

SIEMENS

SIMATIC

S7-300

《SM331; AI 8x12 位使用入门》 第 1 部分: 4 - 20 mA

入门指南

前言	1
要求	2
引言	3
实例站的机械设置	4
电气连接	5
SIMATIC 管理器的组态	6
测试用户程序	7
诊断中断	8
硬件中断	9
附录	A

安全技术提示

为了您的人身安全以及避免财产损失，必须注意本手册中的提示。人身安全的提示用一个警告三角表示，仅与财产损失有关的提示不带警告三角。警告提示根据危险等级由高到低如下表示。



危险

表示如果不采取相应的小心措施，**将会**导致死亡或者严重的人身伤害。



警告

表示如果不采取相应的小心措施，**可能**导致死亡或者严重的人身伤害。



小心

带有警告三角，表示如果不采取相应的小心措施，可能导致轻微的人身伤害。

小心

不带警告三角，表示如果不采取相应的小心措施，可能导致财产损失。

注意

表示如果不注意相应的提示，可能会出现不希望的结果或状态。

当出现多个危险等级的情况下，每次总是使用最高等级的警告提示。如果在某个警告提示中带有警告可能导致人身伤害的警告三角，则可能在该警告提示中另外还附带有可能导致财产损失的警告。

合格的专业人员

仅允许安装和驱动与本文件相关的附属设备或系统。设备或系统的调试和运行仅允许由**合格的专业人员**进行。本文件安全技术提示中的合格专业人员是指根据安全技术标准具有从事进行设备、系统和电路的运行，接地和标识资格的人员。

按规定使用

请注意下列说明：



警告

设备仅允许用在目录和技术说明中规定的使用情况下，并且仅允许使用西门子股份有限公司推荐的或指定的其他制造商生产的设备和部件。设备的正常和安全运行必须依赖于恰当的运输，合适的存储、安放和安装以及小心的操作和维修。

商标

所有带有标记符号 © 的都是西门子股份有限公司的注册商标。标签中的其他符号可能是一些其他商标，这是出于保护所有者权利的 目地由第三方使用而特别标示的。

责任免除

我们已对印刷品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，因此我们不保证印刷品中所述内容与硬件和软件完全一致。印刷品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。

目录

1	前言	5
1.1	常规	5
2	要求	7
2.1	基本要求	7
3	引言	9
3.1	应用实例	9
4	实例站的机械设置	11
4.1	安装实例站	11
4.2	模拟模块组件的安装	13
4.2.1	常规	13
4.2.2	SM331 的组件	14
4.2.3	模拟模块的特性	15
4.2.4	量程卡	16
4.2.5	安装 SM331 模块	18
5	电气连接	19
5.1	概述	19
5.2	为电源模块和 CPU 接线	20
5.3	为模拟模块接线	22
5.3.1	要求	22
5.3.2	电流变送器接线 — 原理	22
5.3.3	为模拟模块接线	23
5.3.4	测试	26
6	SIMATIC 管理器的组态	27
6.1	创建新的 STEP7 项目	27
6.1.1	创建新项目	27
6.1.2	CPU 选择	29
6.1.3	定义基本用户程序	30
6.1.4	分配项目名称	31
6.1.5	创建了结果 S7 项目	32

6.2	硬件组态	33
6.2.1	创建硬件组态	33
6.2.2	添加 SIMATIC 组件	34
6.2.3	组态模拟模块	36
6.2.4	测试	39
6.3	STEP7 用户程序	43
6.3.1	用户程序的任务	43
6.3.2	创建用户程序	44
7	测试用户程序	51
7.1	下载系统数据和用户程序	51
7.2	传感器值的可视化	53
7.3	模拟值表达式	57
8	诊断中断	59
8.1	从 PG 中读取诊断信息	59
8.2	常规诊断	61
8.3	与通道相关的诊断中断	62
8.3.1	存在五种与通道相关的诊断中断	62
8.3.2	组态/编程错误	62
8.3.3	共模错误	62
8.3.4	断线	63
8.3.5	下溢	63
8.3.6	上溢	64
9	硬件中断	65
9.1	硬件中断	65
A	附录	67
A.1	用户程序的源	67
	索引	73

前言

1.1 常规

本使用入门的用途

本使用入门完整地概述了模拟模块 **SM331** 的调试。它可帮助您安装和配置 **4 到 20 mA** 传感器的硬件以及使用 **SIMATIC S7** 管理器进行组态。

本使用入门的目标读者是在对自动化系统进行组态、调试和维修方面仅有基本经验的初学者。

内容简介

逐步介绍了从安装模块到在 **STEP7** 用户程序中存储模拟值的步骤，并基于实例详细地进行说明。以下各节中将向您介绍：

- 问题分析
- 实例站的机械设置
- 实例站的电气连接
- 使用 **SIMATIC** 管理器组态硬件
- 使用 **STEP7** 创建小用户程序，用于将读取的模拟值存储在数据块中
- 触发和解释诊断和硬件中断

要求

2.1 基本要求

需要的基本知识

理解使用入门指南不需要具有自动化技术领域的专业知识。因为模拟模块的组态要使用软件 STEP7 来完成，所以精通 STEP7 有助于您的工作。

有关 STEP7 的更多信息可以在 STEP7 随附的电子手册中找到。

还需要知道如何使用安装了 Windows 95/98/2000/NT 或 XP 的计算机或类似于 PC 的设备（例如，编程设备）。

需要的硬件和软件

模拟模块的交货范围包含两部分：

- 模块本身
- 前连接器，它使您可以方便地连接电源和数据电缆

模拟模块组件

数量	项目	订货号
1	SM 331, 电隔离的 8 AI, 报警诊断	6ES7331-7KF02-0AB0
1	带有弹簧触点的 20 针前连接器	6ES7392-1BJ00-0AA0

该实例所需的常规 SIMATIC 组件如下：

实例站的 SIMATIC 组件

数量	项目	订货号
1	PS 307 电源 AC 120/230V, DC 24V, 5A	6ES7307-1EA00-0AA0
1	CPU 315-2 DP	6ES7315-2AG10-0AB0
1	微型存储卡, NFLASH, 4 MB	6ES7953-8LM00-0AA0
1	SIMATIC S7-300, 导轨长度=530MM	6ES7390-1AF30-0AA0
1	带有 MPI 接口和 MPI 电缆的编程设备 (PD) 带有相应接口卡的 PC	取决于组态

软件 STEP7

数量	项目	订货号
1	STEP7 软件 V5.2 或更高版本（已安装在编程设备上）。	6ES7810-4CC06-0YX0

2.1 基本要求

以下电流变送器可用于采集模拟信号：

电流变送器

数量	项目	订货号
1	2 线制电流变送器	取决于制造商
1	4 线制电流变送器	取决于制造商

说明

本“使用入门”仅介绍采用 2 线制或 4 线制模型的 4 到 20 mA 电流变送器的应用。如果要使用其它变送器，则需要对 SM331 进行不同的接线和组态。

常规工具和材料：

数量	项目	订货号
多种	M6 螺栓和螺母（长度取决于安装位置）	通常可用
1	刀口宽度为 3.5 mm 的螺丝刀	通常可用
1	刀口宽度为 4.5 mm 的螺丝刀	通常可用
1	侧铣刀和剥线工具	通常可用
1	线端套管卷边工具	通常可用
X m	用于将横截面为 10 mm ² 的装配导轨和孔为 6.5 mm 的环形端子接地的电缆，其长度取决于实际需要	通常可用
X m	带装线端套且直径为 1 mm ² 的软线，3 种不同颜色的 A 型线（蓝色、红色和绿色）	通常可用
X m	带有保护触点插座的 3 线制电源线 (AC 230/120V)，其长度取决于实际需要	通常可用
1	校准设备（用于进行调试的测量仪表，可测量电流和提供电流）	取决于制造商

引言

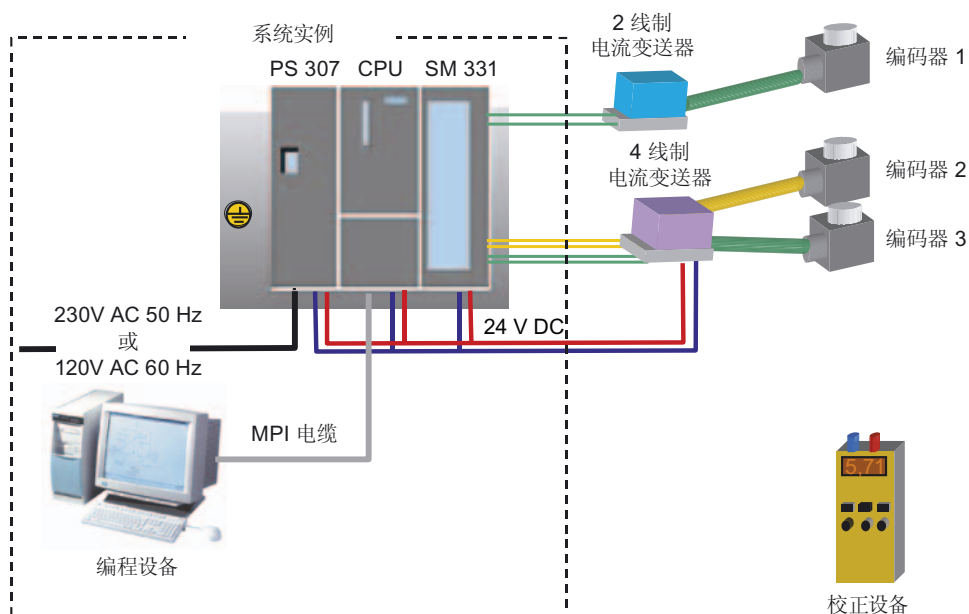
3.1 应用实例

概述

您要将三个模拟输入连接到站。其中一个模拟输入应具有 2 线制电流变送器，另外两个应共享一个 4 线制电流变送器。

您需要故障诊断功能，并希望两个传感器可以触发硬件中断。

您可使用模拟输入模块 SM331, AI8x12 位（订货号 6ES7 331-7KF02-0AB0）。该模块具有诊断和硬件中断功能，且最多可处理 8 个模拟输入。该模块具有诊断和硬件中断功能，且最多可处理 8 个模拟输入（例如 4 到 20 mA；PT 100；热电偶）。



图片 3-1 实例站组件

以下各节中将向您介绍：

- 实例站的机械设置
 - S7-300 模块的常规安装说明
 - 两种选定测量变送器类型的 SM331 的组态
- 实例站的电气连接
 - 为电源模块和 CPU 接线
 - 为模拟模块接线
 - 两种测量变送器类型的标准针脚分配
 - 为未使用的输入接线
- 组态 SIMATIC 管理器
 - 使用项目向导
 - 完成自动生成的硬件组态
 - 集成提供的用户程序源
- 用户程序测试
 - 译解读取值
 - 将测量值转换为可读取的模拟数据
- 利用 SM331 模块的诊断功能
 - 触发诊断中断
 - 评估诊断：
- 硬件中断的应用
 - 硬件中断的组态
 - 硬件中断的组态和分析

实例站的机械设置

4.1 安装实例站

概述

实例站的设置分为两步。首先介绍电源和 CPU 的设置。熟悉模拟模块 SM331 以后，再介绍如何安装该模块。


要求

您需要对常规 SIMATIC S7-300 组件进行基本设置，然后才能使用模拟输入模块 SM331。

按照从左到右的顺序进行安装：

- 电源 PS307
- CPU 315-2DP
- SM331

说明（无 SM331）

步骤	图形控制器	说明
1		用螺钉固定在装配导轨上（螺钉规格：M6），以便在导轨的上下至少各保留 40 mm 的空间。
2		在将其安装到接地钢板或钢制接地设备安装板时，请确保装配导轨与安装表面之间具有低阻抗连接。
		将装配导轨与保护性导体相连。为此，装配导轨上提供了一个 M6 保护性导体螺钉。

步骤	图形控制器	说明
3		装配电源： <ul style="list-style-type: none"> • 将电源悬挂到导轨的顶端 • 拧紧螺钉，将电源固定在导轨下面
4		将总线连接器（SM331 随附的）连接到 CPU 背面左侧的连接器
5		安装 CPU： <ul style="list-style-type: none"> • 将 CPU 悬挂到导轨的顶端 • 一直向左推动 CPU，直到到达电源 • 放下 CPU • 拧紧螺钉，将 CPU 固定在导轨下面

4.2 模拟模块组件的安装

4.2.1 常规

概述

在实际安装 SM331 之前，必须为该模块安装前连接器且按需求设置输入的测量模式。

本节将向您介绍：

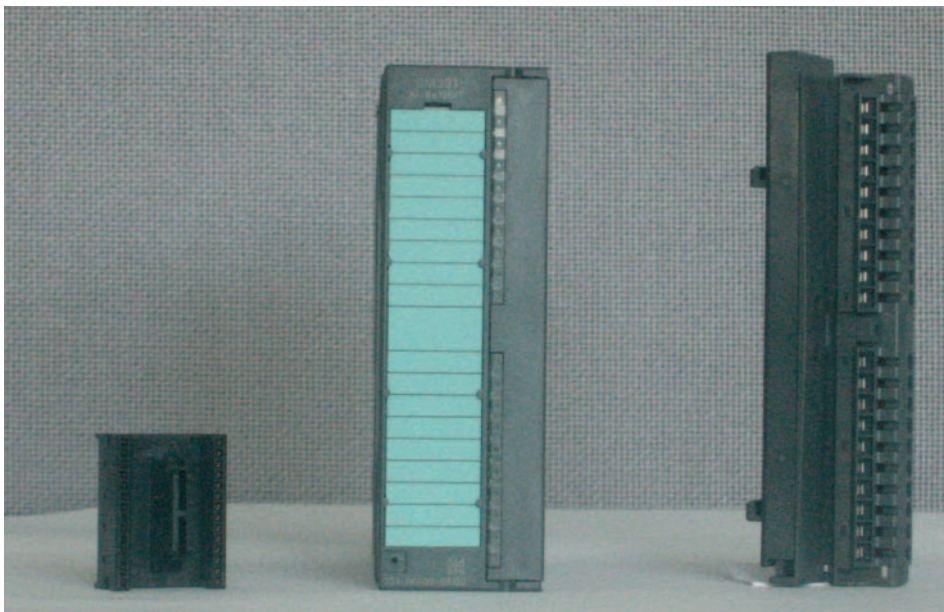
- 您需要的组件
- 模拟输入模块的属性
- 什么是量程卡以及如何对其进行组态
- 安装已组态的模块

4.2.2 SM331 的组件

概述

功能模拟模块包括以下组件：

- 模块 SM331（在我们的实例中为 6ES7331-7KF02-0AB0）
- 20 针前连接器。存在两种不同类型的前连接器：
 - 带有弹簧触点（订货号 6ES7392-1BJ00-0AA0）
 - 带有螺钉触点（订货号 6ES7392-1AJ00-0AA0）



图片 4-1 SM331 的组件

SM331 的交付范围

组件
模块
标签条
总线连接器
2 个电缆夹（图片中未显示），用于固定外部接线

4.2.3 模拟模块的特性

特性

- 4 个通道组中 8 个输入（每个组具有两个同类型的输入）
- 每个通道组的测量精度均可调整
- 每个通道组的用户定义测量模式：
 - 电压
 - 电流
 - 电阻
 - 温度
- 可编程诊断中断
- 带有限制报警的两个通道（仅通道 0 和通道 2 可组态）
- 与背板总线电隔离
- 与负载电压电隔离（例外：至少将一个模块设置到位置 D）

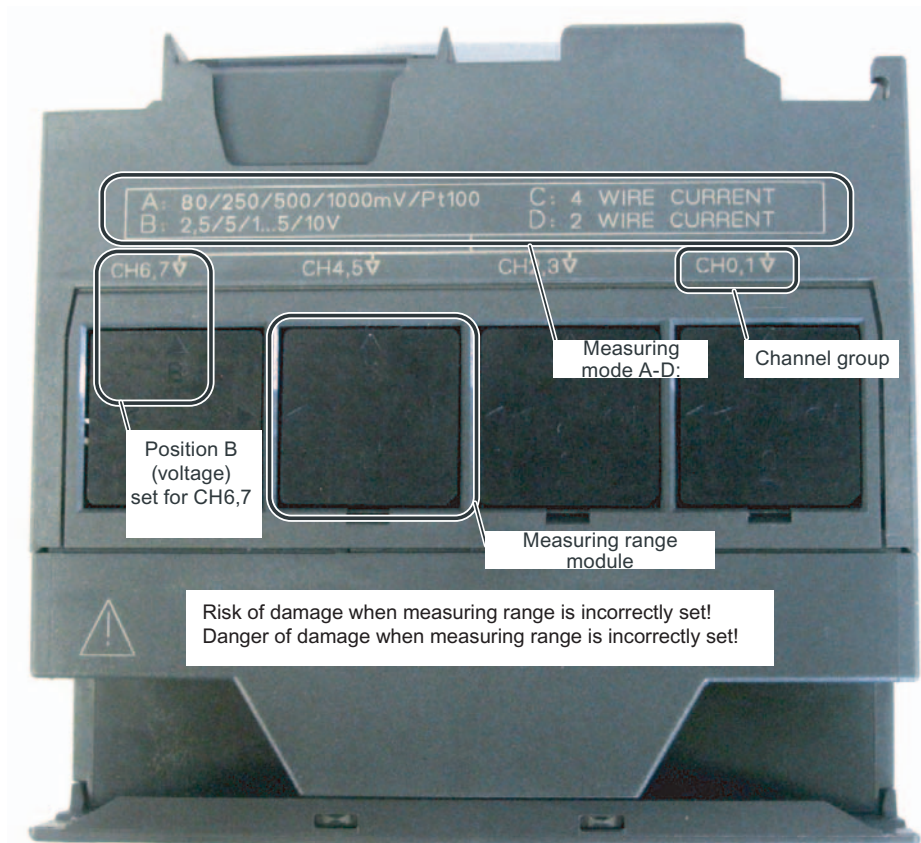
该模块是为最常用的应用程序分配的通用模拟模块。

应使用量程卡在该模块上直接设置需要的测量模式。

4.2.4 量程卡

端子

模块 SM331 具有 4 个量程卡（每个通道组具有一个）。可以将量程卡设置到 4 个不同的位置（A、B、C 或 D）。



图片 4-2 4 个量程卡的默认设置为 B（电压）

量程卡的位置

由位置确定连接到各个通道组的变送器。



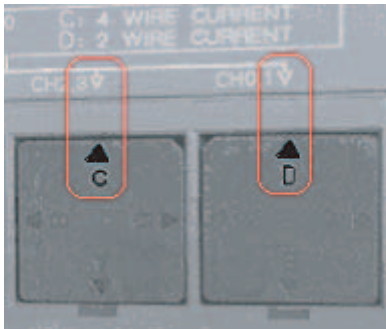
位置	测量类型
A	热电偶/电阻测量
B	电压（出厂设置）
C	电流（4 线制变送器）
D	电流（2 线制变送器）

在我们的实例中，带有 4 到 20 mA 2 线制变送器的传感器将连接到输入 0 上的通道组 1。

4 线制变送器将连接到输入 2 和 3 上的通道组 2。

因此，第一个量程卡应具有位置 D，而第二个量程卡应具有位置 C。

定位量程卡

步骤	图形控制器	说明
1		使用螺丝刀卸下两个量程卡
2		将量程卡转到要求的位置
3		将量程卡插回模块 在我们的实例中，该模块应具有以下位置： CH0, 1: D CH2, 3: C

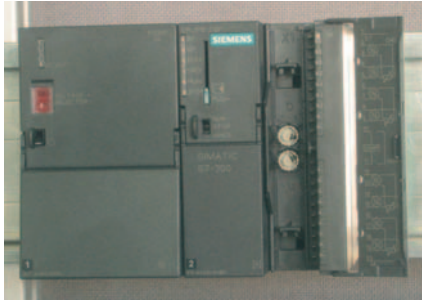

说明

当使用 2 线制变送器时，模块（至少一个量程卡设置到位置 D）中所有通道都将失去与负载电压的电隔离

4.2.5 安装 SM331 模块

请按以下步骤进行操作

相应地准备好模拟模块后，也将其安装到导轨上。

步骤	图形控制器	说明
1		<p>安装 SM331:</p> <ul style="list-style-type: none">• 将 SM311 悬挂到导轨的顶端• 一直向左推动模块，直到达到 CPU• 放下模块• 拧紧螺钉，将模块固定在导轨下面
2		<p>安装前连接器:</p> <ul style="list-style-type: none">• 按下前端子块的上释放按钮• 将前连接器插入模块直到其卡入到位

现在已机械安装实例站。

电气连接

5.1 概述

概述

本章将介绍如何将实例站的各种备件从电源连接到模拟模块。



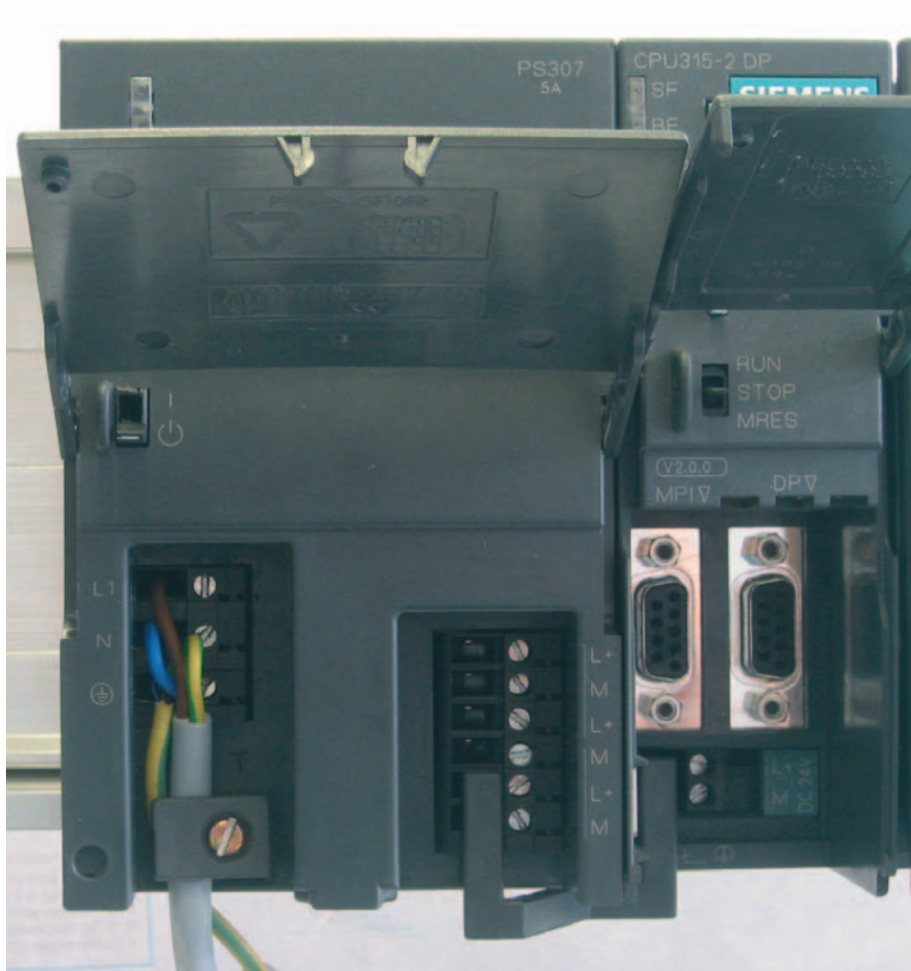
警告

如果接通电源 PS307 或将电源线连接到线路，可能会遭到电击。

在开始为 S7-300 接线之前，请始终关闭电源。


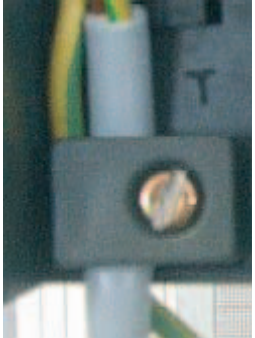
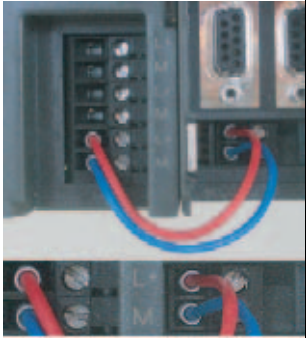
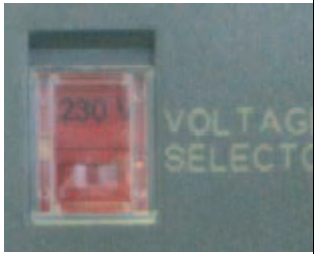
5.2 为电源模块和 CPU 接线

概述



图片 5-1 为电源模块和 CPU 接线

实例站需要一个电源。按照以下步骤进行接线：

步骤	图形控制器	说明
1		打开电源模块和 CPU 的前面板盖。
2		拧松电源上的电缆夹。
3		剥除电源线上的绝缘材料，连接电缆端套管（对于铰链导体）并将其连接到电源。
4		拧紧电缆夹的夹子。
5		在电源和 CPU 之间插入两根连接电缆并拧紧。
6		检验选择器开关的设置与电源电压是否匹配。 电源模块的默认线路电压设置为 230 VAC。要更改此设置，请按以下步骤进行操作：使用螺丝刀卸下保护帽，设置选择器开关使其与线路电压匹配，然后重新插入保护帽。

5.3 为模拟模块接线

5.3.1 要求

常规

模拟测量变送器的接线取决于其类型而不是 SM331 模块。

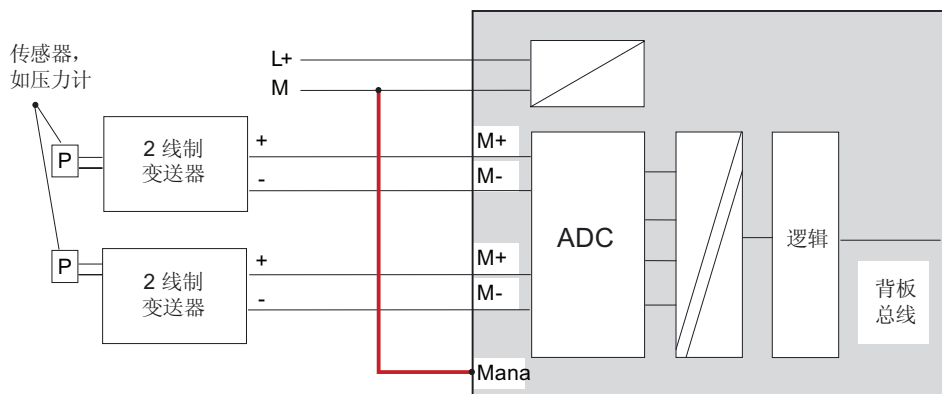
5.3.2 电流变送器接线 — 原理

选项

您必须根据使用的电流变送器修改电源的接线。我们将对 2 线制电流变送器和 4 线制电流变送器的接线进行区分。

2 线制电流变送器的接线原理

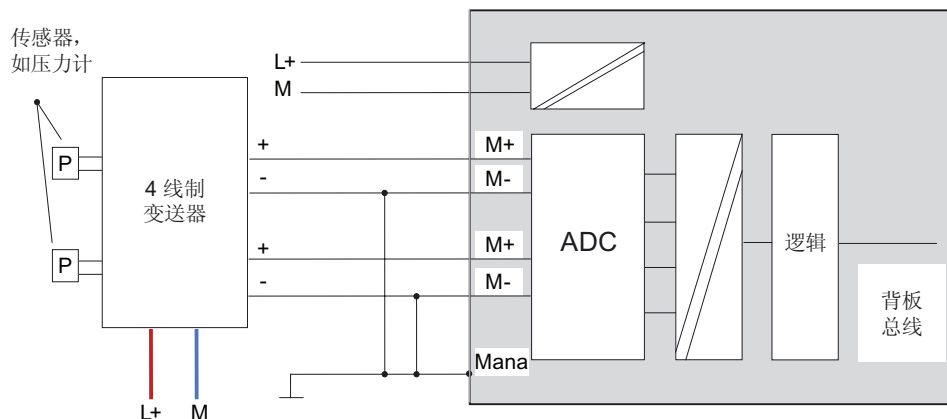
这种类型变送器由模拟输入模块供电。



图片 5-2 接线：2 线制电流变送器

4 线制电流变送器的接线原理

与 2 线制变送器不同，该变送器自身具有电源。



图片 5-3 接线：4 线制电流变送器

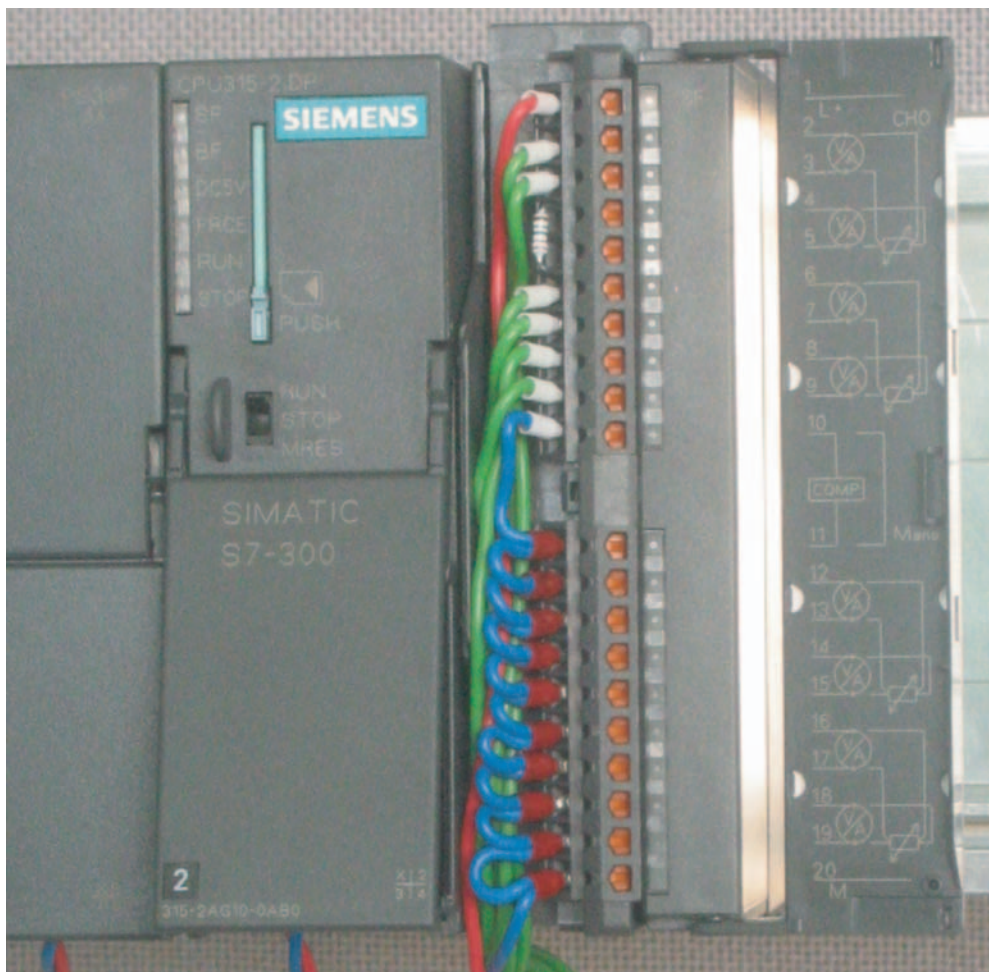
5.3.3 为模拟模块接线

任务

模拟模块的接线包括以下任务：

- 连接电源（红色电缆）
- 连接 2 线制电流变送器（绿色电缆）
- 不使用的通道用电阻器端接
- 连接 4 线制电流变送器（绿色电缆）
- 连接 4 线制电流变送器（绿色电缆）
- 接地并短路其它不使用的通道（蓝色电缆）

SM331 前连接器接线



图片 5-4 SM331 前连接器接线







注意

可能会损坏模块！

如果将有故障的 4 线制电流变送器连接到输入，而该输入已组态为用于 2 线制变送器，则可能会损坏模块！

下面将逐步介绍需要的接线任务：

请按以下步骤进行操作

步骤	图形控制器	连接	注释
1		打开 SM331 的前门	连接图印刷在前盖上
2		剥除电缆一端 6 mm 的绝缘材料,插入前连接器。将电缆端套管连接到这些线端。	
3		按照以下步骤为前连接器接线： 端子 1: L+	模块的电源
4		端子 2: M+ 传感器 1 端子 3: M- 传感器 1	2 线制电流变送器的标准接线
5		连接端子 4 和 5 至 1.5 到 3.3 kΩ 的电阻器	为了维护通道组 0 的诊断功能,必须将第二个不使用的输入连接到电阻器
6		端子 6: M+ 传感器 2 端子 7: M- 传感器 2	4 线制电流变送器的标准接线
7		端子 8: M+ 传感器 3 端子 9: M- 传感器 3	
8		将端子 10 (Comp) 和端子 11 (M _{ana}) 连接到 M 使端子 12 对 19 短路并与 M _{ana} 连接 端子 20: M	对于测量电流, 不使用 comp 强制用于 2 线制电流变送器应通过 M _{ana} 将未使用的通道组短路, 以获得最大干扰阻抗

5.3.4 测试

请按以下步骤进行操作

如果要测试接线，您现在就可以接通电源。

请记住将 CPU 设置为 STOP（请参阅红色圆圈）



图片 5-5 已成功接线，CPU 处于 STOP 位置

如果红色 LED 亮起，则表明存在接线错误。请验证接线情况。

SIMATIC 管理器的组态

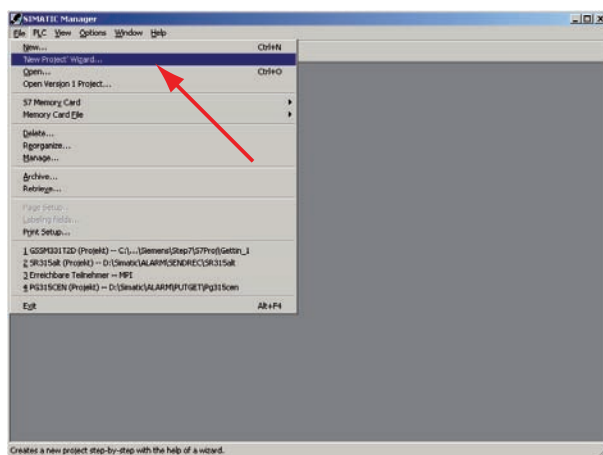
6.1 创建新的 STEP7 项目

6.1.1 创建新项目

“New Project...”（新建项目...）向导

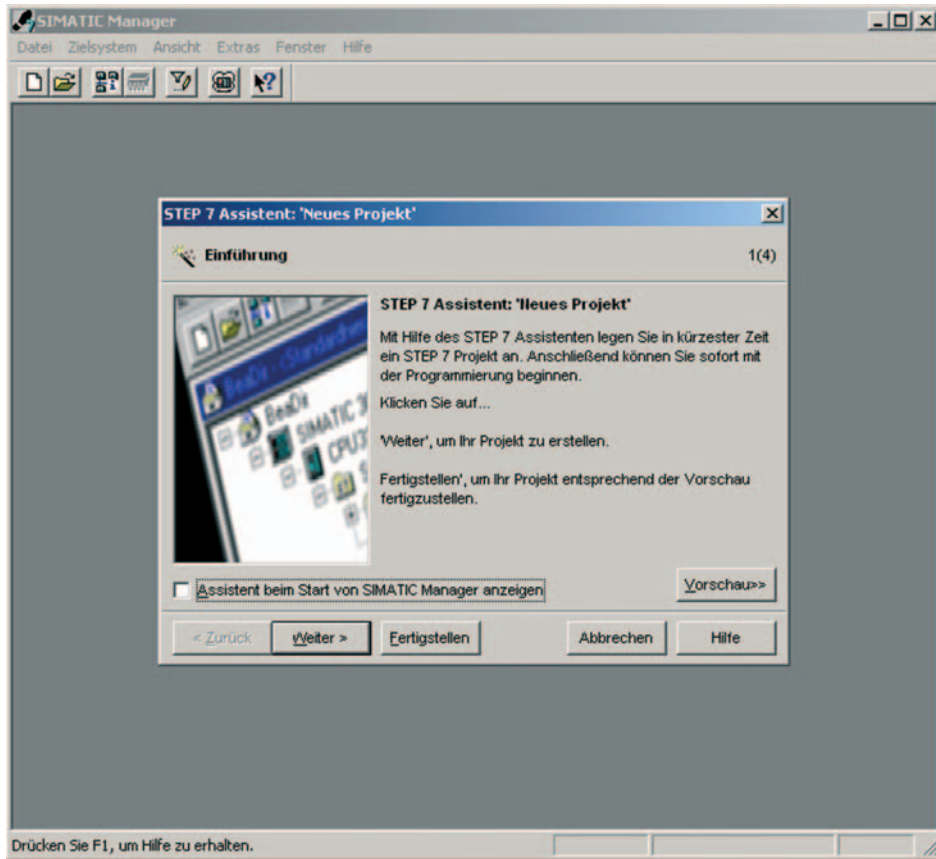
使用 STEP7 V5.2 或更高版本对新 CPU 315-2 DP 进行组态。

单击 Windows 桌面上的“SIMATIC 管理器”图标启动 SIMATIC 管理器，并使用 STEP7 “New Project”（新建项目）向导创建新项目。



图片 6-1 启动“New Project...”（新建项目...）向导

将显示一个项目向导介绍窗口。该向导将引导您完成创建新项目的步骤。



图片 6-2 向导“New Project”（新建项目），启动

创建过程中必须指定以下内容：

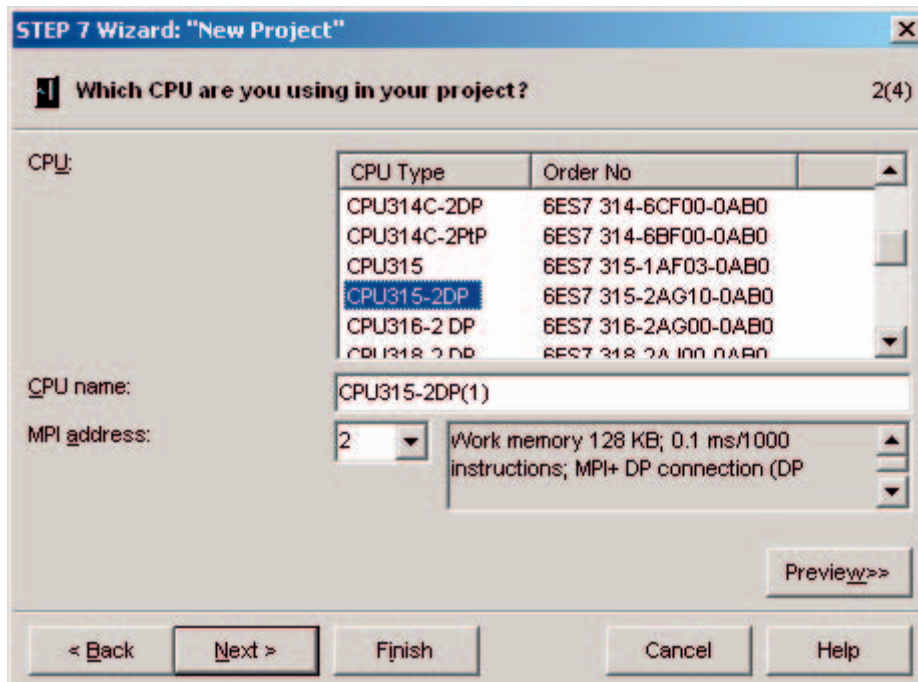
- CPU 类型
- 基本用户程序
- 组织块
- 项目名称

单击“Next”（下一步）。

6.1.2 CPU 选择

请按以下步骤进行操作

为实例项目选择 CPU 315-2DP。（您也可以将我们的实例用于不同的 CPU。这种情况下请选择适当的 CPU。）



图片 6-3 “New Project”（新建项目）向导：选择 CPU

单击“Next”（下一步）。

6.1.3 定义基本用户程序

请按以下步骤进行操作

选择 SIMATIC 语言 STL 并选择以下组织块 (OB):

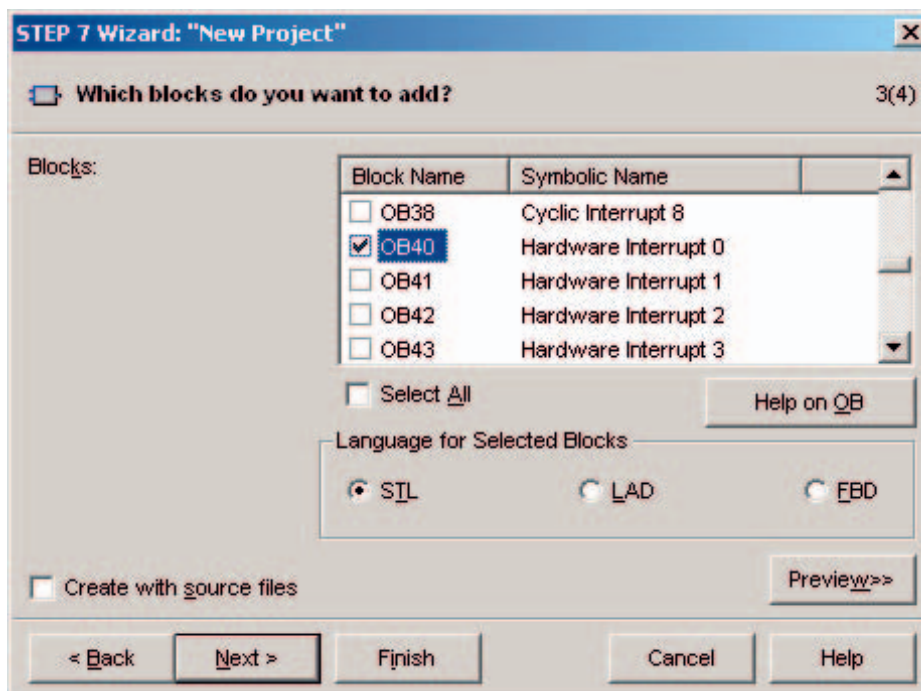
- OB1 周期性执行的块
- OB40 硬件中断
- OB82 诊断中断

每个项目都需要 OB1，并将周期性调用 OB1。

如果发生硬件中断，则调用 OB40。

如果发生诊断中断，则调用 OB82。

如果使用具有诊断功能的模块且未插入 OB82，则发生诊断报警时，CPU 将更改为 STOP 模式。



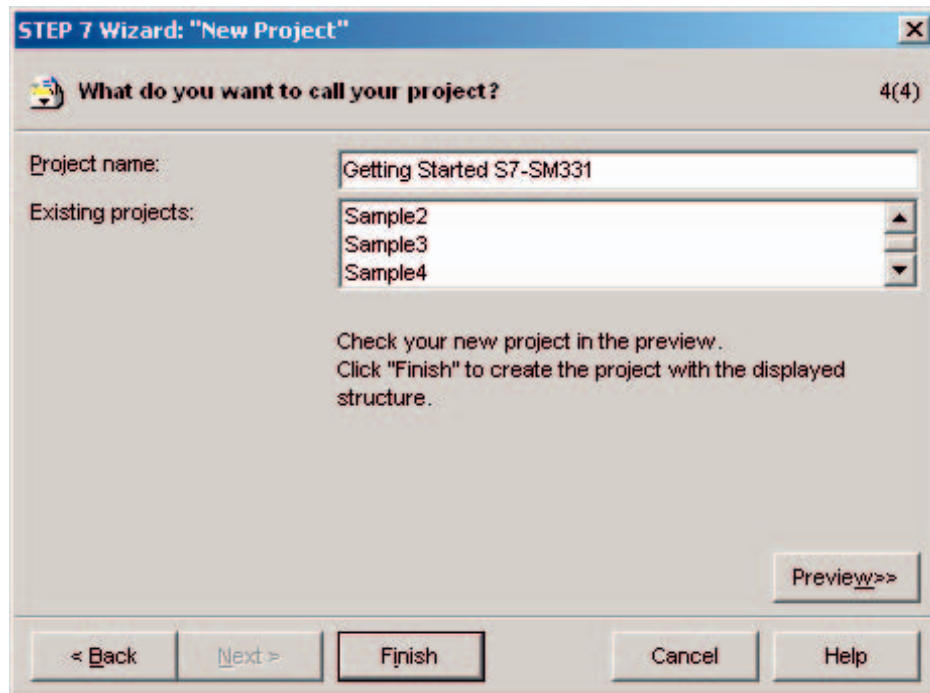
图片 6-4 “New Project”（新建项目）向导：插入组织块

单击“Next”（下一步）。

6.1.4 分配项目名称

请按以下步骤进行操作

选择“Project name”（项目名称）文本框并使用“S7 SM331 使用入门”覆盖其名称。



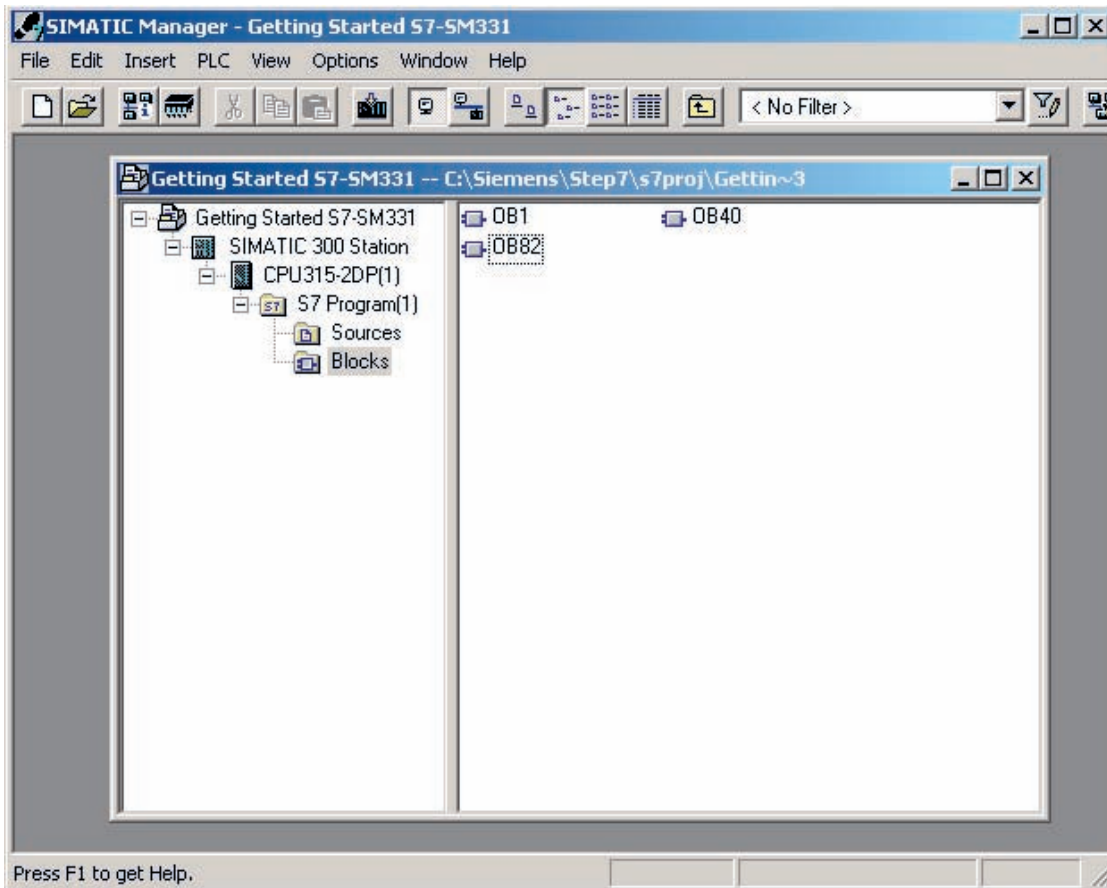
图片 6-5 “New Project”（新建项目）向导：分配项目名称

单击“Finish”（完成）。将自动创建基本的 STEP7 项目。

6.1.5 创建了结果 S7 项目

结果

该向导已创建项目“S7-SM331 使用入门”。您可以在右侧窗口中查看插入的组织块。



图片 6-6 “New Project”（新建项目）向导结果

6.2 硬件组态

6.2.1 创建硬件组态

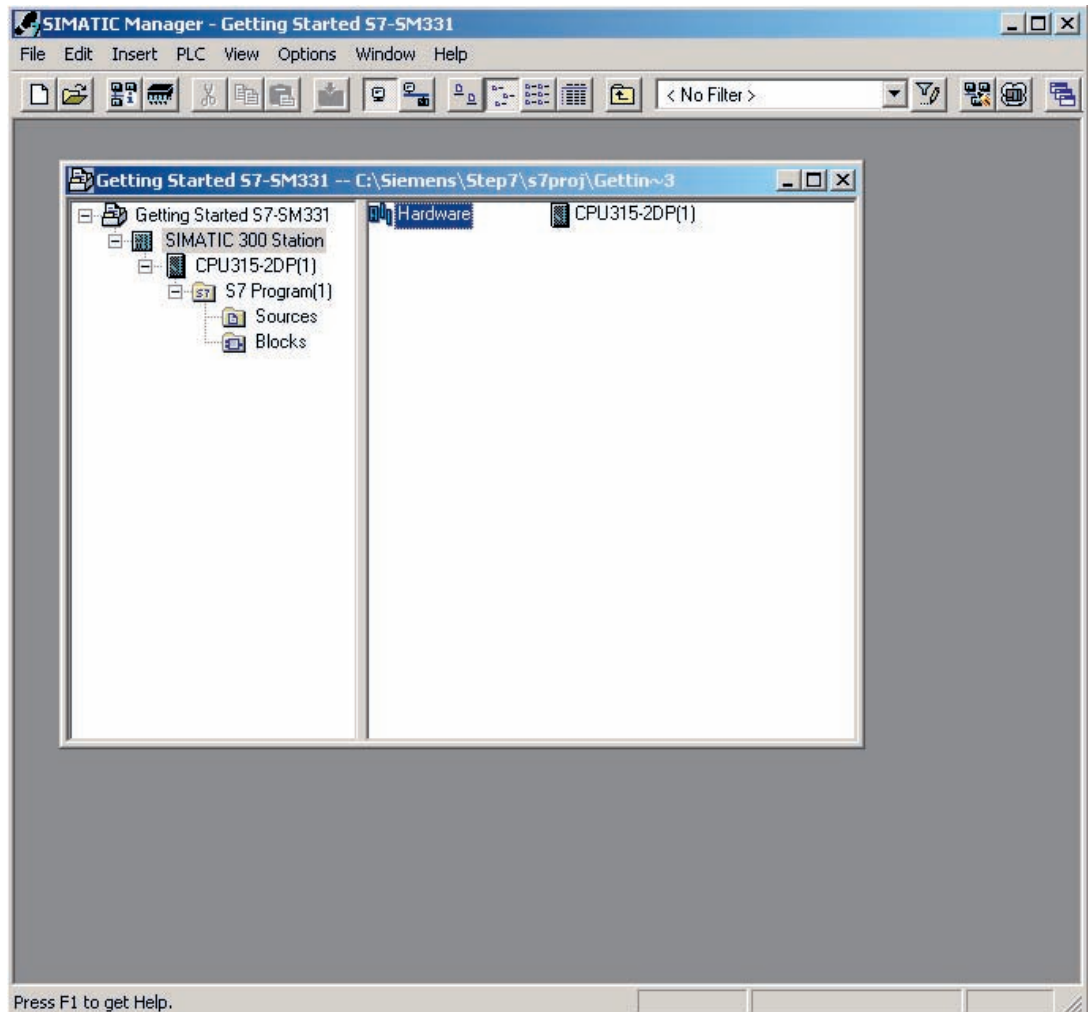
要求

STEP7 向导已创建基本 S7 项目。您还需要完成硬件组态才能为 CPU 创建系统数据。

请按以下步骤进行操作

您可以使用 SIMATIC 管理器创建实例站的硬件组态。

要执行此操作，请在左侧窗口中选择文件夹“SIMATIC 300 Station”（SIMATIC 300 站）。
双击右侧窗口中的文件夹“Hardware”（硬件）启动硬件组态。



图片 6-7 打开硬件组态

6.2.2 添加 SIMATIC 组件

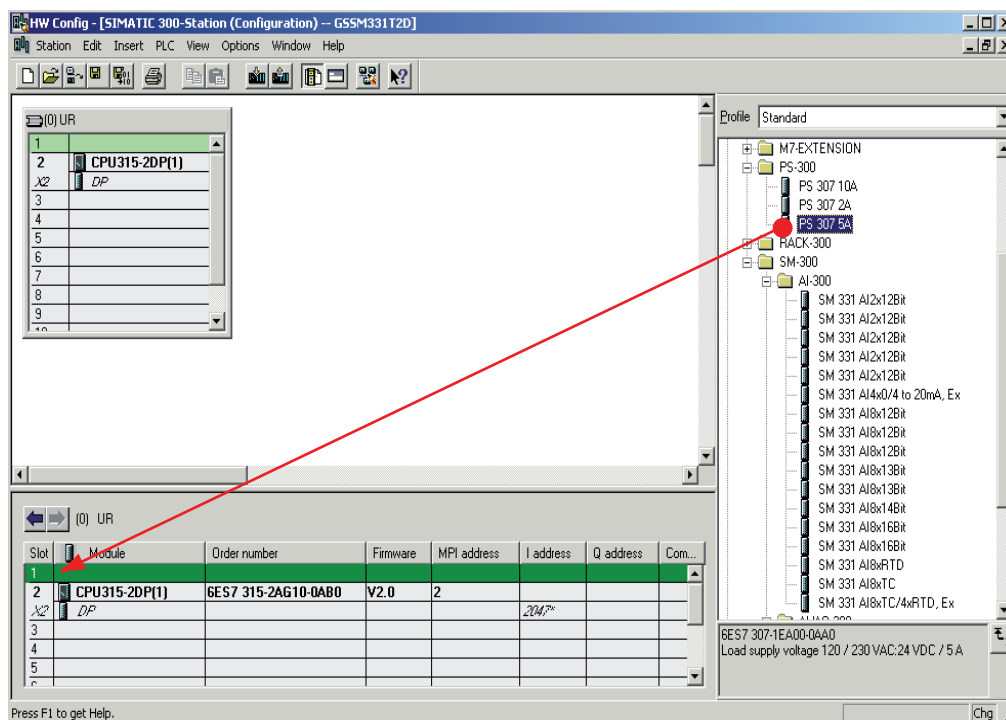
请按以下步骤进行操作

首先从硬件目录中选择电源模块。

如果硬件目录不可见，请使用快捷键 **Ctrl+K** 或单击目录图标（蓝色箭头）将其打开。

在硬件目录中，您可以从文件夹 **SIMATIC 300** 浏览到文件夹 **PS-300**。

选择 **PS307 5A** 并将其拖到插槽 1 中（请参阅红色箭头）。



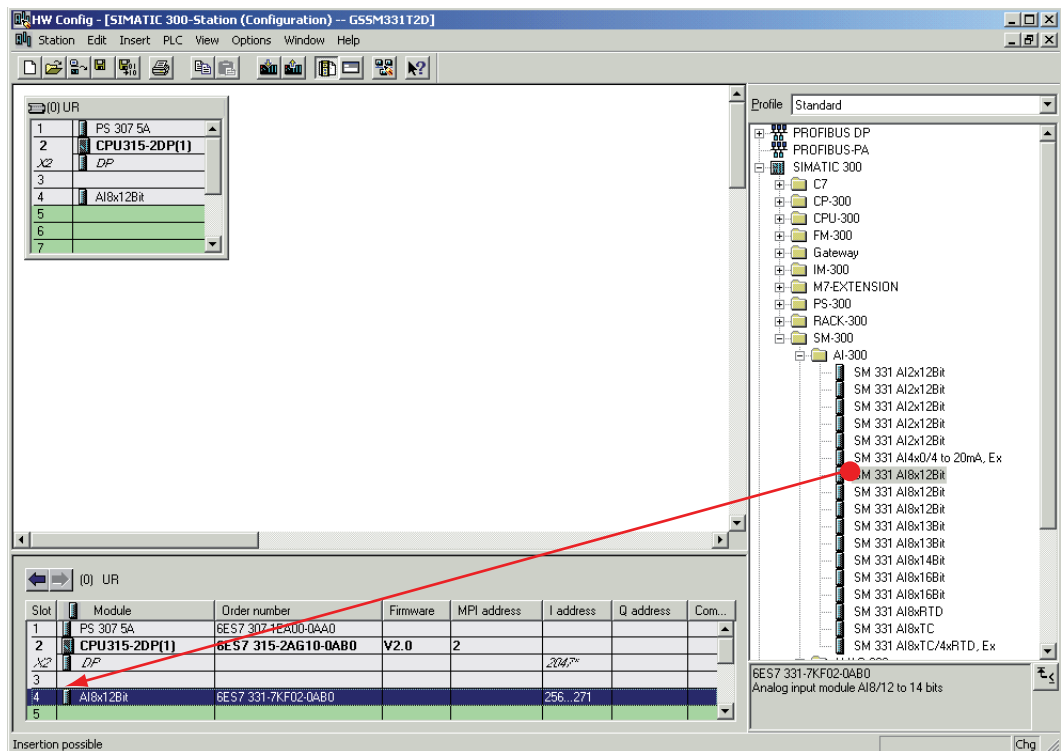
图片 6-8 硬件组态：基本组态

结果：PS 307 5A 显示在机架组态中。

插入模拟模块

有多种 SM331 模拟模块。在此项目中，我们使用订货号为 6ES7 331-7KF02-0AB0 的 SM331, AI8x12 位。

订货号将显示在硬件目录的底部（请参阅蓝色箭头）。



图片 6-9 硬件组态：SM331 插入

将模块拖到机架插槽 4 上的第一个可用域（请参阅红色箭头）。

您已将所有模块插入硬件组态中。在下一步中，将对模块进行组态。

6.2.3 组态模拟模块

概述

SIMATIC 管理器将插入具有默认设置的模拟模块。您可以修改参数以更改传感器类型、诊断和中断功能。

安装实例站

表中显示了必须为实例站设置的参数。

实例站的 SM331 功能

功能	说明
过程响应	<ul style="list-style-type: none"> • 诊断 — 启用 • 超出限制时硬件中断 — 启用
编码器 1	<ul style="list-style-type: none"> • 2 线制电流变送器 • 组诊断 • 检查断线 • 测量范围 6 mA 到 18 mA
编码器 2 和 3	<ul style="list-style-type: none"> • 4 线制电流变送器 • 组诊断 • 检查断线 • 测量范围 6 mA 到 18 mA

打开组态

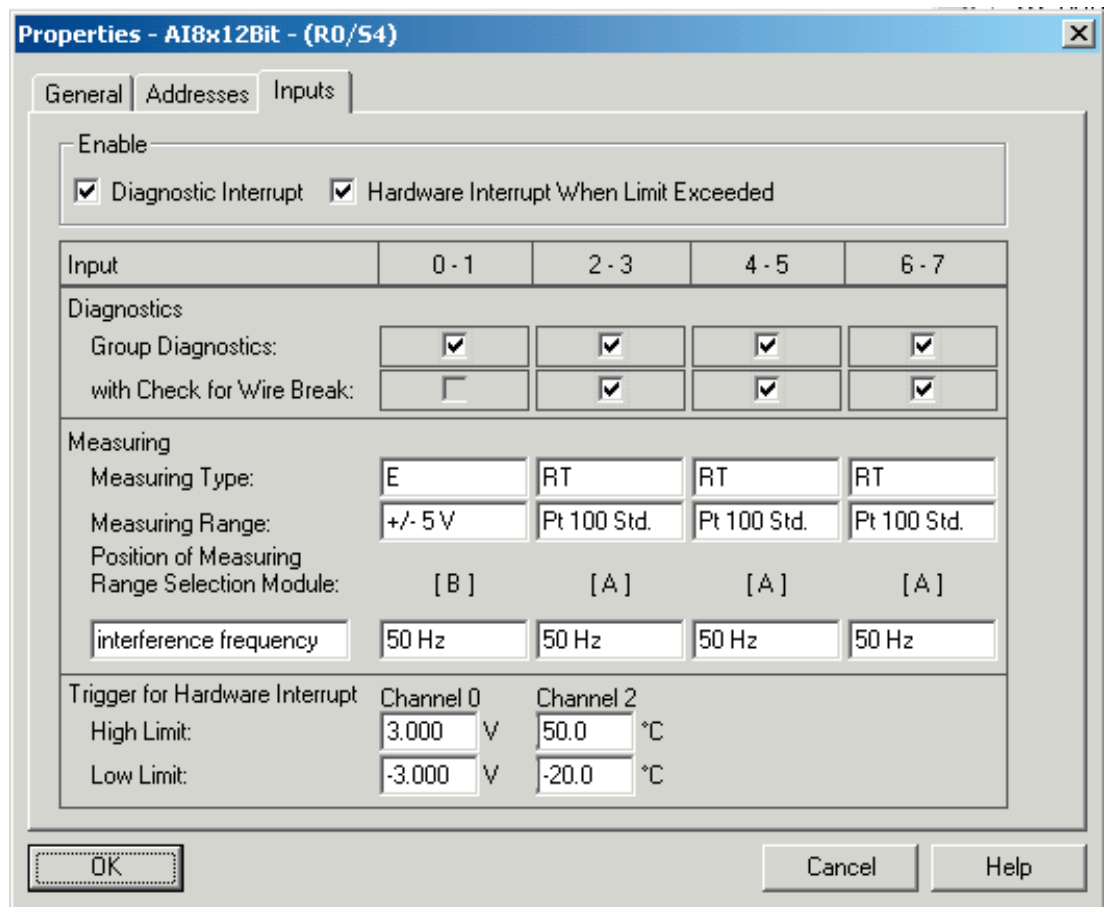
双击 SM331 所在的插槽 4

选择“Inputs”（输入）标签。

组态以下功能：

- 已启用的诊断中断
- 已启用的硬件中断
- 输入 0-1:
 - 测量类型：2DMU
 - 已启用的组诊断
 - 已启用的断线
- 输入 2-3:
 - 测量类型：2DMU
 - 已启用的组诊断
 - 已启用的断线

- 输入 4-5 和 6-7
 - 测量类型: 禁用 (---)
- 干扰频率
 - 选择电源频率 (50 Hz 或 60 Hz)
- 硬件中断触发
 - 上限值 18 mA
 - 下限值 6 mA



图片 6-10 SM331: 组态

各个设置的说明

测量类型:

2DMU 和 4DMU 代表 2 线制和 4 线制电流变送器

--- 表示通道已取消激活。如果您取消激活一些通道，则处理剩余通道的速度会更快。

量程卡

将显示量程卡需要的设置。

干扰频率（干扰频率抑制）

交流电源系统的频率可干扰测量值，尤其是在低电压范围内进行测量和使用热电偶元件时。可使用该参数指定站点上电源的频率。

该参数同时会影响间隔、积分时间和通道组的基本执行周期。

- 分辨率（精度）

模拟值存储在 16 位字中。

- 积分时间

模块需要一段时间测量模拟信号。这段时间称为积分时间。要求的精度越高，模块测量电压需要的时间就越长。

- 基本处理时间

除了积分时间，模块还需要一段时间以提供二进制值。

精度、干扰频率和积分周期之间的关系

分辨率	干扰频率	积分时间	基本处理时间
9 位	400 Hz	2.5 ms	24 ms
12 位	60 Hz	16.6 ms	136 ms
12 位	20 Hz	20 ms	176 ms
14 位	10 Hz	100 ms	816 ms

硬件中断:

仅通道 0 和 2 具有硬件中断功能。您可以在模拟信号超出其上下限时使用硬件中断触发报警。

完成硬件组态:

关闭带有组态的窗口。

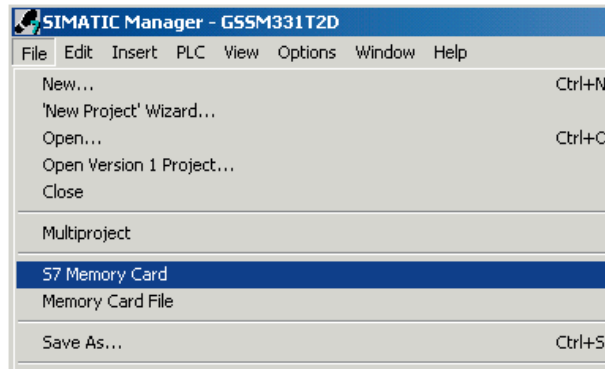


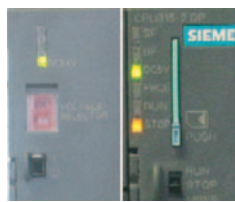

使用命令 **Station（站） > Save and Compile（保存并编译）**（或按 **Ctrl+S**）来编译并保存该项目

至此便完成该项目的硬件组态。

6.2.4 测试

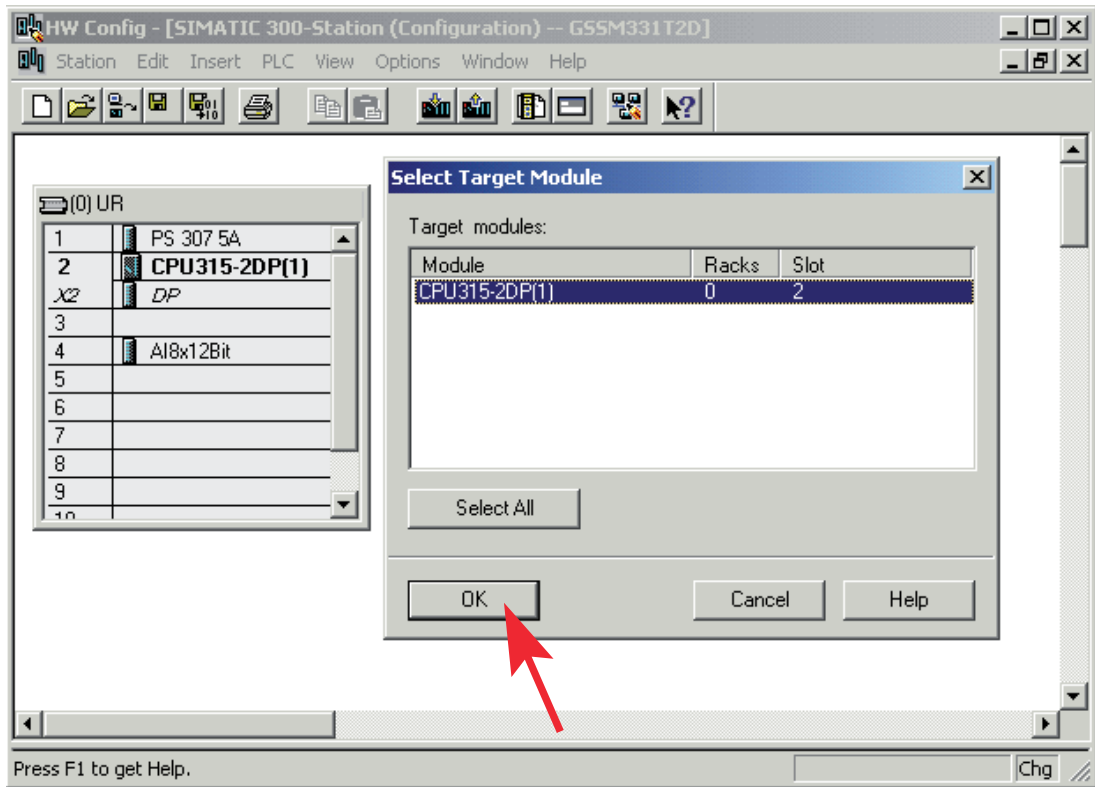
请按以下步骤进行操作

要进行测试，请做一个加电测试并下载系统数据。

步骤	图形控制器	说明
1		<p>使用 Power PG 或带有外部编程设备的 PC 擦除微型存储卡上的内容： 在 SIMATIC 管理器中单击“File”（文件）->“S7 Memory Card”（S7 存储卡）>“Delete ...”（删除...）。</p> <p>将删除 MCC 卡。</p>
2		<p>切断 CPU 的电源。 将 MMC 卡插入 CPU。 接通电源。</p>
3		<p>如果 CPU 处于 RUN 模式，则将其设置为 STOP 模式。</p>
4		<p>再次接通电源。 如果 STOP LED 闪烁，则表示 CPU 请求复位。将模式切换为 MRES 片刻以确认该操作。</p>
5		<p>使用 MPI 电缆将 CPU 连接到 PG。 要执行此操作，请将 MPI 电缆连接到 CPU 的 MPI 端口。将另一端连接到编程设备的 PG 接口。</p>

下载硬件组态

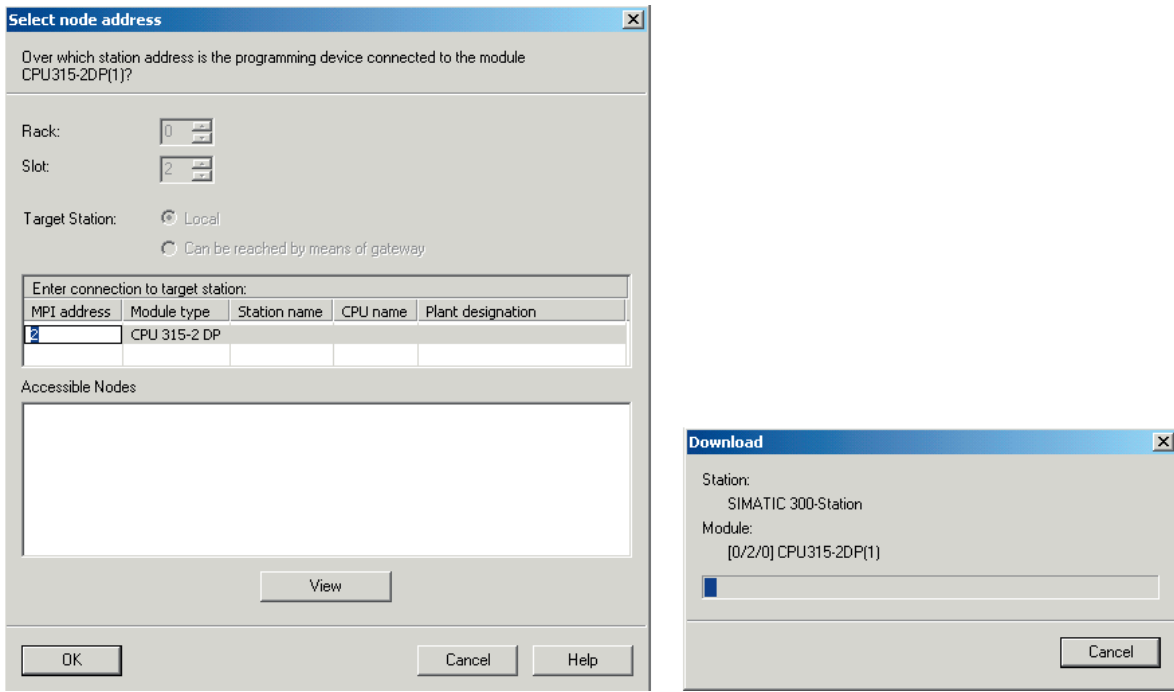
使用 HW Config 将硬件组态下载到 CPU。



图片 6-11 下载 CPU 硬件组态 (1)

单击“Load to module”（装载到模块）图标（显示在红色圆圈内）。

出现“Select target module”（选择目标模块）对话框时，单击“OK”（确定）。



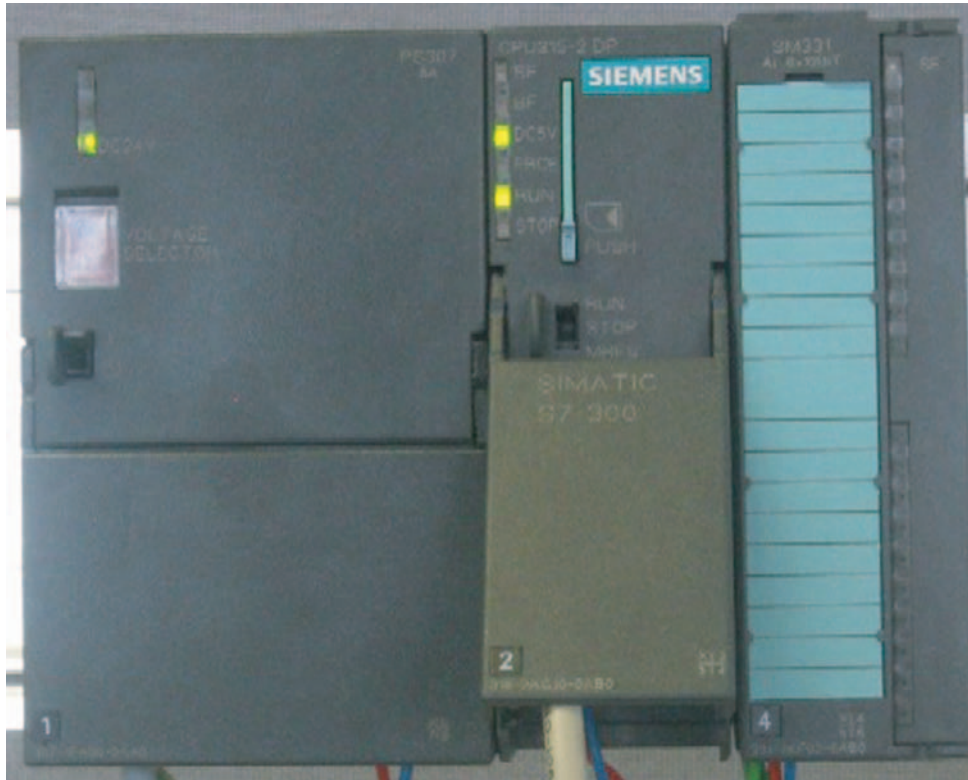
图片 6-12 下载 CPU 硬件组态 (2)

将显示“Select target address”（选择目标地址）对话框。单击“OK”（确定）。现在，系统数据将传输到 CPU 中。

启动 CPU

将 CPU 切换为 RUN。

如果已正确进行硬件组态，CPU 上的两个红色 LED（RUN 和 DC5V）将亮起。



图片 6-13 CPU 处于无错状态

6.3 STEP7 用户程序

6.3.1 用户程序的任务

概述

在我们的实例中，传感器值存储在数据块中。同时，硬件中断状态应存储在标记字中。这样就可以通过一个位确认状态信息。

此外，通道值（输入字的值）应存储在另一个数据块中。

必须在用户程序中执行以下任务：

- 将模拟输入值周期性存储在数据块中 (DB1)。
- 将模拟输入值周期性转换为浮点数值 (FC1) 并存储在数据块中 (DB2)。
- 当确认标记 (M200.0) 为 TRUE 时确认硬件中断状态。
- 当发生硬件中断时，将状态保存在标记字中 (MW100)。

下表描述了用户程序的结构：

调用类型	负责组织块	要编程的任务	使用的块或标记
周期性执行	OB1	保存模拟输入值	DB1
		转换并存储传感器值	FC1、DB2
		确认硬件中断	M200.0
硬件中断触发的调用	OB40	保存状态	MW100
诊断中断触发的调用	OB82	必须执行，因为正在使用具有诊断功能的模块	---

关于 OB82

OB82 用于具有诊断功能的模块。如果已启用这种模块的诊断报警，则当检测到故障（进入和离开事件）时 OB82 将请求诊断。操作系统则会调用 OB82。

在我们的实例中，将使用 OB82 来防止 CPU 更改为 STOP 模式。您可以对 OB82 中的诊断中断编写响应程序。

6.3.2 创建用户程序

请按以下步骤进行操作

存在两种创建用户程序的方法。

- 如果您了解如何编写 STEP7 SCL 程序，则可以在 STEP7 的“Blocks”（块）文件夹中创建必需的块和功能块，并对这些块编程。
- 您可以将用户程序从 SCL 源插入到项目中。在本“使用入门”中，将介绍该方法。

在 STEP7 中创建用户程序需要三个步骤：

1. 直接从 HTML 页下载源文件
2. 导入源文件
3. 编译源

下载源文件

您可以从下载该“使用入门”的 HTML 页直接下载源文件。

德语版源文件的名称为“GSSM331T1DE.AWL”。

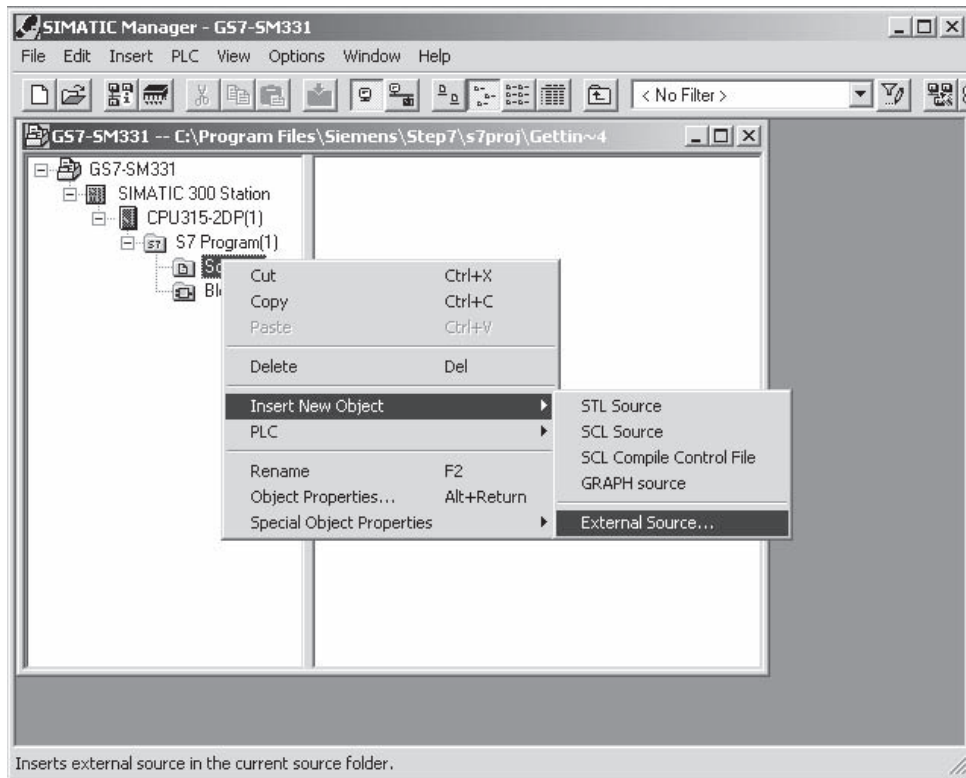
将源文件保存到硬盘上。

导入源文件

您可以按照以下步骤将源文件导入 SIMATIC 管理器：

在文件夹“Sources”（源）上单击鼠标右键。

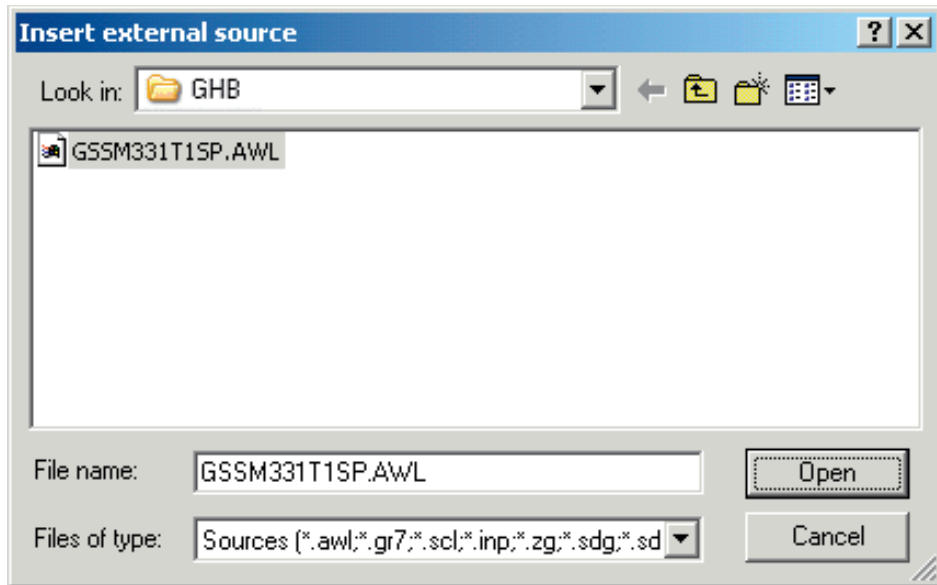
选择“Insert new Object”（插入新对象）>“External Source...”（外部源...）。



图片 6-14 导入外部源

在“Insert external source”（插入外部源）对话框中，浏览源文件 GSSM331T1DE.AWL（已下载且保存在硬盘上）。

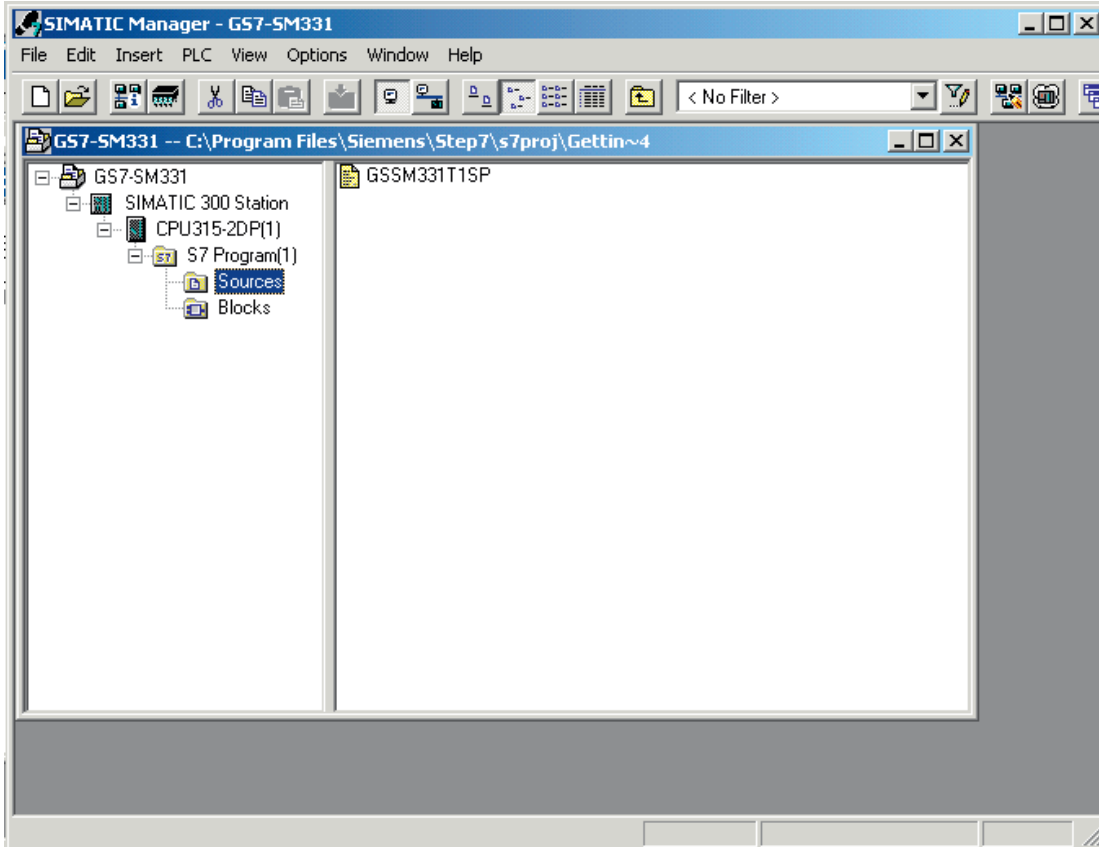
选择源文件 GSSM331T1DE.AWL（红色箭头）。



图片 6-15 导入外部源

单击“Open”（打开）。

SIMATIC 管理器已打开源文件。在右侧窗口中，您可以看到源文件已插入。

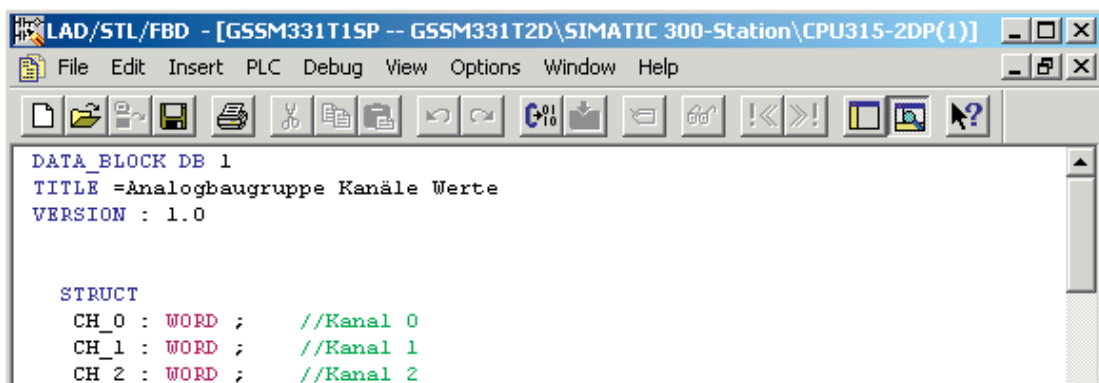


图片 6-16 编译源代码

编译源代码

要创建可执行 STEP7 程序，必须编译 STL 源。

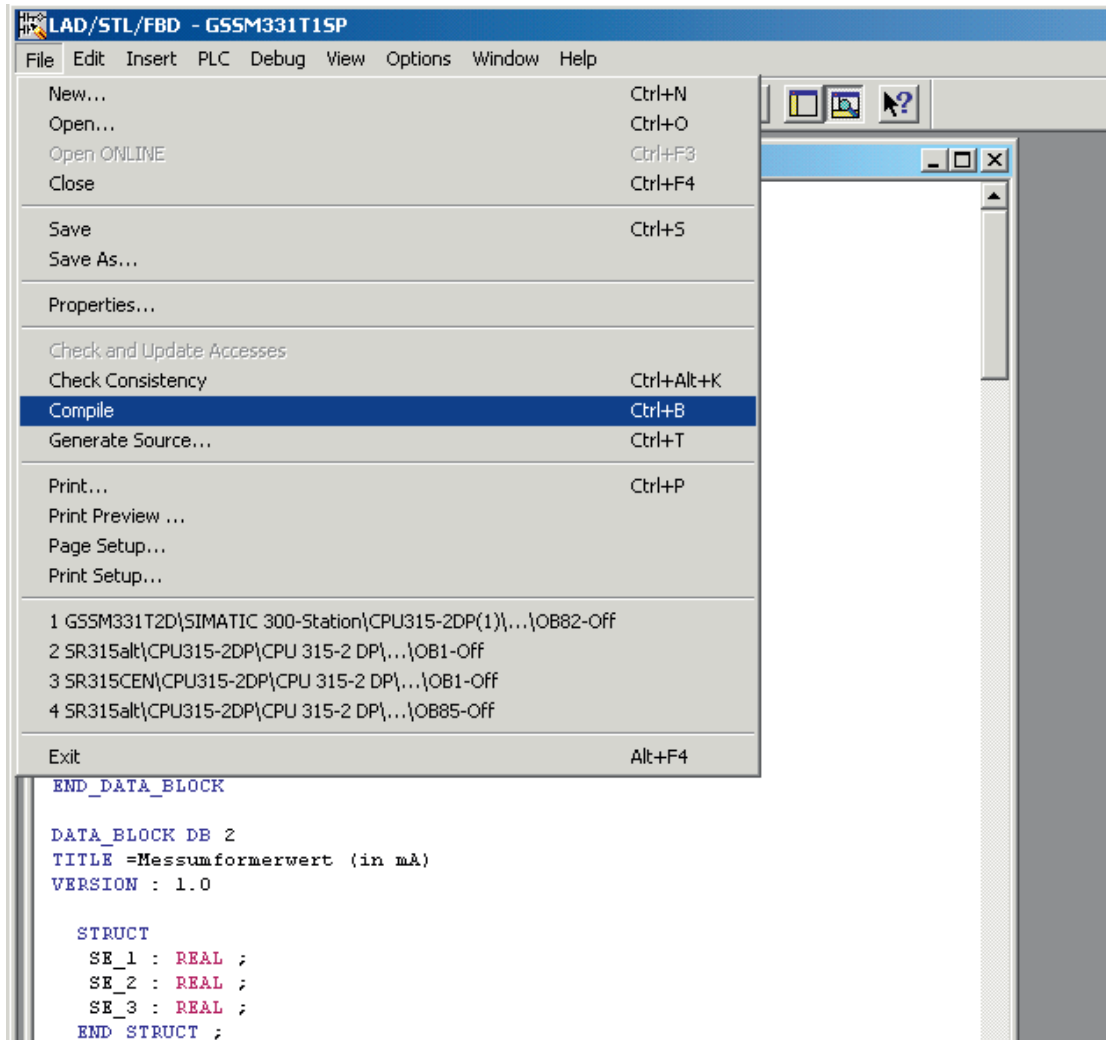
双击“Sources”（源）文件夹中的源文件 GSSM331T1DE。将打开源代码编辑器。
在源代码编辑器的窗口中，您可以查看源代码。



图片 6-17 源代码编辑器

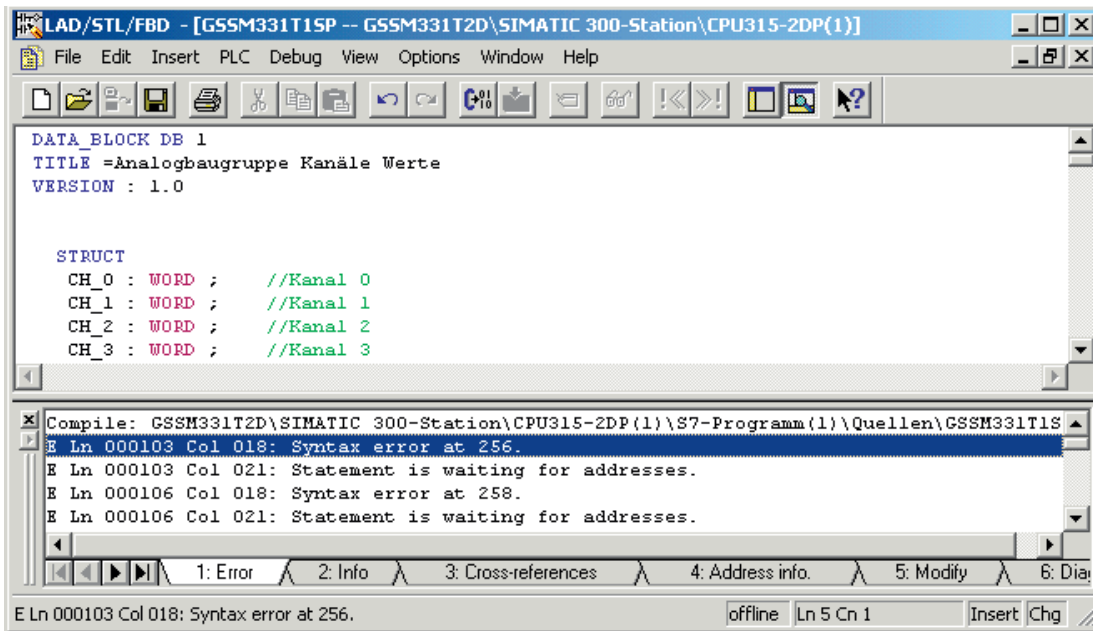
装载源代码后，开始编译。

按快捷键 **Ctrl+B** 或选择 **File (文件) > Compile (编译)**。将立即开始编译。



图片 6-18 编译 STL 源

如果出现警告或错误消息，请检查源代码。

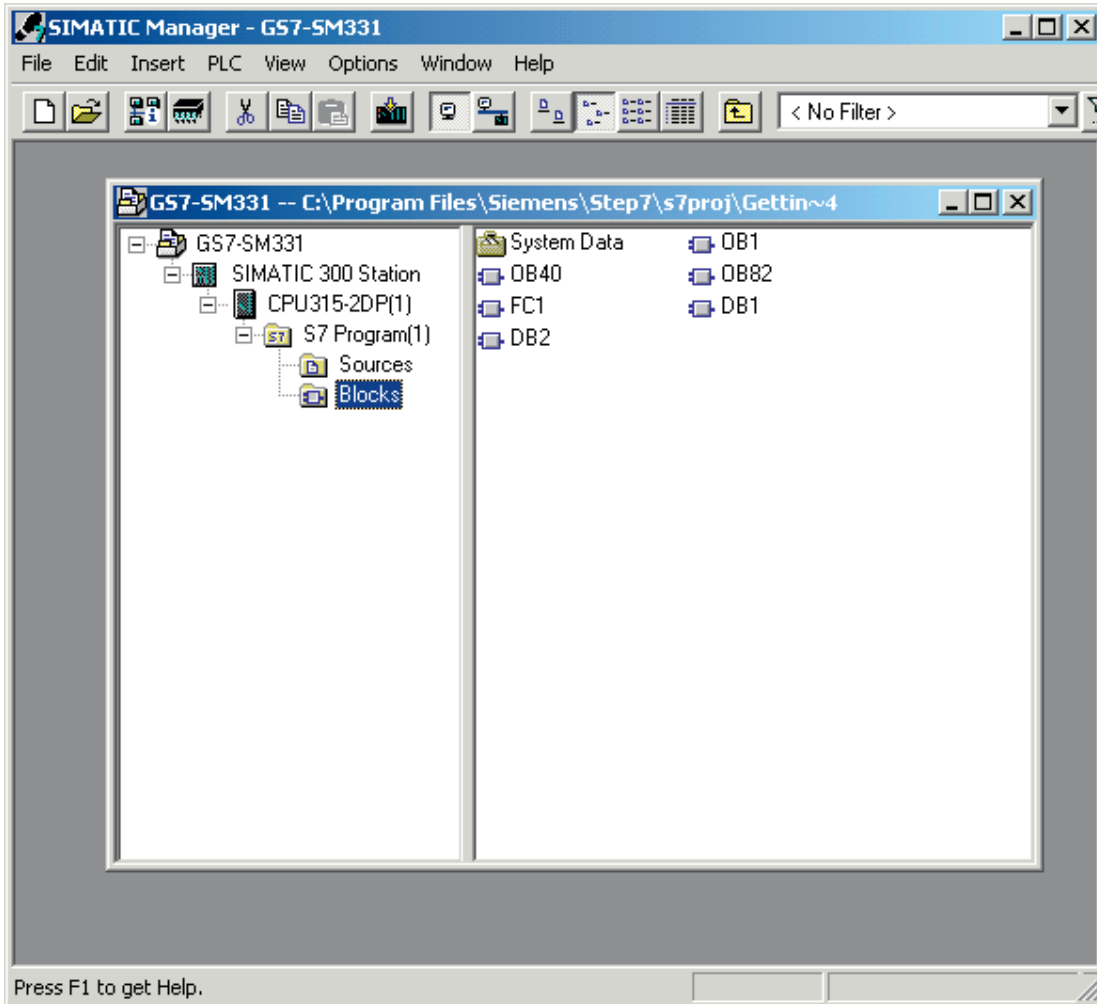


图片 6-19 源代码编辑器，编译后的消息

关闭源代码编辑器。

编译 STL 源完成且没有出现错误时，“Blocks”（块）文件夹中应显示以下块：

OB1、OB40、OB82、FC1、DB1 和 DB2



图片 6-20 生成的块

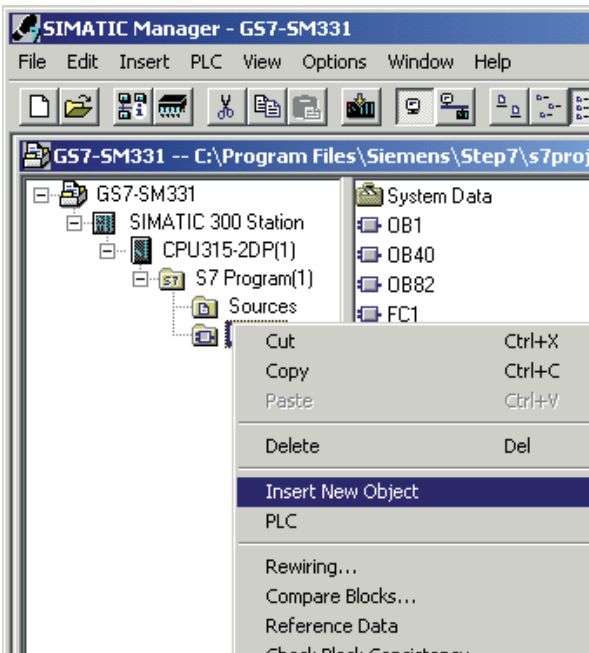
测试用户程序

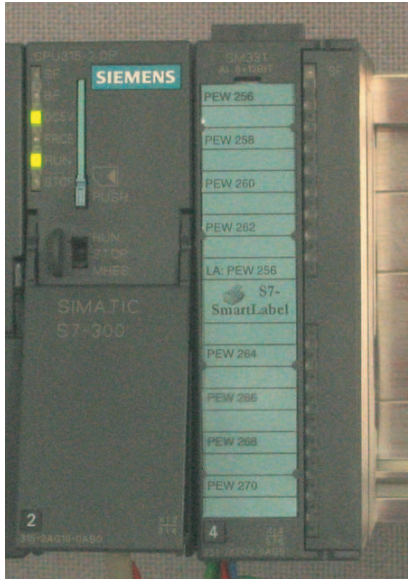
7.1 下载系统数据和用户程序

请按以下步骤进行操作

硬件和软件已准备就绪。下一步就是将系统数据和用户程序下载到自动化系统中。要执行此操作，请按以下步骤进行操作：

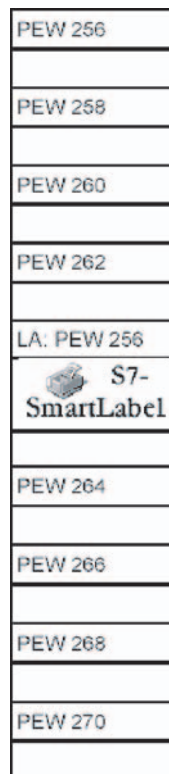
下载系统数据和用户程序

步骤	图形控制器	说明
1		使用 SIMATIC 管理器将用户程序和系统数据（包含硬件组态）下载到 CPU 中。

步骤	图形控制器	说明
2		<p>请按照屏幕上显示的说明进行操作。</p> <p>如果所有传感器均已正确连接，CPU 和 SM331 将不显示错误指示灯。</p> <p>绿色的“RUN”指示灯将显示 CPU 的状态。</p>

Smart Label

已使用 Siemens S7 Smart Label 创建模块的标签条（订货号为 2XV9 450-1SL01-0YX0）。
实际大小的标签条：



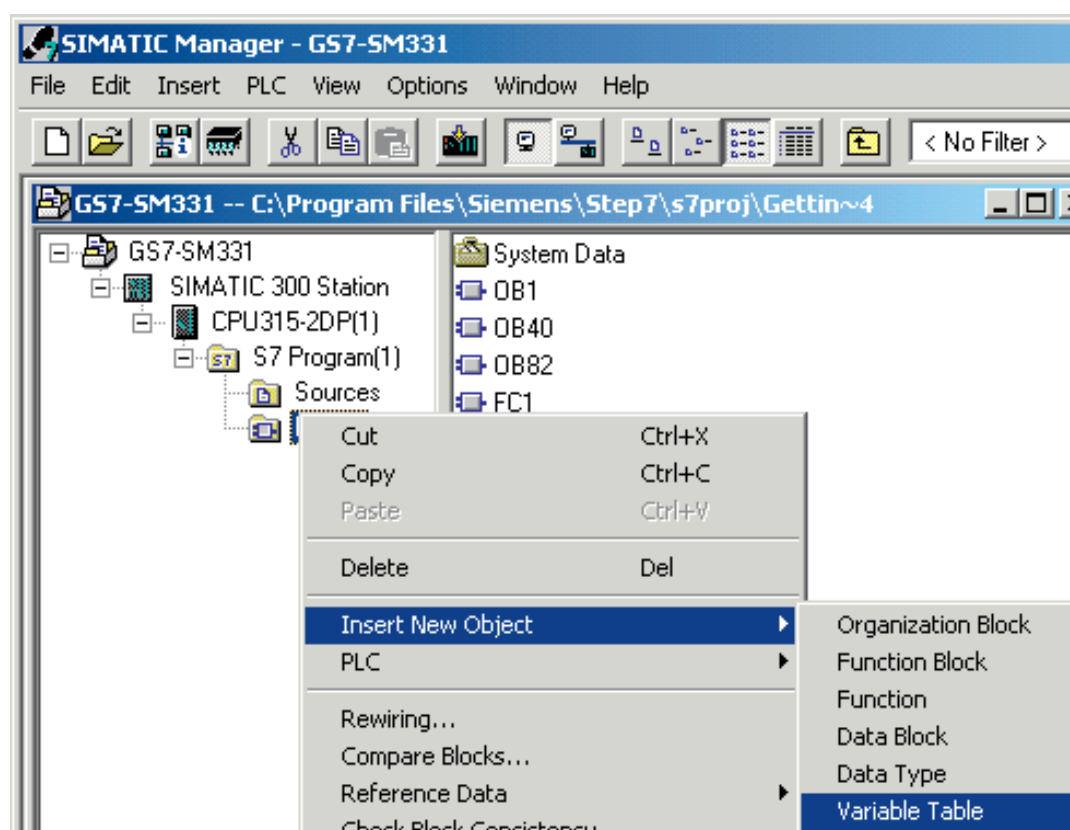
图片 7-1 实例中的 S7-SmartLabel 标签条

7.2 传感器值的可视化

请按以下步骤进行操作

要使传感器值可视化，请将如下变量表插入到项目中。要执行此操作，请从“Blocks”（块）文件夹的右键快捷菜单中选择：

Insert new object（插入新对象）> Variable Table（变量表）



图片 7-2 插入变量表

如下填写新变量表:

	Address	Symbol	Display format	Status value	Modify value
1	// Channel values				
2	DB1.DBW 0		HEX		
3	DB1.DBW 2		HEX		
4	DB1.DBW 4		HEX		
5	DB1.DBW 6		HEX		
6	DB1.DBW 8		HEX		
7	DB1.DBW 10		HEX		
8	DB1.DBW 12		HEX		
9	DB1.DBW 14		HEX		
10					
11	// Analog values				
12	DB2.DBD 0		FLOATING_POINT		
13	DB2.DBD 4		FLOATING_POINT		
14	DB2.DBD 8		FLOATING_POINT		
15	DB2.DBD 12		FLOATING_POINT		
16	// Process control status				
17	M 200.0		BOOL		
18	MW 100		BIN		
19					

- (1) 在该区域中, 您可以监视通道值
- (2) 在该区域中, 您可以查看模拟值
- (3) 在该区域中, 您可以监视和控制状态信号

变量说明

变量	说明
DB1.DBW 0	模拟值的通道 0 显示
DB1.DBW 2	模拟值的通道 1 显示
DB1.DBW 4	模拟值的通道 2 显示
DB1.DBW 6	模拟值的通道 3 显示
DB1.DBW 8	模拟值的通道 4 显示
DB1.DBW 10	模拟值的通道 5 显示
DB1.DBW 12	模拟值的通道 6 显示
DB1.DBW 14	模拟值的通道 7 显示
DB2.DBD 0	变送器 1 的电流 (mA)
DB2.DBD 4	变送器 2 的电流 (mA)
DB2.DBD 8	变送器 3 的电流 (mA)
MW 100	状态硬件中断
MW 200.0	确认硬件中断
M101.0	通道 0 已超出下限
M101.1	通道 0 已超出上限
M101.2	通道 2 已超出下限
M101.3	通道 0 已超出上限

监视值

要监视值，请单击眼镜符号打开控制器的在线视图。现在，您可以监视数据块和标记中的值。

	Address	Symbol	Display format	Status value	Modify value
1	// Channel values				
2	DB1.DBW 0		HEX	W#16#0000	
3	DB1.DBW 2		HEX	W#16#0000	
4	DB1.DBW 4		HEX	W#16#009C	
5	DB1.DBW 6		HEX	W#16#7FFF	
6	DB1.DBW 8		HEX	W#16#0114	
7	DB1.DBW 10		HEX	W#16#7FFF	
8	DB1.DBW 12		HEX	W#16#01AF	
9	DB1.DBW 14		HEX	W#16#7FFF	
10					
11	// Analog values				
12	DB2.DBD 0		FLOATING_POINT	0.0	
13	DB2.DBD 4		FLOATING_POINT	15.6	
14	DB2.DBD 8		FLOATING_POINT	27.6	
15	DB2.DBD 12		FLOATING_POINT	43.1	
16	// Process control status				
17	M 200.0		BOOL	false	<input type="checkbox"/> true
18	MW 100		BIN	2#0000_0000_0000_0000	
19					

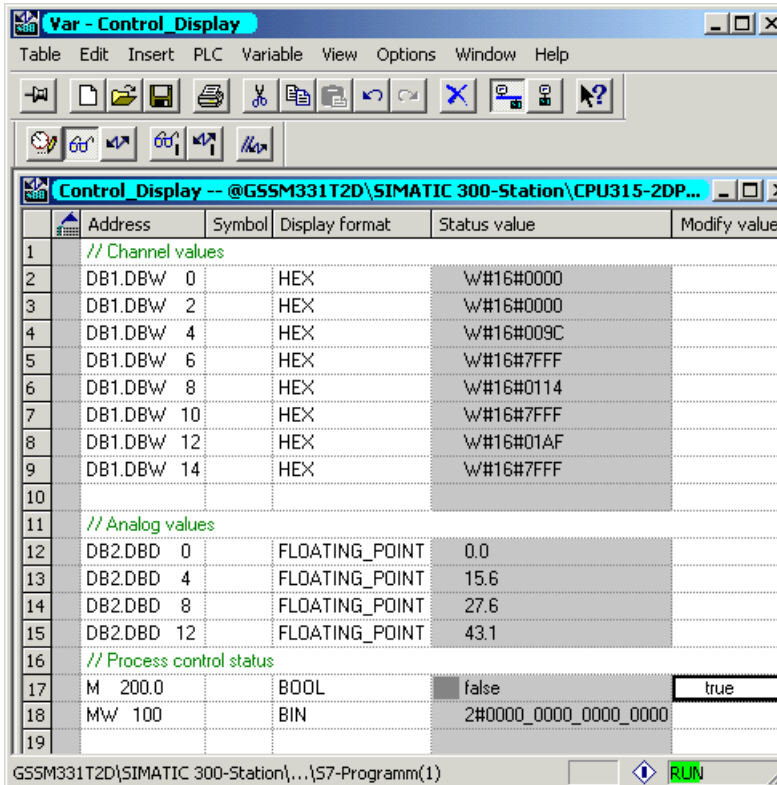
图片 7-3 变量表的在线视图

- (1) 十六进制格式的通道值
- (2) 转换的模拟值
- (3) 状态信息

控制值

要控制过程确认，请将所需值（TRUE 或 FALSE，取决于您要激活还是取消激活确认）输入列“Control Value”（控制值）中，并单击具有两个箭头的图标。

控制变量



- (1) 通道值
- (2) 模拟值
- (3) 状态

监视值时的特性

监视值时，您必将发现通道值与模拟值不同。原因是模拟值仅支持二进制格式“字”（16 位）。因此，必须转换模拟模块的值。

7.3 模拟值表达式

请按以下步骤进行操作

CPU 仅处理二进制格式的模拟值。模拟输入模块将模拟过程信号转换为数字格式（16 位字）。

当将数字值转换为模拟值时，必须考虑五个范围：

以电流表示模拟值，其范围为 4 到 20 mA

十六进制值	电流范围	注释	含义
7FFF	22.96 mA	上溢	从十六进制值 16#F700 开始，传感器值高于组态的测量值范围，将不再有效。
7F00			
7EFF	22.81 mA	过载范围	达到上溢范围之前，该范围相当于一个容差区。可是在该范围内精度不是最优的。
6C01			
6C00	20 mA	额定范围	额定范围是记录测量值的正常范围。该范围可保证最优精度。
5100	15 mA		
1	4 mA + 578.7 nA		
0	4 mA		
FFFF		欠载范围	与过载范围对应的范围，但用于较低的值。
ED00	1.185 mA		
ECFF		下溢	从十六进制值 16#ECFF 开始，传感器值低于组态的测量值范围，将不再有效。
8000			

需要转换二进制格式的为模拟过程值以便显示。在我们的实例中，将显示 mA。通过在已编程的功能 (FC1) 中将模拟值的显示转换为以 mA 表示可实现此目的。

说明

在我们的实例中，可以从变送器的输出中查看这些值。

现在，您可以一个电流计比较仪表上的值和显示的模拟值。这两个值将是相同的。

诊断中断

8.1 从 PG 中读取诊断信息

概述

诊断中断功能使用户程序可以对硬件错误进行响应。

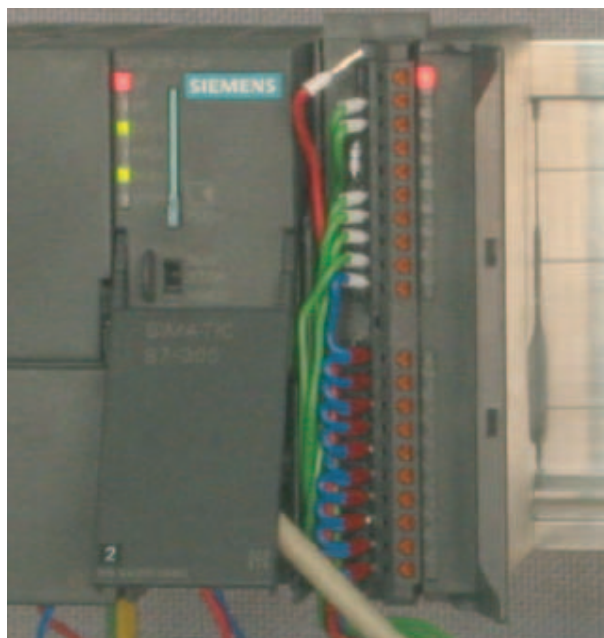
模块必须具有诊断功能才能生成诊断中断。

在 OB82 中，您可以编写诊断中断的响应程序。

显示

模拟输入模块 SM331 AI8x12 具有诊断功能。

SM331 和 CPU 上的红色“SF”LED 发出信号指示发生了诊断中断。



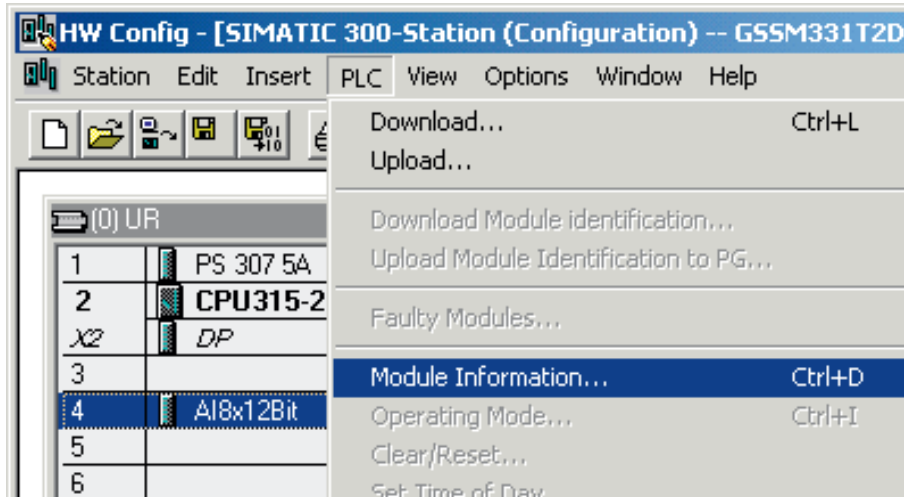
图片 8-1 硬件错误

通过请求硬件状态可以“在线”确定错误原因。

8.1 从 PG 中读取诊断信息

要“在线”确定模块状态，请按以下步骤进行操作：

选择硬件组态中的 SM331。单击菜单命令 CPU -> Module Information...（模块信息...）以执行硬件诊断。

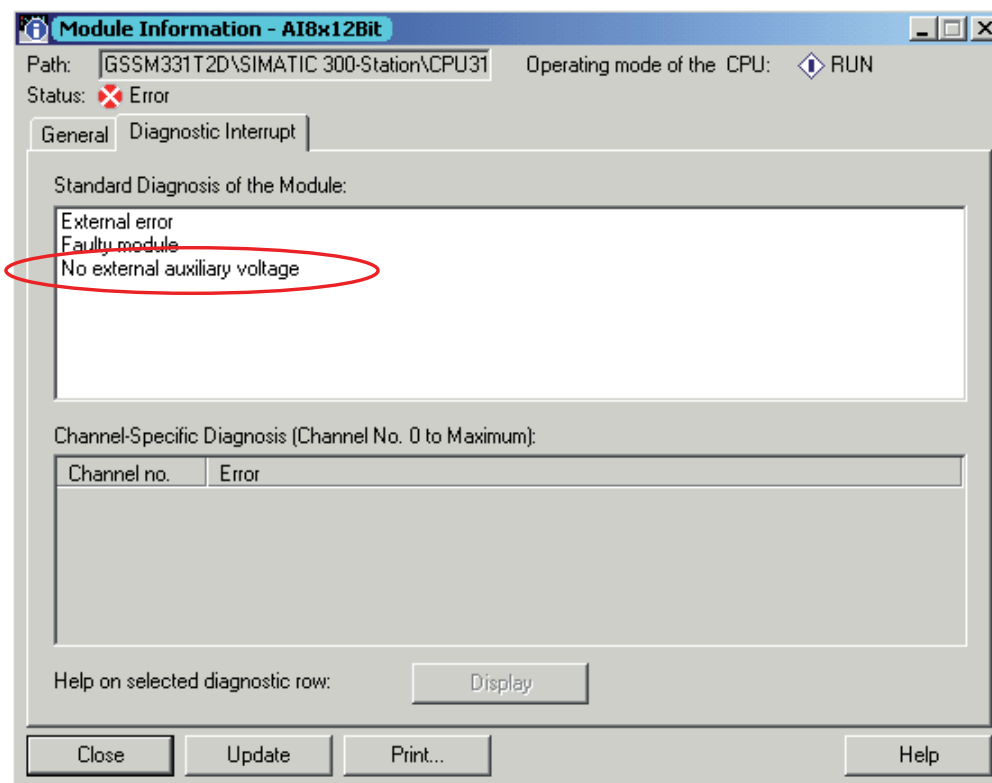


图片 8-2 模块状态

8.2 常规诊断

Diagnostic interrupt (诊断中断) 标签

在 Diagnostic Interrupt (诊断中断) 标签上, 您将看到报告的错误的信息。
中断与通道无关且适用于整个模块。



图片 8-3 SM331 的诊断

8.3 与通道相关的诊断中断

8.3.1 存在五种与通道相关的诊断中断

存在五种与通道相关的诊断中断：

- 组态/编程错误
- 共模错误
- 断线
- 下溢
- 上溢

说明

在此仅说明对测量模式 2 或 4 线制电流变送器的特定通道诊断。其它测量模式与此相似，因此不再加以说明。

8.3.2 组态/编程错误

含义

量程卡的位置与硬件组态中设置的测量模式不匹配。

8.3.3 共模错误

含义

输入 (M-) 和测量电路 (M_{ana}) 的共模电压电位之间的电压差 U_{cm} 过高。

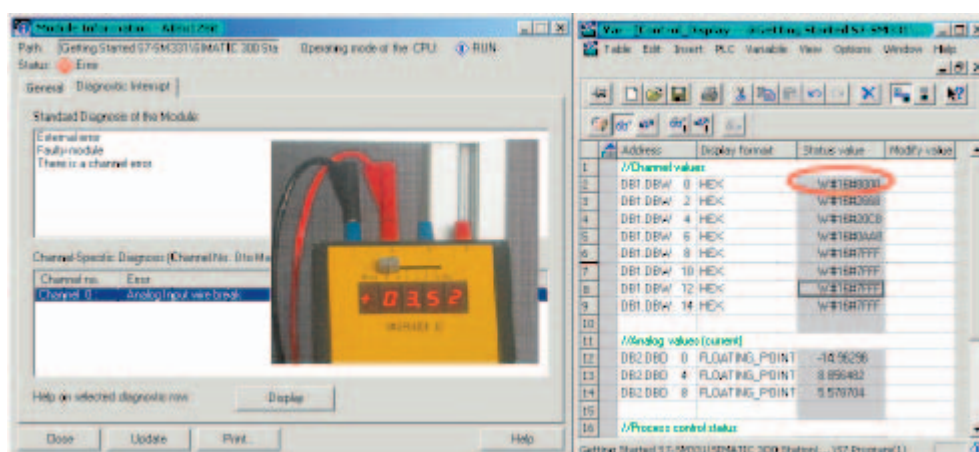
在我们的实例中，不会发生该错误，因为 M_{ana} 连接到了 2 线制变送器（固定电位）的 M。

8.3.4 断线

含义

如果对 2 线制变送器启用了断线检测，将不会直接检查断线。但诊断将反映下限电流值的不足。

如果使用 4 到 20 mA 电流变送器，则当电流低于 3.6 mA 时，模块诊断中将显示诊断消息“Analog input wire break”（模拟输入断线）。



图片 8-4 左：出现断线时的诊断消息/右：变量表

即使测量电流明显高于 1.1185 mA，模拟值的显示仍将立即显示下溢（十六进制 8000）。

仅当禁用了断线检测时，才允许电流低于 3.6 mA。

8.3.5 下溢

含义

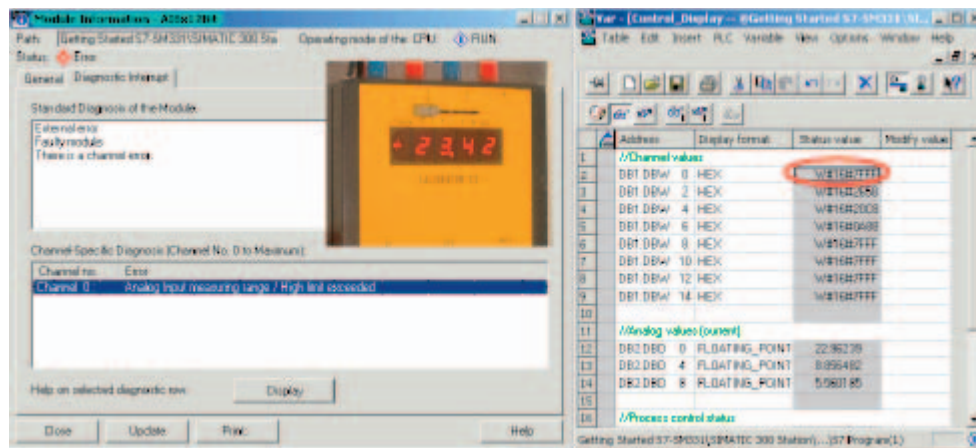
即使测量电流明显高于 1.1185 mA，模拟值的显示仍将立即显示下溢。

8.3.6 上溢

含义

如果电流超过 22.81 mA，将显示上溢消息，表明 “Analog in-put measuring range / High limit exceeded”（超出模拟输入测量范围/上限）。

模拟值的显示（十六进制 7FFF）位于上溢范围内。



图片 8-5 左：出现上溢时的诊断消息/右：变量表

说明

禁用通道的模拟显示值也为 7FFF（十六进制）。

硬件中断

9.1 硬件中断

概述

SM331 AI8x12 位的一大特色就是其触发硬件中断的功能。可以对两个通道（0 和 2）进行相应组态。

硬件中断通常触发 CPU 中的报警组织块。在我们的实例中，将调用 OB40。

必须以 mA 为单位指定硬件中断的限制值。

实例：

已将带有 4 到 20 mA 变送器的压力传感器连接到通道 0。这里应该以 mA 而不是 Pa 为单位指定限制值。

限制值

为触发硬件中断，限制值必须在测量模式的额定值内。

实例：

如果启用了断线检测 (3.6 mA)，而选择 3.5 mA 作为下限值，则系统将接受该设置。因为始终首先触发诊断报警，所以不会触发硬件中断。

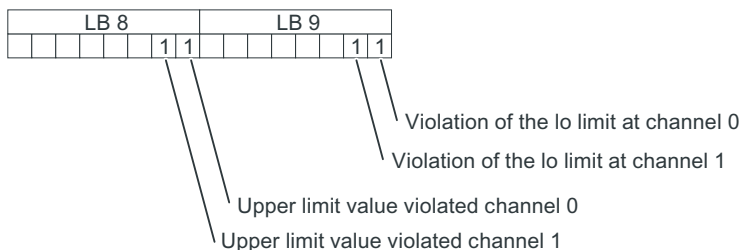
在我们的实例中，将在以下限制下组态两个通道（传感器 1 和 2）：

- 下限值 6 mA
- 上限值 18 mA

决定功能

如果发生硬件中断，则调用 OB40。在 OB40 的用户程序中，您可以编写自动化系统对硬件中断进行响应的程序。

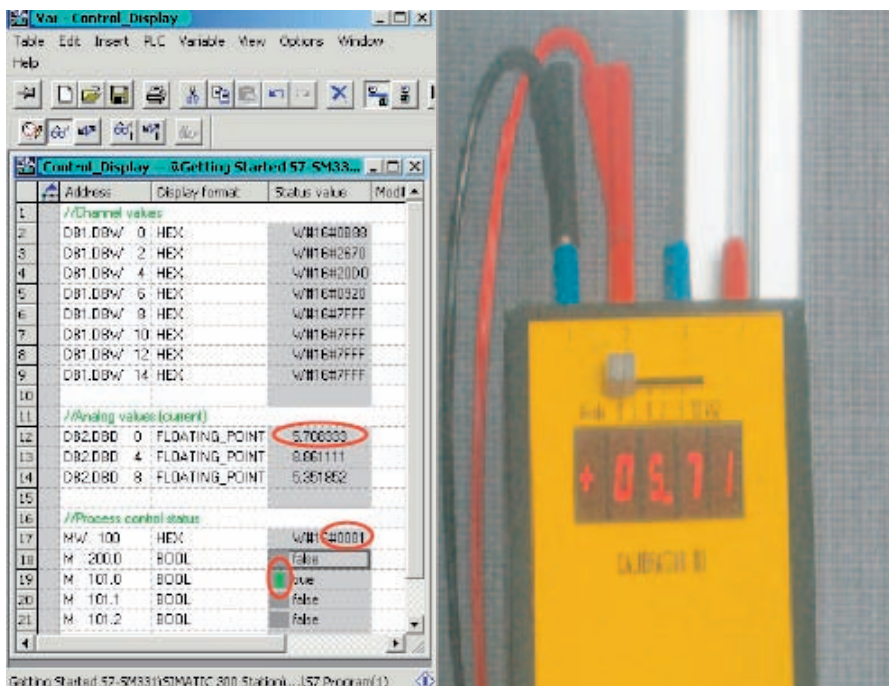
在实例的用户程序中，OB40 将读取硬件中断的原因。这可以在临时变量结构 OB40_POINT_ADDR（本地字 8 到 11）中找到。



图片 9-1 OB40 启动信息：在该实例中，还触发了硬件中断

在该实例中，OB40 只将 LD8 和 LD9 传送到标记字 (MW100)。在现有变量表中监视该标记字。您可以通过设置标记位 M200.0 或在变量表中将其设置为 TRUE，在 OB1 中确认该标记字。

如果使用校准设备为通道 0 提供 5.71 mA 的电流，则将在变量表中获得 MW100 的值 0001（十六进制）。这意味着已调用 OB40 且通道 0 已超出其下限值 (6 mA)。



图片 9-2 硬件中断：通道 0 已超出下限值

附录

A.1 用户程序的源

STL 源代码

在本节中，您将从实例中看到用户程序的源代码。

您可以从下载该“使用入门”的 HTML 页直接下载源文件。

```
DATA_BLOCK DB 1
TITLE =Analog module channel values
VERSION :1.0
STRUCT
    CH_0 : WORD ; //Channel 0
    CH_1 : WORD ; //Channel 1
    CH_2 : WORD ; //Channel 2
    CH_3 : WORD ; //Channel 3
    CH_4 : WORD ; //Channel 4
    CH_5 : WORD ; //Channel 5
    CH_6 : WORD ; //Channel 6
    CH_7 : WORD ; //Channel 7
END_STRUCT ;
BEGIN
    CH_0 := W#16#0;
    CH_1 := W#16#0;
    CH_2 := W#16#0;
    CH_3 := W#16#0;
    CH_4 := W#16#0;
    CH_5 := W#16#0;
    CH_6 := W#16#0;
    CH_7 := W#16#0;
END_DATA_BLOCK

DATA_BLOCK DB 2
TITLE =Current transducer (in mA)
VERSION :1.0

STRUCT
    SE_1 : REAL ; //Sensor 1 current value (mA)
    SE_2 : REAL ; //Sensor 2 current value (mA)
    SE_3 : REAL ; //Sensor 3 current value (mA)
```

A.1 用户程序的源

```

END_STRUCT ;

BEGIN
    .SE_1 := 0.000000e+000;
    SE_2 := 0.000000e+000;
    SE_3 := 0.000000e+000;
END_DATA_BLOCK

FUNCTION FC 1 : VOID
TITLE =Conversion of a channel's raw values in mA
VERSION : 1.0

VAR_INPUT
    Raw : WORD ;           // Analog value display
END_VAR
VAR_OUTPUT
    Current : REAL ;      // Current in mA
END_VAR
VAR_TEMP
    TDoubleInt : DINT ;
    TInt : INT;
END_VAR
BEGIN
NETWORK
TITLE = Conversion of raw values in mA

    L   #Raw;
    T   #TInt;

// Only long integers can be converted into REAL format
    L   #TInt;
    ITD ;
    T   #TDoubleInt;
    L   #TDoubleInt;      //          HEX value
    DTR ;                 // Current = -----
    T   #Current;         //          1728

    L   1.728000e+003;    //          !          /
    /R   ;                //          !          /
    T   #Current;         //          !          /
                                // ----- +----- /----- +-----
                                //                                     4          20

    L   4.000000e+000;    // Offset correction
    +R   ;
    T   #Current;

END_FUNCTION

```

```

ORGANIZATION_BLOCK OB 1
TITLE = "Main Program Sweep (Cycle)"
VERSION : 1.0

VAR_TEMP
  OB1_EV_CLASS : BYTE ;           //Bits 0-3 = 1 (Coming event),
                                   Bits 4-7 = 1 (Event class 1)
  OB1_SCAN_1 : BYTE ;           //1 (Cold restart scan 1 of OB 1),
                                   3 (Scan 2-n of OB 1)
  OB1_PRIORITY : BYTE ;         //Priority of OB Execution
  OB1_OB_NUMBR : BYTE ;         //1 (Organization block 1, OB1)
  OB1_RESERVED_1 : BYTE ;       //Reserved for system
  OB1_RESERVED_2 : BYTE ;       //Reserved for system
  OB1_PREV_CYCLE : INT;         //Cycle time of previous OB1 scan (milliseconds)
  OB1_MIN_CYCLE : INT;         //Minimum cycle time of OB1 (milliseconds)
  OB1_MAX_CYCLE : INT;         //Minimum cycle time of OB1 (milliseconds)
  OB1_DATE_TIME : DATE_AND_TIME ; //Date and time OB1 started
END_VAR
BEGIN
NETWORK
TITLE =Read channels
// Channel values 0 to 7 are loaded and stored in DB1 (channel values)

      L      PEW 256;           //Channel 0
      T      DB1.DBW      0;

      L      PEW 258;           //Channel 1
      T      DB1.DBW      2;

      L      PEW 260;           //Channel 2
      T      DB1.DBW      4;

      L      PEW 262;           //Channel 3
      T      DB1.DBW      6;

      L      PEW 264;           //Channel 4
      T      DB1.DBW      8;

      L      PEW 266;           //Channel 5
      T      DB1.DBW     10;

      L      PEW 268;           //Channel 6

```

A.1 用户程序的源

```

        T      DB1.DBW      12;

        L      PEW 270;          //Channel 7
        T      DB1.DBW      14;

NETWORK
TITLE = Conversion
// Conversion of the channel's raw data into current values (mA)
        CALL  FC          1 (
                Raw              := DB1.DBW 0,
                Current          := DB2.DBW 0);
        CALL  FC          1 (
                Raw              := DB1.DBW 4,
                Current          := DB2.DBW 4);
        CALL  FC          1 (
                Raw              := DB1.DBW 6,
                Current          := DB2.DBW 8);

NETWORK
TITLE = Reset hardware interrupt
// Even though the hardware interrupt was reset by the hardware upon terminating OB40
// the value of the hardware interrupt must be reset manually
        U      M          200.0;
        SPBN          1b10;
        L      MW          100;
        SSI          4;
        T      MW
        1b10: NOP      0;      100;

NETWORK
TITLE =The End

        BE;

END_ORGANIZATION_BLOCK

ORGANIZATION_BLOCK OB 40
TITLE = "Hardware Interrupt"
// Processing OB40_POINT_ADDR (L8 to L11)
//
//L8 High limit value exceeded
//L9 Low limit value exceeded
VERSION : 1.0

VAR_TEMP
        OB40_EV_CLASS : BYTE ;          //Bits 0-3 = 1 (Coming event),
                                        Bits 4-7 = 1 (Event class 1)
        OB40_STRT_INF : BYTE ;          //16#41 (OB 40 has started)

```

```

OB40_PRIORITY : BYTE ;           //Priority of OB Execution
OB40_OB_NUMBR : BYTE ;           //40 (Organization block 40, OB40)
OB40_RESERVED_1 : BYTE ;         //Reserved for system
OB40_IO_FLAG : BYTE ;            //16#54 (input module), 16#55 (output module)
OB40_MDL_ADDR : WORD ;           //Base address of module initiating interrupt
OB40_POINT_ADDR : DWORD ;        //Interrupt status of the module
OB40_DATE_TIME :                 //Date and time OB40 started
DATE_AND_TIME ;

END_VAR
BEGIN
NETWORK
TITLE =Sensor 1 (Channel 0): Lower limit value

      U      L      9.0;          // Channel 0 low limit value
      SPBNB  L001;
      L      W#16#1;
      L      MW      100;
      OW     ;
      T      MW      100;
L001:  NOP    0;

NETWORK
TITLE =Sensor 1 (Channel 0): Upper limit value

      U      L      8.0;          // Channel 0 upper limit value
      SPBNB  L002;
      L      W#16#2;
      L      MW      100;
      OW     ;
      T      MW      100;
L002:  NOP    0;

NETWORK
TITLE =Sensor 2 (Channel 2): Lower limit value

      U      L      9.2;          // Channel 2 low limit value
      SPBNB  L003;
      L      W#16#4;
      L      MW      100;
      OW     ;
      T      MW      100;
L003:  NOP    0;

```

A.1 用户程序的源

```
NETWORK
TITLE =Sensor 2 (Channel 2): Upper limit value

      U      L      8.2;      // Channel 2 upper limit value
      SPBNB  L004;
      L      W#16#8;
      L      MW      100;
      OW     ;
      T      MW      100;
L004:  NOP    0;

NETWORK
TITLE =Sensor 3 (Channel 3): Lower limit value
// Only for demonstration purposes. Channel 3 has now hardware interrupt capabilities
      U      L      9.3;      // Channel 3 low limit value
      SPBNB  L005;
      L      W#16#10;
      L      MW      100;
      OW     ;
      T      MW      100;
L005:  NOP    0;

NETWORK
TITLE =Sensor 3 (Channel 3): Upper limit value
// Only for demonstration purposes. Channel 3 has now hardware interrupt capabilities
      U      L      8.3;      // Channel 3 upper limit value
      SPBNB  L006;
      L      W#16#20;
      L      MW      100;
      OW     ;
      T      MW      100;
L006:  NOP    0;

END_ORGANIZATION_BLOCK
```


索引

数字

- 2 线制电流变送器
 - 接线原理, 22
- 4 线制电流变送器
 - 接线原理, 23

A

安装

- CPU, 12
- SM331, 18
- 电源, 12
- 模拟模块, 13
- 前连接器, 18
- 设备, 11

B

变量表

- 填写, 54
- 变量说明, 54
- 变送器
 - 端子, 16

C

CPU

- 安装, 12
- 具有编程设备的 CPU, 39
- 启动, 42
- 为电源接线, 20
- 选择, 29

CPU 315-2DP

- 组态, 27

参数

- 设置, 36

测量类型, 38

测试

- 连接, 26
- 设置, 39
- 用户程序, 51

插入

- 模拟模块, 35
- 总线连接器, 12

产品的组件

- SM331 模块, 14

传感器值

- 可视化, 53

创建

- STEP7 项目, 27
- 用户程序, 44

错误

- 连接, 26
- 错误显示, 61

D

导入

- 源文件, 45

电流变送器

- 接线原理, 22

电源

- 安装, 12
- 螺钉, 12
- 为 CPU 接线, 20

电源频率, 38

调用

- 硬件组态, 33
- 组态, 36

定义

- 定义基本用户程序, 30
- 发生硬件中断时的功能, 66

定义基本用户程序

- 定义, 30

读出

- 从 PG 中读取诊断信息, 59

端子

- 变送器, 16

断线, 63

F

- 分辨率, 38
- 分配
 - 项目名称, 31
- 负载电源
 - 选择, 34

G

- 干扰频率, 38
- 干扰频率抑制, 38
- 共模错误, 62
- 过程确认, 56

J

- 积分时间, 38
- 基本处理时间, 38
- 加电, 39
- 监视
 - 值, 55
- 检查
 - 线路电压, 21
- 接线
 - SM331 前连接器, 25
 - 模拟模块, 22, 23
 - 为电源和 CPU 接线, 20
- 接线原理
 - 2 线制电流变送器, 22
 - 4 线制电流变送器, 23
 - 电流变送器, 22
- 结构
 - 用户程序, 43

K

- 可视化
 - 传感器值, 53
- 控制
 - 值, 56
- 控制值, 56

L

- LED, 26
 - 红色, 59
 - 绿色, 42
- 连接
 - 测试, 26
 - 错误, 26
 - 具有编程设备的 CPU, 39
- 量程卡, 38
 - 定位, 17
 - 位置, 16
- 螺钉
 - 电源, 12
 - 装配导轨, 11

M

- 模块的标签条, 52
- 模拟模块
 - 安装, 13
 - 插入, 35
 - 接线, 22, 23
 - 特性, 15
 - 需要的硬件和软件, 7
 - 组态, 36
- 模拟值表达式, 57
- 目标地址
 - 选择, 41

O

- OB82, 43

Q

- 启动
 - CPU, 42
 - SIMATIC 管理器, 27
- 前连接器
 - 安装, 18
- 清除/复位, 39

S

- SIMATIC 管理器, 27
 - 启动, 27
 - 硬件组态, 33
- SM331
 - 安装, 18
 - 组件, 14
- SM331 模块
 - 产品的组件, 14
- SM331 前连接器
 - 接线, 25
- Smart Label, 52
- STEP7 项目
 - 创建, 27
- STEP7 用户程序, 43
 - 任务, 43
- STL, 30
- STL 源代码, 67
 - 删除
 - 微型存储卡, 39
 - 上溢, 64
- 设备
 - 安装, 11
- 设置
 - 参数, 36
 - 测试, 39
- 数字值
 - 转换为模拟值, 57

T

- 特性
 - 模拟模块, 15
- 添加
 - SIMATIC 组件, 34
- 添加 SIMATIC 组件, 34
- 填写
 - 变量表, 54
- 通道组
 - 未使用, 25

W

- 完成
 - 硬件组态, 38
- 微型存储卡
 - 删除, 39
- 位置
 - 量程卡, 16, 17

X

- 下溢, 63
- 下载
 - 将系统数据和用户程序下载到自动化系统, 51
 - 硬件组态, 40
 - 源代码, 67
 - 源文件, 44
- 下载系统数据和用户程序
 - 下载到自动化系统, 51
- 显示
 - 错误, 61
- 线路电压
 - 更改, 21
 - 检查, 21
- 限制值
 - 硬件中断, 65
- 向导“New Project...” (新建项目...), 27
- 项目名称
 - 分配, 31
- 选择
 - CPU, 29
 - 负载电源, 34
 - 目标地址, 41
 - 组织块, 30

Y

- 以电流表示模拟值,
 - 其范围为 4 到 20 mA, 57
- 硬件故障
 - 发现, 59
- 硬件和软件
 - 模拟模块, 7
- 硬件目录
 - 打开, 34
- 硬件中断
 - 限制值, 65
- 硬件中断, 38, 65
- 硬件组态
 - 调用, 33
 - 完成, 38
 - 下载, 40
 - 组态, 33
- 用户程序
 - 测试, 51
 - 创建, 44
 - 结构, 43
 - 源代码, 67