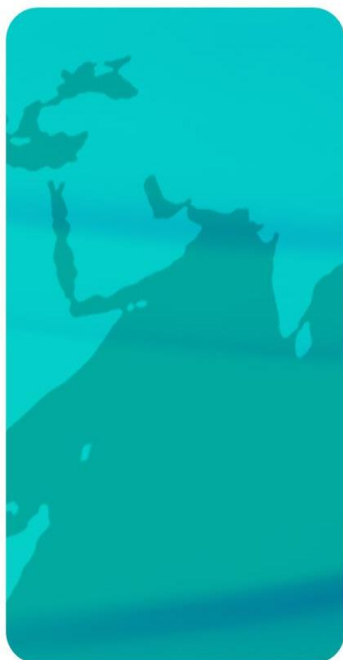


LKS 安全型控制系统指令手册

智能化成就卓越

和利时集团

北京和利时智能技术有限公司



LKS 安全型控制系统 指令手册

V1.0 版

2019 年 06 月

版权声明

本手册内容，包括文字、图表、标志、标识、商标、产品型号、软件程序、版面设计及其它内容等，均受《中华人民共和国著作权法》、《中华人民共和国商标法》、《中华人民共和国专利法》及与之适用的国际公约中有关著作权、商标权、专利权或其他财产所有权法律的保护，为北京和利时智能技术有限公司专属所有或持有。

由于本手册中所描述的设备有多种使用方法，用户以及设备使用责任人必须保证每种方法的可容许性。对由使用或错误使用这些设备造成的任何直接或间接损失，北京和利时智能技术有限公司将不负法律责任。

由于实际应用时的不确定因素，北京和利时智能技术有限公司不承担直接使用本手册中提供的数据的责任。

本手册仅供商业用户阅读，在未得到北京和利时智能技术有限公司书面授权的情况下，无论出于何种目的和原因，不得以任何形式（包括电子、机械或其它形式）传播或复制本手册的任何内容。违者我公司将依法追究其相关责任。

已核对本手册中的内容、图表与所述硬件设备相符，但误差难以避免，并不能保证完全一致。同时，会定期对手册的内容、图表进行检查、修改和维护，恕不另行通知。

HollySys、和利时、 **HollySys** 的字样和徽标均为北京和利时智能技术有限公司的商标或注册商标。

手册中涉及到的其他商标或注册商标属于它们各自的拥有者。

北京和利时智能技术有限公司版权所有。

地址：北京经济技术开发区地盛中路 2 号院

邮编：100176

商务电话：010-5898 1588

产品咨询热线：4008-111-999

技术支持电话：010-58981514

技术支持手机：13611160213

传真：010-5898 1558

网址：<http://www.hollysys.com/>

Email：PLC@hollysys.com

新浪微博：<http://weibo.com/hollysysplc>

目录

第 1 章 关于手册	1
1.1 文档更新	1
1.2 文档用途	1
1.3 阅读对象	1
1.4 使用约定	1
1.4.1 菜单命令	1
1.4.2 鼠标指令	1
1.4.3 键盘指令	1
1.4.4 在线帮助	2
1.4.5 重要信息	2
1.5 产品文档目录	3
第 2 章 库管理器	5
2.1 库指令使用说明	5
2.2 标准库	5
2.2.1 数学运算	5
2.2.1.1 ADD (加)	5
2.2.1.2 SUB (减)	6
2.2.1.3 MUL (乘)	7
2.2.1.4 DIV (除)	7
2.2.1.5 MOD (取余)	8
2.2.1.6 ADD_WORD (加)	9
2.2.1.7 SUB_WORD (减)	9
2.2.1.8 MUL_WORD (乘)	10
2.2.1.9 DIV_WORD (除)	10
2.2.1.10 MOD_WORD (取余)	11
2.2.1.11 ADD_INT (加)	11
2.2.1.12 SUB_INT (减)	12
2.2.1.13 MUL_INT (乘)	12
2.2.1.14 DIV_INT (除)	13
2.2.1.15 MOD_INT (取余)	14
2.2.1.16 DIV_DWORD (除)	14
2.2.1.17 DIV_DINT (除)	15
2.2.1.18 DIV_REAL (除)	16
2.2.1.19 DIV_LREAL (除)	17
2.2.1.20 ABS (绝对值)	18
2.2.1.21 SQRT (平方根)	19
2.2.1.22 LOG (常用对数)	19

2.2.1.23	LN (自然对数)	20
2.2.1.24	EXP (指数)	21
2.2.1.25	SIN (正弦)	21
2.2.1.26	COS (余弦)	22
2.2.1.27	TAN (正切)	23
2.2.1.28	ASIN (反正弦)	24
2.2.1.29	ACOS (反余弦)	24
2.2.1.30	ATAN (反正切)	25
2.2.1.31	EXPT (幂)	26
2.2.1.32	SQRT_R (单精度平方根运算)	27
2.2.1.33	LOG_R (单精度常用对数运算)	27
2.2.1.34	LN_R (单精度自然对数运算)	28
2.2.1.35	EXP_R (单精度指数运算)	29
2.2.1.36	ASIN_R (单精度反正弦运算)	30
2.2.1.37	ACOS_R (单精度反余弦运算)	31
2.2.1.38	EXPT_R (单精度幂运算)	32
2.2.1.39	SQRT_LR (双精度平方根运算)	33
2.2.1.40	LOG_LR (双精度常用对数运算)	34
2.2.1.41	LN_LR (双精度自然对数运算)	35
2.2.1.42	EXP_LR (双精度指数运算)	36
2.2.1.43	ASIN_LR (双精度反正弦运算)	37
2.2.1.44	ACOS_LR (双精度反余弦运算)	38
2.2.1.45	EXPT_LR (双精度幂运算)	39
2.2.1.46	MOVE (赋值)	40
2.2.2	逻辑运算	40
2.2.2.1	AND (与)	40
2.2.2.2	OR (或)	41
2.2.2.3	XOR (异或)	43
2.2.2.4	NOT (非)	44
2.2.2.5	AND_FB (与)	44
2.2.2.6	OR_FB (或)	46
2.2.2.7	NOT_FB (非)	47
2.2.2.8	AND3 (3 取与)	48
2.2.2.9	OR3 (3 取或)	48
2.2.3	比较运算	49
2.2.3.1	GT (大于)	49
2.2.3.2	LT (小于)	50
2.2.3.3	GE (大于等于)	51
2.2.3.4	LE (小于等于)	52
2.2.3.5	EQ (等于)	53
2.2.3.6	NE (不等于)	54
2.2.3.7	GT_DINT_FB (大于 (长整型))	55
2.2.3.8	LT_DINT_FB (小于 (长整型))	56
2.2.3.9	GE_DINT_FB (大于等于 (长整型))	57
2.2.3.10	LE_DINT_FB (小于等于 (长整型))	58
2.2.3.11	EQ_DINT_FB (等于 (长整型))	59
2.2.3.12	GT_REAL_FB (大于 (实型))	60
2.2.3.13	LT_REAL_FB (小于 (实型))	61

2.2.3.14	GE_REAL_FB (大于等于 (实型))	62
2.2.3.15	LE_REAL_FB (小于等于 (实型))	63
2.2.3.16	EQ_REAL_FB (等于 (实型))	64
2.2.4	选择运算	65
2.2.4.1	SEL (二选一)	65
2.2.4.2	MAX (取最大值)	66
2.2.4.3	MIN (取最小值)	66
2.2.5	移位运算	67
2.2.5.1	SHL (左移)	67
2.2.5.2	SHR (右移)	68
2.2.5.3	ROL (循环左移)	69
2.2.5.4	ROR (循环右移)	70
2.2.6	数据类型转换	71
2.2.6.1	BYTE_TO (BYTE 类型转换)	71
2.2.6.2	WORD_TO (WORD 类型转换)	73
2.2.6.3	DWORD_TO (DWORD 类型转换)	75
2.2.6.4	SINT_TO (SINT 类型转换)	77
2.2.6.5	INT_TO (INT 类型转换)	79
2.2.6.6	DINT_TO (DINT 类型转换)	81
2.2.6.7	REAL/LREAL_TO (REAL/LREAL 类型转换)	83
2.2.6.8	TIME_TO (TIME 类型转换)	85
2.2.6.9	DATE_TO (DATE 类型转换)	87
2.2.6.10	DT_TO (DT 类型转换)	89
2.2.6.11	TOD_TO (TOD 类型转换)	90
2.2.7	沿触发器	93
2.2.7.1	R_TRIG (上升沿检测触发器)	93
2.2.7.2	F_TRIG (下降沿检测触发器)	93
2.2.8	双稳态触发器	94
2.2.8.1	SR (置位优先双稳态器)	94
2.2.8.2	RS (复位优先双稳态器)	95
2.2.9	计数器	97
2.2.9.1	CTD (递减计数器)	97
2.2.9.2	CTU (递增计数器)	98
2.2.9.3	CTUD (递增/递减计数器)	99
2.2.10	定时器	101
2.2.10.1	TON (开延时定时器)	101
2.2.10.2	TONR (保持型开延时定时器)	102
2.2.10.3	TOF (关延时定时器)	104
2.2.10.4	TP (普通定时器)	105
2.2.10.5	RTC (设定时间计时)	106
2.3	基础应用库	108
2.3.1	BCD 转换指令	108
2.3.1.1	BCD_TO_INT (BCD 码转整型)	108
2.3.1.2	INT_TO_BCD (整型转 BCD 码)	109
2.3.2	数据拆分合成指令	109
2.3.2.1	UNPACK (位拆分指令)	109
2.3.2.2	EXTRACT (位提取指令)	111

2.3.2.3	PACK (位整合指令)	112
2.3.2.4	PUTBIT (位赋值命令)	113
2.3.2.5	DWORD_UNPACK_WORD (DWORD 拆分 WORD 指令)	114
2.3.2.6	DINT_UNPACK_WORD (DINT 拆分 WORD 指令)	115
2.3.2.7	REAL_UNPACK_WORD (REAL 拆分 WORD 指令)	115
2.3.2.8	WORD_PACK_DWORD (WORD 合并 DWORD 指令)	116
2.3.2.9	WORD_PACK_DINT (WORD 合并 DINT 指令)	116
2.3.2.10	WORD_PACK_REAL (WORD 合并 REAL 指令)	117
2.3.3	数学辅助功能	118
2.3.3.1	DERIVATIVE (微分)	118
2.3.3.2	INTEGRAL (积分)	119
2.3.3.3	STATISTICS_INT (整型统计)	120
2.3.3.4	STATISTICS_REAL (实型统计)	121
2.3.3.5	VARIANCE (平方偏差)	122
2.3.4	控制器指令	124
2.3.4.1	PID (比例积分微分控制器)	124
2.3.5	信号发生器指令	127
2.3.5.1	BLINK (脉冲信号发生器)	127
2.3.5.2	GEN (典型周期信号发生器)	128
2.3.5.3	RAND (随机数发生器)	132
2.3.6	模拟量处理指令	133
2.3.6.1	RAMP_INT (整型限速)	133
2.3.6.2	RAMP_REAL (实型限速)	135
2.3.6.3	CHARCURVE (特征曲线)	136
2.3.6.4	HYSTERESIS (滞后)	137
2.3.6.5	LIMITALARM (上下限报警)	139
2.3.6.6	HEX_ENGIN (16 进制数转换为工程量数据)	140
2.3.6.7	ENGIN_HEX (工程量数据转换为 16 进制数据)	142
2.4	系统库	143
2.4.1	时间相关指令	143
2.4.1.1	GetTickCnt (获取开机计数)	143
2.4.1.2	TimeDelay (时间延时)	144
2.4.1.3	SET_RTC (设置实时时钟)	144
2.4.1.4	GET_RTC (读取实时时钟)	145
2.4.2	模块信息指令	148
2.4.2.1	sysGetCPUModuleVern (获取控制器的固件版本信息)	148
2.4.2.2	sysGetComModuleVern (获取通信模块固件版本信息)	149
2.4.2.3	sysGetCheckTime (获取检验时间)	151
2.4.2.4	sysGetEthMac (获取 Mac 地址)	153
2.4.2.5	sysGetBatteryAlarm (获取电池电量报警)	155
2.4.2.6	sysGetTEMP (获取主控 CPU 温度)	156
2.4.2.7	sysGetModel (获取模块型号)	157
2.4.2.8	sysGetModuleSN (获取模块序列号)	158
2.4.3	系统状态指令	162
2.4.3.1	sysGetRunTime (获取任务运行时间)	162
2.4.3.2	sysGetKeyState (获取钥匙开关状态)	165
2.4.3.3	sysGetDPMasterState (获取 DP 主卡状态)	166

2.4.3.4	sysGetCPULoad（获取控制器负荷）	168
2.4.3.5	sysGetRedState（获取系统冗余状态）	169
2.4.3.6	sysMasterSwitchToSlave（主从切换）	171
2.4.3.7	sysGetDP SlaveState（获取 DP 从站诊断）	172
2.4.3.8	sysGetCPUDiagInfo（获取主控模块诊断）	173
2.4.3.9	sysGetComModuleDiagInfo（获取通讯模块诊断）	174
2.4.3.10	sysSetProfiOareq（安全从站 Profisafe 确认）	177

第1章 关于手册

1.1 文档更新

版本	日期	说明
V1.0	2017.08.30	创建
V1.1	2018.02.02	更新
V1.2	2018.11.31	新增章节 2.3.2 数据拆分合成指令
V1.0	2019.06.13	版权信息更新

1.2 文档用途

本手册给出了每个算法块的全面、详细的介绍和实例，帮助您更好地理解每个块的信息，从而正确选择和使用。

1.3 阅读对象

LKS 系统的编程人员、操作人员以及系统维护人员。

使用本手册的人员需要具备自动控制相关知识。

1.4 使用约定

1.4.1 菜单命令

本手册中菜单命令、菜单中的子命令、标签页名称用【】括注表示，如【文件】、【编辑】菜单。

1.4.2 鼠标指令

按下：按下鼠标左键一次并且不松开。

单击：按下鼠标左键一次并松开。

右击：按下鼠标右键一次并松开。

双击：快速按下鼠标左键两次并松开。

拖动：移动鼠标时按住鼠标左键不放。

1.4.3 键盘指令

本手册中涉及键盘上的按键时，统一用加粗表示，如 **Enter**。

1.4.4 在线帮助

集成于软件中的在线帮助是对本手册的补充。提供在线帮助的目的在于，在使用该软件时提供详细的支持。

该帮助系统通过一些界面集成于软件中：

- 按下 **F1** 键准确定位到相关的帮助说明。
- 可以通过菜单栏中【帮助】—【内容】调用当前软件的帮助。

由于手册和在线帮助有些内容的编排顺序不同，因此使用在线帮助时，可以借用它的【搜索】功能进行全文查找、定位。

1.4.5 重要信息

文档中使用以下标识明确相应信息：



- 危险图标，标识该操作有造成物理伤害或人身伤亡的潜在威胁。



- 电击图标，标识该操作有造成电击伤害的潜在威胁。



- 警告图标，标识该操作有造成软硬件设备故障或损坏的潜在威胁。



- 重要图标，标识需要理解的操作或功能的重要信息。



- 操作图标，标识该对象的打开或操作方法。

操作图标中的内容解读：

操作所发生的位置：鼠标指令+操作顺序。



- 菜单栏：单击【插入】—【输入元件】；
- 工具栏：；
- 快捷键：**Ctrl+I**；
- **POU**：右击空白区域，单击【输入元件】。

如上图中的第 1 条操作方法，在菜单栏处进行，依次单击【插入】菜单和子菜单中的【输入元件】命令。

SEE ALSO

- 参考图标，标识理解该功能需要参考的页面或内容。

1.5 产品文档目录



Safety FA-AutoThink 用户手册



LKS 安全型控制系统产品手册



LKS 安全型控制系统指令手册

第2章 库管理器

2.1 库指令使用说明

以下指令不支持用户使用：

指令名	所在库
sysGetDPConfirmState	HS_DPDIAG.HLF
sysGetDPSlaveComState	
DIProc	HS_IOPROC.HLF

2.2 标准库

2.2.1 数学运算

2.2.1.1 ADD（加）

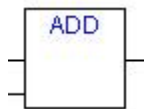


图 1 加

1. 参数说明

变量	说明	数据类型
输入	输入变量	DWORD、DINT、REAL、LREAL、TIME
输出	输出变量	DWORD、DINT、REAL、LREAL、TIME

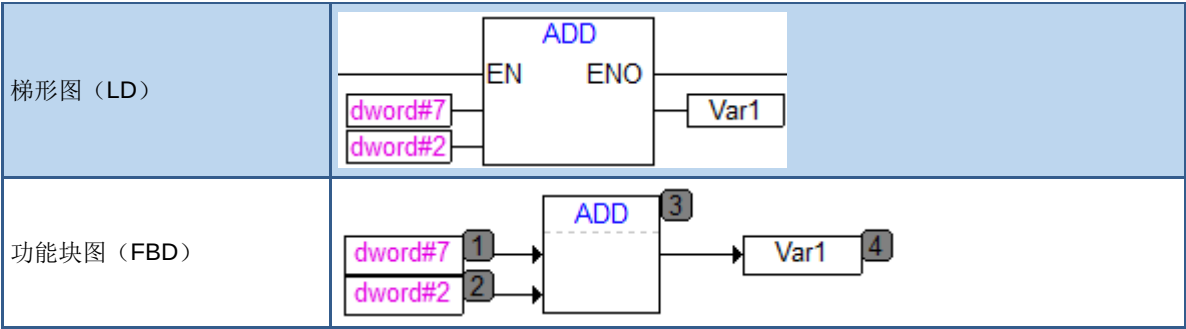
2. 详细说明

对输入变量完成加法运算。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		DWORD	0	G区

编程语言	程序
------	----



2.2.1.2 SUB (减)

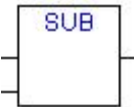


图 2 减

1. 参数说明

变量	说明	数据类型
输入	输入变量	DWORD、DINT、REAL、LREAL、TIME
输出	输出变量	DWORD、DINT、REAL、LREAL、TIME

2. 详细说明

对输入变量完成减法运算。

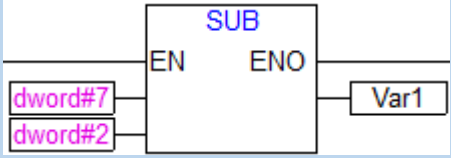
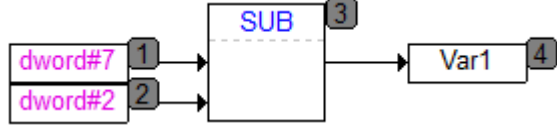
3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		DWORD	0	G区

编程语言

程序

梯形图 (LD)	
功能块图 (FBD)	

2.2.1.3 MUL（乘）

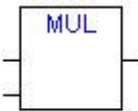


图 3 乘

1. 参数说明

变量	说明	数据类型
输入	输入变量	DWORD、DINT、REAL、LREAL
输出	输出变量	DWORD、DINT、REAL、LREAL

2. 详细说明

对输入变量完成乘法运算。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		DWORD	0	G区

编程语言

程序

梯形图（LD）

功能块图（FBD）

2.2.1.4 DIV（除）

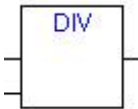


图 4 除

1. 参数说明

变量	说明	数据类型
输入	输入变量	DWORD、DINT、REAL、LREAL

输出	输出变量	DWORD、DINT、REAL、LREAL
----	------	-----------------------

2. 详细说明

对输入变量完成除法运算。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		DWORD	0	G区

编程语言

程序

梯形图 (LD)

功能块图 (FBD)

2.2.1.5 MOD（取余）

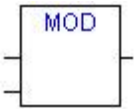


图 5 取余

1. 参数说明

变量	说明	数据类型
输入	输入变量	DWORD、DINT
输出	输出变量	DWORD、DINT

2. 详细说明

输入变量相除取余，结果为两数相除后的余数。

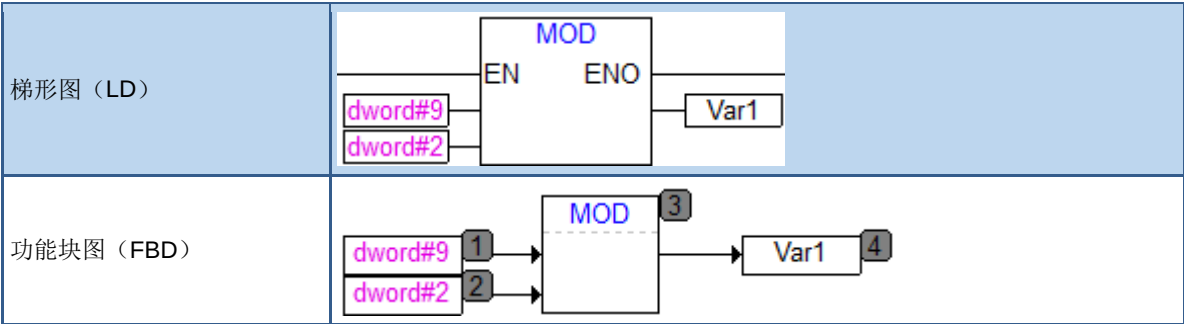
3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		DWORD	0	G区

编程语言

程序



2.2.1.6 ADD_WORD (加)

对两个 WORD 类型的输入变量完成加法运算。

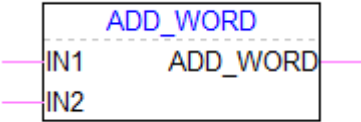


图 6 加法运算

1. 参数说明

输入	数据类型
IN1	WORD
IN2	WORD
输出	
ADD_WORD	WORD

2. 指令使用举例

请参见 2.2.1.1 ADD (加)。

2.2.1.7 SUB_WORD (减)

对两个 WORD 类型的输入变量完成减法运算。

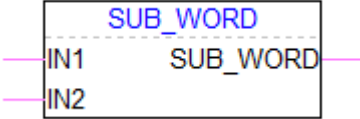


图 7 减法运算

1. 参数说明

输入	数据类型
----	------

IN1	WORD
IN2	WORD
输出	
SUB_WORD	WORD

2. 指令使用举例

请参见 2.2.1.2 SUB（减）。

2.2.1.8 MUL_WORD（乘）

对两个 WORD 类型的输入变量完成乘法运算。

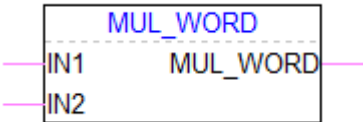


图 8 乘法运算

1. 参数说明

输入	数据类型
IN1	WORD
IN2	WORD
输出	
MUL_WORD	WORD

2. 指令使用举例

请参见 2.2.1.3 MUL（乘）。

2.2.1.9 DIV_WORD（除）

对两个 WORD 类型的输入变量完成除法运算。

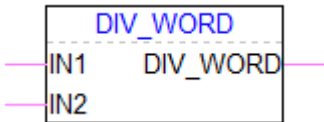


图 9 除法运算

1. 参数说明

输入	数据类型
IN1	WORD
IN2	WORD
输出	
DIV_WORD	WORD

2. 指令使用举例

请参见 2.2.1.4 DIV（除）。

2.2.1.10 MOD_WORD（取余）

两个 WORD 类型输入变量相除取余，结果为两数相除后的余数。

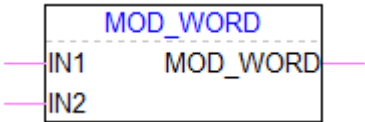


图 10 取余运算

1. 参数说明

输入	数据类型
IN1	WORD
IN2	WORD
输出	
MOD_WORD	WORD

2. 指令使用举例

请参见 2.2.1.5 MOD（取余）。

2.2.1.11 ADD_INT（加）

对两个 INT 类型的输入变量完成加法运算。

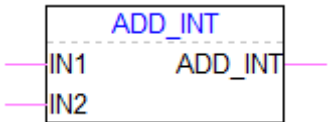


图 11 加法运算

1. 参数说明

输入	数据类型
IN1	INT
IN2	INT
输出	
ADD_INT	INT

2. 指令使用举例

请参见 [2.2.1.1 ADD（加）](#)。

2.2.1.12 SUB_INT（减）

对两个 INT 类型的输入变量完成减法运算。

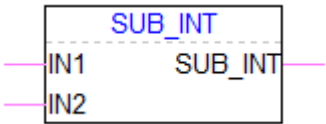


图 12 减法运算

1. 参数说明

输入	数据类型
IN1	INT
IN2	INT
输出	
SUB_INT	INT

2. 指令使用举例

请参见 [2.2.1.2 SUB（减）](#)。

2.2.1.13 MUL_INT（乘）

对两个 INT 类型的输入变量完成乘法运算。

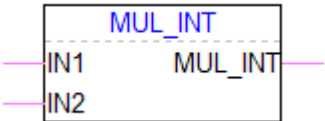


图 13 乘法运算

1. 参数说明

输入	数据类型
IN1	INT
IN2	INT
输出	
MUL_INT	INT

2. 指令使用举例

请参见 2.2.1.3 MUL（乘）。

2.2.1.14 DIV_INT（除）

对两个 INT 类型的输入变量完成除法运算。

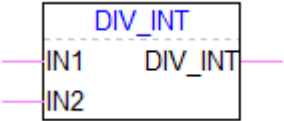


图 14 除法运算

1. 参数说明

输入	数据类型
IN1	INT
IN2	INT
输出	
DIV_INT	INT

2. 指令使用举例

请参见 2.2.1.4 DIV（除）。

2.2.1.15 MOD_INT（取余）

两个 INT 类型输入变量相除取余，结果为两数相除后的余数。

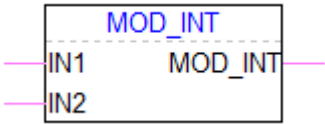


图 15 取余运算

1. 参数说明

输入	数据类型
IN1	INT
IN2	INT
输出	
MOD_INT	INT

2. 指令使用举例

请参见 2.2.1.5 MOD（取余）。

2.2.1.16 DIV_DWORD（除）

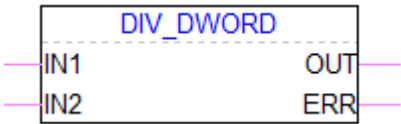


图 16 除法运算算法

1. 参数说明

输入	数据类型	描述	初始值	补充
IN1	DWORD	被除数	0	
IN2	DWORD	除数	0	不能为 0
输出				
OUT	DWORD	除法运算后的计算结果	0	

ERR	BOOL	ERR=0 时，表示功能块正常运算 ERR=1 时，表示发生除 0 运算，此时 OUT 值保持为上一拍值	0	
-----	------	---	---	--

2. 详细说明

当除数 IN2 不为 0 时，对输入变量做除法运算后输出；若除数 IN2 等于 0，输出保持上一拍值并置位 ERR 端口。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	DIV_DWORD_1		DIV_DWORD		G区
0002	Var1		DWORD	0	G区
0003	Var2		BOOL	FALSE	G区

编程语言

程序

梯形图 (LD)

功能块图 (FBD)

2.2.1.17 DIV_DINT (除)

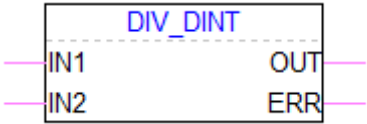


图 17 除法运算算法

1. 参数说明

输入	数据类型	描述	初始值	补充
IN1	DINT	被除数	0	
IN2	DINT	除数	0	不能为 0
输出				
OUT	DINT	除法运算后的计算结果	0	

ERR	BOOL	ERR=0 时，表示功能块正常运算 ERR=1 时，表示发生除 0 运算，此时 OUT 值保持为上一拍值	0	
-----	------	---	---	--

2. 详细说明

当除数 IN2 不为 0 时，对输入变量做除法运算后输出；若除数 IN2 等于 0，输出保持上一拍值并置位 ERR 端口。

3. 指令使用举例

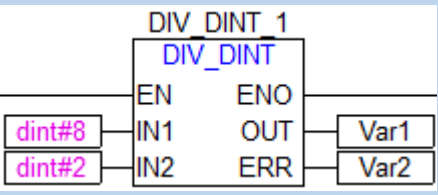
变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	DIV_DINT_1		DIV_DINT		G区
0002	Var1		DINT	0	G区
0003	Var2		BOOL	FALSE	G区

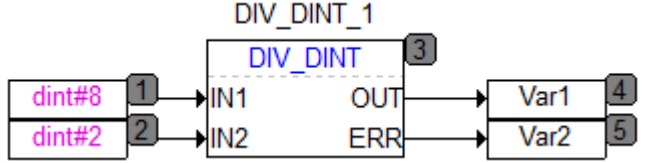
编程语言

程序

梯形图 (LD)



功能块图 (FBD)



2.2.1.18 DIV_REAL (除)

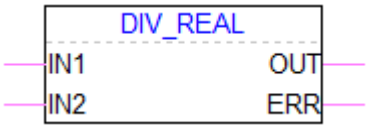


图 18 除法运算算法

1. 参数说明

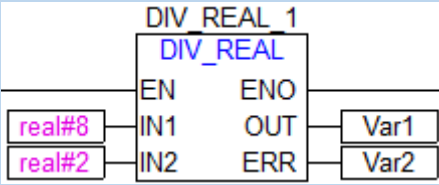
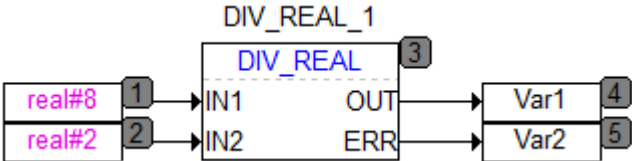
输入	数据类型	描述	初始值	补充
IN1	REAL	被除数	0	
IN2	REAL	除数	0	不能为 0
输出				
OUT	REAL	除法运算后的计算结果	0	
ERR	BOOL	ERR=0 时，表示功能块正常运算	0	

		ERR=1 时，表示发生除 0 运算，此时 OUT 值保持为上一拍值		
--	--	------------------------------------	--	--

2. 详细说明

当除数 IN2 不为 0 时，对输入变量做除法运算后输出；若除数 IN2 等于 0，输出保持上一拍值并置位 ERR 端口。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	DIV_REAL_1		DIV_REAL		G区
0002	Var1		REAL	0	G区
0003	Var2		BOOL	FALSE	G区
编程语言		程序			
梯形图 (LD)					
功能块图 (FBD)					

2.2.1.19 DIV_LREAL (除)

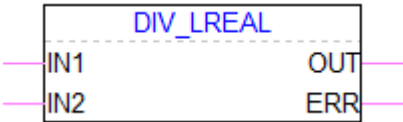


图 19 除法运算算法

1. 参数说明

输入	数据类型	描述	初始值	补充
IN1	LREAL	被除数	0	
IN2	LREAL	除数	0	不能为 0
输出				
OUT	LREAL	除法运算后的计算结果	0	
ERR	BOOL	ERR=0 时，表示功能块正常运算 ERR=1 时，表示发生除 0 运算，此时 OUT 值保持为上一拍值	0	

2. 详细说明

当除数 IN2 不为 0 时，对输入变量做除法运算后输出；若除数 IN2 等于 0，输出保持上一拍值并置位 ERR 端口。

3. 指令使用举例

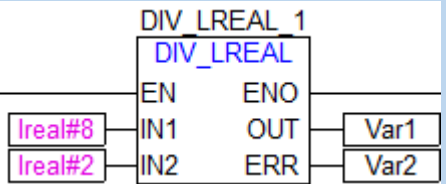
变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	DIV_LREAL_1		DIV_LREAL		G区
0002	Var1		LREAL	0	G区
0003	Var2		BOOL	FALSE	G区

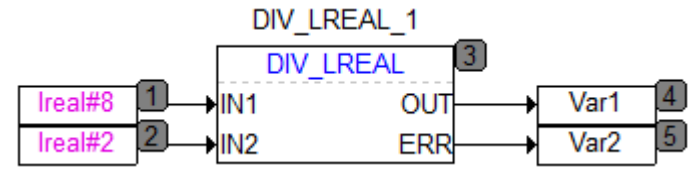
编程语言

程序

梯形图 (LD)



功能块图 (FBD)



2.2.1.20 ABS（绝对值）



图 20 取绝对值

1. 参数说明

输入	说明	数据类型
输入	输入变量	LREAL
输出		
输出	取绝对值后的计算结果	LREAL

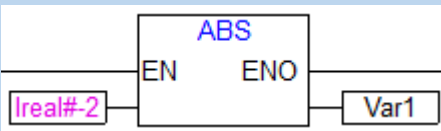
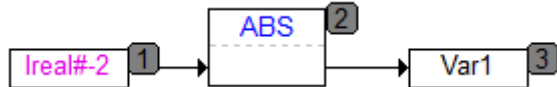
2. 详细说明

对输入变量取绝对值后输出。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		LREAL	0	G区

编程语言	程序
梯形图（LD）	
功能块图（FBD）	

2.2.1.21 SQRT（平方根）



图 21 开平方运算

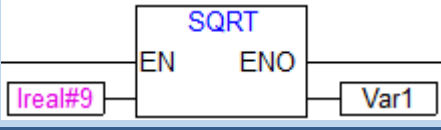
1. 参数说明

输入	说明	数据类型
输入	输入变量	LREAL
输出		
输出	开方后的计算结果	LREAL

2. 详细说明

对输入变量开平方后输出。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		LREAL	0	G区
编程语言	程序				
梯形图（LD）					
功能块图（FBD）					

2.2.1.22 LOG（常用对数）



图 22 以 10 为底的对数运算

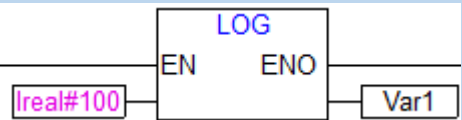
1. 参数说明

输入	说明	数据类型
输入	输入变量	LREAL
输出		
输出	对数运算后的计算结果	LREAL

2. 详细说明

对输入变量取以 10 为底的对数后输出结果。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		LREAL	0	G区
编程语言		程序			
梯形图 (LD)					
功能块图 (FBD)					

2.2.1.23 LN（自然对数）



图 23 自然对数运算

1. 参数说明

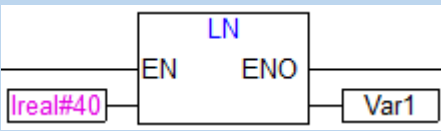
输入	说明	数据类型
输入	输入变量	LREAL
输出		
输出	对数运算后的计算结果	LREAL

2. 详细说明

对输入变量取自然对数后输出结果。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		LREAL	0	G区

编程语言	程序
梯形图（LD）	
功能块图（FBD）	

2.2.1.24 EXP（指数）



图 24 指数函数运算

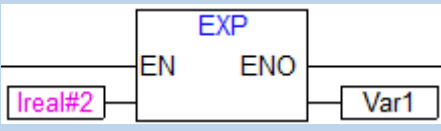
1. 参数说明

输入	说明	数据类型
输入	输入变量	LREAL
输出		
输出	指数运算后的计算结果	LREAL

2. 详细说明

对 e 进行输入变量值次方后输出结果。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		LREAL	0	G区
编程语言	程序				
梯形图（LD）					
功能块图（FBD）					

2.2.1.25 SIN（正弦）



图 25 正弦函数运算

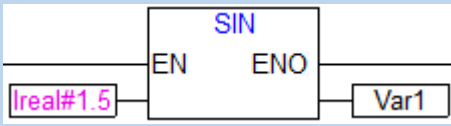
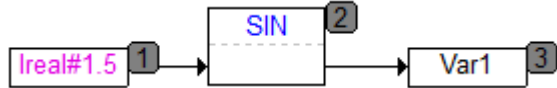
1. 参数说明

输入	说明	数据类型	备注
输入	输入变量	LREAL	弧度值 LREAL 类型输入最大支持二进制 52 bits 位所能表示的数据范围，并非默认的 64 bits
输出			
输出	正弦运算后的计算结果	LREAL	

2. 详细说明

对输入变量进行正弦运算后输出结果，输入是弧度值。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		LREAL	0	G区
编程语言		程序			
梯形图 (LD)					
功能块图 (FBD)					

2.2.1.26 COS（余弦）



图 26 余弦函数运算

1. 参数说明

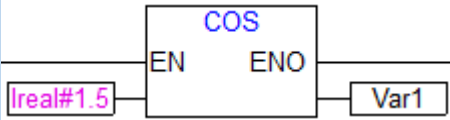
输入	说明	数据类型	备注
输入	输入变量	LREAL	弧度值 LREAL 类型输入最大支持二进制 52 bits 位所能表示的数据范围，并非默认的 64 bits。
输出			
输出	余弦运算后的计算结果	LREAL	

2. 详细说明

对输入变量进行余弦运算后输出结果，输入是弧度值。

3. 指令使用举例

变量定义					
------	--	--	--	--	--

序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		LREAL	0	G区
编程语言		程序			
梯形图（LD）					
功能块图（FBD）					

2.2.1.27 TAN（正切）



图 27 正切函数运算

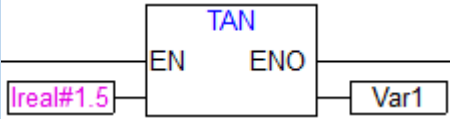
1. 参数说明

输入	说明	数据类型	备注
输入	输入变量	LREAL	弧度值 LREAL 类型输入最大支持二进制 52 bits 位所能表示的数据范围，并非默认的 64 bits。
输出			
输出	正切运算后的计算结果	LREAL	

2. 详细说明

对输入变量进行正切运算后输出，输入是弧度值。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		LREAL	0	G区
编程语言		程序			
梯形图（LD）					
功能块图（FBD）					

2.2.1.28 ASIN（反正弦）



图 28 反正弦函数运算

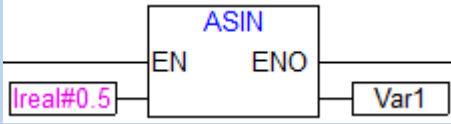
1. 参数说明

输入	说明	数据类型	备注
输入	输入变量	LREAL	
输出			
输出	反正弦运算后的计算结果	LREAL	弧度值

2. 详细说明

对输入变量进行反正弦运算后输出。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		LREAL	0	G区
编程语言		程序			
梯形图（LD）		 The diagram shows a ladder logic network. A pink box labeled "lreal#0.5" is connected to the "EN" input of a function block labeled "ASIN". The "ENO" output of the "ASIN" block is connected to a variable box labeled "Var1".			
功能块图（FBD）		 The diagram shows a function block diagram. A pink box labeled "lreal#0.5" with a small grey box "1" is connected to the input of a function block labeled "ASIN" with a small grey box "2". The output of the "ASIN" block is connected to a variable box labeled "Var1" with a small grey box "3".			

2.2.1.29 ACOS（反余弦）



图 29 反余弦函数运算

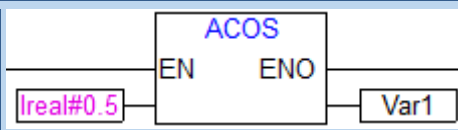
1. 参数说明

输入	说明	数据类型	备注
输入	输入变量	LREAL	
输出			
输出	反余弦运算后的计算结果	LREAL	弧度值

2. 详细说明

对输入变量进行反余弦运算后输出。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		LREAL	0	G区
编程语言		程序			
梯形图（LD）					
功能块图（FBD）					

2.2.1.30 ATAN（反正切）



图 30 反正切函数运算

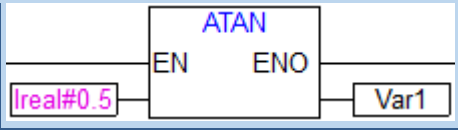
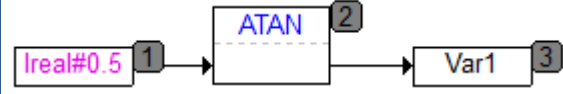
1. 参数说明

输入	说明	数据类型	备注
输入	输入变量	LREAL	
输出			
输出	反正切运算后的计算结果	LREAL	弧度值

2. 详细说明

对输入变量进行反正切运算后输出。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		LREAL	0	G区
编程语言		程序			
梯形图（LD）					
功能块图（FBD）					

2.2.1.31 EXPT（幂）

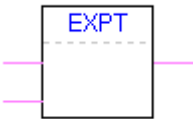


图 31 变量与变量的乘幂运算

1. 参数说明

输入	说明	数据类型
输入 1	输入变量 1（Y）	LREAL
输入 2	输入变量 2（n）	LREAL
输出		
输出	幂运算后的计算结果	LREAL

2. 详细说明

对输入 1 进行输入 2 次方运算后输出结果。



- 输入 1 和输入 2 在赋值时应符合数学运算规则。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		LREAL	0	G区
编程语言		程序			
梯形图（LD）					
功能块图（FBD）					

2.2.1.32 SQRT_R（单精度平方根运算）



图 32 开平方运算算法

1. 参数说明

输入	数据类型	描述	初始值	补充
IN	REAL	输入变量	0	为大于或等于零的实数
输出				
OUT	REAL	开方后的计算结果	0	
ERR	BOOL	ERR=0 时，表示功能块正常运算 ERR=1 时，表示输入 IN 为负数， 此时 OUT 值保持为上一拍值	0	

2. 详细说明

若 $IN \geq 0$ ，对输入变量开平方后输出；若 $IN < 0$ ，ERR 置位并保持上一拍输出值。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	SQRT_R_1		SQRT_R		G区
0002	Var1		REAL	0	G区
0003	Var2		BOOL	FALSE	G区
编程语言		程序			
梯形图（LD）					
功能块图（FBD）					

2.2.1.33 LOG_R（单精度常用对数运算）

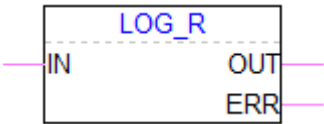


图 33 以 10 号底的对数运算算法

1. 参数说明

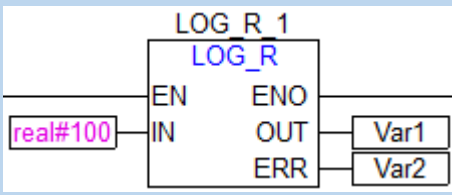
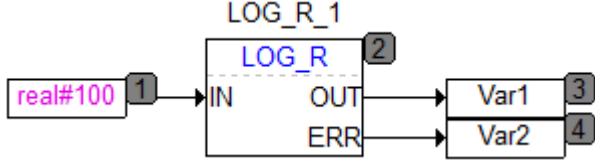
输入	数据类型	描述	初始值	补充
IN	REAL	输入变量	0	为大于零的实数
输出				
OUT	REAL	对数运算后的计算结果	0	
ERR	BOOL	ERR=0 时，表示功能块正常运算； ERR=1 时，表示输入小于等于零，此时 OUT 值保持为上一拍值	0	

2. 详细说明

若 $IN > 0$ ，对输入变量取以 10 为底的对数后输出结果；若 $IN \leq 0$ ，ERR 置位并保持上一拍输出值。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	LOG_R_1		LOG_R		G区
0002	Var1		REAL	0	G区
0003	Var2		BOOL	FALSE	G区

编程语言	程序
梯形图 (LD)	
功能块图 (FBD)	

2.2.1.34 LN_R（单精度自然对数运算）

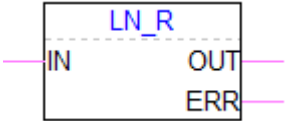


图 34 自然对数运算算法

1. 参数说明

输入	数据类型	描述	初始值	补充
IN	REAL	输入变量	0	为大于零的实数

输出				
OUT	REAL	对数运算后的计算结果	0	
ERR	BOOL	ERR=0 时，表示功能块正常运算 ERR=1 时，表示输入小于等于零，此时 OUT 值保持为上一拍值	0	

2. 详细说明

若 $IN > 0$ ，对输入变量取自然对数后输出结果；若 $IN \leq 0$ ，ERR 置位并保持上一拍输出值。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	LN_R_1		LN_R		G区
0002	Var1		REAL	0	G区
0003	Var2		BOOL	FALSE	G区

编程语言	程序
梯形图 (LD)	
功能块图 (FBD)	

2.2.1.35 EXP_R (单精度指数运算)

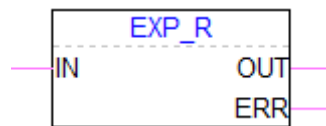


图 35 指数函数运算算法

1. 参数说明

输入	数据类型	描述	初始值	补充
IN	REAL	输入变量	0	为小于等于 88 的实数
输出				
OUT	REAL	指数运算后的计算结果	0	
ERR	BOOL	ERR=0 时，表示功能块正常运算 ERR=1 时，表示发生大于 88 的指数运算，此时 OUT 值保持为上一拍值	0	

2. 详细说明

若 $IN \leq 88$ ，对 e 进行输入变量值次方后输出结果；若 $IN > 88$ ，ERR 置位并保持上一拍输出值。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	EXP_R_1		EXP_R		G区
0002	Var1		REAL	0	G区
0003	Var2		BOOL	FALSE	G区

编程语言

程序

梯形图 (LD)

功能块图 (FBD)

2.2.1.36 ASIN_R（单精度反正弦运算）



图 36 反正弦函数运算算法

1. 参数说明

输入	数据类型	描述	初始值	补充
IN	REAL	输入变量	0	取值范围为[-1,1]
输出				
OUT	REAL	反正弦运算后的计算结果	0	
ERR	BOOL	ERR=0 时，表示功能块正常运算 ERR=1 时，表示发生在 [-1,1] 范围外的反正弦运算，此时 OUT 值保持为上一拍值	0	

2. 详细说明

若 $-1 \leq IN \leq 1$ ，对输入变量进行反正弦运算后输出；若 $IN > 1$ 或者 $IN < -1$ ，ERR 置位并保持上一拍输出值。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	ASIN_R_1		ASIN_R		G区
0002	Var1		REAL	0	G区
0003	Var2		BOOL	FALSE	G区

编程语言

程序

梯形图 (LD)

功能块图 (FBD)

2.2.1.37 ACOS_R（单精度反余弦运算）

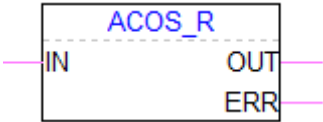


图 37 反余弦函数运算算法

1. 参数说明

输入	数据类型	描述	初始值	补充
IN	REAL	输入变量	0	取值范围为[-1,1]
输出				
OUT	REAL	反余弦运算后的计算结果	0	
ERR	BOOL	ERR=0 时，表示功能块正常运算 ERR=1 时，表示发生在[-1,1]范围外的反余弦运算，此时 OUT 值保持为上一拍值	0	

2. 详细说明

若 $-1 \leq IN \leq 1$ ，对输入变量进行反余弦运算后输出；若 $IN > 1$ 或者 $IN < -1$ ，ERR 置位并保持上一拍输出值。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	ACOS_R_1		ACOS_R		G区
0002	Var1		REAL	0	G区
0003	Var2		BOOL	FALSE	G区
编程语言		程序			
梯形图（LD）					
功能块图（FBD）					

2.2.1.38 EXPT_R（单精度幂运算）



图 38 变量与变量的乘幂运算算法

1. 参数说明

输入	数据类型	描述	初始值	补充
IN1	REAL	输入变量 1	0	取值范围为[0, +∞)
IN2	REAL	输入变量 2	0	
输出				
OUT	REAL	变量与变量的乘幂运算后的计算结果	0	
ERR	BYTE	ERR=0 时，表示功能块正常运算 ERR=1 时，表示 IN1 值超出范围，此时 OUT 值保持为上一拍值 ERR=2 时，表示 IN1=0，IN2<=0，此时 OUT 值保持为上一拍值 ERR=3 时，表示 $\ln \ln(IN1) + \ln \ln2 > 4.466$ ，此时 OUT 值保持为上一拍值	0	

2. 详细说明

在输入 1 和输入 2 满足正确范围时，对输入 1 进行输入 2 次方运算后输出结果；当输入 1 或者输入 2 不满足正确范围时，ERR 输出不为 0 且输出保持上一拍值。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	EXPT_R_1		EXPT_R		G区
0002	Var1		REAL	0	G区
0003	Var2		BYTE	0	G区

编程语言	程序
梯形图 (LD)	
功能块图 (FBD)	

2.2.1.39 SQRT_LR（双精度平方根运算）



图 39 开平方运算算法

1. 参数说明

输入	数据类型	描述	初始值	补充
IN	LREAL	输入变量	0	为大于或等于零的实数
输出				
OUT	LREAL	开方后的计算结果	0	
ERR	BOOL	ERR=0 时，表示功能块正常运算 ERR=1 时，表示输入 IN 为负数，此时 OUT 值保持为上一拍值	0	

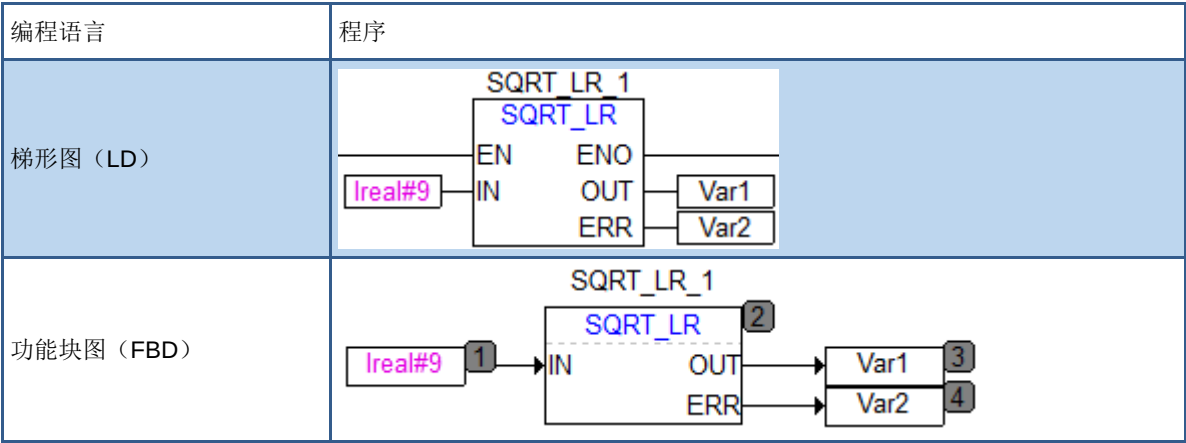
2. 详细说明

若 $IN \geq 0$ ，对输入变量开平方后输出；若 $IN < 0$ ，ERR 置位并保持上一拍输出值。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	SQRT_LR_1		SQRT_LR		G区
0002	Var1		LREAL	0	G区
0003	Var2		BOOL	FALSE	G区



2.2.1.40 LOG_LR（双精度常用对数运算）



图 40 以 10 号底的对数运算算法

1. 参数说明

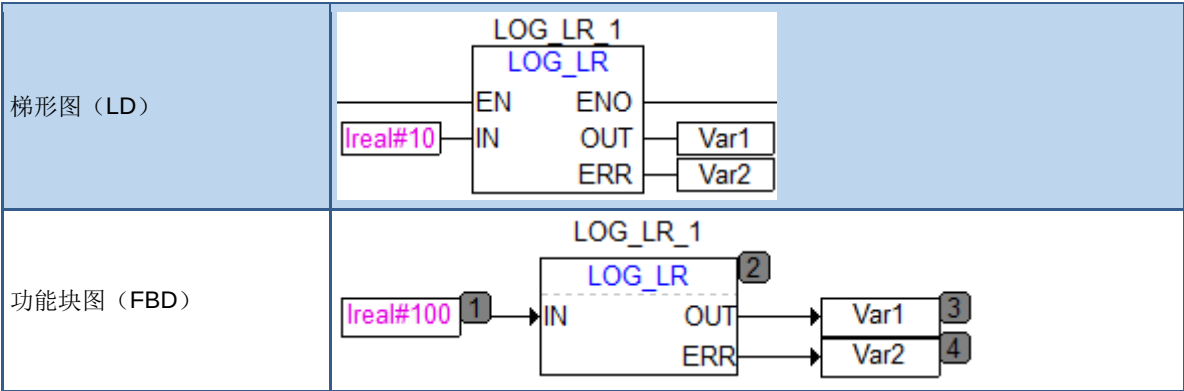
输入	数据类型	描述	初始值	补充
IN	LREAL	输入变量	0	为大于零的实数
输出				
OUT	LREAL	对数运算后的计算结果	0	
ERR	BOOL	ERR=0 时，表示功能块正常运算 ERR=1 时，表示输入小于等于零，此时 OUT 值保持为上一拍值	0	

2. 详细说明

若 IN>0，对输入变量取以 10 为底的对数后输出结果；若 IN<=0，ERR 置位并保持上一拍输出值。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	LOG_LR_1		LOG_LR		G区
0002	Var1		LREAL	0	G区
0003	Var2		BOOL	FALSE	G区
编程语言		程序			



2.2.1.41 LN_LR（双精度自然对数运算）

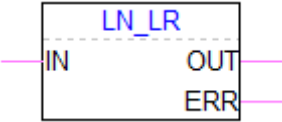


图 41 自然对数运算算法

1. 参数说明

输入	数据类型	描述	初始值	补充
IN	LREAL	输入变量	0	为大于零的实数
输出				
OUT	LREAL	对数运算后的计算结果	0	
ERR	BOOL	ERR=0 时，表示功能块正常运算 ERR=1 时，表示输入小于等于零，此时 OUT 值保持为上一拍值	0	

2. 详细说明

若 $IN > 0$ ，对输入变量取自然对数后输出结果；若 $IN \leq 0$ ，ERR 置位并保持上一拍输出值。

3. 指令使用举例

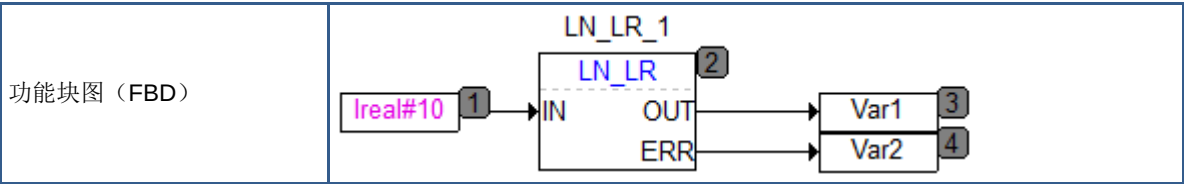
变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	LN_LR_1		LN_LR		G区
0002	Var1		LREAL	0	G区
0003	Var2		BOOL	FALSE	G区

编程语言

程序

梯形图（LD）



2.2.1.42 EXP_LR（双精度指数运算）

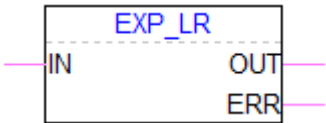


图 42 指数函数运算算法

1. 参数说明

输入	数据类型	描述	初始值	补充
IN	LREAL	输入变量	0	为小于等于 709 的实数
输出				
OUT	LREAL	指数运算后的计算结果	0	
ERR	BOOL	ERR=0 时，表示功能块正常运算 ERR=1 时，表示发生大于 709 的指数运算，此时 OUT 值保持为上一拍值	0	

2. 详细说明

若 IN≤709，对 e 进行输入变量值次方后输出结果；若 IN>709，ERR 置位并保持上一拍输出值。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	EXP_LR_1		EXP_LR		G区
0002	Var1		LREAL	0	G区
0003	Var2		BOOL	FALSE	G区

编程语言

程序

梯形图（LD）

功能块图（FBD）

2.2.1.43 ASIN_LR（双精度反正弦运算）

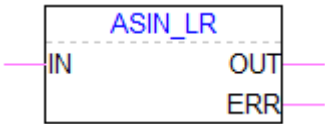


图 43 反正弦函数运算算法

1. 参数说明

输入	数据类型	描述	初始值	补充
IN	LREAL	输入变量	0	取值范围为[-1,1]
输出				
OUT	LREAL	反正弦运算后的计算结果	0	
ER	BOOL	ERR=0 时，表示功能块正常运算 ERR=1 时，表示发生在[-1,1]范围外的反正弦运算，此时 OUT 值保持为上一拍值	0	

2. 详细说明

若 $-1 \leq IN \leq 1$ ，对输入变量进行反正弦运算后输出；若 $IN > 1$ 或者 $IN < -1$ ，ERR 置位并保持上一拍输出值。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	ASIN_LR_1		ASIN_LR		G区
0002	Var1		LREAL	0	G区
0003	Var2		BOOL	FALSE	G区
编程语言		程序			
梯形图（LD）					
功能块图（FBD）					

2.2.1.44 ACOS_LR（双精度反余弦运算）

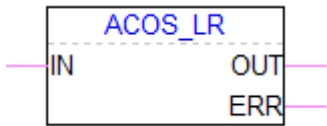


图 44 反余弦函数运算算法

1. 参数说明

输入	数据类型	描述	初始值	补充
IN	LREAL	输入变量	0	取值范围为[-1,1]
输出				
OUT	LREAL	反余弦运算后的计算结果	0	
ERR	BOOL	ERR=0 时，表示功能块正常运算 ERR=1 时，表示发生在[-1,1]范围外的反余弦运算，此时 OUT 值保持为上一拍值	0	

2. 详细说明

若 $-1 \leq IN \leq 1$ ，对输入变量进行反余弦运算后输出；若 $IN > 1$ 或者 $IN < -1$ ，ERR 置位并保持上一拍输出值。

3. 指令使用举例

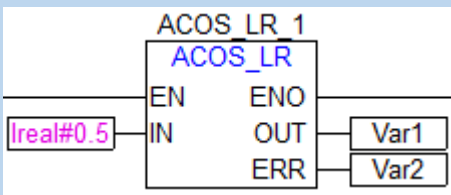
变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	ACOS_LR_1		ACOS_LR		G区
0002	Var1		LREAL	0	G区
0003	Var2		BOOL	FALSE	G区

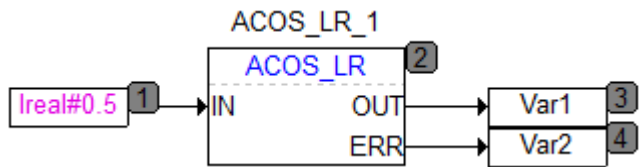
编程语言

程序

梯形图（LD）



功能块图（FBD）



2.2.1.45 EXPT_LR（双精度幂运算）

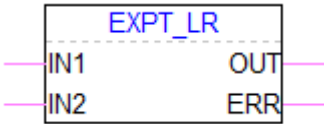


图 45 变量与变量的乘幂运算算法

1. 参数说明

输入	数据类型	描述	初始值	补充
IN1	LREAL	输入变量 1	0	取值范围为[0, +∞)
IN2	LREAL	输入变量 2	0	
输出				
OUT	LREAL	变量与变量的乘幂运算后的计算结果	0	
ERR	BYTE	ERR=0 时，表示功能块正常运算 ERR=1 时，表示 IN1 值超出范围，此时 OUT 值保持为上一拍值 ERR=2 时，表示 IN1=0 时 IN2<=0，此时 OUT 值保持为上一拍值 ERR=3 时，表示 $\ln \ln(IN1) + \ln \ln(IN2) > 6.562$ ，此时 OUT 值保持为上一拍值	0	

2. 详细说明

在输入 1 和输入 2 满足正确范围时，对输入 1 进行输入 2 次方运算后输出结果；当输入 1 或者输入 2 不满足正确范围时，ERR 输出不为 0 且输出保持上一拍值。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	EXPT_LR_1		EXPT_LR		G区
0002	Var1		LREAL	0	G区
0003	Var2		BYTE	0	G区

编程语言

程序

梯形图（LD）

EXPT_LR 1

EXPT_LR

EN

ENO

IN1

OUT

IN2

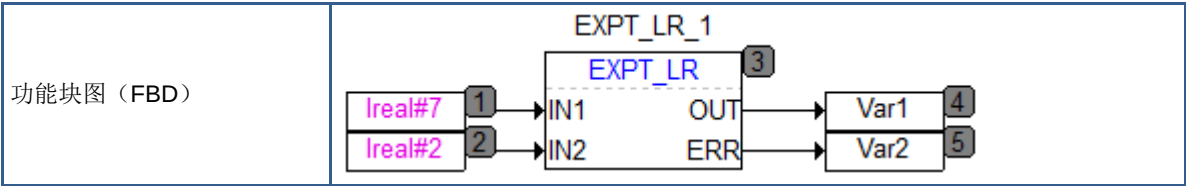
ERR

lreal#7

lreal#2

Var1

Var2



2.2.1.46 MOVE（赋值）

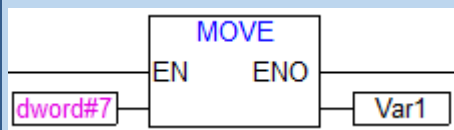


图 46 赋值

1. 参数说明

变量	说明	数据类型
输入	输入变量	BOOL、BYTE、WORD、DWORD、SINT、INT、DINT、REAL、LREAL、TIME、TOD、DT、DATE
输出	输出变量	BOOL、BYTE、WORD、DWORD、SINT、INT、DINT、REAL、LREAL、TIME、TOD、DT、DATE

2. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		DWORD	0	G区
编程语言		程序			
梯形图（LD）					

2.2.2 逻辑运算

2.2.2.1 AND（与）

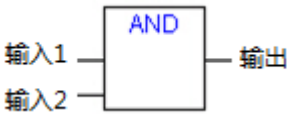


图 47 与

1. 参数说明

变量	说明	数据类型
输入 1	输入变量	BOOL、BYTE、WORD、DWORD

输入 2	输入变量	BOOL、BYTE、WORD、DWORD
...	...	...
输出	输出变量	BOOL、BYTE、WORD、DWORD

真值表:

输入 1	输入 2	输出
0	0	0
0	1	0
1	0	0
1	1	1

2. 详细说明
- 对输入变量按位完成与运算。
3. 指令使用举例

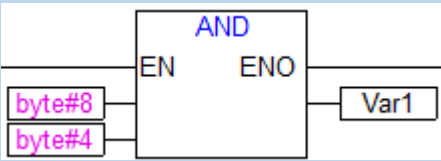
变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		BYTE	0	G区

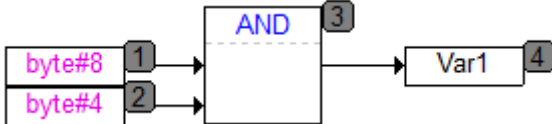
编程语言

程序

梯形图 (LD)



功能块图 (FBD)



2.2.2.2 OR (或)

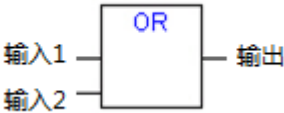


图 48 或

1. 参数说明

变量	说明	数据类型
输入 1	输入变量	BOOL、BYTE、WORD、DWORD
输入 2	输入变量	BOOL、BYTE、WORD、DWORD
...	...	...
输出	输出变量	BOOL、BYTE、WORD、DWORD

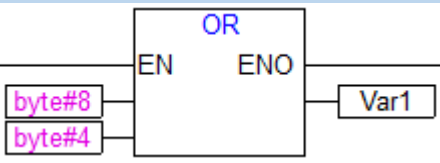
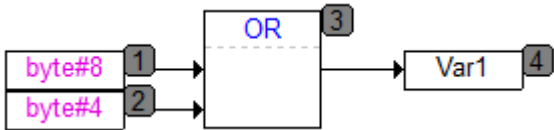
真值表：

输入 1	输入 2	输出
0	0	0
0	1	1
1	0	1
1	1	1

2. 详细说明

对于输入变量按位完成或运算。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		BYTE	0	G区
编程语言		程序			
梯形图（LD）					
功能块图（FBD）					

2.2.2.3 XOR（异或）

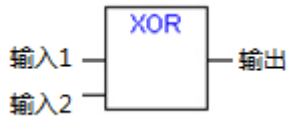


图 49 异或

1. 参数说明

变量	说明	数据类型
输入 1	输入变量	BOOL、BYTE、WORD、DWORD
输入 2	输入变量	BOOL、BYTE、WORD、DWORD
输出	输出变量	BOOL、BYTE、WORD、DWORD

真值表：

输入 1	输入 2	输出
0	0	0
0	1	1
1	0	1
1	1	0

2. 详细说明

对于输入变量按位完成异或运算。

3. 指令使用举例

变量定义

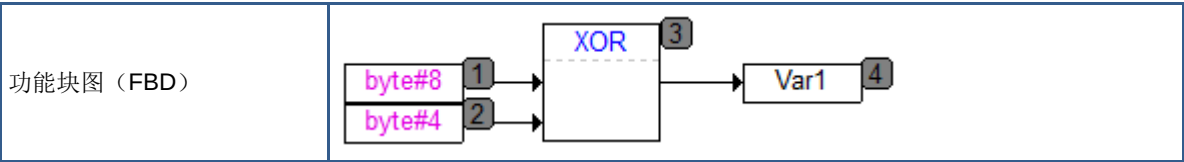
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		BYTE	0	G区

编程语言

程序

梯形图（LD）

A ladder logic diagram for an XOR instruction. It shows a rectangular block labeled "XOR" in blue. On the left side of the block, there are two input terminals labeled "byte#8" and "byte#4" in pink. On the right side, there is an output terminal labeled "Var1" in a white box. The block also has "EN" and "ENO" labels on its left and right sides respectively.



2.2.2.4 NOT（非）

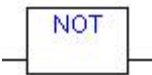


图 50 非

1. 参数说明

变量	说明	数据类型
输入	输入变量	BOOL、BYTE、WORD、DWORD
输出	输出变量	BOOL、BYTE、WORD、DWORD

2. 详细说明

对于输入变量按位完成取反运算。

3. 指令使用举例

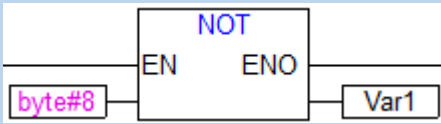
变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		BYTE	0	G区

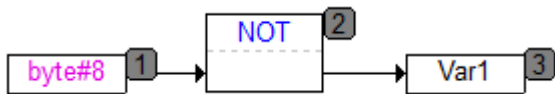
编程语言

程序

梯形图（LD）



功能块图（FBD）



2.2.2.5 AND_FB（与）

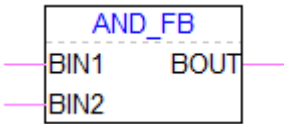


图 51 与

1. 参数说明

输入	说明	数据类型
----	----	------

BIN1	输入 1	BOOL
BIN2	输入 2	BOOL
输出		
BOUT	输出	BOOL

真值表:

输入 1	输入 2	输出
0	0	0
0	1	0
1	0	0
1	1	1

2. 详细说明

对输入布尔变量完成与运算。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	AND_FB_1		AND_FB		G区
0002	Var1		BOOL	FALSE	G区

编程语言

程序

梯形图 (LD)

AND FB 1
AND_FB

EN ENO
BIN1 BOUT

TRUE
FALSE

BIN2

Var1

功能块图 (FBD)

AND_FB_1
AND_FB

1 2

BIN1 BIN2

3 BOUT

4 Var1

2.2.2.6 OR_FB（或）

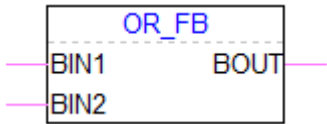


图 52 或

1. 参数说明

输入	说明	数据类型
BIN1	输入 1	BOOL
BIN2	输入 2	BOOL
输出		
BOUT	输出	BOOL

真值表：

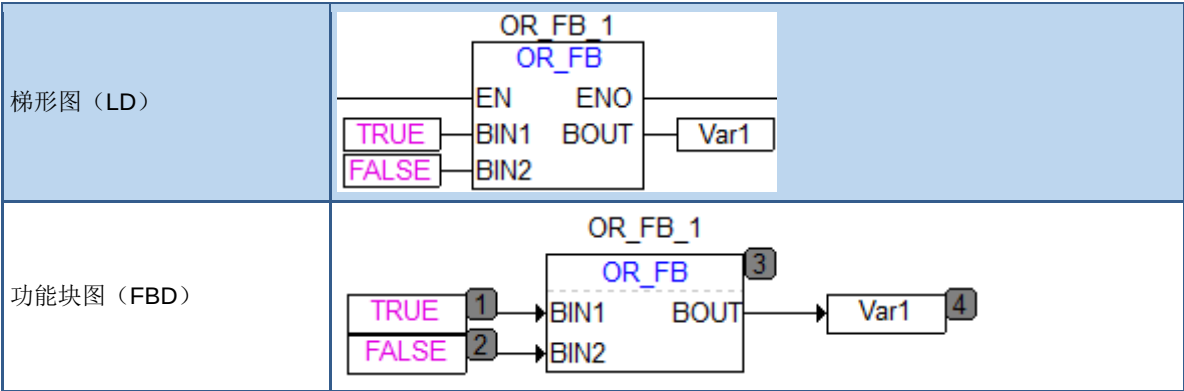
输入 1	输入 2	输出
0	0	0
0	1	1
1	0	1
1	1	1

2. 详细说明

对输入布尔变量完成或运算。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	OR_FB_1		OR_FB		G区
0002	Var1		BOOL	FALSE	G区
编程语言		程序			



2.2.2.7 NOT_FB (非)

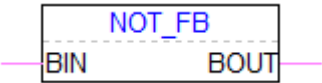


图 53 非

1. 参数说明

输入	说明	数据类型
BIN	输入	BOOL
输出		
BOUT	输出	BOOL

2. 详细说明

对输入布尔变量完成取反运算。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	NOT_FB_1		NOT_FB		G区
0002	Var1		BOOL	FALSE	G区

编程语言

程序

梯形图 (LD)	
功能块图 (FBD)	

2.2.2.8 AND3（3 取与）

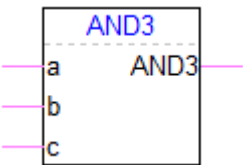


图 54 取与

1. 参数说明

输入	数据类型	描述	初始值
a	BOOL	变量 a	0
b	BOOL	变量 b	0
c	BOOL	变量 c	0

2. 指令使用举例

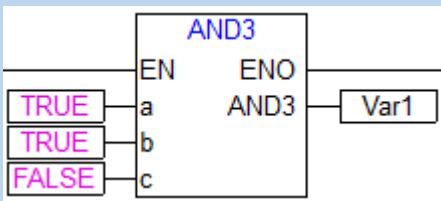
变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		BOOL	FALSE	G区

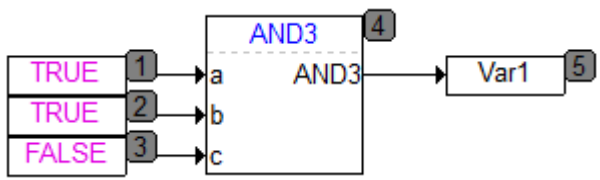
编程语言

程序

梯形图（LD）

A ladder logic diagram showing an AND3 block. The EN input is connected to a common rail. The ENO output is connected to a variable box labeled 'Var1'. The three inputs are labeled 'a', 'b', and 'c' in pink. The values 'TRUE', 'TRUE', and 'FALSE' are shown in pink boxes next to inputs 'a', 'b', and 'c' respectively.

功能块图（FBD）

A function block diagram showing an AND3 block. The three inputs are labeled 'a', 'b', and 'c' in pink. The values 'TRUE', 'TRUE', and 'FALSE' are shown in pink boxes next to inputs 'a', 'b', and 'c' respectively. The inputs are numbered 1, 2, and 3 in grey boxes. The output is connected to a variable box labeled 'Var1' which is numbered 5 in a grey box. The block is numbered 4 in a grey box.

2.2.2.9 OR3（3 取或）

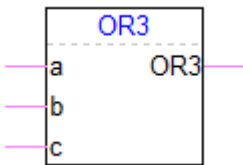
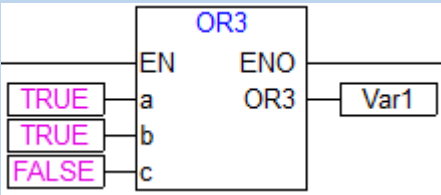
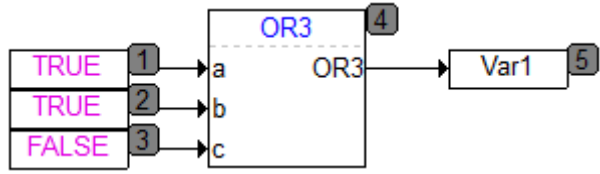


图 55 取或

1. 参数说明

输入	数据类型	描述	初始值
a	BOOL	变量 a	0
b	BOOL	变量 b	0
c	BOOL	变量 c	0

2. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		BOOL	FALSE	G区
编程语言		程序			
梯形图 (LD)					
功能块图 (FBD)					

2.2.3 比较运算

2.2.3.1 GT (大于)

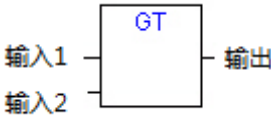


图 56 大于

1. 参数说明

输入	说明	数据类型
输入 1	输入变量	BYTE、WORD、DWORD、SINT、INT、DINT、REAL、LREAL、TIME、DATE、TOD、DT
输入 2	输入变量	BYTE、WORD、DWORD、SINT、INT、DINT、REAL、LREAL、TIME、DATE、TOD、DT

输出		
输出	输出变量	BOOL



- 比较运算两个输入端的数据类型必须相同。

2. 详细说明

当输入 1 大于输入 2 时输出为 TURE;

当输入 1 小于等于输入 2 时输出为 FALSE。

3. 指令使用举例

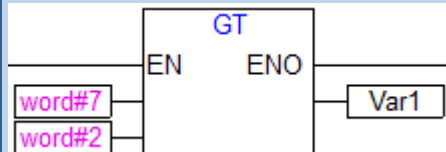
变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		BOOL	FALSE	G区

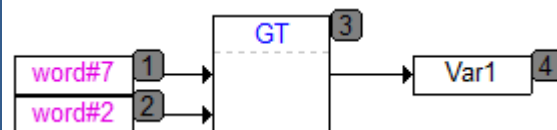
编程语言

程序

梯形图 (LD)



功能块图 (FBD)



2.2.3.2 LT (小于)

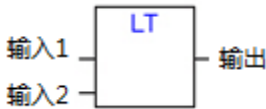


图 57 小于

1. 参数说明

输入	说明	数据类型
输入 1	输入	BYTE、WORD、DWORD、SINT、INT、DINT、REAL、LREAL、TIME、DATE、TOD、DT
输入 2	输入	BYTE、WORD、DWORD、SINT、INT、DINT、REAL、LREAL、TIME、DATE、TOD、DT
输出		
输出	输出	BOOL



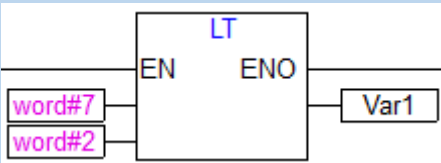
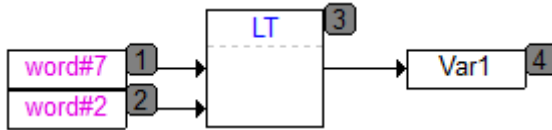
- 比较运算两个输入端的数据类型必须相同。

2. 详细说明

当输入 1 小于输入 2 时输出为 TRUE;

当输入 1 大于等于输入 2 时输出为 FALSE。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		BOOL	FALSE	G区
编程语言		程序			
梯形图 (LD)					
功能块图 (FBD)					

2.2.3.3 GE (大于等于)

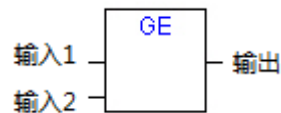


图 58 大于等于

1. 参数说明

输入	说明	数据类型
输入 1	输入	BYTE、WORD、DWORD、SINT、INT、DINT、REAL、LREAL、TIME、DATE、TOD、DT
输入 2	输入	BYTE、WORD、DWORD、SINT、INT、DINT、REAL、LREAL、TIME、DATE、TOD、DT
输出		
输出	输出	BOOL



- 比较运算两个输入端的数据类型必须相同。

2. 详细说明

当输入 1 大于等于输入 2 时输出为 TURE；
当输入 1 小于输入 2 时输出为 FALSE。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		BOOL	FALSE	G区

编程语言

程序

梯形图 (LD)

功能块图 (FBD)

2.2.3.4 LE（小于等于）

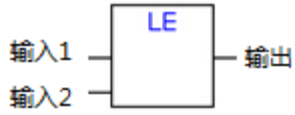


图 59 小于等于

1. 参数说明

输入	说明	数据类型
输入 1	输入	BYTE、WORD、DWORD、SINT、INT、DINT、REAL、LREAL、TIME、DATE、TOD、DT
输入 2	输入	BYTE、WORD、DWORD、SINT、INT、DINT、REAL、LREAL、TIME、DATE、TOD、DT
输出		
输出	输出	BOOL

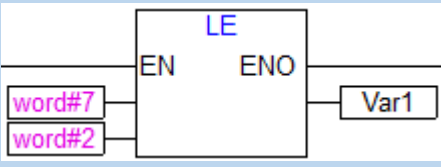
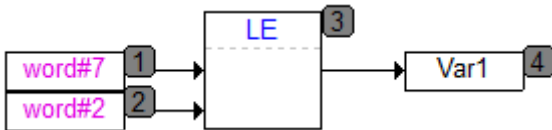
- 比较运算两个输入端的数据类型必须相同。

2. 详细说明

当输入 1 小于等于输入 2 时输出为 TURE；

当输入 1 大于输入 2 时输出为 FALSE。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		BOOL	FALSE	G区
编程语言		程序			
梯形图 (LD)					
功能块图 (FBD)					

2.2.3.5 EQ (等于)

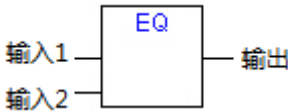


图 60 等于

1. 参数说明

输入	说明	数据类型
输入 1	输入	BOOL、BYTE、WORD、DWORD、SINT、INT、DINT、TIME、DATE、TOD、DT
输入 2	输入	BOOL、BYTE、WORD、DWORD、SINT、INT、DINT、TIME、DATE、TOD、DT
输出		
输出	输出	BOOL



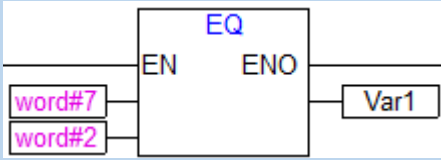
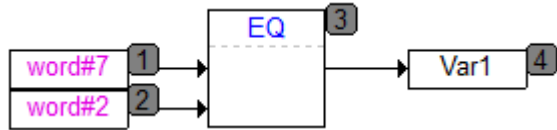
- 比较运算两个输入端的数据类型必须相同。

2. 详细说明

当输入 1 等于输入 2 时输出为 TRUE；

当输入 1 不等输入 2 时输出为 FALSE。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		BOOL	FALSE	G区
编程语言		程序			
梯形图 (LD)					
功能块图 (FBD)					

2.2.3.6 NE (不等于)

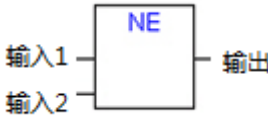


图 61 不等于

1. 参数说明

输入	说明	数据类型
输入 1	输入	BOOL、BYTE、WORD、DWORD、SINT、INT、DINT、TIME、DATE、TOD、DT
输入 2	输入	BOOL、BYTE、WORD、DWORD、SINT、INT、DINT、TIME、DATE、TOD、DT
输出		
输出	输出	BOOL

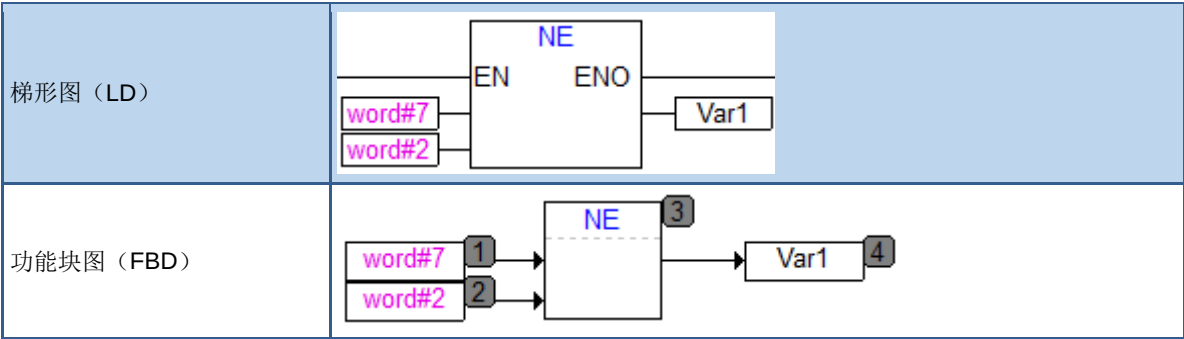
2. 详细说明

当输入 1 不等于输入 2 时输出为 TURE；

当输入 1 等于输入 2 时输出为 FALSE。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		BOOL	FALSE	G区
编程语言		程序			



2.2.3.7 GT_DINT_FB (大于 (长整型))

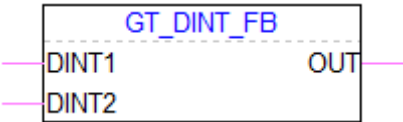


图 62 大于

1. 参数说明

输入	说明	数据类型
DINT1	输入变量 1	DINT
DINT2	输入变量 2	DINT
输出		
OUT	输出变量	BOOL

2. 详细说明

当输入 1 大于输入 2 时输出为 TURE；
当输入 1 小于等于输入 2 时输出为 FALSE。

3. 指令使用举例

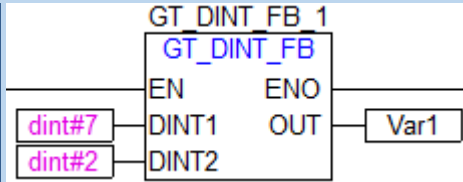
变量定义

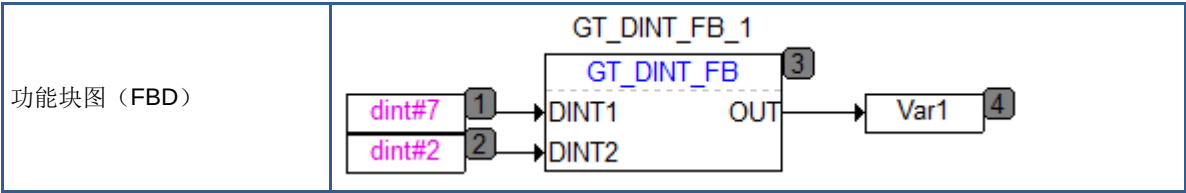
序号	变量名	变量说明	变量类型	初始值	区域
0001	GT_DINT_FB_1		GT_DINT_FB		G区
0002	Var1		BOOL	FALSE	G区

编程语言

程序

梯形图 (LD)





2.2.3.8 LT_DINT_FB（小于（长整型））

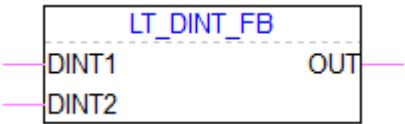


图 63 小于

1. 参数说明

输入	说明	数据类型
DINT1	输入变量 1	DINT
DINT2	输入变量 2	DINT
输出		
OUT	输出变量	BOOL

2. 详细说明

当输入 1 小于输入 2 时输出为 TURE；
当输入 1 大于等于输入 2 时输出为 FALSE。

3. 指令使用举例

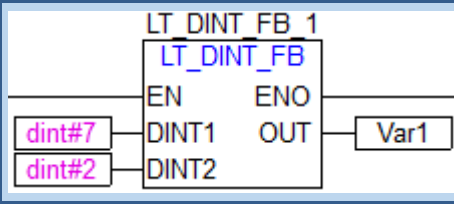
变量定义

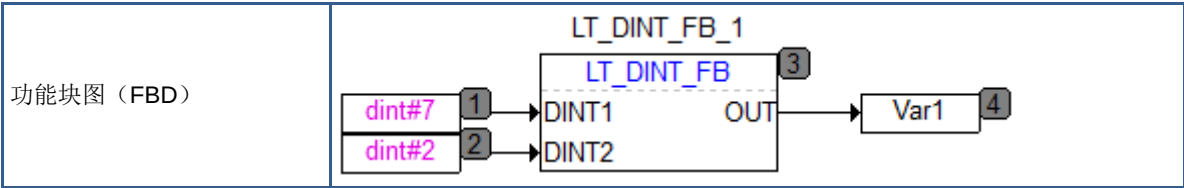
序号	变量名	变量说明	变量类型	初始值	区域
0001	LT_DINT_FB_1		LT_DINT_FB		G区
0002	Var1		BOOL	FALSE	G区

编程语言

程序

梯形图（LD）





2.2.3.9 GE_DINT_FB（大于等于（长整型））

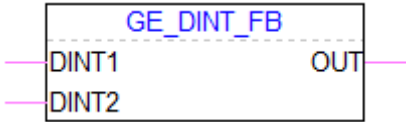


图 64 大于等于

1. 参数说明

输入	说明	数据类型
DINT1	输入变量 1	DINT
DINT2	输入变量 2	DINT
输出		
OUT	输出变量	BOOL

2. 详细说明

当输入 1 大于等于输入 2 时输出为 TURE；

当输入 1 小于输入 2 时输出为 FALSE。

3. 指令使用举例

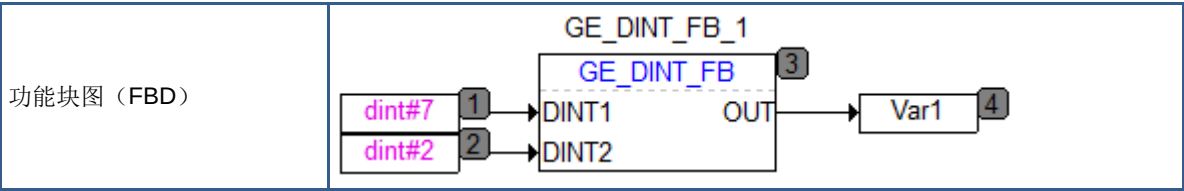
变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	GE_DINT_FB_1		GE_DINT_FB		G区
0002	Var1		BOOL	FALSE	G区

编程语言

程序

梯形图（LD）



2.2.3.10 LE_DINT_FB（小于等于（长整型））

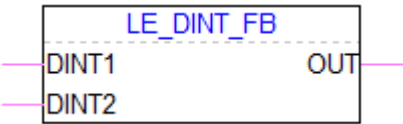


图 65 小于等于

1. 参数说明

输入	说明	数据类型
DINT1	输入变量 1	DINT
DINT2	输入变量 2	DINT
输出		
OUT	输出变量	BOOL

2. 详细说明

当输入 1 小于等于输入 2 时输出为 TURE；
当输入 1 大于输入 2 时输出为 FALSE。

3. 指令使用举例

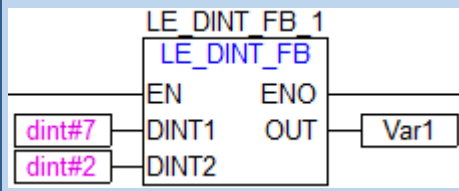
变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	LE_DINT_FB_1		LE_DINT_FB		G区
0002	Var1		BOOL	FALSE	G区

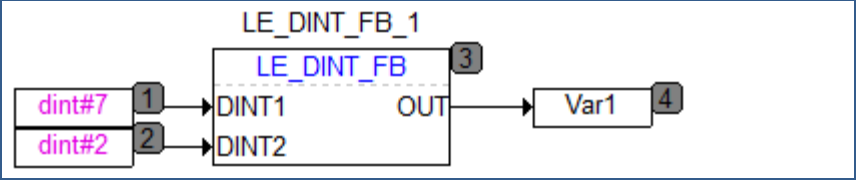
编程语言

程序

梯形图（LD）



功能块图（FBD）



2.2.3.11 EQ_DINT_FB（等于（长整型））

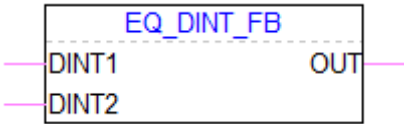


图 66 等于

1. 参数说明

输入	说明	数据类型
DINT1	输入变量 1	DINT
DINT2	输入变量 2	DINT
输出		
OUT	输出变量	BOOL

2. 详细说明

当输入 1 等于输入 2 时输出为 TURE；
 当输入 1 不等输入 2 时输出为 FALSE。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	EQ_DINT_FB_1		EQ_DINT_FB		G区
0002	Var1		BOOL	FALSE	G区

编程语言

程序

梯形图（LD）

功能块图（FBD）

2.2.3.12 GT_REAL_FB（大于（实型））

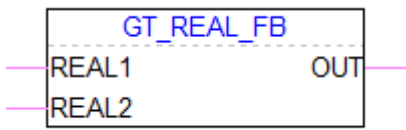


图 67 大于

1. 参数说明

输入	说明	数据类型
REAL1	输入变量	REAL
REAL2	输入变量	REAL
输出		
OUT	输出变量	BOOL

2. 详细说明

当输入 1 大于输入 2 时输出为 TURE；
当输入 1 小于等于输入 2 时输出为 FALSE。

3. 指令使用举例

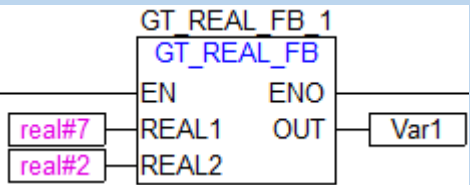
变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	GT_REAL_FB_1		GT_REAL_FB		G区
0002	Var1		BOOL	FALSE	G区

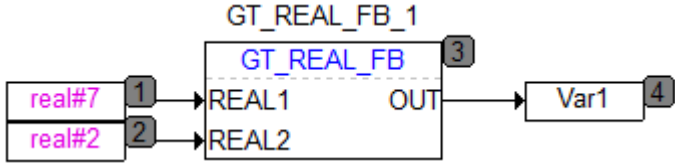
编程语言

程序

梯形图（LD）



功能块图（FBD）



2.2.3.13 LT_REAL_FB（小于（实型））

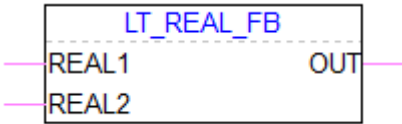


图 68 小于

1. 参数说明

输入	说明	数据类型
REAL1	输入变量	REAL
REAL2	输入变量	REAL
输出		
OUT	输出变量	BOOL

2. 详细说明

当输入 1 小于输入 2 时输出为 TURE；
当输入 1 大于等于输入 2 时输出为 FALSE。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	LT_REAL_FB_1		LT_REAL_FB		G区
0002	Var1		BOOL	FALSE	G区

编程语言

程序

梯形图（LD）

功能块图（FBD）

2.2.3.14 GE_REAL_FB（大于等于（实型））

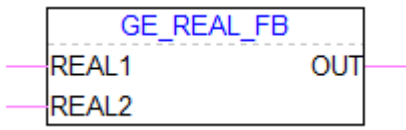


图 69 大于等于

1. 参数说明

输入	说明	数据类型
REAL1	输入变量	REAL
REAL2	输入变量	REAL
输出		
OUT	输出变量	BOOL

2. 详细说明

当输入 1 大于等于输入 2 时输出为 TURE;

当输入 1 小于输入 2 时输出为 FALSE。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	GE_REAL_FB_1		GE_REAL_FB		G区
0002	Var1		BOOL	FALSE	G区

编程语言

程序

梯形图 (LD)

The diagram shows a ladder logic network. It starts with a normally open contact labeled 'real#7'. This is followed by another normally open contact labeled 'real#2'. The output of this series connection is connected to the 'OUT' port of a block labeled 'GE_REAL_FB'. The block also has an 'EN' (enable) input and an 'ENO' (enable output) port. The block is also labeled 'GE_REAL_FB_1' at the top.

功能块图 (FBD)

The diagram shows a function block diagram. Two inputs, 'real#7' and 'real#2', are connected to the 'REAL1' and 'REAL2' ports of a block labeled 'GE_REAL_FB'. The output of the block, 'OUT', is connected to a variable 'Var1'. The block is also labeled 'GE_REAL_FB_1' at the top.

2.2.3.15 LE_REAL_FB（小于等于（实型））

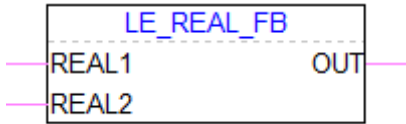


图 70 小于等于

1. 参数说明

输入	说明	数据类型
REAL1	输入变量	REAL
REAL2	输入变量	REAL
输出		
OUT	输出变量	BOOL

2. 详细说明

当输入 1 小于等于输入 2 时输出为 TURE；
当输入 1 大于输入 2 时输出为 FALSE。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	LE_REAL_FB_1		LE_REAL_FB		G区
0002	Var1		BOOL	FALSE	G区

编程语言

程序

梯形图（LD）

功能块图（FBD）

2.2.3.16 EQ_REAL_FB（等于（实型））

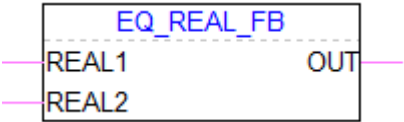


图 71 等于

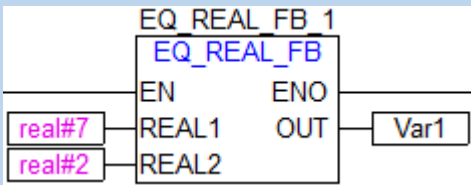
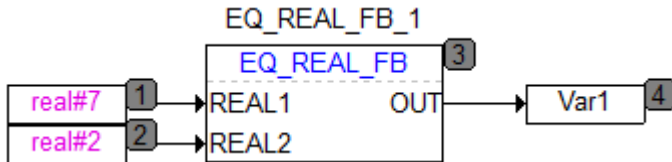
1. 参数说明

输入	说明	数据类型
REAL1	输入变量	REAL
REAL2	输入变量	REAL
输出		
OUT	输出变量	BOOL

2. 详细说明

当输入 1 等于输入 2 时输出为 TURE；
当输入 1 不等输入 2 时输出为 FALSE。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	EQ_REAL_FB_1		EQ_REAL_FB ▾		G区 ▾
0002	Var1		BOOL ▾	FALSE ▾	G区 ▾
编程语言		程序			
梯形图（LD）					
功能块图（FBD）					

2.2.4 选择运算

2.2.4.1 SEL（二选一）

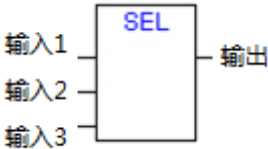


图 72 二选一

1. 参数说明

输入	说明	数据类型
输入 1	选择变量	BOOL
输入 2	被选变量	任意
输入 3	被选变量	任意
输出		
输出	当输入 1 为 FALSE 时输入 2 作为输出；当输入 1 为 TRUE 时输入 3 作为输出。	任意，但是在运算中需保证输入的类型与输出的类型匹配。

2. 详细说明

输入 1 为 FALSE 时将输入 2 的值输出，输入 1 为 TRUE 时将输入 3 的值输出。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		WORD	0	G区
编程语言		程序			
梯形图（LD）					
功能块图（FBD）					

2.2.4.2 MAX（取最大值）



图 73 取最大值

1. 参数说明

输入	说明	数据类型
输入 1	输入变量	任意，BOOL 类型除外
输入 2	输入变量	任意，BOOL 类型除外
输出		
输出	输出变量	任意，但是在运算中需保证输入的类型与输出的类型匹配。

2. 详细说明

将两输入的中最大的一个值输出。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		WORD	0	G区
编程语言		程序			
梯形图（LD）					
功能块图（FBD）					

2.2.4.3 MIN（取最小值）



图 74 取最小值

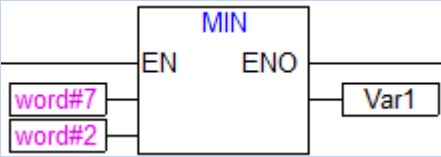
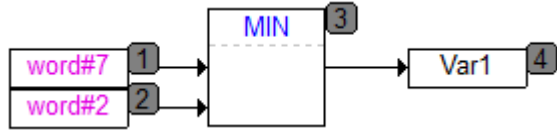
1. 参数说明

输入	说明	数据类型
输入 1	输入	任意，BOOL 类型除外
输入 2	输入	任意，BOOL 类型除外
输出		
输出	输出变量	任意，但是在运算中需保证输入的类型与输出的类型匹配

2. 详细说明

将两输入中最小的一个输出。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		WORD	0	G区
编程语言		程序			
梯形图 (LD)					
功能块图 (FBD)					

2.2.5 移位运算

2.2.5.1 SHL (左移)

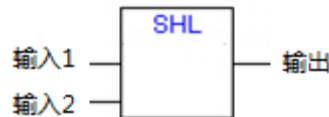


图 75 左移

1. 参数说明

输入	说明	数据类型
输入 1	被操作变量	DWORD、DINT
输入 2	移位位数	BYTE

输出		
输出	结果	DWORD、DINT

2. 详细说明

对于输入变量按指定位向左移指定位，对右边空位补 0。

3. 指令使用举例

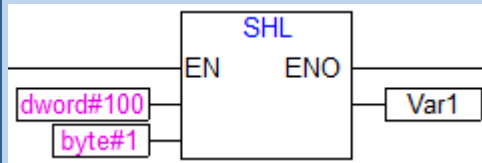
变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		DWORD	0	G区

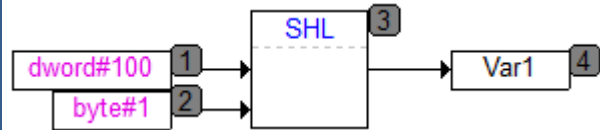
编程语言

程序

梯形图（LD）



功能块图（FBD）



2.2.5.2 SHR（右移）

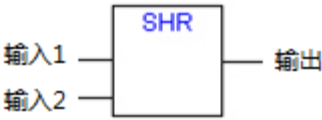


图 76 右移

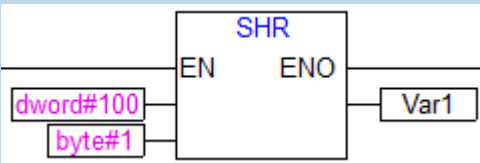
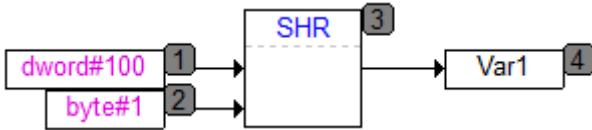
1. 参数说明

输入	说明	数据类型
输入 1	被操作变量	DWORD、DINT
输入 2	移位位数	BYTE
输出		
输出	结果	DWORD、DINT

2. 详细说明

对于输入变量按指定位向右移指定位，对于左边空位补 0。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		DWORD	0	G区
编程语言		程序			
梯形图 (LD)					
功能块图 (FBD)					

2.2.5.3 ROL (循环左移)

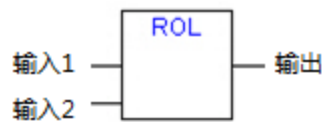


图 77 循环左移

1. 参数说明

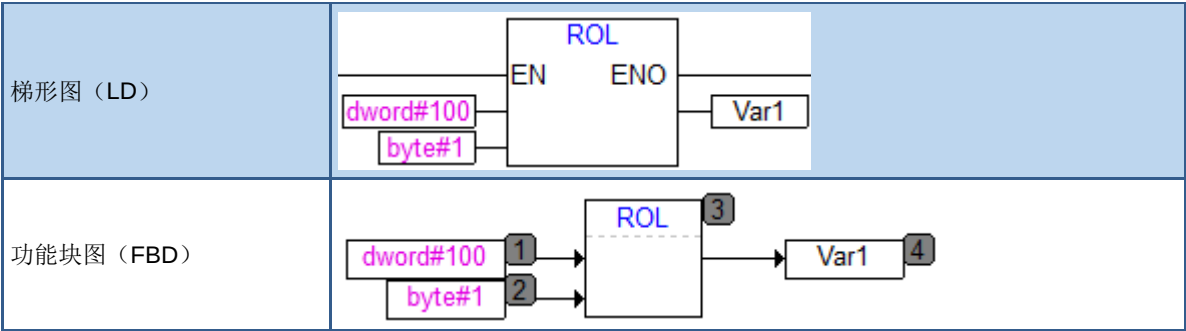
输入	说明	数据类型
输入 1	被操作变量	DWORD、DINT
输入 2	移位位数	BYTE
输出		
输出	结果	DWORD、DINT

2. 详细说明

对于输入变量按指定位向左移指定位，把左边位插到右边。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		DWORD	0	G区
编程语言		程序			



2.2.5.4 ROR (循环右移)

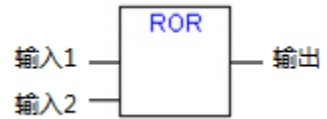


图 78 循环右移

1. 参数说明

输入	说明	数据类型
输入 1	被操作变量	DWORD、DINT
输入 2	移位位数	BYTE
输出		
输出	结果	DWORD、DINT

2. 详细说明

对于输入变量按指定位向右移指定位，把右边位插到左边。

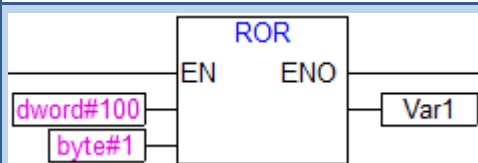
3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		DWORD	0	G区

编程语言

程序

梯形图 (LD)	
功能块图 (FBD)	

2.2.6 数据类型转换

2.2.6.1 BYTE_TO (BYTE 类型转换)

1. 功能概述

BYTE 转换为其它类型。

2. 参数说明

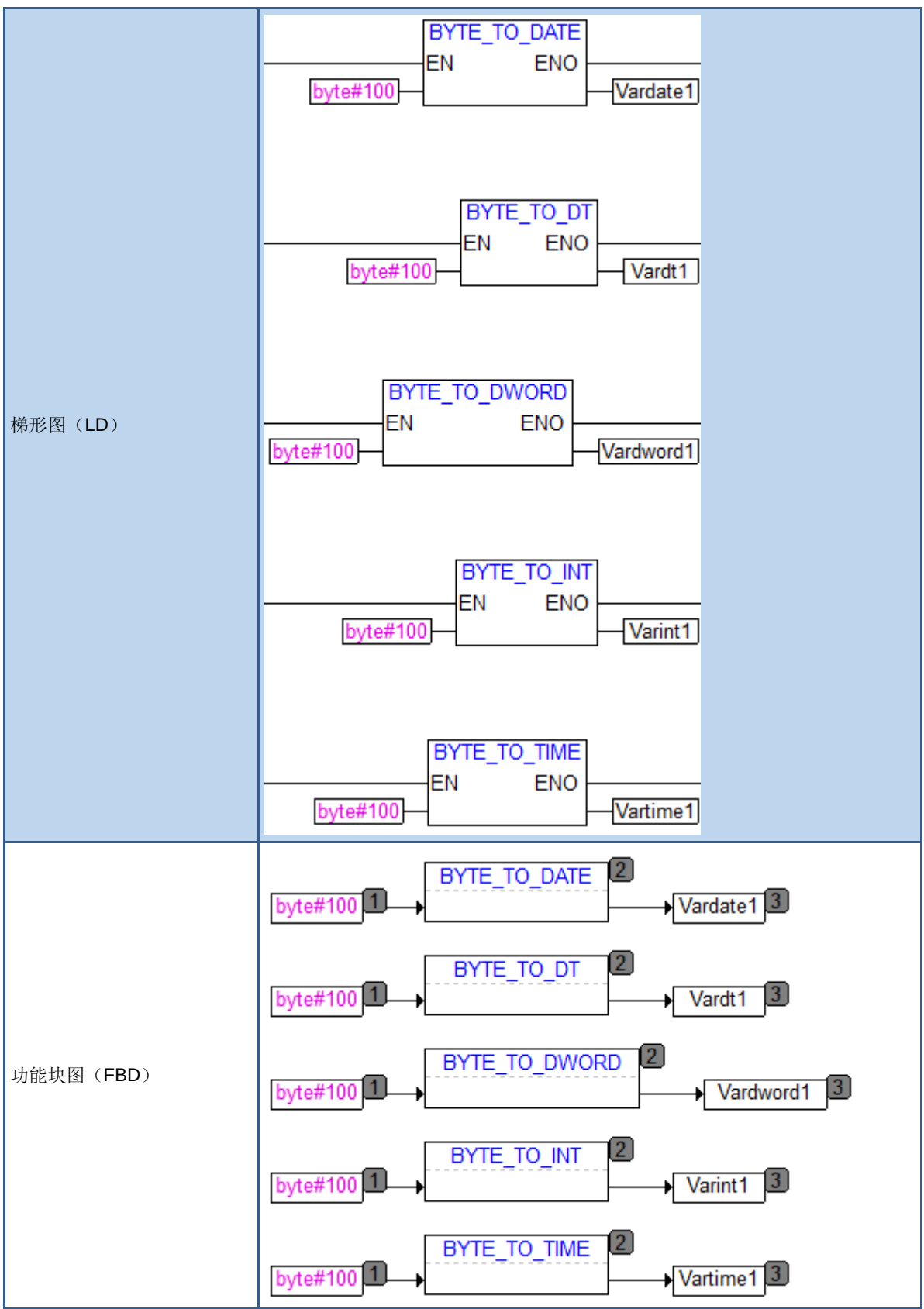
输入	说明	数据类型
输入	输入变量	BYTE
输出		
输出	输出变量	DATE、DINT、DT、DWORD、INT、SINT、TIME、TOD、WORD

3. 详细说明

从字节型转换为其他类型 (BYTE_TO_XXX)，XXX 表示其他数据类型，完成字节型到其它类型的转换。

4. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Vardate1		DATE	D#1970-01-01	G区
0002	Vardt1		DT	DT#1970-01-0...	G区
0003	Vardword1		DWORD	0	G区
0004	Varint1		INT	0	G区
0005	Vartime1		TIME	T#0MS	G区
编程语言		程序			



2.2.6.2 WORD_TO (WORD 类型转换)

1. 功能概述

WORD 转换为其它类型。

2. 参数说明

输入	说明	数据类型
输入	输入变量	WORD
输出		
输出	输出变量	BYTE、DATE、DINT、DT、DWORD、INT、SINT、TIME、TOD

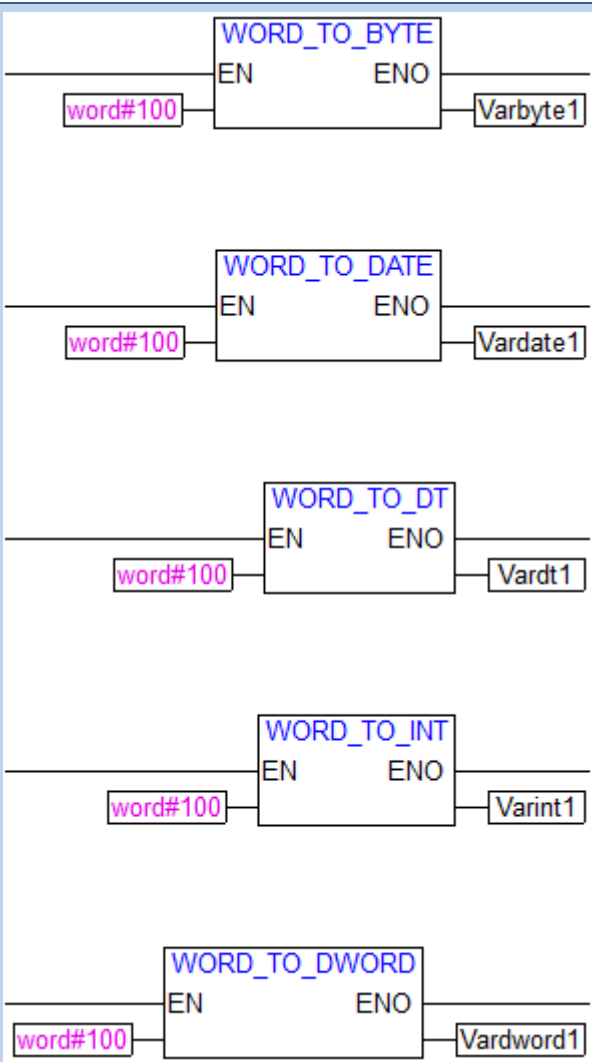
3. 详细说明

从字类型转换为其他类型 (WORD_TO_xxx)，xxx 表示其他数据类型，完成字型到其它类型的转换。

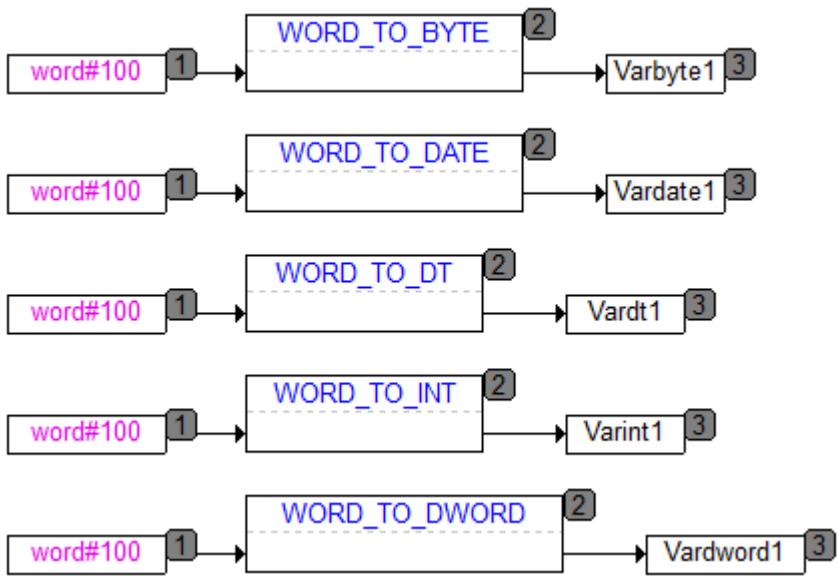
4. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Vardate1		DATE	D#1970-01-01	G区
0002	Vardt1		DT	DT#1970-01-0...	G区
0003	Vardword1		DWORD	0	G区
0004	Varint1		INT	0	G区
0005	Varbyte1		BYTE	0	G区
编程语言		程序			

梯形图 (LD)



功能块图 (FBD)



2.2.6.3 DWORD_TO (DWORD 类型转换)

1. 功能概述

DWORD 转换为其它类型。

2. 参数说明

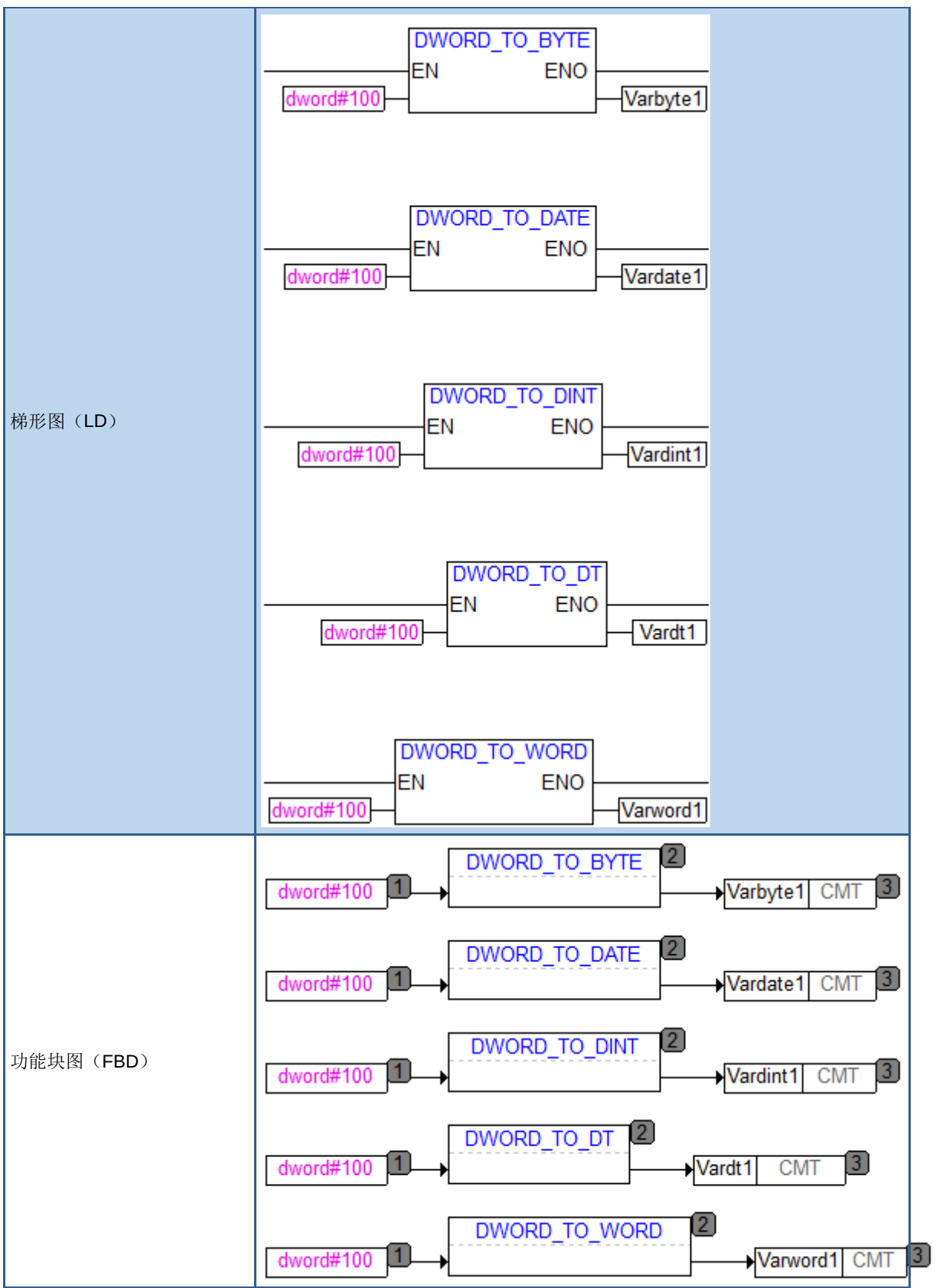
输入	说明	数据类型
输入	输入变量	DWORD
输出		
输出	输出变量	BYTE、DATE、DINT、DT、INT、LREAL、REAL、SINT、TIME、TOD、WORD

3. 详细说明

双字类型转换为其他类型 (DWORD_TO_xxx)，xxx 表示其他数据类型，完成双字型到其它类型的转换。

4. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Vardate1		DATE ▼	D#1970-01-01	G区 ▼
0002	Vardt1		DT ▼	DT#1970-01-01-0...	G区 ▼
0003	Varbyte1		BYTE ▼	0	G区 ▼
0004	Vardint1		DINT ▼	0	G区 ▼
0005	Varword1		WORD ▼	0	G区 ▼
编程语言		程序			



2.2.6.4 SINT_TO (SINT 类型转换)

1. 功能概述

SINT 转换为其它类型。

2. 参数说明

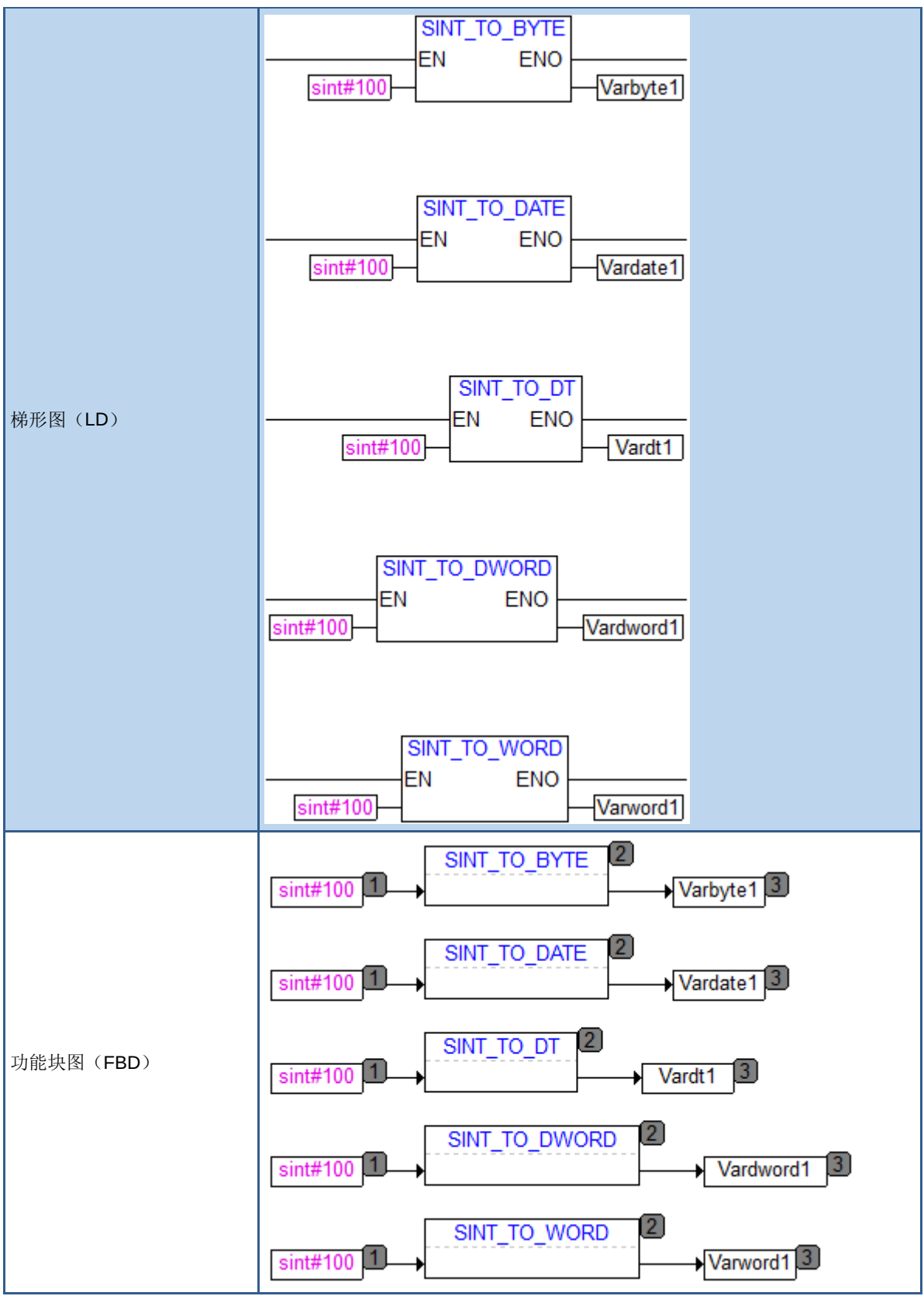
输入	说明	数据类型
输入	输入变量	SINT
输出		
输出	输出变量	BYTE、DATE、DINT、DT、DWORD、INT、TIME、TOD、WORD

3. 详细说明

短整型转换为其他类型 (SINT_TO_xxx)，xxx 表示其他数据类型，完成短整型到其它类型的转换。

4. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Vardate1		DATE	D#1970-01-01	G区
0002	Vardt1		DT	DT#1970-01-0...	G区
0003	Varbyte1		BYTE	0	G区
0004	Varword1		WORD	0	G区
0005	Vardword1		DWORD	0	G区
编程语言		程序			



2.2.6.5 INT_TO（INT 类型转换）

1. 功能概述

INT 转换为其它类型。

2. 参数说明

输入	说明	数据类型
输入	输入变量	INT
输出		
输出	输出变量	BYTE、DATE、DINT、DT、DWORD、SINT、TIME、TOD、WORD

3. 详细说明

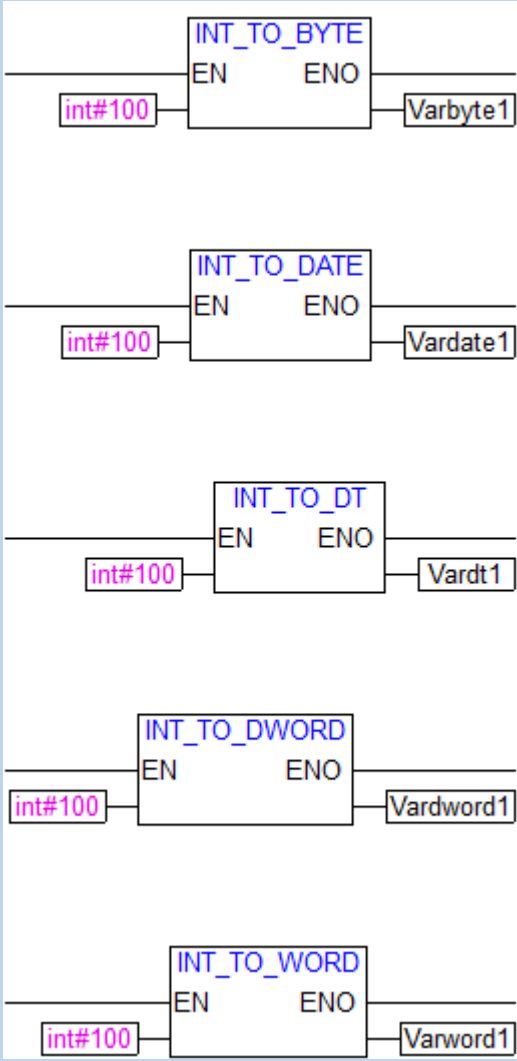
从整型转换为其它类型算法（INT_TO_xxx），xxx 表示其它数据类型，完成的功能：

- 完成整型到其它类型的转换；
- 转换值超限时，忽略这个数的第一个字节。

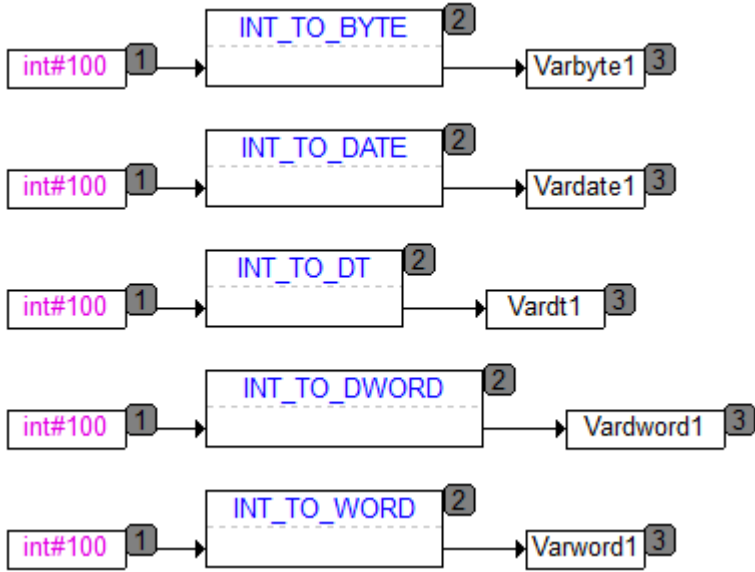
4. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Vardate1		DATE ▼	D#1970-01-01	G区 ▼
0002	Vardt1		DT ▼	DT#1970-01-0...	G区 ▼
0003	Varbyte1		BYTE ▼	0	G区 ▼
0004	Varword1		WORD ▼	0	G区 ▼
0005	Vardword1		DWORD ▼	0	G区 ▼
编程语言		程序			

梯形图（LD）



功能块图（FBD）



2.2.6.6 DINT_TO (DINT 类型转换)

1. 功能概述

DINT 转换为其它类型。

2. 参数说明

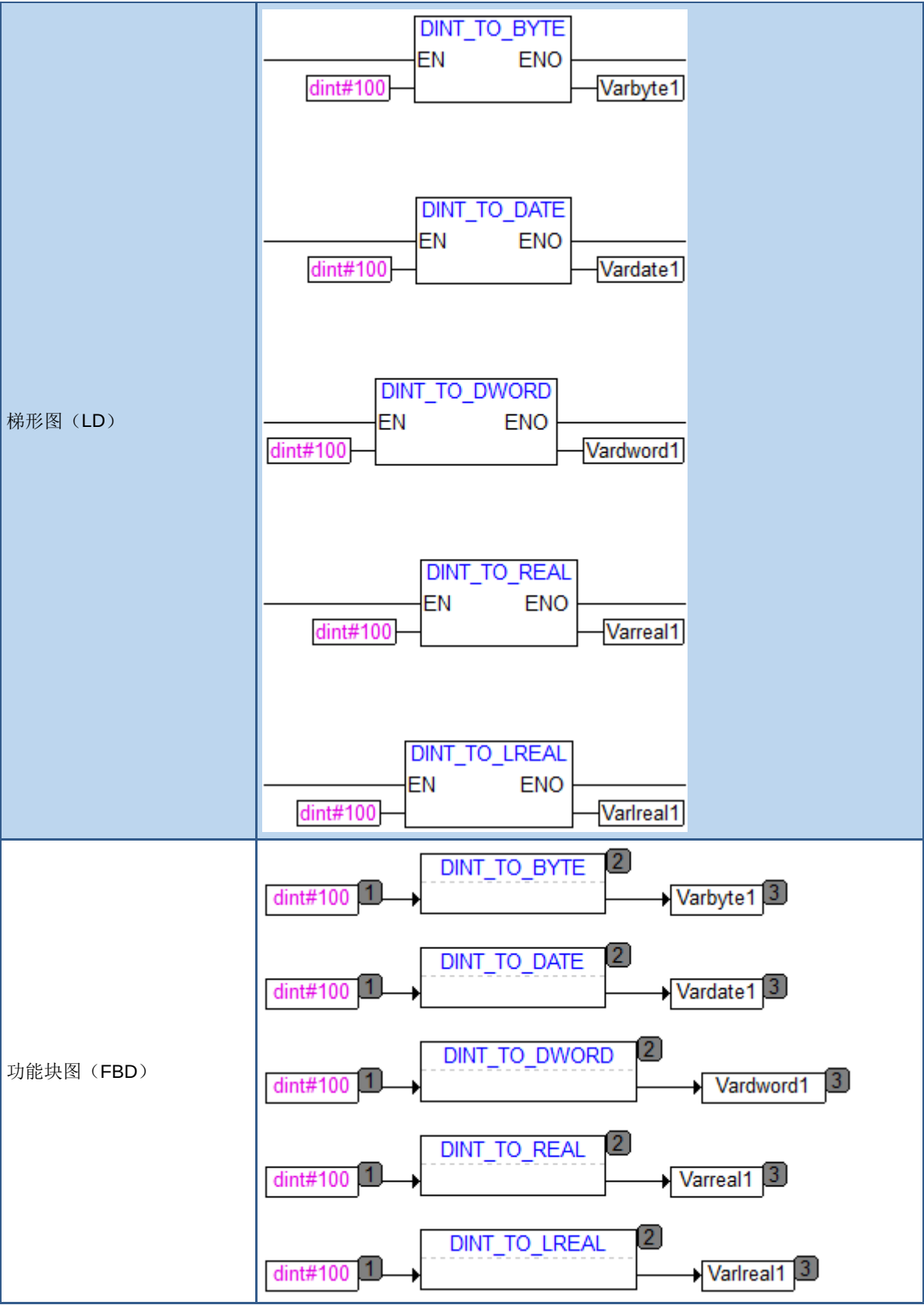
输入	说明	数据类型
输入	输入变量	DINT
输出		
输出	输出变量	BYTE、DATE、DT、DWORD、INT、LREAL、REAL、SINT、TIME、TOD、WORD

3. 详细说明

从双整型转换为其它类型算法 (DINT_TO_xxx)，xxx 表示其它数据类型，完成双整型到其它类型的转换。

4. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Vardate1		DATE	D#1970-01-01	G区
0002	Varbyte1		BYTE	0	G区
0003	Vardword1		DWORD	0	G区
0004	Varreal1		REAL	0	G区
0005	Varlreal1		LREAL	0	G区
编程语言		程序			



2.2.6.7 REAL/LREAL_TO（REAL/LREAL 类型转换）

1. 功能概述

REAL/LREAL 转换为其它类型。

2. 参数说明

输入	说明	数据类型
输入	输入变量	REAL/LREAL
输出		
输出	输出变量	DATE、DINT、DT、DWORD、TIME、TOD 另外 REAL 的输入还可以转换为 LREAL、LREAL 的输入还可以转换为 REAL

3. 详细说明

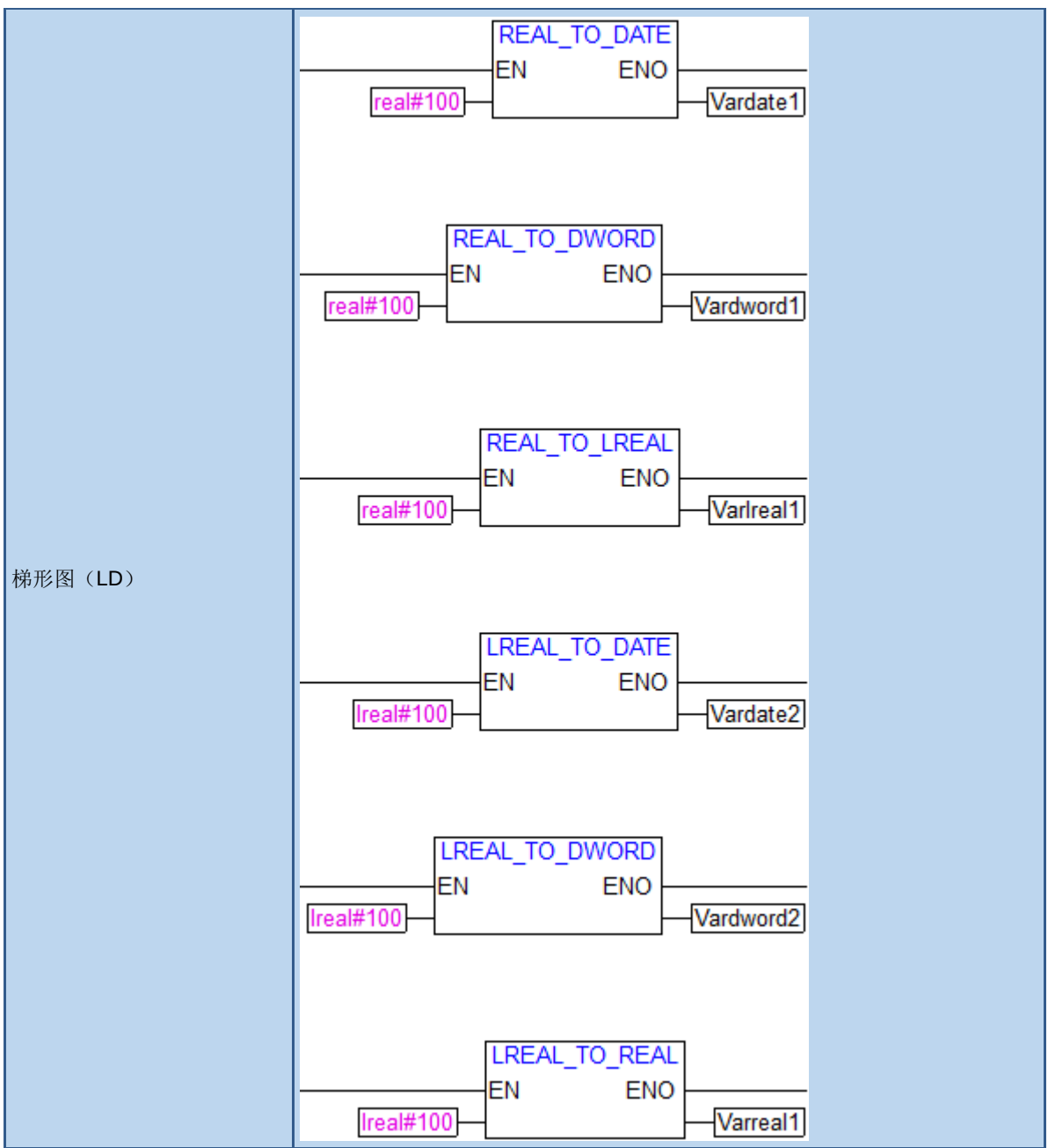
REAL/LREAL 转换为其它类型算法（REAL_TO/LREAL_TO_xxx），将值取整，舍弃小数部分，然后转成新的变量类型，xxx 表示其它数据类型。

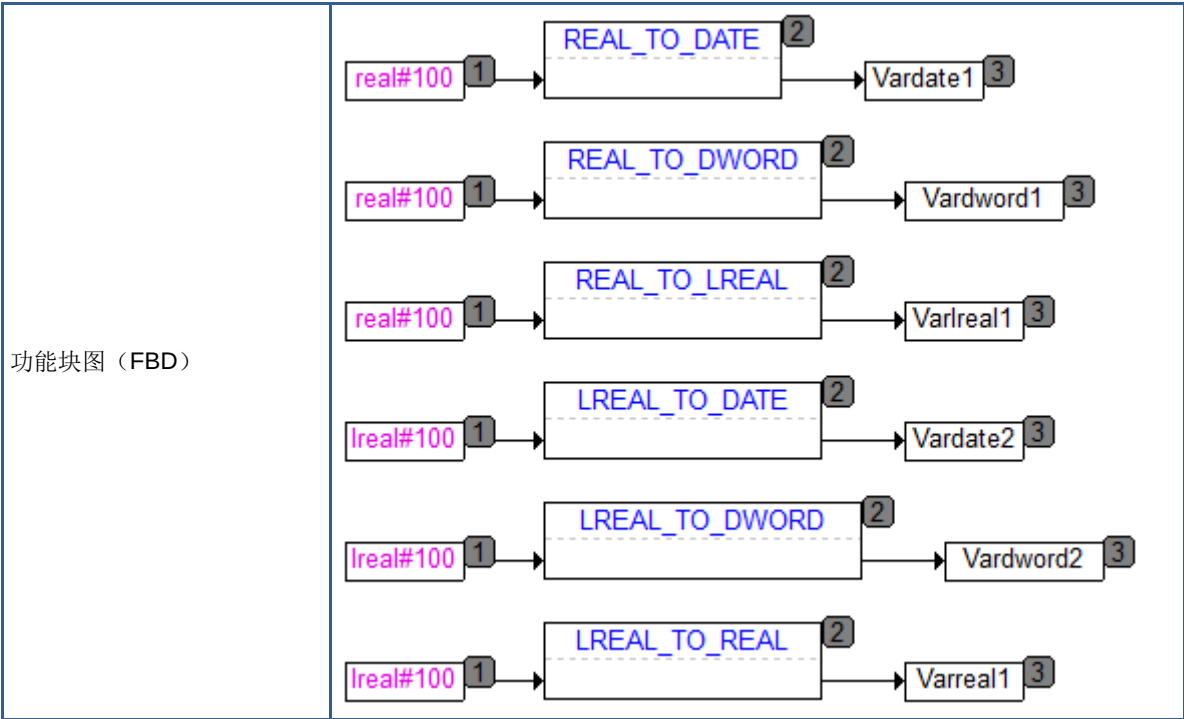


- 当从复杂数据类型转为简单数据类型时，有可能丢失信息。

4. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Vardate1		DATE	D#1970-01-01	G区
0002	Vardword1		DWORD	0	G区
0003	Varreal1		REAL	0	G区
0004	Varlreal1		LREAL	0	G区
0005	Vardate2		DATE	D#1970-01-01	G区
0006	Vardword2		DWORD	0	G区
编程语言		程序			





2.2.6.8 TIME_TO（TIME 类型转换）

1. 功能概述

TIME 转换为其它类型。

2. 参数说明

输入	说明	数据类型
输入	输入变量	TIME
输出		
输出	输出变量	BYTE、DINT、DWORD、INT、REAL、LREAL、SINT、WORD

3. 详细说明

TIME 转换为其它类型算法（TIME_TO_xxx），xxx 表示其它数据类型，实现 TIME 向其它类型数据转换。

4. 指令使用举例

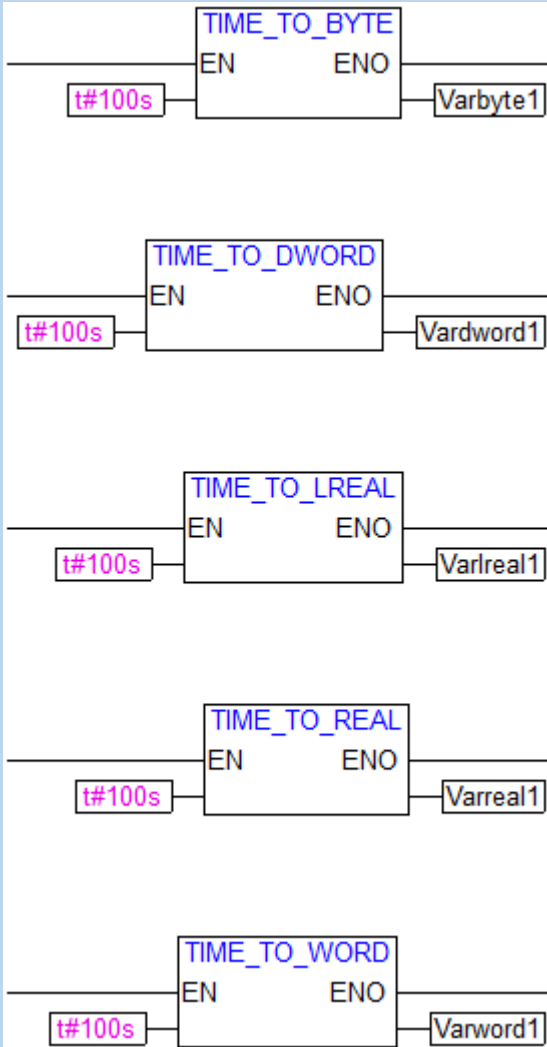
变量定义

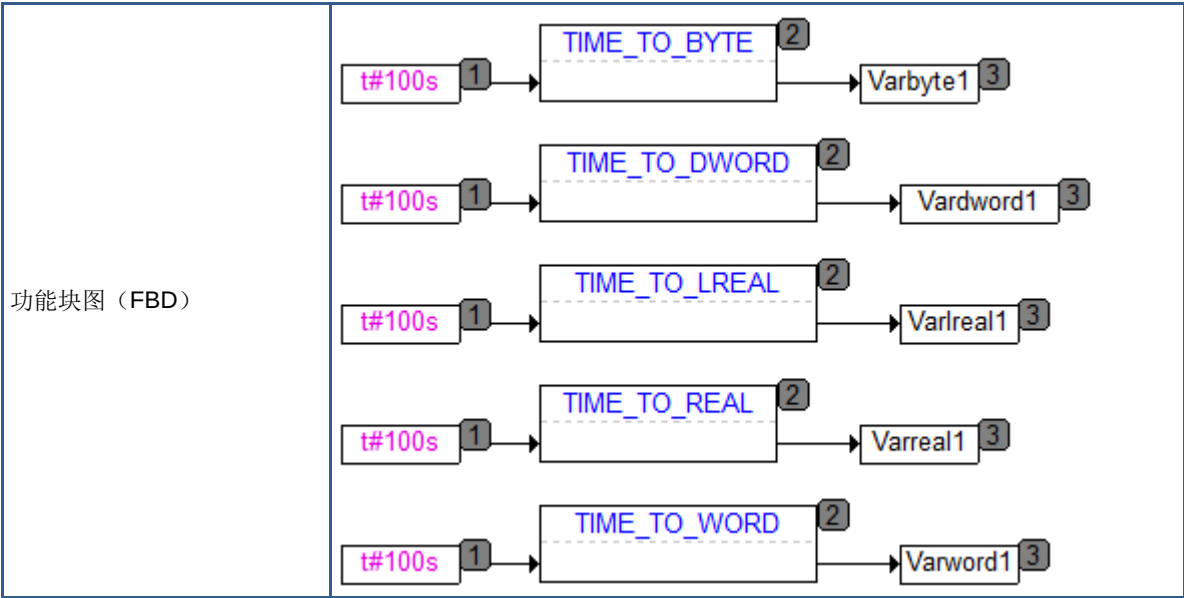
序号	变量名	变量说明	变量类型	初始值	区域
0001	Varbyte1		BYTE ▾	0	G区 ▾
0002	Vardword1		DWORD ▾	0	G区 ▾
0003	Varlreal1		LREAL ▾	0	G区 ▾
0004	Varreal1		REAL ▾	0	G区 ▾
0005	Varword1		WORD ▾	0	G区 ▾

编程语言

程序

梯形图 (LD)





2.2.6.9 DATE_TO（DATE 类型转换）

1. 功能概述

DATE 转换为其它类型。

2. 参数说明

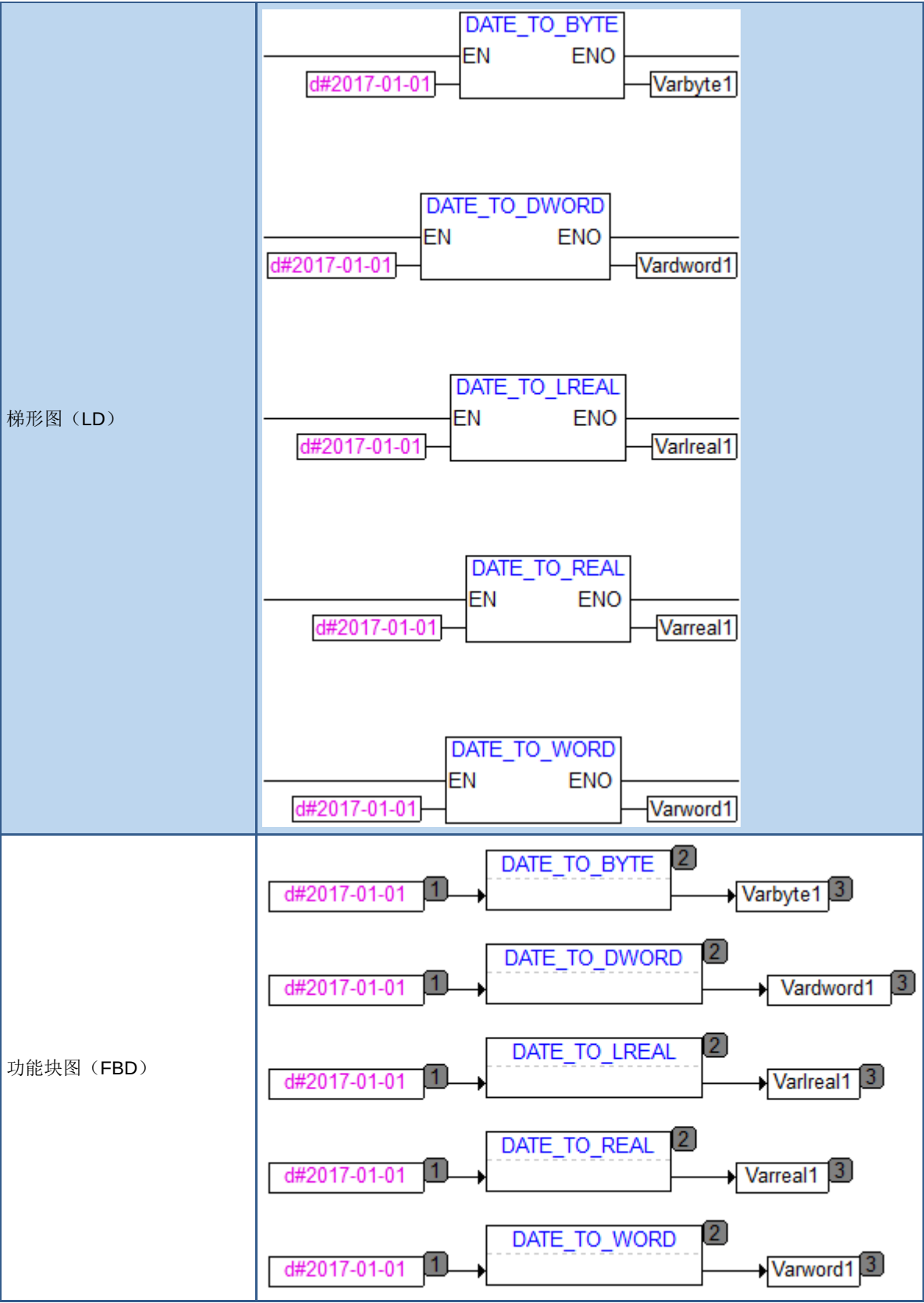
输入	说明	数据类型
输入	输入变量	DATE
输出		
输出	输出变量	BYTE、DINT、DWORD、INT、LREAL、REAL、SINT、WORD

3. 详细说明

DATE 转换为它类型算法（DATE_TO_xxx），xxx 表示其它数据类型，实现 DATE 向其它类型的转换。

4. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Varbyte1		BYTE	0	G区
0002	Vardword1		DWORD	0	G区
0003	Varlreal1		LREAL	0	G区
0004	Varreal1		REAL	0	G区
0005	Varword1		WORD	0	G区
编程语言		程序			



2.2.6.10 DT_TO (DT 类型转换)

1. 功能概述

DT 转换为其它类型。

2. 参数说明

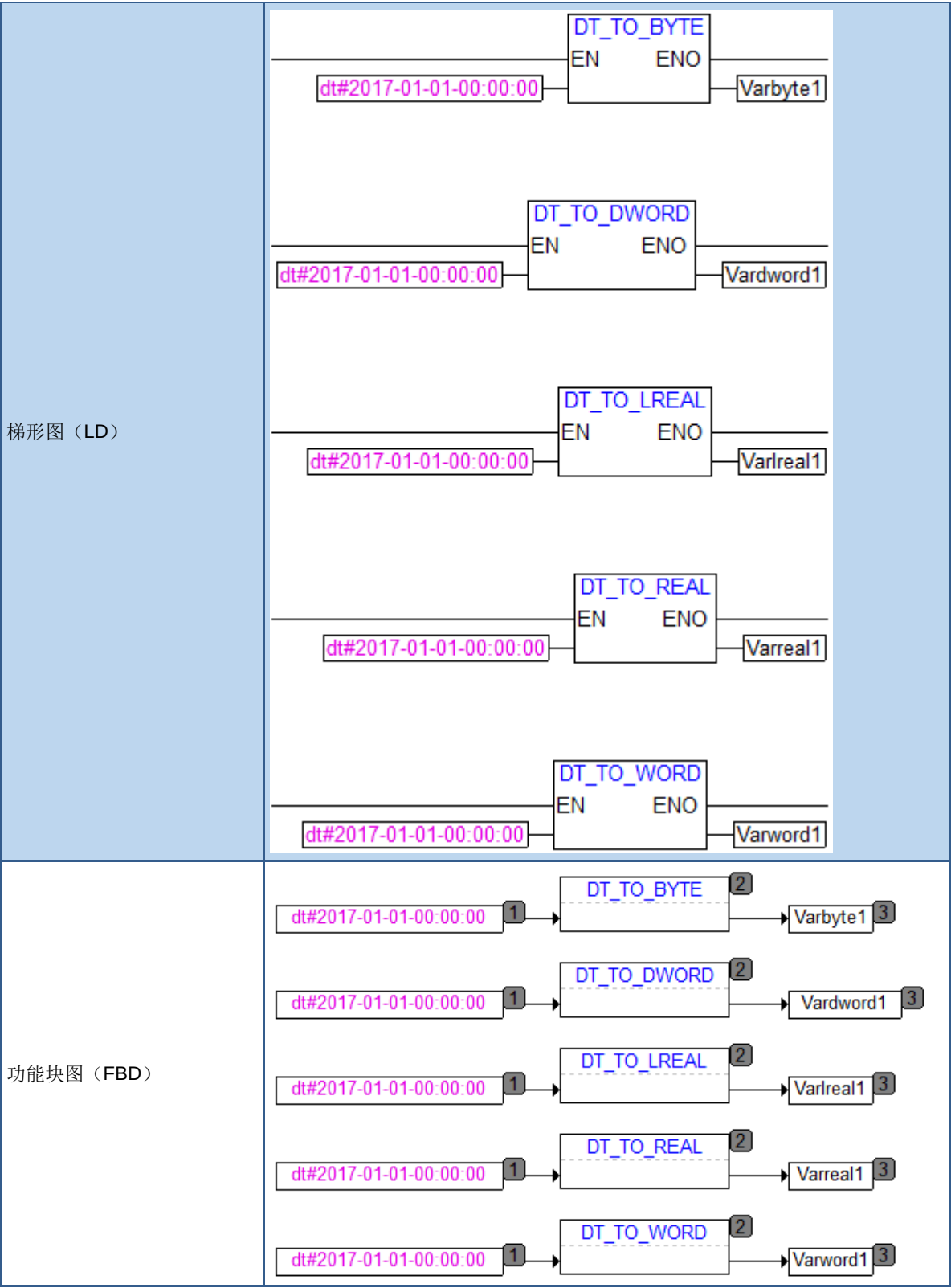
输入	说明	数据类型
输入	输入变量	DT
输出		
输出	输出变量	BYTE、DINT、DWORD、INT、LREAL、REAL、SINT、WORD

3. 详细说明

DT 类型转换为其他类型 (DT_TO_xxx)，xxx 表示其他数据类型，完成 DT 到其它类型的转换。

4. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Varbyte1		BYTE ▾	0	G区 ▾
0002	Vardword1		DWORD ▾	0	G区 ▾
0003	Varlreal1		LREAL ▾	0	G区 ▾
0004	Varreal1		REAL ▾	0	G区 ▾
0005	Varword1		WORD ▾	0	G区 ▾
编程语言		程序			



2.2.6.11 TOD_TO (TOD 类型转换)

1. 功能概述

TOD 转换为其它类型。

2. 参数说明

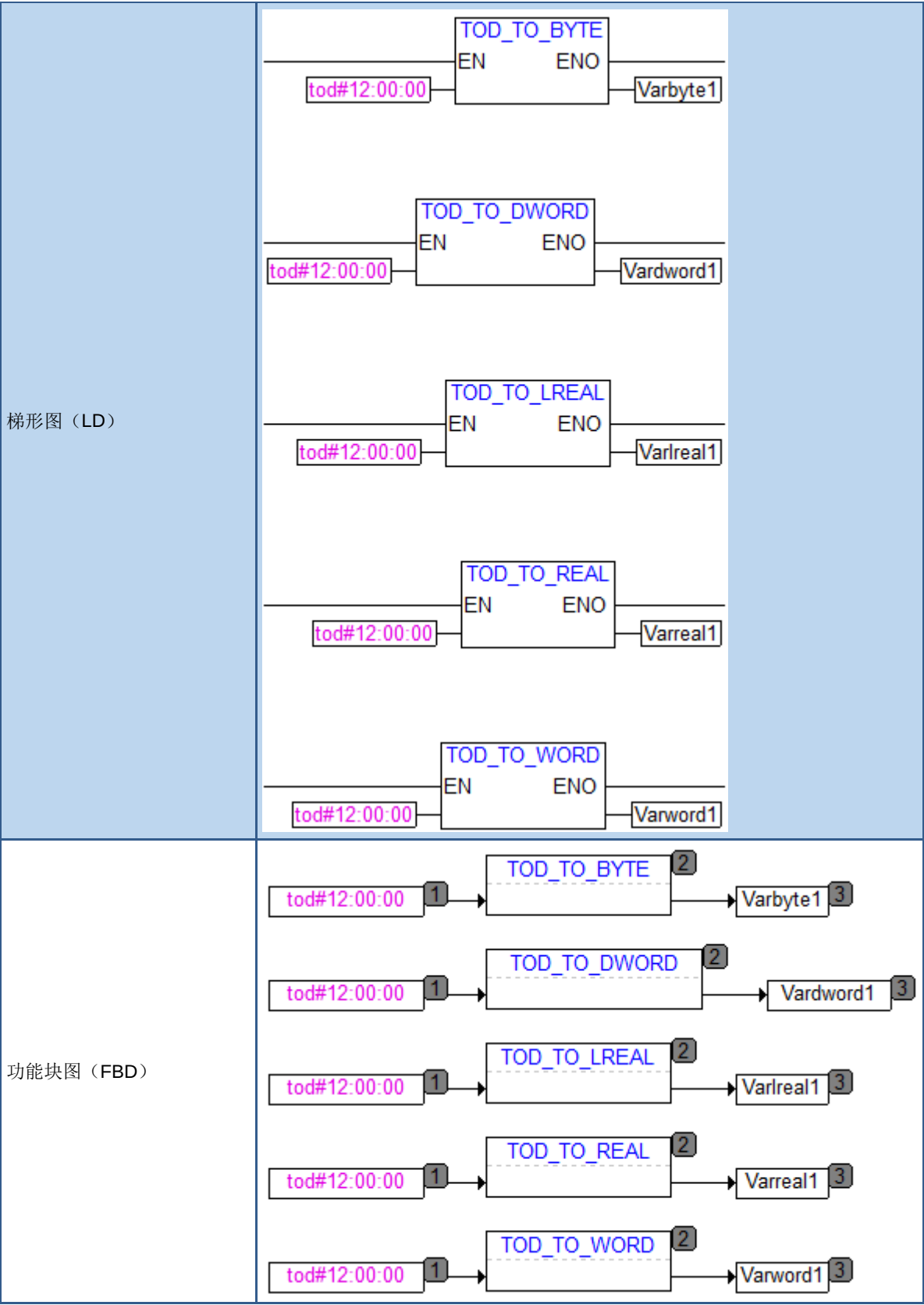
输入	说明	数据类型
输入	输入变量	TOD
输出		
输出	输出变量	BYTE、DINT、DWORD、INT、LREAL、REAL、SINT、WORD

3. 详细说明

TOD 类型转换为其他类型 (TOD_TO_xxx)，xxx 表示其他数据类型，完成 TOD 到其它类型的转换。

4. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	Varbyte1		BYTE ▼	0	G区 ▼
0002	Vardword1		DWORD ▼	0	G区 ▼
0003	Varlreal1		LREAL ▼	0	G区 ▼
0004	Varreal1		REAL ▼	0	G区 ▼
0005	Varword1		WORD ▼	0	G区 ▼
编程语言		程序			



2.2.7 沿触发器

2.2.7.1 R_TRIG（上升沿检测触发器）

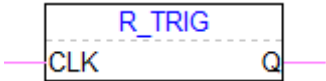


图 79 上升沿检测

1. 参数说明

输入	数据类型	描述	初始值
CLK	BOOL	输入	0
输出			
Q	BOOL	输出	0

2. 详细说明

上升沿检测算法（R_TRIG）完成检测输入 CLK 上升沿的功能，输入 CLK、输出 Q 均为 BOOL 类型，功能详细描述如下：

当输入变量 CLK 存在从 FALSE 到 TRUE 的上升沿时，输出 Q 将输出一个周期的 TRUE，其他时刻，Q 为 FALSE。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	R_TRIG_1		R_TRIG		G区
0002	Var1		BOOL	FALSE	G区
0003	Var2		BOOL	FALSE	G区

编程语言

程序

梯形图（LD）

功能块图（FBD）

2.2.7.2 F_TRIG（下降沿检测触发器）

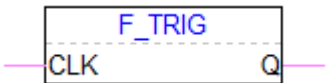


图 80 下降沿检测

1. 参数说明

输入	数据类型	描述	初始值
CLK	BOOL	输入	0
输出			
Q	BOOL	输出	0

2. 详细说明

下降沿检测算法（F_TRIG）完成检测输入 CLK 下降沿的功能，输入 CLK、输出 Q 均为 BOOL 类型，功能详细描述如下：

当输入变量 CLK 存在从“TRUE”到“FALSE”的下降沿时，输出 Q 将输出一个周期的 TRUE，其他时刻，Q 为 FALSE。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	F_TRIG_1		F_TRIG		G区
0002	Var1		BOOL	FALSE	G区
0003	Var2		BOOL	FALSE	G区

编程语言	程序
梯形图（LD）	
功能块图（FBD）	

2.2.8 双稳态触发器

2.2.8.1 SR（置位优先双稳态器）

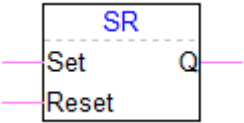


图 81 置位优先双稳态功能算法

1. 参数说明

输入	数据类型	描述	初始值
Set	BOOL	置位输入端	0
Reset	BOOL	复位输入端	0

输出			
Q	BOOL	输出端	0

真值表：

Reset	Set	Q(n+1)
0	0	Q(n)
0	1	1
1	0	0
1	1	1

2. 详细说明

SR 功能块的输入端 Set、Reset, 输出端 Q 均为 BOOL 型变量, 实现如下表达式功能: $Q = (\text{NOT Reset AND } Q) \text{ OR Set}$, 即实现置位优先的 RS 触发器功能。

3. 指令使用举例

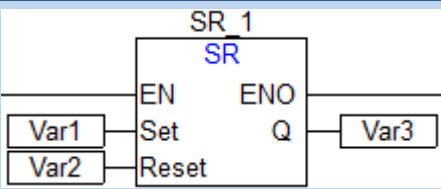
变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	SR_1		SR		G区
0002	Var1		BOOL	FALSE	G区
0003	Var2		BOOL	FALSE	G区
0004	Var3		BOOL	FALSE	G区

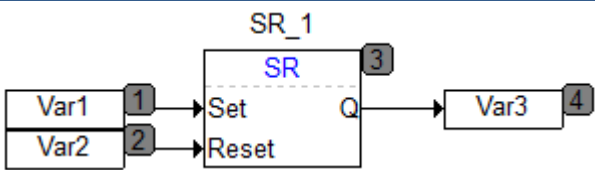
编程语言

程序

梯形图 (LD)



功能块图 (FBD)



2.2.8.2 RS（复位优先双稳态器）

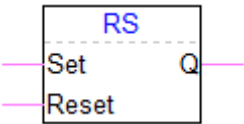


图 82 复位优先双稳态功能算法

1. 参数说明

输入	数据类型	描述	初始值
Set	BOOL	置位输入端	0
Reset	BOOL	复位输入端	0
输出			
Q	BOOL	输出端	0

真值表：

Reset	Set	Q(n+1)
0	0	Q(n)
0	1	1
1	0	0
1	1	0

2. 详细说明

RS 功能块的输入端 Set、Reset, 输出端 Q 均为 BOOL 型变量, 实现如下表达式功能: $Q = \text{NOT Reset AND } (Q \text{ OR Set})$, 即实现复位优先的 RS 触发器功能。

3. 指令使用举例

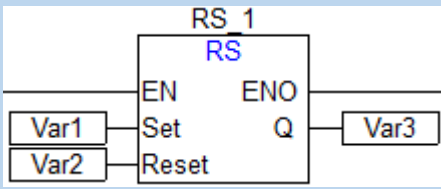
变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	RS_1		RS		G区
0002	Var1		BOOL	FALSE	G区
0003	Var2		BOOL	FALSE	G区
0004	Var3		BOOL	FALSE	G区

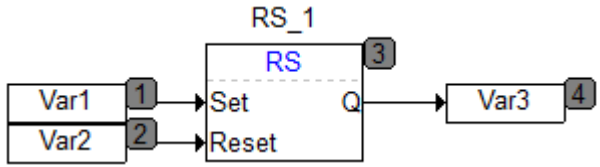
编程语言

程序

梯形图 (LD)



功能块图 (FBD)



2.2.9 计数器

2.2.9.1 CTD（递减计数器）

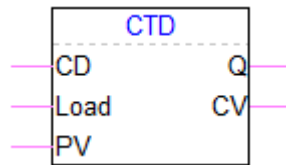


图 83 递减计数算法

1. 参数说明

输入	数据类型	描述	初始值
CD	BOOL	上升沿检测端	0
Load	BOOL	Load 为 TRUE 时计数变量 CV 初始化为 PV	0
PV	WORD	设定值	0
输出			
Q	BOOL	当 CV 小于等于 0 时, Q 输出 TRUE	1
CV	WORD	计数输出	0

2. 详细说明

减计数算法（CTD）实现当输入 CD 存在上升沿时计数值递减功能，输入端 CD、Load、输出端 Q 均为 BOOL 型变量，输入端 PV、输出端 CV 均为 WORD 型变量，详细描述如下：

■ 初始化功能：

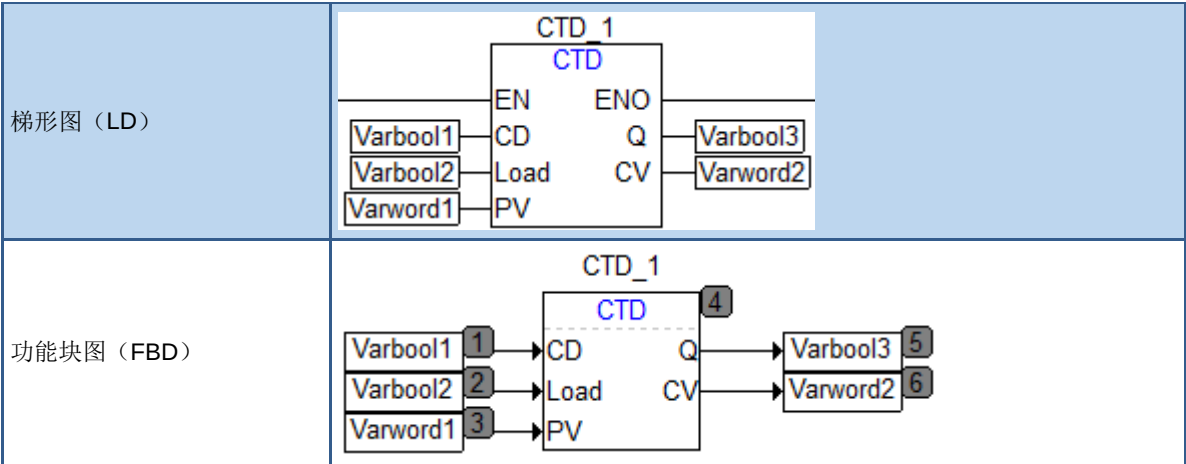
当 Load 为 TRUE 时，计数变量 CV 将被初始化为上限 PV。

■ 递减计数功能：

如果 CD 有从 FALSE 到 TRUE 的上升沿时，计数输出 CV 递减 1。当 CV 小于等于 0 的时候，Q 输出 TRUE。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	CTD_1		CTD		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Varbool2		BOOL	FALSE	G区
0004	Varword1		WORD	0	G区
0005	Varbool3		BOOL	FALSE	G区
0006	Varword2		WORD	0	G区
编程语言		程序			



2.2.9.2 CTU（递增计数器）

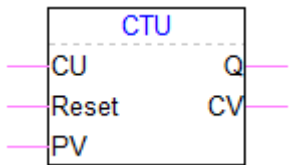


图 84 递增计数功能算法

1. 参数说明

输入	数据类型	描述	初始值
CU	BOOL	上升沿检测端	0
Reset	BOOL	Reset 为 TRUE 时计数变量 CV 初始化为 0	0
PV	WORD	设定值	0
输出			
Q	BOOL	当 CV 值大于或者等于 PV 时，Q 输出 TRUE	0
CV	WORD	计数输出	0

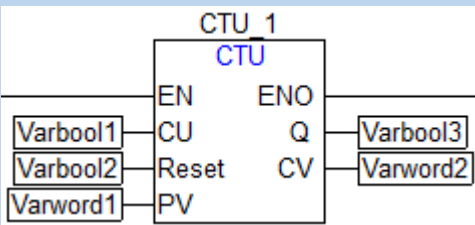
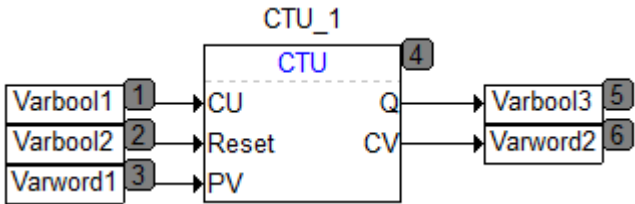
2. 详细说明

加计数算法（CTU）实现当输入 CU 存在上升沿时计数递增功能，输入变量 CU 和 Reset、输出变量 Q 都为 BOOL 类型，输入变量 PV 和输出变量 CV 都为 WORD 类型，详细描述如下：

- 初始化功能：
当 Reset 为 TRUE 时，计数输出 CV 初始化为 0。
- 递增计数功能：
如果 CU 有从 FALSE 到 TRUE 的上升沿时，计数输出 CV 递增 1。当 CV 大于等于设定值 PV 时，Q 输出 TRUE。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	CTU_1		CTU		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Varbool2		BOOL	FALSE	G区
0004	Varword1		WORD	0	G区
0005	Varbool3		BOOL	FALSE	G区
0006	Varword2		WORD	0	G区
编程语言		程序			
梯形图 (LD)					
功能块图 (FBD)					

2.2.9.3 CTUD（递增/递减计数器）

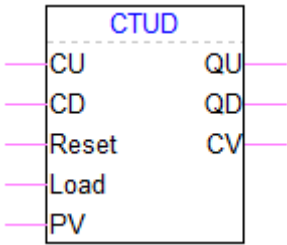


图 85 递增/递减计数功能算法

1. 参数说明

输入	数据类型	描述	初始值
CU	BOOL	上升沿检测端	0
CD	BOOL	上升沿检测端	0
Reset	BOOL	Reset 为 TRUE 时计数变量 CV 初始化为 0	0
Load	BOOL	Load 为 TRUE 时计数变量 CV 初始化为 PV；但当 Reset 也为 TRUE 时，CV=0	0
PV	WORD	设定值	0
输出			

QU	BOOL	当 CV 值大于等于 PV 时，QU 输出 TRUE	1
QD	BOOL	当 CV 值小于等于 0 时，QD 输出 TRUE	1
CV	WORD	计数输出	0

2. 详细说明

递加/递减算法，输入变量 CU、CD、Reset、Load，输出变量 QU、QD 都为 BOOL 类型，输入变量 PV，输出变量 CV 都为 WORD 类型。详细功能描述如下：

■ 初始化功能：

当 Reset 为 1 时，计数输出 CV 初始化为 0；当 Load 为 1 时，计数输出 CV 初始化为 PV。

■ 递加/递减计数功能：

如果 CU 有从 FALSE 到 TRUE 的上升沿时，CV 加 1。如果 CD 有从 FALSE 到 TRUE 的上升沿时，CV 减 1（假定不会使 CV 的值小于 0）。

当 CV 大于等于 PV 时，QU 输出 TRUE。

当 CV 小于等于 0 时，QD 输出 TRUE。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	CTUD_1		CTUD		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Varbool2		BOOL	FALSE	G区
0004	Varbool3		BOOL	FALSE	G区
0005	Varbool4		BOOL	FALSE	G区
0006	Varword1		WORD	0	G区
0007	Varbool5		BOOL	FALSE	G区
0008	Varbool6		BOOL	FALSE	G区
0009	Varword2		WORD	0	G区

编程语言

程序

梯形图（LD）

CTUD_1

CTUD

EN

CU

CD

Reset

Load

PV

ENO

QU

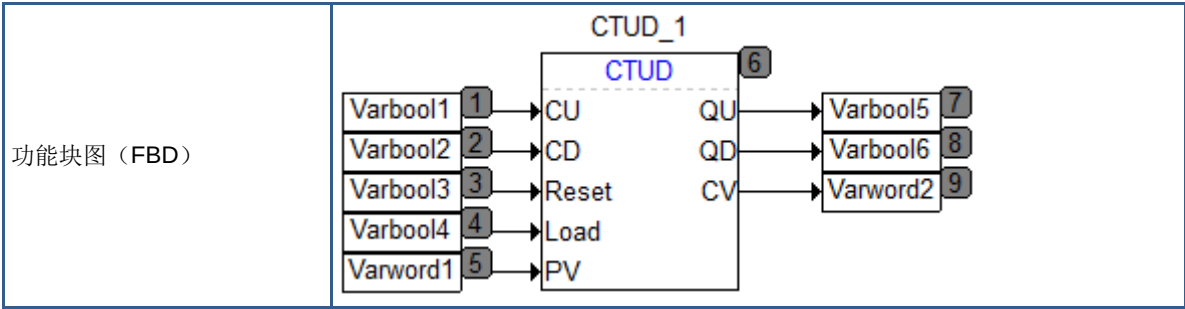
QD

CV

Varbool5

Varbool6

Varword2



2.2.10 定时器

2.2.10.1 TON（开延时定时器）

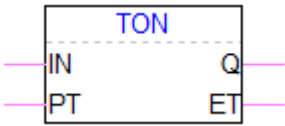


图 86 开延迟算法

1. 参数说明

输入	数据类型	描述	初始值
IN	BOOL	输入	0
PT	TIME	延时时间	T#0MS
输出			
Q	BOOL	输出	0
ET	TIME	计时时间	T#0MS

- 在 LD 程序中使用该功能块时，EN 使能引脚使用注意事项：
当 EN 使能引脚从 TRUE 变为 FALSE 后，其内部定时器继续计时，当 EN 再次变为 TRUE 时，EN 在 FALSE 期间的时间累计值生效，因此输出可能出现跃变。

2. 详细说明

延迟开功能算法（TON）完成延时输出开指令，输入 IN、输出 Q 是 BOOL 类型，输入 PT、输出 ET 是 TIME 类型，功能详细描述如下：

按照延时时间 PT 延时输出开指令。当输入 IN 为 FALSE 时，输出 Q 为 FALSE。当输入 IN 有上升沿时，ET 开始计时，如果在 PT 时间内，IN 一直没有变位，则当 ET>=PT 时，Q 为 TRUE 且 ET=PT；如果在 PT 时间内，输入 IN 再次为 FALSE，则 Q 为 FALSE 且 ET 清 0。

- 时序图：

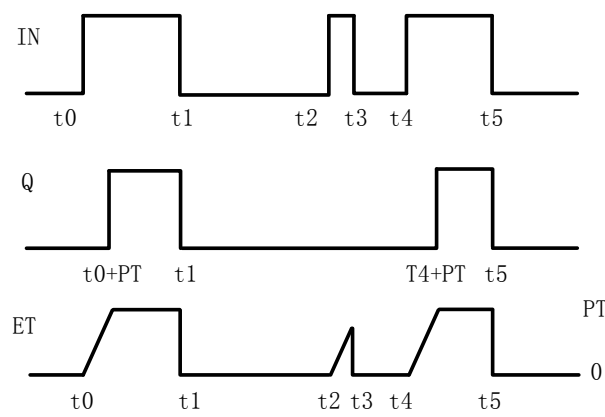


图 87 TON 的时序图

3. 指令使用举例

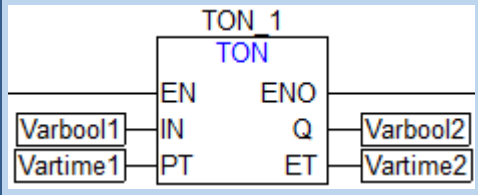
变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	TON_1		TON		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Vartime1		TIME	T#0MS	G区
0004	Varbool2		BOOL	FALSE	G区
0005	Vartime2		TIME	T#0MS	G区

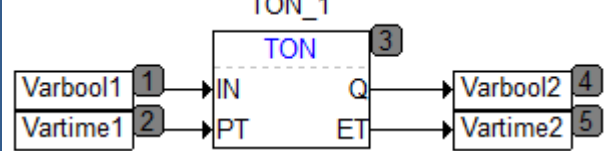
编程语言

程序

梯形图 (LD)



功能块图 (FBD)



2.2.10.2 TONR（保持型开延时定时器）

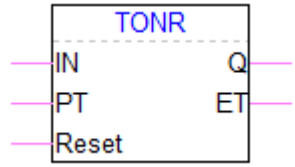


图 88 开延迟算法

1. 参数说明

输入	数据类型	描述	初始值
IN	BOOL	输入	0
PT	TIME	延时时间	T#0MS
Reset	BOOL	复位端	FALSE
输出			
Q	BOOL	输出	0
ET	TIME	计时时间	T#0MS

■ 在 LD 程序中使用该功能块时，EN 使能引脚使用注意事项：

当 EN 使能引脚从 TRUE 变为 FALSE 后，其内部定时器继续计时，当 EN 再次变为 TRUE 时，EN 在 FALSE 期间的时间累计值生效，因此输出可能出现跃变。

2. 详细说明

当 IN 变成 TRUE 时，ET 以毫秒计时直到 ET 等于 PT。如有外部触发导致在延时时间未达设定时间值前 IN 变成 FALSE，此时 ET 保持计时停止前的时间值，等到 IN 再次触发变成 TRUE 时，ET 继续计时，直到 ET 等于 PT，输出 Q 置位。RESET 为 1，则复位所有的输出值和中间变量，结束定时，为 FALSE 时，允许执行延时定时功能。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	TONR_1		TONR		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Vartime1		TIME	T#0MS	G区
0004	Varbool2		BOOL	FALSE	G区
0005	Varbool3		BOOL	FALSE	G区
0006	Vartime2		TIME	T#0MS	G区

编程语言	程序
梯形图 (LD)	
功能块图 (FBD)	

2.2.10.3 TOF（关延时定时器）

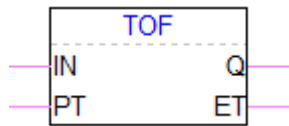


图 89 关延时算法

1. 参数说明

输入	数据类型	描述	初始值
IN	BOOL	输入	0
PT	TIME	延时时间	T#0MS
输出			
Q	BOOL	输出	0
ET	TIME	计时时间	T#0MS

- 在 LD 程序中使用该功能块时，EN 使能引脚使用注意事项：
当 EN 使能引脚从 TRUE 变为 FALSE 后，其内部定时器继续计时，当 EN 再次变为 TRUE 时，EN 在 FALSE 期间的的时间累计值生效，因此输出可能出现跃变。

2. 详细说明

TOF 实现延时输出关指令，输入 IN、输出 Q 是 BOOL 类型，输入 PT、输出 ET 是 TIME 类型，功能详细描述如下：

按照延迟时间 PT 延时输出关指令。当输入 IN 为 TRUE 时，输出 Q 为 TRUE。当输入 IN 有下降沿时，ET 开始计时，如果在 PT 时间内，IN 一直没有变位，则当 ET>=PT 时，Q 为 FALSE 且 ET=PT；如果在 PT 时间内，输入 IN 再次为 TRUE，则 Q 为 TRUE，且 ET 清 0。

- 时序图：

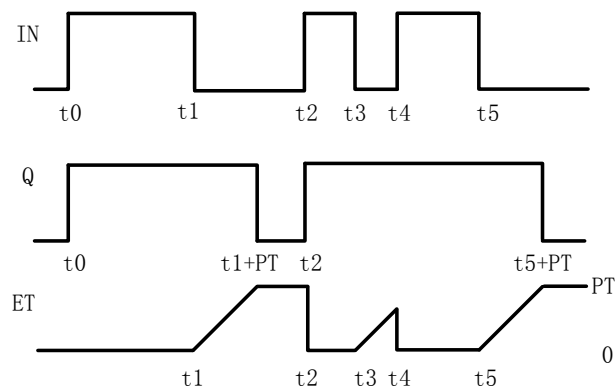
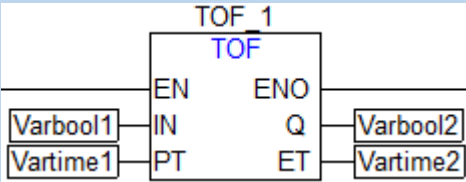
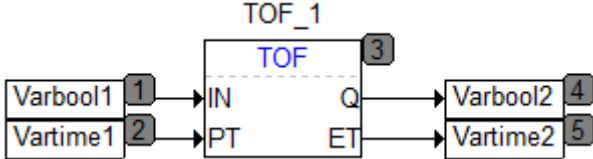


图 90 TOF 的时序图

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	TOF_1		TOF		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Vartime1		TIME	T#0MS	G区
0004	Varbool2		BOOL	FALSE	G区
0005	Vartime2		TIME	T#0MS	G区

编程语言	程序
梯形图 (LD)	
功能块图 (FBD)	

2.2.10.4 TP（普通定时器）

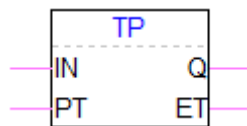


图 91 计时器算法

1. 参数说明

输入	数据类型	描述	初始值
IN	BOOL	输入	0
PT	TIME	延时时间	T#0MS
输出			
Q	BOOL	输出	0
ET	TIME	计时时间	T#0MS

- 在 LD 程序中使用该功能块时，EN 使能引脚使用注意事项：

当 EN 使能引脚从 TRUE 变为 FALSE 后，其内部定时器继续计时，当 EN 再次变为 TRUE 时，EN 在 FALSE 期间的的时间累计值生效，因此输出可能出现跃变。

2. 详细说明

计时器算法（TP）完成计时功能，输入端 IN、输出端 Q 为 BOOL 类型，输入端 PT、输出端 ET 为 TIME 类型，功能详细描述如下：

当 IN 为 FALSE 时，Q 为 FALSE，ET 为 0。当 IN 变为 TRUE 时，Q 为 TRUE 并且 ET 开始计时。在 PT 时间内，IN 无效，则 $ET \geq PT$ 时，Q 为 FALSE；当 IN 为 FALSE 时，计时完毕后，ET 为 0。

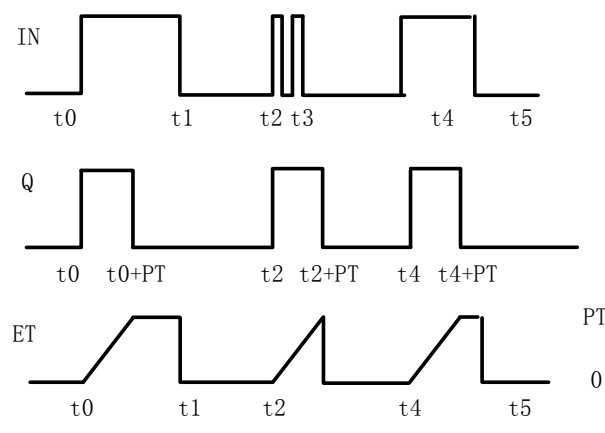


图 92 TP 的时序图

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	TP_1		TP		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Vartime1		TIME	T#0MS	G区
0004	Varbool2		BOOL	FALSE	G区
0005	Vartime2		TIME	T#0MS	G区

编程语言

程序

梯形图 (LD)

功能块图 (FBD)

2.2.10.5 RTC（设定时间计时）

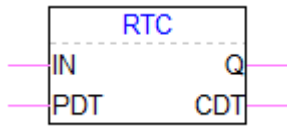


图 93 日期/时间返回算法

1. 参数说明

输入	数据类型	描述	初始值
----	------	----	-----

IN	BOOL	使能端子	0
PDT	DT	设定时间	DT#1970-01-01-00:00:00
输出			
Q	BOOL	有效标识	0
CDT	DT	当 IN 为 FALSE 时输出为 1970-01-01-00-00: 00: 00; 当 IN 为 TRUE 时输出为 PDT 按秒递增时间	DT#1970-01-01-00:00:00

■ 在 LD 程序中使用该功能块时，EN 使能引脚使用注意事项：

当 EN 使能引脚从 TRUE 变为 FALSE 后，其内部定时器继续计时，当 EN 再次变为 TRUE 时，EN 在 FALSE 期间的时间累计值生效，因此输出可能出现跃变。

2. 详细说明

实时日期时间（RTC）功能块在一个给定的时间下，输出当前日期时间。当使能端 IN 为 TRUE 时，CDT 在 PDT 的基础上以秒为单位递增，当 IN 为 FALSE 时，CDT 输出初始值 1970-01-01-00:00:00。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	RTC_1		RTC		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Vardt1		DT	DT#1970-01-0...	G区
0004	Varbool2		BOOL	FALSE	G区
0005	Vardt2		DT	DT#1970-01-0...	G区

编程语言	程序
梯形图（LD）	
功能块图（FBD）	

2.3 基础应用库

2.3.1 BCD 转换指令

BCD 码的一个字节包含 0 到 99 之间的整数。每个十进制位对应 4 位，十位数存储在 4-7 位，个位数存储在 0-3 位。BCD 码格式和 16 进制表达方式很相似，差别在于 BCD 字节值是 0-99，而 16 进制是 0-FF。

BCD 码用 4 位二进制数表示一个十进制数位，整个十进制数用一串 BCD 码来表示。例如，十进制数 59 表示成 BCD 码为 0101 1001，但表示成二进制为 2#111011。十进制 51 转换成 BCD 码，5 的二进制是 0101，1 的二进制是 0001，那么 51 转换 BCD 码为 0101 0001。

2.3.1.1 BCD_TO_INT（BCD 码转整型）

1. 功能

BCD 码转换为整型值算法（BCD_TO_INT）要求输入值是 BYTE 类型，而输出值则是 INT 类型。完成 BCD 码到整型值的转换。

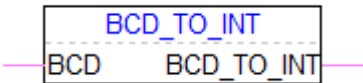


图 94 BCD_TO_INT 功能块

2. 参数说明

输入参数	数据类型	参数值说明	默认值
BCD	BYTE	需要转换的 BCD 码	0

输出参数	数据类型	参数值说明	默认值
BCD_TO_INT	INT	转换后的 INT 值 如果输入的字节的不是 BCD 码，输出是-1	0

3. 指令使用举例

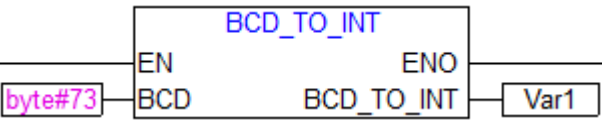
变量定义

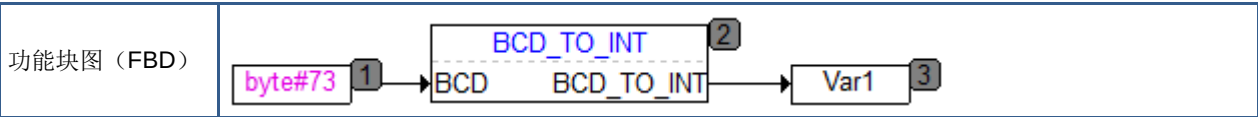
序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		INT	0	G区

编程语言

程序

梯形图（LD）





2.3.1.2 INT_TO_BCD（整型转 BCD 码）

1. 功能

整数值转换为 BCD 码算法（INT_TO_BCD）要求输入值是 INT 类型，而输出值则是 BYTE 类型。完成整型值到 BCD 码的转换。当整数值不能转换成 BCD 码字节时，输出数值 255。

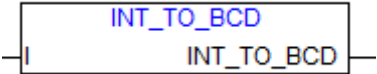


图 95 INT_TO_BCD 功能块

2. 参数说明

输入参数	数据类型	参数值说明	默认值
I	INT	需要转换的整型数据值 如果整数值为 49，则此处输入整型数据 49	0

输出参数	数据类型	参数值说明	默认值
INT_TO_BCD	BYTE	转换后的 BCD 码	0

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		BYTE	0	G区

编程语言 程序

梯形图（LD）

```
graph LR; I1[int#73] -- I --> INT_TO_BCD[INT_TO_BCD]; INT_TO_BCD -- INT_TO_BCD --> Var1[Var1];
```

功能块图（FBD）

```
graph LR; I1[int#49] -- 1 --> INT_TO_BCD[INT_TO_BCD]; INT_TO_BCD -- 2 --> INT_TO_BCD; INT_TO_BCD -- 3 --> Var1[Var1];
```

2.3.2 数据拆分合成指令

2.3.2.1 UNPACK（位拆分指令）

1. 功能

将输入 B 的第 7~0 位依次取出赋值给 B7~B0。与 PACK 指令相反。

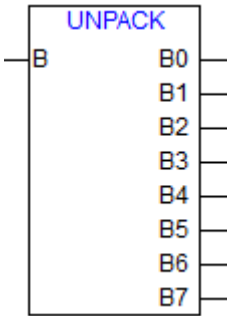


图 96 UNPACK 功能块

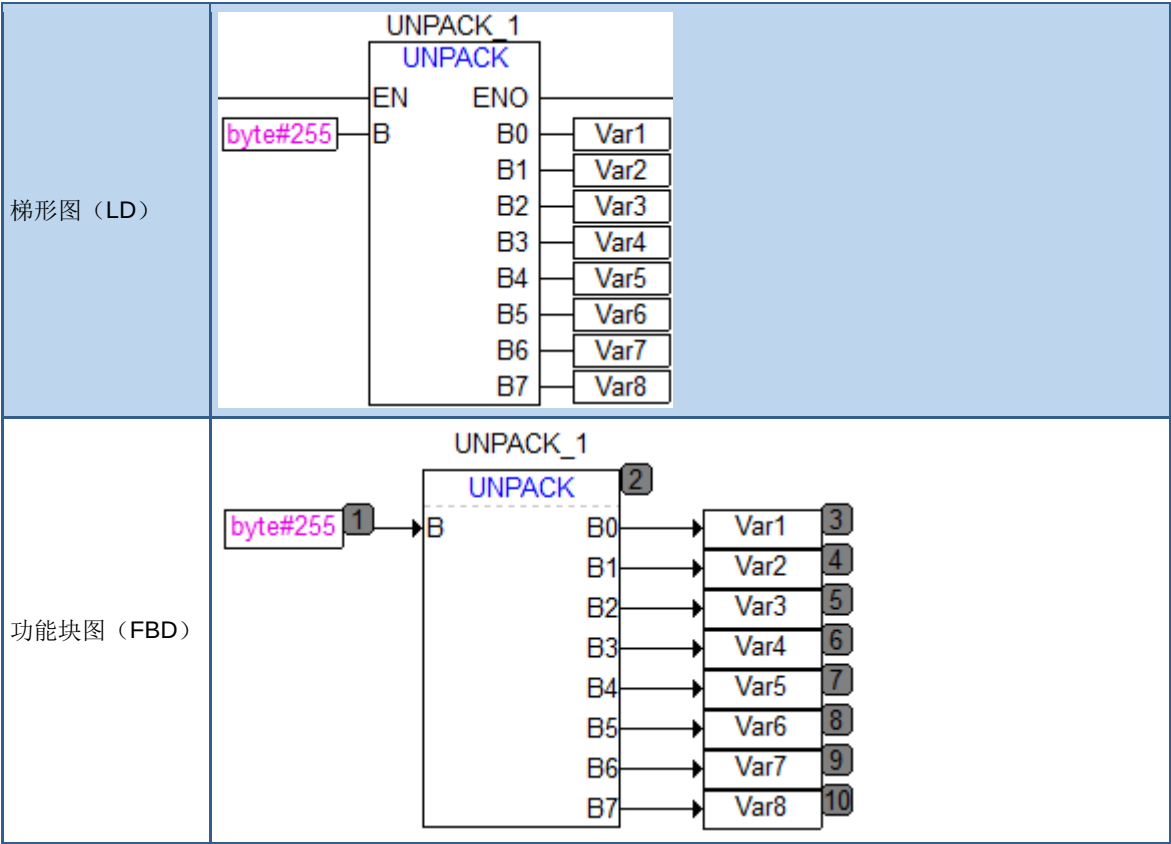
2. 参数说明

输入参数	说明	变量类型	初始值（默认）
B	BYTE 型输入变量	BYTE	0

输出参数	说明	变量类型	初始值（默认）
B0	BOOL 型输出变量	BOOL	FALSE
B1		BOOL	FALSE
B2		BOOL	FALSE
B3		BOOL	FALSE
B4		BOOL	FALSE
B5		BOOL	FALSE
B6		BOOL	FALSE
B7		BOOL	FALSE

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	UNPACK_1		UNPACK		G区
0002	Var1		BOOL	FALSE	G区
0003	Var2		BOOL	FALSE	G区
0004	Var3		BOOL	FALSE	G区
0005	Var4		BOOL	FALSE	G区
0006	Var5		BOOL	FALSE	G区
0007	Var6		BOOL	FALSE	G区
0008	Var7		BOOL	FALSE	G区
0009	Var8		BOOL	FALSE	G区
编程语言		程序			



2.3.2.2 EXTRACT（位提取指令）

1. 功能

将输入变量 X 的指定第 N 位输出。

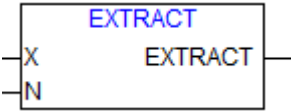


图 97 EXTRACT 功能块

2. 参数说明

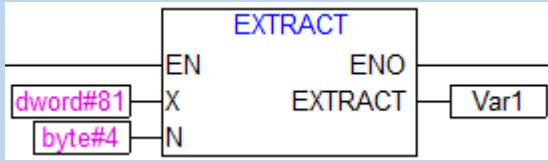
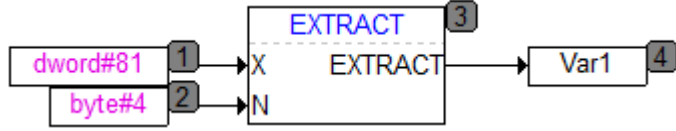
输入参数	数据类型	功能描述	初始值（默认值）
X	DWORD	被操作变量	0
N	BYTE	位置	0

输出参数	数据类型	功能描述	初始值（默认）
EXTRACT	BOOL	输出	0

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		BOOL	FALSE	G区

编程语言	程序
梯形图（LD）	
功能块图（FBD）	

2.3.2.3 PACK（位整合指令）

1. 功能

将 B0~B7 按照从第 0 位到第 7 位的顺序组合成一个 BYTE 变量。与这个指令相对应的指令是 UNPACK。

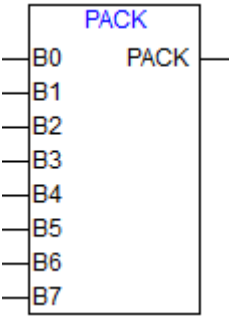


图 98 PACK 功能块

2. 参数说明

输入参数	说明	变量类型	初始值（默认）
B0	BOOL 型输入变量	BOOL	0
B1		BOOL	0
B2		BOOL	0
B3		BOOL	0
B4		BOOL	0
B5		BOOL	0
B6		BOOL	0
B7		BOOL	0

输出参数	数据类型	功能描述	初始值（默认）
Pack	BYTE	BYTE 输出	0

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		BYTE	0	G区

编程语言

程序

梯形图 (LD)

功能块图 (FBD)

2.3.2.4 PUTBIT (位赋值命令)

1. 功能

置变量指定位为指定值算法 (PUTBIT) 完成置输入值 X 指定的第 N 位为指定值 B。

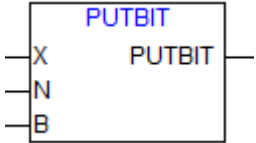


图 99 PUTBIT 功能块

2. 参数说明

输入参数	数据类型	说明	初始值 (默认)
X	DWORD	被操作变量	0
N	BYTE	位置	0
B	BOOL	指定值	FALSE

输出参数	数据类型	说明	初始值（默认）
PutBit	DWORD	输出完成指定值算法的变量值	

3. 指令使用举例

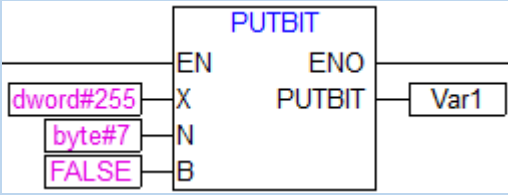
变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	Var1		DWORD	0	G区

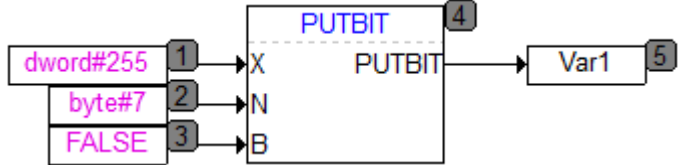
编程语言

程序

梯形图（LD）



功能块图（FBD）



2.3.2.5 DWORD_UNPACK_WORD（DWORD 拆分 WORD 指令）

1. 功能

将一个 DWORD 数按内存存储值拆分为两个 WORD 变量。

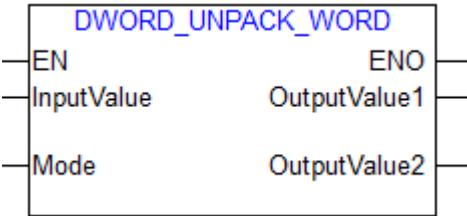


图 100 DWORD_UNPACK_WORD 功能块

2. 参数说明

输入参数	数据类型	含义
InputValue	DWORD	需要拆分的原始数据
Mode	BOOL	模式 FALSE: 高 16 位与低 16 位不交换 TRUE: 高 16 位与低 16 位交换

输出参数	类型	数据含义
OutputValue1	WORD	Mode 为 FALSE 时，对应原始数据的高 16 位字节

		Mode 为 TRUE 时，对应原始数据的低 16 位字节
OutputValue2	WORD	Mode 为 FALSE 时，对应原始数据的低 16 位字节 Mode 为 TRUE 时，对应原始数据的高 16 位字节

2.3.2.6 DINT_UNPACK_WORD（DINT 拆分 WORD 指令）

1. 功能

将一个 DINT 数按内存存储值拆分为两个 WORD 变量。

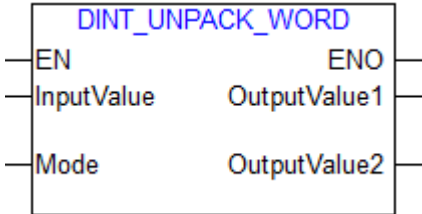


图 101 DINT_UNPACK_WORD 功能块

2. 参数说明

输入参数	数据类型	含义
InputValue	DINT	需要拆分的原始数据
Mode	BOOL	模式 FALSE：高 16 位与低 16 位不交换 TRUE：高 16 位与低 16 位交换

输出参数	类型	数据含义
OutputValue1	WORD	Mode 为 FALSE 时，对应原始数据的高 16 位字节 Mode 为 TRUE 时，对应原始数据的低 16 位字节
OutputValue2	WORD	Mode 为 FALSE 时，对应原始数据的低 16 位字节 Mode 为 TRUE 时，对应原始数据的高 16 位字节

2.3.2.7 REAL_UNPACK_WORD（REAL 拆分 WORD 指令）

1. 功能

将一个 REAL 数按内存存储值拆分为两个 WORD 变量。

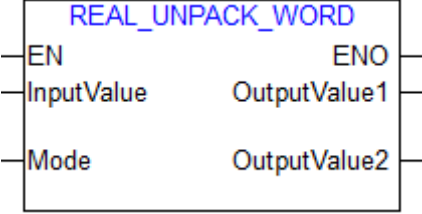


图 102 REAL_UNPACK_WORD 功能块

2. 参数说明

输入参数	数据类型	含义
------	------	----

InputValue	REAL	需要拆分的原始数据
Mode	BOOL	模式 FALSE: 高 16 位与低 16 位不交换 TRUE: 高 16 位与低 16 位交换

输出参数	类型	数据含义
OutputValue1	WORD	Mode 为 FALSE 时，对应原始数据的高 16 位字节 Mode 为 TRUE 时，对应原始数据的低 16 位字节
OutputValue2	WORD	Mode 为 FALSE 时，对应原始数据的低 16 位字节 Mode 为 TRUE 时，对应原始数据的高 16 位字节

2.3.2.8 WORD_PACK_DWORD（WORD 合并 DWORD 指令）

1. 功能

将两个 WORD 变量按内存存储值整合为一个 DWORD 变量。

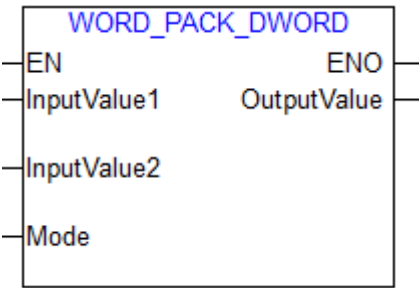


图 103 WORD_PACK_DWORD 功能块

2. 参数说明

输入参数	数据类型	含义
InputValue1	WORD	需要合成的原始数据 1
InputValue2	WORD	需要合成的原始数据 2
Mode	BOOL	模式 FALSE: InputValue1 为合成之后的高 16 位，InputValue2 为合成之后的低 16 位 TRUE: InputValue1 为合成之后的低 16 位，InputValue2 为合成之后的高 16 位

输出参数	类型	数据含义
OutputValue	DWORD	按照 Mode 合成之后的数据

2.3.2.9 WORD_PACK_DINT（WORD 合并 DINT 指令）

1. 功能

将两个 WORD 变量按内存存储值整合为一个 DINT 变量。

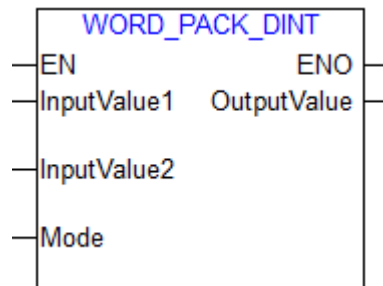


图 104 WORD_PACK_DINT 功能块

2. 参数说明

输入参数	数据类型	含义
InputValue1	WORD	需要合成的原始数据 1
InputValue2	WORD	需要合成的原始数据 2
Mode	BOOL	模式 FALSE: InputValue1 为合成之后的高 16 位, InputValue2 为合成之后的低 16 位 TRUE: InputValue1 为合成之后的低 16 位, InputValue2 为合成之后的高 16 位

输出参数	类型	数据含义
OutputValue	DINT	按照 Mode 合成之后的数据

2.3.2.10 WORD_PACK_REAL (WORD 合并 REAL 指令)

1. 功能

将两个 WORD 变量按内存存储值整合为一个 REAL 变量。

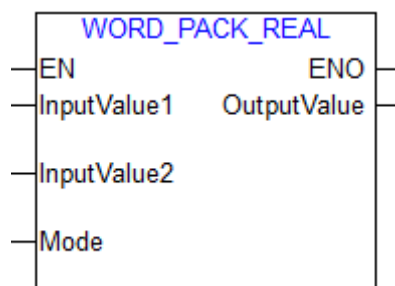


图 105 WORD_PACK_REAL 功能块

2. 参数说明

输入参数	数据类型	含义
InputValue1	WORD	需要合成的原始数据 1
InputValue2	WORD	需要合成的原始数据 2
Mode	BOOL	模式 FALSE: InputValue1 为合成之后的高 16 位, InputValue2 为合成之后的低 16 位 TRUE: InputValue1 为合成之后的低 16 位, InputValue2 为合成之后的高 16 位

输出参数	类型	数据含义
OutputValue	REAL	按照 Mode 合成之后的数据

2.3.3 数学辅助功能

2.3.3.1 DERIVATIVE（微分）

1. 功能

该指令对连续输入的变量进行微分运算。为了获得最好结果，DERIVATIVE 指令只对最新的四个输入值进行微分，减小因输入参数不精确产生的误差。

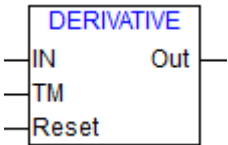


图 106 DERIVATIVE 功能块

2. 微分迭代公式

$$OUT = \frac{3 * [IN(k) - IN(k - 3)] + IN(k - 1) - IN(k - 2)}{3 * TM(k - 2) + 4 * TM(k - 1) + 3 * TM(k)} * 1000$$

k-3、k-2、k-1、k 为连续四次输入值的标记。

3. 参数说明

输入参数	数据类型	功能描述	初始值（默认）
IN	REAL	输入变量	0
TM	DWORD	时间常数	0
Reset	BOOL	重启信号	FALSE

输出参数	数据类型	功能描述	初始值（默认）
OUT	REAL	微分算法的输出	0

4. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	DERIVATIVE_1		DERIVATIVE		G区
0002	Varreal1		REAL	0	G区
0003	Varbool1		BOOL	FALSE	G区
0004	Varreal2		REAL	0	G区
编程语言		程序			

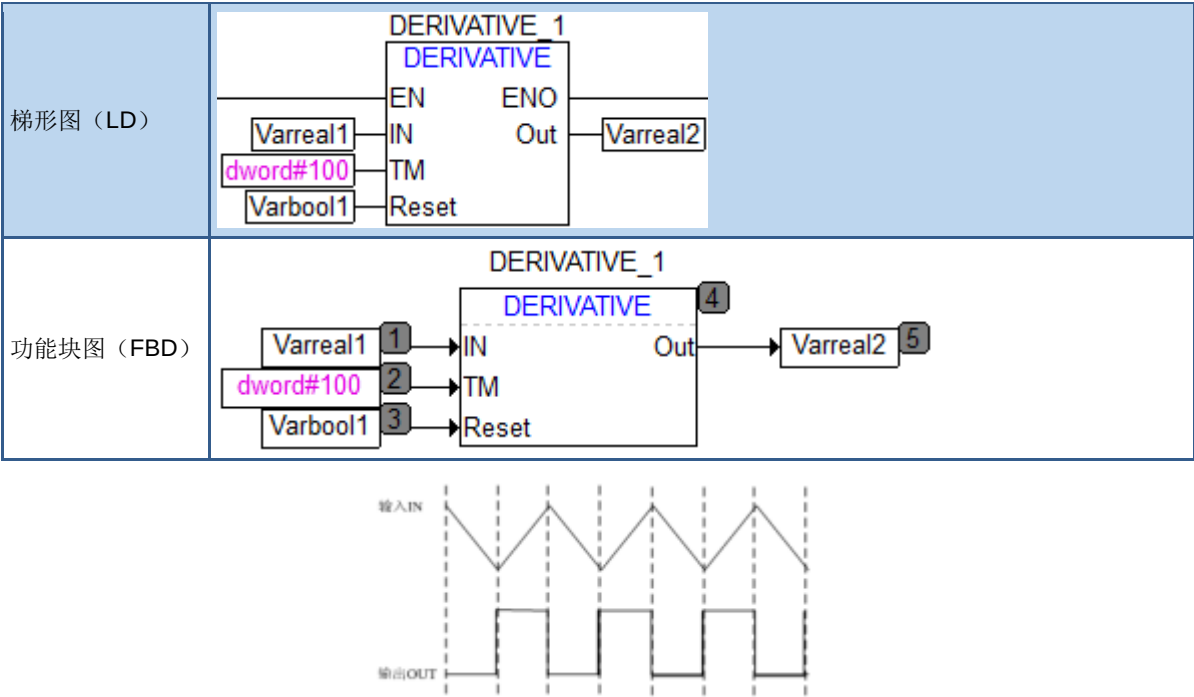


图 107 输入输出对应关系

2.3.3.2 INTEGRAL（积分）

1. 功能

积分算法（INTEGRAL）近似求解输入函数的积分。

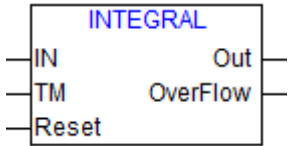


图 108 INTEGRAL 功能块

2. 积分迭代公式

$$OUT(k)=OUT(k-1)+TM*IN(k);$$

k-1、k 为连续两次输入值的标记。

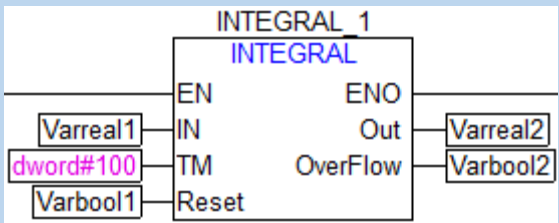
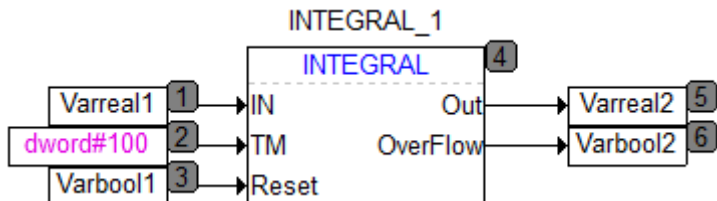
3. 参数说明

输入参数	数据类型	功能描述	初始值（默认）	掉电保护
IN	REAL	输入变量	0	FALSE
TM	DWORD	时间常数	0	TRUE
Reset	BOOL	重启信号	FALSE	TRUE

输出参数	数据类型	功能描述	初始值（默认）	掉电保护
OUT	REAL	近似积分值	0	TRUE

OverFlow	BOOL	溢出标志	FALSE	TRUE
----------	------	------	-------	------

4. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	INTEGRAL_1		INTEGRAL		G区
0002	Varreal1		REAL	0	G区
0003	Varbool1		BOOL	FALSE	G区
0004	Varreal2		REAL	0	G区
0005	Varbool2		BOOL	FALSE	G区
编程语言		程序			
梯形图（LD）					
功能块图（FBD）					

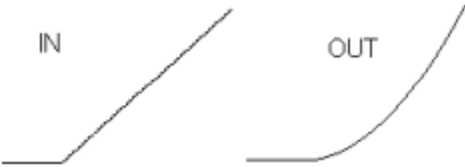


图 109 输入输出对应关系

2.3.3.3 STATISTICS_INT（整型统计）

1. 功能

计算标准统计算法（STATISTICS_INT）统计所有 INT 型输入变量中的最小值、最大值和平均值。

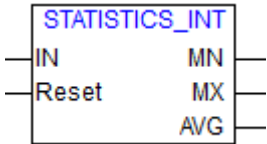


图 110 STATISTICS_INT 功能块

2. 参数说明

输入参数	数据类型	功能描述	初始值（默认）	掉电保护
------	------	------	---------	------

IN	INT	被操作变量	0	FALSE
Reset	BOOL	初始化	FALSE	TRUE

输出参数	数据类型	功能描述	初始值（默认）	掉电保护
MN	INT	最小值输出	32,767	TRUE
MX	INT	最大值输出	-32,768	TRUE
AVG	INT	平均值输出	0	FALSE

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	STATISTICS_INT_1		STATISTICS_INT		G区
0002	Varint1		INT	0	G区
0003	Varbool1		BOOL	FALSE	G区
0004	Varint2		INT	0	G区
0005	Varint3		INT	0	G区
0006	Varint4		INT	0	G区

编程语言

程序

梯形图（LD）

功能块图（FBD）

2.3.3.4 STATISTICS_REAL（实型统计）

1. 功能

统计所有 REAL 型输入变量中的最大值、最小值和平均值。

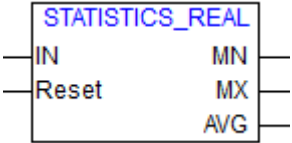


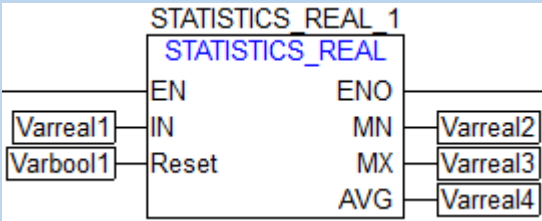
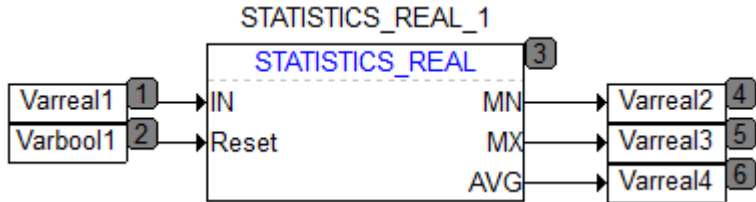
图 111 STATISTICS_REAL 功能块

2. 参数说明

输入参数	数据类型	功能描述	初始值（默认）	掉电保护
IN	REAL	被操作变量	0	FALSE
Reset	BOOL	初始化	FALSE	TRUE

输出参数	数据类型	功能描述	初始值（默认）	掉电保护
MN	REAL	最小值输出	3.4E+38	TRUE
MX	REAL	最大值输出	- 3.4E+38	TRUE
AVG	REAL	平均值输出	0	TRUE

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	STATISTICS_REAL_1		STATISTICS_REAL		G区
0002	Varreal1		REAL	0	G区
0003	Varbool1		BOOL	FALSE	G区
0004	Varreal2		REAL	0	G区
0005	Varreal3		REAL	0	G区
0006	Varreal4		REAL	0	G区
编程语言		程序			
梯形图（LD）					
功能块图（FBD）					

2.3.3.5 VARIANCE（平方偏差）

1. 功能

计算输入值方差算法（VARIANCE）计算所有输入变量的方差。标准偏差可以由平方偏差的平方根得到。

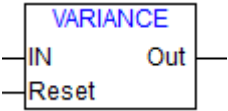


图 112 VARIANCE 功能块

2. 参数说明

输入参数	数据类型	功能描述	初始值（默认）	掉电保护
IN	REAL	被操作变量	0	FALSE
Reset	BOOL	初始化	FALSE	TRUE

输出参数	数据类型	功能描述	初始值（默认）	掉电保护
Out	REAL	方差值输出	0	FALSE

中间变量	数据类型	初始值（默认）	掉电保护
Z	DWORD	0	TRUE
A	LREAL	0	TRUE
B	LREAL	0	TRUE

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	VARIANCE_1		VARIANCE		G区
0002	Varreal1		REAL	0	G区
0003	Varbool1		BOOL	FALSE	G区
0004	Varreal2		REAL	0	G区

编程语言

程序

梯形图（LD）

A ladder logic diagram for the VARIANCE_1 block. It shows a single rungs with two inputs: 'Varreal1' connected to the 'IN' port and 'Varbool1' connected to the 'Reset' port. The output 'Out' is connected to 'Varreal2'.

功能块图（FBD）

A function block diagram for the VARIANCE_1 block. The block is labeled 'VARIANCE_1' and 'VARIANCE'. It has two inputs: 'Varreal1' (labeled 1) connected to 'IN' and 'Varbool1' (labeled 2) connected to 'Reset'. The output 'Out' (labeled 3) is connected to 'Varreal2' (labeled 4).

2.3.4 控制器指令

2.3.4.1 PID（比例积分微分控制器）

1. 功能

比例积分微分控制器，对现场输入的过程值与设定值比较，对其进行 PID 调节控制。将调节的指令输出给现场设备从而达到及时响应的调节任务，同时对输出值进行限幅操作，以保证输出在许可的范围内操作，并对超限数值及时报警。

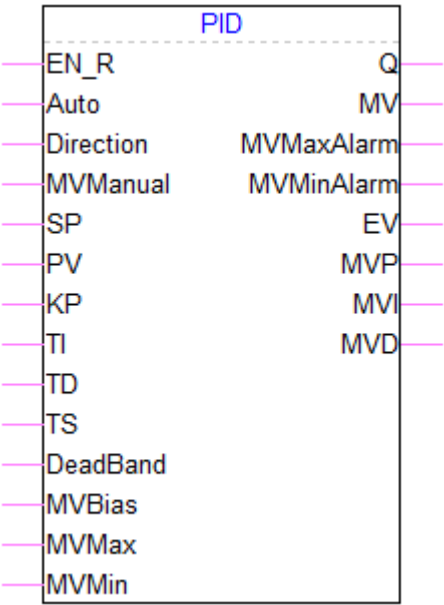


图 113 PID 功能块

2. 参数说明

输入参数	数据类型	功能描述	参数值说明	初 始 值 (默认)	掉电保护
EN_R	BOOL	使能	FALSE: 无效 TRUE: 上升沿设定、高电平有效	FALSE	TRUE
Auto	BOOL	“自动”模式 选择	FALSE: 手动 TRUE: 自动	FALSE	TRUE
Direction	BOOL	“正作用”方 式选择	FALSE: 反作用 (EV=SP-PV) TRUE: 正作用 (EV=PV-SP)	FALSE	TRUE
MVManual	REAL	手动输入值		0	TRUE
SP	REAL	设定值		0	TRUE
PV	REAL	过程值		0	FALSE
KP	REAL	比例增益		1.0	TRUE
TI	REAL	积分时间 (S)	Ti>=0, 如果积分时间为 0, 表示没有积分环 节	0	TRUE
TD	REAL	微分时间 (S)	Td>=0, 如果微分时间为 0, 表示没有微分环 节	0	TRUE

TS	REAL	运算周期 (S)	Ts>0	0.1	TRUE
DeadBand	REAL	偏差死区限	(1) DeadBand>=0 (2) 若 DeadBand=0, 表示没有死区 (3) 偏差绝对值大于该值时有效, 否则偏差为 0	0	TRUE
MVBias	REAL	前馈控制量		0.0	TRUE
MVMax	REAL	输出值上限		100.0	TRUE
MVMin	REAL	输出值下限		-100.0	TRUE

输出参数	数据类型	功能描述	参数值说明	初始值 (默认)	掉电保护
Q	BOOL	使能状态标志	FALSE: 无效 TRUE: 上升沿设定、高电平有效	FALSE	FALSE
MV	REAL	控制量输出		0	TRUE
MVMaxAlarm	BOOL	输出值超上限报警	FALSE: 未超上限 TRUE: 已超上限	FALSE	FALSE
MVMinAlarm	BOOL	输出值超下限报警	FALSE: 未超下限 TRUE: 已超下限	FALSE	FALSE
EV	REAL	设定值与过程值的偏差	若有死区, EV 则为经死区处理后的偏差值	0	FALSE
MVP	REAL	比例分量		0	FALSE
MVI	REAL	积分分量		0	FALSE
MVD	REAL	微分分量		0	FALSE

3. 功能详细说明

■ 使能

当使能端 EN_R 为真时, 程序正常运行, 否则程序返回不运算。

■ 判断设定值

- ☐ 输出上下限 MVMax、MVMin 设置判断。

输出下限一定要小于输出上限, 否则输出值超上下限标志均置位且程序返回不运算。

- ☐ 积分时间 Ti、微分时间 Td 和死区范围 DeadBand 的设置判断。

积分时间 Ti、微分时间 Td 和死区范围 DeadBand 必须为非负数, 否则按照初始值进行运算。

■ 计算

- ☐ 手动模式

输出 MV 等于手动输入值 MVManual, 并且 SP 跟踪 PV。

- ☐ 自动模式

根据正反作用方式 DerectAction、设定值 SP、测量值 PV 和死区范围 DeadBand 计算偏差; 再根据输出 MV、输出上下限值以及偏差的正负计算积分饱和系数。

根据比例增益 K_p 、积分时间 T_i 、微分时间 T_d 和积分饱和系数计算控制输出的比例分量 MV_p 、积分分量 MV_i 、微分量 MV_d 以及控制输出 MV 。

PID 功能块的逻辑结构如图 114 所示。

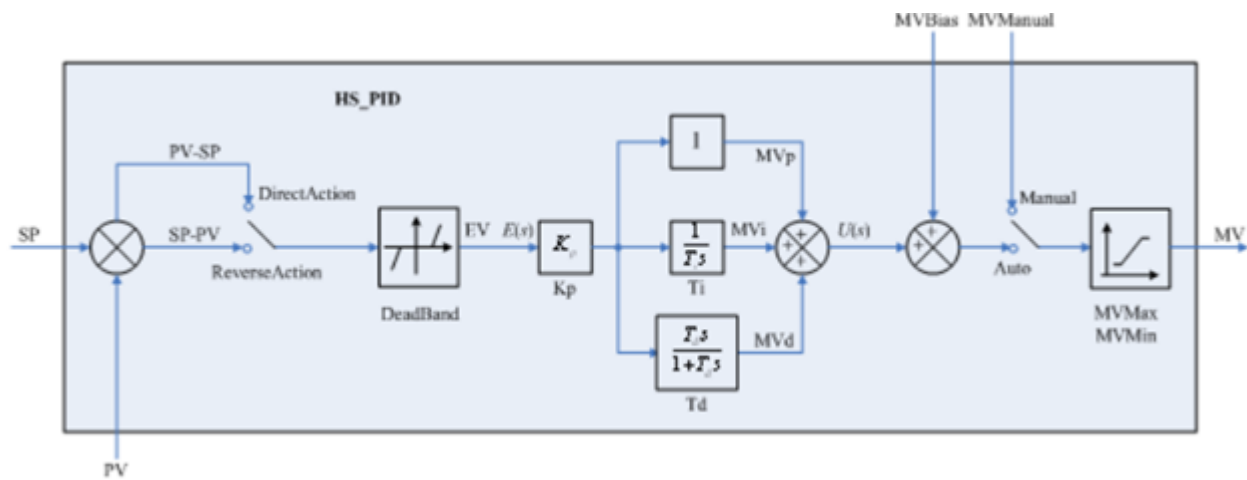


图 114 PID 功能块逻辑结构

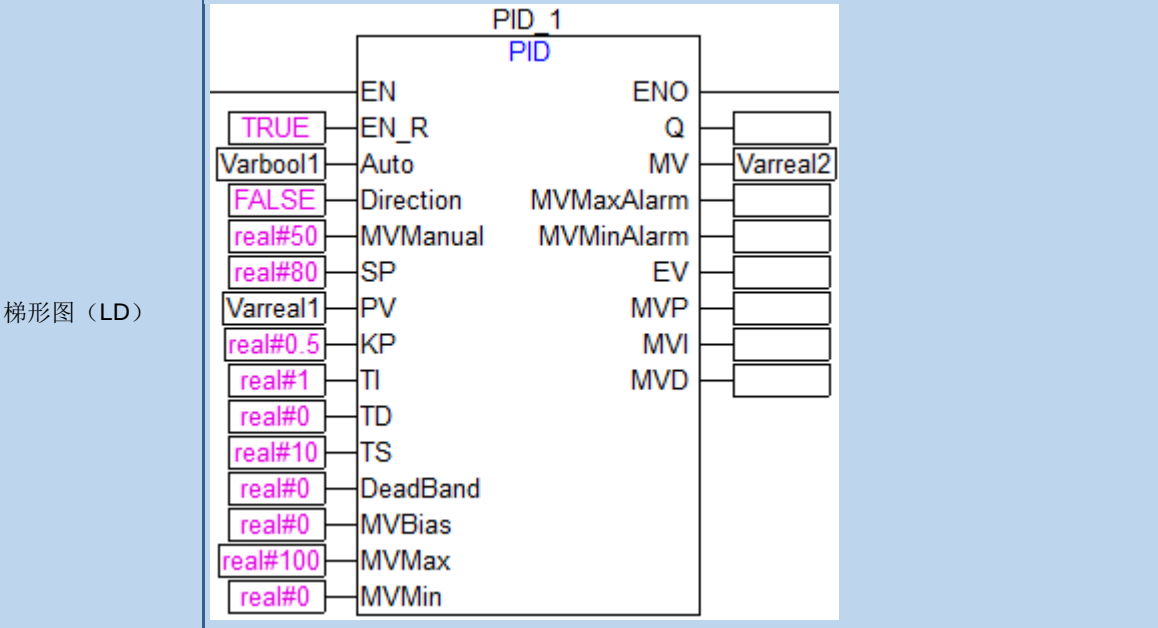
4. 指令使用举例

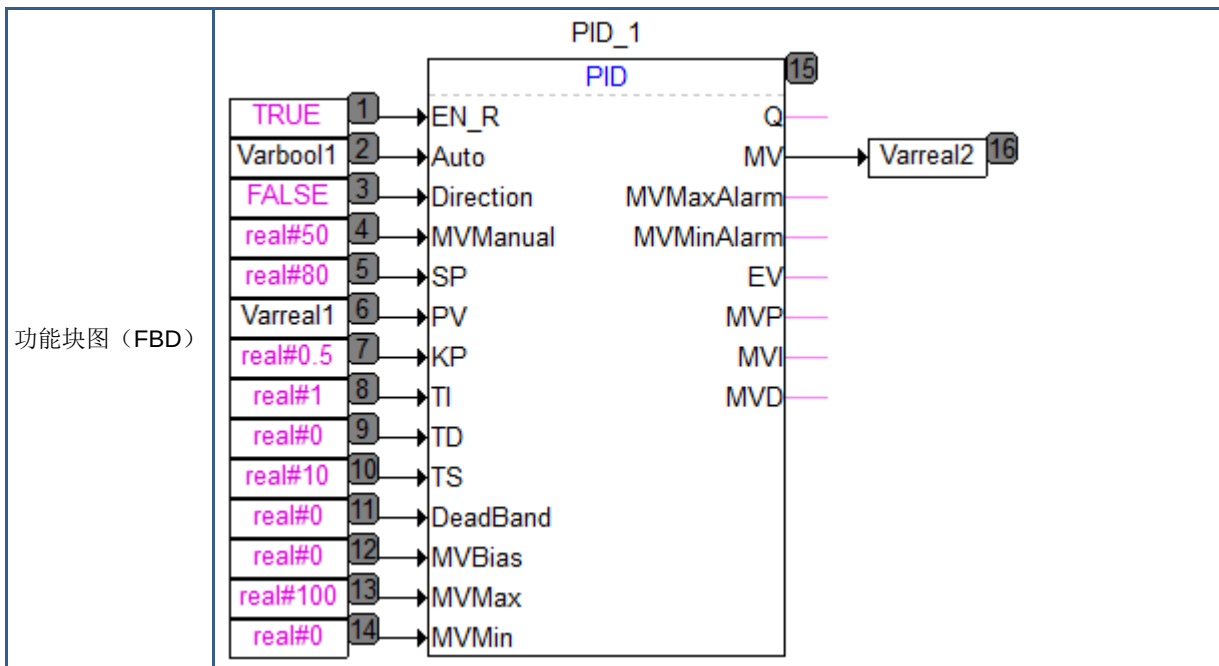
变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	PID_1		PID		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Varreal1		REAL	0	G区
0004	Varreal2		REAL	0	G区

编程语言

程序





2.3.5 信号发生器指令

2.3.5.1 BLINK（脉冲信号发生器）

1. 概述

脉冲信号发生器算法（BLINK）完成按指定时间输出高电平和低电平的矩形波。

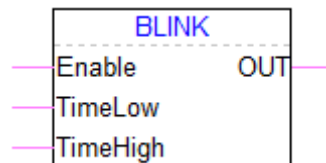


图 115 BLINK 功能块

2. 参数说明

输入参数	说明	变量类型	初始值（默认）	掉电保护
Enable	使能端	BOOL	FALSE	TRUE
TimeLow	低电平时间	TIME	T#0S	TRUE
TimeHigh	高电平时间	TIME	T#0S	TRUE

输出参数	说明	变量类型	初始值（默认）	掉电保护
OUT	输出	BOOL	FALSE	TRUE

- 在 LD 程序中使用该功能块时，EN 使能引脚使用注意事项：

当 EN 使能引脚从 TRUE 变为 FALSE 后，其内部定时器继续计时，当 EN 再次变为 TRUE 时，EN 在 FALSE 期间的时间累计值生效，因此输出可能出现跃变。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	BLINK_1		BLINK		G区
0002	Var1		BOOL	FALSE	G区

编程语言

程序

梯形图 (LD)

功能块图 (FBD)

当指令执行时，Out 所示波形输出如图 116 所示。

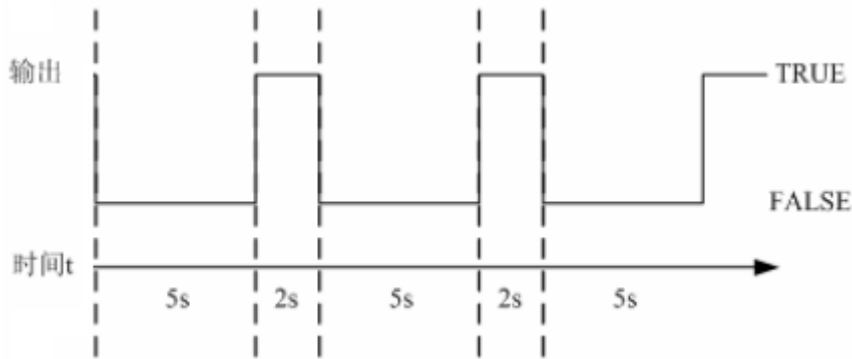


图 116 输出波形

2.3.5.2 GEN（典型周期信号发生器）

1. 概述

产生典型周期函数算法（GEN）生成指定周期函数的类型、周期和振幅的信号。

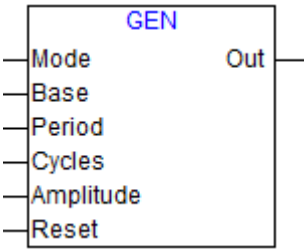


图 117 GEN 功能块

2. 参数说明

输入参数	变量类型	功能描述	参数值说明	初始值（默认）	掉电保持
Mode	BYTE	指定要产生的信号类型	直接在 MODE 处输入各波形对应的数值，则产生对应的波形	0	TRUE
		0: TRIANGLE	三角波		
		1 : TRIANGLE_POS	零起点三角波		
		2 : SAWTOOTH_RISE	上升锯齿波		
		3 : SAWTOOTH_FALL	下降锯齿波		
		4: RECTANGLE	方波		
		5: SINUS	正弦波		
		6: COSINU	余弦波		
Base	BOOL	循环方式选择	当 BASE 为 TRUE 时，信号发生器与定义的循环周期有关。当 BASE 为 FALSE 时，信号发生器周期为当前任务周期的个数	FALSE	TRUE
Period	TIME	定义循环周期（Base 为真时有效）		T#0MS	TRUE
Cycles	DWORD	定义循环周期（Base 为假时有效）		0	TRUE
Amplitude	INT	信号的振幅		0	TRUE
Reset	BOOL	初始化	当 RESET=TRUE 时，信号发生器被重新设置为 0	FALSE	TRUE
输出参数		变量类型	功能描述	初始值（默认）	掉电保持
Out		INT	波形信号输出值	0	TRUE

■ 在 LD 程序中使用该功能块时，EN 使能引脚使用注意事项：

当 EN 使能引脚从 TRUE 变为 FALSE 后, 其内部定时器继续计时, 当 EN 再次变为 TRUE 时, EN 在 FALSE 期间的时间累计值生效, 因此输出可能出现跃变。

3. 功能详细说明

■ 初始化功能

当 RESET=1 时, 输出和相关计数变量清零。

■ 计算生成波形

当 BASE=0 时, 按照周期 CYCLES 和赋值 AMPLITUDE 生成 MODE 类型的波形; 当 BASE=1 时, 只是周期不同, 周期变为 PERIOD。

4. 注意事项

- 当 BASE=0 且函数类型为 SAWTOOTH_RISE 或 SAWTOOTH_FALL 时, CYCLE 所示循环周期如图 118 黑色字体所示, 而不是红色字体所示 (因为实际的信号产生中, 不会有直上直下的波形, 即一个时间点对应两个值的情况)。

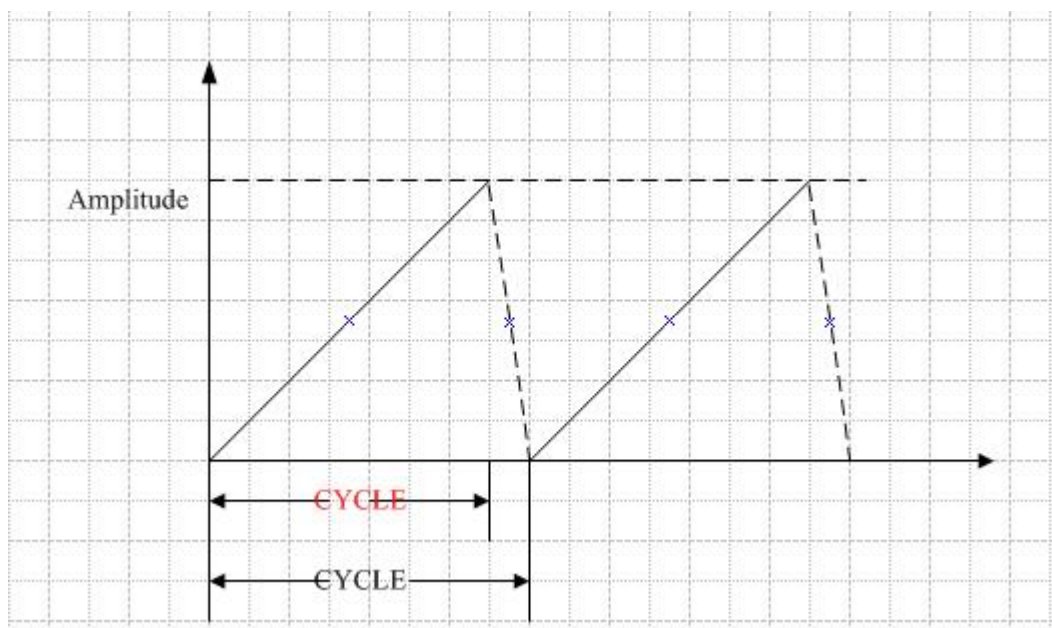


图 118 周期图

- BASE=1 且函数类型为 SINUS 或 COSINUS 时, 如果 IEC 运算周期与波形循环周期 PERIOD 之比在 0.1 之内, 可以实现全波形的采集; 如果比值大于 0.1, 有可能采集不到波峰波谷值, 必然能采集到的最大值与最小值与比值大小相关。

根据 Mode 处输入不同, 产生的波形如下图所示。

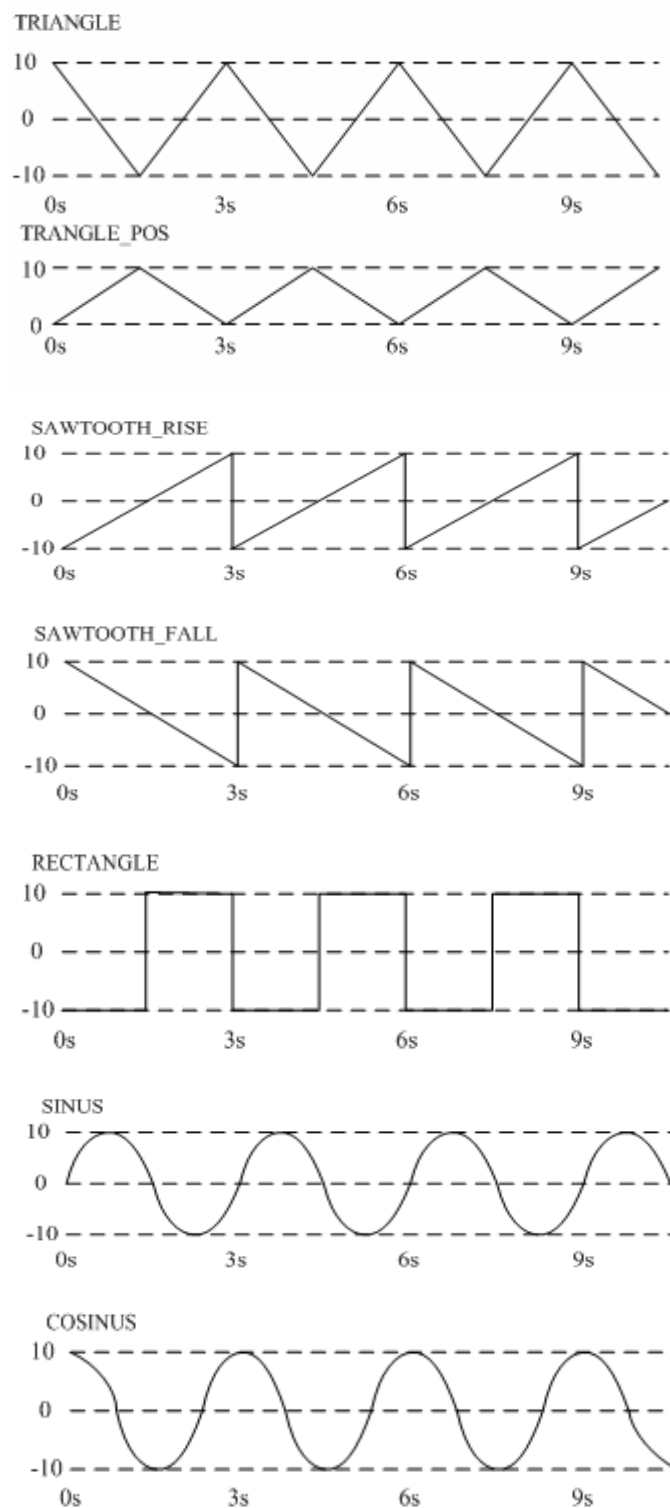
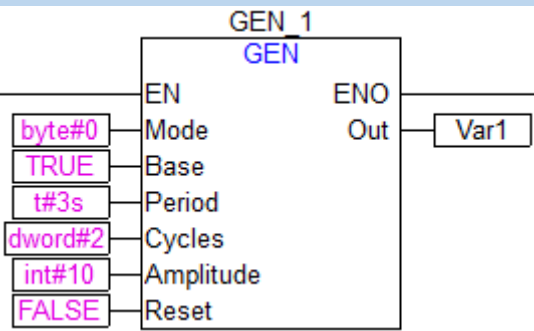
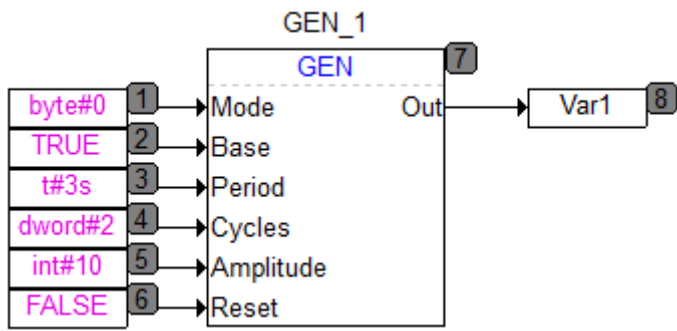


图 119 波形图

5. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	GEN_1		GEN		G区
0002	Var1		INT	0	G区
编程语言		程序			
梯形图（LD）					
功能块图（FBD）					

2.3.5.3 RAND（随机数发生器）

1. 功能

产生随机数。

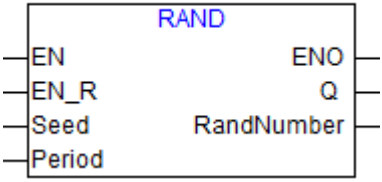


图 120 RAND 功能块

2. 参数说明

输入参数	数据类型	功能描述	参数值说明	初始值（默认）	掉电保持
EN_R	BOOL	使能	0：无效，停止产生随机数，RandNumber 保持最后一次产生的随机数不变 1：上升沿使能，且高电平有效	FALSE	FALSE
Seed	WORD	种子	种子是指计算随机序列的初始数值，取值范围为：0~65,535	0	FALSE
Period	DWORD	产生随机数	0：每个扫描周期产生一个随机数	0	FALSE

		的周期，单位为毫秒（ms）	非 0：如设置为 200，则每 200 ms 产生一个随机数。 取值范围为：0~4,294,967,295		
--	--	---------------	--	--	--

输出参数	数据类型	功能描述	参数值说明	初始值（默认）	掉电保持
Q	BOOL	是否有效数据	0：无效数据 1：有效数据	FALSE	FALSE
RandNumber	WORD	随机数	0-65,535	0	FALSE

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	RAND_1		RAND		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Varword1		WORD	0	G区

编程语言

程序

梯形图（LD）

功能块图（FBD）

程序说明：

EN_R 置位时，产生随机数，每个周期产生 1 个随机数。

EN_R 复位时，停止产生随机数，RandNumber 保持最后一次产生的随机数不变。

2.3.6 模拟量处理指令

2.3.6.1 RAMP_INT（整型限速）

1. 功能

限制整型输入函数的升降速度。

输出 OUT 是输入 IN 经上升和下降限制整定后的函数值。当 TIMEBASE 设置为 t#0s，ASCEND 和 DESCEND 与时间间隔无关；当 TIMEBASE 设置为 t#0s 之外的值时，ASCEND 和 DESCEND 是时间间隔 TIMEBASE 时间长短的限制值。

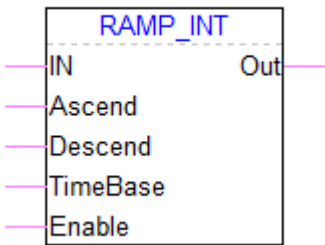


图 121 RAMP_INT 功能块

2. 参数说明

输入参数	数据类型	功能描述	初始值（默认）	掉电保护
IN	INT	被操作变量	0	FALSE
Ascend	INT	时间间隔内最大上升值	0	TRUE
Descend	INT	时间间隔内最大下降值	0	TRUE
TimeBase	TIME	时间间隔	T#1S	TRUE
Enable	BOOL	速率限幅使能开关	TRUE	FALSE

输出参数	数据类型	功能描述	初始值（默认）	掉电保护
Out	INT	经上升和下降限制后的值	0	TRUE

- 在 LD 程序中使用该功能块时，EN 使能引脚使用注意事项：
当 EN 使能引脚从 TRUE 变为 FALSE 后，其内部定时器继续计时，当 EN 再次变为 TRUE 时，EN 在 FALSE 期间的的时间累计值生效，因此输出可能出现跃变。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	RAMP_INT_1		RAMP_INT		G区
0002	Var1		INT	0	G区
0003	Var2		INT	0	G区

编程语言

程序

梯形图（LD）

RAMP INT 1

RAMP_INT

EN

ENO

Out

Var1

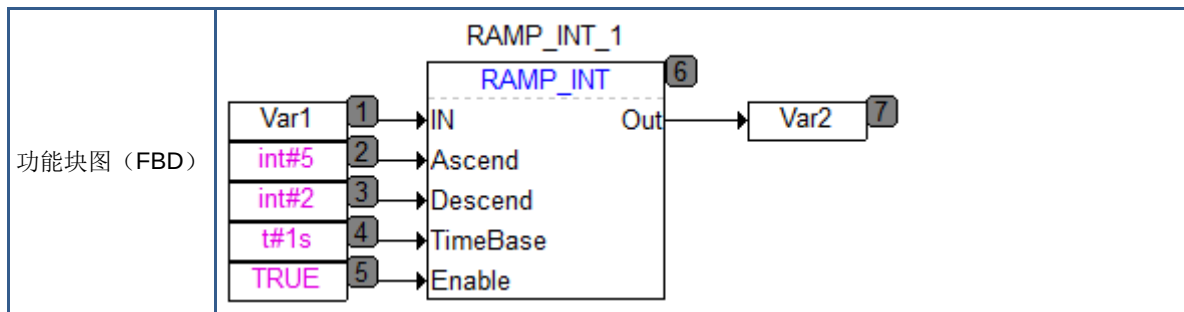
int#5

int#2

t#1s

TRUE

Var2



上例中，当指令执行时，根据输入值 IN 的变化，对应的输出值如图 122 所示。

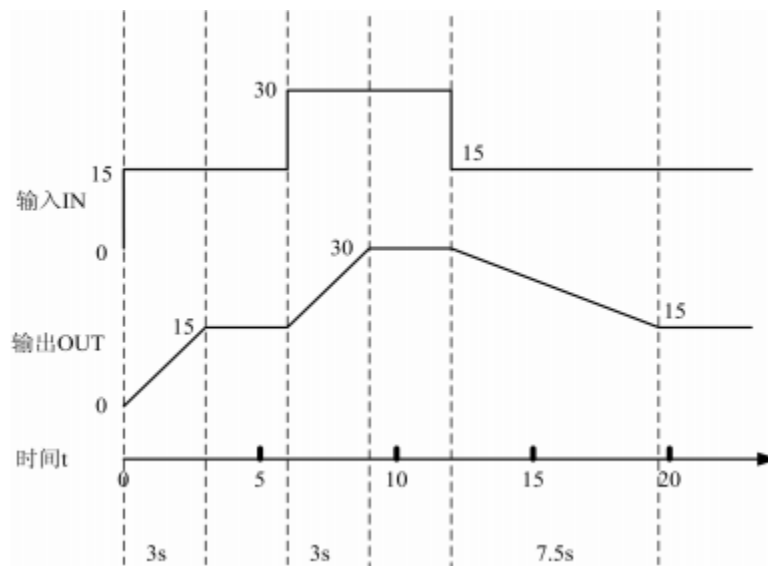


图 122 IN 与 OUT 对应关系

2.3.6.2 RAMP_REAL（实型限速）

1. 功能

RAMP_REAL 指令可以限制实型输入函数的升降速度。

输出 OUT 是输入 IN 经上升和下降限制整定后的函数值。当 TIMEBASE 设置为 t#0s, ASCEND 和 DESCEND 与时间间隔无关；当 TIMEBASE 设置为 t#0s 之外的值时，ASCEND 和 DESCEND 是时间间隔 TIMEBASE 时间长短的限制值。

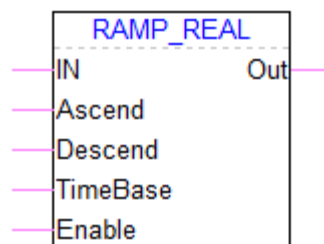


图 123 RAMP_REAL 功能块

2. 参数说明

输入参数	数据类型	功能描述	初始值（默认）	掉电保护
IN	REAL	被操作变量	0	FALSE
Ascend	REAL	时间间隔内最大上升值	0	TRUE
DEScend	REAL	时间间隔内最大下降值	0	TRUE
TimeBase	TIME	时间间隔	T#1S	TRUE
Enable	BOOL	速率限幅使能开关	TRUE	TRUE

输出参数	数据类型	功能描述	初始值（默认）	掉电保护
Out	REAL	经上升和下降限制后的值	0	TRUE

- 在 LD 程序中使用该功能块时，EN 使能引脚使用注意事项：
- 当 EN 使能引脚从 TRUE 变为 FALSE 后，其内部定时器继续计时，当 EN 再次变为 TRUE 时，EN 在 FALSE 期间的的时间累计值生效，因此输出可能出现跃变。

3. 指令使用举例

请参见 2.3.6.1 RAMP_INT（整型限速）。

2.3.6.3 CHARCURVE（特征曲线）

1. 功能

IN 是 INT 类型，输入要操作的值。N 是字节类型，决定了特征曲线的点数且 $2 \leq N \leq 11$ 。在 INT 类型数组 PX[0..10]和 PY[0..10]中产生一个特征行。INT 型输出 OUT 为操作后的值。字节型变量 ERR 输出错误信息。

数组中的点 PX[0]..PX[N-1]必须根据从小到大存储，否则 ERR 的值为 1。如果 $IN < PX[0]$ ，ERR=2； $IN > PX[N-1]$ ，ERR=3，输出包含相应的限制值 PY[0]或 PY[N-1]。

如果 N 超出了 2 和 11 之间的允许值，ERR=4。



图 124 CHARCURVE 功能块

2. 参数说明

输入参数	数据类型	功能描述	初始值（默认）	掉电保护
IN	INT	被操作变量	0	FALSE
N	BYTE	特征曲线的点数	0	TRUE

输出参数	数据类型	功能描述	初始值（默认）	掉电保护
OUT	INT	操作后的值	0	FALSE
ERR	BYTE	错误信息	0	FALSE

其它参数	数据类型	功能描述	初始值（默认）	掉电保护
PX	ARRAY[0..10] OF INT	特征曲线的 X 轴点列	0	TRUE
PY	ARRAY[0..10] OF INT	特征曲线的 Y 轴点列	0	TRUE

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	CHARCURVE_1		CHARCURVE		G区
0002	Varint1		INT	0	G区
0003	Varbyte1		BYTE	0	G区

编程语言	程序
梯形图（LD）	
功能块图（FBD）	

2.3.6.4 HYSTERESIS（滞后）

1. 功能

当输入值低于低限值时，输出为 TRUE；当输入值高于高限值时，输出为 FALSE；当输入值在高低限制之间时，输出保持。

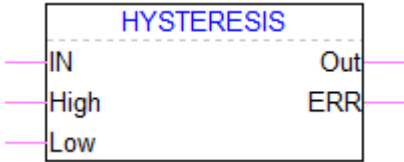


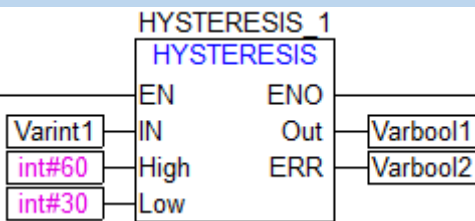
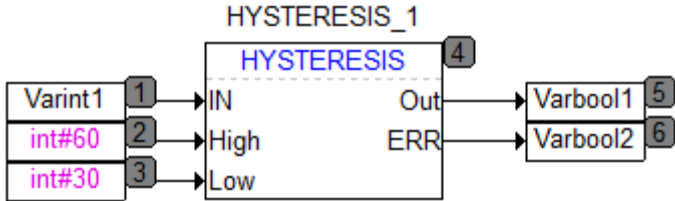
图 125 HYSTERESIS 功能块

2. 参数说明

输入参数	数据类型	功能描述	初始值（默认）	掉电保护
IN	INT	输入值	0	FALSE
Low	INT	低限值	0	TRUE
High	INT	高限值	0	TRUE

输出参数	数据类型	功能描述	初始值（默认）
Out	BOOL	输出，低下限值后为 TRUE，直到上限值后恢复为 FALSE	TRUE
ERR	BOOL	输入的高低限颠倒时，ERR 报警	FALSE

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	HYSTERESIS_1		HYSTERESIS		G区
0002	Varint1		INT	0	G区
0003	Varbool1		BOOL	FALSE	G区
0004	Varbool2		BOOL	FALSE	G区
编程语言		程序			
梯形图（LD）					
功能块图（FBD）					

上例中，当指令执行时，根据输入值的变化，对应的输出值如图 126 所示。

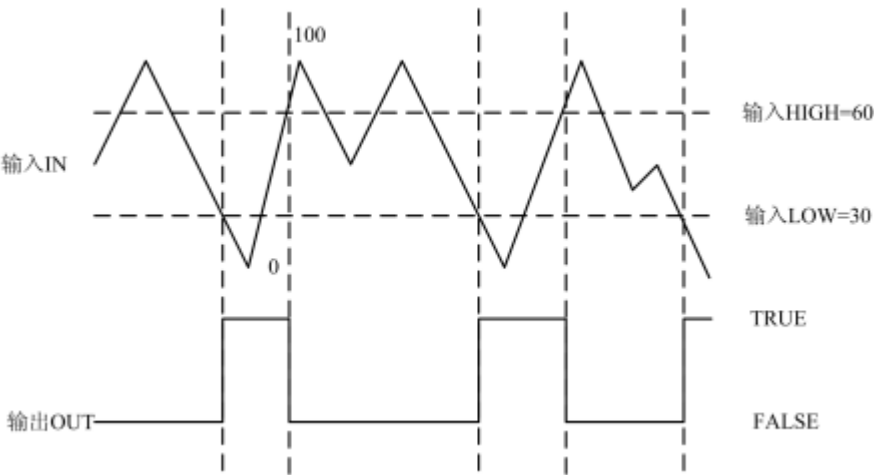


图 126 输入与输出变化关系

2.3.6.5 LIMITALARM（上下限报警）

1. 功能

当输入值低于低限值时，输出 U 为 TRUE；当输入值高于高限值时，输出 O 为 TRUE；当输入值在高低限制之间时，输出 IL 为 TRUE。

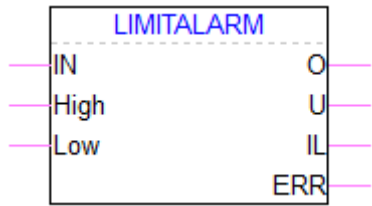


图 127 LIMITALARM 功能块

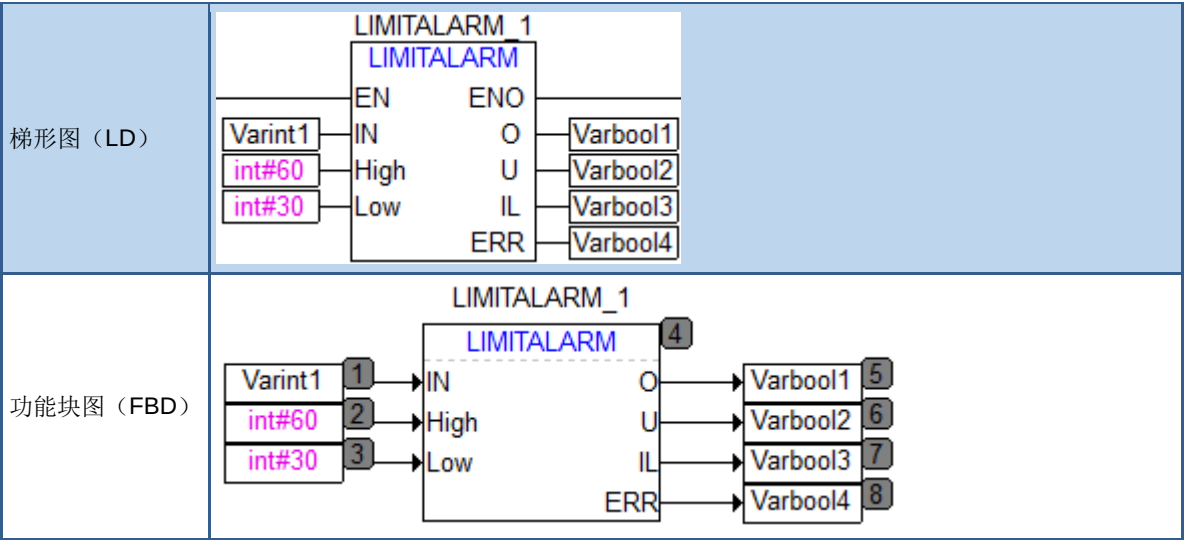
2. 参数说明

输入参数	功能描述	数据类型	初始值（默认）	掉电保护
IN	输入值	INT	0	FALSE
Low	低限值	INT	0	TRUE
High	高限值	INT	0	TRUE

输出参数	功能描述	数据类型	初始值（默认）	掉电保护
O	输入大于高限值	BOOL	FALSE	FALSE
U	输入小于低限值	BOOL	FALSE	FALSE
IL	输入在高低限之间	BOOL	FALSE	FALSE
ERR	输入的高低限颠倒时，ERR 报警	BOOL	FALSE	FALSE

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	LIMITALARM_1		LIMITALARM		G区
0002	Varint1		INT	0	G区
0003	Varbool1		BOOL	FALSE	G区
0004	Varbool2		BOOL	FALSE	G区
0005	Varbool3		BOOL	FALSE	G区
0006	Varbool4		BOOL	FALSE	G区
编程语言		程序			



上例中，当指令执行时，根据输入值的变化，对应的输出值如图 128 所示。

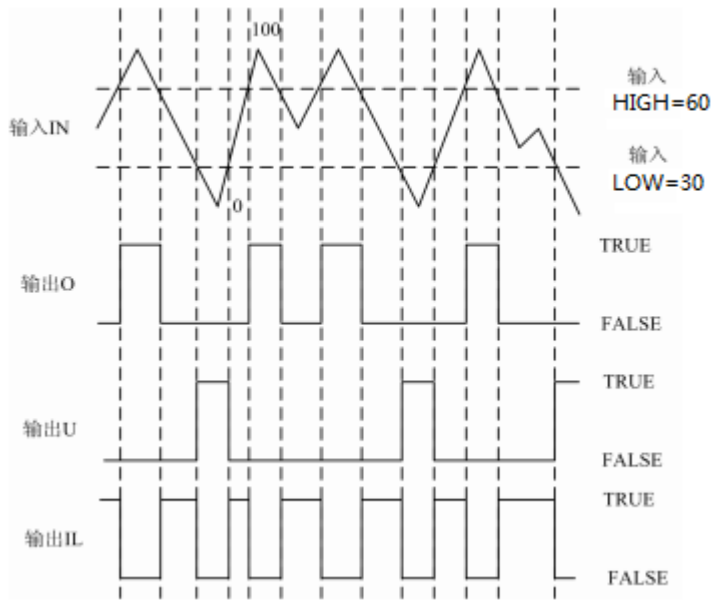


图 128 输入与输出变化关系

2.3.6.6 HEX_ENGIN（16 进制数转换为工程量数据）

1. 功能

将输入的码值 WH 转换成对应的工程量 AV 值。

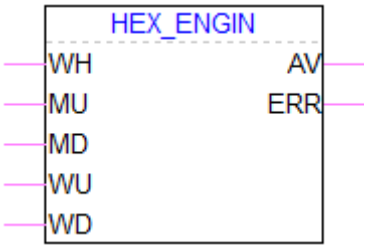


图 129 HEX_ENGIN 功能块

2. 参数说明

输入参数	说明	数据类型	初始值（默认）	掉电保护
WH	输入码值	WORD	0	FALSE
MU	工程量上限值	REAL	0	TRUE
MD	工程量下限值	REAL	0	TRUE
WU	PLC 的模拟量输入码值上限	WORD	0	TRUE
WD	PLC 的模拟量输入码值下限	WORD	0	TRUE

输出参数	说明	数据类型	初始值（默认）	掉电保护
AV	转换后的工程量	REAL	0	FALSE
ERR	工程量上下限颠倒时，ERR 报警	BOOL	FALSE	FALSE

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	HEX_ENGIN_1		HEX_ENGIN		G区
0002	Varword1		WORD	0	G区
0003	Varreal1		REAL	0	G区
0004	Varbool1		BOOL	FALSE	G区

编程语言

程序

梯形图（LD）

HEX_ENGIN 1

HEX_ENGIN

EN ENO

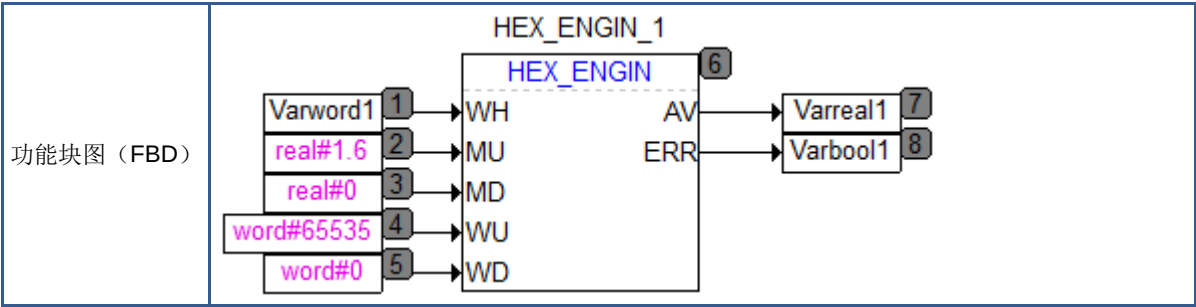
Varword1 WH AV Varreal1

real#1.6 MU ERR Varbool1

real#0 MD

word#65535 WU

word#0 WD



程序说明：

该指令一般用于模拟量输入数据处理；

EN 置位时，进行转换，输入数据为 Varword1，工程量上限是 1.6 Mpa，工程量下限是 0 Mpa，对应模块码值范围为 0-65,535。假设当 Varword1 的值为 32,767 时，对应输出值为 0.799。

2.3.6.7 ENGIN_HEX（工程量数据转换为 16 进制数据）

1. 功能

将输入的工程量 AV 转换成相应的码值 WH。

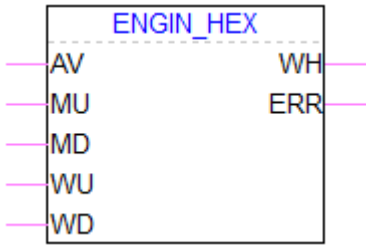


图 130 ENGIN_HEX 功能块

2. 参数说明

输入参数	说明	数据类型	初始值（默认）	掉电保护
AV	输入工程量	REAL	0	FALSE
MU	工程量上限值	REAL	0	TRUE
MD	工程量下限值	REAL	0	TRUE
WU	PLC 的模拟量输入码值上限	WORD	0	TRUE
WD	PLC 的模拟量输入码值下限	WORD	0	TRUE

输出参数	说明	数据类型	初始值（默认）	掉电保护
WH	转换后的码值	WORD	0	FALSE
ERR	工程量上下限颠倒时，ERR 报警	BOOL	FALSE	FALSE

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	ENGIN_HEX_1		ENGIN_HEX		G区
0002	Varreal1		REAL	0	G区
0003	Varword1		WORD	0	G区
0004	Varbool1		BOOL	FALSE	G区

编程语言	程序
梯形图 (LD)	
功能块图 (FBD)	

程序说明：

该指令一般用于模拟量输出数据处理；EN 置位时，进行转换，工程量实际数据为 Varreal1，工程量上限是 50 Hz，工程量下限是 0 Hz，对应模块范围为 0~65,535，当 Varreal1 的值为 25 Hz 时，对应 Varword1 输出值为 32,768。

2.4 系统库

2.4.1 时间相关指令

2.4.1.1 GetTickCnt（获取开机计数）

1. 功能

获取控制器自开机以来的时间。时间范围为 0~2³² 微秒，若超过上限，则从 0 开始计数。

2. 参数说明

	数据项	数据类型	参数描述
输出端	GetTickCnt	DWORD	返回控制器开机以来的所经过的时间（单位为微秒）

2.4.1.2 TimeDelay（时间延时）

1. 功能

延迟 IEC 执行时间。

2. 参数说明

	数据项	数据类型	参数描述
输入端	uiTimeDelay	DWORD	延时时间（单位为毫秒），默认值为 0
输出端	TimeDelay	DINT	0：执行延时成功；-1：执行延时失败

2.4.1.3 SET_RTC（设置实时时钟）

1. 功能

设置 PLC 实时时钟的起始时间。仅设置主控制器 RTC，从控制器自动冗余主控制器 RTC。

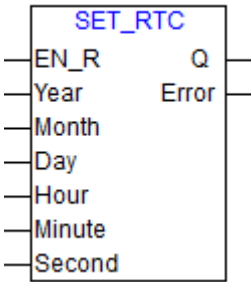


图 131 SET_RTC 功能块

2. 参数说明

输入参数	数据类型	功能描述	参数值说明	默认值
EN_R	BOOL	使能	FALSE：无效 TRUE：上升沿有效	FALSE
Year	WORD	年	范围 1971~2036	2011
Month	BYTE	月	1~12	1
Day	BYTE	日	1~31	1
Hour	BYTE	时	0~23	0
Minute	BYTE	分	0~59	0
Second	BYTE	秒	0~59	0

输出参数	数据类型	功能描述	参数值说明	默认值
Q	BOOL	设置完成	0：未完成 1：已完成	FALSE
Error	BYTE	参数设置错误	=0：设置 RTC 时间没有错误 非 0：表示设置 RTC 时间时出现错误 =1：设置年超出规定范围 =2：设置月不合理	0

			=3: 设置日不合理 =4: 设置时不合理 =5: 设置分不合理 =6: 设置秒不合理 =200: 模块重复定义并调用	
--	--	--	---	--

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	SET_RTC_1		SET_RTC		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Varbool2		BOOL	FALSE	G区
0004	Varbyte1		BYTE	0	G区

编程语言

程序

梯形图 (LD)

功能块图 (FBD)

2.4.1.4 GET_RTC（读取实时时钟）

1. 功能

该功能块读取 PLC 实时时钟的时间。

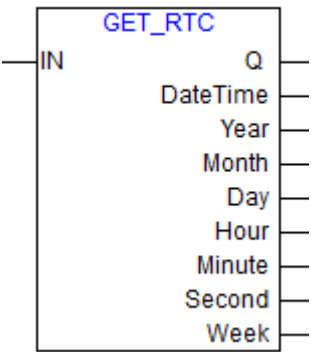


图 132 GET_RTC 功能块

2. 参数说明

输入参数	数据类型	功能描述	参数值说明	默认值
IN	BOOL	使能	0: 无效 1: 有效	0

输出参数	数据类型	功能描述	参数值说明	默认值
Q	BOOL	是否读出数据	0: 未读出数据 1: 正确取出数据	FALSE
DateTime	DT	日期/时间	DT#1971-01-01-00:00:00~DT#2036-12-31-23:59:59	DT#1970-01-01-00:00:00
Year	WORD	年	1971-2036	0
Month	BYTE	月	1-12	0
Day	BYTE	日	1-31	0
Hour	BYTE	时	0-23	0
Minute	BYTE	分	0-59	0
Second	BYTE	秒	0-59	0
Week	BYTE	星期	1-7	0

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	GET_RTC_1		GET_RTC		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Vardt1		DT	DT#1970-01-01-00:00:00	G区
0004	Varword1		WORD	0	G区
0005	Varbyte1		BYTE	0	G区
0006	Varbyte2		BYTE	0	G区
0007	Varbyte3		BYTE	0	G区
0008	Varbyte4		BYTE	0	G区
0009	Varbyte5		BYTE	0	G区
0010	Varbyte6		BYTE	0	G区

编程语言	程序
梯形图（LD）	GET_RTC_1 GET_RTC EN IN Q DateTime Year Month Day Hour Minute Second Week Varbool1 Vardt1 Varword1 Varbyte1 Varbyte2 Varbyte3 Varbyte4 Varbyte5 Varbyte6
功能块图（FBD）	GET_RTC_1 GET_RTC 1 TRUE IN 2 Q DateTime Year Month Day Hour Minute Second Week 3 Varbool1 4 Vardt1 5 Varword1 6 Varbyte1 7 Varbyte2 8 Varbyte3 9 Varbyte4 10 Varbyte5 11 Varbyte6

程序说明：

IN 置位时，DateTime 以 DT 型数据格式输出 PLC 实时时钟的日期、时间，Year 以 WORD 型数据格式输出年，Month、Date、Hour、Minute、Second、Week 分别以 BYTE 型数据格式输出月、日、时、分、秒、星期，Q 等于 TRUE。

EN 复位时，Q 等于 FALSE，DateTime、Year、Month、Date、Hour、Minute、Second、Week 保持最后一次输出值不变。

2.4.2 模块信息指令

2.4.2.1 sysGetCPUModuleVern（获取控制器的固件版本信息）

1. 功能

获取控制器的固件版本信息。

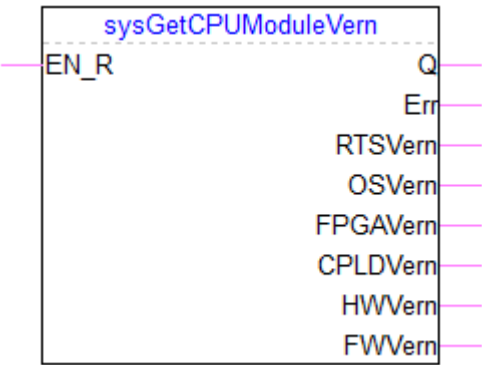


图 133 sysGetCPUModuleVern 指令

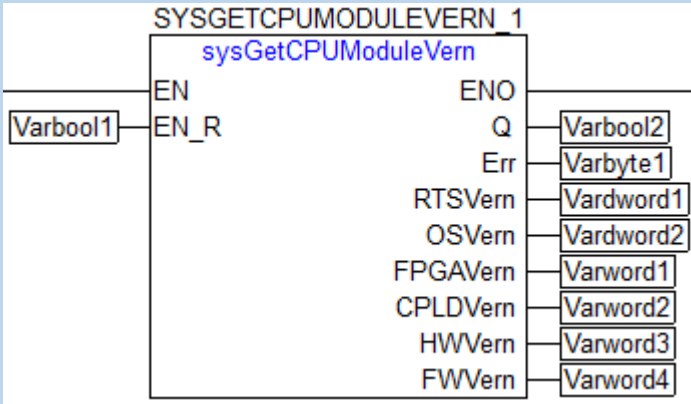
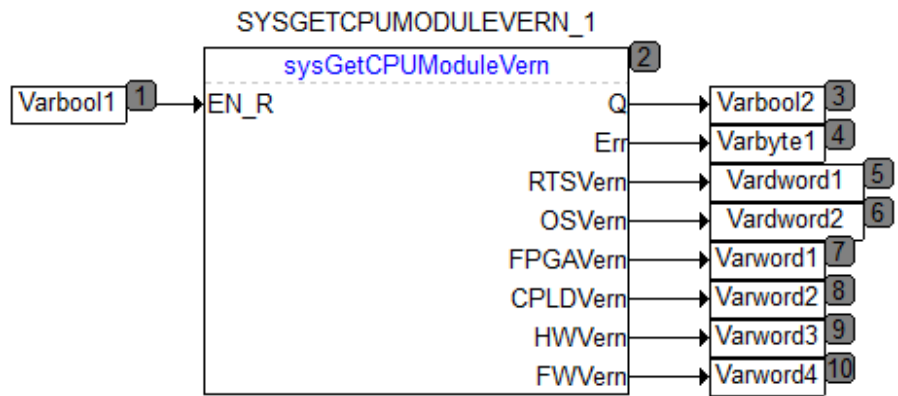
2. 参数说明

输入参数	类型	参数描述
EN_R	BOOL	使能 FALSE: 无效（默认值） TRUE: 上升沿有效

输出参数	类型	参数描述
Q	BOOL	输出（默认值 FALSE） TRUE: 指令执行成功 FALSE: 指令执行失败
Err	BYTE	错误（默认值 0） 128: 获取版本失败
RTSVern	DWORD	RTS 固件版本 build 号 高 16 位为核 1 RTS 版本 bulid 号，低 16 为核 0 RTS 版本 bulid 号
OSVern	DWORD	OS 版本 build 号 高 16 位为核 1 的 OS 版本 bulid 号，低 16 位为核 0 的 OS 版本 bulid 号
FPGAVern	WORD	FPGA 版本 build 号
CPLDVern	WORD	CPLD 版本 build 号
HVVern	WORD	硬件版本号 返回值为 1~26，对应 A~Z
FWVern	WORD	固件版本号 对应“LK220S-A01”中的“01”

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	SYSGETCPUMODULEVERN_1		SYSGETCPUMODULEVERN		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Varbool2		BOOL	FALSE	G区
0004	Varbyte1		BYTE	0	G区
0005	Vardword1		DWORD	0	G区
0006	Vardword2		DWORD	0	G区
0007	Varword1		WORD	0	G区
0008	Varword2		WORD	0	G区
0009	Varword3		WORD	0	G区
0010	Varword4		WORD	0	G区

编程语言	程序
梯形图（LD）	
功能块图（FBD）	

2.4.2.2 sysGetComModuleVern（获取通信模块固件版本信息）

1. 功能
- 获取通信模块固件版本信息。

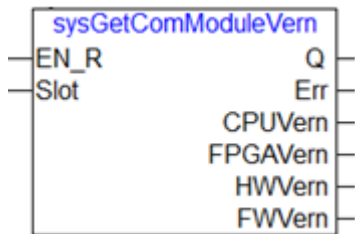


图 134 sysGetComModuleVern 指令

2. 参数说明

输入参数	类型	含义
EN_R	BOOL	使能 FALSE: 无效（默认值） TRUE: 上升沿有效
Slot	BYTE	槽位号（默认值 2，有效值范围 2-5） 主控模块槽位号固定为 1 其它模块依据与主控位置向右递增 使用时输入想得到版本信息模块所在的槽位号

输出参数	类型	参数描述
Q	BOOL	输出（默认值 FALSE） TRUE: 指令执行成功 FALSE: 指令执行失败
Err	BYTE	错误（默认值 0） 0: 正确 1: Slot 配置错误 128: 对应槽位上不存在模块 129: 获取失败 200: 模块重复定义并调用
CPUVern	WORD	CPU 固件版本 build 号
FPGAVern	WORD	FPGA 版本 build 号
HWVern	WORD	硬件版本号 返回值为 1~26，对应 A~Z
FWVern	WORD	固件版本号 对应“LK240S-A01”中的“01”

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	SYSGETCOMMODULEVERN_1		SYSGETCOMMODULEVERN		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Varbool2		BOOL	FALSE	G区
0004	Varbyte1		BYTE	0	G区
0005	Varword1		WORD	0	G区
0006	Varword2		WORD	0	G区
0007	Varword3		WORD	0	G区
0008	Varword4		WORD	0	G区

编程语言	程序
梯形图 (LD)	SYSGETCOMMODULEVERN_1 sysGetComModuleVern EN Varbool1 EN_R byte#3 Slot ENO Q Err CPUVern FPGAVern HWVern FWVern Varbool2 Varbyte1 Varword1 Varword2 Varword3 Varword4

输入参数	类型	参数描述
EN_R	BOOL	使能 FALSE: 无效（默认值） TRUE: 上升沿有效
Slot	BYTE	槽位号（默认值 1，有效值 1-5） 主控模块槽位号固定为 1 其它模块依据与主控位置向右递增

输出参数	类型	参数描述
Q	BOOL	输出（默认值 FALSE） TRUE: 指令执行成功 FALSE: 指令执行失败
Err	BYTE	错误（默认值 0） 0: 正确 1: Slot 配置错误 128: 对应槽位上不存在模块 129: 获取失败 130: 年份错误（2015~2099） 131: 月份错误 132: 日期错误 133: 小时错误 134: 分钟错误 200: 模块重复定义并调用
Year	WORD	年（默认值 0）
Month	BYTE	月（默认值 0）
Day	BYTE	日（默认值 0）
Hour	BYTE	时（默认值 0）
Minute	BYTE	分（默认值 0）

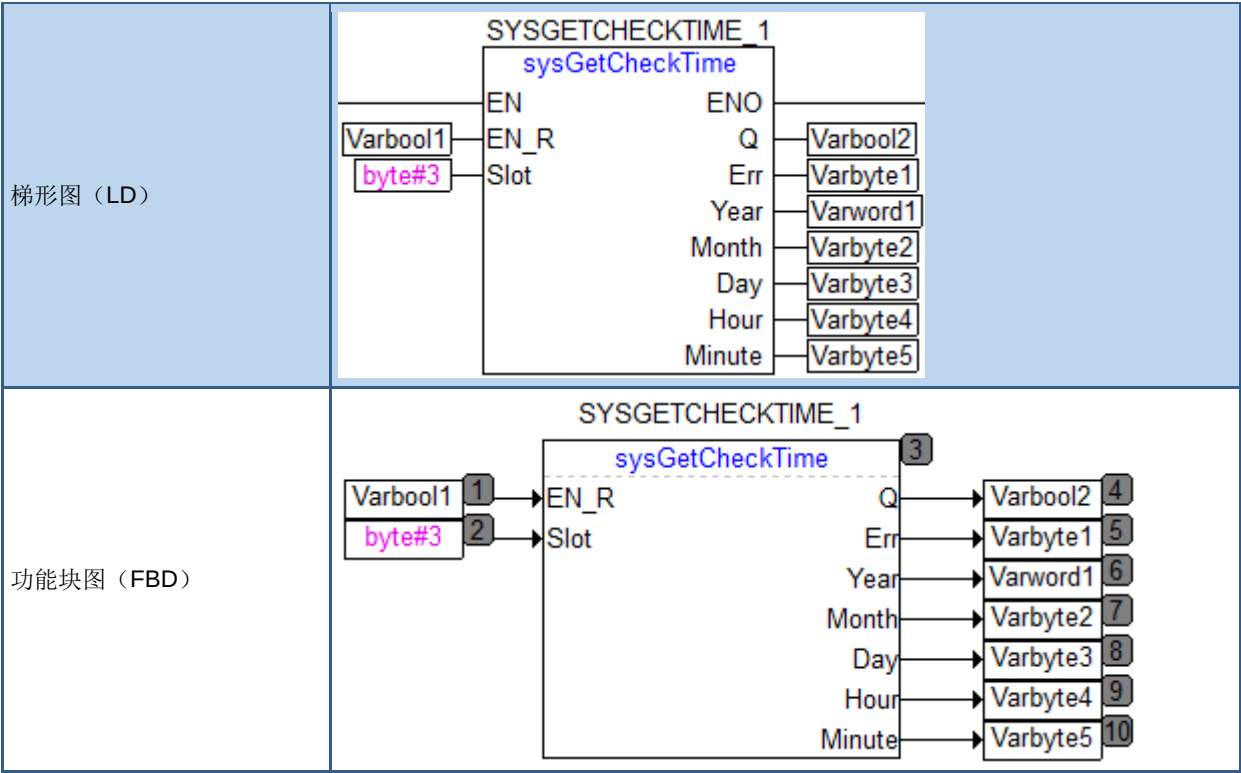
3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	SYSGETCHECKTIME_1		SYSGETCHECKTIME		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Varbool2		BOOL	FALSE	G区
0004	Varbyte1		BYTE	0	G区
0005	Varword1		WORD	0	G区
0006	Varbyte2		BYTE	0	G区
0007	Varbyte3		BYTE	0	G区
0008	Varbyte4		BYTE	0	G区
0009	Varbyte5		BYTE	0	G区

编程语言

程序



2.4.2.4 sysGetEthMac (获取 Mac 地址)

- 1. 功能
- 获取 Mac 地址。

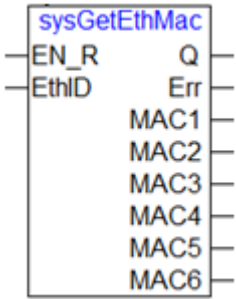


图 136 sysGetEthMac 指令

2. 参数说明

输入参数	类型	参数描述
EN_R	BOOL	使能 FALSE: 无效 (默认值) TRUE: 上升沿有效
EthID	BYTE	网卡 ID 号 (默认值 0, 有效值为 0-1) 0: 第 1 块网卡 1: 第 2 块网卡

输出参数	类型	参数描述
Q	BOOL	输出（默认值 FALSE） TRUE：指令执行成功 FALSE：指令执行失败
Err	BYTE	错误（默认值 0） 1：EthID 配置错误 128：获取 mac 失败 200：模块重复定义并调用
MAC1	BYTE	MAC 地址第 1 部分
MAC2	BYTE	MAC 地址第 2 部分
MAC3	BYTE	MAC 地址第 3 部分
MAC4	BYTE	MAC 地址第 4 部分
MAC5	BYTE	MAC 地址第 5 部分
MAC6	BYTE	MAC 地址第 6 部分

3. 指令使用举例

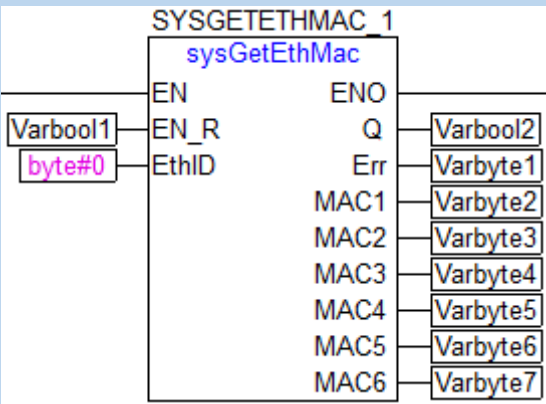
变量定义

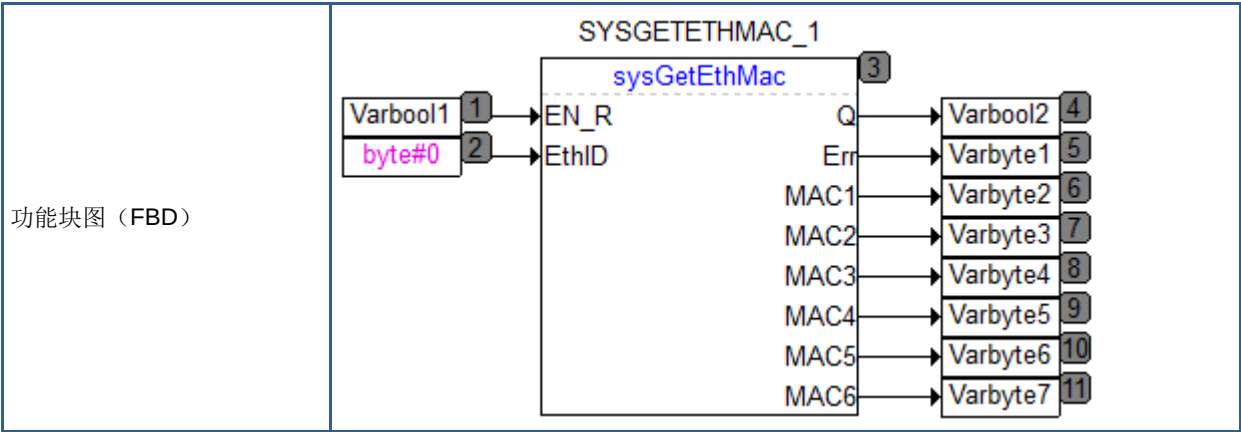
序号	变量名	变量说明	变量类型	初始值	区域
0001	SYSGETETHMAC_1		SYSGETETHMAC		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Varbool2		BOOL	FALSE	G区
0004	Varbyte1		BYTE	0	G区
0005	Varbyte2		BYTE	0	G区
0006	Varbyte3		BYTE	0	G区
0007	Varbyte4		BYTE	0	G区
0008	Varbyte5		BYTE	0	G区
0009	Varbyte6		BYTE	0	G区
0010	Varbyte7		BYTE	0	G区

编程语言

程序

梯形图（LD）





程序说明：
获取第 1 块网卡的 MAC 地址。

2.4.2.5 sysGetBatteryAlarm（获取电池电量报警）

1. 功能
- 获取主控模块电池电量信息。

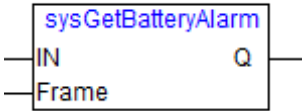


图 137 sysGetBatteryAlarm 指令

2. 参数说明

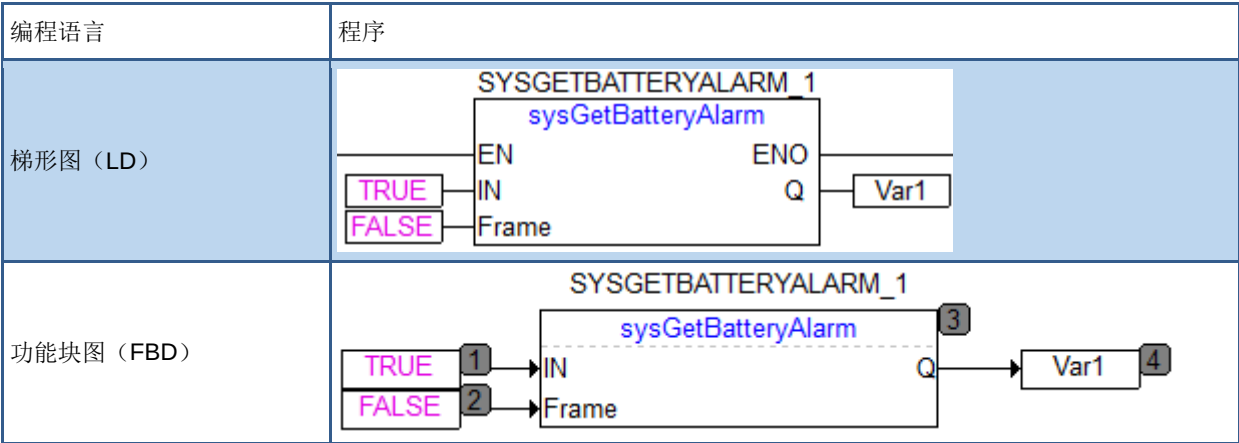
输入参数	类型	含义
IN	BOOL	高电平使能（默认值 FALSE）
Frame	BOOL	机架选择（默认值 FALSE） FALSE：本机架（主） TRUE：对方机架（从）

输出参数	类型	含义
Q	BOOL	电池电量 TURE：没电池或者电池电量不足，需更换电池 FALSE：电池电量正常

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	SYSGETBATTERYALARM_1		SYSGETBATTERYALARM		G区
0002	Var1		BOOL	FALSE	G区



2.4.2.6 sysGetTEMP（获取主控 CPU 温度）

1. 功能

获取主控模块温度信息。

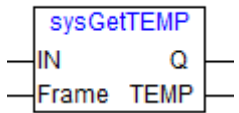


图 138 sysGetTEMP 指令

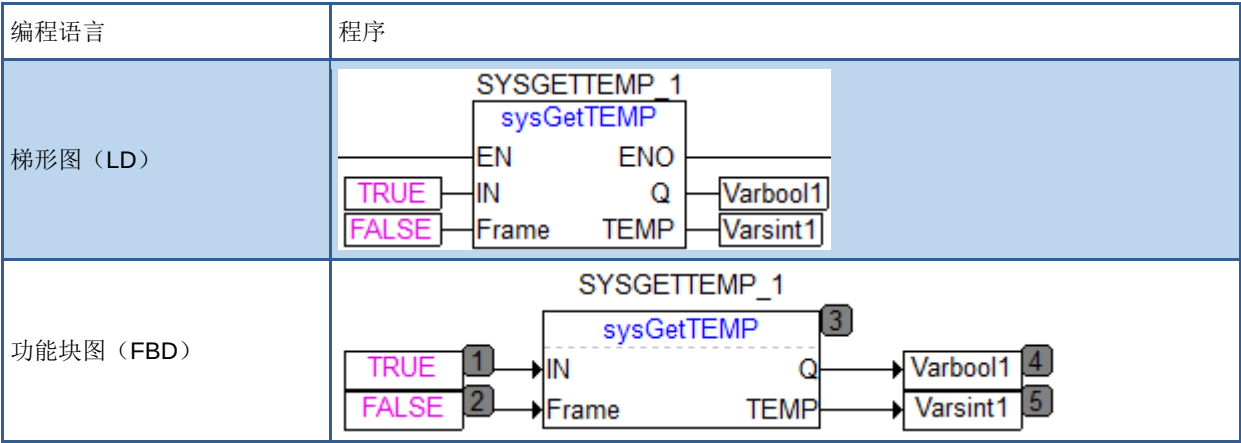
2. 参数说明

输入参数	类型	含义
IN	BOOL	高电平使能（默认值 FALSE）
Frame	BOOL	机架选择（默认值 FALSE） FALSE: 本机架（主） TRUE: 对方机架（从）

输出参数	类型	含义
Q	BOOL	输出（默认值 FALSE） TRUE: 指令执行成功 FALSE: 指令执行失败
TEMP	SINT	温度值，单位为摄氏度（默认值为 0）

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	SYSGETTEMP_1		SYSGETTEMP		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Varsint1		SINT	0	G区



2.4.2.7 sysGetModel (获取模块型号)

1. 功能
- 获取模块型号。

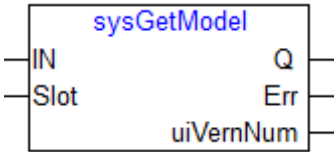


图 139 sysGetModel 指令

2. 参数说明

输入参数	类型	含义
IN	BOOL	高电平使能（默认值 FALSE）
Slot	BYTE	槽位号（默认值 1） 主控模块槽位号固定为 1 其它模块依据与主控位置向右递增

输出参数	类型	含义
Q	BOOL	输出（默认值 FALSE） TRUE：指令执行成功 FALSE：指令执行失败
Err	BYTE	错误（默认值 0） 1：槽号 slot 错误 128：对应槽上不存在模块
uiVernNum	BYTE	模块型号编号（默认值 0） 1：LK220S 2：LK249S 3：LK240S

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	SYSGETMODEL_1		SYSGETMODEL		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Varbyte1		BYTE	0	G区
0004	Varbyte2		BYTE	0	G区

编程语言	程序
梯形图 (LD)	SYSGETMODEL_1 sysGetModel EN ENO Q Err uiVernNum IN Slot TRUE byte#2 Varbool1 Varbyte1 Varbyte2
功能块图 (FBD)	SYSGETMODEL_1 sysGetModel IN Slot Q Err uiVernNum 1 2 3 4 5 6 TRUE byte#2 Varbool1 Varbyte1 Varbyte2

2.4.2.8 sysGetModuleSN（获取模块序列号）

1. 功能

获取模块序列号。

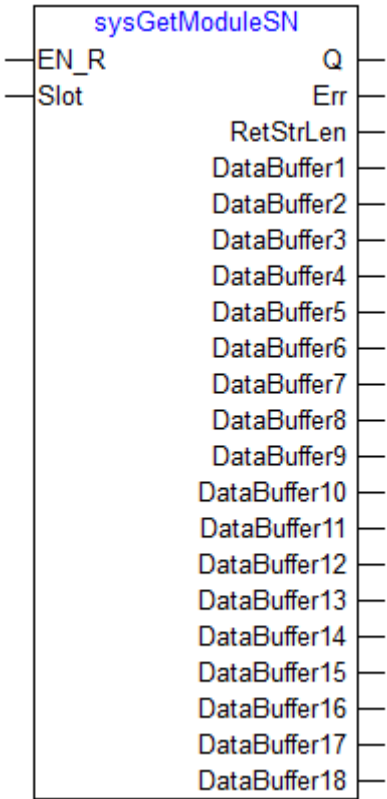


图 140 sysGetModuleSN 指令

2. 参数说明

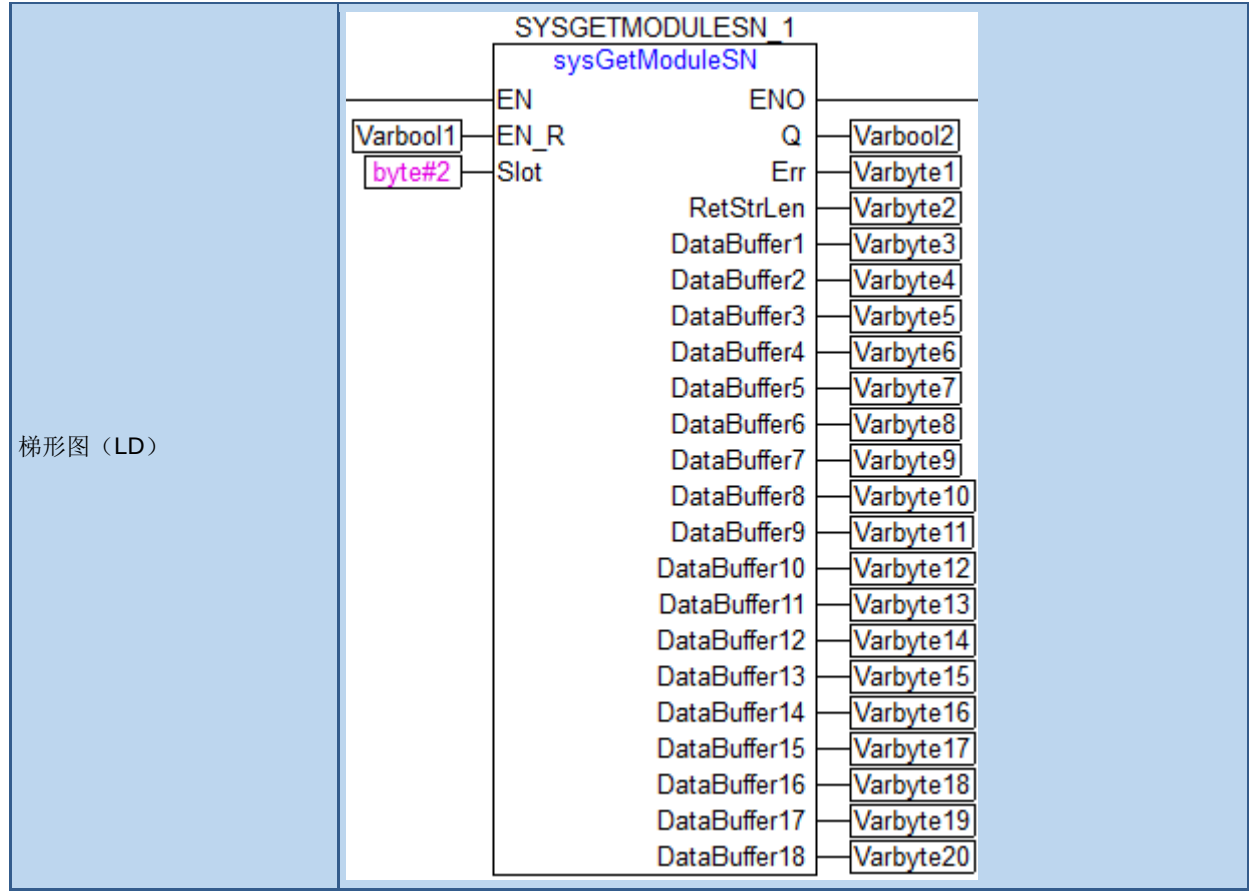
输入参数	类型	含义
EN_R	BOOL	使能 FALSE: 无效（默认值） TRUE: 上升沿有效
Slot	BYTE	槽位号（默认值 1，有效值为 1-5） 主控模块槽位号固定为 1 其它模块依据与主控位置向右递增

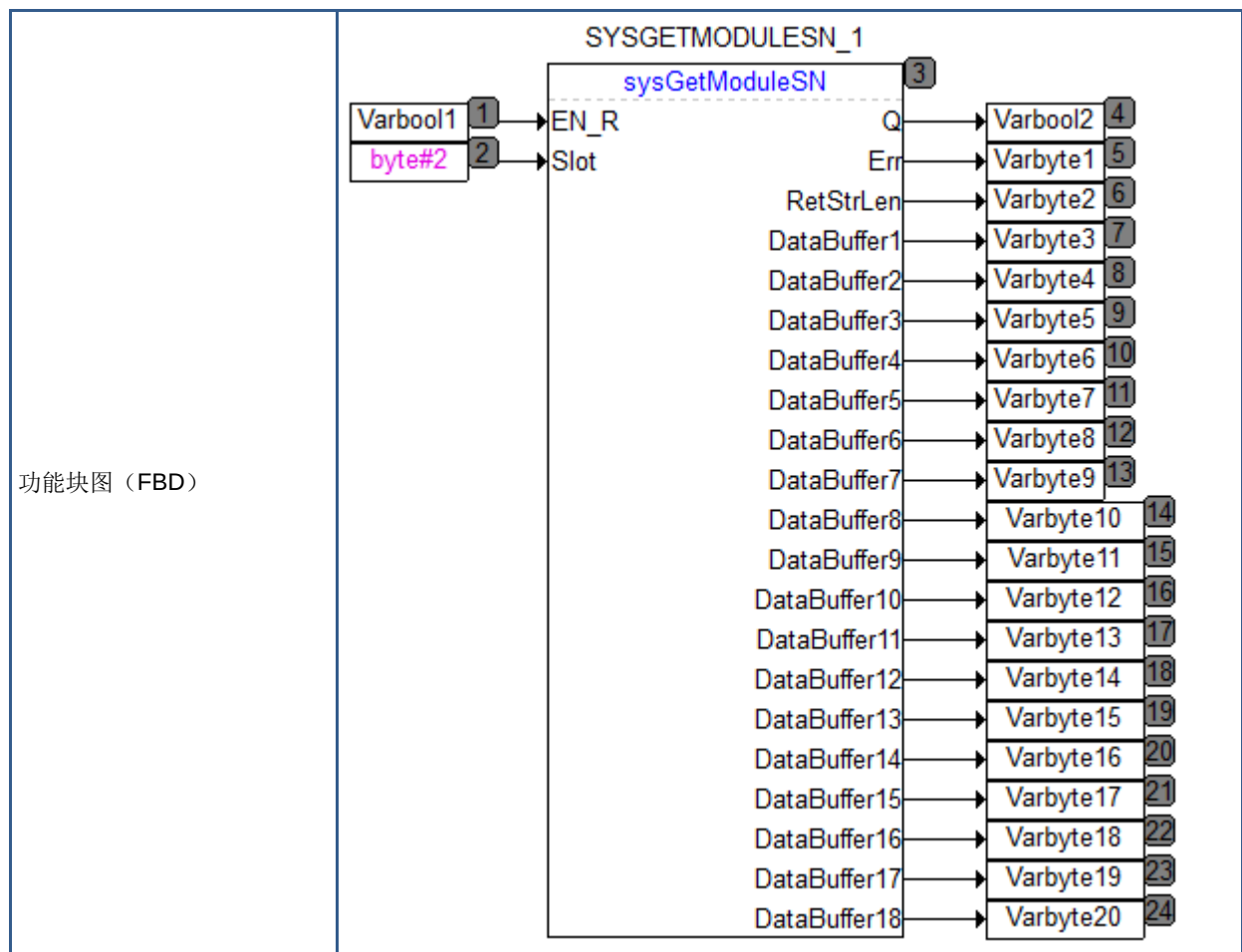
输出参数	类型	含义
Q	BOOL	输出（默认值 FALSE） TRUE: 指令执行成功 FALSE: 指令执行失败
Err	BYTE	错误（默认值 0） 0: 正确 1: 槽号 slot 错误 128: 对应槽上不存在模块 130: 返回信息错误或读取过程中暂时未得到正确数据（最终读取成功后会变为 0） 200: 模块重复定义并调用
RetStrLen	BYTE	模块序列号长度（默认值 0，最长支持 18 字节的 ASCII 码）

uiDataBuffer[1]~ uiDataBuffer[18]	BYTE	模块序列号，共 18 个字节，一个字节代表一个 ASCII 码
--------------------------------------	------	---------------------------------

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	SYSGETMODULESN_1		SYSGETMODULESN		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Varbool2		BOOL	FALSE	G区
0004	Varbyte1		BYTE	0	G区
0005	Varbyte2		BYTE	0	G区
0006	Varbyte3		BYTE	0	G区
0007	Varbyte4		BYTE	0	G区
0008	Varbyte5		BYTE	0	G区
0009	Varbyte6		BYTE	0	G区
0010	Varbyte7		BYTE	0	G区
0011	Varbyte8		BYTE	0	G区
0012	Varbyte9		BYTE	0	G区
0013	Varbyte10		BYTE	0	G区
0014	Varbyte11		BYTE	0	G区
0015	Varbyte12		BYTE	0	G区
0016	Varbyte13		BYTE	0	G区
0017	Varbyte14		BYTE	0	G区
0018	Varbyte15		BYTE	0	G区
0019	Varbyte16		BYTE	0	G区
0020	Varbyte17		BYTE	0	G区
0021	Varbyte18		BYTE	0	G区
0022	Varbyte19		BYTE	0	G区
0023	Varbyte20		BYTE	0	G区
编程语言		程序			





2.4.3 系统状态指令

2.4.3.1 sysGetRunTime（获取任务运行时间）

- 1. 功能
获取控制器任务运行时间。

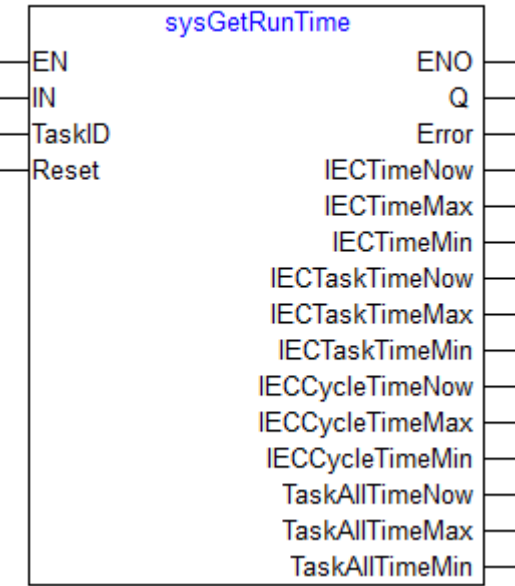


图 141 sysGetRunTime 指令

2. 参数说明

输入参数	类型	含义
IN	BOOL	高电平使能（默认值 FALSE）
TaskID	BYTE	IEC 任务 ID，1~8 LK220S 只支持一个 IEC 任务，ID 默认为 1；如果为 2~8 则前三组 9 个时间数据返回 0 值
Reset	BOOL	清零输出参数，高电平使能（默认值 FALSE）

输出参数	类型	含义
Q	BOOL	输出，默认值 FALSE TRUE：指令执行成功 FALSE：指令执行失败
Error	BYTE	错误，默认值 0 1：TaskID 配置错误
IECTimeNow	DWORD	当前组态逻辑本次运行时间，默认值 0
IECTimeMax	DWORD	当前组态逻辑最大运行时间，默认值 0
IECTimeMin	DWORD	当前组态逻辑最小运行时间，默认值 0
IECTaskTimeNow	DWORD	当前组态逻辑任务本次运行时间，默认值 0
IECTaskTimeMax	DWORD	当前组态逻辑任务最大运行时间，默认值 0
IECTaskTimeMin	DWORD	当前组态逻辑任务最小运行时间，默认值 0
IECCycleTimeNow	DWORD	当前组态逻辑任务周期本次运行时间，默认值 0
IECCycleTimeMax	DWORD	当前组态逻辑任务周期最大运行时间，默认值 0
IECCycleTimeMin	DWORD	当前组态逻辑任务周期最小运行时间，默认值 0

TaskAllTimeNow	DWORD	当前任务本次运行时间，默认值 0
TaskAllTimeMax	DWORD	当前任务最大运行时间，默认值 0
TaskAllTimeMin	DWORD	当前任务最小运行时间，默认值 0

- 参数详细说明
- ☐

IECTime: 当前工程中所有 POU 执行所用时间;
- ☐

IECTaskTime: 当前组态逻辑任务执行时间, 包含逻辑任务调度、自检时间, 不包含冗余时间;
- ☐

IECCycleTime: 组态逻辑任务执行周期;
- ☐

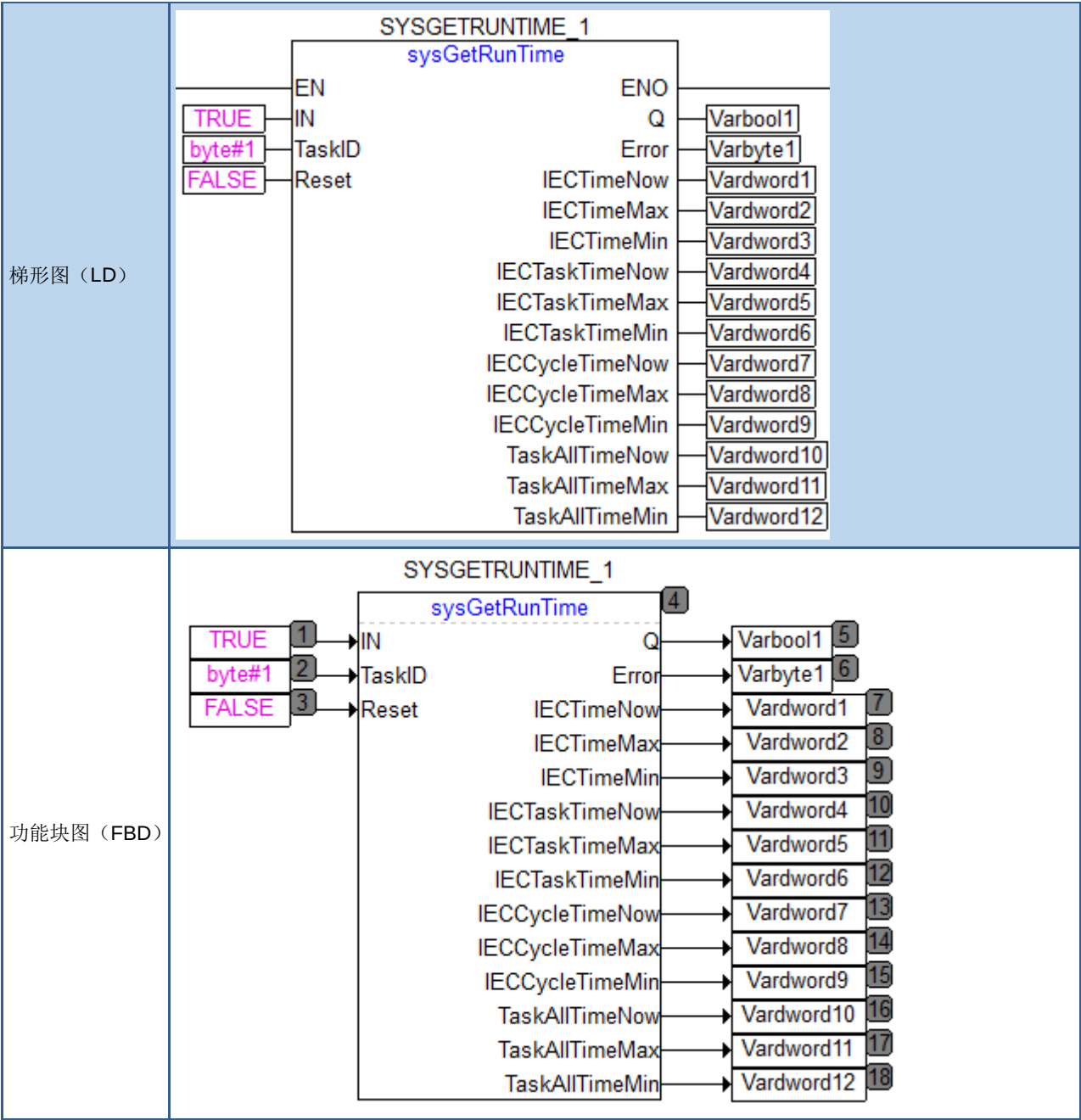
TaskAllTime: 当前组态逻辑任务所有执行时间, 包含逻辑任务调度、自检时间, 包含冗余时间。



- 当发生外部操作时, 例如改变拨码开关状态、AT 复位、AT 冷复位、AT 下装等, 建议用户将 **Reset** 引脚置为 **TRUE** 以复位该外部库输出值, 待数据变化趋于稳定后, 获取最新时间数据。

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	SYSGETRUNTIME_1		SYSGETRUNTIME ▾		G区 ▾
0002	Varbool1		BOOL ▾	FALSE ▾	G区 ▾
0003	Varbyte1		BYTE ▾	0	G区 ▾
0004	Vardword1		DWORD ▾	0	G区 ▾
0005	Vardword2		DWORD ▾	0	G区 ▾
0006	Vardword3		DWORD ▾	0	G区 ▾
0007	Vardword4		DWORD ▾	0	G区 ▾
0008	Vardword5		DWORD ▾	0	G区 ▾
0009	Vardword6		DWORD ▾	0	G区 ▾
0010	Vardword7		DWORD ▾	0	G区 ▾
0011	Vardword8		DWORD ▾	0	G区 ▾
0012	Vardword9		DWORD ▾	0	G区 ▾
0013	Vardword10		DWORD ▾	0	G区 ▾
0014	Vardword11		DWORD ▾	0	G区 ▾
0015	Vardword12		DWORD ▾	0	G区 ▾
编程语言		程序			



2.4.3.2 sysGetKeyState（获取钥匙开关状态）

1. 功能

获取机架 CPU 模块钥匙开关状态。

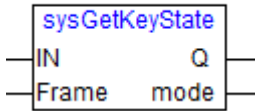


图 142 sysGetKeyState 指令

2. 参数说明

输入参数	类型	含义
IN	BOOL	高电平使能 (默认值 FALSE)
Frame	BOOL	机架选择 (默认值 FALSE) FALSE: 本机架 (主) TRUE: 对方机架 (从)

输出参数	类型	含义
Q	BOOL	输出 (默认值 FALSE) TRUE: 指令执行成功 FALSE: 指令执行失败
Mode	BYTE	钥匙开关状态 (默认值 0) 1: RUN, 运行模式 2: REM, 远程模式 3: PRG, 编程模式

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	SYSGETKEYSTATE_1		SYSGETKEYSTATE		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Varbyte1		BYTE	0	G区

编程语言

程序

梯形图 (LD)

功能块图 (FBD)

2.4.3.3 sysGetDPMasterState (获取 DP 主卡状态)

1. 功能

获取 DP 主卡诊断信息。

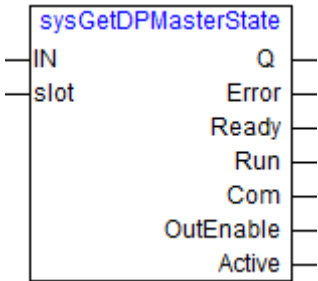


图 143 sysGetDPMasterState 指令

2. 参数说明

输入参数	类型	含义
IN	BOOL	高电平使能（默认值 FALSE）
Slot	BYTE	槽位号 （默认值 2，有效值为 2~5） 主控模块槽位号固定为 1 其它模块依据与主控位置向右递增

输出参数	类型	含义
Q	BOOL	输出，默认值 FALSE TRUE：指令执行成功 FALSE：指令执行失败
Error	BYTE	错误，默认值 0 0：正确 1：Slot 配置错误 128：在对应槽位上没有检测到 DP 主站
Ready	BOOL	主站初始化完成标识 FALSE：主站初始化未完成 TRUE：主站初始化完成
Run	BOOL	主站是否已运行标识 FALSE：主站未接收完下装数据 TRUE：主站接收下装数据完成，开始与从站通讯
Com	BOOL	主站是否与从站建立数据交换标识 FALSE：还未与从站建立数据交换 TRUE：至少与一个从站建立了数据交换
OutEnalbe	BOOL	主站输出使能标识 FALSE：输出不使能 TRUE：输出使能
Active	BOOL	主站活动状态标识 FALSE：热备状态 TRUE：活动状态

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	SYSGETDPMASSTATE_1		SYSGETDPMASSTATE ▾		G区 ▾
0002	Varbool1		BOOL ▾	FALSE ▾	G区 ▾
0003	Varbyte1		BYTE ▾	0 ▾	G区 ▾
0004	Varbool2		BOOL ▾	FALSE ▾	G区 ▾
0005	Varbool3		BOOL ▾	FALSE ▾	G区 ▾
0006	Varbool4		BOOL ▾	FALSE ▾	G区 ▾
0007	Varbool5		BOOL ▾	FALSE ▾	G区 ▾
0008	Varbool6		BOOL ▾	FALSE ▾	G区 ▾

编程语言	程序
梯形图 (LD)	SYSGETDPMASSTATE_1 sysGetDPMasterState EN Q Error Ready Run Com OutEnable Active IN slot Varbool1 Varbyte1 Varbool2 Varbool3 Varbool4 Varbool5 Varbool6
功能块图 (FBD)	SYSGETDPMASSTATE_1 sysGetDPMasterState 1 2 3 4 5 6 7 8 9 10 IN slot Q Error Ready Run Com OutEnable Active Varbool1 Varbyte1 Varbool2 Varbool3 Varbool4 Varbool5 Varbool6

2.4.3.4 sysGetCPULoad（获取控制器负荷）

1. 功能

获取控制器 CPU 和内存使用率。

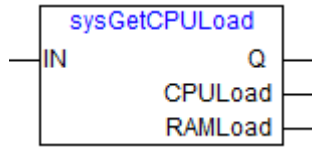


图 144 sysGetCPULoad 指令

2. 参数说明

输入参数	类型	含义
------	----	----

IN	BOOL	高电平使能（默认值 FALSE）
----	------	------------------

输出参数	类型	含义
Q	BOOL	输出（默认值 FALSE） TRUE：指令执行成功 FALSE：指令执行失败
CPUload	REAL	CPU 使用率（默认值 0），单位：%
RAMload	REAL	内存使用率（默认值 0），单位：%

- CPU 负荷：DP 数据输入至 DP 数据输出共用时间/组态逻辑任务执行周期。超 100%，则说明实际需要的任务周期大于设置的任务周期，此时需要增大任务周期。
- RAM 负荷：数据区的使用比例，即工程数据大小/数据区总大小。

3. 指令使用举例

变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	SYSGETCPULOAD_1		SYSGETCPULOAD		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Varreal1		REAL	0	G区
0004	Varreal2		REAL	0	G区

编程语言

程序

梯形图（LD）

功能块图（FBD）

2.4.3.5 sysGetRedState（获取系统冗余状态）

1. 功能

获取主从机架冗余状态信息。

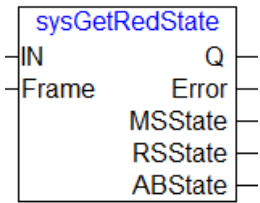


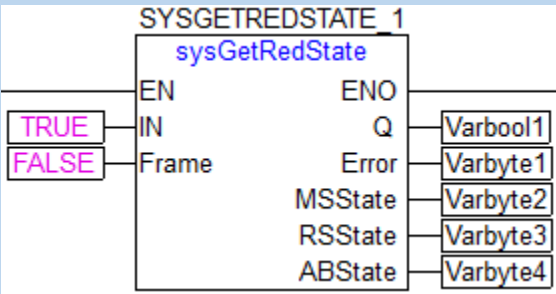
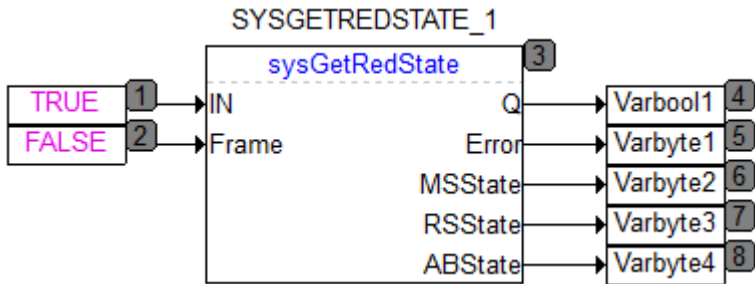
图 145 sysGetRedState 指令

2. 参数说明

输入参数	类型	含义
IN	BOOL	高电平使能（默认值 FALSE）
Frame	BOOL	机架选择（默认值 FALSE） FALSE: 本机架（主） TRUE: 对方机架（从）

输出参数	类型	含义
Q	BOOL	输出（默认值 FALSE） TRUE: 指令执行成功 FALSE: 指令执行失败
Error	BYTE	错误（默认值 0） 128: 本机目前为冗余异常状态（对应主从机状态中的 0~6 及 11 项） 129: 主从机钥匙开关状态不一致 130: 主从机之间数据不同步
MSState	BYTE	主从机状态（默认值 0） 0: 未知态 1: 初始态 2: 硬件 Ready 态 3: 双机 Ready 态 4: 单机 Ready 态 5: 故障态 6: 错误态 7: 工程冗余 8: 工程验证 9: 从机 10: 主机 11: 不合格主机
RSState	BYTE	单双机状态（默认值 0） 0: 未知态 1: 单机 2: 双机
ABState	BYTE	AB 机状态（默认值 0） 0: 未知态 1: A 机 2: B 机

3. 指令使用举例

变量定义					
序号	变量名	变量说明	变量类型	初始值	区域
0001	SYSGETREDSTATE_1		SYSGETREDSTATE		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Varbyte1		BYTE	0	G区
0004	Varbyte2		BYTE	0	G区
0005	Varbyte3		BYTE	0	G区
0006	Varbyte4		BYTE	0	G区
编程语言		程序			
梯形图（LD）					
功能块图（FBD）					

2.4.3.6 sysMasterSwitchToSlave（主从切换）

1. 功能

将主机切换为从机。



图 146 sysMasterSwitchToSlave 指令

2. 参数说明

输入参数	类型	含义
EN_R	BOOL	使能 FALSE: 无效（默认值） TRUE: 上升沿有效
SwitchFlag	BOOL	FALSE: 不允许切换 TRUE: 允许切换

输出参数	类型	含义
Q	BOOL	输出，默认值 FALSE TRUE: 指令执行成功 FALSE: 指令执行失败
Error	BYTE	错误，默认值 0 0: 正确 128: 单机系统 129: 主从同步尚未完成 130: 从机的钥匙开关在 PRG 状态

3. 指令使用举例

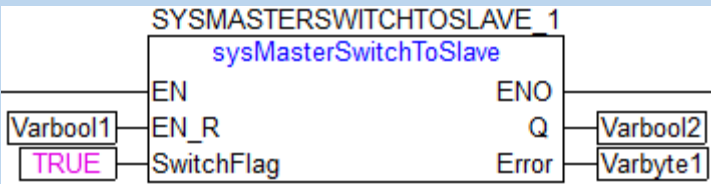
变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	SYSMASTERSWITCHTOSLAVE_1		SYSMASTERSWITCHTOSLAVE		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Varbool2		BOOL	FALSE	G区
0004	Varbyte1		BYTE	0	G区

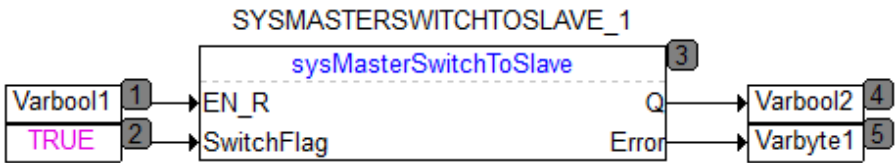
编程语言

程序

梯形图 (LD)



功能块图 (FBD)



2.4.3.7 sysGetDPSlaveState (获取 DP 从站诊断)

1. 功能

获取 DP 从站诊断信息。

2. 参数说明

输入参数	类型	含义
IN	BOOL	高电平使能，默认值 FALSE
DPAddr	BYTE	从站地址（默认值为 0，有效值为 2-125）
Slot	BYTE	DP 主卡所在槽号

输出参数	类型	含义
Q	BOOL	输出，默认值 FALSE TRUE: 指令执行成功

		FALSE: 指令执行失败
Error	BYTE	错误（默认值 0） 0: 正确 1: 槽号错误 2: 从站地址配置错误 128: 没有 dp 主卡或槽号不匹配
ConfigState	BOOL	从站是否被配置标识: TRUE: 已被配置 FALSE: 未被配置
ComState	BOOL	从站通讯状态标识: TRUE: 通讯正常 FALSE: 通讯异常
SlaveID	WORD	从站 ID 注：LKS 从站模块的 ID 范围为 0x0500~0x05FF
DiagDataLen	BYTE	从站诊断信息长度
DiagData1~ DiagData54	BYTE	诊断信息（具体含义根据从站类型而变） 请参见《LKS 安全型控制系统产品手册》 各 IO 模块的诊断说明 DiagData51: 从站 AB 网收发器故障状态 bit0: A 网的收发器故障状态 bit1: B 网的收发器故障状态

2.4.3.8 sysGetCPUDiagInfo（获取主控模块诊断）

1. 功能

获取主控模块诊断信息。

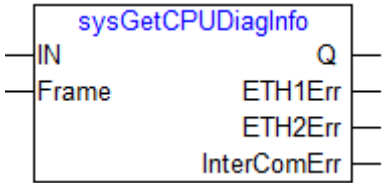


图 147 sysGetCPUDiagInfo 指令

2. 参数说明

输入参数	类型	含义
IN	BOOL	高电平使能，默认值 FALSE
Frame	BOOL	机架选择（默认值 FALSE） FALSE: 本机架（主） TRUE: 对方机架（从）

输出参数	类型	含义
Q	BOOL	输出（默认值 FALSE） TRUE: 指令执行成功 FALSE: 指令执行失败

ETH1Err	BOOL	第 1 路以太网故障信息，当断线时此值为 TRUE（默认值 FALSE）
ETH2Err	BOOL	第 2 路以太网故障信息，当断线时此值为 TRUE（默认值 FALSE）
InterComErr	BYTE	内部通信链路故障信息，当不同槽位出现故障时，此值对应 bit 为 1（默认值 0）： bit0=1：槽 2 内部通信链路故障 bit1=1：槽 3 内部通信链路故障 bit2=1：槽 4 内部通信链路故障（6 槽背板预留） bit3=1：槽 5 内部通信链路故障（6 槽背板预留） bit4~bit7：保留

3. 指令使用举例

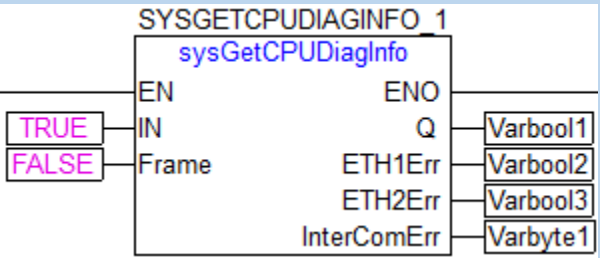
变量定义

序号	变量名	变量说明	变量类型	初始值	区域
0001	SYSGETCPUDIAGINFO_1		SYSGETCPUDIAGINFO		G区
0002	Varbool1		BOOL	FALSE	G区
0003	Varbool2		BOOL	FALSE	G区
0004	Varbool3		BOOL	FALSE	G区
0005	Varbyte4		BYTE	0	G区

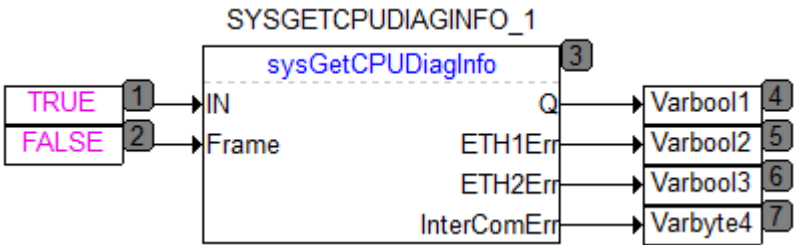
编程语言

程序

梯形图（LD）



功能块图（FBD）



程序说明：

示例为获取主机架中 CPU 模块的诊断信息。

2.4.3.9 sysGetComModuleDiagInfo（获取通讯模块诊断）

1. 功能

获取通讯模块诊断信息。

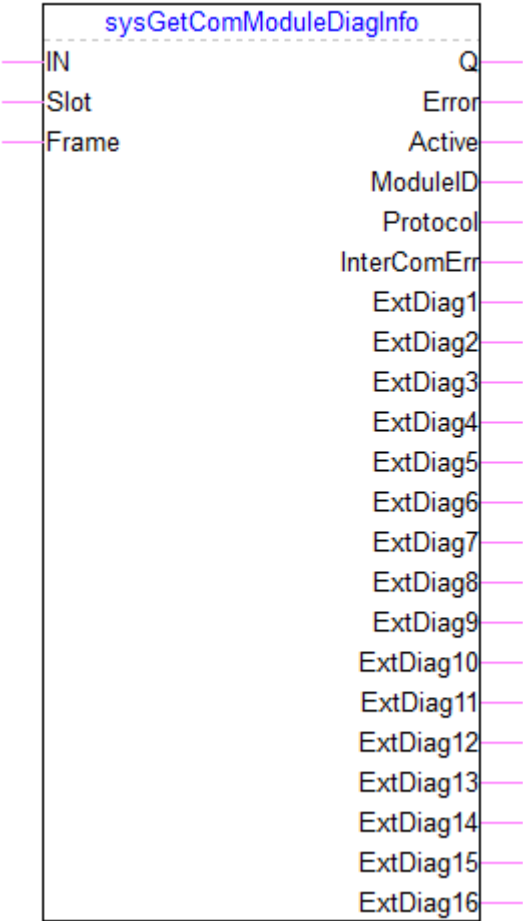


图 148 sysGetComModuleDiagInfo 指令

2. 参数说明

输入参数	类型	含义
IN	BOOL	高电平使能（默认值 FALSE）
Slot	BYTE	槽位号（默认值 2，有效值为 2~5） 主控模块槽位号固定为 1 其它模块依据与主控位置向右递增 使用时输入想得到版本信息模块所在的槽位号
Frame	BOOL	机架选择（默认值 FALSE） FALSE：本机架（主） TRUE：对方机架（从）

输出参数	类型	含义
Q	BOOL	输出（默认值 FALSE） TRUE：指令执行成功 FALSE：指令执行失败
Error	BYTE	错误（默认值 0）

		0: 正确 1: Slot 配置错误 128: 本机架槽位上不存在模块 129: 对方机架槽位上不存在模块
Active	BOOL	通信模块工作状态（默认值 0） FALSE: 离线 TRUE: 在线
ModuleID	BYTE	模块 ID 号 1: LK220(保留) 2: LK240 3: LK249 4: LK247 5: LK248 6: LK220S 安全模块(保留) 7: LK240S 安全模块 8: LK249S 安全模块
Protocol	BYTE	通信协议标识号 1: 内部冗余协议 2: DP 协议 3: CANOpen 协议 4: EtherCAT 协议 5: PowerLink 协议 6: ModbusRTU 协议 7: 自由口协议
InterComErr	BOOL	内部通信链路故障信息, 当出现故障时此值为 TRUE (默认值 0)
ExtDiag1	BYTE	扩展诊断信息 LK240S: 第 1 个字节代表第 1 路光纤断线故障 (对应值为 1) LK249S: 第 1 个字节代表 DP1 断线故障 (对应值为 1)
ExtDiag2	BYTE	扩展诊断信息 LK240S: 第 2 个字节代表第 2 路光纤断线故障 (对应值为 1) LK249S: 第 2 个字节代表 DP2 断线故障 (对应值为 1)
ExtDiag3	BYTE	扩展诊断信息 LK240S: 第 3 个字节代表 A/B 系开关冲突故障 (对应值为 1) LK249S: 第 3 个字节保留
ExtDiag4	BYTE	扩展诊断信息 LK240S: 第 4 个字节代表清看门狗复位结果标志 (对应值为 1) LK249S: 第 4 个字节代表清看门狗复位结果标志 (对应值为 1)
ExtDiag5	BYTE	扩展诊断信息 LK240S: 第 5 个字节保留 LK249S:

		第 5 个字节保留
ExtDiag6	BYTE	扩展诊断信息 LK240S: 第 6 个字节代表第 1 路光纤的误码率数值 (0-100 之间) LK249S: 第 6 个字节保留
ExtDiag7	BYTE	扩展诊断信息 LK240S: 第 7 个字节代表第 2 路光纤的误码率数值 (0-100 之间) LK249S: 第 7 个字节保留
ExtDiag8~16	BYTE	保留

2.4.3.10 sysSetProfiOareq（安全从站 Profisafe 确认）

1. 功能

安全从站 Profisafe 确认。

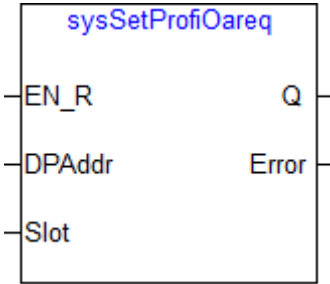


图 149 sysSetProfiOareq 指令

2. 参数说明

输入参数	类型	含义
EN_R	BOOL	上升沿使能（默认值 FALSE）
DPAddr	BYTE	从站地址（默认值为 2，有效值为 2-125）
Slot	Byte	DP 主卡所在槽号

输出参数	类型	含义
Q	BOOL	输出，默认值 FALSE TRUE：指令执行成功 FALSE：指令执行失败
Error	BYTE	错误（默认值 0） 0：正确 1：槽号错误 2：安全从站不需要确认 3：安全从站 Profisafe 确认失败



和利时集团

HollySys Group

地址：北京经济技术开发区地盛中路2号院

邮编：100176

电话：010-58981000

传真：010-58981100

<http://www.hollysys.com>