H44 Series Hall Effect Joystick

The H44 series Hall effect joysticks are available in 1-axis, 2-axis, or 3-axis configurations, panel mounting and embedded designs, made of stainless steel and alloy materials, featuring a spring-loaded self-return mechanism, a high-precision German Hall effect sensor, linear correction across the entire temperature range, IP65 or higher protection rating, smooth operation, and an ergonomic mechanical design.
Suitable for robots, drones, medical equipment, aerospace, marine, and broadcasting equipment.

MATE TECHNOLOGY SHENZHEN LIMITED
info@matend.com | www.matend.com | www.matend.cn

I. Product Features:

- Materials: Stainless Steel + Alloy + Engineering Plastic
- Hall Sensor
- Automatic Spring Return
- XY Axis: ±20 degrees; Z Axis: ±18 degrees
- Operation limiter: Straight, all-around (circular), Cross, Square;
- Output: Analog Voltage, RS485, RS422, RS232, USB, CAN
- Power supply: DC5V, DC12-28V;
- Less than 9-25mA (5V power supply, related to signal output and number of axes)
- More than 10 million cycles;
- -40 degrees to +70 degrees
- IP65 (panel and above)
- 44.4X44.
- Installation dimensions: 35x35

Controller Selection

MKF-JS08
No button
Axes: 1, 2, 3

MKF-JS03
1 button Axes: 1, 2, 3

MKF-JS20
No button
Axes: 1, 2, 3
Handle is retractable,
Shorten length 10mm
For portable devices

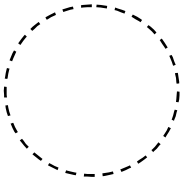
MKF-JS04
D1=1 button
D0=No button
Axes: 1, 2, 3

MKF-JS10
No button
Axes: 2, 3
For portable devices

MKF-JS47
1 button
Axes: 1, 2, 3

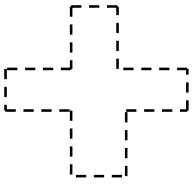
MKF-JS48
Axes: 2, 3

G=Circular limit



Q=Square limit

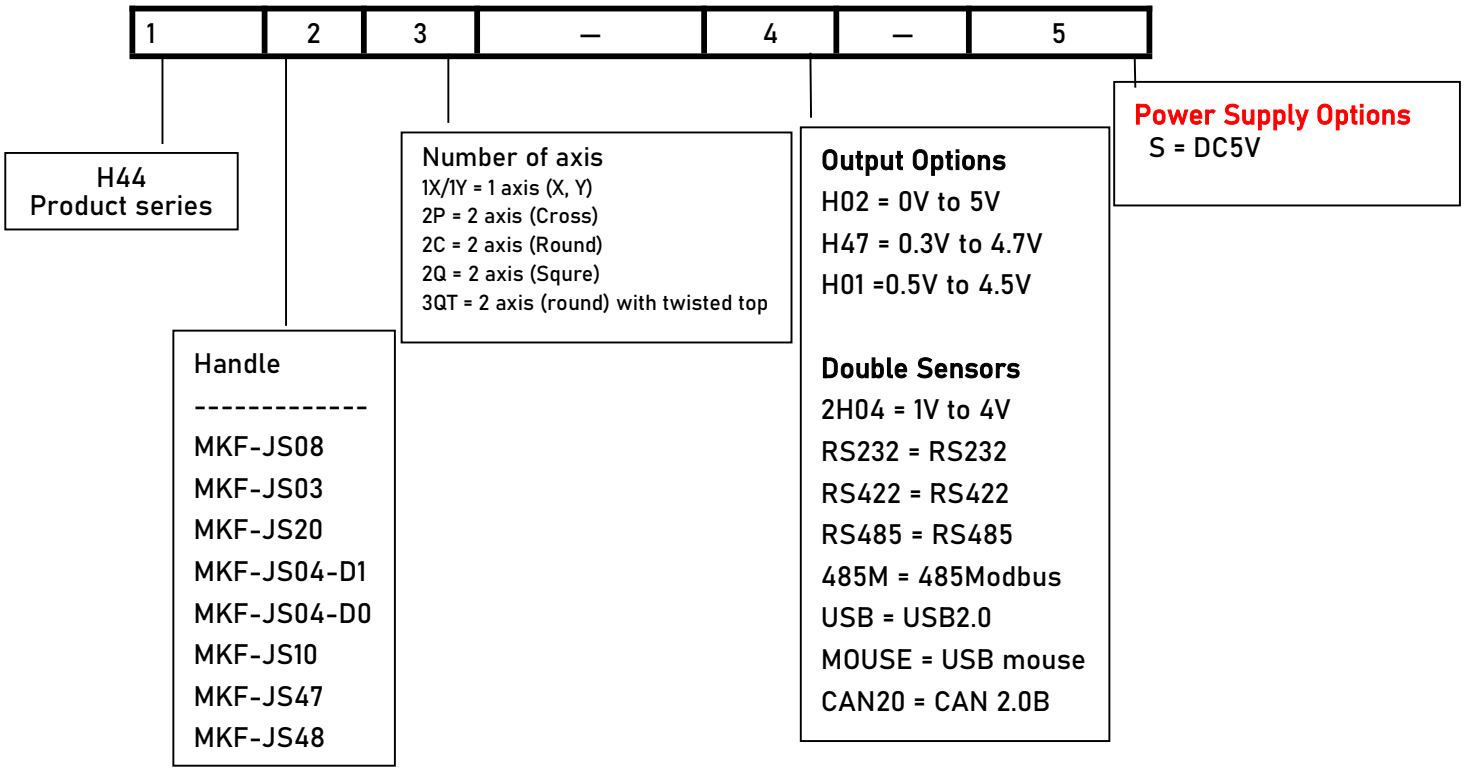
C=Cross limit switch



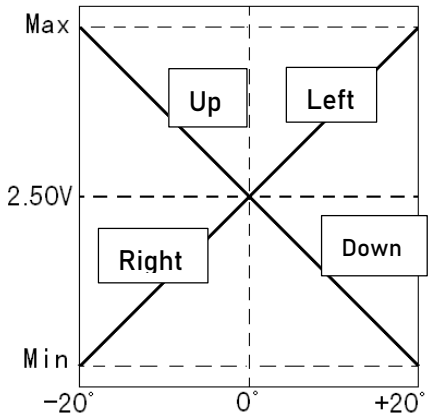
S=One-word limit

MATE TECHNOLOGY SHENZHEN LIMITED
info@matend.com | www.matend.com | www.matend.cn

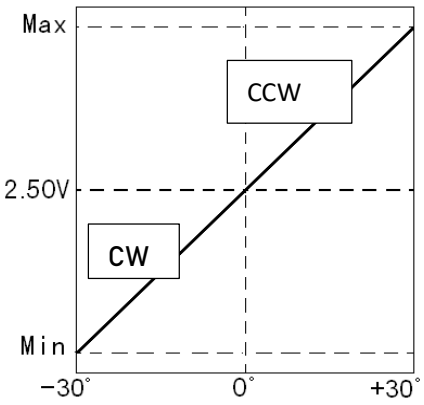
Product model and parameters (Overview)



Signal output of analog voltage:



X Y Axis



Z axis

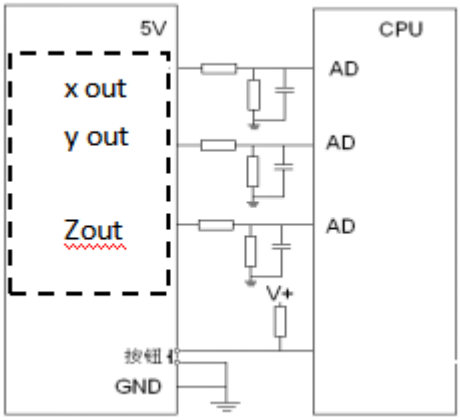
Push button switch schematic diagram:

Push button Switch Technical Parameters:

Switch Type: Normally Open
Contact Current: B Handle 1A/24V, D Handle 0.5A/24V
Operating Life: Greater than 1 million cycles

Electrical Parameters:
Minimum Operating Voltage: 3.1V (5V power supply), 8V (12-24V power supply)
Maximum Input Voltage: 5.5V (5V power supply), 28V (12-24V power supply)
Operating Current: Less than 25mA (5V power supply, analog voltage signal output)
Analog Voltage Signal Output Load: Greater than 1KΩ
Analog Voltage Signal Output Center Voltage: 2.50V or 50%Vdd
Analog Voltage Output Signal: 0V~5V / 0.3V ~4.7V / 0.5V ~4.5V / 1V~ 4V

WIRING DIAGRAM

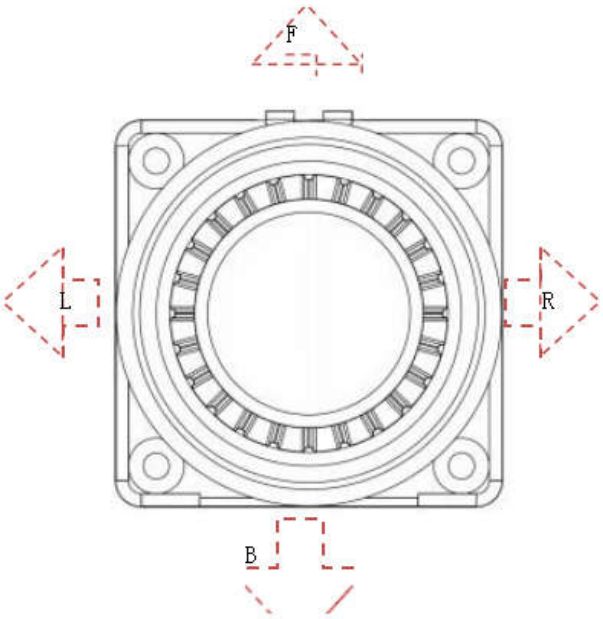


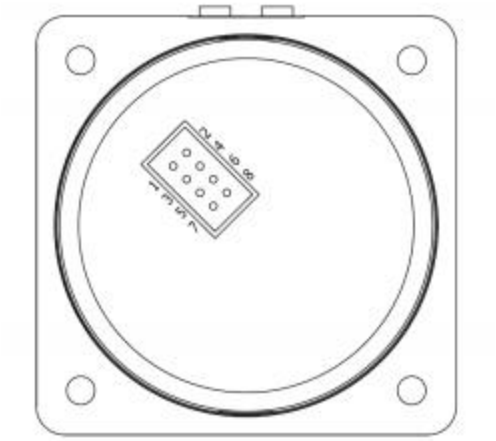
If CPU voltage is 3.3V, then the divider resistance is 1.5KΩ and 3KΩ Capacitance =1NF

If CPU voltage is 5V, cancel the divider resistance and connect to CPU pin directly.
The capacitance is not essential;

Connect resistance 10KΩ to the upper button

Joystick terminal diagram





Lead wire:
Lead wire length: 20cm (including terminals)
Terminal type: PHD2.0-8P

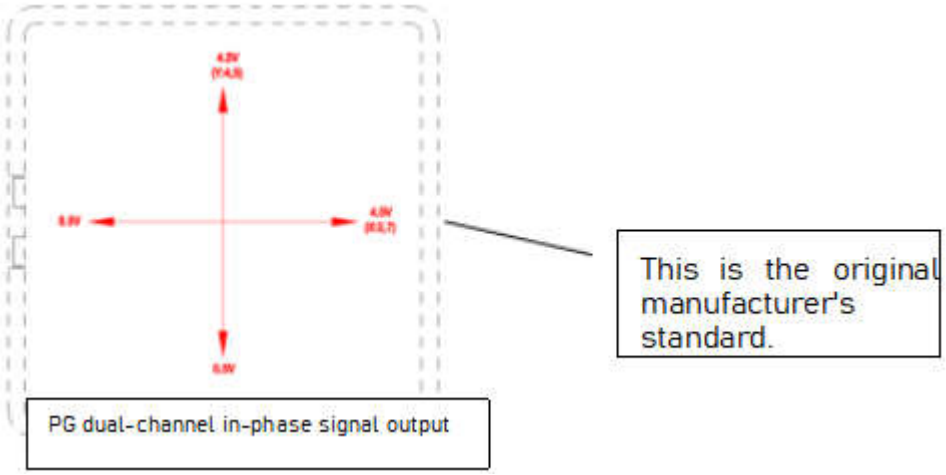
Connector pin definition - (3-axis) Analog voltage signal output:

Pin	Symbol	Color	Function Description
1	V+	red	Power+
2	Y	green	Y-axis signal output
3	GND	black	GND power supply
4	X	yellow	X-axis signal output
5	B	white	Button (The button has 2 pins, pin 1)
6	Cen	brown	Center potential signal (optional)
7	Z	blue	Z-axis signal output
8	B	white	Button (The button has 2 pins, pin 2)

Connector pin definitions - (2-axis) analog voltage signal output:

Pin	Symbol	Color	Function description
1	V+	Red	Power+
2	X	Yellow	X axis signal output
3	GND	Black	GND power supply -
4	Y	Green	Y axis signal output
5	B	White	Button (The button has 2 pins, pin 1)
6	Cen	Brown	Center potential signal (optional)
7	B	white	Button (The button has 2 pins, pin 2)
8	NC		No connection

Connector pin definitions - (2-axis redundant signal) Analog voltage signal output:



Connector pin definitions - 2-axis co-directional dual signal output (signal curves as shown in the figure above)

Pin	Symbol	Color	Function description
1	V+	Red	Power+
2	X1	Green	X1 signal output
3	GND	Black	GND power supply -
4	Y1	Yellow	Y1 signal output
5	Y2	White	Y2 signal output
6	Cen	Brown	Center potential signal (VCC/2)
7	X2	Blue	X2 signal output
8	B	White	Button (button connected to pins 1 and 8)

Model number for redundant signal output: For example, H44B2G-45D-5 (with a "D" after the voltage 45).

Connector pin definitions - RS232 signal output:

Pin	Symbol	Color	Function description
1	V+	Red	Power+
2	RX	Yellow	RS232 data reception
3	GND	Black	GND power supply -
4	TX	Green	RS232 data transmission
5	X		(Factory Test X)
6	NC		Null
7	Y		(Factory Test Y)
8	B		(Factory Test Z)

Connector pin definitions - RS422 signal output:

Pin	Symbol	Color	Function description
1	V+	Red	Power+
2	RX+	Yellow	RS422 data reception is positive
3	GND	Black	GND power supply -
4	RX-	Green	RS422 data reception negative
5	X		(Factory Test X)
6	TX+	Blue	RS422 data transmission is in progress
7	Y		(Factory Test Y)
8	TX-	White	RS422 data transmission negative / (Factory Test Z)

Connector pin definitions - RS485 signal output:

Pin	Symbol	Color	Function description
1	V+	Red	Power+
2			
3	GND	Black	GND power supply -
4			
5	X		(Factory Test X)
6	RS485A+	Blue	RS485 data transmission is in progress
7	Y		(Factory Test Y)
8	RS485A-	White	RS485 data transmission negative / (Factory Test Z)

Connector pin definitions - CAN signal output:

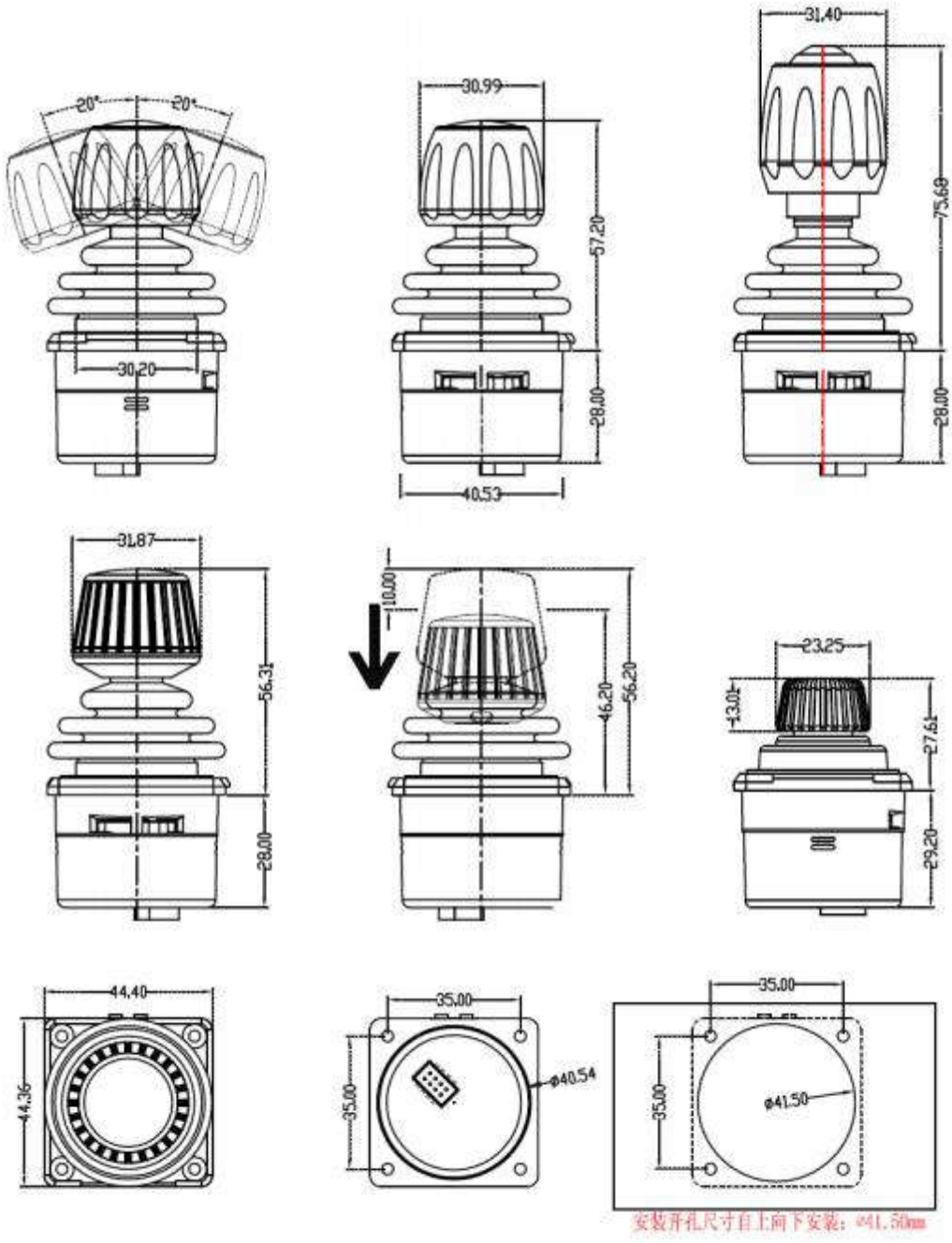
Pin	Symbol	Color	Function description
1	V+	Red	Power+
2	TX	Green	RS232 data transmission (RS232 is used for parameter settings)
3	GND	Black	GND power supply -
4	RX	Yellow	RS232 data reception (RS232 is used for parameter setting)
5	CAN-H	Blue	CAN signal high
6	Y		(Factory Test Y)
7	CAN-L	White	CAN signal low / (Factory Test Z)
8	X		(Factory Test Y)

Only four lines out: red, black, blue, and white. The RS232 is used for setting CAN communication parameters.

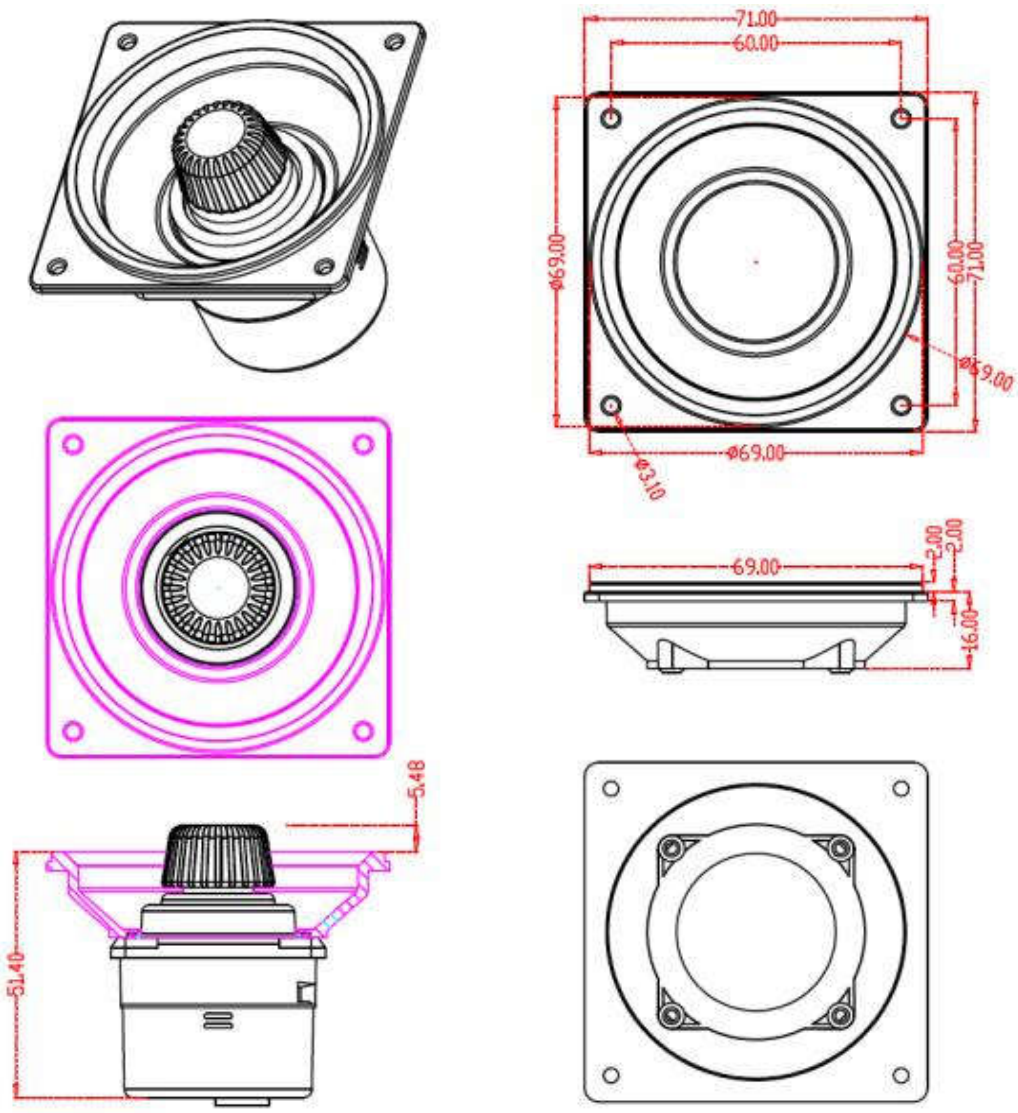
Connector pin definitions - USB signal output:

Pin	Symbol	Color	Function description
1	Z		Test-Z (for factory testing)
2	V+	Red	Power+
3	X		Test-X (for factory testing)
4	D-	White	USB signal output D-
5	Y		Test-Y (for factory testing)
6	D+	Blue	USB signal output D+
7	GND	Shield	USB cable shielding layer
8	GND	Black	GND power supply -

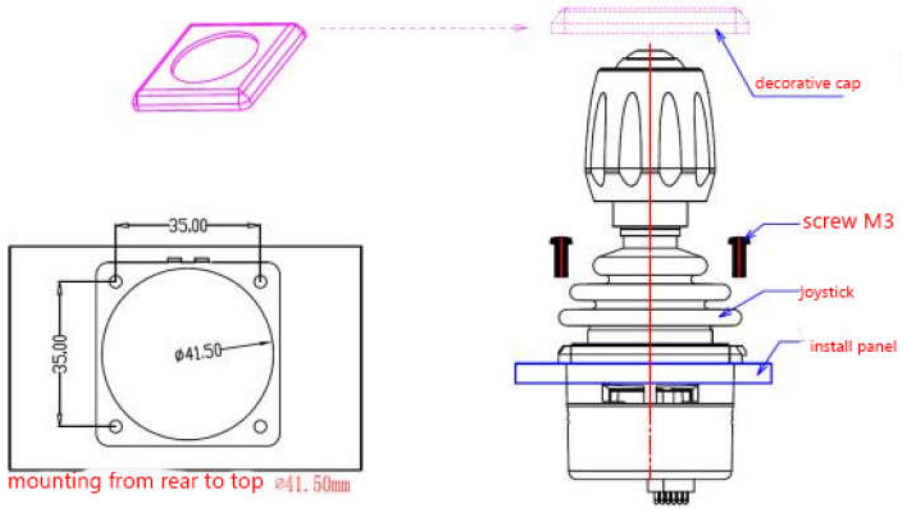
Outline dimensions drawing:



Mounting hole size, mount from top to bottom



Installation method:



MATE TECHNOLOGY SHENZHEN LIMITED
info@matend.com | www.matend.com | www.matend.cn

CAN Bus Communication Method:

- CAN2.0B
- CAN ID: Factory preset for standard or extended frames (modifiable via RS485)
- Baud Rate: 125K/250K/500K/1000K, default 250K (modifiable via RS485)
- Transmission Method: Timed transmission at intervals of 5-200ms, default 20ms (modifiable via RS485)
- Master-Slave Lookup Method (modifiable via RS485)

Joystick Data Transmission Format: (Hexadecimal HEX data)

BYTE0	XXL X Low position of axis	X-axis data 0X01C0~0X0800~0X0E40
BYTE1	XXH X Axis High Position	
BYTE2	YYL Y Low position of axis	Y-axis data 0X01C0~0X0800~0X0E40
BYTE3	YYH Y Axis High Position	
BYTE4	0X00 Z Low position of axis	Z-axis data 0X01C0~0X0800~0X0E40
BYTE5	0X00 Z Axis High Position	
BYTE6	Button	Button
BYTE7	0XA5	Tail

XXL X low position	XXH X high position	YYL Y low position	YYH Y high position	0X00 0X00	0X00 0X00	Button	A5 Tail

YYYY Y-axis angle
XXXX X-axis angle
Button

X-axis parameters

MAX LEFT MIN	STOP	MIN RIGHT MAX
0X01C0- -- -- -----0X07ff	0800	0X0801- -- --------- 0X0E40

Y-axis parameters

MAX UP MIN	STOP	MIN DOWN MAX
0X01C0- -- --------- 0X07ff	0800	0X0801- -- -------- 0X0E40

Z-axis parameters

MAX CCW MIN	STOP	MIN CW MAX
0X01C0- -- ----- 0X07ff	0800	0X0801- -- -- ----0X0E40

Button parameters

Bit7	Biy6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
		Button					

button = 1 indicates a button is pressed, 0 indicates no button is pressed

For example: 00 08 00 08 00 00 00 A5

I. General Protocol

(No address bits, applicable to RS232, RS422, RS485) This is the factory default mode;

1. Joystick Data Transmission (9 bytes) (Joystick-PC):

Function: Transmits the position parameters of each axis of the joystick.

Baud rate: 9600.8.1.N

Joystick data transmission format: (Hex data in hexadecimal)

FF	XXH	XXL	YYH	YYL	ZZH	ZZL	Button	CH
Header	X high	X low	Y high	Y low	Z high	Z low	button	checksum

XXXX X-axis angle
YYYY Y-axis angle
ZZZZ Z-axis angle

BB joystick button
CH =XXH+XXL+YYH+YYL+ZZH+ZZL+Button (00-FF)

X-axis parameters

MAX LEFT MIN	STOP	MIN RIGHT MAX
0X0020- -- -- ---0X01ff	0200	0X0201- -- ------ 0X03E0

Y-axis parameters

MAX Next/After MIN	STOP	MIN Up/before MAX
0X0020- -- -- 0X01ff	0200	0X0201- -- -- 0X03E0

Z-axis parameters (top view)

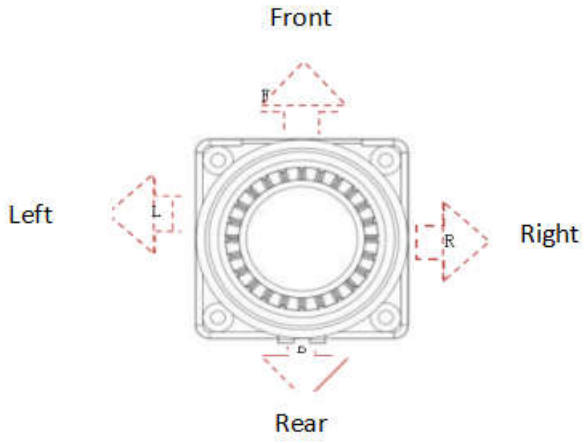
MAX CCW MIN	STOP	MIN CW MAX
0X0020- ------ 0X01ff	0200	0X0201- -- -- 0X03E0

Button parameters

Bit7	Biy6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	Joystick button	0	0000			

Joystick button = 1 indicates a button is pressed, 0 indicates no button is pressed.

Example: FF 02 00 02 00 03 E0 00 E7



2. Joystick Data Transmission with Address Bits (10 bytes) (Joystick-PC):

Function: Transmits the position parameters of each axis of the joystick.

Baud Rate: 9600.8.1.N

Joystick Data Transmission Format: (Hexadecimal data)

FF	Adr	YYH	YYL	XXH	XXL	ZZH	ZZL	Button	CH
Header	Adr	Y High	Y Low	X High	X Low	Z High	Z Low	Button	Checksum

Adr Address (01-7F)
YYYY Y-axis angle
XXXX X-axis angle
ZZZZ Z-axis angle
BB joystick button
CH = Adr + XXH + XXL + YYH + YYL + ZZH + ZZL + Button (00-FF) Low-order byte of the sum (1 byte)

Example: FF 01 02 00 02 00 02 00 00 07
Address is 01, checksum is 07

Mod bus Communication Protocol (RTU Mode)

Joystick Master Mode:

1. Baud Rate: 9600
2. Data Bits: 1 start bit, 8 data bits, 1 stop bit, parity bit
3. Communication Interface: RS485 and RS232 can only be selected, not used simultaneously
4. Data Format: Modbus
5. Operating Mode: Master (Master sends data to Slave 1)
6. Operating Mode: Joystick -> Slave
Command: Timed data transmission, frame interval 17ms, approximately 20Hz/frame;
Command: Frame interval can be changed by modifying the refresh rate (see Section 1.5 Refresh Rate Settings);
Command: Slave does not respond;

Function	Data	Parameter Range
Device address	0x01	Modbus Station Number
Function code	0x10	
First register address - high byte	0x40	register address
First register address - low byte	0x01	
Number of registers - high byte	0x00	
Number of registers - low byte	0x04	
Data length	0x08	
Button high bits Bit15-Bit8	0x00	1-16 buttons (Bit0 = Button 1) 1 = ON, 0 = OFF
Button low bits Bit7-Bit0	0X00	
Joystick 1 X-axis high position	0x02	0x0020-----0x01FF, 0x0100, 0x0201-----0x03E0
Joystick 1 X-axis low position	0x00	(max left min) stop (min right max)
Joystick 1 Y-axis high position	0x02	0x0020-----0x01FF, 0x0100, 0x0201-----0x03E0
Joystick 1 Y-axis low position	0x00	(max Down min) stop (min UP max)
Joystick 1 Z-axis high position	0x02	0x0020-----0x01FF, 0x0100, 0x0201-----0x03E0
Joystick 1 Z-axis low position	0x00	(max backward min) stop (min forward max)
CRC high bits	B4	
CRC low bits	43	

Send one frame every 45ms.
Example: 10 40 01 00 04 08 00 00 02 00 02 00 02 00 B4 43

Joystick Slave Mode:

(Master-Slave Mode)

1. Baud Rate: 9600/115200
2. Data Bits: 1 start bit, 8 data bits, 1 stop bit, parity bit
3. Communication Interface: RS485 and RS232 can only be selected, not used simultaneously
4. Data Format: Modbus
5. Operating Mode: Slave
6. Operating Mode: Master-Slave Mode (Receives a read command, retrieves 1 frame of data)

Register Address: 4001 (HEX)

Modbus master data reading and slave response format (function code 03) (PC → joystick)

1 byte	2 byte	3 byte	4 byte	5 byte	6 byte	7 byte	8 byte
Address	Read command	start address		number of registers		CRC	
0x01	0x03	High	Low	High	Low	High	Low

Example: 01 03 40 01 00 04 00 09

When the joystick receives this command, it switches from master mode to slave mode. (It stops actively sending data; the joystick only responds with the following data after receiving this read data);

Joystick Response (Joystick → PC)

Function	Data	Parameter Range
Device address	0x01	Device address
Function code	0x03	
Data length	0x08	
Button high bits Bit15-Bit8	0x00	1-16 buttons (Bit0 = Button 1) 1 = ON, 0 = OFF
Button low bits Bit7-Bit0	0X00	
Joystick 1 X-axis high position	0x02	0x0020-----0x01FF, 0x0200, 0x0201-----0x03E0
Joystick 1 X-axis low position	0x00	(max left min) stop (min right max)
Joystick 1 Y-axis high position	0x02	0x0020-----0x01FF, 0x0200, 0x0201-----0x03E0
Joystick 1 Y-axis low position	0x00	(max down min) stop (min up max)
Joystick 1 Z-axis high position	0x02	0x0020-----0x01FF, 0x0200, 0x0201-----0x03E0
Joystick 1 Z-axis low position	0x00	(max CCW min) stop (min CW max)
CRC high bits	94	
CRC low bits	ED	

For example: Host → Joystick: Device Address = 1: 01 03 40 01 00 04 00 09 Joystick → Host (Response): 01 03 08 00 00 02 00 02 00 02 00 94 ED

Joystick Communication Parameter Settings

Version: 17.11.20
Users may need to set and modify the joystick's communication parameters (including CAN, RS232, RS422);

All the above "parameter modifications" can only be performed through the joystick's RS422 or RS232 interface, including CAN parameters. PC → Joystick (RS422, RS485, or RS232) The host computer (serial port assistant) software sends commands to the joystick.

(If the serial port assistant software is not available, please contact our technical staff.)

If the host PC does not have an RS232 connector (DB9 9-pin connector), a USB to RS232 converter is available (a standard converter, not a TTL level converter).

The joystick has an RS422, RS485, or RS232 communication interface with a factory default baud rate of 9600.8.1.N.

I. Basic Commands:

1. ACK Confirmation (Joystick-PC) AA 55 AF
This indicates that the joystick successfully received the address setting command and completed its execution.

2. Set the joystick ID address;

ID refers to the ID in the RS232/RS422 communication protocol, or the ID in the CANopen protocol.

(PC->Joystick)

0xaf 0x0d 00 00 00 Add 0xf5
Header Command Data 1 Data 2 Data 3 Data 4 Tail
Add=0x01~0x7F Address 1-127
Add=0x00 Invalid

For example:
Set address 1 af 0d 00 00 00 01 f5 (HEX)
Set address 2 af 0d 00 00 00 02 f5 (HEX)
The joystick receives this command, executes it, and replies with ACK.

3. Reset the joystick (PC->Joystick)

0xaf 0x15 00 00 00 Add 0xf5
Header Command Data 1 Data 2 Data 3 Data 4 (tails)

Add=0x01~0x7f The address must match the joystick address to reset.
Add=0x00 Resets all joystick addresses; any address will be reset.
Add range outside 0-0x7f is invalid.

For example:
Reset all joystick addresses: af 15 00 00 00 00 f5 (HEX)
Reset joystick address 1: af 15 00 00 00 01 f5 (HEX)
Reset joystick address 2: af 15 00 00 00 02 f5 (HEX)

4. Set the joystick's center point (used to correct the center point position) (PC->Joystick) This is factory-set and can be ignored.

The PC connects to the joystick's RS422 port, baud rate 9600.
0xaf 0x09 00 00 00 00 0xf5
Header Command Data 1 Data 2 Data 3 Data 4 Tail
Sends some data to the joystick to reset its stop position (center point).

For example: af 09 00 00 00 00 f5 (HEX)

5. Communication port selection: (PC->Joystick)
Select one of the joystick's communication ports: RS232, RS422, or CAN; (Factory-set)
0xaf 0x05 XX 00 00 00 0xf5
Header Command Data 1 Data 2 Data 3 Data 4 Tail

XX=00 CAN Communication;
XX=01 RS232 Communication
XX=02 RS422 Communication
XX=03 RS485 Communication (Standard RS232/422/485 Protocol)
XX=04 RS485 Modbus RTU Communication

Example:
af 05 00 00 00 00 f5 (HEX) CAN Communication
af 05 01 00 00 00 f5 (HEX) RS232 Communication
af 05 02 00 00 00 f5 (HEX) RS422 Communication
af 05 03 00 00 00 f5 (HEX) RS485 Communication
af 05 04 00 00 00 f5 (HEX) RS485 Modbus RTU Communication

6. Refresh Rate Setting (PC->Joystick)

Refresh rate = frame interval for sending data, for example, setting it to 20ms (sending one frame of data to the host every 20ms)

0xaf	0x11	00	00	00	Ref	0xf5
Header	Command	Data 1	Data 2	Data 3	Data 4	Trailer

Ref = 0x0A~0x64 (10-100)ms, unit is "milliseconds"; (Factory default: 20ms)
Setting this parameter requires a reset or restart to take effect.

For example: Setting the joystick's data refresh rate to 20ms (sending one frame of data every 20ms, 50 times per second)

Set 20MS af 11 00 00 00 14 f5 (HEX)
Set 25MS af 11 00 00 00 19 f5 (HEX)
Set 33MS af 11 00 00 00 21 f5 (HEX)
Set 50MS af 11 00 00 00 32 f5 (HEX)

The joystick receives this command → replies with ACK → resets the joystick

Note: Lower baud rates require longer frame intervals.

Factory default: Refresh rate 20ms (CAN baud rate 250K, RS232 and RS422 baud rates 9600)

7. Communication Mode (Master-Slave Query, Timed Automatic Transmission, including CAN and RS232/422 Communication) (PC->Joystick)

Master-Slave Query: The joystick is the slave device. It only sends data back to the master when it receives a query command from the master. Automatic Timed Data Sending: The joystick sends data to the host computer upon power-on. The sending rate is specified in the "Refresh Rate Settings".

This parameter is permanently stored on the joystick (factory preset).

Format:

0xaf	0x08	00	00	00	Mode	0xf5
Header	Command	Data 1	Data 2	Data 3	Data 4	Tail

Mode=00 Timed Sending (Master Mode)
Mode=01 Master-Slave Query (Slave Mode)

Slave Mode: The joystick only responds when the host sends a query command.

Example: (PC->Joystick)

Timed Sending Mode (Master Mode) af 08 00 00 00 00 f5 (HEX)
Master-Slave Query Mode (Slave Mode) af 08 00 00 00 01 f5 (HEX)
After successful setting, the joystick returns ACK (AA 55 AF) (Joystick->PC)

8. Number of Joystick Signal Axes: (PC->Joystick) For manufacturer use only. Factory-configured; no user setup required:

2-axis: af 0c 00 00 00 00 f5 (HEX)
3-axis: af 0c 01 00 00 00 f5 (HEX)
4-axis: af 0c 02 00 00 00 f5 (HEX)

9. Query Device Information: (PC->Joystick)
af 20 00 00 00 00 f5 (HEX)

Joystick Response Format: Joystick->PC

FF	45	18	18 01 20	53 4D 43 34 35 41	ch
Header	Model	Version	Date	S M C 4 5 A	Checksum

Checksum = Sum of all bytes except the header (FF), taking the least significant byte of the result.

The above information includes the following:
Model Code: 45
Version: 1.8
Date: 2018-01-20
Name: SMC45A

9.1. Setting the Mid-Die Size (PC->Joystick):

Data Format:

0xaf	0x19	D1	D2	D3	D4	0xf5
Header	Command	Data 1	Data 2	Data 3	Data 4	Tail

D1: Dead Size - High Byte
D2: Dead Size - Low Byte
(D3 and D4 = 0x00)
Factory Default Register Address = 0x01F0
Permanently stored in the joystick after setting
Example: Dead Size 0X0123: af 19 01 F0 00 00 f5 (HEX)

II. RS232, RS422, and RS485 Communication Parameter Settings

10. Setting the Baud Rate for RS232, RS422, and RS485 (PC->Joystick) RS232 and RS422 baud rates are the same; settings are effective simultaneously.

0xaf	0x0b	00	00	00	Baud	0xf5
Header	Command	Data 1	Data 2	Data 3	Data 4	Trailer

Baud=0X00 Baud Rate=9600
Baud=0X01 Baud Rate=19200
Baud=0X02 Baud Rate=57600
Baud=0X03 Baud Rate=115200
Baud=0X04 Baud Rate=2400
Baud=0X05 Baud Rate=4800

Example:
Setting 9600 af 0b 00 00 00 00 f5 (HEX)
Setting 19200 af 0b 00 00 00 01 f5 (HEX)
Setting 57600 af 0b 00 00 00 02 f5 (HEX)
Setting 115200 af 0b 00 00 00 03 f5 (HEX)
Setting 2400 af 0b 00 00 00 04 f5 (HEX)
Setting 4800 af 0b 00 00 00 05 f5 (HEX)
The joystick receives this command, executes it, and replies with ACK.

11. Check Joystick Position (PC->Joystick)
This command is only effective in "Master-Slave Query" mode.
No query command outputs any data from the joystick; the joystick replies once for each query.

0xaf	0x07	00	00	00	Addr	0xf5
Header	Command	Data 1	Data 2	Data 3	Data 4	Tail

Addr Address = 0x01-0x7f When the address is correct, send back the current position.

The joystick sends back the current position when it receives this data. It sends back the current position once the address is checked; no data is sent if the address is not checked. For example, in RS232 communication, the query is:

(PC->Joystick) af 07 00 00 00 01 f5 (HEX)

(Joystick->PC) FF 01 08 00 70 00 00 00 00 00 79
The joystick sends back the current position when it receives this data.

III. Setting Communication Parameters for Controlling the Camera

12. Setting the Camera Address (PC->Joystick)

0xaf 0x0f Cam 00 00 00 0xf5
Header Command Data 1 Data 2 Data 3 Data 4 Trailer

Cam=0x00-0x7f

Example: Camera Address 01; af 0f 01 00 00 00 f5
Camera Address 02; af 0f 02 00 00 00 f5

13. Setting the Communication Protocol (PC->Joystick)
Only applicable to RS232/RS422/RS485

0xaf 0x0e Pro 00 00 00 0xf5
Header Command Data 1 Data 2 Data 3 Data 4 Trailer

Pro =0X00 Standard 9-byte/10-byte or Modbus protocol
Pro =0X01 Camera communication protocol PELCO-D
Pro =0X02 Camera communication protocol PELCO-P

Standard protocol: af 0e 00 00 00 00 f5 (HEX) Default
PELCO-D: af 0e 01 00 00 00 f5 (HEX)
PELCO-P: af 0e 02 00 00 00 f5 (HEX)

III. CAN Communication Parameter Settings: (CAN parameter settings must also be configured via RS232 or RS422 port;)

14. CAN Port Baud Rate: (PC->Joystick)
0xaf 0x06 XX 00 00 00 0xf5
Header Command Data 1 Data 2 Data 3 Data 4 Trailer

XX=00 125K
XX=01 250K (Default)
XX=02 500K
XX=03 1000K

Example: af 06 00 00 00 00 f5 (HEX) CAN Baud Rate = 125K
af 06 01 00 00 00 f5 (HEX) CAN Baud Rate = 250K (Default)
af 06 02 00 00 00 f5 (HEX) CAN Baud rate = 500K
af 06 03 00 00 00 f5 (HEX) CAN Baud rate = 1000K

15. CAN Protocol Settings: (PC->Joystick)

0xaf	0x0a	00	00	TP	SS	0xf5
Header	Command	Data 1	Data 2	Data 3	Data 4	Tail

SS=00 Normal Protocol ID=Sending Node ID, see (11) Joystick Sending Node ID Settings) Default
SS=01 CANopen Protocol ID=180+ID (see (2) Setting Joystick ID Address) The factory has already set it for the customer
SS=02 Northern Vehicle Instrument 2-Axis Protocol (500K, 0X2A1, 10hz, 0-128-255) Refresh rate is set separately;
SS=03 Northern Vehicle Instrument 3-Axis Protocol (500K, 0X789, 10hz, 0-3F, 7F, C0-FF) Refresh rate is set separately;

TP is TPDO in the CAN OPEN protocol

TP=00: TPDO1 sends ID 0X0180+ID (see I. 2, setting the joystick)
TP=01: TPDO2 sends ID 0X0280+ID (see I. 2, setting the joystick)
TP=02: TPDO3 sends ID 0X0380+ID (see I. 2, setting the joystick)
TP=03: TPDO4 sends ID 0X0480+ID (see I. 2, setting the joystick ID address)

For example:
Af 0a 00 00 00 00 f5 (HEX) Standard Protocol
Af 0a 00 00 00 01 f5 (HEX) CAN open Protocol TPDO1
Af 0a 00 00 00 02 f5 North Vehicle Instrument 2-Axis Protocol
Af 0a 00 00 00 03 f5 North Vehicle Instrument 3-Axis Protocol

16. Joystick "Send Node ID" Setting: (PC->Joystick)
Only applicable to "Normal Protocol," this command is not used in the CANopen protocol.

0xaf	0x01	D1	D2	D3	D4	0xf5
Header	Command	Data 1	Data 2	Data 3	Data 4	Trailer

D1.7=0 Extended Frame 29 bits
D1.7=1 Standard Frame 11 bits

29-bit Extended Frame: Data range 0X0-0X0FFFFFFF, data D1-D4 correspond to "Node Identifier Code." For example: Setting the send node identifier code - extended frame "0X00F0F101"
af 01 00 f0 f1 01 f5 (HEX)

11-bit Standard Frame: Data range 0X000-0X7FF, data D3-D4 correspond to "Node Identifier Code." For example: Setting the send node identifier code - standard frame "0X181"
af 01 80 00 01 81 f5 (HEX)

17. Joystick “Receiver Node ID” Setting: (PC->Joystick)
Only applicable to “Normal Protocol”, CAN open protocol does not use this command

0xaf	0x02	D1	D2	D3	D4	0xf5
Header	Command	Data 1	Data 2	Data 3	Data 4	Trailer

D1.7=0 Extended Frame 29 bits
D1.7=1 Standard Frame 11 bits

29-bit Extended Frame: Data range 0X0-0X0FFFFFFF, data D1-D4 correspond to “Node Identifier Code”. For example: Setting Receiver Node Identifier Code - Extended Frame “0X00F0F101”
af 02 00 f0 f1 01 f5 (HEX)

11-bit Standard Frame: Data range 0X000-0X3FF, data D3-D4 correspond to “Node Identifier Code” For example: Set the receiving node identifier code - standard frame "0X1E1"
af 02 80 00 01 E1 f5 (HEX)

18. Joystick "Mask Node ID" setting: (PC->Joystick)

0xaf	0x03	D1	D2	D3	D4	0xf5
Header	Command	Data 1	Data 2	Data 3	Data 4	Trailer

D1.7=0 Extended frame 29 bits
D1.7=1 Standard frame 11 bits

29-bit extended frame: Data range 0X0-0X0FFFFFFF, data D1-D4 correspond to "node identifier code"
For example: Set the mask node identifier code - extended frame "0X00002201"
af 03 00 00 22 01 f5 (HEX)

11-bit standard frame: Data range 0X000-0X3FF, data D3-D4 Corresponding to the "Node Identifier Code", for example: Setting the masked node identifier code - standard frame "0X122"
af 03 80 00 01 22 f5 (HEX)

IV. Mod bus RTU (RS485) Communication Parameter Settings:

ACK is returned upon successful setting. Data format: AA 55 AF

1. Joystick Operating Mode (PC->Joystick):
Format: 0xaf 0x08 00 00 00 Mode 0xf5
Header Command Data 1 Data 2 Data 3 Data 4 Tail

Mode=00 Master
Mode=01 Slave

Master af 08 00 00 00 00 f5 (HEX)
Slave af 08 00 00 00 01 f5 (HEX)
ACK is returned upon successful setting.

2. Setting "Device Address" (PC->Joystick) Factory Default 1

0xaf	0x0d	00	00	00	Add	0xf5
Header	Command	Data 1	Data 2	Data 3	Data 4	Tail

Add=0x01~0x7F Address 1-127
Device Address=1 af 0d 00 00 00 01 f5 (HEX)
Device Address=2 af 0d 00 00 00 02 f5 (HEX)
Device Address=3 af 0d 00 00 00 03 f5 (HEX)
Device Address=4 af 0d 00 00 00 04 f5 (HEX)
The joystick receives this instruction, executes it, and replies with ACK

3. Set Register Address (PC->Joystick): The setting is permanently stored in the joystick.

Data Format:

0xaf	0x18	D1	D2	D3	D4	0xf5
Header	Command	Data 1	Data 2	Data 3	Data 4	Tail

D1: High byte of register address
D2: Low byte of register address
Factory default register address=0x4001 (16385)
Register address = 0X4001 af 18 40 01 00 00 f5
Register address = 0X4002 af 18 40 02 00 00 f5

Siemens - Register address 40001 af 18 00 00 00 00 f5
Siemens - Register address 40002 af 18 00 01 00 00 f5
Siemens - Register address 40003 af 18 00 02 00 00 f5

For example: Set register address = 0x4001 (hexadecimal). If it is octal, convert it to hexadecimal.

Register address = 0X4001 af 18 40 01 00 00 f5 Successful setting returns a read value: 01 03 40 01 00 04 00 09

USB Interface Connection Cables and Communication Protocols

USB Communication Protocol: USB 2.0 HID Human-Machine Interface Protocol Standard

Supports Microsoft operating systems, driver less; supports Direct X library

Related examples can be found online under "joystick direct x input"

I. Data Format Sent by USB Keyboard (7-byte HEX):

The USB keyboard sends the angle parameters of the keyboard's 3-axis joystick and the status values of key presses.

byte1	byte2	byte3	byte4	byte5	byte6	byte7
XXL	XXH	YYL	YYH	ZZL	ZZH	BB1

XXXX: X-axis data, 0000-03FF, (BYTE2 high byte, BYTE1 low byte)
0X0020-0X01FF Left
0X0200 Stop
0X0200-0X03DF Right

YYYY: Y-axis data, 0000-03FF, (BYTE4 high byte, BYTE3 low byte)
0X0020-0X01FF Up
0X0200 Stop
0X0200-0X03DF Down

ZZZZ: Z-axis data, 0000-03FF, (BYTE6 high byte, BYTE5 low byte)

0X0020-0X01FF Counterclockwise (wide)
0X0200 Stop
0X0200-0X03DF Clockwise (tele)

BB1: Button #1 Group

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Button 8	Button 7	Button 6	Button 5	Button 4	Button 3	Button 2	Button 1

1. USB Set Center Point Command (Command sent via USB): The joystick receives the command from the host. f5 00 00 00 00 01 55 56 This is factory set and not required by the user.

2. Set Low-order Byte of USB PID (Command sent via USB):

Format: f5 0A Pid 00 00 00 CH (CH checksum: low-order byte of the sum of 2-6 bytes)

Example:
PID=0061 F5 0a 61 00 00 00 00 6b (Default)
PID=0062 F5 0a 62 00 00 00 00 6c
PID=0063 F5 0a 63 00 00 00 00 6d
PID=0064 F5 0a 64 00 00 00 00 6e
PID=0065 F5 0a 65 00 00 00 00 6f

The following technical support materials are available for development:

1. USB testing software
2. USB programming examples
3. USB keyboard communication protocol
4. Driver-free, supports Windows Direct X. Please search "direct X language related design" on Baidu. Input joystick. Various options are available online.