

博途官方提供了很多库，比如：基本函数库、通信库、安全库、驱动库等等，用户可以使用库中的函数/函数块来完成具体的控制任务。除了官方的库，我们也可以创建自己的库（用户自定义库）。比如，把项目中经常使用的函数/函数块/数据类型等存放到自定义库中，方便自己使用及与别人共享。博途具有很强的库管理功能，包括：库版本管理，库的更新及清扫等等。本系列文章我将给大家介绍项目库、全局库、库的更新/清扫等功能，今天这篇文章，我们先来介绍项目库。

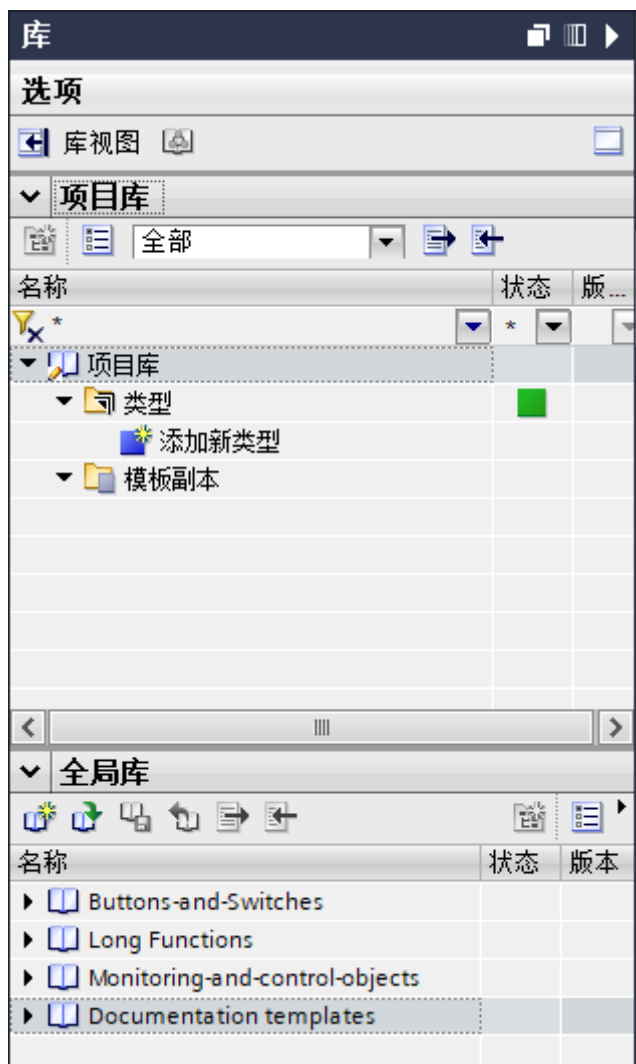


## 博途自定义库的使用（1）

博途的库可分为两类：项目库和全局库。

- 项目库：顾名思义，与具体的项目相关联。打开任意一个博途项目，在其右侧的“库”任务卡中都有一个“项目库”的窗格（虽然它可能是空的）。项目库随着博途项目的打开而打开，随着项目的关闭而关闭，它没有单独的存放路径；
- 全局库：顾名思义，在全局范围可用，它并不属于某个项目。全局库有单独的存放路径，可被打开/关闭，也能像博途项目一样归档（后续文章详细介绍）；

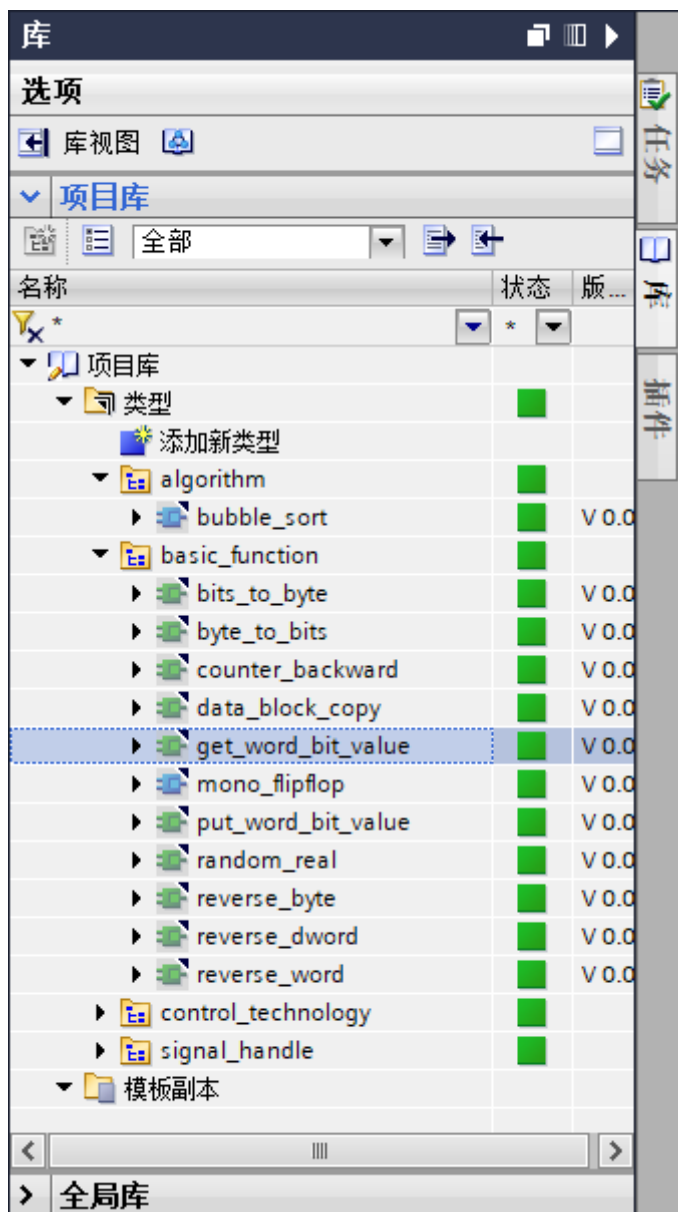
项目库和全局库在“库”任务卡中的布局如下图所示：



在上图的【项目库】中，我们看到有两个节点：类型和模板副本。

## 1、类型 (types)

类型是一种模板，在其中可以存放函数/函数块/用户自定义数据类型。“模板”就像一个模具，由它复刻出的函数/函数块/用户自定义类型等被称为其“实例”。如果你有面向对象编程的基础，可以把这里的“类型”理解为面向对象编程的“类”，由它复刻的即为其“实例”。将项目中的函数/函数块/数据类型通过拖拽的方式放到【类型】节点就会自动创建其“类型”。反之，从【类型】节点中将元素拖放到项目中会自动创建其“实例”。比如下面这张图是我自定义的库，其中“类型”中很多函数/函数块来自我之前文章用的SCL语言编写的代码：



注意：这里的函数/函数块图标 的 右上角都有一个小的黑色三角形，与普通的函数/函数块图标是不同的，表示这是一个“类型”。顺便说一句，我之前写的SCL函数/函数块的名称都是以“FBxxx\_<功能描述>”的格式命名，其中“xxx”表示函数块的编号，比如“FB5011\_BubbleSort”。这种命名格式其实是延续经典Step7的风格，因为在经典Step7环境下，函数/函数块的编号是比较重要的，不能有重复。但是在博途环境下，即使函数/函数块的编号有重复，编译时开发环境也会自动帮我们处理（重新编号）。因此，函数/函数块的编号已经不太重要的，更关心其名称。所以，在我的用户库中我采用了新的命名规则，采用小写单词+下划线的方式命名（也被称为下划线命名法）。另外博途还支持分组，可以把功能类似的指令放到同一组，方便管理。

类型中的实例不能单独修改，需要修改时要对模板进行修改，然后发布新的版本。

## 2、模板副本 (Master Copies)

模板副本就是项目中函数/函数块/数据类型的副本（拷贝），从项目中将相应的元素拖放到【模板副本】节点就自动创建一个同名的副本，其作用相当于备份。与类型不同，模板副本可被单独修改，当其对应的函数/函数块/数据类型发生变更时它也不会自动更新。

当项目库创建完成后，可以通过更新的方式将其同步到指定的全局库中。关于项目库就先介绍到这里，下一篇文章介绍全局库。

下面是PLC基础入门的文章归档链接：

[》》PLC基础入门文章归档《《](#)

我的书《西门子S7-1200/1500 PLC SCL语言编程——从入门到精通》从硬件到软件，比较详细的介绍了SCL语言的编程，感兴趣的话可以扫描下面的二维码查看：



识别图中小  
程序码购买