

Microduino mCookie-Motor

USER GUIDE

Content

Features	2
Specification.....	2
Development	3
DC Motor Wire Connection	3
Project.....	3
Connection and disconnection control	3
PWM rate control	4

mCookie-Motor is a DC motor controller with PWM (Pulse Width Modulation) for various speed control. The speed of the motor can be controlled through PWM and direction (forward or backwards). This module needs to be powered with a sufficient power supply such as:

- BM shield with a BM module
- mBattery.

Features

- Current & speed limited by internal PWM;
- Synchronous double-channel can reduce power consumption;
- DC motor braking mode;
- Capable of controlling two DC motors.
- Speed and direction control.
- Uses standard connectors: 2 pin, 1.25mm pitch JST connectors.

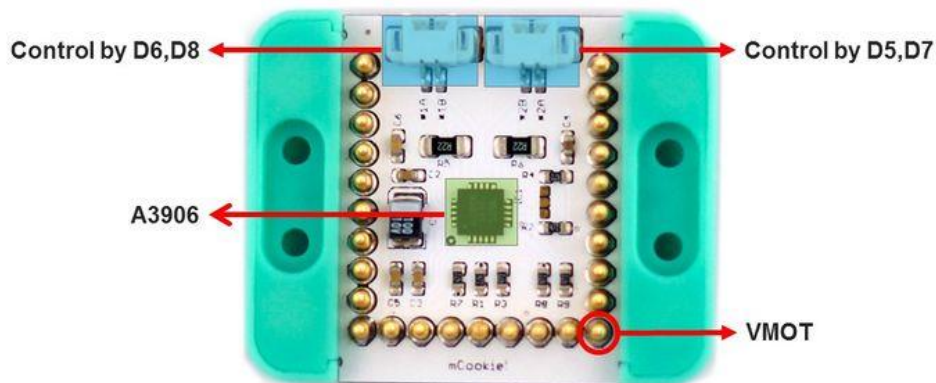
Specification

mCookie pin	Motor chip pin	Function
D6	1A	Control 1A high level and rotate clockwise
D8	1B	Control 1B high level and rotate counter clockwise
D5	2A	Control 2A high level and rotate clockwise
D7	2B	Control 2A high level and rotate counter clockwise
VMOT	BM module	Power supply
GND	GND	Ground

Note:

- The pins including 5, 6, 7 and 8 cannot be occupied by other sensors while using the Motor or it may cause problems.
- Cannot be power without sufficient power supply such as through only Core USB. Module Motor must be powered on by:
 - BM shield with a BM module
 - mBattery.

mCookie-Motor



microduino | a product of makermodule | confidential | www.microduino.cc

Development

DC Motor Wire Connection

- Connect one DC motor to (OUT1A, OUT1B) and the other to (OUT2A, OUT2B);
- DC motor pin control:

```
1. //(D6, D8) controls motor (1A, 1B)
2. #define OUT1A 6
3. #define OUT1B 8
4. //(D5, D7) controls motor (2A, 2B)
5. #define OUT2A 5
6. #define OUT2B 7
```

Project

Connection and disconnection control

6	8	5	7	1A	1B	2A	2B	Function
Low	Low	Low	Low	Off	Off	Off	Off	Close (Stop habitually)
High	Low	High	Low	High	Low	High	Low	Rotate clockwise
Low	High	Low	High	Low	High	Low	High	Rotate counter clockwise
High	High	High	High	Low	Low	Low	Low	Halt

```
1. #define OUT1A 6
2. #define OUT1B 8
3. #define OUT2A 5
4. #define OUT2B 7
5.
6. void setup()
7. {
8.   pinMode(OUT1A, OUTPUT);
9.   pinMode(OUT1B, OUTPUT);
10.  pinMode(OUT2A, OUTPUT);
11.  pinMode(OUT2B, OUTPUT);
12. }
13.
14. void loop()
15. {
16.  head();
17.  delay(2000);
18.  back();
19.  delay(1000);
20.  stop();
21.  delay(500);
22. }
23.
24. void head()
25. {
26.   digitalWrite(OUT1A, HIGH);
27.   digitalWrite(OUT1B, LOW);
28.   digitalWrite(OUT2A, HIGH);
29.   digitalWrite(OUT2B, LOW);
30. }
31. void back()
32. {
33.   digitalWrite(OUT1A, LOW);
34.   digitalWrite(OUT1B, HIGH);
35.   digitalWrite(OUT2A, LOW);
36.   digitalWrite(OUT2B, HIGH);
37. }
38. void stop()
39. {
40.   digitalWrite(OUT1A, LOW);
41.   digitalWrite(OUT1B, LOW);
42.   digitalWrite(OUT2A, LOW);
43.   digitalWrite(OUT2B, LOW);
44. }
```

PWM rate control

```
1. //(D6, D8) controls motor (1A, 1B)
2. #define OUT1A 6
3. #define OUT1B 8
4. //(D5, D7) controls motor (2A, 2B)
5. #define OUT2A 5
6. #define OUT2B 7
7.
8. void setup()
9. {
10.  pinMode(OUT1A, OUTPUT);
11.  pinMode(OUT1B, OUTPUT);
12.  pinMode(OUT2A, OUTPUT);
13.  pinMode(OUT2B, OUTPUT);
14. }
15.
16. void loop()
```

```
17. {
18.   for (int fadeValue = 0; fadeValue <= 255; fadeValue += 5)
19.     //Loop statement. Along with PWM rate increases, you can change brightne
    ss level by controlling fadeValue.
20.   {
21.     analogWrite(OUT1A, fadeValue); //Write rate level into the motor
22.     digitalWrite(OUT1B, LOW);
23.     analogWrite(OUT2A, fadeValue);
24.     digitalWrite(OUT2B, LOW);
25.     delay(100); //Delay time of rate. (The unit is ms)
26.   }
27.   for (int fadeValue = 255; fadeValue >= 0; fadeValue -= 5)
28.     // Loop statement. Along with PWM rate decreases, you can change brightn
    ess level by controlling fadeValue.
29.   {
30.     digitalWrite(OUT1A, LOW);
31.     analogWrite(OUT1B, fadeValue); // Write rate level into the motor
32.     digitalWrite(OUT2A, LOW);
33.     analogWrite(OUT2B, fadeValue);
34.     delay(100); // Delay time of rate. (The unit is ms)
35.   }
36.   delay(1000);
37. }
```