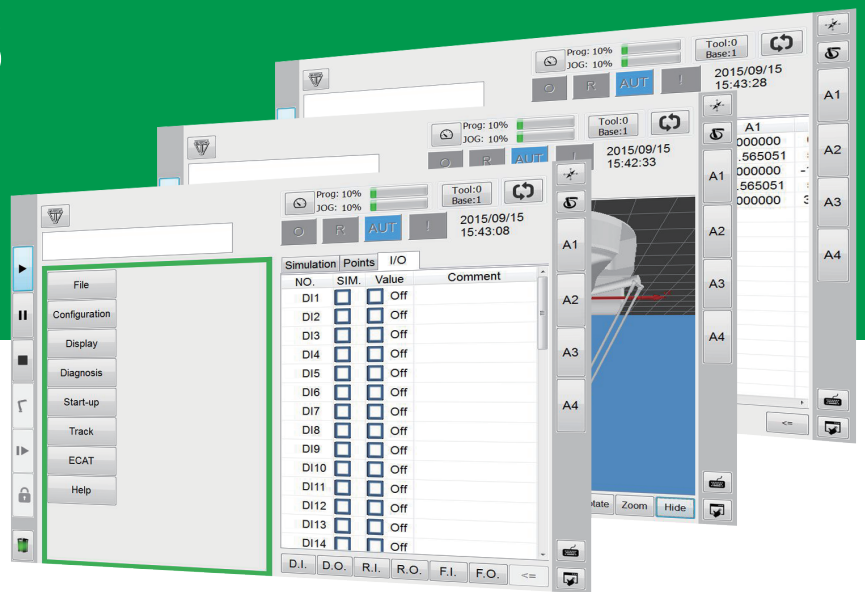




Robot System Software -HRSS 3.3

User Manual

Original Instruction



HIWIN Support



About HIWIN



Semiconductor Subsystem

Semiconductor / LED / Panel

- EFEM (Equipment Front End Module)
- Wafer Robot
- Load Port
- Wafer Aligner



Multi-Axis Robot

Pick-and-Place / Assembly / Array and Packaging / Semiconductor / Electro-Optical Industry / Automotive Industry / Food Industry

- Articulated Robot
- SCARA Robot
- Electric Gripper
- Integrated Electric Gripper



Single-Axis Robot

Precision / Semiconductor / Medical / FPD

- KK, SK
- KS, KA
- KU, KE, KC
- KA B-TYPE
- KC B-TYPE



Torque Motor Rotary Table

Medical / Automotive Industry / Machine Tools / Machinery Industry

- RAB Series
- RAS Series
- RCV Series
- RCH Series

Wire-cut EDM / Die-sinker EDM / Small-hole Drilling EDM

- RAS-E Series
- RCV-E Series
- RCH-E Series



Ball screw

Precision Ground / Rolled

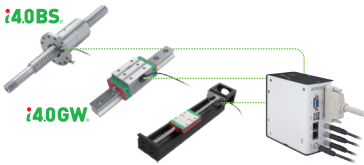
- Super S Series
- Super T Series
- Mini Roller
- Ecological & Economical Lubrication Module E2
- Auxiliary Lubrication Module EL
- Rotating Nut (R1, R2)
- Energy-Saving & Thermal-Controlling (Cool Type)
- Heavy Load Series (RD)



Linear Guideway

Automation / Semiconductor / Medical

- Ball Type-HG, EG, WE, MG, CG
- Quiet Type-QH, QE, QW, QR
- Other-Stainless Steel, AG, RG, E2, EL, PG, SE, RC



i-Series

Semiconductor / Automation Equipment / Industrial Machines / Machine Tools

- Intelligent 4.0 Ball screw
- Intelligent 4.0 Guideway
- Intelligent Single-Axis Robot



DATORKER® Strain Wave Gear

Robot / Automation / Semiconductor / Machine Tools

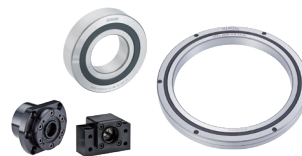
- Standard-DSC, DSH Type
- Heavy Load-DGC, DGH Type
- Lightweight-DLC Type



Ball Spline

Robot / Medical / Automation Equipment / Industrial Machines / Machine Tools / Semiconductor

- Linear ball spline-RS Type, FS Type, FSR Type
- Compound ball spline-FBR type



Bearing

Machine Tools / Robot / Industrial Machines / Automation

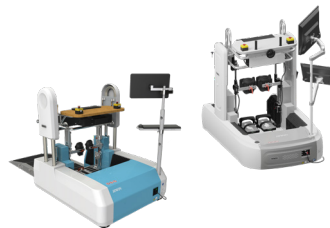
- Crossed Roller Bearing
- Ball screw Bearing
- Support Unit



Controller / Drive/ AC Servo Motor

Semiconductor / SMT / 3C Electronics / Automation Equipment / New Energy Equipment / Industrial Machinery

- Controller-HIMC Series
- Drives-E1, E2, D1, D2T/D2T-LM Series
- Motors-E, FR Series



Medical Equipment

Hospital / Rehabilitation Centers / Long-term Care Facility

- Robotic Gait Training System MRG-P110
- Robotic Gait Training System MRG-P100



Linear Motor / Linear Motor System

Automated Transport / AOI Application / Precision Positioning / Semiconductor Application

- Ironcore Linear Motor
- Ironless Linear Motor
- Tubular Motors
- Air Bearing Platform
- XY Stage • Gantry Systems
- Single-Axis Linear Motor Stage



Torque Motor / Direct Drive Motor

Machine Tools / Lithium-ion Battery / Gear Machining and Inspection

- Torque Motor-TM-2 / IM-2, TMRW, TM-2[J0] Series

Display / Automation / Semiconductor / Lithium-ion Battery / Robot / Laser Cutting / AOI Inspection

- Direct Drive Motor-DMS, DMY, DMN, DMT, DMH Series

Content

<u>1. Product Description.....</u>	<u>29</u>
1.1. Software Overview.....	29
1.2. Robot Overview	30
1.3. Hiwin Robot System Software (HRSS) Overview.....	31
1.4. Manual Description of HIWIN technologies corp.....	32
<u>2. Operation.....</u>	<u>34</u>
2.1. Teach Pendant.....	34
2.1.1. Front view	34
2.1.2. Rear View	35
2.2. HRSS Interface.....	36
2.2.1. Status Bar	38
2.2.2. Status Display “Interpreter”	38
2.2.3. Keyboard	38
2.3. Connect to Control System, and Start HRSS	38
2.4. Open Main Menu	39
2.5. Language Settings	39
2.6. Time and NTP Settings.....	40
2.7. Change User Group	41
2.8. Change Run.....	44
2.9. Change Password	45
2.10.Reset Password	46
2.11.Coordinate System	48
2.12.JOG.....	50
2.13.Manual Move.....	50
2.13.1. Jog Velocity Change	50
2.13.2. BASE/TOOL Coordinate.....	51
2.13.3. Teach Pendant Position Configuration.....	53

2.13.4. Jogging velocity Ratio	54
2.13.5. Axis-specific jogging with the jog keys	54
2.13.6. Cartesian jogging with the jog keys	55
2.13.7. Teach Pendant Alignment	55
2.13.8. Move with Space Operation Key	56
2.14.Display	57
2.14.1. Display Actual Position.....	57
2.14.2. Display Digital Input/Output	58
2.14.3. Display Remote Input/Output.....	60
2.14.4. Display External Functional Input/Output	61
2.14.5. Display System Status Input/Output	61
2.14.6. Display Robot Simulation Screen.....	63
2.14.7. Display Counter Variable	64
2.14.8. Display Timer Variable	65
2.14.9. Display Point List.....	66
2.14.10. Display Mileage	68
2.14.11. Display Utilization Rate	69
2.14.12. Display Load Percentage	69
2.14.13. Display Alarm and Zero Position History Message.....	71
2.15.Field Bus Setting (Optional)	73
2.15.1. Set CC-Link Connection Parameters	73
2.15.2. Set Profinet Connection Parameters	74
2.15.3. Set ModbusTCP Client Connection Parameters	75
2.15.4. Set ModbusTCP Server Connection Parameters	77
2.15.5. Set ModbusRTU Client Connection Parameters.....	78
2.15.6. Set ModbusRTU Server Connection Parameters.....	79
2.15.7. Sets EtherNet/IP Adapter Connection Parameter.....	80
2.15.8. Confirmation of Connection Status	81
2.15.9. Use FieldBus Input (SI[n]).....	82
2.15.10. Use FieldBus Output (SO[n])	83

2.15.11. Use FieldBus Register (SRR 、SRW)	84
2.15.12. Use FieldBus Register Mapping	85
2.15.13. Modbus Monitor Page	90
2.15.14. Fieldbus Firmware Setting	90
2.16.Communication.....	92
2.16.1. TCP/IP Communication	92
2.16.2. Setting connection for IP address.....	93
2.16.3. RS232 Communication.....	94
2.17.Electric Gripper Setting.....	96
2.18.Module I/O Function	100
2.19.Pick/Place DI Detection Function	101
2.20.Remote Monitoring Function	102
2.21.Conveyor Belt Tracking Sensor Trigger Object Identification Function	107
2.22.External Axis	108
2.22.1. External axis parameter settings	108
2.22.2. External axis zero point correction	111
2.22.3. External Axis Positive and Negative Limit Signal Monitoring Interface Setting	111
2.22.4. Manual Operations of External Axis	113
2.22.5. Display information about external axis position	114
2.22.6. Point list information	115
2.23.Welding function (optional)	116
2.23.1. Welding system and device settings.....	116
2.23.2. Welding procedure setting.....	118
2.23.3. Welding I/O	120
2.23.4. Weaving function	124
2.23.5. Torch(TCP) recovery.....	127
2.23.6. Resume Program	130
2.24.Collision Detection	130
2.24.1. Parameter Setting.....	131
2.24.2. Description on the Usage Process of Collision Detection.....	133

2.24.3. Program Command.....	135
2.25.Payload.....	136
2.25.1. Parameter Setting.....	137
2.25.2. Description on the Usage Process of Load Setting	140
2.25.3. Program Command.....	141
2.25.4. Payload calculation	141
2.26.Soft Limit	146
2.27.Software Update	149
2.28.Export Manual	150
2.29.Controller Boot/Shutdown Program Description	151
2.29.1. Standard Power ON Procedure	151
2.29.2. Standard Power OFF Procedure	152
2.29.3. Standard Power On-OFF Sequence	155
2.30.Reference Point Function.....	156
<u>3. Initial Settings</u>	<u>156</u>
3.1. Check Parameters.....	157
3.2. Calibration Flow	158
3.3. Adjust Origin Position of Hardware Mechanism.....	159
3.3.1. Instructions for Restoring Default Factory Origin Positions.....	163
3.3.2. Reset Origin Position Instructions	166
3.3.3. Reset Encoder Error.....	168
3.4. Calibrate Coordinates (Six-axis Robot).....	169
3.4.1. Base calibration	169
3.4.2. Tool calibration	171
3.5. Calibrate Coordinates (Delta Robot).....	173
3.5.1. Calibrate Base Coordinates	173
3.5.2. Calibrate Tool Coordinates	175
3.6. Conveyor Calibration	178
3.6.1. Conveyor Image Calibration.....	178

3.6.2.	Conveyor Arm Calibration	179
3.7.	Conveyor Parameters	181
3.7.1.	Image Parameters.....	181
3.7.2.	Object Parameters.....	181
3.8.	Home and Position Check Configuration.....	184
3.8.1.	Home Configuration	184
3.8.2.	Configuration of Position Check Alarm.....	185
3.8.3.	Position Check	187
3.9.	Infinite Rotation Function	188
3.9.1.	Axis 6 Infinite Rotation Function	188
4.	<u>Program Management.....</u>	190
4.1.	Document Navigator	190
4.2.	Add Program.....	192
4.3.	Copy Program	192
4.4.	Open Program.....	193
4.5.	HRSS STRUCTURE	195
4.5.1.	Home Position	195
4.6.	Start Program.....	196
4.6.1.	Select Program Running.....	196
4.6.2.	Program Running.....	196
4.6.3.	Pre-reading.....	196
4.6.4.	Set Program Ratio.....	197
4.6.5.	Driver status	197
4.6.6.	Decoder Status Display	197
4.6.7.	Start a Program (Manual)	197
4.6.8.	Start a Program (Auto).....	198
4.6.9.	Start External Auto Run	198
4.7.	Edit Program	198
4.7.1.	Copy Program Bar.....	198

4.7.2.	Paste Program Bar	199
4.7.3.	Cut Program Bar	199
4.7.4.	Delete Program Line	200
4.7.5.	Comment Program Bar	200
4.7.6.	Cancel Comment Program Bar	200
4.7.7.	Indent Program Bar	201
4.7.8.	Cancel Indent Program Bar	201
4.7.9.	Recovery Program Bar	201
4.7.10.	Cancel Recovery Program Bar	201
4.8.	Backup and Recovery Data	202
4.8.1.	Backup Data	202
4.8.2.	Recovery Data	202
4.8.3.	Backing Up Database	203
4.8.4.	Restoring Database	204
4.9.	Rules for naming files	205
5.	<u>Motion Program Design.....</u>	<u>206</u>
5.1.	Motion Overview	206
5.2.	Point-to-point (PTP) Motion.....	206
5.3.	LIN Motion.....	206
5.4.	CIRC Motion.....	207
5.5.	Blend	207
5.6.	Singular Point.....	208
6.	<u>Programming with inline forms.....</u>	<u>210</u>
6.1.	Name in Inline Form	210
6.2.	Programming for PTP, LIN, CIRC, SPINE Motion	210
6.2.1.	Programming for PTP Motion	210
6.2.2.	Programming for LIN Motion	212
6.2.3.	Programming for CIRC Motion	215

6.2.4.	Perform Programming for SPLINE Motion	217
6.3.	Variable Configuration	218
6.3.1.	REAL	218
6.3.2.	INT (programmed by keyboard)	218
6.3.3.	UINT (programmed by keyboard).....	219
6.3.4.	BOOL (programmed by keyboard).....	219
6.3.5.	CHAR (programmed by keyboard).....	219
6.3.6.	STRING(programmed by keyboard).....	220
6.3.7.	E6POS Coordinate Points (programmed by keyboard).....	220
6.3.8.	E6AXIS Axis Points (programmed by keyboard)	220
6.3.9.	E6POINT Coordinate Points (programmed by keyboard)	221
6.3.10.	FRAME Coordinate Points (programmed by keyboard).....	221
6.4.	Register Configuration	221
6.4.1.	Using COUNTER Register	221
6.4.2.	Enable TIMER Register	222
6.5.	Programming for Logic Command	223
6.5.1.	Input/Output.....	223
6.5.2.	Digital OUTPUT	223
6.5.3.	WAIT FOR (SEC)	224
6.5.4.	WAIT FOR (INPUT)	224
6.5.5.	QUIT (programmed by keyboard)	225
6.6.	Perform Program Design for the Judgment Loop Instruction	225
6.6.1.	IF	225
6.6.2.	FOR (programmed by keyboard)	226
6.6.3.	LOOP (programmed by keyboard).....	226
6.6.4.	WHILE.....	227
6.6.5.	REPEAT (programmed by keyboard).....	227
6.6.6.	GOTO (programmed by keyboard)	228
6.6.7.	SWITCH (programmed by keyboard)	228
6.7.	Simulate Environment Instruction	229

6.7.1.	ADDTOOL Newly Add Tool (programmed by keyboard)	229
6.7.2.	SHOW_TOOL Show Tool (programmed by keyboard)	229
6.7.3.	ADDOBJ Newly Add Workpiece (programmed by keyboard)	230
6.7.4.	SHOW_OBJ Show Workpiece (programmed by keyboard)	230
6.7.5.	MOVEFLOOR Position of Moving Floor (programmed by keyboard)	231
6.7.6.	AXISON Display Coordinate System.....	231
6.7.7.	AXISOFF Hidden Coordinate System.....	232
6.8.	Definition of Structure (programmed by keyboard).....	232
6.9.	Subprogram (programmed by keyboard)	232
6.10.	Communication Configuration	233
6.10.1.	Using RS232 to Connect with External Equipment	233
6.10.2.	Use Ethernet to Connect with External Equipment	236
6.11.	Conveyor Command	239
6.11.1.	CNV_START.....	239
6.11.2.	CNV_END	239
6.11.3.	CNV_PICK_QUANTITY	240
6.11.4.	CNV_TRIGGER_TIMES[NUM]	240
6.11.5.	CNV_PICK	240
6.11.6.	CNV_PLACE	241
6.11.7.	CNV_OBJECT.....	243
6.11.8.	CNV_FULL.....	243
6.11.9.	CNV_EMPTY	243
6.11.10.	CNV_SET_DELAY_TIME[NUM].....	244
6.11.11.	CNV_QUEUE_SIZE[NUM]	245
6.11.12.	CNV_OBJ_CNT_DIST[NUM].....	246
6.11.13.	CNV_PLACE_BATCH[NUM].....	246
6.11.14.	CNV_RESET_ENC	246
6.11.15.	CNV_QUEUE_REMOVE[NUM].....	247
6.11.16.	CNV_PICK_ACC[NUM]	247
6.11.17.	CNV_OFFSET_X[NUM]	247

6.11.18. CNV_OFFSET_Y[NUM]	248
6.11.19. CNV_OFFSET_Z[NUM]	248
6.11.20. CNV_SPEED[NUM].....	248
6.12.DO switching on the path (SYN)	249
6.13.Setting of External Procedure Input/Output	250
6.13.1. Mode Setting of External Procedure	250
6.13.2. External Procedure Function of RSR Mode.....	250
6.13.3. External Procedure Function of PNS Mode	251
6.14.Positioning Check of Arm Position.....	253
6.15.Self-defined Digital Input Control Function	255
6.15.1. Clear Error	255
6.15.2. External Alarm.....	256
6.15.3. External Shutdown Input	256
6.16.Self-defined Digital Output Control Function	256
6.16.1. Motor Warning.....	257
6.16.2. System Start Up.....	257
6.16.3. Teach Pendant Manual/Auto Mode Selection Output Signal Setting	257
6.16.4. Reset Driver.....	257
6.16.5. Emergency Output.....	258
6.16.6. Reset Safety Relay.....	258
6.17.Setting of Motion Parameters (programmed by keyboard)	258
6.17.1. SET_OVERRIDE_SPEED.....	259
6.17.2. SET_SPEED.....	259
6.17.3. SET_ACC	260
6.17.4. SET_ROTATION_SPEED	260
6.17.5. SET_TOOL.....	261
6.17.6. SET_BASE	261
6.17.7. GETPOINT.....	262
6.17.8. GET_MOTION_STATUS.....	262
6.17.9. BRAKE.....	262

6.17.10. EXT_TCP (Optional).....	263
6.17.11. CHECK_LIN.....	263
6.17.12. FWD - Forward Robot Kinematics Convert Command	264
6.17.13. STOP_MOTION - Trigger to Stop Previous Motion According to Signal	264
6.18.Location Register	265
6.18.1. Using Interface to Enter Location Register	265
6.18.2. Using Instruction to Enter Location Register	266
6.18.3. Use Location Register for Motion.....	266
6.19.User Self-defined Alarm.....	267
6.19.1. Content of Defined Alarm	267
6.19.2. Issue Self-defined Alarm (programmed by keyboard).....	267
6.20.Electric Gripper Command	268
6.20.1. Command Description	268
6.21.Infinite Rotation Command (Optional)	272
6.21.1. CT_A6.....	272
6.22.External Axis Command	272
6.22.1. EX_AX.....	272
6.22.2. EX_AX_ASYNC.....	272
6.22.3. EX_AX_SET_ACC	273
6.22.4. EX_AX_SET_SPEED.....	273
6.22.5. EX_AX_CT	273
6.22.6. ASYPTP	274
6.22.7. PTP	274
6.22.8. LIN.....	275
6.22.9. CIRC.....	275
6.22.10. E1 Motor Selection Specifications	276
6.23.Welding Command.....	278
6.23.1. WELD_START	278
6.23.2. WELD_END	278
6.23.3. WEAVE	278

6.23.4. WEAVE_END	279
6.24. Interrupt instructions	279
6.24.1. RESUME_PROG.....	279
6.24.2. CLEAR_RESUME_PROG.....	280
6.25. Collision Detection command	280
6.25.1. CD_START	280
6.25.2. CD_END.....	280
6.26. Payload setting command.....	281
6.26.1. SET_PAYLOAD	281
6.27. String Function	281
6.27.1. String Cascade	281
6.27.2. STRCMP	282
6.27.3. STRPOS.....	282
6.27.4. STRCPY	283
6.27.5. STRLEN	283
6.27.6. INTTOSTR	283
6.27.7. REALTOSTR	284
6.27.8. STRTOINT.....	284
6.27.9. STRTOREAL.....	285
6.28. Interrupt	286
6.28.1. Declaration	286
6.28.2. Enable and Disable	286
6.28.3. Functional limitations and specifications	286
6.29. Multiple(1~3) Dimension Array	287
6.29.1. Declare	287
6.29.2. Assign And Access Elements of an Array	288
6.29.3. Pass Array as Function Parameter	288
6.30. PULSE Command	289
6.30.1. PULSE(time).....	289
6.31. Timeout command	290

6.31.1. TIMEOUT(time, label)	290
6.32.P# Command	290
6.32.1. P#(index)	290
6.32.2. P#_EXIST(index).....	291
6.33.Torque Limit Monitoring	291
6.33.1. TL_SET - Set Torque Limit Monitoring Parameter Commands	291
6.33.2. TL_START -Enable Torque Limit Monitoring.....	292
6.33.3. TL_END -End Torque Limit Monitoring	293
<u>7. Error Message</u>	<u>294</u>
7.1. Robot System Software(01-XX-XX)	294
7.1.1. System Error Message(01-01-XX)	294
7.1.2. Program Error(01-02-XX)	300
7.1.3. Motion Error(01-03-XX)	301
7.1.4. Operation Error(01-04-XX)	310
7.1.5. IO & Communication(01-05-XX).....	314
7.1.6. Operator Error (01-06-XX).....	317
7.1.7. External Axis Error (01-07-XX).....	319
7.1.8. Conveyor Tracking Error(01-08-XX)	320
7.1.9. User-Defined Error (01-09-XX)	322
7.1.10. Welding Function Error (01-0C-XX)	323
7.1.11. Database Error (01-0D-XX)	325
7.1.12. Reserved Point Error (01-0E-XX)	326
7.1.13. Windows Information	326
7.2. HIWIN Robot Controller(02-XX-XX).....	328
7.2.1. Safety Input(02-01-XX).....	328
7.2.2. Hardware Error(02-02-XX)	328
7.3. Axis Amplifier(03-XX-XX)	328
7.3.1. Function Name and Number Description	329
7.3.2. Driver Alert Number	329

7.3.3.	DAC – Y Driver Alarm Code(Y-XXX)	338
7.3.4.	DAC – S Driver Alarm Code(S-XXXX)	349
7.3.5.	DAC – H Driver Alarm Code(H -XXX)	353
7.4.	Electric gripper(04-XX-XX).....	355
7.4.1.	Hardware Error (04-01-XX)	355
7.4.2.	Operation Error (04-02-XX, 04-01-8X)	357
7.4.3.	Electric Gripper Controller Alarm Signal Error (04-01-3X).....	358
7.4.4.	Electric Gripper Command Communication Timeout (04-01-4X).....	359
7.5.	SDK Server (SDK server,05-XX-XX)	360
7.5.1.	Server Error (05-01-XX).....	360
8.	<u>Program Examples.....</u>	<u>361</u>
8.1.	Register	361
8.1.1.	COUNTER Register	361
8.1.2.	TIMER Register	361
8.2.	Variable Type	362
8.2.1.	REAL	362
8.2.2.	INT.....	362
8.2.3.	UINT	362
8.2.4.	BOOL	362
8.2.5.	CHAR	363
8.2.6.	STRING	363
8.2.7.	E6POS Point.....	363
8.2.8.	E6AXIS Point	364
8.2.9.	E6POINT Point	364
8.3.	Operator	365
8.3.1.	Arithmetic Operator	365
8.3.2.	Logic Operator.....	365
8.3.3.	Relation Operator	365
8.4.	Input/Output	366

8.4.1.	Digital Input.....	366
8.4.2.	Digital Output.....	366
8.4.3.	Robot Input.....	366
8.4.4.	Robot Output.....	366
8.4.5.	Valve Output.....	366
8.5.	Motion Function	367
8.5.1.	PTP.....	367
8.5.2.	PTP_REL.....	368
8.5.3.	LIN.....	369
8.5.4.	LIN_REL.....	370
8.5.5.	LIN_REL_TOOL.....	370
8.5.6.	CIRC.....	371
8.5.7.	CIRC_REL.....	372
8.5.8.	SPLINE.....	373
8.5.9.	Array Accumulation.....	373
8.5.10.	CT_A6.....	374
8.5.11.	BRAKE.....	374
8.5.12.	EXT_TCP.....	375
8.5.13.	CHECK_LIN.....	376
8.5.14.	FWD.....	376
8.5.15.	STOP_MOTION.....	377
8.6.	Control Function	377
8.6.1.	IF.....	377
8.6.2.	FOR.....	379
8.6.3.	LOOP.....	381
8.6.4.	WHILE.....	382
8.6.5.	REPEAT.....	383
8.6.6.	GOTO.....	384
8.6.7.	SWITCH.....	386
8.6.8.	WAIT.....	389

8.6.9. QUIT	390
8.7. Motion Parameter	390
8.7.1. CONT	390
8.7.2. FINE	393
8.7.3. VEL	393
8.7.4. ACC.....	393
8.8. Definition of Structure.....	394
8.9. Function & Subprogram.....	395
8.9.1. Definition & Using Method of Function	395
8.9.2. Definition & Using Method of Subprogram	396
8.10.External Function & Subprogram	398
8.10.1. Definition & Using Method of External Function(EXTFCT)	398
8.10.2. Definition &Using Method of External Subprogram(EXT)	399
8.11.RS232 Configuration	400
8.12.NET Configuration	401
8.13.Conveyor Configuration	402
8.13.1. Pick Program(1)	402
8.13.2. Pick Program(2)	404
8.13.3. Pick Program(3)	405
8.14.DO switching on the path(SYN OUT).....	406
8.14.1. Program Example 1 of SYN.....	406
8.14.2. Program Example 2 of SYN.....	406
8.14.3. Program Example 3 of SYN.....	408
8.14.4. Program Example 4 of SYN.....	409
8.14.5. Program Example 5 of SYN.....	410
8.15.Electric Gripper	411
8.16.External Axis	414
8.17.Welding(Optional).....	414
8.18.Collision Detection	415
8.19.Collision Detection Learning Example.....	415

8.20.Payload	416
8.21.Interrupt	416
8.22.Array	419
8.23.PULSE	419
8.24.TIMEOUT	420
8.25.P#	420
<u>9. Appendix</u>	<u>421</u>
9.1. Software commands	421
9.1.1. Motion commands	421
9.1.2. RS232 or EtherNet Commands.....	422
9.1.3. Conveyor tracking commands.....	424
9.1.4. Register commands	426
9.1.5. Variable types.....	427
9.1.6. Math Calculation	428
9.1.7. Control function	429
9.1.8. Gripper commands.....	431
9.1.9. External Axis Instruction	431
9.1.10. Welding Instruction	432
9.1.11. Collision Detection Instrcutio	433
9.1.12. Payload Setting Instrcutio	433
9.1.13. String Function Instructions.....	433
9.1.14. Interruption Process Instrcutio	433
9.1.15. Other commands.....	434

Warranty Terms and Conditions

The period of warranty shall commence at the received date of HIWIN product (hereafter called “product”) and shall cover a period of 12 months. The warranty does not cover any of the damage and failure resulting from:

1. The damage caused by using with the production line or the peripheral equipment not constructed by HIWIN.
2. Operating method, environment and storage specifications not specifically recommended in the product manual.
3. The damage caused by changing installation place, changing working environment, or improper transfer after being installed by the professional installer.
4. Product or peripheral equipment damaged due to collision or accident caused by improper operation or installation by the unauthorized staff.
5. Installing non-genuine HIWIN products.

The following conditions are not covered by the warranty:

1. Product serial number or date of manufacture (month and year) cannot be verified.
2. Using non-genuine HIWIN products.
3. Adding or removing any components into/out the product without authorized.
4. Any modification of the wiring and the cable of the product.
5. Any modification of the appearance of the product; removal of the components inside the product. e.g., remove the outer cover, product drilling or cutting.
6. Damage caused by any natural disaster. i.e., fire, earthquake, tsunami, lightning, windstorms and floods, tornado, typhoon, hurricane etc.

HIWIN does not provide any warranty or compensation to all the damage caused by above-mentioned circumstances unless the user can prove that the product is defective.

For more information towards warranty terms and conditions, please contact the technical staff or the dealer who you purchased with.

WARNING

1. Improper modification or disassemble the robot might reduce the robot function, stability or lifespan.
2. The end-effector or the cable for devices should be installed and designed by a professional staff to avoid damaging the robot and robot malfunction.
3. Please contact the technical staff for special modification coming from production line set up.
4. For the safety reason, any modification for HIWIN product is strictly prohibited.

Safety Precautions

1. Safety Information

- Safety Responsibility and Effect
 1. This chapter explains how to use the robot safely. Be sure to read this chapter carefully before using the robot.
 2. The user of the HIWIN industrial robot has responsibility to design and install the safety device meeting the industrial safety regulations in order to ensure personal safety.
 3. In compliance with the safety information on industrial robot described in this manual can't guarantee that *HIWIN* robot will not occur any safety problems.
 4. This machine is defined as a partly completed machinery, the associated hazards must be handled by system integrator in accordance with ISO 10218-1/ ISO 10218-2.
 5. A safety-related part of control system (SRP/CS) should conform to the requirement of performance level d and category 3 according to ISO 13849-1.
 6. The installation for emergency functions shall be defined by the system integrator in accordance with ISO 10218-1/ ISO 10218-2.

- Safety Operation Principle
 1. Before connecting the power supply for HIWIN industrial robot startup assembly procedure, check whether the specification of factory output voltage matches the specification of input voltage of the product. If it does not match, ensure to use the corresponding transformer (HIWIN optional transformer is recommended).
 2. Emergency Stop button (on Teach Pendant or from external emergency stop switch) must be pressed before turning off the power, and then switch off the power switch.
 3. While connecting to the external I/O or the signal, please operate in the condition that the power switch is turned off to prevent from a shortcut caused by mistaken touch in the process, and resulting in damage.

Safety Precautions

i. General

All personnel involved in the use or setup of the industrial robot arm must read the safety related literature for the robot arm and instruction manual in detail and operate it in accordance with the specifications.

Safety Symbol



Users must strictly abide by the content description, otherwise it will cause serious casualties.



Users must strictly abide by the content instructions, otherwise it may cause minor injuries or equipment damage.



User must strictly abide by the content description, otherwise it may cause poor product performance.

Use Limit

Robotic arm is prohibited for use in the following environments and uses

- Personnel carrying purposes
- Explosive environment
- Environment without safety precautions
- Outdoor environment
- Environment affected by oil, water, dust, etc.

ii. Relevant Personnel

Electrical or mechanical work on industrial robot arms is only permitted by professionals.

WARNING

All personnel working on industrial robotic arms must read and understand the manual containing the safety section of the system of the robotic arm.

System Integrator

Refers to the person who integrates the industrial robot arm into a set of equipment according to safety regulations and puts it into operation.

The system integrator is responsible for the following tasks:

- Install industrial robot arm.
- Industrial machinery arm related equipment connection work.
- Risk assessment of the overall system.
- Use safe guard devices.
- Confirm that the components used by the safe guard devices follow regulations.
- Placement, replacement, setup, operation, maintenance and repair work is only permitted for specially trained personnel in accordance with the operating instructions for the components of the industrial robot arm.

User

Users must be professionally trained, have the knowledge and experience in this area, and be familiar with the prescribed standards, and thus be able to make a correct judgment of the work to be performed and identify potential hazards.

Users can be defined into three categories based on operational permissions:

1. Operator

- System startup and shutdown
- Power on and off
- Alarm system status recovery

2. Engineer

- Operating personnel usage authority
- Programming and changing
- Arm teaching operation

3. Expert

- Engineer usage authority

■ Mechanical arm maintenance work

System Operation

Those who do not use functional safety kits must implement safety-fence guidance.

The system operation of personnel is divided into the following three levels

1. Operator
2. Engineer
3. Expert

Its control permissions are shown in the following table.

No.	Function	Operator	Engineer	Expert
	Function Table			
1	File	X	X	O
2	Configuration>User group	O	O	O
3	Display>Input/Output	X	O	O
4	Display>Variable	X	O	O
5	Display>Mileage	O	O	O
6	Display>Utilization	O	O	O
7	Display>Motor Torque	O	O	O
8	Display>Fieldbus	X	X	O
9	Diagnosis>Logbook	O	O	O
10	Start-up>Calibrate	X	X	O
11	Start-up>Master	X	X	O
12	Start-up>Robot data	X	O	O
13	Start-up>Network Config	X	X	O
14	Start-up>RS-232	X	X	O
15	Start-up>System Setting	X	X	O
16	Start-up>Electric Gripper	X	X	O
17	Start-up>External Axis	X	X	O
18	Start-up>Arc Weld	X	X	O
19	Track>Setting	X	O	O
20	Track>Vision Setting	X	O	O
21	Track>Vision Object	X	O	O
22	Track>Calibration	X	O	O
23	Track>Monitor	O	O	O
24	Help>About	O	O	O
25	Help>Operating Time	O	O	O
26	Help>Update	X	X	O

No.	Function	Operator	Engineer	Expert
27	Help>TP Calibration	O	O	O
28	Help>Manual	O	O	O
	Interface			
29	Message box	O	O	O
30	Velocity configuration	X	O	O
31	Tool/base coordinate	X	X	O
32	Teach Pendant configuration	X	O	O
33	Change JOG coordinate system	X	O	O
34	JOG	X	O	O
35	On-screen keyboard	O	O	O
36	Remove teach pendant	X	O	O
37	Step execution	X	X	O
38	Program execution	O	O	O
39	Program selection	O	O	O
40	Modify program	X	X	O
41	Tool/base calibration	X	O	O
42	IO operation	X	O	O
43	Functional IO modification	X	X	O

WARNING

Electrical or mechanical work is only allowed to be carried out by professionals.

Operator Safety Precautions

The manner and scale of the work and the possible hazards must be explained to the relevant personnel before work, and relevant training courses must be carried out on a regular basis. In the event of an accident or technical correction, a training course must be re-run.

System Set Up Safety Precautions

The system set up only allows specially trained personnel to perform and work in accordance with the installation, setup, operation and other relevant documents provided by the original manufacturer.

Maintenance Personnel's Precautions

Maintenance should only be carried out by specially trained personnel in accordance with the instructions and operating instructions.

iii. Robotic Arm Working Range Definition

- Working area

The working area of the robot is defined as the area of motion under motion constraints, and the working area must be limited to the minimum required.

- Protective area

A protected area is an area of the working area that is protected by a safe guard device. The area must include working areas and collaboration areas, and the safety areas ensure safety in the working area.

iv. Description of Safety Functions

Industrial robotic arms must have the following safety features:

- Selection of operating mode of the robot arm
- Safe guard devices
- Emergency stop device
- Teach pendant enable switch

The safety function of the robot arm system is to prevent loss of personnel or property. If the function is not complete or in failure state, the industrial robot arm must be prohibited from operating.

Operation Mode Selection

Operating mode application and speed description

Mode	Application	Velocity
T1	Used for run test, programming and instruction	Programmed velocity, maximum 250 mm/s
T2	Used for run test	Programmed initial velocity, less than 250 mm/s
AUT	Used for robot without the higher-level controllers	No speed limit Unable to perform manual control
EXT	Used for the robot with the higher-level controllers (For example, PLC)	No speed limit Unable to perform manual control

Manual Operation Mode (T1,T2 Mode)

The manual operation mode is used for program design, program operation check or teaching, etc. When performing manual operation, pay attention to the followings:

- All actions must be operated within the protection area.
- Do not damage or potentially damage the relevant equipment due to operates the robotic arm.

- Operation must be carried out outside the protected area as much as possible.

Both manual and automatic modes of operation in the protected area are not permitted unless the arm is equipped with a certified speed monitoring accessory from the manufacture.

Automatic Mode

The automatic mode startup should include the following conditions:

- The safe guard devices have been set up and confirmed that their functions are working properly.
- All suspended security should restore its full functionality.
- Confirm that there are no people in the protected area.
- Relevant workflow rules are complied.

To enter the protection area in this mode, the emergency stop function must be activated before entering.

Safe Guard Devices Description

The safe guard device must use the components approved by the safety regulations and set and plan according to the relevant regulations.

The robotic arm system must be automatically activated to receive the safety signal. In the event of a connection failure during automatic mode operation, an emergency stop must be triggered. When reconnecting after disconnection, the device cannot be automatically started directly and must be started manually. Manual slow running (T1) and manual fast running (T2) modes allow the guard not activate. A method must be provided to confirm that no personnel are in the protected area when the automatic mode is activated.

Users must strictly abide by the content description, otherwise it will cause serious casualties.

Temporary fences can be used during system installation and can be set according to ISO 10218-2 regulations

Stop Functions

Stop Category Description

Stop Category 0: The drive immediately cuts off the power after triggered.

Stop Category 1: The drive cuts off the power after the robot stops moving.

Stop Category 2: The drive maintains the power supply after the robot stops moving.

Stop Mode of Operation Mode

Trigger	T1,T2	AUT, AUT EXT
Release the Start Button	STOP 2	N/A
Press the Stop Button	STOP 2 (stop 2)	
Disconnect the drive device	STOP 1 (stop 1)	
No "run allowance" at input	STOP 2 (stop 2)	
Disconnect the control system (power disconnection)	STOP 0 (stop 0)	

Trigger	T1,T2	AUT, AUT EXT
Internal failure in the control system not related to the safety	STOP 0 or STOP 1 (depend on the failure reason)	
Operating mode changed	STOP 2	
Open the safety door (safety device)	-	SAFETY STOP 1
Release Enabling Switch	STOP 2	-
Press Enabling Switch or failure	STOP 2	-
Trigger emergency stop	SAFETY STOP 1	

Emergency Stop Description

Emergency stop related precautions

- Confirm that the function is functioning normally every six months.
- System integrators should provide emergency stop devices to ensure that the machine is operational or that a hazardous situation exists.
- At least one external emergency stop device is installed. Make sure that additional emergency stop devices are available for use without or losing the teach pendant.
- Provide interface to connect external emergency stop devices.
- The emergency stop function can be triggered when the safety control system connected to the robot arm is cut off.
- The risk assessment should assess whether the emergency stop is not triggered when the robotic arm control system is turned off and provides a response.
- If a tool or other device connected to the robot is dangerous, it must be connected to the emergency stop circuit on the equipment side.

Teach Pendant Enabling Switch

The teach pendant is equipped with two three-stage enabling switches:

Three-stage enable switch position

- When the switch is in the first stage, it will trigger the stop state of Stop Category 2
- When the switch is in the second stage, the industrial robot arm motion command can be executed in the teach mode.
- When the switch is in the third stage (fully pressed), it is the alarm position, which will trigger the stop state of Stop Category 2

Do not use any methods or tools to affect the function of the enable switch, otherwise it may cause serious danger and property damage.

v. Warnings and Precautions

General considerations

DANGER

1. All operating procedures should be assessed by professional and in compliance with related industrial safety regulations.
2. When operating robot, operator needs to wear safety equipment, such as workwear for working environment, safety shoes and helmets.
3. When encountering danger or other emergency or abnormal situation, please press the emergency stop button immediately. After danger is eliminated, move the robot away with low speed in manual mode.
4. When considering safety of the robot, the robot and the system must be considered at the same time. Be sure to install safety fence or other safety equipment and the operator must stand outside the safety fence while operating the robot.
5. A safety zone should be established around the robot with an appropriate safety device to stop the unauthorized personnel from access.
6. While installing or removing mechanical components, be aware of a falling piece which may cause injury to operator.
7. Ensure the weight of workpiece does not exceed the rated load or allowable load moment at wrist. Exceeding these values could lead to the driver alarm or malfunction of the robot.
8. Do not climb on manipulator.
9. Do not store the machine in the environment with corrosion and flammable gas or close to the flammable object.
10. Do not operate the machine in the environment with moisture, water or grease.
11. Do not operate the machine at the place where vibration or the strong impact occurs.
12. Do not immerse the electric wires into grease or water.
13. Do not connect or operate the machine with wet hands.

14. Do not operate the machine in potentially explosive environment.
15. Please ensure the controller is grounded.
16. Keep hands away from the inner part of the controller while it is connecting to the power or during operating.
17. Do not touch the heat sink, regenerative resistance, the power supply or the computer inside the controller while it is operating due to its high temperature.
18. Be sure power is disconnected prior to repair and maintenance, and ensure to operate under the condition of no electrical shock risk.
19. Do not disassembly the controller without permission. If there's any issues, please contact our engineers.

WARNING

1. The personnel installing robot should be trained and licensed.
2. To ensure personal safety, robot installation must comply with this manual and related industrial safety regulations.
3. The control cabinet should not be placed near high voltage or machines that generate electromagnetic fields to prevent interference that could cause the robot to deviation or malfunction.
4. Using non-HIWIN spare parts to repair may cause robot damage or malfunction.
5. Beware of the heat generated by the controller and servo motor.
6. Do not overbend the cable to avoid poor circuit contact or unexpected damage.
7. Do not stand on the controller or put heavy objects on it.
8. Do not block the vent or put foreign objects into the controller.
9. Please ensure the controller is fixed on the base.
10. Do not pull the connector violently or twist the electric wires excessively.
11. Do not frequently switch ON/OFF the power switch and the control button.
12. Please ensure that the robot, the emergency stop switch and the controller are functioning properly before performing any work.
13. Do not shutdown the power switch during the operation.
14. Do not open, modify, disassemble and maintain the machine without permission.
15. The power must be disconnected when the machine does not operate in a long time.
16. Do not turn off the power of the controller when modifying the program or parameter. Otherwise, the data stored in the controller will be damaged.
17. When changing the program or parameters inside the robot controller, do not turn off the power of the controller. Otherwise, the internal data of the controller will be damaged.
18. After the brake of a servo motor is released, the robot will be moved due to gravity and it may injured the operator.
19. The industrial robots can be applied for the different industrial environments.
20. When the operating procedures are interrupted, the special attention should be paid during the troubleshooting.

Precautions during operations

DANGER

1. Teaching, jogging or programming should be done outside of the safety fence. If it is inevitable to enter the safety fence, press the emergency stop button before entrance. Operation should be restricted at low speed and beware of surrounding safety.
2. All operations shall be executed by trained staff.

Maintenance Precautions

DANGER

1. Please contact us if the procedure not specified by HIWIN is needed.
2. Please contact us if the replacement of the component not specified by HIWIN is needed.
3. Be sure to carry out regular maintenance, otherwise it will affect the service life of the robot or other unexpected danger.
4. Prior to repair and maintenance, please switch off power supply.
5. Maintenance and repair should be performed by a qualified operator with a complete understanding of the entire system to avoid risk of robot damage and personal injury.
6. When replacing the components, avoid foreign object going into the robot.

Precautions for using End Effector

End effectors can be basically divided into the following two categories:

- A. Gripper: Mainly for pick and place operations, such as pneumatic, electric gripper, vacuum suction cup, etc.
- B. Tools: Mainly for processing operations, such as welding, cutting, surface treatment, etc.

DANGER

1. More attention must be paid to the design of the end effector to prevent power loss or any other errors that could lead to workpiece falling or damage.
2. The tool-type end effector is usually equipped with high voltage, high temperature and active rotary shaft. Special attention should be paid to the operating safety.
3. The end effector should be mounted firmly on the robot to avoid workpiece fall during operation which may cause personal injury or hazard.

WARNING

1. The end effector may be equipped with its own control unit. During installation, pay attention to installed location. Ensure that the control unit does not interfere with robot operation.
2. The gripper-type end effector should prevent the workpiece from dropping or damaging when the robot experiences a power error or other errors. If potential dangers or abnormal situations exist when using end effector, the associated hazards must be handled by the system integrator in accordance with the related standards.

Precautions for using Hydraulic and Pneumatic

 **DANGER**

1. When using the pneumatic or hydraulic system, the gripped workpiece may fall due to insufficient pressure or gravity.
2. The pneumatic or hydraulic system must be equipped with the relief valve, so that it can be applied in an emergency.

 **WARNING**

1. More attention should be paid to the pressure remained in the pneumatic systems after the power is disconnected.
2. The internal pressure must be released before the pneumatic systems are maintained.
3. More attention should be paid to the pressure in the pneumatic system as it is several times more than the atmosphere pressure.

Emergency Stop Switch Precautions

 **DANGER**

1. The robot or other control component should have at least one device for immediate halt, such as an emergency stop switch.
2. The emergency stop button must be installed in an easily accessible location for quick stop.
3. While executing an emergency stop, power to the servo motor will be cut, and all movements will be stopped. And the control system will be shut down. Emergency stop should be reset if the restoration of operating procedure is wanted.
4. Avoid using emergency stop to replace a normal stop procedure. This could reduce the lifespan of the robot.

 **WARNING**

1. When an emergency stop is performed, the power of the drive is cut off, all operations are stopped, and the control system of the robot arm is turned off.
2. To resume execution, reset the emergency stop switch.
3. Emergency stop is immediate stop: Immediately stop the movement of the robot arm and cut off the power of the drive.
4. The emergency stop switch is for emergency stop only.
5. HIWIN's industrial robot arm has two emergency stop switches, one of which is located on the teach pendant and the other is automatically connected to the controller via a dedicated cable. If there is a need for other emergency stop switches, the other means of connection can be used to achieve the purpose of emergency stop.
6. Based on the relevant industrial safety regulations, the emergency stop switch needs to be directly connected to the control box of the robot arm through a physical connection line.

1. Product Description

1.1. Software Overview

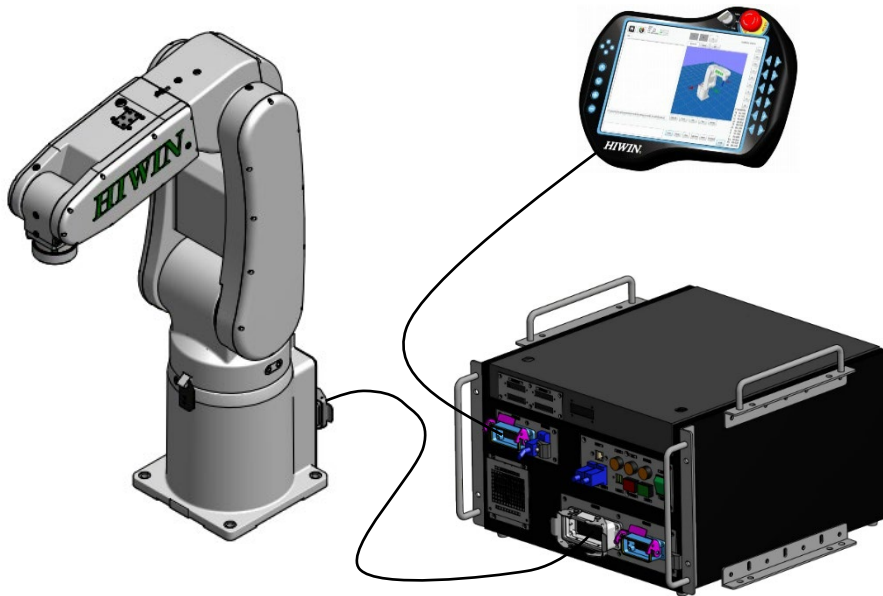
The following software will be applied:

- ◆ HIWIN Robot System Software
- ◆ Windows 7 Embedded

1.2. Robot Overview

The robot is comprised of the following parts:

1. Robot (Six-axis robot or Four-axis robot)
2. Cable
3. Control System
4. Teach Pendant (HRSS software)



Robot illustration

1.3. Hiwin Robot System Software (HRSS) Overview

Description

The HIWIN Robot System Software (HRSS) controls all basic functions for the robot.

- ◆ Path planning
- ◆ I/O management
- ◆ Data and file management...
- ◆ HRSS

The interface is called HIWIN ROBOT SYSTEM SOFTWARE (HRSS).

Features:

- ◆ User management
- ◆ Program editor
- ◆ Robot language
- ◆ Inline forms for programming
- ◆ Message display
- ◆ Configuration windows

Offline version recommended environment

- ◆ Window 7
- ◆ Resolution 1360x768 above
- ◆ Console ->all console project ->Display: Small (100%)

CAUTION







The operating interface may differ from the standard model depending on the user's settings.

1.4. Manual Description of HIWIN technologies corp.

Each product manual for robot of Hiwin Technologies Corp. is mainly classified into the Manipulator Manual, Controller Manual and Software Manual, Remote Robot Controller Software Manual, Software Development Kit Manual and Programming Command Set Manual etc., total six manuals. This manual is the manual of the software system (HRSS), specifically for describing the software functions and instructions of the Robot HRSS system software. If there are any other requirements, please refer to other manuals. The applicable products collocating with their manuals are described as follows:

Name Robot model	Manipulator Manual	Controller Manual	Software Manual	Remote Robot Controller Software Manual	Software Development Kit Manual	Programming Command Set Manual
Articulated Robot Series (RA605, RA610, RA620)	V	V	V	X	V	V
SCARA Robot Series (RS405, RS410)	V	V	V	V	V	V
Delta Robot Series (RD403)	V	V	V	X	V	V
Wafer Series (RW Series)	V	V	V	X	X	X
Electric Gripper Jaw X Series	V	X	V	X	X	X
Electric Gripper Jaw S Series	V	X	X	X	X	X

※Example: RA605-710-GC articulated robot product.

No.	1	2	3	4
Picture				
Name	Manipulator Manual	Controller Manual	Software Manual	Remote Robot Controller Software Manual
Content	Specification of robot manipulator, optional parts, maintenance/repair.	Specification of robot controller, optional parts of controller, maintenance/repair.	HRSS software installation of HIWIN robot system software, interface operation, example of application program.	Installation of remote connection software Caterpillar, interface operation, example of application program. (Note: Applicable for SCARA-LU series, not applicable for 6-axis)
No.	5	6		
Picture				
Name	Software Development Kit Manual	Programming Command Set Manual		
Content	Provide robot function library. SDK is furnished for users to write their own software interface to manipulate the robot.	Hiwin robot system software HRSS shares with remote connection software Caterpillar, Description of how to use program Commands.		

2. Operation

2.1. Teach Pendant

2.1.1. Front view

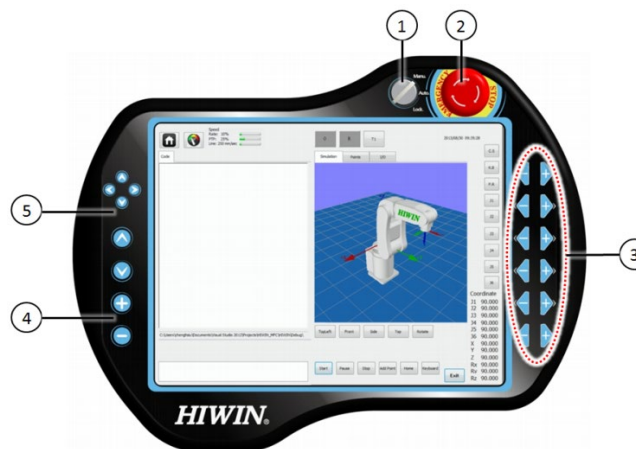
Function

The Teach Pendant is a portable programming device for the robot, which can provide both programming operation and display. It is equipped with a touch screen: the HRSS can operate by finger or stylus without an external mouse and external keyboard.

CAUTION

In this reference, the Teach Pendant is referred as “TP”

Overview

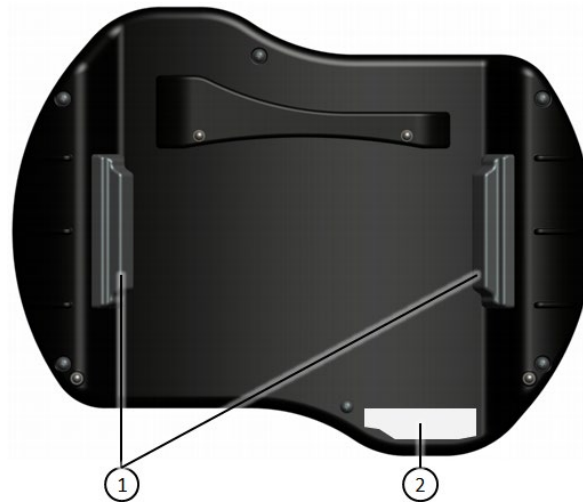


Front view of HIWIN's teach pendant

No.	Description
1	Used to change operation mode and turn on/off the monitor. Only after the key is inserted, the switch can be turned.
2	Emergency Stop Button, used to stop the robot in emergency. It will lock when the Emergency Stop Button is pressed.
3	Run Button, used for JOG.
4	Velocity adjustment
5	Space Operation Button

2.1.2. Rear View

Overview



Rear View of HIWIN's Teach Pendant

1. Enabling Switch
2. Model Plate

Description

Component	Description
Enabling Switch	<p>The Enabling Switch has three positions:</p> <ul style="list-style-type: none"> ● Not pressed ● Center position ● Pressed <p>In T1 or T2, the Enabling Switch must be kept in the center position to start the robot. In the Auto Run and External Auto Run modes, the Enabling Switch will not activate.</p>
Model Plate	Model Plate

⚠ WARNING

It is prohibited to remove or insert Teach Pendant when the controller is under powered condition, this is to prevent damaging Teach Pendant. If Teach Pendant is removed under powered condition, the emergency stop will be activated immediately. Only reinsert the Teach Pendant can deactivate the emergency stop.

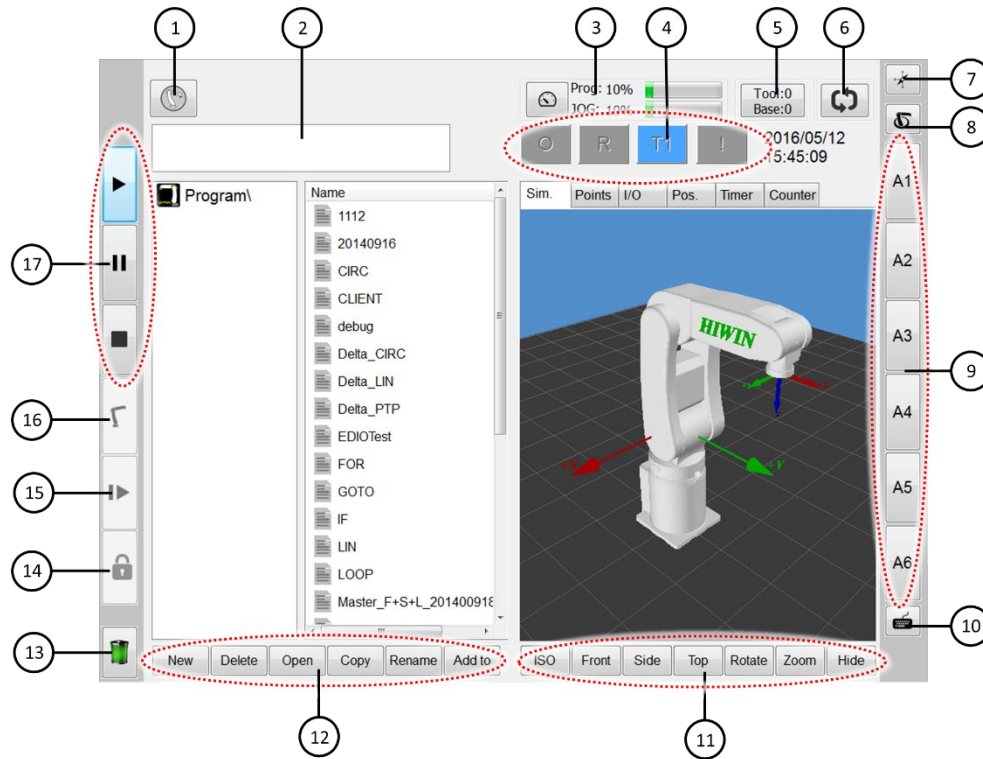
⚠ WARNING

If the Teach Pendant is removed, the robot cannot be stopped from the Emergency Stop Button on the Teach Pendant. An external Emergency Stop Device is required to connect to the control system.

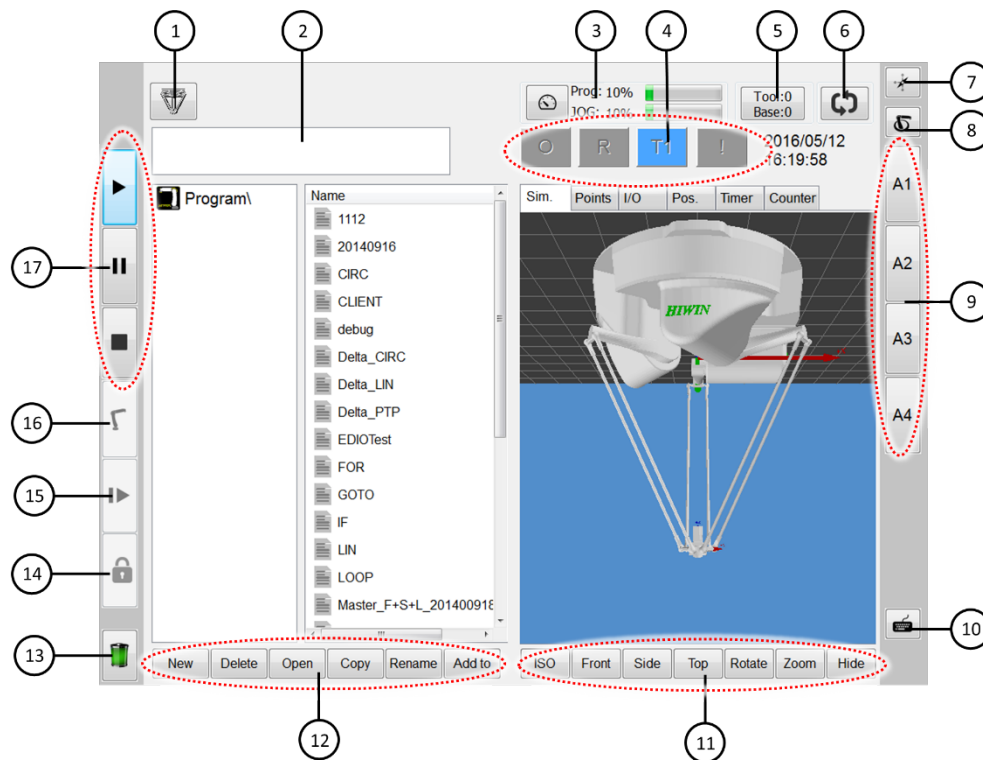
The providers should take responsibility to remove the Teach Pendant from the robot and keep it safe. It should be stored out of operator's view and away from contact. The purpose is to avoid confusion of valid and invalid emergency stop devices.

Failure to take these measurements could lead to serious injury, death or equipment damage

2.2. HRSS Interface



6-axis robot HRSS interface



Delta robot HRSS interface

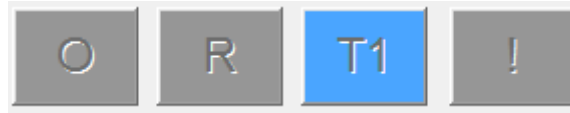
No.	Name	Description
1	Main Menu	Display the Main Menu

2	Error Information Window	Displays the error information according to the default configuration.
3	Program Ratio and Jogging Ratio	Displays by the program to change the ratio.
4	Status Bar	
5	Tool and Base	Displays the selected tool and base number. Click to change the tool or base number.
6	Step motion	Step motion and continue motion
7	Teach Pendant Configuration	Touch the display to select the relative position where the Teach Pendant faces the robot.
8	Coordinate system Button	Displays the current coordinate system. Touch to display all coordinate systems and select another one.
9	Run Button	If axis-specific jogging is selected, the axis number (A1, A2 etc) will be displayed here. If Cartesian jogging is selected, the direction of the coordinate (6-axis) system (X, Y, Z, A, B, C) will be displayed; If coordinate (4-axis) system is selected (X, Y, Z, A4) will be displayed.
10	Keyboard Button	Click to display the keyboard.
11	Simulation View Angle Button	Shift to simulate the view angle
12	Status button	The Status button will dynamically be changed according to the window currently activated by the HRSS. Button edit is on the right, use this to transfer different command from Teach Pendant.
13	Battery Figure	Display the status of absolute encoder's battery.
14	Lock Button	When the program is executed, button will lock or unlock it.
15	Next step	motion Button
16	Home Button	Hold this button to return the robot to the home position
17	Run Control Button	Three buttons are used for run, pause and stop the program.

2.2.1. Status Bar

The status bar displays the configuration status of the robot.

Overview



HRSS status bar

2.2.2. Status Display “Interpreter”

Figure	Color	Description
	Orange	Interpreter running.
	Gray	Interpreter failure or stop.

Program execution status display for submit encoder of the robot.

2.2.3. Keyboard

The Teach Pendant is equipped with a touch screen. The HRSS can be operated by finger or stylus.

The keyboard on the HRSS can be used to enter alphabets and numbers.

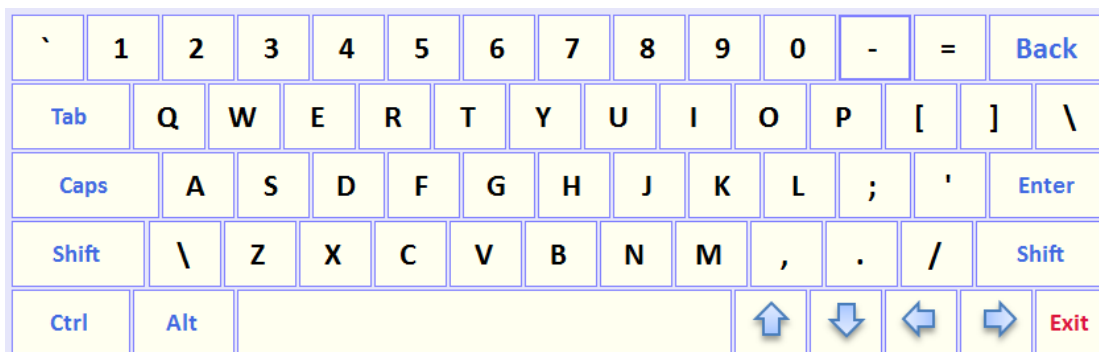


Illustration of HRSS keyboard

2.3. Connect to Control System, and Start HRSS

Operation steps

The main switch on the control system is shifted to ON.

The operating system and the HRSS will automatically start.

2.4. Open Main Menu

Operation steps

Click the Main Menu on the Teach Pendant, and then open it.

Description

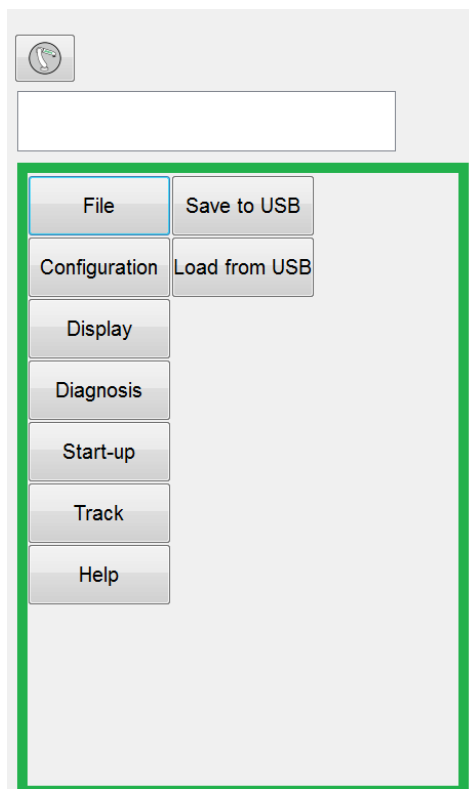
1. Property for Main Menu window

The Main Menu is displayed in the left window.

Click an item to display the next-level function table (e.g. File).

Display table or opened interface by pressing the menu button on the upper left.

You can select these function items again without closing the next-level table.



Example: opened next function table

2.5. Language Settings

Description

The interface enable three different languages to be selected for the settings: English, Traditional Chinese and Simplified Chinese.

When setting is completed, most of the interface will change to the language selected by the user immediately.

The rest will be changed after the Teach Pendant is restarted.



Language Setting Interface

Operation steps

Main menu>Configuration>Language

1. Select language
2. Restart Teach Pendant to complete full language change

2.6. Time and NTP Settings

Description

User could use the Time setting in HRSS to increase the Network Time Protocol (NTP) server function. When using time and NTP settings mention above, user will be notified with warning signal under continuous operation. Data will be recorded in LOGBOOK for further references to know when the time has been edited. Only under the Expert Mode has the permission to use time and NTP settings.

Operation steps

1. Time Settings
 1. Main menu>Configuration>User group>Expert
 2. Main menu>Configuration>Time setting
 3. Enter required time under time setting
 4. Press SET button

2. NTP Settings
 1. Main menu>Configuration>User group>Expert
 2. Main menu>Configuration>Time setting
 3. Enter required NTP under NTP setting
 4. Press SET button

Time Setting :

Year :

Month :

Day :

Hour :

Minute :

Second :

NTP Setting :

NTP :

Time and NTP Settings Interface

2.7. Change User Group

Operation steps

1. Select [Configuration]>[User group] on the Main Menu to display the current user group.
2. To switch to the default user group, press [Change]. To switch to other user groups, press [Login] ... select the user group.
3. If needed, enter password and login.

Description

In the HRSS, the functions can be selected according to the user group. The following user group exist:

1. Operator
Operator group
This is the default user group.

2. Engineer

Engineer group

This user group is protected by a password. The default password is “HIWIN”.

Modification unavailable.

3. Expert

Expert group

This user group is protected by a password. The default password is “HIWIN”.

Modification unavailable.

No.	Function	Operator	Engineer	Expert
	Function Table			
1	File	X	X	O
2	Configuration>User group	O	O	O
3	Display>Input/Output	X	O	O
4	Display>Variable	X	O	O
5	Display>Mileage	O	O	O
6	Display>Utilization	O	O	O
7	Display>Motor Torque	O	O	O
8	Display>Fieldbus	X	X	O
9	Diagnosis>Logbook	O	O	O
10	Start-up>Calibrate	X	X	O
11	Start-up>Master	X	X	O
12	Start-up>Robot data	X	O	O
13	Start-up>Network Config	X	X	O
14	Start-up>RS-232	X	X	O
15	Start-up>System Setting	X	X	O
16	Start-up>Electric Gripper	X	X	O
17	Start-up>External Axis	X	X	O
18	Start-up>Arc Weld	X	X	O
19	Track>Setting	X	O	O
20	Track>Vision Setting	X	O	O
21	Track>Vision Object	X	O	O
22	Track>Calibration	X	O	O
23	Track>Monitor	O	O	O
24	Help>About	O	O	O
25	Help>Operating Time	O	O	O
26	Help>Update	X	X	O
27	Help>TP Calibration	O	O	O

No.	Function	Operator	Engineer	Expert
28	Help>Manual	O	O	O
	Interface			
29	Message box	O	O	O
30	Velocity configuration	X	O	O
31	Tool/base coordinate	X	X	O
32	Teach Pendant configuration	X	O	O
33	Change JOG coordinate system	X	O	O
34	JOG	X	O	O
35	On-screen keyboard	O	O	O
36	Remove teach pendant	X	O	O
37	Step execution	X	X	O
38	Program Execution	O	O	O
39	Program selection	O	O	O
40	Modify program	X	X	O
41	Tool/base calibration	X	O	O
42	IO operation	X	O	O
43	Functional IO modification	X	X	O

2.8. Change Run

CAUTION

Don't modify the run during programming period. If it is changed, the robot will stop.

Prerequisite

The controller doesn't processing any program.
Insert the key for the mode selector switch.

Operation steps

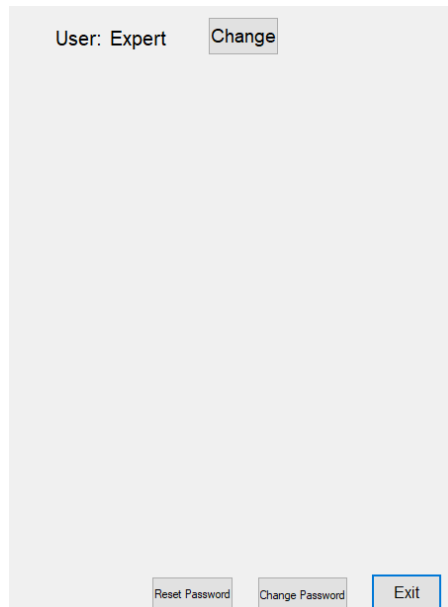
1. Turn the mode selector switch on the Teach Pendant, and display the mode selection.
2. Select the run.
3. The selected mode will be displayed in the status bar of the Teach Pendant.

Mode	Application	Velocity
T1	Used for test operation, programming and teaching	Program verification: Programmed velocity, maximum 250 mm/s Jog Mode: Jog velocity, maximum 250 mm/s
T2	Used for run test	Program verification: Programmed velocity Jog Mode: Unable to run
AUT	Used for robot without the higher-level controller	Programming mode: Programmed velocity Jog mode: Unable to run
EXT	Used for the robot with the higher-level controllers (For example, PLC)	Programming mode: Programmed velocity Jog mode: Unable to run

2.9. Change Password

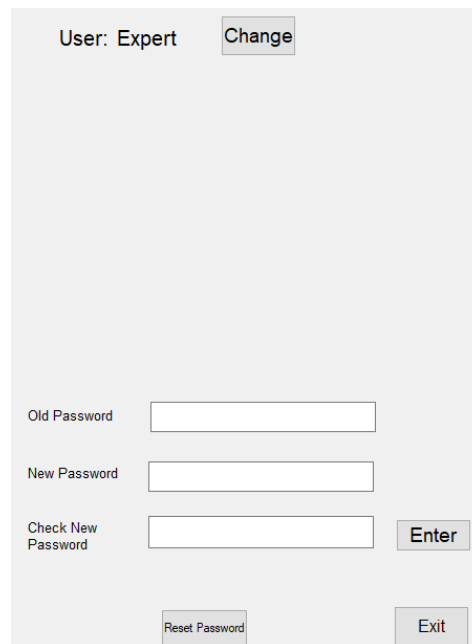
Operation steps

1. Select [Configuration]>[User group] on the Main Menu to display the current user group.



The screenshot shows a user interface with the text "User: Expert" at the top left. To its right is a button labeled "Change". At the bottom of the screen, there are three buttons: "Reset Password", "Change Password", and "Exit". The "Exit" button is highlighted with a blue border.

2. To click Change Password.



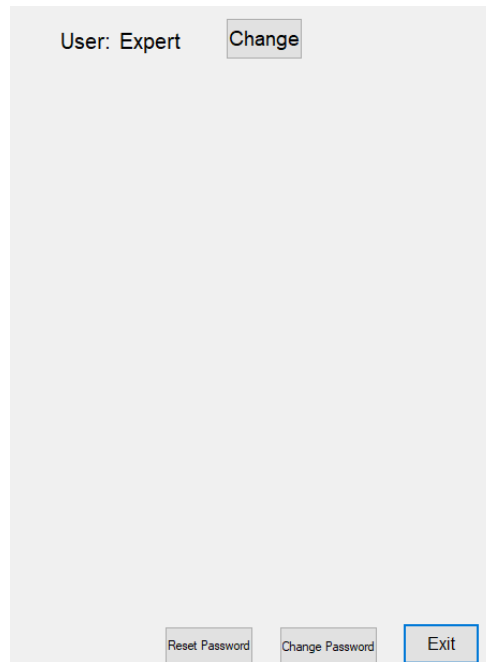
The screenshot shows the same user interface as the previous one, but now the "Change Password" button at the bottom is highlighted with a blue border. Below the "User: Expert" text, there are three input fields: "Old Password", "New Password", and "Check New Password". To the right of the "Check New Password" field is an "Enter" button. At the bottom, the "Reset Password" and "Exit" buttons are visible.

3. Enter old password, new password, and check new password (new password and check new password must be the same).

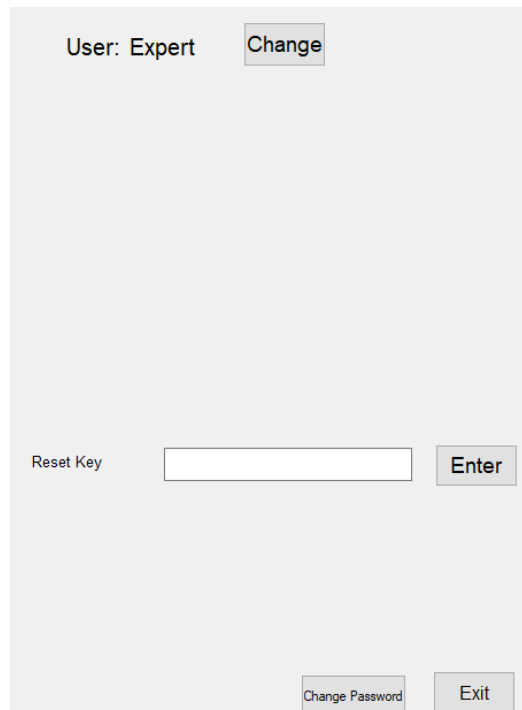
2.10. Reset Password

Operation steps

1. Select [Configuration]>[User group] on the Main Menu to display the current user group.



2. To click Reset Password.



3. Enter Reset Key password will reset to “HIWIN” as default. Reset Key should be asked from robotservice@hiwin.tw with Mac address.

Address: **No.7, Jingke Road, Taichung Precision Machinery Park, Taichung 40852, Taiwan**

Phone: **+886-4-2359-4510 #8562**

Website: **http://www.hiwin.tw**

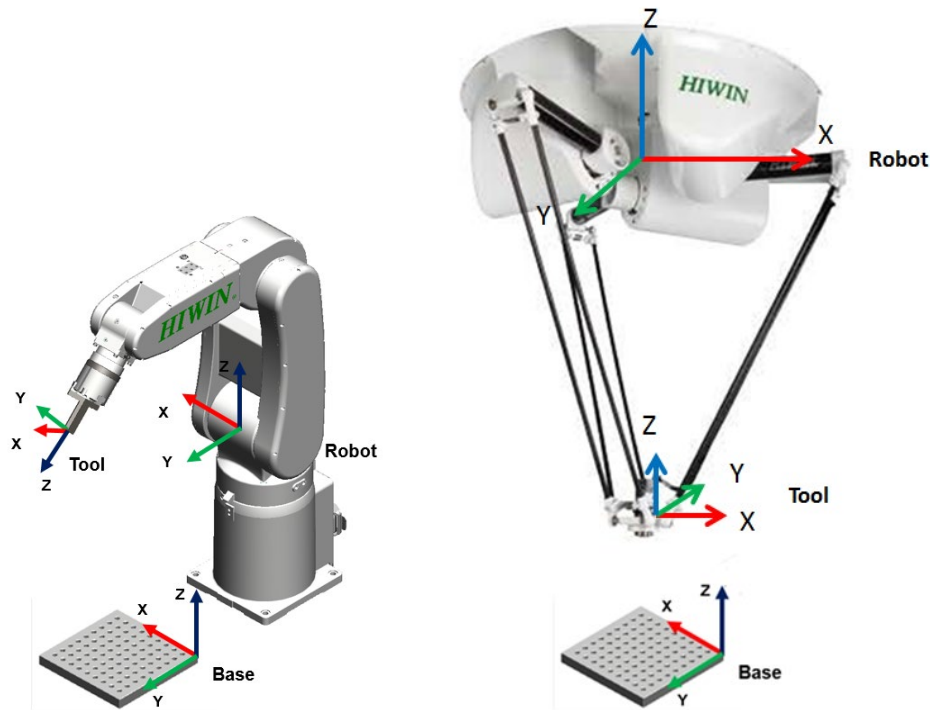
Robot ID: **RCA000000000**

Version: **HRSS 3.3.7a.6883 RS410-800-400-LU 32-bit
RBT 1.3.37.6799
RtdII 1.0.2.2705D offline
Hrc 6829
EGdII 1.1.1.823
Mac address 00:00:00:00:00:00**

State	Function
<input checked="" type="checkbox"/> Activated	HRSDK
<input checked="" type="checkbox"/> Activated	Fieldbus
<input checked="" type="checkbox"/> Activated	External Axis
<input checked="" type="checkbox"/> Activated	External TCP

2.11. Coordinate System

Overview



Coordinate System Overview

Description

1. ROBOT

The ROBOT used the Cartesian coordinate system. If it is a 6 axes robot, it will be fixed at the location of the 1st –axis center point and the 2nd –axis center point of the robot. If it is a 4 axes robot, it will be fixed at the robotic foot. This is used as the origin coordinate system of the base coordinate system.

In the default configuration, the coordinate system of ROBOT is consistent with the BASE coordinate system.

2. BASE

The BASE Coordinate System is Cartesian system used to describe the position of the workpiece. It is based on the ROBOT Coordinate System.

By default, the Base Coordinate System is consistent with the ROBOT system. A user can move it to the workpiece.

3. TOOL

The TOOL Coordinate System is a Cartesian system, located at the tool center point.

By default, the home of the Tool Coordinate System is located at the flange center point (called the Flange Coordinate System). The Tool Coordinate System is offset to the tool center point by the user.

Rotation of the six axes robot coordinate system

Corner	Rotation around axis
A	Rotate around X axis
B	Rotate around Y axis
C	Rotate around Z axis

Rotation of delta robot coordinate system

Corner	Rotation around axis
A4	Rotate around Z axis

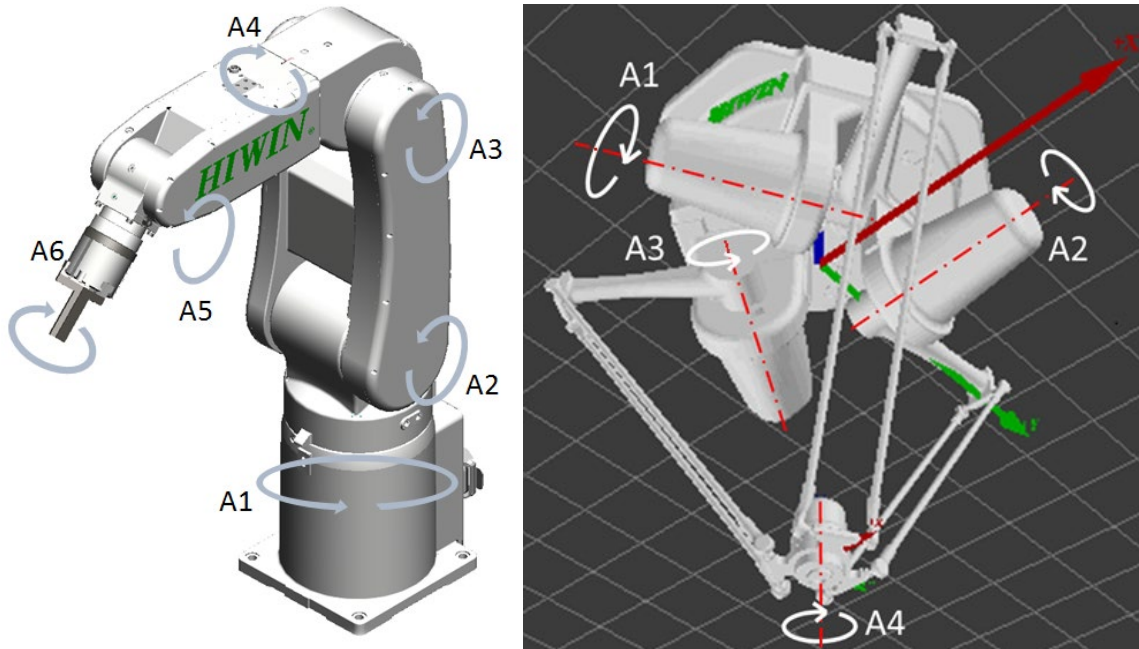
2.12. JOG

Description

There are two types of jogging:

Cartesian jogging, TCP (Tool Center Point) is jogged in the positive or negative direction along an axis of the coordinate system.

Axis-specific jogging, each axis can independently be moved in a positive or negative direction.



Axis-specific jogging

2.13. Manual Move

Description

When Administrator rights are Engineer or Expert, under the manual mode of T1, it is possible to configure all the parameters in the manual move options window.

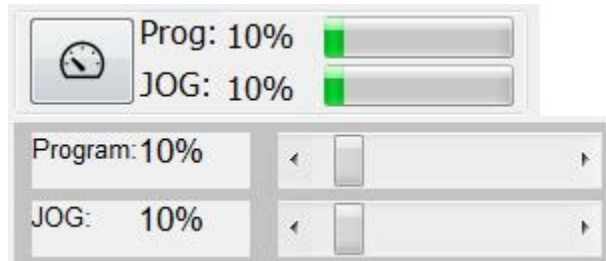
2.13.1. Jog Velocity Change

Description

Manually drag the bar to change the jog velocity or see 4.11.4 by using the +/- button at the left side of the Teach Pendant to set the ratio.

Operation steps

1. Open the speed options window (shown as the chart button).
2. Change the JOG speed.
3. It is also possible to use the left button of teach pendant to change the JOG speed.



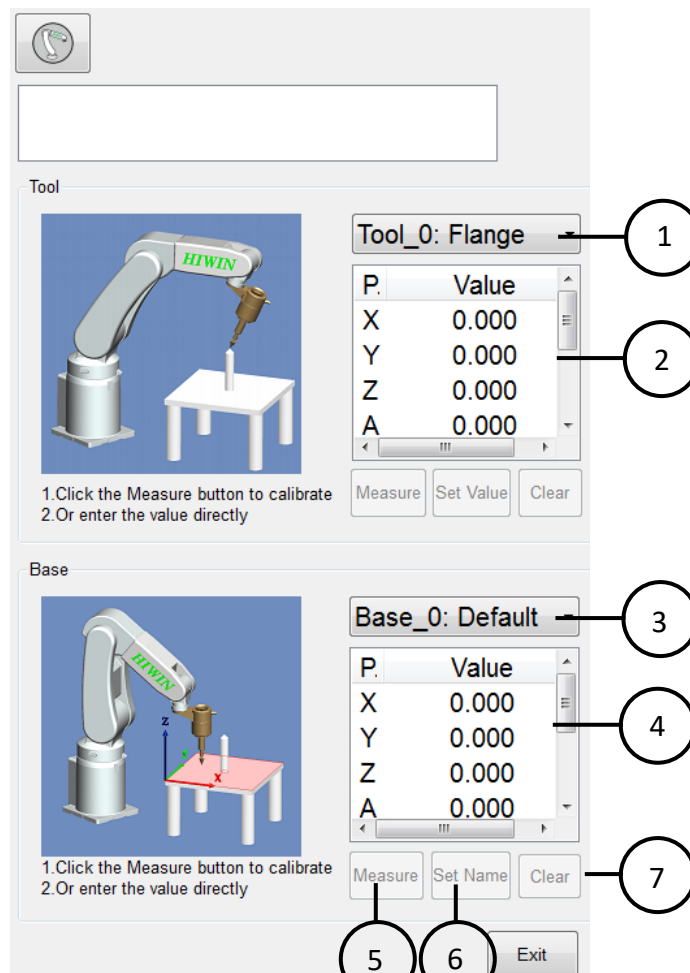
Configuration related to speed of jog

2.13.2. BASE/TOOL Coordinate

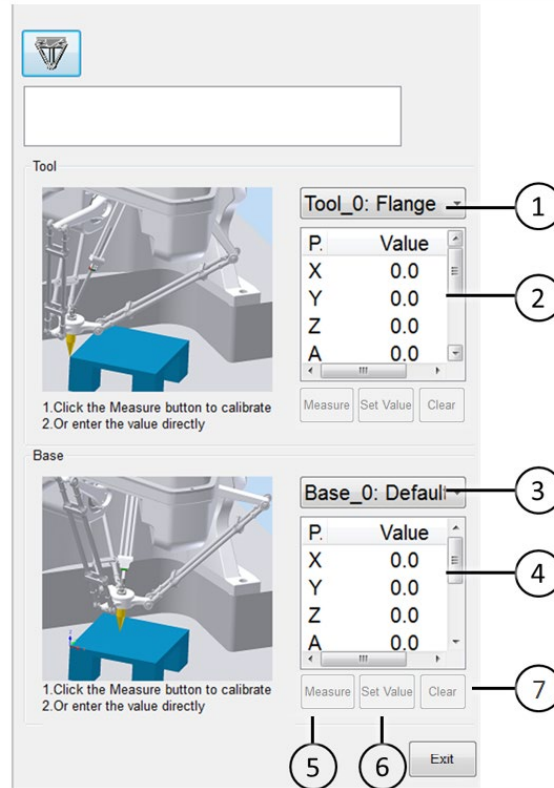
Description

View and modify the base or tool coordinate

16 tool and 32 base coordinate systems can be saved in the control system at most. When you apply the Cartesian jogging, you must select a tool (Tool Coordinate System) and a base (Base Coordinate System).



6-axis robot base/tool window



Delta robot base/tool window

No.	Description
1	Tool coordinate currently selected
2	Parameters relate to selected tool coordinate
3	Base coordinate currently selected
4	Parameters relate to selected base coordinate
5	Recalibrate
6	Directly enter a value on the selected item to calibrate
7	Delete the selected coordinate parameters and name

Operation steps

1. Open the TOOL/BASE window.
2. You can directly choose the tool/base coordinate system by click on ① and ③ in figure respectively.

2.13.3. Teach Pendant Position Configuration

Robot installation method

Floor Mode

Operation Steps

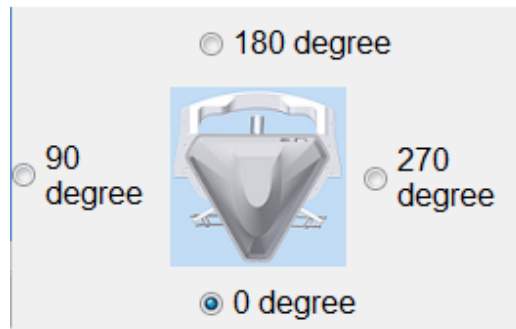
1. Start-up -> Robot data -> Mount Position
2. Select Floor mode

Description

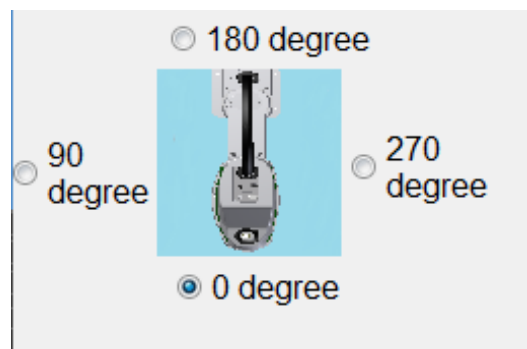
Define the user's position relative to the robot before you use space button.



Window of 6-axis robot teach pendant configuration



Window of delta robot teach pendant configuration



Window of delta robot teach pendant configuration

Robot Installation method

Ceiling Mode

Operation Steps

1. Start-up -> Robot data -> Mount Position
2. Select Ceiling mode

Description

Define the user's position relative to the robot before you use space button.



Window of 6-axis robot teach pendant configuration

2.13.4. Jogging velocity Ratio

Description

The jogging velocity ratio is the robot velocity during jogging. It is presented by percentage, based on the maximum velocity when the robot is jogging. That value is 250mm/s.

Operation steps

1. Click the jogging ratio button.
2. Set the desired jogging ratio. Set with the +/- button or the adjustor.
3. Touch the area outside the jogging ratio window. The window closes and the ratio is applied,

Other method

Use the +/- button at the left side of the Teach Pendant to set the ratio.

2.13.5. Axis-specific jogging with the jog keys

Prerequisite

T1 mode

Operation steps

1. Select the "JOINT" as the coordinate system for the jog key.
2. Set jogging velocity ratio.
3. Hold the enabling Switch.
4. Axis A1 to A6(or A1 to A4) beside the jog keys.

Remarks: In the joint coordinate axis, the arm of the articulated robot labeled Joint1 (J1) is represented as A1 in the software, Joint2 (J2) is represented as A2 in the software and Joint3 (J3) is represented as A3, and so on to Joint6.

5. Press the +/- button, so that the axis move towards the positive or negative direction.

2.13.6. Cartesian jogging with the jog keys

Prerequisite

T1 mode

The tool and base coordinate systems have been selected.

Operation steps

1. Select the "XYZ" as the coordinate system for the jog keys.
2. Set manual ratio.
3. Hold the enabling Switch.
4. Axis X, Y, Z and RX, RY, RZ(or axis X, Y, Z and A) will display beside the jog keys.
 - ◆ X, Y, Z: Used for linear motion along the axis of the selected coordinate system.
 - ◆ RX, RY, RZ: Used for rotation motion along the axis of the selected coordinate system.
 - ◆ A: Used for rotation motion along Z axis.
5. Press the +/- button, so that the axis move towards the positive or negative direction.

CAUTION

The position where the TCP is located can be displayed in the following method: select Main Menu>Display > Actual Position.

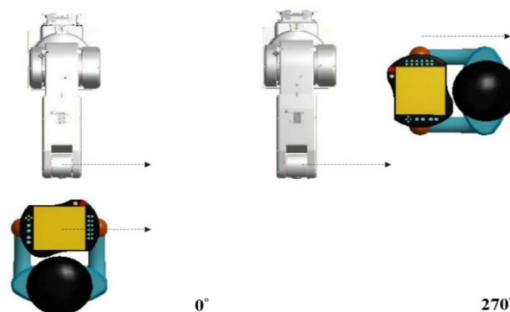
2.13.7. Teach Pendant Alignment

Description

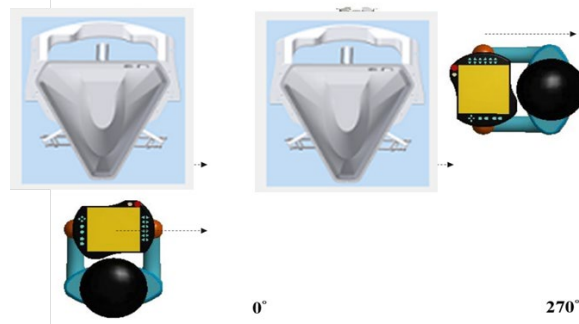
Adjust according to the user location, so that the direction of TCP movement adapts the rotation of the Teach Pendant.

The user location is selected in a unit of angle. The reference point for that angle is on the base.

Default: 0°. This corresponds to a user standing on the opposite side of the robot.



Teach Pendant configured at 0° and 270°(Six axes robot)



Teach Pendant configured at 0° and 270°(Delta robot)

Prerequisite

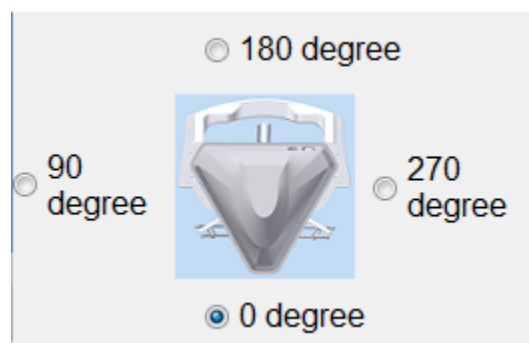
T1 mode

Operation steps

1. Open the window of Teach Pendant Configuration.



Window of six axes robot teach pendant configuration



Window of delta robot teach pendant configuration

2. Set the position where the Teach Pendant is located with regards to the robot.
3. Close the window of the Teach Pendant configuration.

! CAUTION

When switching to the external auto run, the Space Operation Button will be automatically positioned as 0°.

2.13.8. Move with Space Operation Key

Description

Operate the arm to forwardly, backwardly, left, right, up and down according to the angle and direction set in 2.13.7.

Prerequisite

T1 mode

Ensure the Teach Pendant is positioned

Operation steps

1. Set the manual ratio
2. Hold the Enable Switch.
3. Press the Space Operation button, so that the arm can move to the relative direction.

CAUTION

When the space operation is used for manual movement, the moving direction depends on the position of Teach Pendant, not relevant to Base coordinate.

2.14. Display

2.14.1. Display Actual Position

Operation steps

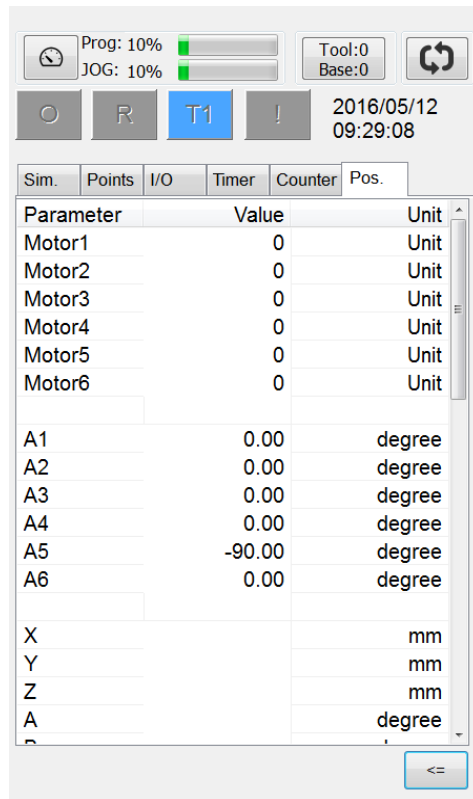
Click the operating page of [Pos.].

Description

Display the motor position, the axis angle and the Cartesian coordinate of the current base.

If the 6-axis robot is operated, 6-axis information will be displayed.

If the 4-axis robot is operated, 4-axis information will be displayed.



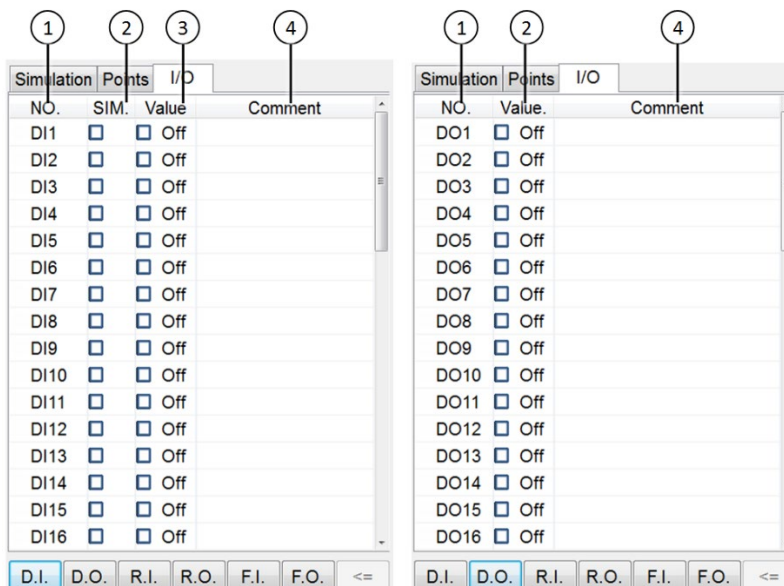
Window of actual position

2.14.2. Display Digital Input/Output

Operation steps

1. Click the operating page of [I/O].
2. Click [D.I.] or [D.O.].

Description



Digital input/output interface

No.	Description
1	Input/Output No.
2	Simulation, opened as red
3	The input/output simulated signal (can be used when the simulation is selected) ON is displayed in red and showed On. OFF is displayed in white and showed Off.
4	Input/output name (double-click to modify)

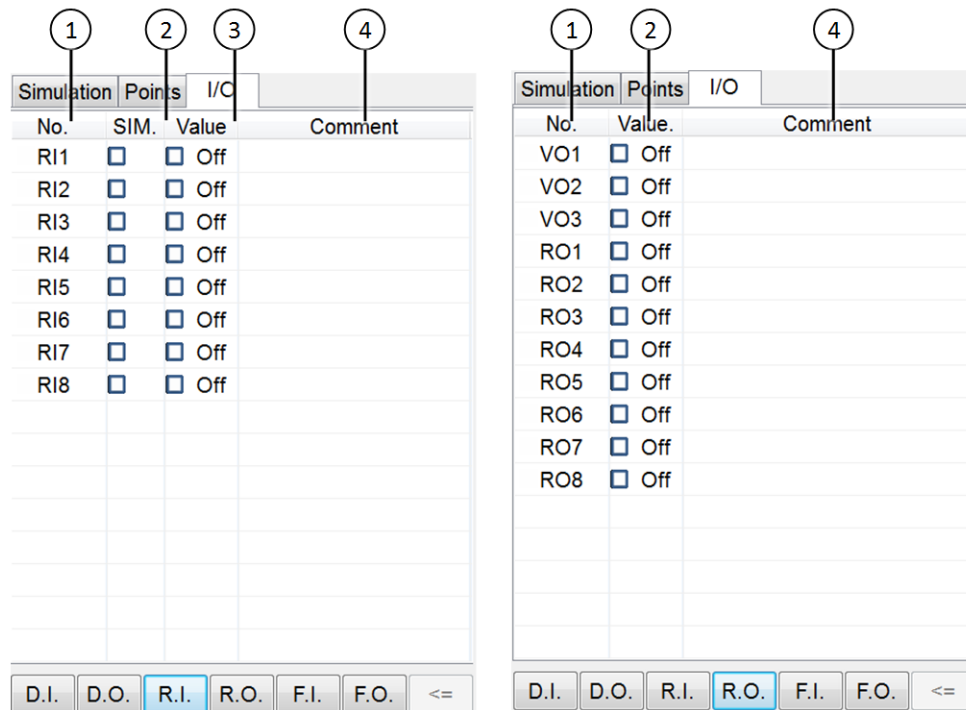
※When equipped with one I/O card (standard), 16 DIO can be used in HRSS, if there is two I/O card (optional), HRSS can use 48 DIO.

2.14.3. Display Remote Input/Output

Operation steps

1. Click the operating page of [I/O].
2. Click [R.I.] or [R.O.].

Description



Robot input/output interface

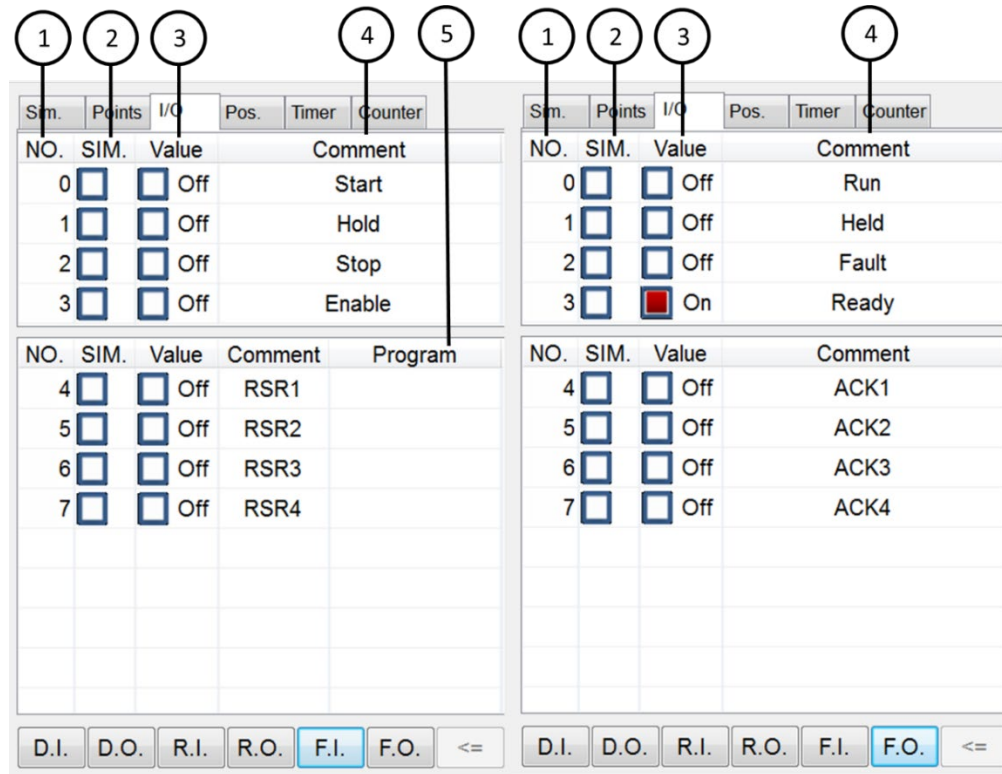
No.	Description
1	Input/Output No.
2	Simulation, opened as red.
3	The input/output simulated signal (can be used when the simulation is selected) ON is displayed in red and showed On. OFF is displayed in white and showed Off.
4	Input/output name (double-click to modify)

2.14.4. Display External Functional Input/Output

Operation steps

1. Click the operating page of [I/O].
2. Click [F.I.] or [F.O.].

Description



External Auto Run input/output interface

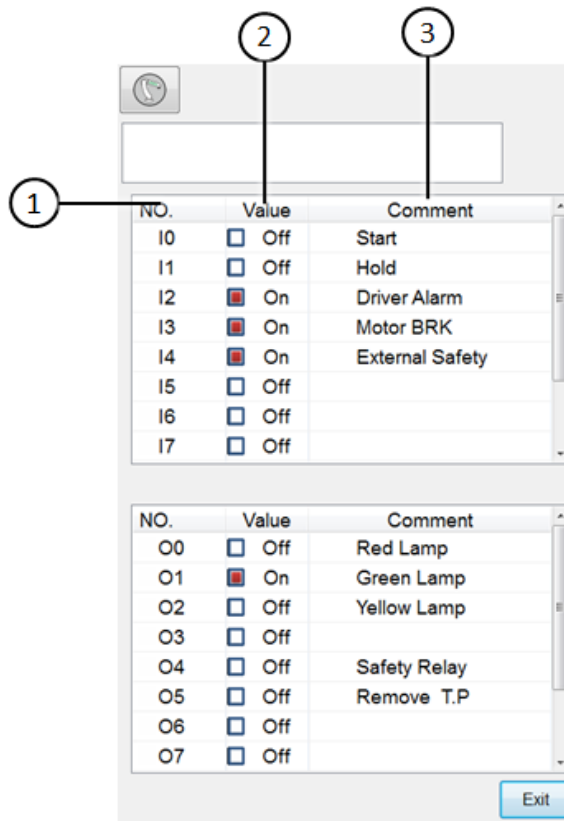
No.	Description
1	Input/Output No.
2	Simulation, opened as red
3	The input/output simulated signal (can be used when the simulation is selected) ON is displayed in red and showed On. OFF is displayed in white and showed Off.
4	Input/output name
5	Program name Click and hold two seconds to remove the program

2.14.5. Display System Status Input/Output

Operation steps

Main Menu>Display > System I/O

Description



System input/output interface

No.	Description
1	Input/Output No.
2	The opened input/output signal displays in red and shows On. OFF is displayed in white and showed Off.
3	Input/output name

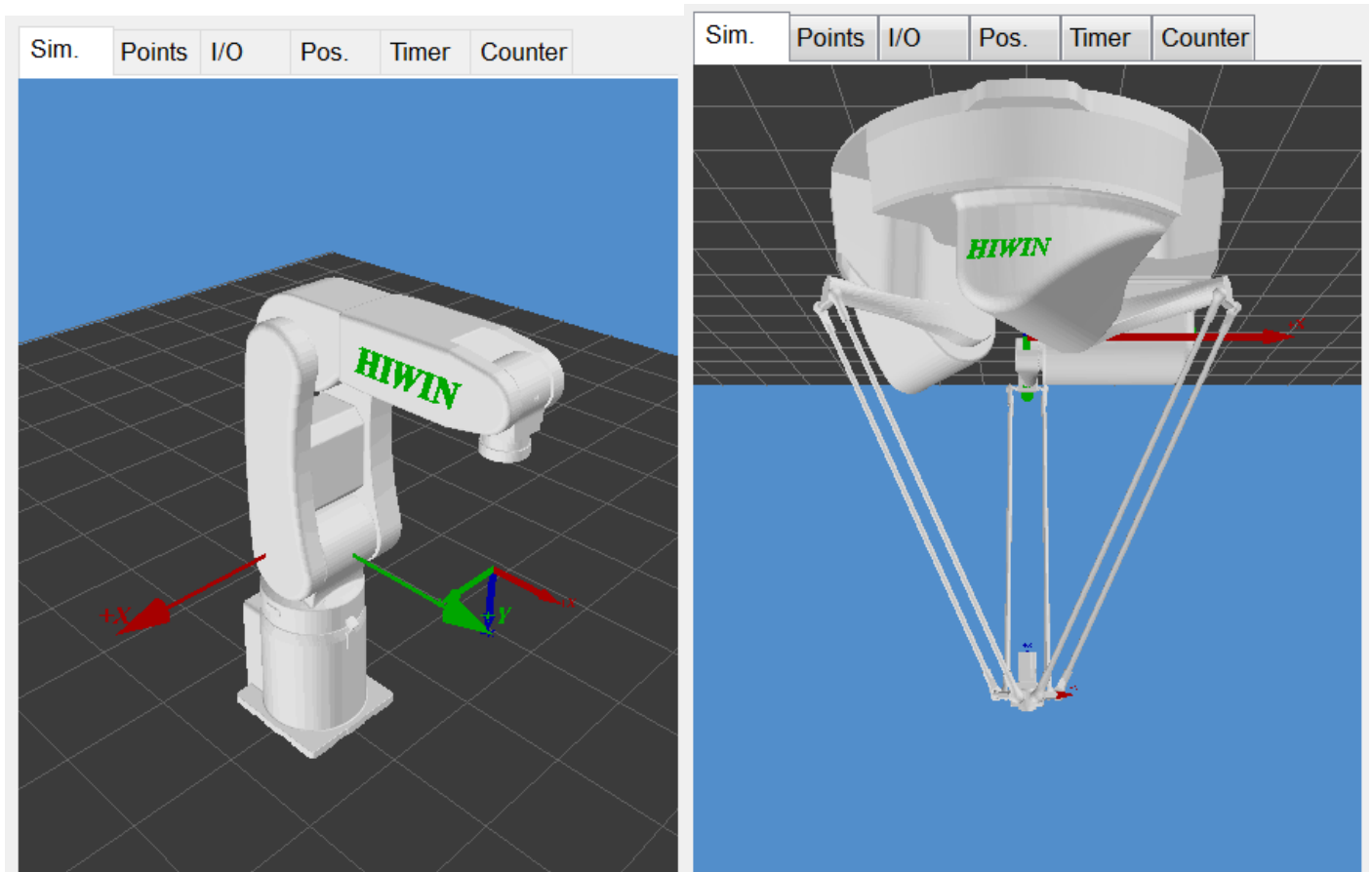
2.14.6. Display Robot Simulation Screen

Operation steps

Click the [Sim.] on the screen.

Description

Displays the posture when the robot runs or simulates the program.



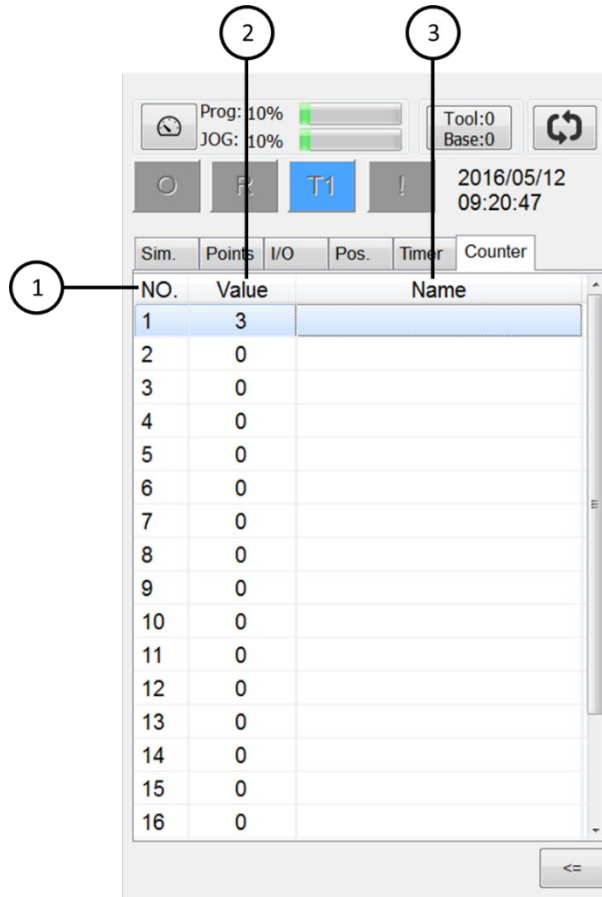
Robot simulation screen

2.14.7. Display Counter Variable

Operation steps

Click the operating page of [Counter].

Description



Counter interface

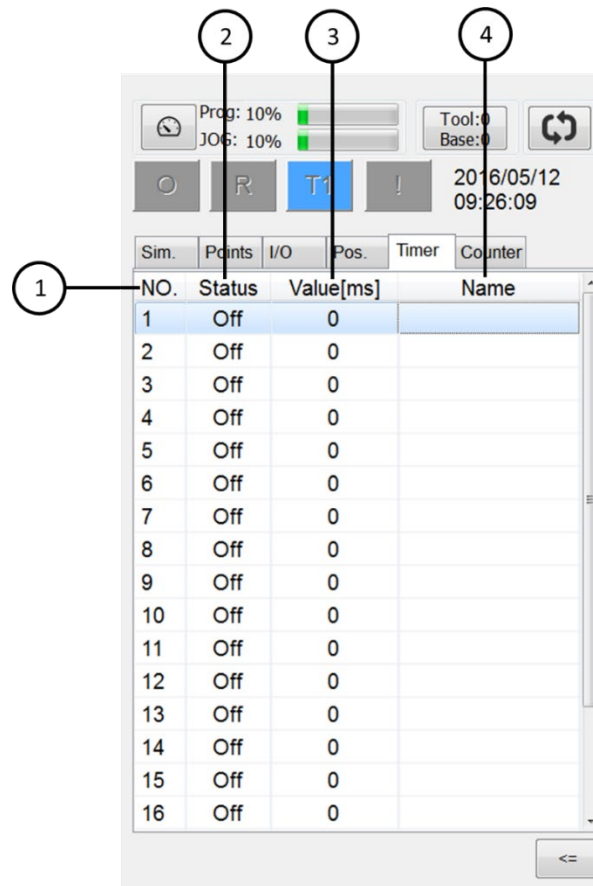
No.	Description
1	Counter No.
2	Counter value
3	Counter name (double-click to change the name)

2.14.8. Display Timer Variable

Operation steps

Click the operating page of [Timer].

Description



Timer interface

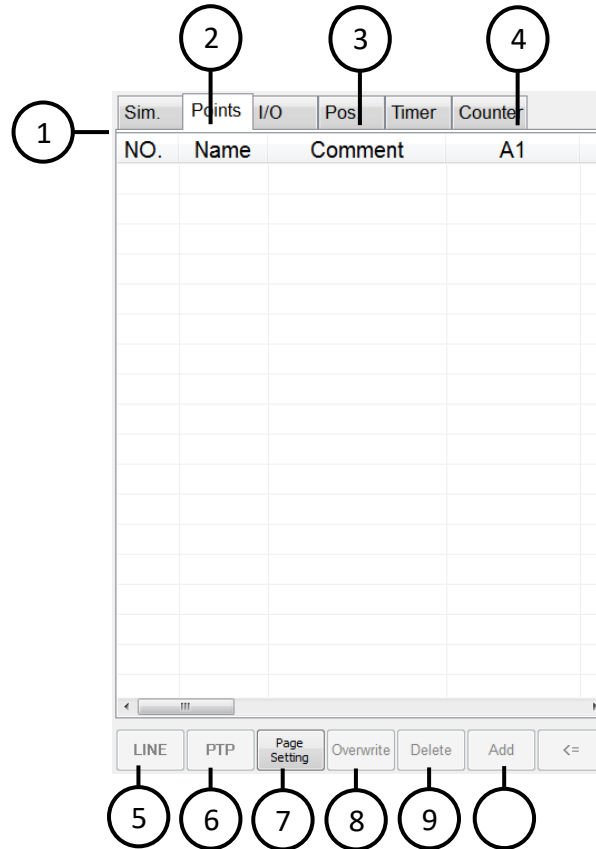
No.	Description
1	Timer No.
2	Timer status On Off
3	Timer value
4	Timer name (double-click to change the name)

2.14.9. Display Point List

Operating Steps

Click the operating page of [Points].

Description

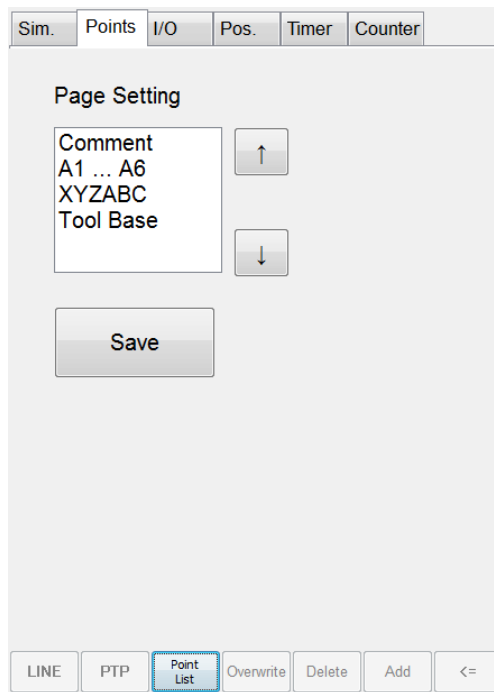


Points interface

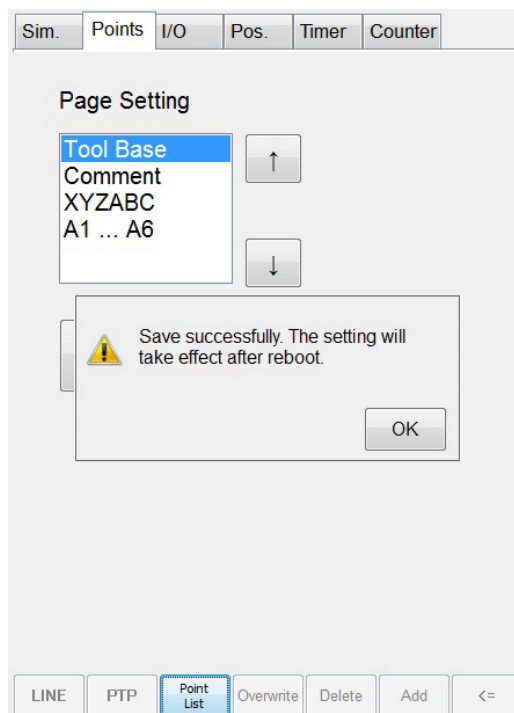
Item No.	Description
1	Numbering for points
2	Name for points
3	Functional comment for points (free to edit by oneself)
4	Information for points, includes angle of each axis (A1~A6), Cartesian coordinates (X, Y, Z, A, B, C), plus numbering of Tool and Base used.
5	Select a point and move to that point with LINE mode.
6	Select a point and move to that point with PTP mode.
7	Function to adjust the arrangement of data
8	Select a point and replicate the information for that point.
9	Select a point and delete that point.
10	Newly add a point with the current information.

Description of Adjusting Arrangement of Data

When [Page Setting] is clicked, figure below will appear, user can self-adjust the order of arrangement. Once the adjustment is completed, click [Save] button to store the setting. A notification will appear to remind user that the setting will take effect after reboot (figure below). After reboot, point list will appear with the format set by the user (figure below).



Adjusting arrangement of data interface



[Save] clicked, notification message appeared

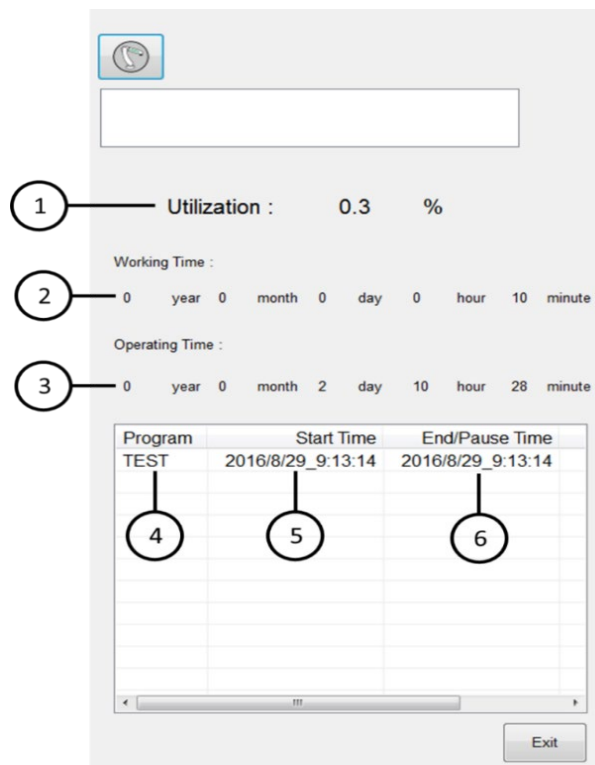
1	Numbering of motor axis
2	Total accumulated mileage
3	Current mileage (can be zeroed)
4	Unit of mileage (number of turns)
5	Zeroed reset 「 Current Mileage 」

2.14.11. Display Utilization Rate

Operating Steps

Main Menu>Display > Utilization

Description



Utilization Interface

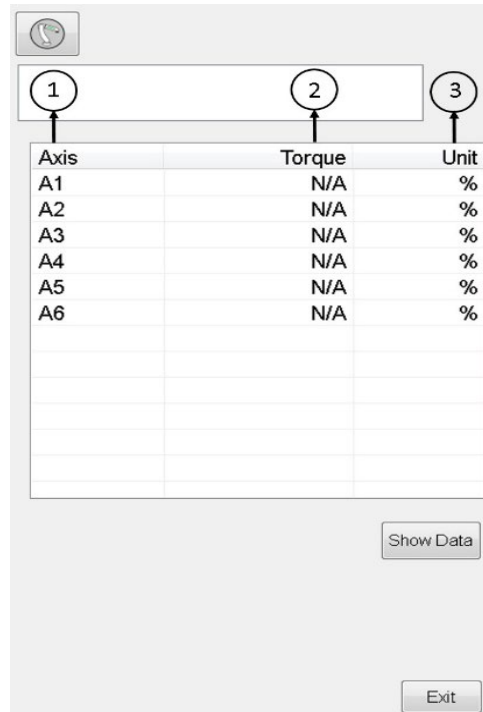
Item No.	Description
1	Utilization rate
2	Total time of executing program
3	Total time of turning on power
4	Name of executing program
5	Start time of executing program
6	End/pause time of executing program

2.14.12. Display Load Percentage

Operating Steps

Main Menu>Display > Motor Torque

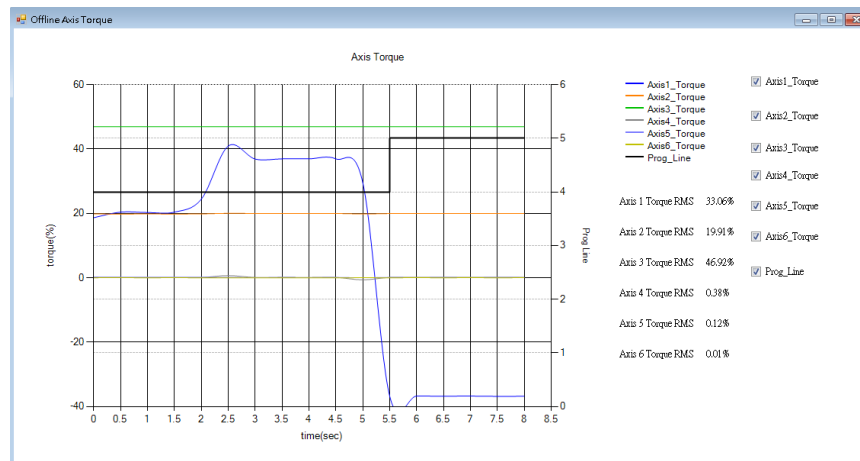
Description



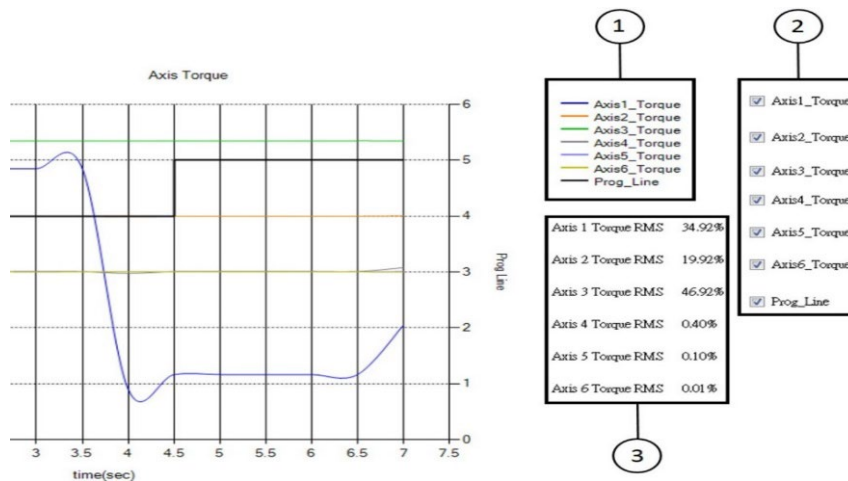
Motor Torque Interface

Item No.	Description
1	Numbering of motor axis
2	Load percentage of motor
3	Unit (percentage)

In offline, when 6-axis robot motion stop, press Show Data can display torque analysis information of each axis.



Torque analysis interface



Torque analysis item No.

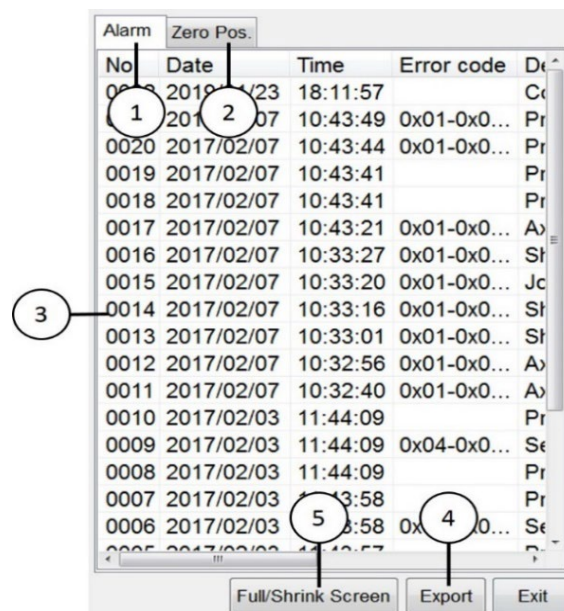
Item No	Description
1	Name and color of motor. Ps. Prog_Line is execution line of the program.
2	Root mean square value of motor torque. If the value is over 100%, there is a problem.
3	The item for adjust the visibility of the line.

2.14.13. Display Alarm and Zero Position History Message

Operating Steps

Main Menu>Diagnosis > Logbook

Description

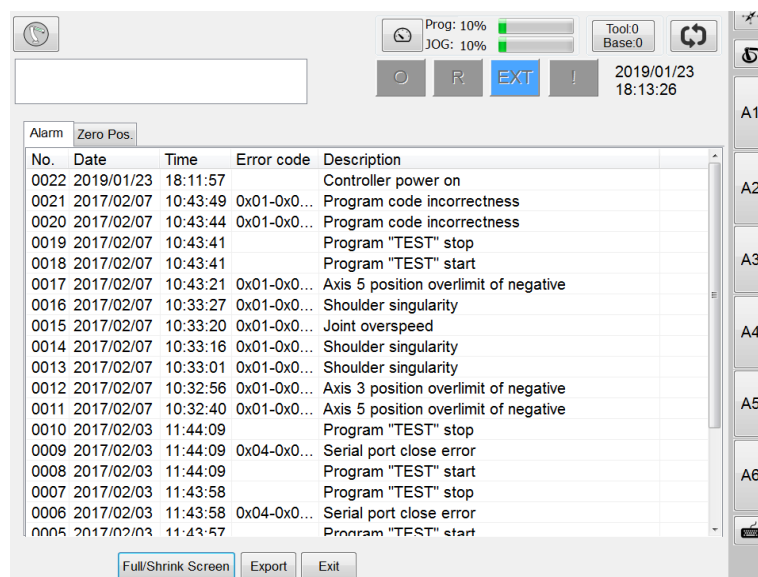


Alarm Message Interface

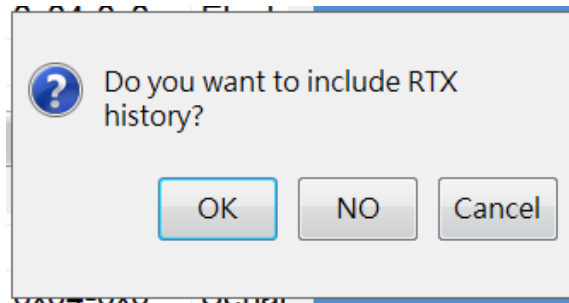


Zero Position Message Interface

Item No.	Description
1	Alarm history message page
2	Zero position history message page
3	Alarm history message
4	Export historical information file
5	Adjust the screen of historical information
6	Zero position history message



Full screen of historical information

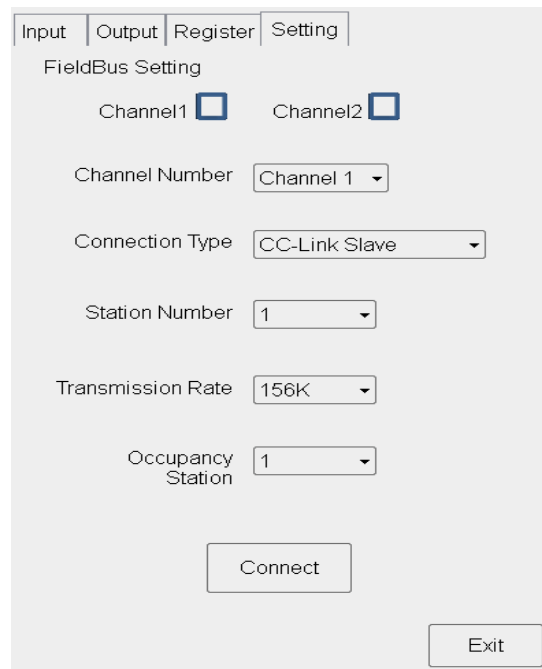


Exports historical messages, select type screen

2.15. Field Bus Setting (Optional)

2.15.1. Set CC-Link Connection Parameters

A. Open HRSS, Click Main Menu → Display → FieldBus → Setting (Opened parameters setting interface)

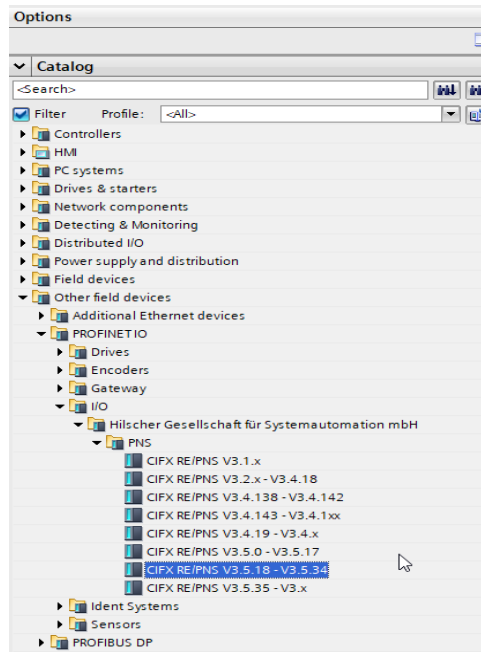


CC-Link connection parameters setting

- B. Channel Number, choose Channel1 or Channel2 for current communication device.
- C. Connection Type, can choose the type of connection, please choose 「 CC-Link Slave 」 .
- D. Station Number, this device can be selected as the station number of the entire CC-Link network, select between 1~64.
- E. Transmission Rate, choose the transfer rate of connection for this device.
- F. Occupancy Station, number of stations (1~4) can be choose from this device.
- G. Once setting is complete, click 「 Connect 」 to connecting, next reboot would use current setting for connection.

2.15.2. Set Profinet Connection Parameters

- A. Uses Siemens PLC to connect with the arm controller; needs to install the PROFINET IO description file GSDML (can be downloaded from the official website of HIWIN). After installation is complete, go to the Options menu in the Siemens TIA Portal development software and select Other field devices->PROFINET IO->I/O->Hilscher Gesellschaft für Systemautomation mbH ->PNS->CIFX RE/PNS V3.5.18–V3.5.34.



- B. Open HRSS, Click Main Menu → Display → FieldBus → Setting (Open parameters setting interface)

Input	Output	Register	Setting
FieldBus Setting			
Channel1	<input type="checkbox"/>	Channel2	<input type="checkbox"/>
Channel Number	Channel 1		
Connection Type	Profinet Slave		
Station Name	station1		
IP Address	192.168.0.1		
Input	1 byte	Output	1 byte
Connect			
Exit			

Profinet connection parameters setting

- C. Channel Number, choose Channel1 or Channel2 for current communication device.
D. Connection Type, can choose the type of connection, please choose 「 Profinet Slave 」 .
E. Station Name, set same name as Master drive.

- F. IP Address, set corresponding to Master drive
- G. Input \ Output, setting of IO quantity should correspond to the setting of Master drive. IO supports up to 16 Byte each.
- H. Once setting is complete, click 「 Connect 」 to connecting, next reboot would use current setting for connection.

2.15.3. Set ModbusTCP Client Connection Parameters

- A. Open HRSS, Click Main Menu → Display → FieldBus → Setting (Open parameters setting interface)

The screenshot shows a software interface for setting ModbusTCP client parameters. At the top, there are four tabs: 'Input', 'Output', 'Register', and 'Setting'. The 'Setting' tab is selected. Below the tabs is a 'FieldBus Setting' section. It contains several controls: two checkboxes for 'Channel1' and 'Channel2' (both are unchecked); a 'Channel Number' dropdown menu set to 'Channel 1'; a 'Connection Type' dropdown menu set to 'Modbus Client TCP'; a 'Remote IP' text field containing '192.168.0.1'; a 'Remote Port' text field containing '502'; three pairs of dropdown menus for 'Input Begin' (0), 'Input Size' (10), 'Output Begin' (0), 'Output Size' (10), and 'Register Begin' (0), 'Register Size' (10). At the bottom of the setting area is a 'Connect' button, and at the bottom right of the entire window is an 'Exit' button.

ModbusTCP Client connection parameters setting

- B. Channel Number, choose Channel1 or Channel2 for current communication device.
- C. Connection Type, can choose the type of connection, please choose 「 Modbus Client TCP 」 .
- D. Server Number, set the number of the remote device.
- E. Remote IP, set corresponding to Server.
- F. Remote Port, set corresponding to Server.
- G. Discrete Input Begin (0~255), Coil Begin (0~255), Input Register Begin (0~999) and Holding Register Begin (0~999), set the starting position of the remote device.
- H. Discrete Input Size, Coil Size, Input Register Size and Holding Register Size sets the data length of the remote device, which corresponds to SI, SO, SRR and SRW, respectively. Data length 0 means disabled.
- I. You can use Modbus Function: 01, 02, 03, 15, 16 to communicate.

- J. Once setting is complete, click 「Connect」 to connecting, next reboot would use current setting for connection.

2.15.4. Set ModbusTCP Server Connection Parameters

A. Open HRSS, Click Main Menu → Display → FieldBus → Setting (Open parameters setting interface)

ModbusTCP Server connection parameters setting

- B. Channel Number, choose Channel1 or Channel2 for current communication device.
- C. Connection Type, can choose the type of connection, please choose 「 Modbus Server TCP 」 .
- D. Local IP1, Local IP2, external devices connected to the local IP.
- E. Local Port, set the external device to connect to the local port.
- F. You can use Modbus Function: 01, 02, 03, 15, 16 to communicate.
- G. Channel1-Input and Output use address: 0 ~ 127, Register use address: 0 ~ 15. Channel2-Input and Output use address: 128 ~ 255, Register use address: 16 ~ 31.
- H. Once setting is complete, click 「 Connect 」 to connecting, next reboot would use current setting for connection.

2.15.5. Set ModbusRTU Client Connection Parameters

A. Open HRSS, Click Main Menu → Display → FieldBus → Setting (Open parameters setting interface)

The screenshot shows a software interface for setting ModbusRTU client connection parameters. It features a tabbed menu at the top with 'Input', 'Output', 'Register', and 'Setting' tabs. The 'Setting' tab is active, displaying 'FieldBus Setting'. There are two checkboxes for 'Channel1' and 'Channel2'. Below them is a 'Channel Number' dropdown menu currently set to 'Channel 1'. The 'Connection Type' is set to 'Modbus Client RTU' in a dropdown menu. Other parameters are set as follows: Slave ID (0), Parity (None), Stop bit (1), Baud rate (9600), Data bits (8), Input Begin (0), Input Size (10), Output Begin (0), Output Size (10), Register Begin (0), and Register Size (10). At the bottom, there are 'Connect' and 'Exit' buttons.

ModbusRTU Client connection parameters setting

- B. Channel Number, choose Channel1 or Channel2 for current communication device.
- C. Connection Type, can choose the type of connection, please choose 「 Modbus Client RTU 」 .
- D. Server Number, set the number of the remote device.
- E. Set the slave ID of the slave station number of the connected device.
- F. Set COMPORT parameters, Parity: Parity, Stop bit: Stop bit, Baud rate: Baud rate, Data bits: Communication data length.
- G. Discrete Input Begin (0~255), Coil Begin (0~255), Input Register Begin (0~999), and Holding Register Begin (0~999), set the starting position of the remote device.
- H. Discrete Input Size, Coil Size, Input Register Size and Holding Register Size sets the data length of the remote device, which corresponds to SI, SO, SRR and SRW, respectively. Data length 0 means disabled.
- I. You can use Modbus Function: 01, 02, 03, 15, 16 to communicate.
- J. Once setting is complete, click 「 Connect 」 to connecting, next reboot would use current setting for connection.

2.15.6. Set ModbusRTU Server Connection Parameters

A. Open HRSS, Click Main Menu → Display → FieldBus → Setting (Open parameters setting interface)

ModbusRTU Server connection parameters setting

- B. Channel Number, choose Channel1 or Channel2 for current communication device.
- C. Connection Type, can choose the type of connection, please choose 「 Modbus Server RTU 」 .
- D. Set the slave ID of the local slave station.
- E. Set COMPORT parameters, Parity: Parity, Stop bit: Stop bit, Baud rate: Baud rate, Data bits: Communication data length.
- F. You can use Modbus Function: 01, 02, 03, 15, 16 to communicate.
- G. Channel1-Input and Output use address: 0 ~ 127, Register use address: 0 ~ 15. Channel2-Input and Output use address: 128 ~ 255, Register use address: 16 ~ 31.
- H. Once setting is complete, click 「 Connect 」 to connecting, next reboot would use current setting for connection.

2.15.7. Sets EtherNet/IP Adapter Connection Parameter

Description

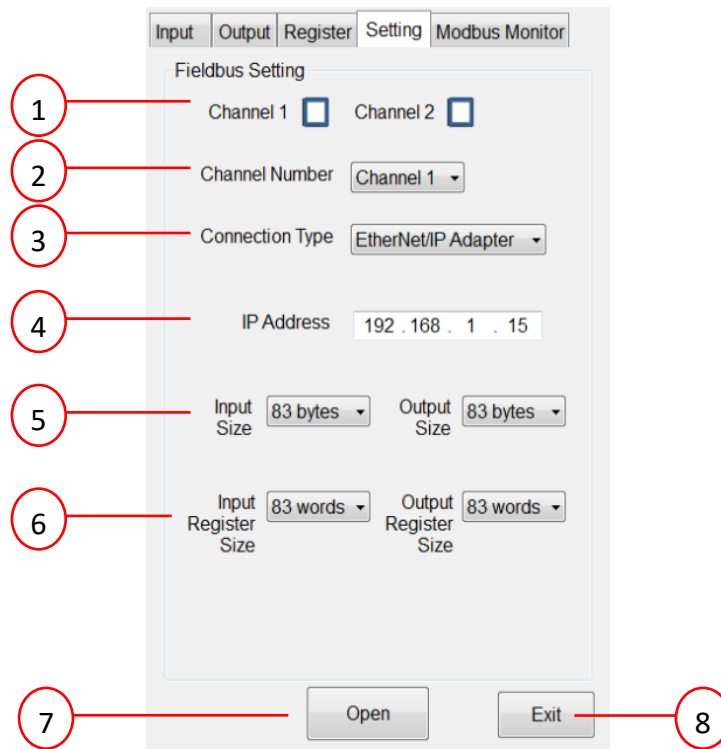
The EtherNet/IP Adapter is the industrial Ethernet communication protocol developed by Rockwell Automation; it is managed by ODVA and can be applied for process control and other automated applications. It is a part of the Common Industrial Protocol (CIP). The IP in the EtherNet/IP name is the abbreviation of “Industrial Protocol.” EtherNet/IP is realized by combining CIP protocol, TCP/IP and Ethernet. It can be seen that network cable is used as the connection method and the transmission method is the CIP protocol and combination. When starting EtherNet/IP communication, one device needs to open the communication line of the connection with respect to the other device. The side that opens the connection is called the scanner, which is similar to the master station; the side that was opened is called the adapter, which is similar to the slave station.

Adding EtherNet\IP Adapter communication into the current Fieldbus function; this is mainly used for hidden message communication (Implicit Message). It provides IO and Register data transmission, which means SI, SO, SRR and SRW data transmissions.

Set the interface to expand the connection types and related parameters of the EtherNet\IP Adapter; HRSS SI and SO will take up 83 bytes, SRR and SRW will take up 166 bytes, therefore, a channel can use a maximum of 664 Inputs/Outputs and 83 Input/Output registers, in which the first 8 of SI and SO are the same with FI and FO. The HRL instruction can use \$SI[n], \$SO[n], \$SRR[n], \$SRW[n] to read and write the IO and register, and SRW can perform the mapping function.

Operating Steps

- 1) After opening the HRSS software and switching to expert mode identity, click Main Menu → Display → Fieldbus → Setting to open the parameter setting interface as shown in the figure below.
- 2) Set the EtherNet\IP Adapter parameters as shown in the chart below.
- 3) When selection is completed, click on “Open” to connect and save settings. Next time it restarts, this setting will be used to perform connection.



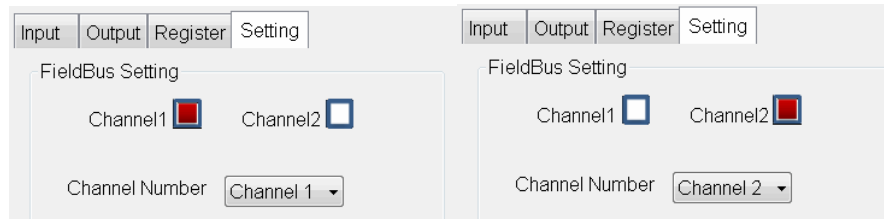
Fieldbus EtherNet/IP Adapter parameter setting interface diagram

FieldBus EtherNet IP parameter description table

NO.	Name	Description
1	Protocol Status Display	Shown channel (protocol) connection situation. If connected successfully, a red block will be displayed; if connection failed or it is not set as enabled, a white block will be displayed.
2	Channel Number	The channel protocol to connect can be selected; select Channel 1 or Channel 2.
3	Connection Type	Select the connection type; please select [EtherNet/IP Adapter].
4	IP Address	The IP address of the IP Adapter must be set.
5	Input/Output Size	Input/output length must be set; allowed range is 0~83 Bytes.
6	Input/Output Register Size	Input/output register length must be set; allowed range is 0~83 Words.
7	Open/Close	Perform connect (Open) or disconnect (Close).
8	Exit	You will leave this setting interface and return to the previous level interface.

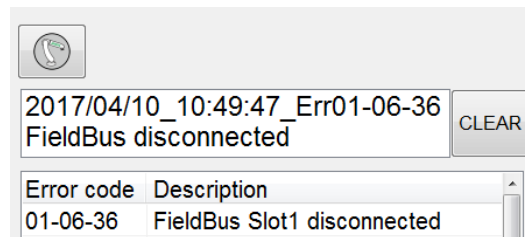
2.15.8. Confirmation of Connection Status

- A. Start-up connection success or fail: in the Setting page, signal will be shown, if connection is successful, the box will appear red, if connection is failed or setting is not switched on, the box will appear white.



Situation where Channel1 \ Channel2 appear red

- B. If the setting is switched on and disconnection occur due to wrong parameters setting, connection abnormal or etc. A warning signal will appear.



FieldBus disconnection error

2.15.9. Use FieldBus Input (SI[n])

- A. According to the Slave Number used and Occupancy, the number of Input Number that can be controlled can be different.
- a. When using Slave1, SI[1]~SI[128] can be used.
 - i. When Occupancy is 1, SI[1]~SI[32] can be used.
 - ii. When Occupancy is 2, SI[1]~SI[64] and so on.
 - b. When using Slave2, SI[129]~SI[256].
 - i. When Occupancy is 1, SI[129]~SI[160].
 - ii. When Occupancy is 2, SI[129]~SI[192] and so on.
- B. SI[1]~SI[8] are reserved and have similar function as FI[1]~FI[8].
- C. Interface can be used directly for selection
- a. Main menu → Display → FieldBus → Input
 - b. When logging into EXPERT user group, click SIM. (Simulation function) and test the SI function.
 - c. Comment for SI[1]~SI[8] is unmodifiable, the rest will be stored.
- D. Can be controlled by command.
- a. The command \$SI[n] can be used to read Input status
 - b. Other instruction can be used such as IF \ WAIT FOR

```

Code
1 WAIT FOR $SI[10] == TRUE
2 WAIT SEC 1
3
4 IF $SI[11] == FALSE THEN
5 WAIT SEC 1
6 |

```

Motion Function Configure Program Edit Exit

Code modify SI[n]

2.15.10. Use FieldBus Output (SO[n])

- A. According to the Slave Number used and Occupancy, the number of Output Number that can be controlled can be different.
 - a. When using Slave1, SO[1]~SO[128] can be used.
 - i. When Occupancy is 1, SO[1]~SO[32] can be used.
 - ii. When Occupancy is 2, SO[1]~SO[64] and so on.
 - b. When using Slave2, SO[129]~SO[256].
 - i. When Occupancy is 1, SO[129]~SO[160].
 - ii. When Occupancy is 2, SO[129]~SO[192] and so on.
- B. SO[1]~SO[8] are reserved and have similar function as FO[1]~FO[8].
- C. Interface can be used directly for selection
 - a. Main menu → Display → FieldBus → Output
 - b. When logging into EXPERT user group, click Value directly to change the Output status and test SO function.
 - c. SO[1]~SO[8] not allow to set Value status directly °
 - d. Comment for SO[1]~SO[8] is unmodifiable, the rest will be stored.
- D. Can be controlled by comment
 - a. Command \$SO[n] can be used to set Output status.

```

Code
1 $SO[10] = TRUE
2 $SO[11] = FLASE
3 |

```

Motion Function Configure Program Edit Exit

Code modify SO[n]

2.15.11. Use FieldBus Register (SRR 、 SRW)

- A. SRR (Register for Read) and SRW (Register for Write) ◦
- B. When using Slave1, SRR[1]~SRR[16] 、 SRW[1]~SRW[16]
 - a. When Occupancy is 1, SRR[1]~SRR[4] 、 SRW[1]~SRW[4]
 - b. When Occupancy is 2, SRR[1]~SRR[8] 、 SRW[1]~SRW[8] so on.
- C. When using Slave2, SRR[17]~SRR[32] 、 SRW[17]~SRW[32]
 - a. When Occupancy is 1, SRR[17]~SRR[20] 、 SRW[17]~SRW[20]
 - b. When Occupancy is 2, SRR[17]~SRR[24] 、 SRW[17]~SRW[24] so on.
- D. Values can be written or read directly through interface.
 - a. Main menu → Display → FieldBus → Register ◦
 - b. SRR column can be read but not modify.
 - c. SRW column can modify when clicked.
 - d. Range of value input 32767~-32767.
 - e. Comment after modified will be saved.
- E. Can be controlled by command
 - a. Command \$SRW[n] used to set SRW status.
 - b. Command \$SRR[n] used to set SRR status.

```

Code
1 $SRW[1] = 1
2 $SRW[10] = -100
3
4 IF $SRR[15] == 15 THEN
5 WAIT SEC 1
6
7 WAIT FOR $SRR[20] == 20
8 WAIT SEC 1
9
10

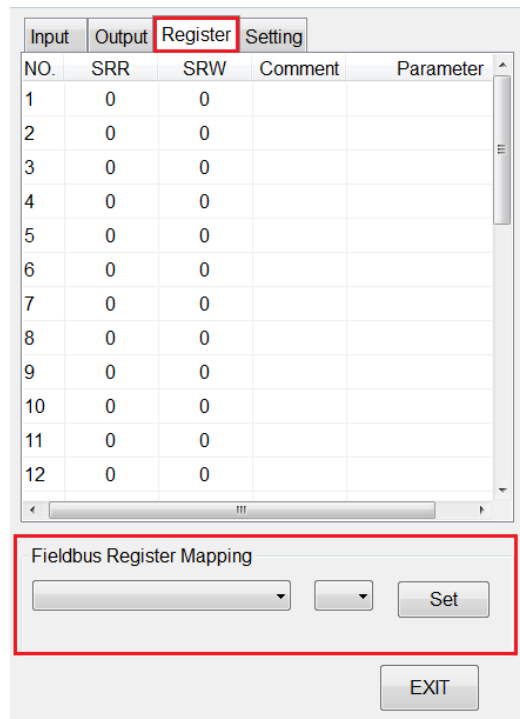
```

Motion Function Configure Program Edit Exit

Code modify SRW[n] 、 SRR[n]

2.15.12. Use FieldBus Register Mapping

- A. Open HRSS, click Main Menu → Display → Fieldbus → Register (open Fieldbus Register Mapping setting interface)



Fieldbus Register Mapping setting interface

- B. Select system parameters that user required

Parameters name

A1_ACTUAL: Actual angle of 1st axis

A2_ACTUAL: Actual angle of 2nd axis

A3_ACTUAL: Actual angle of 3rd axis

A4_ACTUAL: Actual angle of 4th axis

A5_ACTUAL: Actual angle of 5th axis

A6_ACTUAL: Actual angle of 6th axis

X_ACTUAL: Actual X coordinate of TCP

Y_ACTUAL: Actual Y coordinate of TCP

Z_ACTUAL: Actual Z coordinate of TCP

A_ACTUAL: Actual A coordinate of TCP

B_ACTUAL: Actual B coordinate of TCP

C_ACTUAL: Actual C coordinate of TCP

A1_TORQUE: The torque value of the first axis (the unit is per thousand percent)

A2_TORQUE: The torque value of the second axis (the unit is per thousand percent)

A3_TORQUE: The torque value of the third axis (the unit is per thousand percent)

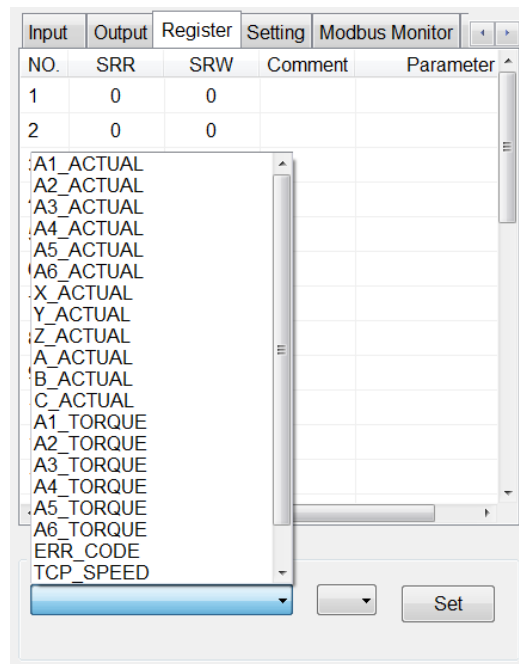
A4_TORQUE: The torque value of the fourth axis (the unit is per thousand percent)

A5_TORQUE: The torque value of the fifth axis (the unit is per thousand percent)

A6_TORQUE: The torque value of the sixth axis (the unit is per thousand percent)

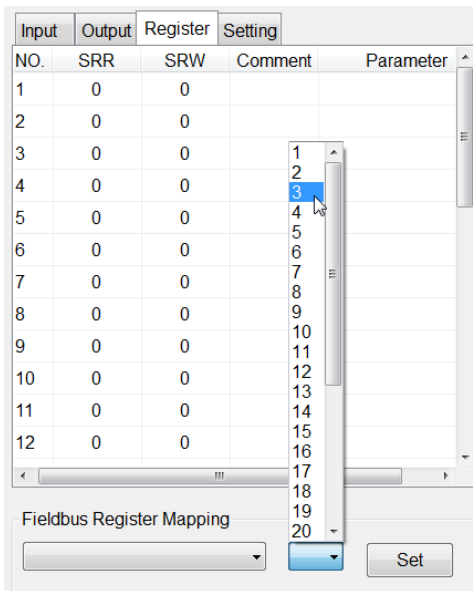
ERR_CODE: Error code

TCP_SPEED: Actual speed of TCP



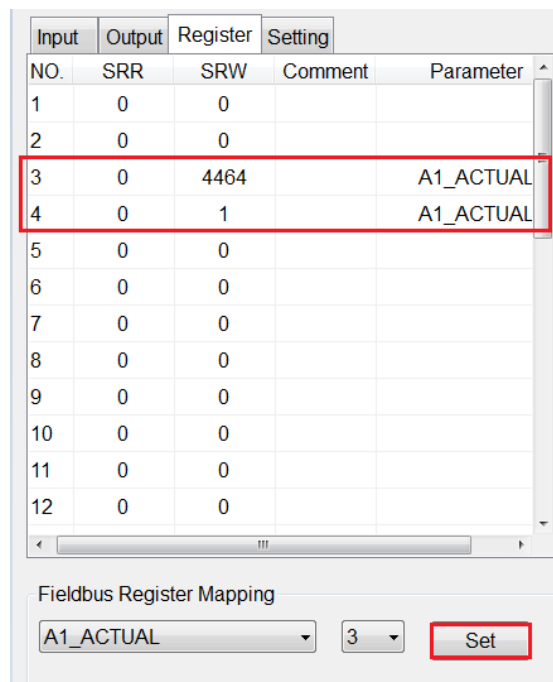
Selection of System Parameters interface

C. Select the Register Number that user wanted to store the parameter into



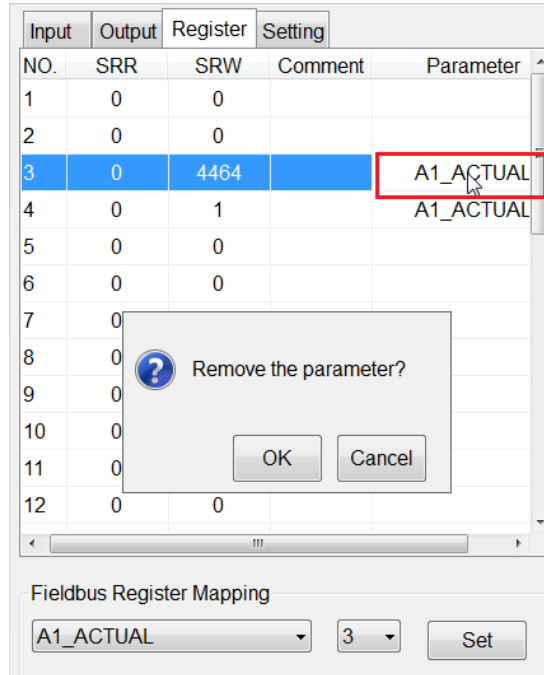
Register Number storage for parameter interface

D. Click set to save the setting.



Set button to store setting interface

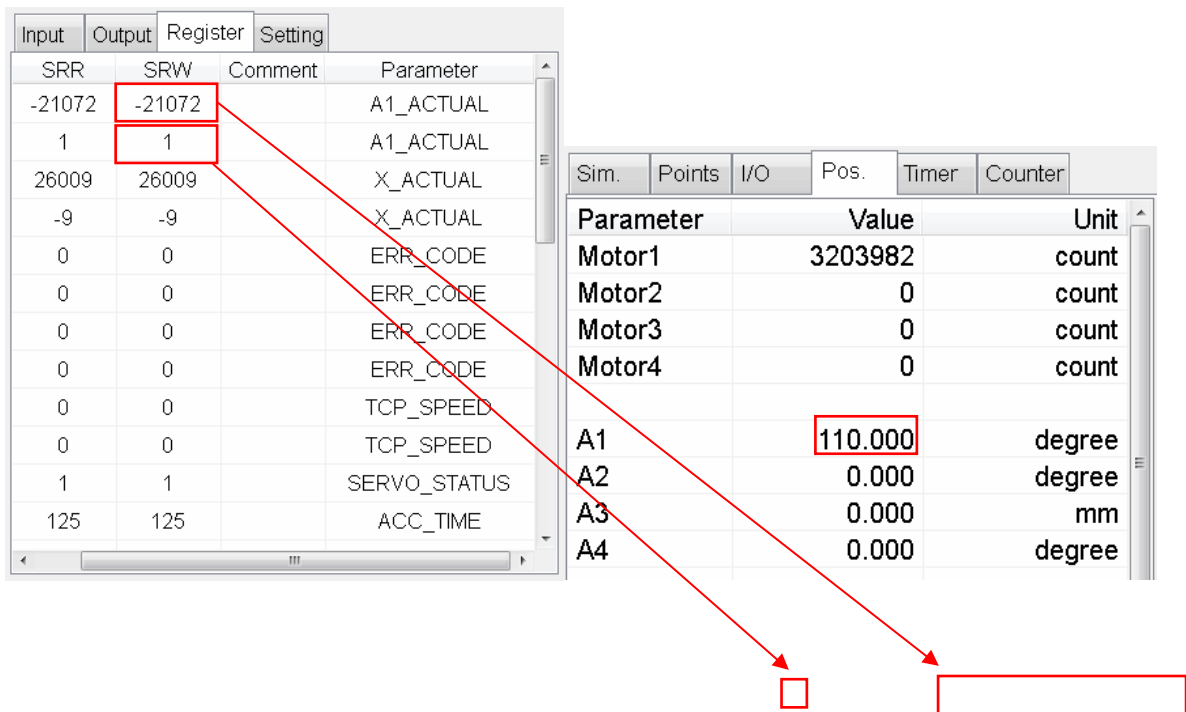
E. Click parameter column to remove the stored parameter from Register.



Remove parameter from Register

F. How to transfer parameter:

A1_ACTUAL:



Input	Output	Register	Setting
SRR	SRW	Comment	Parameter
20536	20536		A1_ACTUAL
-1	-1		A1_ACTUAL
31048	31048		X_ACTUAL
6	6		X_ACTUAL
0	0		ERR_CODE
0	0		ERR_CODE
0	0		ERR_CODE
0	0		ERR_CODE
0	0		TCP_SPEED
0	0		TCP_SPEED
1	1		SERVO_STATUS
125	125		ACC_TIME

Sim.	Points	I/O	Pos.	Timer	Counter
Parameter			Value	Unit	
Motor1			-1310720		count
Motor2			0		count
Motor3			0		count
Motor4			0		count
A1			-45.000		degree
A2			0.000		degree
A3			0.000		mm
A4			0.000		degree

ERR_CODE:

2019/10/29_17:28:49_Err03-03-41
A3 error - S-2220

CLEAR

Input	Output	Register	Setting
SRR	SRW	Comment	Parameter
0	0		A1_ACTUAL
0	4437		A1_ACTUAL
0	1		A1_ACTUAL
0	8736		ERR_CODE
0	0		ERR_CODE
0	833		ERR_CODE
0	3		ERR_CODE
0	0		
0	0		
0	0		

2020/07/28_15:17:54_Err03-E1-FF
E1 error

CLEAR

Input	Output	Register	Setting
SRR	SRW	Comment	Parameter
0	0		A1_ACTUAL
0	0		A1_ACTUAL
0	0		X_ACTUAL
0	0		X_ACTUAL
0	0		ERR_CODE
0	0		ERR_CODE
-7681	-7681		ERR_CODE
3	3		ERR_CODE
0	0		TCP_SPEED
0	0		TCP_SPEED
0	0		SERVO_STATUS
125	125		ACC_TIME

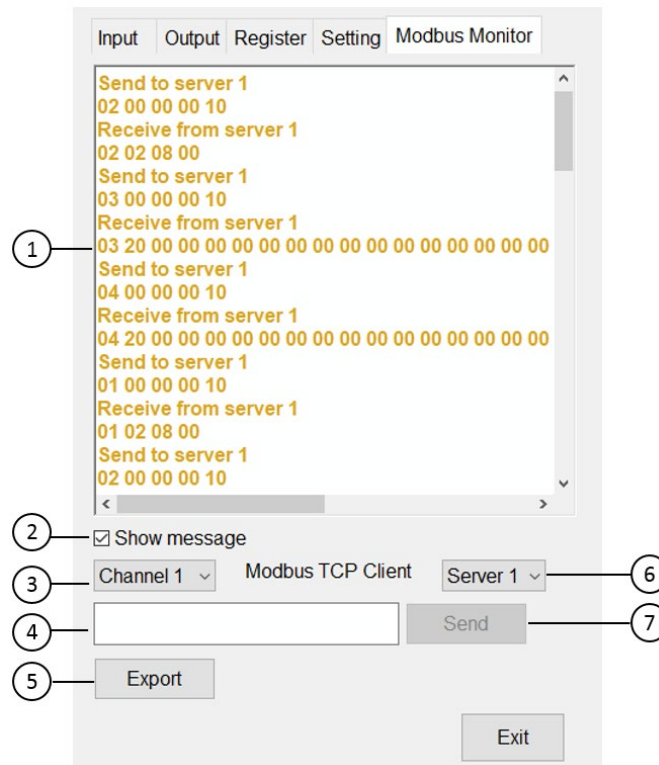
2.15.13. Modbus Monitor Page

Description

This interface is used to display Modbus communication messages.

Operating Steps

Main Menu > Display > Fieldbus > Modbus Monitor

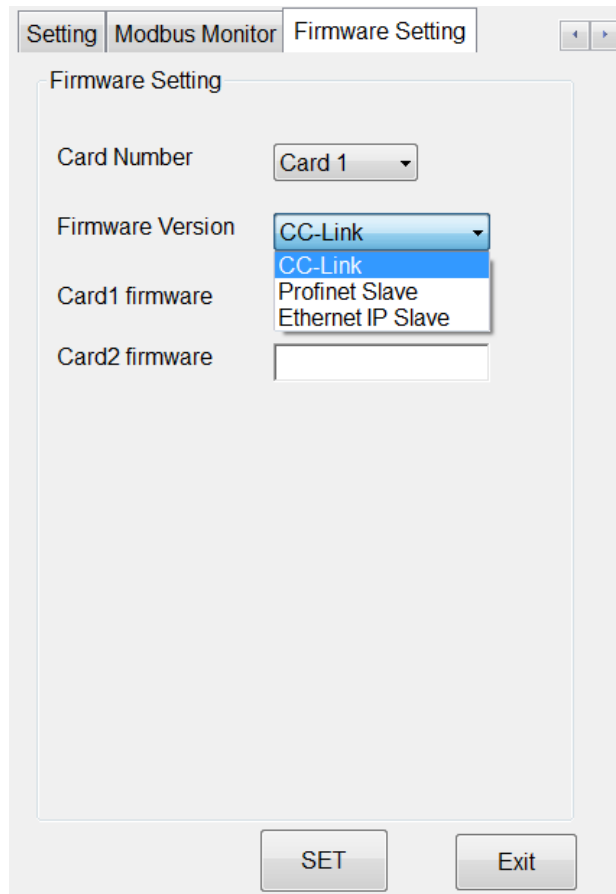


Modbus Monitor interface diagram (Modbus TCP Client is an example)

Modbus monitor setting description table

No.	Name	Description
1	Modbus message display	Modbus monitoring message display window.
2	Message display selection	Check whether the display/hide transmission messages.
3	Channel selection	Select Channel; it is a drop-down menu. Channel 1 or Channel 2 can be selected.
4	Enter the message to send	Enter the data to send to the server end.
5	Export button	Requires connecting a USB hard drive to output the message window to the USB hard drive.
6	Corresponding server number	The message sending server number selected for client mode.
7	Send message button	Users enter the message to send into the message field at the front, then click on the message send button for manual message send test.

2.15.14. Fieldbus Firmware Setting



After selecting the card number to install, it can be replaced according to the firmware version selected; the controller will restart twice when changing firmware, do not disconnect the power while setting.
Note: Only the CC-Link firmware can be installed on the CC-Link card; the firmware is interchangeable between Profinet and Ethernet IP

2.16. Communication

2.16.1. TCP/IP Communication

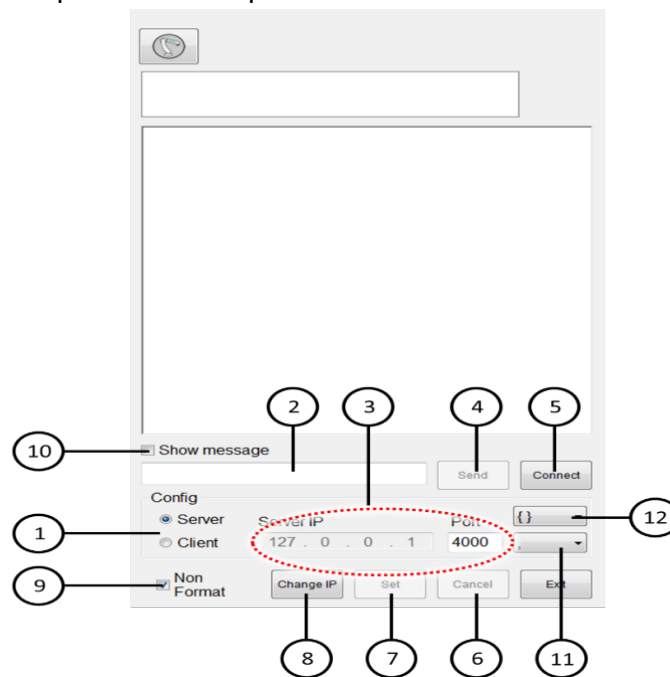
Description

Send and transfer the data by network communication.

You can select RC as Client or Server to connect.

The parameter type is the floating decimal.

The communication format has two parentheses (The type of parenthesis can be selected.), including the value form such “{xxx}” For example, if “{123,456}” is sent, two sets of value “123” and “456” will be received, which there are up to 50 sets of parameter.



TCP/IP interface

No.	Description
1	Server/Client configuration
2	Message sending field
3	IP and Port configuration
4	Send message
5	Connect/Disconnect button
6	Cancel
7	Set
8	Change IP
9	Cancel format
10	Display Content of Sending Message and Receiving Message
11	Division symbol

No.	Description
12	Parenthesis type

Operation steps

Main Menu>Start-up >Network Config

- Client
 1. Enter the Server’s IP and Port
 2. Press [Connect].
 3. Display “Connection is successful!” to represent the connection success.

- Server
 1. Enter the port you want to connect.
 2. Press [Connect].
 3. Display “Server is opened!” to represent opened.

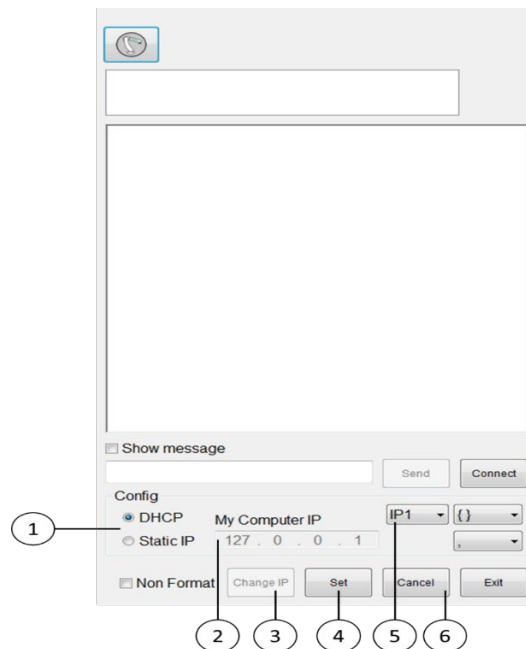
2.16.2. Setting connection for IP address

Description

Set IP address for robot controller.

Robot controller consist of two internet port, they are : Port 1 and Port 2.

User can choose to change IP address for either Port 1 or Port 2, using DHCP mode (automatic obtain IP address) or Static mode (specify specific IP address).



Change IP interface

No.	Description
1	DHCP / Static IP mode selection

No.	Description
2	Static IP, specific IP address
3	Enter Change IP interface
4	Confirm setting
5	Select to change Port 1/ Port2 IP address
6	Cancel setting

Operation Steps

Main menu >Start-up >Network Config>Change IP

- DHCP
 1. Click [DHCP] option.
 2. Press [Set] button.
 3. Wait for the bar to finish loading, setting is completed.
- Static IP
 - A. Click [Static IP] option.
 - B. In [My Computer IP] column enter required IP address.
 - C. Press [Set] button.
 - D. Wait for the bar to finish loading, setting is completed.

If setting failed message appeared, please check the internet connection to see if it is connected properly or there is a problem in IP setting.

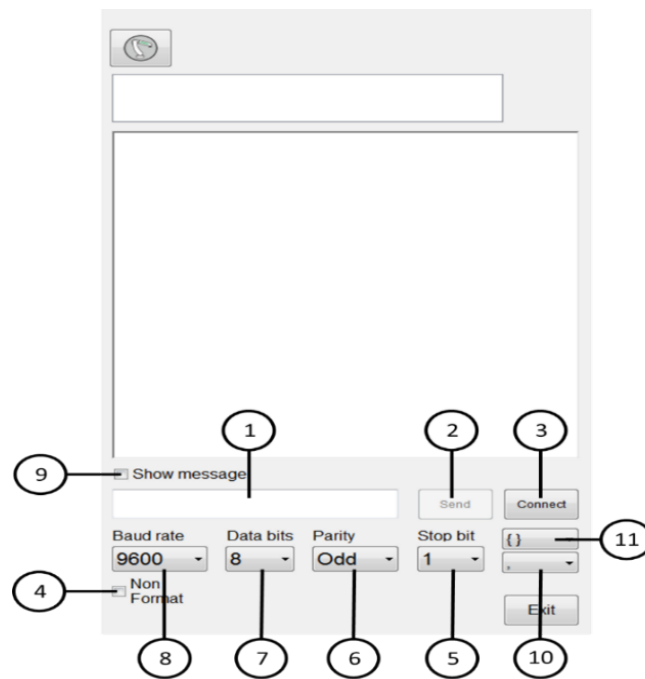
2.16.3. RS232 Communication

Description

Send and transfer the data by serial communication.

The parameter type is the floating decimal.

The communication format has two parentheses (The type of parenthesis can be selected.), including the value form such "{xxx}" For example, if "{123,456}" is sent, two sets of value "123" and "456" will be received, which there are up to 50 sets of parameter.



RS232 interface

No.	Description
1	Message sending field
2	Send message
3	Connect/Disconnect button
4	Cancel format
5	RS232 Stop bit
6	RS232 Parity
7	RS232 Data bit
8	RS232 Baud rate
9	Display Content of Transmitting Message & Receiving Message
10	Division symbol
11	Parenthesis type

Operation steps

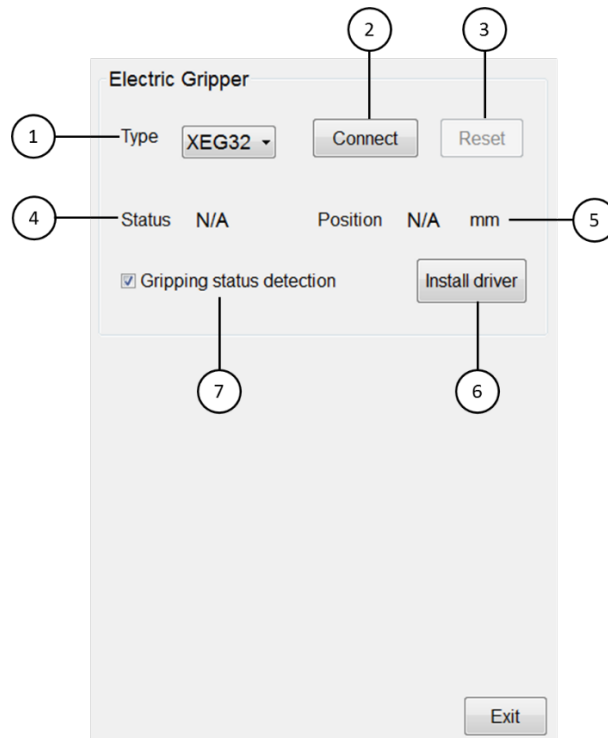
Main Menu>Start-up>RS-232

1. Enter RS232 parameters.
2. Press [Connect].
3. Display "Connection is successful!" to represent the connection success.

2.17. Electric Gripper Setting

Description

User can use HRSS to set the series of electric gripper XEG and IEG's connection and reset it to allow user to manually move the gripper and use HRL relevant command to control under T1 mode.



Electric Gripper operation interface

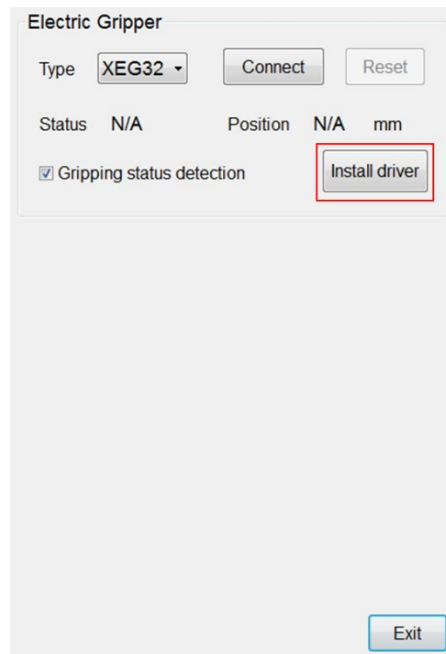
No.	Description
1	Set the model of Gripper
2	Connect/Disconnect with the Gripper
3	Rest Gripper
4	Display current Gripper status
5	Display current Gripper position
6	Install Gripper driver
7	Gripper status detection function

Operation Steps

Main menu>Start-up>Electric Gripper

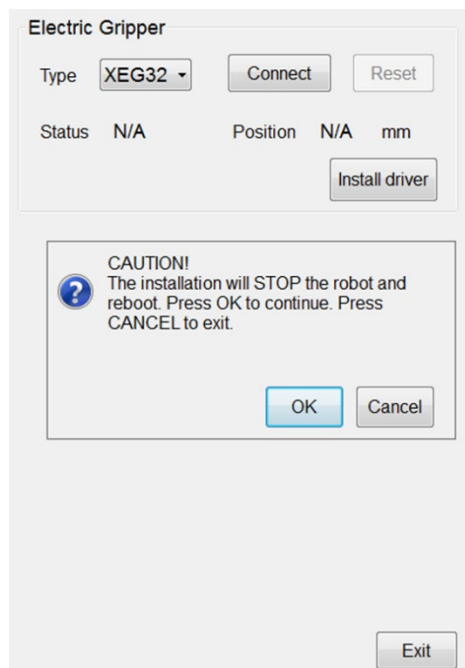
1. If driver not installed, driver needed to be installed first, installation steps are as follows:
 - A. HRSS software version has to be updated to 3.2.5 and above.
 - B. Download the Electric Gripper Driver.exe from official website and save the file in USB under HIWIN folder. Then insert the USB into controller.

- C. In the main screen of HRSS, select Start-up->Electric Gripper, once Electric Gripper interface is opened, click the install driver button.



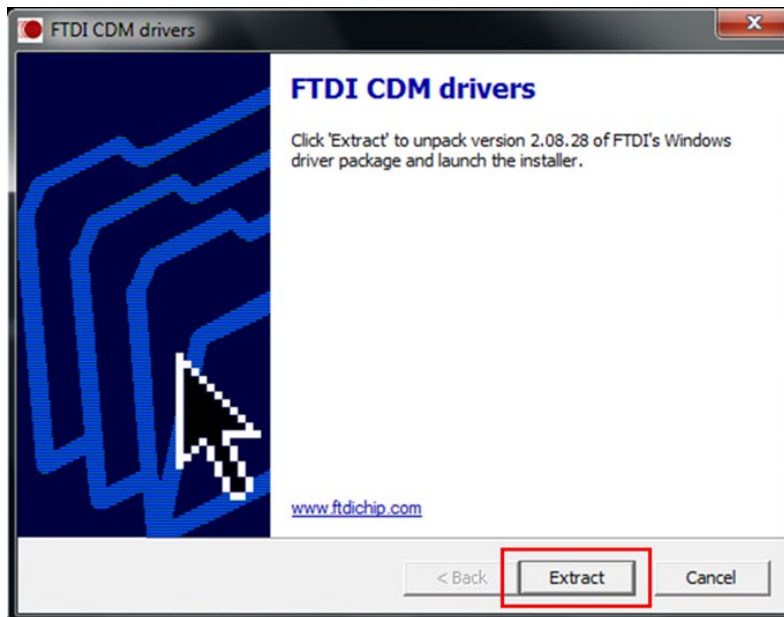
Electric Gripper operation interface

- D. If HRSS detected the driver installation file from the USB, a warning signal will appear to inform user that installation will stop the robot and reboot. Press OK to start driver installation; Press CANCEL to exit.



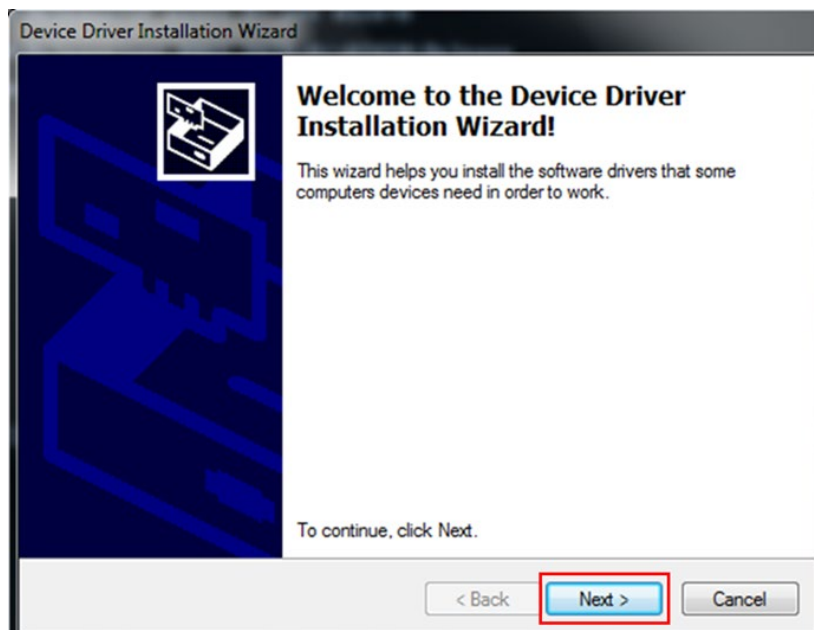
Install Electric Gripper warning

- E. Press OK to end HRSS and start rebooting. Once reboot completed, the driver installation will appear, click Extract.



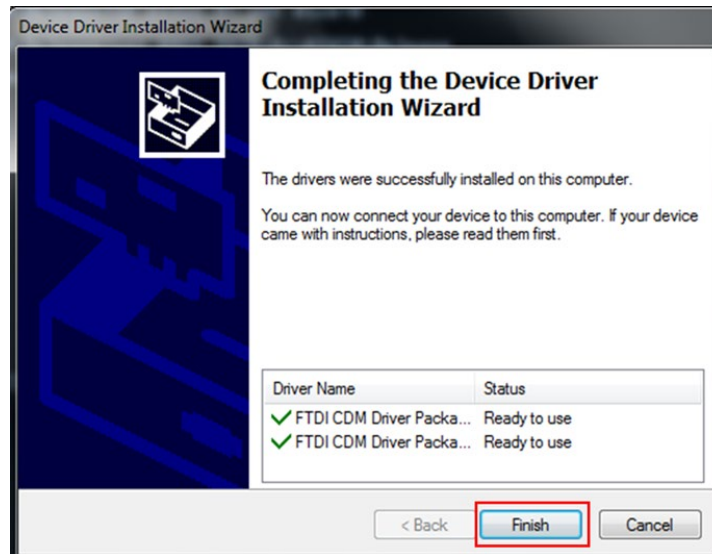
Electric Gripper driver installation step 1

F. Click Next to continue



Electric Gripper driver installation step 2

- G. Click Finish, the system will operate according to different anti-write system. If FBWF anti-write system is set, the system will start rebooting and open HRSS. If EWF anti-system is set, the system will open HRSS once the driver installation is completed.



Electric Gripper driver installation step 3



Installation completed, enter HRSS

3. Select type of Gripper to connect
4. Click Connect button to start connection
5. Click Reset button to reset Gripper position, the gripper is operated in T1 mode with enabling switch pressed, please beware that the status of the gripper turned from Busy to Idle indicated that the reset is completed.
6. The gripper detection function is preset to be on. This function will notify that the gripper is not gripped to any object. If user does not need the detection, the function can be unchecked.

2.18. Module I/O Function

- Description

User can set module I/O functions in HRSS, allowing user to monitor several I/Os simultaneously.

- Operation steps

Main Menu>Display>Variable>Module I/O

1. Module Input setting is shown in the figure below:

Start is the starting Input number, and End is the ending Input number. When Input is set to On in the module, the Module Input is displayed as On.

MI	MO						Sim.	Points	I/O	Pos.	Timer	Coun
	SIM.	Value	Type	Start	End		NO.	SIM.	Value	Comme		
I20	<input type="checkbox"/>	<input checked="" type="checkbox"/> On	DI	1	5		DI1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> On			
I21	<input type="checkbox"/>	<input type="checkbox"/> Off	DI	0	0		DI2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> On			
I22	<input type="checkbox"/>	<input type="checkbox"/> Off	DI	0	0		DI3	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> On			
I23	<input type="checkbox"/>	<input type="checkbox"/> Off	DI	0	0		DI4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> On			
I24	<input type="checkbox"/>	<input type="checkbox"/> Off	DI	0	0		DI5	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> On			
I25	<input type="checkbox"/>	<input type="checkbox"/> Off	DI	0	0		DI6	<input type="checkbox"/>	<input type="checkbox"/> Off			
I26	<input type="checkbox"/>	<input type="checkbox"/> Off	DI	0	0		DI7	<input type="checkbox"/>	<input type="checkbox"/> Off			
I27	<input type="checkbox"/>	<input type="checkbox"/> Off	DI	0	0		DI8	<input type="checkbox"/>	<input type="checkbox"/> Off			
I28	<input type="checkbox"/>	<input type="checkbox"/> Off	DI	0	0		DI9	<input type="checkbox"/>	<input type="checkbox"/> Off			
I29	<input type="checkbox"/>	<input type="checkbox"/> Off	DI	0	0		DI10	<input type="checkbox"/>	<input type="checkbox"/> Off			
I30	<input type="checkbox"/>	<input type="checkbox"/> Off	DI	0	0		DI11	<input type="checkbox"/>	<input type="checkbox"/> Off			
I31	<input type="checkbox"/>	<input type="checkbox"/> Off	DI	0	0		DI12	<input type="checkbox"/>	<input type="checkbox"/> Off			
I32	<input type="checkbox"/>	<input type="checkbox"/> Off	DI	0	0		DI13	<input type="checkbox"/>	<input type="checkbox"/> Off			
							DI14	<input type="checkbox"/>	<input type="checkbox"/> Off			

Exit D.I. D.O. R.I. R.O. F.I. F.O.

Module Input

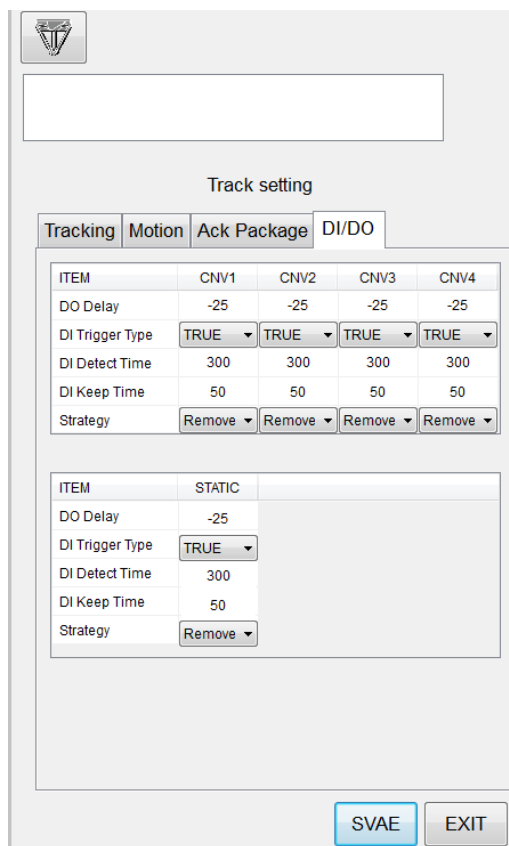
2. The module Output setting interface is shown below:
Start is the starting Output number, End is the ending Output number. When the module Output is On after setting, all Outputs in the module are displayed as On.

MI	MO	Sim.	Points	I/O	Pos.	Timer
NO.	Value	Type	Start	End	NO.	Value
MO1	<input checked="" type="checkbox"/> On	DO	1	5	DO1	<input checked="" type="checkbox"/> On
MO2	<input type="checkbox"/> Off	DO	0	0	DO2	<input checked="" type="checkbox"/> On
MO3	<input type="checkbox"/> Off	DO	0	0	DO3	<input checked="" type="checkbox"/> On
MO4	<input type="checkbox"/> Off	DO	0	0	DO4	<input checked="" type="checkbox"/> On
MO5	<input type="checkbox"/> Off	DO	0	0	DO5	<input checked="" type="checkbox"/> On
MO6	<input type="checkbox"/> Off	DO	0	0	DO6	<input type="checkbox"/> Off
MO7	<input type="checkbox"/> Off	DO	0	0	DO7	<input type="checkbox"/> Off
MO8	<input type="checkbox"/> Off	DO	0	0	DO8	<input type="checkbox"/> Off
MO9	<input type="checkbox"/> Off	DO	0	0	DO9	<input type="checkbox"/> Off
MO10	<input type="checkbox"/> Off	DO	0	0	DO10	<input type="checkbox"/> Off
MO11	<input type="checkbox"/> Off	DO	0	0	DO11	<input type="checkbox"/> Off
MO12	<input type="checkbox"/> Off	DO	0	0	DO12	<input type="checkbox"/> Off
MO13	<input type="checkbox"/> Off	DO	0	0	DO13	<input type="checkbox"/> Off
					DO14	<input type="checkbox"/> Off

Module Output

2.19. Pick/Place DI Detection Function

- Description
DI detection function can be used to prevent object from falling off during picking and placing when operating conveyor belt pick and place.
- Operation Steps
Main Menu>Track>Setting
- The interface is as follows and four parameters can be set: DI Trigger Type 、 DI Detect Time 、 DI Keep Time 、 Strategy
 1. DI Trigger Type: The state when manipulator picks an object.
 2. DI Detect Time: After DO signal is sent, the time of DI detection is maintained.
 3. DI Keep Time: Detecting the duration of DI, when over the duration, determine the detect DI.
 4. Strategy: During pick, response strategy DI is detected



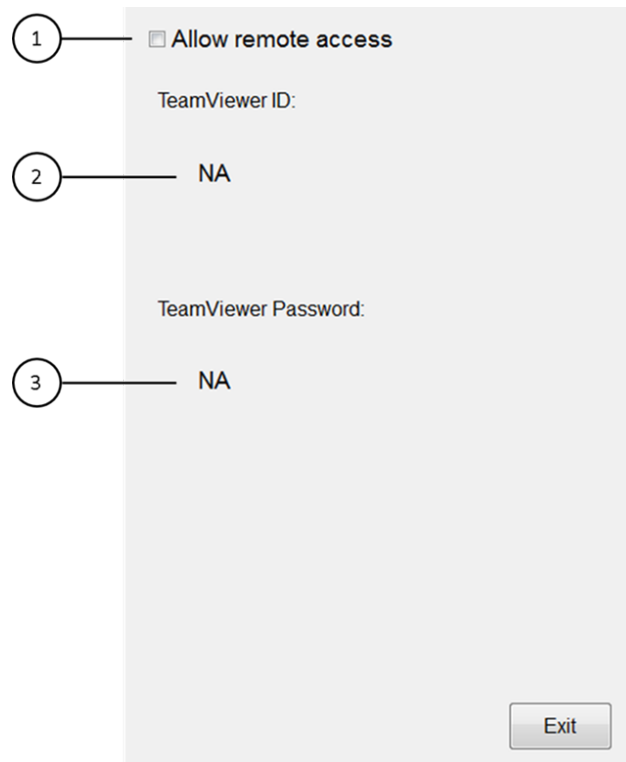
Module Output

2.20. Remote Monitoring Function

● Description

HRSS uses TeamViewer software to achieve remote monitoring requirements, allowing customer service to conduct operation teaching, error diagnosis and remote control, etc. User can open TeamViewer in HRSS interface and obtain TeamViewer ID and password. The function needs following requirements:

1. The robot controller needs to be able to connect to network.
2. The TeamViewer software has been installed automatically on the robot controller for HRSS software version 3.2.8 or higher. If the version is updated in 3.2.7 or before, manually installation is required.



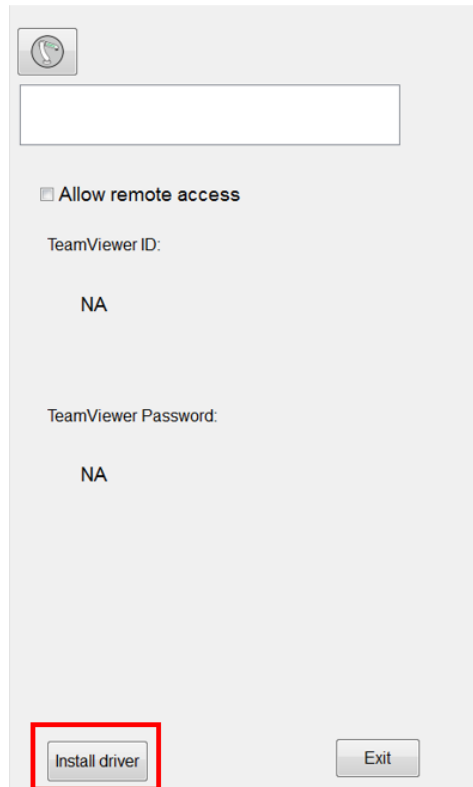
Remote monitor setting interface

No.	Description
1	Allow remote monitoring of robot controller
2	Display TeamViewer ID
3	Display TeamViewer password

● Operation Step

Main Menu> Start-up>System Setting>Remote Access

1. Ensure robot controller is connected to network
2. Ensure TeamViewer is installed on the robot controller, HRSS version 3.2.8 and above is already installed automatically, if user has version 3.2.7 or earlier, please install the software manually, the manual installation steps is as follows:
 - A. The HRSS software version need to be updated to 3.2.8 or higher.
 - B. Please go to the official website to download the Remote Desktop Driver.exe file, place the file in the HIWIN folder in the USB device, and insert the USB device into the controller.
 - C. Select Start-up>System Setting>Next>Remote Access in the HRSS main menu to open the remote monitoring function interface.
 - D. Click the Install Drive button.



Remote monitoring operation interface

- E. If HRSS detects the driver installation file in USB, it will pop up a warning window to remind the user to stop the robot. When the driver is installed and reboot several times, pressing OK will start the driver installation automatically, pressing Cancel will cancel the installation.



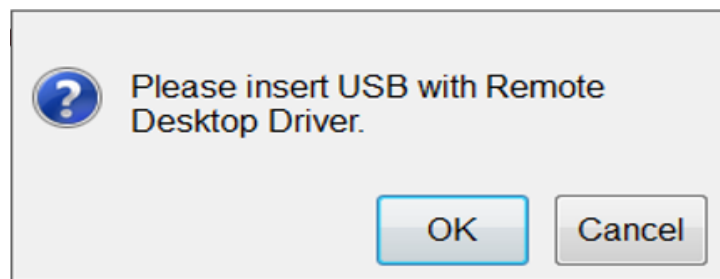
Installation remote monitor warning

- F. After pressing OK, HRSS will automatically shut down and reboot, and automatically start the driver installation. After the program is installed, it will automatically restart again, and open HRSS to complete the installation process.



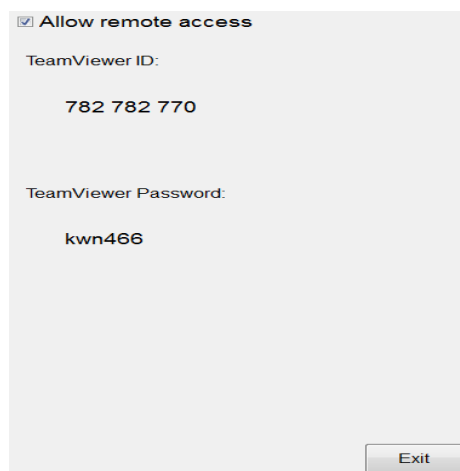
Installation complete automatically enter HRSS

- G. If the driver installation file is not placed in the USB device, or the folder path name is incorrect, it will cause failure in installation and the warning window will pop up.



Pick file failure warning

- H. Tick allow remote access, after TeamViewer successfully opened and the ID and password will be displayed on HRSS shown in figure below. If TeamViewer install incorrectly, the open error will appear, show in figure below.



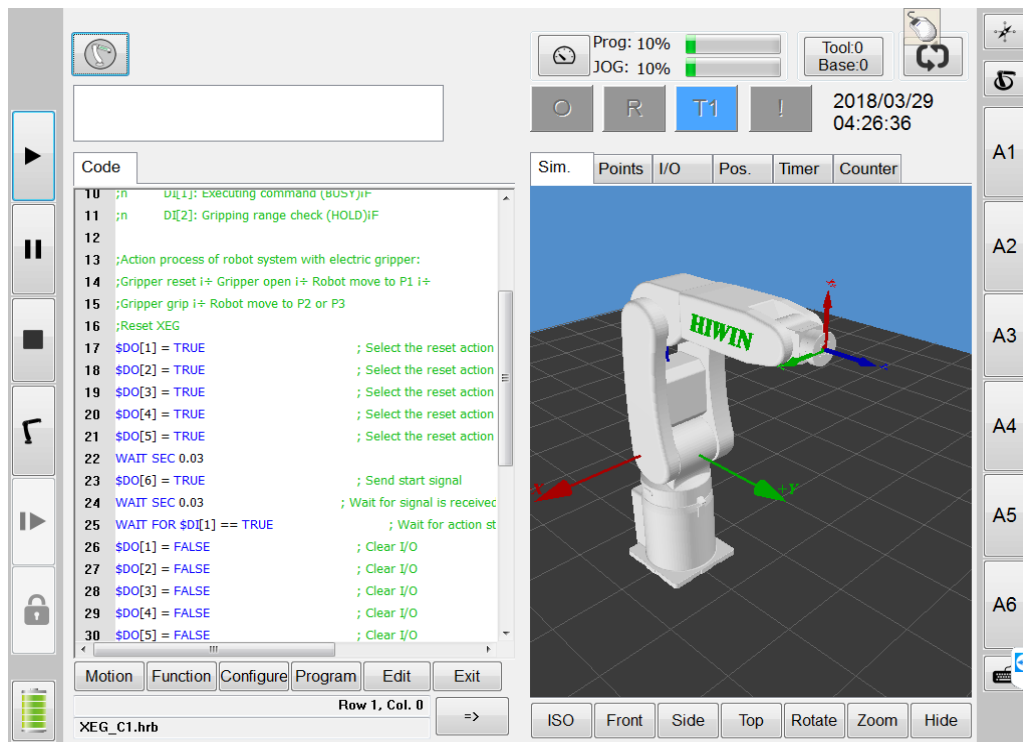
Successfully open TeamViewer

2018/03/20_19:13:08_Err01-01-56
TeamViewer open error

CLEAR

TeamViewer open error warning

- I. Inform the sales about the remote ID of the ID and password displayed on the interface of the robot.
- J. After the customer service has successfully connected, the robot controller screen can be obtained and the manipulator can remotely operated.

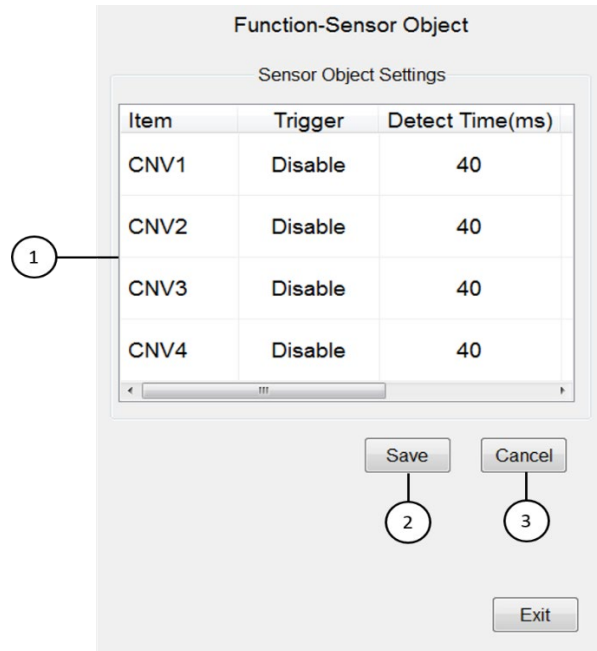


Connection successful, obtained robot control screen

2.21. Conveyor Belt Tracking Sensor Trigger Object Identification Function

- Description

In the belt tracking, if the trigger source is selected to use the sensor trigger (Sensor Latch), the corresponding input detection signal when the object is triggered can be set in the specified conveyor belt in the HRSS. If the input detection signal reaches the set time (Keep Time) in the Detec Time set after the object is triggered, the object is identified as the desired type, and generally can be applied to the good product detection, and used the command CNV_OBJECT to perform different program processing.



Sensor trigger object identification function

No.	Description
1	Set the object recognition trigger signal, detection time, duration of the specified conveyor belt.
2	Save setting
3	Cancel setting

- Operation steps

Main Menu>Track>Sensor Object

1. Conveyor belt related setting calibration is completed, and sensor trigger is selected.
2. Set the selection of the conveyor object identification input trigger source.
3. Set the input signal detection time for the selected conveyor object identification.
4. Set the input signal duration for the selected conveyor object identification.
5. The program uses the value of CNV_OBJECT after the CNV_PICK instruction for subsequent programming planning

2.22. External Axis

- Description

HRSS provides external axis operation and control functions and robot language (HRL) commands. After the user-related hardware devices are properly configured, you can use the functions related to external axes in HRSS, such as controlling external axes to perform synchronization or asynchronous motion control or synchronous coordinated control of external axes. HRSS can control up to 3 external axes, and can set up a set of 2-axis positioner kinematics system. In application, it can control linear moving stage, end tool and positioner., etc.

2.22.1. External axis parameter settings

- Description

After the hardware has been connected to the external axis to be used and HRSS is turned on, you must set parameters for the detected external axis before operating the external axis. The parameters of each axis to be operated are set and saved before setting the group.

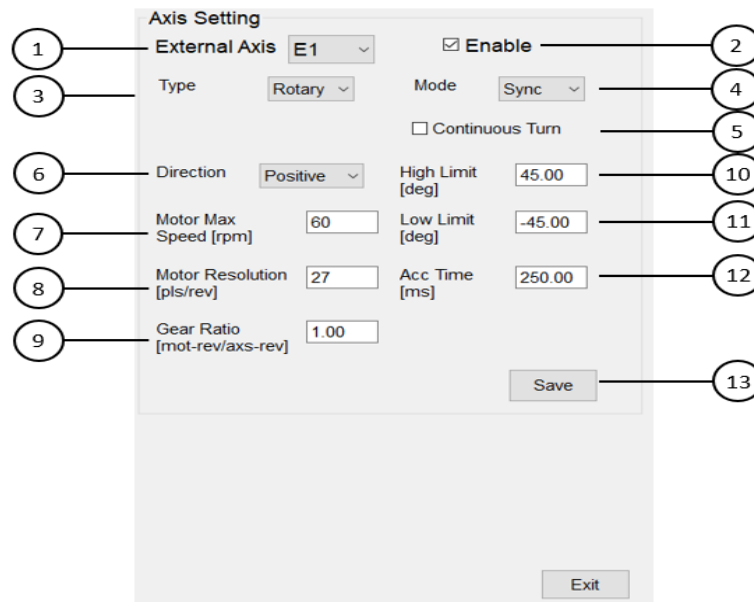
The image shows a software interface titled "Axis Setting" with various configuration options. Numbered callouts (1-13) point to specific elements:

- 1: External Axis dropdown menu (set to E1)
- 2: Enable checkbox (checked)
- 3: Type dropdown menu (set to Linear)
- 4: Mode dropdown menu (set to Sync)
- 5: Direction dropdown menu (set to Positive)
- 6: Motor Max Speed [rpm] input field (set to 60)
- 7: Motor Resolution [pls/rev] input field (set to 27)
- 8: Gear Ratio [mot-rev/axis-rev] input field (set to 1.00)
- 9: High Limit [mm] input field (set to 45.00)
- 10: Low Limit [mm] input field (set to -45.00)
- 11: Acc Time [ms] input field (set to 250.00)
- 12: Pitch [mm/axis-rev] input field (set to 360.00)
- 13: Save button

An Exit button is also visible at the bottom right of the dialog.

External axis linear axis setting

No.	Description
1	Select external axis
2	Enable/disable the selected external axis
3	Set the external axis type to linear / rotary axis
4	Set the motion mode of external axis to synchronous / asynchronous axis <ul style="list-style-type: none"> ● Synchronous motion: Synchronous motion means that the motion of all axes including the robot and the external axis starts and ends at the same time. In the synchronous motion mode, calibrate the motion path relationship between the robot and the external axis and set its kinematics type to control the external axis under the synchronous coordination control. ● Asynchronous motion: Asynchronous motion means that the motion of the robot and the external axis starts or ends at different times, and the motion of the robot and the external axis does not depend on each other.
5	Set the direction of the external axis motor
6	Set the maximum speed of the external axis motor
7	Set the resolution of the external axis motor
8	Set the reduction ratio of the external axis motor assembly reducer
9	Set the upper limit of external axis movement
10	Set the lower limit of external axis movement
11	Set external axis acceleration and deceleration time
12	Set the pitch of the external axis
13	Save settings



External axis rotary axis setting

No.	Description
1	Select external axis
2	Enable/disable the selected external axis
3	Set the external axis type to linear / rotary axis
4	<p>Set the motion mode of external axis to synchronous / asynchronous axis</p> <ul style="list-style-type: none"> ● Synchronous motion: Synchronous motion means that the motion of all axes including the robot and the external axis starts and ends at the same time. In the synchronous motion mode, calibrate the motion path relationship between the robot and the external axis and set its kinematics type to control the external axis under the synchronous coordination control. ● Asynchronous motion: Asynchronous motion means that the motion of the robot and the external axis starts or ends at different times, and the motion of the robot and the external axis does not depend on each other.
5	Sets the infinite rotation function for the external axis
6	Set the direction of the external axis motor
7	Set the maximum speed of the external axis motor
8	Set the resolution of the external axis motor
9	Set the reduction ratio of the external axis motor assembly reducer
10	Set the upper limit of external axis movement
11	Set the lower limit of external axis movement
12	Set external axis acceleration and deceleration time
13	Save settings

Operating Steps

Main Menu> Start up>External Axis>Setting

1. Select the external axis to be set
2. Enable external axis
3. Set the external axis type
4. Set external axis motion interpolation mode
5. Set the motor direction
6. Set the maximum speed of the motor
7. Set the reduction ratio
8. Set high and low limits
9. Set the motor resolution
10. Set the pitch (only linear axis)
11. Set infinite rotation function (rotation axis only)

12. Save settings
13. Repeat the above steps to set other external axes in sequence

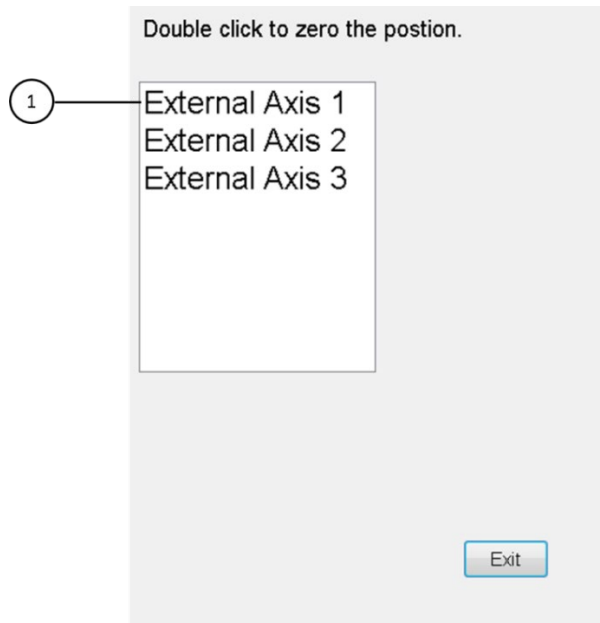
⚠ CAUTION

When setting the motor resolution, please pay attention to the maximum resolution that the driver can support.

2.22.2. External axis zero point correction

Description

To reset the zero point of the external axis, based on safety considerations, after completing the external axis parameter setting, it is necessary to move to the zero point position to be set.



External axis zero point correction

No.	Description
1	Double-click the external axis to reset the zero point

Operating Steps

Main Menu> Start up>External Axis>Zero Position

1. Double-click the external axis to reset the zero point
2. After reset, confirm whether the external axis position and simulation screen are correct

2.22.3. External Axis Positive and Negative Limit Signal Monitoring Interface Setting

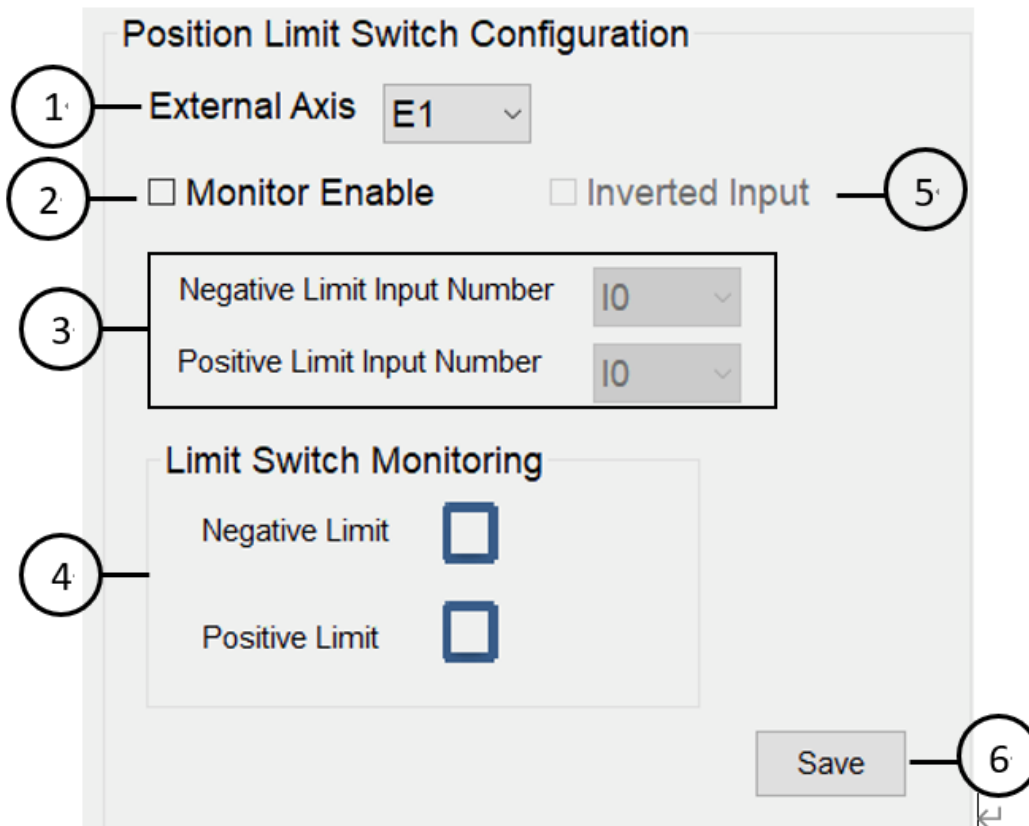
Description

HRSS can monitor the positive and negative limit status of the driver of the external axis (hardware configuration for the external sensor of the external axis driver must be completed first; please refer to the driver manual of each company for the wiring method). HRSS reads the positive and negative limit signal returned by the driver through EtherCat communication; when the external axis is moving, and the limit signal trigger is received, it can be stopped immediately and have warning sounded to avoid violent impact or collision from occurring.

Operating Steps

Main Menu >Start up> External Axis> Position Limit Switch

1. You must first confirm whether the positive and negative limit function parameters of the external axis driver are set (After HRSS has started, if the positive and negative limit function parameters on the driver were adjusted, HRSS needs to be restarted to ensure related parameters of the driver can be read correctly during initialization).
2. Select the external axis number (E1~E3) from the drop-down menu and enable the Monitor Enable monitoring function.
3. The driver has **no** setting function related parameters; the monitoring positive and negative pins must be set manually and set reverse input for the signal according to the requirement. If the driver **has** setting parameters, the monitoring positive and negative pins are preset, and the input pins and reverse input function on the interface cannot be set (Items 3 and 5 in the figure below).
4. Press the Save button to complete the settings.
5. When the sensor or signal source detects the limit, the Limit Switch Monitoring light will turn on; it is red in the frame: Sensor was triggered and turned ON; if it is white in the frame: The sensor was not triggered and is OFF; if the signal display is the opposite, inverting input can be selected to switch the signals (Item 5 in the figure below).



External axis positive and negative limit signal monitoring interface diagram

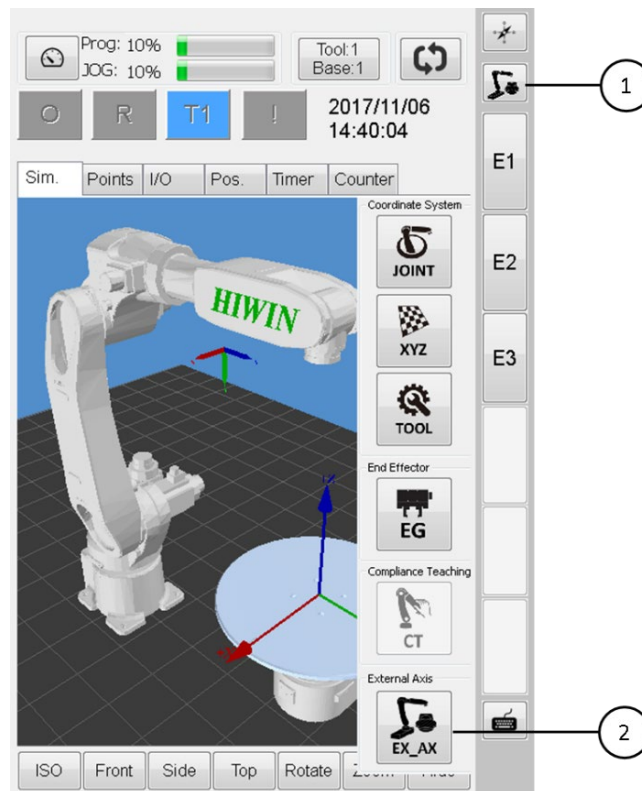
External axis positive and negative limit signal monitoring description table

No.	Function	Description
1	Select the external axis number	Select the E1~E3 external axis number; they must be selected and set separately.
2	Monitor enable selection	Set enable/disable limit signal monitoring for the external axis; selected means enabled signal monitoring.
3	Positive and negative limit pin selection	Sets the external axis positive and negative limit pin selection; if the driver has no setting function related parameters, please plan the pins by referring to the driver manual of each company.
4	Positive and negative limit determination display	Displays the positive and negative limit light; red in the frame: Sensor was triggered and turned ON; white in the frame: Sensor was not triggered and is OFF.
5	Reverse input selection	Sets whether the signal is/not reverse input
6	Save settings button	Save settings

2.22.4. Manual Operations of External Axis

Description

Each external axis that is enabled and set correctly can be manually moved in T1 mode and with the user authority is Expert or Engineer. Each external axis can independently move forward or reverse and adjust the speed override. If synchronous coupled control is turned on, manually moving the external axis will control the robot movement at the same time, and it can be verified whether the calibrated point is correct.



Manual movement of external axis

No	Description
1	Toggle manual movement mode

2	Select manual movement mode for the external axis
---	---

Operation Step

Main Menu> Start up>External Axis>Zero Position

1. Click to switch to T1 mode
2. Select external axis manual movement mode
3. Press and hold the enabling switch
4. Press the positive and negative direction buttons of each axis on the teach pendant to move the external axis
5. Or click the E1 ~ E3 buttons to enter the value to move the external axis

2.22.5. Display information about external axis position

Description

The external axis position information is displayed on the Pos. tab of the main screen. You can observe whether the external axis moves to the specified position during manual operation or running the program. In addition, observe the acceleration and deceleration and rotation speed information to confirm whether the external axis related parameters are set correctly.

Sim.	Points	I/O	Pos.	Timer	Counter
			Parameter	Value	Unit
1			E1 Motor	-4220227	Unit
			E2 Motor	-9426931	Unit
			E3 Motor	0	Unit
2			E1	11.320	degree
			E2	25.285	degree
			E3	N/A	mm
3			E1 Acc	250.000	unit
			E2 Acc	250.000	unit
			E3 Acc	250.000	unit
4			E1 Dec	315.000	unit
			E2 Dec	315.000	unit
			E3 Dec	315.000	unit
5			E1 RPM	0.000	unit
			E2 RPM	0.000	unit
			E3 RPM	0.000	unit
			Robot <=>		
			6		

External axis position information

No.	Description
1	Display the value of the motor encoder of each axis of the external axis
2	Display the position of each axis of the external axis
3	Display the acceleration of each axis of the external axis
4	Display the deceleration of each axis of the external axis
5	Display the speed of each axis of the external axis
6	Switch the position information of the robot or external axis

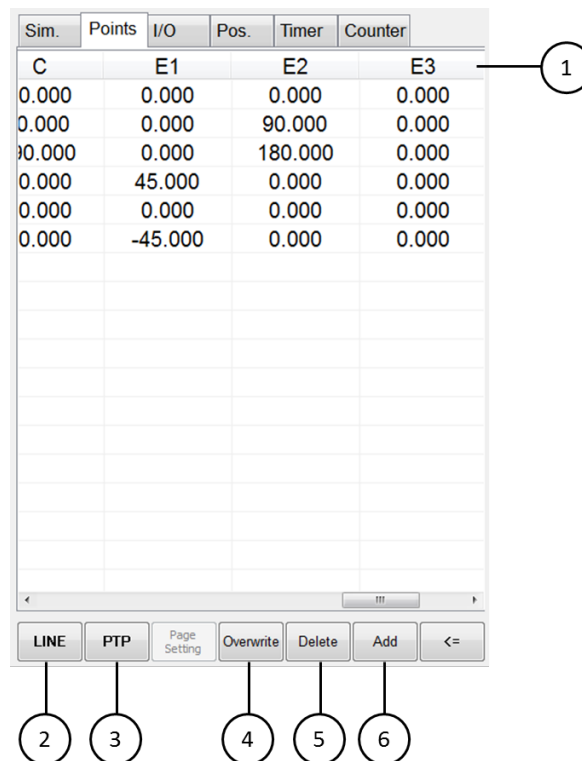
Operation Step

1. Click on the Pos. tab
2. Click the Ext Axis button to switch the position information to the external axis

2.22.6. Point list information

Description

Under the Points tab of the main screen, the points recorded by the currently open program are displayed. The point position information increases the external axis position information of E1 ~ E3 and each external axis mode, and can perform the synchronization of LINE and PTP including external axis movements to perform a point check.



External axis point information

No.	Description
-----	-------------

1	Display external axis point position and external axis mode information
2	Select a point to execute LINE motion
3	Select a point to execute PTP motion
4	Overwrite selected point
5	Delete selected point
6	Record the current position of the robot and external axis

Operation Step

1. Click the Points tab
2. Click the Add button to record the current point
3. Select the point and click Delete to delete the point
4. Select the point and click Overwrite to overwrite the point
5. Select a point and hold down the enabling switch and click LINE to execute LINE motion to that point
6. Select a point and hold down the enabling switch and click PTP to execute PTP motion to that point

2.23. Welding function (optional)

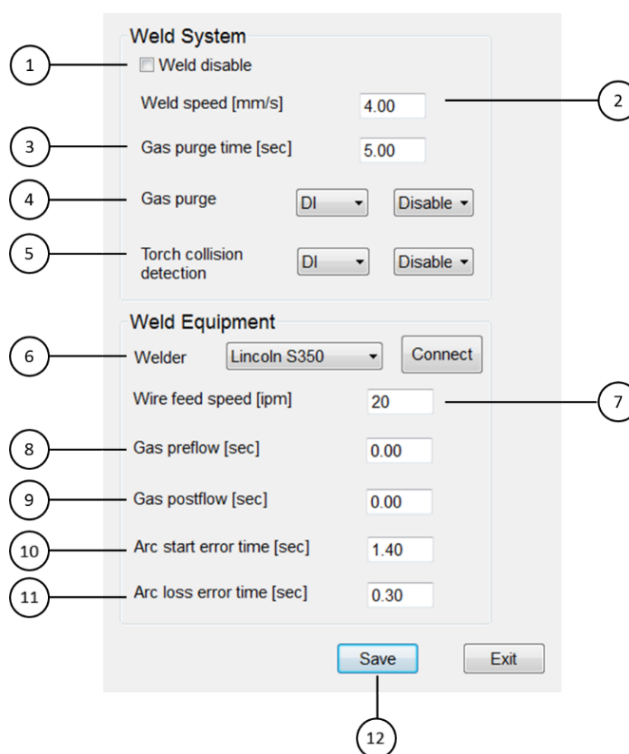
Description

The HRSS provides the welding function of the six-axis robot arm and related HRL commands. It can currently support two types of welding machines, Lincoln S350 and Binzel IROB400, including welding related settings and mode selection, related welding I / O monitoring Weaving setting, welding torch (TCP) recovery and other functions.

2.23.1. Welding system and device settings

Description

After the hardware has been properly connected to the welding machine you want to use, you first need to make relevant settings for the welding system and its device. The welding system needs to be set to enable the arc starting command, the speed of the robot arm during welding, the gas cleaning of the welding gun, the welding gun impact, etc. The settings related to the welding device include the selection of the connected welding machine and connection settings, the wire feed speed of the welding machine, the air supply time before and after welding, the detection time of the arc start failure and the arc shutdown failure, etc.



Welding system and equipment settings

No.	Description
1	Disable welding function, user can confirm welding teaching path
2	Welding speed setting, need to match WELD_SPEED instruction, for example:LIN WELD_SPEED
3	Torch gas purge time setting
4	Torch gas purge input signal setting
5	Torch collision detection input signal setting
6	Welder setting, select the welder and connect
7	Wire feed speed setting
8	Gas preflow time setting
9	Gas postflow time setting
10	Arc start error time setting
11	Arc loss error time setting
12	Save settings

Operation Step

Main Menu> Start up> Arc Weld>Weld Equipment

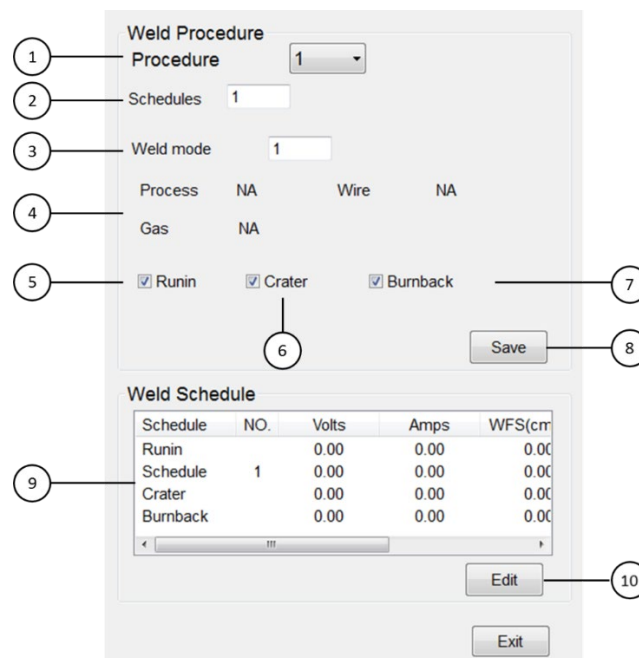
1. Set whether to disable the arc for the teaching path during welding
2. Set the speed during welding
3. Set the torch gas purge time
4. Set the welding torch gas purge input signal

5. Set the input signal for torch collision detection
6. Set up the welder and connect
7. Set wire feed speed
8. Set gas preflow time
9. Set gas postflow time
10. Set arc start error time
11. Set arc loss error time
12. Save settings

2.23.2. Welding procedure setting

Description

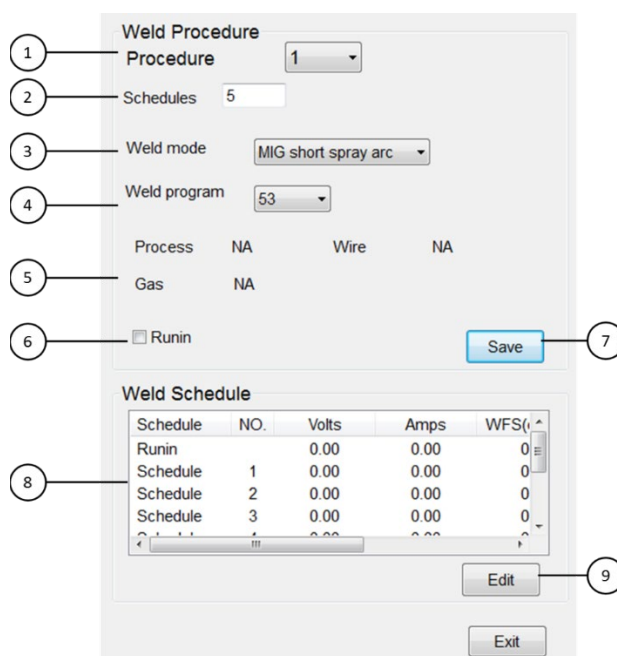
After the related welding system and device settings are completed, the related procedure settings are performed according to the set welder. The user can write and execute welding at the weld start and end commands according to the set welding procedures and conditions.



Lincoln welder welding procedure setting

No.	Description
1	Welding procedure
2	Number of welding schedules in the welding procedure
3	Welding mode
4	Selected workpiece, wire diameter and gas information in the welding mode
5	Welding start-up process for smooth start-up of welding
6	Crater treatment is used to specify a lower voltage and current at a certain time at the end

No.	Description
	of welding to avoid the crater hole caused by a sharp drop in voltage
7	Post-welding treatment is used to prevent welding wire and workpiece from welding by applying voltage at an appropriate time after the wire feed is completed.
8	Save welding procedure settings
9	Welding schedule parameter setting, generally including voltage, current, arc length, wire feed speed and execution time and other parameters
10	Edit welding schedule parameters



Binzel welder procedure setting

No.	Description
1	Welding procedure
2	Number of welding schedules in the welding procedure
3	Welding mode
4	Specified program in welding mode
5	Selected workpiece, wire diameter and gas information in the welding mode
6	Welding start-up process for smooth start-up of welding
7	Save welding procedure settings
8	Welding schedule parameter setting, generally including voltage, current, arc length, wire feed speed and execution time and other parameters
9	Edit welding schedule parameters

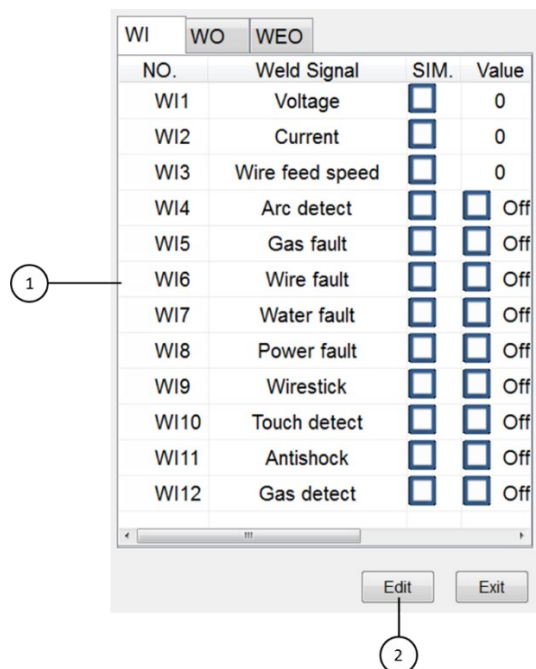
Main Menu> Start up>Arc Weld> Weld Procedure

1. Select welding procedure
2. Set welding schedules
3. Set the welding mode
4. Set the specified program in the welding mode
5. Sets the torch impact detection input signal
6. Set whether to enable runin process, crater process (Lincoln welding machine only), burnback process (Lincoln welding machine only)
7. Save welding procedure settings
8. Select welding schedule to set

2.23.3. Welding I/O

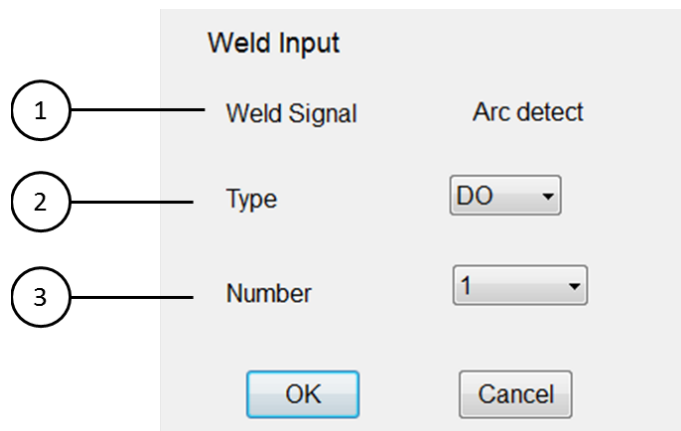
Description

After completing the welding settings and successfully connecting with the welder, you can monitor the relevant welding I/O and output the signal to the peripheral equipment for control. The welding input (WI) signal can correspond to the digital output (DO or SO) signal. The welding output (WO) signal can be triggered by a digital input (DI or SI), and the welding external output (WEO) can be mapped to a digital output (DO or SO).



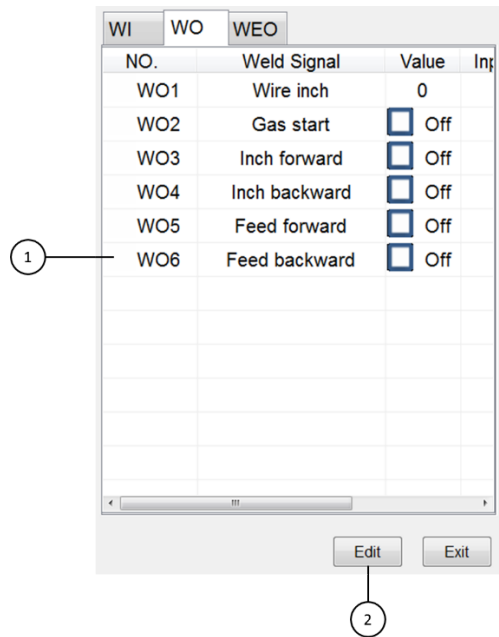
No.	Description
1	Welding input signal <ul style="list-style-type: none"> ● Voltage: voltage feedback value during welding ● Current: current feedback value during welding

No.	Description
	<ul style="list-style-type: none"> ● Wire feed speed: wire feed speed feedback value ● Arc detection: Check whether the welding torch emits voltage to confirm whether welding is being performed ● Gas fault: Check whether the gas is exhausted. If this signal is detected during welding, an alarm will be issued to stop the welding ● Wire fault: Check the wire feeder failure or welding wire exhaustion ● Water fault: check whether the cooling device or circulating hose is faulty. If this signal is detected during welding, an alarm will be issued to stop the welding ● Power fault: check whether the power supply of the welder is abnormal. If this signal is detected during welding, an alarm will be issued to stop the welding ● Wirestick: check whether welding wire and workpiece are fused ● Touch detect: check whether the welding torch touches the workpiece. This signal is currently only enabled by the torch (TCP) recovery ● Antishock(Binzel welder only): check whether the welding torch has an impact. If this signal is detected during welding, an alarm will be issued to stop welding ● Gas detect: check whether the welding machine has gas supply
2	Edit welding input settings



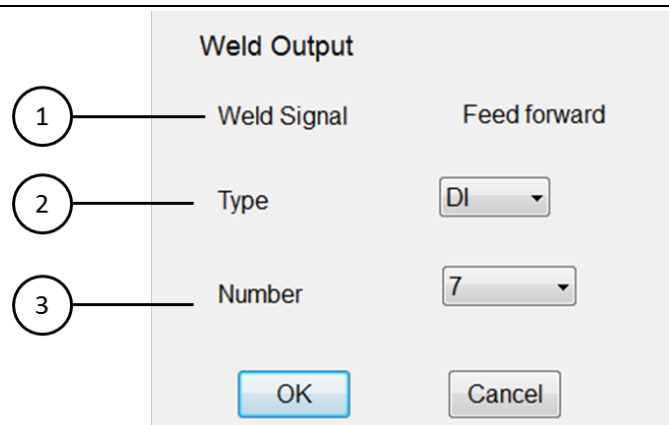
Welding input setting

No.	Description
1	Display edited welding input signal
2	Set the corresponding input signal type to DO or SO
3	Set the corresponding input signal number

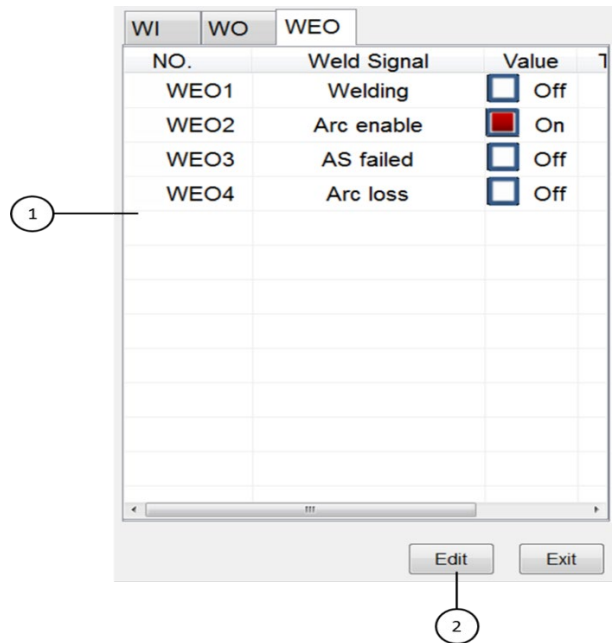


Welding output

No.	Description
1	<p>Welding output signal</p> <ul style="list-style-type: none"> ● Wire inch: the wire feeding amount of the jog wire feeding function ● Gas start: start welding machine gas supply ● Inch forward: feed the wire feed amount set when the signal is triggered ● Inch backward: when this signal is triggered, withdrawn the set wire feed amount ● Feed forward: when this signal is triggered, the welder continuously feeds wire ● Feed backward: when this signal is triggered, the welder continuously draws back wire
2	Edit welding output settings

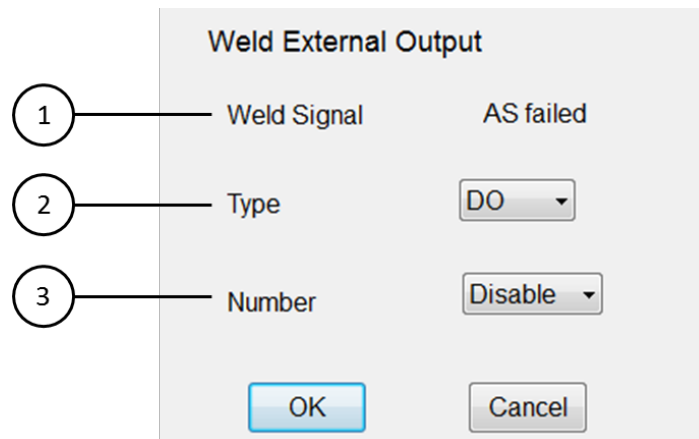


No.	Description
1	Display edited welding output signal
2	Set the corresponding output signal type to DI or SI
3	Set the corresponding output signal number



Weld external output

No.	Description
1	Weld external output signal 1.) Signal while welding: This signal is outputted while welding 2.) Welding effective: This signal is outputted when welding is enabled 3.) Arc failed: This signal is outputted when arc failed 4.) Arc exhausted: This signal is outputted when arc is not detected while welding
2	Edit weld output settings



Weld external output settings

No.	Description
1	Show the edited weld external output signals
2	Sets the corresponding output signal type as DO or SO
3	Sets the corresponding output signal number

Operation Step

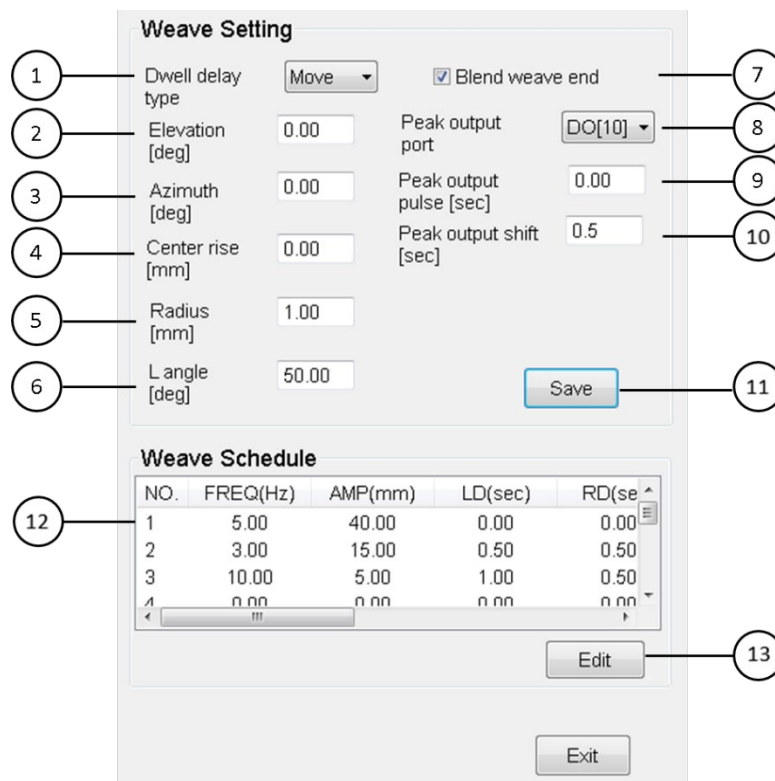
Main Menu> Start up >Arc Weld> Weld I/O

1. Edits the corresponding output of the weld input
2. Edits the trigger input signal of the weld output
3. Edits the corresponding output of the weld external output

2.23.4. Weaving function

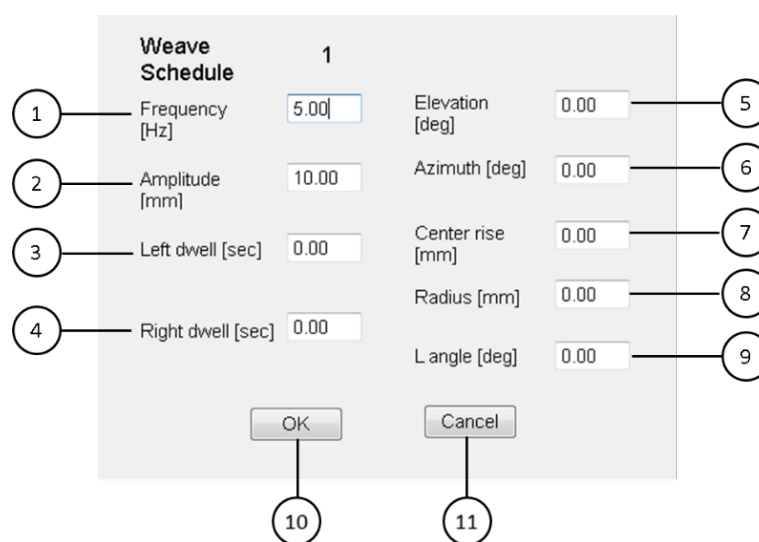
Description

In addition to the elevation, azimuth, center rise and radius in the Weave setting interface, the rest of the settings will be applied to all weaving movements, and the parameters such as elevation, azimuth, center rise, radius and other parameters are used WEAVE Pattern Freq = frequency Hz Amp = amplitude mm LD = left dwell sec RD = right dwell sec, the type of command will be applied. There are 10 weaving conditions. The setting parameters of the weaving conditions will be applied when the WEAVE Pattern WS = Schedule number type command is used.



Weaving setting

No.	Description
1	Dwell delay type: Stop: stop completely at both ends of the weaving. Move: at the two ends of the weaving, only the motion of the weaving stops, and during this period, it moves in the direction of the welding travel.
2	Elevation angle: the tilt of the swing angle by a specified angle relative to the weaving coordinate system.
3	Azimuth angle: the tilt of the azimuth angle of the weaving on the weaving plane.
4	Center rise: the bulge of the welding torch at the center of the weaving, which will only be applied in the sinusoidal weaving.
5	Radius: the amplitude value of the circle-shaped weaving relative to the welding direction.
6	L angle: set in the L-shaped weaving, the angle between the left and right of the weaving and the plane.
7	Blend weave end: setting the weaving connection can smoothly connect the weaving trajectory before and after the teaching point of the action command.
8	Peak output port: set the endpoint output DO. During the welding process, when the torch reaches the endpoint, the specified DO signal is output.
9	Set the pulse width of the endpoint output DO.
10	Set the delay time of endpoint output DO.
11	Save weaving settings
12	Weaving schedule list
13	Edit specified weaving conditions



Weaving schedule setting

No.	Description
1	Frequency: the number of cycles per second for weaving

No.	Description
2	Amplitude: the distance from the welding line to the end point
3	Left dwell: the left dwell time and specify the stop time at the left end of the weaving. When the dwell delay type is moving, the robot moves in the welding direction. This setting is invalid during the circular weaving.
4	Right dwell: the right dwell time, specify the stop time at the right end of the weaving. When the dwell delay type is moving, the robot moves in the welding direction. This setting is invalid during the circular weaving.
5	Elevation: the elevation angle and tilt the swing angle by a specified angle relative to the weaving coordinate system.
6	Azimuth: the tilt of the azimuth angle of the weaving on the weaving plane.
7	Center rise: the center bulge and specify the bulge of the welding torch at the center of the weaving, which will only be applied in the sinusoidal weaving.
8	Radius: the amplitude value of the circular weaving relative to the welding direction.
9	L angle: set in the L-shaped weaving, the angle between the left and right of the weaving and the plane.
10	Confirm and save condition settings
11	Cancel condition setting

● Operation Step

Main Menu> Start up>Arc Weld> Weaving

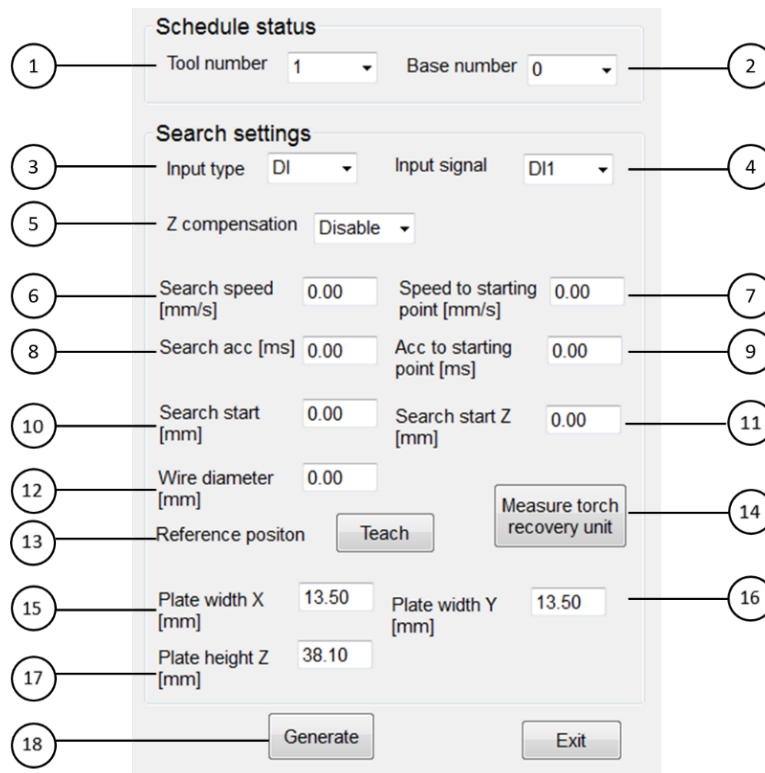
1. Set the dwell delay type of weaving
2. Set the elevation angle of weaving
3. Set the azimuth angle of weaving
4. Set the center rise of weaving
5. Set the radius of weaving(circle weaving only)
6. Set the L angle of weaving(L-shaped weaving only)
7. Set the weaving connection
8. Set the endpoint output DO
9. Set the pulse width of the endpoint output DO
10. Set the delay time of endpoint output DO
11. Save the weaving settings
12. Select weaving schedules for editing
13. Set the frequency of the weaving schedule
14. Set the amplitude of the weaving schedule
15. Set the left dwell time of the weaving schedule
16. Set the right dwell time of the weaving schedule
17. Set the angle of the weaving schedule

18. Set the azimuth angle of the weaving schedule
19. Set the center rise of the weaving schedule
20. Set the L angle of the weaving schedule(L-shaped weaving only)
21. Confirm and save the weaving schedules

2.23.5. Torch(TCP) recovery

Description

After the robot performs welding for a long time, the tip position of the welding wire (TCP) may be shifted due to the wear of the contact tip, the replacement of the contact tip and the welding torch, or the human operation error. When offset, the correct welding cannot be performed, and the position of the wire tip needs to be reset leading to long-term shutdown. The torch recovery function can automatically correct the position of the wire tip in a short time. Using this function can shorten the shutdown time and obtain stable welding quality.



TCP recovery function setting

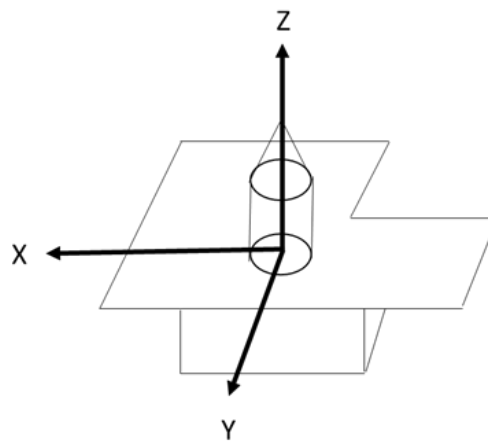
No.	Description
1	Set the tool number to perform TCP recovery
2	Set the base number to perform TCP recovery
3	Set the input signal type to be triggered when the welding wire contacts the torch recovery fixture: <ul style="list-style-type: none"> ● DI: general digital input ● SI: fieldbus digital input

No.	Description
	● WI: welding digital input
4	Set the input signal number to be connected when the welding wire contacts the torch recovery fixture.
5	Set whether to perform TCP compensation in the Z direction.
6	Set the approach speed for searching in the X and Y directions (including Z when Z compensation is enabled). If you slow down the search speed, you can improve the correction accuracy.
7	Set the speed of moving to the X and Y directions (including Z when Z compensation is enabled) to the starting point for searching.
8	Set the approach acceleration time for searching in the X and Y directions (including Z when Z compensation is enabled). If the acceleration time is reduced, the correction accuracy can be improved.
9	Set the acceleration to move to the X and Y directions (including Z when Z compensation is enabled) to the starting point for searching.
10	Set the search distance in X and Y directions
11	Set the search distance in the Z direction
12	Set wire diameter
13	Set the reference position where the welding wire contacts the TCP setting pin point for the correct TCP
14	Perform automatic measurement of the torch recovery fixture function. After the measurement is completed, the distance from the center of the TCP setting pin to the edge of the fixture in the X and Y directions and the height of the TCP setting pin can be obtained.
15	The distance from the center of the pin to the edge of the fixture in the X direction.
16	The distance from the center of the pin to the edge of the fixture in the Y direction.
17	TCP fixture pin height.
18	Generate TCP recovery program, the file name is torch_mate.

Operation Step

Main Menu> Start up>Arc Weld> Torch Mate

1. Set the base to perform TCP recovery. The relationship between the set base and the fixture is as shown in the figure below. At the same time, move the direction along the edge of the fixture to confirm the correction accuracy.



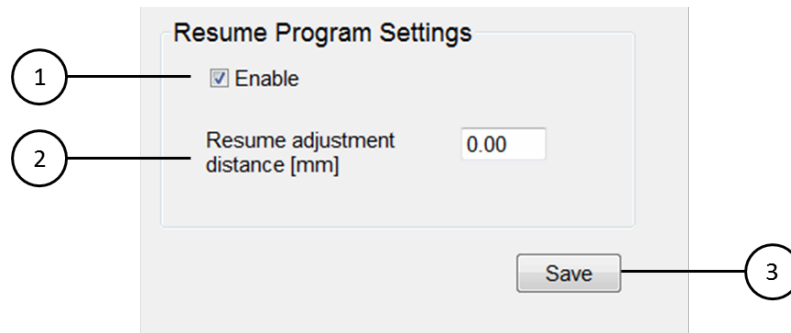
Schematic diagram of base calibration

2. Set the tool coordinates to perform TCP recovery. If the tool coordinates have not been calibrated, you need to perform the 7-point calibration method of the tool or enter the values to confirm that the position and posture of the tool coordinates are correct, and the Z direction of the tool coordinates must match the set The Z direction of the base mark is parallel
3. Set the type and number of the trigger input signal when the welding torch touches the fixture
4. Set whether to perform Z direction compensation
5. Set the search speed and acceleration in X, Y, (Z) direction
6. Set the speed and acceleration of moving to X, Y, (Z) direction to the starting point for searching
7. Set the diameter of the welding wire
8. After moving the welding torch to the tip point of the TCP setting pin, press the teaching button and record the reference position
9. After the above parameters are set, you can manually enter the fixture size or perform automatic measurement function
10. If the automatic measuring fixture function is executed, it must be executed in T1 mode. After the execution, the interface will display the fixture size parameters.
11. After the fixture size parameters are obtained, click the Generate button to generate the TCP recovery program, named torch_mate
12. Use RSR, PNS external signal trigger or manually execute the torch_mate program to automatically correct the TCP position of the torch that has deviated

2.23.6. Resume Program

Description

When the robot arm is running, the program is stopped due to various alarms or manual operations. After the operation is restored, the original program needs to be restarted. For example, when the robot arm strikes the welding gun during the welding process and the alarm stops the program, at this time, you may need to use this function to manually move the robot arm to a safe position, and then start the arc at the interrupted position to continue welding. The user can achieve this by opening this function and using the commands RESUME_PROG and CLEAR_RESUME_PROG and LINE motion instruction interruption will restart from the interrupted position.



Weld restart function setting

No.	Description
1	Enable the resume program function, the default is enabled.
2	Set the adjustment distance of the resume position when specifying the resume program
3	Save settings

Operation Step

Main Menu> Start up>System Setting>Resume Program

1. Enable the resume program function
2. Set the adjustment distance of the resume position when specifying the resume program
3. Save settings
4. Write a program, use the RESUME_PROG and CLEAR_RESUME_PROG instructions. If the alarm occurs in the LINE motion command within the specified range or the program is manually stopped, the original program can be restarted from the interrupted position

2.24. Collision Detection

Description

HRSS provides the collision detection, after the user sets the parameters, it can be turned on or off during the teaching mode or executing programs. When collision detection is enable, damage to robots and

equipment can be reduced.

WARNING

Under the situation that this function is turned on, although it can reduce the damage of the robot and equipment. However, this force will still cause injury, operators should pay attention to their own safety.

CAUTION

This function is currently only available for specific models. This function is for detecting the collision of the robot itself; joints set in TL_SET do not have collision functions.

This function needs to set the correct parameters of payload before it can be used normally.

The specific models are

- Joint (six-axis) robot series:

RA/RT605-710-GB/GC

RA/RT605-909-GB/GC

RA/RT610-1476-GB/GC

RA610-1151-GC

RA/RT610-1355-GC/GB

RA/RT610-1672-GC/GB

RA/RT610-1869-GC/GB

RA620-1621

RA620-1739

SCARA robot series: (RS410-LU all series (including P), and the controller model is RC4-A)

RS410-600-200-LU

RS410-600-400-LU

RS410-700-200-LU

RS410-700-400-LU

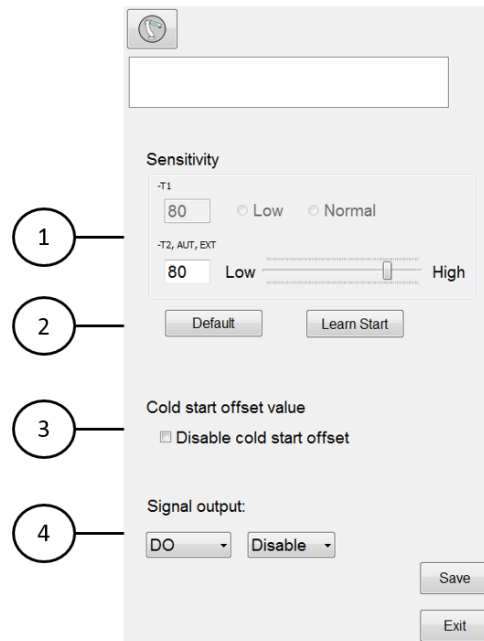
RS410-800-200-LU

RS410-800-400-LU

2.24.1. Parameter Setting

Description

Main Menu>Start-up >System setting > Collision Detection



Collision Detection interface

No.	Description
1	Sensitivity adjustment range is 0~100; the sensitivity is set according to the current mode. When the sensitivity is 0, collision detection is disabled. Under T1 mode, Low is 60 and Normal is 80. Under non-T1 mode, the default is 80.
2	When Learn Start is clicked, collision detection learning begins and it shows Learn Stop; when Learn Stop is clicked, collision detection learning ends and it starts to calculate and movement trajectories during the process from begin to end. The sensitivity will be updated automatically in the field to the left. The learning function is for users to learn the suitable sensitivity value during normal movement process; this can only be operated under T2, AUT and EXT modes.
3	Its use is to prevent excessive resistance from triggering the collision warning when the robot arm is operated before it is fully warmed up; it is enabled by default If the robot arm has already been warmed up or has been operated continuously for a period of time, users can assess whether to disable it
4	When a collision occurs, select DO or SO (Fieldbus Output) as the signal output.

Restriction

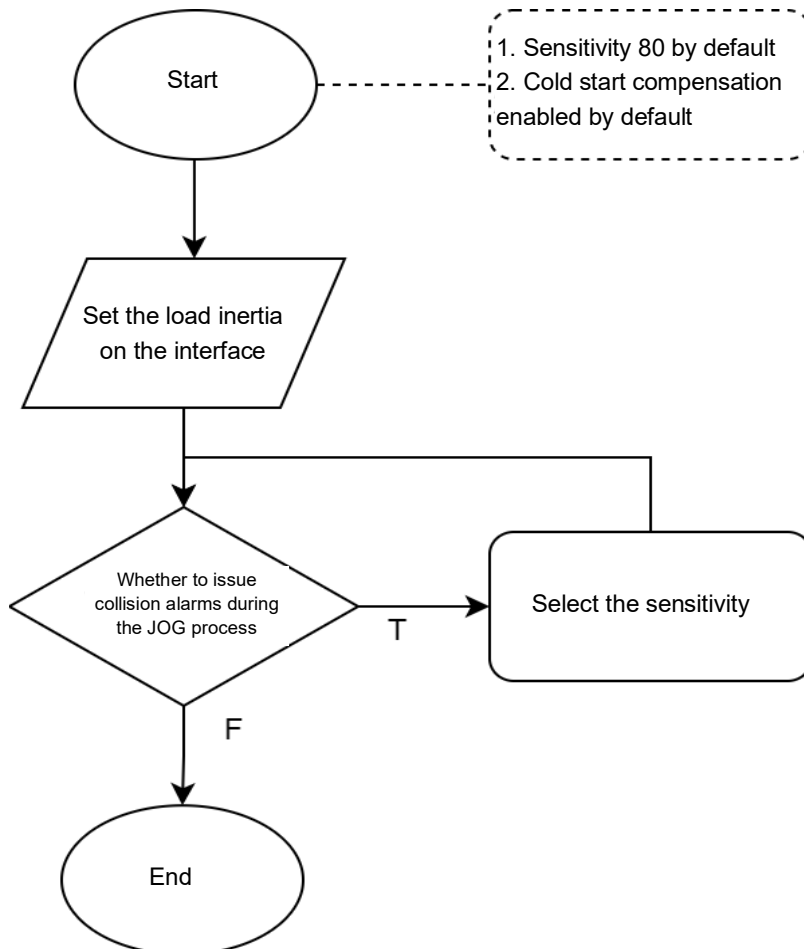
1. While the brake is operating, the collision detection function is invalid.
2. During the payload estimation, collision detection is invalid.
3. While executing collision detection learning, the collision detection function is invalid.
4. When the installation plane of the robot is not perpendicular to the gravity direction (such as: tilted or wall-mounted), or external devices that are too heavy are installed on the arm itself, collision detection

cannot be used correctly at this time. Please disable collision detection manually under such circumstances.

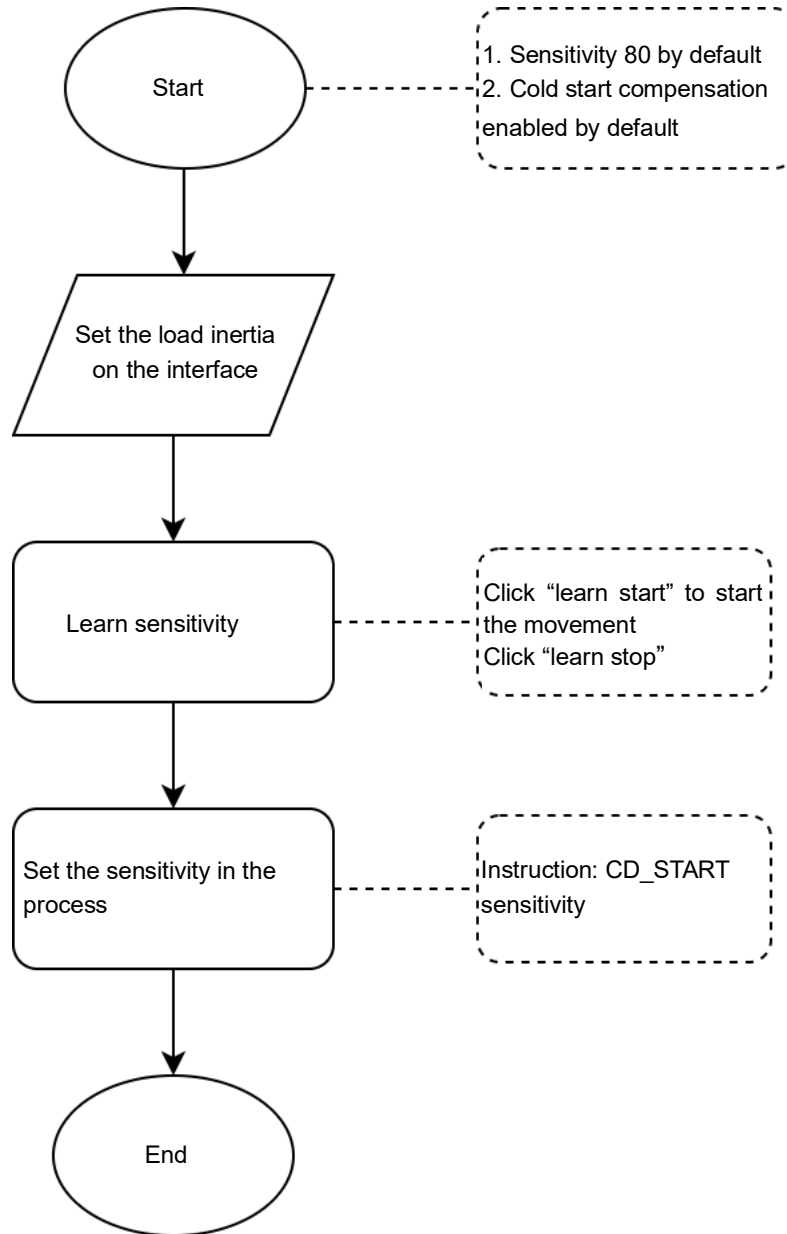
- 5. False alarm condition:
 - A. Parameters of payload are not set correctly.
 - B. Payload exceeds the maximum limit.
 - C. Using SET_ACC command, causing the acceleration to be too high.
 - D. Movement of the arm to near the singular point will cause high speed operation.
 - E. Some commands of motion will have high acceleration.
- 6. False alarm can be avoided by modification of commands.

2.24.2. Description on the Usage Process of Collision Detection

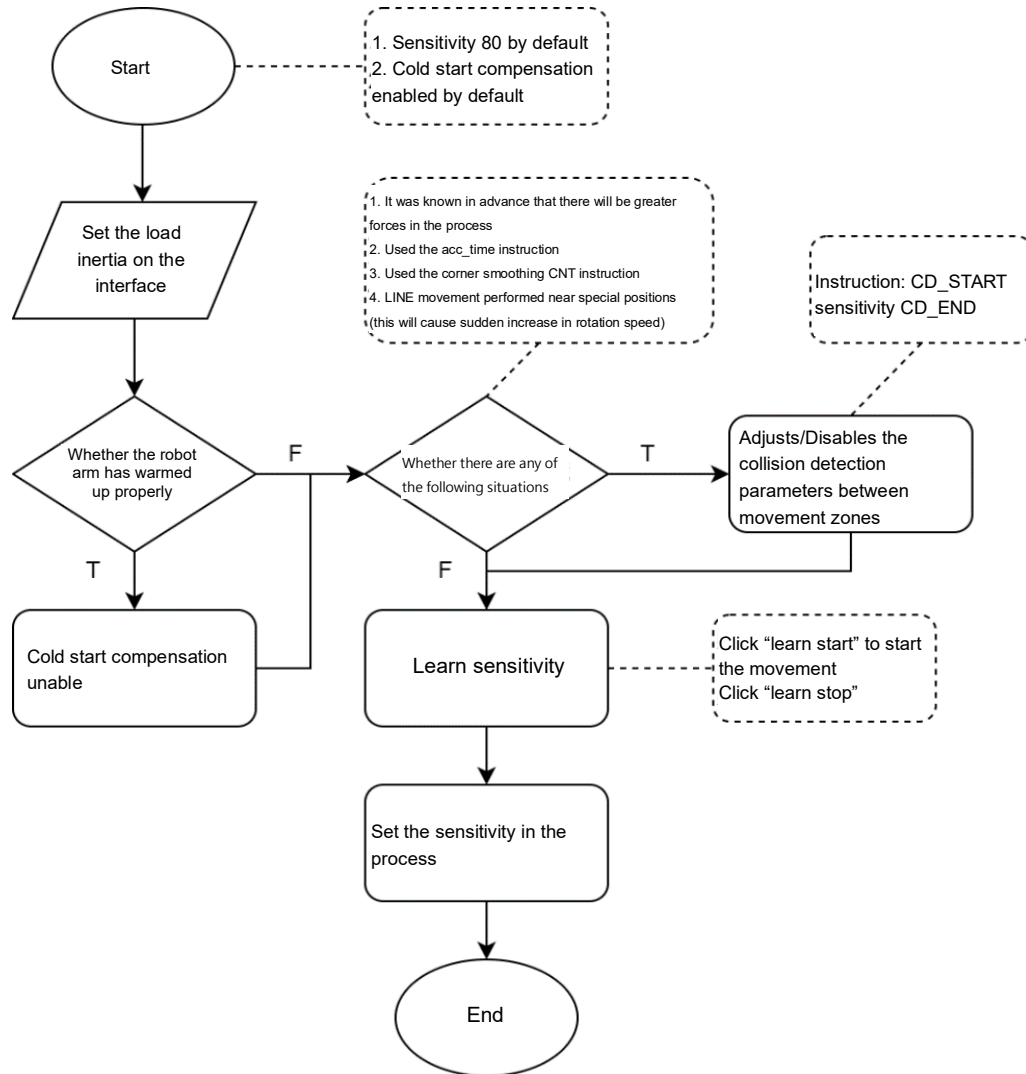
T1 usage process



T2, AUT, EXT usage process



Advanced usage process



2.24.3. Program Command

Description

There are total two commands.

CD_START 80 (turn on collision detection, the value of sensitivity is 80)

CD_END (turn off collision detection)

```

1  CD_START 80
2  LIN P1 FINE=1 Vel=100mm/s Acc=100% TOOL[0] BASE[0]
3  LIN P1 FINE=1 Vel=100mm/s Acc=100% TOOL[0] BASE[0]
4  CD_END
5  LIN P3 FINE=1 Vel=100mm/s Acc=100% TOOL[0] BASE[0]
6  LIN P4 FINE=1 Vel=100mm/s Acc=100% TOOL[0] BASE[0]
  
```

In the sample code, CD_START can turn on collision detection and set the sensitivity to 80 first. During Lin to position of P1, P2, it check whether a collision occurred. After reaching P2, CD_END can turn off collision detection, and subsequent motion commands will not detect for collision events.

2.25. Payload

Description

Set the weight of the end of the robot, including the end tool and the object of gripping, to improve the performance of the movement, adjust the speed limit of the robot and ensure the accuracy of the robot for collision detection. In terms of payload estimation, whether the robot arm will warm up and the weight of payload will affect the accuracy of its estimation.

CAUTION

If it is a non-collision detection model (RA/RT605-710、RA/RT610-1476、RA620-1739), the load estimation page will not be displayed, and LU series model have parameters for setting the center of mass and moment of inertia.

CAUTION

When the payload estimation is performed in T2 mode, the program will be automatically executed, and the information required for the estimation will be obtained by running a specific posture.

Before estimating, please pay attention to the position of the equipment to avoid collision. At the same time, you need to confirm that the related accessories on the arm to avoid interference or falling off.

2.25.1. Parameter Setting

Description

Main Menu>Start-up > System setting>Payload

Different methods of parameter setting:

- A. Payload estimation, automatically estimate the load information at the end of the robot. (The action path is as follows, but the speed of actual execution will be higher)

PTP{A1 0,A2 0,A3 0,A4 0,A5 -90,A6 0}FINE=2 Vel=10% Acc=100% TOOL[0] BASE[0]

PTP{A6 175}FINE=2 Vel=10% Acc=100% TOOL[0] BASE[0]

PTP{A6 -175}FINE=0 Vel=10% Acc=100% TOOL[0] BASE[0]

PTP{A6 175}FINE=0 Vel=10% Acc=100% TOOL[0] BASE[0]

PTP{A5 0}FINE=2 Vel=10% Acc=100% TOOL[0] BASE[0]

PTP{A6 -175}FINE=0 Vel=10% Acc=100% TOOL[0] BASE[0]

PTP{A5 -45,A6 0}FINE=2 Vel=10% Acc=100% TOOL[0] BASE[0]

PTP{A3 45,A5 45}FINE=0 Vel=10% Acc=100% TOOL[0] BASE[0]

PTP{A3 0,A5 -45}FINE=0 Vel=10% Acc=100% TOOL[0] BASE[0]

PTP{A4 90, A5 -60,A6 0}FINE=2 Vel=10% Acc=100% TOOL[0] BASE[0]

PTP{A5 60}FINE=0 Vel=10% Acc=100% TOOL[0] BASE[0]

PTP{A5 -60}FINE=0 Vel=10% Acc=100% TOOL[0] BASE[0]

PTP{A6 45}FINE=2 Vel=10% Acc=100% TOOL[0] BASE[0]

PTP{A5 60}FINE=0 Vel=10% Acc=100% TOOL[0] BASE[0]

PTP{A5 -60}FINE=0 Vel=10% Acc=100% TOOL[0] BASE[0]

PTP{A6 90}FINE=2 Vel=10% Acc=100% TOOL[0] BASE[0]

PTP{A5 60}FINE=0 Vel=10% Acc=100% TOOL[0] BASE[0]

PTP{A5 -60}FINE=0 Vel=10% Acc=100% TOOL[0] BASE[0]

PTP{A1 0,A2 0,A3 0,A4 0,A5 -90,A6 0}FINE=2 Vel=10% Acc=100% TOOL[0] BASE[0]

B. Manual setting, parameters can be set according to the manufacturer, CAD file and simple geometric calculation.

The Manual Setting interface consists of three sequential windows:

- Window 1 (Top):** Initial state with all numerical fields set to 0.000. Callouts 1-7 point to: 1) 'Set payload data' dropdown, 2) 'Mass [kg]' field, 3) 'Center [mm]' section, 4) 'Moment of inertia [kg*mm²]' section, 5) 'Payload Estimation' section, 6) 'User Define' button, and 7) 'Set active' button.
- Window 2 (Bottom Left):** Updated values: Mass [kg] = 1.000, Center [mm] (Xc, Yc, Zc) = 0.000, Moment of inertia (Ixx, Iyy, Izz) = 0.000. Callouts 8-10 point to: 8) 'Confirm' button, 9) 'Edit' button, and 10) 'Payload moment of inertia related to C.G. (center of gravity)' diagram.
- Window 3 (Bottom Right):** Further updated values: Mass [kg] = 0.00, Moment of inertia (Izz) = 0.00. Callouts 8-9 point to: 8) 'Confirm' button and 9) 'Edit' button.

Manual Setting interface

No.	Description
1	Store 20 groups of payload data and comment
2	Set the mass of payload, unit: kilogram
3	Set the center position of the payload, unit: mm
4	Set the moment of inertia, unit: kg*mm ²
5	Payload estimation, it is executed in T2 mode
6	Enter the manual setting page
7	Set the desired group of payload
8	Save parameters of manual setting
9	Edit parameters of manual setting
10	Back to payload estimation page

If the mass, center of mass, and moment of inertia parameters are set and used, it will affect the acceleration and change the upper limit of speed.

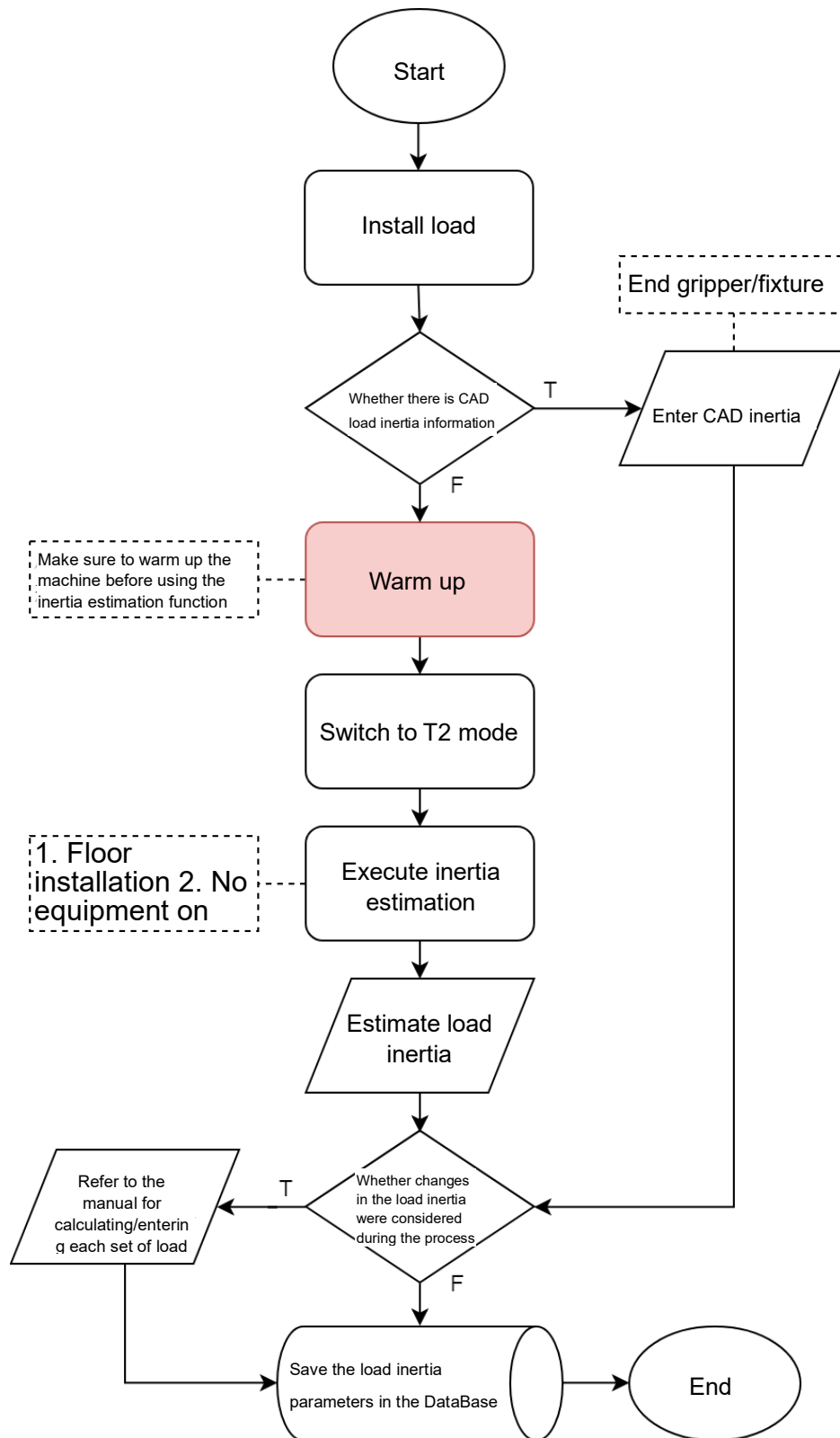
Operation steps (non-collision detection model)

1. Click Edit
2. Enter the mass value
(LU series models can set the center of mass and moment of inertia parameters according to the requirements.)
3. Click Confirm to save data
4. After clicking Set active, set the current payload as active

Operation steps (collision detection model)

1. Select T2 mode
2. Execute Step 1, this action is to ensure that there will be no interference, if there is no problem, you can end early.
3. Step 2 is executed, and switch can only be released after the action is completed. If there is an interruption during the execution, you need to click Execute again, and the action will restart.
4. Execute step 3 to calculate the payload. The calculation result will be directly set to the current group of payload and stored it.
5. If there are parameters to be adjusted, click User Define to enter the page.
6. After modification, click Confirm to save the parameters.
7. After clicking Set active, set the current payload as active.

2.25.2. Description on the Usage Process of Load Setting



2.25.3. Program Command

Description

SET_PAYLOAD[1] (Using the first group of payload data)

```

1  CD_START 80
2  SET_PAYLOAD[1]
3  LIN P1 FINE=1 Vel=100mm/s Acc=100% TOOL[0 BASE[0]
4  LIN P2 FINE=1 Vel=100mm/s Acc=100% TOOL[0 BASE[0]
5  CD_END
6  LIN P3 FINE=1 Vel=100mm/s Acc=100% TOOL[0 BASE[0]
7  $DO[1] = TRUE
8  SET_PAYLOAD[2]
9  CD_START 50
10 LIN P4 FINE=1 Vel=100mm/s Acc=100% TOOL[0 BASE[0]

```

The first group of payload data is the tool at the end of the robot. After moving to P2, gripping the object and it is changed to the second group of payload data, which is the sum of the tool and the object.

2.25.4. Payload calculation

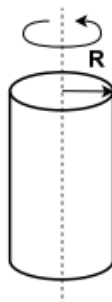
Description

When setting the load, if there is no CAD, the simplified geometric model can also be used to estimate the payload, including the mass, the position of the center of mass, and the moment of inertia relative to the center of mass. Next, the basic formula of moment of inertia will be listed, and then use an example to illustrate how to calculate.

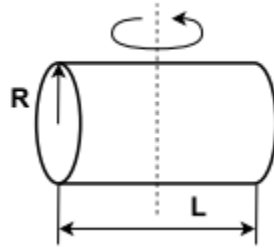
1. Moment of Inertias' basic formula

When the mass is uniformly distributed, the moment of Inertia of the object can be calculated using the following formulas.

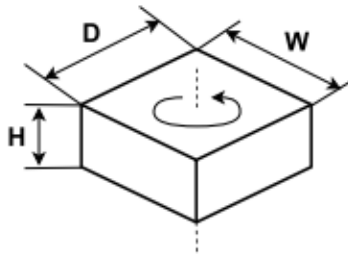
A. Cylinder 1: $I = \frac{1}{2}mR^2$



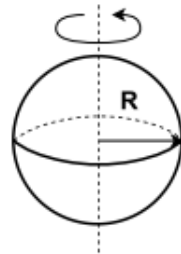
B. Cylinder 2: $I = \frac{1}{12}m(3R^2 + L^2)$



C. Cuboid: $I = \frac{1}{12}m(W^2 + D^2)$

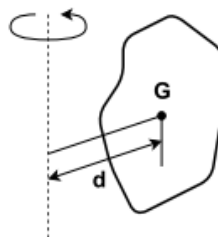


D. Sphere: $I = \frac{2}{5}mR^2$



E. Parallel axis theorem: $I_{new} = I_G + md^2$

Where I_G is the moment of inertia of the object relative to the center of mass. When the axis of rotation translates to a distance d from the center of mass, the moment of inertia of the object for the axis of rotation is I_{new} .



2. Example of payload calculation

In this example, real grippers and objects are used to illustrate the payload calculation. The purpose is to use a simplified geometric model to represent it, and set the corresponding payload data to the interface. As shown in Figure 1, locking the connecting plate to the end of the robot and install the grippers, and then grab a cylindrical object. Geometry can be simplified with 2 cuboids and 1 cylinder, as shown by the dotted line in Figure 1. Among them, the mass is m , the centroid distances are $X_c \cdot Y_c \cdot Z_c$, and the moment of inertia relative to the centroid is $I_{xx} \cdot I_{yy} \cdot I_{zz}$. In addition, the disassembled three simplified models use subscript 1 to represent the connecting plate (cuboid), subscript 2 to represent the gripper (cuboid), and subscript 3 to represent the object(cylinder).

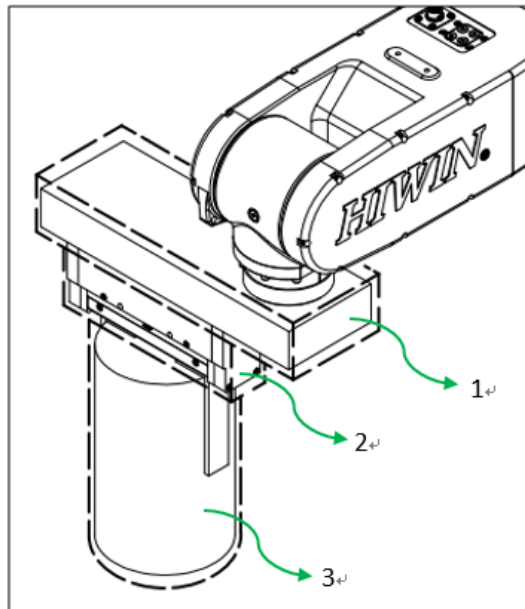


Fig1. Real gripper and the object

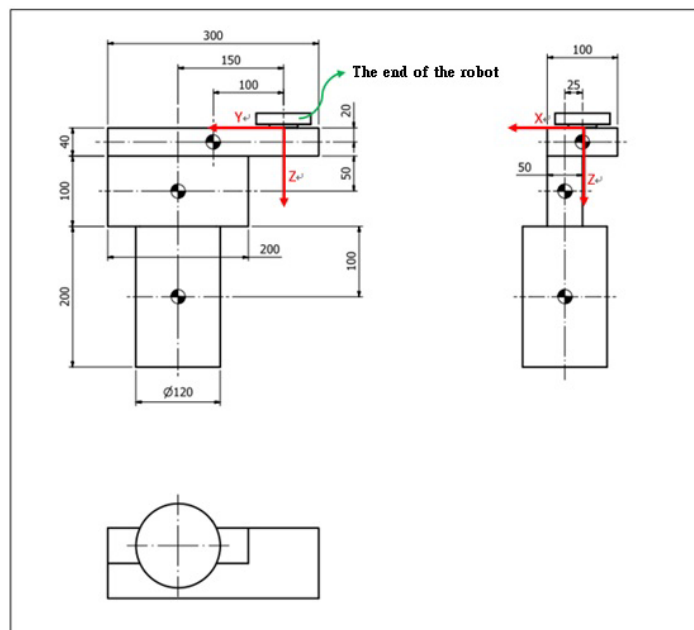


Fig2. Simplified model size

The size of the simplified model is shown in Figure 2; the mass of the connecting plate (cuboid) is 2kg, the length, width, and height are 300, 100, and 40mm; the mass of the gripper (cuboid) is 1kg, and the length, width, and height are respectively 200, 50, 100mm; the mass of the object (cylinder) is 1kg, the diameter is 120mm, and the height is 200mm. According to this parameter, the load information can be calculated:

A. Mass

$$m = m_1 + m_2 + m_3 = 2 + 1 + 1 = 4$$

B. The distance of centroid

Connecting plate (cuboid):

$$X_{c1} = 0$$

$$Y_{c1} = 100$$

$$Z_{c1} = 20$$

Gripper (cuboid):

$$X_{c2} = 25$$

$$Y_{c2} = 150$$

$$Z_{c2} = 40 + 50 = 90$$

The object (cylinder):

$$X_{c3} = 25$$

$$Y_{c3} = 150$$

$$Z_{c3} = 40 + 100 + 100 = 240$$

The calculation:

$$X_c = X_{c1} \cdot \frac{m_1}{m} + X_{c2} \cdot \frac{m_2}{m} + X_{c3} \cdot \frac{m_3}{m} = 0 \cdot \frac{2}{4} + 25 \cdot \frac{1}{4} + 25 \cdot \frac{1}{4} = 12.5$$

$$Y_c = Y_{c1} \cdot \frac{m_1}{m} + Y_{c2} \cdot \frac{m_2}{m} + Y_{c3} \cdot \frac{m_3}{m} = 100 \cdot \frac{2}{4} + 150 \cdot \frac{1}{4} + 150 \cdot \frac{1}{4} = 125$$

$$Z_c = Z_{c1} \cdot \frac{m_1}{m} + Z_{c2} \cdot \frac{m_2}{m} + Z_{c3} \cdot \frac{m_3}{m} = 20 \cdot \frac{2}{4} + 90 \cdot \frac{1}{4} + 240 \cdot \frac{1}{4} = 92.5$$

C. Moment of inertia relative to the center of mass

Connecting plate (cuboid):

$$I_{xx1} = \frac{1}{12} \cdot 2 \cdot (40^2 + 300^2) \cong 15266$$

$$I_{yy1} = \frac{1}{12} \cdot 2 \cdot (40^2 + 100^2) \cong 1933$$

$$I_{zz1} = \frac{1}{12} \cdot 2 \cdot (100^2 + 300^2) \cong 16666$$

Gripper (cuboid):

$$I_{xx2} = \frac{1}{12} \cdot 1 \cdot (100^2 + 200^2) \cong 4166$$



$$I_{yy2} = \frac{1}{12} \cdot 1 \cdot (100^2 + 50^2) \cong 1041$$

$$I_{zz2} = \frac{1}{12} \cdot 1 \cdot (50^2 + 200^2) \cong 3541$$

The object (cylinder):

$$I_{xx3} = \frac{1}{12} \cdot 1 \cdot (3 \cdot 60^2 + 200^2) \cong 4233$$

$$I_{yy3} = \frac{1}{12} \cdot 1 \cdot (3 \cdot 60^2 + 200^2) \cong 4233$$

$$I_{zz3} = \frac{1}{2} \cdot 1 \cdot 60^2 = 1800$$

The calculation:

$$\begin{aligned} I_{xx} &= I_{xx1} + m_1((Y_c - Y_{c1})^2 + (Z_c - Z_{c1})^2) + \\ & I_{xx2} + m_2((Y_c - Y_{c2})^2 + (Z_c - Z_{c2})^2) + \\ & I_{xx3} + m_3((Y_c - Y_{c3})^2 + (Z_c - Z_{c3})^2) \\ & \cong 15266 + 2(((125 - 100)^2 + (92.5 - 20)^2)) + \\ & 4166 + 1(((125 - 150)^2 + (92.5 - 90)^2)) + \\ & 4233 + 1(((125 - 150)^2 + (92.5 - 240)^2)) \cong 58442 \end{aligned}$$

$$\begin{aligned} I_{yy} &= I_{yy1} + m_1((X_c - X_{c1})^2 + (Z_c - Z_{c1})^2) + \\ & I_{yy2} + m_2((X_c - X_{c2})^2 + (Z_c - Z_{c2})^2) + \\ & I_{yy3} + m_3((X_c - X_{c3})^2 + (Z_c - Z_{c3})^2) \\ & \cong 1933 + 2(((12.5 - 0)^2 + (92.5 - 20)^2)) + \\ & 1041 + 1(((12.5 - 25)^2 + (92.5 - 90)^2)) + \\ & 4233 + 1(((12.5 - 25)^2 + (92.5 - 240)^2)) \cong 40108 \end{aligned}$$

$$\begin{aligned} I_{zz} &= I_{zz1} + m_1((X_c - X_{c1})^2 + (Y_c - Y_{c1})^2) + \\ & I_{zz2} + m_2((X_c - X_{c2})^2 + (Y_c - Y_{c2})^2) + \\ & I_{zz3} + m_3((X_c - X_{c3})^2 + (Y_c - Y_{c3})^2) \\ & \cong 16666 + 2(((12.5 - 0)^2 + (125 - 100)^2)) + \\ & 3541 + 1(((12.5 - 25)^2 + (125 - 150)^2)) + \\ & 1800 + 1(((12.5 - 25)^2 + (125 - 150)^2)) \cong 25133 \end{aligned}$$

Finally, filling in the calculation results in the load setting page:

Set payload data. 1 (active)

Payload data

Mass [kg] 4.0

Center [mm] Xc 12.5 Yc 125 Zc 92.5

Moment of inertia [kg*mm²] Ixx 58442 Iyy 40108 Izz 25133

Back Exit

2.26. Soft Limit

Description

The software limit function is provided in HRSS. This function mainly enables the robot to immediately issue an alarm and stop the movement when it issues a motion command that exceeds the set limit, or a position that exceeds the set limit during movement.

According to the articulated, SCARA or Delta robot arm, set the software motion range limit of each axis and space Cartesian coordinates.

The high and low limits of each axis and the high and low limits of the Cartesian coordinates can be set in the software limit setting interface. The limit range of the Cartesian coordinates is set according to the tool center position (TCP) of the robot when the base is marked as 0 (Base 0). Choose whether to enable the limit check for each axis or the Cartesian coordinate setting, and the setting user interface is different according to different robot arms. The articulated robot can set the high and low limits of the A1 ~ A6 axis on each axis limit, and the Cartesian coordinate limit can be set. The high and low limits of X, Y, and Z. The SCARA robot can set the high and low limits of the A1 to A4 axes on the limits of each axis. The limit of the

Cartesian coordinate can also set the high and low limits of X, Y, and Z. The parallel robot limits on each axis
The high and low limits of the A1 ~ A3 axis are set uniformly, and the A4 axis is set separately. The limit of the Cartesian coordinate is to set the high and low limits of the radius and the high and low limits of Z.

Soft Limit		
Joint		
	Low Limit	High Limit
<input checked="" type="checkbox"/> Enable		
A1[degree] (-165.000~165.000)	-60.000	30.000
A2[degree] (-125.000~85.000)	-60.000	30.000
A3[degree] (-55.000~185.000)	-60.000	30.000
A4[degree] (-190.000~190.000)	-300.000	-30.000
A5[degree] (-115.000~115.000)	-100.000	100.000
A6[degree] (-360.000~360.000)	-30.000	30.000
Cartesian		
	Low Limit	High Limit
<input checked="" type="checkbox"/> Enable		
X[mm]	-100.000	300.000
Y[mm]	200.000	900.000
Z[mm]	-150.000	0.000
Save		Exit

Articulated robot soft limit setting interface

Soft Limit		
Joint		
	Low Limit	High Limit
<input checked="" type="checkbox"/> Enable		
A1~A3[degree] (-62.000~85.000)	-60.000	30.000
A4[degree] (-360.000~360.000)	-300.000	-30.000
Cartesian		
	Low Limit	High Limit
<input checked="" type="checkbox"/> Enable		
Radius[mm]	20.000	60.000
Z[mm]	-150.000	0.000
Save		Exit

Delta robot soft limit setting interface

Soft Limit

Joint

Enable Low Limit High Limit

A1[degree] -30.000 30.000
(-130.000~130.000)

A2[degree] -30.000 30.000
(-150.000~150.000)

A3[mm] -30.000 30.000
(-175.000~45.000)

A4[degree] -100.000 100.000
(-360.000~360.000)

Cartesian

Enable Low Limit High Limit

X[mm] -100.000 300.000

Y[mm] 200.000 900.000

Z[mm] -150.000 0.000

Save Exit

SCARA robot soft limit setting interface

No.	Description
1	Enable/disable the soft limit function of joint
2	Set software limits for each axis
3	Enable/disable Cartesian coordinate soft limit function
4	Set the soft limit of each Cartesian coordinate
5	Save settings

Operation Step

Main Menu>Start up>System Setting>Soft Limit

1. Set whether to enable the soft limit function of each axis
2. Set software limits for each axis
3. Set whether to enable the Cartesian coordinate soft limit function
4. Set the soft limit of each Cartesian coordinate
5. Save settings

2.27. Software Update

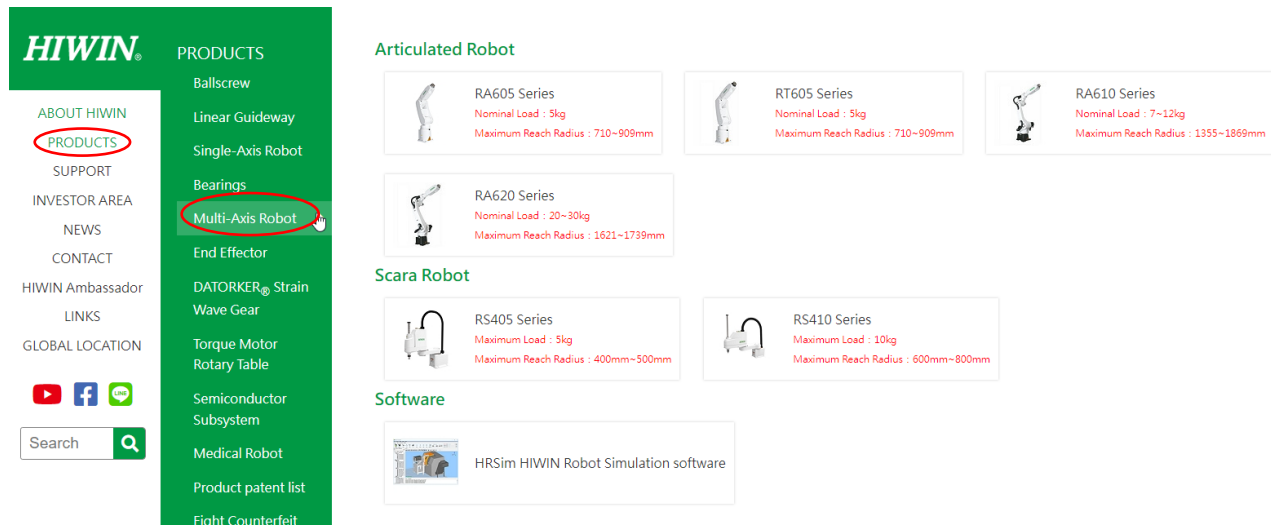
Description

User can download HRSS program updates from the official HIWIN website, and operate update in HRSS interface.

Operation steps

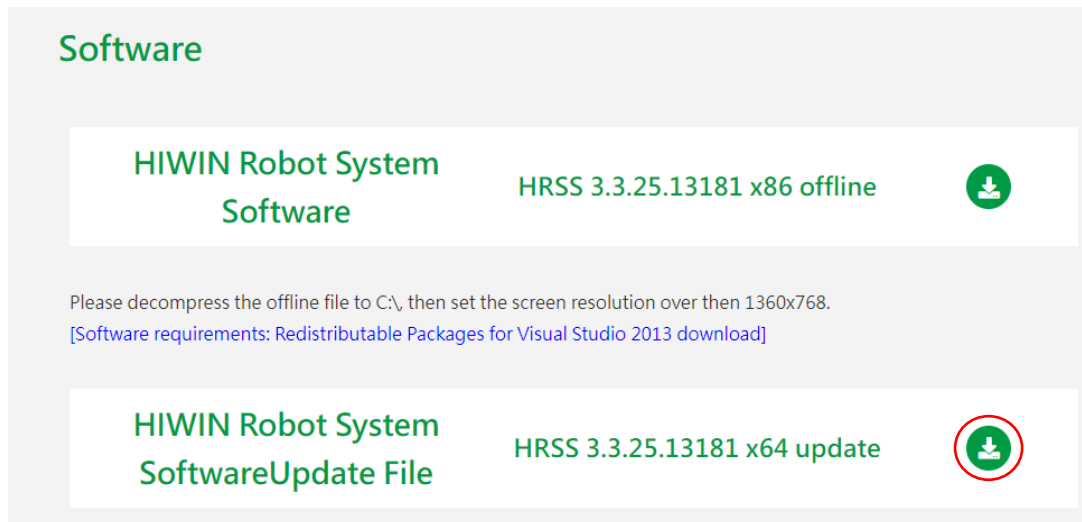
1. Connect to the official website of HIWIN (www.hiwin.tw).

On the web page select: PRODUCTS>Robot



Robot Model

2. Find the model number of robot from the Industrial Robot area below, and then click to enter.



Download documents

3. Select 「 Documents 」 from the above list.

4. Click the updated version of Download from below, and download to USB, the path is placed under the HIWIN folder of root, insert into the USB port of controller.

5. Select Main Menu>Help>Update.
6. A window will appear warning that the Robot will stop when updating software. After pressing OK, software will start to update; to cancel update, press CANCEL.
7. HRSS will reboot and complete update.

 **CAUTION**

Before updating the software, please check current software version, for example: HRSS 3.2.1.2673, please download version with same two number at the front, e.g. HRSS 3.2.2.2775 or HRSS 3.2.4.2925. Do not download version that has two different number at the front, e.g. HRSS 3.3.x.x to avoid incompatible.

2.28. Export Manual

Description


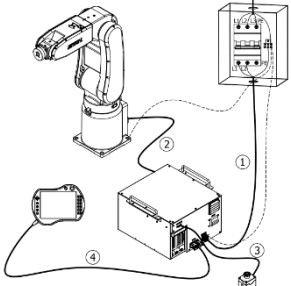


User can export the manual from controller to the USB memory stick.

Operation Steps

1. Insert USB to the controller
2. Select Main menu>Help>Manual
3. Display save successfully
4. Manual save in the path USB:\HIWIN\Manual\Folder

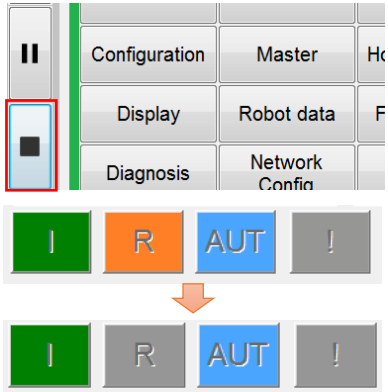

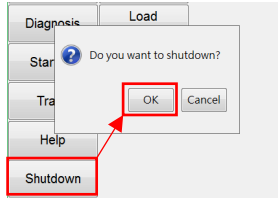


2.29. Controller Boot/Shutdown Program Description

2.29.1. Standard Power ON Procedure

Standard Power on Procedure		
No	Procedure Description	Schematic/Photo
1	Before connecting the main power supply to the controller, please check whether the specification of factory output voltage is in the range of single-phase 200~240V. If it does not match, please use the corresponding transformer (optional item of HIWIN transformer is recommended).	 <p>HIWIN Robot Controller Read the instruction manual carefully before operate the product.</p> <p>Model: RCA610-GC Controller S/N: R19004A-2-11 Robot S/N: R20003Z-2-02 Manufacture Date: Apr.2021 Weight: 38 KG Power Supply: 1Φ, 200-240 VAC Rated Current: 18A Frequency: 50/60 Hz Protection Class: IP23 Document No. C24U104</p> <p>Made in Taiwan No. 7, Jingke Road, Taichung Precision Machinery Park, Taichung 40852, Taiwan</p> <p style="text-align: right;">CE</p>
2	<p>Please confirm:</p> <p>(1) Robot controller must connect to the below 30mA of electricity leakage breaker.</p> <p>(2) Connect the ground directly, instead of connecting the ground by devices or system ground.</p> <p>(3) Please use the high-quality wires with the equal diameter or larger than 14AWG for grounding.</p>	 <p>The diagram shows a robot arm connected to a control cabinet. A ground wire (1) is connected to the cabinet's ground terminal. Another ground wire (2) is connected to the robot's ground terminal. A third ground wire (3) is connected to a separate ground point. A fourth wire (4) is connected to the power supply.</p>
3	<p>Please confirm:</p> <p>(1) Whether an EMO-Emergency stop switch box (EN 60947-5-1 positive opening) with a safety module already connected to the CN3 port of the controller.</p> <p>(2) Whether the robot is installed securely in the protection area.</p>	 <p>A yellow emergency stop switch box with a red button and a black cable.</p>
4	Please manually operate the power switch to upper side to turn on the power.	 <p>A white power switch with a blue handle and the word "POWER" printed above it.</p>

Warning: Please don't turn on the power immediately after turning off the power by switch. Please wait for 30 seconds to turn-on again.

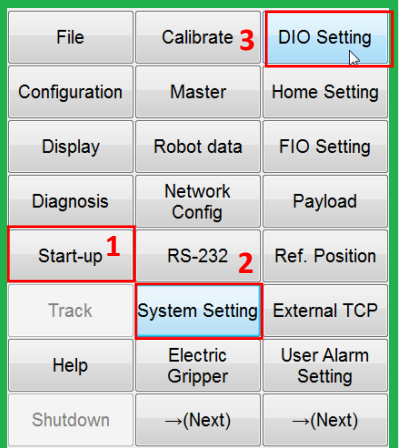
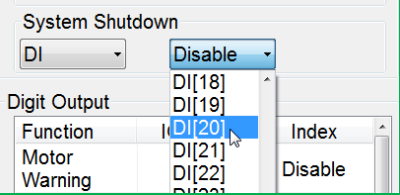
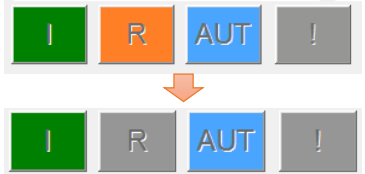

2.29.2. Standard Power OFF Procedure

Standard Power OFF Procedure		
Method 1. By using teach pendant		
No	Procedure Description	Schematic/Photo
1	<p>Please stop the robot, and confirm the motion is completely stopped. <u>The following any one of the methods you can stop the robot:</u></p> <p>(1) Click the Stop button on the teach pendant interface.</p> <p>(2) Execute an appropriate program of return, and move the robot back to the standby/home position.</p> <p><i>Warning: Unless there is an emergency, please do not use the EMG-stop to stop the robot while the robot in the normal motion or still the program is running.</i></p>	
2	Please press the emergency stop button.	
3	Please open the main menu of teach pendant, and click "Shutdown", and then click "OK".	
4	<p>Please confirm one of the below items is finished:</p> <p>(1) The screen of teach pendant is off.</p> <p>(2) The green light indicator on the controller is turned off.</p>	
5	Please manually operate the power switch to lower side to turn off the power.	

Warning: Please do not directly switch off the power of controller (Step-4) or the main power source in the factory before completing the above standard power turn off process (Step1-3). Otherwise, the controller may be damaged due to the abnormal power off procedure.

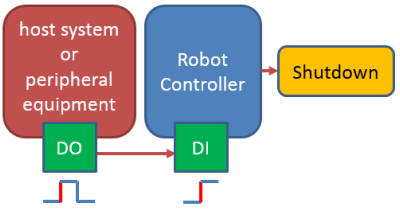

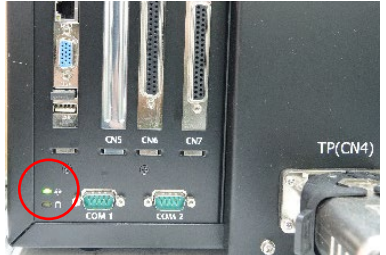

Standard Power OFF Procedure

Method 2. Through digital input signal (DI)

No	Procedure Description	Schematic/Photo
1	<p><u>By Operating teach pendant :</u></p> <p>(1) Please open the main menu of teach pendant, and then click sequentially:</p> <ol style="list-style-type: none"> 1. Start-up. 2. System Setting. 3. DIO Setting. <p>(2) Choose and set the DI signal for System Shutdown.</p>	<p><u>Step 1.</u></p>  <p><u>Step 2.</u></p> 
2	<p>Please execute an appropriate program of return for moving the robot back to the standby/home position, and then confirm the motion is completely stopped.</p> <p style="color: green;">Warning: Unless there is an emergency, please do not use the EMG-stop to stop the robot while the robot in the normal motion or still the program is running.</p>	
3	<p>Please press the emergency stop button.</p>	

Standard Power OFF Procedure

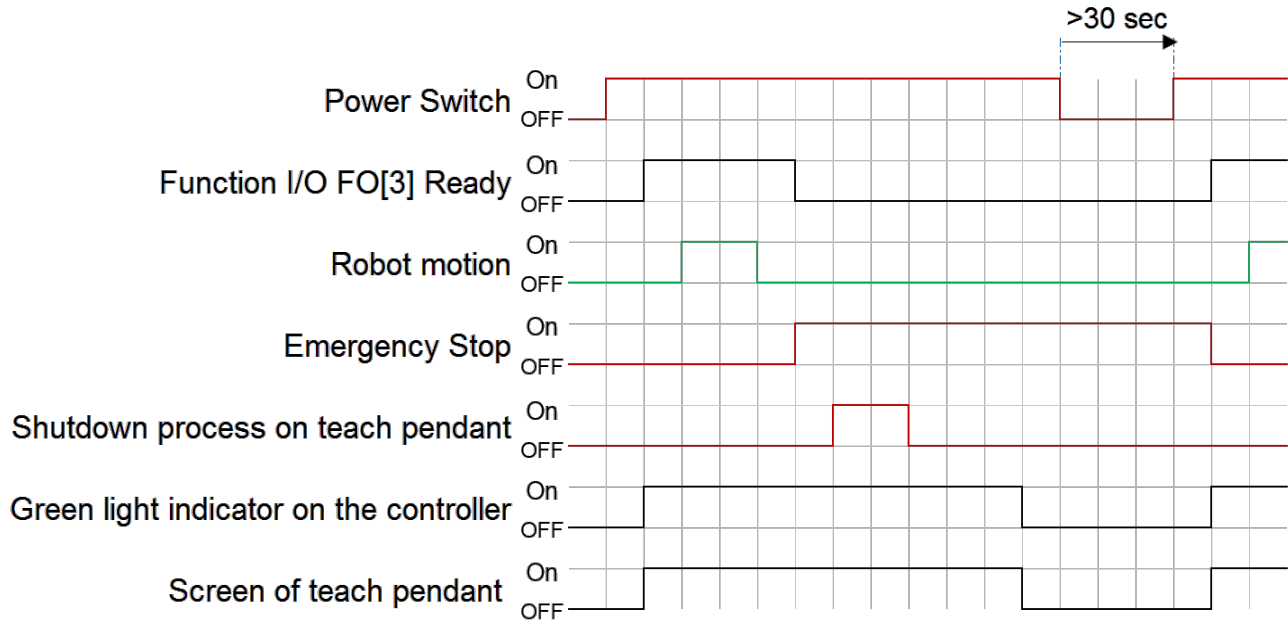
Method 2. Through digital input signal (DI)

No	Procedure Description	Schematic/Photo																																																								
4	<p>Please trigger a Digital Output signal (DO) through method of manual or programming, from the host system or peripheral equipment. This DO at least be "ON" status more than 500 millisecond, and then back to "Off" status.</p> <p>This DO signal connects and triggers the set controller DI in step1, and make the DI status changed from "Off" to be "ON", then the controller will execute System Shutdown.</p>	 <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Position</th> <th>Point</th> <th>I/O</th> <th>Timer</th> <th>Counter</th> <th>Alarm</th> <th>LogBoo</th> </tr> </thead> <tbody> <tr> <td>DI</td> <td>DO</td> <td>SI/O</td> <td>FIO</td> <td>PNS</td> <td>DIO Setting</td> <td></td> </tr> <tr> <td colspan="7">Base</td> </tr> <tr> <th>DI</th> <th>DI SIM</th> <th>DI Value</th> <th colspan="4">DI Comment</th> </tr> <tr> <td>DI20</td> <td><input type="checkbox"/></td> <td><input checked="" type="checkbox"/> On</td> <td colspan="4">系統關閉機</td> </tr> <tr> <td>DI21</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/> Off</td> <td colspan="4"></td> </tr> <tr> <td>DI22</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/> Off</td> <td colspan="4"></td> </tr> <tr> <td>DI23</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/> Off</td> <td colspan="4"></td> </tr> </tbody> </table>	Position	Point	I/O	Timer	Counter	Alarm	LogBoo	DI	DO	SI/O	FIO	PNS	DIO Setting		Base							DI	DI SIM	DI Value	DI Comment				DI20	<input type="checkbox"/>	<input checked="" type="checkbox"/> On	系統關閉機				DI21	<input type="checkbox"/>	<input type="checkbox"/> Off					DI22	<input type="checkbox"/>	<input type="checkbox"/> Off					DI23	<input type="checkbox"/>	<input type="checkbox"/> Off				
Position	Point	I/O	Timer	Counter	Alarm	LogBoo																																																				
DI	DO	SI/O	FIO	PNS	DIO Setting																																																					
Base																																																										
DI	DI SIM	DI Value	DI Comment																																																							
DI20	<input type="checkbox"/>	<input checked="" type="checkbox"/> On	系統關閉機																																																							
DI21	<input type="checkbox"/>	<input type="checkbox"/> Off																																																								
DI22	<input type="checkbox"/>	<input type="checkbox"/> Off																																																								
DI23	<input type="checkbox"/>	<input type="checkbox"/> Off																																																								
5	<p>Please confirm one of the below items is finished:</p> <ol style="list-style-type: none"> (1) The screen of teach pendant is off. (2) The green indicator light on the controller is turned off. 	 																																																								
6	<p>Please manually operate the power switch to lower side to turn off the power.</p>																																																									

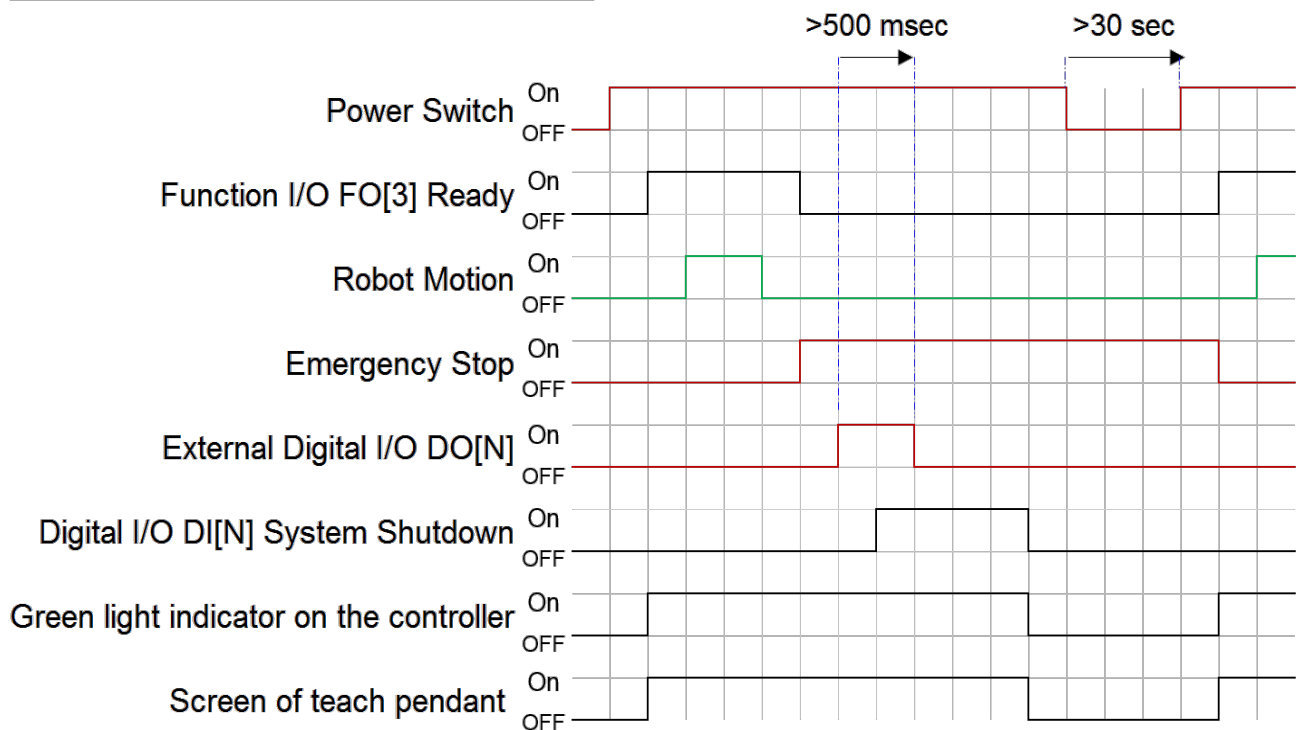
Warning: Please do not directly switch off the power of controller (Step-5) or the main power source in the factory before completing the above standard power turn off process (Step1-4). Otherwise, the controller may be damaged due to the abnormal power off procedure.

2.29.3. Standard Power On-OFF Sequence

Method 1. By Using teach pendant



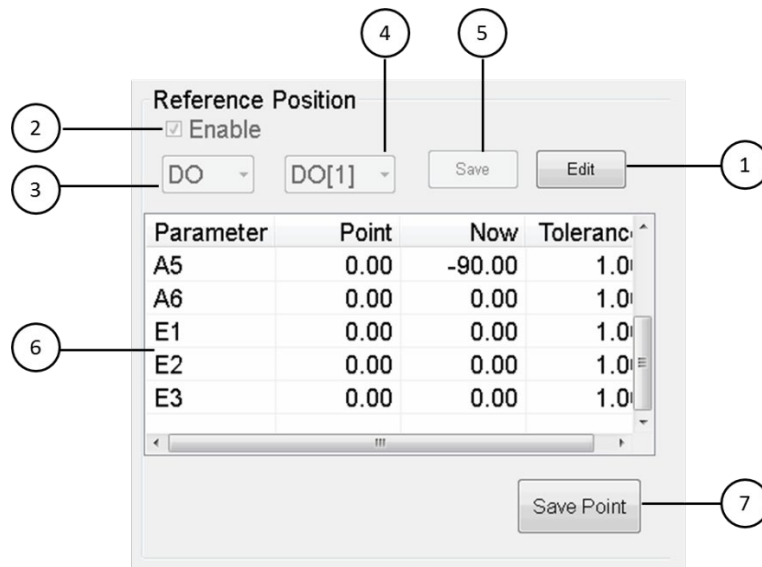
Method 2. Through digital input signal (DI)



2.30. Reference Point Function

Description

The reference point function allows setting of a reference point, including the information of angle of robots A1 to A6 (to A4 for Delta robots) and the information of the angle of external axis E1 to E3, and specifies one output. If the locations of the robot, external axis and the set point are currently within the range set, specifying the output will output signals. This function can be used to check whether the robot and external axis have reached the designated point, such as the origin or specific work points, etc.



Reference point function settings

No.	Description
1	Edit function
2	Enable function
3	Select output as DO or SO (Fieldbus output)
4	Select output pin
5	Save output related setting values
6	Show information on the point location and the current location; double-click on the field to set the tolerance range value for each axis check
7	Set and save the current locations of the robot and external axis as reference points

Operation Steps

1. Press Edit to edit settings
2. Check Enable to enable functions
3. Select output type as DO or SO
4. Select output pin
5. Save output related setting values
6. Double-click on list settings to check the tolerance range value
7. Move to the point to set and press Save Point
8. Write the program and monitor designated output; it is used for point-in-position inspection

3. Initial Settings

3.1. Check Parameters

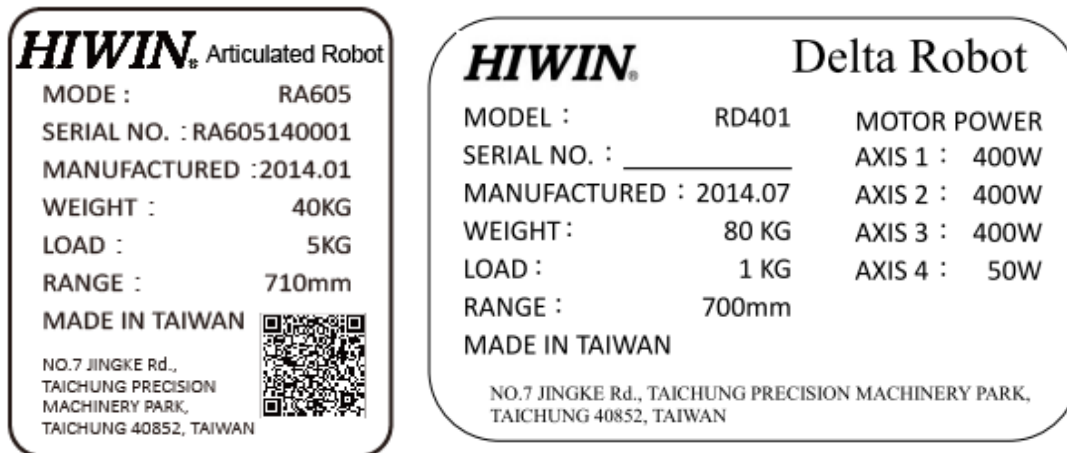
Description

The correct robot program data must be loaded. During parameter check, the loaded robot data must match with the data of the model plate.

If loading the new data is required, the status of the robot data must fully match with the HRSS. This is to ensure that when the data is applied, it can be submitted with the HRSS.

DANGER

If the wrong data is loaded, the robot should not be operated! Failure to take these measurements could lead to serious injury, death or equipment damage



Model Plate(Left:RA605, Right:RD401)

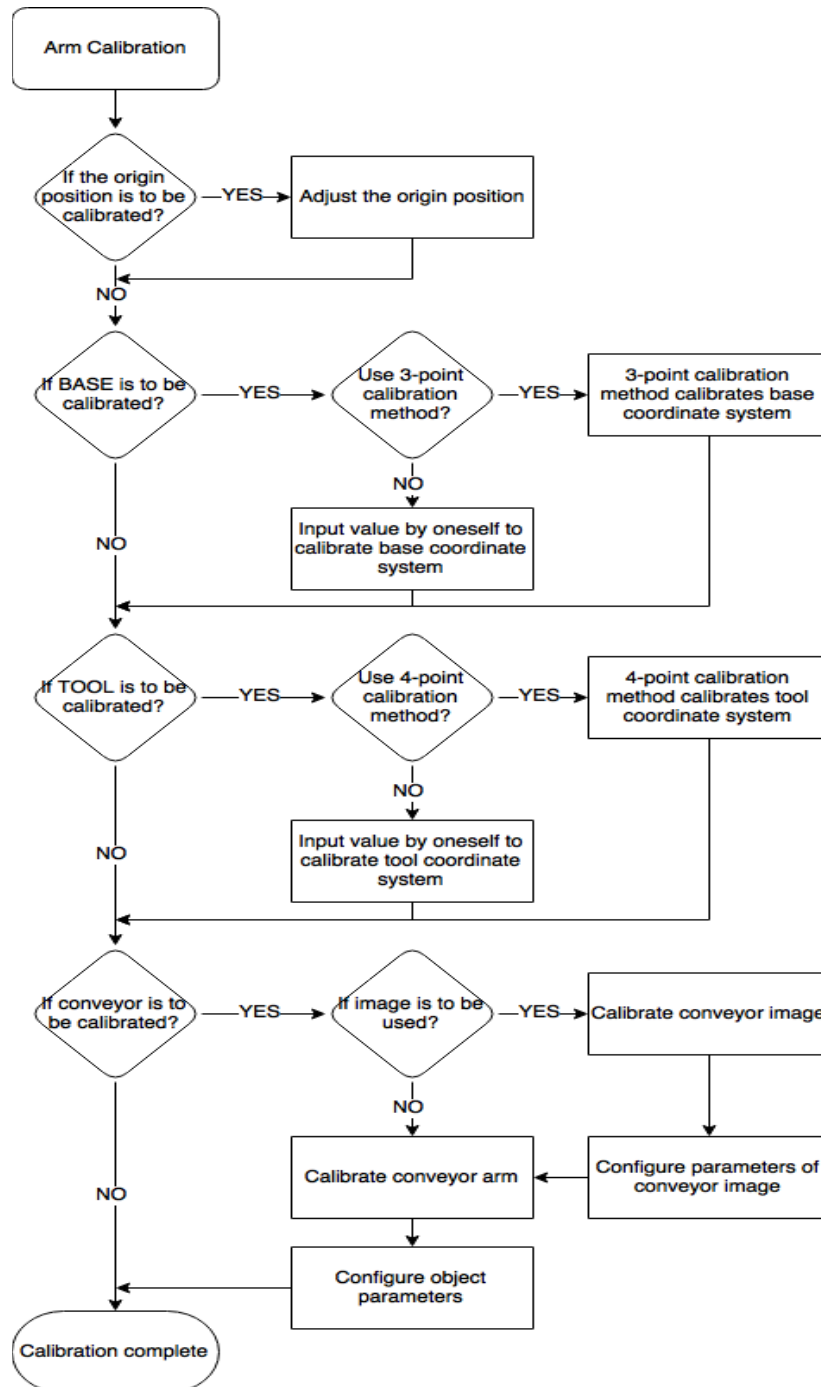
Operation steps

Main Menu>Help>About

3.2. Calibration Flow

Fig. 5.2 is the calibration flowchart of robotic arm. According to the user's requirements, they are: Adjusting the origin position (5.3) → Calibrate the base coordinate system (5.4.1,5.5.1) → Calibrate the tool coordinate system (5.4.2,5.5.2) → Calibration of conveyor image (5.6.1) → Calibration of conveyor arm (5.6.2) → Configure the parameters of conveyor image (5.7.1) → Configure the parameters of conveyor object (5.7.2).

The above mentioned calibration steps will be introduced in the subsequent sections.

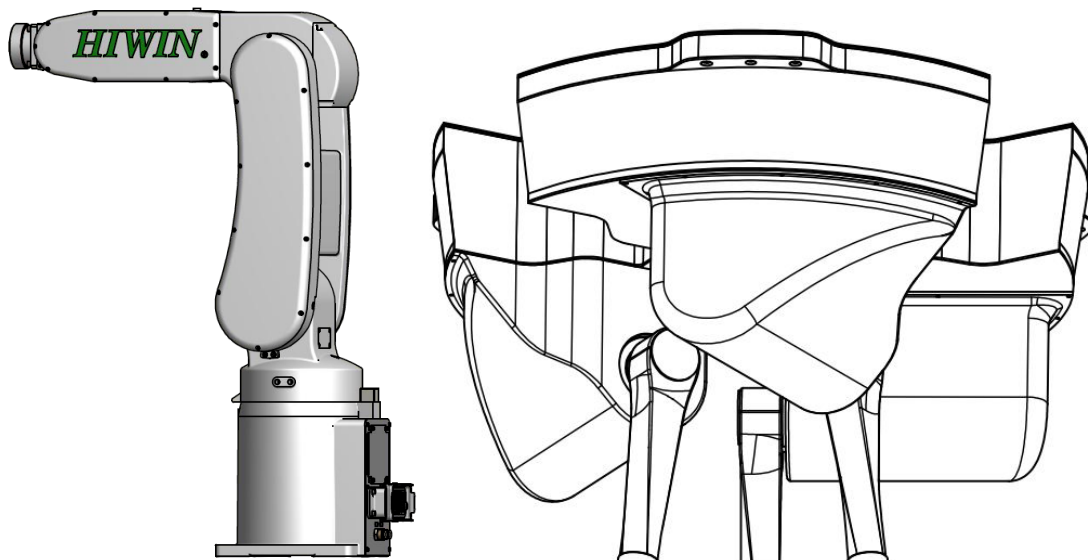


Calibration Flowchart of Robot

3.3. Adjust Origin Position of Hardware Mechanism

Overview

Each robot must be mastered. The robot can make Cartesian motion only after being mastered and moved to the programmed position. The mechanical position of the robot will be made consistent with the encoder during mastering. The robot must be placed on a defined mechanical position, which is the mastered position. The encoder value of each axis will be saved.



Adjust the approximate position of origin

Origin calibration is divided into two ways: [Zero position] and [Calibration],

(1) Clicking on [Zero position] will restore the default factory origin location; the posture of the arm needs to be close to the origin of the mechanism.

(2) Clicking on [Calibration] will change the default factory origin location.

Select different origin calibration methods according to the different situations below:

Identifying origin calibration situations and practices

Situation	Remark
The value of motor position is lost after maintenance such as replacement of a motor.	Please execute calibration
After replacement of gear unit.	Please execute calibration
After a collision.	Please execute calibration
If the absolute position is missing after replacing the battery.	Please execute Zero position

CAUTION

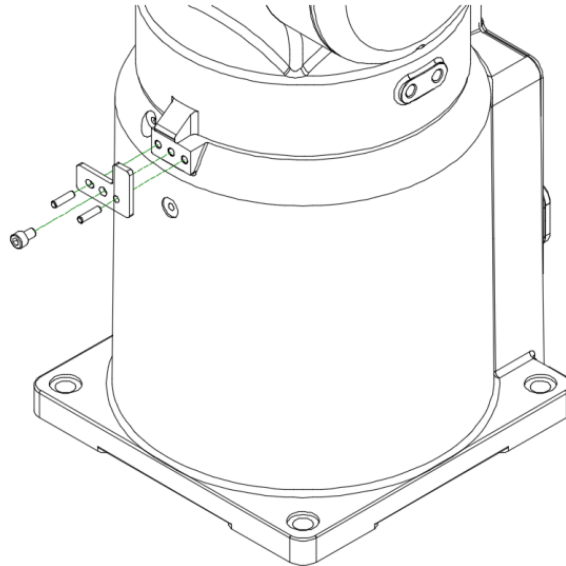
If the posture of the simulated robot arm is outside the limit range and cannot move, please execute [Zero Position] first

Hardware mechanism origin adjustment

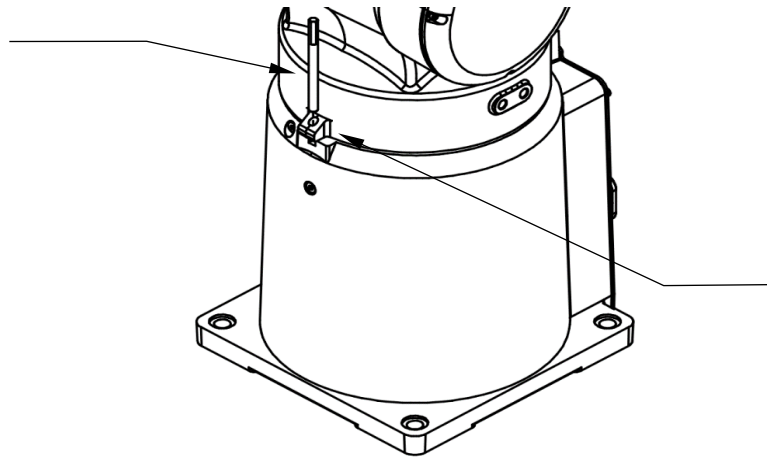
6-axis robot

1. Setting the origin of the first axis

Calibration of the first axis is to use the calibration pin to fix the calibration sheet metal on the first axis, reduce the rotation speed of the first axis to the slowest until the first axis is against the calibration sheet metal; this completes the calibration of the first axis, as shown in the figure below.



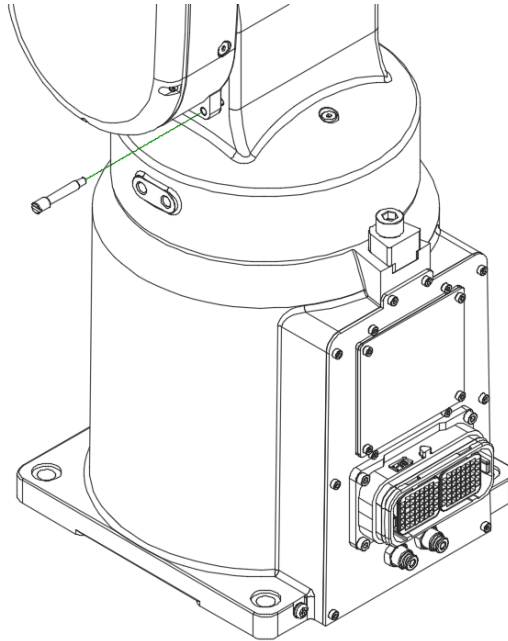
Schematic diagram of origin calibration for the first axis (the version for products shipped before 2021)



Schematic diagram of origin calibration for the first axis (the version for products shipped after 2021)

2. Setting the origin of the second axis

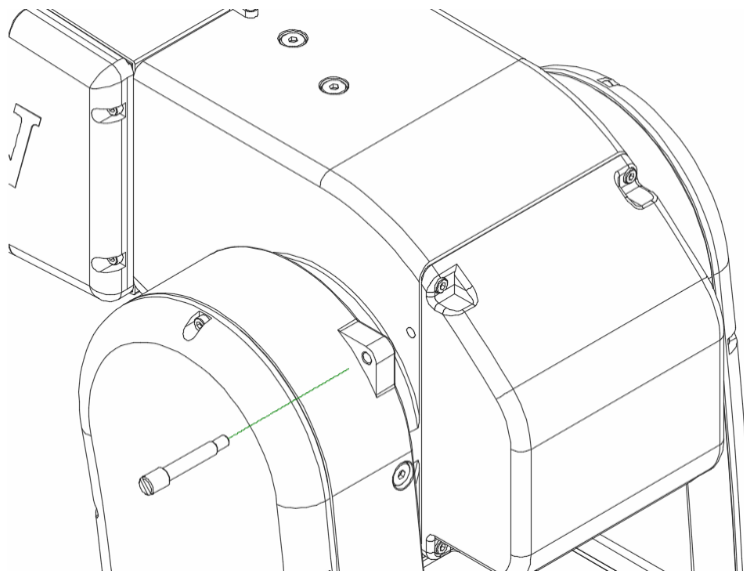
Calibration of the second axis is to reduce the rotation speed of the second axis to the slowest until the second axis matches the calibration hole of the first axis and can be positioned using the calibration rod. This completes the calibration of the second axis, as shown in the figure below.



Schematic diagram of origin calibration for the second axis

3. Setting the origin of the third axis

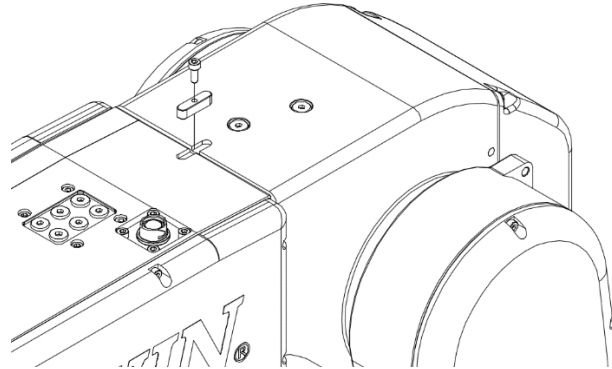
Calibration of the third axis is to reduce the rotation speed of the third axis to the slowest until the third axis matches the calibration hole of the second axis and can be positioned using the calibration rod. This completes the calibration of the third axis, as shown in the figure below.



Schematic diagram of origin calibration for the third axis

4. Setting the origin of the forth axis

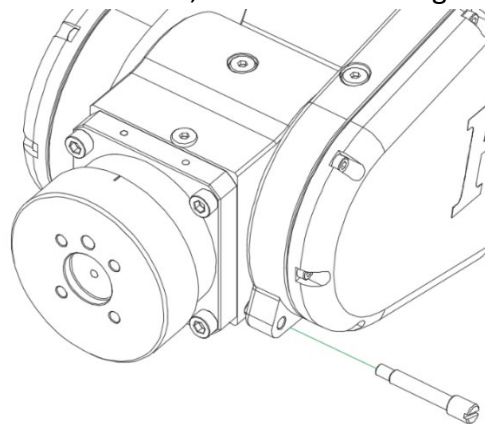
Calibration of the forth axis is to reduce the rotation speed of the forth axis to the slowest until the forth axis matches the calibration hole of the third axis and can be positioned using the calibration key. This completes the calibration of the forth axis. After the calibration, screws can be used to remove the calibration key out of the slot, as shown in the figure below.



Schematic diagram of origin calibration for the forth axis

5. Setting the origin of the fifth axis

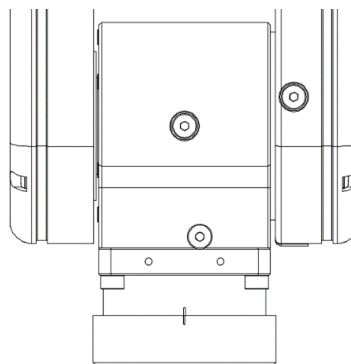
Calibration of the fifth axis is to reduce the rotation speed of the fifth axis to the slowest until the fifth axis matches the calibration hole of the sixth axis and can be positioned using the calibration rod. This completes the calibration of the fifth axis, as shown in the figure below.



Schematic diagram of origin calibration for the fifth axis

6. Setting the origin of the sixth axis

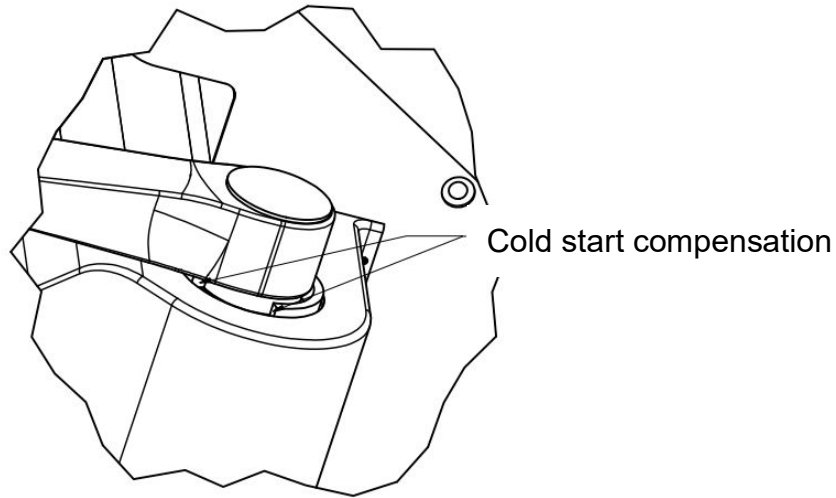
Calibration of the sixth axis is to reduce the rotation speed of the sixth axis to the slowest until the two calibration marks of the six axis matches. This completes the calibration of the sixth axis, as shown in the figure below.



Schematic diagram of origin calibration for the sixth axis

1. Setting the origin of axis one~three

Release the brake manually and then move the arm joint to the top (bottom); use the hardware brake.



Schematic diagram of calibration bearing surface

2. Setting the origin of the forth axis

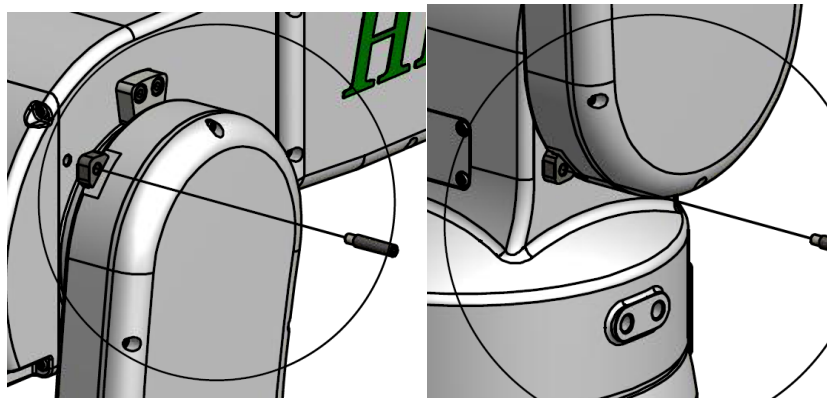
The forth axis has no adjustment mechanism marks; currently, the end device is mainly used as the origin position. Please set the position according to the usage situation.

3.3.1. Instructions for Restoring Default Factory Origin Positions

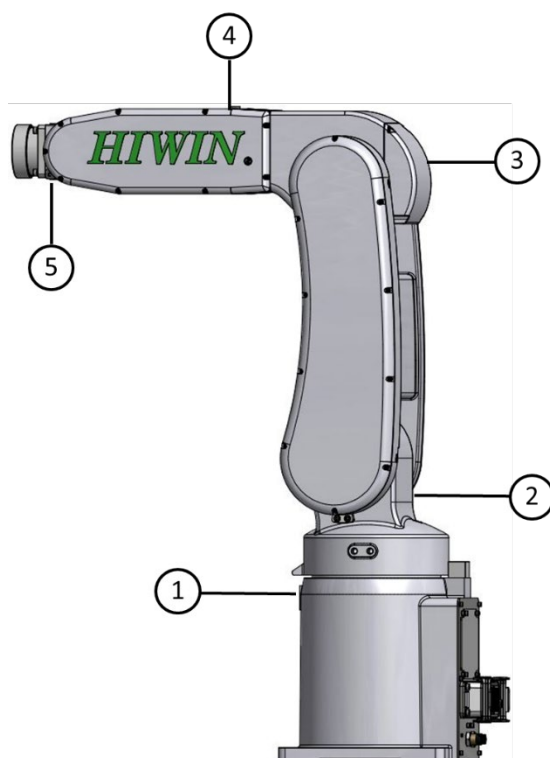
3.3.1.1. Mastering Method (6-axis robot)

Description

Move each axis, so that it can overlap with the mastering mark.



Move an axis to the mastering position



Mastering marks on the robot

⚠ CAUTION

Based on the model number, the position of the mastering marks could be slightly different from the illustration. For origin calibration method and image, please refer to the manual of each model.

Prerequisite

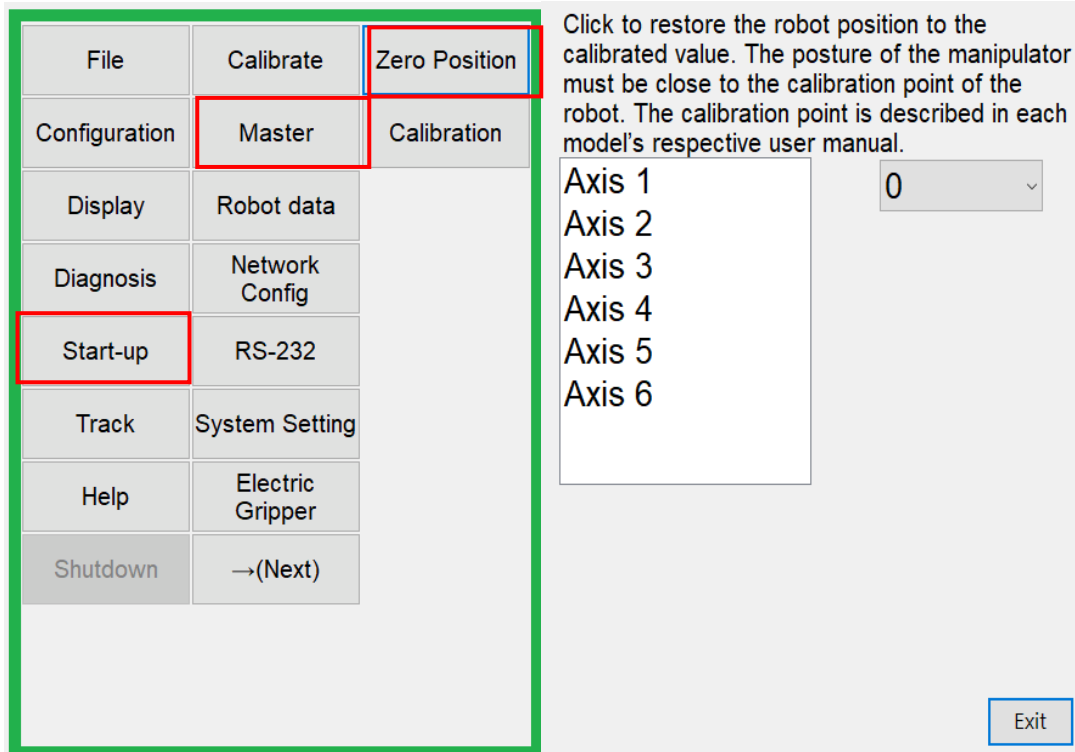
The “jog key” has been activated.

T1 mode

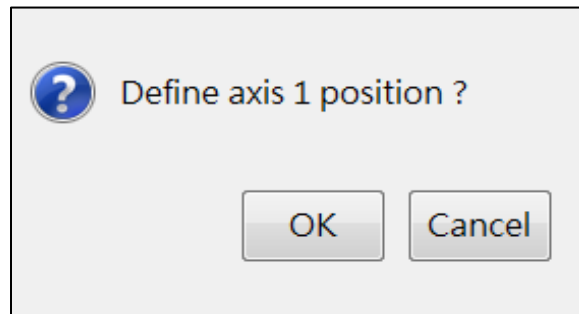
Expert group

Operation steps

1. Select the axis as the coordinate system for the jog keys. (Please refer to Section 2.13.5)
2. Hold the Enabling Switch. The axis A1 to A6 will be displayed beside the jog keys.
3. Press the + or – button, so that the axis moves to the positive or negative direction.
4. Start to jog from the axis A1, so that it can overlap with the mastering mark.
5. After mastering, click Main Menu>Start-up>Master>Zero Position. Zero Position or Calibration button selection; please refer to the identifying the origin calibration situation and practice table above.



- Double-click the axis to position in the list in [Zero Position] and the figure below will appear; press the [Cancel] button to cancel execution, or press [OK] to complete the origin position setting of the first axis. The adjustment method for this origin position is based on the origin set by the manufacturer and performing angle conversion. The angle at this time will be closer to the calibration position; for example, the first axis of R605-710-GC should be close to 0 degrees.



Zero Position reset selection screen

- And so on for 2nd - axis to 6th - axis.
- After completing the clearing, press Home button to confirm if the angle is correct, if the position shown on the screen is different from the actual position, please clear the position again.

3.3.1.2. Mastering Method (Delta Robot)

Description

Move each axis, so that it can overlap with the mastering mark.

⚠ CAUTION

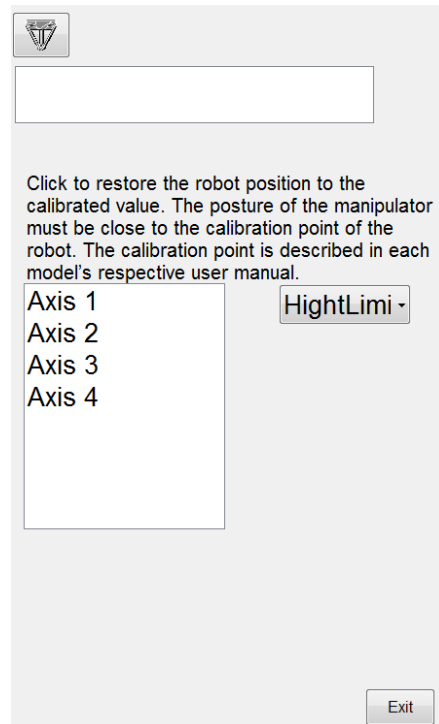
Depending on model number, the position of the calibration surface may be slightly different from the illustration. For origin calibration method and image, please refer to the manual of each model.

Prerequisite

Expert group

Operation steps

1. Manually release the robot brake.
2. Move the arm (Axis-1) to the top (lowest) side.
3. Use the hardware brake.
4. Select HightLimit or LowLimit, and click Axis 1 twice.
5. Appear “Define axis 1 position?”
6. Press OK to complete the origin position setting for the first axis; the adjustment method for this origin position is based on the origin set by the manufacturer and performing angle conversion. The angle at this time will be closer to the calibration position; for example, the first axis of RD403-1300 should be close to -65.24 degrees.
7. Follow the previous procedures for second and third axis.
8. Once cleared, press the Home button to ensure the angle is correct. If the position on the figure is different from the actual position, clear the position again.



Zero Position interface

3.3.2. Reset Origin Position Instructions

Prerequisite

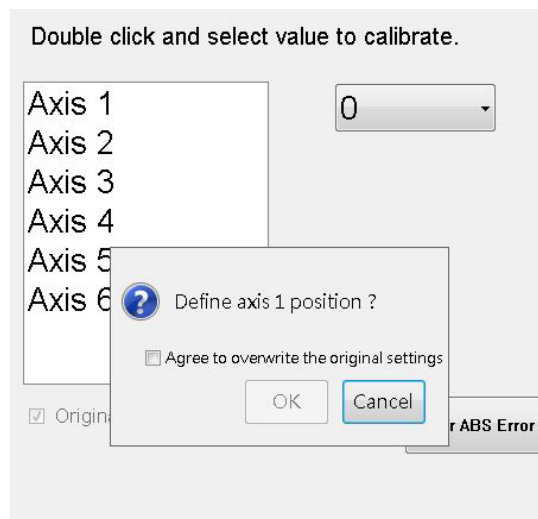
The “jog key” has been activated.

T1 mode

Expert group

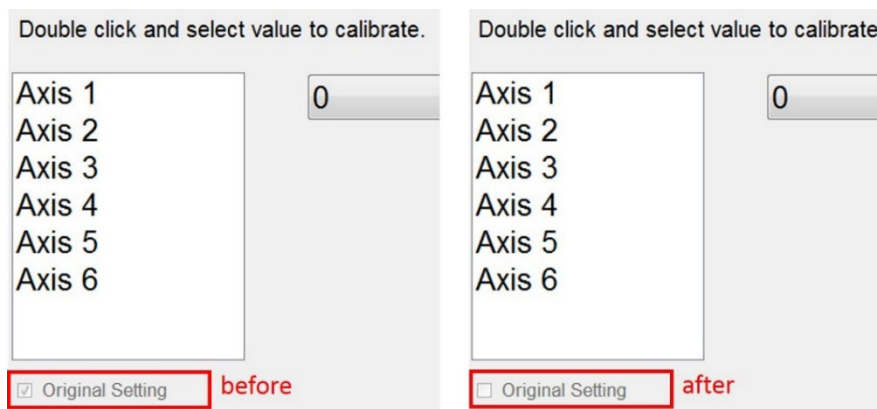
Operation steps

1. Adjust Origin Position of Hardware Mechanism (Please refere to Section 3.3.1.1 and 3.3.1.2)
2. Main Menu >Start-up>Master>Calibration.
3. Click Axis 1 twice, and appear “Define axis 1 position?”, agree and operate the step. Ok button can press after assenting and checking the item, this adjustment method is to save the current encoder value and overwrite the default.



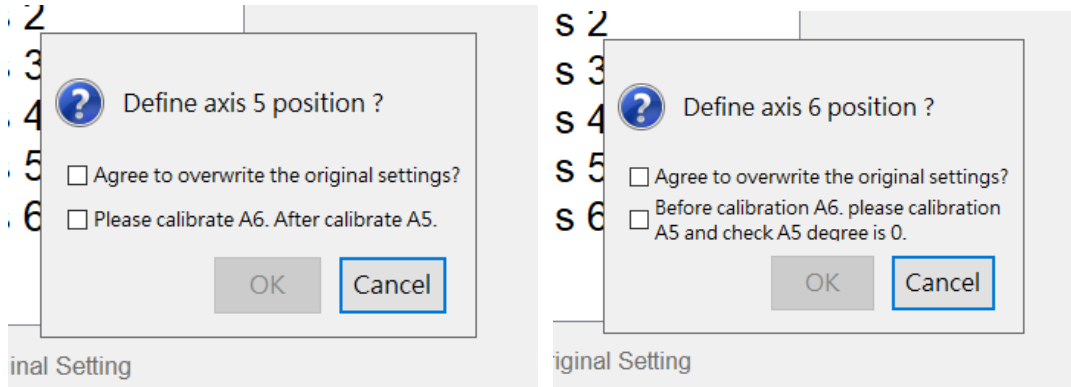
Calibration confirm interface

4. Press the OK button to set the position. Confirmed cell of original setting has been canceled, if the step has been finish.



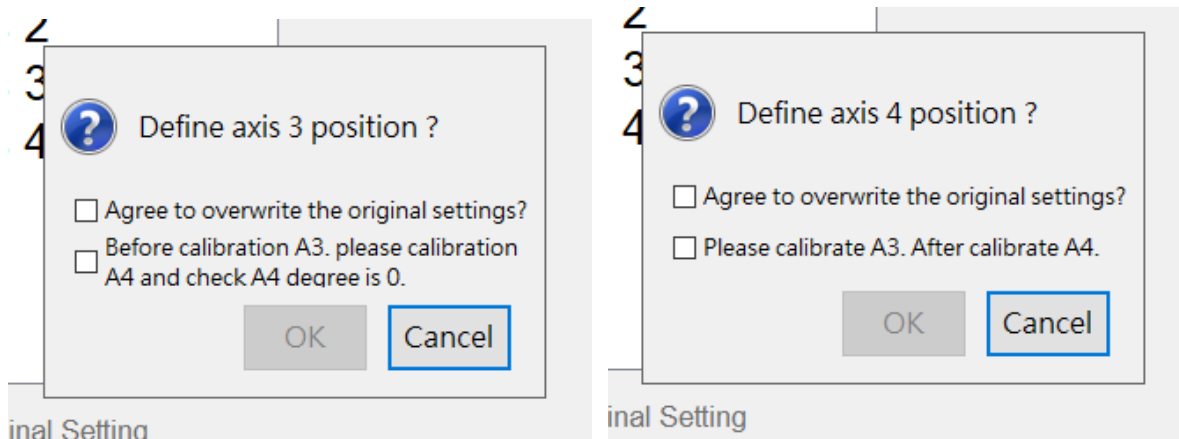
5. Follow the previous procedures for second and sixth axis.
6. Once operated, press the Home button to ensure the angle is correct. If the position on the figure is different from the actual position, reset the position again.

※Please pay attention to Axis 5 and Axis 6 of the six axes; calibration of Axis 5 must be completed first, then calibrate Axis 6.



Calibration confirmation window for Axis 5 and Axis 6 of the six axes

※Please pay attention to Axis 3 and Axis 4 of the four axes; calibration of Axis 4 must be completed first, then calibrate Axis 3.



Calibration confirmation window for Axis 3 and Axis 4 of the four axes

! WARNING

Before operating this step, please pay attention to the following parts:

6-axis robot:

To calibrate sixth axis, please make sure fifth axis angle is zero and encoder value is in the single turn, if it does not match, fifth axis must be calibrated first.

4-axis robot:

To calibrate third axis, please make sure forth axis angle is zero and encoder value is in the single turn, if it does not match, forth axis must be calibrated first.

If the operation is not accurate enough, it may cause deviation in absolute position and lose the default.

3.3.3. Reset Encoder Error

Description

When the encoder occurs an error, it can be reset. If the error can not be cleared after reset, please check if there is any problem for the circuit, battery and driver. If the error still exists, please contact HIWIN. After reset, the encoder must be cleared.

Operation steps

1. Reset origin position method. (Please refer to Chapter3.3.2)

3.4. Calibrate Coordinates (Six-axis Robot)

3.4.1. Base calibration

Description

During base calibration, the user assigns a Cartesian coordinate system (BASE coordinate system) to a work surface or the work piece. The BASE coordinate system has its origin at a user-defined point.

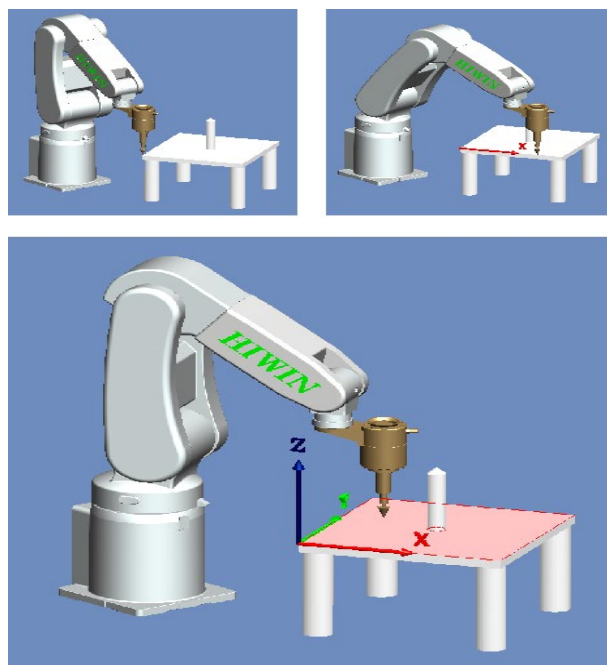
CAUTION

If a workpiece has been installed on the mounting flange, the calibration described here will not apply.

Advantages of base calibration:

1. TCP can be jogged along the work plane or the edge of workpiece.
 2. Points can be taught relative to base. If the base has to move, for example, because the work plane is moved, these points will be moved as well, but no need to be taught again.
- 32 base coordinates can be saved at most. Variable: BASE [0...31].

3.4.1.1. 3-point Method



3-point Method

Prerequisite

Install a calibrated tool on the mounting flange.

T1 mode



Operation steps

1. In the main menu select Start-up > Calibrate > Base.
2. In the pull-down menu, given a number and a name for base coordinate system. Confirm with OK button.
3. Select a number for the base to be calibrated, and then press [Measure].
4. Give a name for the base to be calibrated.
5. Move TCP to the origin of new base coordinate. Click [Measure], and then confirm by pressing the OK button.
6. Move TCP to a point on positive X axis of new base coordinate. Click [Measure], and then confirm by pressing the OK button.
7. Move TCP to a point with positive Y on the XY plane. Click [Measure], and then confirm by pressing the OK button.
8. Press the OK button after completed. The data will be saved.

3.4.1.2. Enter Value

Description

Known the following values, for example, obtain from CAD:

1. Distance between the base origin and global origin
2. Rotation for base coordinate relative to global coordinate

T1 mode

Prerequisite

Known X, Y, Z, A, B and C relative to the global coordinate system

T1 mode

Operation steps

1. In the main menu select Start-up > Calibrate > Base
2. Select a number for the base to be measured.
3. Select the coordinate you want to enter, and then press [Set Value].
4. Give a name for the base to be measured.
5. Enter the value.
6. Press the OK button after completed. The data will be saved.

3.4.2. Tool calibration

Description

When the tool is calibrated, the user will give a set of Cartesian Coordinates (Tool Coordinate System) to the tool mounted on the flange. The tool coordinate system has its origin at a user-defined point. This point is called as TCP (Tool Center Point). Usually, TCP is located at the working point of the tool.

CAUTION

The calibration method described here must not be used for a fixed tool.

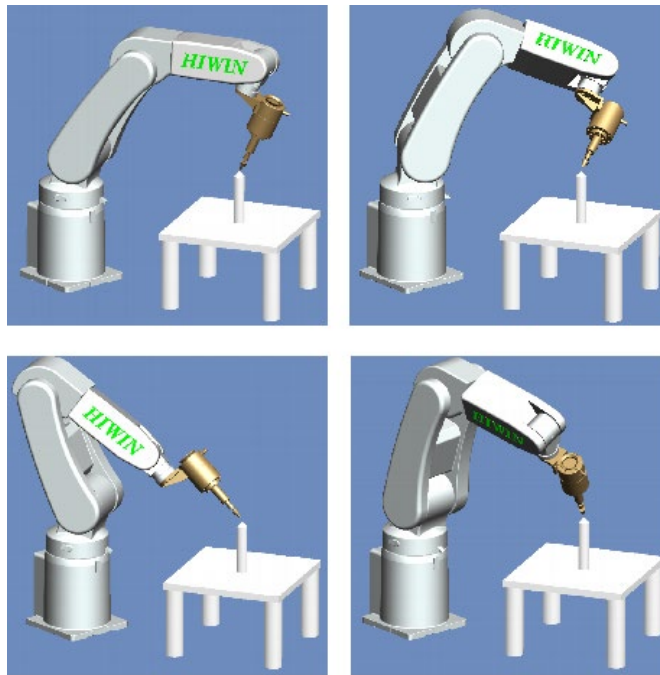
3.4.2.1. 4-Point Method

Description

The TCP of the tool to be calibrated is moved to a reference point from 4 different directions. The reference point can be freely selected. The robot control system calculates the TCP from the different flange positions.

CAUTION

The 4 flange positions at the reference point must be sufficiently apart from one another.



4-point method

Prerequisite

1. Install the tool to be calibrated on the mounting flange.
2. T1 mode

3. Select a fixed reference point position, it is recommended that this reference point is a shape that is easy to recognize, for example: cuspidal point.

Operation steps

1. In the main menu select Start-up > Calibrate > Tool
2. Select a tool number from the pull-down menu for the tool to be measured, and then press [Measure].
3. Give a name for the tool to be calibrated.
4. Use TCP to move to the configured reference point. Click [Measure], if the position of reference point is to be confirmed, please use OK button for confirmation, otherwise, use No or Cancel to cancel the operation.
5. Use TCP to replace another posture position of arm, move to the reference point. Click [Measure], if the position of reference point is to be confirmed, please use OK button for confirmation, otherwise, use No or Cancel to cancel the operation.
6. Repeat Step 5 twice.
7. Press confirmation button after completion. The data displays the reference point of new base coordinate system and is saved, the changed coordinate system can also be observed from the 3D simulation window.

3.4.2.2. Enter Value

Description

Tool data can be manually entered.

Possible data source:

1. In the CAD diagram file, acquire the size information of tool.
2. Tool size from the measurement of actual object.
3. Instruction manual of tool manufacturer.

Prerequisite

Known X, Y, Z, A, B, C distance dimension as relative to the flange coordinate system.

T1 mode

Operation steps

1. In the main menu select Start-up > Calibrate > Tool
2. Select a number for the tool to be measured in the pull-down menu.
3. Select the coordinate axis you want to enter, and then press [Set Value].
4. If it is the first time entering the value, then a name shall be given to the tool to be measured.
5. Enter a value.

6. Press confirmation button after completion. The data displays the reference point of new base coordinate system and is saved. The changed coordinate system can also be observed from the 3D simulation window.

3.5. Calibrate Coordinates (Delta Robot)

When Delta is used to set the coordinate, the base coordinate can be set the same with the vision and conveyor coordinates. It can be convenient to set the calibration of the position in future. There are two methods to set the base coordinate, the first one is the Three-point Method and the second one is to use the coordinates from CAD to enter the values of X, Y, Z, A, B and C coordinates.

3.5.1. Calibrate Base Coordinates

Description

During base calibration, the user assigns a Cartesian coordinate system (BASE coordinate system) to a work surface or the work piece. The BASE coordinate system has its origin at a user-defined point.

CAUTION

If a workpiece has been installed on the mounting flange, the calibration described here will not apply.

Advantages of base calibration:

1. TCP can be jogged along the work plane or edge of workpiece.
 2. Points can be taught relative to base. If the base must be moved, for example, because the work plane is moved, the points will be moved as well, and no need to be taught again.
- 32 base coordinates can be saved at most. Variable: BASE[0...31].

3.5.1.1. 3-point Method

Prerequisite

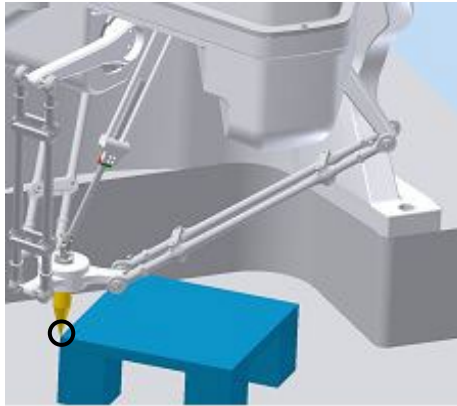
Install a calibrated tool on the mounting flange.

T1 mode

Operation steps

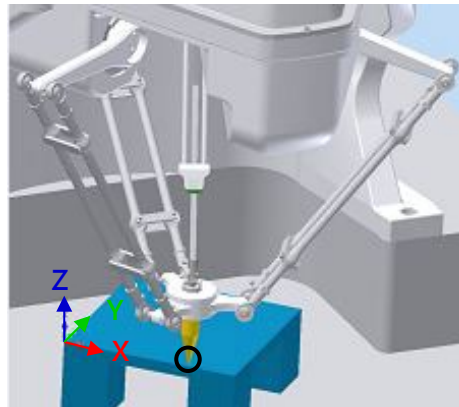
1. In the main menu select Start-up > Calibrate > Base
2. A number and a name is given to the Base Coordinate System, and then press the CONTINUE button to confirm.
3. Select a number for the base to be calibrated, and then press [Measure].
4. Give a name for the base to be calibrated.

5. Move TCP to the origin of new base coordinate. Click [Measure], and confirm by pressing the OK button.



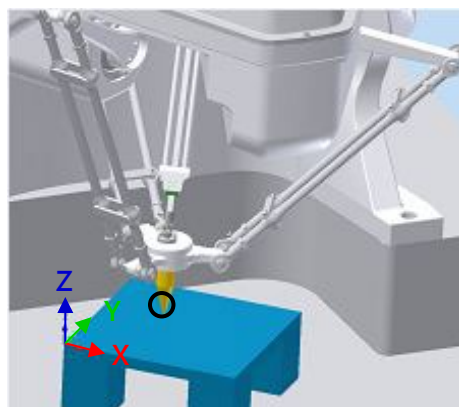
Set the origin of base coordinate system

6. Move TCP to a point on positive X axis of new base coordinate. Click [Measure], and confirm by pressing the OK button.



Set a point on positive X axis

7. Move TCP to a point with positive Y on the XY plane. Click [Measure], and confirm by pressing the OK button.



Set a point with positive Y on XY plane

8. Press the OK button after completed. The data will be saved.

3.5.1.2. Enter Value

Description

Values are known, from CAD, for example:

1. Distance between the base origin and global origin
2. Rotation for base coordinate relative to global coordinate

T1 mode

Prerequisite

X, Y, Z, A, B and C relative to the flange coordinate system is known

T1 mode

Operation steps

1. In the main menu Select Start-up > Calibrate > Base
2. Select a number for the base to be measured.
3. Select the coordinate you want to enter, and then press [Set Value].
4. Give a name for the base to be measured.
5. Enter the value.
6. Press the OK button after completed. The data will be saved.

3.5.2. Calibrate Tool Coordinates

Description

When the tool is calibrated, the tool coordinate system has its origin at a user-defined point. This point is called as TCP (Tool Center Point). Usually, TCP is located at the working point of the tool.

CAUTION

The calibration method described here must not be used to a fixed tool.

Advantage of tool calibration:

1. The tool can rotate along the TCP. The position of TCP will not change.
2. Program running: The track along TCP keeps the programmed velocity.
16 tool coordinates can be saved at most. Variable: TOOL [0...15]).

The following data will be saved:

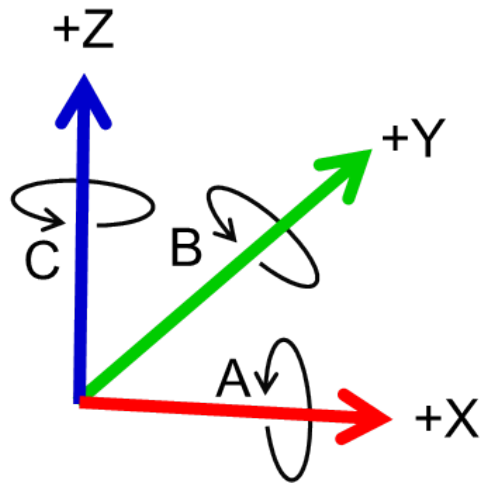
X, Y, Z:

The origin of tool coordinate, relative to flange coordinate

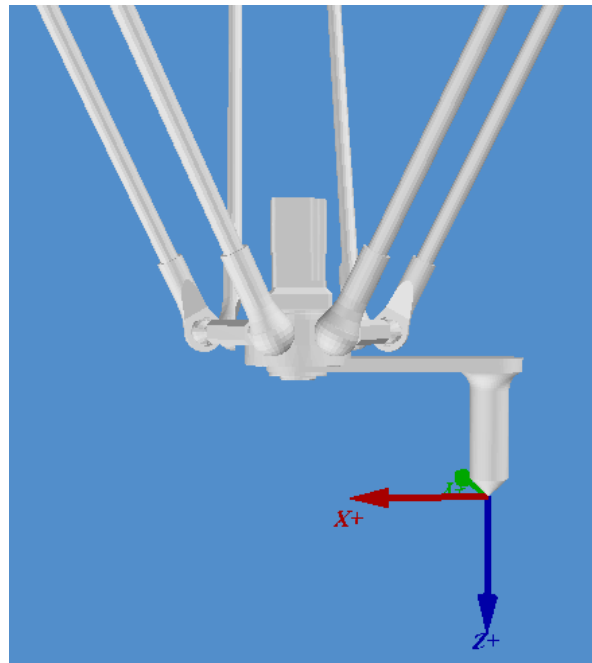
A, B, C:

The rotation of tool coordinate, relative to flange coordinate

- X: X coordinate
- Y: Y coordinate
- Z: Z coordinate
- A: Rotate along X coordinate
- B: Rotate along Y coordinate
- C: Rotate along Z coordinate



Flange coordinates



TCP calibration principle

3.5.2.1. 3-point Method

Description

The TCP of the tool to be calibrated is moved to a reference point from 3 different directions. The reference point can be freely selected. The robot control system calculates the TCP from the different flange positions. Achieve the tool coordinates (X, Y, Z, C value, where value of C [J4 rotational angle] is the C value of the first calibrated value).

CAUTION

The 3 flange positions at the reference point must be sufficiently apart from one another.

Prerequisite

1. Install the tool to be calibrated on the mounting flange.
2. T1 mode
3. Select a fixed reference point position, it is recommended that this reference point is a shape that is easy to recognize, for example: cuspidal point.

Operation steps

1. In the main menu select Start-up > Calibrate > Tool
2. Select a tool number from the pull-down menu for the tool to be measured, and then press [Measure].
3. Give a name for the tool to be calibrated.
4. Use TCP to move to the configured reference point.
5. Click [Measure] to confirm the first calibration point, if the position of reference point is to be confirmed, please use OK button for confirmation, otherwise, use No or Cancel to cancel the operation.
6. Use TCP to replace another posture position of arm, move to the reference point. Click [Measure] to confirm second point, if the position of reference point is to be confirmed, please use OK button for confirmation, otherwise, use No or Cancel to cancel the operation.
7. Repeat Step 6 to confirm third point.
8. Press confirmation button after completion. The data displays the reference point of new tool coordinate system and is saved, the changed coordinate system can also be observed from the 3D simulation window.

3.5.2.2. Enter Value

Description

The tool data can be manually entered.

Possible data source:

1. CAD
2. External measurement tool
3. Description from tool manufacture

Prerequisite

Known X, Y, Z, A, B and C relative to flange coordinate

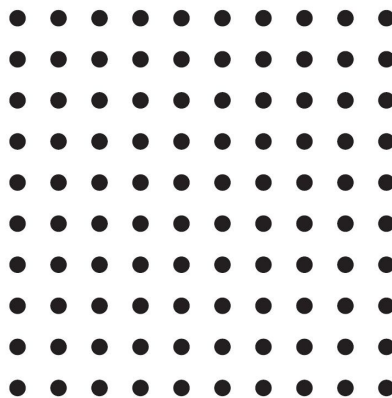
T1 mode

Operation steps

1. In main menu select Start-up > Calibrate > Tool
2. Select a value for the tool to be measured.
3. Select the coordinate axis you want to enter, and then press [Set Value].
4. Give a name for the tool to be measured.
5. Enter a value.
6. Press the OK button after completed. The data will be saved.

3.6. Conveyor Calibration

3.6.1. Conveyor Image Calibration



Calibration plate

- The Image System needs to decide the origin of image coordinate, the ratio of pixel length and the reading of conveyor encoder via the calibration plate (Figure above).

- Please ensure that there is a reading on the conveyor encoder before performing the calibration.
- Figure below is an illustration of picking. From the direction of the conveyor, the upstream is before the robot, and the downstream is after the robot.

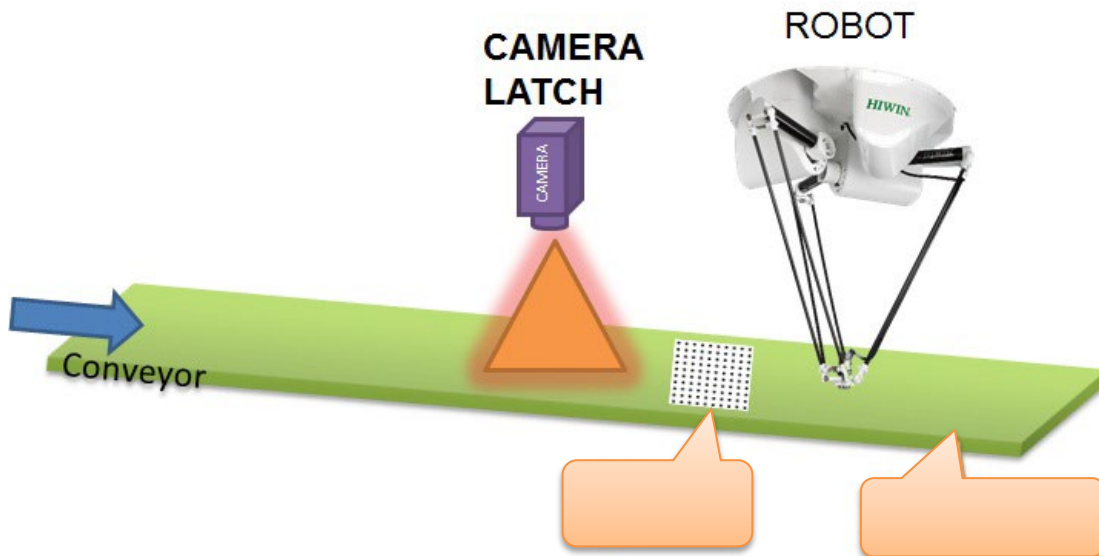
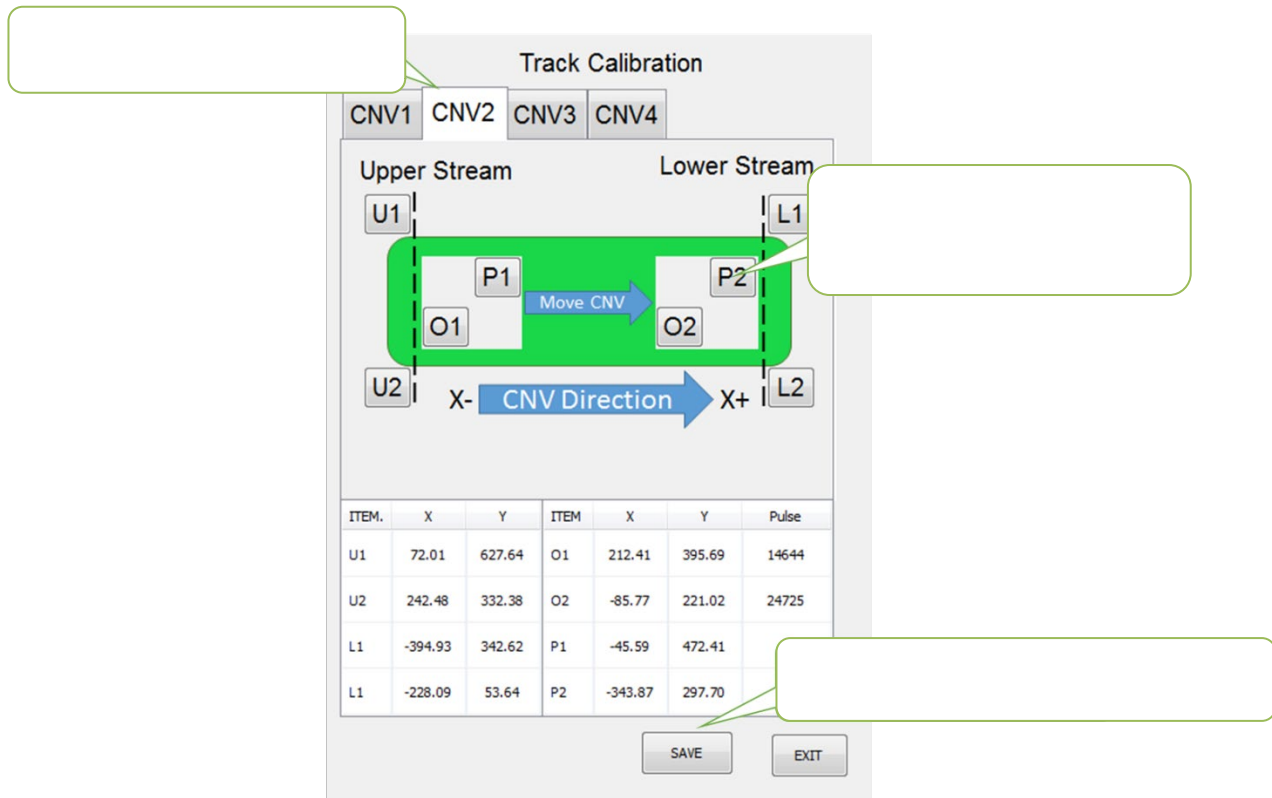


Illustration of Delta and CCD with conveyor

3.6.2. Conveyor Arm Calibration

1. After the images are calibrated, the upstream and downstream of the arm is moved by the conveyor without moving the calibration plate so that the arm can relate with the image coordinate and conveyor via the calibration plate.
2. Install the calibration rod on the arm end when performing the calibration.
3. Click the Main Menu>Track>Calibration.
4. Select the number of the conveyor.
5. When calibration plate is located at the upstream of the arm, and the center of the calibration rod is aligned with the origin of the calibration plate (the same with the vision origin) and press O1, and then aligned with the calibration point of the calibration point and press P1 (If it is in the mode of Sensor Latch, P1 can be omitted.).
6. The calibration plate is moved to the downstream of the arm, the center of the calibration rod is aligned with the origin of the calibration plate (the same with the vision origin) and press O2, and then aligned with the calibration point of the calibration plate and press P2 (The value of the encoder needs to be positive.) (If it is in the mode of Sensor Latch, P2 can be omitted.).
7. The arm is moved to the left and right limit of upstream, pressing U1 and U2 (U1 and U2 cannot be the same point.) respectively; the arm is moved to the left and right limit of downstream, pressing L1 and L2 (L1 and L2 cannot be the same point.) respectively, as shown in below.

8. If the counting direction of CNT is found as counted downward when the conveyor moves during the calibration process, open the function of REV column and make that conveyor counted backward, and then change to count upward.
9. Press "CLEAR" to zero CNT of all conveyors and recount.



Arm calibration screen

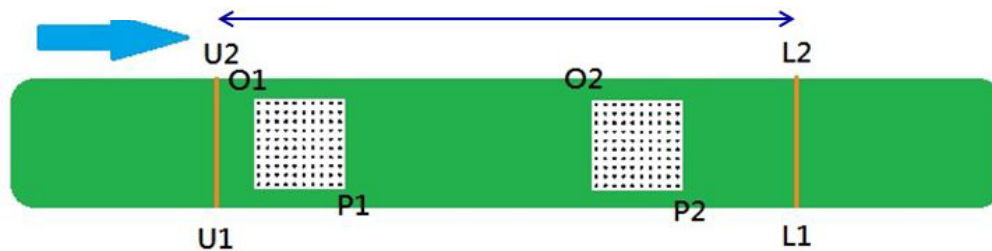
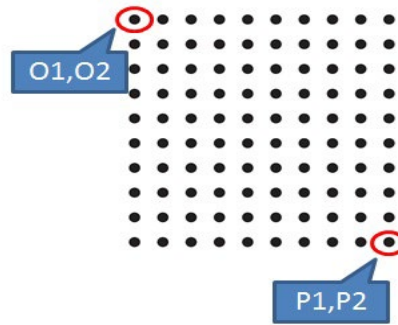


Illustration to calibrate arm position



Set O and P screen

3.7. Conveyor Parameters

3.7.1. Image Parameters

1. Main Menu>Track>Vision
2. Conveyor number: The information set in the Image Parameter screen (Figure below) will be recorded according to the number of the conveyor. When setting the parameters, select the number first. After setting completed, press the Save button.
3. Calibration information: Before using the conveyor track, the calibration point is set to correspond to the coordinate position (mm) in the image system, which is the distance from the origin of the calibration plate to calibration point, as well as the counting value of the encoder when the image is calibrated.
4. Connection: IP for the system and the connection port.

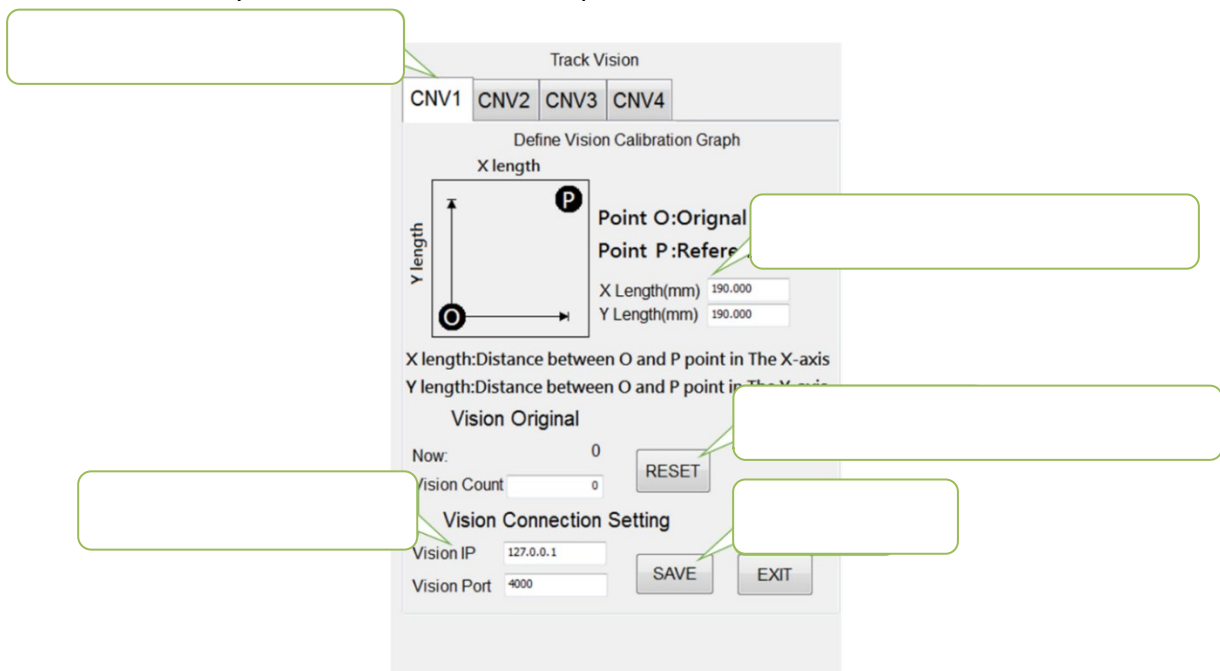
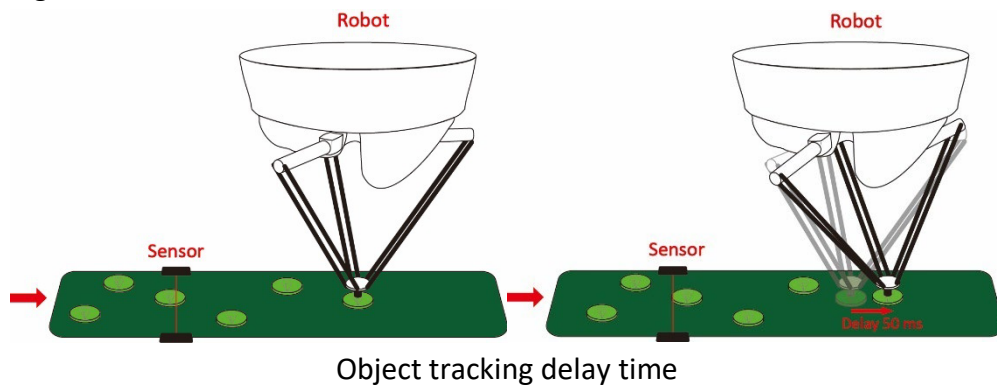


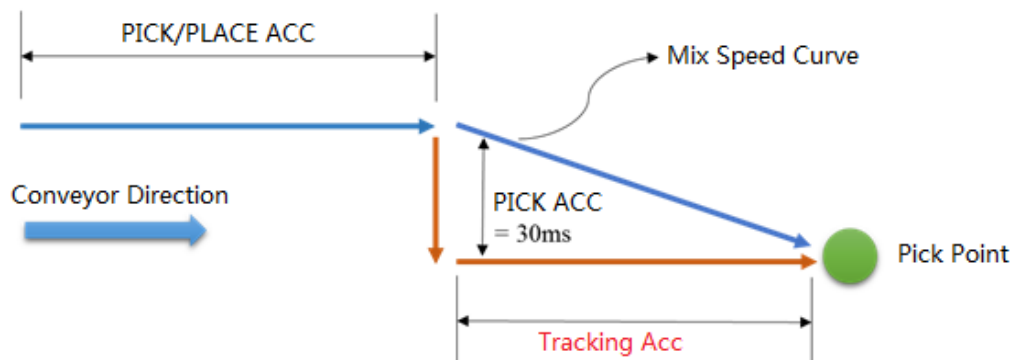
Image parameter configuration interface

3.7.2. Object Parameters

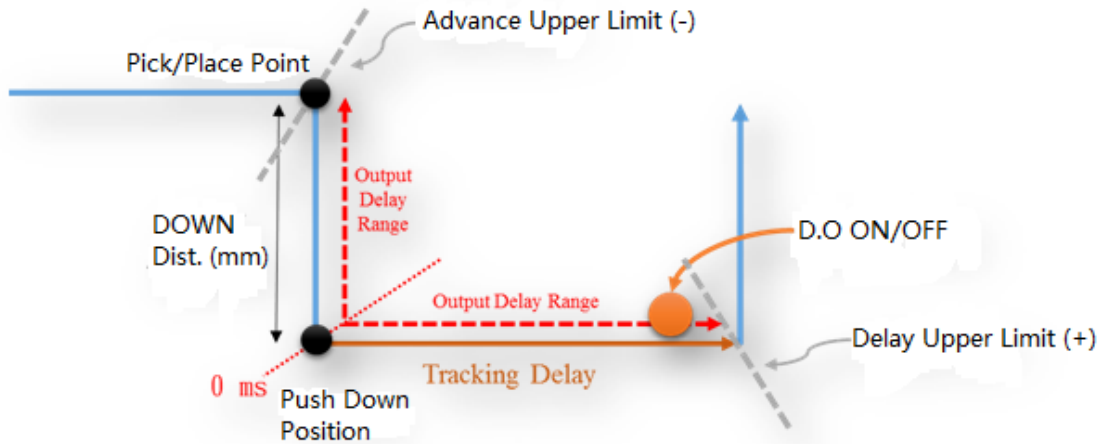
- Main Menu>Track>Setting.
- CNV STATUS : Configure if this conveyor is to be used.
- DIRECTION : Configure the encoder counting direction of this conveyor.
- Trigger Type : Configure the retrieving type of conveyor object, 1 for using image triggering, 2 for using SENSOR triggering.
- Trigger times: Conveyor tracking state variables, conveyor is set to be used when sensor is triggered. When the sensor is triggered, the arm will receive a signal to perform pick or place. The variable can be set to trigger the sensor several times before adding another work task. Setting range is 1~100, default value as 1.
- Place Batch: The place variable is used when multiple objects are placed in the same workspace. When the sensor on conveyor is triggered, the robot will obtain a position where the objects can be placed. The maximum number of place times which the robot will be at that position can be set by this variable. Input range 1~100, default value is 1.
- Tracking Delay : Set how long does take to follow the object and return to action when object is tracked, unit is ms, range 0~1500, default value 0.



- Tracking Acc : Synchronize acceleration/deceleration time in conjunction with conveyor, unit is ms, range 4~1000, default value is 150.

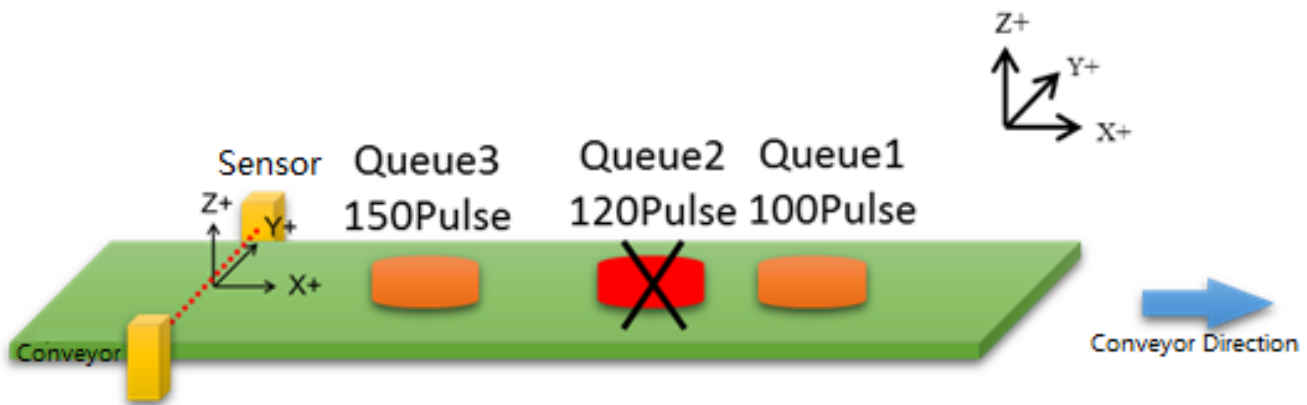


- Output Delay : Configure the D.O. time required to open when the arm is tracked in place, use the positive/negative value to advance or delay. Range -1500~1500, default value is -25.



Output delay time

- Min Latch Cnt : Sensor triggers the filtering. Configure the minimum interval of Latch. For example: if the difference between the previous object of Latch and the current object of Latch is less than the setting of Count, the current object of Latch will be ignored, default value is 0. Figure below shows 3 object on the conveyor, when Min Latch Cnt is set as 25, Queue 2 will be ignored.



Description of Min Latch Cnt

- Compare Nb : Vision triggers the filtering function, it is able to configure the size of filtering Buffer, this size affects when the vision acquires the information of new object, it is required to compare the distance with the n number of determined Latch objects, and then determine if it is the Latch object. Range x0~20, default value is 0.
- Compare Dist : Vision triggers the filtering function, undertake the functional description of above point, this parameter is the length of compared distance, and default value is 0.00.
- Ack Package Setting : Customize image return signal content, Default return "{Conveyor number}".

Track Setting				
ITEM	CNV1	CNV2	CNV3	CNV4
CNV STATUS	Not use	Used	Not use	Used
DIRECTION		Reverse		Forward
TRIGGER TYPE		Sensor		Sensor
TRIGGER TIMES		1		1
PLACE BATCH		1		1

Motion Setting				
ITEM	CNV1	CNV2	CNV3	CNV4
Tracking Delay	0	0	0	0
Tracking Acc	150	150	150	150
Output Delay	-25	-25	-25	-25
Min Latch Cnt	0	0	0	0
Compare Nb	0	0	0	0
Compare Dist	0.00	0.00	0.00	0.00

Ack Package Setting	
<input checked="" type="radio"/> Default	
<input type="radio"/> User Defined	<input type="text"/>

Object parameters setting interface

3.8. Home and Position Check Configuration

3.8.1. Home Configuration

Description

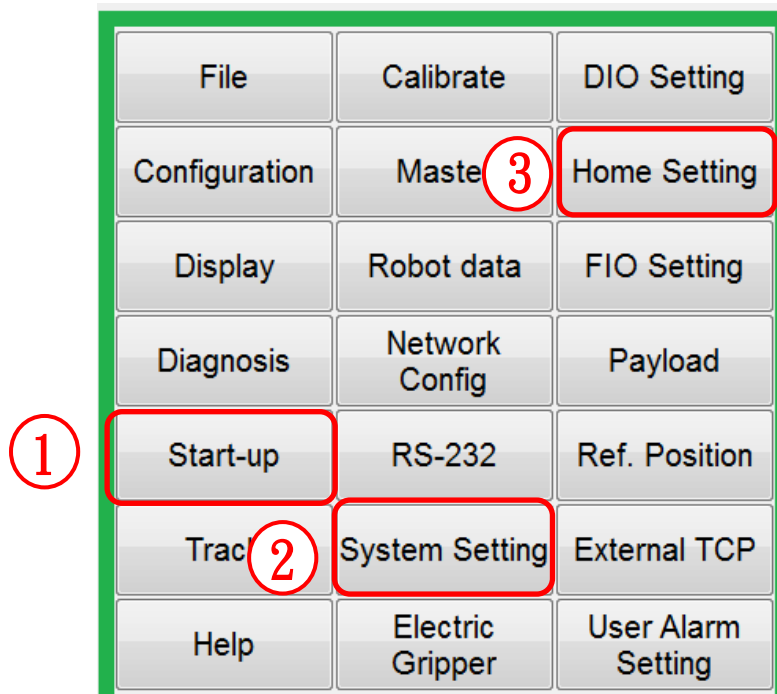
Self-setting or recover the origin (Home) position.

Prerequisite

T1 mode

Operation steps

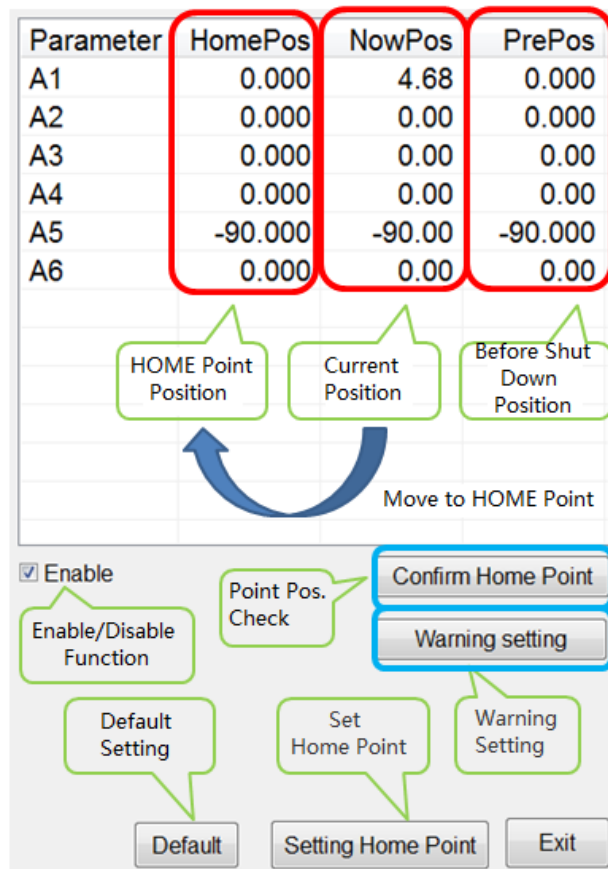
1. Select Start-up>System Setting>Home Setting. (Figure below)
2. Use Jog to move to the user-defined origin position.
3. Press the Setting Home Point button after moving. (Figure below)
4. The "Do you want to set new home point?" window appears.
5. Press the OK button to complete.
6. If you want to recover the initial setting, press the Default button.
7. The "Do you want to recover default value?" window appears.
8. Press the OK button to complete.



Home Setting

3.8.2. Configuration of Position Check Alarm

- Description
The parameters for position check alarm appears.
- Prerequisite
T1 mode
- Operation steps
 1. Select Start-up>System Setting>Home Setting. (Figure below)
 2. Click **Warning Setting** button (Figure below).



Home Setting interface

1. In the **Define allowable error value** field, the tolerant error range of the first axis to the sixth axis can be set. If the angle difference before shutdown and after booting is greater than this setting value, the alarm will appear "Start pos declination error." Code 01-04-30.
2. In the **Define the range of home position** field, you can set the range of angles that can remove the alarm. When the position check alarm appears, you need to manually move back to the origin. If the final position is located in this range, you can use Confirm Home Point to remove the alarm.
3. Press the **Save** button to save the set value.

The image shows a software interface for setting warning parameters. It is divided into two main sections, each with a title in a rounded rectangle:

- Define allowable error value** (title in a red box):
 - Axis 1 angle: ± 1.000 °
 - Axis 2 angle: ± 1.000 °
 - Axis 3 angle: ± 1.000 °
 - Axis 4 angle: ± 1.000 °
 - Axis 5 angle: ± 1.000 °
 - Axis 6 angle: ± 1.000 °
- Define the range of home position** (title in a blue box):
 - Axis 1 range: ± 1.000 °
 - Axis 2 range: ± 1.000 °
 - Axis 3 range: ± 1.000 °
 - Axis 4 range: ± 1.000 °
 - Axis 5 range: ± 1.000 °
 - Axis 6 range: ± 1.000 °

At the bottom right, there are two buttons: "Save" (highlighted with a green border) and "Exit" (highlighted with a blue border).

Warning setting interface

3.8.3. Position Check

- Description

The alarm can be removed because of the difference before and after booting “Start pos declination error”

- Operation steps

1. Select Start-up>System Setting>Home Setting. (Figure below)
2. The NowPos field shows the current axle angle, and the HomePos field shows the setting Home. (Figure below)
3. Using hand to move the robot manually, so that the angle of NowPos moves closer to HomePos.
4. When it is close to Home, even though there is an angle within 1° difference, Press **Confirm Home Point** to finish the position check, and remove the alarm.

3.9. Infinite Rotation Function

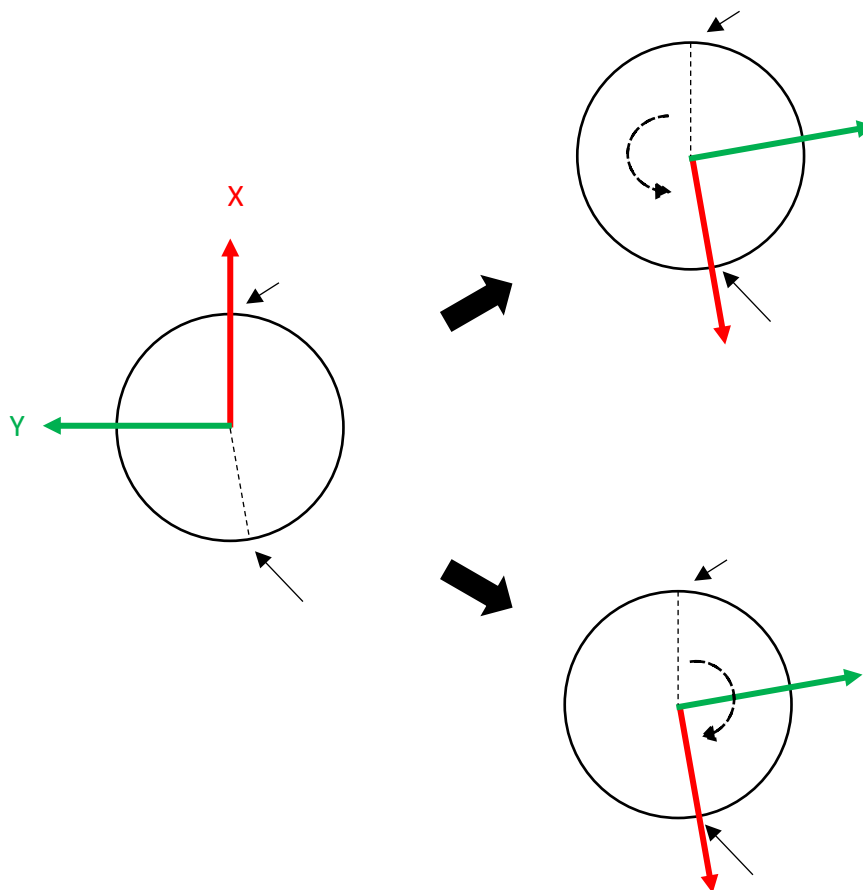
3.9.1. Axis 6 Infinite Rotation Function

Description

This can set the infinite rotation function of the last axis of the robot arm so that the last axis of the robot arm does not have single-rotation limits while operating, allowing it to achieve applications with special purposes under special situations (light grinding, rotating discharge without torque). When enabling infinite rotation functions for SCARA robot arms, the home point check function will also be enabled; refer to the explanations in 3.8.2.

The angle of the end axis will be converted to a relative position representation of ± 180 .

When executing the infinite rotation function, the end axis will move in the direction with fewer rotations from the current position to the target position in order to shorten operation time.



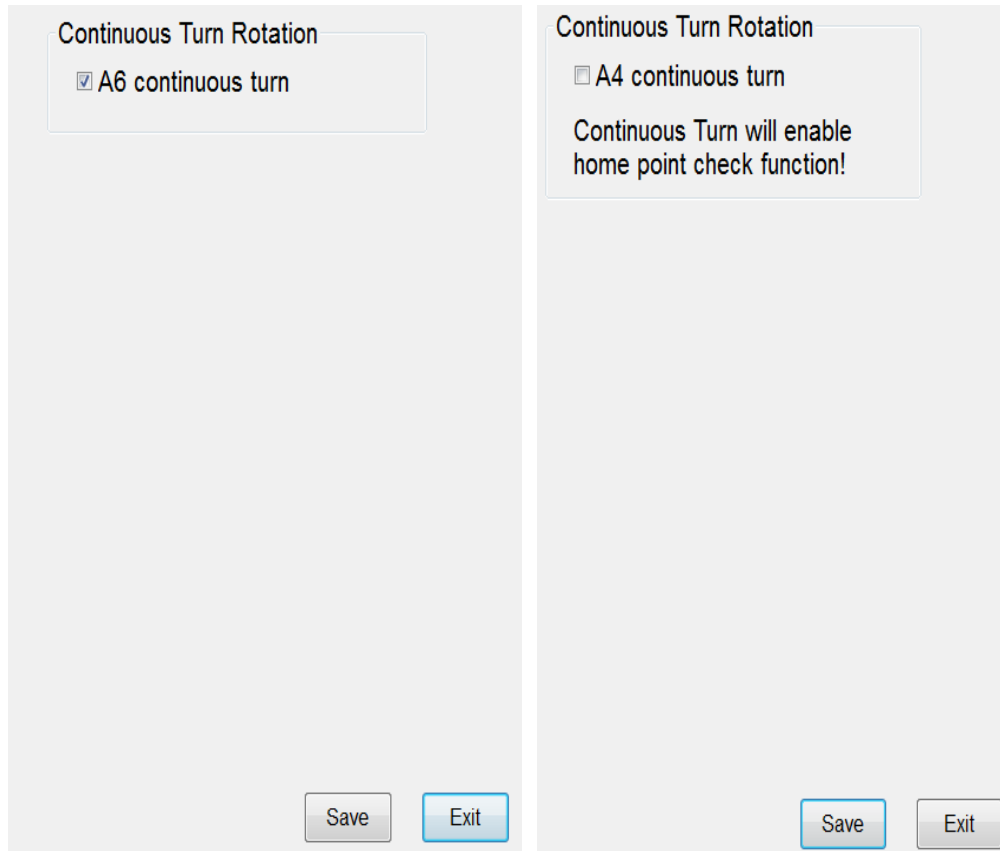
Prerequisite

This function can only be used by joint robot arms and SCARA robot arms for now.

Expert user group

Operation Steps

1. Main Menu> Start-up>System Setting>(Next)>Continuous Turn
2. Check A6 continuous turn for six axes; check A4 continuous turn for SCARA.
3. Click **Save** button, "Saved Successfully!" window will pop up when setting is saved.
4. The user can manually move the sixth axis of the manipulator without limit.
5. The user can use the CT_A6 command to perform the infinite rotation of the sixth axis of the manipulator.

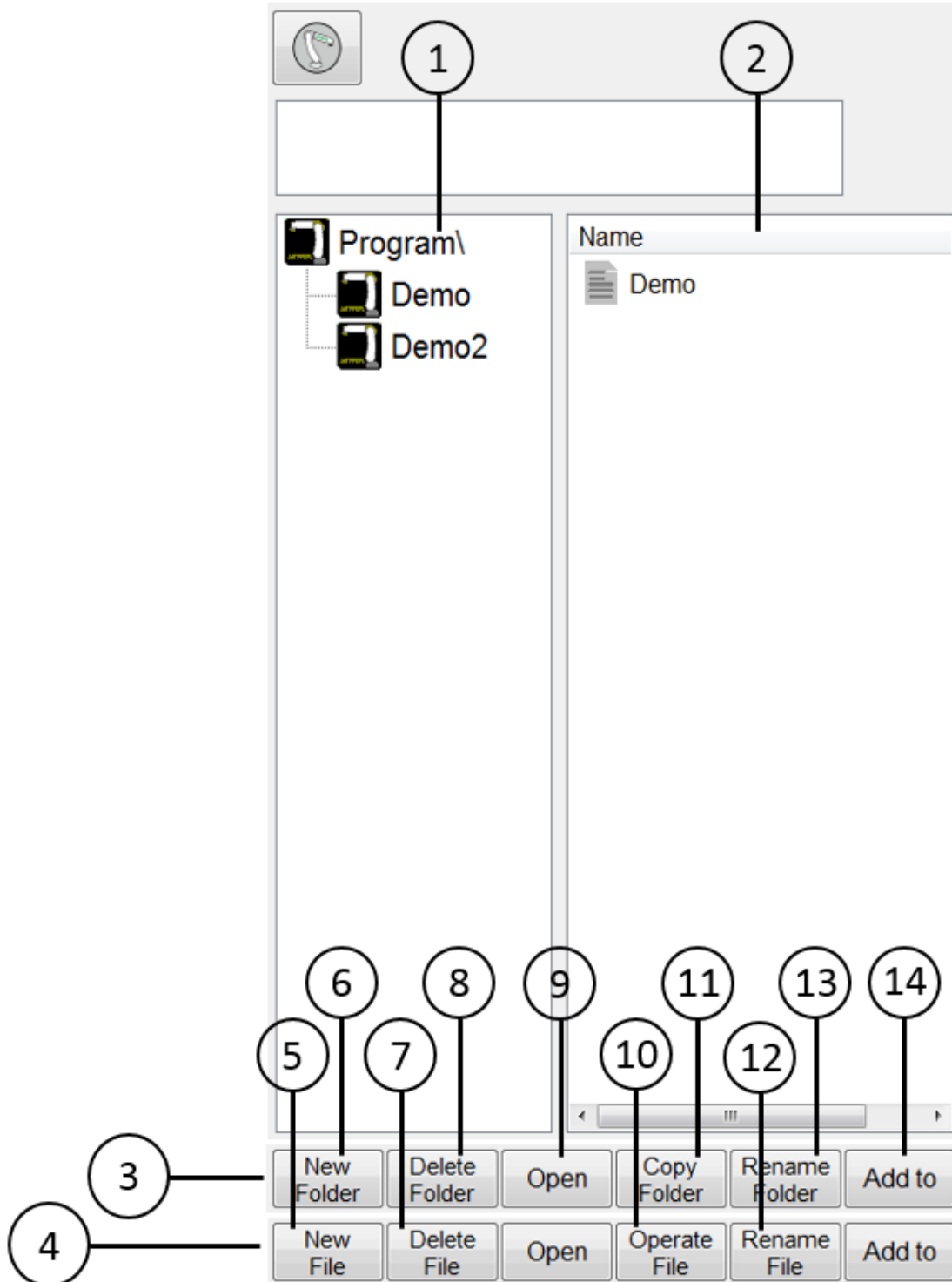


Infinite rotation interface of the last axis (left figure: joint robot arm, right figure: SCARA robot arm)

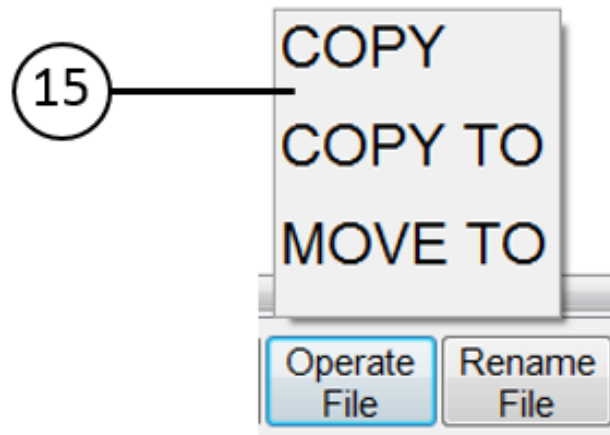
4. Program Management

4.1. Document Navigator

Overview



Navigator



Program file operation interface

Description

A user can manage the program in the navigator.

No.	Description
1	Content structure When double click the folder, refresh the document list.
2	Document list Display the program in the content structure
3	Click on No. 1 to display the folder operation
4	Click on No. 2 to display the program file operation
5	Add new program
6	Add new file
7	Delete program file
8	Delete folder
9	Open program
10	Operate program file
11	Copy folder
12	Rename program
13	Rename folder
14	Add the program to Functional I/O
15	Click on No. 10 to open this operation. Program file can be "COPY", "COPY TO" and "MOVE TO".

4.2. Add Program

Prerequisite

T1 mode, T2, AUT and EXT

T2, AUT and EXT cannot edit the program.

Overview

One program file can be added.

Operation Steps

1. Click on New File
2. Enter the name of the new file, make sure it meets the file name specification.
3. Press Ok to complete adding new program.

4.3. Copy Program

Prerequisite

T1 mode, T2, AUT and EXT

T2, AUT and EXT cannot edit the program.

Overview

One program file can be copied.

Operation Steps

1. Click the program file from the program list.
2. Select the COPY from the operate file or COPY TO folder
3. If selected COPY, the current program in the folder will be copied.
4. If selected COPY TO, user can choose which folder to copy.
5. The name of the copied file is called "xxx_copy.hrb", if the name is existed, the rename window will appear. Please type in the name that has not been used and make sure the file of the name meets the file name specification.

4.4. Open Program

Prerequisite

T1 mode, T2, AUT and EXT

T2, AUT and EXT can not edit the program.

Overview

Select or open a program. An editor and a program will be displayed, but not the navigator.

The program has been opened:

1. The program can start.
2. Only experts can edit the program.

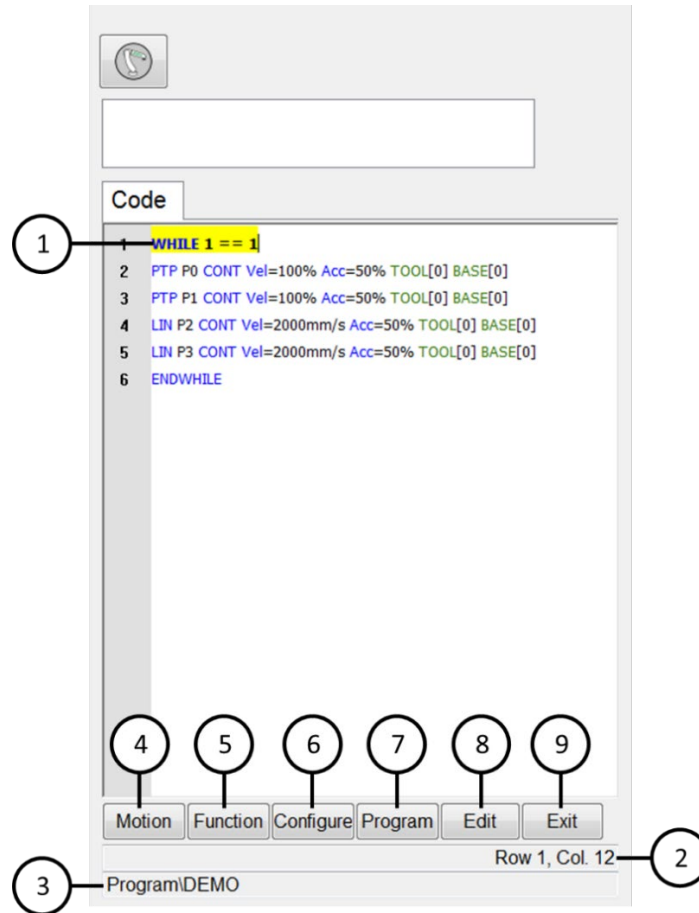
Operation steps

1. Select the program in the navigator and press the Open button. That program will be displayed on the editor.
2. Edit the program.
3. Close the program.

WARNING

When bring external computer-edited program file is loaded into the controller, the file cannot be opened if the file name does not meet the format of the file name.

Description



Program interface

No.	Description
1	Cursor & highlighted line The line is highlighted when a cursor is moved on it.
2	Cursor position
3	Program path and filename
4	Basic motion command
5	Determination and loop command
6	Control command (Timer, Counter and Output)
7	Operation control (copy, cut, paste, delete, comment, cancel annotation, indentation, cancel indentation, recovery, cancel recovery)
8	Edit the program content which has already been established.
9	Exit

4.5. HRSS STRUCTURE

```
...
8 LIN P1 CONT Vel= 200 mm/s Acc=50% Tool[3] Base[4]
...
14 PTP P1 CONT Vel= 100 % Acc=50% Tool[3] Base[4]
...
```

Line	Description
8	LIN motion
14	PTP motion

If the first motion command is not a default home position or that position has changed, one of the following commands must be used:

- Complete PTP command
- Complete LIN command

“Complete” represents all contents which must enter the target points.

WARNING

If you change the home position, all programs will be affected and may cause the injury and property loss.

4.5.1. Home Position

The home position is located in the effective range of the whole program. It is generally used for the start and end. It is clearly defined, but it does not take critical effect.

By default, the home position is located in the control system, and provided with the following values:

Six axis robot:

Axis	A1	A2	A3	A4	A5	A6
Angle	0°	0°	0°	0°	-90°	0°

Delta robot:

Axis	A1	A2	A3	A4
Angle	0°	0°	0°	0°

Other home positions are possible under following conditions:

1. Good starting position for program execution
2. Good standstill position. For example, the stationary robot must not be an-obstacle.

WARNING

If the home position is changed, all programs will be affected which could lead to serious injury, death or equipment damage.



4.6. Start Program

4.6.1. Select Program Running

Operation steps

1. Click "Single Step", and execute the Select button.
2. Select Program Running.

4.6.2. Program Running

Program Running	Description
Continuous 	The program continuously runs to the end.
Single step 	The program will pause after each line. The unseen lines and empty lines are included. The Start button must be pressed again for each line.

4.6.3. Pre-reading

Description

Pre-reading means that the controller will pre-read the program to calculate the motion of the smooth track for example:

```

LIN P1 CONT
LIN P2 CONT
IF $DI[1] == TRUE THEN
LIN P3 CONT
ENDIF

```

During the execution of the program, if IF \$DI[1] == TRUE the condition is true, the controller will pre-read LIN P3 command. If you want to reach P2 before condition is judged, please add WAIT SEC command on the next line of LIN P2. With the command, the program will then judge the condition of \$DI[1] after P2 is reached to decide whether execute LIN P3

4.6.4. Set Program Ratio

Description

The program ratio is used to set the robot velocity. It is represented with a percentage, based on the programmed velocity.

CAUTION

In T1, the maximum velocity is 250mm/s, nothing to do with the set value.

Operation steps



1. Open the velocity window.
2. Set the program ratio. The +/- button or scroll can be used to set.
3. Select the area outside the window. Close the window or apply the ratio.

Another method



The +/- button on the left side of Teach Pendant can be used to set the ratio.

4.6.5. Driver status

The driver status will be displayed in the status bar.

Figure	Color	Description
	Green	Driver ready
	Gray	Driver not ready

4.6.6. Decoder Status Display

Figure	Color	Description
	Orange	Decoder is under running.
	Gray	Decoder is not running.

4.6.7. Start a Program (Manual)

Prerequisite

- Program selected
- T1 mode or T2 mode

Operation steps

1. Select Program Running.
2. Press the Enabling Switch until it displays the status bar “Driver ready”:
3. Press the Start button.
4. The program starts to execute.
5. To stop a program with manual start, press the Stop button or release the Enabling Switch.



4.6.8. Start a Program (Auto)

Prerequisite

Program selected

AUT

Operation steps

1. Select program running.
2. Press the Start button.
3. The program starts to run.
4. To stop a program in automatic mode, press the Stop button.

4.6.9. Start External Auto Run

Prerequisite

Program selected

EXT

Operation steps

1. Select “EXT”.
2. The program starts to activate from the higher-level control system (PLC).
In order to stop the program in EXT Auto Run, press the Stop button.

4.7. Edit Program

Overview

A running program can't be edited.

The program can't be edited in T2, AUT and EXT .

CAUTION

If a selected program is edited in the expert group, a cursor must be moved to another line from the edited line after edited. This will ensure the contents are saved when the program is closed.

4.7.1. Copy Program Bar

Premise

Program is selected or already opened

Expert user group

Operation mode T1

Operating Steps

1. Select the program bar to be copied.
(No need to highlight the program bar, only the cursor is needed.)
2. Select menu Program > Copy

4.7.2. Paste Program Bar

Premise

Program is selected or already opened

Expert user group

Operation mode T1

Operating Steps

1. Select the location where the program bar is to be pasted.
2. Select menu Program > Paste

4.7.3. Cut Program Bar

Premise

Program is selected or already opened

Expert user group

Operation mode T1

Operating Steps

1. Select the program bar to be cut.
(No need to highlight the program bar, only the cursor is needed.)
2. Select menu Program > Cut

4.7.4. Delete Program Line

Prerequisite

Program is selected or already opened
Expert user group
T1 mode

Operation steps

1. Select the program lines to be deleted. It is not necessary to highlight the program line. If a cursor is in the program line, it is ok. (No need to highlight that program bar, only need the cursor in that program bar.)
2. Select menu Program>Delete.

CAUTION

The program lines deleted cannot be recovered again!

4.7.5. Comment Program Bar

Premise

Program is selected or already opened
Expert user group
Operation mode T1

Operating Steps

1. Select the program bar to be commented.
(No need to highlight the program bar, only the cursor is needed.)
2. Select Menu Program > Comment

4.7.6. Cancel Comment Program Bar

Premise

Program is selected or already opened
Expert user group
Operation mode T1

Operating Steps

1. Select the program bar required to cancel comment.
(No need to highlight the program bar, only the cursor is needed.)
2. Select Menu Program > Uncomment

4.7.7. Indent Program Bar

Premise

Program is selected or already opened
Expert user group
Operation mode T1

Operating Steps

Select Menu Program > Indent

4.7.8. Cancel Indent Program Bar

Premise

Program is selected or already opened
Expert user group
Operation mode T1

Operating Steps

1. Select the program bar required to cancel indent.
(No need to highlight the program bar, only the cursor is needed.)
2. Select Menu Program > Unindent

4.7.9. Recovery Program Bar

Premise

Program is selected or already opened
Expert user group
Operation mode T1

Operating Steps

Select Menu Program > Undo

4.7.10. Cancel Recovery Program Bar

Premise

Program is selected or already opened
Expert user group
Operation mode T1

Operating Steps

Select Menu Program > Redo

4.8. Backup and Recovery Data

4.8.1. Backup Data

Description

This function will create a HIWIN/Backup on the USB memory stick with the year and date as the name of the folder for the program.

Prerequisite

Connect a USB memory stick to the control system.

Operation steps

1. Main Menu>File>Save to USB.
2. Wait for the completion window, and close it.
3. Remove the USB memory stick when the LED indicator turns off.

4.8.2. Recovery Data

Description

WARNING

Only load saved file data with the same model and same software version in the robot program. If other files are loaded, the following results could appear:

1. Errored information
2. The robot controller will not run
3. Injury or property damage could occur

Prerequisite

If data should be restored from the flash drive: Connect the flash drive with files saved; if the flash drive is used for the first time, a new folder with the name “HIWIN” must be added. In the HIWIN folder, add another folder named “Backup,” then place the program files (.hrb) to place in HRSS into the “Backup” folder (the arm will only read the files in the path E/HIWIN/Backup).

You can now connect the flash drive onto the control system.

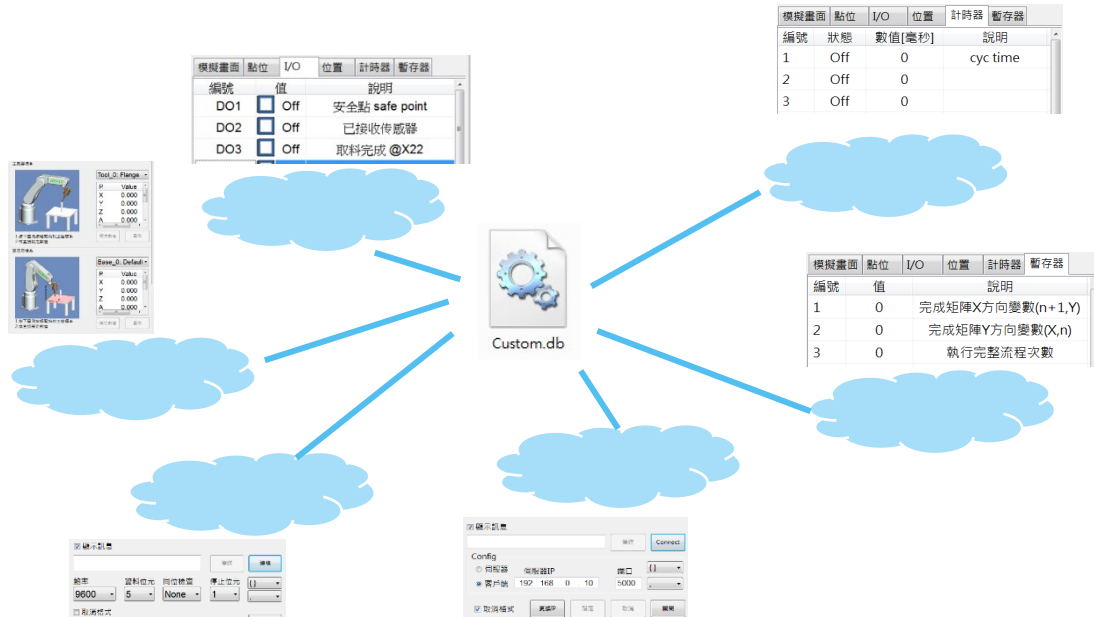
Operation steps

1. Main Menu>File>Load from USB, and then select the program you need.
2. Click Yes to answer the safety inquiry. The saved files will be recovered on the control system.
3. Remove the USB memory stick when the LED indicator turns off.
4. Reboot the control system.

4.8.3. Backing Up Database

Description

This menu item will generate the folder HIWIN/Backup on the flash drive (USB). And will create a folder with the current year and date and backup the contents of the database; the extension of database files is .db files. The contents recorded in the database include the tool/base coordinate systems, comment of the I/O window (DI/O, RI/O, MI/O, SI/O), comment of the counter register, comment of the timer register, comment of the position register, setting information of the network connection interface and the setting information of the RS-232 communication interface.



Database backup content information diagram

Prerequisite

You can connect the flash drive to the control system of the robot arm.

Operation steps

1. Insert the flash drive to the USB port of the controller
2. Main menu>File>Save DataBase.
(A selection window will pop up if on HRSS 3.3.16 offline software or above, allowing users to specify the folder path of the database to backup.)
3. Enter the name of the backup database and click the "OK" button; wait for the save complete window to pop up and click the "OK" button to close the completion window.
4. After the LED indicator on the flash drive goes off, it can be removed.

4.8.4. Restoring Database

Description

Restoring the contents of the database file on the flash drive (USB) onto a new robot arm allows the database to restore the original settings or quickly copy the same parameters and remarks onto multiple robot arms. It is also possible to selectively update the text remarks of the database file through the flash drive (USB). (No updates for the data of the database). It is recommended to back up the database before restoring the database.

WARNING

Only load saved file data with the same model and same software version in the robot program. If other files are loaded, the following results could appear:

1. Errored information
2. The robot controller will not run
3. Injury or property damage could occur

Prerequisite

Connect USB memory stick with files to the controller.

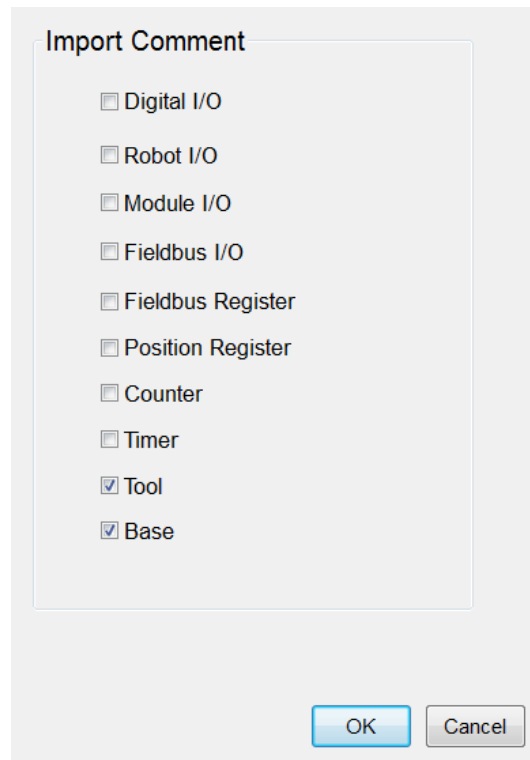
Operation steps

■ **Restoring the database:**

1. Place the files on the flash drive; the flash drive is used for the first time, a specific path (HIWIN\Backup) folder needs to be created and insert the flash drive into the USB port of the controller.
2. Main menu>File>Load DataBase, then select the database name needed and press the “Import” button.
3. Click on the “Yes” button to confirm the security query. The saved database will be restored onto the control system of the robot arm.
4. The control system of the robot arm needs to be restarted for the contents of the new database to be imported.

■ **Importing the comment function of the database:** (This method is only applicable to HRSS version 3.3.16 and above)

1. Place the files on the flash drive; the flash drive is used for the first time, a specific path (HIWIN\Backup) folder needs to be created and insert the flash drive into the USB port of the controller.
2. Main menu>File>Import Comment.
Select the name of the database (.db) to import in the window.
(A selection window will pop up on HRSS 3.3.16 offline software or above; select the name of the database (.db) to import in the file window and click the “Open” button.)
3. Check options will pop up for the selected database to import; check the frames for the database contents to import, multiple choices can be made. Take the figure below for example. The Tool and Base databases are checked simultaneously for import operation.



Import Comment selection window diagram

4. Click on the “OK” button to confirm the safety inquiry. The saved database will be restored into the control system of the robot arm; click on the “Clear” button to return to the Load Database screen. The confirm database text comment is displayed on the window (no need to restart); please note that the existing text comment will be overwritten.

4.9. Rules for naming files

Overview

Rules for naming files, the program file has to satisfy this rule to ensure adding or copying of the file successfully.

Content

1. Only Arabic numerals (0-9), English letters (a-z, A-Z) and underscore (_) can be used for naming.
2. Special symbols 「 ~!@#\$\$%^&*()-+={}[]<>.,?/\| 」 are not allowed.
3. The first character cannot be a number.
4. Not more than 100 words.

WARNING

If the name of the program file does not correspond to the rules, follow situation may occur when operating other function :

1. Unable to open the file
2. Unable to copy the file
3. Unable to be added to external startup functions list
4. Unable to use external subroutine functions

5. Motion Program Design

5.1. Motion Overview

Program designed by the following motion:

- Point-to-point motion (PTP)
- Linear motion (LIN)
- Circular motion (CIRC)

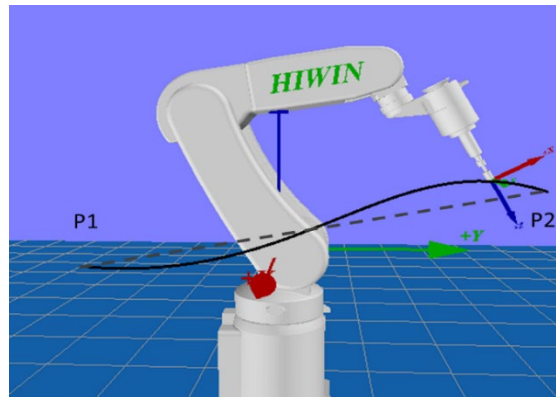
LIN and CIRC Motion is also called as “CP motion” (CP = Continuous Path).

A start point must begin at the end point of the previous motion.

5.2. Point-to-point (PTP) Motion

The robot guides TCP to the target point along the fastest path. Generally the fastest path is not the shortest one. This means that it is not a straight line. Because the axis performs rotational motion, the curved path is faster than the straight one.

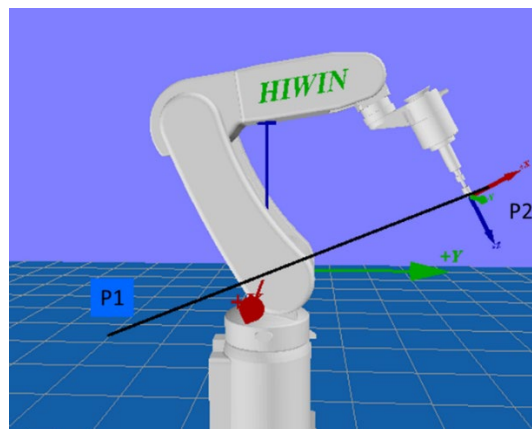
The motion cannot be accurately known in advance.



PTP motion

5.3. LIN Motion

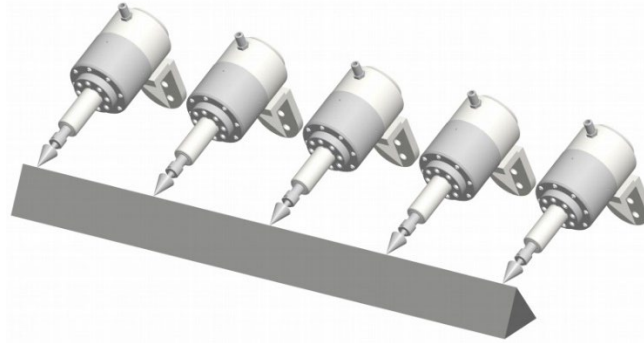
The velocity defined by the robot along a straight line moves TCP to the target point.



LIN motion

TCP at the start point of motion could be in an orientation different from the target point. The orientation of TCP

will gradually change during the motion. When the TCP is at the start point of motion and in the same orientation as the target point, the orientation of TCP will remain the same in the motion.



Start point in same orientation of target point



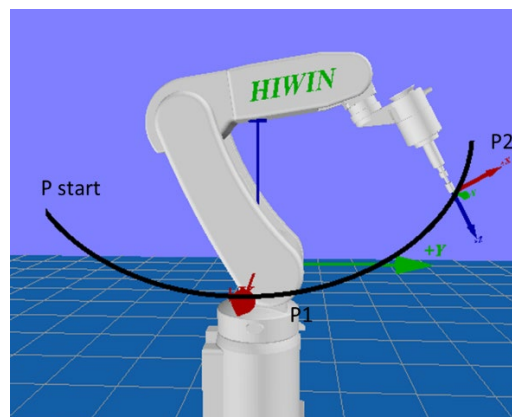
Start point in different orientation of target point

5.4. CIRC Motion

The velocity defined by the robot along the circular path moves the TCP to the target point. A circular track is defined by the start point, auxiliary point and target point.

For the CIRC motion, the orientation guide is the same orientation as with LIN motion.

In the CIRC motion, the control system only considers the orientation of the target point. The orientation of auxiliary point is usually ignored.



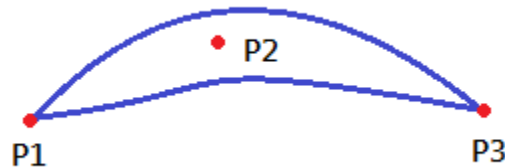
CIRC motion

5.5. Blend

Blend: Not accurately moved to the point programmed. The over blending is another option that can be selected during the motion program.

- PTP motion

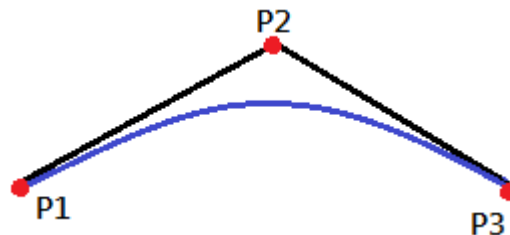
The TCP will leave a track where it can accurately reach the target point, and adopt the faster one. When over blending takes place in a PTP motion, the track change cannot be foreseen. The point through which side on the track cannot be forecasted.



PTP motion and P2 blended

- LIN motion

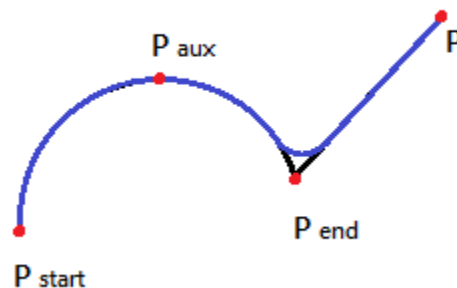
TCP will leave a track where it should accurately move to the target point, and run on a shorter track. The region where the track path runs is not an arc.



LIN motion and P2 blended

- CIRC motion

TCP will leave a track where it should accurately move to the target point, and run on a shorter track. The auxiliary point can reach accurately. The region where the track path runs is not an arc.



CIRC motion and Pend blended

5.6. Singular Point

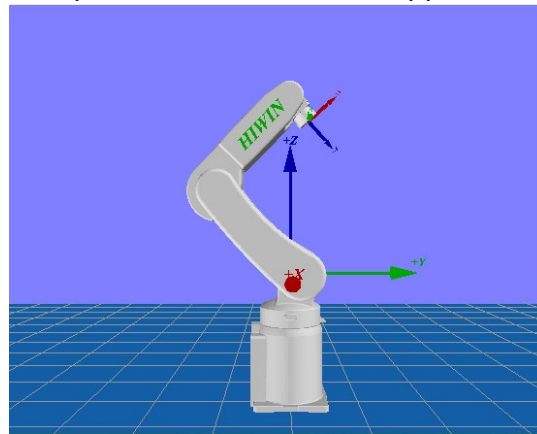
The HIWIN's robot with six degrees of freedom has three kinds of singular point.

1. Overhead singular point
2. Singular point at extended position
3. Singular point at wrist axis

It is considered as a singular point position only when one value can't be obtained by the inverse conversion (converted from Cartesian coordinate to articulated coordinate). In this situation, it is a position of singular point when the minimum Cartesian variation could cause a large change of axis angle.

- Overhead

For the overhead singular point, the wrist point (the middle point of axis A5) is vertical to the axis A1. The position of the axis A1 cannot be confirmed by the inverse conversion, and it can be any value. At this point, if the inverse motion is performed, an error will appear.



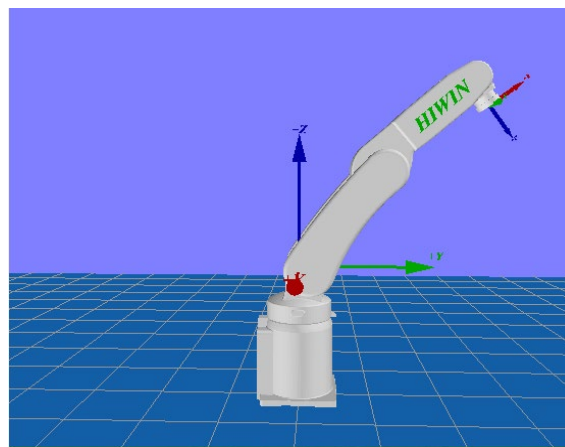
- Extended position

For the singular point at the extended position, the wrist point (the middle point of axis A5) is located in the extension of axis A2 and A3.

The robot is located at the edge of the workspace.

Although only one axis angle can be obtained by the inverse conversion, the small Cartesian variation will cause the large velocity of the axis A2 and A3.

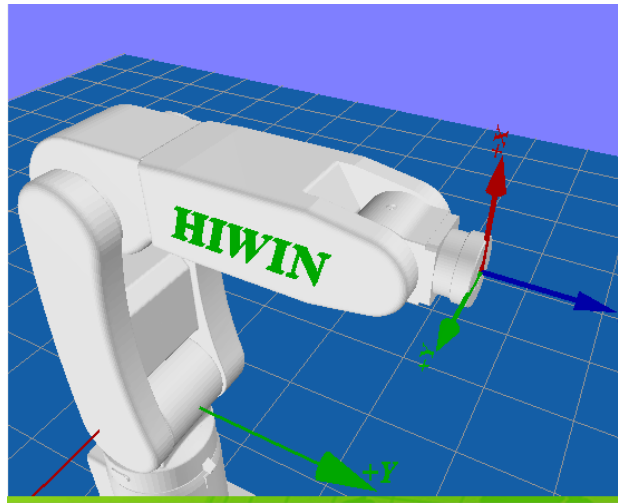
At this point, if the inverse motion is performed, an error will appear.



- Wrist axis

For the singular point of the wrist axis, the axis A4 is parallel with A6, and the axis A5 is within the range $\pm 0.1^\circ$. The positions of two axes can't be confirmed by the inverse conversion. Although the axis A4 and A6 can have

many positions but the sum of the axis angle is the same.
At this point, if the inverse motion is performed, an error will appear.



6. Programming with inline forms

Frequently used commands are provided in the HRSS inline forms. These commands can simplify program design.

CAUTION

Commands can be program without inline forms. The HRL (HIWIN robot language) is used to program.

6.1. Name in Inline Form

The program can enter the data name. For example, the name of the motion data set.

The name must satisfy the following limits:

1. No more than 30 characters.
2. Special characters other than \$ are not allowed.
3. The first character can not be a number.

This limit does not exist for the output name.

6.2. Programming for PTP, LIN, CIRC, SPINE Motion

6.2.1. Programming for PTP Motion

CAUTION

When motion is programmed, ensure the power supply will not wind or be damaged when the program is run.

6.2.1.1. PTP

Prerequisite

Program selected

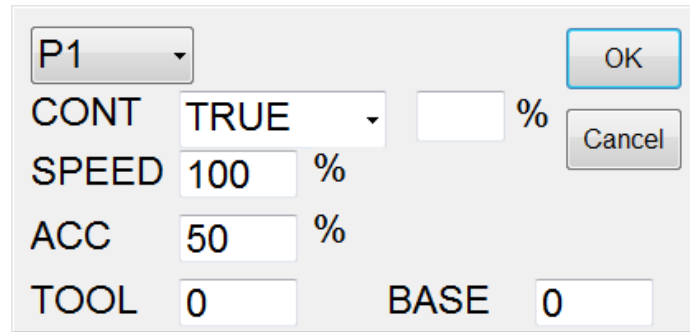
T1 mode

Operation steps

1. The TCP move should be configured as the target position.
2. Put the cursor behind, and insert on the line of the motion command.
3. Select Motion>PTP ◦

4. Change the relevant parameters.
5. Press the OK button.

Overview



PTP interface

6.2.1.2. PTP (programmed by keyboard)

Prerequisite

- Program selected
- T1 mode
- Keyboard connected

Operation steps

1. Put the cursor behind, and insert on the line of the motion command.
2. Input the command by keyboard.

Example:

E6POS POINT = {X 0,Y 300,Z 200}

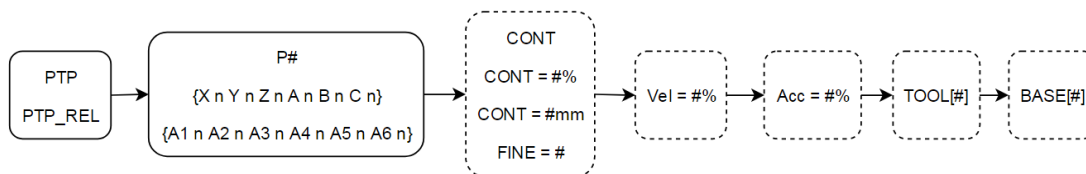
PTP POINT CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

Example:

PTP {X 100} CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

Example:

PTP {A1 45} CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]



PTP&PTP_REL flowchart

6.2.1.3. PTP_REL (programmed by keyboard)

Prerequisite

- Program selected

T1 mode

Keyboard connected

Operation steps

1. Put the cursor behind, and insert on the line of the motion command.
2. Input the command by keyboard.

Example:

PTP_REL {X 100} CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

Example:

PTP_REL {A1 45} CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

Command flowchart

Refer to PTP

6.2.2. Programming for LIN Motion

⚠ CAUTION

When designing the movement program, you should make sure that the cable of the power supply system is not wound or damaged while the written program is running.

6.2.2.1. LIN

Prerequisite

Program selected

T1 mode

Operation steps

1. The TCP move should be configured as the target position.
2. Put the cursor behind, and insert on the line of the motion command.
3. Select Motion>LIN.
4. Change the relevant parameters.
5. Press the OK button.

Overview

The screenshot shows a configuration dialog box with the following fields and values:

- Program: P1
- CONT: TRUE
- SPEED: 2000 mm/s
- ACC: 50 %
- TOOL: 0
- BASE: 0

LIN interface

6.2.2.2. LIN (programmed by keyboard)

Prerequisite

- Program selected
- T1 mode
- Keyboard connected

Operation steps

1. Put the cursor behind, and insert on the line of the motion command.
2. Input the command by keyboard.

Example:

E6POS POINT = {X 0,Y 300,Z 200}

LIN POINT CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

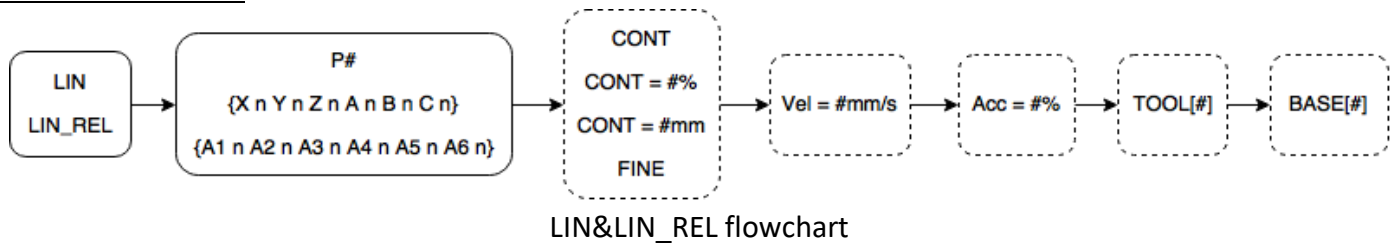
Example:

LIN {X 100} CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

Example:

LIN {A1 45} CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

Command flowchart



6.2.2.3. LIN_REL (programmed by keyboard)

Prerequisite

Program selected
T1 mode
Keyboard connected

Operation steps

1. Put the cursor behind, and insert on the line of the motion command.
2. Input the command by keyboard.

Example:

```
LIN_REL {X 100} CONT=100% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]
```

Example:

```
LIN_REL {A1 45} CONT=100% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]
```

Command flowchart

Refer to LIN

6.2.2.4. LIN_REL_TOOL (programmed by keyboard)

Prerequisite

Program selected
T1 mode
Keyboard connected

Operation steps

1. Put the cursor behind, and insert on the line of the motion command.
2. Input the command by keyboard.
3. One axis can be rotated at a time

Example:

```
LIN_REL_TOOL {X 100} CONT=100% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]
```

Example:

```
LIN_REL_TOOL {A 45} CONT=100% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]
```

Command flowchart

Refer to LIN

6.2.3. Programming for CIRC Motion

CAUTION

When the motion is programming, it is sure the power supply will not wind or damage when programming to run.

6.2.3.1. CIRC

Prerequisite

Program selected

T1 mode

Operation steps

1. The TCP move should be configured as the arc position.
2. Put the cursor behind, and insert on the line of the motion command.
3. Select Motion>CIRC.
4. Click [SET].
5. The TCP move should be configured as the target position.
6. Click [SET].
7. Click [FINSH].
8. Enter the Paramer interface to change the relevant parameters.
9. Press the OK button.

Overview

Auxiliary	Actual end	SET
P1	P2	Cancel
P1 P2		OK
CONT	TRUE	%
SPEED	2000	mm/s
ACC	50	%
TOOL	0	BASE 0
		Cancel

CIRC interface

6.2.3.2. CIRC (programmed by keyboard)

Prerequisite

Program selected

T1 mode

Keyboard connected

Operation steps

1. Put the cursor behind, and insert on the line of the motion command.
2. Input the command by keyboard.

Example:

E6POS POINT1 = {X 0,Y 300,Z 200}

E6POS POINT2= {X 20,Y 320,Z 220}

CIRC POINT1 POINT2 CONT=100% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]

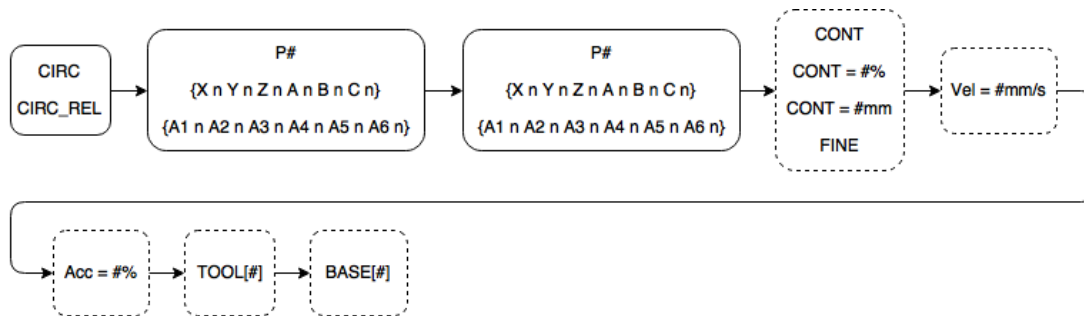
Example:

CIRC {X 0, Y 450} {X -150, Y 300} CONT=100% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]

Example:

CIRC {A1 5.0, A2 5.0, A3 5.0, A4 5.0} {A1 10.0, A2 10.0, A3 10.0, A4 10.0,} CONT=100% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]

Command flowchart



CIRC&CIRC_REL flowchart

6.2.3.3. CIRC_REL (programmed by keyboard)

Prerequisite

- Program selected
- T1 mode
- Keyboard connected

Operation steps

1. Put the cursor behind, and insert on the line of the motion command.
2. Input the command by keyboard.

Example:

CIRC_REL {X -150, Y 150} {X -150, Y -150} CONT=100% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]

Example:

CIRC_REL {A1 5.0, A2 5.0, A3 5.0, A4 5.0} {A1 10.0, A2 10.0, A3 10.0, A4 10.0,} CONT=100% Vel=2000mm/s
Acc=50% TOOL[0] BASE[0]

Command flowchart

Refer to CIRC

6.2.4. Perform Programming for SPLINE Motion

CAUTION

When perform the motion programming, please assure that the power supply system will not be wound or damaged while running the written program.

6.2.4.1. SPLINE (Keyboard Writing)

Premise

Program selected

T1 mode

Keyboard connected

Operating Steps

1. Put the cursor behind, and insert on the line of the motion command.
2. Input the command by keyboard.

Example:

E6POINT P1 = { X 95 , Y 0 , Z -500 }

E6POINT P2 = { X 94.63849632 , Y 3.922008424 , Z -500 }

E6POINT P3 = { X 93.55673654 , Y 7.814167995 , Z -500 }

.....

SPLINE

SPL P1

SPL P2

SPL P3

.....

ENDSPLINE

6.3. Variable Configuration

6.3.1. REAL

Prerequisite

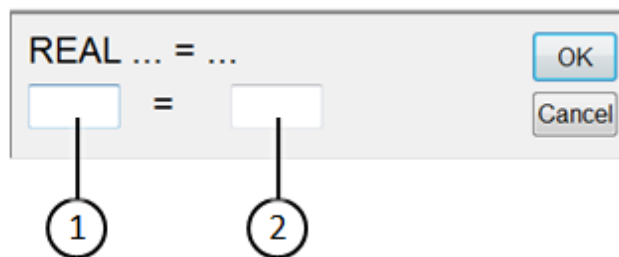
Program selected

T1 mode

Operation steps

1. Put a cursor behind, and insert on the line of the logic command.
2. Select Configure>Variable>REAL.
3. Set the parameters in the interface.
4. Save the command by pressing OK.

Overview



REAL interface

No.	Description
1	Variable name
2	Initial value

6.3.2. INT (programmed by keyboard)

Prerequisite

Program selected

T1 mode

Keyboard connected

Operation steps

1. Put the cursor behind, and insert on the line of the motion command.
2. Input the command by keyboard.

Example:

INT Two = 2

6.3.3. UINT (programmed by keyboard)

Prerequisite

Program selected
T1 mode
Keyboard connected

Operation steps

1. Put the cursor behind, and insert on the line of the motion command.
2. Input the command by keyboard.

Example:

UINT Two = 2

6.3.4. BOOL (programmed by keyboard)

Prerequisite

Program selected
T1 mode
Keyboard connected

Operation steps

1. Put the cursor behind, and insert on the line of the motion command.
2. Input the command by keyboard

Example:

BOOL K = TRUE

6.3.5. CHAR (programmed by keyboard)

Prerequisite

Program selected
T1 mode
Keyboard connected

Operation steps

1. Put the cursor behind, and insert on the line of the motion command.
2. Input the command by keyboard.

Example:

CHAR COLOR = 'R'

6.3.6. STRING(programmed by keyboard)

Prerequisite

Program selected

T1 mode

Keyboard connected

Operation steps

1. Put the cursor behind, and insert on the line of the motion command.
2. Input the command by keyboard.

Example:

STRING str = "HIWIN"

6.3.7. E6POS Coordinate Points (programmed by keyboard)

Prerequisite

Program selected

T1 mode

Keyboard connected

Operation steps

1. Put the cursor behind, and insert on the line of the motion command.
2. Input the command by keyboard.

Example:

E6POS POINT={X 0,Y 300,Z 200}

PTP POINT CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

6.3.8. E6AXIS Axis Points (programmed by keyboard)

Prerequisite

Program selected

T1 mode

Keyboard connected

Operation steps

1. Put the cursor behind, and insert on the line of the motion command.
2. Input the command by keyboard.

Example:

E6AXIS POINT = {A1 90}

PTP POINT CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

6.3.9. E6POINT Coordinate Points (programmed by keyboard)

Prerequisite

Program selected

T1 mode

Keyboard connected

Operation steps

1. Put the cursor behind, and insert on the line of the motion command.
2. Input the command by keyboard.

Example:

E6POINT HOME = {X 0,Y 200,Z -1000,A 90}

PTP HOME CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

Tip: E6POINT does not have the definition of axis point, hence not able to directly set the value of A1~A6.

6.3.10. FRAME Coordinate Points (programmed by keyboard)

Description

Write this instruction in the program, and it will declare a variable including the information of X, Y, Z, A, B, C.

Format

FRAME POINT_A

POINT_A.X = 0

POINT_A.Y = 1

POINT_A.A = 2

POINT_A.C = 3

Format Description

The variable type is the floating-point number, if there is no input, it will set to the default 0.

6.4. Register Configuration

6.4.1. Using COUNTER Register

Prerequisite

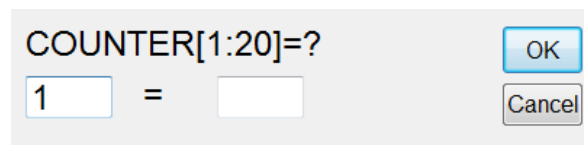
Program selected

T1 mode

Operation steps

1. Put the cursor behind, and insert on the line of the variable command.
2. Select Configure>Variable>Counter.
3. Set the parameters in the interface.
4. Save the command by pressing OK.

Overview



COUNTER interface

6.4.2. Enable TIMER Register

Premise

Program selected

T1 mode

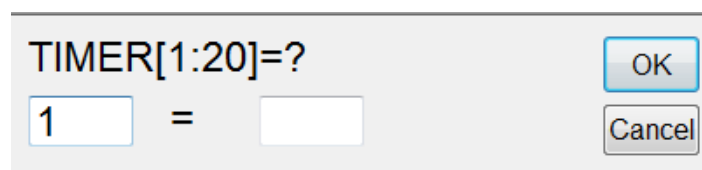
Operating Steps

1. Put the cursor behind, and insert on the line of the variable command.
2. Select Configure > Variable > Stop Timer
3. Set the parameters in the interface.
4. Add “\$T_STOP[n]=FALSE” to the previous line to start timing.
5. Add “\$ T_STOP [n]=TRUE” to the next line to end timing.
6. Save the command by pressing OK.

Overview

⚠ CAUTION

TIMER is based on 1 ms as an unit, where the accuracy is 55 ms.



TIMER Setting Interface

6.5. Programming for Logic Command

6.5.1. Input/Output

- Digital Input/Output

The control system can manage up to 24 digital inputs and 24 digital outputs. The configuration can set depending on the user requirements.

Input/output can be managed by the following variables:

	Input	Output
Digital	\$DI[1]...\$DI[24]	\$DO[1]...\$DO[24]
Valve	--	\$VO[1]...\$VO[3]
Robot	\$RI[1]...\$RI[8]	\$RO[1]...\$RO[8]
Module	\$MI[1]...\$MI[32]	\$MO[1]...\$MO[32]
Fieldbus	\$SI[1]...\$SI[128]	\$SO[1]...\$SO[128]

Remark: The signal above uses RA605-710-GB as the example; different models have different numbers of I/O; please refer to the corresponding robot controller manual, and the fieldbus signal is an optional function.

6.5.2. Digital OUTPUT

Prerequisite

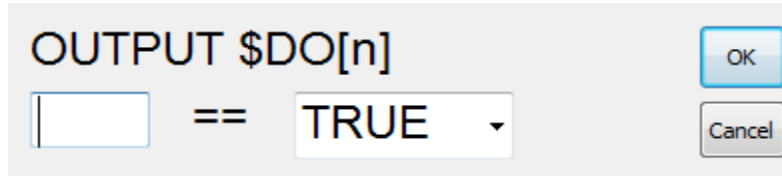
Program selected

T1 mode

Operation steps

1. Put the cursor behind, and insert on the line of the logic command.
2. Select Configure> Output>Digital, or other IO output interface.
3. Set the parameters in the selection menu sequence; enter the signal value in n, select ON for TRUE, select OFF for FALSE.
4. Save the command by pressing OK.

Overview



OUTPUT interface

6.5.3. WAIT FOR (SEC)

Prerequisite

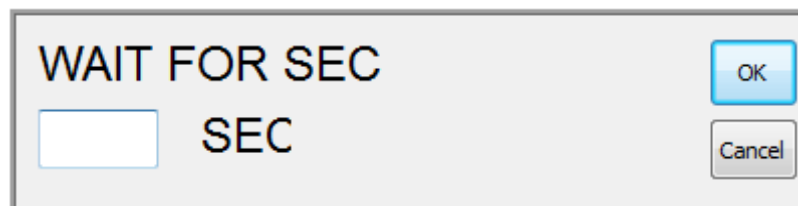
Program selected

T1 mode

Operation steps

1. Put the cursor behind, and insert on the line of the logic command.
2. Select Function> WAIT FOR> SEC
3. Set the parameters in the interface, the unit is second; the smallest unit is 0.01 second.
4. Save the command by pressing OK.

Overview



WAIT FOR SEC interface

6.5.4. WAIT FOR (INPUT)

Prerequisite

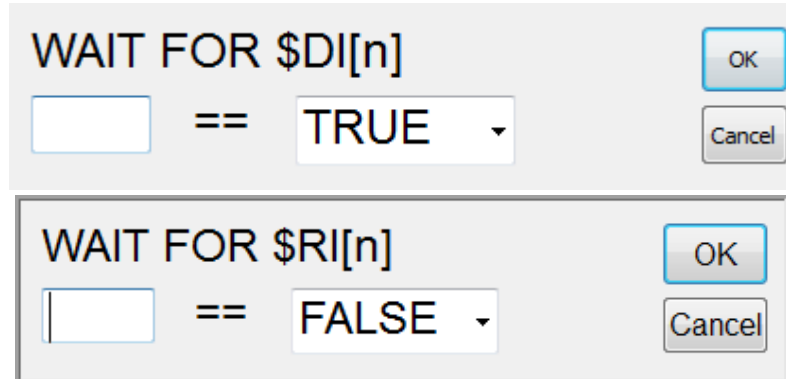
Program selected

T1 mode

Operation steps

1. Put the cursor behind, and insert on the line of the logic command.
2. Select Function> WAIT FOR > INPUT.
3. Set the parameters in the interface; enter the signal value in n, select ON for TRUE, select OFF for FALSE.
4. Save the command by pressing OK.

Overview



WAIT FOR interface

6.5.5. QUIT (programmed by keyboard)

Prerequisite

- Program selected
- T1 mode
- Keyboard connected

Operation steps

1. Put the cursor behind, and insert on the line of the motion command.
2. Input the command by keyboard.

Example:

QUIT

6.6. Perform Program Design for the Judgment Loop Instruction

6.6.1. IF

Prerequisite

- Program selected
- T1 mode

Operation steps

1. Put the cursor behind, and insert on the line of the logic command.
2. Select Function> IF... ENDIF> Determination.
3. Set the parameters in the interface.
4. Save the command by pressing OK.

Overview



IF interface

6.6.2. FOR (programmed by keyboard)

Prerequisite

- Program selected
- T1 mode
- Keyboard connected

Operation steps

1. Put the cursor behind, and insert on the line of the motion command.
2. Input the command by keyboard.

Example:

```
FOR start TO last STEP increment  
.....  
ENDFOR
```

6.6.3. LOOP (programmed by keyboard)

Prerequisite

- Program selected
- T1 mode
- Keyboard connected

Operation steps

1. Put a cursor behind, and insert on a line of the motion command.
2. Input the command by keyboard.

Example:

```
LOOP  
.....  
ENDLOOP
```

6.6.4. WHILE

Prerequisite

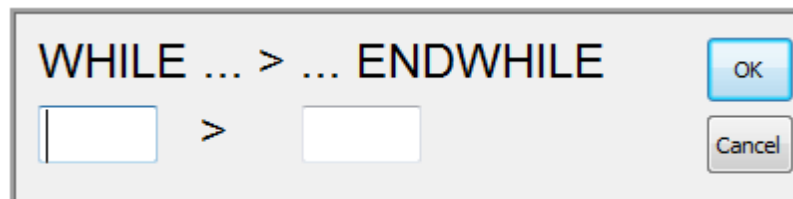
Program selected

T1 mode

Operation steps

1. Put a cursor behind, and insert on a line of the logic command.
2. Select Function> WHILE ENDWHILE> Determination.
3. Set the parameters in the interface.
4. Save the command by pressing OK.

Overview



WHILE interface

6.6.5. REPEAT (programmed by keyboard)

Prerequisite

Program selected

T1 mode

Keyboard connected

Operation steps

1. Put a cursor behind, and insert on a line of the motion command.
2. Input the command by keyboard.

Example:

REPEAT

.....

UNTIL *condition*

6.6.6. GOTO (programmed by keyboard)

Prerequisite

Program selected
T1 mode
Keyboard connected

Operation steps

1. Put a cursor behind, and insert on a line of the motion command.
2. Input the command by keyboard.

Example:

```
IF condition THEN  
GOTO LABEL1  
ENDIF  
IF condition THEN  
GOTO LABEL2  
ENDIF  
IF condition THEN  
GOTO LABEL3  
ENDIF  
LABEL1:  
.....  
LABEL2:  
.....  
LABEL3:  
.....
```



CAUTION

The label specified by the GOTO grammar must be in the current function, cross functions cannot be applied.

6.6.7. SWITCH (programmed by keyboard)

Prerequisite

Program selected
T1 mode
Keyboard connected

Operation steps

1. Put a cursor behind, and insert on a line of the motion command.
2. Input the command by keyboard

Example:

SWITCH *number*

.....

CASE *number1*

.....

CASE *number2*

.....

ENDSWITCH

6.7. Simulate Environment Instruction

6.7.1. ADDTOOL Newly Add Tool (programmed by keyboard)

Premise

Program selected

T1 mode

Stl file putted in the stl folder

Operating Steps

1. Put a cursor behind, and insert on a line of the instruction command
2. Input the command by keyboard.

Example:

ADDTOOL File Name (no need to enter the extension)

Take ee.stl as an example

ADDTOOL ee

Attention: stl file will be in binary format, and the coordinate system of file will be consistent with the end coordinate system of the flange surface. Currently the file named with underscore (_) or started with Upper/Lower case are all acceptable.

6.7.2. SHOW_TOOL Show Tool (programmed by keyboard)

Premise

Program selected

T1 mode

Stl file putted in the stl folder

Operating Steps

1. Put a cursor behind, and insert on a line of the instruction command
2. Input the command by keyboard.

Example:

SHOW_TOOL File Name (no need to enter the extension) TRUE/FALSE

Take ee.stl as an example

Show ee.stl: SHOW_TOOL ee TRUE

Hide ee.stl: SHOW_TOOL ee FALSE

Attention: stl file will be in binary format, and the coordinate system of file will be consistent with the end coordinate system of the flange surface. Currently the file named with underscore (_) or started with Upper/Lower case are all acceptable.

6.7.3. ADDOBJ Newly Add Workpiece (programmed by keyboard)

Premise

Program selected

T1 mode

Stl file putted in the stl folder

Operating Steps

1. Put a cursor behind, and insert on a line of the instruction command
2. Input the command by keyboard.

Exampe:

ADDOBJ File Nmae (no need to enter the extension) P: X, Y, Z, A, B, C C: R, G, B

P: Displacement (mm) & rotation (degree) as relative to the robot origin

C: Color, RGB Value

Take table.stl as an example

ADDTOOL table P: 500 C:200

ADDTOOL table P: 500, 200 C: 200,50

Attention: stl file must be in binary format, and the coordinate system of file must be consistent with the end coordinate system of flange surface. Currently the file named with underscore (_) or started with Upper/Lower case are all acceptable.

6.7.4. SHOW_OBJ Show Workpiece (programmed by keyboard)

Premise

Program selected

T1 mode

Stl file putted in the stl folder

Operating Steps

1. Put a cursor behind, and insert on a line of the instruction command
2. Input the command by keyboard.

Example:

SHOW_OBJ File Name (no need to enter the extension) TRUE/FALSE

Take table.stl as an example

Show table.stl: SHOW_OBJ table TRUE

Hide table.stl: SHOW_OBJ table FALSE

Attention: stl file will be in binary format, and the coordinate system of file will be consistent with the end coordinate system of the flange surface. Currently the file named with underscore (_) or started with Upper/Lower case are all acceptable.

6.7.5. MOVEFLOOR Position of Moving Floor (programmed by keyboard)

Premise

Program selected

T1 mode

Operation Steps

1. Put a cursor behind, and insert on a line of the instruction command
2. Input the command by keyboard.

Example:

MOVEFLOOR 100 (moving distance)

6.7.6. AXISON Display Coordinate System

Premise

Program selected

T1 mode

Operating Steps

1. Put a cursor behind, and insert on a line of the instruction command
2. Input the command by keyboard.

Example:

AXISON

6.7.7. AXISOFF Hidden Coordinate System

Premise

Program selected

T1 mode

Operating Steps

1. Put a cursor behind, and insert on a line of the instruction command
2. Input the command by keyboard

Example:

AXISOFF

6.8. Definition of Structure (programmed by keyboard)

Prerequisite

Program selected

T1 mode

Keyboard connected

Operation steps

1. Put a cursor behind, and insert on a line of motion command
2. Input the command by keyboard.

Example:

STRUC LABEL INT PARAMETER1, REAL PARAMETER2

DECL LABEL PART1 ,PART2, ,

PART1 = { PARAMETER1 10, PARAMETER2 500 }

PART2 = { PARAMETER1 20, PARAMETER2 100 }

.....

.....

6.9. Subprogram (programmed by keyboard)

Prerequisite

Program selected

T1 mode

Keyboard connected

Operation steps

1. Put a cursor behind, and insert on a line of motion command
2. Input the command by keyboard.

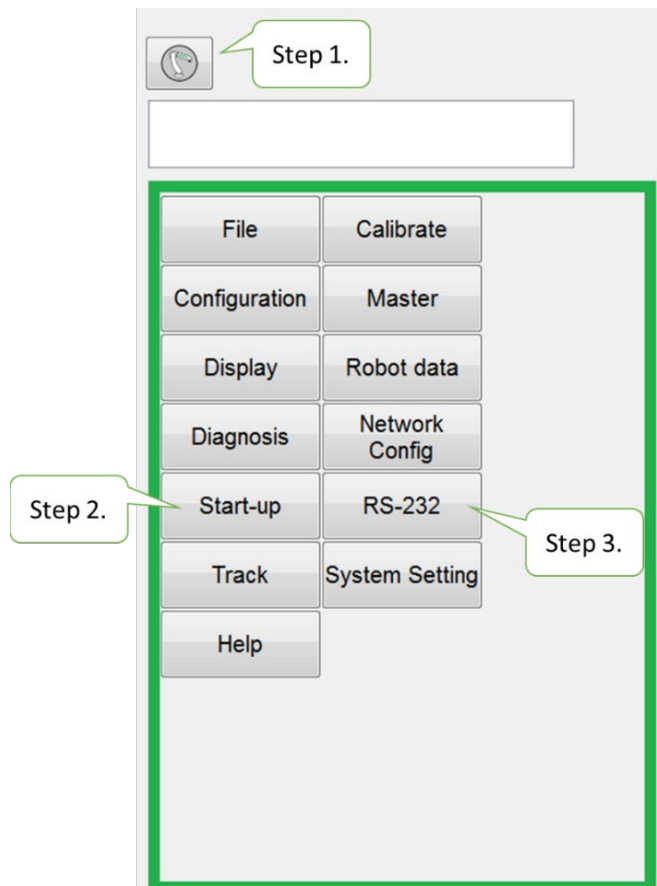
Example:

```
.....  
LABEL  
DEFFCT VOID LABEL  
.....  
ENDFCT
```

6.10. Communication Configuration

6.10.1. Using RS232 to Connect with External Equipment

- Confirm the transfer setting
Step 1. Click Main Icon
Step 2. Click Start-up
Step 3. Click RS-232



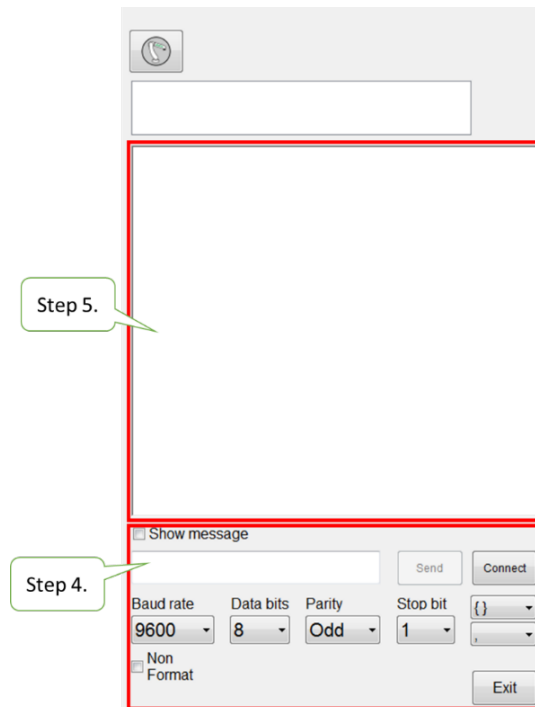
RS232 Process Setting Interface (I)

Step 4. Configure the communication and packet formats

(The default starting and ending symbols are “{” and “}”, and are separated using a comma “,”. If Non Format is selected, there will be no start and end symbols or separation symbols; data will be interpreted one by one using ASCII CODE (decimal).

Step 5. The communication data will be displayed in the middle of dialog box.

NOTE: It is possible to perform a manual test on this page to check if the connection and transfer value are successful.



RS232 Process Setting Interface (II)

● Description of Relevant Instructions

Prerequisite:

Program selected

T1 mode

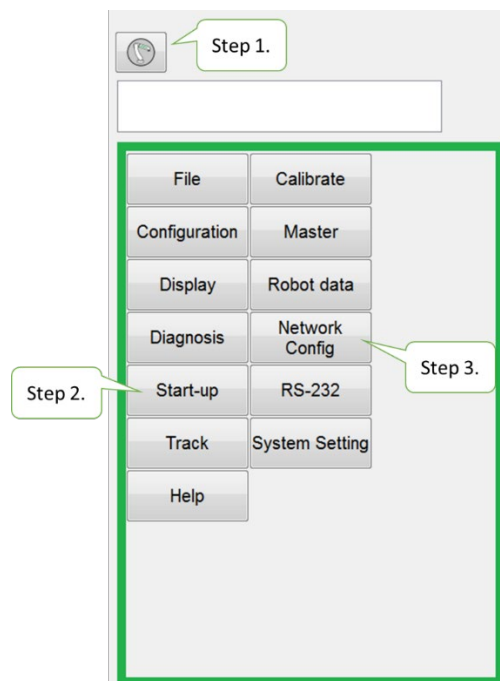
Code	Comment
<code>COPEN(SER,*NAME*)</code>	Open communication SER indicates the opened communication is RS232 Save the opened state to *NAME*
<code>CREAD(*NAME*,*val*)</code> <code>CREAD(*NAME*,*val1*,*val2*)</code>	Read and save the data to *val*, When read multiple data, it may use multiple variables *val1*, *val2* .., however, please beware that each
<code>CREAD(*NAME*,*val*)</code> <code>CREAD(*NAME*,*val1*,*val2*)</code>	CREAD instruction will read one set of packet, therefore, when the variable does not match the number of transfer value, it will take 0 as the value or

Code	Comment
<p><code>CREAD(*NAME*,*val*)</code> <code>CREAD(*NAME*,*val1*,*val2*)</code></p>	<p>be negligible.</p> <p>e.q.1 Camera : <code>send {100,200,45}</code> Robot : <code>CREAD(*NAME*,*val1*,*val2*,*val3*)</code> <code>val1 = 100 ; val2 = 200 ; val3 = 45</code></p> <p>e.q.2 Camera : <code>send {100,200}</code> Robot : <code>CREAD(*NAME*,*val1*,*val2*,*val3*)</code> <code>val1 = 100 ; val2 = 200 ;val3 = 0</code></p> <p>e.q.3 Camera : <code>send {100,200,45,50}</code> Robot : <code>CREAD(*NAME*,*val1*,*val2*,*val3*)</code> <code>val1 = 100 ; val2 = 200 ; val3 = 45</code></p>
<p><code>CCLEAR(*NAME*)</code></p>	<p>Clear the data in the temporary storage When the packet number is greater than the instruction number of CREAD, the old data will be read, therefore, CCLEAR will be added before or after the reading instruction.</p>
<p><code>CWRITE(*NAME*,*val*)</code> <code>CWRITE(*NAME*,*val1*,*val2*)</code></p>	<p>Transfer the data in *val* Multiple data *val1*, *val2*...can be sent at one time However, please be aware that val can only be real type and unable to send text, plus each CWRITE instruction is a set of packet.</p> <p>e.q.1 Robot : <code>CWRITE(*NAME*,100,200,45)</code> Camera : <code>read {100,200,45}</code></p>

Code	Comment
<pre>CWRITE(*NAME*,*val*) CWRITE(*NAME*,*val1*,*val2*)</pre>	<p>e.q.2</p> <p>Robot :</p> <pre>CWRITE(*NAME*,100) CWRITE(*NAME*,200,45)</pre> <p>Camera :</p> <pre>read {100}{200,45}</pre>

6.10.2. Use Ethernet to Connect with External Equipment

- Confirm the transfer setting
 - Step 1. Click Main Icon
 - Step 2. Click Start-up
 - Step 3. Click Network Config



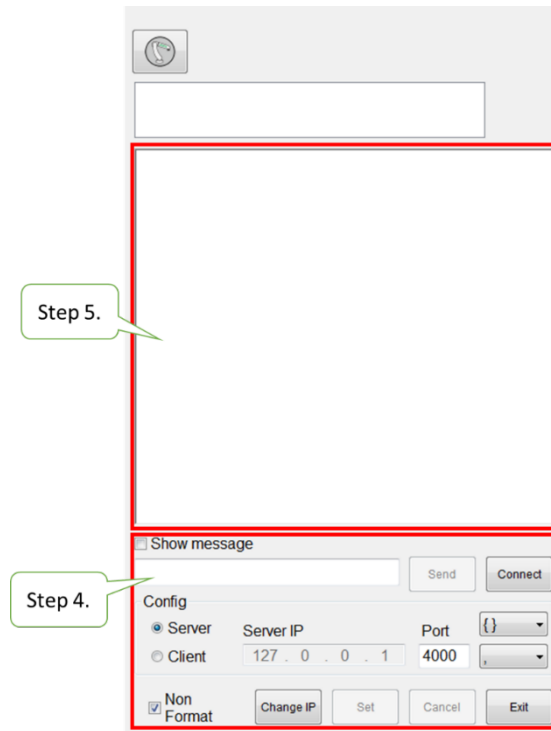
Ethernet TCP/IP Process Setting Interface (I)

Step 4. Configure the Server/Client communication method and packet format

(The default starting and ending symbols are “{” and “}” and are separated using a comma “,”. If Non Format is selected, there will be no start and end symbols or separation symbols; data will be interpreted one by one using ASCII CODE (decimal).

Step 5. The communication data will be displayed in the middle of dialog box.

NOTE : It is possible to perform a manual test on this page to check if the connection and transfer value are successful.



Ethernet TCP/IP Process Setting Interface (II)

● Description of Relevant Instructions

Prerequisite:

Program selected

T1 mode

Code	Comment
<ul style="list-style-type: none"> ● Single connection: <code>COPEN(ETH,*NAME*)</code> ● Multiple connections: <code>*NAME* = COPEN(ETH, IP_val1, IP_val2, IP_val3, IP_val4, port)</code> 	<p>Open communicatio</p> <p>ETH indicates the opened communication is Ethernet TCP/IP</p> <p>Save the opened state to *NAME*</p> <p>Using multiple connection channels; if the IP is 192.168.0.5 and the port is 5000, it is written as below: HANDLE = <code>COPEN(ETH, 192, 168, 0, 5, 5000)</code></p>
<code>CREAD(*NAME*,*val*)</code> <code>CREAD(*NAME*,*val1*,*val2*)</code>	<p>Read and save the data to *val*,</p> <p>When read mulitple data, it may use multiple variables</p>
<code>CREAD(*NAME*,*val*)</code> <code>CREAD(*NAME*,*val1*,*val2*)</code>	<p>*val1*, *val2*.., however, please beaware that each CREAD instruction will read one set of packet, therefore, when the variable does not meet the number of transfer value, it will take 0 as the value or be negligible.</p> <p>e.q.1 Camera : <code>send {100,200,45}</code></p>

Code	Comment
	<p>Robot :</p> <p>CREAD(*NAME*,*val1*,*val2*,*val3*)</p> <p>val1 = 100 ; val2 = 200 ; val3 = 45</p> <p>e.q.2</p> <p>Camera :</p> <p>send {100,200}</p> <p>Robot :</p> <p>CREAD(*NAME*,*val1*,*val2*,*val3*)</p> <p>val1 = 100 ; val2 = 200 ; val3 = 0</p> <p>e.q.3</p> <p>Camera :</p> <p>sned {100,200,45,50}</p> <p>Robot :</p> <p>CREAD(*NAME*,*val1*,*val2*,*val3*)</p> <p>val1 = 100 ; val2 = 200 ; val3 = 45</p>
<p>CCLEAR(*NAME*)</p>	<p>Clear the data in temporary storage</p> <p>When the packet number is greater than the instruction number of CREAD, the old data will be read, therefore, CCLEAR will be added before or after the reading instruction.</p>
<p>CWRITE(*NAME*,*val*)</p> <p>CWRITE(*NAME*,*val1*,*val2*)</p>	<p>Transfer the data in *val*</p> <p>Multiple data *val1*, *val2*...can be sent at one time</p> <p>However, please be aware that val can only be real type and unable to send text, plus each CWRITE instruction is a set of packet.</p>
<p>CWRITE(*NAME*,*val*)</p> <p>CWRITE(*NAME*,*val1*,*val2*)</p>	<p>e.q.1</p> <p>Robot :</p> <p>CWRITE(*NAME*,100,200,45)</p> <p>Camera :</p> <p>read{100,200,45}</p> <p>e.q.2</p> <p>Robot :</p> <p>CWRITE(*NAME*,100)</p>

Code	Comment
	CWRITE(*NAME*,200,45) Camera : read {100}{45}

6.11. Conveyor Command

6.11.1. CNV_START

Description:

Start conveyor procedures, and connect with the system. The command for other conveyor can be executed after this command, used for the start of the conveyor program.

Format:

CNV_START CNV=1

...

CNV_END CNV=1

Format description:

CNV is the parameter for the conveyor number, which can be input from 1 to 4.

6.11.2. CNV_END

Description:

End the conveyor and the connection with the system, used for the end of the conveyor program.

Format:

CNV_START CNV=1

...

CNV_END CNV=1

Format description:

CNV is the parameter for the conveyor number, which can be input from 1 to 4.

6.11.3. CNV_PICK_QUANTITY

Description:

The variables for the conveyor are used to set the maximum quantity of the object that the robot can pick every time. When the quantity reaches this value, the following pick commands will not be executed.

This variable will simultaneously affect CNV_FULL and CNV_EMPTY

Format:

CNV_PICK_QUANTITY = 1

Format description:

The variable type is positive integer. The default is 1, which can be input from 1 to 8.

6.11.4. CNV_TRIGGER_TIMES[NUM]

Description:

This is a variable for the conveyor, used when the conveyor is set as a sensor trigger. When the conveyor sensor is triggered, the robot will receive a task to pick or place. This variable can be set to increase the speed of a task after the sensor is triggered several times, require to specify a conveyor number, # as the conveyor number.

Format:

CNV_TRIGGER_TIMES[NUM] = 1

Format description:

This variable type is positive integer from 1 to 100. If no quantity is assigned, the default is 1. NUM is the number of the conveyor. Input can be from 1 to 4, and represented by CNV1 to CNV4.

6.11.5. CNV_PICK

Description:

Pick the object. Automatically waits for messages from the system when the command is given that it can pick the object. After a successful pick, the robot will return to the height that the pick started from. If the pick fails, the robot will return to the starting position.

Format:

CNV_PICK CNV=1 OBJ=1 \$DO[1] P1 Down=5.000mm CONT=50% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]

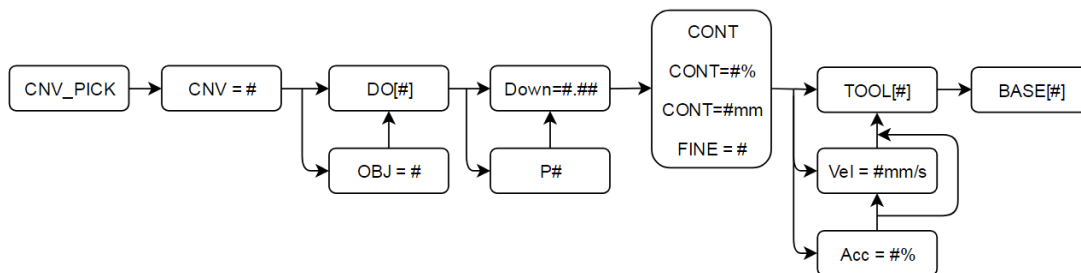
Format description:

■ CNV is the parameter for the conveyor number. When it is necessary to track the object, the CNV number

will be set. If the number is omitted, it will not track. The input range 1 to 4.

- OBJ is the object parameter, used to assign a number to the object. When it is omitted, no number is assigned to the object. All objects will be picked. The input range input is 1 to 8.
- \$DO[] is the parameter for the number O, which represents the Digital Output position to pick the object. The input range input is 1 to 48.
- P is the position parameter. The number for this position is the coordinate when the object triggers the sensor signal; if the image trigger is employed, it can be omitted.
- Down is the height that pressed downwardly to pick the object. During picking, the robot will stop a distance over the object and move downwardly. This parameter is used to assign this distance. The input range is a positive integer.
- FINE and CONT are the parameters for the discontinuous and continuous motion. The percentage behind the parameter is the smooth extent. For the description of CONT, please see the Appendix at P368.
- Vel is the velocity parameter. The default is 2000mm/s.
- Acc is the acceleration. The default is 100%.
- TOOL is the parameter for the tool coordinates, which can be used to set the position of different end tools, input is from 0 to 15.
- BASE is the parameters for base coordinate, which can be used to set the base number that the conveyor is calibrated, input is from 0 to 31.

Command flowchart:



PICK flowchart

● Flowchart description:

- # is the number.
- CNV, OBJ, P, Vel and Acc can be omitted to input.
- Please select either CONT, CONT = #%, CONT = #mm or FINE.

6.11.6. CNV_PLACE

Description:

The objects can be picked and placed or selected according to the object number or O (chosen object); the object

will return the safety height after successfully placed. When the place fails, the conveyor will return to the starting position.

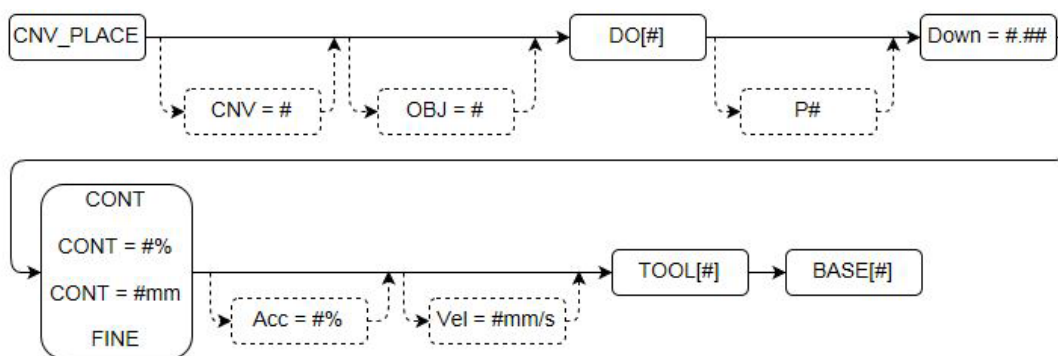
Format:

CNV_PLACE CNV=1 OBJ=1 \$DO[1] P1 Down=5.000mm CONT=100% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]

Format description:

- CNV is the parameter for the conveyor number. When it is necessary to track the object, the CNV number will be entered. If the number is omitted, it will not track. The input range is from 1 to 4.
- OBJ is the object parameter, which can be used to assign the picked object number. If it is omitted, the object number is not assigned. All objects are placed. The input range is 1 to 8.
- \$DO[] is the parameter for the number O, which represents the position to place the object in this time. If it is omitted, it will represent to place according to the object number.
- P is the position parameter, which represents the position to place the object.
- Down is the height that pressed downwardly to pick the object. During picking, the robot will stop a distance over the object and move downwardly. This parameter is used to assign this distance, which should be positive integer or 0.
- FINE and CONT are the paramaters for the discontinuous and continuous motion. The percentage behind the paramter is the smooth extent. For the description of CONT, please see the Appendix at P368.
- Vel is the velocity parameter. The default is 2000mm/s.
- Acc is the acceleration parameter. The default is 100%.
- TOOL is the parameter for tool coordinate.
- BASE is the parameter for the base coordinate.

Command flowchart:



PLACE flowchart

● Flowchart description

- # is the number.

- CNV, OBJ, P, Vel and Acc can be omitted to input.
- Please select either CONT, CONT = #%, CONT = #mm or FINE.

6.11.7. CNV_OBJECT

Description:

The variable for picking represents the latest object number picked. After the object is placed, the number will be automatically reset, which can be used to determine the current object and perform the specific action. (ATTENTION: CNV_OBJECT can be used only after CNV_PICK)

Format:

```
CNV_PICK CNV=1 $DO[1] P1 Down=5.000mm CONT=50% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]
IF CNV_OBJECT == 1 THEN
CNV_PLACE CNV=1 $DO[1] P3 Down=5.000mm CONT=100% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]
ENDIF
CNV_PLACE CNV=1 $DO[1] P2 Down=5.000mm CONT=100% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]
```

Format description:

The variable type is integer, which can be used for WHILE or IF.

6.11.8. CNV_FULL

Description:

The Boolean variable for picking (For the description of Boolean, please see the Appendix at P338) represents when the quantity of objects that have been picked by the robot reaches the upper limit. When the picking quantity has reached the value set by CNV_PICK_QUANTITY, this variable is TRUE; if the quantity doesn't reach the setting value, it will be FALSE.

Format:

```
CNV_PICK_QUANTITY = 2
WHILE CNV_FULL == FALSE
...
ENDWHILE
```

Format description:

The variable type is Boolean, which can be used for WHILE or IF.

6.11.9. CNV_EMPTY

Description:

The Boolean variable for picking (For the description of Boolean, please see the Appendix at P362) represents the quantity of the objects that have been picked by the robot. When no object is picked, this variable is TRUE; if one or more object is picked, this variable is FALSE.

Format:

```
WHILE CNV_EMPTY == FALSE
...
ENDWHILE
```

Format description:

The variable type is Boolean, and can be used for WHILE or IF.

6.11.10. CNV_SET_DELAY_TIME[NUM]

Description:

This parameter is used to set the delay time for the conveyor. By setting this variable, the robot can continue to move with the object in the specific time and leave after reaching the position to pick or place.

As shown in below, the robot will move with the object in 50ms and leave after picking or placing.

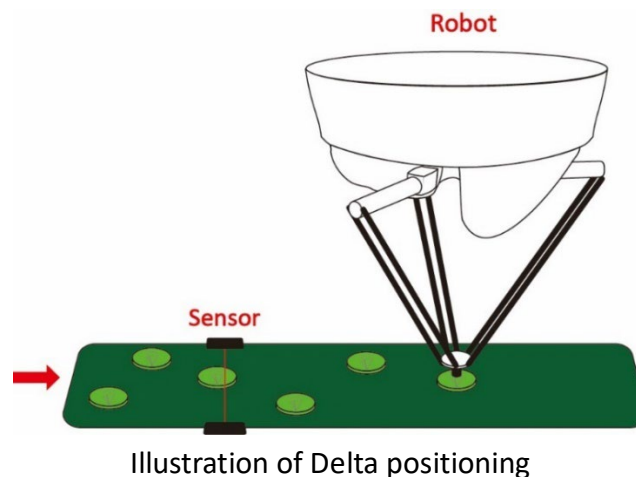
Format:

```
CNV_SET_DELAY_TIME[NUM] = 50
```

Format description:

NUM is the number of the conveyor. Input can be from 1 to 4, and represented by CNV1 to CNV4.

The variable type is positive integer. The default is 0, which can be input from 0 to 1500 with a unit of ms.



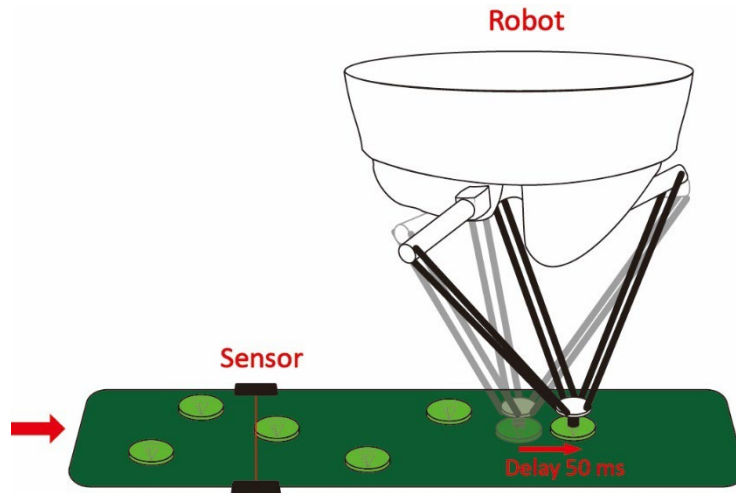


Illustration of Delta delay 50 ms

6.11.11. CNV_QUEUE_SIZE[NUM]

Description:

This is the pick variable. This variable shows the sensor has been triggered on the conveyor, but there is a quantity of object not picked.

As shown in below, the sensor for the Conveyor 2 has triggered three objects, but the robot has not picked them. Therefore, this variable is 3.

Format:

```
IF CNV_QUEUE_SIZE[NUM] > 0 THEN
```

```
...
```

```
ENDIF
```

Format description:

NUM is the number of the conveyor. Input can be from 1 to 4, and represented by CNV1 to CNV4.

The variable type is a positive integer, and can be used for WHILE or IF.

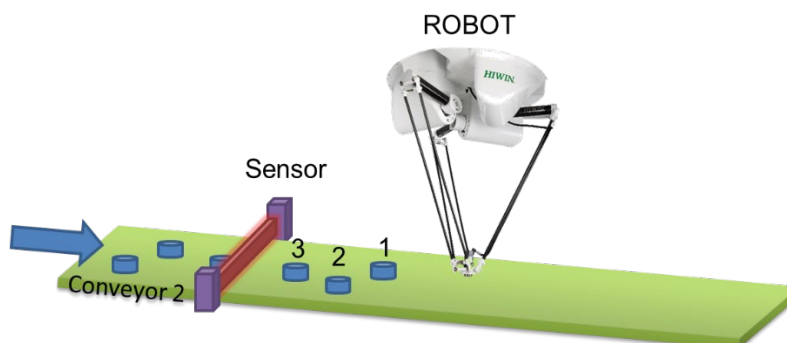


Illustration of CNV_QUEUE_SIZE

6.11.12. CNV_OBJ_CNT_DIST[NUM]

Description:

This is the conveyor variable. When the variable CNV_QUEUE_SIZE[NUM] is greater than or equal to 2 (Two or more objects on the conveyor have been triggered.) can be used immediately.

This variable can display the difference between the position of the first object and the second object triggered by the sensor from the difference in Encoder value. It is usually used to determine if the triggered objects are continuous.

Format:

```
IF CNV_QUEUE_SIZE[NUM] > 1 THEN
IF CNV_OBJ_CNT_DIST[NUM] < 2600 THEN
...
ELSE
...
ENDIF
ENDIF
```

Format Description:

NUM is the number of the conveyor. Input can be from 1 to 4, and represented by CNV1 to CNV4.
The variable type is positive integer, and can be used for WHILE or IF.

6.11.13. CNV_PLACE_BATCH[NUM]

Description:

The place variable is used when many objects are placed in the same work space.

When the sensor that releases an object is triggered, the robot will obtain a position where the object can be placed. The maximum number of times that the robot can place an object in this position can be set by this variable.

Format:

```
CNV_PLACE_BATCH[NUM] = 1
```

Format Description:

The variable type is a positive integer. If no quantity is assigned, the default is 1. The input range is 1 to 100 and represented by CNV1 to CNV4.

6.11.14. CNV_RESET_ENC

Description:

Conveyor Tracking Instruction. The user can use this instruction to clear the counting value of the external encoder when writing program.

The effect of using this instruction is same as the effect of pressing “CLEAR” on the conveyor calibration interface. (Please refer to the description of P155).

Format:

CNV_RESET_ENC

Format Description:

No need to enter parameter.

6.11.15. CNV_QUEUE_REMOVE[NUM]

Description:

Flying pick/flying place state variable. The user is able to remove the temporary value placed at the forefront of the waiting queue by using this instruction during the process of writing the program.

Format:

CNV_QUEUE_REMOVE[NUM]

Format Description:

NUM is the number of the conveyor. Input can be from 1 to 4, and represented by CNV1 to CNV4.

6.11.16. CNV_PICK_ACC[NUM]

Description:

Flying pick state variable. The user is able to configure the acceleration time of tracking push-down by using this instruction when writing program.

Format:

CNV_PICK_ACC[NUM]

Format Description:

NUM is the number of the conveyor. Input can be from 1 to 4, and represented by CNV1 to CNV4. The default value is 30, range from 10 to 100.

6.11.17. CNV_OFFSET_X[NUM]

Description:

Flying pick/flying place state variable. The user is able to configure the offset value of X by using this instruction when writing program.

Format:

CNV_OFFSET_X[NUM] = 10

Format Description:

NUM is the number of the conveyor. Input can be from 1 to 4, the offset value of X is configured as 10mm.

6.11.18. CNV_OFFSET_Y[NUM]

Description:

Flying pick/flying place state variable. The user is able to configure the offset value of Y by using this instruction when writing program.

Format:

CNV_OFFSET_Y[NUM] = 10

Format Description:

NUM is the number of the conveyor. Input can be from 1 to 4, the offset value of Y is configured as 10mm.

6.11.19. CNV_OFFSET_Z[NUM]

Description:

Flying pick/flying place state variable. The user is able to configure the offset value of Z by using this instruction during the process of writing program.

Format:

CNV_OFFSET_Z[NUM] = 10

Format Description:

NUM is the number of the conveyor. Input can be from 1 to 4, the offset value of Z is configured as 10mm.

6.11.20. CNV_SPEED[NUM]

Description:

Conveyor state variable, user is able to read the current speed of conveyor.

Format:

INT ISpeed

ISpeed = CNV_SPEED[NUM]

Format Description:

NUM is the number of the conveyor. Input can be from 1 to 4, and represented by CNV1 to CNV4.

6.12. DO switching on the path (SYN)

Prerequisite

Program selected

T1 mode

Keyboard connected

Description

TCP can be output in motion. The command for START, END and PATH can be used.

The delay time of START and END is ± 1000 ms.

The range of PATH is ± 2000 mm.

Attention: Due to the limit of software memory, it limits the count of SYN, the maximum count is eight (8) counts, if the SYN instruction is entered over 8 counts before the motion instruction, Error Code 3010 will be pop-out: SYNC_CMD_QUEUE_FULL_ERROR.

Operation steps

1. Put a cursor behind, and insert on a line of the motion command.
2. Input the command by keyboard.

Format:

1. SYN \$DO[INDEX] = STATUS START DELAY = TIME ms
2. SYN \$DO[INDEX] = STATUS END DELAY = TIME ms
3. SYN \$DO[INDEX] = STATUS PATH = DISTANCE mm DELAY = TIME ms

Format Description:

INDEX is the DO number; the range is from 1~48.

STATUS is the DO status; enter TRUE or FALSE.

TIME is the delay time; the range is -1000~1000 ms.

DISTANCE is the moving distance; the range is -2000~2000 mm.

6.13. Setting of External Procedure Input/Output

6.13.1. Mode Setting of External Procedure

Operating Steps:

1. Select Start-up>System Setting>FIO Setting °
2. After pressing Edit in the Mode column, it is possible to select the mode of FIO from the pull-down menu, the first mode is RSR, PNS is the second mode.
3. Press Save to save the setting.

6.13.2. External Procedure Function of RSR Mode

Prerequisite

EXT mode

Program edit is complete.

Select RSR mode.

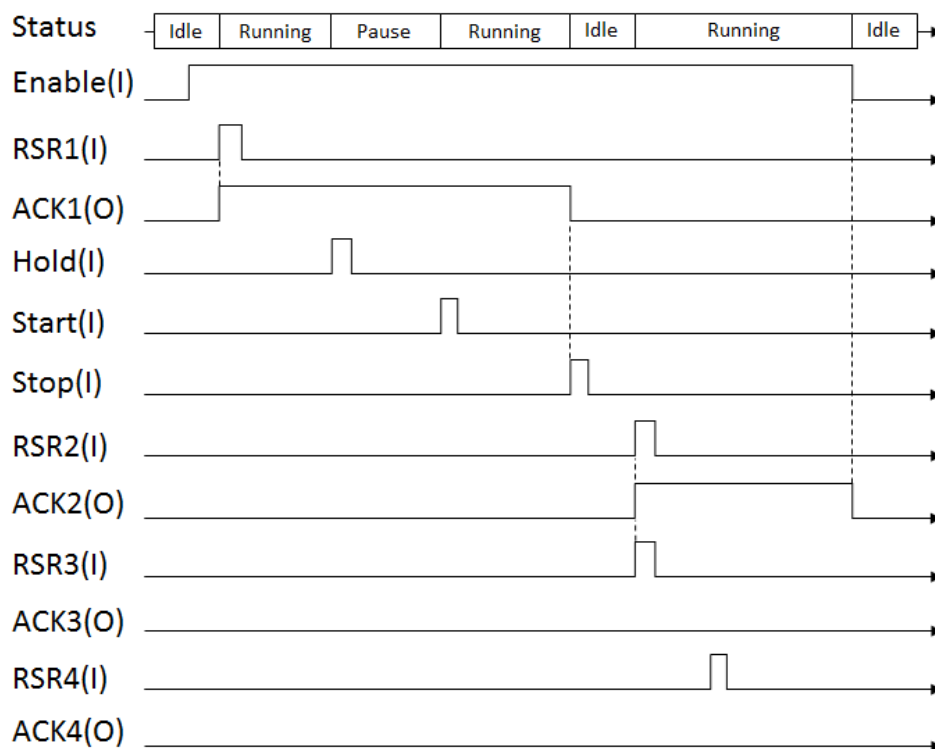
RSR (Robot Server Request): When the enable signal is enabled (Enable on), the robot will execute the movement of the preset corresponding program (name) directly according to the single signal fed back (cannot have multiple signals simultaneously); there is a total of four signals RSR1~RSR4 that can be set. For related settings, refer to the function input FI/output FO settings on the I/O page.

Operating Steps

1. Click the completed program in the program directory, press "Add to" and then add the program to the list of RSR Program.
2. In the paging of I/O, click F.I. and F.O., it is possible to observe the execution of the program triggered by exterior source.
3. When Enable of F.I. is "On", and if RSR1~RSR4 are "On", it is possible to enable the corresponding RSR Program, and execute that program.
4. ACK1~ACK4 of F.O. will be able to output the signal corresponded to F.I.
5. Double click on the column of RSR Program, it will be able to delete the program from the list.

Period Chart

1. When two RSR signal is detected simultaneously, execute the one with the lowest number, from the period chart, RSR2 and RSR3 appeared simultaneously, RSR2 will be executed while RSR3 will be ignored.
2. During execution of RSR, when other RSR signal is detected, they will be ignored, from period chart, RSR4 is detected when RSR2 is being executed, and thus it is ignored.



RSR Period Chart

6.13.3. External Procedure Function of PNS Mode

Prerequisite

EXT mode

Program edit is complete.

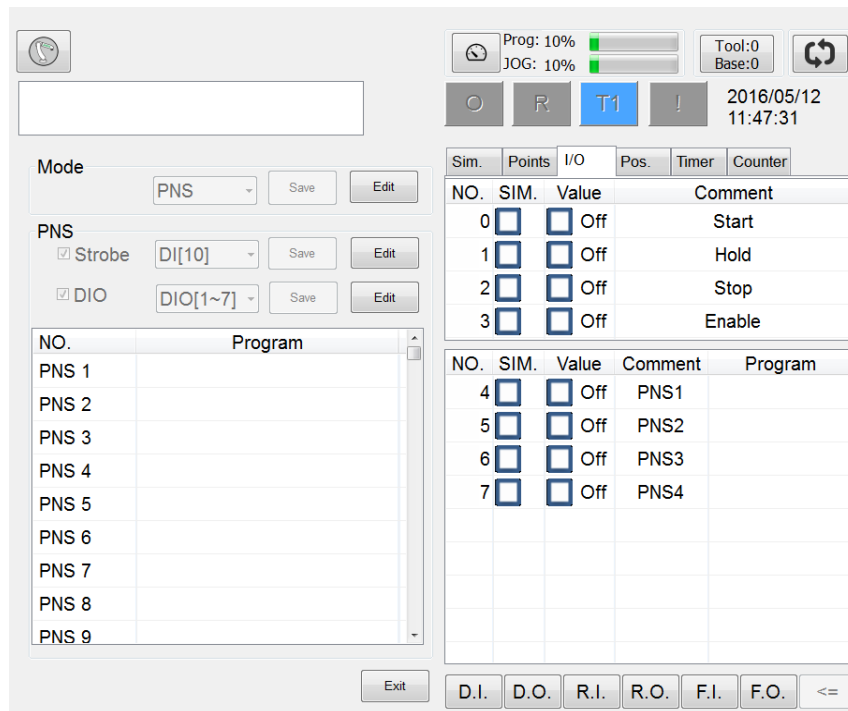
Select PNS mode.

PNS (Program Need Selection) is a single signal; change it to 2-bit judgment multi-signal, select and open the program. Expand the application to 15 programs, which are advance uses of RSR. In addition to having different opening methods, the applications are also different from RSR.

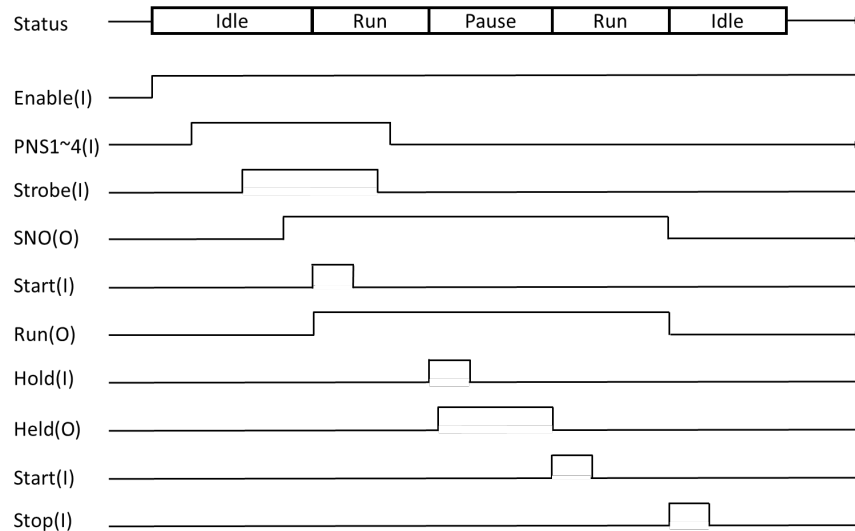
Operating Steps

1. Select the completed program in the program directory, press "Add to" and then add the program to the list of PNS Programs.
2. Select Start-up>System Setting>FIO Setting.
3. Confirm the "Mode" option is the selection of PNS mode, if not, please press "Edit" to proceed the change, and then press "Save" for saving.
4. Check "Strobe" already set.
5. In the paging of I/O, tap F.I. and F.O., it is possible to observe execution of the program triggered by exterior source.

6. When Enable of F.I. is “On”, it will then be possible to execute the relevant function of the external procedure.
7. Trigger F.I.[4~7] to choose program needed execute.
8. Set DI which is “Strobe” as On to open program selected.
9. Check program number through SNO in F.O.
10. When Start of F.I. is “On”, according to current state of I.O., it will automatically enable the program that has already been added to Program column and execute enabled program automatically.
11. When Hold of F.I. is “On”, the program under execution will be temporarily stopped, if the program must be enabled again, the state of Start must be set to “On”.
12. When Stop of F.I. is “On”, the program under execution will be stopped.
13. If “DIO” option is checked, it is possible to select D.I. as the expansion of triggering program, the highest support is up to DI [1~7] which is able to trigger 2047 types of program.
14. Double click under the column of “NO.”, to view the I.O. state that requires to trigger this program.
15. Double click on the column of Program to delete the program from the list.



FIO Setting interface



6.14. Positioning Check of Arm Position

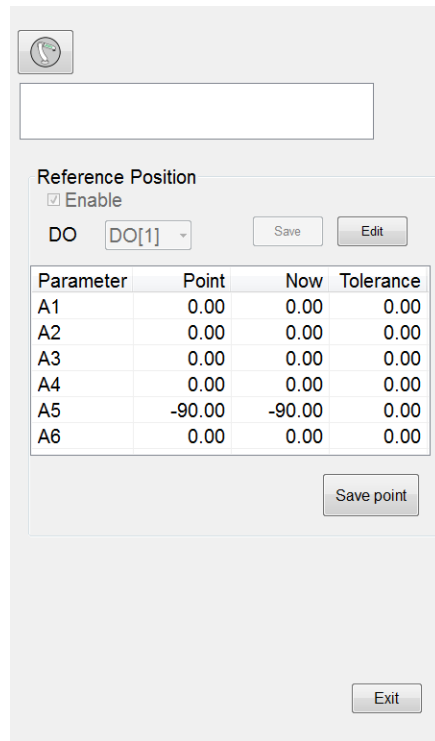
Description

User is able to define their own point position (Point column) and tolerance range (Tolerance column) of one arm, if the current angle position (Now column) of arm enters into the tolerance range of point position previously configured and when the arm is moved manually or moved by the program, it will change the specified output DO to the state of “ON”.

This function can be used to check if the position of arm has been returned to the position as predetermined by the user before enabling the program or after ending the program.

Operating Steps

1. Select Start-up>System Setting>Ref. Position
2. Click “Edit”, check Enable and configure DO, and then press “Save” for saving the setting.
3. Move the arm to the position to be configured as the check point.
4. Press “Save point” button, save the Point column with the new value.
5. Click twice on the table, it enables the tolerance range of each axis (Tolerance) to be edited.



The screenshot shows a software interface for setting reference positions. At the top left is a circular icon with a right-pointing arrow. Below it is a large empty rectangular input field. The main section is titled "Reference Position" and contains a checked "Enable" checkbox. Below the checkbox is a label "DO" followed by a dropdown menu showing "DO[1]". To the right of the dropdown are "Save" and "Edit" buttons. A table with four columns: "Parameter", "Point", "Now", and "Tolerance" is displayed. The table contains six rows of data. Below the table is a "Save point" button. At the bottom right of the interface is an "Exit" button.

Parameter	Point	Now	Tolerance
A1	0.00	0.00	0.00
A2	0.00	0.00	0.00
A3	0.00	0.00	0.00
A4	0.00	0.00	0.00
A5	-90.00	-90.00	0.00
A6	0.00	0.00	0.00

Reference Position interface

6.15. Self-defined Digital Input Control Function

The screenshot shows a software interface for configuring Digital Input/Output (DIO) settings. It is divided into two main sections: Digit Input and Digit Output.

Digit Input Section:

- Clear Error:** A dropdown menu set to 'DI' and a 'Disable' button.
- External Alarm:** A dropdown menu set to 'DI' and a 'Disable' button.
- Show Text:** A text input field containing 'User Define Alarm'.
- System Shutdown:** A dropdown menu set to 'DI' and a 'Disable' button.

Digit Output Section:

Function	IO Type	Index
Motor Warning	DO	Disable
System Start Up	DO	Disable
Mode Output	DO	Disable
Reset Driver	DO	Disable
Emergency	DO	Disable

DIO setting interface

6.15.1. Clear Error

Prerequisite

Expert user group.

Operating Steps

1. Select Start-up>System Setting>DIO Setting.
2. Select the specific D.I. from the Clear Error option, it will enable to use as the functional signal of clearing error through the configured D.I.
3. If Disable is selected, it indicates that this function is disabled.
4. Press Save to save the setting.

6.15.2. External Alarm

Prerequisite

Expert user group.

Operating Steps

1. Select Start-up>System Setting>DIO Setting
2. Select the specific D.I. from the External Alarm option, it will enable to use as the functional signal of external alarm through the configured D.I.
3. If Disable is selected, it indicates that this function is disabled.
4. Set the word to be appeared in Show Text when the alarm is triggered.
5. Press Save to save the setting.

6.15.3. External Shutdown Input

Prerequisite

Expert user group

Operation Steps

1. Select Start-up>System Setting>DIO Setting.
2. Select the specific D.I. from the System Shutdown option, it will enable to use as the functional signal of system shutdown through the configured D.I.
3. If Disable is selected, it indicates that this function is disabled.
4. Press Save to save the setting.

6.16. Self-defined Digital Output Control Function

The screenshot shows a software interface for configuring Digital Input/Output (DIO) settings. It is divided into two main sections: Digit Input and Digit Output.

Digit Input Section:

- Clear Error:** A dropdown menu set to 'DI' and a 'Disable' button.
- External Alarm:** A dropdown menu set to 'DI' and a 'Disable' button.
- Show Text:** A text input field containing 'User Define Alarm'.
- System Shutdown:** A dropdown menu set to 'DI' and a 'Disable' button.

Digit Output Section:

Function	IO Type	Index
Motor Warning	DO	Disable
System Start Up	DO	Disable
Mode Output	DO	Disable
Reset Driver	DO	Disable
Emergency	DO	Disable

DIO setting interface

6.16.1. Motor Warning

Prerequisite

Expert user group.

Operating Steps

1. Select Start-up>System Setting>DIO Setting.
2. Select the specific D.O. from the Motor Warning option, it will enable to use as the functional signal of motor warning through the configured D.O.
3. If Disable is selected, it indicates that this function is disabled.

6.16.2. System Start Up

Prerequisite

Expert user group.

Operating Steps

1. Select Start-up>System Setting>DIO Setting.
2. Select the specific D.O. from the System Start Up option, it will enable to use as the functional signal of starting up through the configured D.O.
3. If Disable is selected, it indicates that this function is disabled.

6.16.3. Teach Pendant Manual/Auto Mode Selection Output Signal Setting

Prerequisite

Expert user group.

Operating Steps

1. Select Start-up>System Setting>DIO Setting.
2. Select the specific D.O. from the Mode Output option, it will enable to use as the functional signal of selected manual/auto mode through the configured D.O.
Select T1/T2 mode, D.O. is OFF ; AUT/EXT mode, D.O. is ON.
3. If Disable is selected, it indicates that this function is disabled.

6.16.4. Reset Driver

Prerequisite

Expert user group.

Operating Steps

1. Select Start-up>System Setting>DIO Setting.
2. Select the specific D.O. from the Reset Driver option, it will enable to use as the functional signal of Reset Driver through the configured D.O., it is the same as System I/O 04.
3. If Disable is selected, it indicates that this function is disabled.

6.16.5. Emergency Output

Prerequisite

Expert user group.

Operating Steps

1. Select Start-up>System Setting>DIO Setting.
2. Select the specific D.O. from the Emergency Output option, when emergency is triggered, it will enable to use as the functional signal of Emergency Output through the configured D.O..
3. If Disable is selected, it indicates that this function is disabled.

6.16.6. Reset Safety Relay

Prerequisite

Expert user group.

Operating Steps

1. Select Start-up>System Setting>DIO Setting.
2. Select the specific D.O. from the Reset Safety Relay option, it will enable to use as the functional signal of Reset Safety Relay through the configured D.O., it is the same as System I/O 07.
3. If Disable is selected, it indicates that this function is disabled.

6.17. Setting of Motion Parameters (programmed by keyboard)

Prerequisite

Program selected

T1 mode

Keyboard connected

6.17.1. SET_OVERRIDE_SPEED

Description

Use this instruction in the program to change the speed magnification of the program during movements. The inputted parameters represent the execution speed percentage of the run; its relationship with the execution speed of the arm is the product value of the overall speed multiplied by the speed of each single line.

Using this command in the program is to change the overall speed percentage of the program, and adjust the speed to adjust the line. After completing the setting, it can be placed in the initial running speed of the program or in any stroke i.e. set the percentage of speed value.

Format

SET_OVERRIDE_SPEED 100

PTP P0 **CONT**=100% **Vel**=80% **Acc**=50% **TOOL**[0] **BASE**[0]

(The overall speed is 50% and the speed of a single line is 80%, then the final result will be 40% execution of P0 position movement)

Format Description

The variable type is a positive integer, the allowable range is from 1 to 100 and can not be 0.

6.17.2. SET_SPEED

Description

Use this instruction in the program to configure the moving speed of tangent track and circular orbit during movement.

The input parameters indicate the configured speed, unit is mm/s. Range between 1 to 6000.

Format

SET_SPEED 2000

Format Description

The variable type is a positive integer that can not be 0, different model has a different default.

WARNING

1. Using SET_SPEED instruction gives the robot a higher speed of operation. However, when the value is set too high, it may exceed the robot load and cause false alarm.
2. Please adjust the parameters according to the actual requirement of use to avoid equipment damage caused by the excessive operating speed.

6.17.3. SET_ACC

Description

Use this instruction in the program to configure the 100% acceleration time of motion.

This is the time used to accelerate to the required motion speed, the unit is ms, range between 20 to 400.

If this setting is not used, different model has a different default.

Format

SET_ACC 250

Format Description

The variable type is the positive integer which cannot be 0.

WARNING

1. Using SET_ACC instruction gives the robot a higher speed of operation. However, when the value is set too low, it may exceed the robot load and cause false alarm.
2. Please adjust the parameters according to the actual requirement of use to avoid equipment damage caused by the excessive operating speed.

6.17.4. SET_ROTATION_SPEED

Description

Use this instruction in the program to configure the speed of rotation of the gesture during the motion.

The input parameters indicate the configured speed, the unit is deg/sec.

Format

SET_ROTATION_SPEED 100

Format Description

The variable type is the positive integer which cannot be 0.

WARNING

1. Using SET_ACC instruction gives the robot a higher speed of operation. However, when the value is set too low, it may exceed the robot load and cause false alarm.
2. Please adjust the parameters according to the actual requirement of use to avoid equipment damage caused by the excessive operating speed.

6.17.5. SET_TOOL

Description

Use this instruction in the program to allow the arm to select the specified number of TOOL setting, or change the setting of current parameters of TOOL.

Format

FRAME T_ONE

T_ONE.X = 100

SET_TOOL 1

SET_TOOL T_ONE

Format Description

SET_TOOL supports the input of a positive integer and 2 types of parameter for FRAME.

The allow the range of positive integer from 0 to 15, if the positive integer is entered, it will set the arm to select specified number of Tool setting, the Tool number on the upper right corner will also be changed.

If FRAME is entered, it will change the currently selected Tool setting parameter to the value of FRAME.

6.17.6. SET_BASE

Description

Use this instruction in the program to allow the arm to select the specified number of BASE setting, or change the setting of current parameters of BASE.

Format

FRAME B_ONE

B_ONE.Y = 100

SET_BASE 1

SET_BASE B_ONE

Format Description

SET_BASE supports the input of a positive integer and 2 types of parameter for FRAME.

The allow the range of positive integer from 0 to 31, if a positive integer is entered, it will set the arm to select speified number of Base setting, the Base number on the upper right corner will also be changed.

If FRAME is entered, it will change the currently selected Base setting parameter to the value of FRAME.

6.17.7. GETPOINT

Description

Acquire the coordinate value and angle value of current position.

Format

E6POINT E6TEST

E6TEST = GETPOINT

Format Description

E6TEST acquires the coordinate value and angle value of current position.

6.17.8. GET_MOTION_STATUS

Description

Acquire the current motion status.

Format

INT Istatus

Istatus = GET_MOTION_STATUS

Format Description

Istatus acquires the value of current motion status.

0 is the idle status, 1 is the running status, 2 is the hold status.

6.17.9. BRAKE

Description

Stop and clear the motion command which contained motion queue command.

Format

LIN P1

LINP2

...

BRAKE

Format Description

When executed to BRAKE, the motion will stop.

6.17.10. EXT_TCP (Optional)

Description

This command is required when the robot takes a workpiece to perform a LIN or CIRC motion on an external tool point, such as a polishing job.

Format

EXT_TCP_START

LIN P1

LINP2

...

EXT_TCP_END

Format Description

The motion command between EXT_TCP_START and EXT_TCP_END will move as external tool point.

6.17.11. CHECK_LIN

Description

The manipulator may move to the singular point in addition to the PTP command during the movement. At this time, the manipulator will trigger the alarm to stop the manipulator. This command can be checked whether the singular point occurs between two points in advance, and perform different motion design to avoid moving to singular point and stop in the midway, this increase the efficiency of use.

Format

IF CHECK_LIN(P1,P2) == FALSE THEN

LIN P1

LIN P2

ENDIF

Format Description

Using CHECK_LIN command to determine that P1 and P2 will not be singular, then execute the movement of LIN P1 and LIN P2.

6.17.12. FWD - Forward Robot Kinematics Convert Command

Description

When users only know the angles of the various axes on a single point, they need to get the coordinate values of TCP under the current Tool and Base coordinates of that point. This instruction can be used to convert and get the coordinates.

Format

```
E6POS P_COOR  
E6AXIS P_DEGREE = {A1 0, A2 0, A3 0, A4 0, A5 -90, A6 0}
```

```
P_COOR = FWD(P_DEGREE)
```

```
$C[1] = P_COOR.X *1000  
$C[2] = P_COOR.Y *1000  
$C[3] = P_COOR.Z *1000  
$C[4] = P_COOR.A *1000  
$C[5] = P_COOR.B *1000  
$C[6] = P_COOR.C *1000
```

Format Description

The given angle points P_DEGREE of each axis is converted into Cartesian coordinate points P_COOR through the FWD instruction and its X, Y, Z, A, B and C are saved in the Counter separately.

6.17.13. STOP_MOTION - Trigger to Stop Previous Motion According to Signal

Description

This instruction is used when the movement instruction of the previous line is CONT or FINE; when the designated DI, SI or WI is triggered, the current movement will be stopped, and it will move back in the opposite direction at a speed of 1mm/s until the designated DI, SI or WI is OFF.

Format

```
LIN P0 CONT Vel=2000mm/s Acc=100% TOOL[0] BASE[0]  
STOP_MOTION $DI[1]  
LIN P1 CONT Vel=2000mm/s Acc=100% TOOL[0] BASE[0]
```

Format Description

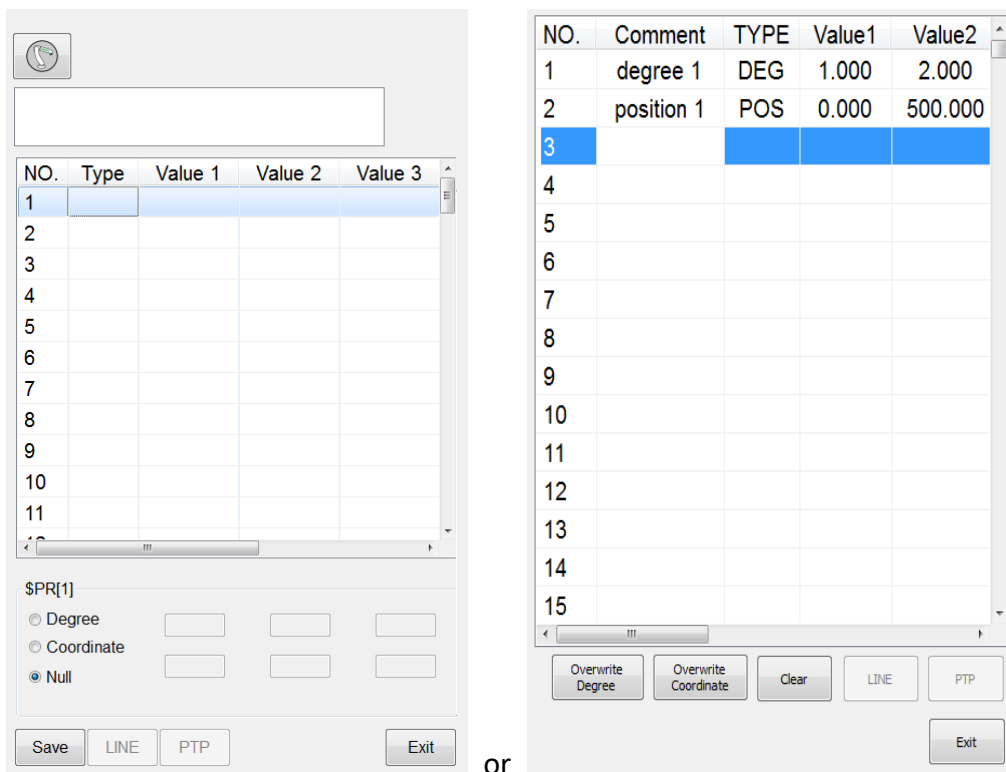
During the process when the LNE instruction is moving to point P0, if DI [1] was triggered, the current movement will stop, and it will move backwards in the opposite direction at a speed of 1mm/s until DI [1] is OFF.

6.18. Location Register

6.18.1. Using Interface to Enter Location Register

Operating Steps

1. Select Display > PR.
2. Click any column in the list.
3. User may select Degree, Coordinate or Null from the options below.
4. If angle Degree (DEG) input is selected as the type, the current A1 to A6 angles (four axes do not have A5 and A6) and the E1~E3 positions of the external axis will be filled in by default; users can edit this value directly.
5. If position coordinates (POS) input is selected as the type, the current Cartesian coordinates (X, Y, Z, A, B, C; four axes do not have A and B) and E1~E3 positions of the external axis will be filled in by default; users can edit this value directly.
6. If Null is selected, all content will be cleared.
7. Press Save to save the setting.



Location Register setting interface

6.18.2. Using Instruction to Enter Location Register

Operating Steps

1. Put a cursor behind, and insert on a line of the instruction command
2. Input the command by keyboard

Example 1:

\$PR[1] = {A1 1 , A2 2 , A3 3 , A4 4 ,A5 5 , A6 6,E1 1,E2 2,E3 3}

\$PR[2] = {X 7 ,Y 8 , Z 9 , A 10 ,B 11 , C 12,E1 1,E2 2,E3 3}

Example 2:

E6POS A = {X 10 ,Y 10 ,Z 10 ,A 10 ,B 10 ,C 10}

E6AXIS B = {A1 20 , A2 20 , A3 20 , A4 50 ,A5 10 , A6 20}

E6POINT C = { X 5 ,Y 15 ,Z 25 ,A 35 ,B 45 ,C 55}

\$PR[1] = A

\$PR[2] = B

\$PR[3] = C

Example 3:

\$PR [1] = GETPOINT

6.18.3. Use Location Register for Motion

Operating Steps

1. Put a cursor behind, and insert on a line of the instruction command
2. Input the command by keyboard

Example:

LIN \$PR [1]

LIN_REL \$PR [1]

PTP \$PR [1]

PTP_REL \$PR [1]

CIRC \$PR [1] \$PR [2]

CIRC_REL \$PR [1] \$PR [2]

Attention: TYPE of two PR used by CIRC and CIRC_REL must be the same (same DEG or same POS).

6.19. User Self-defined Alarm

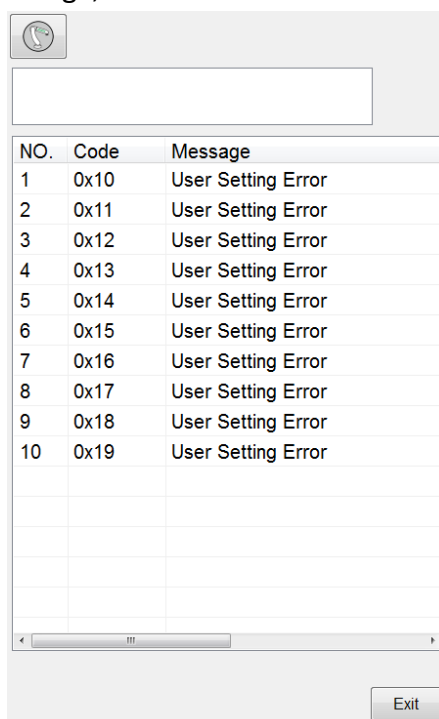
6.19.1. Content of Defined Alarm

Description

User is able to define 10 sets of alarm text content, issued by program instruction.

Operating Steps

1. Select Main menu > Start-up > System Setting > User Alarm Setting
2. Click twice on the column of Message, and then edit the alarm text content of that column.



User Alarm Setting interface

6.19.2. Issue Self-defined Alarm (programmed by keyboard)

Description

If the user requires an alarm based on the self-determined condition during the execution of program, it is possible to use this instruction function to issue an alarm while the program is running.

When using this instruction to issue the alarm, it has the function as temporary motion stop, the "Start" button will be used to continue executing the operation.

Format

USER_ALARM [n]

Format Description

The variable type is a positive integer, from 1 to 10 and cannot be 0.

6.20. Electric Gripper Command

6.20.1. Command Description

In HRSS, there are 8 command in total related to XEG series electric gripper for user to program, each command will be executed or completed first in order to continue the next execution.

6.20.1.1. EG_OPEN (str Type)

Description

Connect XEG series electric gripper. Before executing any command to operate the electric gripper, this command must be executed to open the connection with the electric gripper.

Format

EG_OPEN(Type)

Format Description

Type will be representing the model of the XEG series electric gripper, the code for each model will be represented below:

Code	Electric Gripper Model
X16	XEG-16
X32	XEG-32
X32PR	XEG-32-PR
X48	XEG-48
X64	XEG-64

6.20.1.2. EG_CLOSE

Description

Close the connection for current XEG electric gripper. Can be used to close the current electric gripper and switch on connection for other model of XEG electric gripper.

Format

EG_CLOSE

Format Description

No other parameters

6.20.1.3. EG_RESET

Description

Execute reset for XEG series electric gripper. After the connection with electric gripper is completed, the reset has to be performed to ensure that each control parameters are read and write.

Format

EG_RESET
EG_RESET CONT

Format Description

No other parameters

6.20.1.4. EG_GET_STATUS

Description

Obtain the status of XEG series electric gripper, each status code are as followed:

Code	Status of Electric Gripper	
0	Idle	Ready
1	In action	Busy
2	Grip	Hold
3	In position	Pos
-1	Abnormal position	Alarm 1
-2	Over trip	Alarm 2
-3	Unusual origin return	Alarm 3

Used to determine whether the electric gripper is currently grip on to an object or in an action as one of the basis of object recognition.

Format

```
IF EG_GET_STATUS == 2 THEN
...
ENDIF
```

Format Description

Using IF condition to determine the status of XEG series electric gripper in order to execute different operation.

6.20.1.5. EG_RUN_MOVE

Description

Execute the movement of XEG series electric gripper. According to the speed set by the user to control the electric gripper to move to the specified position (absolute coordinates).

Format

EG_RUN_MOVE(MovPos, MovSpeed)

Format Description

- **MovPos** Moving electric gripper to the specified position, minimum unit 0.01mm.
- **MovSpeed** Movement speed of the electric gripper, minimum unit 1mm/s.

Setting range for each electric gripper as followed:

Model of Electric Gripper	MovPos	MovSpeed
XEG-16	0~16 (mm)	0~60 (mm/s)
XEG-32	0~32 (mm)	0~80 (mm/s)
XEG-32-PR	0~32 (mm)	0~80 (mm/s)
XEG-48	0~48 (mm)	0~80 (mm/s)
XEG-64	0~64 (mm)	0~100 (mm/s)

Inputting CONT after the instruction means that the instructions will not wait until the movement of the robot arm or other electric claws to stop and will continue executing downwards.

6.20.1.6. EG_RUN_GRIP

Description

Execute the gripping movement of XEG series electric gripper. According to the direction, displacement, speed and force of gripping set by the user to control the electric gripper for operating gripping action (relative coordinates).

Format

EG_RUN_GRIP(Dir, Str, GriSpeed ,GriForce)

Format Description

- **Dir** Electric gripper moving direction, C represent moving inwards, O represent moving outwards.
- **Str** Displacement of the grip, minimum unit as 1mm;
(XEG-16 range(0~16mm) 、 XEG-32 range(0~32mm) 、 XEG-32-PR range(0~32mm) 、 XEG-48 range(0~48mm) 、 XEG-64 range(0~64mm))
- **GriSpeed** Speed of the grip, L represent Low, M represent Medium, H represent High
- **GriForce** Force of the grip, L represent Low, M represent Medium, H represent High

Inputting CONT after the instruction means that the instruction will not wait until the movement of the robot arm or other electric claws to stop, and will continue executing downwards.

6.20.1.7. EG_RUN_EXPERT

Description

Execute gripping movement of XEG series electric gripper in expert mode. According to the direction, displacement, speed and force of gripping set by the user to control the electric gripper for operating fast movement and slow grip (relative coordinates).

Format

EG_RUN_EXPERT(Dir, MovStr, MovSpeed, GriStr, GriSpeed, GriForce)

Format Description

- **Dir** Electric gripper moving direction, C represent moving inwards, O represent moving outwards.
- **MovStr** Movement displacement, minimum unit 0.01mm.
- **MovSpeed** Movement speed, minimum unit 1mm/s.
- **GriStr** Displacement of the grip, minimum unit 1mm/s
- **GriSpeed** Speed of the grip, minimum unit 1mm/s
- **GriForce** Force of the grip, minimum unit 5%

Setting range for each electric gripper as followed:

Model of Electric Gripper	MovStr	MovSpeed	GriStr	GriSpeed	GriForce
XEG-16	0~16 (mm)	0~60 (mm/s)	0~16 (mm)	1~10 (mm/s)	50~100%
XEG-32	0~32 (mm)	0~80 (mm/s)	0~32 (mm)	1~20 (mm/s)	40~100%
XEG-32-PR	0~32 (mm)	0~60 (mm/s)	0~32 (mm)	1~10 (mm/s)	50~100%
XEG-48	0~48 (mm)	0~80 (mm/s)	0~48 (mm)	1~20 (mm/s)	50~100%
XEG-64	0~64 (mm)	0~100 (mm/s)	0~64 (mm)	1~20 (mm/s)	40~100%

Inputting CONT after the instruction means that the instructions will not wait until the movement of the robot arm or other electric claws to stop and will continue executing downwards.

6.20.1.8. EG_GET_POS

Description

Obtain the position of XEG series electric gripper, minimum unit 0.01mm. Used to confirm whether the electric gripper has moved to the specified position or within a range, as one of the basis of object recognition.

Format

IF EG_GET_POS > 5.00 AND EG_GET_POS < 7.00 THEN

...

ENDIF

Format Description

Using IF condition to determine the position of XEG series electric gripper in order to execute different operation.

6.21. Infinite Rotation Command (Optional)

6.21.1. CT_A6

Description

Execute the infinite rotation of the sixth axis of the robot, this command must be used with Keypro.

Format

CT_A6 velocity_ratio

Format Description

velocity_ratio is the ratio of infinite rotation speed, the input range is -100 to 100, the sign indicates the direction of rotation, and the input 0 stops the infinite rotation.

6.22. External Axis Command

6.22.1. EX_AX

Description :

Enable/disable the connected external axis. Users can use this command to enable the corresponding external axis to set and control it, or disable the corresponding external axis to cancel its related functions.

Format :

EX_AX[AXIS_NUM] = TRUE

Format Description

AXIS_NUM is the external axis number, the range is 1 ~ 3

6.22.2. EX_AX_ASYNC

Description :

Set the external axis to be a synchronous axis or an asynchronous axis. You can use this command to switch the

motion mode of the external axis to perform synchronous or asynchronous related control.

Format :

EX_AX_ASYNC[AXIS_NUM] = FALSE

Format Description

AXIS_NUM is the external axis number, the range is 1 ~ 3

6.22.3. EX_AX_SET_ACC

Description :

Set the acceleration and deceleration time of the external axis, you can use this command to modify the acceleration and deceleration of the specified external axis.

Format :

EX_AX_SET_ACC[AXIS_NUM] 100

Format Description

AXIS_NUM is the external axis number, the range is 1 ~ 3

6.22.4. EX_AX_SET_SPEED

Description :

Set the speed of the specified external axis, the linear axis unit is mm / sec, and the rotary axis unit is deg / sec. Will NOT change the speed of PTP and CT_A6 command when using in hrb program.

Format :

EX_AX_SET_SPEED[AXIS_NUM] 100

Format Description

AXIS_NUM is the external axis number, the range is 1 ~ 3

6.22.5. EX_AX_CT

Description :

Perform the infinite rotation of the external axis at the specified infinite rotation speed of the external axis (the external infinite rotation function must be enabled first).

Format :

EX_AX_CT[AXIS_NUM] VEL_RATIO

Format Description

AXIS_NUM is the external axis number, the range is 1 ~ 3.

VEL_RATIO is the infinite rotation speed ratio, the range is -100 ~ 100, the sign indicates the direction of rotation, enter 0 to stop the infinite rotation.

6.22.6. ASYPTP

Description :

When the external axis is set as an asynchronous axis, you can use this command to control it execute point-to-point motion.

Format :

ASYPTP{E1 60} CONT=100% Vel=100% Acc=50%

ASYPTP{E1 60, E2 100} CONT=100% Vel=100% Acc=50%

Format Description

ASYPTP can input the position of E1 ~ E3, and set its smoothing, speed and acceleration and deceleration settings.

6.22.7. PTP

Description :

When the external axis is set as an synchronize axis, you can use this command to control it execute point-to-point motion.

Format :

E6POINT P1{A1 50, E1 60}

PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0]BASE[0]

PTP{A1 40, E1 50} CONT=100% Vel=100% Acc=50% TOOL[0]BASE[0]

PTP{X 100, E1 50} CONT=100% Vel=100% Acc=50% TOOL[0]BASE[0]

Format Description

PTP instruction can increase the position of input E1 ~ E3.

6.22.8. LIN

Description :

When the external axis is set as a synchronous axis, this command can be used to control the synchronous linear motion of the robot and the external axis.

Format :

E6POINT P1{A1 50, E1 60}

LIN P1 CONT=100% Vel=2000mm/s Acc=50% TOOL[0]BASE[0]

LIN{A1 40, E1 50} CONT=100% Vel=2000mm/s Acc=50% TOOL[0]BASE[0]

LIN{X 100, E1 50} CONT=100% Vel=2000mm/s Acc=50% TOOL[0]BASE[0]

Format Description

LIN instruction can increase the position of input E1 ~ E3.

6.22.9. CIRC

Description :

When the external axis is set as the synchronous axis, this command can be used to control the synchronous arc motion of the robot and the external axis.

Format :

E6POINT P1{A1 50, E1 60}

E6POINT P2{X 100, E1 0}

CIRC P1 P2 CONT=100% Vel=2000mm/s Acc=50% TOOL[0]BASE[0]

CIRC{A1 50, E1 60} {X 100, E1 0}CONT=100% Vel=2000mm/s Acc=50% TOOL[0]BASE[0]

CIRC{A1 50, E1 60} {X 100, E1 0} CONT=100% Vel=2000mm/s Acc=50% TOOL[0]BASE[0]

Format Description

CIRC instruction can increase the position of input E1 ~ E3.

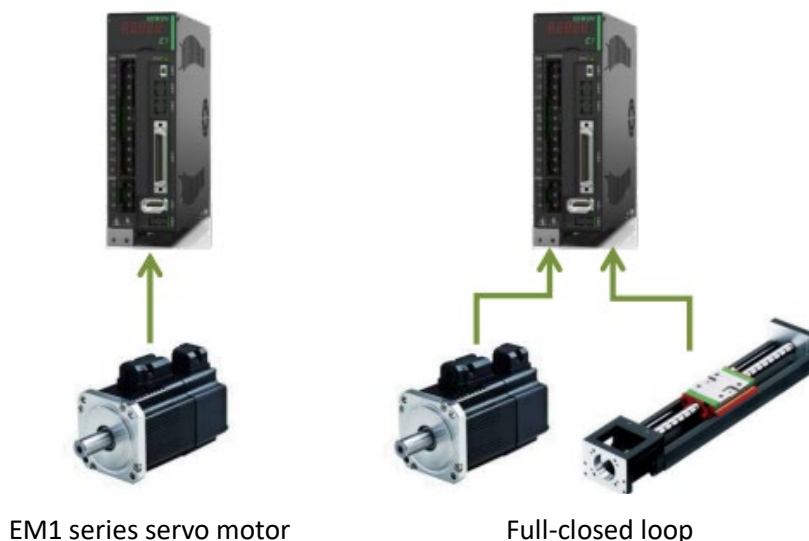
6.22.10. E1 Motor Selection Specifications

E1 driver specifications can be determined according to the table below; currently supported driver models are bus type EtherCAT control interfaces (the table refers to MD09UC01-2110, E1 Series Servo Drive User Manual V2.1 (TW).pdf Page. 2-2)

Code	1	2	3	4	-	5	6	-	7	8	9	10	-	11	12	-	13	14
Example	E	D	1	S	-	V	G	-	0	4	2	2	-	0	1	-	0	0
1, 2, 3: E1 Series Servo Drive	ED1																	
4: Type	S = Standard									F = Fieldbus								
5: Control Interface	V = Voltage command and pulse									E = EtherCAT (CoE) H = mega-ulink (with HIWIN MoE HIMC motion controller or API/MPI motion control command library) L = MECHATROLINK-III P = PROFINET								
6: Special Function	G = Gantry N = No special function																	
7, 8: Rated Output	04 = 400 W 05 = 500 W 10 = 1 kW 12 = 1.2 kW 20 = 2 kW 40 = 4 kW 50 = 5 kW 75 = 7.5 kW																	
9: AC Phase	2 = Single/Three-phase (For 400 W / 500 W / 1 kW / 1.2 kW / 2 kW model) 3 = Three-phase (For 4 kW / 5 kW / 7.5 kW model)																	
10: AC Power	2 = 110 V/220 V (100 Vac ~ 240 Vac) 3 = 400 V (380 Vac ~ 480 Vac)																	
11: Applicable Category	0 = AC, LM, DM, and TM A = AC only T = GT																	
12: Safety Version	1 = STO function security approval																	
13, 14: Reserved	Reserved																	

Refer to the following table for the AC motor models that can be used with the E1 driver
(The table refers to MD09UC01-2110, E1 Series Servo Drive User Manual V2.1 (TW).pdf Page. 2-5)

Servo drive and servo motor combination



Note:

- (1) : If servo drive for AC (ED1□-□□-□□22-A□) is used, full-closed loop internal encoder only supports EM1 series AC servo motor.
- (2): □ or □□ represents cable length. Please fill in Part No. based on cable length.
- (3): For the information of applicable servo motors and cables, please refer to section 16.1.1 and 16.1.2.

The allowable combinations of servo drives and servo motors are listed in table below.

Servo Motor Model	Capacity	Servo Drive	Main Circuit Power
EM1-□-□-05-2	50 W	ED1□-□□-0422-A□	Single-phase or Three-phase AC 200 ~ 240 V
EM1-□-□-10-2	100 W		
EM1-□-□-20-2	200 W		
EM1-□-□-40-2	400 W	ED1□-□□-1022-A□	Single-phase or Three-phase AC 200 ~ 240 V
EM1-□-□-75-2	750 W		
EM1-□-□-1K-2	1 kW	ED1□-□□-2022-A□	Single-phase or Three-phase AC 200 ~ 240 V
EM1-□-□-1A-2	1.2 kW		
EM1-□-□-2K-2	2 kW		

6.23. Welding Command

6.23.1. WELD_START

Description :

Welding start instruction, welding in accordance with the specified welding procedures and conditions.

Format :

WELD_START WP = procedure number WS= schedule number

Format Description

procedure number is the set welding procedure number, the range is 1 ~ 20

schedule number is the set welding condition number, the range is 1 ~ 32

6.23.2. WELD_END

Description :

Welding end instruction, according to the specified welding procedures and conditions for crater processing and end welding.

Format :

WELD_END WP = procedure number WS= schedule number

Format Description

procedure number is the set welding procedure number, the range is 1 ~ 20

schedule number is the set welding condition number, the range is 1 ~ 32

6.23.3. WEAVE

Description :

According to the pre-set weaving conditions or directly specify the frequency, amplitude, left stop and right stop conditions when executing weaving, the command to start the weaving in the specified mode, the remaining settings will be based on the settings of the pendulum welding setting interface.

Format :

WEAVE pattern WS = schedule number

WEAVE pattern Freq = frequency Hz Amp= amplitude mm LD = left dwell sec RD= right dwell sec

Format Description

pattern is the weaving mode, you can input SINE, CIRCLE and L
schedule number is the number of the weaving condition, the range is 1 ~ 10
frequency is the weaving frequency
amplitude is the amplitude of weaving
left dwell is the left dwell time
right dwell is the right dwell time

6.23.4. WEAVE_END

Description :

End the current weaving motion.

Format :

WEAVE_END

Format Description

No parameters

6.24. Interrupt instructions

HRSS interrupt instructions RESUME_PROG are used to set the start label of the interrupt program and CLEAR_RESUME_PROG set the end label of the interrupt program. If the LIN instruction between these two instructions generates an alarm or manually stops the program, the original program has the function of re-execution. The related instructions are in T1. , T2, AUT mode is valid.

6.24.1. RESUME_PROG

Description :

Set the start label of the range of the interrupt reset program to be executed. If the program is not specified, only the original program will continue to be interrupted. If the specified program is specified, the original program will be executed after the interrupted program, and then the original interrupt program will be executed. In this case, an instruction to return to the original interrupt position will be automatically added when the reset program is executed, and the adjustment will be made according to the setting.

Format :

RESUME_PROG

RESUME_PROG = resume program

Format Description

Resume program indicates the name of the resume program to be executed after the original program is interrupted

6.24.2. CLEAR_RESUME_PROG

Description :

Set the end tag of the range of the interrupt reset program to be executed. The program after this line of instruction will no longer have the interrupt reset function, unless the RESUME_PROG instruction is programmed.

Format :

CLEAR_RESUME_PROG

Format Description

None.

6.25. Collision Detection command

6.25.1. CD_START

Description :

Use this command in the program to enable the collision detection function and set the sensitivity in the program.

The input parameter represents the sensitivity of the operation.

Format :

CD_START

CD_START 0

CD_START 80

Format Description

The variable type is a positive integer, and the input range is from 1 to 100. Cannot be 0.

6.25.2. CD_END

Description :

Use this command in the program to turn off the collision detection function in the program.

Format :

CD_END

Format Description

None.

6.26. Payload setting command

6.26.1. SET_PAYLOAD

Description :

Use this command in the program to change the load setting in the program.
The input parameter represents the number of load data.

Format :

SET_PAYLOAD[1]

Format Description

The variable type is a positive integer, and the input range is 1-20.

6.27. String Function

6.27.1. String Cascade

Description :

String cascade

Format :

str3 = str1 + str2

Format Description :

Let str1 concatenate str2 and save to str3

Example :

STRING str1 = "abc"

STRING str2 = "def"

STRING str3

str3 = str1 + str2

;str3 == "abcdef"

6.27.2. STRCMP

Description :

Compare two strings. If two strings are the same, return true. If two strings are not the same, return false.

Format :

BOOL STRCMP(String str1, String str2)

Format Description :

Parameter:STRING Return: BOOL

Example :

```
STRING str1 = "abcd"  
STRING str2 = "abcd"  
BOOL flag  
flag = STRCMP (str1, str2)  
;flag == true
```

6.27.3. STRPOS

Description :

Search str2 in str1 and return the pos which is found in str1.

Format :

INT STRPOS (String str1, String str2)

Format Description :

Parameter:STRING Return: INT

Example :

```
flag = COMPARE(str1, str2) STRING str1 = abcde  
STRING str2 = cd  
INT pos  
pos = STRPOS (str1,str2)  
;pos == 3  
pos = STRPOS (str,"ac")  
;pos == 0
```

6.27.4. STRCPY

Description :

Copy string with specific pos and length from source string and save to destination string.

Format :

VOID STRCPY (STRING dst, STRING source, INT pos, INT length)

Format Description :

Parameter:STRING, INT

Example :

```
STRING str1 = "aaaaaaa"  
STRING str2 = "abcdefg"  
STRCPY (str1,str2,2,3)  
;str1 == "bcd"
```

6.27.5. STRLEN

Description :

Return length of string

Format :

INT STRLEN (STRING str)

Format Description :

Parameter:STRING Return: INT

Example :

```
STRING str = "abcde"  
INT strLength  
strLength = STRLEN(str)  
;strLength == 5
```

6.27.6. INTTOSTR

Description :

Return result which is string casted from integer value.

Format :

STRING INTTOSTR(INT num)

Format Description :

Parameter: INT Return: STRING

Example :

INT num = 5

STRING str

str = INTTOSTR(num)

;str == 5

6.27.7. REALTOSTR

Description :

Return result which is string casted from floating point value.

Format :

STRING REALTOSTR(REAL num)

Format Description :

Parameter: REAL Return: STRING

Example :

REAL num = 5.5

STRING str

str = REALTOSTR(num)

;str == 5.5

6.27.8. STRTOINT

Description :

Return result which is INT casted from string.

Format :

INT STRTOINT(STRING str)

Format Description :

Parameter:STRING Return:INT

Example :

```
STRING str = "20"  
INT num  
num = STRTOINT(str)  
;num == 20
```

6.27.9. STRTOREAL

Description :

Return result which is REAL casted from string.

Format :

```
REAL STRTOREAL(STRING str)
```

Format Description :

Parameter:STRING Return: REAL

Example :

```
STRING str = "130.600"  
REAL num  
num = STRTOREAL(str)  
;num == 130.600
```

6.28. Interrupt

6.28.1. Declaration

Description :

Declare priority, trigger event, and subprogram of interrupt

Format :

INTERRUPT Priority WHEN Event DO Subprogram

Format Description :

Priority: INT 1~8 represent the priority of interrupt. 1 is the highest priority.

Event: DI as an input signal to trigger interrupt ex. \$DI[1] == true

Subprogram: the activity when interrupt is triggered.

Example :

```
INTERRUPT 1 WHEN $DI[1] == TRUE DO PRO_1()
```

6.28.2. Enable and Disable

Description :

Enable or disable an interrupt or all interrupt.

Format :

INTERRUPT ON/OFF priority/ALL

Format Description :

Priority: INT 1~8 represent the priority of interrupt. 1 is the highest priority.

ON: Enable

OFF: Disable

Example :

```
INTERRUPT ON 1      ;Enable interrupt with 1 priority
INTERRUPT OFF 2     ; Disable interrupt with 2 priority
INTERRUPT ON ALL    ; Enable all interrupt
INTERRUPT OFF ALL   ; Disable all interrupt
```

6.28.3. Functional limitations and specifications

1. Stop action immediately when an interrupt occurs.
2. When the interrupt subprogram is executed, if other interrupts are triggered, it will be scheduled and execute according to the priority.
3. Interrupt is global and needs to be declared in the main program.
4. When an interrupt finishes, it will resume the command which was interrupted. However, the robot's original position will not be recovered. If the user needs the robot's original position to be recovered to the status which was before the interrupt, the user should record the point (using GETPOINT) when the interrupt starts and use PTP or LIN to this point before the interrupt finishes.
5. When multiple interrupts are triggered, the robot will execute the interrupt subprogram according to priority, not according to trigger time.
6. When the program is paused or stopped, an interrupt will not be triggered.
7. Interrupt trigger events only support DI signals.
8. Interrupt subprograms do not support parameter input or output.

6.29. Multiple(1~3) Dimension Array

6.29.1. Declare

Description :

Support INT 、 REAL 、 CHAR 、 E6POS 、 E6POINT 、 E6AXIS data type to use array

Format :

Type Name[num1] ; 1 dimension array
Type Name[num1,num2] ; 2 dimension array
Type Name[num1,num2,num3] ; 3 dimension array

Format Description :

Type: INT 、 REAL 、 CHAR 、 E6POS 、 E6POINT 、 E6AXIS

Name: name of variable

num: quantity of the elements

Example :

```
INT i1DArr[10]
INT i2DArr[10,10]
INT i3DArr[10,10,10]
REAL rArr[10,10,10]
CHAR cArr[10,10,10]
E6POS PosArr[10,10,10]
```

E6POINT PointArr[10,10,10]

E6AXIS AXISArr[10,10,10]

6.29.2. Assign And Access Elements of an Array

Description :

Assign or access an element value of array.

Format :

Name[Index] = value

value = Name[Index]

Format Description :

Name: name of variable

Index: The index of elements of array. The index of the first element of array is 1

Example :

i2Darr[1] = 10

i2Darr[1,1] = 10

i3Darr[1,1,1] = 10

rArr[2,1,2] = 2.5

PointArr[3,6,3].x = 500

6.29.3. Pass Array as Function Parameter

Description :

User can pass array to subprogram or subfunction through parameter IN and OUT, but user cannot transfer array by subfunction RETRUN.

Example :

INT ARRAY_IN[10]

INT ARRAY_OUT[10]

PROG1(ARRAY_IN, ARRAY_OUT)

DEF PROG1(num_in[10]:IN, num_out[10]:OUT)

INT num_in[10]

INT num_out[10]

INT index

FOR index = 1 to 10 step 1

```
num_out[index] = num_in[index] + 1  
ENDFOR  
END
```

6.30. PULSE Command

6.30.1. PULSE(time)

Description :

This command support \$DO output pulse signal, and it will stop pre-read command. Motion commands will not be influenced during which pulse command is executing. If there are other commands which can operate IO when pulse command is doing, other commands will have higher priority.

Format :

\$DO[index] = PULSE(time)

Format information :

index: the number of \$DO

time: pulse output time

time: default 0.1sec

time: minimum time 0.1 sec

time: maximum time 30 sec

Command limitation :

Only support three pulse output simultaneously.

Pulse will stop output when QUIT is called.

INTERRUPT will not stop pulse output.

Emergency stop, user stop, and alarm will not stop pulse output

Example :

PTP P0

\$DO[1] = PULSE(10)

PTP P1



6.31. Timeout command

6.31.1. TIMEOUT(time, label)

Description :

This command can assign waiting time for wait-input command and handle time out situation to avoid robot unexpectable error.

Format :

WAIT FOR \$DI[index] == TRUE TIMEOUT(time, label)

Format information :

Index: Index of \$DI

time: waiting time

time: default 0.1sec

time: minimum time 0.1sec

time: maximum time 86400sec (one day)

label: label for handling time out event

Example :

```
LOOP
  WAIT FOR $DI[1] == TRUE TIMEOUT(10,ERROR)
ENDLOOP
ERROR:
USER_ALARM[1]
```

6.32. P# Command

6.32.1. P#(index)

Description :

When index is 1, P#(index) is P1; when index is 2, P#(index) is P2.

Format :

PTP P#(index)

Format information :

index: the index of point

Command limitation :

Only support point which is generate by teaching. Point variable, \$PR will not be supported by this command.

Example :

```
INT i = 0
FOR i = 0 TO 100 STEP 1
PTP P#(i)
ENDFOR
```

6.32.2. P#_EXIST(index)

Description :

This command can check the point if it exists, it will return true when point exists, or it will return false when point does not exist.

Example :

```
INT i = 0
FOR i = 0 TO 100 STEP 1
IF P#_EXIST(i) THEN
PTP P#(i)
ENDIF
ENDFOR
```

6.33. Torque Limit Monitoring

6.33.1. TL_SET - Set Torque Limit Monitoring Parameter Commands

Format :

TL_SET(INT A1_TL, INT A2_TL, INT A3_TL, INT A4_TL, INT A5_TL, INT A6_TL, INT E1_TL, INT E2_TL, INT E3_TL)

Description :

The torque limit value of each axis can be set separately. Under the open of torque monitoring function, when the torque of the axis reaches the limit value, then the robot motion is stopped. Setting 「 -1 」 indicates that the torque limit monitoring is not enabled for this axis.

Parameter	Description
A1_TL	Torque limit value for axis 1 of the robot: The enable value is 1~300 and enter -1 to disable Unit: %
A2_TL	Torque limit value for axis 2 of the robot: The enable value is 1~300 and enter -1 to

Parameter	Description
	disable Unit: %
A3_TL	Torque limit value for axis 3 of the robot: The enable value is 1~300 and enter -1 to disable Unit: %
A4_TL	Torque limit value for axis 4 of the robot: The enable value is 1~300 and enter -1 to disable Unit: %
A5_TL	Torque limit value for axis 5 of the robot: The enable value is 1~300 and enter -1 to disable Unit: %
A6_TL	Torque limit value for axis 6 of the robot: The enable value is 1~300 and enter -1 to disable Unit: %
E1_TL	Torque limit value for external axis 1 of the robot: The enabled value is 1~300 and enter -1 to disable Unit: %
E2_TL	Torque limit value for external axis 2 of the robot: The enabled value is 1~300 and enter -1 to disable Unit: %
E3_TL	Torque limit value for external axis 3 of the robot: The enabled value is 1~300 and enter -1 to disable Unit: %

Example :

TL_SET(50, -1, 20, 30, -1, -1, 100, -1, -1)

Example Description :

Set the torque limit value of axis 1 to 50%, torque limit value of axis 3 to 20%, torque limit value of axis 4 to 30%, limit value of external axis 1 to 100% and disable the remaining axes.

6.33.2. TL_START -Enable Torque Limit Monitoring

Description :

Start executing the torque limit monitoring function; this instruction must first have the torque limit value set using TL_SET.

Example :

TL_SET(50, -1, 20, 30, -1, -1, 100, -1, -1)

TL_START

LIN P1 FINE Vel=100mm/s Acc=50% TOOL[0] BASE[0]
TL_END

Example Description :

Set the torque limit value of axis 1 to 50%, torque limit value of axis 3 to 20%, torque limit value of axis 4 to 30%, limit value of external axis 1 to 100%, disable the remaining axes and execute the torque limit monitoring function.

6.33.3. TL_END -End Torque Limit Monitoring

Description :

Ends the torque limit monitoring function.

Example :

TL_SET(50, -1, 20, 30, -1, -1, 100, -1, -1)
TL_START
LIN P1 FINE Vel=100mm/s Acc=50% TOOL[0] BASE[0]
TL_END

Example Description :

Set the torque limit value of axis 1 to 50%, torque limit value of axis 3 to 20%, torque limit value of axis 4 to 30%, limit value of external axis 1 to 100%, disable the remaining axes, execute the torque limit monitoring function, and monitor whether torque limit was exceeded while moving to point P1.

7. Error Message

The error message with * symbol, on behalf of this error does not stop robot.

7.1. Robot System Software(01-XX-XX)

7.1.1. System Error Message(01-01-XX)

Error code	Error	Message	Reason	Solution
01-01-10	System initialization failure	System initialization failure	Software damaged or lost	1.Check the drive EtherCAT connection status. 2.Please turn off the power and then restart. 3.Please contact the engineer from manufacturer.
01-01-11	Motion library load failure	Motion library load failure	Motion library damaged or lost	
01-01-12	Motion library initialization failure	Motion library initialization failure		
01-01-13	Motion library memory initialization failure			
01-01-14	Motion library start failure	Motion library start failure		
01-01-20	EtherCAT library loading failure	EtherCAT library loading failure	Software damaged	
01-01-21	EtherCAT disconnection	EtherCAT anomalies	EtherCAT connection anomalies	
01-01-22	EtherCAT initialization failure			

Error code	Error	Message	Reason	Solution
01-01-23	EtherCAT line crossing alarm	EtherCAT anomalies	EtherCAT connection anomalies	1.Check the drive EtherCAT connection status. 2.Please turn off the power and then restart. 3.Please contact the engineer from manufacturer.
01-01-24	EtherCAT none slave alarm			
01-01-25	EtherCAT can't check slave			
01-01-26	EtherCAT slave none response			
01-01-27	EtherCAT cycle alarm			
01-01-28	EtherCAT cycle jitter			
01-01-29	EtherCAT cycle counter error			

Error code	Error	Message	Reason	Solution
01-01-2A	EtherCAT cycle watchdog error	EtherCAT anomalies	EtherCAT connection anomalies	<ol style="list-style-type: none"> 1.Check the drive EtherCAT connection status. 2.Please turn off the power and then restart. 3.Please contact the engineer from manufacturer.
01-01-2B	EtherCAT INIT switching error			
01-01-2C	EtherCAT PREOP switching error			
01-01-2D	EtherCAT SAFEOP switching error			
01-01-2E	EtherCAT OP switching error			
01-01-2F	EtherCAT master none response			
01-01-30	EtherCAT master initialization error			
01-01-31	EtherCAT busbar scan error	EtherCAT anomalies	EtherCAT connection anomalies	<ol style="list-style-type: none"> 1.Check the drive EtherCAT connection status. 2.Please turn off the power and then restart. 3.Please contact the engineer from manufacturer.
01-01-32	EtherCAT frame response error			
01-01-33	EtherCAT frame lost			
01-01-34	EtherCAT master counter error of initialization command			
01-01-35	EtherCAT master response error of initialization command			
01-01-36	EtherCAT slave counter error of initialization command			
01-01-37	EtherCAT slave response error of initialization command			
01-01-38	EtherCAT mailbox time out			
01-01-39	EtherCAT mailbox SDO cancel			
01-01-3A	EtherCAT mailbox COE			

Error code	Error	Message	Reason	Solution
	counter receive error		connection anomalies	1.Check the drive EtherCAT connection status. 2.Please turn off the power and then restart. 3.Please contact the engineer from manufacturer.
01-01-3B	EtherCAT mailbox COE counter send error			
01-01-3C	EtherCAT mailbox receive invalid data			
01-01-3D	EtherCAT master alarm			
01-01-3E	System error	memory configuration	Out of memory	1. Please turn off the power and then restart. 2. Please contact the engineer from manufacturer.

Error code	Error	Message	Reason	Solution
01-01-40	Axis 1 parameter setting fail	System anomalies	Software damaged or lost	If reinstall software is required, please contact engineer from the original equipment manufacturer.
01-01-41	Axis 2 parameter setting fail			
01-01-42	Axis 3 parameter setting fail			
01-01-43	Axis 4 parameter setting fail			
01-01-44	Axis 5 parameter setting fail			
01-01-45	Axis 6 parameter setting fail			
01-01-50	Conveyor 1 encoder initial fail	System anomalies	Software damaged or lost	If reinstall software is required, please contact engineer from the original equipment manufacturer.
01-01-51	Conveyor 2 encoder initial fail			
01-01-52	Conveyor 3 encoder initial fail			
01-01-53	Conveyor 4 encoder initial fail			
01-01-54	External parameter initial fail	System anomalies	Software damaged or lost	If reinstall software is required, please contact engineer from

Error code	Error	Message	Reason	Solution
01-01-55	HRSS Loading fail			the original equipment manufacturer.
01-01-56	Failed to open TeamViewer	Failed to open TeamViewer	TeamViewerSoftware not installed correctly	Need to reinstall software, please contact engineer from the original equipment manufacturer.
01-01-57	HRSS last shutdown error	HRSS last shutdown abnormality	<p>1. The HRSS is not turned off properly, and the HRSS is not turned off when the program is stopped.</p> <p>2. Directly cut off the main power instead of turning off the controller.</p>	<p>1. Please confirm whether the data is stored correctly, and avoid turning off the HRSS in the same way (the HRSS is not turned off when the program is stopped, or the main power is turned off instead of turning off the controller power).</p> <p>2. If the program has stopped and the main power is not cut off directly, please contact the engineer to check and repair the controller.</p>
01-01-58	FBWF memory consumption 128MB	FBWF memory consumption 128MB	FBWF anti-write memory is full to 128 MB	User needs to reboot
01-01-59	FBWF memory consumption 512MB	FBWF memory consumption 512MB	FBWF anti-write memory is full to 512 MB	User needs to reboot
01-01-5A	FBWF file failed to open	FBWF file failed to open	File damage	Confirm that the file is damaged
01-01-60	E-CAT device error	Inconsistent number of EtherCAT Input devices	EtherCAT Input device is abnormal.	<p>1. Please confirm EtherCAT connection status</p> <p>2. Please turn off the power and then restart.</p> <p>2. Please contact the engineer from manufacturer.</p>
01-01-61		Inconsistent order of EtherCAT Input devices		
01-01-62		Inconsistent number of EtherCAT device inputs		

Error code	Error	Message	Reason	Solution
01-01-63	E-CAT device error	Inconsistent number of EtherCAT output devices	EtherCAT Input device is abnormal.	1. Please confirm EtherCAT connection status 2. Please turn off the power and then restart. 2. Please contact the engineer from manufacturer.
01-01-64		Inconsistent order of EtherCAT output device		
01-01-65		Inconsistent number of EtherCAT device outputs		
01-01-70	E-CAT Axis-1 disconnected	E-CAT Axis-1 disconnected	EtherCAT Axis-1 driver disconnected	Please confirm EtherCAT connection status
01-01-71	E-CAT Axis-2 disconnected	E-CAT Axis-2 disconnected	EtherCAT Axis-2 driver disconnected	Please confirm EtherCAT connection status
01-01-72	E-CAT Axis-3 disconnected	E-CAT Axis-3 disconnected	EtherCAT Axis-3 driver disconnected	Please confirm EtherCAT connection status
01-01-73	E-CAT Axis-4 disconnected	E-CAT Axis-4 disconnected	EtherCAT Axis-4 driver disconnected	Please confirm EtherCAT connection status
01-01-74	E-CAT Axis-5 disconnected	E-CAT Axis-5 disconnected	EtherCAT Axis-5 driver disconnected	Please confirm EtherCAT connection status
01-01-75	E-CAT Axis-6 disconnected	E-CAT Axis-6 disconnected	EtherCAT Axis-6 driver disconnected	Please confirm EtherCAT connection status
01-01-76	E-CAT Axis-E1 disconnected	E-CAT Axis-E1 disconnected	EtherCAT Axis-E1 driver disconnected	Please confirm EtherCAT connection status
01-01-77	E-CAT Axis-E2 disconnected	E-CAT Axis-E2 disconnected	EtherCAT Axis-E2 driver disconnected	Please confirm EtherCAT connection status
01-01-78	E-CAT Axis-E3 disconnected	E-CAT Axis-E3 disconnected	EtherCAT Axis-E3 driver disconnected	Please confirm EtherCAT connection status
01-01-79	Single axis excitation/de-excitation setting error	Single axis excitation/de-excitation setting error	Single axis excitation/de-excitation instruction failure	Please confirm whether the excitation/de-excitation instruction can be used for this axis
01-01-7A	Movement axis not excited	Movement axis not excited	Movement axis in not excited status	Please excite the movement axis excitation
01-01-7B	Single axis excitation/de-excitation instruction timeout	Single axis excitation/de-excitation instruction	driver is abnormal.	Please confirm driver connection status

Error code	Error	Message	Reason	Solution
		timeout		
01-01-7C	Direction of the end of arm must face downwards	Direction of the end of arm must face downwards	Direction of the end of the palletizing machine arm must face downwards	Please change the direction of the end of arm
01-01-80	Set infinite rotation for the external axis failed	Set infinite rotation for the external axis failed	The external axis is set to enable coupled coordinated movement, and cannot be set as an infinite rotation function.	Please disable the coordinated movement of the external axis first and then set the infinite rotation function.
01-01-81	The external axis must be set to synchronous mode	The external axis must be set to synchronous mode	Not set to synchronized mode correctly; movement abnormal.	Please set the external axis as synchronous mode.
01-01-82	The external axis must be set to asynchronous mode	The external axis must be set to asynchronous mode	Was not set as asynchronous mode correctly; movement abnormal.	Please set the external axis as asynchronous mode.
01-01-83	External axis movement type error	External axis movement type error	External axis movement type setting error.	If it was originally set as linear axis, please change it to rotation axis; if it was originally set as rotation axis, please change it to linear axis.
01-01-84	External axis tracking number error	External axis tracking number error	External axis tracking currently only supports E2 axis tracking E1; tracking number setting error.	The correct setting is to set the E2 axis to track the E1 axis.
01-01-85	External axis motion base error	External axis motion base error	Motion base of the external axis could not be correctly set.	Please confirm whether the calibration point for the coordinated coupling of the external axis is correct.

7.1.2. Program Error(01-02-XX)

Error code	Error	Message	Reason	Solution
01-02-10	Program code incorrectness	Program code format incorrect	Syntax error.	Check robot language. Ref 9.1.11
	Program code incorrectness	Start Symbol should less than two	Start Symbol should less than two	Set the correct starting symbol
	Program code incorrectness	Split Symbol should less than two	Split Symbol should less than two	Set the starting symbol for the delimiter
	Program code incorrectness	End Symbol should less than two	End Symbol should less than two	Set the starting symbol for the end
	Program code incorrectness	Format error	1.It is set as both the starting symbol and ending symbol simultaneously 2.Delimiter not set Set the symbol as matching format	1.Set the start symbol and end symbol 2.Set the delimiter Confirm that the symbol input conforms to the hexadecimal ASCII CODE
01-02-11	Try to repair the corrupted file. Please confirm the program content is correct before execute.	Program file open failure.	Files are damaged or lost.	Use backup file or create new file.
01-02-12	Program copy error	Program file copy error	Program file copy error	Please export the history record and send it back to original factory for analysis.
01-02-13	Program label name not defined	Program label name not defined	Program label name not defined	Please define the program label

7.1.3. Motion Error(01-03-XX)

Error code	Error	Message	Reason	Solution
01-03-10	Axis 1 following error too big	Axis 1 position over deviation	Motion speed too fast or actual position exceeded deviation	1. Reduce speed ◦ 2. Reduce load. 3. Reduce acceleration.
01-03-11	Axis 2 following error too big	Axis 2 position over deviation		
01-03-12	Axis 3 following error too big	Axis 3 position over deviation		
01-03-13	Axis 4 following error too big	Axis 4 position over deviation		
01-03-14	Axis 5 following error too big	Axis 5 position over deviation		
01-03-15	Axis 6 following error too big	Axis 6 position over deviation		
01-03-16	Axis 1 position overlimit of positive	Axis 1 exceeded positive rotation limit	Motion to Axis 1 reach positive limit	Axis 1 move negative
01-03-17	Axis 1 position overlimit of negative	Axis 1 exceeded negative rotation limit	Motion to Axis 1 reach negative limit	Axis 1 move positive
01-03-18	Axis 2 position overlimit of positive	Axis 2 exceeded positive rotation limit	Motion to Axis 2 reach positive limit	Axis 2 move negative
01-03-19	Axis 2 position overlimit of negative	Axis 2 exceeded negative rotation limit	Motion to Axis 2 reach negative limit	Axis 2 move positive
01-03-1A	Axis 3 position overlimit of positive	Axis 3 exceeded positive rotation limit	Motion to Axis 3 reach positive limit	Axis 3 move negative
01-03-1B	Axis 3 position overlimit of negative	Axis 3 exceeded negative rotation limit	Motion to Axis 3 reach negative limit	Axis 3 move positive
01-03-1C	Axis 4 position overlimit of positive	Axis 4 exceeded positive rotation limit	Motion to Axis 4 reach positive limit	Axis 4 move negative
01-03-1D	Axis 4 position overlimit	Axis 4 exceeded	Motion to Axis 4	Axis 4 move

Error code	Error	Message	Reason	Solution
	of negative	negative rotation limit	reach negative limit	positive
01-03-1E	Axis 5 position overlimit of positive	Axis 5 exceeded positive rotation limit	Motion to Axis 5 reach positive limit	Axis 5 move negative
01-03-1F	Axis 5 position overlimit of negative	Axis 5 exceeded negative rotation limit	Motion to Axis 5 reach negative limit	Axis 5 move positive
01-03-20	Axis 6 position overlimit of positive	Axis 6 exceeded positive rotation limit	Motion to Axis 6 reach positive limit	Axis 6 move negative
01-03-21	Axis 6 position overlimit of negative	Axis 6 exceeded negative rotation limit	Motion to Axis 6 reach negative limit	Axis 6 move positive

Error code	Error	Message	Reason	Solution
01-03-30	XY coordinate overlimit of software	XY coordinates reached the limit	Motion to XY coordinate limit	Clear error and move in opposite limit direction
01-03-31	Joint overspeed	Shaft over speed	Reverse solution to determine a shaft speed too fast.	Clear error and use PTP motion
01-03-32	Wrist singularity	Near wrist singular point	Near wrist singular point	Try to avoid the singular point of motion
01-03-33	Shoulder singularity	Near shoulder singular point	Near shoulder singular point	
01-03-34	Elbow singularity	Near elbow singular point	Near elbow singular point	
01-03-40	Circle command 3 reference points on the same line	Circle command on the same line	Command setting error.	Check CIRC description.
01-03-41	Circle comm can't found center point	Unable to calculate center of circle in two point space	Command setting error.	Check CIRC description.

Error code	Error	Message	Reason	Solution
01-03-42	Circle comm can't calculate transpose matrix	Circle command parameter error, unable to calculate transpose matrix		
01-03-50	Synchronize output queue overflow	Synchronize output command buffer overflow	Synchronize output command too much, causing buffer overflow	1. Please check if the connecting line is correctly connected, and turn off the power and then re-start. 2. Please contact engineer from the original equipment manufacturer.
01-03-51	Synchronize output overlimit	Synchronize output control command overlimit	Synchronize activate output command too much	
01-03-52	Found motion command when compliance teaching	During compliance tuning, send motion command	Motion command cannot be performed during compliance tuning.	Clear error and stop sending motion command
01-03-53	Axis 1 happen collision	Robot happen collision	1. Collision 2. False alarm	1. Clear error and check if the action caused collision 2. Please contact engineer from the original equipment manufacturer.
01-03-54	Axis 2 happen collision			
01-03-55	Axis 3 happen collision			
01-03-56	Axis 4 happen collision			
01-03-57	Axis 5 happen collision			
01-03-58	Axis 6 happen collision			
01-03-59	E1 Axis happen collision			
01-03-5A	E2 Axis happen collision			
01-03-5B	E3 Axis happen collision			
01-03-60	Payload data processing error	The data is abnormal	1. Missing data 2. The data in the file is incorrect	1. Execute payload estimation again 2. Please contact engineer from the original equipment manufacturer.
01-03-61	Payload identification file is abnormal		1. Missing data 2. Open file error	
01-03-62	The identified parameter is			

Error code	Error	Message	Reason	Solution
	abnormal			
01-03-63	Axis-1 excessive acceleration	Axis-1 excessive acceleration	Axis-1 excessive acceleration	Please slow down your arm speed
01-03-64	Axis-2 excessive acceleration	Axis-2 excessive acceleration	Axis-2 excessive acceleration	Please slow down your arm speed
01-03-65	Axis-3 excessive acceleration	Axis-3 excessive acceleration	Axis-3 excessive acceleration	Please slow down your arm speed
01-03-66	Axis-4 excessive acceleration	Axis-4 excessive acceleration	Axis-4 excessive acceleration	Please slow down your arm speed
01-03-67	Axis-5 excessive acceleration	Axis-5 excessive acceleration	Axis-5 excessive acceleration	Please slow down your arm speed
01-03-68	Axis-6 excessive acceleration	Axis-6 excessive acceleration	Axis-6 excessive acceleration	Please slow down your arm speed
01-03-70	Axis-1 exceeded the lower limit of the software	The position of axis-1 exceeds the software lower limit setting value	1. The end position command issued exceeded the setting value for the software lower limit of axis 1 2. The position of the robot while moving exceeded the setting value for the software lower limit of axis 1	1. Adjust the movement instruction position and avoid issuing commands that exceed the software limit movement direction.
01-03-71	Axis-1 exceeded the upper limit of the software	The position of axis-1 exceeds the software upper limit setting value	1. The end position command issued exceeded the setting value for the software upper limit of axis 1 2. The position of the robot while moving exceeded the setting value for the software upper limit of axis 1	2. Adjust the software limit setting value so that the robot movement range does not exceed the setting value.
01-03-72	Axis-2 exceeded the lower limit of the software	The position of axis-2 exceeds the software lower limit setting value	1. The end position command issued exceeded the setting value for the software lower limit of axis 2 2. The position of the robot while moving exceeded the setting value for the software lower limit of axis 2	1. Adjust the movement instruction position and avoid issuing commands that exceed the software limit movement direction. 2. Adjust the software limit setting value so that the robot

Error code	Error	Message	Reason	Solution
01-03-73	Axis-2 exceeded the upper limit of the software	The position of axis-2 exceeds the software upper limit setting value	1. The end position command issued exceeded the setting value for the software upper limit of axis 2 2. The position of the robot while moving exceeded the setting value for the software upper limit of axis 2	movement range does not exceed the setting value.
01-03-74	Axis-3 exceeded the lower limit of the software	The position of axis-3 exceeds the software lower limit setting value	1. The end position command issued exceeded the setting value for the software lower limit of axis 3 2. The position of the robot while moving exceeded the setting value for the software lower limit of axis 3	1. Adjust the movement instruction position and avoid issuing commands that exceed the software limit movement direction.
01-03-75	Axis-3 exceeded the upper limit of the software	The position of axis-3 exceeds the software upper limit setting value	1. The end position command issued exceeded the setting value for the software upper limit of axis 3 2. The position of the robot while moving exceeded the setting value for the software upper limit of axis 3	2. Adjust the software limit setting value so that the robot movement range does not exceed the setting value.
01-03-76	Axis-4 exceeded the lower limit of the software	The position of axis-4 exceeds the software lower limit setting value	1. The end position command issued exceeded the setting value for the software lower limit of axis 4 2. The position of the robot while moving exceeded the setting value for the software lower limit of axis 4	1. Adjust the movement instruction position and avoid issuing commands that exceed the software limit movement direction.
01-03-77	Axis-4 exceeded the upper limit of the software	The position of axis-4 exceeds the software upper limit setting value	1. The end position command issued exceeded the setting value for the software upper limit of axis 4 2. The position of the robot while moving exceeded the setting value for the software upper limit of axis 4	2. Adjust the software limit setting value so that the robot movement range does not exceed the setting value.

Error code	Error	Message	Reason	Solution
01-03-78	Axis-5 exceeded the lower limit of the software	The position of axis-5 exceeds the software lower limit setting value	1. The end position command issued exceeded the setting value for the software lower limit of axis 5 2. The position of the robot while moving exceeded the setting value for the software lower limit of axis 5	1. Adjust the movement instruction position and avoid issuing commands that exceed the software limit movement direction.
01-03-79	Axis-5 exceeded the upper limit of the software	The position of axis-5 exceeds the software upper limit setting value	1. The end position command issued exceeded the setting value for the software upper limit of axis 5 2. The position of the robot while moving exceeded the setting value for the software upper limit of axis 5	2. Adjust the software limit setting value so that the robot movement range does not exceed the setting value.
01-03-7A	Axis-6 exceeded the lower limit of the software	The position of axis-6 exceeds the software lower limit setting value	1. The end position command issued exceeded the setting value for the software lower limit of axis 6 2. The position of the robot while moving exceeded the setting value for the software lower limit of axis 6	1. Adjust the movement instruction position and avoid issuing commands that exceed the software limit movement direction.
01-03-7B	Axis-6 exceeded the upper limit of the software	The position of axis-6 exceeds the software upper limit setting value	1. The end position command issued exceeded the setting value for the software upper limit of axis 6 2. The position of the robot while moving exceeded the setting value for the software upper limit of axis 6	2. Adjust the software limit setting value so that the robot movement range does not exceed the setting value.
01-03-7C	Tool center exceeded the software lower limit of X	Position of tool center exceeded the setting value for the software lower limit of X	1. The end position command issued exceeded the setting value for the software lower limit of X 2. The position of the robot while moving exceeded the setting value for the software lower limit of X	1. Adjust the movement instruction position and avoid issuing commands that exceed the software limit movement direction. 2. Adjust the software limit setting value so that the robot

Error code	Error	Message	Reason	Solution
01-03-7D	Tool center exceeded the software upper limit of X	Position of tool center exceeded the setting value for the software upper limit of X	1. The end position command issued exceeded the setting value for the software upper limit of X 2. The position of the robot while moving exceeded the setting value for the software upper limit of X	movement range does not exceed the setting value.
01-03-7E	Tool center exceeded the software lower limit of Y	Position of tool center exceeded the setting value for the software lower limit of Y	1. The end position command issued exceeded the setting value for the software lower limit of X 2. The position of the robot while moving exceeded the setting value for the software lower limit of X	1. Adjust the movement instruction position and avoid issuing commands that exceed the software limit movement direction. 2. Adjust the software limit setting value so that the robot movement range does not exceed the setting value.
01-03-7F	Tool center exceeded the software upper limit of Y	Position of tool center exceeded the setting value for the software upper limit of Y	1. The end position command issued exceeded the setting value for the software upper limit of X 2. The position of the robot while moving exceeded the setting value for the software upper limit of X	1. Adjust the movement instruction position and avoid issuing commands that exceed the software limit movement direction.
01-03-80	Tool center exceeded the software lower limit of Z	Position of tool center exceeded the setting value for the software lower limit of Z	1. The end position command issued exceeded the setting value for the software lower limit of X 2. The position of the robot while moving exceeded the setting value for the software lower limit of X	2. Adjust the software limit setting value so that the robot movement range does not exceed the setting value.
01-03-81	Tool center exceeded the software upper limit of Z	Position of tool center exceeded the setting value for the software upper limit of Z	1. The end position command issued exceeded the setting value for the software upper limit of Z 2. The position of the robot while moving exceeded the setting value for the software	1. Adjust the movement instruction position and avoid issuing commands that exceed the software limit movement direction. 2. Adjust the software limit setting value so

Error code	Error	Message	Reason	Solution
			upper limit of Z	that the robot movement range does not exceed the setting value.
01-03-82	Tool center exceeded the software lower limit of the radius	Position of tool center exceeded the setting value for the software lower limit of the radius	1. The end position command issued exceeded the setting value for the software lower limit of the radius 2. The position of the robot while moving exceeded the setting value for the software lower limit of the radius	
01-03-83	Tool center exceeded the software upper limit of the radius	Position of tool center exceeded the setting value for the software upper limit of the radius	1. The end position command issued exceeded the setting value for the software upper limit of the radius 2. The position of the robot while moving exceeded the setting value for the software upper limit of the radius	1. Adjust the movement instruction position and avoid issuing commands that exceed the software limit movement direction. 2. Adjust the software limit setting value so that the robot movement range does not exceed the setting value.
01-03-84	Instruction TL_SET execution error	Instruction TL_SET execution error	1. Torque limit detection limit range setting error 2. Setting limit parameter issued while stopping the execution of torque limit detection	Refer to the software manual and use instructions correctly
01-03-85	Instruction TL_START execution error	Instruction TL_START execution error	Start instruction issued while torque limit parameters not set	Refer to the software manual and use instructions correctly
01-03-86	Excitation timeout	Excitation timeout	Excitation timeout	Please confirm the status of the motor and the driver
01-03-87	Torque of axis 1 exceeded maximum setting value	Axis 1 overload, resulting in excessive torque	Axis-1 over load	1. Reduce the speed of operation. 2. reduce load.
01-03-88	Torque of axis 2 exceeded maximum setting value	Axis 2 overload, resulting in excessive torque	Axis-2 over load	
01-03-89	Torque of axis 3 exceeded maximum setting value	Axis 3 overload, resulting in excessive torque	Axis-3 over load	

Error code	Error	Message	Reason	Solution
01-03-8A	Torque of axis 4 exceeded maximum setting value	Axis 4 overload, resulting in excessive torque	Axis-4 over load	1. Reduce the speed of operation. 2. reduce load.
01-03-8B	Torque of axis 5 exceeded maximum setting value	Axis 5 overload, resulting in excessive torque	Axis-5 over load	
01-03-8C	Torque of axis 6 exceeded maximum setting value	Axis 6 overload, resulting in excessive torque	Axis-6 over load	

7.1.4. Operation Error(01-04-XX)

Error code	Error	Message	Reason	Solution
01-04-10	Read driver 1 encoder is abnormality	Axis 1 absolute encoder position error	Read axis encoder under moving status	Please confirm whether the brake shaft is falling.
01-04-11	Read driver 2 encoder is abnormality	Axis 2 absolute encoder position error		
01-04-12	Read driver 3 encoder is abnormality	Axis 3 absolute encoder position error		
01-04-13	Read driver 4 encoder is abnormality	Axis 4 absolute encoder position error		
01-04-14	Read driver 5 encoder is abnormality	Axis 5 absolute encoder position error		
01-04-15	Read driver 6 encoder is abnormality	Axis 6 absolute encoder position error		
01-04-16	Write data to driver 1 is abnormality	Axis 1 driver parameter write back failed	Driver connection is abnormality	Check driver connection.
01-04-17	Write data to driver 2 is abnormality	Axis 2 driver parameter write back failed		
01-04-18	Write data to driver 3 is abnormality	Axis 3 driver parameter write back failed		
01-04-19	Write data to driver 4 is abnormality	Axis 4 driver parameter write back failed		
01-04-1A	Write data to driver 5 is abnormality	Axis 5 driver parameter write back failed		
01-04-1B	Write data to driver 6 is abnormality	Axis 6 driver parameter write back failed		

Error code	Error	Message	Reason	Solution
01-04-1C	Clear driver 1 encoder is abnormality	Clear Axis 1 driver encoder failed	1.Driver connect is abnormality. 2. The command is forbidden	1.Check driver connected. 2.Check driver status.
01-04-1D	Clear driver 2 encoder is abnormality	Clear Axis 2 driver encoder failed		
01-04-1E	Clear driver 3 encoder is abnormality	Clear Axis 3 driver encoder failed		
01-04-1F	Clear driver 4 encoder is abnormality	Clear Axis 4 driver encoder failed		
01-04-20	Clear driver 5 encoder is abnormality	Clear Axis 5 driver encoder failed		
01-04-21	Clear driver 6 encoder is abnormality	Clear Axis 6 driver encoder failed		
01-04-30	Start position declination is abnormality	Robot position declination	The robot's position is different from when it was last powered off.	Please move to the origin and confirm that the angle is correct. Refer 5.8
01-04-31	A1 declination is abnormality			
01-04-32	A2 declination is abnormality			
01-04-33	A3 declination is abnormality			
01-04-34	A4 declination is abnormality			
01-04-35	A5 declination is abnormality			
01-04-36	A6 declination is abnormality			
01-04-37	E1 axis position declination is abnormality			
01-04-38	E2 axis position declination is abnormality			
01-04-39	E3 axis position declination is abnormality			

Error code	Error	Message	Reason	Solution
01-04-40	RSR(&NUM) no file	RSR file not set	RSR execution file not set	Confirm that the execution file is set.
01-04-41	PNS(&NUM) no file	PNS file not set	PNS execution file not set	
01-04-50	ISR delay stack overflow	ISR delay buffer overflow	ISR delay buffer overflow	1.Please turn off the power and then re-start. 2.If it is still unable to resolve, please contact engineer from the original equipment manufacturer.
01-04-51	Motion command queue overflow	Motion command buffer overflow	Motion command too much, causing buffer overflow	
01-04-52	Jog queue overflow	Jog command buffer overflow	Jog command too much, causing buffer overflow	
01-04-53	Interpolation buffer overflow	Interpolation command buffer overflow	Interpolation command too much, causing buffer overflow	
01-04-60	Modify Time Setting	* Time is modified, will not shutdown.	Time Setting is modified, will not shutdown	
01-04-61	Modify NTP Setting	*NTP is modified, will not shutdown.	NTP Setting is modified, will not shutdown	Inform user NTP is modified, will not shutdown
01-04-70	Infinite rotation is not turned on.	Infinite rotation is not turned on.	User operates infinite rotation in infinite rotation function interface, and executes the CT_A6 command.	After the user turns on the infinite rotation function in the interface, the user executes the CT_A6 command.
01-04-71	Torque record data exceeded upper limit	*Torque record data exceeded upper limit and will not stop.	Offline torque record data exceeded upper limit.	Offline torque record data exceeded upper limit; stop recording and execute again.
01-04-72	Instruction SET_LATCH_PARAM Execution error	Instruction SET_LATCH_PARAM Execution error	1. Set parameter issued during the execution of the dynamic get point function. 2. Designated channels exceeded upper limit	Refer to the software manual and use instructions correctly
01-04-73	Instruction LATCH_START Execution error	Instruction LATCH_START Execution error	1. The dynamic get point function is not set and the start instruction was issued	Refer to the software manual and use instructions

Error code	Error	Message	Reason	Solution
			<ul style="list-style-type: none"> 2. Enable instruction issued during the execution of the dynamic get point function 3. Designated channels exceeded upper limit 	correctly
01-04-74	Instruction GET_LATCH_RESULT execution error	Instruction GET_LATCH_RESULT execution error	<ul style="list-style-type: none"> 1. Get result instruction issued during the execution of the dynamic get point function 2. Get result instruction issued when dynamic get point function was not executed 3. Get result instruction issued when dynamic get point function parameters were not set 4. Designated channels exceeded upper limit 	Refer to the software manual and use instructions correctly
01-04-75	Instruction LATCH_END execution error	Instruction LATCH_END execution error	<ul style="list-style-type: none"> 1. Designated channels exceeded upper limit 	Refer to the software manual and use instructions correctly
01-04-76	Collision detection execution error	This model does not support collision detection	<ul style="list-style-type: none"> 1. Part of the models do not support the collision detection function 	Confirm whether this model supports collision detection

7.1.5. IO & Communication(01-05-XX)

Error code	Error	Message	Reason	Solution
01-05-10	Teach Pendant connection error	TP connection error	1.TP destroy. 2.TP connection port is abnormal.	1.Change TP. 2.Check connect port.
01-05-20	ROBOT IO connection error	Robot IO connection error	Interference	Confirm RIO wire.
01-05-21	ROBOT IO disconnection	Robot IO disconnection	1.Robot IO destroy 2.Robot IO port is abnormal.	1. Change Robot IO. 2. Confirm RIO port.
01-05-30	Network disconnection	Network disconnection	Network is abnormal.	Check network connection.
01-05-31	Network connect failure	Network connect failure	Network server is abnormal.	1.Check network connection server. 2.Check network domain. 3.Check connection IP and PORT setting
01-05-32	Server opened failure	Server opened failure	Server opened failure	Check connection IP and PORT setting
01-05-33	Server closed the connection	Sever connection closed	Sever automatically closed connection	Prevent sever automatically disconnect from client
01-05-34	Network port setting error	Network port setting error	Network port setting error	Check port setting.
01-05-35	Network client disconnect time out	Network client disconnect time out	Network client disconnect time out	Check sever whether interact with client disconnect message
01-05-36	Fieldbus connection failed	Fieldbus slot 1 open failed	The fieldbus PCI card is not installed correctly.	Confirm whether the fieldbus PCI card is installed correctly.
01-05-37		Fieldbus slot 2 open failed		

Error code	Error	Message	Reason	Solution
01-05-38		Fieldbus slot 1 communication error	1. The fieldbus line is not connected properly. 2. The connection parameter setting does not correspond to the PLC. 3. The PLC is not operating normally.	1. Check whether the connection line is normal. 2. Check whether the connection parameters are set correctly. 3. Check whether the PLC device operates normally.
01-05-39		Fieldbus slot 2 communication error		
01-05-40		Fieldbus slot 1 connection timeout	The fieldbus related files were not imported correctly.	
01-05-41		Fieldbus slot 2 connection timeout		
01-05-42	Serial IO disconnected	Serial IO disconnected	Serial IO reconnection failed three consecutive times; hardware configuration issue.	1. Disconnect the power and then restart 2. Check whether the hardware circuits and devices have errors.
01-05-50	Welding machine communication abnormal	Welding machine communication abnormal	DeviceNet communication circuit not connected correctly.	Confirm whether the DeviceNet communication circuit is connected correctly.
01-05-51	Pulse number out of limit	Using over three pulse	Using over three pulse	Check if program use over three pulse at the same time
01-05-52	Serial transmission connection error	Serial transmission connection error	Serial transmission connection error	Please check the serial transmission connection settings
01-05-60	MBC_RTU_OPEN instruction execution error	MBC_RTU_OPEN instruction execution error	MBC_RTU_OPEN instruction execution error	Please confirm the instruction parameter format
01-05-61	MBC_TCP_OPEN instruction execution error	MBC_TCP_OPEN instruction execution error	MBC_TCP_OPEN instruction execution error	Please confirm the instruction parameter format
01-05-62	MBC_READ_DINPUT instruction execution	MBC_READ_DINPUT instruction execution	MBC_READ_DINP UT instruction	Please confirm the instruction

Error code	Error	Message	Reason	Solution
	error	error	execution error	parameter format
01-05-63	MBC_READ_COIL instruction execution error	MBC_READ_COIL instruction execution error	MBC_READ_COIL instruction execution error	Please confirm the instruction parameter format
01-05-64	MBC_WRITE_COIL instruction execution error	MBC_WRITE_COIL instruction execution error	MBC_WRITE_COIL instruction execution error	Please confirm the instruction parameter format
01-05-65	MBC_READ_INPUT instruction execution error	MBC_READ_INPUT instruction execution error	MBC_READ_INPU T instruction execution error	Please confirm the instruction parameter format
01-05-66	MBC_READ_HOLDING instruction execution error	MBC_READ_HOLDING instruction execution error	MBC_READ_HOLD ING instruction execution error	Please confirm the instruction parameter format
01-05-67	MBC_WRITE_HOLDING instruction execution error	MBC_WRITE_HOLDIN G instruction execution error	MBC_WRITE_HOL DING instruction execution error	Please confirm the instruction parameter format
01-05-68	Ethernet card IP setting failure	Ethernet card IP setting failure	<ol style="list-style-type: none"> Using a hub that conflicts with other controller Ethernet card IP. Conflicts with other Ethernet cards of the controller itself. Ethernet card abnormal. 	<ol style="list-style-type: none"> Avoid using 192.168.0.1 or 192.168.0.2 or 192.168.0.3 as the network IP. Make sure that the IP of each controller does not conflict with one another. If you confirmed that the hub does not conflict with the IPs of the Ethernet cards of each controller, but is still unable to set the IP, please contact the engineer of the original manufacturer to check and repair the controller.

7.1.6. Operator Error (01-06-XX)

Error code	Error	Message	Reason	Solution
01-06-10	Movement delay command abnormal.	Parameter cannot be set	Parameter not in settable range.	Confirm parameters.
01-06-11	Acceleration setting command abnormal.			
01-06-12	Point-to-point movement command abnormal.	Point-to-point movement failure.	1. Command format error. 2. Movement command cannot be issued.	1. Confirm the command format. 2. Confirm the status of the movement function.
01-06-13	Arc movement command abnormal.	Arc movement failed.		
01-06-14	Straight line movement command abnormal.	Straight line movement failed.		
01-06-15	Feedrate setting command abnormal.	Parameter cannot be set	Parameter not in settable range.	Confirm parameters.
01-06-16	Path abnormal.	Movement path abnormal.	Movement path exceeded working range.	1. Redesign target position and movement instruction. 2. Confirm whether TOOL and BASE are correct.
01-06-17	Conveyor belt tracking acceleration setting error.	Parameter setting error.	Parameter exceeded range.	Check whether the parameter settings are correct.
01-06-18	Conveyor belt capture acceleration speed setting error.			
01-06-19	Start smooth movement error.			
01-06-1A	Close smooth movement error.			
01-06-20	Counter number abnormal.	Parameter cannot be set	Number not within setting range.	Confirm number.
01-06-21	Timer number abnormal.			
01-06-22	Counter stop number abnormal.			
01-06-23	DI number abnormal.			
01-06-24	DO number abnormal.			
01-06-25	RI number abnormal.			
01-06-26	RO number abnormal.			

Error code	Error	Message	Reason	Solution
01-06-27	VI number abnormal.	Parameter cannot be set	Number not within setting range.	Confirm number.
01-06-28	VO number abnormal.			
01-06-29	SI number abnormal.			
01-06-2A	SO number abnormal.			
01-06-2B	SR number abnormal.			
01-06-2C	SRW number abnormal.		DI setting not available	DI not set
01-06-30	DI can't be setting		RI setting not available	RI not set
01-06-31	RI can't be setting		SI setting not available	SI not set
01-06-32	SI can't be setting		Specific SO setting not available	Specific SO not set
01-06-33	SO can't be setting		SRR setting not available	SRR not set
01-06-34	SRR can't be setting			
01-06-35	SRW value abnormality	SRW value is abnormal.	Parameter error.	Check setting command.
01-06-36	Fieldbus Slot1 abnormality	Fieldbus Slot1 abnormality	1. Parameter setting error. 2. Driver is not installed. 3. Fieldbus connection abnormal.	1. Set the correct parameters. 2. Confirm that the driver installation is completed. 3. Check the hardware wiring.
01-06-37	Fieldbus Slot2 abnormality	Fieldbus Slot2 abnormality		
01-06-40	MI number abnormal	Parameter cannot be set	Number not within setting range.	Confirm number.
01-06-41	MO number abnormal			
01-06-42	MI can't be setting			
01-06-60	No stop movement input signal	No stop movement input signal	Input signal not detected during the moving process of the STOP_MOTION instruction.	Confirm whether the trigger input device is normal.
01-06-70	Weave welding movement abnormal	Weave welding frequency	Weave welding frequency setting	Please set the parameters

Error code	Error	Message	Reason	Solution
		exceeded range	exceeded range.	within the described range
01-06-71		Weave welding dwell time exceeded range	Weave welding dwell time setting exceeded range.	Please set the parameters within the described range
01-06-72		Weave welding mode error	Weave welding mode setting error.	Please set the correct weave welding mode (SINE, CIRCLE, L)
01-06-73		Weave welding acceleration too large	Executed weave welding movement acceleration too large.	Please adjust the weave welding action
01-06-74		Weave welding acceleration too small	Executed weave welding movement acceleration too small.	

7.1.7. External Axis Error (01-07-XX)

Error code	Error	Message	Reason	Solution
01-07-10	E1 axis following error overlimit	E1 axis position over deviation	E1 axis motion command and actual position exceeded deviation	1. Reduce the speed 2. Reduce the load weight 3.Reduce acceleration in percentage
01-07-11	E2 axis following error overlimit	E2 axis position over deviation	E2 axis motion command and actual position exceeded deviation	
01-07-12	E3 axis following error overlimit	E3 axis position over deviation	E3 axis motion command and actual position exceeded deviation	
01-07-13	E1 axis position overlimit of positive	E1 axis exceeded positive rotation limit	Over the positive limit.	E1 axis move towards negative
01-07-14	E1 axis position overlimit of negative	E1 axis exceeded negative rotation limit	Over the negative limit.	E1 axis move towards positive

Error code	Error	Message	Reason	Solution
01-07-15	E2 axis position overlimit of positive	E2 axis exceeded positive rotation limit	Over the positive limit.	E2 axis move towards negative
01-07-16	E2 axis position overlimit of negative	E2 axis exceeded negative rotation limit	Over the negative limit.	E2 axis move towards positive
01-07-17	E3 axis position overlimit of positive	E3 axis exceeded positive rotation limit	Over the positive limit.	E3 axis move towards negative
01-07-18	E3 axis position overlimit of negative	E3 axis exceeded negative rotation limit	Over the negative limit.	E3 axis move towards positive
01-07-19	E1 axis clear encoder error	E1 axis driver clear encoder failed	1.Connection with axis is abnormal. 2. Axis prohibits this command.	1.Check Axis is connected. 2.Check Axis status.
01-07-1A	E2 axis clear encoder error	E2 axis driver clear encoder failed		
01-07-1B	E3 axis clear encoder error	E3 axis driver clear encoder failed		
01-07-1C	Axis-E1 excessive acceleration	Axis-E1 excessive acceleration	Axis-E1 excessive acceleration	Please slow down external axis speed
01-07-1D	Axis-E2 excessive acceleration	Axis-E2 excessive acceleration	Axis-E2 excessive acceleration	Please slow down external axis speed
01-07-1E	Axis-E3 excessive acceleration	Axis-E3 excessive acceleration	Axis-E3 excessive acceleration	Please slow down external axis speed
01-07-1F	E1 torque exceeded maximum setting value	E1 axis overload, resulting in excessive torque	Axis-E1 excessive load	1. Reduce the speed of operation. 2. reduce load.
01-07-20	E2 torque exceeded maximum setting value	E2 axis overload, resulting in excessive torque	Axis-E2 excessive load	
01-07-21	E3 torque exceeded maximum setting value	E3 axis overload, resulting in excessive torque	Axis-E3 excessive load	

7.1.8. Conveyor Tracking Error(01-08-XX)

Error code	Error	Message	Reason	Solution
01-08-10	Camera connection failure	Connection with the vision system is failed.	1.IP setting error. 2.PORT setting error.	1.Check IP setting. 2.Check Port setting.
01-08-11	Camera connection abnormality		Vision system no response.	Confirm the connection with the vision system.
01-08-12	Camera disconnect fail			
01-08-13	Pick command error	Execution instruction is failed.	Point setting error.	If the tracking function of conveyor is triggered by Sensor, the position of point is required to enter into the instruction. Please confirm if the information of point position is entered into the instruction.
01-08-14	Place command error			
01-08-15	Conveyor encoder clear error	Encoder clear failed.	Encoder clear failed.	Check the conveyor wiring is correct
01-08-16	Setting latch source error	Setting latch source error	Setting latch failed	
01-08-17	Start conveyor command error	Start conveyor failed	Conveyor setting failed	Check conveyor setting is correct.
01-08-18	Read encoder count error	Read encoder error	Encoder fault	Check encoder and wiring.
01-08-19	Clear place data error	Data clearing failed	Place clearing failed	Contact an engineer from the original equipment manufacturer.
01-08-1A	CNV_OBJECT can't be setting	Unavailabel to set parameters	Unavailabel to set parameters	Check Robot Language.
01-08-1B	CNV_FULL can't be setting	Unavailabel to set parameters		

Error code	Error	Message	Reason	Solution
01-08-1C	CNV_EMPTY can't be setting	Unavailabel to set parameters		
01-08-1D	Error reading tracking data.	Error reading tracking data.	Cannot get conveyor belt visual tracking value.	Confirm whether the visual device installation and related settings are correct.
01-08-1E	Encoder latch value inconsistent	Encoder latch value inconsistent	Trigger sensor or encoder error.	Check the trigger sensor and the encoder is normal.

7.1.9. User-Defined Error (01-09-XX)

Error code	Error	Message	Reason	Solution
01-09-10	User-defined error 1	User-defined error 1	User-defined error.	User-defined error 1
01-09-11	User-defined error 2	User-defined error 2		User-defined error 2
01-09-12	User-defined error 3	User-defined error 3		User-defined error 3
01-09-13	User-defined error 4	User-defined error 4		User-defined error 4
01-09-14	User-defined error 5	User-defined error 5		User-defined error 5
01-09-15	User-defined error 6	User-defined error 6		User-defined error 6
01-09-16	User-defined error 7	User-defined error 7		User-defined error 7
01-09-17	User-defined error 8	User-defined error 8	User-defined error.	User-defined error 8
01-09-18	User-defined error 9	User-defined error 9		User-defined error 9
01-09-19	User-defined error 10	User-defined error 10		User-defined error 10

7.1.10. Welding Function Error (01-0C-XX)

Error code	Error	Message	Reason	Solution
01-0C-10	Arc on error	Arc on error	The arc has not been detected for a period time after the arc start command was issued.	<ol style="list-style-type: none"> 1. Check whether the communication is abnormal. 2. Check whether the wire feeder feeds the wire correctly. 3. Restart the welder
01-0C-11	Welding tool not ready	Welding tool not ready	Welding related equipment is not connected properly, such as wire feeder, gas cylinder, torch, etc.	Confirm that the wire feeder, gas cylinder, torch and other welding devices are properly connected.
01-0C-12	Wire sticked	Wire sticked	The welding torch electrode and the workpiece are short-circuited.	Cut off the welding wire and restart welding.
01-0C-13	No gas	No gas	Gas control device is abnormal.	<ol style="list-style-type: none"> 1. Check whether the gas cylinder is connected properly. 2. Replace the gas cylinder and related control devices.
01-0C-14	No water	No water	No cooling water device	Check whether the cooling water device is properly connected.
01-0C-15	Piece touched	Piece touched	The welding torch and the workpiece touched before welding.	The adjustment procedure moves the welding torch to a position where it does not touch the workpiece before performing welding.

Error code	Error	Message	Reason	Solution
01-0C-16	Safety limit no active	Safety limit no active	Safety limit switch is not turned on.	Clear the alarm after turning on the safety limit switch.
01-0C-17	Antishock alarm	Antishock alarm	The torch is collided.	Check the welding torch mechanism and confirm the TCP position to clear the alarm.
01-0C-20	Arc detect	Arc detect	An arc is detected during the arc extinguishing phase, and the torch may leak electricity.	Check whether the torch connection configuration is normal.
01-0C-21	Touch sensed	Touch sensed	The welding torch and the workpiece touched before welding.	The adjustment procedure moves the welding torch to a position where it does not touch the workpiece before performing welding.
01-0C-22	Gas fault	Gas fault	Gas control device is abnormal.	<ol style="list-style-type: none"> 1. Check whether the gas cylinder is connected properly. 2. Replace the gas cylinder and related control devices.
01-0C-23	Wire fault	Wire fault	Wire feeder is abnormal.	<ol style="list-style-type: none"> 1. Manually run the wire feed mechanism to check whether the wire feed is normal. 2. Change the wire feeder control device.
01-0C-24	Water fault	Water fault	Welding water device does not function correctly	Please check the welding water device

Error code	Error	Message	Reason	Solution
01-0C-25	Inverter fault	Inverter fault	Abnormality in the power source of the welder	Restart the welder after checking the power supply.
01-0C-26	Wire stick	Wire stick	The welding torch electrode and the workpiece are short-circuited.	Cut off the welding wire and restart welding.
01-0C-27	DN fault	DN fault	An error occurred in the DeviceNet I/O of the welder.	Confirm the DeviceNet connection and restart the welder.
01-0C-29	Process select no ack	Process select no ack	The welding parameters have been set over limit.	Confirm whether the welding parameters are set in the correct range.
01-0C-30	Weld command timeout	Weld command timeout	Welding command is not waiting for the correct return signal	Confirm the DeviceNet connection and restart the welder.
01-0C-31	Torch collided	Torch collided	Issued at the input signal of the collision of the welding torch connection	Confirm the status of the welding torch and perform TCP reset.
01-0C-32	Weld down slope error	Weld down slope error	Not waiting for the correct return signal	Confirm the DeviceNet connection and restart the welder.

7.1.11. Database Error (01-0D-XX)

Error code	Error	Message	Reason	Solution
01-0D-10	Database version error	Database version error	Database version error	Please update the version
01-0D-11	Database rebuild error	Database rebuild error	Database cannot be updated or rebuilt	1. Please update the software version Contact engineer of original manufacturer

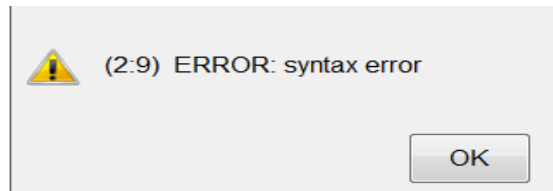
Error code	Error	Message	Reason	Solution
01-0D-12	Database save failed	Database save failed	Database content error, cannot save	1. Please update the software version Contact engineer of original manufacturer

7.1.12. Reserved Point Error (01-0E-XX)

Error code	Error	Message	Reason	Solution
01-0E-10	SIO reserved point conflict error	SIO reserved point conflict error	SIO reserved point conflict error	Please reset the ISO reserved point

7.1.13. Windows Information

When the program is executed, if the program syntax is wrong, the following window will appear



The above figure as an example, the message is divided into two main parts:

1. Error location: (2: 9), on behalf of 9th words on line 2 is wrong.
2. Error message: syntax error.

Error code	Error	Message	Reason	Solution
01-02-10	syntax error	Syntax error	Command spelling error. Wrong space.	Check spelling and spaces.
	ID not exist	Variable not exist	Variable not declared.	Declare variable before use.
	Unknown character	Character cannot be recognized	Use special symbols.	Change variable name.
	... is not declared	Variable not declared	Variable not declared.	Declare variable before use.
	Invalid value	Invalid value	Value out of range	Modify value according to instruction
	Index of ... is out of range	Index is out of range	Array index out of range	Modify array index
	Type should be ...	Type error	Type error.	Change to the correct

Error code	Error	Message	Reason	Solution
				type.
	Fail in handling STRUC member expression	Structure member variable expression error	Struct member not declared.	Check the declaration of structure variables.
	string assign error	Can not assign string	1.Assign data with wrong data type to string.	1.Please check data type.
	string copy error	String copy error	1.copy data which is not string.	1.Please check source of string.
01-02-10	can not plus string with other data type	Can not cascade string	1.Casecad string with other data which is not string.	1.Cast data to string type.
	casting error: wrong data type	Input data type error	1.Input parameter with wrong data type.	1. Please check data type.
	can not compare string	Can not compare string	1.Compare data which is not string. 2.Compare two constant value.	1. Please check data type.
	can not find string	Can not search string	1.Input parameter with wrong data type.	1. Please check data type.
	can not get length	Can not get length of string	1.Input parameter with wrong data type.	1. Please check data type.
	can not copy string	Can not copy string	1.Copy string to constant.	1.Pleas check copy destination is variable
01-02-10	STRCPY pos and length should be int	STRCPY pos and length should be int	1.Input parameter with wrong data type.	1.Please check pos and length is int
	string out of size	string out of size	1. string out of size	1.Please check length of string is under 255.
	string copy out of boundary	string copy out of boundary	1 string copy out of boundary	1.Please Pos and Length is correct.

Error code	Error	Message	Reason	Solution
	CREAD string out of size	CREAD string out of size	1.string oversize	1.Please check length of receive string is under 255
	CWRITE string out of size	CWRITE string out of size	1.string oversize	1. .Please check length of transfer string is under 1000
01-02-10	pulse sec out of range	pulse sec out of range	pulse sec out of range	check pulse time
	time out sec out of range	time out sec out of range	time out sec out of range	check time out sec

7.2. HIWIN Robot Controller(02-XX-XX)

7.2.1. Safety Input(02-01-XX)

Error code	Error	Message	Reason	Solution
02-01-10	Emergency input	Emergency stop signal disconnect	Emergency stop trigger.	Release the emergency stop and clear the error.
02-01-11	Enable switch down	Enable switch down	Enable switch is pressed to the third paragraph.	Release enable switch.

7.2.2. Hardware Error(02-02-XX)

Error code	Error	Message	Reason	Solution
02-02-11	No motor brake signal	Do not receive motor brake signal	<ol style="list-style-type: none"> 1. Hardware abnormalities. 2. The emergency stop status is excluded within 500 milliseconds. 	<ol style="list-style-type: none"> 1. Contact with the engineer from the original equipment manufacturer. 2. Press emergency stop again, over 500 millisecond.

7.3. Axis Amplifier(03-XX-XX)

7.3.1. Function Name and Number Description

Function	No.	Description
Axis number(m)	0m	m: axis number. ex 03-01-21 : axis 1 alarm, 03-02-21 : axis 2 alarm, and so on.
Ext axis number (n)	En	n: external axis number ex : 03-E1-21-> external axis 1 alarm, 03-E2-21->external axis 2 alarm, and so on.

7.3.2. Driver Alert Number

Error code	Error	Message	Reason	Solution
03-0m(En)-21	overcurrent	Current exceeds the specified value	1. Driver is abnormal. 2. Motor U, V, W is short circuit. 3. Motor is broken.	1. Check the servomotor main circuit cable connection. 2. Replace the driver. 3. Replace the motor.
03-0m(En)-24	STO Active	Safe input protection.	Security input signal action.	Confirm the security input signal status.
03-0m(En)-25	STO	Safety input protection.	Safety input signal.	Check the safety input signal status.
03-0m(En)-26	Safe torque off error	Safe torque off error	Safety input signal.	Check the safety input signal status.
03-0m(En)-27	Single STO	Safety input protection.	Safety input signal.	Check the safety input signal status.
03-0m(En)-30	Current control error	Current control error	1. Servo malfunction. 2. Motor malfunction.	1. Check the parameter id_pwr_board, write 100W: manufacture_data =2 I2CEE.Sate=11 400W: manufacture_data =3 I2CEE.Sate=11

Error code	Error	Message	Reason	Solution
				2. Change the motor.
03-0m(En)-32	HFLT inconsistent	HFLT inconsistent	1. Servo malfunction. 2. Electrical interference.	1. Change the DA module of that axis. 2. Keep away and remove interference sources, or implement EMC countermeasures of magnetic rings.
03-0m(En)-34	DC bus voltage abnormal	DC bus voltage abnormal	1. Servo malfunction 2. Insufficient capacitance for the driver.	1. Change the DA/DP module. 2. Change the HCV1 module.
03-0m(En)-35	ECAT interface cannot be detected	ECAT interface cannot be detected	1. Firmware version error. 2. EtherCAT communication interface of driver malfunctioned.	1. Confirm that the firmware version is 8.197 or above. 2. Check whether the connector pins of the DA module are short-circuited and replace the DA module.
03-0m(En)-36	CiA-402 reset error	CiA-402 reset error	This process cannot be completed.	Reconnect the power and execute (reload the DA module parameters).
03-0m(En)-37	Fan error	Fan error	1. Fan cable disconnected, fan stuck or too dirty. 2. Continuous alarm from the DA module of a specific axis.	1. Change the fan. 2. Change the HCV1 backboard or DA module.

Error code	Error	Message	Reason	Solution
03-0m(En)-38	Absolute encoder error	Absolute encoder error	<ol style="list-style-type: none"> 1. Power of encoder in robot abnormal. 2. Battery has no voltage or low voltage. 3. Parallel battery failure or reversed polarity 4. Battery wiring installation abnormal. 5. Did the upper position send the correct clear instruction? 	<ol style="list-style-type: none"> 1. Change the power supply. 2. Change the battery. 3. Confirm the battery life or change the battery. 4. Adjust the wiring. 5. Send the correct clear instruction (enter the value 7 for SDO object number: 0x2060).
03-0m(En)-41	overload	Torque is too large.	<ol style="list-style-type: none"> 1.The effective torque exceeds the rated torque. 2. The motor's hold brake is not released. 3. Power supply wiring is incorrect 	<ol style="list-style-type: none"> 1. Change the motion plan, or reduce load. 2. Check that the wiring and the driver voltage are correct.
03-0m(En)-43	regenerative resistor overload	Regenerative load rate is too large.	<ol style="list-style-type: none"> 1. Insufficient external regeneration resistor capacity. 2. Amplifier failed. 	<ol style="list-style-type: none"> 1. Replace the external regeneration resistor capacity 2. Replace amplifier
03-0m(En)-45	overspeed	Exceeded average rotational speed	The servomotor speed is above the maximum rotational speed	Change operating conditions.

Error code	Error	Message	Reason	Solution
03-0m(En)-51	amplifier thermal abnormality	The amplifier temperature is too high.	<ol style="list-style-type: none"> 1. Regenerative power is too large. 2. The surrounding air temperature is too high. 3. Built-in Fan in amplifier Stopped. 	<ol style="list-style-type: none"> 1. Change the amplifier installation conditions. 2. Check whether the cooling fan is running.
03-0m(En)-52	Anti-surge resistor overheat	Anti-Surge resistor overheated.	<ol style="list-style-type: none"> 1. Power switch frequency is too high. 2. Ambient temperature is too high. 	<ol style="list-style-type: none"> 1. Reduce the power switch frequency. 2-1. Check the cooling fan is running. 2-2. Change the amplifier installation conditions.
03-0m(En)-53	dynamic brake resistor overheat	Dynamic brake resistor overheated.	Dynamic brake action frequency is too high.	Used within the allowable operating frequency range
03-0m(En)-58	Drive temperature overheat	Drive temperature overheat	<ol style="list-style-type: none"> 1. Drive environment is overheated. 2. Motor overload. 3. Motor speed too fast. 	<ol style="list-style-type: none"> 1. Confirm drive cooling mode is normal. 2. Confirm electrical control box is in a ventilated condition 3. Reduce the load weight. 4. Reduce arm speed.
03-0m(En)-61	overvoltage	Main circuit DC voltage is excessively high.	<ol style="list-style-type: none"> 1. The power supply exceeded the allowable range. 2. The moment of inertia ratio exceeded the allowable value. 	<ol style="list-style-type: none"> 1. Measure the power supply voltage 2. Confirm that the moment of inertia ratio is within the allowable range.

Error code	Error	Message	Reason	Solution
03-0m(En)-62	undervoltage	Main circuit DC voltage is excessively low.	<ol style="list-style-type: none"> 1. Input supply voltage is below the allowable range. 2. The power supply is unstable, or was influenced by a lightning surge. 	Set AC power supply voltage within the specified range.
03-0m(En)-63	Phase error	Main power supply phase error.	Any one of the three phases of the main circuit is abnormal.	Please check the circuit.
03-0m(En)-71	control power source voltage shortage	The voltage of the control power is too low.	<ol style="list-style-type: none"> 1. Input supply voltage is below the allowable range. 2. The power supply is unstable, or was influenced by a lightning surge. 	<ol style="list-style-type: none"> 1. Set AC power supply voltage within the specified range. 2. .Confirm the power and ensure that the power has no abnormal phenomenon such as instant outage and decreased voltage.
03-0m(En)-72	control circuit voltage shortage	The control circuit voltage is insufficient	<ol style="list-style-type: none"> 1. Contact fault of connector or incorrect wiring for encoder cable. 2.The amplifier internal circuit is bad. 	<ol style="list-style-type: none"> 1. Re-insert the connector and confirm that the encoder is correctly wired. 2.Replace amplifier.
03-0m(En)-81	encoder disconnect	Encoder signal is disconnected.	<ol style="list-style-type: none"> 1. Wrong connection. 2. Connector off. 3. Poor connection 	<ol style="list-style-type: none"> 1. Check the encoder cable. 2. Check the power supply voltage on the motor side.

Error code	Error	Message	Reason	Solution
03-0m(En)-84	encoder communication abnormality	Encoder Communications Error	<ol style="list-style-type: none"> 1. Malfunction caused by noise interference. 2. Contact fault of connector or incorrect wiring for encoder cable. 	<ol style="list-style-type: none"> 1. Correct the wiring around the encoder by separating the encoder cable from the servomotor main circuit cable or by checking the grounding and other wiring. 2. Check the encoder cable.
03-0m(En)-85	encoder initial error(5V abnormality)	Encoder initial error	<ol style="list-style-type: none"> 1. Wrong connection. 2. Connector off. 3. Poor connection. 	<ol style="list-style-type: none"> 1. Check the encoder cable. 2. Check the power supply voltage on the motor side.
03-0m(En)-87	encoder CS abnormality	Encoder CS signal disconnect.	<ol style="list-style-type: none"> 1. Wrong connection. 2. Connector off. 3. Poor connection. 	<ol style="list-style-type: none"> 1. Check the encoder cable. 2. Check the power supply voltage on the motor side.
03-0m(En)-A1	encoder multi-turn data error (battery abnormality)	Encoder Backup Error	<ol style="list-style-type: none"> 1. The encoder cable disconnected, and connected again. 2. The battery voltage is low. 	<ol style="list-style-type: none"> 1. Check the encoder connector battery or the connect or contact status. 2. Measure the battery voltage.
03-0m(En)-A3	encoder overspeed	Servomotor speed is too high.	Motor acceleration exceeds allowable acceleration range.	Modify motion condition, increase acceleration/deceleration time
03-0m(En)-A5	encoder single turn error	Detected encoder single turn error	1. Excessive noise to the encoder cable.	1. Check noise in the cable between the

Error code	Error	Message	Reason	Solution
03-0m(En)-A6	encoder multi-turn error	Detected encoder single turn error	2. The amplifier internal circuit is bad.	SERVOPACK and the host controller. 2. Re-insert the connector and confirm that the encoder is correctly wired.
03-0m(En)-A9	encoder overheat	The amplifier temperature is too high.	1. The surrounding air temperature is too high. 2. Motor is overheated.	Change motor installation method.
03-0m(En)-AB	encoder error	An encoder error was detected.	1. Excessive noise to the encoder cable. 2. The amplifier internal circuit is bad.	1. Check noise in the cable between 2. If the restart cannot be solved, please replace the motor.
03-0m(En)-C1	speed overlimit	The speed of the motor exceeds 120% of the maximum speed.	Overshoot too big.	1. Adjust the servo parameters. 2. Slow command acceleration and deceleration mode.
03-0m(En)-D1	position error too big	Position deviation exceeded the set value	1. Load inertia is too large. 2. The brake is not released. 3. The position command frequency is too high.	1. Change the load conditions, or replace a larger capacity motor. 2. Check the encoder cable. 3. Change the controller's position command.
03-En-D2	External axis reached positive limit of encoder.	Axis En reached positive limit.	Movement to the positive rotation limit of the external axis triggered the limit signal.	External axis moving towards the negative direction of the encoder.
03-En-D3	External axis reached the	Axis En reached the negative limit.	Movement to the negative rotation	External axis moving towards the positive

Error code	Error	Message	Reason	Solution
	negative limit of the encoder.		limit of the external axis triggered the limit signal.	direction of the encoder.
03-0m(En)-E1	EEPROM abnormality	EEPROM abnormality	The driver internal circuit is bad.	Replace the driver.
03-0m(En)-E2	EEPROM check is abnormality	EEPROM check is abnormality	The CPU cannot read the correct data from the driver's built-in EEPROM.	
03-0m(En)-EF	Motor not matching	The amplifier does not match the motor.	Use the wrong driver or motor.	Replace the correct driver or motor.
03-0m(En)-F3	amplifier error	amplifier error	amplifier error	According to the driver brand, compare the driver Error code.
03-0m(En)-F4	software thermal reach limit	Motor reaches temperature limit.	Motor temperature is too high.	Reduce speed or reduce load.
03-0m(En)-F5	motor disconnect	Motor cannot connect.	Motor disconnect.	Check the motor cable.
03-0m(En)-F6	amplifier phase initial error	amplifier phase initial error	Phase initialization failed.	1. Replace motor or driver. 2. Check the cable.
03-0m(En)-F7	Hall sensor error	Hall sensor error.	Hall sensor error.	
03-0m(En)-F8	Hall phase error	Hall phase error.	Hall phase check error.	
03-0m(En)-F9	overload warning	* Overload warning. Robot will not stop	The effective torque exceeds the set torque.	Relax the conditions of use.
03-0m(En)-FA	amplifier overheating warning	* Amplifier overheated warning. Robot will not stop.	The temperature around the amplifier is greater than the preset temperature range.	Reduce the ambient temperature.
03-0m(En)-FB	regenerated overload warning	* Regenerative overload warning. Robot will not stop	Regenerated resistance overload.	Relax the conditions of use.

Error code	Error	Message	Reason	Solution
03-0m(En)-FC	detecting power failure	* Detecting power failure. Robot will not stop.	Detected control power input voltage is insufficient.	<ol style="list-style-type: none"> 1. Check if the input power supply has momentary or low voltage status. 2. Maybe the internal circuit of the amplifier is abnormal. If this alarm occurs for a long time, replace an amplifier.
03-0m(En)-FD	main circuit is abnormal	* Main voltage is abnormal. Robot will not stop.	Main power voltage exceeds DC 105V.	<ol style="list-style-type: none"> 1. Check input mains voltage is within specifications. (Three-phase : AC200~230V+10 , -15% , 50/60Hz±3Hz) 2. The inertia of the load may be too large, reducing the load inertia. 3. For regenerative resistors, the wiring may not be correct or the impedance does not match the cause of the problem. Check that the impedance of the wiring or external resistor meets the specifications in this manual.
03-0m(En)-FE	battery insufficient	* The battery voltage is low.	Measure the battery voltage.	Replace the battery.
03-0m(En)-FF	battery empty	The battery voltage is empty.	Battery is empty	User should replace with a new battery immediately.

7.3.3. DAC – Y Driver Alarm Code(Y-XXX)

Error code	Error	Message	Reason	Solution
Y-020	Parameters and check abnormal.	Servo unit is abnormal	Data of internal parameter of SERVOPACK is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-021	Parameters format abnormal.	Servo unit is abnormal	Data format of internal parameter of SERVOPACK is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-022	System and calibration abnormal.	Servo unit is abnormal	Data of internal parameter of SERVOPACK is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-030	Main loop detected abnormal.	Servo unit is abnormal	Servo unit is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-040	Parameters setting abnormal.	Servo unit is abnormal.	Data of internal parameter of SERVOPACK is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-041	Division pulse output setting abnormal.	Parameter setting abnormal.	Parameter setting is abnormal.	1. Please turn off the power and restart.

Error code	Error	Message	Reason	Solution
				2. Please contact the engineer from manufacturer.
Y-042	Parameters combination abnormal.	Parameter setting abnormal.	Parameter setting is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-044	Semi/closed loop/ Full close loop parameters.	Parameter setting abnormal.	Parameter setting is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-050	Combination error.	Servo unit is abnormal.	Servo unit is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-051	Product is not supported.	Servo unit is abnormal.	Servo unit is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-0B0	Servo ON command is invalid.	Servo unit is abnormal.	Servo unit is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.

Error code	Error	Message	Reason	Solution
Y-100	Overcurrent detection	Servo unit is abnormal.	Servo unit is abnormal.	<ol style="list-style-type: none"> 1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-300	Abnormal regeneration	Servo unit is abnormal.	Servo unit is abnormal.	<ol style="list-style-type: none"> 1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-320	Regenerative overload	Regeneration overload alarm.	Regenerative resistor capacity is insufficient or it is in a continuous regeneration state.	<ol style="list-style-type: none"> 1. Please turn off the power and restart. 2. Review the operating conditions. 3. Please contact the engineer from manufacturer..
Y-330	Main circuit power wiring error.	Servo unit is abnormal.	Servo unit is abnormal.	<ol style="list-style-type: none"> 1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-400	Overvoltage	Servo unit is abnormal.	Servo unit is abnormal.	<ol style="list-style-type: none"> 1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.

Error code	Error	Message	Reason	Solution
Y-410	Insufficient voltage	Servo unit is abnormal.	Servo unit is abnormal.	<ol style="list-style-type: none"> 1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-450	Main circuit capacitor overvoltage.	Servo unit is abnormal.	Servo unit is abnormal.	<ol style="list-style-type: none"> 1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-510	Overspeed	Motor speed is above maximum speed.	Command input value is too high or the servo unit is abnormal.	<ol style="list-style-type: none"> 1. Adjust the operating conditions. 2. Please turn off the power and restart. 3. Please contact the engineer from manufacturer.
Y-511	Division pulse output overspeed.	Servo unit is abnormal.	Servo unit is abnormal.	<ol style="list-style-type: none"> 1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-520	Vibration alarm	Abnormal vibration of motor speed is detected.	Command input value is too high or the servo unit is abnormal.	<ol style="list-style-type: none"> 1. Adjust the operating conditions. 2. Please turn off the power and restart. 3. Please contact the engineer from manufacturer.

Error code	Error	Message	Reason	Solution
Y-521	Advanced auto-tune alert.	The vibration was detected in the adjustment-free function.	When the adjustment function is executed, the motor vibrates greatly.	<ol style="list-style-type: none"> 1. Adjust the operating conditions. 2. Please turn off the power and restart. 3. Please contact the engineer from manufacturer.
Y-710 (moment) Y-720 (continuous)	Overload	Exceeded the maximum payload.	The motor runs beyond the overload protection feature.	<ol style="list-style-type: none"> 1. Adjust the operating conditions. 2. Please turn off the power and restart. 3. Please contact the engineer from manufacturer.
Y-730 Y-731	DB overload.	The power consumption of the detected DB is too large.	The motor is driven by an external force or the servo unit is abnormal.	<ol style="list-style-type: none"> 1. Do not drive the motor by external force. 2. Please turn off the power and restart. 3. Please contact the engineer from manufacturer.
Y-740	The surge current limit resistor is overloaded.	Main circuit is energized too high.	Servo unit is abnormal.	<ol style="list-style-type: none"> 1. Adjust the operating conditions. 2. Please turn off the power and restart. 3. Please contact the engineer from manufacturer.

Error code	Error	Message	Reason	Solution
Y-7A0	Heat sink is overheated.	The heat sink temperature exceeds 100 °C.	The ambient temperature is too high or the servo unit is abnormal.	<ol style="list-style-type: none"> 1. Adjust the operating conditions. 2. Please turn off the power and restart. 3. Please contact the engineer from manufacturer.
Y-7AB	Built-in fan stopped.	The internal fan of the SERVOPACK stopped.	There is a foreign object entering, or the servo unit is abnormal.	<ol style="list-style-type: none"> 1. Remove foreign objects. 2. Please turn off the power and restart. 3. Please contact the engineer from manufacturer.
Y-810	Encoder backup alert.	The encoder data is abnormal.	The power is turned on for the first time, or the servo unit is abnormal.	<ol style="list-style-type: none"> 1. Make the settings of the encoder. 2. Please turn off the power and restart. 3. Please contact the engineer from manufacturer.
Y-820	Encoder and number alarm.	Encoder and number verification errors.	Servo unit is abnormal.	<ol style="list-style-type: none"> 1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-830	Encoder battery alarm.	The battery voltage of the absolute encoder is	The battery voltage is	<ol style="list-style-type: none"> 1. Replace the battery.

Error code	Error	Message	Reason	Solution
		lower than the specified value.	insufficient or the servo unit is abnormal.	2. Please turn off the power and restart. 3. Please contact the engineer from manufacturer.
Y-840	Encoder data alert.	The encoder is malfunctioning.	The servo unit is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-850	Encoder overspeed.	When the control power is turned on, the encoder overspeed is detected.	The servo unit is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-860	The encoder is overheated.	The encoder exceeds the upper temperature limit.	The ambient temperature is too high or the servo unit is abnormal.	1. Adjust the ambient temperature to below 40 °C. 2. Please turn off the power and restart. 3. Please contact the engineer from manufacturer.
Y-B10	The speed command A/D is abnormal.	When the servo is turned ON, the speed command input is incorrectly operated.	The servo unit is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.

Error code	Error	Message	Reason	Solution
Y-B11	The speed command A/D conversion data is abnormal.	The speed command input is incorrectly operated.	The servo unit is abnormal.	<ol style="list-style-type: none"> 1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-B20	The torque command A/D is abnormal.	When the servo is turned ON, the torque command input is incorrectly operated.	The servo unit is abnormal.	<ol style="list-style-type: none"> 1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-B31	Current detection error 1	U phase current detection loop is abnormal.	The servo unit is abnormal.	<ol style="list-style-type: none"> 1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-B32	Current detection error 2	V phase current detection loop is abnormal.	The servo unit is abnormal.	<ol style="list-style-type: none"> 1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-B33	Current detection error 3	The current detection loop is abnormal.	The servo unit is abnormal.	<ol style="list-style-type: none"> 1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.

Error code	Error	Message	Reason	Solution
Y-BF0 Y-BF1 Y-BF2 Y-BF3 Y-BF4	System alarm 0~4	The servo unit is abnormal.	The servo unit is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-C10	Detected out of control.	When the servo is turned ON, the detected motor is out of control.	The motor wiring is incorrect or the servo unit is abnormal.	1. Confirm that there is no problem with the motor wiring. 2. Please turn off the power and restart. 3. Please contact the engineer from manufacturer.
Y-C80	The clearing of encoder is abnormal.	The upper limit of the number of revolutions setting is abnormally.	The servo unit is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-C90	The encoder communication is abnormal.	The encoder is malfunctioning.	The servo unit is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-C91	The encoder communication position data acceleration is abnormal.	The encoder is malfunctioning.	The servo unit is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-C92	The encoder communication timer is abnormal.	The encoder is malfunctioning.	The servo unit is abnormal.	1. Please turn off the power and restart.

Error code	Error	Message	Reason	Solution
				2. Please contact the engineer from manufacturer.
Y-CA0	The encoder parameters are abnormal.	The encoder is malfunctioning.	The servo unit is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-CB0	Encoder calibration returned abnormal.	The encoder is malfunctioning.	The servo unit is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-CC0	The upper limit of the number of revolutions is inconsistent.	The encoder is malfunctioning.	The servo unit is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-D00	The position deviation is too large.	In the state of servo ON, the position deviation exceeds the upper limit.	The position command is too fast, or the servo unit is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-D01	The position deviation is too large when the servo is turned ON.	When the servo is OFF and the position deviation is too large, the servo is directly turned ON.	The servo unit is abnormal.	1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.

Error code	Error	Message	Reason	Solution
Y-D02	The positional deviation caused by the speed limit at servo ON is too large.	In the accumulated position deviation state, the servo is ON, and the position command is input in this state, and the position deviation excessive alarm value is exceeded.	The servo unit is abnormal.	<ol style="list-style-type: none"> 1. Please turn off the power and restart. 2. Please contact the engineer from manufacturer.
Y-F10	The power cable is out of phase.	When the main circuit power is ON, the low voltage state of one of the R, S, and T phases lasts for more than 1 second.	The three-phase power supply wiring is defective, or the servo unit is abnormal.	<ol style="list-style-type: none"> 1. Confirm that there is no problem with the power wiring. 2. Please turn off the power and restart. 3. Please contact the engineer from manufacturer.

7.3.4. DAC – S Driver Alarm Code(S-XXXX)

Error code	Error	Message	Reason	Solution
S-3110	Power supply overvoltage	Power supply overvoltage	Main circuit AC voltage is out of range.	Check if the power supply voltage is within the specified range or install an external regenerative resistor.
S-3130	Main power phase error	Main power phase error	One of the phase is disconnected from three-phase main power supply.	Check wiring or replace the drive.
S-3211	Overvoltage	Overvoltage	Mains DC overvoltage.	Replace the drive. Reduce the power supply voltage to the specified range. Reduce the load rate.
S-3212	Regenerative resistor overload.	Regenerative resistor overload.	Regenerative resistance load is too large.	Confirm that the operating conditions are correct.
S-3220	Main circuit low voltage.	Main circuit low voltage.	Main circuit DC low voltage.	Check if the power supply voltage is within the specified range. Replace the drive.
S-4110	Drive temperature error.	Drive temperature is abnormal.	Ambient temperature is too high or the drive is damaged.	Confirm that the drive ambient temperature does not exceed 55°C. Replace the drive.
S-4210	Anti-surge resistor overheating.	Anti-surge resistor overheating.	Drive failure or ambient temperature is too high.	Replace the drive. Confirm that the ambient temperature does not exceed 55°C.

Error code	Error	Message	Reason	Solution
S-5113	Control power supply low voltage 2.	Control power supply low voltage 2.	Undervoltage $\pm 5V$ control switching power supply.	Replace the drive or Confirm external circuit..
S-5114	Control power supply low voltage.	Control power supply low voltage.	Control power supply voltage is too low.	Replace the drive Check if the power supply voltage is within the specified range.
S-5115	Control power supply low voltage 1.	Control power supply low voltage 1.	Undervoltage $\pm 12V$ control switching power supply.	Replace the drive or Confirm external circuit.
S-5210	Abnormal current detection.	Abnormal current.	Drive damage or motor damage.	Replace the motor or drive.
S-5220	System error.	System error.	Setting mismatch.	Replace the drive.
S-5400	Main power supply equipment error.	Abnormal power supply.	Abnormal power supply, over-current or overheating of the servo module.	Confirm wiring, replace servo motor or drive. Confirm that the environment does not exceed 55°C.
S-5510	Memory error.	Memory error.	CPU access error of CPU built-in memory.	Replace the drive.
S-5530	EEPROM error	EEPROM error	Drive built-in EEPROM abnormal.	Replace the drive.
S-6010	Initialization thread timeout.	Initialization thread timeout.	The initialization thread was not completed within the initialization time.	Replace the drive. Confirm that the drive is properly grounded.

Error code	Error	Message	Reason	Solution
S-6310	EEPROM calibration code error.	EEPROM calibration code error.	CPU access error of CPU built-in EEPROM.	Replace the drive.
S-6320	System parameter error.	System parameter error.	System parameter abnormal.	Replace the drive.
S-7120	Motor temperature error.	Abnormal motor temperature.	Motor damage, high ambient temperature, short circuit.	Replace the servo motor. Confirm that the ambient temperature does not exceed 55°C. Confirmation cable.
S-7122	Speed feedback error.	Speed feedback error.	Motor power cable disconnection.	Confirm wiring. Replace the drive or motor.
S-7300	Encoder initialization failed.	Encoder initialization failed.	Cable break.	Confirm wiring. Check if the encoder power supply is higher than 4.75V Replace the motor or drive.
S-7305	Encoder connector 1 is broken.	Encoder connector 1 is broken.	Power supply cable disconnection.	Confirm wiring. Check if the encoder power supply is higher than 4.75V or replace the motor
S-7510	Communication error.	Communication error.	Abnormal communication.	Check if the communication format is correct.
S-7520	Link lost.	Communication disconnect.	Communication cable is damaged or not connected.	Confirm that the communication cable is connected or normal.
S-8311	Overload	Overload	Motor load is too large.	Reduce load or slow down.
S-8312	STO safe torque off abnormal.	STO safe torque off abnormal.	STO input is abnormal.	Confirm stop.
S-8400	Average continuous	Average continuous	Motor speed overspeed.	Reduce operating speed.

Error code	Error	Message	Reason	Solution
	speed overspeed.	speed overspeed.		
S-8500	Position command error.	Position command error.	Position command is out of setting range.	Reduce the amount of input movement command.
S-8611	Position deviation is too large.	Position deviation is too large.	Position deviation exceeds the set value.	Confirm wiring. Confirm the power supply voltage. Replace the drive or motor.
S-8700	Task thread error.	Task thread error.	CPU interrupt error.	Replace the drive.

7.3.5. DAC – H Driver Alarm Code(H -XXX)

Error code	Error	Message	Reason	Solution
H-2310	Over-current	Over-current	<ol style="list-style-type: none"> 1. Driver internal short-circuit or ground short-circuit. 2. Wiring of the regenerative resistor is incorrect or resistance value too low. 3. Motor internal short-circuit or ground short-circuit or motor failure. 4. Motor power wiring error or ground short-circuit. 5. Running deceleration time too short and regenerative energy too large. 6. Dynamic brake not released. 	<ol style="list-style-type: none"> 1. Please replace the driver module. 2. Please replace the battery module. 3. Please replace the motor. 4. Please fix the wiring or replace cables. 5. Please reduce the load or increase deceleration time. 6. Reload parameters and restore settings.
H-3110	Over-voltage	Over-voltage	<ol style="list-style-type: none"> 1. AC power voltage exceeded specified range of the driver. 2. Regenerative energy too high, exceeded the regenerative processing capability of the external regenerative resistor. 3. Movement exceeded allowable inertia ratio. 4. Servo malfunction. 	<ol style="list-style-type: none"> 1. Please adjust the voltage of the AC power to within the specified range of the driver. 2. Select a suitable external regenerative resistor. 3. Please reduce the deceleration speed or load. 4. Connect the power of the control circuit again while the power of the main circuit is not connected; if this alarm still occurs, it may be due to driver failure. Please replace the driver.
H-8611	Position error too large	Position error too large	<ol style="list-style-type: none"> 1. Incorrect wiring of the U, V and W phases of the servo motor. 2. Command acceleration too high. 3. Position deviation too large and alarm value too low. 4. Limit function set for the driver I/O or malfunctioned. 5. Frequent full load operation of the motor that caused the encoder to overheat abnormally. 	<ol style="list-style-type: none"> 1. Confirm whether the power cable of the motor or the encoder cable are connected correctly. 2. Set the position command acceleration and deceleration time constant. 3. Adjust the position deviation too large alarm value. 4. Reload the parameters or connect the power again; if this alarm still occurs, it may be due to driver failure; please replace the driver. 5. Reduce the load or replace a motor with higher capacity.

Error code	Error	Message	Reason	Solution
H-7380	Encoder abnormal	Encoder abnormal	The internal parameters of DA's EEPROM are abnormal.	Replace the DA or update the EEPROM.
H-2350	Software temperature limit reached	Software temperature limit reached	1. Frequent full load operations of the motor. 2. Excessive continuous load of the motor.	1. Please recalculate and adjust the load conditions and movement conditions or select the motor model again. 2. Correct the mechanisms.
H-7180	Motor cannot be connected	Motor cannot be connected	1. Servo malfunction. 2. Motor malfunction. 3. Motor power cable has poor wiring or poor connection.	1. Replace the driver module or battery module. 2. Change the motor. 3. Please fix the wiring or replace cables.
H-4310	Driver temperature abnormal	Driver temperature abnormal	1. Servo malfunction. 2. The ambient temperature exceeds 45°C.	1. Reconnect the power to see if it is eliminated; otherwise replace the driver. 2. Implement countermeasures to reduce the temperature of the operating environment.
H-3220	Insufficient main circuit voltage	Insufficient main circuit voltage	1. Voltage of AC power is under 140 Vac or there is temporary power outage. 2. Servo malfunction. 3. Voltage of the power decreased while running.	1. Please adjust the voltage of the AC power to within the specified range of the driver. 2. Reconnect the power to see if it is eliminated; otherwise replace the battery module (DP)/backboard of the main box. 3. Decrease the acceleration magnitude.
H-5280	Encoder initialization abnormal (5V abnormal)	Encoder initialization abnormal (5V abnormal)	DA module abnormal.	Reconnect the power and if the error is not eliminated, replace the DA module.
H-FF06	Driver phase initialization error	Driver phase initialization error	Motor angle of the encoder is abnormal.	Reconnect the power and if the error is not eliminated, replace the motor.
H-7381	Encoder communication abnormal	Encoder communication abnormal	1. Encoder cable broken. 2. Encoder failure. 3. Encoder communication interfered or encoder cable broken.	1. Connect the encoder cable correctly. 2. Please reconnect the power again; if the alarm still occurs, please replace the motor. 3. Please add an anti-interference magnetic ring or replace the encoder cable with ones that have shielding effects.
H-FF02	Current control error	Current control error	1. Servo malfunction. 2. Motor malfunction.	1. Check the parameter id_pwr_board, write 100W: manufacture_data =2 I2CEE.Sate=11 400W: manufacture_data =3

Error code	Error	Message	Reason	Solution
				I2CEE.Sate=11 2. Change the motor.
H-FF03	STO triggered	STO triggered	1. Abnormal emergency stop switch connection or unreleased triggered emergency stop switch. 2. DIO module abnormal. 3. Electrical interference.	1. Release the emergency stop switch trigger. 2. Replace the DIO module. 3. Keep away and remove interference sources, or implement EMC countermeasures of magnetic rings.
H-FF04	HFLT inconsistent	HFLT inconsistent	1. Servo malfunction. 2. Electrical interference.	1. Change the DA module of that axis. 2. Keep away and remove interference sources, or implement EMC countermeasures of magnetic rings.
H-3210	DC bus voltage abnormal	DC bus voltage abnormal	1. Servo malfunction 2. Insufficient capacitance for the driver.	1. Change the DA/DP module. 2. Change the HCV1 module.
H-7580	ECAT interface cannot be detected	ECAT interface cannot be detected	1. Firmware version error. 2. EtherCAT communication interface of driver malfunctioned.	1. Confirm that the firmware version is 8.197 or above. 2. Check whether the connector pins of the DA module are short-circuited and replace the DA module.
H-8613	CiA-402 return to origin error	CiA-402 return to origin error	This process cannot be completed.	Reconnect the power and execute (reload the DA module parameters).
H-5180	Fan error	Fan error	1. Fan cable disconnected, fan stuck or too dirty. 2. Continuous alarm from the DA module of a specific axis.	1. Change the fan. 2. Change the HCV1 backboard or DA module.
H-FF07	Absolute encoder error	Absolute encoder error	1. Power of encoder in robot abnormal. 2. Battery has no voltage or low voltage. 3. Parallel battery failure or reversed polarity 4. Battery wiring installation abnormal. 5. Did the upper position send the correct clear instruction?	1. Change the power supply. 2. Change the battery. 3. Confirm the battery life or change the battery. 4. Adjust the wiring. 5. Send the correct clear instruction (enter the value 7 for SDO object number: 0x2060).

7.4. Electric gripper(04-XX-XX)

7.4.1. Hardware Error (04-01-XX)

Error code	Error	Message	Reason	Solution
04-01-11	Electric gripper data return error.	Electric gripper data return error.	Electric gripper connection failed, and no data was returned.	Check that the 24V power supply is properly connected. Check that the USB cable is properly connected, Check that the serial port is set correctly. Refer to the manual to install the gripper driver.
04-01-12	Number of gripper serial port exceeds the upper limit.	Number of gripper serial port exceeds the upper limit.	Exceeded the connection port name limit.	Modify the connection port setting is less than or equal to COM99.
04-01-13	Gripper hardware is not connected.	Gripper hardware is not connected.	Connection port is disconnected.	Re-plug the USB cable and reconnect it.
04-01-14	Gripper serial port are closed.	Gripper serial port are closed.	Gripper serial port is not turned on.	Close this serial port and reconnect.
04-01-15	Gripper serial port not available.	Gripper serial port not available.	Unable to achieve serial port.	Re-plug the USB cable and reconnect it. Replace the USB cable.
04-01-16	Gripper reconnection failed.	Gripper reconnection failed.	Connection port is interrupt and an attempt to reconnect failed.	Re-plug the USB cable and reconnect it. Replace the USB cable.
04-01-17	Gripper serial port is repeatedly connected.	Gripper serial port is repeatedly connected.	When a duplicate connection port is detected, the connection is automatically disconnected.	Check if the gripper is repeatedly connected.

7.4.2. Operation Error (04-02-XX, 04-01-8X)

Error code	Error	Message	Reason	Solution
04-01-20	Gripper model setting error.	Gripper model setting error.	Gripper type setting is incorrect.	Check that the gripper type setting is correct.
04-01-21	Gripper id error	Gripper id error	Setting multiple gripper, but some grippers are not connect.	This function is currently unavailabel.
04-01-22	Repeat gripper command.	Repeat gripper command.	Repeat the instructions in succession.	Wait for the gripper Busy to end, and then issue a new order.
04-01-23	Gripper position setting error.	Gripper position setting error.	Gripper position setting is greater than the total stroke.	Check that the gripper movement position input is correct.
04-01-24			Gripper position setting is less than zero	
04-01-25	Gripper speed setting error.	Gripper speed setting error.	Gripper moving speed setting is greater than the preset range.	Check if the gripper moving speed input is correct.
04-01-26			Gripper moving speed setting is less than the preset range.	
04-01-27	Gripper position direction setting error.	Gripper position direction setting error.	Gripper movement direction setting is incorrect.	Check that the gripper movement direction input is correct.
04-01-28	The gripping displacement setting is incorrect.	The gripping displacement setting is incorrect.	Gripping displacement setting is greater than the range of motion.	Check that the gripping displacement input is correct.

Error code	Error	Message	Reason	Solution
04-01-29			Gripping displacement setting is smaller than the range of motion.	
04-01-2A	Gripping speed setting is incorrect.	Gripping speed setting is incorrect.	Gripping speed is greater than the preset range.	Check that the gripping speed input is correct.
04-01-2B			Gripping speed is smaller than the preset range.	
04-01-2C	Gripping force setting is incorrect.	Gripping force setting is incorrect.	Gripping force is greater than the preset range.	Check that gripping force input is correct.
04-01-2D			Gripping force is smaller than the preset range.	
04-01-2E	Gripping failed.	Gripper failed to grip.	After the user turns on the grip detection function, the gripping action is performed and the electric gripper detects the unwound object.	This alarm is used to detect if the jaws are clamped to the object. If you do not need to send this detection alarm, you can cancel this function in the setting interface.
04-01-2F	Gripper is set incorrectly with respect to the total stroke.	Gripper is set incorrectly with respect to the total stroke.	Gripper exceeds the movable range with respect to the total stroke.	Check if the gripper moving distance and the total gripping displacement are correct.
04-01-80			Gripper is less than the movable range relative to the total stroke	
04-01-81	Gripper speed setting error.	Gripper speed setting error.	Gripper moving speed is less than the gripping speed.	Check that the gripper moving speed input is correct.

7.4.3. Electric Gripper Controller Alarm Signal Error (04-01-3X)

Error code	Error	Message	Reason	Solution
04-01-30	Gripper reset error	Gripper reset error	Some workpiece have not been removed during the route. Finger design interferes with the stroke	Check that there are no foreign objects in the itinerary. Modify the finger design.
04-01-31	Gripper position error	Gripper position error	Obstacles in the movement of the gripper.	Check and eliminate obstacles in the route.
04-01-32	Gripper overtravel	Gripper overtravel	Gripper displacement setting is greater than the range of motion.	Check that the gripper displacement input is correct.
04-01-33	Electric gripper returns to origin error	Electric gripper returns to origin error	Electric gripper power not connected There are obstacles in the stroke of the electric gripper	Check whether the power of the electric gripper is connected Check for and eliminate obstacles in the stroke

7.4.4. Electric Gripper Command Communication Timeout (04-01-4X)

Error code	Error	Message	Reason	Solution
04-01-41	Gripper connection timeout	Gripper connection timeout	Electric gripper command communication failed and data returned timeout.	Check that the 24V power supply is properly connected. Check that the USB cable is properly connected, Check that the serial port is set correctly. Refer to the manual to install the jaw driver. Replace the controller unit.
04-01-42	Gripper firmware communication timeout	Gripper firmware communication timeout		
04-01-43	Gripper stop action timeout	Gripper stop action timeout	Electric gripper command	Check that the 24V power supply is properly connected.

Error code	Error	Message	Reason	Solution
04-01-44	Gripper reset timeout	Gripper reset timeout	communication failed and data returned timeout.	Check that the USB cable is properly connected, Check that the serial port is set correctly. Refer to the manual to install the jaw driver. Replace the controller unit.
04-01-45	Gripper movement timeout	Gripper movement timeout.		
04-01-46	Gripping timeout	Gripping timeout		
04-01-47	Gripper expert mode action timeout	Gripper expert mode action timeout		
04-01-48	Gripper state reading timeout	Gripper state reading timeout		

7.5. SDK Server (SDK server,05-XX-XX)

7.5.1. Server Error (05-01-XX)

Error code	Error	Message	Reason	Solution
05-01-10	SDK server disconnected abnormally	SDK server disconnected abnormally	SDK server disconnected abnormally	Please confirm the connection status of the SDK server
05-01-11	SDK server port error	SDK server port error	SDK server port opening failed	Please confirm the connection status of the SDK server

8. Program Examples

This manual does not contain all instructions; if you need the complete instruction set, please refer to the “Robot software instruction set manual” on our official website.

8.1. Register

8.1.1. COUNTER Register

Program:

```
$C[1] = 10
```

Description:

The constant 10 is saved into COUNTER 1. After the program is closed, the number of the variable definition still registered.

Hint:

There are 20 COUNTERs from 1 to 20. The saved number is integer. The storage capacity is 32bit, which is -2147483648 ~ 2147483647.

8.1.2. TIMER Register

Program:

```
$T[1] = 0
```

```
WAIT SEC 0
```

```
$T_STOP[1] = FALSE
```

```
PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
```

```
WAIT SEC 0
```

```
$T_STOP[1] = TRUE
```

Description:

Calculate the period when the robot moves from the original position to P0. After the program is closed, the number of the variable definition still registered.

Parameter explanation:

Start counting when \$T_STOP[n]=FALSE. And stop when \$T_STOP[n]=TRUE.

Hint:

There are 20 TIMERs, from 1 to 20. \$T[n] represents the TIMER n. Before starting and ending \$T_Stop, "WAIT SEC 0" command which can stop pre-read is necessary. Every TIMER is 32bit, the display range is from -2147483648 ~ 2147483647(ms).

8.2. Variable Type

8.2.1. REAL

Program:

REAL One

One = 1

Description:

The format is similar to the data type of the decimal data. This variable will disappear after the program is closed.

Hint:

The storage capacity is 32bit about $10^{-37} \sim 10^{38}$, effective to 6 digits after the decimal point.

8.2.2. INT

Program:

INT Two = 2

Description:

Which is a format of the integer-type data, and will disappear after the program is closed.

Hint:

The storage capacity is 32bit, which are -2147483648 ~ 2147483647. You can enter a decimal integer or start with 0x to indicate a hexadecimal integer.

8.2.3. UINT

Program:

UINT Two = 2

UINT ARRAY[3]

Description:

The format is a no-symbol integer type, meaning a non-negative value. After the program is closed, the value defined by the variable will disappear.

Hint:

The storage capacity is 32bit. Range: $0 \sim 4294967295 (2^{32}-1)$. You can enter a decimal integer or start with 0x to indicate a hexadecimal integer.

8.2.4. BOOL

Program:

BOOL K = TRUE

Description:

Which means “Boolean”, is a logically variable. Will disappear after the program is closed.

Hint:

Used to declare the variable represents TRUE or FALSE.

8.2.5. CHAR

Program:

CHAR COLOR = 'R'

Description:

Which represents the character variable. Will disappear after the program is closed.

Hint:

Used to declare the variable represents the specific characters.

8.2.6. STRING

Program:

STRING STR = “HIWIN”

Description:

String variable; after the program is closed, the value defined by the variable will disappear.

Hint:

Used to declare that the variable represents specific strings; the string length is 0~255.

8.2.7. E6POS Point

Program:

E6POS POINT = {X 0,Y 300,Z 200}

PTP POINT **CONT**=100% **Vel**=100% **Acc**=50% **TOOL**[0] **BASE**[0]

Description:

Define POINT in Cartesian coordinate, and move the robot to POINT.

Hint:

If the parameter is not set, its value will not be changed (A, B, C in this case). This point doesn't define E6AXIS (A1~A6) values.

8.2.8. E6AXIS Point

Program:

E6AXIS POINT = {A1 90}

PTP POINT CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

Description:

Define POINT in the joint coordinate, and move the robot to POINT.

Hint:

Parameter A2、A3、A4 are not set, and will remain the original value. This point doesn't define E6POS (X, Y, Z, A, B, C) value.

8.2.9. E6POINT Point

Program:

E6POINT HOME = {Y 200,Z -1000,A 90}

PTP HOME CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

Description:

The first definition of HOME is in Cartesian coordinate, and then move the robot to HOME. The second definition is in Joint coordinate.

Hint:

If there is parameter not defined, it will remain the current value.

8.3. Operator

8.3.1. Arithmetic Operator

Program:

INT a, b, e

REAL c, d, f

REAL total

a= 3

b= 5

c= 0.6

d= 12.2

e= 10

f= 10.0

a= a*b ; a= 3*5= 15

b= b+d ; b= 5+12.2= 17.2 →Round down , b= 17

c= c*d ; c= 0.6*12.2= 7.32

d= b+d ; d= 17+12.2= 29.2

e= e/2 ; e= 5

e= 10/4 ; e= 2(Round down)

e= f/4 ; e= 2(Round down)

f= f/4 ; f= 2.5

f=b*c

d=b%a ; d= 17%15(Take the remainder) = 2

total = a*c

;INT*REAL will convert REAL into INT and then multiply them

; 15*INT(7.32) = 15*7 = 105

Hint: If the format is INT and there are decimals after operation, decimals will be removed. After INT and REAL are operated by "+", "-", or "*", the result format will be REAL.

8.3.2. Logic Operator

Logic Operator		A AND B	A OR B
A=TRUE	B=TRUE	TRUE	TRUE
A=TRUE	B=FALSE	FALSE	TRUE
A=FALSE	B=TRUE	FALSE	TRUE
A=FALSE	B=FALSE	FALSE	FALSE

8.3.3. Relation Operator

Relation Operator	A > B	A >= B	A < B	A <= B	A == B	A != B
A = 2, B = 1	TRUE	TRUE	FALSE	FALSE	FALSE	TRUE
A = 1, B = 1	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE
A = 1, B = 2	FALSE	FALSE	TRUE	TRUE	FALSE	TRUE

8.4. Input/Output

8.4.1. Digital Input

Program:

`WAIT FOR $DI[1] == TRUE`

Parameter explanation:

Wait for the Digital Channel 1 inputs become TRUE.

8.4.2. Digital Output

Program:

`$DO[1] = TRUE`

Parameter explanation:

The Digital Channel 1 outputs TRUE.

8.4.3. Robot Input

Program:

`WAIT FOR $RI[1] == TRUE`

Parameter explanation:

Wait for the Channel 1 of Robot signal inputs become TRUE.

8.4.4. Robot Output

Program:

`$RO[1] = TRUE`

Parameter explanation:

The Channel 1 of Robot signal outputs TRUE.

8.4.5. Valve Output

Program:

\$VO [1] = TRUE

Parameter explanation:

The Channel 1 of Solenoid Valve outputs TRUE.

8.5. Motion Function

The way to define the point can be:

1. Establish the point with the software frame.
2. Establish the point of E6POS or E6AXIS.
3. Define the point parameter directly. The coordinates not defined will remain the same, for example, PTP {X 200}.
4. Define the joint angle directly, and the parameter not defined will be the current value, like PTP {A1 90, A3 60}.

8.5.1. PTP

Point Definition 1

Program:

PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

Description:

Only require the position of the starting point and the terminal point. There is no limit for the middle process. TCP will be guided with the fastest trace of the robot to the target point. P0 is additionally established for TCP except for the Home status. TCP will move point-to-point from Home to P0.

Parameter explanation:

PTP ; name of point-to-point command, the shortest trace for the robot

P0 ; any point except for Home

CONT ; smooth extent

Vel ; moving velocity relative to maximum velocity

Acc ; moving acceleration relative to maximum acceleration

Point Definition 2

Program:

E6POS POINT = {X 0,Y 300,Z 200}

PTP POINT CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

Description:

Move to POINT.

Hint: The same way to establish points with E6AXIS.

Point Definition 3

Program:

PTP {X 100} CONT=100% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]

Description:

The TCP moves to this coordinate (refer to the base coordinate). The parameters not defined will remain the same.

Point Definition 4

Program:

PTP {A1 45} CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

Description:

The A1 axis of TCP moves to +45° (refer to the base coordinate). For the axis not defined, the angle will not change.

8.5.2. PTP_REL

Point Definition 1

Program:

PTP_REL {X 100 } CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

Description:

The TCP moves to this coordinate (refer to the base coordinate). The coordinates not defined will remain the same.

Point Definition 2

Program:

PTP_REL {A1 45} CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

Description:

The A1 axis of TCP moves at +45° relative to the original A1 axis (refer to the base coordinate). For the axis not defined, the angle will not change.

8.5.3. LIN

Point Definition 1

Program:

```
LIN P0 CONT=100% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]
```

Description:

P0 is additionally established for TCP except for the Home status. TCP will move point-to-point from Home via P0 to P1. The robot will guide TCP to the target point along the linear trace with the defined velocity.

Parameter explanation:

LIN ; name of point-to-point command, linear trace connecting two points

P0 ; any point except for Home

CONT ; smooth extent

Vel ; velocity moving on linear trace

Acc ; acceleration moving on linear trace

Point Definition 2

Program:

```
E6POS POINT = {X 0,Y 368,Z 293}
```

```
LIN POINT CONT=100% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]
```

Description: move to POINT

Hint: Same method to establish points with E6AXIS

Point Definition 3

Program:

```
LIN {X 100} CONT=100% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]
```

Description:

The TCP moves to this coordinate (refer to the base coordinate). The coordinates not defined will remain the same.

Point Definition 4

Program:

`LIN {A1 45} CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]`

Description:

The A1 axis of TCP moves at +45° relative to the original A1 axis (refer to the base coordinate). For the axis not defined, the angle will not change.

8.5.4. LIN_REL

Point Definition 1

Program:

`LIN_REL {X 100 } CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]`

Description:

The coordinates of TCP moves in relative to this coordinate (refer to the base coordinate). For the direction not defined, the coordinates will not change.

Point Definition 2

Program:

`LIN_REL {A1 45} CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]`

Description:

The A1 axis of TCP moves at +45° relative to the original A1 axis (refer to the base coordinate). For the axis not defined, the angle will not change.

Program:

`LIN_REL {X 100 } CONT=100% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]`

8.5.5. LIN_REL_TOOL

Point Definition 1

Program:

`LIN_REL_TOOL {X 100 } CONT=100% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]`

Description:

The TCP of the robot will move along the axis of the command(“X” in this case) by increasing the value(“100” in this case).

Point Definition 2

Program:

`LIN_REL_TOOL {A 45} CONT=100% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]`

Description:

In this case, the TCP will rotate +45° along the X axis of the TCP coordinate. And the command “B”(“C”) means to rotate along “Y”(“Z”) axis.

8.5.6. CIRC

Point Definition 1

Program:

```
CIRC P0 P1 CONT=100% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]
```

Description:

P0 and P1 are additionally established for TCP except for the Home status. TCP will move with circular trace from Home via P0 to P1. The robot will guide TCP to the target point along the circular trace with the defined velocity.

Parameter explanation:

CIRC; name of point-to-point command, the starting point arrives the target point via the auxiliary point along the circular trace

P0 ; any point except for Home as auxiliary point

P1 ; any point except for Home as target point

CONT ; smooth extent

Vel ; velocity moving on circular trace

Acc ; acceleration moving on circular trace

Hint:

P0 and P1 should be established first.

Point Definition 2

Program:

```
E6POS POINT1 = {X 0,Y 300,Z 200}
```

```
E6POS POINT2= {X 20,Y 320,Z 220}
```

```
CIRC POINT1 POINT2 CONT=100% Vel=2000mm/s Acc=50% TOOL[0] BASE[0]
```

Description: Move to POINT2 via POINT1.

Hint: The points are established the same method as E6AXIS.

Point Definition 3

Program:

CIRC {X 0, Y 450} {X -150, Y 300} **CONT**=100% **Vel**=2000mm/s **Acc**=50% **TOOL**[0] **BASE**[0]

Description:

TCP based on the starting point as the original coordinate moves to the auxiliary point and then arrives the destination point (refer to the base coordinates).

Point Definition 4

Program:

CIRC {A1 5.0, A2 5.0, A3 5.0, A4 5.0} {A1 10.0, A2 10.0, A3 10.0, A4 10.0} **CONT**=100% **Vel**=2000mm/s **Acc**=50%
TOOL[0] **BASE**[0]

Description:

TCP based on the starting point as the original coordinate moves to the auxiliary point and then arrives at the destination point (refer to the base coordinates).

8.5.7. CIRC_REL

Point Definition 1

Program:

CIRC_REL {X -150, Y 150} {X -150, Y -150} **CONT**=100% **Vel**=2000mm/s **Acc**=50% **TOOL**[0] **BASE**[0]

Description:

TCP based on the starting point as the original coordinate moves to the auxiliary point and then arrives the destination point (refer to the base coordinates).

Point Definition 2

Program:

CIRC_REL {A1 5.0, A2 5.0, A3 5.0, A4 5.0} {A1 10.0, A2 10.0, A3 10.0, A4 10.0} **CONT**=100% **Vel**=2000mm/s
Acc=50% **TOOL**[0] **BASE**[0]

Description:

TCP based on the starting point as the original coordinate moves to the auxiliary point and then arrives at the destination point (refer to the base coordinates).

8.5.8. SPLINE

Point Definition 1

Program:

```
E6POINT P1 = { X 95 , Y 0 , Z -500 }  
E6POINT P2 = { X 94.63849632 , Y 3.922008424 , Z -500 }  
.....  
E6POINT P54 = { X -8.279795561 , Y -44.82876141 , Z -500 }  
E6POINT P55 = { X 0 , Y -45 , Z -500 }  
E6POINT P56 = { X 8.279795561 , Y -44.82876141 , Z -500 }  
.....  
E6POINT P73 = { X 95 , Y 0 , Z -500 }  
SPLINE  
SPL P1  
SPL P2  
.....  
SPL P54  
SPL P55  
SPL P56  
.....  
SPL P73  
ENDSPLINE
```

Description:

Start from P1 point and move to P73 point with B-Spline curvilinear motion.

8.5.9. Array Accumulation

Program:

```
PTP P0 CONT=100% Vel=100% Acc=100% TOOL[0] BASE[1]  
P0.A1 = P0.A1 + 10  
PTP P0  
P0.A1 = P0.A1 + 10  
PTP P0
```

Description:

The A1 coordinate of P0 accumulates 10 degrees every time, and the other coordinates will not change.

8.5.10. CT_A6

Program:

```
LIN P0 FINE=1 Vel=100mm/s Acc=100% TOOL[0] BASE[0]
CT_A6 100
WHILE $C[1] <2
$C[1] = $C[1]+1
LIN P1 FINE=1 Vel=100mm/s Acc=100% TOOL[0] BASE[0]
LIN P2 FINE=1 Vel=100mm/s Acc=100% TOOL[0] BASE[0]
ENDWHILE
$C[1] = 0
CT_A6 -50
WHILE $C[2] <2
$C[2] = $C[2]+1
LIN P1 FINE=1 Vel=100mm/s Acc=100% TOOL[0] BASE[0]
LIN P2 FINE=1 Vel=100mm/s Acc=100% TOOL[0] BASE[0]
ENDWHILE
$C[2] = 0
CT_A6 0
WAIT SEC 1
LIN P0 FINE=1 Vel=100mm/s Acc=100% TOOL[0] BASE[0]
```

Description:

The sixth axis of the robot first reciprocates between P1 and P2 at a speed of 100% in the positive direction, and then reciprocates between P1 and P2 at a speed of 50% in the negative direction, and then ends infinite rotation and returns to the P0 point.

8.5.11. BRAKE

Program:

```
LIN P0 FINE=1 Vel=3000mm/s Acc=100% TOOL[0] BASE[0]
LIN_REL {Z -200}
LOOP
IF $DI[1] == TRUE THEN
BRAKE
EXIT
ENDIF
ENDLOOP
LIN P1 FINE=1 Vel=3000mm/s Acc=100% TOOL[0] BASE[0]
```

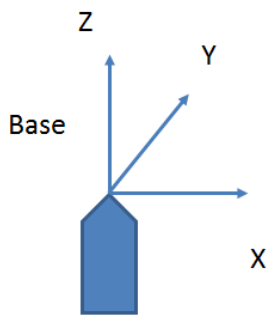
Description:

The robot moves to P0, it moves 200mm down along Z axis. The sensor of DI[1] is triggered on the way, the robot stop the motion and moves linearly from the stop point to P1.

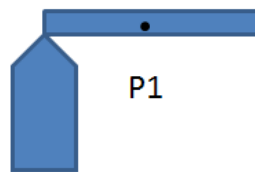
8.5.12. EXT_TCP

Front work:

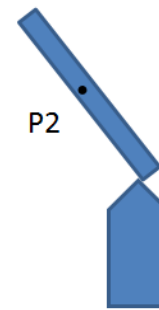
1. At the external tool point, teach a Base coordinate system, and the origin of the Base coordinate system is at the tool processing point.
2. Teaching starting point is at point P1 and ending point at P2



Teach Base



Starting Point



Ending Point

Program:

`EXT_TCP_START`

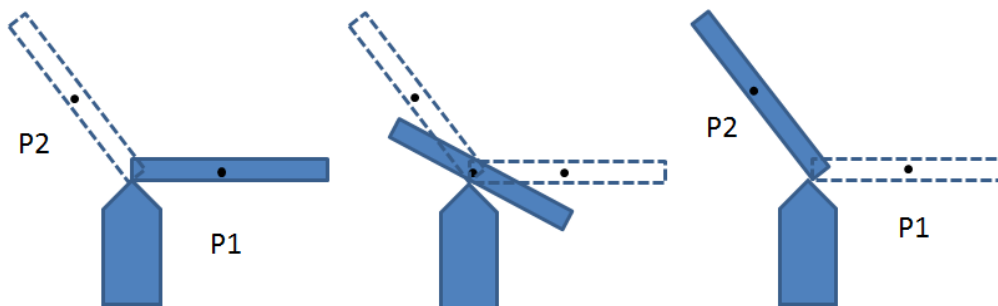
`LIN P1 FINE=1 Vel=1000mm/s Acc=100% TOOL[0] BASE[1]`

`LIN P2 FINE=1 Vel=1000mm/s Acc=100% TOOL[0] BASE[1]`

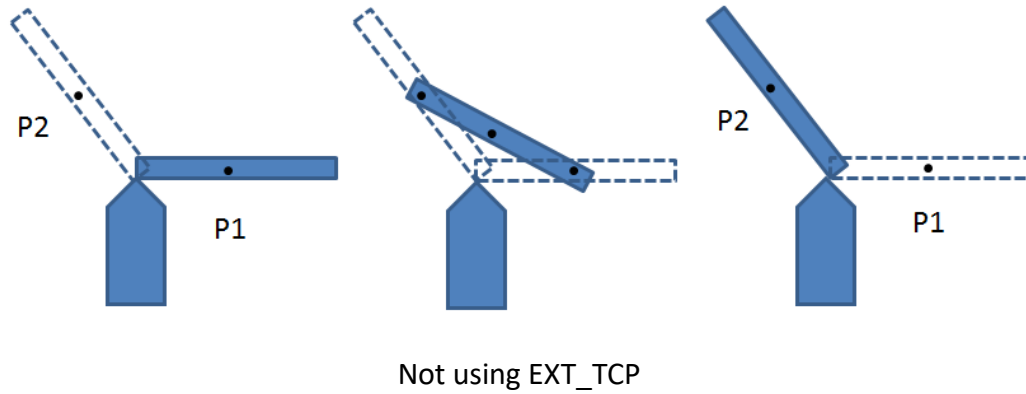
`EXT_TCP_END`

Description:

After the robot moves to P1, it is processed (polished) along the straight line of the workpiece and moved to P2. During the process, the workpiece remains in contact with the tool. If EXT_TCP is not used, the workpiece and tool will only be in contact at the start and end points



Using EXT_TCP



8.5.13. CHECK_LIN

Program:

```
$DO[1] = FALSE
```

```
$DO[2] = FALSE
```

```
LIN P0 FINE=1 Vel=2000mm/s Acc=100% TOOL[0] BASE[0]
IF CHECK_LIN(P0,P1) == TRUE THEN
$DO[1] = TRUE
LIN P1 FINE=1 Vel=2000mm/s Acc=100% TOOL[0] BASE[0]
ELSE
$DO[2] = TRUE
LIN P2 FINE=1 Vel=2000mm/s Acc=100% TOOL[0] BASE[0]
ENDIF
```

Description:

Use the CHECK_LIN instruction to confirm whether there are singular points between P0 and P1; if there are none, output DO [1] and move to point P1, if singular points exist, output DO [2] and move to P2.

8.5.14. FWD

Program:

```
E6AXIS P1_AXIS = {A1 0, A3 0, A4 0, A5 -90, A6 0}
```

```
E6POS P1_POS
```

```
E6POS P1={X 100}
```

```
P1_POS = FWD(P1_AXIS)
```

```
IF P1.X - P1_POS.X > 90 THEN
```

```
LIN P0 FINE=1 Vel=2000mm/s Acc=100% TOOL[0] BASE[0]
```

```
ELSE
```

```
LIN P1 FINE=1 Vel=2000mm/s Acc=100% TOOL[0] BASE[0]
ENDIF
```

Description:

Declare that one axis degree point P1_AXIS is converted to P1_POS coordinates through the FWD instruction, and determine whether its X-coordinate is more than 90mm from the X-coordinate of P1. If so, move to P0; otherwise, move to P1.

8.5.15. STOP_MOTION

Program:

```
E6POS RP = { X 222.4, Y 356.5, Z 293.500, A 180, B 0, C 90}
E6POINT P_COL
```

```
LIN RP
STOP_MOTION $DI[1]
P_COL = GETPOINT
RP.X = P_COL.X-10
RP.Y = P_COL.Y-10
RP.Z = P_COL.Z-10
LIN RP
LIN_REL {Z -100}
```

Description:

Specifies a target position RP. If DI [1] was triggered during the process, it means that obstacles were touched. Back up to where the DI [1] signal goes OFF and record the point, which is the P_COL position, then adjust the original RP position according to this and move down 100mm in the Z-direction.

8.6. Control Function

8.6.1. IF

- **Format 1 of IF**

```
IF condition THEN
```

```
.....
```

```
ENDIF
```

Program:

```
INT n = 1
IF n > 0 THEN
PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
```

ENDIF

Description:

Because the *condition* is true, TCP will move to P0.

Parameter description:

Condition; condition

Because the condition is true, the statement in IF will be executed.

● Format 2 of IF

IF *condition* THEN

.....

ELSE

.....

ENDIF

Program:

INT n = 0

IF n > 0 THEN

PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

ELSE

PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

ENDIF

Description:

Because the condition is false, TCP will execute ELSE and move to P0.

● Application for determining IF condition

IF ((TRUE) AND (TRUE)) THEN

.....

ENDIF

Program:

INT n, m

n = 1

m = 2

IF ((n == 1) AND (m == 2)) THEN

PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

ENDIF

Description:

Because the condition is true, TCP will move to P0.

```
IF ((TRUE) OR (FALSE)) THEN
```

```
.....
```

```
ENDIF
```

Program:

```
INT n,m
```

```
n =1
```

```
m =3
```

```
IF ((n == 1) OR (m == 2)) THEN
```

```
PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
```

```
ENDIF
```

Description:

Because the condition is true, TCP will move to P0.

```
IF condition THEN
```

```
.....
```

```
ENDIF
```

Program:

```
IF $DI[1] == TRUE THEN
```

```
PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
```

```
ENDIF
```

Description:

If DI[1] is true, the condition will be true. TCP will move to P0.

8.6.2. FOR

● FOR TO STEP ENDFOR

FOR *start* TO *last* STEP *increment*

```
.....
```

```
ENDFOR
```

Program:

```
INT n
```

```
FOR n = 0 TO 2 STEP 1
```

```
PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
```

```
PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
ENDFOR
```

Description:

TCP moves to and fro between P0 and P1 three times.

Parameter explanation:

start ; start

last ; condition

increment ; increment

After FOR is executed from the start to the condition, FOR will end.

If the STEP increment is omitted, the increment default is 1.

● FOR application

Program:

```
INT n
FOR n = 0 TO 20 STEP 10
PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
ENDFOR
```

Description:

TCP moves to and fro between P0 and P1 three times.

Program:

```
INT n
FOR n = 2 TO 0 STEP 1
PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
ENDFOR
```

Description:

TCP moves to and fro between P0 and P1 three times.

Program:

```
INT n
FOR n = -1 TO 3 STEP 2
PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
```

ENDFOR

Description:

TCP moves to and fro between P0 and P1 three times.

8.6.3. LOOP

● LOOP ENDLOOP

LOOP

.....

ENDLOOP

Program:

LOOP

PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

ENDLOOP

Description:

TCP repeatedly moves to and fro between P0 and P1.

Parameter explanation:

LOOP is an infinite loop.

● LOOP EXIT ENDLOOP

LOOP

.....

EXIT

.....

ENDLOOP

Program:

INT n =0

LOOP

IF n == 1 THEN

EXIT

ELSE

n = n + 1

ENDIF

PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

ENDLOOP

Description:

TCP will move to P0.

Parameter explanation:

LOOP execute to EXIT and end LOOP.

8.6.4. WHILE

● WHILE ENDLOOP

WHILE *condition*

.....

ENDWHILE

Program:

INT n = 2

WHILE n > 0

PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

n = n - 1

ENDWHILE

Description:

TCP moves to and fro between P0 and P1 twice.

Parameter explanation:

condition ; condition

When the condition of WHILE is true, repeatedly execute the statement in WHILE until the condition is false and ends.

● Application for determining WHILE condition

WHILE ((TRUE) AND (TRUE))

.....

ENDWHILE

Program:

INT n,m

n = 1

m = 2

```

WHILE ((n == 1) AND (m == 2))
PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
n = n + 1
ENDWHILE

```

Description:

TCP moves to and fro between P0 and P1 once.

```

WHILE ((TRUE) OR (FALSE))
.....
ENDWHILE

```

Program:

```

INT n,m
n = 1
m = 2
WHILE ((n == 1) OR (m == 3))
PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
ENDWHILE

```

Description:

TCP moves to and fro between P0 and P1 once.

8.6.5. REPEAT

- **REPEAT UNTIL**

```

REPEAT
.....
UNTIL condition

```

Program:

```

INT n =0
REPEAT
PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
n = n + 1
UNTIL n > 2

```

Description:

TCP will move to P0 and P1 as well as repeatedly execute twice.

Parameter explanation:

Condition ; condition

Repeatedly execute the statement in REPEAT until the condition is true, and end REPEAT.

● **Application for determining REPEAT condition**

REPEAT

.....

UNTIL((FALSE) OR (TRUE))

Program:

INT n =0

INT k =1

REPEAT

PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

n = n + 1

UNTIL (k ==2) OR (n > 2)

Description:

TCP will move to P0 and P1 as well as repeatedly execute twice.

REPEAT

.....

UNTIL((TRUE) AND (TRUE))

Program:

INT n =0

INT k =1

REPEAT

PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

n = n + 1

UNTIL(k ==1) AND (n > 2)

Description:

TCP will move to P0 and P1 as well as repeatedly execute twice.

8.6.6. GOTO

```
IF condition THEN  
GOTO LABEL1  
ENDIF  
IF condition THEN  
GOTO LABEL2  
ENDIF  
IF condition THEN  
GOTO LABEL3  
ENDIF
```

LABEL1:

.....

LABEL2:

.....

LABEL3:

.....

Program:

```
INT n =0  
LOOP  
IF n == 0 THEN  
GOTO STEP0  
ENDIF  
IF n == 1 THEN  
GOTO STEP1  
ENDIF  
IF n == 2 THEN  
GOTO STEP2  
ENDIF
```

PRO:

```
n = n + 1  
ENDLOOP
```

STEP0:

```
PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]  
GOTO PRO
```

STEP1:

```
PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
```

GOTO PRO
STEP2:

Description:

TCP moves from P0 to P1, and then ends LOOP.

Parameter explanation:

LABEL ; label

The label of GOTO corresponds to the following statement of the label. If the label doesn't have the statement, it will end program. The specified label must be in the current function, cross function cannot be applied.

8.6.7. SWITCH

- **SWITCH without default**

SWITCH *number*

.....

CASE *number1*

.....

CASE *number2*

.....

ENDSWITCH

Program:

INT n =0

LOOP

SWITCH n

CASE 0

PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

CASE 1

PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

CASE 2

EXIT

ENDSWITCH

n = n + 1

ENDLOOP

Description:

TCP moves from P0 to P1, and then executes EXIT to end LOOP.

Parameter explanation:

number ; argument

The argument of SWITCH corresponds to the statement of CASE.

When the argument of SWITCH doesn't correspond to CASE, it will directly correspond to ENDSWITCH.

● **SWITCH with default**

SWITCH *number*

.....

CASE *number1*

.....

CASE *number2*

.....

DEFAULT

EXIT

ENDSWITCH

Program:

INT n =0

LOOP

SWITCH n

CASE 0

PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

CASE 1

PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

DEFAULT

EXIT

ENDSWITCH

n = n + 1

ENDLOOP

Description:

TCP moves from P0 to P1, and then executes EXIT to end LOOP.

Parameter explanation:

The argument of SWITCH corresponds to CASE. If there is no correspondence, the statement of DEFAULT will be executed.

When the argument of SWITCH doesn't correspond to CASE, the statement with DEFAULT will jump to the statement of DEFAULT.

● **SWITCH Extension 1**

SWITCH *number*

.....

CASE *number1, number3, number5*

.....

CASE *number2, number4*

.....

DEFAULT

EXIT

ENDSWITCH

Program:

INT n =0

LOOP

SWITCH n

CASE 0,2,4

PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

CASE 1,3

PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

CASE 5

EXIT

ENDSWITCH

n = n + 1

ENDLOOP

Description:

TCP moves to and fro between P0 and P1, moves to P0, and then executes EXIT to end LOOP.

● **SWITCH Extension 2**

SWITCH *character*

.....

CASE *character1*

.....

CASE *character2*

.....

DEFAULT

EXIT

ENDSWITCH

Program:

CHAR COLOR = 'R'

LOOP

```
SWITCH COLOR
CASE 'R'
PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
CASE 'G'
PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
DEFAULT
EXIT
ENDSWITCH
IF COLOR =='G' THEN
COLOR ='Y'
ENDIF
IF COLOR =='R' THEN
COLOR ='G'
ENDIF
ENDLOOP
```

Description:

TCP moves from P0 to P1, and then executes EXIT to end LOOP.

8.6.8. WAIT

- **WAIT SEC**

Program:

```
WAIT SEC 3
```

```
PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
```

Description:

After the program waits for three second, TCP will move to P0.

- **WAIT INPUT**

Program:

```
WAIT FOR $DI[1] == TRUE
```

```
PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
```

Description:

When the program waits the Digital INPUT Channel 1 is TRUE, TCP will move to P0.

Program:

```
WAIT FOR $RI[1] == TRUE
```

```
PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
```

Description:

When the program waits the INPUT Channel 1 for the robot is TRUE, TCP will move to P0.

8.6.9. QUIT

QUIT:

LOOP

IF \$DI[1] == TRUE THEN

QUIT

ENDIF

ENDLOOP

Description:

The program will be closed when executing "QUIT" command(when DI[1] == TRUE in this case).

8.7. Motion Parameter

8.7.1. CONT

Continue Trajectory

If the "CONT" command is called, the robot controller will consider the next motion point, and will move in a smoother path. The path smooth level will depend on the motion velocity and acceleration.

There are three kinds of CONT command: CONT, CONT = #%, CONT = #mm. Last two kinds represent the fixed path, like CONT = 50% or CONT = 30mm.

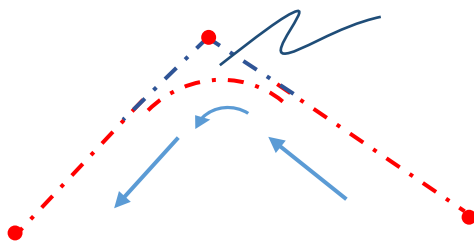
- CONT

LIN P1 CONT

LIN P2 CONT

LIN P3 CONT

When execute the commands open, there will be the smooth path in "LIN P2 CONT", but there won't move to P1 and P3(because they are the origin and destination).



CONT motion

- CONT = #%

LIN P1 CONT

LIN P2 CONT = 50%

LIN P3 CONT

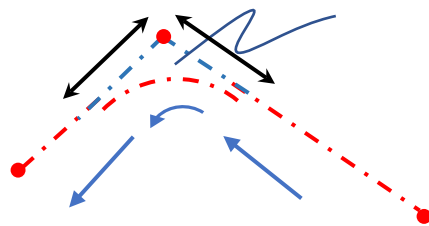
As Figure 10.2 show:

The distance between P2, P3 is shorter than the one between P1, P2, which is called the short length.

When execute the open program, the trajectory will start fairing when 50% short length away before P2.

CAUTION

Because the "CONT= # %" command will create a fixed path, so the velocity of the robot may change.



CONT = 50%

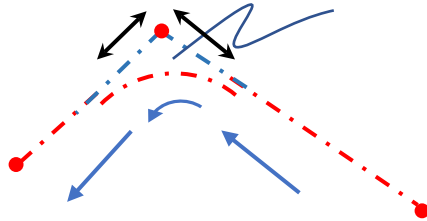
- CONT = #mm

LIN P1 CONT

LIN P2 CONT = 5mm

LIN P3 CONT

The path will start fairing when 5mm away before P2.



CONT = 5mm

⚠ CAUTION

Because the “CONT= # mm” command will create a fixed path, so the velocity of the robot may change.

⚠ CAUTION

When using “CONT= # mm” command, it does not mean that the value specified by the user will be exactly the same. However, the system will attempt at the distance specified by the user.

If there is “DO” command before the “CONT”, then the path won’t be smoothed.

Example :

LIN P1 CONT

LIN P2 CONT

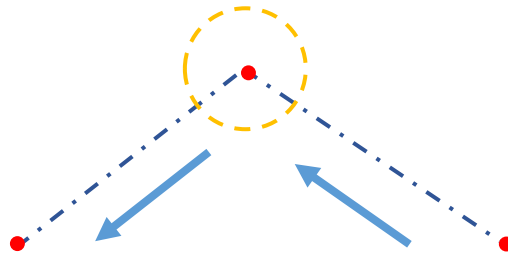
\$DO[1] = TRUE

LIN P3 CONT

Originally, the path moving to P2 should be faired, but in this case, the path won’t be smoothed.

8.7.2. FINE

Discontinuous Motion



FINE motion

While the program is executing, when executed to when the movement parameter is CONT, the program will read the program information after CONT in advance in order to plan the movement path after the CONT program in advance, so that the TCP movement path has smooth trajectory.

If the point is only a point on the path of TCP movement, CONT can be used to make the operation smooth. If the point is a point that TCP needs to reach precisely, FINE must be used.

FINE has four kinds:

- FINE, pre-read the next command, not check the actual position.
- FINE = 0, pre-read the next command, not check the actual position.
- FINE = 1, not pre-read the next command, not check the actual position.
- FINE = 2, not pre-read the next command, check the actual position.

8.7.3. VEL

Vel=100%

Define the velocity. If the PTP motion is used, the expression is the percentage that the maximum velocity can move. If this parameter is not entered, the default is 20%.

Vel=2000mm/s

Define the velocity. If the LINE and CIRC motions are used, the expression is mm/s. If this parameter is not entered, the default is 250mm/s.

8.7.4. ACC

Acc=50%

Define the acceleration. The expression is the percentage of maximum acceleration.

8.8. Definition of Structure

STRUC LABEL INT PARAMETER1, REAL PARAMETER2

DECL LABEL PART1 ,PART2, ,

PART1 = { PARAMETER1 10, PARAMETER2 500 }

PART2 = { PARAMETER1 20, PARAMETER2 100 }

.....

.....

Program:

STRUC CASTING_TYPE INT MASS, REAL VOLUME

DECL CASTING_TYPE PART1 ,PART2

PART1 = {MASS 10, VOLUME 500 }

PART2 = {MASS 20, VOLUME 100 }

Description:

For the different objects in the specific type, the different parameters can be assigned in the same variable.

Parameter explanation:

STRUC LABEL ; define the type name

INT PARAMETER1 ; define the format of object parameter

REAL PARAMETER2 ; define the format of object parameter

PART1 ; define the object

PART2 ; define the object

Hint:

PART1. PARAMETER1 = K, which can obtain the parameter.

8.9. Function & Subprogram

8.9.1. Definition & Using Method of Function

Function is a program code which allows the user to execute the specific task or specific motion. User may write the frequently repeated program code in the function, and may also decide to write the program code with any length in the function. Usually, one function only performs one task.

The declaration of function tells the compiler with respect to the function name, post back value and parameters.

Definition of Function:

```
DEFECT return_type function_name ( parameter list )
```

```
statement body of the function
```

```
RETURN...
```

```
ENDEFCT
```

The declaration of function includes the function header and function body. The description of each part is shown as follows:

return_type: Data type returns from function.

function_name: Function name.

parameter list: Function parameters. User may deliver the parameters into the function. The data type of parameter will refer to the data type and support point type declared in the function field. If the parameter is input, then use "parameter: IN" for indication, use as the input parameter, it will not affect the incoming variable even it is modified in the function. If it is used as the output variable and modified in the function, then use "parameter: OUT". As the output variable, if it is modified in the function, the originally incoming variable will also be changed accordingly. One function may have no function parameters, and up to five (5) parameters as the maximum.

statement body: Function body. If the function has parameter, then the user needs to declare the type of parameter in order to undertake.

Example of program 1:

```
INT iFUN
```

```
iFUN = FCT_1(2,3)
```

```
DEFECT INT FCT_1(num1:IN,num2:IN)
```

```
INT num1
```

```
INT num2
```

```
RETURN num1+num2
```

ENDFCT

Description:

Declare one function named as FCT_1, income two (2) INT parameters i.e. num1 and num2 respectively, and then post back after adding these two parameters.

Example of program 2:

```
E6POINT RE_E6,OUT_E6
```

```
INT iX
```

```
OUT_E6 = P1
```

```
RE_E6 = FCT_2(P0,OUT_E6)
```

```
DEFFCT E6POINT FCT_2(A:IN,B:OUT)
```

```
E6POINT A
```

```
E6POINT B
```

```
A.X = B.X
```

```
B.X = 100
```

```
PTP A
```

```
RETURN A
```

```
ENDFCT
```

```
iX = OUT_E6.X
```

Description:

Declare one function named as FCT_2, income one parameter of E6POINT and one output parameter B of E6POINT, the function assigns the X value of B to X of A first, and then configures X of B to 100, and then executes the point to point moving to A, and finally post back A, and B is taken as output returning to the calling procedure.

8.9.2. Definition & Using Method of Subprogram

The difference between the defined subprogram and function are: the subprogram has no post back value and the declaration is different.

Definition of subprogram:

```
DEF subprogram_name ( parameter list )
```

```
statement body of the subprogram
```

```
END
```

The declaration of function includes the subprogram header and subprogram body. Description of each part is shown as follows:

subprogram_name: Subprogram name.

parameter list: Subprogram parameter and up to five (5) parameters as the maximum.

statement body: Subprogram body.

Example of program 1:

```
INT iNUM
```

```
iNUM = 4
```

```
$C[4] = 0
```

```
PROG_1(3,iNUM)
```

```
$C[4] = iNUM
```

```
DEF PROG_1(num1:IN,num2:OUT)
```

```
INT num1
```

```
INT num2
```

```
num2= num1+num2
```

```
END
```

Description:

Declare one subprogram named as PROG_1, income one parameter num1 of INT and one output parameter num2 of INT, and then add these two parameters, and assign to num2 as the output.

Example of program 2:

```
E6POINT E6_OUT_A,E6_OUT_B
```

```
E6_OUT_A = P0
```

```
E6_OUT_B = P1
```

```
PROG_2(E6_OUT_A,E6_OUT_B)
```

```
DEF PROG_2(A:OUT,B:OUT)
```

```
E6POINT A
```

```
E6POINT B
```

```
A.X = B.X
```

```
B.X = 100
```

```
PTP A
```

```
END
```

Description:

Declare one subprogram named as PROG_2, income two (2) output parameters of E6POINT i.e. A and B respectively, the subprogram assigns X of B to X of A first, and then configures X of B to 100, and then executes the pint to point moving A, takes the modified A and B as the output returning to the calling procedure.

8.10. External Function & Subprogram

8.10.1. Definition & Using Method of External Function(EXTFCT)

Declare the external function which indicates that the user writes this function into a separate independent file, and the name of this independent file shall be the same as the function name, and call this function outside this independent file. The external function locates on the first line of program code and must begin with the keyword of DEFFCT, one file can only define one external function. If it desires to call the external function, it will have to declare the external function at the calling program. The declaration of external function must use the keyword of EXTFCT. After declaration, it will be the same as the calling of general function.

Definition of declaration for external function:

```
EXTFCT return_type function_name ( parameter list)
```

Description of each part of external function is shown as follows:

return_type: Type of post back value, structure of supporting point position.

function_name: Function name.

parameter list: Function parameters. Please be aware that the declaration of function parameter name (located in the procedure desired to call) must be consistent with the definition (located at the procedure being called) and up to five (5) parameters as the maximum, support the type of point position.

Example of program:

The program content of file named as FCT_1:

```
DEFFCT INT FCT_1 ( num1:IN,num2:IN)
INT num1
INT num2
RETURN num1+num2
ENDFCT
```

Content of external program:

```
EXTFCT INT FCT_1(num1:IN,num2:IN)
INT iNum
```

```
iNum = 10
```

```
iNum = FCT_1(6,8)
```

Description:

In the program with the file name of FCT_1, declare one function named as FCT_1, income two (2) parameters of INT i.e. num1 and num2 respectively, and then add these two parameters, and post back to the calling procedure, in addition, in another external file program, use EXTFACT to declare the external function of FCT_1, and then perform the calling by using the function mode directly, please be aware that the calling of external function can be up to eight (8) layers as the deepest, the compiler will report error if exceeded.

8.10.2. Definition &Using Method of External Subprogram(EXT)

Declare the external subprogram which indicates that the user writes this subprogram into a separate independent file, and the name of this independent file shall be the same as the subprogram name, and call this program outside this independent file. The external subprogram locates on the first line of program code and must begin with the keyword of DEF, one file can only define one external function. If it desires to call the external subprogram, it will have to declare the external subprogram at the calling program. The declaration of external subprogram must use the keyword of EXT. After declaration, it will be the same as the calling of general subprogram.

Definition of declaration for external subprogram:

```
EXT subprogram_name ( parameter list )
```

Description of each part of external subprogram is shown as follows:

subprogram_name: Subprogram name.

parameter list: Subprogram parameters. The declaration of subprogram parameter name (located in the procedure desired to call) must be consistent with the definition (located at the procedure being called) and up to five (5) parameters as the maximum, support the type of point position.

Example of program:

The program content of file named as PROG_1:

```
DEF PROG_1 ( num1:IN,num2:OUT )  
INT num1  
INT num2  
num2 = num1+num2  
END
```

Content of external program:

```
EXT PROG_1( num1:IN,num2:OUT )  
INT iNum
```

```
iNum = 7
```


CWRITE (HANDLE, SERDATA) ; write the number of SERDATA into
HANDLE
CREAD (HANDLE, SERDATA) ; give the number of HANDLE to SERDATA
CCLEAR (HANDLE) ; clear the number of HANDLE
CINQUIRE(HANDLE,NUM) ; read the received quantity

8.12. NET Configuration

Program:

- Single connection:

```
INT HANDLE
INT NUM
REAL ETHDATA
STRING ETH_STR
COPEN ( ETH , HANDLE)
LOOP
IF HANDLE > -1 THEN
CINQUIRE(HANDLE,NUM)
If NUM>0 THEN
CREAD (HANDLE, ETH_STR, ETHDATA)
ENDIF
CCLEAR (HANDLE)
ETHDATA = ETHDATA + 1
ETH_STR = "RETURN:"
CWRITE (HANDLE, ETH_STR, ETHDATA)
ENDIF
WAIT SEC 0.3
ENDLOOP
```

- Multiple connection:

```
INT HANDLE_1
INT HANDLE_2
INT NUM_1
INT NUM_2
REAL ETHDATA_1
REAL ETHDATA_2
HANDLE_1 = COPEN(ETH,192,168,0,5,5000)
HANDLE_2 = COPEN(ETH,192,168,0,6,5000)
```

```
LOOP
IF HANDLE_1 > -1 THEN
CINQUIRE(HANDLE_1,NUM_1)
If NUM_1>0 THEN
CREAD (HANDLE_1, ETHDATA_1)
ENDIF
CCLEAR (HANDLE_1)
ETHDATA_1 = ETHDATA_1 + 1
CWRITE (HANDLE_1, ETHDATA_1)
ENDIF

IF HANDLE_2 > -1 THEN
CINQUIRE(HANDLE_2,NUM_2)
If NUM_2>0 THEN
CREAD (HANDLE_2, ETHDATA_2)
ENDIF
CCLEAR (HANDLE_2)
ETHDATA_2 = ETHDATA_2 + 1
CWRITE (HANDLE_2, ETHDATA_2)
ENDIF

WAIT SEC 0.3
ENDLOOP
```

Description:

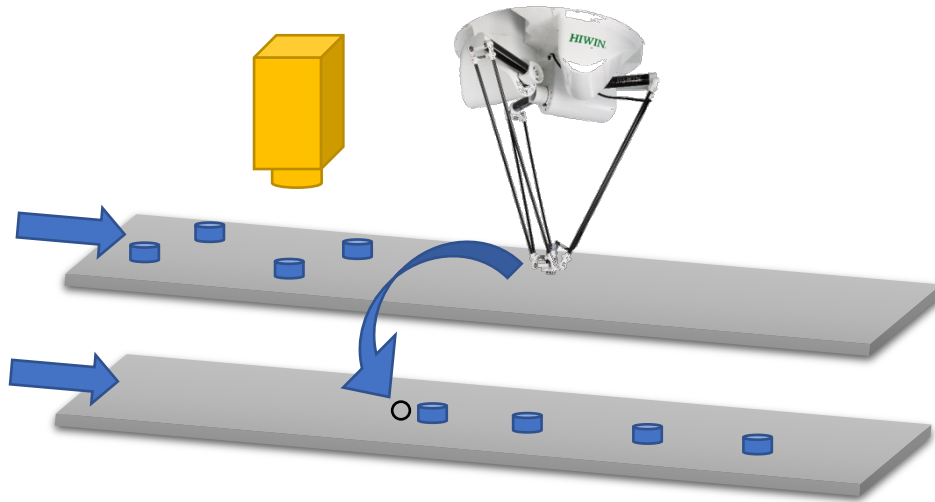
Program writing and reading the number via network

Parameter explanation:

ETH ; Internet
HANDLE ; target folder
CWRITE (HANDLE, ETHDATR) ; write the number of ETHDATR into HANDLE
CREAD (HANDLE, ETHDATR) ; give the number of HANDLE to ETHDATR
CCLEAR (HANDLE) ; clear the number of HANDLE
CINQUIRE(HANDLE,NUM) ; Read the received quantity

8.13. Conveyor Configuration

8.13.1. Pick Program(1)



Pick Example 1

Program description:

This is a visual example.

The robot picks the object from the Conveyor 1 to place on the Conveyor 2. The position is visually picked, and place P2 on the Conveyor 2.

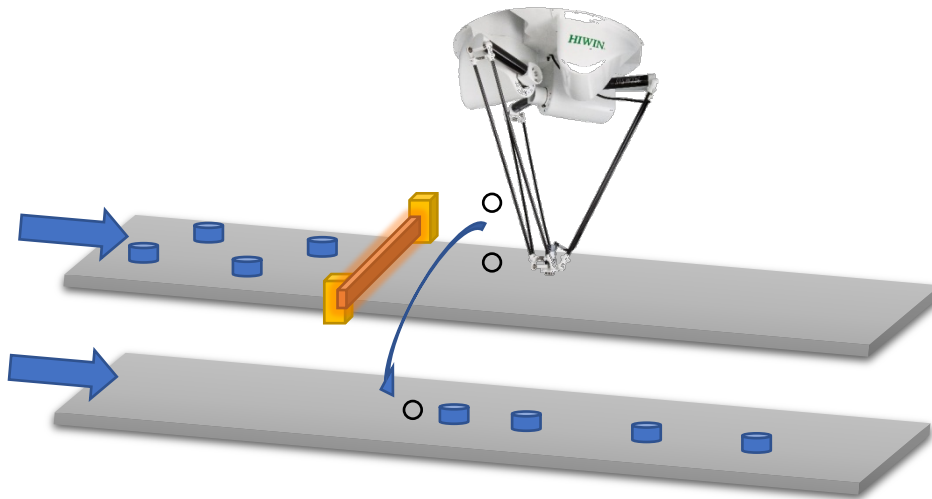
Program:

```

CNV_START CNV=1           ; start pick&place
CNV_PICK_QUANTITY = 2    ; set the maximum quantity to pick object
WHILE CNV_FULL == FALSE  ; go to loop when the quantity on the robot doesn't reach the upper limit..
CNV_PICK CNV=1 OBJ=1 $DO[1] Down=5.000mm FINE Vel=2000mm/s Acc=50% TOOL[0] BASE[0]      ;
execute pick
ENDWHILE
WHILE CNV_EMPTY == FALSE ; go to loop when the quantity on the robot is not empty.
CNV_PLACE $DO[1] P2 FINE Vel=2000mm/s Acc=50% TOOL[0] BASE[0]           ; execute place
ENDWHILE
CNV_END CNV=1           ; end pick&place

```

8.13.2. Pick Program(2)



Pick Example 2

Program description:

When the position to trigger a sensor is within the picking range, P can be directly set as the pick and place position.

The robot picks and places the object from the Conveyor 1 to the Conveyor 2. When the object is triggered by the sensor, the robot will move to P0 and pick, and then move to P1 and finally place P2.

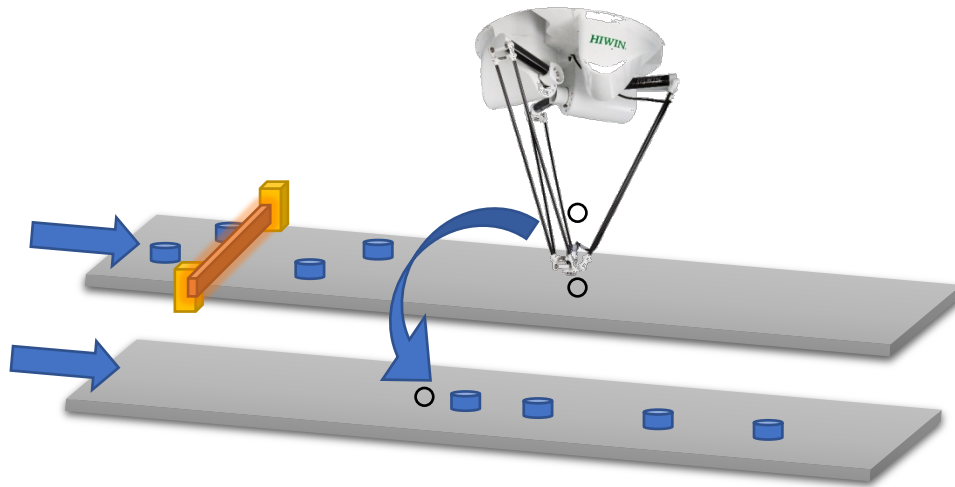
Program:

```

CNV_START CNV=1           ; start pick&place
CNV_PICK_QUANTITY = 2    ; set the maximum quantity to pick object
WHILE CNV_FULL == FALSE  ; go to loop when the quantity on the robot doesn't reach the upper limit.
CNV_PICK CNV=1 $DO[1] P0 Down=5.000mm FINE Vel=2000mm/s Acc=50% TOOL[0] BASE[0] ; execute pick
ENDWHILE
PTP P1 CONT Vel=100% Acc=50% TOOL[0] BASE[0]           ; move to P1
WHILE CNV_EMPTY == FALSE ; go to loop when the quantity on the robot is not empty.
CNV_PLACE CNV=2 $DO[1] P2 FINE Vel=2000mm/s Acc=50% TOOL[0] BASE[0] ; execute place
ENDWHILE
CNV_END CNV=1           ; end pick&place

```

8.13.3. Pick Program(3)



Pick Example 3

Program description:

When the position to trigger a sensor is beyond the picking range, the command E6POINT can be used to set the pick and place position.

(Before using the command E6POINT, please ensure the ToolBase coordinates have been parallel with those for the conveyor. So, you just need to adjust X coordinate or Y coordinate following P is adjusted).

The robot picks from the Conveyor 1 to the Conveyor 2, waits for the object to move to PICKPOINT, and then place to PLACEPOINT after moving to P1.

This example will release two objects after they are simultaneously picked.

Program:

```

CNV_START CNV=1                ; start pick&place
CNV_PICK_QUANTITY = 2          ; set the maximum quantity to pick object
E6POINT PICKPOINT = P0         ; set the pick point of E6POINT
PICKPOINT.X = PICKPOINT.X - 200
; If our ToolBase coordinate is parallel with the conveyor coordinate, X for PICKPOINT will be needed.
; the coordinate position minus 200, no change for Y coordinate
E6POINT PLACEPOINT = P2        ; set place point for E6POINT
PLACEPOINT.X = PLACEPOINT.X - 50
; If our ToolBase coordinate is parallel with the conveyor coordinate, X for PLACEPOINT will need to minus 50
and there is no change for Y coordinate.
WHILE CNV_FULL == FALSE        ; go to loop when the quantity on the conveyor doesn't reach the upper
limit
CNV_PICK CNV=1 $DO[1] PICKPOINT Down=0.000mm FINE Vel=2000mm/s Acc=50% TOOL[0] BASE[0]
; pick the first object

```

```

CNV_PICK CNV=1 $DO[2] PICKPOINT Down=0.000mm FINE Vel=2000mm/s Acc=50% TOOL[0] BASE[0]
    ; pick the second object
ENDWHILE
PTP P1 CONT Vel=100% Acc=50% TOOL[0] BASE[0]    ; move to P1
WHILE CNV_EMPTY == FALSE    ; go to loop when the quantity on the conveyor is not empty
CNV_PLACE CNV=2 $DO[1] PLACEPOINT FINE Vel=2000mm/s Acc=50% TOOL[0] BASE[0]    ; execute pick
ENDWHILE
CNV_END CNV=1    ; end pick&place

```

8.14. DO switching on the path(SYN OUT)

8.14.1. Program Example 1 of SYN

Program:

```

LIN P1 FINE Vel=100% Acc=50% TOOL[0] BASE[0]
SYN $DO[1] = TRUE START DELAY = 50 ms
SYN $DO[2] = TRUE END DELAY = -50 ms
LIN P2 FINE Vel=100% Acc=50% TOOL[0] BASE[0]

```

Description:

As shown in below, the command for SYN is given when moving from P1 to P2. P1 and P2 are not in the smooth circumstance. The range of START is from the position of the accurate position for P1 to P2. The range for END is from P2 to P1; the command for START Delay in SYN is given to 50ms, which executes the command for DO[1]=True after the time elapses 50ms. The command for END Delay in SYN is given to -50ms, which backwards 50ms from P2 to execute DO[2]=True.

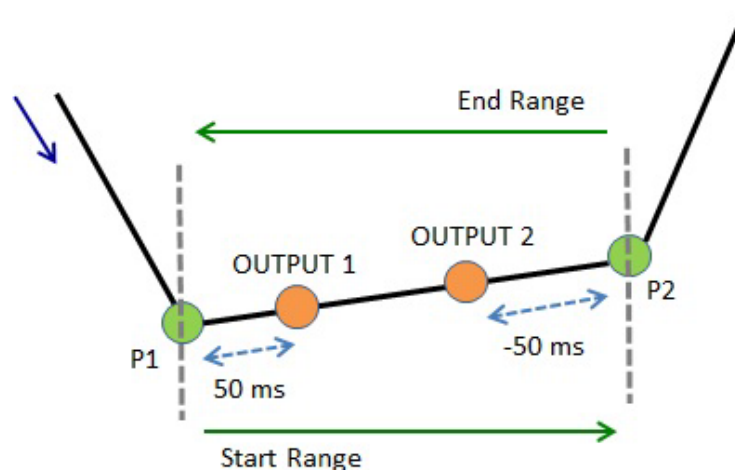


Illustration of Example 1

8.14.2. Program Example 2 of SYN

Program:

```
LIN P1 FINE Vel=100% Acc=50% TOOL[0] BASE[0]
SYN $DO[1] = TRUE START DELAY = 50 ms
SYN $DO[2] = TRUE END DELAY = -50 ms
LIN P2 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
```

Description:

As shown in below, the command for SYN is given when moving from P1 to P2. P2 is in the smooth circumstance. The range of START is from the position of the smooth termination for P1 to the smooth start for P2. The range for END is from the position of the smooth start for P2 to the smooth termination for P2; the command for START Delay in SYN is given to 50ms, which executes the command for DO[1]=True after the time elapses 50ms from the position of the smooth termination for P1. The command for END Delay in SYN is given to -50ms, which executes the command for DO[2]=True after the time elapses 50ms forward from the central point of the Bezier curve in the smooth range of P2. For the description of CONT, please see the Appendix at Chap.8.7.1.

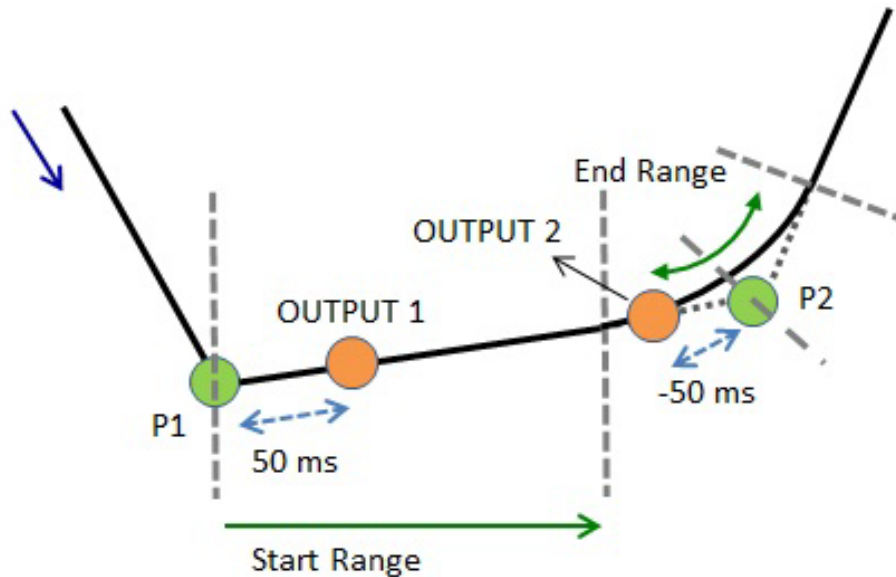


Illustration of Example 2

8.14.3. Program Example 3 of SYN

Program:

```
LIN P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
SYN $DO[1] = TRUE START DELAY = 50 ms
SYN $DO[2] = TRUE END DELAY = -50 ms
LIN P2 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
```

Description:

As shown in below, the command for SYN is given when moving from P1 to P2. P1 and P2 are in the smooth circumstance. The range of START is from the position of the smooth termination for P1 to the smooth start for P2. The range for END is from the position of the smooth start for P2 to the smooth termination for P2; the command for START Delay in SYN is given to 50ms, which executes the command for DO[1]=True after the time elapses 50ms from the position of the smooth termination for P1. The command for END Delay in SYN is given to -50ms, which executes the command for DO[2]=True after the time elapses 50ms forward from the central point of the Bezier curve in the smooth range of P2. For the description of CONT, please see the Appendix at Chap.8.7.1.

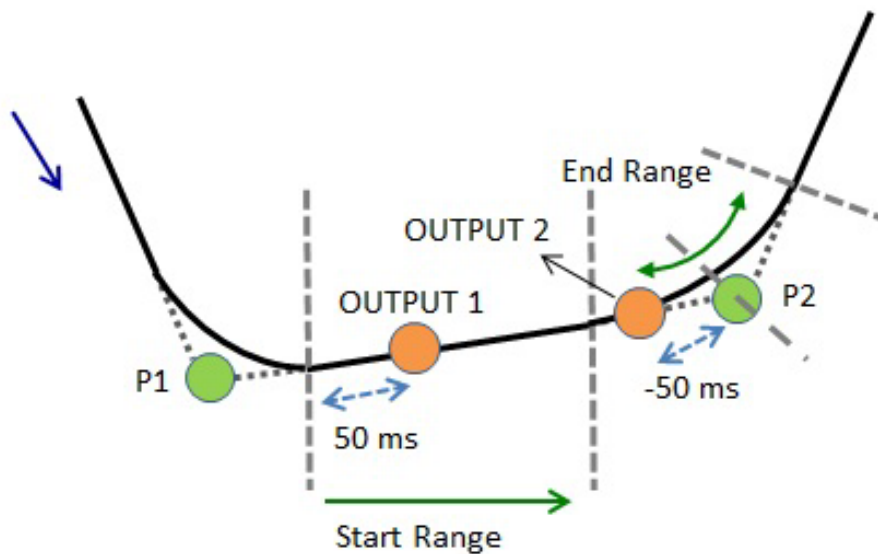


Illustration of Example 3

8.14.4. Program Example 4 of SYN

Program:

```

LIN P1 FINE Vel=100% Acc=50% TOOL[0] BASE[0]
SYN $DO[1] = FALSE START PATH = 50 mm DELAY = -50 ms
LIN P2 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
LIN P3 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
LIN P4 FINE Vel=100% Acc=50% TOOL[0] BASE[0]
    
```

Description:

As shown in below, the command for SYN is given when moving from P1 to P2. The path is used, as well as P2 and P3 are in the smooth circumstance. The range of START is from the position of the smooth start for P1 to P4; PATH=50mm and DELAY = -50ms are in SYN, counting 50mm from the start of P1, moving to 50ms and executing DO[1] = False; if P3 is the accurate position, the range of START is from the smooth start of P1 to P3. For the description of CONT, please see the Appendix at Chap.8.7.1.

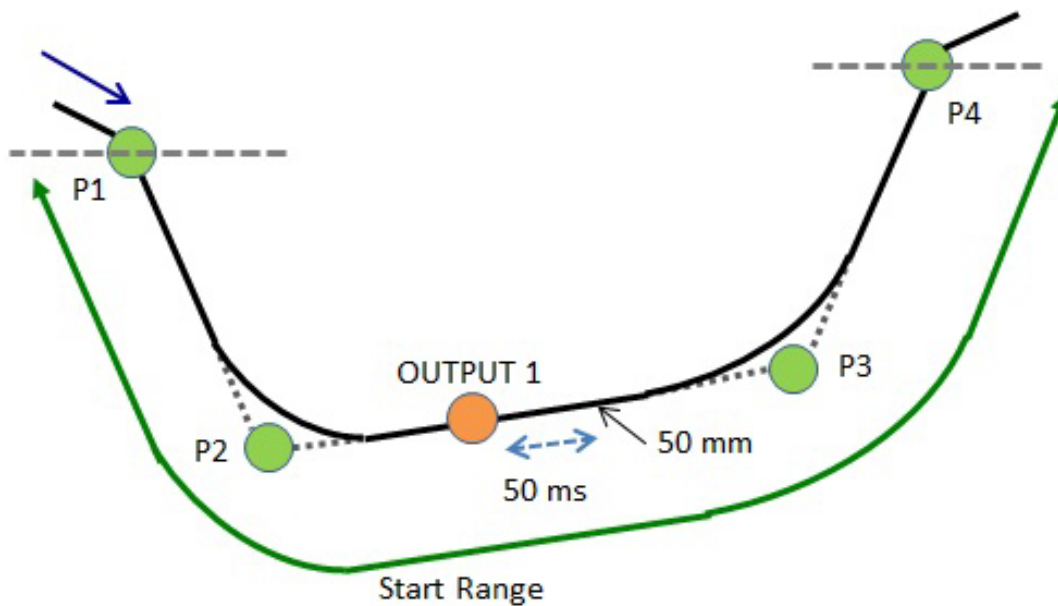


Illustration of Example 4

8.14.5. Program Example 5 of SYN

Program:

```

LIN P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
SYN $DO[1] = FALSE START PATH = 50 mm DELAY = -50 ms
LIN P2 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
LIN P3 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]
LIN P4 FINE Vel=100% Acc=50% TOOL[0] BASE[0]
  
```

Description:

As shown in below, the command for SYN is given when moving from P1 to P2. The path is used, as well as P1, P2 and P3 are in the smooth circumstance. The range of START is from the position of the smooth start for P1 to P4; PATH=50mm and DELAY = -50ms are in SYN, counting 50mm from the smooth start of P1, moving to 50ms and executing DO[1] = False; if P3 is the accurate position, the range of START is from the smooth start of P1 to P3. For the description of CONT, please see the Appendix at Chap.8.7.1.

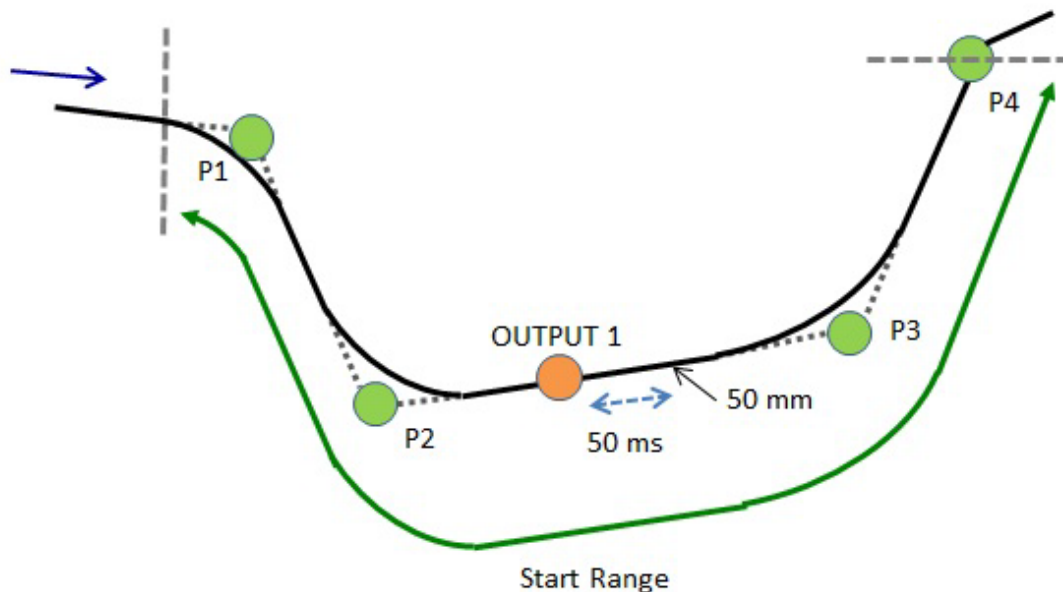


Illustration of Example 5

8.15. Electric Gripper

There is a sample program below. First of all, set the parameter “Wait Idle” to “ON”. This sample will be using all commands of XEG (a kind of electric gripper), including “pick”, “place”, and changing to expert mode to recognize different items by picking status. Users can refer to this sample to develop their own programs.

;Initialize: move the robot to the original position, connect to the XEG, and reset the XEG.

```
PTP P1 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
```

;try to connect to the XEG

```
EG_OPEN(X32)
```

;reset the XEG

```
EG_RESET
```

```
$C[1]=0
```

;The major part of the program: recognize two different objects by pick and place.

```
WHILE $C[1] <= 100
```

```
$C[1] = $C[1]+1
```

;move XEG to a specific position

```
IF EG_GET_STATUS <0 THEN
```

;TO DO the handling commands if XEG gets errors

```
ENDIF
```

```
EG_RUN_MOVE(26.5,80)
```

```
PTP P6 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
```

```
PTP P3 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
```

;execute the picking command

```
IF EG_GET_STATUS <0 THEN
```

;TO DO the handling commands if XEG gets errors

```
ENDIF
```

```
EG_RUN_GRIP(C,25,H,M)
```

;recognize the objects by the position and status of XEG

```
IF SelectObject(EG_GET_POS, EG_GET_STATUS) ==2 THEN
```

```
PTP P6 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
```

```
PTP P2 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
```

```
ENDIF
```

;move XEG to a known position

```
IF EG_GET_STATUS <0 THEN
```

;TO DO the handling commands if XEG gets errors

```
ENDIF
```

```
EG_RUN_MOVE(26.5,80)
PTP P7 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
PTP P8 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
PTP P9 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
PTP P4 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
;change to expert mode to move XEG to pick
IF EG_GET_STATUS <0 THEN
;TO DO the handling commands if XEG gets errors
ENDIF
EG_RUN_EXPERT(C,3.5,60,20.5,20,50)
IF SelectObject(EG_GET_POS, EG_GET_STATUS) ==1 THEN
PTP P9 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
PTP P11 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
PTP P5 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
ENDIF
; move XEG to a known position
IF EG_GET_STATUS <0 THEN
;TO DO the handling commands if XEG gets errors
ENDIF
EG_RUN_MOVE(26.5,80)
PTP P10 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
PTP P5 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
;pick
IF EG_GET_STATUS <0 THEN
;TO DO the handling commands if XEG gets errors
ENDIF
EG_RUN_GRIP(C,25,H,M)
; recognize the objects by the position and status of XEG
IF SelectObject(EG_GET_POS, EG_GET_STATUS) ==1 THEN
PTP P10 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
PTP P8 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
PTP P4 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
ENDIF
; move XEG to a known position
IF EG_GET_STATUS <0 THEN
;TO DO the handling commands if XEG gets errors
ENDIF
EG_RUN_MOVE(26.5,80)
```

```

PTP P8 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
PTP P7 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
PTP P2 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
; change to expert mode to move XEG to pick
IF EG_GET_STATUS <0 THEN
;TO DO the handling commands if XEG gets errors
ENDIF
EG_RUN_EXPERT(C,3.5,60,20.5,20,50)
; recognize the objects by the position and status of XEG
IF SelectObject(EG_GET_POS, EG_GET_STATUS) ==2 THEN
PTP P7 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
PTP P6 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
PTP P3 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
ENDIF
; move XEG to a known position
IF EG_GET_STATUS <0 THEN
;TO DO the handling commands if XEG gets errors
ENDIF
EG_RUN_MOVE(26.5,80)
PTP P6 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
PTP P1 FINE Vel=100% Acc=100% TOOL[0] BASE[0]
ENDWHILE
;disconnect from XEG
EG_CLOSE
;Subprogram: the function to recognize different objects
DEFECT INT SelectObject(POSITION:IN,STATUS:IN)
REAL POSITION
INT STATUS
IF POSITION>=18.5 AND POSITION<=20.5 AND STATUS==2 THEN
RETURN 1
ELSE
IF POSITION>=3 AND POSITION<=4 AND STATUS==2 THEN
RETURN 2
ELSE
RETURN 0
ENDIF
ENDIF
ENDFCT

```

8.16. External Axis

The following is an example program that uses external axis related commands. After the external axis parameters are correctly set, the external axis can be enabled, the axis mode can be switched, the infinite rotation function, and the synchronization coupled motion control can be executed by the relevant commands.

```

; Enable external axis
EX_AX[1] =TRUE
EX_AX[2] =TRUE
EX_AX[3] =TRUE
; Specify the second axis as an asynchronous axis
EX_AX_ASYNC[2] =TRUE
; Move the arm and synchronize the external axis to the specified position
LIN P0 FINE=1 Vel=100mm/s Acc=100% TOOL[1] BASE[1]
; Move the asynchronous external axis to the specified position
ASYPTP{E2 90} FINE=1 Vel=100% Acc=100% TOOL[1] BASE[1]
; Open the synchronous external axis coupled function
EX_AX_SYNC_COUPLE[1] = TRUE
EX_AX_SYNC_COUPLE[3] = TRUE
; The second axis executes an infinite rotation with an asynchronous external axis
EX_AX_CT[2] 100
; Move the arm and synchronize the external axis to the specified position
LIN P1 FINE=1 Vel=100mm/s Acc=100% TOOL[1] BASE[1]

```

8.17. Welding(Optional)

The following is an example program that uses welding and weaving related commands. Welding is performed according to the welding procedure and welding conditions set by the interface when starting the arc. The welding operation is performed according to the set weaving parameters and welding speed. Welding procedures and welding conditions are processed by arc pits and the welding is ended. During the welding process, an error resume function is set. When an alarm occurs between points P0 or a program is manually stopped, the original program can be restarted and moved to At point P0, if an alarm occurs or a program is manually stopped at point P1, the reset program named WELD_RESUME in the same directory will be executed, then the arc will start and move to point P1.

```

; Execute welding according to welding procedure 1 and condition 1
WELD_START WP = 1 WS = 1
; Perform SINE-type swing welding according to swing welding condition 1
WEAVE SINE WS = 1

```

```

; Set resume label
RESUME_PROG
; Line motion, it will perform the above-mentioned weaving motion
LIN P0 CONT WELD_SPEED Acc=100% TOOL[0] BASE[0]
; Perform circular weaving according to weaving instructions and settings
WEAVE CIRCLE Freq = 1Hz Amp = 10mm LD = 0.30s RD = 0.30s
; Set the resume label and specify the reset program
RESUME_PROG = WELD_RESUME
; Line motion, it will perform the above-mentioned weaving motion
LIN P1 CONT WELD_SPEED Acc=100% TOOL[0] BASE[0]
; End weaving
WEAVE_END
; Perform crater processing according to welding procedure 1 and condition 1 to end welding
WELD_END WP = 1 WS = 1
; Clear interrupt reset label
CLEAR_RESUME_PROG

```

8.18. Collision Detection

There is a sample program below. Using command to enable/disable about collision detection after setting parameters of payload correctly. During the execution, it can detect whether a collision has occurred. If it is an expected collision, turn off collision detection and turn it on again after the collision.

```

CD_START 80
;Turn on collision detection, and set the sensitivity to 80
SET_PAYLOAD[1]
LIN P1 FINE=1 Vel=100mm/s Acc=100% TOOL[0] BASE[0]
LIN P2 FINE=1 Vel=100mm/s Acc=100% TOOL[0] BASE[0]
CD_END
;When gripping an object, it is an expected collision, so turn off collision detection
LIN P3 FINE=1 Vel=100mm/s Acc=100% TOOL[0] BASE[0]
$DO[1] = TRUE
SET_PAYLOAD[2]
CD_START 50
;Turn it on again, and set the sensitivity to 50
LIN P4 FINE=1 Vel=100mm/s Acc=100% TOOL[0] BASE[0]

```

8.19. Collision Detection Learning Example

As shown in the sample program in 8.18, we can use learning to obtain a set of suitable sensitivity values.

STEP 0 CD_END and CD_START can be used to select the interval you want to learn;

STEP 1 Switch the mode to T2, AUT or EXT;

STEP 2 Click on the “Learn Start” button on the collision detection interface;

STEP 3 Starts operating;

STEP 4 Click on the “Learn Stop” button on the collision detection interface; assuming that the value learned is 70;

STEP 5 Change both CD_START 80 and CD_START 50 in the program to CD_START 70; if there are multiple CD_START parameter instructions in the same process, you can replace it with instructions with no parameters after the second one in order to reduce the number of inputs.

8.20. Payload

There is a sample program below. Using command to change different payloads after setting parameters of payload correctly.

```
SET_PAYLOAD[1]
;Setting payload 1, the weight is only TOOL self
LIN P1 FINE=1 Vel=100mm/s Acc=100% TOOL[0] BASE[0]
LIN P2 FINE=1 Vel=100mm/s Acc=100% TOOL[0] BASE[0]
$DO[1] = TRUE
;Gripping an object
SET_PAYLOAD[2]
;Setting payload 2, the weight is include TOOL and object
LIN P3 FINE=1 Vel=100mm/s Acc=100% TOOL[0] BASE[0]
LIN P4 FINE=1 Vel=100mm/s Acc=100% TOOL[0] BASE[0]
```

8.21. Interrupt

```
EXT TEST1()
;-----Interrupt declaration-----
INTERRUPT 1 WHEN $DI[1] == TRUE DO TEST1() ; If DI[1] is TRUE trigger
INTERRUPT 2 WHEN $DI[2] == TRUE DO TEST2() ; If DI[2] is TRUE trigger
INTERRUPT 3 WHEN $DI[3] == TRUE DO TEST3() ; If DI[3] is TRUE trigger
INTERRUPT 4 WHEN $DI[4] == TRUE DO TEST4() ; If DI[4] is TRUE trigger
INTERRUPT 5 WHEN $DI[5] == TRUE DO TEST5() ; If DI[5] is TRUE trigger
INTERRUPT 6 WHEN $DI[6] == TRUE DO TEST6() ; If DI[6] is TRUE trigger
INTERRUPT 7 WHEN $DI[7] == TRUE DO TEST7() ; If DI[7] is TRUE trigger
INTERRUPT 8 WHEN $DI[8] == TRUE DO TEST8() ; If DI[8] is TRUE trigger
```

```
;INTERRUPT ON 1 ; ENABLE INTERRUPT
;INTERRUPT ON 2; ENABLE INTERRUPT
;INTERRUPT ON 3; ENABLE INTERRUPT
;INTERRUPT ON 4; ENABLE INTERRUPT
;INTERRUPT ON 5; ENABLE INTERRUPT
;INTERRUPT ON 6; ENABLE INTERRUPT
;INTERRUPT ON 7; ENABLE INTERRUPT
;INTERRUPT ON 8; ENABLE INTERRUPT
```

INTERRUPT ON ALL

```
;INTERRUPT OFF ALL
```

```
;-----Initialize-----
```

```
$C[1] = 0
```

```
$C[2] = 0
```

```
$C[3] = 0
```

```
$C[4] = 0
```

```
$C[5] = 0
```

```
$C[6] = 0
```

```
$C[7] = 0
```

```
$C[8] = 0
```

```
;-----Main Activity-----
```

LOOP

```
PTP P0 CONT Vel=100% Acc=100% TOOL[0] BASE[0]
```

```
PTP P1 CONT Vel=100% Acc=100% TOOL[0] BASE[0]
```

```
ENDLOOP
```

```
;-----
```

```
;-----Interrupt subprogram-----
```

```
DEF TEST2()
```

```
$C[2] = $C[2] + 1 ;Record times of interrupt triggered
```

```
E6POINT P10
```

```
P10 = GETPOINT ; Record position
```

```
PTP P3
```

```
PTP P10 Vel = 50% ; Recover robot position
```

```
END
```

```
DEF TEST3()
```

```
$C[3] = $C[3] + 1 ;Record times of interrupt triggered
```

E6POINT P10

P10 = **GETPOINT** ; Record position

PTP P4

PTP P10 **Vel** = 50% ; Recover robot position

END

DEF TEST4()

\$C[4] = \$C[4] + 1 ;Record times of interrupt triggered

E6POINT P10

P10 = **GETPOINT** ; Record position

PTP P5

PTP P10 **Vel** = 50% ; Recover robot position

END

DEF TEST5()

\$C[5] = \$C[5] + 1 ;Record times of interrupt triggered

E6POINT P10

P10 = **GETPOINT** ; Record position

PTP P6

PTP P10 **Vel** = 50% ; Recover robot position

END

DEF TEST6()

\$C[6] = \$C[6] + 1 ;Record times of interrupt triggered

E6POINT P10

P10 = **GETPOINT** ; Record position

PTP P7

PTP P10 **Vel** = 50% ; Recover robot position

END

DEF TEST7()

\$C[7] = \$C[7] + 1 ;Record times of interrupt triggered

E6POINT P10

P10 = **GETPOINT** ; Record position

PTP P8

PTP P10 **Vel** = 50% ; Recover robot position

END

DEF TEST8()

```
$C[8] = $C[8] + 1 ;Record times of interrupt triggered  
E6POINT P10  
P10 = GETPOINT ; Record position  
PTP P9  
PTP P10 Vel = 50% ; Recover robot position  
END
```

8.22. Array

The following is a sample program that uses array related instructions

```
E6POS P_ARR[3,3]  
INT i,j  
FOR i = 1 TO 3 STEP  
FOR j = 1 TO 3 STEP  
P_ARR[i,j].x = i * 50  
P_ARR[i,j].y = 300-j*20  
P_ARR[i,j].z = 250  
ENDFOR  
ENDFOR
```

8.23. PULSE

```
$DO[1] = PULSE(10)
```

Description: The 1st digital output signal is a 10ms pulse signal.

8.24. TIMEOUT

```
LOOP  
WAIT FOR $DI[1] == TURE TIMEOUT(10,ERROR)  
$C[1] = $C[1] + 1  
ENDLOOP
```

```
ERROR:  
QUIT
```

8.25. P#

```
INT i  
FOR i = 1 TO 1000 STEP  
IF P#_EXIST(i) THEN  
PTP P#(i)  
ENDIF  
ENDFOR
```

9. Appendix

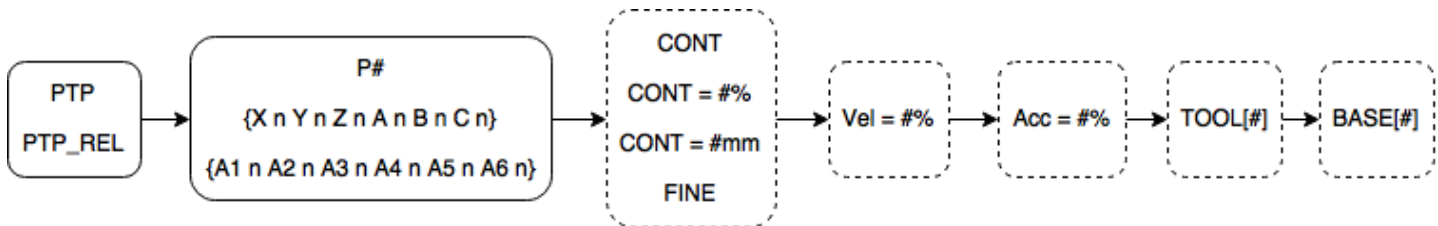
9.1. Software commands

Complete instruction, Please refer to 「ROBOT Software Programming Instruction Set Manual」.

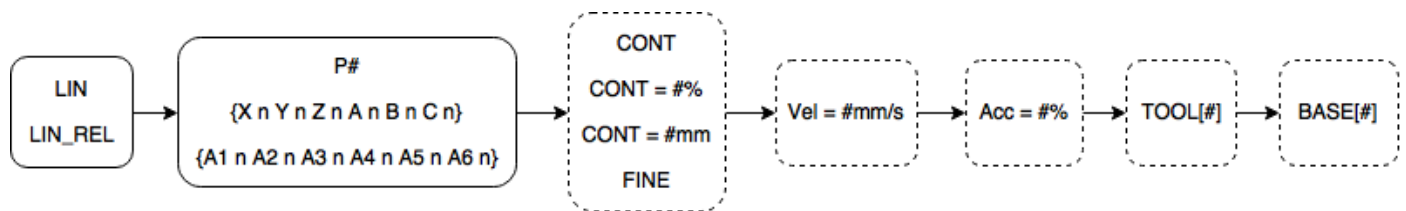
9.1.1. Motion commands

Commands	Description
PTP	Point to point motion
PTP_REL	Point to point relative motion
LIN	Linear motion
LIN_REL	Linear relative motion
LIN_REL_TOOL	The relative movement of the straight line in the tool coordinate system
CIRC	Circular motion
CIRC_REL	Circular relative motion
SPLINE...SPL...ENDSPLINE	B-Spline curvilinear motion

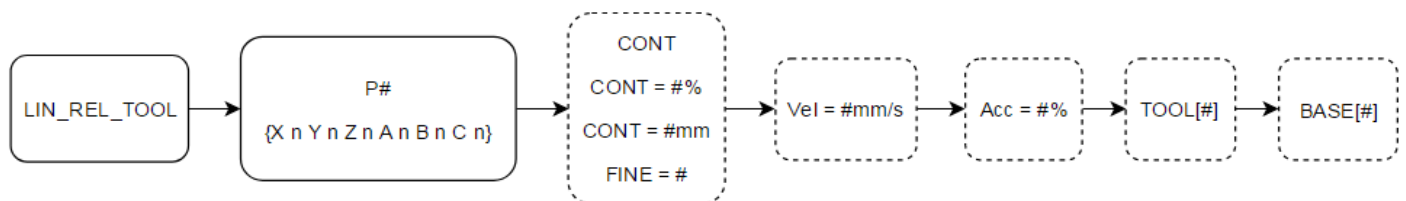
PTP&PTP_REL flowchart:



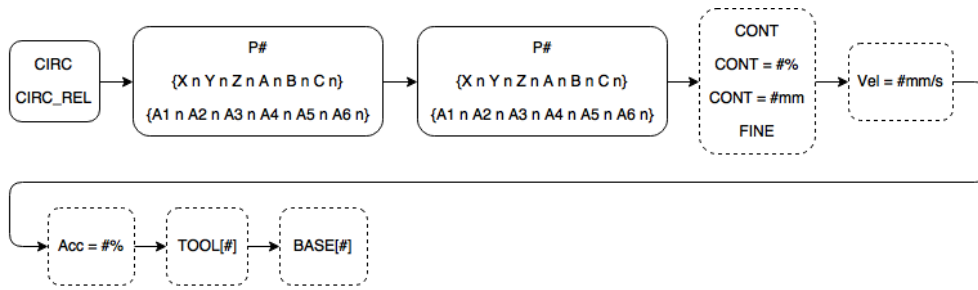
LIN&LIN_REL flowchart:



LIN_REL_TOOL flowchart:



CIRC&CIRC_REL flowchart:



SPLINE Instructions :

SPLINE

SPL P1

.....

SPL P73

ENDSPLINE

9.1.2. RS232 or EtherNet Commands

Commands	Description
COPEN	Open RS232 or EtherNet
CCLOSE	Close RS232 or EtherNet
CCLEAR	Delete RS232 or EtherNet data
CREAD	Read received data from RS232 or EtherNet
CWRITE	Write RS232 or EtherNet data
CINQUIRE	Inquire RS232 or EtherNet package numbers

Example:

INT HANDLE

INT NUM

REAL SERDATA

STRING SER_STR

COPEN (SER , HANDLE)

LOOP

IF HANDLE > -1 THEN

CINQUIRE(HANDLE,NUM)

If NUM>0 THEN

CREAD (HANDLE,SER_STR ,SERDATA)

ENDIF

CCLEAR (HANDLE)

```
SERDATA = SERDATA + 1  
SER_STR = "RETURN:"  
CWRITE (HANDLE, SER_STR, SERDATA)  
ENDIF  
WAIT SEC 0.3  
ENDLOOP  
CCLOSE (HANDLE)
```

9.1.3. Conveyor tracking commands

Commands	Description
CNV_START	Startup the tracking procedure of conveyor, and connect with the dispatching system/vision
CNV_END	End the tracking of conveyor and connection of dispatching system/vision
CNV_PICK	Flying-pick to pick object
CNV_PLACE	Flying-place to place object
CNV_SET_DELAY_TIME[#]	Configure the ending time of tracking delay for the flying-pick/flying-place
CNV_QUEUE_REMOVE[#]	Remove the forefront queue of flying-pick/flying-place temporary storage
CNV_PICK_ACC[#]	Configure the acceleration time of tracking push-down
CNV_EMPTY	If the pick quantity is zero
CNV_FULL	If the pick quantity is up to the upper limit
CNV_OBJECT	The numbering of latest object picked currently
CNV_PICK_QUANTITY	The maximum quantity able to pick
CNV_QUEUE_SIZE[#]	The quantity of already sensed but not pick yet
CNV_TRIGGER_TIMES	Configure the triggering times of sensor for adding one working task
CNV_OFFSET_X	X Offset value of flying-pick/flying-place
CNV_OFFSET_Y	Y Offset value of flying-pick/flying-place
CNV_OFFSET_Z	Z Offset value of flying-pick/flying-place
CNV_PLACE_BATCH	Configure the maximum times for flying-place
CNV_OBJ_CNT_DIST[#]	Position difference between the first object and second object
CNV_RESET_ENC	Clear the count value of external encoder
CNV_SPEED[#]	Read the speed of specified conveyor

Example: Use vision to collaborate with flying-pick

```

CNV_START CNV=1           ;start pick & place
CNV_SET_DELAY_TIME[1] = 50 ; delay 50ms and leave flying-pick/flying-place
CNV_PICK_ACC[1] = 50     ; push-down acceleration of flying-pick is 50ms
CNV_PICK_QUANTITY = 2   ; set the maximum quantity to pick object
WHILE CNV_FULL == FALSE ; go to loop when the quantity on the robot doesn't reach the upper limit
CNV_PICK CNV=1 OBJ=1 $DO[1] Down=5.000mm FINE Vel=2000mm/s Acc=50% TOOL[0] BASE[0] ; execute
pick
ENDWHILE
IF CNV_OBJECT == 1 THEN ; if the object numbering is 1

```

```

CNV_OFFSET_X = 10           ; configure X Offset value of flying-pick/flying-place as 10
CNV_OFFSET_Y = 10           ; configure Y Offset value of flying-pick/flying-place as 10
CNV_OFFSET_Z = 10           ; configure Z Offset value of flying-pick/flying-place as 10
ENDIF
WHILE CNV_EMPTY == FALSE ; go to loop when the quantity on the conveyor is not empty
CNV_PLACE $DO[1] P2 FINE Vel=2000mm/s Acc=50% TOOL[0] BASE[0]
    ; execute place
ENDWHILE
CNV_END CNV=1               ; end pick & place

```

Example : Use sensor to collaborate with flying-pick

```

INT ISpeed
ISpeed = CNV_SPEED[1]       ; read the speed of conveyor 1
CNV_START CNV=1             ; start pick & place
CNV_RESET_ENC               ; clear the count value of external encoder
CNV_TRIGGER_TIMES = 1       ; sensor is triggered once, add working task once
CNV_PLACE_BATCH = 1         ; allowable number of time for flying-place in one working task is one time
CNV_PICK_QUANTITY = 2       ; set the maximum quantity to pick object
WHILE CNV_FULL == FALSE    ; go to loop when the quantity on the robot doesn't reach the upper limit
CNV_PICK CNV=1 $DO[1] P0 Down=5.000mm FINE Vel=2000mm/s Acc=50% TOOL[0] BASE[0] ; execute pick
ENDWHILE
IF CNV_OBJECT == 1 THEN     ; if the object numbering is 1
CNV_QUEUE_REMOVE[1]         ; remove the first queue
ENDIF

PTP P1 CONT Vel=100% Acc=50% TOOL[0] BASE[0] ;move to P1
IF CNV_QUEUE_SIZE[1] > 1 THEN ; determine if the queue content is more than 1
IF CNV_OBJ_CNT_DIST[1] > 2600 THEN; determine if the difference value is more than 2600 ea counts
WHILE CNV_EMPTY == FALSE ; go to loop when the quantity on the conveyor is not empty
CNV_PLACE CNV=2 $DO[1] P2 FINE Vel=2000mm/s Acc=50% TOOL[0] BASE[0] ;execute place
ENDWHILE
ENDIF
ENDIF
CNV_END CNV=1               ;end pick & place

```

9.1.4. Register commands

Commands	Description
\$C[#]	Counter register
\$DI[#]	Digital input point register
\$DO[#]	Digital output point register
\$PR[#]	Robot input point register
\$RI[#]	Robot output point register
\$RO[#]	Timer register
\$T[#]	Start timer register
\$T_STOP[#]	Valve output register
\$VO[#]	Counter register

Example:

```
$C[1] = 0
$DO[1] = TRUE
WAIT FOR $DI[1] == TRUE
$RO[1] = TRUE
WAIT FOR $RI[1] == TRUE
$VO[1] = TRUE
$T_STOP[1] = TRUE
$T[1] = 0
```

PR Example 1:

```
$PR[1] = {A1 1 , A2 2 , A3 3 , A4 4 ,A5 5 , A6 6}
$PR[2] = {X 7 ,Y 8 , Z 9 , A 10 ,B 11 , C 12}
$PR[3] = {A1 1 , A2 2 , A3 3 , A4 4 ,A5 5 , A6 6 , X 7 ,Y 8 , Z 9 , A 0,B 0 , C 0}
```

PR Example 2:

```
E6POS A = {X 10 ,Y 10 ,Z 10 ,A 10 ,B 10 ,C 10}
E6AXIS B = {A1 20 , A2 20 , A3 20 , A4 50 ,A5 10 , A6 20}
E6POINT C = {A1 20 , A2 20 , A3 20 , A4 50 ,A5 10 , A6 20 ,X 10 ,Y 10 ,Z 10 ,A 10 ,B 10 ,C 10}
$PR[1] = A
$PR[2] = B
$PR[3] = C
```

PR Example 3:

```
$PR[1] = GETPOINT
```

9.1.5. Variable types

Commands	Description
BOOL	Boolean variable type
CHAR	Character variable type
E6AXIS	Angular variable value type
E6POINT	Coordinates or angular variable type
E6POS	Coordinates variable type
FRAME	BASE or TOOL coordinate system
INT	Integer variable type
REAL	Real point variable type

Example:

BOOL K = TRUE

CHAR COLOR = 'R'

INT I = 0

REAL R = 0

FRAME :

FRAME POINT = {A1 90}

E6POS/E6AXIS :

E6POS POINT = {X 0,Y 300,Z 200}

E6AXIS POINT = {A1 90}

PTP POINT **CONT**=100% **Vel**=100% **Acc**=50% **TOOL**[0] **BASE**[0]

E6POINT :

E6POINT HOME = {Y 200,Z -1000,A 90}

or

E6POINT HOME = {A1 90}

PTP HOME **CONT**=100% **Vel**=100% **Acc**=50% **TOOL**[0] **BASE**[0]

9.1.6. Math Calculation

Commands	Description
ACOS	Arc cosine(X)
ASIN	Arc sine(X)
ATAN	Arc tangent(X)
ATAN2	Arc tangent(X, Y)
COS	Cosine(X)
SIN	Sine(X)
TAN	Tangent(X)

Example:

REAL TESTA

TESTA=ACOS(0)

TESTA=ASIN(0)

TESTA=ATAN(0)

TESTA=ATAN2(0,1)

TESTA=COS(0)

TESTA=SIN(0)

TESTA=TAN(0)

9.1.7. Control function

Commands	Description
FOR...ENDFOR	For loop
GOTO	Go to label position
IF...ENDIF	IF statement
LOOP...ENDLOOP	LOOP
REPEAT...UNTIL	Repeat loop
SWITCH...ENDSWITCH	Switch statement
WHILE...ENDWHILE	While loop

Example:

FOR...ENDFOR :

INT n

FOR n = 0 TO 2 STEP 1

PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

ENDFOR

GOTO :

FOUND:

PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

GOTO FOUND

IF...ENDIF :

INT n = 1

IF n > 0 THEN

PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

ENDIF

LOOP...ENDLOOP :

LOOP

PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

ENDLOOP

REPEAT...UNTIL :

INT n = 0

REPEAT

PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

n = n + 1

UNTIL n > 2

SWITCH...ENDSWITCH :

INT n = 0

LOOP

SWITCH n

CASE 0

PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

CASE 1

PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

CASE 2

EXIT

ENDSWITCH

n = n + 1

ENDLOOP

WHILE...ENDWHILE :

INT n = 2

WHILE n > 0

PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0]

n = n - 1

ENDWHILE

9.1.8. Gripper commands

Commands	Description	Example
EG_OPEN	Connect with XEG series electric gripper	EG_OPEN(Type)
EG_CLOSE	Disconnect current XEG series electric gripper connection	EG_CLOSE
EG_RESET	Reset XEG series electric gripper	EG_RESET
EG_GET_STATUS	Get XEG series electric gripper status	IF EG_GET_STATUS == 2 THEN ... ENDIF
EG_RUN_MOVE	Move XEG series electric gripper	EG_RUN_MOVE(10,20)
EG_RUN_GRIP	Grip action of XEG series electric gripper	EG_RUN_GRIP(C,5,L,M)
EG_RUN_EXPERT	Grip action and movement of XEG series electric gripper	EG_RUN_EXPERT(C,10,20,5,10,100)
EG_GET_POS	Get XEG series electric gripper position	IF EG_GET_POS > 5.00 THEN ... ENDIF

9.1.9. External Axis Instruction

Commands	Description	Example
EX_AX	Turns the connected external axis on or off	EX_AX[1] = TRUE ... EX_AX[1] = FALSE
EX_AX_ASYNC	Sets the external axis as synchronous axis or asynchronous axis	EX_AX_ASYNC[1] = TRUE EX_AX_ASYNC[2] = FALSE
EX_AX_SET_ACC	Sets the acceleration and deceleration time for the external axis	EX_AX_SET_ACC[1] 250
EX_AX_SET_SPEED	Sets the speed of the specified external axis	EX_AX_SET_SPEED[1] 1000
EX_AX_CT	Executes infinite rotation for the external axis	EX_AX_CT[1] 100
ASYPTP	Performs point to point movement control for the asynchronous external axis	ASYPTP{E1 60, E2 100} CONT=100% Vel=100% Acc=50%
PTP	Point-to-point Move Command	E6POINT P1{A1 50, E1 60} PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0]BASE[0] PTP{A1 40, E1 50} CONT=100% Vel=100% Acc=50% TOOL[0]BASE[0]

Commands	Description	Example
		PTP{X 100, E1 50} CONT=100% Vel=100% Acc=50% TOOL[0]BASE[0]
LIN	Linear Move Command	E6POINT P1{A1 50, E1 60} LIN P1 CONT=100% Vel=2000mm/s Acc=50% TOOL[0]BASE[0] LIN{A1 40, E1 50} CONT=100% Vel=2000mm/s Acc=50% TOOL[0]BASE[0] LIN{X 100, E1 50} CONT=100% Vel=2000mm/s Acc=50% TOOL[0]BASE[0]
CIRC	Circular Arc Move Command	E6POINT P1{A1 50, E1 60} E6POINT P2{X 100, E1 0} CIRC P1 P2 CONT=100% Vel=2000mm/s Acc=50% TOOL[0]BASE[0] CIRC{A1 50, E1 60} {X 100, E1 0}CONT=100% Vel=2000mm/s Acc=50% TOOL[0]BASE[0] CIRC{A1 50, E1 60} {X 100, E1 0} CONT=100% Vel=2000mm/s Acc=50% TOOL[0]BASE[0]

9.1.10. Welding Instruction

Commands	Description	Example
WELD_START	Performs welding according to the processes and conditions specified	WELD_START WP = 1 WS = 1 LIN P0 WELD_SPEED
WELD_END	Performs arc processing and ends welding according to the processes and conditions specified.	LIN P1 WELD_SPEED WELD_END WP = 1 WS = 1
WEAVE	Performs weave welding actions according to the weave welding conditions set	WEAVE SINE WS = 1 LIN P0 CONT WELD_SPEED Acc=100% TOOL[0] BASE[0]

Commands	Description	Example
WEAVE_END	End weave welding action	WEAVE CIRCLE Freq = 1Hz Amp = 10mm LD = 0.30s RD = 0.30s LIN P1 CONT WELD_SPEED Acc=100% TOOL[0] BASE[0] WEAVE_END
RESUME_PROG	Sets the start label of the range in which the interrupted recovery program is to be executed	RESUME_PROG WELD_START WP = 1 WS = 1 LIN P0 CONT WELD_SPEED Acc=100%
CLEAR_RESUME_PROG	Clears the range label of the interrupted reset program	TOOL[0] BASE[0] RESUME_PROG = Test LIN P1 CONT WELD_SPEED Acc=100% TOOL[0] BASE[0] CLEAR_RESUME_PROG WELD_END WP = 1 WS = 1

9.1.11. Collision Detection Instruction

Commands	Description	Example
CD_START	Enables collision detection and sets the sensitivity	CD_START CD_START 100
CD_END	Disables collision detection	CD_END

9.1.12. Payload Setting Instruction

Commands	Description	Example
SET_PAYLOAD	Set End Payload Weight Data	SET_PAYLOAD[1]

9.1.13. String Function Instructions

Commands	Description	Example
STRCMP	String comparison	flag = STRCMP(str,"HIWIN")
STRPOS	Search strings	num = STRPOS(str,"HIWIN")
STRCPY	String copy	STRCPY(str1,str2,1,255)
STRLEN	Get string length	num = STRLEN(str)
INTTOSTR	Convert INT to string	str = INTTOSTR(1)
REALTOSTR	Convert REAL to string	str = REALTOSTR(1.5)
STRTOINT	Convert string to INT	num = STRTOINT(str)
STRTOREAL	Convert string to REAL	num = STRTOREAL(str)

9.1.14. Interruption Process Instruction

Commands	Description	Example
INTERRUPT...WHEN...DO...	Interruption Condition Command	INTERRUPT 1 WHEN \$DI[1] == TRUE DO PROG1()

Commands	Description	Example
INTERRUPT ON ...	Enable Interruption Command	INTERRUPT ON 1 INTERRUPT ON ALL
INTERRUPT OFF ...	Stop Interruption Command	INTERRUPT OFF1 INTERRUPT OFF ALL

9.1.15. Other commands

Commands	Description	Example
ADDTOOL	Add tool	ADDTOOL ee
ADDOBJECT	Add object	ADDTOOL table P:500,200 C:200,50
SET_TOOL	Set tool coordinate system	FRAME T_ONE T_ONE.X = 100 SET_TOOL 1 SET_TOOL T_ONE
SET_BASE	Set base coordinate system	FRAME B_ONE B_ONE.Y = 100 SET_BASE 1 SET_BASE B_ONE
SET_OVERRIDE_SPEED	Set override speed ratio	SET_OVERRIDE_SPEED 100
SET_SPEED	Set line speed	SET_SPEED 2000
SET_ROTATION_SPEED	Sets the posture rotation speed, unit deg/sec	SET_ROTATION_SPEED 100
SET_ACC	Set acceleration	SET_ACC 250
TRUE_PATH	Open or close trajectory accuracy control	TRUE_PATH = TRUE
USER_ALARM	Configure user alarm	USER_ALARM[1]
SYN	Synchronous switch O pint in motion path	LIN P1 FINE Vel=100% Acc=50% TOOL[0] BASE[0] SYN \$DO[1] = TRUE START DELAY = 50 ms SYN \$DO[2] = TRUE END DELAY = -50 ms LIN P2 FINE Vel=100% Acc=50% TOOL[0] BASE[0]
MOVEFLOOR	Move floor position	MOVEFLOOR 100
DEFFCT...ENDFCT	Define subprogram	PTP P0 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0] MY() DEFFCT INT MY() PTP P1 CONT=100% Vel=100% Acc=50% TOOL[0] BASE[0] RETURN 100 ENDFCT
GETPOINT	Get coordinates or angular value	E6POINT E6TEST

Commands	Description	Example
		E6TEST = GETPOINT
AXISON	Axis coordinates on	AXISON
AXISOFF	Axis coordinates off	AXISOFF
GET_MOTION_STATUS	Get motion status	INT Istatus Istatus = GET_MOTION_STATUS
WAIT SEC	Wait second	WAIT SEC 10
WAIT FOR \$DI[#]	Wait digital input	WAIT FOR \$DI[1] == TRUE
STRUC	Define structure	STRUC CASTING_TYPE INT MASS, REAL VOLUME
CHECK_LIN	Confirms whether there are singular points between the two points	IF CHECK_LIN(P0,P1) == TRUE THEN LIN P1 ELSE LIN P2 ENDIF
FWD	Forward kinematics conversion function	E6AXIS P1_AXIS = {A1 0, A3 0, A4 0, A5 -90, A6 0} E6POS P1_POS P1_POS = FWD(P1_AXIS)
STOP_MOTION	Specify the triggering of DI, SI and WI to stop the movement of the previous line and move in the opposite direction until the specified DI, SI and WI are OFF.	LIN P0 CONT Vel=2000mm/s Acc=100% TOOL[0] BASE[0] STOP_MOTION \$DI[1] LIN P1 CONT Vel=2000mm/s Acc=100% TOOL[0] BASE[0]
SERVO_ON	All axes excitation instruction (currently only open for axis 1 and 2 of the SCARA series)	SERVO_ON
SERVO_OFF	All axes excitation instruction (currently only open for axis 1 and 2 of the SCARA series)	SERVO_OFF
SERVO_ON(#)	Single axis excitation instruction (currently only open for axis 1 and 2 of the SCARA series)	SERVO_ON(1) SERVO_ON(2)
SERVO_OFF(#)	Single axis de-excitation instruction (currently only open for axis 1 and 2 of the SCARA series)	SERVO_OFF(1) SERVO_OFF(2)

This manual does not contain all instructions; if you need the complete instruction set, please refer to the “Robot software instruction set manual” on our official website.

Robot System Software-HRSS 3.3 (Original Instruction) User Manual

Publication Date : April 2025

-
1. HIWIN is a registered trademark of HIWIN Technologies Corp.. For your protection, avoid buying counterfeit products from unknown sources.
 2. Actual products may differ from specifications and photos provided in this catalog. These differences may be the result of various factors including product improvements.
 3. HIWIN website for patented product directory: http://www.hiwin.tw/Products/Products_patents.aspx
 4. HIWIN will not sell or export products or processes restricted under the "Foreign Trade Act" or related regulations. Export of restricted products should be approved by proper authorities in accordance with relevant laws and shall not be used to manufacture or develop nuclear, biochemical, missiles or other weapons.



Global Sales and Customer Service Site

HIWIN GmbH

OFFENBURG, GERMANY
www.hiwin.de
www.hiwin.eu
info@hiwin.de

HIWIN USA

CHICAGO, U.S.A.
www.hiwin.com
info@hiwin.com

HIWIN JAPAN

KOBE · NAGOYA · TOKYO · TOHOKU ·
NAGANO · SHIZUOKA · HOKURIKU ·
HIROSHIMA · FUKUOKA · KUMAMOTO,
JAPAN
www.hiwin.co.jp
info@hiwin.co.jp

HIWIN Srl

BRUGHERIO, ITALY
www.hiwin.it
info@hiwin.it

HIWIN Schweiz GmbH

JONA, SWITZERLAND
www.hiwin.ch
info@hiwin.ch

HIWIN s.r.o.

BRNO, CZECH REPUBLIC
www.hiwin.cz
info@hiwin.cz

HIWIN FRANCE

STRASBOURG, FRANCE
www.hiwin.fr
info@hiwin.de

HIWIN SINGAPORE

SINGAPORE
www.hiwin.sg
info@hiwin.sg

HIWIN KOREA

SUWON · CHANGWON, KOREA
www.hiwin.kr
info@hiwin.kr

HIWIN CHINA

SUZHOU, CHINA
www.hiwin.cn
info@hiwin.cn

HIWIN BULGARIA

SOFIA, BULGARIEN
www.hiwin.bg
info@hiwin.bg

HIWIN TECHNOLOGIES CORP.

No. 7, Jingke Road,
Taichung Precision Machinery Park,
Taichung 408208, Taiwan
Tel: +886-4-23594510
Fax: +886-4-23594420
www.hiwin.tw
www.hiwinsupport.com
business@hiwin.tw (Sales)
robotservice@hiwin.tw (Customer Service)