

## 学习/培训文档

西门子自动化教育合作项目 (SCE) | NX MCD V12/TIA 博途 V15.0 以上版本

**数字双胞胎@教育模块 150-006** 在 CAE 系统 Mechatronics Concept Designer 中为动 态 3D 模型生成信号

siemens.com/sce



Global Industry Partner of WorldSkills International



#### 本学习/培训文档适用于以下 SCE 教育培训产品

用于培训的 SIMATIC STEP 7 软件(包括 PLCSIM Advanced)

- SIMATIC STEP 7 Professional V15.0 单独许可证 订货号: 6ES7822-1AA05-4YA5
- SIMATIC STEP 7 Professional V15.0 6 套教室许可证 订货号: 6ES7822-1BA05-4YA5
- SIMATIC STEP 7 Professional V15.0 6 套升级许可证 订货号: 6ES7822-1AA05-4YE5
- SIMATIC STEP 7 Professional V15.0 20 套学生版许可证 订货号: 6ES7822-1AC05-4YA5

TIA 博途中的软件 SIMATIC WinCC Engineering/Runtime Advanced

- SIMATIC WinCC Advanced V15.0 升级版 6 套教室许可证 6AV2102-4AA05-0AS5
- SIMATIC WinCC Advanced V15.0 20 套学生版许可证 6AV2102-0AA05-0AS7

NX V12.0 教育套装(针对中学和大学,禁止用于商业培训机构)

• 联系人: <u>academics.plm@siemens.com</u>

#### 有关 SCE 的其它信息

siemens.com/sce

#### 使用说明

通用型自动化解决方案 - 全集成自动化 (TIA) 的学习/培训文档属于"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 对其内容不提供任何担保。

本文档仅可用于 Siemens 产品/系统的首次培训。即允许全部或部分复印本文档并当面转交给培训人员/学 生们,令其在培训框架/学习范围内使用。允许在公共培训和进修场合出于培训或学习目进行转发、复制本 资料或传播其内容。

例外情况需经西门子书面许可。如有疑问,请联系 <u>scesupportfinder.i-ia@siemens.com</u>。

违者须承担赔偿损失责任。保留包含翻译在内的所有权利,尤其针对发明专利、实用新型专利以及外观设 计专利。 严禁用于工业客户培训课程。我们绝不允许该文档用于商业目的。

衷心感谢达姆施塔特应用科技大学,特别是 Heiko Webert 先生(理学硕士), Stephan Simons 教授(工程博士)以及所有参与支持编纂此份 SCE 学习/培训文档的参与人员。

## 目录

1		目标	
2		前提	条件7
3		所需	的硬件和软件
4		理论	9
	4.1	1	与外部源的通信9
	4.2	2	Mechatronics Concept Designer 中的信号属性10
5		任务	分配
6		规划	
7		结构	化的逐步式引导指南13
	7.1	1	为动态模型创建信号14
	7.2	2	在虚拟 PLC 和数字化双胞胎之间建立信号连接
	7.3	3	使用虚拟 PLC 测试数字化双胞胎
8		检查	清单 – 步骤说明
9		更多	相关信息

## 图片目录

图 1:	: 本模件中所需的软件和硬件组件概述	8
图 2:	: NX 中的应用程序 "Mechatronics Concept Designer",以及用于注释各区域的文本标记	10
图 3	: NX 菜单中的命令搜索栏,以橙色高亮显示	13
图 4:	: 在 NX 中打开一个模块	14
图 5	· 为信号适配器中的信号添加动态属性参数	15
图 6:	:参数的读/写属性	16
图 7:	: 为参数创建合适的信号	17
图 8	: 定义信号和参数之间的公式	18
图 9	: 为光传感器创建输出信号	19
图 1(	0: 光传感器系统 "csLightSensorCylinder"的信号公式	20
图 1 <sup>-</sup>	1: 创建数据类型为 "double" 的速度信号	23
图 12	<b>2</b> : 为运输表面创建参数	25
图 1:	3: 创建信号适配器 "saSortingPlant"	26
图 14	4: 准备为信号适配器创建新的符号表	27
图 1:	5: 完成为信号适配器创建新的符号表	28
图 10	6: 完成将符号分配给信号适配器	29
图 17	7: 通过 PLCSIM Advanced 选择信号分配	31
图 18	8: 启用 PLCSIM Advanced 实例中的变量进行信号分配	32
图 19	9:将 MCD 信号分配给外部信号	33
图 20	0: 通过自动分配连接所有信号	34
图 2 <sup>-</sup>	<b>1:</b> 再次断开信号分配	35
图 22	2: 确认动态模型和虚拟 PLC 之间的信号分配	35

## 表格目录

表 1: "在 CAE 系统 Mechatronics Concept Designer 中为动态 3D 模型生成信号"的检查清单......37

# 在 CAE 系统 Mechatronics Concept Designer 中为动态 3D 模型生成信号

## 1 目标

在数字化双胞胎@教育研讨会系列的模块 4 中,我们已经学会完全独立创建分拣系统的静态 3D 模型。最终我们得到了分拣系统必要独立组件的集合并在空间中插入和正确放置。在此基础上,我们将在模块 5 处理 3D 模型的动态化。通过为其分配物理属性,分拣系统的各个组件可以实现相互交互。

为了能够使数字化双胞胎与虚拟 PLC 相互配合,最后需要在 Mechatronics Concept Designer (MCD) 和 PLCSIM Advanced 之间建立连接,该连接用于通过自动化程序模拟 PLC。本模块旨在 创建信号并将其分配给两个程序。然后我们将使用数字化双胞胎@教育培训系列模块 1 中的自动 化程序来验证数字化双胞胎的功能。

## 2 前提条件

需要了解模块 5 中使用的模型的动态属性。必须学习模块 1-2 了解自动化程序的工作方式,本模块 将再次用到这些知识。

## 3 所需的硬件和软件

此模块需要以下组件:

- **1 工程师站**: 需要具备硬件和操作系统(更多信息请参阅 TIA 博途安装 DVD 以及 NX 软件包中的 自述文件)
- 2 TIA 博途中的 SIMATIC STEP 7 Professional 软件 V15.0 及以上版本
- 3 TIA 博途中的软件 SIMATIC WinCC Runtime Advanced V15.0 及以上版本
- 4 软件 SIMATIC S7-PLCSIM Advanced V2.0 及以上版本
- 5 软件 NX, 带有 Mechatronics Concept Designer 扩展模块 V12.0 及以上版本



图 1: 本模件中所需的软件和硬件组件概述

根据图1所示,工程师站是系统的唯一硬件组件。其余组件均为软件。

## 4 理论

#### 4.1 与外部源的通信

在本研讨会系列的模块 5 中,我们创建了动态属性,并使用 Mechatronics Concept Designer 中的运行时监控器对它们的功能进行了检查。我们需要在数字化双胞胎系统中建立与控制系统的连接,以通过控制器更改 MCD 中的动态属性,并将 MCD 的结果提供给控制系统。

MCD 中提供了多种与外部程序进行通信的方式(参见<u>章节 9</u>,链接 [1])。其中包括:

- 通过 MATLAB 协议与 MATLAB 进行通信
- 连接 OPC 服务器(从 MCD V12.01 起可连接 OPC UA 服务器)
- 通过 PLCSIM Advanced 建立 S7 通信或直接使用 PROFINET 协议
- 访问共享内存 (Shared Memory), 例如针对 SIMIT
- TCP/UDP 连接

与本研讨会系列的模块 1-3 一样,我们将在此模块中配置与 PLCSIM Advanced 的通信。我们将连接虚拟 PLC 与 NX/MCD。

#### 4.2 Mechatronics Concept Designer 中的信号属性

通过定义和分配信号在 NX 的扩展模块 Mechatronics Concept Designer 中实现与外部程序的通信。 Mechatronics Concept Designer 的工作界面如图2 所示。请使用屏幕右上角已讲述过的命令搜索 功能打开 NX 中的"Mechatronics Concept Designer"应用程序。



图 2: NX 中的应用程序 "Mechatronics Concept Designer",以及用于注释各区域的文本标记

该应用程序中的以下窗口用于定义信号并测试数字化双胞胎:

- 屏幕中央(参见图2,区域1)包含三维工作界面,在模拟过程中可以将其与虚拟 PLC 一起用于监控动态 3D 模型的功能。
- 在菜单栏的中间部分(参见<u>图</u>,区域2)可以对模型的模拟进行控制。我们将在<u>章节7.3</u>中 用到这些功能。

- 用于电气领域的信号属性可以在用于机械领域的动态属性旁边的菜单栏中找到(参见图 2,区域 3)。在这里可以创建信号和表格。下面对一些命令进行简要说明。
  - 您可以使用 Signal (信号)命令在模型中创建一个信号,通过运行时表达式控制对象的物理属性。运行时表达式是一个非静态值,可以在模拟的运行时期间改变。
     该表达式可以在内部与 MCD 连接,或者通过连接到来自外部源(例如 PLCSIM Advanced)的信号来确定。
  - 创建 Symbol Table (符号表), 由此定义一个用于唯一信号标识的符号列表。
     也可以从外部源(例如 STEP 7)导入符号表。
  - Signal Adapter (信号适配器)可用于连接信号和运行时表达式。每个信号适配器可以使用多个信号和运行时表达式。该命令还可用于生成信号和运行时表达式。
- MCD 的菜单栏中还列出了用于自动化领域的信号属性(参见图2,区域4)。在这里我们将使用以下属性:
  - o Signal Mapping (信号分配) 在 MCD 的信号与来自外部程序的信号之间创建
     连接。这也包括来自 PLCSIM Advanced 的信号。
- 此外,还可以通过 MCD 中屏幕左侧的资源栏(参见图 2,区域 5)调用 "Physics Navigator" (物理导航器)。那里存储了信号和连接。



## 5 任务分配

下面我们将为在模块 5 中创建的分拣系统的动态 3D 模型添加信号,并建立与虚拟 PLC 之间的连接。此外还会将数字化双胞胎@教育培训研讨会系列模块 1 中的自动化程序加载到虚拟 PLC 中,并独立对自己创建的数字化双胞胎进行验证。

我们还需要再次使用 NX 的应用程序 Mechatronics Concept Designer (MCD)。不过现在我们重点 讲述如何将动态 3D 模型连接到外部程序。

## 6 规划

要为动态 3D 模型分配信号并进行调试,至少需要 V12.0 版本的 CAD 系统 NX。另外 NX 中必须 包含附加模块 Mechatronics Concept Designer (MCD)。

可以在模块 4 和 5 中找到关于**静态**和动态 3D 模型的说明。还需要温习数字化双胞胎@教育培训 系列模块 1 – 2 中讲述的自动化程序的工作方式。如果不确定分拣系统的工作原理,请再次阅读模 块 1,章节 4.2 中的理论部分。

请再次温习**模块 1** 中关于**虚拟 PLC 和数字化双胞胎之间的交互**的内容,并准备好本系列研讨会的 模块 1 的培训内容,因为我们会用到其中的<u>章节 7.2</u> 和 <u>7.3</u>。

在命名各种信号属性时,我们使用了西门子的"标准化指南"。您可以在<u>章节9</u>中给出的链接[3] 下找到它。

## 7 结构化的逐步式引导指南

此模件提供"150-006\_DigitalTwinAtEducation\_NX\_dynModelSignals"文件夹。该文件夹包 含三个子文件夹:

- "fullDynModel"包含模块 5 中的分拣系统的所有动态 3D 模型。如果模块 5 中得到的结果不 完整,则可以将此模型用于此模块的起点。
- "fullDigTwin"包含了带有数字化双胞胎的该模块的完整解决方案。如果卡在某一步骤,它 可以起到辅助作用。
- "fullPlcBasic"提供了模块 1 中提到的带有集成 HMI 的自动化程序。我们需要用它来测试数 字化双胞胎。

如果在模块学习过程中无法在开发环境中找到某些命令或应用程序,请再次参考命令搜索。此功能 位于 NX 用户界面的右上部,如图 3 中所示。

NX		) 🗟 • 🛷	Switch W	indow 📘	Window	v <del>-</del> -		NX 12		_		×
File	e H	ome Tools	3Dconne	xion					Find a Command 🔎		$\diamond$	0
		2		S	a		?					
Net	w Open	Open a Recent Part •	Assembly Load Options	Customer Defaults	Touch Mode	Window	Help					Ŧ
1	Menu 🕶		Standard									•
ø	History				🐎 Welc	ome Page	×					

图 3: NX 菜单中的命令搜索栏,以橙色高亮显示

从找到的建议条目中挑选出合适的命令。此外,NX 还可显示该命令所在位置,以便将来也可直接 从菜单中选择该命令。

**重要:**随着 NX 的版本更新,用户界面以及各命令在菜单中的排列也在发生变化。此外,所有用户 均可创建自定义的界面。以下内容描述了 NX12.0 的默认界面,您所使用的版本可能与之存在差异。 如果因此未能在窗口中的指定位置上找到某个命令,请使用命令搜索功能。

此外还应注意,此说明仅为建议的解决方案。我们力求以一种易于理解的方式描述如何让我们创建的数字化双胞胎与模块 1 至 3 中的虚拟 CPU 进行交互。

请注意,特定的段落以节段的形式加以凸显。由于在整个描述当中常常会提示参考这些段落,所以这些标记可用于提供向导。

#### 7.1 为动态模型创建信号

在本章中我们将为分拣系统创建所有的必要信号,这些信号可通过 PLC 在外部控制。请执行如下 操作:

- → 使用操作系统复制模块 5 中使用的模型。将它们保存在文件系统中的新文件夹中。如果存在不 完整的动态模型,则也可以使用<u>章节 7</u> 中提到的"fullDynModel"项目,并创建该文件夹的 工作副本。
- → 启动 NX 并等待程序打开,屏幕上将显示起始页。单击按钮"Open"(打开)(参见图 4, 步骤 1)然后导航到您先前创建的文件夹。现在您将看到在模块 5 中使用的零件。选择模块 "assSortingPlant",其中包含分拣系统的完整动态 3D 模型(参见图 4,步骤 2)。选择 "Partially Load"选项(部分加载)(参见图 4,步骤 3),以便仅加载模块中单个零件的模型和动态属性,而不是加载其他附加图纸或坐标系。最后单击"OK"(确认)(参见图 4, 步骤 4)。

NX		🤔 🔹 🛷 🔁 s	witch Window	Window <del>▼</del> ∓	0 0	V.	
File	Hom	e Tools 3	Dconnexion		(1) $(2)$		
	1 🧖	2		🧼 📃 🍳			
New	/ Open R	e 💆 Open					×
T I	<u>l</u> enu <del>•</del>	Suchen in:	SortingPlan	t_dynModel	- = = = =		
Ō	History	-	Name		Änderungsdatum	Тур	
~	Thistory	Schnellzugriff	🧕 assSorting	Plant	12.02.2020 16:53	Siem	
		John Heizughin	Container		17.10.2019 11:59	Siem	
			Conveyorl	ong	15.10.2019 12:39	Siem	
		Desktop	2 conveyor	Short	15.10.2019 12:24	Siem	
2		-	2 cylinderH	ead	17.10.2019 15:55	Siem	
0"			CylinderLi	ner	21.10.2019 14:08	Siem	
		Bibliotheken	lightRay		21.10.2019 15:41	Siem	
			UlightSense	or	21.10.2019 15:41	Siem	
		Dieser PC	IightSense	or_mirror	20.01.2020 16:07	Siem	
			2 limitSwite	hSensor	17.01.2020 13:30	Siemi IV Preview	
Ð			workpiece	Cube	14.10.2019 12:46	Siem	
-		Netzwerk	Workpiece	Cylinder	14.10.2019 16:25	Siemi	
33							
<b>E</b>			<			· A	
	(2)		Dateiname:	assSortingPlant	•	ок	2
	(3)		Dateityp:	Part Files (*.prt)	- I	Abbrechen	
	~						
		E L Questo	0-1				
		Ontion	Only				
		Parti	ally Load	<b>_</b>			
		Options					1
							11

图 4: 在 NX 中打开一个模块

#### 节段: 创建信号并将其与信号适配器相匹配

→ NX 应用程序 "Mechatronics Concept Designer"中打开该模块之后,就可以创建第一个示例了。现在我们将创建并链接一个用于激活对象源的信号,这是生成长方体工件的必要步骤。要添加信号并将其与动态属性链接,请首先从"Electrical"(电气)区域打开"Signal Adapter"(信号适配器)命令,如图 5,步骤 1 所示。随后显示"Signal Adapter"(信号适配器)命令窗口。首先选择要与信号连接的动态属性参数。为此,请单击"Parameters"(参数)命令选项卡下的"Select Physical Object"(选择物理学对象)按钮(参见图 5,步骤 2)。导航到资源栏中的"Physics Navigator"(物理导航器)(参见图 5,步骤 3),然后选择对象源"osWorkpieceCube"作为第一个参数(参见图 5,步骤 4)。选择此项后,现在可以在命令窗口中的"Parameter Name"(参数名称)下选择要分配给信号的相应参数。此时选择所选对象源的参数名称"active"(激活)(参见图 5,步骤 5)。单击"Add Parameter"(添加参数)按钮(参见图 5,步骤 6),它用于把参数加入该信号适配器。



图 5: 为信号适配器中的信号添加动态属性参数

→ 在命令组"Parameters"(参数)下的表中可以找到刚刚选择的参数。将其"Alias"(别名) 设置为"paOsWorkpieceCube\_SetActive"(参见图7,步骤1)。前缀"pa"代表 "Parameter"参数,用于与信号名称明确区别。也可以单击表格开头的复选框,以便之后为参 数分配信号。选中后复选标记 会进行显示。如果在"Parameters"(参数)命令选项卡中向 右滚动,则可以添加参数的其他属性,例如属性"Read/Write"(读/写),它指示参数是否可 读(写为"R")还是可写(写为"W")。当前参数"paOsWorkpieceCube\_SetActive"只 能被写入(参见图6,步骤1)。

🗘 s	ignal Adapter						ა x
Para	ameters						^ ^
*	Select Physics Object (0)				$\bigcirc$		<b>+</b>
Para	meter Name						-
Add	Parameter					•	*
Α	Alias	Object	P	<b>V</b> .	Data Type	Read/Write	×
	paOsWorkpieceCube_SetActive	osWorkpieceCube	а.	t	bool	W	

图 6: 参数的读/写属性

→ 现在我们需要一个与该参数连接的所属信号。为此,请单击"Signals"(信号)命令选项卡下的"Add"(添加)按钮(参见图7,步骤2)。屏幕中出现一个新信号。根据参数调整信号的属性。为此双击信号"Signal\_0"的标准名称,然后将信号重命名为"osWorkpieceCube\_SetActive"。必须选择与所属参数相同的类型作为数据类型。当下是一个"bool"型。必须根据参数的"Read/Write"(读/写)属性选择"Input/Output"(输入/输出)属性的值。对于必须写入的参数,从 MCD 的角度来看,信号必须来自外部源的输入。对于需要读取的参数,信号必须是来自外部程序的输出。由于已写入当前参数"paOsWorkpieceCube\_SetActive",因此必须为信号"osWorkpieceCube\_SetActive"选择值"Input"(输入)。将表中为信号选择的输出值设置为信号的初始值。在这种情况下,它应该为"false"(参见图7,步骤3)。

#### 学习/培训文档 | 数字双胞胎@教育模块 150-006,版本 2020/08 | 数字化工厂,FA

NX	🖬 🤊 • 🕫 😽 🖻 🕀	2 • 4		_ 🗆 ×
File	Home Modeling Asser	mblies C	Curve Analysis View Render Tools Application 3Dconnexion Find a Command 🔎 🔳	) \land 🕜
Requ	irement Sketch	<ul> <li>Play</li> <li>Stop</li> <li>Simulat</li> </ul>	Image: Angular Spring Joint       Image: Angular Spring Joint         Image: Angular Spri	
<u>∃</u> ≩ №	<u>M</u> enu ▼ No Selection Filter ▼ E	intire Assem	<u>uby</u> 🖞 🐈 🐂	🔊 • 👇
¢	Physics Navigator		SortingPlant.prt ×	
	Name 🔺		Signal Adapter	ଏ X
<u></u>	E 🔁 Basic Physics	^	Parameters	<u>^</u>
	osWorkpieceCube		Select Physics Object (0)	<u></u>
-0-	osWorkpieceCylinder		* Select Physics Object (0)	Ψ
	+ M 😚 rbContainer		Parameter Name	Ŧ
0	+ V G rbConveyorLong		Add Parameter	*
	The second secon		A Alias Object Object Type Parameter Value	X
P1= P2=	₩ ₩ IbCylinderLiner		☑ paOsWorkpieceCube_SetActive osWorkpieceCube Object Source active true	
a				
Fø	🗉 🗹 🍯 rbWorkpieceCylinder			•
	😑 🚘 Joints and Constraints		Signals	^
	- <b>⊠</b> R <sup>⊉</sup> fjContainer			
<b>a</b>			A Name Data Type Input/Out Initial Value Measure Unit	14
Fo		~	osworkpiececube_selActive_bool input laise	X
÷.	<	>		*
- - 	Dependencies	× ×	З ОК Арріу С	Cancel
Select	object to get its parameter		Enter a new value	

图 7: 为参数创建合适的信号

→ 现在必须通过逻辑方式链接参数和相关信号。为此请在命令窗口中向下滚动到"Formula" (公式)命令选项卡。此处可以将公式分配给"paOsWorkpieceCube\_SetActive"参数。单 击表中的相应行(参见图 8,步骤 1)。现在可以在"Formula"公式输入字段中进行适当的 分配。在这种情况下,只需将"osWorkpieceCube\_SetActive"信号分配给该参数,如图 8, 步骤 2 所示。单击键盘上的 Enter 键后,可以在"Formula"(公式)列下的表格中看到刚刚 进行的分配。

NX 🖬 🤊 • 🤊 🖗 📾 📾 :	55	nter Switch Window	w 🔝 Window 🕶 🖛	NX 12 - M	echatronics Conce	ept Designer	_ 🗆 ×
File Home Modeling Assemb	lies C	Curve Analysis Vi	ew Render To	ols Application	3Dconnexion	Find a Comman	P 🗆 🛆 🕜
Requirement states and the states of the sta	Play Stop Simulat		Angular Spring Join Linear Spring Joint Angular Limit Joint Mechanical	nt → → → → → → → Elect	Operation	Add Design C	國, • 2. • 回, • olla •
Menu ▼ No Selection Filter ▼ Entited	re Assem	nbly 👻 🕄 👻 🕈	• • • •		🗏 O 🍠 🛛	3 🖬 • 🚭 •	🇊 • 🕪 •
Physics Navigator		assSortingPlant.p	rt ×				
Name 🔺		Signal Adapter					ა x
Basic Physics	^	Signals					^ ^
> osWorkpieceCylinder		A Name	Data	Type Input/Out.	Initial Value	Measure	Unit 🍫
- 🖉 🍯 rbContainer		osWorkpieceC	ube_SetActive bool	Input	false		×
→ + Ø							
+ 🗹 🍯 rbConveyorShort							
P1= 🐨 🗹 🎯 rbCylinderHead		<					> +
P2= 🔂 🎯 rbCylinderLiner		Formulas		(1)			^
🥵 🕂 🗹 🎯 rbWorkpieceCube		- official as					
FØ		Assign to	Eerm	ula			*
Joints and Constraints		<pre>paOsWorkpieceCu </pre>	be_SetActive osWo	kpieceCube_SetActiv	e		, ×
		Formula		(2)			
Fo fjConveyorShort	, <b>~</b>	osWorkpieceCube	e_SetActive				f(x) <u>%</u>
Details	~	Name					^ v
÷ Dependencies	v					ОК Арр	ly Cancel
elect object to get its parameter			Enter a new value				[m]

图 8: 定义信号和参数之间的公式



现在我们已经独立创建了数字化双胞胎的第一个信号。现在根据<u>章节 7.1</u>, "**节段: 创建信号并将 其与信号适配器相匹配**"中的内容创建其他信号。请使用以下关键数据:

- → 应从对象源 "osWorkpieceCylinder"中以别名 "paOsWorkpieceCylinder\_SetActive"在 信号适配器中创建参数 "active"(激活)。再次选中参数前面的复选标记 ☑ 以分配信号。 关联的信号应具有名称 "osWorkpieceCylinder\_SetActive",并且数据类型应为 "bool"。 该信号定义为 "Input" 输入,输出值为 "false"。直接分配 "osWorkpieceCylinder\_SetActive" 信号作为参数 "paOsWorkpiece Cylinder\_SetActive"的公式。
- → 为信号适配器中的碰撞传感器"csLightSensorCube"的参数"triggered"(触发)"创建 一个别名为"paCsLightSensorCube\_Detected"的新参数。为所属信号输入名称为 "csLightSensorCube\_Detected"的布尔信号。但是,此信号必须定义为"Output"(输 出),因为参数"paCsLightSensorCube\_Detected"在"Read/Write"(读/写)列中显示 值为"R",这意味着必须对其进行读取。将输出值"false"分配给信号。如图9,步骤1所 示,选中信号前面的复选标记 I 以便为输出信号分配公式。直接分配参数"paCsLight SensorCube\_Detected"作为"csLightSensorCube\_Detected"信号的公式。



图 9: 为光传感器创建输出信号

→ 输送过程中的下一个光传感器系统由两个碰撞传感器组成。因此必须创建两个参数。首先在信号适配器中为传感器"csLightSensorCylinder"的"triggered"(触发)参数定义一个别名为"paCsLightSensorCylinder\_Detected"的参数。在信号适配器中为第二个碰撞传感器"csLightSensorCylinderTop"的"triggered"(触发)参数创建一个别名为"paCsLightSensor CylinderTop\_Detected"的参数。现在我们就创建了一个组合信号,它对两个参数都做出反应。应将其命名为"csLightSensor Cylinder\_Detected",并且其数据类型为"bool"。由于在这种情况下也必须读取上述两个参数,因此组合信号应配置为"Output"(输出),并且输出值为"false"。如图 10,步骤 1 所示,为"csLightSensorCylinder\_Detected"输入以下公式:

" ((paCsLightSensorCylinderDetected) & (!paCsLightSensorCylinderTop\_Detected)) " 。

此公式表示两个参数的与运算,第二个参数取反,即当"paCsLightSensorCylinderDetected" 采用值"true"且同时"paCsLightSensorCylinderTop\_Detected"为"false"时,输出信 号"csLightSensorCylinder\_Detected"为"true"。根据本系列研讨会模块 1 中有关分拣 系统功能的说明,由于工件自身的高度,只有在光传感器系统的下部光传感器被触发,而同时 上部光传感器未检测到碰撞时方可识别出圆柱体工件。此逻辑通过公式表示。



图 10: 光传感器系统 "csLightSensorCylinder"的信号公式



将输入信号分配给可写参数时,必须选中参数前面的复选标记 ☑。同样的,将可读参数分配给输出信号时,必须选中信号前面的复选标记 ☑。

仅当复选标记 🗹 可见时才能定义相应的公式。

- → 针对碰撞传感器 "csLightSensorWorkpiece"的 "triggered"(触发)参数,在信号适配器中需要一个参数,应为其赋予别名 "paCsLight SensorWorkpiece\_Detected"。为此,请创建一个名称为 "csLightSensorWorkpiece\_Detected"的布尔信号。同样将其定义为初始值为 "false"的 "Output"(输出)信号。信号 "csLightSensor Workpiece\_Detected"的公式为 "paCsLightSensorWorkpiece\_Detected"。
- → 为信号适配器中的碰撞传感器 "csLimitSwitchCylinder NotExtended"的参数 "triggered" (触发)创建一个别名为 "paCsLimitSwitch CylinderNotExtended\_Activated"的参数。 添加一个称为 "csLimit SwitchCylinderNotExtended\_Activated"的布尔信号。将此 信号定义为 "Output" (输出)类型,并且输出值为 "false"。将 "paCsLimit SwitchCylinderNotExtended\_Activated" 作为信号 "csLimitSwitchCylinderNotExtended\_ Activated"的公式。
- → 在信号适配器中同样使用别名 "paCsLimitSwitchCylinderRetracted\_Activated"为最后剩余的碰撞传感器 "csLimitSwitch CylinderRetracted"创建参数 "triggered"(触发)。所属信号 "csLimitSwitchCylinderRetracted\_Activated"为数据类型 "bool"。此外,应将其定义为输出值为 "false"的 "Output"(输出)信号。直接指定 "csLimitSwitchCylinderRetracted\_Activated"的公式为参数 "paCsLimitSwitchCylinderRetracted\_Activated"。
- → 针对位置控制器 "pcCylinderHeadExtend"的参数 "active" (激活),应在信号适配器中 创建一个别名为 "paPcCylinderHeadExtend\_SetActive"的参数。继续生成数据类型为 "bool"的名为 "pcCylinderHeadExtend\_ SetActive"的新信号。将信号定义为 "Input" (输入),输出值为 "false"。将参数 "paPcCylinderHeadExtend\_SetActive" 指定为信 号 "pcCylinderHeadExtend\_SetActive"的公式。

- → 下面我们为位置控制器 "pcCylinderHeadRetract"的参数 "active" (激活)在信号适配器 中生成一个别名为 "paPcCylinderHeadRetract\_SetActive"的参数。相应信号的名称为 "pcCylinderHeadRetract\_SetActive"。将此信号定义为布尔,并将其定义为 "Input" (输入)信号,值为 "false"。 "paPcCylinderHeadRetract\_SetActive"的公式为 "pcCylinderHeadRetract\_SetActive"。
- → 速度控制器 "scConveyorLongConstSpeed"的参数 "activ" (激活)在信号适配器中需要
   一个别名为 "paScConveyorLongConst Speed\_SetActive"的参数。其关联的信号应命名
   为 "scConveyorLongConst Speed\_SetActive"。此信号应为 "bool"数据类型,定义为
   "Input" (输入)信号,输出值为 "false"。将 "scConveyorLongConstSpeed\_
   SetActive"分配为 "paScConveyorLongConstSpeed\_SetActive"的公式。
- → 必须为速度控制器 "scConveyorLongVarSpeed"在信号适配器中定义两个参数和两个信号。 信号适配器中的第一个参数和所关联的信号用于激活速度控制器。为此在信号适配器中为 速度控制器 "scConveyorLongVarSpeed"的 "active"(激活)参数创建一个别名为 "paScConveyorLongVarSpeed\_SetActive"的新参数。另外还要创建"scConveyorLong VarSpeed\_SetActive"信号,并将数据类型设为"bool"。信号应用于"Input"(输入)。 初始值为 "false"。最后输入信号 "scConveyorLongVarSpeed\_Set Active"作为 "paScConveyorLongVarSpeed\_SetActive"的公式。

第二个信号能够为速度控制器设定可变速度。为此请在信号适配器中创建一个新参数,该参数 应与速度控制器 "scConveyorLongVarSpeed"的 "speed"(速度)参数连接,并使用别 名 "paScConveyorLongVarSpeed\_SetSpeed"。将关联的信号命名为 "scConveyorLong VarSpeed\_SetSpeed"。由于使用该信号指定速度,因此必须将数据类型声明为 "double"。 如果信号的数据类型不是 "bool",则必须在 "Measure"(测量)栏中输入物理类型,并在 "Unit"(单位)栏中输入值的关联物理单位。同时请为当前信号在 "Measure"(测量)列 中输入值 "speed"(速度)以及在 "Unit"(单位)列中输入 "mm/s"(参见图 11,步骤 1)。这是一个 "Input"(输入)信号。另外请将初始值设为 "0.0"。将信号 "scConveyor LongVarSpeed\_SetSpeed"分配给参数 "paScConveyorLongVarSpeed\_SetSpeed"作 为公式。



图 11: 创建数据类型为 "double" 的速度信号

- → 针对速度控制器 "scConveyorShortConstSpeed"将参数 "active" (激活)作为新的参数 加入信号适配器中。为此参数指定别名 "paScConveyorShortConstSpeed\_SetActive"。
   创建数据类型为 "bool"的所属信号 "scConveyorShortConstSpeed\_SetActive"。请将 其标识为 "Input" (输入)信号,并将初始值设为 "false"。最后,将信号 "scConveyor ShortConstSpeed\_SetActive" 直接分配给 "paScConveyorShortConstSpeed\_SetActive" 作为其公式。
- → 速度控制器 "scConveyorShortVarSpeed" 在信号适配器中同样也需要两个信号。

信号适配器中的第一个参数应为速度控制器"scConveyorShortVarSpeed"的"active" (激活)参数,并具有别名"paScConveyorShortVarSpeed\_SetActive"。并生成一个称 为"scConveyorShortVarSpeed\_SetActive"的布尔信号作为其关联信号。这应该是一个 "Input"(输入)信号,初始值为"false"。将信号"scConveyorShort VarSpeed\_ SetActive"分配给参数"paScConveyorShortVarSpeed\_SetActive"。

第二个信号用于指定速度控制器的目标速度。为此,请在信号适配器中创建一个新参数,该参数基于速度控制器的"scConveyorShortVarSpeed"参数"speed"(速度)。指定该参数的别名为"paScConveyorShortVarSpeed\_SetSpeed"。在定义新信号"scConveyorShortVarSpeed\_SetSpeed"时,必须再次确保将数据类型指定为"double",在"Measure"(测量)栏中指定"speed"(速度)和单位"mm/s"。它是"Input"(输入)信号,输出值为"0.0"。参数"paSc ConveyorShortVarSpeed\_SetSpeed"的公式为"scConveyor ShortVarSpeed\_Set Speed"。

 → 为确保仅在两个关联的速度控制器中的一个处于激活状态时才移动运输表面 "tsConveyorLong",必须在信号适配器中添加一个附加参数。为此运输表面选择名称为 "active"(激活)的参数,并将新参数的别名设置为 "paTsConveyorLong\_SetActive"。 最后如图 12,步骤 1 中所述,为参数分配以下公式:

"((scConveyorLongConstSpeed\_SetActive) | (scConveyorLongVarSpeed\_SetActive))"

字符"|"代表 OR (或)运算。这样可以确保至少一个速度控制器针对该传送带激活后,传送 带方才移动。由于正在开发的自动化程序中的逻辑设定,在正常运行顺序中,运输表面只能通 过一个速度控制器同时运行。

NX	🔚 🤊 • (* 🖻 🗟 🛠 • .	nter Switch Window 📘 Window 🕶 🖛	NX 12 - Mechatronics Concept Designer	_ 🗆 X
File	Home Modeling Assemblies	Curve Analysis View Render Too	ols Application 3Dconnexion Find a Command 🔎	■   3
Requ	ins Engineeri • Mechanical C • Play	Image: Constraint of the system     Image: Constraint of the system       Image: Constraint of the system     Image: Constraint of the system       Image: Constraint of the system     Image: Constraint of the system       Image: Constraint of the system     Image: Constraint of the system       Image: Constraint of the system     Image: Constraint of the system       Image: Constraint of the system     Image: Constraint of the system       Image: Constraint of the system     Image: Constraint of the system       Image: Constraint of the system     Image: Constraint of the system       Image: Constraint of the system     Image: Constraint of the system       Image: Constraint of the system     Image: Constraint of the system       Image: Constraint of the system     Image: Constraint of the system       Image: Constraint of the system     Image: Constraint of the system       Image: Constraint of the system     Image: Constraint of the system       Image: Constraint of the system     Image: Constraint of the system       Image: Constraint of the system     Image: Constraint of the system       Image: Constraint of the system     Image: Constraint of the system       Image: Constraint of the system     Image: Constraint of the system       Image: Constraint of the system     Image: Constraint of the system       Image: Constraint of the system     Image: Constraint of the system       Image: Constraint of the system <td>tt</td> <td>· • •</td>	tt	· • •
1	Menu   No Selection Filter   Entire Assen	nbly 🔻 🖽 🐄 🍢 🕈 🛉 🦷 🦛 L		* 🕪 * 🛛 •
Q	Physics Navigator	Signal Adaptor		10 ×
	Name	Gronghai Adapter		0 ^
F	CSLightSensorCube	Parameter Name		· · ·
	stightSensorCylinderTon	Add Parameter		*
-9-	stightSensorWorkniece	A Alias	Object Object Type Parameter Value	Je 🗙
an	sclimitSwitchCylinderNotEx	paTsConveyorLong_SetActive	tsConveyorLong Transport Surface active true	^
CC .	🗹 🏠 csLimitSwitchCylinderRetrac	4		. * 🐷
P1=	- 🗹 📌 pcCylinderHeadExtend			
P2=	- 🗹 🥕 pcCylinderHeadRetract	Signals		v
Re	- 🗹 🥓 scConveyorLongConstSpeed	Formulas	(1)	^
Fø				
M	- 🗹 🧨 scConveyorShortConstSpee	Assign to		•
		<pre>palsConveyorLong_SetActive ((scCo </pre>	nveyorLongConstSpeed_SetActive)   (scConveyorLongVarSpi	×
0_	sconveyorLong	Formula		_
Fo	tsConveyorShort v			2 0
÷		((scConveyorLongConstSpeed_SetActr	ve) [ (scConveyorLongvarSpeed_SetActive)) /(x)	
÷	Details V			
-	Dependencies V		OK Apply	Cancel
Select	object to get its parameter			

图 12: 为运输表面创建参数

→ 按照针对 "tsConveyorLong"相同的步骤处理运输表面 "tsConveyorShort"。为名为
 "paTsConveyorShort SetActive"的新参数选择 "active" (激活)状态。

使用以下公式:

" ((scConveyorShortConstSpeed\_SetActive)|(scConveyorShortVarSpeed\_SetActive))" 以确保运输表面正常运行。 现在我们定义了信号适配器中所有必要的参数和信号。最后,输入"saSortingPlant"作为信号适 配器的名称(参见图 13,步骤 1)。前缀"sa"代表英文名称"signal adapter"(信号适配器)。单击"OK"(确认)按钮确认新信号适配器的配置(参见图 13,步骤 2)。

NX 🖬 🤊 · 🤊 🕆 🖻 🗟 👯 ·	🛷 📅 Switch Window 🔲 Window 👻 🔹 NX 12 - Mechatronics Concept Designer 💦 _ 🖂 🗙
File Home Modeling Assemblies	Curve Analysis View Render Tools Application 3Dconnexion Find a Command 🔊 🗐 🐟 🤪
Requirement Systems Engineeri * Mechanical C*	Image: Product of the second state
Menu      No Selection Filter      Entire Asser	nbly - 🖏 🐂 🐂 - 14 🦷 64 🗔 - 🚳 📦   🔯 💷 O 🍠 🦉 🖽 - & - 📦 - 160 - 1
Physics Navigator	Sissional Advanta
	🕹 Sighal Adapter
He csLightSensorCube	Parameters V
QL Image: State of the state of t	Signals V
	Formulas
C tsLimitSwitchCylinderNotEx	Name 1
SchultzwitchCylinderKetrac	saSortingPlant
P2=	
🚱 🖉 🧨 scConveyorLongConstSpeed	OK Apply Cancel
For scConveyorLongVarSpeed	
scConveyorShortConstSpeed	
SeconveyorLong	
Ho IsConveyorShort ✓	Lxc
× >	
Details	
Dependencies V	
Select object to get its parameter	

图 13: 创建信号适配器 "saSortingPlant"

随后系统将显示新窗口"Add Symbols to Symbol Table"(将符号添加到符号表),要求我们 指定将信号适配器的信号作为符号添加至其中的符号表。此时可以扩展现有的符号表或创建新的符 号表。由于我们尚未在当前项目中创建符号表,因此单击"Create new symbol table"(创建新 的符号表)按钮(参见图 14,步骤 1)。



图 14: 准备为信号适配器创建新的符号表

现在将出现命令窗口"Symbol Table"(符号表)。在该窗口中可以定义新的符号,也可以指定 符号表的名称。由于可以完全应用信号适配器的信号,因此无需在此处定义新信号。将符号表命名 为"stSortingPlant"(参见图 15,步骤 1),然后单击"OK"(确认)按钮(参见图 15,步骤 2)。前缀"st"代表英文名称"signal table"(信号表)。

NX		🤣 🗄 Switch Window 📃 Window 🕶 🗟	NX 12 - Mechatronics Concept Designer	_ 🗆 ×
File	Home Modeling Assemblies	Curve Analysis View Render Tools	Application 3Dconnexion Find a Command	₽ 🖻 🗞 🔞
Requ	irement 🐮 Ketch 🏐 Sketch 🗊 Stop	Image: Spring Joint       Image: Sprin		a▼
± ∎	enu 👻 🦳 👻 Within Wor	k Part O 🔻 🕄 🍖 🖓 🌪 🕈 🏠 🎧 🛄	• 🖉 📦   👧 🖳 🗘 🎻 🦫 🖬 • 💩 • 1	🗊 T 🚧 T 🔤 🗸
ø	Physics Navigator	assSortingPlant.prt ×		
	Name 🔺	Symbol Table	<u> </u>	
<b>9</b> _	CsLightSensorCube	Symbols	^	
	csLightSensorCylinder	Symbol Name IO Type Data Typ	e Comr 🐪	
-9-	csLightSensorCylinder lop		×	-
a»	CSLightSethortPrece			
Res (				
[P1=]	- 🗹 🕕 pcCylinderHeadExtend			
P2=	- CylinderHeadRetract			
<b>%_</b>	- ScConveyorLongConstSpeed			
F	scConveyorLongVarSpeed			
	scconveyorShortVarSpeed	<	>	
-	- SconveyorLong	Name	^	
F⊚		(1)		
÷	< >	stSortingPlant	(2)	
*	Details V	0	Cancel	
-	Dependencies V		Cancel	

图 15: 完成为信号适配器创建新的符号表

现在,您将返回到"Add Symbols to Symbol Table"(将符号添加到符号表)窗口。如未返回,则应该选择刚刚创建的符号表"stSortingPlant",如图 16,步骤 1 所示。单击"OK"(确认)按钮完成创建过程(参见图 16,步骤 2)。



图 16: 完成将符号分配给信号适配器

现在我们已将所有必需的信号加入动态 3D 模型中,下面我们通过它们与虚拟 PLC 建立信号连接。

首先通过单击"Save"(保存) 🔤 按钮将更改保存到模型。

#### 7.2 在虚拟 PLC 和数字化双胞胎之间建立信号连接

要想创建信号连接,虚拟 PLC 必须已经运行。因此本节段将再次使用 TIA 博途和 PLCSIM Advanced。请按照下列步骤建立连接:

- → 在操作系统中解压缩此模块随附的压缩文件(参见<u>章节 7</u>),并将"fullPlcBasic"文件夹的 内容保存在我们选择的文件夹中。该文件夹包含了模块 1 中使用过并在模块 2 中进行了描述的 自动化程序。
- → 现在打开 TIA **博途,**并从刚刚创建的文件夹中解压缩项目 "150-006\_DigitalTwinAtEducation\_ TIAP\_Basic.zap15"。按照**数字化双胞胎@教育研讨会系列模块 1 第 7.1 章**的说明进行操作。
- → 编译硬件配置和自动化程序软件。请遵循本研讨会系列模块1第7.2章中的说明。
- → 打开 "S7-PLCSIM Advanced"程序,并启动虚拟 PLC 的新实例。将此实例命名为 "DigTwinAtEdu\_PLCSIM"。然后将我们的自动化程序加载到虚拟 PLC 中,并等待直到 CPU 状态更改为 "Start"(启动),此时实例名称前面会出现一个绿色框。请遵循本研讨会 系列模块 1 第 7.3 章中的说明。

现在虚拟 PLC 可以运行了,随后可以配置到动态 3D 模型的信号连接。在 Mechatronics Concept Designer 中切换回带有信号的动态 3D 模型,然后执行以下操作:

→ 在 "Automation"(自动化)菜单栏中执行 "Signal Mapping"(信号分配)命令(参见图 <u>17</u>,步骤 1)。系统将打开 "Signal Mapping"(信号分配)命令窗口。首先必须在此选择 外部信号源。我们尝试连接到 PLCSIM Advanced,为此请转到 "External Signal Type" (外部信号类型)选项卡,然后选择 I 类型 "PLCSIM Adv"(PLCSIM 助手)(参见图 <u>17</u>, 步骤 2)。目前我们的动态模型尚未确定要与 PLCSIM Advanced 中的哪个实例建立连接。因 此,单击 "PLCSIM Adv Instances"(PLCSIM Adv 实例)下的 "Setting"(设置)按钮 (参见图 <u>17</u>,步骤 <u>3</u>)。

NX	📓 🤊 • 🍽 🖗 🛣 • 🔺	Switch Window Window - NX 12 - Mechatronics Concept Designer _ X
File	Home Modeling Assemblies C	arve Analysis View Render Tools Application 3Dconnexion Find a Command 🔎 🖲 🐟 😮
Requ	irement Karlow Mechanical C Y Simulat	Image: Spring Joint Spring
₫	Menu → Within Work	Part O 👻 🖞 🐂 🐂 🐐 🐐 🕼 🛄 🕶 🚳 🔯 🖉 🐼 🍼 🕼 🖉 🗸 🖉 🖌
ø	Physics Navigator	🧕 assSortingPlant.prt 🖸 🗙
	Name 🔺	Signal Mapping $\mathfrak{O}$ X
● 	CLightSensorCube     SclightSensorCylinder     ClightSensorCylinderTop     ClightSensorCylinderTop     ClightSensorWorkpiece     SclightSensorWorkpiece     SclightSwitchCylinderNotEx	External Signal Type Type PLCSIMAdv Instances Signals
Diel	- PcCylinderHeadExtend	MCD Signals (15) A External Signals (0)
P2=		Find     Match Case     Match Whole Word     Find     Match C
Fa	scConveyorLongVarSpeed	Name Adapter Name IO Type Dat Name IO Type
		osWorkpieceCu saSortingPlant Input t 🗘
a_	🗹 💋 tsConveyorLong	Do Auto Mapping
) Fo	< tsConveyorShort <	Mapped Signals
* • •	Details V Dependencies V	OK Cancel
		[m]

图 17: 通过 PLCSIM Advanced 选择信号分配



→ 现在系统将打开"External Signal Configuration"(外部信号配置)窗口。这里可以选择所需的实例并释放用于信号分配的关联变量。首先单击"Refresh Registered Instances"(更新注册的实例)按钮(参见图 18,步骤 1)。系统将显示先前启动并加载的虚拟 PLC 实例。状态"Run"(运行)表示可以访问该虚拟 PLC。选择该实例后,如图 18,步骤 2 所示,系统将显示自动化程序的 I/O 信号。通过单击框上的"Select All"(全选)以勾选所有可用变量(参见图 18,步骤 3)。单击"OK"(确认)选择(参见图 18,步骤 4)。

CSI	M Adv	(1)			
nstar	nces	<u> </u>			^
		Refresh Re	egistered Instance	5	
Nam	ne ID	CPU	Status	Owner Par	t Mes
DigT	winAtEdu_PLC 0	1516Fv2	Run	assSorting	Plant
	(2)				
	$\smile$				
<					>
Tags	; (15)	$\bigcirc$			^
Sho	w	3		ю	-
Find	4		Match	Case 🗌 Match Wh	ole Word 🔺
	Select All				
<b>S</b> .	Name	Ю Туре	Data Type	Area Type	Message
$\checkmark$	csLightSensorCube	Input	bool	Input	^
$\checkmark$	csLightSensorCylin	Input	bool	Input	
V	csLightSensorWork	Input	bool	Input	
V	csLimitSwitchCylin	Input	bool	Input	
	csl.imitSwitchCylin	Input	bool	Input	

图 18: 启用 PLCSIM Advanced 实例中的变量进行信号分配



→ 返回"Signal Mapping"(信号分配)命令窗口。窗口的右侧显示了刚刚选择的虚拟 PLC 及相应可用的外部信号。现在可以开始分配信号了。首先在窗口左侧的"MCD Signals" (MCD 信号)表中选择"osWorkpieceCube\_SetActive"信号(参见图 19,步骤 1)。然后在"External Signals"(外部信号)表中查找来自自动化程序的相应信号。从这两个程序中选择相同的信号,如图 19,步骤 2 所示。单击"Map Signal"(分配信号)按钮,在两个信号之间建立连接(参见图 19,步骤 3)。注意 MCD 中的输入信号只能与 PLC 的输出信号相连接,反之亦然。

Signal Mapping											υ
External Signal Type											
Туре									PLCSIM Ad	v	
PLCSIMAdv Instances									DigTwinAtEdu_P	LCSIN 🔻	4
Signals											
MCD Signals (15)	(1)		۸		External S	ignals (15)		2			~
Find	Match Case	Match Whole W	ord 🔶		Find			Match C	ase 🗌 Match Whol	e Word	•
Name	Adapter Name	IO Type	Da		Name			IO Type	Data Type	Ma	ppir
osWorkpieceCube_SetActive	saSortingPlant	Input	^	0-0	csLimitS	witchCylinderRetracted	_Activated	Input	bool	0	^
osWorkpieceCylinder_SetActive	saSortingPlant	Input			osWorkpieceCylinder_SetActive osWorkpieceCube_SetActive			Output	bool	0	
csLightSensorCube_Detected	saSortingPlant	Output						Output	bool	0	
csLightSensorCylinder_Detected	saSortingPlant	Output			pcCylinderHeadRetract_SetActive			Output	bool	0	•
< NULL DIVISION	C 11 DI 1	<u></u>	>		10 m			<u></u>		^>	Ť
			Do A	uto I	Mapping						
Mapped Signals						3)					
Connection Name	MCD Signal Name	Direction	Exter	nal S	ignal Name	Owner Component	Message				5
<										>	
			Check fo	r N-	>1 Mapping						
									ОК	Can	icel

图 19: 将 MCD 信号分配给外部信号

→ 可以在命令组 "Mapped Signals" (已分配信号)中看到已经分配的信号。现在加入更多分配。由于此模型中 MCD 信号的名称与自动化程序的变量名称匹配,因此可以选择 "Do Auto Mapping" (执行自动分配)按钮以使程序自动执行此过程(参见图 20,步骤 1)。

Signal Mapping											0
External Signal Type											
Туре									PLCSIM Adv		•
PLCSIMAdv Instances								[	DigTwinAtEdu_PLC	SIN 🔻	1
Signals											
MCD Signals (15)			^		External Signa	ls (15)					^
Find	Match Case	Match Whole Wo	rd 🔶		Find			Match Case	e 🗌 Match Whole	Word	•
Name	Adapter Name	IO Type	Da		Name			IO Type	Data Type	Ma	ppir
osWorkpieceCube_SetActive	saSortingPlant	Input	^	19-49 10	csLimitSwitc	sLimitSwitchCylinderRetracted_Activated		Input	bool	0	^
osWorkpieceCylinder_SetActive	saSortingPlant	Input			osWorkpieceCylinder_SetActive		Output	bool	0		
csLightSensorCube_Detected	saSortingPlant	Output			osWorkpieceCube_SetActive		Output	bool	1		
csLightSensorCylinder_Detected	saSortingPlant	Output			pcCylinderHeadRetract_SetActive		Output	bool	0	· .	
<ul> <li>C</li> </ul>	C 12 DL 1	<u></u>	>				11 A.	<u>.</u>		^>	~
			🗩 Do A	uto N	1apping						
Mapped Signals	(1	$\overline{)}$									
Connection Name			MCD Si	gnal N	lame	Direction	External Si	gnal Name	Owner Compone	ent 🕴	\$
🖃 🖌 PLCSIM Adv.DigTwinAtEdu_PLCSIM											-
saSortingPlant_osWorkpieceCub	pe_SetActive_osWorkpiece	Cube_SetActive	osWork	oiece(	Cube_SetActive	←	osWorkpie	ceCube_SetActive			
<										>	
			Check fo	r N->	1 Mapping						1
									ОК	Can	cel

#### 图 20: 通过自动分配连接所有信号

	如果分配了错误的信号,则可以在" <b>Mapped Signals"</b> (已分配信 号)命令项下的表中选择错误的分配,然后单击" <b>Break"</b> (断开)
提示	▶ 按钮以断开连接(参见 <u>图 21</u> ,步骤 1)。然后必须再次进行正确的分配。

Mapped Signals		(1)			/			
Connection Name	MCD Signal Name	Direction	External Signal Name	-0+	S			
🖃 🗹 PLCSIM Adv.DigTwinAtEdu_PLCSIM					-			
saSortingPlant_osWorkpieceCube	osWorkpieceCube	<del>~</del>	osWorkpieceCube_Se					
٢				>				
Check for N->1 Mapping								

- 图 21: 再次断开信号分配
- → 整个过程已在动态 3D 模型和虚拟 PLC 之间连接了所有 15 个信号。检查分配的正确性,然后 单击"OK"(确认)按钮完成信号分配(参见图 22,步骤 1)。

Signal Mapping											ა
External Signal Type											
Туре									PLCSIM Adv		
PLCSIMAdv Instances									DigTwinAtEdu_PL0	CSIN 🔻	,di
ignals											_
MCD Signals (15)			^		External Signa	s (15)					1
Find	Match Case	Match Whole Word	i 🌩		Find			Match Case	Match Whole	Word	٠
Name	Adapter Name	IO Type	Di	8-9 1	Name		Ю Туре	Data Type	Mar	ppir	
osWorkpieceCube_SetActive	saSortingPlant	Input	^		csLimitSwitchCylinderNotExtended_Activated		Input	bool	1	^	
osWorkpieceCylinder_SetActive	saSortingPlant	Input			csLimitSwitchCylinderRetracted_Activated		Input	bool	1		
csLightSensorCube_Detected	saSortingPlant	Output			osWorkpieceCylinder_SetActive		Output	bool	1		
csLightSensorCylinder_Detected	saSortingPlant	Output			osWorkpieceCube_SetActive		Output	bool	1		
<	- · · · ·	<b>•</b> • •	>		<			<b>^</b> · ·		ं>	Ť
			Do A	uto N	Napping						-
1apped Signals											
Connection Name		1	MCD Si	gnal l	Name	Direction	External Sigr	nal Name	Owner Compon	ent M	4
PLCSIM Adv.DigTwinAtEdu_PLCSIN	Λ									^	1
🖌 🗸 saSortingPlant_osWorkpieceCu	be_SetActive_osWorkpiece	Cube_SetActive	sWork	piece	Cube_SetActive	←	osWorkpiece	Cube_SetActive			
saSortingPlant_osWorkpieceCylinder_SetActive_osWorkpieceCylinder_Set			sWorkpieceCylinder_SetAc osWorkpieceCylinder_SetAc								
C 100 Control of the second										>	
		C	heck fo	or N->	1 Mapping						
							(1)		OK	Can	col
							4	/	UK	Cano	lei

图 22: 确认动态模型和虚拟 PLC 之间的信号分配

这样我们就在 NX/MCD 中的动态 3D 模型与虚拟 PLC 中的自动化程序之间建立了连接。通过单击

"Save" (保存) 按钮 🔜 来保存模型。

#### 7.3 使用虚拟 PLC 测试数字化双胞胎

本章我们将配合运行数字化双胞胎与虚拟 PLC 中的自动化程序,并验证其功能。请按照下列步骤 执行:

- → 在将<u>章节 7.2</u>中的自动化程序加载到虚拟 PLC 实例中之后,使用仿真工具 "WinCC Runtime Advanced" 启动 HMI。通过 TIA 博途完成这一操作。请采用数字化双胞胎@教育培训研讨会 系列模块 1 章节 7.4 中的过程。
- → 然后切换到"Mechatronics Concept Designer"程序,并启动数字化双胞胎的模拟。在
   "Simulation"(模拟)菜单栏中按下 "Play" (启动)命令。
- → 为数字化双胞胎执行本系列研讨会第一个模块中的两个测试方案,并验证数字化双胞胎的功能。 请遵循本研讨会系列模块 1 章节 7.6 中的描述。可以看到,我们在本研讨会系列的模块 4-6 所 创建的数字化双胞胎模型的行为与用于前三个模块的预设模型的行为相同。在系列测试后请停 止 MCD 中的模拟,结束模拟的 HMI 实例并关闭虚拟 PLC。

当然现在也可以使用我们在模块3中优化的自动化程序检查数字化双胞胎。

现在我们已经完成了本培训模块。利用所获得的知识,我们现在可以独立创建自己的数字化双胞胎并为自动化项目进行虚拟调试了。

## 8 检查清单 – 步骤说明

以下检查清单用于帮助培训人员/学生们独立检查是否已认真完成了步骤说明中的所有工作步骤并 支持其独立完成该模件的学习。

编号	说明	已检查
1	已使用所需的信号成功扩展了模块 5 中的动态模型。	
2	已在数字化双胞胎和虚拟 PLC 之间已建立了有效的信号连接。	
3	通过对本系列研讨会模块 1 中的测试场景进行模拟,可以成功、 完整地验证自行创建的数字化双胞胎。	

表 1: "在 CAE 系统 Mechatronics Concept Designer 中为动态 3D 模型生成信号"的检查清单

## 9 更多相关信息

为帮助您进行入门学习或深化学习,您可以找到更多指导信息作为辅助学习手段,例如:入门指南、视频、辅导材料、APP、手册、编程指南及试用版软件/固件,单击链接:

#### 预览"更多相关信息"-正在准备中

- 以下列出一些相关链接供参考:
- [1] <u>support.industry.siemens.com/cs/document/90885040/programming-guideline-for-s7-1200-s7-1500?dti=0&lc=en-US</u>
- [2] support.industry.siemens.com/cs/document/109756737/guide-to-standardization?dti=0&lc=en-US
- [3] omg.org/spec/UML/2.5.1/PDF
- [4] geeksforgeeks.org/unified-modeling-language-uml-activity-diagrams/
- [5] geeksforgeeks.org/unified-modeling-language-uml-state-diagrams/

### 更多相关信息

西门子自动化教育合作项目 siemens.com/sce

#### SCE 学习/培训文档 siemens.com/sce/documents

SCE 培训包 siemens.com/sce/tp

SCE 联系人 siemens.com/sce/contact

数字化企业 siemens.com/digital-enterprise

全集成自动化 (TIA) siemens.com/tia

TIA 博途 siemens.com/tia

TIA 选型工具 siemens.com/tia/tia-selection-tool

SIMATIC 控制器 siemens.com/controller

SIMATIC 技术文档 siemens.com/simatic-docu

工业支持中心 support.industry.siemens.com

产品目录和在线订购系统网上商城 mall.industry.siemens.com

Siemens 数字化工厂,FA P.O. Box 4848 90026 Nürnberg Germany

如有改动和错误, 恕不另行通知 © Siemens 2020

siemens.com/sce