|  |  |  |  |
| --- | --- | --- | --- |
| 作者： | 杨靖华 | | [Logo_Beckhoff_Red](http://www.beckhoff.com.cn/)  中国上海市静安区汶水路 299 弄 9-10 号  市北智汇园4号楼（200072）  TEL: 021-66312666  FAX: 021-66315696 |
| 职务： | 西南区 | 技术工程师 |
| 日期： | 2019-11-20 | |
| 邮箱： | j.yang@beckhoff.com.cn | |
| 电话： | 0871-63550636 | |

|  |
| --- |
| **Tc3\_DataBase 与 Microsoft SQL Server2012通讯测试** |
| **摘 要**：本文介绍了如何用TwinCAT 3 调用Tc3\_DataBase库实现与Microsoft SQLServer的控制通讯测试。 |
| **关键字：**Win 7,Win10，TF6420-Database-Server, Tc3\_DataBase,Microsoft SQL, Microsoft.NET Farmwork 4.0 |
| **附 件：**   |  |  |  | | --- | --- | --- | | 序 号 | 文件名 | 备注 | | 1 | CurrentConfigDataBase | DataBaseServer配置文件 | | 2 | Data\_Base.tnzip | TC3源程序 | |  |  |  | |  |  |  | |
| **历史版本：**   |  |  |  | | --- | --- | --- | | 2018-09-20 | 张洪彪 | TwinCAT 3 DataBase Server与 Microsoft SQL通讯测试.docx | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| **免责声明：**  我们已对本文档描述的内容做测试。但是差错在所难免，无法保证绝对正确并完全满足您的使用需求。本文档的内容可能随时更新，也欢迎您提出改进建议。 |
| **参考信息：** |

**目 录**

[1. 测试环境介绍 3](#_Toc26643316)

[2. 软件安装 3](#_Toc26643317)

[3. 数据库表单建立及简单操作介绍 3](#_Toc26643318)

[4. 控制器操作介绍 6](#_Toc26643319)

[5. 调试及总结 11](#_Toc26643320)

**文档正文要求**

|  |
| --- |
| 内容组成：   * 目录：模板中目录为“域”，如果内容超过10页，右键单击刷新域即可，否则可以删除 * 测试条件：软件版本，如果有必要，记录下硬件的型号和版本 * 准备工作：只要做一次但必须保证正确无误的步骤，比如接线、IP设置、加路由等 * 操作步骤：正常的操作截图 * 常见问题：在不同的测试和应用条件下，由不同的工程师积累的故障处理经验   截图：   * 只截必要的部分。避免全屏截图，不美观，并且不能突出重点。 * 尽量在截图中圈红标注关键信息 * 尽量使全文截图显示为同样的比例   正文字体和编号：   * 直接在模板上编辑文字，即可延用字体和编号设置 * 如果是复制粘贴来的文字，可使用格式刷，使之与全文风格一致 |

# 测试环境介绍

硬件：win10 企业版电脑2台，其中1台安装TwinCAT3，另外1台安装Microsoft SQL 2012。

软件：Microsoft SQL 2012软件；Microsoft.NET Farmwork 4.0；TF6420- Database -Server.exe；TwinCAT3 4022.30

# 软件安装

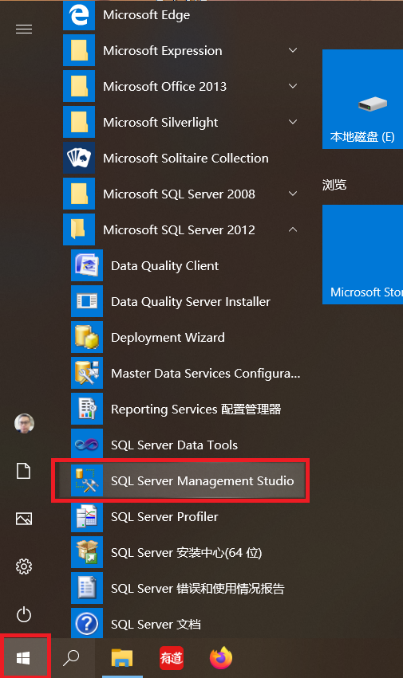
1. 笔记本电脑B（Win10 X64企业版）上安装Microsoft SQL2012 软件，安装方法参见百度“SQL Server 2012 下载和安装详细教程”。此处不再详述。

（安装时设置登录名：sa，密码：123，用户根据实际情况自己更改设置）。

1. 控制器(测试用笔记本电脑A（Win10 X64企业版）做控制器)上安装Microsoft.NET Farmwork 2.0以上版本，本次测试使用的是Microsoft.NET Farmwork 4.0（Win10自带）。
2. 控制器上安装TwinCAT 3软件，版本越新越好，测试用的是4022.30版本；
3. 控制器上安装TF6420-Database-Server服务选件包，该服务选件包可从官网下载最新的版本安装，V3.1以上版本才可以，低于V3.1可能测试不通过。
4. 计算机上安装好相关软件后测试正常才能进行下一步。

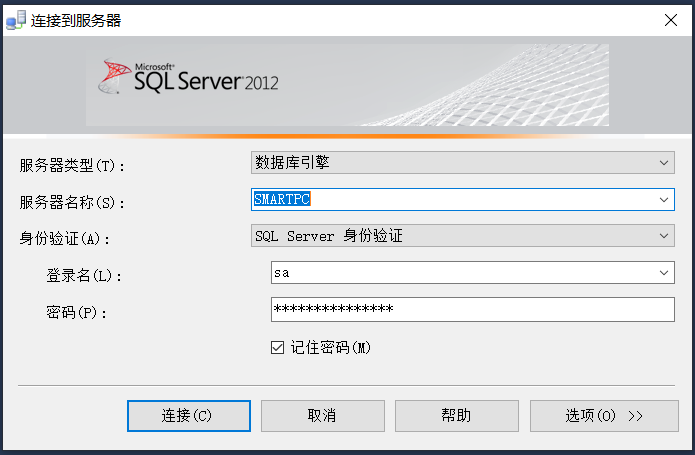
# 数据库表单建立及简单操作介绍

1. 打开SQL软件，新建数据库，新建Table，新建变量等信息。打开SQL软件方法如下图所示：

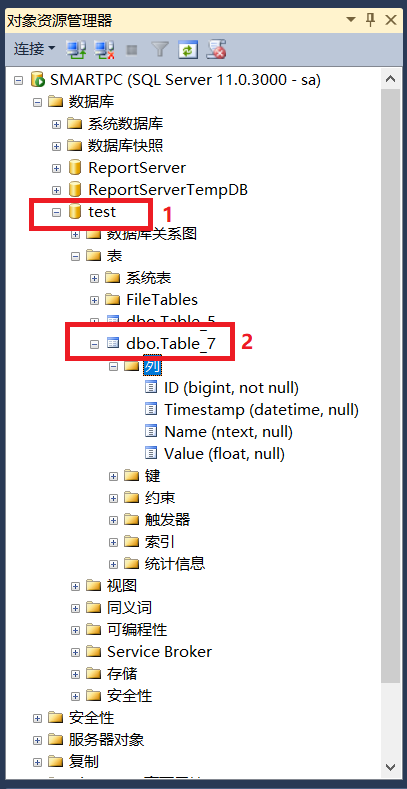


点击“开始”菜单🡪Microsoft SQL Server 2012🡪SQL Server Management Studio打开数据库软件。

1. 打开时会提示你与数据库服务建立连接，该建立连接的方法与“Twincat3database设置.pdf”中连接数据库方法类似，注意用户名为安装时默认的用户名sa，密码为安装时设置的数据库密码，本例为123。Server name为默认，也可用IP地址代替，如下图所示：



1. 点击“连接”后进入数据库管理界面，有个“对象资源管理器”如下图所示，需要在这里新建我们的数据库及表单。

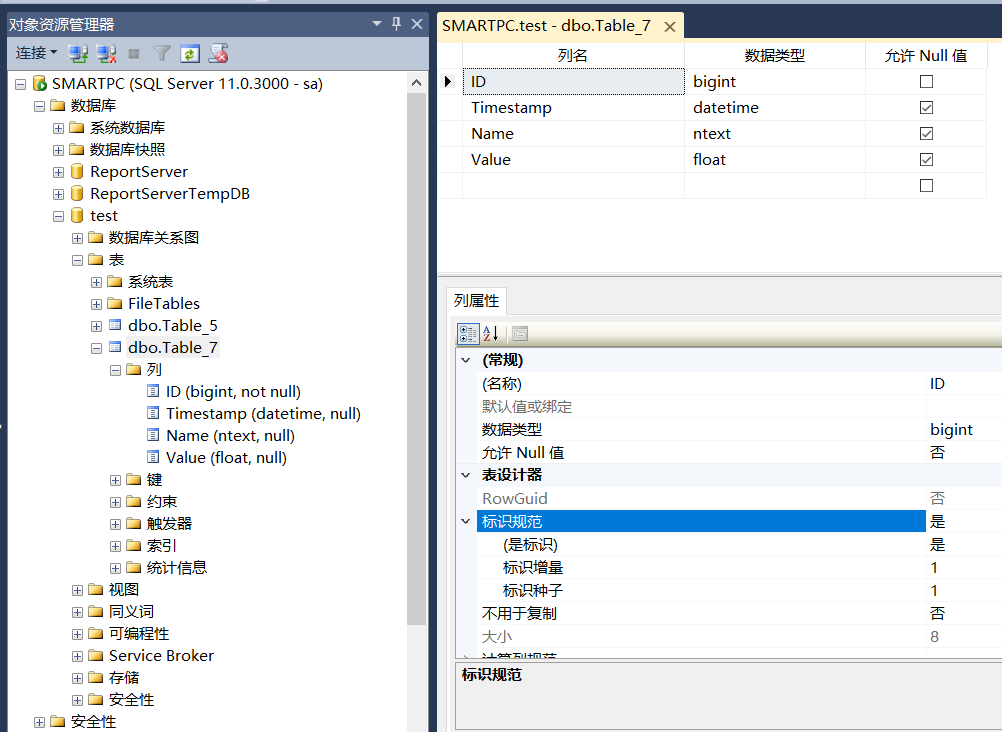


--test（1）：数据库的名称；

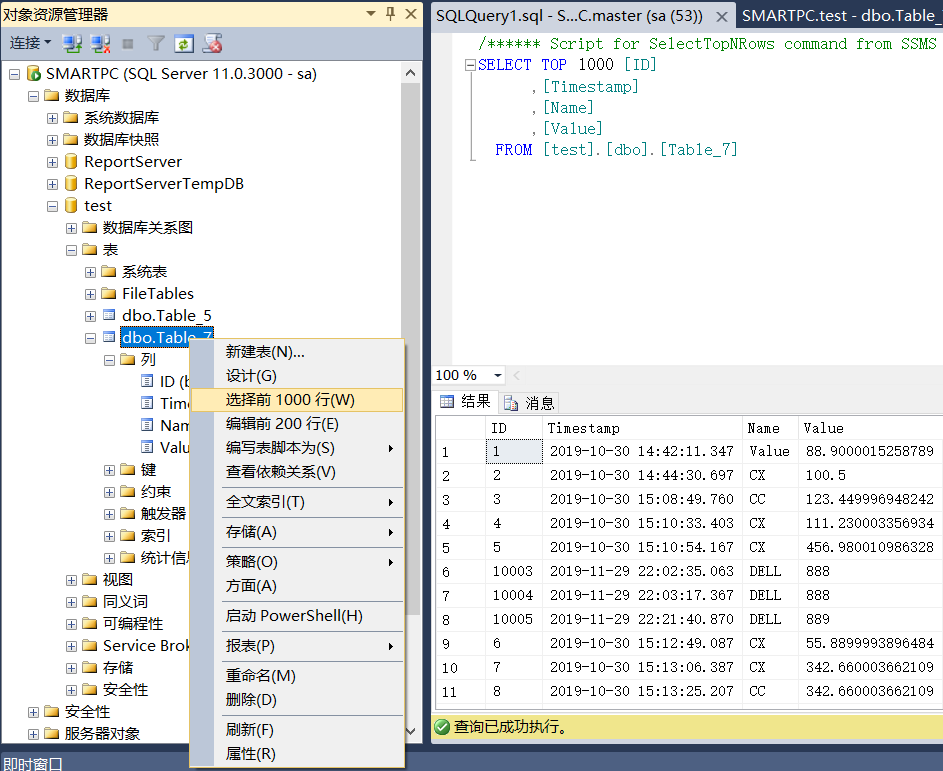
--dbo.Table\_7（2）：表单名称（其实表单名称是Table\_7）

1. 新建与TwinCAT3通信的表单。TC3的数据库通信采用自己的标准表及自定义表格式。这里介绍的是标准表的方式。如下: Table\_7第一第二个参数必须是ID和Timestamp，余下的才是参数，与PLC中参数对应一致，尤其注意变量类型须一致。

上图中，右键点击dbo.Table\_7，选择“设计”弹出如下的右侧窗口。按照下图新建表单的项，需要注意的是除了ID外其他3项需要勾选上“允许Null值”。需要注意的是ID项需要在属性栏的“标识规范”项中“（是标识）”选择“是”，标识增量选择1，否则TwinCAT中会报错。表建好后点击全部保存。

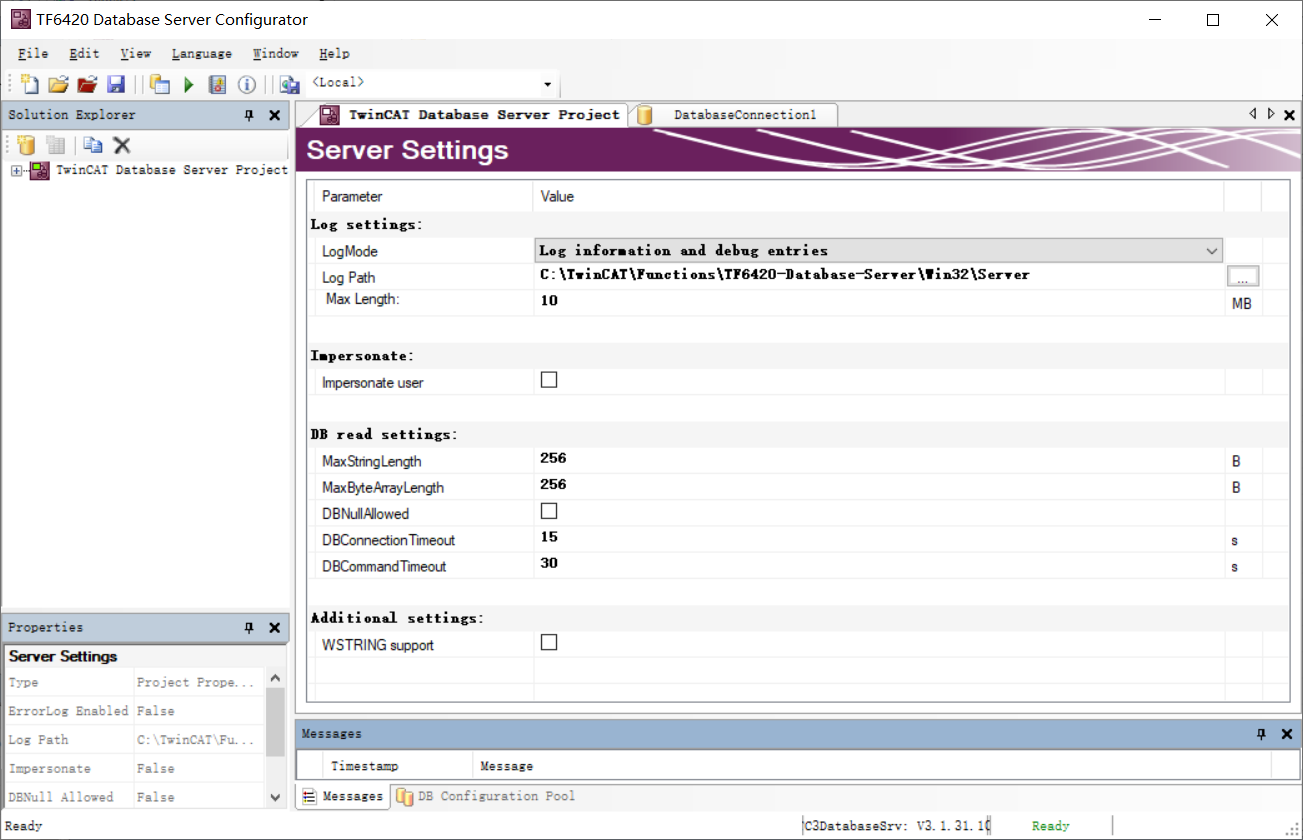


1. 查看数据库记录的数值。如下图所示Table\_7点击右键，选择“选择前1000行”即可查看记录的前一千行，在图的右侧把“SELECT TOP 1000”改成“SELECT TOP 10000”即可查看10000行，最后按F5键执行查询。

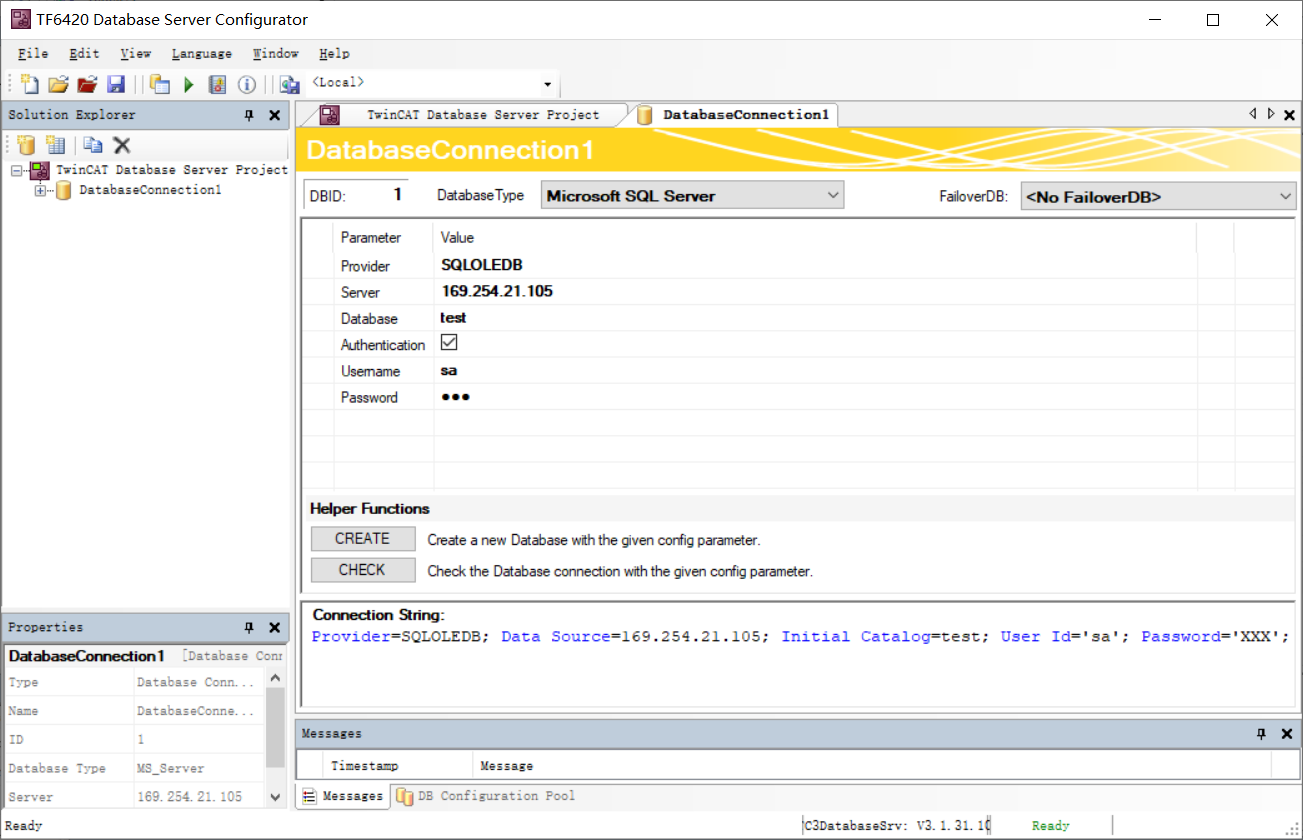


# 控制器操作介绍

1. 在控制器中安装TF6420选件包后打开TF6420 Configurator,新建个配置，如下图所示，Log settings最好选择Log information and debug entries并且设置消息日志大小，这样方便我们在调试PLC程序时查看错误报警的具体信息。



1. 新建数据库的连接，如下图所示



--DBID：这是数据库连接的ID号，在PLC中需要用到这个ID号，这里ID号是1；

--DataBase Type：数据库的类型，这里选择Microsoft SQL Server；

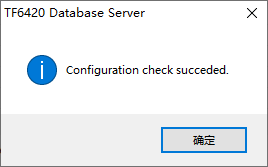
--Provider：这里选择默认的SQLOLEDB；

--Server：数据库服务器的IP地址，这里使用的是169.254.21.105；

--Database：数据库的名称，这里是test，和第7步的1一致；

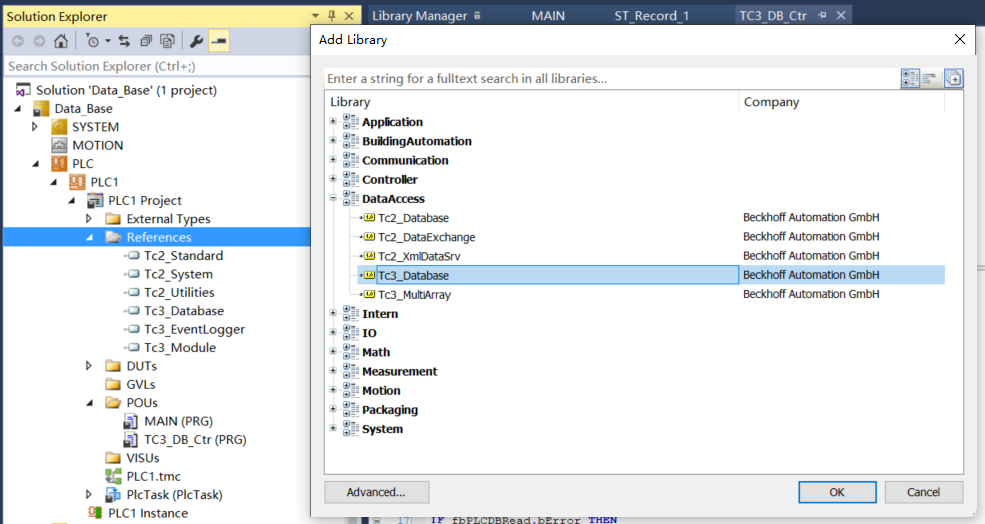
--Authentication 勾选上并且在Username填入数据库服务器的登录名sa；Password填入密码123。

设置完成后需要点击“CHECK”来测试连接是否成功，成功则弹出如下窗口。

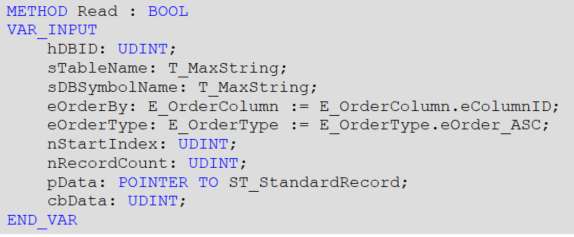


设置到这里基本就完成了，下面的ADSDevice、Symbols、DBTable这些不需要去设置了。设置完成需要点击Activate Configuration按钮来激活当前配置使之生效。

1. 编写PLC程序，如下图所示，首先添加TC3的数据库访问库，在DataAccess下面的 Tc3\_Database库，然后还需要添加Tc3\_EventLogger，Tc3\_Module库等。



1. 在PLC中调用FB\_PLCDBRead来读取数据库的数据记录，需要注意的是这个功能块有两种调用方法：.Read方式是读取的倍福特定结构的表的方式，也就是我们上面建的那种表的格式；.ReadStruct是读取用户自定义表的方式。.Read方法的输入接口如下图所示。



--hDBID：填第11步配置的DBID号1；

-- sTableName：填第11步Database配置的表的名称test；

-- sDBSymbolName：这个是字符串变量，对应需要写入数据库的数据的名称，这个名称会在数据库表中的Name中显示；每个写入的变量都有一个变量名，这个对应的就是PLC 变量的变量名。

-- eOrderBy：查询数据的排序方式，一般选择ID号排序，也就是E\_OrderColumn.eColumnID。

-- nStartIndex：查询的起始数，这个的意义就是如果CC这个变量记录了1000条记录，如果选择ID方式排序，那么这里输入5就是从第5条开始查询；

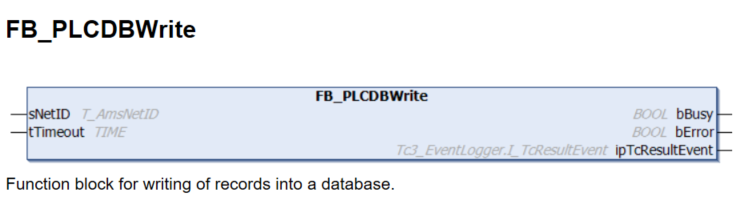
--nRecordCount：查询的条数，配合nStartIndex使用。

--pData：查询回来的数的存放地方，这里一般是用的一个ST\_StandardRecord类型的数组来存放查询到的数，那么这里的输入就是ADR(ST\_StandardRecord类型数组变量名)；

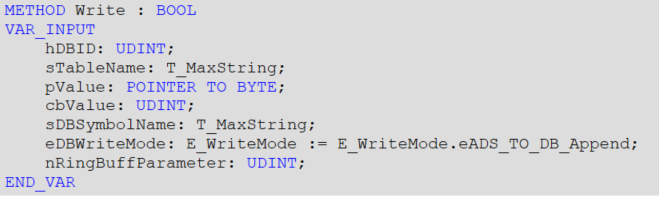
--cbData：存放查询数的数组长度，一般用sizeof(ST\_StandardRecord类型数组变量名)来表示。



1. 在PLC中调用FB\_PLCDBWrite来向数据库中写入PLC变量。数据库写入块有3种调用方法：.write按照倍福标准方式写入；. WriteBySymbol通过符号名方式按照倍福标准方式写入；. WriteStruct按照用户自定义结构体方式写入。



.write的输入参数如下图所示：



上图中hDBID、sTableName和13步介绍的一样。

--pValue：需要写入数据库的PLC变量的地址，一般用ADR(变量名)方式表示。

--cbValue：需要写入数据库的PLC变量的长度，一般用sizeof(变量名)方式表示。

-- sDBSymbolName:需要写入数据库的变量在数据库的名称，这个名称会写入表格的Name栏中；

-- eDBWriteMode：数据写入方式，有3种方式：append方式是追加（数据会越写越长）、update方式刷新数据（只有一条记录）、ring buffer方式循环缓存（数据不会无限增大，适合于数据库在控制器中这种方式）。

-- nRingBuffParameter：eDBWriteMode采用ring buffer方式时的参数，就是可以存多少条记录的一个设置。

1. 程序，这个程序需要在main主程序中进行调用。程序采用了2个布尔型变量，b\_read用来读取数据记录，b\_write用来写入数据记录；ipTcResultEvent、ipTcResultEvent1是Tc3\_EventLogger.I\_TcResultEvent类型的变量，用于指示读写数据块的错误代码，不能显示具体的错误描述，具体错误描述可以通过TF6420 Configurator来查看，程序如下：

变量定义：

PROGRAM TC3\_DB\_Ctr

VAR

fbPLCDBRead : FB\_PLCDBRead(sNetID := '', tTimeout := T#5S);

ReadStruct : ARRAY[1..5] OF ST\_StandardRecord;

nState:UDINT;

b\_read: BOOL;

fbPLCDBWrite : FB\_PLCDBWrite(sNetID := '', tTimeout := T#5S);

myValue : REAL := 43.23;

ipTcResultEvent: Tc3\_EventLogger.I\_TcResultEvent;

ipTcResultEvent1: Tc3\_EventLogger.I\_TcResultEvent;

b\_write: BOOL;

nState1: USINT;

myCustomStruct : ST\_Record\_1;

ColumnNames:ARRAY[0..4] OF STRING;

str1: T\_MaxString := 'Value';

str2: T\_MaxString := 'CX';

END\_VAR

程序编写：

//读取数据库数据

IF b\_read THEN

fbPLCDBRead.Read(

hDBID:= 1,

sTableName:= 'Table\_7',

sDBSymbolName:= str2,

eOrderBy:= E\_OrderColumn.ID,

eOrderType:= E\_OrderType.DESC,

nStartIndex:= 0,

nRecordCount:= 4,

pData:= ADR(ReadStruct),

cbData:= SIZEOF(ReadStruct));

END\_IF

IF fbPLCDBRead.bError THEN

ipTcResultEvent := fbPLCDBRead.ipTcResultEvent;

nState1 := 255;

ELSE

nState1 := 0;

END\_IF

IF NOT fbPLCDBRead.bBusy THEN

b\_read:=0;

END\_IF

//PLC变量写入数据库中

IF b\_write THEN

fbPLCDBWrite.Write(

hDBID:= 1,

sTableName:= 'Table\_7',

pValue:= ADR(myValue),

cbValue:= SIZEOF(myValue),

sDBSymbolName:= str1,

eDBWriteMode:=E\_WriteMode.eADS\_TO\_DB\_Append,

nRingBuffParameter:= 0) ;

END\_IF

IF fbPLCDBWrite.bError THEN

ipTcResultEvent1:= fbPLCDBWrite.ipTcResultEvent;

nState := 255;

ELSE

nState := 0;

END\_IF

IF NOT fbPLCDBWrite.bBusy THEN

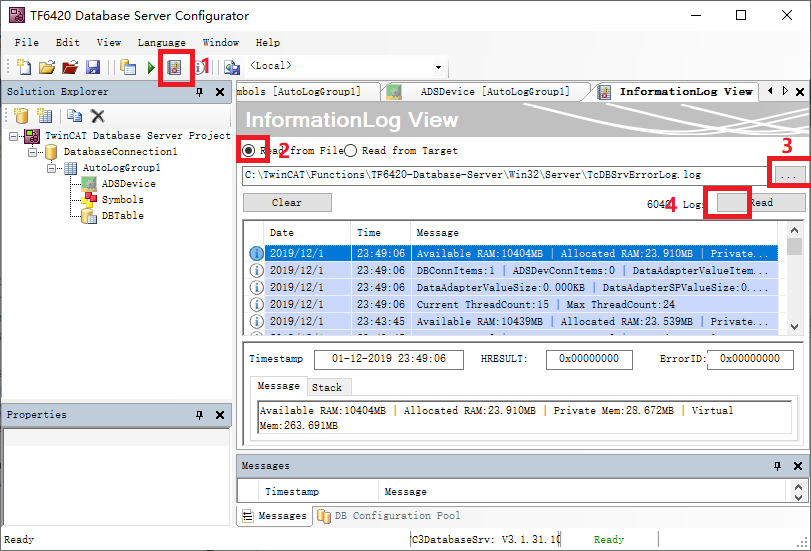
b\_write:=0;

END\_IF

# 调试及总结

1. 程序调试

程序在运行后不免有错误，我们需要更加错误代码在TF6420 Configurator中查看具体的错误消息,如下图所示：



点击（1）处的图标打开消息日志，选择（2）处的打开方式，在（3）处选择第10步的Log path设置的日志文件地址，在（4）鼠标点击按钮读取日志。这时相关的日志就在下方的表中显示出来，我们可以选择相应的行在Message框中就会显示详细的错误。

1. 测试总结

采用TC3\_Database比TC2\_Database的优势在于TC3\_Database只要2个块就能完成数据库的读写，操作比较简单，而且数据库读写方式有多种选择可选，是作为不懂数据库的人连接数据的比较好的选择。

# 常见问答

1. 数据库中修改表提示另存的话，有什么解决办法么？

参考下图：

