

Build this tiny micro-controller driven lightshow

Micro-controller driven LED displays are popping up everywhere nowadays, from shop windows, to tiny belly-button raver lights. However, most of these have a few problems. Firstly, the cheap ones are usually just that—so cheap that they are basically a throw-away item.

Secondly, most of these lights are not programmable, so you are stuck with the basic program and sequence they are built with.

This kit allows you to build a high-quality, fully programmable light display capable of producing any colour you desire. What's more, it is a mere 50mm x 50mm in size, and can run from three 1.5 volt cells (or four 1.2 volt rechargeables, which are recommended) for many hours. It also has the option of using either a push-button switch or mercury for controlling the sequence. The mercury switch allows you to turn the light on and off with a simple shake of the circuit board!

With careful programming, you can even program it with more than one sequence, and use the switch to flip between them!

The brain behind the colours

The brain of the circuit is the Picaxe08 pre-programmed microcontroller from Revolution Education in the UK. This company provides a range of microcontroller and micro-based boards and products, and one of those is the Picaxe series.

These consist of various Pic microcontrollers that have been programmed with a bootstrap program that allows them to be connected directly to a serial port on your PC without the need for special programmer boards. Just two resistors are required for the three-wire interface. This makes it possible to produce projects where the micro can be reprogrammed in-situ in seconds.

If you look at the circuit board, you will notice the three-pin connector and the resistors R1 and R2—they are all that is needed to connect the microcontroller to the serial port of a PC or other device. No other microcontroller is this simple to use. This reason, and the availability of a free basic interpreter for this range of micros,

are why these micros were chosen for this and similar kits. They are simply easier to use and far cheaper than anything else on the market. However, they do have programming space and speed limitations, but in most cases, unless you are trying to sample or switch at radio frequencies or whatever, this generally doesn't cause problems

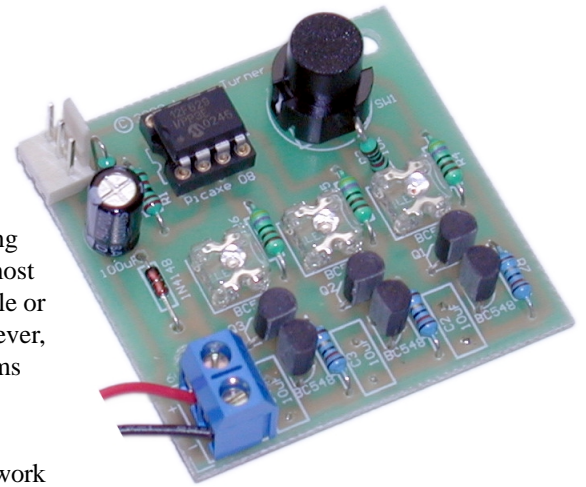
The circuit

As already mentioned, all of the hard work is done by the microcontroller, and there is not much else to the rest of the circuit. The micro drives three superflux LEDs using three of its output pins, via constant current circuits set to provide 50mA of current through each LED. A fourth pin on the micro is used to monitor the state of the switch.

The 10k resistor, R3, is used to tie the switch input to V+ when the switch is not being pressed, rather than letting it float. And that's it!

We will look at the constant current circuits by looking at one of them, composed of R4 and R7, and BC558 transistors Q1 and Q4.

Current flows through the LED and through Q1 to ground via the 13 ohm resistor, R7. Q1 is biased on by a small current flowing through R4, a 4k7 resistor. When the current flow through R7 reaches approximately 50mA, the voltage across it reaches 0.65 volts, which starts to turn on Q4. Q4 robs some of the base current from Q1, thus preventing the cur-



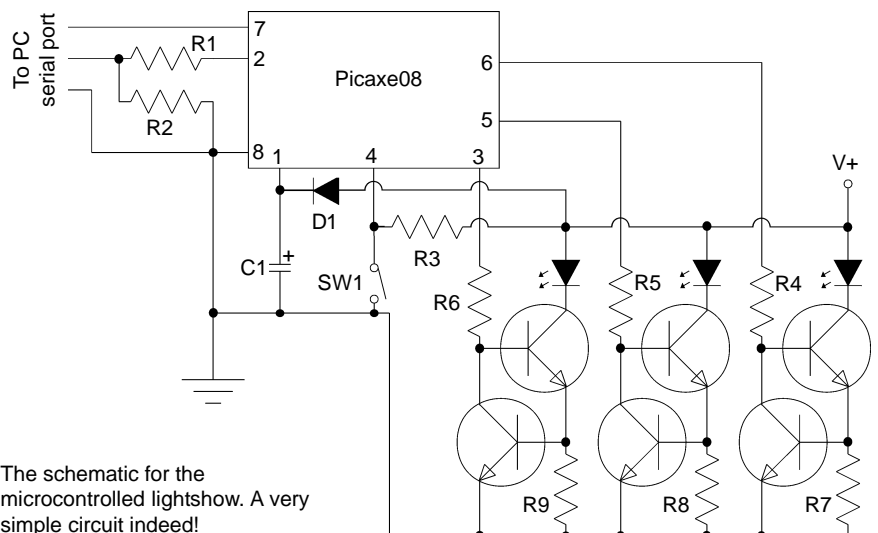
rent through the LED from rising further. This simple and effective circuit is stable and can be pulsed on and off as fast as the micro can manage, and will always limit the LED current to 50mA maximum.

Assembly

Well, being so simple, there isn't much to say here, other than make sure you get the microcontroller, LEDs, the diode and transistors the correct way around, and be sure to use the supplied IC socket for the microcontroller.

Also note that the capacitor, C1, is generally not needed unless you are powering the circuit from a power source that is a long way from the circuit itself. The capacitor actually makes it hard to effect a power-on reset by disconnecting and reconnecting, as it holds enough charge to stop this reset from happening sometimes.

Lastly, the other three capacitors, C2 to



The schematic for the microcontrolled lightshow. A very simple circuit indeed!

C4 are used to remove any glitches in the output. In one of the sample programs, the LEDs flash off for a fraction of a second while the micro goes forward through its program. The capacitors prevent this glitch, but do slow the response of the LEDs. The capacitors supplied are 47uF, 6 volt surface mount tantalums. As 90% of people won't need to fit them, these were supplied in the cheapest way possible—there is no point including a couple of dollars worth of caps that most people won't use! They can be soldered directly to the underneath of the board if you wish, or solder some component lead offcuts to the capacitors' pads and mount them like a normal component.

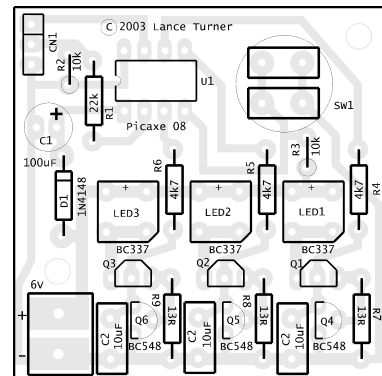
Once everything is in place, apply a **regulated** (ie, from a three or four cell battery pack or regulated plugpack) DC voltage of up to 6 volts and shake the board. The LED should start going through an animated sequence (the microcontroller is supplied with a sequencing program already programmed in). If nothing happens, check your work, making sure there are no shorts between tracks or pads anywhere, and the micro is the correct way around. Also check that the input pin on the micro changes from 0 to 6 volts when you turn the board upside down. Another test is to measure the voltages on the output pins. If they are changing, but the LEDs are not glowing, then the LEDs are most likely in the wrong way around.

If the LEDs start to flash, this means that you have opted for the pushbutton option—the mercury switch is normally closed, the pushbutton is normally open. You simply need to upload the pushbutton version of the program (or any other suitable program) into the microcontroller.

Something to be aware of is that the Picaxe08 can behave strangely when driven from a full 6 volts. If you suspect this is the problem, then use three batteries instead of four, or use rechargeable cells instead—this will give you a 4.8 volt supply. The 0.6 volt drop across the diode should stop this happening in most cases.

Programming

If you get bored with the pre-programmed fade sequence, you



The layout of the circuit board, reproduced approximately full size. It is small enough to use as the basis for a bike light, party light, strobe or anything else you can think up!

can program your micro with any sequence that will fit into the 128 byte program space. A number of programs are available for download from my website at www.alphalink.com.au/~spod/kit_examples/Programming cables are available from my website also. However, you can make your own by using an old 9-pin serial cable (an old mouse is ideal for this) and a 3-pin header socket. Instructions for making the cable, as well as the programming software, are available on the Revolution Education website at www.rev-ed.co.uk

Parts list

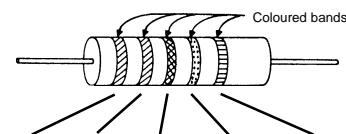
- R1 22k
- R2-R3 10k
- R4-R6 4k7
- R7-R9 13R
- C1 100uF 16V
- Q1-Q6 BC547
- D1 1N4148
- U1 Picaxe08
- SW1 Mercury switch/pushbutton
- TB1 2 way terminal block
- CN1 3 way header
- SOC1 IC socket
- LED1-LED3 Superflux LEDs

Identifying the parts of your Picaxe controller kit

The following guide should help you to identify the parts and assemble it successfully.

Resistors

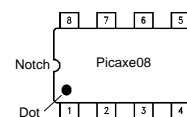
The coloured bands represent numbers and multipliers as shown in the table. Some resistors have four bands (two digits, a multiplier band and a tolerance band) while some resistors have five bands. Five-band resistors are read in the same manner as four-band resistors, except that the first three bands are digits, the fourth a multiplier and the fifth the tolerance band.



Colour	Hundreds	Tens	Units	Multiplier	Tolerance
Black	0	0	0	1	20%
Brown	1	1	1	10	1%
Red	2	2	2	100	2%
Orange	3	3	3	1000	
Yellow	4	4	4	10,000	
Green	5	5	5	100,000	
Blue	6	6	6	1,000,000	
Violet	7	7	7		
Grey	8	8	8		
White	9	9	9		
Gold				0.1	5%
Silver				0.01	10%

The Picaxe08

This is the 8 pin device, and will be marked 12F629. Note that it must be installed the correct way around. Pin 1 is marked with a small dot, and the pin 1 end of the chip will have a semicircular notch or mark. The IC socket also has this notch, so make sure it is installed the correct way around also, as this notch is used to indicate device polarity.



Diode

The 1N4148 diode is the small glass barrel-shaped thing. There is only one diode in the kit.

