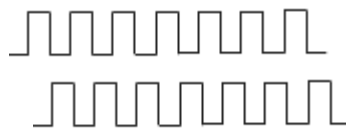


可升级的控制器 灵活的自动化控制系统

PWM 脉冲输出功能使用说明



声明

(1) 在实际的系统构建时，请先确认系统组成设备、装置，如使用过程中对额定值、性能留有余量，以及万一发生故障时将危险降到最低的安全电路等。

(2) 为了安全使用系统，请获取组成系统的各个设备、装置的指南及安装说明书，在确认好包括“安全注意事项”、“安全要点”等内容后再使用。

(3) 本文件所属的产品/系统只允许由具备相关工作经验要求的合格人员进行操作，确认适合系统的规格、法规及规定。

(4) 未经ABB公司许可，严禁擅自对本资料的一部分或全部内容进行篡改及散发。

(5) 本资料的记录内容为测试指导，在应用过程，请根据现场实际情况适当调整，或者咨询ABB技术支持人员。

(6) 我们已对文档中所述内容与硬件和软件的一致性作过检查。然而不排除存在偏差的可能性，因此我们不保证文档中所述内容与硬件和软件完全一致。文档中的数据都按规定经过检测，必要的修正值包含在下一版本中。

(7) 本资料的记录内容若有改版，恕不另行通知。

目录

1	简介	4
1.1	参考文档	4
1.2	硬件和软件需求	4
1.3	脉宽调制模式 PWM	4
2	CD522 模块的 PWM 功能	5
2.1	概述	5
2.2	脉冲输出连接输出负载的接线	5
2.3	频率输出应用	7
2.3.1	设置过程	7
2.3.2	编程开发	12
2.3.3	PWM_OUT 功能块说明	12
2.4	脉冲输出应用	17
2.4.1	设置过程	17
2.4.2	编程开发	18
2.4.3	PULSE_OUT 功能块说明	18
3	eCO 集成 I/O 的 PWM 输出	24
3.1	概述	24
3.2	脉冲输出连接输出负载的接线	24
3.3	频率输出应用	25
3.3.1	设置过程	25
3.3.2	编程开发	25
3.3.3	ONB_IO_PWM_FREQ 功能块说明	26
3.3.4	ONB_IO_PWM_TIME 功能块说明	29
4	DC541 模块的 PWM 功能	33
4.1	概述	33
4.2	脉冲输出连接输出负载的接线接线	33
4.3	PWM 输出应用	34

4.3.1 设置过程..... 34

4.3.1 编程开发..... 36

4.3.2 DC541_PWM 功能块说明 38

5 程序下载.....42

6 程序监控.....46

1 简介

本文档由浅入深地指导实现 AC500 系列 CPU 实现 PWM 功能用于控制步进电机或伺服控制器。该例程为用于培训目的的简单样例，不可直接用于实际工程设计。



NOTICE

用户需具备 AC500 系列 PLC 系统、Control Builder Plus、CoDeSys 编程、ASCII 通讯方面的基础知识。



NOTICE

如需了解关于 Control Builder Plus、CoDeSys 编程及 AC500 系列 PLC 应用方面的更多知识，请查看软件在线帮助。

1.1 参考文档

	Reference	Description
1	CoDeSys 帮助手册	

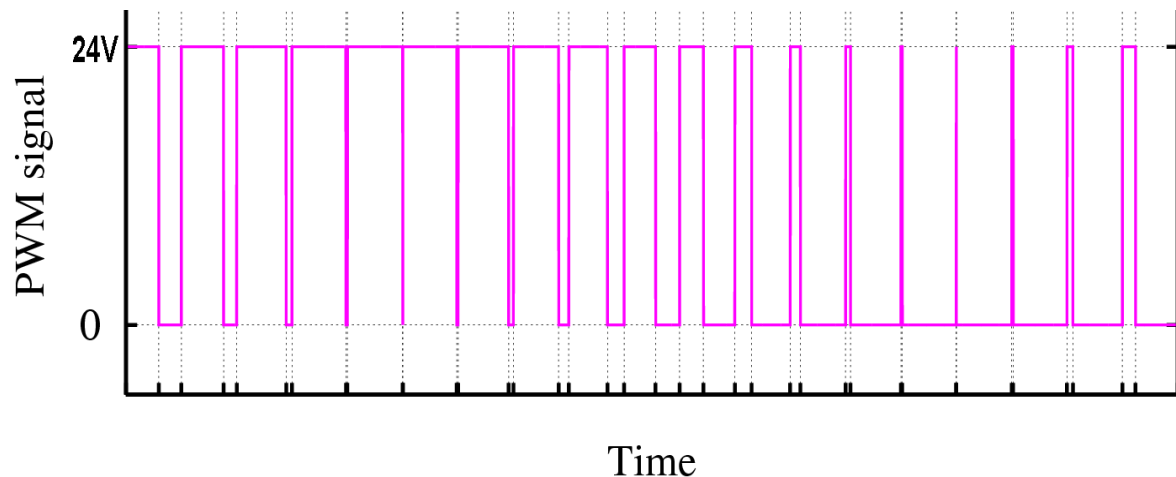
1.2 硬件和软件需求

Hardware	Software
PM573-ETH V2.2	PS501: V2.2
PM554-T V2.2	
CD522	
DC541	

1.3 脉宽调制模式 PWM

概述

- ◇ 脉宽调制产生方波信号，这些被调制的方波可改变波形的平均值。
- ◇ 信号平均值直接依赖于占空比。
- ◇ 典型 PWM 输出信号：



2 CD522 模块的 PWM 功能

2.1 概述

2 个脉宽模块或脉冲输出

输出特点

◇ 推挽式输出：24 V DC, 最大 100 mA

◇ 限制电流（过热或过流）

PWM 规格

◇ 频率从 1 Hz 到 100 kHz

◇ 值从 0 到 100 %

◇ 脉冲特点

◇ -频率从 1 Hz 到 15 kHz

◇ 输出脉冲从 1 到 65535 个脉冲

◇ 脉冲输出数量指示(0 到 100 %)

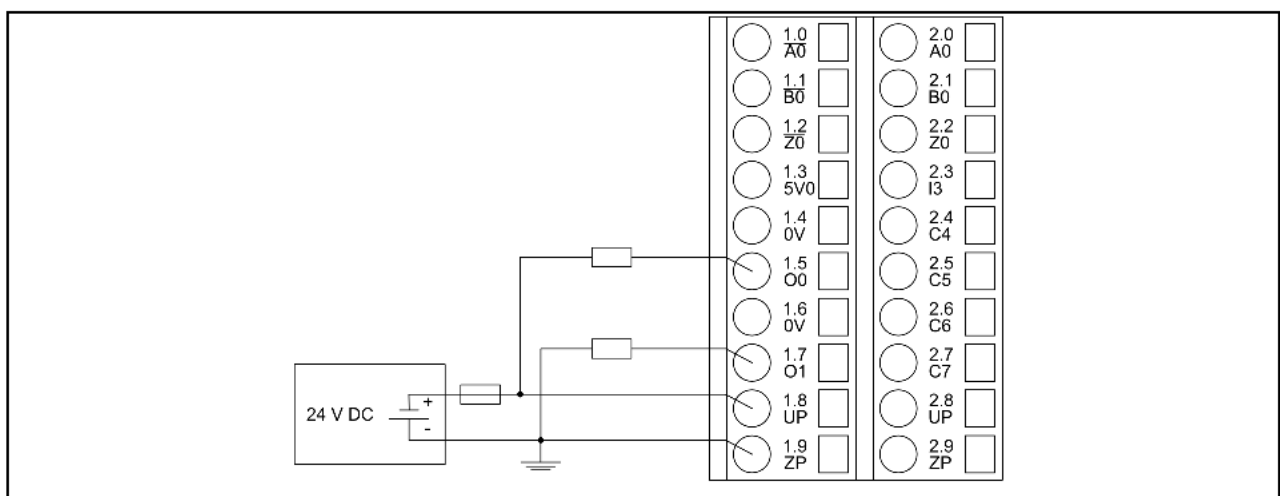
频率特点

◇ 当占空比设为 50%时，频率输出=100 kHz

◇

2.2 脉冲输出连接输出负载的接线

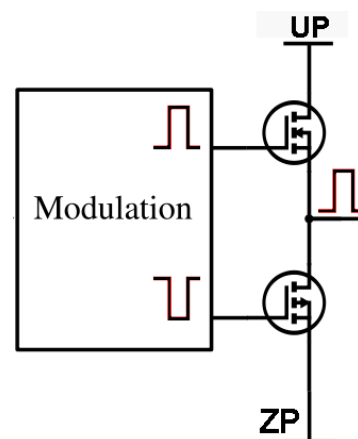
下图显示 CD522 模块的 PWM/脉冲输出与输出负载的电气连接



图：CD522 模块的 PWM/脉冲输出与输出负载的连接

- 推挽式输出原理
 - 推挽式输出使用两个激励晶体管
 - 第一个 (高位) 连接 UP,
 - 第二个 (地位) 连接 ZP
 - 晶体管由相反的信号驱动(180° 相位差)
 - 当输出驱动低信号不能“打开”并作为

标准的输出时，它被引向 ZP 。



2.3 频率输出应用

利用 CD522 计数器与 PWM 功能，结合安川 SGM 系列伺服，实现位置控制功能。

2.3.1 设置过程

➤ 步骤 1：按如下方式设置安川伺服控制器输入方式。

伺服单元侧的输入形态请根据指令控制器的规格设定用户参数 Pn200. 0。

用户参数	指令形态	输入倍增	正转指令	反转指令
Pn200	n. □□□0 符号 + 脉冲列 (正逻辑) (出厂时的设定)	—	PULS (CN1-7) SIGN (CN1-11) H电平	PULS (CN1-7) SIGN (CN1-11) L电平
	n. □□□1 CW+CCW (正逻辑)	—	PULS (CN1-7) SIGN (CN1-11) L电平	PULS (CN1-7) SIGN (CN1-11) L电平
	n. □□□2 90° 相位差	×1	PULS (CN1-7) SIGN (CN1-11) 90°	PULS (CN1-7) SIGN (CN1-11) 90°
	n. □□□3 2 相脉冲	×2		
	n. □□□4 (正逻辑)	×4		
	n. □□□5 符号 + 脉冲列 (负逻辑)	—	PULS (CN1-7) SIGN (CN1-11) L电平	PULS (CN1-7) SIGN (CN1-11) H电平
	n. □□□6 CW+CCW (负逻辑)	—	PULS (CN1-7) SIGN (CN1-11) H电平	PULS (CN1-7) SIGN (CN1-11) H电平
	n. □□□7 90° 相位差	×1	PULS (CN1-7) SIGN (CN1-11) 90°	PULS (CN1-7) SIGN (CN1-11) 90°
	n. □□□8 2 相脉冲	×2		
n. □□□9 (负逻辑)	×4			

■补充

90° 相位差 2 相脉冲指令形态时，可设定输入倍增。

→ 正转

← 反转

PULS (CN1-7)
SIGN (CN1-11)

×1倍

×2倍

×4倍

内部处理

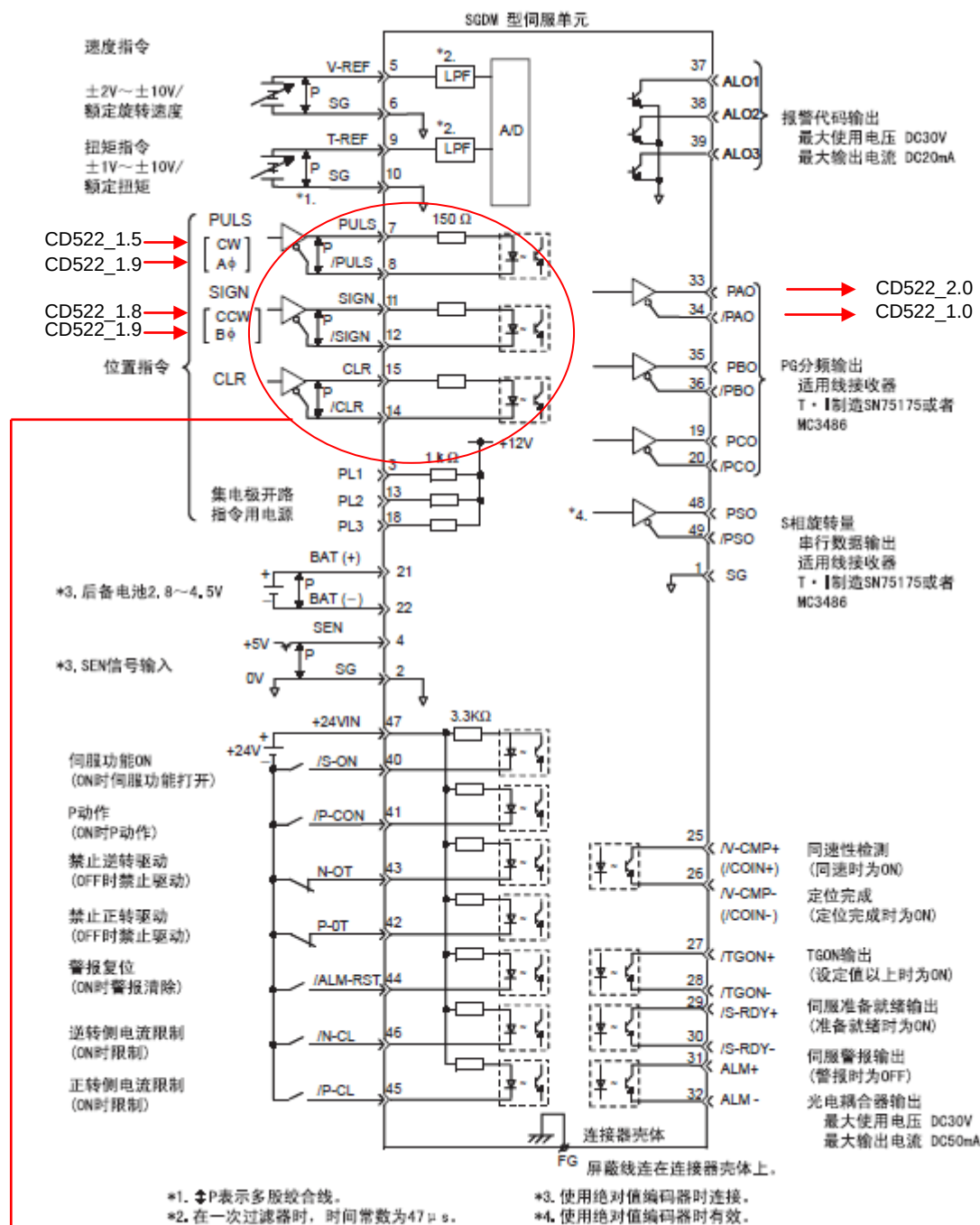
伺服电机的移动指令脉冲



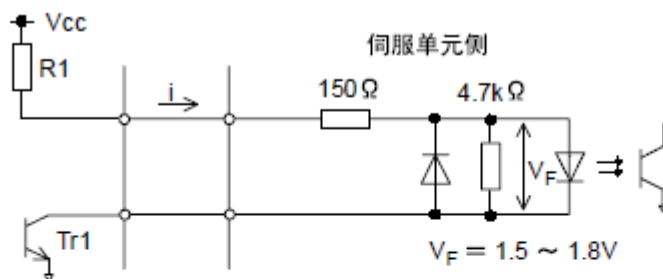
注释

对于SGM伺服的其它参数设置，请查阅相关说明书，本文不另做说明。

➤ 步骤 2：按安川伺服接线说明，将编码器与控制信号接入 CD522 模块。本例中采用相对编码器 A 相作为计数信号，B 相和 Z 相未使用。



上级装置侧



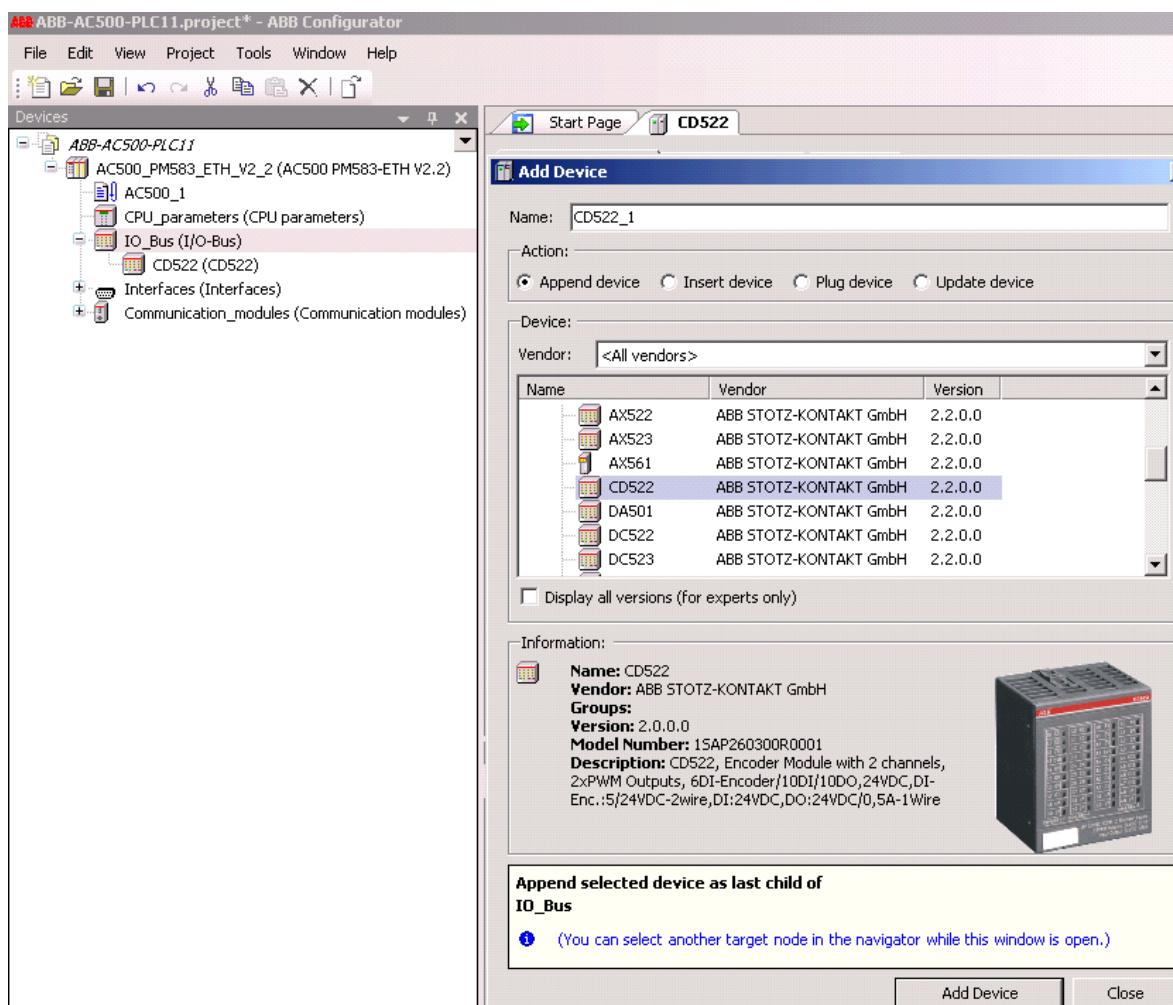
请参考以下的适用实例, 对工作电阻 R1 的值进行设定, 使其满足输入电流 i 在 7mA ~ 15mA 范围之间的条件。

适用例		
Vcc 为 24V ± 5% 时 R1=2.2kΩ	Vcc 为 12V ± 5% 时 R1=1kΩ	Vcc 为 5V ± 5% 时 R1=180Ω

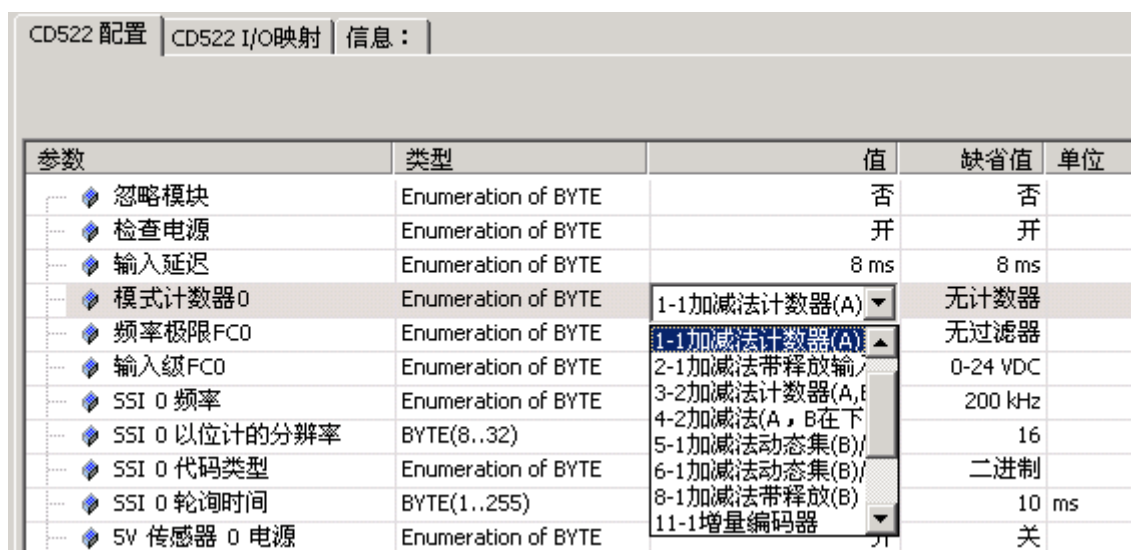


SGM伺服的编码器电压等级为5V。

- 步骤 3: 在 ABB 配置器中，以下位置右击，并在下拉框中选择并添加 CD522 模块。



- 步骤 4: 选择计数方式（模式 1）；



- 步骤 5：选择编码器电信号类型(0~5V)：

CD522 配置 CD522 I/O映射 信息：			
参数	类型	值	缺省值
忽略模块	Enumeration of BYTE	否	否
检查电源	Enumeration of BYTE	开	开
输入延迟	Enumeration of BYTE	8 ms	8 ms
模式计数器0	Enumeration of BYTE	1-1加减法计数器(A)	无计数器
频率极限FC0	Enumeration of BYTE	无过滤器	无过滤器
输入级FC0	Enumeration of BYTE	0-5 VDC	0-24 VDC
SSI 0 频率	Enumeration of BYTE	0-24 VDC	200 kHz
SSI 0 以位计的分辨率	BYTE(8..32)	0-5 VDC	16
SSI 0 代码类型	Enumeration of BYTE	差动	二进制
SSI 0 轮询时间	BYTE(1..255)	正弦 1 Vpp	10

- 步骤 6：激活 CD522 模块+5VDC 供电；

CD522 配置 CD522 I/O映射 信息：			
参数	类型	值	缺省值
忽略模块	Enumeration of BYTE	否	否
检查电源	Enumeration of BYTE	开	开
输入延迟	Enumeration of BYTE	8 ms	8 ms
模式计数器0	Enumeration of BYTE	1-1加减法计数器(A)	无计数器
频率极限FC0	Enumeration of BYTE	无过滤器	无过滤器
输入级FC0	Enumeration of BYTE	0-5 VDC	0-24 VDC
SSI 0 频率	Enumeration of BYTE	200 kHz	200 kHz
SSI 0 以位计的分辨率	BYTE(8..32)	16	16
SSI 0 代码类型	Enumeration of BYTE	二进制	二进制
SSI 0 轮询时间	BYTE(1..255)	10	10
5V 传感器 0 电源	Enumeration of BYTE	开	关
模式计数器1	Enumeration of BYTE	关	无计数器
频率极限FC1	Enumeration of BYTE	开	无过滤器

- 步骤 7: CD522 配置中, 点击 PWM/Pulse 子目录用以选择通道来配置 PWM 模式。

CD522 配置 CD522 I/O映射 信息:			
参数	类型	值	缺省值
SSI 0 频率	Enumeration of BYTE	200 kHz	200 kHz
SSI 0 以位计的分辨率	BYTE(8..32)	16	16
SSI 0 代码类型	Enumeration of BYTE	二进制	二进制
SSI 0 轮询时间	BYTE(1..255)	10	10
5V 传感器 0 电源	Enumeration of BYTE	开	关
模式计数器 1	Enumeration of BYTE	无计数器	无计数器
频率极限FC1	Enumeration of BYTE	无过滤器	无过滤器
输入级FC1	Enumeration of BYTE	0-24 VDC	0-24 VDC
SSI 1 频率	Enumeration of BYTE	200 kHz	200 kHz
SSI 1 以位计的分辨率	BYTE(8..32)	16	16
SSI 1 代码类型	Enumeration of BYTE	二进制	二进制
SSI 1 轮询时间	BYTE(1..255)	10	10
5V 传感器 1 电源	Enumeration of BYTE	关	关
短路探测和传感器	Enumeration of BYTE	关	关
通信错误时输出行为	Enumeration of BYTE	关	关
替换值	WORD(0..65535)	0	0
PWM / 脉冲			
输出O0, 通道配置	Enumeration of BYTE	PWM	数字输出
输出O1, 通道配置	Enumeration of BYTE	数字输出	数字输出
计数器0			
输入I3, 通道配置	Enumeration of BYTE	PWM	数字输入

- 步骤 8: 定义模块的输入输出标识。CD522 库文件的功能块将按定义的这些标识进行计数器寻址。

1. 输入:

2. 输出:

CD522 配置

CD522 I/O映射

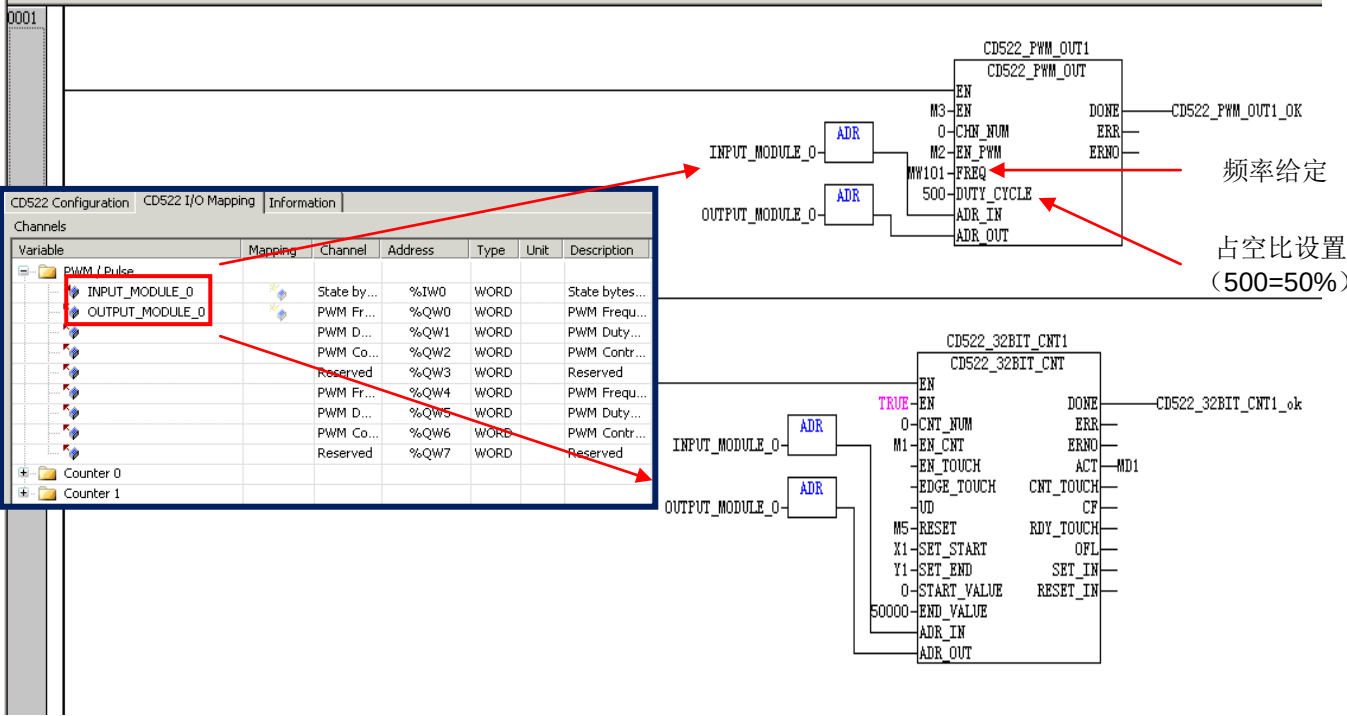
信息：

通道

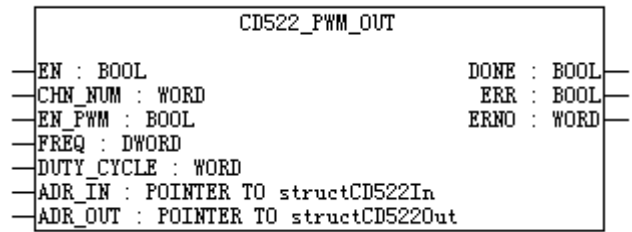
变量	映射	通道	地址	类型	单位	描述
PWM / 脉冲						
INPUT_MODULE0		状态 ...	%IW0	WORD		状态 ...
OUTPUT_MODULE0		PWM ...	%QW0	WORD		PWM ...
		PWM ...	%QW1	WORD		PWM ...
		PWM ...	%QW2	WORD		PWM ...
		保留	%QW3	WORD		保留
		PWM ...	%QW4	WORD		PWM ...
		PWM ...	%QW5	WORD		PWM ...
		PWM ...	%QW6	WORD		PWM ...
		保留	%QW7	WORD		保留
计数器0						
计数器1						

2.3.2 编程开发

调用 CD522_PWM_OUT 功能块。



2.3.3 PWM_OUT 功能块说明



功能块信息

适用的PLC运行系统:	V2.2 版本
适用的S500 I/O 模块(DC551/DC505)固件	V2.2 版本
适用的CD522固件	V2.2 版本
所属库:	CD522_AC500_V13.lib

功能块类型

功能块（带有历史数据）

参数

实例	输入	CD522_PWM_OUT	实例名称
EN	输入	BOOL	启用功能块处理
CNT_NUM	输入	WORD	计数器选择: 0/1

EN_PWM	输入	BOOL	启用PWM输出
FREQ	输入	DWORD	PWM频率值从1到100 KHz
DUTY_CYCLE	输入	WORD	占空循环从0到100.0 (0到100%), 设置为TRUE的时间百分比
ADR_IN	输入	POINTER TO STRUCTCD522COUNTERIN	到计数器输入结构的指针
ADR_OUT	输入	POINTER TO STRUCTCD522COUNTEROUT	到计数器输入结构的指针
DONE	输出	BOOL	功能块处理完成
ERR	输出	BOOL	功能块故障信息
ERNO	输出	WORD	故障代码

描述

CD522 模块可用于以可调整的占空比(ON/OFF 比率)控制输出脉冲信号。在 PLC 配置中使用模块的参数可以配置 PWM 操作模式（见下图）。

CD522 配置 CD522 I/O映射 信息：			
参数	类型	值	缺省值
SSI 0 频率	Enumeration of BYTE	200 kHz	200 kHz
SSI 0 以位计的分辨率	BYTE(8..32)	16	16
SSI 0 代码类型	Enumeration of BYTE	二进制	二进制
SSI 0 轮询时间	BYTE(1..255)	10	10
5V 传感器 0 电源	Enumeration of BYTE	开	关
模式计数器1	Enumeration of BYTE	无计数器	无计数器
频率极限FC1	Enumeration of BYTE	无过滤器	无过滤器
输入级FC1	Enumeration of BYTE	0-24 VDC	0-24 VDC
SSI 1 频率	Enumeration of BYTE	200 kHz	200 kHz
SSI 1 以位计的分辨率	BYTE(8..32)	16	16
SSI 1 代码类型	Enumeration of BYTE	二进制	二进制
SSI 1 轮询时间	BYTE(1..255)	10	10
5V 传感器 1 电源	Enumeration of BYTE	关	关
短路探测和传感器	Enumeration of BYTE	关	关
通信错误时输出行为	Enumeration of BYTE	关	关
替换值	WORD(0..65535)	0	0
PWM / 脉冲			
输出O0, 通道配置	Enumeration of BYTE	PWM	数字输出
输出O1, 通道配置	Enumeration of BYTE	数字输出	数字输出
计数器0			
输入I3, 通道配置	Enumeration of BYTE	脉冲	数字输入

然后，在初始化阶段（电源开通、冷启动、热启动）被激活。

高速计数器与用户程序之间的数据交换是通过输入和输出操作数来进行的。当模块 CD522 被添加到 I/O 总线配置时，这些必要的操作数被自动创建并保留。

CD522 模块提供了 2 个独立的可用于 PWM 模式的输出（O0 和 O1）。两者有相同的特性，可以独立工作。

功能块 CD522_PWM_OUT 由输入 EN_PWM 进行控制，由输入 FREQ 配置频率，由 PWM 输出（脉宽模块）的输入占空循环 DUTY_CYCLE 配置频率。

EN BOOL (使能)

要启用功能块处理，输入 EN 必须持续设置为 TRUE。如果 EN=FALSE，功能块将不被处理。

当第一次调用功能块时，系统将自动对输入的有效性和真实性核查。若非如此，处理将中断，对应的误差显示在输出端 ERR/ERNO。

CNT_NUM WORD (计数器选择)

计数器选择：0/1

EN_PWM BOOL (启用 PWM 输出)

如果 EN_PWM = TRUE，则启用脉宽调制。如果 EN_PWM = FALSE，则不进行脉宽调制。输入 EN_PWM 与控制字节的输出位 7 相对应。

FREQ DWORD (频率)

输入 FREQ 用于设定 PWM 的频率。频率从 1 到 100 KHz。

为选择高于 32767 Hz (字限制)的频率，功能块自动选择 FREQ 值的单位（1 Hz 到 10 Hz）。在手动选择时，它取决于“控制字节”（control byte）的第 0 位。如果控制字节的第 0 位是 TRUE, 则频率 x10。

输入 FREQ 对应于输出字“PWM 频率”（PWM Frequency）。

DUTY_CYCLE WORD (占空循环)

输入 DUTY_CYCLE 用于设定时间的百分比。占空循环从 0.0 到 100.0 (0 %-100 %); 这个值不包括小数点。就是说，例如要设置为 75.8 %，则这个值为 758。最大许可的值为 1000，设定的占空循环为 100%。

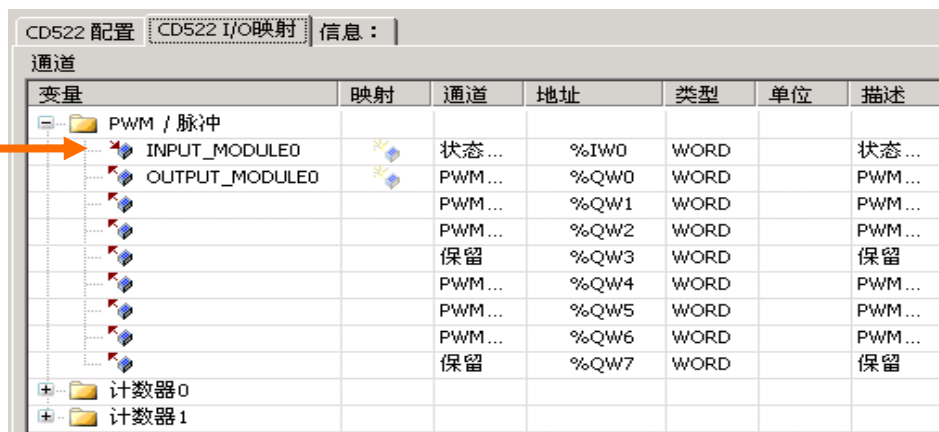
对于高速输出 O0 和 O1，黄色 LED 灯的亮度取决于占空循环值的设定（从 0 到 100%）。如果这个值超过 1000 或低于 0，在 ERR 和 ERNO 输出端将显示错误代码，红色错误 LED 灯 CH-ERR1 将闪烁。以最后输入的有效值为准，如果以前没有输入值，则这个值是 0。输入 DUTY_CYCLE 对应于输出字“PWM 占空循环/脉冲”（PWM Duty cycle/pulse）。

ADR_IN POINTER TO structCD522counterin (地址输入)

在这个输入，应该连接 CD522 计数器输入结构的第一个输入数据的地址。需要使用 ADR 操作数。如果输入 ADR_IN 没有连接，输出 DONE=FALSE 且 ERR=TRUE。

示例（对于计数器 0）：

1. 输入



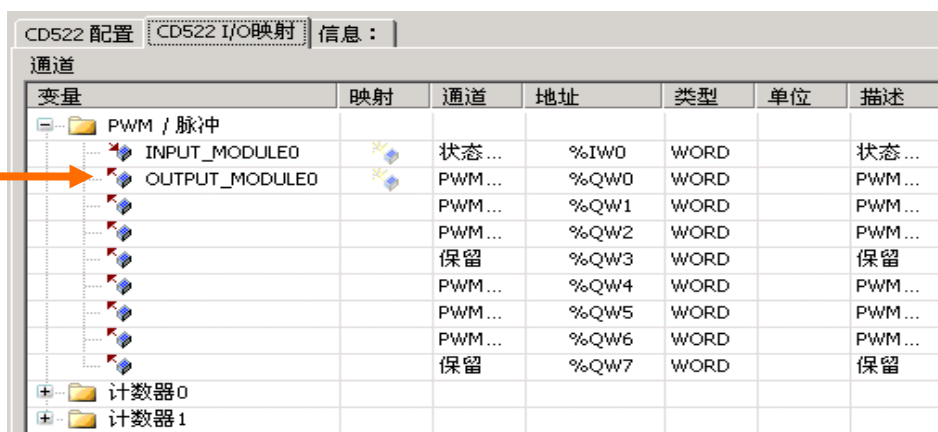
CD522 配置 CD522 I/O映射 信息:							
通道							
变量	映射	通道	地址	类型	单位	描述	
PWM / 脉冲							
INPUT_MODULE0		状态...	%IW0	WORD		状态...	
OUTPUT_MODULE0		PWM...	%QW0	WORD		PWM...	
		PWM...	%QW1	WORD		PWM...	
		PWM...	%QW2	WORD		PWM...	
		保留	%QW3	WORD		保留	
		PWM...	%QW4	WORD		PWM...	
		PWM...	%QW5	WORD		PWM...	
		PWM...	%QW6	WORD		PWM...	
		保留	%QW7	WORD		保留	
计数器0							
计数器1							

ADR_OUT POINTER TO structCD522counterout (地址输出)

在这个输入，应该连接 CD522 计数器输出结构的第一个输出数据的地址。需要使用 ADR 操作数。如果输入 ADR_OUT 没有连接，输出 DONE=FALSE 且 ERR=TRUE。

示例（对于计数器 0）：

2. 输出:



CD522 配置 CD522 I/O映射 信息:							
通道							
变量	映射	通道	地址	类型	单位	描述	
PWM / 脉冲							
INPUT_MODULE0		状态...	%IW0	WORD		状态...	
OUTPUT_MODULE0		PWM...	%QW0	WORD		PWM...	
		PWM...	%QW1	WORD		PWM...	
		PWM...	%QW2	WORD		PWM...	
		保留	%QW3	WORD		保留	
		PWM...	%QW4	WORD		PWM...	
		PWM...	%QW5	WORD		PWM...	
		PWM...	%QW6	WORD		PWM...	
		保留	%QW7	WORD		保留	
计数器0							
计数器1							

DONE BOOL (完成)

完成处理时通过 EN 输入触发输出 DONE 信号。完成处理或发生错误后，DONE 为 TRUE 至少保持一个 PLC 循环。

这个输出 DONE 总是要与 ERR 输出一起综合判断。如果 ERR 是 TRUE，输出 ERNO 反映相应的错误号码。否则，输出参数值是有效的。

ERR BOOL (故障)

输出 ERR 表示在功能块运行过程中检测到的任何错误。这个输出总是要与输出 DONE 一起综合判断。如果 DONE=TRUE 且 ERR=TRUE，则检测到处理错误。ERNO 输出的值提供了相应的错误号码。

ERNO WORD (故障代码)

输出 ERNO 提供了根据故障原因给出的错误代码。ERNO 的值仅在 DONE 为 TRUE 以及 ERR 为 TRUE 时有效。

IL 中的功能调用

```
LD          CD522PWMOUT_ADR_IN
ADR
ST          CD522PWMOUT.ADR_IN

LD          CD522PWMOUT_ADR_OUT
ADR
ST          CD522PWMOUT.ADR_OUT

CAL          CD522PWMOUT (
EN:= CD522PWMOUT_EN,
FREQ:= CD522PWMOUT_FREQ,
DUTY_CYCLE:= CD522PWMOUT_DUTY_CYCLE)

LD          CD522PWMOUT.DONE
ST          CD522PWMOUT_DONE

LD          CD522PWMOUT.ERR
ST          CD522PWMOUT_ERR

LD          CD522PWMOUT.ERNO
ST          CD522PWMOUT_ERNO
```



注释

在IL中，功能调用必须写在一行中。

ST 中的功能调用

```
CD522PWMOUT    (EN:= CD522PWMOUT_EN,
                FREQ:= CD522PWMOUT_FREQ,
                DUTY_CYCLE:= CD522PWMOUT_DUTY_CYCLE,
                ADR_IN:= ADR(CD522PWMOUT_ADR_IN),
                ADR_OUT:= ADR(CD522PWMOUT_ADR_OUT));

CD522PWMOUT_DONE := CD522PWMOUT.DONE;
CD522PWMOUT_ERR  := CD522PWMOUT.ERR;
CD522PWMOUT_ERNO := CD522PWMOUT.ERNO;
```

2.4 脉冲输出应用

利用 CD522 计数器与 PULSE 功能，结合安川 SGM 系列伺服，实现位置控制功能。

2.4.1 设置过程

- 步骤 1~4：参见上节中的步骤 1~4；
- 步骤 5：CD522 配置中，点击 PWM/Pulse 子目录用以选择通道来配置 PULSE 模式。

CD522 配置 CD522 I/O映射 信息：			
参数	类型	值	缺省值
SSI 0 频率	Enumeration of BYTE	200 kHz	200 kHz
SSI 0 以位计的分辨率	BYTE(8..32)	16	16
SSI 0 代码类型	Enumeration of BYTE	二进制	二进制
SSI 0 轮询时间	BYTE(1..255)	10	10
5V 传感器 0 电源	Enumeration of BYTE	开	关
模式计数器1	Enumeration of BYTE	无计数器	无计数器
频率极限FC1	Enumeration of BYTE	无过滤器	无过滤器
输入级FC1	Enumeration of BYTE	0-24 VDC	0-24 VDC
SSI 1 频率	Enumeration of BYTE	200 kHz	200 kHz
SSI 1 以位计的分辨率	BYTE(8..32)	16	16
SSI 1 代码类型	Enumeration of BYTE	二进制	二进制
SSI 1 轮询时间	BYTE(1..255)	10	10
5V 传感器 1 电源	Enumeration of BYTE	关	关
短路探测和传感器	Enumeration of BYTE	关	关
通信错误时输出行为	Enumeration of BYTE	关	关
替换值	WORD(0..65535)	0	0
PWM / 脉冲			
输出 O0，通道配置	Enumeration of BYTE	PWM	数字输出
输出 O1，通道配置	Enumeration of BYTE	数字输出	数字输出
计数器0			
输入 I3，通道配置	Enumeration of BYTE	脉冲	数字输入

- 步骤 6：定义模块的输入输出标识。CD522 库文件的功能块将按定义的这些标识进行计数器寻址。

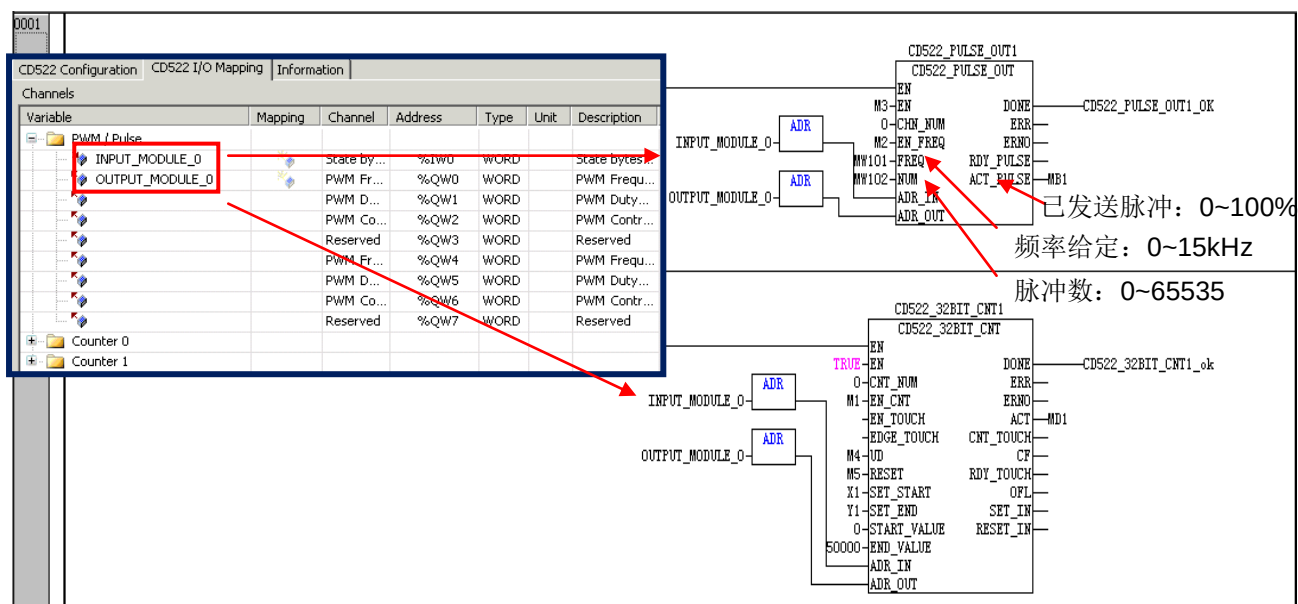
1. 输入:

2. 输出:

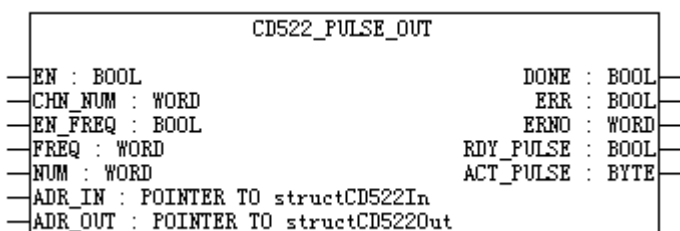
CD522 配置							
CD522 I/O映射							
信息:							
变量	映射	通道	地址	类型	单位	描述	
PWM / 脉冲							
INPUT_MODULE0		状态...	%IW0	WORD		状态...	
OUTPUT_MODULE0		PWM...	%QW0	WORD		PWM...	
		PWM...	%QW1	WORD		PWM...	
		PWM...	%QW2	WORD		PWM...	
		保留	%QW3	WORD		保留	
		PWM...	%QW4	WORD		PWM...	
		PWM...	%QW5	WORD		PWM...	
		PWM...	%QW6	WORD		PWM...	
		保留	%QW7	WORD		保留	
计数器0							
计数器1							

2.4.2 编程开发

调用 CD522_PULSE_OUT 功能块。



2.4.3 PULSE_OUT 功能块说明



功能块信息

适用的PLC运行系统:

V1.0.2版本

适用的S500 I/O 模块(DC551/DC505)固件	V1.11版本
适用的CD522固件	V2.1版本
所属库:	CD522_AC500_V13.lib

功能块类型

功能块（带有历史数据）

参数

实例	输入	CD522_PULSE_OUT	实例名称
EN	输入	BOOL	启用功能块处理
CNT_NUM	输入	WORD	计数器选择: 0/1
EN_FREQ	输入	BOOL	启用脉冲输出
FREQ	输入	DWORD	输出从0到15 KHz频率值。
NUM	输入	WORD	发送的脉冲数量从0到65535
ADR_IN	输入	POINTER TO STRUCTCD522COUNTERIN	到计数器输入结构的指针
ADR_OUT	输入	POINTER TO STRUCTCD522COUNTEROUT	到计数器输入结构的指针
DONE	输出	BOOL	功能块处理完成
ERR	输出	BOOL	功能块故障信息
ERNO	输出	WORD	故障代码
RDY_PULSE	输出	BOOL	当所有的脉冲都发送完成时, 设置为TRUE
ACT_PULSE	输出	BYTE	发送的脉冲百分比, 从0到100 %。 当发送了全部脉冲后才会获得100 %。

描述

CD522 模块可用于控制固定占空比(ON/OFF 比率为 50 %)的输出脉冲信号和多个以固定频率（可修改）发送的脉冲。在编程软件中可以通过设定模块的参数，设定为脉冲输出模式。（见下图）

CD522 配置 CD522 I/O映射 信息：			
参数	类型	值	缺省值
SSI 0 频率	Enumeration of BYTE	200 kHz	200 kHz
SSI 0 以位计的分辨率	BYTE(8..32)	16	16
SSI 0 代码类型	Enumeration of BYTE	二进制	二进制
SSI 0 轮询时间	BYTE(1..255)	10	10
5V 传感器 0 电源	Enumeration of BYTE	开	关
模式计数器1	Enumeration of BYTE	无计数器	无计数器
频率极限FC1	Enumeration of BYTE	无过滤器	无过滤器
输入级FC1	Enumeration of BYTE	0-24 VDC	0-24 VDC
SSI 1 频率	Enumeration of BYTE	200 kHz	200 kHz
SSI 1 以位计的分辨率	BYTE(8..32)	16	16
SSI 1 代码类型	Enumeration of BYTE	二进制	二进制
SSI 1 轮询时间	BYTE(1..255)	10	10
5V 传感器 1 电源	Enumeration of BYTE	关	关
短路探测和传感器	Enumeration of BYTE	关	关
通信错误时输出行为	Enumeration of BYTE	关	关
替换值	WORD(0..65535)	0	0
PWM / 脉冲			
输出 O0, 通道配置	Enumeration of BYTE	PWM	数字输出
输出 O1, 通道配置	Enumeration of BYTE	数字输出	数字输出
计数器0			
输入 I3, 通道配置	Enumeration of BYTE	PWM	数字输入

然后，在初始化阶段（电源开通、冷启动、热启动）被激活。

高速计数器与用户程序之间的数据交换是通过输入和输出操作数来进行的。当模块 CD522 被添加到 I/O 总线配置时，这些必要的操作数被自动创建并保留。

CD522 模块提供了 2 个独立的可用于 PULSE 模式的输出（O0 和 O1）。两者有相同的特性，可以独立工作。

功能块 CD522_PULSE_OUT 使用输入 EN_FREQ 来控制脉冲输出，使用输入 FREQ 配置频率，使用 NUM 配置脉冲数量。发送的脉冲数量能以百分比的形式显示出来（从 0 到 100%）。

EN BOOL (使能)

要启用功能块处理，输入 EN 必须持续设置为 TRUE。如果 EN=FALSE，功能块将不被处理。

当第一次调用功能块时，系统将自动对输入的有效性和真实性核查。若非如此，处理将中断，对应的误差显示在输出端 ERR/ERNO。

CNT_NUM WORD（计数器选择）

计数器选择：0/1

EN_FREQ BOOL（启用频率输出）

当 EN_FREQ=TRUE 时，则启用脉冲发送功能。当 EN_FREQ=FALSE 时，则不发送脉冲。如果已经开始发送脉冲且 EN_FREQ=FALSE，则停止发送脉冲。当 EN_FREQ 再次=TRUE 时，将重新开始发送脉冲，从上一次发送的脉冲数量开始到输入 NUM 设定的脉冲值。输入 EN_FREQ 与控制字节的输出位 7 相对应。

FREQ DWORD (频率)

输入 FREQ 用于设定输出的频率。频率从 0 到 15 KHz。
输出 FREQ 对应于输出字“PWM 频率”（PWM Frequency）。

NUM WORD (脉冲数)

输入 NUM 用于设置要发送的脉冲数量。数量从 1 到 65535 个脉冲。
输出 NUM 对应于输出字“PWM 频率”（PWM Frequency）。

ADR_IN POINTER TO structCD522counterin (地址输入)

在这个输入，应该连接 CD522 计数器输入结构的第一个输入数据的地址。需要使用 ADR 操作数。如果输入 ADR_IN 没有连接，输出 DONE=FALSE 且 ERR=TRUE。
示例（对于计数器 0）：

1. 输入



变量	映射	通道	地址	类型	单位	描述
PWM / 脉冲						
INPUT_MODULE0		状态...	%IW0	WORD		状态...
OUTPUT_MODULE0		PWM ...	%QW0	WORD		PWM ...
		PWM ...	%QW1	WORD		PWM ...
		PWM ...	%QW2	WORD		PWM ...
		保留	%QW3	WORD		保留
		PWM ...	%QW4	WORD		PWM ...
		PWM ...	%QW5	WORD		PWM ...
		PWM ...	%QW6	WORD		PWM ...
		保留	%QW7	WORD		保留
计数器 0						
计数器 1						

ADR_OUT POINTER TO structCD522counterout (地址输出)

在这个输入，应该连接 CD522 计数器输出结构的第一个输出数据的地址。需要使用 ADR 操作数。如果输入 ADR_OUT 没有连接，输出 DONE=FALSE 且 ERR=TRUE。
示例（对于计数器 0）：

2. 输出:



变量	映射	通道	地址	类型	单位	描述
PWM / 脉冲						
INPUT_MODULE0		状态...	%IW0	WORD		状态...
OUTPUT_MODULE0		PWM ...	%QW0	WORD		PWM ...
		PWM ...	%QW1	WORD		PWM ...
		PWM ...	%QW2	WORD		PWM ...
		保留	%QW3	WORD		保留
		PWM ...	%QW4	WORD		PWM ...
		PWM ...	%QW5	WORD		PWM ...
		PWM ...	%QW6	WORD		PWM ...
		保留	%QW7	WORD		保留
计数器 0						
计数器 1						

DONE BOOL (完成)

完成处理时通过 EN 输入触发输出 DONE 信号。完成处理或发生错误后，DONE 为 TRUE 至少保持一个 PLC 循环。

这个输出 DONE 总是要与 ERR 输出一一起综合判断。如果 ERR 是 TRUE，输出 ERNO 反映相应的错误号码。否则，输出参数值是有效的。

ERR BOOL (故障)

输出 ERR 表示在功能块运行过程中检测到的任何错误。这个输出总是要与输出 DONE 一起综合判断。如果 DONE=TRUE 且 ERR=TRUE，则检测到处理错误。ERNO 输出的值提供了相应的错误号码。

ERNO WORD (故障代码)

输出 ERNO 提供了根据故障原因给出的错误代码。ERNO 的值仅在 DONE 为 TRUE 以及 ERR 为 TRUE 时有效。

RDY_PULSE BOOL (准备发送脉冲)

当所有的脉冲都被发送了时 RDY_PULSE 被设置为 TRUE

ACT_PULSE BYTE (实际发送的脉冲百分比)

输出 ACT_PULSE 返回发送的脉冲百分比，从 0 到 100 %。

当发送完全部脉冲后才会获得 100 %。

对于高速输出 O0 和 O1，黄色 LED 灯的亮度取决于脉冲输出的数量（从 0 到 100 %），当到达 100 % 的值时，黄色 LED 灯为熄灭状态。

输出 ACT_PULSE 对应于“脉冲状态字节”的输入字节。

IL 中的功能调用

LD	CD522PULSEOUT_ADR_IN
ADR	
ST	CD522PULSEOUT.ADR_IN
LD	CD522PULSEOUT_ADR_OUT
ADR	
ST	CD522PULSEOUT.ADR_OUT
CAL	CD522PULSEOUT (EN:= CD522PULSEOUT_EN, EN_FREQ:=CD522PULSEOUT_EN_FREQ, FREQ:= CD522PULSEOUT_FREQ, NUM:= CD522PULSEOUT_NUM)
LD	CD522PULSEOUT.DONE
ST	CD522PULSEOUT_DONE
LD	CD522PULSEOUT.ERR
ST	CD522PULSEOUT_ERR

LD	CD522PULSEOUT. ERNO
ST	CD522PULSEOUT_ERNO
LD	CD522PULSEOUT. RDY_PULSE
ST	CD522PULSEOUT_RDY_PULSE
LD	CD522PULSEOUT. ACT_SEND
ST	CD522PULSEOUT_ACT_PULSE



注释

在IL中，功能调用必须写在一行中。

ST 中的功能调用

```
CD522PULSEOUT      (EN:= CD522PULSEOUT_EN,
                    FREQ:= CD522PULSEOUT_FREQ,
                    NUMBER:= CD522PULSEOUT_NUMBER,
                    ADR_IN:= ADR(CD522PULSEOUT_ADR_IN),
                    ADR_OUT:= ADR(CD522PULSEOUT_ADR_OUT));
```

```
CD522PULSEOUT_DONE := CD522PULSEOUT.DONE;
CD522PULSEOUT_ERR  := CD522PULSEOUT.ERR;
CD522PULSEOUT_ERNO := CD522PULSEOUT.ERNO;
CD522PULSEOUT_RDY_PULSE := CD522PULSEOUT.RDY_PULSE;
CD522PULSEOUT_ACT_PULSE := CD522PULSEOUT.ACT_PULSE;
```

3 eCO 集成 I/O 的 PWM 输出

3.1 概述

集成 I/O 支持 2 个 PWM 输出通道，最高频率为 20KHz。可通过 2 种不同的配置模式控制 2 项输出：

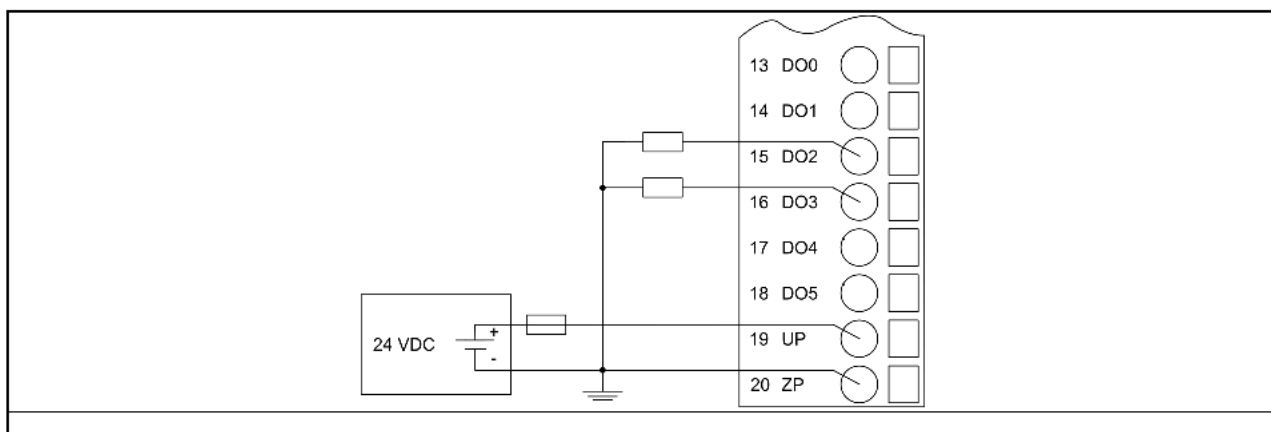
- ✧ 频率模式
- ✧ 循环时间模式

在频率模式下，通过设置 PWM 通道的输入变量 FREQ 和 DUTY_CYCLE，集成 I/O 的输出通道 2 和 3 能产生 125 Hz~20 kHz 的频率，具有 0..100 % 的占空比波形。

在循环时间模式下，通过设置 PWM 通道的输入变量 CYCLE_TIME 和 CYCLE_DUTY，集成 I/O 的输出通道 2 和 3 能产生 125 Hz~20 kHz 的频率，具有 0..100 % 的占空比波形。

3.2 脉冲输出连接输出负载的接线

下图显示 eCO CPU 上集成 I/O 的 PWM/脉冲输出与输出负载的电气连接



图：eCO 集成 I/O PWM 输出接线图



NOTICE

输出ON延迟：典型 25 ℃时，1.5 μ s @ 0.5 A 负载。

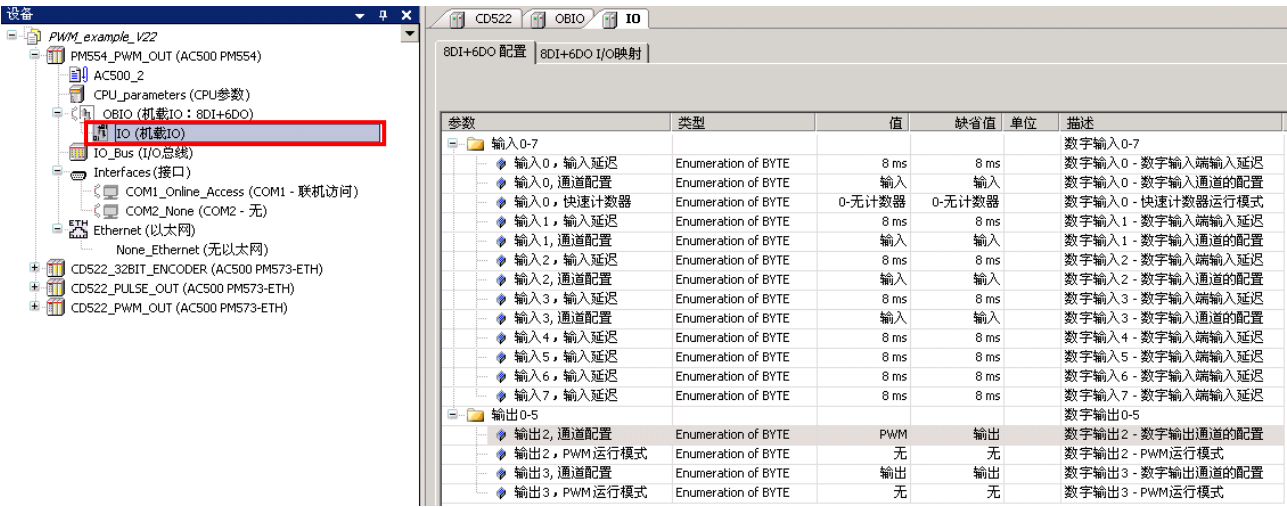
输出OFF延迟：典型 25 ℃时，3 μ s @ 0.5 A 负载。

当使用高频率及非常高或低占空比时，需要考虑上述情况。

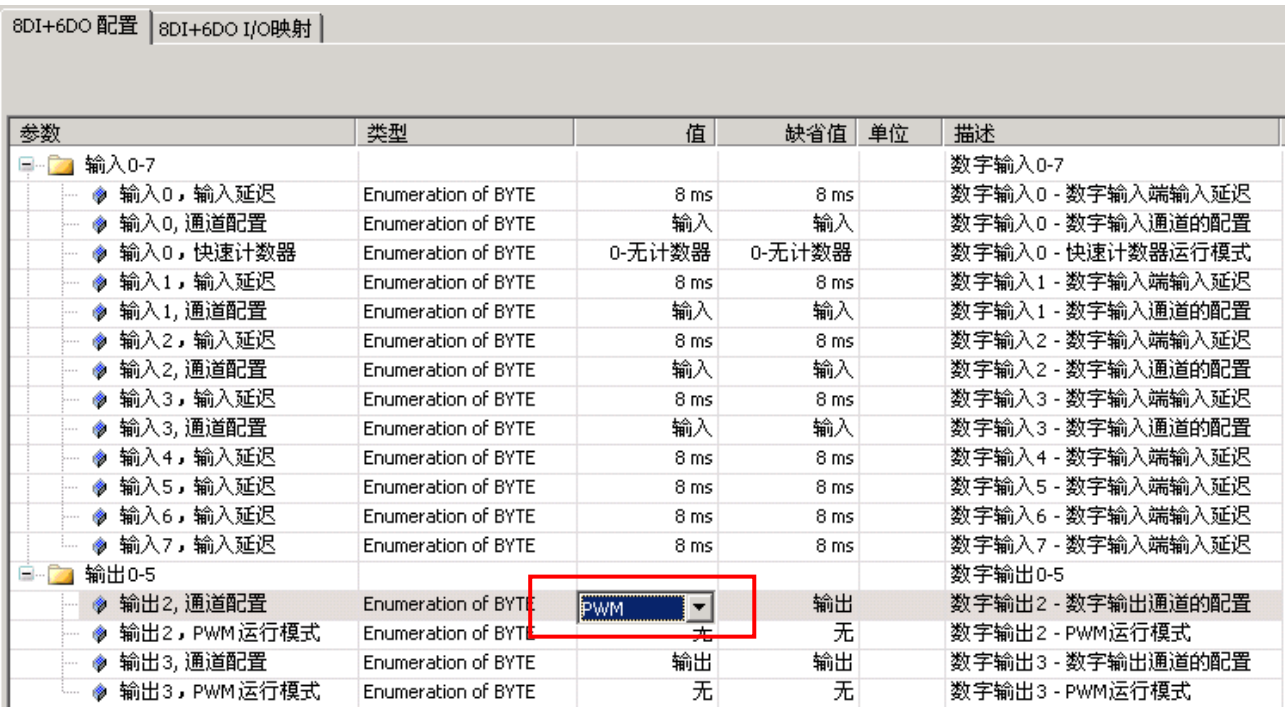
3.3 频率输出应用

3.3.1 设置过程

- 步骤 1: 在 ABB 配置器中，双击“PM554 I/O（机载 IO）”，并在右侧界面中选择“8DI+6DO 配置”。

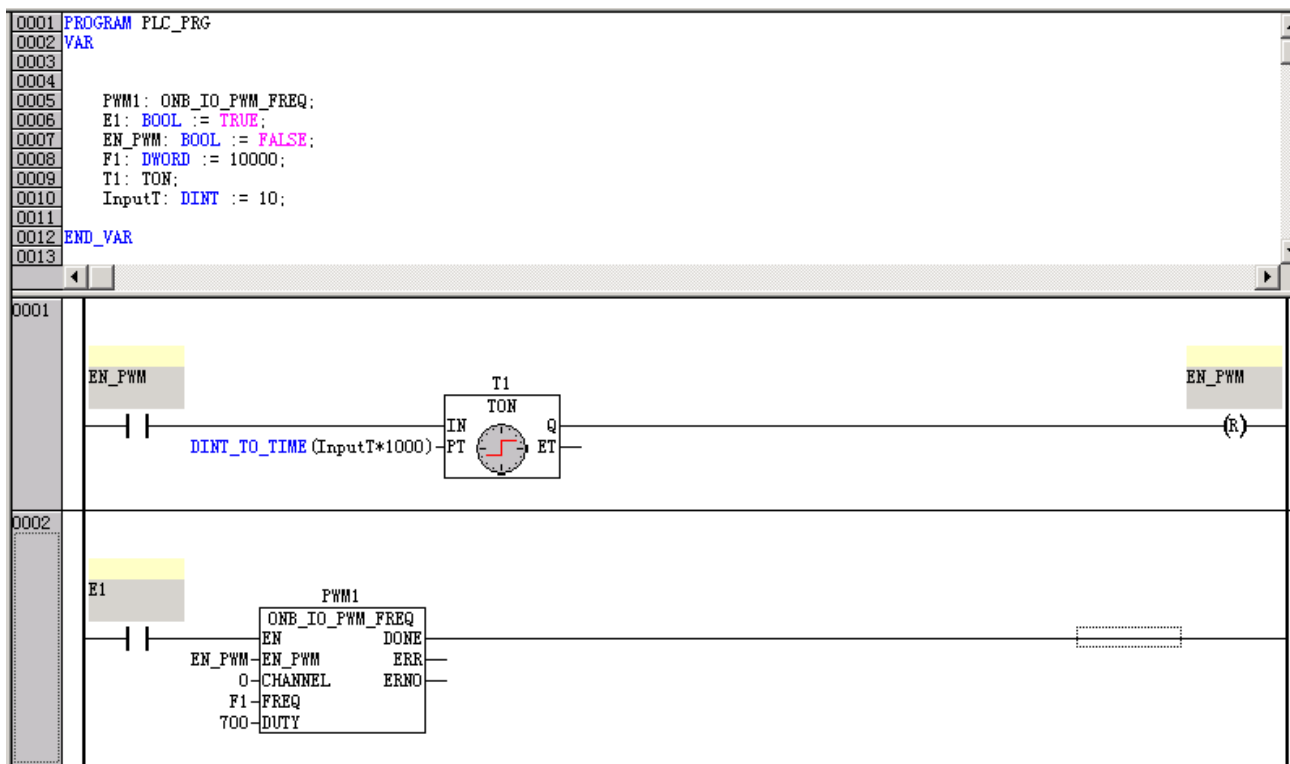


- 步骤 2: 选择通道 2 的输出方式为“PWM”；



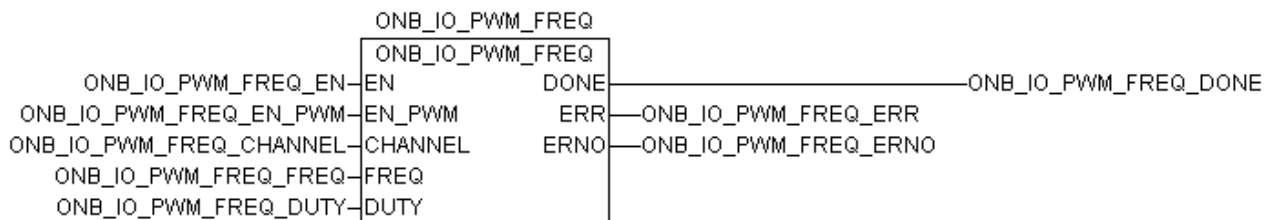
3.3.2 编程开发

调用 ONB_IO_PWM_FREQ 功能块。



3.3.3 ONB_IO_PWM_FREQ 功能块说明

ONB_IO_PWM_FREQ 通过设置板载 I/O 上的频率和占空比来产生 PWM 信号



功能块 ONB_IO_PWM_FREQ 用于通过板载 I/O 生成一个带有可调节的频率和占空比的 PWM 信号。

功能块数据

适用的 PLC 运行系统:	V1.3.0 版本	注释:
所属库:	OnBoardIO_AC500_V13.LIB	

参数

实例		ONB_IO_PWM_FREQ	功能描述
EN	输入	BOOL	使能功能块
EN_PWM	输入	BOOL	使能 PWM 功能
CHANNEL	输入	BYTE	板载 I/O 上 PWM 通道的数量
FREQ	输入	DWORD	PWM 的频率

DUTY	输入	WORD	适于 PWM 的占空比
DONE	输入	BOOL	就绪信息
ERR	输出	BOOL	功能块错误信息
ERNO	输出	WORD	功能块错误代码



注意

两个 PWM 通道的频率必须是相同的。如果使用这两个通道,第一个 PWM 通道 (CHANNEL=0) 的频率对于两个通道都有效。第二个通道 (CHANNEL=1) 的频率必须设置为 0

描述

板载数字量输出 O2...O3 可用于可调整的占空比(ON/OFF 比率)控制输出脉冲信号。

操作模式通过初始化操作（接通电源、冷启动、热启动）激活。

模块与用户程序之间通过输入和输出操作数来进行数据交换。目标设置为 CPU PM55x 或 PM56x 时，自动创建和保留这些必要的操作数。

CPU PM55x 或 PM56x 提供可用于 PWM 模式的 2 个输出（O2 和 O3）。两者规格相同，并使用相同的频率。占空时间可独立调节。

功能块 ONB_IO_PWM_FREQ 由输入 EN_PWM 进行控制，由输入 FREQ 配置频率，由 PWM 输出（脉宽模块）的输入占空比 DUTY 配置频率。

通过使用两个功能块可实现 PWM 功能：

- ONB_IO_PWM_FREQ（此功能块）
- ONB_IO_PWM_TIME

EN BOOL (使能)

通过输入 EN，激活(EN = TRUE) 或取消激活(EN = FALSE)该功能块。当该功能块使能时，当前输出数据有效。

EN_PWM BOOL (使能 脉宽 调制)

如果 EN_PWM=TRUE，则使能脉宽调制输出。如果 EN_PWM = FALSE，则不进行脉宽调制。

输入 EN_PWM 与“控制字节 PWM x”的输出位 7 相对应。

CHANNEL BYTE (通道)

CHANNEL 包含由功能块管理的通道编号：

- CHANNEL = 0: 输出 O2 被管理
- CHANNEL = 1: 输出 O3 被管理

FREQ DWORD (方波)

输入 FREQ 用于设定 PWM 的频率。频率从 125Hz 调整为 20000Hz（20KHZ）。

频率值由双字(DWORD)定义,但内部通讯通过 WORD 实现。对于大于 65535 的频率，则在内部使用一个附加位作为乘法器。因此对于这些频率则需 x10 Hz。

附加的乘法器位是“控制字节 PWM x”中的位 0..1。

输入 FREQ 对应于输出字“PWM x -频率/循环时间”（PWM x -Frequency/Cycle time）。

DUTY WORD (占空比)

输入 DUTY 用于设定时间的百分比，置为 TRUE。占空比从 0 到 1000(0.0 %-100.0 %); 这个值不包括小数点。例如要设置 75.8%，则这个值应写为 758。许可的最大值为 1000，即设定的占空比为 100.0 %。

对于高速输出 O2 和 O3，黄色 LED 灯的亮度取决于占空比的设定（从 0 到 100%）。



注意

如果写入值大于 1000，将读取 1000。如果写入值小于 1000，将读取 0。在这两种情况下不会产生任何错误消息。以最后输入的有效值为准，如果没有输入数值，则该值为 0。

输入 DUTY 对应于输出字“PWM x -占空比/占空时间”（PWM x - Duty cycle/Duty time）。

DONE BOOL (完成)

输出 DONE 表示任务执行的状态。在过程完成或者终止后（由于出错），每循环一个周期 DONE 被置为 TRUE。因此，这一输出要与输出 ERR 一起综合判断。如果 ERR 为 TRUE，则说明发生错误，此时，在输出 ERNO 中可以读出错误代码。

ERR BOOL (错误)

输出 ERR 标示功能块处理过程中是否有错误。该输出要与输出 DONE 一起综合判断。如果 DONE=TRUE 且 ERR=TRUE，则说明出错。此时，在输出 ERNO 中可以读出错误代码。

ERNO WORD (错误代码)

当输入无效的数值或在处理过程中出错时，输出 ERNO 将显示错误信息。ERNO 通常要与输出 DONE 和 ERR 一起综合判断。只有 DONE 为 TRUE，且 ERR 也为 TRUE 时，ERNO 的输出值才有效。输出 ERNO 的错误信息代码，在功能块描述部分有解释。

IL 中的功能调用

```
CAL   ONB_IO_PWM_FREQ(  
      EN                      := ONB_IO_PWM_FREQ_EN,  
      EN_PWM                  := ONB_IO_PWM_FREQ_EN_PWM,  
      CHANNEL                  := ONB_IO_PWM_FREQ_CHANNEL,  
      FREQ                     := ONB_IO_PWM_FREQ_FREQ,  
      DUTY                     := ONB_IO_PWM_FREQ_DUTY)
```

```
LD     ONB_IO_PWM_FREQ.DONE  
ST     ONB_IO_PWM_FREQ_DONE
```

```
LD     ONB_IO_PWM_FREQ.ERR  
ST     ONB_IO_PWM_FREQ_ERR
```

```
LD     ONB_IO_PWM_FREQ.ERNO  
ST     ONB_IO_PWM_FREQ_ERNO
```



注意

在 IL 中，功能调用必须写在一行中。

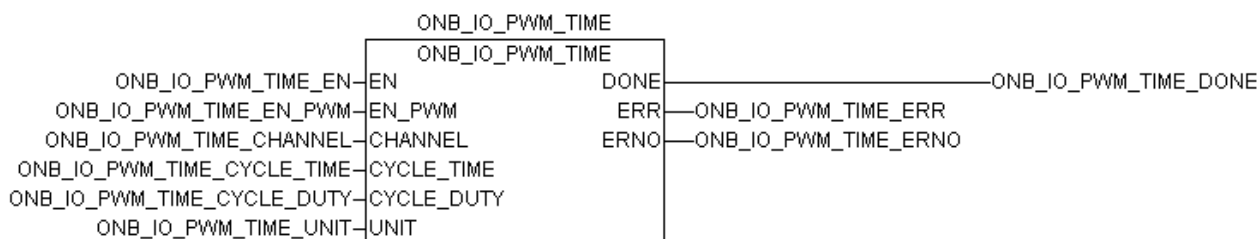
ST 中的功能调用

```
ONB_IO_PWM_FREQ      (EN           := ONB_IO_PWM_FREQ_EN,  
                        EN_PWM      := ONB_IO_PWM_EN_PWM,  
                        CHANNEL     := ONB_IO_PWM_FREQ_CHANNEL,  
                        FREQ        := ONB_IO_PWM_FREQ_FREQ,  
                        DUTY        := ONB_IO_PWM_FREQ_DUTY);
```

```
ONB_IO_PWM_FREQ_DONE      := ONB_IO_PWM_FREQ.DONE;  
ONB_IO_PWM_FREQ_ERR       := ONB_IO_PWM_FREQ.ERR;  
ONB_IO_PWM_FREQ_ERNO      := ONB_IO_PWM_FREQ.ERNO;
```

3.3.4 ONB_IO_PWM_TIME 功能块说明

ONB_IO_PWM_TIME 通过设置板载 I/O 上的周期和占空比来产生 PWM 信号



功能块 ONB_IO_PWM_TIME 用于通过设置板载 I/O 产生一个带有可调节的周期和占空比的 PWM 信号。

功能块数据

适用的 PLC 运行系统:	V1.3.0 版本	注释:
所属库:	OnBoardIO_AC500_V13.LIB	

参数

实例		ONB_IO_PWM_TIME	功能描述
EN	输入	BOOL	使能功能块
EN_PWM	输入	BOOL	使能 PWM 功能
CHANNEL	输入	BYTE	PWM 通道的数量 (0: O2; 1: O3)
CYCLE_TIME	输入	WORD	PWM 的循环时间 [μs 或 ms, 取决于 UNIT]

CYCLE_DUTY	输入	WORD	PWM 的占空时间 [μ s 或 ms, 取决于 UNIT]
UNIT	输入	BOOL	FALSE = [μ s], TRUE = [ms]
DONE	输入	BOOL	TRUE = 功能块处理完成
ERR	输出	BOOL	功能块错误信息
ERNO	输出	WORD	功能块错误代码



注意

两个 PWM 通道的循环时间必须是相同的。如果使用这两个通道，第一个 PWM 通道 (CHANNEL=0) 的循环时间对于两个通道都有效。第二个通道 (CHANNEL=1) 的循环时间必须设置为 0

描述

板载数字量输出 O2...O3 可用于可调整的占空比(ON/OFF 比率)控制输出脉冲信号。

操作模式通过初始化操作（接通电源、冷启动、热启动）激活。

模块与用户程序之间通过输入和输出操作数来进行数据交换。目标设置为 CPU PM55x 或 PM56x 时，自动创建和保留这些必要的操作数。

CPU PM55x 或 PM56x 提供可用于 PWM 模式的 2 个输出 (O2 和 O3)。两者规格相同，并使用相同的频率。占空时间可独立调节。

功能块 ONB_IO_PWM_TIME 由输入 EN_PWM 进行控制，由输入 CYCLE_TIME 配置频率，由 PWM 输出（脉宽模块）的输入 CYCLE_DUTY 配置频率。

通过使用两个功能块可实现 PWM 功能：

- ONB_IO_PWM_FREQ
- ONB_IO_PWM_TIME（此功能块）

EN BOOL (使能)

通过输入 EN，激活(EN = TRUE) 或取消激活(EN = FALSE)该功能块。当该功能块使能时，当前输出数据有效。

EN_PWM BOOL (使能 脉宽 调制)

如果 EN_PWM=TRUE，则使能脉宽调制输出。如果 EN_PWM = FALSE，则不进行脉宽调制。

输入 EN_PWM 与“控制字节 PWM x”的输出位 7 相对应。

CHANNEL BYTE (通道)

CHANNEL 包含由功能块管理的通道编号：

- CHANNEL = 0: 输出 O2 被管理
- CHANNEL = 1: 输出 O3 被管理

CYCLE_TIME WORD (周期时间)

输入 CYCLE_TIME 用于设定 PWM 的循环时间。循环时间从 50 μ s（即 20 kHz）调节为 8000 μ s（即 125 Hz）。也可通过配置输入变量 UNIT=TRUE 把该单位设置为“ms”。
输入 CYCLE_TIME 对应于输出字“PWM x -频率/循环时间”（PWM x -Frequency/Cycle time）。

CYCLE_DUTY WORD (占空比)

输入 CYCLE_DUTY 用于预定义输出的占空时间。输入 CYCLE_DUTY 用于设定 TRUE 信号的时间。正常的操作模式下，CYCLE_DUTY 的值应该小于或等于 CYCLE_TIME 的值。对于高速输出 O2 和 O3，黄色 LED 灯的亮度取决于设定的占空时间的值。

以最后输入的有效值为准，如果没有输入数值，则该值为 0。

输入 CYCLE_DUTY 对应于输出字“PWM x -占空比/占空时间”（PWM x - Duty cycle/Duty time）。

UNIT BOOL (单元)

输入 UNIT 用于预定义 PWM 输出的时间基。FALSE 将时基设置为 μ s，TRUE 为 ms。

DONE BOOL (完成)

输出 DONE 表示任务执行的状态。在过程完成或者终止后（由于出错），每循环一个周期 DONE 被置为 TRUE。因此，这一输出要与输出 ERR 一起综合判断。如果 ERR 为 TRUE，则说明发生错误，此时，在输出 ERNO 中可以读出错误代码。

ERR BOOL (错误)

输出 ERR 标示功能块处理过程中是否有错误。该输出要与输出 DONE 一起综合判断。如果 DONE=TRUE 且 ERR=TRUE，则说明出错。此时，在输出 ERNO 中可以读出错误代码。

ERNO WORD (错误代码)

当输入无效的数值或在处理过程中出错时，输出 ERNO 将显示错误信息。ERNO 通常要与输出 DONE 和 ERR 一起综合判断。只有 DONE 为 TRUE，且 ERR 也为 TRUE 时，ERNO 的输出值才有效。输出 ERNO 的错误信息代码，在功能块描述部分有解释。

IL 中的功能调用

```
CAL    ONB_IO_PWM_TIME(  
        EN                      := ONB_IO_PWM_TIME_EN,  
        EN_PWM                  := ONB_IO_PWM_TIME_EN_PWM,  
        CHANNEL                 := ONB_IO_PWM_TIME_CHANNEL,  
        CYCLE_TIME              := ONB_IO_PWM_TIME_CYCLE_TIME,  
        CYCLE_DUTY              := ONB_IO_PWM_TIME_CYCLE_DUTY,  
        UNIT                    := ONB_IO_PWM_TIME_UNIT)
```

```
LD      ONB_IO_PWM_TIME.DONE  
ST      ONB_IO_PWM_TIME_DONE
```

```
LD      ONB_IO_PWM_TIME.ERR  
ST      ONB_IO_PWM_TIME_ERR
```

```
LD      ONB_IO_PWM_TIME.ERNO  
ST      ONB_IO_PWM_TIME_ERNO
```



注意

在 IL 中，功能调用必须写在一行中。

ST 中的功能调用

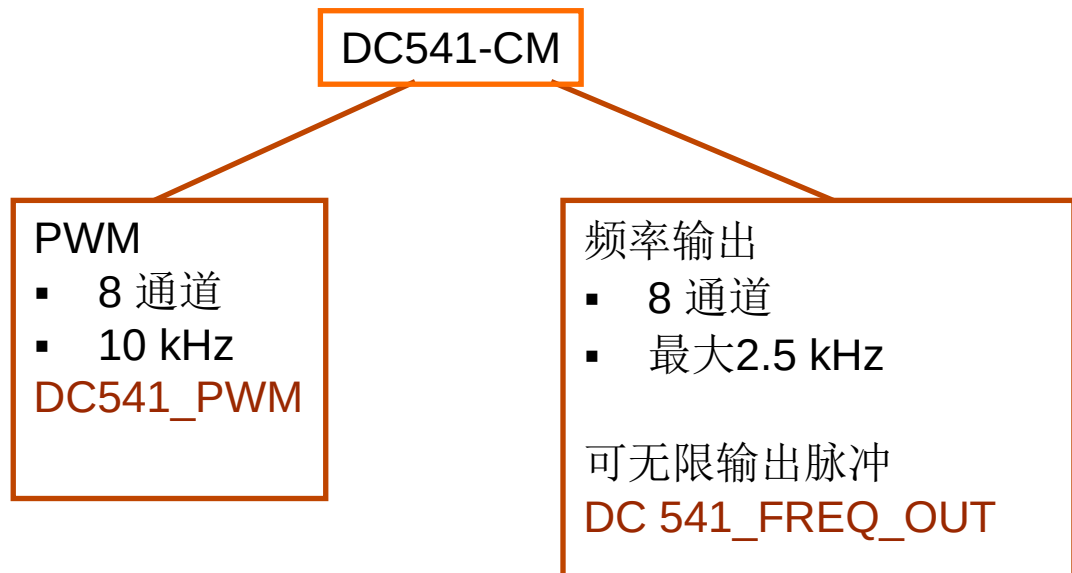
```
ONB_IO_PWM_TIME (EN                := ONB_IO_PWM_TIME_EN,  
                  EN_PWM            := ONB_IO_PWM_TIME_EN_PWM,  
                  CHANNEL           := ONB_IO_PWM_TIME_CHANNEL,  
                  CYCLE_TIME       := ONB_IO_PWM_TIME_CYCLE_TIME,  
                  CYCLE_DUTY      := ONB_IO_PWM_TIME_CYCLE_DUTY,  
                  UNIT             := ONB_IO_PWM_TIME_UNIT);
```

```
ONB_IO_PWM_TIME_DONE      := ONB_IO_PWM_TIME.DONE;  
ONB_IO_PWM_TIME_ERR      := ONB_IO_PWM_TIME.ERR;  
ONB_IO_PWM_TIME_ERNO     := ONB_IO_PWM_TIME.ERNO;
```

4 DC541 模块的 PWM 功能

4.1 概述

与其他 I/O 模块不同，DC541-CM 数字 I/O 模块（多功能模块）安装在 AC500 CPU 左侧的联机模块插槽内。与内部连接总线相连。这样，连接总线的所有功能对 DC541-CM 模块都可用。



通过功能块 DC541_PWM 使用 DC541 的脉宽模块功能。

功能块 DC541_PWM 输出一个占空比可调的脉冲信号。使用 8 位数字调整 ON 和 OFF 的时间。

输入 CYCLE 设定了最小的切换时间,即如果输出通过 PWM 切换到 FALSE 或 TRUE, 输出保持这个状态的最小时间(CYCLE μ s)。

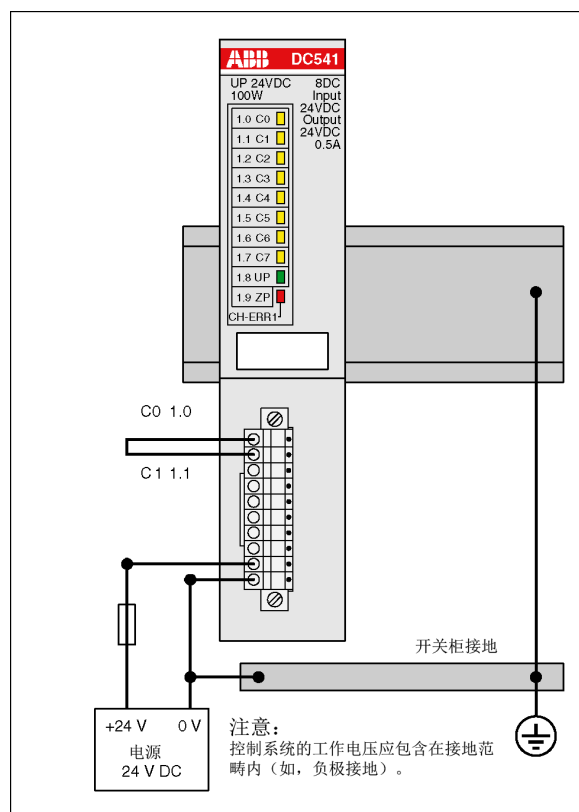
输入 CYCLE 设定的最小时间必须小于设备 DC541 的循环时间。根据其配置，DC541 的循环时间可以是 50、100 或 200 μ s。可以使用功能块 DC541_GET_CFG (输出 CYCLE) 获得循环时间。

4.2 脉冲输出连接输出负载的接线接线

在本例配置中，DC541 的通道 0 被配置为脉宽调制（PWM）。使用 DC541 的“时间和频率测量”功能测量输出信号。

按照如下连接通道：

DC541 / C0 ----- DC541 / C1



4.3 PWM 输出应用

硬件配置：

这个控制系统实例应包含以下配置：

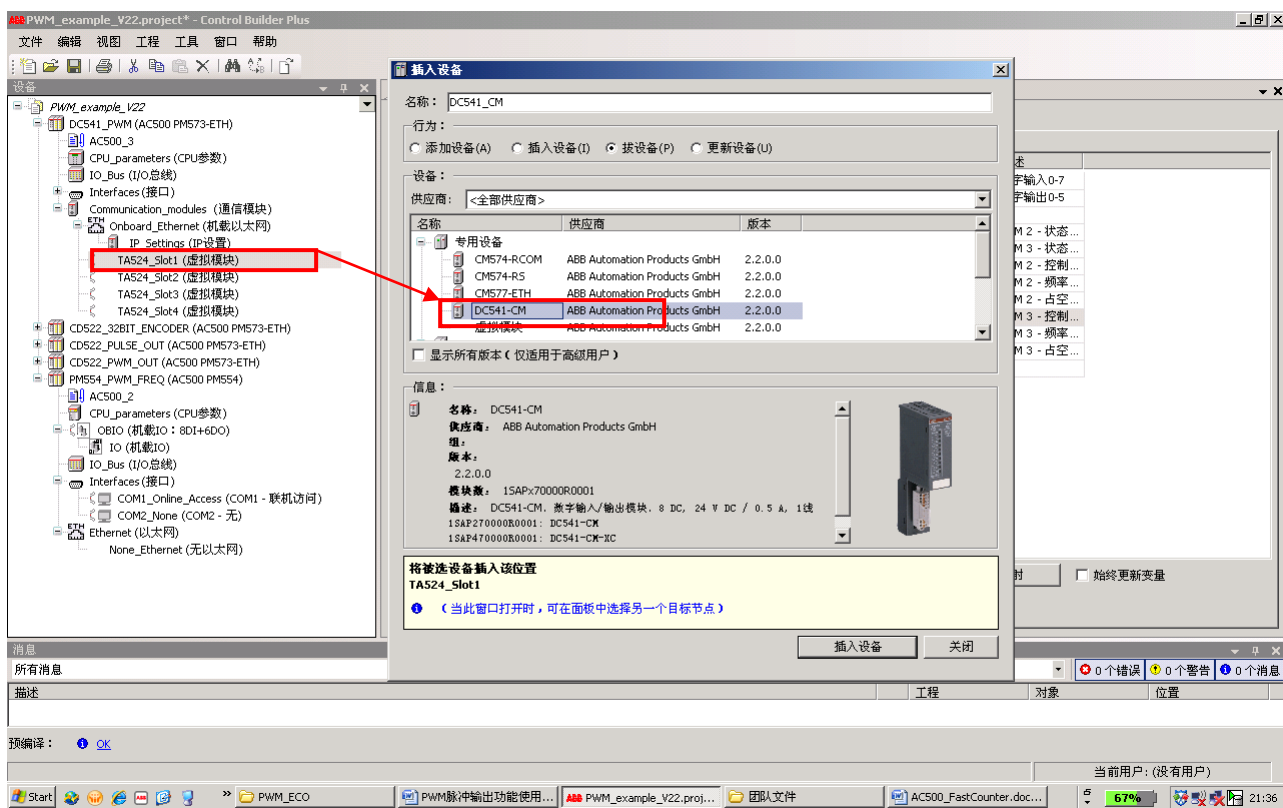
- CPU 底板 TB521 (两个通讯模块插槽)
- DC541 在通讯模块槽 1(CPU 左侧的第一个槽)
- PM573-ETH CPU 自带的内部以太网通讯模块
- I/O 总线上的 I/O 模块 DC532

PLC 配置：

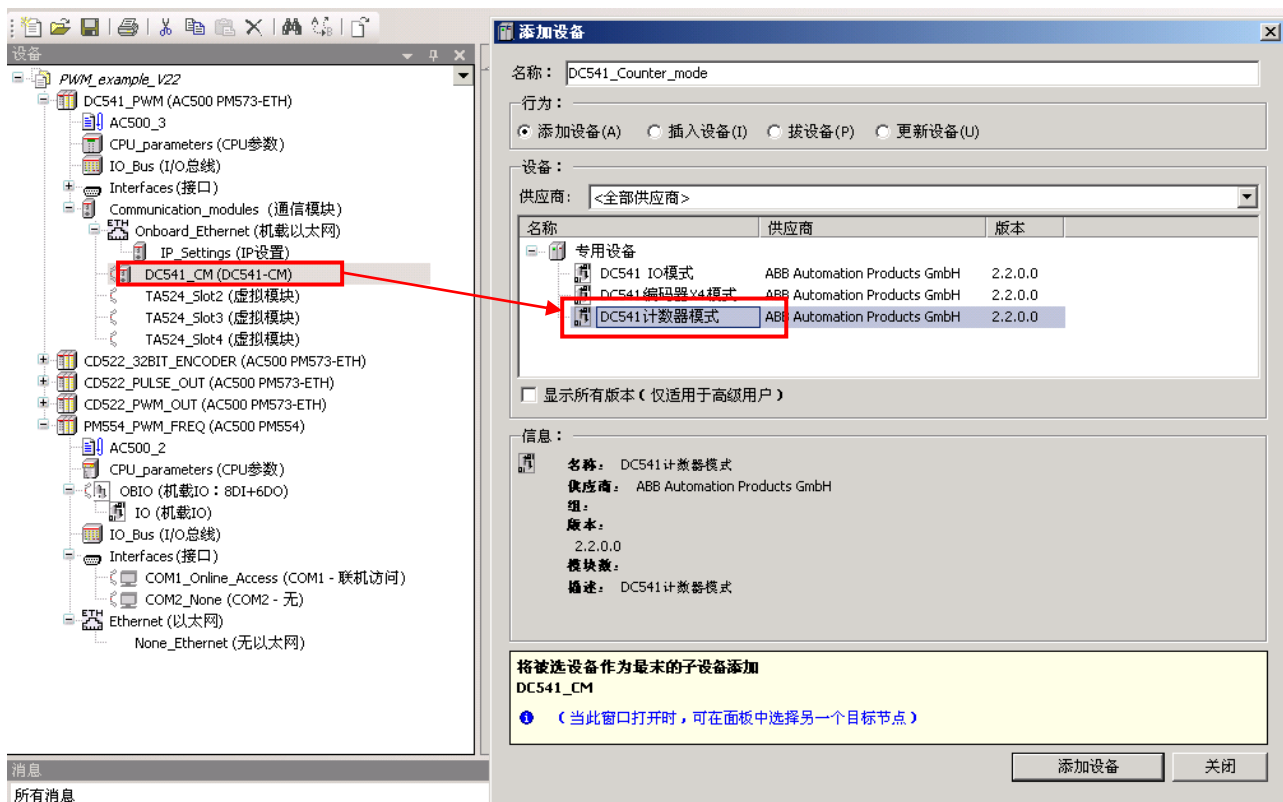
-DC541 位于插槽 1，为“计数器模式”			
-配置：	-通道	C0	PWM
		C1	FREQ
		C2...C7	输入

4.3.1 设置过程

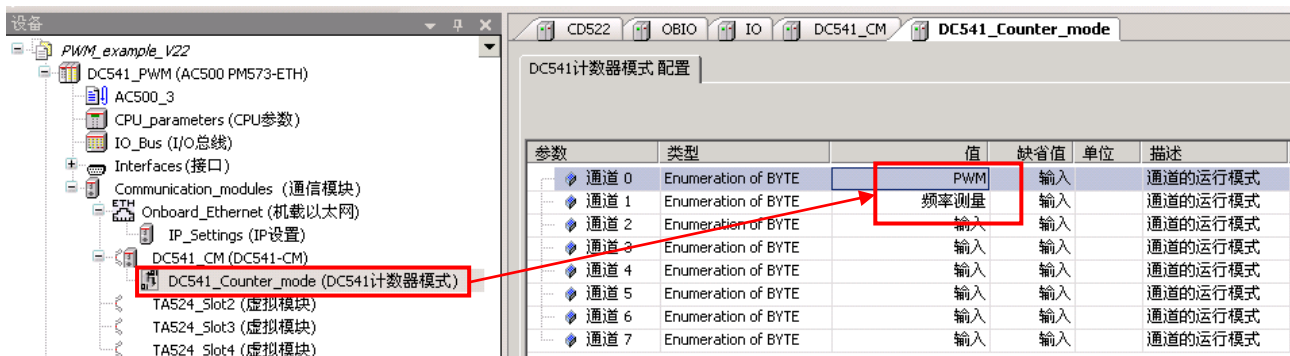
- 步骤 1：在 ABB 配置器中，选择“TA524_Slot1”，并右击插入“DC541-CM”。



- 步骤 2: 点击刚刚插入的“DC541-CM”，右击弹出菜单，并选择“添加设备”，设置 DC541 为“计数器模式”。



- 步骤 3: 双击刚刚插入的“DC541-CM 计数器模式”，设置通道 0 为“PWM”，设置通道 1 为“频率测量”。



4.3.1 编程开发

任务配置:

-任务 1: 循环程序/优先级= 10 / 间隔= t#1ms / PLC_PRG

循环程序 PLC_PRG 的用途:

循环程序 PLC_PRG 包含下面的功能:

- 读取 PLC_PRG 的循环

调用功能块 TASK_INFO;

-读取 DC541 的配置

调用功能块 DC541_GET_CFG

-读取 DC541 的状态

调用功能块 DC541_STATE

-读/写 DC541 的通道状态

调用功能块 DC541_IO

-调用 PWM 和 FREQ 的顺序控制

调用程序 proPWM_FREQ

```

0001 PROGRAM proPWM_FREQ
0002 VAR
0003     byIndex          : BYTE;                (* index in array of measured values *)
0004     adwMeasure       : ARRAY[0..cbyMaxIndex] OF DWORD;  (* measured values *)
0005
0006     fbDC541Pwm       : DC541_PWM;            (* instance of FB DC541_PWM *)
0007     bDC541Pwm_EN     : BOOL := TRUE;        (* enable PWM *)
0008     byDC541Pwm_SLOT  : BYTE := cbySlot;      (* slot number of DC541 *)
0009     bDC541Pwm_CH     : BYTE := 0;           (* channel number for PWM, same as in PLC configuration *)
0010     byDC541Pwm_PULSE : BYTE := 1;           (* relation TRUE signal *)
0011     byDC541Pwm_PAUSE : BYTE := 2;           (* relation FALSE signal *)
0012     wDC541Pwm_CYCLE  : WORD := 2000;        (* min duration of TRUE/FALSE signal *)
0013
0014     fbDC541Freq      : DC541_FREQ;          (* instance of FB DC541_FREQ *)
0015     byDC541Freq_SLOT : BYTE := cbySlot;      (* slot number of DC541 *)
0016     byDC541Freq_CH   : BYTE := 1;           (* channel number for FREQ, same as in PLC configuration *)
0017     bDC541Freq_EN_VISU : BOOL := TRUE;      (* enable control by visualization *)
0018     bDC541Freq_EN    : BOOL := TRUE;        (* enable measurement *)
0019     bDC541Freq_EN_0  : BOOL := TRUE;        (* measure all edges *)
0020     bDC541Freq_EN_1  : BOOL := TRUE;        (* measure all edges *)
0021     bDC541Freq_EN_FREQ : BOOL := FALSE;     (* no frequency and RPM measure *)
0022     wDC541Freq_PREC  : WORD := 0;           (* precision = 0 *)
0023 END_VAR
0024 VAR CONSTANT
0025     cbySlot          : BYTE := 1;            (* SLOT=1, first external coupler slot *)
0026     cbyMaxIndex      : BYTE := 24;          (* number of entries in array *)
0027 END_VAR
0028
0029
0030
0001 fbDC541Pwm(      EN      := bDC541Pwm_EN,          (* call DC541_PWM *)
0002                  SLOT    := byDC541Pwm_SLOT,
0003                  CH       := bDC541Pwm_CH,
0004                  PULSE    := byDC541Pwm_PULSE,
0005                  PAUSE    := byDC541Pwm_PAUSE,
0006                  CYCLE    := wDC541Pwm_CYCLE);
0007
0008 fbDC541Freq(      SLOT    := byDC541Freq_SLOT,      (* call DC541_FREQ *)
0009                  CH       := byDC541Freq_CH,
0010                  EN_VISU  := bDC541Freq_EN_VISU,
0011                  EN       := bDC541Freq_EN,
0012                  EN_0     := bDC541Freq_EN_0,
0013                  EN_1     := bDC541Freq_EN_1,
0014                  EN_FREQ  := bDC541Freq_EN_FREQ,
0015                  PREC     := wDC541Freq_PREC);
0016
0017 (* write the measured values in ring buffer abyMeasure *)
0018 IF fbDC541Freq.NEW THEN
0019     adwMeasure[byIndex] := fbDC541Freq.DUR;      (* new measure *)
0020     byIndex := byIndex + 1;                       (* write duration in array *)
0021     IF byIndex > cbyMaxIndex THEN byIndex := 0; END_IF; (* next entry *)
0022 END_IF; (* NEW *)
0023
0024
0025

```

功能块 DC541_GET_CFG, DC541_STATE 和 DC541_IO 包含在 DC541_AC500_V11.lib 库中，详细的描述见库文档。

功能块 TASK_INFO 包含在 SysInt_AC500_V1.0 库中，在相应的文档中详细描述。在 proPWM_FREQ 程序中调用脉宽调制功能以及测量和获取测量值。proPWM_FREQ 程序包含对功能块 DC541_PWM 和 DC541_FREQ 的调用以及测量值的获取。功能块 DC541_FREQ 被配置为测量每个沿变化的时间。

本例中程序可视化执行，显示所有的状态：

visTest_DC541

PWM - FREQ EXAMPLE

PLC_PRG cycle :	1605293	Duration [µs]	0	2000
Index number :	21		1	4000

DC541 Configuration

Slot	Cycle	Mode
1	100	Counter mode

Channel 0	PWM
Channel 1	Frequency
Channel 2	Input
Channel 3	Input
Channel 4	Input
Channel 5	Input
Channel 6	Input
Channel 7	Input

DC541 Frequency

Slot	Channel	New measure
1	1	Substitute

Enable En 0 En 1 En Freq

Precision 0

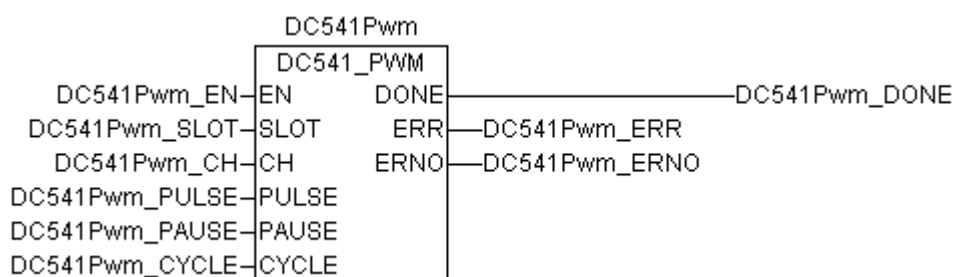
Duration	2000
Frequency	0.000
RPM	0.000

0	2000
1	4000
2	2000
3	4000
4	2000
5	4000
6	2000
7	4000
8	2000
9	4000
10	2000
11	4000
12	2000
13	4000
14	2000
15	4000
16	2000
17	4000
18	2000
19	4000
20	2000
21	2000
22	4000
23	2000
24	4000

功能块 DC541_FREQ 的输入 EN_VISU 是 TRUE。因此，在可视化时，功能块的输入可以使用按钮<Enable>、<En 0>、<En 1>和<En Freq>进行修改。

根据要测量的沿，测量值为 2000、4000 或 6000 µs。

4.3.2 DC541_PWM 功能块说明



功能块 DC541_PWM 输出一个占空比可调的脉冲信号。

功能块数据

适用的 PLC 运行系统:	V1.1.3 版本	注释:
适用的 DC541 固件:	V1.1 版本	
所属库:	DC541_AC500_V11.LIB	

功能块类型

功能块（带有历史数据）

参数

实例		DC541_PWM	功能描述
----	--	-----------	------

EN	输入	BOOL	启用功能块
SLOT	输入	BYTE	DC541 的插槽
CH	输入	BYTE	输出选择 C0...C7
PULSE	输入	BYTE	TRUE 信号的比率
PAUSE	输入	BYTE	FALSE 信号的比率
CYCLE	输入	WORD	最小的切换时间 [μ s]
DONE	输出	BOOL	功能块运行完毕
ERR	输出	BOOL	功能块故障信息
ERNO	输出	WORD	故障代码

描述

功能块 DC541_PWM 输出一个占空比可调的脉冲信号。高电平和低电平时间通过 8 位数字进行调整。

输入 CYCLE 设定了最小的切换时间,即如果输出通过 PWM 切换到 FALSE 或 TRUE, 输出保持这个状态的最小时间(CYCLE μ s)。

在输入 CYCLE 设定的最小时间必须小于 DC541 的循环时间。根据其配置, DC541 的循环时间可以是 50、100 或 200 μ s。可以使用 DC541_GET_CFG (输出 CYCLE)获得循环时间 (参见功能块 DC541_GET_CFG/CYCLE)。

举例:

PULSE	PAUSE	CYCLE	结果 (x=DC541 的循环次数)
DC541 的循环时间=50 μs			
1	2	500	10 x TRUE / 20 x FALSE / 10 x TRUE / 20 x FALSE / ... 即 500 μ s = TRUE, 1000 μ s = FALSE
4	8	500	10 x TRUE / 20 x FALSE / 10 x TRUE / 20 x FALSE / ... 即 500 μ s = TRUE and 1000 μ s = FALSE (如例 1,即比率为 1 : 2)
3	2	3000	90 x TRUE / 60 x FALSE / 90 x TRUE / 60 x FALSE / ... 即 4500 μ s = TRUE, 3000 μ s = FALSE
DC541 的循环时间=100 μs			
1	2	500	5 x TRUE / 10 x FALSE / 5 x TRUE / 10 x FALSE / ... 即 500 μ s = TRUE, 1000 μ s = FALSE
4	8	500	5 x TRUE / 10 x FALSE / 5 x TRUE / 10 x FALSE / ... 即 500 μ s = TRUE and 1000 μ s = FALSE (如例 1,即比率为 1 : 2)
3	2	3000	45 x TRUE / 30 x FALSE / 45 x TRUE / 30 x FALSE / ... 即 4500 μ s = TRUE, 3000 μ s = FALSE
DC541 的循环时间=200 μs			
1	2	500	3 x TRUE / 6 x FALSE / 3 x TRUE / 6 x FALSE / ... 即 600 μ s = TRUE, 1200 μ s = FALSE

4	8	500	3 x TRUE / 6 x FALSE / 3 x TRUE / 6 x FALSE / ... 即 600 μ s = TRUE , 1200 μ s = FALSE (如例 1, 即比率为 1 : 2)
3	2	3000	22 x TRUE / 15 x FALSE / 23 x TRUE / 15 x FALSE / ... 即首先 4400 μ s = TRUE, 3000 μ s = FALSE, 然后 4600 μ s = TRUE , 3000 μ s = FALSE

设备 DC541 必须被配置为计数设备（计数器模式）。

EN BOOL (使能)

输入 EN 必须持续设置为 TRUE。

当第一次调用功能块时，检查输入 SLOT 的有效性和合理性，检查设备的“计数模式”配置是否正确，检查通道 CH 的配置是否为“PWM”。若非如此，处理将中断，相应的错误信息显示在输出端 ERR/ERNO。只要输入 EN 为 TRUE，功能块的输出就被更新。

SLOT BYTE (插槽)

输入 SLOT 用于选择 DC541 的插槽（模块号码）。

插槽从右到左连续编号。插槽 1 是在 CPU 左侧的第一个插槽

CH BYTE (通道)

输入 CH 用于选择输出脉冲的输出。

PULSE BYTE (脉冲)

输入 PULSE 用于设定 TRUE 信号的比率（高电平时间比率）。

PAUSE BYTE (暂停)

输入 PAUSE 用于设定 FALSE 信号的比率（低电平时间比率）。

CYCLE WORD (循环时间)

输入 CYCLE 用于设定输出的最小切换时间[μ s]。输出最少保持在 CYCLE 设定时间的 TRUE 或 FALSE 状态。

在输入 CYCLE 设定的最小时间必须小于 DC541 的循环时间。根据其配置，DC541 的循环时间可以是 50、100 或 200 μ s。可以使用 DC541_GET_CFG (输出 CYCLE)获得循环时间 (参见功能块 DC541_GET_CFG/CYCLE)。

DONE BOOL (完成)

输出 DONE 表示功能块的处理状态。在过程完成或被终止后（由于出错），DONE 被置为一个周期的 TRUE。这一输出要与输出 ERR 一起综合判断。如果 ERR 为 TRUE，说明发生错误。此时，在输出 ERNO 中可以读出故障代码。

ERR BOOL (故障)

输出 ERR 表示在功能块运行过程中是否有错误发生。这一输出要与输出 DONE 一起综合判断。如果 DONE 为 TRUE，且 ERR 为 TRUE，说明出错。此时，在输出 ERNO 中可以读出故障代码。

ERNO WORD (故障代码)

当一无效的数值应用到输入端时，或者在处理过程中出错时，输出 ERNO 将提供一错误指示。ERNO 要与输出 DONE 和 ERR 一起综合判断。ERNO 的输出数值仅在 DONE 为 TRUE

以及 ERR 为 TRUE 时有效。在 ERNO 中的错误信息输出编码的解释，请参见“功能块库的错误信息表格”。

IL 中的功能调用

```
CAL      DC541Pwm(  
          EN          := DC541Pwm_EN,  
          SLOT        := DC541Pwm_SLOT,  
          CH           := DC541Pwm_CH,  
          PULSE        := DC541Pwm_PULSE,  
          PAUSE        := DC541Pwm_PAUSE,  
          CYCLE        := DC541Pwm_CYCLE)  
  
LD        DC541Pwm.DONE  
ST        DC541Pwm_DONE  
  
LD        DC541Pwm.ERR  
ST        DC541Pwm_ERR  
  
LD        DC541Pwm.ERNO  
ST        DC541Pwm_ERNO
```

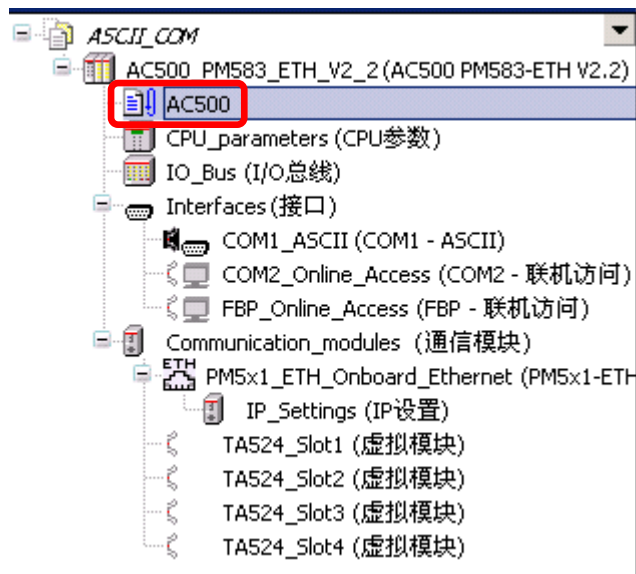
注：在 IL 中，功能调用必须写在一行中。

ST 中的功能调用

```
DC541Pwm(EN          := DC541Pwm_EN,  
          SLOT        := DC541Pwm_SLOT,  
          CH           := DC541Pwm_CH,  
          PULSE        := DC541Pwm_PULSE,  
          PAUSE        := DC541Pwm_PAUSE,  
          CYCLE        := DC541Pwm_CYCLE);  
  
DC541Pwm_DONE      := DC541Pwm.DONE;  
DC541Pwm_ERR       := DC541Pwm.ERR;  
DC541Pwm_ERNO      := DC541Pwm.ERNO;
```

5 程序下载

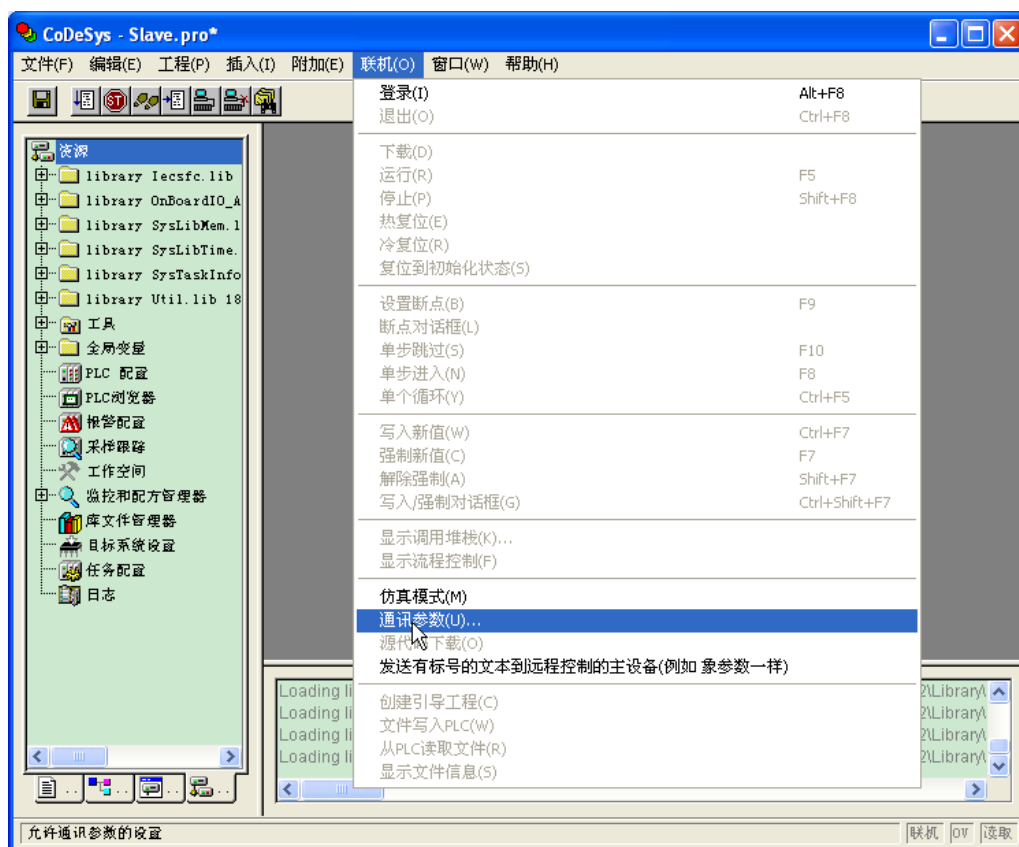
双击设备树中程序图标，点击确定，进入 CodeSys 编程界面



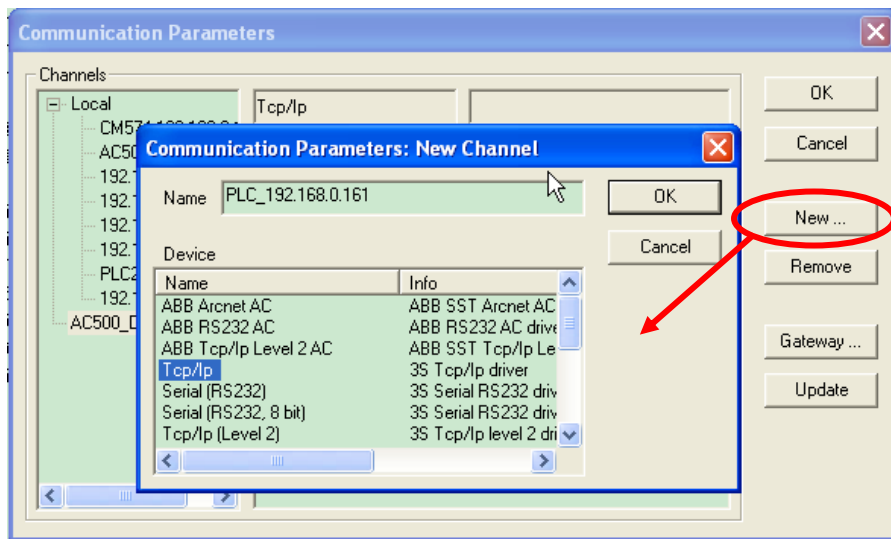
在编程界面中，设置程序下载的通讯接口，将程序下载即可。操作步骤如下

- 选择或创建网关

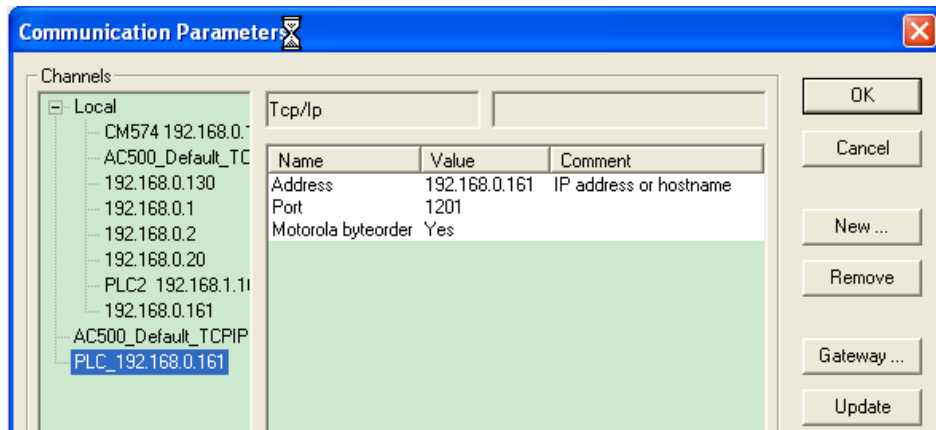
点击“联机”→“通讯参数”



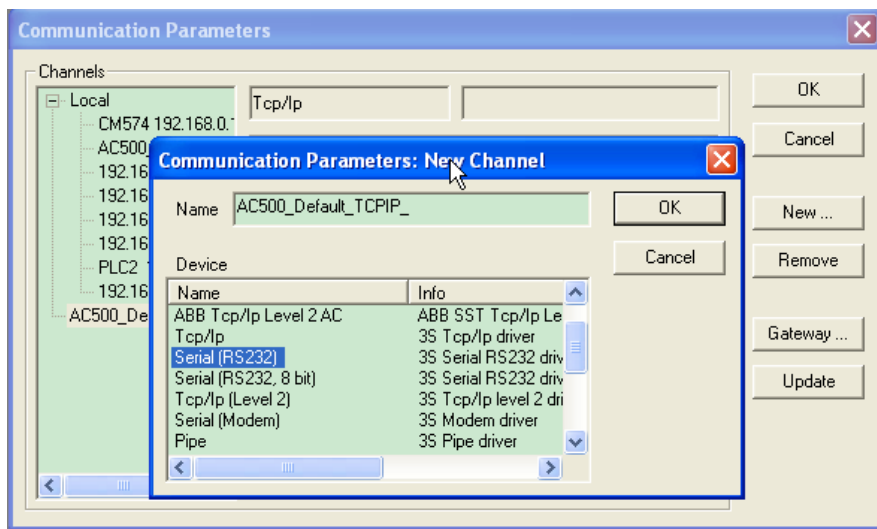
在弹出窗口中新建一个 TCP/IP 类型的网关

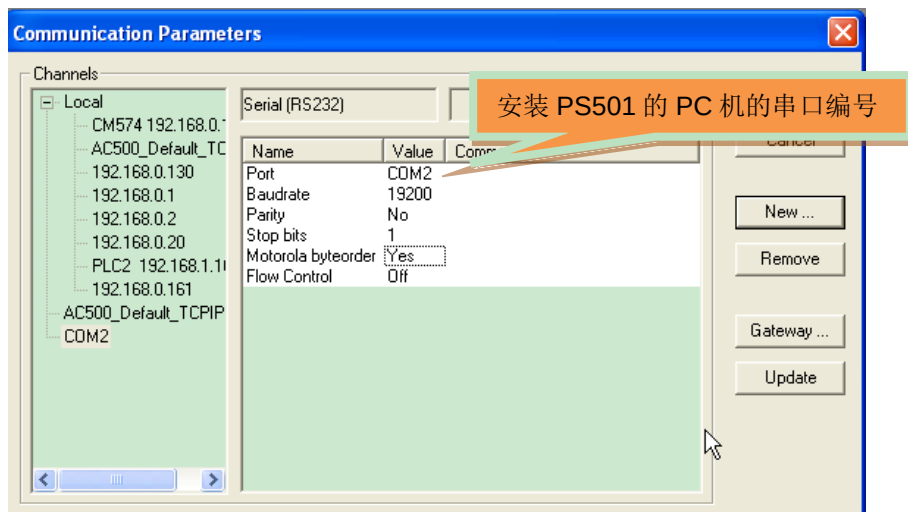


修改网关参数如下；



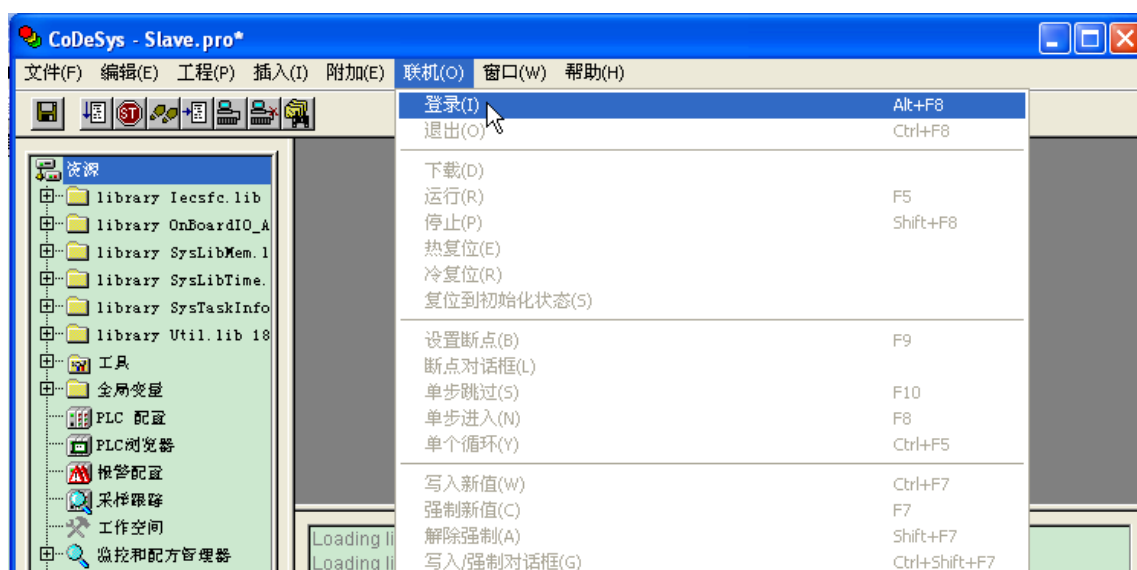
当然，也可以通过 COM2 口下载程序，此时的网关设置如下：



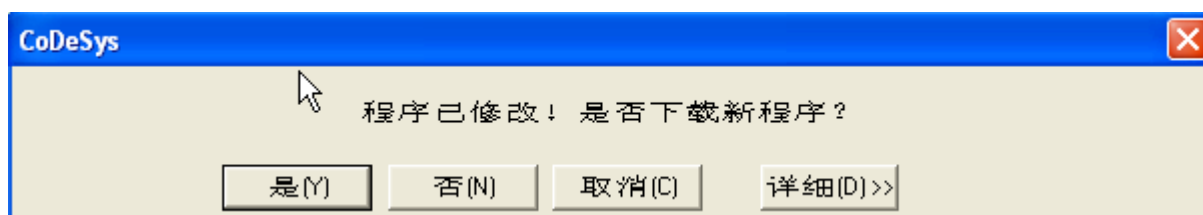


- 下载程序

点击“联机”→“登录”

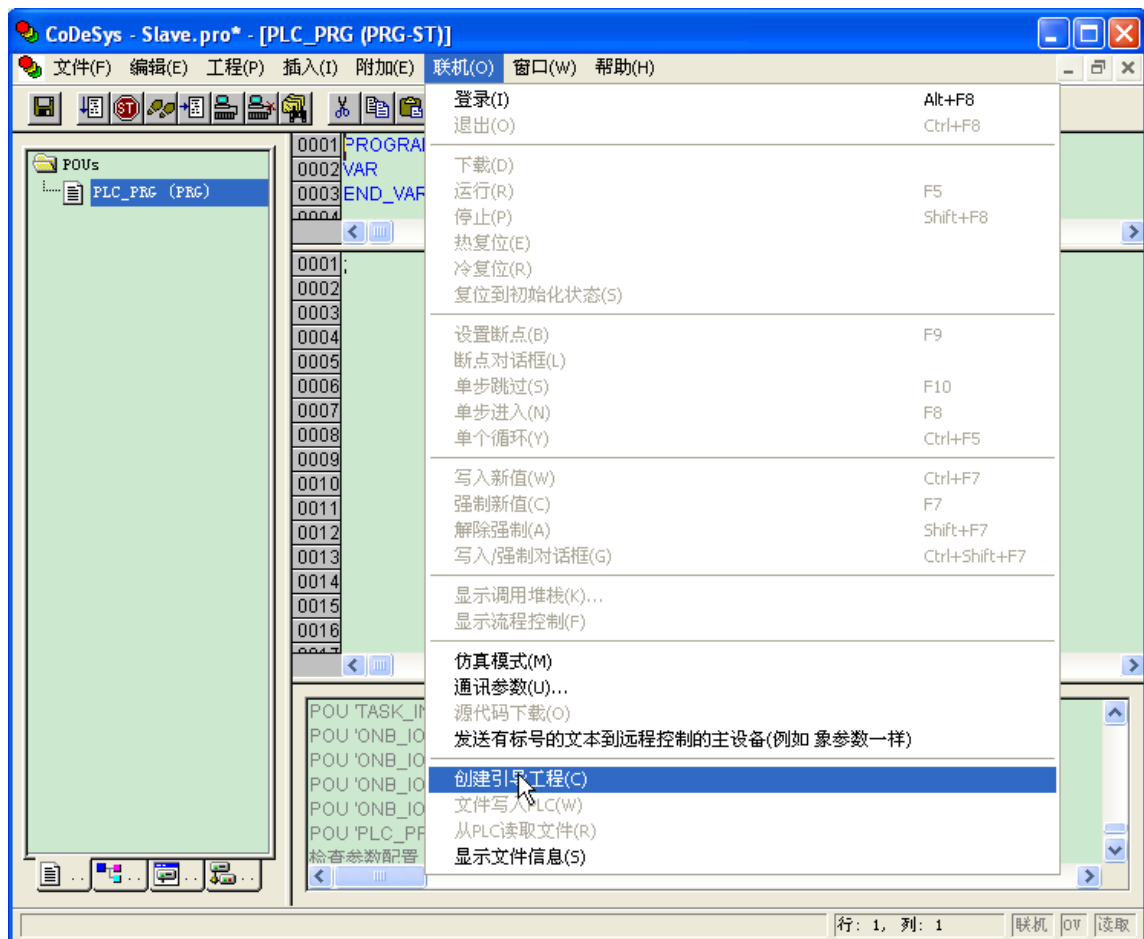


在弹出的对话框中，点击“是”按钮，将程序下载到 PLC



- 创建引导工程

点击“联机”→“创建引导工程”，将程序写入到 PLC 的程序存储器中

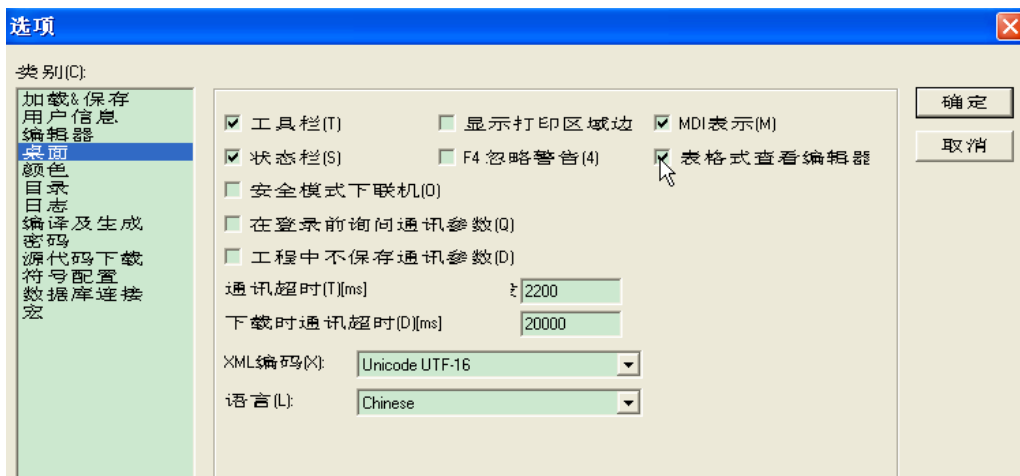


程序下载接口设置可参见帮助文档“目标系统，系统技术，AC500 CPU 系统技术，编程和测试”一节

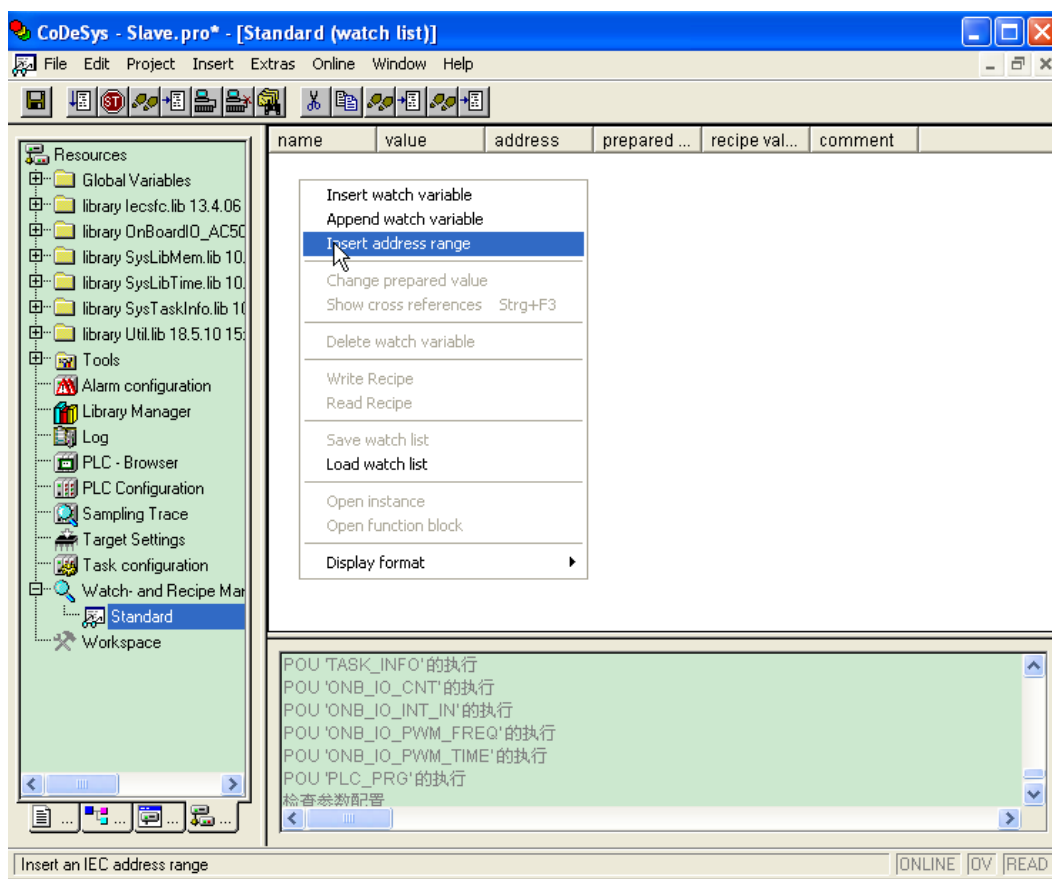
6 程序监控

在 Modbus 从站中，不需要创建程序，但可以添加变量表用于监控运行状态。

首先，点击菜单栏“工程”→“选项”，在弹出的对话框选择“表格式查看编辑器”



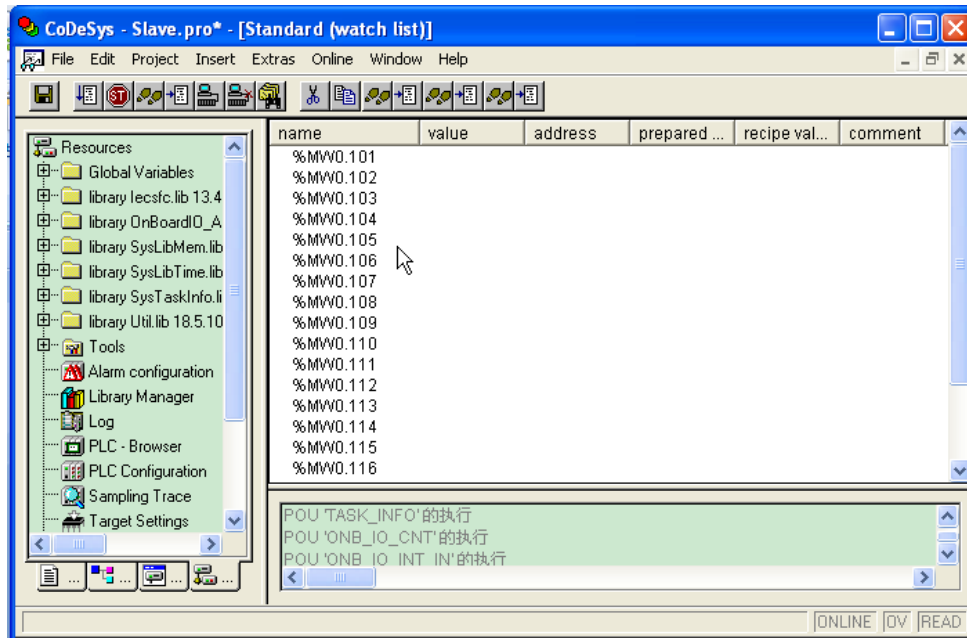
在“资源”选项卡中，双击“配方和监控管理器”，在右侧窗口点击右键，选择“插入地址范围”。



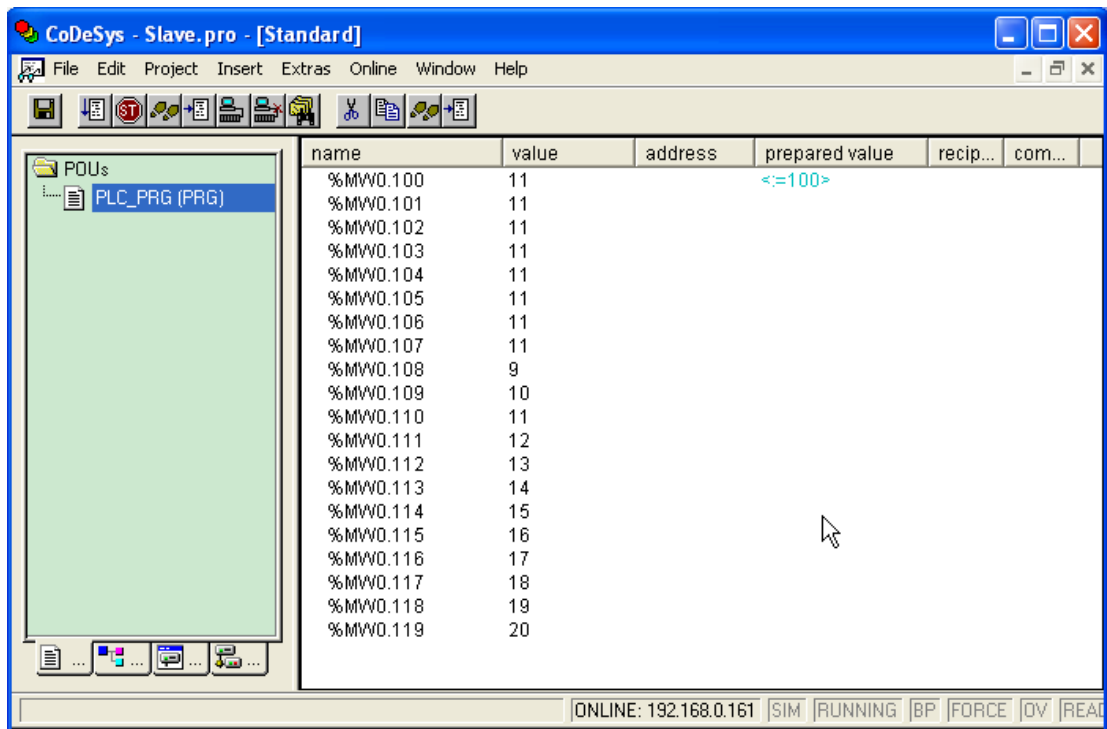
在弹出的对话框中，可以选择要监控的变量起始地址和长度



创建的变量表如下：



在联机状态下，可以查看并修改变量数值



联系我们

北京 **ABB** 电气传动系统有限公司

地址：北京市朝阳区酒仙桥北路甲 10 号 D 区 1 号

邮编：100015

总机：(86-10)58217788

传真：(86-10)58217518

服务热线：400 810 8885

E-Mail: plc.service@cn.abb.com

网址：<http://www.abb.com.cn/plc>