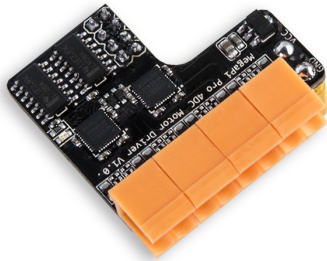


MegaPi Pro Quad DC Motor Driver



overview

This module is used for MegaPi Pro, which can expand 4 DC motor interfaces, and has a convenient plug-in structure. Can drive 4 25 DC motors/37 DC motors/solenoid valves. The purpose of the DC motor plus the drive circuit is to provide a large enough current. The H-bridge drive circuit is a common circuit designed to control the DC motor. It mainly realizes the forward and reverse drive of the DC motor. Its shape is similar to the letter H, 4 The position of a switch (MOSFET) is called a bridge arm, and the DC motor as a load is mounted on it like a bridge, so it is called an H-bridge drive. With the different opening and closing states of 4 switches, 4 working states of the motor can be generated: forward rotation, reverse rotation, braking and coasting.

technical specifications

- Motor drive: MP80495
- Motor Channels: 4
- Minimum working voltage: 6V
- Maximum working voltage: 12V
- Logic voltage: 5V
- Rated working current of each channel: 3A
- Peak operating current: 5.5A
- Module size: 39mmx39mm (length x width)

Features

- Supports motors with a working voltage of 6~12V
- When powered by 12V power supply, the working current can reach 3A (peak current can reach 5.5A)
- The module has over-voltage protection, over-current protection, and over-temperature protection to ensure the safety of use in all aspects
- Can drive 4 DC motors
- Encoded motor drive end has echo electromotive force protection (TVS protection)
- The motor direction of this module is opposite to that of the encoding/DC motor drive module, which is convenient for users to use in a variety of ways

programming guide

• Arduino programming

If you use Arduino programming, you need to call the library `Makeblock-Library-master` to control the four-way motor driver module.

function	Function
MeDCMotor (uint8_port)	Select the interface (dcfourmotor_1(1)~dcfourmotor_4(4))
run(int16_t speed)	Set speed (-255~+255)
stop()	stop motor

programming example

After the following program is run, the DC motor of interface 1 of the four-way circuit module rotates at full speed for 1 second, stops for 1 second, and moves circularly.

```
#include <Arduino.h>
#include <Wire.h>
#include <SoftwareSerial.h>


#include <MeMegaPiPro.h>
MeDCMotor dcfourmotor_1(1);

void loop(){
    dcfourmotor_1.run(255);
    delay(1);
    dcfourmotor_1.run(0);
    delay(1);
    loop();
}
```

[Download sample program](#)

• mBlock programming

Block description

building blocks	illustrate
	Select interface (M9~M10); set speed (-255~+255).

After the following program runs, the motor connected to the four-way DC motor drive interface 1 rotates at full speed for 1 second, stops for 1 second, and repeats.



[Download sample program](#)

• mBlock programming

Block description

building blocks	illustrate
直流电机 M9 ▾ 以动力 50 %转动	Select interface (M9~M12); set parameters (-100~+100).
电磁阀 M9 ▾ 设置状态为 断电 ▾	Select interface (M9~M12); set parameters (power on/off).

After the following program runs, the DC motor M9 runs at 50% power, and the solenoid valve M10 is energized. After 1 second, the DC motor M9 stops, the solenoid valve M10 is powered off, and the process is repeated.



[Download sample program](#)

• Python3 programming

1. The main control board MegaPi Pro is connected to the Raspberry Pi.
2. Install the latest makeblock library on Raspberry Pi `pip3 install makeblock --upgrade`.
3. Create a new Python file with the suffix .py.
4. Write the program in the Python file.
5. Run the Python file, such as "python123.py".

statement

function	Function
DC Motor (port)	Create a DC motor object. port: MegaPiPro.M9~MegaPiPro.M12
run(speed)	Rotate at percentage speed. speed: speed percentage, range: -100~+100

After the following program is run, the M9 DC motor will run at 50% power for 2 seconds, stop for 1 second, run at -50% power for 2 seconds, stop for 1 second, and cycle.

```
from time import sleep
from makeblock import MegaPiPro
board = MegaPiPro.create()
motor = board.DCMotor(MegaPiPro.M9)
while True:
    motor.run(50)
    sleep(2)
    motor.run(0)
    sleep(1)
    motor.run(-50)
    sleep(2)
    motor.run(0)
    sleep(1)
```

[Download sample program](#)

connection method

