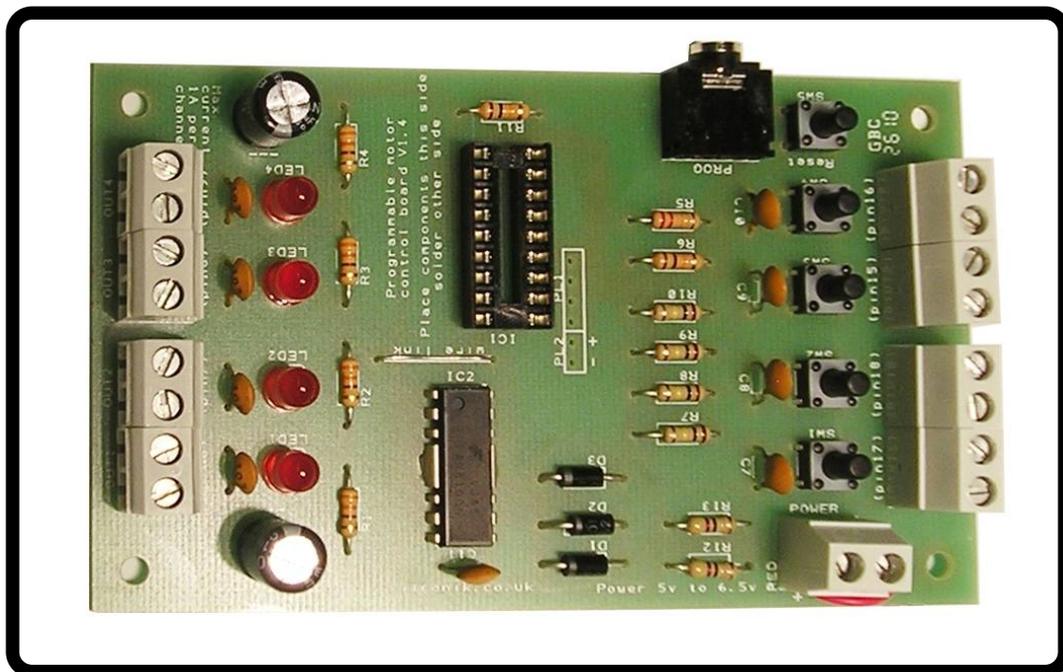




Programmable Motor Controller

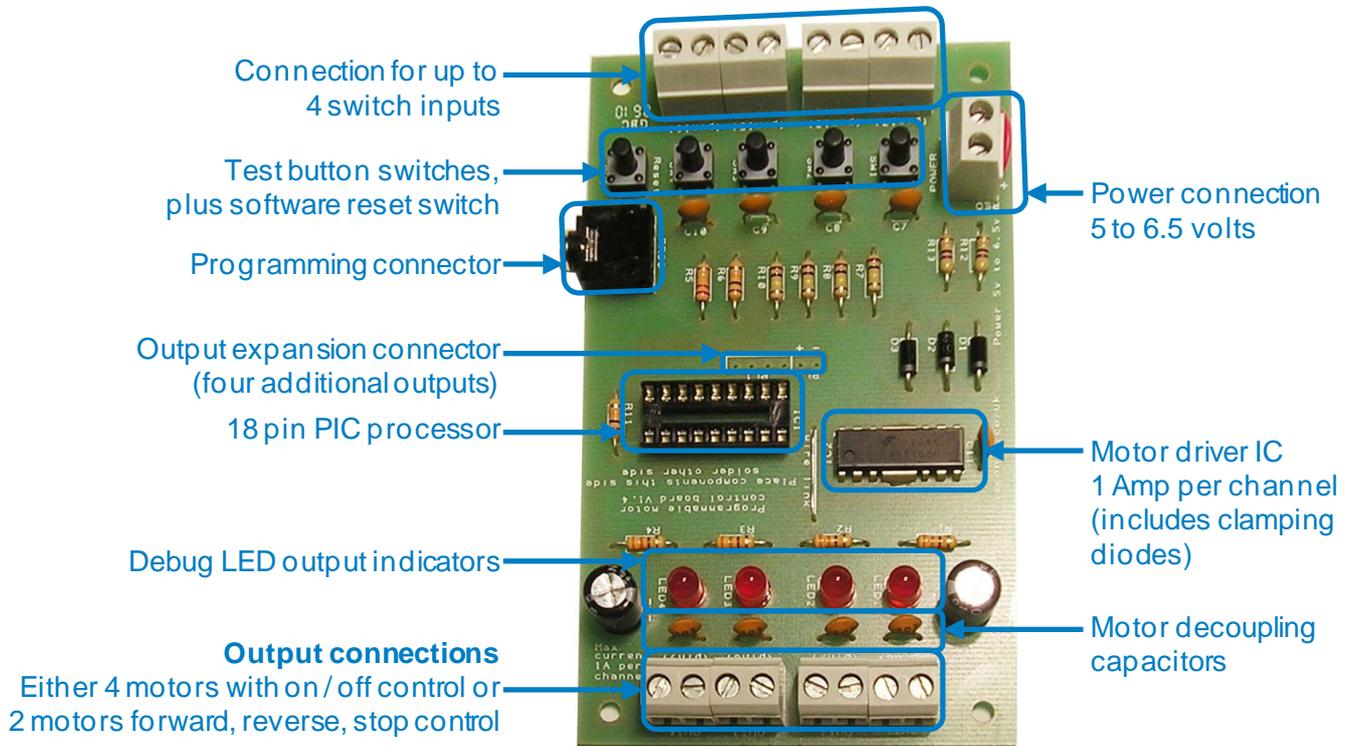


Build Instructions

Issue 1.4



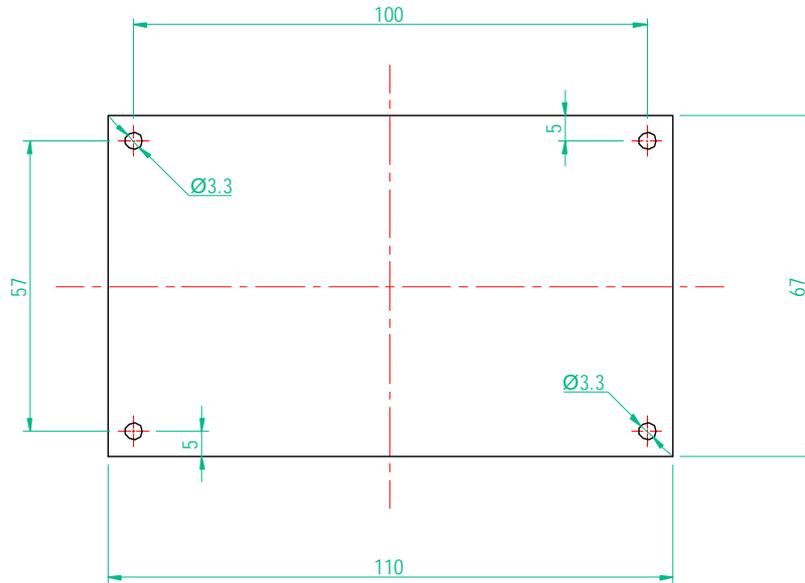
Features of the motor controller board



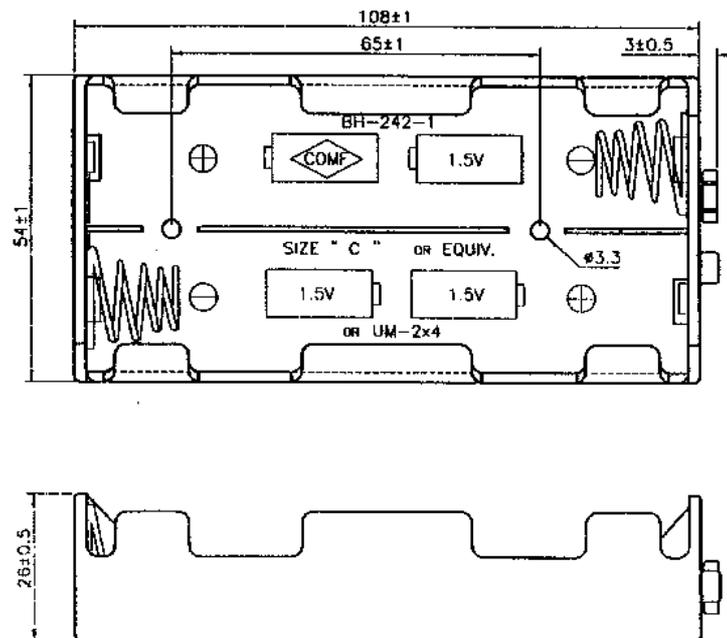


Mechanical information

All dimensions are in mm.



Motor controller PCB



Battery holder x4 C cell



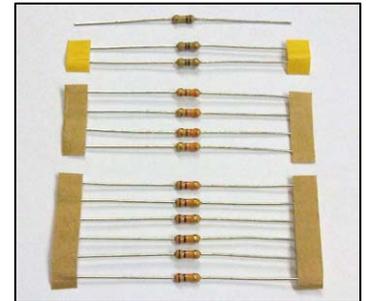
Build instructions

Before you put any components in the board or pick up the soldering iron, just 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. You will find it easiest to start with the small components and work up to the taller larger ones. If you've not soldered before get your soldering checked after you have done the first few joints.

Step 1

Start with the 13 resistors (shown right):
The text on the board shows where R1, R2, etc go. Make sure that you put the resistors in the right place.

R1- R4 are 330 Ω (Orange, Orange, Brown coloured bands).
R6 is a 100K Ω (Brown, Black, Yellow coloured bands).
R5 & R7 - R11 are 10K Ω (Brown, Black, Orange coloured bands).
R12 - R13 are 47 Ω (Yellow, Purple, Black coloured bands).

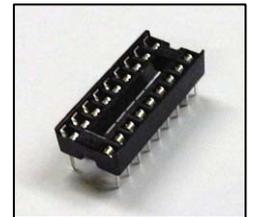


Step 2

Solder the 3 diodes (shown left) into the board. When putting these into the board, be sure to get them the right way around. The band on the diode must match up with the band on the PCB. The diodes are marked D1 to D3.

Step 3

Solder the IC holder (shown right) into IC1. When putting it 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.

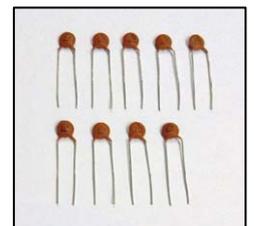


Step 4

Solder the programming connector (shown left) into the board where it is labeled 'PROG'.

Step 5

Solder the nine ceramic disc capacitors (shown right) into the board, where it is labeled C3 - C11. All nine capacitors are the same value (10nF and marked 103).



Step 6

Solder the five switches (shown left) in to the board where it is labeled SW1-SW5. Once you have got the pins lined up with the holes they can be pushed firmly into place and soldered.

Step 7

Solder the Light Emitting Diodes (LEDs) (shown right) in to LED1 - LED4. The LEDs won't work if they don'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.
Please note: Don't use the red & green LED in the separate bag (they are 5v LEDs, and won't work).





Build instructions continued

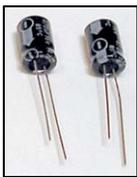


Step 8

Solder the four, 4-way terminal blocks (shown left) into the board. Two are for the inputs and two are for the outputs. Make sure the side that the wire goes into is facing the edge of the board.

Step 9

Solder the 2-way terminal block (shown right) into the 'POWER' connection.



Step 10

Solder the two big electrolytic capacitors (shown left) into C1 & C2. These need to be put in the board the right way around. The white band on the two large capacitors should line up with the '---' markings on the board.

Step 11

Using a resistor leads add a link to the board where it is labeled 'wire link'.

Step 12

Using a small screwdriver connect the battery clip (shown right) into the 'POWER' terminal block. The red wire must go to the '+' terminal (also marked 'red') and the black wire must go to the '-' terminal (also marked 'black').



Step 13

Clip the heat sink on to the FAN 8100N motor driver. **This must be done before it is inserted into the PCB as it cannot be put on afterwards.** Now insert the IC into the PCB where it is labeled IC2 ensuring the notch on the chip lines up with the notch on the PCB outline. It is hard to see the outline on the chip once the heat sink is on top, so take care when doing this.

Step 14

Your choice of 18 pin PIC chips can now be put into the holder labeled IC1 ensuring the notch on the chip lines up with the notch on the holder.

Checking your motor controller 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) are filled with the lead of a component.
- All these leads are soldered.
- Pins next to each other are not soldered together.

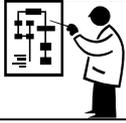
Check the top of the board to ensure that:

- The notch on the ICs and the IC holder are in the same orientation as the markings on the printed circuit board.
- The four resistors R1-R4 have Orange, Orange, Brown coloured bands.
- R12 and R13 have Yellow, Purple, Black coloured bands.
- R5 has Brown, Green, Orange coloured bands.
- The silver bands on diodes D1 and D2 are next to IC2 and the silver band on D3 is next to C7.
- The four LEDs are in the right way around (the flat edge on the LED matches the markings on the board).
- The white band on the two large capacitors match the '---' markings on the board.
- The red wire on the battery connector goes to the '+' terminal on the power terminals and the black wire goes to the '-' terminal.



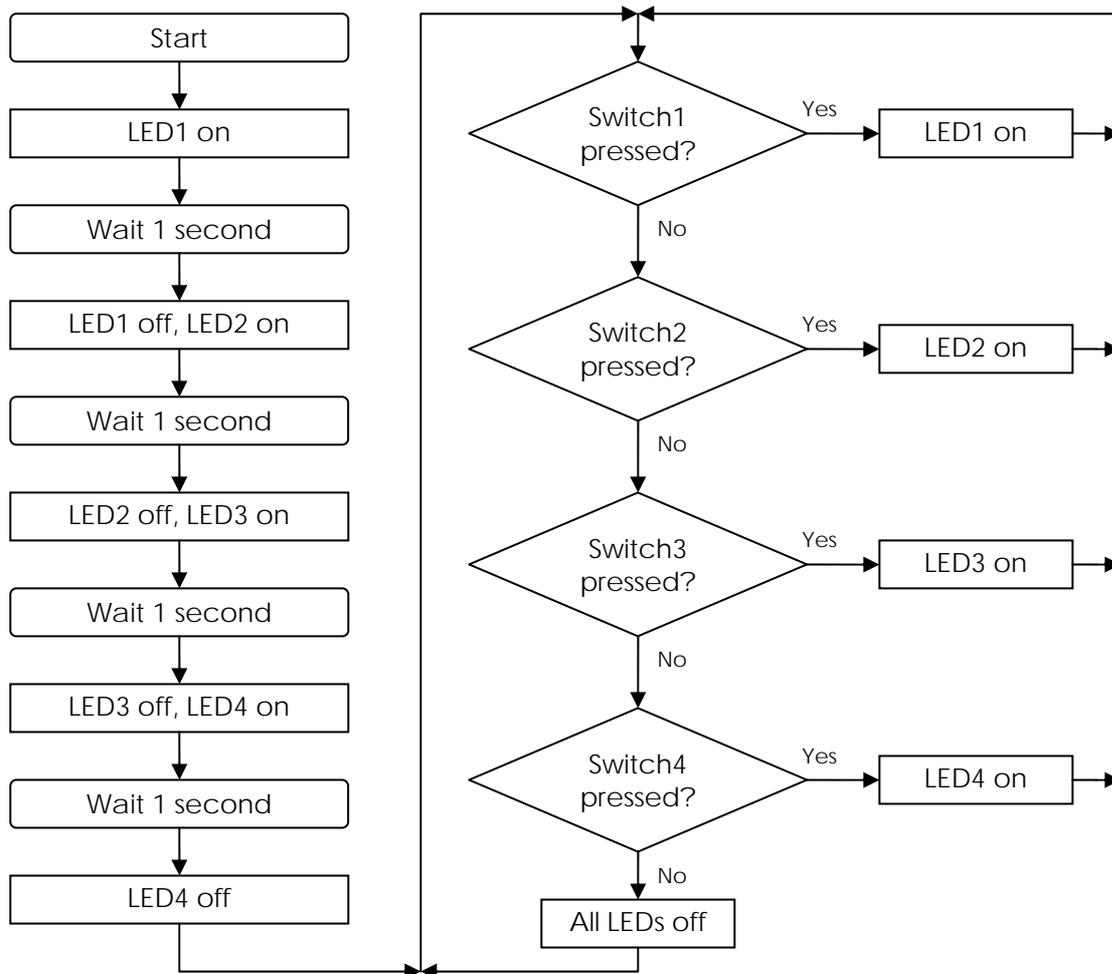
Parts list

Ref	Description	Qty
R1 - R4	330R resistor 5%	4
R6	100K resistor 5%	1
R5 & R7 - R11	10K resistor 5%	6
R12 - R13	47R resistor 5%	2
D1-D3	1N4001 diode, 1A	3
IC1	18 pin DIL socket	1
IC2	FAN8100N motor driver IC	1
PROG	3.5mm stereo PCB mount jack socket	1
C1, C2	470uF electrolytic capacitor	2
C3 - C11	Capacitor, ceramic, 50V, 10nF	9
SW1 - SW5	6mm PCB mount switch - height 9.5mm	5
LED1 - LED4	LED red 5 mm	4
POWER	2 way terminal block	1
INPUT1 - INPUT4, OUTPUT1 - OUTPUT4	4 way terminal block	4

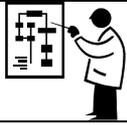


Testing the PCB

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 LEDs can turn on and off and that the switches are functioning. The test software works as shown below:

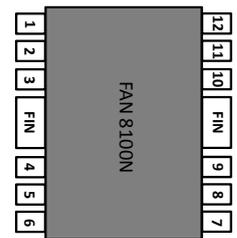


If you have problems with any of the above tests, use the fault finding tables to find the cause of the fault.



Fault finding table

The image to the right shows the pin layout for the motor driver IC.
The following tables list which parts to check for various fault conditions:



An LED didn't work as expected

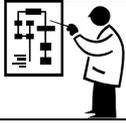
LED number	Parts to check
LED1	R1 for dry joints LED1 is in the right way around, for dry joints or shorts For dry joints or shorts on IC1 pin 6, IC2 pins 8, 10 & 12 The wire link between IC1 & IC2 is present and for dry joints
LED2	R2 for dry joints LED2 is in the right way around, for dry joints or shorts For dry joints or shorts on IC1 pin 7, IC2 pins 8, 9 & 11
LED3	R3 for dry joints LED3 is in the right way around, for dry joints or shorts For dry joints or shorts on IC1 pin 8, IC2 pins 2, 4 & 5
LED4	R4 for dry joints LED4 is in the right way around, for dry joints or shorts For dry joints or shorts on IC1 pin 9, IC2 pins 1, 3 & 5

A switch didn't work as expected

Switch number	Parts to check
SW1	SW1 for dry joints R7 for dry joints For dry joints or shorts on IC1 pin 17
SW2	SW2 for dry joints R8 for dry joints For dry joints or shorts on IC1 pin 18
SW3	SW3 for dry joints R9 for dry joints For dry joints or shorts on IC1 pin 15
SW4	SW4 for dry joints R10 for dry joints For dry joints or shorts on IC1 pin 16

Other fault conditions

Fault	Parts to check
No LEDs work	The 'POWER' connector is in the right way around and for dry joints on the connections The power source is good The processor is programmed with the test program The notch on both IC1 & IC2 line up with the outline on the board For dry joints on D1, D2, or D3 For dry joints on IC1 pins 5 & 14 For dry joints on IC2 R12 & R13 for dry joints R12 & R13 are 47R resistors (Yellow, Purple, Black)
The board won't program	R5 & R6 for dry joints R5 is a 10K resistor (Brown, Black, Orange) R6 is a 100K resistor (Brown, Black, Yellow) For dry joints on the 'PROG' connector For dry joints or shorts on IC1 pins 3 & 7 The notch on IC1 is next to R11



Fault finding tables continued

Other fault conditions continued

Fault	Parts to check
The reset switch doesn't work	SW5 for dry joints R11 for dry joints For dry joints or shorts on IC1 pins 4

With motors connected, the software restarts when the following motors begins to spin:

Output number	Parts to check
All outputs	C1 & C2 for dry joints
Output1	C3 for dry joints The wire link next to IC2 is present and for dry joints
Output2	C4 for dry joints
Output3	C5 for dry joints
Output4	C6 for dry joints

The PCB mount test switch or LED works, but the peripheral doesn't

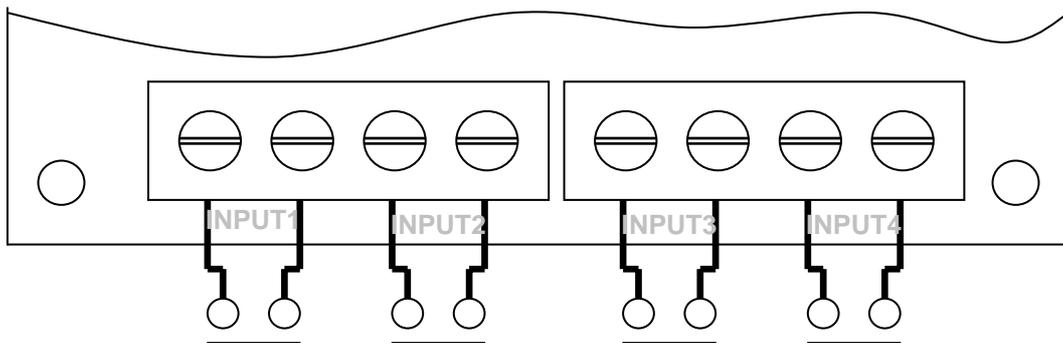
Check the corresponding terminal block for dry joints



Board connections

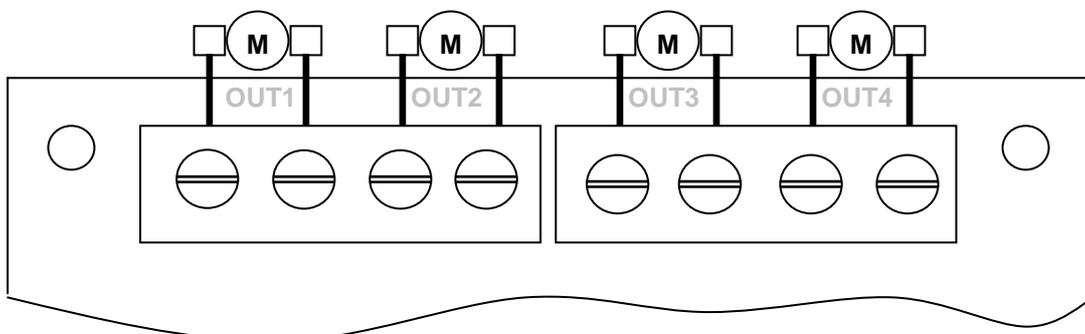
Input connectors

The following diagram shows how to connect up to 4 input switches to the board:



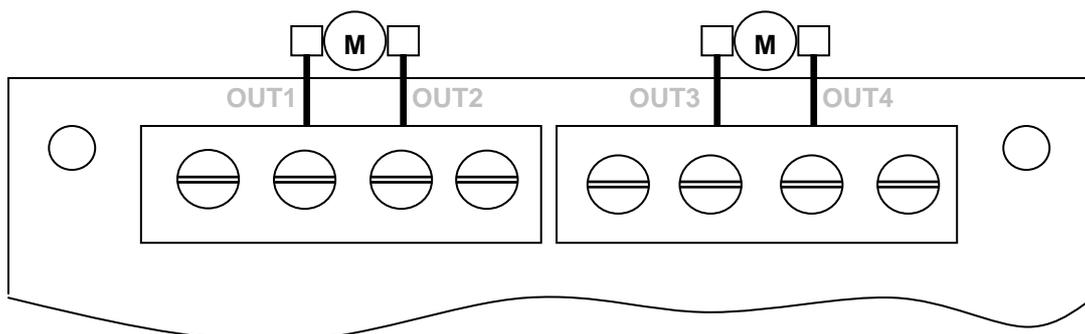
Output connectors – 4 motors on / off control

The following diagram shows how to connect up to 4 motors, where the motor just needs to be turned on or off: (In this configuration each motor should be driven on their own. It is not possible to drive outputs 1&2 at the same time, nor can outputs 3&4 be driven at the same time).



Output connectors – 2 motors forward, reverse, stop control

The following diagram shows how to connect up to 2 motors, where full forward, reverse and stop control is required:

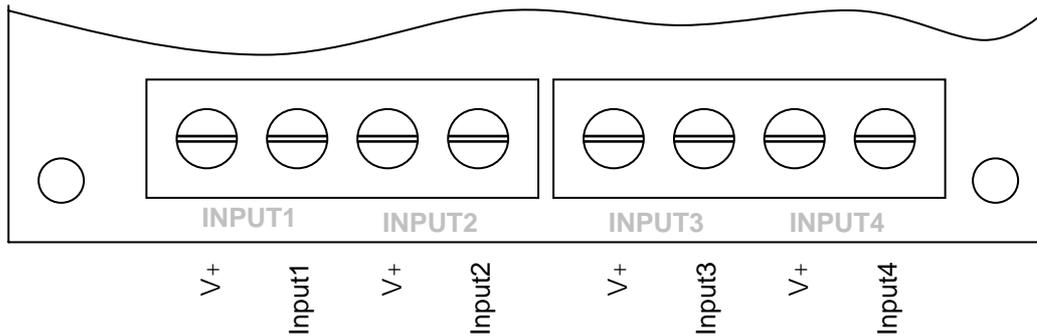




Advanced board connections

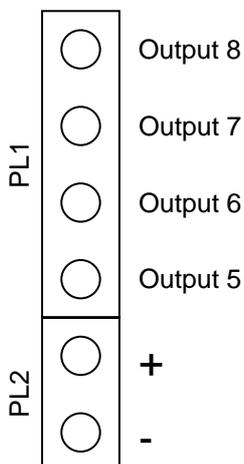
Input connectors

Although the input connectors are designed for switches to be connected to them, a logic output from a sensor or separate board can be connected. Each pair of connections has a V+ connection and the input connection. Other electronics can be connected to the board, just attach the ground or '-' to the '-' on the 'POWER' connector, then attach your inputs as shown:



Output expansion connector

The processor has four extra outputs in addition to the four outputs used to control the motors. A 6-way connector on a tenth of an inch spacing can be used to connect a separate board or other output devices to the board. The maximum current for each of these pins is 25mA. Power connections are also present on this connector, the pins are as follows:





Controlling motors

On / off control

One side of the motor is connected to 0 Volts, the other side is connected to a processor pin, via the driver IC. The following table indicates what the motor will be doing depending upon what the software has set the output pin to:

State of output pin	Motor
Set, high, 1	Spinning
Cleared, low, 0	Stopped

Direction control

In this set up either side of the motor is connected to a processor pin again through the driver IC. In this configuration the processor pins can be set to allow the motor to spin in either direction, as follows:

State of output pin1	State of output pin2	Motor
Cleared, low, 0	Cleared, low, 0	Stopped
Cleared, low, 0	Set, high, 1	Spinning forward
Set, high, 1	Cleared, low, 0	Spinning reverse
Set, high, 1	Set, high, 1	Stopped

Software port names

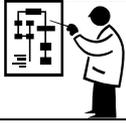
Unfortunately different software packages use different notations for the name they give to a particular port pin. If the software allows you to set the name of the pin change it to reflect what the pin controls, if not you may wish to write the name / number used by your software in the end column of these tables.

Inputs

PCB marking	Pin number	Microchip port name	Software name
Input1	17	RA0	
Input2	18	RA1	
Input3	15	RA6	
Input4	16	RA7	

Outputs

PCB marking	Pin number	Microchip port name	Software name
Output1	6	RB0	
Output2	7	RB1	
Output3	8	RB2	
Output4	9	RB3	
PL1 (Output5)	10	RB4	
PL1 (Output6)	11	RB5	
PL1 (Output7)	12	RB6	
PL1 (Output8)	13	RB7	



How the motor controller works

The motor drive board is based around two IC's. One is a PIC and the other is motor driver IC.

PIC

The PIC has four inputs and eight outputs. Four of the outputs are used to control the motor driver IC and the other four are left spare. You can use these four outputs to add extra functionality should you wish to.

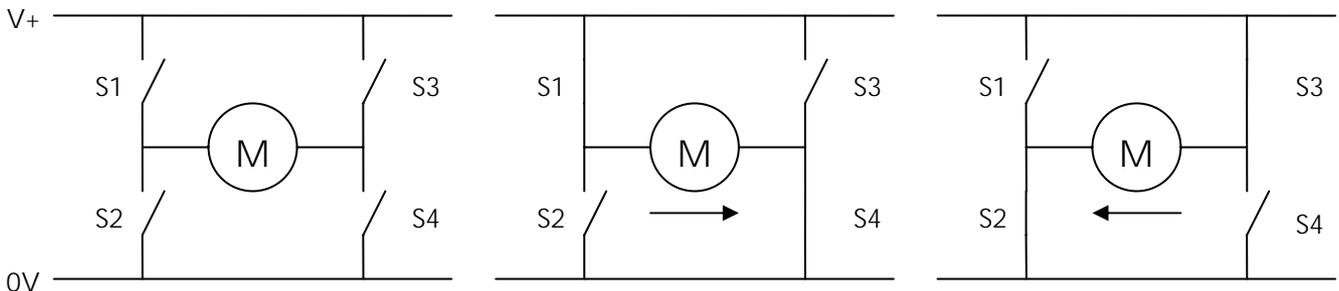
The four inputs to the PIC are held at a low voltage by pull down resistors. Each of these inputs has a test switch. When these are pressed (closed) the PIC pin they feed changes to a high voltage. You can use these switches if you wish or you can add your own (open going closed on activation) switching device to the input terminals.

Motor driver IC

The driver IC is used to drive the motors as they are high current inductive loads which you would not be able to drive directly from the PIC.

H-bridge

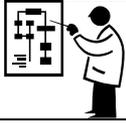
An H-bridge is a type of circuit that allows a DC motor to be driven forwards and in reverse. The circuit derives its name from the way in which the circuit is drawn, which looks like the letter 'H'.



As you can see from the diagram above the circuit consists of four switches. To run the motor in the forward direction switch S1 and S4 are closed while the others are left open. This creates a forward voltage across the motor (see the middle diagram). To run it in reverse switches S2 and S3 are closed while the others are left open. This creates a reverse voltage across the motor (see the right hand diagram).

S1	S2	S3	S4	Result
1	0	0	1	Motor forward
0	1	1	0	Motor reverse

Each of the outputs from the motor driver IC is a half H-bridge output. This means it performs the job of two of the switches (S1 + S2 or S3 + S4). Although the diagram has been drawn with switches, inside the driver IC these switches are implemented with transistors. They are connected such that one transistor is always on and the other is always off. Two of these half bridges can be used as shown above to drive a motor with both forward and reverse functions. In this configuration the board can drive a total of two motors. If you only want to drive motors in one direction each of the four outputs can be connected to a separate motor (the other side of the motor should be connected to 0V) allowing four motors to be used.



How the motor controller works continued

Each of the four motor driver outputs also has:

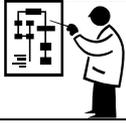
- Clamping diodes to prevent the high voltage spikes that are created when a motor is turned on and off from getting on to the power rails. These can't be seen on the circuit diagram as they are built into the motor driver IC.
- De-coupling capacitor to remove the noise on the power rails that is created when the motor is running.
- An LED (with current limit resistor) for visual feedback. These let you know when one of the outputs is high which will help when developing and testing your software.

Motors are electrically very noisy, this means that they cause spikes to be picked up on the power supply, when they start, stop and also when they are running. Without the clamping diode and de-coupling capacitors the circuit would exhibit unusual behavior and can cause the PIC to reset, which is why these parts are needed.

Other items

There are a couple of other parts in the circuit:

- There is a programming connector which allows the PIC to be programmed. This is connected via a resistor network that limits current and holds the input low whilst not in use.
- There is a switch connected to the reset input on the PIC. Normally this pin is held high by a resistor, but it is pulled down to 0 volts when switch is pressed. This causes the PIC to be reset.
- This circuit runs off a split supply, this is to keep the power for the processor 'clean' even though the supply that drives the motors is noisy. The diodes D1 – D3 and resistors R12 & R13 split the supply, with C1 and C2 being used to smooth the separated supplies when the high current devices (such as motors) are switched on and off.



Sequence controller setup instructions

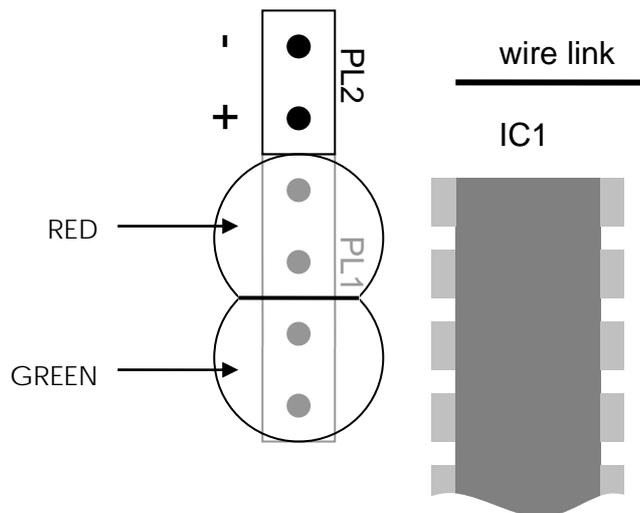
Your motor controller kit is supplied with two 5 volt LEDs, one red one green. You can add these to the board, download some free software and turn your motor controller into a standalone sequence controller. The sequence controller allows the user to enter a sequence of output patterns and then have them replayed at the required speed. The LEDs are fitted on the expansion connector so you can't use the additional four outputs, but you can still use the board with PC flowchart software.

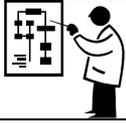
The sequence controller software has been written in Revolution basic and is designed to be run on a PICAXE-18X chip. Free programming software is available from the Revolution website or for your convenience a CD can be purchased from Kitronik. Alternatively if you have Logicator software, you can add a basic box to a flowchart to program the chip.

This sequence controller software can be downloaded from:
www.kitronik.co.uk/quicklinks/motor.htm

Fitting the program / run LED indicators

You must use the 5 volt LEDs supplied in the kit; you will destroy normal LEDs if you fit them. Connect up the red and green LED as shown in the diagram below:





Using the sequence controller

Modes

The sequence control is either in run mode when the red run LED is illuminated, or is in program mode when the green program LED is illuminated.

Changing modes:

Program mode to Run mode - Press and hold one of the switches (SW1 to SW4) for two seconds.

Run mode to Program mode - Press one of the switches (SW1 to SW4).

Program mode

During program mode the green program LED will be illuminated.

To set the output LEDs press switches 1 to 4.

Each time the switch is pressed the corresponding output will be toggled.

Once you have entered the right pattern, press the reset switch.

The pattern will be stored, if you require the pattern to stay the same, press the reset button again to store the same pattern. Alternatively use the switches 1 to 4 to set the next pattern in the sequence. Continue entering patterns and pressing reset until the complete sequence is stored.

You can enter up to 250 steps in your program.

Run mode

During run mode the red run LED will be illuminated.

The stored pattern will run in sequence.

To restart the sequence, press the reset switch.

The sequence is stored in EEPROM, this means that you can remove the power and the sequence is remembered.

Editing the program

To edit the program, return to program mode.

The first pattern will be shown, use the reset switch to step through the sequence.

When you have got to the line that needs editing, change the pattern, then press reset to store.

To add to the end of the program keep pressing reset until the end of the sequence is reached.

Setting the run speed

To set the run speed:

Press and hold SW1.

Press and release the reset switch.

The program LED will quickly flash to indicate the run speed is being set.

When the required delay has passed release SW1.

You can set any time up to 25 seconds.

Erasing the program

To clear the program:

Press and hold SW1, SW2, SW3 & SW4 together, then press and release the reset switch. Both the red & green run / program LEDs will illuminate to indicate the program has been erased.

Please note:

The motor controller drives the outputs in pairs, and it is not possible to have the two output or LEDs on at the same time. If you find that when you press a button that a different LED changes, you will first need to set the corresponding output low.