

伺服组件

- 1、学习伺服配置
- 2、学习伺服定位
- 3、学习程序编写

伺服组件

- 1.1、伺服配置
- 此处讲解了如何对伺服配置

伺服

气缸

检测器

特殊

全局

伺服00 - 平移伺服

当前位置:	0	手动速度:	2000
单位脉冲数:	100	正限位(软):	99999999
加减速时间:	30	负限位(软):	-99999999
找原速度:	2000	正限位(X):	8
爬行速度:	500	负限位(X):	8

固定参数: Y0脉冲 Y4方向 X4原点、定位M0、增量M1

点动进

点动退

找原点-M14

定原点-M15

停止-M16

回零点-M17

- 1、当前位置: 此处显示实时位置
- 2、单位脉冲数: 更具伺服驱动器设置的参数来配置, 驱动器设1000, 此处就设1000
- 3、加减速时间: 此处为30, 代表加速到设定速度需要30MS
- 4、找原速度: 以设定速度快速接近原点
- 5、爬行速度: 以找原速度接近原点以后以爬行速度离开原点
- 6、手动速度: 以此速度手动调试伺服电机
- 7: 正限位(软): 以正转运行的最大距离, 超过此距离设备不会再往前运行
- 8: 负限位(软): 以反转运行的最大距离, 超过此距离设备不会往后运行
- 9: 正限位: 在模组前进的极限位置处按照的感应器, 此感应器亮后设备不会在继续往前运行
- 10: 负限位: 在模组后退的极限位置处按照的感应器, 此感应器亮后设备不会在继续往后运行
- 11: 点动进: 鼠标点击此按钮设备往前运行 (如果不是前进是后退运行, 此时需要更改伺服驱动器参数把运行方向更改一下)
- 12: 点动退: 鼠标点击此按钮设备往后运行 (如果不是后退是前进运行, 此时需要更改伺服驱动器参数把运行方向更改一下)
- 13: 找原点: 如果设备有设置原点, 可以点击此按钮会以找原速度快速找到原点, 再以爬行速度离开原点, 此时当前位置变0, 代表找到原点
- 14: 定原点: 在设备任意位置, 点击此按钮定原点, 然后设备依照此位置每次回零点
- 15: 停止: 在设备运行过程中, 点击此按钮运行停止, 伺服回停止
- 16: 回零点: 在设备定原后, 点击此按钮伺服会回零点

伺服组件

• 2.1、伺服定位

- 此处讲解了如何对伺服进行定位

点位1:	<input type="text" value="0"/>	速度1:	<input type="text" value="2000"/>	取点1	定位1-M30
点位2:	<input type="text" value="0"/>	速度2:	<input type="text" value="2000"/>	取点2	定位2-M31
点位3:	<input type="text" value="0"/>	速度3:	<input type="text" value="2000"/>	取点3	定位3-M32
点位4:	<input type="text" value="0"/>	速度4:	<input type="text" value="2000"/>	取点4	定位4-M33
点位5:	<input type="text" value="0"/>	速度5:	<input type="text" value="2000"/>	取点5	定位5-M34
点位6:	<input type="text" value="0"/>	速度6:	<input type="text" value="2000"/>	取点6	定位6-M35
增量7:	<input type="text" value="0"/>	速度7:	<input type="text" value="2000"/>		增量7-M36

下载到PLC

1、定位：目前伺服定位支持6个位置，如果不够可以直接自己写指令写位置（定位1-M30---定位2-M35）

M30 M31 M32 M33 M34 M35

2、取点：手动点击前进或者后退，在设备合适的位置取点会把当前位置自动给输入进来

（假设点击取点1，定位1的数据就会保存记录下来）

当取点全部取好，配置全部配置好，需要点击下载PLC把数据下载进PLC，这时直接运行M30就会取取点1的位置，运行M31就会去取点2的位置，往后类推

伺服组件

- 3.1、编写伺服程序

- 此处讲解了伺服程序如何编写

扫描帧<开机自启>	主任务<S1>	子任务<S2>	子任务<S3>	子任务<S4>	子任务<S5>	子任务<S6>	子任务<S7>
0	执行动作/M30			//平移伺服-定位1			
1	等待动作/M30/完成						
2	执行动作/M17			//平移伺服-回零点			
3	等待动作/M17/完成						
4	结束任务						

此处编写了一个简单的实例
程序流程：

- 1、伺服去定位1的位置
- 2、等待伺服移动到位置
- 3、伺服回零点也就是回待机位置
- 4、等待伺服回零点
- 5、程序结束