

MKF-JS16 Dual Engine Controller



A2 version(diesel engine control) 2021

MKF-JS16 Dual engine throttle controller. Suitable for ship engine throttle control, can control 1 engine, with throttle control, gear control function. Mechanical gear DNR , synchronous indication. Adopt Hall sensor, full temperature range linear compensation, stainless steel housing, waterproof and salt spray proof design, friction resistance positioning.

1. Features:

- Sensor: Hall sensor
- Operating Angle: 156 Spend
- Gear :3 File (DNR)
- life :Greater than500 10,000 times ;
- Signal output:Analog voltage0~4.7V , CAN , 485
- powered by :DC5V (DC12V/DC24V)
- Operating temperature: -40 °C ~ +70 °C
- Protection :IP67

DNR indicator light

D : Forward gear

When the handle is at this position, D When the red light of the gear is on and the handle is pushed forward, it is throttle control . The green light of the gear is on;

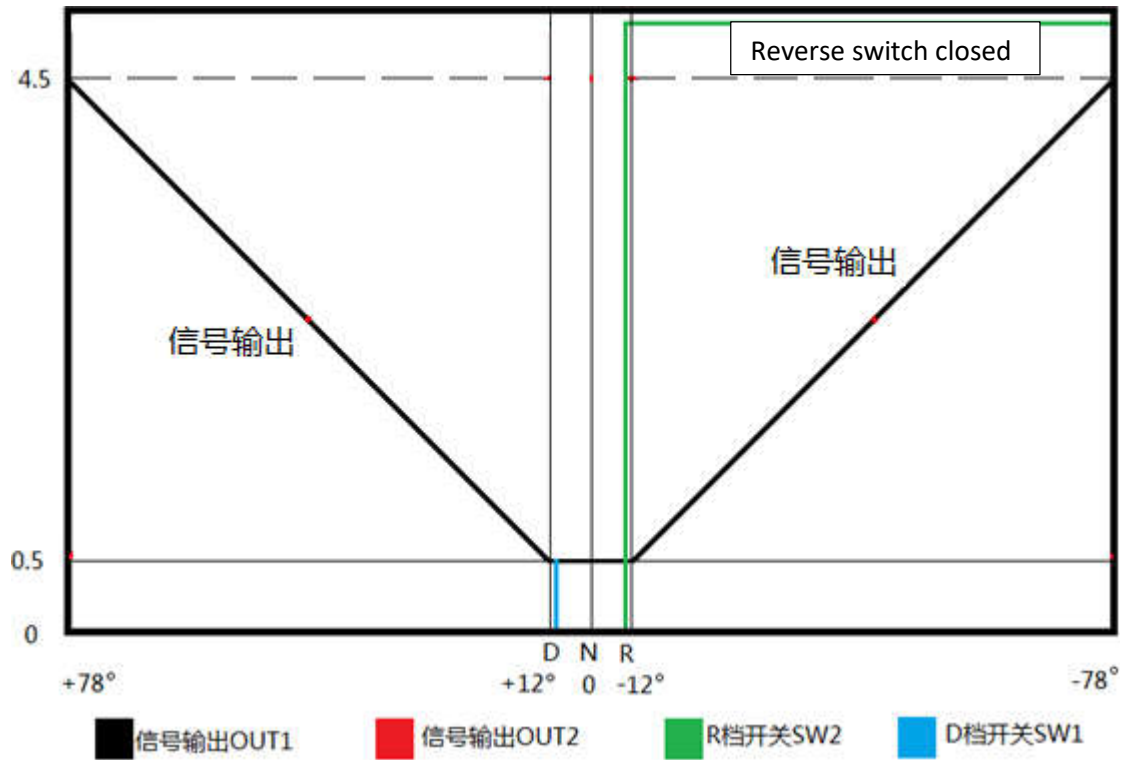
R : Reverse gear

When the handle is at this position, R The red light of the gear is on, and the handle is pulled back, which is the throttle control . The green light of the gear is on.

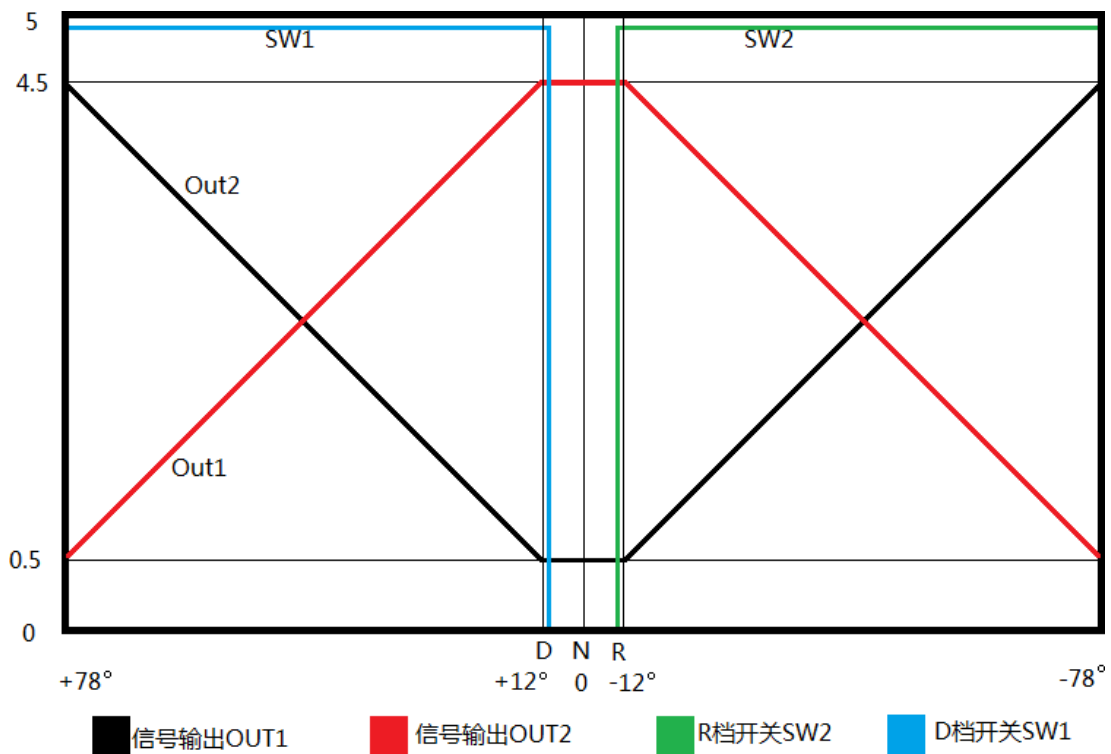
N : Neutral

When the handle is in this position, N The red light of the gear is on;

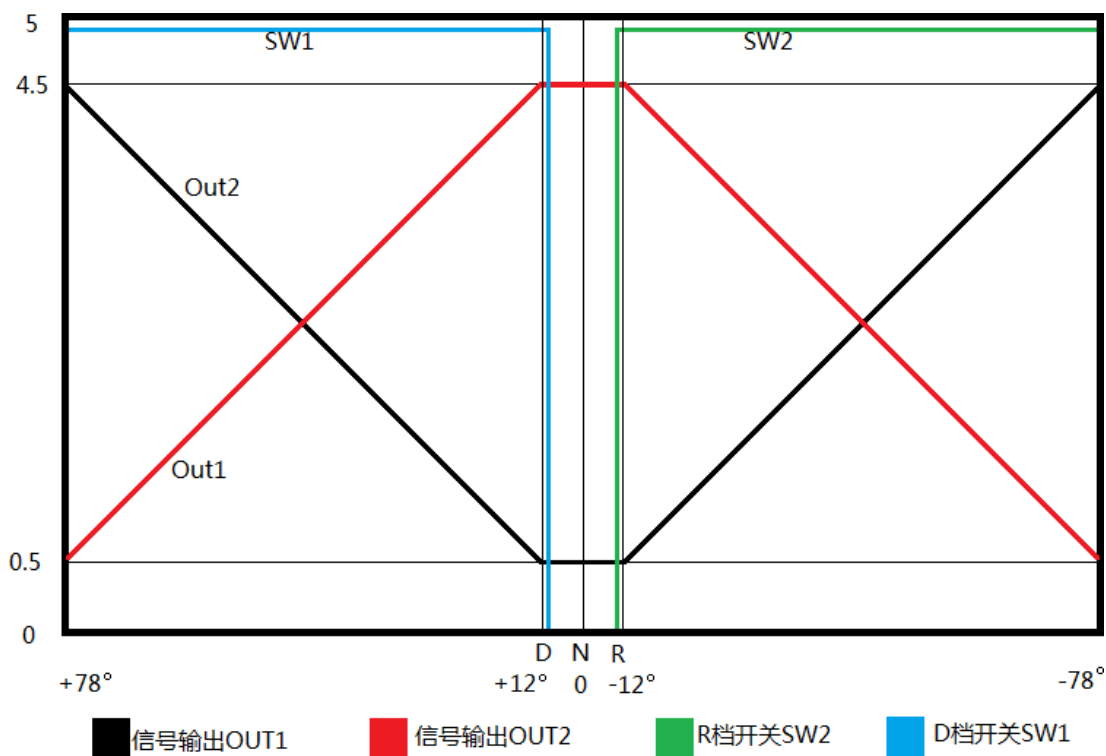
Dual signal output, V Shape signal output, reverse switch



Dual signal output,



Analog voltage signal output: left and right independent voltage signal output



2. Analog voltage signal wiring diagram: (2 * 7Core aviation plug)
 Matching plug-in model: Xianpu SP1310 /P7 , as shown below



Same wiring for left and right engines

2*7 aviation plugs Core, 1 each for left and right Engines (Aviation plug 7 core)

Pins	symbol	color	Function
1	P+		Power supply +5V or12-24V
2	OUT1		Signal output1 (voltage or current)
3	OUT2		Signal output2 (voltage or current)
4	P- (GND)		Power supply 0V (GND)
5	COM		Gear common terminal
6	SW- D		D gear switch
7	SW- R		R gear switch

5. CAN+RS485 connector wiring diagram Power supply
 (aviation plug 4 core)

Pins	symbol	Internal color	Function
1	GND	black	0V(GND)
2	V+	red	V+ (power positive)
3			
4			

Signal (Aviation Plug 7 core)

Pins	symbol	color	Function
1	GND	black	Shield ground (0V)
2	CAN- H	blue	CAN- H
3	CAN- L	White	CAN- L
4	RS485A	yellow	RS485+
5	RS485B	green	RS485-
6			
7			

CAN Parameter setting is done via RS485 interface

5. CAN+RS485X2 Connector wiring
diagram (Eurocard) power
supply (aviation plug 4 -pin)

Pins	symbol	Internal color	Function
1	GND	black	0V(GND)
2	V+	red	V+ (power positive)
3			
4			

Signal (Aviation Plug 7 core)

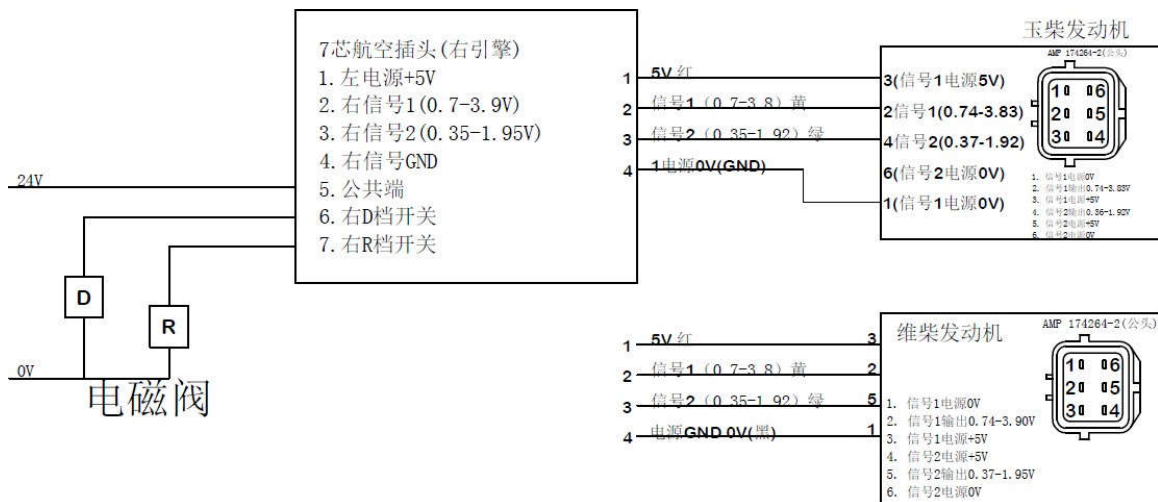
Pins	symbol	color	Function
1	GND	black	Shield ground (0V)
2	CAN- H	blue	CAN- H
3	CAN- L	White	CAN- L
4	1- RS485A	yellow	RS485+
5	1- RS485B	green	RS485-
6	2- RS485A	orange color	RS485+
7	2- RS485B	grey	RS485-

CAN Parameter setting is done via RS485 interface

7. Yuchai / Weichai engine connector wiring diagram

2 7 aviation plugs Core, 1 each for left and right Engines (Aviation plug 7 core)

Pins	symbol	color	Function
1	+5V		Power supply +5V
2	OUT1		Signal output (0.74-3.83)
3	OUT2		Signal 2 Output (0.37-1.94V)
4	GND		Power GND
5	COM		Switch common
6	SW- D		D gear switch
7	SW- R		R gear switch

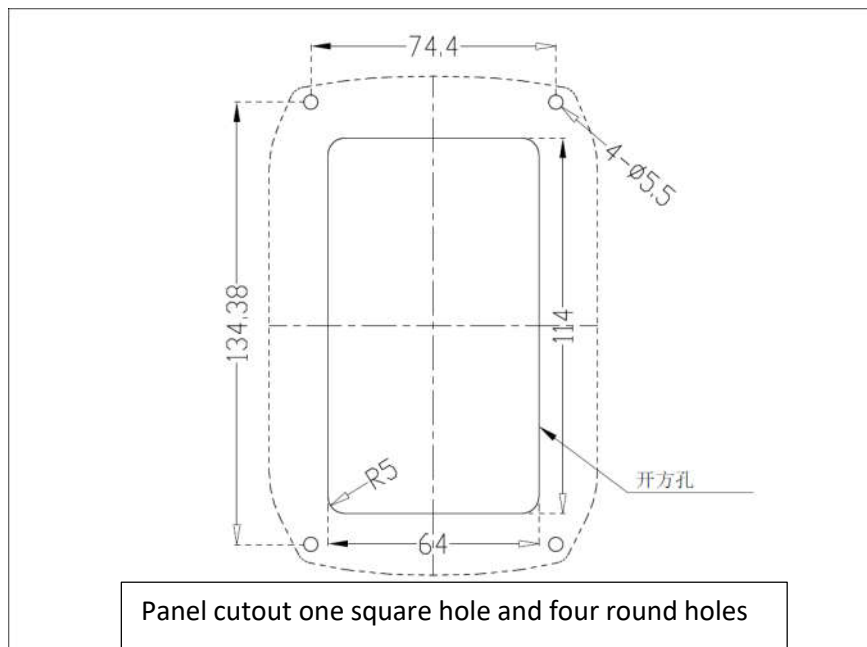
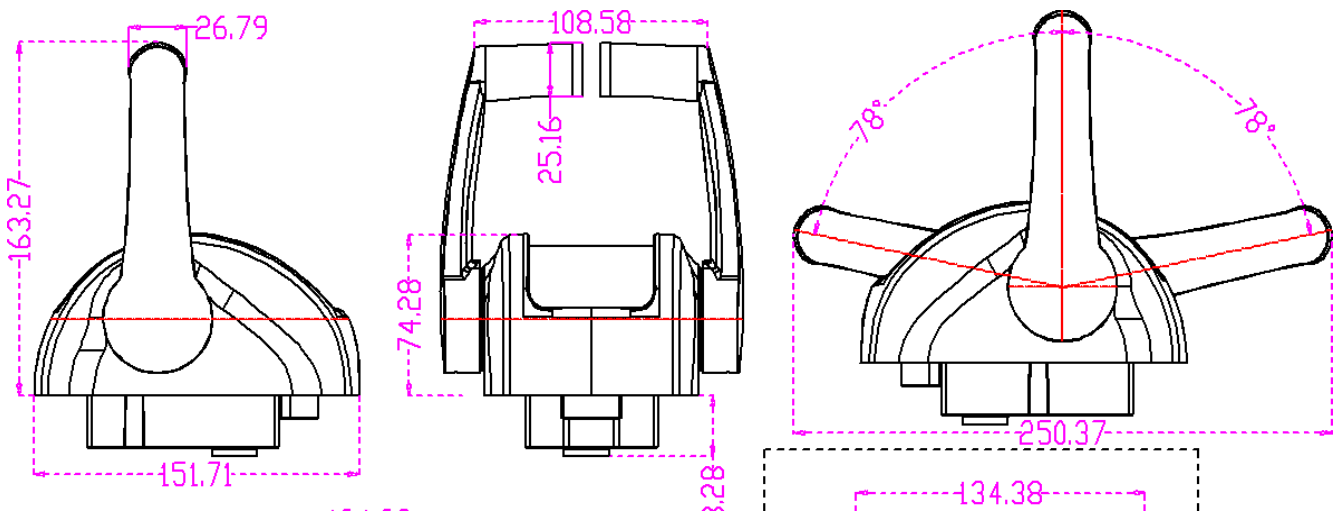


8. 4-20mA Wiring Diagram

2*7 aviation plugs Core, left and right rudders are the same (aviation plug 7 core)

Pins	symbol	color	Function
1	+24V	red	Power supply 24V+
2	GND	black	Power supply 0V
3	20mA	yellow	Signal output 4-20mA
4	GND	purple	Signal ground GND
5	COM	Brown	R&D Switch common terminal
6	D	white	D Shift / forward switch
7	R	blue	R Shift / reverse switch

9. Installation dimension drawing:



RS485 Protocol Format

Data bits: 8 bytes

Baud rate: 9600

Start bit 1, stop bit 1, invalid check bit

Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0xFF	0x01	xxR	xxL	ssR	ssL	Button	CH
	Fixed value	Right handle gear	Left handle gear	Right handle angle	Left handle angle	Button	Checksum

Note: CH = Byte1 + Byte2 + Byte3 + Byte4 + Byte5 + Byte6, the low byte of the sum of all bytes except the header

xxR/xxL Parameter Explanation

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
				Reverse throttle	R gear	D gear	Forward throttle

Note: Valid = 1

Invalid = 0

N gear=0x00

D gear=0x02

R gear=0x04

Forward throttle =0x01

Reverse throttle =0x08

ssL/ssR throttle angle value

Parameter range: 0x00 -0x64 (hex adecimal) or 0-100 (decimal)

Numerical 0 means throttle0%

Numerical100 means throttle100%

Button Parameter Explanation

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
				Lower left button	Upper left button	Upper right button	Lower right button

Note: Valid = 1 Invalid = 0

Button pressed = 0x01 Button Release = 0x00

CAN Protocol Format

Data frame: 8 bytes

CAN ID : Supports extended frames and standard frames (see CAN ID settings below), default ID=0CFA0001

BYTE0	BYTE1	BYTE2	BYTE3	BYTE4	BYTE5	BYTE6	BYTE7
0x01	xxR	xxL	ssR	ssL	Button	0x00	0xA5
Fixed value	Right handle gear	Left handle gear	Right handle angle	Left handle angle	Button	Fixed value	Fixed value

Note: The right handle is not used on the MKF-JS16 series products (xxR and ssR are not used and their values are meaningless)

xxL parameter explanation

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
				Reverse throttle	R gear	D gear	Forward throttle

Note: Valid = 1 Invalid = 0

N gear=0x00

D gear= 0x02

R gear= 0x04

Forward throttle = 0x01

Reverse throttle = 0x08

ssL throttle speed value

Parameter range: 0x00 -0x64 (hex adecimal) or 0-100 (decimal)

Numerical0 means throttle0%

Numerical100 means throttle100%

Button Parameter Explanation

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
				Lower left button	Upper left button	Upper right button	Lower right button

Note: Valid = 1 Invalid = 0

Button pressed = 0x01

Button Release = 0x00

2. Data format received by the throttle controller (host → throttle controller)

pass CAN bus Bus to control the indicator light, receive ID

Can be set to receive CAN 2.0B: ID standard frame 0x102

Data frame: 8 bytes

BYTE0	BYTE1	BYTE2	BYTE3	BYTE4	BYTE5	BYTE6	BYTE7
0xFF	Left LED	Right LED	0x00	0x00	0x00	0x00	0xA5
head							tail

Left LED

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	0	0	0	Lower left red light	Upper left red light	Upper left Green Light

Note: On = 1 Off = 0

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	0	0	0	Lower right red light	Upper right red light	Upper right green light

Right LED

Note: On = 1 Off = 0

RS485 Modbus RTU Communication Protocol

1. Baud rate: 9600 (modifiable)
2. Data bits: 1 start bit , 8 data bits , 1 stop bit, invalid check bit
3. Communication interface: RS485

Modbus RTU master mode :

Function	data	Parameter range
Device Address	0x01	Modbus Station No.
Function code	0x10	
1st Register address - high bit	0x00	Register address: default 0X0001 Register address can be modified
1st Register address - low	0x01	SiemensPLC: 40001+1=40002
Register number - high	0x00	
Register number - low	0x03	
Data length	0x06	
Button high bit Bit15- Bit8	0x00	00
Button low bit Bit7- Bit0	0X00	Bit3 Lower left button /Bit2 Upper left button / Bit1 Upper right button /Bit0 Lower right button
Right handle gear	0x00	N = 0x00/D = 0x02/R = 0x04/Forward Throttle = 0x01/ Reverse throttle = 0x08
Left handle gear	0x00	N = 0x00/D = 0x02/R = 0x04/Forward Throttle = 0x01/ Reverse throttle = 0x08
Right throttle opening	0x00	0-100
Left throttle opening	0x00	0-100
CRC high bit		
CRC low bit		

Button : 1=ON, 0=OFF

Timing 25ms Send one frame (the sending cycle rate can be set)

Modbus RTU slave mode

Modbus Host reads data and slave responds in format (Function code **03**) (PC → Joystick)

Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
address	Read command	Start address		Number of registers		CRC	
0x01	0x03	high	Low	high	Low	high	Low

Example : 01 03 00 01 00 03 54 0B

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

Joystick response (joystick → PC)

Function	data	Parameter range
Device Address	0x01	Device Address
Function code	0x03	
Data length	0x06	
Button high	0x00	00
Button low	0x00	Bit3 Lower left button / Bit2 Upper left button / Bit1 Upper right button / Bit0 Lower right button
Right handle gear	0x0c	N = 0x00/D = 0x02/R = 0x04/Forward Throttle = 0x01/ Arrival throttle = 0x08
Left handle gear	0x0c	N = 0x00/D = 0x02/R = 0x04/Forward Throttle = 0x01/ Arrival throttle = 0x08
Right throttle opening	0x64	0-100
Left throttle opening	0x64	0-100
CRC high bit		
CRC low bit		

01 03 06 00 00 0c 0c 64 64 C9 0D

For example:

1 03 00 01 00 03 54 0B

Received: 01 03 06 00 00 0c 0c 64 64 C9 0D

Joystick communication parameter settings

Version:Ver: 19.12.01

Users may need to set and modify the communication parameters of the joystick (including CAN, RS485)

All the above-mentioned "parameter modifications" can only be set through the RS485 interface of the joystick, including CAN parameters. PC → Joystick (RS485) The host computer (serial port assistant) software sends commands to the joystick.

(If the serial port assistant software is not available, you can ask our technical staff for it)

If the host PC does not have RS232 (DB99 -pin connector)you need to buy a USB to RS485 converter; if your computer has an RS232 port, you need to buy an RS232 to RS485 converter.

RS485 communication interface on the joystick , the factory default baud rate is 9600.8.1.N

1. Basic instructions:

1. ACK confirmation (joystick - PC)

AA 55 AF

This indicates that the joystick has successfully received the address setting instruction and has completed the execution.

2. Set the joystick ID address;

ID refers to the ID in the RS232/RS422 communication protocol , or the ID in the CAN open protocol (PC->joystick)

0xaf 0x0d 00 00 00 Add 0xf5

Header command data1 data2 data3 data4

tails

Add=0x01~0x7F Address1-127

Add=0x00 is invalid (when address=0 ,there is no address bit in RS232 or RS422)

For example:

set up1Address af 0d 00 00 00 01 f5 (HEX)

set up2Address af 0d 00 00 00 02 f5 (HEX)

The joystick receives this command, executes it, and responds ACK

3. Reset the joystick (PC -> joystick)

0xaf 0x15 00 00 00 Add 0xf5

Header command

data1 data2 data3 data4

tail Add=0x01~0x7f

The address must be the same as the joystick address to be reset. Add = 0x00

resets the joystick of all addresses. Any address is reset .

Not in range 0-0x7f invalid

For example:

```

Reset all address      af 15 00 00 00 00 f5      (HEX)
  joysticks:
Reset the joystick at address af 15 00 00 00 01 f5      (HEX)
  1 :
Reset the joystick at address af 15 00 00 00 02 f5      (HEX)
  2 :

```

3. Set the center point of the joystick (for center point correction) (PC -> Joystick)

It has been set at the factory, so you can ignore this command.

Connect the PC to the RS422 port of the joystick, baud rate 9600

```

0xaf  0x09  00  00      00  xx  0xf5
head  Order  Data1 Data2      data3 data4  tails

```

```

xx=01  D  gear  position
xx=02  N  gear  position
xx=03  R  gear  position

```

Send some data to the joystick to reset the joystick's stop

position. For example (HEX):

```

D gear      af 09 00 00 00 01 f5
setting:
N gear      af 09 00 00 00 02 f5
setting:
R gear      af 09 00 00 00 03 f5
setting:

```

4. Signal output port selection: (PC -> Joystick)

Joystick signal output port, select one of the following options; (factory settings have been made for customers)

```

0xaf  0x05  XX  00      00  00  0xf5
head  Order  Data1 Data2      data3 data4  tails

```

```

XX=00  CAN communication;
XX=01  RS485 communication
XX=02  RS485 Modbus RTU communication
XX=10  analog voltage (0.5-4.5V), 2 Signal output, cross curve;
XX=11  analog voltage (0.5-4.5V), 2 signal output, independent
front and rear curves, forward 1 signal output, backward 1 signal,
front and rear signals are separated;
XX=12  analog voltage (0.5-4.5V, mid 2.5V), 2 Signal output, cross
curve
XX=13  analog voltage dual -channel same-direction signal output
(Yuchai, Weichai engine)

```

(1 = 0.74-3.84V/ median0.74V; 2 = 0.37-1.92V median0.37)

XX=14 analog voltage (0.1-4.6V), 2 -way signal output, cross curve;

XX=15 analog voltage 1-4V, 2- way signal output in the same direction, DC motor control ;

XX=20 analog current signal (4-20mA), median12mA;

XX=21 analog current signal (4-20mA), mid-position 12mA, FNR The middle value can change

XX=22 analog current signal (4-20mA), the middle 4mA, forward4-20, backward 4ma constant

eg:

af 05 00 00 00 00 f5	CAN communication
af 05 10 00 00 00 f5	Analog voltage signal, median 0.5V, 2- way cross curve;
af 05 11 00 00 00 f5	Analog voltage signal, median 0.5V, independent front and back curves
af 05 12 00 00 00 f5	Analog voltage signal, median 2.5V, 2- way cross curve
af 05 13 00 00 00 f5	0.74-3.84V & 0.37- 1.92V Yuchai & Weichai
af 05 14 00 00 00 f5	Voltage (0.1-4.6V median 0.1V), 2 -way cross curve

af 05 20 00 00 00 f5 analog current signal (4-20mA), N gear/neutral position 12mA;

5. (Dual CPU board/Weichai) Median analog voltage signal output - error adjustment setting (PC > joystick)

0xaf 0x0E 01 01 00 00 0xf5
head Order Out1/2 +/- tail

Out1/2 : 01=No.1 signal 02=No.1 signal

+/- : 01=+1 02=-1 (minus 1)

Signal1 +1: af 0E 01 01 00 00 f5

Signal1 - 1: af 0E 01 02 00 00 f5

Signal2 +1: af 0E 02 01 00 00 f5

Signal2 - 1: af 0E 02 02 00 00 f5

5.2 Output Maximum - Error Adjustment Setting (PC -> Joystick)

For example, if the maximum value of the output signal is not enough, fine-tuning is required

0xaf 0x0f 01 00 00 00 0xf5
head Order +/- tail

+/- : 01=+1 02=-1 (minus1)

Signal +1: af 0f 01 00 00 00 f5

Signal - 1: af 0f 02 00 00 00 f5

6. Send cycle setting (PC -> Joystick)

Refresh rate = the frame interval time of sending data, such as setting 20ms (per 20MS

Send a frame of data to the host) 0xaf 0x11 00 00 00 Ref 0xf5

Header command data1 data2 data3 data4 tails

Ref = 0x0A ~ 0x64 (10-100)ms, unit is "millisecond"; (factory default:

20ms) After setting this parameter, reset or restart will take effect

For example: Set the refresh rate of the joystick to send data to 20MS (send 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

ACK → resets the joystick Note: If the baud rate is

low, the corresponding frame interval will be longer

Factory default: refresh rate 20ms (CAN baudrate250K, RS232 andRS422 baudrate9600)

7. Communication mode (master-slave query, timed automatic transmission, including CAN and RS485 communication) (PC -> joystick)

Master-slave query: The joystick is a slave device and will only send data back to the host after receiving the host's query command. Timing automatic transmission: The joystick will send data to the host when it is turned on. The transmission rate refers to the "Refresh rate setting". This parameter is permanently stored in the joystick (it has been set for the customer before leaving the factory).

Format:

0xaf	0x08	00	00	00	Mode	0xf5
head	Order	Data 1	Data 2	Data 3	Data 4	tail

Mode=00 scheduled transmission

Mode=01 master-slave query

For example: (PC -> Joystick)

Master mode	af 08 00 00 00 00 f5	(HEX) scheduled sending
Slave mode	af 08 00 00 00 01 f5	(HEX) Query

8. Query device information: (PC -> Joystick)

af 20 00 00 00 00 f5 (HEX)

Joystick reply format: Joystick->PC

FF	60	18	18 01 20	53 4D 43	31	38	30	CH	
Header	Model	Version	Date	S	M	C	1	8	0

Checksum Checksum = Add all the bytes except the header (FF) and

take the low byte of the result

RS485 communication parameter settings

9. Set RS485 baud rate (PC -> joystick)

RS485 baud rate is the same, the settings are valid at the same time

0xaf	0x0b	00	00	00	Baud	0xf5
head	Command		Data2	Data3	Data4	tail
	data1					
	Baud=0X00	Baud rate = 9600				
	Baud=0X01	Baud rate = 19200				
	Baud=0X02	Baud rate = 57600				
	Baud=0X03	Baud rate = 115200				

For example:

Set9600	af 0b 00 00 00 00 f5	(HEX)
Set19200	af 0b 00 00 00 01 f5	(HEX)
Set57600	af 0b 00 00 00 02 f5	(HEX)
Set115200	af 0b 00 00 00 03 f5	(HEX)

The joystick receives this command, executes it, and responds with ACK

10. Check the joystick position (PC -> joystick)

This command is only valid when the "master-slave query" is in trust mode.

When there is no query command, the joystick will not output any data. The joystick will respond once the query is made.

0xaf	0x07	00	00	00	Addr	0xf5
head	Order	Data1	Data2	Data3	Data4	tail

- ◆ AddrAddress = 0x01-0x7f When the address is correct, send back
- ◆ The joystick will send back the current position after receiving this data. The joystick will send back the current position once it checks the data. If it does not check the data, no data will be sent.
- ◆ For example, when querying during RS232 communication:
 (PC->joystick) af 07 00 00 00 01 f5 (HEX)
 (Joystick->PC) FF 01 08 00 70 00 00 00 00 79
 The joystick receives this data and sends back the current position

3. Parameter setting of CAN communication:

CAN parameter settings also need to be done through the RS485 port;

11. CAN port baud rate: (PC -> joystick)

```
0xaf 0x06 XX 00 00 00 0xf5
head Order Data1 Data2 Data3 Data4 tail
XX=00 125K
```

XX=01 250K (default)

XX=02 500K

XX=03 1000K

XX=04 100K

```
eg: 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
     af 06 04 00 00 00 f5 (HEX) CAN baud rate = 100K
```

12. CAN protocol settings: (PC -> Joystick)

```
0xaf 0x0a 00 Pro 00 00 0xf5
head Command Data2 data3 data4 tails
      data1
```

Pro: Protocol Format

```
af 0a 00 00 00 00 f5 (HEX) Manufacturer protocol format (default)
```

13. Joystick "Send Node ID" setting: (PC -> Joystick)

Joystick CAN ID, CAN ID for sending data frames (Joystick CAN ID)

Only applicable to "normal protocol," CANopenThe protocol does not use this instruction

```
0xaf 0x01 D1 D2 D3 D4 0xf5
Header command
data1 data2 data3 data4 tail
```

D1.7=0 extended frame 29 Bit

D1.7=1 standard frame 11 bits

- 29-bit extended frame: data range 0X0-0X0FFFFFFF, data D1-D4 corresponds to the "node identification code". For example: Set the sending node identification code - extended frame "0X0cfa0001"

```
af 01 0c fa 00 01 f5 (HEX)
```

- 11-bit standard frame: data range 0X000-0X3FF, data D3-D4 corresponds to the "node identification code". For example: Set the sending node identification code

- standard frame "0X181"
af 01 80 00 01 81 f5 (HEX)

14. Joystick "Receive Node ID" setting: (PC -> Joystick)

CAN ID frames that the joystick can receive , suitable for customized protocols.

0xaf	0x02	D1	D2	D3	D4	0xf5
header	command	data1	data2	data3	data4	tails

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 corresponds to the "node identification code". For example: Set the receiving node identification code - extended frame "0X00F0F101"

af 02 00 f0 f1 01 f5 (HEX)

- 11-bit standard frame: data range 0X000-0X3FF, data D3-D4 corresponds to the "node identification code". For example: Set the receiving node identification code - standard frame "0X1E1"

af 02 80 00 01 E1 f5 (HEX)

15. Joystick "Shield Node ID" setting: (PC-> Joystick)

only applicable to customized protocol products

0xaf	0x03	D1	D2	D3	D4	0xf5
header	command		data1	data2	data3	data4
						tails

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 corresponds to the "node identification code". For example: Set the shielded node identification code - extended frame "0X00002201"

af 03 00 00 22 01 f5 (HEX)

- 11-bit standard frame: data range 0X000-0X3FF, data D3-D4 corresponds to the "node identification code". For example: Set the shielded node identification code - standard frame "0X122"

af 03 80 00 01 22 f5 (HEX)

4. Modbus Parameter settings for RTU (RS485) communication:

(Modified on January 19, 2018)

16. Set "Device Address": Factory default 1

0xaf	0x0d	00	00	00	Add	0xf5
head	Command		Data2	Data3	Data4	tail
		data1				

Add=0x01~0x7F address 1-127 address = 0

When the joystick receives this command, it will

respond after execution.ACK

Setaddress1 af 0d 00 00 00 01 f5 (HEX)

Setaddress2 af 0d 00 00 00 02 f5 (HEX)

Setaddress3 af 0d 00 00 00 03 f5 (HEX)

17. Joystick working mode (PC->joystick):

Master mode af 08 00 00 00 00 f5 (HEX) Scheduled transmission

Slave mode af 08 00 00 00 01 f5 (HEX) Query

18. Set register address (PC->joystick):

Data format:

0xaf	0x18	D1	D2	D3	D4	0xf5
head	Order	Data1	Data 2	00	00	tail

D1: register address

high bit D2: register

address low bit

Factory default register

address = 0x0001 is

permanently stored in the

joystick after setting

For example: Set the register address = 0x0001 (hexadecimal), if octal needs to be converted to hexadecimal .

Register address = af 18 12 34 00 00 f5
0x1234

Register address = af 18 00 01 00 00 f5 (PLC: 40001+1=40002)
0x0001

Register address = af 18 00 02 00 00 f5 (PLC: 40001+2=40003)
0x0002

Register address = af 18 00 03 00 00 f5 (PLC: 40001 + 3 = 40004)
0x0003

Register address = af 18 00 08 00 00 f5 (PLC: 40001 + 8 = 40009)
0x0008