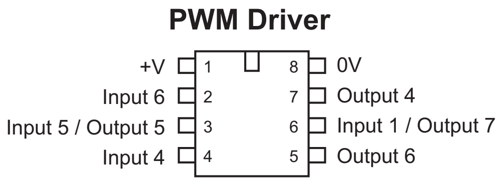
PWM Driver Firmware

Order Code

FRM050 PWM Driver Firmware Chip

Contents

1 x PWM Driver (pre-programmed PIC12F629)
1 x data sheet



Introduction

The PWM driver firmware was custom designed for the PICAXE microrobot system (part AXE120). It has now been made available as a separate item due to customer requests. The firmware operates at 3 to 5.5V DC.

The PWM driver chip sits 'between' the controlling PICAXE microcontroller and the L293D motor driver chip, and provides PWM control of both the L293D motor outputs. Due to the unique design the PWM driver chip uses the existing 4 wire connection to the L293D, no additional microcontroller output pins are required.

Operation

The 2 motors on a robot buggy can be controlled to make the robot turn, and the speed of the motor can also be adjusted. The motors are controlled in this example by PICAXE outputs 4 to 7.

The table shows how to control the direction of the robot.
To move forwards use the command **let pins = %10100000**
To move backwards use the command l**et pins = %01010000**

7	**6**	**5**	**4**	**Direction Control**
0	0	0	0	Stop
1	0	1	0	Forward
1	0	0	1	Turn Left
0	1	0	1	Reverse
0	1	1	0	Turn Right
				Speed Control
0	Pulse	1	1	Set Speed Left
1	1	0	Pulse	Set Speed Right

The speed of the robot is controlled by a technique called PWM (pulse width modulation), where the output is rapidly switched on and off at high frequency. By varying the 'on' time to 'off' time ratio, the speed of the micro-robot motors can be varied. The PWM pulsing function is provided continuously by the 8 pin PWM speed controller chip fitted to the control board.

The speed of each motor can be adjusted individually, although it is more common to keep both motors running at roughly the same speed. Speed is adjusted by first setting two pins high, and then using a 'pulsout' command, followed by a number between 50 (slow) and 255 (fast). Note that numbers less than 50 will probably cause the gearbox to stall. See sample program overleaf for an example.

Note that it is normal for DC motors to run at slightly different speeds due to manufacturing tolerances of each motor - this is NOT a fault, and may cause the robot to veer slightly to one side when moving in a straight line. It may be possible to set each motor at slightly different speeds to compensate this veer e.g. left motor at speed 50 and right motor at speed 51. Experiment to find the best values for your motors.

On power up the motors default to speed 128. Note that as the PWM speed controller chip takes a few microseconds to initialise after power-up, each micro-robot PICAXE program should always start with a 'pause 100' command to allow the chip to initialise. before speed data is transmitted.

See overleaf for an example typical circuit.

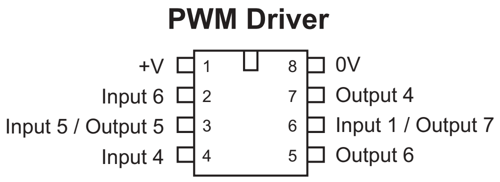
Sample program

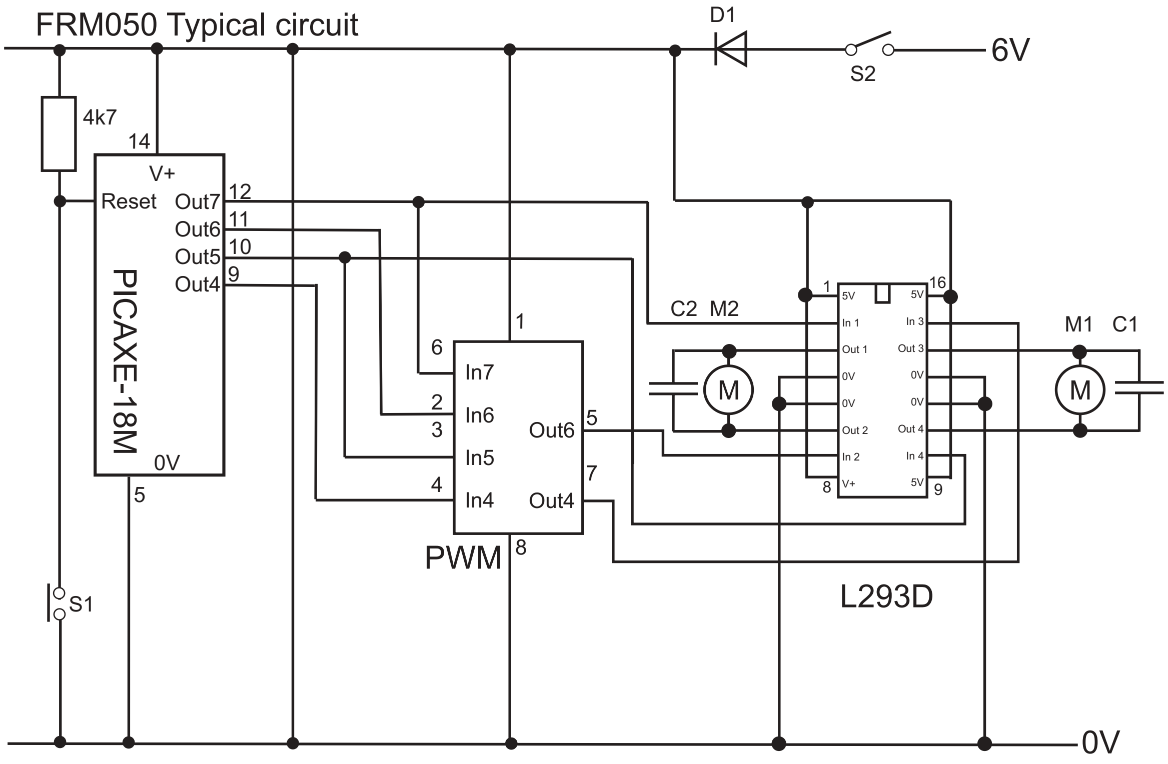
This sample program starts the micro-robot forwards at full speed for 3 seconds. It then reverses for 3 seconds at slow speed, turns and then starts going fowards again.

```
symbol speedR = b1
symbol speedL = b2

      pause 100               ' motor controller start-up pause
main:
      let speedR = 255        ' maximum speed
      let speedL = 255        ' maximum speed
      gosub set_speed         ' set the speed
      let pins = %10100000    ' buggy forward
      pause 3000              ' wait 3 seconds
      let pins = %00000000    ' stop
      let speedL = 60         ' set slow speed
      let speedR = 60         ' set slow speed
      gosub set_speed         ' set the speed
      let pins = %01010000    ' reverse
      pause 3000              ' wait 3 seconds
      let pins = %10010000    ' turn
      pause 2000              ' wait 2 seconds
      goto main               ' loop

set_speed:
      let pins = %00110000    ' set left speed
      pulsout 6,speedL        ' send a pulse of length in 'speedL'
      pause 10                ' short delay
      let pins = %11000000    ' set right speed
      pulsout 4,speedR        ' send a pulse of length in 'speedR'
      pause 10                ' short delay
      return
```


PWM Driver
+V 1
Input 6 2
Input 5 / Output 5 3
Input 4 4
8 0V
7 Output 4
6 Input 1 / Output 7
5 Output 6


FRM050 Typical circuit
D1
S2
6V
4k7
14
V+
Reset
Out7 12
Out6 11
Out5 10
Out4 9
PICAXE-18M
0V
5
S1
PWM
1
6 In7
2 In6
3 In5
4 In4
Out6 5
Out4 7
8
C2 M2
M
M1 C1
L293D
0V