

基于 SIMATIC IOT2040 的 Node-red S7 通信

1. 主要目标

基于 Node-red S7 通信，实现 SIMATIC IOT2040 读写 Siemens S7-300/400/1200/1500 或 S7-200 SMART PLC 中的数据。

2. Node-red 及 S7 节点介绍

2.1 Node-red

Node-red 是一种用于以新的有趣的方式将硬件设备、API 和在线服务连接在一起的编程工具。

它提供了一个基于浏览器的编辑器，可以很容易地使用节点栏中的大量节点来组成流，这些节点可以通过一次点击完成部署。

2.2 Node-red S7 节点

S7 节点是一个允许使用西门子 S7 以太网协议 RFC1006 与 S7-300/400/1200/1500 PLC 通信的库。同时，它也适用于 S7-200 SMART PLC。

2.3 安装 Node-red 和 S7 节点

SIMATIC IOT2040 的示例镜像已经预装了 Node-red，而用于 S7 通信的 S7 节点需要用户自行安装，安装时需要连接网络。

安装之前，需要确认 Node-red 的安装目录，本文 Node-red 的安装目录为 `/usr/lib/node_modules`。

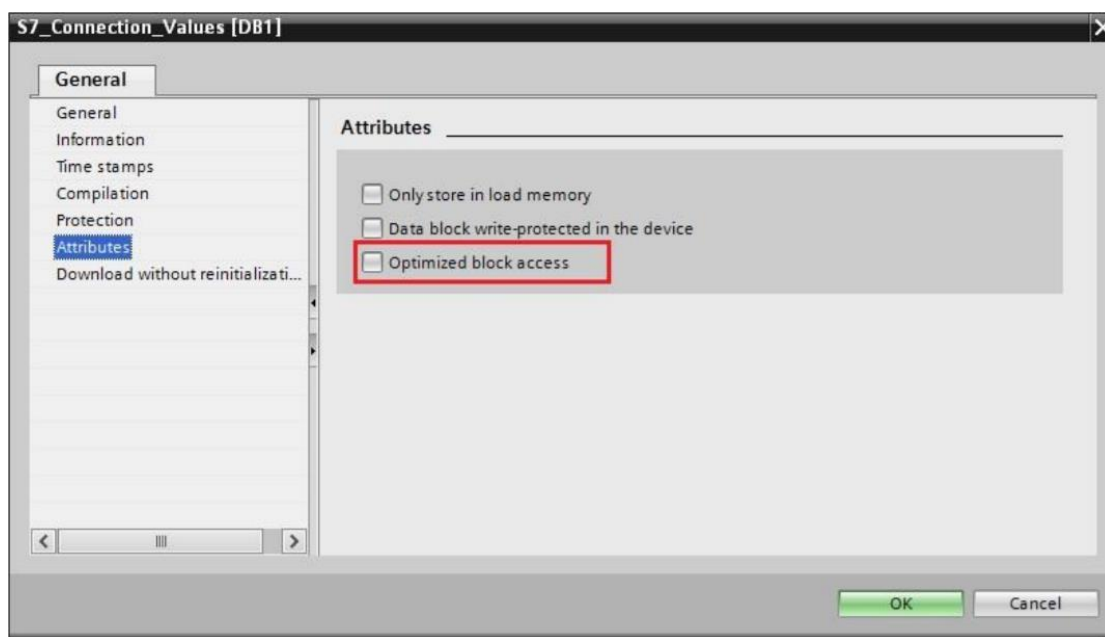
- (1) 打开 putty，建立与 IOT2040 的连接。
- (2) 输入指令 `cd /usr/lib/node_modules`，切换至 Node-red 安装目录。
- (3) 输入指令 `npm install node-red-contrib-s7`，安装 S7 节点。

```
COM3 - PuTTY
root@iot2000:/# cd /usr/lib/node_modules/
root@iot2000:/usr/lib/node_modules# npm install node-red-contrib-s7
npm WARN unmet dependency /usr/lib/node_modules/node-red/node_modules/node-red-n
ode-serialport/node_modules/serialport/node_modules/tar-pack requires debug@'^2.
2.0' but will load
npm WARN unmet dependency /usr/lib/node_modules/node-red/node_modules/node-red-n
ode-serialport/node_modules/serialport/node_modules/debug,
npm WARN unmet dependency which is version 2.3.3
node-red-contrib-s7@0.2.2 node-red-contrib-s7
└─ nodes7@0.1.11
root@iot2000:/usr/lib/node_modules#
```

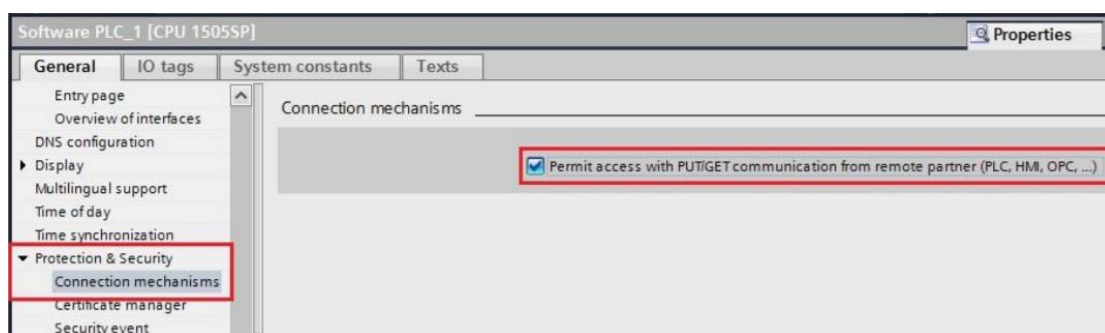
3. 编辑 Node-red S7 通信程序

3.1 S7 通信准备工作

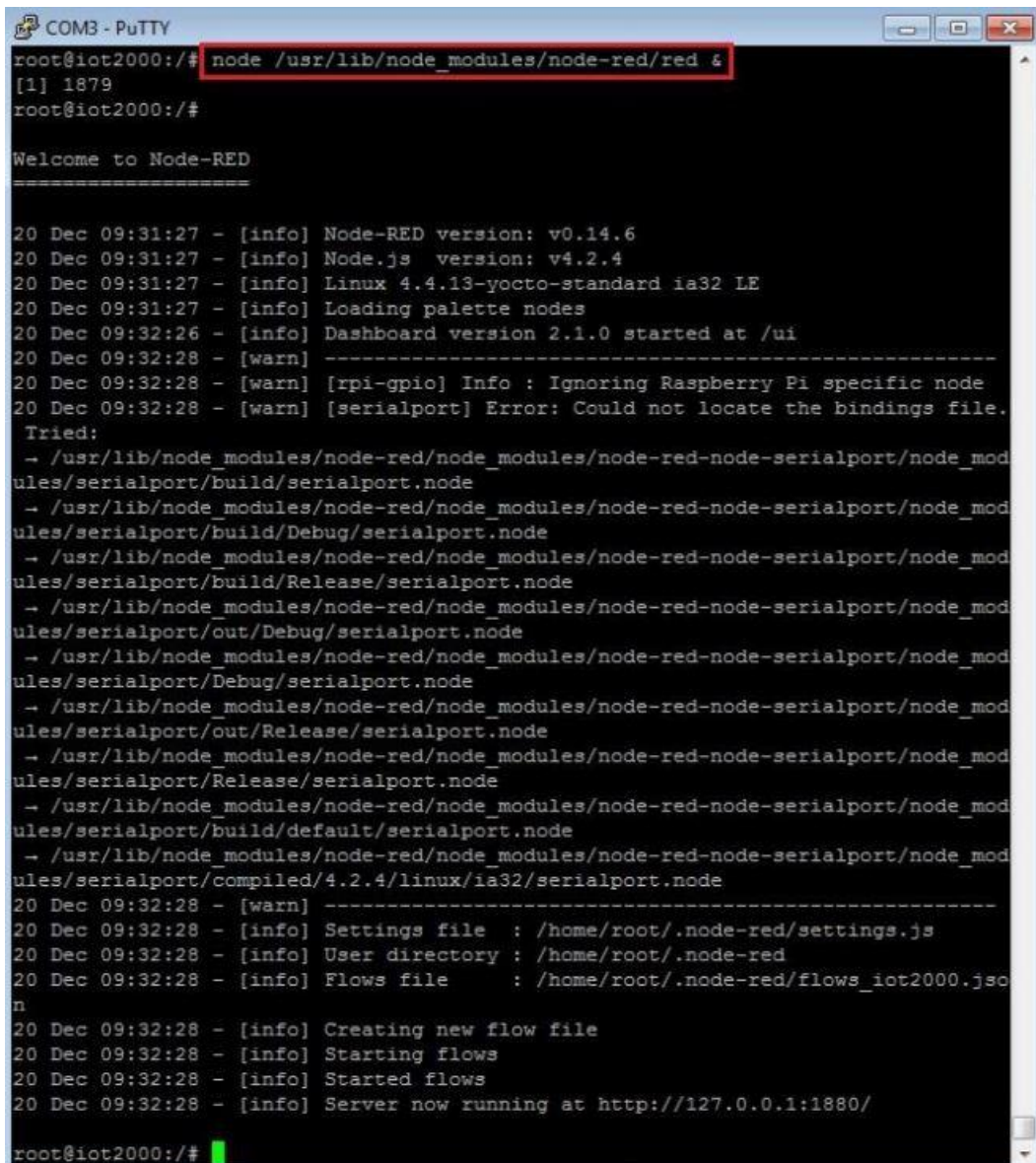
- (1) 只可以使用没有优化的数据块，可以通过 TIA-Portal 在数据块的属性中进行设置。
请确保**没有**勾选该选项。



- (2) 必须在 S7-300/400/1200/1500 PLC CPU 配置中启用 PUT/GET 通信，请确保**勾选**该选项。



- (3) 在 putty 中输入指令 `node /usr/lib/node_modules/node-red/red &`，启动 Node-red。



```
root@iot2000:/# node /usr/lib/node_modules/node-red/red &
[1] 1879
root@iot2000:/#

Welcome to Node-RED
=====

20 Dec 09:31:27 - [info] Node-RED version: v0.14.6
20 Dec 09:31:27 - [info] Node.js version: v4.2.4
20 Dec 09:31:27 - [info] Linux 4.4.13-yocto-standard ia32 LE
20 Dec 09:31:27 - [info] Loading palette nodes
20 Dec 09:32:26 - [info] Dashboard version 2.1.0 started at /ui
20 Dec 09:32:28 - [warn] -----
20 Dec 09:32:28 - [warn] [rpi-gpio] Info : Ignoring Raspberry Pi specific node
20 Dec 09:32:28 - [warn] [serialport] Error: Could not locate the bindings file.
Tried:
  - /usr/lib/node_modules/node-red/node_modules/node-red-node-serialport/node_modules/serialport/build/serialport.node
  - /usr/lib/node_modules/node-red/node_modules/node-red-node-serialport/node_modules/serialport/build/Debug/serialport.node
  - /usr/lib/node_modules/node-red/node_modules/node-red-node-serialport/node_modules/serialport/build/Release/serialport.node
  - /usr/lib/node_modules/node-red/node_modules/node-red-node-serialport/node_modules/serialport/out/Debug/serialport.node
  - /usr/lib/node_modules/node-red/node_modules/node-red-node-serialport/node_modules/serialport/Debug/serialport.node
  - /usr/lib/node_modules/node-red/node_modules/node-red-node-serialport/node_modules/serialport/out/Release/serialport.node
  - /usr/lib/node_modules/node-red/node_modules/node-red-node-serialport/node_modules/serialport/Release/serialport.node
  - /usr/lib/node_modules/node-red/node_modules/node-red-node-serialport/node_modules/serialport/build/default/serialport.node
  - /usr/lib/node_modules/node-red/node_modules/node-red-node-serialport/node_modules/serialport/compiled/4.2.4/linux/ia32/serialport.node
20 Dec 09:32:28 - [warn] -----
20 Dec 09:32:28 - [info] Settings file : /home/root/.node-red/settings.js
20 Dec 09:32:28 - [info] User directory : /home/root/.node-red
20 Dec 09:32:28 - [info] Flows file : /home/root/.node-red/flows_iot2000.json
20 Dec 09:32:28 - [info] Creating new flow file
20 Dec 09:32:28 - [info] Starting flows
20 Dec 09:32:28 - [info] Started flows
20 Dec 09:32:28 - [info] Server now running at http://127.0.0.1:1880/

root@iot2000:/#
```

(4) 在浏览器中打开 IOT2040 IP 地址的 1880 端口 (例如 <http://192.168.200.1:1880>)。



3.2 编辑 S7 in 节点

利用“s7 in”节点可以基于 S7 协议从 S7 PLC 读取数据。

(1) 在左侧节点栏中选择“s7 in”节点，拖动添加至编辑区域。



(2) 双击“s7 in”节点。将 Mode 选为 All variables。点击图中图标编辑 S7 端点。


Edit s7 in node

Delete

Cancel

Done

PLC

Add new s7 endpoint... 

Mode

All variables ▼

☒ Emit only when value changes (diff)

Name

Name

(3) 配置 S7 端点的连接信息。

S7 端点的连接信息包括 S7 PLC 的 IP 地址、端口号、机架号、槽号、读取周期等信息。其中，默认的端口号是 102。不同的 S7 PLC，对应的机架号与槽号也不同。

s7 in > Add new s7 endpoint config node

Cancel

Add

Connection

Variables

IP Address

192.168.200.10

Port

102

Mode

Rack/Slot ▼

Rack

0

Slot

1

Cycle time

500

ms

Timeout

1500

ms

Debug

Default (command line) ▼

Name

Name

(4) 配置 S7 端点的变量信息列表。

S7 端点的变量信息包括变量的寻址方式以及变量名称。使用“+Add”按钮来添加新的变量，“Export”按钮可以将变量列表导出至 .csv 文件中，“Import”按钮可以通过 .csv 文件导入变量列表。

s7 in > Add new s7 endpoint config node

Cancel Add

Connection Variables

Variable list

Address	Name
---------	------

+Add Remove all Import Export

(5) 示例

STEP 7-Micro/WIN SMART 中 DB 块数据（以 S7-200 SMART SR60 为例）：

数据块	
VD0	-22
VB4	0
VB5	
VB6	'C'
VB8	"abc"

Node-red 中对应的部分变量信息列表：

Variable list

Q0.1	Q	✕
MB0	MB	✕
DB1,B4	AC	✕
DB1,C6	CHAR	✕
DB1,S7.3	STRING	✕

+ Add
Remove all
Import
Export

TIA Portal 中 DB 块数据（以 S7-1500 为例）：

项目1
PLC_1 [CPU 1511-1 PN]
程序块
数据块_1 [DB1]

数据块_1

	名称	数据类型	偏移量	启动值	保持性	可从 HMI ...	在 HMI ...	设置值	注释
1	Static				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
2	Temperature	Real	0.0	20.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
3	Auto Count	Byte	4.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
4	Check	Bool	5.0	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
5	Char	Char	6.0	'c'	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
6	String	String	8.0	'abc'	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Node-red 中对应的部分变量信息列表：

Variable list

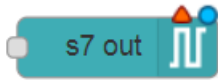
Q0.1	Q	✕
MB0	MB	✕
DB1,B4	AC	✕
DB1,C6	CHAR	✕
DB1,S8.3	STRING	✕

+ Add
Remove all
Import
Export

3.3 编辑 S7 out 节点

利用“s7 out”节点可以基于 S7 协议向 S7 PLC 写入数据。

(1) 在左侧节点栏中选择“s7 out”节点，拖动添加至编辑区域。



(2) 双击“s7 out”节点，编辑节点。

Edit s7 out node

Delete

Cancel

Done

⚡ PLC

Add new s7 endpoint... ▾

🔄 Variable

Select a variable ▾

🏷️ Name

Name

Caution when writing data to production PLCs!

(3) 如“s7 in”节点已添加过 S7 端点，则直接选择相应 S7 端点即可；如未添加过，请参考“s7 in”节点添加 S7 端点过程进行添加。

Edit s7 out node

Delete

Cancel

Done

⚡ PLC

Add new s7 endpoint... ▾

192.168.2.1:102:0:1

192.168.200.10:102:0:1

Add new s7 endpoint...

🔄 Variable

Select a variable ▾

🏷️ Name

Name

Caution when writing data to production PLCs!

(4) 通过下拉菜单选择要写入的变量，每个“s7 out”节点只能对一个变量进行写入操作。

Edit s7 out node

Delete
Cancel
Done

⚡ PLC
192.168.200.10:102:0:1

🔄 Variable
Select a variable

📁 Name

Q
MB
AC
CHAR
STRING

Caution when v

在选择完成后，点击“Done”按钮确定。

Edit s7 out node

Delete
Cancel
Done

⚡ PLC
192.168.200.10:102:0:1

🔄 Variable

Q
Q0.1

📁 Name
Name

Caution when writing data to production PLCs!

说明：

- 1) S7-300/400/1200/1500 不可对 I 区进行写入
- 2) S7-200 Smart 不可对 I 区及 Q 区写入
- 3) 待写入的变量只可从已输入的变量列表中选择

4. S7 PLC 数据寻址方式及 Node-red 对应寻址方式

下表给出了 S7 PLC 中数据寻址方式及 Node-red 对应寻址方式。

由于 CPU 存储方式的不同及 Node-red S7 节点限制，有些数据无法直接读写，但是可以将它们转化为其他数据区域中的变量进行读写。

4.1 S7-300/400/1200/1500

	数据区域	数据类型	PLC 寻址方式	Node-red 寻址方式
PLC 变量	I	Bool	Ix.y	Ix.y
		Byte	IBx	IBx
		Char	IBx	ICx
		Word	IWx	IWx
		Int	IWx	IIx
		DWord	IDx	IDx
		DInt	IDx	IDIx
		Real	IRx	IRx
	Q	Bool	Qx.y	Qx.y
		Byte	QBx	QBx
		Char	QBx	QCx
		Word	QWx	QWx
		Int	QWx	QIx
		DWord	QDx	QDx
		DInt	QDx	QDIx
		Real	QRx	QRx
	M	Bool	Mx.y	Mx.y
		Byte	MBx	MBx
		Char	MBx	MCx
		Word	MWx	MWx
		Int	MWx	MIx
		DWord	MDx	MDx
		DInt	MDx	MDIx
		Real	MRx	MRx
用户数据块	DB	Bool	DBn.DBXx.y	DBn,Xx.y
		Byte	DBn.DBBx	DBn,Bx / DBn,BYTEx
		Char	DBn.DBBx	DBn,Cx / DBn,CHARx
		Word	DBn.DBWx	DBn,WORDx
		Int	DBn.DBWx	DBn,Ix / DBn,INTx
		DWord	DBn.DB Dx	DBn,DWx / DBn,DWORDx
		DInt	DBn.DB Dx	DBn,DIx / DBn,DINTx

		Real	DBn.DB Dx	DBn,Rx / DBn,REALx
		String		DBn,Sx.length

说明：

- 1) n 为数据区域偏移量
- 2) x 为数据字节偏移量
- 3) y 为数据位偏移量
- 4) 将“String”数据类型 Node-red 寻址方式中的 length 替换为字符串长度

4.2 S7-200 SMART

	数据区域	数据类型	PLC 寻址方式	Node-red 寻址方式
PLC 变量	I	Bool	Ix.y	Ix.y
		Byte	IBx	IBx
		Char	IBx	ICx
		Word	IWx	IWx
		Int	IWx	IIx
		DWord	IDx	IDx
		DInt	IDx	IDIx
		Real	IRx	IRx
	Q	Bool	Qx.y	Qx.y
		Byte	QBx	QBx
		Char	QBx	QCx
		Word	QWx	QWx
		Int	QWx	QIx
		DWord	QDx	QDx
		DInt	QDx	QDIx
		Real	QRx	QRx
	M	Bool	Mx.y	Mx.y
		Byte	MBx	MBx
		Char	MBx	MCx
		Word	MWx	MWx
		Int	MWx	MIx
		DWord	MDx	MDx
		DInt	MDx	MDIx
		Real	MRx	MRx
用户数据块	V	Bool	Vx.y	DB1,Xx.y
		Byte	VBx	DB1,Bx / DB1,BYTEx
		Char	VBx	DB1,Cx / DB1,CHARx
		Word	VWx	DB1,WORDx
		Int	VWx	DB1,Ix / DB1,INTx
		DWord	VDx	DB1,DWx / DB1,DWORDx
		DInt	VDx	DB1,DIx / DB1,DINTx

		Real	VDx	DB1,Rx / DB1,REALx
		String		DB1,S(x-1).length

说明：

- 1) n 为数据区域偏移量
- 2) x 为数据字节偏移量
- 3) y 为数据位偏移量
- 4) 将“String”数据类型 Node-red 寻址方式中的 length 替换为字符串长度，数据字节偏移量 x 需大于 0