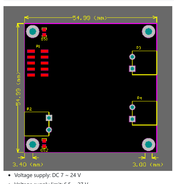
# 7A Dual DC Motor Driver SKU DRI0041

## Introduction

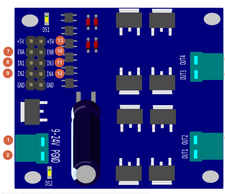
This is an ultra-small dual DC motor driver for space limited projects. It features a UVLO (Under Voltage Lock Out) circuit, ESD protection, and opto-isolated inputs. The opto-isolated inputs prevent motor power from damaging or interfering with your control circuit.

## Specification



- Voltage supply: DC 7 – 24 V
- Voltage supply limit: 6.5 – 27 V
- Control signal Level (Compatible 3.3V/5V)
  - High: DC 3.0 – 6.5 V
  - Low: DC0 – 0.8 V
- Output Channels: 2
- Control signal current: 3 – 11 mA (per input)
- Maximum continuous operating current: 7 A
- Peak current: 50 A
- Speed control: PWM
  - Minimum valid Pulse Width: 5 us
- Working Temperature: -25 – 85 °C
- Mounting Hole: M3
- Dimension (Length * Width * Height): 55 x 55(mm)/2.165 x 2.165(in)
- Weight: 32g

## Board Overview



| Num | Label | Description |
|---|---|---|
| 1 | 9 - 24V | Power Supply, + |
| 2 | PGND | Power Supply, GND |
| 3 | OUT1 | Motor1_+ |
| 4 | OUT2 | Motor1_- |
| 5 | OUT3 | Motor2_+ |
| 6 | OUT4 | Motor2_- |
| 7 | ENA | Motor1 PWM |
| 8 | IN1 | Motor1 control signal |
| 9 | IN2 | Motor1 control signal |
| 10 | ENB | Motor2 PWM |
| 11 | IN3 | Motor2 control signal |
| 12 | IN4 | Motor2 control signal |
| 13 | +5V | Voltage Reference Input, +5V OR 3.3V |

**Control Method**

| IN1 | IN2 | ENA/ ENB | Motor1/2 Behavior |
|---|---|---|---|
| 0 | 0 | x | Stop (brake) |
| 1 | 1 | x | Vacant |
| 1 | 0 | 1 | Forward 100% |
| 0 | 1 | 1 | Reverse 100% |
| 1 | 0 | PWM | Forward at PWM speed |
| 0 | 1 | PWM | Reverse at PWM speed |

**In this table**

- IN1 & IN2: the control signal input to change the motor behavior
- "0": TTL_Low
- "1": TTL_High
- "x": Any TTL and it is default TTL_Low while no PWM signal.
- "ENA/ ENB": PWM speed setting

**Note:**

**IN1 & IN2**

To protect your motor, before switching the motor drive direction, first BRAKE the motor by setting **IN1=0 & IN2=0**. This is especially important when the PWM is set to 100% (full speed). The suggested braking time is >0.1sec, depending on your motor.

**+5V**

This signal is a reference that must be set to the same power supply that your microcontroller operates on. Connect this to the 3.3v or 5v power supply used by the controller.

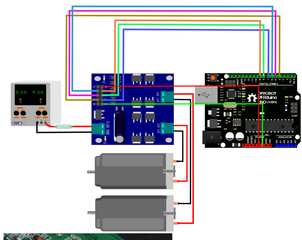
## Tutorial

This tutorial will cover how to use PWM to control a motor using the shield. Do the wiring according to the Connection Diagram below, and then upload the sample code below to the Arduino board. Here we use a arduino UNO as the controller, you should see your motor run forward for 3 seconds and then run reverse for another 3 seconds and then repeat.

### Requirements

- Hardware
  - 1 x DFRduino UNO (or similar)
  - 1 x This 7A Dual DC Motor Driver
  - 2 x DC Motor
  - 1 x Fuse@20A
  - Jumper wires
- Software
  - Arduino IDE Click to Download Arduino IDE from Arduino®.

### Connection Diagram



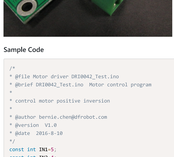
**Facility Safety and the Personal Safety:**
Please add a fuse@20A between the Power source and this shield.

**Installation Tips**

The back of the driver board has some large bare traces. It is important to make sure that these do not get shorted out against conductive surfaces of your project. Please measure carefully and if needed, apply 1mm of non-conductive epoxy to protect the board.

**Pluggable Connector**

The connectors are designed to be pluggable. This allows you to attach wires with male or female terminations.



### Sample Code

```
/*!
* @file Motor_driver_DRI0042_Test.ino
* @brief DRI0042_Test.ino  Motor control program
*
* control motor positive inversion
*
* @author bernie.chen@dfrobot.com
* @version  V1.0
* @date  2016-8-10
*/
const int IN1=5;
const int IN2=4;
const int ENA=6;

const int IN3=8;
const int IN4=7;
const int ENB=9;

void setup() {
    pinMode(IN1, OUTPUT);
    pinMode(IN2, OUTPUT);
    pinMode(ENA, OUTPUT);

    pinMode(IN4, OUTPUT);
    pinMode(IN3, OUTPUT);
    pinMode(ENB, OUTPUT);
}

void loop() {
  Motor1_Brake();
  Motor2_Brake();
  delay(100);
  Motor1_Forward(200);
  Motor2_Forward(200);
  delay(1000);
  Motor1_Brake();
  Motor2_Brake();
  delay(100);
  Motor1_Backward(200);
  Motor2_Backward(200);
  delay(1000);
}

void Motor1_Forward(int Speed)
{
    digitalWrite(IN1,HIGH);
    digitalWrite(IN2,LOW);
    analogWrite(ENA,Speed);
}

void Motor1_Backward(int Speed)
{
    digitalWrite(IN1,LOW);
    digitalWrite(IN2,HIGH);
    analogWrite(ENA,Speed);
}
void Motor1_Brake()
{
    digitalWrite(IN1,LOW);
    digitalWrite(IN2,LOW);
}
void Motor2_Forward(int Speed)
{
    digitalWrite(IN3,HIGH);
    digitalWrite(IN4,LOW);
    analogWrite(ENB,Speed);
}

void Motor2_Backward(int Speed)
{
    digitalWrite(IN3,LOW);
    digitalWrite(IN4,HIGH);
    analogWrite(ENB,Speed);
}
void Motor2_Brake()
{
    digitalWrite(IN3,LOW);
    digitalWrite(IN4,LOW);
}
```

## FAQ

For any questions, advice or cool ideas to share, please visit the DFRobot Forum.

## More Documents

Back to Top