|  |  |  |
| --- | --- | --- |
| 作者： | 冯建鹏 | 上海市江场三路市北工业园区  163号5楼（200436）  TEL: 021-66312666  FAX: 021-66315696 |
| 日期： | 2018.03 |
| 版本： | V1.0 |
| E\_mail: | 909843413@qq.com |

|  |
| --- |
| **关于TC2、TC3寻址差异的说明** |

|  |
| --- |
| 概 述 |

|  |
| --- |
| TC2地址寻找采用的是字节寻址（byte-addressing）模式、TC3采用的是面向字（word-orientated）的IEC寻址模式。本文介绍两种寻址的差异以及注意。 |

|  |
| --- |
| 文档中包含的文件 |

|  |  |
| --- | --- |
| 文件名称 | 文件说明 |
| byteaddressing\_word-orientated1.pro | TwinCAT2的寻址测试 |
| byteaddressing\_word-orientated2.tpzip | TwinCAT3的寻址测试 |

|  |
| --- |
| 备 注 |

|  |
| --- |
| 关键字：地址分配 |

|  |
| --- |
| 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。  *文档内容可能随时更新*  *如有改动，恕不事先通知* |

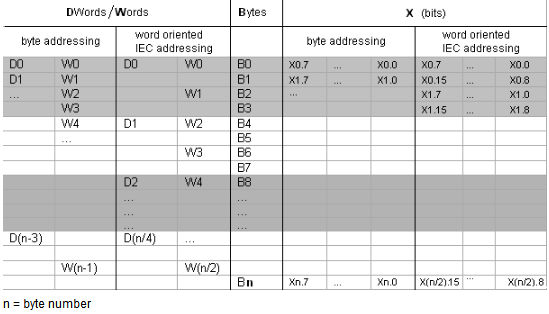
1. **TC2和TC3在变量地址分配上的区别。**

在PLC编程中时我们经常使用%I\*、%Q\*、%M\*给变量指定地址以方便变量绑定或者其他用途。但是由于TwinCAT2和TwinCAT3采用不同的IEC 寻址模式，在部分指令的含义上有很大不同。比如PLC中同一条分配地址语句（如%MD10）在TwinCAT2和TwinCAT3中可能指向的是不同的内存区域。

TwinCAT 2的寻址方式采用的是字节寻址（byte addressing）模式，TwinCAT 3 采用的是面向字（word-orientated）的IEC寻址模式。

两者主要在地址的解释上有差异，下面是两者的具体差异：

<http://infosys.beckhoff.com/content/1033/tc3_plc_intro/9007201784101515.html?id=2366751467445941503>



1. 在Byte寻址上两者是一样，比如%IB1、%QB2、%MB3等，他们在TC2、TC3指向的是同一个内存区域。
2. 在Word、DWord寻址上不一样

在TC2中，Word、DWord的标号是按照Byte的标号排序的。比如%MW(n) ，表示这个Word数据的首地址为M区第n个Byte；同理%ID(n) ，表示这个DWord数据的首地址为I区第n个Byte。

在TC3中，Word、DWord的标号跟Byte的标号是成倍数关系的。比如%MW(n) , 表示这个Word数据的首地址为M区的第2\*n个Byte（Word=2 Byte）；同理%ID(n) , 表示这个DWord数据的首地址为I区的第4\*n个Byte（DWord=4 Byte）

1. 在位寻址上也需要结合上述情况

TC2中，直接指定某个Byte的标号加上位号就能找到这个位。比如 %QX8.5 表示这个位在Q区的第8个Byte的第5位。%QX8.14表示Q区的第8个WORD的第14位。（等同于M区第9个Byte的第7位）

TC3中，比如 %MX8.14 表示这个位在M区的第4个Word的第14位（等同于M区的第9个Byte的第7位）。

1. **注意事项**
2. 请在TwinCAT2中给Word、DWord数据分配地址时需要留足够的空间，以免两个数据的地址发生重叠，造成数据丢失的情况。

比如：

Var1 AT%MD0 : DWORD;

Var2 AT%MD2 :DWORD;

以上两个变量的地址会发生重叠，因为Var1将占用Byte0~Byte3的空间，Var2将占用Byte2~Byte5的空间，Byte2~Byte3重叠。

1. TwinCAT2项目转TwinCAT3项目时，应该注意程序中变量分配的地址有没有发生改变。
2. 当给变量分配的地址小于变量的长度时，可能会有溢出。

比如： Var3 AT%MB0 : WORD;

Var3是一个Word型变量占用两个字节，但是只分配在%MB0 这一个字节上，实际上Var3将会占用%MB0和%MB1这两个字节，当Var3的数值大于一个字节时，将会改变%MB1的内容，造成溢出这种情况。