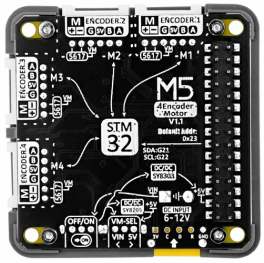
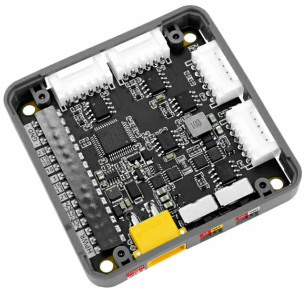
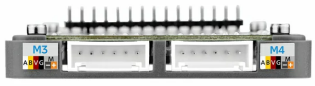
Module 4EncoderMotor v1.1

SKU:M138-V11







Description

Module 4EncoderMotor v1.1 is a 4-channel encoder motor driver module that uses the STM32 + BL5617 H-bridge driver IC solution. It supports I2C communication, allowing for slave address modification and providing flexible control methods. Through AB pulse encoder signal input, it achieves precise motor motion state and position detection. It supports duty cycle control, absolute position positioning, and speed adjustment modes, enabling various functions such as forward rotation, reverse rotation, stop, and braking of the motor. It integrates INA199 power monitoring to monitor current status in real-time. The onboard power input switch allows for DC 5V or external DC 6 ~ 12V power input. Compared to the previous 4EncoderMotor Module, this product has optimized the interface, adopting a unified HY2.0-6P interface. It is suitable for applications such as robot motion control, automation equipment, smart vehicles, laboratory equipment, and industrial automation systems.

Features

- 4-channel encoder motor driver
- AB pulse signal input
- Duty cycle, absolute position positioning, and speed adjustment control modes
- I2C communication
- Motor voltage input and current monitoring

Includes

- 1 x Module 4EncoderMotor v1.1
- 4 x HY2.0-6P single-ended cables (20cm)
- 4 x PH2.0-6P connectors
- 1 x DC5521 Female to XT30 Female cable

Applications

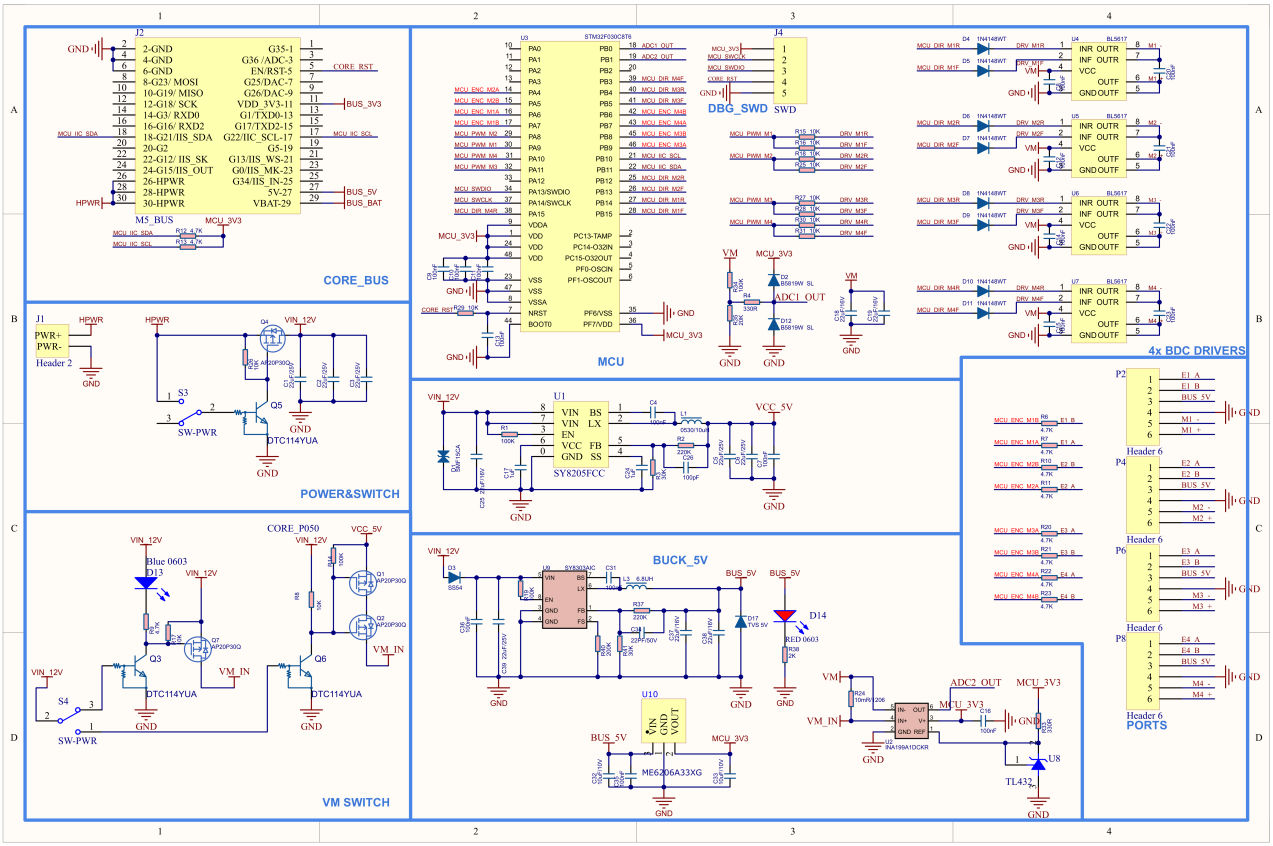
- Robot motion control
- Automation equipment
- Industrial automation systems

Specifications

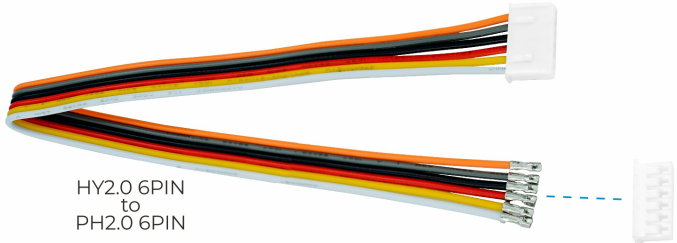
Specification	Parameter
MCU	STM32F030C8T6
Encoder Motor Driver IC	BL5617
Current Detection Chip	INA199
Maximum Supported Current	3.0A
Power	Maximum 10W
External DC Power	6-12V
I2C Communication Address	0x24
Standby Current	DC 6V@35.03mA DC 12V@19.25mA
PWM Drive Signal Frequency	1KHz
Operating Temperature	0-40°C
Product Size	54.0 x 54.0 x 13.1mm
Product Weight	16.1g
Package Size	95.0 x 66.0 x 26.0mm
Gross Weight	57.8g

Schematics

- Module 4EncoderMotor v1.1 Schematics PDF



Encoder Motor Wiring Example



Adjusting Wire Sequence for Different Encoder Motors

PinMap

I2C Communication and Current/Voltage Detection Pins

STM32	MCU_IIC_SDA(PB11)	MCU_IIC_SCL(PB10)	PB0	PB1
Core(Basic)	G21	G22		
Core2	G21	G22		
CoreS3	G12	G11		
Motor Voltage Detect			ADC1_OUT	
Current Detect				ADC2_OUT

Motor Direction Control Pins

STM32	PB14/PB15	PB12/PB13	PB4/PB5	PA15/PB3
BL5617	MCU_DIR_M1R/MCU_DI	MCU_DIR_M2R/MCU_DI	MCU_DIR_M3R/MCU_DI	MCU_DIR_M4R/MCU_DI
(Direction)	R_M1F	R_M2F	R_M3F	R_M4F

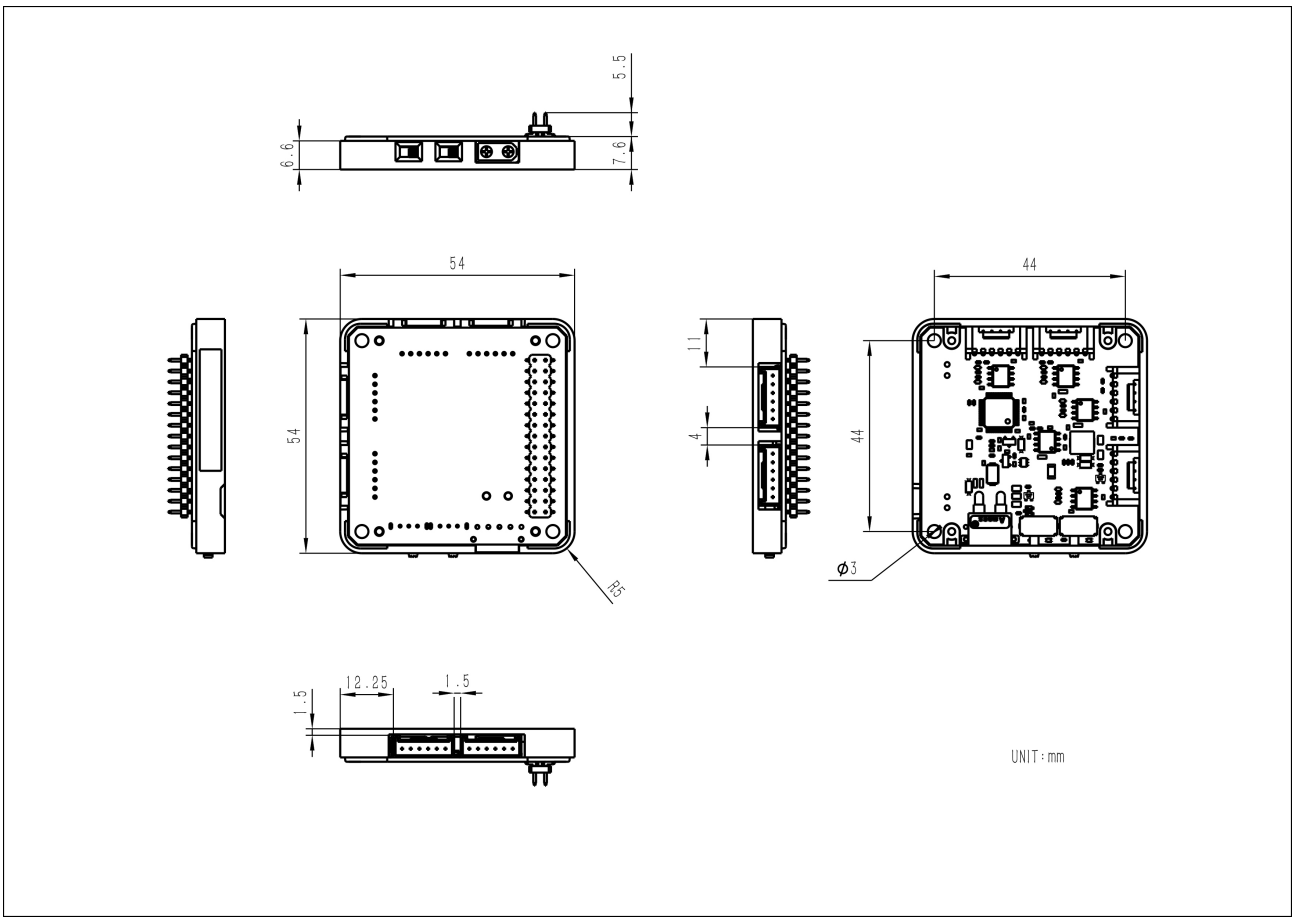
PWM Control Pins

STM32	PA9	PA8	PA11	PA10
BL5617 (PWM)	MCU_PWM_M1	MCU_PWM_M2	MCU_PWM_M3	MCU_PWM_M4

A/B Signal Detection Pins

STM32	PA6/PA7	PA4/PA5	PB9/PB8	PB7/PB6
Encodering motors	E1_A/E1_B	E2_A/E2_B	E3_A/E3_B	E4_A/E4_B

Model Size



Datasheets

- BL5617 Datasheet

Softwares

Arduino

- Module 4EncoderMotor v1.1 Arduino Library
- Module 4EncoderMotor v1.1 Example with Basic
- Module 4EncoderMotor v1.1 Example with Core2
- Module 4EncoderMotor v1.1 Example with CoreS3

Internal Firmware

- Module 4EncoderMotor v1.1 Internal Firmware

Firmware Version	Update	Protocol
v3	First Release Version	Module 4EncoderMotor v1.1 I2C Protocol v3
v4	1. Adjust the soft start function and allow setting individual rotation directions for each motor.	Protocol same with v3

M5 DAPLink

If you don't have the STM32 downloader tool, you can refer to the M5 DAPLink tutorial and use the Core2 or CoreS3 as a programmer to complete the firmware update for the device.

Protocol

- Module 4EncoderMotor v1.1 I2C Protocol

M5Stack Module 4EncodeMotor V1.1 I2C Protocol																		V3 (FW Version) 2024/3/1
REG MAP (Addr:0x24)		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	note
Motor PWM Duty INT8	0x20 R/W	Motor1_PWM_Duty	Motor2_PWM_Duty	Motor3_PWM_Duty	Motor4_PWM_Duty													-127 ~ 127
Motor Encoder INT32	0x30 R/W	Motor1_Encoder-byte0	Motor1_Encoder-byte1	Motor1_Encoder-byte2	Motor1_Encoder--byte3	Motor2_Encoder-byte0	Motor2_Encoder-byte1	Motor2_Encoder-byte2	Motor2_Encoder--byte3	Motor3_Encoder-byte0	Motor3_Encoder-byte1	Motor3_Encoder-byte2	Motor3_Encoder--byte3	Motor4_Encoder-byte0	Motor4_Encoder-byte1	Motor4_Encoder-byte2	Motor4_Encoder--byte3	Motor Encoder = Motor Encoder-byte0 + Motor Encoder-byte1 * 256 + Motor Encoder-byte2 * 65536 + Motor Encoder-byte3 * 16777216
Motor Speed INT8	0x40 R	Motor1_Speed	Motor2_Speed	Motor3_Speed	Motor3_Speed													-127 ~ 127 Motor encoder increments every 20 ms
Motor1 Mode[1]	0x50 R/W	Motor1_Mode	Motor1_Position_P	Motor1_Position_I	Motor1_Position_D	Motor1_Position_Point-Byte0	Motor1_Position_Point-Byte1	Motor1_Position_Point-Byte2	Motor1_Position_Point-Byte3	Motor1_Position-MAX-Speed	Motor1_Speed_P	Motor1_Speed_I	Motor1_Speed_D	Motor1_Speed_Point				
Motor2 Mode[1]	0x60 R/W	Motor2_Mode	Motor2_Position_P	Motor2_Position_I	Motor2_Position_D	Motor2_Position_Point-Byte0	Motor2_Position_Point-Byte1	Motor2_Position_Point-Byte2	Motor2_Position_Point-Byte3	Motor2_Position-MAX-Speed	Motor2_Speed_P	Motor2_Speed_I	Motor2_Speed_D	Motor2_Speed_Point				
Motor3 Mode[1]	0x70 R/W	Motor3_Mode	Motor3_Position_P	Motor3_Position_I	Motor3_Position_D	Motor3_Position_Point-Byte0	Motor3_Position_Point-Byte1	Motor3_Position_Point-Byte2	Motor3_Position_Point-Byte3	Motor3_Position-MAX-Speed	Motor3_Speed_P	Motor3_Speed_I	Motor3_Speed_D	Motor3_Speed_Point				
Motor4 Mode[1]	0x80 R/W	Motor4_Mode	Motor4_Position_P	Motor4_Position_I	Motor4_Position_D	Motor4_Position_Point-Byte0	Motor4_Position_Point-Byte1	Motor4_Position_Point-Byte2	Motor4_Position_Point-Byte3	Motor4_Position-MAX-Speed	Motor4_Speed_P	Motor4_Speed_I	Motor4_Speed_D	Motor4_Speed_Point				
VIN Current Float (A)	0x90 R	current-byte0	current-byte1	current-byte2	current-byte3													float
VIN Current X100 Int (A)	0xC0 R	VIN Current X100-byte0	VIN Current X100-byte1	VIN Current X100-byte2	VIN Current X100-byte3													VIN Current X100 Int = VIN Current X100-byte0 + VIN Current X100-byte1 * 256 + VIN Current X100-byte2 * 65536 + VIN Current X100-byte3 * 16777216
VIN ADC 8bits[2]	0xA0 R	ADC Value 8bits																Vaule: 0~255
VIN ADC 12bits[3]	0xB0 R	ADC Value 12bits-L	ADC Value 12bits-H															Vaule: 0~4095
Encoder AB or BA	0xD0 R/W	Encoder AB or BA																Vaule: 0~1 0: AB(Default) 1: BA * Need to restart module to affect * Writing to this register will save the value to flash. Please do not write to this register frequently to prevent flash damage.
Soft start and stop	0xD0 R/W		Soft start and stop															Soft start and stop(0:disable, 1:enable): bit0: Motor1 bit1: Motor2 bit2: Motor3 bit3: Motor4
Firmware Version	0xF0 R															Version		Version: firmware version number
I2C Address	0xF0 R/W																Address	Address: 1~127 Writing to this register will save the value to flash. Please do not write to this register frequently to prevent flash damage.

[1] (1)Mode:
0:Normal(Open loop)
1: Position Lock
2: Speed Lock
(2)P/I/D: 0~255
(3)Motor_Position_Point = Motor_Position_Point-byte0 + Motor_Position_Point-byte1 * 256 + Motor_Position_Point-byte * 65536 + Motor_Position_Point-byte3 * 16777216
(4)Motor_Position-MAX-Speed: -127 ~ 127
(5)Motor_Speed_Point: -127 ~ 127

[2] Voltage = ADC Value 8bits/255*3.3/0.16

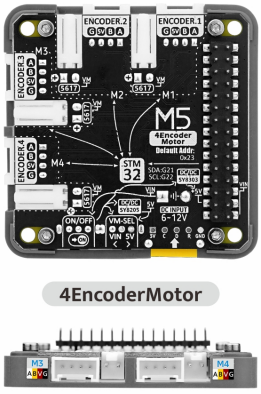
[3] Voltage = ((ADC Value 12bits-L) + (ADC Value 12bits-H)*256)/255*3.3/0.16

Video

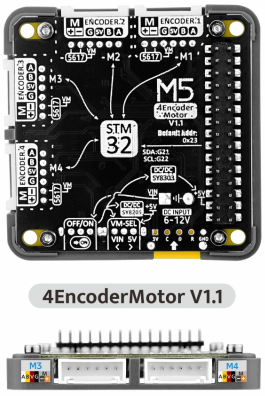
M138-V11 4EncoderMotor V1.1 视频.mp4

Product Comparison

Product	Communication Protocol	Chip Solution	Motor Type Supported	Channels	Control Modes	Notes
4EncoderMotor Module(M138)	I2C(0x24)	STM32+BL5617	DC Motor/Encoder Motor	4	Duty cycle, absolute position, speed control	
4EncoderMotor Module(M138-V11)	I2C(0x24)	STM32+BL5617	DC Motor/Encoder Motor	4	Duty cycle, absolute position, speed control	M138-V11 modified encoder motor interface to HY2.0-6P Grove port
DC Motor Module (M021)	I2C(0x56)	MEGA328+L293D	DC Motor/Encoder Motor	4	Speed control mode	



4EncoderMotor



4EncoderMotor V1.1

Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

M5Stack:

M138-V11