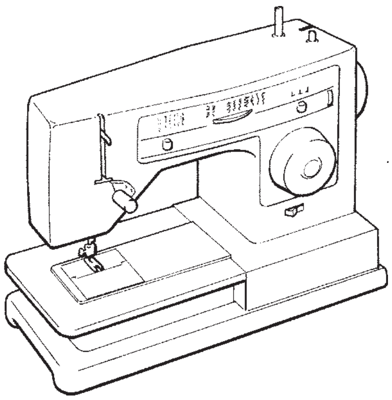
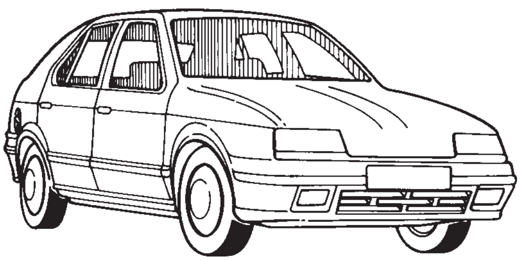
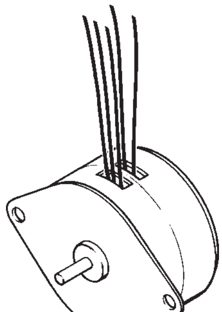
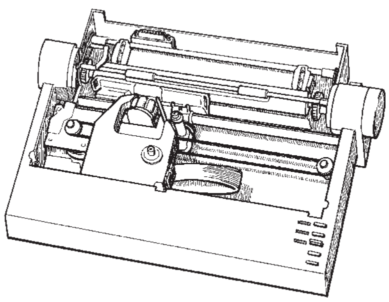
STEPPER MOTOR CONTROL AND APPLICATIONS

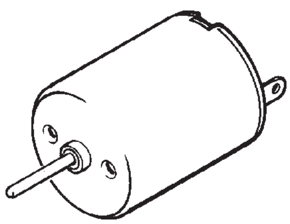
Electric motors of different types and sizes are used in increasingly large numbers in manufactured products ranging from cars to children's toys. This reflects increasing consumer expectations, advances in manufacturing methods and motor design (with cost reductions), and the need for specialised motors generated by a new generation of products such as video recorders, computer and computer accessories.

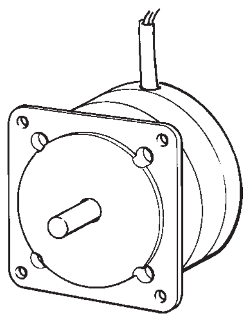




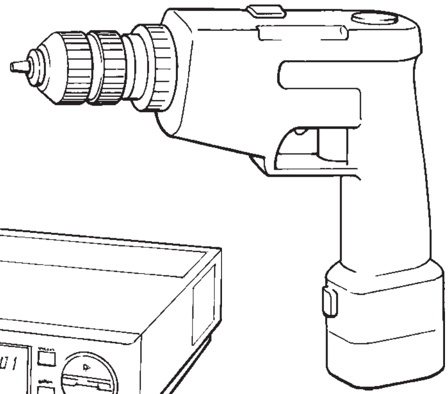


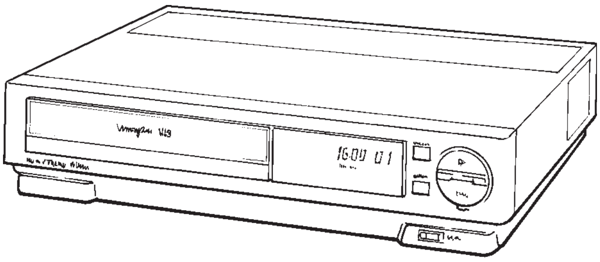






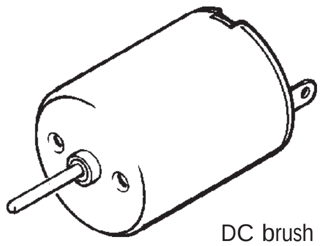
Increasingly, these electric motors need to be controlled by means other than an operator simply switching them of or off. Products using one or more motors often incorporate electronic control as a matter of course. Many cordless electric drills, for example, feature an electronic speed controller - and, of course, we take it for granted that a video recorder will perform a range of complex mechanical operations to start up at the momentary push of one button. (When this happens the machine is performing the complex tape lacing operation that has really made home video a practical proposition.)



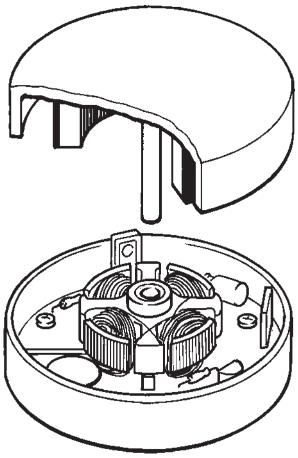


Many types and sizes of electric motor are available - with three main categories operating in the lower voltage range (6-24volts). These are:

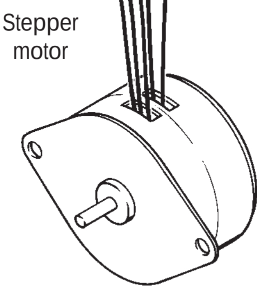
- DC brush motors
- brushless motors
- stepper motors



DC brush motor

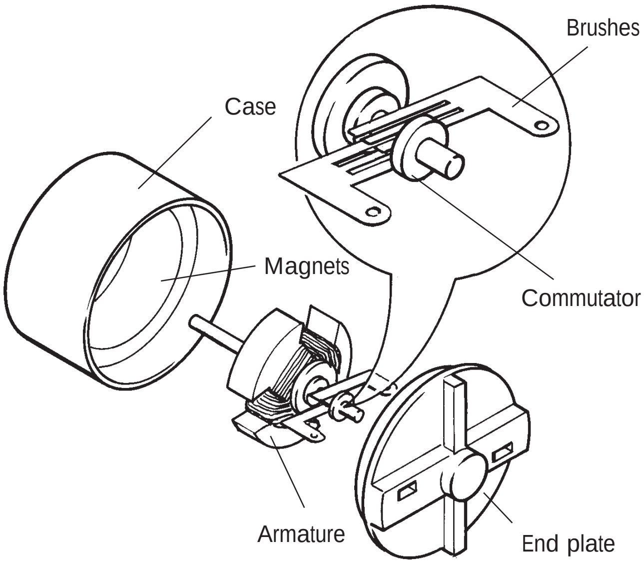


Brushless motor

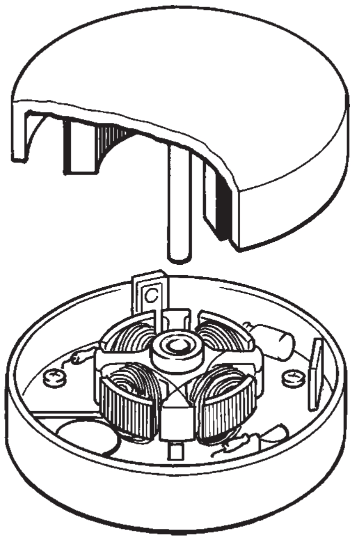


Stepper motor

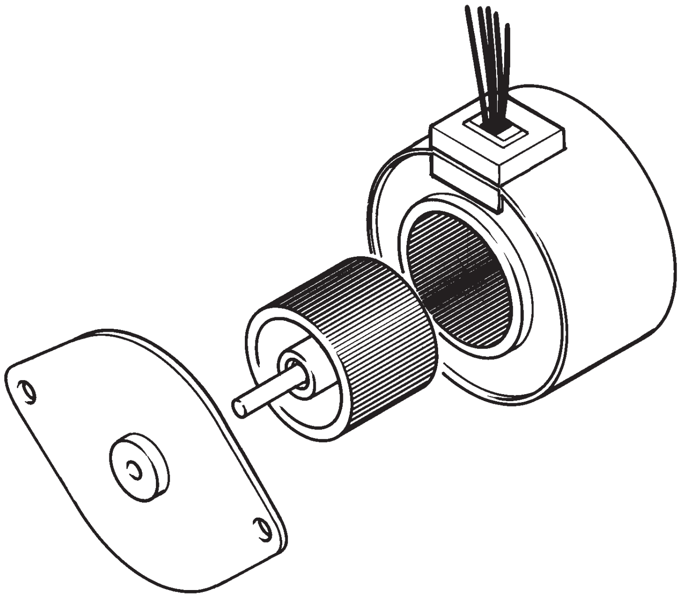
A DC brush motor uses a *commutator* to cause the magnetic field in the armature coils to change so that the coils will rotate between permanent magnets.



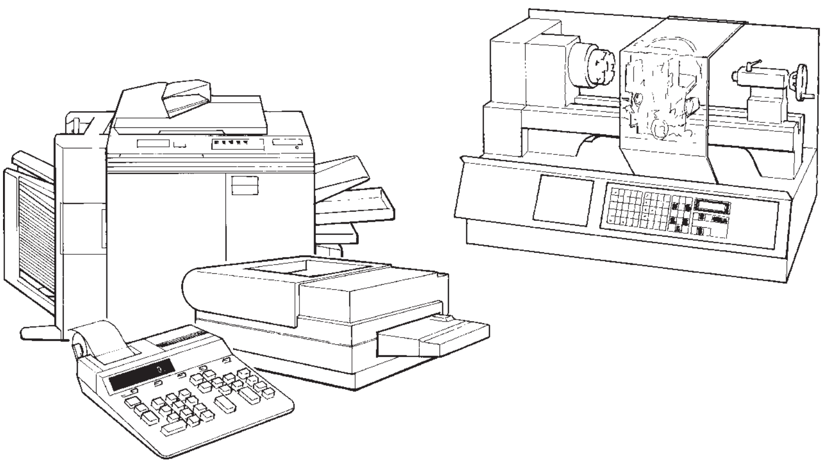
A brushless DC motor has a permanent magnet *rotor* and fixed *stator* coils - but no commutator. As the rotor turns, one or more sensors close to its edge send a signal to a control circuit that energises the stator coils in the correct sequence.



A stepper motor - sometimes called a *stepping motor* - has a permanent magnet rotor that revolves within fixed stator coils. Unlike a brushless motor, however, there are no sensors. The rotor is driven around by switching the coils on and off in a special sequence using a driver circuit.

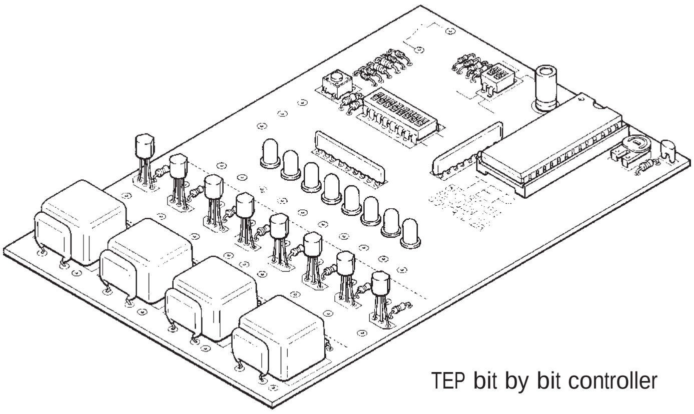


Although generally more expensive than the other types, stepper motors offer many more opportunities for precision control because they can be made to turn through an exact number of steps. Stepper motors are therefore used, for example, to drive the moving parts of CNC machine tools and computer printers. Up to 15 stepper motors might be used to effect the mechanical paper feeding operations in a modern photocopier.



DESIGN OPPORTUNITIES

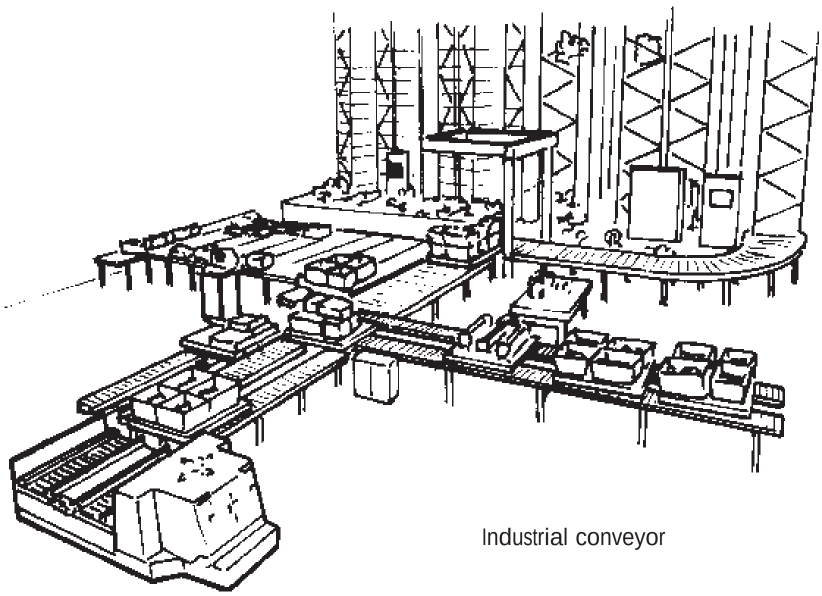
It is often thought that stepper motors are difficult to use and are suitable only for specialised commercial applications. This is not correct. It is very easy to learn how to use and drive a stepper motor using, for instance, the TEP 'Bit by Bit' controller. (This can also be turned into a very powerful dedicated control board by adding external logic etc.) It is also very straightforward to design and make other types of stepper motor controller.



TEP bit by bit controller

There are likely to be many local problems in an industrial, domestic or commercial context where the use of a stepper motor offers an advantage in providing precise positional control. Examples include the following:

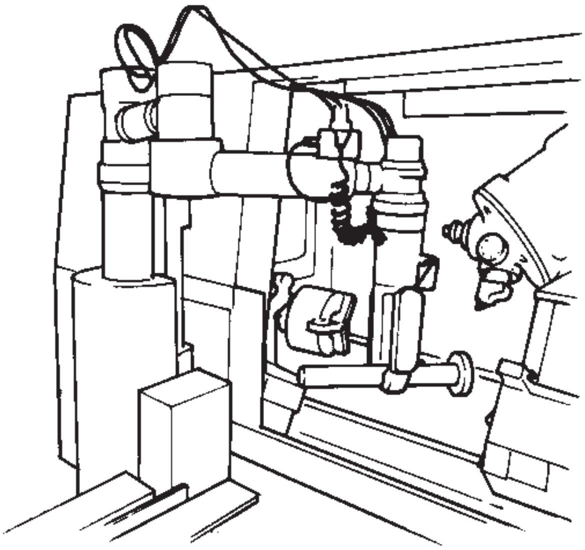
- A small conveyor system driven by a single stepper motor that needs to be started or stopped more or less instantly by a control signal.



Industrial conveyor

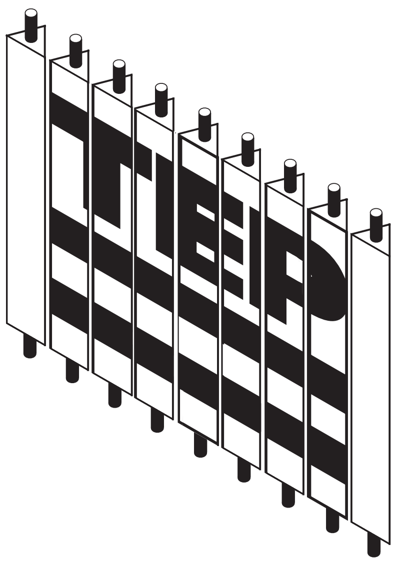
- A stepper-motor-driven linear table which fits onto a drilling machine. The steps of the motor are displayed on a inexpensive electronic counter to show the precise distance moved by the table. Uses of the table include drilling holes at precise spacings for making gear boxes whose gears need to be precisely depthed.

- A 'micro-manipulator' robot using a single stepper motor to transport very small components over short distances. (This is a SCARA - Semi Compliant Articulated Robot Arm - which moves in just one plane.)



Industrial robot actuated by stepper motors

- An advertising system using a number of vertical columns which, viewed as a whole, display an image. When the columns are rotated through an angle, one image fades and another takes its place.



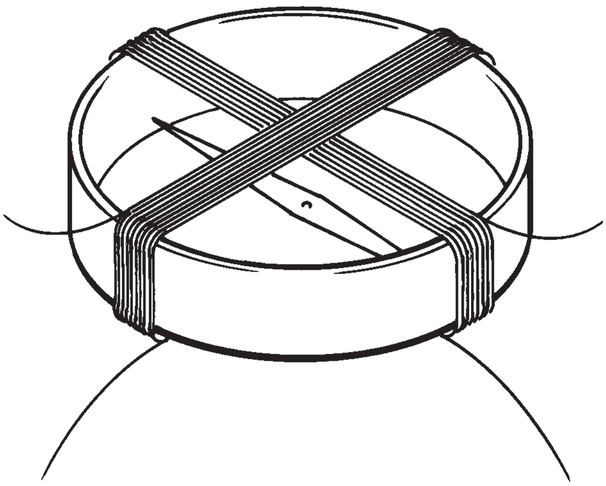
DESIGN SPECIFICATION

When a design brief has been agreed on, it is necessary to draw up a design specification. This is a detailed description of what a product or system would be like, what it will do and who will use it.

If a stepper motor is to be used, you must decide on which commercial type and how to drive it. There are a variety of commercial stepper motors available, each of which comes with its own performance specification. The TEP SM42 is possibly the least expensive, and is described on page 31. You are advised to experiment with this or a similar motor using the TEP 'Bit by Bit' controller board. This will quickly familiarise you with the way stepper motors are controlled. The same board might also be used as a dedicated control board for your project.

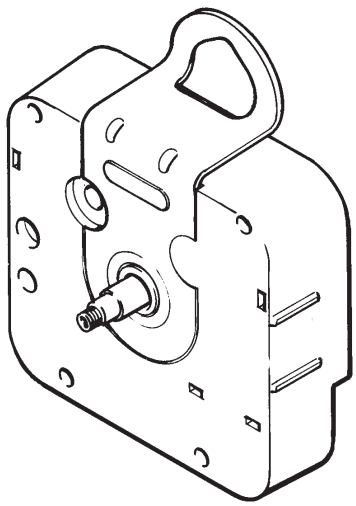
HOW A STEPPER MOTOR WORKS

A stepper motor consists of a permanent magnet rotor placed between a number of stator coils. When these are energised in the correct sequence the magnetic field set up in the coils causes the rotor to move around step by step. A simple demonstration of this principle uses a compass which has two fine copper wire coils wound over it. The compass needle is the equivalent of a very simple stepper motor rotor having just north and south poles. When coil 1 is energised, it produces a magnetic field which causes the needle to line up at right angles to it. When coil 1 is switched off and coil 2 energised, the needle *steps* around through 90° to line up with coil 2. This process can be repeated to keep the needles moving in steps of 90°. The switching sequence or *control signal* can be represented in a diagram.

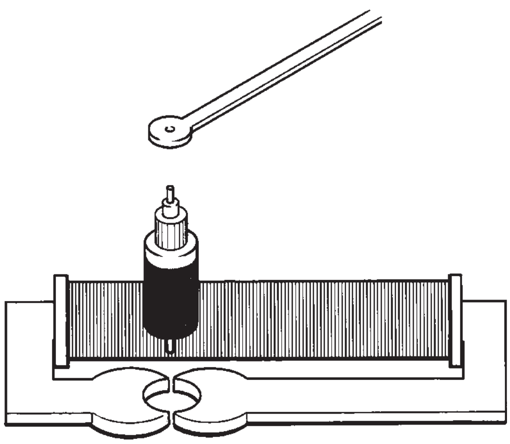


The needle can be turned around through smaller steps of 45° if both coils are energised after each separate coil is turned off. Under these conditions, the needle takes up a halfway position. In stepper motor terminology, this is called *half-stepping.* The switching sequence would be as follows:

The compass ‘motor’ suffers from three problems: it does not produce much torque for useful work, the steps are very coarse or jerky - and someone in the experiment has to switch the coils on and off manually ! It is interesting to add, however, that almost every home contains one or more stepper motors similar in scale to this example. Quartz movement clocks are driven by an electronic circuit that produces very regular pulses of current.

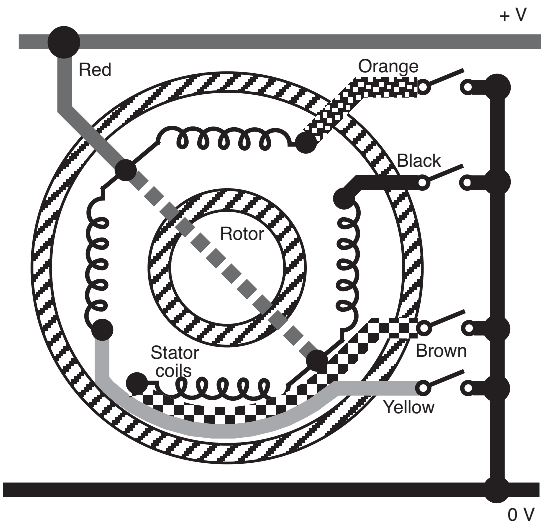


These pulses are fed to a single coil wound around a steel armature. Between the two pole pieces of this armature is a tiny magnetic rotor which rotates in steps as the pulses energise the coil. Typically, the rotor goes around in steps of 90° and is geared to the seconds hand of the clock. If you look closely, this hand moves in small jerks or steps. *The most obvious difference between this motor and the compass experiment is that the rotation is controlled by a regular series of pulses from a circuit.*



Unipolar Stepper Motors

The most common smaller scale stepper motor is the *unipolar* type. This is so called because of the way the stator field coils are arranged and connected. A typical unipolar motor has four stator coils with each of two pairs of coils having a centre tap. Each of the four coils can be energised independently and the motor will need only six connection leads. This can be shown diagrammatically - using manual switches for illustration only. The two centre tapped leads are usually the same colour and are both connected to the positive side of the power supply.



The TEP SM42 stepper motor is a unipolar type (with red centre-tap leads). To make the motor run, its coils have to be energised in the following sequence:

Motor coils

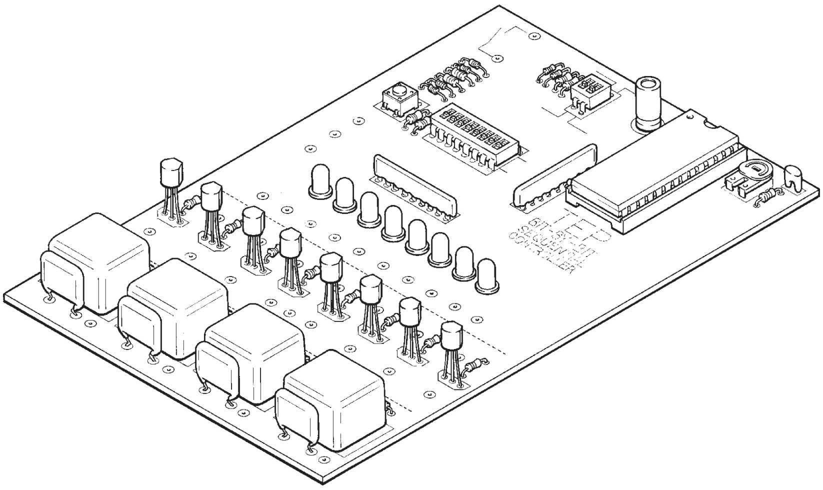
	Yellow	Orange	Black	Brown				
1	on	off	off	on				
2	on	off	on	off				
3	off	on	on	off				
4	off	on	off	on				

If the rotor in this motor had only a north and south pole like the compass, it would only turn in steps of 45°. This is too coarse and jerky for most applications. The solution to the problem is to create a rotor with several N/S poles so that it steps around in smaller increments when the coils are energised. The SM42 actually steps around in increments of 7.5° - i.e. 48 steps per revolution. However, like the compass 'motor', SM42 can be made to rotate in half steps - 96 per revolution - using a different control signal.

It is even possible to obtain smaller steps from a stepper motor using a more complex control signal. This is called *micro-stepping* and provides for very smooth turning and power output.

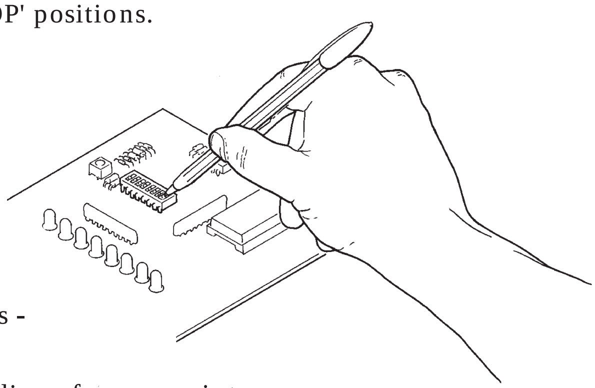
Controlling a Stepper Motor

Commercial stepper motor driver ICs and complete control boards providing the control signal and other functions are available commercially. (These will be dealt with in the last section). Perhaps the easiest way to learn about stepper motor switching and control is to use the TEP 'Bit by Bit' controller which quickly enables you to program a series of outputs to switch on and off in sequence at different speeds. The controller is relatively inexpensive and can also be used as a dedicated stepper motor control board for up to two motors.



The controller is programmed as follows:

1. Make sure the RUN and PROGRAM switches at the top of the board are set at the 'PROG OFF' and 'STOP' positions.
2. Connect the battery or power supply.
3. Set the program switch to 'PROG ON'.
4. Write a line of program by setting each 'PROGRAM DATA' switch to either 'ON' or 'OFF'. This will turn the LED outputs on of off. Because the program switches are small, it is more convenient to operate them with a stylus - e.g., the tip of an empty pen.



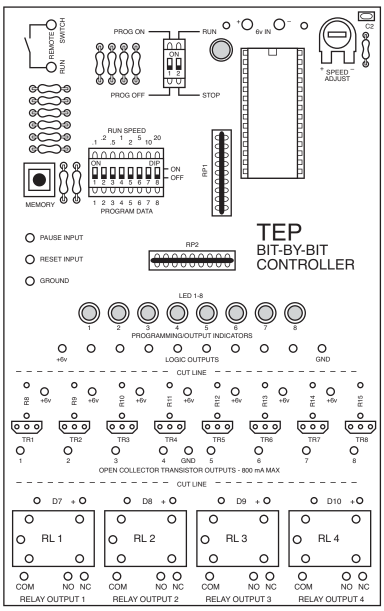
5. Press the 'MEMORY' switch to write this line of program into memory. When you do so, all of the LEDs will flash on briefly to confirm this has happened.
6. Repeat steps 4 and 5 above up to 64 times - once for each page of memory. If you try to go beyond 64 lines of programming, the extreme left hand LED will flash continuously.

   There is no problem if you write a program less than 64 lines. When the program is run, it will loop back to the beginning after the final line.

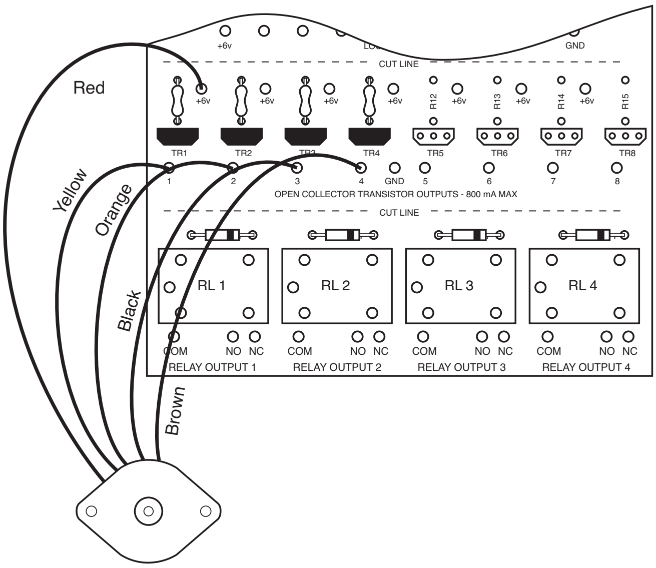
To run the programme:

- Set the program run switch to 'RUN'. The program will now run at approximately 1 line per second. (Setting one of the other 'PROGRAM DATA' switches will run the program at a different speed - see below.) The LEDs will turn on and off according to the stored program in memory.

For further details of setting up and running the 'Bit by Bit' controller, see study file 4.



For single motor use, four output transistors and *clamping diodes* are also added to the left hand side of the board. The diodes prevent any high voltages arising from the coils damaging the transistors and are soldered into the position they would occupy if relays were used.



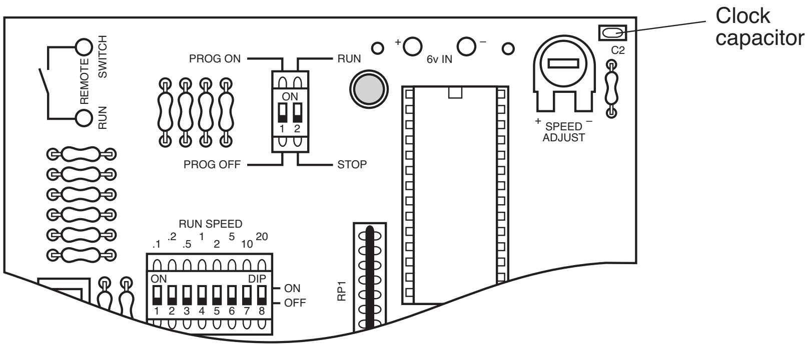
(Note: if two motors are used, clamping diodes for the second motor must also be used. These may be placed on a small additional PCB connecting the second motor to the 'bit by Bit' controller.)

The 'Bit by Bit' controller is programmed with a four line program. This is all that is required to run the motor continuously because any program shorter than the maximum 64 lines simply repeats or 'loops' when run. You might also experiment with half step programming.

Motor coils

	Yellow	Orange	Black	Brown
1	on	off	off	on
2	on	off	on	off
3	off	on	on	off
4	off	on	off	on

The 'Bit by Bit' controller is normally set up to run programs at relatively low speeds. The highest possible speed is still too low for many stepper motor applications - although it is ideal for your first stages of experimentation. To obtain higher run speeds, remove the small 390pF capacitor at the extreme top right hand corner of the board and replace it with a lower value (no lower than 50 pF). This has the effect of increasing the main chip’s clock speed. However, it will also have the effect of flashing the LEDs more rapidly when the memory button in pressed during programming. The controller’s standby current will also increase slightly.

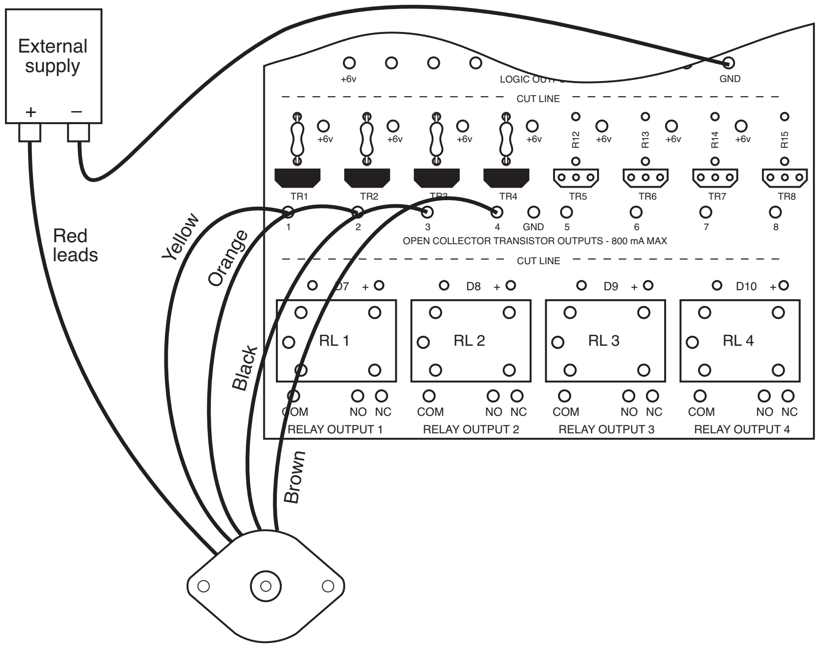


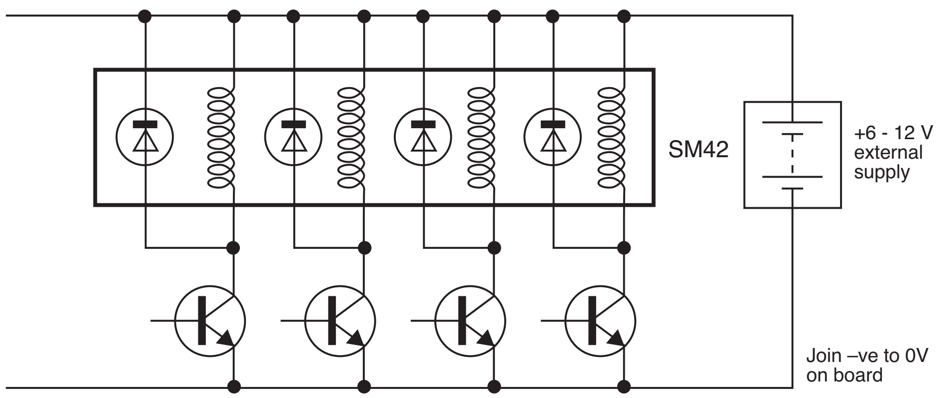
It should also be noted that stepper motors carry a specification for acceleration and deceleration because the rotor cannot reach high speed (or stop) instantaneously due to its inertia. If a high speed signal is suddenly applied, the motor ‘loses steps’ before it catches up with the signal, or over-runs if the signal is suddenly stopped. Commercial controller boards use a circuit (or program) to *ramp* the signal up to speed and ramp down again. The TEP controller does not allow for this and so you should operate the motor within the recommended limits of:

SM42 SPECIFICATIONS

Rated voltage	DC 12V
Working voltage	DC 6V - 12V
Rated current/phase	260 mA
Coil DC resistance	50Ω
Step angle	7.5° step
Holding torque	80g.cm (@12V)
Pull-in torque	270g.cm/200pps (@12V)
Max pull-in pulse rate	365pps

It should also be remembered that even when the switching sequence stops, unless the current is switched off altogether, two coils of the motor will still be energised and drawing current. The advantage of this is that the motor has *holding* torque - or a braking characteristic. The disadvantage for a battery powered system is continuous current consumption even when the motor is not running. If the 'Bit by Bit' controller is used as a dedicated motor driver board then you should ensure that the power supply is adequate or, for example, that the motor is powered by a separate battery or power supply. This is done by connecting into the transistor open collector as shown. **Remember: the maximum voltage of the SM 42 stepper motor is 12v.**

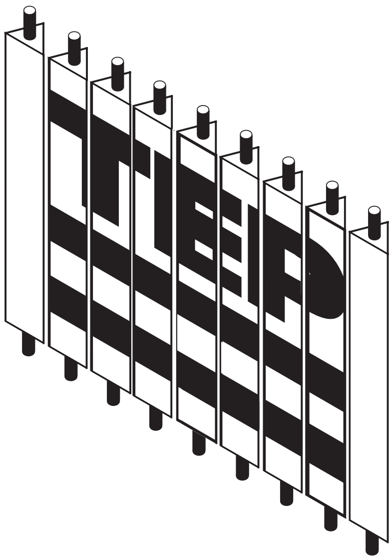




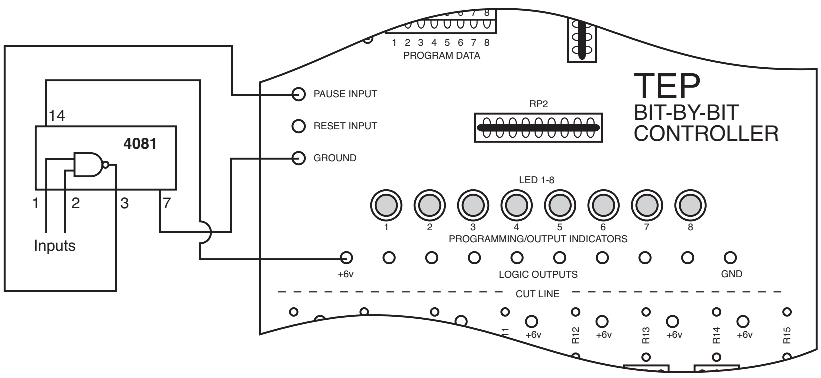
To protect the transistors, clamping diodes (1N4001) should be added across the motor coils as shown. This may entail a separate board to connect diodes between each of the coloured leads and +ve on the external supply.

CONTROL OPTIONS AND POSSIBILITIES

Try programming the controller to drive the motor in different directions within its 64 line program capacity. Turning very slowly step by step a motor can be used in an unusual capacity as a micro-positioner if a low-inertia arm is attached. Remember that if a heavy load is attached, the motor may lose steps. With gearing, an extremely slow running motor can be used for driving display systems such as a multi-column advertising system.



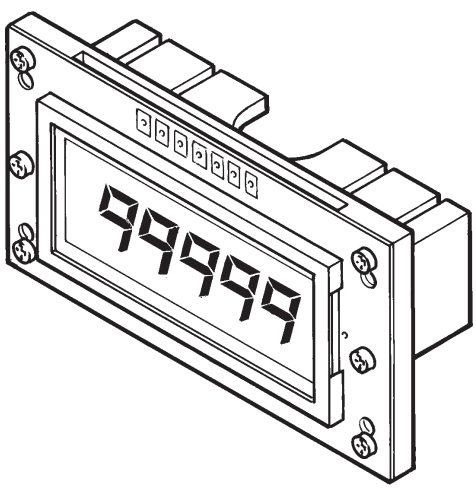
A four line program is all that is needed to run the motor continuously. After this program is executed, the motor can be stopped and started by connecting any +v point on the board to the 'PAUSE INPUT'. Alternatively the program will pause if a logic signal greater than +1.2v is applied. This means that external logic or other circuitry can be added as part of a more sophisticated control system. For example, the motor can be made to stop when, and only when, the two inputs to a dual-input AND gate are taken to logic 1.



If only one motor is being used, the other four outputs might be used either for a synchronised or an independent aspect of control within a larger system. But remember, the program will loop from the highest program line number and this might entail writing in the motor program several times over to keep the motor running in parallel with any other program element.

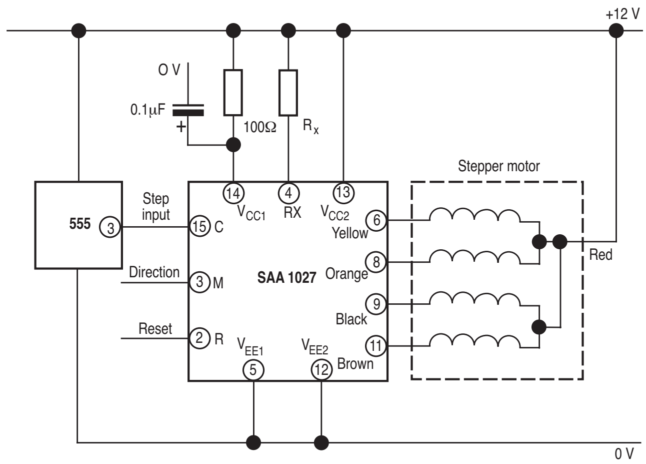
	Stepper motor				Other control functions			
1	on	off	off	on	on	on	off	on
2	on	off	on	off	off	on	on	off
3	off	on	on	off	off	off	on	on
4	off	on	off	on	off	on	off	on
					off	on	on	on
Do not leave blank					off	off	on	off
					on	off	off	off
					on	on	off	on

Other devices can be added to a control system such as a digital readout module. This might be used to indicate the number of steps accumulated, a number of revolutions or, for example, a linear measurement if the motor is connected to a leadscrew or rack and pinion. An inexpensive LCD read-out module (e.g. Teaching Resources stock number ET5 004) can be connected to any 'LOGIC OUTPUT' on the board. This can be the basis for a motor-driven rotary or linear table for small machining operations which require a digital readout.



OTHER METHODS OF STEPPER MOTOR DRIVING

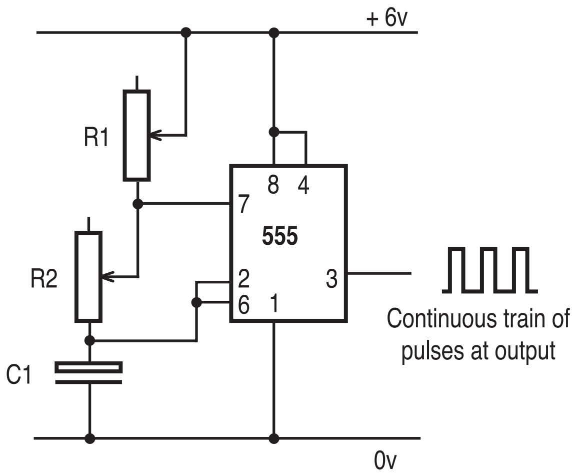
A common method of driving a unipolar stepper motor is by means of the integrated circuit SAA1027. Although small, this contains all the necessary logic and output buffers to drive a motor such as SM42 directly. It does, however, require an external electronic clock to regulate the switching signal. The general arrangement is shown below.



A 555 timer wired in its astable mode will supply the signal needed for the SAA1027. The details are as follows:

The 555 is a general purpose timer I.C. that can be used in one of two modes, either **astable** or **monostable**.

Astable mode - The device will supply a continuous train of pulses at its output. The frequency and duration of the pulses is set by a network of two resistors and one capacitor (R_1, R_2, C_1).

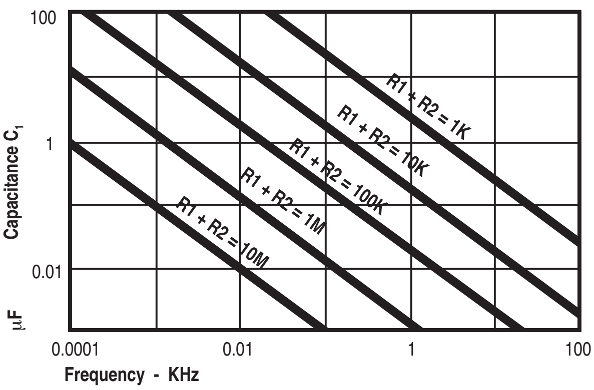


It is possible to calculate the frequency and duration of the pulses using the following formulas:

$$\text{pulse duration} = 0.693\,(R_1+R_2)C_1$$

$$\text{frequency} = \frac{1}{(R_1+2R_2)C_1}$$

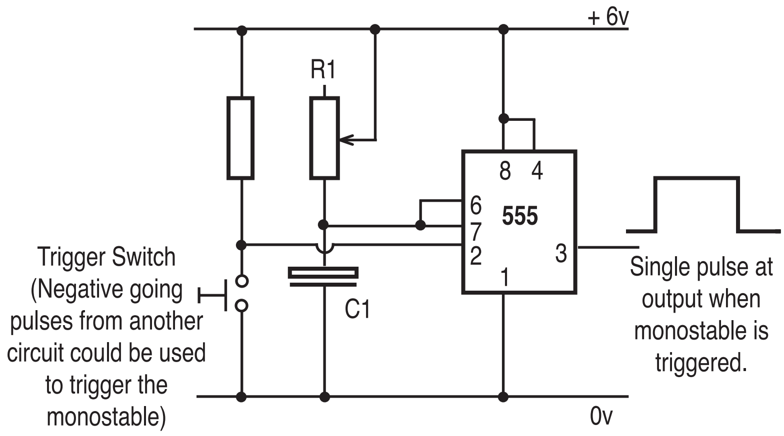
It can be easier, though, to use a table, or series chart, to look up the **very approximate** starting values for R_1, R_2 and C_1.



Simply select the desired frequency on the horizontal axis, draw a line from this point up the graph until you intersect with one of the diagonal resistance lines. This gives you the value for $R_1 + R_2$.

Then draw a horizontal line across from this intersection point to the capacitance values on the vertical axis. This gives you a value for C_1.

Monostable mode - The output from the device will switch from low to high when it is triggered. (Low = 0 volts, high = supply voltage.) The output will stay high for an amount of time that is set by the resistor capacitor network of R_1 and C_1. Once this time has passed the output will go low again. It will stay low until the device is triggered again.

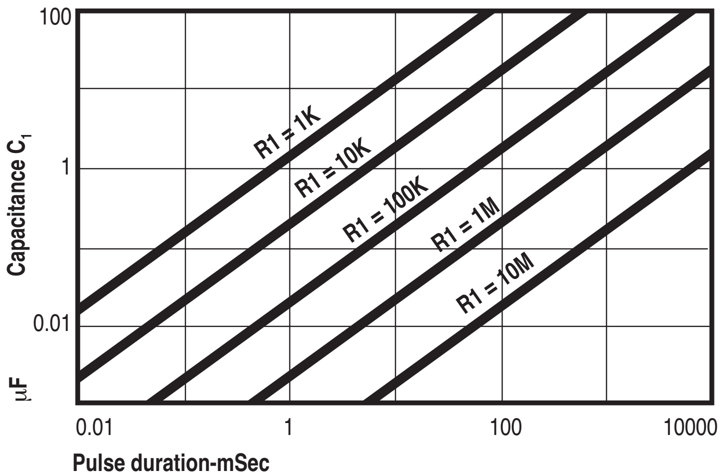


The monostable is triggered by pulling pin 2 down from the supply voltage to 0 volts. This can be done using a switch or by another control circuit.

The duration of the output pulse can be calculated by using the formula:

$$\text{Pulse duration} = 1.1\ R_1C_1$$

Again you can use a series chart to find the values of R_1 and C_1.



Select the desired pulse duration from the horizontal axis. Draw a line up the graph until you intersect with one of the diagonal resistance lines. This gives you the value for R_1. Draw a line across from the intersection point towards the capacitance values on the vertical axis. This gives you the value for C_1.

An alternative method to using SAA27 is to build a driver board based on a decade counter IC such as CMOS 4017. When this device is connected to a clock - e.g. the 555 timer - each of its 10 outputs goes to logic 1 in turn while the other 9 remain at logic 0. If you connected LEDs to each of the outputs, you would see the first LED light up, then the next - and so on.
See Technology Study File 1.

To produce the signal needed to run the motor, the first four outputs are connected via diode links to four output transistors. A diode link represents an 'on' signal and one missing represents an 'off' signal. *Each output (with diodes) from the 4017 represents a line of program*. When an output goes to logic 1, the transistors connected via diodes will 'turn on'.

The full diagram is shown. Please note that output 5 of the 4017 is connected to the reset pin so that only the first four outputs loop in sequence when a clock signal is applied.