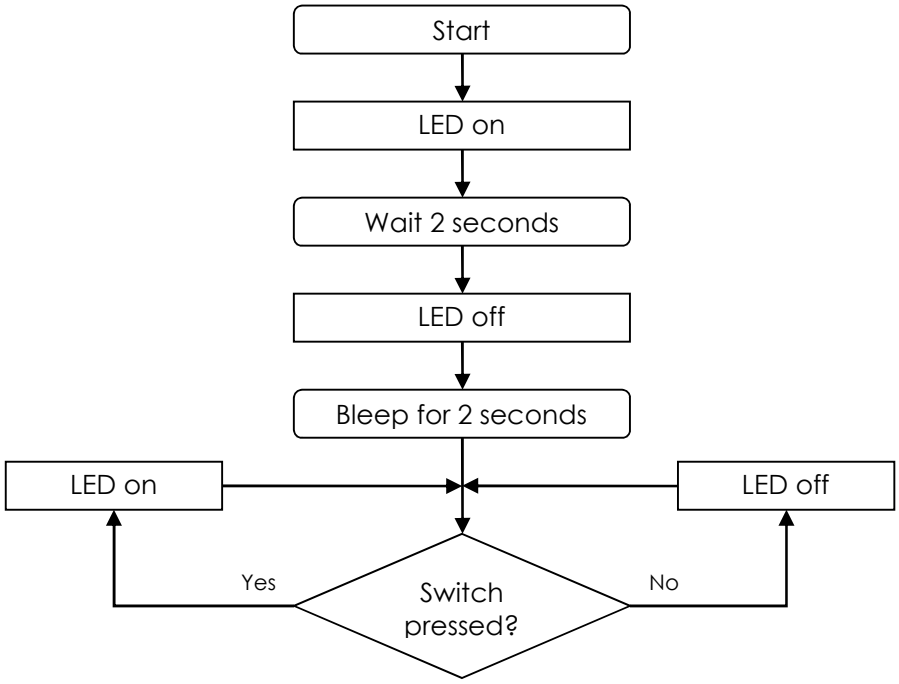
- All holes (except the 4 large 3 mm holes & INPUT2) are filled with the lead of a component.
- All these leads are soldered.
- Pins next to each other are not soldered together.

Check the top of the board to ensure that:

- The notch on the IC and the IC holder are in the same orientation as the markings on the printed circuit board.
- R2 has Brown, Black, and Orange colored bands.
- R3 has Orange, Orange, and Brown colored bands.
- The red wire on the battery connector goes to the + terminal on the power terminals and the black wire goes to the – terminal.
- The LED is in the right way around (the flat edge on the LED matches the markings on the board).

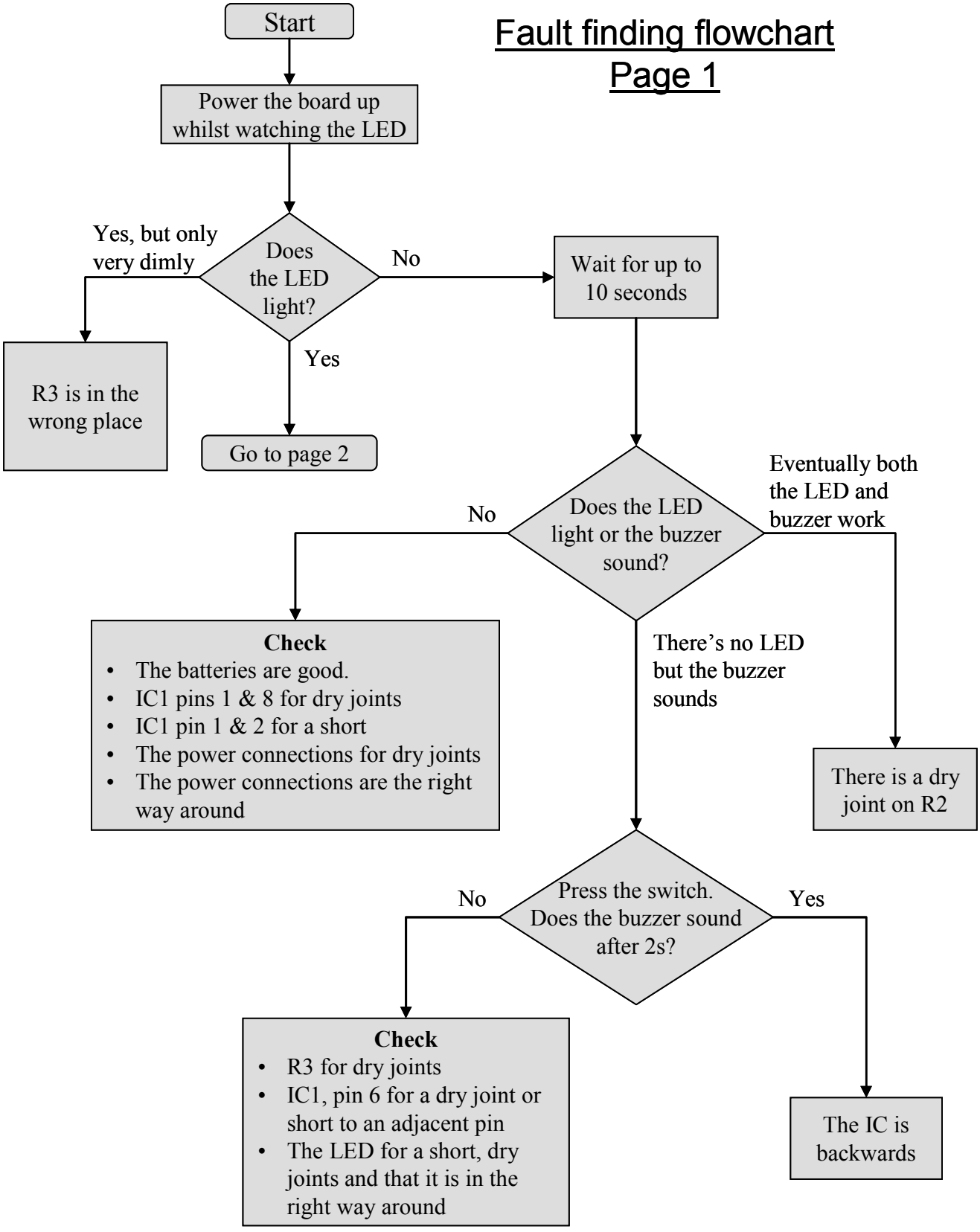
Testing the PCB

The circuit has been designed to allow easy testing of the PCB. To test the PCB you will first have to insert a chip programmed with some software to allow it to be tested. The purpose of the test program is to test that the LED can turn on and off, the buzzer can sound and that the switch is functioning (used to turn the LED on and off). The test software works as shown below.

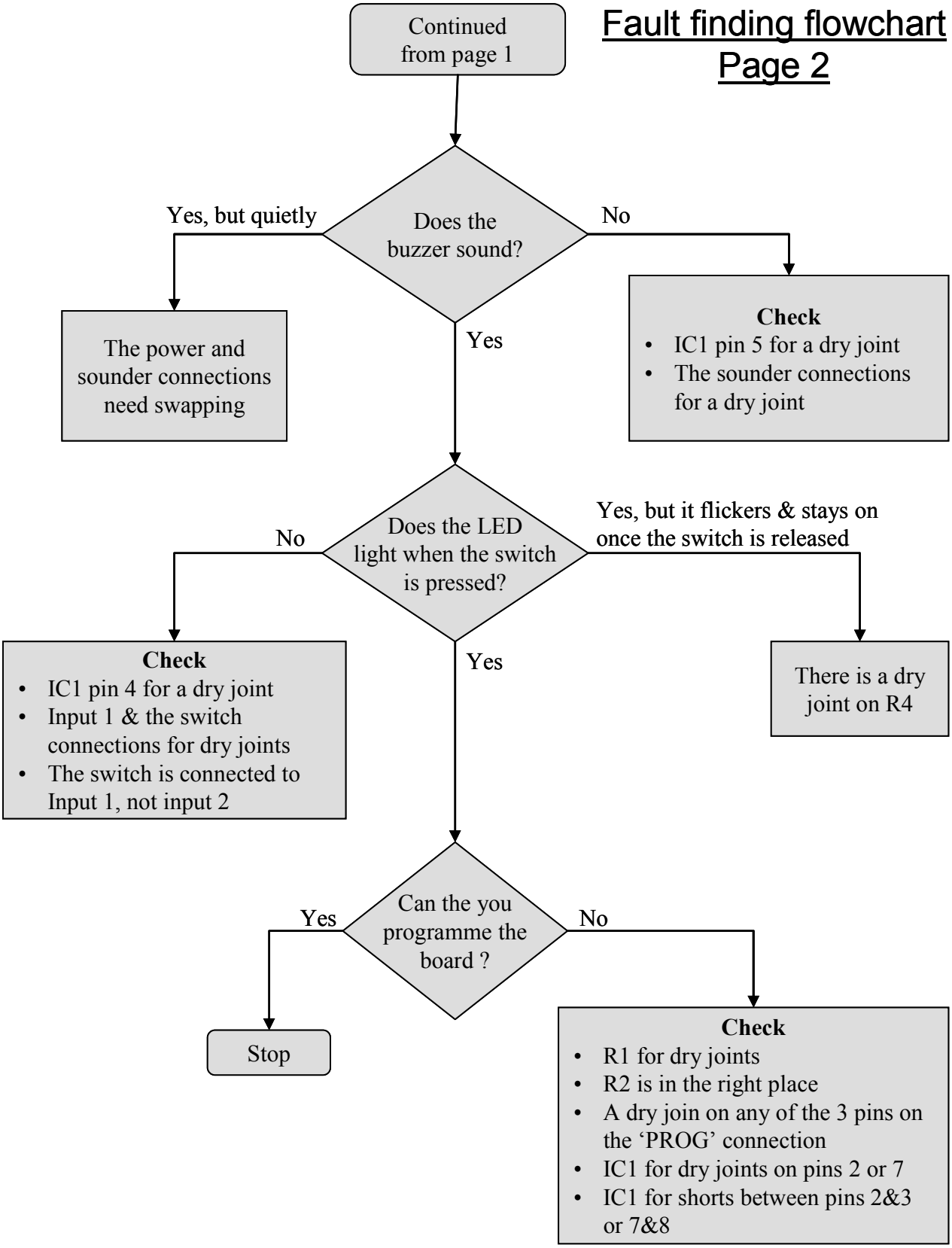


If you have problems with any of the above use the fault finding flow charts to find the cause of the fault. Please note that these fault finding diagrams have been based around using the test software outlined above. If you are using an alternative test program they will not be suitable.

Fault Finding



Fault Finding continued

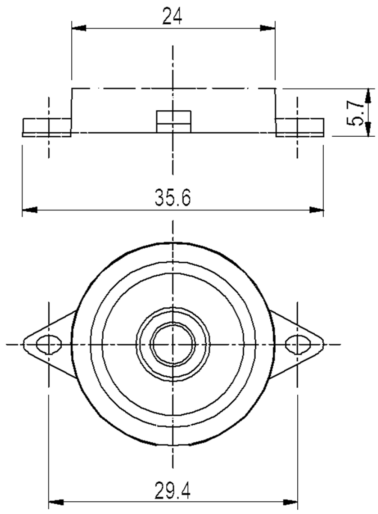


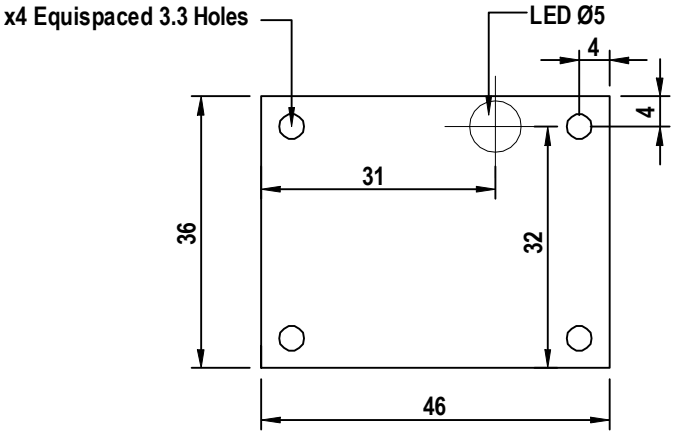
Designing the Enclosure

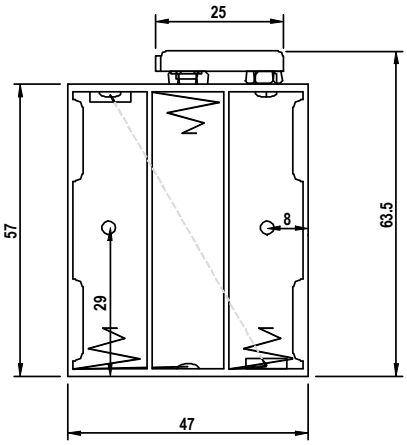
When you design the enclosure, you will need to consider:

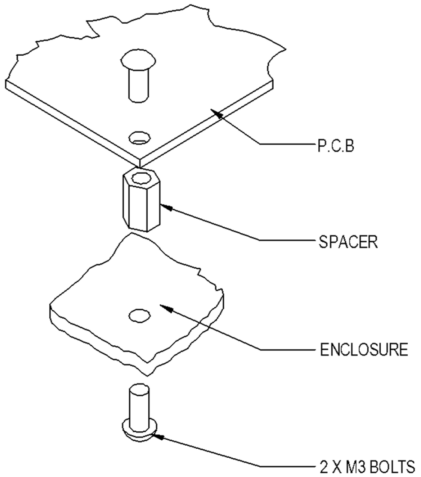
- The size of the PCB (below left).
- Where the LED is located on the PCB.
- Where the start switch will be mounted (bottom right).
- Where the sounder will be mounted (right).
- Access to the batteries to allow them to be changed (below right).

Technical drawings of these items are illustrated on this page, which should help you design your enclosure.
All dimensions are in mm.





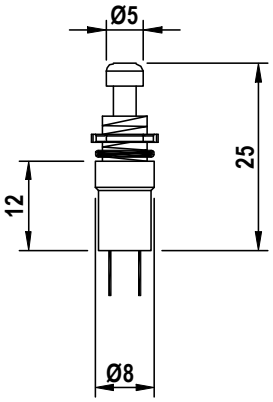




Mounting the PCB to the enclosure

The drawing to the left shows how a hex spacer can be used with two bolts to fix the PCB to the enclosure.

Your PCB has four mounting holes designed to take M3 bolts.



Possible student programming tasks

Test program

The test program referenced in the 'Testing the PCB' section can be downloaded from the resource section of the Kitronik website at the web address listed at the bottom of this page. We would suggest that you program up a chip with this test software. This can then be placed in an IC holder (to protect its legs) and used by students to test their PCBs once they have completed building them.

Task overview

We would suggest that you split the programming of the timer board into up to four separate tasks.
These are:

Task 1 - Basic timer

When the button is pressed the LED will light for 10 seconds, then the buzzer will sound for one second.

Task 2 - Early warning bleeps

A few seconds before the one second out of time buzzer sounds, it will give a short bleeps to warn the time is almost up.

Task 3 - User configurable delay

A special mode will be added to set how long the delay is so the timer can be reprogrammed during normal use without a PC.

Task 4 - Musical

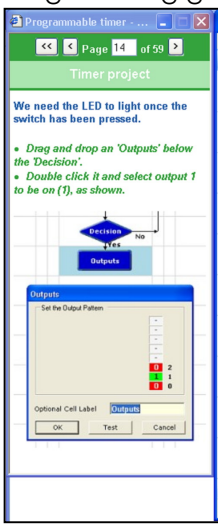
The final task will be to replace the time out tone with a tune.

Student guides available on-line

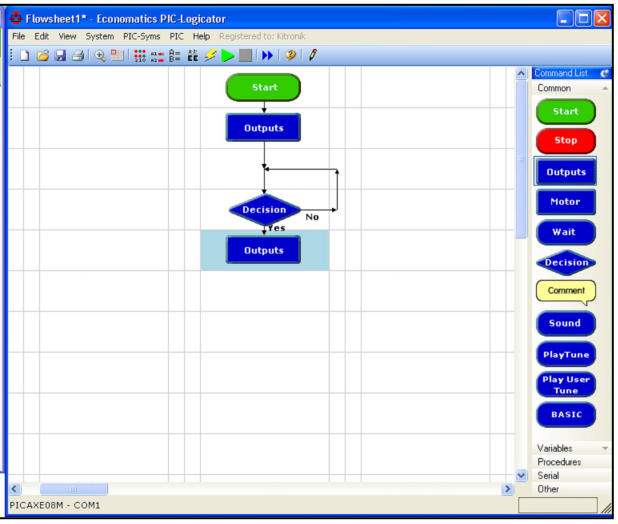
To guide students through the programming tasks listed above an electronic guide can be run or downloaded from our website. This guide runs in any web browser, so you don't need to install any software, however the files can be copy onto the PC if you don't have internet access. It is intended that the guide takes up ¼ of the screen, leaving space for the flowchart software. The screen shot below is of the PIC logicator guide, but guides for other software packages are available. To find out if there is a guide for your software and to give it a try go to:

www.kitronik.co.uk/timer_tasks.htm

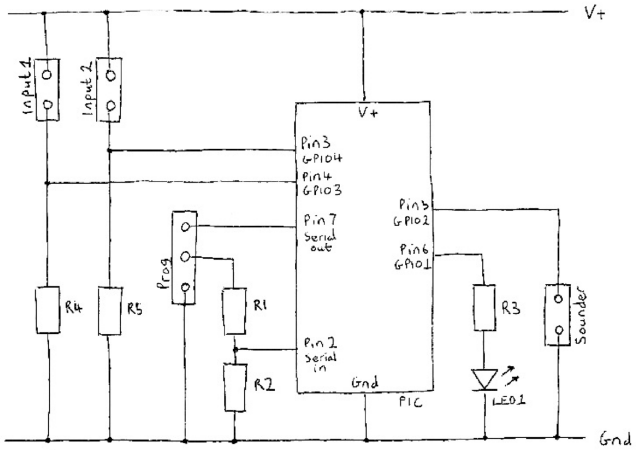
Programming guide



Programming software



How the programmable timer works



The timer is based around an eight pin PIC microcontroller device. A PIC is in effect a small computer that behaves in a way determined by the software it's programmed with. This software is generated by the user / student. It is this code that will determine the eventual function of the project. To aid the design of this software the following describes the function of the hardware (circuit) that this software controls. From the programmable timer circuit above you can see that the PIC has two usable inputs and two usable outputs.

Input/Output	Pin	Connected to
Input 1	Pin 4, GPIO3	Switch
Input 2	Pin 3, GPIO4	User defined
Output	Pin 6, GPIO1	LED1
Output	Pin 5, GPIO2	Sounder

The other connections to the PIC are to provide it with power (V+ and Gnd) and also allow it to be programmed with the user defined software (Pins 2 and 7).

Input 1 is connected to the switch which is of the push to make variety. When this switch is not pressed resistor R4 pulls the voltage on the input to the PIC to a low voltage. When the switch is press (closed) the voltage on the PIC pin is pulled up to a high voltage. You will be able to read this change of state in your software.

Input 2 has been left as a spare input should you wish to add another input device. The pull down resistor R5, which is the same as that used on Input 1 is already included.

The purpose of resistor R3 is to limit (restrict) the flow of current into the LED. This controls the brightness of the LED and prevents it from becoming damaged, which would happen if no resistor was used.

The sounder (buzzer) that has been used in the circuit does not have any drive circuitry. This means that it needs driving with square wave (alternating high low voltage signal) to make it produce a tone. The frequency of the tone will be the same as that of the square wave that is used to drive it.

One other point worth noting is the processor clock. For any micro-controller to work it requires a clock source. The micro-controller uses this clock so that it knows when to execute the next line of software. Often these clocks are generated externally but in the chip used in this circuit the clock is built into the chip itself. This is why it does not appear on the circuit diagram.

Online Information

Two sets of information can be downloaded from the product page where the kit can also be reordered from. The 'Essential Information' contains all of the information that you need to get started with the kit and the 'Teaching Resources' contains more information on soldering, components used in the kit, educational schemes of work and so on and also includes the essentials. Download from:

www.kitronik.co.uk/2121

This kit is designed and manufactured in the UK by Kitronik

Telephone: +44 (0) 845 8380781
Sales email: sales@kitronik.co.uk
Tech support email: support@kitronik.co.uk
Web: www.kitronik.co.uk

www.kitronik.co.uk/twitter
www.kitronik.co.uk/facebook
www.kitronik.co.uk/youtube
www.kitronik.co.uk/google