**TF6020 TwinCAT 3 JSON Data Interface使用方法**

|  |  |
| --- | --- |
|  | 作者：徐良定  职务：华东区 技术工程师  邮箱：ld.xu@beckhoff.com.cn  日期：2024-04-19 |
| **摘 要：**  TC3 JSON Data Interface是一个通信接口，用于灵活地在TwinCAT系统和用户特定的应用程序之间交换数据。通信以JSON格式进行。因为使用了JSON格式和ADS或MQTT，确保了不同的编程环境都能够进行通用的访问。 | |
| **附 件：**   |  |  |  | | --- | --- | --- | | 序 号 | 文件名 | 备注 | |  |  |  | |  |  |  | |  |  |  | |  |  |  | | |
| **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 4](#_Toc168306581)

[1.1. 倍福Beckhoff 4](#_Toc168306582)

[1.1.1. 控制器硬件 4](#_Toc168306583)

[1.1.2. 控制软件 4](#_Toc168306584)

[1.2. 其它 4](#_Toc168306585)

[1.2.1. 硬件 4](#_Toc168306586)

[1.2.2. 开发软件 4](#_Toc168306587)

[2. 准备工作 4](#_Toc168306588)

[2.1. 创建PLC程序 4](#_Toc168306589)

[3. ADS通信测试 4](#_Toc168306590)

[3.1. Overview 5](#_Toc168306591)

[3.2. 新建Winform项目 5](#_Toc168306592)

[3.3. 添加Ads通信库引用 5](#_Toc168306593)

[3.4. 在NuGet上面安装Newtonsoft.Json包 7](#_Toc168306594)

[3.5. 调用ADS通信库的方法，建立PLC连接 8](#_Toc168306595)

[3.6. 读PLC变量 8](#_Toc168306596)

[3.6.1. API 介绍 9](#_Toc168306597)

[3.6.2. 在用户程序里面拼接Json文本 9](#_Toc168306598)

[3.6.3. 调用TcAdsClient的ReadWrite方法 10](#_Toc168306599)

[3.6.4. 返回的Json文本 11](#_Toc168306600)

[3.6.5. 解析Json文本 12](#_Toc168306601)

[3.7. 写PLC变量 12](#_Toc168306602)

[3.7.1. API介绍 12](#_Toc168306603)

[3.7.2. 在用户程序里面拼接Json文本 13](#_Toc168306604)

[3.7.3. 调用TcAdsClient的ReadWrite方法 14](#_Toc168306605)

[4. 常见问题 15](#_Toc168306606)

[4.1. Ads-Error 0x705:Parameter size is incorrect 15](#_Toc168306607)

[4.2. Ads-Error 0x706 :Parameter value(s) is invalid. 16](#_Toc168306608)

文档正文要求

**（文档编写完毕后，删除本页）**

|  |
| --- |
| 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：  1. 标注硬件完整型号及IMG，如CX5020-0125（IMG版本：CX1800-0411-0007 v3.92）； 2. 软件版本，如：TwinCAT 3.1 Build 4024.7  * 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（有条件建议使用示意图）、IP设置、加路由等 * 操作步骤：正常的操作截图，需保证截图完整清晰、步骤连续不跳步；部分操作说明需参考资料部分，请附带资料链接。 * 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；也可对客户提出的一些特征性问题进行总结进行记录，有步骤部分请分步说明、必要时需配图。   截图：   * 必须截出查找路径及关键部分。避免软件整个界面截图，影响阅读，使用红框突出重点，图片可适当添加文字说明。 * 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 |

# 软硬件版本

## 倍福Beckhoff

### 控制器硬件

无

### 控制软件

笔记本Win10

基于TwinCAT 3.1 Build 4024.53版本

TF6020 试用授权

## 其它

### 硬件

无

### 开发软件

VS2019 Community

# 准备工作

## 创建PLC程序

PLC程序里面包含了一些基础类型的变量，以及嵌套的结构体变量。

# ADS通信测试

在通常ADS通信中，读写PLC数据的方法有很多，比如通过结构体的方式、SumCommand的方式，但这些方式对于复杂的结构体或者多个变量同时读取，不够灵活。TF6020提供了以Json格式的方式读写变量，提高了变量读写的灵活性。

## Overview

文本, 信件

描述已自动生成

## 新建Winform项目

测试环境的框架版本的.NET Framework 4.5，根据实际情况进行选择。

图形用户界面, 应用程序

描述已自动生成

## 添加Ads通信库引用

1：右键选择添加引用

图形用户界面, 文本, 应用程序

描述已自动生成

2：点击浏览按钮

图形用户界面, 文本, 应用程序

描述已自动生成

3：选择对应文件目录下的TwinCAT.Ads.dll

截图里有图片

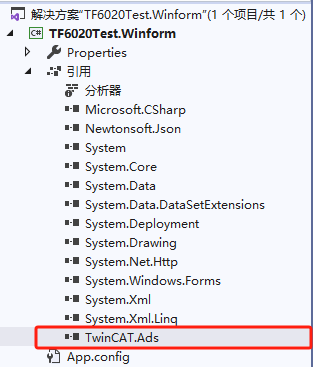
描述已自动生成

4：点击确定按钮

图形用户界面, 文本, 应用程序

描述已自动生成

添加ADS通信库后，如图所示



## 在NuGet上面安装Newtonsoft.Json包

图形用户界面, 应用程序, Word

描述已自动生成

图形用户界面, 文本, 应用程序

描述已自动生成

该包可以更灵活的拼接Json字符串以及解析Json字符串。

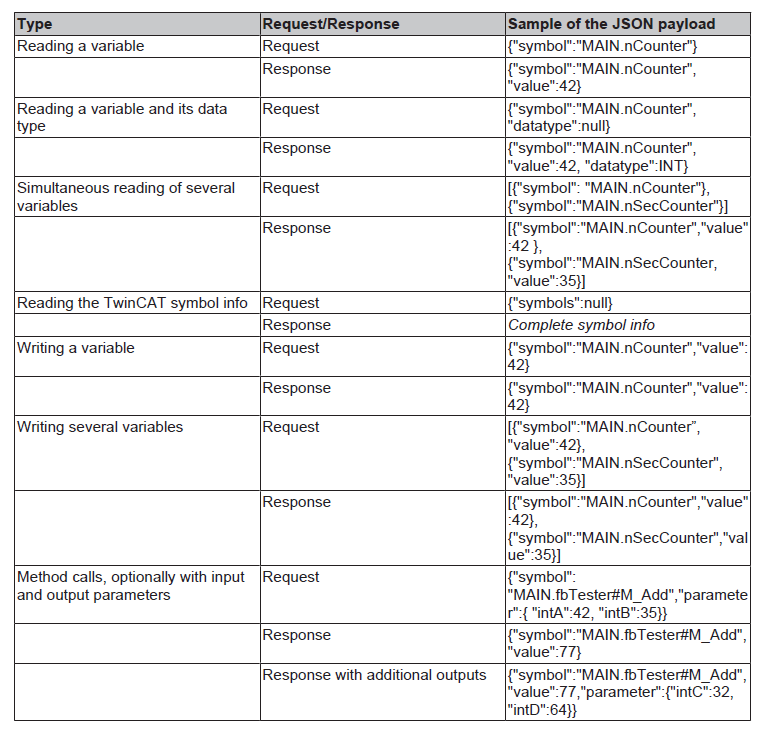
## 调用ADS通信库的方法，建立PLC连接

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

## 读PLC变量

### API 介绍



根据不同的指令，发起请求的Json的格式如上。

在本次测试中，需要一次性的把PLC里面多个变量的信息读取上来，以减少通信往返的次数，节省通讯时间。需要使用下图这种格式的Json字符串。



### 在用户程序里面拼接Json文本

把需要读取的变量按照上面介绍的API格式要求进行拼接。代码参考如下。

图形用户界面, 文本, 应用程序

描述已自动生成

此处Json文本拼接仅作演示，实际项目可以使用例如Newtonsoft.Json等库进行拼接。

在这边可以发现，变量读取有以下灵活的地方。

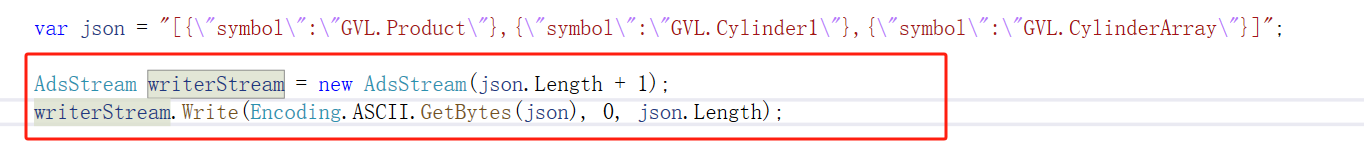
1：对于PLC里面的结构体，用户程序里面不需要提前定义一个和PLC里面结构体一样的Class，只需要传入该结构体变量在PLC里面的符号名称。该结构体里面的变量，可以在运行时从返回的Json中进行解析。

2：可以一次通信往返同时读取多个结构体变量或者普通的变量。需要读取的变量在Json文件里面添加即可。

3：PLC结构体里面新增变量、调整变量的位置，用户程序这边不需要调整代码。

### 调用TcAdsClient的ReadWrite方法

定义writeStream，把上面生成的Json文本填充到字节数组里面。注意字节数组的长度要在Json文本的长度上加1。



定义readStream，用来接收读写方法返回的字节数组。字节数组的长度需要根据实际变量的多少来改变，长度可以设置的大一点。

图片包含 图表

描述已自动生成

调用ReadWrite方法，读取PLC里面的数据。ReadWrite方法的前两个参数填写固定值，分别是0xf070以及0。

数据结果是字节数组，需要用Encoding.ASCII.GetString方法转换成string类型，也就是返回的Json文本。

文本

描述已自动生成

返回的Json文本如下图：

图形用户界面, 文本

描述已自动生成

### 返回的Json文本

首先观察Json文本的格式

由于请求的方法是读取多个变量，那么返回的结果也是数组。对于数组里面每个Json对象，都由“symbol”和“value”这两个对象组成。“symbol”包含了变量的符号地址信息，“value”里面则包含了这个符号的变量值。

对于PLC里面嵌套的结构体，返回的Json文本里面同样也是嵌套的结构。这样利用Json解析的库，可以很容易的解析出我们需要的变量值。

### 解析Json文本

具体解析Json文本的代码如下，经供参考。此处使用了Newtonsoft.Json库。

文本

低可信度描述已自动生成

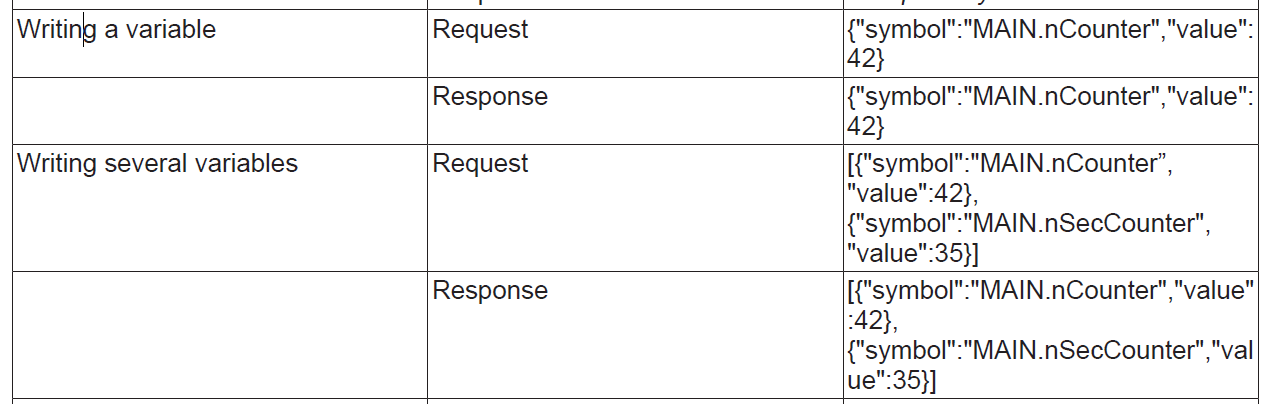
## 写PLC变量

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

根据API，写PLC变量值可以分成写单个变量和写多个变量。本文档演示写多个PLC变量值。写多个变量用户程序和PLC之间也是一次通信往返，因此不会造成通信时间的增加。

### API介绍



根据API，写PLC变量值可以分成写单个变量和写多个变量。本文档演示写多个PLC变量值。写多个变量用户程序和PLC之间也是一次通信往返，因此不会造成通信时间的增加。

### 在用户程序里面拼接Json文本

把需要写值的变量按照上面介绍的API格式要求进行拼接。代码参考如下。



此处Json文本拼接仅作演示，可以根据实际情况选择使用哪种方式或者其它Json库进行字符串的拼接。代码运行后，拼接完成后的Json如下：

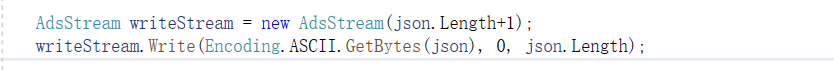
图形用户界面, 文本, 应用程序

描述已自动生成

写入变量可以选择写入结构体里面的单个变量，而不会影响结构体里面其它数据。

### 调用TcAdsClient的ReadWrite方法

定义writeStream，把上面生成的Json文本填充到字节数组里面。注意字节数组的长度要在Json文本的长度上加1。

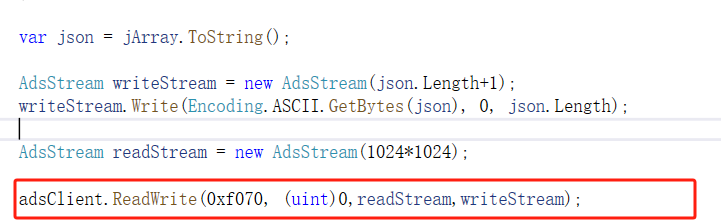


定义readStream，用来接收读写方法返回的字节数组。字节数组的长度需要根据实际变量的多少来改变，长度可以设置的大一点。



调用ReadWrite方法，读写PLC里面的数据。ReadWrite方法的前两个参数填写固定值，分别是0xf070以及0。

数据结果是字节数组，需要用Encoding.ASCII.GetString方法转换成string类型，也就是返回的Json文本。此处返回的数据是写入的变量之后的值，因为是写入，此处返回的结果可以不进行解析。



# 常见问题

## Ads-Error 0x705:Parameter size is incorrect

图形用户界面, 文本, 应用程序

描述已自动生成

用于接收读取结果的缓冲区长度不够。增加读取字节数组的长度。

图形用户界面, 文本, 应用程序

描述已自动生成

## Ads-Error 0x706 :Parameter value(s) is invalid.

图形用户界面, 文本, 应用程序

描述已自动生成

检查writeStream的长度是否是在json文本的长度上+1.

