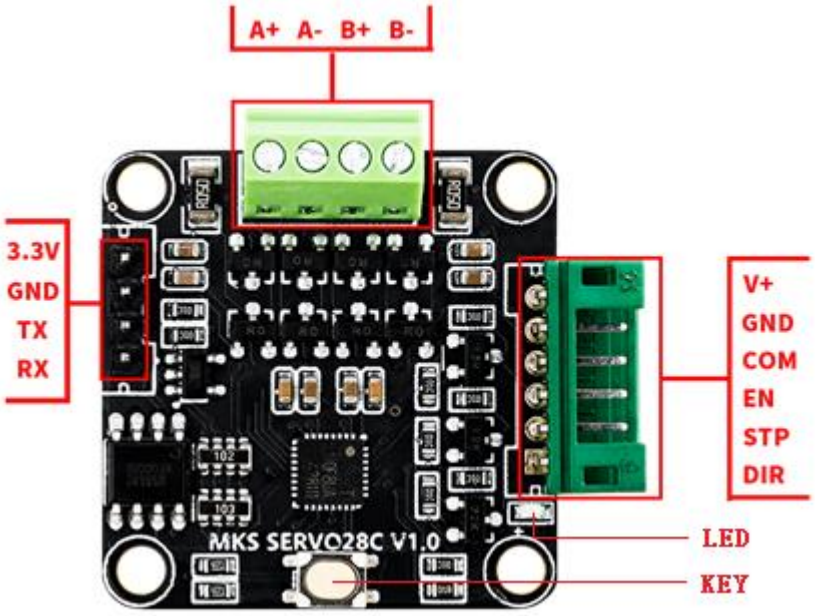
MKS SERVO28C USER MANUAL

V1.1

Part1. Feature

1.1 Interface



NOTE1: Power(V+,GND) input voltage range is 12V-24V.

NOTE2: The input signal (EN,STP,DIR) voltage range is 3.3V-24V.

NOTE3: The signal (COM) no connect.

1.2 Key operation

Operation	Function		LED status	Serial output
Press 2 times quickly	Restore default parameters		LED twinkle	reboot
Press about 3s until the LED on	Calibrate the motor	Calibrating	LED ON	Cal…
		Calibration complete	LED OFF	Cal Done

Note 1： The serial parameters: baudrate 38400, 8bit data, no check, 1 bit stop.

Note 2： After restore the default parameters, please restart the motor and calibrate it again.

1.3 Status description

Status	LED Status	Serial output
Power on	Turn on about 1s	mks Servo28c running... motoType=xx (xx=1.8 or 0.9)
Not Cal	Twinkle 2 tims every 2s	Not Cal
Phase wire error	Twinkle 3 tims every 2s	Phase Line Error
Motor type error	Turn on	Motor Type Error
Magnet error	Turn on	Magnet Error
Encoder error	Turn on	Encoder Error
Low power	Turn on	Waiting Power
Rerence voltage error	Turn on	Offset Error
Calibration err1	Turn on	dx=0 Error
Calibration err2	Turn on	Reverse Error
Stall protected	Twinkle	Protect Error

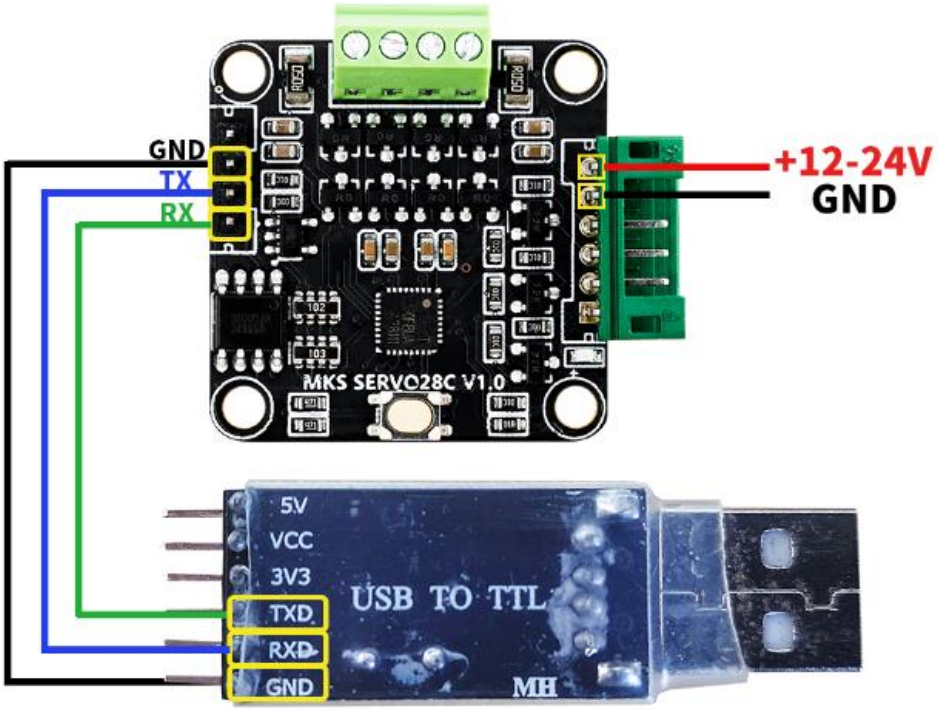
1.4 Work mode

Work mode	Signal input	
CR_OPEN	EN,STP,DIR	
CR_vFOC	EN,STP,DIR	default
CR_UART	Serial	

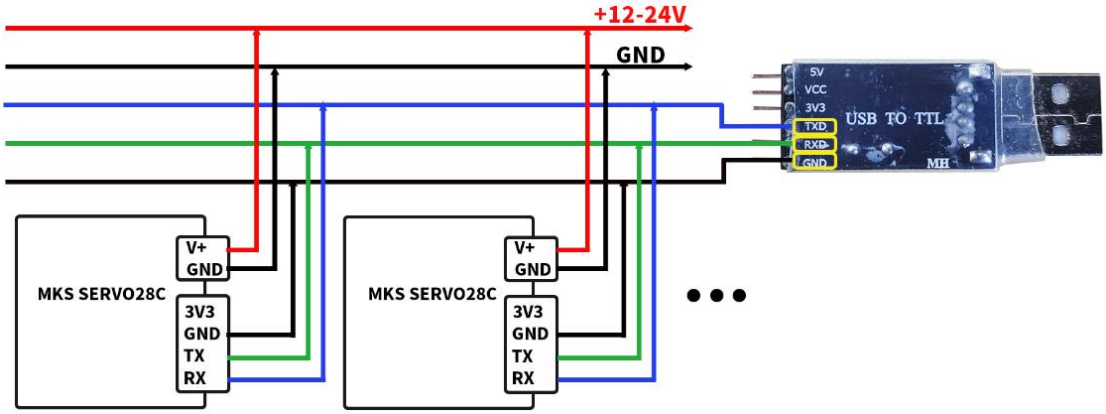
Note: The work mode can be configured by Command10 : e0 82 xx CRC

Part2. Serial Wiring

2.1 Single serial connection



2.2 Multiple serial connection



Part3. Serial data format

Downlink package (PC➔SERV028C)			
Slave address	Function	data	check code
addr	code	…	CRC
Uplink package (PC➔SERV028C)			
Slave address		data	check code
addr		…	CRC

1. The slave address (addr) ranges from e0 to e9, default is e0.
2. The function (code) executes the corresponding command,
   for example, code 0x80 executes the calibrate command.
3. check code is checksum 8bit.
   for example, package “e0 30 CRC”
   CRC = (0xe0 + 0x30) & 0xFF = 0x110 & 0xFF = 0x10

Part4. Serial command description

4.1 Read parameter command

Note: Please set the serial slave address first.(default:e0)

The default address for the following chapters is e0.

1. command1 : e0 30 CRC

   read the encoder value (the motor should be calibrated).
   return e0 and the value(Vh(int16_t),Vl(uint16_t)) of encoder and CRC.

   Vh: the carry vaule of the encoder,range(-32767~32767)

   Vl: the current vaule of the encoder,range(0~65535)

   Note:The encoder value is updated regardless of whether the motor is enabled or not.

2. command2 : e0 31 CRC

   read the firmware version。
   Return e0, version(uint8_t),and CRC.
   version (0x11) V1.1
   version (0x12) V1.2
   …

3. command3 : e0 33 CRC

Read the number of pulses received

Return e0 and the number of pulses(int32_t) and CRC.

Example:

Send e0 33 13

return e0 00 00 01 00 e1 (256 pulses)

4. command4 : e0 36 CRC

read the angle of the motor shaft.

Return e0 and the angle(int32_t) and CRC.

The motor rotates one circle, the corresponding angle value range is 0~65535.

Example:

Send e0 36 16

return e0 00 00 40 00 20 (angle 90 °)

5. command5: e0 39 CRC

read the error of the motor shaft angle.

Return e0 and the error(int16_t) and CRC.

The error is the difference between the angle you want to control minus the real-time angle of the motor, 0~FFFF corresponds to 0~360° , for example, when the angle error is 1° , the return error is 65536/360= 182.444, and so on.

Example:

Send e0 39 19

return e0 00 B7 97 (error 1 °)

6. command6: e0 3a CRC

read the En pins status.

Return e0 and the status (uint8_t) and CRC.

Status: 01-enable;02-disable;00-error

Example:

Send e0 3a 1a

return e0 01 e1 (enable)

7. command7: e0 3e CRC

read the motor shaft status.

Return e0 and the status (uint8_t) and CRC.

Status: 01-blocked; 02- unblocked; 00-error

Example:

Send e0 3e 1e

return e0 02 e2 (unblocked)

4.2 Set parameters command

1. Calibrate the encoder

Command8 : e0 80 00 CRC

(The motor must be unloaded.)

Return e0 and the result (uint8_t) and CRC.
result: 01-successful; 00-Failure
Example:
Send e0 80 00 60
return e0 01 e1 (successful)

2. Set the motor type

Command9 : e0 81 xx CRC

00 0.9 degree motor
01 1.8 degree motor
Return e0 and the result (uint8_t) and CRC.
result: 01-successful; 00-Failure
Example:
Send e0 81 01 62 (1.8 degree motor)
return e0 01 e1 (successful)

3. Set the work mode

Command10 : e0 82 xx CRC

00 CR_OPEN (open loop mode)
01 CR_vFOC (Closed loop mode (STP/DIR interface))
02 CR_UART (Closed loop mode (UART interface))
Return e0 and the result (uint8_t) and CRC.
result: 01- successful; 00- Failure
Example:

Send	e0 82 01 63	(CR_vFOC)
return	e0 01 e1	(successful)

4. Set the current gear

Command11 : e0 83 xx CRC

00 0ma
01 200ma
02 400ma
…
09 1800ma
0A 2000ma
Return e0 and the result (uint8_t) and CRC.
result: 01- successful; 00- Failure
Example:

Send	e0 83 06 69	(1200ma)
return	e0 01 e1	(successful)

5. Set subdivision

Command12 : e0 84 xx CRC
Set subdivision from 1 to 256, the value will show in the screen of MStep option.
For example:
Send “e0 84 07 6b”, set 7 subdivision
Send “e0 84 4e b2”, set 78 subdivision
Send “e0 84 00 64”, set 256 subdivision
And so on...
Return e0 and the result (uint8_t) and CRC.
result: 01- successful; 00- Failure
Example:

Send	e0 84 1a 7e	(26 subdivision)
return	e0 01 e1	(successful)

6. Set the active of the En pin

Command13 : e0 85 xx CRC

00 active low (L)
01 active high (H)
02 active always (Hold)
Return e0 and the result (uint8_t) and CRC.
result: 01-successful; 00-Failure
Example:

Send	e0 85 00 65	(active low)
return	e0 01 e1	(successful)

7. Set the direction of motor rotation

Command14 : e0 86 xx CRC

00 CW
01 CCW
Return e0 and the result (uint8_t) and CRC.
result: 01-successful; 00-Failure
Example:

Send	e0 86 00 66	(CW)
return	e0 01 e1	(successful)

8. Set the stall protection

Command15 : e0 88 xx CRC

00 Disable
01 Enable
Return e0 and the result (uint8_t) and CRC
result: 01-successful; 00-Failure
Example:

Send	e0 88 00 68	(Disable)
return	e0 01 e1	(successful)

9. Set the subdivision interpolation function

Command16 : e0 89 xx CRC

00 Disable
01 Enable
Return e0 and the result (uint8_t) and CRC.
result: 01-successful; 00-Failure
Example:

Send	e0 89 00 69	(Disable)
return	e0 01 e1	(successful)

10. Set the baud rate

Command17 : e0 8a xx CRC

01 9600
02 19200
03 25000
04 38400
05 57600
06 115200
Return e0 and the result (uint8_t) and CRC.
result: 01-successful; 00-Failure
Example:

Send	e0 8a 04 6e	(38400)
return	e0 01 e1	(successful)

11. Set the slave address

Command18 : e0 8b xx CRC

00 e0
01 e1
02 e2
…
09 e9
Return e0 and the result (uint8_t) and CRC.
result: 01-successful; 00-Failure
Example:

Send	e0 8b 02 6d	(e2)
return	e0 01 e1	(successful)

4.3 Set zeroMode parameters command

1. Set the mode of zeroMode

Command19 : e0 90 xx CRC
00 Disable
01 DirMode
02 NearMode
Return e0 and the result (uint8_t) and CRC.
result: 01- successful; 00- Failure

Example:

Send	e0 90 01 71	(DirMode)
return	e0 01 e1	(successful)

2. Set the zero of zeroMode

Command20 : e0 91 xx CRC
Return e0 and the result (uint8_t) and CRC.
result: 01- successful; 00- Failure
Example:
Send e0 91 00 71 (DirMode)
return e0 01 e1 (successful)
Note: The mode of "0_Mode" needs to be set first.

3. Set the speed of zeroMode

Command21 : e0 92 xx CRC
(Range 0~4, the smaller the value, the faster the speed)
Return e0 and the result (uint8_t) and CRC.
result: 01- successful; 00- Failure
Example:
Send e0 92 02 74
return e0 01 e1 (successful)
Note: The mode of "0_Mode" needs to be set first.

4. Set the dir of zeroMode

Command22 : e0 93 xx CRC
00 CW
01 CCW
Return e0 and the result (uint8_t) and CRC.
result: 01- successful; 00- Failure
Example:
Send e0 93 00 73 (CW)
return e0 01 e1 (successful)
Note: For NearMode, the setting of 0_Dir should be consistent with the actual running direction of the motor, otherwise it will fail to return to zero.

5. Return to zero

Command23 : e0 94 00 CRC
Return e0 and the result (uint8_t) and CRC.
result: 01- successful; 00- Failure
Example:
Send e0 94 00 74
return e0 01 e1 (successful)

4.4 Set PID/ACC/Torque command

1. Set the position Kp parameter

command24: e0 a1 XX XX CRC

(Default 0x650).
Return e0 and the result (uint8_t) and CRC.
result: 01- successful; 00- Failure
Example:

Send	e0 a1 01 20 a2	(Kp = 0x120)
return	e0 01 e1	(successful)

2. Set the position Ki parameter

command25: e0 a2 XX XX CRC

(Default 1).
Return e0 and the result (uint8_t) and CRC.
result: 01- successful; 00- Failure
Example:

Send	e0 a2 00 02 84	(Ki = 0x02)
return	e0 01 e1	(successful)

3. Set the position Kd parameter

command26: e0 a3 XX XX CRC

(Default 0x650).
Return e0 and the result (uint8_t) and CRC.
result: 01- successful; 00- Failure
Example:

Send	e0 a3 02 50 d5	(Kd = 0x250)
return	e0 01 e1	(successful)

4. Set the acceleration (ACC) parameter

command27: e0 a4 XX XX CRC

(Default 0x11e).
Return e0 and the result (uint8_t) and CRC.
result: 01- successful; 00- Failure
Example:

Send	e0 a4 00 80 04	(ACC = 0x80)
return	e0 01 e1	(successful)

Important note: If the ACC parameter is set too large, it may damage the driver board, please set the parameters carefully!

5. Set the maximum torque (MaxT) parameter

command28: e0 a5 XX XX CRC

MaxT value range (0 ~ 0x4B0). (Default 0x4B0)
Return e0 and the result (uint8_t) and CRC.
result: 01- successful; 00- Failure
Example:
Send e0 a5 02 58 df (MaxT = 0x80)
return e0 01 e1 (successful)

4.5 Restore default parameter command

1. command29: e0 3f CRC

Return e0 and the result (uint8_t) and CRC.
result: 01- successful; 00- Failure
Example:
Send e0 3f 1f
return e0 01 e1 (successful)
Note: After restoring the default parameters, you need to power on again and reset the serial port baud rate before serial communication.

4.6 Serial control command

1. command30: e0 f3 xx CRC

Set the En pin status in CR_UART mode.

status: 01-enable; 00-disable

Return e0 and the result (uint8_t) and CRC.

result: 01-successful; 00-Failure

Example:

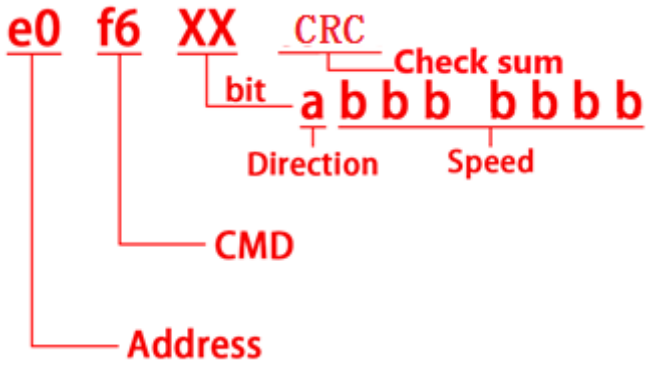
Send e0 f3 01 d4 (enable)

return e0 01 e1 (successful)

2. command31: e0 f6 xx CRC

run the motor forward / reverse in a Constant speed.

The XX highest bit indicates direction, and the XX lowest 7 bits indicate 128 speed gears, for example:



The speed (Vrpm) calculation formula is:

Vrpm = (Speed × 30000)/(Mstep × 200)(RPM) for 1.8 degree motor

Vrpm = (Speed × 30000)/(Mstep × 400)(RPM) for 0.9 degree motor

(Mstep is set by command8 or by the screen menu)

For example:

1.8 degree motor, Mstep = 150, Speed=1

Vrpm = (1 × 30000)/(150 × 200) = 1

Part of the speed is as follows:

1.8 degree motor	Vrpm (RPM)	
speed	Mstep=16	Mstep=150
1	9.375	1
2	18.75	2
3	28.125	3
4	37.5	4
…	…	…
8	75	8
…	…	…
0x10	150	16
0x20	300	32
0x40	600	64
0x7F	1200	127

Such as:

Send “e0 f6 01 d7”, The motor rotates forward at the speed of 1;

Send “e0 f6 81 57”, The motor reverses at the speed of 1;

Send “e0 f6 5a 30”, The motor rotates forward at the speed of 90;

Send “e0 f6 da b0”, The motor reverses at the speed of 90;

Return e0 and the result (uint8_t) and CRC.

result: 01-successful; 00-Failure

Example:

Send	e0 f6 10 e6	(speed of 16)
return	e0 01 e1	(successful)

3. command32: e0 f7 CRC

stop the motor.

Return e0 and the result (uint8_t) and CRC.

result: 01-successful; 00-Failure

Example:

Send	e0 f7 d7	
return	e0 01 e1	(successful)

4. command33: e0 ff xx CRC

Save/Clear the status set in command9.

xx: c8-save; ca-clean

Return e0 and the result (uint8_t) and CRC.

result: 01-successful; 00-Failure

Example:

Send	e0 ff c8 a7	(save the status)
return	e0 01 e1	(successful)

Attention1: The motor can rotates clockwise or counterclockwise at a constant speed when powered on.

Attention2: After the save is successful, the driver board will be disable, and it needs to be re-enabled.

For example:

Step1: Send e0 f6 10 e6

return e0 01 e1

Step2: Send e0 ff c8 a7

return e0 01 e1

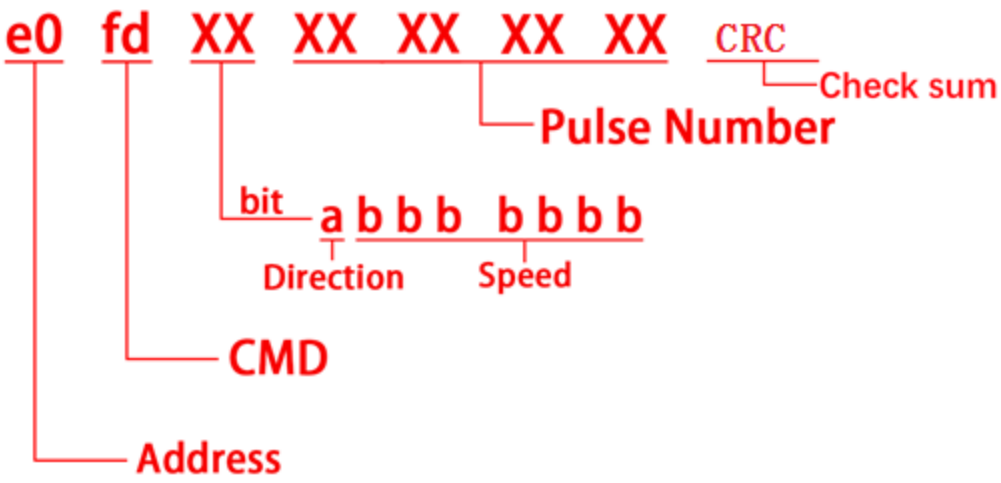
the motor will always rotates clockwise in speed(16) when power on.

5. command34: e0 fd XX XX XX XX XX CRC

The motor run to an angle in a speed.

The third byte(XX) defines the direction and speed.

The last four bytes(XX XX XX XX) define the number of pulses.if(XX XX XX XX = 00 00 0C 80),the motor(1.8 ° , subdivision 16) will rotates one circle.



The speed (Vrpm) calculation formula is the same as command31.

Return e0 and the result (uint8_t) and CRC.

result: 01-successful; 00-Failure

For example: (the motor(1.8 ° , subdivision 16))

send “e0 fd 01 00 00 0c 80 6a”,the motor rotates forward 360 ° in speed 1.

send “e0 fd 86 00 00 0c 80 ef”, the motor rotates reverses 360 ° in speed 6.

Example:

Send e0 fd 32 00 00 0c 80 9b

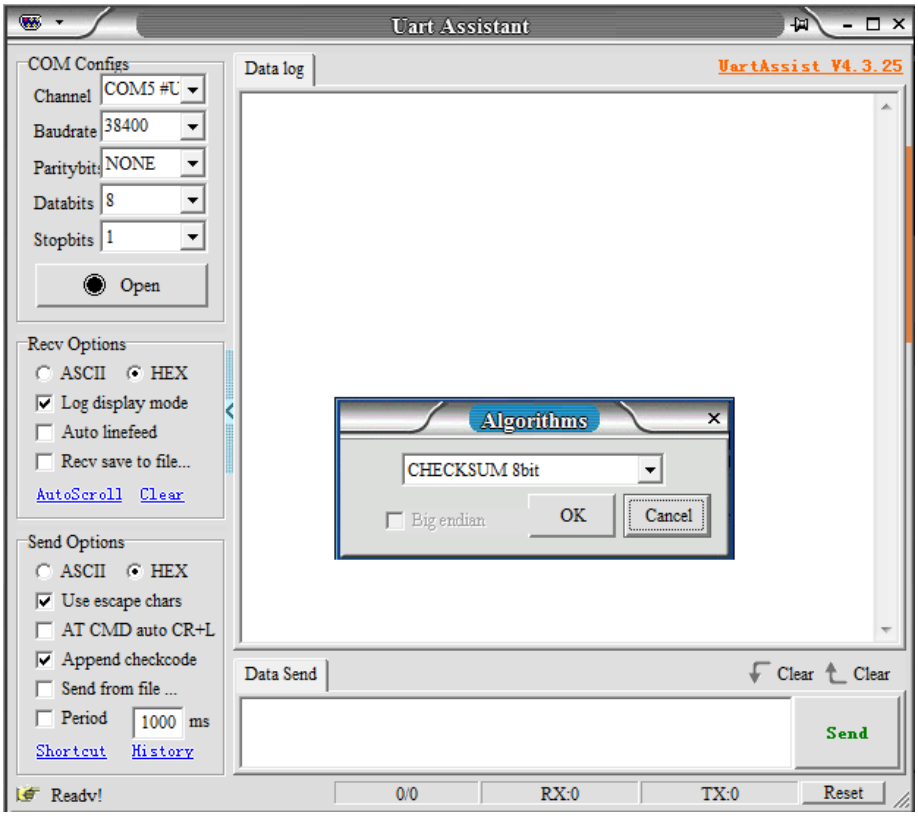
return e0 01 e1 (successful)

Part5. Serial example

5.1 Config the Uart Assistant

1. Select the Channel; (such as COM5).
2. Select the Baudrate; (such as 38400, Must be equal to motor baudrate).
3. Recv Options: select “HEX”.
4. Send Options: select “HEX”.
5. Append checkcode: select “CHECKSUM-8”.

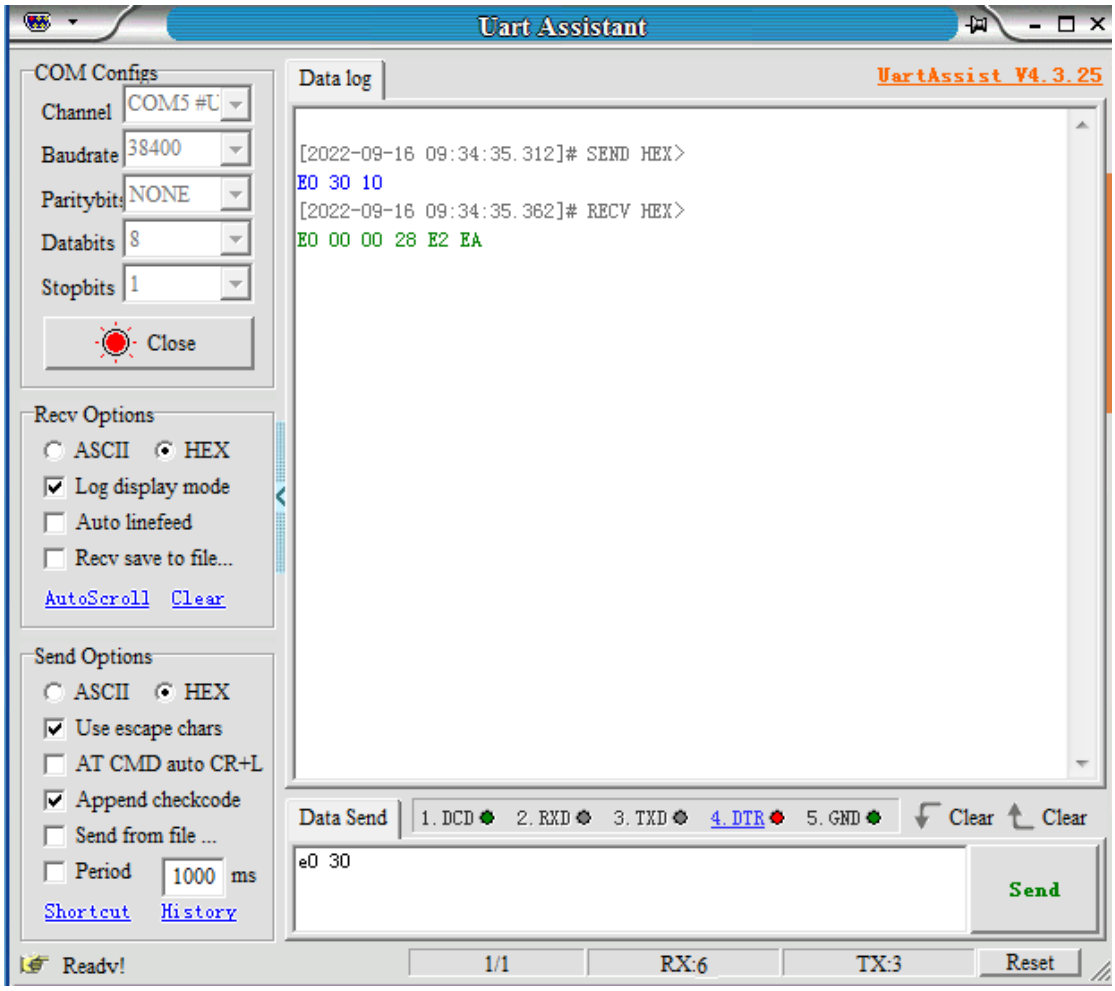
Such as below:



5.2 Read the encoder value

send “e0 30”

return “E0 00 00 28 E2 EA”



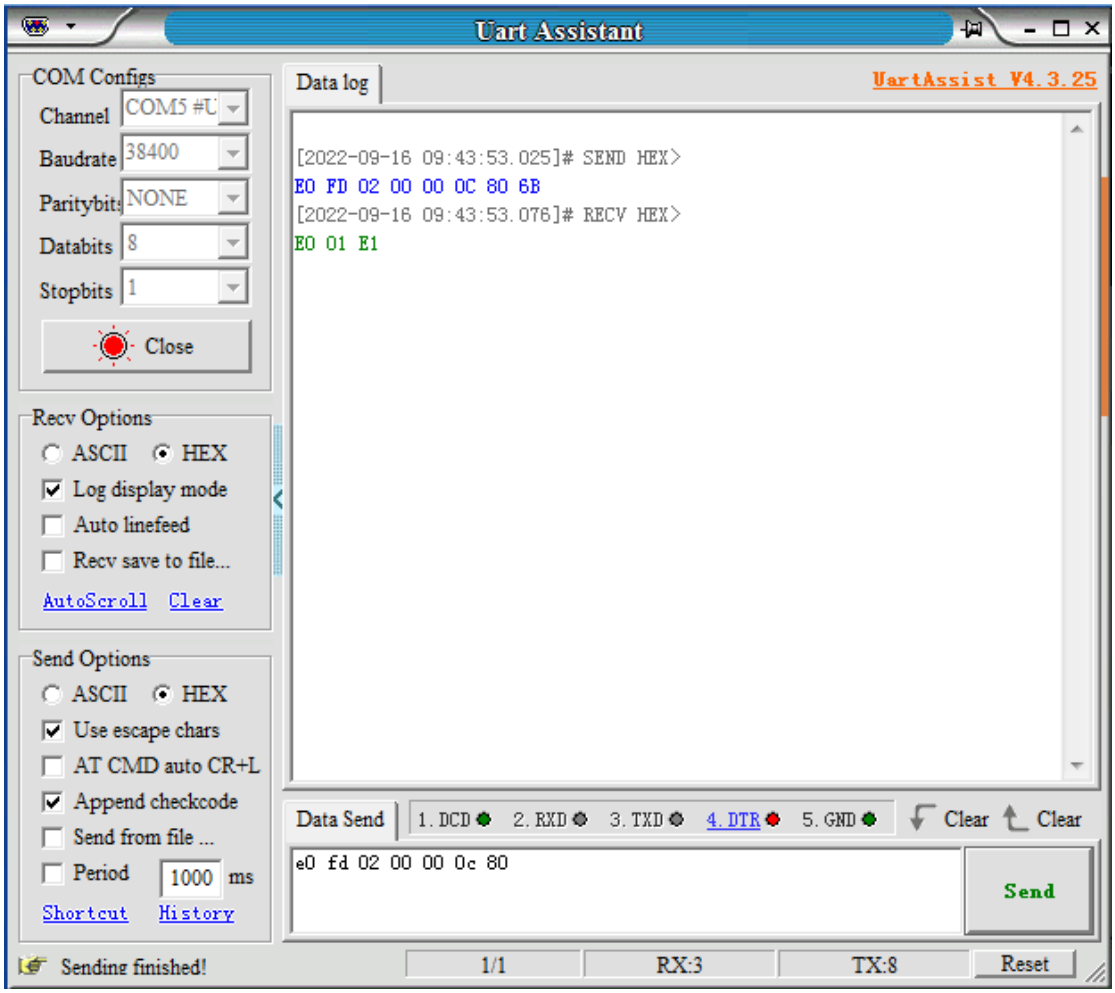
5.3 Run the motor

Note : Please configure the working mode to “CR_UART”.

Send “e0 fd 02 00 00 0c 80”

Return “E0 01 E1”

The motor will run one revolution at speed 2. (subdivision 16)



Part6. FAQ

6.1 NOTE

1. Power input voltage is 12V-24V.
2. Don’t hot plug motor cable and data cable.
3. When the motor is calibrating, do not carry a load.
4. After installed the motor, or changed the motor wiring sequence, you need to re-calibrate the motor again.
5. The default work mode is CR_vFOC(EN/STP/DIR interface).

6.2 FAQ

No	Serial output	Solution
1	Not Cal	Calibrate the motor.
2	Phase Line Error	Swap motor line sequence.
3	Motor Type Error	Reset to 1.8 degrees or 0.9 degrees
4	Magnet Error	Check if the magnet is fixed
5	Encoder Error	Check if the magnet is fixed
6	Waiting Power	Check the supply voltage
7	Offset Error	Check the supply voltage
9	dx=0 Error	Check magnet and motor shaft
10	Reverse Error	Check magnet and motor shaft