

New Automation Technology

TwinCAT 数据保存指南 (V1.0)



TwinCAT 中数据保存的方法

TwinCAT 中数据保存有多种方式，客户必须根据自己的实际情况合理选择。目前使用较多的保存方式大体可以分为如下四种：

- 通过 NOVRAM 进行保存
- 通过 Persistent 变量进行保存
- 通过文件读写的方式来进行保存
- 借助数据库来进行保存

一. NOVRAM 保存

NOVRAM 是一个特殊的存储器组件，一个 NOVRAM 模块由两部分组成，一个是易失性存储器，一个是非易失性存储器。通常情况下，TwinCAT 只是用易失性存储器中的数据，发生断电时，数据会自动从易失性存储器复制到非易失性存储器。这个过程需要的电量由一个板载电容来提供。一旦供电恢复，数据将自动从非易失性存储器复制回易失性存储中，以供 TwinCAT 继续使用，这整个过程是自动完成的。

可以使用该方法的前置条件：**配备 NOVRAM 硬件模块**，如以下几种情况：

- CX9020 自带 128kB 大小的 [NOVRAM](#)
- CX2020 自带 128kB 大小的 [NOVRAM](#)
- 工控机可选配件 [C9900-R266 NOVRAM PCIe 模块](#)，占用一个扩展槽，可使用 512kB 大小的 NOVRAM
- [EL6080 EtherCAT](#) 模块，安装后可使用 128kB 大小的 NOVRAM
- 关于具体某一款控制器/工控机是否配备有 NOVRAM，可以在官网搜索控制器型号，在详细参数栏目中进行查询
- **使用 NOVRAM 的优势：**
 - 配置简单，保存数据以及恢复数据的过程由硬件芯片自动完成，工程师通过变量链接的方式即可完成保持数据的配置
 - 工程师也可以使用配套的功能块 [FB_NovRamReadWrite](#) 来对 NOVRAM 进行手动的保存和配置，需要注意的是，如果通过手动触发来保存，只是保存了当前时刻的数据
- **使用 NOVRAM 的劣势：**
 - 由于数据是保存在板载的存储器上，保存数据容量有限，以 CX9020 为例，容量大小仅为 128kB，最大的 NOVRAM 模块 C9900-R266 也只能保存 512kB 大小数据

- 对于 EL6080 模块，需要通过 EtherCAT 总线来传输数据，如果 EtherCAT 通讯出现报错或故障，数据将无法保存
- 适合用 NOVRAM 来保存的数据：
 - 设备生产过程中，实时变化，需要实时保存的数据，总体数据容量较小
- 不适合用 NOVRAM 来保存的数据：
 - 设备运行过程中，不常发生变化，数据量较大，较少被修改的数据，如工艺配方参数，电机的零点位置，初始化时使用的默认参数等
- 配置样例：[倍福虚拟学院](#)

二. Persistent 变量

采用 Persistent 方式来保存变量的值是 TwinCAT 工程师们常用的一种数据保存方式。变量在 TwinCAT 意外终止，Reset Cold (TC3)、Reset (TC2) 或重新下载程序的情况下会自动恢复，在 PLC 环境中执行 Reset Origin (TC3) 或 Reset All (TC2) 后 Persistent 数据会恢复其初始值。

Persistent 变量通常搭配 UPS 来完成数据的自动保存，系统在识别到电源供应中断，启用 UPS 供电后，通知 TwinCAT 从 Run 切换到 Stop 状态，Persistent 数据在 TwinCAT 由 Run 切换到 Stop 期间系统自动执行保存动作。该保存动作为通过文件写入的方式，将数据写入到系统盘，保存在 [boot 文件夹](#) 下的文件中。保存完成之后，根据 UPS 配置的参数，对系统执行安全关机，或等待电能耗尽后自行关断。上电之后，系统将读取 Persistent 变量的保存文件，读取成功之后，会对其中的变量做恢复。

Persistent 数据不是放在一个特定的存储区域（如特定的 ROM 中）而是直接保存在内存里。Persistent 数据的恢复是从文件中直接拷贝信息到内存，因此内存与 Persistent 文件中的变量信息等必须保持完全一致。其中出现任何差异或错误都将是致命的，可能会使系统崩溃。为了应对这种情况，TwinCAT 为 Persistent 变量设置了一个签名，签名信息包括路径、名称、长度、类型等。当系统检查签名信息符合要求时，才会为变量执行恢复操作。这种操作方法可靠性更高，更安全。正常情况下，系统不会对 Persistent 数据进行清除操作，这些变量不会丢失。排除人为因素（比如人为进入 boot 文件夹删除了文件）。只有当 TwinCAT 检查变量签名信息，发现签名不匹配，此时变量不能正常恢复，系统会执行数据清除操作，以防止变量异常导致的系统崩溃。

- **可以使用该方法的前置条件：配备 UPS 硬件，且通过软件配置开启了 UPS 功能，如以下几种情况**
 - 对于 CX52, CX51, CX50 系列等带有板载 1s UPS 功能的嵌入式控制器（以 CX51 为例）：
 - CX51 系列工控机自带[板载 1s UPS](#)，通过电容进行储能和放电

- 以 CX51 系列嵌入式控制器为例，工程师需要使用 [FB S UPS CX51x0](#) 功能块，对 UPS 进行配置启用，系统才会调用板载 UPS 电容，来执行变量保存的动作
 - 注意，CX52 系列需要用 3.3.8.0 以上版本的 TC2_SUPS 中的 [FB S UPS BAPI](#) 功能块，使用方法同 CX51 系列控制器
- 对于 CX20 系列嵌入式控制器：
 - CX20 系列控制器不自带 UPS，需要通过额外的电源模块来增加 UPS 功能
 - CX20 系列的电源模块有多种型号：
 - [CX2100-0904](#) 带有内嵌的电容型 UPS
 - [CX2100-0914](#) 本身不带电容，但是可以搭配 [CX2900-0192](#) 电池模块来实现 UPS 功能
 - [CX2100-0024](#) 则需要搭配 [CU81xx](#) 系列 UPS 使用
 - 工程师需要使用操作系统中的软件 [Beckhoff UPS Configuration](#) 对 UPS 进行配置，方可以启用 UPS 功能，一般来说，通过显示器/远程桌面接入控制器的操作系统，可以在桌面或系统菜单中找到该软件，打开即可进行设置
- 对于工控机：
 - 常见的 C69 系列工控机需要选配 2 个配件来实现 UPS 功能：[C9900-U209](#)，[C9900-U330](#)
 - 工程师需要使用 [Beckhoff UPS Configuration](#) 对 UPS 进行配置，方可以启用 UPS 功能，设置方法同 CX20 系列控制器
- 关于具体某一款控制器/工控机是否配备有 UPS，可以在官网搜索控制器型号，在详细参数栏目中进行查询
- 配置 UPS 的时候，配置为“保存数据并安全关机”模式，不要使用“电能耗尽后自行关断”模式，[说明链接](#)
- 使用 **Persistent** 变量保存的优势：
 - 配置简单，工程师只需要将变量申明到 **Persistent** 关键字下，并通过功能块/软件配置开启 UPS 功能即可使用，保存和读取动作由系统完成
 - 工程师也可以使用 [FB_WritePersistentData](#) 来进行手动保存，需要注意的是，如果通过手动触发来保存，只是保存了当前时刻的数据
- 使用 **Persistent** 变量保存的劣势：

- 保存容量有限，如果数据容量非常大，可能会出现 UPS 中的电能不足以维持系统写入完成的情况，进而导致数据无法正常保存，建议不要使用 **Persistent** 方式保存超过 1MB 大小的数据
- 保存的文件是二进制格式文件，无法直观地看出保存变量的值，也没有办法进行修改。如果修改变量的数据结构（如修改了结构体中的参数）后，工程师操作不当可能会导致部分保存数据错误（正确的操作是重新编译解决方案并激活配置）
- 由于使用的是写入文件的方式来进行保存，如果文件被人为破坏，数据也会丢失
- **可以用 **Persistent** 来保存的数据：**
 - 设备生产过程中，实时变化，需要实时保存的数据
- **不适合用 **Persistent** 来保存的数据：**
 - 设备运行过程中，不常发生变化，数据量较大，较少被修改的数据，如工艺配方参数，电机的零点位置，初始化时使用的默认参数等
- **CX5x 系列控制器 1sUPS 配置参考样例：** [倍福虚拟学院](#)

三. 通过自定义文件读写的方式来保存数据

自定义文件读写的保存数据方式原理上和 **Persistent** 方式相同，都是通过将数据写入到文件，保存在不会掉电丢失的硬盘中，在上电时进行读取并执行数据的恢复，来实现数据的保存功能。

相比于 **Persistent** 方式，自定义文件读写的方式更加灵活，工程师可以依据自己的需求选择相应的格式，如 **recipe**, **txt**, **xml**, **csv**, **Json** 等来进行保存。

此外，对于保存的时间和读取动作的触发，工程师可以自定义设置，不局限于“上电读取”，“保存数据后关机”等操作。且由于使用了标准的 **txt**, **csv** 等格式，数据可以被轻松解析，也可以方便进行修改。存储目的地也不再局限于特定的文件目录，可以自定义任意的文件夹，也可以指向外部存储设备（通常不推荐把重要的配置文件保存到外部存储设备上）等等。

- **使用该方法时的注意事项：**
 - 需要工程师针对保存方式编写程序，并在 TwinCAT 中做相应的配置，包括但不限于：
 - 配置 **RecipeManager**，搭配 [RecipeManCommands](#) 功能块进行配方保存
 - 使用 [FB_FileWrite](#) 进行文本文件读写
 - 配置 **XMLserver** 进行 XML 文件保存
 - ...
 - 触发状态由用户自定义，包括但不限于：
 - **TON** 功能块实现脉冲定时触发

- 根据特定的事件（如某个工位完成工作）进行逻辑触发，或通过逻辑检测变量是否发生变更，一旦发生变更，执行写入动作
- 外部 IO 人为触发（通过按钮和输入模块采集外部信号），或者通过上位机写入触发位进行触发
- ...
- 持续写文件对磁盘寿命的影响：
 - 以 CX2000 系列标配的 40GB 的 CFAST 卡为例，其标称擦写容量为 1200TBW（即可以反复擦写 1200TB 大小的数据）
 - 假设某型号设备允许工程师写入 1000TB 大小的数据，每隔 1 秒钟写入 1MB 的数据，则实际可满足持续写入的时间 $1000 \times 1024 \times 1024 / (60 \times 60 \times 24 \times 365) \approx 33.2$ 年，对于有更大擦写需求的客户，可以选配更大容量的存储设备，此计算数据供参考
- 使用自定义文件保存来存储数据的优势：
 - 高度定制化，工程师可以根据自己的需求对数据进行保存
 - 不受 UPS 等硬件模块的影响，理论上该方法适用于任何嵌入式控制器/工控机
 - 保存出来的数据可以很方便地与其它设备或高级语言程序进行交互
- 使用自定义文件保存来存储数据的劣势：
 - 需要工程师编程实现，使用起来相较 UPS 以及 NOVRAM 的处理方法更为繁琐
 - 单个文件不宜保存过多的变量，如单个 recipe 建议不要超过 100 个变量，且读写文件瞬间，会造成 CPU 占用率的上升，以 CX5130 为例，将会造成 15%-30% 的 CPU 占用率上升，如果控制器常态负载在 50% 以上，写大文件操作可能会引起系统卡顿
 - 并不能保存掉电瞬间的数据，且如果掉电瞬间正在执行写文件操作，文件可能会损坏，可以通过逻辑对文件进行备份来保护数据
 - 由于使用的是写入文件的方式来进行保存，如果文件被人为破坏，数据也会丢失
- 可以用自定义文件来保存的数据：
 - 在设备生产过程中，不常发生变化，较少被修改的数据，如工艺配方参数，电机零点偏移等等
- 不适合用自定义文件来保存的数据：
 - 设备运行过程中，实时变化并且需要实时保存的参数
- 参考配置样例：[TwinCAT XML Server](#)，[CSV 文件存储](#)，[JSON](#)，[配方工艺管理](#)

四. 通过数据库来保存参数

如果数据量非常大，用户可以采用数据库的方式来保存和管理数据，将数据保存在数据库服务器中，可以方便地使用数据库的固有特性来对大批量的数据进行查询，管理和设置，包括但不限于：

1. 对于某个参数的修改，可以保存历史记录或事件
2. 利用数据库的高级语言接口，可以方便地备份数据
3. 某些数据库还可以支持数据结构的更改和扩展
4. ...

- **使用该方法时的注意事项：**

- 工程师需要单独配置一个数据库服务器，可以安装在控制器本地，也可以在内网（局域网）或云服务器中独立部署，可以参考 [Database Server 拓扑结构与工作模式](#) 中的说明
- 对于数据库的类型，可选的有 MySQL，MongoDB，InfluxDB 等，用户需要根据自己的保存需求以及数据库的特性来选择
 - MySQL 是最流行的关系型数据库管理系统，在 WEB 应用方面 MySQL 是最好的 RDBMS(Relational Database Management System: 关系数据库管理系统)应用软件之一
 - MongoDB 是一个基于分布式文件存储的数据库。旨在为 WEB 应用提供可扩展的高性能数据存储解决方案。MongoDB 是一个介于关系数据库和非关系数据库之间的产品，在非关系数据库中，其功能最丰富、最像关系数据库
 - InfluxDB 是一个由 InfluxData 开发的开源时序型数据库。它由 Go 写成，着力于高性能地查询与存储时序型数据。InfluxDB 被广泛应用于存储系统的监控数据，IoT 行业的实时数据等场景
 - ...
- 需要在 TwinCAT 中 [配置 Database Server](#) 并编写程序

- **数据库保存参数的优势：**

- 数据量非常大时，触发写文件动作瞬间会造成整个系统负载增加，所需的写入时间也会较长，甚至有可能导致写入失败，而使用数据库来保存变量，可以针对单个变量进行 update，而不采用全部数据写入的方式
- 数据库对于变量的修改，删除等都有记录和追责的功能，非常适合需要复杂，严格数据管理的用户
- 数据库普遍都具有高级语言接口，可以方便地与高级语言开发的上位程序进行交互

- **数据库保存参数的劣势：**

- 需要工程师自行搭建数据库服务器并编程实现，使用起来相较前面的几种方法更为繁琐
- 数据保存在数据库服务器中，如果服务器受到攻击，或者发生文件层面的损坏，数据依旧会丢失
- 并不能保存掉电瞬间的数据
- 适合用数据库保存的场合：
 - 数据容量较大，数据频繁变更

五. 数据保存方案的选择

建议客户对于数据进行分类，根据数据的重要程度以及不同的保存需求，来选择不同的保存方式。且单一保存策略均有失效风险，可以考虑多重保存策略。

- **数据经常变化，需要实时保存的数据，重要程度高，不可丢失，建议采用两种或更多的保存策略**
 - 数据量较小部分可以选择 NOVRAM 或 Persistent 搭配硬件使用
 - 调用一种写文件策略来进行保存，或搭配数据库使用
 - 如果采用写文件的方式，可以增加备份文件防止丢失写到一半的时候断电导致文件损坏，比如采用两个文件交替写入的方式，两个文件不会同时损坏
- **数据不经常变化，不需要严格实时保存的数据，重要程度一般**
 - 采用文件读写的方式保存，每次上电第一个周期读取文件然后对变量赋值，条件允许的情况下写入周期可以适当拉长，可以设置一个备份文件来进行保护
 - 也可以采用数据库来进行保存
- **设备初始化参数，驱动器原点信息，各个工位的运行速度，工艺配方等基本不会变动，但是数据量较大，且需要进行保存，保持的数据**
 - 可以使用文件读写的方式来保存
 - 将参数做成配置文件后，每次上电第一个周期读取文件然后对变量赋值即可
 - **不要用 Persistent 或者 NOVRAM 来保持此类数据**
 - 设备部署完成后，建议对相应的配置文件建议做一个异地备份（如拷贝到工程师电脑上）



请用微信扫描二维码
通过公众号与技术支持交流

上海（中国区总部）

中国上海市静安区汶水路 299 弄 9 号 (市北智汇园)

电话: 021-66312666

北京分公司

北京市西城区新街口北大街 3 号新街高和大厦 407 室

电话: 010-82200036

邮箱: beijing@beckhoff.com.cn

广州分公司

广州市天河区珠江新城珠江东路 16 号高德置地 G2603 室

电话: 020-38010300/1/2

邮箱: guangzhou@beckhoff.com.cn

成都分公司

成都市锦江区东御街 18 号 百扬大厦 2305 房

电话: 028-86202581

邮箱: chengdu@beckhoff.com.cn

倍福官方网站:

<https://www.beckhoff.com.cn>

在线帮助系统:

https://infosys.beckhoff.com/index_en.htm

倍福虚拟学院:

<https://tr.beckhoff.com.cn/>

招贤纳士: job@beckhoff.com.cn

技术支持: support@beckhoff.com.cn

产品维修: service@beckhoff.com.cn

方案咨询: sales@beckhoff.com.cn