

# 西门子PLC模块授权总经销商 6ES7155-6AR00-0AN0 ET 200SP PROFINET 接口模块

产品名称	西门子PLC模块授权总经销商 6ES7155-6AR00-0AN0 ET 200SP PROFINET 接口模块
公司名称	浔之漫智控技术(上海)有限公司
价格	.00/件
规格参数	西门子:全国授权销售 ET200SP:全新 德国:现货
公司地址	上海市松江区石湖荡镇塔汇路755弄29号1幢一层 A区213室
联系电话	15801997124 15801997124

## 产品详情

西门子PLC模块授权总经销商 6ES7155-6AR00-0AN0 ET 200SP PROFINET 接口模块

6ES7155-6AR00-0AN0

SIMATIC ET 200SP, PROFINET 接口模块 IM  
155-6PN Basic, \*多 12 个周边模块, 2x 集成的 RJ45  
插槽 包括服务器模块

发生启动事件时将导致以下响应：对于已分配 OB 的事件，将触发执行所分配的 OB。该事件将按照优先级在队列中排列。如果事件来自尚未分配 OB 的事件源，则 CPU 将执行默认的系统响应。说明 某些事件源无需组态就已存在（如启动、拔出/插入）。事件源与 OB 间的分配 OB 类型将确定 OB 与事件源间分配的位置：

对于硬件中断和等时同步模式中断，将在配置硬件或创建 OB 时进行分配。在 MC-Servo、MC-PreServo、MC-PostServo 和 MC-Interpolator 中，STEP 7 将在添加工艺对象后自动分配 OB 91/92。对于其它所有的 OB 类型，将在创建 OB 时（也可能在组态事件源后）进行分配。

对于硬件中断，可在指令 ATTACH 和 DETACH 的运行期间对之前的分配进行更改。在这种情况下，只更改实际有效的分配，而不是已组态的分配。组态的分配将在加载后以及每次启动时生效。将忽略那些组态中没有分配 OB 的硬件中断以及 DETACH 指令后发生的硬件中断。当一个事件到达时，CPU 不会检查是否为该事件分配了一个 OB，而只在实际执行硬件中断之前进行检查。OB 优先级和运行时特性 如果 OB 被分配给事件，则 OB 将拥有该事件的优先级。CPU 支持的优先级从 1（\*低）到 26（\*高）。以下条目对于事件处理必不可少：调用和处理已分配的 OB 更新已分配 OB 的过程映像分区 用户程序只按优先级处理 OB。这意味着同时发出多个 OB

请求时，程序将首先处理优先级\* 高的 OB。如果所发生事件的优先级高于当前执行的 OB，则中断此 OB 的执行。对于优先级相同的事件，用户程序按发生的时间顺序进行处理。说明 通信通信（例如，采用 PG 的测试功能）始终具有优先级 15。为了避免时间关键型应用中的程序运行时间发生不必要的延长，应确保这些 OB 不会被通信中断。为这些 OB 分配 >15 的优先级。参考有关组织块的更多信息，请参见 STEP 7 在线帮助。142 分布式 I/O 系统系统手册, 11/2022, A5E03576855-AM 程序执行的基本知识 9.1 事件和 OB 9.2 异步指令 简介

在程序处理过程中，同步和异步指令有着显著不同。

“同步”和“异步”属性与指令调用与执行间的时间顺序相关。

以下情况适用于同步指令：同步指令调用结束时，指令执行也结束。

而异步指令，则情况有所不同：异步指令调用结束时，异步指令的执行不一定结束。这也就意味着，异步指令的执行可以跨多次调用。在 CPU 中，异步指令的执行与循环用户程序同时执行。异步指令会在 CPU 中生成作业进行处理。

异步指令通常用于数据的传输（模块的数据记录、通信数据、诊断数据）。

同步/异步指令之间的不同之处

下图显示了异步指令和同步指令处理的不同之处。在该图中，指令完成之前（如，完全传输数据记录），异步指令指令调用了五次。使用同步指令，在每次调用中全面执行该指令。

第一次调用异步指令，处理开始 中间调用异步指令，处理继续

\*后一次调用异步指令，处理终止 每次调用时，同步指令都完整执行 完整执行的持续时间 图

9-1 同步指令和异步指令之间的不同之处 并行处理异步指令作业 CPU

可并行执行多个异步指令作业。在以下情况下，CPU 将并行执行多个作业：

同时调用多个异步指令作业。未超出指令可并行运行的\*大作业数量。143 程序执行的基本知识 9.2

异步指令 分布式 I/O 系统系统手册, 11/2022, A5E03576855-AM 下图显示了两个 WRREC

指令作业的并行处理。在该图中，在一个特定的周期内并行执行这两个指令。:55(&B :55(&B

:55(&B :55(&B :55(&B :55(&B 图 9-2 并行执行异步指令 WRREC 说明 异步指令之间的依赖关系

用户程序中的调用顺序可能不同于异步指令的处理顺序。这可能导致异步指令之间的依赖关系出现问题。

解决方案：为确保能够正确地按先后顺序进行处理，请在顺控器中使用异步指令的状态输出。

仅当异步指令已完成且已通过参数 DONE 确认后，才能开始执行下一条异步指令。示例：对于 RecipeImport 和 RecipeExport 配方功能，需要使用 CSV 文件存储配方数据。如果导入和导出时使用同一 CSV 文件，则两个异步语句会建立相互依赖关系。在顺控器中，将 RecipeImport 指令的参数 DONE 状态关联到将执行 RecipeExport 的下一步。进行此关联后，可确保正确进行处理。指令作业的调用分配要跨多个调用执行一个指令，CPU 需向该指令正在运行的作业唯一分配一个后续调用。CPU 可通过以下两种方式为作业分配一个调用，具体取决于指令的类型：

通过指令的背景数据块（“SFB”类型）

使用指令的输入参数标识该作业。在异步指令的执行过程中，这些输入参数必须与执行过程中的各调用相匹配。示例：通过输入参数 LOW\_LIMIT、UP\_LIMIT、COUNT、ATTRIB 和 SRCBLK，标识指令作业“Create\_DB”下表列出指令与用于标识的输入参数的关系。指令标识作业的输入参数 DPSYC\_FR LADDR、GROUP、MODE D\_ACT\_DP LADDR DPNRM\_DG LADDR WR\_DPARM LADDR、RECNUM WR\_REC LADDR、RECNUM RD\_REC LADDR、RECNUM CREATE\_DB LOW\_LIMIT、UP\_LIMIT、COUNT、ATTRIB、SRCBLK READ\_DBL SRCBLK、DSTBLK 144

分布式 I/O 系统系统手册, 11/2022, A5E03576855-AM 程序执行的基本知识 9.2 异步指令 指令

标识作业的输入参数 WRIT\_DBL SRCBLK、DSTBLK RD\_DPARA LADDR、RECNUM DP\_TOPOL DP\_ID

异步指令的状态 异步指令通过块参数 STATUS/RET\_VAL 和 BUSY

显示指令的状态。有些异步指令也会使用块参数 DONE 和 ERROR 显示。下图显示了两个异步指令 WRREC 和 CREATE\_DB. 的执行 输入参数 REQ 用于启动作业，执行异步指令。

输出参数 DONE 用于指示该作业已完成且无错误。 输出参数 BUSY

用于指示作业是否正在执行。BUSY =1 时，为该异步指令分配资源。BUSY = 0 时，未分配资源。

输出参数 ERROR 用于指示发生了错误。 输出参数 STATUS/RET\_VAL

用于提供有关作业执行的状态信息。发生错误后，输出参数 STATUS/RET\_VAL 用于接收错误信息。图

9-3 指令 WRREC 和 CREATE\_DB 示例中，异步指令的块参数说明。总结

下表简要列出了上文中介绍的参数关系。在该表格中，还特别列示了调用后但过程尚未执行完

成时可能的输出参数值。说明 每次调用之后，需对用户程序中的相关输出参数进行评估。

作业“运行”过程中，REQ、STATUS/RET\_VAL、BUSY 和 DONE 间的相互关系。调用的顺序号调用的类型 REQ STATUS/RET\_VAL BUSY DONE ERROR 1 第一次调用 1 W#16#7001 1 0 0 错误代码（如，W#16#80C3 表示资源不足）0 0 1 145 程序执行的基本知识 9.2 异步指令 分布式 I/O 系统 系统手册, 11/2022, A5E03576855-AM 调用的顺序号调用的类型 REQ STATUS/RET\_VAL BUSY DONE ERROR 2 到 (n - 1) 中间调用 - W#16#7002 1 0 0 n \* 后一次调用 - W#16#0000（如果无错误）0 1 0 错误代码（如果出错）0 0 1 资源占用 异步指令在执行过程中，会占用 CPU 的资源。根据 CPU 和指令的类型不同，这些资源的使用具有一定限值；CPU 同时可处理\*大数目的异步指令作业。在作业成功完成后或在执行过程中发生错误，这些资源将再次可用。示例：对于 RDREC 指令，1512SP1 PN CPU \*多可并行处理 20 个作业。如果超出指令的\*大并行作业数量，则会出现以下情况：该指令将在块参数 STATUS 中返回错误代码 80C3（资源不足）。CPU 不执行作业直到资源再次释放。说明 低层级的异步指令多个异步指令可使用一个或多个低层级的异步指令进行处理。下表列出了这种相关性。请注意，如果有多个低层级指令，通常情况下一次只占用一个。扩展指令：可并行运行的\*大作业数量 表格 9-2 所用的扩展异步指令和级别较低的指令可同时运行的\*大作业数 扩展指令 CPU 1510SP-1 PN CPU 1510SP F-1 PN CPU 1512SP-1 PN CPU 1512SP F-1 PN 分布式 I/O RDREC 20 RD\_REC 10 WRREC 20 WR\_REC 10 D\_ACT\_DP 8 ReconfigIOSystem 使用 RDREC、WRREC、D\_ACT\_DP、DPSYC\_FR 2 DPNRM\_DG 8 DP\_TOPOL 1 ASI\_CTRL 使用 RDREC、WRREC PROFenergy PE\_START\_END 使用 RDREC、WRREC PE\_CMD 使用 RDREC、WRREC