



原创文章，转载请注明出处。

更多实用资料请登录方正智芯官网：www.founderchip.com

作者：北岛李工

上一篇文章我们谈到了SCL的沿检测指令——R_TRIG和F_TRIG。细心的小伙伴在阅读文章后提出一个问题：如果每一个沿检测指令都要使用一个数据块（DB），那多次使用的情况下程序中岂不是会有很多的数据块？这篇文章，我们再来谈谈这个话题。

SIEMENS
Ingenuity for life

西门子PLC高级编程语言SCL入门教程

第六篇：再谈沿检测指令

<https://www.founderchip.com>

的确，如果在FC（功能）中使用沿检测指令，我们必须为其创建相应的数据块。这是因为FC没有属于自己的背景数据块，不能保存静态变量。

如果在程序中需要多次使用沿检测指令，建议使用FB（功能块）而非FC（功能）。由于FB有自己的背景数据块，这样，我们可以把沿检测指令需要保存的静态数据存放到FB的背景数据块中，并且可以采用多重背景数据块的方式来减少程序中需要添加的数据块（DB）的数量。

举个例子：

假设我们有三台电机需要进行起停控制。

启动按钮的PLC地址分别是：

I0.0（标签名：E_Start_Motor1）；

I0.1（标签名：E_Start_Motor2）；

I0.2（标签名：E_Start_Motor3）；

添加FB到主程序中，命名为：FB_SCL_Test.

向FB中添加上升沿触发指令：R_TRIG,在出现添加背景数据块对话框中选择“多重背景数据”，并命名为：R_TRIG_Motor1，如下图：

此时可以看到，FB_SCL_Test的静态存储区中增加了R_TRIG_Motor1一项，如下图：

FB_SCL_Test								
	Name	Data type	Default value	Retain	Accessible f...	Visible in ...	Setpoint	Comment
6	▀ <Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
7	▾ Static				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
8	▀ ▸ R_TRIG_Motor1	R_TRIG			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
9	▀ <Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
10	▾ Temp				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
11	▀ <Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
12	▾ Constant				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
13	▀ <Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

同样的方式添加Motor2和Motor3的上升沿触发指令，背景数据块名称分别为：R_TRIG_Motor2和R_TRIG_Motor3，如下图：

FB_SCL_Test								
	Name	Data type	Default value	Retain	Accessible f...	Visible in ...	Setpoint	Comment
6	▀ <Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
7	▾ Static				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
8	▀ ▸ R_TRIG_Motor1	R_TRIG			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
9	▀ ▸ R_TRIG_Motor2	R_TRIG			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
10	▀ ▸ R_TRIG_Motor3	R_TRIG			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
11	▀ <Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
12	▾ Temp				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
13	▀ <Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
14	▾ Constant				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
15	▀ <Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

接下来我们编程来显示三台电机启动按钮的上升沿检测，这里我们采用两种代码方式：

第一种方式我们采用上篇文章介绍的方法，使用中间变量M0.0（标签名称：Interflag_Motor0），代码如下：

```
IF... CASE... FOR... WHILE... (*...*)
OF... TO DO... DO...
1 //motor 1 rising edge detect
2 //www.founderchip.com
3 #R_TRIG_Motor1(CLK:="E_Start_Motor1",
4 | Q=>"Interflag_Motor0");
5 |
```

第二种方式采用简单的方法，不使用中间变量，代码如下：

```
6 //motor 2 rising edge detect
7 #R_TRIG_Motor2(CLK:="E_Start_Motor2");
8 //motor 3 rising edge detect
9 #R_TRIG_Motor3(CLK:="E_Start_Motor3");
```

第一种方式的上升沿检测，可以通过判断中间变量M0.0的值是否为1来启动电机，代码如下：

```

11 //start motor 1
12 IF "Interflag_Motor0" = true THEN
13     "Q_Motor1" := 1;
14 END_IF;
15

```

第二种方式的上升沿检测，则需要判断背景数据块中的静态变量来启动电机，代码如下：

```

16 //start motor 2
17 //www.founderchip.com
18 IF #R_TRIG_Motor2.Q = true THEN
19     "Q_Motor2" := 1;
20 END_IF;
21 //start motor 3
22 IF #R_TRIG_Motor3.Q = true THEN
23     "Q_Motor3" := 1;
24 END_IF;
25

```

可以看出，第一种代码的可读性很好，但显得有点啰嗦；第二种代码很简洁，但可读性稍微差些。选择哪种方式，完全取决于你的爱好。

下降沿的多重背景设置与此类似，不再赘述。

代码写到这里并没有完成。虽然我们在FB_SCL_Test中设置存储静态变量，但到目前为止FB_SCL_Test还是没有背景数据块。

所以，接下来我们要创建一个背景数据块。

可以手动创建。不过简单的办法是在主程序块Main中调用FB_SCL_Test，系统会自动提示创建背景数据块FB_SCL_Test_DB，单击确认即可。

打开数据块FB_SCL_Test_DB，可以看到其静态存储区中有我们设置的R_TRIG_Motor1、R_TRIG_Motor2和R_TRIG_Motor3，这样程序就可以正常运行了。如下图：

Project13_SCL > PLC_1 [CPU 1215C DC/DC/DC] > Program blocks > FB_SCL_Test_DB [DB1]							
FB_SCL_Test_DB							
	Name	Data type	Start value	Retain	Accessible f...	Visible in ...	Setpoint
1	Input			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	Output			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3	InOut			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	Static			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5	▶ R_TRIG_Motor1	R_TRIG		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
6	▶ R_TRIG_Motor2	R_TRIG		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
7	▶ R_TRIG_Motor3	R_TRIG		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

好了，关于沿检测指令我们又讨论了一篇，感谢那个提问题的小伙伴。看似很简单的沿检测指令，仔细思考还是有些内涵。欢迎扫描下方二维码关注我们的微信公众号。



方正智芯
Founder Chip

长按扫码关注我们



方正智芯

公众号：founderchip
官方网站：www.founderchip.com
原创工业智能控制领域（PLC、单片机、通信）的技术分享

