**ProjectManager新增通讯设备**

修订历史记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **日期** | **版本** | **作者** | **审核者** | **说明** |
| 2019-04-08 | V1.0 | VelsonWang |  | 初稿 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**目 录**

目录

[1 新建ModbusRTU插件工程 3](#_Toc6584247)

[1.1 新建插件工程 3](#_Toc6584248)

[1.2 ModbusRTU\_dependencies.pri文件说明 3](#_Toc6584249)

[1.3 ModbusRTU.pro文件说明 3](#_Toc6584250)

[1.4 插件接口说明 4](#_Toc6584251)

[1.5 实现ModbusRTU插件 4](#_Toc6584252)

[1.6 编译插件工程 7](#_Toc6584253)

[2 ProjectManager相关配置 7](#_Toc6584254)

[2.1 配置文件 7](#_Toc6584255)

[2.2 ModbusRTU配置 9](#_Toc6584256)

[3 ProjectManager插件相关源码和UI 10](#_Toc6584257)

本文以ModbusRTU为例说明如何在ProjectManager新增通讯设备。本文仅供参考，具体请看实现源码。

# 1 新建ModbusRTU插件工程

## 新建插件工程

在{文件QStudioSCADA.pro所在目录} \Devices新建ModbusRTU目录, 在ModbusRTU目录新建如下文件:

ModbusRTU.pro

ModbusRTU.cpp,

ModbusRTU.h

ModbusRTU.json

ModbusRTU\_dependencies.pri

文件内容可能有改动，详细内容请阅读源码文件！！

## ModbusRTU\_dependencies.pri文件说明

文件内容：

QTC\_PLUGIN\_NAME = ModbusRTU

QTC\_LIB\_DEPENDS +=

QTC\_PLUGIN\_DEPENDS +=

其中变量“QTC\_PLUGIN\_NAME”的值为生成的插件名称。最后生成的插件位于目录：QStudioSCADABin\deviceplugins

## ModbusRTU.pro文件说明

文件内容：

include(../../QStudioSCADADevicePlugin.pri)

SOURCES += ModbusRTU.cpp

HEADERS += ModbusRTU.h

DISTFILES += ModbusRTU.json

其中文件“QStudioSCADADevicePlugin.pri”指定了工程模板，配置，依赖等。

## 插件接口说明

插件接口{文件QStudioSCADA.pro所在目录} \Devices\IDevicePlugin\IDevicePlugin.h

文件内容：

#ifndef IDEVICEPLUGIN\_H

#define IDEVICEPLUGIN\_H

#include <QStringList>

/\*

\* 注意本类不要派生自QObject

\*/

class IDevicePlugin

{

public:

virtual ~IDevicePlugin() {}

// 获取设备类型名称

virtual QString GetDeviceTypeName() = 0;

// 获取设备支持的所有协议

virtual QStringList GetDeviceSupportProtocol() = 0;

// 获取设备支持的所有寄存器区

virtual QStringList GetDeviceSupportRegisterArea() = 0;

// 获取设备支持的所有数据类型

virtual QStringList GetDeviceSupportDataType() = 0;

};

QT\_BEGIN\_NAMESPACE

#define DevicePluginInterface\_iid "QStudioSCADA.Device.PluginInterface"

Q\_DECLARE\_INTERFACE(IDevicePlugin, DevicePluginInterface\_iid)

QT\_END\_NAMESPACE

#endif // IDEVICEPLUGIN\_H

## 实现ModbusRTU插件

新建ModbusRTU实现类，继承类QObject ，类IDevicePlugin实现插件接口。

文件ModbusRTU.h内容如下：

#ifndef MODBUSRTU\_H

#define MODBUSRTU\_H

#include <QObject>

#include "../IDevicePlugin/IDevicePlugin.h"

class ModbusRTU : public QObject, IDevicePlugin

{

Q\_OBJECT

Q\_PLUGIN\_METADATA(IID DevicePluginInterface\_iid FILE "ModbusRTU.json")

Q\_INTERFACES(IDevicePlugin)

public:

ModbusRTU();

// 获取设备类型名称

QString GetDeviceTypeName();

// 获取设备支持的所有协议

QStringList GetDeviceSupportProtocol() Q\_DECL\_OVERRIDE;

// 获取设备支持的所有寄存器区

QStringList GetDeviceSupportRegisterArea() Q\_DECL\_OVERRIDE;

// 获取设备支持的所有数据类型

QStringList GetDeviceSupportDataType() Q\_DECL\_OVERRIDE;

};

#endif // MODBUSRTU\_H

文件ModbusRTU.cpp内容如下：

#include "ModbusRTU.h"

ModbusRTU::ModbusRTU()

{

}

/\*

\* 获取设备类型名称

\*/

QString ModbusRTU::GetDeviceTypeName()

{

return QString(tr("串口设备"));

}

/\*

\* 获取设备支持的所有协议

\*/

QStringList ModbusRTU::GetDeviceSupportProtocol()

{

QStringList list;

list<<"ModbusRTU";

return list;

}

/\*

\* 获取设备支持的所有寄存器区

\*/

QStringList ModbusRTU::GetDeviceSupportRegisterArea()

{

QStringList list;

list<<"DO线圈";

list<<"DI离散输入寄存器";

list<<"AO保持寄存器";

list<<"AI输入寄存器";

return list;

}

/\*

\* 获取设备支持的所有数据类型

\*/

QStringList ModbusRTU::GetDeviceSupportDataType()

{

QStringList list;

list<<"Bit1开关量";

list<<"Char8位有符号数";

list<<"Byte8位无符号数";

list<<"Short16位有符号数";

list<<"Word16位无符号数";

list<<"ASCII2个字符";

list<<"Long32位有符号数";

list<<"Dword32位无符号数";

list<<"Float单精度浮点数";

list<<"String字符串";

list<<"Double双精度浮点数";

list<<"BCD";

//list<<"LongLong64位有符号数";

//list<<"DwordDword64位无符号数";

return list;

}

## 编译插件工程

编译ModbusRTU插件工程，生成的ModbusRTU.dll文件位于目录QStudioSCADABin\deviceplugins\ModbusRTU.dll

# ProjectManager相关配置

## 2.1 配置文件

修改配置文件QStudioSCADABin\bin\Config\communication\_device.ini

配置文件内容如下：

####################################

## 设备列表

[DeviceSupportList]

list-count=6 **#配置项个数**

list-1=GenDataProgram-模拟数据程序

list-2=IOModule-IO模块

list-3=PlcDevice-PLC

**list-4=ModbusDevice-Modbus设备**

list-5=GaugeDevice-仪表设备

list-6=CustomDevice-用户自定义设备

####################################

# 模拟数据程序

[GenDataProgram]

list-count=0

list-1=

######################################

# IO模块

[IOModule]

list-count=0

list-1=

######################################

# PLC设备

[PlcDevice]

list-count=1

list-1=SIEMENS

# SIEMENS-PLC设备

[SIEMENS]

list-count=1

list-1=S7\_200-COM

######################################

# Modbus设备

[ModbusDevice]

list-count=2

list-1=SerialModbus

list-2=TCPIPModbus

# Serial\_Modbus设备

# COM, NET, BUS, OPC

[SerialModbus]

list-count=4

list-1=ModbusRTU\_slave-COM

list-2=ModbusRTU-COM

list-3=ModbusASCII\_slave-COM

list-4=ModbusASCII-COM

# TcpIP\_Modbus设备

[TCPIPModbus]

list-count=2

list-1=TCPIPModbus\_slave-NET

list-2=TCPIPModbus-NET

######################################

# 仪表设备

[GaugeDevice]

list-count=0

list-1=

######################################

# 用户自定义设备

[CustomDevice]

list-count=0

list-1=

## 2.2 ModbusRTU配置

ModbusRTU属于“ModbusDevice-Modbus设备”设备类中的“ModbusDevice”类别中的“SerialModbus”类，在“SerialModbus”类中ModbusRTU属于串口通信类别的，顾取值为“ModbusRTU-COM”。

####################################

## 设备列表

[DeviceSupportList]

list-count=配置项数量

list-第几项=ModbusDevice-Modbus设备

######################################

# Modbus设备

[ModbusDevice]

list-count=配置项数量

list-第几项=SerialModbus

######################################

# Serial\_Modbus设备

# COM, NET, BUS, OPC

[SerialModbus]

list-count=配置项数量

list-第几项=ModbusRTU-COM

# ProjectManager插件相关源码和UI

文件NewComDeviceDialog.cpp

文件VariableEditDialog.cpp



