V0.0.0 Date:2025/01/13

类别	内容
关键词	DPort-ECT, EtherCAT, AWorks
摘要	EtherCAT 从站参考手册



AWorksLP

修订历史

版本	日期	原因



AWorksLP

目 录

1.	名词/缩写解释1
2.	SSC 的配置及其使用 2
	2.1 SSC 选项栏使用方法
	2.1.1 File
	2.1.2 Project
	2.1.3 Tool
	2.1.4 help
	2.2 导入模板文件
	2.2.1 模板文件 (.xml)
	2.2.2 导入步骤9
	2.3 SSC 配置项 ······11
	2.3.1 SlaveInformation 11
	2.3.2 generic 11
	2.3.3 Hardware
	2.3.4 EtherCAT State Machine14
	2.3.5 Synchronisation
	2.3.6 Application
	2. 3. 7 Maibox
	2.3.8 Compiler
	2.4 新建工程
	2.5 新建/导入对象字典
	2.5.1 新建/导入 Excel 步骤
	2.5.2 Excel 文件编与规则······18
	2.5.3 周期数据 (Input/Output) ····································
	2.6 源码和 ESI 义件生成 ····································
	2.7 应用柱序与按口函数······27
	2. 7. 1 Vold APPL_InputMapping(UINI16* pData)
	2. 7. 2 void APPL_Outputmapping(UINITO* pData)
	2. 7. 3 Volu AFFL_Application(Volu) ······ 28
3.	TwinCAT 操作及使用
	3.1 TwinCAT 下载及安装
	3.1.1 TwinCAT3 硬件配置要求30
	3.1.2 TwinCAT3 安装步骤30
	3.1.3 TwinCAT3 网卡配置30
	3.2 新建 TwinCAT 工程32
	3.3 试用版授权激活
	3.4 TwinCAT ESI 文件烧录35



	3.5 TwinCAT 主从栈通信
	3.5.1 EtherCAT 状态机详解 ······37
	3.6 TwinCAT 实时模式 ······39
	3.7 TwinCAT 设置 Box 热拔插42
4.	ESC 相关硬件配置
	4.1 DPort-ECT
	4.2 协议栈硬件适配接口45
	4.3 EEPROM
	4.4 ESC Configuration Data(Handware Config)48
	4.4.1 PDI Control register(SPI interface select)
	4.4.2 ESC Configuration register(SYNCO/1 Int Out enabled)50
	4.4.3 PDI Configuration register(SPI mode and PDI Int Config)51
	4.4.4 Pulse Length of SYNC Signals(SYNCO/1 Signals length)52
	4.5 ESC SPI Slave
	4.5.1 ESC SPI Read Access
	4.5.2 ESC SPI Write Access
	4.5.3 ESC Read/Write 通信时序54
	4.5.4 ESC 访问函数伪代码实现56



1. 名词/缩写解释

SSC EtherCAT Slave Stack Code Tool

- ESI EtherCAT Slave Information
- PDI Process Data Interface
- ESC EtherCAT Slave Controller
- DC Distributed Clock



- 注意

 \wedge

2. SSC 的配置及其使用



- 此文档以 EPC103-DP EtherCAT 评估板为例;
 - DPort-ECT 相关例程均使用 SSC_V5i13 版本,注意核对;
 - 倍福官方 SSC 工具下载链接¹。

¹ https://www.beckhoff.com.cn/zh-cn/support/download-finder/search-result/?search=slave%20stack



2.1 SSC 选项栏使用方法

sse Sla	ave Stack C	ode To	ol				_	×
File	Project	Tool	Help	, ,				
Slave H	roject Nav	igation		Slave Settings				
				C (1 / -) -				
				Confilters				
Failed	to create a	new sla	ive pro	oject				

图 2.2 SSC 主界面

2.1.1 File

- New: 新建工程;
- Open: 打开已有的工程;
- Save: 保存工程配置;
- Save As: 将工程另存为;
- Exit: 退出。

2.1.2 Project

• Find Setting: 查找配置项;



AWorksLP

File Project Tool ave Project Navigation	Help Slave Settings				
-Bio -SlaveInformation -Generic Horducro	Slave Stack Code Too	bl Find Setting	- 0 X		
- EtherCAT State Mac - Synchronisation - Application	h Name AI EVENT ENABLED	Value 1	Description	_	Version 0.0.0.1
- FrocessData Mailbox Compiler	SERIAL_NUMBER	- 0x0000000	Define: EL9800_HW Shall be set if the Slave code is executed		- 5.11
			on the FLU mounted on the EL9800 EtherCAT Evaluation Board. (if the MCI interface		5. 11 5. 13 5. 12
			provided by EL9800 board should be used MCI_HW shall be set and this define shall be reset)		5.11 5.12
			This settings should also be enabled if the ESC is connected via a serial interface and no specific hardware access filor ore convilotion to	Remove	5.12

图 2.3 Find Setting

• Create new Slave Files: 生成源码和 ESI 文件等

2.1.3 Tool

• Show Confict Window:显示信息提示窗口

File Project Lool Hel	p Slava Sattinan					
<pre>lave Project Rawigation</pre>	SLave Settings Name EL9800_HW MCT_HW FC1100_HW HW_ACCESS_FILE CONTROLLER_16BIT CONTROLLER_32BIT _PIC18 _PIC24 ESC_16BIT_ACCESS ESC_32BIT_ACCESS BIG_ENDIAN_16BIT BIG_ENDIAN_16BIT BIG_ENDIAN_FORMAT Conflicts @ Info Warning (Select only one control	Value 1 0 0 1 0 1 1 0 1 0 1 1 0 1 0 1 0 1 0	A Descript Define: Shall b control	tion CONTROLLER_321 be set if the h ller is a 32Bit	BIT ost archited	ture

图 2.4 Show Confict Window



• Options:

8io - Slave S File Project	tack Code Tool	_		×
He Project Slave Project Bio Slave - Gener - Hardw - Sthere - Stave - Gener - Hardw - Btheri - Synch - Synch - Pro- - Mailb - Compi	 I Dol Help ave Stack Code Tool Options Generic Editor Create Files Configurations On startup Open last project Oteck for updates Default Patch Ox201811 Default Patch Open Open 		×	a 1
	0K	Canc	el .:	

图 2.5 Generic

File Project	stack Code Tool	-		×
Slave Projec S	lave Stack Code Tool Options		×	
- 810 Slave	Generic Editor Create Files Configurations			
Hardw Ether Synch	 ✓ Show advanced settings ✓ Show read only settings ✓ Show read only settings 			n
- Appli Pr Wailb	☑ Block type mismatching setting values 显示块类型与设置值不匹配			1
- Compi	ApplyAll VDdate dependent settings 更新依赖设置			
	.xlsx v Application file format 应用又件格式			
	Merge tool Tool path 			
	Arguments			
	"%TargetFile%" "%SourceFile%"			
				d
	ОК	Cane	el	

图 2.6 Editor



AWorksLP

User Manual

8io - Slave Stack Code Tool	-		\times
File Project Tool Help			
Slave Projec Slave Stack Code Tool Options		×	
Slave Generic Editor Create Files Configurations			
- Har dw			
Ether			
Synch Add comments if obsolete code was skipped 生成源码时,提示被跳过的代码			
Pr Create device description (ESI) 创建设备描述文件(.xml)			1 .
			a
OK	Cane	el	



🚥 8io - Slave Stack Code Tool	—		×
File F Slave Pro		×	
⊟-Bio -Sla Generic Editor Create Files Configurations			
- Ger Har - Ett - Syr - Apr - Mai - Con		i on). 1). 11	•
		Add	
ок	Cancel		
			.:;

图 2.8 Configurations

• EEPROM Programmer: EEPROM ESI 烧录



User Manual

EtherCAT 从站参考手册

AWorksLP

File Project Tool H	telp					
lave Project Navigation	Slave Settings					
EtherCAT Slave SlaveInformation Generic	SSC Version 5.13 Config File Vers:1.5.	3				
- Hardware	File name	Description			Version	
	aceappl. c	AoE ADS over EtherCAT			5.11	
- Application	aoeappl. h				5.11	- 1
Mailbox	EEPROM Programme	er (V.1.3.0.0) —			5.13	
····Compiler	File Slaves 用干≠	H描从站和校验FFPROM			5.12	
	打开ESI文件				5.11	
	Device Description	T 更換录的FSI文件			5.12	
	SSC-Device (rev 0x1)				5.12	
	Slaves				5.13	
	0x1 0x1 扫描と	出的从站	~		5.13	
	Dacket counter 1 / 1 1 Eth	EtherCAT slave(s) found			5.13	
			Reload	Remove	Add	
	Conflicts					
	👥 Info 🔥 Warni	ng 🔞 Error				

图 2.9 EEPROM Programmer

• Application: 导入或新建对象字典文件 (之后章节会详细介绍)

2.1.4 help

• About: SSC 版本相关信息

se Ethe	erCAT Slave	e* - Slave	Stack Code Tool	—		\times
File	Project	Tool	Help			
ave Pr Ethe	About Sla	ave Stack	Code Tool		×	
S b b	Tool 1.9 ET9300 J ET9300 J	5.3.0 Project J Applicat:	Mandler 1.5.3.0 on Parser 1.6.4.0			n
4	Copyri	ght © Bea	khoff Automation GmbH Co. KG 2019			
N	Beckhoi	££ Autom≪	tion GmbH Co. KG			
	ET9300	(EtherCA	I Slave Stack Code) Tool			
	The Sla softwar	ve Stack e compon	Code Tool and ET9300Utilities. ApplicationHandler are using the following open sourcents:	;e	I	
	SharpZi	pLib whi	ch is published under the MIT license:			
	Author	:	ICSharpCode			ld
	Contrib	utors:	John Reilly			Г
				OK		J
lew pro	oject crea	ted				

图 2.10 About



- Contact: 倍福官方联系方式
- Documentation: SSC 相关文档
- Check for Updates: 检查更新

2.2 导入模板文件

2.2.1 模板文件 (.xml)

点击 File -> New 新建工程,会出现如下界面:

Slave Stack Code Tool New Project	×
O Default	
Custom EL9800 2Axis CiA402 Sample	~
EL9800 2Axis CiA402 Sample This confit EL9800 8Bit Digital I/0, 16Bit Analog Input EtherCAT Ex EL9800 Bootloader Sample The correst FC1100/FC1121 4Byte I/0 Sample Application The correst FC1100/Win8 or previous (x86) 4Byte I/0 Sample Application \SlaveStacl Infineon XMC EtherCAT Slave Sample <infineon technologies=""> tool. Microchip-HBI-INDEX-16 (Microchip) Microchip-HBI-HDEX-16 (Microchip) Microchip-HBI-MDP-6 (Microchip) Microchip-HBI-MDP-16 (Microchip) Microchip-HBI-MDP-16 (Microchip) Microchip-HBI-MDP-16 (Crofichip GmbH) Renesas RI/N1L-DB (Renesas Electronics) Renesas RZ/N1L-DB (Renesas Electronics) Renesas RZ/N1L-DB (Renesas Electronics)</infineon>	
Import SK-FM3-176FMC-FA _ Hardware Only (Spansion Inc.) SK-FM3-176FMC-FA (Spansion Inc.) TI AM335x Sample (Texas Instruments Incorporated)	

图 2.11 SSC 默认工程模板

如图所示,下拉菜单中的各种选项是 ESC 厂商根据自身硬件平台所提供的模板例程,如 果所选模板平台与自身需求吻合,便可进行复用,

我们也向用户提供了基于 EPC103-DP/EPC6450-DP 平台的模板文件,以便参考。

打开 EPC103-DP 评估板中 {例程}/ EtherCAT-Slave-Adapter/ DPort-ECT_SSC_Config.xml 文件,内容如图下所示。

```
程序清单 2.1 DPort-ECT_SSC_Config.xml 中的部分内容
```

```
<!--EtherCAT State Machine-->
<Setting CodeDefine="BOOTSTRAPMODE_SUPPORTED" Value="0"/>
<Setting CodeDefine="OP_PD_REQUIRED" Value="1"/>
<Setting CodeDefine="PREOPTIMEOUT" Value="0x7D0"/>
<Setting CodeDefine="SAFEOP2OPTIMEOUT" Value="0x2328"/>
<Setting CodeDefine="CHECK_SM_PARAM_ALIGNMENT" Value="0"/>
<!--Synchronisation-->
```

对比 SSC 中的配置项:



AWorksLP

🏧 ecat - Slave Stack Code Tool		
File Project Tool Help		
Slave Project Navigation	Slave Settings	
ecat	Name	Value
SlaveInformation Generic	BOOTSTRAFMODE_SUPPORTED	0
Hardware	OP_PD_REQUIRED	1
<mark>EtherCAT State Machine</mark> Synchronisation	PREOPTIMEOUT	0x7D0
- Application	SAFEOP2OPTIMEOUT	0x2328
Frocessuata Mailbox	CHECK_SM_PARAM_ALIGNMENT	0
·····Lompiler		

图 2.12 SSC 配置界面对比

可以发现, **DPort-ECT_SSC_Config.xml**中的内容对应着 SSC 中的配置项,当新建工程时, SSC 的配置项会按照所选的工程模板中的键值进行设置。

注意 ______ 注意 ______ DPort-ECT_SSC_Config.xml 文件必须与 EtherCAT-Slave-Adapter 驱动组件配套使用。

2.2.2 导入步骤

• 1. 打开 SSC 软件,点击 Tool -> Option -> Configurations,出现如下界面:

Slave Stack Code Tool | Options

Generic Editor Create Files Configurations

FC1100/Win8 or previous (x86) ... ,
FC1100/FC1121 | 4Byte I/O Samp...
EL9800 | 8bit Digital I/O, 16B...
EL9800 | Bootloader Sample

EL9800 | Bootloader Sample

OK Cancel

图 2.13 工程模板导入界面

• 2. 点击符号 +,选择例程下的模板文件,以 EPC103-DP 中 8io 为例,导入 DPort-ECT_SSC_Config.xml 文件。



AWorksLP

User Manual

Slave Stack Code Tool | Options

Generic Editor Create Files Conf:	gurations			
Generic Editor Create Files Confi FC1100/Win8 or previous (x86) FC1100/FC1121 4Byte I/O Samp EL9800 8Bit Digital I/O, 16B EL9800 2Axis CiA402 Sample EL9800 Bootloader Sample ZLG DPort-ECT SSC setting (ZLG)	gurations			
			ОК	Cancel

图 2.14 工程模板成功添加界面

• 3. 新建工程查看是否存在添加的模板, File->New。

sse EtherCAT Slave* - Slave	Stack Code To	ol							—		\times
File Project Tool	Help										
Slave Project Navigation	Slave	Settings									
- EtherCAT Slave	Name			Value			Descript	ion			
SlaveInformation Generic	MAILBO	X_QUEUE		1			Define:	MAILBO	X_QUEUE		
Hardware	10 R SI	PPORTED		n				_			
EtherCAT State Mac Synchronisation	Slave Stack C	ode Tool	New Project					× ^{ch} 1	is set, be store	the mai d in a	lbox
- Application	🔿 Default										
	🗿 Custom	EL9800	2Axis CiA402 Samp	le				V it	ch reset	only or	ie ,
Compiler	-	EL9800	2Axis CiA402 Samp	le				10	ce can be	process	:ed
	Ihis confi EtherCAT E The corres \SlaveStac tool.	KL9800 FC1100/ FC1100/ FC1100/ Infineo Microch Microch Microch Microch Microch Microch Microch SK-FM3- SK-FM3- SK-FM3- TI C28x TI C28x	8Bit Digital I/O, Bootloader Sample FC1121 4Byte I/O S WinB or previous (xi a XMC EtherCAT Slav, ip-HBI-INDEX-16 (Micro ip-HBI-MDP-6 (Micro ip-HBI-MDP-6 (Micro ip-HBI-MDP-16 (Micro ip-HBI-MDP-16 (Micro ip-HBI-MDP-16 (Micro ip-HBI-STI (Microch ip-HBI-STI (Micro	16Bit Analog Sample Applics 36) 4Byte I/ schip> rochip> ochip> ochip> drip> ip> fichip GmbH> as Electronics> e Only Spansi Inc.> struments Inco ample applicat ample applicat o ESC(ASYNC16 D ESC(SYI PDI) des Sample App Instruments I otion Control> (ZLG)	Input ation () Sample A (neon Techn () () () () () () () () () () () () ()	ppli olog ET11 ET11 DSEC ATCN <tex d></tex 	cation (ies) 00 ESC (A 00 ESC (S ACTORD37 ICD379D k (as Instr	SYN PI it ume			

图 2.15 新建测试模板工程



AWorksLP

2.3 SSC 配置项

2.3.1 SlaveInformation

SSC 配置项	含义	注解
VENDOR_ID	厂商 id	由 EtherCAT 官方进行分配
VENDOR_NAME	厂商名称	
VENDOR_IMAGE	厂商图标	bmp 格式 (16x14)px
GROUP_NAME	组别名称	
GROUP_IMAGE	组别图标	bmp 格式 (16x14)px
DEVICE_IMAGE	设备图标	bmp 格式 (16x14)px
PRODUCT_CODE	产品码	
REVISION_NUMBER	修订号	
SERIAL_NUMBER	序列号	
DEVICE_PROFILE_TYPE	设备描述类型	
DEVICE_NAME	设备名称	
DEVICE_HW_VERSION	设备硬件版本	
DEVICE_SW_VERSION	设备软件版本	

— 注意 —

此为设备版本相关的描述信息,需要由用户自定义,有的主站会对厂商信息产品码等 有硬性要求,在 init 阶段会进行检查,若检查失败则可能会导致状态转换的失败。

2.3.2 generic

表 2.2 SYSTEM_HEADER_FILE

SSC 配置项	含义
SYSTEM_HEADER_FILE	系统头文件

注: 通常格式为 #include "xxx.h",头文件将会包含在 ecat_def.h 文件中,协议栈的所有代码皆可包含到 xxx.h。若使用 EtherCAT-Slave-Adapter 组件中的 DPort-ECT_SSC_Config.xml 模板文件时,此配置项默认为 #include "aw_ecat_ssc.h"。

表 2.3 EXPLICIT_DEVICE_ID

SSC 配置项	含义
EXPLICIT_DEVICE_ID	显示从站 ID

注: EXPLICIT_DEVICE_ID 用于设置寻址,在链路初始化阶段,从站协议站会调用 UINT16 APPL_GetDeviceID(void) 函数 (需用户实现) 获取用户设置的设备 ID,并将其作为此从站寻址依据,这 就意味着在设置寻址下,从站的地址与从站的物理连接位置无关。



使用此功能时需要主站设置相应的热拔插 («TwinCAT 操作及使用»章节介绍)。

表 2.4 ESC_SM_WD_SUPPORTED



AWorksLP

SSC 配置项	含义
ESC_SM_WD_SUPPORTED	同步管理器看门狗设置

注:用于检测是否在规定时间内接收到周期数据,与之关联的错误码为 ALSTATUSCODE_SMWATCHDOG。

—— 注意 –

▲ 在进入 op 阶段时,若出现错误码为 ALSTATUSCODE_SMWATCHDOG 而导致的状态 转换失败,其原因在于规定时间内未接受到周期数据,此时需要检测一下 PDI 中断的 配置情况。

表 2.5 STATIC_OBJECT_DIC

SSC 配置项	含义
STATIC_OBJECT_DIC	对象存储方式

注:此宏只会在 EL9800_APPLICATION 中使用,无实际配置作用。

表 2.6 ESC_EEPROM_ACCESS_SUPPOT

SSC 配置项	含义
ESC_EEPROM_ACCESS_SUPPOT	是否支持访问 EEPROM

注: ESC 若支持访问 EEPROM,则会提供访问接口:

- UINT16 ESC_EepromAccess(UINT32 wordaddress, UINT16 wordsize, UINT16 MBXMEM *pData, UINT8 access);
- UINT32 ESC_EepromWriteCRC(void).

── 注意 -⚠ EEPR

▲ EEPROM 读写规则

主站可以通过操作 ESC EEPROM 控制器访问内部的数据,而从站操作 EEPROM 则需要主站赋予操作权限(由寄存器 0x500 体现,对于从站是只读的),在以下情况 EtherCAT 主站会通过写 0x500 将访问权限交给 PDI:

- 1. 在 *init -> preop* 转换时;
- 2. 在 init -> boot 转换时,包括处于 boot 状态下;
- 3. 在 ESI 文件中定义 Assign ToP di 元素,除 init 外,主站会将权限交付给 PDI。

2.3.3 Hardware

SSC 配置项	含义
EL9800_HW	倍福官方开发板驱动
MCI_HW	
FC1100_HW	

SSC 配置项	含义
HW_ACCESS_FILE	硬件接口文件

注:格式为 #include "xxx.h",指定协议栈硬件接口文件 (需要由用户提供),若使用 EtherCAT-Slave-



Adapter 组件中的 DPort-ECT_SSC_Config.xml 模板文件时, 此配置项默认为 #include "aw_ecat_ssc.h",此文 件会提供例如下的 IO 读写等函数:

- HW_EscRead(pData, Address, Len);
- HW_EscWrite(pData, Address, Len);
- ·····

SSC 配置项	含义
CONTROLLER_16BIT	指定微控制器架构:16Bit or 32Bit
CONTROLLER_32BIT	

SSC 配置项	含义
MEMORY_UNIT_16BIT	内存单元最小是否为 16bit

注: 常见的 32 位芯片最小寻址单元为 1Byte(8Bit), 但是在一些 DSP 芯片中, 内部最小寻址单元就是 16bit。

SSC 配置项	含义
_PIC18	倍福基于 PIC 芯片做的示例
_PIC24	

SSC 配置项	含义
ESC_16BIT_ACCESS	ESC 内存访问对齐限制:16Bit or 32Bit
ESC_32BIT_ACCESS	



- 注意 -

★ 大部分 ESC 芯片对 EtherCAT 内核部分寄存器的访问并没有限制,但是部分集成了 EtherCAT 外设的 MCU(On-Chip-Bus) 对 RAM 区域的访问必须是4字节对齐,协议栈 在配置时也需要指定此处为四字节对齐。

SSC 配置项	含义
MBX_16BIT_ACCESS	微控制器邮箱区域内存 16bit 对齐访问限制
BIG_ENDIAN_16BIT	大端 16bit 模式
BIG_ENDIAN_FORMAT	大端模式
EXT_DEBUGER_INTERFACE	协议栈中_PIC24 例程所使用的宏, 启用 debug 信
	息输出

SSC 配置项	含义
UC_SET_ECAT_LED	从站状态指示灯 (微控制器控制)

注: 用户需实现 void HW_SetLed(UINT8 RunLed, UINT8 Errled) 函数,由协议栈调用,由微控制器需控制两个 led 的亮灭:

- RunLed == 1: run 灯亮;
- Errled == 1: err 灯亮;
- RunLed == 0: run 灯灭;
- Errled == 0: err 灯灭。



AWorksLP

User Manual

在 EPC103-DP 例程中实现为:

```
void __us_set_stack_led(uint8_t run_led, uint8_t err_led)
{
    run_led ? LL_GPIO_ResetOutputPin(RUN_GPIO_Port, RUN_Pin):
        LL_GPIO_SetOutputPin(RUN_GPIO_Port, RUN_Pin);
        err_led ? LL_GPIO_ResetOutputPin(ERR_GPIO_Port, ERR_Pin):
        LL_GPIO_SetOutputPin(ERR_GPIO_Port, ERR_Pin);
}
```

2.3.4 EtherCAT State Machine

SSC 配置项	含义
BOOTSTRAPMODE_SUPPORTED	boot 状态支持
OP_PD_REQUIRED	转换到 op 状态时,指定是否需要接收到周期数据
PREOPTIMEOUT	Init -> preop 状态转化超时时间
SAFEOP2OPTIMEOUT	safeop -> op 状态转换超时时间
CHECK_SM_PARAM_ALIGNMENT	检查 SM 的相关参数是否按要求对齐

2.3.5 Synchronisation

SSC 配置项	含义
AL_EVENT_ENABLED	PDI 中断使能
DC_SUPPORTED	ESC 是否支持分布式时钟

注: AL_EVENT_ENABLED 与 DC_SUPPORTED 不同组合会形成不同的通信模式:

	AL_EVENT_ENABLED	DC_SUPPORTED
freerun	0	0
SM-Sync	1	0
DC-Sync	0	1
SM-DC-Sync	1	1

SSC 配置项	含义
ECAT_TIMER_INT	是否使用定时器中断

注: ECAT_TIMER_INT 置一后需要为协议栈提供一个 1ms 的定时器中断,在中断回调内调用协议栈函数: void ECAT_CheckTimer(void), STM32 代码示例如下:

```
void TIM2_IRQHandler(void)
{
    if(LL_TIM_IsActiveFlag_UPDATE(TIM2) == SET)
    {
        LL_TIM_ClearFlag_UPDATE(TIM2);
#if ECAT_TIMER_INT
        ECAT_CheckTimer();
        __g_ecat_timer_inc++;
#endif
```



}

}

若不使用定时器中断则需要向协议栈提供以下函数和宏:

- fn void HW_ClearTimer(void): 清除当前的时间计数值
- fn UINT16 HW_GetTimer(void): 获取当前的时间计数值
- macro ECAT_TIMER_INC_P_MS: 每毫秒有多少计数值

SSC 配置项	含义
MIN_PD_CYCLE_TIME	周期数据最小的循环时间
MAX_PD_CYCLE_TIME	周期数据最大的循环时间
PD_OUTPUT_DELAY_TIME	周期数据最小输出延时时间
PD_OUTPUT_CALC_AND_COPY_TIME	周期 OUTPUT 数据最小计算拷贝时间
PD_INPUT_CALC_AND_COPY_TIME	周期 INPUT 数据最小计算拷贝时间
PD_INPUT_DELAY_TIME	周期数据最大输出延时时间

2.3.6 Application

SSC 配置项	含义
TEST_APPLICATION	测试例程
TEST_APPLICATION_REDUCED_MEMORY	
EL9800_APPLICATION	EL9800 设备例程
CiA402_SAMPLE_APPLICATION	cia402 电机例程
SAMPLE_APPLICATION	独立于硬件的示例例程
SAMPLE_APPLICATION_INTERFACE	
BOOTLOADER_SAMPLE	boot loader 示例

SSC 配置项	含义
APPLICATION_FILE	应用程序头文件

注: 通常格式为 #include "xxx.h"。在导入时 Excel 文件后,此配置项的值会自动设置为含 Excel 的名称 的头文件,例如在例程 8io 当中,当导入 8io.xlsx 后,此配置项就会变为 #include "8io.h"。

reason:

在导入 Excel 文件后,SSC 会在源码中生成三个带有 Excel 名称的应用相关的文件。以 8io 例程为例,当 导入 8io.xlsx 后, Src 中会生成一系列的源码文件,其中除 8io.h、8io.c、8ioObjects.h 外,其余源码皆是与协议栈相关的 (内容是固定的):

- 8io.c/8io.h: 用户应用文件,用户需实现几个接口函数(见 << 应用程序与接口函数 >>),用于周期数 据处理;
- 8ioObjects.h: Excel 中编写的数据对象会在此文件以 c 语言的方式体现出来。

SSC 配置项	含义				
USE_DEFAULT_MAIN	协议栈提供 main 函数				

注:协议栈会提供一个默认的 main 函数,方便用户调用,大致结构如下所示。



User Manual

EtherCAT 从站参考手册

AWorksLP

```
void main(void)
```

```
{
    /* initialize the Hardware and the EtherCAT Slave Controller */
    HW_Init();
    MainInit();
    bRunApplication = TRUE;
    do
    {
        MainLoop();
    } while (bRunApplication == TRUE);
    HW_Release();
}
```

• ProcessData

SSC 配置项	含义
MIN_PD_WRITE_ADDRESS	OUTPUT 周期数据在 DPROM 中映射的最小起始
	地址
DEF_PD_WRITE_ADDRESS	OUTPUT 周期数据在 DPROM 中映射的默认地址
MAX_PD_WRITE_ADDRESS	OUTPUT 周期数据在 DPROM 中映射的最大终止
	地址
MIN_PD_READ_ADDRESS	INPUT 周期数据在 DPROM 中映射的最小起始地
	址
DEF_PD_READ_ADDRESS	INPUT 周期数据在 DPROM 中映射的默认地址
MAX_PD_READ_ADDRESS	INPUT 周期数据在 DPROM 中映射的最大终止地
	址
MAX_PD_INPUT_SIZE	INPUT 周期数据最大 Size
MAX_PD_OUTPUT_SIZE	OUTPUT 周期数据最大 Size

2.3.7 Maibox

SSC 配置项	含义
MAILBOX_QUEUE	使能邮箱队列
AOE_SUPPORTED	AOE 协议使能
COE_SUPPORTED	COE 协议使能
COMPLETE_ACCESS_SUPPORTED	完全访问
SEGMENTED_SDO_SUPPORTED	sdo 分段是否支持
SDO_RES_INTERFACE	
USE_SINGLE_PDO_MAPPING_ENTRY_DESCR	
BACKUP_PARAMETER_SUPPORTED	
STORE_BACKUP_PARAMETER_IMMEDIATELY	
DIAGNOSIS_SUPPORTED	
MAX_DIAG_MSG	
EMERGENCY_SUPPORTED	
MAX_EMERGENCIES	
VOE_SUPPORTED	VOE 协议使能
SOE_SUPPORTED	SOE 协议使能



AWorksLP

续上表

SSC 配置项	含义
EOE_SUPPORTED	EOE 协议使能
STATIC_ETHERNET_BUFFER	以太网 buffer 存储形式
FOE_SUPPORTED	FOE 协议栈使能
MAILBOX_SUPPORTED	邮箱使能
MIN_MBX_SIZE	邮箱最小 size(ESC RAM 区域)
DEF_MBX_SIZE	邮箱默认 size(ESC RAM 区域)
MAX_MBX_SIZE	邮箱最大 size(ESC RAM 区域)
MIN_MBX_WRITE_ADDRESS	写邮箱区域最小起始地址
DEF_MBX_WRITE_ADDRESS	写邮箱区域默认起始地址
MAX_MBX_WRITE_ADDRESS	写邮箱区域最大起始地址
MIN_MBX_READ_ADDRESS	读邮箱区域最小起始地址
DEF_MBX_READ_ADDRESS	读邮箱区域默认起始地址
MAX_MBX_READ_ADDRESS	读邮箱区域最大起始地址

2.3.8 Compiler

• 略

2.4 新建工程

新建模板工程,File->New选择所要使用的模板,并点击OK。

EtherCAT Slave* - Slave	Stack Code To	ol						_		\times	
File Project Tool	Help										
Slave Project Navigation	Slave	Settings									
■ EtherCAT Slave	Name			Value			Description				
Generic	MAILBO	X_QUEUE		1			Define: MAII	LBOX_QUEUE			
- Hardware EtherCAT State Mer		PPORTED		n							
Synchronisation	Slave Stack C	ode Iool Nev	w Project				×	L be stored in a			
- Application	🔿 Default			1				itch reset only one ice can be processed			
	🗿 Custom	ZLG DPort-EC	CT SSC settin	ug (ZLG)			~		ne sed		
	Import	to <u>A</u> War	n is not pro ; may be adde automation Gm	vided by Bea d which are bH.	khoff Automat NOT covered b	ion s	and 2 OK				

图 2.16 新建工程

新工程的配置就会和模板工程中的一致,且支持修改配置。



AWorksLP

2.5 新建/导入对象字典

— 注意 —

/!\

SSC 中对象字典是在 Excel(CSV 也可但不常用) 中定义的,在编辑前建议补充一些 CanOpen 协议的内容和概念。

2.5.1 新建/导入 Excel 步骤

新建: Tool -> Application -> Create new:

创建新的空白的对象字典文件,需要用户编辑并添加自己的数据对象,创建的对 象字典如下所示。

Device Profile:	5001	Modular Devis	e Profile	Usage Notes:								
Modul Profile:	0			- The PDO mapping object and SyncMan	ager assignme	nt object	doesn t	need to be	defined	. In that c	ase they are	created automatically.
IndexIncrement	0			- Ine following objects are fixed in	cluded in the	SSC and s	snall no	t be derine	a in the	file : OXD	000, 0x1001,	0x1008, 0x1009, 0x100a, 0x1010
PdoIndexIncrement:	0			OXIOII, OXIOIS, OXIOFO, OXIOFI, OXIO	F3, 0x1c00, 0	x1c32, 0x1	1033					
				- Entries less or equal one SBit sha	11 not over18	ip byte boi	rders.					
				- Entries greater obit shall always	start at an e	xact woru	Doluei					
The object diction	nary defined	here shall b	e used compl	ementary with ETG. 5001 and ETG. 1000								
Index	ObjectCode	ST	DataType	 Name 	• Default 1	W/0/C	- B/S -	Access	• rr/ta •	CoeRead -	Coelirite .	Description
//0x04xx	Units		Ducutype	- Tune	Deruure	(14/ 0) 0	0/0	neccos	(A) (A) (B) (A)	cocheda	COCHITCC	beset iperon
,												
//0×8××	Enume (0x0800	- Overe										
/ 02022	Enders (0x0000	VALUE /										
//0×6nn×	Input Data of	the Medule ((1×6000 - 0×600	p)								
/ UAUMIA	input butu oi	the abduite (t		• /								
//0x700x	Output Doto o	f the Medule i	(0x7000 - 0x7E	PP)								
/ VATILIA	output pata o	i ene abuaic .	021000 0211	117								
(/0.0	0.01.01.01	D (C ())	. 1.1 (0.0000	0.0777)								
/ox8nnx	Configuration	Data of the M	todule (0x8000	- 0X8FFF)								
//0x9nnx	Information D	ata of the Moo	iule (0x9000 -	0x9FFF)								
//0xAnnx	Diagnosis Dat	a of the Modul	le (0xA000 - 0	xAFFF)								
//0xFxxx	Device Object	s (0xF000 - 0)	(FFFF)									
0xF000	RECORD			Modular Device Profile		m						
			1 UINT16	Index distance	0x10	m		ro				
			0.1111/11/0	Manimum numbers of modules		1. C						

图 2.17 Excel 空白文件

导入: Tool -> Application -> Import:

若已有 Excel 对象文件,则只需要导入即可。

2.5.2 Excel 文件编写规则

Index

在 CANopen 协议中,索引 (Index) 是对象字典中的一个关键概念。对象字典是 CANopen 设备中用于存储配置和过程数据的表,每个条目都由一个 16 位的索引和一个 8 位的子索引 (Sub-index) 组成。

1. 索引的作用:

- 唯一标识: 索引用于唯一标识对象字典中的一个特定对象;
- •数据访问:通过索引和子索引,可以访问和修改设备的参数、状态或配置信息;
- •标准化通信: CANopen标准定义了一些必须实现的索引,以确保设备之间的兼容性和通信。
- 2. 索引的范围和分配:

标准范围: CANopen 协议规定了一些标准的索引范围,例如:

• 0x0001 - 0x001F: 静态数据类型;



- 0x1000 0x1FFF: 通信对象子协议区;
- 0x2000 0x5FFF: 制造商特定子协议区。

制造商特定:制造商可以在指定的范围内定义自己的索引来满足特定设备的 需求。

3. 应用实例:

- 设备配置: 例如, 设备类型 (索引 0x1000) 和设备名称 (索引 0x1008) 等信息都存储在对象字典中。
- 控制信号: 某些索引可以用于发送控制信号, 如启动或停止设备。

通过索引和子索引的组合,CANopen协议实现了设备之间的高效通信和数据交换

ObjectCode

在 CANopen 协议中,对象代码 (Object Code) 用于描述对象字典中对象的内部结构类型。对象代码定义了对象的数据结构,以便于正确地访问和处理这些数据。以下是 CANopen 中定义的几种主要对象代码:

ObjectCode	描述
VARIABLE	表示一个简单的变量,通常用于存储单一的数据值。
ARRAY	表示一个数组结构,数组中的所有元素具有相同的数据类型。例如,一个数组可以
	包含多个相同类型的值,如多个温度读数。
RECORD	表示一个记录结构,类似于 C 语言中的结构体。记录中的各个元素可以有不同的
	数据类型。例如,一个记录可以包含温度、压力和湿度等不同类型的值。

对象代码在 CANopen 设备的电子数据表 (EDS) 文件中被定义为"对象类型"。通过这些定义, CANopen 协议能够确保设备之间的兼容性和数据交换的准确性。

SI

在 CANopen 协议中, 子索引 (Subindex) 是用于进一步细分对象字典中某个索引 (Index) 下的数据结构的标识符。每个对象字典条目由一个 16 位的索引和一个 8 位的子索引组成。 子索引的范围是从 0x00 到 0xFF。

- 1. 子索引的作用:
 - •数据细分:对于复杂的数据结构,如数组或记录,子索引用于访问这些结构 中的各个元素或字段;
 - 数量指示: 在数组或记录中, 子索引 0 通常用于指示该数据结构中元素的数量;
 - 唯一标识: 每个索引和子索引的组合在对象字典中是唯一的,确保了对特定 数据的准确访问。

2. 应用实例:

- •数组:例如,一个设备可能有一个温度传感器数组,索引为0x2000,子索引 0x00表示数组中元素的数量,而子索引0x01、0x02等分别表示每个传感器的 温度值;
- 记录: 对于一个记录结构,如设备状态信息,索引为 0x2001,子索引 0x01 可能表示温度,子索引 0x02 表示湿度。



AWorksLP

通过使用子索引,CANopen协议能够有效地管理和访问设备对象字典中的复杂数据结构,从而实现设备之间的高效通信和配置.

DataType

在 CANopen 协议中,数据类型 (DataType) 用于定义对象字典中对象的数据结构和格式。数据类型可以分为基本数据类型和复合数据类型:

- 1. 基本数据类型:
 - BOOLEAN(0x01): 布尔型, 表示一个逻辑值 (真或假);
 - INTEGER8(0x02): 8 位有符号整型;
 - INTEGER16(0x03): 16 位有符号整型;
 - INTEGER32(0x04): 32 位有符号整型;
 - INTEGER64(0x15): 64 位有符号整型;
 - UNSIGNED8(0x05): 8 位无符号整型;
 - UNSIGNED16(0x06): 16 位无符号整型;
 - UNSIGNED32(0x07): 32 位无符号整型;
 - UNSIGNED64(0x1B): 64 位无符号整型;
 - REAL32(0x08): 单精度浮点型, 32 位;
 - REAL64(0x11):双精度浮点型, 64 位。
 - ·····

2. 复合数据类型:

- VISIBLE STRING(0x09): 可见字符串,由8位有符号字符组成的字符串;
- OCTET_STRING(0x0A): 八位字符串,由 8 位无符号字符组成的字符串;
- UNICODE_STRING(0x0B): Unicode 字符串,由 16 位无符号字符组成的字符 串;
- TIME_OF_DAY(0x0C): 时间类型,表示一天中的时间;
- TIME DIFFERENCE(0x0D): 时间差类型,表示两个时间点之间的差值;
- DOMAIN(0x0F): 域类型,用于传输大量数据。
- 3. SSC(Excel) 中对数据类型的实现为:



AWorksLP

	TG. 2000 ETG. 1001 Part 5 /	ETG. 102	CoE data type index (deftype)	CoE data type name	base data type (ESI, ENI name)	Bit size	Description	Range	lassification	Classification Group
1		¥ .	• J		-		101 D4100		•	
2	x x	х	0x0001	BOOLEAN	BIT	1	0': FALSE '1': TRUE		Base Data Type	-
3	x x	х	0x0002	INTEGERS	SINT	8	Short Integer	-128 to 127	Base Data Type	Signed Integer
4	x x	х	0x0003	INTEGER16	INT	16	Integer	-32 768 to 32 167	Base Data Type	Signed Integer
5	x x	X	0x0004 0x0005	INTEGER32 INSIGNED8	DINT	32	Double Integer Unsigned Short Integer	-2 ³¹ to 2 ³¹⁺¹ 0 to 255	Base Data Type	Signed Integer
7	x x	x	0x0006	UNSIGNED 16	UINT	16	Unsigned Integer / Word	0 to 65 535	Base Data Type	Unsigned Integer
8	x x	x	0x0007 0x0008	UNSIGNED32 REAL32	UDINT REAL	32	Unsigned Double Integer Floating point	0 to 2 ³²⁻¹	Base Data Type	Unsigned Integer Floating
		Â		1212010	100100				Bage Data Type	
	x x	x	0x0009	VISIBLE_STRING	STRING (n)	8*n	VisibleString (1 octet per character)		with variable	Strings
10									Base Data Type	
	x x	х	0x000&	OCTET_STRING	ARRAY [0n] OF BYTE	8*(n+1)	Sequence of octets (data type BYTE)		with variable	Octet Field
					ARRAY [0, m] OF	16a (n+			Base Data Type	
12	x x	x	0x000B	UNICODE_STRING	UINT	1)	Sequence of UINT		with variable length	Octet Field
13	x		0x0000	TIME_OF_DAY						
14	x		0x0008	reserved for future use						
16	z		0x000F	DOMAIN	THEOR				D	Court Televis
18	x x x x	x	0x0010	REAL64	LREAL	64	Long float		Baze Data Type	Floating
19	x x	x	0x0012	INTEGER40	INT40	40			Base Data Type	Signed Integer
21	x x	x	0x0014	INTEGER56	INT56	56			Base Data Type	Signed Integer
22	x x	x	0x0015	INTEGER64	LINT	64	Long Integer	-2 ⁸³ to 2 ⁸³⁻¹	Base Data Type	Signed Integer
24	x		0x0017	reserved for future use	UNITA				Date Data type	WINSTONA THEORY
25	x x	x	0x0018	UNSIGNED40 INSIGNED48	UINT40 UINT48	40			Base Data Type	Unsigned Integer
27	x x	x	0x001A	UNSIGNED56	UINT56	56			Base Data Type	Unsigned Integer
28	x x	x	0x001B 0x001C	UNSIGNED64	OLINT	64	Unsigned Long Integer	0 to 2**-1	Base Data Type Other	Unsigned Integer
	z	Y	0x001D	GUID	GUID	128	globally unique identifier with a		GUID Data Type	GUID Type
30	x x	x	0x001E	BITE	BYTE	8	Length of 126 Dit Octet		Baze Data Type	Octet Field
32		x	0x001F	WORD Description	WORD DWORD	16	Two Octets		Base Data Type	Octet Field
33		x	JAUVEU	= + JW		32	For PDO_MAPPING definition see		Pre-defined	
	x	x	0x0021	FDO_MAPPIN;			ETG. 1000 For Record usage in ESI see		Record	
34				received for connetibility			ETG. 2000		DEFSTRUCT	
35		x	0x0022	reasons (D\$301)			Des TADATTER J.C.			
			0x0023	IDENTITY			ETG. 1000		Pre-defined Record	
36		î.					For Record usage in ESI see ETG.2000		DEFSTRUCT	
37			0x0024	reserved for future use			For COMMAND PAR definition re-			
	x	х	0x0025	COMMAND_PAR			ETG. 1000		Fre-defined Record	
38							ETG. 2000		DEFSTRUCT	
39		х	0x0028	reserved for future use			For PDO PARAMETER definition zee			
		х	0x002T	FDO_PARAMETER			ETG. 1020 For Becord wrate in FSI res		Record	
40							ETG. 2000		DEFSTRUCT	
			0+0028	ENTIM			ETG. 1020		Pre-defined Record	
41							For Record usage in ESI see ETG. 2000		DEFSTRUCT	
							For SYNC_PAR definition see ETG 1000		Pre-defined	
42	x	x	0x0029	SM_SYNCHMONISATION			For Record usage in ESI see		DEFSTRUCT	
			0.0001	PR-2-72			No pre-defined Record structure.			
43		x	URUUZA	n2con0			for Records.		Record	
			0-0025	BACKIER DADAMETER			For BACKUP_PARAMETER definition see ETG.1020 (t.b.d), Object 0x10F0		Pre-defined	
44		^	0,0012	Control _1 Apple 1111			For Record usage in ESI see ETG, 2000		DEFSTRUCT	
							For MODULAR_DEVICE_PROFILE definition rea ETG 5001 Object		Prendefined	
		x	0x002C	MODULAR_DEVICE_PROFILE			0xF000		Record	
45							For Record usage in ESI see ETG. 2000		DEFSTRUCT	
46	x x	x	0x0020 0x0028	BITARRS BITARR16	BITARRS BITARRIS	8	8 individual Bits		Base Data Type Base Data Type	Bit String Bit String
48	x x	x	0x002F	BITARR32	BITARR32	32	32 individual Bits		Base Data Type	Bit String
49 50	x x x x	x	0x0030	BIT2	BIT2	2	2-Bit		Bage Data Type	Bit Field
51	x x	x	0x0032	BIT3	BIT3	3	3-Bit		Baze Data Type	Bit Field
52	x x x x	x	0x0033 0x0034	BITS	BITS	- 4	4-Bit 5-Bit		Baze Data Type Baze Data Type	Bit Field Bit Field
54	x x	x	0x0035	BIT6	BIT6	6	6-Bit		Base Data Type	Bit Field
56	x x	x	0x0030	BITS	BITS	8	8-Bit		Base Data Type	Bit Field
57	x		0x0038 - 0x0035	reserved for future use						
58	x		0x40-0x005F	Complex Data Type						
35	^		040000-04020F	service riorite o i specific		16+(n+			Baze Data Type	
60		x	0x0260	AAAAT_OF_INI	ARKAI [U] OF INI	1)	sequence of INI		with variable length	octet Field
		x	0x0261	ARRAY_OF_SINT	ARRAY [0n] OF	8* (n+1	Sequence of SINT		Base Data Type with variable	Octet Field
61					2101	,			length Base Data Tune	
62		x	0x0262	ABRAY_OF_DINT	ARRAY [0n] OF DINT	32*(n+ 1)	Sequence of DINT		with variable	Octet Field
02				IDDUIT ON IDDUIT	ARRAY [0n] OF	32* (n+	0 0.007W		Base Data Type	
63		x	0x0263	ANNAY_OF_UDINT	UDINT	1)	Sequence of UDINI		with variable length	Octet Field
64			0x0254-0x0280	reserved			For DEFINER ERRORMANDI ING			
			0x0281	ERROR SETTING			definition see ETG. 1020, Object		Pre-defined Record	
		x	030201	Press/10-20111304			For Record usage in ESI see		DEFSTRUCT	
65							For DEFTYPE_DIAGHISTORY definition		Pre-defined	
		х	0x0282	DIAGNOSIS_HISTORY			see EIG. 1020, Object Ox10F3 For Record urage in ESI see		Record	
66							ETG. 2000 For DEFTYPE STRESTATUS definition		DEPSIROUT	
		x	0x283	EXTERNAL_SYNC_STATUS			zee EIG. 1020, Object Ox10F4		Fre-defined Record	
67							ETG. 2000		DEFSTRUCT	
			0x284	EXTERNAL SYNC SETTINGS			For DEFITYE_SYNCSEITINGS definition see ETG. 1020, Object Dx10F5		Pre-defined Becord	
68			UALON .	CULTURADE_0100_00111000			For Record usage in ESI see ETG. 2000		DEFSTRUCT	
							For DEFTYPE_FSOEFRAME definition see EIG.5120		Pre-defined	
69		x	0x0285	URFTYPE_FSOEFRAME			For Record usage in ESI see		Kecord DEFSTRUCT	
01							For DEFTYPE_FSOECOMMPAR definition		Pre-defined	
		x	0x0286	DEFTYPE_FSOECOMMPAR			see EIG.5120 For Record usage in ESI see		Record	
70			0x0287-0x0799	reserved			ETG. 2000		- ar o mort	
72	×		0x0800-0x0FFF	Enumerated Data Type Area						
73 74										
75										
77										
78 79										
80										
81 82										
00										

图 2.18 数据类型规范

4. 应用:

这些数据类型在 CANopen 设备的对象字典中被广泛使用,用于定义设备的 各种参数和状态信息。例如,设备温度可能使用 REAL32 类型来表示,而设 备状态标志可能使用 BOOLEAN 类型。通过这些数据类型,CANopen 协议 能够确保设备之间的数据交换是准确和一致的。

Name



AWorksLP

在 CANopen 协议中,"name"通常指的是对象字典中的对象名称。对象的名称用于描述该对象的功能或用途,例如设备类型、错误寄存器、制造商设备名称等。例如,在对象字典中,索引 1008h 的对象名称是"Manufacturer device name",表示制造商设备名称。这些名称在 EDS 文件 (Electronic Data Sheet) 中也有详细的定义和描述。

Default

在 CANopen 协议中," default" 通常指的是设备或网络的默认设置或参数值。这些默认 值在设备启动或重置时被使用,以确保设备能够以一种已知和预期的状态运行。

以下是一些常见的 CANopen 默认设置:

- 节点 ID: 节点 ID 的默认值通常由设备制造商设定,但可以通过 LSS(Layer Setting Services)进行更改。
- PDO(过程数据对象)参数:
- •传输类型: PDO 的传输类型也有默认设置,例如默认为事件触发 (Type 255)。

这些默认设置确保了设备在启动时能够以一种标准化的方式运行,并且可以通过网络 管理工具进行配置和优化。

M/O/C

М	Mandatory(强制的)
0	Optional(可选的)
С	Conditional(条件性的)

表示一个索引的实现属性,例如在 ETG5001 规范中,索引 0xF000 为必须实现的对象 (Mandatory)



AWorksLP

Index	Name	Fieldbus Gateway	Modular Device	Module Device
0xF000	Modular Device Profile	Μ	М	М
0xF002	Detect Modules Command	С	С	-
0xF00E	Module PDO Group Mapping Alignment PDO Number	С	С	-
0xF00F	Module PDO Group Mapping Alignment	С	С	-
0xF0100xF01F	Module Profile List Profile information of the modules	С	С	C M if modules have different profiles
0xF0200xF02F	Address list of the configured Modules	Μ	С	-
0xF0300xF03F	Module Ident list of the configured Modules		0	
0xF0400xF04F	Address list of detected Modules	C M if 0xF002 is supported	С	-
0xF0500xF05F	Module Ident list of detected Modules		0	
0xF0600xF06F	Profile Version list of detected Modules	0	0	0
0xF1000xF10F	Device Status (TxPDO mappable)	С	С	С
0xF1100xF11F	Device Diagnosis	С	С	С
0xF1200xF12F	Device Statistics	С	С	С
0xF2000xF20F	Device Control (RxPDO mappable)	С	С	С
0xF5000xF5FF	Vendor/manufacturer specific device objects	0	0	0
0xF6000xF6EF	Input Area of device itself (TxPDO mappable)	С	С	С
0xF6F00xF6FF	Input Area of device itself (TxPDO mappable) according ETG.5003.1	0	0	0
0xF7000xF7EF	Output Area of device itself (RxPDO mappable)	С	С	С

Table 5: Object Areas of the Device

B/S

暂时未知

Access

在 CANopen 协议中,指的是对设备对象字典 (Object Dictionary) 中参数的访问权限。

在 EtherCAT 中的访问权限类型:

权限	描述
ro	只读
wr	只写
rw	可读可写
preop_ro/wr/rw	只能在 preop 阶段进行,只读/只写/可读可写
safeop_ro/wr/rw	只能在 safeop 阶段进行,只读/只写/可读可写
op_ro/wr/rw	只能在 op 阶段进行,只读/只写/可读可写



©2025 Guangzhou ZHIYUAN Electronics Co., Ltd.

User Manual

AWorksLP

1. 访问方式:

- SDO(服务数据对象):用于访问对象字典中的参数。SDO 允许设备之间进行 点对点的通信,以读取或写入对象字典中的数据。例如,一个主设备可以通 过 SDO 读取从设备的参数值,或者修改从设备的参数设置;
- PDO(过程数据对象):用于传输实时数据,如传感器读数或执行器控制信号。 PDO 的数据传输是基于事件或周期性的。

2. 访问控制:

- NMT(网络管理):负责管理设备的状态和访问权限。NMT 可以控制设备的启动、停止和复位等操作;
- 安全机制:某些设备可能实现额外的安全机制,以限制对特定参数的访问。例如,只有在设备处于特定的 NMT 状态时,才能修改某些关键参数。

通过这些访问权限和方式,CANopen协议确保了设备之间的有效通信和数据交换,同时提供了灵活的设备管理和配置手段。

rx/tx

表示此对象是否可以进行 PdoConfig。

- rx: Output 数据标识;
- tx: Input 数据标识。

标识过的对象,可以在主站上进行 Input 或 Output 的数据分配。

CoeRead/CoeWrite

coe 读写函数回调定义,在 Excel 中定义 CoeRead/CoeWrite 的函数名称,例如:

Index	▼ ObjectCode ▼	SI	DataType	Name	▼ Default \	M/0/C 🔻	B/S 🔻	Access	rx/tx 🔻	CoeRead	• CoeWrite •	Description
//0x04xx	Units											
//0x8xx	Enums (0x0800	- 0xFFF)										
//0x6nnx	Input Data of	the Module (0	x6000 - 0x6FFF)									
110-2	0	abs Madula /	0-2000 0-200									
0x7000	RECORD	the module (0x7000 - 0x7FF	.)						Coe0x7000E	Re Coe0x7000W	ite
			1 UINT8	var1								
//0x8nnx	Configuration	Data of the M	lodule (0x8000 -	- 0x8FFF)								
//0x9nnx	Information Da	ta of the Mod	ule (0x9000 - 0	x9FFF)								
//Ortoor	Diognosia Data	of the Medul	a (0x4000 - 0x)	(PPP)								
// VANIIIA	Diagnosis Data	or the mount	.e (0xA000 - 0x/	urr)								
//0xFxxx	Device Objects	(0xF000 - 0x	FFFF)									
0xF000	RECORD			Modular Device Profile		m						
			1 UINT16	Index distance	0x10	m		ro				
		:	2 UINT16	Maximum number of modules		m		ro				

生成代码时,会以定义的函数名称生成一个空白函数(需用户实现),在 coe 进行读写时 便会调用此函数以代替默认的 coe 读写规则。

///////////////////////////////////////	///////////////////////////////////////						
⇔/////							
/**							
\param	index	index of the requested object.					
\param	subindex	subindex of the requested object.					
\param	objSize	size of the requested object data, calculated with					
⊶OBJ_GetO	ojectLength						
\param	pData	Pointer to the buffer where the data can be copied to					
\param	bCompleteAccess	Indicates if a complete read of all subindices of the					
		object shall be done or not					



User Manual

EtherCAT 从站参考手册

AWorksLP

```
\return
         result of the read operation (0 (success) or an abort code (ABORTIDX_...
\hookrightarrow. defined in
          sdosrv.h))
→//////
UINT8 Coe0x7000Read(UINT16 index, UINT8 subindex, UINT32 dataSize, UINT16 MBXMEM *-

→pData, UINT8 bCompleteAccess) {

#if WIN32
#pragma message ("Warning: Implement CoE read callback")
#else
#warning "Implement CoE read callback"
#endif
return 0;
}
→//////
/**
                         index of the requested object.
        index
\param
\param
        subindex
                         subindex of the requested object.
                         size of the requested object data, calculated with.
         objSize
\param
→OBJ_GetObjectLength
         pData
                          Pointer to the buffer where the data can be copied to
\param
                         Indicates if a complete read of all subindices of the
\param
         bCompleteAccess
                          object shall be done or not
         result of the read operation (0 (success) or an abort code (ABORTIDX_...
\return
\hookrightarrow. defined in
          sdosrv.h))
→//////
UINT8 Coe0x7000Write(UINT16 index, UINT8 subindex, UINT32 dataSize, UINT16 MBXMEM *_

→pData, UINT8 bCompleteAccess) {

#if _WIN32
#pragma message ("Warning: Implement CoE write callback")
#else
#warning "Implement CoE write callback"
#endif
```

```
}
```

return 0;

注: 函数类型: xxxRead/xxxWrite(UINT16 index, UINT8 subindex, UINT32 dataSize, UINT16 MBXMEM * pData, UINT8 bCompleteAccess) 例:

Index 0x7000								
Subindex	DataType	Name						
1	UINT8	Var1						



AWorksLP

User Manual

		续上表						
Index 0x7000								
2	UINT8	Var2						
3	UINT8	Var3						
4	UINT8	Var4						

主站通过 sdo 对 0x7000 的 0x03 索引写值时,协议栈会调用 xxxWrite,此时函数参数

- index: 0x7000
- subindex: 0x03
- dataSize: sizeof(UINT8)
- pData: 写的值
- bCompleteAccess: 是否完全访问

2.5.3 周期数据 (Input/Output)

1. 周期数据的总数据量必须为 Byte 的整数倍,例如:

		Ī	确示	例					错误示例					
ndex	Object	Coc SI	[DataType	Name		Index	Object(CocSI		DataType	e Name	l
//0x7nnx 0x7000	Output RECORD	Data	ı of	the	Module	(0x7000 -	0x7FFF)	//0x7nnx 0x7000	Output RECORD	Data	of tl	he Module	(0x7000 -	-
				1	BOOL	Var1						1 BOOL	Var1	
				2	BOOL	Var2						2 B00L	Var2	
				3	BOOL	Var3						3 BOOL	Var3	
				4	BOOL	Var4						4 BOOL	Var4	
				5	BOOL	Var5						5 BOOL	Var5	
				6	BOOL	Var6						6 BOOL	Var6	
				7	BOOL	Var7						7 BOOL	Var7	
				8	BOOL	Var8						8 BOOL	Var8	
				9	UINT8	Var9						9 BOOL	Var9	
				10	UINT16	Var10					1	0 BOOL	Var10	
		总	十:3E	Byte	(字节对齐	-)				总计:	10Bi	t(字节未对	齐)	
	8Bit	11	Byte		2B	yte			8Bit	1Bi	t	1Byte		

如图所示,正确实例中的总数据量为 3Byte 是 Byte 的整数倍故,而错误示例中的总数 据量为 10Bit 数据长度未对齐 Byte. 错误的长度定义会导致 SSC 工具检查的报错。

2. 同一个主索引下的周期数据存在对齐 (以最大的数据类型),对齐过程中不允许产生多 余字节:

	正确示例								错误示例							
Index	Object(CoćS	Ι		DataType	Name			Index	Object	CoćSI	[DataType	Name	
//0x7nnx	Output	Dat	a of	' the	Module	(0x7000 -	0x7FFF)		//0x7nnx	Output	Data	a of	the	Module	(0x7000 -	0x7FFF)
0x7000	RECORD								0x7000	RECORD						
				1	BOOL	Var1							1	BOOL	Var1	
				2	BOOL	Var2							2	BOOL	Var2	
				3	BOOL	Var3							3	BOOL	Var3	
				4	BOOL	Var4							4	BOOL	Var4	
				5	BOOL	Var5							5	BOOL	Var5	
				6	BOOL	Var6							6	BOOL	Var6	
				7	BOOL	Var7							7	BOOL	Var7	
				8	BOOL	Var8							8	BOOL	Var8	
				9	UINT8	Var9							9	UINT16	Var9	
				10	UINT16	Var10							10	UINT16	Var10	
内存分布					_	内存分布										
	8Bit			8Bit		8Bit 8				8H	Bit//对齐产	主				
	16Bit							16Bit								
									16Bit							



AWorksLP

错误示例中由于前八个成员总数居量为 8Bit,而第九个成员数据宽度为 16Bit,此时会因对齐产生多余的 8Bit,此举在 SSC 中同样不允许。

Description:

• 描述

```
—— 注意 -
```

▲ Excel 编辑对象是基于 CanOpen 协议规则

2.6 源码和 ESI 文件生成

保存项目后点击 **Project** -> **Create new Slave Files**,最后会生成一份 xml 文件 (ESI) 和 Src 源码,若要修改对象字典只需重新打开工程,重复之前步骤 Import 要导入的 Excel 文件 即可。

2.7 应用程序与接口函数

除固定协议栈代码以外,SSC还会依据用户导入的Excel生成相关的应用文件(源码名 会与Excel文件名相关),以 8io 为例:

应用相关的代码会生成于 8io.c/8io.h/8ioObjects.h 这些文件中, 8io.c 中存在三个已定义 但内容未实现的应用函数 (需要用户实现),这三个函数会在协议栈处理数据阶段调用。

2.7.1 void APPL_InputMapping(UINT16* pData)

- Description: 数据发送时的回调函数 (EtherCAT Slave to Master)
- Parameter(pData): 指向发送缓冲区的指针。
- Usage: pData 是用作数据发送的缓冲区,用户将数据填充入 pData 后,协议栈会将 其发送往主站.pData 的长度在 safeop 阶段就会被计算,并将其长度信息存储在变量 nPdInputSize 中。

```
void APPL_InputMapping(UINT16* pData)
{
    extern TOBJ6000 PDIChannel0x6000;
    memcpy(
    pData,
    (uint8_t*)(&PDIChannel0x6000) + sizeof(PDIChannel0x6000.u16SubIndex0),
    1);
}
```



AWorksLP

2.7.2 void APPL_OutputMapping(UINT16* pData)

- Description: 数据接收时的回调函数 (EtherCAT Master to Slave)。
- Parameter(pData): 指向接收缓冲区的指针。
- Usage: pData 是用作数据接收的缓冲区,协议栈将数据从 ESC 中取出后会存储在 pData 中等待处理.pData 的长度在 safeop 阶段就会被计算,并将其长度信息存储在变 量 nPdOutputSize 中。

```
void APPL_OutputMapping(UINT16* pData)
```

```
{
    /* PD0Channel0x7000 include a variable in type UINT16,
    * boolean variables are not allow to take address,
    * so we move two bytes to the address of PD0Channel0x7000.LED1,
    * attention:
    * Guaranteed byte alignment,if PD0 use a boolean type
    */
    extern TOBJ7000 PD0Channel0x7000;
    memcpy(
    (uint8_t*)(&PD0Channel0x7000) + sizeof(PD0Channel0x7000.u16SubIndex0),
    pData,
    1);
}
```

2.7.3 void APPL_Application(void)

- Description: 周期性循环函数。
- Usage: 此函数会被周期性调用,在于同步模式下,APPL_Application 的调用会处于 APPL_OutputMapping 与 APPL_InputMapping 之间,通常用于数据接收后进行硬件输入并采集硬件的反馈。

```
void APPL_Application(void)
```

{

```
HAL_GPI0_WritePin(LED1_GPI0_Port,LED1_Pin,!(PD0Channel0x7000.LED1));
HAL_GPI0_WritePin(LED2_GPI0_Port,LED2_Pin,!(PD0Channel0x7000.LED2));
HAL_GPI0_WritePin(LED3_GPI0_Port,LED3_Pin,!(PD0Channel0x7000.LED3));
HAL_GPI0_WritePin(LED5_GPI0_Port,LED4_Pin,!(PD0Channel0x7000.LED4));
HAL_GPI0_WritePin(LED5_GPI0_Port,LED5_Pin,!(PD0Channel0x7000.LED5));
HAL_GPI0_WritePin(LED6_GPI0_Port,LED6_Pin,!(PD0Channel0x7000.LED6));
HAL_GPI0_WritePin(LED7_GPI0_Port,LED7_Pin,!(PD0Channel0x7000.LED7));
HAL_GPI0_WritePin(LED8_GPI0_Port,LED8_Pin,!(PD0Channel0x7000.LED7));
HAL_GPI0_WritePin(LED8_GPI0_Port,LED8_Pin,!(PD0Channel0x7000.LED8));
memcpy(
(uint8_t*)(&PDIChannel0x6000) + sizeof(PDIChannel0x6000.u16SubIndex0),
(uint8_t*)(&PD0Channel0x7000) + sizeof(PD0Channel0x7000.u16SubIndex0),
1);
```



AWorksLP

	注意	_
\wedge	•	ŧ

- 协议栈配置好后,SSC 会将不必要的配置项和文件过滤,最后生成一份"剪裁"
 后的代码,所以当修改协议栈的配置项或者是对象字典时,需要由 SSC 重新生成,而不是修改 ecat_def.h 文件;
 - •代码和 ESI 文件需要配套使用 (指同时生成的一套);
 - 查看协议栈的完整源码可参考安装包目录下的 SlaveFiles.zip 压缩包。



3. TwinCAT 操作及使用

- 注意 --

- DPort-ECT 相关所有例程测试主站均为 TwinCAT3
- 倍福 TwinCAT3 官方下载链接²

² https://www.beckhoff.com.cn/zh-cn/support/download-finder/search-result/?download group=97028248

3.1 TwinCAT 下载及安装

3.1.1 TwinCAT3 硬件配置要求

实时模式下,TwinCAT 对 PC 的处理器以及网卡型号有特殊要求,处理器和网卡要须是 Intel,具体网卡型号可通过倍福官方网站进行验证。网卡型号验证链接地址^[3]

如果只是使用 TwinCAT 做一些非实时性的验证,大部分处理器和网卡也是支持的。

3.1.2 TwinCAT3 安装步骤

网页链接: TwinCAT3.1(4026) 官方安装文档^[4]

3.1.3 TwinCAT3 网卡配置

第一次安装 TwinCAT, 需要对网卡进行设置,设置网卡步骤如下:

1. 点击 TwinCAT -> show Realtime Ethernet Compatible Device。



^[3] https://infosys.beckhoff.com/english.php?content=../content/1033/tc3_overview/9309844363.html&id= 1489698440745036069

^[4] https://tr.beckhoff.com.cn/mod/resource/view.php?id=2900

User Manual

 \times

EtherCAT 从站参考手册

AWorksLP

调试(D)	Twi	nCAT TwinSAFE PLC 团队(M) Scope 工具	Į(Τ)	窗囗(W) 帮助(H)					
9-9		Windows	•	اھ -					
× 🕲 🛛	>	Activate Configuration		·					
	#	Restart TwinCAT System							
	#	Restart TwinCAT (Config Mode)							
	\$	Reload Devices							
	14	Scan							
	۲	Toggle Free Run State		Channe Treast					
	\bigcirc	Show Online Data		Choose Target					
	20	Show Sub Items							
	ø	Hide Disabled Items							
		Software Protection							
	RE6	Access Bus Coupler/IP Link Register		Level					
	_	Update Firmware/FEPROM	-	Local V3.1 (Build 4024.10)					
		Show Realtime Ethernet Compatible Devices		2 Pin Version					
	-	The Handling							
		Selected Item	٠						
		EtherCAT Devices	•						
	٩	TcProjectCompare							
		Target Browser	٠						
		Bode Plot	٠						
		Filter Designer	•						
		About TwinCAT							
	_		-	-					

2. 选择与从站连接的网卡,点击安装:

Installation of TwinCAT RT-Ethernet Adapters





AWorksLP

3.2 新建 TwinCAT 工程

• 1. 文件 -> 新建 -> 项目



图 3.1 新建工程

 2. TwinCAT Projects -> TwinCAT XAE Project(XML format),修改项目名称与存 放路径,点击确认。

新建项目								?	×	
▶ 最近		排序依据: 默认值	•	# E		搜索(Ctrl+E)			ρ-	
▲ 已安装		TwinCAT XAE P	roject (XML form	mat)	TwinCAT Projects	类型: TwinC	AT Projects			
 TwinCAT Measur TwinCAT Projects TwinCAT PLC TcXaeShell Soluti 	ement s					TwinCAT XA Configuratic	E System Mar n	nager		
未找到你要查 打开 Visual Stu	我的内容? idio 安装程序									
名称(N):	TwinCAT Project3									
位置(L):	C:\Users\wenwe\D	ocuments\TcXaeShell			-	浏览(B)				
解决方案(S):	创建新解决方案				•					
解决方案名称(M):	TwinCAT Project3						✓ 为解决方案创建目录(D)			
						── 添加到源代码	3管理(U)			
							确定	取消	٤	

图 3.2 新建工程



AWorksLP

• 3. 执行以上两个步骤,工程界面如下所示。

TwinCAT Project3 - TcXaeShell	🗸 🚱 快速启动 (C	trl+Q) 🔑 🗕 🗗 🗙
文件(F) 编辑(E) 视图(V) 项目(P) 生成(B) 调试(D) TwinCAT TwinSAFE PLC 团队(M) Scope 工具(T) 窗口(W) 帮助(H	i)	
G - O 🎦 - 🎦 - 😩 🗎 🎽 🐰 印 合 🤊 - ペー・ Release 🔹 TwinCAT RT (x64) 🔹 🕨 附加	- 🖉 🔎 💭 🚔 🎎 🚳) D + _
🔋 Build 4024.62 (Loaded) 📲 🚽 🔛 🧧 🚺 🥏 🔨 🚳 🐚 🐾 🄏 🛛 TwinCAT Project3 🔹 <local> 🔹 🚽</local>		a 首 ひ ひ 。
解决方案资源管理器 - · · · ×		
捜索解決方案资源管理器(Ctrl+;) ・ ・		
😡 解决方案"TwinCAT Project3"(1 个项目) 4 🔐 TwinCAT Project3		
SYSTEM		
O SAFETY		
We C++		
ANALYTICS		
A 🔤 I/O		
and the second se		
	10	- ₩ X
整个解决方案 ▼ 100 错误0 A 警告0 0 消息0 Clear 生成 + Int	alliSense -	授家領境列表
100.99	坝目	X19 (T
解决方案资源管理器 团队资源管理器 错误列表 輸出 属性 工具箱		
□ 正在创建项目"TwinCAT Project3"项目创建成功。		↑ 添加到源代码管理 ▲

图 3.3 工程界面

3.3 试用版授权激活

• 1. 双击 License, 弹出如下界面。

WinCAT Project3 - TcXaeShell 文件(F) 編輯(E) 掲題(V) 現員(P) 生成(B) 講題 ※ ○ · ○ ⑤ · * 10 · * 10 ※ ● · ○ ⑦ · 10 ※ ● · ○ ⑦ · 10 ※ ● · ○ ◎ · 10 ※ ● · ○ ● ·	t(D) TwinCAT TwinSAFE - C - Release - T S ()	PLC 团队(M) Scope winCAT RT (x64) ▶ 即 oject3 <local></local>	工具(T) 宮口(W) 帮助(H) 問加	■ 	マ P 快速部 いし P 日本 2 - 〇 F 日 - 1 〇 I -)(Ctrl+Q) P - 8 : (6 回 合 出 当 つ つ -
	TwinCAT Project3 • × Order Information (Runti License Device T System Id: 8F09FB01-A2F3-FB6C License Request Uconse Request Provider: Beckhi License Id: Comment: License Activation T Days Triation 7 Days Triation 1	ne) Manage Licenses Project urget (Hardware Id) -9228-DEE43D566F84 ff Automation Customer License_	Licenses Online Licenses Licenses Online Licenses Add. Platform: other (91) Generate File License Response File			
undereits.	Order No TC1000 描述列表 種心力素 種心力素	Ucense TC3 AD5	調表 0) Clear (生成 + IntellSen	Instances cpu license se •	Ucense TAN 72日	Current Status missing 文件 文件 行

图 3.4 激活界面

• 2. 点击 Manage License, 选择需要的 License 添加。



EtherCAT 从站参考手册 AWorksLP

User Manual

WinCAT Project3 - TcXaeShell 文件(F) 線積(E) 視意(Y) 项目(P) 生成(B) 调证 ○ - 〇 〇 〇 〇 - 〇 - 〇 〇 〇 〇 〇 〇 〇 〇 〇 〇 〇	試(D) TwinCAT TwinSAFE PLC 団 ・ へ・ Release ・ TwinCAT RT (xi の ゆう。 G& TwinCAT Project3 ・	L(M) Scope III(T) ₩□(W) ₩₩0(H) 50 - ▶ ₩102 β# - clocal>		. 8 ×
	TwinCAT Project3 * Order Information (Runtime Manage Disable automatic detection of req Order No TC1000 TC1100 TC1200 TC1200 TC1200 TC1200 TC1200 TC1200 TC1200 TC1200 TC1200 TC1250 TC1275 TC1276 TC1300 TC1701 TC1702 TC1900 TE1010 TE1030 TE1110	Licenses Project Licenses uired licenses for project License TC3 ADS TC3 ADS TC3 ADS TC3 PLC TC3 PLC TC3 PLC / C++ TC3 PLC / CPP 10 / NC I TC3 PLC / NC PP 10 / NC I / CNC-E TC3 PLC / NC PP 10 / NC I / CNC-E TC3 C+ TC3 UserMode Runtime - ExternalControl TC3 SPLC / TC3 Sector TC3 Sector TC3 Sector TC3 Sector TC3 UserMode Runtime - FastAsPossible TC3 Sector TC3 Realtime Monitor TC3 Sector TC3 Sector TC3 Sector TC3 Sector TC3 Realtime Monitor TC3 Sector TC3 Sector TC3 Sector TC3 Sector TC3 Realtime Monitor TC3 Sector TC3 Sector TC3 Sector TC3 S	Add License Image: Provide the set of the set	
	错误列表			
	整个解决方案 🔹 🕄 错误 0	▲ 警告 0 ① 消息 0 Clear 生成 + IntelliSense -	搜索错误列表	ρ-
经出方实际资源和188 1791 经206320192	说明 建油制度 给出 网络 工具箱		项目 文件	行

图 3.5 添加 License

• 3. 返回 Order Information (Runtime) 点击 7 Days Trial License



图 3.6 生成 7 天的 License

• 4. 出现如下界面则表示激活成功,超过使用期限后重复激活即可。



图 3.7 激活成功提示



3.4 TwinCAT ESI 文件烧录

 1. 将 SSC 配套生成的 ESI(.xml) 文件放置在 \${TwinCAT 安装目录}/3.1/Config/Io/ EtherCAT 目录下。

🖵 〉 此电脑 〉 OS (C:) 〉 TwinCAT 〉 3.1 〉 Config 〉 lo 〉 EtherCAT 〉						
□ ④ ⓒ □ へ 排序 ~	☰ 查看 ∽ ••••					
名称	修改日期	类型	大小			
📒 Beckhoff AX5xxx	2024/10/22 9:58	文件夹				
📒 Beckhoff AX8yxx	2024/10/22 9:58	文件夹				
ES RES	2024/10/22 9:58	文件夹				
🗾 8io.xml	2024/11/6 16:52	XML 文件	52 KB			
Beckhoff AMI8xxx.xml	2024/9/14 0:10	XML 文件	4,240 KB			
Beckhoff AMP86xx.xml	2024/9/14 0:10	XML 文件	1,704 KB			
Beckhoff AMP88xx.xml	2024/9/14 0:10	XML 文件	462 KB			
Beckhoff APS1xxx.xml	2024/9/14 0:10	XML 文件	224 KB			
Beckhoff APS4xxx.xml	2024/9/14 0:10	XML 文件	476 KB			

图 3.8 TwinCAT ESI 文件路径

 2. 回到 TwinCAT 点击 TwinCAT -> EtherCAT Devices -> Reload Device Description, 重新加载 ESI 文件。

Twi	nCAT TwinSAFE PLC 团队(M) Scope 工具((T) 窗口(W) 帮助(H)
	Windows	 ب
•>	Activate Configuration	
*	Restart TwinCAT System	
*	Restart TwinCAT (Config Mode)	
2	Reload Devices	
22	Scan	
۲	Toggle Free Run State	anced Settings
\bigcirc	Show Online Data	Configuration File
6 6-0	Show Sub Items	
%	Hide Disabled Items	Jnit Assignment
<u>æ</u>	Software Protection	Topology
RE6	Access Bus Coupler/IP Link Register	
	Update Firmware/EEPROM	Units (max) Likilization (%) Size (Duration (Max Id)
	Show Realtime Ethernet Compatible Devices	
	File Handling	▶ .000
	Selected Item	• .000 0.17 58 / 6.72 0
	EtherCAT Devices	Update Device Descriptions (via ETG Website)
Ø	TcProjectCompare	Reload Device Descriptions
8	Multiuser Explorer	Manage User Defined Whitelist
	Target Browser	Manage User Defined Blacklist
	AutomationML	>
	Bode Plot	
	Filter Designer	▶
	About TwinCAT	In Size Out Size E-Bus (mA)

图 3.9 重加载 ESI 文件

• 3. 点击 右键点击 Device -> Scan,出现任意方框点击是即可。



AWorksLP

解决方案资源管理器			- ₽ ×	TwinCAT Project	2 🕘 🗙 起始页
G O 🔒 🛗 🔹 🐻 🗸	a .	- 4		N Device	Туре
搜索解决方案资源管理器(Ct	rl+;)		، م		
 □ 解决方案"TwinCAT Project2 □ TwinCAT Project2 □ SYSTEM □ PLC □ SAFETY □ C++ □ VISION □ ANALYTICS □ I/O 	oject2	2"(1 个项目)			
Te Devices	*ם	添加新项(W)		Ins	
	* D	添加现有项(0	G)	Shift+Alt+A	A Contraction
	X	重命名(M)			
		Add New Fo	older		
		Export EAP	Config File		
	***	Scan			
	£	粘贴(P)		Ctrl+V	
		Paste with L	inks		

图 3.10 扫描设备

• 4. 弹出网卡扫描框后,若物理接线网卡出现√,则表示扫描到了设备,否则请检查 TwinCAT 的网卡设置,成功检测后继续点击 OK。

2 new I/O devices found	×
□Device 1 (EtherCAT Automation Protocol) [WLAN (MediaTek Wi-Fi 6E MT7922 160MHz Wit ✔Device 3 (EtherCAT) [以太网 (Realtek USB GbE Family Controller)]	OK Cancel
	Select All

图 3.11 网卡扫描结果



AWorksLP

出现 Activate Free Run, 点击是 (Y)

• 5. 此时界面如图下所示,对比从站名称,若与 ESI 文件中记载相同,则 ESI 文件成功烧录。

解决方案资源管理器	- ₽ ×	TwinCAT Pr	oject2	<mark>⇒ ×</mark> 起始页							
○ ○ 🏠 🖆 · 🐻 · 🗿 🏓 🗕		General	Adapte	r EtherCAT O	nline	CoE - Or	nline				
搜索解决方案资源管理器(Ctrl+;)	ρ-										
☑ 解决方案"TwinCAT Project2"(1 个项目)		NetId:		192.168.1.154.4	.1		A	dvanced Set	tings		
TwinCAT Project2 General SYSTEM MOTION		Datarat	e:	100 MBit/s			Expo	rt Configura	tion File		
							Syn	nc Unit Assig	nment		
5. C++								Topology	·		
VISION											
ANALYTICS		Eramo	Cmd	Addr	Lon	WC	Sunc Unit	Cyclo (mc)	Litilization (%)	Size / Duration (Map Id
▲ 🔁 I/O			LRD	0x09000000	1	VVC	Sync Onic	4.000	Othization (76)	Size / Duration (Map Iu
▲ ➡ Device 3 (EtherCAT) = 34		0	LRW	0x01000000	1	3	<default></default>	4.000			
📲 Image		0	BRD	0x0000 0x01	2	1		4.000	0.17	56 / 6.72	0
🕄 Image-Info									0.17		
SyncUnits											
Inputs											
Outputs											
InfoData											
▷ SS Box 1 (8io) ₩st											
Mappings											

图 3.12 设备

3.5 TwinCAT 主从栈通信

3.5.1 EtherCAT 状态机详解

在扫描出从站后,TwinCAT 会自动与从站建立通讯,点击 box -> Online 查看从站当前的状态。

TwinCAT Project3 - TcXaeShell										7 8	快速启动 (Ctrl+Q)	ρ.	- 0	×
又件(F) 编辑(E) 视图(V) 项目(P) 生成(B) 调调	式(D) IwinCAI Iwin	SAFE PLC 团队(P	I) Scope	⊥具(I)	图L1(V	V) 帮助(F	1)							
🔋 O • O 🎦 • 🖆 • 🖆 🗎 🗳 🗶 🗗 🖄 🤊	- C - Release	 TwinCAT RT (x64) 	- •	附加 *			م		-	1 I I I I I I I I I I I I I I I I I I I	💭 🏛 🏜 🕓 🖂 🛫			
Build 4024.62 (Loaded) 👻 🚽 🔛 🧾 💋 🛠 🔨	😂 🙋 🐾 🔏 🛛 TwinC/	AT Project3 🔹	<local></local>							÷ ~ ~		5 10 g		
解决方案资源管理器 ▼ ↓ ×	TwinCAT Project3 🗢 💈	ĸ												-
○ ○ ☆ ☆ · Ìo · ♂ / ≁ -	General EtherCAT	DC Process Data	a Pic St	artup Co	oE - Onlin	e Online	2							ъł
捜索解決方案资源管理器(Ctrl+;) ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・	State Machine						_							ш
 	Init	Bootstrap					_	_						-1
A 🧧 SYSTEM	Pre-On	Safe-On	Current	State:	OP			3						
License			Request	ed State:	OP									
P V V Real-Time	Op	Clear Error												
Routes														
Type System	Name	Online	Type	Size	>Addr	. In/Out	User ID	Linked to						
TcCOM Objects	LED1_status	0	BIT	0.1	39.0	Input	0							
MOTION	📌 LED2 status	0	BIT	0.1	39.1	Input	0							
PLC	LED3_status	0	BIT	0.1	39.2	Input	0							
SAFETY	📌 LED4_status	0	BIT	0.1	39.3	Input	0							
5- C++	LED5_status	0	BIT	0.1	39.4	Input	0							
	📌 LED6_status	0	BIT	0.1	39.5	Input	0							
ANALYTICS	LED7_status	0	BIT	0.1	39.6	Input	0							
4 🕎 1/0	📌 LED8 status	0	BIT	0.1	39.7	Input	0							
A Se Devices	🕶 WcState	0	BIT	0.1	1522.1	Input	0							
A Toppice 3 (EtherCAT)	🔊 InputToggle	1	BIT	0.1	1524.1	Input	0							
= Device 5 (Ethercent)	🕶 State	8	UINT	2.0	1548.0	Input	0							
♦∎ Image Infe	🔊 AdsAddr	192.168.1.154.4.1:	AMSADDR	8.0	1550.0	Input	0							
sunglinite	E>LED1	0	BIT	0.1	39.0	Output	0							
b b laguta	EP LED2	0	BIT	0.1	39.1	Output	0							
 Inputs Outputs 	E>LED3	0	BIT	0.1	39.2	Output	0							
V Gutputs	EPLED4	0	BIT	0.1	39.3	Output	0							
b the Box 1 (Sio) 1	-													
Ampings	错误列表												- - +	• ×
- mappings	整个解决方案	🔹 😧 错误 0 🔒	警告 0 🔢 🚺	消息 0	Clear	生成 + Int	elliSens	ie +			搜索	错误列表		- م
	说明									项目	文件		行	
													13	
经违方案资源等理器 因此资源等理论	# 20 素 給出 原始	工具箱												
2000人20年2408年4世的 图形页IR目标的	HE 04754X 制山 周性	1940												

图 3.13 从站状态

若成功到达 OP,则表示通信可以正常进行。 EtherCAT 的状态转换如下:



AWorksLP



图 3.14 从站状态

EtherCAT 状态机 (EtherCAT State Machine, ESM) 用于协调主站和从站应用程序在初始 化和运行时的状态关系及转换。

以下是 EtherCAT 状态机的各个状态及其功能:

- Init(初始化): 从设备上电后处于此状态。主站可以读取从站的设备信息并做一些准备进入预运行状态的配置。可以清除从站 FMMU 和 SyncManagers 配置,设置从站的地址和 mailbox 信息等。如果使用 DC(分布式时钟),可以设置 DC 的传输延时和启动的偏移时间,并进行若干次的时钟同步。
- Pre-Operational(预运行): 邮箱通信可用,但过程数据通信不可用。主站可以初始化用于过程数据的同步管理器通道、FMMU通道,以及(如果从站支持可配置映射)PDO映射或同步管理器 PDO分配。在此状态下,过程数据传输的设置以及可能与默认设置不同的终端特定参数也会被传输。
- Safe-Operational(安全运行): 邮箱通信和过程(输入)数据通信可用。在此状态下,只 有输入被评估;输出保持在"安全"状态。从站会检查用于过程数据通信的同步 管理器通道是否正确,如果需要,还会检查分布式时钟设置是否正确。
- Operational(运行): 过程数据输入和输出有效。从站程序读取输入数据,主站应用程 序发出输出数据,从站设备产生输出信号。
- Bootstrap(引导状态): 可选状态,但推荐使用,特别是在需要固件更新时。无过程数据通信,仅通过邮箱在应用层通信可用。可以进行特殊的邮箱配置,例如更大的邮箱大小。通常使用 FoE(File Access over EtherCAT)协议进行固件下载。

状态转换:

从初始化状态到运行状态的转换必须按照"初始化->预运行->安全运行->运行" 的顺序进行。从运行状态返回时可以越级转换。所有的状态改变都由主站发起, 主站向从站发送状态控制命令请求新的状态,从站响应此命令,执行所请求的状 态转换,并将结果写入从站状态指示变量。如果请求的状态转换失败,从站将给 出错误标志



AWorksLP

3.6 TwinCAT 实时模式

当 SM-Sync 模式到达 op 状态后,我们就可以开始测试 DC-Sync 模式了。

1. 右键点击 Tasks,选择添加新项



2. 选择 TwinCAT Task With Image 后点击 OK。

WinCAT Project16 - TcXaeShell 文件(F) 編集(E) 彩黝(V) 项目(P) 生成(B) 得成(D) TwinC ○ - 〇 〇 〇 - 〇 - 〇 - 〇 - 〇 - 〇 - 〇 - 〇 -	スT TwinSAFE PLC EBLAM Scope ILR() 1800(M) 180(H) Meane - TwinSAFE PLC EBLAM Scope ILR() 1800(M) 180(H) Meane - TwinSAFE TO (180) - ● 1800 - ● 18	ア (快速日前 (Ctrl+Q) P - 6 X
解决方案资源管理器 → 3 ×	TwinCAT Project16 + ×	- 尾性 - 『×
〇 〇 〇 〇 〇 〇 〇 〇 - 〇 - 〇 - 〇 - 〇 - 〇	Onine	
Image Trained Project (1) + Statis) Image And Project (1) + Statis)	Insert Task X Name Task 2 Task 2 Task 2 Task 1 Task	
	整合解決方案 • 😵 错误 0 🛕 雪岩 0 🚺 湯豊 0 Clear 生成 + IntelliSense • 澄泉情况列表 🔎	
	े ¹ शर्म	
解决万英资源管理器 国际资源管理器	「「「ない」の「「」」「「」」」「「」」」「「」」」「「」」」「「」」」「「」」	現在 単共能

3. 点击 Input 或者 Output 添加新项 (这里以 Input 为例),选择与要绑定数据相同的类型 并点击 ok。



AWorksLP

Innost Projecto - Tooleanen	· •	
文件(F) 编辑(E) 视图(V) 项目(P) 生成(B) 谱试(D) TwinCA	AT TwinSAFE PLC 回队(M) Scope 工具(T) 窗口(W) 報助(H)	
80-0 18-10-10 日 🔐 🗶 🗗 白 り - C - Re	ease - TwinCAT RT (x64) - ▶ RthL	
Build 4024.54 (Loaded 12 🔯 🔯 🖉 🖉 🔞 🐂 🖉	TwinCAT Project 6 · storal> · · · · · · · · · · · · · · · · · · ·	
第決力算式課告理解 * * × ×	iwincal projectio # x	#tt + + ×
○ ○ ☆ ☆ · `o · d' ≯	Name [X] Online Type Size >Add In/Out Linked to	-
搜索解决方套资源管理器(Ctrl+;) ・		Bi: 94 /
解决方案"TwinCAT Project16"(1 个项目)		
 TwinCAT Project16 		
A SYSTEM		
A A Real-Time		
👔 I/O Idle Task		
🔺 🏙 Tasks		
 Task 2 		
i image		
📲 Outos 🖆 添加新项(W) Ins		
and Routes Recalc Addresses		
Type System		
TcCOM Objects		
DI DI C		
SAFETY		
🔛 C++		
I VISION		
ANALYTICS		
4 ^C Devices		
 Device 3 (EtherCAT) 		
Tmage		
Image-Info		
SyncUnits		
Dutouts		
InfoData		
Box 1 (32io)		
and Mappings		
	##決別來	
	登台解決方案 ・ 😮 错误 0 1 編巻 0 10 編巻 0 Clear 生成 + IntelliSense ・ 指気地決別法 🔎 -	
	⁷ 说明 ▲ 项目 文件 行	
解决方案资源管理器 团队资源管理器	(1) 「「「「」」」 「「」」 「」」 「」」 「」」 「」」 「」」 「」」 「	履性 工具相

4. 点击 linked to 选择要绑定的周期数据。

TwinCAT Project16 - TcXaeShell		7 8	快速启动 (Ctrl+Q) 🔑 🗕 🗗 🗙
文件(F) 編輯(E) 祝蜀(V) 项目(P) 生成(B) 请试(D) TwinCA	AT TwinSAFE PLC 团队(M) Scope 工具(T) 雷口(W) 帮助(H)		
10-0 話·加·加日 - X 日 カ ワ - C - Refe	lease - TwinCAT RT (x64) - ▶ Rtfl		
Build 4024.54 (Loaded 12 🖬 🖪 🖉 🖉 🛞 🍥 🐂 🧭	TwinCAT Projectio - <local> - III - III = EI 1 2 1 = O A B B D D -</local>		
			Mate
	reame [A] Unline type size >Add InvOut Linked to		10 D. L C.
現実解決万葉資源管理論(Ctrl+;) P・			all 2* /
副 解决方案"TwinCAT Project16"(1 个项目)			
A A SYSTEM			
License			
Real-Time			
i I/O Idle Task			
A Task 2			
1 Image	Insert Variable		
Inputs	General 2 DK		
Cutputs	Native: Voil Multiple: 1 0		
ste Koutes	Start Address: Byte: 0 Bit: 0 Cancer Cancer		
TcCOM Objects	Show All		
MOTION			
PLC PLC	Usa type >>pe Note Space		
SAFELY	8178 1		
VISION	800L 1		
ANALYTICS	E A/S000 P 0275 ActiveFeedbackAndMemory 1 A/S000		
4 🖾 I/O	EPicPersistentStatus 1 PLC		
 The Devices The Device 2 (Ether("AT)) 	SINT 1		
in botter o (control)			
Tmage-Info	C		
SyncUnits	Sauch Ture Dash draw Ture Craste Stimp Ture		
P inputs	Jaan Me. Enger and Merce and Merce		
InfoData			
Box 1 (32io)			
Mappings			
	告我 "读	≁ î ×	
	整个解决方案 • 🔂 错误 0 🛕 警告 0 0 Gear 生成 + IntelliSense • 投票错误列表	ρ.	
	³ 说明 ▲ 项目 文件	fi	
	1		
	1		
	1		
解決方面资源管理器 加以资源管理器	(a)水外表 船出		歴性 工具相

5. 数据绑定完毕后,我们可以根据自己的需求调整任务的发帧周期。



AWorksLP

		B/I -		- 1
U۶	ser	IVIA	nu	а

TwinCAT Project16 - TcXaeShell 文(柱)7 編載: 現面(2) 現面(2) 生成(8) 運成(8) 運成(9) TwinC ○ - 〇 (2) - ① - ② = ● ● ● ※ ※ ① ① ⑦ - ♡ - ○ - Re Build 4024.54 (Losded ● ※ ◎ ◎ ② ♥ ◎ ◎ ● ◎ ● ◎ ●	T TuninGAF FAC ERL(M) Scope IA(T) 第日(M) NB(H) ease TuninGAT R(664) ・ ▶ NBL-・	² (快速局別) (Ctrl+Q) P = の X
解決方案的運営理器 ・ ♀ × ○ ○ ☆ 部・ ◎ ・ 部	TwinCAT Projecti 6 • x	• 歴性 • P ×
Divide	Name: Vertels iso Vertels iso iso Type: IffE iso Type: IffE iso Addrew: Orion Iddex Addrew: Orion Iddex Comment: Iso Iso Iso <	
解決方實資源管理器 国际资格资源	· · · · · · · · · · · · · · · · · · ·	履住 工具箱

6. 回到从站界面,同步模式我们选择 DC。

TwinCAT Project16 - TcXaeShell	7	7 🔐 快速启动 (Ctrl+Q)	Р — 8 ×
	T TwinSAFE PLC 回队(M) Scope 工具(T) 酸口(W) 释助(H)		
〇 - 〇 智 - 🏠 🔛 🖕 💾 👗 🗇 🗇 フ - C - Rele	asse - TwinCAT RT (x64) - 🕨 Rttl 🔎 - 🗊 🏕 🖓 🕲 🖸 -		
Build 4024.54 (Loaded - 🕌 🔐 🛄 🔲 🖉 🌾 🛞 🍋 🔧	TwinCAT Project16 • «Local» •		
40.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1	Tue AT Project A & X	- 90	- 1 -
0.0 G H I To I H		Task 2	
	Task Online Parameter (Online) Add Symbols	20 24 2	
	Norman Tank 2 Back 201 *	E 215	
A TwinCAT Project16		(Name)	Task 2
A A SYSTEM	Auto start Object Id: 0x02010020	Disabled	Enabled
License	Auto Priority Management Options	ItemType	1
 Real-Time Up tidle Tests 	Priority: 1 Disable	PathName	TIKT^Task 2
A Stasks			
4 in Task 2	Create symbols		
image Image	Start tick (modulo): 0 🚊 Include external symbols		
 Inputs Was 1 	□ Separate input update		
Outputs	Pre ticks: 0 ‡		
glig Routes	Extern sync		
Type System	Warning by exceeding		
MOTION	Message box		
II PLC	Watching Outline 0		
SAFETY	Watchdog stack		
	Comment:		
4 🔯 I/O			
 ^{dig} Devices 			
 Device 3 (EtherCAT) 			
Image-Info			
SyncUnits			
Þ 🔄 Inputs			
Outputs			
 Box 1 (32io) 			
and Mappings	##25/#	8 × 1	
	- 10-10-10 安く航空大学 - 〇 勝美の ▲ 安美の ① 演算の Clay: 生成 & IntelliGence - 安安地学科等	<i>p</i> -	
		5700	
解决方案资源管理器 团队出源管理器	· · · · · · · · · · · · · · · · · · ·	歴性 工具箱	

5. 点击 Advanced Settings,此时会弹出 DC 的配置界面,首先需要配置 DC 的周期循环时间,配置后从站的 DC 就会按照这个循环时间周期性的触发中断,数据的处理和动作的同步。配置完毕后,勾选 Enable SYNC0 如果需要使用 SYNC1 同步信号,则一并勾选。



AWorksLP

2 TwinCAT Project16 - TcXaeShell 文中(F) 編集(E) 総要(V) 坂田(P) 生成(B) 編載(D) TwinC 日 - 0 3 - 10 - 10 日本 日 - 0 3 - 10 - 10 日本 Build 4024.54 (Loaded - 10 年春年年年年年年年年年年年年年年年年年年年年年年年年年年年年年年年年年年	AT TwinSAFE P(C EBL(M) Scope IE(T) 田口(M) 秘訣p4 ease - TwinSATE(D(A) -) ド田(快速用动 (Ctrl+Q) P - 8 ×
		[ms
第元方案は20世紀 ・ * * ×	WinCall Projection 4 X	Mate
○ ○ @ ⁸ - `o - Ø' ♪ <mark>-</mark>	General EtherCAT DC Process Data Startup CoE - Online Online	Box 1 (32io) 32io -
接索解决方案资源管理器(Ctrl+;)		[11] 및4 <i>)</i> 위
a 解决方案"TwinCAT Project16"(1 个项目)	Operation Mode: Synchron	Persistent
 TwinCAT Project16 	Synchron	Save in own File False
A 🧧 SYSTEM	DC	日 杂颂
License		(Name) Box 1 (32io)
A eal-Time		Disabled Enabled
📑 I/O Idle Task		ItemType 5
A the Tasks		Paulivame TIID*Device 3 (EtherCA
a lask z		
In inage		
 War 1 		
Outputs		
ply Routes		
Type System		
TcCOM Objects		
MOTION		
PLC .		
SAFETY		
64 C++		
VISION		
ANALYTICS		
2 20 1/0 1 1 10 10 10 10 10 10 10 10 10 10 10 10		
 Bevice 3 (EtherCAT) 	Name Online Tune Site addid In/Oct Uner Linked to	
1 Image	Vanie Online Type Size #Add In/Out Oset Diffeet to	
Image-Info	Wi X UDINI 4.0 39.0 Input 0 40-75 Var1.Input	
SyncUnits	VINZ UDINI 4.0 43.0 Input 0	
Inputs	VINS UDINI 4.0 47.0 Input 0	
Dutputs	VI IN4 UDINT 4.0 51.0 Input 0	
👂 🝓 InfoData	VINS UDINT 4.0 55.0 Input 0	
Box 1 (32io)	▼ IN6 UDINT 4.0 59.0 Input 0 Y	
and Mappings	tåg9jæ - • • ×	
	整合解決方案 · 図 物法 0 ▲ 岩田 0 0 道思 0 Clear 生成 + IntellSense · 複変換決別版 P ·	
	7 (4) (1) → (
	90H AT 13	
		Persistent
解决方案资源管理器 田认资政管理器		居住 工具相

6. 勾选完毕后点击激活配置,此时 TwinCAT 会重新启动,启动成功后我们可以看见右 下角图标变成绿色并且状态到达 op(配置模式下是蓝色)。

TwinCAT Project16 - TcXaeShell		快速启动 (Ctrl+Q) 🛛 🔎 🗕 🗗 🗙
	nCAT TwinSAFE PLC 图队(M) Scope 工具(T) 酸目(W) 報助(H)	
0-0 8-1-0 4 × 0 0 9-0-0	Release - TwinCATRT (r64) - ト 別10	
Build 4024.54 (Loaded - 2 1) 10 10 2 10 10 10 10 10 10 10 10 10 10 10 10 10		
解決方案資源管理論 • 平 >	VunčAl Projectib S X	<u>席住</u> マリン
0 0 A B • 0 • 0 / / -	General EtherCAT DC Process Data Startup CoE - Online Online	Box 1 (32io) 32io .
接索解决方案资源管理器(Ctrl+;)		
a] 解决方案"TwinCAT Project16"(1 个项目)	Operation Mode: DC ~	Persistent
 TwinCAT Project16 		Save in own hile False
A SYSTEM	Advanced Settings	(Name) Box 1 (32io)
A eal-Time	Advanced Settings X	Disabled Enabled
💼 I/O Idle Task		ItemType 5
4 🏙 Tasks	Distributed Lick Distributed Clock	PathName TIID^Device 3 (EtherCA)
A task 2	Cyclic Mode	
+ image	Operation Mode DC	
F Var 1		
Outputs	Zenable Sync Unit Cycle (µs): 1000	
illa Routes	SYNC 0	
Type System	Cycle Time (us):	
MOTION		
III PLC	Sync Unit Cycle x 1 V User Defined 0	
SAFETY SAFETY	User Defined + SYNC0 Cycle	
See C++		
▲ ₩ I/O	Based on Input Reference	
▲ ^a ^B _E Devices		
 Device 3 (EtherCAT) 	Name +	
inage	1 I I I I I I I I I I I I I I I I I I I	
image-into	1 N2	
Inputs	92 IN3	
Outputs	WINA SYNC 1	
InfoData	Sync Unit Cycle Cycle Time (µs): 1000	
 Box I (320) PDICharpel process data mapping 	SYNC 0 Cycle X1 Y Shift Time (usk 0	
PDOChannel process data mapping	± # ×	
WcState	整个編 Enable SYNC 1 現象情况引表 クー	
P InfoData	文件 行	
Mappings	Discussed and the second	
	· 利亚 和266	杂项
解决方案资源管理器 团队资源管理器	街 谈行後 線出	履性 工具箱
Provide Annual		

3.7 TwinCAT 设置 Box 热拔插

主站下添加从站项 (box),或者以扫描的从站 (box)

1. 右键 box,将 box 添加入热拔插组



AWorksLP

解决方案资源管理器 ◎ ◎ ☆ 🛱 ▾│ 🍳	• 🗇	- ₽ ×		
搜索解决方案资源管理器	(Ctrl	+;)		
 ↓ 解决方案"TwinCAT ▲ I winCAT Project ▲ I SYSTEM ▲ License ↓ I cense 	Proj ct3	ect3"(1 个项目)		
造 Tasks	*1	添加新项(W)	Ins	
🔀 Routes <u>द</u> Type Sys		Insert New Item Insert Existing Item		
	X	8除(V)	Del	
	X	重命名(M)		
SAFETY		Save Box 1 (32io) As		
<u>‰</u> C++	-51	与型(M)	C+rl+C	
VISION	V	复刑(T) 前扣(T)	Ctrl+C	
	ക		Ctrl+X	
	Ċ_	枯贴(P)	Ctrl+v	
∠ a bevices		Paste with Links		
‡∎ Im		Independent Project File		
📑 Im	•	Disable		
▷ 🗢 Sy		Change to Compatible Type		
▶ <mark>⊸</mark> In	ſ	Add to HotConnect group		
		Delete from HotConnect grou	lp	
⊳ <mark>vo</mark> Bo	x 1 (32io)		
📸 Mapping	S			

2. 设置将用于校验从站 ID 的地址,若总线中检查到插入相应设备 ID 的从站,则会与 其建立连接并到达 OP 状态



AWorksLP

	_	_		B /	-			_	
U	s	e	r.	IV	ы	n	u	а	
-	-	~	-				-	~	-

r Add Hot Connect Group	×
Select Connected Slaves	Identification by ✓ Identification Value: 0
	Optional Group Name



4. ESC 相关硬件配置

→ 注意 → ESC 相关配置以 DPort-ECT 为例

4.1 DPort-ECT

DPort-ECT 内部芯片同属于 ET1100 系, 驱动方式与 ET1100 类似。



图 4.1 DPort-ECT 连接及协议栈拓扑图

事实上大部分外接 ESC 拓扑都基本相同,除集成的 IP 核以外 (PDI,SYNC(0/1) 等信号由内核处理),协议栈所需码

- PDI: 用于与 ESC 通信交互的接口,例如 SPI,HBI 等;
- Sync Int:由 ESC 产生中断信号,MCU 处理的三个同步中断 (PDI,SYNC0/1 中断);
- Timer: 为协议栈提供同步检测和超时处理。

4.2 协议栈硬件适配接口

SSC 协议栈所需用户实现的 ESC 访问相关函数如下:

Read Access:

- fn HW_GetALEventRegister()
- fn HW_GetALEventRegister_Isr()
- fn HW_EscRead(pData, Address, Size)
- fn HW_EscReadIsr(pData, Address, Size)
- fn HW_EscReadWord(WordValue, Address)
- fn HW_EscReadWordIsr(WordValue, Address)
- fn HW_EscReadDWord(DWordValue, Address)
- fn HW_EscReadDWordIsr(DWordValue, Address)
- fn HW_EscReadMbxMem(pData, Address, Size)



Write Access:

- fn HW_EscWrite(pData, Address, Size)
- fn HW_EscWriteIsr(pData, Address, Size)
- fn HW_EscWriteWord(WordValue, Address)
- fn HW_EscWriteWordIsr(WordValue, Address)
- fn HW_EscWriteDWord(DWordValue, Address)
- fn HW_EscWriteDWordIsr(DWordValue, Address)
- fn HW_EscWriteMbxMem(pData, Address, Size)

Key Words:

- HW_EscReadXXX: 从 ESC 中读取数据;
- HW_EscWriteXXX: 向 ESC 中写入数据;
- HW_EscXXXIsr: 其中后缀带有 Isr 的函数表示此函数将会在中断管中调用, 且函数的执行不能够被打断;

User Manual

- HW_EscXXXWordXX: 其中带有 Word 的函数,代表所操作的字节数为 2;
- HW_EscXXXDWordXX: 其中带有 DWord 的函数, 代表所操作的字节数为 4;
- Address: 代表要操作的 ESC 寄存器或 DPRAM 地址;
- Size: 代表要操作的字节数;
- pData: 在读访问中代表用于存储从 ESC 中取出的数据的 buffer。在写访问中代表 指向将要写入 ESC 的数据 buffer;
- WordValue/DWordValue: 代表要进行读写的变量。

注: HW_EscXXXWordXXX 或 HW_EscXXXDWordXXX 比较特殊, 其参数 WordValue/DWordValue 代表的是一 个变量而不是一个指向 buffer 的指针。在 DPort-ECT 中通常会将其实现为 (附带 HW_GetALEventRegister() 的 实现):

```
#define HW_EscReadWord(WordValue, Address) \
    HW_EscRead(((MEM_ADDR *)&(WordValue)),((UINT16)(Address)),2)
#define HW_EscReadDWord(DWordValue, Address) \
    HW_EscRead(((MEM_ADDR *)&(DWordValue)),((UINT16)(Address)),4)
#define HW_EscReadWordIsr(WordValue, Address) \
    HW_EscReadIsr(((MEM_ADDR *)&(WordValue)),((UINT16)(Address)),2)
#define HW_EscReadDWordIsr(DWordValue, Address) \
    HW_EscReadIsr(((MEM_ADDR *)&(DWordValue)),((UINT16)(Address)),4)
#define HW_EscReadIsr(((MEM_ADDR *)&(DWordValue)),((UINT16)(Address)),4)
#define HW_EscWriteWord(WordValue, Address) \
    HW_EscWrite(((MEM_ADDR *)&(WordValue)),((UINT16)(Address)),2)
#define HW_EscWriteDWord(DWordValue, Address) \
    HW_EscWrite(((MEM_ADDR *)&(WordValue)),((UINT16)(Address)),4)
```



AWorksLP

```
User Manual
```

```
#define HW EscWriteWordIsr(WordValue, Address) \
       HW_EscWriteIsr(((MEM_ADDR *)&(WordValue)),((UINT16)(Address)),2)
#define HW_EscWriteDWordIsr(DWordValue, Address) \
       HW EscWriteIsr(((MEM ADDR *)&(DWordValue)),((UINT16)(Address)),4)
/* 0x220U 为ESC内部的事件寄存器地址 */
void HW GetALEventRegister(void)
{
   UINT16 ALEventReg;
   HW_EscRead(&ALEventReg, 0x220U, 2);
   return ALEventReg;
}
void HW_GetALEventRegister_Isr(void)
{
   UINT16 ALEventReg;
   HW_EscReadIsr(&ALEventReg, 0x220U, 2);
   return ALEventReg;
}
```

若是使能 (AL_EVENT_ENABLED == 1) 后,周期数据需要由中断处理,在中断中运行 void PDI_Isr(void) 函数。

若是使能 (DC_SUPPORTED == 1) 后,主站开启 DC 模式,从站则需要在中断中运行 void Sync0_Isr(void)/void Sync1_Isr(void) 函数。

SSC 协议栈所需用户实现的定时器相关函数如下:

未使用定时器中断时 (ECAT_TIMER_INT == 0) 或者使能 ESC_EEPROM_ACCESS_SUPPOTOM(EEPROM 的访问函数需要超时处理),需要实现 以下函数和宏:

- marco ECAT_TIMER_INC_P_MS: 定时器计时 1ms 会有多少个 tick;
- fn UINT16 HW_GetTimer(void): 获取定时器当前 tick 值;
- fn void HW_ClearTimer(void): 清除定时器 tick 值。

若 (ECAT_TIMER_INT == 1) 时,需要提供一个 1ms 的定时器中断,周期性的运行协议 栈提供的 void ECAT_CheckTimer(void) 函数。

4.3 EEPROM

EEPROM 的大小实际取决于 ESC 上电时的 bootstrap 配置:

表 4.1	Register EEPROM	Control/Status	(0x502[7])
-------	-----------------	----------------	------------

Bit	Description
7	Selected EEPROM Algorithm:
	0: 1 address byte (1Kbit - 16Kbit EEPROMs)
	1: 2 address bytes (32Kbit - 4 Mbit EEPROMs)



AWorksLP

通常 ESC 会提供外部引脚来于用户锁存 EEPROM 的配置信息,寻址模式 (1 address byte or 2 address byte) 需要结合实际的 EEPROM 的时序选择。



DPort-ECT 的 EEPROM 硬件连接的是 16Kbit, 也就是说 DPort-ECT 的大小为 16Kbit(0x800 Byte)。

4.4 ESC Configuration Data(Handware Config)

Configuration data 定义于 ESI 文件中, 烧写 ESI 文件时, 会将其烧写在 EEPROM 中前八 个 Word(16Byte)

程序清单 4.1	ESI 文件中的	ConfigData	字段
----------	----------	------------	----



Word



Configuration data 在 ESC 上电或者复位时加载进入对应的 ESC 寄存器中



AWorksLP

Word Address	Parameter	Register Address
0x0	PDI Control/ESC Configuration	0x0140:0x0141
0x1	PDI Configuration	0x0150:0x0151
0x2	Pulse Length of SYNC Signals	0x0982:0x0983
0x3	Extended PDI Configuration	0x0152:0x0153
0x4	Configured Station Alias	0x0012:0x0013
0x5	Reserved	
0x6	Reserved	
0x7	Checksum	

4.4.1 PDI Control register(SPI interface select)

topology:

Bit	Description
7:0	Process data interface:
	0x00:interface deactivated(no PDI)
	0x01:4 Digital Input
	0x02:4 Digital Output
	0x03:2 Digital Input and 2 Digital Output
	0x04:Digital I/O
	0x05:SPI Slave
	0x06:Oversampling I/O
	0x07:EtherCAT Bridge(port3)
	0x08:16 Bit asynchronous Microcontroller interface
	0x09:8 Bit asynchronous Microcontroller interface
	0x0A:16 Bit synchronous Microcontroller interface
	0x0B:8 Bit synchronous Microcontroller interface
	0x10:32 Digital Input and 0 Digital Output
	0x11:24 Digital Input and 8 Digital Output
	0x12:16 Digital Input and 16 Digital Output
	0x13:8 Digital Input and 24 Digital Output
	0x14:0 Digital Input and 32 Digital Output
	0x80:On-chip bus
	Others:reserved

Usage:

用于选择 ESC 的通信接口 (PDI), 例如 DPort-ECT 的 PDI 只支持 SPI Slave。



⚠️ 此寄存器的内容并非一成不变,各家厂商会根据设计的 EtherCAT IP 外围电路添加些 许内容, 典型的就是 MicroChip LAN925x 系列。



4.4.2 ESC Configuration register(SYNC0/1 Int Out enabled)

topology:

表 4.3 E	SC Configuration	register
---------	------------------	----------

Bit	Description	
0	Device emulation (control of AL status):	
	0:AL status register has to be set by PDI	
	1:AL status register will be set to value Written to AL control register	
1	Enhanced Link detection all port:	
	0:disabled (if bits[7:4]=0)	
	1:enabled at all port(overrides bits[7:4])	
2	Distributed Clocks SYNC Out Unit:	
	0:disabled(power saving)	
	1:enabled	
3	Distributed Clocks Latch In Unit:	
	0:disabled(power saving)	
	1:enabled	
4	Enhanced Link port0:	
	0:disabled(if bit=0)	
	1:enabled	
5	Enhanced Link port1:	
	0:disabled(if bit=0)	
	1:enabled	
6	Enhanced Link port2:	
	0:disabled(if bit=0)	
	1:enabled	
7	Enhanced Link port3:	
	0:disabled(if bit=0)	
	1:enabled	

Usage:

• bit[0]:

set 0: 主站和从站状态转换规律如下:

(1). 主机控制从机状态转换,将目的状态写入从站 AL 控制位 (0x0120[0:3]);

(2). 从机读取新状态请求后, 检查自身状态:

a: 如果可以转换,则将新状态写入状态机一实际状态位 (0x130[0:3]);

b:如果不能转换,则不改变实际状态位,设置错误指示位 (0x130[4]),并 将错误码写入 (0x134~0x135);

(3). 主站读取状态机实际状态 (0x130):

a:如果正常转换,则执行下一步操作;

b:如果转换出错,主站读取错误码,并写 AL 错误应答 (0x120[4]) 来清除 AL 错误指示;

set 1: AL status register 的值由控制 AL 控制寄存器的值决定



• bit[1]: 使能增强型来链路检测, 这里贴上官方概念解释

Enhanced Link Detection

For Ethernet, the enhanced MII link detection feature is a feature of link error detection and reaction. This has to be distinguished from the actual link detection, which tells the ESC if a physical link is available (i.e., the LINK_MII signal or the MI link detection and configuration mechanism). Enhanced MII link detection will additionally disconnect a link if at least 32 RX errors (RX_ER) occur in a fixed interval of time (~10 μ s). The local loop is closed and the link partner is informed by restarting the Auto-Negotiation mechanism via the MII Management Interface. This informs the link partner of the error condition, and the link partner will close the loop. The ESC keeps the port closed until the link goes down during Auto-Negotiation and comes up again (the port remains closed if the link does not go down). The availability of Enhanced MII Link Detection depends on a supported PHY address configuration, otherwise it has to be disabled.

- bit[2]: 分布式时钟同步输出单元使能
- bit[3]: 分布式时钟锁存单元使能
- bit[4:7]: 端口增强型链路使能

4.4.3 PDI Configuration register(SPI mode and PDI Int Config)

topology:

Description
SPI mode:
00:SPI mode0
01:SPI mode1
10:SPI mode2
11:SPI mode3
NOTE:SPI mode3 is recommended for Slave Sample Code
NOTE:SPI status flag is not available in
SPI modes0 and 2 with normal data out sample
SPI_IRQ output driver.polarity:
00: Push-Pull active low
01: Open Drain (active low)
10: Push-Pull active high
11: Open Source (active high)
SPI_SEL polarity:
0: Active low
1: Active high
Data Out sample mode:
0: Normal sample (SPI_DO and SPI_DI
are sampled at the same SPI_CLK
edge)
1: Late sample (SPI_DO and SPI_DI are

表 4.4 PDI Configuration register

sampled at different SPI_CLK edges)



AWorksLP

User Manual

续上表

Bit	Description	
7:6	Reserved, set EEPROM value to 0	

Usage:

- bit[1:0]: SPI 通信模式, 示例代码中为 mode3
- bit[3:2]: PDI 中断信号的有效极性
- bit[4]: SPI CS 引脚有效极性
- bit[5]: 数据输出采样模式

4.4.4 Pulse Length of SYNC Signals(SYNC0/1 Signals length)

topology:

表 4.5 PDI Configuration register

Bit	Description
15:0	ECAT PDI Reset Value Pulse length of SyncSignals (in Units of 10ns)
	0: Acknowledge mode: SyncSignal will be cleared by reading SYNC[1:0] Status register

Usage: SYNC0/1 中断信号有效电平长度

4.5 ESC SPI Slave

ConfigData(050C034410270000000) 对应着以下的硬件配置:

SPI:

- mode3(空闲时是高电平,从第二个跳变沿(下降沿)开始采样);
- msb(最高位有效);
- 8bit;
- 软件控制 CS 硬件;
- CS 低电平有效。

PDI/SYNC0/SYNC1 中断:

• 触发方式:下降沿触发。

CMD[2]	CMD[1]	CMD[0]	Command
0	0	0	NOP(no operation)
0	0	1	reserved
0	1	0	Read
0	1	1	Read with following Wait State bytes
1	0	0	Write
1	0	1	reserved
1	1	0	Address Externsion(3 address/command bytes)
1	1	1	reserved



spi 接口支持两种寻址模式,2byte addressing 和 3byte addressing 模式

注:

- 2byte mode(寻址: 0~8k): 13bit address + 3bit command
- 3byte mode(寻址: 0~64k): 16bit address + 6bit command + 2bit reserved

4.5.1 ESC SPI Read Access

表 4.7 Address modes for Read access with Wait state byte

Byte	2 Byte address mode	3 Byte address mode
0	A[12:5] address bit [12:5]	A[12:5] address bit [12:5]
1	A[4:0] address bit [4:0]	A[4:0] address bit [4:0]
	CMD0[2:0] read command: 011b	CMD0[2:0] read command: 011b
2	0xFF wait state byte	A[4:0] address bit [4:0]
		CMD0[2:0] read command: 011b
		res[1:0] two reserved bits, set to 00b
3	D0[7:0] data byte 0	0xFF wait state byte
4	D1[7:0] data byte 1	D0[7:0] data byte 0
5 ff.	D2[7:0] data byte 2	D1[7:0] data byte 1

• Read Access

在数据阶段,从站会发送读取的数据到主站

• Read Wait State

在最后一个地址/命令字节和第一个数据字节之前,SPI 主设备必须等待从站获取内部的数据,后续字节读取自动预取,不需再等待

1. 在最后一个地址/命令字节和第一个数据字节之间,SPI 主设备需要等待最 坏的内部读取时间

2. 在最后一个地址/命令字节和第一个数据字节之间,SPI 主设备可以插入一个 0xFF 等待字节

• Read Termination

在最后一个数据字节,SPI 主设备需要发送一个 0xFF 代表数据读取终止,从站 识别终止字节后,就不会再预取内部数据。如果 SPI 主设备发送 0x00,则代表 从设备至少要预取一个字节的数据

4.5.2 ESC SPI Write Access

表 4.8 Write access for 2 and 4 Byte SPI Masters

Byte	2 Byte SPI Master	4 Byte SPI Master
0	A[12:5] address bit [12:5]	A[12:5] address bit [12:5]
1	A[4:0] address bit [4:0]	A[4:0] address bit [4:0]
	CMD0[2:0] write command: 100b	CMD0[2:0] 3 byte addressing:110b
2	D0[7:0] data byte 0	A[15:13] address bit [15:13]
		CMD1[2:0] 3 byte addressing:110b
		res[1:0] two reserved bits, set to 00b



+	
1.	<u>+</u> =
- <u>-</u>	1 70

		——————————————————————————————————————
Byte	2 Byte SPI Master	4 Byte SPI Master
3	D1[7:0] data byte 1	A[15:13] address bit [15:13]
		CMD1[2:0] 3 byte addressing:100b
		res[1:0] two reserved bits, set to 00b
4	D2[7:0] data byte 2	D0[7:0] data byte 0
5	D3[7:0] data byte 3	D1[7:0] data byte 1
6	D4[7:0] data byte 4	D2[7:0] data byte 2
7	D5[7:0] data byte 5	D3[7:0] data byte 3

• Write Access

在写访问的数据阶段,SPI 主设备向从站发送写入的数据,在发送最后一个数据后,需要将 SPI 片选释放,至此 SPI 写访问终止

4.5.3 ESC Read/Write 通信时序



图 4.2 SPI read access (2 byte addressing,1 byte read data) with Wait State byte





图 4.3 SPI read access (2 byte addressing,2 byte read data) with Wait State byte



图 4.4 SPI write access (2byte addressing,1byte write data)







4.5.4 ESC 访问函数伪代码实现

伪代码实现 ESC 读写访问函数

```
/**
 * \brief send command and address
 */
static void __handle_setup_phase(uint16_t address, uint8_t command)
#define __ADDR_MODE_BYTE_MAX (4U)
    uint8_t tx_data[__ADDR_MODE_BYTE_MAX] = {0, 0, 0, 0}, tx_len = 0;
    tx_data[0] = address >> 5;
    tx_len++;
    /**
    * \brief check address is 2byte or 3byte mode
    */
    if((address + 8) & ~0x1fff)
    {
        tx_data[1] = (address << 3) + 0x06;</pre>
        tx_data[2] = ((address >> 8) & 0x0e) + (command << 2);</pre>
        tx_len += 2;
    }
    else
    {
        tx_data[1] = (address << 3) | command;</pre>
        tx_len++;
    }
    /**
    * \brief add wait state byte
```



User Manual

EtherCAT 从站参考手册

AWorksLP

```
*/
    if(__ESC_RWS == command)
    {
        tx_data[tx_len] = 0xff;
        tx_len++;
   }
    __g_esc_obj->pfn_spi_write(tx_data, tx_len);
}
void aw_ecat_esc_read(void *p_data, uint16_t address, uint16_t len)
{
    uint8_t tx_data = 0xFF;
   while(len-- > 0)
    {
        __g_esc_obj->pfn_spi_entry_critical();
        __g_esc_obj->pfn_spi_cs_select();
        /**
        * \brief send address and cmd
        */
        __handle_setup_phase(address,
#if /*if ESC is same as ET1100*/
                            ___ESC_RWS
#else /*if ESC is same as ET1200*/
                            ___ESC_RD
#endif
                            );
        /**
        * \brief read data one by one,
        * it can be interrupted by PDI_Isr/Syncx_Isr
        * for realtime
        */
        __g_esc_obj->pfn_spi_write_and_read(&tx_data, p_data, 1);
        __g_esc_obj->pfn_spi_cs_deselect();
        __g_esc_obj->pfn_spi_exit_critical();
        (uint8_t*)p_data++;
        address++;
    }
}
void aw_ecat_esc_read_isr(void *p_data, uint16_t address, uint16_t len)
{
    uint8_t tx_data = 0x00U;
    __g_esc_obj->pfn_spi_entry_critical();
    __g_esc_obj->pfn_spi_cs_select();
    /**
    * \brief send address and cmd
   */
    __handle_setup_phase(address,
#if /*if ESC is same as ET1100*/
```



AWorksLP

```
ESC RWS
#else /*if ESC is same as ET1200*/
                            ___ESC_RD
#endif
                            );
    /**
    * \brief read data
   */
    while(len-- > 1)
    {
        __g_esc_obj->pfn_spi_write_and_read(&tx_data, p_data, 1);
        (uint8_t*)p_data++;
    }
    /**
    * \brief send terminate data
   */
   tx_data = 0xFFU;
    __g_esc_obj->pfn_spi_write_and_read(&tx_data, (uint8_t*)p_data, 1);
    __g_esc_obj->pfn_spi_cs_deselect();
    __g_esc_obj->pfn_spi_exit_critical();
}
void aw_ecat_esc_write(void *p_data, uint16_t address, uint16_t len)
ł
    while(len-- > 0)
   {
        __g_esc_obj->pfn_spi_entry_critical();
        __g_esc_obj->pfn_spi_cs_select();
        /**
        * \brief send address and cmd
        */
        __handle_setup_phase(address, __ESC_WR);
        __g_esc_obj->pfn_spi_write(p_data, 1);
        __g_esc_obj->pfn_spi_cs_deselect();
        __g_esc_obj->pfn_spi_exit_critical();
        (uint8_t*)p_data++;
        address++;
   }
}
void aw_ecat_esc_write_isr(void *p_data, uint16_t address, uint16_t len)
{
    __g_esc_obj->pfn_spi_entry_critical();
    __g_esc_obj->pfn_spi_cs_select();
    /**
    * \brief send address and cmd
   */
    __handle_setup_phase(address, __ESC_WR);
    __g_esc_obj->pfn_spi_write(p_data, len);
```



AWorksLP

}

```
__g_esc_obj->pfn_spi_cs_deselect();
__g_esc_obj->pfn_spi_exit_critical();
```

User Manual



诚信共赢,持续学习,客户为先,专业专注,只做第一

广州致远电子股份有限公司

更多详情请访问 www.zlg.cn 欢迎拨打全国服务热线 400-888-4005

