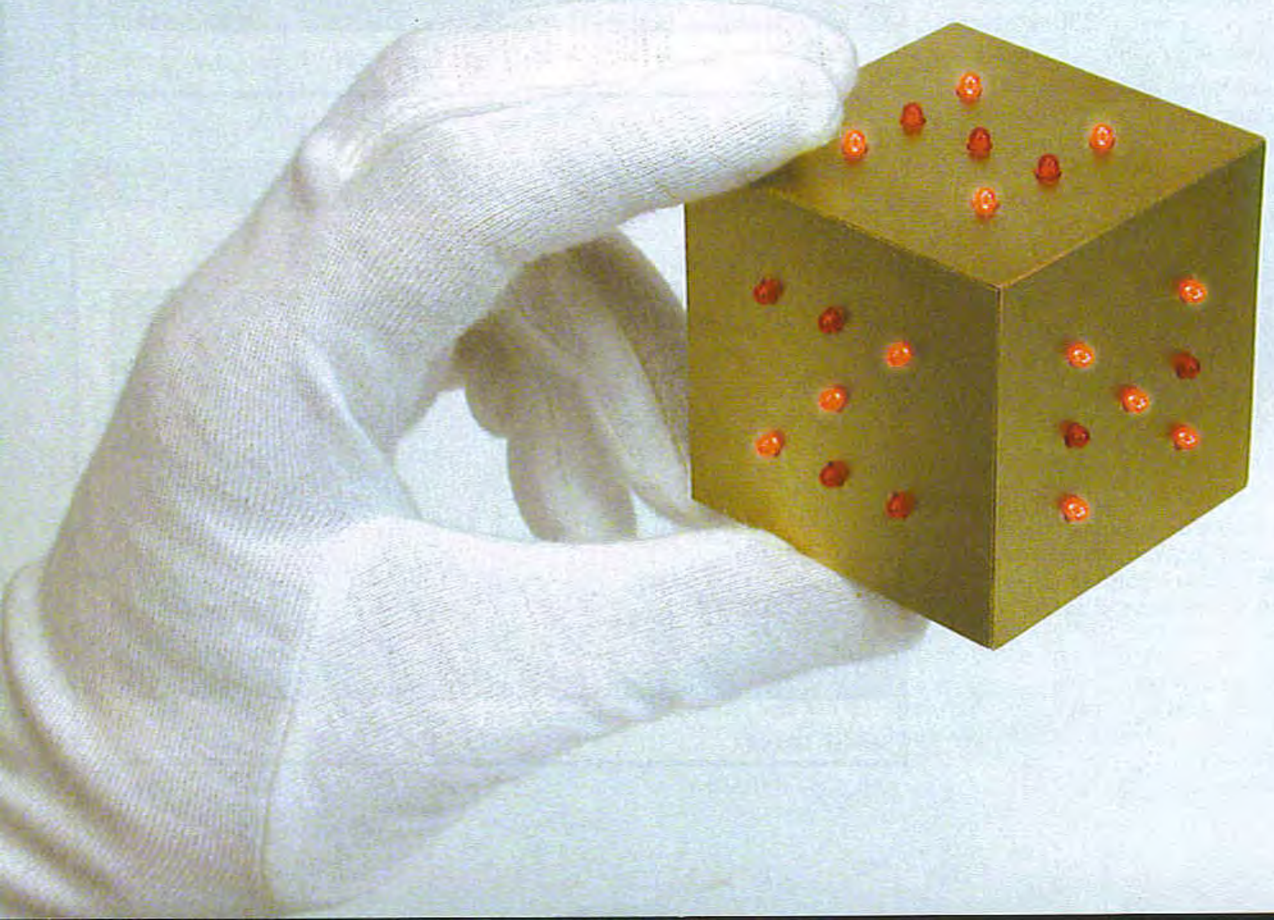
Rolling

Paul Goossens

Projects for electronic dice have previously appeared in Elektor Electronics. Different designs and shapes have found their way into the magazine, but one thing they had in common: they were all two-dimensional dice. But now that is about to change!



Dice

Three-dimensional electronics

Electronic dice are usually constructed on flat boards. To give the impression of a 'real' dice, seven LEDs are often used in a pattern similar to that on a dice. The original intention was to base this circuit on a traditional 'flat' design, until somebody from the editorial staff suggested that you should really be able to throw a dice. After all, a dice is not an item that lies still on the table.

'But that is the advantage of an electronic dice!' you'll say. You don't need extra space to throw it and it won't roll off the table either. And this was exactly how our design staff reacted to this unusual suggestion. But the editor was adamant: this dice has to roll!

Fortunately the team at *Elektor Electronics* handles such situations smoothly and it wasn't long before the lab were prepared to have another look at it.

It didn't take long before the first ideas were put on paper. One obvious design for an electronic rolling dice is to mount LEDs onto a cube, showing the proper value on every side. The LEDs are driven via current limiting resistors from a 9V battery and Bob's your uncle! It quickly became apparent that with a homemade dice it was extremely difficult to get the centre of gravity exactly in the centre. Such a dice would therefore have a bias. The chance that the lightest side ends on top is greater than for any of the other sides. The value on that side will therefore occur more often than the value on the 'heavier' side.

How else?

After some more thought we came to the conclusion that a dice with seven LEDs on every side was the best option. After throwing the dice, the six sides would each take on a different random value. In this way you can guarantee that the dice is completely 'honest'.
This solution does require a bit more electronics. First of all, you have to detect when the dice is in the process of being thrown. For this we used a mercury tilt-switch. This also makes it possible to 'roll' the dice just by shaking it.
The electronics then has to generate the random numbers that appear on the six sides of the dice. The use of a microcontroller comes to mind straight away. This keeps the circuit relatively small, which comes in handy for a dice. Another part of the design that required some thought is the method used to connect the boards together. It doesn't look very nice if dozens of connections go from a main PCB to the other five boards. To keep the number of connections down we've chosen serial connections between the boards. On every board a shift register takes care of driving the LEDs. This way only four connections are required: two lines for the supply, one for data and the fourth for a clock signal.
Another area that needs to be considered is the current consumption and on/off switch. In this design a standard switch can't be used of course, because nothing may protrude from the dice. It would otherwise roll very strangely and at worst switch itself off. We got round this problem with the addition of a (recessed) push-button to turn the circuit on. It is turned off completely automatically.

Main PCB

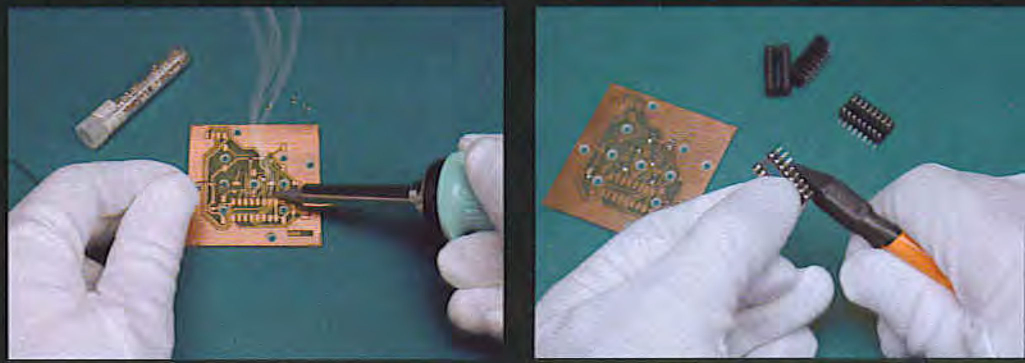
After these considerations we arrived at the circuits for the dice. At the hart of the circuit is IC1, a microcontroller by Atmel (**Figure 1**). We have used this type previously and have given a detailed description in past issues of Elektor Electronics. It therefore suffices just to mention that the software for this project can be freely downloaded from the Elektor Electronics website. As this controller has an internal flash memory for the program there is no need to add external memory. Crystal X1 in conjunction with C2 and C1 provide a clock signal. IC1 drives the 7 LEDs directly via its I/O pins. As we have used low-current LEDs there is no need for an extra buffer.

The supply section may appear a little unusual at first. This is because we want to turn the circuit on with a push-button and let the microcontroller itself turn the supply off. The part of the circuit round T1 and T2 provides this functionality. K2 is the connector for the 9 V battery. When push-button S1 on board 2 is pressed, a small current will flow from + 9 V via the base/emitter junction of T1 and R8 to ground. This causes T1 to conduct, feeding a current to voltage regulator IC2. This then supplies the rest of the circuit with 5 V.

When the push-button is released, the current can no longer flow through R8. It is of course undesirable that the circuit would then lose its power. Just try to keep the push-button held down during a throw; it's not exactly user-friendly for a dice.

To get round this problem we've added resistor R9 and transistor T2. T2 is driven via the microcontroller. When the circuit is switched on, T2 is also turned on, causing a current to flow from the base of T1 to ground via R9 and T2. The supply is then no longer dependent on S1 being pressed.
When the processor notices that it hasn't moved for a while (several min-

Figure 3. Construction of the dice.



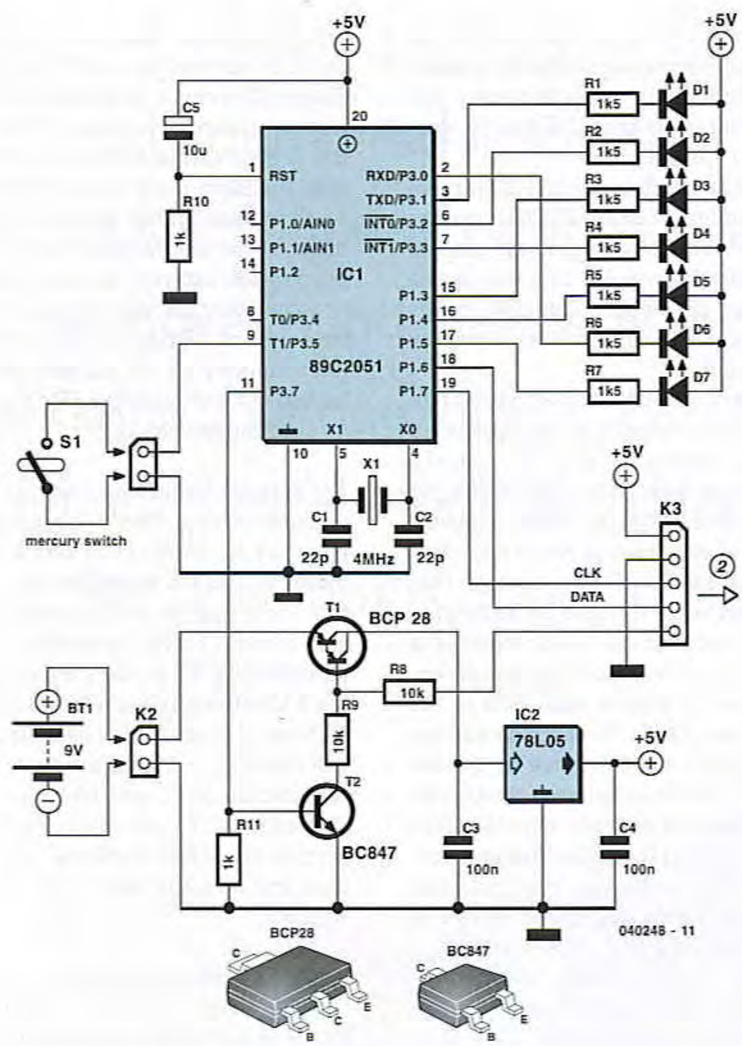


Figure 1. The main board is more complex.

utes), it stops driving T2. The dice then turns itself off automatically. When switch S1 is pressed again, the whole sequence repeats itself.

When the supply is applied a reliable reset signal is generated for the microcontroller by C5 and R10.

The CLK and DATA signals are fed to the next board via connector K3. K3 also has connections for the supply and the on-switch (pin 5). The switch had to placed on board 2 due to a lack of room on the main board.

Other boards

The circuits for the other boards (2 to 6) are almost identical. The main difference is that circuit 2 has a switch, which isn't present on the other boards.

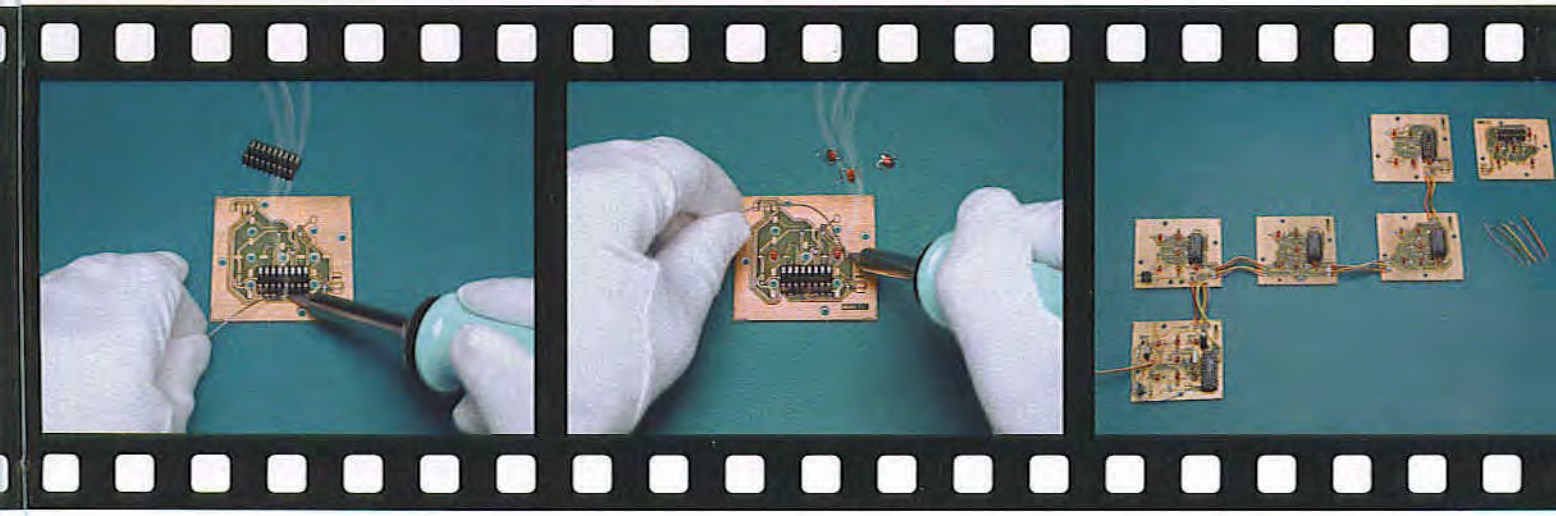
Figure 2 shows the circuit diagrams for board 2 and the other four boards. There is obviously no need for a feed-through connector (K2) on the last board.

At the centre of the circuit is the shift register, a 74HCT4094. This type of IC has a parallel output register. Every rising edge at the clock input causes the 8 bit data in the IC to shift by one bit. The first bit then takes the value of the data input. The last bit shifted out is connected via pin 9 to the data input of the shift register on the following board. This effectively turns all boards into one large shift register.

Construction

The construction of the boards differs in a few ways from the usual method. In several places we have used standard components, which have to be surface mounted. The reason for this is that we'd rather not have any leads sticking out through the boards. This would result in a number of sharp points on the outside of the dice, which would damage your furniture when the dice is thrown, and this obviously isn't our intention. To make things clear, we've included a series of photos of the construction of the prototype in **Figure 3**, as a picture is often worth a thousand words.

The component layouts for all boards are shown in **Figure 4**. First of all, the SMD components are soldered to the boards. Then it is the turn of the IC sockets, which also need to be surface mounted. You need to bend the pins outwards and cut them to length (take a measurement from the board first!). Next it's the turn of the 42 (!) LEDs. The



leads of the LEDs need to bent in the right form, as shown in **Figure 5**. Take great care that the anode and cathode are positioned correctly on the board. The case of the LED is then stuck through the hole. The ends of the leads should now be flat on the board, where they can be soldered. At this stage it is not important if the LEDs are not all at the same height. After soldering all LEDs they are pushed outwards such that they stick out just a little above the surface of the dice. If we now press the board onto the table, the LEDs should all be at the right height.

Apart from the ICs in DIL packages, there are a few other components that have 'normal' leads. As we don't want the outside of the dice blighted by protruding leads and solder, these also have to be surface mounted. This includes the crystal and the mercury tilt-switch. The leads of these components have to be cut fairly short, and the ends turned through 90 degrees. These ends are then soldered onto the board. Take care that you don't shorten the leads too much!

Connecting the boards

Once all boards have been populated it's time to connect them together. **Figure 4** shows how this is done. You should use individual pieces of wire for this, with a length of about 5 cm. If the wires are too short it makes it more difficult to fix the boards together at a later stage. There would also be a greater chance that one or more of the wires became loose!

Lastly the ICs are plugged into their sockets. (Make sure that you plug them in the right way round!) Now that the electrical construction is finished, the circuit should be tested. It is wise to do this *before* the dice has been fully assembled. At this stage the electronics are still easily accessible, should you need to make any repairs. After you have carefully checked for short circuits you can connect the 9 V battery. When you press the switch (on board 2), the circuit will come alive and a random number of LEDs lights up on the boards.

If you then shake board 1, the dice will simulate a roll. The numbers on all sides will change regularly. After a few seconds the changes slow down and the result of the roll is shown on the boards.

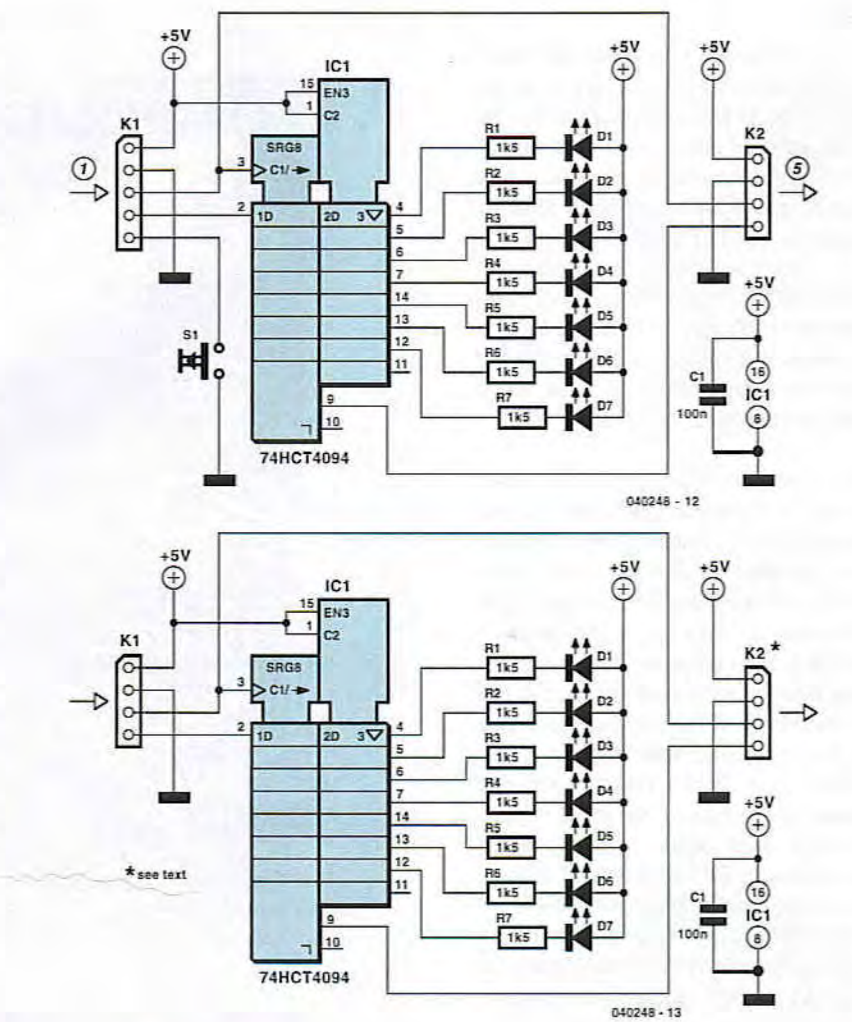
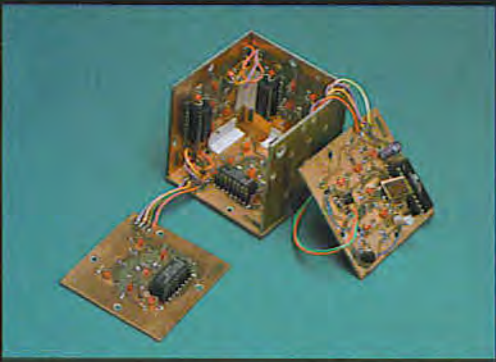
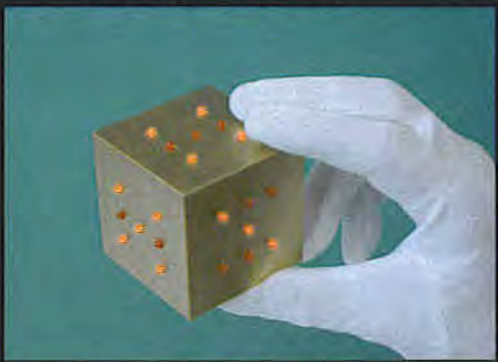


Figure 2. Above is the circuit for board 2, beneath it the circuit for the other four.





3D

The electronics may now be completed, but it doesn't look anything like a dice yet. The boards now have to be fixed together into a cube. Board number 4 is placed flat on the table and boards 2, 3, 5 and 6 are fixed at right angles to it (see **Figure 6**).

Use a plastic angle section (available from most DIY stores) to fix the boards together. You should cut the angle section into small lengths and use these to glue the boards together.

You shouldn't glue the last board (board 1), since this has to be removed every time the battery needs replacing. You need to glue a piece of angle section on the other boards, such that they line up with the 3 mm holes in board 1. Now carefully drill a 2.5 mm hole through each angle section at the point where it meets the mounting hole of board 1, then make a 3 mm thread in it. If you want a stronger thread you can glue a flat piece of plastic onto each angle section, which should be drilled and threaded as well. The inside of the dice should be filled with cotton wool or foam rubber, preventing the 9 V battery from moving or causing a short circuit. You can now screw board 1 into place.

And finally

You could give the dice a nice lick of paint, as we did with our prototype, but we leave that to your own preference. In any case, we hope that you enjoy constructing it and have fun using the dice in many games!

(040248-1)

COMPONENTS LIST

PCB # 2

Resistors:
R1-R7 = 1kΩ5 SMD
R8,R9 = 10kΩ SMD
R10 = 1kΩ SMD
R11 = 100kΩ SMD

Capacitors:
C1,C2 = 22pF
C3,C4 = 100nF SMD
C5 = 10µF 16V radial

Semiconductors:
D1-D7 = LED, 3mm, red, low-current
IC1 = AT89C2051-12PI, programmed, order code **040248-41**
IC2 = 78L05
T1 = BCP28 (Conrad Electronics # 153225-8B)
T2 = BC547B

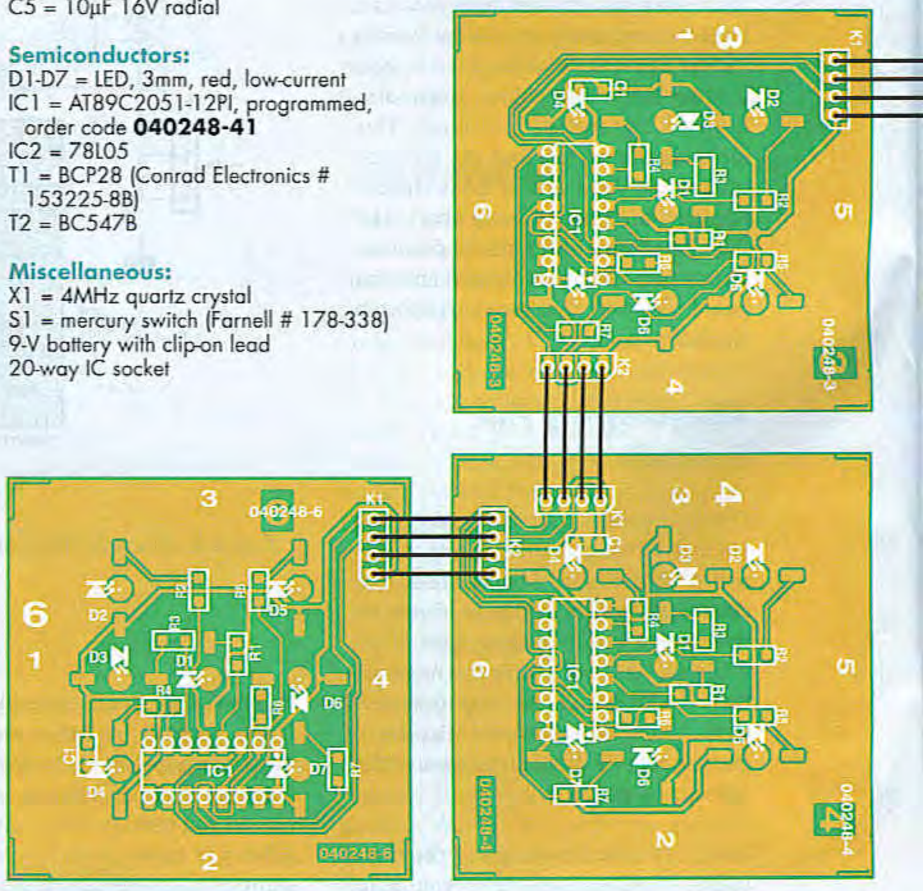
Miscellaneous:
X1 = 4MHz quartz crystal
S1 = mercury switch (Farnell # 178-338)
9-V battery with clip-on lead
20-way IC socket
ABS (hard plastic) angle section
2 M3 screws (countersunk head)

PCB # 2 through # 6
(per board)

Resistors:
R1-R7 = 1kΩ5 SMD

Capacitors:
C1 = 100nF SMD

Semiconductors:
D1-D7 = LED, 3mm, red, low-current
IC1 = 74HCT4094



Order of construction

- Solder the SMD resistors and capacitors
- Bend and cut the pins of the IC sockets
- Solder the IC sockets
- Bend and solder the LEDs
- Prepare the connections between the boards
- Testing
- Connect the boards together
- Add cotton wool and the batter
- Fix the last board into place
- Final test

Miscellaneous:
16-way IC socket
S1 = switch type DTS61K (only on board # 2)

PCB, order code 040248-1 (contains six sections for a complete dice)

Disk, source and hex code files, order code **040248-11** or Free Download

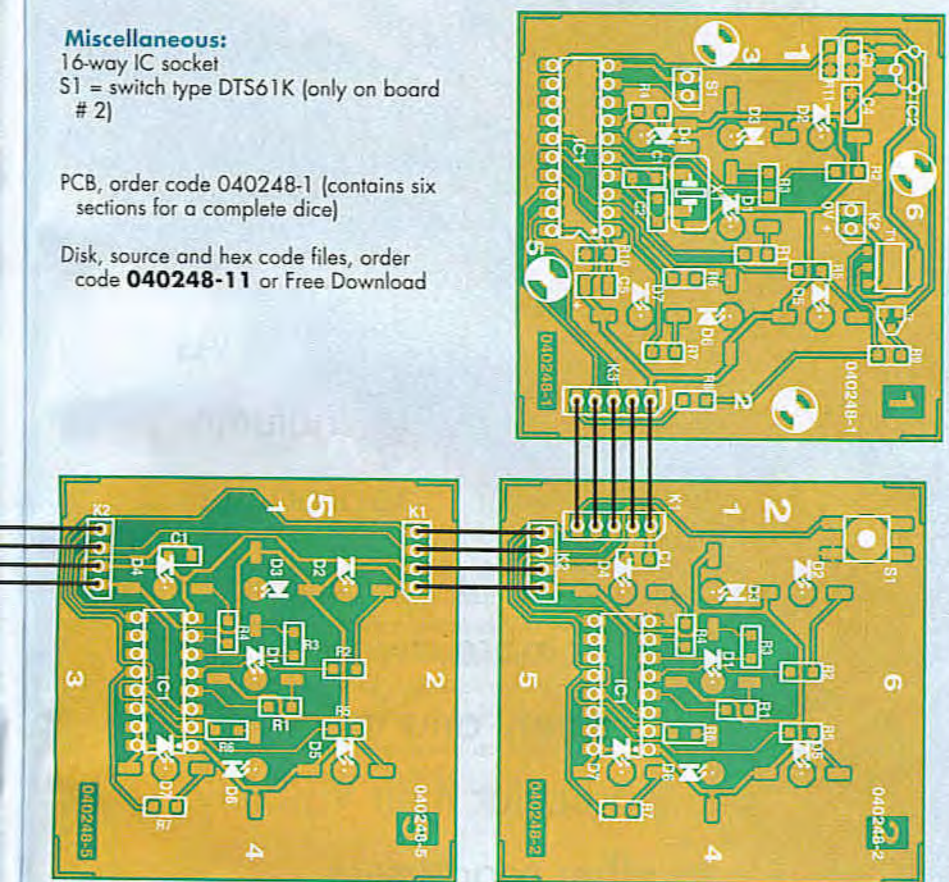


Figure 4. Component layout. Four of the six boards are almost identical.

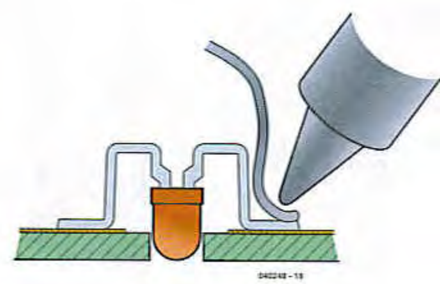


Figure 5. Bend the leads of the LEDs according to this example.

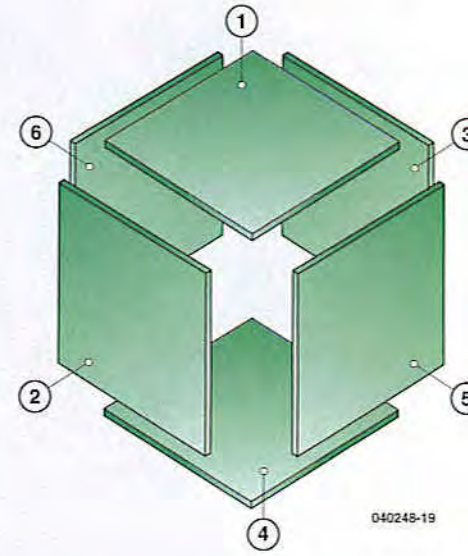


Figure 6. The six boards make up a cube.