

R8.3 USER MANUAL

Original User Manual

This manual provides the information necessary to properly install, operate and service Wittmann Robot Systems, and ensure correct and safe operation of our equipment.

Our robots must be installed and operated in accordance with these instructions by authorized and qualified personnel only.



Improper installation startup and operation may result in injury to personnel and/or damaged equipment.

Before installation, please read this manual completely, paying special attention to the

Safety Warnings marked .

To ensure optimal use of our equipment, we recommend that you attend the training courses regularly offered at our Service Centers.



Proper function of the equipment can be ensured only if maintenance is performed according to the instructions given in this manual.

The descriptions in this manual refer to the robot and equipment according to the specification sheet. These specifications are subject to change without notice.

This manual is for the sole use of the user of the robot and may not be distributed to unauthorized persons. No part of this manual may be copied or reproduced without our written consent.



Use original Wittmann USB-sticks only. The function with USB-sticks of third party suppliers is explicitly not guaranteed.



Do not use hard or pointed objects to operate the touch screen of the Teachbox. This could damage the touch screen.

The noise emission of this device does not exceed 70 dB(A).



Before starting up the robot for the first time, please review this manual thoroughly and familiarize yourself with the operation of the robot. Improper use may injure personnel and/or damage the robot, mold or molding machine.

Table of Contents

1 SAFETY	11
1.1 SAFETY REGULATIONS	11
1.2 SAFETY FEATURES.....	11
1.2.1 Safeguarding.....	11
1.2.2 Emergency STOP Buttons	11
1.2.3 Permit Key on Teachbox	11
1.3 LIMITATIONS	12
1.4 CAPABILITY OF THE SAFETY CONTROL.....	12
1.5 CONTROL OF HAZARDOUS ENERGY – (LOCKOUT / TAGOUT)	13
1.5.1 Glossary.....	13
1.5.2 Lockout/Tagout Kit	14
1.5.3 Lockout/Tagout Procedure	14
1.5.4 Removal of LOTO	15
2 INSTALLATION.....	16
2.1 MECHANICAL INSTALLATION.....	16
2.1.1 Dimensional Drawing – Adapter Unit.....	16
2.1.2 Mounting of the Robot	16
2.1.3 Demounting of the Robot.....	17
2.2 PNEUMATICS	18
2.2.1 Compressed Air Connection.....	18
2.3 ELECTRICAL CONNECTIONS.....	19
2.3.1 Mains Power Supply.....	19
2.3.1.1 Fault Current Safety Switches.....	19
2.3.2 Interface with the Injection Molding Machine	20
2.3.3 Mold Monitoring S5/S6	21
2.3.3.1 Settings for Virtual Roller Switches (Mold Safety Switches)	22
2.3.3.2 Initialize Roller Switch Monitoring.....	23
2.3.3.3 Test Procedure for Mold Safety S5 / S6	24
2.4 SAFETY PACKAGE/SAFEGUARDING	25
2.4.1 Safety Package	25
2.4.2 EXTERNAL Safety Package for Safety Door or Access Hatch	26
2.4.3 Safety Guarding	28
2.5 ELECTRICAL INTERFACE.....	29
2.5.1 Euromap 67 interface	29
2.5.1.1 Injection Molding Machine (IMM) Signals.....	30
2.5.1.2 Robot Signals	31
2.5.2 Euromap 12 Interface.....	33
2.5.2.1 Injection Molding Machine (IMM) Signals.....	34

2.5.2.2 Robot Signals	35
3 OPERATING INSTRUCTIONS FOR R8.3 CONTROL	36
3.1 BASIC OPERATION	36
3.1.1 Introduction	36
3.1.2 Robot Axes.....	37
3.1.3 Emergency Stop Block	38
3.1.4 Numerical Input	39
3.1.5 Text Input	40
3.1.6 Operating Modes.....	41
3.1.7 Operating Mode Reference	42
3.1.7.1 Reference travel in the operating mode Reference Missing	42
3.1.7.2 Reference travel in the operating mode Manual.....	43
3.1.8 Override	44
3.1.9 ECO-Mode	46
3.1.10 Safety System	47
3.1.10.1 Startup with Safety System	47
3.1.10.1.1 Manual Setup Mode.....	47
3.1.10.1.2 Automatic Mode, Step Mode, Reference Travel.....	47
3.1.10.2 Selection of Operating Modes	47
3.1.10.2.1 Manual Setup Mode.....	47
3.1.10.2.2 Automatic Mode, Step Mode, Reference.....	47
3.1.10.3 Monitoring of the Safety Gate.....	48
3.1.10.4 Error Effect Analysis	48
3.1.11 User Administration and Passwords	49
3.1.11.1 Administrator.....	49
3.1.11.2 Users	50
3.1.11.2.1 User (create, change)	51
3.1.11.3 Profiles.....	52
3.1.11.4 Login.....	53
3.1.11.5 Logout.....	54
3.1.11.6 Forgot Password.....	54
3.1.12 Manual Functions	55
3.1.12.1 Numerical Axes.....	55
3.1.12.2 Axis Release in Manual Mode (optional)	57
3.1.12.3 Pneumatic Axes	58
3.1.12.4 Digital (B-) Axes	59
3.1.12.5 Counters	60
3.1.12.6 Placing Counters.....	61
3.1.12.7 Conveyors.....	62
3.1.12.8 Peripheral Outputs	63
3.1.12.9 Vacuums.....	64
3.1.12.10 Grippers	65
3.1.12.11 Cylinders.....	66
3.1.13 FILE Menu of the main screen.....	67
3.1.13.1 Properties	67
3.1.13.2 Backup stored Teachprograms	68
3.1.14 INIT Menu	69
3.1.14.1 Stroke Limits	69
3.1.14.2 Roller Switches	70
3.1.14.3 Absolute Encoders (Admin only)	70
3.1.14.4 Deactivate Monitoring	71
3.1.14.5 Maintenance - Lubrication Interval	72
3.1.14.6 Setup: Robot Settings	73

3.1.14.7 Setup: Date / Time	75
3.1.14.8 Setup: Language.....	75
3.1.14.9 Setup: Network Settings (Admin only).....	76
3.1.14.10 Setup: Remote Users Settings (Admin only)	77
3.1.14.10.1 Create / Edit Remote Users (Admin only).....	78
3.1.14.11 Licenses (Admin only).....	78
3.1.14.12 Setup: Email Settings.....	79
3.1.14.12.1 Tab Basic.....	79
3.1.14.12.2 Tab Content.....	79
3.1.14.12.3 Tab Timeouts.....	80
3.1.14.12.4 Tab Recipients.....	80
3.1.14.12.5 Tab Events	81
3.1.14.13 Setup: Screensaver Properties	82
3.1.14.14 Setup: Calibrate Touch	83
3.1.15 View Menu	84
3.1.15.1 Operating Systems	84
3.1.15.2 Operating Data	85
3.1.15.3 Axes positions.....	86
3.1.15.4 I / O.....	87
3.1.15.5 Safety Areas	88
3.1.15.6 Counters	89
3.1.15.7 Placing Counters.....	90
3.1.15.8 Conveyors.....	91
3.1.15.9 Cylinders.....	91
3.1.15.10 POs - Peripheral Outputs	92
3.1.15.11 PIs - Peripheral Inputs.....	92
3.1.15.12 IMM I / O.....	93
3.1.15.13 AutoSwitches	94
3.1.15.14 Stopwatches	94
3.1.15.15 Vacuums.....	95
3.1.15.16 Default Vacuum Level.....	95
3.1.15.17 Grippers	96
3.1.15.18 Alarms	97
3.1.15.19 Error Buffer	98
3.1.15.19.1 Export the Error Buffer.....	99
3.1.15.20 Command Buffer.....	101
3.1.15.20.1 Export the Command Buffer.....	102
3.1.15.21 Virtual Subpendant (Emergency Stop Block)	104
3.1.15.21.1 Tab Standard.....	104
3.1.15.21.2 Tab Easy	105
3.1.15.22 Memory State.....	106
3.1.16 System Report.....	107
3.1.17 Email Functions.....	108
3.1.17.1 Manual Email Function.....	108
3.1.17.2 Automatic Email Function	109
3.1.18 RFID / Digital EOAT (optional).....	110
3.1.18.1 Link a Teach Program with an EOAT	110
3.1.18.2 EOAT Change	110
3.2 TEACH-MODE USING THE TEXT EDITOR	111
3.2.1 Operation of the Texteditors	111
3.2.1.1 Entering the Text Editor	111
3.2.1.2 Exit the Texteditor and transfer the program to the CPU	111
3.2.1.3 Editing Functions	112
3.2.1.4 Online Editor	113
3.2.1.5 Offline Editor	113
3.2.1.6 Online Conversion	113
3.2.2 Step Operation	114

3.2.2.1 Procedure Step	114
3.2.3 Part Programs, PROGRAM Menu	115
3.2.3.1 Part Programs.....	115
3.2.3.1.1 Robot Program (ROBOT-PRG).....	115
3.2.3.1.2 Peripheral Program (PERI-PRG)	115
3.2.3.1.3 Robot Reference Program (ROBOT-REF)	116
3.2.3.1.4 Peripheral Reference Program (PERI-REF).....	116
3.2.3.1.5 115 additional Part Programs.....	116
3.2.3.1.6 All Modes Program (ALLMODE PRG).....	116
3.2.3.1.7 Sub-Allmodes Part Programs.....	117
3.2.3.1.8 EOAT Change Program.....	117
3.2.3.2 Manage Part Programs	118
3.2.3.2.1 Edit a Part Program	119
3.2.4 FILE Menu of the text editor	120
3.2.4.1 New	120
3.2.4.2 Open.....	121
3.2.4.3 Import Macro.....	122
3.2.4.4 Load Program from CPU.....	124
3.2.4.5 Save (NAME.ZIP)	124
3.2.4.6 Save As	125
3.2.4.7 Export Selection.....	126
3.2.4.8 Transfer program to CPU	128
3.2.4.9 Properties	129
3.2.4.10 Print	130
3.2.4.11 Exit to Main	132
3.2.4.12 Properties Window of the File Dialogue	133
3.2.4.13 Picture Viewer.....	134
3.2.4.13.1 Associate a picture with a Teach program.....	134
3.2.5 EDIT Menu	136
3.2.5.1 Selection Mode	136
3.2.5.2 Copy, Cut, Paste.....	137
3.2.5.3 Search-Toolbar	139
3.2.5.4 Show Prefixes.....	140
3.2.5.5 Show Default Names	141
3.2.5.6 Show Internal IDs.....	141
3.2.5.7 Show Line Numbers.....	141
3.2.5.8 Show Execution Times	142
3.2.6 Teach program Settings in the SETUP Menu	143
3.2.6.1 Online / Offline Editor.....	143
3.2.6.2 Axes Definition.....	143
3.2.6.3 Safety Areas	144
3.2.6.3.1 Create / Edit Safety Areas.....	146
3.2.6.3.2 Properties of the Working Area	147
3.2.6.3.3 Graphical Display of the Safety Areas.....	148
3.2.6.4 Cylinder Definition.....	150
3.2.6.4.1 Create / Edit Cylinders.....	151
3.2.6.5 Auto Switch Definition	152
3.2.6.6 Counter Names.....	154
3.2.6.7 Conveyor Names	155
3.2.6.8 Peripheral Output Names.....	156
3.2.6.9 Peripheral Input Names	157
3.2.6.10 Stacking Sensor Names.....	158
3.2.6.11 Stopwatch Names.....	159
3.2.6.12 Vacuum Names	160
3.2.6.13 Gripper Names.....	161
3.2.6.14 Vacuum Reference States	162
3.2.6.15 Gripper Reference States	163
3.2.6.16 Setup Standard Vacuum Pressure.....	164
3.2.7 Part program Settings in the PART PRG SETUP menu	166

3.2.7.1 Positions	166
3.2.7.1.1 Create / Edit Positions	166
3.2.7.2 Labels and Subroutines	168
3.2.7.3 Placing Program	168
3.3 TEACH COMMANDS OF THE TEXTEDITOR	169
3.3.1 Axes	169
3.3.1.1 Axes Selection	169
3.3.1.2 Terminology	169
3.3.1.2.1 Pneumactical Axes	169
3.3.1.2.2 Numerical Axes	169
3.3.1.2.3 Path Axes	169
3.3.1.3 Move	170
3.3.1.3.1 Absolute Movement	171
3.3.1.3.2 Relative Motion	171
3.3.1.3.3 Parallel Movements	172
3.3.1.3.4 Curve	174
3.3.1.3.5 Stop	175
3.3.1.3.6 Soft Torque Push	175
3.3.1.3.7 Release (optional)	177
3.3.1.3.8 Stack Sensor (optional)	178
3.3.1.3.9 Find 'n' Pick	179
3.3.1.3.10 Follow Conveyor (optional)	180
3.3.1.3.11 Mold Opening Synchronization (optional)	183
3.3.1.4 3D Motions	185
3.3.1.4.1 3D Absolute Motion	187
3.3.1.4.2 3D Relative Motion	187
3.3.1.4.3 3D Curve	187
3.3.1.5 Speed	188
3.3.1.5.1 VMAX	188
3.3.1.5.2 VABS	188
3.3.1.5.3 AABS	189
3.3.1.5.4 R3D	190
3.3.1.5.5 V3D	190
3.3.1.6 Wait Sync	191
3.3.1.6.1 Wait Path Sync	191
3.3.1.7 Movements and Parameters for Scara Robots	192
3.3.1.7.1 Pathmode	192
3.3.1.7.2 Angular Motions - „in degrees“	193
3.3.2 Placing Programs	194
3.3.2.1 Create/Edit Placing Programs	195
3.3.2.1.1 Dialogue for Axes Selection	196
3.3.2.1.2 Tab 1. - 1st Position, Name	197
3.3.2.1.3 Tab 2. - Stacking Pattern	198
3.3.2.1.4 Tab Options	199
3.3.2.1.5 Tab STS - Stacking Sensor	200
3.3.2.1.6 Tab Displacement	201
3.3.2.2 Add Placing Program Commands to the Teach Program	202
3.3.3 Injection Molding Machine IMM	203
3.3.3.1 Injection Molding Machine IMM - Standard Functions	203
3.3.3.1.1 MOLD CLOSE and FULLY MOLD OPEN	204
3.3.3.1.2 EJECTORS FORWARD/BACK and COREPULLERS PULL/SET	205
3.3.3.2 Injection Molding Machine IMM - Automatic Functions	206
3.3.3.2.1 MOLD CLOSE! WAIT	207
3.3.3.2.2 EJECTORS FORWARD/BACK (WAIT) and CORE PULLERS PULL/SET (WAIT)	207
3.3.3.3 Injection Molding Machine IMM - Logical Functions – Mold Was Closed	208
3.3.4 Smart Removal	209
3.3.4.1 Smart Removal: In	209
3.3.4.2 Smart Removal: Out	212
3.3.5 Peripherals	215
3.3.5.1 Conveyor	215
3.3.5.2 Peripheral Outputs PO	216
3.3.5.3 Cylinder function	217
3.3.5.3.1 Applying cylinders in a teach program	218

3.3.5.4 Vacuums.....	219
3.3.5.5 Vacuum Level	220
3.3.5.6 Grippers.....	221
3.3.5.7 Monitoring of Vacuum Circuits, Grippers and Peripheral Outputs	222
3.3.6 Counters.....	225
3.3.6.1 64 freely usable Counters	225
3.3.6.2 Reset / Manipulate Placing Counters	227
3.3.7 Program branching.....	229
3.3.7.1 Jump.....	229
3.3.7.2 Label.....	230
3.3.7.3 If - Elseif - Else - Endif	230
3.3.7.3.1 Variant 1:.....	231
3.3.7.3.2 Variant 2:.....	231
3.3.7.3.3 Variant 3:.....	232
3.3.7.4 Call - Subr – Ret	233
3.3.7.5 And.....	235
3.3.7.6 Or	235
3.3.7.7 End.....	236
3.3.7.8 Jump Conditions	237
3.3.7.8.1 Jump condition: Counter	237
3.3.7.8.2 Jump Condition: Placing Program.....	238
3.3.7.8.3 Jump Condition: PI-Peripheral Input	238
3.3.7.8.4 Jump Condition: IMM - Injection Molding Machine	239
3.3.7.8.5 Jump Condition: Vacuum.....	240
3.3.7.8.6 Jump Condition: AutoSwitch	240
3.3.7.8.7 Jump Condition: Part program	240
3.3.7.8.8 Jump Condition: Area	241
3.3.7.8.9 Jump condition: Position	241
3.3.7.8.10 Jump Condition: Mode.....	242
3.3.7.8.11 Jump Condition: Stacking sensor.....	243
3.3.7.8.12 Jump Condition: Start	243
3.3.7.8.13 Jump Condition: Production End Active	243
3.3.7.8.14 Jump Condition: Input.....	243
3.3.7.8.15 Jump Condition: Output	244
3.3.7.9 Start/Stop Part Program.....	245
3.3.8 Wait functions.....	246
3.3.8.1 Wait	246
3.3.8.2 Wait Timer	246
3.3.8.3 Time.....	247
3.3.9 More Commands.....	248
3.3.9.1 More	248
3.3.9.2 Remark (REM).....	248
3.3.9.3 Production End	249
3.3.9.4 Blockstop lock/unlock.....	250
3.3.9.5 Stopwatch.....	251
3.3.9.6 Operating Mode	253
3.3.9.7 Eco Mode	254
3.3.9.8 Area Switch.....	255
3.3.10 AutoSwitches	256
3.3.10.1 The Teach command AutoSwitch	257
3.3.11 User Defined Error Messages	259
3.3.11.1 Introduction.....	259
3.3.11.2 Creating and Storing the File UserErr.txt.....	259
3.3.11.3 The Teach Command ALARM	261
3.4 QUICK EDITOR.....	262
3.4.1 Introduction	262
3.4.2 Operation of the Quick Edit	262
3.4.2.1 Entering the Quick Edit	262
3.4.2.2 Display and Editing Functions	262

3.4.2.3 Exit the Quick Edit.....	263
3.4.3 FILE Menu of the Quick Edit.....	264
3.4.3.1 Open.....	264
3.4.3.2 Open Template	265
3.4.3.3 Save (NAME.ZIP)	266
3.4.3.4 Save As	266
3.4.3.5 Properties	268
3.4.3.6 Print	269
3.4.3.7 Exit to Main	270
3.4.4 Parameters for the Quick Edit.....	271
3.4.4.1 Create Parameters.....	271
3.4.4.2 Apply Parameters	273
4 HOW TO TEACH A PROGRAM.....	274
4.1 USING THE QUICK NEW WIZARD AND THE QUICK EDIT	274
4.2 PROGRAMMING THE ROBOT IN THE TEXT EDITOR	279
4.2.1 Creating a New Program in the Text Editor	279
4.2.2 Verifying Your Program Using STEP Mode	280
4.2.3 Storing Your Program on a storage medium.....	280
4.3 PROGRAM EXAMPLE	281
4.3.1 Function of Program.....	281
4.3.2 Sequence	281
4.3.3 Sketch of Sequence	282
4.3.4 Sketch of Depositing Program	282
4.3.5 Reference Travel of Robot	282
4.3.6 Robot Teach Program (Robot Prg.).....	283
4.3.6.1 Placing Program Placingp-001.....	284
4.3.7 Reference Program for the Robot (Robot Ref.)	284
5 TROUBLESHOOTING.....	285
5.1 OPERATING ERRORS	285
5.1.1 Operating Errors in Plain Text Messages	286
5.1.1.1 Axis Errors	286
5.1.1.2 Vacuum Errors.....	289
5.1.1.3 Gripper Errors	289
5.1.1.4 Peripheral Equipment Output Messages.....	290
5.1.1.5 Cylinder Messages	290
5.1.1.6 Conveyor Belt Messages	290
5.1.1.7 General Hardware Error Messages.....	291
5.1.1.8 System Error Messages.....	293
5.1.1.9 CAN Device Messages	293
5.1.1.10 Teach program Compiler Messages	294
5.1.1.11 Teach program Interpreter Messages	295
5.1.1.12 Configuration Errors.....	296
6 MAINTENANCE.....	297
6.1 GENERAL	297
6.2 LUBRICATION INSTRUCTIONS FOR LINEAR GUIDES AND GEAR RACKS	298
6.2.1 Lubrication Intervals for Standard Robots.....	298
6.2.2 Lubrication Procedures.....	298

6.2.3 Grease Specification	298
6.3 DRIVE BELTS	299
6.4 MOTOR BRAKES.....	299
6.5 BATTERY REPLACEMENT.....	300
6.5.1 IPC.....	300
6.5.2 Sanyo R Inverters at Axes with Absolute Encoders.....	301
6.6 MOLD SAFETY RELAYS.....	303
6.7 MAINTENANCE SCHEDULE	304
6.8 AUTOMATIC LUBRICATION.....	305
6.8.1 Description	305
6.8.2 System Maintenance.....	306
6.8.2.1 Lubrication System Maintenance Procedure:.....	306
6.8.2.2 Suitable greases	306
7 KEY TERMS	307

1 Safety



Wittmann Robots can be operated at high speeds so as to allow short removal and cycle times. In order to minimize the risk of injury to personnel, the robot and automation equipment must always be operated with the proper safeguards, in accordance with legal regulations.

1.1 Safety Regulations



In many countries the obligation of robot vendors to supply systems meeting applicable safety regulations and the responsibility of the operator of these devices are governed by law or by safety standards of professional associations. Always comply to the regulations applicable for you.

1.2 Safety Features

1.2.1 Safeguarding

The safeguarding required for operation of the robot is not included in our standard scope of supply, since adaptation to specific site conditions is required. If such safeguarding is provided by you, please note that it must be installed **prior** to startup of the equipment in order to be included in the safety circuit of the system upon startup. See Chapter 2.4 Safety Package/Safeguarding.



The user bears the legal responsibility for following safety regulations.

1.2.2 Emergency STOP Buttons

One emergency stop button is located on the Teachbox.

When an emergency stop button is pressed, the power is turned off. The gripper and vacuum valves and the vacuum pump are not disconnected, to avoid dropping parts from the gripper. In addition, the controls and the Teachbox will remain under power to allow indication of error messages.

The emergency stop button can be released by pulling it.

The emergency stop circuits of the robot and of the injection molding machine are connected by the robot-machine interface. Therefore when the emergency stop button on the molding machine is pressed the robot will also go into emergency stop and vice versa.

1.2.3 Permit Key on Teachbox

This key must be depressed for setting up for manual operation in order to be able to carry out travel motions with the robot. For more precise operation, see Chapter 3.1.10. Should this permit key be released during setup, the safety functions will interrupt all travel motion of the robot.

1.3 Limitations

Limitations of use		
Specified use	This appliance is to be used to handle (plastic-) parts within the production process of an injection molding machine. The robot may only be used after safeguarding according to EN ISO 10218 has been established, and the safety concept described in this user manual has been realized.	
Reasonably foreseeable misuse	Installation and operation without consideration of EN ISO 10218	
Field of application		
Commerce	Yes	
Industry	Yes	
Categories of users	Task	Qualifications / Impairments
Qualified personnel	Installation / commissioning / decommissioning / repairs / Maintenance / removal	Professional training
Apprentices	Guided operation	semiskilled
Operators	Operation	skilled / semiskilled

Spatial limitations	
Description of the system	The robot consists of the z-axis part, which contains the electrical system and the CPU of the control system; furthermore it guides the x-axis. The x-axis part also holds the y-axis that executes the vertical movements of the robot. The y-axis additionally holds a rotary device.

Time limitations	
Specified time of use	10 to 20 years
Recommended maintenance intervals	See chapter 6 Maintenance

Other limitations	
Ambient temperature	+5 to +40 °C
Relative humidity	15 to 85 %
Required degree of cleanliness	No special requirements
Height above sea level	0 to 1000 m
Electrical supply	Depending on version, see type plate
Required pressure for compressed air	4 – 7 bar
Hazardous areas (EX)	Do not use in hazardous areas (EX)

1.4 Capability of the Safety Control

The capability of the safety control of the robot conforms with PLr = d of EN ISO 13849-1.

1.5 Control of Hazardous Energy – (Lockout / Tagout)

Lockout / Tagout (LOTO) refer to specific practices and procedures to safeguard employees from the unexpected energy, startup of machinery/equipment, or the release of hazardous energy during service or maintenance activities.

This requires that a designated individual turn off and disconnect the machinery/equipment from its energy source(s) before performing service or maintenance and that the authorized employee(s) lock and tag the energy-isolating device(s) to prevent the release of hazardous energy and take steps to verify that the energy has been isolated effectively.

The following is a glossary and **general** description of the LOTO procedure.



Always refer to applicable local codes in performing Lockout/Tagout procedures

1.5.1 Glossary

Affected Employee	An employee whose job requires them to operate a machine or piece of equipment on which service or maintenance is being performed.
Authorized Employee	A person who locks or implements a tagout system procedure on machines or equipment to perform service or maintenance on that machine or equipment. An authorized employee and an affected employee may be the same person when the affected employee's duties also include performing service or maintenance.
Energy Source	Any source of electrical, mechanical, hydraulic, pneumatic, chemical, thermal, or other energy.
Lockout	The placement of a lockout device (such as a lock) on an energy-isolating device, in accordance with an established procedure that ensures the device and the equipment cannot be operated until the lockout device is removed.
Servicing and / or Maintenance	Workplace activities such as constructing, installing, setting up, adjusting, inspecting, modifying, maintaining or servicing machines or equipment. These activities include lubrication, cleaning or un-jamming of machines or equipment, and making adjustments or tool changes where the employee may be exposed to the unexpected energy, start-up of equipment or release of hazardous energy.
Tagout	The placement of a tagout device (such as a tag) on an energy-isolating device, in accordance with an established procedure that ensures the device and the equipment may not be operated until the tagout device is removed.

1.5.2 Lockout/Tagout Kit

Use of an approved LOTO kit is required. It is the customer's responsibility to provide this kit.



Fig.: Examples of LOTO devices

Customers are also responsible for ensuring that all employees are properly trained and utilize the LOTO procedures.

1.5.3 Lockout/Tagout Procedure

Before service/maintenance activities begin, the following procedures must be implemented in the order listed below when locking or tagging out equipment:

1. Prepare for shutdown

The Authorized Employee shall evaluate the equipment to be serviced and identify all sources of hazardous energies and the methods necessary to control them.

2. Notify all Affected Employees

The Authorized Employee turning off the power shall notify Affected Employees in the work area that power will be shut off, the reason for the shut-down, and that the equipment will be locked/tagged out.

3. Shut down equipment

The equipment/machine shall be shut down by the normal stopping procedure. When appropriate, a "DO NOT OPERATE" tag shall be affixed to the power switch.

4. Isolate equipment

The equipment/machine shall be de-energized, secured and isolated from hazardous energy sources. An orderly shutdown must be utilized to avoid any increased or additional hazard(s) to employees.

5. Lockout/Tagout

The Authorized Employee shall place locks and/or tags in the appropriate energy isolating locations.

6. Release stored energy

After lockout devices have been placed on the equipment, all stored electrical, pneumatic, gravitational, mechanical and/or thermal energy must be disconnected and drained to a zero-energy state or otherwise made safe.

7. Verify isolation

Before performing maintenance on the machine, the Authorized Employee verifies the system is isolated. This is generally accomplished by first establishing that no personnel are exposed and then turning the machine switch to the ON position using the normal operating controls

8. Perform the service/maintenance activity

LOTO devices should be removed promptly following completion of Service/maintenance activities.

1.5.4 Removal of LOTO

Once the specific maintenance or service work has been completed, the person who attached the lock or tag is responsible for promptly removing that device.

Removal of LOTO devices will be accomplished by following the steps listed below:

1. Inspect the work area.

Ensure that the equipment/machine is fully assembled and operational, all tools and nonessential items are removed, and all safety guards are reinstalled.

2. Ensure that all employees are clear of the equipment/machine.

3. Remove the LOTO device.

Each device must be removed by the person who put it on.

4. Reenergize the equipment/machine.

5. Notify Affected Employees that servicing has been completed and the machine is ready for use.

2 Installation



Installation and startup shall be performed by authorized personnel only, in order to avoid injury to personnel or damage to equipment.

Our trained service engineers are available to assist you.

2.1 Mechanical Installation

2.1.1 Dimensional Drawing – Adapter Unit

Please see the accompanying drawing for the dimensions of the robot with the dimensions for attachment of the gripper.

The robot is normally mounted on the fixed platen of the injection molding machine by means of an adapter unit in accordance with VDMA 24466 or FHK-85-12-22.

If the adapter is supplied by us, a drawing for installation on the machine is enclosed.

2.1.2 Mounting of the Robot

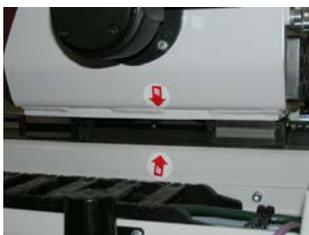
The weight of the robot is shown on the robot identification plate on the Z beam.

Depending on the version the robots are equipped with one or more of the following devices for lifting:

- lifting eyes on the main beam (Z axis).
- machined slots for lifting with a fork lift (e.g., in low bays) or with belts.
- Additional holes for the use of lifting rods.



Before lifting the transport pallet, you have to make sure, that the main carriage (Z axis) and kick stroke (X axis) are in their balanced position.



The axes are in a balanced position, when the respective arrow marks are aligned.

The picture on the left shows the red arrow marks of the correctly aligned z-axis of a W811 as an example.



The main carriage and kick stroke can only be moved by applying voltage from an external power supply in the X and Y axes. Because of the high accelerations of servo drives, this operation is hazardous and therefore should be performed only by our trained service engineers.



Before lifting the robot, the vertical axis must be secured against tipping. When lifting with a fork lift, secure the robot against slipping.

2.1.3 Demounting of the Robot

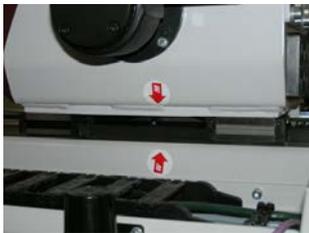
The weight of the robot is shown on the robot identification plate on the Z beam.

Depending on the version the robots are equipped with one or more of the following devices for lifting:

- lifting eyes on the main beam (Z axis).
- machined slots for lifting with a fork lift (e.g., in low bays) or with belts.
- Additional holes for the use of lifting rods.



Before dismantling and lifting away from the console, you have to make sure, that the main carriage (Z axis) and kick stroke (X axis) are in their balanced position.



The axes are in a balanced position, when the respective arrow marks are aligned.

The picture on the left shows the red arrow marks of the correctly aligned z-axis of a W811 as an example.



The easiest way to move the robot axes into balance, is to do it with the manual functions provided by the Teachbox, before the robot is decommissioned.



Before dismounting, all electrical and pneumatical connections have to be disconnected.



Before lifting the robot away, all cables and hoses hanging down from the robot have to be secured properly, in order to avoid damages, locking the transport, and to avoid the danger of personal injury.



Before lifting the robot, the vertical axis must be secured against tipping. When lifting with a fork lift, secure the robot against slipping.

2.2 Pneumatics

2.2.1 Compressed Air Connection

This connection is made on the service unit with a 1/2" hose. A shutoff valve (ball cock or quick connection coupling) should be installed at the point of connection of the main compressed air line.

Required air pressure: 60 - 100 psi (4 – 7 bar)

Air consumption per robot cycle:

The air consumption depends on the optional equipment installed on your robot.

Air consumption of the robot may be calculated from the following table:

CNC Robots	Consumption per stroke *		Cons. per sec. **
	C axis	A axis	Vacuum per circuit
W811	0,3	0,4	0,4
W813, W823	0,2	0,2	0,4
W821, W831	1,3	0,4	0,4
W832, W833	1,1	1,3	0,4
W843	1,1	1,3	0,4
W853	6,4	8,3	0,4
W873	6,6	4,5	0,4

*: In ft³ at 90 psi (6 bar)

** : For venturi vacuum circuits only

Example of calculation of compressed air consumption:

W833CNC: with X , Y, Z axes motor driven
 A and C axes pneumatic
 2 vacuum circuits, operated 10 sec. each

1)	A axis: moved twice per cycle:	2 x 1.3 =	2,6 dm ³
2)	C axis: moved twice per cycle:	2 x 1.1 =	2,2 dm ³
3)	2 vacuum circuits: each one operated for 10 sec./cycle:	2 x (10 x 0.4) =	8,0 dm ³
4)	<u>Total consumption:</u>	2,6 + 2,2 + 8 =	<u>12,8 dm³</u>

2.3 Electrical Connections

2.3.1 Mains Power Supply

The required electrical voltage and power supply are indicated on the identification plate of the robot. Power connection normally is made with cables and CEE plugs.



The power connection must be made by a qualified electrician in accordance with federal, state and local regulations.

For personnel safety, proper installation of the earth ground must be verified.

After connection to the power supply, the direction of rotation must be checked (e.g., on the vacuum pump). The phasing must be clockwise.

2.3.1.1 Fault Current Safety Switches



Fault current (old: FI, new: RCD) safety switches may be used only **under certain conditions** in conjunction with frequency converters:

- In frequency converters with 3-phase input voltage, in the event of a ground leak a part of the fault current may **prevent release of an FI / RCD safety switch**.
- Parasitic capacities in the converter can in principle cause leakage currents, which may result in **false tripping**.

Alternative safety measures are for example grounding or protective isolation.

If a fault current safety switch is used in robots with 3-phase converters, it must be a new RCD TYPE B or RCMA .

If a fault current safety switch is used in robots with single-phase converters, it must be a new RCD TYPE B or RCD TYPE A .

2.3.2 Interface with the Injection Molding Machine

The robot is equipped with an electrical interface (plug type HAN 32A) according to Euromap E67 / E12. In robots supplied as integrated systems, molding machines are equipped with CAN-BUS controls and safety signals are executed by an emergency stop. See wiring diagram at the end of this chapter and pin description (Section 2.6).



Connection of the interface plug to the machine and testing of all signals must be done by a specialist in injection molding machines and robots. Preferably, this should be done by one of our service engineers together with a qualified service engineer for the injection molding machine.



The interface signal functions must be carefully tested, as improper operation may cause malfunction or damage to the robot and molding machine.



In particular, the functions of the safety circuits must be thoroughly checked:

- Emergency stop signals from and to the IMM
- Function of mold safety interlock switches S5/S6
- External safety circuits for access door to the protected area

2.3.3 Mold Monitoring S5/S6



Proper adjustment of the switching points of the roller switches S5 and S6 is extremely important, since they secure the closing and opening motions of the IMM. These roller switch signals are **independent** from the permission signals of the robot program and are **directly** wired through the electrical interface to the IMM (redundant safety).

The switches for S5 and S6 can be physical switches (proximity sensors or roller switches) or virtual switches (software switching points).

Function and Adjustment of Roller Switch S5:

The roller switch S5 is mounted on the vertical axis and is operated by the cam on the vertical axis. Only when S5 is actuated, the closing/opening motion is released.

Adjust the cam (the switching point) in a way that S5 is only actuated, when gripper and part are at a position with sufficient safety distance above the mold.

Function and Adjustment of Roller Switch S6:

The S6 switch is mounted to the main traversing carriage and is operated by the cam on the Z beam. S6 overrides the function of S5, so that the robot arm can move down outside the machine without interrupting closing or opening of the IMM.

Adjust the cam (the switching point) in a way that S6 is actuated only while providing a sufficient safety distance **outside** the protective guarding of the machine (with the largest part on the gripper!).



After changing switching points for Mold Monitoring, the roller switch monitoring has to be re-initialized.



After proper adjustment of the cams, double-check the function of the safety switches S5 and S6. If neither switch is actuated (robot arm inside mold) ensure that even with a closing signal **present** from the robot the closing and opening motion of the IMM is **not** possible!

2.3.3.1 Settings for Virtual Roller Switches (Mold Safety Switches)

Robots with absolute encoders use virtual mold safety switches instead of real switches and cams.

Open the **Init** menu **Roller Switches** to set the switching points and behavior of those virtual switches.

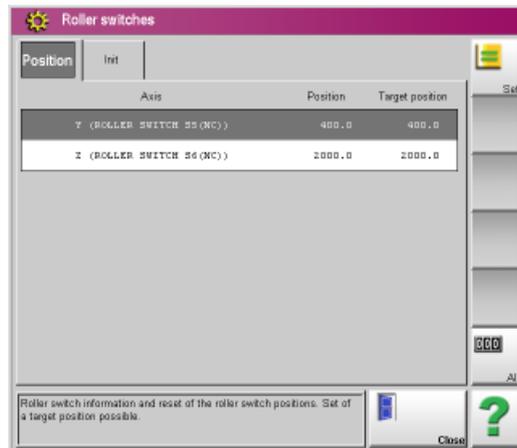


Fig.: Roller Switches Position

On the tab **Position** select the desired axes and press  to input the position of the switching point for the related virtual roller switch with the dialogue for [numerical input](#).

Use the tab **Init** to define if the 0-positions of the axes are situated inside or outside the mold area.

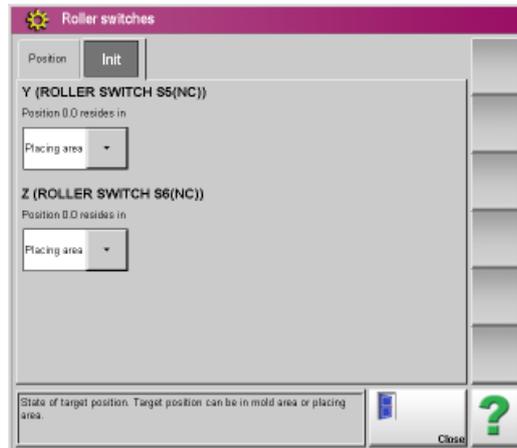


Fig.: Roller Switches Init

Once you finished all settings, press  to exit the window.

2.3.3.2 Initialize Roller Switch Monitoring

The software constantly monitors the switching points of the roller switches for mold safety, in order to detect any malfunctions in time. This serves as an additional safety for avoiding collisions with the mold of the IMM.

Open the **Init** menu **Roller Switches**.

The tab **Position** shows the positions of the switching points as determined by the software for all mold safety switches present at the robot.

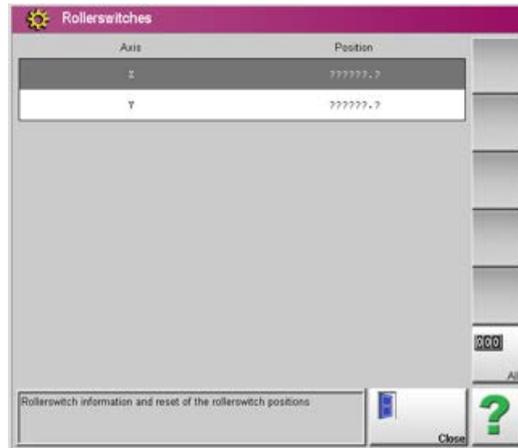


Fig.: Window for roller switches

Switching points that have not been detected since the last time the roller switch monitoring was reset, will display '???????.?' as their position.

If during a product change or during initial setup switching points have been changed

by altering the cams, the roller switch monitoring has to be reset by pressing . All Positions will display '???????.?' and will automatically be re-determined during movements of the respective axes in operation modes manual and automatic.

Press  to close the window.

2.3.3.3 Test Procedure for Mold Safety S5 / S6

- Switch off **Operation with Robot**. The LED at the button  is not lit.
- Slightly close the mold. The Signal **Mold Open** must be off.
- Now try to move the Y-axis into the mold area.
- The Y-axis must stop latest when leaving the S5. The error message "Mold Monitoring" or "Mold Monitoring Braking Distance" must appear on the Teachbox.
- Confirm the message.
- Check if the robot has left the S5. Use the menu **View – I/O** to do so; the input **482 Roller Switch S5** must be off. If this is not the case, deactivate mold monitoring in the menu **Init – Deactivate monitoring** and carefully move the Y-axis off the switch.
- Now try to close and open the mold. Both movements must be impossible, and an error message must appear at the IMM.
- Move the Y-axis up and out of the mold area.
- Activate mold monitoring in the menu **Init – Deactivate monitoring**, if you have deactivated it previously.

In case of a negative test result, please contact our service center.

2.4 Safety Package/Safeguarding

As outlined in Chapter 1, the work envelope of the robot must be properly safeguarded in accordance with legal regulations.

2.4.1 Safety Package

Our robots with R8.3 controls are equipped as a standard with a safety package.

The **safety package** includes:

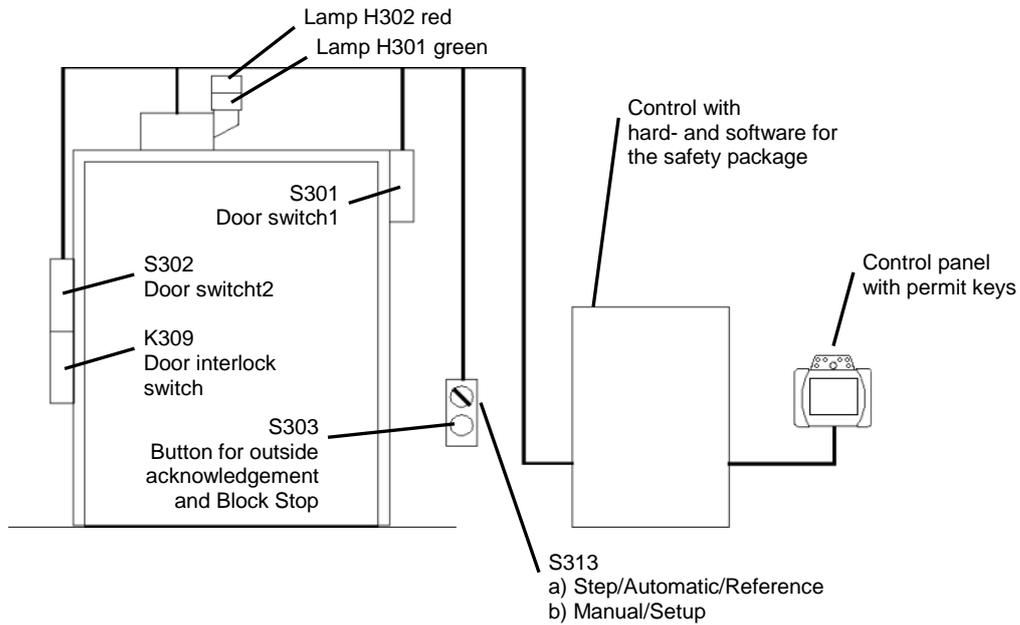
- Emergency Stop Disconnect Unit with two channels
- Safety contactor(s) for hardware shutoff of the axis drive when the safety door (hatch) is opened. In addition, the software monitors these contactors for proper operation (redundant safety).
- Remote control with 2 permit keys
- Software safety package (monitoring of inputs and outputs)
- Cable with plug for ease of connection of all necessary switching and display elements for mounting on an access door or hatch



The robot can be operated only if all safety requirements have been met, i.e., all protective devices have been properly installed **prior** to startup of the robot.

For a detailed description of the function of the safety package see Section 3.1.10.

2.4.2 EXTERNAL Safety Package for Safety Door or Access Hatch



External safety package includes:

- Door monitoring switch „Safety Door Open“ (S301)
- Door interlock with contact for software monitoring of function
The door can be opened only when the robot motions have come to a complete stop (K309).
- Door opening contact with roller switch and cam for confirmation of position:
Door open/closed (S301/S302)
- Outside acknowledgment button (S304)
- Selector switch for manual/automatic operation (S313)
- Mounting panel with junction box, plug and 2 indicator lamps:
(H301): robot disabled (H302): robot enabled (automatic operation possible)

These parts are required in order to confirm the function of the safety door in the safety circuit of the robot program and are obtainable, as an option, with or without safety door.

Before starting up the robot, check the safety circuits as follows:

- 1) Install the complete safety package.
- 2) Check K24V/**X6 PIN7 (wiring board)**:
If you can measure 24V between **X6/PIN 7 (wiring board)** and **X6/ PIN 8 (wiring board)** the emergency stop circuit is enabled.
- 3) Check of A24V on **X3 PIN 2 (wiring board)** (robot disabled):
If a measurement between terminals **X3 PIN 2 (wiring board)** and **X6/ PIN 8 (wiring board)** indicates 0V, the circuit is working properly.
- 4) Open the safety door.
- 5) Close the safety door.
- 6) Press the outside acknowledgment button (lamp H301 lights).
- 7) Turn the selector switch to „Auto“ setting (lamp H302 lights).
- 8) Check **X3 PIN 2 (Verdrahtungsprint)** (robot enabled):
If a measurement between terminals **X3 PIN 2 (Verdrahtungsprint)** and **X6/ PIN 8 (wiring board)** indicates 24V, the circuit is working properly.
- 9) Turn the selector switch to „Manual“ (lamp H301 lights).
- 10) Check of both permit keys for remote control:
When one of the two permit keys is held in its **intermediate position**, the lamp H301 lights and operation of the robot with safety door open is possible.
If the permit key is pressed fully, operation of the robot is not possible (panic).
- 11) Turn selector switch to „Auto“ position (lamp H302 lights).
- 12) Teach the robot a „blank“ program and start automatic cycle:
It must not be possible to open the safety door during automatic operation.

Function of Safety Package:

- A)** Interruption of cycle in automatic operation
The selector switch is in the **AUTO/STEP/REFERENCE** position.
- 1) Pressing the „**Block Stop**“ key on the Teachbox or the button at the safety door allows the robot to complete the last motion, then the door interlock magnet is released and the safety door can be opened.
The safety package immediately interrupts the motion in both the hardware and software upon forcible opening of the safety door.
 - 2) To resume operation:
 - Close doors
 - Press outside acknowledgment button
 - Press automatic start key. Cycle will resume with the next instruction in the teach program.
- B)** Manual or setup operation
- 1) Set selector switch to **MANUAL/SETUP**.
 - 2) The door interlock magnet will be released, the door can be opened, and the safety area can be entered.
 - 3) All motions can be performed by remote control by simultaneously pressing the permit keys and the function keys. Before every new motion, the permit key must be released **one time** and then pressed again.
A maximum drive speed of 250 mm/sec is possible In *Manual/Setup Operation*.
 - 4) To resume operation in automatic mode:
 - Close doors
 - Press outside acknowledgment button
 - Set selector switch to automatic operation
 - Press automatic start key; this ensures that the system cannot inadvertently be started by a third party if an operator is still in the protected area.

Note:

According to applicable safety regulations, slow motions of the robot axes (max. of 250 mm/sec or 10 in/sec) are not sufficient to allow access to the work envelope of the robot and to avoid accidents. Access is allowed **only** if the automatic mode is **completely** interrupted and this condition is secured by hardware contactors that prevent any motion.

2.4.3 Safety Guarding

This consists of safety door and safety guarding elements.

These safety guarding elements are available as an option in the following sizes (width * height):

- 1000 mm * 2000 mm
- 500 mm * 2000 mm

These provide complete safety guarding of the work envelope of the robot and/or the entire automated system.

2.5 Electrical Interface

This section is not applicable to robots with an integrated system, since here the signals are transmitted to the injection molding machine via a CAN-BUS interface.

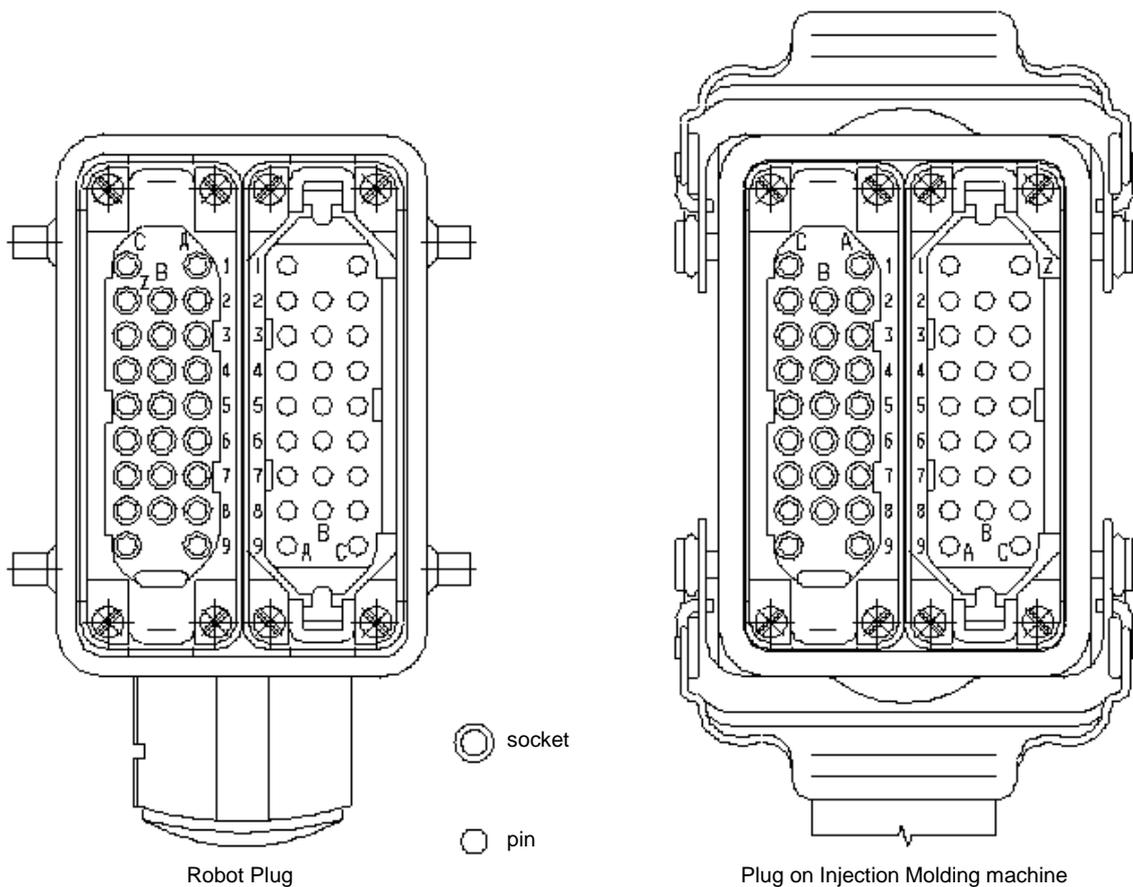
Robots equipped with a R8.3 control unit are available with 2 different interface versions to communicate with the injection molding machine:

- Euromap 67
- Euromap 12

Both versions are described in the following chapters.

2.5.1 Euromap 67 interface

The Euromap 67 interface defines the connection plug between the injection molding machine and the robot:



The robot-injection molding machine interface is designed according to the directives of Euromap 67, which states:

Unless otherwise noted, the signals level signals, which are maintained during the described function.

2.5.1.1 Injection Molding Machine (IMM) Signals

Pin contact no.	Function
ZA1 ZC1	Emergency Stop Channel 1 The emergency stop switch of the injection molding machine is used to interrupt the emergency stop circuit of the robot.
ZA2 ZC2	Emergency Stop Channel 2 The emergency stop switch of the injection molding machine is used to interrupt the emergency stop circuit of the robot.
ZA3 ZC3	Safety system active Channel 1 For protecting against hazardous motions of the robot. The switch is closed when the safety system of the injection molding machine is active.
ZA4 ZC4	Safety system active Channel 2 For protecting against hazardous motions of the robot. The switch is closed when the safety system of the injection molding machine is active.
ZA5	Reject The signal is HIGH when the molded piece is a reject. The switch must be closed when the tool is open and must remain HIGH at least until "close tool enabled" (see pin contact No. A6).
ZA6	Mold closed HIGH signal when tool closing has been completed; the signal "close tool enabled" is no longer necessary (see pin contact No. A6).
ZA7	Mold open HIGH Signal if the mold opening position is equal or more than the required position. Inadvertent alteration to mold opening stroke smaller than that required for the robot to approach must be impossible.
ZA8 Optional	Mold at intermediate position HIGH signal when the mold opening has reached the specified intermediate position and remains HIGH until the mold is completely open. The signal may be used in two ways: 1.) The mold stops in the intermediate position, whereupon a signal is sent to the robot. Complete opening of the IMM takes place through the signal "complete mold opening enabled" (see pin A7). 2.) The IMM transmits the signal, but does not remain in the intermediate position. Signal is LOW when the intermediate position is not used.
ZA9	Signal Voltage Robot 24V DC
ZB2	Fully automatic mode injection molding machine HIGH signal, when operating the injection molding machine together with the robot is possible.
ZB3	Ejector back position HIGH signal when the ejector is back, regardless of the position of the movable tool plate. The signal acknowledges "ejector back enabled" (see pin contact No.B3).
ZB4	Ejector forward position HIGH signal when the ejector is forward. The signal acknowledges "ejector forward enabled" (see pin contact No. B4).
ZB5	Core pullers 1 free for robot to approach HIGH signal when the core pullers, regardless of the position of the movable tool plate, are in position for the robot to approach.
ZB6	Core pullers 1 in position to remove molding HIGH signal when the core pullers are in position for removal of the injection molding.
ZB7 Optional	Core pullers 2 free for robot to approach HIGH signal when the core pullers, regardless of the position of the movable tool plate, are in position for the robot to approach.

Pin contact no.	Function
ZB8 Optional	Core pullers 2 in position to remove molding HIGH signal when the core pullers are in position for removal of the injection molding.
ZC5	reserved for future EUROMAP signal
ZC6	reserved for future EUROMAP signal
ZC7	reserved for future EUROMAP signal
ZC8	free
ZC9	Signal Ground Robot 0V

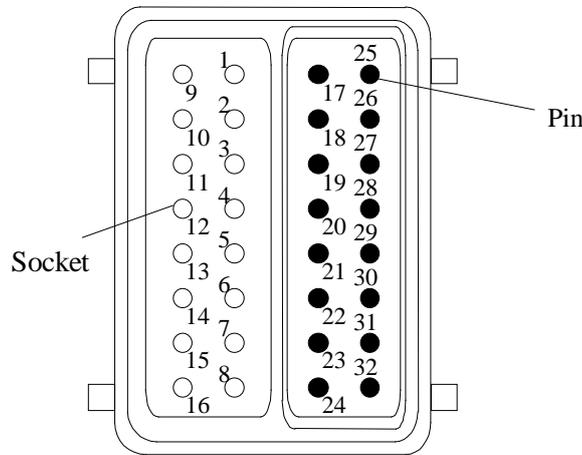
2.5.1.2 Robot Signals

Pin contact no.	Function
A1 C1	Emergency stop of robot Channel 1 Opening of the switch contacts of the robot must shut off the control system of the molding machine.
A2 C2	Emergency stop of robot Channel 2 Opening of the switch contacts of the robot must shut off the control system of the molding machine.
A3 C3	Mold area free Signaling is effected by the limit switch at the travel-in rail. The switch is opened when the travel-in rail, in the region of the injection molding machine, leaves its starting position before it is moved into the tool area. If the switch is open, neither a closing nor opening motion of the tool may take place. Even when the control system of the robot is shut off, the switch must work as described.
A4 C4	reserved for future EUROMAP signal
A5	free
A6	Close mold enabled HIGH Signal when the robot is far enough out of the tool that it can be closed and when other robot control systems enable closing of the tool. The signal remains HIGH for the duration of the tool-closing operation. In the event of a LOW signal due to a disturbance, the tool-closing motion must be aborted. Note: The signal "close tool enabled" may not be linked with other OR signals in any operating mode.
A7 optional	Complete mold opening enabled HIGH Signal when the robot has removed the piece from the mold and permits further opening of the mold. The contact must remain closed until the IMM gives the signal "mold open".
A8	reserved for future EUROMAP signal
A9	Signal Ground IMM 24V DC
B2	Operation with robot At operation with robot the signal is LOW.
B3	Ejector back enabled HIGH signal when the removal operation has been performed far enough for the motion "ejector back" to be carried out. The signal is HIGH for the duration of the motion "ejector back". The signal must be maintained at least until the signal "ejector back" from the molding machine (see pin contact No. ZB3).
B4	Ejector forward enabled HIGH signal, when the removal operation has been performed far enough for the motion "ejector forward" to be carried out. The signal is HIGH for the duration of the motion "ejector forward." The signal must be maintained at least until the signal "ejector forward" from the molding machine (see pin contact No. ZB4).

Pin contact no.	Function
B5	Enable movement of core pullers 1 to position for the robot to approach freely. HIGH signal when the motion of the core pullers is to the position for the robot to approach freely is enabled.
B6	Enable movement of core pullers 1 to position for removal of the molding HIGH signal when the motion of the core pullers is to the position for removal of the molding is enabled.
B7 optional	Enable movement of core pullers 2 to position for the robot to approach freely. HIGH signal when the motion of the core pullers is to the position for the robot to approach freely is enabled.
B8 optional	Enable movement of core pullers 2 to position for removal of the molding HIGH signal when the motion of the core pullers is to the position for removal of the molding is enabled.
C5	free
C6	reserved for future EUROMAP signal
C7	reserved for future EUROMAP signal
C8	free
C9	Signal Ground IMM 0V DC

2.5.2 Euromap 12 Interface

The interface consists of the plug connection between the injection molding machine and the robot :



The robot-injection molding machine interface is designed according to the directives of VDMA 24465, and/or according to Euromap 12, which state:

Unless otherwise noted, the signals are maintained during the described function and should not be shorter than 0.5 sec.

2.5.2.1 Injection Molding Machine (IMM) Signals

Pin contact No.	Function
1, 9	Emergency Stop The emergency stop switch of the injection molding machine is used to interrupt the emergency stop circuit of the robot.
2	Mold open The signal is produced by closing the switch. Signal must be interlocked so that it can be produced only when the minimum tool opening for insertion is reached.
3, 11	Safety system active For protecting against hazardous motions of the robot. The switch is closed when the safety system of the injection molding machine is active.
4	Ejector back Switch is closed when the ejector is back, regardless of the position of the movable tool plate. The signal acknowledges "ejector back enabled" (see pin contact No. 21).
5	Ejector forward Switch is closed when the ejector is forward. The signal acknowledges "ejector forward enabled" (see pin contact No. 22). The signal may be an impulse.
6	Core pullers free for robot to travel in Switch is closed when the core pullers, regardless of the position of the movable tool plate, are in position for free travel-in of the robot.
7	Core pullers in position for removal of injection moldings Switch is closed when the core pullers are in position for removal of the injection molding. The signal may be an impulse.
8	Reject Switch is closed when the molded piece is a reject. The switch must be closed when the tool is open and must remain closed at least until "close tool enabled" (see pin contact No. 17).
10	Fully automatic mode injection molding machine Switch is closed when the operating mode selector switch is on "fully automatic mode."
11	Reserved for SUVA Safety Package: IMM safety door closed +24V
12	Mold closed Switch is closed when tool closing has been completed; the signal "close tool enabled" is no longer necessary (see pin contact No. 17).
13	Free
14	Mold at intermediate position Switch is closed when the IMM has reached the specified intermediate position and remains closed until the IMM is completely open. The signal may be used in two ways: 1.) The mold stops in the intermediate position, whereupon a signal is sent to the robot. Complete opening of the IMM takes place through the signal "complete mold opening enabled" (see pin 28). 2.) The IMM transmits the signal, but does not remain in the intermediate position. Switch is open when the intermediate position is not used. (Option: Reserved for SUVA Safety Package: IMM safety door closed Channel 2)
15	No part available
16	Signal voltage of robot

2.5.2.2 Robot Signals

Pin contact No.	Function
17	<p>Close tool enabled Switch closes when the robot is far enough out of the tool that it can be closed and when other robot control systems enable closing of the tool. The switch is closed for the duration of the tool-closing operation. In the event of contact release due to a disturbance, the tool-closing motion must be aborted. Note: The signal "close tool enabled" may not be linked with other OR signals in any operating mode.</p>
18, 26	<p>Tool area free Signaling is effected by the limit switch at the travel-in rail. The switch is opened when the travel-in rail, in the region of the injection molding machine, leaves its starting position before it is moved into the tool area. If the switch is open, neither a closing nor opening motion of the tool may take place. Even when the control system of the robot is shut off, the switch must work as described.</p>
19, 27	<p>Emergency stop of robot Opening of the switch contacts of the robot must shut off the control system of the molding machine.</p>
20	<p>Operation with robot With the robot in operation the switch is open.</p>
21	<p>Ejector back enabled Switch closes when the removal operation has been performed far enough for the motion "ejector back" to be carried out. The switch is closed for the duration of the motion "ejector back". The signal must be maintained at least until the signal "ejector back" from the molding machine (see pin contact No. 5).</p>
22	<p>Ejector forward enabled Switch closes when the removal operation has been performed far enough for the motion "ejector forward" to be carried out. The switch is closed for the duration of the motion "ejector forward." The signal must be maintained at least until the signal "ejector forward" from the molding machine (see pin contact No. 5).</p>
23	<p>Enable motion of core pullers for removal of injection moldings Switch is closed when the motion of the core pullers is enabled.</p>
24	<p>Enable motion for free travel-in of robot</p>
25	<p>Option: Reserved for SUVA Safety Package: Robot safety door acknowledged outside 1st channel</p>
28	<p>Complete mold opening enabled Switch is closed when the robot has removed the piece from the mold and permits further opening of the mold. The contact must remain closed until the IMM gives the signal "mold open." If the contact is not used, it must remain open. (Option: Reserved for SUVA Safety Package: Robot safety door acknowledged outside Channel 2)</p>
29	<p>reserved for future EUROMAP signal</p>
30,31	<p>free</p>
32	<p>Signal voltage of molding machine</p>

3 Operating Instructions for R8.3 Control

3.1 Basic Operation

3.1.1 Introduction

The R8.3 control unit is a high-performance control system with a teach programming interface for the robot and peripheral equipment.



Before starting up the robot for the first time, please review the manual thoroughly and familiarize yourself with the operation of the robot, to avoid improper use or programming.



Improper use or programming may result in injury to personnel and damage to the robot, gripper, mold or injection molding machine. Programming of the robot must be carried out only by trained personnel who are fully familiar with the operation of the robot.

We offer training sessions at our service centers. Please contact us.

3.1.2 Robot Axes

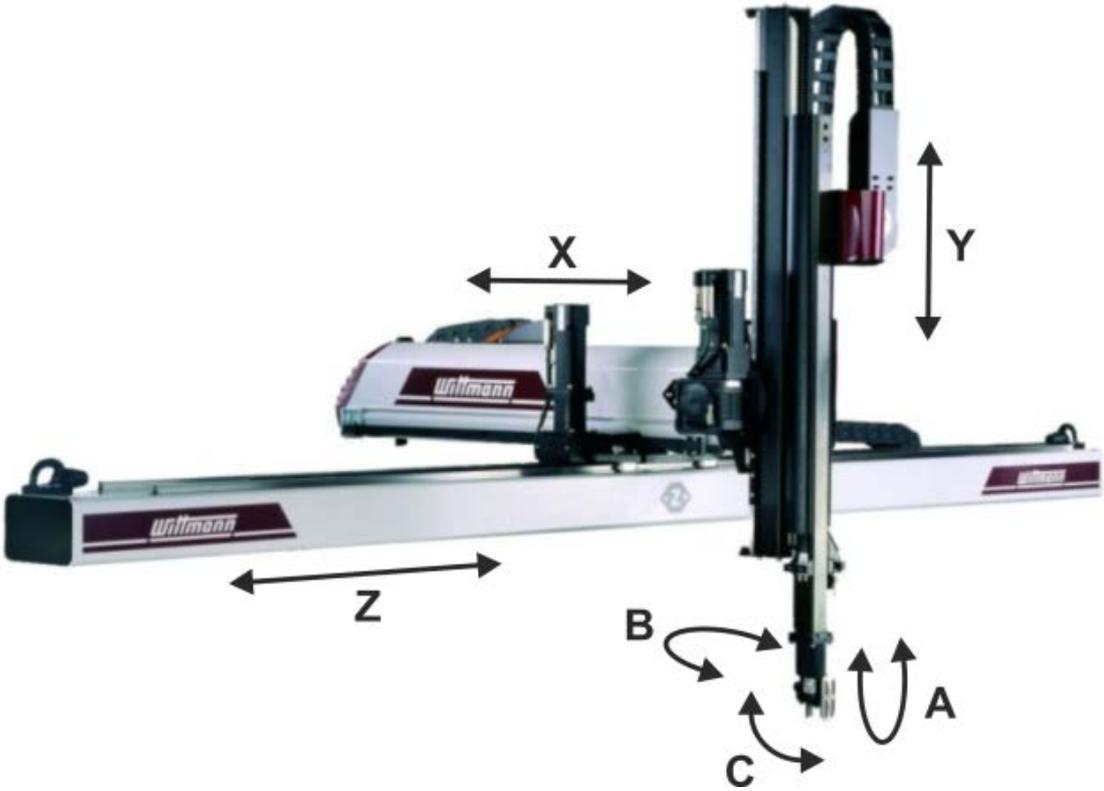


Fig.: Standard Linear Robot

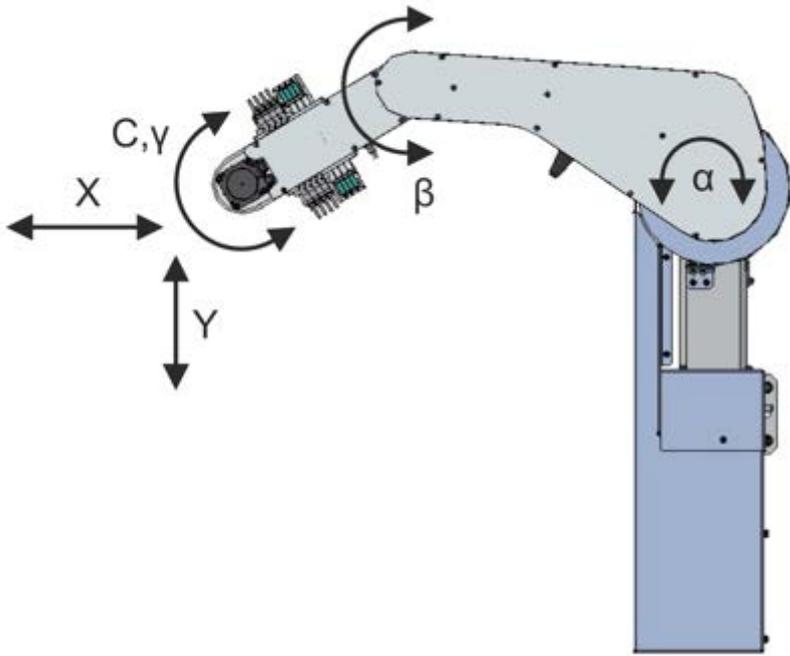


Fig.: Scara Robot

3.1.3 Emergency Stop Block

The buttons on the Emergency Stop Block are used for basic operation of the robot.

This buttons work regardless whether the Teachbox is connected or disconnected. Depending on the mode of operation the LED of the corresponding button is lit.

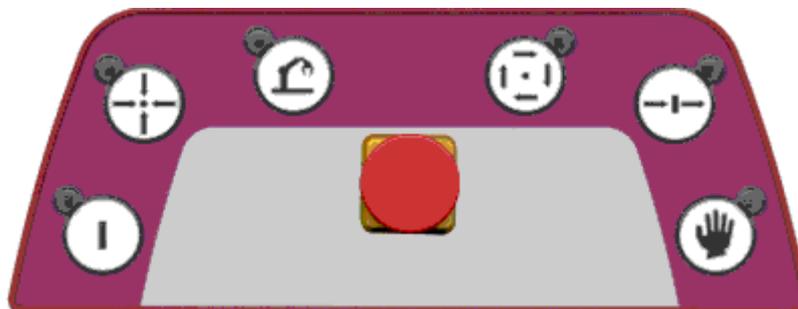


Fig.: Emergency Stop Block

- 

Reference travel
Performing the reference travel. Possible in the mold area only when the loaded teach program contains a reference travel.
- 

IMM: Operation with robot
Enables operation of IMM with robot. In shut-off state the IMM can be operated without robot. The LED will flash, when turned on while mold, ejectors or core pullers are not in a defined position. In this case the function will be turned on with a delay, in order to allow reaching such a position.
- 

Automatic-Start
Start of automatic operation. The robot must be in reference mode, the IMM must be in automatic mode and operation with robot must be on.
- 

Block Stop
Ends automatic operation after complete execution of the currently processed instruction or group of instructions. Automatic operation may be continued by pressing the Automatic Start button again.
- 

Manual operation / Stop
Causes an immediate stop and switches to manual operation.
- 

Control Voltage On
Press to switch on the control voltage after the mains switch has been turned on, or after an emergency stop.
- 

EMERGENCY STOP
Causes the robot to stop immediately. Relevant outputs are disabled through hardware, the EMERGENCY STOP circuit to the IMM is interrupted and the LED of the  button is lit. The error message EMERGENCY STOP appears on the display.

See also: 3.1.15.21 Virtual Sub Pendant

3.1.4 Numerical Input

This window is used to input numbers when necessary.



Fig.: Numerical Input

The title bar displays the name of the parameter. The minimum and maximum values for the parameter are shown left and right of the current value and on the status bar at the bottom of the window.

Input the digits using the numeric keys.

 discards the changes.

 and  move the Cursor to the desired direction.

 deletes from the current cursor position to the right.

 deletes from the current cursor position to the left.

 discards the changes and closes the window.

 confirms the value and closes the window.

3.1.5 Text Input

This window is used to input text when necessary.



Fig.: Text Input

The name of the parameter is shown on the status bar at the bottom of the window.

Input is done using the alphanumeric keys that are positioned like on a typewriter.

 discards the changes.

 and  move the Cursor to the desired direction.

 deletes from the current cursor position to the right.

 deletes from the current cursor position to the left.

 discards the changes and closes the window.

 confirms the value and closes the window.

3.1.6 Operating Modes

Operation of the robot requires knowledge about the various operating modes of the robot.

These divide to the following:



Reference Missing

Status of the control unit after activation. All manual functions are available, but the numerical axis positions shown on the Teachbox display are invalid. In order to be able to use all functions of the robot, a reference travel must be performed.



Manual

All functions of the robot are available; the Teachbox displays valid position values. In order to be able to start automatic operation, a reference travel must be performed.



Reference

All axes, vacuum and gripper circuits are in their respective reference positions. The taught reference programs have been executed. It is possible to start automatic operation.



Automatic

The Teach program loaded to the robot is being executed.



Blockstop

End of automatic operation after complete performance of the currently processed instruction or group of instructions. Automatic operation can be resumed at the next program line by pressing the button



Wait IMM

The robot is waiting for the injection molding machine.

The robot will switch from Automatic to Wait IMM:

- When the machine changes to Manual.
- When the machine is not in Automatic and the robot reaches a command that waits for a signal of the machine.
- When the machine is not in Automatic or Operation With Robot is not on when the robot executes the command WAIT OP. WITH IMM = ON.

The robot will continue automatic operation automatically once the machine changes to Automatic or when Operation With Robot is being switched on.

See also:

3.1.7 Operating Mode Reference

3.1.7 Operating Mode Reference



The Reference travel brings the robot and any peripheral equipment present into a defined home position in order to enable operating mode **Automatic**. The reference positions of the [gripper](#) and [vacuum](#) circuits are selectable.

As long as no reference travel has been successfully executed after powering up the unit, only the operating mode  **Reference Missing** will be available.

Depending on the current operating mode of the robot, two different types of reference travel are executed when the button  is pressed:

- Reference travel in the operating mode Reference Missing
- Reference travel in the operating mode Manual

3.1.7.1 Reference travel in the operating mode Reference Missing

In the operating mode  **Reference Missing** no reference travel can be executed in the mold area. First, the axes have to be moved out of the mold area using [manual functions](#); furthermore the [B-axis](#) (if present) must be moved to its 0-position.

Press and hold the button  to start the initial standard reference travel for the robot.

The axes will be traveled to their reference positions using the sequence **Y-X-Z-C-A**.

Pneumatic axes simply travel to their 0-position limit switch.

Also the **numerical** axes travel to their 0-position limit switch at first. Axes with an incremental encoder then travel to the reference mark of the encoder. After that the respective axis travels to its configured reference position. In standard configuration this is 10.0 mm (0.04") in plus direction from the 0-position limit switch. After reaching this reference position the numerical position of the axis is set to zero.

If the button is released before the reference position of the robot is reached, the movement is stopped immediately.

When you press the button  again, the sequence will continue.

When all axes have reached their reference position, the robot switches to operating mode **Manual**.

The icon  is shown at the status bar in the upper left corner of the display; the LED of the button  is lit.

If no corresponding reference travel (ROBOT-REF) has been programmed, now the [grippers](#) have to be brought to their defined [reference position](#). At last the reference travel in the operating mode Manual has to be carried out, in order to enable automatic operation.

3.1.7.2 Reference travel in the operating mode Manual

Press and hold the button  to start the reference travel.

If reference teach programs for the robot (ROBOT-REF) and the peripheral equipment (PERI-REF) are programmed, at first these sequences are carried out simultaneously (!). This makes it possible to reference the robot from inside the mold area, and to bring peripheral equipment to its home position at the same time. Furthermore the vacuum and gripper circuits can be switched to their reference states by the reference teach programs.

If no reference travel for the robot has been programmed, or if after carrying out the reference teach programs individual axes have not yet reached their 0-position, these axes are moved into their reference position in the sequence Y-X-Z-C-A.

When the button  is released, the movement is stopped immediately. When the button is pressed again, the reference travel will start from the beginning.

When the reference position is obtained, the robot switches to operating mode **Reference**.

The icon  is shown at the status bar in the upper left corner of the display; the LED of the button  is lit.

3.1.8 Override

This function is used to set the travel speeds of the motor axes in connection with the maximum speed (VMAX) set in the teach program.

The key  can be used in every operating mode to open the override window.

Here you can use the sliders to change the Overrides for the numerical axes of the robot.

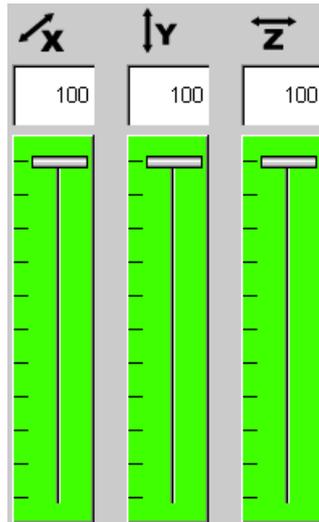


Fig.: Sliders

When the option **locked** is activated with a all sliders are moved together. When the option **locked** is deactivated, the overrides of the axes can be altered individually.

The keys  and  are used to change the Overrides in steps of 1%.

The key  can be used to set the Overrides of all axes to 30% simultaneously.

Use the option [ECO-mode](#) to turn the mode for economic travel on or off.



is used to exit the override window.

- i** Preferably the speeds of the numerical axes should be set in the Teach programm using VMAX and VABS commands. This should happen in a way that the Overrides of the axes can be set to 100% when the robot is executing the standard production cycle.
- i** Depending on the [robot settings](#) the Overrides are automatically reduced to 30%, when:
 - a new teach program is started.
 - the robot is being powered up.

Example:

In the teach program a maximum speed of 80% of the maximum design speed (design Vmax) was programmed for the Z axis (Vmax=80%).

The manual override is set at 50% (of Vmax).

Therefore the drive speed is 40% of the maximum design speed.

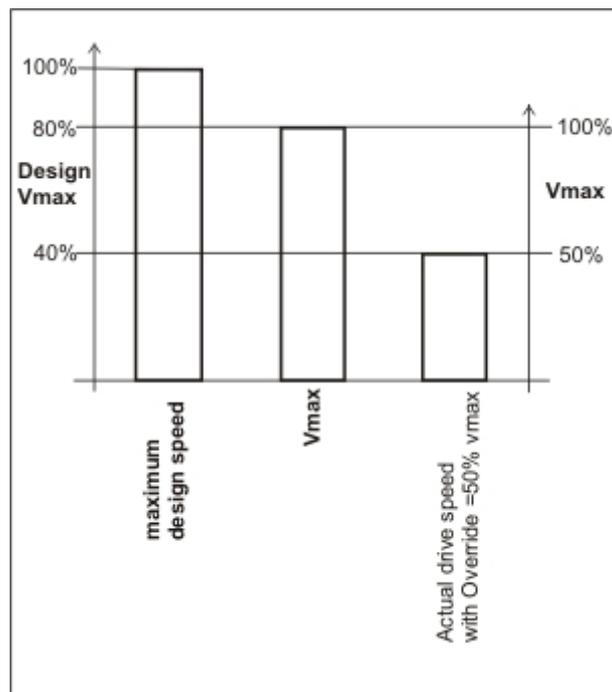


Fig.: VMAX and Override

Refer to: [VMAX](#), [VABS](#)

3.1.9 ECO-Mode

The function ECO-Mode adapts speed and acceleration of the numerical axes while traveling outside the mold in a way that the robot arrives at its waiting position outside the mold at ECO-Buffer time before the next mold opening starts. The movements inside the mold remain unchanged by the ECO-Mode. This way - without changing the overall cycle time - a more economic operation of the robot is achieved, compared to running the robot with full speed and acceleration outside the machine.

You can turn the ECO-Mode on and off in the [Override](#) window, using the option **ECO-Mode**.

With the ECO-Mode turned on, the robot is calculating the optimum setting during the first few cycles after starting automatic mode. Only after this the robot will execute the placing motion with constant low speed and acceleration. The settings made by the ECO-Mode will be shown with the small sliders in the Override window.

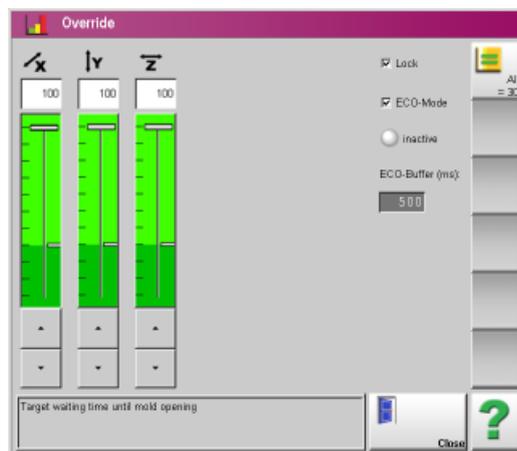


Fig.: Override window with ECO-Mode active

Should the robot out of any reason be too late (mold opens before the robot arrives at the waiting position), the rest of the cycle will be executed with full speed and acceleration. During the next cycles the robot will then retry to find the correct settings.

For alternative routines executing special motions (that take significantly longer or shorter to finish than the standard motion) the ECO-Mode can be deactivated for the current cycle using the Teach command [ECO MODE](#). The robot will then use the settings done with VMAX, VABS, AABS and the Override for the current cycle. The ECO-Mode will be automatically re-activated with the next cycle. Furthermore the ECO-Mode can be turned on by the command automatically and can be reset.

For permanent irregular travel times, the **ECO-Buffer** time should be increased to cover the time difference between the longest and shortest trip, and thus to achieve a better control behavior.



In order for the ECO-Mode to work, please use a compatible sequence to [wait for the open mold](#) or [Smart Removal: In](#).

Refer to: 3.3.9.7 Teach command ECO MODE

3.1.10 Safety System

3.1.10.1 Startup with Safety System

3.1.10.1.1 Manual Setup Mode

In order to be able to travel in the manual mode after the control unit has been activated, the selector switch must be in the MANUAL SETUP position, and one of the permit keys on the bottom of the teachbox must be actuated.

3.1.10.1.2 Automatic Mode, Step Mode, Reference Travel

After the control unit has been activated, safety must be ensured, since otherwise the AUTOMATIC, STEP and REFERENCE TRAVEL modes cannot be used with the robot.

To ensure safety:

- 1.) Open safety door
- 2.) Close safety door
- 3.) Press outside acknowledgment key
- 4.) Set selector switch to position Auto-Step-Reference

The period between opening and acknowledgment outside may be as long as desired (no time monitoring features are in place).

3.1.10.2 Selection of Operating Modes

3.1.10.2.1 Manual Setup Mode

If the selection switch is in the position MANUAL SET UP, each axis may be moved while actuating on of the permit keys on the bottom of the teachbox. If you release the key the selected function will be stopped immediately (software axis stop function; power supply 24 V for outputs will be disconnected through hardware).

During each change of function (e.g., X axis to Y axis) the permit key must be released and then pressed again. In order to enter the teach mode, you must first perform referencing (ensure safety - set selector switch to position Auto - Stepping - Reference). Then call up the teach mode and set the selector switch to position MANUAL SET UP. Now the safety door can be opened and it will be possible to teach the positions inside the protective guarding using the permit key.

3.1.10.2.2 Automatic Mode, Step Mode, Reference

In order to travel to the reference position, all doors must be checked for safety. The selector switch must be set to AUTOMATIC - STEPPING - REFERENCE position. This applies to the automatic and stepping modes as well. Should it be necessary to access

the machine during the automatic cycle, press the  key first in order to unlock the door. The door will be unlocked when all axes have completed their motions. If the door

is not opened, automatic operation can be continued with the  key. However, if the door is opened, safety must be ensured after the intervention and the removal cycle



will be continued with the key. If Safety ensured is shown on the input, the reference position may be approached and automatic operation started. Safety must also be ensured for the STEP mode.

3.1.10.3 Monitoring of the Safety Gate

The safety gate monitoring system is active only when the selector switch is in the AUTOMATIC - STEP - REFERENCE position.

Safety Gate:

The safety gates are electrically monitored by a safety module which ensures that the robot can be used only in manual operation when someone is within the safeguarded area, and that no one can accidentally enter the hazardous area of the plant while the robot is used in the automatic mode. In addition, it is verified that no switches have been bridged or other interventions made in the safety circuits.

3.1.10.4 Error Effect Analysis

- Safety gate is forced open during automatic operation
-> The safety contactor circuit disconnects all outputs from the power supply by means of hardware. At the same time, the automatic cycle is interrupted by the software and shift to manual operation is made.
- The safety gate is opened during the block stop mode
-> The safety contactor circuit disconnects all outputs from the power supply by means of hardware. At the same time, the software does not permit start of a new function. Hence a defective output or a faulty program will not lead to an undesirable motion.
- In the MANUAL SETUP MODE (safety gate not monitored) an axis motion can be executed only with the permit key. If this key is pressed for a longer period (30 sec.) without a motion being performed, no new travel instruction may be executed (by the software). The entire system will go into a failure state (time monitoring permit key).
- In addition, in an unsafe state all motor axes will be disconnected by means of hardware from the frequency converter/servo module feeder line through contactor disconnects.

3.1.11 User Administration and Passwords

User administration makes it possible to assign different rights of access to fifteen different users for operation of the robot. For example, operators may be authorized only for robot operation, shift supervisors for manual functions as well, while machine programmers could also be allowed to load and change programs, and so forth.

User's rights may be freely specified in 8 different user profiles. Profile 7 defines what can be accessed without entering a password. The eighth profile (profile 0) is permanently defined and is reserved for the administrator (Admin) of the system. The administrator is assigned all rights; as sole user authorized to do so, he may also create profiles and change passwords. Each user receives a personal password and is assigned one of the user profiles 1 to 6.



The user administration of the R8.3 control is always active, and cannot be deactivated. In order to enable operation without a password, enable read/write access for all functions in profile 7. Thus only the settings reserved for the Administrator will require a login.



Write down the passwords, especially the administrator password and keep them in a suitable safe place.



If you lose the administrator password the functions reserved for the administrator (Changes in the password system and several other base settings) will be no longer accessible! In this case please contact our nearest service center.

3.1.11.1 Administrator

In order to be able to manage users with their profiles and passwords, and for some other base settings you will have to [login](#) as administrator.

The user name for the administrator is always **ADMIN**.
The password for the administrator is **1234** at delivery.



For real use of the user administration the password of the administrator must be changed in order to avoid unauthorized access.

3.1.11.2 Users

Creating, changing and deleting of users is only possible if you [login](#) as administrator.

In the menu **Init** select **User Administration**, and switch to the tab **Users**.



Here you can assign [User Profiles](#) to existing users by activating the desired option in the line with the user by tapping the option.



is used to [create](#) a new user.



is used to delete the highlighted user.



is used to [edit](#) the highlighted user



discards the entries and closes the window.



confirms the new user data.

3.1.11.2.1 User (create, change)

Here you can create a user or change the data of a user.



Fig.: Window to edit a user

After tapping the respective input field, the desired value can be entered using the dialogue for [text input](#).



Cancel

discards the entries and closes the window.



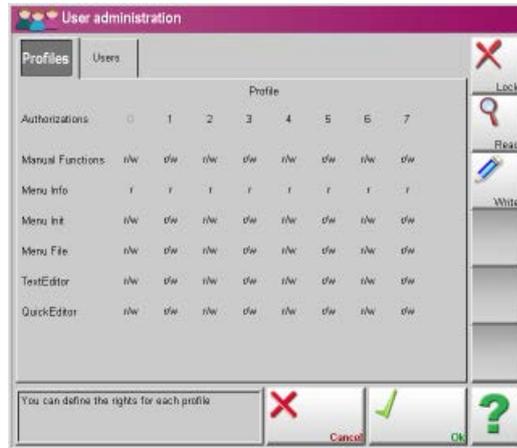
Ok

confirms the new profile data.

3.1.11.3 Profiles

You have to [login](#) as administrator to access the user profiles.

In the menu **Init** select **User Administration** and switch to the tab **Profiles**.



Here you can set up user profiles 1 to 7 to suit your needs.



is used to lock the highlighted function in the respective profile. This will be indicated by an **x** in the table.



is used to give a "read only" permission for the highlighted function in the respective profile. This will be indicated by an **r** in the table.



is used to give a "read and write" permission for the highlighted function in the respective profile. This will be indicated by an **r/w** in the table.



discards the entries and closes the window.



confirms the new profile data.



Profile 7 is used to permit functions for use **without password!**



If a "read and write" permission is given for the Text Editor, automatically a "read and write" permission is given for the Manual Functions.

3.1.11.4 Login

With user administration activated in the logged out state (e.g. after powering up the unit) only the buttons to change the mode of operation, the help function and the functions activated in profile 7 are available. All other functions are deactivated.

The icon  will be displayed in the status bar.

Each user has to login with name and password to access the functions attached to his profile. Tap the menu **Init** and there **Login** to do so.

You then can login using the following window:



Enter your user name after tapping the input field **Insert Username** using the dialogue for [Text Input](#).

Enter your user password after tapping the input field **Insert User Password** using the dialogue for [Text Input](#).



discards the entries and closes the window.



will confirm the entries and log you in.

3.1.11.5 Logout

With user administration activated, one should logout when leaving the Teachbox, in order to avoid misuse of the appliance.

Tap the menu **Init** and **Logout** to do so.

The Icon  will be displayed in the status bar.

Only the buttons to change the mode of operation, the help function and the functions activated in profile 7 will be available. All other functions will be deactivated.

3.1.11.6 Forgot Password

In case you forgot the administrator password, you can press the button **Forgot Password** in the [Login](#) window to get the passwords encrypted to a code:

Please contact your administrator, or call our local agency, pass the following number and you will get your password.
2816 2816 2816 2816 2816 2816 2816 2816 2816 2816 039

With this code you can contact our local service center to help you.

3.1.12 Manual Functions

3.1.12.1 Numerical Axes



This function is used to manually move the selected numerical axis in the operating modes **Manual**, **Reference** and **Reference Missing**. Furthermore the current position of the axis is displayed in all operating modes.

At robots with secondary axes press the axis button longer, if you want to display the buttons for toggling between main axis and secondary axis.

Whether the numerical axes can be moved using variable or fixed manual speeds is being set in the Init Menu [Setup: Robot Settings](#).

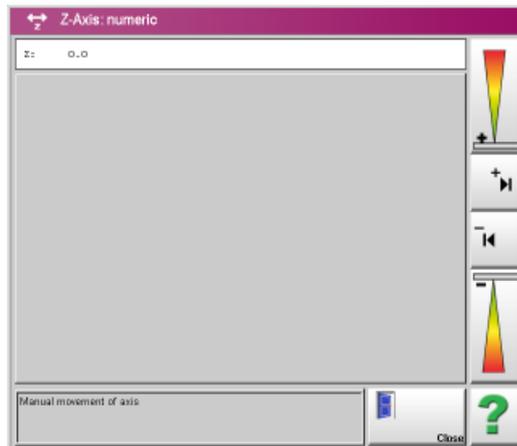


Fig.: Numerical axis with variable manual speed

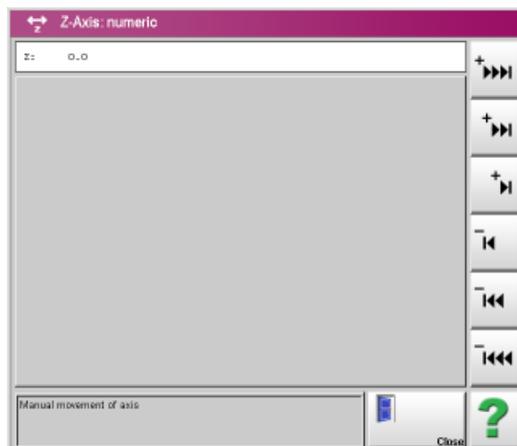


Fig.: Numerical axis with fixed manual speeds



moves the axis into plus direction with variable speed.



moves the axis into plus direction with high manual speed.



moves the axis into plus direction with low manual speed.



moves the axis about 0,1mm into plus direction.



moves the axis about 0,1mm into minus direction.



moves the axis into minus direction with low manual speed.



moves the axis into minus direction with high manual speed.



moves the axis into minus direction with variable speed.



is used to close the window.

3.1.12.2 Axis Release in Manual Mode (optional)

When horizontal, numerical axes are equipped with the optional feature of Axis Release, it can also be activated in operating mode Manual. Releasing an axes in manual mode will open the brakes of the respective axes, which results in the possibility to shift those axes by hand.

Activation is done in the View Menu **Axes Positions**:

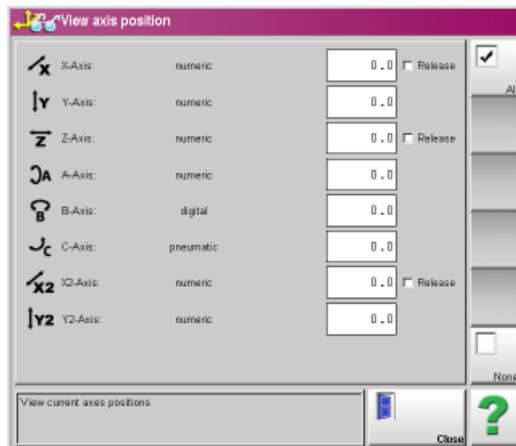


Fig.: View Menu Axes Positions with Release

With a at the option **Release** the respective axis will be released immediately.

The release can be switched off manually tapping the option **Release** again. Additionally the release will be switched off automatically, if any other drive command is sent to the axis.



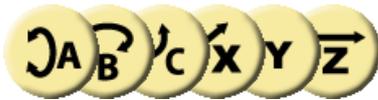
is used to release all releasable axes.



is used to deactivate all releases.

See also: 3.3.1.3.7 Teach command Axis Release

3.1.12.3 Pneumatic Axes



This function is used to manually move the selected pneumatic axis in the operating modes **Manual**, **Reference** and **Reference Missing**. Furthermore the current position of the axis is displayed in all operating modes.

At robots with secondary axes press the axis button longer, if you want to display the buttons for toggling between main axis and secondary axis.

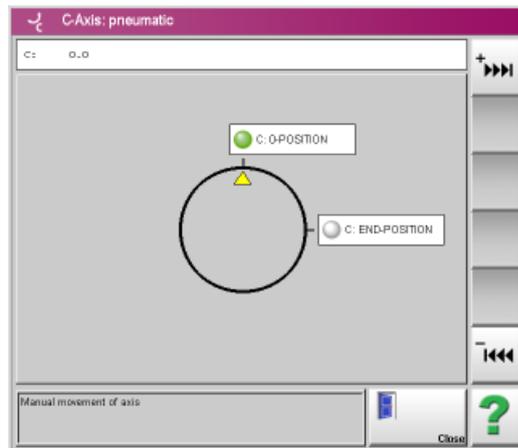


Fig.: Window for pneumatic axes

moves the axis directly to the endposition in plus direction.

moves the axis to the next intermediate position in plus direction.

moves the axis to the next intermediate position in minus direction.

moves the axis directly to the endposition in minus direction.

The number of positions depends on the type and equipment of the selected axis.

is used to close the window.

3.1.12.4 Digital (B-) Axes



This function is used to manually move the selected digital axis in the operating modes **Manual**, **Reference** and **Reference Missing**. Furthermore the current position of the axis is displayed in all operating modes.

At robots with secondary axes press the axis button longer, if you want to display the buttons for toggling between main axis and secondary axis.

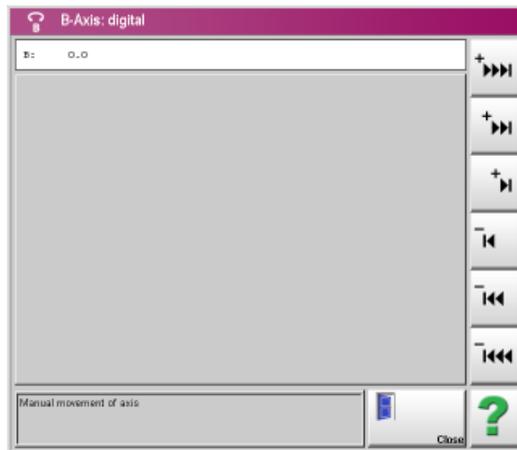


Fig.: Window for digital axes

moves the axis 90° into plus direction.

moves the axis 60° into plus direction.

moves the axis 15° into plus direction.

moves the axis 15° into minus direction.

moves the axis 60° into minus direction.

moves the axis 90° into minus direction.

When traveling a digital axis, the respective button should be held until the control unit has finished the selected movement. Otherwise the axis may stop between two positions and cause an error message.

is used to close the window.

3.1.12.5 Counters



This function is used to display, reset or set the 64 counters of the Teach program in any mode of operation.

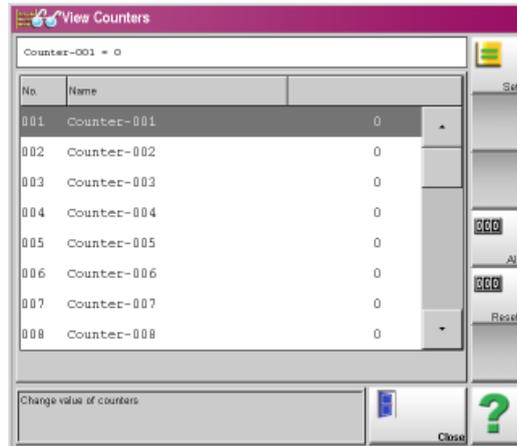


Fig.: Counter window



is used to set the selected counter to a desired value. The numeric value is entered using the dialogue for [numerical input](#).



is used to reset the selected counter to zero.



is used to reset all 64 counters to zero.



Depending upon the application of the counter in the teach program, modifying counters (**especially in automatic mode**) can result in serious changes in execution of the program, as well as in damage due to collisions! Therefore, counters should (especially in automatic mode) be modified only with caution by persons who are familiar with the effects of such modification on the teach program.



is used to close the window.

3.1.12.6 Placing Counters



This function is used to display the current values of the counters of the 16 placing programs of the teach program in all operating modes. In operating modes



Manual,



Reference and



Reference Missing it is furthermore possible to reset the counters to zero.

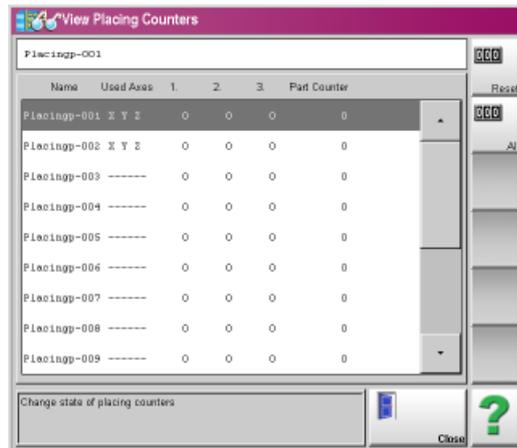


Fig.: Window for placing counters

Name

Shows the name of the respective placing program.

Used Axes

This shows which axes are used in the respective placing program.

1. - 2. - 3.

This shows the current values of the placing counters of the respective placing program.

Part Counter

This shows the current value of the part counter of the respective placing program.



is used to reset the counters of the selected placing program.

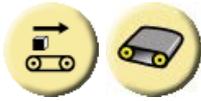


is used to reset the counters of all 16 placing programs.



is used to close the window.

3.1.12.7 Conveyors



This function is used to manually switch the conveyors during operating modes **Manual**, **Reference** and **Reference Missing**. Furthermore the current states of the conveyors are displayed in all operating modes.

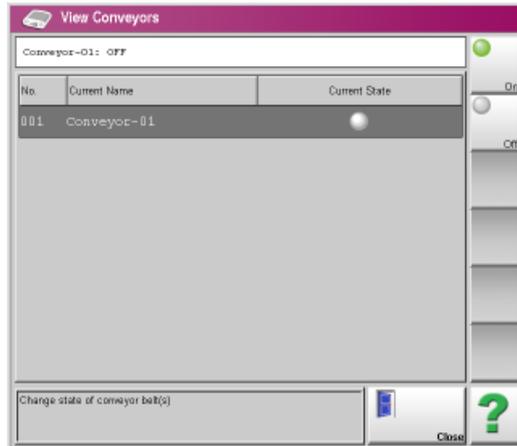


Fig.: Window for conveyor belts

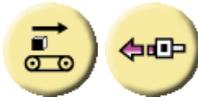
- means the respective conveyor is OFF.
- means the respective conveyor is ON.

is used to switch the selected conveyor on.

is used to switch the selected conveyor off.

is used to close the window.

3.1.12.8 Peripheral Outputs



This function is used to manually switch the peripheral outputs during operating modes **Manual**, **Reference** and **Reference Missing**. Furthermore the current states of the peripheral outputs are displayed in all operating modes.

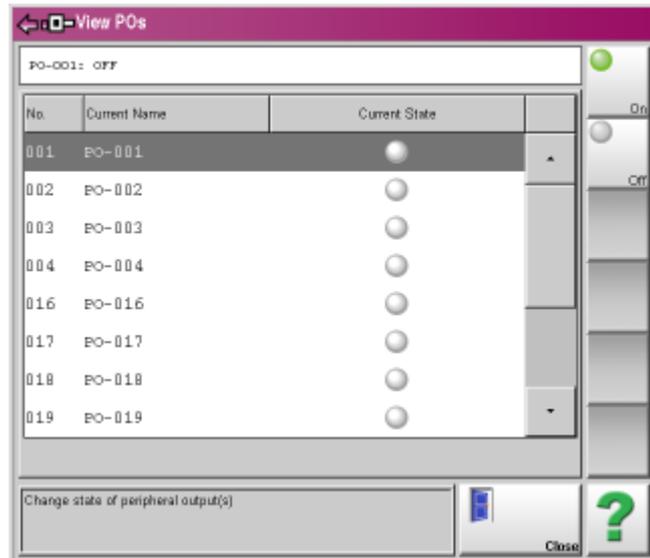


Fig.: Window for peripheral outputs

means the respective peripheral output is OFF (LOW).

means the respective peripheral output is ON (HIGH).



is used to switch the selected peripheral output on.



is used to switch the selected peripheral output off.



is used to close the window.

3.1.12.9 Vacuums



This function is used to manually switch the vacuum circuits during operating modes



Manual,



Reference and



Reference Missing. Furthermore the current states of the vacuum circuits are displayed in all operating modes.

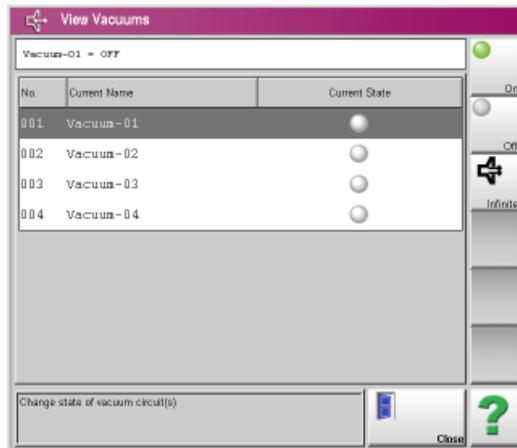


Fig.: Window for vacuum circuits

 means the respective vacuum is OFF.

 means the respective vacuum is ON.

 means the respective vacuum is blowing off (optional).

 **On** is used to switch the selected vacuum on.

 **Off** is used to switch the selected vacuum off.

 **Infinite** is used to switch the selected vacuum to blow off (optional).

 **Close** is used to close the window.

3.1.12.10 Grippers



This function is used to manually switch the grippers during operating modes

Manual, **Reference** and **Reference Missing**. Furthermore the current states of the grippers are displayed in all operating modes.

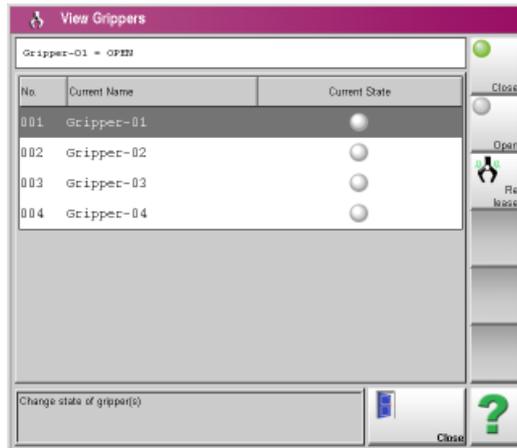


Fig.: Window for grippers

- means the respective gripper is OPEN.
- means the respective gripper is CLOSED.
- indicates that the gripper is RELEASED.
- indicates that the state of the gripper is UNKOWN.

is used to switch the selected gripper on.

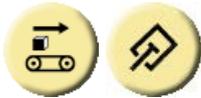
is used to switch the selected gripper off.

is used to release the selected gripper.

The state **RELEASED** is only available at gripper circuits equipped with the required valves.

is used to close the window.

3.1.12.11 Cylinders



This function is used to manually switch the Cylinders defined within the teach program during operating modes **Manual**, **Reference** and **Reference Missing**. Furthermore the current states of the cylinders are displayed in all operating modes.

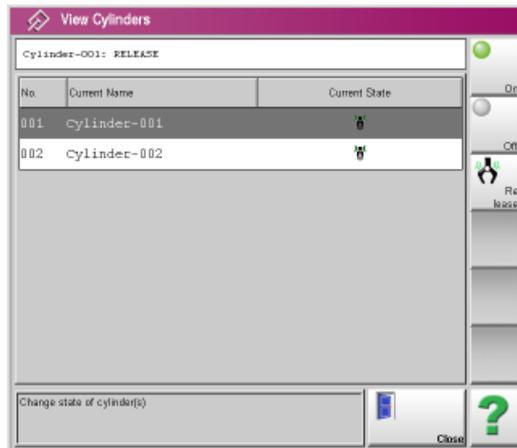


Fig.: Window for cylinders

- indicates that the cylinder is in OFF position.
- indicates that the cylinder is in ON position.
- indicates that the cylinder is RELEASED.
- indicates that the state of the cylinder is UNKOWN.

is used to switch the selected cylinder to its ON position.

is used to switch the selected cylinder to its OFF position.

is used to release the selected cylinder.

is used to close the window.

3.1.13 FILE Menu of the main screen

3.1.13.1 Properties

Here the properties of the Teach program of the robot are displayed. That is the program which is executed when the robot is in automatic operation.

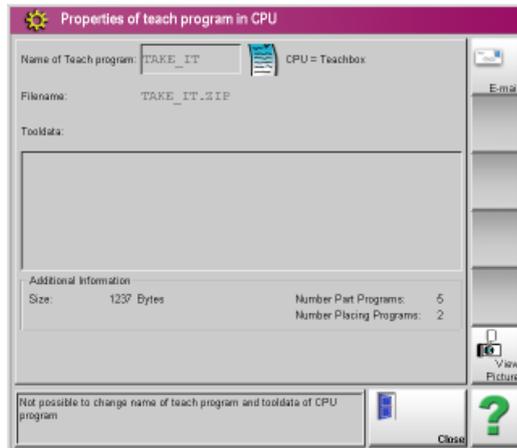


Fig.: Window for properties of the CPU program

Name of Teach program

Shows the name of the teach program of the robot. This name is not necessarily the same as the file name of the program.

CPU = Teachbox or CPU <> Teachbox

Displays whether the teach programs in the robot (CPU) and the Teachbox are equal or unequal.

Tooldata

Displays the Tooldata text of the teach program of the robot. The text for Tooldata can only be edited in the properties window of the **text editor**.

Size

Displays the size of the teach program in the robot. The maximum size of a teach program is limited to 32.768 bytes.

Number Part Programs

Shows the number of part programs of the teach program in the robot: ROBOT-PRG, PERI-PRG, ROBOT-REF, PERI-REF, ALLMODE-PRG and up to 120 additional part programs.

Number Placing Programs

Shows the number of placing programs used in the teach program in the robot. The maximum number of placing programs available in a teach program is limited to 16.

If a picture has been associated with the selected teach program,  can be used to open the [picture viewer](#).

 can be used to email the Teach program as an email attachment (*.wip, *.htm and *.txt file) using the [manual email function](#) of the robot.

 is used to close the window.

3.1.13.2 Backup stored Teachprograms

The R8.3 control gives you the possibility to store Teachprograms on the internal storage medium of the robot control unit. Nevertheless you should regularly backup the stored programs externally, in order to avoid data loss.

Everything you store on the internal storage medium of the robot will end up in the folder C:\PUBLIC and its subfolders. Therefore use the backup function for the public folder to backup your data on an USB stick.

Proceed like this:

- 1) Plug in your backup-USB-stick to the Teachbox.
- 2) [Login](#) as [Administrator](#).

- 3) Open the menu **File – System Backup/Restore** and select the tab .

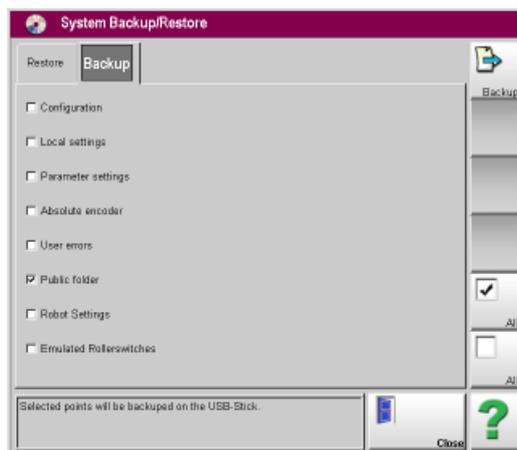


Fig.: Selection for backup of the Public folder

- 4) Activate **only** the option **Public folder** and press .
- 5) Answer the questions about overwriting. At the end of the backup process you will have the possibility to display a report about the copied files.

- 6) Close the window with .

- 7) [Logout](#) as [Administrator](#).



Storage location for the backed up files on the USB stick is the folder **Backup**. Each robot will add its own subfolder inside that folder. The name of the subfolder will consist of the respective robot type and serial number using the format **Wxxx-xxxx**. Thus you can use one USB stick to backup data from several robots.

3.1.14 INIT Menu

The INIT menu is used to adjust robot settings. Some settings can be accessed by the logged-in administrator (Admin) only.

3.1.14.1 Stroke Limits

The stroke limits are software limit switches to limit the maximum travel range of an axis.

In operating mode  **Reference Missing** the stroke limits are not active, the axes travel is limited by the hardware limit switches. If a stroke limit is reached in manual mode, the robot will stop at the position value of the stroke limit without an error message. In automatic mode the error message 004-007 STROKE LIMIT will be shown.

Axes commands in a Teachprogram will be checked already during transfer to the CPU. If an axis command targets a position outside the stroke limits, the error message 160-005 TARGET OUT OF STROKE LIMITS including the affected line number and axis will be shown.

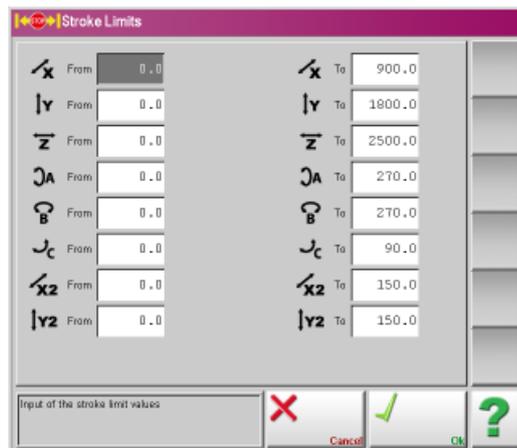


Fig.: Window for stroke limits

After tapping the respective input field, the desired axis value can be input using the dialogue for [numerical input](#). The title bar of the dialog box will show which axis value is to be entered while doing so.



discards the entries and closes the windows.



changes the stroke limits to the entered values.



The respective stroke limit of a numerical axis will be set automatically, when a position limit switch is approached while the robot is not in operating mode  **Reference Missing**.

3.1.14.2 Roller Switches

The roller switches are used for mold monitoring.

Refer to chapter 2.3.3 Mold Monitoring S5/S6

3.1.14.3 Absolute Encoders (Admin only)

The menu **Init – Absolute** encoders serves for setting the zero points of axes with absolute encoders.

Setting the zero points of axes with absolute encoders will be necessary at initial startup in the factory; after installing a new drive belt, motor, gearbox, etc.; and after a battery failure of the Sanyo R drive.



Caution! Changing the zero points will lead to severe danger of collisions, because all positions in all Teachprograms of the robot will be invalid!

See also:

6.5.2 Battery replacement at Sanyo R inverters.

3.1.14.4 Deactivate Monitoring

This function is only available for [users](#) with [write permission](#) for the text editor.

In the selection window **Init / Deactivate monitoring** you can disable various monitoring functions temporarily. Thus you can manually move the robot out of hazardous areas, after one or more of the monitoring functions were triggered.

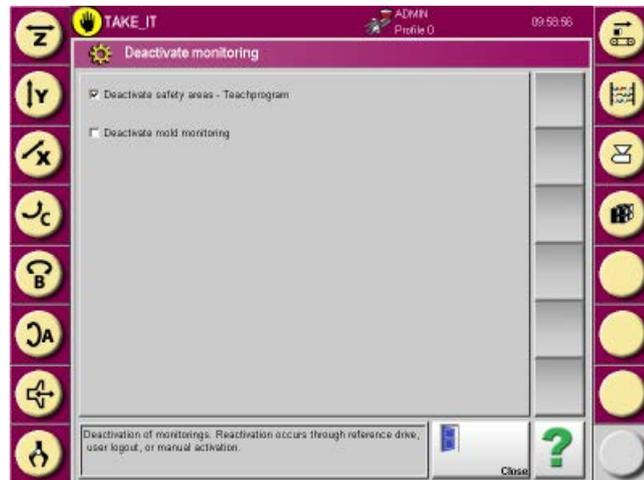


Fig.: Selection window for Deactivate Monitoring

The following monitoring functions can be deactivated:

- Mold Monitoring
- Safety Areas (if present)
- Collision Monitoring (if present)

The selection window only shows monitoring functions which are currently present.

A deactivates the respective monitoring.

When monitoring functions are deactivated, the icon for the mode of operation changes its color to yellow.

The monitoring functions remain deactivated until one of the following events:

- A reference travel is executed.
- The users logs out or is being logged out.
- The monitoring functions are re-activated manually.

Press  to close the window.

3.1.14.5 Maintenance - Lubrication Interval

This function was introduced in order to call the user's attention to regular lubrication of the guides.

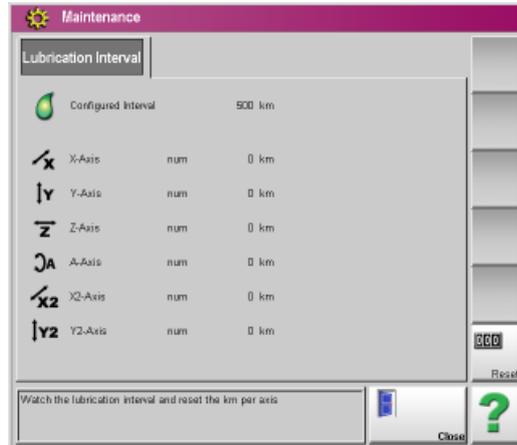


Fig.: Window for lubrication

In the line **Configured Interval** the configured lubrication interval in kilometers is shown.

At the axes the distances traveled since the last lubrication are shown.

If the distance traveled by one axis exceeds the configured interval, the symbol  will be shown in the status bar, to remind that lubrication is needed.

After the robot has been lubricated, the traveled distances can be reset to zero using

the button .

 is used to close the window.

3.1.14.6 Setup: Robot Settings

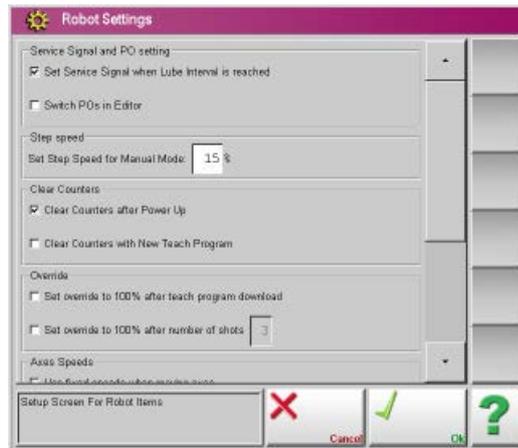


Fig.: Window for robot settings

Set Service Signal when Lubrication Interval is reached

This is used to configure whether or not an output is to be switched on once the lubricating interval has been exceeded. Any desired signal transmitter (horn, lamp, etc.) may be connected to this output.

Switch POs in Editor On

This function is used to configure whether or not peripheral outputs and conveyors are turned on or off during teaching when the corresponding function is programmed. I.e., if this option is activated and the command to turn on peripheral output 2 is entered in the editor, output 2 will in fact be turned on **immediately**. This will not happen if the option Switch POs in Editor On is deactivated.

Step speed

Speed setting in percent of the configured maximum axis speed used for axes movements in step mode and while executing programmed reference travels. After tapping the input field the desired value can be entered using the dialogue for [numerical input](#). Value range is from 1 to 30%.

Clear Counters

This is to determine when the counters and placing programs of the teach programs are to be cleared automatically. If the option Clear Counters with New Teach Program is deactivated, the request Clear all Placing programs and Counters? is shown every time a new or changed teach program is being transferred.

Override

The two options are to determine the automatic behavior of the Override. When none of the options is activated, the Override is automatically reduced to 30% when a new teach program is started or the robot is being powered up.

Axes Speeds

This is to determine if numerical axes use variable or fixed speeds for manual movements.

Units

This is to determine if distances and positions are displayed in Millimeters or Inches.

End-Of-Arm-Tooling by RF-Id

Use this option to determine the behavior when the optional RF-ID End-Of-Arm-Tooling recognition loads teach programs automatically.

Show mouse cursor

Use this option to display a mouse pointer at the coordinates of the last registered touch at the touch screen.



discards the entries and closes the window.



confirms the entries and closes the window.

3.1.14.7 Setup: Date / Time

Here you can set date and time.

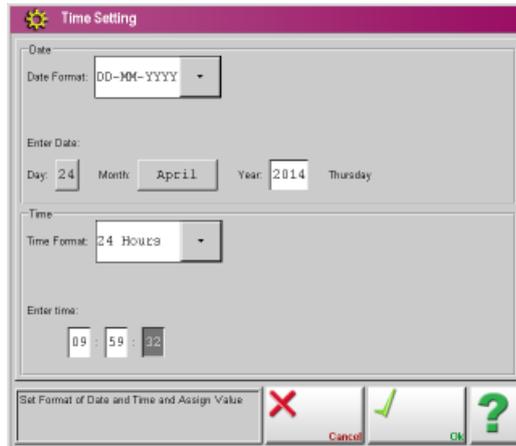


Fig.: Window for date and time settings



discards the entries and closes the window.



confirms the entries and closes the window.

3.1.14.8 Setup: Language

This is to change the display language of the Teachbox.



Fig.: Window for language setting

First select the desired language from the list by tapping it.

Then press  to load the selected language.

During the loading process a window with a progress bar is shown.



is used to close the window without changing the language.

3.1.14.9 Setup: Network Settings (Admin only)

This is to do the settings for the network connection of a robot.

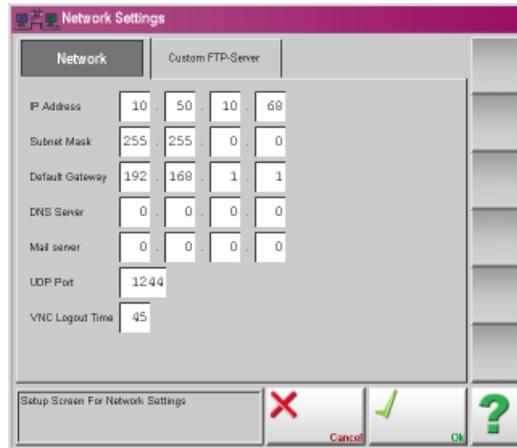


Fig.: Window for network settings

IP-Address

Distinct, static network address of the robot.

Subnet Mask

Mask for limiting the number of participants in the local network.

Default Gateway

Address for establishing connection with participants outside the own local network segment.

DNS Server

Address of the server for name resolution.

Mailserver

Address of the server handling email traffic.

UDP Port

Local port for UDP traffic. Use the standard setting.

VNC Logout Time

Time of inactivity after which (remote-) users automatically release the Teachbox access.

The values for IP-Address, Subnet Mask, Default Gateway, DNS Server and Mailserver have to be assigned by the network administrator.

After tapping the respective input field use the dialogue for [numerical input](#) to enter the desired value.



discards the entries and closes the window.



confirms the entries and closes the window.



The current network settings can be viewed in View / [Operating Systems](#) also.

3.1.14.10 Setup: Remote Users Settings (Admin only)

This is to do the settings necessary for VNC remote users access.

Using VNC (Virtual Network Computing) and a VNC viewer software the Teachbox can be displayed on a remote PC via a network and the robot being remote controlled respectively. At integrated solutions with injection molding machines VNC is used to enable accessing the functionalities of the Teachbox also on the panel of the injection molding machine.

For using this function it is required that the robot is attached to a network. Furthermore the **Network Settings** must be correct and **Licenses** for VNC access must be available.

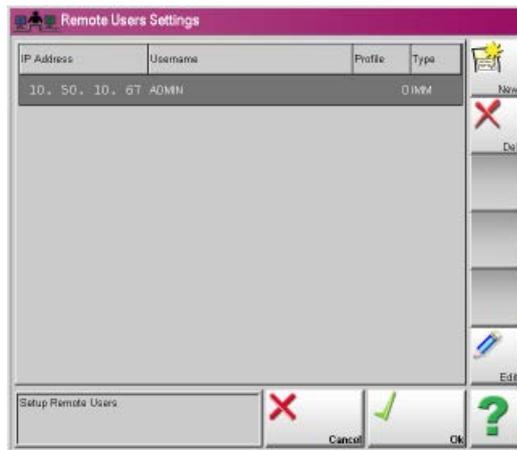


Fig.: Window for remote users settings

Press  to open the window for creating a new remote user.

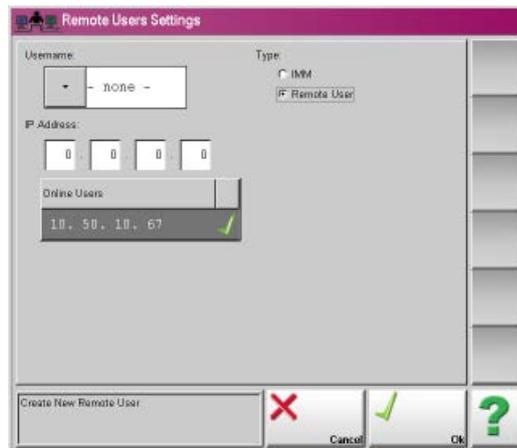
Press  to open the window for editing the selected remote user.

Press  to delete the selected remote user.

 discards the entries and closes the window.

 confirms the entries and closes the window.

3.1.14.10.1 Create / Edit Remote Users (Admin only)



Type

A remote user of the type IMM can access all functions of the Teachbox, if his assigned local user has the respective permissions. This is used for IMM integrations. Also a remote user of the type Remote User can only access functions, his locally assigned user has permissions for – additionally he can neither use manual functions nor perform changes of the operating mode. The type IMM is only available at integrated solutions.

Username

Here you choose a user from the local User Administration to be used for the remote user. The remote user will always be automatically logged in with the user name of the selected user, and will have the respective permissions.

IP-Address

The IP-Address of the remote computer of the remote user. The remote user will be identified with his IP-Address, and will be given access with that address only.

Online Users

This displays a list of IP-addresses of remote computers that are requesting VNC access successfully or unsuccessfully.



discards the entries and closes the window.



confirms the entries and closes the window.

3.1.14.11 Licenses (Admin only)

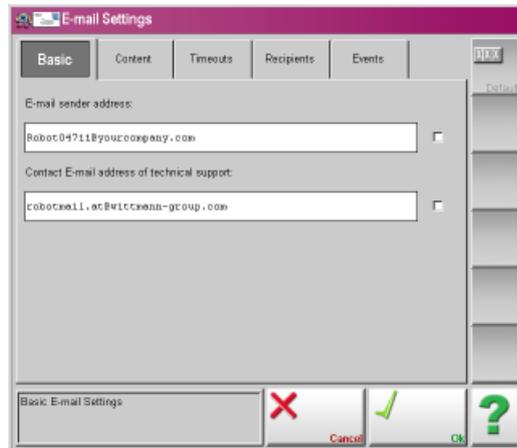
This window displays the number of available VNC licenses for remote users. Furthermore this is where you activate additionally acquired licenses.

In case you should need additional licenses, please contact our service office responsible for you.

3.1.14.12 Setup: Email Settings

This is to do the settings for the manual and automatic [email functions](#) of a robot.

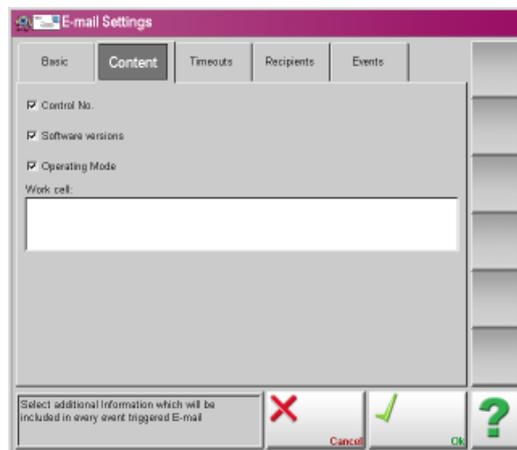
3.1.14.12.1 Tab Basic



Here you input the **E-mail sender address** for the robot and the **contact email address** of our **technical support** facility responsible for you, using the dialogue for [text input](#).

After selecting the desired entries with a ✓ you can press  to use the default addresses.

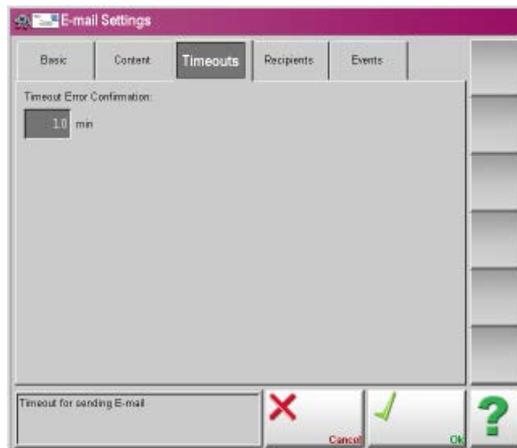
3.1.14.12.2 Tab Content



Select the content to be included in [automatically generated emails](#).

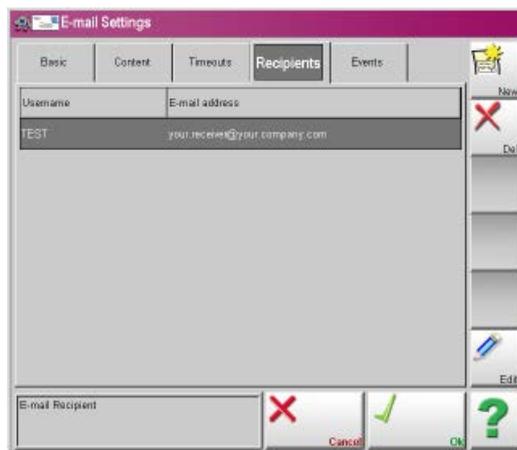
You can add description text for the **work cell** of the robot using the dialogue for [text input](#) after tapping the input field.

3.1.14.12.3 Tab Timeouts



Set the timeout for the [automatic email function](#). In case of the robot stopping with an error message (switching from automatic mode to manual mode) and nobody confirming the error message(s) within this timeout, the robot will send emails with the selected content to the selected recipients.

3.1.14.12.4 Tab Recipients



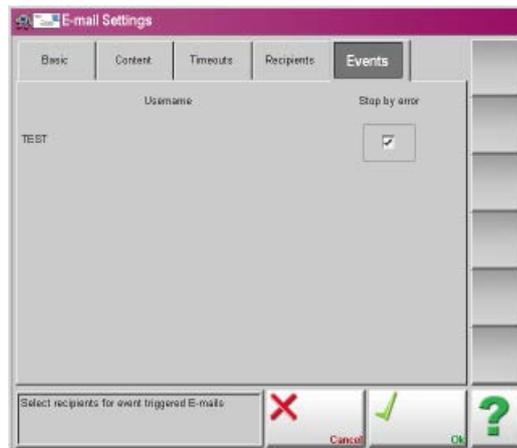
Create, change or delete possible recipients for the [automatic email function](#).

Press  to open the window for creating a new recipient.

Press  to open the window for editing the selected recipient.

Press  to delete the selected recipient.

3.1.14.12.5 Tab Events



Select the recipients to sent an email to with the [automatic email function](#). You can create additional recipients on the tab **Recipients**.

3.1.14.13 Setup: Screensaver Properties

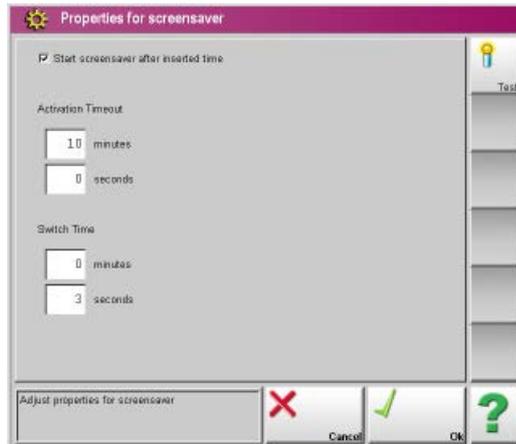


Fig.: Window for the properties of the screensaver

Start screensaver after inserted time

This is used to activate or deactivate the screensaver.

Activation Timeout

This is the time to pass after the last touching of the touchscreen, before the screensaver starts. After touching either the minutes or seconds field, values are input using the dialogue for [numerical input](#).

Switch Time

This is the time the screensaver uses to show each picture. After touching either the minutes or seconds field, values are input using the dialogue for [numerical input](#).



is used to start the screensaver immediately for testing purposes.



discards the entries and closes the window.



confirms the entries and closes the window.

3.1.14.14 Setup: Calibrate Touch

This function is used to calibrate the touchscreen of the Teachbox. This is necessary for the Teachbox to correctly calculate digital X/Y coordinates from the two analogue resistance values supplied by the touchscreen. Only then it is possible for the Teachbox to react correctly when the touchscreen is tapped on.



Calibrating the touchscreen is only necessary when pressed buttons are not interpreted correctly.

After calling the function please follow the instructions on the screen. Tap the center of the displayed crosses as exactly as possible.

3.1.15 View Menu

The VIEW menu is used to display information out of the robot control unit.

3.1.15.1 Operating Systems

This window displays the versions of software loaded to the robot and the [network settings](#).

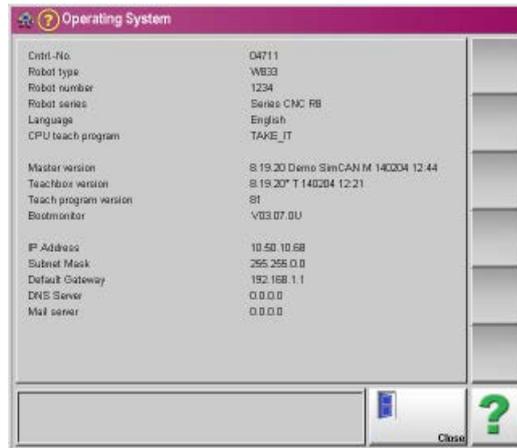


Fig.: Window for Operating Systems



Close is used to close the window.

3.1.15.2 Operating Data

This window displays operational data of the robot.

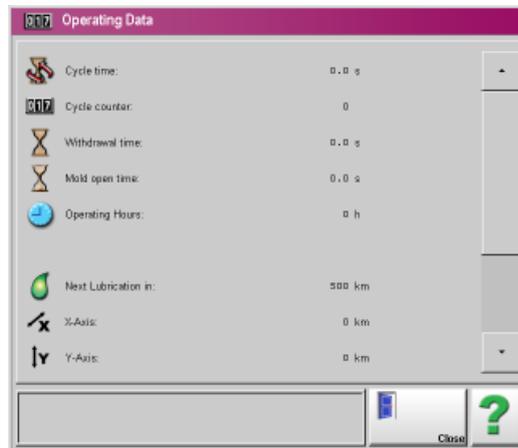


Fig.: Window for Operating Data

Cycletime

Displays the time between the last but one and the last positive slope of the signal "Mold Open" from the IMM. This is the time needed to produce one part (round).

Cyclecounter

The cycle counter counts the positive slopes of the signal "Mold Open" from the IMM during automatic operation of the robot. The cycle counter is only reset by using the boot medium.

Withdrawaltime

Displays the time between the positive slope of the signal "Ejector is forward" of the IMM and the moment when the robot is travelling onto the bar of the S5 rollerswitch of the Y-arm. This is the time the robot needs to get out of the mold with the part on the gripper.

Mold open time

Displays the time between the positive slope of the signal "Mold Open" of the IMM and the positive slope of the permit signal "Close mold" of the robot to the IMM. This is the time the IMM has to wait for the robot before it can mold the next part.

Operating Hours

Displays the overall time the robot worked in automatic mode. The operating hours are only reset by using the boot medium.

Lubrication

In the first line the [lubrication interval](#) is displayed, in the following lines the kilometers travelled by the linear axes since the last lubrication are shown.



is used to close the window.

3.1.15.3 Axes positions

This window displays the current positions of the robot axes and their drive type.

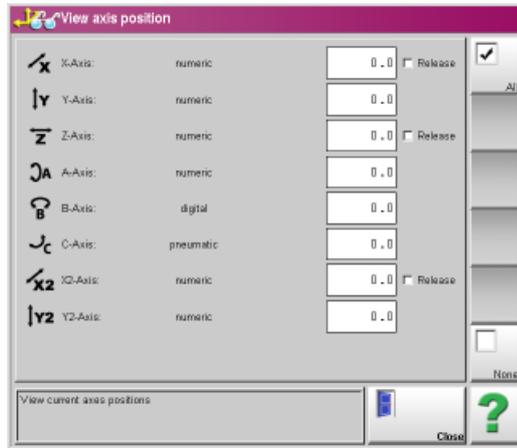


Fig.: Window for axes positions

The number and type of the axes shown is depending on the configuration of the actual robot.

The positions are shown in millimeters for linear axes and in degrees for rotary axes. The resolution of the values displayed is always one tenth.



is used to close the window.

See also: 3.1.12.2 Axis Release in Manual Mode

3.1.15.4 I/O

This window displays the current states of the inputs and outputs of the robot control unit.

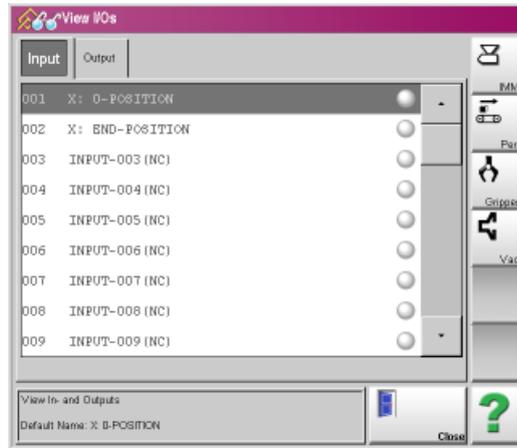


Fig.: Window for inputs and outputs

Tap one of the tabs **Input** or **Output** to select if either the inputs or the outputs are to be shown.



All 512 theoretically possible inputs and outputs of a robot are shown. If any input or output is displayed in this window, it does not indicate that this input or output is really existent on the actual robot. If (NC) is displayed right next to the name, the respective input or output is **not** configured on the current robot.

-  indicates that the input/output is OFF (LOW).
-  indicates that the input/output is ON (HIGH).

The four buttons on the right are shortcuts to certain function groups:

 inputs and outputs of the IMM interface

 peripheral inputs and peripheral outputs

 inputs and outputs of the grippers

 inputs and outputs of the vacuum circuits

 is used to close the window.

3.1.15.5 Safety Areas

Here you can view the settings of the current safety areas.



Fig.: View Menu safety areas with working area and example areas

The **Working Area** is always present. It is defined by the [Stroke Limits](#) of the robot.



is used to open a window showing the coordinates of the selected safety area.



When you press  with the **working area** selected, the window for [setting the visibility of the working area](#) in the graphical display will be opened.



is used to open the [graphical display](#) of the safety areas.



is used to close the window.

See also:

3.2.6.3 Setup Menu: Safety Areas

3.2.6.3.3 Graphical display of the safety areas

3.1.15.6 Counters

This window displays the current values of the 64 counters of the teach program.

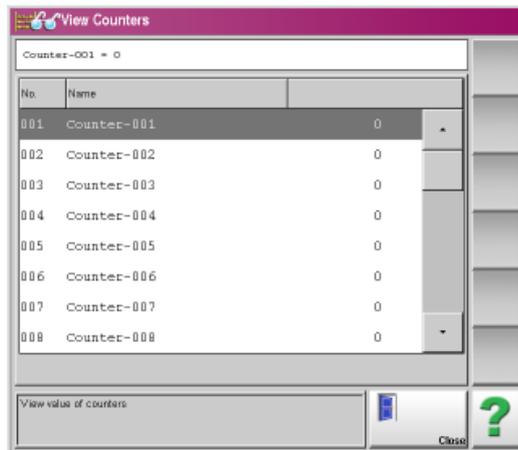


Fig.: Window for counters



is used to close the window.

3.1.15.7 Placing Counters

This window displays the current counter values of the 16 placing programs of the teach program.

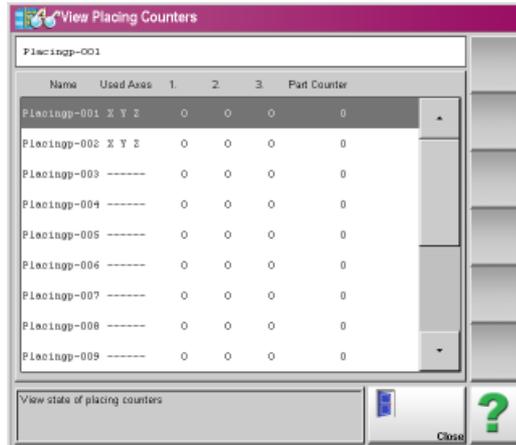


Fig.: Window for placing counters

Name

Shows the name of the respective placing program.

Used Axes

This shows which axes are used in the respective placing program.

1. - 2. - 3.

This shows the current values of the placing counters of the respective placing program.

Part Counter

This shows the current value of the part counter of the respective placing program.



is used to close the window.

3.1.15.8 Conveyors

This window shows the current state of operation of the conveyor belts.

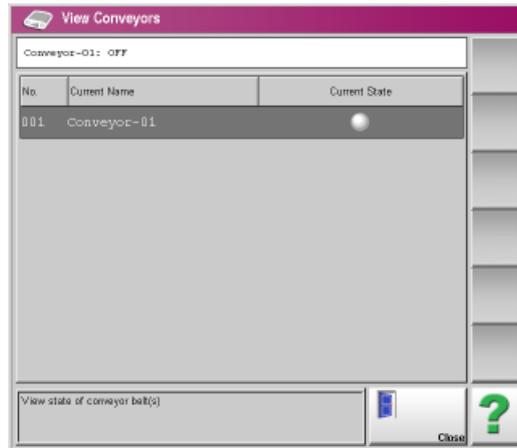


Fig.: Window for conveyors

indicates that the respective Conveyor is OFF.

indicates that the respective Conveyor is ON.



is used to close the window.

3.1.15.9 Cylinders

This window shows the current state of the cylinders defined in the teach program.

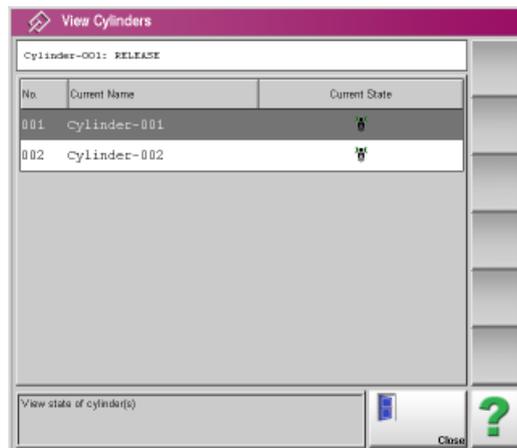


Fig.: Window for cylinders

indicates that the cylinder is in OFF position.

indicates that the cylinder is in ON position.

indicates that the cylinder is RELEASED.

indicates that the state of the cylinder is UNKOWN.



The state **RELEASED** is only available at gripper circuits equipped with the required valves.



is used to close the window.

3.1.15.10 POs - Peripheral Outputs

This window displays the current states of the peripheral outputs of the robot control unit.

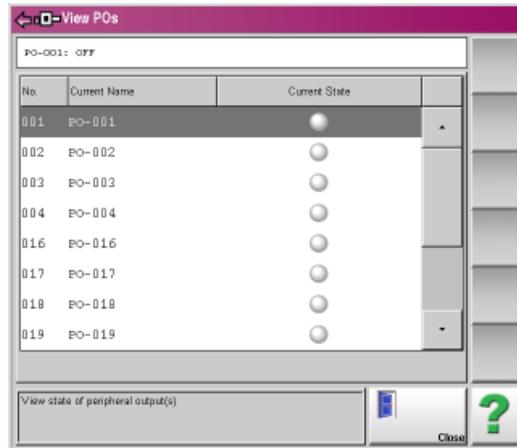


Fig.: Window for peripheral outputs

- indicates that the peripheral output is OFF (LOW).
- indicates that the peripheral output is ON (HIGH).



is used to close the window.

3.1.15.11 PIs - Peripheral Inputs

This window displays the current states of the peripheral inputs of the robot control unit.

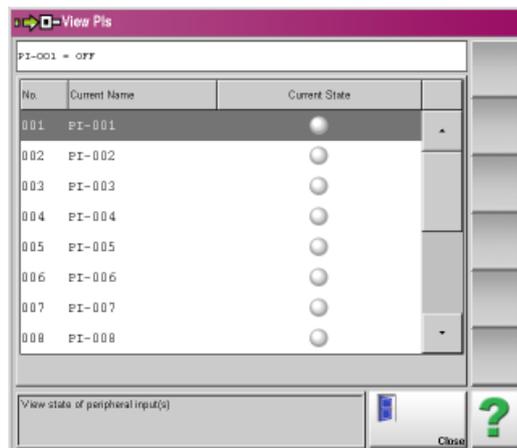


Fig.: Window for peripheral inputs

- indicates that the peripheral input is OFF (LOW).
- indicates that the peripheral input is ON (HIGH).



is used to close the window.

3.1.15.12 IMM I / O

This window displays the current states of the signals of the IMM interfaces.

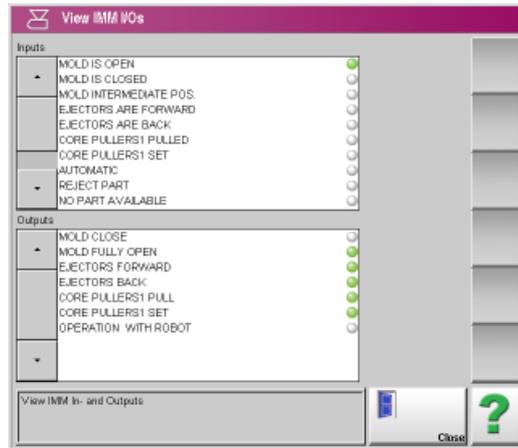


Fig.: Window for IMM interfaces

If the robot is equipped with 2 interfaces for injection molding machines, tap one of the options **IMM 1** or **IMM 2** to select the interface to be displayed.

-  indicates that the interface signal is OFF (LOW).
-  indicates that the interface signal is ON (HIGH).



is used to close the window.

3.1.15.13 AutoSwitches

This window displays the current states of the 32 AutoSwitches.

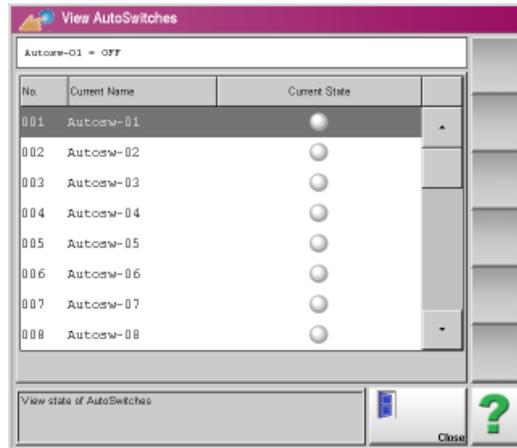


Fig.: Window for AutoSwitches

-  indicates that the AutoSwitch is OFF.
-  indicates that the AutoSwitch is ON.



is used to close the window.

3.1.15.14 Stopwatches

This window displays the values measured by the four stopwatches available in the teach program.

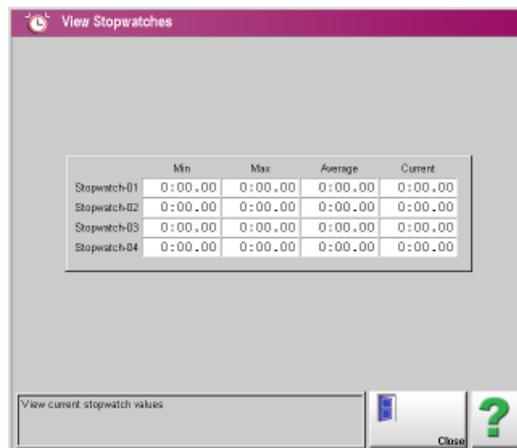


Fig.: Window for stopwatches

The **minimum value**, **maximum value** and **average value** calculated from the measurements done, as well as the **current value** are displayed for each stopwatch.



is used to close the window.

3.1.15.15 Vacuums

This window shows the current state of operation of the vacuum circuits.

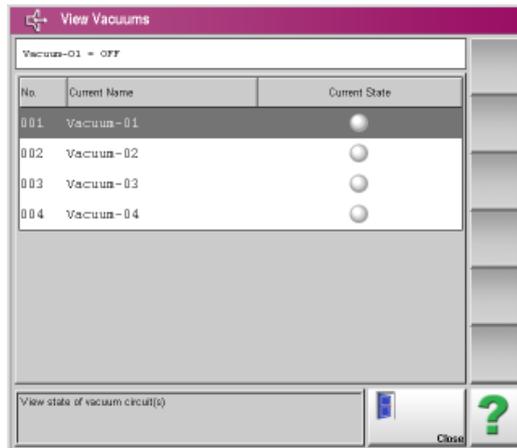


Fig.: Window for vacuum circuits

- indicates that the vacuum circuit is OFF.
- indicates that the vacuum circuit is ON.
- indicates that the respective vacuum is blowing off (optional).



is used to close the window.

3.1.15.16 Default Vacuum Level

This window shows the current pressure values and states of the analogue vacuum circuits.

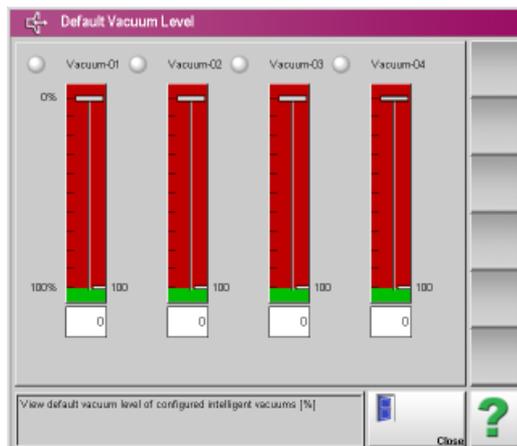


Fig.: Window for vacuum pressure values

- indicates that the vacuum circuit is OFF.
- indicates that the vacuum circuit is ON.

The big bar and the field below the scale show the current under-pressure, the small bar and the number right next to it display the set default vacuum level.



is used to close the window.

3.1.15.17 Grippers

This window shows the current state of the grippers.

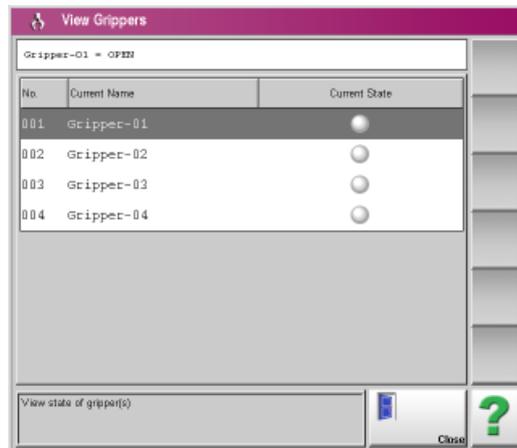


Fig.: Window for grippers

-  indicates that the gripper is OPEN.
-  indicates that the gripper is CLOSED.
-  indicates that the gripper is RELEASED.
-  indicates that the state of the gripper is UNKOWN.

 The state **RELEASED** is only available at gripper circuits equipped with the required valves.



is used to close the window.

3.1.15.18 Alarms

This window displays all occurring operational error messages in plain text, each with a specific error number.

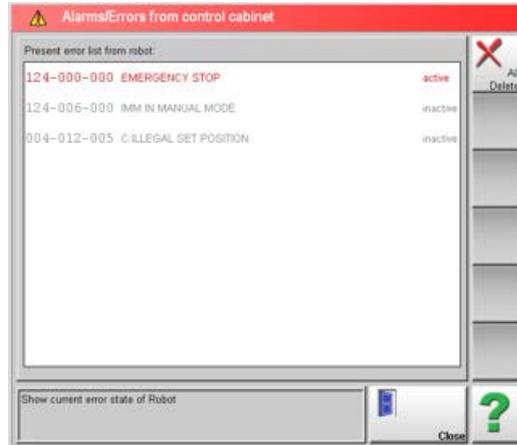


Fig.: Error Window

The error messages are displayed in two colors:

Red

Error and cause currently exist. The error is **active**. The cause of the error has to be eliminated, before the error can be confirmed. The icon  will be displayed in the status bar if one or more errors are active.

Gray

The error occurred, but the cause no longer exists. The error is **inactive** and can be confirmed.



is used to confirm inactive errors.



Press  to display a help text for each error message. Press  to turn off the help texts again.



is used to close the window.

3.1.15.19 Error Buffer

This window displays the last 500 operational error messages in chronological order.

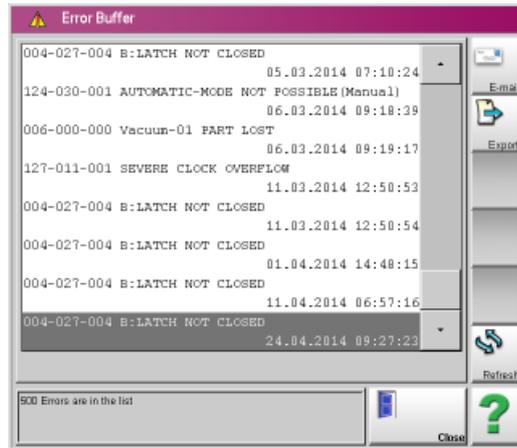


Fig.: Error Buffer



is used to refresh the list.



can be used to email the contents of the error buffer by means of the [manual email function](#) of the robot. This requires the robot being connected to a network.



is used to close the window.

3.1.15.19.1 Export the Error Buffer

The contents of the error buffer can be stored to a UNICODE text file *.TXT on the USB stick. Path and name of the file are selectable. The default folder is D:\ROBOT\LOG\ERRORS. The generated file can be opened, displayed and printed with a PC, or can be forwarded to our service department if needed.

Press  to open the dialogue for exporting the error buffer:

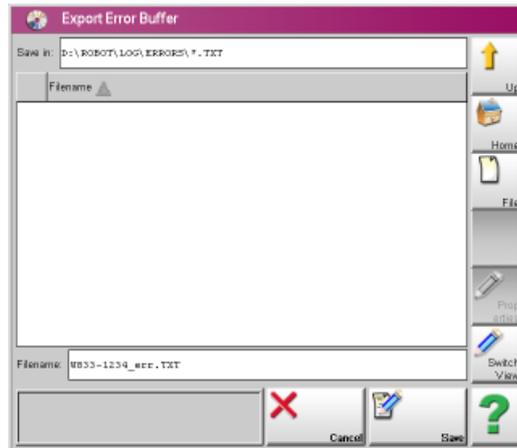


Fig.: Dialogue for exporting the error buffer

The display of the contents of the current folder uses the following icons:

-  for a sub-folder
-  for a text file *.TXT

After tapping the field **Filename** you can use the dialogue for [text input](#) to input the desired filename. By tapping on it in the display of the current folder, you can also use the name of an existing file.

 is used to step up one folder.

 opens the standard folder for this file operation.

  could be used to create a new folder.

  is used to delete the selected file or folder.

  is used to rename the selected file or folder.



is used to optimize the display of the contents of the current folder for either short or long file names.



is used to open the selected folder.



is used to save the file to the current folder, applying the name from the field **Filename**.



is used to abort the function and to close the window.

Before the file is actually stored, the Teachbox checks if a file with the same name is already stored in the selected folder, and a check back message may appear.

3.1.15.20 Command Buffer

This window displays the last actions (operating steps) executed in chronological order.

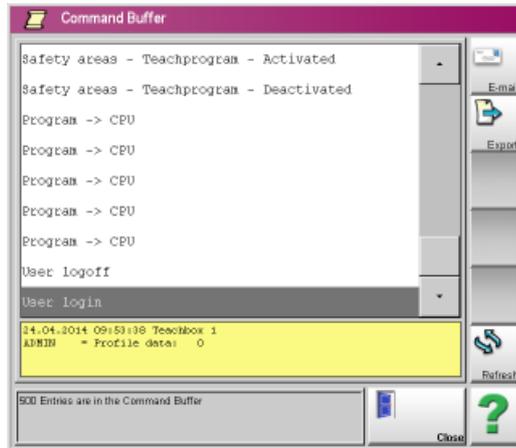


Fig.: Command Buffer



Refresh is used to refresh the list.



E-mail can be used to email the contents of the command buffer by means of the [manual email function](#) of the robot. This requires the robot being connected to a network.



Close is used to close the window.

3.1.15.20.1 Export the Command Buffer

The contents of the command buffer can be stored to an UNICODE text file *.TXT on the USB stick. Path and name of the file are selectable. The default folder is D:\ROBOT\LOG\COMMANDS. The generated file can be opened, displayed and printed with a PC, or can be forwarded to our service department if needed.



Press **Export** to open the dialogue for exporting the command buffer:

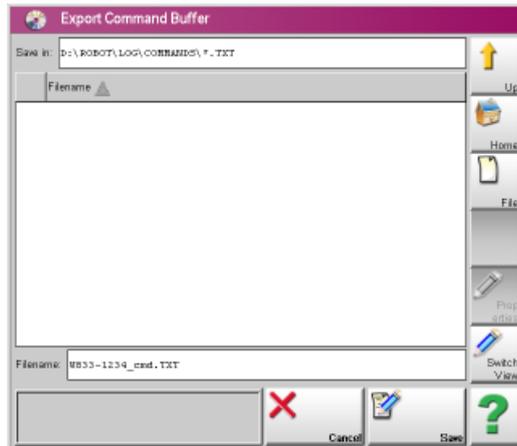


Fig.: Dialogue for exporting the command buffer

The display of the contents of the current folder uses the following icons:

- for a sub-folder
- for a text file *.TXT

After tapping the field **Filename** you can use the dialogue for [text input](#) to input the desired filename. By tapping on it in the display of the current folder, you can also use the name of an existing file.

Up is used to step up one folder.

Home opens the standard folder for this file operation.

New Dir could be used to create a new folder.

Del is used to delete the selected file or folder.

Re name is used to rename the selected file or folder.



is used to optimize the display of the contents of the current folder for either short or long file names.



is used to open the selected folder.



is used to save the file to the current folder, applying the name from the field **Filename**.



is used to abort the function and to close the window.

Before the file is actually stored, the Teachbox checks if a file with the same name is already stored in the selected folder, and a check back message may appear.

3.1.15.21 Virtual Subpendant (Emergency Stop Block)

The buttons on the Emergency Stop Block are used for basic operation of the robot. The virtual Subpendant displays these buttons on the touch screen of the Teachbox.

Touch the icon for the operating mode at the upper left corner of the touch screen to access the virtual Subpendant directly.

3.1.15.21.1 Tab Standard

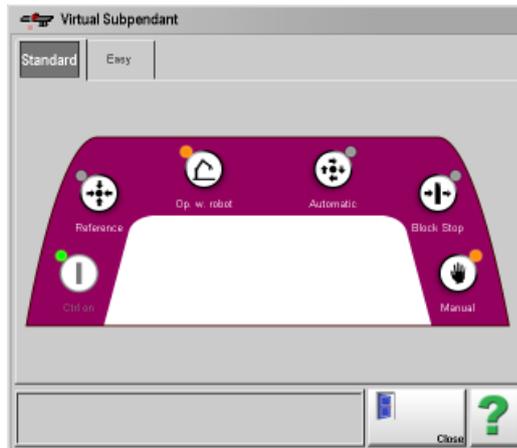


Fig.: Tab Standard of the virtual Subpendant (Emergency Stop Block)



Reference travel

Press and hold to execute the reference travel. In the mold area this is possible only when a reference program has been programmed.



IMM: Operation with robot

This enables operation of IMM with robot. In shut-off state the IMM can be operated without robot.



Automatic-Start

Start of automatic operation. The robot must be in reference position and operation with robot must be activated.



Block Stop

Press to stop automatic operation after complete execution of the currently processed instruction or group of instructions. Automatic operation can be continued by pressing the Automatic Start button again.



Manual operation / Stop

Causes an immediate stop and switches to manual operation.

See also: 3.1.3 Emergency Stop Block

3.1.15.21.2 Tab Easy

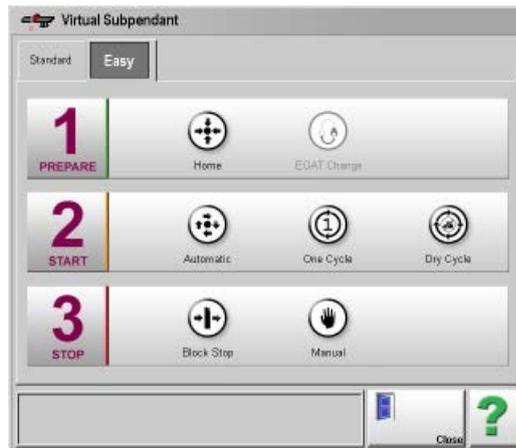


Fig.: Tab Easy of the virtual Subpendant (Emergency Stop Block)

Here, the buttons of the Subpendant and some additional functions for easy operation of the robot are collected in the groups **Prepare**, **Start** and **Stop**.



Dry Cycle

Use this function to test a Teachprogram, without producing parts with the IMM. When the Teachprogram runs on to decisions that would account for the IMM to run in full automatic mode, a dialogue will be displayed, where the user can decide how the robot shall continue.



One Cycle

Use this function to make the robot take out and place one shot only. The mold of the IMM will remain open. The robot will travel back above the IMM and will switch to Blockstop mode at the command that waits for the opening of the mold. Only after  or being pressed again, the mold of the IMM will be closed, and the next cycle will start.



EOAT Change

Use this to execute the [EOAT change program](#) .

3.1.15.22 Memory State

This window displays information about the memory load of the Teachbox. For service purposes only.

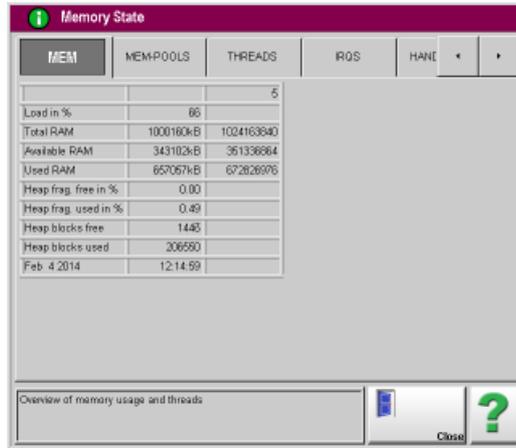


Fig.: Window for Memory State



is used to close the window.

3.1.16 System Report

The System Report is useful for simple and effective transmission of all data out of the robot control unit that might be helpful for the service department to do a remote diagnosis in case of a problem.

The following information will be stored in a file:

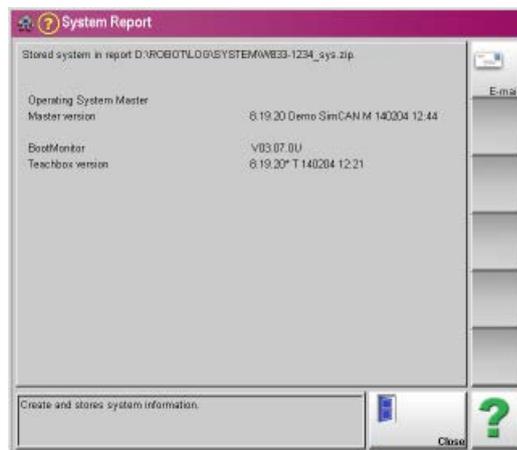
- Versions of software and operating systems
- Operating data
- Teach program out of the robot
- Contents of the error buffer
- Contents of the command buffer
- Configuration data

When creating a system report, plug a USB stick into the Teachbox first. Then open the Help menu "?" and start the function System Report.

The dialogue for [text input](#) will appear, and a comment about the report can be entered.

After this the System Report will be stored on the USB stick using the following file name and path: **..\ROBOTLOG\SYSTEMWxxx-xxx_sys.zip**

After a successful completion of the storage process, a window like this will appear:



[E-mail](#) can be used to email the system report file by means of the [manual email function](#) of the robot. This requires the robot being connected to a network.



[Close](#) is used to close the window.

The System Report can now be loaded from the USB stick to a PC, and can be forwarded via email, for example.

3.1.17 Email Functions

3.1.17.1 Manual Email Function

By means of the manual email function the user can send the current Teach program, the error buffer, command buffer or system report to any choice of recipients directly from the robot.

For using this function it is required that the robot is attached to a network and has access to a mail server. Furthermore the **Network Settings** and the **Email Settings** in the menu Init/Setup must be correct.

After accessing the function with , the window for selecting email recipients will appear:

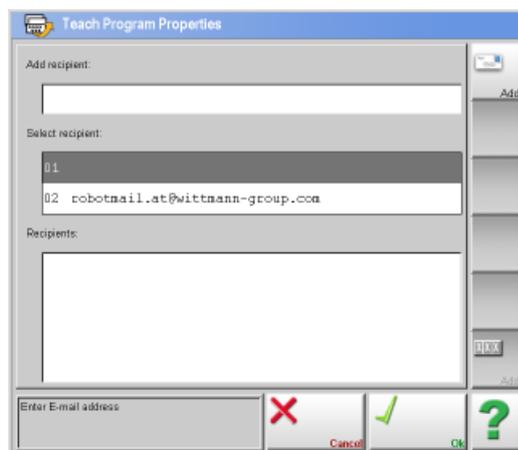


Fig.: Window for selecting email recipients

Add recipient

Tap this field to enter a new recipient with the dialogue for [text input](#).

Select recipient

Tap to select an existing recipient and press  to add this recipient to the list of recipients for the current email.

Recipients

This field displays the list of recipients chosen for the current email. Press  if you want to remove all entries from this list of recipients.



is used to send the email to the chosen recipients.



is used to cancel the function without sending an email.

3.1.17.2 Automatic Email Function

By means of the automatic email function the user can receive automatically generated emails from the robot in case it has stopped with an error message and nobody confirmed the error within a set timeout.

For using this function it is required that the robot is attached to a network and has access to a mail server. Furthermore the **Network Settings** and the **Email Settings** in the menu Init/Setup must be correct.

The email generated by the robot will have the subject `Alarms/Errors from control cabinet`.

With all possible content activated, the message text of the email will look somehow like this example:

```
This E-mail was automatically created by the robot.
```

```
Control No.: 4711  
Teachbox-Version: 8.02  
Master-Version: 8.02  
Operating Mode: Manual  
Work cell: MY TEST ROBOT
```

```
Alarms/Errors from control cabinet  
active: 0  
inactive: 1  
006-000-000 Vacuum-01 PART LOST inactive
```

3.1.18 RFID / Digital EOAT (optional)

With the optional EOAT recognition systems **RFID** and **Digital EOAT** the robot can (semi-) automatically load the correct teach program for the application at an EOAT change.

3.1.18.1 Link a Teach Program with an EOAT

If a gripper is present on the EOAT recognition system, the dialogue for linking the teach program with an EOAT-ID will be shown when storing the teach program using the menu [File – Save As](#).

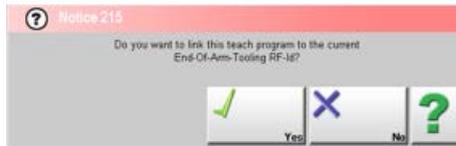


Fig.: Dialogue for linking a teach program with an EOAT-ID

Press  if you want to link the program with the EOAT.

3.1.18.2 EOAT Change

When the EOAT recognition system recognizes a newly attached gripper, the dialogue for selecting one of the teach programs linked with the EOAT will appear.

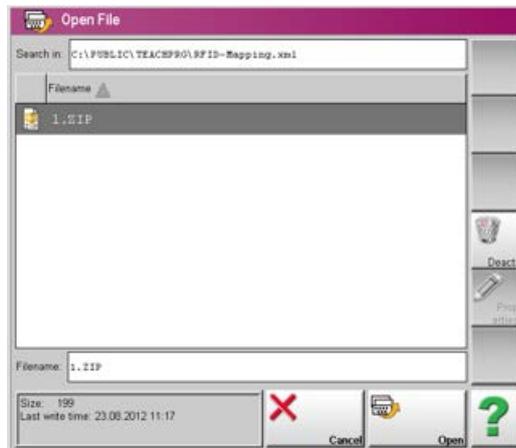


Fig.: Dialogue for selecting a linked teach program

Highlight the desired program and press  to load it.

You can use  to unlink the highlighted teach program.

Press  if you don't want to load a program.

With the corresponding setting made in the menu [Init – Setup – Robot Settings](#) the program will be loaded automatically, if only one single program is linked with the EOAT. In this case the selection dialogue will not be shown.

3.2 Teach-Mode using the Text Editor

Teach-Mode is used to create a Teach program for the robot and the attached peripheral equipment. This program contains all procedures and movements necessary for automatic operation and reference travel. In general the text editor is used for this, in order to be able to use the full functionality of the R8.3 control unit.

3.2.1 Operation of the Texteditors

3.2.1.1 Entering the Text Editor

The text editor can be called up in all operating modes using the button .

If the text editor is entered in the operating mode  **Reference Missing**, the [Offline Editor](#) will be activated automatically, because no valid positions are available for the numerical axis in this operating mode.

If the text editor is entered in one of the operating modes  **Automatic** or  **Blockstop**, only times, speeds and positions of numerical axes (+/- 10mm) can be adapted. Adding or deleting of commands will not be possible, except the [Offline Editor](#) is activated

Only if the text editor is entered in one of the operating modes  **Manual** or  **Reference**, the full functionality of the [Online Editor](#) will be available.

See also:
3.2.3 Part Programs

3.2.1.2 Exit the Texteditor and transfer the program to the CPU

Exit the Text Editor by using the button  or by using the option **Exit to Main** of the **File** menu.

Refer to: 3.2.4.11 File Menu: Exit to Main.

3.2.1.3 Editing Functions

The most common editing functions of the text editor can be accessed using the buttons on the bottom of the editor window.:

 Home takes you to the first line of the Teach program.

 End takes you to the last line of the Teach program.

 Del deletes the highlighted line(s).

 Ins adds an empty line before the highlighted one.

 Edit opens a window to edit the highlighted command.

See also:

3.1.6 Selection Mode

3.2.1.4 Online Editor

Teaching in the online editor is the most commonly used method to program a teach program. The entire program is created by approaching the various axis positions. It can be tested and optimized immediately.

3.2.1.5 Offline Editor

The offline editor can be turned on in the teach editor using the option **OfflineEditor** of the **Setup** menu. This puts the teachbox offline, but the Emergency-Stop-Block with the buttons for basic operation of the appliance remains active.

The teach program is created or edited without travelling the robot, while the robot processes another program in the Master CPU. However, no axis positions can be entered.

The turned-on offline editor is indicated by a flashing  on the status bar at the upper left of the display. If a numerical axis is selected, the characters "?????." are shown

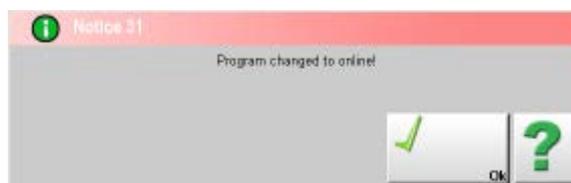
instead of the current axis position. Upon acknowledgment with  the axis command is added to the teach program with the undefined position. [Predefined positions](#) are taken over correctly.

For a program edited or created offline, all axis instructions created offline must be [converted online](#), before the program can be transferred to the master CPU and before it can be executed.

3.2.1.6 Online Conversion

The Online Conversion is started by selecting the option **OnlineEditor** in the **Setup** menu.

After a successful conversion the following notice will appear:



3.2.2 Step Operation

To test a program or a program section, you can use Step Operation run through individual instructions or the complete program. Step operation may be carried out in various operating modes:

- Step operation in Block Stop mode
- Step operation in Reference and Manual modes

To do this, the one must be in the editor, and the program in the editor must be already transferred to the CPU. The highlighted instruction is executed by holding down the



key. Execution of the instruction may be aborted at any time by releasing the key.

Step Operation in Block Stop Mode

The program is stepped through chronologically according to the automatic sequence. All travel instructions of the numerical axes are executed at the full speed of the automatic sequence.

Step Operation in Reference and Manual Modes

The speeds of travel motions of the numerical axes can be set from 1 to 30% of the maximum speed.

For safety reasons, a newly created program must be run through in Step operation before the first automatic operation.



The branch to the alternative sequence when a [monitoring](#) triggers will not be executed in step operation.

3.2.2.1 Procedure Step

By tapping the menu point **Edit-Debug-Step is Procedure Step** the user can activate or deactivate Procedure Step.

Procedure Step means, that when a CALL command is being stepped, not only the CALL command is executed, but the whole subroutine called up by the CALL command.

In this case the cursor remains on the CALL command, until the execution of the subroutine is finished.



A subroutine may only be tested using a procedure step, when it has already been tested with "normal" step operation.

3.2.3 Part Programs, PROGRAM Menu

3.2.3.1 Part Programs

Part Programs A complete standard teach program includes the following part programs:

- Robot Program (ROBOT-PRG)
- Peripheral Program (PERI-PRG)
- Robot Reference Program (ROBOT-REF)
- Peripheral Reference Program (PERI-REF)
- All Modes Program (ALLMODE-PRG)

Up to **115 additional part programs** can be added.

Each of these additional part programs can be a **Sub-Allmodes Part Program**.

One of the additional part programs can be used as **EOAT Change Program**.

After [opening the editor](#) the robot program **ROBOT-PRG** will be displayed for editing. To switch the editor to a different part program, simply select the desired program from the Menu **Program**. The status field on the bottom left corner of the editor window indicates, which part program is currently shown in the editor.



Fig.: Indication of current part program

3.2.3.1.1 Robot Program (ROBOT-PRG)

This program is used to program all movements and procedures the robot has to execute together with the IMM in  **Automatic** mode.

The robot program is executed cyclically. After execution of the last command line of the robot program or an [END](#) command within, the execution of the robot program will continue at its first line.

3.2.3.1.2 Peripheral Program (PERI-PRG)

This program is used to control all sequences (in peripheral equipment) that have to be executed independently from the movements of the robot.

The program must be [started](#) in the robot program or in one of the 115 additional part programs. After that it will be executed in parallel to (at the same time as) the other part programs running in  **Automatic** mode.

The Peripheral Program will not be executed cyclically. After execution of the last command line of a peripheral program or an [END](#) command within, the execution of the peripheral program will be stopped. If the peripheral program should be executed permanently (cyclically) this feature has to be programmed accordingly (using a [Jump command](#)).

3.2.3.1.3 Robot Reference Program (ROBOT-REF)

This program is used to control all movements and procedures executed by the robot during the reference travel in operating mode  **Manual**.

3.2.3.1.4 Peripheral Reference Program (PERI-REF)

This program is used to control all movements and procedures executed by peripheral equipment during the reference travel in operating mode  **Manual**. This program will be executed in parallel to the robot reference program. The two reference programs can perform coordinated sequences using handshakes with counters.

3.2.3.1.5 115 additional Part Programs

Like the peripheral program the additional part programs are used to control all sequences (in peripheral equipment) that have to be executed independently from the movements of the robot.

The programs must be [started](#) in the robot program or in one of the other additional part programs. After that they will be executed in parallel to (at the same time as) the other part programs running in  **Automatic** mode.

Additional part programs will not be executed cyclically. After execution of the last command line of an additional part program or an [END](#) command within, the execution of the additional part program will be stopped. If the additional part program should be executed permanently (cyclically) this feature has to be programmed accordingly (using a [Jump command](#)).

3.2.3.1.6 All Modes Program (ALLMODE PRG)

The ALL MODES program (ALLMODE-PRG) is executed by the R8.3 control unit in the background, regardless of the current mode of operation the robot is in. Meaning this program is not only executed in the operating mode  **Automatic**, but also in the operating modes  **Manual**,  **Reference** and  **Blockstop**.

The ALL MODES program cannot be executed in the operating mode  **Reference Missing**.

This program is used to control movements and procedures, that have to be executed in every operating mode of the robot (except **Reference Missing**).

The ALL MODES program does not start automatically. Like the peripheral program, it has to be started either from a different part program using the respective [Teach commands](#); or it is started manually using the function "START/STOP ALLMODE-PRG." of the INIT menu.

Once the ALL MODES program is activated, it is executed cyclically. After execution of the last command line of the ALL MODES program or an [END](#) command within, the execution of the ALL MODES program will continue at its first line.

The Icon  is shown in the status bar of the Teachbox right next to the operating mode, to indicate that the ALL MODES program is running. Changes of the mode of operation will not interrupt the ALL MODES program. Only the respective [STOP-command](#), or an emergency stop or powering down the robot will stop the ALL MODES program.

Even more flexibility can be added to ALL MODES programs using the event [Mode](#). This gives the possibility to react on the actual robot mode.

3.2.3.1.7 Sub-Allmodes Part Programs

Sub-Allmodes Part Programs are additional part programs which, like the ALL MODES program are not only executed in operating mode  **Automatic**, but also in the operating modes  **Manual**,  **Reference** and  **Blockstop**.

Sub-Allmodes Part Programs do not start automatically. They also do not start automatically, when the ALL MODES program is started. Sub-Allmodes Part Programs must be started from within the **ALL MODES program** or from within another **Sub-Allmodes Part Program** using the respective [start teach command](#).

Sub-Allmodes Part Programs will not be executed cyclically. After execution of the last command line of a Sub-Allmodes Part Program or an [END](#) command within, the execution of the Sub-Allmodes Part Program will be stopped. If a Sub-Allmodes Part Program should be executed permanently (cyclically) this feature has to be programmed accordingly (using a [Jump command](#)).

Once the ALL MODES program is stopped, all Sub-Allmodes Part Programs will be stopped also.

3.2.3.1.8 EOAT Change Program

The EOAT Change Program is used for moving to an EOAT changing position, or to change the EOAT automatically.

The EOAT Change Program has to be programmed in the text editor. All functions of the robot can be used.

When the button  on the virtual Subpendant is pressed and held during operating modes Manual or Reference, first a reference travel (according to the reference program) will be executed. After this the EOAT Change Program will run.

If the button  should be released and pressed again during execution of the EOAT Change Program, the EOAT Change Program will not start all over, but will continue working at the current program line.

3.2.3.2 Manage Part Programs

This function is used to manage the part programs of the teach program.

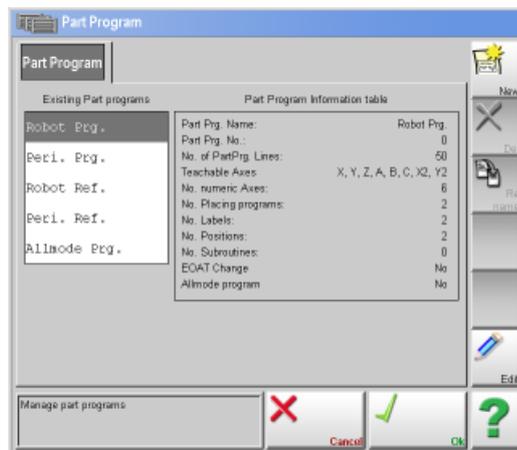


Fig.: Window to manage part programs

The **Part Program Information Table** shows information about the part program selected in the list of **Existing Part Programs**.



is used to create a [new part program](#).



is used to [edit](#) the settings of the part program selected in the list of **Existing Part Programs**.



is used to delete an **additional** part program selected in the list of **Existing Part Programs**. The standard programs ROBOT_PRG, PERI-PRG, ROBOT-REF, PERI-REF and ALLMODE-PRG cannot be deleted.



is used to close the window, and to discard the entries.



is used to confirm the entries, and to close the window.

3.2.3.2.1 Edit a Part Program

This function is used to create a new additional part program, or to edit the settings for an existing part program.

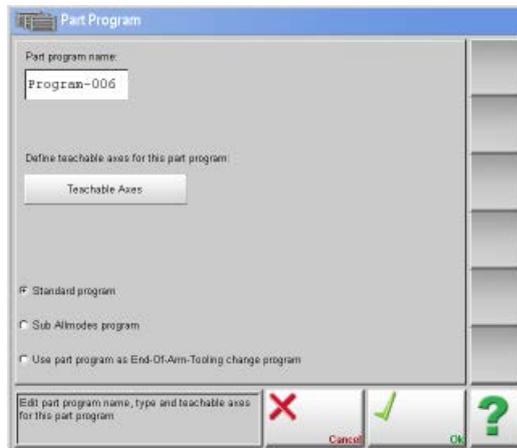


Fig.: Window to edit part program settings

After tapping the field **Part program name** you can change the name of an additional part program using the dialogue for [text input](#). The names of the standard programs ROBOT_PRG, PERI-PRG, ROBOT-REF, PERI-REF and ALLMODE-PRG cannot be altered.

The button **Teachable Axes** is used to open the dialogue for [axes definition](#). Her you can assign axes to the part program.

Use one of the three options on the left bottom of the window to define the type of the new part program:

Standard Program

Use this option to create a standard **additional part program**.

Allmode Program

Use this option to create a **sub-allmodes part program**.

Use part program as End-Of-Arm-Tooling change program

Use this option to create the **EOAT change program**. The standard name **EOAT Change** is assigned to this program automatically.



is used to close the window, and to discard the entries.



is used to confirm the entries, and to close the window.

3.2.4 FILE Menu of the text editor

3.2.4.1 New

This function is used to create a new empty teach program in the Teachbox.

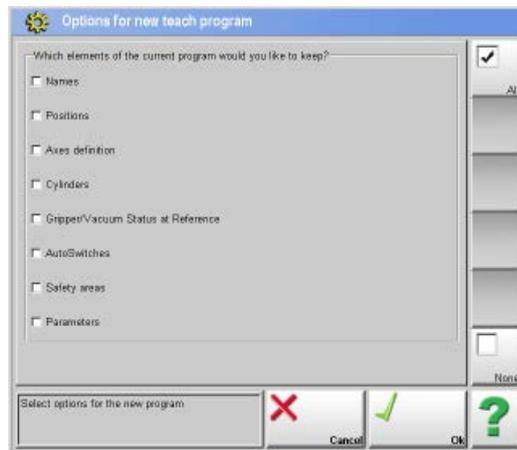


Fig.: Options window for a new teach program

The window **Options for new Teach program** is used to select which elements from the existing program are kept with the new teach program. This is possible for:

- Names
- Positions
- Axis Definitions
- Cylinders
- Reference settings for vacuum and gripper circuits
- AutoSwitches
- Safety areas

The different elements are selected or deselected by tapping the respective option. A ✓ means, that the element is selected to be taken over to the new program.



is used to select all elements.



is used to deselect all elements.



is used to cancel the process.



is used to create a new empty teach program.



The program currently loaded in the Teachbox will be irretrievably deleted. The program in the robot remains unchanged.

3.2.4.2 Open

This function is used to load a teach program from a storage medium to the Teachbox.

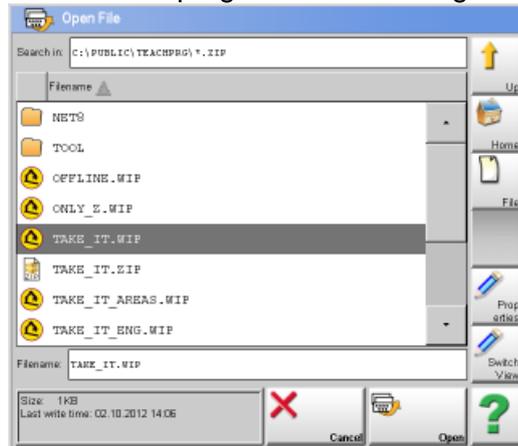


Fig.: Window to open a file

The field **Search in** displays the path of the current folder. By touching the field you can select one of the standard folders C:\PUBLIC (internal storage medium of the Teachbox) or D:\ROBOT\TEACHPRG (USB-Stick).

The display of the contents of the current folder uses the following icons:

- for a sub-folder
- for a teach program file *.WIP
- for a teach program file *.ZIP

is used to step up one folder.

opens the standard folder for this file operation.

could be used to create a new folder.

is used to delete the selected file or folder.

is used to rename the selected file or folder.

is used to open the [properties](#) window of the selected program.

can be used to optimize the display of the contents of the folder for either long or short file names.

 is used to open the selected folder.

 is used to load the selected program to the Teachbox.

 is used to abort the function and close the window.

 The program stored in the teachbox will be irretrievably deleted by loading another teach program.

 Before the program can be executed, it has to be [transferred to the robot](#).

 Every newly loaded teach program must be run through at least once in step mode before automatic operation.

 The file name of a stored teach program is not necessarily the same as the teach program name.

3.2.4.3 Import Macro

A macro is a file *.WIM which contains one or more program lines, that were exported from a (different) teach program previously.

The function **Import macro** is used to insert a macro *.WIM from a storage medium to the teach program before the current position of the cursor.

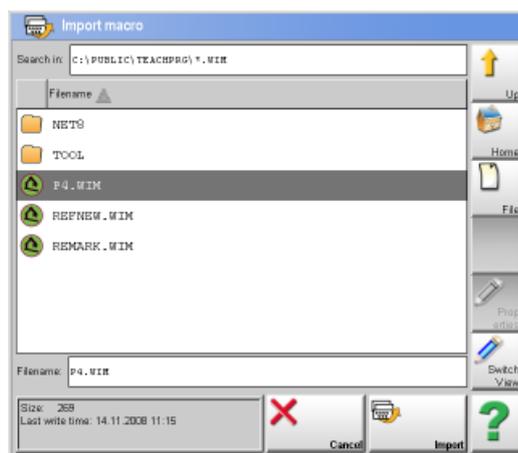


Abb.: Window to import a Macro

The field **Search in** displays the path of the current folder. By touching the field you can select one of the standard folders C:\PUBLIC (internal storage medium of the Teachbox) or D:\ROBOT\TEACHPRG (USB-Stick).

The display of the contents of the current folder uses the following icons:

 for a sub-folder

 for a macro file *.WIM

 is used to step up one folder.

 opens the standard folder for this file operation.

  could be used to create a new folder.

  is used to delete the selected file or folder.

  is used to rename the selected file or folder.

 can be used to optimize the display of the contents of the folder for either long or short file names.

 is used to open the selected folder.

 is used to load the selected macro to the Teachbox, and to insert it to teach program before the selected line.

 is used to abort the function and close the window.

If labels or subroutines whose names already exist in the teach program are loaded, they are automatically renamed to LABEL_XXX or SUBR_XXX. "XXX" represents a 3 digit number.

If an associated JUMP instruction is loaded together with an already existing label, the JUMP instruction will also be changed to the new label name. The same applies to CALL instructions that are loaded together with their subroutines.

If a placing program, which has the same name as an existing placing program in the teach program but otherwise differs from the latter in some way, it will automatically be renamed with the name of the first unused placing program.

If a macro that does not correspond to configured robot hardware (e.g., an axis not present) is loaded, a flag will appear and the offline editor will be activated. The corresponding instructions will now have to be changed or removed manually, after

which the program can be switched back ONLINE again with the function [ONLINE EDITOR](#) of the Setup menu.

If an axis instruction with a reference to a previously defined position, which has the same name as a previously defined position existing in the teach program but otherwise differs in some way from the latter, is loaded, it will be automatically renamed to POSITION_XXX. "XXX" represents a 3 digit number.



When a macro is loaded, axis positions of configured axes will be transferred without comment. However, this does not mean that they can be approached without risk. Programs expanded by macros must therefore always be tested in STEP mode.



When a macro is loaded, instructions with configured peripheral input/outputs are accepted without comment. However, this does not mean that these instruction sequences are compatible with the attached peripheral, especially when the macro comes from another control system.

3.2.4.4 Load Program from CPU

This function is used to load the teach program from the robot to the Teachbox.



The teach program loaded in the Teachbox including all names, cylinder definitions and position definitions will be irreversibly deleted.

3.2.4.5 Save (NAME.ZIP)

This function is used to save the teach program from the Teachbox to a storage medium as a *.ZIP file. Both name and path of the file will be the ones used before. The name of the target file is shown in brackets right next to the menu entry **Save**. If no name is displayed, **Save** will call the function **Save as** instead.



The previous version of the program stored on the storage medium with the same name will be deleted irretrievably.

3.2.4.6 Save As

This function is used to save the teach program from the Teachbox to a storage medium as a *.ZIP file. Path and name of the file are selectable.

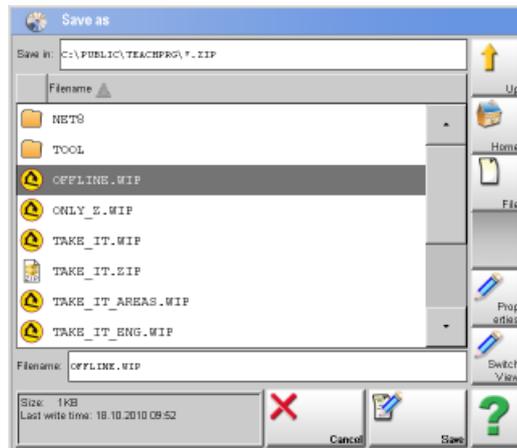


Fig.: Window to save a teach program

The field **Save in** displays the path of the current folder. By touching the field you can select one of the standard folders C:\PUBLIC (internal storage medium of the Teachbox) or D:\ROBOT\TEACHPRG (USB-Stick).

The display of the contents of the current folder uses the following icons:

- for a sub-folder
- for a teach program file *.WIP
- for a teach program file *.ZIP

The field **Filename** suggests the last filename used or the teach program name as filename. After tapping the field you can use the dialogue for [text input](#) to change the filename as desired. The extension will always be ZIP. By tapping it in the display of the current folder, you can also use the name of an existing file.

is used to step up one folder.

opens the standard folder for this file operation.

is used to create a new folder.

is used to delete the selected file or folder.

is used to rename the selected file or folder.



is used to open the [properties](#) window of the selected program.



can be used to optimize the display of the contents of the folder for either long or short file names.



is used to open the selected folder.



is used to save the teach program to the current folder, applying the name from the field **Filename**.



is used to abort the function and close the window.

Before storing is executed, a check is made to see whether a program of the same name is already stored in the selected folder, and a check back message may appear.

If a coded gripper is present on the optional [RFID / Digital-EOAT](#) gripper recognition system, you will be asked if you want to assign the program to that gripper.

3.2.4.7 Export Selection

This function is used to save previously selected lines from the teach program to a storage medium as a macro *.WIM. Path and name of the file are selectable.

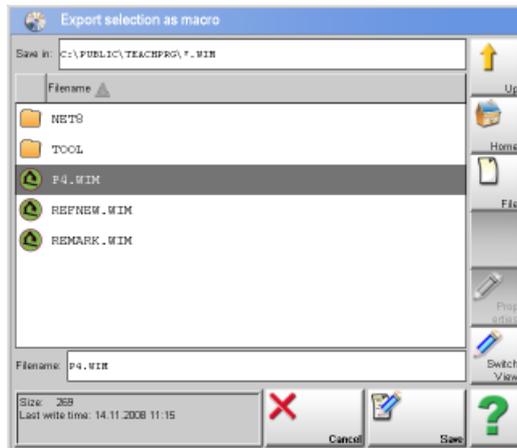


Fig.: Window to save a macro

The field **Save in** displays the path of the current folder. By touching the field you can select one of the standard folders C:\PUBLIC (internal storage medium of the Teachbox) or D:\ROBOT\TEACHPRG (USB-Stick).

The display of the contents of the current folder uses the following icons:



for a sub-folder



for a macro file *.WIM

After tapping the field **Filename** you can use the dialogue for [text input](#) to enter the desired filename. The extension will always be WIM. By tapping it in the display of the current folder, you can also use the name of an existing file.



Up is used to step up one folder.



Home opens the standard folder for this file operation.



File New Dir could be used to create a new folder.



File Del is used to delete the selected file or folder.



File Re name is used to rename the selected file or folder.



Switch View can be used to optimize the display of the contents of the folder for either long or short file names.



Open is used to open the selected folder.



Save is used to save the macro to the current folder, applying the name from the field **Filename**.



Cancel is used to abort the function and close the window.

Before storing is executed, a check is made to see whether a macro of the same name is already stored in the selected folder, and a checkback message may appear.

See also:

3.2.4.3 File Menu: Import Macro

3.2.5.1 Edit Menu: Selection Mode

3.2.4.8 Transfer program to CPU

This is used to transfer the teach program from the Teachbox to the robot. This is necessary to enable execution of the program.

A progress bar is shown during the process.

With the appropriate setting of the option **Clear counters** in the Init Menu [Setup: Robot Settings](#) a request about resetting counters and placing programs will pop up:



The program previously stored in the robot will be irretrievably overwritten.

3.2.4.9 Properties

Here the properties of the teach program in the text editor of the Teachbox are displayed. That is the program which can be edited. Before this program can be executed in automatic operation, it has to be transferred to the robot.

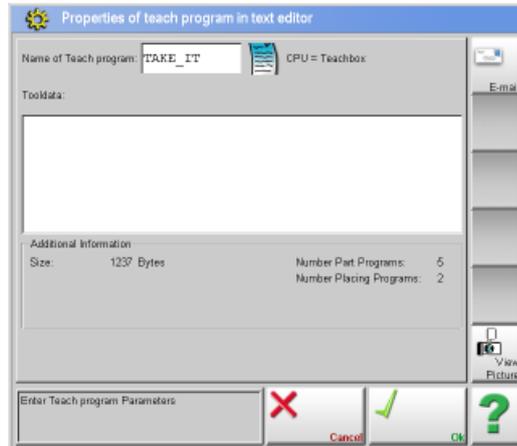


Fig.: Window for properties of the program in the texteditor

Name of Teach program

Shows the name of the teach program in the text editor. This name is not necessarily the same as the file name of the program. After tapping the input field, the name can be edited using the dialogue for [text input](#).

CPU = Teachbox or CPU <> Teachbox

Displays whether the teach programs in the robot (CPU) and the Teachbox are equal or unequal.

Tooldata

Displays the Tooldata text of the teach program in the text editor. After tapping the input field, the Tooldata text can be edited using the dialogue for [text input](#).

Size

Displays the size of the teach program in the text editor. The maximum size of a teach program is limited to 32.768 bytes.

Number Part Programs

Shows the number of part programs of the teach program in the text editor: ROBOT-PRG, PERI-PRG, ROBOT-REF, PERI-REF, ALLMODE-PRG and up to 11 additional part programs.

Number Placing Programs

Shows the number of placing programs used in the teach program in the text editor. The maximum number of placing programs available in a teach program is limited to 16.



can be used to open the picture viewer. There you can either [associate a picture](#) with the teach program, or [watch the picture](#) associated with the teach program.



can be used to email the teach program by means of the [manual email function](#) of the robot as an email attachment (*.wip, *.htm and *.txt file).



is used to discard the entries and to close the window.



is used to confirm the entries and to close the window.

3.2.4.10 Print

This function is used to print the teach program loaded in the Teachbox to either a UNICODE-textfile *.TXT or an HTML-file *.HTM on the USB stick. Path and name of the file are selectable. The default folder is D:\ROBOT\PRINT. The generated file can be opened, displayed and printed with a PC.

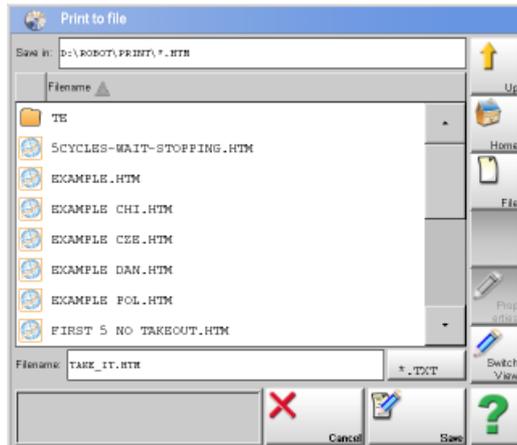


Fig.: Window to print a teach program to a file

The button right next to the field for the filename is used to select the type of file for printing:



activates printing to a textfile



activates printing to an HTML-file

The field **Save in** displays the path of the current folder.

The display of the contents of the current folder uses the following icons:



for a sub-folder



for a textfile *.TXT



for an HTML-file *.HTM

Only files of the same type as selected for the printout will be displayed.

The field **Filename** suggests the last filename used or the teach program name as filename. After tapping the field you can use the dialogue for [text input](#) to change the filename as desired. By tapping it in the display of the current folder, you can also use the name of an existing file.



Up is used to step up one folder.



Home opens the standard folder for this file operation.



File New Dir could be used to create a new folder.



File Del is used to delete the selected file or folder.



File Re name is used to rename the selected file or folder.



Switch View can be used to optimize the display of the contents of the folder for either long or short file names.



Open is used to open the selected folder.



Save is used to save the printout to the current folder, applying the name from the field **Filename**.



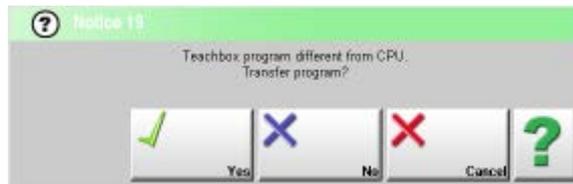
Cancel is used to abort the function and close the window.

Before storing is executed, a check is made to see whether a file of the same name is already stored in the selected folder, and a checkback message may appear.

3.2.4.11 Exit to Main

This function can also be called with the button , and is used to exit the editor.

If the teach program in the editor is different than the one in the robot, then it is possible to select whether or not the program shall be transferred from the Teachbox to the robot:



This is necessary to enable execution of the program.

A progress bar will be shown during the process.

With the appropriate setting of the option **Clear counters** in the Init Menu [Setup: Robot Settings](#) a request about resetting counters and placing programs will pop up:



The program previously stored in the robot will be irretrievably overwritten, if the program from the Teachbox is transferred to the robot.

3.2.4.12 Properties Window of the File Dialogue

Here the properties of the teach program selected in the **file dialogue** are displayed.

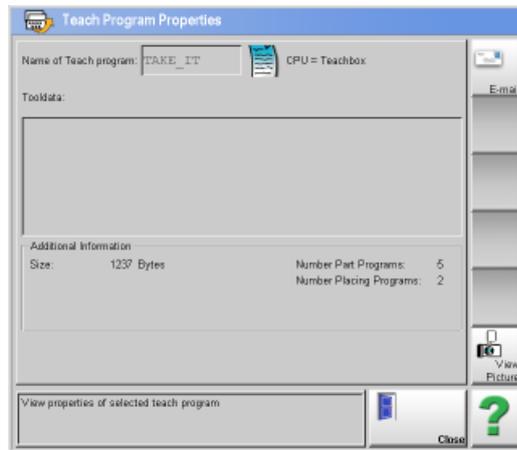


Fig.: Window for properties of a stored program

Name of Teach program

Shows the name of the selected teach program. This name is not necessarily the same as the file name of the program.

CPU = Teachbox or CPU <> Teachbox

Displays whether the selected teach program and the one in the robot (CPU) are equal or unequal.

Tooldata

Displays the Tooldata text of the selected teach program. The text for Tooldata can only be edited in the properties window of the **text editor**.

Size

Displays the size of the selected teach program. The maximum size of a teach program is limited to 32.768 bytes.

Number Part Programs

Shows the number of part programs of the selected teach program: ROBOT-PRG, PERI-PRG, ROBOT-REF, PERI-REF, ALLMODE-PRG and up to 11 additional part programs.

Number Placing Programs

Shows the number of placing programs used in the selected teach program. The maximum number of placing programs available in a teach program is limited to 16.

If a picture has been associated with the selected teach program,  can be used to open the [picture viewer](#).

 can be used to email the teach program by means of the [manual email function](#) of the robot as an email attachment (*.wip, *.htm and *.txt file).

 is used to exit the window.

3.2.4.13 Picture Viewer

Here the picture associated with the teach program can be viewed, provided that the storage medium containing the associated picture is available.

The picture viewer can be accessed in the properties windows of the teach program

using the button 

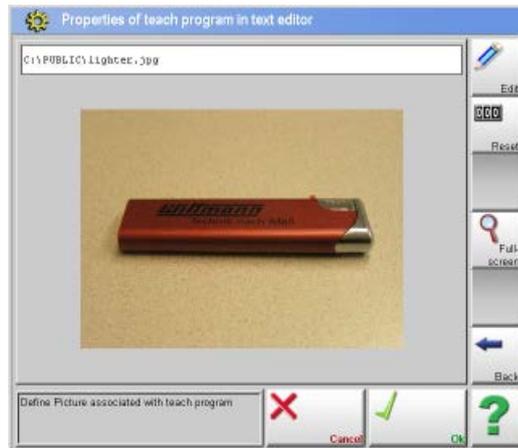


Fig.: Picture viewer showing a picture

If the picture is too big, it can be moved horizontally and vertically using the sliders.

Associating a picture to a teach program is only possible, if the respective properties window has been opened with the menu **File / Properties** inside one of the editors.



is used to return to the properties window.

3.2.4.13.1 Associate a picture with a Teach program

In order to be able to associate a picture to a teach program, first the properties window must be opened with the menu **File / Properties** inside one of the editors. Now

the  is used to open the picture viewer.

Here the button  opens the file dialogue for selecting a picture.

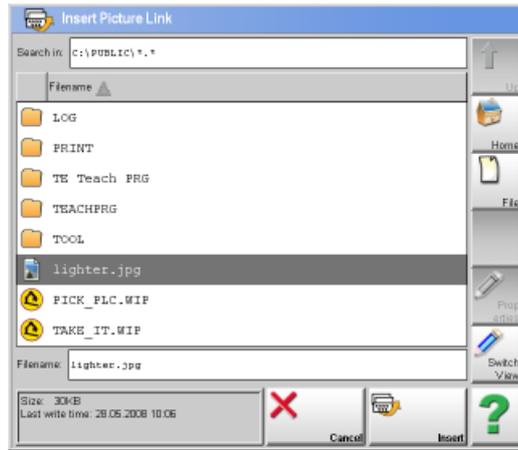


Fig.: Dialogue for selecting a picture

The following file formats are supported:

- BMP files with 256 colors
- BMP files with 24 bits color depth
- GIF files with 256 arbitrary colors
- JPG files (not all compression rates and color depths)

The icons used by the display of the contents of the current folder include the following:

 for a sub-folder

 for supported picture files

 is used to step up one folder.

 opens the standard folder for this file operation.

  could be used to create a new folder.

  is used to delete the selected file or folder.

  is used to rename the selected file or folder.

 can be used to optimize the display of the contents of the folder for either long or short file names.

 is used to open the selected folder.

 associates the selected picture with the teach program.

 is used to abort the function and to close the window.

3.2.5 EDIT Menu

3.2.5.1 Selection Mode

The option **Selection Mode On/Off** is used to switch on or switch off the selection mode of the text editor. The selection mode is needed to mark several program lines.

When the selection mode is off, single program lines can be selected by tapping on them in the editor window. The currently selected line is displayed on a gray background.

```
0011 *X: 486.4
0012 Y: 206.0
0013 Gripper-01 = CLOSE
0014 Gripper-02 = CLOSE
0015 Vacuum-01 = ON
0016 Vacuum-02 = ON
0017 WAIT Vacuum-01 = ON
0018 WAIT Vacuum-02 = ON
0019 *X: 45.6
```

Fig.: a single selected line

If the selection mode is activated now, the selection can be expanded to several lines, by tapping the last line to be included to the selection.

```
0011 *X: 486.4
0012 Y: 206.0
0013 Gripper-01 = CLOSE
0014 Gripper-02 = CLOSE
0015 Vacuum-01 = ON
0016 Vacuum-02 = ON
0017 WAIT Vacuum-01 = ON
0018 WAIT Vacuum-02 = ON
0019 *X: 45.6
```

Fig.: several selected lines

The block of lines being selected this way, can now either be moved or copied via the clipboard, or can be exported to a storage medium as a macro.



Del could be used to delete the selected lines.

3.2.5.2 Copy, Cut, Paste

Using these functions of the **Edit** menu you can copy or move previously [selected](#) program lines to a different location in the program via the clipboard.

Copy

The function Copy copies the selected lines to the clipboard.

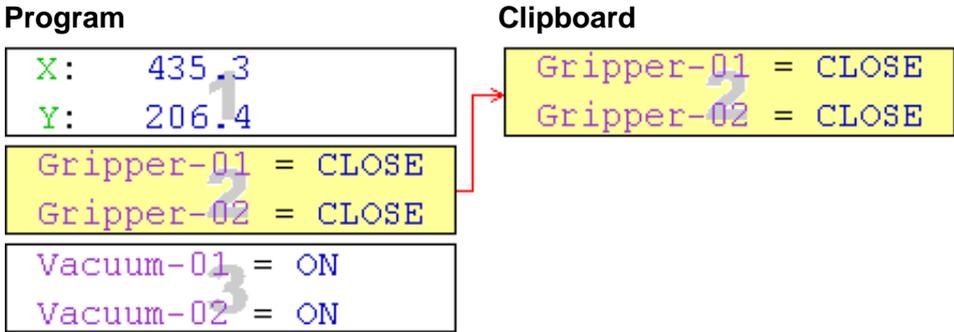


Fig.: Copy

Cut

The function Cut moves the selected lines to the clipboard and deletes them at their original position in the program.

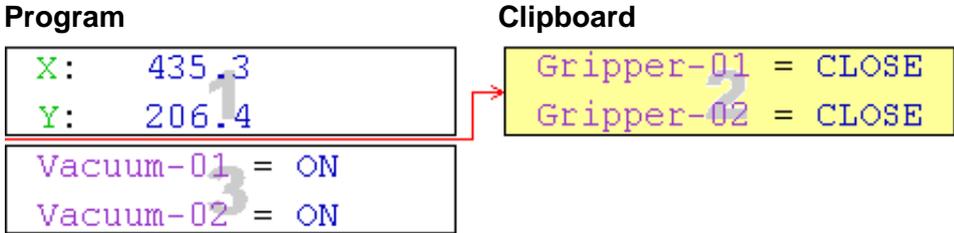


Fig.: Cut

Paste

The function Paste copies the lines, that were previously put to the clipboard, from the clipboard to back to the program just before the selected line. In the example shown below, the block of lines 2 has been previously cut out between blocks 1 and 3, and is now inserted behind block 3.

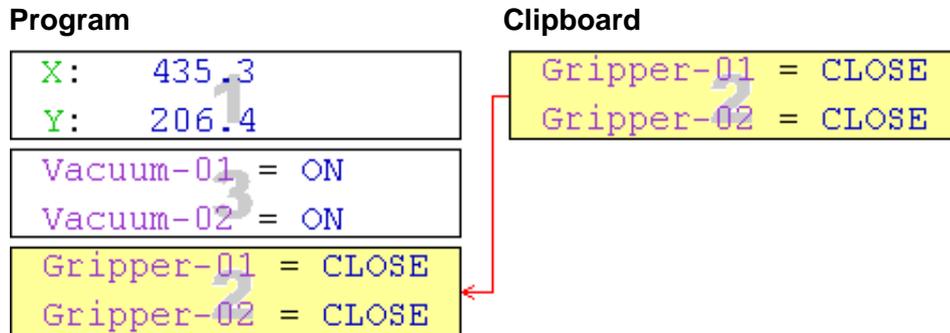


Fig.: Paste

-  If labels or subroutines are copied, they are automatically renamed to LABEL_XXX or SUBR_XXX. "xxx" represents a 3 digit number.
-  If an associated JUMP instruction is copied together with its label, the JUMP instruction will also be changed to the new label name. The same applies to CALL instructions that are copied together with their subroutines.
-  If a placing program is copied, it will automatically be renamed with the name of the first unused placing program.

3.2.5.3 Search-Toolbar

Use the option **Show Search-Toolbar** of the EDIT menu to activate or deactivate the display of the Search-Toolbar of the text editor. The Search-Toolbar is used to search for arbitrary text in the Teach program.



Fig.: Search-Toolbar

After tapping the input field use the dialogue for [text input](#) to input the desired search text.

Use  to access a list with previously used search texts.

Use  to find the next occurrence of the search text before the current cursor position in the Teach program.

Use  to find the next occurrence of the search text after the current cursor position in the Teach program.

Use  to remove the Search-Toolbar from the display.

Use  to open the window for the settings of the Search Toolbar.

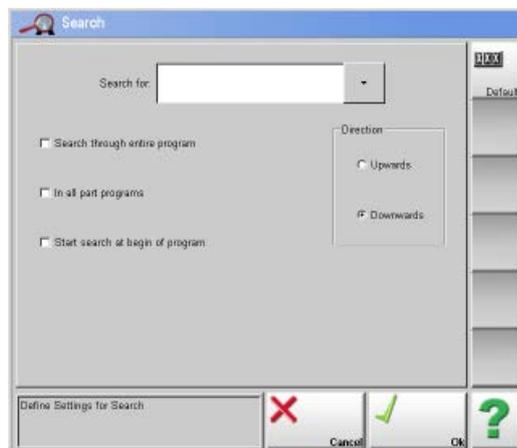


Fig.: Window for the settings of the Search Toolbar

Search through entire program

The search will not stop at the end or begin of the program, but will continue until it reaches the starting position of the search again.

In all part programs

The search will not only be performed in the current part program, but in all part programs.

Start search at begin of program

The search will not start at the current cursor position, but at the begin of the program.

Direction upwards / downwards

The search will be performed in the selected direction.

Press  to recall the standard settings.

3.2.5.4 Show Prefixes

Use the option **Debug - Show Prefixes** of the EDIT menu to activate or deactivate the display of prefixes before the names of elements in Teachcommands. Displaying prefixes can increase the readability of a teach program, especially if the user has given similar or equal names to different kinds of elements.

Example:

As they deal with the same function, an Autoswitch, a peripheral output and a counter have been given the same custom name "EXAMPLE".

Without prefixes a snippet of that Teach program would be displayed by the text editor like this:

```
IF EXAMPLE = ON
EXAMPLE: ON
EXAMPLE = EXAMPLE + 1
ENDIF
```

With prefixes, the same snippet would result in this display:

```
IF ASW_EXAMPLE = ON
PO_EXAMPLE: ON
CNT_EXAMPLE = CNT_EXAMPLE + 1
ENDIF
```

3.2.5.5 Show Default Names

Use the option **Debug - Show Default Names** of the EDIT menu to activate or deactivate the display of standard names in parentheses after the current names of elements in Teachcommands. Displaying standard names can increase the readability of a teach program, especially if the user has given similar or equal names to different kinds of elements.

Example:

As they deal with the same function, an Autoswitch, a peripheral output and a counter have been given the same custom name "EXAMPLE".

Without standard names a snippet of that Teach program would be displayed by the text editor like this:

```
IF EXAMPLE = ON
EXAMPLE: ON
EXAMPLE = EXAMPLE + 1
ENDIF
```

With standard names, the same snippet would result in this display:

```
IF EXAMPLE(Autosw-01) = ON
EXAMPLE(PO-001): ON
EXAMPLE(Counter-001) = EXAMPLE(Counter-001) + 1
ENDIF
```

3.2.5.6 Show Internal IDs

Use the option **Debug - Show Internal IDs** of the EDIT menu to activate or deactivate the display of internal IDs in parentheses after the current names of Placing Programs.

Example:

Without internal ID a Placing Program with the name EXAMPLE will be displayed in the text editor like this:

```
Placing Program: EXAMPLE
```

With internal ID the same Placing Program will be displayed in the text editor like this:

```
Placing Program: EXAMPLE(ID 000)
```

3.2.5.7 Show Line Numbers

Use the option **Debug - Show Line Numbers** of the EDIT menu to activate or deactivate the display of line numbers left of the teach commands in the text editor.

3.2.5.8 Show Execution Times

Use the option **Debug - Show Execution Times** of the EDIT menu to activate the display of the actually measured execution times of each command of the teach program.

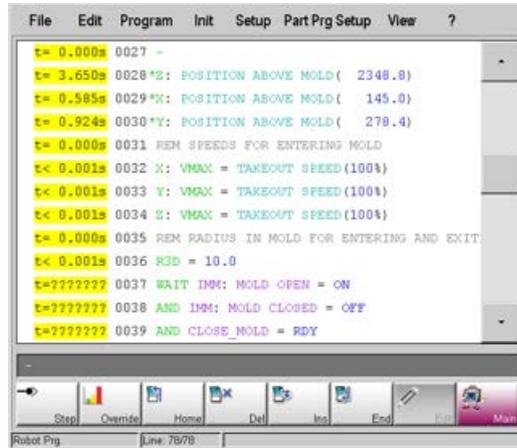


Fig.: Display of execution times

The times displayed are being refreshed with every execution of the respective command line.

t=?????? indicates that the respective command line has not been executed yet.

3.2.6 Teach program Settings in the SETUP Menu

3.2.6.1 Online / Offline Editor

This is to switch between Offline and Online Editor.

The Offline Editor is used to edit a teach program without the robot.

See also:

3.2.1.4 Online Editor

3.2.1.5 Offline Editor

3.2.1.6 Online Conversion

3.2.6.2 Axes Definition

Here you can assign the axes controlled by the robot control unit to the different part programs and paths for automatic operation.

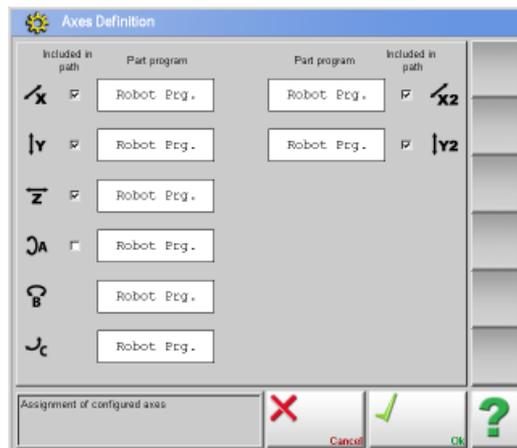


Fig.: Window for axes definition

Using the selection field **Part Program** the respective axis can be assigned to a specific part program. The movements of that axis are then controlled by this part program only.

With a at the option **Included in path** the respective (numerical) axis is included to the path calculation. This enables the axis to be moved coordinated together with the other [path axes](#) of the respective part program using 3D and Curve movements.



discards the entries and closes the window.



confirms the settings and closes the window.

3.2.6.3 Safety Areas

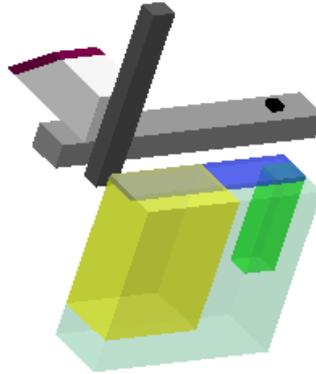


Fig.: 3d view of a typical simple safety area configuration including the working area

The R8.3 gives the possibility of setting up Teach program specific safety areas to prevent the robot from crashing obstacles in all operating modes, except **Reference Missing**. This is achieved by creating several overlapping, box-shaped, **inclusive** areas, in which the robot can move safely. Every attempt to exit the resulting safety area will make the robot stop at the border of the area and trigger an error message.

Furthermore, there is the possibility to use **exclusionary** areas to exclude box shaped regions from the working area of the robot.

Individual safety areas can be activated dependent on the state of an input or output of the robot.

A peripheral output can be switched on, when the robot is situated inside or outside a certain safety area.

All settings for the safety areas of the robot are done in the Setup menu Safety Areas. The safety areas will be automatically stored and loaded as part of the respective Teach program. When you create a [new program](#), you can choose to keep the safety areas of the currently loaded program.

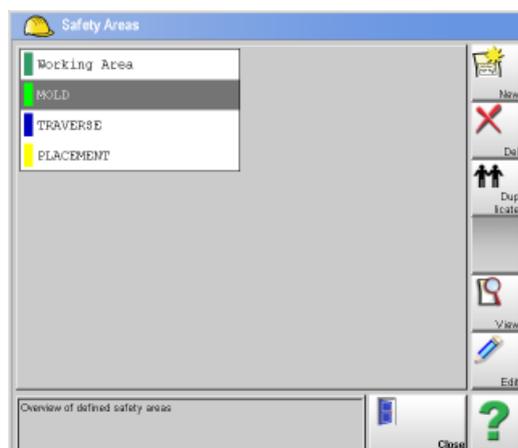


Fig.: Setup menu safety areas with working area and example areas

The **Working Area** is always present. It is defined by the [Stroke Limits](#) of the robot.



is used to create a [new safety area](#).



is used to [edit](#) the selected safety area.



When you press  with the **working area** selected, the window for [setting the visibility of the working area](#) in the graphical display will be opened.



is used to delete the selected safety area.



is used to copy the selected safety area. The dialog for [editing](#) the new safety area will pop up.



is used to open the [graphical display](#) of the safety areas.



is used to close the window.



The safety areas of the R8.3 are Teach program specific. A new safety area or changes to an existing safety area will only become valid after the Teach program has been [transferred](#) to the CPU!



The safety areas of the R8.3 are inoperative in operating mode **Reference Missing!**

3.2.6.3.1 Create / Edit Safety Areas

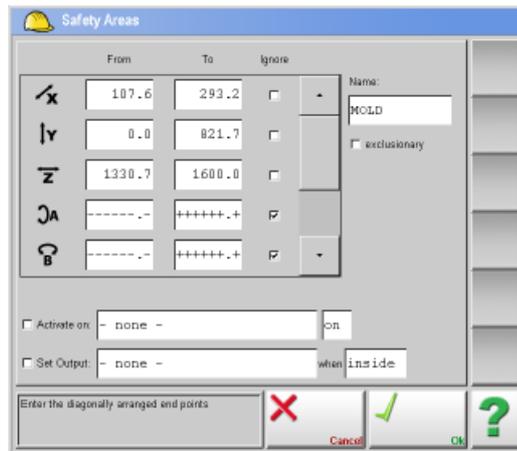


Fig.: Window to edit safety areas

Activate each single axis coordinate **From / To** with a ✓, and approach them one after the other to create the desired box-shaped safety area.

The desired name for the safety area can be entered after tapping the field **Name** using the dialogue for [text input](#).

The safety areas of the R8.3 are **inclusive** by default. This means, that as soon as at least 1 area has been defined, the robot is only allowed to travel within the defined areas. With a ✓ at the option **exclusionary** you can also define exclusionary areas. The robot is not allowed to enter exclusionary areas.

Use the option **Activate on** to activate the safety area dependent on the state of an input or output of the robot.

Use the option **Set Output** to switch on a peripheral output, when the robot is situated inside or outside the safety area.



discards the entries and closes the window.



confirms the safety area and closes the window.



The safety areas of the R8.3 are Teach program specific. A new safety area or changes to an existing safety area will only become valid after the Teach program has been [transferred](#) to the CPU!



The safety areas of the R8.3 are inoperative in operating mode **Reference Missing!**

3.2.6.3.2 Properties of the Working Area

Here you can set the visibility of the working area in the [graphical display](#) of the safety areas.

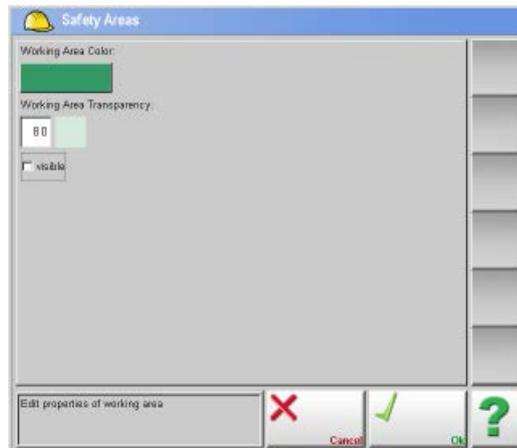


Fig.: Window for properties of the working area

Working Area Color

Shows the color of the working area.

Working Area Transparency

With the dialogue for [numerical input](#) enter the percentage of transparency used for the working area in the graphical display of the safety areas. The higher the value, the more transparent the working area will be; and the more clearly visible the safety areas will be, when displayed together with the working area.

Visible

Use this option to decide whether the working area will be shown in the graphical display of the safety areas or not.



Cancel

discards the settings and closes the window.



OK

confirms the settings and closes the window.

3.2.6.3.3 Graphical Display of the Safety Areas

The graphical display of the safety areas can be opened either in the View menu

Safety Areas or in the Setup menu Safety Areas using the button



It graphically displays the safety areas and the working area of the robot.

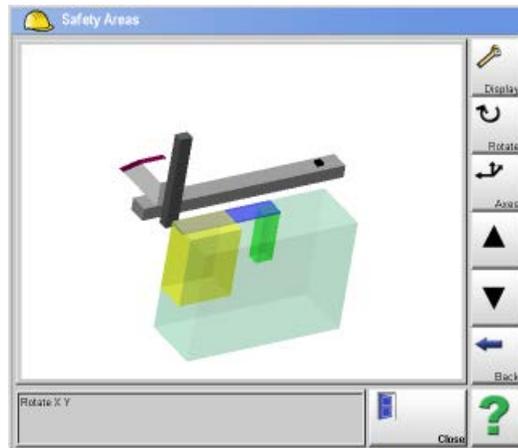


Fig.: Window showing the 3D view

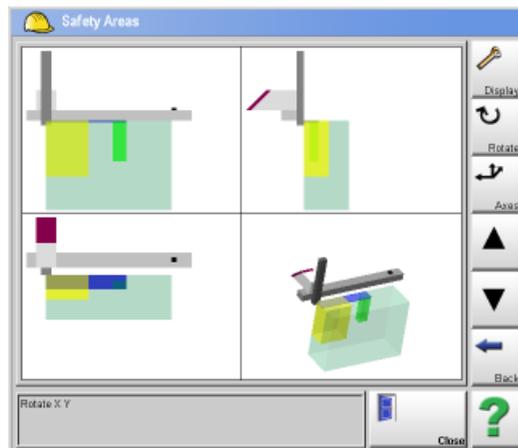


Fig.: Window showing the projected views

Press   to turn the transparency of the areas on or off.

Press   to toggle between 3D view and projected views.

Press   to reset the 3D view.

Press   or  to select the action to be carried out, when dragging on the display area or when pressing  or .

Press  followed by ,  or  to define in which axes the previously selected action will work, when triggered with  or . Multiple selections are possible.

Press  to quit the graphical display.

See also:

3.2.6.3 Setup Menu: Safety Areas

3.1.15.5 View Menu: Safety Areas

3.2.6.4 Cylinder Definition

The cylinder function is useful to ease the teaching of pneumatic cylinders by combining two peripheral outputs to a cylinder. Additionally you can use two peripheral inputs and a time limit to monitor the correct function of the pneumatic cylinder. Does the cylinder not approach the respective target position within the set time limit an error message will be displayed on the teachbox.

Up to 255 such cylinders are possible in one teach program.

A cylinder consists of:

- 1 to 2 peripheral outputs (O1, O2).
O2 is optional, and negates the status of O1.
- 0 to 2 peripheral inputs (I1, I2) for acknowledging or monitoring the action controlled by O1/O2 (optional), where I1 always acknowledges O1 and I2 always acknowledges O2.
- A limit for the time that is allowed to elapse between setting of Ox and acknowledgment of Ix.
Only if inputs are used.

If no cylinders have been defined previously, the editor will directly jump to the function to [edit cylinders](#). If any cylinders are defined already, the dialogue **Cylinder Definition** will be shown.

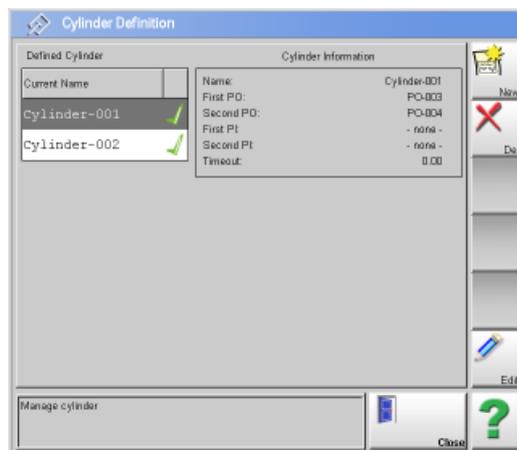


Fig.: Dialogue Cylinder Definition



is used to create a [new cylinder](#).



is used to [edit](#) the selected cylinder.



is used to delete the selected cylinder, if it is not applied in the teach program.



is used to close the window.

3.2.6.4.1 Create / Edit Cylinders

This is to edit an existing cylinder or to create a new one.

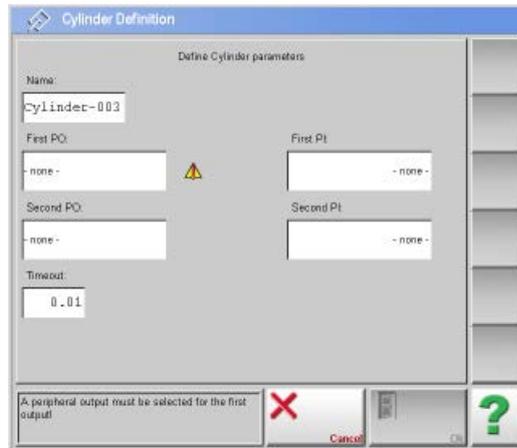


Fig.: Window to edit a cylinder

After tapping the respective input fields you can pick the desired peripheral outputs and peripheral inputs from selection windows.

If any conflict happens to occur because of some PO or PI being already used by the current or any other cylinder, the icon  will be shown next to the affected input or output. The status field on the left bottom of the window will contain a description of the conflict.

If inputs for position feedback are defined, it is possible to enter a time limit for the movements of the cylinder after tapping the input field **Timeout**. The desired time is entered using the dialogue for [numerical input](#). If no inputs are defined, the timeout must be set to zero.

A name for the cylinder can be entered after tapping the input field **Name** using the dialogue for [text input](#).



discards the entries and closes the window.



confirms the cylinder and closes the window.

See also:

3.3.5.3.1 Applying cylinders in a teach program

3.3.5.3 Cylinder function

3.2.6.5 Auto Switch Definition

Here the basic settings for the [AutoSwitches](#) can be done:

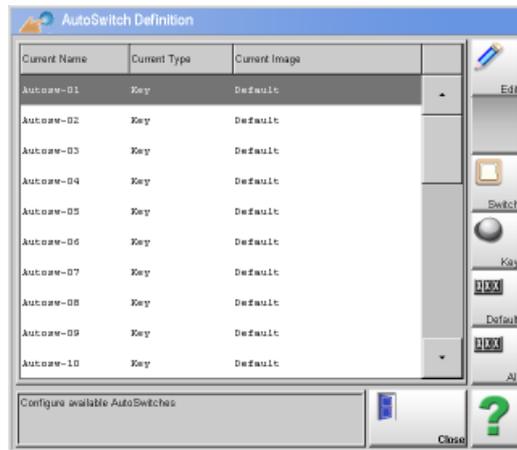


Fig.: Setup Menu - AutoSwitch Definition



and are used to define if the selected AutoSwitch has to act like a switch (maintained contact) or like a key (momentary contact).



is used to reset the selected AutoSwitch to the factory settings.



is used to reset all AutoSwitches to the factory settings.



is used to open the setup window of the selected AutoSwitch:



Fig.: Setup window of an AutoSwitch

Current Name

This is the name of the AutoSwitch. That name is not only displayed on the AutoSwitch, but also used as an identifier for the AutoSwitch in the Teach program. After tapping the input field, the name can be altered using the dialogue for [text input](#).

Current Type

Activating the desired option selects whether the AutoSwitch has to act like a switch (maintained contact) or like a key (momentary contact).

Current Image

This is the picture that will be shown on the AutoSwitch. Tapping the input field will display a list with the available pictures already stored in the Teachbox:

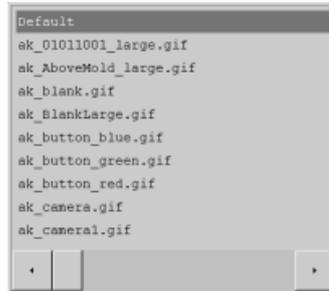


Fig.: selection list for AutoSwitch pictures

Tap on the name of the desired picture to select it.



can be used to import pictures from the USB stick for display on the AutoSwitches. Once imported, the pictures are available for all AutoSwitches in all Teach programs permanently.

Imported pictures not exceed a width of 190 pixels and a height of 64 pixels. The recommended size is **64 x 64 pixels** - as the standard picture. The following file formats are supported:

- BMP files with 256 colors (**recommended**)
- BMP files with 24 bits color depth
- GIF files with 256 arbitrary colors
- JPG files (not all compression rates and color depths)



can be used to remove the current picture from the memory of the Teachbox.



discards the entries and closes the window.



confirms the settings and closes the window.



An AutoSwitch will be displayed only if it is actually used in the Teach program in the robot.

See also:

- 3.3.10 AutoSwitches
- 3.3.7.8.6 Jump condition AutoSwitch
- 0 View Menu: AutoSwitch

3.2.6.6 Counter Names

Here you can rename the 64 counters of the teach program. The name of a counter should describe its function as good as possible, in order to ease reading the teach program.

The names are always valid for the teach program loaded in the Teachbox. This means the names of the counters can be different in every program.

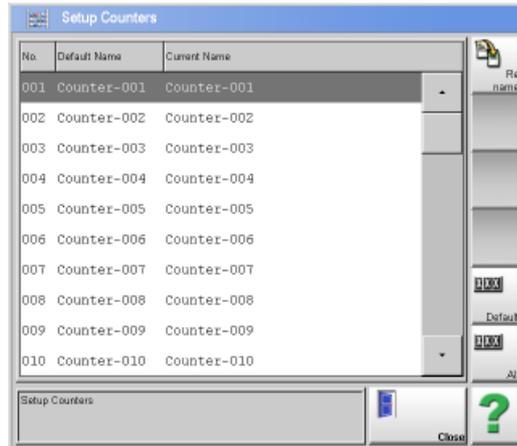


Fig.: Window for counter names



is used to rename the selected counter. The name is entered using the dialogue for [text input](#).



renames the selected counter to its standard name.



renames all 64 counters to their standard names.



For easier distinction of different kinds of elements you can activate the display of [prefixes](#) or [default names](#).



is used to close the window.

See also:

3.3.6 Counters

3.3.7.8.1 Condition: Counters

3.2.6.7 Conveyor Names

Here you can rename the conveyors controlled by the robot control unit. The name of a conveyor should describe its function as good as possible, in order to ease reading the teach program.

The names are always valid for the teach program loaded in the Teachbox. This means the names of the conveyors can be different in every program.

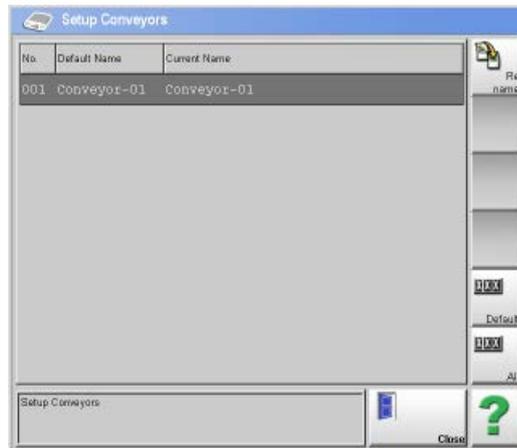


Fig.: Window for conveyor names



is used to rename the selected conveyor. The name is entered using the dialogue for [text input](#).



renames the selected conveyor to its standard name.



renames all conveyors to their standard names.



For easier distinction of different kinds of elements you can activate the display of [prefixes](#) or [default names](#).



is used to close the window.

See also:

3.3.5.1 Conveyor

3.2.6.8 Peripheral Output Names

Here you can rename the peripheral outputs controlled by the robot control unit. The name of a peripheral output should describe its function as good as possible, in order to ease reading the teach program.

The names are always valid for the teach program loaded in the Teachbox. This means the names of the peripheral outputs can be different in every program.

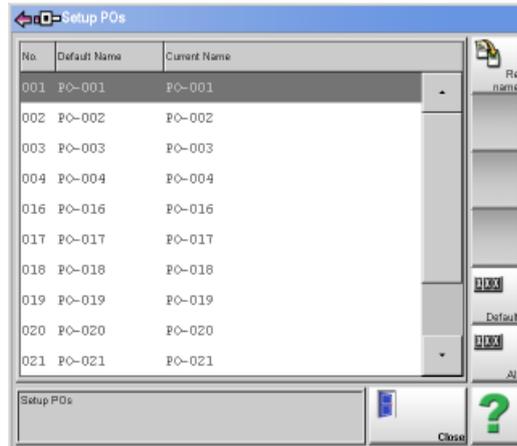


Fig.: Window for PO names



is used to rename the selected peripheral output. The name is entered using the dialogue for [text input](#).



renames the selected peripheral output to its standard name.



renames all peripheral outputs to their standard names.



For easier distinction of different kinds of elements you can activate the display of [prefixes](#) or [default names](#).



is used to close the window.

See also:
3.3.5.2 Peripheral Outputs

3.2.6.9 Peripheral Input Names

Here you can rename the peripheral inputs used by the robot control unit. The name of a peripheral input should describe its function as good as possible, in order to ease reading the teach program.

The names are always valid for the teach program loaded in the Teachbox. This means the names of the peripheral inputs can be different in every program.

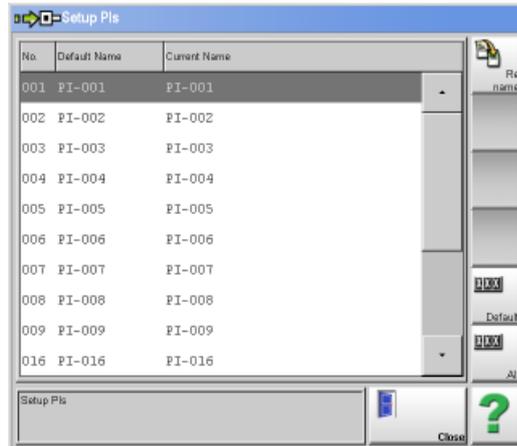


Fig.: Window for PI names



is used to rename the selected peripheral input. The name is entered using the dialogue for [text input](#).



renames the selected peripheral input to its standard name.



renames all peripheral inputs to their standard names.



For easier distinction of different kinds of elements you can activate the display of [prefixes](#) or [default names](#).



is used to close the window.

See also:

3.3.7.8.3 Condition: Peripheral Inputs

3.2.6.10 Stacking Sensor Names

Here you can rename the stacking sensors used by the robot control unit. The name of a stacking sensor should describe its function as good as possible, in order to ease reading the teach program.

The names are always valid for the teach program loaded in the Teachbox. This means the names of the stacking sensors can be different in every program.

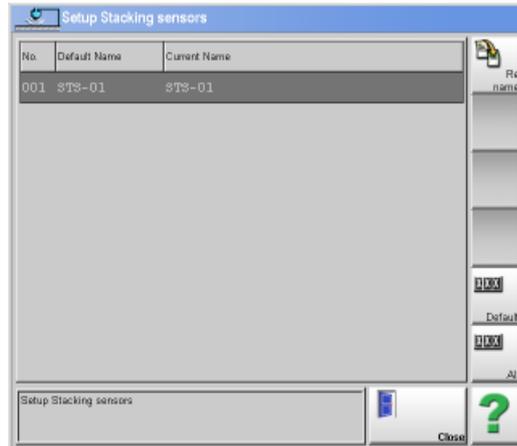


Fig.: Window for stacking sensor names



is used to rename the selected stacking sensor. The name is entered using the dialogue for [text input](#).



renames the selected stacking sensor to its standard name.



renames all stacking sensors to their standard names.



For easier distinction of different kinds of elements you can activate the display of [prefixes](#) or [default names](#).



is used to close the window.

See also:

3.3.7.8.11 Condition: Stacking Sensor

3.3.1.3.8 Axis Command: Stacking Sensor

3.2.6.11 Stopwatch Names

Here you can rename the 4 stopwatches of the teach program. The name of a stopwatch should describe its function as good as possible, in order to ease reading the teach program.

The names are always valid for the teach program loaded in the Teachbox. This means the names of the stopwatches can be different in every program.

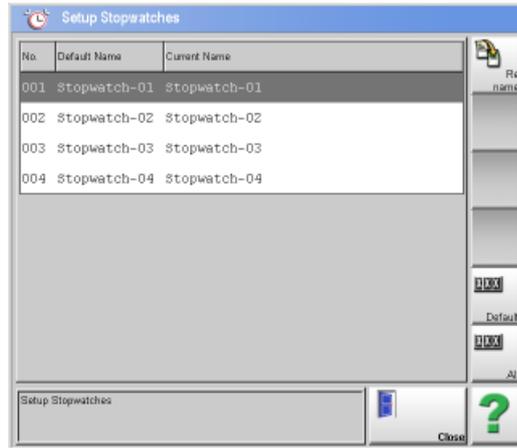


Fig.: Window for stopwatch names



is used to rename the selected stopwatch. The name is entered using the dialogue for [text input](#).



renames the selected stopwatch to its standard name.



renames all 4 stopwatches to their standard names.



For easier distinction of different kinds of elements you can activate the display of [prefixes](#) or [default names](#).



is used to close the window.

See also:
3.3.9.5 Stopwatch

3.2.6.12 Vacuum Names

Here you can rename the vacuum circuits controlled by the robot control unit. The name of a vacuum circuit should describe its function as good as possible, in order to ease reading the teach program.

The names are always valid for the teach program loaded in the Teachbox. This means the names of the vacuum circuits can be different in every program.

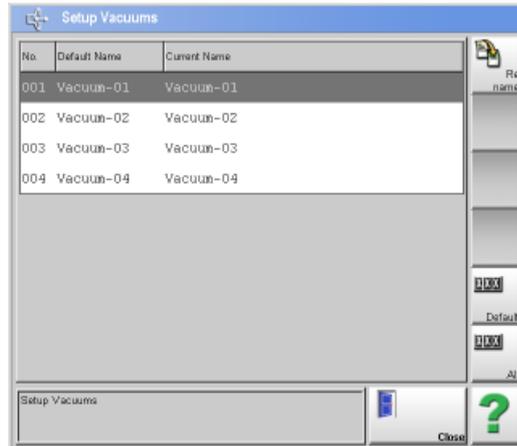


Fig.: Window for vacuum names



is used to rename the selected vacuum circuit. The name is entered using the dialogue for [text input](#).



renames the selected vacuum circuit to its standard name.



renames all vacuum circuits to their standard names.



For easier distinction of different kinds of elements you can activate the display of [prefixes](#) or [default names](#).



is used to close the window.

See also:

3.3.5.4 Vacuum

3.2.6.13 Gripper Names

Here you can rename the grippers controlled by the robot control unit. The name of a gripper should describe its function as good as possible, in order to ease reading the teach program.

The names are always valid for the teach program loaded in the Teachbox. This means the names of the grippers can be different in every program.

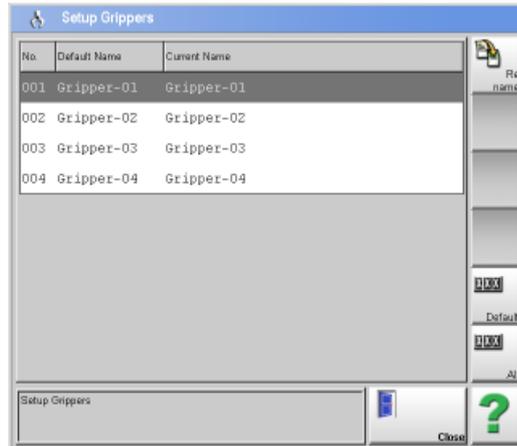


Fig.: Window for gripper names



is used to rename the selected gripper. The name is entered using the dialogue for [text input](#).



renames the selected gripper to its standard name.



renames all grippers to their standard names.



For easier distinction of different kinds of elements you can activate the display of [prefixes](#) or [default names](#).



is used to close the window.

See also:
3.3.5.6 Gripper

3.2.6.14 Vacuum Reference States

This is to set the state that the vacuum circuits must be in to enable operating mode Reference. The vacuum circuits have to be switched to the state by either the reference program or manually. If at the end of the reference travel one of the vacuum circuits happens not to be in the state set here, an error message will be shown; switching to operating mode Reference will not be possible in this case.

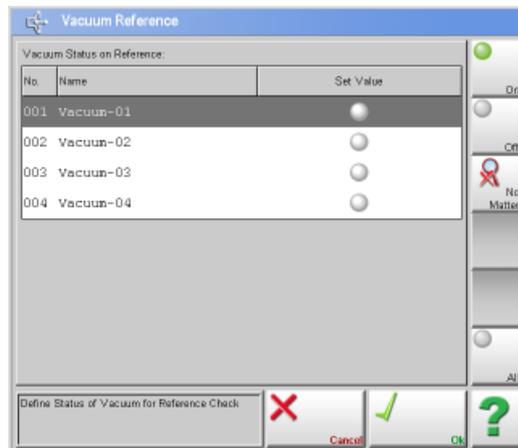


Fig.: Window for vacuum reference states

-  means the reference state of the vacuum circuit is ON.
-  means the reference state of the vacuum circuit is OFF.
-  means the reference state of the vacuum circuit is not checked.

  changes the reference state of the selected vacuum to ON.

  changes the reference state of the selected vacuum to OFF.

  sets the reference state of the vacuum to NOT CHECKED.

  or   or   are used to set the reference states of all vacuum circuits to the state of the selected one.

  discards the entries and closes the window.

  confirms the settings and closes the window.

See also:
3.3.5.4 Vacuum

3.2.6.15 Gripper Reference States

This is to set the state that the grippers must be in to enable operating mode Reference. The grippers have to be switched to the state by either the reference program or manually. If at the end of the reference travel one of the grippers happens not to be in the state set here, an error message will be shown; switching to operating mode Reference will not be possible in this case.

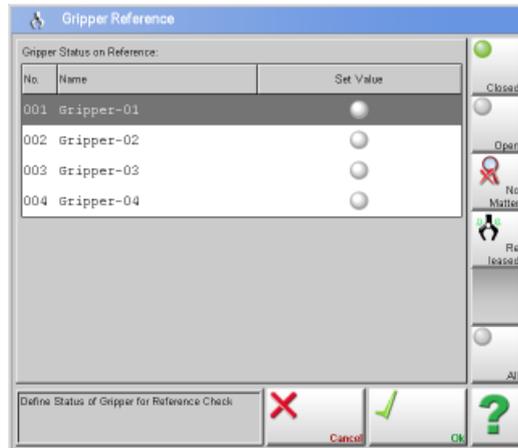


Fig.: Window for gripper reference states

- means the reference state of the gripper is OPEN.
- means the reference state of the gripper is CLOSED.
- means the reference state of the gripper is RELEASED.
- means the reference state of the gripper is not checked.

Close changes the reference state of the selected gripper to CLOSED.

Open changes the reference state of the selected gripper to OPEN.

Re lease changes the reference state of the selected gripper to RELEASED.

No Matter sets the reference state of the selected gripper to NOT CHECKED.

All, **All**, **All** or **No Matter** are used to set the reference states of all grippers to the state of the selected one.

Cancel discards the entries and closes the window.

Ok confirms the settings and closes the window.

See also:
3.3.5.6 Grippers

3.2.6.16 Setup Standard Vacuum Pressure

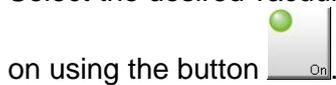
Here you set the threshold under-pressure that is used by the [vacuum monitoring](#) and the [condition vacuum](#) to determine whether the parts on the respective vacuum circuit are sucked on or not. This threshold must be set for every product or Teach program. The settings for vacuum pressure will be automatically stored and loaded as part of the respective Teach program.



This setting is **not** valid, when the threshold has been set to a different value than STANDARD with the [Vacuum Level command](#) in the Teach program.

The suggested method for doing the pressure setting is:

- 1) Select the desired vacuum circuit by tapping on the respective scale, then switch it



- 2) Put all parts onto all suction caps of the vacuum circuit.
- 3) Use the slider and the up/down buttons on the right side of the window to set the threshold to a value just smaller than the displayed value of the current under-pressure.

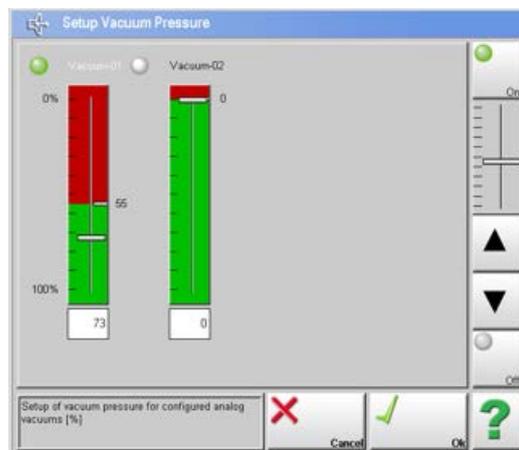


Fig.: Display with part on vacuum 1 and the threshold value set

The big bar and the field below the scale show the current under-pressure, the small bar and the number right next to it display the set threshold value. At a correct setting the big bar must be located in the green area of the scale, when all parts of the vacuum circuit are sucked on.

4) Remove one part from one suction cap of the vacuum circuit.



Fig.: Display with a part missing on vacuum 1 and the threshold value set

At a correct setting, now, the big bar must be located in the red area of the scale, and the field below the scale must be red, too. If this does not apply, the setting of the threshold under-pressure must be adapted accordingly.

After a successful setting you can switch off the vacuum circuit using the button 



discards the entries and closes the window.



confirms the settings and closes the window.

See also:

[Vacuum](#)

[Vacuum monitoring](#)

[Condition: vacuum](#)

3.2.7 Part program Settings in the PART PRG SETUP menu

3.2.7.1 Positions

Up to 255 different axis positions for the robot can be predefined here in each part program. This is especially advisable for positions that are approached fairly often at various locations in the teach program. In addition, offline teaching is made easier.

The predefined positions can be called up later using the selection field **Position** in the windows of the axis commands.

If no positions have been defined previously, the editor will directly jump to the function to [Create New Positions](#). If any positions are defined already, the dialogue **Positions** will be shown.

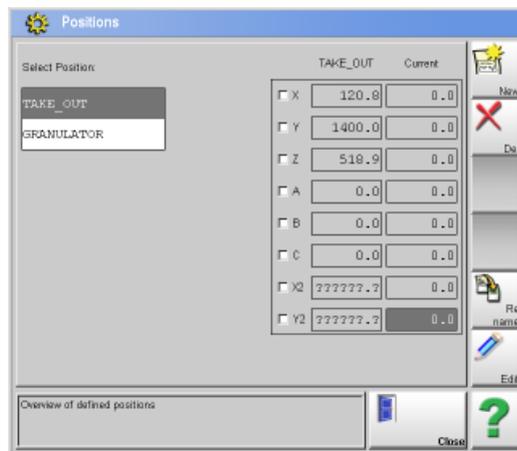


Fig.: Dialogue for Positions



is used to create a [new position](#).



is used to [edit](#) the selected position.



is used to delete the selected position, if it is not applied in the teach program.

After activating an axis with a this selected axis can be moved to its target within the selected position.



is used to close the window.

3.2.7.1.1 Create / Edit Positions

Here you can edit an existing position or create a new one.



Fig.: Window to edit positions

Each single axis is activated with a and is moved one after the other until the robot is on the desired position.

The desired name for the position can be entered after tapping the field **Position name** using the dialogue for [text input](#).



discards the entries and closes the window.



confirms the position and closes the window.

3.2.7.2 Labels and Subroutines

This displays a list of all the labels and subroutines that exist in the respective part program.

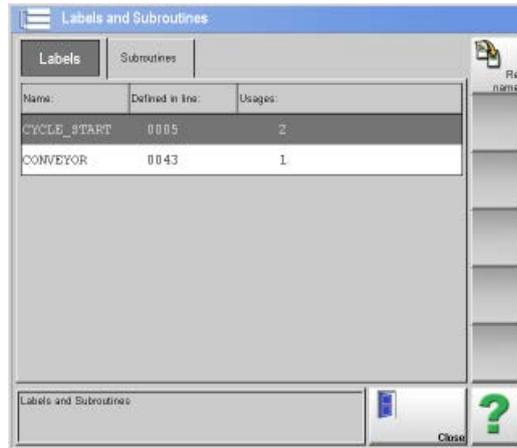


Fig.: Window for labels and subroutines

For each label (subroutine) the line number is displayed and the number of references to it.



is used to rename the selected label (subroutine). The desired name is entered using the dialogue for [text input](#). Commands accessing the renamed label (subroutine) are changed accordingly automatically.



is used to close the window.

See also:

3.3.7.2 Label

3.3.7.4 Call - Subr - Ret

3.2.7.3 Placing Program

Refer to [Placing Programs](#).

3.3 Teach Commands Of The Texteditor

3.3.1 Axes

3.3.1.1 Axes Selection

At robots with secondary axes press the axis button longer, if you want to display the buttons for toggling between main axis and secondary axis.

3.3.1.2 Terminology

3.3.1.2.1 Pneumatical Axes

Pneumatical axes are axes with pneumatical drive. They are controlled with magnetic valves; position feedback is done with position limit switches.

3.3.1.2.2 Numerical Axes

Numerical axes are axes with numerical position feedback and control. On R8.3 robots numerical axes are driven by synchronous servo motors.

3.3.1.2.3 Path Axes

Path axes are **numerical** axes which are included in a path axes calculation of a part program. Path axes of the same part program can be move coordinated with each other using 3D and Curve movements.

The main axes X, Y, Z of a standard robot are included in the path axes calculation of the robot program **Robot Prg.** by default.

You can assign numerical axes to the path axes calculations of the different part programs of the current Teachprogram using the Setup menu [Axes Definition](#).

3.3.1.3 Move



This tab provides teach commands to teach movements of the selected axes.

What kinds of movements are available is depending on the kind of drive of the axis and if it is a [path axis](#).

After selecting the kind of movement, the axis is moved to the desired target position using the drive keys.

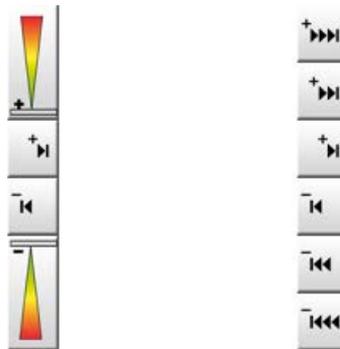


Fig.: Buttons for moving axes

If there are [predefined positions](#) available, alternatively one of these positions can be picked from the selection list. In this case the axis is taken to the selected positions using the buttons shown below. The axis will stop automatically when it has reached the position.



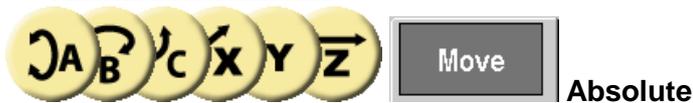
Fig.: Buttons to move axes to predefined positions

Whether the numerical axes can be moved using variable or fixed manual speeds is being set in the Init Menu [Setup: Robot Settings](#).



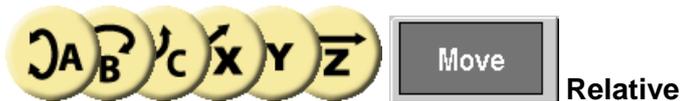
is used to confirm and write the command to the teach program.

3.3.1.3.1 Absolute Movement



This instruction is used to approach a particular position with the corresponding axis.

3.3.1.3.2 Relative Motion



If a position was approached with a numerical axis, this function can be used to move the axis relative to this position. This instruction is identified by the letter "R" after the position entry.

Example:

Z: 500.0
X: 100.0
Z: 100.0R

With the Z axis, move to position 500.0 mm, and with the X axis, to position 100.0 mm. Starting from this position, move the Z axis to position Z = 600.0 mm. Now if the first Z position is modified, the second Z position will also shift by the same value, but the distance between the two positions will remain constant and exactly 100.0 mm.

3.3.1.3.3 Parallel Movements

A parallel movement is the simultaneous motion of several axes.

As a parallel command simply causes command processing to acknowledge the command before its execution is finished, for a coordinated ending of all previously started commands you have to use a [WAIT SYNC](#) command after every parallel movement. This ensures that all previously started commands have to be finished before the next command is executed.

The parallel commands are not available for [path axes](#). You have to use [3D movements](#) to execute simultaneous movements with several path axes.

Two kinds of parallel movements are available:

- Synchronous
- Parallel At

3.3.1.3.3.1 Synchronous



When this instruction is started, the following (travel) instruction is also started simultaneously.

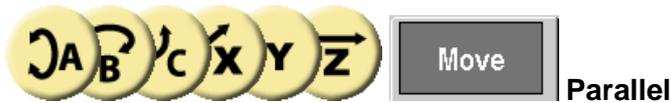
Synchronous parallel motion is identified by the symbol "*" after the position to be approached.

Example:

```
C: 0.0*
A: 0.0
WAIT SYNC
```

The A and C axes are started simultaneously and are moved to their 0 position. The instruction [WAIT SYNC](#) ensures that both axis motions are completed before the program is continued.

3.3.1.3.3.2 Parallel at



This function is used to drive the respective axis alone to an intermediate position (**Intermediate Pos.**). When this intermediate position has been passed, the next (travel) instruction is started, while the axis continues traveling to its target position (**Final position**) in parallel.

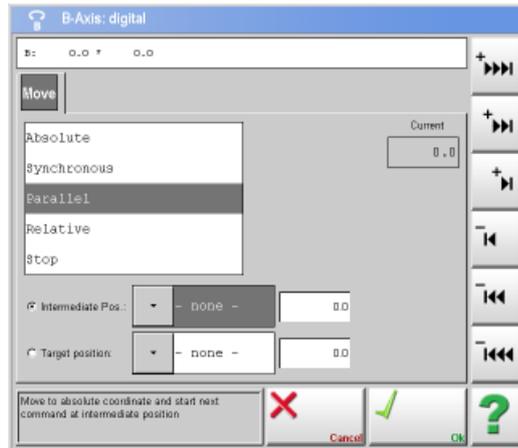


Fig.: Parallel at

To teach this command, first the **Intermediate Pos.** is activated by tapping it. Now the axis is moved to the position, where when passing it in automatic mode, the next instructions shall be started.

The next step is to activate the **Final position** by tapping it, and to move the axis to its target position.

Now the command can be added to the teach program using .

 is used to exit without changing the teach program.

Example:

```
Z: 500.0* 200.0
C: 90.0
WAIT SYNC
```

The Z axis will travel to the position 500.0, and from position Z = 200.0 the instruction C: 90 will be processed in parallel. The instruction [WAIT SYNC](#) ensures that both axis motions will be carried to the end before processing of the program is continued with the next instruction.

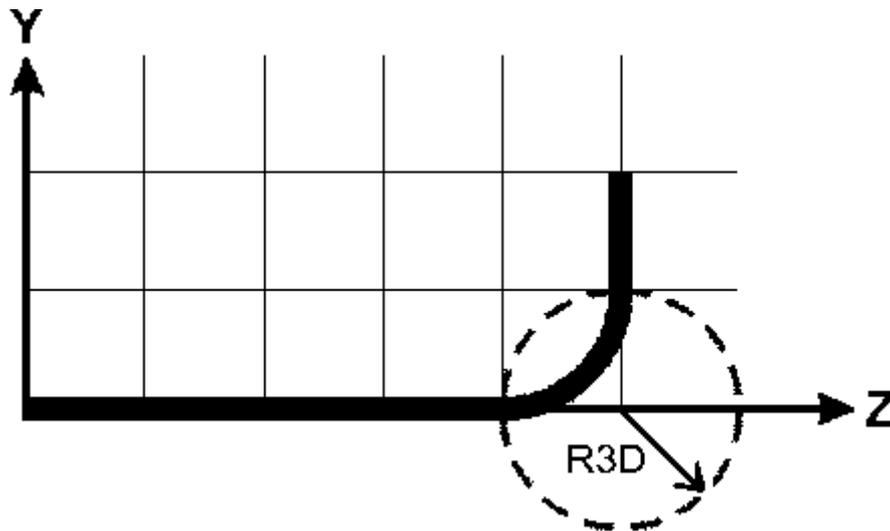
3.3.1.3.4 Curve



With the curve function, the relevant position of a [path axis](#) is not approached, but is passed using an adjustable radius R3D. The function is identified by the symbol "*" before the respective axis.

Example:

```
*Z: 500.0
Y: 200.0
```



The position with the coordinates Z = 500.0 and Y = 200.0 is approached in one motion. The Y axis is started around R3D before reaching the position Z = 500.

The curved movement can also be used to start a parallel motion of a non path axis during the motion of a path axis when the path axis moves past a given position.

Example:

```
*Z: 500.0
C: 90.0*
Z: 1000.0
WAIT SYNC
```

The Z axis travels in one motion to the position 1000.0. When the Z axis travels past the position 500.0, pivoting of the C axis is started in parallel. Parallel travels have to be completed with a [WAIT SYNC](#) command.

Similarly, peripheral outputs for example can also be switched when a given position is passed, without the motion of the axis concerned being affected.

3.3.1.3.5 Stop



Processing of this instruction in automatic mode produces an immediate stop of the selected numerical axis.

See also:

3.3.5.7 Vacuum Monitoring

3.3.1.3.6 Soft Torque Push

The Soft Torque Push function enables the robot to take out long parts with the respective axis being pushed by the ejector.

The EOAT is moved to the piece and the piece is being fixed on the gripper already before the ejector starts moving forward.

The PUSH command is used to start the ejector and to let the respective axis being pushed by the ejector. This is done with adjustable torque assistance, thus eliminating the resistance of the robot mechanicals.

From the Intermediate Position given by the command, the axis will use either maximum velocity or the last measured velocity to travel to the target position. In the second case, regarding the braking ramp of the ejector, the Intermediate Position should be placed in a way to let the robot continue its travel with sufficient speed.

The permit for **Ejector Forward** must be turned off in the Teach program by the user.

The sequence for teaching (shown for the X-axis) is:

- 1) Use **Move** **Absolute** to select the X-axis.
 - 2) Travel to the part removal position.
 - 3) Confirm the command with
 - 4) Program everything that is necessary to take the part (switch on vacuums, close grippers).
 - 5) Use **Push** to select the Soft Torque Push function.
- The following window will appear:

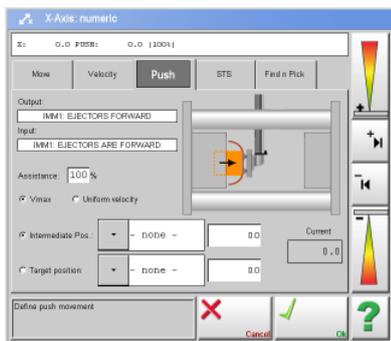


Fig.: Window for the Soft Torque Push function

- 6) Activate the option **Intermediate Pos.** and move the X-axis to the end position of the ejector.
- 7) Activate the option **Final position** and move the X-axis to the final kick stroke position in the mold, from which the vertical arm can be moved up and out of the mold.
- 8) Use the value **Assistance** to set the amount of torque assistance from 0 to 120% of the base settings.
- 9) Use the options **Vmax** and **Uniform velocity** to decide whether the movement from the intermediate position to the target position shall be carried out with either maximum speed or the last measured (ejector-) speed.
- 10) Press to add the command to the program.

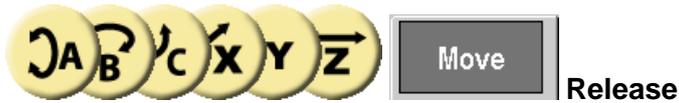
Example:

```

...
*Y: 1000.0                travel to the part
Vacuum-01 = ON
X: 600.0
WAIT Vacuum-01 = ON      fix the part
X: 100.0 PUSH 300.0 (100%) travel synchronously with the ejector
PERMIT IMM:EJECTORS FORWARD!=OFF switch off Ejector Forward
...

```

3.3.1.3.7 Release (optional)



The use of this function requires that the respective axis is equipped with a suitable gearbox.

 and  are used to determine whether the command shall switch the axis release on or off.

 is used to add the command to the Teach program.

When the release is activated, the respective axis can be moved by auxiliary force.

After deactivating the axis release, the position of the axis will be controlled actively again.

Example:

```
X: RELEASE = ON
FUNCTION IMM: EJECTORS FORWARD! WAIT
X: RELEASE = OFF
```

Here the X-axis is pushed by the ejector. The ejector is started after the axis has been released. When the ejector has arrived at its most forward position, the axis release is being deactivated again.

See also: 3.1.12.2 Axis Release in Manual Mode

3.3.1.3.8 **Stack Sensor (optional)**



This function makes it possible to react to obstacles in travel of an axis by means of a stack sensor. The axis is traveled until either the stack sensor is actuated or a teachable maximum position is reached.

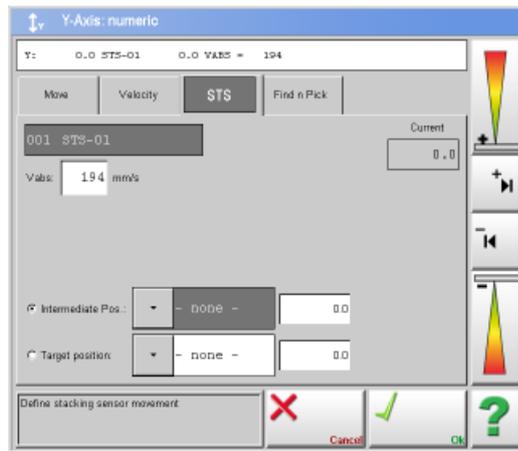


Fig.: Stack Sensor

At first the desired **Stack Sensor** is selected from the list box on the upper left of the tab.

From Position: is the position that is first approached absolutely at the current speed; this is where travel to the stack sensor starts.

To Position: is the position at which stack sensor travel ends when the stack sensor is not activated during travel.

Each position has to be selected by tapping on it, and has to be approached with the axis using the drive keys. Alternatively predefined positions can be selected.

Vabs: Absolute speed in mm/s, by which travel to the stack sensor is carried out between From Position and To Position. (is not affected by the override)

Current: shows the current position of the axis.



discards the changes and closes the window.



writes the command to the teach program and closes the window.

3.3.1.3.9 Find 'n' Pick



With this function the axis will use one of the analogue vacuum monitors to search for the part to take or for any other obstacle. This may be useful when approaching the takeout position in molding machines with big tolerances on the mold open position. The function is also an alternative for a “real” stack sensor.

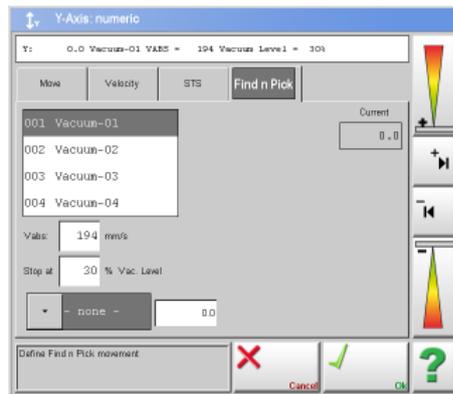


Fig.: Find 'n' Pick

Use the list box on the upper left corner of the window to select the desired **vacuum circuit**. During automatic operation the selected vacuum circuit will be automatically switched on when execution of the teach command begins.

Vabs

Determines the absolute speed in mm/s used for the search movement. (Not influenced by the Override)

Vac. Level

This is the trigger level used for the vacuum monitor of the selected vacuum circuit. When the vacuum under pressure exceeds this level, the robot recognizes the part/obstacle and stops the search movement. In order for the robot to recognize the part/obstacle as early as possible, the value should be set as closely as possible above the value shown for the empty vacuum circuit shown in the View menu [Vacuum Pressure](#).

End position

Use the input box on the bottom of the window to determine the position where the movement ends, if the trigger level of the vacuum monitor has not been exceeded during the search. After tapping the input box, you can use the drive buttons to move to the desired position. Alternatively you can select a predefined position.

Current

This shows the current position of the axis.



discards the entries and closes the window.



adds the command to the teach program and closes the window.

Example:

```

...
*Y: 567.8
*X: 234.5
 X: 345.6 Vacuum-01 VABS = 100 Vacuum Level = 34%
 Wait Vacuum-01 = ON
 X: 123.4 PUSH: 200.0 (100%)
...

```

In this example the robot tries to find the part in a mold with a varying open position. It uses the Find 'n' Pick command to search for the part between X=234.5 and X=345.6. After the robot has sucked on to the part ([Wait Vacuum](#)), it lets itself being pushed back by the ejector using the [Push](#) command.

3.3.1.3.10 Follow Conveyor (optional)



Additional hardware for measuring the velocity of the conveyor is prerequisite for the Follow Conveyor function.

The Follow Conveyor function enables the robot to follow a conveyor synchronously. Thus the robot can place products on a moving conveyor, or take them away from there.

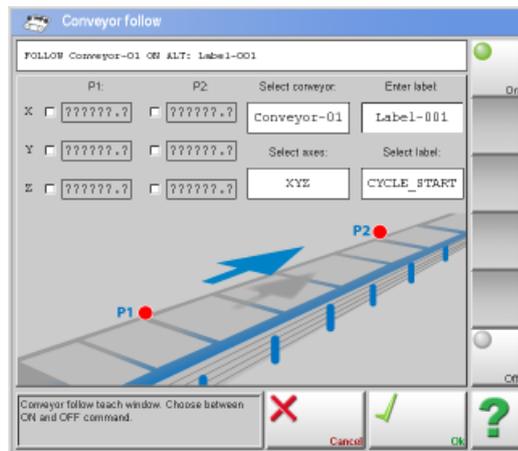


Abb.: Window for the Follow Conveyor commands

After opening the window in the text editor you can use the buttons  and  to determine whether you want to create the command for starting the conveyor following movement or the command for stopping the conveyor following movement.



discards the entries and closes the window.



adds the command to the teach program and closes the window.

The indications in parentheses (1, 2, 3, 4, 5, ALT) in the description of the Follow Conveyor commands relate to the schematic of a conveyor following movement in the figure further down this chapter.

FOLLOW Conveyor-xx ON ALT Label

This command starts the conveyor following movement. The command will “wait” until the robot moves synchronously with the conveyor (1 to 2). From this moment on, the program will be executed parallel to the conveyor following movement (2 bis 5). For the axes only relative movements are allowed during the conveyor following movement, they will be executed additionally.

The robot will not travel to the positions P1 and P2 during the actual conveyor following movement. The line between the coordinates of P1 and P2 only serves as a vector for the direction of the conveyor following movement.

In order to teach program the positions P1 and P2, open the window for the Follow Conveyor commands and activate every single axis position one after the other with a ✓. Use the axes drive buttons to move the currently activated axis until you reach the desired position. Use two distant points for P1 and P2, located close above the same border of the conveyor, in order to make the vector as parallel to the actual moving direction of the conveyor as possible.

The normal plane of this vector at P2 is used as a limit for the conveyor following movement. If the position of the robot touches this limit during the conveyor following movement, the alarm 019-001-xxx Conveyor-xx FOLLOWING TARGET POSITION REACHED will be triggered and the program will continue at the label for the alternative routine. This will not stop the conveyor following movement; you will rather have to program an escape movement in the alternative routine in order to make stopping possible (ALT).

Thus the position of P2 should be selected to enable the planned conveyor following movement (1 to 5) to be finished before reaching the normal plane, and on the other side leave enough space for the escape movement (ALT) of the alternative routine to be finished before reaching the stroke limits.

For the label of the alternative routine you can either tap the input field **Enter label** to input a new label using the dialogue for [text input](#), or you can select an existing label using the selection field **Select Label**.

FOLLOW Conveyor-xx OFF

This command ends the conveyor following movement (5, ALT).

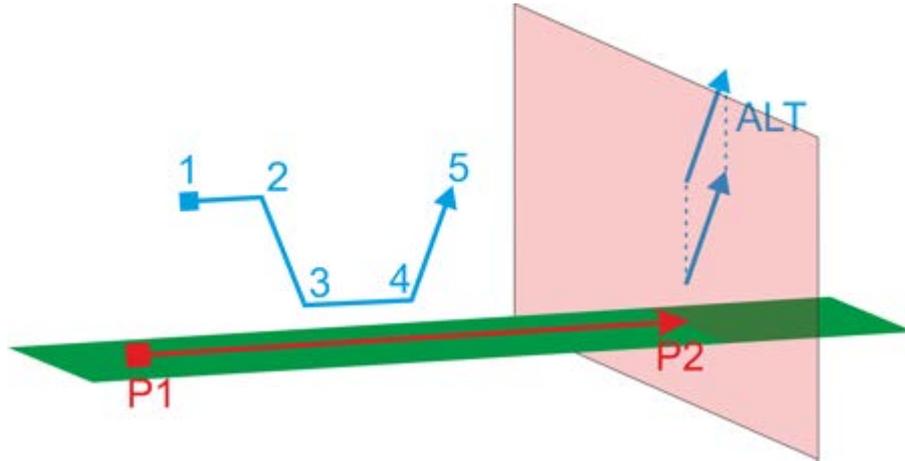


Fig.: Schematic of a conveyor following movement

Example:

A part is taken away from a continuously moving conveyor. The indications in parentheses relate to the schematic of a conveyor following movement in the figure.

...	
*Z: 92.1	Move to the start position (1) above the conveyor, where the conveyor can still move without colliding and where also is enough space upwards for the escape movement of the alternative routine.
*X: 123.4	
Y: 300.0	
-	
WAIT PI-001 = OFF	(1) Wait for the positive slope of a trigger signal
WAIT PI-001 = ON	(1) (light sensor, camera, etc.).
-	
FOLLOW Conveyor-01 ON ALT: TOO_FAR	(1) to (2) Start the conveyor following movement and switch to the next line once the robot moves synchronously with the conveyor.
Vacuum-01 = ON	(2) Switch on the vacuum.
Y: 200.0R	(2) to (3) Move down.
WAIT Vacuum-01 = ON	(3) to (4) Wait until the part is sucked on.
Y: -200.0R	(4) to (5) Move up with the part.
FOLLOW Conveyor -01 OFF	(5) End the conveyor following movement.
-	
...	
JMP ...	
-	
TOO_FAR:	In case the conveyor following movement did not end before the normal plane at P2:
Y: -200.0R	to (ALT) Execute an escape movement,
FOLLOW Conveyor -01 OFF	(ALT) and stop the conveyor following movement.
...	Execute whatever else shall happen in this error case.

3.3.1.3.11 **Mold Opening Synchronization (optional)**

Mold Opening Synchronization (MOS) enables the robot to enter the mold during its opening movement, thus saving valuable time.

A way measuring system the opening width of the mold of the IMM is necessary for this feature.

The command WAIT MOLDPOSITION is used to wait until the mold has opened wide enough for the robot to enter.

Now the (normally) X-axis can follow the moveable plate of the mold with a teachable distance using the MOS command. The MOS command behaves like a Synchronous command (the respective axis is taken out of the pre-calculated path), enabling the robot to enter the mold with the Y-axis simultaneously. The MOS command is terminated by the signal **Mold Is Open** of the IMM.

Finally a WAIT SYNC command **must** be introduced to end the parallel movement. This also includes the MOS-axis to the pre-calculated path again.

The two commands for mold opening synchronization are added to the program like this:

WAIT MOLD POSITION

This is used to wait until the mold is opened wide enough for the robot to enter. After

accessing the command with    **Mold Position** you have to pick one of the operators \leq , \geq , $<$, $>$, $<>$, $=$ to select whether you want to wait until the actual position of the mold is equal, below or above the selected value. Move the Mold of the IMM to the desired position to select the value for the command.

Press  to add the command to the teach program.

RELATIVE TO MOLD

This axis command lets a robot axis (most likely X) follow the mold in a teachable distance.

In order to teach the command, open the mold and access the command with



R to mold. Now move the robot to the desired distance from the moving plate of the mold, which the robot should use to follow.

Press  to add the command to the teach program.

Example:

```

...
X: 50.0                travel X to the waiting position over the mold.
WAIT MOLDPOSITION > 400.0  wait until the mold is opened sufficiently for the robot to enter
X: 100.0 R TO MOLD        let the X-axis follow the mold with the chosen distance; and at the
                           same time execute the next commands
                           (like "synchronous").
Y: 800.0                travel Y to 800.0 mm (while X is following the mold).
WAIT SYNC                wait until all axes (X and Y) have finished their movements.
...

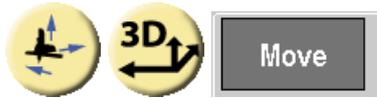
```

 The distance displayed within the MOS command does not show the actual value of the distance between EOAT and mold plate, but shows the calculated difference between the value measured by the measurement system for the width of the mold opening and the value of the axis position.

 The value displayed within the command WAIT MOLDPOSITION only relates to the actual value of the width of mold opening, if the MOS-Offset has been adjusted for the mold currently in use.

 The measured value of the **mold position** must not change, when one of the signals **Mold Is Open** or **Mold Is Closed** is active. This would trigger the error message Mold Monitoring.

3.3.1.4 3D Motions



In 3D motions, a position is approached simultaneously with the selected [path axes](#).

3D-motions can be executed as

- 3D-Absolute Motion
- 3D-Curve
- 3D-Relative Motion

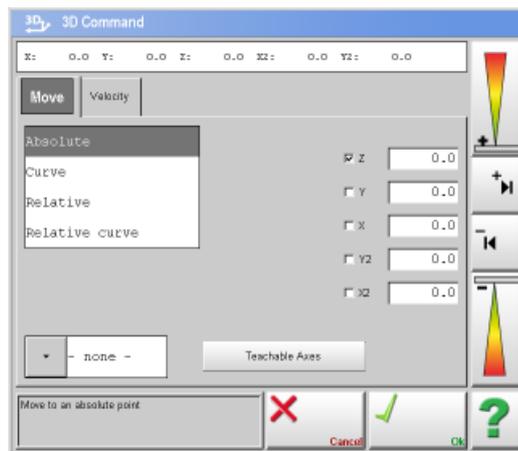


Fig.: Window for 3D-Motions

 can be used to open the window for selecting the path axes to participate in the 3D movement.



Fig.: Axes selection for 3D commands

Use  to confirm the selection.

Use  to close the window and discard the selection.

After selecting the type of motion, the position is approached with the participating path axes. Use the window for 3D motions to activate every single axis with a ✓ and move one after the other, using the drive keys, until the desired target position is reached.

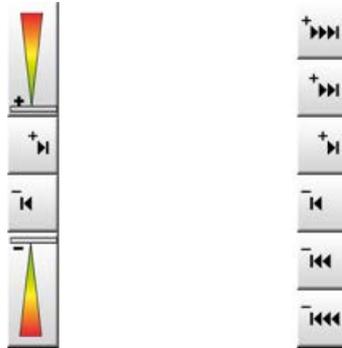


Fig. Drive keys for numerical axes

When all axes are on target position, use  to add the command to the teach program.

Use  to close the window, without changing the teach program.

3.3.1.4.1 3D Absolute Motion



In automatic operation, in processing of this instruction, the [path axes](#) are moved to the indicated position simultaneously in a straight movement.

3.3.1.4.2 3D Relative Motion



In automatic operation, in processing of this instruction, the [path axes](#) are moved to the indicated position simultaneously in a straight movement.

This function is identified in the teach program by the letter "R" after the axis coordinates.

The axis coordinates in this instruction are added to the axis coordinates of the starting position of the axes, i.e., motion is executed relative to this starting position.

The position to which the relative motion is to refer must already have been taught before this function is entered.



If the starting position is changed, the position that is approached with the 3D relative motion is also changed. The distances between the two points remain the same.

3.3.1.4.3 3D Curve



This function is identified in the teach program by the symbol "*" in front of the axis coordinates.

In automatic operation, when this instruction is processed, the indicated position is bypassed using the adjustable radius [R3D](#) and switched to the next instruction.

See also: 3.3.1.3.4 Curve

3.3.1.5 Speed



This tab provides teach commands to set speed and acceleration for subsequently taught movements of the specific axis.

3.3.1.5.1 VMAX



This function is used to set the maximum speed as a percentage of the configured maximum speed of a numerical axis. The actual axis speed in automatic operation is in addition affected by the [Override](#). The Vmax instruction remains effective until the next Vmax or [Vabs](#) instruction, and affects all subsequently taught motions of the axis concerned.

The value of VMAX can be altered by moving the slider, or can be input using the [numerical Input](#) after tapping the input field.



is used to confirm and write the command to the teach program.

3.3.1.5.2 VABS



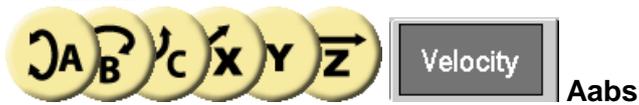
This function permits [Override](#)-independent speed setting in mm/s of a numerical axis for adaptation to secondary motions (for example, ejector motion in piece transfer). The Vabs instruction is effective until the next [Vmax](#) or Vabs instruction, and affects all subsequently taught motions of the axis concerned.

The value of VABS can be altered by moving the slider, or can be input using the [numerical Input](#) after tapping the input field.



is used to confirm and write the command to the teach program.

3.3.1.5.3 AABS



This function permits the maximum acceleration or retardation in mm/s² of a numerical axis to be adjusted when the configured acceleration (deceleration) is too big for the application.

This instruction is effective until the next AABS instruction and affects all subsequently programmed motions of the respective axis.

The value of AABS can be altered by moving the slider, or can be input using the [numerical Input](#) after tapping the input field.



is used to confirm and write the command to the teach program.

3.3.1.5.4 R3D



R3D is the radius for curve motions motion in mm. This is the distance before the taught end position at which the motion of the next axis instruction is started.

The parameter has to be taught in the program routine **before** the 3D motion to be affected.

The value of R3D can be altered either by using the slider, or by typing the value using the [numerical input](#) dialogue after tapping the input field.



is used to confirm and write the command to the teach program.

3.3.1.5.5 V3D



This parameter limits the resulting path speed of [path axes](#) movements. The path speed is the geometrical sum of the speeds of the involved single axes. The value of V3D is given in mm/s.

V3D as a limiter overrules the settings performed with [VMAX](#), [VABS](#) and the [Override](#).

A limit set with V3D is valid until the next V3D command in the teach program. Without a V3D command, the path speed in a teach program remains unlimited.

The parameter has to be programmed in the program sequence **before** the path axes movement to be affected.

The value of V3D can be altered either by using the slider, or by typing the value using the [numerical input](#) dialogue after tapping the input field.



is used to confirm and write the command to the teach program.

3.3.1.6 Wait Sync



This instruction is used to terminate parallel motions.

Program processing is stopped until all active axis motions have been completed.

3.3.1.6.1 Wait Path Sync



WAIT PATH SYNC is a version of WAIT SYNC especially designed for moving [path axes](#). The pre-calculation of the path axes is stopped until the last path-movement is active and all other axes are no longer active.

If at the beginning of the curve radius of the path axes still any other axis is not in its target position, the path axes will stop at their respective last target positions. The pre-calculation will be continued after all other axes have reached their target positions.

Example:

```
*Y: 200.0
  C: 90.0*
*Y: 0.0
*X: 0.0
*Z: 2000.0
  WAIT PATH SYNC
*Z: 0.0
```

At Y=200 the synchronous movement to C = 90 is started. If the C-Axes reaches its position more than the curve radius prior to Z = 2000 the robot will continue traveling to Z = 0 without stopping. If the C-Axes does not reach its position the robot stops at Z = 2000 and waits until C reaches its position. Only after that the robot will continue traveling to Z = 0.

3.3.1.7 Movements and Parameters for Scara Robots

The „normal“ (3D) Motion commands are also available for Scara robots. Those commands are used to move the robot within the spatial system of coordinates. With the additional angular movements it is possible to move the axes of the robot outside the spatial system of coordinates.

3.3.1.7.1 Pathmode



The Parameter Pathmode is used to determine how the robot moves to its target position.

The parameter has to be programmed in the program sequence **before** the movement to be affected. A setting made with PATHMODE is valid until the next PATHMODE command in the teach program.

Linear Mode:

The axes move to the target position coordinated in a way that the resulting movement draws a straight line in the spatial system of coordinates. The single axes cannot move with full speed. In linear mode the robot cannot change the bending orientation of its elbow joint β .

Joint Mode:

The axes move optimized for speed. The resulting movement in the spatial system of coordinates is not straight. Angular movements („in degrees“) will always be executed in joint mode.

Select the desired mode by tapping it.

Press  to add the command to the teach program.

3.3.1.7.2 Angular Motions - „in degrees“



Absolute in degree



Curve in degree

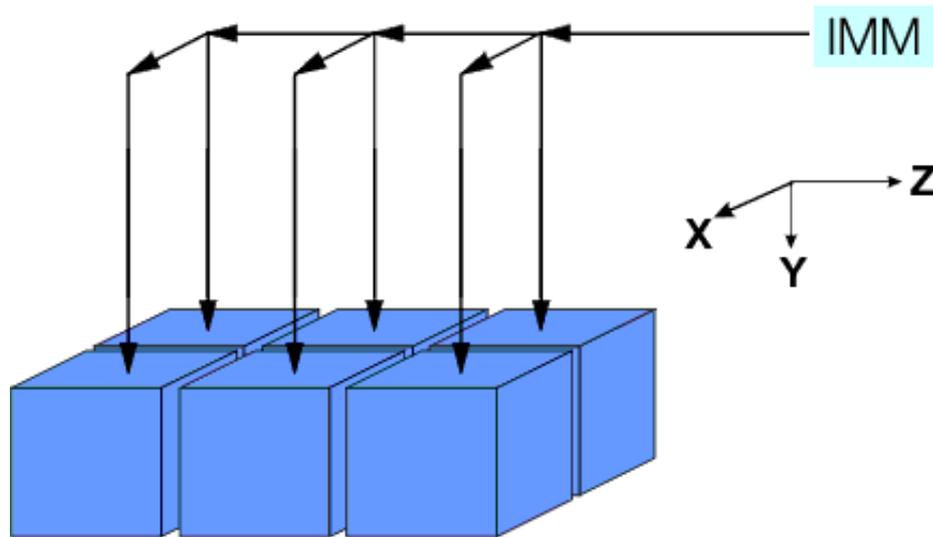
Use the angular motions to move the axes (joints) α , β , γ of a Scara robot outside the spatial system of coordinates in angular degrees.

The angular motions are [3D-motions](#).



Angular motions will always be carried out in [joint mode](#), regardless of the Pathmode set.

3.3.2 Placing Programs



Placing Programs are used for the automatic calculation and approach of several stacking positions. This makes it possible to stack parts without programming each position separately.

The **R8.3 control** offers the following possibilities for this:

- 16 freely definable placing programs
- up to 6 different placing sequences
- up to 6 different axis sequences
- approach of stacking positions with 3D function
- programmable box height
- programmable displacement
- approach stack sensors with adjustable speed
- stacking / un-stacking
- support of double arm robots

Programming a Placing Program has to be done in two steps:

- 1) [Creating](#) the Placing Program.
- 2) [Adding](#) the created Placing Program as a command to the teach program.

3.3.2.1 Create/Edit Placing Programs

For creating and editing of placing programs open the menu **Part Prg Setup – Placing Program**. The dialogue for selecting existing placing programs will be shown.

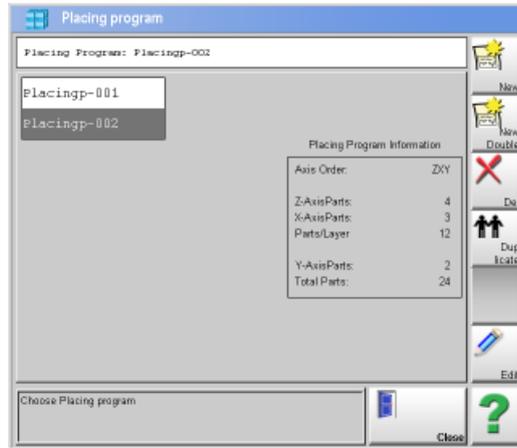


Fig.: Dialogue for selecting a placing program

In the selection list on the left side of the window you can select one of the existing placing programs. The table **Placing Program Information** shows how many positions in the different axes dimensions the selected placing program will travel to. If no placing programs have been defined in the current teach program yet, the message "No placing programs available" will be displayed instead of the selection list.



is used to create a new placing program.



is used to create a new placing program for a double arm robot that additionally contains the parallel axes X2 and Y2. Prerequisite for this is that the parallel axes have been [added to the path](#) in the current teach program.



is used to edit the selected placing program.



can be used to delete the selected placing program, if it is not used in the teach program.



can be used to create a copy of the selected placing program. The dialogue for [text input](#) will appear. Here you can input a name for the copy of the placing program,

or confirm the suggested standard name with .

When you create a new placing program, first the dialogue for axes selection will be displayed.

Editing of the other parameters of the placing program will be done using five tabs:

- 1. Position of the first piece, Name
- 2. Number and distances of the other pieces
- Options Axis sequence, placing sequence, 3D, box height
- STS Stacking Sensor
- Displacement Displacement



The dialogue for axes selection and the tabs for stacking sensor and displacement will not be displayed at a placing program for a double arm robot.

See also:

3.3.6.2 Reset Placing Counters

3.1.15.7 VIEW Menu: Placing Counters

3.3.2.1.1 Dialogue for Axes Selection

When creating a new placing program, this window is used to select the axes available for the new placing program.

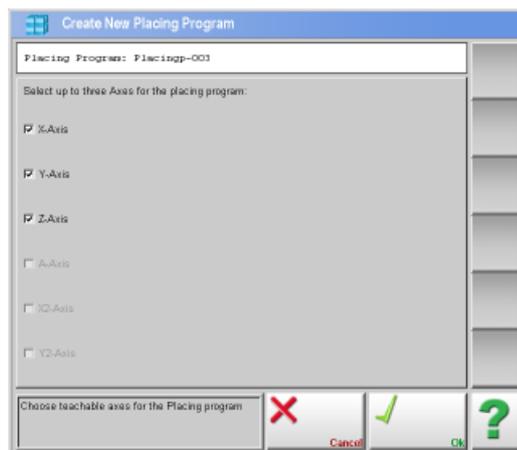


Fig.: Dialogue for axes selection

Select up to 3 axes to be used with the placing program with a ✓.

Press  to confirm the entries, the tab for the first stacking position will be displayed next.

Press  to cancel the process and discard the entries.



The dialogue for axes selection will not be displayed at a placing program for a double arm robot.

3.3.2.1.2 Tab 1. - 1st Position, Name

This tab is used to determine the stacking position of the first piece in the placing sequence; furthermore the name of the placing program can be changed here.

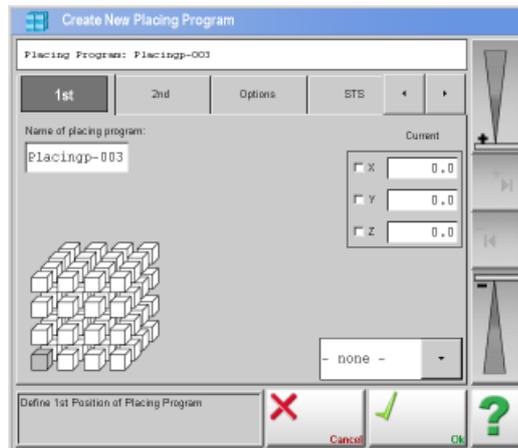


Fig.: Tab for the 1st position

Tap the input field **Name of placing program** and input a name for the placing program using the dialogue for [text input](#).

After selecting one of the axes, you can move it to its previously programmed 1st stacking position using the buttons for moving to an existing position.

If you want to change the axes positions for the first stacking position, first press the



button. Now select each axis and move it to its desired position, until all axes of the robot are on their first stacking position.

You can also use the selection field on the right bottom of the window to recall a [predefined position](#) for the first stacking position.

When all parameters on this tab are entered, either open the next tab by tapping it, or

accept the placing program by pressing .

Press  to cancel the process and discard the entries.

3.3.2.1.3 Tab 2. - Stacking Pattern

This tab is used to determine the stacking pattern.

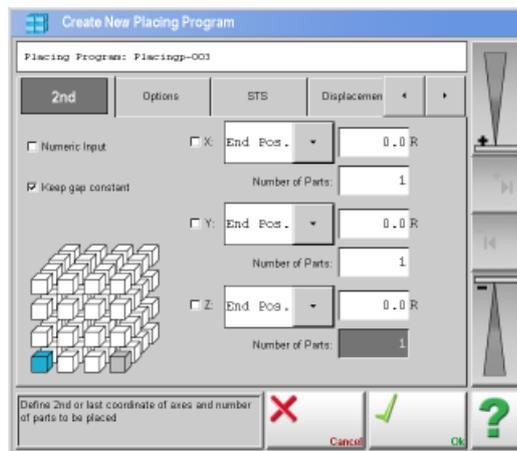


Fig.: Tab for the stacking pattern

In order to determine the stacking pattern, for each axis the number of pieces to be placed in the axis direction and the position of either the second or the last piece have to be specified. The second or last position is calculated as a distance relative to the first stacking position (this is indicated by the letter **R** after the positions)

i When the first position is changed, the entire placing program will be shifted and the stacking pattern remains intact.

Choose one of the axes and select whether you want to determine the position of the second piece (2nd Pos.) or the position of the last piece (End Pos.) in the respective axis direction.

Now move the axis to its desired position using the drive buttons.

i When you activate the option **Numeric Input** you can input the positions using the dialogue for [numerical input](#) without moving axes.

Finally tap the field **Number of parts** and use the dialogue for [numerical input](#) to input the number of pieces to be placed in the respective axis direction.

i With the option **Keep gap constant** the distance between stacking positions will be kept constant automatically, when changing the number of parts in an existing placing program.

When all parameters on this tab are entered, either open the next tab by tapping it, or

accept the placing program by pressing .

Press  to cancel the process and discard the entries.

3.3.2.1.4 Tab Options

This tab is used to set options about approach and placing order for the placing program.

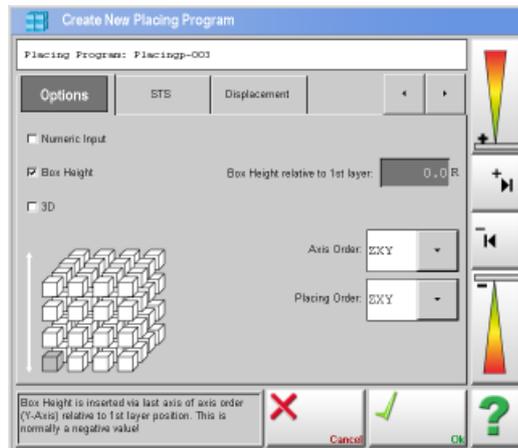


Fig.: Tab for approach and placing order

Axis Order

This sequence will be used during automatic operation to move the axes to the current stacking position.

Placing Order

This sequence will be used during automatic operation for going to the different stacking positions.

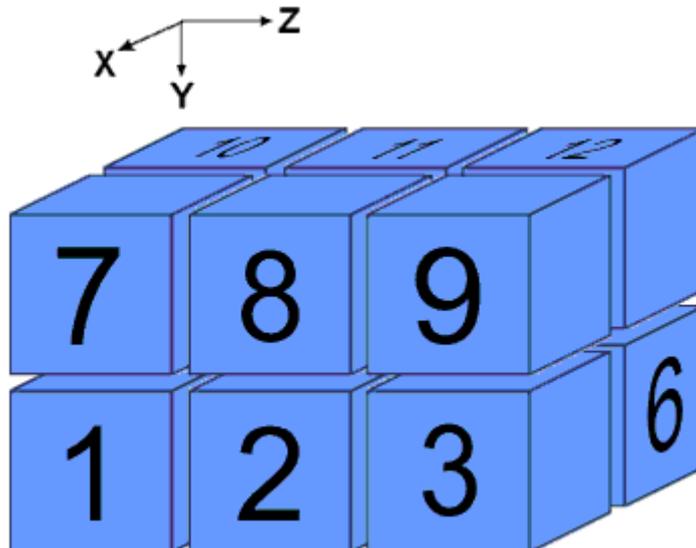


Fig.: Example - Placing Order ZXY

3D

A ✓ at this option activates the 3D approach. The axes will move to their positions simultaneously by means of a 3D motion. The parameters Box Height and Axis Sequence make the last axis in the axis sequence end the 3D motion displaced at the box height, and then travel into position alone. Otherwise, the robot would "move through" the wall of the crate. R8.3 does not execute the 3D movements as a straight line, but as a curve to box height. Use the option **straight line** to get a straight line.

Box Height

This value depends upon the dimensions of the crate. Select the axis to be moved into the crate. This has already been done in the Axis Sequence parameter; select an axis sequence in which this axis is the last to be moved (e.g., ZXY for the Y axis). Move the selected axis toward a position outside the crate that can be approached without collision (with a 3D motion). Box height will be used during single axis approach also, in order to trigger the error message 170-007 – BOX CRASH MONITORING. If you do not want to use box height, deactivate the option.

When all parameters on this tab are entered, either open the next tab by tapping it, or

accept the placing program by pressing .

Press  to cancel the process and discard the entries.

3.3.2.1.5 Tab STS - Stacking Sensor

On this tab you can activate that the last axis of the axis sequence uses a stacking sensor to approach the stacking positions. Above all this is helpful during un-stacking.

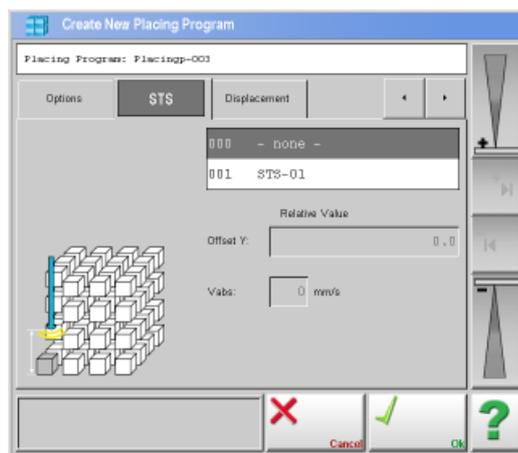


Fig.: Tab for a stacking sensor

With the selection field at the right top of the tab you can select which stacking sensor is to be used. With the selection "000 - none -" the stacking sensor function is deactivated, the axis will travel to the calculated position directly.

Offset

This is the distance from the calculated stacking position where the search movement

to the stack sensor shall start. After pressing , set the Offset by moving the respective axis using the drive buttons.

Vabs

This is the absolute speed used for the stacking sensor search movement. After tapping the input field, input the desired value using the dialogue for [numerical input](#).

When all parameters on this tab are entered, either open the next tab by tapping it, or

accept the placing program by pressing .

Press  to cancel the process and discard the entries.

 This tab will not be shown at a placing program for a double arm robot.

3.3.2.1.6 Tab Displacement

Certain pieces cannot be stacked one over the other (or side by side), but may have to be displaced in relation to the previous layer. Use the parameter Displacement of the placing program to achieve this for the respective axis direction.

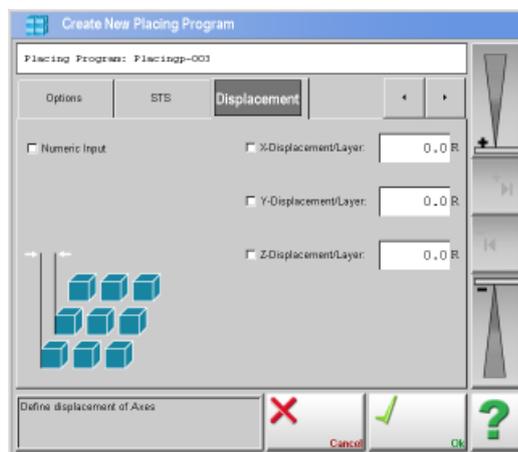


Fig.: Tab for displacement

With every layer change of the last axis in the placing order, the displacement value set for the respective axis will be added to the stacking positions of that axis.

Set the displacement value can be set after activating the respective axis by moving the axis with the drive buttons.

When all parameters on this tab are entered, either open the next tab by tapping it, or

accept the placing program by pressing .

Press  to cancel the process and discard the entries.

 This tab will not be shown at a placing program for a double arm robot.

3.3.2.2 Add Placing Program Commands to the Teach Program

Before you can add a placing program to the teach program, you have to [create](#) it in the menu **Part Prg Setup – Placing Program**.

When you have created a placing program in the current teach program already, and you want to add it as a command, press  to open the selection window for adding placing programs.

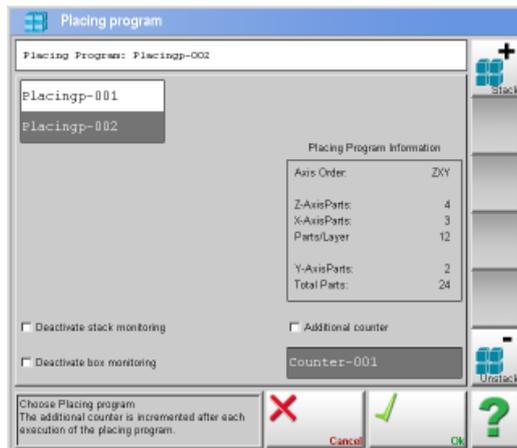


Fig.: Selection window for adding placing programs

Select the desired placing program from the list on the left top of the window. The table **Placing Program Information** shows how many positions in the different axes dimensions the selected placing program will travel to. If no placing programs have been defined in the current teach program yet, the message "No placing programs available" will be displayed instead of the selection list.

Use the buttons on the right side of the window to decide whether the placing program shall stack or un-stack:



creates a placing program command for stacking. The counters of the placing program will be incremented **after** traveling to a position. Pieces will be “added” to the stack.



creates a placing program command for un-stacking. The counters of the placing program will be decremented **before** traveling to a position. The placing program is running “in reverse” – pieces will be “removed” from the stack. In the program this will be indicated by adding the extension UNSTACK to the command.

Use the option **Additional Counter** and the selection box below to select a [counter](#) that will be incremented when stacking and decremented when un-stacking.

Press  to add the command to the teach program.

Press  to cancel the action.

3.3.3 Injection Molding Machine IMM

3.3.3.1 Injection Molding Machine IMM - Standard Functions



Here you can program signals to the IMM.

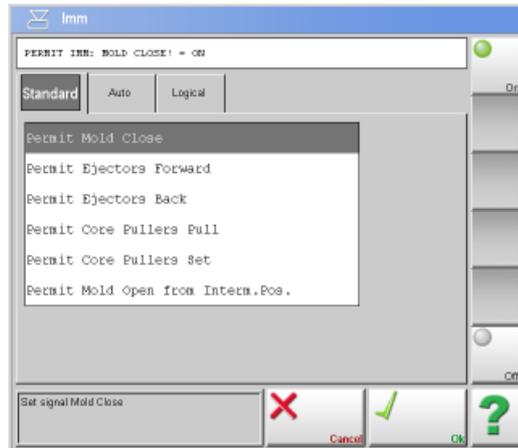


Fig.: Window for IMM Signals

First pick the desired signal from the selection list.



are used to select if the chosen signal is to be turned on or off.

When the robot is equipped with 2 interfaces for injection molding machines, one of the options **IMM 1** or **IMM 2** has to be tapped in order to select which Injection Molding Machine is the targeted by the signal.



is used to add the command to the Teach program.



is used to cancel the entry.



If a signal has been enabled in the teach program, it also has to be disabled again.



These signals are only permits (inhibits) to the injection molding machine. The actual control of the movements is done by the IMM.

3.3.3.1.1 MOLD CLOSE and FULLY MOLD OPEN



Use the permission signals MOLD CLOSE and FULLY MOLD OPEN to program the permits for the movements of the mold of the injection molding machine. The signal FULLY MOLD OPEN is only available with the extended E67 / E12 interface; here the mold is automatically only opened to an intermediate position. The robot does not control the standard opening of the mold - the IMM opens the mold when the part is finished.

Furthermore the IMM utilizes E67 / E12 signals to send feedback about the position of the mold. This is done using the signals MOLD OPEN and MOLD CLOSED. Only when the mold is fully open, it is possible for the robot to enter the mold.

The extended E67 / E12 interface additionally supplies the feedback signal MOLD INTERMEDIATE POSITION. Here the robot can already enter the mold, when it is opened to the intermediate position. According to the mode used at the IMM the robot then may have to permit the mold to fully open.

Example for standard E67 / E12

WAIT IMM: MOLD OPEN = ON	wait until the mold is fully open
AND IMM: MOLD WAS CLOSED = ON	and was fully closed
PERMIT IMM: MOLD CLOSE! = OFF	switch off permission
enter the mold area, remove the part, exit the mold area, then ...	
PERMIT IMM: MOLD CLOSE! = ON	close the mold.
Part placement ...	

Example for extended E67 / E12

WAIT IMM: MOLD INTERMEDIATE POS = ON	wait until the mold is open
AND IMM: MOLD WAS CLOSED = ON	and was fully closed
PERMIT IMM: MOLD CLOSE! = OFF	switch off permission
enter the mold area; fix the part on the EOAT	
PERMIT IMM: FULLY MOLD OPEN = ON	fully open the mold
WAIT IMM: MOLD OPEN = ON	wait until the mold is fully open
PERMIT IMM: FULLY MOLD OPEN = OFF	switch off permission
exit the mold area	
PERMIT IMM: MOLD CLOSE! = ON	close the mold.
Part placement ...	

3.3.3.1.2 EJECTORS FORWARD/BACK and COREPULLERS PULL/SET



Using the commands EJECTORS FORWARD/BACK and COREPULLERS PULL/SET the movements of the ejectors and core pullers of the IMM are taught. The most basic way to do so, is to switch on the permit for the desired movement, then wait until the target position is reached using a [WAIT](#) command, and then to switch off the permit again.

The following example moves the ejectors forward:

```
PERMIT IMM: EJECTORS FORWARD! = ON      switch on permission
WAIT IMM: EJECTORS FORWARD = ON         wait ...
PERMIT IMM: EJECTORS FORWARD! = OFF     switch off permission
```

3.3.3.2 Injection Molding Machine IMM - Automatic Functions



The automatic functions simplify programming the movements of the IMM.

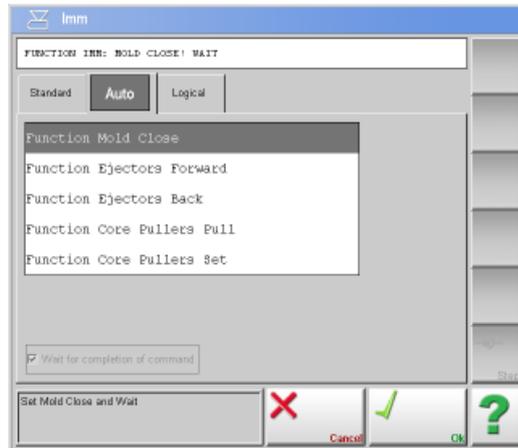


Fig.: Window for IMM signals

First pick the desired movement from the selection list.

The permits are toggled correctly by the commands automatically. When the option **Wait for completion of the command** is activated, the command will wait until the movement is finished. In that case command will have the extension `WAIT` added in the Teach program.

When the robot is equipped with 2 interfaces for injection molding machines, one of the options **IMM 1** or **IMM 2** has to be tapped in order to select which Injection Molding Machine is the targeted by the signal.



is used to add the command to the Teach program.



is used to cancel the entry.



These signals are only permits (inhibits) to the injection molding machine. The actual control of the movements is done by the IMM.

3.3.3.2.1 MOLD CLOSE! WAIT



The command

```
FUNCTION IMM: MOLD CLOSE! WAIT
```

eases programming the closing movement of the mold. The command equals the following sequence of "standard" commands:

```
PERMIT IMM: MOLD CLOSE! = ON
WAIT IMM: MOLD CLOSED = ON
AND IMM: MOLD OPEN = OFF
```

3.3.3.2.2 EJECTORS FORWARD/BACK (WAIT) and CORE PULLERS PULL/SET (WAIT)



The automatic functions will ease the programming of the movements of the ejectors and core pullers essentially. The respective permits are simply toggled by the commands automatically.

The automatic function for "ejector back" will look like this:

```
FUNCTION IMM: EJECTOR BACK!
```

This equals the following sequence of "standard" commands:

```
PERMIT IMM: EJECTOR FORWARD! = OFF
PERMIT IMM: EJECTOR BACK! = ON
```

If additionally the option **Wait for completion of command** is activated, the extension **WAIT** is added to the command, and the command will wait for the respective movement to be finished.

The Wait function for "ejectors back" will look like this:

```
FUNCTION IMM: EJECTOR BACK! WAIT
```

This equals the following sequence of "standard" commands:

```
PERMIT IMM: EJECTOR FORWARD! = OFF
PERMIT IMM: EJECTOR BACK! = ON
WAIT IMM EJECTOR BACK = ON
AND IMM EJECTOR FORWARD = OFF
```

3.3.3.3 Injection Molding Machine IMM - Logical Functions – Mold Was Closed



Here you can program the command for manipulating the latch **Mold Was Closed** for the [condition](#) with the same name.

This will for example be necessary when at the first cycle the condition **Mold Was Closed** has to be overridden for entering the mold, because the mold is going to be closed for the first time only after inserting inserts.

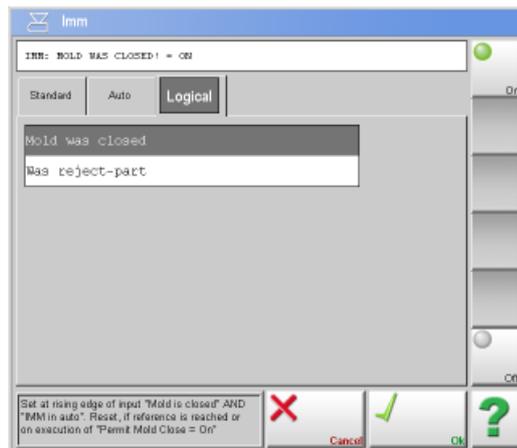


Fig.: Window for logical IMM commands



and are used to select if the latch shall be set or reset.

When the robot is equipped with 2 interfaces for injection molding machines, one of the options **IMM 1** or **IMM 2** has to be activated, in order to select the Injection Molding Machine, whose latch will be targeted by the action.



is used to add the command to the Teach program.



is used to cancel the entry.

3.3.4 Smart Removal

3.3.4.1 Smart Removal: In



The command `Smart Removal: IN` is used to program a time optimized movement into the mold of the IMM in a very simple way. For this purpose the command starts a curved movement of Y and X during the mold opening movement already, in order to travel off the Y mold safety switch shortly after full mold opening has been achieved. Waiting for the mold signals is processed automatically. A distance for X moving in straight can be set. The integrated "parallel at time" allows pre-triggering of the ejectors.

Use the first window to determine, which axes will be moved in what order by the Move-In command.

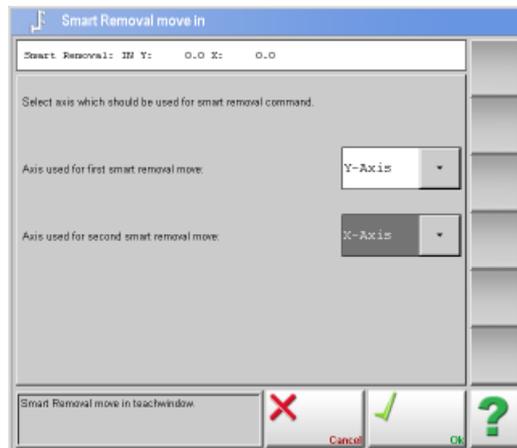


Fig.: Axes selection for Smart Removal In



Use  to confirm the selection and to open the dialogue for inputting the other parameters.

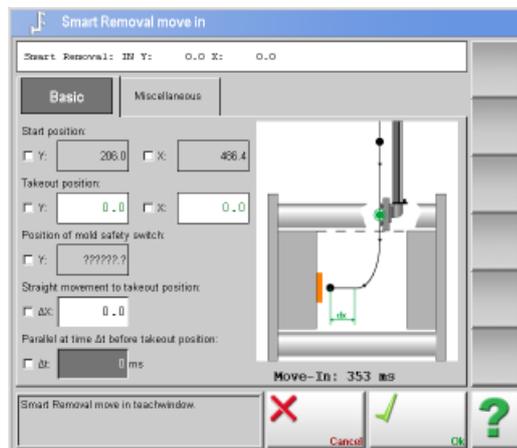
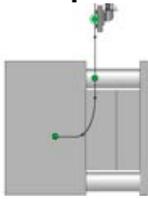


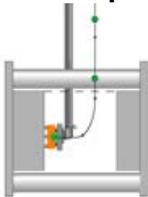
Fig.: Dialogue for Smart Removal In

Start position



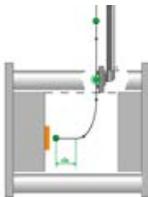
This is the position where Smart Removal starts moving in. Is retrieved automatically from the teach commands before `Smart Removal: IN`.

Takeout position



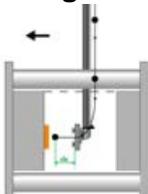
This is the target position of the Smart Removal movement into the mold. At this position the gripper of the robot should take over the parts from the IMM. After activation with a ✓, the participating axes can be moved to their desired positions one by one using the drive buttons.

Position of the S5 mold safety switch



This is the position where the Y axis travels off the cam of its mold safety switch, thus entering the mold. It is determined automatically.

Straight movement to takeout position from dx

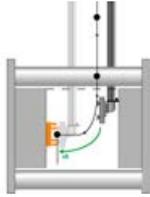


This is the distance before the X takeout position, where the Y movement must be finished, in order for X to travel to the takeout position with a straight movement. Input

the desired distance after tapping the input field or after pressing  using the dialogue for [numerical input](#). Furthermore, after tapping the input field you can use

 and  to tune the distance.

Parallel at time dt before takeout-position



This is the time before the takeout position is reached with Y and X, when the command `Smart Removal: IN` acknowledges itself, thus starting the Teach commands after itself, before it is actually finished. This way you can pre-trigger the ejectors, for example. The resulting parallel movement has to be completed with a [WAIT SYNC](#) command at an adequate position in the Teach program. Input the

desired time after tapping the input field or after pressing  using the dialogue for [numerical input](#). Furthermore, after tapping the input field you can use  and  to tune the time.

Mold Intermediate Position

If `Smart Removal: IN` shall enter the mold already when mold opening reaches the **mold intermediate position**, access the tab  and activate the option **Use mold Intermediate position** there. This option is only available when the optional signal Mold Intermediate Position is available on the machine interface.

Press  to add the command to the Teach program.

Press  to discard the entries.

Example:

```

...
Travel to the start position of Smart Removal over the mold.
*Y: 0.0
*X: 123.4
*Z: 4711.0

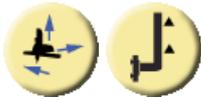
Smart Removal: IN

Pre-triggered vacuum and ejectors using the time dt.
Vacuum-01 = ON
FUNCTION IMM: EJECTORS FORWARD! WAIT
WAIT SYNC

WAIT Vacuum-01 = ON
...

```

3.3.4.2 Smart Removal: Out



The command `Smart Removal: OUT` is used to program a time optimized movement out of the mold of the IMM in a very simple way. For this purpose the command triggers the closing of the mold already while it moves X and Y out of the mold with a curved movement. A distance for a straight movement of X can be set. An adjustable time allows permitting the closing movement of the mold before the Y axis travels onto the cam of its mold safety switch, thus eliminating any switching delays of the machine interface.

Use the first window to determine, which axes will be moved in what order by the Move-Out command.

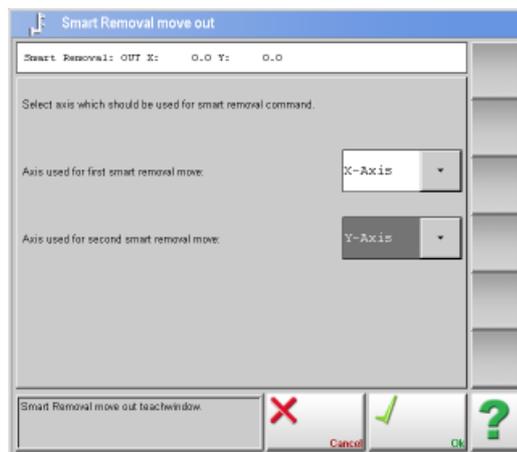


Fig.: Axes selection for Smart Removal Out

Use  to confirm the selection and to open the dialogue for inputting the other parameters.

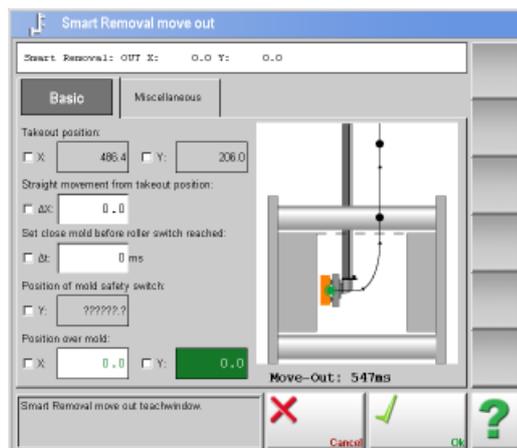
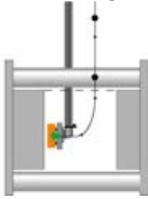
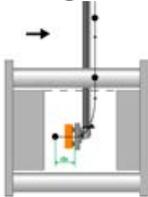


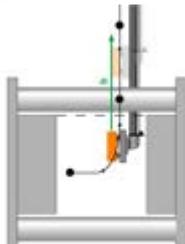
Fig.: Dialogue for Smart Removal Out

Takeoutposition

This is the position, where Smart Removal starts moving out. Is retrieved automatically from the teach commands before `Smart Removal: OUT`.

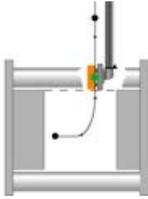
Straight movement form takeout position to dx

This is the distance from the X takeout position, where the Y movement is started, after X has moved away from its takeout position using a straight motion. Input the desired distance after tapping the input field or after pressing  using the dialogue for [numerical input](#). Furthermore, after tapping the input field you can use  and  to tune the distance.

Set close mold before S5 is reached dt

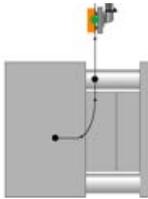
This is the time before the S5 cam is reached, when the command `Smart Removal: OUT` already turns on the permit to close the mold, in order to eliminate any switching delays of the machine interface. Input the desired time after tapping the input field or after pressing  using the dialogue for [numerical input](#). Furthermore, after tapping the input field you can use  and  to tune the time.

Position of the S5 mold safety switch



This is the position, where the Y axis reaches the cam of the mold safety switch, i.e. where it exits the mold. It is determined automatically.

Position over mold



This is the target position of the Smart Removal movement out of the mold. From this position it should be possible for the robot to move off the machine. After activation with a ✓, the participating axes can be moved to their desired positions one by one using the drive buttons.

Press  to add the command to the Teach program.

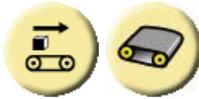
Press  to discard the entries.

Example:

```
...  
After the parts have been delivered to the robot gripper using the ejectors.  
FUNCTION IMM: EJECTORS BACK! WAIT  
  
Smart Removal: OUT  
  
Travel to the placement position ...  
Z: 123.4  
...
```

3.3.5 Peripherals

3.3.5.1 Conveyor



This command is used to control a conveyor.

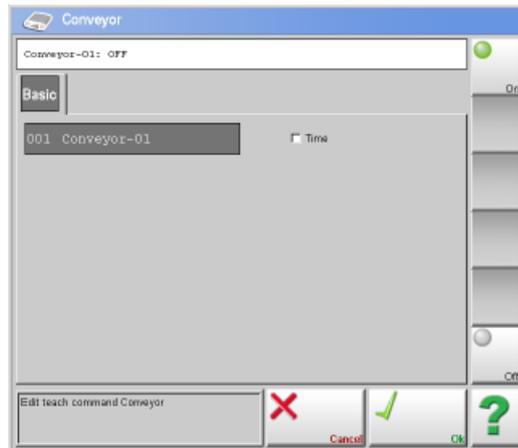


Fig.: window for conveyors



are used to select if the command shall switch the conveyor on or off.

When the option **Time** is activated with a ✓, the selected state is only kept for the selected time, without interrupting the teach program. The value for the time is input using the [numerical input](#) dialogue after tapping the input field.



is used to add the command to the teach program.

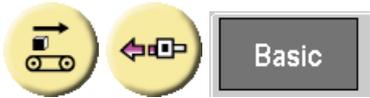


is used to cancel the input.

See also:

3.2.6.7 Setup Menu: Conveyor Names

3.3.5.2 Peripheral Outputs PO



This command is used to control peripheral outputs (POs).

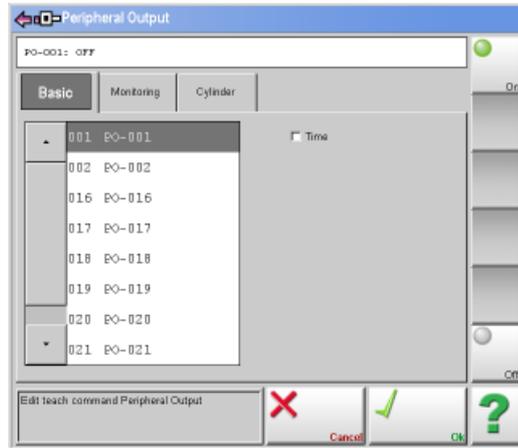


Fig.: window for peripheral outputs

First, the desired peripheral output is selected.



are used to select whether the command shall switch the selected peripheral output on or off.

When the option **Time** is activated with a ✓, the selected state is only kept for the selected time, without interrupting the teach program. The value for the time is input using the [numerical input](#) dialogue after tapping the input field.



is used to add the command to the teach program.



is used to cancel the input.



[Part monitoring](#) can be activated for peripheral outputs. In monitoring, outputs and inputs having the same number will always correspond to one another. As an example, monitoring for peripheral output 1 will be performed through peripheral input 1.



Peripheral outputs will remain set when shifting to manual, block stop or in the event of an error.

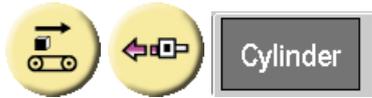
See also:

3.3.5.7 Monitoring

3.3.5.3 Cylinders

3.2.6.8 Setup Menu: PO-Names

3.3.5.3 Cylinder function



This function is useful to ease the teaching of pneumatic cylinders of some peripheral equipment, by combining two peripheral outputs to a cylinder. Additionally you can use two peripheral inputs and a time limit to monitor the correct function of the pneumatic cylinder. If the cylinder does not reach the respective target position within the set time limit, an error message will be displayed on the teachbox.

Up to 255 cylinders can be defined. Mapping of the peripheral outputs and inputs, also setting the time limit is done using the [cylinder definition](#).

After selecting the function, first the desired cylinder is selected from the list.



are used to select if the command has to move the cylinder to either one of its final positions or to release it. Release means, that both outputs are switched off.

In case of cylinders that are equipped with position limit switches, in automatic operation the next command is not executed before the desired target position of the gripper is approached. The option **Parallel** enables to move several cylinders simultaneously in that case.



This allows starting additional functions during the motion of the cylinder already.

In order to prevent uncontrolled motions, the instruction [WAIT SYNC](#) has to be entered after every parallel motion. This will ensure that all motions have been completed before the next instruction is started.



is used to add the command to the teach program.



is used to cancel the input.



Peripheral outputs will remain set when shifting to manual, block stop or in the event of an error.

See also:

3.2.6.4 Setup Menu: Cylinder Definition

3.3.5.3.1 Applying cylinders in a teach program

Without using the cylinder function, moving a pneumatic cylinder forward and backward will be programmed somehow like this:

```
PO-002: OFF          switch off the valve for "backward"  
PO-001: ON          switch on the valve for "forward"  
WAIT PI-001 =ON     wait until the limit switch "cylinder forward" is actuated
```

...

```
PO-001: OFF        switch off the valve for "forward"  
PO-002: ON        switch on the valve for "backward"  
WAIT PI-002 =ON   wait until the limit switch "cylinder backward" is actuated
```

Using the cylinder function simplifies the example like this:

```
CYL-01: ON         switch off the valve for "backward"  
                  switch on the valve for "forward"  
                  wait until the limit switch "cylinder forward" is actuated
```

...

```
CYL-01: OFF       switch off the valve for "forward"  
                  switch on the valve for "backward"  
                  wait until the limit switch "cylinder backward" is actuated
```

If also the time limit is set correctly at the cylinder definition, one benefits a monitoring function with error message for time limit exceeded without additional programming.

3.3.5.4 Vacuums



This command is used to control vacuum circuits.

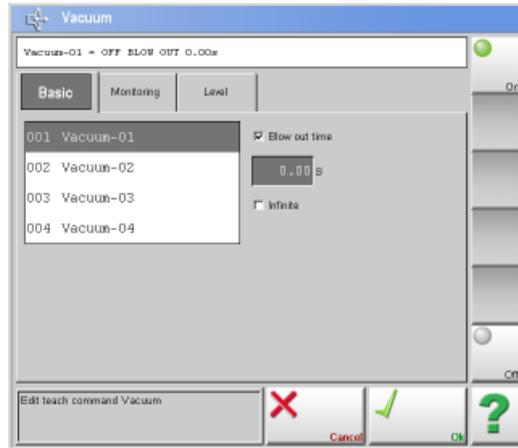


Fig.: window for vacuum circuits

First, the desired vacuum circuit is selected.



On and Off are used to select whether the command shall switch the selected vacuum circuit on or off.

When the robot is equipped with the corresponding valves, a blow-off function will be available at the turn off command. You can activate the blow off function with a ✓ at **Blow out time**. After tapping the input field, input the duration of blowing using the dialogue for [numerical input](#). With the option **Infinite** the command will switch on blowing permanently.



Ok is used to add the command to the teach program.



Cancel is used to cancel the input.

See also:

- 3.3.5.5 Vacuum Level
- 3.3.5.7 Monitoring
- 3.2.6.12 Setup Menu: Vacuum Names
- 3.2.6.14 Setup Menu: Vacuum Reference States

3.3.5.5 Vacuum Level



Use the Vacuum Level command to set the vacuum threshold for [vacuum monitoring](#) and the [condition vacuum](#) independent from the preset Standard Vacuum Pressure. Thus you can use different thresholds in different parts of the Teach program. The threshold for a specific vacuum circuit set with the Vacuum Level command will be valid until the next Vacuum Level command for the same vacuum circuit.



Fig.: Window for the Vacuum Level command

After selecting the desired vacuum circuit and tapping the input field **Set vacuum level** you can input the desired threshold using the dialogue for [numerical input](#). Alternatively you can use the slider on the right side of the window.

Select the option **Standard** with a to generate the command for using the threshold preset with [Setup Standard Vacuum Pressure](#).

Use  to add the command to the teach program.

Use  to discard the entries.

- See also:
- 3.3.5.4 Vacuums
 - 3.3.5.7 Monitoring
 - 3.3.7.8.5 Condition: Vacuum
 - 3.2.6.16 Setup Standard Vacuum Pressure

3.3.5.6 Grippers



This command is used to control gripper circuits.

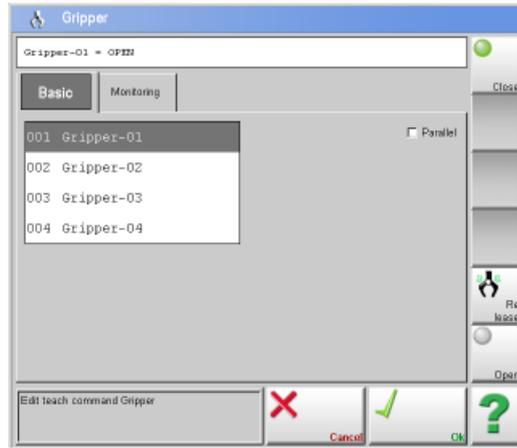


Fig.: window for gripper circuits

First, the desired gripper circuit is selected.



are used to select whether the command shall switch the selected gripper circuit to opened, closed or released.



The state **RELEASED** is only available at gripper circuits equipped with the required valves.

In case of gripper circuits that are equipped with position feedback, in automatic operation the next command is not executed before the desired target position of the gripper is approached. The option **Parallel** enables to move several grippers simultaneously in that case.



This allows starting additional functions during the closing or opening motion.

In order to prevent uncontrolled motions, the instruction [WAIT SYNC](#) has to be entered after every parallel motion. This will ensure that all motions have been completed before the next instruction is started.



is used to add the command to the teach program.



is used to cancel the input.

See also:

3.3.5.7 Monitoring

3.2.6.13 Setup Menu: Gripper Names

3.2.6.15 Setup Menu: Gripper Reference States

3.3.5.7 Monitoring of Vacuum Circuits, Grippers and Peripheral Outputs

Using (part-) monitoring the robot can react on the loss of parts, in order to avoid damage caused by parts that got stuck, were not taken out or have fallen down.

The monitorings are background functions that from their activation until their deactivation check, if all parts are still present on the respective vacuum circuit, gripper or peripheral circuit.

At **vacuum circuits** part monitoring is done using the analogue vacuum pressure monitoring. Use the menu [Setup Vacuum Pressure](#) to set the trigger value for the current Teachprogram.

At **grippers** the state of the respective gripper input *Part Monitoring* is monitored. This input must be actuated by a suitable sensor.

At **peripheral outputs** the state of the peripheral input with the same number is monitored.

When a monitoring is triggered (by part loss), the action (Stop or Alternative Routine) selected on activation of the monitoring will be carried out.

Action STOP

When the monitoring is triggered, the robot will switch to operating mode Manual. This results in an immediate stop of the axes. The error message *Element: PART LOST* will be emitted.

Action ALT: *label*

When the monitoring is triggered, the robot will remain in operating mode Automatic. Nevertheless an immediate stop of the axes will be carried out. The execution of the Teachprogram will continue at the designated [label](#) (in the alternative sequence). The error message *Element: PART LOST* will be emitted.



Alternative sequences have to be made dependant on the current position, in order to avoid collisions.

It is possible to assign a separate alternative sequence to each of the vacuum, gripper and peripheral circuits; however, it is likewise possible to use a single alternative sequence for all circuits.



Alternative sequences have to be ended with an **unconditional jump** and the **IMM signals** have to be switched according to the relevant needs. In alternative sequences, the monitoring system of the addressed circuit or the circuit itself should be turned off; otherwise the program will jump back to this alternative sequence after having completed the alternative sequence.



In order to avoid collisions during or after an alternative sequence, the Y-axis should possibly be moved to its initial position before the other axes are being moved.



If an alternative sequence is used by several monitorings, all monitorings must be disabled at the beginning of the alternative sequence! Otherwise multiple call-ups of the alternative sequence may occur, and instructions at the beginning of the alternative sequence will be processed more than once!

After activating the tab  in the command window of the *element* (vacuum circuit, gripper or peripheral output) to be monitored, you can select the desired monitoring command:

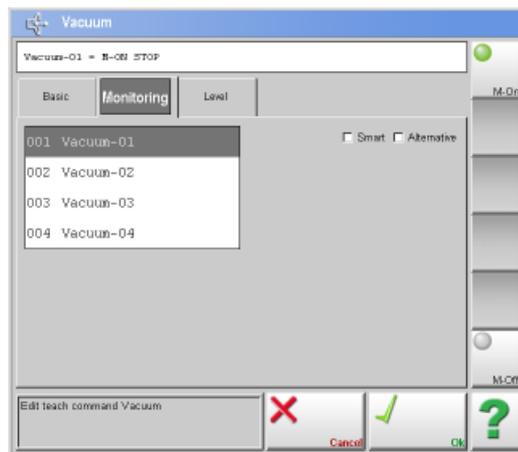


Fig.: Window for the vacuum monitoring

Use the list on the left side of the window to select the *element* to be monitored.

Deactivating a monitoring

Deactivation is done **automatically** when switching off / opening the respective *element* or with the monitoring OFF command.

Press  to select the command for deactivation of a monitoring:

Element = M-OFF

Activating a monitoring

Press  to select the command for activation of a monitoring.

Without further options the command to activate a monitoring with the **action STOP** will be created:

Element = M-ON STOP

Using the option **Alternative** the command to activate a monitoring with the **action ALT** will be created:

Element = M-ON ALT: *label*

You can either input a new label or select an existing label.

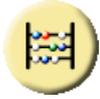
Using the option **Smart** (only with vacuum) the monitoring will be activated delayed to let it become active in the moment before the permit to close the mold is being switched on. This is especially helpful in connection with the `Smart Removal: OUT` command, in order to avoid false triggering during acceleration of the exiting (Y-) axis, while making sure that the mold will not close in case of a part lost inside it.

Example:

```
...
Smart Removal: IN Y: 678.9 X: 234.5
Vacuum-01 = ON
FUNCTION IMM: EJECTORS FORWARD! WAIT
WAIT Vacuum-01 = ON
FUNCTION IMM: EJECTORS BACK! WAIT
Vacuum-01 = M-Smart ON STOP
Smart Removal: OUT Y: 0.0 X: 123.4
...
```

3.3.6 Counters

3.3.6.1 64 freely usable Counters



The primary use of counters is to program repeat functions or to carry out an event count. Furthermore counters can be used like variables to save e.g. states and settings. Counters are also useful to interchange parameters and to do handshakes between part programs. Incorporating Stopwatches makes timer functions possible.

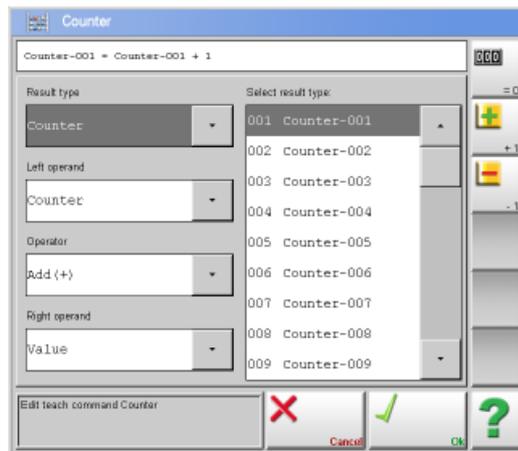


Fig: Window for counters

64 freely usable counters are available.

After entering the counter menu or after tapping the field **Result type** use the field **Select result type** to select the counter to be modified by the operation.

Use the buttons ,  and  to select one of three common cases:

- Counter-xxx = 0
- Counter-xxx = Counter-xxx + 1
- Counter-xxx = Counter-xxx - 1

This selection also defines the basic format of the command (1 or 2 operands).

Use the field **Operator** to select one of the operators **+** (Addition), **-** (Subtraction), ***** (Multiplication), **DIV** (Division) or **REM** (Division Remainder).

After tapping one of the fields **Left Operand** or **Right Operand** you can choose the values to be placed right and left of the operator. Possible are:

Counter ... the value of a counter

Row Counters X/Y/Z ... the value of one of the row counters of a placing program

Part Counter ... the value of the part counter of a placing program

Value ... a number ranging from -2147483648 to 2147483647

Stopwatch ... the result of a time measurement of a Stopwatch in 1/100s of seconds

According to the type of the chosen operator, the suitable field for completion of the entry will be shown on the right side of the window.

For the operand **Stopwatch** additionally use the buttons , , ,  to select the desired value of the stopwatch.

Use  to add the command to the teach program.

Use  to close the window without changing the teach program.

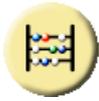
Example:

```
Counter-001 = 0
-
START:
...
Removal process
...
Counter-001 = Counter-001 + 1
JMP QUALI Counter-001 = 50
...
Normal placing
...
JMP START

QUALI:
...
Placing for Quality Control
...
Counter-001 = 0
JMP START
```

Every 50th piece will be placed on a special position for quality control.

3.3.6.2 Reset / Manipulate Placing Counters



The **placing counters** are required in the placing program in order to memorize how many pieces have been already placed for each axis and to approach the next placing position. Once the placing program is finished (the last piece has been placed), the associated placing counters for the axes X, Y and Z are reset to zero prior to a new start of the placing program.

The reading of the number of pieces previously deposited by the respective placing program is in the **part counter**. After the placing program has been completed (last piece has been placed), the value of the part counter remains unchanged until the first part of the sequence has been placed again.

Placing counters and part counters always change together.

By manipulating the part counter it is possible to let the placing program continue on any position of the placing sequence.

The teach program function `Placingp-XXX RESET` can be used to clear the placing counters and part counter of the placing program XXX at any time.

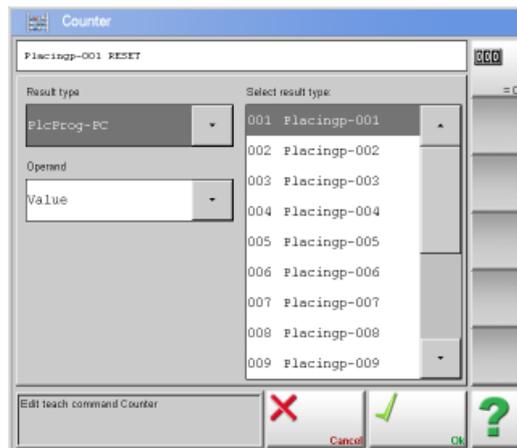


Fig.: Window for placing counters

After opening the counter menu select `PlcProgram-PC` as the **Result type**. Now you can select the placing program to be manipulated by the command in the field **Select result counter**.

The command `Placingp-XXX RESET` will be offered by default. Tap the field **Operand** if you want to select another option for manipulating the part counter.

Press  to add the command to the teach program.

Press  to close the window without changing the teach program.

Example:

```
SUBR NEW_BOX  
Conveyor-01: ON 5.0 s  
Placingp-001 RESET  
RET
```

The [subroutine](#) NEW_BOX executes a box change, when called. The placing program is being reset, and will continue with the first part in the new box.

3.3.7 Program branching

3.3.7.1 Jump



Basically a teach program is executed command after command, from top to bottom, starting at program line no. 0001. If the robot has to be able to react on certain events, a change in this sequence of execution will be required.

Using the JMP (Jump-) command it is possible not to continue with the next program line, but to jump to a different location somewhere in the program, and to continue the execution of the program from there on. The JMP command requires a [Label](#) as a target to jump to.

You can either type in the name of a new label after tapping the input field **Enter label**, or choose an existing label from the selection field **Select label**.

If the jump command is not combined with a condition, the jump to the label will be executed with every execution of the command. This unconditional jump is referred to as absolute jump.



is used to generate such an absolute jump.

Example:

JMP START



is used to generate a conditional jump, that is only executed when the selected [jump condition](#) is true.

Example:

JMP QUALI CNT01 = 15

The label QUALI is only jumped to, if counter CNT01 equals 15.



is used to confirm all parameters and to add the command to the teach program.

3.3.7.2 Label

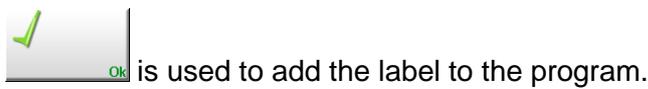


A label is the target for [Jump](#) commands.

Tap the input field **Input label** to enter the alphanumeric name of the label using the [Text input](#) dialogue. The name can be up to 12 characters long.

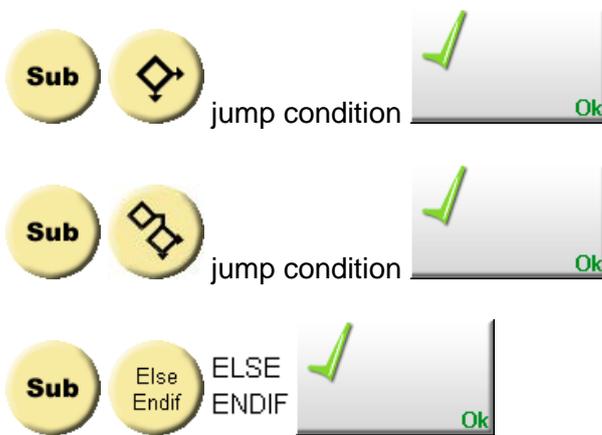
Example:

BOX_01 :



- A label with a certain name may only exist once in a teach program, but it can be targeted by any number of Jump commands.

3.3.7.3 If - Elseif - Else - Endif



Basically a teach program is executed command after command, from top to bottom, starting at program line no. 0001. If the robot has to be able to react on certain events, a change in this sequence of execution will be required.

Using the IF - ELSEIF - ELSE - ENDIF commands it is possible to execute or not execute program parts if [jump conditions](#) are true or not true.

- You must not exit the program part between IF and ENDIF using a jump command.

There are 3 possible variants.

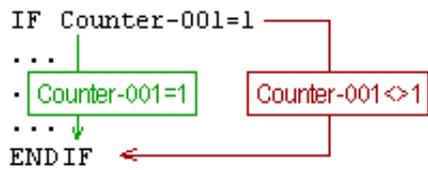
3.3.7.3.1 Variant 1:

The instruction IF defines a condition for the execution of an alternative program sequence. This sequence is programmed in the program lines after the instruction IF. The END IF instruction is programmed at the end of the alternative sequence.

If the jump condition is false, the instructions between IF and END IF will not be carried out. The program routine will be continued with the first instruction after END IF.

If the jump condition is true, first the instructions between IF and END IF will be carried out. Only then will the program routine be continued after END IF.

Example:



3.3.7.3.2 Variant 2:

The instruction IF is used to specify a condition for the execution of two alternative program sequences.

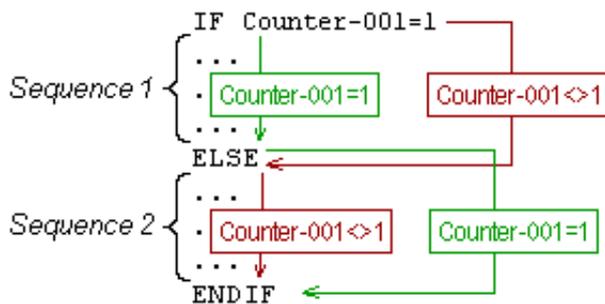
Sequence 1 is to be executed only when the condition is true. Sequence 2 is to be executed only when the condition is false.

Sequence 1 is programmed in the program lines after the instruction IF. Then the instruction ELSE and, following that, sequence 2 is programmed. At the end of sequence 2, the instruction END IF is programmed.

If the jump condition is false, the instructions between IF and ELSE will not be carried out, and the program routine will be continued with the first instruction after ELSE (= sequence 2).

If the jump condition is true, first the instructions between IF and ELSE will be carried out (= sequence 1), and then the program routine after END IF will be continued.

Example:



3.3.7.3.3 Variant 3:

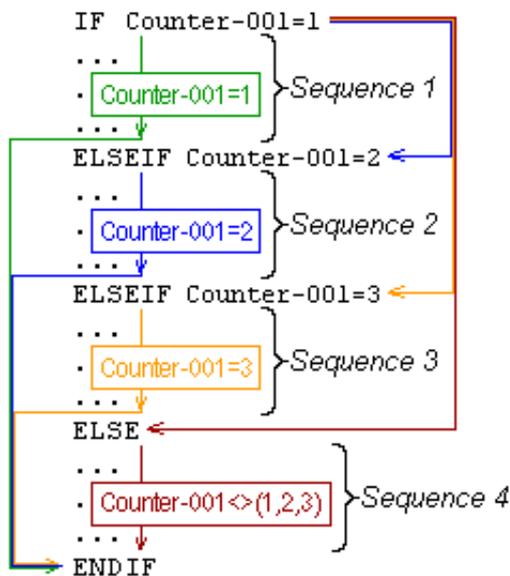
The commands IF and ELSEIF are used to define conditions for the execution of various program routines; where only one of them has to be executed at a time.

In automatic mode the conditions of the IF command and of the ELSEIF commands are checked one after the other, following their sequence in the program. The program routine attached to the **first true** condition will be executed, after that the execution of the program will continue behind the ENDIF.

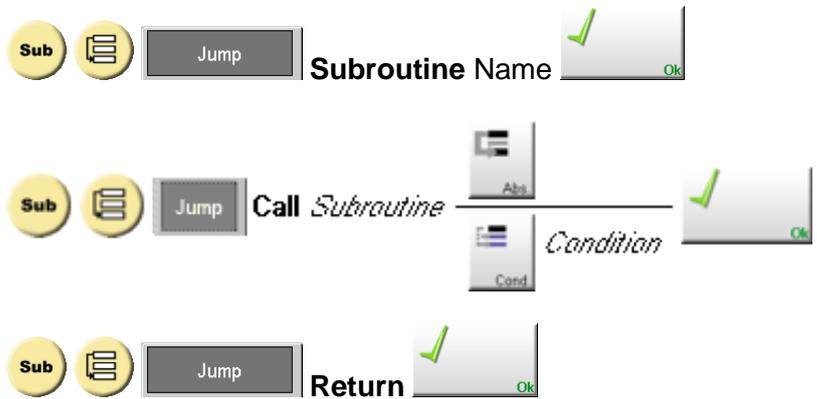
The optional ELSE command is used to define a routine, which will only be executed if **none** of the conditions of the IF and ELSEIF commands had been true.

This variant may especially be helpful creating sequencers, where in dependence of the value of a counter a different step of a sequence has to be executed.

Example:



3.3.7.4 Call - Subr – Ret



Basically a teach program is executed command after command, from top to bottom, starting at program line no. 0001. If the robot has to be able to react on certain events, a change in this sequence of execution will be required.

Using CALL-SUBR-RET commands it is possible to structure a Teach program with subroutines, and to call these subroutines conditional (based on the [jump conditions](#)) or unconditional.

Subroutine callups permit branching into a sequence of instructions. After their execution, the process will jump back to an instruction located directly after the related callup. In this way, program sections that are needed more often have to be programmed only once.



When programming subroutines, the following rules have to be kept in mind in order to ensure proper execution:

- Subroutines must always be terminated with the instruction RET.
- A subroutine must not be exited with the instruction JUMP.
- A subroutine routine must never be processed without callup with the instruction CALL.

Instruction CALL

The instruction CALL Name is used to call up the subroutine Name. Execution of the program will be continued at the subroutine.

Instruction SUBROUTINE

The instruction SUBR Name is used to define the start of a subroutine.

Instruction RETURN

RET is used to terminate a subroutine and to return to the main program. Return is made to the line after the CALL instruction by which the subroutine was called up.

Example

```
CALL INIT
START:
CALL REMOVAL
CALL PLACE
JMP START
```

```
SUBR INIT
...
RET
```

```
SUBR REMOVAL
...
RET
```

```
SUBR PLACE
...
RET
```

This example shows the possibility for structured programming. The first 5 lines form the main routine, that simply calls the subroutines one after the other, using a sequence of CALL-commands. The subroutine INIT is only called once when the program is started, during the further cycles only the subroutines REMOVAL and PLACE are executed. By using conditional CALL commands, routines for events like reject part or quality control can be added to the program easily. The advantage of this programming method is the short main routine, which gives a quick overview of the general sequence and the functions of the program. The details of the functions are programmed in the subroutines.



Subroutines can call other subroutines, i.e. they can be nested. Maximum nesting level is 160.

3.3.7.5 And



The AND and OR functions allow to execute IF, JMP, CALL and WAIT commands on the basis of several [jump conditions](#).

First, program the IF or JMP command with the first jump condition. Then, using the AND/OR function, program the other jump conditions in the following program lines.

Example:

```
JMP Label Counter-001 = 1  
AND Counter-002 = 1
```

The Jump is only executed when both counters Counter-001 and Counter-002 have the value 1.



AND and OR instructions must be programmed in the line(s) directly after the instruction to which they refer. Otherwise logical operation will not take place.

3.3.7.6 Or



The AND and OR functions allow to execute IF, JMP, CALL and WAIT commands on the basis of several [jump conditions](#).

First, program the IF or JMP command with the first jump condition. Then, using the AND/OR function, program the other jump conditions in the following program lines.

Example:

```
JMP Label Counter-001 = 1  
OR Counter-002 = 1
```

The Jump is executed when at least one of the counters Counter-001 or Counter-002 has the value 1.



AND and OR instructions must be programmed in the line(s) directly after the instruction to which they refer. Otherwise logical operation will not take place.

3.3.7.7 End



Every program should be terminated with the instruction END, in order to avoid unintended processing of additional subroutines.

The robot program and the all modes program will continue their automatic processing at program line 0001 after processing the END command.

Reference programs, the peripheral program, the additional part programs and also the EOAT Change program will terminate at the END command.

3.3.7.8 Jump Conditions

Using jump conditions the commands JMP, IF, CALL, WAIT, OR and AND can make the robot react on internal and external events.

After calling up the above mentioned commands, the following tabs are available to access the desired event:

- Counter
- Placing Program
- PI
- IMM
- Vacuum
- AutoSwitch
- Misc part program, area, position, mode, stacking sensor, start, production end active
- Input
- Output

3.3.7.8.1 Jump condition: Counter

The branch is executed dependant on the result of a comparison. The operands for the comparison can be counters, placing counters, part counters and constant values.

After selecting the event, the desired operands for the left and the right side of the comparison are picked from the respective lists.

If counters are chosen, a window to select one of the 64 counters is shown.

If placing counters or part counters are chosen, a window with the available placing programs is shown.

If value is chosen as an operand, it can be input after tapping the **Input value** field, using the dialogue for [numerical input](#).

Additionally, one of the operators <=, >=, <, >, <>, = must be chosen. This is to define how the numerical value of the left operand must relate to the numerical value of the right operand in order to trigger the branch.

Example:

```
IF PLCPRG01-PC = 12
CONVEYOR_01 ON 5.0 S
ENDIF
...
```

The conveyor is switched on if 12 parts have been placed by the placing program.

3.3.7.8.2 Jump Condition: Placing Program

The branch will be triggered by events of placing programs.

After selecting the condition, the desired placing program and one out of three events can be chosen.

Stack ready

The stack is completed. All row counters are zero, the next piece will be the first of the placing sequence again. This can be used to index the conveyor.

Layer ready

A layer is completed. The row counters of the first two axes of the placing sequence are zero, the next piece will be the first of the next layer. This can be used to trigger intermediate layers.

Row ready

A row is completed. The row counter of the first axis of the placing sequence is zero; the next piece will be the first of the next row.

Example:

```

...
IF Placingp-001 ZXY STACK READY
Conveyor-01: ON 5.0 S
ENDIF
...

```

The conveyor is switched on every time a complete stack has been placed by the placing program.

3.3.7.8.3 Jump Condition: PI-Peripheral Input

The branch will be executed if a selectable signal of the peripherals is ON or OFF.

After selecting the event, a list with the available signals is displayed. Select the desired signal by tapping it.



are used to specify if the branch has to be executed when the signal is ON or OFF.

With a ✓ at Latch the latch function is activated. At the displayed command an exclamation mark is added after the peripheral input:

```
JMP LABEL PIxxx! = ON
```

With this function activated, the control also triggers on very short ON-impulses of the input, and memorizes them until the next request occurs in the respective part program.

3.3.7.8.4 Jump Condition: IMM - Injection Molding Machine

The branch will be executed dependent on one of the signals of the IMM.

After selecting the event, a list with the available signals is displayed. Select the desired signal by tapping it.



are used to determine if the branch has to be executed when the signal is ON or OFF.

When a signal of the IMM interface is ON, the condition described by the signal is true.

The condition `MOLD WAS CLOSED = ON` indicates if the mold has been closed since the last takeout cycle. You can override the automated latch for this condition using its [logical teach command](#).

The condition `WAS REJECT-PART = ON` indicates if the reject part signal of the machine had been set for the last part taken out. When dealing with rejects, you should prefer checking this latch instead of checking the signal directly, because the reject part signal may be invalid after switching on the permission to close the mold. The latch will be set during the execution of the command for switching on the permit *Mold Close*, and will be reset next time when entering the mold or at the start of the reference travel.

The condition `OP. WITH IMM = ON` represents the result of a logical AND operation with the signals **Operation with Robot** and **Automatic**. Both signals must be ON in order for the condition to be TRUE. If the condition is used in a WAIT command, the robot will go to [operating mode Wait IMM](#) while waiting for the condition to be fulfilled.



When waiting for the mold to open, both signals should be tested. This will detect a possible failure of the interface, thus avoiding crashing the mold. Additionally, waiting for **Mold was closed** inhibits the robot from entering the mold, if no part has been produced since the last cycle. The first Wait command should wait for the **Mold is open** signal in order for the waiting time detection of the speed calculation of the [ECO-Mode](#) to work correctly.

Example:

```
WAIT IMM: MOLD OPEN = ON
AND IMM: MOLD CLOSED = OFF
AND IMM: MOLD WAS CLOSED = ON
```

3.3.7.8.5 Jump Condition: Vacuum

The branch will be executed depending on the feedback of the vacuum pressure monitoring of one of the vacuum circuits. The state of a vacuum pressure monitor is ON, when all parts of all suction cups of the respective vacuum circuit are completely sucked on. Use the Setup menu [Vacuum Pressure](#) to set the trigger values of the vacuum pressure monitors for the current Teachprogram.

This serves to ensure, upon pickup of a piece, that the piece is really held tightly on the cup before further motions are carried out.

After selecting the event, a list with the available vacuum circuits is displayed. Select the desired vacuum circuit by tapping it.



are used to specify if the branch has to be executed when the state of the vacuum pressure monitor is ON or OFF.

3.3.7.8.6 Jump Condition: AutoSwitch

The branch will be executed depending on the selectable ON / OFF state of an [AutoSwitch](#).

After selecting the event, a list with the 32 AutoSwitches is displayed. Select the desired AutoSwitch by tapping it.



are used to specify if the branch has to be executed whether the AutoSwitch is ON or OFF.

When AutoSwitches are configured as keys, in many cases the use of the latch function may prove as useful. With the latch function activated, the control unit will trigger on even very short ON impulses, memorizing them until the next request occurs in the respective part program.

The latch function is activated with a ✓ at the option **latch**. At the displayed command an exclamation mark is added after the AutoSwitch:

3.3.7.8.7 Jump Condition: Part program

The branch will be executed if a peripheral program is running or ready.

After selecting the event, a list with the available peripheral programs is displayed. Select the desired program by tapping it.



are used to specify if the branch has to be executed when the program is running or not.

3.3.7.8.8 Jump Condition: Area

The branch will be executed dependant on the roller switch S6 of the Z-axis. The jump conditions Mold Area and Placing Area are used to execute different reference travels and alternative routines.

Mold Area

The roller switch of the Z-axis(S6) is not actuated. The robot is above or inside the IMM.

Placing Area

The roller switch of the Z-axis(S6) is actuated. The robot is outside the IMM.

3.3.7.8.9 Jump condition: Position

The branch will be executed in accordance with the actual position of one of the robot axes.

After selecting the event, the desired axis is picked from the list with of available axes. It is now possible to move the chosen axes to the desired target position by using the drive buttons on the right side of the window.

At last, you have to select one of the operators \leq , \geq , $<$, $>$, $<>$, $=$ to define how the actual position of the axis has to relate to the target position given within the command for triggering the branch.

Example:

```
JMP INPERI Z  $\geq$  1000.0  
AND Z  $\leq$  1500.0  
AND Y  $>$  350.0  
...
```

Could be part of a reference program. It only jumps to the label INPERI if the Z-axis is positioned within the range from 1000 to 1500 mm, and at the same time the Y-arm holds a position bigger than 350 mm.



The operator $=$ should possibly not be used together with numerical axes.

3.3.7.8.10 Jump Condition: Mode

The jump condition **Mode** is available in an **All Modes Program** only.

The branch is executed dependent on the mode of operation of the robot.

After selecting the event, the desired mode of operation is chosen from the list **Select mode**.

Then select one of the operators \leq , \geq , $<$, $>$, $<>$, $=$ to define how the attached value of the actual has to relate to the mode of operation given within the command in order to trigger the branch.

The attached values of the different modes of operation are:

- 0 ...  Reference missing
- 1 ...  Manual
- 2 ...  Reference
- 3 ...  Blockstop
- 4 ...  Automatic

Use `IN BLOCK STOP DRIVE` to check if the robot finishes the current block after a block stop request. The effective mode during this is **Automatic** still.

Use `IN REFERENCE DRIVE` to check if the robot is currently executing one of the reference travels in operating mode *Manual*. This mode has an attached value of 1,5. The reference travel in the operating mode *Reference Missing* will be ignored.

Example:

```
IF MODE < BLOCK STOP
...
ENDIF
```

The program part between `IF` and `ENDIF` will only be executed during operation modes *Reference missing*, *Manual*, *Reference* and during the reference travels.

3.3.7.8.11 Jump Condition: Stacking sensor

The branch will be executed if the signal of one of the up to 8 stacking sensors is ON or OFF.

After selecting the event, a list with the available stacking sensors is displayed. Select the desired stacking sensors by tapping it.



are used to specify if the branch has to be executed when the stacking sensor is actuated or not.

3.3.7.8.12 Jump Condition: Start

The branch is executed when the automatic start button  is pressed.

3.3.7.8.13 Jump Condition: Production End Active

The branch will be executed, if earlier during automatic mode, with the [Production End](#) function activated, the command for closing the mold has been processed (without actually closing the mold).

3.3.7.8.14 Jump Condition: Input

The branch will be executed if a selectable input signal of the robot is ON or OFF.

After selecting the event, a list with the available signals is displayed. Select the desired signal by tapping it.

For keys on the right side of the window help you to access certain groups of input signals:



signals of the injection molding machine



signals of the peripheral inputs



signals of the grippers



signals of the vacuums



are used to specify if the branch has to be executed when the signal is ON or OFF.

3.3.7.8.15 **Jump Condition: Output**

The branch will be executed if a selectable output signal of the robot is ON or OFF.

After selecting the event, a list with the available signals is displayed. Select the desired signal by tapping it.

For keys on the right side of the window help you to access certain groups of output signals:



IMM signals of the injection molding machine



Peri signals of the peripheral outputs



Gripper signals of the grippers



Vac signals of the vacuums



On and Off are used to specify if the branch has to be executed when the signal is ON or OFF.

3.3.7.9 Start/Stop Part Program



The commands *START part program* and *STOP part program* are used to start or stop the peripheral program, all modes program or one of the 115 additional part programs from another part program.

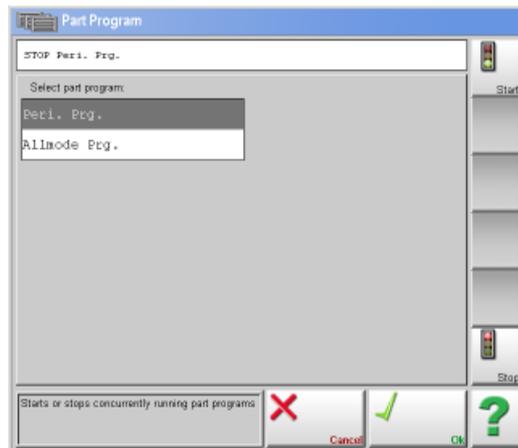


Fig.: Window for the Start/Stop part program commands

First pick the desired part program from the selection list **Select part program**.

Then use  or  to determine whether the command shall start or stop the selected part program.

 is used to add the command to the teach program.

 is used to discard the entries and to close the window.

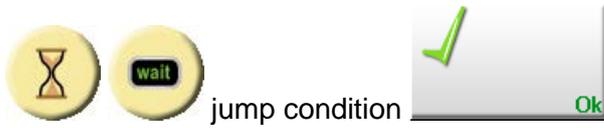
Example:

START PERI. PRG.

Starts the peripheral program

3.3.8 Wait functions

3.3.8.1 Wait



The WAIT command is used to halt the execution of the program until the event defined by the [jump condition](#) occurs.

3.3.8.2 Wait Timer



The command WAIT TIMER is used to link waiting for events with a time (limit).

First the time is set with the command WAIT TIMER, then the other [jump conditions](#) are added using OR and AND commands.

After opening the menu and tapping the input field **Time** the desired value for the time can be entered using the dialogue for [numerical input](#).

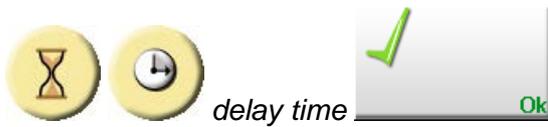


Example:

```
WAIT TIMER = 3.00 s
OR Vacuum-01 = ON
```

This waits for a part to be sucked on. If the set underpressure is not achieved within the time limit of 3 seconds, the wait command terminates, and the execution of the program is continued without the part. Afterwards a service routine for that case could easily be triggered using a command like `JMP label Vacuum-01 = OFF`.

3.3.8.3 Time



If a certain delay in the sequence is required, this can be achieved by means of the time function. The time function can be used to program time delays from 0.00 to 9999.99 seconds.

After calling up the function, the desired delay time can either be adjusted using the slider, or after tapping the field **input time** the desired time can be input using the dialogue for [numerical input](#).

In the automatic mode, processing of a program will be halted at a time function until the time has elapsed. Only then will the next instruction be started.

3.3.9 More Commands

3.3.9.1 More



This key opens a window with some additional teach commands.

Those commands are:

- Remark (REM)
- Production end
- Block stop
- Stopwatch
- Operating Mode
- ECO Mode
- Area Switch

3.3.9.2 Remark (REM)



Remark

In order to facilitate future editing, every program should be provided with a sufficient number of remarks. Remarks are shown in the teach program preceded by "REM":

```
0045 REM THIS IS A REMARK
```

After calling up the function and tapping the input field "Enter remark" use the dialogue for [text input](#) to enter the desired text.



Use  to add the command to the teach program.

3.3.9.3 Production End



This instruction is used for selective stopping of automatic operation at the end of production, without a part remaining inside the mold or on the EOAT.

Activate this function in automatic operation by actuating the  (REFERENCE) key. The control lamp of the key will begin to flash in confirmation of this function.

The exact sequence is like this:

Automatic operation

During normal automatic operation the command PRODUCTION END is being ignored, the jump condition [Production End Active](#) will be FALSE.

The button REFERENCE is being actuated during automatic operation

The LED of the button REFERENCE starts flashing. By pressing the button AUTOMATIC the production end function could be aborted. The jump condition [Production End Active](#) continues to be FALSE. The robot remains in normal automatic operation until the mold is not closed and the command for closing the mold is being executed.

At the command that switches on the permit MOLD CLOSE

The permission signal MOLD CLOSE is not being switched on, the mold stays open. The jump condition [Production End Active](#) will be TRUE from here on. Production end can no longer be aborted.

After the command that switches on the permit MOLD CLOSE

Commands waiting on the mold not-being-open will be ignored, in order to be able to deposit the last piece. The jump condition [Production End Active](#) can be used for triggering a special sequence (e.g. moving to the reference position) during production end.

At the command PRODUCTION END

The robot switches to the operating mode given by the command. Production end becomes inactive again, from now on all mold signals will be observed again.

After the command PRODUCTION END

When automatic operation is continued or restarted, the state of the function Production End will be normal **automatic operation** as described above.

Use ,  and  to select the operating mode to be triggered by the command.

Press  to add the command to the Teach program.

3.3.9.4 Blockstop lock/unlock



Normally, if you press the block stop key  in automatic operation the currently processed instruction (axis motion) will be completed and the robot will be halted and shifted into the block stop mode. The LED of the key will flash during this process.

The instruction LOCK BLOCK STOP allows a block stop to be carried out only after the UNLOCK BLOCK STOP instruction has been reached and all instructions found before it in the routine have been processed.

Thus an entire motion sequence (e.g., travel out of the mold area) can be completed with a block stop.

After calling up the function, use the keys  (lock) and  (unlock) to select the desired command.

Use  to add the command to the teach program.

3.3.9.5 Stopwatch



The Stopwatch function allows measurement of times that go by during the processing of arbitrary parts of a teach program. Four such software timers with a resolution of 0,01 seconds are available.

After calling up the function, first the select the desired stopwatch, then use the keys on the right side of the window to select the desired command.



Stopwatch-xx START

Starts the selected stopwatch. The current value of the stopwatch is incremented in steps of 0,1 seconds starting from zero while processing the program.



Stopwatch-xx STOP

Stops the selected stopwatch. The measured value is used to recalculate the minimum, average and maximum values for the [Info menu Stopwatch](#).



Stopwatch-xx RESET

Resets the calculation of the minimum, average and maximum values of the selected stopwatch.



Use  to add the command to the teach program.



The actual measured value can be loaded to a [Counter](#) for further processing, even while the stopwatch is running. 1 in the counter will represent 0,01 seconds.

Example:

```
-  
Vacuum-01 = ON  
Enter the IMM, use the ejectors to push the part to the vacuum suction cap(s)  
-  
Stopwatch-01 START  
-  
WAIT_VAC:  
Counter-001 = Stopwatch-01 CURRENT  
JMP TIMEOUT_VAC Counter-001 > 500  
JMP WAIT_VAC Vacuum-01 = OFF  
-  
Standard sequence with part on the EOAT  
-  
JMP ...  
  
TIMEOUT_VAC:  
Alternative sequence if the vacuum switch does not detect a part on the suction caps within timeout  
-  
JMP ...
```

Instead of WAIT VACUUM here a jump back to WAIT_VAC occurs until either the part is sucked on, or the time set for timeout is over. If the part is sucked on within time (here 5 seconds), the standard sequence is continued. During each run in the jumpback loop, the actual value of the stopwatch is loaded to Counter-001. If the value is bigger than 500 hundreds of a second, the program will jump to the alternative routine after the label TIMEOUT_VAC.

3.3.9.6 Operating Mode



The commands OPERATING MODE BLOCK STOP and OPERATING MODE MANUAL can be used to interrupt the automatic execution of the teach program, and to switch from operating mode **Automatic** to operating modes **Block Stop** or **Manual**. This is useful to stop the robot, especially if the teach program is able to detect error conditions.

The command OPERATING MODE PRODUCTION END requests [Production End](#). This is the equivalent of pressing the button during automatic operation.

The command OPERATING MODE REFERENCE can be used in the Robot Reference Teach program ROBOT-REF to switch the operation mode from **Manual** to **Reference**. This opens the possibility to start automatic execution of the Teach program at any desired position within the working range of the robot.



When the operating mode is set to Reference by the command OPERATING MODE REFERENCE, no verification of any kind is performed about axes positions or vacuum and gripper states. Also the standard reference travel will not be executed. This implies that if the command OPERATING MODE REFERENCE is used, any mistake in the Reference Teach program will almost certainly lead to collisions and damage during automatic operation.

After selecting the function use **Block Stop**, **Manual** or **Prod. End** to choose the desired command.



is used to add the command to the teach program.

3.3.9.7 Eco Mode



The command ECOMODE is used to control the ECO-Mode from within the Teach program.

Press  to select the command ECOMODE ON. The command ECOMODE ON turns the ECO-Mode on. Thus, there is no need then for the operator to manually turn on the ECO-Mode with a ✓ at the option **ECO-Mode** in the window [Override](#).

Press  to select the command ECOMODE BUFFER-TIME. This command sets the ECO-Mode buffer time. Thus, there is no need then for the operator to manually set the ECO-Mode buffer time in the window [Override](#) each time when changing programs with different requirements for that time. After tapping the input field **ECO-Buffer (s)** you can input the desired buffer time with the dialogue for [numerical input](#).

Press  to select the command ECOMODE INACTIVE. The command ECOMODE INACTIVE sets the ECO-Mode inactive for the current cycle, the ECO-Mode returns to active state automatically with the next cycle. This is used to execute special motions (that take significantly longer or shorter than the standard motion, e.g. reject part sequence) without ECO-Mode. For this cycle the robot will use the settings done with VMAX, VABS, AABS and the Override.

Press  to select the command ECOMODE RESET. The command ECOMODE RESET resets the calculations of the ECO-Mode. The robot will use the settings done with VMAX, VABS, AABS and the Override until it will have determined new values, and will be able to reduce acceleration and speed again.

Press  to add the command to the teach program.

See also: 3.3.9.7 ECO-Mode

3.3.9.8 Area Switch



This command is only available on robots with absolute encoders.

You can use the command Area Switch to set the switching points of the virtual mold interlock switches S5 and S6 according to the needs of the respective teach program. This should be done in the first lines of the robot reference teach program Robot Ref., in order for the conditions [Mold Area and Placing Area](#) to work correctly, when executing the teach programmed reference travel for the first time.



The new position values for the mold interlock switches will not be set before executing the area switch commands. While moving manually before executing the teach programmed reference travel, values may be active, that keep the [mold monitoring](#) from working correctly.



The area switch commands will set the position values for the mold interlock switches permanently. For the settings to be correct at all times, the commands – once they are used - have to be used in **all** Teachprograms of the robot.

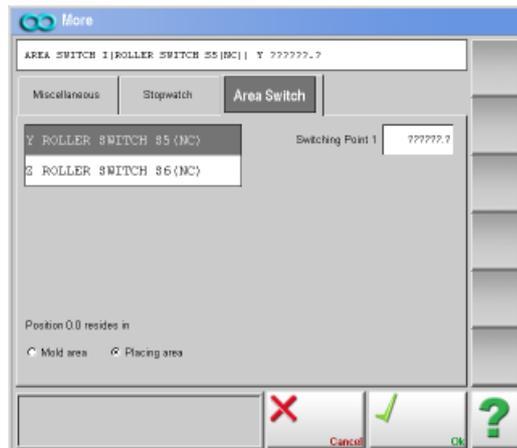


Fig.: Window for the Area Switch command

Use the selection field in the upper left corner of the window to select which switch you want to set with the command.

After tapping a field **Switching Point** on the right side of the window, you can set the selected switching point by either travelling there with the connected axes, or by inputting it numerically.

Use the options on the left bottom of the window to select the state of the switch on the 0 Position of the connected axis. **Mold Area** means off; **Placing Area** means on.

Press  to add the command to the teach program.

3.3.10 AutoSwitches

AutoSwitches are virtual switches and keys, that are displayed on the main screen of the Teachbox, when they are used in the Teach program. They serve as a replacement for real switches, which would have to be connected using peripheral inputs in order to use them in a Teach program. The AutoSwitches can be accessed directly by means of their [jump condition](#).



Fig.: Main screen with AutoSwitches

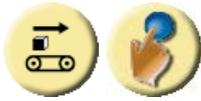
Up to 32 AutoSwitches are possible in one Teach program. If all the AutoSwitches used cannot be displayed in one line, a scroll bar for shifting to the desired line is shown.

The button  in the bottom right corner is used to select whether the AutoSwitches are displayed or not.

Settings in the Setup Menu

The basic settings for the AutoSwitches are made using the Setup menu [AutoSwitch Definition](#).

3.3.10.1 The Teach command AutoSwitch



The Teach command AutoSwitch is used to switch on or off an AutoSwitch that is configured as a switch from within a Teach program.

After calling the function and selecting the desired AutoSwitch, the buttons  and  are used to define if the AutoSwitch shall be switched on or off.

 discards the entries.

 adds the command to the Teach program.

See also:

3.2.6.5 Setup Menu: AutoSwitch Definition

3.3.7.8.6 Jump condition AutoSwitch

0 View Menu: AutoSwitch

Example: Part for quality control

The Autosw-01 has been renamed to CHECKPART and has been configured as key.

```

...
START :                               cycle start (label START)
...
Removal from IMM
...
JMP CONTROL CHECKPART!=ON           Branch to the label CONTROL, if the key CHECKPART has
                                     been pressed previously (latch!)
...
Placement, etc.
...
JMP START                             Jump to the cycle start
-
CONTROL :                             Label CONTROL
...
Deliver the part to the control station
...
JMP START                             Jump to the cycle start
-
    
```

After the key CHECKPART has been pressed shortly, the next part is delivered to the quality control station.

Example: start-up-parts

The Autosw-01 has been renamed to STARTUP, and has been configured as switch.

```
...
STARTUP=ON                               Switch on the switch STARTUP
...
STARTUPCYCLE:                             Start of the startup cycle - IMM "without" robot
PERMIT IMM:MOLD CLOSE! = ON
WAIT IMM:MOLD CLOSED = ON
JMP START STARTUP=OFF                     If the switch STARTUP is off, and the mold is closed, branch to
                                           the start of the normal cycle

WAIT IMM:MOLD OPEN = ON
FUNCTION IMM:EJECTORS FORWARD! WAIT
FUNCTION IMM:EJECTORS BACK! WAIT
JMP STARTUPCYCLE                           Jump to the start of the startup cycle
-
START:                                     Start of the "normal" cycle (label START)
...
Robot moves over the IMM, waits for the mold to open, etc. ...
...
```

The robot and the IMM are started together. The robot automatically switches on the AutoSwitch STARTUP at the beginning of the program. The sequence between the label STARTUPCYCLE and JMP STARTUPCYCLE toggled the permits for the IMM in a way to let the machine produce parts "without" the robots (and let them fall down). When the switch STARTUP has been switched off, the normal production cycle at the label START will be executed after closing the mold.

3.3.11 User Defined Error Messages

3.3.11.1 Introduction

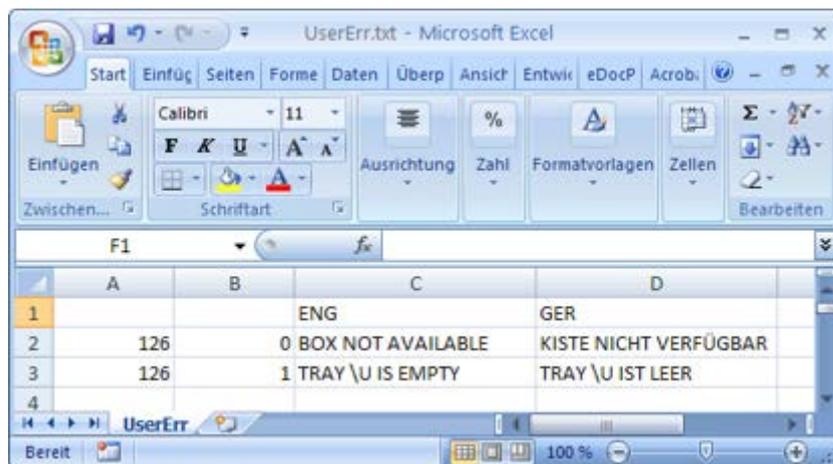
At the R8.3 control you can use the teach command **Alarm** to emit your own error messages.

Prerequisite is that a message table **UserErr.txt** is stored in the robot. Without this file, the teach command **Alarm** will not be offered by the Teachbox.

3.3.11.2 Creating and Storing the File UserErr.txt

The file Usererr.txt is a Unicode text file. The columns of the contained table are separated by tab stops. Use spreadsheet software like Microsoft Excel to create it, and save it as Unicode text.

Use ISO 639-2 language codes to define messages in up to 40 languages. The file must contain at least 3 columns (i.e. at least 1 language).



	A	B	C	D
1		ENG	GER	
2	126	0 BOX NOT AVAILABLE	KISTE NICHT VERFÜGBAR	
3	126	1 TRAY \U IS EMPTY	TRAY \U IST LEER	
4				

Fig.: Example for a UserErr.txt with 2 messages in 2 languages

In row 1 you define the columns for the different languages.

In column A you define error group 126 of the error number 126-XXX-YYY.

In column B define the unique error XXX inside of group 126. XXX can be any number from 0 to 255.

Write the message texts for the different languages into the other columns.



If a language is set on the Teachbox which is not available in the table, the texts from the first defined column will be displayed. In the example shown above this would be English from the column ENG.

The supplemental number YYY will be set by the teach command **Alarm**. You can display the contained information in the message texts using escape sequences:

\U ... 3-digit number with leading zeros. It shows the number YYY+1.
 \N ... 3-digit number with leading zeros. It shows the number YYY.
 \I ... 3-digit number no leading zeros. It shows the number YYY.
 \i ... 3-digit number no leading zeros. It shows the number YYY. 0 displays nothing.
 \J ... 3-digit number no leading zeros. It shows the number YYY. 0 displays nothing.
 If this results in two spaces in a row, only one will be shown.
 \A ... axis designation: 0=X, 1=Y, 2=Z, 3=A, 4=B, 5=C, 6=X2, 7=Y2, 8=Z2, 9=A2, 10=B2, 11=C2
 \V ... vacuum name (current name of vacuum YYY+1)
 \G ... gripper name (current name of gripper YYY+1)
 \O ... peripheral output name (current name of peripheral output YYY+1)
 \Y ... cylinder name (current name of cylinder YYY+1)
 \C ... conveyor name (current name of conveyor YYY+1)
 \S ... stacking sensor name (current name of stacking sensor YYY+1)
 \R ... safety area, 3 digits, YYY+1.
 \E ... peripheral input name (current name of peripheral input YYY+1)
 \Z ... counter name (current name of counter YYY+1)
 \M ... IMM Name (current name of IMM YYY+1)
 \K ... Autoswitch Name (current name of Autoswitch YYY+1)
 \W ... stopwatch name (current name of stopwatch YYY+1)
 \P ... placing program name (current name of placing program YYY+1)
 \p or \T ... part program name
 \H ... mode of operation: 0=Reference missing, 1=Manual, 2=Reference, 3=Block stop, 4=Automatic



If you actually want to show a „\“ in your text, you'll have to write „\\“ instead.

Store the finished table as a Unicode text file with the name **UserErr.txt** into the folder **..robot\tablo\language** on the **system USB stick** of the robots and use this to boot the robot. The file UserErr.txt will automatically be stored in the robot during the boot process.



Once the file has been stored on the robot, the included error messages will be available for all teach programs on that robot.

3.3.11.3 The Teach Command ALARM



Using the teach command **Alarm** you can emit your own error messages.

Prerequisite is that a message table [UserErr.txt](#) is stored in the robot. Without this file, the teach command **Alarm** will not be offered by the Teachbox.

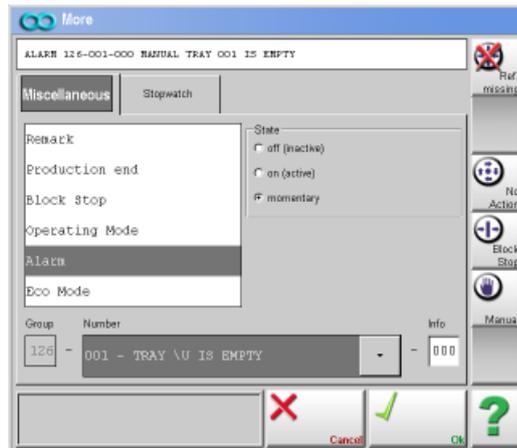


Fig.: Window for the teach command ALARM

After selecting the command, you select the desired message in the selection field **Number**.

Use the field **Info** to input the desired supplemental number. If the supplemental number is used in the message text, you can see the result already in the command display at the top of the window.

With the buttons and you define to which mode of operation the robot shall switch when the error message is triggered.

With the button you define that the mode of operation of the robot will not change, when the error message is triggered.

With the options for **State** you select whether the error shall be displayed as *active* (red) or *inactive* (gray):

Off (inactive) results in the error being shown in the state *inactive* permanently, or that an *active* error changes its state to *inactive*.

On (active) results in the error being shown in the state *active* permanently.

Momentary results in the error being shown in the state *active* for some seconds, and then changing to the state *inactive* automatically.

Press to add the command to the teach program.

3.4 Quick Editor

3.4.1 Introduction

The Quick Edit gives the possibility to display (even complex) Teach programs in a way that makes it easy to do simple program changes and adaptations.

Therefore only those parameters are displayed, which have been released for possible editing in the Teach Program.

3.4.2 Operation of the Quick Edit

3.4.2.1 Entering the Quick Edit

Use the button  on the main screen to enter the Quick Edit.

This is not only possible in Manual and Reference modes, but also when the robot is working in Automatic mode. In Automatic mode the same limitations as in the Text Editor apply; meaning that positions of numerical axes can only be changed within +/- 10mm.

3.4.2.2 Display and Editing Functions

The Quick Editor always shows the Teach program currently loaded in the robot.

Use Buttons on the right side of the screen select what types of parameters to display:



displays percentage parameters only (speed, acceleration, vacuum level).



displays time parameters only.



displays absolute speed parameters only.



displays position parameters only.



displays counter value parameters only.



displays placing program parameters only.



displays distance parameters only



displays all parameters.

The parameters out of the Teach program are displayed in numbered fields:

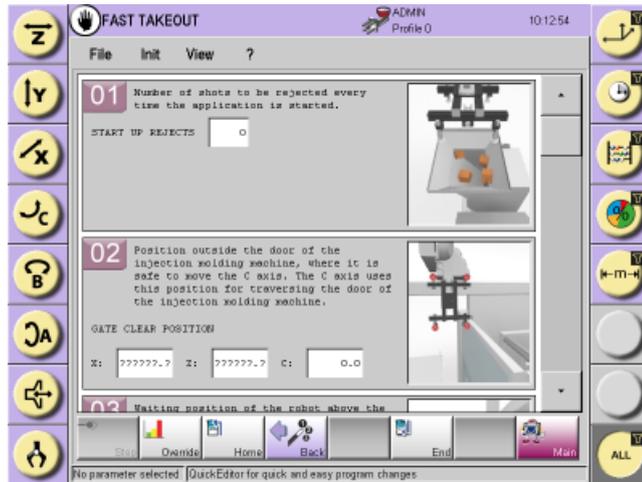


Fig.: Fields in the Quick Edit

The upper part of a field contains explanations of what the displayed parameter is used for in the program sequence. The quality of those explanations strongly depends on the skills of the creator of the respective Teach program.

The lower part of a field displays the parameter with its name and its changeable values. Tap the respective display field to open a pop up that you can use to edit the respective value.

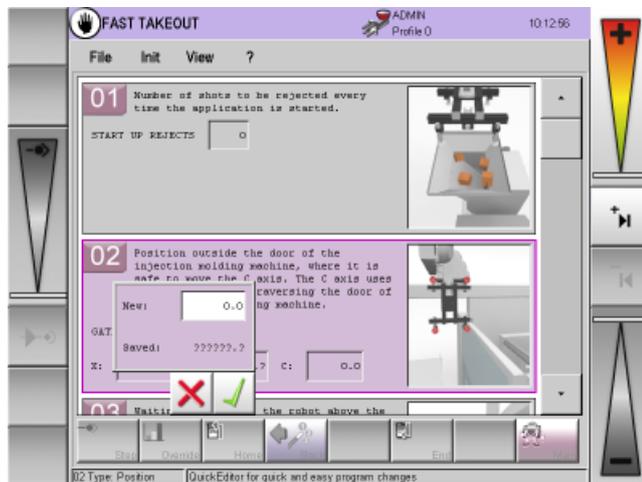


Abb.: Pop-Up in the Quick Edit to change a Position

Changes will be transferred to the Teach program in the robot immediately

- i** If the alterations are intended to be permanent ones, the Teach program has to be stored to a storage medium using the functions of the File menu.

3.4.2.3 Exit the Quick Edit

Use the button or the option **Exit to Main** of the **File** menu to exit the Quick Edit.

3.4.3 FILE Menu of the Quick Edit

3.4.3.1 Open

This function is used in the Quick Edit to load a teach program from a storage medium to the robot.

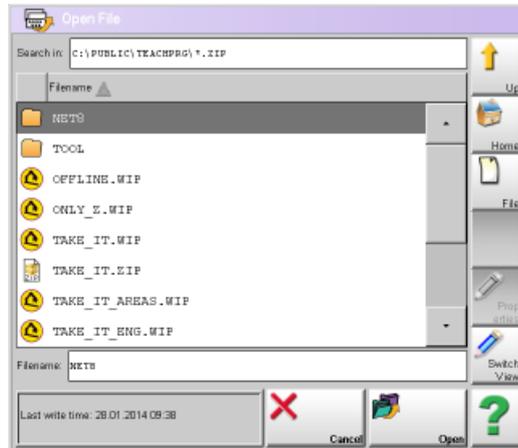


Fig.: Window to open a file in the Tooling Editor

The field **Search in** displays the path of the current folder. By touching the field you can select one of the standard folders C:\PUBLIC (internal storage medium of the Teachbox) or D:\ROBOT\TEACHPRG (USB-Stick).

The display of the contents of the current folder uses the following icons:

- for a sub-folder
- for a teach program file *.WIP
- for a teach program file *.ZIP

is used to step up one folder.

opens the standard folder for this file operation.

could be used to create a new folder.

is used to delete the selected file or folder.

is used to rename the selected file or folder.

is used to open the [properties](#) window of the selected program.



can be used to optimize the display of the contents of the folder for either long or short file names.



is used to open the selected folder.



is used to load the selected program to the robot.



is used to abort the function and close the window.



The program stored in the robot will be irretrievably deleted by loading another teach program.



Every newly loaded teach program must be run through at least once in step mode before automatic operation.



The file name of a teach program is not necessarily the same as the teach program name.

3.4.3.2 Open Template

Use this function to open Teachprogram Templates. Templates are programs prepared for certain standard applications in a way that you only need to parameterize them in the Quick Edit.

Teachprogram Templates are stored in a Read-Only location, where you can only load them from, but not overwrite. The Teachprogram you create from the template, you can store to any location available for storing Teachprograms using the function **Save As**.

The rest of the functionality of the menu **Open Template** is the similar to the menu **Open**.

3.4.3.3 Save (NAME.ZIP)

This function is used to save the teach program from the Teachbox a storage medium as a *.ZIP file. Both name and path of the file will be the ones used before. The name of the target file is shown in brackets right next to the menu entry **Save**. If no name is displayed, **Save** will call the function **Save as** instead.



The previous version of the program stored on the storage medium with the same name will be deleted irretrievably.

3.4.3.4 Save As

This function is used to save the teach program from the Teachbox to a storage medium as a *.ZIP file. Path and name of the file are selectable.

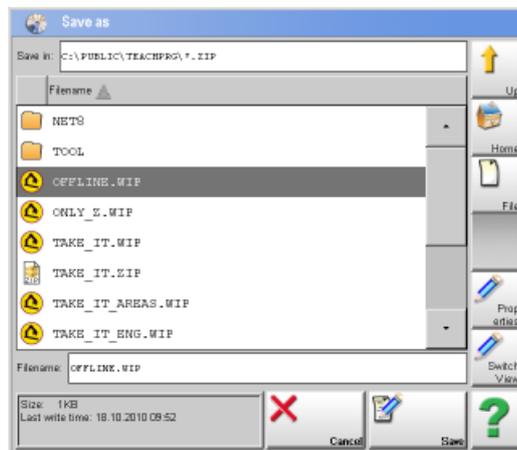


Fig.: Window to save a teach program

The field **Save in** displays the path of the current folder. By touching the field you can select one of the standard folders C:\PUBLIC (internal storage medium of the Teachbox) or D:\ROBOT\TEACHPRG (USB-Stick).

The display of the contents of the current folder uses the following icons:



for a sub-folder



for a teach program file *.WIP



for a teach program file *.ZIP

The field **Filename** suggests the last filename used or the teach program name as filename. After tapping the field you can use the dialogue for [text input](#) to change the filename as desired. The extension will always be ZIP. By tapping it in the display of the current folder, you can also use the name of an existing file.



is used to step up one folder.



opens the standard folder for this file operation.



is used to create a new folder.



is used to delete the selected file or folder.



is used to rename the selected file or folder.



is used to open the [properties](#) window of the selected program.



can be used to optimize the display of the contents of the folder for either long or short file names.



is used to open the selected folder.



is used to save the teach program to the current folder, applying the name from the field **Filename**.



is used to abort the function and close the window.

Before storing is executed, a check is made to see whether a program of the same name is already stored in the selected folder, and a checkback message may appear.

3.4.3.5 Properties

Here the properties of the teach program in the robot are displayed. That is the program which is executed when the robot is in automatic operation.

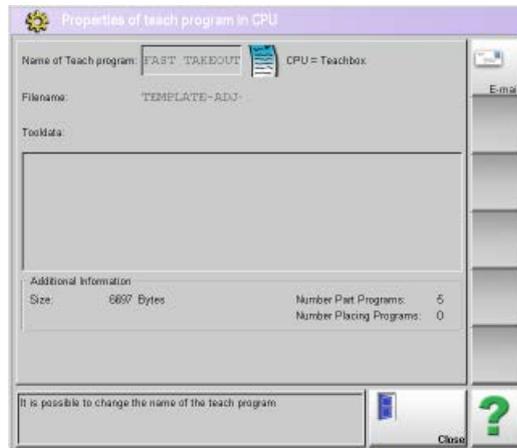


Fig.: Window for properties of the CPU program

Name of Teach program

Shows the name of the teach program of the robot. This name is not necessarily the same as the file name of the program. After tapping the input field, the name can be edited using the dialogue for [text input](#).

CPU = Teachbox or CPU <> Teachbox

Displays whether the teach programs of the robot (CPU) and the Teachbox are equal or unequal.

Tooldata

Displays the Tooldata text of the teach program of the robot. The text for Tooldata can only be edited in the properties window of the [text editor](#).

Size

Displays the size of the teach program of the robot. The maximum size of a teach program is limited to 32.768 bytes.

Number Part Programs

Shows the number of part programs of the teach program of the robot: ROBOT-PRG, PERI-PRG, ROBOT-REF, PERI-REF, ALLMODE-PRG and up to 11 additional part programs.

Number Placing Programs

Shows the number of placing programs used in the teach program of the robot. The maximum number of placing programs available in a teach program is limited to 16.

If a picture has been associated with the selected teach program,  can be used to open the [picture viewer](#).

 can be used to email the Teach program as an email attachment (*.wip, *.htm and *.txt file) using the [manual email function](#) of the robot.

Use  to discard the entries and to close the window.

Use  to confirm the entries and to close the window.

3.4.3.6 Print

This function is used to print the teach program loaded in the Teachbox to either a UNICODE-textfile *.TXT or an HTML-file *.HTM on the USB stick. Path and name of the file are selectable. The default folder is D:\ROBOT\PRINT. The generated file can be opened, displayed and printed with a PC.

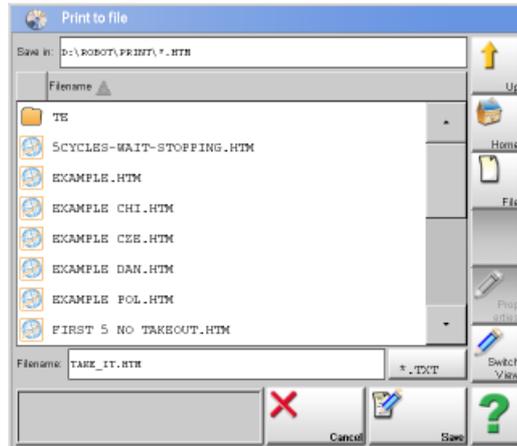


Fig.: Window to print a teach program to a file

The button right next to the field for the filename is used to select the type of file for printing:

activates printing to a textfile

activates printing to an HTML-file

The field **Save in** displays the path of the current folder.

The display of the contents of the current folder uses the following icons:

 for a sub-folder

 for a textfile *.TXT

 for an HTML-file *.HTM

Only files of the same type as selected for the printout will be displayed.

The field **Filename** suggests the last filename used or the teach program name as filename. After tapping the field you can use the dialogue for [text input](#) to change the filename as desired. By tapping it in the display of the current folder, you can also use the name of an existing file.



is used to step up one folder.



opens the standard folder for this file operation.



could be used to create a new folder.



is used to delete the selected file or folder.



is used to rename the selected file or folder.



can be used to optimize the display of the contents of the folder for either long or short file names.



is used to open the selected folder.



is used to save the printout to the current folder, applying the name from the field **Filename**.



is used to abort the function and close the window.

Before storing is executed, a check is made to see whether a file of the same name is already stored in the selected folder, and a checkback message may appear.

3.4.3.7 Exit to Main

Use the button  or the option **Exit to Main** of the **File** menu to exit the Quick Edit.

3.4.4 Parameters for the Quick Edit

In order to display values out of a Teachprogram in the Quick Edit, you will have to create parameters for those values and apply them in the Teachprogram.

3.4.4.1 Create Parameters

You can create parameters in the Text Editor using the function **Quick Edit Definition** of the **Setup** menu.

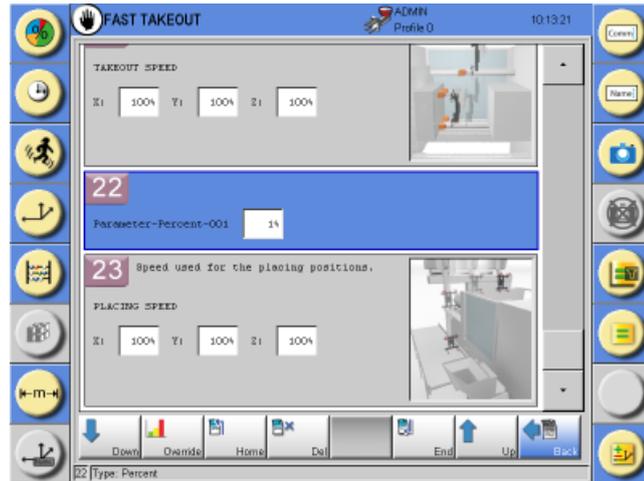


Fig.: Quick Edit Definition

Use the buttons on the left side of the screen to create parameters:

-  creates a percentage parameter for speed, acceleration and vacuum pressure.
-  creates a time parameter.
-  creates an absolute speed parameter.
-  creates a position parameter.
-  creates a counter value parameter.
-  creates a placing program parameter.
-  creates a distance parameter for commands with relative position values.
-  creates a parameter for predefined positions.

Press  to open the dialogue for changing the name of the parameter.

Press  to open the dialogue for inputting an explanation text for the parameter.

Press  to open the dialogue for managing axes values for axes related parameters. Here, press  to add an axis value to the parameter. You can remove axes values marked with a ✓ from the parameter with .

Press  to open the dialogue for determining the initial values, minimum and maximum values for the parameter.

Press  to open the dialogue for activating one or more display filters for the parameter. The filter for the parameter type of the parameter is activated by default.

Press  to open the dialogue for adding a picture or an animation to the parameter. For pictures use **PNG** files. For animations **MNG** files are supported. The recommended size is **150 x 175 pixels**. Press  to remove pictures and animations from the parameter.

If several parameters are present, you can use the buttons  Up and  Down to move the blue active one up and down in the sequence of parameters.

Press  to exit Quick Edit Definitions.

3.4.4.2 Apply Parameters

Previously created parameters can be applied to teach commands in the text editor. For this display the parameter toolbar with **Show Parameter Toolbar** in the **Edit** menu first.

When you select a teach command now, for which a parameter is available, you can use  to open the dialogue for applying parameters.

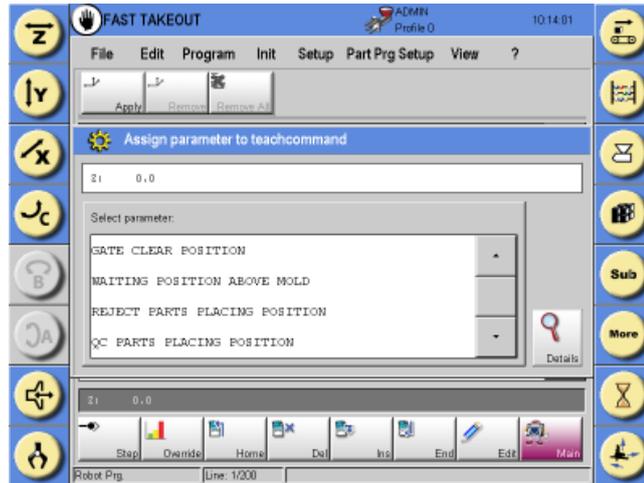


Fig.: Applying a parameter

Tap the parameter you want to apply to the teach command.

You can use  to remove the parameter from the selected teach command.

You can use  to remove all parameters from the selected teach command.

4 How to Teach a Program



Before teaching the robot a program for the first time, please read the safety instructions and the operating manual completely.

Familiarity with the operating functions of the R8.3 control is essential for creating a teach program.

4.1 Using the Quick New Wizard and the Quick Edit

Using the Quick New Wizard and the Quick Edit you can create simple pick and place programs. **No** programming skills are necessary.

First the Quick New Wizard will ask you some questions about the requirements for the program to be created. From your answers the wizard will then create the desired program and open it in the Quick Edit. Finally you will use the Quick Edit to input the required values for positions, times and speeds.

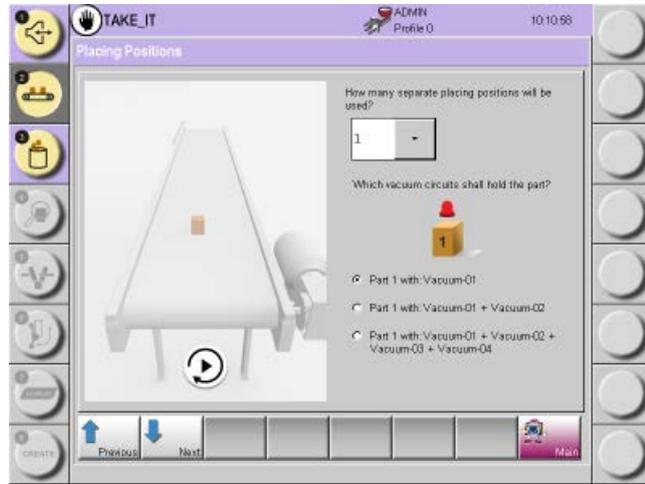
Proceed as follows:

- 1) Press  to start the Quick New Wizard.
- 2) In the first step you define if you want to takeout the parts using vacuum suction caps or grippers. Gripper-01 is reserved for the sprue, but can be used for parts nevertheless, if you do not take out sprue. Decide if you want to take out sprue using Gripper-01, and if you want to place the sprue before or after placing the parts.



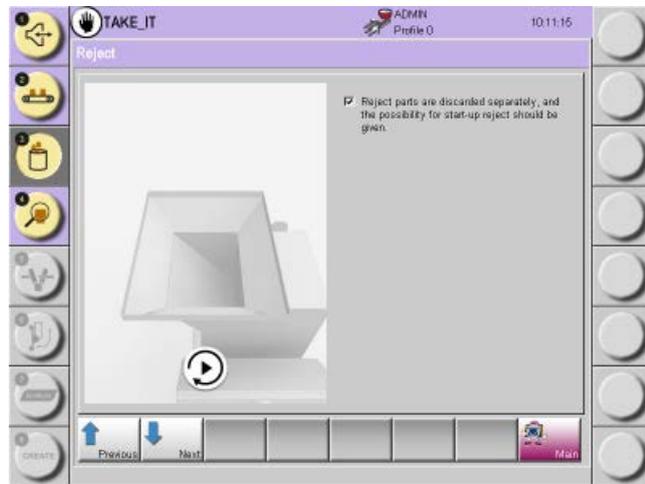
Press  to enter the next step of the wizard.

- 3) Define the number of placing positions per cycle that the robot has to travel to. Additionally define which vacuum / gripper circuits shall be used to carry the parts to the different placing positions.



Press  to enter the next step of the wizard.

- 4) Decide if rejects shall be placed separately. This functionality uses the reject part signal of the injection molding machine. Activating this option additionally gives you the possibility to separate an adjustable number of parts as startup rejects.



Press  to enter the next step of the wizard.

- 5) Decide for the possibility of using a button on the screen of the Teachbox to request a quality sample to be placed on a dedicated position.



Press  to enter the next step of the wizard.

- 6) Decide if the robot shall bring the removed parts to a device for cutting sprue, and define how many cutting positions shall be used. The cutting device will be triggered using peripheral output PO-001.



Press  to enter the next step of the wizard.

- 7) Decide if the ejectors shall move parallel to the in and out movements of the robot in order to save cycle time. Activate a softer, slower departure at the beginning of the out movement of the robot, if the parts have to be separated from the mold slowly.



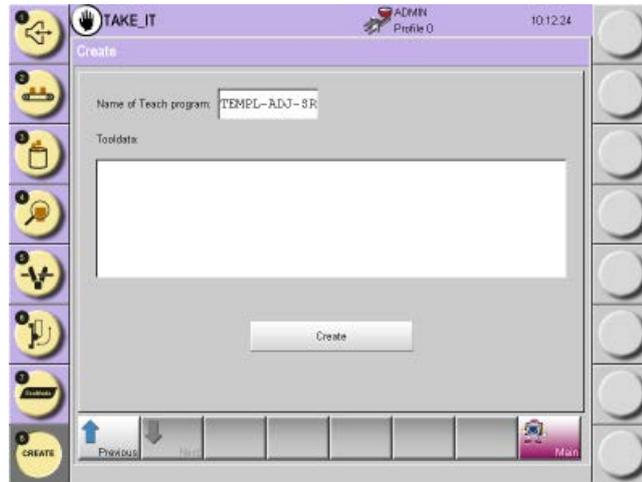
Press  to enter the next step of the wizard.

- 8) Decide if the program shall use the [EcoMode](#) for automatic adaption of the speed of the placing travel to the requirements of the machine cycle.



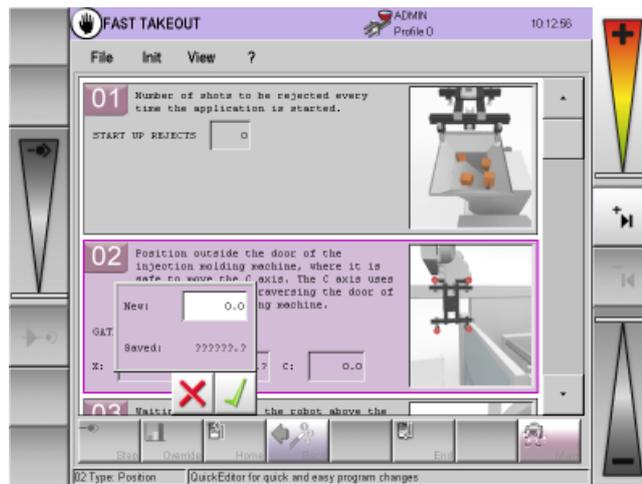
Press  to enter the next step of the wizard.

- 9) Name your new program, and add a comment, if necessary.



Press  for the wizard to create your program and to open it in Quick Edit.

- 10) Work through the displayed parameter values from top to bottom, and set the desired values. Use the [display- and editing functions of the Quick Edit](#) for this.



- 11) Test your program with reduced [velocity](#) using the [Dry Cycle function of the virtual sub pendant](#), or use the [Step functionality of the Text Editor](#).
- 12) Use the File menu [Save as](#) to store your new program on a storage medium. This is possible in the Quick Edit and in the Text Editor.

4.2 Programming the Robot in the Text Editor

The robot is programmed simply by manually moving the robot through the motions and functions that you want it to make in automatic mode. The instruction sequence created in this way is stored in a teach program and is then processed step by step in automatic operation. When the last instruction is reached, the program sequence is begun again at the first instruction.

A program may be created completely new (as described below) or may be a revision of an existing program. Once you are familiar with programming, revising an existing program and saving it with a new name is a much quicker way to create programs.

4.2.1 Creating a New Program in the Text Editor

- 1) Enter the Text Editor by pressing .
- 2) To start with a blank program, the program stored in the editor must first be deleted. This can be done by selecting the option **New** in the menu **File**. In the dialogue **Options for new teach program** certain elements of the existing program can be selected to be kept with the new program.
- 3) Set the [threshold of the vacuum under-pressure](#) for the utilized vacuum circuits.
- 4) Starting from the reference position, move the robot manually to trace the movements that you want the robot to make. Manual movements are made by

selecting the desired axis (labeled on the robot) with the buttons 
. After selecting the kind of motion, the axis can be moved to the desired position using the drive keys. (**+ direction** is away from reference position and **- direction** is toward reference position.)

When you reach the desired position, press  and the instruction will be stored as a line in the program. (Only the instruction active when you press the

 key will become a line in the teach program – all other motions made will be ignored.)

- 5) Repeat the process for the remainder of the motions and functions that you want the robot to reproduce.
- 6) The program is complete once you have programmed all of the motions and functions desired and the robot is in a suitable position to restart from the beginning of the program.

Caution!

After placing the parts the Y and X axes must be positioned suitable for traveling over the IMM, since the first motion in the program is typically a Z-axis (traverse) motion towards the molding machine.

4.2.2 Verifying Your Program Using STEP Mode

- 1) Bring the cursor back up to the beginning of the program using the button  .
- 2) Move the robot to reference position with the  key.
- 3) Test the program one line at a time in the STEP mode.
 Each time the  key is held down, the next instruction will be executed.
- 4) Releasing the  key will cause the robot to be halted immediately, in order to avoid collisions or other program errors. A flag will appear.
 Correct the program line if necessary and then continue to step through the program.
- 5) When the end of the program is reached, press the  key and the cursor will jump back up to the beginning of the program.
It is important to continue stepping through the program a second time to verify that there is no problem with the robot's position at the end of the program.

4.2.3 Storing Your Program on a storage medium

- 1) Connect a USB stick to the Teachbox, if required.
- 2) Press the button  to enter the text editor again.
- 3) Use the option **Save as** in the menu **File** to call up the function for saving programs. After tapping the field **Save in** select either the internal storage medium of the Teachbox or the USB stick as the target medium. The teach programs *.WIP already stored on the selected medium will be displayed.
- 4) Enter the desired name for the program using the dialogue for text input and confirm with  .
- 5) Press  to store the program and to exit the function.
- 6) Use  to exit the editor, press  to transfer your new program into the robot. Decide to reset all counters and placing programs or not.
- 7) You are now ready to run the program in full AUTOMATIC mode.

4.3 Program Example

The following example will help you to further understand and practice the teach programming of the system.



Please note that this is only an example. At a minimum, axis positions, IMM interface signals and program functions must be adjusted to your specific application!

The program example is based on a robot with 2 vacuum circuits and 2 gripper circuits.

4.3.1 Function of Program

Plastic housings are to be injection molded.

The robot's job is first to move into waiting position above the injection molding machine. When the mold opens, the robot removes the molded part. The molded part is stacked on a conveyor by means of a placing program. When the stack is finished, it is carried away by the conveyor.

4.3.2 Sequence

- The mold is closed manually for the first time. Then the robot is started in automatic operation. It is also possible to start robot and machine simultaneously.
- The robot travels to the waiting position above the mold, and waits for the „Mold Open“ signal from the machine.
- The molded part is removed from the mold using vacuum circuit 1.
- The robot travels back and up out of the mold and turns on the „Mold Close“ signal.
- The removed part is stacked using a placing program.
- When the stack is finished, the conveyor is indexed for 5 seconds.
- The presence of the removed part is sensed during the entire cycle by the vacuum sensor. If the part is lost, a STOP of the robot is executed.
- Speed settings shall be automated using Eco Mode.
- The robot shall execute a programmed reference sequence. If a part is on the gripper, it shall be placed on a safe position.

4.3.3 Sketch of Sequence

The ejector presses the parts on the vacuum cup.

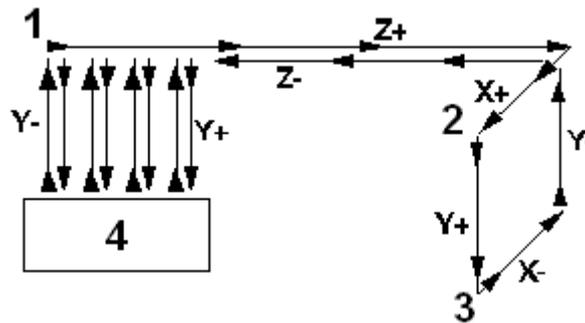
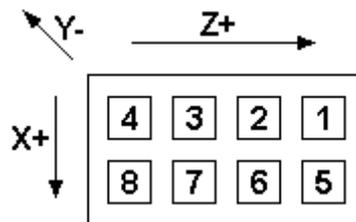


Fig.: Sequence

- 1 ... Reference position
- 2 ... Waiting position over the mold
- 3 ... Part removal position inside the mold
- 4 ... Placing positions in the box on the conveyor

4.3.4 Sketch of Depositing Program

The numbers below indicate the sequence of part depositing. Three layers are stacked in the vertical (Y) direction.



4.3.5 Reference Travel of Robot

The robot reference program must include axis motions in the mold, in order to allow reference travel out from the mold area. If a part is on the gripper, it shall be placed on a safe position. In reference position, all vacuum circuits are switched off and all grippers are opened.

4.3.6 Robot Teach Program (Robot Prg.)

0001	ECOMODE ON	Turn on automatic speed settings.
0002	PERMIT IMM: MOLD CLOSE! = ON	Close the mold for the first time.
0003	CALL CONVEYOR	Clear the conveyor belt using the subroutine CONVEYOR (line 32).
0004	-	
0005	CYCLE-START:	Label for starting a new cycle.
0006	PRODUCTION END MANUAL	Stopping point for production end.
0007	*Y: 0.0	Travel to a waiting position above the mold.
0008	*Z: 2353.0	
0009	*X: 123.6	
0010	Smart Removal: IN Y: 648.4 X: 202.4 ΔX: 30.0 Δt = 250ms	Travel to the takeout position using Smart Removal: Checks if the mold was closed, waits for mold opening, turns off the permit for mold closure, moves to takeout position time efficiently, with a straight X-movement of 30mm, and continues with the next line 0,25s before reaching the takeout position to turn the vacuum on earlier.
0011	Vacuum-01 = ON	
0012	WAIT SYNC	Wait for finishing moving in.
0013	FUNCTION IMM: EJECTORS FORWARD! WAIT	Move ejectors forward.
0014	WAIT Vacuum-01 = ON	Wait until the product is sucked on.
0015	FUNCTION IMM: EJECTORS BACK! WAIT	Move ejectors back.
0016	Vacuum-01 = M-Smart ON STOP	Activate vacuum monitoring before Smart Removal permits mold closure.
0017	Smart Removal: OUT X: 119.0 Y: 0.0 ΔX: 50.0 Δt = 0ms	Travel out of the mold with Smart Removal: After 50mm of straight X movement, it travels out of the mold time efficiently, and permits mold closure when reaching the Y mold safety switch.
0018	*X: 0.0	
0019	*Z: 1927.0	Outside of the door of the machine start turning the C axis.
0020	C: 90.0*	
0021	Placing Program: Placingp-001	Move to the next placing position.
0022	Vacuum-01 = OFF	Place the product.
0023	TIME = 1.00s	
0024	*Y: -200.0R	Move the axes from the placing position to a position that enables moving over the machine.
0025	C: 0.0*	
0026	*Y: 0.0	
0027	X: 0.0	
0028	WAIT SYNC	Wait for the C axis.
0029	CALL CONVEYOR Placingp-001 ZXY STACK READY	Once the stack is finished, index the conveyor using the subroutine CONVEYOR (line 32).
0030	JMP CYCLE-START	Begin a new cycle
0031	-	(label CYCLE-START in line 0005).
0032	SUBR CONVEYOR	Subroutine for the conveyor.
0033	Conveyor-01: ON 5.00 s	Turn the conveyor on for 5 seconds, and reset the placing program.
0034	Placingp-001 RESET	
0035	RET	Continue in the „main routine“, after the CALL command that called the subroutine.
0036	-	

4.3.6.1 Placing Program Placingp-001

```

Name of placing program: Placingp-001
Drive: 1D
Sequence of axes movement: ZXY
Sequence of placing: ZXY
Stacking sensor: No
Axis:
Start position:           X       Y       Z
relative endposition:    100.0  1000.0  600.0
Number of Parts:         2       3       4
Displacement:            0.0    0.0    0.0
    
```

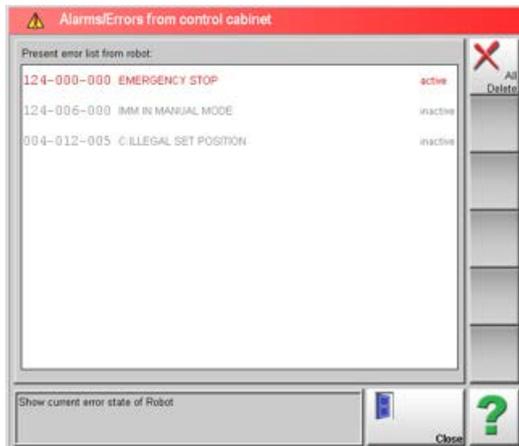
4.3.7 Reference Program for the Robot (Robot Ref.)

0001	IF MOLD AREA	Reference travel from in and over the mold.
0002	IF I(ROLLER SWITCH S5) = OFF	If inside the mold,
0003	WAIT IMM: EJECTORS BACK = ON	check for the ejectors,
0004	X: 114.9	and position the X axis first.
0005	ENDIF	
0006	Y: 0.0	Move out from the machine.
0007	X: 0.0	
0008	C: 0.0	
0009	ELSE	Reference travel in the placing area.
0010	Y: 0.0	Move up from the placing area.
0011	X: 0.0	
0012	ENDIF	
0013	-	
0014	IF Vacuum-01 = ON	If a product is on,
0015	Z: 1267.8	
0016	C: 90.0	
0017	X: 173.8	
0018	Y: 804.7	
0019	Vacuum-01 = OFF	place it on a safe position.
0020	TIME = 1.00s	
0021	Y: 0.0	
0022	X: 0.0	
0023	ENDIF	
0024	-	
0025	C: 0.0	Move to the reference position.
0026	Z: 0.0	
0027	Vacuum-01 = OFF	Switch all vacuum and
0028	Vacuum-02 = OFF	
0029	Gripper-01 = OPEN	gripper circuits to their reference state.
0030	Gripper-02 = OPEN	
0031	-	

5 Troubleshooting

5.1 Operating Errors

These errors are the result of operational causes. If such an error occurs, the relevant text error message will be displayed on the teachbox. Multiple errors are listed according to their priority:



Every error message is preceded by an error code with the following meaning:

Error group – Error number – Supplementary number

In addition, the error icon  is displayed on the right in the status line.

The majority of these errors cause an immediate stop of the robot and a switch to *MANUAL* mode. In order to continue automatic operation, first clear the errors by

pressing the  key, after their causes have been eliminated. The robot must be moved to reference position before restarting automatic operation.

Errors whose cause is still present are displayed in red. If the cause of error is eliminated or is no longer active, the display color changes to gray.

 Errors whose cause is still present cannot be acknowledged.

 Pressing  closes the window, but cannot acknowledge or delete errors.

5.1.1 Operating Errors in Plain Text Messages

5.1.1.1 Axis Errors

Group	No.	Supplement	Message	Cause/Remedy
004	000	0-11	xx: CONTOURING ERROR	Actual speed of axis lags behind internal calculations; can be caused by mechanical friction in guides/bearings or by too high a payload. May also indicate a defective converter or servo module.
004	001	0-11	xx :ENDSWITCH 0-POSITION	While the axis was moved, the end switch in - direction was hit.
004	002	0-11	xx :ENDSWITCH END-POSITION	While the axis was moved, the end switch in + direction was hit.
004	003	0-11	xx :MOTOR TEMPERATURE	
004	004	0-11	xx :FREQ. INVERTER FAULT	There may be many causes of converter fault. The display on the Inverter should be read and our nearest service facility should be contacted. The error messages of the controller can be confirmed by sending a drive command (selecting the axis emits the release signal). If for example 3 errors are active at the controller, the axis has to be selected 3 times. However, this error may also indicate a mechanical defect or sluggishness of guides/bearings.
004	006	0-11	xx :ENCODER FAULT	This error indicates that the incremental encoders do not count as provided. The reason for this could be a cable failure in the incremental transmitter line or a defect in the scoring logic. Another cause might be increased mechanical friction between guide shaft and bearing. In this case this error occurs mainly at very low speeds, especially in the first acceleration or last braking phase.
004	007	0-11	xx :STROKE LIMIT	The axis reported an actual position outside its stroke limits.
004	008	0-11	xx :SPEED LIMIT FAILED	During operation with dead-man-keys the axis moved faster than 250 mm/s
004	009	0-11	xx : SAFETY CONTACT FAILED	
004	010	0-11	xx : AXISPOSITION NOT ALLOWED	The pneumatic axis has stopped between valid positions, none of the position limit switches is actuated.
004	012	0-11	xx :ILLEGAL SET POSITION	There is an invalid set position in the teach program for the robot or the stroke limit was set too low. With activated safety areas this message appears when the robot would leave the safety areas.
004	013	0-11	xx :UNCONTROLLED SPEED	The numerical axis has left its set position during standstill. Brake or axes contactors might be defective.
004	015	0-11	xx : ROLLER SWITCH UNEXPECTED	The axis moved and the roller switch input changed its state without a memorized switching point near this position. The related mold safety switch (or its circuitry) is defective, or the position of its cam has changed.

Group	No.	Supplement	Message	Cause/Remedy
004	016	0-11	xx :OUT OFSAFETY AREA	
004	017	0-11	xx :AXIS NOT IN 0 POSITION	Before the start of the standard reference travel the B axis is checked for being in 0 position. If it is not, the reference travel is aborted with this error message.
004	018	0-11	xx : ROLLER SWITCH EXPECTED	The axis moved past a memorized switching point, but the roller switch input has not changed its state. The related mold safety switch (or its circuitry) is defective, or the position of its cam has changed.
004	019	0-11	xx :POSITION MONITORING	Position monitoring compares internally the listed end position with the actual position. If they do not coincide this error message appears.
004	020	0-11	xx :LATCH NOT OPEN	
004	021	0-11	xx :EJECTOR TOO FAST	At the Push function. The movement of the ejector was too fast, or the distance between intermediate position and target position is too small.
004	022	0-11	xx :TIME MONITORING	A pneumatic axis has not reached its set position within the time limit.
004	024	0-11	xx :COLLISION MONITORING	One of the (partly optional) collision monitorings has been triggered: Mold Monitoring, Roller Switch Monitoring, Safety Areas, Collision Monitoring AND/OR, Carrier Monitoring. According to the configuration of the robot.
004	025	0-11	xx :MANUAL MOVEMENT DENIED	Either the dead-man-key has not been released and pressed again, when switching between axes in manual mode of the safety package; or somebody tried to move further towards the obstacle, with the axis already positioned in violation of a safety area.
004	026	0-11	xx : ROLLER SWITCH REPEATEDLY	The roller switch input changed its state without an axis movement. The axis moved in negative direction and has travelled over the first memorized switching point already, but the roller switch input changed its state again. The axis moved in positive direction and has travelled over the last memorized switching point already, but the roller switch input changed its state again. The related mold safety switch (or its circuitry) is defective, or the position of its cam has changed.

Group	No.	Supplement	Message	Cause/Remedy
004	027	0-11	xx : LATCH NOT CLOSED	The latch of a digital B-axis could not be closed at the end of the movement.
004	028	0-11	xx : CRANE IN ROBOT AREA	With the optional crane interlock active, the robot may not be moved, while the crane is in the robot area.
004	029	0-11	xx : CABLE BREAK	All four inputs 0° Position, 90° Position, Track A and Track B of a digital (B-) axis are LOW.
004	030	0-11	xx : TORQUE MONITORING	The actual torque was bigger than the calculated maximum torque. Possible causes include collision, overload and sluggishness.
004	031	0-11	xx : CARRIER MONITORING	The collision monitoring "carrier monitoring" has been triggered.
004	032	0-11	xx : POSITION MONITORING (WAIT IMM)	The robot left its set position while being in the operation mode Wait IMM.
004	033	0-11	xx : BELT INSPECTION INTERVAL REACHED	When the belt inspection interval is reached the error message will display one time. Additionally, it will appear whenever the robot is being powered up and the interval has already been reached prior to powering off. The belt should be checked and checking must be confirmed.
004	034	0-11	xx : BELT REPLACEMENT INTERVAL REACHED	When the belt replacement interval is reached the error message will display one time. Additionally, it will appear whenever the robot is being powered up and the interval has already been reached prior to powering off. The belt must be replaced and the replacement must be confirmed.
004	035	0-11	xx : CARRIER MONITORING VABS	The collision monitoring "carrier monitoring" has been triggered during automatic mode while performing a VABS movement.
004	036	0-11	xx : PAYLOAD EXCEEDED (ENABLE DYNAMIC DRIVE)	Overload was detected. Reduce payload or acceleration or enable Dynamic Drive.
004	037	0-11	xx : ROLLERSWITCH OUT OF STROKE LIMIT	A switching point of an emulated roller switch is located outside of the stroke limits of the axis.
004	038	0-11	xx : BRAKE HOLDING FORCE TOO LOW	The automated brake test returned not enough holding force for the brake. This may lead to unexpected, uncontrolled movements of the axis at an idle robot. Have the brake (motor) of the respective axis be repaired/replaced absolutely.

The supplementary number indicates the axis, xx stands for the axis designation. 000 ... X, 001 ... Y, 002 ... Z, 003 ... A, 004 ... B, 005 ... C, 6 ... X2, 7 ... Y2, etc.

5.1.1.2 Vacuum Errors

Group	No.	Supplement	Message	Cause/Remedy
006	000	0..n	VACUUM xx PART LOST	This error message appears when the vacuum monitoring for the indicated circuit is activated and the vacuum pressure of that circuit falls below the preset threshold.
006	001	0..n	VACUUM xx NOT OFF	The vacuum reference position was chosen to be OFF, but at the end of the reference travel this vacuum circuit was not OFF.
006	002	0..n	VACUUM xx NOT ON	The vacuum reference position was chosen to be ON, but at the end of the reference travel this vacuum circuit was not ON.
006	003	0..n	VACUUM PUMP THERMO-MONITORING	Thermal contact of the vacuum pump responded to increased temperature of motor.
006	004	0...n	VACUUM xx DEAD-MAN-KEY NOT PRESSED	With safety not met, the selection switch has to be on manual and a dead-man-key must be actuated to enable switching of a vacuum.
006	005	0...n	VACUUM xx BLOW OUT NOT OFF	Vacuum blow out was not off at the end of the reference travel.

xx indicates the vacuum number, e.g., 01

Supplementary numbers: 000 = VACUUM 01
001 = VACUUM 02, etc.

5.1.1.3 Gripper Errors

Group	No.	Supplement	Message	Cause/Remedy
007	000	0..n	GRIPPER xx PART LOST	This error message appears when the gripper monitoring for the indicated circuit has been activated and the part monitoring input of that gripper signals part loss.
007	001	0..n	GRIPPER xx NOT OPEN	The gripper reference position was chosen to be OPEN, but at the end of the reference travel this gripper was not OPEN.
007	002	0..n	GRIPPER xx NOT CLOSED	The gripper reference position was chosen to be CLOSED, but at the end of the reference travel this gripper was not CLOSED.
007	003	0..n	GRIPPER xx TIME MONITORING	The gripper has not reached its set position within the time limit.
007	004	0...n	GRIPPER xx DEAD-MAN-KEY NOT PRESSED	With safety not met, the selection switch has to be on manual and a dead-man-key must be actuated to enable switching of a gripper.
007	005	0...n	GRIPPER xx NOT PRESSURELESS	The reference state of the gripper is RELEASED, but the gripper was actually OPENED or CLOSED at the end of the reference travel.

xx indicates the gripper number, e.g., 01

Supplementary numbers: 000 = GRIPPER 01
001 = GRIPPER 02, etc.

5.1.1.4 Peripheral Equipment Output Messages

Group	No.	Supplement	Message	Cause/Remedy
009	000	0..n	xx MONITORING	This error message is displayed when part monitoring for the indicated peripheral output is activated, and the peripheral input with the same number is OFF.

xx shows the PO names: e.g., PO-001

Supplementary numbers: 000 = PO-001
001 = PO-002, etc.

5.1.1.5 Cylinder Messages

Group	No.	Supplement	Message	Cause/Remedy
010	000	0...n	CYLINDER xx NOT ON	The cylinder has moved off its position limit switch during standstill. Possible causes are loss of air pressure or mechanical problems.
010	001	0...n	CYLINDER xx NOT OFF	The cylinder has moved off its position limit switch during standstill. Possible causes are loss of air pressure or mechanical problems.
010	003	0...n	CYLINDER xx TIME MONITORING	The cylinder has not reached its target position within the set timeout.
010	004	0...n	CYLINDER xx DEAD-MAN-KEY NOT PRESSED	With safety not met, the selection switch has to be on manual and a dead-man-key must be actuated to enable moving a cylinder.

5.1.1.6 Conveyor Belt Messages

Group	No.	Supplement	Message	Cause/Remedy
019	000	0..n	CONV. xx TEMPERATURE MONITORING	The thermal contact of the conveyor belt responded to increased temperature of the motor.
019	001	0..n	CONV. xx FOLLOWING TARGET POSITON REACHED	The conveyor following command has reached the target position P2. The error message has been triggered, and the Teach program will continue at the label for the alternative routine.

xx shows the belt number: e.g., 01

Supplementary numbers: 000 = CONVEYOR BELT 01
001 = CONVEYOR BELT 02, etc.

5.1.1.7 General Hardware Error Messages

Group	No.	Supplement	Message	Cause/Remedy
124	000	000	EMERGENCY STOP	An emergency stop button was actuated either at the IMM or at the robot. After resetting the emergency stop switch, the power on key at the subpendant must be pressed to switch on the control-voltage again.
124	001	000	AIR PRESSURE TOO LOW	Operating pressure dropped below 3.5 to 4 bars.
124	002	000	PHASE MONITORING	Phase monitoring in the control cabinet reports voltage fluctuations and asymmetries in supply exceeding a given adjustable range.
124	005	000	OPERATION WITHOUT ROBOT	In order to change to automatic operation, "Operation With Robot" must be on. Use the button on the Subpendant to switch on "Operation With Robot".
124	006	000	IMM IN MANUAL MODE	The injection molding machine was in manual operation when the AUTOMATIC key was pressed.
124	007	000	MOLD MONITORING	Mold safety ensures that the robot does not travel into a closed mold or that the mold does not close while the robot is in the mold area. Mold safety is realized with the roller switches S5 (Y axis) and S6 (Z axis).
124	008	000	IMM SAFETY DOOR OPEN	The safety door of the injection molding machine was opened during automatic operation.
124	009	000	DOOR NOT LATCHED	The door of the safety package could not be locked mechanically. The door contact may have become bent, not allowing the bolt to lock.
124	010	000	DOOR NOT SEQUENCED	The safety system of the robot reports that the safety requirements for operation have not been met.
124	011	000	DEAD-MAN-KEY TIME MONITORING	The permit key on the Teachbox must be pressed and released at regular intervals.
124	012	000	REFERENCE IN MOLD NOT ALLOWED	For safety reasons, the standard reference travel may not be carried out in the mold area. The error message appears when the robot is in the mold area and the operator did not program a reference travel. Note that in the NO REFERENCE mode the reference travel cannot be carried out in the mold area.
124	015	000	RELEASE DEAD-MAN-KEY	The permit key on the teachbox must be pressed and released at regular intervals.
124	016	000	COMMUNICATION FAILED	A communication problem with an inverter with CAN bus connection has occurred.
124	017	000	TIME MONITORING PROCESS COMMUNICATION	Time out at communication with an emergency stop block or with a Teachbox. In most cases normal operation is possible after confirming the error message.
124	018	000	SAFETY SYSTEM NOT ENABLED	The safety system of the robot reports that the safety requirements for operation have not been met. Also check EMERGENCY STOP of robot and IMM.
124	019	000	BM IN MANUAL MODE	
124	020	000	SR: x VIOLATED	A robot axis tried to enter an exclusionary safety area. Move the axis back out of the safety area.
124	028	000	CHECK CENTRAL LUBRICATION	Stop the robot as soon as possible and check the central lubrication system. (Run out of grease?)

Group	No.	Supplement	Message	Cause/Remedy
124	029	000	CENTRAL LUBRICATION FAILED	Check and repair the central lubrication system (Run out of grease?)
124	030	000	AUTOMATIC-MODE NOT POSSIBLE "opm"	"opm" represents the current mode of operation: 0...Reference Missing, 1...Reference Travel, 2...Reference, 3...Manual, 4...Block Stop, 5...Block Stop Travel, 6...Automatic
124	031	000	CENTRAL LUB. PRESSURE DROP NOT OCCURED	Check and repair the central lubrication system (blocked?)
124	032	mmm	IMM "mmm" MOLD IS OPEN	The "Easy Automatic Start Sequence" cannot be started.
124	033	mmm	IMM "mmm" MOLD POSITION INDEFINITE	The "Easy Automatic Start Sequence" cannot be started.
124	034	mmm	IMM "mmm" MOLD MONITORING BRAKING DISTANCE	An attempt was made to enter the respective IMM while its mold was not completely open.
124	035	000	LUBRICATION INTERVAL REACHED	Is triggered once, when the lubrication interval is being exceeded. Additionally the message will appear each time the robot is being powered up with the lubrication interval exceeded.
124	036	000	CONTROL VOLTAGE NOT ON	Indicates that the control voltage is not on. In order to be able to meet safety, the control voltage must be switched on, using the button "Control Voltage On" at the Subpendant (emergency stop block).
124	037	000	CONTROL VOLTAGE CIRCUIT PROBLEM	Hardware defect in connection with the control voltage relay.
124	038	000	CENTRAL LUBRICATION TANK EMPTY	The central lubrication system signals that its tank is empty.
124	039	000	IMM-INPUT DIFFERENCE DURING STOP	When restarting the robot after a stop, one or more IMM signals states are different than at the moment of stopping.
124	040	mmm	IMM "mmm" ROBOT OUTSIDE MOLD CIRCUIT PROBLEM	A problem with the generation of the "robot mold area free" signal was detected. Check related electrical circuitry and relay.
124	041	mmm	IMM "mmm" MOLD STATE NOT SAFE FOR ENTRY	The states of the inputs from the iMM are different from their reference (stored at the falling edge of the IMM signal Mold Is Closed) and they do not match the states of the outputs to the IMM.
124	043	000	REFERENCEDRIVE BLOCKED BY INPUT	The function "Prevent the reference drive" is configured. The standard reference drive is inhibited because the input S5 and/or S5.2 is not ON. Use a reference teach program to avoid the error message or use manual functions to drive onto the S5/S5.2 switch.
124	044	000	SAFETYRELAY MONITORING 24 V CIRCUIT PROBLEM	The safety relay monitoring hardware signals a 24 volts circuit problem.
124	045	000	SAFETYRELAY MONITORING 0 V CIRCUIT PROBLEM	The safety relay monitoring hardware signals a 0 volts circuit problem.
124	046	mmm	"mmm" MOLD OPEN SYNC IMM DIRECTION CHANGE	The mold movement changed its direction while the Mold opening synchronization command was being executed.
124	047	mmm	"mmm" ROBOT MOVES INTO MOLD WITH ACTIVE MOLD CLOSE SIGNAL	An attempt was made to move in to the mold while the permit Mold Close was on.
124	048	mmm	"mmm" MOLD POSITION MEASUREMENT INPUT OPEN	Nothing is connected to the analog input for the mold position measurement.
124	049	000	OUTSIDE WORKING AREA SR: x	A robot axis tried to exit the inclusive safety areas. Move the axis back in to the permitted area.

5.1.1.8 System Error Messages

Group	No.	Supplement	Message	Cause/Remedy
127	003	ggg	EXTERNAL TEACH PROGRAM CHANGE	A Teachprogram from an external source was received. The supplemental error number indicates the source like this: 0...Teachbox 1, 1...Teachbox 2, 10...IMM, 23...Remote Access
127	010	ttd	REAL-TIME VIOLATION	Internal software fault. Processing of the axes calculations was started too late. The supplemental error number shows the delay in milliseconds.
127	011	ttd	SEVERE CLOCK OVERFLOW	Internal software fault. Processing of the axes calculations was started too late. The supplemental error number shows the delay in milliseconds.
127	012	000	QUIT QUEUE FULL	Internal software fault while processing axes drive commands.
127	013	000	STROKE LIMIT CHANGE NOT ALLOWED	In case of axes being assigned to the All Modes Program, the Stroke Limits cannot be changed while the All Modes Program is running.
127	014	000	USER REALTIME-FUNCTION: EXECUTION TIME TOO LONG	Executing the user function "RTFunc" has taken too much time. The execution is stopped. Please check the user dll functions and restart the CPU.

5.1.1.9 CAN Device Messages

Group	No.	Supplement	Message	Cause/Remedy
141	000	aaa	xx :FREQ. INVERTER FAULT ff	Only on robots equipped with CAN-BUS controlled inverters. Please contact our service department for further information. The error messages of the controller can be confirmed by sending a drive command (selecting the axis emits the release signal). If for example 3 errors are active at the controller, the axis has to be selected 3 times. However, this error may also indicate a mechanical defect or sluggishness of guides/bearings.
141	001	mmm	mm : CAN-MODULE COMMUNICATION ERROR	An error occurred while trying to communicate with the CAN module with the indicated number.
141	002	bbb	Can Bus "bbb" WARNING or Can Bus "bbb" BUS OFF	This indicates communication problems on the indicated CAN bus line. WARNING indicates bad communication. BUS OFF indicates no communication.
141	003	bbb	"bbb" ROBOT INTEGRATION COMMUNICATION ERROR	The communication between two robots is interrupted because of an unpowered second robot or a cable break.

5.1.1.10 Teach program Compiler Messages

Group	No.	Supplement	Message	Cause/Remedy
160	000	000	TEACH PROGRAM TOO LARGE	The program exceeds the memory size.
160	001	lll	LABEL NOT FOUND	The label used by a JMP command could not be found. The supplemental error number indicates the label number.
160	002	sss	SUBROUTINE NOT FOUND	The subroutine used by a CALL command could not be found. The supplemental error number indicates the subroutine number.
160	003	000	COMMAND UNSUPPORTED	The command is not supported by the software version used in this robot.
160	004		NOT CONFIGURED DEVICE	The Teach program uses a device that is not present on this robot.
160	005	aaa	aa: TARGET OUT OF STROKE LIMITS	The target for the respective axis is outside its stroke limits.
160	006	000	TEACH PROGRAM INVALID	The Teach program is broken. Try to use a safety copy of the Teach program, if available.
160	007	000	CHECKSUM ERROR IN TEACH PROGRAM	The Teach program is broken. Try to use a safety copy of the Teach program, if available.
160	008	ppp	PLACING PROGRAM UNDEFINED	The Teach program contains the undefined placing program with the number indicated by the supplemental error number.
160	009	nnn	“part program name“ AXIS DEFINITION INCORRECT	The axis definition of the respective part program contains an axis that is not present on this robot.
160	010		CYLINDER DEFINITIONS DO NOT MATCH THE RUNNING ALLMODE PROGRAM	An attempt was made to transfer a Teach program, while the All Modes Program was running, with cylinder definitions not matching the running program.
160	011		VACUUM DEFINITIONS DO NOT MATCH THE RUNNING ALLMODE PROGRAM	An attempt was made to transfer a Teach program, while the All Modes Program was running, with analog vacuum settings not matching the running program.
160	012		CAN NOT REPLACE “part program name“ WITH NEW ALLMODE PROGRAM	An attempt was made to transfer a Teach program, while the All Modes Program was running.
160	013		TEACH PROGRAM NOT ACCEPTED IN „opm“	In the current mode of operation it is not possible to transfer the Teachprogram. The supplemental error number indicates the current mode of operation like this: 0...Reference Missing, 1...Reference Travel, 2...Reference, 3...Manual, 4...Block Stop, 5...Block Stop Travel, 6...Automatic
160	014		TEACH PROGRAM HEADER INVALID	The Teach program is broken. Use a valid Teach program.
160	015		TEACH PROGRAM VERSION NOT SUPPORTED	The version of the Teachprogram loaded to the robot is not supported. Eventually the program can be converted to a compatible version by loading it to the Teachbox and transferring it back to the robot.
160	016		ERROR ON READING TEACH PROGRAM FROM FILE	When a broken Teach program was transferred as a file, e.g. at integration.
160	017		TEACH PROGRAM FROM ANOTHER ROBOT	The control number stored in the Teach program does not match the control number of the robot. The Teach program can be loaded to the Teachbox, and after thorough checking be stored back into the robot. CAREFUL: The sequence and the positions of the program probably do not suit the current application! !
160	018		LOGICAL AND/OR SEQUENCE ERROR	AND/ OR commands were found without a preceding IF, JMP, WAIT, CALL command with a condition in the lines directly before the AND/OR commands

Group	No.	Supplement	Message	Cause/Remedy
160	019		RELOADED PREVIOUS TEACH PROGRAM	The new Teach program could not be transferred; the existing Teach program in the robot has been retained.
160	020	ppp	“ppp“ AXISCOUNTERVALUE NOT SUITABLE	The axes used in the indicated placing program do not match the axes definition of the part program.
160	021		INVALID COMMAND	The execution of the command is either not possible or not allowed. Examples: Release axis teach command but release function not configured. Numeric axis teach command but the axis is a digital axis.
160	022		MANUAL CONVERSION OF TEACH PROGRAM NECESSARY	The All Modes program has the wrong part program number. The program must be converted manually.
160	023		INVALID PARAMETER	A Teachcommand has a parameter applied that doesn't exist.
160	024		POSITION NAME NOT DEFINED	An axis movement command or placing program uses an undefined position name.
160	025		“ppp“ INVALID DIRECTION OF STACKINGSSENSORDRIVE	The stacking sensor movement of the indicated placing program is pointing into the opposite direction of the approaching movement.
160	026		GLOBAL TEACHPROGRAM COMPILE ERROR	The global teach program has been loaded, but it can't be executed. In a global teach program only All Modes programs may contain commands; axes may not be used.

The supplementary number indicates the axis, aa stands for the axis designation.
000 ... X, 001 ... Y, 002 ... Z, 003 ... A, 004 ... B, 005 ... C, 6 ... X2, 7 ... Y2, etc.

5.1.1.11 Teach program Interpreter Messages

Group	No.	Supplement	Message	Cause/Remedy
170	000	000	DIVISION BY ZERO	The Teach program attempted to do a division by zero
170	001	000	TOO MANY CALL-COMMANDS	This error appears when the nesting depth of subprogram calls becomes too big. The cause could be a teach program error or too frequent calling of CALL instructions without previous encounter of RET instructions.
170	002	000	RET WITHOUT CALL-COMMAND	Execution of the teach program has encountered a RET instruction before having executed a CALL command.
170	003	000	“part program name“ ALREADY ACTIVE	The Teach program attempted to start a part program that was already running.
170	004	000	RESULT OUT OF RANGE	The result of a counter / placing counter calculation is out of the range of possible values.
170	005	000	RUN ON SUBROUTINE WITHOUT CALL COMMAND	The program ran on a Subroutine command, without calling it with a CALL command.
170	006	sss	“stacking sensor name“ UNEXPECTED	The stacking sensor was actuated while processing a stacking sensor command or a placing program with stacking sensor, before the intermediate position was reached.
170	007	ppp	„ppp“ BOX CRASH MONITORING	An attempt was made to use the placing program to travel over the box below the box height. Check the position of the robot before execution of the placing program.

Group	No.	Supplement	Message	Cause/Remedy
170	008	ppp	“ppp” STACK CRASH MONITORING	An attempt was made to use the placing program to travel over the box below the height of the stack. Check the position of the robot before execution of the placing program.
170	009	ppp	“ppp” MOVEMENT CONTRARY TO STACKING DIRECTION	The traveling direction of the placing axis does not match the calculated placing direction. Check the position of the robot before execution of the placing program.
170	010	aaa	“aaa” ABSOLUTE MOVEMENT DURING CONVEYOR FOLLOWING	During the execution of the conveyor following command only relative movements are allowed.
170	011	000	SMART REMOVAL STRAIGHT FORWARD DISTANCE TOO LONG	The distance for straight travel in the Smart Removal command is bigger than the overall distance programmed for the respective axis.
170	012	aaa	“aaa” IS RELEASED	An attempt was made to drive the released axis using an axis drive command.
170	013	000	„part program name“ POSITION UNDEFINED	A Teach program command uses an undefined or not completely defined position.
170	014	000	AXIS PUSH INVALID MOTION DIRECTION	Axis Push command: The direction from current actual axis position to intermediate position is different than the direction from intermediate position to target position. A correct motion is not possible.
170	015	000	IMM PERMISSION TEACHCMD BUT NO OPERATION WITH ROBOT	Occurs on execution of a IMM permission teachcommand when the operation with robot for that IMM is inactive.
170	016	000	WAIT IMM INPUT SIGNAL, BUT IMM IN MANUAL MODE	Occurs on execution of a Wait IMM teachcommand when that IMM is in manual mode.
170	017	000	MONITORING ALREADY USED	A Teachcommand tried to activate a device monitoring that is already active.
170	018	000	PERMIT MOLD CLOSE ALREADY ACTIVE	The Teach program attempted to switch on the permit Mold Close although it was already on.
170	019	000	MOLD CLOSE THAUGHT INSIDE MOLD	The Teach program attempted to switch on the permit Mold Close while the robot was situated inside the mold of the IMM.

5.1.1.12 Configuration Errors

Group	No.	Supplement	Message	Cause/Remedy
180	000	000	CONTROL VOLTAGE CONFIGURATION	The output 511 - Control Voltage On is configured but not all other required IOs are configured. Required IOs: Input 511 - Control Voltage On, Input 510 - Control Voltage On Monitoring, Input 164 - Key Control Voltage On, Output 511 - Control Voltage On
180	001	000	ANALOG VACUUM CONFIGURATION	An analog vacuum monitoring is not correctly assigned to a vacuum device or a hardware device.

6 Maintenance

Following the recommended maintenance schedule is required to ensure trouble-free operation of the robot. The robot warranty does not cover neglected or poorly maintained equipment.

Maintenance must be performed by qualified personnel only.

6.1 General



The user of the robot is responsible for maintenance and monitoring of the safety devices and systems.



In particular, the safety systems marked  must be checked according to the instructions to ensure a safe and fully functioning robot system.



Before entering the work envelope of the robot to perform maintenance work, lock out the main power disconnect switch and turn off the compressed air supply to the robot.

IMPORTANT! Robots with pneumatic linear axes often use 5-way or 3-way valves, which can leave trapped high-pressure air in a cylinder even with air pressure removed. This can result in uncontrolled axis motions if one side of the cylinder is exhausted during a maintenance procedure. For this reason, it is recommended that all pneumatic axes be blocked to prevent unintended hazardous motions.



For workings that cannot be executed standing on the floor use standardized and certified ascent devices only.



For safe operation of the robot use original spare parts only.

6.2 Lubrication Instructions for Linear Guides and Gear Racks

6.2.1 Lubrication Intervals for Standard Robots

The lubrication intervals are calculated automatically by the control system and are displayed with a message "lube interval reached" and the lubrication flag  on the upper right-hand corner of your screen. Generally, it is necessary to lubricate the tracks and the bearings after a distance of 4000 km; the axis that completes this distance first is the one that determines the beginning of the lubrication interval.

6.2.2 Lubrication Procedures

LINEAR RAILS:

Remove old grease from the bearing wipers with a rag. Use a grease gun to fill the bearing until grease begins to come out of the bearing wipers.

GEAR RACKS:

Remove old grease with a rag and apply fresh grease to the entire length of the gear rack using a spatula or a brush.

6.2.3 Grease Specification

GREASE:

MANUFACTURER:	TYPE:
Bosch Rexroth AG	Dynalub 510
Castrol	Longtime PD2
Chemie-Technik	Ekalub GLS 135/N2
Klüber	Klüberplex BEM 34-132

6.3 Drive Belts

Drive belts have to be visually checked for wear and tear or cracks every 6 month. If any such are found, the affected drive belt has to be replaced immediately, in order to prevent damages and idle times.

For safety reasons we recommend to replace the drive belts every 24 months during a general maintenance session. Drive belts of high speed robots should be replaced every 12 months.



The refit of a drive belt may only be carried out by trained personal equipped with the proper tools, because also the tension of the belt has to be set correctly.

Our skilled technicians are available to perform the refit.

6.4 Motor Brakes

When entering the protective gate (operating mode BlockStop or Manual), please check if any sagging is visible at the axes.

If this should be the case, the brake of the respective motor must be renewed immediately.

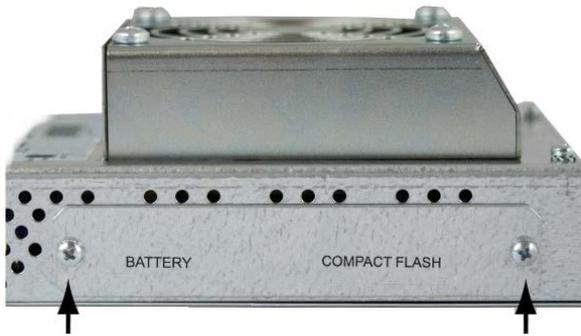


A brake refit may only be carried out by trained personal. Before starting to work on a vertical axis ensure that the axis is inhibited from falling down by suitable precautions.

6.5 Battery Replacement

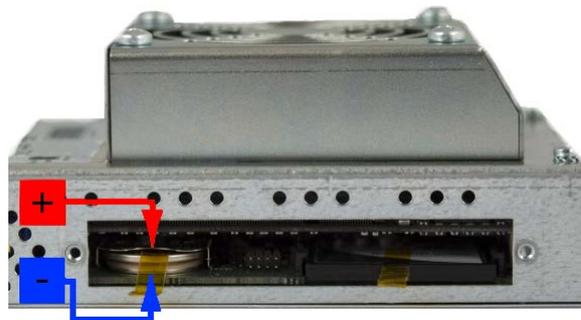
6.5.1 IPC

The capacity of the battery of the IPC in the cabinet is sufficient to secure the data over a period of 3 years without supply voltage. We recommend replacing the battery accordingly.



In order to replace the battery, remove the cover plate of the IPC.

The two screws securing the cover plate are marked with arrows in the adjacent illustration.



Now you can remove the old battery and install the new one.

When inserting the new battery, take care about the correct polarity according to the adjacent illustration.

Finally, re-attach the cover plate of the IPC.



The battery should only be replaced while the robot is powered down! After switching off the 24V supply, the battery of the IPC will be buffered for 5 minutes. The battery must be replaced within this time, else data loss will occur!

6.5.2 Sanyo R Inverters at Axes with Absolute Encoders

The capacity of the batteries in the Sanyo R inverters is sufficient to secure the zero points of the axes for a period of 5 years without supply voltage. We recommend replacing the batteries accordingly.

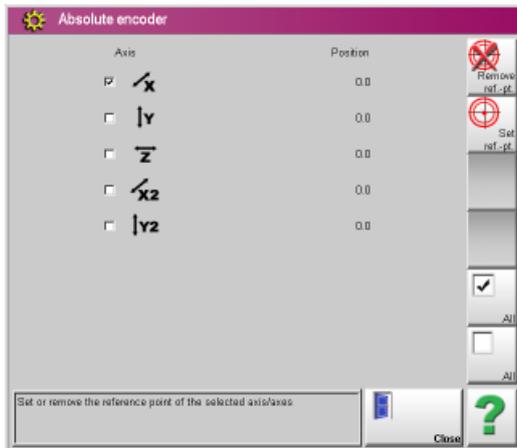


Fig.: Battery in a Sanyo R inverter

The procedure for replacing the batteries is as follows:

1. Move all affected axes to their zero points (with a [reference travel](#)).
2. Power down the robot.
3. Open or dismantle all covers as needed for accessing the inverters.
4. Flip open the front lid of the inverter.
5. Pull out the old battery and unplug it.
6. Push the new battery into the holder and plug it on. Make sure you feel the plug locking in correctly.
7. Close the lid of the inverter.
8. Repeat steps 4 to 7 for all affected inverters.
9. Close and re-attach all covers.
10. Power up the robot.
11. Login as [Admin](#).

12. Open the menu **Init – Absolute Encoders**.



13. Select all affected axes.



14. Press  in order to use the current positions of the axes as the zero point. Wait until the positions of all selected axes show 0.0.



15. Exit the menu with .



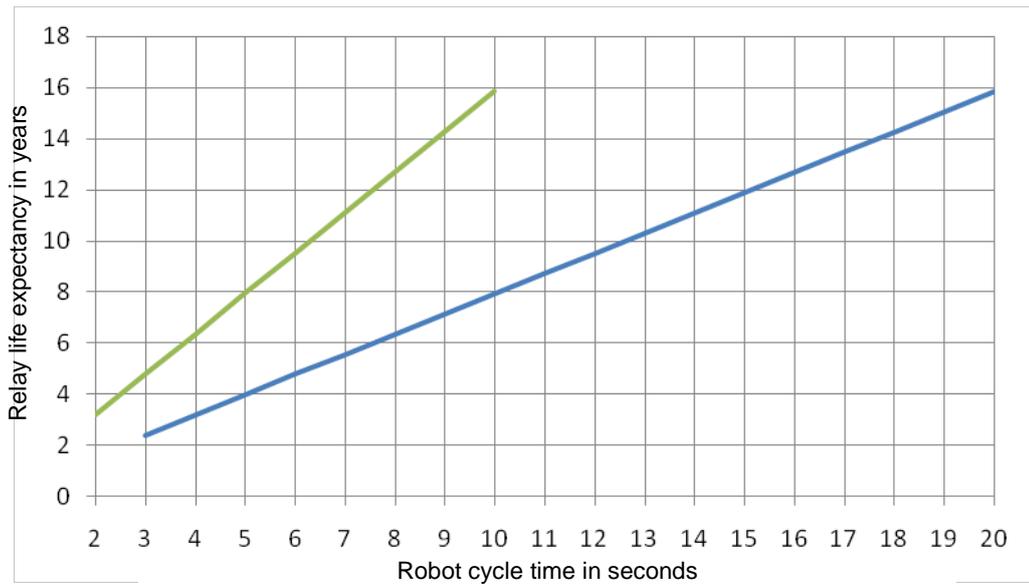
Do not move the robot while you carry out steps 3 to 15. Else the zero point settings of the robot will be lost!

6.6 Mold Safety Relays

The signals of the mold safety switches (“Roller Switches”) are processed and forwarded using safety relays. These have a limited number of switching cycles as their life expectancy.

We recommend having the mold safety relays replaced before the end of their life expectancy.

The life expectancy of the mold safety relays is determined by the cycle time of the robot and the number of switching cycles of the relays per robot cycle.



— shows the life expectancy of relays with 1 switching cycle per robot cycle:
 Z-axes mold safety relays of standard robots with 1 placing area,
 Mold safety relays of standard robots with absolute encoders,
 Mold safety relays of IML units.

— shows the life expectancy of relays with 2 switching cycles per robot cycle:
 Y- axes mold safety relays of standard robots.

Example:

A robot runs an application with a cycle time of 10 seconds.
 The life expectancy of the Y-axis mold safety relay is 8 years.
 The life expectancy of the Z-axis mold safety relay is 16 years.

6.7 Maintenance Schedule

<u>Description of Service</u>	INTERVAL IN OPERATING HOURS PER CYCLE TIME		
	<10 s	10-30 s	>30 s
<p>1) Safety Systems and Devices </p> <p><u>Emergency Stop Buttons:</u> When the device is at a standstill, activate the Emergency Stop switch and check to see if „EMERGENCY STOP“ is indicated on the hand set. The injection-molding machine must also indicate Emergency Stop.</p> <p><u>Function of Permit Keys on the Teachbox:</u> Switch robot to manual operation, set selector switch at the safety door to <i>Set-up Mode</i>. When the permit key is pressed, the “robot enabled“ light must go on. This function must be tested with both permit keys.</p> <p><u>Safety Roller Switches S5+S6:</u> Switch to operation without robot (robot LED key does not light). Close mold slightly (Mold Open signal must be out). Now try to ride into the mold area with the Y axis. The Y axis should ride only until leaving S5. The error message “mold monitoring“ must be displayed on the robot. Acknowledge message with ESC. Now try to close and open the mold. This should not be possible, and an error message should appear on the IMM. Now raise the Y axis.</p>	150	150	150
<p>2) Guides and Gear Racks</p>	See 6.2	See 6.2	See 6.2
<p>3) Other Components</p> <p><u>Function of Vacuum Switches:</u> The vacuum switch inputs should be checked using the Input Display screen on the teachbox. The input must only be on when a complete set of parts is on the vacuum cups. Be sure that the input switches back off when just one part is removed.</p> <p><u>Air Filter:</u> Drain the filter bowl and check and fill the oiler as required. IMPORTANT: Use only oil in accordance with ISO VG 32 KL1.</p> <p><u>Control Cabinet Cooling Fans:</u> Visually check if all fans are operating. Clean the filters of the fans or replace them when dirty (In high dust environments reduce the interval for checking).</p> <p><u>Drive Belts:</u> Visually check belt alignment and tension while the axis is moved through first full stroke. Inspect the belt for damage and “necked-down“ areas which indicate approaching belt failure.</p> <p><u>Rack and Pinion Clearance:</u> Check for minimum rack to pinion clearance of 0.1 mm (0.004“).</p> <p><u>Gearbox Gaskets and Drain Plugs:</u> Check for leaks.</p> <p><u>Cable Chains:</u> Inspect for proper alignment of cables in cable chain (IMPORTANT: Loose cable clamps can cause cable binding and chafing)</p> <p><u>Motor Brakes:</u> Visually check, if vertical axes sag during standstill.</p> <p><u>Screws:</u> Check robot screws and bolts for tightness.</p>	150	150	150
	150	300	500
	150	150	150
	150	300	500
	300	600	800
	300	600	800
	500	800	1000
	500	800	1000
	800	1200	2000
	800	1200	2000

6.8 Automatic Lubrication

6.8.1 Description

The optional, automatic system lubricates all linear bearings and the respective sprockets and racks of the robot. The system measures the X, Y, and Z-axis travel distances to determine when grease should be applied. When the lubrication interval (50 km) is reached, a fixed quantity of grease is injected into each bearing through the grease distributors.

The grease-pump is equipped with a pressure sensor. If the given pressure is not reached within 15 seconds after switching on the pump, or if the pressure doesn't drop below the given value within 15 seconds after switching off the pump, the error message „CHECK CENTRAL LUBRICATION“ will be displayed on the Teachbox. This will not interrupt the automatic execution of the robot teach program, but can be

confirmed with . The lubrication system should be checked as soon as possible, and the cause of the error should be fixed. Possible causes for the error are:

- Lack of grease in the container
- Leaking, unplugged or burst hoses
- Functional breakdown of the pump

If the cause of the error is not fixed before the next lubrication interval, the error message “CHECK CENTRAL LUBRICATION” will be displayed again, and can be

confirmed with  again.

Should the cause of the error again not be fixed until the 3rd lubrication interval, the error message “CENTRAL LUBRICATION FAILED” will be displayed, and the robot will switch to block stop. Now the cause of the error must be fixed.. After the cause of the

error was fixed, the grease pump can be started by pressing  inside the menu INIT/LUBRICATION INTERVAL a number of times until lubrication was executed correctly without error message. This enables the start of automatic mode again.

During filling of the grease pump container, extra precaution must be taken to avoid air bubbles in the grease and to use only the specified greases.

6.8.2 System Maintenance

It is important to remember that the automatic lubrication system does not eliminate maintenance requirements for the robot! It just reduces the frequency of the maintenance intervals. Lubrication system maintenance is crucial in detecting system faults, such as:

- Burst Hoses
- Plugged Tubes
- Air Pockets
- Faulty Lubrication Pump or Solenoid Valve
- Faulty Injectors

6.8.2.1 Lubrication System Maintenance Procedure:

1. Inspect the system for burst, leaking, or disconnected tubes and hoses.
2. Check that some grease build-up is present on all linear bearings wipers to verify that grease is reaching all bearings. Clean the grease off of the wipers to "reset" this test for next time.

Refill the grease pump as needed, being careful not to leave any air pockets. Use only the specified greases!

6.8.2.2 Suitable greases

Manufacturer	Type
Bosch Rexroth AG	Dynalub 520
Castrol	Longtime PD00
Chemie-Technik	Ekalub GLS 135/N00

7 Key Terms

For better understanding of these operating instructions, some key terms are explained below:

Term	Explanation
EJ	Ejector
BM	Blow machine
DIAS BUS	DIAS – Decentralized Intelligent Automation System
CNC	CNC - Computer Numerically Controlled – microprocessor control system
CPU	Central Processor Unit
Euromap 12	Interface standard between IMM and robot
Flag	Various letters/symbols indicating particular operating conditions
CP	Core puller
Position control	In the case of servo devices, in automatic operation the motor will be triggered continuously after a numerical axis has been positioned, so as to hold the position.
LCD Display	Liquid Crystal Display
Master	Control system in control cabinet
MS-DOS	Microsoft Disk Operating System – standard operating system for IBM-PCs and compatibles
Emergency Stop keypad	Additional keypad for full operator control of the robot without manual control unit
Offline	No communication between partial subgroup control units
Online	Communication between all partial subgroup control units is functioning
Override	Adjustment - speed adjustment
PO	Peripheral outputs
PI	Peripheral inputs
Robot	Control on the robot
IMM	Injection molding machine
Soft key	Key of uppermost row of keys on the teachbox, whose assignment varies depending on the current function.
Soft key line	Shows current assignment of soft keys.
Teach mode	Operating mode for programming of sequence programs
Teachbox	Manual control unit
Teaching	Programming of robot
Teaching	Programming of sequence programs
Teach program	Sequence program
T/M	Tool / Mold