PRACTICAL PICAXE

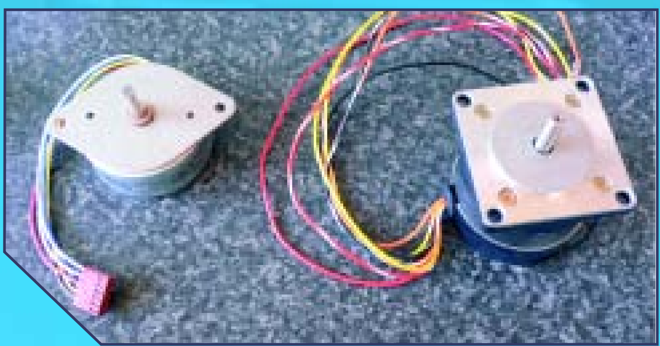
PART 6

Driving Stepper Motors using PICAXE

There will come a time when students either want to run a motor constantly at a very specific speed or through a set angle. The only way this can be achieved is by using a stepper motor. Articles have already been written about the workings of stepper motors so this article will concentrate on how they can be used with PICAXE. The circuits from this article can be downloaded from the TEP website

Inputs

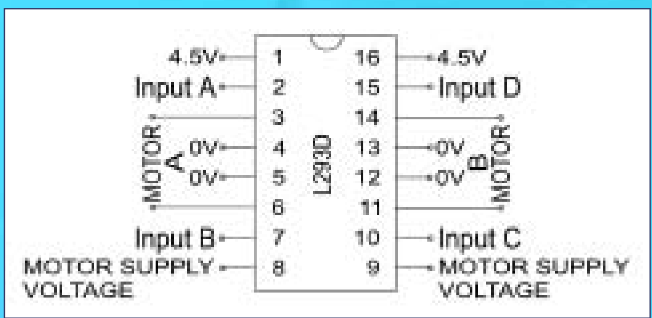
There are two common stepper motors the M425P (which colleagues may have purchased from Teaching Resources in the past) and the T163-3, an industry standard stepper motor and the only one available in the latest Teaching Resources Catalogue. The price of these motors has precluded them from common use in the classroom, £8.80 for the T-163-3, but there are lots in use in relatively cheap products and finding a good source is an ongoing quest. The steps produced by the M425P are 7.5∞ whilst the T-163-3 are1.8∞.



M452P T-163-3

A concern with the T-163-3 is that it may be too accurate for school use if the flowchart programming method is to be used. To turn an angle of 15∞ requires only two stages or steps for the 7.5∞ motors but eight stages for the 1.8∞ motors. This should be considered when choosing which motor and the ability of the pupil must also be taken into account.

As mentioned in other articles the LM 293 D is a stepper motor driver chip that enables pupils to use these motors with relative ease.



Instead of driving two DC motors the four motor outputs are connected to the four coils of a stepper motor. The essential part is getting the correct sequences to achieve the desired outcome.

Single Phase Excitation for the M452P

	1 Black	2 Orange	3 Brown	4 Yellow
Step 1	ON	OFF	OFF	OFF
Step 2	OFF	ON	OFF	OFF
Step 3	OFF	OFF	ON	OFF
Step 4	OFF	OFF	OFF	ON

This sequence is simple but gives low torque. Reverse programme to reverse direction – i.e. output 4 is turned on first the 3 and so on.

Dual Phase Excitation for the M452P

	1 Black	2 Orange	3 Brown	4 Yellow
Step 1	ON	ON	OFF	OFF
Step 2	OFF	ON	ON	OFF
Step 3	OFF	OFF	ON	ON
Step 4	ON	OFF	OFF	ON

This sequence gives high torque. Reverse programme to reverse direction. For these motors keep the running speeds relatively low, otherwise the motor will tend to overrun when stopped and stall when it is turned on. The red wire goes to positive power supply.

For the 55S1 Motors the same sequences but the colour order is different.
The two white wires are twisted together and insulated with tape.

	1 Brown	2 Blue	3 Red	4 Yellow
Step 1	ON	OFF	OFF	OFF
Step 2	OFF	ON	OFF	OFF
Step 3	OFF	OFF	ON	OFF
Step 4	OFF	OFF	OFF	ON

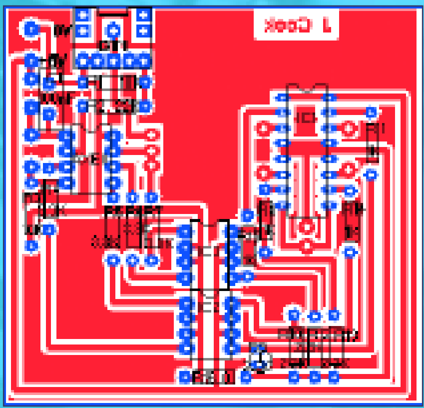
For the T163-3 Motors use the same sequences but the colour order is different.
The four striped leads are connected together and go to the positive supply.

	1 Black	2 Yellow	3 Orange	4 Red
Step 1	ON	OFF	OFF	OFF
Step 2	OFF	ON	OFF	OFF
Step 3	OFF	OFF	ON	OFF
Step 4	OFF	OFF	OFF	ON

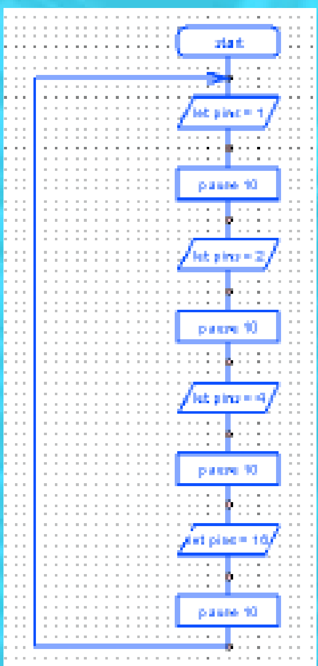
Which stepper motor is used, the correct sequencing and deciding the type of excitation to be used need to be established at the outset. With so many permutations, only the information necessary should be given to the pupils if confusion and frustration are to be avoided.

The PCB's have been produced using PCB Wizard, but will also be available in Real PCB from Crocodile Technology, their latest piece of software. The layouts continue to follow the modular approach as described in previous articles.

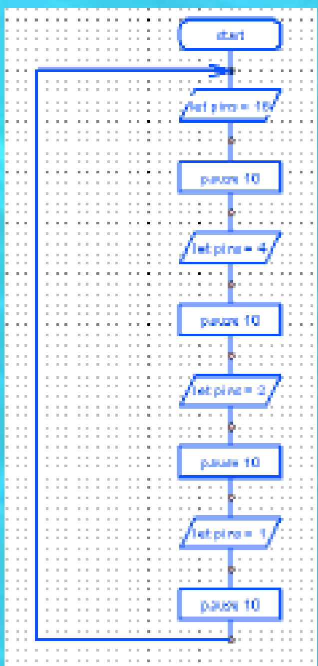
If all you want to do is drive a stepper motor at a constant speed you could use the PICAXE 08. The PCB layout is shown below.



This uses four outputs and leaves one input that could be connected to a switch to set the programme running.



Forward

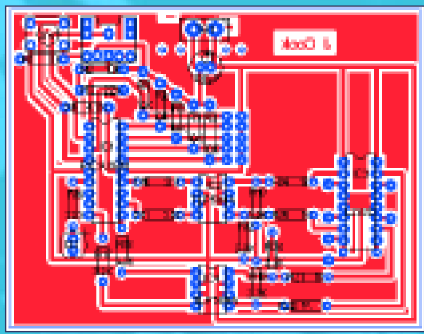


Reverse

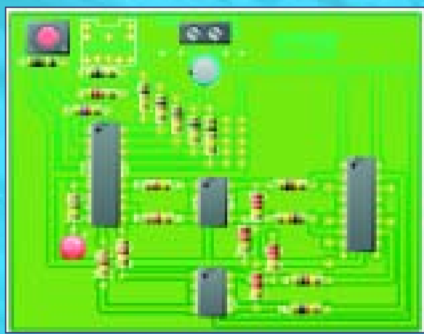
```
'BASIC converted from flowchart:
'C:\JOHN COOK\JC'S STUFF\TEP\STEPPER DRIVE.CAD
'Converted on 19/05/2005 at 11:00:01

main:
label_6:    let pins = 16      ' %00010000
            pause 10
            let pins = 4       ' %00000100
            pause 10
            let pins = 2       ' %00000010
            pause 10
            let pins = 1       ' %00000001
            pause 10
            goto label_6
```

Use the pause command that is in milliseconds to get very accurate speeds of rotation and high speed steps. These can easily be converted to Basic in the usual way.

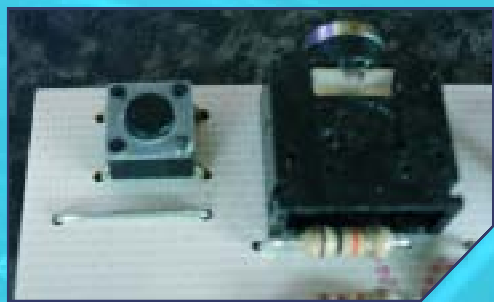
If more complex programming requirements are needed then, unless you are able to programme direct in basic, the safe option is to use the PICAXE 18. The range of three chips with ever increasing memory size, allows students to produce the most complex flow charts and still be able to download them successfully to their PCB.



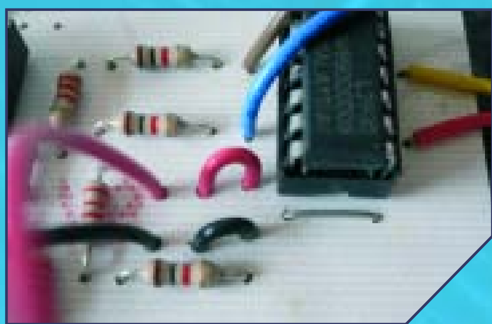
Normal View



Real World View



A wire link is required to replace the 0 ohm resistor below the reset switch.



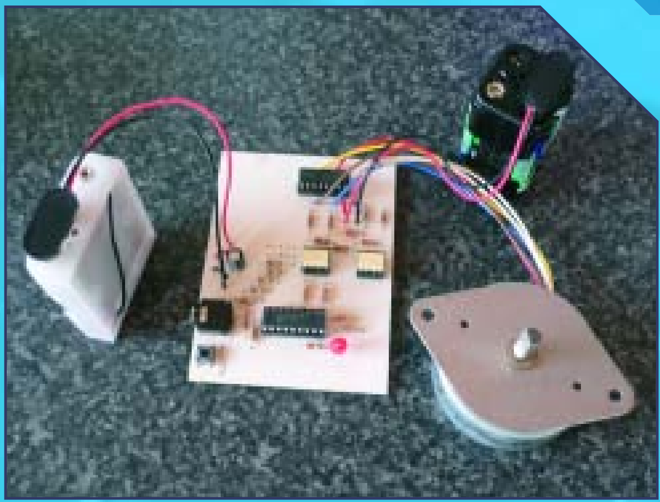
A wire link is required across the two pads at the bottom of the motor driver chip.

Continued overleaf

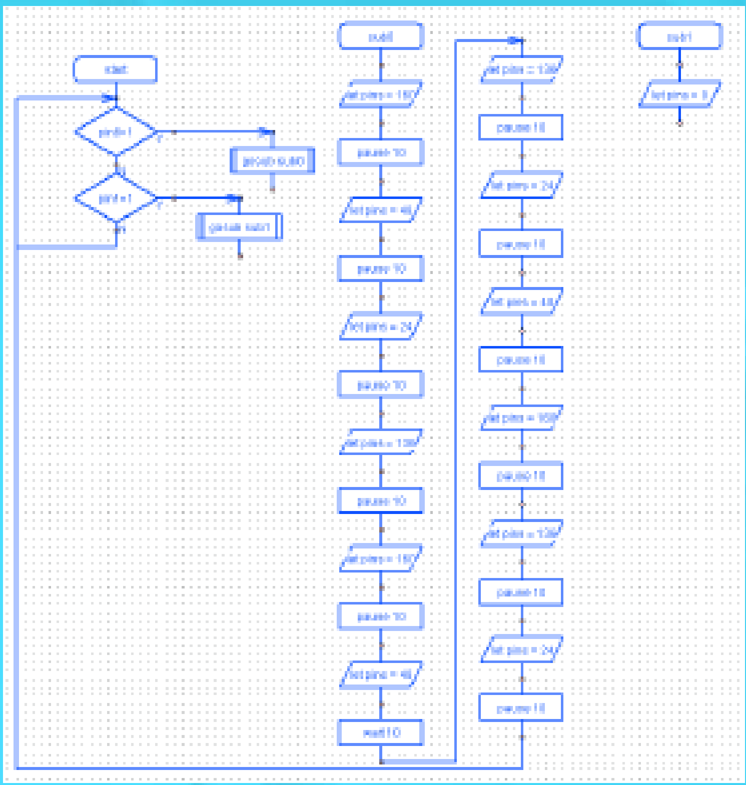
PRACTICAL PICAXE

PART 6

Continued



This is the completed circuit board. Note the separate battery supply for the stepper motor driver chip. This is essential as the maximum voltage for all PIC's is 5v and most stepper motors require 6v, but it also reduces the chance of voltage drops on starting that may cause the programme to hang up.

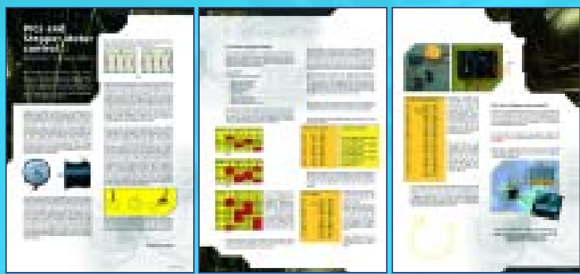


This is a programme I produced for a TEP Colleague who wanted to operate a pneumatic valve. To operate the valve required high torque so dual phase excitation was used, hence the large values in the pinout boxes. The second subroutine had to be left off because the programme was too big for a standard PICAXE 18.

```
'BASIC converted from flowchart:
'C:\JOHN COOK\JC'S STUFF\TEP\STEPPER VALVE.CAD
'Converted on 18/05/2005 at 11:28:12

main:
label_6:    if pin0=1 then label_10
            if pin1=1 then label_26
            goto label_6

label_10:   gosub sub0

label_26:   gosub sub1

sub0:
            let pins = 160      ' %10100000
            pause 10
            let pins = 48       ' %00110000
            pause 10
            let pins = 24       ' %00011000
            pause 10
            let pins = 136      ' %10001000
            pause 10
            let pins = 160      ' %10100000
            pause 10
            let pins = 48       ' %00110000
            wait 10
            let pins = 136      ' %10001000
            pause 10
            let pins = 24       ' %00011000
            pause 10
            let pins = 48       ' %00110000
            pause 10
            let pins = 160      ' %10100000
            pause 10
            let pins = 136      ' %10001000
            pause 10
            let pins = 24       ' %00011000
            pause 10
            goto label_6

sub1:
            let pins = 0        ' %00000000
```

In circumstances like this the ability to simply use a chip with the same configuration but a bigger memory such as the 18A or 18X has to be a good reason for using PICAXE.

John Cook's PCB designs are available from the TEP website under the News and Views support tab. Further useful background on PICs and Stepper Motors appeared in Issue 8 of News and Views and can be downloaded from the TEP website too.



PICs and Stepper Motor Control – N&V 8 page 25

For further help or assistance you can email John Cook at: jcooklggs@hotmail.com