



M4313SXX User's Manual

This document is the User's Manual for M4313SXX, the force/torque sensor (loadcell) manufactured by SRI (Sunrise Instruments Co., Ltd). It's strongly recommended that anyone who uses M4313SXX should read this document before any operation. SRI reserves all the rights of this document. Please do not hesitate to contact SRI if there is any question.

Content

1. INTRODUCTION.....	3
2. INSTALLATION.....	5
2.1 STANDARD INSTALLATION	5
2.2 INSTALLATION METHOD FOR M4313SFX	9
2.3 INSTALLATION METHOD FOR M4313S1X.....	12
3. COMMUNICATION	13
3.1 RS232	13
3.1.1 BY IDAS R&D	13
3.1.2 BY CUSTOMER SOFTWARE.....	15
3.2 RS485	16
3.2.1 BY IDAS R&D	17
3.2.2 BY CUSTOMER SOFTWARE.....	18
3.3 ETHERCAT	20
3.3.1 BY IDAS R&D	22
3.3.2 BY TWINCAT3.....	23
3.4 ETHERNET TCP/IP	24
3.4.1 BY IDAS R&D	25
3.4.2 BY CUSTOMER SOFTWARE.....	26
3.5 CAN.....	27
4. COMMANDS.....	29
4.1 COMMANDS TO CONFIGURE RS232/ETHERNET/CAN	31
4.1.1 UARTCFG / COMMUNICATION PARAMETERS FOR RS232	31
4.1.2 EIP / ETHERNET IP ADDRESS.....	32
4.1.3 EMAC / ETHERNET MAC.....	32
4.1.4 EGW / ETHERNET GATEWAY ADDRESS	33
4.1.5 ENM / ETHERNET NETMASK.....	33
4.1.6 CIDT / ID TYPE FOR CAN BUS	34
4.1.7 CFIDL / ID OF CAN BUS	34
4.1.8 CRATE / BAUD RATE OF CAN BUS.....	35
4.1.9 CFI / INTERVAL TIME BETWEEN FRAMES OF CAN BUS	36
4.2 SFWV / FIRMWARE VERSION	36
4.3 DCPM / READ OR SET DECOUPLED MATRIX	37
4.4 DCPCU / CALCULATION UNIT FOR DECOUPLED DATA	37
4.5 SMPF / READ OR SET SAMPLING RATE.....	38
4.6 DCKMD / SET DATA-CHECK METHOD	38
4.7 GOD / GET ONE PACKAGE DATA FROM M4313SXX.....	39
4.8 GSD / GET DATA CONTINUOUSLY	39
4.9 ADJZF/ ADJUST ZERO OFFSET OF SENSOR	41

1. Introduction

The M4313SXX series is an industrial-grade high-precision 6-axis force/torque sensor with excellent anti-shock performance, high stability, and low zero drift. It is widely used in collaborative robots, industrial robots, force control systems, and automation equipment. M4313SXX series sensor supports RS232, RS485, CAN, Ethernet TCP/IP and EtherCAT communication.



Specifications:

- Sensor
 - 6 axis force/torque sensor: FX,FY,FZ,MX,MY,MZ
 - High accuracy: 0.5%F.S
 - Anti-shock: 500%F.S overload capacity
 - High stability: Do not failure after 100 cycles of 5 times overload
 - Zero drift: 0.05%/10°C

- Digital
 - RS232, RS485, CAN, Ethernet TCP/IP and EtherCAT
 - 24 bit sigma-delta ADC (16 bit effective), Sampling rate: 10 ~ 2000 Hz

- Power supply
 - DC 11V to 25V, 2.5W

- Software
 - iDAS RD: Debugging software to display real-time sampling curve.
 - Open communication protocol
 - RS232 and Ethernet TCP sample code

Selection Guide:

Model #: M4313S 2 E 2

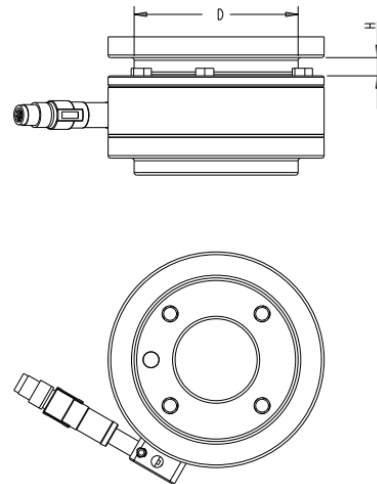
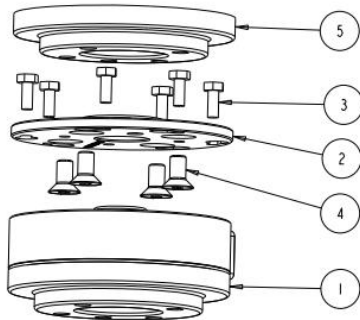
Series			
M4313S			
Outer diameter			
1 = 60mm 2 = 83mm 3 = 91mm			
Communication			
A = RS485 C = EtherCAT E = Ethernet TCP/IP J = CAN			
Capacity			
2 = FX,FY,FZ:100N MX,MY,MZ:10Nm 4 = FX,FY,FZ:200N MX,MY,MZ:20Nm 5 = FX,FY,FZ:250N MX,MY,MZ:25Nm 6 = FX,FY,FZ:300N MX,MY,MZ:30Nm 7 = FX,FY,FZ:400N MX,MY,MZ:40Nm 9 = FX,FY,FZ:600N MX,MY,MZ:60Nm B = FX,FY,FZ:800N MX,MY,MZ:80Nm			

2. Installation

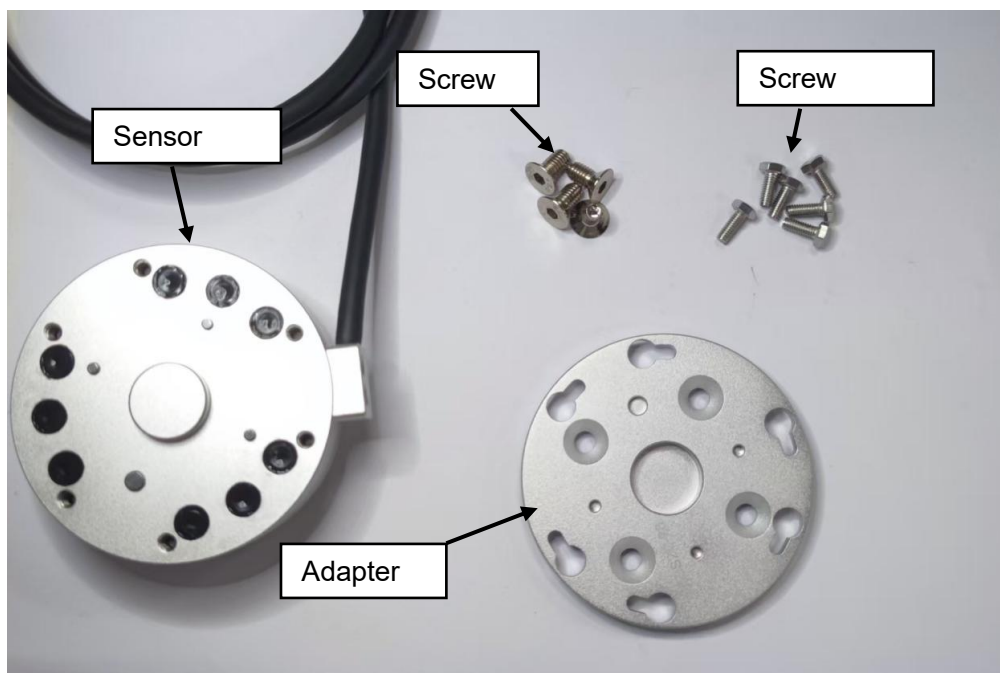
There are three installation methods for M4313SXX series, standard installation method, method for M4313SFx series and method for M4313S1x series.

2.1 Standard Installation

2.1.1 Parts



- Part #1: Sensor
- Part #2: Adapter
- Part #3: Screw
- Part #4: Screw
- Part #5: Robot flange

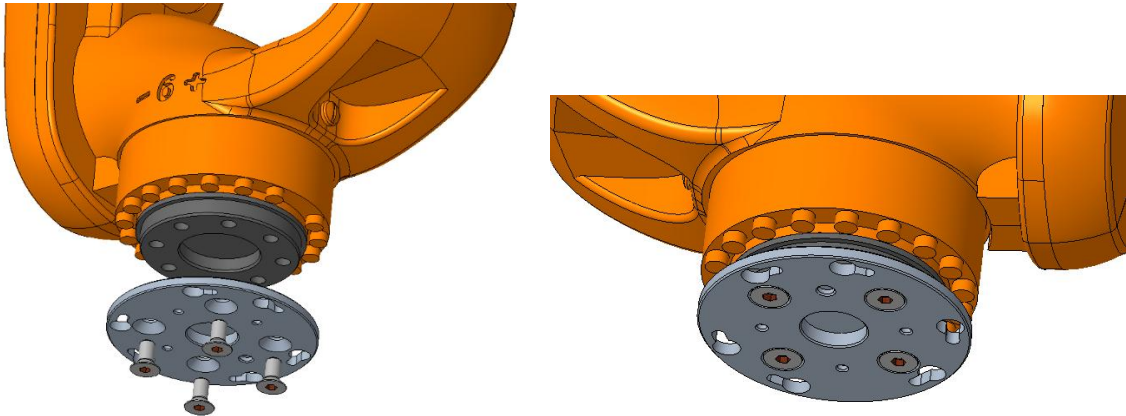


2.1.2 Installation note

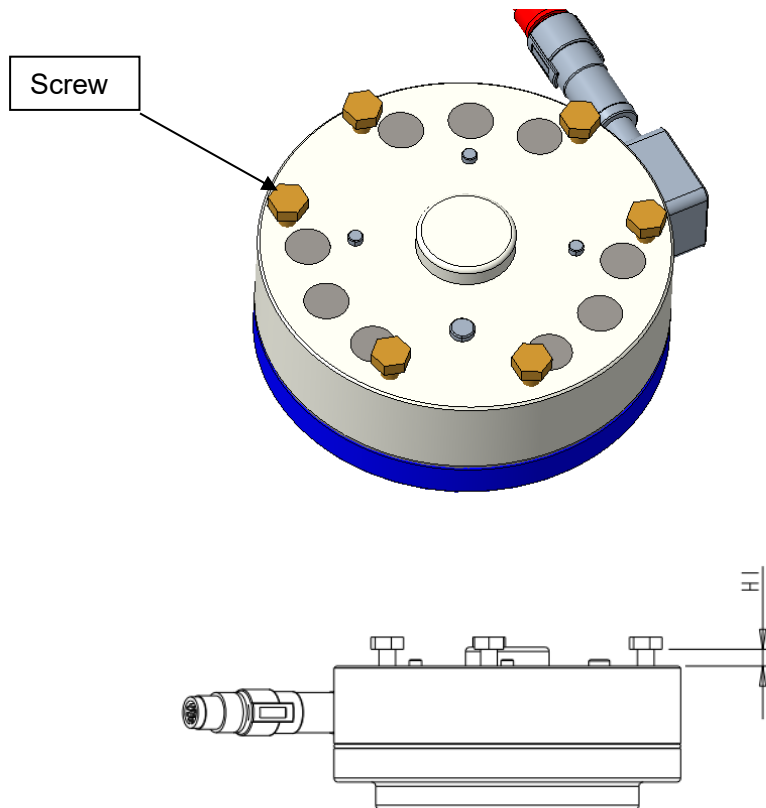
1. If part #2 is compatible with robot flange, please install the sensor to robot by following the steps below
2. If part #2 is not compatible with robot flange, please design a new flange to install the sensor to robot.

2.1.3 Installation steps

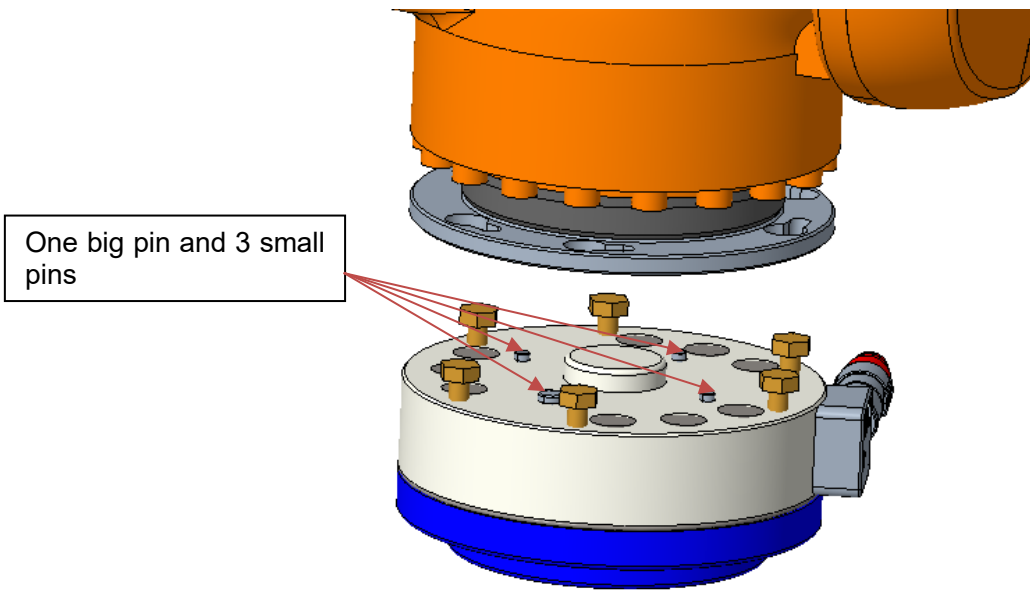
Step1: Installation part #2 to robot flange by 4 screws (FHCS M6).



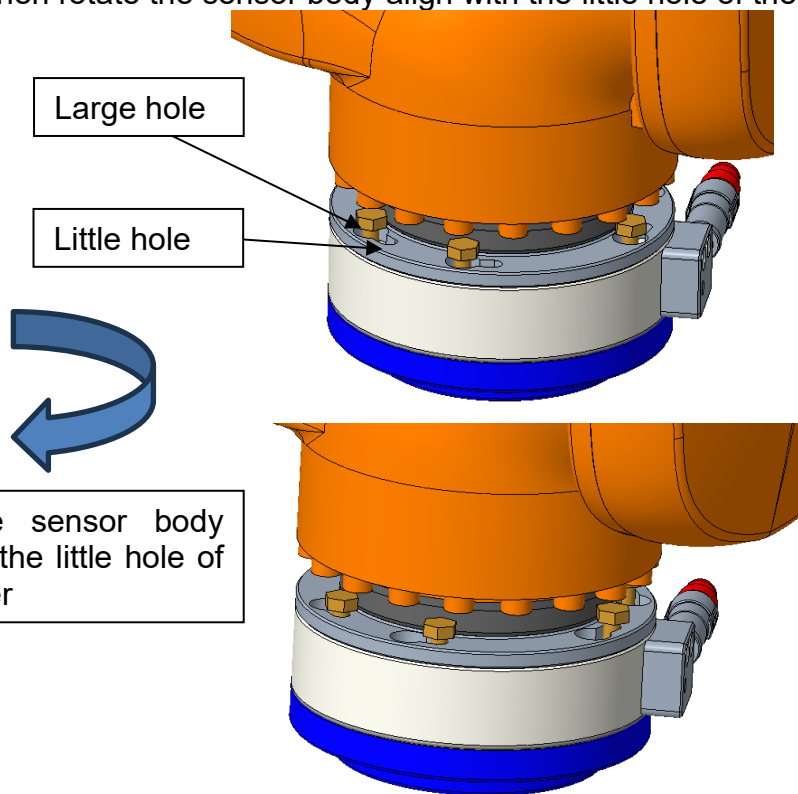
Step2: Tighten crew(part #3) to sensor. As shown in picture below, make the distance H1 2-3mm more than the height of the adapter (part #2).



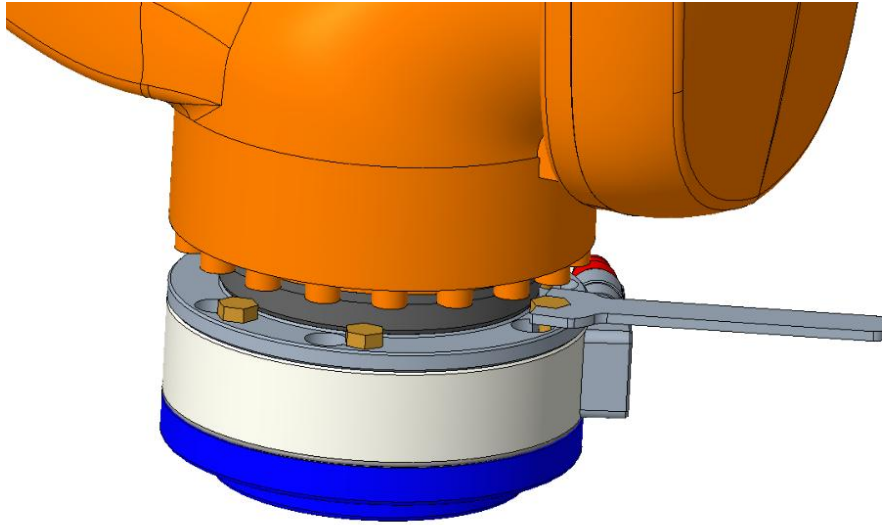
Step3: Make sure the position of pins on sensor are matched with adapter(part #2).



Step4: Pass the screw on the sensor through the large hole of the adapter (part #2), then rotate the sensor body align with the little hole of the adapter.

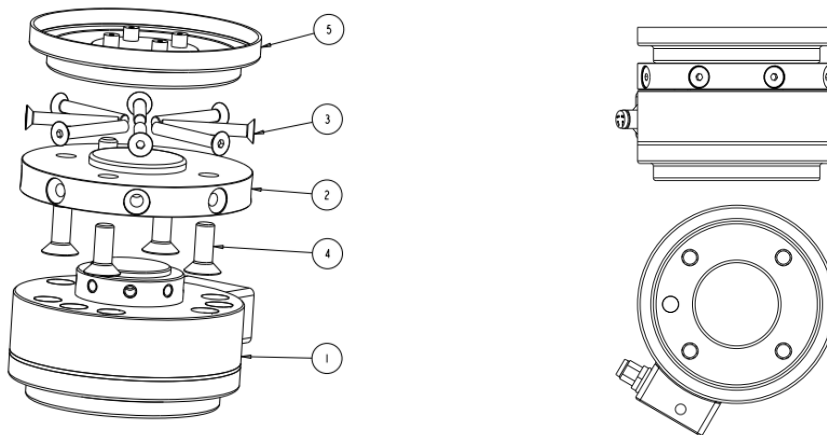


Step5: Tighten the screws by spanner.



2.2 Installation method for M4313SFx

2.2.1 Parts



Part #1: Sensor

Part #2: Adapter

Part #3: M4X30 screw

Part #4: M6 screw

Part #5: Robot flange

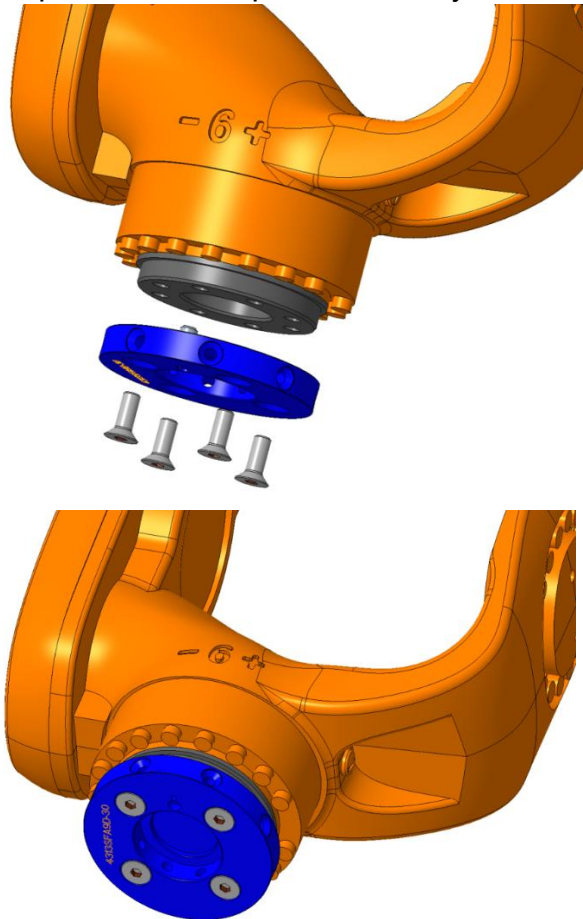
2.2.2 Installation note

If part #2 is compatible with robot flange, please install the sensor to robot by following the steps below

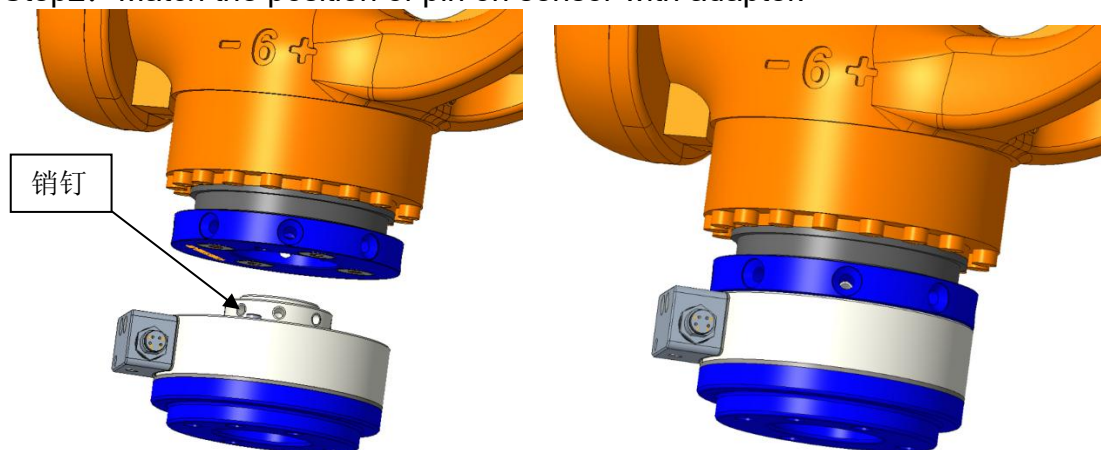
If part #2 is not compatible with robot flange, please design a new flange to install the sensor to robot.

2.2.3 Installation Steps

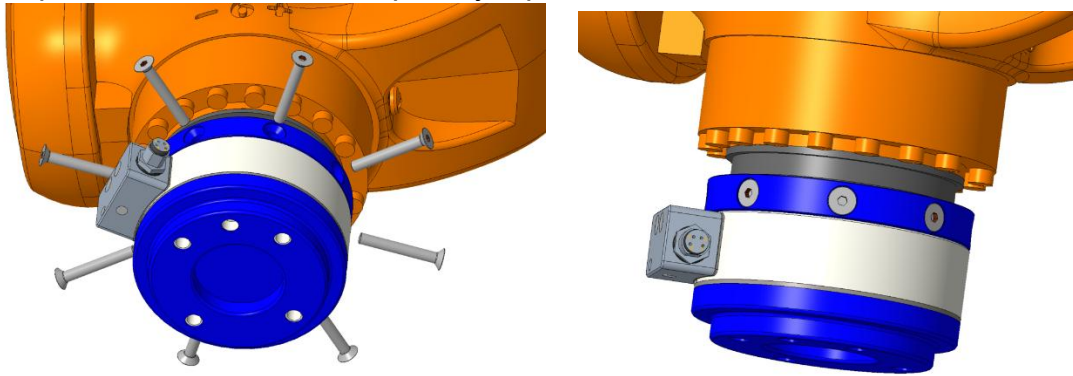
Step1: Install adapter to robot by 4 screws(FHCS M6).



Step2: Match the position of pin on sensor with adapter.



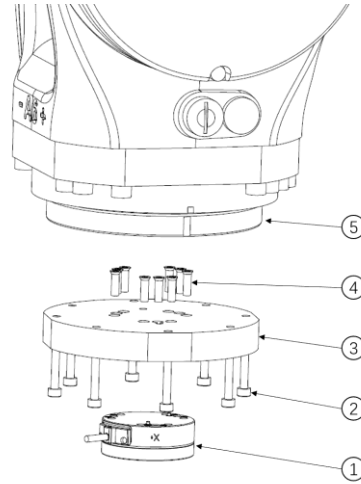
Step3: Install sensor to adapter by 8 pieces of M4X30 screws.



2.3 Installation method for M4313S1x

SRI do not supply the adapter of M4313S1x, customer should design it by themselves

2.3.1 Parts



Part #1: Sensor

Part #2: M10X20 Screw

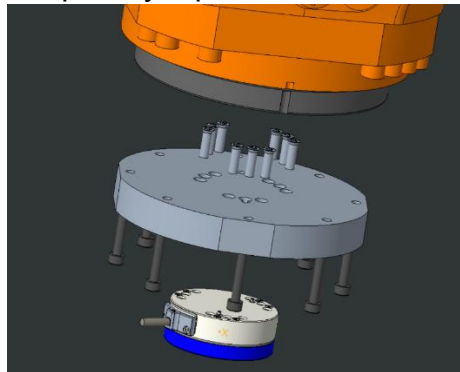
Part #3: Adapter

Part #4: M5X16 Screw

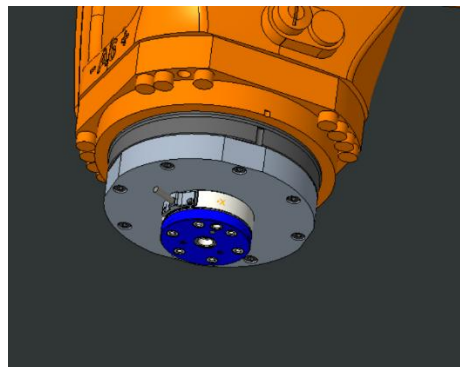
Part #5: Robot flange

2.3.2 Installation steps

Step1: Install sensor to adapter by 9 pieces of M5X16 screws.



Step2: Install the adapter to robot.



3. Communication

M4313SXX supports RS232, RS485, CAN, Ethernet TCP/IP and EtherCAT communication.

3.1 RS232

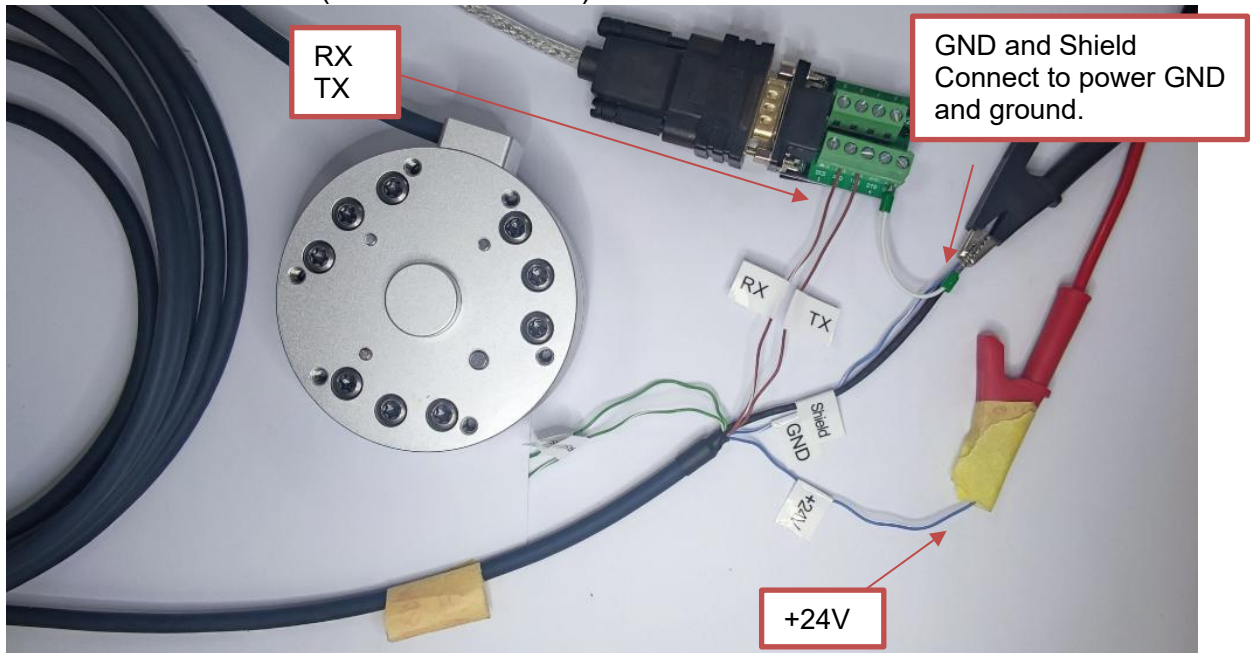
RS232 default configuration:

Baud Rate	Even parity	Stop Bit	Data Bits
115200bps	None	1	8

RS232 wire definition:

Color	Definition	Note
Blue	+24V	
White Blue	GND	
White brown	RX	
Brown	TX	
Shield line	Shield	To reduce noise, it is recommended to connect the shield to the true ground in your lab.

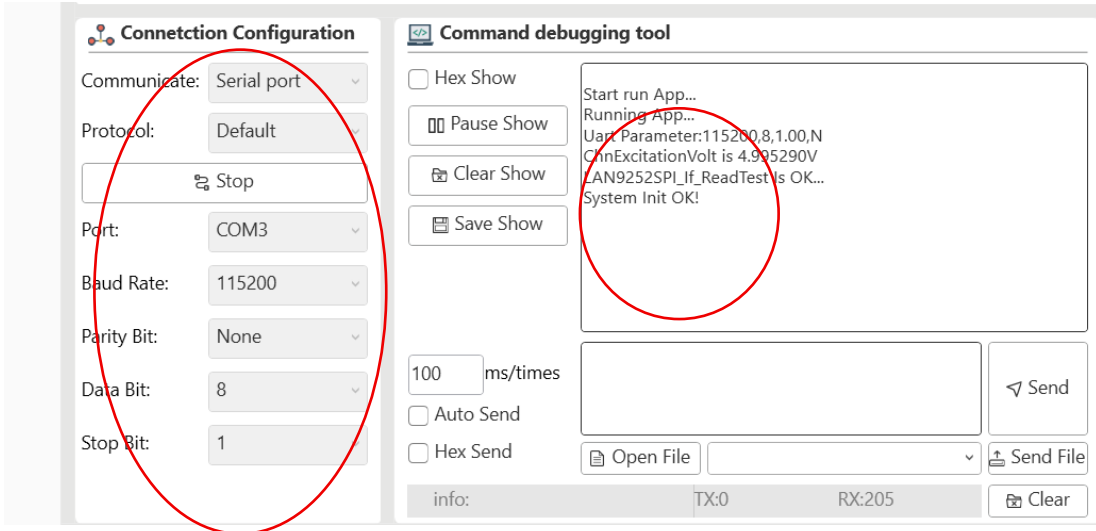
Connect sensor to PC(or other controller) via RS232:



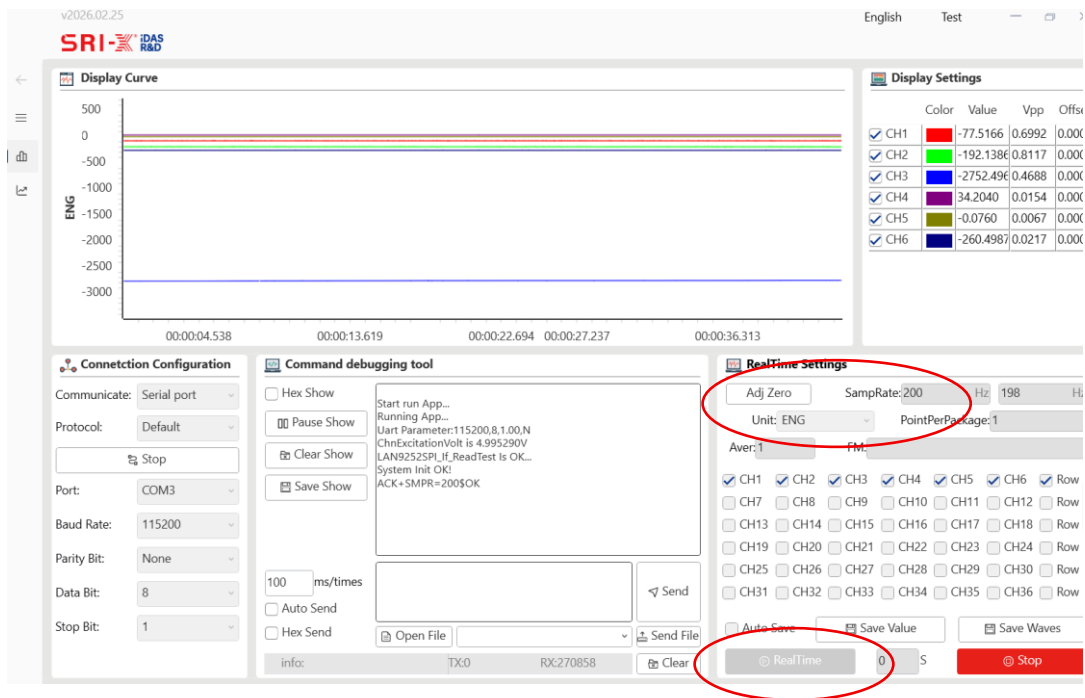
Software iDAS R&D can be used to get real time data from sensor via RS232. Customer also can develop their own software by AT commands to read data from sensor.

3.1.1 By iDAS R&D

Step1: Run iDAS R&D.exe. Then select PortName and BaudRate, and click “Connect” to connect to serial port.When connect successfully, the button will switch to “Stop”.



Step2: Turn power on. Initialization information will appear in the information box.
 Step 3: Select Eng in the “Unit” box, check CH1~CH6 boxes to represent FX, FY, FZ, MX, MY, MZ, then click “RealTime” button to start obtaining real-time force curves.



Note:

With RS232 communication, SampRate is up to 300 Hz for 6-channel data upload and 1 kHz for 1-channel data upload at BaudRate 115200bps.

3.1.2 By customer software

Step 1: Use command SMPF to set sampling rate. If the sampling rate is 100Hz:

```
AT+SMPF=100\r\n
```

Step 2: Use command GOD to get one package data or command GSD to get data continuously from sensor.

**Note:**

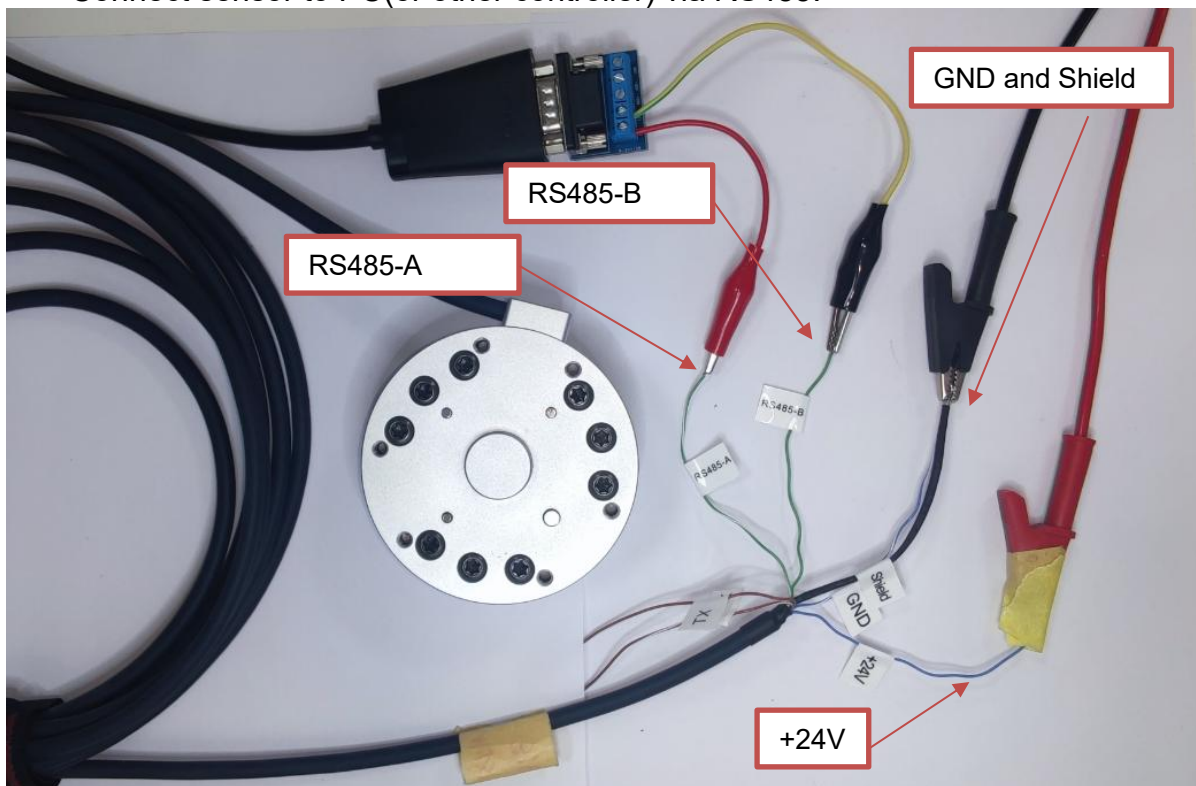
- ✍ The parameters set by Command SMPF are saved in sensor, and they are still available after power off.
- ✍ If iDAS R&D software is used, sensor is required to restart (Power off and Power on) before debugging your own codes.

3.2 RS485

RS485 wire definition:

Color	Definition	Note
Blue	+24V	
White Blue	GND	
White brown	RX	
Brown	TX	
White green	RS485-A	
Green	RS485-B	
Shield line	Shield	To reduce noise, it is recommended to connect the shield to the true ground in your lab.

Connect sensor to PC(or other controller) via RS485:

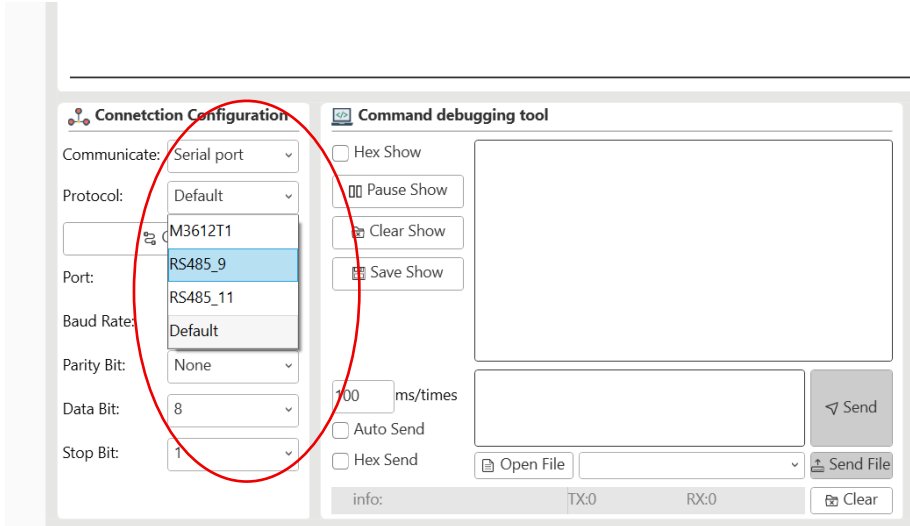


RS485 default configuration:

Baud Rate	Even parity	Stop Bit	Data Bits
460800bps	E	1	8

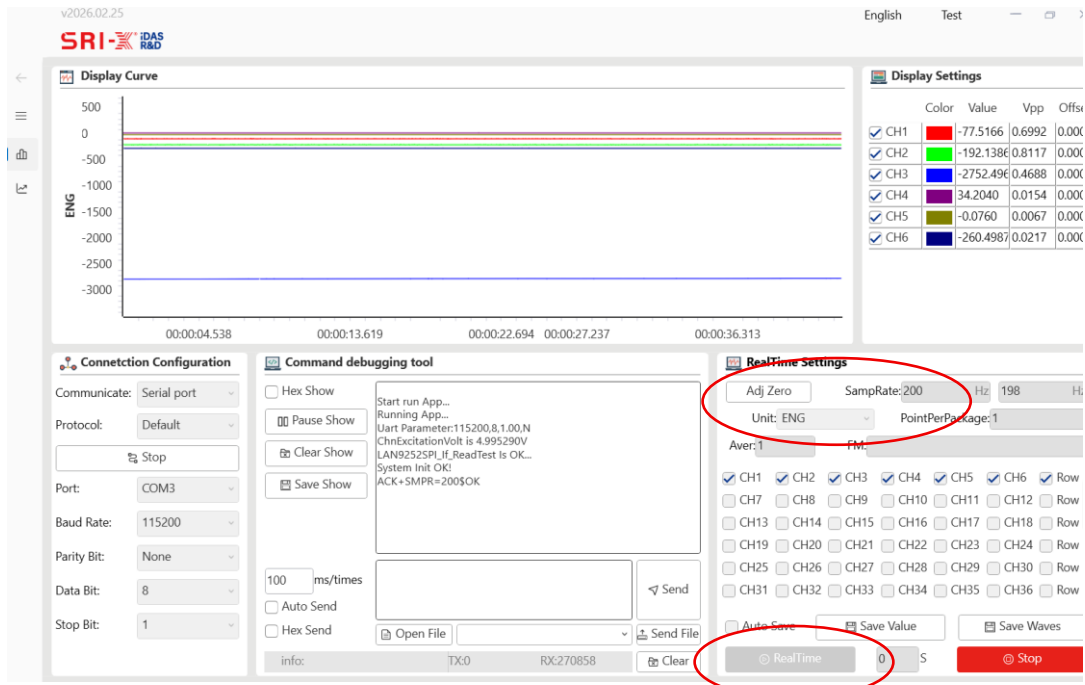
3.2.1 By iDAS R&D

Step1: Run iDAS R&D.exe. Then select PortName and BaudRate, select RS485 protocol(different sensor has different protocol) and click “Connect” to connect to serial port.When connect successfully, the button will switch to “Stop”.



Step2: Power on the sensor.

Step3: Select Eng in the “Unit” box, check CH1~CH6 boxes to represent FX, FY, FZ, MX, MY, MZ, then click “RealTime” button to start obtaining real-time force curves.



3.2.2 By customer software

RS485 is supported by M4313S1A. RS485 works in half-duplex.

After powered on, sensor keeps pumping the data out in 2000HZ without any commands. When powered off, sensor stop to send data out.

Communication parameters of RS485:

Baud Rate	Even parity	Stop Bit	Data Bits
460800bps	E	1	8

Application Layer Protocol of RS485:

One data frame includes 14 bytes, shown as below:

Header	Fx	Fy	Fz	Mx	My	Mz	Reserved	CRC-Check
0xAA, 0x55	Total 11 Bytes (88bits)							1 Byte
	17 bits	17 bits	17 bits	12 bits	12 bits	12 bits	1 bit	
	0~16	17~33	34~50	51~62	63~74	75~86	87	

 **Note:**

- ✍ 1) 0xAA ,0x55: Frame header of data package.
- ✍ 2) Force and Torque data, total 87bits.
 - 0 ~ 16 bits: 10 times of Fx, ranges from -6553.5 to 6553.5.
 - 17 ~ 33 bits: 10 times of Fy, ranges from -6553.5 to 6553.5.
 - 34 ~ 50 bits: 10 times of Fz, ranges from -6553.5 to 6553.5.
 - 51 ~ 62 bits: 10 times of Mx, ranges from -204.7 to 204.7.
 - 63 ~ 74 bits: 10 times of My, ranges from -204.7 to 204.7.
 - 75 ~ 86 bits: 10 times of Mz, ranges from -204.7 to 204.7.

```

✍ 3) CRC-Check: CRC check function is shown as below.
uint8_t MyCRC_GetCRC8(uint8_t *pDat,uint8_t bytes)
{
uint8_t CRC8_POLY = 0x8C; // X^8+ X^5 + X^4 + 1
uint8_t crc;
uint8_t temp = 0x00, nLen = 0x00;
uint8_t i = 0x00;
crc = 0x00;
nLen = bytes;
while(nLen--) //crc check
{
    crc = crc^*pDat;
    for(i = 0x00; i < 8; i++)
    {
        temp = (crc & 0x01);
        crc >>= 1;
        if(temp)
            crc ^= CRC8_POLY;
    } pDat++;
} return crc;
}
    
```

- ✍ 4) Big endian. The highest bit is the sign bit.

How to get FX to MZ from the data package:

Define the arrui8RxBuf[] is all the data package.

So, arrui8RxBuf[0]=0xAA and arrui8RxBuf[1]=0x55. arrui8RxBuf[2] to arrui8RxBuf[12] are the data of FZ to MZ.

$FX0 = (\text{arrui8RxBuf}[2] \ll 9) + (\text{arrui8RxBuf}[3] \ll 1) + ((\text{arrui8RxBuf}[4] \& 0x80) \gg 7);$

$FY0 = ((\text{arrui8RxBuf}[4] \& 0x7F) \ll 10) + ((\text{arrui8RxBuf}[5] \& 0xFF) \ll 2) + ((\text{arrui8RxBuf}[6] \& 0xC0) \gg 6);$

$FZ0 = ((\text{arrui8RxBuf}[6] \& 0x3F) \ll 11) + ((\text{arrui8RxBuf}[7] \& 0xFF) \ll 3) + ((\text{arrui8RxBuf}[8] \& 0xE0) \gg 5);$

$MX0 = ((\text{arrui8RxBuf}[8] \& 0x1F) \ll 7) + ((\text{arrui8RxBuf}[9] \& 0xFE) \gg 1);$

$MY0 = ((\text{arrui8RxBuf}[9] \& 0x01) \ll 11) + ((\text{arrui8RxBuf}[10] \& 0xFF) \ll 3) + ((\text{arrui8RxBuf}[11] \& 0xE0) \gg 5);$

$MZ0 = ((\text{arrui8RxBuf}[11] \& 0x1F) \ll 7) + ((\text{arrui8RxBuf}[12] \& 0xFE) \gg 1);$

The highest bit is the sign bit.

Force and torque of sensor are:

$FX = FX0/10$

$FY = FY0/10$

$FZ = FZ0/10$

$MX = MX0/10$

$MY = MY0/10$

$MZ = MZ0/10$

For example, we get $FX0 = 0x100E0 = 1\ 0000\ 0000\ 1110\ 0000$, "1" is the sign bit and denotes FX is a negative number.

Convert 0000 0000 1110 0000 to decimal, we get 224. So, $FX = FX0/10 = -22.4$.

The other example:

We get $FX0 = 0x1AC = 0\ 0000\ 0001\ 1010\ 1100$, "0" is the sign bit and denotes FX is a positive number.

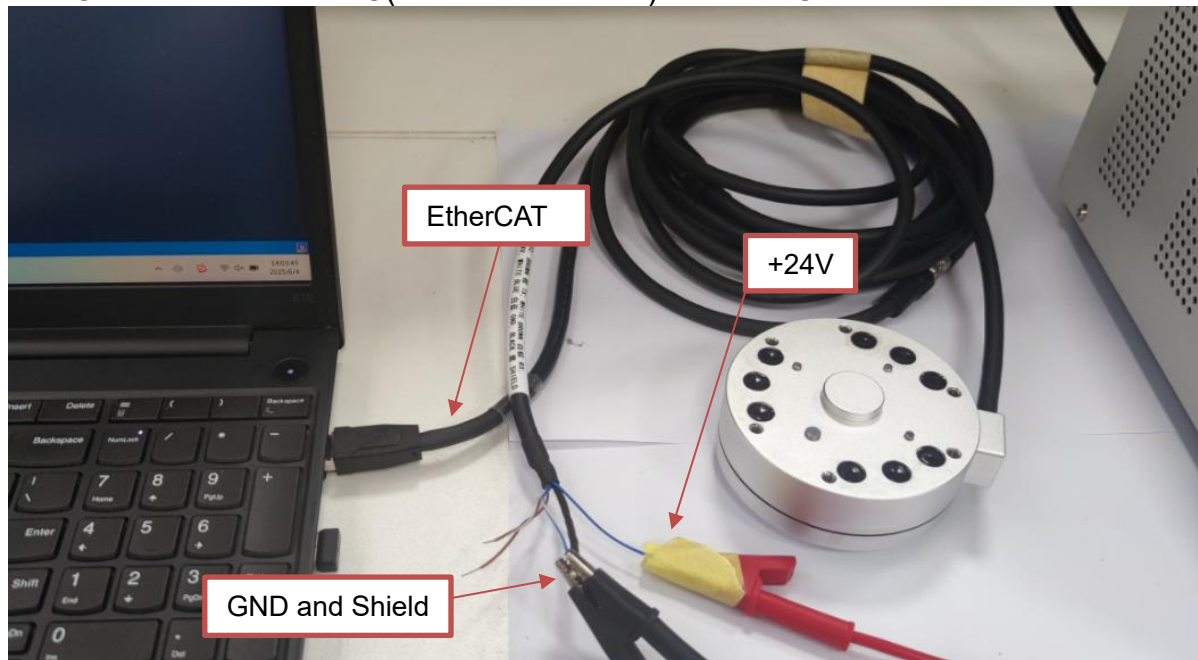
Convert 0000 0001 1010 1100 to decimal, we get 428. So, $FX = FX0/10 = 42.8$.

3.3 EtherCAT

EtherCAT wire definition:

Color	Definition	Note
Blue	+24V	
White Blue	GND	
White brown	RX	
Brown	TX	
White orange	TXP	EtherCAT, RJ45 connector
Orange	TXN	
White green	RXP	
Green	RXN	
Shield line	Shield	To reduce noise, it is recommended to connect the shield to the true ground in your lab.

Connect sensor to PC(or other controller) via EtherCAT:



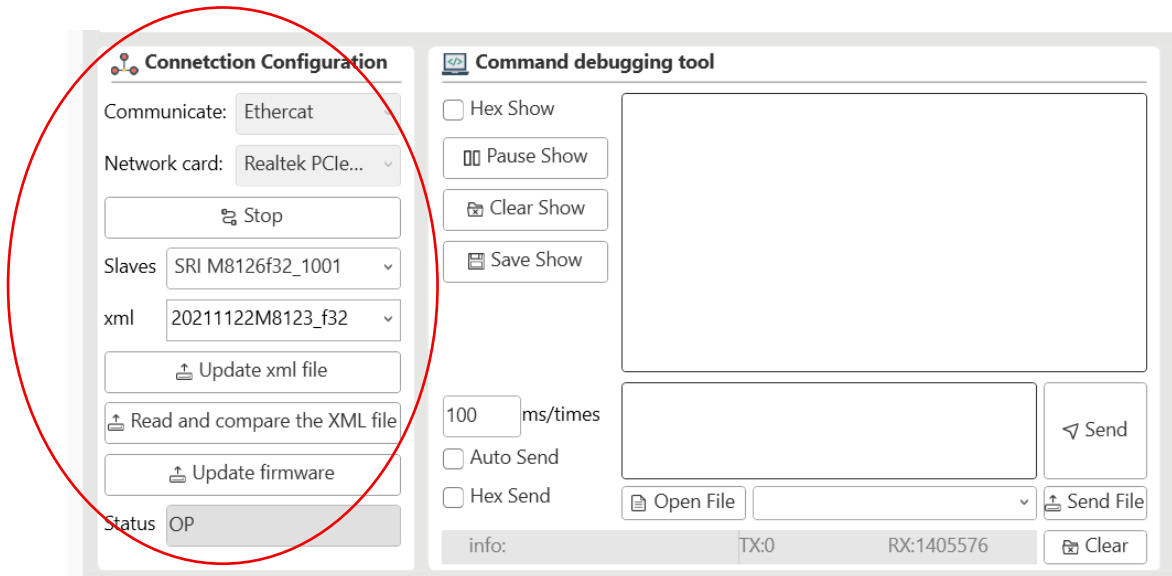
The dictionary file(*.xml) is supplied, can be connected into EtherCAT controller easily.The EtherCAT dictionary is shown below.

Object	Name	Data Type	W/R	Discription
0x1000	Device Type	UINT32	RO	
0x1008	Manufacturer Device Name	STRING	RO	
0x1009	Hardware Version	STRING	RO	
0x100A	Software Version	STRING	RO	
0x1018	Identity Object	RECORD	RO	
0x1018.01	Vendor ID	UINT32	RO	
0x1018.02	Product Code	UINT32	RO	
0x1018.03	Revision Number	UINT32	RO	
0x1018.04	Serial Number	UINT32	RO	
0x1601	RPDO	UINT32	RO	
0x1A03	TPDO	UINT32	RO	
0x1A02	TPDO	UINT32	RO	
0x1C12	SM2 PDO	REC	RW	
0x1C13	SM3 PDO	REC	RW	
0x6030.01	DataNo	UINT16	RO	unsigned int 16bit 0~65535 increment each sample.
0x6030.02	Fx	REAL	RO	float 32bit eng. unit: N
0x6030.03	Fy	REAL	RO	float 32bit eng. unit: N
0x6030.04	Fz	REAL	RO	float 32bit eng. unit: N
0x6030.05	Mx	REAL	RO	float 32bit eng. unit: Nm
0x6030.06	My	REAL	RO	float 32bit eng. unit: Nm
0x6030.07	Mz	REAL	RO	float 32bit eng. unit: Nm
0x6020.01	DataNo	UINT16	RO	unsigned int 16bit 0~65535 increment each sample.
0x6020.02	Fx	INT32	RO	Fx = INT32/10000 eng. unit: N
0x6020.03	Fy	INT32	RO	Fy = INT32/10000 eng. unit: N
0x6020.04	Fz	INT32	RO	Fz = INT32/10000 eng. unit: N
0x6020.05	Mx	INT32	RO	Mx = INT32/10000 eng. unit: Nm
0x6020.06	My	INT32	RO	My = INT32/10000 eng. unit: Nm
0x6020.07	Mz	INT32	RO	Mz = INT32/10000 eng. unit: Nm
0x7010.01	Para1	UINT16	RW	NA, reserved for future use
0x7010.02	Para2	INT16	RW	NA, reserved for future use
0x7010.03	Para3	INT16	RW	NA, reserved for future use

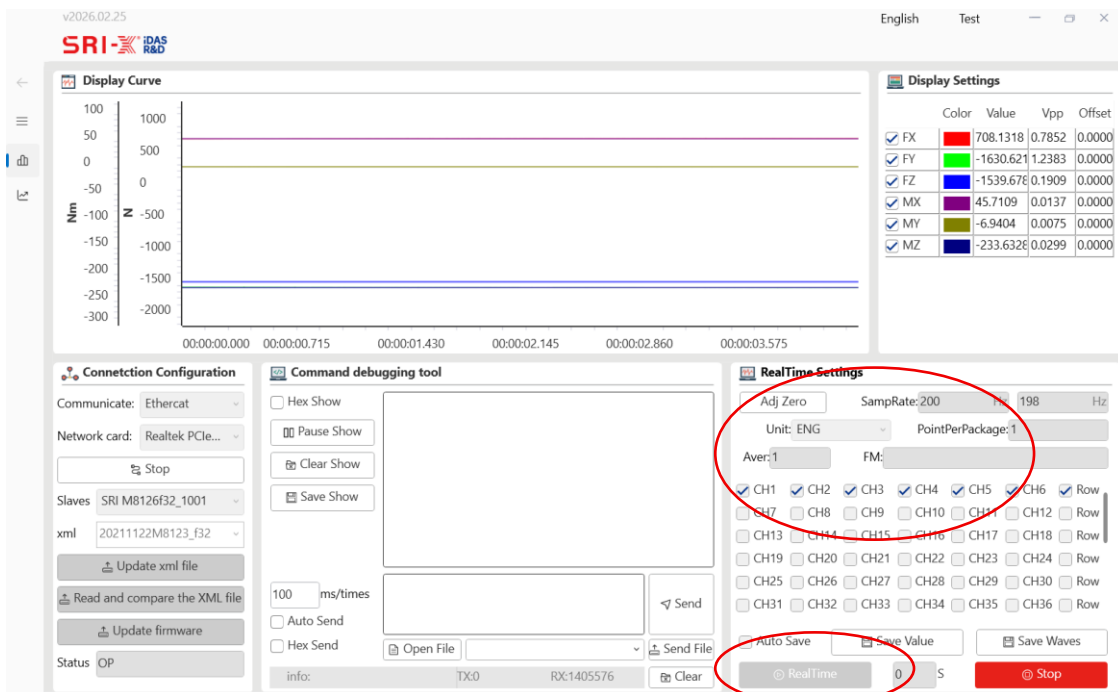
3.3.1 By iDAS R&D

Step 1: Run iDAS R&D.exe. Then set Communicate to EtherCAT, select Network card (your PC's Ethernet card).

Click “Connect” button, software will connect to M8229 automatically. When connect successfully, the button will switch to “Stop”.



Step 2: Select Eng in the output Unit box; check CH1~CH6 boxes to represent FX, FY, FZ, MX, MY, MZ; then click “RealTime” button to start obtaining real-time force/torque curves.



3.3.2 By TwinCAT3

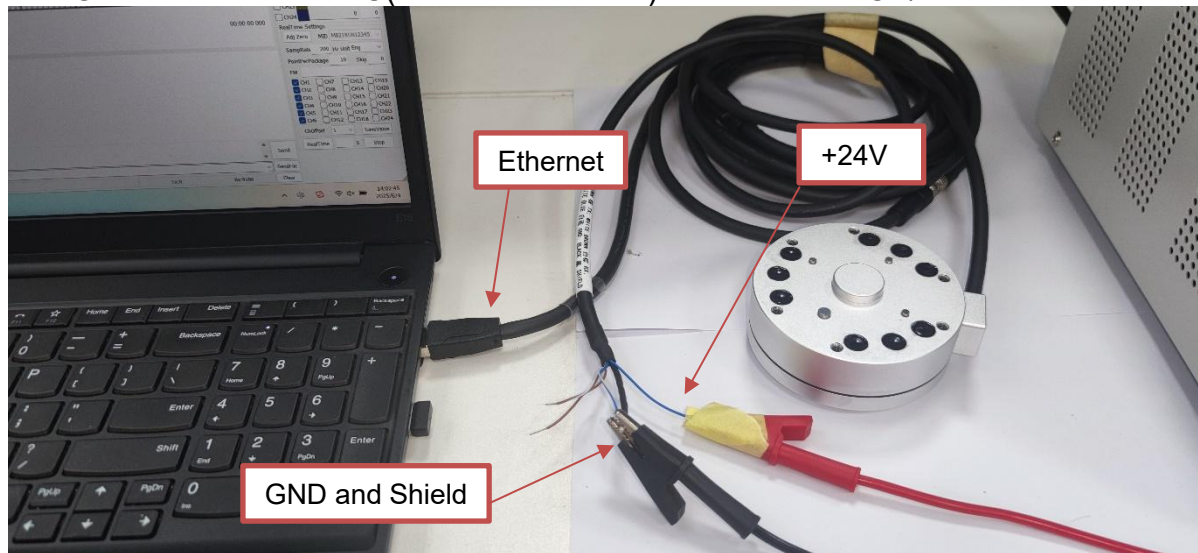
Please contact SRI to get the user's manual "How to debug SRI's sensor in TwinCAT.pdf"

3.4 Ethernet TCP/IP

Ethernet TCP/IP wire definition:

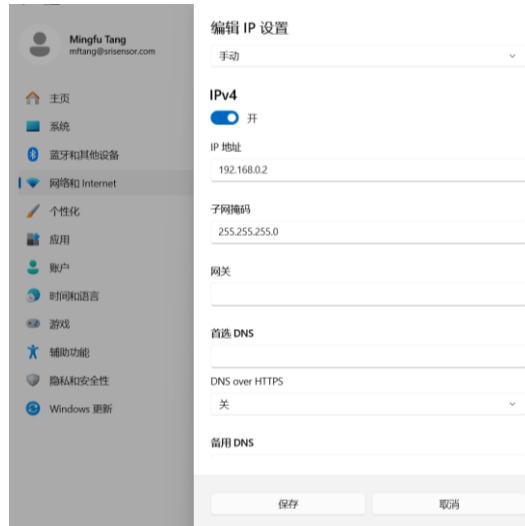
Color	Definition	Note
Blue	+24V	
White Blue	GND	
White brown	RX	
Brown	TX	
White orange	TXP	Ethernet, RJ45 connector
Orange	TXN	
White green	RXP	
Green	RXN	
Shield line	Shield	To reduce noise, it is recommended to connect the shield to the true ground in your lab.

Connect sensor to PC(or other controller) via Ethernet TCP/IP:

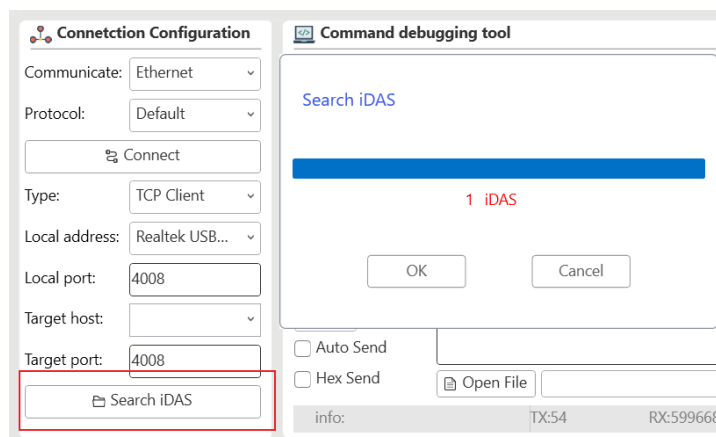


3.4.1 By iDAS R&D

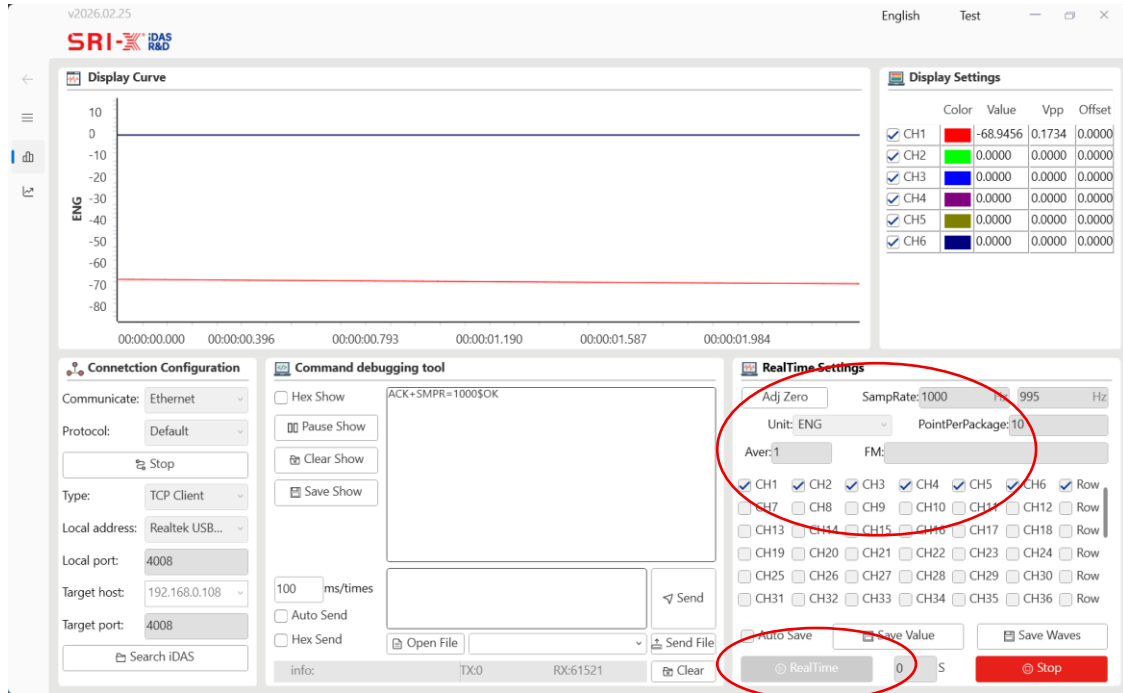
Step 1: Configure Ethernet on PC. Enter 192.168.0.2 in the [IP Address] field and 255.255.255.0 in the [Subnet Mask] field. Click OK to complete the configuration.



Step 2: After connecting and powering on the data acquisition card and PC computer, set the software communication to Ethernet and select protocol Default. Choose the network type according to the data acquisition card protocol, select the local address corresponding to the network adapter connected to the data acquisition card, and use the default port 4008. For TCP protocol data acquisition cards, click Search iDAS to automatically locate the IP address of the data acquisition card, then click Connect.



Step 3: Select Eng in the output Unit box; check CH1~CH6 boxes to represent FX, FY, FZ, MX, MY, MZ; then click “RealTime” button to start obtaining real-time force/torque curves.



3.4.2 By customer software

Step 1: Use command SMPF to set sampling rate. If the sampling rate is 100Hz:

AT+SMPPF=100\r\n

Step 2: Use command GOD to get one package data or command GSD to get data continuously from sensor.

Note:

- The parameters set by Command SMPF are saved in sensor, and they are still available after power off.
- If iDAS R&D software is used, sensor is required to restart (Power off and Power on) before debugging your own codes.

3.5 CAN

CAN wire definition:

Color	Definition	Note
Orange	+24V	电源正极
Black	GND	电源负极
Grey	RX	接 RS232 的 RX
Brown	TX	接 RS232 的 TX
Yellow	CANH	CANH
Green	CANL	CANL
Shield line	Shield	To reduce noise, it is recommended to connect the shield to the true ground in your lab.

The default baud rate of CAN is 1Mbps. It can be configured by command CRATE via RS232.

CAN bus command:

CAN ID	Command words	Parameters (2 Bytes)	Function
CAN ID	0x8A	0x00 00	Stop get data
CAN ID	0x8A	0x00 10	0x0010=16, means ask sensor to send data out continuously with time gap 16ms.
CAN ID	0x8A	0x00 0A	0x000A=10, means ask sensor to send data out continuously with time gap 10ms.



Note:

 The length of parameters must be 2 bytes.

CAN bus data protocol:

Data will be sent from sensor to CAN master with 3 CAN IDs, the default IDs are 0x291, 0x292 and 0x293. They can be configured by command CTXIDL via RS232.

CAN ID	0 ~ 3Bytes	4 ~ 7Bytes	Note
0x291	FX	FY	Float, comply with IEEE R32.24
0x292	FZ	MX	Float, comply with IEEE R32.24
0x293	MY	MZ	Float, comply with IEEE R32.24

Example:

Send command: ID: 0x80; command and parameter: 8A 00 01

Data from sensor:

ID:0x291; Data: 2B 20 37 40 4E 83 39 41

ID:0x292; Data: AE F8 D3 3F 68 6C 41 BF

ID:0x293; Data: BB E7 85 3E 64 95 BF 3E

FX: 2B 20 37 40, float(4037202B) convert to decimal data is 2.861338;

FY: 4E 83 39 41, float(4139834E) convert to decimal data is 11.594556;

FZ: AE F8 D3 3F, float(3FD3F8AE) convert to decimal data is 1.656026;

MX: 68 6C 41 BF, float(BF416C68) convert to decimal data is -0.755560;

MY: BB E7 85 3E, float(3E85E7BB) convert to decimal data is 0.261533;

MZ: 64 95 BF 3E, float(3EBF9564) convert to decimal data is 0.374186;

4. Commands

Definition:

Master: The equipment that send commands to M4313SXX, such as PC or the user's control system. M4313SXX is called as Slave Equipment.

ASCII Code: America Standard Code for Information Interchange, refer to ISO 646.

M4313SXX commands are comprised of ASCII codes.

Command structures are shown as follows:

Send to Slave Equipment:

AT+CMD=Parameter\r\n

Response from Slave Equipment: (Except for the command GOD and GSD)

ACK+CMD=Parameter\$ResponseCode\r\n



All data that sent to slave equipment must be ASCII code.
 All data that received from slave equipment are ASCII code.
 Before sent or after received, the data must be converted to or from ASCII

Descriptions:

AT: Frame Header when sending data. All data that are sent to Slave Equipment must be started with AT.

ACK: Frame Header when receiving data. All data that are received from Slave Equipment are started with ACK.

CMD: Command, such as SMPF.




Parameter: Parameters follow a command.

\r\n: Return symbol (Enter), which denotes the end of a command. ASCII code of "\r\n" is "0D 0A".

ResponseCode: Response code, such as OK or ERROR.

\$: Interval symbol.

Note:

-  Parameter '?' denotes that Master is asking for a response from Slave Equipment.
-  M4313SXX will not respond until the command sent by Master is executed.
-  When debugging with iDAS R&D software, the software performs ASCII conversion and put \r\n at the end of the command automatically.

Example:

Send: AT+SMPF=?\r\n

ASCII: 41 54 2B 53 4D 50 52 3D 3F 0D 0A

Response: ACK+SMPF=100\$OK\r\n

ASCII: 41 43 4B 2B 53 4D 50 46 3D 31 30 30 24 4F 4B 0D 0A

Send: AT+GOD\r\n

ASCII: 41 54 2B 47 4F 44 0D 0A

Response: AA 55 00 1B 04 BB A1 8C B8 41 E0 19 30 42 DD 82 B0 40 A2 62

B8 C0 DB 68 75 40 9B EB 16 40 30

Command	Function	Note
Configuration of RS232/CAN/Ethernet		
UARTCFG	Read or set parameters of RS232	Immediate active
EIP	Read or set Ethernet IP address	Active after restart
EMAC	Read or set Ethernet MAC address	Active after restart
EGW	Read or set Ethernet gateway	Active after restart
ENM	Read or set Ethernet netmask	Active after restart
CRATE	Read or set CAN baud rate	Active after restart
CIDT	Read or set ID type of CAN Bus	Active after restart
CFIDL	Read or set ID of CAN Bus	Active after restart
CFI	Read or set interval time between frames of CAN bus	Active after restart
System parameters		
SMPF	Read or set sampling rate	
DCPM	Read or set decoupling matrix coefficient	
DCPCU	Read or set matrix calculation unit	(mV or mV/V)
SFWV	Read firmware version	
Get real-time data from sensor		
DCKMD	Read or set data validation method	
GSD	Get data from M8228 continuously	
GOD	Get one package data from M8228	
ADJZF	Adjust zero offset of sensor	

4.1 Commands to configure RS232/Ethernet/CAN

4.1.1 UARTCFG / Communication Parameters for RS232

Description: To read or set parameters for RS232

Command Syntax:AT+UARTCFG=Rate,DataBit,StopBit,ParityBit

Command		Possible Response(s)
AT+UARTCFG=? \r\n		Rate,DataBit,StopBit,ParityBit
AT+UARTCFG=Rate, DataBit, StopBit, ParityBit\r\n		OK/ERROR
Note: Immediate effect.		
Parameters		
Parameter	Variable Type (Valid Range)	Description
Rate	Unsigned long int (0 ~ 2 ³² -1)	Baud Rate of RS232 in bps. Baud Rate of RS232 in M4313SXX can be 9600, 14400, 19200, 38400, 56000, 57600,115200, 230400, 256000, 460800, 921600bps. Default is 115200
DataBit	Int	Number of data bits in RS232 communication. The choices are: 5,6,7 and 8. Default is 8
StopBit	float	Number of stop bits in RS232 communication. The choices are: 0.5,1.0.1.5 and 2.0. Default is 1.0
ParityBit	char	Parity in RS232 communication. Choices are N,O and E, which denote none, odd and even respectively. Default is N

Example:

Send: AT+UARTCFG=?\r\n

Response: ACK+UARTCFG=115200,8,1.00,N\$OK\r\n

Send: AT+UARTCFG=19200,8,1.00,N\r\n

Response: Messy codes

The baud rate of the master equipment should be changed to the current set value of 19200 in order to communicate correctly.

4.1.2 EIP / Ethernet IP address

Description: To set Ethernet IP address

Command Syntax:AT+EIP=addr0.addr1.addr2.addr3

Command		Possible Response(s)
AT+EIP=?\r\n		addr0.addr1.addr2.addr3
AT+EIP= addr0.addr1.addr2.addr3\r\n		OK/ERROR
Note: Effect after M4313SXX is restarted.		
Parameters		
Parameter	Variable Type (Valid Range)	Description
addr0.addr1.addr2.addr3	String	Default: 192.168.0.108

Example:

Send: AT+EIP=?\r\n

Response: ACK+EIP=192.168.0.108\$OK\r\n

Send: AT+ EIP=192.168.0.108\r\n

Response: ACK+EIP=192.168.0.108\$OK\r\n

4.1.3 EMAC / Ethernet MAC

Description: To set Ethernet MAC.

Command Syntax:AT+EMAC=addr0-addr1-addr2-addr3-addr4-addr5

Command		Possible Response(s)
AT+EMAC=?\r\n		addr0-addr1-addr2-addr3-addr4-addr5
AT+EMAC= addr0-addr1-addr2-addr3-addr4-addr5\r\n		OK/ERROR
Note:		
Parameters		
Parameter	Variable Type (Valid Range)	Description
addr0-addr1-addr2-addr3-addr4-addr5	String	12-13-14-15-16-17

Example:

Send: AT+EMAC=?\r\n

Response: ACK+EMAC=12-13-14-15-16-17\$OK\r\n

Send: AT+EMAC=12-13-14-15-16-17\r\n

Response: ACK+EMAC=12-13-14-15-16-17\$OK\r\n

4.1.4 EGW / Ethernet Gateway address

Description: To set Ethernet gateway address.

Command Syntax: AT+EGW= addr0.addr1.addr2.addr3

Command		Possible Response(s)
AT+EGW=?\r\n		addr0.addr1.addr2.addr3
AT+EGW=addr0.addr1.addr2.addr3\r\n		OK/ERROR
Note:		
Parameters		
Parameter	Variable Type (Valid Range)	Description
addr0.addr1.addr2.addr3	String	192.168.0.1

Example:

Send: AT+EGW=?\r\n

Response: ACK+EGW=192.168.0.1\$OK\r\n

Send: AT+EGW=192.168.0.1\r\n

Response: ACK+EGW=192.168.0.1\$OK\r\n

4.1.5 ENM / Ethernet netmask

Description: To set Ethernet netmask.

Command Syntax: AT+ENM= addr0.addr1.addr2.addr3

Command		Possible Response(s)
AT+ENM=?\r\n		addr0.addr1.addr2.addr3
AT+ENM= addr0.addr1.addr2.addr3\r\n		OK/ERROR
Note:		
Parameters		
Parameter	Variable Type (Valid Range)	Description
addr0.addr1.addr2.addr3	String	255.255.255.0

Example:

Send: AT+ENM=?\r\n

Response: ACK+ENM=255.255.255.0\$OK\r\n

Send: AT+ENM=255.255.255.0\r\n

Response: ACK+ENM=255.255.255.0\$OK\r\n

4.1.6 CIDT / ID type for CAN Bus

Description: To read or set ID type for CAN Bus.

Command Syntax: AT+CIDT=Type

Command		Possible Response(s)
AT+CIDT=?\r\n		Type
AT+CIDT=Type\r\n		OK/ERROR
Note:		
Parameters		
Parameter	Variable Type (Valid Range)	Description
Type	String	STD: Standard 11 bits ID EXT: Extended 29 bits ID

Example:

Send: AT+CIDT=?\r\n

Response: ACK+CIDT=STD\$OK\r\n

Send: AT+CIDT=EXT\r\n

Response: ACK+CIDT=EXT\$OK\r\n

4.1.7 CFIDL / ID of CAN Bus

Description: To read or set ID of CAN Bus.

Command Syntax: AT+CFIDL= id1,id2,id3,...,idn

Command		Possible Response(s)
AT+CFIDL=?\r\n		id1,id2,id3,...,idn
AT+CFIDL=id1,id2,id3,...,idn\r\n		OK/ERROR
Note:		
Parameters		
Parameter	Variable Type (Valid Range)	Description
idn	0 ~ 2 ¹¹ or 0 ~ 2 ²⁹	NULL: no filter Up to 14 IDs: allow these IDs to be passed through

Example:

Send: AT+CFIDL=?\r\n

Response: ACK+CFIDL=NULL\$OK\r\n

Send: AT+CFIDL=0,125,126,127,128\r\n

Response: ACK+CFIDL=0,125,126,127,128\$OK\r\n

4.1.8 CRATE / Baud Rate of CAN Bus

Description: To read or set mode and baud rate of CAN Bus.

Command Syntax: AT+CRATE=Mode:rate

Command	Possible response(s)
AT+CRATE=?	Mode:rate
AT+CRATE=Mode:rate	OK/ERROR

Note:

1. The default baud rate of CAN is 1Mbps. For CAN FD, default arbitration phase baud rate is 500kbps, default data phase baud rate is 2Mbps(max.5Mbps).
2. Send “AT+CRATE= CAN,500000” to set the mode as CAN and the baud rate to 500kbps.
3. Send “ AT+CRATE=CANFD,500000,5000000” to set the mode as CANFD and the arbitration phase baud rate is 500kbps,the data phase baud rate is 5Mbps.
3. Only one method can be used each time.
4. New Baud Rate will be active after M4313SXX is restarted.

Parameters		
Parameter	Variable Type (Valid Range)	Description
Mode	String	Mode, can be CAN or CANFD
rate	Unsigned long int (0 ~ 2 ³² -1)	Baud Rate in bps. For CAN, the baud rate can be: 1000000, 800000, 500000, 250000, 125000, 100000, 50000 For CANFD, arbitration phase baud rate can be: 1000000, 800000, 500000, 250000, 125000, 100000, 50000 For CANFD, data phase baud rate can be: 5000000, 4000000, 2000000, 1000000, 800000, 500000, 250000

Example:

Send: AT+CRATE=?\r\n

Response: ACK+CRATE=CAN,500000\$OK\r\n

Send: AT+CRATE=CANFD,500000,2000000\r\n

Response: ACK+CRATE=CANFD,500000,2000000\$OK\r\n

4.1.9 CFI / Interval time between frames of CAN Bus

Description: To set Interval time between frames of CAN Bus

Command Syntax: AT+CFI=IntervalTime

Command		Possible response(s)
AT+CFI=?		IntervalTime
AT+CFI=IntervalTime		OK/ERROR
Note: New interval time will be effective after sesnor is restarted.		
Parameters		
Parameter	Variable Type (Valid Range)	Description
IntervalTime	0 ~ 10000	Interval time in μ s. The default value in firmware is 0 μ s.

Example:

Send: AT+CFI=?\r\n

Response: ACK+CFI=10\$OK \r\n

Send: AT+CFI=10\r\n

Response: ACK+CFI=10\$OK \r\n

4.2 SFWV / Firmware version

Description: To read firmware version.

Command Syntax:AT+SFWV=?

Command		Possible Response(s)
AT+SFWV=?\r\n		version
Note:		
Parameters		
Parameter	Variable Type (Valid Range)	Description
version	String	Version #

Example:

Send: AT+SFWV=?\r\n

Response: ACK+SFWV=V11.00\$OK\r\n

4.3 DCPM / Read or set decoupled matrix

Description: To read or set decoupled matrix

Command Syntax:AT+DCPM=Matrix

Command		Possible Response(s)
AT+DCPM=?\r\n		Matrix data
AT+DCPM=Matrix\r\n		OK/ERROR
Note:		
Parameters		
Parameter	Variable Type (Valid Range)	Description
Matrix	String	The format is as follows.

Example:

Send: AT+DCPM=?\r\n

Response: ACK+DCPM=(0.000041,-0.020164,-0.000348,0.020287,-0.000145,-0.000047);(-0.000160,-0.011703,-0.000089,-0.011668,-0.000217,0.023526);(-0.031415,-0.000185,-0.032273,0.000010,-0.031708,-0.000481);(-0.000888,-0.000014,0.000951,-0.000006,0.000029,0.000009);(-0.000521,0.000011,-0.000531,-0.000009,0.001061,0.000015);(0.000002,0.000754,-0.000008,0.000753,-0.000007,0.000768)\$OK\r\n

4.4 DCPCU / Calculation unit for decoupled data

Description: To set or read calculation unit.

Command Syntax:AT+DCPCU=Unit

Command		Possible Response(s)
AT+DCPCU=?\r\n		Unit
AT+DCPCU=Unit\r\n		OK/ERROR
Note:		
Parameters		
Parameter	Variable Type (Valid Range)	Description
Unit	String	MV and MVPV(mV/V)

Example:

Send: AT+DCPCU=?\r\n

Response: ACK+DCPCU=MV\$OK\r\n

Send: AT+DCPCU=MVPV\r\n

Response: ACK+DCPCU=MVPV\$OK\r\n

4.5 SMPF / Read or set sampling rate

Description: To read or set sampling rate.

Command Syntax: AT+SMPF=SampleFreq

Command		Possible response(s)
AT+SMPF=?		SampleFreq
AT+SMPF=SampleFreq		OK/ERROR
Note:		
Parameters		
Parameter	Variable Type (Valid Range)	Description
SampleFreq	Unsigned short int (1 ~ 2000)	Sampling Freq in Hz. For example, 200.

Example:

Send: AT+SMPF=?\r\n

Response: ACK+SMPF=300\$OK\r\n

Send: AT+SMPF=200\r\n

Response: ACK+SMPF=200\$OK\r\n

4.6 DCKMD / Set data-check method

Description: To set data-check method.

Command Syntax: AT+DCKMD=Mod

Command		Possible response(s)
AT+DCKMD=Mod		OK/ERROR
Note: iDAS R&D debugging software only supports SUM check.		
Parameters		
Parameter	Variable Type (Valid Range)	Description
Mod	String	Data check methods include SUM and CRC32; The default method SUM is used for data accumulation and verification.

Example:

Send: AT+DCKMD=?\r\n

Response: ACK+ DCKMD =SUM\$OK\r\n

4.7 GOD / Get one package data from M4313SXX

Description: To get one package data from M4313SXX.

Command Syntax: AT+GOD

Command		Possible response(s)
AT+GOD		DataFormat
Note:		
Parameters		
Parameter	Variable Type (Valid Range)	Description
DataFormat		Data package, refer to the following for details.

4.8 GSD / Get data continuously

Description: To get data continuously.

Command Syntax: AT+GSD

Command		Possible response(s)
AT+GSD		DataFormat
Note: To stop receiving data, send "AT+GSD=STOP\r\n" to M4313SXX.		
Parameters		
Parameter	Variable Type (Valid Range)	Description
DataFormat		Data package, refer to the following for details.

"DataFormat" is defined as follows:

Frame Header	PackageLength	PackageNo	Data	CRC32 / SUM
0xAA ,0x55	HB,LB	2Byte	(ChNum*N*DNpCH)Byte	4Byte / 1Byte



Note:

- ✍ 0xAA ,0x55: Frame header of data package.
PackageLength: Unsigned short int,16-bits, highest byte first, The length of data of each channel, which equals to 2+ChNum*N*DNpCH+1(SUM check) or 2+ ChNum*N*DNpCH+4(CRC32 check)
Where
ChNum: Total number of uploading channels, Default value 6
N: the output unit, Default value 4
DNpCH: Number of sampling points to upload in one package, Default value 1
- ✍ PackageNo: Every package is labeled, which increases in sequence from 0 to 65535.
- ✍ Data: Uploading data with the lowest byte first.
- ✍ CRC32/SUM: CRC32 or Checksum. The default data validation method is Checksum. Use Command DCKMD to select the data validation method (Checksum or CRC32).
CRC32 function (MyCRC_GetCRC32 (uint8_t *pData, uint16_t Length)) in C program is included in the CD-ROM.

Example:

Send: AT+DCKMD=SUM\r\n

Response: ACK+DCKMD=SUM\$OK\r\n

Send: AT+GOD\r\n

Response: AA 55 00 1B C4 C7 01 6A F4 C0 EF 7D 33 C0 49 62 C9 C0 A2
5C C6 BD A6 19 8F BD AF DA 69 3E 6E

Where

0xAA ,0x55: Frame header

00 1B: PackageLength $2+6*4*1+1=27$ bytes;

C4 C7: Package No 50375;

Channel 1 Engineering Unit: 01 6A F4 C0, single-precision float
(C0F46A01) converted into -7.637940;

Channel 2 Engineering Unit: EF 7D 33 C0, single-precision float
(C0337DEF) converted into -2.804561;

Channel 3 Engineering Unit: 49 62 C9 C0, single-precision float
(C0C96249) converted into -6.293248;

Channel 4 Engineering Unit: A2 5C C6 BD, single-precision float
(BDC65CA2) converted into -0.096856;

Channel 5 Engineering Unit: A6 19 8F BD, single-precision float
(BD8F19A6) converted into -0.069873;

Channel 6 Engineering Unit: AF DA 69 3E, single-precision float
(3E69DAAF) converted into 0.228373

SUM Check: 6E

4.9 ADJZF/ Adjust zero offset of sensor

Description: Adjust zero offset of sensor

Command Syntax: AT+ADJZF

Command	Possible response(s)
AT+ADJZF=?	0;0;0;0;0;0 or 1;1;1;1;1;1
AT+ADJZF=1;1;1;1;1;1	OK/ERROR

Note:

Send "1;1;1;1;1;1" to M4313SXX to adjust zero offset of sensor. M4313SXX takes more than 2 seconds to complete zero offset adjusting process. During this process, sensor must keep static and do not move.

Send "0;0;0;0;0;0" to make M4313SXX back to the status which is before zero offset adjusting.

Example:

Send: AT+ADJZF=?\r\n

Response: ACK+ADJZF=0;0;0;0;0;0\$OK\r\n

Send: AT+ADJZF=1;1;1;1;1;1\r\n

Response: ACK+ADJZF=1;1;1;1;1;1\$OK\r\n