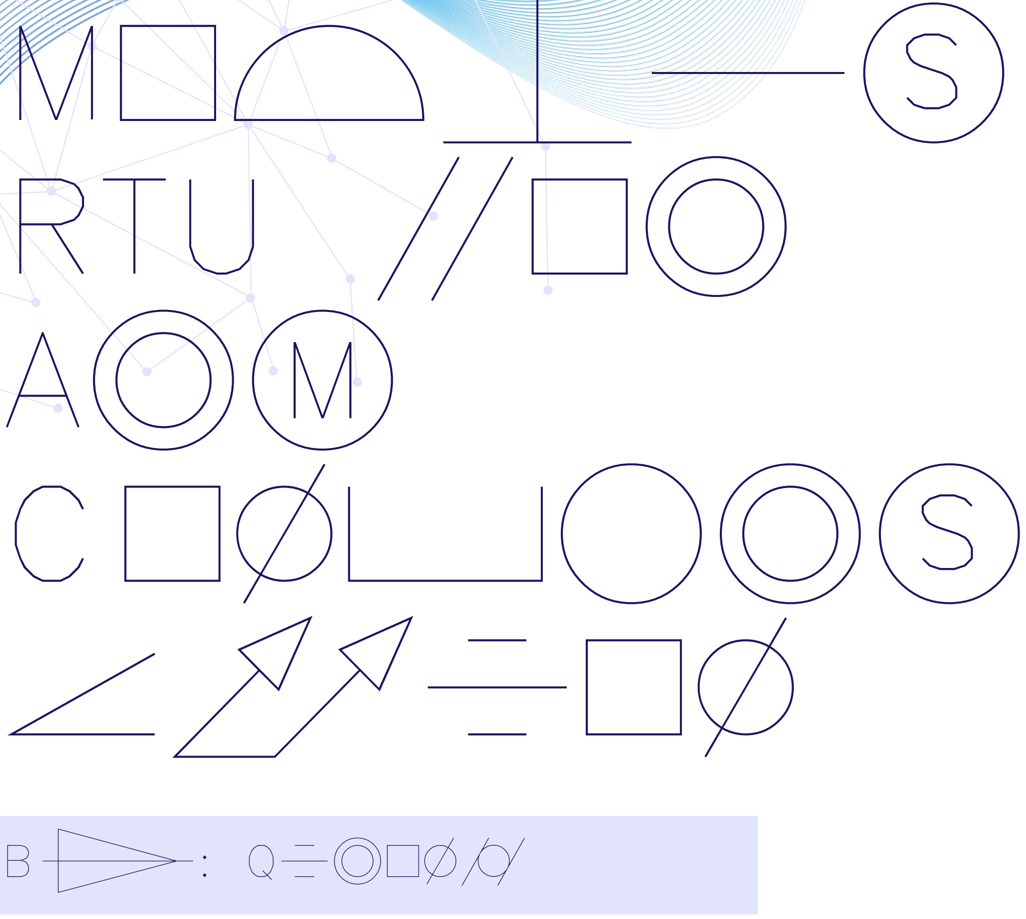


T∠⟂Ⓛ○ □//
⏥□⌀⌰○⌀⌰Ⓢ

01

I⌀⌰◎□⌓⏤⏥
⌰⌯□⌀ □//
M□⌓⟂⏤Ⓢ

02

I⌀⌰◎□⌓⏤⏥
⌰⌯□⌀ □//
RS-485

03

B◎⌯○// □//
Ⓜ⌲
⌵□◎⌒

04

C⌯◎⏥⏤⌯⌰
∠⌀⌓
↗∠◎⌓⌵∠◎
○

05

C□⌓○
○↧ⓅⓁ∠⌀∠⌰
⌯□⌀

06

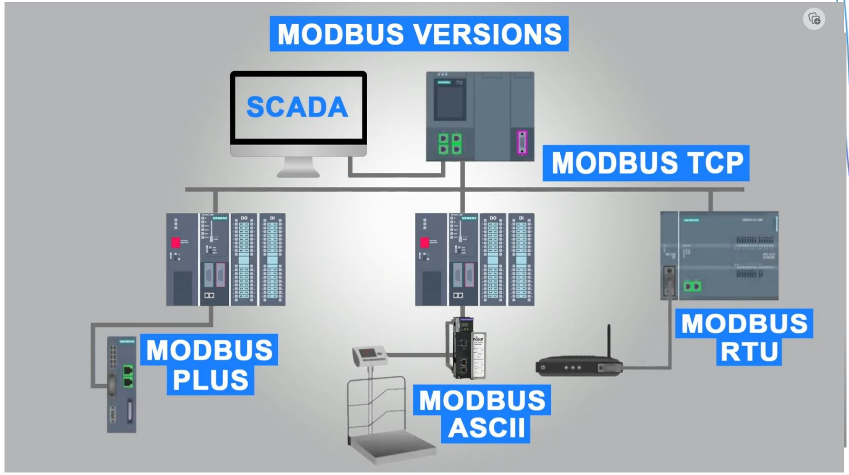
E↧Ⓟ○◎⌯Ⓜ○⌀
⌰
Ⓟ◎□⏥○ⓈⓈ
∠⌀⌓
◎○Ⓢ⏤Ⓛ⌰

01

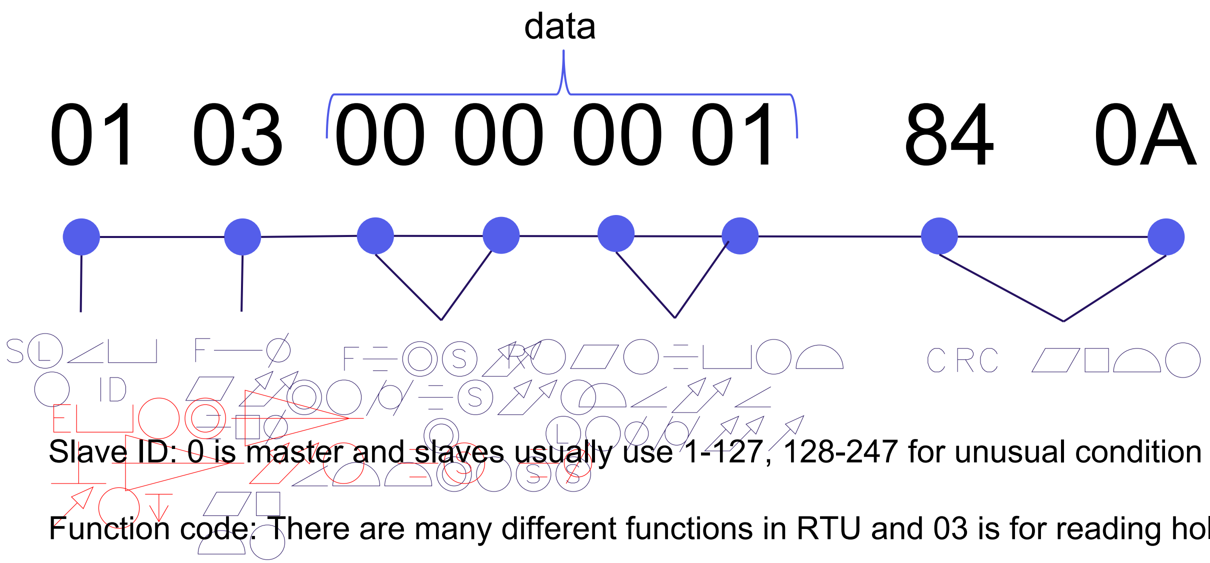
What is Modbus

Software-level communication protocol used in the field of industrial automation

- Master-Slave Architecture:
- The Master station sends requests and receives responses from slave stations.
- Up to 247 slaves
- Data Formats:



DATA FORMAT OF MODBUS RTU

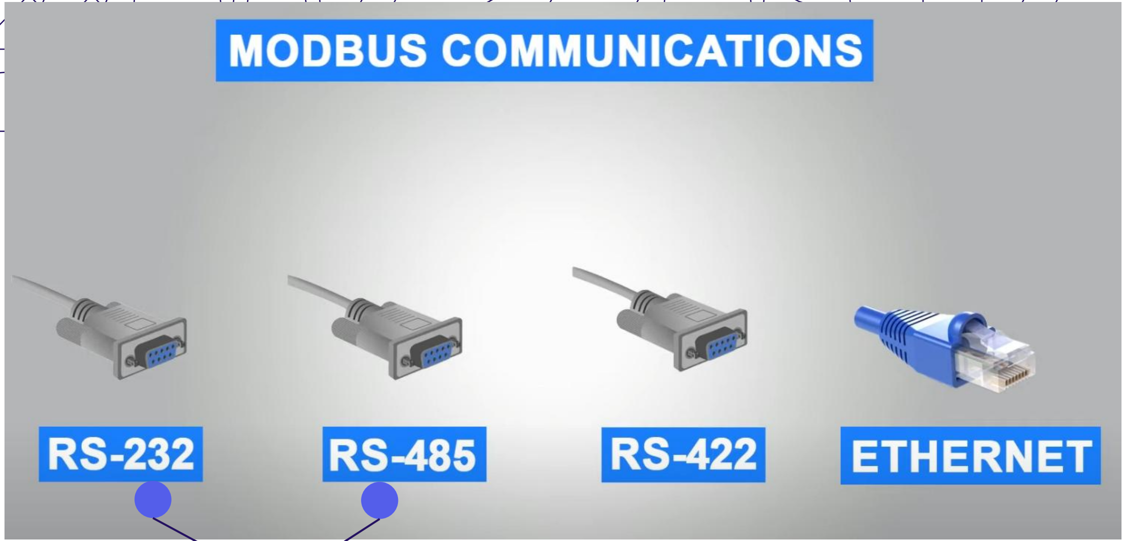


Slave ID: 0 is master and slaves usually use 1-127, 128-247 for unusual condition

Function code: There are many different functions in RTU and 03 is for reading holding registers

Data: Recognize the start address and end address of register, for this example means read 40001 address of holding register only.

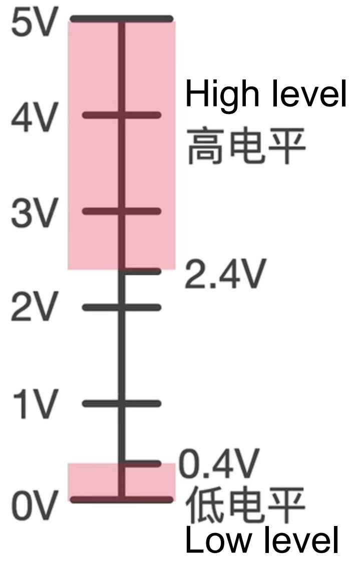
CRC code: To check if the received data is same with the sent data. CRC code is based on some function to compute and RTU has its own function



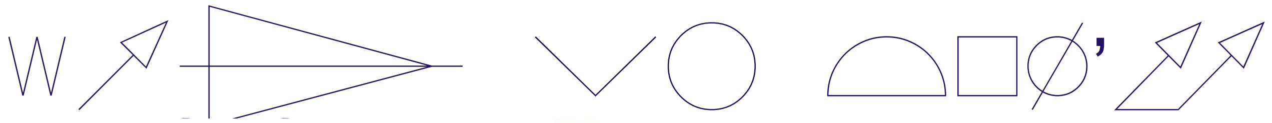
MODBUS
RTU

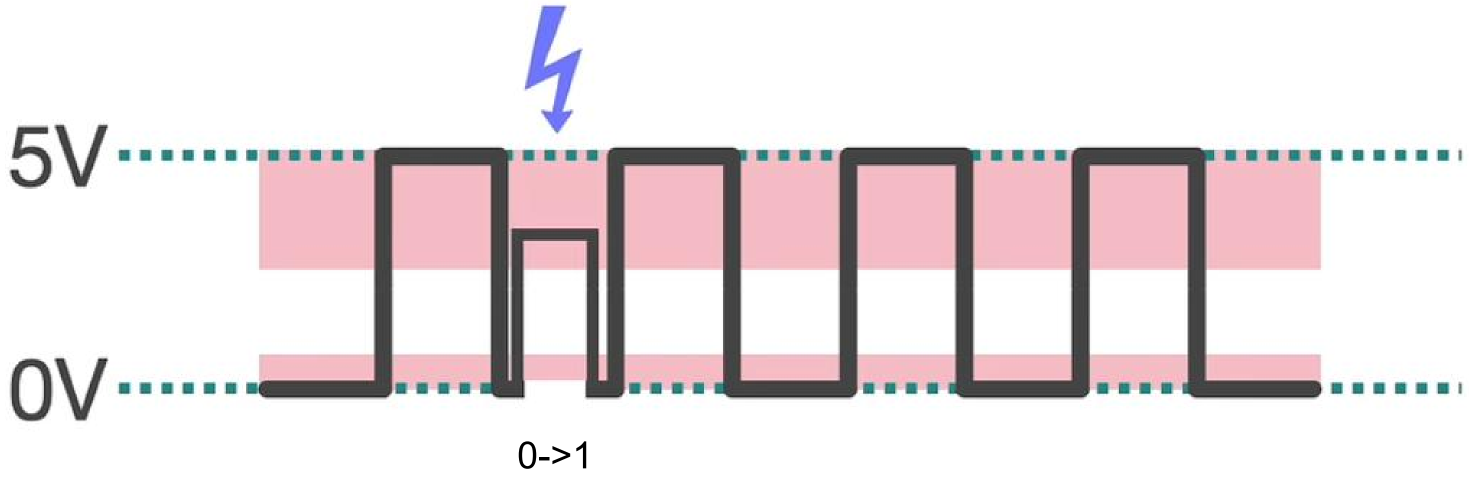
02

RS-485



Very poor anti-interference ability



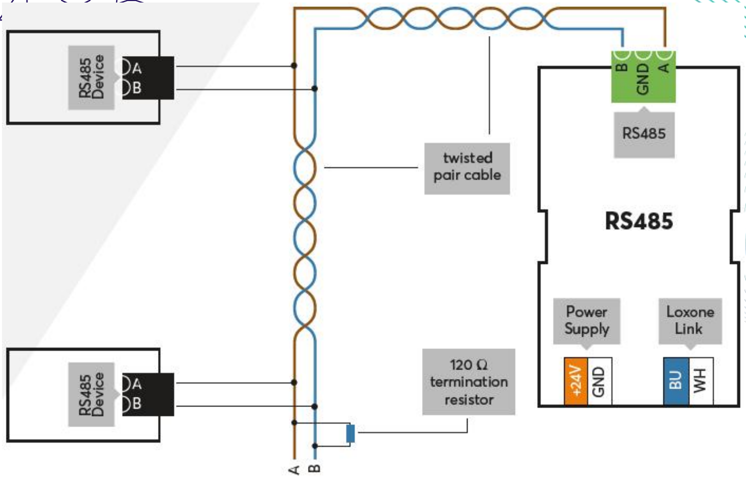


Very poor anti-interference ability

Transfer distance: less than 1 meter

W [illegible] RS-485

R [illegible]
S [illegible]
EIA
A [illegible] P [illegible]
[illegible]
[illegible]
S [illegible]

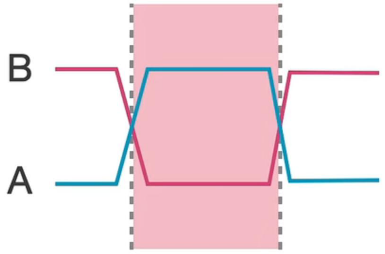


Features

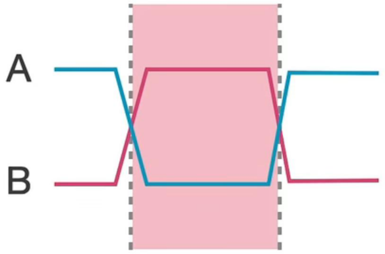
T [illegible]
[illegible] P [illegible]

H [illegible]

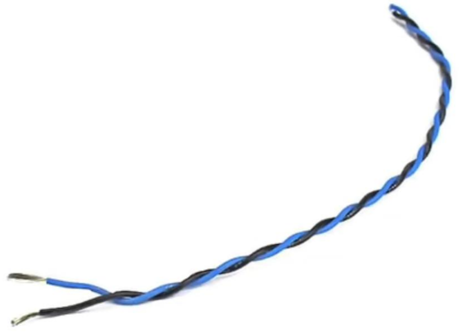
RS-485



Logic 0

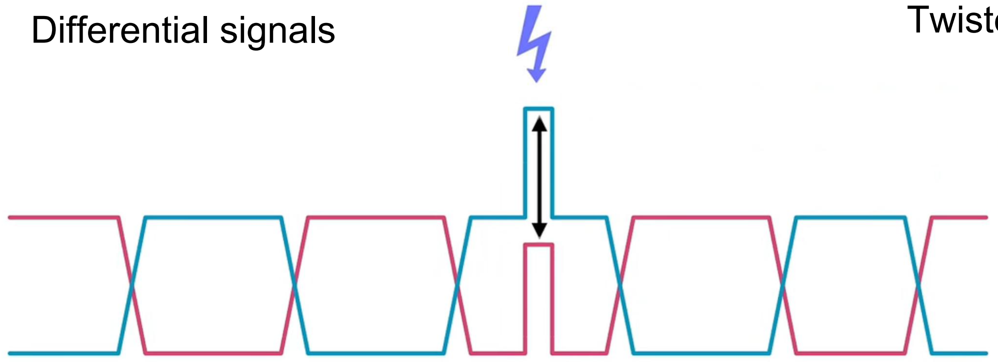


Logic 1



Differential signals

Twisted pair



Difference remain the same

Transfer distance form 1m to 1200m!

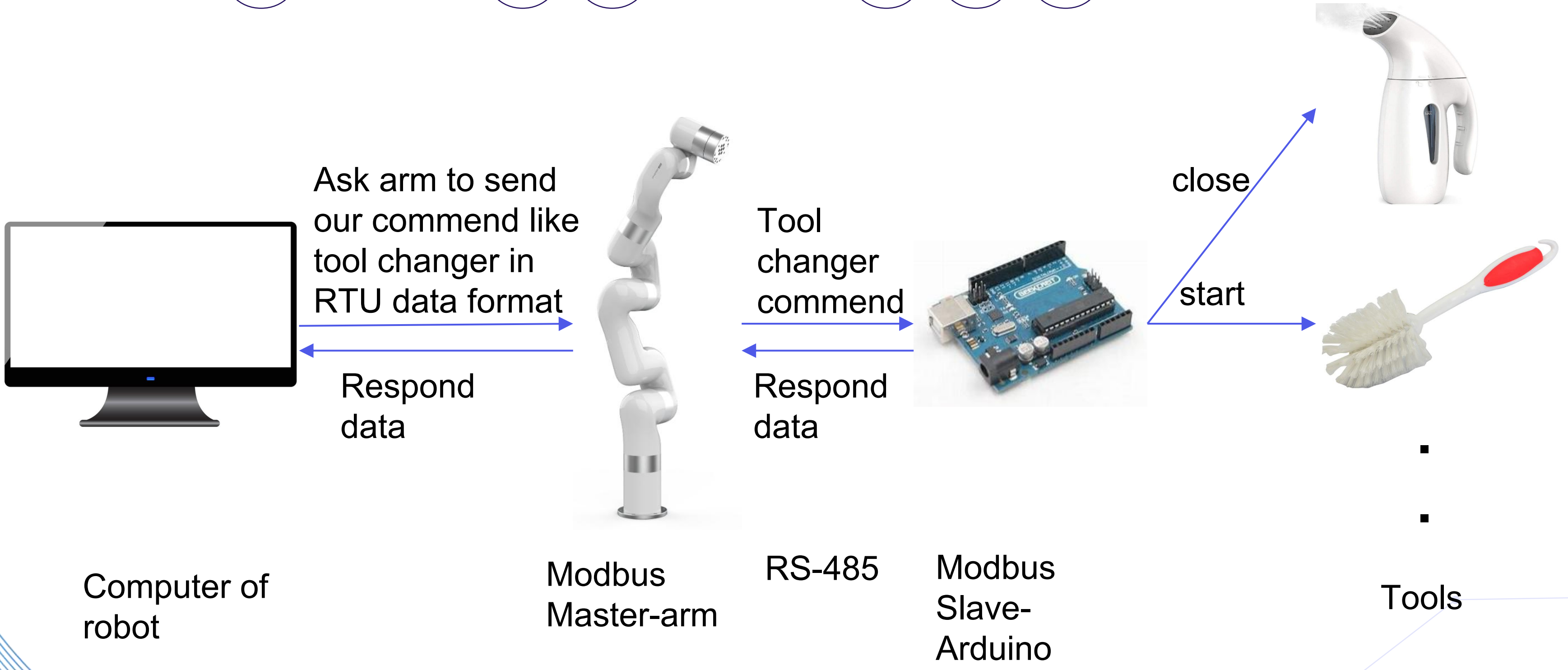
F□◎ □—◎ Ⓟ◎□⌖○▱⌰

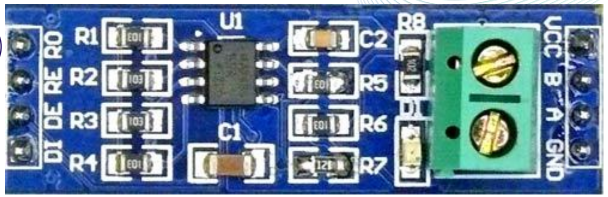
- M□⌓⊥—Ⓢ ∠⌀⌓
  ◎Ⓢ–485 ∠ⓁⓁ□⌵
  —Ⓢ
  ⌯ⓂⓅⓁ○Ⓜ○⌀⌰ 1
  ⌰□ Ⓜ∠⌀⌲
  ▱□ⓂⓂ—⌀⌯▱∠
  ⌰⌯□⌀

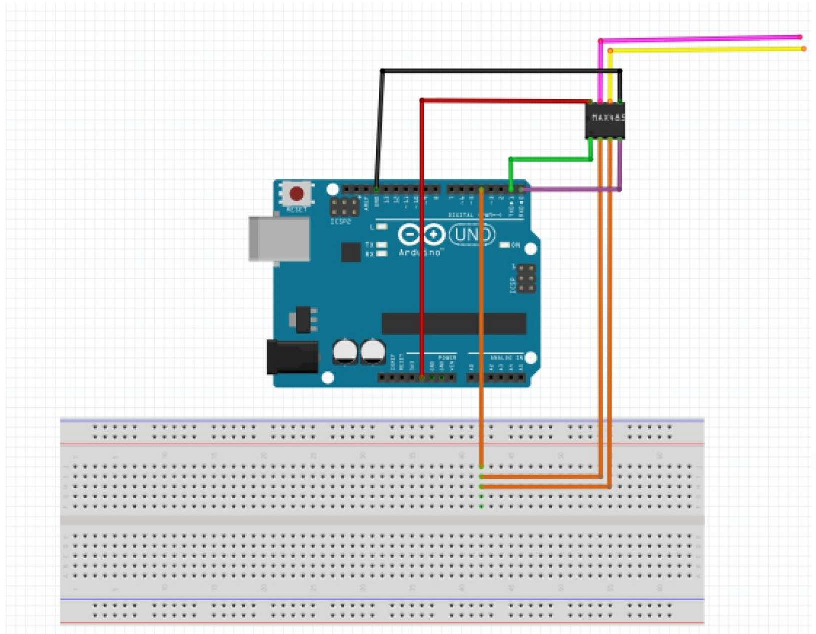
- M□⌓⊥—Ⓢ RTU
  ∠ⓁⓁ□⌵ —Ⓢ
  ⌓□—⊥Ⓛ○
  ▱↗○▱⌒ ⌰↗○
  ⌓∠⌰∠ ⊥⌲
  —Ⓢ⌯⌀⌀ CRC

03

WORK PROCESS



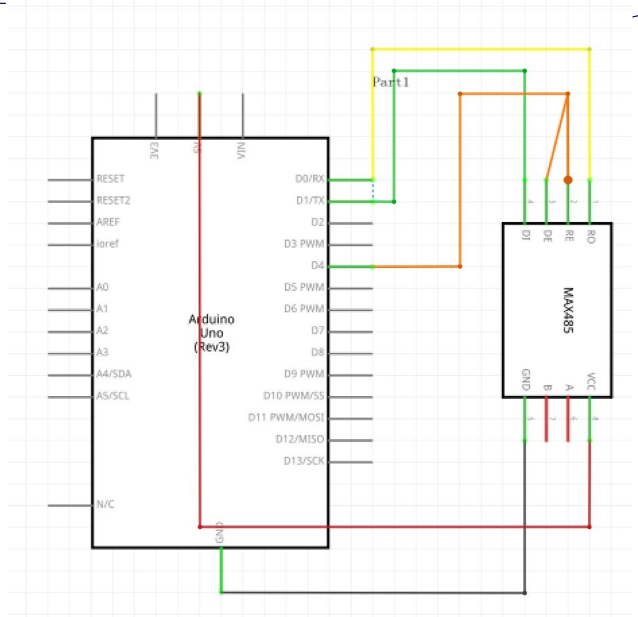


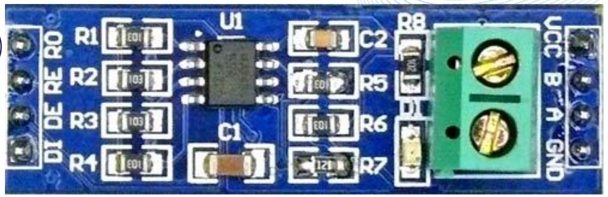


MAX485: 485-TTL

Pin Name	Pin Description
VCC	5V
A	Non-inverting Receiver Input Non-Inverting Driver Output
B	Inverting Receiver Input Inverting Driver Output
GND	GND (0V)
RO	Receiver Out (TX pin)
RE	Receiver Output (LOW-Enable)
DE	Driver Output (HIGH-Enable)
DI	Driver Input (RX pin)

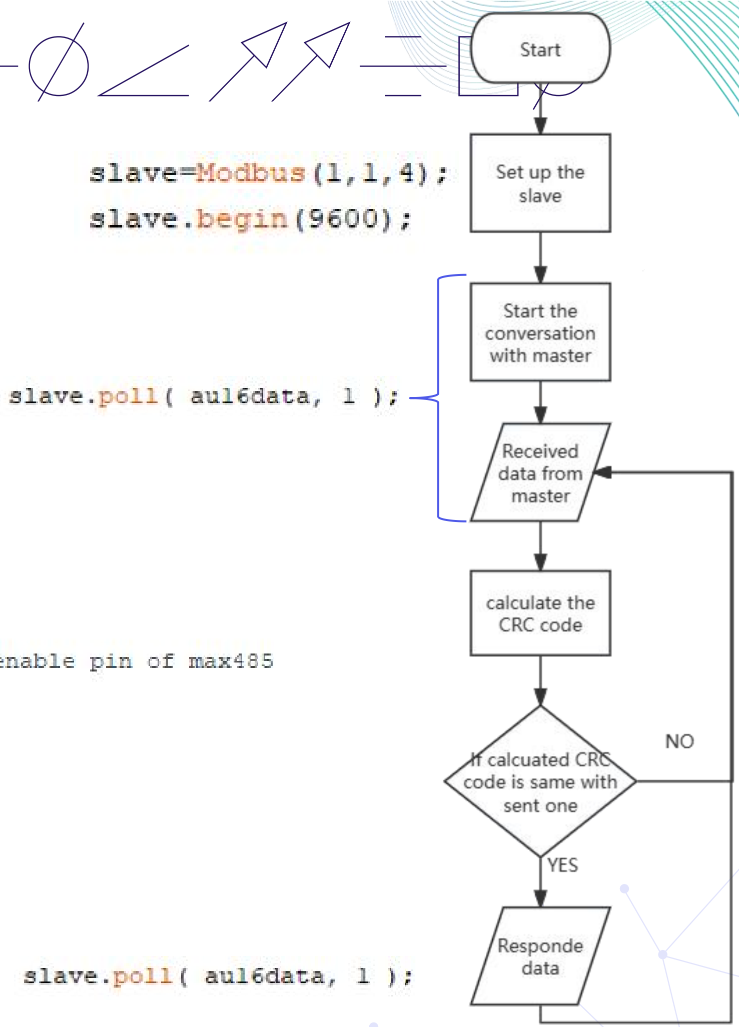
RS485





Pin Name	Pin connected
VCC	5V
A	A wire of end-effector
B	B wire of end-effector
GND	GND of Arduino
RO	RX pin of Arduino
RE	Pin 4
DE	Pin 4
DI	TX pin of Arduino

```
#include <ModbusRtu.h>

// data array for modbus network sharing
uint16_t au16data[1] = {
  3 };



/**
 *  Modbus object declaration
 *  u8id : node id = 0 for master, = 1..247 for slave
 *  port : serial port
 *  u8txenpin : 0 for RS-232 and USB-FTDI
 *               or any pin number > 1 for RS-485
 */
Modbus slave;

void setup() {
//set up the slave, 1 is ID and second 1 is RS485model,4 means the enable pin of max485
 slave=Modbus(1,1,4);
//start the slave and set baud rate
 slave.begin(9600);

}

void loop() {
  slave.poll( au16data, 1 );
  //begin the communication,1 is the length of the array
}
```


Start
Set up the slave
slave=Modbus(1,1,4);
slave.begin(9600);
Start the conversation with master
slave.poll(au16data, 1);
Received data from master
calculate the CRC code
If calcuated CRC code is same with sent one
NO
YES
Responde data
slave.poll(au16data, 1);

```
roslaunch xarm_bringup xarm7_server.launch robot_ip:=192.168.1.128 report_type:=normal
```

//Start all xarm server, change the control box ip in real condition

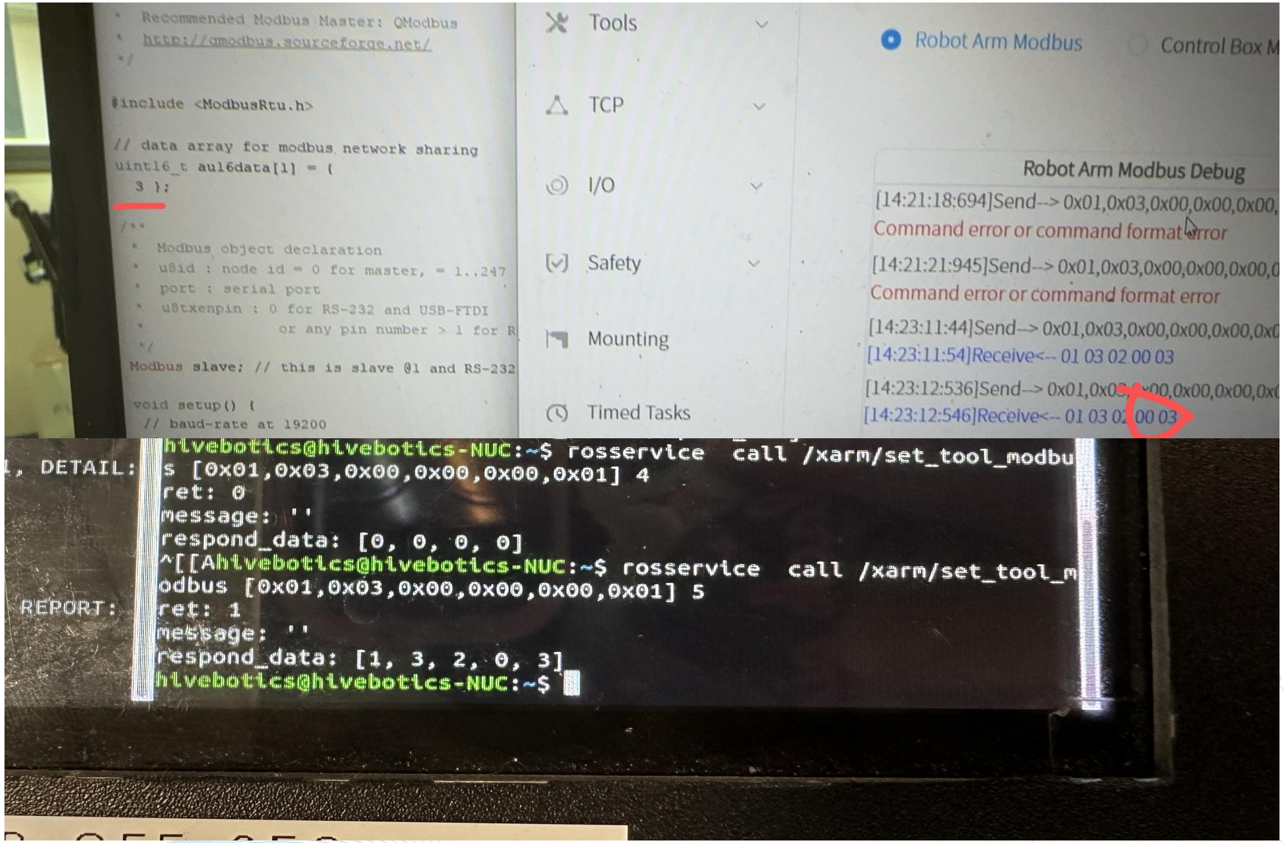
```
rosservice call /xarm/config_tool_modbus 9600 20
```

//Set proper baud rate and timeout(ms） parameters

```
rosservice call /xarm/set_tool_modbus [0x01,0x03,0x00,0x00,0x00,0x01] 5
```

//Set data array to be sent to the modbus tool device, and second is the number of characters to be received as a response from the device. No need to set CRC code

ROS—L



```
 *  Recommended Modbus Master: QModbus
 *  http://qmodbus.sourceforge.net/
 */

#include <ModbusRtu.h>

// data array for modbus network sharing
uint16_t au16data[1] = {
  3 };

/**
 *  Modbus object declaration
 *  u8id : node id = 0 for master, = 1..247
 *  port : serial port
 *  u8txenpin : 0 for RS-232 and USB-FTDI
 *               or any pin number > 1 for R
 */
Modbus slave; // this is slave @1 and RS-232

void setup() {
  // baud-rate at 19200
```

```
Robot Arm Modbus Debug
[14:21:18:694]Send--> 0x01,0x03,0x00,0x00,0x00,
Command error or command format error
[14:21:21:945]Send--> 0x01,0x03,0x00,0x00,0x00,0
Command error or command format error
[14:23:11:44]Send--> 0x01,0x03,0x00,0x00,0x00,0x0
[14:23:11:54]Receive<-- 01 03 02 00 03
[14:23:12:536]Send--> 0x01,0x03,0x00,0x00,0x00,0x0
[14:23:12:546]Receive<-- 01 03 02 00 03
```

```
, DETAIL:  hivebotics@hivebotics-NUC:~$ rosservice  call /xarm/set_tool_modbu
           s [0x01,0x03,0x00,0x00,0x00,0x01] 4
           ret: 0
           message: ''
           respond_data: [0, 0, 0, 0]
           ^[[Ahivebotics@hivebotics-NUC:~$ rosservice  call /xarm/set_tool_m
           odbus [0x01,0x03,0x00,0x00,0x00,0x01] 5
 REPORT:   ret: 1
           message: ''
           respond_data: [1, 3, 2, 0, 3]
           hivebotics@hivebotics-NUC:~$ 
```

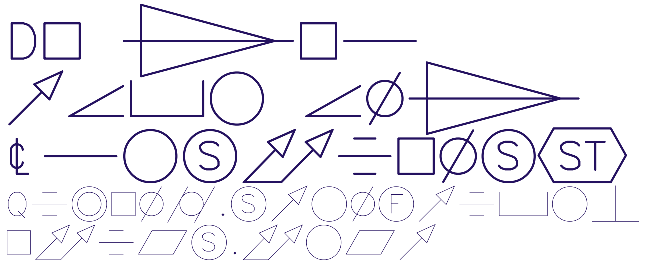
[illegible]

[illegible] RTU [illegible]

[illegible] CRC [illegible]

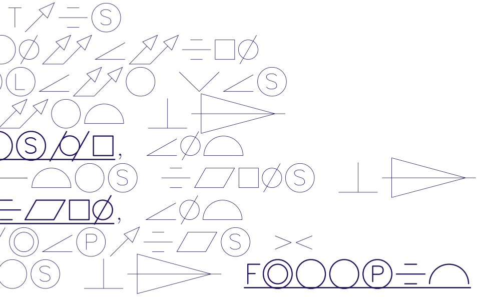
[illegible] RS-485 [illegible]

[illegible] RS-485 [illegible]



+65 888 550 49

CREDITS:



10

M L

L

[illegible]