

江国民 688129

江国民 688129

江国民 688129

江国民 688129

江国民 688129

# SF5181: Interpolator操作手册



中科时代  
SINSEGYE

<SF5181: Interpolator>

作者

中科时代 | 基于PC技术的工智机新时代

日期	江国民 688129		江国民 688129	深圳市南山区粤海街道百度国际大厦西塔楼
版本	江国民 688129 V1.0.0		江国民 688129	官网: <a href="http://www.sinsegye.com.cn">www.sinsegye.com.cn</a>
Email	江国民 688129		江国民 688129	邮箱: <a href="mailto:Sales@sinsegye.com.cn">Sales@sinsegye.com.cn</a> 热线电话: 400-013-2158



# 基于PC技术的工智机新时代

## Interpolator使用介绍

### 前言

### 一、文件说明

本说明专为熟悉相关国家标准且经过专业培训的控制与自动化技术专家而制定。

在安装与调试部件时，务必仔细审阅所有相关文件及以下说明。

合格人员应始终采用最新的有效文档进行操作。

责任人员必须确保所述产品的应用或使用完全符合所有安全要求，涵盖所有相关法律法规、指导原则及标准。

### 1、免责声明

本文件经过精心编制，但鉴于所描述产品处于持续的开发与升级过程中，中科时代（深圳）计算机系统有限公司保留随时对文件进行修改和更新的权利，且无需事先通知。请注意，禁止依据数据图及本文件描述对已交付的产品进行任何改动。

对于因使用或信赖本手册所载明或未明示的信息而造成的任何损失或损害，中科时代计算机系统有限公司不承担任何责任。

### 2、版权所有

本手册的所有权归中科时代计算机系统有限公司所有。未经书面许可，任何人不得以任何形式复制、分发、翻译或以其他方式使用本手册的全部或部分内容。

本手册受版权法保护。任何对本手册内容的复制、分发、翻译、展示、表演、演绎或使用，无论出于何种目的，均需得到中科时代计算机系统有限公司的明确许可。未经许可，任何行为均视为侵犯中科

## 二、安全声明

### 1、安全规程

为了您的安全，请阅读以下说明。始终遵守产品特定的安全说明，您可以在本文档的适当位置找到这些说明。

### 2、责任免除

所有组件都提供了硬件和软件配置。不允许对文件中所述以外的硬件或软件配置进行修改，中科时代不对文件所述外的硬件或软件负责。

### 3、人员资格

本说明仅适用于熟悉适用国家标准的经过培训的控制、自动化和驱动技术专家。

### 4、信号词

文档中使用的信号词分类如下。为了防止人员和财产受到伤害和损害，请阅读并遵守安全和警告通知。

### 5、个人伤害警示

	<p><input type="checkbox"/> <b>警告</b></p> <p>危险的类型 说明不避开危险的后果 说明如何避免危险的发生</p>	<p>警告表示一种潜在的危险情况，如果不加以避免，可能会导致严重的伤害或死亡。</p>
	<p><input type="checkbox"/> <b>注意</b></p> <p>危险的类型 说明不避开危险的后果 说明如何避免危险的发生</p>	<p>注意表示潜在的危险情况，如果不避免，可能会导致轻度受伤或中度受伤，或导致设备损坏。</p>
	<p><b>提醒</b></p> <p>危险的类型 说明不避开危险的后果 说明如何避免危险的发生</p>	<p>注意表示一种潜在的危险情况，如果不加以避免，可能只导致设备的损坏。</p>

### 6、对财产或环境造成损坏的警告

	<p>环境、设备或数据可能会被损坏。</p>
--	------------------------

注意
危险的类型
说明不避开危险的后果
说明如何避免危险的发生

## 7、产品处理信息

例如，这些信息包括：行动建议、援助或有关产品的进一步信息。

### 概述

本产品是一款高性能的运动控制工具，提供高效、精确的XYZ三轴连续插补和避障功能，旨在优化路径规划和提高运动控制的安全性。该库适用于多轴运动控制，尤其是在复杂环境中执行精密任务时，能够有效避免碰撞并确保最优路径。

库实现了从起始位置到目标位置的多段连续轨迹插补，支持设置每段插补的速度、加速度、减速度等参数。若未明确设置参数，则会使用默认值进行插补。提供了T形和S形加减速选项及灵活的过渡曲线类型，确保运动的平滑性与精确性。

避障功能适用于XYZ直角坐标系的机器人设备。障碍物的尺寸和位置通过虚拟障碍物模型定义，库支持设定安全边距和障碍物的上下表面高度，机器人根据设定的高度和距离调整Z轴运动，确保避免障碍物的干扰，安全到达目标位置。

下表概述了各个产品组件

产品组件	描述说明
FB_InterpolatorXYZ	XYZ三轴连续插补
FB_PickBond	避障功能

## 安装卸载

### 一、安装要求

本节描述工程和/或运行时系统所需的最低要求。

#### 1、硬件要求

支持MetaFacture平台的设备（PLC或嵌入式系统）。

#### 2、软件要求

已安装MetaFacture的开发环境。

操作系统支持：MetaOS或其他兼容平台。

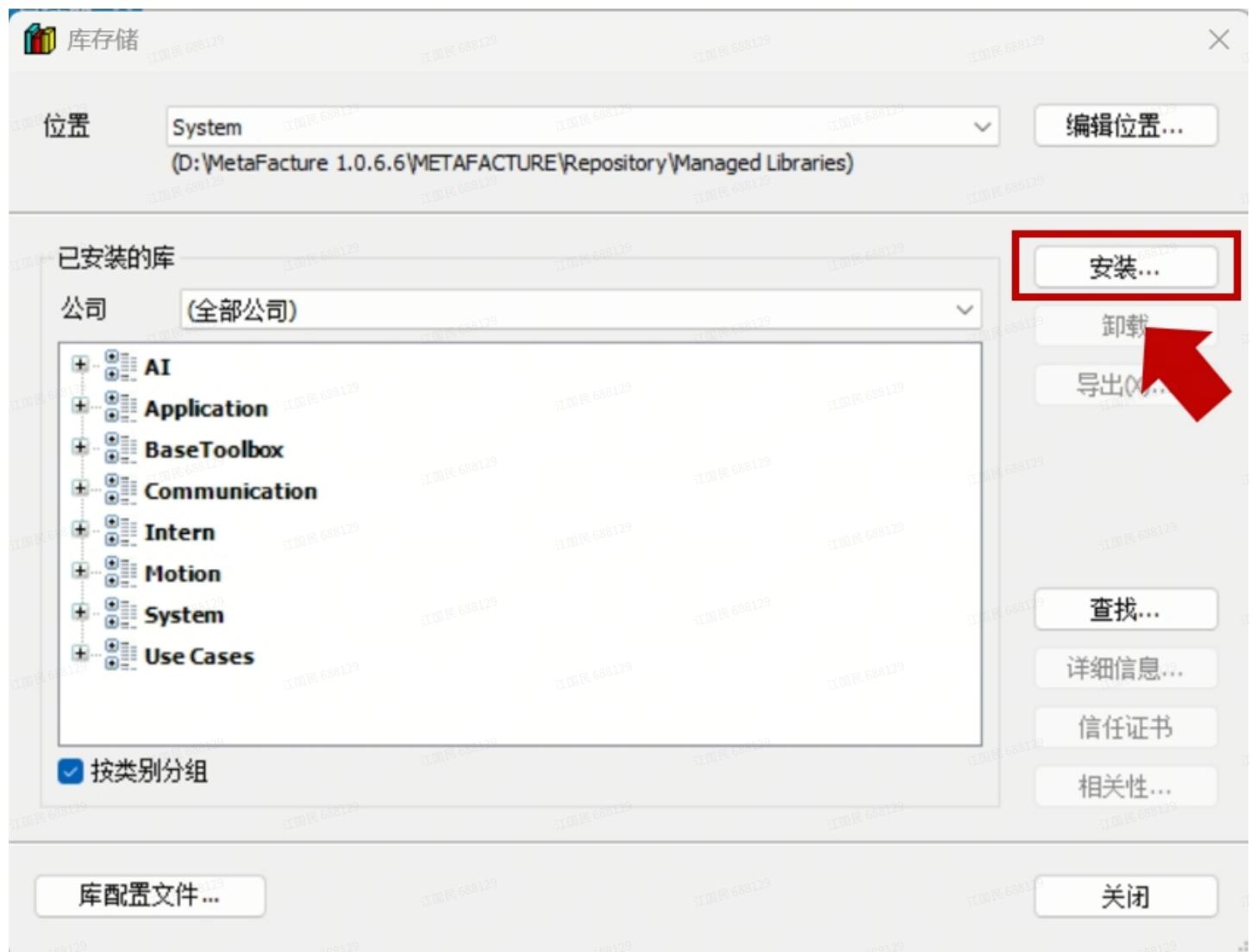
必须具备相关的库或模块依赖（如基础 I/O 库）

## 二、安装过程

根据 **ZDSKInterpolator** 版本和使用的操作系统，产品可以通过不同的方式安装，下面将详细介绍。

在 MetaFacture 中打开工程项目。

通过菜单选择“工具”->“库存储”。



点击安装，打开 **ZDSKInterpolator** 库文件。

点击“添加库”



选中 ZDSKInterpolator，点击确定。

## 三、更新安装

### 1、覆盖安装

下载最新版本的 ZDSKInterpolator 库文件。

打开 MetaFacture 工程，在“库管理器”中删除旧版本的库，导入新版本库文件。

保存工程并重新编译。

### 2、版本兼容性验证

检查新版本是否与现有工程兼容（例如检查参数和接口）。

根据文档调整旧版本与新版本可能的差异。

## 四、卸载过程

### 1、通过库管理器卸载

打开 MetaFacture 项目，进入“库管理器”。

选择 ZDSKInterpolator 库，点击“删除库”卸载库文件。

### 2、清理项目依赖

打开工程项目，检查并移除所有使用 ZDSKInterpolator 库的功能块或接口。

确保工程重新编译时无报错。

## 功能介绍

### 一、XYZ三轴连续插补功能

以功能块启动时的 X、Y、Z 轴位置为起点，已知后续多段连续轨迹坐标，实现从起点到终点的三轴连续插补。

每段插补的速度，加速度，减速度，加加速度可以在指令结构体数组中单独设置，若不设置或输入0则采用默认参数结构体中对应的默认参数。

可选 T形/S 形加减速，及直线间的过渡曲线类型，可在默认参数结构体中设置相应的参数。

## 使用要求

开发环境	目标平台	需要包含的PLC 库
MetaFacture	PC	ZKSDInterpolator

## 二、避障功能

避障功能适用于XYZ直角坐标系机器人设备。

pickbond从pick点到bond点只允许单个长方体障碍物且长宽高与XYZ坐标轴平行。

pick点和bond点按照实际设备伺服读到的位置设置。

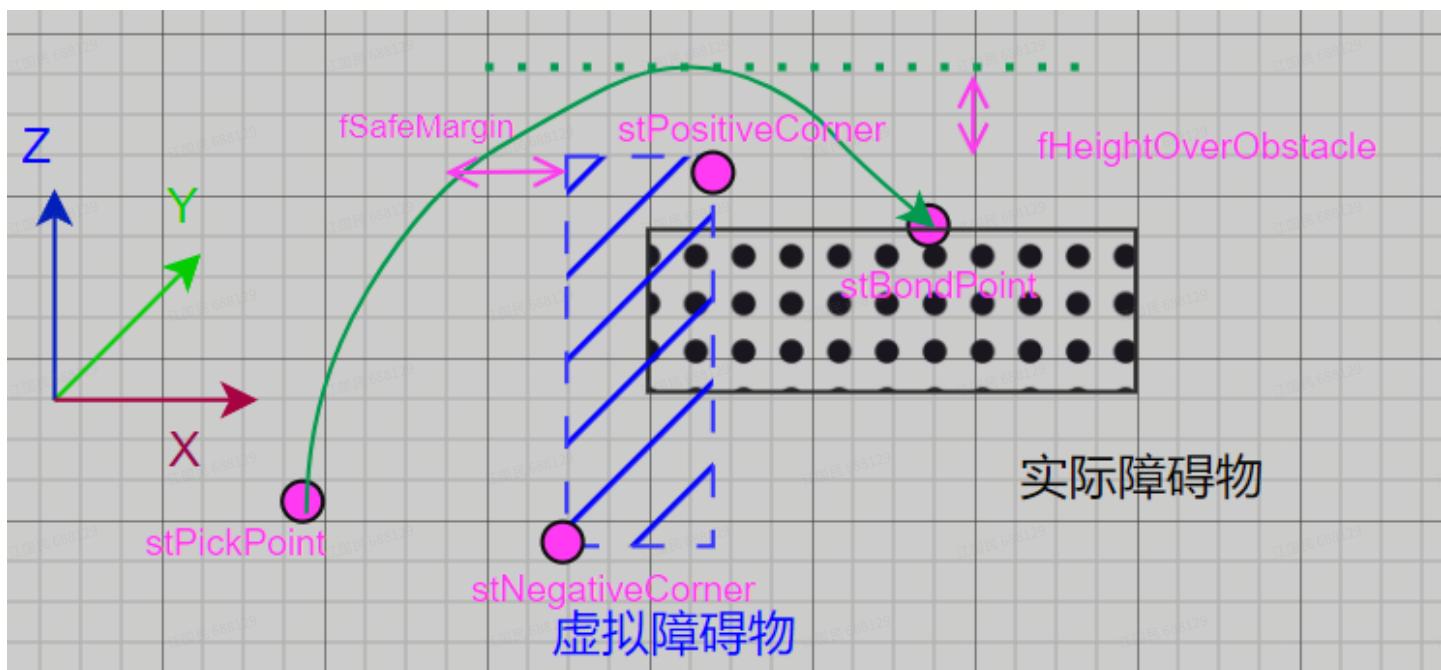
pickbond认为障碍物沿着X轴正方向和负方向都无限长，沿着Z轴负方向无限长，即Z轴值增大才能远离障碍物。

设置虚拟障碍物，两个角点位置，负角点3个坐标值必须分别小于正角点的3个坐标值，以确定障碍物位置和尺寸。

角点坐标值来自实际设备伺服读到的位置来比较，最小的值作为负角点，最大的值作为正角点。

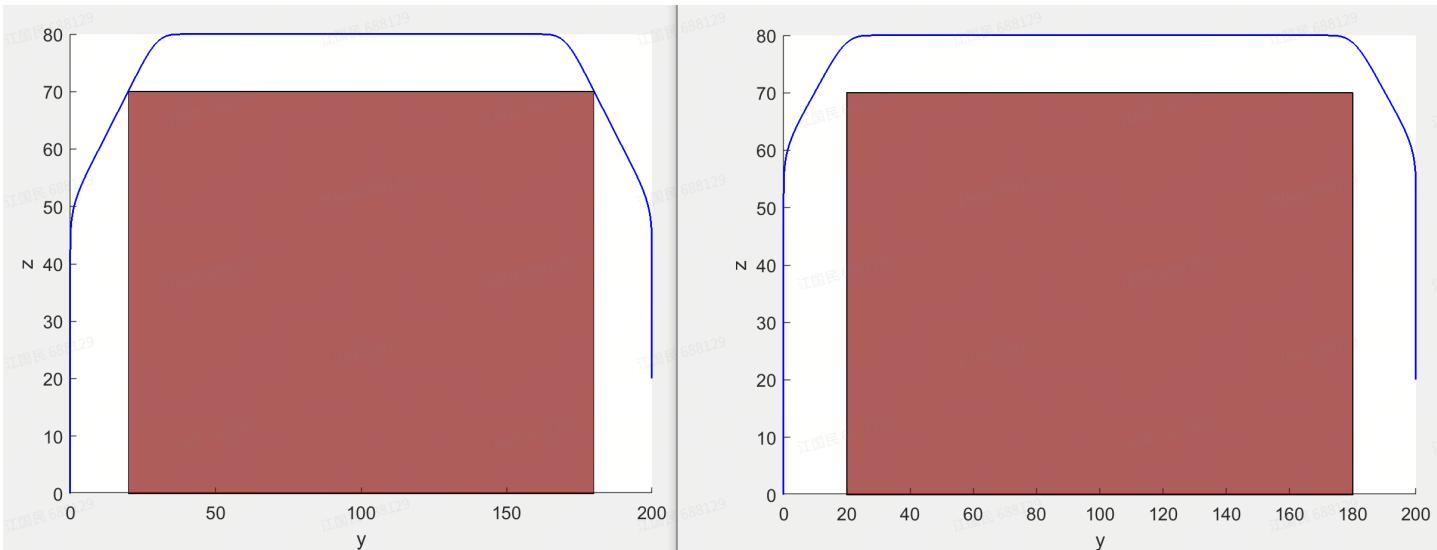
障碍物的上表面高度设置必须高于起pick点和Bond点的高度Z。

障碍物要包裹住关键部分，如图。



高度fHeightOverObstacle，指定Z轴超过障碍物上表面的高度，Z轴向上运动到此高度停止并反向运动。

在其他条件允许时，最优时间路径会与障碍物边缘非常接近，示意图如下左图，变量safe\_margin，作用是远离障碍物，示意图如下右图。



在pick点运动后，顶针轴和Z轴先同步运动一段很小的距离，同步运动结束后，Z轴位置不会超过障碍物上表面加fHeightOverObstacle。随后顶针轴减速到零，并回到设定的起始点。XYZ轴避开障碍物运动到Bond点。

## 使用要求

开发环境	目标平台	需要包含的PLC 库
MetaFacture	PC	ZKSDInterpolator

## 技术说明

### 一、功能块

#### (一) FB\_InterpolatorXYZ

**FB\_InterpolatorXYZ**

```

[ bEnable BOOL := FALSE]
[ bPause BOOL := FALSE]
stCurrentPoint ST_CartesianCoordinate
[ fInterpolationCycleTime LREAL := 0]
pInterpolatorXYZParams POINTER TO ST_InterpolatorXYZParams
[ nInterpolationPointNumber UDINT := 0]

BOOL bBusy
BOOL bDone
BOOL bCommandAborted
BOOL bError
E_ZMCError eErrorID
UDINT nActInterpolationIndex
LREAL fActInterpolationPercent
ST_InterpolatorXYZProfile Profile

```

XYZ 三轴连续插补功能块 **FB\_CTRL\_InterpolatorXYZ** 具有多个输入和输出，具体描述如下。所有控制器的参数均通过结构体传递给功能块。

## 接口

### VAR\_INPUT

1	VAR_INPUT	
2	bEnable	: BOOL;

```

江国民 688129 3 bPause : BOOL;
江国民 688129 4 stCurrentPoint : ST_CartesianCoordinate;
江国民 688129 5 fInterpolationCycleTime : LREAL;
江国民 688129 6 pInterpolationXYZParams : POINTER TO ST_InterpolatorXYZParams;
江国民 688129 7 nInterpolationPointNumber : UDINT;
江国民 688129 8 END_VAR

```

名称	类型	初始化	描述
bEnable	BOOL	FALSE	使能信号，上升沿开始插补
bPause	BOOL	FALSE	暂停信号
stCurrentPoint	ST_CartesianCoordinate		读到的当前位置，作为运动的起始点
fInterpolationCycleTi me	LREAL	0	插补周期时间[s]
pInterpolationXYZPara ms	POINTER TO ST_InterpolatorXYZPara ms		插补参数
nInterpolationPointNu mber	UDINT	0	插补点数，最大9个点。第一个点为起点

## VAR\_OUTPUT

```

江国民 688129 1 VAR_OUTPUT
江国民 688129 2 bBusy : BOOL;
江国民 688129 3 bDone : BOOL;
江国民 688129 4 bCommandAborted : BOOL;
江国民 688129 5 bError : BOOL;
江国民 688129 6 bErrorID : E_ZMCError ;
江国民 688129 7 nInterpolatorIndex : UDINT;
江国民 688129 8 fActInterpolatorPercent : LREAL;
江国民 688129 9 Profile : ST_InterpolatorXYZProfile
江国民 688129 10 END_VAR

```

名称	类型	初始化	描述
bBusy	BOOL	FALSE	插补计算中
bDone	BOOL	FALSE	插补计算完成

bCommandAborted	BOOL	FALSE	插补过程终止
bError	BOOL	FALSE	出错，错误码
bErrorID	E_ZMCError		统一错误码
nInterpolatorIndex	UDINT		当前插补序号
fActInterpolatorPercent	LREAL		当前插补百分比
Profile	ST_InterpolatorXYZProfile		插补输出

## 使用要求

开发环境	目标平台	需要包含的PLC库
MetaFacture/Codesys	PC or CX	ZKSDInterpolator

## (二) 结构体定义

## ST\_CartesianCoordinate

名称	类型	初始化	描述
fX	LREAL	0.0	[mm]
fY	LREAL	0.0	[mm]
fZ	LREAL	0.0	[mm]

fA	LREAL	0.0	[deg]
fB	LREAL	0.0	[deg]
fC	LREAL	0.0	[deg]

## ST\_InterpolatorXYZParams

```

1  TYPE ST_InterpolatorXYZParams :
2  STRUCT
3      nPatType          : USINT;
4      nVelocityType     : USINT;
5      stKinematicsParams: ST_KinematicsParams;
6      stTargetPoint      : ST_CartesianCoordinate;
7      fCirdeArcRadius    : LREAL;
8      stMiddlePoint       : ST_CartesianCoordinate;
9      nSmoothType        : USINT;
10     fSmoothRadius      : LREAL;
11     nSmoothAccWay      : USINT;
12 END_STRUCT
13 END_TYPE

```

名称	类型	初始化	描述
nPatType	USINT	0	指令类型：1=直线绝对运动；2=三点圆弧（中间点+末端点）；3=XY平面顺圆弧（末端点+半径）；4=XY平面逆圆弧（末端点+半径）；5=直线相对运动
nVelocityType	USINT	1	速度曲线类型：预留参数，未使用
stKinematicsParams	ST_KinematicsParams		运动学参数
stTargetPoint	ST_CartesianCoordinate		目标点
fCirdeArcRadius	LREAL	0	当类型是XY平面顺/逆圆弧
stMiddlePoint	ST_CartesianCoordinate		圆弧中间点：预留参数，未使用
nSmoothType	USINT	1	平滑类型：预留参数，未使用
fSmoothRadius	LREAL	0	过渡平滑半径预留参数，未使用

nSmoothAccWa y	USINT	0	平滑过渡曲线加速度采用对应指令加速度：预留参数， 未使用
-------------------	-------	---	---------------------------------

## ST\_InterpolatorXYZProfile

```

1  TYPE ST_InterpolatorXYZProfile :
2  STRUCT
3      X : ST_InterpolatorXYZProfileData;
4      Y : ST_InterpolatorXYZProfileData;
5      Z : ST_InterpolatorXYZProfileData;
6  END_STRUCT
7  END_TYPE

```

名称	类型
X	ST_InterpolatorXYZProfileData
Y	ST_InterpolatorXYZProfileData
Z	ST_InterpolatorXYZProfileData

## ST\_InterpolatorXYZProfileData

```

1  TYPE ST_InterpolatorXYZProfileData :
2  STRUCT
3      fPosition : LREAL;
4      fVelocity : LREAL;
5      fAcceleration : LREAL;
6      fJerk : LREAL;
7  END_STRUCT
8  END_TYPE

```

名称	类型
fPosition	LREAL
fVelocity	LREAL
fAcceleration	LREAL
fJerk	LREAL

### (三) FB\_PickBond



避障功能块 **FB\_PickBond** 具有多个输入和输出，具体描述如下。所有控制器的参数均通过结构体传递给功能块。

#### 接口

##### VAR\_INPUT

```
1 VAR_INPUT
2   bEnable           : BOOL;
3   bPause            : BOOL;
4   stPickBondParams : ST_PickBondParams
5   fInterpolationCycleTime : LREAL;
6 END_VAR
```

名称	类型	初始化	描述
bEnable	BOOL	FALSE	使能信号，上升沿开始插补，下降沿所有信息复位
bPause	BOOL	FALSE	暂停信号
stPickBondParams	ST_PickBondParams		PickBond参数
fInterpolationCycleTime	LREAL	0	插补周期时间[s]

##### VAR\_OUTPUT

```
1 VAR_OUTPUT
2   bBusy             : BOOL;
3   bDone              : BOOL;
4   bCommandAborted   : BOOL;
5   bError             : BOOL;
```

```

江国民 688129 6 bErrorID : E_ZMCError ;
江国民 688129 7 stOverallInfo : ST_InterpolatorOverallInfomation;
江国民 688129 8 stProfile : ST_InterpolatorXYZABCProfile
江国民 688129 9 END_VAR

```

名称	类型	初始化	描述
bBusy	BOOL	FALSE	插补计算中
bDone	BOOL	FALSE	插补计算完成
bCommandAborted	BOOL	FALSE	插补过程终止
bError	BOOL	FALSE	出错，错误码
bErrorID	E_ZMCError		统一错误码
stOverallInfo	ST_InterpolatorOverallInfomation		路径段
stProfile	ST_InterpolatorXYZABCProfile		插补输出

## 使用要求

开发环境	目标平台	需要包含的PLC库
MetaFacture	PC	ZKSDInterpolator

## (四) 结构体定义

### ST\_PickBondParams

```

江国民 688129 1 TYPE ST_PickBondParams :
江国民 688129 2 STRUCT
江国民 688129 3     stPickBondEjector : ST_PickBondEjector;
江国民 688129 4     ePickBondProcess : E_PickBondPhase;
江国民 688129 5     stObstacleCuboid : ST_ObstacleCuboid;
江国民 688129 6     stCoordinateFrame : ST_CoordinatePositionOrientationInGlobalFrame;
江国民 688129 7     stKinematicsParams : ST_KinematicsParams;
江国民 688129 8     stPickPoint : ST_CartesianCoordinate;
江国民 688129 9     stBondPoint : ST_CartesianCoordinate;
江国民 688129 10    fHeightOverObstacle : LREAL;
江国民 688129 11    fSaveMargin : LREAL;
江国民 688129 12 END_STRUCT
江国民 688129 13 END_TYPE

```

名称	类型	描述
stPickBondEjecto r	ST_PickBondEjector	顶针机构参数
ePickBondProces s	E_PickBondPhase	PickBond过程
stObstacleCuboid	ST_ObstacleCuboid	障碍物坐标
stCoordinateFram e	ST_CoordinatePositionOrientationInGlob alFrame	原点在全局坐标系中的位置 以及各轴是否 反向
stKinematicsPara ms	ST_KinematicsParams	运动学参数
stPickPoint	ST_CartesianCoordinate	起点坐标
stBondPoint	ST_CartesianCoordinate	终点坐标
fHeightOverObsta cle	LREAL	避障高度
fSaveMargin	LREAL	安全距离

## ST\_InterpolatorOverallInfomation

```

1  TYPE ST_InterpolatorOverallInfomation :
2  STRUCT
3      fDuration           : LREAL;
4      ePickBondProcess    : E_PickBondPhase;
5      eEjectPhase         : E_PickBondEjectPhase;
6      eZAxisPhase         : E_PickBondZAxisPhase;
7  END_STRUCT
8  END_TYPE

```

名称	类型	描述
fDuration	LREAL	[s]
ePickBondProces s	E_PickBondPhase	PickBond过程

eEjectPhase	E_PickBondEjectPhase	江国民 688129	顶针运行阶段	江国民 688129
eZAxisPhase	E_PickBondZAxisPhase	江国民 688129	Z轴运行阶段	江国民 688129

## ST\_InterpolatorXYZABCProfile

```

1  TYPE ST_InterpolatorXYZABCProfile :
2  STRUCT
3      X : ST_InterpolatorXYZProfileData;
4      Y : ST_InterpolatorXYZProfileData;
5      Z : ST_InterpolatorXYZProfileData;
6      A : ST_InterpolatorXYZProfileData;
7      B : ST_InterpolatorXYZProfileData;
8      C : ST_InterpolatorXYZProfileData;
9  END_STRUCT
10 END_TYPE

```

名称	类型
X	ST_InterpolatorXYZProfileData
Y	ST_InterpolatorXYZProfileData
Z	ST_InterpolatorXYZProfileData
A	ST_InterpolatorXYZProfileData
B	ST_InterpolatorXYZProfileData
C	ST_InterpolatorXYZProfileData

## ST\_PickBondEjector

```

1  TYPE ST_PickBondEjector :
2  STRUCT
3      fCAMMinPosition : LREAL;
4      fCAMMaxPosition : LREAL;
5      fCAMMinAngle : LREAL;
6      fCAMMaxAngle : LREAL;
7      fEjectSpeed : LREAL;
8      fEjectAcceleration : LREAL;
9      fEjectDeceleration : LREAL;
10     fEjectJerk : LREAL;

```

```

11     fBlueMembranePosition      : LREAL;
12     fEjectSyncDistance        : LREAL;
13 END_STRUCT
14 END_TYPE

```

名称	类型	初始化	描述
fCAMMinPosition	LREAL	0.0	顶针凸轮最小位置[mm]
fCAMMaxPosition	LREAL	0.0	顶针凸轮最大位置[mm]
fCAMMinAngle	LREAL	0.0	顶针凸轮最小角度[deg]
fCAMMaxAngle	LREAL	0.0	顶针凸轮最大角度[deg]
fEjectSpeed	LREAL	0.0	顶针运动速度[mm/s]
fEjectAcceleration	LREAL	0.0	顶针运动加速度[mm/s <sup>2</sup> ]
fEjectDeceleration	LREAL	0.0	顶针运动减速度[mm/s <sup>2</sup> ]
fEjectJerk	LREAL	0.0	顶针运动加加速度[mm/s <sup>3</sup> ]
fBlueMembranePosition	LREAL	0.0	在顶针坐标系中蓝膜的绝对位置[mm]
fEjectSyncDistance	LREAL	0.0	同步顶出距离[mm]

## ST\_ObstacleCuboid

```

1  TYPE ST_ObstacleCuboid :
2  STRUCT
3      stNegativeCorner       : ST_CartesianCoordinate;
4      stPositiveCorner        : ST_CartesianCoordinate;
5  END_STRUCT
6  END_TYPE

```

名称	类型	描述
stNegativeCorner	ST_CartesianCoordinate	负向角点坐标
stPositiveCorner	ST_CartesianCoordinate	正向角点坐标

## ST\_CoordinatePositionOrientationInGlobalFrame

```
1  TYPE ST_CoordinatePositionOrientationInGlobalFrame :  
2  STRUCT  
3      fXOffset          : LREAL;  
4      fYOffset          : LREAL;  
5      fZOffset          : LREAL;  
6      fAOffset           : LREAL;  
7      fBOffset           : LREAL;  
8      fCOffset           : LREAL;  
9      bXReverse         : BOOL;  
10     bYReverse         : BOOL;  
11     bZReverse         : BOOL;  
12     bAReserve         : BOOL;  
13     bBReverse         : BOOL;  
14     bCReverse         : BOOL;  
15 END_STRUCT  
16 END_TYPE
```

名称	类型	初始化	描述
fXOffset	LREAL	0.0	坐标系相对机器人坐标系偏移[mm]
fYOffset	LREAL	0.0	坐标系相对机器人坐标系偏移[mm]
fZOffset	LREAL	0.0	坐标系相对机器人坐标系偏移[mm]
fAOffset	LREAL	0.0	坐标系相对机器人坐标系偏移[mm]
fBOffset	LREAL	0.0	坐标系相对机器人坐标系偏移[mm]
fCOffset	LREAL	0.0	坐标系相对机器人坐标系偏移[mm]
bXReverse	BOOL	FALSE	TRUE or FALSE
bYReverse	BOOL	FALSE	TRUE or FALSE
bZReverse	BOOL	FALSE	TRUE or FALSE
bAReserve	BOOL	FALSE	TRUE or FALSE
bBReverse	BOOL	FALSE	TRUE or FALSE
bCReverse	BOOL	FALSE	TRUE or FALSE

## ST\_KinematicsParams

```
1  TYPE ST_KinematicsParams :  
2  STRUCT  
3      X           : ST_KinematicsParamsAxis;  
4      Y           : ST_KinematicsParamsAxis;  
5      Z           : ST_KinematicsParamsAxis;  
6      A           : ST_KinematicsParamsAxis;  
7      B           : ST_KinematicsParamsAxis;  
8      C           : ST_KinematicsParamsAxis;  
9      stCartesianTranslation : ST_KinematicsParamsAxis;  
10     stCartesianRotation   : ST_KinematicsParamsAxis;  
11     eRampType          : E_ZMCProfileType;  
12 END_STRUCT  
13 END_TYPE
```

名称	类型	描述
X	ST_KinematicsParamsAxis	轴坐标系下运动时使用
Y	ST_KinematicsParamsAxis	轴坐标系下运动时使用
Z	ST_KinematicsParamsAxis	轴坐标系下运动时使用
A	ST_KinematicsParamsAxis	轴坐标系下运动时使用
B	ST_KinematicsParamsAxis	轴坐标系下运动时使用
C	ST_KinematicsParamsAxis	轴坐标系下运动时使用
stCartesianTranslation	ST_KinematicsParamsAxis	笛卡尔中位置运动时使用
stCartesianRotation	ST_KinematicsParamsAxis	笛卡尔中姿态旋转时使用
eRampType	E_ZMCProfileType	速度曲线类型：预留参数，未使用

## ST\_KinematicsParamsAxis

```
1  TYPE ST_KinematicsParamsAxis :  
2  STRUCT  
3      fVelocity       : LREAL;  
4      fAcceleration   : LREAL;
```

```

5     fDeceleration      : LREAL;
6     fJerk                : LREAL;
7 END_STRUCT
8 END_TYPE

```

名称	类型	初始化	描述
fVelocity	LREAL	0	速度[mm/s]
fAcceleration	LREAL	0	加速度[mm/s <sup>2</sup> ]
fDeceleration	LREAL	0	减速度[mm/s <sup>2</sup> ]
fJerk	LREAL	0	加加速度[mm/s <sup>3</sup> ]

## ST\_CartesianCoordinate

见本节 [\(二\) ST\\_CartesianCoordinate](#), 此处不详述

## E\_PickBondPhase

```

1   TYPE ST_PickBondPhase :
2   STRUCT
3     PICK_TO_BOND        : INT;
4     BOND_TO_PICK        : INT;
5   END_STRUCT
6   END_TYPE

```

名称	类型	初始化	描述
PICK_TO_BOND	INT	1	从Pick到Bond
BOND_TO_PICK	INT	2	从Bond到Pick

## E\_PickBondEjectPhase

```

1   TYPE ST_PickBondPhase :
2   STRUCT
3     MOVE_TO_PREEJECT_POSITION    : INT;
4     IN_PREEJECT_POSITION         : INT;
5     MOVE_TO_DIE_BOTTOM_SURFACE   : INT;
6     IN_DIE_BOTTOM_SURFACE        : INT;

```

```

7     EJECTOR_SYNC_LIFT_UP      : INT;
8     EJECTOR_DECELERATION    : INT;
9     IN_STOP_POSITION        : INT;
10    END_STRUCT
11    END_TYPE

```

名称	类型	初始化	描述
MOVE_TO_PREEJECT_POSITION	INT	0	运动到预顶针位置
IN_PREEJECT_POSITION	INT	1	在预顶针位置
MOVE_TO_DIE_BOTTOM_SURFACE	INT	2	运动到芯片下表面
IN_DIE_BOTTOM_SURFACE	INT	3	在芯片下表面
EJECTOR_SYNC_LIFT_UP	INT	4	顶针同步上升
EJECTOR_DECELERATION	INT	5	顶针减速停止
IN_STOP_POSITION	INT	6	到达减速停止位置

## E\_PickBondZAxisPhase

```

1   TYPE ST_PickBondZAxisPhase :
2   STRUCT
3     TO_PICK_MAX_FORCE      : INT;
4     MOVE_TO_DIE_FRONT_SURFACE : INT;
5     IN_DIE_FRONT_SURFACE    : INT;
6     SYNC_LIFT_UP            : INT;
7     TO_BOND_MAX_FORCE      : INT;
8     MOVE_TO_BOND_POSITION   : INT;
9     IN_BOND_POSITION        : INT;
10    END_STRUCT
11    END_TYPE

```

名称	类型	初始化	描述
TO_PICK_MAX_FORCE	INT	1	
MOVE_TO_DIE_FRONT_SURFACE	INT	2	运动到芯片上表面

IN_DIE_FRONT_SURFACE	INT	3	在芯片上表面	江国民 688129
SYNC_LIFT_UP	INT	4	和顶针同步提升阶段	江国民 688129
TO_BOND_MAX_FORCE	INT	5		江国民 688129
MOVE_TO_BOND_POSITION	INT	6	运动到Bond位置	江国民 688129
IN_BOND_POSITION	INT	7	在Bond位置	江国民 688129

## 二、示例代码

以下示例代码用于插补器的测试。运行代码后，用户可以通过网页访问可视化界面，在虚拟环境中对轴运动进行测试和验证。

```

1  IF NOT bInitCmd THEN
2
3      bInitCmd := TRUE;
4      stXYZInterpolatorParams[0].nPathType
5          := 1;
6      stXYZInterpolatorParams[0].fCircleArcRadius
7          := 0;
8      stXYZInterpolatorParams[0].stMiddlePoint.fX
9          := 20;
10     stXYZInterpolatorParams[0].stMiddlePoint.fY
11     := 20;
12     stXYZInterpolatorParams[0].stMiddlePoint.fZ
13     := 0;
14     stXYZInterpolatorParams[0].stTargetPoint.fX
15     := 0;
16     stXYZInterpolatorParams[0].stTargetPoint.fY
17     := 1;
18     stXYZInterpolatorParams[0].stTargetPoint.fZ
19     := 0;
20
21     stXYZInterpolatorParams[0].stKinematicsParams.stCartesianTranslation.fVelocity
22     := 500;
23
24     stXYZInterpolatorParams[0].stKinematicsParams.stCartesianTranslation.fDecelera
25     tion := 1400;
26
27     stXYZInterpolatorParams[0].stKinematicsParams.stCartesianTranslation.fAccelera
28     tion := 1500;
29
30     stXYZInterpolatorParams[0].stKinematicsParams.stCartesianTranslation.fJerk
31     := 5000;

```

```
16         stXYZInterpolatorParams[0].fSmoothRadius  
17             := 20;  
18         stXYZInterpolatorParams[0].nSmoothType  
19             := 1;  
20         stXYZInterpolatorParams[1].nPathType  
21             := 1;  
22         stXYZInterpolatorParams[1].fCircleArcRadius  
23             := 0;  
24         stXYZInterpolatorParams[1].stMiddlePoint.fX  
25             := 0;  
26         stXYZInterpolatorParams[1].stMiddlePoint.fY  
27             := 0;  
28         stXYZInterpolatorParams[1].stMiddlePoint.fZ  
29             := 0;  
30         stXYZInterpolatorParams[1].stTargetPoint.fX  
31             := 0;  
32         stXYZInterpolatorParams[1].stTargetPoint.fY  
33             := 0;  
34         stXYZInterpolatorParams[1].stTargetPoint.fZ  
35             := 0;  
36         stXYZInterpolatorParams[1].stKinematicsParams.stCartesianTranslation.fVelocity  
37             := 500;  
38         stXYZInterpolatorParams[1].stKinematicsParams.stCartesianTranslation.fDecelera  
39             := 1400;  
40         stXYZInterpolatorParams[1].stKinematicsParams.stCartesianTranslation.fAccelera  
41             := 1500;  
42         stXYZInterpolatorParams[1].stKinematicsParams.stCartesianTranslation.fJerk  
43             := 5000;  
44         stXYZInterpolatorParams[1].fSmoothRadius  
45             := 20;  
46         stXYZInterpolatorParams[1].nSmoothType  
47             := 0;  
48         stXYZInterpolatorParams[2].nPathType  
49             := 1;  
50         stXYZInterpolatorParams[2].fCircleArcRadius  
51             := 0;  
52         stXYZInterpolatorParams[2].stMiddlePoint.fX  
53             := 20;  
54         stXYZInterpolatorParams[2].stMiddlePoint.fY  
55             := 20;
```

```
38         stXYZInterpolatorParams[2].stMiddlePoint.fZ  
39             := 0;  
40         stXYZInterpolatorParams[2].stTargetPoint.fX  
41             := 20;  
42         stXYZInterpolatorParams[2].stTargetPoint.fY  
43             := 40;  
44         stXYZInterpolatorParams[2].stTargetPoint.fZ  
45             := 0;  
46  
47     stXYZInterpolatorParams[2].stKinematicsParams.stCartesianTranslation.fVelocity  
48         := 20;  
49     stXYZInterpolatorParams[2].stKinematicsParams.stCartesianTranslation.fDecelera  
50         tion := 60;  
51     stXYZInterpolatorParams[2].stKinematicsParams.stCartesianTranslation.fAccelera  
52         tion := 70;  
53     stXYZInterpolatorParams[2].stKinematicsParams.stCartesianTranslation.fJerk  
54         := 800;  
55     stXYZInterpolatorParams[2].fSmoothRadius  
56         := 20;  
57  
58     stXYZInterpolatorParams[2].nSmoothType  
59         := 1;  
60  
61     stXYZInterpolatorParams[3].nPathType  
62         := 1;  
63     stXYZInterpolatorParams[3].fCircleArcRadius  
64         := 0;  
65     stXYZInterpolatorParams[3].stMiddlePoint.fX  
66         := 0;  
67     stXYZInterpolatorParams[3].stMiddlePoint.fY  
68         := 0;  
69     stXYZInterpolatorParams[3].stMiddlePoint.fZ  
70         := 0;  
71     stXYZInterpolatorParams[3].stTargetPoint.fX  
72         := 40;  
73     stXYZInterpolatorParams[3].stTargetPoint.fY  
74         := 0;  
75     stXYZInterpolatorParams[3].stTargetPoint.fZ  
76         := 0;  
77  
78     stXYZInterpolatorParams[3].stKinematicsParams.stCartesianTranslation.fVelocity  
79         := 50;  
80  
81     stXYZInterpolatorParams[3].stKinematicsParams.stCartesianTranslation.fDecelera  
82         tion := 140;
```

```
59     stXYZInterpolatorParams[3].stKinematicsParams.stCartesianTranslation.fAcceleration      := 150;
60
61     stXYZInterpolatorParams[3].stKinematicsParams.stCartesianTranslation.fJerk          := 500;
62     stXYZInterpolatorParams[3].fSmoothRadius
63             := 0;
64     stXYZInterpolatorParams[3].nSmoothType
65             := 1;
66
67     stXYZInterpolatorParams[4].nPathType
68             := 2;
69     stXYZInterpolatorParams[4].fcircleArcRadius
70             := 0;
71     stXYZInterpolatorParams[4].stMiddlePoint.fx
72             := 80;
73     stXYZInterpolatorParams[4].stMiddlePoint.fY
74             := 0;
75     stXYZInterpolatorParams[4].stMiddlePoint.fZ
76             := 0;
77
78     stXYZInterpolatorParams[4].stKinematicsParams.stCartesianTranslation.fVelocity
79             := 50;
80
81     stXYZInterpolatorParams[4].stKinematicsParams.stCartesianTranslation.fDeceleration
82             := 140;
83
84     stXYZInterpolatorParams[4].stKinematicsParams.stCartesianTranslation.fAcceleration
85             := 150;
86
87     stXYZInterpolatorParams[4].stKinematicsParams.stCartesianTranslation.fJerk
88             := 500;
89     stXYZInterpolatorParams[4].fSmoothRadius
90             := 0;
91     stXYZInterpolatorParams[4].nSmoothType
92             := 1;
93
94     stXYZInterpolatorParams[5].nPathType
95             := 4;
```

```
80         stXYZInterpolatorParams[5].fCircleArcRadius  
81             := 10;  
82         // stXYZInterpolatorParams[5].stMiddlePoint.fx  
83             := 80;  
84         // stXYZInterpolatorParams[5].stMiddlePoint.fY  
85             := 0;  
86         // stXYZInterpolatorParams[5].stMiddlePoint.fZ  
87             := 0;  
88         stXYZInterpolatorParams[5].stKinematicsParams.stCartesianTranslation.fVelocity  
89             := 50;  
90         stXYZInterpolatorParams[5].stKinematicsParams.stCartesianTranslation.fDecelera  
91             tion := 140;  
92         stXYZInterpolatorParams[5].stKinematicsParams.stCartesianTranslation.fAccelera  
93             tion := 150;  
94         stXYZInterpolatorParams[5].stKinematicsParams.stCartesianTranslation.fJerk  
95             := 500;  
96         stXYZInterpolatorParams[5].fSmoothRadius  
97             := 0;  
98         stXYZInterpolatorParams[5].nSmoothType  
99             := 1;  
100        stXYZInterpolatorParams[6].nPathType  
101        := 3;  
102        stXYZInterpolatorParams[6].fCircleArcRadius  
103        := 10;  
104        stXYZInterpolatorParams[6].stMiddlePoint.fx  
105        := 60;  
106        stXYZInterpolatorParams[6].stMiddlePoint.fY  
107        := 20;  
108        stXYZInterpolatorParams[6].stMiddlePoint.fZ  
109        := 0;  
110        stXYZInterpolatorParams[6].stTargetPoint.fx  
111        := 40;  
112        stXYZInterpolatorParams[6].stTargetPoint.fY  
113        := 0;  
114        stXYZInterpolatorParams[6].stTargetPoint.fZ  
115        := 0;
```

```
102     stXYZInterpolatorParams[6].stKinematicsParams.stCartesianTranslation.fVelocity
           := 50;
103
104     stXYZInterpolatorParams[6].stKinematicsParams.stCartesianTranslation.fDecelera
           tion      := 140;
105
106     stXYZInterpolatorParams[6].stKinematicsParams.stCartesianTranslation.fAccelera
           tion      := 150;
107
108     stXYZInterpolatorParams[6].stKinematicsParams.stCartesianTranslation.fJerk
           := 500;
109
110    stXYZInterpolatorParams[6].fSmoothRadius
           := 0;
111    stXYZInterpolatorParams[6].nSmoothType
           := 1;
112
113    stXYZInterpolatorParams[7].nPathType
           := 5;
114    stXYZInterpolatorParams[7].fCircleArcRadius
           := 0;
115    stXYZInterpolatorParams[7].stMiddlePoint.fX
           := 60;
116    stXYZInterpolatorParams[7].stMiddlePoint.fY
           := 20;
117    stXYZInterpolatorParams[7].stMiddlePoint.fZ
           := 0;
118    stXYZInterpolatorParams[7].stTargetPoint.fX
           := 0;
119    stXYZInterpolatorParams[7].stTargetPoint.fY
           := -20;
120    stXYZInterpolatorParams[7].stTargetPoint.fZ
           := 0;
121
122    stXYZInterpolatorParams[7].stKinematicsParams.stCartesianTranslation.fVelocity
           := 50;
123
124    stXYZInterpolatorParams[7].stKinematicsParams.stCartesianTranslation.fDecelera
           tion      := 140;
125
126    stXYZInterpolatorParams[7].stKinematicsParams.stCartesianTranslation.fAccelera
           tion      := 150;
127
128    stXYZInterpolatorParams[7].stKinematicsParams.stCartesianTranslation.fJerk
           := 500;
129
130    stXYZInterpolatorParams[7].fSmoothRadius
           := 0;
```

```
江国民 122     stXYZInterpolatorParams[7].nSmoothType  
江国民 122         := 1;  
江国民 123  
江国民 124         // stCurrentPoint    := (fX:=IoConfig_Globals.gAxisX.fActPosition,  
江国民 124             fY:=IoConfig_Globals.gAxisY.fActPosition);  
江国民 125         setPos[0] (  
江国民 126             Axis:= gAxisX,  
江国民 127             Execute:= TRUE,  
江国民 128             Position:= -6,  
江国民 129             Mode:= ,  
江国民 130             Done=> ,  
江国民 131             Busy=> ,  
江国民 132             Error=> ,  
江国民 133             ErrorID=> );  
江国民 134  
江国民 135         setPos[1] (  
江国民 136             Axis:= gAxisY,  
江国民 137             Execute:= TRUE,  
江国民 138             Position:= -2,  
江国民 139             Mode:= ,  
江国民 140             Done=> ,  
江国民 141             Busy=> ,  
江国民 142             Error=> ,  
江国民 143             ErrorID=> );  
江国民 144  
江国民 145         setPos[2] (  
江国民 146             Axis:= gAxisZ,  
江国民 147             Execute:= TRUE,  
江国民 148             Position:= 0,  
江国民 149             Mode:= ,  
江国民 150             Done=> ,  
江国民 151             Busy=> ,  
江国民 152             Error=> ,  
江国民 153             ErrorID=> );  
江国民 154  
江国民 155  
江国民 156         stCurrentPoint.fX := IoConfig_Globals.gAxisX.fActPosition;  
江国民 157         stCurrentPoint.fY := IoConfig_Globals.gAxisY.fActPosition;  
江国民 158         stCurrentPoint.fZ := IoConfig_Globals.gAxisZ.fActPosition;  
江国民 159  
江国民 160     END_IF  
江国民 161  
江国民 162  
江国民 163     fbZMCInterpolatorXYZ(江国民 163  
江国民 164         pAxisX江国民 164         := ADR( gAxisX ),  
江国民 165         pAxisY江国民 165         := ADR( gAxisY ),  
江国民 166         pAxisZ江国民 166         := ADR( gAxisZ ),
```

```

167         stCurrentPoint    := stCurrentPoint,
168         pInterpolatorXYZParams
169         stXYZInterpolatorParams ),
170         nInterpolatorDataNumber
171         fInterpolationCycleTime
172         bExecute
173         bPause
174         bBusy=> ,
175         bError=> ,
176         bDone=> ,
177         eErrorID=> );
178
179 IF fbZMCInterpolatorXYZ.bDone THEN
180     bStart := FALSE;
181 END_IF
182 IF NOT bStart THEN
183     bPause := FALSE;
184 END_IF
185 fbTONStart( IN := NOT bStart, PT := T#500MS );
186 IF fbTONStart.Q AND ( fbMCReadStatus[0].StandStill AND
187 gHmiPlcIF.Output.Monitor.Axis[0].bIsHomed ) AND (
188 fbMCReadStatus[1].StandStill AND gHmiPlcIF.Output.Monitor.Axis[1].bIsHomed )
189 AND
190     ( fbMCReadStatus[2].StandStill AND
191 gHmiPlcIF.Output.Monitor.Axis[2].bIsHomed ) THEN
192     bStart := TRUE;
193 END_IF
194
195 (* 轴控 *)
196 gHmiPlcIF.Input.pAxis[0] := ADR( gAxisX );
197 gHmiPlcIF.Input.pAxis[1] := ADR( gAxisY );
198 gHmiPlcIF.Input.pAxis[2] := ADR( gAxisZ );
199 FOR nAxisIndex := 0 TO 2 BY 1 DO
200     fbMCPower[nAxisIndex](
201         Axis          := gHmiPlcIF.Input.pAxis[nAxisIndex]^,
202         Enable        := TRUE,
203         bRegulatorOn:= TRUE,
204         bDriveStart   := gHmiPlcIF.Input.Command.bPower[nAxisIndex],
205         Status=> ,
206         bRegulatorRealState=> ,
207         bDriveStartRealState=> ,
208         Busy=> ,
209         Error=> ,
210         ErrorID=> );

```

```

208     fbMCHome[nAxisIndex] (
209         Axis := gHmiPlcIF.Input.pAxis[nAxisIndex]^,
210         Execute := gHmiPlcIF.Input.Command.bHome[nAxisIndex],
211         Position := 0,
212         Done=> ,
213         Busy=> ,
214         CommandAborted=> ,
215         Error=> ,
216         ErrorID=> );
217     IF fbMCHome[nAxisIndex].Done OR fbMCHome[nAxisIndex].Error THEN
218         gHmiPlcIF.Input.Command.bHome[nAxisIndex] := FALSE;
219         gHmiPlcIF.Output.Monitor.Axis[nAxisIndex].bIsHomed := TRUE;
220     END_IF
221
222     fbMCJog[nAxisIndex] (
223         Axis := gHmiPlcIF.Input.pAxis[nAxisIndex]^,
224         JogForward := gHmiPlcIF.Input.Command.bJogPos[nAxisIndex],
225         JogBackward := gHmiPlcIF.Input.Command.bJogNeg[nAxisIndex],
226         Velocity := 1000,
227         Acceleration := 15000,
228         Deceleration := 15000,
229         Jerk := 150000,
230         Busy=> ,
231         CommandAborted=> ,
232         Error=> ,
233         ErrorId=> );
234
235     fbMCMoveAbsolute[nAxisIndex] (
236         Axis := gHmiPlcIF.Input.pAxis[nAxisIndex]^,
237         Execute := gHmiPlcIF.Input.Command.bGo2Zero[nAxisIndex],
238         Position := 0,
239         Velocity := 1000,
240         Acceleration := 15000,
241         Deceleration := 15000,
242         Jerk := 150000,
243         Direction := MC_Direction.shortest,
244         BufferMode := MC_BUFFER_MODE.Aborting,
245         Done=> ,
246         Busy=> ,
247         Active=> ,
248         CommandAborted=> ,
249         Error=> ,
250         ErrorID=> );

```

```
251      IF fbMCMoveAbsolute[nAxisIndex].Done OR  
252          fbMCMoveAbsolute[nAxisIndex].Error THEN  
253              gHmiPlcIF.Input.Command.bGo2Zero[nAxisIndex] := FALSE;  
254          END_IF  
255      fbMCStop[nAxisIndex] (  
256          Axis           := gHmiPlcIF.Input.pAxis[nAxisIndex]^,  
257          Execute        :=  
258          gHmiPlcIF.Input.Command.bStop[nAxisIndex],  
259          Deceleration   := 1000,  
260          Jerk          := 10000,  
261          Done          => ,  
262          Busy          => ,  
263          Error         => ,  
264          ErrorID       => );  
265      IF fbMCStop[nAxisIndex].Done OR fbMCStop[nAxisIndex].Error THEN  
266          gHmiPlcIF.Input.Command.bStop[nAxisIndex] := FALSE;  
267          END_IF  
268      fbMCReset[nAxisIndex] (  
269          Axis           := gHmiPlcIF.Input.pAxis[nAxisIndex]^,  
270          Execute        := gHmiPlcIF.Input.Command.bReset[nAxisIndex],  
271          Done=> ,  
272          Busy=> ,  
273          Error=> ,  
274          ErrorID=> );  
275  
276      fbMCReadStatus[nAxisIndex] (  
277          Axis           := gHmiPlcIF.Input.pAxis[nAxisIndex]^,  
278          Enable         := TRUE,  
279          Valid=> ,  
280          Busy=> ,  
281          Error=> ,  
282          ErrorID=> ,  
283          Disabled=> ,  
284          Errorstop=> ,  
285          Stopping=> ,  
286          StandStill=> ,  
287          DiscreteMotion=> ,  
288          ContinuousMotion=> ,  
289          SynchronizedMotion=> ,  
290          Homing=> ,  
291          ConstantVelocity=> ,  
292          Accelerating=> ,  
293          Decelerating=> ,  
294          FBErrorOccured=> );
```

```
295
296
297     fbMCReadStatus[nAxisIndex] (
298         Axis          := gHmiPlcIF.Input.pAxis[nAxisIndex]^,
299         Enable        := TRUE,
300         Valid=> ,
301         Busy=> ,
302         Error=> ,
303         ErrorID=> ,
304         Disabled=> ,
305         Errorstop=> ,
306         Stopping=> ,
307         StandStill=> ,
308         DiscreteMotion=> ,
309         ContinuousMotion=> ,
310         SynchronizedMotion=> ,
311         Homing=> ,
312         ConstantVelocity=> ,
313         Accelerating=> ,
314         Decelerating=> ,
315         FBErrorOccured=> );
316
317 (* Monitor *)
318 gHmiPlcIF.Output.Monitor.Axis[nAxisIndex].bError      :=
319     fbMCReadStatus[nAxisIndex].Error;
320 gHmiPlcIF.Output.Monitor.Axis[nAxisIndex].nErrorID      :=
321     fbMCReadStatus[nAxisIndex].ErrorID;
322 gHmiPlcIF.Output.Monitor.Axis[nAxisIndex].bSynchronizedMotion      :=
323     fbMCReadStatus[nAxisIndex].SynchronizedMotion;
324 gHmiPlcIF.Output.Monitor.Axis[nAxisIndex].bDiscreteMotion      :=
325     fbMCReadStatus[nAxisIndex].DiscreteMotion;
326 gHmiPlcIF.Output.Monitor.Axis[nAxisIndex].bContinuousMotion      :=
327     fbMCReadStatus[nAxisIndex].ContinuousMotion;
328 gHmiPlcIF.Output.Monitor.Axis[nAxisIndex].bStopping      :=
329     fbMCReadStatus[nAxisIndex].Stopping;
330 gHmiPlcIF.Output.Monitor.Axis[nAxisIndex].bHoming      :=
331     fbMCReadStatus[nAxisIndex].Homing;
332 gHmiPlcIF.Output.Monitor.Axis[nAxisIndex].bStandstill      :=
333     fbMCReadStatus[nAxisIndex].Standstill;
334 gHmiPlcIF.Output.Monitor.Axis[nAxisIndex].bDisabled      :=
335     fbMCReadStatus[nAxisIndex].Disabled;
336 gHmiPlcIF.Output.Monitor.Axis[nAxisIndex].bErrorstop      :=
337     fbMCReadStatus[nAxisIndex].Errorstop;
338 gHmiPlcIF.Output.Monitor.Axis[nAxisIndex].fPosition      :=
339     gHmiPlcIF.Input.pAxis[nAxisIndex]^ . fActPosition;
340 gHmiPlcIF.Output.Monitor.Axis[nAxisIndex].fVelocity      :=
341     gHmiPlcIF.Input.pAxis[nAxisIndex]^ . fActVelocity;
```

```
江国民 330  
331 gHmiPlcIF.Output.Monitor.nErrorNumber := fbZMCInterpolatorXYZ.eErrorID;  
332 END_FOR
```