

<b>RFH6xx _PROFINET 中文调试手册 TIA 博途版</b>			
部门:	产品管理部	类型:	调试指导
编写人:	Carlos Wang	日期:	2017-5-26
版本:	V1.0		



西克中国  
官方微信



西克中国  
微官网



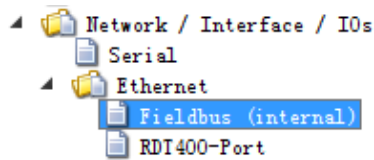
## 目录

1. SOPAS 调试软件设置 .....	3
1.1. 激活 Profinet 功能 .....	3
1.2. 配置 IP 地址 .....	3
2. TIA 博图配置 .....	4
2.1. 安装 GSDML 文件 .....	4
2.2. 硬件组态 .....	5
3. 开始数据通讯 .....	8
3.1. 设置 SOPAS 触发方式 .....	8
3.2. 首先利用串口调试工具进行读取测试 .....	8
3.3. 监控 PLC 变量表 .....	8
3.4. 以下为握手模式的通讯说明: .....	9
4. 安装 SICK FB 功能块 .....	12
4.1. 安装 FB 功能块 .....	12
4.2. 添加功能块 .....	13
5. 功能块介绍 .....	15
5.1. “HWInputIdent” .....	15
5.2. “HWInputlength” 和 “HWOutputlength” .....	16
5.3. “Tout” .....	16
5.4. “CANId” .....	16
5.5. 接下去的都为控制字部分: .....	16
5.6. 控制字的时序图 .....	17
6. 使用功能块 .....	18
6.1. SOPAS 配置: .....	18
6.2. 读取标签 UID .....	19
6.3. 读取 BLOCK 中的值 .....	20
6.4. 通过功能块写对应的数据 .....	22

## 1. SOPAS 调试软件设置

### 1.1. 激活 Profinet 功能

输入设备名称，并且需要和 PLC 内的一致：



#### Ethernet fieldbus selection

Fieldbus type:

**Note:** Button "Apply" saves Parameter permanently and reboots the device.  
Only necessary when deactivating Profinet.

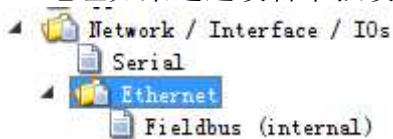
#### Network Options

Device ID:  Device Name:

注意事项: Device Name 是关键, 在西门子博途中的设置必须和 SOPAS 中的名字一致, 如果不一致会导致 PLC 硬件报错红灯闪烁, 系统无法建立 PN 连接。

### 1.2. 配置 IP 地址

IP 地址如果通过设备单独设置，那么设置成与 PLC 同一个网段。



#### General

Addressing Mode:

IP-Address:

Subnet-Mask:

Default Gateway:

## 2. TIA 博图配置

### 2.1. 安装 GSDML 文件

2.1.1. 打开 SICK 官网，下载对应产品对应的 GSDML 文件和 FB 功能块；

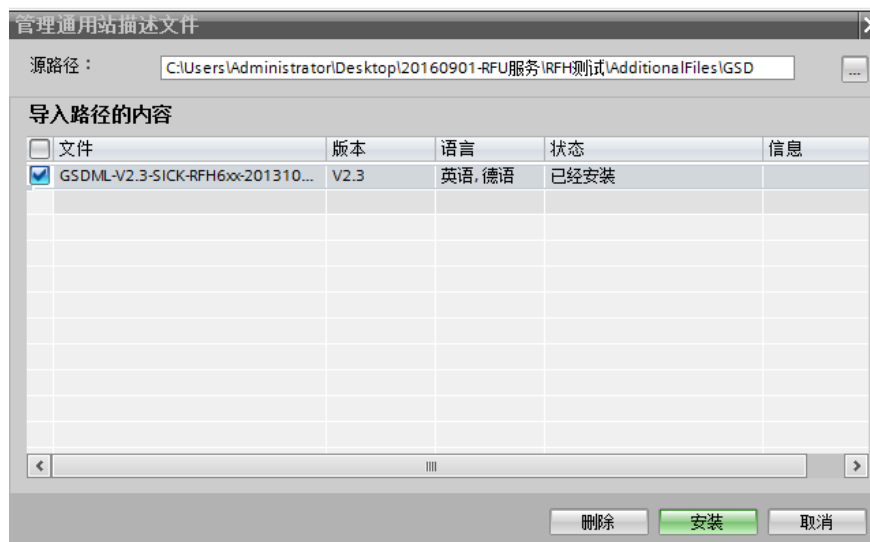
<https://www.sick.com/cn/zh/>

本测试采用的是基于 TIA(博途)的西门子 S7-1200，下载链接如下：

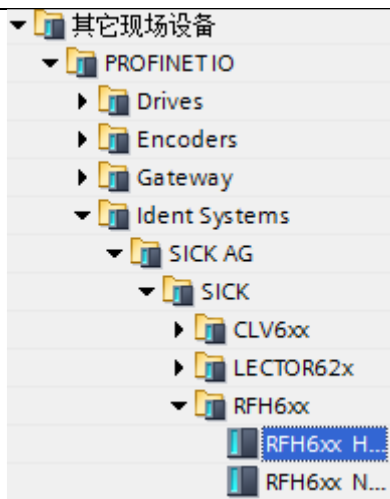


2.1.2. 安装 GSDML 文件，添加 ProfiNet 设备到网络上；

在工具栏里找到“选项”→“管理通用站描述文件”，将先前下载的 GSDML 文件导入。



2.1.3. 安装完成后，可以在右侧栏的硬件目录中的“其他外部设备”中找到 RFH 的设备：



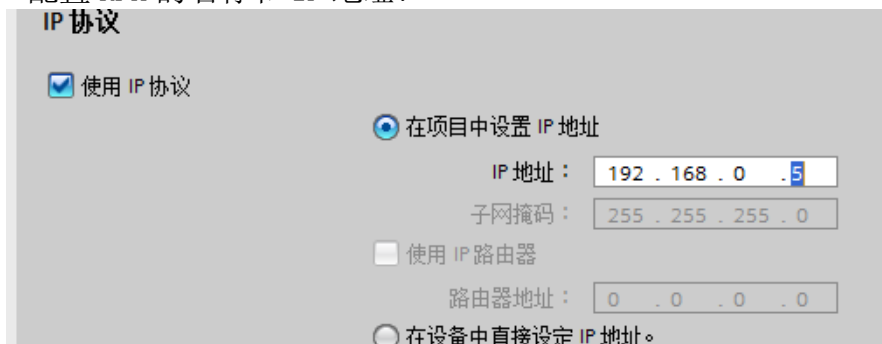
## 2.2. 硬件组态

2.2.1. 将使用的 SICK 产品拖放到组态窗口中，并且进行连接：



注意：如果使用 SICK 功能块，必须使用握手模式：Handshake (HS)。此模式下，每次数据传输都需要 PLC 进行回应；支持长数据的快速传输；非握手模式，通常用于短数据传输和通讯测试。

2.2.2. 配置 RFH 的名称和 IP 地址：



不要自动生成 PROFINET 设备名称, 把 SOPAS 的设备名字填写在下面

**PROFINET**

自动生成 PROFINET 设备名称

PROFINET 设备名称: rfh6xx

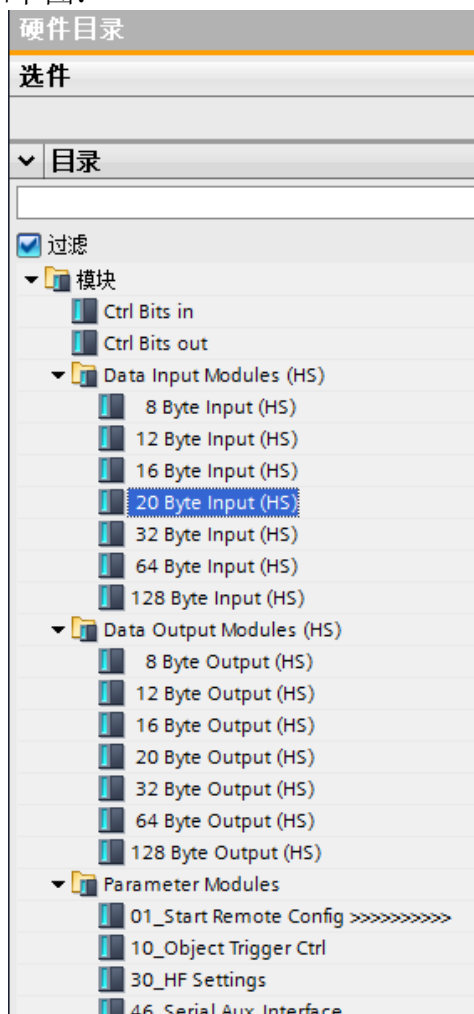
转换的名称: rfh6xx

设备编号: 1

完成后，可以在左侧看到设置好的设备：



2.2.3. 在硬件目录中的模块配置双击配置到 RFH 下，添加控制字输入输出、数据输入输出和感兴趣的参数设置，如下图：



2.2.4. Ctrl bit in/out 控制字节的详细功能如下表所示：  
此功能可选：  
PLC 输入控制字：

**Control Word In (Ctrl Bits in) is assigned as follows:**

Bit	Assignment	Name	Comment
D0	Fixed	Device Ready	State of the CLV6xx bar code scanner
D1	Fixed	System Ready	Not implemented for the CLV6xx bar code scanner
D2	Fixed	Good Read	State of the reading result
D3	Fixed	No Read	State of the reading result
D4	Fixed	External output 1	OUT 1 output level of the field bus module
D5	Fixed	External output 2	OUT 2 output level of the field bus module
D6	Fixed	Result 1	Result 1 output level of the bar code scanner
D7	Fixed	Result 2	Result 2 output level of the bar code scanner
D8	Fixed	External input 1	IN 1 input level of the field bus module
D9	Fixed	External input 2	IN 2 input level of the field bus module
D10	Fixed	Sensor 1	Sensor 1 input level of the bar code scanner
D11	Fixed	Sensor 2	Sensor 2 input level of the bar code scanner
D12	Definable	Defined by the configuration of the bar code scanner	
D13	Definable		
D14	Definable		
D15	Definable		

PLC 输出控制字：

PLC 可以通过控制字节中的 Trigger 位来触发 SICK 产品开始读取，此时产品在 SOPAS 里需要设置为通过 Fieldbus 触发。

**Control Word Out (Ctrl Bits out) is assigned as follows:**

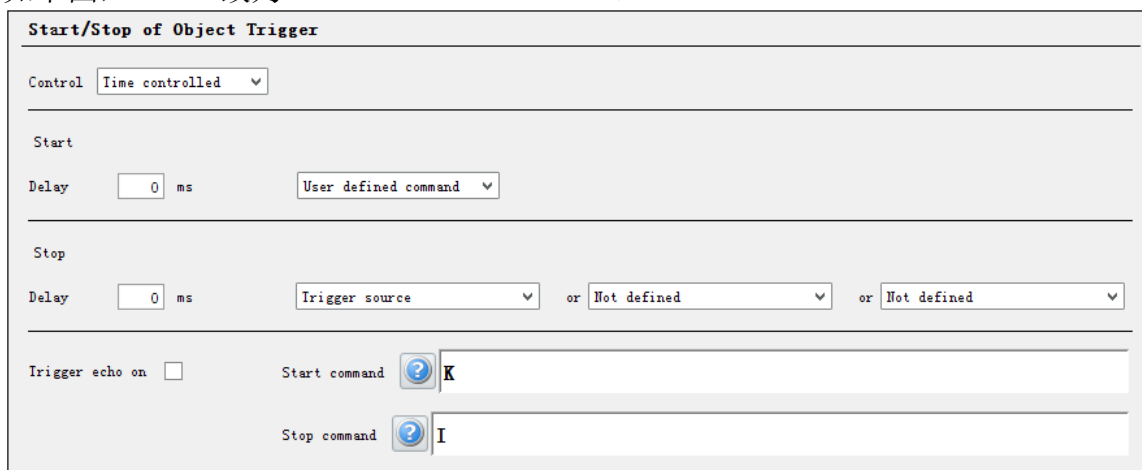
Bit	Assignment	Name	Comment
D0	Fixed	Trigger	Object trigger for the CLV6xx bar code scanner
D1	Fixed	Sensor idle	
D2	Fixed	Teach-in 1	Control of Teach-In 1
D3	Fixed	Teach-in 2	Control of Teach-In 2
D4	Fixed	External output 1	Output on OUT 1 of the field bus module
D5	Fixed	External output 2	Output on OUT 2 of the field bus module
D6	Fixed	Result 1	Output on Result 1 of the bar code scanner
D7	Fixed	Result 2	Output on Result 2 of the bar code scanner
D8	Fixed	PLC_Out_08	
D9	Fixed	PLC_Out_08	
D10	Fixed	PLC_Out_08	
D11	Fixed	PLC_Out_08	
D12	Definable	Distance_Config_0	
D13	Definable	Distance_Config_1	
D14	Definable	Distance_Config_2	
D15	Definable	Distance_Config_3	

### 3. 开始数据通讯

为了方便测试，我们先在 SOPAS 中将触发信号改成字符指令触发：

#### 3.1. 设置 SOPAS 触发方式

如下图，Start 改为 User Defined command。



Command 定义如下：



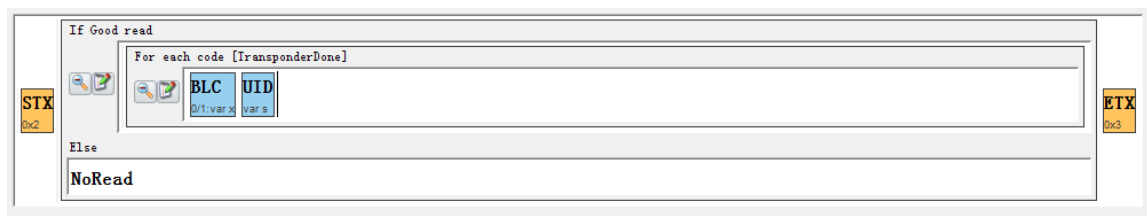
#### 3.2. 首先利用串口调试工具进行读取测试

发送 K, I 后可以得到标签传回的信息。

**KI**20202020E00401002F2E615D

该信息对应 SOPAS 中的设置：

Dataformat1:



#### 3.3. 监控 PLC 变量表

我们在到博图中监视前面设置好的接收字节地址：

按照下一节的握手规则，就可以看到 RFH 读取到的数据：



	i	名称	地址	显示格式	监视值
1			%IB256	字符	'\$00'
2			%IB257	字符	'\$01'
3			%IB258	字符	'\$00'
4			%IB259	字符	'\$1A'
5			%IB260	字符	'\$00'
6			%IB261	字符	'\$02'
7			%IB262	字符	'2'
8			%IB263	字符	'0'
9			%IB264	字符	'2'
10			%IB265	字符	'0'
11			%IB266	字符	'2'
12			%IB267	字符	'0'
13			%IB268	字符	'2'
14			%IB269	字符	'0'
15			%IB270	字符	'E'
16			%IB271	字符	'0'
17			%IB272	字符	'0'
18			%IB273	字符	'4'
19			%IB274	字符	'0'
20			%IB275	字符	'1'
21			%IB276	字符	'0'
22			%IB277	字符	'0'
23			%IB278	字符	'2'
24			%IB279	字符	'F'
25			%IB280	字符	'2'
26			%IB281	字符	'E'
27			%IB282	字符	'6'
28			%IB283	字符	'1'
29			%IB284	字符	'5'
30			%IB285	字符	'D'
31			%IB286	字符	'\$03'

### 3.4. 以下为握手模式的通讯说明：

\*如果使用 SICK FB 功能块，请忽略此部分内容。

Address	Inputs (data CLV → PLC)		Output (data PLC → CLV)
1	Binary Inputs		Binary Outputs
2	ReceiveCount (counter)	→	ReceiveCountBack (counter)
3	TransmitCountBack (counter)	←	TransmitCount (counter)
4	ReceiveLength Lowbyte		TransmitLength Lowbyte
5	ReceiveLength Highbyte		TransmitLength Highbyte
6	ReceiveData, Byte 1		TransmitData, Byte 1
7	ReceiveData, Byte 2		TransmitData, Byte 2
...	...		...
n	ReceiveData, Byte n - 5		TransmitData, Byte n - 5

在握手模式下，前 5 个字节用于数据的握手，因此，如果组态时选择了 32 字节输入，那么实际每次可传输 27 字节。

以下是数据交换的步骤：

- a. PLC 接收信息  
读码器读到一个数据后，会把数据发送到 PLC 输入区的 Data 区；同时，ReceiveCount 会累加，表示有新数据进来；数据的长度在 ReceiveLength 区显示。
- b. PLC 回应 RFHxxx  
PLC 收到数据后，必须回应读码器。PLC 需要在 10 秒以内把 ReceiveCount 的值拷贝到输出的 ReceiveCountBack 里，
- c. 读码器发送下一个数据  
读码器收到 ReceiveCountBack 的值后，开始发送下一个数据给 PLC，ReceiveCount 值再累加。

ReceiveCount 的值范围是 1 到 255, 0 会自动跳过。  
PLC 只需要做 Copy 动作就可以了，通常不需要其他动作。  
如果 PLC 收到 0，这是读码器的一个故障显示，PLC 也需要回应 0。

以下是通讯实例：

### 3.4.1. 短数据通讯

Input: 16 bytes, Output: 16 bytes,  
数据共 11 bytes → 单次数据可以满足。

		Time: 1 2 3 4				
// Input Byte - 16 Byte (Result Data received from CDM / Sensor)						
EB 50	"Stat-In-Bits"	BIN	2#0000_0100	2#0000_0100	2#0000_0000	2#0000_0000
EB 51	"Rec-Cnt"	DEC	1	1	2	2
EB 52	"Tr-Cnt-back"	DEC	0	0	0	0
EB 53	"Rec-Len-Low"	DEC	11	11	9	9
EB 54	"Rec-Len-High"	DEC	0	0	0	0
EB 55	"Rec-Data-1"	CHARACTER	'C'	'C'	'1'	'1'
EB 56	"Rec-Data-2"	CHARACTER	'L'	'L'	'2'	'2'
EB 57	"Rec-Data-3"	CHARACTER	'V'	'V'	'3'	'3'
EB 58	"Rec-Data-4"	CHARACTER	'6'	'6'	'4'	'4'
EB 59	"Rec-Data-5"	CHARACTER	'X'	'X'	'5'	'5'
EB 60	"Rec-Data-6"	CHARACTER	'X'	'X'	'6'	'6'
EB 61	"Rec-Data-7"	CHARACTER	'L'	'L'	'7'	'7'
EB 62	"Rec-Data-8"	CHARACTER	'D'	'D'	'8'	'8'
EB 63	"Rec-Data-9"	CHARACTER	'a'	'a'	'9'	'9'
EB 64	"Rec-Data-10"	CHARACTER	't'	't'	B#16#00	B#16#00
EB 65	"Rec-Data-11"	CHARACTER	'a'	'a'	B#16#00	B#16#00
// Output bytes - 16 Byte (e.g. Commands send to CDM / Sensor)						
AB 50	"Stat-Out-Bits"	BIN	2#0000_0000	2#0000_0000	2#0000_0000	2#0000_0000
AB 51	"Rec-Cnt-Back"	DEC	0	1	1	2
AB 52	"Tr-Cnt"	DEC	0	0	0	0
AB 53	"Tr-Len-Low"	DEC	0	0	0	0
AB 54	"Tr-Len-High"	DEC	0	0	0	0
AB 55	"Tr-Data-1"	CHARACTER	B#16#00	B#16#00	B#16#00	B#16#00
AB 56	"Tr-Data-2"	CHARACTER	B#16#00	B#16#00	B#16#00	B#16#00
AB 57	"Tr-Data-3"	CHARACTER	B#16#00	B#16#00	B#16#00	B#16#00
AB 58	"Tr-Data-4"	CHARACTER	B#16#00	B#16#00	B#16#00	B#16#00
AB 59	"Tr-Data-5"	CHARACTER	B#16#00	B#16#00	B#16#00	B#16#00
AB 60	"Tr-Data-6"	CHARACTER	B#16#00	B#16#00	B#16#00	B#16#00
AB 61	"Tr-Data-7"	CHARACTER	B#16#00	B#16#00	B#16#00	B#16#00
AB 62	"Tr-Data-8"	CHARACTER	B#16#00	B#16#00	B#16#00	B#16#00
AB 63	"Tr-Data-9"	CHARACTER	B#16#00	B#16#00	B#16#00	B#16#00
AB 64	"Tr-Data-10"	CHARACTER	B#16#00	B#16#00	B#16#00	B#16#00
AB 65	"Tr-Data-11"	CHARACTER	B#16#00	B#16#00	B#16#00	B#16#00

### 3.4.2. 长数据通讯

input: 16 bytes, output: 16 bytes,  
数据报文“2MC10j1234567890”共16字节 → 分为两次发送

Time:            1/2            3/4

// Input Byte - 16 Byte (Result Data received from CDM / Sensor)				
EB 50	"Stat-In-Bits"	BIN	2#0000_0100	2#0000_0000
EB 51	"Rec-Cnt"	DEC	0	0
EB 52	"Tr-Cnt-back"	DEC	1	2
EB 53	"Rec-Len-Low"	DEC	1	1
EB 54	"Rec-Len-High"	DEC	0	0
EB 55	"Rec-Data-1"	CHARACTER	'1'	'1'
EB 56	"Rec-Data-2"	CHARACTER	B#16#00	B#16#00
EB 57	"Rec-Data-3"	CHARACTER	B#16#00	B#16#00
EB 58	"Rec-Data-4"	CHARACTER	B#16#00	B#16#00
EB 59	"Rec-Data-5"	CHARACTER	B#16#00	B#16#00
EB 60	"Rec-Data-6"	CHARACTER	B#16#00	B#16#00
EB 61	"Rec-Data-7"	CHARACTER	B#16#00	B#16#00
EB 62	"Rec-Data-8"	CHARACTER	B#16#00	B#16#00
EB 63	"Rec-Data-9"	CHARACTER	B#16#00	B#16#00
EB 64	"Rec-Data-10"	CHARACTER	B#16#00	B#16#00
EB 65	"Rec-Data-11"	CHARACTER	B#16#00	B#16#00

// Output bytes - 16 Byte (e.g. Commands send to CDM / Sensor)				
AB 50	"Stat-Out-Bits"	BIN	2#0000_0000	2#0000_0000
AB 51	"Rec-Cnt-Back"	DEC	0	0
AB 52	"Tr-Cnt"	DEC	1	2
AB 53	"Tr-Len-Low"	DEC	16	5
AB 54	"Tr-Len-High"	DEC	0	0
AB 55	"Tr-Data-1"	CHARACTER	'2'	'6'
AB 56	"Tr-Data-2"	CHARACTER	'M'	'7'
AB 57	"Tr-Data-3"	CHARACTER	'C'	'8'
AB 58	"Tr-Data-4"	CHARACTER	'1'	'9'
AB 59	"Tr-Data-5"	CHARACTER	'0'	'0'
AB 60	"Tr-Data-6"	CHARACTER	'j'	B#16#00
AB 61	"Tr-Data-7"	CHARACTER	'1'	B#16#00
AB 62	"Tr-Data-8"	CHARACTER	'2'	B#16#00
AB 63	"Tr-Data-9"	CHARACTER	'3'	B#16#00
AB 64	"Tr-Data-10"	CHARACTER	'4'	B#16#00
AB 65	"Tr-Data-11"	CHARACTER	'5'	B#16#00

在成功组态 RFHxxx 之后，可选择调用 SICK FB 功能块，对 RFH 进行读写操作。

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/465321203232011242>