

0822 【万泉河】解读一个用循环法 PK 万泉河 80 模拟量例子的例子程序

大约两个月前，写过一篇文章：

《0628 【万泉河】优雅的 PLC 程序一定是用 EXCEL 写出来的》，其中介绍了我用 EXCEL 做出 80 个模拟量转换程序。

众多网友看到后，大呼有收获。

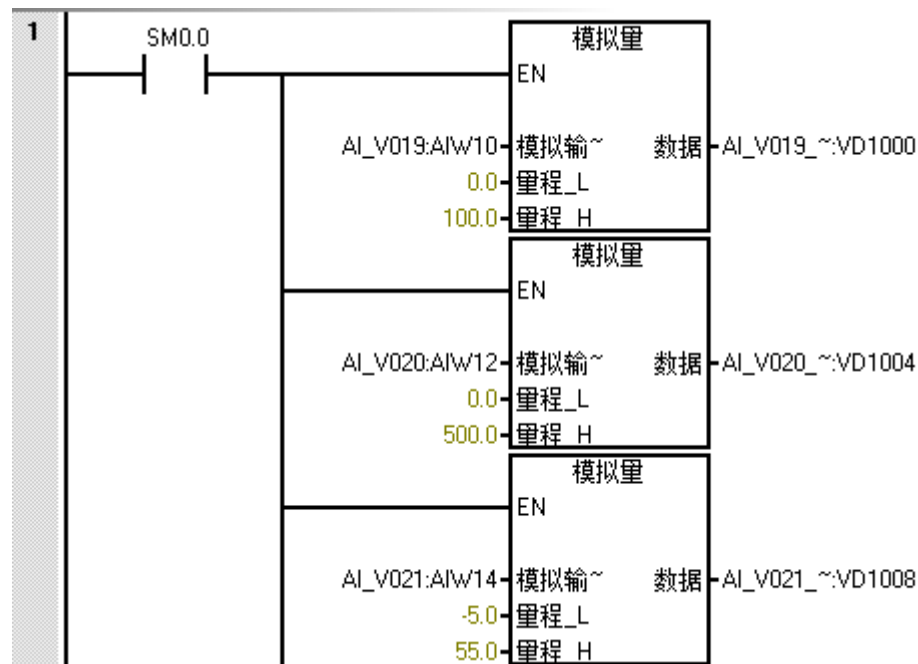
然而也有另外一批同行，见到我的程序之后，直呼笑掉大牙。嘲笑我竟然连循环语法都不会。所以，经常在一些微信群中，遇到一些热心网友要教给我循环语法。

然后我就说，不要只看循环的那三五句语法，不要做口炮党，动下手，你们按同样的变量表，把同样功能的程序做一遍出来。然后大家比较一下。

然后来了。

下面从原理开始讲解一下。

我原始的优雅程序，在 PLC 程序中，就是如图这样的简单罗列调用。

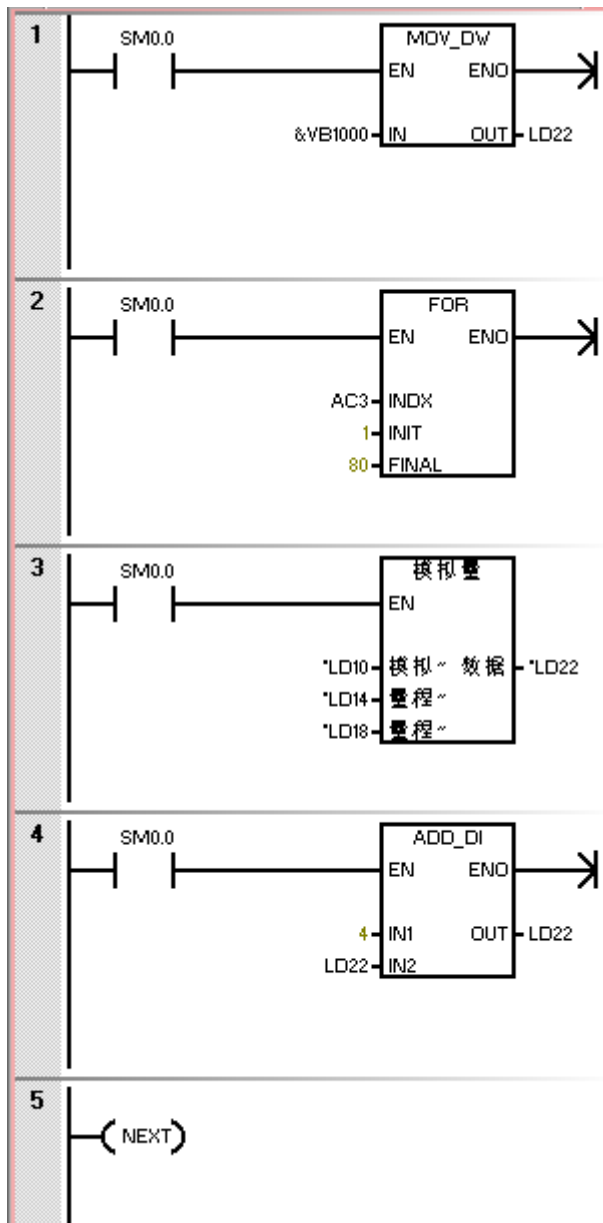


循环派说，你看看，后面的数据结果 VD1000, VD1004, VD1008 是有地址规律的数据，可以用指针，间接寻址或者数组的方式给序列化。具体方法取决于所使用的 PLC 平台支持情况。比如 SMART 200，就可以用指针。

先把 VD1000 指针化，送到 LD22 中，循环中调用一次数值送到 LD22 指向的地址，执行完成之后指针增加 4，即指向了 VD1004，下一周期即处理了第二个模拟量数据。

如此循环 80 次，即只使用区区 5-6 行语句就实现了所有模拟量的处理。而如果数量有改变，比如改为 800 个，那也只是把上面的循环数由 80 改为 800 即可。

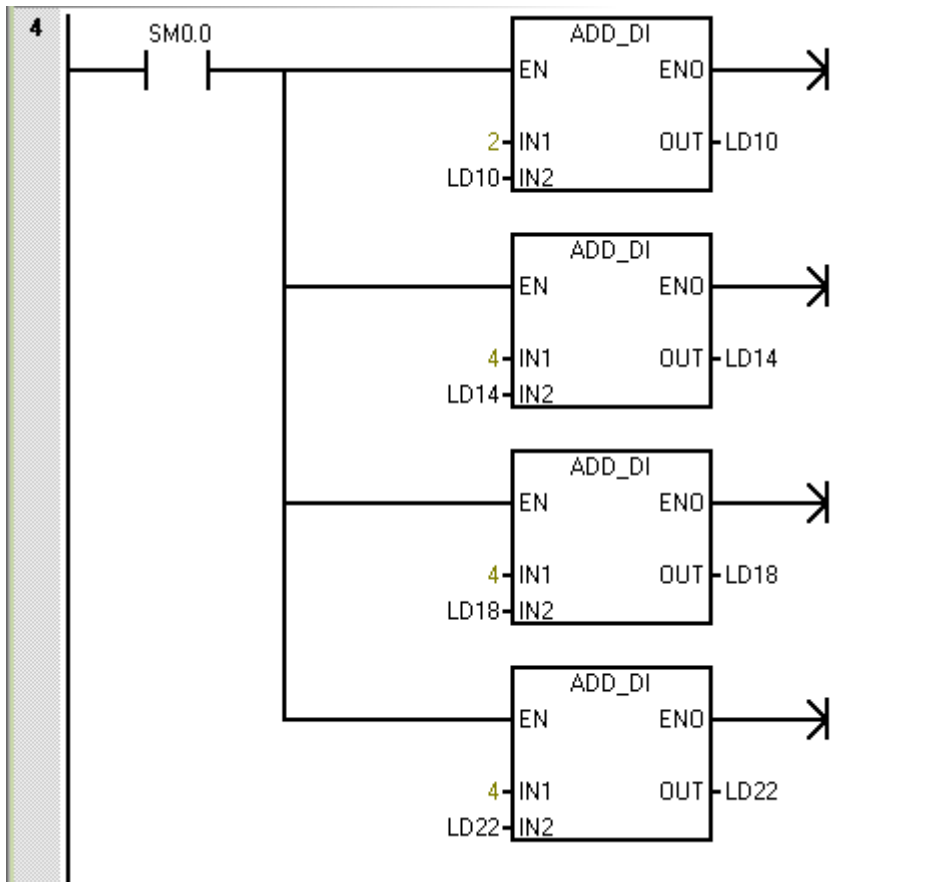
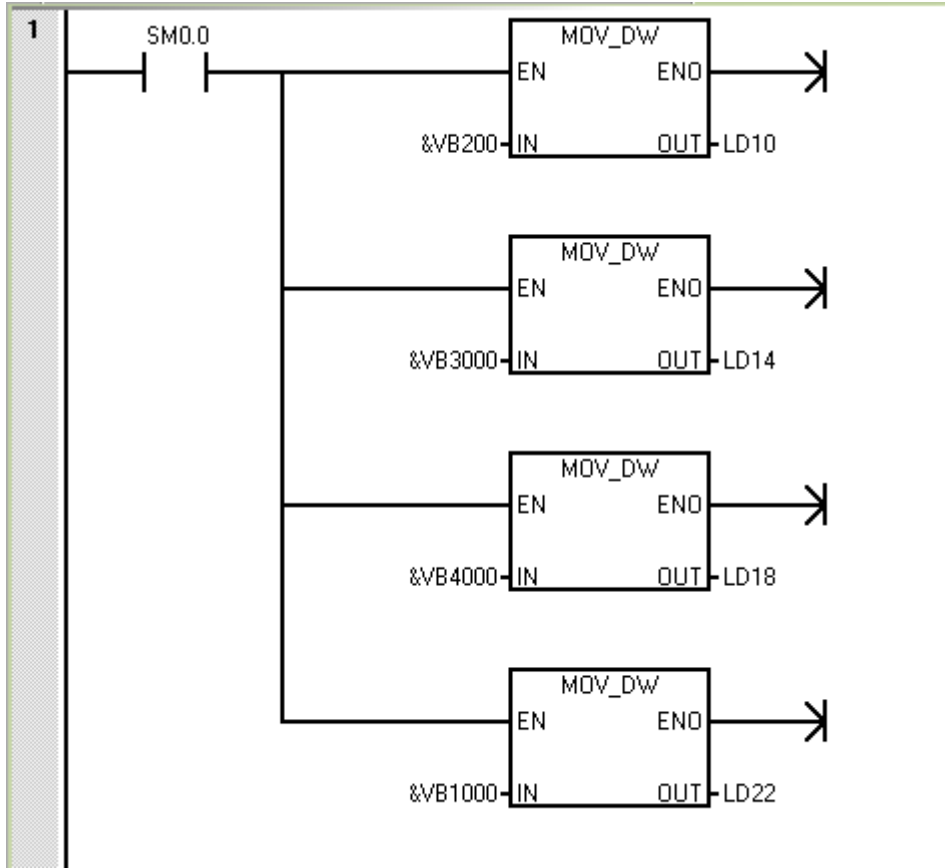
即



然而，事情其实并非如此简单。不仅输出侧变量需要序列化，输入侧的变量也同样需要。首先，量程上下限对每一个测点来说都不一样，那么就无法使用固定值，也需要做成变量组，而输入的模拟量通道地址，也不会是完美连续的。比如有时候数据来自通讯。比如我做例子的时候有一部分数据直接来自 VW。

那么需要对这些数据做 IO 映射处理。即，模拟量通道部分，分配了 VW2000 开始的数据区，量程上下限则分别分配了 VD3000,VD4000 的数据区，分别传送到指针 LD10, LD14, LD18 中。

所以，真实可用的程序是：



程序行数增加了一点点，忽略不计。

然而 IO 映射就没有那么容易了。因为地址并不能保证规则连续，所以必须逐行用 MOVE 指令来传送。这样子：

```
MOVW AI_V064:AIW100, VW2090
MOVW AI_V065:AIW102, VW2092
MOVW AI_V066:AIW104, VW2094
MOVW AI_V067:AIW106, VW2096
MOVW AI_V068:AIW108, VW2098
MOVW AI_V069:AIW110, VW300
```

```
MOVW AI_V070:VW2, VW302
MOVW AI_V071:VW4, VW304
MOVW AI_V072:VW6, VW306
```

总计有 80 行程序，一行都少不了。我这里只是截取了中间的部分，看到数据地址是不规则的。

而量程上下限，常数值可以用 MOVD 方法同样实现，也可以直接定义到数据块中，数据块增加 2 个表量程 L 和量程 H，录入数据分别为：

```
VD3000 0.0
VD3004 0.0
VD3008 -5.0
VD3012 0.0
VD3016 0.0
VD3020 0.0
VD3024 -5.0
VD3028 0.0
VD3032 0.0
VD3036 0.0
VD3040 -5.0
VD3044 0.0
VD3048 0.0
VD3052 0.0
VD3056 -5.0
VD3060 0.0
VD3064 0.0
VD3068 0.0
```



```
VD4000 100.0
VD4004 500.0
VD4008 55.0
VD4012 100.0
VD4016 100.0
VD4020 500.0
VD4024 55.0
VD4028 100.0
VD4032 100.0
VD4036 500.0
VD4040 55.0
VD4044 100.0
VD4048 100.0
VD4052 500.0
VD4056 55.0
VD4060 100.0
VD4064 100.0
VD4068 500.0
VD4072 55.0
VD4076 100.0
VD4080 100.0
VD4084 500.0
```



每个表分别是 80 行， $80*3=240$ 。

然而这样的程序是不完整的。使用掉的 V 区数据都需要做到符号表中，给分配符号名称。就像我原本给 QOUT 分配到 V1000 一样。

那么数据工作量为 $80*3=240$ 行。

这些数据，以及 MOVE 指令做 IO 映射的程序，都需要从 EXCEL 整理好之后复制过来。所以可以认为其实数据块和符号表与程序基本相同的工作量，都是程序的一部分。

所以程序的总量 480 行比我不用循环的做法其实反而增加了 400 行。

然而，我做程序时候，EXCEL 表格是一次性完成的，这里需要把工艺变量表拆分到 6 个表中，整理好了之后分别复制 6 次。这里模拟量处理的输出部分恰巧是内部数据，所以可以是完整的序列。而其它类的设备，输出部分也需要做 IO 映射，也仍然需要工作量。

是的。这个 PK 程序其实也仍然是我自己亲自做的。我虽然早就知道这里面的全部的工作量，我写上一篇文章原本就是针对的某些文章和视频做的所谓循环处理技巧，我自己工作中断然不会使用，但为了把例子贡献出来，还是忍着恶心给完整做了一下。

这个循环法实现的程序，为了爽那一下下的几句指令，同时还消耗了 V2000-V4000 的 3000byte 的数据区。而且，程序也仍然没有做到彻底标准化。比如，1000 个 byte，只能存放 250 个浮点数数据，即，如果真的要处理 800 个模拟量，可不是仅仅把 80 改为 800，而是所有数据区规划都需要更改。因为原本的规划区不够用了，循环跑不成去了。

而倒过来，如果只是 8 个模拟量的程序，现在的这套浪费了 3000 个数据区的程序，也是搞笑的。浪费量惊人，也需要优化修改。

所以其实就是，你没针对一个新项目，都需要根据实际的模拟量数量，重新做数据区优化规划。

那些嘲笑我，要教我循环语法的骄傲的年轻人，只执着于跟我演示那三五句语法，我都懒得看，他们却非常自豪的要反复演示，以为那已经是自动化工作顶尖绝技。我反复提醒他们动动手把程序做完整的时候，理都不理。还只在那里教育我勿要做井底之蛙，不晓得天外有天。

指望着那些又轻浮又懒惰的年轻人，我永远也得不到跟我 PK 的样本。

我这里辛辛苦苦把对标 PK 的例子程序做出来，提供给那些懒得动手的循环派年轻人，如果我了解有偏差之处，还可以在我基础上再修改完善，哪些工作是因为我愚蠢做多了，可以做减法删除掉，这样对他们来说，就不会畏惧亲自动手了。

每一个年轻人，未来要有成长，在勤奋与聪明之间，必须拥有一样。而且很多时候勤奋比聪明还更重要一点。尤其是你怎么点化都不能领悟的自以为是的小聪明。

我这里把 PK80 程序同样放到 80 工位双联开关 QQ 群文件中，群 1 已经满员，群 2 还有空位，可以申请加入 870566908 后获取。