PROJECT DESCRIPTION

**Project: IO BOX using modbus RTU**

TMA Solutions

TMA Tower

Quang Trung Software City - Hochiminh City

VIETNAM

<https://tmasolutions.com.vn/>

Table of Contents

[1 INTRODUCTION 2](#_Toc108700974)

[2 PURPOSE OF DOCUMENT 2](#_Toc108700975)

[3 DEVICE’S INFORMATION 3](#_Toc108700976)

1. READ VALUE
2. CONNECTION

[4 MANUAL 4](#_Toc108700979)

1. SET UP MODBUS POLL APPLICATION
2. TURN ON/OFF RELAY
3. READ STATE OF RELAYS
4. READ STATE OF INPUTS
5. READ STATE OF INPUTS BY PYTHON
6. READ STATE OF RELAYS BY PYTHON

[5 SCOPE OF THE WORK 11](#_Toc108700980)

[6 SYSTEM INFORMATION 11](#_Toc108700981)

# INTRODUCTION

IO BOX is 16 ports on/off controller which can read or write 16 IO ports. IO BOX controls through RS485 bus and uses Modbus RTU protocol. RS458 bus uses pair of wires increasing connection distance up to 1500m, controls up to 32 devices, via computer or IOT device.

Modbus RTU is an open serial protocol which derived from the Master/Slave architecture (Client/Server) developed by Modicon. It is a widely accepted serial level protocol due to its ease of use and reliability. Modbus RTU is widely used within Building Management Systems (BMS) and Industrial Automation Systems (IAS).

# PURPOSE OF DUCUMENT

This document outlines the introduce for IO BOX consisting of STM32F446RE, 16 inputs and 16 relays (ouputs)

Contained is an introduction to the context

* DEVICE’S INFORMATION
* MANUAL
* SCOPE OF THE WORK
* SYSTEM INFORMATION

# DEVICE’S INFORMATION

1. **READ VALUE**

Slave ID = 1 (can change id device at 539th line)

BaudRate = 115200,

StartBit:1,

DataBit:8,

CheckBit: none parity

StopBit:1.

1. **CONNECTION**

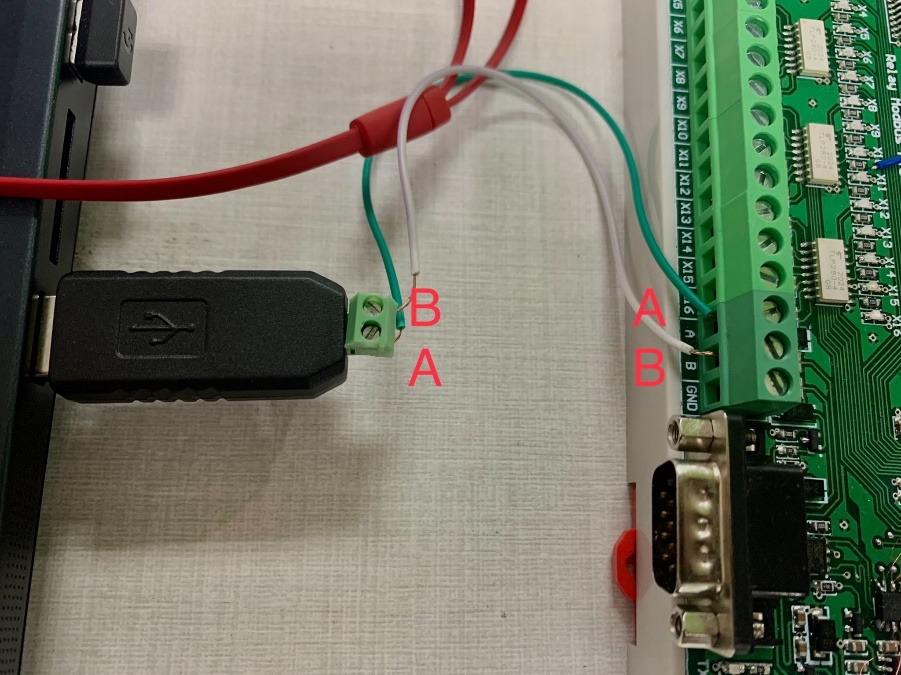


Figure 1. RS485 connection

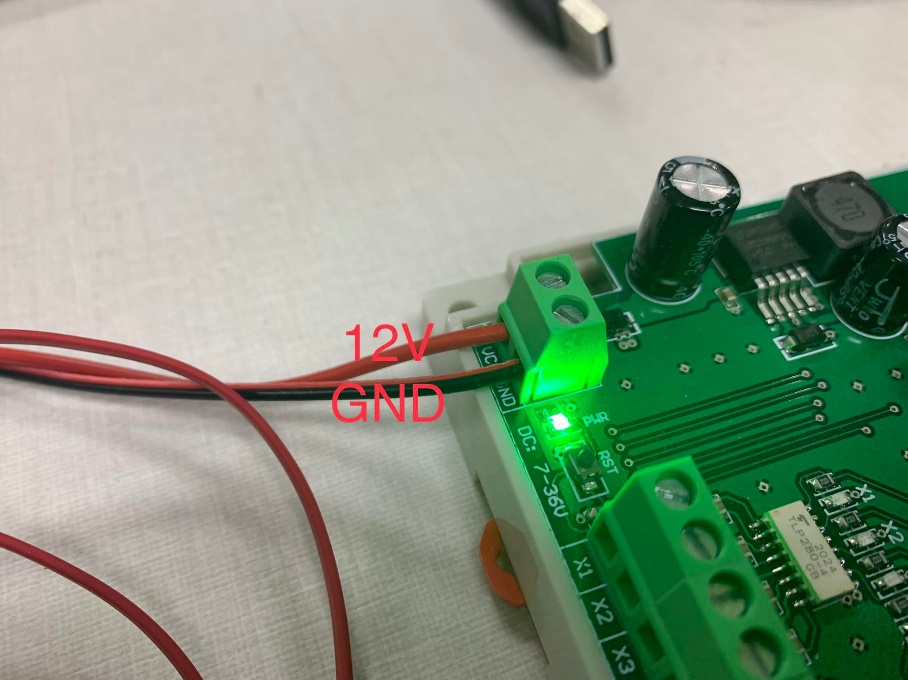


Figure 2. power supply connection

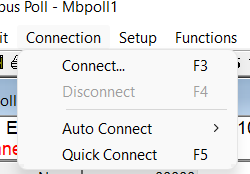
# MANUAL

1. **SETUP MODBUS POLL APPLICATION**

Step 1: Plug & connect devices as Figure 1;

Step 2: Open Modbus Poll Software (Down [here](https://www.modbustools.com/download.html))

Step 3: Click on Connection -> Connect



Step 4: Setting up -> OK

Graphical user interface, application

Description automatically generated

1. **TURN ON/OFF RELAY**

Step 1: Function -> Write single coil (0x05)

Step 2: set relay at 5th address to turn on

Graphical user interface, application

Description automatically generated

Step 2: set relay at 5th address to turn off

Graphical user interface, application

Description automatically generated

1. **READ STATE OF RELAYS**

Step 1: Setup -> Read/Write definition

Step 2: set function Read Coils (0x01) and quantity for 16 relays

Graphical user interface

Description automatically generated

Step 3: Follow the table to take the result.

Table

Description automatically generated

Figure 3. All relays off

Table

Description automatically generated

Figure 4. Relay which have 5th address,is on

1. **READ STATE OF INPUTS**

Step 1: Setup -> Read/Write definition

Step 2: set function Read Coils (0x02) and quantity for 16 inputs

Graphical user interface

Description automatically generated

Step 3: Follow the table to take the result.

Table

Description automatically generated

Figure 6. All inputs is high

Table

Description automatically generated

Figure 7. Input which have 1st address is toggle

1. **READ STATE OF INPUTS BY PYTHON**

Step 1: Open vscode to run python code

Step 2: Check library and set up COM before running code

Text

Description automatically generated

Figure 8. All library in python

Step 3: Using read\_discrete\_inputs function to read the state of 16 inputs

A picture containing shape

Description automatically generated

Figure 9. All inputs is high

Text

Description automatically generated

Figure 10. Input which have 1st address is toggle

1. **READ STATE OF RELAY BY PYTHON**

Step 1: Open vscode to run python code

Step 2: Check library and set up COM before running code

Text

Description automatically generated

Figure 11. All library in python

Step 3: Using write\_coil function to turn on (255)/off (0) relay and read\_coils function to read the state of relays

Text

Description automatically generated

Text

Description automatically generated

Figure 12. Turn on the relay which have 1st address

Text

Description automatically generated

Graphical user interface

Description automatically generated with medium confidence

Figure 12. Turn off the relay which have 1st address

# SCOPE OF THE WORK

* C language
* Python Language
* Stm32 cube and keil C

# SYSTEM INFORMATION

*IoT Server:* [*https://iot.innovation.com.vn*](https://iot.innovation.com.vn)

*User: huynguyen231000@gmail.com*