

SCE 培训资料

西门子自动化教育合作项目 | 2017/09

Beschreibung: SIE_Logo_Layer_Petrol_RGB_A4_56mmTIA 博途模块 052-201

高级语言编程  
使用 SCL 和 SIMATIC S7-1500

**本培学习/培训文档适用于以下 SCE 教育培训产品**

* **SIMATIC S7 CPU 1516F-3 PN/DP**

订货号：6ES7516-3FN00-4AB2

* **SIMATIC STEP 7 Professional V14 SP1 - 单用户许可证**

订货号：6ES7822-1AA04-4YA5

* **SIMATIC STEP 7 Professional V14 SP1 - 6 套课堂许可证**

订货号：6ES7822-1BA04-4YA5

* **SIMATIC STEP 7 Professional V14 SP1 - 6 套升级许可证**   
  订货号：6ES7822-1AA04-4YE5
* **SIMATIC STEP 7 Professional V14 SP1 - 20 套学生许可证**   
  订货号：6ES7822-1AC04-4YA5

请注意，必要时会使用后续培训产品代替本培训产品。

可通过以下网页获得最新的 SCE 可用培训产品概览：[siemens.com/sce/tp](http://www.siemens.com/sce/tp)

**培训课程**

各地的 Siemens SCE 课程培训请联系当地的 SCE 联系人。

[siemens.com/sce/contact](http://www.siemens.com/contact)

**有关 SCE 的其它信息**

[siemens.com/sce](http://www.siemens.com/sce)  
  
**使用说明**   
通用型自动化解决方案 - 全集成自动化 (TIA) 的培训资料属于“西门子自动化教育合作项目 (SCE)”，专门用于公共教育机构和研发机构的培训。Siemens AG 对其内容不承担任何担保责任。

本资料仅可用于 Siemens 产品/系统的首次培训。即允许全部或部分复印本资料并当面转交给培训人员，令其在培训框架范围内使用。允许在公共培训和进修场合出于培训目的转发、复制本资料或传播其内容。例外情况需经 Siemens AG 的书面许可。联系人：Roland Scheuerer 先生，   
[roland.scheuerer@siemens.com](mailto:roland.scheuerer@siemens.com)。

违者须承担赔偿损失责任。保留包含翻译在内的所有权利，尤其针对申请专利或实用新型登记注册时的权利。

严禁用于工业客户培训课程。我们绝不允许该资料用于商业目的。

感谢德累斯顿工业大学、Michael Dziallas Engineering 公司和其他全体参与者在本 SCE 教学资料编写过程提供的支持。

目录

[1. 目标 4](#_Toc501703718)

[2. 前提条件 4](#_Toc501703719)

[3. 必要的硬件和软件 5](#_Toc501703720)

[4. 理论 6](#_Toc501703721)

[4.1 用于编程语言 S7-SCL 6](#_Toc501703722)

[4.2 针对 S7-SCL 开发环境 6](#_Toc501703723)

[5. 任务 9](#_Toc501703724)

[5.1 储存箱容量示例任务 9](#_Toc501703725)

[5.2 示例任务扩展 9](#_Toc501703726)

[6. 规划 9](#_Toc501703727)

[6.1 全局数据模块“Data\_Tank” 9](#_Toc501703728)

[6.2 “Calculate\_Volume”函数 10](#_Toc501703729)

[6.3 “Calculate\_Volume”扩展函数 10](#_Toc501703730)

[7. 结构化的逐步式引导指南 11](#_Toc501703731)

[7.1 现有项目归档 11](#_Toc501703732)

[7.2 将项目保存在新名称下 12](#_Toc501703733)

[7.3 创建数据模块“Data\_Tank” 12](#_Toc501703734)

[7.4 创建函数“Calculate\_Volume” 14](#_Toc501703735)

[7.5 确定函数“Calculate\_Volume”的接口 15](#_Toc501703736)

[7.6 函数“Calculate\_Volume”编程 16](#_Toc501703737)

[7.7 组织模块“Main [OB1]”编程 17](#_Toc501703738)

[7.8 编译程序并加载 19](#_Toc501703739)

[7.9 监控并测试组织模块 20](#_Toc501703740)

[7.10 “Calculate\_Volume”函数扩展 22](#_Toc501703741)

[7.11 调整组织模块 27](#_Toc501703742)

[7.12 编译程序并加载 28](#_Toc501703743)

[7.13 监控并测试组织模块 29](#_Toc501703744)

[7.14 监控并测试“Calculate\_Volume”函数 31](#_Toc501703745)

[7.15 项目成功归档 34](#_Toc501703746)

[8. 检查清单 34](#_Toc501703747)

[9. 练习 35](#_Toc501703748)

[9.1 任务 - 练习 35](#_Toc501703749)

[9.2 规划 35](#_Toc501703750)

[9.3 检查清单 - 练习 36](#_Toc501703751)

[10. 更多相关信息 37](#_Toc501703752)

S7-SCL 高级语言编程

# 目标

本章中将了解高级语言 S7-SCL 的基本功能。此外，还将展示用于排除逻辑性编程错误的测试功能。

可以使用第 3 章所列的 SIMATIC S7 控制器。

# 前提条件

本章建立在已对 SIMATIC S7 进行了硬件配置的基础上。可以通过数字输入/输出卡中包含的任一硬件配置功能实现。执行本章操作时可以参考以下项目：

“SCE\_DE\_012\_101\_Hardwarekonfiguration\_CPU1516F…..zap13”

此外应了解有关高级语言编程（如 Pascal）的基础知识。

# 必要的硬件和软件

**1** 工程组态站点：需求包括硬件和操作系统 （其他相关信息，参见 TIA 博途安装 DVD 上的 Readme/自述文件）

**2** TIA 博途中的软件 SIMATIC STEP 7 Professional - V13 以上版本

**3** 控制系统 SIMATIC S7-1500/S7-1200/S7-300，例如 CPU 1516F-3 PN/DP - 固件 V1.6 以上版本，带存储卡和 16DI/16DO 以及 2AI/1AO

**4** 工程组态站点和控制系统之间通过以太网连接



**2** V13 以上版本的 SIMATIC STEP 7 Professional   
(TIA Portal)



**1** 工程组态站点

**4** 以太网连接

****

**3** SIMATIC S7 控制系统

# 理论

## 用于编程语言 S7-SCL

S7-SCL (Structured Control Language) 是一种高级的编程语言，以 Pascal 为基础并实现结构化编程。该语言符合标准 DIN EN-61131-3 (IEC 61131-3) 中规定的流程语言 SFC“顺序功能图”的特点。除了高级语言元素外，S7-SCL 还包含典型 PLC 语言元素，如输入、输出、时间、存储器、模块调用等。它支持 STEP 7 的模块设计方式，实现了语句表 (STL)、梯形图 (LAD) 和功能块图 (FDB) 以及符合标准的模块化编程。即 S7-SCL 是编程软件 STEP 7 及其编程语言 LAD、FDB 和 STL的补充和扩展。

它无需自己创建任何函数，可以使用预制模块，如中央处组件操作系统中现有的系统功能或系统功能模块。

通过 S7-SCL 进行编程的模块可以与 STL、LAD 和 FDB 模块混合使用。也就是说，通过 S7-SCL 进行编程的模块可以调用 STL、LAD 或 FDB 中编程的其他模块。相应地，也可以在 STL、LAD 和 FDB 程序中调用 S7-SCL 模块。

即便编译过程无误，S7-SCL 的测试功能仍可在其中进行逻辑编程错误查找。

## 针对 S7-SCL 开发环境

为了更好的应用 S7-SCL，其开发环境不仅与 S7-SCL 的特定属性，也与 STEP 7 相匹配。该开发环境由编辑器/编译器和调试器构成。

**用于 S7-1500 的 S7-SCL**

调试器

编辑器/编译器

**编辑器/编译器**

S7-SCL 编辑器是一种可以编辑任意文本的文本编辑器。其主要用来创建并编辑用于 STEP 7 的模块。在输入过程中即可进行基本的语法检查，从而简化无错误编程过程。语法错误将以不同颜色显示。

编辑器具备以下功能：

* 以 S7-SCL 语言编程 S7 模块。
* 通过拖放操作便捷插入语言元素并调用模块。
* 编程过程中直接进行语法检查。
* 根据要求设置编辑器，例如通过对不同语言要素按照语法进行文字上色。
* 通过编译检查固定模块。
* 编译时显示出现的所有错误和警告。
* 定位模块中的出错位置，排除故障时可选择显示故障描述和相关说明。

**调试器**

S7-SCL 调试器可按照自动化系统 (AS) 中的顺序检查编程情况，从而找到可能的逻辑性错误。

S7-SCL 提供两种不同的测试模式：

* 持续监控
* 逐步监控

通过“持续监控”可以在模块内测试指令组。测试过程中变量和参数值将按时间顺序显示并定期更新（在可能情况下）。

“逐步监控”时将按照逻辑性编程顺序进行测试。可以按照一条条指令执行编程算法，并在结果窗口监控在此期间编辑的变量内容的更改情况。

“逐步监控”可用与否，取决于所使用的 CPU。它必须支持停止点的使用。本文档中使用的 CPU 不支持停止点。

# 任务

## 储存箱容量示例任务

第一部分，对储存箱容量计算进行编程。

## 示例任务扩展

第二部分，扩展任务，并对故障评估进行编程。

# 规划

储存箱形状为直立式圆柱体。将利用模拟传感器测量填充物位。进行第一次测试时物位值（单位：米）应符合标准。

全局参数（如储存箱直径和高度）应结构化存储在全局数据模块“Daten\_Tank”中。

用于填料计算进行的编程应写入“Calculate\_Volume”函数中，且此参数以米或升为单位。

## 全局数据模块“Data\_Tank”

全局参数应以多种结构存储在全局数据模块中。

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 数据类型 | 初始值 | 注释 |
| dimensions | STRUCT |  |  |
| height | REAL | 12.0 | 单位为米 |
| Diameter | REAL | 3.5 | 单位为米 |
| measured\_data | STRUCT |  |  |
| filling\_leve\_per | INT | 0 | 值在 0 - 27648 之间 |
| filling\_level\_scal | REAL | 0.0 | 值在 0 - 12.0 之间 |
| volume\_liquid | REAL | 0.0 | 储存箱容量，单位：升 |
| fault\_flags | STRUCT |  |  |
| Calculate\_Volume | BOOL |  | 故障状态下 = 正确 |

表 1：数据模块“Data\_Tank”参数

## “Calculate\_Volume”函数

该模块以升为单位计算储存箱容量。

第一步不对所传输参数的合理性进行检查。

进行第一步时需要以下参数：

|  |  |  |
| --- | --- | --- |
| Input | 数据类型 | 注释 |
| Diameter | REAL | 圆柱形储存箱直径，单位：米 |
| Filling\_level | REAL | 储存箱物位，单位：米 |
| Output |  |  |
| Volume | REAL | 圆柱形储存箱容量，单位：升 |

表 2：用于第一步的函数“Calculate\_Volume”的参数

使用以下公式计算直立式圆柱体容积。使用换算系数 1000，以升为单位计算结果。

2

 => 

## “Calculate\_Volume”扩展函数

第二步检查直径是否大于零。此外，应测试物位是否大于等于零或等于储存箱高度。在错误情况下，将新参数“er”设为 TURE，参数“容量”获得值 -1。

将接口扩展，加入参数“er”和“Height”。

|  |  |  |
| --- | --- | --- |
| Input | 数据类型 | 注释 |
| height | REAL | 圆柱形储存箱高，单位：米 |
| Diameter | REAL | 圆柱形储存箱直径，单位：米 |
| Filling\_level | REAL | 储存箱物位，单位：米 |
| Output |  |  |
| er | BOOL | 故障标记；故障时 = TURE |
| Volume | REAL | 圆柱形储存箱容量，单位：升 |

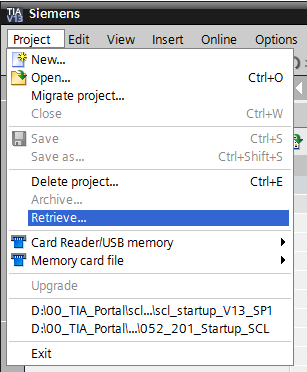
表 3：用于第二步的“Calculate\_Volume”函数的参数

# 结构化的逐步式引导指南

以下为实现编程规划的指南。如果您已经掌握了相关的知识，只需要使用带标号的步骤标题作为参考。否则，也可以简单地跟随指南中的图示一步步操作。

## 现有项目归档

* 开始编程之前，我们需要带有硬件配置的项目（例如 SCE\_DE\_012-101\_   
  Hardwarekonfiguration\_S7-1516F\_....zap）。现有项目归档时必须从 → 项目 → 归档下的项目视图中查找相应文档。接着点击打开确认选择。   
  （ → 项目 → 归档 → 选择 .zap-Archivs → 打开）

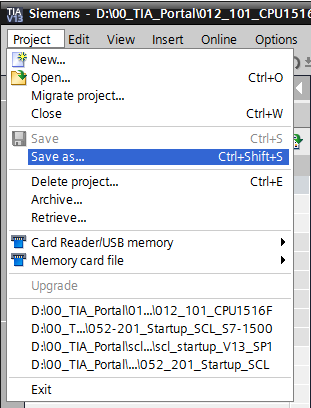


* 接着可以选择保存归档项目的目标目录。点击确定 (OK) 确认选择。  
  （ → 项目 → 保存在 → OK）

## 将项目保存在新名称下

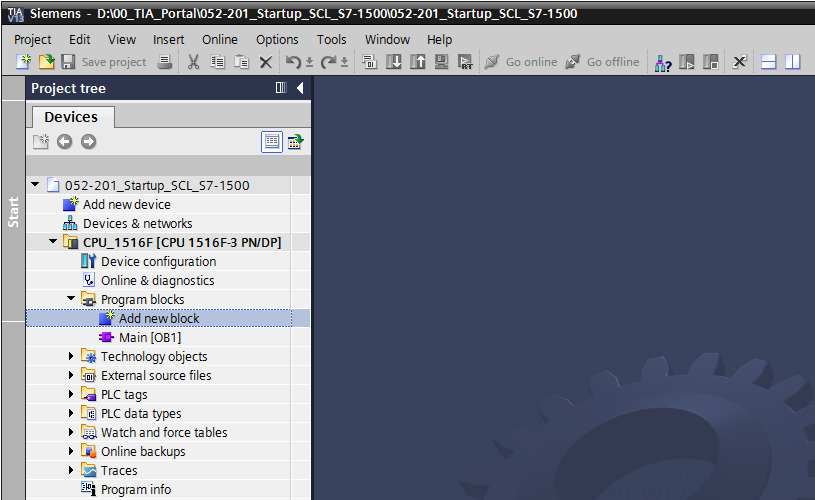
* 将打开的项目保存在名称 052-201\_Startup\_SCL 下。

（ → 项目 → 保存在 … → 052-201\_Startup\_SCL → 保存）

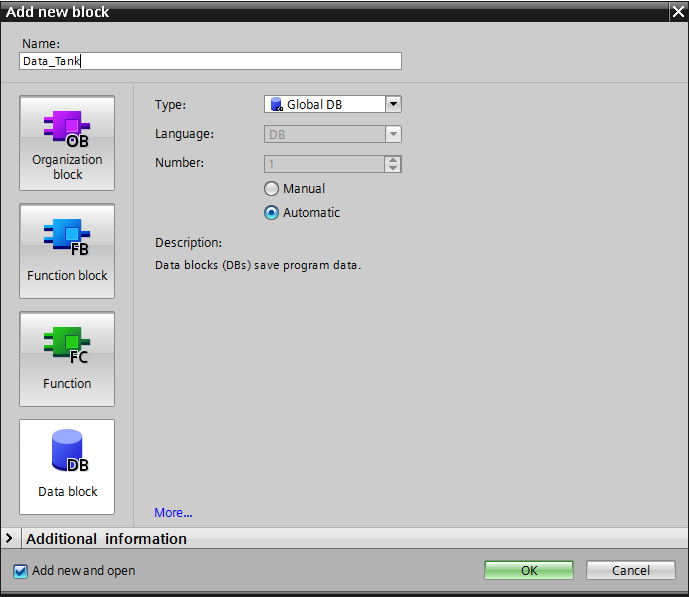


## 创建数据模块“Data\_Tank”

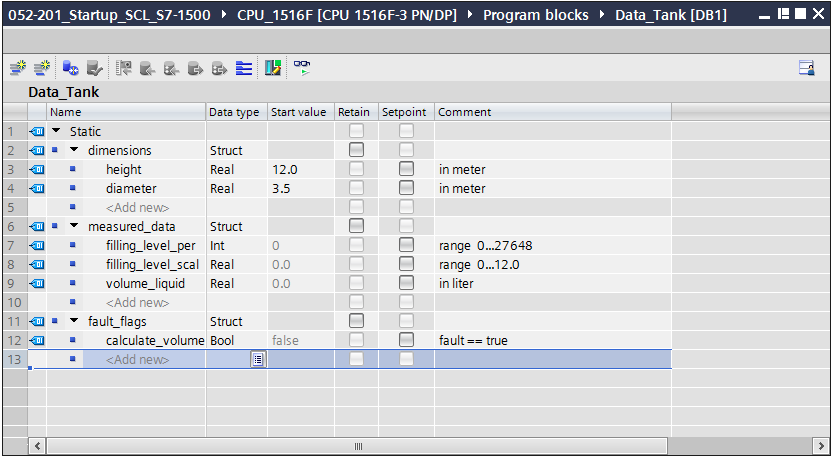
* 在项目视图中导航至 → 项目模块，并通过双击添加 → 新模块，创建新模块。



* 现在请选择数据模块并输入名称。（ →→“Data\_Tank”→ OK）



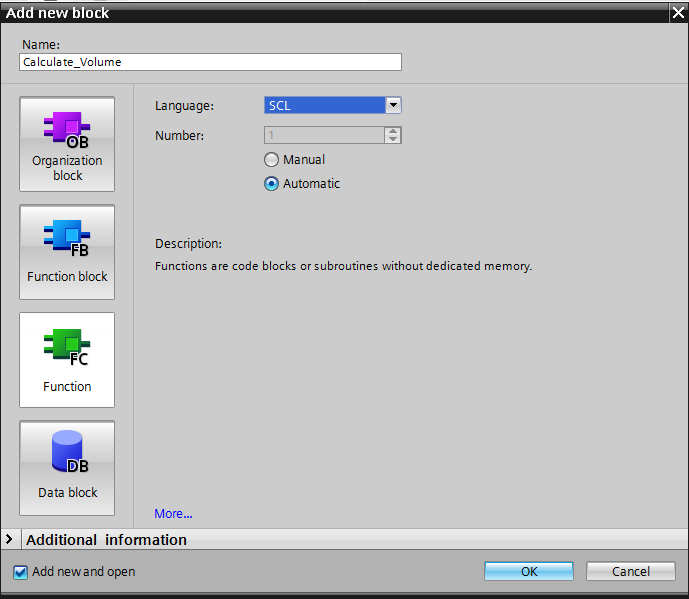
* 现在请输入以下规定的变量名称与数据类型、初始值和注释。



## 创建函数“Calculate\_Volume”

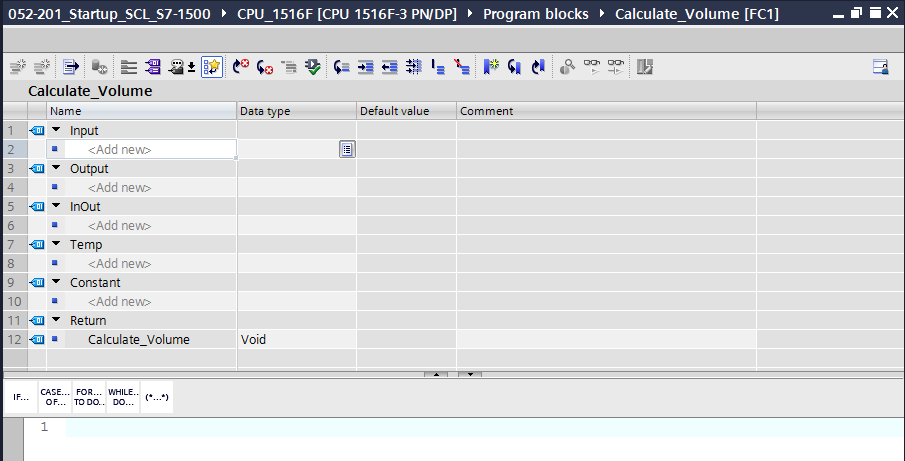
* 现在请添加函数，输入名称并选择语言。

（ → 添加新模块 →  →“Calculate\_Volume”→ SCL →OK）

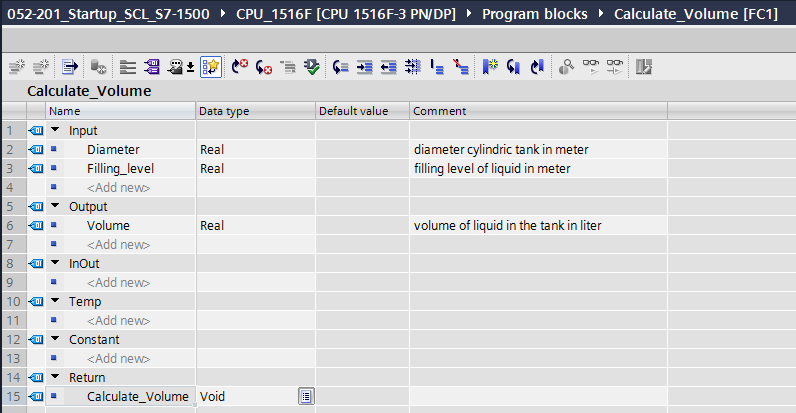


## 确定函数“Calculate\_Volume”的接口

* 在上一节编程视图查找函数接口描述。

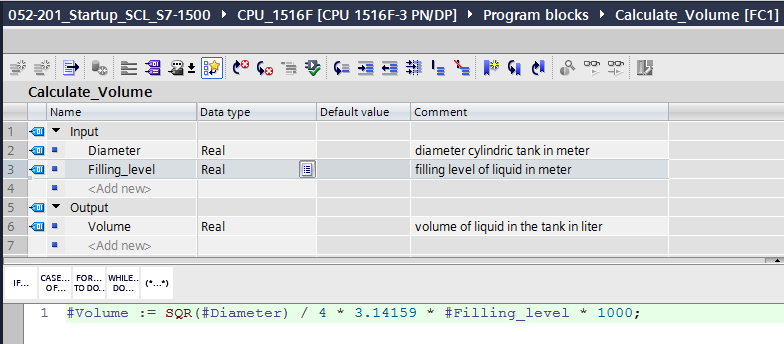


* 创建以下输入和输出参数。（ → 名称 → 数据类型 → 注释）

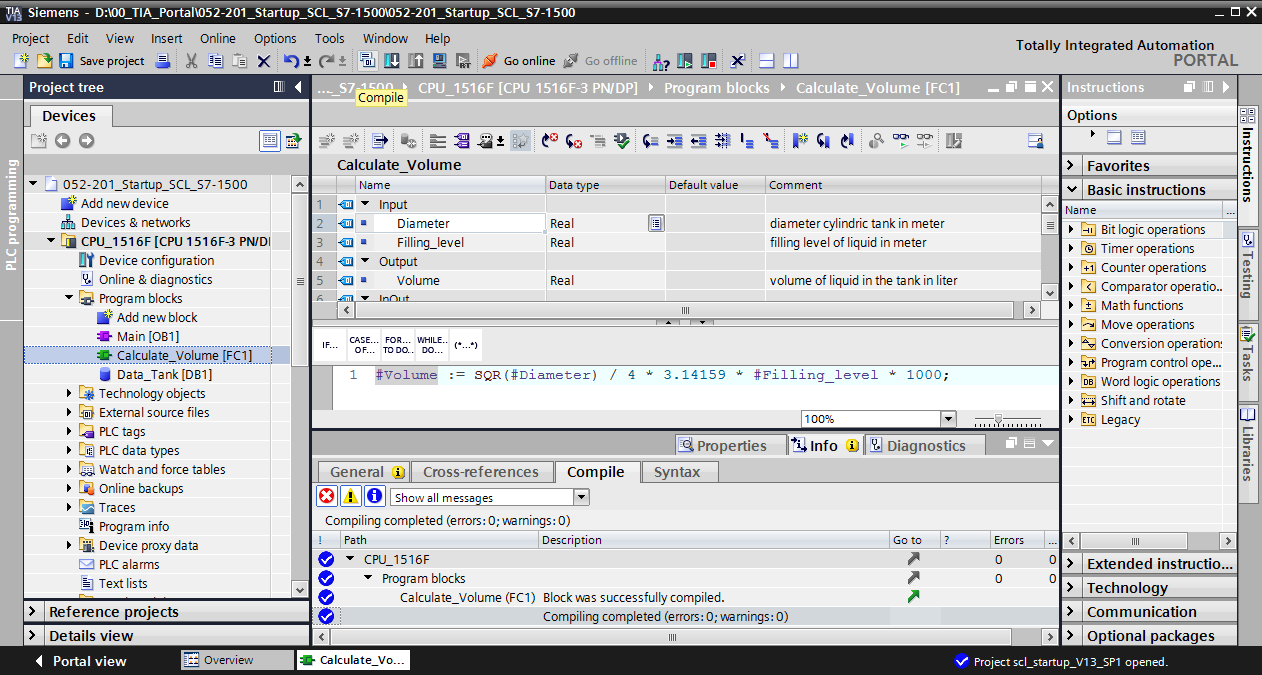


## 函数“Calculate\_Volume”编程

* 输入下列程序。（ → 输入程序）



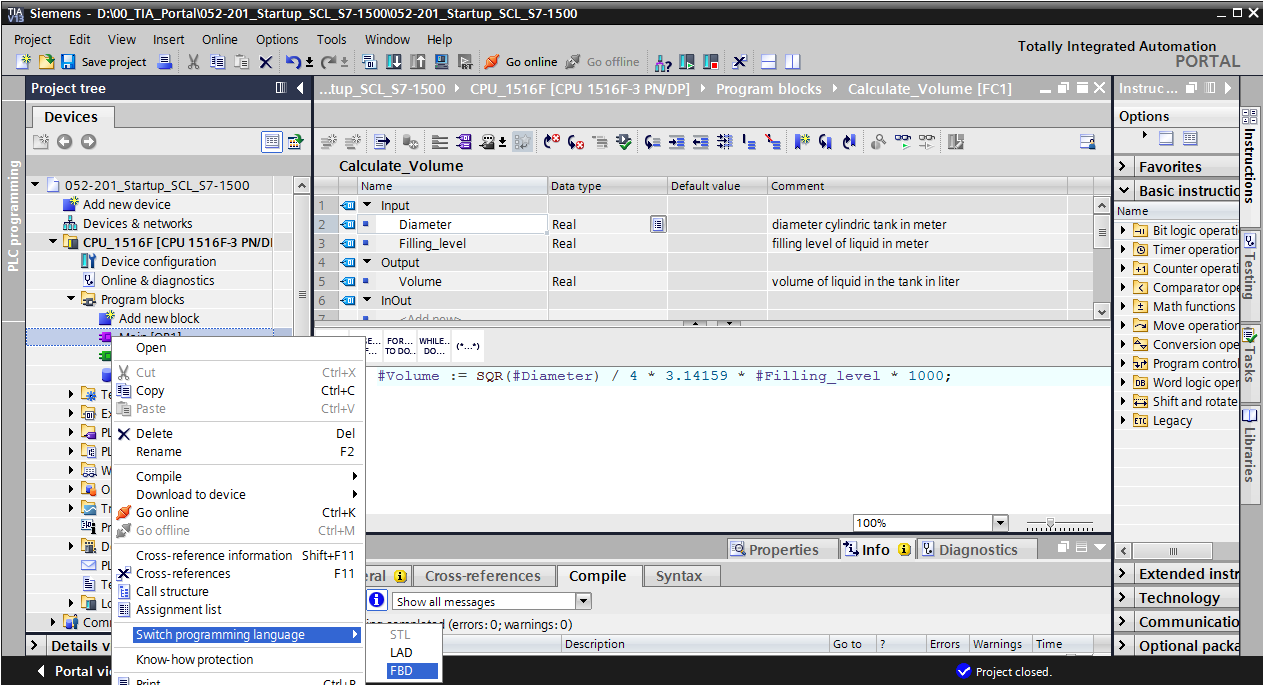
* 现在编译程序并检查是否有语法错误。错误情况将显示在编程下方的检查窗格中。必要时排除错误，并重新编译。之后保存程序。   
  （ →  → 排除错误 → ）



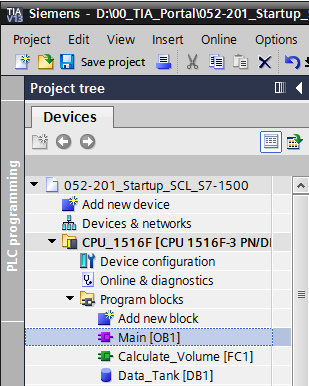


## 组织模块“Main [OB1]”编程

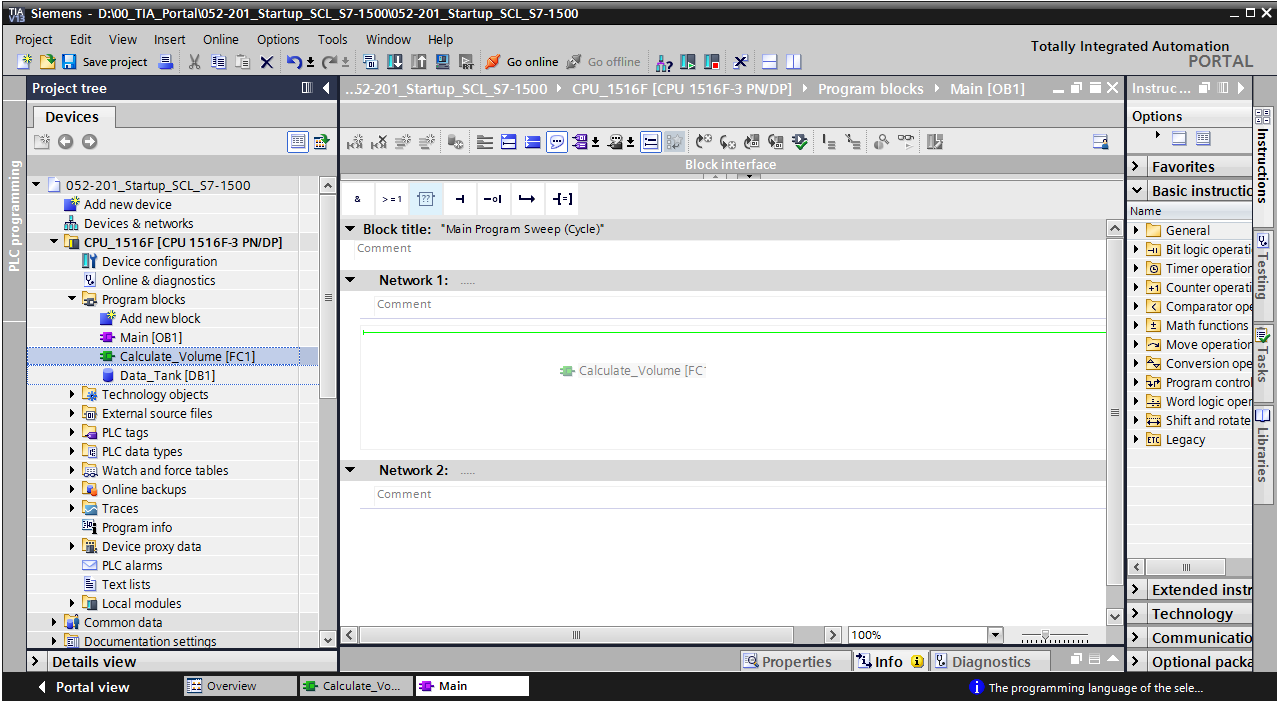
* 组织模块“Main [OB1]”编程之前将程序语言转换为 FDB。为此需用鼠标左键点击“程序块”文件夹中的“Main [OB1]”。（ → CPU\_1516F[CPU 1516F-3 PN/DP] → 程序模块 → Main [OB1] → 切换程序语言 → FUP）

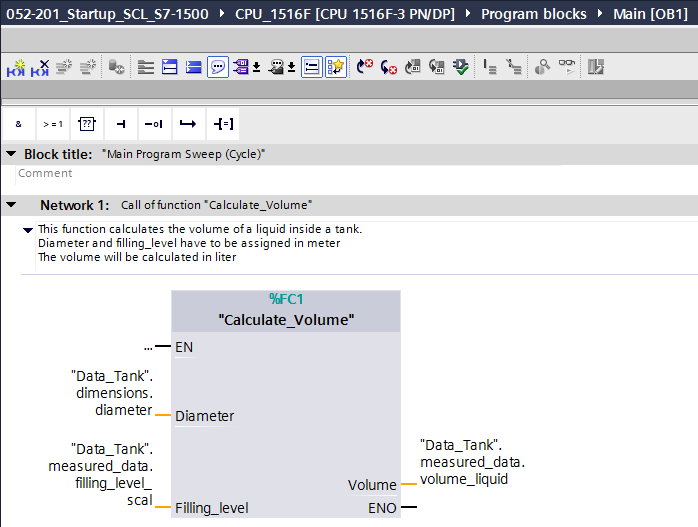


* 现在请双击打开组织模块“Main [OB1]”。



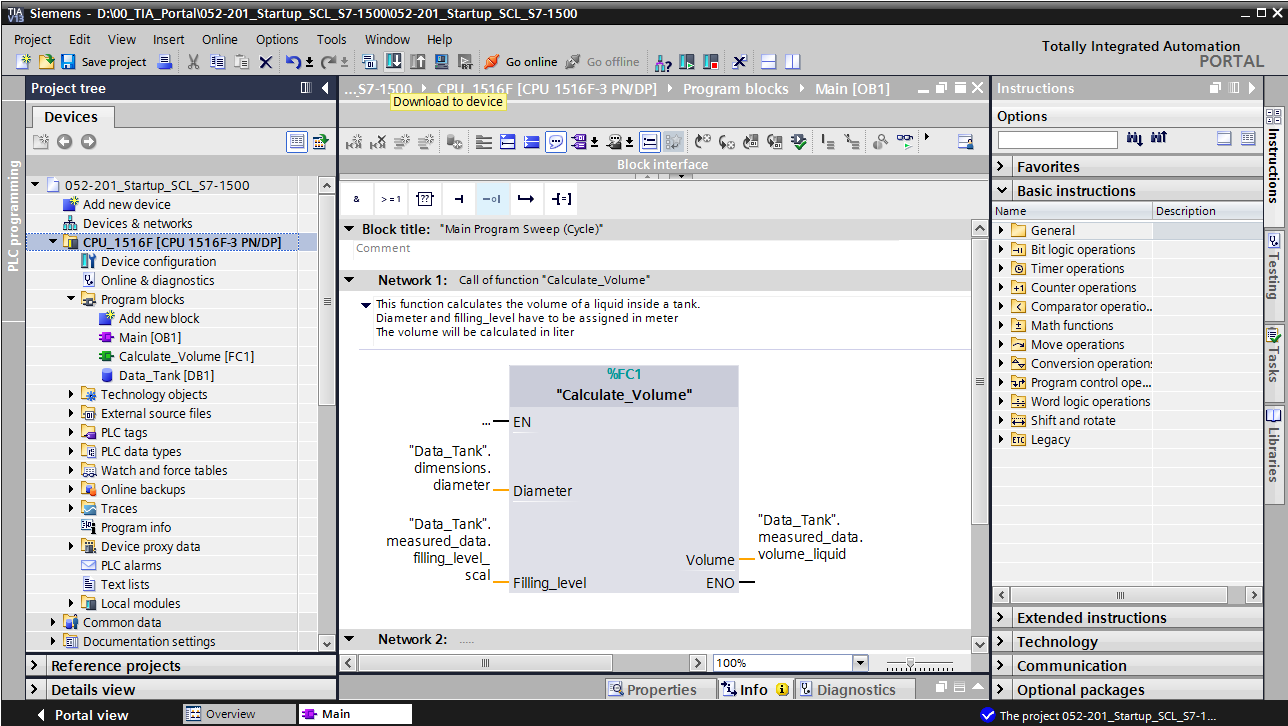
* 调用第一个网络中的函数“Calculate\_Volume”。分配网络标题、注释并连接参数。之后保存项目。 （ → 调用“Calculate\_Volume”→ 分配网络标题 → 写入网络注释 → 连接参数 → ）



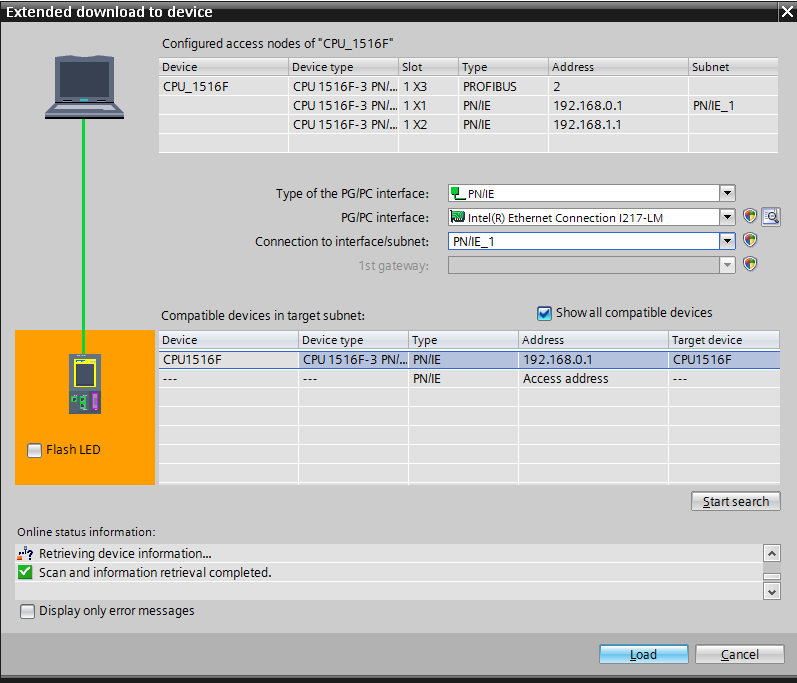


## 编译程序并加载

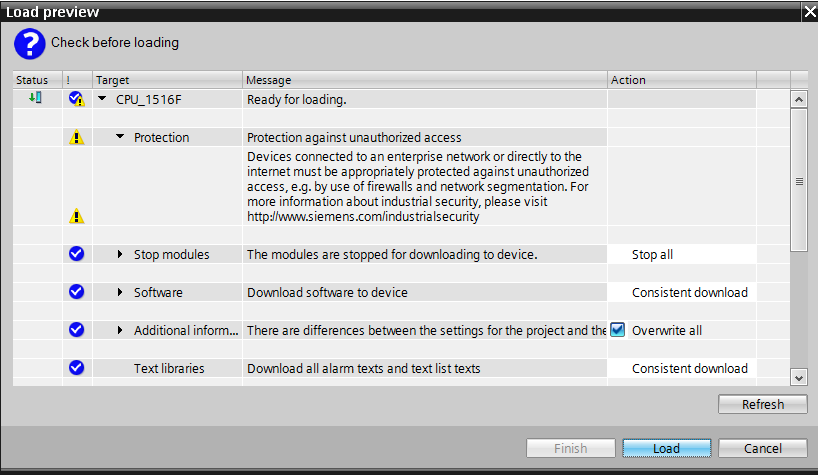
* 点击“程序模块”文件夹，并编译整个程序。编译成功后将项目加载到控制系统中。  
  ( →  →  )



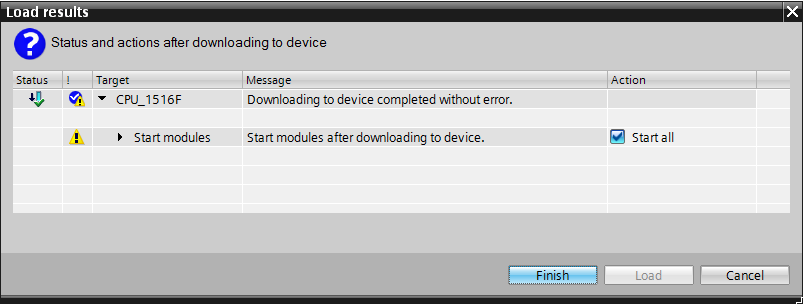
* 选择 PG/PC 接口 → 选择子网络 → 启动搜索 → 加载



* 有条件选择 → 加载

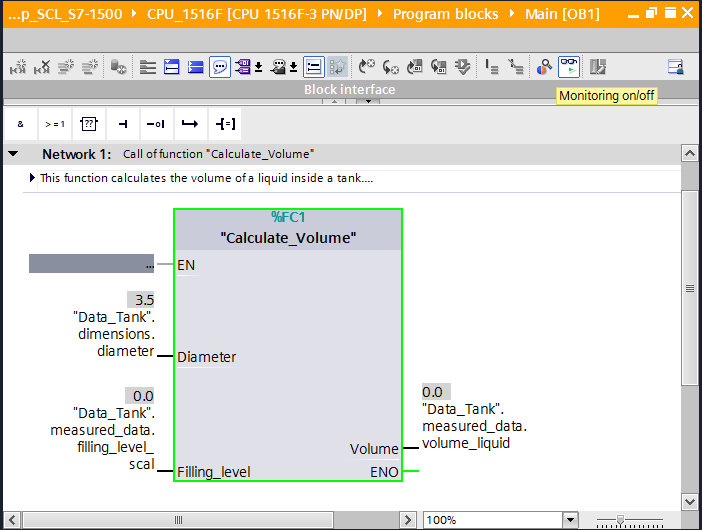


* 项目完成

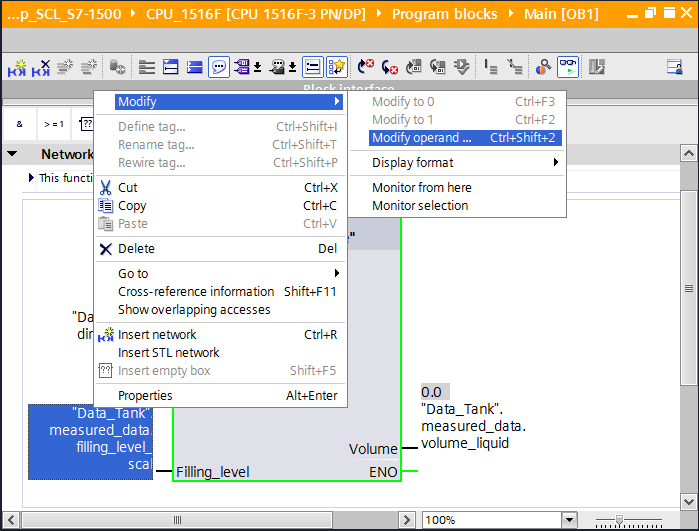


## 监控并测试组织模块

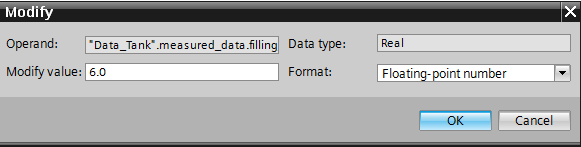
* 在打开的 OB1 上点击图标 D:\00_DATA\SIEMENS\Unterlagen\08_Ausbildungsunterlage_TIA-Portal_R1502_dt\032-100 FC-Programmierung\pics\055b.jpg，监控模块。



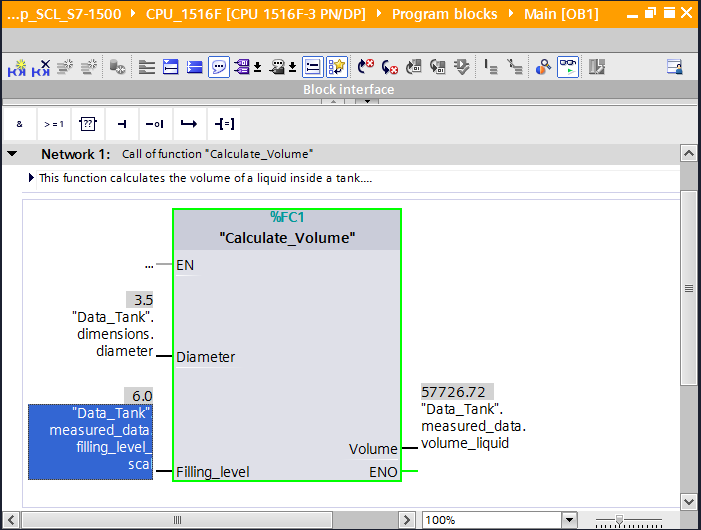
* 将值写入数据模块上的变量“Filling\_level\_scal”中，测试程序。  
  （ → 右击“Filling\_level\_scal”→“控制”菜单 → 控制运算数）



* 输入值 6.0 → OK

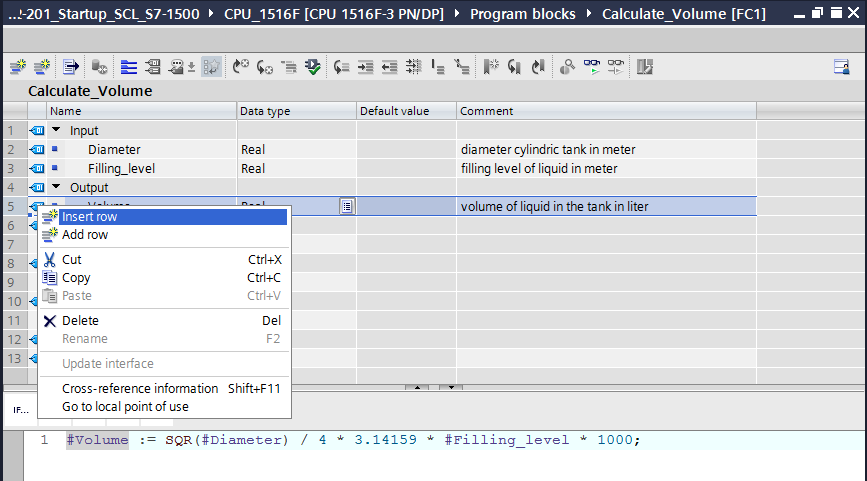


* 检查结果正确性。

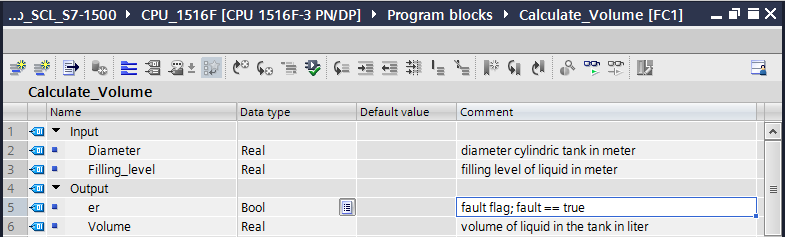


## “Calculate\_Volume”函数扩展

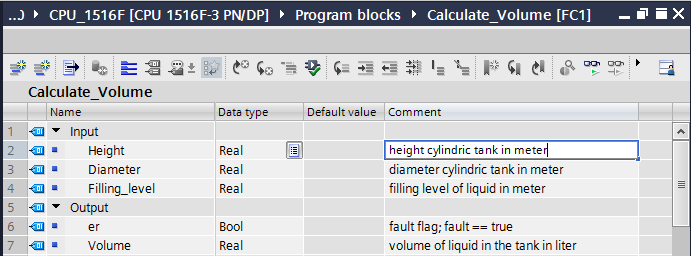
* 打开函数“Calculate\_Volume”并右击接口中的行，在输出参数中添加行。  
  （ → 打开“Calculate\_Volume”→ 右击第 5 行 → 添加行）



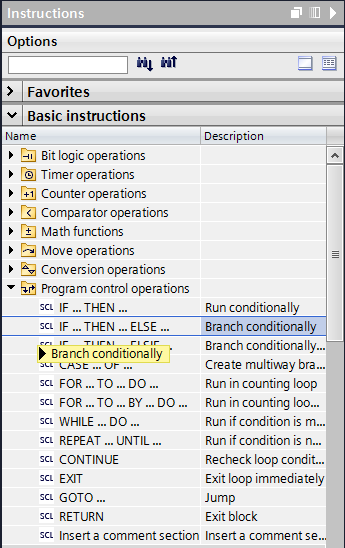
* 输入参数“er”与数据类型 BOOL 和注释。



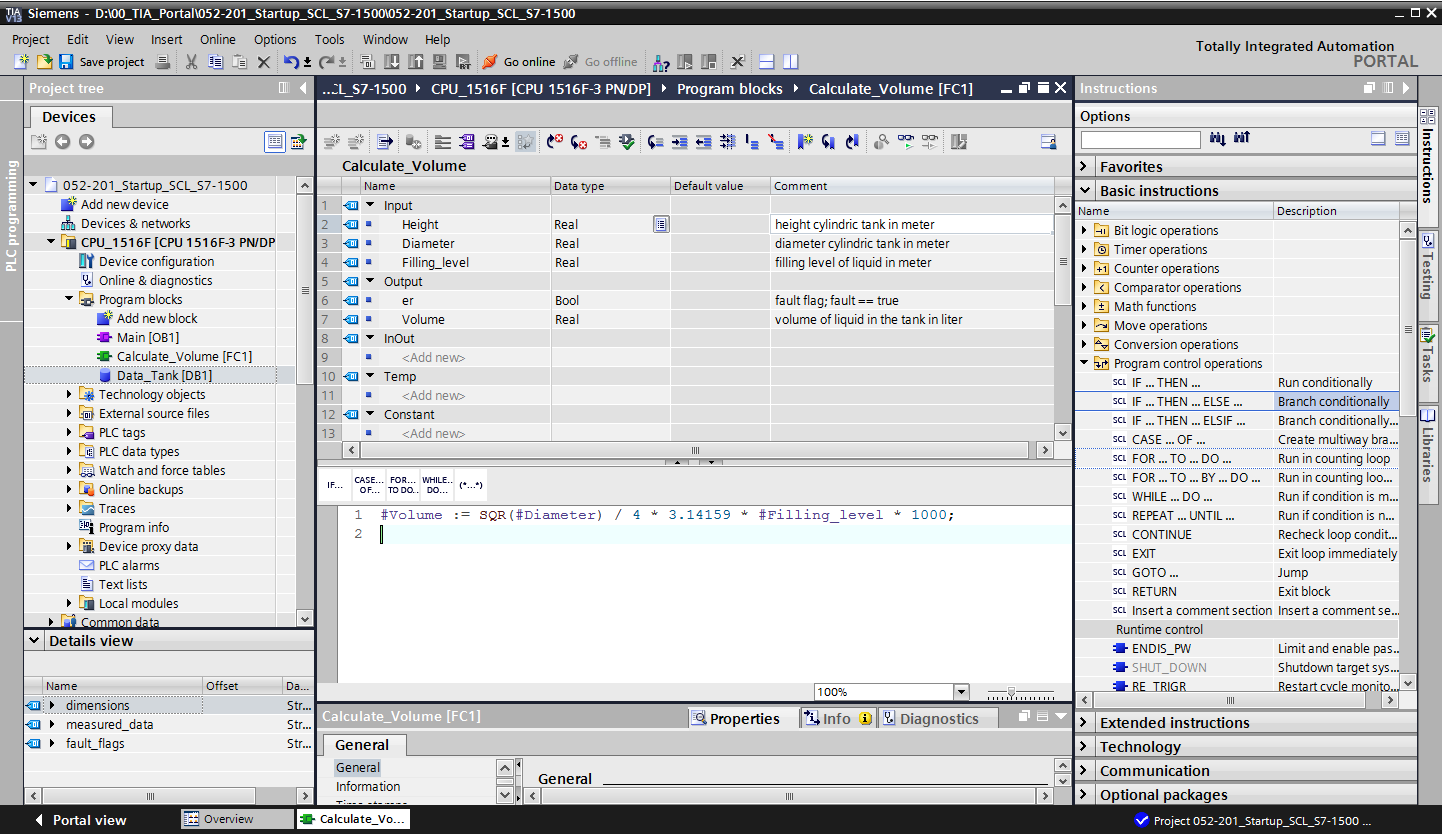
* 之后以相同方式添加变量“height”与数据类型 Real 和注释。

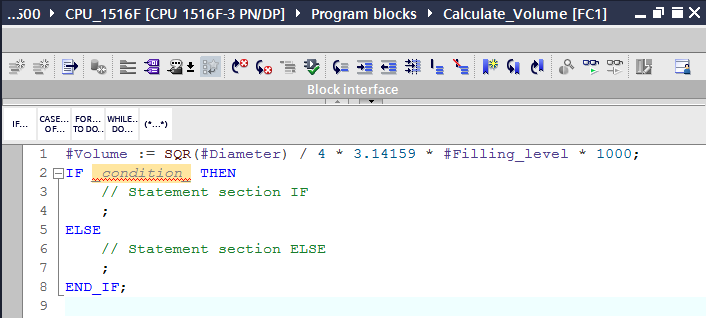


* 然后从基础指令文件夹“程序控制”导航至检查结构“IF…THEN…ELSE”。  
  （ → 指令 → 基础指令 → 程序控制 →“IF...THEN…ELSE”）

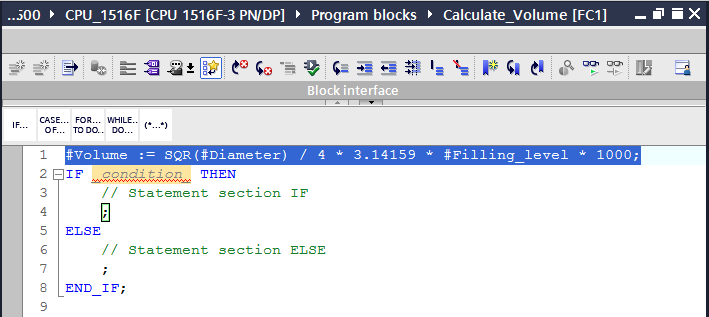


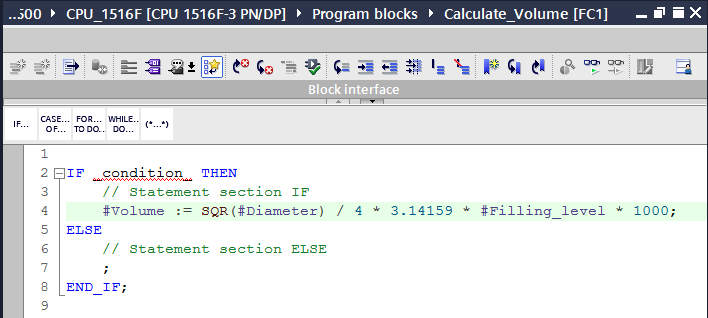
* 接着通过拖放操作将检查结构“IF...THEN...ELSE”移动至程序第二行。（→“IF…THEN…ELSE”→ 拖放）



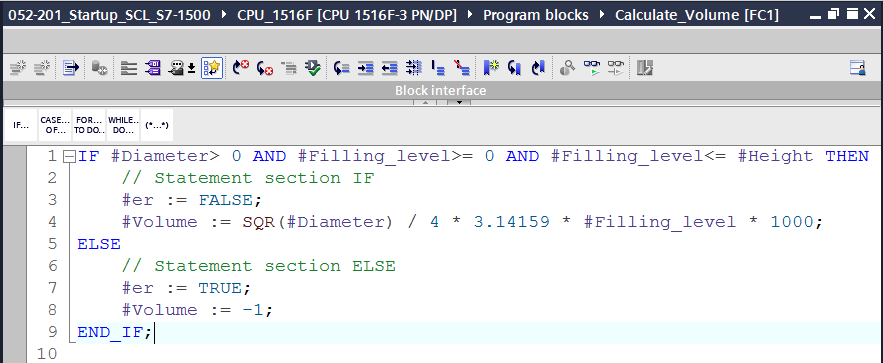


* 标记数学公式，并将其拖放至 ELSE 前的分号处。（→ 标记 → 拖放）

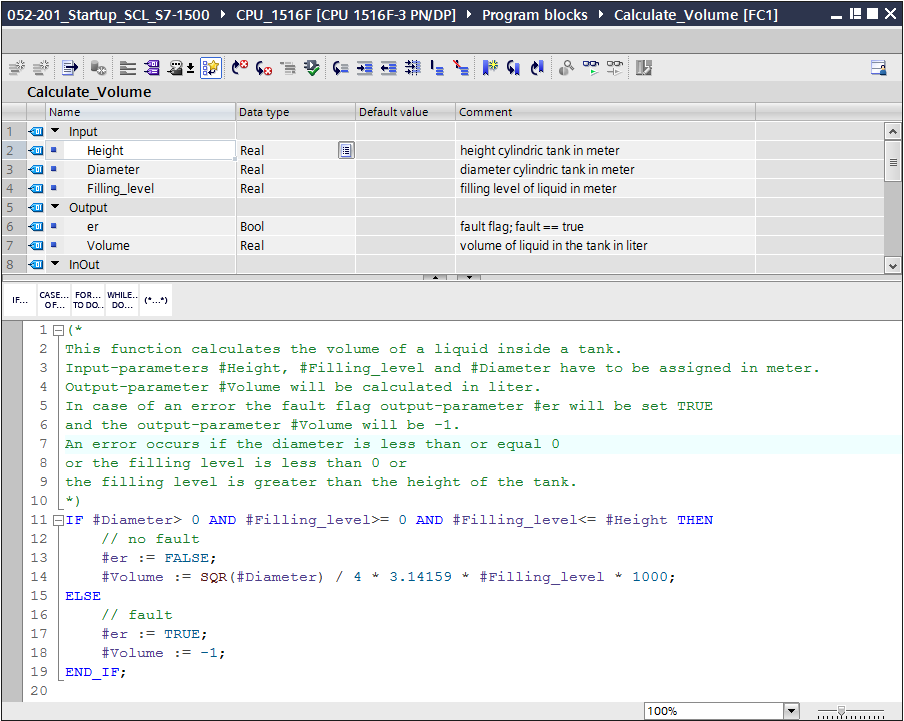




* 完成函数并检查编译的程序。（ → 补充程序 → ）

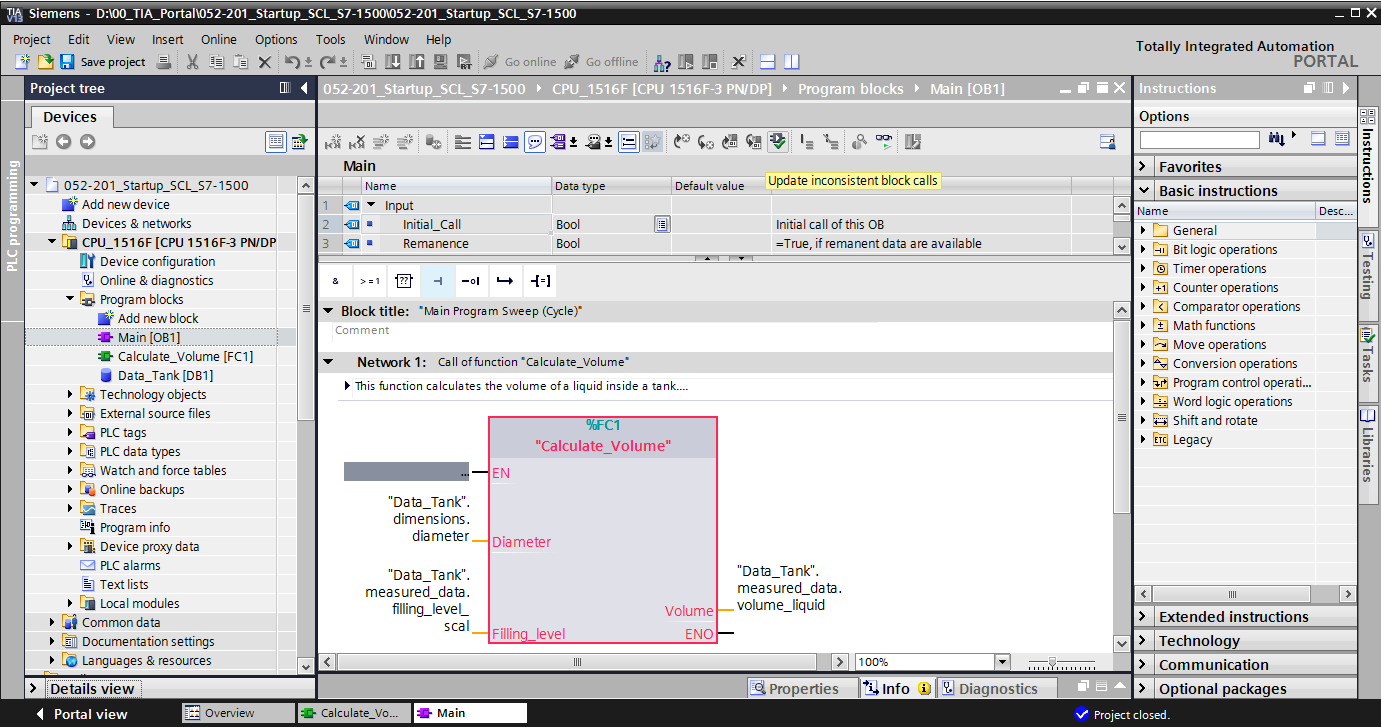


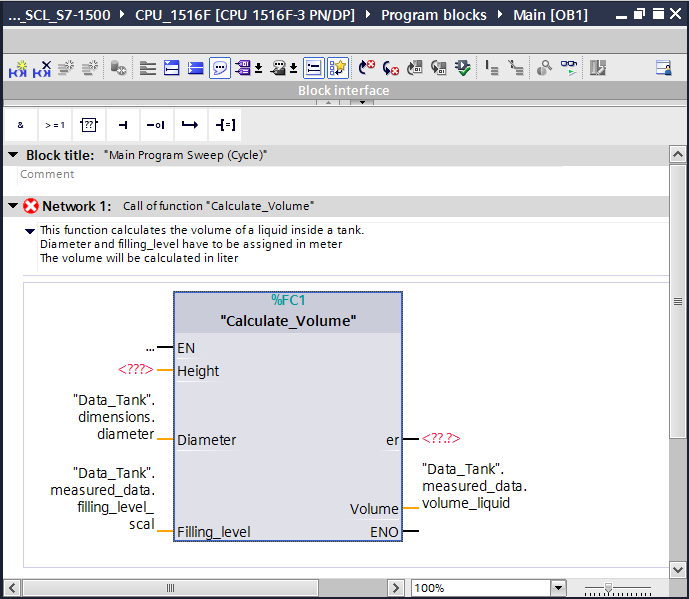
* 可以添加带“(\*\*)”的块注释，和带“//”的行注释。现在可以添加注释补充程序。  
  （ → 从第 1 行开始添加块注释 → 在第 12 行和第 16 行添加行注释）



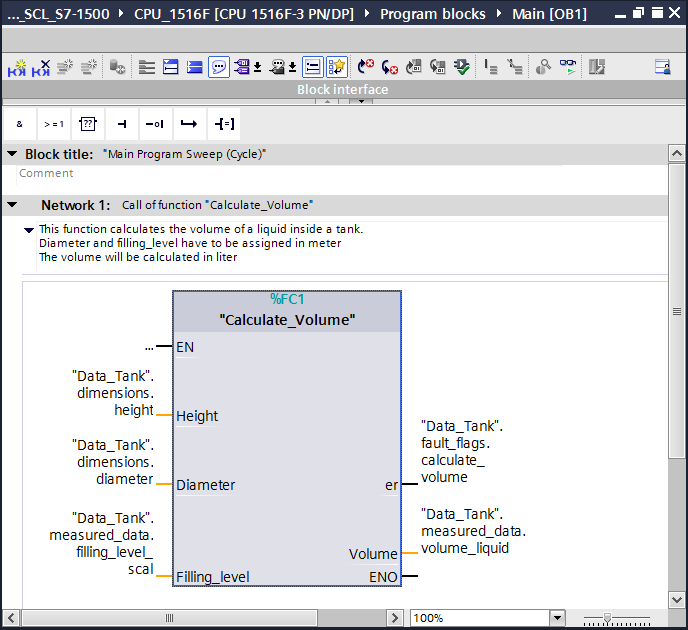
## 调整组织模块

* 打开 OB1 并点击 ，更新不一致的模块调用。（ → 打开 OB1 → ）





* 扩展参数“er”和“height”。



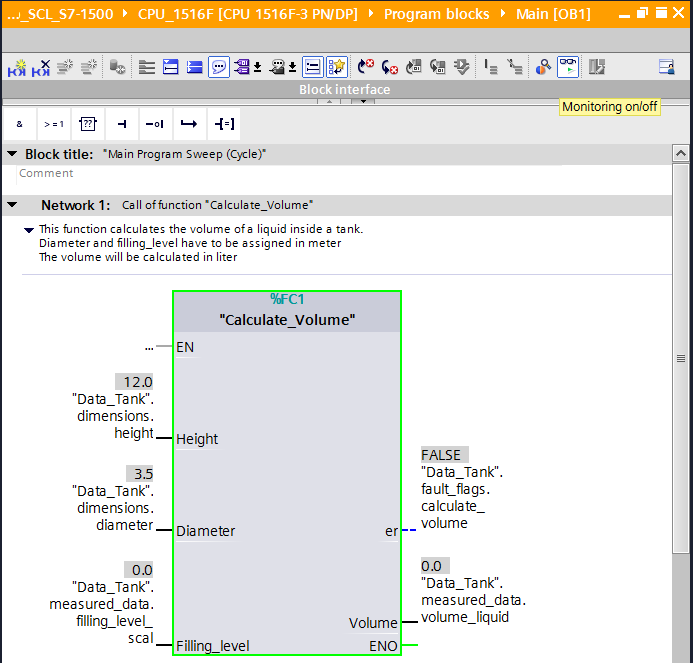
## 编译程序并加载

* 点击“程序模块”文件夹，并编译整个程序。编译成功后将项目加载到控制系统中。之后保存项目。 （ → 程序模块 →  →  → ）



## 监控并测试组织模块

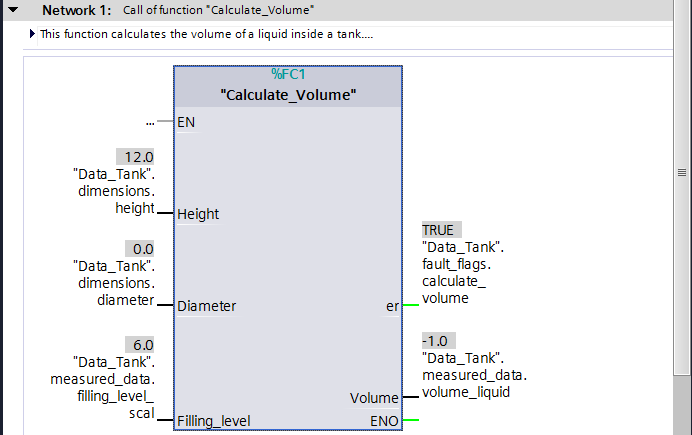
* 在打开的 OB1 上点击图标 D:\00_DATA\SIEMENS\Unterlagen\08_Ausbildungsunterlage_TIA-Portal_R1502_dt\032-100 FC-Programmierung\pics\055b.jpg，监控模块。



* 将值写入数据模块上的变量“Filling\_level\_scal”中，测试程序。  
  （ → 右击“Filling\_level\_scal”→“控制”菜单 → 控制运算数 → 输入值 6.0 → OK → 检查）

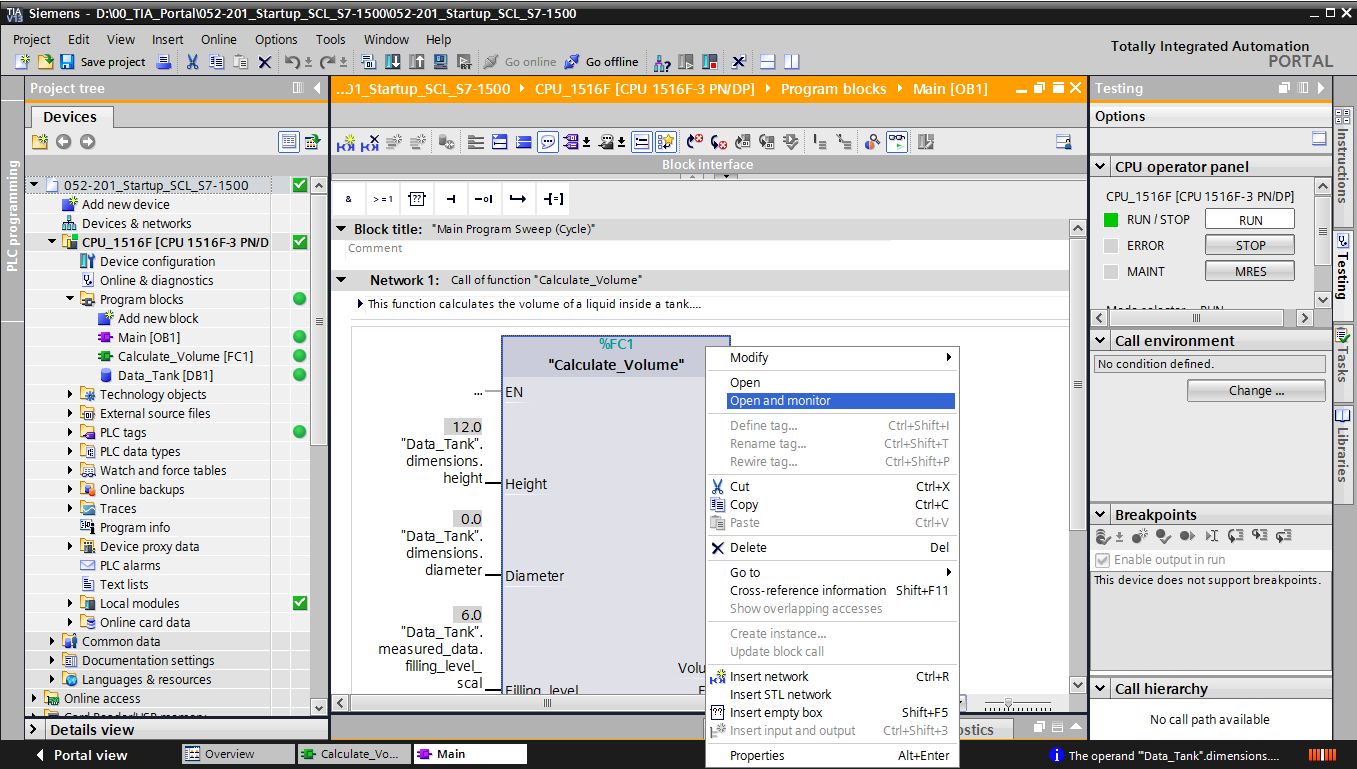


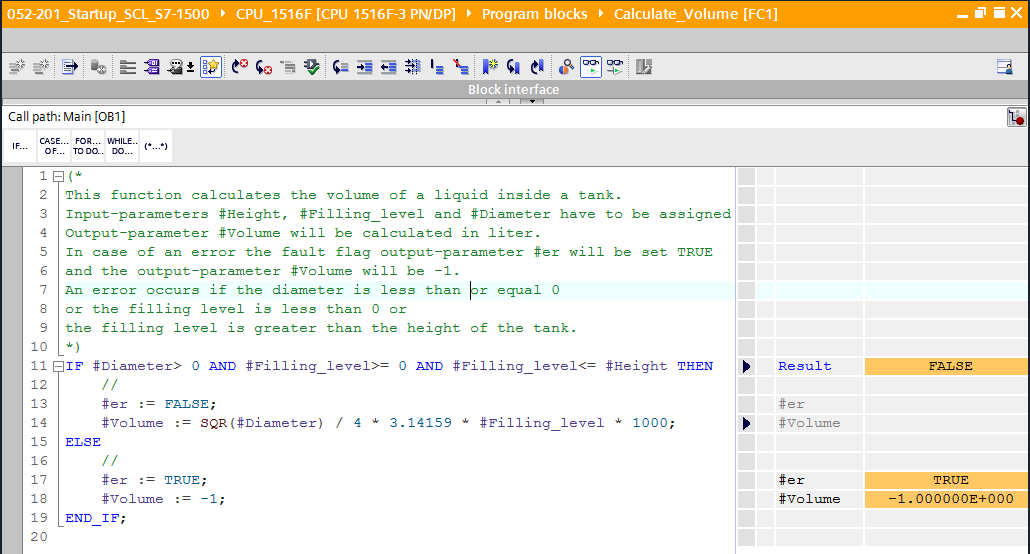
* 之后将直径设为零，测试是否输出错误。  
  （ → 右击“直径”→“控制”菜单 → 控制运算数 → 输入值 0.0 → OK → 检查）



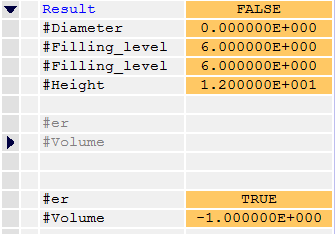
## 监控并测试“Calculate\_Volume”函数

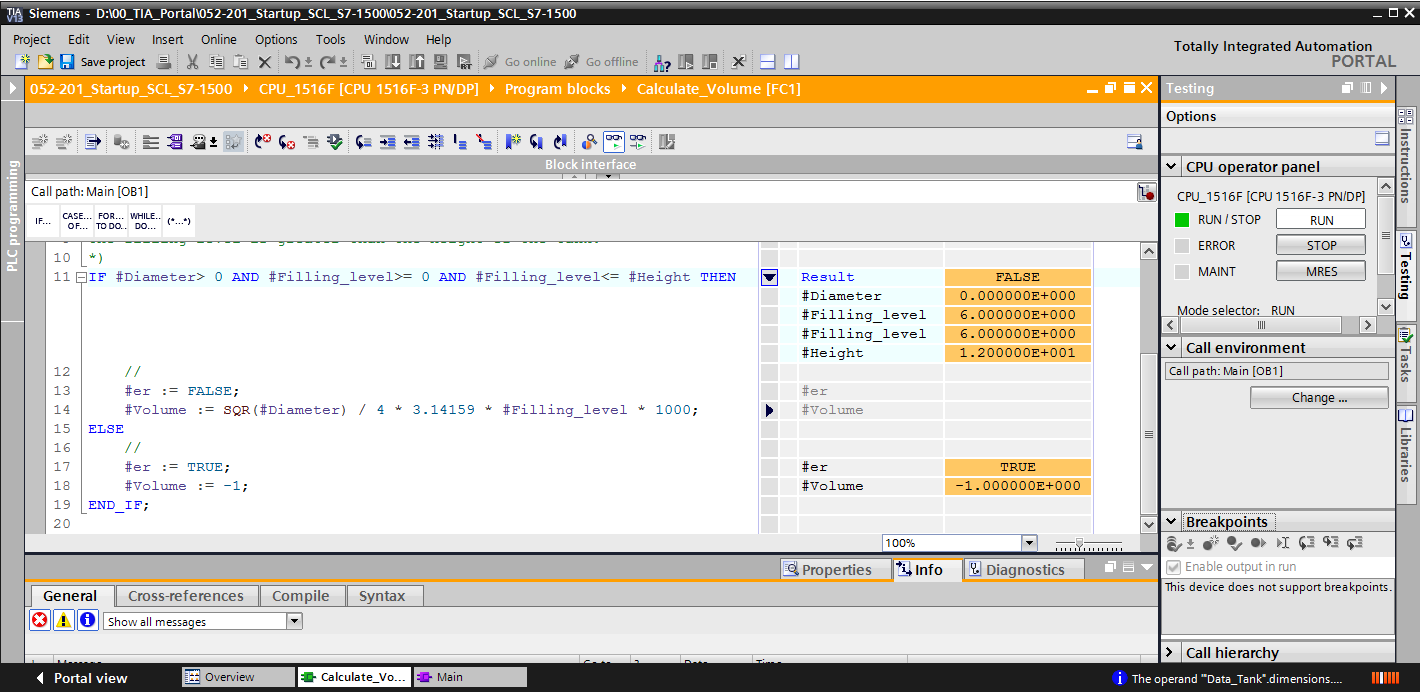
* 最后，右击函数，选择“打开并监控”菜单项，打开并监控“Calculate\_Volume”。  
  （→ 右击函数 → 打开并监控）



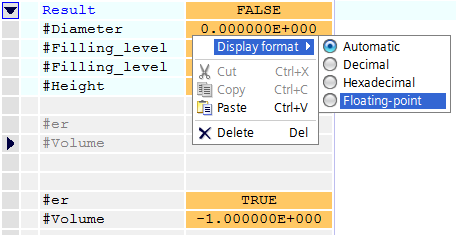


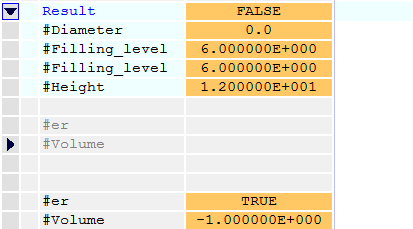
* 可以点击黑色箭头 ，显示单个 IF 查询变量值。( →  )



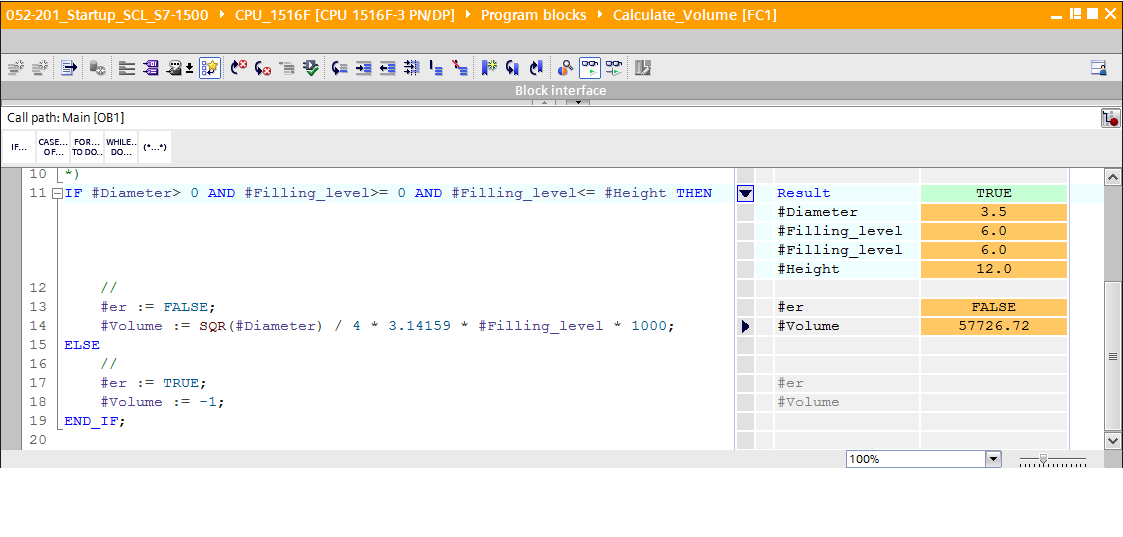


* 可以右击变量调整显示格式。（ → 右击变量 → 显示格式 → 滑动点）



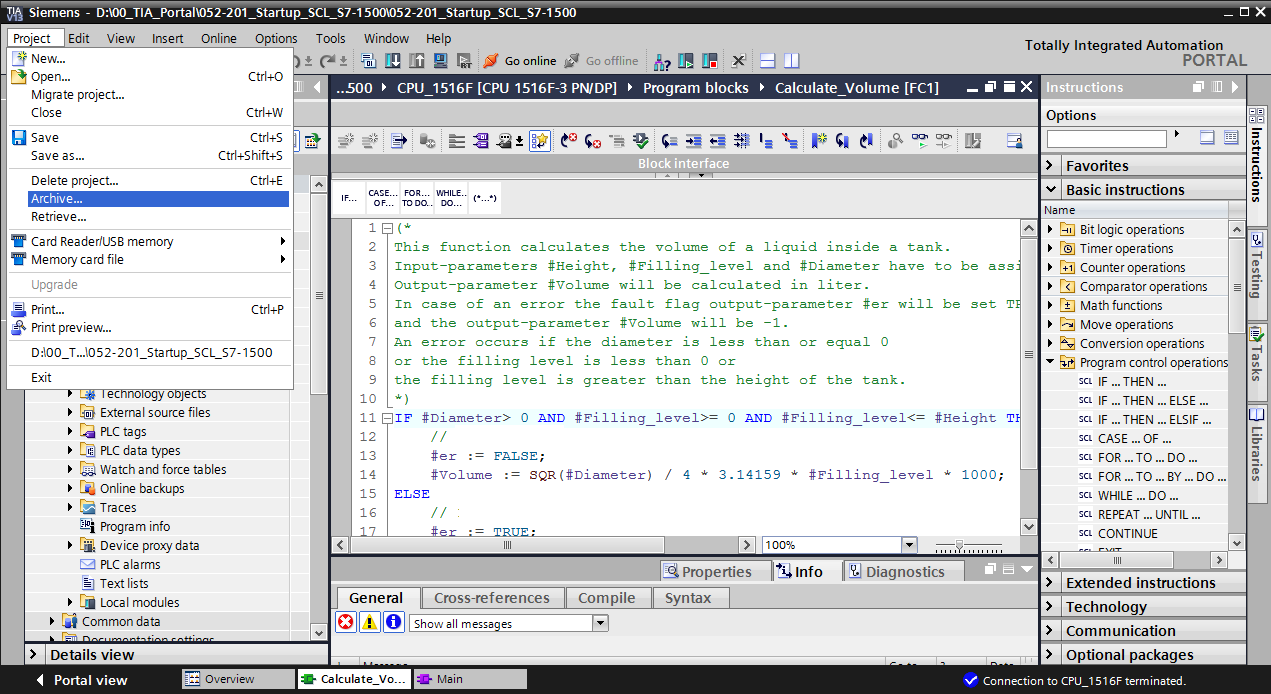


* 现在将 OB1 中的直径重新控制在 3.5 米，以测试 IF 分支的其他分路。  
  （ → 打开 OB1 → 将直径控制在 3.5 米 → 打开并监控函数）



## 项目成功归档

* 最后应将整个项目进行归档。在菜单项 →‘项目’→‘中选择归档 ...’。打开需要归档项目的文件夹，并保存为文件格式‘TIA Portal 项目归档’。（ → 项目 → 归档 → TIA Portal 项目归档 → SCE\_DE\_052-201 Startup SCL\_S7-1500… → 保存）



# 检查清单

|  |  |  |
| --- | --- | --- |
| **编号** | **描述** | **已检查** |
| 1 | 编译成功，无错误提示 |  |
| 2 | 加载成功，无错误提示 |  |
| 3 | 控制运算数（Diameter = 0.0）  结果：变量 Volume = -1  结果：变量“er”= TRUE |  |
| 4 | 控制运算数（Diameter = 3.5，Filling\_level\_scal = 0）  结果：Volume = 0  结果：变量“er”= FALSE |  |
| 5 | 控制运算数（Filling\_level\_scal = 6.0）  结果：Volume = 57726.72  结果：变量“er”= FALSE |  |
| 6 | 控制运算数（Filling\_level\_scal = 12.0）  结果：Volume = 115453.4  结果：变量“er”= FALSE |  |
| 7 | 控制运算数（Filling\_level\_scal = 14.0）  结果：Volume = -1  结果：变量“er”= TRUE |  |
| 8 | 项目成功归档 |  |

# 练习

## 任务 - 练习

本练习中将对“测量”函数进行编程。该程序通常应用于任何正模拟值。在我们的示例任务“储存罐”中将通过模拟传感器读取物位，并通过扩展函数保存在数据块中。

在错误情况下，该模块将错误标记“er”设为 TRUE，并将参数“Analog\_scal”结果设为零。如果参数“mx”小于或等于“mn”，程序报错。

该函数必须包含以下参数。

|  |  |  |
| --- | --- | --- |
| **Input** | **数据类型** | **注释** |
| Analog\_per | INT | 外围设备模拟值，在 0 - 27648 之间 |
| mx | REAL | 新标尺的最大值 |
| mn | REAL | 新标尺的最小值 |
| **Output** |  |  |
| er | BOOL | 错误记录，无错误 = 0，错误 = 1 |
| Analog\_scal | REAL | 模拟值，在 mn - mx 之间  错误状态下 = 0 |

完成作业时将使用以下公式：



本练习任务需要模拟信号。必须将对此所使用的操作数输入 PLC 变量列表中。

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **数据类型** | **地址** | **注释** |
| B1 | INT | %EW64 | 物位，在 0 - 27648 之间 |

## 规划

现在请自主执行任务！

## 检查清单 - 练习

|  |  |  |
| --- | --- | --- |
| **编号** | **描述** | **已检查** |
| 1 | 已将操作数插入 PLC 变量列表中 |  |
| 2 | FC 函数：已创建“Scaling” |  |
| 3 | 已定义接口 |  |
| 4 | 已编程函数 |  |
| 5 | 已将函数“Scaling”插入 OB1 网络 1 中 |  |
| 6 | 已连接输入变量 |  |
| 7 | 已连接输出变量 |  |
| 8 | 编译成功，无错误提示 |  |
| 9 | 加载成功，无错误提示 |  |
| 10 | 物位模拟值设为零  结果：Filling\_level\_scal = 0  结果：er = FALSE |  |
| 11 | 物位模拟值设为 27648  结果：Filling\_level\_scal = 12.0  结果：er = FALSE |  |
| 12 | 物位模拟值设为 13824  结果：Filling\_level\_scal = 6.0  结果：er = FALSE |  |
| 13 | 控制运算数（mx = 0.0）  结果：Filling\_level\_scal = 0  结果：变量 er = TRUE |  |
| 14 | 项目成功归档 |  |

# 更多相关信息

在进行入门培训或深化学习时，可查找其他信息作为指导，例如：入门指南、视频、教程、应用程序、手册、编程操作手册和试用软件/固件，链接如下：   
  
[www.siemens.com/sce/s7-1500](http://www.siemens.com/sce/s7-1500)