**TwinCAT HMI EventGird和Line控件的使用方法**

|  |  |
| --- | --- |
|  | 作者：余洋  职务：技术工程师  公司：BECKHOFF中国  邮箱：yang.yu@beckhoff.com.cn  日期：2023-08-29 |
| **摘 要：**  TwinCAT HMI中有很多控件，其中 EventGird和EventLine可以为用户提供TwinCAT项目中自定义的报警、报错、消息等等提示。除此之外，还可以自定义筛选规则，筛选或确认自定义提示，自由度非常高。此文档讲解了如何安装HMI EventLogger拓展包、自定义TwinCAT事件、编写PLC相关程序以及调试EventGird控件。 | |
| **附 件：**   |  |  |  | | --- | --- | --- | | 序 号 | 文件名 | 备注 | | 1 | TwinCAT HMI EventGird Test.tnzip | 例程 | |  |  |  | |  |  |  | |  |  |  | | |
| **历史版本：**   |  |  |  | | --- | --- | --- | |  |  |  | |  |  |  | |  |  |  | |  |  |  | | |
| **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，如有改动，恕不事先通知，也欢迎您提出改进建议。  **参考信息：** | |
| 目 录  [1. 软硬件版本 3](#_Toc144200232)  [1.1. 控制硬件 3](#_Toc144200233)  [1.2. 控制软件 3](#_Toc144200234)  [2. 准备工作 3](#_Toc144200235)  [3. TwinCAT项目配置 4](#_Toc144200236)  [3.1. 自定义TwinCAT事件 4](#_Toc144200237)  [3.2. 编写PLC程序 6](#_Toc144200238)  [4. 配置Event Gird控件 11](#_Toc144200239)  [4.1. Fliter 11](#_Toc144200240)  [4.2. Columns 12](#_Toc144200241)  [5. 使用Event Gird控件 13](#_Toc144200242)  [6. 使用Event Line控件 13](#_Toc144200242)  [7. 常见问题 16](#_Toc144200243) | |

# 软硬件版本

## 控制硬件

测试电脑（Windows11专业版）

## 控制软件

TwinCAT3（文档测试版本为v 3.1.4024.47）

TwinCAT HMI开发组件TE2000

Beckhoff.TwinCAT.HMI.EventLogger Nuget安装包

# 准备工作

在使用EventGird控件之前需要安装相关安装包，TwinCAT 3和TE2000开发环境是必须的。在安装完TE2000之后，自带的Beckhoff.TwinCAT.HMI.Controls组件中会有Event Gird控件：

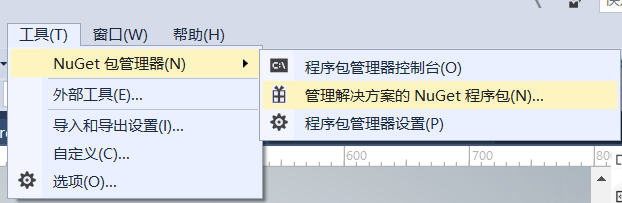
图形用户界面, 文本, 应用程序

描述已自动生成

但是想要使用它，必须安装Beckhoff.TwinCAT.HMI.EventLogger这个额外的拓展组件，安装此组件最简单的方式是右键Packages或者在TwinCAT最上方的工具选择Nuget包管理器—管理解决方案的Nuget包。

图形用户界面, 文本, 应用程序

描述已自动生成



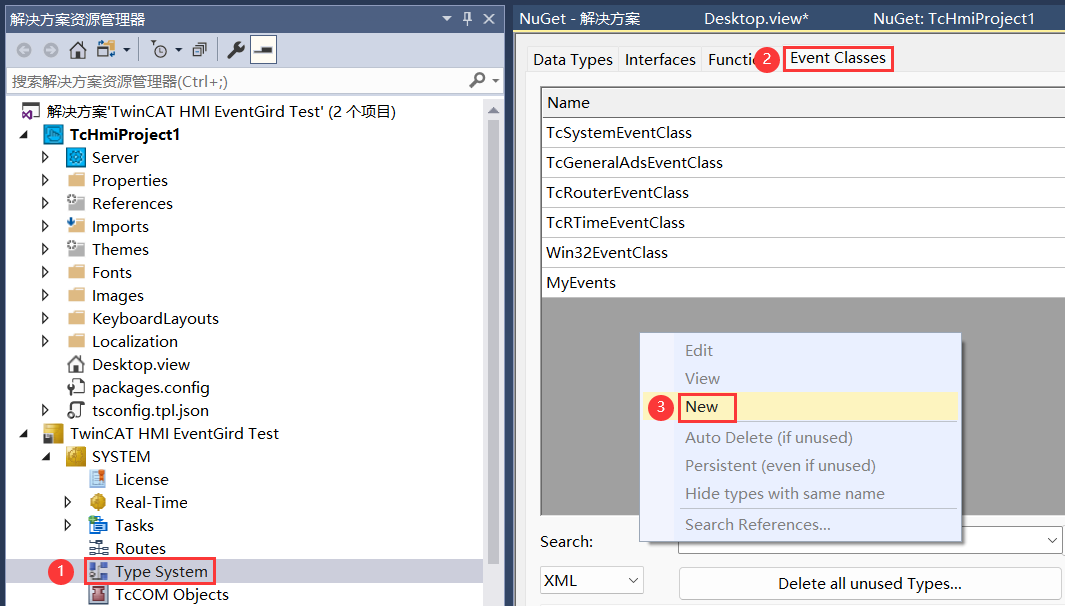
进入Nuget包管理器之后选择合适的程序包源，有网络状态可以选nuegt.org进行全网搜索已发布的Nuegt包。如果编程电脑无法联网，可以在Github上下载独立的EventLogger包，放至自定义位置之后再自定义一个包源名称就可以选择安装了。



# TwinCAT项目配置

## 自定义TwinCAT事件

TwinCAT HMI EventGird的消息提示源于TcEventlogger相关接口，自定义事件必须在TwinCAT以TMC文件的形式编辑。首先来到TwinCAT System选择Type System，所有TwinCAT的系统变量，接口，事件和函数都在此处。接着选中Event Classes，右键灰色处选择New进入TMC编辑器新建一个自定义的事件：

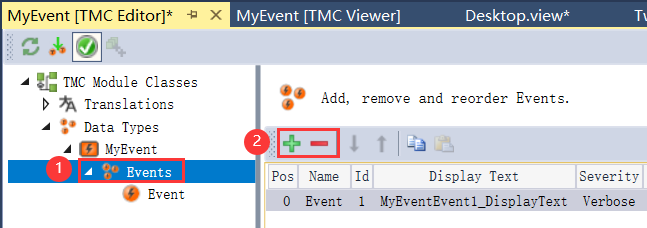


下一步是在TMC编辑器里面自定义事件内容，首先修改事件名，这个可以根据自己的想法随意修改：

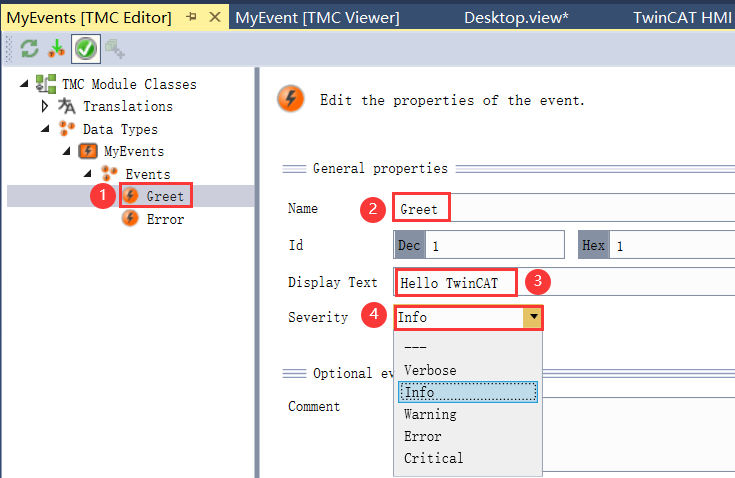
图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

在此事件中，可以新增各种子事件，默认情况会自带一个，点击Events可以新增或者删除事件：



在子事件中，可以修改事件名，修改事件等级和显示的事件内容：



完成所有的自定义设置之后，便可关闭TMC编辑器，选择TwinCAT中的Save All即可。这样一来，自定义TwinCAT事件的步骤就完成了

## 编写PLC程序

编辑完自定义TwinCAT事件之后，还需要编写相关的PLC程序才能对事件进行监控和其他操作。第一步是添加对应的功能块库文件：Tc3\_EventLogger。接下来声明相对应的功能块并且调用其中的方法。PLC程序的声明和调用如下所示：

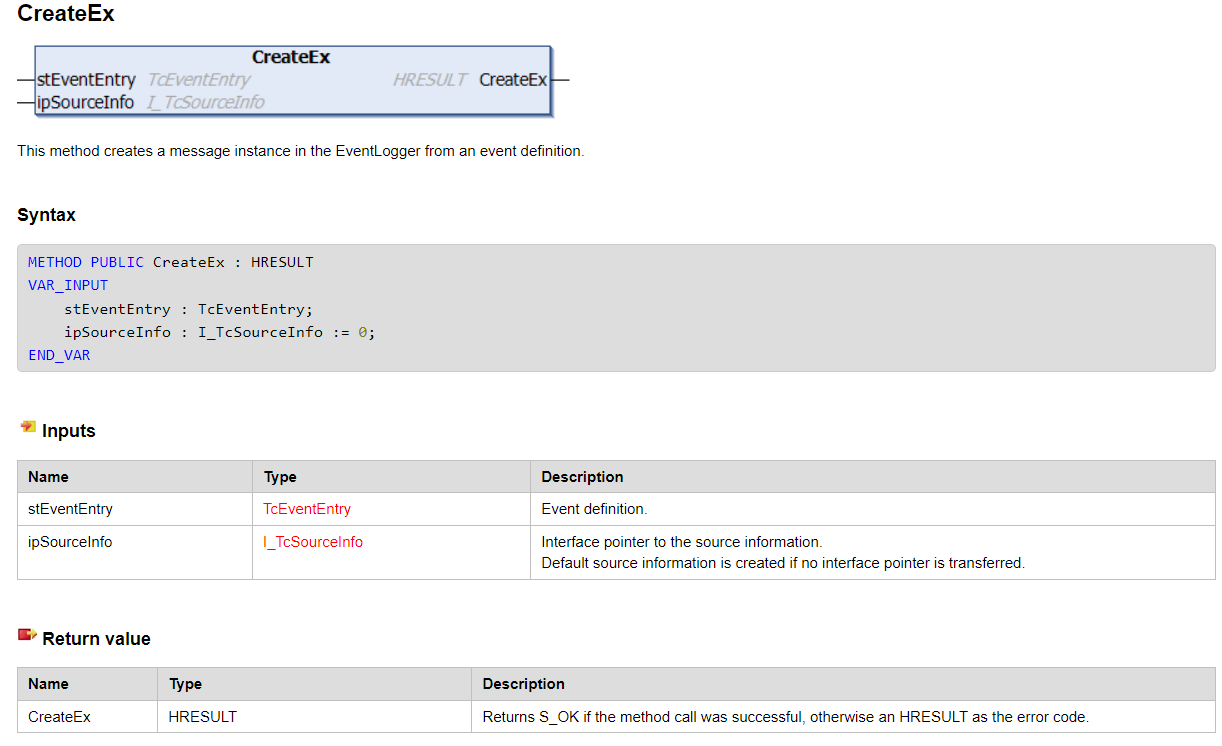
图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

首先，必须声明功能块FB\_TcEventLogger这个功能块，可以理解为这个功能块就代表了TwinCAT3 EventLogger本体。其他的功能块都是基于它的。其次，例程中还声明了其他两个消息以及警报功能块。以及声明了各种触发的引脚。

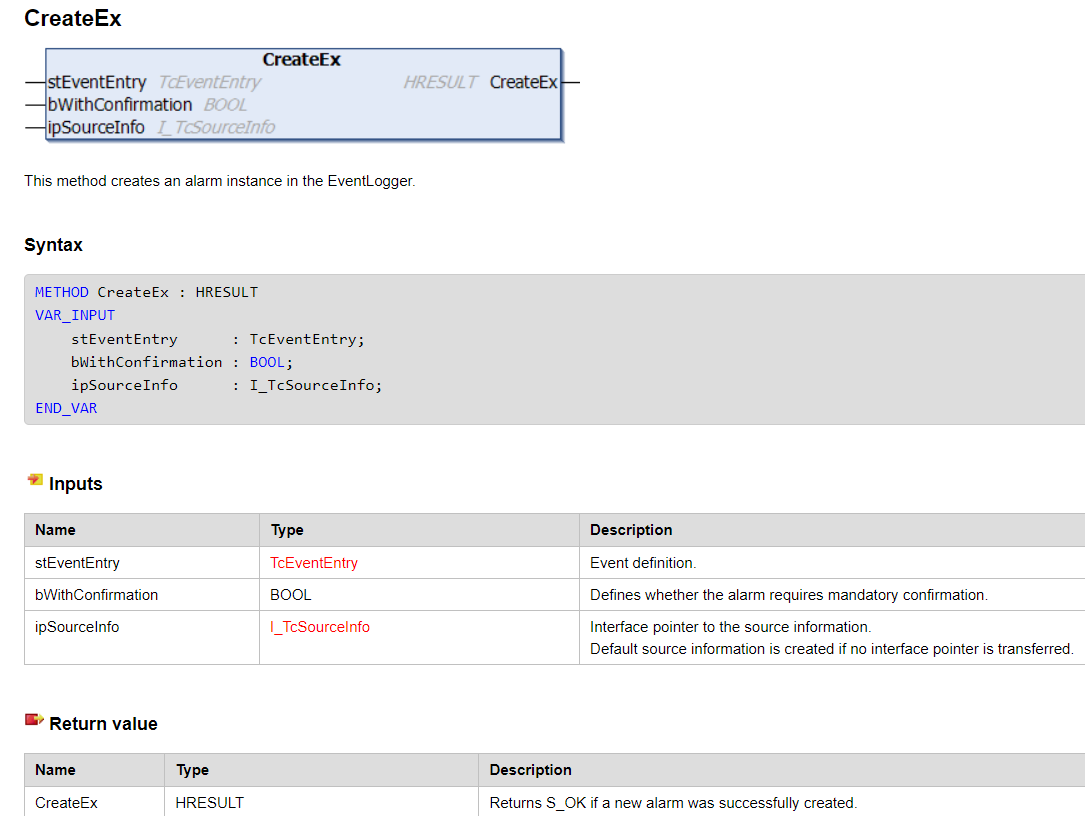
在程序中，第一步是进行初始化，当bIsInitalized被置为TRUE时，分别调用fbMessage和fbAlarm中的方法CreateEX进行创建。

以下是fbMessage.CreateEx方法的详细说明：



这个方法一共有两个输入引脚：stEventEntry和ipSourceInfo。前者是用户自己编辑的TwinCAT事件，也就是上一章节讲述的TMC编辑器中的配置步骤，这里填的是自定义事件的名字，而ipSouceInfo，本质上是一个接口，这里默认情况就填0即可。

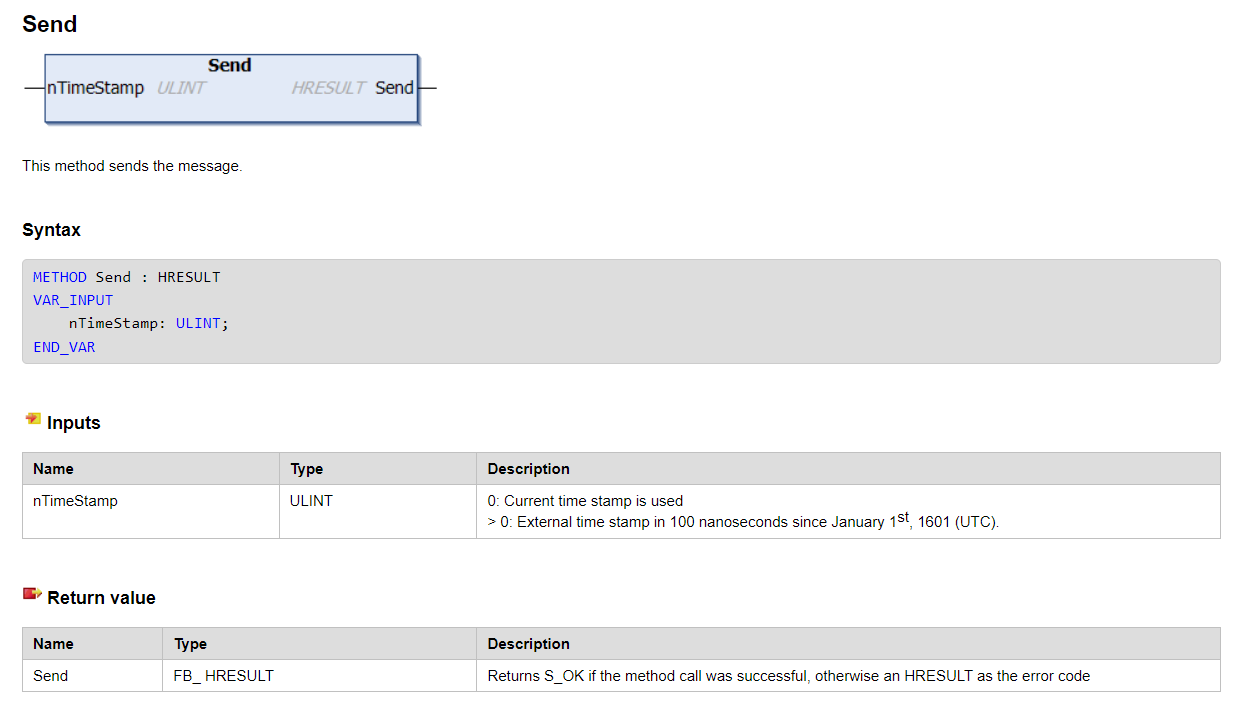
以下是fbAlarm.CreateEx方法的详细说明：



这个方法和之前的fbMessage.CreateEx非常类似，仅仅是多出了一个BOOL输入引脚，其含定义为“是否需要强制确认警报”，指的是警报功能是否需要用户在解除警报之前进行确认。如果警报需要强制确认，用户必须采取行动，如输入代码或按下按钮，对应的是Comfirm方法。这里开放用户可以手动确认来消除警报，所以选择TRUE。

当bMessage被置为TRUE时，TwinCAT发送创建的消息，用的方法为fbMessage.Send。功能块fbMessage本身是没有这个方法的，但是它继承了FB\_TcEventBase并且实现了I\_TcMessage接口。在I\_TcMessage中定义了Send方法：





这个方法仅有一个输入引脚，当该值为0时，选用的是当前的时间戳；当改制不为0时，选用外部时间戳。在本例程中，都使用本地时间戳，所以填0即可。

同理，当bAlarm被置为TRUE时，调用fbAlarm.Raise方法，其用法于Send一模一样，这里就不赘述了。

最后，当bClearAlarm被置为TRUE时，调用fbAlarm.Clear方法：

图形用户界面, 文本, 应用程序

中度可信度描述已自动生成

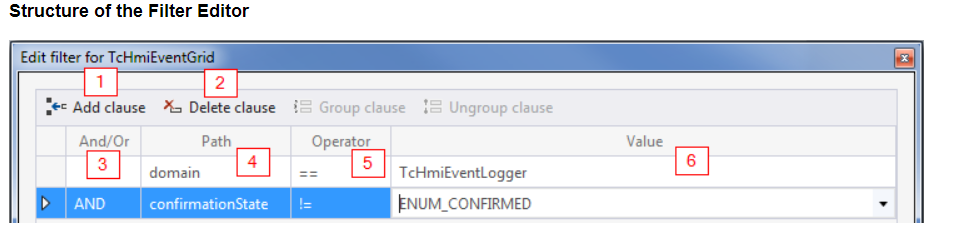
此方法中有两个输入引脚：前者是之前的时间戳，后者是一个BOOL变量。如果为TRUE，那Conform状态被重置，如果为FALSE，那么confirm状态不变。例程中清除报警>确认报警，所以此引脚被置为TRUE。

# 配置Event Gird控件

在TwinCAT HMI中新建一个Event Gird控件，简单的属性与其他普通控件基本一致，用户可以做参考和类比，这里也不赘述了。有两个比较重要的属性是Event Gird特有的：

## Fliter

顾名思义，这个属性就是改列表的过滤设置，在此可以设置错误的触发条件：



1. add clause：添加语句
2. Delete clause：删除语句
3. And/Or：语句的逻辑组合方式
4. Path：过滤的方式（如事件来源、名称、严重等级、ID、触发事件、复位事件、事件类型、事件ID、事件内容、报警状态、确认的状态等等）
5. Operator：语句中的运算符
6. Value：事件输出的内容

## Columns

该属性可以设置在Event Gird中显示事件的哪些信息：

图形用户界面, 应用程序

描述已自动生成

1. Type：这里可以修改事件的类型，有两种选项：Column和Custom Column。前者有默认的内容：比如事件类型，触发时间，是否被确认等等。后者可以显示自定义的内容。
2. Column Name：这里可以修改Column Name，如果是系统默认的选项（Type选择Column），内容名称可以选择上述提到的事件类型，触发时间，是否被确认。而自定义选项（Type选择Column）可以用关键字 “params::”（写的时候无引号）+ 参数名来引用参数，如：



1. Width：该设置可以更改显示内容在每一行中占有的宽度。

# 使用Event Gird控件

图形用户界面, 应用程序, Word

描述已自动生成

如图所示，控件中有12个小按钮或者显示框，接下来依次讲解一下这些子控件：

（1）开启/关闭显示警报（警告、错误、关键性错误等等）

图形用户界面, 文本, 应用程序, 聊天或短信

描述已自动生成

（2）开启/关闭显示消息

图形用户界面, 文本, 应用程序

描述已自动生成

（3）

图形用户界面, 文本, 应用程序

描述已自动生成

（4）开启/关闭显示消息类型的事件内容

图形用户界面, 文本, 应用程序

描述已自动生成

（5）开启/关闭显示警告类型的事件内容

图形用户界面, 文本, 应用程序, 聊天或短信

描述已自动生成

（6）开启/关闭显示错误类型的事件内容

图形用户界面, 文本, 应用程序

描述已自动生成

（7）开启/关闭显示关键性错误类型的事件内容

图形用户界面, 文本, 应用程序

描述已自动生成

（8）配置过滤器也同样可以在控件运行时操作

图形用户界面, 应用程序

描述已自动生成

（9）配置事件显示的内容也同样可以在控件运行时操作

日程表

描述已自动生成

（10）确认选中的警报

图形用户界面, 文本, 应用程序

描述已自动生成

（11）确认所有警报

图形用户界面, 应用程序

描述已自动生成

（12）显示事件的内容区域

图形用户界面, 文本, 应用程序

描述已自动生成

# 使用Event Line控件

EventLine是EventGird是精简版，没有配置各种事件的按钮，只有事件内容的显示。也就是说，它的作用是仅仅显示TwinCAT Eventlogger所产生的内容。使用之前编写好PLC程序（和EventGird一致），然后将改控件拖动到界面当中即可：

日程表

描述已自动生成

图形用户界面, 文本

中度可信度描述已自动生成

EventLine有一个特殊的属性设置。当显示内容大于消息栏的长度时，可以选择自动滚动播放，设置路径为：Text —— TextOverflow。改属性一共有三种选项：（1）Ellipsis：超出消息栏长度的事件内容会被截断。 （2）：MarqueeLeftToRight：事件内容从左往右进行滚动。 （3）：MarqueeRightToLeft：事件内容从右往左进行滚动（下方的MarqueeSpeed则是用来设置滚动的速度）：

图形用户界面

描述已自动生成

# 常见问题

1. 可以在TwinCAT Event中编辑带有中文的内容吗？

在Text当中可以。

1. Event Line和Event Gird有什么区别？

接口都是TwinCAT3 EventLogger，没有区别，只不过更加简洁。

1. Event Gird和TwinCAT3软件底下的错误列表有什么关系？

可以理解为是精简版，Event Gird显示的内容一定会同时也在Error List下方产生，

Error List还带一些系统的报错或提示：

图形用户界面, 文本, 电子邮件

描述已自动生成

**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32号利通广场1303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 室

电话: 028-86202581 邮箱: chengdu@beckhoff.com.cn

|  |  |
| --- | --- |
| 请用微信扫描二维码  通过公众号与技术支持交流 | 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 |
|  |