基于Persistent的数据存储介绍

**王建成、陈利君、孔惠增**

Version: 1.0.3

Date: 2025.2.26

目录

[1. 前言： 3](#_Toc194306422)

[2. TwinCAT 3系统的数据存储： 3](#_Toc194306423)

[2.1. 确定PLC Project的ADS端口号； 4](#_Toc194306424)

[2.2. 数据恢复； 4](#_Toc194306425)

[2.3. 常数设置； 5](#_Toc194306426)

[2.4. Task创建； 6](#_Toc194306427)

[2.5. Clear invalid persistent data不可勾选 6](#_Toc194306428)

[2.6. CrcValue 定义 6](#_Toc194306429)

[2.7. Persistent使用注意事项 6](#_Toc194306430)

[2.8. Persistent文件1M可以存储多少数据 7](#_Toc194306431)

[2.9. FB\_Persistent功能引脚介绍 8](#_Toc194306432)

[2.10 程序调用效果 10](#_Toc194306433)

[2.11 程序示例 10](#_Toc194306434)

[3. TwinCAT 2系统的数据存储： 11](#_Toc194306435)

[3.1. 确定PLC Project的ADS端口号； 11](#_Toc194306436)

[3.2. 数据恢复； 12](#_Toc194306437)

[3.3. 常数设置； 12](#_Toc194306438)

[3.4. Task创建； 13](#_Toc194306439)

[3.5. Clear invalid persistent data不可勾选 13](#_Toc194306440)

[3.6. CrcValue 定义 13](#_Toc194306441)

[3.7. Persistent使用注意事项 14](#_Toc194306442)

[3.8. Persistent文件1M可以存储多少数据 14](#_Toc194306443)

[3.9. FB\_Persistent功能引脚介绍 15](#_Toc194306444)

[3.10. 程序调用效果 17](#_Toc194306445)

[3.11. 程序示例 17](#_Toc194306446)

## 前言：

Persistent数据类型在工业自动化项目的实施过程中，被广泛应用于实时过程数据的记录与存储。本文将针对以下四种系统环境进行详细分析与比较：TwinCAT 3的CE系统、TwinCAT 3的非CE系统（包括Windows 10和Windows 7等操作系统）、TwinCAT 2的CE系统以及TwinCAT 2的非CE系统。

注意：本文是以UPS容量足够，工作正常为前提。

## TwinCAT 3系统的数据存储：

该部分主要针对TwinCAT 3系统的时候，如何合理通过调用FB\_Persistent功能块对数据进行管理。

表格

描述已自动生成

在TwinCAT 3系统中，合理调用FB\_Persistent功能块是进行数据管理的有效方式。系统启动时会加载Port\_851.bootdata文件以恢复persistent数据。当Port\_851.bootdata文件损坏时，系统将自动加载Port\_851.bootdata-old文件。需要注意的是，由于Port\_851.bootdata-old文件是上次系统启动时生成的备份文件，其数据可能存在一定滞后性。

通过调用FB\_Persistent功能块，系统将在设定周期内定时生成新的Port\_851.bootdata文件，随后删除旧的Port\_851.bootdata-old文件，并将新生成的bootdata文件重命名为Port\_851.bootdata-old。这一机制确保了即使主数据文件损坏，系统仍可获得相对较新的备份数据。

此外，FB\_Persistent功能块还提供多版本备份功能，可生成7份历史备份文件。这种多重保护机制有效增强数据的安全性，即使Port\_851.bootdata和Port\_851.bootdata-old文件同时损坏，系统仍可从历史备份中恢复数据。

对不同系统环境，数据恢复方案有所差异：

对于Windows 10、Windows 7等非CE系统，可以通过FB\_Persistent的restore功能实现自动恢复；

对于CE系统，若发生数据丢失，则需要部分人工干预，具体恢复流程详见2.2节数据恢复部分。

### 2.1. 确定PLC Project的ADS端口号；

在TwinCAT系统中，用户可通过右键单击PLC对象来查看或修改其ADS端口号。系统新建的PLC项目默认使用851作为端口号。当项目中包含多个PLC实例时，需要为每个PLC实例分别调用独立的FB\_Persistent。

功能块会自动识别当前PLC项目的端口号。

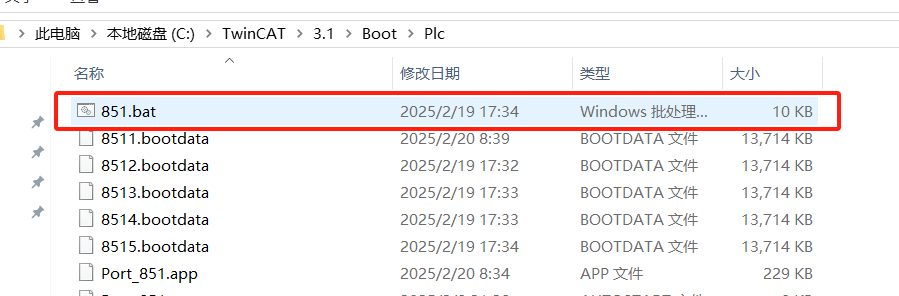
图形用户界面, 文本, 应用程序

描述已自动生成

### 2.2. 数据恢复；

对于win10、win7等非CE系统而言：

在进行数据恢复时， FB\_Persistent功能块指定路径C:\TwinCAT\3.1\Boot\Plc\中自动生成所需的bat文件。该bat文件以当前的端口号命名。



该bat文件的主要功能是在数据丢失时自动执行恢复操作，批处理命令依次包含：

* + - 1. 系统切换至配置模式
      2. 等待30秒后自动执行bat文件
      3. 删除boot目录下的Port\_851.bootdata和Port\_851.bootdata-old文件
      4. 将备份文件8511.bootdata重命名为Port\_851.bootdata
      5. 重新启动控制器

对于CE系统而言：



数据恢复操作需要通过远程桌面连接访问CE系统，然后执行如下步骤：

* 1. 定位至控制器的Boot目录
  2. 删除以下两个文件：

Port\_851.bootdata

Port\_851.bootdata-old

* 1. 将备份文件8511.bootdata重命名为Port\_851.bootdata
  2. 手动执行控制器重启操作

### 2.3. 常数设置；

在系统配置中，需要设置以下四个关键参数：

* 1. 最大扫描文件数（MAX\_SCAN\_FILES\_NUM）

默认值：20个文件

功能：控制系统最大可扫描的文件数量

* 1. 最大Persistent数据备份文件数（MAX\_BAK\_FILES\_NUM）

默认值：7个备份文件

取值范围：1至9（必须小于10）

功能：控制Persistent数据备份的最大数量

* 1. 将Twincat切为Config模式后执行bat文件的延时时间（TIME\_TO\_CONFIG）

默认值：3

取值范围：1至9（必须小于10）

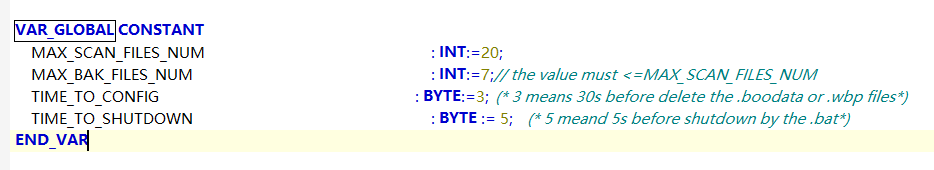
功能：默认值3代表30s，若改为5，则代表50s

* 1. 执行完bat后重启控制器的延时时间（TIME\_TO\_SHUTDOWN）

默认值：5

取值范围：1至9（必须小于10）

功能：5代表5s



### 2.4. Task创建；

创建一个20ms的task，该功能块应该被放在一个20ms的task中进行周期调用

### 2.5. Clear invalid persistent data不可勾选

注意！！！！！该选项不可勾选，否则会让TC忽略掉.bootdata-old文件，导致数据丢失。

图形用户界面, 应用程序, Word

描述已自动生成

### 2.6. CrcValue 定义

在gvl中定义Persistent变量CrcValue，初值赋值为0，用来做数据校验。该步骤是必须的。

图形用户界面, 应用程序

描述已自动生成

### 2.7. Persistent使用注意事项

* 1. 在项目调试过程中，如果点击这个图标，必定会导致包含Persisent在内的所有变量复位。

图形用户界面, 应用程序, Teams

描述已自动生成

* 1. 切勿重命名包含Persistent变量的全局变量文件或者子程序

在TwinCAT 3中，全局的Persistent变量保存时包含了文件名，例如GVL和GVL\_01下的同名变量，在.bootdata中是不同的。

子程序中Persistent变量保存时也包含了程序名，例如Main.iVar1。如果重命名了Persistent变量的全局变量文件或者子程序，例如将Main重命名为PRG01，正常关机时TwinCAT写入新的Persistent文件就只会写入PRG01.iVar1，而不包含Main.iVar1。

* 1. Persistent变量使用建议

1. 尽量使用数组类型，以提高有效数据占比
2. 尽量集中放置，最好放在全局变量中，避免自己找不到
3. 避免随处随意新增Persistent变量，有计划地使用变量，避免最后再来清理
   1. 配方数据务必使用XML

实际的项目开发和调试人员可能是TwinCAT经验丰富的资深工程师，也可能是TwinCAT初学者，在调试阶段无法完全避免误操作。一旦出现数据丢失，可能会慌张求助，也不一定能在最短时间内找到能提供本文所述帮助的人。甚至在重命名了程序，自己又不记得是哪些程序，不知道哪些程序里面有保持型变量的时候，恢复Persistent数据将会非常困难。所以，重要的配方参数务必使用XML文件保存。

### 2.8. Persistent文件1M可以存储多少数据

由于TwinCAT的Persistent变量是以文件形式保存，其中不仅要保存变量的值，还要保存变量的名称，所以1M Byte的Persisent文件能保存多少个变量，以变量的名称、类型有很大关系。

在TwinCAT 3中，Persisent文件中每个字节顺序虽然也有大致的排列规则，变量名长度或者数组元素的增加，也会引起它在Persisent文件中的占用字节数有所增加，但TwinCAT 3中考虑了地址对齐等因素，每个变量的占用字节数只会以4为单位增减。

由此得出：

* 1. 以数组的形式保存Persisent变量效率最高

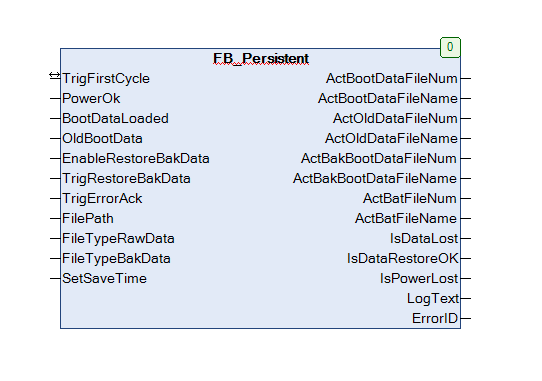
数组型的Persisent变量保存时只用1个变量名称，数组元素越多，其Data在所有占用空间中的比例越大。

* 1. 掉电保持数据是否超出1M文件，最好的办法是直接观察备份文件的属性

TwinCAT 3：C:\TwinCAT \3.1\Boot\Plc\Port\_851.bootdata-old'

之所以直接观察备份文件，是因为正常启动后.bootdata文件就被自动删除了。只有手动切到Config模式，才能看到正式的备份文件。启动后一旦从中装载了数据就会将其重命名为.bootdata-old文件。

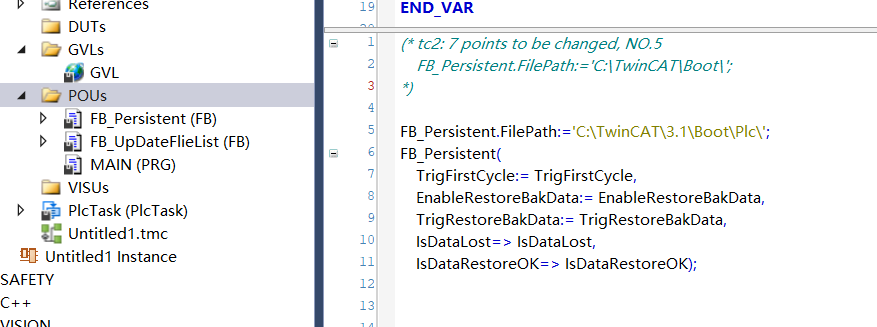
### 2.9. FB\_Persistent功能引脚介绍



该功能块主要有两大功能，第一是对persistent文件的备份文件进行更新，并且对persistent文件进行备份，第二是当由于UPS没有正常工作导致数据损坏后从persistent文件进行快速数据恢复。

|  |  |  |  |
| --- | --- | --- | --- |
| 引脚名字 | 引脚类型 | 数据类型 | 描述 |
| TrigFirstCycle | In\_out | Bool | 在一个新项目中，初始调用该功能块时候由于系统中没有Persistent文件，功能块会报数据丢失的错误，此时需要触发这个引脚生成一个Persistent文件，需要注意的是理论上该引脚在一个项目中只能被触发一次，需要给此引脚配备管理员权限 |
| PowerOk | InPut | Bool | 读取ups状态，将控制器UPS的外部供电正常信号引入，必须项！！！！！ |
| BootDataLoaded | InPut | Bool | 保留 |
| OldBootData | InPut | Bool | 保留 |
| EnableRestoreBakData | InPut | Bool | 开启数据恢复，当开启数据恢复后，数据的自动保存功能自动停止运行，该按钮应该设置为管理员权限 |
| TrigRestoreBakData | InPut | Bool | 当开启数据恢复按钮后，常安此按钮10s开始进行数据恢复，触发后系统会自动重启，进行数据恢复，慎重操作 |
| TrigErrorAck | InPut | Bool | 报错复位，通常与系统的错误复位按钮相连 |
| FilePath | InPut | String | 对应boot文件的路径，一般来说为C:\TwinCAT\3.1\Boot\Plc\ |
| FileTypeRawData | InPut | String | Persistent数据的原始文件格式，一般为.bootdata |
| FileTypeBakData | InPut | String | Persistent数据的备份文件格式，一般为.bootdata-old |
| SetSaveTime | InPut t | Real | 自动备份Persistent数据文件和更新备份文件的间隔，单位为s，默认值为10s，可设置范围10-10800 |
| ActBootDataFileNum | Out | Int | 当前实际Persistent数据的bootdata文件数量 |
| ActBootDataFileName | Out | String | 当前实际Persistent数据的bootdata文件名字 |
| ActOldDataFileNum | Out | Int | 当前实际Persistent数据的bootdata-old文件数量 |
| ActOldDataFileName | Out | String | 当前实际Persistent数据的bootdata-old文件名字 |
| ActBakBootDataFileNum | Out | Int | 当前实际Persistent数据的bootdata文件备份的数量 |
| ActBakBootDataFileName | Out | String | 当前实际Persistent数据的bootdata文件备份的名字 |
| ActBatFileNum | Out | String | 当前boot文件夹中bat文件数量 |
| ActBatFileName | Out | String | 当前boot文件夹中bat文件名字 |
| IsDataRestoreOK | Out | Bool | 当参数初始化完成，且数据有效的情况下IsDataRestoreOK置为true |
| IsDataLost | Out | Bool | 当参数初始化失败或者即使成功的情况下，如果通过比较CrcValue不通过的情况下该引脚会置为true，此时设备不应该再有下一步操作，此时可以尝试EnableRestoreBakData来恢复数据 |
| IsPowerLost | Out | Bool | 指示当前设备已经断电 |
| LogText | Out | Bool | 日志 |
| ErrorID | Out | Udint | 16#8001：在恢复数据时删除bootdata文件报错  16#8002：在恢复数据时删除bootdaata-old文件报错  16#8003：在恢复数据时重命名文件报错  16#8004：恢复数据报错，因为无备份文件  16#8005：恢复数据报错，因为重启系统失败  16#8006：自动保存时，生成bootdata文件报错  16#8007：自动保存时，删除备份bootdata文件报错  16#8008：自动保存时，删除bootdata-old文件报错  16#8009：自动保存时，重命名bootdata文件报错  16#800A：自动保存时，重命名bootdata-old文件报错  16#800B：恢复数据时，创建bat文件报错 |

### 2.10 程序调用效果



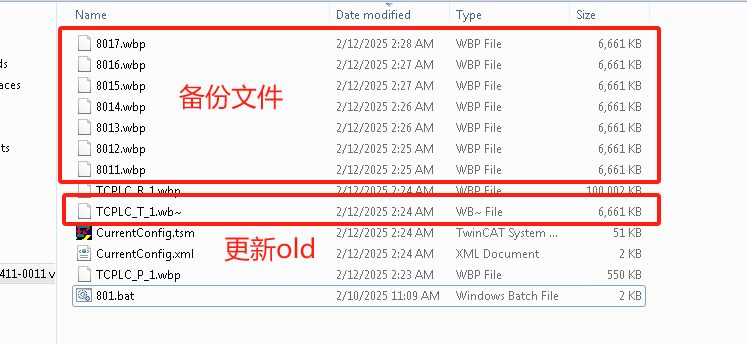
### 2.11 程序示例

该项目中包含了FB\_Persistent的源代码，以及调用示例，可将该文件复制到本地电脑解压后查看。



## TwinCAT 2系统的数据存储：

该部分主要针对TwinCAT 2系统的时候，如何合理通过调用FB\_Persistent功能块对数据进行管理。



系统启动时，TwinCAT（TC） 将自动加载 TCPLC\_T\_1.wbp 文件，以恢复 persistent 数据。若 TCPLC\_T\_1.wbp 文件损坏，系统会自动加载备份文件 TCPLC\_T\_1.wb~。需要注意的是，TCPLC\_T\_1.wb~ 文件是在上次系统启动时生成的备份文件，因此其数据可能存在一定滞后性，导致恢复的数据并非最新状态。

通过调用 FB\_Persistent 功能块，系统会按照设定周期定时生成 TCPLC\_T\_1.wbp 文件（即 persistent 数据文件）。随后，系统会删除 TCPLC\_T\_1.wb~ 文件，并将新生成的 wbp 文件重命名为 TCPLC\_T\_1.wb~。这一机制确保即使 wbp 文件损坏，系统仍然保留一个较新的备份文件，以便进行数据恢复。

此外，FB\_Persistent 功能块还会额外生成 7 份备份文件，即使因各种原因导致 TCPLC\_T\_1.wbp 和 TCPLC\_T\_1.wb~ 均损坏，系统仍可使用这 7 份数据备份文件进行恢复。

对不同系统环境，数据恢复方案有所差异：

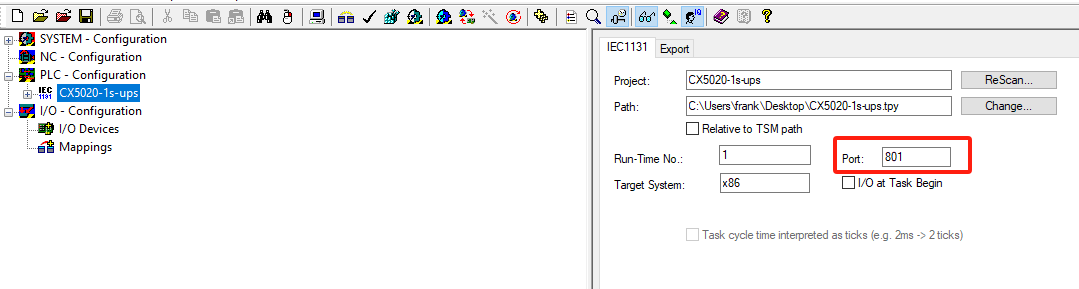
对于Windows 10、Windows 7等非CE系统，可以通过FB\_Persistent的restore功能实现自动恢复；

对于CE系统，若发生数据丢失，则需要部分人工干预，具体恢复流程详见3.2节数据恢复部分。

### 3.1. 确定PLC Project的ADS端口号；

要查看 PLC 的 ADS 端口号，请注意系统新建的 PLC 项目 默认使用端口号 801。

如果一个 TwinCAT 项目中包含 多个 PLC，则需要调用相应数量的 FB\_Persistent 功能块，并在每个功能块的对应接口中填写正确的端口号，以确保数据持久化功能能够正确应用于各个 PLC 实例。功能块会自动识别当前PLC项目的端口号。



### 3.2. 数据恢复；

对于win10、win7等非CE系统而言：

在进行数据恢复时， FB\_Persistent功能块指定路径C:\TwinCAT\Boot\中自动生成所需的bat文件。该bat文件以当前的端口号命名。

.bat批处理文件自动执行的命令如下：

1. 系统切换至配置模式
2. 等待30秒后自动执行bat文件
3. 删除boot目录下的TCPLC\_T\_1.wbp和TCPLC\_T\_1.wb~
4. 将备份文件8011.wbp重命名为TCPLC\_T\_1.wbp
5. 重启控制器，完成数据恢复

对于CE系统而言：



数据恢复操作需要通过远程桌面连接访问CE系统，具体步骤如下：

1.定位至控制器的Boot目录

2.删除以下两个文件：

TCPLC\_T\_1.wbp

TCPLC\_T\_1.wb~

3.将备份文件8011. wbp重命名为TCPLC\_T\_1.wbp

4.手动执行控制器重启操作

注意：对于TwinCAT 2来说，如果端口号为811则对应的文件为TCPLC\_T\_2.wbp和TCPLC\_T\_2.wb~。

### 3.3. 常数设置；

在系统配置中，需要设置以下两个关键参数：

1. 最大扫描文件数（MAX\_SCAN\_FILES\_NUM）

默认值：20个文件

功能：控制系统最大可扫描的文件数量

1. 最大Persistent数据备份文件数（MAX\_BAK\_FILES\_NUM）

默认值：7个备份文件

取值范围：1至9（必须小于10）

功能：控制Persistent数据备份的最大数量

注意事项：

两个参数均需在系统常数中进行配置

MAX\_BAK\_FILES\_NUM的取值必须严格小于10

建议根据实际存储容量和应用需求合理配置参数值

形状, 矩形

描述已自动生成

### 3.4. Task创建；

创建一个20ms的task，该功能块应该被放在一个20ms的task中进行周期调用

### 3.5. Clear invalid persistent data不可勾选

注意！！！！！该选项不可勾选，否则会让TC忽略掉.wb~文件，导致数据丢失。

图形用户界面, 文本, 应用程序

描述已自动生成

### 3.6. CrcValue 定义

在gvl中定义Persistent变量CrcValue，初值赋值为0，用来做数据校验。该步骤是必须的。

图形用户界面, 文本, 应用程序

描述已自动生成

### 3.7. Persistent使用注意事项

1. 在项目调试过程中，如果点击这个图标，必定会导致包含Persisent在内的所有变量复位。

图形用户界面, 文本, 应用程序

描述已自动生成

1. 切勿重命名包含Persistent变量的子程序

在TwinCAT 2中，全局的Persistent变量保存时不包含文件名，例如Global\_Variables和Global\_01下的同名变量，在.wbp中是相同的。

子程序中Persistent变量保存时则包含了程序名，例如Main.iVar1。如果重命名了Persistent变量的全局变量文件或者子程序，例如将Main重命名为PRG01，正常关机时TwinCAT写入新的Persistent文件就只会写入PRG01.iVar1，而不包含Main.iVar1。

1. Persistent变量使用建议
2. 尽量使用数组类型，以提高有效数据占比
3. 尽量集中放置，最好放在全局变量中，避免自己找不到
4. 避免随处随意新增Persistent变量，有计划地使用变量，避免最后再来清理
5. 配方数据务必使用XML

实际的项目开发和调试人员可能是TwinCAT经验丰富的资深工程师，也可能是TwinCAT初学者，在调试阶段无法完全避免误操作。一旦出现数据丢失，可能会慌张求助，也不一定能在最短时间内找到能提供本文所述帮助的人。甚至在重命名了程序，自己又不记得是哪些程序，不知道哪些程序里面有保持型变量的时候，恢复Persistent数据将会非常困难。所以，重要的配方参数务必使用XML文件保存。

### 3.8. Persistent文件1M可以存储多少数据

由于TwinCAT的Persistent变量是以文件形式保存，其中不仅要保存变量的值，还要保存变量的名称，所以1M Byte的Persisent文件能保存多少个变量，以变量的名称、类型有很大关系。

在TwinCAT 2中，Persisent文件中每个字节顺序的排列规则是明确的，变量名长度或者数组元素的增加，就会引起它在Persisent文件中的占用字节数对应增加。

由此得出：

1. 以数组的形式保存Persisent变量效率最高

数组型的Persisent变量保存时只用1个变量名称，数组元素越多，其Data在所有占用空间中的比例越大。

1. 掉电保持数据是否超出1M文件，最好的办法是直接观察备份文件的属性

TwinCAT 2：C:\TwinCAT \Boot\TCPLC\_T\_1.wb~'

之所以直接观察备份文件，是因为正常启动后. wbp文件就被自动删除了。只有手动切到Config模式，才能看到正式的备份文件。启动后一旦从中装载了数据就会将其重命名为. wbp~文件。

### 3.9. FB\_Persistent功能引脚介绍

表格

描述已自动生成

该功能块主要有两大功能，第一是对persistent文件的备份文件进行更新，并且对persistent文件进行备份，第二是当由于UPS没有正常工作导致数据损坏后从persistent文件进行快速数据恢复。

|  |  |  |  |
| --- | --- | --- | --- |
| 引脚名字 | 引脚  类型 | 数据  类型 | 描述 |
| TrigFirstCycle | In\_out | Bool | 在一个新项目中，初始调用该功能块时候由于系统中没有Persistent文件，功能块会报数据丢失的错误，此时需要触发这个引脚生成一个Persistent文件，需要注意的是理论上该引脚在一个项目中只能被触发一次，需要给此引脚配备管理员权限 |
| PowerOk | InPut | Bool | 读取ups状态，将控制器UPS的外部供电正常信号引入，必须项！！！！！ |
| BootDataLoaded | InPut | Bool | 保留 |
| OldBootData | InPut | Bool | 保留 |
| EnableRestoreBakData | InPut | Bool | 开启数据恢复，当开启数据恢复后，数据的自动保存功能自动停止运行，该按钮应该设置为管理员权限 |
| TrigRestoreBakData | InPut | Bool | 当开启数据恢复按钮后，常安此按钮10s开始进行数据恢复，触发后系统会自动重启，进行数据恢复，慎重操作 |
| TrigErrorAck | InPut | Bool | 报错复位，通常与系统的错误复位按钮相连 |
| FilePath | InPut | String | 对应boot文件的路径，一般来说为C:\TwinCAT\Boot\ |
| FileTypeRawData | InPut | String | Persistent数据的原始文件格式，一般为.wbp |
| FileTypeBakData | InPut | String | Persistent数据的备份文件格式，一般为.wb~ |
| SetSaveTime | InPut | Real | 自动备份Persistent数据文件和更新备份文件的间隔，单位为s，默认值为3600s，也就是1h，可设置范围3600-10800 |
| ActBootDataFileNum | Out | Int | 当前实际Persistent数据的wbp文件数量 |
| ActBootDataFileName | Out | String | 当前实际Persistent数据的wbp文件名字 |
| ActOldDataFileNum | Out | Int | 当前实际Persistent数据的wb~文件数量 |
| ActOldDataFileName | Out | String | 当前实际Persistent数据的wb~文件名字 |
| ActBakBootDataFileNum | Out | Int | 当前实际Persistent数据的wbp文件备份的数量 |
| ActBakBootDataFileName | Out | String | 当前实际Persistent数据的wbp文件备份的名字 |
| IsDataRestoreOK | Out | Bool | 当参数初始化完成，且数据有效的情况下IsDataRestoreOK置为true |
| IsDataLost | Out | Bool | 当参数初始化失败或者即使成功的情况下，如果通过比较CrcValue不通过的情况下该引脚会置为true，此时设备不应该再有下一步操作，此时可以尝试EnableRestoreBakData来恢复数据 |
| IsPowerLost | Out | Bool | 指示当前设备已经断电 |
| LogText | Out | Bool | 日志 |
| ErrorID | Out | Udint | 16#8001：在恢复数据时删除bootdata文件报错  16#8002：在恢复数据时删除bootdaata-old文件报错  16#8003：在恢复数据时重命名文件报错  16#8004：恢复数据报错，因为无备份文件  16#8005：恢复数据报错，因为重启系统失败  16#8006：自动保存时，生成bootdata文件报错  16#8007：自动保存时，删除备份bootdata文件报错  16#8008：自动保存时，删除bootdata-old文件报错  16#8009：自动保存时，重命名bootdata文件报错  16#800A：自动保存时，重命名bootdata-old文件报错  16#800B：恢复数据时，创建bat文件报错 |

### 3.10. 程序调用效果

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

### 3.11. 程序示例

该项目中包含了FB\_Persistent的源代码，以及调用示例，可将该文件复制到本地电脑解压后查看。

