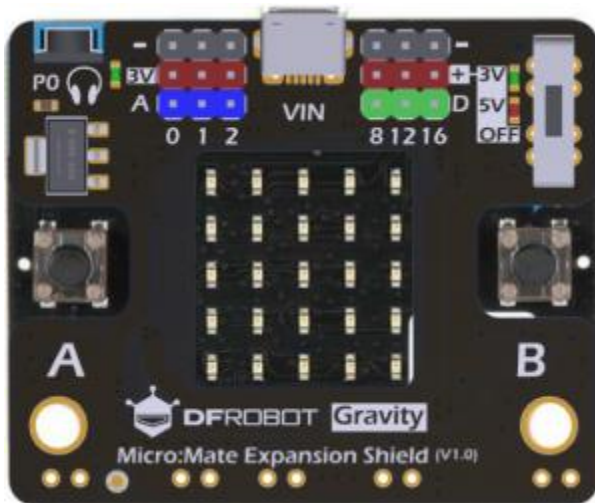


Micro:Mate- A Mini Expansion Board for micro:bit SKU: DFR0518

Introduction



Micro: Mate is a tiny micro: bit I/O expansion board for learning electronics and building DIY projects.

Micro:Mate expands 6 sets of 3-pin I/O interfaces, capable of connecting DFRobot Gravity series modules, servo motors, sensors and jumper wires. Additionally, Pin 8, 12, 16 support voltage switch between 3V-5V, allowing up to 5V 2A digital (PWM) output.

Micro:Mate is in the same dimension of micro: bit. It connects to micro: bit through contact pins (with spring loaded), ensuring easy, compact and secure connection. The rubber bumpers and the 3.5mm audio jack on the back side keep the expansion installed stably on the board, meanwhile prevents reversed connection.



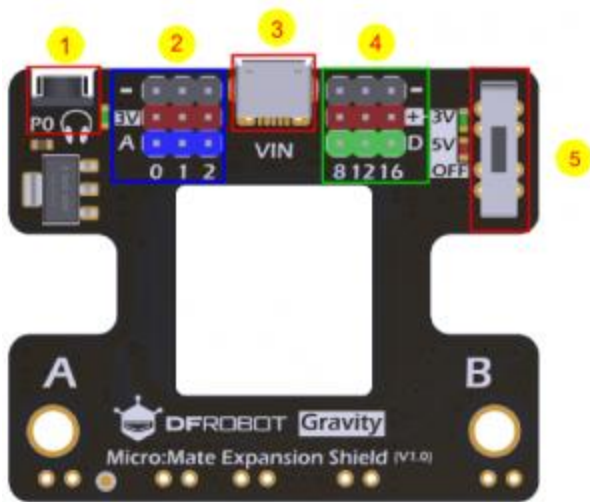
NOTE:

The mounting screws should be well tightened to ensure a secured connection.
Micro: Mate only supports 3V (3.3V) analog input from Pin 0, 1, 2. The Micro USB Power port on Micro: Mate cannot be used for data transmission.
Components with large power consumption should be connected to Pin 8, 12, 16, with 5V power supply.

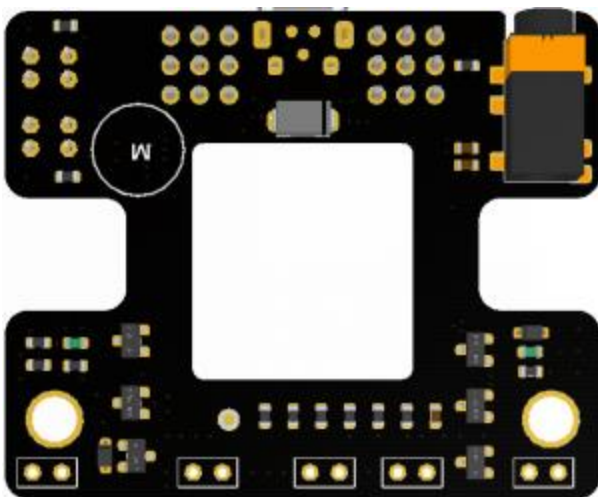
Specification

- Interface: 6 x Gravity 3Pin
- Operating Voltage: 5V/3.3V
- Input Power: <10W
- Measuring Range: 0-3.3V (Analog), 0-5V (Digital):
- Dimension: 68.5 * 53.3mm
- Weight: 22.8g

Board Overview



Front side



Back side

Micro:Mate Layout

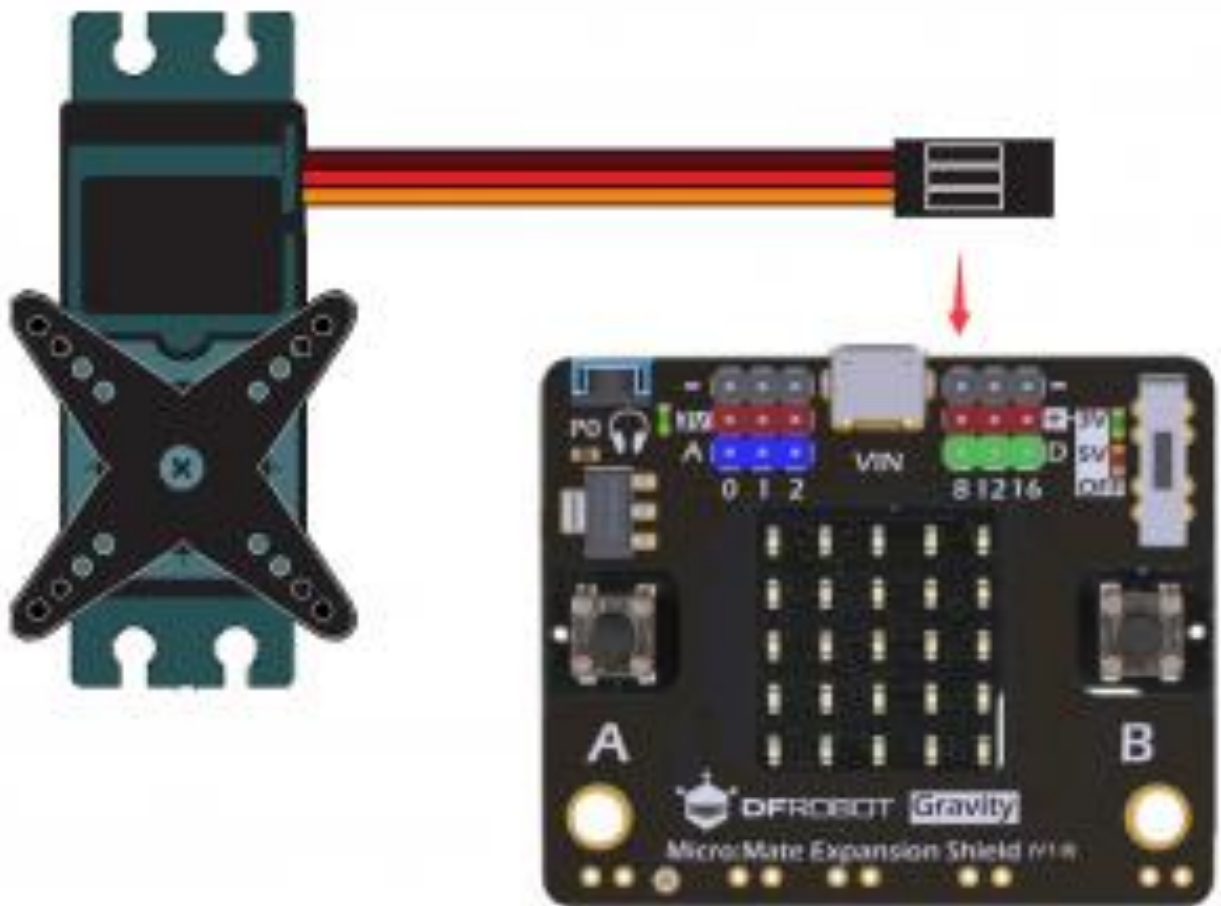
Number	Name	Description
1	Audio Jack	3.5mm Audio Jack
2	I/O Pin	Gravity series compatible (3V only)
3	External USB power port	External power supply, does not support program uploading
4	I/O Pin	Gravity series compatible (3V and 5V)
5	Voltage switch	Switch between 3V, 5V(Pin 8,12,16 only) and OFF

Servo Control Tutorial

Requirements

- Hardware
 - [micro:bit](#) x 1
 - Servo x 1
 - MicroUSB power supply
- Software
 - MakeCode Block Editor/ JavaScript Editor / BXY Python Editor

Connection Diagram



Connection Diagram

Sample Code

[Microsoft MakeCode Editor](#)



MakeCode

[Microsoft MakeCode JavaScript Editor](#)

```
basic.forever(() => {  
  basic.pause(1000)  
  pins.servoWritePin(AnalogPin.P8, 0)  
  basic.pause(1000)  
  pins.servoWritePin(AnalogPin.P8, 90)  
  basic.pause(1000)  
  pins.servoWritePin(AnalogPin.P8, 180)  
  basic.pause(1000)  
  pins.servoWritePin(AnalogPin.P8, 90)  
})
```

[BXY micro:bit Python editor](#)

```
#http://docs.dfrobot.com.cn/bxy/examples/servo.html
from microbit import *
import Servo
sv=Servo(pin8)
while True:
    sv.angle(0)
    sleep(1000)
    sv.angle(90)
    sleep(1000)
    sv.angle(180)
    sleep(1000)
    sv.angle(90)
    sleep(1000)
```

Expected Results

The Servo turns back and forth from 0-180 in every 4 seconds

More Documents

- [Schematic & Layout](#)