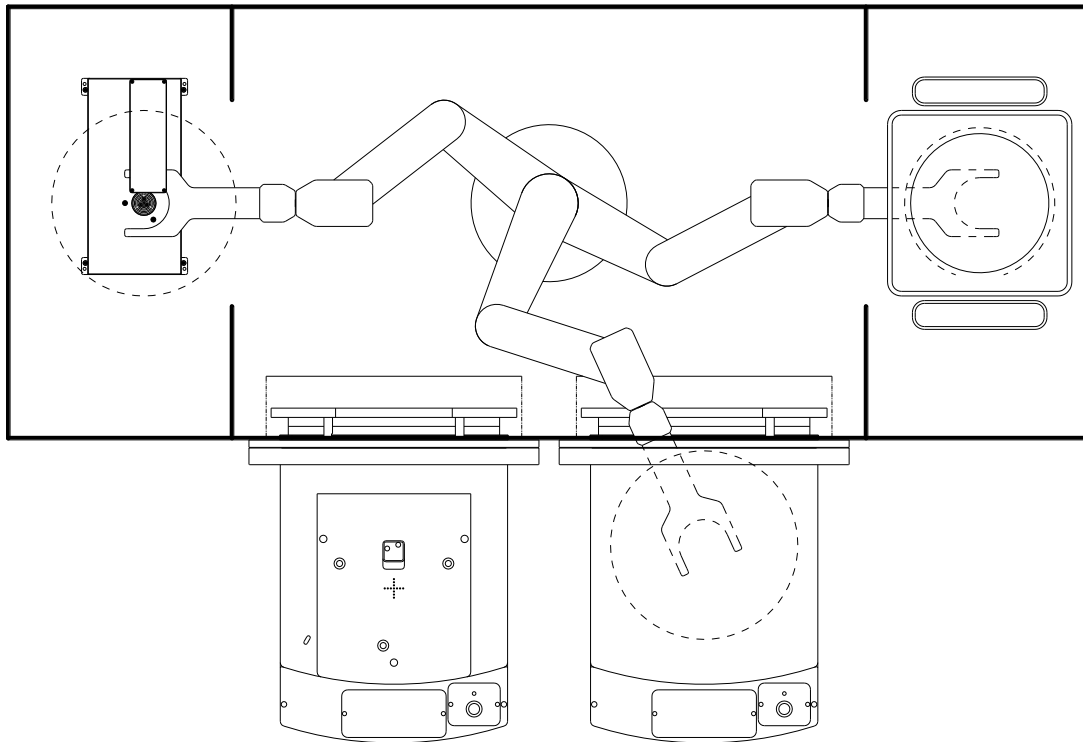


## **LOGOSOL WAFER HANDLING PLATFORM** **ROBOT APPLICATION MANUAL**

DOCUMENT #710010010

DATE: AUGUST 2021



The information herein has been carefully checked and is believed to be reliable, however no responsibility can be assumed for inaccuracies. Critical evaluation of this manual by the user is welcomed.

Comments assist us in preparation of future documentation. Please email feedback to [support@logosolinc.com](mailto:support@logosolinc.com).

All information herein is subject to change without notice and should not be construed as a commitment by Logosol, Inc.

Copyright © 1991-2021 Logosol, Inc. All rights reserved.

The information contained in this manual is the property of Logosol, Inc. This publication may not be reproduced, photocopied, stored on any retrieval system, or transmitted in whole or in part in any manner without prior written approval of Logosol, Inc.

All registered or unregistered trademarks are the exclusive property of their respective holders, and are used for reference purposes only.

## REVISION HISTORY

REVISION	DATE	DESCRIPTION	CONTACT
1.0	JULY 2004	INITIAL VERSION	G. Georgiev
2.0	NOVEMBER 2006	UNIFIED MANUAL FOR ROBOT AND SINGLE-AXIS PREALIGNER	G. Georgiev
3.0	APRIL 2007	UNIFIED USER, SOFTWARE AND SERVICE MANUALS	G. Georgiev
4.0	OCTOBER 2007	ADDED RCC INSTALLATION AND USAGE, GRIPPER CONFIGURATION, BATTERY BACKUP WARNING	G. Georgiev
5.0	AUGUST 2008	LOGOSOL MANUAL. DUAL-ARM SUPPORT, EXTENDED SINGLE-AXIS PREALIGNER	G. Georgiev
6.0	MAY 2009	UPDATED FOR H-SERIES WAFER HANDLING ROBOTS	G. Georgiev
7.0	MAY 2010	UPDATED FOR VACUUM-OPERATING WAFER HANDLING ROBOTS	G. Georgiev
8.0	AUGUST 2010	UPDATED FOR LATEST FIRMWARE RELEASES; PARAMETER FILES DESCRIPTION	G. Georgiev
9.0	SEPTEMBER 2012	UPDATED FOR LATEST FIRMWARE RELEASES: 2.4.7.0, 2.5.1.0	G. Georgiev
10.0	MAY 2017	NEW RELEASE: REVISED CONTENT, UPDATED FIGURES, FORMATTING	T. Norman
10.1	APRIL 2018	UPDATED FOR LATEST FIRMWARE RELEASES: 3.7.3.0 (SINGLE-ARM), 3.5.10.6 (DUAL-ARM)	G. Georgiev
10.2	OCTOBER 2019	LXB42 EXTERNAL CONTROLLER INTERFACE PANEL	V. Mihaylov
10.3	OCTOBER 2020	UPDATED FOR CURRENT FIRMWARE RELEASE	G. Georgiev
10.4	AUGUST 2021	TYPOGRAPHY, UPDATES FOR CURRENT FIRMWARE RELEASE	G. Georgiev

# TABLE OF CONTENTS

REVISION HISTORY .....	II
TABLE OF CONTENTS .....	III
TABLE OF FIGURES .....	VII
LIST OF TABLES .....	VII
<b>1 OVERVIEW .....</b>	<b>1</b>
1.1 INTRODUCTION .....	1
1.2 ENVIRONMENTAL, FACILITY, AND POWER REQUIREMENTS .....	1
1.3 SHIPMENT AND STORAGE SPECIFICATIONS .....	1
1.4 UNPACKING, INSPECTING, AND REPACKING .....	1
1.5 SERIAL NUMBER LABELS .....	2
1.6 HOW CAN I GET HELP? .....	2
<b>2 SAFETY .....</b>	<b>3</b>
2.1 DANGERS, WARNINGS, CAUTIONS, AND NOTES.....	3
2.2 PRECAUTIONS AND REQUIRED SAFEGUARDS .....	3
2.3 EQUIPMENT WARNING LABELS .....	3
2.4 PRECAUTIONS AND REQUIRED SAFEGUARDS .....	4
2.5 DEFINITION OF A MANIPULATING INDUSTRIAL ROBOT .....	4
2.6 SAFETY BARRIERS .....	4
2.7 WORK AREAS .....	4
2.8 IMPACT AND TRAPPING POINTS.....	5
2.9 POTENTIAL HAZARDS DUE TO EXPULSION OF A PART OR ATTACHED TOOLING.....	5
2.10 RISK ASSESSMENT.....	5
2.10.1 EXPOSURE.....	6
2.10.2 SEVERITY OF INJURY.....	6
2.10.3 AVOIDANCE .....	6
2.10.4 CONTROL SYSTEM BEHAVIOR CATEGORY .....	6
2.10.5 E-STOP - CE AND SEMI S2 COMPLIANCE .....	6
2.10.6 RISKS THAT CANNOT BE AVOIDED .....	7
2.10.7 RISKS DUE TO INCORRECT INSTALLATION OR OPERATION .....	7
2.11 INTENDED USE .....	8
2.12 ROBOT MODIFICATIONS.....	8
2.12.1 ACCEPTABLE MODIFICATIONS .....	8
2.12.2 UNACCEPTABLE MODIFICATIONS .....	9
2.12.3 SAFETY REQUIREMENTS FOR ADDITIONAL EQUIPMENT.....	9
2.13 QUALIFICATION OF PERSONNEL.....	9
2.14 SAFETY EQUIPMENT FOR OPERATORS.....	10
2.15 PROTECTION AGAINST UNAUTHORIZED OPERATION .....	10
2.16 SAFETY ASPECTS WHILE PERFORMING MAINTENANCE .....	10
2.17 WHAT TO DO IN AN EMERGENCY SITUATION.....	10

<b>2.18 ADDITIONAL SAFETY INFORMATION .....</b>	<b>10</b>
<b>3 INSTALLATION .....</b>	<b>12</b>
<b>3.1 INTERFACE PANEL LAYOUT .....</b>	<b>12</b>
3.1.1 EMERGENCY BRAKE RELEASE/EMERGENCY STOP CONNECTION .....	13
3.1.2 USING POWER DISCONNECT INSTEAD OF EMERGENCY STOP .....	16
3.1.3 SERIAL COMMUNICATION PORTS .....	16
3.1.4 SERIAL COMMUNICATION CABLES .....	17
3.1.5 TEACH PENDANT COMMUNICATION CABLE .....	18
3.1.6 EXTERNAL I/O MODULE OR LINEAR TRACK RJ45 CONNECTOR .....	18
3.1.7 ETHERNET RJ45 CONNECTOR.....	18
3.1.8 EXTENDED I/O INTERFACE CONNECTOR.....	18
3.1.9 SENSOR TYPE CONFIGURATION .....	19
<b>3.2 END-EFFECTOR VACUUM LINES .....</b>	<b>19</b>
<b>4 COORDINATE SYSTEMS AND UNITS OF MEASUREMENT .....</b>	<b>22</b>
<b>4.1 COORDINATE SYSTEM USED IN SINGLE-ARM WAFER HANDLING ROBOT .....</b>	<b>22</b>
<b>4.2 COORDINATE SYSTEM USED IN DUAL ARM WAFER HANDLING ROBOT .....</b>	<b>24</b>
<b>4.3 COORDINATE SYSTEM USED FOR LINEAR TRACK-BASED SYSTEMS .....</b>	<b>24</b>
<b>4.4 COORDINATE SYSTEM USED IN SINGLE-AXIS WAFER PREALIGNERS .....</b>	<b>25</b>
<b>4.5 COORDINATE SYSTEM USED IN THREE-AXIS WAFER PREALIGNERS .....</b>	<b>26</b>
<b>4.6 COORDINATE SYSTEM USED IN CASSETTE ELEVATORS .....</b>	<b>28</b>
<b>4.7 LIMITING JOINT TRAVEL.....</b>	<b>28</b>
<b>5 COMMAND TYPES AND CONVENTIONS .....</b>	<b>29</b>
<b>5.1 GENERAL CONSIDERATIONS.....</b>	<b>29</b>
<b>5.2 COMMAND SYNTAX, PROMPTS AND RESULTS .....</b>	<b>29</b>
<b>5.3 COMMAND EXECUTION AND REPORTING .....</b>	<b>30</b>
5.3.1 ECHO-AND-STATUS COMMUNICATION PROTOCOL.....	31
<b>5.4 HOST INTERFACE COMMANDS AND RESPONSES LOGGING .....</b>	<b>32</b>
<b>5.5 PARAMETER TYPES.....</b>	<b>32</b>
<b>5.6 CUSTOM MACRO COMMANDS.....</b>	<b>33</b>
<b>5.7 COMMAND GROUPS .....</b>	<b>33</b>
5.7.1 INFORMATIONAL COMMANDS .....	33
5.7.2 HOUSEKEEPING (PARAMETERS SAVE / RESTORE) COMMANDS .....	34
5.7.3 STATUS AND DIAGNOSTIC COMMANDS .....	34
5.7.4 SUPPORT FOR SYSTEMS WITH ABSOLUTE ENCODERS .....	34
5.7.5 BASIC (NON-AXES, NON-STATION) ACTION COMMANDS.....	35
5.7.6 BASIC AXIS MOTION PARAMETERS .....	35
5.7.7 AXIS PARAMETER SETS, AND CUSTOM PARAMETERS COMMANDS.....	36
5.7.8 BASIC AXIS MOTIONS COMMANDS .....	36
5.7.9 STATION PARAMETERS COMMANDS .....	37
5.7.10 STATION MOTION COMMANDS .....	38
5.7.11 STATION SCANNING PARAMETERS AND MOTION COMMANDS.....	39
5.7.12 ALIGN-ON-FLIGHT (ON-THE-FLIGHT, OTF) PARAMETERS AND COMMANDS .....	39

5.7.13 INTELLIGENT ALIGNER SUPPORT PARAMETERS AND COMMANDS .....	40
<b>5.8 AXES, INPUTS AND OUTPUTS: DESCRIPTION, AND COMMANDS USED .....</b>	<b>41</b>
5.8.1 AXES .....	41
5.8.2 INPUTS AND OUTPUTS.....	44
5.8.3 ENVIRONMENT MONITORING.....	46
<b>5.9 ARM CONFIGURATIONS.....</b>	<b>46</b>
5.9.1 HOMING PROCEDURE FOR SYSTEMS USING MOTORS WITH INCREMENTAL POSITION ENCODERS.....	46
5.9.2 SETTING HOME OFFSETS FOR SYSTEMS USING MOTORS WITH INCREMENTAL POSITION ENCODERS .....	47
<b>5.10 WAFER HANDLING ROBOTS WITH ABSOLUTE POSITION ENCODER .....</b>	<b>49</b>
<b>5.11 STATION PARAMETERS: DESCRIPTION, AND COMMANDS USED.....</b>	<b>50</b>
5.11.1 RADIAL STATIONS COORDINATES .....	51
5.11.2 INLINE STATIONS COORDINATES .....	52
5.11.3 STANDARD PICK PROCEDURE (A.K.A. GET) .....	53
5.11.4 STANDARD PLACE PROCEDURE (A.K.A. PUT).....	54
5.11.5 STATION-LESS IN-LINE MOTION COMMANDS.....	55
5.11.6 ON-THE-FLY WAFER CENTERING .....	55
5.11.7 STATION TEACHING EXAMPLE .....	55
<b>5.12 WAFER MAPPING: DESCRIPTION AND COMMANDS USED .....</b>	<b>57</b>
5.12.1 SCANNING AND MAPPING ALGORITHMS.....	58
5.12.2 WAFER MAPPING EXAMPLE .....	59
<b>5.13 ROBOT SYSTEM WITH STAND-ALONE SINGLE-AXIS SLAVE PREALIGNER .....</b>	<b>61</b>
5.13.1 PREALIGNER-SPECIFIC MACRO COMMANDS .....	62
5.13.2 SAMPLE COMMANDS FOR TEACHING ROBOT WITH SINGLE-AXIS PREALIGNER .....	62
<b>5.14 ROBOT SYSTEM WITH STAND-ALONE THREE-AXIS SLAVE PREALIGNER .....</b>	<b>63</b>
5.14.1 AXES NAMING .....	63
<b>5.15 CONFIGURATION OPTIONS FOR SINGLE-ARM ROBOT .....</b>	<b>63</b>
5.15.1 STANDARD CONFIGURATION.....	63
5.15.2 CONFIGURATION FOR ROTATING END EFFECTOR (REE).....	64
5.15.3 CONFIGURATION FOR EDGE GRIPPER.....	64
<b>5.16 TEACH-PENDANT OPERATION .....</b>	<b>66</b>
5.16.1 MODES OF OPERATION .....	66
5.16.2 KEYPAD DESCRIPTION.....	67
5.16.3 ADDITIONAL OPERATIONS .....	73
5.16.4 CUSTOM KEY ASSIGNMENT AND VELOCITY SETS .....	73
<b>5.17 TROUBLESHOOTING .....</b>	<b>73</b>
5.17.1 MOTION ERRORS.....	73
5.17.2 COMMAND ERRORS .....	74
5.17.3 DIAGNOSTIC COMMANDS .....	74
<b>6 COMMANDS REFERENCE .....</b>	<b>78</b>
<b>6.1 CONVENTIONS USED.....</b>	<b>78</b>
<b>6.2 COMMAND REFERENCE .....</b>	<b>79</b>
<b>6.3 ROBOT WITH SINGLE-AXIS PREALIGNER COMMANDS REFERENCE.....</b>	<b>226</b>
<b>6.4 DUAL-ARM ROBOT SYSTEM.....</b>	<b>254</b>

6.4.1 INTRODUCTION .....	254
6.4.2 AXIS NAMES .....	254
6.4.3 STATION COORDINATES .....	255
6.4.4 CHANGED COMMANDS .....	255
6.4.5 NEW COMMANDS .....	257
<b>6.5 ERROR CODES RETURNED BY EERR.....</b>	<b>261</b>
<b>7 SYSTEMS AND SERVICE INFORMATION .....</b>	<b>264</b>
<b>7.1 THEORY OF OPERATION.....</b>	<b>264</b>
<b>7.2 COMMAND LINE MODE.....</b>	<b>264</b>
7.2.1 ENTERING AND EXITING COMMAND LINE MODE .....	264
7.2.2 COMMANDS RECOGNIZED WHILE IN COMMAND LINE MODE .....	264
7.2.3 SYSTEM-LEVEL FIRMWARE COMMANDS.....	270
<b>7.3 ROBOT CONFIGURATION AND FIRMWARE FILES.....</b>	<b>290</b>
7.3.1 COMMUNICATION PARAMETERS FILE (MCL5.INI) .....	291
7.3.2 GENERIC AND MOTION CONTROL PARAMETERS FILE (PARAMS.INI).....	292
7.3.3 STATION PARAMETERS FILE (STATIONS.INI).....	294
7.3.4 PARAMETERS SET FILE (PARAMSET.INI) .....	296
7.3.5 CHANGES IN CONFIGURATION FILES FOR DUAL-ARM ROBOT SYSTEMS .....	297
7.3.6 HAND-HELD TERMINAL KEY ASSIGNMENTS FILE (HHTPARAM.INI) .....	297
7.3.7 HOST COMMUNICATION LOGGING PARAMETERS .....	299
<b>8 LOGOSOL WAFER HANDLING CONTROL CENTER .....</b>	<b>301</b>
<b>8.1 INSTALLATION AND INITIAL SET-UP .....</b>	<b>301</b>
<b>8.2 TERMINAL EMULATOR .....</b>	<b>304</b>
<b>8.3 STATIONS EDITOR .....</b>	<b>305</b>
<b>8.4 ERROR LOG BROWSER.....</b>	<b>305</b>
<b>8.5 CURRENT SYSTEM STATUS.....</b>	<b>306</b>
<b>8.6 COMMANDS SEQUENCER .....</b>	<b>308</b>
<b>8.7 FILE MANAGEMENT.....</b>	<b>309</b>
<b>9 SYSTEM MAINTENANCE .....</b>	<b>311</b>
<b>9.1 HARDWARE MAINTENANCE .....</b>	<b>311</b>
<b>9.2 SOFTWARE MAINTENANCE .....</b>	<b>311</b>
<b>9.3 FIRMWARE UPGRADE PROCEDURE.....</b>	<b>311</b>
9.3.1 DOWNLOADING NEW MACRO FIRMWARE FILE .....	311
9.3.2 DOWNLOADING NEW SYSTEM FIRMWARE EXECUTABLE.....	312
9.3.3 UPLOADING PARAMETER FILES FROM CONTROLLER .....	313

## TABLE OF FIGURES

Figure 2-1: E-Stop basic circuitry (DB15 connector).....	7
Figure 3-1: Emergency Stop and Brake Release connector – DB15 Connector.....	14
Figure 3-2: : Emergency Stop and Brake Release connector – AMP Connector .....	15
Figure 3-3: Single-arm robot : Dual vacuum line control board in lower arm link.....	21
Figure 3-4: Dual-arm robot: Single vacuum line in octa-box, for each arm .....	21
Figure 4-1: Three-axes Robot Coordinate System.....	22
Figure 4-2: Arm at initial (home) position ( $T=0$ , $R=0$ , $Z=0$ ) .....	22
Figure 4-3: Robot Arm coordinates at arbitrary position ( $T>0$ , $R>0$ , $Z>0$ ) .....	22
Figure 4-4: Dual-arm robot Coordinate System and initial (home) positions .....	24
Figure 4-5: Dual-arm robot at arbitrary position ( $T > 0$ , $R1 > 0$ , $R2 < 0$ , $Z > 0$ ) .....	24
Figure 4-6: Generic Coordinate System for track-mounted robot.....	25
Figure 4-7: Single-axes Prealigner Coordinate System .....	25
Figure 4-8: Single-axes Prealigner to Robot – positioning parameter .....	26
Figure 4-9: Three-axes Prealigner Coordinate System.....	27
Figure 5-1: Axis Homing Diagram.....	47
Figure 5-2: T, R coordinates of a station .....	50
Figure 5-3: Z coordinate of a station .....	50
Figure 5-4: Radial Stations Configuration .....	51
Figure 5-5: In-line Stations Configuration.....	52

## LIST OF TABLES

Table 2-1: Maximum Joint Velocities in Runaway Situations.....	5
Table 3-1: Emergency Stop/Brake release (DB15).....	14
Table 3-2: Emergency Stop/Brake release (AMP) .....	16
Table 5-1: Typical values for velocity parameters .....	42
Table 5-2: Single-arm Robot Standard Inputs.....	45
Table 5-4: I/O number assignments for external I/O interface: Single-arm robot .....	46
Table 5-5: I/O number assignments for external I/O interface: Dual-arm robot .....	46
Table 5-6: Robot configuration for single-arm robot.....	64
Table 5-8: Default Hand-Helt Terminal Key Assignment .....	69

# 1 OVERVIEW

## 1.1 INTRODUCTION

This manual applies to wafer handling devices that are controlled by the Logosol Wafer Handling Platform, including robots designed by Logosol and manufactured by 3<sup>rd</sup> parties, and those jointly designed by Logosol and 3<sup>rd</sup> parties.

**Note:** For simplicity, we refer to the objects being moved or aligned as a “wafer” although the specific application may not be using a silicon wafer. A wide variety of substrate materials may be used, but the principles of operation and function are the same.

**Note:** This manual covers robot wafer handlers firmware build on current MCL5 platform. Refer to older versions of the manual for ETS-based systems.

## 1.2 ENVIRONMENTAL, FACILITY, AND POWER REQUIREMENTS

Wafer handling components are designed to work in following environmental conditions:

WHR Recommended Environment				
Ambient Temperature		5°C to 40°C (41°F to 104°F)		
Humidity		5% to 90%, non-condensing		
Nominal Voltage Range	Frequency/ Phasing	Minimum Operating Voltage	Maximum Operating Voltage	Recommended Power Protection (user-supplied)
100 to 120 VAC	60Hz, 1-phase	100 VAC	120 VAC	20 amps
200 to 240 VAC	50Hz, 1-phase	200 VAC	240 VAC	10 amps

## 1.3 SHIPMENT AND STORAGE SPECIFICATIONS

This equipment can be shipped in a variety of boxes and shipping crates, depending on the manufacturer and sales order, and must be shipped and stored in a temperature-controlled environment, as follows:

- Temperature range -25°C to +65°C;
- Humidity range 5% to 90%, non-condensing.

The equipment should be shipped and stored upright in manufacturer-supplied packaging, which is designed to prevent damage due to normal physical shock, vibration, and/or dust. The package must be protected from excessive shock and vibration.

Use a forklift, pallet jack, or similar device to transport and store the packaged equipment.

## 1.4 UNPACKING, INSPECTING, AND REPACKING

Before signing the carrier’s delivery sheet, the recipient is expected to compare the actual items received (not just the packing slip) with the corresponding equipment purchase order and to verify that all items are present and the shipment is correct and free of visible damage.

If the items received do not match the packing slip or order, or are damaged, customer should not sign the receipt documents and must contact the supplier immediately.

Inspect each item for external damage as it is removed from its container. If any damage is evident, contact your supplier.





**CAUTION:** Never lift the robot by the outer link, inner link, or vertical drive column. Damage and misalignment may occur.



**WARNING:** The weight of the robot is more than 35 kg (77lbs). Do not try to carry the unit by yourself.

All containers and packing materials should be retained as they are necessary to properly re-ship equipment.

If the equipment needs to be relocated, use original packing containers and materials and follow all safety notes used for installation. Improper packaging for shipment will void your warranty. Before unbolting the robot from the mounting surface, place the arms in a folded position and install the arm alignment fixture. The robot must always be shipped in an upright position.

## 1.5 SERIAL NUMBER LABELS

The robot serial number label is located on the interface panel. Always have this serial number available when you call technical support.

## 1.6 HOW CAN I GET HELP?

Visit: [www.logosolinc.com](http://www.logosolinc.com)

Email: [support@logosolinc.com](mailto:support@logosolinc.com)

Contact your supplier.

## 2 SAFETY

### 2.1 DANGERS, WARNINGS, CAUTIONS, AND NOTES

There are four levels of special alert notations used in this manual. In descending order of importance, they are:



**DANGER:** Indicates an imminently hazardous situation which, if not avoided, will result in death or serious injury.



**WARNING:** Indicates a potentially hazardous situation which, if not avoided, could result in serious injury or major damage to the equipment.



**CAUTION:** Indicates a situation which, if not avoided, could result in minor injury or damage to the equipment.

**NOTE:** Provides supplemental information, emphasizes a point or procedure, or gives a tip for easier operation.

### 2.2 PRECAUTIONS AND REQUIRED SAFEGUARDS

This manual must be read by all personnel who install, operate, or maintain the systems that are its subject, including any personnel who work within or near the work cell.



**CAUTION:** The manufacturer strictly prohibits installation, commissioning, or operation of the prealigner without adequate safeguards. These must be compliant with applicable local and national standards. For more information on applicable standards please refer to the manufacturer-provided compliance documentation.

### 2.3 EQUIPMENT WARNING LABELS

There are labels attached to the robot to warn the operator of situations that may result in injury, as follows:

**Equipment Warning Label — Lifting****Equipment Warning Label — Danger**

## 2.4 PRECAUTIONS AND REQUIRED SAFEGUARDS

This manual must be read by all personnel who install, operate, or maintain the systems that are its subject, including any personnel who work within or near the workcell.



**CAUTION:** The manufacturer strictly prohibits installation, commission, or operation of the prealigner without adequate safeguards. These must be compliant with applicable local and national standards. Installations in EU and EEA countries must comply with EN 775 (Esp. Sections 5 and 6), EN 292-2, and EN 60204-1 (Esp. Section 13).

## 2.5 DEFINITION OF A MANIPULATING INDUSTRIAL ROBOT

The definition of a manipulating robot according to (ISO 10218:1992(E)) is as follows:

*"A manipulating robot is an automatically controlled, reprogrammable, multipurpose, manipulative machine with several degrees of freedom, which may be either fixed in place or mobile for use in industrial automation applications."*

## 2.6 SAFETY BARRIERS

Whenever power is applied to the equipment, safety barriers must be provided to prevent personnel from entering the work cell. Wafer handling systems are computer controlled and may be activated by remote devices under program control or move along paths not anticipated by personnel. It is critical that safeguards be in place to prevent personnel from entering the work cell whenever power to the equipment is present.

The robot system integrator (user or operator) must ensure that adequate safeguards, safety barriers, light curtains, safety gates, safety floor mats, etc., are installed. The robot workcell must comply with applicable local and national standards.

The height and the distance of the safety fence from the robot must ensure that personnel cannot reach the danger zone of the robot.

The wafer handling control system has features that aid the user in constructing system safeguards, including customer emergency stop circuitry and digital input/output lines. The emergency power-off circuitry is capable of switching external power systems and can be interfaced to the appropriate user-supplied safeguards.

## 2.7 WORK AREAS

WHRs have Teach and Normal operating modes. While in Normal Mode, personnel are not allowed in the work cell. In Teach Mode, operators with additional safety equipment are allowed to work in the robot work cell. For safety reasons the operator should, whenever possible, stay outside of the robot work envelope to prevent injury. The maximum speed and power of the robot is reduced but it could still cause injury to the operator.

Before performing maintenance in the working envelope of the robot, power supply to the robot must be disconnected. After these precautions are taken, a skilled person is allowed to perform maintenance on the robot.



**CAUTION:** Never remove any safeguarding and never make changes in the system that will decommission a physical safeguard.

## 2.8 IMPACT AND TRAPPING POINTS

Wafer Handling Robot (WHR) Systems are capable of moving at high speeds. If a person is struck by a robot (impacted) or trapped (pinched), serious injury may occur. Robot configuration, joint speed, joint orientation, and attached payload all contribute to the total amount of energy available to cause injury.

## 2.9 POTENTIAL HAZARDS DUE TO EXPULSION OF A PART OR ATTACHED TOOLING

The maximum joint tip speeds that can be achieved by a WHR in a runaway situation are listed in the table below. Any tooling, fixtures, end-effectors, etc., mounted to the user flange, outer link, or inner link of the robot must be attached by sufficient means to resist being expelled. Additionally, any payload must be held by the end-effector in a manner that prevents the payload from being accidentally expelled.

Joint Max Angular/Linear Velocity	Standard Three-Axis WHR
Joint 1 (theta axis) maximum angular velocity	600°/sec
Joint 1 (theta axis) maximum linear velocity	3.50m/sec
Joint 2 (arm axis) maximum linear velocity	1.70m/sec
Joint 3 (vertical axis) maximum linear velocity	3.80m/sec

**Table 2-1: Maximum Joint Velocities in Runaway Situations**

**NOTE:** These velocities can occur only in a runaway or mechanical failure situation. These are not performance specifications.

The safety fence or barrier constructed around the robot must be designed to withstand the impact of any item accidentally expelled from the robot.

## 2.10 RISK ASSESSMENT

The Risk Assessment for the teaching of wafer handling systems depends on the application. In many applications, the programmer will need to enter the robot workcell while servo loop is closed to teach the robot. Other applications can be designed so that the programmer does not have to enter the work envelope while Servo Loop is closed. Examples of alternative methods of programming include:

- Programming from outside the safety barrier.
- Programming with Servo Loop off (using brake release button for joint 3).
- Copying program from another (master) robot.
- Off-line or CAD programming.



**CAUTION:** Without special safeguards in its control system, a WHR could inflict serious injury on an operator working within its work envelope. Safety standards in most countries require appropriate safety equipment to be installed as part of the system. Safeguards must comply with all applicable local and national standards in the location where the robot is installed.

### 2.10.1 Exposure

Exposure of personnel to hazards related to the robot must be limited (seldom and/or short exposure time). When the servo loop is closed, all personnel must be kept out of the robot work envelope by WHR user-provided, interlocked perimeter barriers. The only permitted exception is for teaching the robot in Manual Mode by a skilled programmer, who must wear safety equipment and carry the Manual Control Pendant.



**DANGER:** System components of the WHR provide a Category 1 control system as defined by EN 954. The robot system must be installed with user-supplied interlock barriers. The interlocked barrier should interrupt the AC supply to the controller in the event of personnel attempting to enter the work cell when robot arm power is enabled, except during teaching in Manual Mode. Failure to install suitable safeguards may result in serious injury or death

### 2.10.2 Severity of Injury

Provided that skilled personnel who enter the WHR work envelope are wearing protective headgear, eyeglasses, and safety shoes, it is likely that any injuries caused by the robot would be minor (normally reversible).

### 2.10.3 Avoidance

Due to the light payload capacity of the WHR, it is likely that personnel would be able to avoid being hit even in a high-acceleration, runaway, and failure condition during normal operation..

### 2.10.4 Control System Behavior Category

The following paragraphs relate to the requirements of European (EU/EEA) directives for Machinery, Electric Safety, and Electromagnetic Compatibility (EMC).

In situations with low exposure consideration factors, European Standard EN 1050 specifies use of a Category 1 Control System per EN 954. EN 954 defines a Category 1 Control System as one that employs Category B components designed to withstand environmental influences, such as voltage, current, temperature, EMI, and well-trying safety principles. The standard WHR described in this document employs hardware components in its safety system that meet or exceed the requirements of the EU Machinery Directive and Low Voltage Directive.

This standard control system is fully hardened to all EMI influences per the EU EMC Directive and meets all functional requirements of ISO 10218 (EN 775) Manipulating Robots Safety. In addition, a software-based reduced speed and "Soft-Servo" mode has been incorporated to limit speed and impact forces on the Operator and production tooling when the robot is operated in Manual Mode.

The standard WHR meets or exceeds the requirements imposed by the EN 954 specified Category 1 level of safety.

### 2.10.5 E-Stop - CE and SEMI S2 Compliance

#### CE Compliance

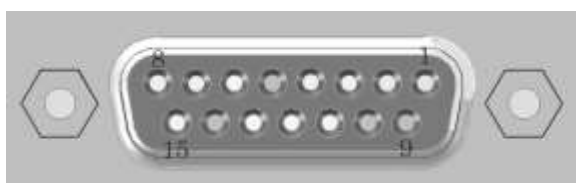
In a CE Compliant configuration, the WHR is integrated into a stand-alone cell configuration. The Emergency Stop (E-Stop) is associated with the robot system itself and other external devices connected to the CIP remote E-Stop circuitry (see Figure 2-1 and Figure 2-2). When E-Stop is actuated, high power voltage to the robot and external devices is shut off.

## SEMI S2 Compliance

When the WHR is part of a larger SEMI S2 compliant configuration, a higher level, customer-supplied E-Stop is required. This higher level E-Stop shuts off all power to the WHR. In a SEMI S2 configuration all references to E-Stop in this manual should be considered as "ROBOT STOP".

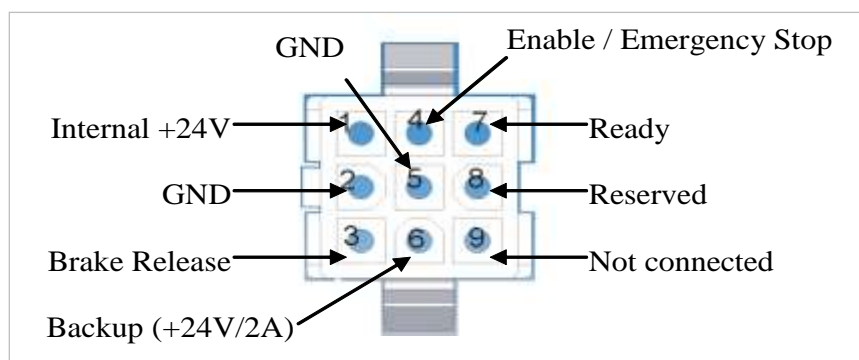
Enable/Emergency Stop signal (pin 5 on DB15, pin 4 on AMP) should be connected to internal +24V (pin 6 on DB15, pin 1 on AMP) in order to enable the system. If the signal is disconnected or connected to GND (pin 2 on DB15, pin 2 on AMP), the emergency stop circuit is activated.

To release the Z brake, the Brake Release signal (pin 1 on DB15, pin 3 on AMP) should be connected to GND (pin 2 on DB15, pin 2 on AMP) if Line Power or External Backup Power are present. If there is no Line Power or Backup power provided to the robot, the brake can be released with an external +24V power source with positive terminal connected to Backup Power (+24V/2A, pin 3 on DB15, pin 1 on AMP) and negative terminal connected to Brake Release (pin 1 on DB15, pin 3 on AMP).



1 Brake Release	5 Enable / Emg Stop
2 Ground	6 Internal +24V/0.1A
3 Backup +24V	7 OutputE (OUT 9)
4 Backup Ground	8 Ground
9, 10, 11, 12, 13, 14, 15	Spare / Reserved

**Figure 2-1: E-Stop basic circuitry (DB15 connector)**



**Figure 2-2: E-Stop basic circuitry (AMP connector)**

**NOTE:** Please refer to the specific robot documentation for details if the EStop connector is different from the one shown above.

### 2.10.6 Risks That Cannot Be Avoided

The WHR includes devices that disable Servo Loop if a system failure occurs. However, certain residual risks or improper situations can cause hazards. The following situations may result in risks that cannot be avoided:

- Failure of software or electronics that may cause high-speed robot motion in Manual Mode;
- Failure of hardware associated with the enabling device or E-Stop system.

### 2.10.7 Risks Due to Incorrect Installation or Operation

Precautions must be taken to ensure that the following situations do not occur:

- The purposeful override of any aspect of the safety E-Stop system;
- Improper installation or programming of the robot system;
- Unauthorized use of cables other than those supplied with the WHR or use of modified components in the system;

- The disabling of the interlock so that operator can enter workcell with High Power ON;
- Ejection of any work piece.

## 2.11 INTENDED USE

The installation and use of wafer handlers must comply with all safety instructions and warnings in this manual. Installation and use must also comply with all applicable local and national requirements and safety standards.

The WHR is intended for use in substrate handling and semiconductor manufacturing environments for payloads less than 0.75 kg.

The WHR must not come into contact with liquids or corrosive fluids, and is not intended for use in any of the following situations:

- Hazardous (explosive) environments;
- Mobile, portable, marine, or aircraft systems;
- Life-support systems;
- Residential installations;
- Situations where the equipment is subject to extremes of heat or humidity.



**CAUTION:** The instructions for operation, installation, and maintenance given in this Instruction Handbook must be strictly observed.

Non-intended use of a WHR can:

- Cause injury to personnel;
- Damage the prealigner or other equipment;
- Reduce system reliability and performance.

All persons that install, commission, operate, or maintain the prealigner must:

- Have the necessary qualifications/training;
- Read and follow the instructions in this User Manual exactly and completely.

If there is any doubt concerning your application's conformity with the described safety guidelines, consult your supplier to determine whether it is an intended use or not.

## 2.12 ROBOT MODIFICATIONS

It is sometimes necessary to modify the robot in order to successfully integrate it into a work cell. Unfortunately, many seemingly simple modifications can either cause a robot failure or reduce the robot's performance, reliability, or life span. The following information is provided as a guideline to modifications.



**CAUTION:** For safety reasons, certain WHR modifications are prohibited.

### 2.12.1 Acceptable Modifications

In general, the following robot modifications should not cause problems, but may affect robot performance:



- Attaching tooling, utility boxes, solenoid packs, vacuum pumps, cameras, lighting, etc., to the robot tool mount flange.
- Attaching hoses, pneumatic lines, or cables to the robot. These should be designed so they do not restrict joint motion or cause robot motion errors.

### 2.12.2 Unacceptable Modifications

The modifications listed below may damage the robot, reduce system safety and reliability, or shorten the life span of the robot. The warranty of the entire robot or certain parts may be voided.



**CAUTION:** Making any of the modifications outlined below voids the warranty of any components that the manufacturer determines were damaged due to the modification.

Contact your supplier if you are considering any of the following modifications:

- Modifying any of the robot harnesses cables
- Modifying any robot access covers or drive system components
- Modifying (including drilling or cutting) any robot surface
- Modifying any robot electrical component or printed-circuit board
- Routing additional hoses, air lines, or wires through the inside of the robot
- Making any modifications that compromise EMC performance, including shielding

### 2.12.3 Safety Requirements for Additional Equipment

Additional equipment used with the WHR (grippers, conveyor belts, etc.) must not reduce the overall work cell safeguards.

Emergency stop switches must be accessible at all times.

If the robot is to be used in an EU or EEA member country, all components in the work cell must comply with the safety requirements in the European Machine Directive 89/392/EEC (and subsequent amendments) and related harmonized European, international, and national standards. For robot systems, these include: EN 775/ISO 10218, sections 5, 6; EN 292-2; and EN 60204. For safety fences, see EN 294.

In other countries, it is strongly recommended that a similar level of safety, and compliance with applicable local and national regulations, including SEMI-S2, be maintained.

## 2.13 QUALIFICATION OF PERSONNEL

This manual assumes that all personnel have attended a training course and have a working knowledge of the system. The user must provide the necessary additional training for all personnel who will be working with the system.

As noted in this handbook, certain procedures should be performed only by skilled or instructed persons. The following standard terms are being used:

- Skilled persons have technical knowledge or sufficient experience to enable them to avoid relevant dangers, whether electrical and/or mechanical.
- Instructed persons are adequately advised or supervised by skilled persons to enable them to avoid relevant dangers, whether electrical or mechanical.

All personnel must observe industry-prescribed safety practices during the installation, operation, and testing of any electrically powered equipment. To avoid injury or damage to equipment, always remove power by disconnecting the AC power from the source before attempting any



repair or upgrade activity. Use appropriate lockout procedures to reduce the risk of power being restored by another person while an operator is working on the system.



**WARNING:** The user must receive confirmation from every entrusted individual before they do any work with the WHR that the person:

- Has received the User Manual;
- Has read the User Manual;
- Understands the User Manual;
- Will work in the manner specified by the User Manual.

## 2.14 SAFETY EQUIPMENT FOR OPERATORS

Operators must wear the following safety equipment while inside the device workcell:

- Protective eyewear aka safety glasses;
- Protective headgear (hard hats);
- Safety shoes.

Warning signs should be clearly posted around the workcell to ensure that anyone working around the system knows they must wear this safety equipment.

## 2.15 PROTECTION AGAINST UNAUTHORIZED OPERATION

The system must be protected against unauthorized use. The user or operator must restrict access to the keyboard and Teach Pendant by placing them in secure/locked storage, or through the use of another adequate method to prevent unauthorized access.

## 2.16 SAFETY ASPECTS WHILE PERFORMING MAINTENANCE

Only skilled persons with the necessary knowledge of the safety and operation of the equipment are allowed to perform maintenance on the robot, controller, and power chassis.



**CAUTION:** During maintenance and repair, the power of the WHR must be turned off. Lockout measures must be used to prevent unauthorized personnel from turning on the power.

## 2.17 WHAT TO DO IN AN EMERGENCY SITUATION

Press any E-Stop button (a red push-button on a yellow background/field) and then follow the appropriate internal emergency procedures of your company or organization. In case of fire, use CO<sub>2</sub> to extinguish.

## 2.18 ADDITIONAL SAFETY INFORMATION

The standards and regulations listed in this manual contain additional guidelines for wafer systems installation, safeguarding, maintenance, testing, start-up, and operator training. The following table lists some sources for various standards:

**BSI, British Standards Institute**

Sales Department  
Linford Wood  
Milton Keynes, MK14 6LE United Kingdom  
Phone: 0181 996 7000  
Fax: 0181 996 7001  
<http://www.bsi.org.uk>

**Beuth Verlag GmbH**

10722 Berlin  
Germany  
Phone: 030 26 01 - 22 60  
Fax: 030 26 01 - 12 60  
<http://www.din.de/en/beuth/Beuth.html>

**IEC, International Electrotechnical Commission**

Rue de Varembe 3, PO Box 131  
CH-1211 Geneva 20, Switzerland  
Phone: 41 22 919-0211  
Fax: 41 22 919-0300  
<http://www.iec.ch>

**American National Standards Institute (ANSI)**

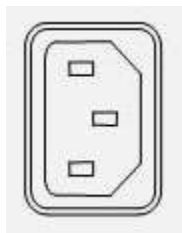
11 West 42nd Street, 13th Floor  
New York, NY 10036, USA  
Phone: 212-642-4900  
Fax: 212-398-0023  
<http://www.ansi.org>

## 3 INSTALLATION

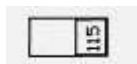
### 3.1 INTERFACE PANEL LAYOUT

An interface panel is connection point where power and vacuum sources are connected to the robot, and interfaces to host, operator, pre-aligner, and linear track are present. Various connector layouts can be implemented for different applications. Please refer to particular robot configuration sheet for interface layout.

Following components may be present on an interface panel:



IEC60320 Mains power input, 115V/230V AC, 50/60Hz



Mains voltage selector 115V, 60Hz – 230V, 50Hz



Mains power LED



Brake Release Button



Absolute encoder backup power (opt)



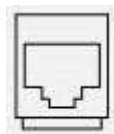
3mm quick-fit vacuum inlet, primary and secondary (on systems so equipped)



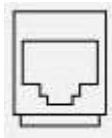
DB9 "male" serial interface connector – opto-isolated host interface



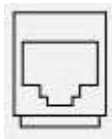
DB9 "female" serial interface connector – Stand-alone or single-axis pre-aligner interface,



RJ12 6-pin receptacle – teach pendant interface with +5V DC power



RJ45 8-pin receptacle – Daisy chained linear track interface



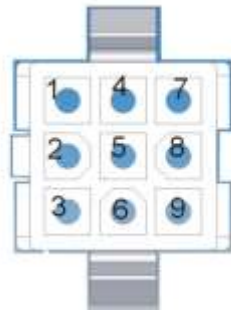
RJ45 8-pin receptacle – 10Base-T Ethernet interface



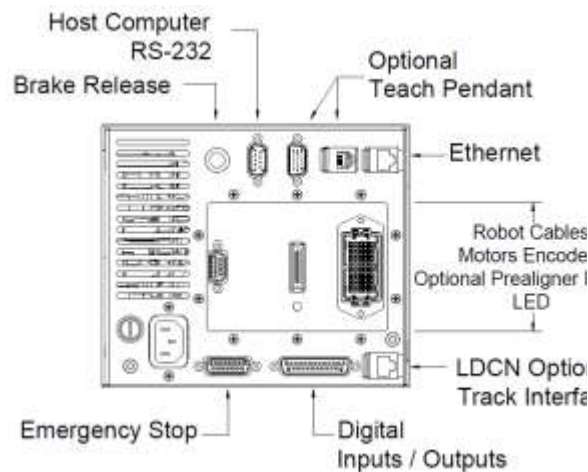
DB25 "female" Extended I/O connector



DB15 "female" Emergency Stop, Brake Release & Backup Power Connector



AMP "female" Emergency Stop, Brake Release & Backup Power Connector

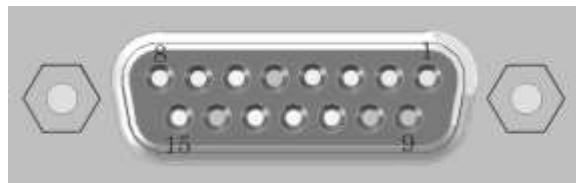


External Controller LXB42 Interface Panel for NX410 and NX420 Robots

### 3.1.1 Emergency Brake Release/Emergency Stop connection

#### DB15 Emergency Stop, Brake Release and Backup power connector

The basic components of the Emergency Stop connector are shown on figure below. Standard DB15 connector housing and pins can be used for connecting external circuit.



1 Brake Release	5 Enable / Emg Stop
2 Ground	6 Internal +24V/0.1A
3 Backup +24V	7 OutputE (OUT 9)
4 Backup Ground	8 Ground
9, 10, 11, 12, 13, 14, 15	Spare / Reserved

**Figure 3-1: Emergency Stop and Brake Release connector – DB15 Connector**

Enable/Emergency Stop signal (pin 5) should be connected to internal +24V (pin 6) in order to enable the system. If the signal is disconnected or connected to GND, the emergency stop circuit will be activated.

To release the Z brake, the Brake Release signal (pin 1) should be connected to Ground (pin 2) if Line Power or External Backup Power is present. If there is no Line Power or Backup power provided to the robot, the brake can be released with an external +24VDC power source with positive terminal connected to Backup Power (+24V/2A, pin 3) and negative terminal connected to Brake Release (pin 1).

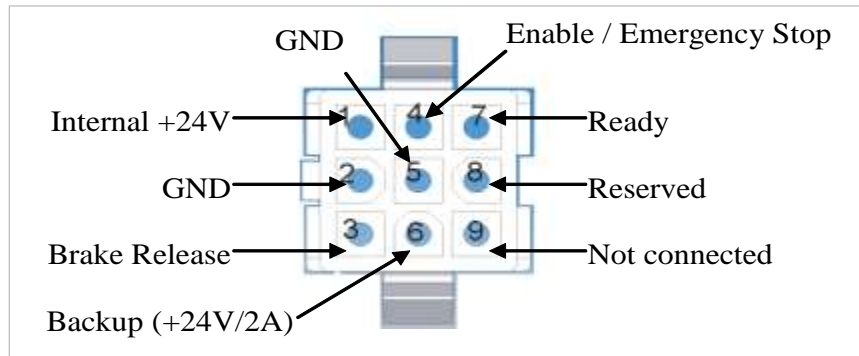
The 15-pin “Emergency Stop/Emergency Brake Release” connector can be used for external control of Z axis brake and emergency stop circuit. Possible uses are shown on following table:

	Emergency Stop only
	Brake release with AC power
	Brake release without backup DC power
	Brake release with backup DC power

**Table 3-1: Emergency Stop/Brake release (DB15)**

### AMP Emergency Stop, Brake Release and Backup power connector

The basic components of the Emergency Stop connector are shown on figure below. The mating connector can be built using AMP P/No 172169-1 (Housing) and AMP P/No 170365-1 (pins).



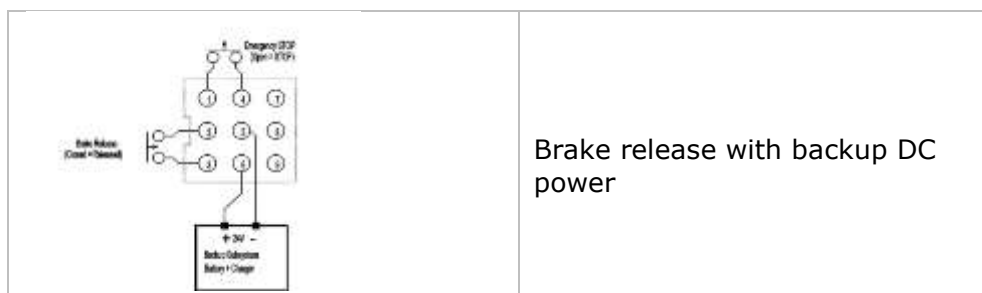
**Figure 3-2: : Emergency Stop and Brake Release connector – AMP Connector**

Enable/Emergency Stop signal (pin 4) should be connected to internal +24V (pin 1) in order to enable the system. If the signal is disconnected or connected to GND, emergency stop circuit will be activated.

To release the Z brake, the Brake Release signal (pin 3) should be connected to Ground (pin 2) if Line Power or External Backup Power is present. If there is no Line Power or Backup power provided to the robot, the brake can be released with an external +24VFC power source with positive terminal connected to Backup Power (+24V/2A, pin 6) and negative terminal connected to Brake Release (pin 3).

The 9-pin “Emergency Stop/Emergency Brake Release” connector can be used for external control of Z axis brake and emergency stop circuit. Possible uses are shown on table below.

	Emergency Stop only
	Brake release with AC power
	Brake release without backup DC power



Brake release with backup DC power

**Table 3-2: Emergency Stop/Brake release (AMP)**

### Axis Brakes and Dynamic Braking

The controller has a dynamic braking feature, which decelerates the robot in an emergency condition such as when the emergency stop circuit is open (controlled E-stop).

In addition, Z axis (and if installed, linear track) has an electromechanical brake. The brake is released when servo loop for Z axis is closed. When servo loop is open, the brake engages and holds the position of Z axis. The other two axes, Theta and R do not have mechanical brakes so these axes may be moved manually when Servo loop is open.

Under some circumstances, you may want to manually position Z axis without closing Servo loop. For such instances, an optional Brake Release button may be located on the robot connector panel. When system power is on, pressing the button releases the brake, which allows movement of Z axis. When servo loop for Axis Z is closed, brake is released automatically, and pressing this button has no effect.



**WARNING:** While the Brake Release button is depressed, Axis Z will drop by its own weight. To prevent possible injury to personnel or damage to the equipment, make sure that Axis Z is supported while releasing the brake and verify that the end-effector or other installed tooling are clear of all obstructions.

If backup DC power supply is connected, it powers low-voltage electronics and controller during AC power down. Robot firmware can be configured to shut down the controller in specified amount of time in order to keep backup DC power from extra depletion. Look in "Command Reference" for description of `_BB` command.

#### 3.1.2 Using Power Disconnect Instead of Emergency Stop

In some cases, power to the robot is disconnected by external means when an emergency stop is requested. In that case, using the battery backup option is recommended in order to avoid damage to the controller.



**WARNING:** Emergency power cut-off without battery backup option can damage controller, requiring authorized service.

#### 3.1.3 Serial Communication Ports

DB9 and RJ12 style connectors are used for the connection to host, slave prealigner, and teach pendant hand-held terminal.

RS-232 Host interface port is DTE-style DB9 "male" connector, with following pin-out:



Pin No	Description
1	N/C – Not connected
2	RxD – Received Data
3	TxD – Transmitted Data
4	N/C – Not connected
5	GND - Signal Ground
6	N/C – Not connected
7	N/C – Not connected
8	N/C – Not connected
9	N/C – Not connected

RS-232 slave interface port is DCE-style DB9 “female” connector, with following pin-out:



Pin No	Description
1	N/C – Not connected
2	TxD – Transmitted Data
3	RxD – Received Data
4	N/C – Not connected
5	GND - Signal Ground
6	N/C – Not connected
7	N/C – Not connected
8	N/C – Not connected
9	N/C – Not connected

### 3.1.4 Serial Communication Cables

The Simple three-wire null-modem cable can be used for host-to-robot communication. Assuming most modern PCs are equipped with a 9-pin male connector, following the cable is enough:

PC end (DB9-F)			Robot end (DB9-F)	
2	RxD	↘	RxD	2
3	TxD	↗	TxD	3
5	Ground	—	Ground	5

The Simple three-wire straight cable can be used for robot-to-prealigner communication, as follows:

Robot end (DB9-M)			Prealigner (DB9-F)	
2	TxD	—	RxD	2
3	RxD	—	TxD	3
5	Ground	—	Ground	5

When a prealigner is connected as a slave device to the robot, corresponding settings for the serial ports have to match. In order to obtain best performance, it is recommended to set both the robot slave port and the prealigner host port at 115200bps. See Chapter 7.3.1 Communication Parameters File (Mcl5.ini), and Document #710000011 "LPA Series 1E Single-Axis Aligners", for configuration details.



### 3.1.5 Teach Pendant Communication Cable

Since the teach pendant needs to be powered in order to operate, +5V DC voltage is present on its interface connector. The current teach pendant uses the following cross-over RJ12 to RJ12 cable:

WHR side (RJ12)			Terminal (RJ12)	
1	Ground	—————	Ground	6
2	RxD	—————	TxD	5
3	TxD	—————	RxD	4
6	+5VDC	—————	+5VDC	1

### 3.1.6 External I/O Module or Linear Track RJ45 Connector

This port can be used to connect an external I/O module, or external Linear Track module supporting LDCN communication protocol. These modules are connected using straight 8-wire RJ45-RJ45 cable, as follows:

WHR side (RJ45)			External LDCN Device	
1	N/C	—————	+5VDC	1
2	Ground	—————	Ground	2
3	+TX Data	—————	+TX Data	3
4	-TX Data	—————	-TX Data	4
5	-RX Data	—————	-RX Data	5
6	+RX Data	—————	+RX Data	6
7	-A Out	—————	-A In	7
8	+A Out	—————	+ A In	8

For more information on the External I/O module, or the Linear Track module please refer to these modules' documentation, available upon request.

### 3.1.7 Ethernet RJ45 Connector

This port can be used to connect to 10BaseT/10Base100 Ethernet network. The connection is using standard Cat5/Cat5e Ethernet cable, as follows:

WHR side (RJ45)			Remote Device (RJ45)	
1	+TX Data	—————	-RX Data	1
2	-TX Data	—————	+RX Data	2
3	+RX Data	—————	-TX Data	3
4	N/C			4
5	N/C			5
6	-RX Data	—————	+TX Data	6
7	N/C			7
8	N/C			8

### 3.1.8 Extended I/O Interface Connector

Eight additional digital inputs and outputs are available on the DB25 connector, as follows:




Pin No	Signal	Pin No	Signal
--------	--------	--------	--------

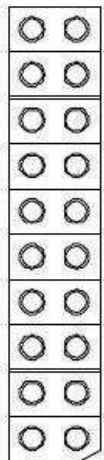
1	Ground	14	Ground
2	InputE	15	OutputE
3	InputF	16	OutputF
4	InputG	17	OutputG
5	InputH	18	OutputH
6	InputJ	19	OutputJ
7	InputK	20	OutputK
8	InputL	21	OutputL
9	InputM	22	OutputM
10	InputN	23	OutputN
11	InputR	24	Ground
12	PullUp/24VDC	25	PullUp/24VDC
13	Reserved		

**NOTE:** Pins 12 and 25 can provide a total of 0.2A current.

### 3.1.9 Sensor Type Configuration

WHR controller provides two jumper blocks for selecting of sensor types (PNP or NPN) connected to controller' input lines, as follows:

Jumper Block J7 – Arm Sensors Configuration (next to Arm Interface Connector, CN16)			
	Sensor D	ON: Pull Up/NPN (default)	Off: Pull Down/PNP
	Sensor C	ON: Pull Up/NPN (default)	Off: Pull Down/PNP
	Sensor B	ON: Pull Up/NPN (default)	Off: Pull Down/PNP
	Sensor A	ON: Pull Up/NPN (default)	Off: Pull Down/PNP
	Home R	ON: Pull Up/NPN (default)	Off: Pull Down/PNP

Jumper Block J10 – Extended I/O Configuration (next to Extended Interface Connector, CN23)			
	Input E	ON: Pull Up/NPN	Off: Pull Down/PNP (default)
	Input F	ON: Pull Up/NPN	Off: Pull Down/PNP (default)
	Input G	ON: Pull Up/NPN	Off: Pull Down/PNP (default)
	Input H	ON: Pull Up/NPN	Off: Pull Down/PNP (default)
	Input J	ON: Pull Up/NPN	Off: Pull Down/PNP (default)
	Input K	ON: Pull Up/NPN	Off: Pull Down/PNP (default)
	Input L	ON: Pull Up/NPN	Off: Pull Down/PNP (default)
	Input M	ON: Pull Up/NPN	Off: Pull Down/PNP (default)
	Input N	ON: Pull Up/NPN	Off: Pull Down/PNP (default)
	Input R	ON: Pull Up/NPN	Off: Pull Down/PNP (default)

## 3.2 END-EFFECTOR VACUUM LINES

The single-arm robot is delivered with a solenoid valve and internal vacuum sensor for a single vacuum end effector. Two 3mm quick connect fittings are located on the robots interface panel. For standard configurations use the top quick connect fitting. Insert a 3mm air line into the vacuum fitting on the connector panel of the robot.



**CAUTION:** Do not apply a positive pressure to either of the vacuum ports. This could cause damage to the robot.

A factory installed dual valve option is available. For the dual valve option the quick connect fittings are used. The upper fitting is dedicated to the first end-effector and the lower fitting is for the second end-effector.

The dual-arm robot is delivered as standard with dual solenoid valves and internal vacuum sensors, one per each paddle. Two 3mm quick connect fittings are located on the robot interface panel. For standard configurations use the top quick connect fitting. Insert a 3mm air line into the vacuum fitting on the connector panel of the robot.

Vacuum sensors' sensitivity can be adjusted by the following procedure:

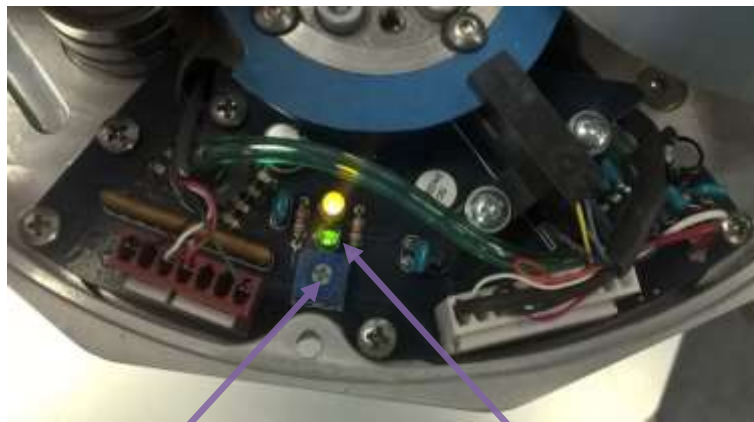
- On dual-arm robots, lift the Z axis by about 5" to facilitate access to retaining screws.  
**MVA Z 5000**
- Turn servo loop off:  
**SOF**
- On single-arm robot: Remove the cover of the lower arm link by removing the screws that attach the cover to the arm.
- On dual-arm robot: Remove the covers on the octa-box by removing the screw attaching the covers from underneath.
- Connect vacuum line(s) and install end effector paddle(s)
- Open vacuum valve 1 or 2:  
**VON 0**  
or  
**VON 1**
- With no wafer placed on the end effector, turn the potentiometer located on the vacuum sensor slowly counter-clockwise until the corresponding LED turns on. Mark this position as A.
- With a wafer placed on the end effector, turn the potentiometer clockwise until the corresponding LED turns off. Mark this position as B.
- Turn the potentiometer midway between positions A and B
- Repeat the procedure for the other end effector, if such is installed.
- Replace the cover(s).

By customer request, external flange-mounted vacuum sensors with indicators can be installed. Please refer to sensor' documentation for details about setting up sensitivity.



Potentiometers 1 and LEDs 1 and 2

**Figure 3-3: Single-arm robot : Dual vacuum line control board in lower arm link**



Potentiometer

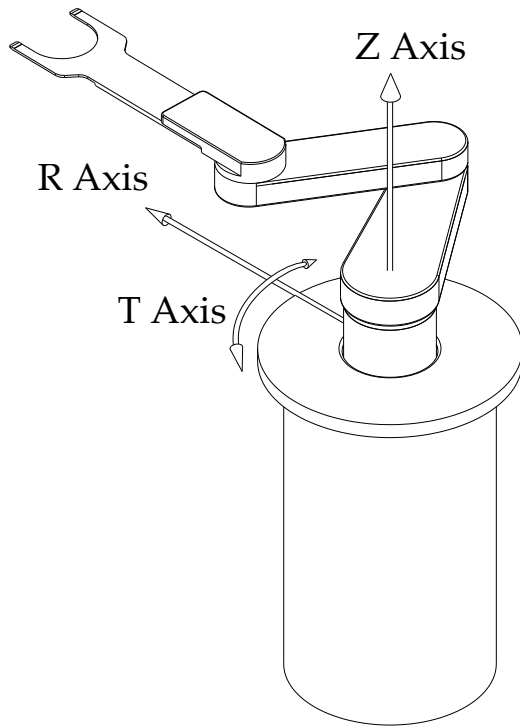
Sensor LED

**Figure 3-4: Dual-arm robot: Single vacuum line in octa-box, for each arm**

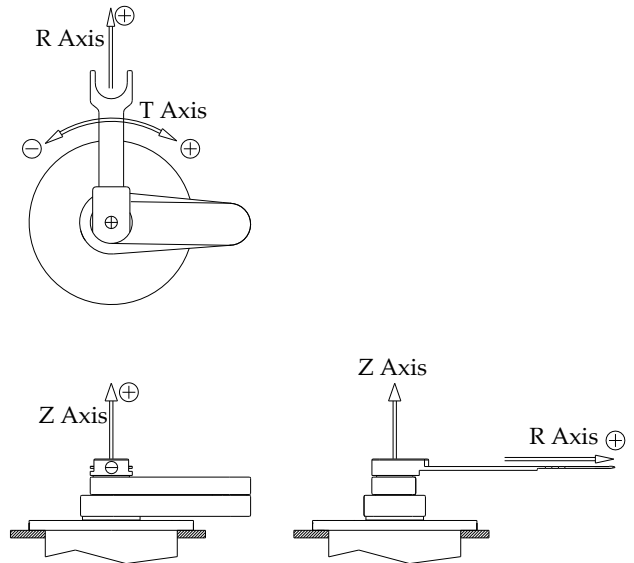
## 4 COORDINATE SYSTEMS AND UNITS OF MEASUREMENT

### 4.1 COORDINATE SYSTEM USED IN SINGLE-ARM WAFER HANDLING ROBOT

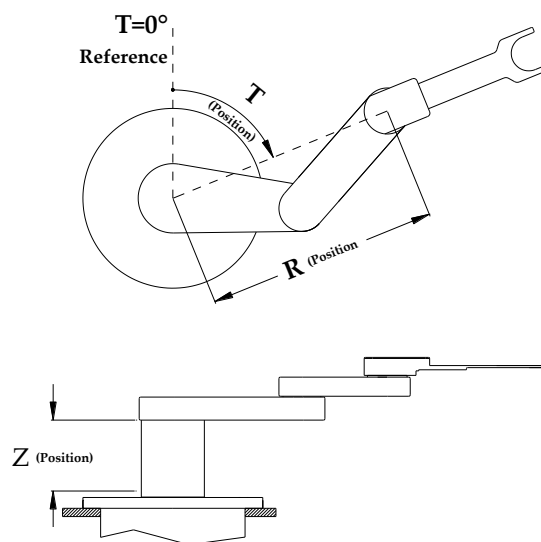
Wafer handling robots use three-axis cylindrical coordinate system. Single-arm robot axes names and orientation are shown on the following figures.



**Figure 4-1: Three-axes Robot Coordinate System**



**Figure 4-2: Arm at initial (home) position ( $T=0$ ,  $R=0$ ,  $Z=0$ )**



**Figure 4-3: Robot Arm coordinates at arbitrary position ( $T>0$ ,  $R>0$ ,  $Z>0$ )**

All measurements are made with respect to a point in the center of the joint between robot arm and end effector. Any end effector mounted should be taken into account when measuring station positions.

The directions and units of measurement are defined as follows:

- T axis: The unit of angular measurement for T axis is one hundredth of an angular degree (that is, 1 unit of T axis equals  $0.01^\circ$ ), and one full revolution of T axis (or  $360^\circ$  would measure 36000 units). The value of the measurements (coordinate) is the angle formed by the reference point, the center of the robot and the end-effector to arm joint. This is illustrated in the figures above. The T coordinate increases when the arm is rotated clockwise.
- R axis: The unit of linear measurement for R axis is one thousand of an inch (that is, 1 unit of R axis equals 0.001", and one inch would measure 1000 units). The value of the measurement is the horizontal distance between the center of the robot flange (reference point) and the center of the joint between the top link and the end-effector. The Zero point of R axis is located above geometric center of robot' flange, when the arm links are in overlapped position (see figures above). R coordinate increases according to the "right hand rule":

Bottom link of the robot arm is the shoulder, top link is the arm, position increases when the arm is extended (top link of robot arm turns clockwise with respect to bottom arm).

- Z-axis: Unit of linear measurement is one thousandth of an inch (that is, 1 unit of Z axis equals 0.001", and one inch would measure 1000 units). Measurement is the vertical distance between the reference point and zero position. Zero position is located above mounting plate (or bottom plane of robot top plate). Z coordinate increases as robot arm is moved upward (away from robot body). Linear speed units are defined as one hundredth inch per second ( $0.01''/\text{sec}$ ), and linear acceleration units are defined as one hundreds of the inch per second per second ( $0.01''/\text{sec}^2$ ).

Rotational speed units are defined as one tenth angular degrees per second ( $0.1^\circ/\text{sec}$ ), and angular acceleration units are defined as one tenth angular degree per second per second ( $0.1^\circ/\text{sec}^2$ ).

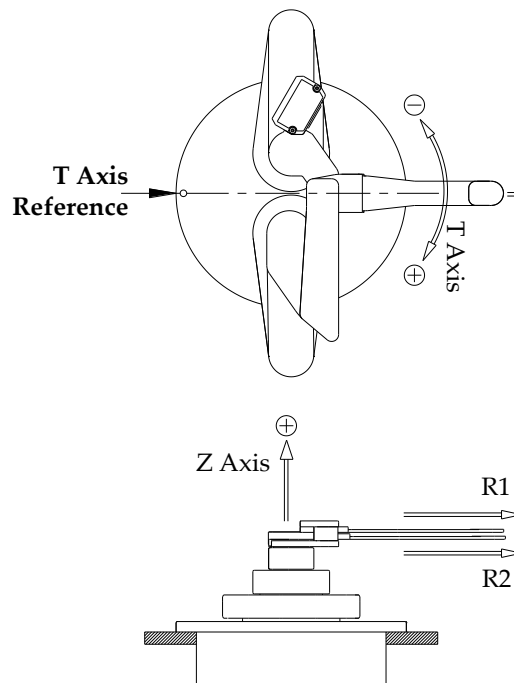
Robot working envelope (maximum position reached) is defined as a cylinder with following coordinates relative to factory zero position:

T:  $-210.00^\circ \dots 210.00^\circ$  (max  $-240.00^\circ \dots 240.00^\circ$ )  
 R:  $-14.400'' \dots 14.400''$  (depending of particular model)  
 Z:  $-0.500'' \dots 12.200''$  (depending of particular model)

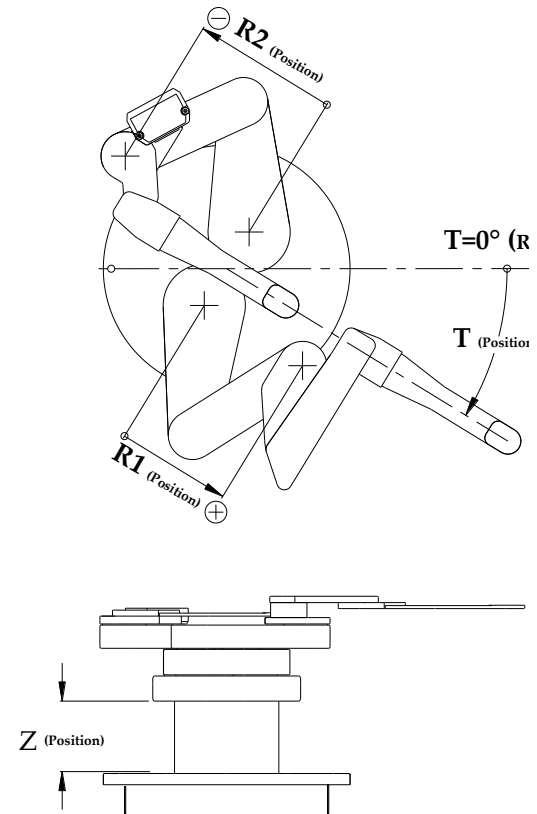
The R coordinate has to be adjusted by the length of installed end effector (and other tooling) when evaluating working envelope of installed robot.

## 4.2 COORDINATE SYSTEM USED IN DUAL ARM WAFER HANDLING ROBOT

Wafer handling robots use three-axis cylindrical coordinate system, with axes names and orientation as shown on the following figures.



**Figure 4-4: Dual-arm robot Coordinate System and initial (home) positions**

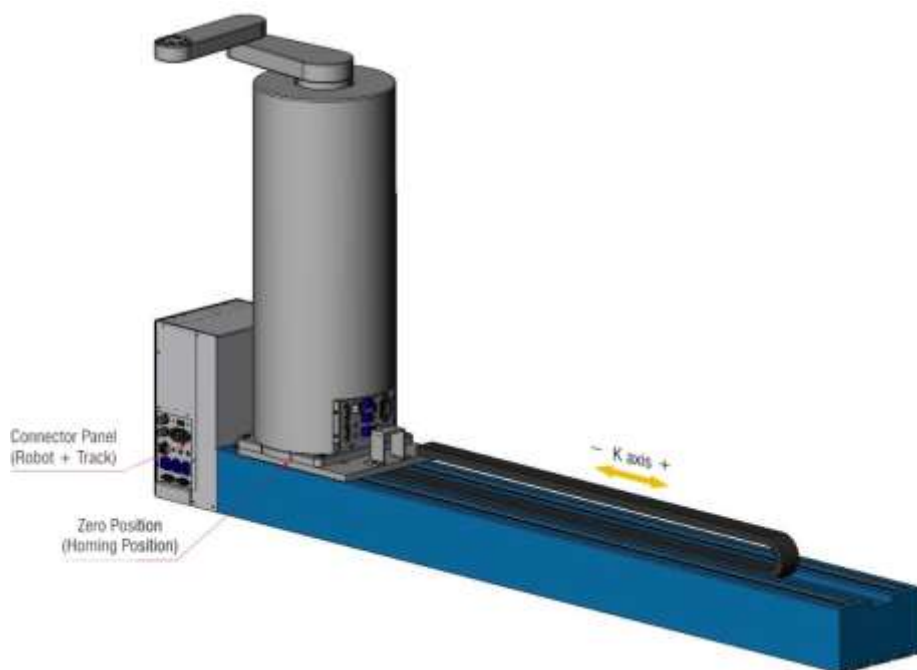


**Figure 4-5: Dual-arm robot at arbitrary position ( $T > 0$ ,  $R1 > 0$ ,  $R2 < 0$ ,  $Z > 0$ )**

## 4.3 COORDINATE SYSTEM USED FOR LINEAR TRACK-BASED SYSTEMS

As an option, the robot can be mounted on a linear track axis, either hung on its top flange, or mounted on its bottom flange. Depending of particular configuration, the track axis can have various lengths. Throughout this document, this axis will be referred to as K axis.

Unit of linear measurement is one thousandth of an inch (that is, 1 unit of K axis equals 0.001", and one inch would measure 1000 units). Measurement is the horizontal distance between the reference point and zero position. The zero position is typically at the track end closer to the controller/motor box, and increases as the robot moves away from this point.



**Figure 4-6: Generic Coordinate System for track-mounted robot**

#### 4.4 COORDINATE SYSTEM USED IN SINGLE-AXIS WAFER PREALIGNERS

Single-axis wafer prealigner has a rotating chuck only, and this axis (C-Axis) is fixed to the prealigner top plate.

The measurement unit is one-tenth of an angular degree (that is, 1 unit of C axis equals  $0.1^\circ$ , and one full revolution of C axis or  $360^\circ$  would measure 3600 units). The C coordinate increases when the chuck is rotated clockwise (looking from the top).

Rotational speed units are defined as one-tenth of an angular degree per second ( $0.1^\circ/\text{sec}$ ), and angular acceleration units are defined as one-tenth of an angular degree per second per second ( $0.1^\circ/\text{sec}^2$ ).



**Figure 4-7: Single-axes Prealigner Coordinate System**

The single-axis wafer prealigner is a measuring device working in union with the robot. Based on the prealigner measurement the robot controller calculates the offset of the wafer center with respect to the chuck center and commands a final C-axis to move at a position. This allows the robot to get a proper center offset compensation and a correct wafer notch/flat compensation. To do so, there are two parameters related to the position of the prealigner with respect to the robot – the distance between the prealigner chuck and the robot center and the angle between the line connecting the prealigner chuck with the measuring sensor and the line connecting the prealigner chuck and the robot center. The positive direction of measuring is clockwise from the chuck - sensor line to the chuck-robot line.

See commands **PRD** (Prealigner to Robot Distance) and **PRA** (Prealigner to Robot Angle) for measuring units.



**NOTE:** The prealigner has different positions of measuring sensors depending on the wafer size, and aligner type. For example, on LPA-312 single-axis aligner, the CCD sensors for wafer diameters 75mm(3"), 100mm(4"), 200mm(8") and 300mm(12") are mounted in parallel orientation with the prealigner light tower. The CCD sensor for the wafer diameter 125mm (5") and 150mm (6") is mounted with an offset of 14.4 degrees with the prealigner light house (see the picture below). Since the angle of the CCD sensor and the light tower is accounted by a prealigner calibration parameter, the robot to prealigner angle (PRA parameter) is always measured with respect to the light tower. Please refer to Document #7100000011, LPA Series 1E Single-Axis Prealigners, for additional information.

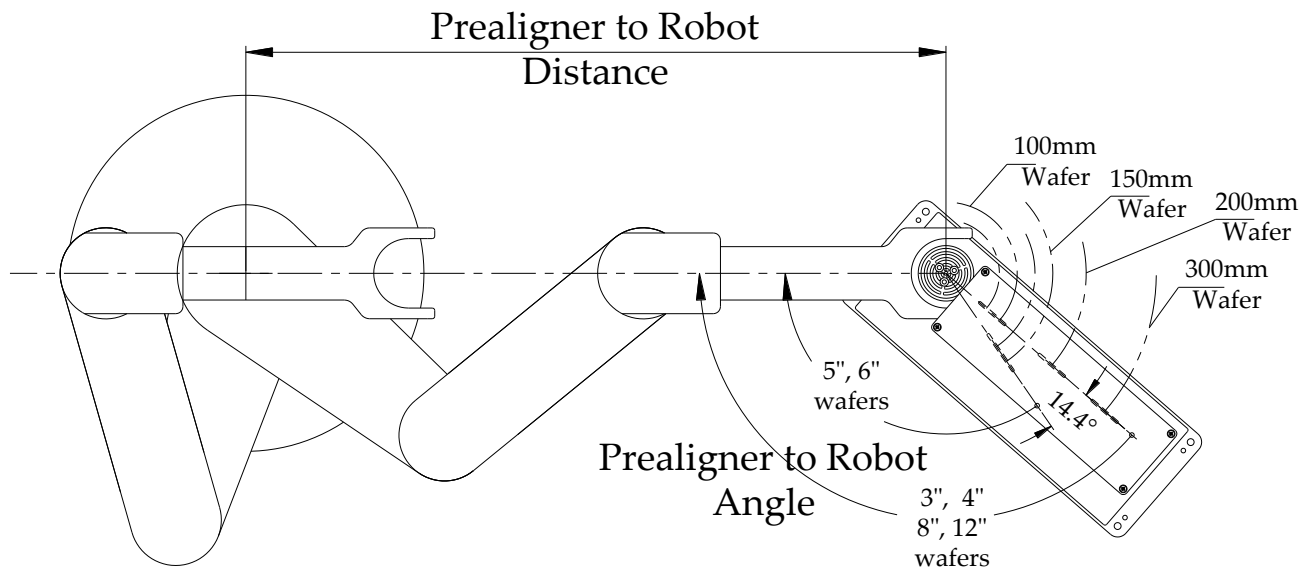
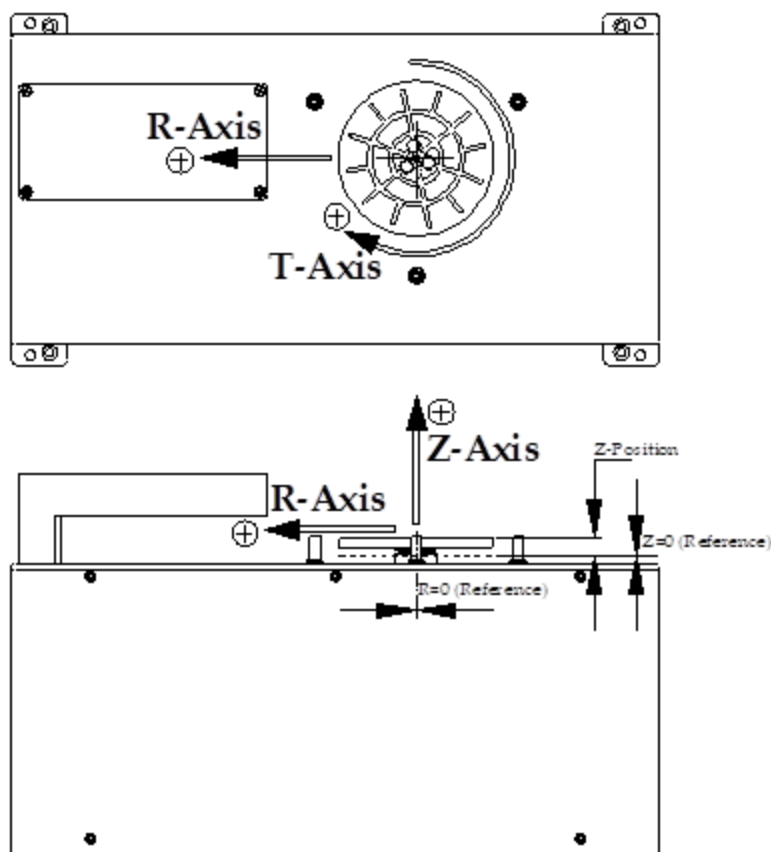


Figure 4-8: Single-axes Prealigner to Robot – positioning parameter

#### 4.5 COORDINATE SYSTEM USED IN THREE-AXIS WAFER PREALIGNERS

Three-axis wafer prealigner is a stand-alone device, and uses three moving axes: rotating chuck axis (T), linear horizontal axis (R) and linear vertical axis (Z) – see below. The Linear axes move the chuck along the length or prealigner, and in an up and down direction. The Three-axis prealigner, when used as a stand-alone device, uses upper-case axis names T, R and Z. When used as a slave to the robot, axis names are upper-case C, X and V.



**Figure 4-9: Three-axes Prealigner Coordinate System**

All measurements are made against the center point of the opening on the top plate of the prealigner. Coordinate system axes, directions and measurement units are defined as follows:

- **T axis (chuck rotational motion):** The measurement unit is one-tenth of an angular degree (that is, 1 unit of T axis equals  $0.1^\circ$ , and one full revolution of T axis or  $360^\circ$  would measure 3600 units). The T coordinate increases when the chuck is rotated clockwise (looking from the top).
- **R axis (chuck linear motion):** The measurement unit of linear motion is one ten-thousandth of an inch (that is, 1 unit of the R axis equals  $0.0001''$ , and one inch would measure 10000 units). The value of the measurement is the horizontal distance between the center of the chuck and the zero position (reference point). The Zero point is factory calibrated and it is located above the geometric center of the opening on the top plate. The R coordinate increases as the chuck is moving toward the lighthouse.
- **Z axis (chuck vertical motion):** The measurement unit of linear motion is one ten-thousandth of an inch (that is, 1 unit of Z axis equals  $0.0001''$ , and one inch would measure 10000 units). The value of the measurement is the vertical distance between the surface of the chuck and the zero position (reference point). The Zero position is factory calibrated and located above prealigner top plate. The Z coordinate increases as the chuck is moving upward.

Linear speed units are defined as one ten-thousandth inch per second ( $0.0001''/\text{sec}$ ), and linear acceleration units are defined as one ten-thousandth of the inch per second per second ( $0.0001''/\text{sec}^2$ ).

Rotational speed units are defined as one-tenth angular degree per second ( $0.1^\circ/\text{sec}$ ), and angular acceleration units are defined as one-tenth angular degree per second per second ( $0.1^\circ/\text{sec}^2$ ).

## 4.6 COORDINATE SYSTEM USED IN CASSETTE ELEVATORS

Vacuum cassette elevators have a single vertical axis, named V1 (for first or only elevator), and V2 (for second elevator if installed).

The Unit of linear measurement is one thousandth of an inch (that is, 1 unit of V axis equals 0.001", and one inch would measure 1000 units). The Measurement is the vertical distance between the reference point and the zero position. The Zero position is located above the mounting plate (or bottom plane of elevator top plate). The V coordinate increases as the elevator axis arm is moved upward (away from elevator body).

## 4.7 LIMITING JOINT TRAVEL

The joint motion or travel is limited by both software and hardware limits. The programmable software limits are known as Software Limits; the fixed hardware limits are referred to as hard stops.

Software Limits are used when the normal motion range of the robot must be limited (if other equipment is installed inside the envelope, for example). The Software Limits for each axis are set to their default value at the factory. To limit any axis's motion range, change the axis's software limits value using the appropriate commands (**\_FL** and **\_RL**).

Software limits are enforced only after the homing procedure is complete. Since **\_FL** and **\_RL** commands report currently set limits, these commands will report zeros unless homing is complete.

## 5 COMMAND TYPES AND CONVENTIONS

### 5.1 GENERAL CONSIDERATIONS

Robots and stand-alone prealigners accept commands over a host interface serial port. Usually, this port is configured at 9600 bps, 1 start bit, 1 stop bit, no parity. Communication speed can be changed if necessary. Each command sent is confirmed back to the host, and no new commands are accepted until a prompt has been displayed back to the host. If a new command is sent to the robot or prealigner while previous one has not been processed, the host will receive a Bell character (**ASCII code 7, 0x07**).

Robots that are more equipped will have two or more interfaces present, the Teach Pendant and Ethernet. Commands can be issued from either HOST or Teach Pendant interface, but not both. When Teach Pendant is active, either as a terminal, or as a teaching tool, host commands are ignored, and status BADA is returned.

Commands can be issued also over the Telnet link, with responses returned to the Telnet client. Detailed explanation of the Teach Pendant and the Telnet interfaces are given at the end of this chapter.

### 5.2 COMMAND SYNTAX, PROMPTS AND RESULTS

Each command string consist of a command name (one to sixteen characters), and parameters zero to sixteen, followed by a carriage return character (**ASCII code 13, 0x0D**). The Command name and parameters are separated by at least one space character (**ASCII code 32**), or comma character (**ASCII code 44**). Responses consist of zero or more characters terminated by a carriage return and line feed (**ASCII code 10, 0x0A**) characters.

No difference is made in the capitalization of commands or parameters, with the exception of station names: both lower case ('a' – 'z') and upper case ('A' – 'Z') letters are allowed, and they refer to different stations. For example, all of following commands would result from moving the Z axis to position 1000:

```
>mva z 1000
>MVA z 1000
>mVa Z 1000
```

Each command submitted returns immediately with either ">" or with "?" prompt. If the command takes longer to execute (e.g. motion, homing, pick/place command etc.), then it may signal the command has finished its execution with the proper setting of the **INF** command. When the Info Mode is set to zero (**INF 0**), the command will not return any status unless asked specifically with the **STA** command.

Two different prompts indicate that the system is ready to accept commands:

- >     The greater-than sign (**ASCII 62**) indicates that the previous command finished without errors.
- ?     The question mark (**ASCII 63**), indicates that the previous command was either not executed at all (e.g. syntax error, unrecognized command etc), or finished with an error condition.

An error prompt "?" could mean that:

- The command name, axis name, or station name are not recognized:

```
>STS
?STA
0008
>CPO H
?
>CPO T
12000
```

```
>NSL A
25
>NSL x
?
```

- The command is not allowed in current state of the robot (e.g. if robot is not homed, no GET and PUT commands are allowed)

```
>INF 1
>STA
0400
>GET a
?SON
>HOM
.....
>0000
>WSA
>GET a 1
.....
>000C
```

- The command is not allowed in current state of the robot (e.g. GET command is executed with wafer already on the paddle)

```
>STA
100C
>GET a 1
?
```

- The error prompt itself will be displayed also if critical robot condition occurs (e.g. servo loop is broken during motion, or emergency stop is activated, or software limit is violated)

Empty command (e.g. spaces only or just Enter) will return a positive prompt

```
? ↵
>
```

If a command is listed as accepting optional parameters, these parameters can be omitted starting with the last one. For example, the **SPD** command can take up to two optional parameters, axis name and value. If only one parameter is specified, this is axis name, not value.

### 5.3 COMMAND EXECUTION AND REPORTING

Commands that return information about single axis return info in one of the following formats (depending of **INF** setting):

- **INF 0,1,2:** A number corresponding to a particular axis parameter, followed by the Carriage Return Line Feed (**ASCII 13, 10**) like in

```
>CPO T<cr>
12000<cr><lf>
>
```

- **INF 3:** Axis name in upper case followed by Equal sign (**ASCII 61**) followed by a number for that axis followed by Carriage Return Line Feed (**ASCII 13, 10**).

```
>CPO Z<cr>
Z=500<cr><lf>
>
```

- **INF 4:** The command name, colon, a number corresponding to a particular axis parameter, followed by the Carriage Return Line Feed (**ASCII 13, 10**) like in

```
>CPO T<cr>
CPO:12000<cr><lf>
>
```

- **INF 5:** The command name, colon, error indicator, space character, system status, space, a number corresponding to a particular axis parameter, followed by the Carriage Return Line Feed (**ASCII 13, 10**) like in

```
>CPO T<cr>
CPO:0,100C 12000<cr><lf>
>
```

Commands that return information about all the axes return info in either of the following formats (depending of **INF** setting):

- **INF 0,1,2:** Three comma-delimited numbers, each number corresponds to each of T, R and Z-axes, followed by Carriage Return Line Feed (**ASCII 13, 10**) like in

```
>CPO<cr>
12000,-6000,500<cr><lf>
>
```

- **INF 3:** Axis names in upper case followed by Equal sign (**ASCII 61**) followed by number for that axis followed by Carriage Return Line Feed (**ASCII 13, 10**). Values for different axes are separated by a space, like in

```
>CPO<cr>
T=12000•R=-6000•Z=500<cr><lf>
>
```

- **INF 4:** The command name, colon, three comma-delimited numbers, each number corresponds to each of T, R and Z-axes, followed by Carriage Return Line Feed (**ASCII 13, 10**) like in

```
>CPO<cr>
CPO:12000,-6000,500<cr><lf>
>
```

- **INF 5:** The command name, colon, , error indicator, space character, system status, space, three comma-delimited numbers, each number corresponds to each of T, R and Z-axes, followed by Carriage Return Line Feed (**ASCII 13, 10**) like in

```
>CPO<cr>
CPO:0,100C 12000,-6000,500<cr><lf>
>
```

Non-axis parameters are reported in similar way but without axis names.

**Note:** "Comma-delimited" will use comma character as defined in system parameters file.

When the **INF** mode is a non-zero value, commands taking some time to execute (e.g. any command involving a motion) will report the current status after they finish their execution. For example, **HOM** command will report current status when the system finishes homing procedure:

```
>STA
0041
>HOM
>
... .. (some time after)
0000
>
```

### 5.3.1 Echo-and-Status Communication Protocol

Robot firmware 2.4.7.0 and newer (for single-arm robots), and 2.4.8.0 (for dual-arm robots) support an additional mode of operation where a command name is echoed back, and (optionally) followed by an error flag and robot status. This is activated by setting the **INF** mode to 4 or 5.

- When the **INF** mode is set to 4, a command name followed by a colon is returned before any results returned by the command.

```
>INF 4
INF:
>SPD
SPD: 801,1802,100
>GET A 1
```

```
>...
GET:000C
>
```

The above example shows results from a command setting a parameter (**INF** 4), command returning results (**SPD**), and motion command (**GET**).

- When **INF** mode is set to 5 the command name followed by a colon, error code, and current status, is returned before any results returned by the command.

```
>INF 5
INF:0,1000
>SPD
SPD:0,1000 801,1802,100
>HOM
...
>HOM:0,1000
>PUT B 1
PUT:1,1002
?EERR
EERR:528,1000
>
```

The above example shows results from a command setting a parameter (**INF** 5), command returning results (**SPD**), and motion commands (**HOM** and **PUT**). The first three commands return 0 as an error flag (indicating no error), while **PUT** command' error flag is set to 1. The **EERR** command reports the last error code (528). Full list of error codes reported by the **EERR** command is available in 6.5 Error codes returned by **EERR**.

Note: Syntax and parameter type errors are not reported by the **EERR** command.

## 5.4 HOST INTERFACE COMMANDS AND RESPONSES LOGGING

Current firmware versions allow for limited amount of host interface communication (commands received by the robot, and responses sent back) to be logged by the robot firmware. Refer to 7.3.7 Host communication logging parameters chapter for details about configuring and using.

## 5.5 PARAMETER TYPES

Commands can be supplied with several different parameter types, as follows:

### Axis

Most of the motion and parameter setting commands take an axis name as a parameter. The axes names are specified in the robot configuration file and can be any alphanumeric sequence starting with a letter. The WHR axis names are single character, mainly T, R and Z. The robot documentation will follow those names for the three motion axes. Axis names are case-insensitive. For example, T and t refers to the same T axis of WHR series robot.

Dual-arm wafer handlers use two-character axis names for arms, as follows:

R1 is the primary arm, R2 is the secondary arm

### Position

Position is a parameter relating to the absolute coordinate of an axis. The position could be expressed in linear or angular measurement units depending on the axis. The available position range is defined by the limit settings of the axis.

### Distance

Distance is a parameter relating to the relative coordinate of an axis like distance to travel. The measurement units and available values are determined by the axis referred in the command, as well as, the position parameter.

### Station

Stations are fixed defined places within a robot envelope where some processing is done. Stations are identified by station names. The station name can be any single lowercase character from a to z or any uppercase character from A to Z. This makes for a total of 52

different stations. When used as a parameter of a command, the station name must refer to one that has an already defined stations. The stations names are case-sensitive.

#### Slot

All commands relating to wafer transfers to / from stations take a slot number as parameters. A slot number can be any positive integer up to the maximum number of slots for selected stations.

#### Value

The parameter value is used as a placeholder for any numerical parameter that does not fall in the above categories. The available range depends on each command.

#### Option

Option is a parameter that can be one of a specified set of values. They are specific for each command, and are listed in the command description. Often, the option parameter can be omitted.

## 5.6 CUSTOM MACRO COMMANDS

Upon customer request, some WHR's are delivered with a custom command set, emulating legacy host communication protocol, or implementing customer-requested command set. In those cases, it is possible that the custom command name overlaps with the standard command as listed in this manual. In order to have both commands available, the following rules should be applied:

- If a command name is present in the custom command set, it has precedence over the standard command.
- If a command name is present in the custom command set, but the standard command is required, then the standard command name must have an underscore character, "\_" appended immediately after command name.

Let's take a custom command set where the **sof** command is present, and it performs a custom operation. The **sof** command is also present in the standard command set. There's also a custom **sta** command, providing custom status to the host application.

```
>STA
S:1000
>sof x 12345
S:OK
>
```

The above examples execute two custom commands – custom status **sta**, and custom **sof** command with two parameters: character 'x' and number 12345

```
>STA_
0000
>sof_
>STA_
0401
>
```

The above examples execute the standard commands, providing status, and switching off servo loop.

## 5.7 COMMAND GROUPS

Commands implemented in the robot firmware can be divided in following groups:

### 5.7.1 Informational commands

Commands in this group provide generic information about the firmware, and hardware running on the robot.

<b>_DT</b>	Set system date and time
<b>_SI</b>	Report system information



<b>RES</b>	Reset the robot controller
<b>SNR</b>	Report serial number
<b>VER</b>	Report firmware version number

### 5.7.2 Housekeeping (Parameters save / restore) commands

Following commands are tasked with proper saving and restoring of various operational parameters of the robot. Refer to corresponding commands for list of parameters affected. See Chapter 7.3, Robot Configuration and Firmware files, for files description.

<b>_CLD</b>	Restore Custom Parameters
<b>_CSV</b>	Save Custom Parameters
<b>RMP</b>	Restore robot's Motion and Calibration Parameters
<b>RPS</b>	Restore Parameters Sets
<b>RSP</b>	Restore station parameters
<b>SMP</b>	Save robot's motion and calibration parameters
<b>SPS</b>	Save Parameters Sets
<b>RPP</b>	Restore Calibration Parameters for single-axis prealigner
<b>SPP</b>	Save Calibration Parameters for single-axis prealigner
<b>SSP</b>	Save station parameters

The interlock mechanisms prevent execution of save and restore commands while the system is executing motion commands.

### 5.7.3 Status and diagnostic commands

Following commands are used to report current or previous state of the robot.

<b>__CS</b>	Report current robot state
<b>_GETTEMP</b>	Report current controller temperature
<b>_GETVOLT</b>	Report current controller motor voltage
<b>CPO</b>	Report axis current position
<b>DIAG</b>	Report last error information
<b>EERR</b>	Report extended error information
<b>ESTA</b>	Report enhanced system status
<b>EST</b>	Report last error technical information
<b>INF</b>	Set/Report inform-after-motion mode
<b>LOP</b>	Report last position object sensor triggered
<b>PER</b>	Report previous error reason (robot)
<b>RHS</b>	Report home switches state
<b>STA</b>	Report command execution status

### 5.7.4 Support for systems with absolute encoders

Some robot systems can be built using motors with absolute encoders. Maintaining these systems might require following specific commands:

<b>_AER</b>	Reset absolute encoder warning or error state
<b>_AEST</b>	Report absolute encoder status
<b>_EO</b>	Set/report encoder offset
<b>_EOR</b>	Set/Report absolute encoder offsets for all axes
<b>_RALMC</b>	Report low-level absolute encoder status

### 5.7.5 Basic (non-axes, non-station) action commands

Some commands perform actions which cannot be linked to axes or stations, and are listed separately.

<b>_BB</b>	Turn backup DC power on/off
<b>_VDCA</b>	Set/Report Motor Voltage Limit Action
<b>CSTP</b>	Enable/Disable QuickStop feature
<b>DIN</b>	Read input value from external I/O module
<b>DOUT</b>	Set output value to external I/O module
<b>EDG</b>	Get status or operate Edge Gripper
<b>INP</b>	Read an input value
<b>LENB</b>	Enable/Disable "Loaded Speeds" feature
<b>OUT</b>	Set an output value
<b>SOF</b>	Turn robot, axis or REE Servo OFF
<b>SON</b>	Turn robot, axis or REE Servo ON
<b>VOF</b>	Turn vacuum OFF
<b>VON</b>	Turn vacuum ON
<b>VST</b>	Check for wafer presence using vacuum

### 5.7.6 Basic axis motion parameters

Axis motion behavior depends on many different types of parameters - low-level PID parameters, speed parameters, motion restrictions. The following commands operate with axes, their state and current parameters:

<b>_CH</b>	Set/Report customized home position
<b>_CL</b>	Set/Report axis current limit
<b>_EL</b>	Set/Report axis position error limit
<b>_FL</b>	Set/Report axis positive software limit
<b>_FSE</b>	Set/Report axis forward safe envelope
<b>_HD</b>	Set/Report axis PID loop differential coefficient during axis hold
<b>_HE</b>	Set/Report axis position error limit during axis hold
<b>_HI</b>	Set/Report axis PID loop integral coefficient during axis hold
<b>_HL</b>	Set/Report axis PID loop integral limit during axis hold
<b>_HO</b>	Set/Report axis home offset
<b>_HP</b>	Set/Report axis PID loop proportional coefficient during axis hold
<b>_IL</b>	Set/Report axis PID loop integral limit
<b>_KD</b>	Set/Report axis PID loop differential coefficient
<b>_KI</b>	Set/Report axis PID loop integral coefficient
<b>_KP</b>	Set/Report axis PID loop proportional coefficient
<b>_RL</b>	Set/Report axis negative software limit
<b>_RSE</b>	Set/Report axis reverse safe envelope
<b>ACL</b>	Set/Report axis acceleration
<b>AJK</b>	Set/Report axis acceleration jerk
<b>DCL</b>	Set/Report axis deceleration
<b>DJK</b>	Set/Report axis deceleration jerk

<b>HFP</b>	Set/Report axis safe flip position
<b>LACL</b>	Set/Report axis loaded acceleration
<b>LAJK</b>	Set/Report axis loaded acceleration jerk
<b>LDCL</b>	Set/Report axis loaded deceleration
<b>LDJK</b>	Set/Report axis loaded deceleration jerk
<b>LSPD</b>	Set/Report axis loaded speed
<b>SPD</b>	Set/Report axis working speed

### 5.7.7 Axis parameter sets, and custom parameters commands

There are several commands which operate on parameter sets as an entity. See below for detailed description of the feature.

<b>_CAX</b>	Set/Report custom axis parameter
<b>_CPA</b>	Set/Report custom generic parameter
<b>_CST</b>	Set/Report custom stations parameter
<b>CSS</b>	Copy one parameter set into another
<b>CSW</b>	Copy parameter set to working values
<b>CWP</b>	Copy PID parameters to all parameter sets
<b>CWS</b>	Copy working values as a parameter set
<b>DAS</b>	Report all parameter sets for single axis
<b>DPS</b>	Report single parameter set for all axes
<b>HSA</b>	Set homing speed and acceleration
<b>LSA</b>	Set loaded speed and acceleration
<b>WSA</b>	Set working speed and acceleration

### 5.7.8 Basic axis motions commands

Following commands execute simple per-axis motion. These commands use an interlocking feature preventing motions if conditions are not safe, eg robot servo is off, or another motion is running, or motions are blocked.

<b>_AFE</b>	Find limit-switch to absolute encoder zero distance
<b>_FE</b>	Find Edge (distance between home switch and motor index)
<b>_FH</b>	Find Home Switch
<b>_HM</b>	Home an Axis
<b>AFE</b>	Find limit-switch to absolute encoder zero distance
<b>HOM</b>	Start homing the robot
<b>MOV</b>	Move relative over vector
<b>MTH</b>	Move all axes to home position
<b>MTB</b>	Move an axis until sensor is triggered
<b>MTP</b>	Move three or four axes to position
<b>MVA</b>	Move an axis to absolute position
<b>MVR</b>	Move an axis to relative position
<b>MVT1</b>	Move over line
<b>MVT2</b>	Move over two lines
<b>MVTP</b>	Set/Report End Effector Length for station-less motions
<b>RDW</b>	Rotate end-effector down

<b>RUP</b>	Rotate end-effector up
<b>STP</b>	Stop motion
<b>WMC</b>	Wait for motion completed

### 5.7.9 Station parameters commands

Complex robot motions are executed using a concept of a "Station" - a set of coordinates, and added properties, representing a physical point in the robot working envelope. Some of the parameters listed here can be modified only when system is placed in "Station Teach Mode". Following commands set or report station parameters:

<b>_CSS</b>	Set/Report current station set number
<b>_PA</b>	Set/Report R axis pushing acceleration
<b>_PS</b>	Set/Report R axis pushing speed
<b>_ZC</b>	Set/Report station coordinate compensation with slot
<b>APP</b>	Report Inline Station' Approach position
<b>COF</b>	Set/Report station center Offset
<b>COFP</b>	Set/Report additional station center Offset when using edge gripper
<b>CSL</b>	Set/Report station current slot
<b>CSP</b>	Clear single or all station parameters
<b>CST</b>	Set/Report current station, by name
<b>CSTN</b>	Set/Report current station, by number
<b>DSP</b>	Duplicate Station Parameters, by name
<b>DSPN</b>	Duplicate Station Parameters, by number
<b>EEN</b>	Set/Report station servicing end effector
<b>EEO</b>	Set/Report rotating end effector orientation for station
<b>ENS</b>	Enumerate existing stations
<b>EOT</b>	End of teaching a station
<b>ILS</b>	Set/Report inline station last segment speed factor
<b>IRR</b>	Set/Report inline station retracted R position
<b>IRT</b>	Set/Report inline station retracted T position
<b>ISA</b>	Set/Report inline station exit angle
<b>ISD</b>	Set/Report inline station exit distance
<b>ISE</b>	Set/Report inline station end effector length
<b>ISR</b>	Set/Report inline station corner radius
<b>IST</b>	Set/Report inline station type
<b>NSL</b>	Set/Report station number of slots
<b>NST</b>	Report number of stations
<b>OFS</b>	Set/Report station offset
<b>PGD</b>	Set/Report station Put and Get Delay
<b>PGF</b>	Put (place) and Get (pick) failure options
<b>PGO</b>	Set/Report Put/Get Option and T-Z axes synchronization mode
<b>PIT</b>	Set/Report station pitch
<b>PSD</b>	Set/Report pushing distance
<b>PSH</b>	Set/Report pusher type to use with a station

<b>PSS</b>	Set/Report start pushing offset
<b>PSZ</b>	Set/Report Z push offset
<b>PUA</b>	Set/Report station pickup Z acceleration
<b>PUS</b>	Set/Report station pickup Z speed
<b>RFL</b>	Turn on or off rotate-on-the-fly permanently
<b>ROF</b>	Turn on or off rotate-on-the-fly for next GET or PUT operation
<b>RPO</b>	Set/Report radial retracted station position
<b>SPO</b>	Set/Report station coordinate, by name
<b>SPON</b>	Set/Report station coordinate, by number
<b>STR</b>	Set/Report station' stroke distance
<b>TBL</b>	Set/Report Trajectory Mode Overlap factor
<b>TCH</b>	Start teaching a station
<b>TMD</b>	Set/Report Trajectory Motion mode
<b>VCHK</b>	Set/Report vacuum check mode before Put and Get
<b>VTO</b>	Set/Report vacuum timeout before Put and Get

#### 5.7.10 Station motion commands

Following is list of commands used to perform motion operations against stations. Commands for scanning, and OtF aligning, are not included in this list.

<b>GET</b>	Get (pick) a wafer from a station (by name), with end option
<b>GETN</b>	Get (pick) a wafer from a station (by number)
<b>GETO</b>	Get (pick) a wafer from a station, with temporary offsets from taught position
<b>GETR</b>	Get (pick) a wafer from a station (by name), with end option
<b>GETRT</b>	Get (pick) a wafer from a station (by name), with retry option
<b>GETW</b>	Get (pick) a wafer from a station, with wait option at pick-up position
<b>GETXY</b>	Get (pick) a wafer from a station with defined X/Y offsets
<b>MOS</b>	Move arm out of station
<b>MSS</b>	Move a wafer from station to station (by name)
<b>MSSN</b>	Move a wafer from station to station (by number)
<b>MTA</b>	Move to Station' Approach position
<b>MTR</b>	Move to Station' Retracted position
<b>MTS</b>	Move to Station Coordinate
<b>PGWA</b>	Set/Report wait-at-station position state
<b>PUT</b>	Put (place) a wafer into a station (by name)
<b>PUTN</b>	Put (place) a wafer into a station (number)
<b>PUTO</b>	Put (place) a wafer into a station, with temporary offsets
<b>PUTW</b>	Put (place) a wafer into a station, with wait option at drop-down
<b>SBUSY</b>	Check whether a station is being serviced
<b>SGP</b>	Move a wafer from station to station, with end option
<b>SLCK</b>	Set/Report Station Lock State
<b>SPG</b>	Place a wafer, pick a wafer, with end option

### 5.7.11 Station scanning parameters and motion commands

A separate set of parameters and commands is available to set, report and execute station scanning (wafer mapping) operations. Some of the parameters listed here can be modified only when system is placed in "Station Teach Mode".

<b>_BT</b>	Set/Report Beam threshold above/below a wafer (global)
<b>_MT</b>	Set/Report Maximum Wafer Thickness (global)
<b>_SA</b>	Set/Report scanning acceleration
<b>_SC</b>	Set current position as Center Scanning Position
<b>_SL</b>	Set current position as Left Scanning Position
<b>_SR</b>	Set current position as Right Scanning Position
<b>_SS</b>	Set/Report scanning speed
<b>_ST</b>	Set/Report Scanner Type and protrusion detection option
<b>DSCN</b>	Report samples data from scanning
<b>MAP</b>	Prints the result of last scanning operation
<b>MSC</b>	Move to Scanning Coordinate
<b>SBT</b>	Set/Report Beam threshold above/below a wafer (per-station)
<b>SCN</b>	Start Scanning procedure
<b>SCNS</b>	Start Scanning procedure with safety scan
<b>SCP</b>	Set/Report scanning center position
<b>SDBG</b>	Set/Report scanning debug mode
<b>SLP</b>	Set/Report scanning left position
<b>SMT</b>	Set/Report Maximum Wafer Thickness (per-station)
<b>SRP</b>	Set/Report scanning right position
<b>SZH</b>	Set/Report scanning highest Z position
<b>SZL</b>	Set/Report scanning lowest Z position
<b>TCS</b>	(re)Teach scanning position
<b>TSP</b>	Test scanning position

### 5.7.12 Align-on-Flight (On-The-Flight, OtF) parameters and commands

OtF feature allows for measuring of wafer center displacement during pick and/or place motions. Following commands are available to define parameters and execute motions:

<b>AOFCN2</b>	Set/Report Check for Notch on Second Scan Option
<b>ACS</b>	Report OtF configuration set number
<b>AOFC</b>	Execute OtF calibration run
<b>AOFCS</b>	Execute OtF calibration run against a station
<b>AOFD</b>	Set/Report OtF debug level
<b>AOFIM</b>	Set/Report OtF samples Ignore Mode
<b>AOFDUMP</b>	Report Samples from Last OtF Operation
<b>AOFMD</b>	Set/Report Default Maximum Allowed center offsets
<b>AOFM</b>	Set/Report Maximum allowed center offsets
<b>AOFNL</b>	Set/Report OtF Notch Limit
<b>AOFPMI</b>	Report Debug Information from last OtF Execution
<b>AOFSA</b>	Set/Report Axis Search Speed and Acceleration

<b>ASL</b>	Set/Report On-The-Fly wafer radius/end effector length
<b>GETC</b>	Get (pick) a wafer from a station with on-the-fly offset calculation
<b>GETCW</b>	Get (pick) a wafer from a station with on-the-fly offset calculation, and wait option
<b>MVRC</b>	Execute relative axis motion with on-the-fly sensor capture
<b>PUTC</b>	Put (place) a wafer into a station (letter) with OtF offset calculation and correction
<b>PUTCW</b>	Put (place) a wafer into a station (letter) with OtF offset and wait option
<b>PUTM</b>	Put (place) a wafer into a station (letter) with OtF Offset Measurement
<b>PUTXY</b>	Put (place) a wafer into a station with X/Y offsets from OtF calculation
<b>PUTXYW</b>	Put (place) a wafer into a station with X/Y offsets from OtF calculation and wait option
<b>RWIC</b>	Report center offset from last OtF operation

### 5.7.13 Intelligent Aligner Support parameters and commands

When intelligent single- or three-axis aligner is configured and connected, following parameters and commands affect its operation:

<b>_CCDA</b>	Set/Report CCD sensor angle
<b>_CCDC</b>	Set/Report CCD sensor center
<b>_CCDH</b>	Set/Report CCD shadow correction coefficient, high
<b>_CCDL</b>	Set/Report CCD shadow correction coefficient, low
<b>_EH</b>	Report Edge-Handling Aligner Type
<b>_GLM</b>	Set/Report transparency (glass) mode
<b>_LD</b>	Set/Report Wafer Load-Down position
<b>_LU</b>	Set/Report Wafer Load-Up position
<b>_MA</b>	Set/Report Measuring Acceleration
<b>_MD</b>	Set/Report Measuring Deceleration
<b>_MND</b>	Set/Report minimum notch depth
<b>_MNW</b>	Set/Report minimum notch width
<b>_MOF</b>	Set/Report maximum calculated offset
<b>_MS</b>	Set/Report Measuring Speed
<b>_MXD</b>	Set/Report maximum notch depth
<b>_MXW</b>	Set/Report maximum notch width
<b>_PSN</b>	Set/Report Prealigner Station Name
<b>_PSR</b>	Set/Report Prealigner Safe Retract distance
<b>_SELECTFLAT</b>	Set/Report selected flat number for orientation
<b>_SM</b>	Set/Report Sampling Mode
<b>_TO</b>	Set/Report Chuck vacuum timeout
<b>_WT</b>	Set/Report Wafer type
<b>ALN</b>	Start Aligning a wafer (wafer on chuck)
<b>BAL</b>	Begin alignment (wafer on paddle)
<b>CCD</b>	Report current value of CCD sensor
<b>CVF</b>	Turn Chuck Vacuum OFF
<b>CVN</b>	Turn Chuck Vacuum ON

<b>DOC</b>	Report Object Detected on Chuck
<b>FWO</b>	Set/Report Final wafer orientation
<b>FWOH</b>	Set/Report Final wafer orientation, high resolution
<b>LTC</b>	Set/Report Load-On-Chuck mode
<b>MTL</b>	Move Wafer to Loading Position
<b>NBN</b>	Report number of bad notches found
<b>NDW</b>	Report depth and width of a notch
<b>NGN</b>	Report number of good notches found
<b>PAR</b>	Begin alignment and retract
<b>PPR</b>	Report Previous Error Reason (aligner)
<b>PRA</b>	Set/Report Angle between Prealigner and Robot
<b>PRD</b>	Set/Report Distance between Prealigner and Robot
<b>PRT</b>	Report Prealigner Type
<b>PST</b>	Report Native Aligner Status
<b>PVF</b>	Turn Pins Vacuum OFF
<b>PVN</b>	Turn Pins Vacuum ON
<b>RCI</b>	Read Correction Information
<b>RWI</b>	Read Wafer Information
<b>SSL</b>	Set/Report Single Step compensation Limit
<b>VMD</b>	Set/Report Prealigner Vacuum Mode
<b>WSZ</b>	Set/Report Wafer size

## 5.8 AXES, INPUTS AND OUTPUTS: DESCRIPTION, AND COMMANDS USED

Each robot or prealigner system consists of several basic components: moving axes, inputs and outputs controlling various subsystems, and the enclosure for them.

### 5.8.1 Axes

Each axis is being driven by a brushless servo motor, geared through a gearbox, a system of belts, or ball-screws. Some axes require simultaneous action of two or more motors to achieve the desired axis motion, which the firmware will translate between the units of measure used for a particular axis, and their low-level implementation.

Apart from the velocity profile parameters associated with each axis (velocity, acceleration, rate of change of acceleration), another set of parameters (servo parameters) is associated with the low-level servo controller driving each motor. While velocity profile parameters can be freely changed by customer, servo parameters are usually set at the factory, and their change should be made with full understanding of their impact of axis motion.

Axes motion commands employ an interlocking mechanism which prevents execution of a motion command while an axis is in motion, or the system is busy executing the user command. Motion commands cannot be executed if the system is in the Emergency Stop state.

For each axis, a predefined position is designated as a zero position, and homing procedure for an axis ensures that at the end of the homing, the axis will always be positioned at the same physical point. For more information about the homing position see Chapter 5.9.1 Homing Procedure for Systems Using Motors with Incremental Position Encoders.

Some typical values for velocity parameters used in robot systems are listed in the following



Parameter and command	Theta (T)	Arm (R)	Vertical (Z)	Track (K)	Elevator (V)
Speed ( <b>SPD</b> )	3200	3200	1600	1800	600
Acceleration ( <b>ACL</b> )	8000	12000	4200	4200	800
Deceleration ( <b>DCL</b> )	7200	10000	4200	3800	800
Accel. Jerk ( <b>AJK</b> )	700	700	750	750	650
Decel. Jerk ( <b>DJK</b> )	600	600	600	600	500

**Table 5-1: Typical values for velocity parameters**

**NOTE:** These values are for reference only, and depend heavily of particular robot configuration.

### Axis Parameter Sets

A feature is being implemented for up to ten pre-defined sets of related parameters for an axis. A parameter set consists of following items:

- Velocity parameters (speed, acceleration, deceleration, acceleration jerk, deceleration jerk), accessed with commands **SPD**, **ACL**, **DCL**, **AJK**, **DJK**;
- PID parameters (proportional, differential, and integral coefficients, integral limit, following error limit and motor current limit) accessed with commands **\_KP**, **\_KD**, **\_KI**, **\_IL**, **\_EL**, **\_CL**;

When a particular parameter set is selected, velocity and PID parameters from this parameter set replace current velocity and parameter sets in the robot controller.

Since parameter sets operate on a level below any robot firmware, they do not take immediate effect on some of commands. Their values are in default units for the robot controller. For example, **.vel** command expects and returns values in thousandth of an inch (0.001"/sec), while robot **SPD** command reports speeds in hundredth of an inch (0.01"/sec). Any changes made to current velocity and PID parameters are not saved as permanent until corresponding **SMP** command has been issued.

If a parameter set does not contain any velocity or PID parameter for any axis, this parameter set is labeled as invalid and it cannot be made active, unless explicitly requested by a parameter.

No default values are provided with robot firmware, therefore all parameter sets are marked as invalid upon first startup.

### Working Examples with Parameter Sets

The following examples will provide details for the above commands.

```

>SPD
500,500,800
>ACC
5000,2000,1000
>DCL
5000,2000,1000
>AJK
750,850,750
>DJK
550,750,600
>_KP
300,300,65
>_KD
4000,4000,2500
>_KI
40,40,45
>_IL
40,150,80

```

```

>_EL
250,40,100
>_CL
201,121,251
>

```

The above commands and responses show current velocity and PID parameters for a robot. For measuring units of velocity, acceleration and deceleration see the descriptions of commands **SPD**, **ACL** and **DCL** in Chapter 6, Commands Reference.

```

>DAS T
Axis      T      0      1      2      3      4      ...      7      8      # 9#
Velocity  500    500    500    500    500    ...    5000   5000   0
Acceler.  1000   1000   1000   1000   1000   ...    50000  50000  50000
Deceler.  1000   1000   1000   1000   1000   ...    50000  50000  50000
Acc Jerk  750    750    750    750    750    ...    750    750    750
Dec Jerk  550    550    550    550    550    ...    550    550    550
PID_KP    300    300    300    300    300    ...    300    300    300
PID_KD    4000   4000   4000   4000   4000   ...    4000   4000   4000
PID_KI     40     40     40     40     40     ...    40     40     40
PID_IL     40     40     40     40     40     ...    40     40     40
PID PosE   250    250    250    250    250    ...    250    250    250
PID CLim   201    201    201    201    201    ...    201    201    201
### Some sets are INVALID ###
>

```

The above command shows the output for **DAS T** command. As seen from this example, parameter set 9 is marked as invalid with two hash signs around its number, and a note is shown after the table. For measuring units see the description of **DAS** in Chapter 6 Commands Reference.

```

>DPS 9
Set No 9      T      R      Z
Velocity  0    5000   8000
Acceler.  50000 20000 10000
Deceler.  50000 20000 10000
Acc Jerk  750    850    750
Dec Jerk  550    750    600
PID_KP    300    300    65
PID_KD    4000   4000   2500
PID_KI     40     40     45
PID_IL     40    150     80
PID PosE   250    40    100
PID CLim   201    121   251
### INVALID SET ###
>

```

The above example shows contents of parameter set 9. Since T axis velocity is zero, this parameter set is marked as invalid. For measuring units see the description of **DPS** in Chapter 6, Commands Reference.

```

>SPD
0,500,800
>SPD T 1200
>_KP T 320
>_KD T 4500
>CWS 8
>CWS 7 p
>CWS 6 v
>DAS T
Axis      T      0      1      2      3      4      ...      7      8      # 9#
Velocity  500    500    500    500    500    ...    5000  12000   0
Acceler.  1000   1000   1000   1000   1000   ...    50000  50000  50000
Deceler.  1000   1000   1000   1000   1000   ...    50000  50000  50000
Acc Jerk  750    750    750    750    750    ...    750    750    750
Dec Jerk  550    550    550    550    550    ...    550    550    550
PID_KP    300    300    300    300    300    ...    320    320    300

```

```

PID _KD      4000  4000  4000  4000  4000  ...  4500  4500  4000
PID _KI       40    40    40    40    40    ...   40    40    40
PID _IL       40    40    40    40    40    ...   40    40    40
PID PosE     250    250    250    250    250  ...  250    250    250
PID CLim     201    201    201    201    201  ...  201    201    201
### Some sets are INVALID ###
>

```

Above commands change velocity of T axis, as well as some of PID parameters. Then, all parameters are copied to parameter set 8, PID parameters only are copied to set 7, and velocity parameters are copied to set 6. All parameter sets for T axis are displayed once again.

```

>CSW 0
>SPD
50,50,20
>CSW 9
?SPD
50,50,20
>

```

In this example, parameters get changed twice – first time from parameter set 0, then from parameter set 9. Since parameter set 9 is invalid, command is not executed, and an error is displayed. Providing V option to **csw** command would allow us to change the speed to desired value.

```

>CSW 9 V
>SPD T 1500
>CWS 9 V
>SPD
1500,500,800
>DPS 9
Set No  9      T      R      Z
Velocity 15000  5000  8000
Acceler.  50000 20000 10000
Deceler.  50000 20000 10000
Acc Jerk   750   850   750
Dec Jerk   550   750   600
PID _KP    300   300   65
PID _KD    4000  4000  2500
PID _KI     40    40    45
PID _IL     40   150    80
PID PosE    250   40   100
PID CLim    201  121  251
>SPS
>

```

Finally, parameter sets are saved back to controller storage.

## 5.8.2 Inputs and Outputs

Each robot and prealigner system uses at least a few inputs and outputs for its normal operation. Some of these inputs and outputs are routed internally to the system (e.g. home switches, vacuum sensors, and vacuum valves), others are accessible for additionally attached equipment (inputs and outputs in the robot arm), while some control internal signals (e.g. type of wafer mapping sensor used). Depending of particular configuration, some inputs and outputs may or may not be present, or have their names and function changed.

On MCL5 systems, input and output numbering can be changed by the user. Additional user-accessed inputs and outputs can be defined as well. Input and output numbers listed below reflect factory-standard assignment.

## Single-arm Robot Systems with optional track inputs and outputs

Input No	Type (A/D)	Description
0	D	Home Switch T Axis (located close to the middle of axis motion range)
1	D	Home Switch R Axis (located close to the center of axis motion range)
2	D	Home Switch Z Axis ( located close to the bottom end of axis motion range)
3	D	Vacuum Sensor 1
4	D	Vacuum Sensor 2 (if installed)
5	D	Wafer mapping sensor (if installed)
6	D	Emergency stop activated
7	D	Home Switch K (Axis)
9	D	Rotating End-Effector (REE) Sensor 1 (if installed)
10	D	Rotating End-Effector (REE) Sensor 2 (if installed)
11	D	Edge Gripper Sensor 1 (if installed)
12	D	Edge Gripper Sensor 2 (if installed)
13	D	Quick-Stop input (if configured)
14	D	Object (wafer) position sensor, if installed

**Table 5-2: Single-arm Robot Standard Inputs**

Output No	Type (A/D)	Power-On Value	Description
0	D	0	Vacuum Valve 1. To keep valve closed (i.e. no vacuum present on end effector), value of 1 has to be written to this output. To open the valve, value of 0 has to be written
1	D	0	Vacuum Valve 2 (if present) Look previous description
5	D	0	Wafer Scanner Enable2 (if installed)
8	D	0	User-defined, OutputE on DB25 I/F
9	D	0	REE 1 Control Output (if installed)
10	D	0	REE 2 Control Output (if installed)
11	D	0	REE Power Control (if installed)
12	D	0	Edge Gripper Valve Control (if installed)
13	D	0	Bernoulli Effect stream control (if installed)

**Table 5-3: Single-arm Robot Standard Outputs**

## External I/O Interface Inputs and Outputs

By request, additional digital I/O interface can be installed on the robot system. **DIN** and **DOUT** commands can be used to control and query the state of this interface. Some of these I/O pins might be used by other firmware capabilities (e.g. controlled stop, or object position sensor, or on-the-fly object alignment), and therefore unavailable (or not affected) by these commands.

For pin layout, look Chapter 3.1.8 Extended I/O Interface Connector

I/O Pin Name	System I/O Number	I/O Pin Name	System I/O Number
InputE	33 ( <b>DIN</b> 0)	OutputE	18 ( <b>DOUT</b> 8)
InputF	34 ( <b>DIN</b> 1)	OutputF	25 ( <b>DOUT</b> 0)
InputG	35 ( <b>DIN</b> 2)	OutputG	26 ( <b>DOUT</b> 1)

I/O Pin Name	System I/O Number
InputH	36 (DIN 3)
InputJ	37 (DIN 4)
InputK	38 (DIN 5)
InputL	39 (DIN 6)
InputM	40 (DIN 7)
InputN	41 (DIN 8)
InputR	42 (DIN 9)

I/O Pin Name	System I/O Number
OutputH	27 (DOU <del>T</del> 2)
OutputJ	28 (DOU <del>T</del> 3)
OutputK	29 (DOU <del>T</del> 4)
OutputL	30 (DOU <del>T</del> 5)
OutputM	31 (DOU <del>T</del> 6)
OutputN	32 (DOU <del>T</del> 7)

Table 5-4: I/O number assignments for external I/O interface: Single-arm robot

I/O Pin Name	System I/O Number
InputE	37 (DIN 0)
InputF	38 (DIN 1)
InputG	39 (DIN 2)
InputH	40 (DIN 3)
InputJ	41 (DIN 4)
InputK	42 (DIN 5)
InputL	43 (DIN 6)
InputM	44 (DIN 7)
InputN	45 (DIN 8)
InputR	46 (DIN 9)

I/O Pin Name	System I/O Number
OutputE	20 (OU <del>T</del> 9)
OutputF	27 (DOU <del>T</del> 0)
OutputG	28 (DOU <del>T</del> 1)
OutputH	28 (DOU <del>T</del> 2)
OutputJ	30 (DOU <del>T</del> 3)
OutputK	31 (DOU <del>T</del> 4)
OutputL	32 (DOU <del>T</del> 5)
OutputM	33 (DOU <del>T</del> 6)
OutputN	34 (DOU <del>T</del> 7)

Table 5-5: I/O number assignments for external I/O interface: Dual-arm robot

**NOTE:** System I/O numbers might change with future firmware releases. Please look at relevant release notes for details. OutputE might not be available as user-accessible output in some configurations.

### 5.8.3 Environment monitoring

Current versions of robot controllers support monitoring of controller temperature and motor voltage. Provisions are made to restrict motion commands if motor voltage goes below predefined threshold. Controller hardware provides for automatic servo shutdown in case of controller overheating.

## 5.9 ARM CONFIGURATIONS

Apart from standard Atmospheric robot with vacuum-operated end effector, several other options exist for tooling of the robot arm. These configurations are defined through a special entry in robot' parameters file, params.ini (in case of three-axis standalone robot).

### 5.9.1 Homing Procedure for Systems Using Motors with Incremental Position Encoders

When an axis is homed (either individually, with **\_HM** command, or as part of robot homing sequence with **HOM** command), following steps are performed:

- Axis velocity profile is set to homing speed and acceleration. These speeds and acceleration can be set with **HSA** command, but they are predefined and there are no user commands to change them. These speeds, however, can be changed in corresponding parameter file, see Chapter 7.3 Robot Configuration and Firmware files for additional information.
- If axis' home switch is activated (corresponding input value, e.g. **INP 0** returns 1), axis starts moving in positive direction. Otherwise, axis starts moving in negative direction;

- As soon as axis home switch changes its value, axis stops with predefined deceleration.
- If Theta axis is homed, (b) and (c) are repeated, if necessary, so that final search for home switch is made always in negative direction.
- Using same speeds and acceleration, axis starts moving in positive direction, until index pulse from the motor encoder is captured. Axis stops, and then moves to captured index position.
- Axis position is reset to zero.
- If homing is part of homing sequence (invoked with **HOM** command), following additional steps are performed:
- Axis position is set to negative value of axis' home offset (defined with **\_HO** command). No motion takes place.
- After all axes are homed, axes are moved to their custom home positions (defined with **\_CH** command).
- Robot velocity profile remains unchanged, with pre-defined homing speeds and accelerations.
- When the robot is homed, and any moment after that, **MTH** command can be used to move robot back to custom home position.
- If normal speed of operation is required after completing home sequence, **WSA** command is a quick way to return the robot to operational speeds and accelerations.
- The diagram on following figure shows the relationships between home switch, index, and zero position. This could be a typical diagram for homing theta or arm axes, where axis' zero position is approximately in the middle of the motion range.

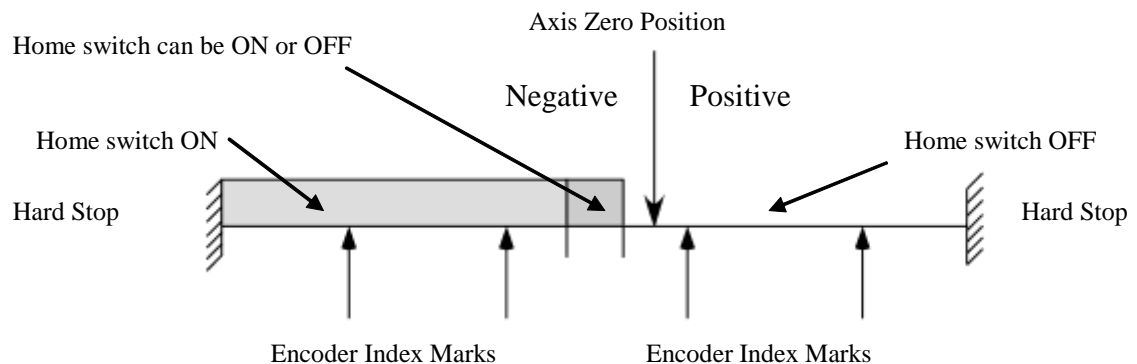


Figure 5-1: Axis Homing Diagram

### 5.9.2 Setting Home Offsets for Systems Using Motors with Incremental Position Encoders

Robots are shipped out of factory with predefined home position, based on customer order. In case factory-shipped home position has to be changed (e.g. to match a replaced tool, or after changing limit switch positions), following simple procedure can be followed:

**NOTE:** Before starting any movement operations with the robot make sure that it is safe to do so, and there are no obstacles in the way of the moving robot arm.

- take a note on current home offsets, and then reset them to zero

```
> _HO
-4250, -315, 215
> _HO T 0
> _HO R 0
> _HO Z 0
>
```

- take a note on current Custom (user) Home Positions, and then reset them to zero

```

>_CH
-4000,-10000,0
>_CH T 0
>_CH R 0
>_CH Z 0
>

```

After all offsets are reset, the homing sequence will cause the axes to stop at the motor's first index position, after the home switch in positive direction.

- Find index position for each axis in following order: T, R and Z

```

>_FE T
Home Switch to index distance: 227 (4572 encoder counts)
>_FE R
Home Switch to index distance: 564 (3056 encoder counts)
>_FE Z
Home Switch to index distance: 358 (2063 encoder counts)
>

```

The values displayed during execution of **\_FE** macro show the distance, both in user units (e.g. 227 == 2.27°) and motor encoder counts. Those values are for reference purposes only, and are used to position the axis precisely each and every time. It's extremely important to keep these values within specified range of encoder counts, or homing will be not accurate.

- Perform complete homing procedure

```

>HOM
>

```

After this procedure finishes, robot's arm will be positioned in such a way that position T=0, R=0, Z=0 would correspond to motors sitting on corresponding index positions. See Chapter 4.11, Coordinate system used in single-arm wafer handling robot, for additional information.

- Find the necessary Home offset so that arm is in overlapped center position (Figure 4-2: Arm at initial (home) position (T=0, R=0, Z=0)).
- Using commands **MVA R <value>** and **MVR R <value>**, position the arm in such a way that both arm's links are exactly in overlapped position, as in picture above.
- When arm links are in overlapped position, enter current R axis position as home offset:

```

>_HO R
Home offset is set to -324
>

```

- Perform same procedure for axes T, Z, and track (if installed) so that zero position of those axes is according to robot's specification.
- Confirm that robot is correctly homed by issuing command

```

>HOM
>

```

- Check that after command is finished, arm links are in overlapped position, and T and Z axes are in checked position.
- Restore previous values for Customized (User) home position

```

>_CH T -4000
>_CH R -10000
>_CH Z 0
>

```

- Restore operational values for axes' speed, acceleration and jerks, and save motion and calibration parameters:

```

>WSA
>SMP
>

```

- Failure to restore operational (working) parameters with **wsa** command will store currently active speed and acceleration values as future operational speed and

acceleration, after resetting the robot controlled. Because last motion commands which the robot executed were performed with much lower (homing) speeds and acceleration, not issuing **WSA** command before **SMP** command would destroy previously saved operational parameters.

## 5.10 WAFER HANDLING ROBOTS WITH ABSOLUTE POSITION ENCODER

The robot axes can be equipped with absolute encoders. Under normal circumstances, the robot maintains the axes positions even when power is turned down. A battery built into the robot controller takes care of supplying power to the encoders during power-off condition, for limited battery-specific time. When extended off-power storage is expected, an external battery power should be connected to the front panel connector (see Interface Panel layout).

However, if supporting battery has been discharged completely, absolute encoders lose their turns counter, and robot must be recalibrated in order to continue normal operation. Homing with initializing the absolute encoders is a procedure provided in the process of calibrating the robot in the factory. It may be needed to repeat the procedure on the field only in case of discharging the battery supporting those encoders. It may happen if the robot is stored for a long time without proper support of the battery. The example below illustrates the a calibration procedure for a 3-axis robot:

Step	Command	Description and Notes
1.	<b>SOF</b>	Turn Servo Off, prepare for initializing (clearing) absolute encoders turns counter
2.	<b>_AER A 1024 1003</b>	Clear all abs. encoders turns counter (1024 – clear, 1003 – password)
3.	<b>SON</b>	Turn Servo On for all the axes of the robot and elevator (if applicable)
4.	<b>AFE R</b>	find home R-axis switch with absolute encoders. The robot starts moving in the direction of the R-axis home switch, which is located close to arm' overlapped position. Make sure there are no obstacles during this motion!
5.	<b>AFE T</b>	find home T-axis switch with absolute encoders
6.	<b>AFE Z</b>	find home Z-axis switch with absolute encoders
7.	<b>SOF</b>	Turn Servo Off, prepare for initializing (clearing) absolute encoders turns counter
8.	<b>_AER T 1024 1003</b>	Clear T abs. encoders turns counter (1024 – clear, 1003 – password)
9.	<b>_AER R 1024 1003</b>	Clear R abs. encoders turns counter (1024 – clear, 1003 – password)
10.	<b>_AER Z 1024 1003</b>	Clear Z abs. encoders turns counter (1024 – clear, 1003 – password)
11.		Move arm to overlap position either moved manually, or by turning servo loop ON and using MVR/MVA commands to move axes to desired position.
12.	<b>_EOR #</b>	set all axes encoder offset
13.	<b>WSA</b>	Set working speeds and acceleration parameters
14.	<b>SMP</b>	Save all calibration parameters



## 5.11 STATION PARAMETERS: DESCRIPTION, AND COMMANDS USED

While basic axis motion commands and input/output control commands are enough for simple processing, robot firmware implements a notion of a Station. A station is a point within working envelope, defined by the positions of the axes when end effector is at that point. In order to facilitate using of the robots in wafer handling and processing, a station has also some additional properties, which will be described below. With stations properly configured, several complex operations are possible with a simple command sent to the robot. Each of these commands can use axis motion in several directions, checking input signals and operating output signals depending on the state and position.

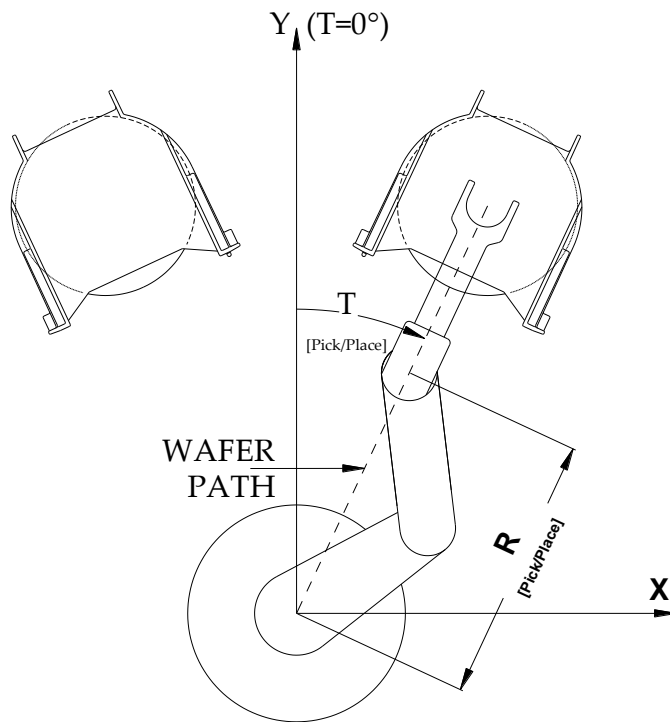


Figure 5-2: T, R coordinates of a station

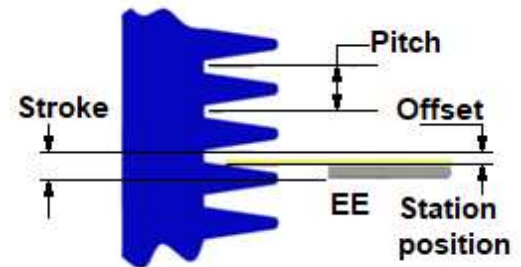


Figure 5-3: Z coordinate of a station

Station coordinates specify the coordinates of the first (or only) slot in a station. Station teaching process starts with **TCH** command, and this mode is indicated with bit 11, or 0x0800 in robot status word. When end-of-teach command, **EOI**, is invoked, then the current positions of the robot axes are stored as station coordinate. If one or several station coordinates are set explicitly during station teaching with **SPO** command, however, only changed coordinate position is stored, thus allowing for changing of a single coordinate of already taught station. Stations coordinates can be changed only when robot is in station teach mode, other station parameters can be changed at any time. Command **MTS** moves the robot to that position.

Station' R coordinate specifies the position of the arm with the end effector fully into the station position, therefore replacing the end effector would require changing the R coordinate (if new end effector is of different length), Z coordinate (if new end effector' working surface is at different height with regard to previous one), or both. When in non-teaching mode, the **SPO** command reports current coordinates of a station.

- Station pitch (defined with **PIT** command) specifies the distance between two adjacent slots in a cassette (e.g. 0.1875", 0.25", 0.3937"). The unit of measurement is defined in ten-thousandths of an inch, i.e. 1875 means 0.1875"; and 3937 means 0.3937" (or 10mm).

- Station offset (defined with **OFS** command) specifies the distance above station Z position where the end effector will leave the station with a wafer for pick operation, or when approaching a station for a place operation
- Station stroke (defined with **STR** command) specifies the vertical distance between end effector Z positions approaching and leaving the station for a pick-up operation. These numbers are usually positive numbers, with stroke larger than offset, so that when picking a wafer the arm will approach from below taught slot position, and will leave above that position.

Two stations types, depending of the trajectory used to enter and exit a station are supported. Radial stations are ones where end effector moves along the line connecting center of the robot with station coordinate. Inline stations are ones where end effector moves along a set of two straight lines connected by a smooth curve. Station type is defined with **IST** command.

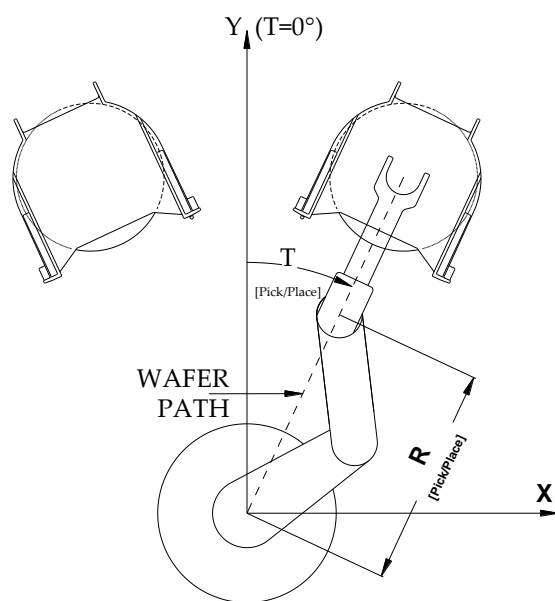
For each station, a timeout is defined with **PGD** command, specifying the time a vacuum sensor needs to detect a wafer placed on the paddle after vacuum valve is opened. This timeout is used for every occasion wafer presence check has to be made for particular station.

Last used station name and slot number become current station name and current station slot, respectively. Commands **CST** and **CSL** can report or change these values. If station command does not specify station name or slot number, current station name and slot number are used.

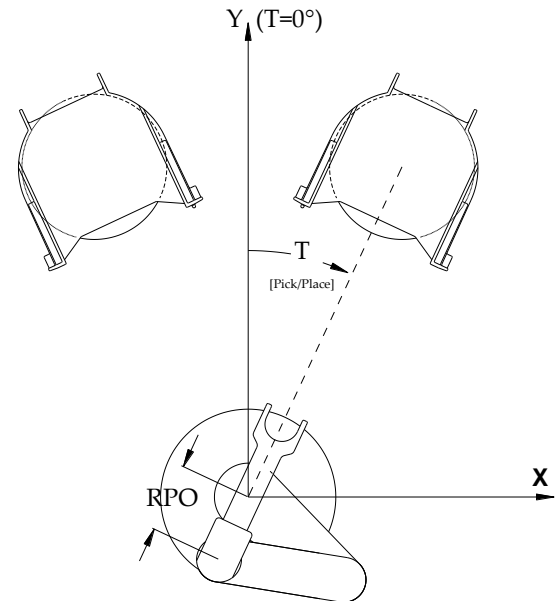
If a station is being pre-defined, **EOT** command stores station coordinates only, retaining all other station parameters.

#### 5.11.1 Radial Stations Coordinates

For each defined radial station, two sets of coordinates are defined: station pick-up coordinates, and retracted station coordinate. Station coordinate is this position of arm with installed end effector, where a wafer placed on the end effector, with center of the wafer seated onto the end effector will sit evenly in a cassette, ready to be picked up. Robot center, upper link arm end and end effector all lay on a common line. Station T coordinate is defined such that a line connecting center of the robot's plate with the center of the wafer sitting on the station is the same as the line where end effector is moving in.



End effector fully extended into an radial station



End effector retracted in front of an radial station

**Figure 5-4: Radial Stations Configuration**

Station retracted position is the position where arm is retracted after pick operation, or before place operation. This position has same Theta coordinate as station position, and R coordinate is usually zero (for dual-paddle end effector), or negative so that wafer center and robot center collide. Station retracted position must be set so that it allows for safe rotation of T axis with or without a wafer on the paddle. No obstacles should be present in the path where arm joint is moving. Command **RPO** can be used to define retracted R coordinate for a station.

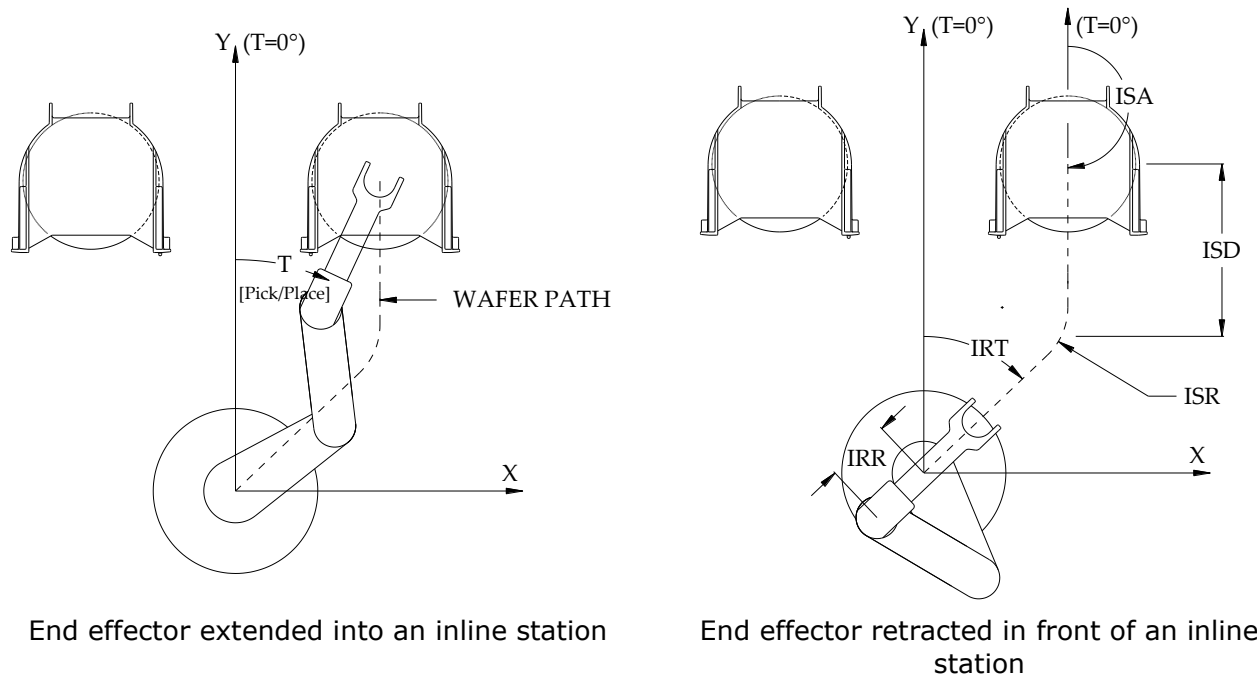
Station search position is pre-defined to be less than R station coordinate. This position is used in vacuum search mode (**PGO 1**), and can be changed by editing station parameters file. For more information refer to Chapter 6, Commands Reference.

### 5.11.2 Inline Stations Coordinates

In-line stations can be useful in space-constrained environments, and are normally used in 300mm applications involving Front-Opening Unified Pods (FOUPs). Since arm mechanics always keeps the paddle oriented along the line crossing robot center, there is only one way the arm can be into a station. Following pictures describe the parameters related to inline stations.

Let's assume that T axis' zero position is set up in such a way that end effector will point toward inline stations, as in the pictures below. Let's assume also that end effector is 7.5in long, and that retracted position for both stations will be the same, at  $T=0$ ,  $R=-4000$ , and that the wafer size is 200mm (roughly 8in).

Wafer trajectory is shown with green line, while axes, dimensions and directions are shown in red.



**Figure 5-5: In-line Stations Configuration**

During inline pick operation, end effector enters the station either radially (as in normal non-inline stations), or along the trajectory, depending of station type defined with **IST** command. Wafer is picked up, and then end effector leaves the station at the direction defined by **ISA** command. The straight path length is defined by **ISD** command, while cornering radius is defined by **TBL** command (**ISR** command is supported only for the syntax backward compatibility, but not used as blending parameter in the MCL5 firmware).

platform). Final wafer position is defined by **IRT** and **IRR** commands, respectively. If the defined trajectory cannot be executed, exit path is built using two straight lines. Ordinary station retracted position (defined with **RPO** command) is used for radial motions.

Since trajectory path depends upon the length of the end effector (it's the end of the paddle which goes along the path), correct end effector length has to be defined with **ISE** command. As a general rule, trajectory should be built in such way that end of paddle (center of the wafer) to travel as far away as possible from center of the robot. In the example shown above it is assumed that Theta axis' zero position points toward the two inline cassettes. If it is not the case then the T-axis orientation pointing toward the stations (or parallel to the exit line from the station) should be taken into account as an inline station exit angle (see **ISA** in Chapter 6, Commands Reference.) Some values for good performance are as follows:

- R retracted position should be as low as possible in negative direction, without wafer going over the center of the robot;
- T retracted position should be about -4000 or +4000 ( $\pm 40^\circ$ ), which is close to exit path of the cassette;
- Exit angle should be 18000 ( $180^\circ$ );
- Exit distance should be longer or close to wafer diameter, 8000 (8") or 12000 (12");
- The smooth curve blending the two straight line segments of the path between the station retracted and pickup positions, is specified by the remaining ratio of first segment length when the approach the the second segment is started, as specified by TBL command parameter.

### 5.11.3 Standard Pick Procedure (a.k.a. GET)

Standard (atmospheric robot) pick procedure consists of following steps:

- Servicing end effector for the station (as defined with **EEN** command) is checked for wafer presence. If a wafer is detected, pick procedure is not executed;
- Arm retracts to target station' retracted position, regardless of inline station type;
- Theta (T), vertical (Z) (and track, K, if present) axes are moved to station' pick coordinates for requested slot. If requested by **PGO** mode, the motion of these two (three) axes is synchronized so that all axes arrive at the same time at the target position. If inline station type requires trajectory motion, Theta is moved to inline T retracted position instead of station position. If REE option is installed, paddle is rotated to match target station orientation;
- Arm is extended to station' R coordinate. If a center offset is defined with **COF** command for that station, R coordinate is adjusted with this offset;
- Vacuum is enabled on servicing end effector;
- Z axis velocity profile is switched to speed and acceleration defined by **PVS** and **PVA** commands. If these values are not specified, homing speed and acceleration are used instead;
- Z axis is moved in positive direction by the distance specified with **STR** command;
- Vacuum sensor is probed for wafer presence for the time defined with **PGD** command.
- If wafer is detected, arm gets retracted to station' retracted position (for radial stations), or over trajectory toward station' T and R inline retracted position (for inline stations).
- If wafer is not detected, and **PGO** mode is set to 3 (wafer required), Z axis strokes back, and arm retracts.
- If wafer is not detected, and **PGO** mode is set to 1 (wafer search mode), arm is retracted to station search position (either linearly, or along straight line along station exit angle), Z axis moves to the next slot, and procedure continues with arm extend back into station.

- Z axis velocity profile is reset back to original speed and acceleration.
- If requested by **INF** mode, status is displayed. Vacuum valve bit 3 (0x0008) and vacuum sensor bit 2 (0x0004) report whether wafer was secured on the paddle. If **PGO** mode requested wafer and no wafer was found, error prompt "?" is displayed, otherwise prompt ">" is displayed.

When rotating end effector is used, and end effector has to be flipped (as defined with **EEO** command), the rotation is done either at station' retracted position, or during T/Z/K move, depending of rotate-on-the-fly mode (defined with **ROF** command). Station' retracted position has to be chosen so that the end effector can freely rotate while at that position. As an alternative, per-request rotate command, **RUP** and **RDW** can be used to flip the wafer at safe position. For upside-down stations, pick procedure accounts for direction of stroke-up and stroke-down vertical motions.

When an edge gripper is used in place of an end effector, edge gripper sensor is used to check for wafer presence. Value defined by **PGD** command is used as a delay to allow for edge gripper to activate and for sensor to detect correct wafer grip. An additional command, **EDG**, is provided to operate the edge gripper and return its current status.

When a Bernoulli Effect is used in place of an end effector, existing capacitance sensor is used to check for wafer presence. Value defined by **PGD** command is used as a delay to allow for sensor to detect the wafer.

### Operating with Pusher Enabled

When pushing is enabled for a station, following changes occur during **GET** macro operation:

- End effector moves to taught station coordinate, as usual
- Robot **R** axis is moved relatively by a distance specified by **PSS** command. This distance can be positive, negative or zero, as necessary.
- Robot **Z** axis is moved relatively by a distance specified by **PSZ** command. End effector is now in position to start pushing. This distance can be positive, negative or zero, as necessary.
- Arm speed and acceleration are set to the values specified by **\_PS** and **\_PA** command, respectively. If those speeds are not specified, homing speed and acceleration are used.
- Robot **R** axis is moved relatively by a distance specified by **PSD** command, effectively pushing the wafer.
- Robot **R** axis is returned back to the position where pushing started.
- R speed and acceleration are returned to the ones before pushing started
- Robot **Z** axis is returned back to the position before pushing started
- Robot **R** axis is returned back to taught station coordinate
- **GET** operation continues normally (**Z** is elevated etc.)

#### 5.11.4 Standard Place Procedure (a.k.a. PUT)

Standard (atmospheric robot) place procedure consists of following steps:

- No checks are made for wafer presence. It's up to the host to check whether there's a wafer present, and to perform the place command;
- Arm retracts to target station' retracted position, regardless of inline station type;
- Theta, vertical (and track, if present) axes are moved to station' place coordinates for requested slot. If requested by **PGO** mode, the motion of these two (three) axes is synchronized so that all axes arrive at the same time at the target position. If inline station type requires trajectory motion, Theta is moved to inline **T** retracted position instead of station position. If REE option is installed, paddle is rotated to match target station orientation.

- Arm is extended to station **R** coordinate, either radially or over trajectory. If a center offset is defined with **COF** command for that station, **R** coordinate is adjusted with this offset;
- Vacuum valve is closed as to release the wafer from the paddle, with timeout as defined with **PGD** command.
- **Z** axis speed and acceleration are switched to speed and acceleration defined by **PVS** and **PVA** commands. If these values are not specified, homing speed and acceleration are used instead;
- **Z** axis is moved in negative direction by the distance specified with **STR** command;
- Arm gets retracted to station's retracted position (for radial stations), or over trajectory toward station' **T** and **R** inline retracted position (for inline stations).
- **Z** axis velocity profile is reset back to original speed and acceleration.
- If requested by **INF** mode, status is displayed. Vacuum valve bit 3 (0x0008) and vacuum sensor bit 2 (0x0004) report whether wafer was successfully released from the paddle, and prompt ">" is displayed.

When edge gripper is used, edge gripper sensors are used to detect wafer presence. A gripping mechanism is used to hold the wafer in place.

When rotating end effector (flipper) is used, and end effector has to be flipped (as defined with **EEO** command), the rotation is done either at station' retracted position, or during **T/Z/K** move, depending of rotate-on-the-fly mode (defined with **ROF** command). Station' retracted position has to be chosen so that a wafer can freely rotate while at that position. As an alternative, per-request rotate command, **RUP** and **RDW** can be used to flip the wafer at safe position. For upside-down stations, pick procedure accounts for direction of stroke-up and stroke-down vertical motions.

#### 5.11.5 Station-Less In-Line Motion Commands

It is necessary sometimes to perform linear motions without defined stations. In these cases, length of end effector, and cornering radius (parameters normally specified by **ISE** and **ISR** commands) must be specified explicitly, via **MVTP** command. Note that these parameters are not saved or restored.

#### 5.11.6 On-the-Fly Wafer Centering

This feature (also called "Align-On-Fly") uses two sensors placed in front of source or receiving station, measuring center offset from current wafer and a reference wafer.

Measuring sensors must be connected to Input E and Input F pins on External I/O connector. Initial calibration must be performed with a reference wafer placed on robot paddle. Each station has assigned calibration parameters, so single pair of sensors can be used with different wafer sizes. After calibration (**AOFC**) is complete, special pick (**GETC**) and place (**PUTC**) commands can be used to retrieve or place a wafer, measure its offset with regard to the reference wafer. When **GETC** command is used, corresponding **PUTXY** command is used to place the wafer at target station with calculated offsets. When **PUTC** command is used, offsets are applied to station coordinates during final place motion.

Current implementation of on-the-fly centering works for atmospheric robots, using radial motions only.

#### 5.11.7 Station Teaching Example

To illustrate above commands, let's take an example. Let's assume we have a front-end tool, consisting of a robot, two FOUPs placed in front of it, and a process station located at the other side. Let's name stations corresponding to FOUPs A and B, and let's name process station C. Stations A and B will be defined as in-line, and station C will be defined as radial.

For easy teaching, and to use more efficiently the robot, let's assume that Theta axis is set up in such a way that when Theta is at zero position, robot's end effector points towards the FOUPs.

Following session would teach three stations, with comments to commands given at the right margin.

```

>TCH A 25                      Enter teach mode for A, 25 slots
>SPO A T 3109
>SPO A R 10950
>SPO A Z 2950
>PIT A 3937                    10mm (0.3937") slot pitch
>OFS A 120                     0.120" offset
>STR A 200                     0.200" vertical stroke
>PUS A 100                     1"/sec pick-up speed
>PUA A 200                     2"/sec2 pick-up acceleration
>PGD A 300                     300ms vacuum time-out
>IST A 1                       Station capable of radial in
>ISE A 10000                   End effector: 10"
>ISA A 18000                   Exit path angle: 180°
>ISD A 12000                   Exit distance: 12"
>IRT A 4000                    Theta retract: 40°
>IRR A -6000                   Arm retract: -6"
>RPO A -6000                   Same station retracted position
>
>TCH B 25                      Repeat same commands for FOUP B
>SPO B T -3204
>SPO B R 10890
>SPO B Z 2945
>PIT B 3937
>OFS B 120
>STR B 200
>PUS B 100
>PUA B 200
>PGD B 300
>IST B 1
>ISE B 10000
>ISA B 18000
>ISD B 12000
>IRT B -4000
>IRR B -6000
>RPO B -6000
>
>TCH C 1                      Process station with single slot
>SPO C T 17950
>SPO C R 12550
>SPO C Z 4845
>PIT C 100                    Irrelevant for single-slot station
>OFS C 150
>STR C 300
>PUS C 100
>PUA C 200
>PGD C 300
>IST C 0
>
>TCH D 1                      Same station but with negative T axis
>SPO D T -18050
>SPO D R 12550
>SPO D Z 4845
>PIT D 100                    Irrelevant for single-slot station
>OFS D 150
>STR D 300
>PUS D 100
>PUA D 200
>PGD D 300
>IST D 0

```



```

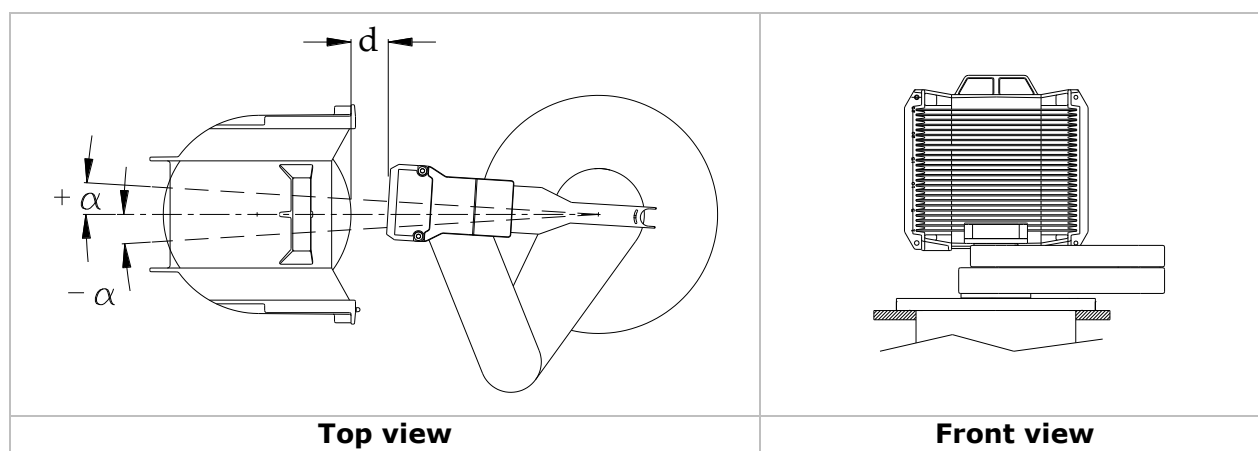
>
>SSP                Save parameters
>

```

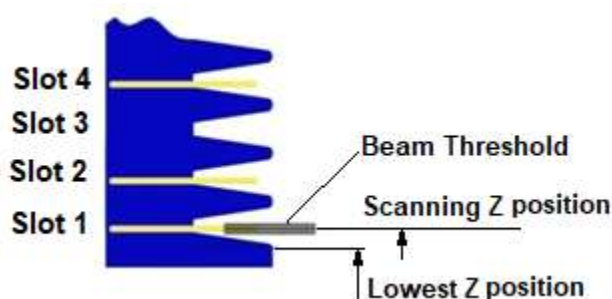
## 5.12 WAFER MAPPING: DESCRIPTION AND COMMANDS USED

Wafer mapping is a process where a sensor travels vertically in front of cassette loaded with wafers, capturing both bottom and top edges of the wafers. Based on these samples, and taught scan positions, a wafer map is produced to facilitate proper wafer processing.

Proper functioning of mapping sensor depends of its correct positioning against the scanned objects. The picture below shows a typical sensor-to-cassette distances and angles. The actual values of the distance ( $d$ ) and the angular orientation ( $\pm \alpha$ ) of the mapping sensor with respect to the wafer center line depend on the type of the sensor (refer to the manufacturer and calibration documentation for details).



The following diagram shows some of parameters being used by the mapping algorithm:



Teaching of scan positions starts with `TCS` command, and must finish with macro `END`. Scan Teaching mode is indicated in a status word with bit 14 (0x4000). Commands, which try to change scanning parameters in normal mode, will be rejected, and error will be indicated.

Thru-beam wafer mapper usually moves very close to the wafers, in order to ensure breaking of the beam with wafer edge. Special provisions are made to prevent wafer damage if there are wafers protruding from the cassette.

Setting of center scanning positions will change some other parameters as well, as follows:

- If left and right scanning coordinates are not set, `_SC` command will copy scan center coordinates to left and right scanning positions, offsetting Theta coordinate by 250 units (2.5°) in negative (for left) and positive (for right) direction.
- Lowest vertical scan position (defined with `SZL` commands) will be set half of slot pitch below first wafer, and highest vertical scan position (defined with `SZH`) will be set half of slot pitch above last wafer. Vertical positions are limited to fit between software limits.



- If vertical coordinate is set with `_SC`, or changed with `SCP` command, this change will be copied to left and right coordinate as well.

One way to teach scan positions is as follows:

- Using `MVA` and `MVR` commands, arm is moved in such a way that wafer sensor is at manufacturer-specified position with regard to the wafer being scanned;
- "Safer" speeds and accelerations are set (e.g. with `HSA` command);
- Wafer mapper is enabled manually, with "OUT 5 1" command (see Chapter 5.8.2, Inputs and Outputs, for detailed information on inputs and outputs);
- Using `MVR Z` command with small increments, arm is moved so that a wafer sensor "sees" the first wafer in cassette. Most mappers have visible indication for that, or "INP 5" can be used instead;
- "INP 5" command must return 1 at this point;
- Arm is lowered so the mapper no longer "sees" the wafer;
- Scan Teaching mode is enabled with "TCS" command;
- Current position is assigned as "center scan position", using `_SC` command;
- Scanning speed is set with `_SS` command;
- Scanner type, maximum wafer thickness, and beam thresholds are set with `_ST`, `_MT` and `_BT` commands;
- Teach mode is exited with `EOT` command.

### 5.12.1 Scanning and Mapping Algorithms

Wafer mapping consist of following steps:

- For reflective wafer mapper (defined with `_ST 0` command): If dual-pass scan is requested, first pass goes up at left scanning coordinates defined by `_SL` and `SLP` commands, and second pass goes down at right scanning coordinates defined by `_SR` and `SRP` commands. If single-pass scan is requested, it is performed at position and direction requested.
- For thru-beam wafer mapper (defined with `_ST 1` command): If dual-pass scan is requested, both passes go up and down at center scanning coordinates defined by `_SC` and `SCP` commands. If single-pass scan is requested, it is performed at position and direction requested.
- Arm retracts to mapped station' retracted position as defined with `RPO` command;
- Theta (and track, if present) axes are moved to station taught scanning position (defined with `SLP`, `SCP` or `SRP` command, depending of whether scanning at left, center or right position). Vertical axis is moved to station' lowest or highest scanning position, as defined with `SZL` and `SZH` commands. If requested by `PGO` mode, the motion of these two (three) axes is synchronized so that all axes arrive at the same time at the target position;
- For first (or only) scan pass, arm is extended to scanning position (defined with `SLP`, `SCP` or `SRP` command, depending of whether scanning at left, center or right position);
- If supported by hardware, wafer mapper is enabled. If wafer mapper reports `ON` state at this position, scan procedure is aborted (it is considered that mapper is sensing an object, so one end of the wafer would be missed);
- Vertical axis velocity is set to scanning speed as defined with `_SS` command;
- Vertical motion starts moving toward other end of scanning range, as defined by `SZL` and `SZH` commands. Using high-precision capture register, each change of

wafer mapper state is recorded. At target vertical position, wafer mapper is disabled.

- If dedicated protrusion sensors are used, scan pass will be terminated, and arm retracted, if any of the sensors is activated during vertical motion;
- 
- If dual-pass scanning was requested, the above process d) .. h) is repeated
- At the end of second (or only) pass, arm retracts to mapped station' retracted position as defined with **RPO** command;
- Data captured during scanning passes is analyzed, and wafer map is produced. If any of scan passes produced odd number of samples, an error is indicated;
- If requested by **INF** mode, status is displayed.

When protruding wafer detection is enabled with safety scan, two additional detection motions are made, at specified distance, before actual wafer mapping is performed.

Mapping algorithm works as follows:

- For each wafer slot, a pocket is defined as follows:
  - bottom end of the pocket equals scan slot position minus first threshold parameter (as defined with **\_BT** command);
  - top end of the pocket equals scan slot position plus second threshold parameter (as defined with **\_BT** command);
- For each station slot, a pair of scanned samples is taken, and wafer thickness is calculated.
- If calculated wafer thickness is more than twice than maximum thickness (defined with **\_MT** command), this wafer is considered to be double slotted;
- If samples pair fails completely inside current wafer pocket, this slot is marked as present;
- If another samples pair falls completely inside already marked pocket, this slot is marked as double slotted;
- If samples pair fails partially inside current wafer pocket, this slot, and next one, are marked as cross slotted.
- If a scanned pair of samples cannot be put into any pocket, this pair is ignored (e.g. if a station is defined with 10 slots, but there are scanned wafers outside these 10 slots, these extra samples will be ignored).

When a cassette is being scanned up and down, if a wafer is sensed during any run, this wafer is considered present. If up- and down passes provide different results about a slot, worse case is taken (i.e. double instead of single, crossed instead of single).

Scanning parameters can be checked with **TCS** command, which reports and prints out a report of wafers scanned, their thickness and deviation from their expected positions.

**DSCN** command provides a full printout of all captured samples. For better accuracy, all measurements and calculations of vertical positions during scanning are made with resolution of 0.0001".

### 5.12.2 Wafer Mapping Example

Comments about the commands used, and their action, are provided with the commands and results below.

```

>NSL B                Check that station exist
25
>PIT B
2500
>
>_ST 0                Set reflective mapper
>_MT 35               Set wafer thickness 0.035"
>_BT 800 800          Set thresholds at 0.080"
>MVA T 12000

```

```

>MVA R 9250
>MVA Z 1450
>HSA
>OUT 5 1          Check wafer mapper
>INP 5
1
>MVR Z -10
>INP 5
0
>OUT 5 0          Disable mapper
>TCS
>_SC B            Designate as center scan for B
>_SS B 500        Set scanning speed of 5"/sec
>EOT              Exit teach mode
>
>SCP B            Check positions: center, left, right
12000,9250,1440
>SLP B
11750,9250,1440
>SRP B
12250,9250,1440
>SZL B            Check lowest and highest scan position
1315
>SZH B
7565              = 1440 + (25-1) * 250
>
>SCN B            Do a test scan
>MAP              and display wafer map
44444444444444444444444444444444000 A lot (all) wafers are cross-slotted
>TSP A            Do a test scan
0:14131; 1:14273; 2:16489; 3:16631; 4:19022; 5:19155; 6:21356 .....
8:24092; 9:24222; 10:26365; 11:26532; 12:28992; 13:29122 .....
16:33903; 17:34070; 18:36583; 19:36717; 20:38905; 21:39099; .....
24:43898; 25:44085; 26:46377; 27:46594; 28:49038; 29:49173; .....
32:54007; 33:54141; 34:56494; 35:56630; 36:58999; 37:59133; .....
40:63922; 41:64105; 42:66424; 43:66559;
44 samples obtained.
Zero pos: 14400 Pitch: 2500 Threshold: 1000
Slot 01: 14131--14273 (142). Middle: 14400 -> -198
Slot 02: 16489--16631 (142). Middle: 16900 -> -340
Slot 03: 19022--19155 (133). Middle: 19400 -> -312
Slot 04: 21356--21533 (177). Middle: 21900 -> -456
Slot 05: 24092--24222 (130). Middle: 24400 -> -243
Slot 06: 26365--26532 (167). Middle: 26900 -> -452
Slot 07: 28992--29122 (130). Middle: 29400 -> -343
Slot 08: 31537--31705 (168). Middle: 31900 -> -279
Slot 09: 33903--34070 (167). Middle: 34400 -> -414
Slot 10: 36583--36717 (134). Middle: 36900 -> -250
Slot 11: 38905--39099 (194). Middle: 39400 -> -398
Slot 12: 41539--41716 (177). Middle: 41900 -> -273
Slot 13: 43898--44085 (187). Middle: 44400 -> -409
Slot 14: 46377--46594 (217). Middle: 46900 -> -415
Slot 15: 49038--49173 (135). Middle: 49400 -> -295
Slot 16: 51475--51643 (168). Middle: 51900 -> -341
Slot 17: 54007--54141 (134). Middle: 54400 -> -326
Slot 18: 56494--56630 (136). Middle: 56900 -> -338
Slot 19: 58999--59133 (134). Middle: 59400 -> -334
Slot 20: 61470--61601 (131). Middle: 61900 -> -365
Slot 21: 63922--64105 (183). Middle: 64400 -> -387
Slot 22: 66424--66559 (135). Middle: 66900 -> -409
Total of 22 wafers found. Adjustments:
Average wafer thickness (0.0001 in): 155
Average offset of slots (0.0001 in): -344
>TCS
>SCP B Z 1405     Lower Z position by 0.035"
>EOT

```

```

>SCN B
>MAP
>11111111111111111111000    Perfect scan
>

```

The above example shows total of 44 samples, corresponding to 22 wafers in a cassette. The wafer in slot 2 is measured to be 0.0142" thick, and is expected to be at Z position 1.6900" ( $= 1.4400 + 0.2500$ ), but its actual center position is found to be 0.0340" below ( $((16489 + 16631)/2 - 16900 = 16560 - 16900 = -340)$ ). On average, wafer thickness is 0.0144", and on average, wafers are 0.0344" too low than expected.

### 5.13 ROBOT SYSTEM WITH STAND-ALONE SINGLE-AXIS SLAVE PREALIGNER

Robot system firmware supports external single axis prealigners connected to the "Slave prealigner interface" connector. When connected and set up, the prealigner adds one additional rotational axis named C to the standard robot axes T, R, and Z. Unit of angular measurement for the prealigner axis is one tenth of an angular degree (that is, 1 unit equals  $0.1^\circ$ , and one full revolution of the chuck or  $360^\circ$  would measure 3600 units). The coordinate increases when the chuck is rotated clockwise.

Rotational speed units are defined as one-tenth angular degree per second ( $0.1^\circ/\text{sec}$ ), and angular acceleration units are defined as one-tenth angular degree per second per second ( $0.1^\circ/\text{sec}^2$ ).

Aligning a wafer always consists of rotating the chuck, collecting the data from the corresponding CCD sensor, and calculating the displacement of the wafer from the chuck center, the angle of the displacement, and the angle of the flat/notch. Depending of the magnitude of the displacement, the next GET command performs single-step or dual-step compensation.

If the displacement is greater than the single step limit parameter, the firmware performs dual step alignment, which first compensates the displacement, then rotates the chuck to position the flat/notch as required, and then the wafer is transferred to the robot end effector.

If the displacement is within the single step limit, the wafer is rotated on the chuck so the flat/notch is at the requested angle, the firmware calculates the T and R positions where the wafer center is located, and then the robot picks up the wafer.

With a wafer placed on prealigner chuck, alignment procedure rotates the wafer, collects samples from CCD, and calculates the displacement and angle of wafer center with regard to chuck. The rotational speed and acceleration are defined with `_MS` and `_MA` commands, respectively. If calculated displacement is smaller than single-step limit (defined with `SSL` command), the wafer is oriented in such a way that subsequent GET operation can pick it up with single motion. If calculated displacement is larger than single-step limit, two-step compensation procedure would be performed by the robot in order to retrieve the wafer oriented.

Several parameters affect how this procedure works. Station name assigned to the prealigner chuck has to be defined with `_PSN` command, so that the firmware will know when an aligned wafer has to be picked up, and displacement compensated for. Distance from the robot center to prealigner chuck has to be defined with `PRD` command, as well as the angle between prealigner CCD sensors line, and robot center.

Three different commands can be used to align a wafer, depending of initial and final state of the wafer to be aligned:

- ALN command expects the wafer to be on the chuck, and leaves it there for subsequent pick command to retrieve it properly
- BAL command expects the wafer to be on robot paddle. It places the wafer on the chuck, aligns it, and takes it back onto the paddle

- PAR command expects the wafer to be on robot paddle. It places the wafer on the chuck and aligns it. If two-step alignment is necessary (calculated displacement is larger than single-step limit), first step of adjustment is done. Robot arm then retracts, with wafer ready to be picked up from the prealigner with single motion.

During wafer rotation, **BAL** and **PAR** commands drop down the arm, and retract it by a distance specified with **\_PSR** command. This would allow for wafer to rotate free without touching parts of end effector.

**WSZ** command specifies wafer size to be aligned, while **\_WT** command defines whether the wafer has a notch, flat(s) or is a square. **FWO** command defines final notch orientation.

In addition to end-effector vacuum valves and sensors, chuck vacuum valve can be turned on with **CVN** command, and off with **CVF** command.

### 5.13.1 Prealigner-Specific Macro Commands

Commands operating on motion and servo parameters are modified as follows:

- Issuing any of these commands without any parameters return only the values for the robot axes T, R, and Z exactly the same way as if there is no prealigner.
- Issuing any of these commands with an axis parameter only or with axis and value parameters work exactly as if there is no prealigner. The only difference is that the commands accept prealigner axis C as an addition to the robot axes T, R, and Z.

```
>SPD
1000 2000 1500
>SPD R
2000
>SPD C
4800
>SPD C 3600
>
```

In the above example, the first command returns the speed setting of the robot axes. The second command returns the speed of the radial axis of the robot R. The third command returns the speed of the prealigner chuck C. The fourth command sets the speed of the prealigner chuck C to 360.0°/sec.

When the host issues **GET** command, the firmware first checks the station parameter. If the station is actually a prealigner, and if there is a wafer, which displacement was calculated, the firmware will compensate for the displacement and will rotate the notch to the requested angle. The compensation and the final orientation of the wafer are done using either single step or dual step alignment depending on the displacement.

### 5.13.2 Sample Commands for Teaching Robot with Single-Axis Prealigner

```
>STA
0000
>MVA T 12000           Move to prealigner chuck
>MVA R 9250
>MVA Z 3100
>TCH P 1               Teach a new station here
>EOT
>STR P 150
>OFS P 70
>PGD P 250
>RPO P -4000
>_PSN P                Designate P as prealigner station
>_PSR 4000             Safe retract distance
>_PSD 16250            Chuck-to-robot distance
>_PRA 18000            CCD-to-robot angle
>_FWO 0                Final wafer orientation - along paddle
>_SSL 500              If horse-shoe is big enough
>
```

```

>GET A 1           Pick a wafer to be aligned
>BAL              Place on P, align, pick back
>...000C
>PUT C            Place in process station, aligned
>

```

## 5.14 ROBOT SYSTEM WITH STAND-ALONE THREE-AXIS SLAVE PREALIGNER

WHR robots and wafer prealigners have an option where a prealigner can be daisy-chained to the robot, through robot's Slave interface. This mode is activated through a configuration change in Robot System Configuration File, as described in Chapter 7.3, Robot Configuration and Firmware files. Limited set of prealigner commands are available thru robot command interface. For parameters or commands not provided with this release, direct connection to the aligner is recommended.

For additional information regarding Logosol Prealigner Series, refer to corresponding documentation.

### 5.14.1 Axes Naming

When working as a stand-alone device, not daisy-chained to the robot, but using a separate communication line, the prealigner should use its native axes names as follow: Upper or lower letters T for chuck rotation, R for chuck horizontal translation and Z for chuck vertical translation.

When working as a slave device, daisy-chained to the robot, the prealigner axes are addressed by the upper or lower letters as follow: Upper or lower letters C for chuck rotation, X for chuck horizontal translation and V for chuck vertical translation. Macro commands expecting an axis name will be routed for execution according to axis name capitalization, as follows:

- Upper- or lower-case axes names **T**, **R**, **Z** refer to robot axes.
- Upper- or lower-case axes names **c**, **x**, **v** refer to prealigner axes.

```

>MVA T 1000
>MVR V 1000
>

```

First command will be executed by the robot, and it will move robot Theta axis to absolute position 1000, or 10°. Second command will be executed by the prealigner, and will move prealigner Z axis relatively 1000 units, or 0.1" forward.

## 5.15 CONFIGURATION OPTIONS FOR SINGLE-ARM ROBOT

To accommodate for different customer needs, various pre-defined hardware configurations are possible through an entry in robot's parameters file, `params.ini`. For dual-arm robots, this file might be called `ParamsD.ini`.

```

[General]
.....
ConfigOption=0

```

### 5.15.1 Standard Configuration

Default configuration, "ConfigOption=0" is for atmospheric robot, with one or two vacuum lines. This configuration provides support for Rotating End Effector, which is enabled using following additional parameters in `params.ini` file:

Parameter Name	Parameter Value	Description
<i>ConfigOption</i>	0	Standard atmospheric robot

Parameter Name	Parameter Value	Description
<i>CommaDelimiter</i>	44	ASCII code of character to separate values when multiple numbers are reported. Common values are 44 (comma) and 32 (space)
<i>RotEEOption</i>	0	(default) No REE installed
<i>StaVacOut</i>	250	Timeout in ms for vacuum operations during homing
<i>BBOption</i>	0	If defined, configures external battery support option - time in ms for operating when mains power fails.
<i>InitialCSTP</i> <i>CStpActLevel</i>	41 0	Defines system input number for soft (controlled) stop function. When input is set to the active level, currently executed motion command is gracefully stopped.
<i>WaferSensorInst</i>	42	Defines input number where wafer position sensor is connected, and sensor used by <b>MTB</b> command. If set to 1, <b>SensorC</b> is used.

Table 5-6: Robot configuration for single-arm robot

More detailed information about other fields in this file can be found in Chapter 7.3.2, Generic and Motion Control Parameters File (params.ini).

### 5.15.2 Configuration for Rotating End Effector (REE)

Rotating end effector can be installed on primary, secondary, or both paddles on single- and dual-arm robots. When this option is activated, corresponding hardware inputs and outputs must be properly defined. Following parameters affect REE operations:

Parameter Name	Parameter Value	Description
<i>ConfigOption</i>	0	Standard atmospheric robot
<i>RotEEOption</i>	3	Single REE on primary paddle
	12	Single REE on secondary paddle
	15	Dual REE on both paddles
<i>RotEETOut1</i>	2000	Timeout in ms for REE to rotate fully, end-to-end
<i>RotEETOut2</i>	200	Timeout in ms for REE to start moving
<i>ReeHomeFlipOrder</i>	0	0 (default): Arm axis is moved to defined HFP position, followed by T, Z and K axes; 1: T, Z, K axes are moved to defined HFP position, followed by arm axis.
<i>DisableReeHome</i>	0	0 (default): REE will be homed as part of HOM procedure; 1: REE will not be homed as part of HOM procedure.
<i>NoReePowerCheck</i>	0	0 (default): Robot motions (except HOM) will be disabled if REE is not powered up 1: Robot motions will be allowed to even if REE is not powered up. REE will be powered and homed at station retract position on first occasion.

Table 5-7 Robot configuration for rotating end effector

**WARNING:** Leaving REE not powered / homed can be dangerous when robot is moved around, or toward a station.

### 5.15.3 Configuration for Edge Gripper

Three standard types of edge-gripping end effectors are supported, as follows:



Parameter Name	Parameter Value	Description
<i>ConfigOption</i>	2	Standard edge gripper
	3	Bernoulli effect edge gripper
	4	Edge-gripper with wafer presence sensor
<i>GripperConfig</i>	114	Variable defining switches configuration, active levels and other settings, described below

**Table 5-8: Robot Configuration for Edge Gripper paddles**

These configurations differ in the way pick and place operations are performed, compared with standard atmospheric robot, and have additional states for correct or wrong grip. Both configurations use gripper sensors as wafer presence sensor.

*GripperConfig* variable is formed using following rules:

- Four gripper states (open, closed with wafer, closed without wafer, and error state) are assigned four distinct values from zero to three, representing the state of two gripper sensors, as follows:
  - 0: SW2=0, SW1=0
  - 1: SW2=0, SW1=1
  - 2: SW2=1, SW1=0
  - 3: SW2=1, SW1=1
- Active level for sensors is defined as either 0 or 1
- Active level for gripper is defined as either 0 or 1
- Single 10-bit number is formed, as follows:
  - $512 * (\text{sensor active level}) + 256 * (\text{gripper active level}) + 64 * (\text{error grip value}) + 16 * (\text{open grip value}) + 4 * (\text{closed grip full value}) + 1 * (\text{closed grip empty value})$
  - If gripper does not have plunger or other activation device (e.g. passive pocket end effector), 1024 is to be added to above number;
  - This number is assigned to *GriperConfig* parameter.

If this value is not defined, or invalid (e.g. two gripper states are assigned same value), following default values are used:

Empty = 2 (SW2=1, SW1=0)

Full = 0 (SW2=0, SW1=0)

Open = 3 (SW2=1, SW1=1)

Error = 1 (SW2=0, SW1=1)

Sensor active = 0

Gripper activate = 0

$GripperConfig = 512 * 0 + 256 * 0 + 64 * 1 + 16 * 3 + 4 * 0 + 1 * 2 = 114$

This value corresponds to a standard edge gripper. Custom edge grippers might require different configuration and motion file, and are subject of custom firmware implementation.



## 5.16 TEACH-PENDANT OPERATION

Teach Pendant allows for greater flexibility in setting up of WHR into a customer's equipment. Teach Pendant allows for easy movement, teaching, moving and deleting stations.

One of interface panel's connector is configured for Teach Pendant. Standard hand-held terminal keypad is replaced with customized template, giving easy access to common wafer handling motion commands. Teach pendant simplifies the process of teaching stations and moving robot arm inside working envelope. Teach pendant can be used also for command entry, but limited screen size (four lines by twenty characters) and non-standard keyboard makes this harder than usual.



### 5.16.1 Modes of Operation

Teach pendant can be in one of four states:

- Idle (host control)
- Mode change
- Terminal
- Teach pendant

Changing between different states is done by pressing <Ctrl-T> key (see below for explanation of keypad operation). For as long as the hand held terminal is in Terminal or Teach pendant state, host communication is blocked, and status "BADA" will be reported in response to every command.

Recent firmware releases block host access to the robot when terminal is in any other mode than idle.

#### Idle Screen

After power-up or in idle state, the Teach Pendant displays "Idle (host control)" screen:

```
<Ctrl-T> to activate
```

If Teach Pendant mode was changed due to host input, following screen will be displayed to inform the user:

```
Deactivated by host
<Ctrl-T> to activate
```

#### Mode Change Screen

"Mode change" screen is invoked upon pressing <Ctrl-T> from Teach Pendant keyboard. If function keys have not been reprogrammed, <F4> function key can be used to enter "Mode change" screen as well.

```
<T> Console here
<E> Teach Pendant
<Q> Back to host
Your choice: _
```

Pressing any of <T>, <E> or <Q> keys on teach Pendant will result in changing states as described above. Pressing any other key has no effect. Mode change screen does halt

communication with the host, and system should not be left in this state for prolonged periods of time.

### Terminal Screen

Terminal screen can be used to enter ordinary commands in the same manner as commands entered through Host interface:

```
1350,6500,975
>STA
0400
>
```

All entered commands and results scroll the output of the screen. If a command is expected, a prompt (“>” or “?”) is displayed on last line. The screenshot above shows how the screen would look like after commands **CPO** and **STA**.

No special handling of terminal output is performed while Teach Pendant is in “Terminal screen” mode. There are no provisions for scrolling back into terminal output. Therefore, commands which output long and/or multiple lines of information (eg **VER**) will look crippled on Teach Pendant’s screen.

### Teach Pendant Screen

Teach Pendant’s primary use is during teaching process, and it is optimized for this function. If firmware loaded into the robot supports Teach Pendant interface, and state is changed to “Teach Pendant”, following screen is displayed:

```
T:+01350  a01 -06000
R:+06500  I0:0 +12500
Z:+00975  O1:1 +00375
```

Information on the screen is divided on several fields:

- Leftmost column shows names of axes (T, R and Z). Uppercase letters mean that corresponding axis is under closed servo loop control; lowercase letter mean that corresponding axis has its servo loop open. If there is a track attached to the robot, third line could show 'K' to denote that track axis is shown and controlled instead of Z axis.
- Next to each axis name, current axis coordinates are displayed
- Next column contains currently active station name (in our example, “a”), currently active slot number for this station (in our example, 1), input state of a chosen input (Input number zero has a state of zero), and current state of a chosen output (output number one is in ON state). Input and output numbers, and output state can be changed with a corresponding keypad command (see below)
- Last column contains the coordinates of currently active station
- Last line is used for displaying status and confirmation messages.

#### 5.16.2 Keypad Description

Teach Pendant keyboard has forty-five keys (nine rows five keys each), arranged in alphabetical order, with dedicated number keys.

Some of the keys on the keypad (<Shift> and <Ctrl>) do not generate any commands, instead, they change meaning of keys pressed immediately after them. Those keys, however, must be pressed independently, as any other key. Pressing any of those keys twice cancels its action.

Each letter key generates corresponding upper-case letter. If lower case letters are needed (e.g. as a station name), corresponding letter key must be preceded by <Ctrl><Shift> keys.

Pressing <Shift> key before any other key generates the character shown in the top half of the corresponding key, if any. Function keys <F1>, <F2>, ... <F5> have corresponding keys <F6>, <F7>, ... <F10>.

Pressing <Ctrl> key before alpha keys generates corresponding **ASCII** control character. The only command supported right now is <Ctrl><T> which changes screen states.

Typing errors made during entry of commands in "Terminal Emulation" mode can be corrected with <BackSp> key (that is, subsequent pressing of <Shift> and <Space> keys).

### Special Key Mapping

While in "Teach Pendant" state, most of alpha keys on the keyboard have special meaning, as described in following table. Detailed description of each command is given in the next chapter, "Keypad command description"

Key	Command	Description
A	<T+>	Jogs Theta axis in positive direction (clockwise)
B	<T->	Jogs Theta axis in negative direction (counterclockwise)
+	<T Speed Decr>	Increase T axis jogging speed
(	<T Speed Incr>	Decrease T axis jogging speed
C	<Input select>	Changes currently displayed input port number
D	<Output select>	Changes currently displayed output port number
E	<Output toggle>	Toggles state of currently chosen output port
F	<R+>	Jogs R axis in positive direction (extend)
G	<R->	Jogs R axis in negative direction (retract)
-	<R Speed Decr>	Increase R axis jogging speed
!	<R Speed Ince>	Decrease R axis jogging speed
H	<StationGet>	Perform a GET operation from current station/slot
I	<StationPut>	Perform a PUT operation to current station/slot
J	<MoveStation>	Moves to currently selected station/slot (MTS)
K	<Z+>	Jogs Z axis in positive direction (up)
L	<Z->	Jogs R axis in negative direction (down)
/	<Z Speed Decr>	Increase Z axis jogging speed
&	<Z Speed Ince>	Decrease Z axis jogging speed
:	<ToggleT>	Toggles servo loop on Theta axis ON/OFF (SON/SOF T)
;	<ToggleR>	Toggles servo loop on R axis ON/OFF (SON/SOF R)
?	<ToggleZ>	Toggles servo loop on Z axis ON/OFF (SON/SOF Z)
P	<Jog Speed +>	Increases jogging speed, up to the working speed
Q	<Jog Speed ->	Decreases jogging speed, down to the homing speed
R	<Restore stations>	Restores station parameters from file (RSP)
S	<Save stations>	Saves station parameters to file (SSP)
T	<Next station>	Selects next (or first) station to become current station
U	<Next slot>	Selects next (or first) slot of current station

Key	Command	Description
^	<Select station>	Select new or existing station as a current station
\$	<Select slot>	Selects new current slot for current station
V	<Store>	Stores current coordinates as a new station
W	<Delete>	Deletes currently selected station from list of stations
X	<Toggle R1/R2>	Toggles arms R1/R2 in dual-arm configuration
Y	<Toggle Z/K>	Toggles between Z and K axes on track systems
Z	<Stop axis motion>	Stops currently active motion
.	<Speed/Step Mode>	Toggles jogging mode between speed and step setting
Esc	<Abort all>	Aborts current motion and kills all pending motion commands

Table 5-9: Default Hand-Held Terminal Key Assignment

### Command Description and Actions

#### Jogging Speed Select

Pressing <JogSpeed+> and <JogSpeed-> keys will cause subsequent speed for currently jogged axis to be changed. Upon entering Teach Terminal mode, all axes speed will be reset to half of their working speeds, with five levels above and below this speed. Minimum possible speed for any axis is homing speed, and maximum speed is working speed. If an attempt is made to choose speed above those limits, an audible signal will be emitted from the terminal.

Due to limited screen resolution, current jogging speed is not displayed on the screen.



**WARNING:** Care has to be taken when choosing speeds – improper motion with high speed can cause damage to the robot or equipment.

#### Jogging in Speed or Step Mode

Pressing <Speed/Step Mode> toggles between two modes of operation when any of jog keys is pressed:

- When in Speed mode, axis keeps moving as long as the jog key is kept pressed. When jog key is released, axis motion stops.
- When in Step mode, each jog key pressing causes the axis to move forward or backward by specified distance.

Speed and acceleration for jogging motions are determined by current speed/acceleration set.

```
T:+01350 a05 -06000
R:+06500 I1:1 +12500
K:+00385 O2:1 +00375
New jog mode: step
```

#### Axes Jogging

Pressing axis jogging keys <T+>, <T->, <R+>, <R->, <Z+>, <Z-> will cause single or continuous movement of designated axis with its current jogging speed. Single pressing of jogging key will cause a movement equivalent to approx. 0.1sec with current jogging

speed. Continuous pressing of the key will result in continuous motion with current jogging speed.



**WARNING:** Care has to be taken when jogging – improper motion with high speed can cause damage to the robot or another equipment.

If software limits are violated during jogging motion, teach mode will be terminated, and standard error procedure will be executed. Please refer to `_FL` and `_RL` commands description for details about software limits.

Current coordinates of axes will be updated during jogging command.

### Toggle Z/K and R1/R2 Axes

On system equipped with a linear track, `<Toggle Z/K>` key is used to toggle between controlling/Reporting of vertical axis Z, and linear track K. Third line will show either Z or K depending of the current state of this key. Jog speed and axis jogging keys will control either Z or K. Care should be taken when jogging as to make sure the necessary axis is being controller.

```
T:+01350  a05 -06000
R:+06500  I1:1 +12500
K:+00385  O2:1 +00375
```

On dual-arm configurations, `<Toggle R1/R2>` key is used to toggle between controlling/Reporting of primary arm R1, and secondary arm R2. Second line will show either R1 or R2 depending of the current state of this key. Jog speed and axis jogging keys will control either R1 or R2. Care should be taken when jogging as to make sure the necessary axis is being controller.

```
T:+01350  a05 -06000
R2+06500  I1:1 +11875
Z:+01225  O1:1 +00375
```

### Input Select

When an `<Input Select>` key is pressed, next configured digital input is selected. If more than 10 inputs are configured, only first ten inputs are displayed and accessible with this command. Each change of input state will be updated on the Teach Pendant' screen.

```
T:+01350  a05 -06000
R:+06500  I1:1 +12500
Z:+01225  O1:1 +00375
```

For a complete list of inputs, please refer to Chapter 5.8.2, Inputs and Outputs.

### Output Select and Output Toggle

When an `<Output Select>` key is pressed, next configured digital output is selected. If more than 10 output are configured, only first ten output are displayed and accessible with this command. Each `<Output Toggle>` key will cause change of output state, which will be updated on the Teach Pendant' screen

```
T:+01350  a05 -06000
R:+06500  I1:1 +12500
Z:+01225  O1:1 +00375
```

For a complete list of outputs, please refer to Chapter 5.8.2, Inputs and Outputs.



**WARNING:** Some outputs can be configured in such a way as to control critical safety and operational features of the system (turning power on or off, turning brakes on or off, etc.).

## Station Teaching and Operations

Station teaching is one of the basic tasks for Teach Pendant terminal. Several commands are available for this process:

**<StationStore>** - Stores current positions as coordinates of the currently displayed station. Confirmation is required as a precaution against non-intentional overwriting of existing station. This command is similar to combination of **TCH** and **EOT** commands.

```
T:+01350  a05 -06000
R:+06500  I1:1 +12500
Z:+01225  O1:1 +00375
Store station a? _
```

**<StationDelete>** - This command deletes currently selected station from the list of stations and makes next station current, if any. Confirmation is required as a precaution against non-intentional pressing of this key.

```
T:+01350  b01 +06000
R:+06500  I1:1 +10500
Z:+01225  O1:1 +00075
Delete station b? _
```

**<StationSelect>** - This command is used for adding new station, or quick access to a particular station. If station name entered is not found in the list of configured stations, a process for entering new station parameters is started.

```
T:+01350  b01 +06000
R:+06500  I1:1 +10500
Z:+01225  O1:1 +00075
Select station _
```

Enter the name of existing or new station here, or press <Esc> to cancel the operation. If selected station does not exist, one more question will appear:

```
T:+01350  b01 +06000
R:+06500  I1:1 +10500
Z:+01225  O1:1 +00075
Number of slots: __
```

Enter the number of slots for new station, or press <Esc> to cancel the operation.

**<StationMove>** - Moves the robot to currently selected station. This command is similar to **MTC** command (see "Software Manual").

**<NextStation>** - This command cycles through all configured stations and displays the name and coordinates of currently selected station. After last station is selected, "next" station becomes first defined station. Stations are listed in the order of adding them to the system.

**<NextSlot>** - This command cycles through all slots in a defined station. After last slot is selected, slot number one will be the next selected slot.

**<SelectSlot>** - This command is used for quick selection of a desired slot in a defined station.

```
T:+01350  b01 +06000
R:+06500  I1:1 +10500
Z:+01225  O1:1 +00075
Current slot: __
```

Enter the desired slot to become current slot, or press <Esc> to cancel the operation.

### Station GET/PUT

The **<StationGet>** and **<StationPut>** commands execute a sequence of motions such that a wafer is taken out from or put into a station (cassette, process point etc). When one of these commands is invoked, a confirmation message is displayed to guard against involuntary pressing of the keys:

```
T:+01350  b01 +06000
R:+06500  I1:1 +10500
Z:+01225  O1:1 +00075
Get from station b _
```

Either <Y> or <y> has to be pressed in order for **GET/PUT** operation to be executed. Pressing any other key, or <Esc> will cancel the operation.

### Servo Control Commands

Sometimes it is necessary to disable servo loop on a particular axis (e.g. to facilitate manual move of the arm to a desired position). Servo toggle commands **<ToggleT>**, **<ToggleR>** and **<ToggleZ>** toggle servo loop on and off for a particular axis, and indicate the result on the screen. Upper-cased axis name means that this axis is under servo control; lower-cased axis name means that this axis is free for movement.

```
t:+01350  b01 +06000
r:+06500  I1:1 +10500
Z:+01225  O1:1 +00075
Get from station b _
```

**NOTE 1:** Due to robot' design, arm axis manual movement requires that both T and R axes servo loops to be opened.

**NOTE 2:** Even with servo loop opened for axis Z, it will stay on place due to brake being engaged. If manual lifting of Z axis is required, both servo loop has to be off, and Manual Brake Release button has to be pressed. See Chapter 3.1.1, Emergency Brake Release/Emergency Stop connection, for more information on Manual Brake Release button).



**WARNING:** Pressing Manual Brake Release while servo loop is off will cause Z axis to move downwards if not supported! Make sure this motion is safe and there are no obstacles to be hit.

### Motion Control Commands

**<Stop Motion>** and **<Abort>** commands are used when an motion command has to be interrupted. While **<Stop Motion>** command stops just currently executed motion,



**<Abort>** command also abort all running macros, and returns the Teach Pendant back to "Idle screen" state.

One possible use of **<Abort>** command could be if e.g. **GET** operation seems to be unnecessary, and has to be stopped. After executing the **<Abort>** command, system will revert back to host for further commands.

### 5.16.3 Additional Operations

Functional keys **<F1>**, **<F2>**, ... **<F10>** are free for user-supplied commands. Please refer to supplier for additional information about how to activate those commands.

### 5.16.4 Custom Key Assignment and Velocity Sets

Custom key assignments and velocity/step settings can be stored in `hhtparam.ini` file. If this file is present, it is read during firmware startup, and values listed in that file override the default key assignments listed above. Full description of this file is available in Chapter 7.3.6, Hand-Held Terminal Key Assignments File (`hhtparam.ini`).

## 5.17 TROUBLESHOOTING

Being a complex system, WHR systems need some time for the user to get used to. From the other hand, these systems are designed for easy integration into an end product, so even when unexpected condition occurs, user interaction is kept to a minimum level.

There are several types of error conditions which can occur to the robot system:

- Motion errors: Motion is interrupted due to some reason
- Command errors: Command is not executed (or not executed as expected) due to some reason
- User-requested stop: Emergency stop, or Break command

### 5.17.1 Motion Errors

When an axis is requested to move from its current position to a new position, with specified velocity and acceleration, the firmware controls the motor in such a way that at every given moment, the difference between expected and actual positions is kept to a minimum. If for some reason this difference cannot be kept, a motion error occurs.

Servo loop can fail to close (**SON** command fails immediately, or after a short time) due to following reasons:

- Improper configuration of motor safety files;
- Emergency stop circuit is engaged. In that case, **SON** command usually returns a status of FFFF. State of emergency stop can be checked by examining corresponding input (usually 6) with **INP** command, please refer to Chapter 5.8.2, Inputs and Outputs, for details.
- Incorrect PID parameters prevent **SON** to turn servo loop on requested (or all axes).
- Faulty cabling prevents proper communication between the controller and axis motor.

There are several cases where the position cannot be maintained:

- Requested acceleration (and deceleration) is too high, while motor current limit (**\_CL**) is too low, and axis stops during acceleration or deceleration phases.
- Requested speed is too high, while position error limit (**\_EL**) is too low, and axis stops when trying to reach desired speed.
- An axis hits an object, or hard stop, causing servo controller to shut down the axis. Safety features built into the motion controller shut down all other axes as well.



### 5.17.2 Command Errors

As already mentioned in Chapter 5.2, Command Syntax, Prompts and Results, there are several reasons when a command cannot be executed, or can terminate prematurely:

- Syntax errors: Too many (or not enough) parameters are supplied, parameters are not of the correct type (e.g. character specified where a number is expected), or command name cannot be recognized;
- Invalid parameters: Parameter supplied to the command is not valid (e.g. requested station name does not exist, or value of an option is not valid);
- Invalid state: Command has correct parameters, but cannot be executed at current state of the system (e.g. motion command requested with servo loop open, or station parameter changed with system not in teach mode).

Some commands indicate that additional information is available by setting Bit 2, "Previous error returned an error", in status word. This error can be retrieved with **PER** command.

Another way to troubleshoot parameter errors by enabling INF mode 5, and examining EERR codes returned by failing commands.

### 5.17.3 Diagnostic Commands

Successful troubleshooting always begins with a question, "what the system was instructed to do". If **INF** mode is set to a non-zero value, every motion error will report current status, followed by a question mark. If **INF** mode is set to zero, no indication is made on motion errors, as it is expected from the host to poll for status.

After this question is answered, there are two commands which can help out diagnosing the problem:

**DIAG** command is oriented more toward human, and it can report a human-readable status of the system at the moment the error occurred and current system state. A simple output of command is shown below:

```
>GET A 10
>0411
?DIAG
LAST FAULT Status:
Stop Reason    -> Axis R:PositionERROR,
0:Axis T       -> Stopped,           HomeT  ON,   MUMS
1:Axis R       -> PositionERROR,    HomeR  ON,   MUMS
2:Axis Z       -> Stopped,           HomeZ  OFF,  MUMS, Brake ENGAGED
MainPOWER: LINE; BackupENABLE: OFF; ScanSOURCE: NONE
ArmPOWER:OFF; SensorA: OFF; SensorB: OFF; SensorC: ON ; SensorD: ON
OutputA: ON ; OutputB: ON ; OutputC: OFF; OutputD: OFF; OutputE: OFF
Current State:
SystemSTATE    -> ACTIVE !!! SupervisorSTOP !!!
Stop Reason    -> Axis R:PositionERROR,
0:Axis T       -> Stopped,           HomeT  ON,   MUMS
1:Axis R       -> PositionERROR,    HomeR  ON,   MUMS
2:Axis Z       -> Stopped,           HomeZ  OFF,  MUMS, Brake ENGAGED
MainPOWER: LINE; BackupENABLE: OFF; ScanSOURCE: NONE
ArmPOWER:DIS; SensorA: OFF; SensorB: OFF; SensorC: ON ; SensorD: ON
OutputA: DIS; OutputB: DIS; OutputC: DIS; OutputD: DIS; OutputE: OFF
>
```

In the above case, it's Axis R, which caused the motion error, due to position error. If emergency stop circuit was engaged, first few lines would look like:

```
>GET A 5
>0411
?DIAG
LAST FAULT Status:
Stop Reason    -> EmergencySTOP, AxisZ: PositionERROR,
0:Axis T       -> Stopped,           HomeT  ON,   MUMS
```

```

1:Axis R      -> Stopped,           HomeR  ON,  MUMS
2:Axis Z      -> PositionERROR, HomeZ  OFF, MUMS, Brake ENGAGED
... ..
>

```

First line shows a one-line summary of axis states, followed by detailed axis state (if available). Each axis can be in one of following states:

State	Description
ServoON	Axis is OK, servo loop is closed, no error condition
AxisOFF	Motor power is OFF, servo loop is open
ServoOFF	Motor power is ON, servo loop is open
PositionERROR	Servo loop broken due to position error
Stopped	Axis stopped due to emergency stop, over-current, or other reason including another axis error
Smart Stop	Axis stopped in controlled manner due to emergency condition on other axes
EncoderERROR	Cabling problem between servo controller and motor
ErrHALL	Controller configuration error
ErrEEPROM	Controller configuration error

On the same line with servo states, home switches information, brake information, and motor configurations are shown.

After axes info is displayed, current state of controller inputs (*SensorA* through *SensorD*) and outputs (*ArmPower*, *OutputA* through *OutputE*), current scanning state (*ScanSOURCE*) is shown. If battery backup option is installed, and mains power is down, *MainPOWER* and *BackupENABLE* states will report meaningful information as well.

The other diagnostic command, **EST**, provides much more information but in much more compressed form. The primary purpose of this information is to get logged into *ErrorLog.log* file. This info could be requested also by manufacturer service personnel.

```

>EST
Error: EC=11 EA=2 RC=8 EL=-1 at 2020-1-1 0:28:22
STA(axis):  T=834884A8 R=834884A8 Z=834806A8
HWS(devic): T=74011951/04 R=74011951/04 Z=74011959/04 331=00C08082/00C08082
POS(units): T=16890 R=-7 Z=1523
ERR(enc.c): T=-7 R=2 Z=32
ENC(enc.c): T=3136907 R=972 Z=534625
TMP(Deg.C): T=29/30 R=32/30 Z=28/32 Vdc=1674:78 1668:78 1658:77 Rob=78
T0=0:;0,0,0,0,0,Res=0449;T1=8407
T0=9935:SON;_NA,0,_NA,_NA,_NA,Res=0000;T1=-2
T0=16275:MVA;2,555,_NA,_NA,_NA,Res=0000;T1=-1
>

```

First line of **EST** command contains error number, reason code (RC), axis registering the error (EA), and a timestamp. Since the system does not keep track of the time, this time stamp could be incorrect.

As of current firmware release, most common error codes are as follows:

Error (EC)	Description
7	Software limit violated
11	Servo Loop Broken - maximum position error exceeded
12	Lost communication (internal)
16	Guard condition met (wafer lost during transport)
18	Emergency Stop Activated

Error (EC)	Description
22	Motor Current Overload
23	Servo Loop Broken - safety, or motor power off
24	Encoder motion during power-up / servo-ON
25	Invalid Motion Mode (PVAT)
26	Motor Voltage Low
27	Communication error with AEs
28	No Mains Power

**STA** and **HWS** lines contain high- and low-level axis status. **POS** line contains axes position at the time error was handled, **ERR** line contains position (following) error in encoder counts at that moment, while **ENC** contain motor encoder positions. Additional lines may be present, depending on firmware release.

Logosol Control Center contains an applet which can help in diagnosing current error, and browsing through error log saved onto controller.

On robot systems with absolute encoder support, **DIAG** command lists also absolute encoder state, and absolute encoder battery support state, as in the following example:

```
?diag
LAST FAULT Status:
Stop Reason    -> User break
0:Axis T       -> ServoON,      HomeT  OFF, PABS-Healthy
1:Axis R       -> ServoON,      HomeR  ON,  PABS-Healthy
2:Axis Z       -> ServoON,      HomeZ  ON,  PABS-Healthy, Brake RELEASED
4:Elev1        -> ServoON,      HomeV1 OFF, MUMS, Brake RELEASED
MainPOWER: LINE; BackupENABLE: OFF; ScanSOURCE: NONE
ArmPOWER:OFF; SensorA: OFF; SensorB: OFF; SensorC: OFF; SensorD: OFF
OutputA: OFF; OutputB: OFF; OutputC: OFF; OutputD: OFF; OutputE: FLT
Current State:
SystemSTATE    -> ACTIVE
0:Axis T       -> ServoON,      HomeT  OFF, PABS-Healthy
1:Axis R       -> ServoON,      HomeR  ON,  PABS-Healthy
2:Axis Z       -> ServoON,      HomeZ  ON,  PABS-Healthy, Brake RELEASED
4:Elev1        -> ServoON,      HomeV1 OFF, MUMS, Brake RELEASED
MainPOWER: LINE; BackupENABLE: OFF; ScanSOURCE: NONE
ArmPOWER:OFF; SensorA: OFF; SensorB: OFF; SensorC: OFF; SensorD: OFF
OutputA: OFF; OutputB: OFF; OutputC: OFF; OutputD: OFF; OutputE: FLT
>
```

Additional fields indicate type and state of absolute encoder, as follows:

Encoder Type	Description
PABS	Panasonic Absolute Encoder A4
P6ABS	Panasonic Absolute Encoder A6, 23/17bit
TABS	Tamagawa Absolute Encoder
T23AE	Tamagawa Absolute Encoder, TBLi4, 23/17bit
NABS	Nikon Absolute Encoder
LBiSS	LBiSS Absolute Encoder (track systems)

State	Description
-Healthy	Encoder valid, internal battery OK
-ErrABS	Absolute encode in error state.
-BatteryLOW	Encoder internal battery is getting low, and might not hold encoder value if power is turned off

State	Description
-BatteryERROR	Encoder internal battery is low or missing.

## 8 LOGOSOL WAFER HANDLING CONTROL CENTER

Logosol Wafer Handling Control Center (LCC for short) is a powerful utility intended for ease of use with Wafer Handling robots systems and prealigners. It contains several modules:

- Terminal emulator
- Station parameters editor
- System state, Axes positions, Input/Outputs state and control
- Error state browser
- Commands sequencer
- File transfer module

**NOTE:** The term “controller” will be used throughout this chapter when referring to either robot or prealigner connected to Logosol Control Center.

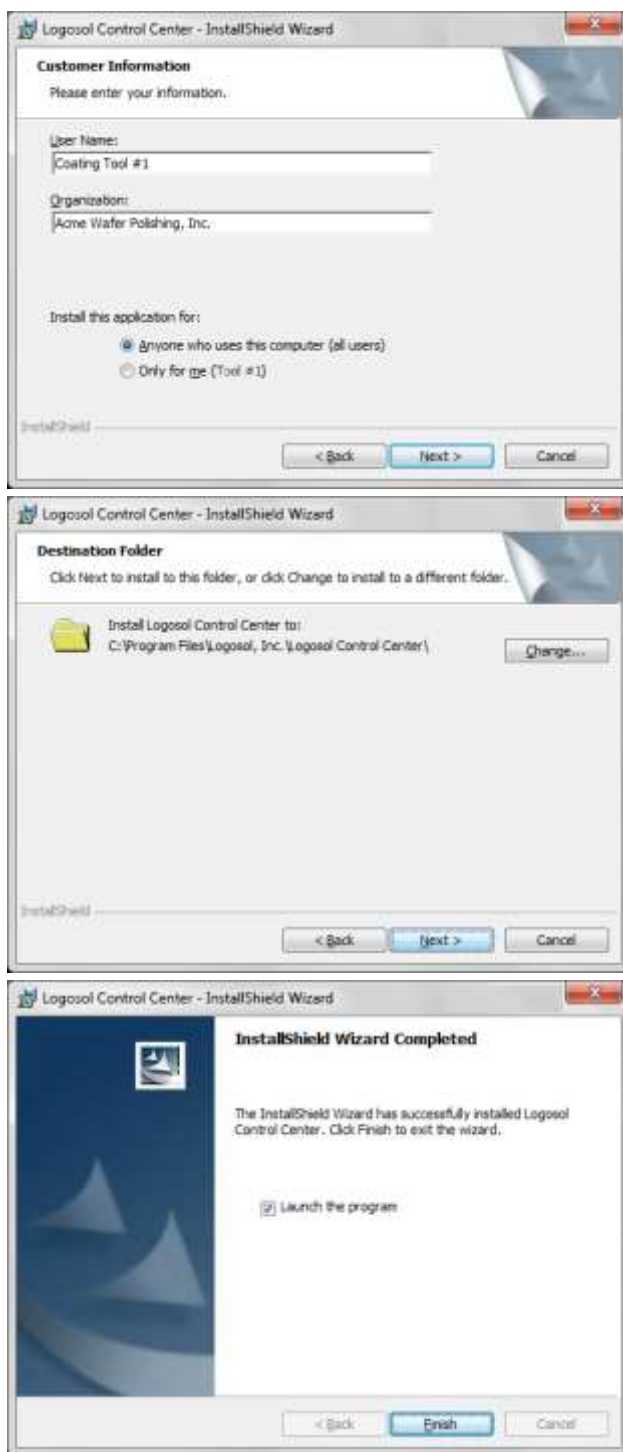
### 8.1 INSTALLATION AND INITIAL SET-UP

Before starting installation, make sure that all previous versions of Logosol Control Center (or similar utilities) are closed, and un-installed. Use “Control Panel -> Add or Remove programs” Windows wizard, or shortcut placed on the “Start Menu” to remove it.



Logosol Control Center (LCC) is installed using Windows Installer file called LccSetup.exe. After starting the setup program RccSetup.exe, a series of installation dialogs appear.

Press “Next” to proceed to next screen, or “Cancel” to terminate installation.



Change "User Name" and "Organization", if necessary. Default values are taken from Windows Installation. Select which users you would like to provide with shortcut icons for this product. Click "Next" to continue

If you want to change the installation folder, click on "Change..." then select new installation folder. Click "Next" to accept currently shown installation folder and continue.

When installation is complete, the confirmation dialog will be displayed. Click on "Finish" to dismiss the dialog. LCC is installed!

Following files and folders are created into installation directory ("C:\Program Files\Logosol, Inc\Logosol Control Centre\" by default, denoted here by [InstallDir]):

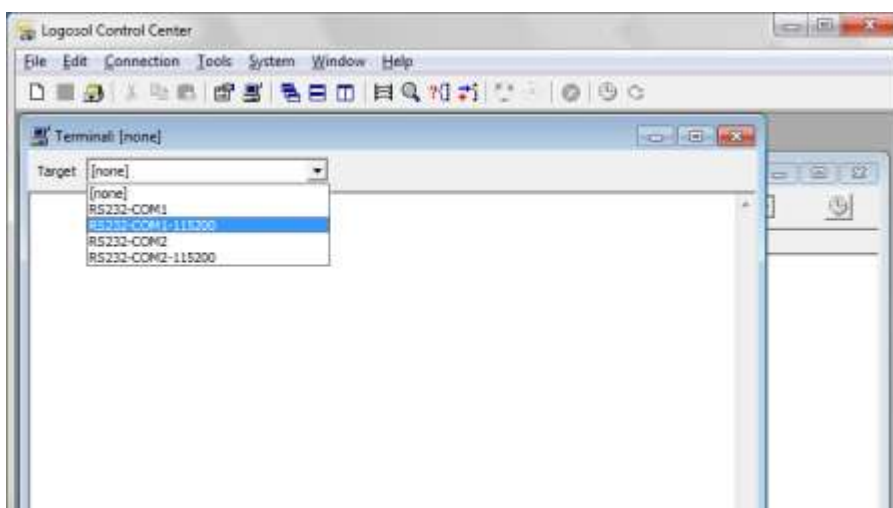
[InstallDir]  
LcCentre.exe, RobotRef.ini,  
PrealRef.ini

Installation directory contains LCC' executable (RcCentre.exe), and files containing Command Reference information.

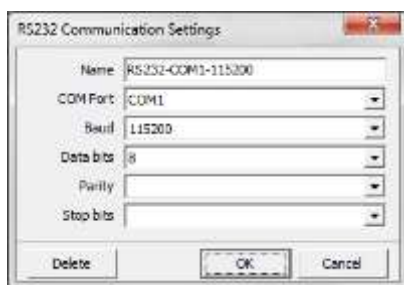
After installation, four communication profiles are created for connection to controllers, with following settings. By default, controllers are shipped for connecting at 9600bps.

RS232-COM1	Serial port: COM1, 9600bps, 8 bits, 1 stop bit, no parity
RS232-COM1-115200	Serial port: COM1, 115200bps, 8 bits, 1 stop bit, no parity
RS232-COM2	Serial port: COM2, 9600bps, 8 bits, 1 stop bit, no parity
RS232-COM2-115200	Serial port: COM2, 115200bps, 8 bits, 1 stop bit, no parity

When started, LCC shows last used applets (if this was selected on "Preferences" dialog), or an empty desktop. The toolbar provides for access to some of most used functions, as well as some standard Windows application shortcuts. The F9 function key (small red button) invokes **STOP** command on the controller, and can be used as an emergency stop button. Applet buttons on the toolbar will select same applet if it is already running (e.g. pressing "Terminal" button twice will not open two terminal applets). If more applets of the same type have to be started simultaneously, subsequent applets must be started from the main menu (e.g. Connection | Terminal).



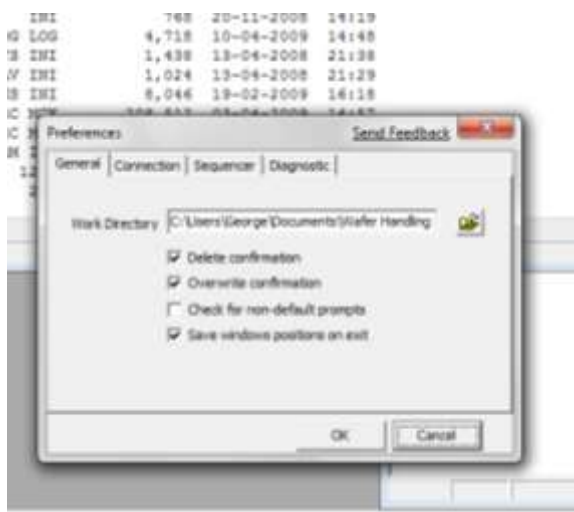
LCC provides for simultaneous connection to several controllers using different serial ports. Each applet operating on a controller has a "Target" selection box to choose required communication setting. Different applets (e.g. terminal window and file manager) can operate on same serial port as long as they use same connection settings (that is, as long as they are using same target).



If another communication profile has to be set up using different communication port (e.g. laptop with USB serial adapter at COM3), following steps have to be performed:

- Create new connection for communication to robot: Connection | New | RS-232
- Enter some descriptive name, e.g. "COM3-Robot-9600". COM port should correspond to PC communication port where WHR robot is connected to, like COM5. Baud should be set to 9600, Data Bits – 8, Parity – None, Stop bits – 1.
- Press OK to save these settings.

When connection is changed, LCC tries to detect the type of controller it is connected to (e.g. robot or prealigners). If the connection parameters are wrong (e.g. baud rate is wrong, or communication port is inaccessible), it could take up to ten seconds for the request for change to time-out.



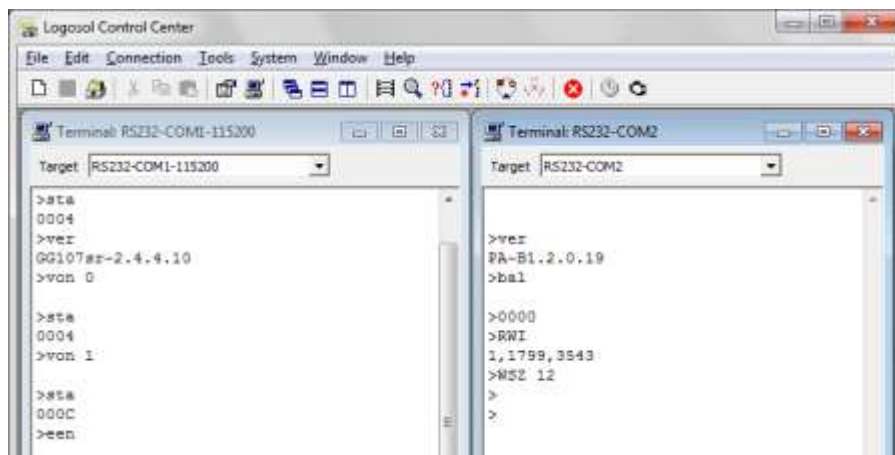
It is recommended to create a folder on the host computer where to store robot related files, and make this folder the "Working Directory" in Tools | Preferences dialog. Information from Preferences dialog is stored in system registry. File Manager applet will start at this folder, making future file management easier. By default, working directory is set to installation folder.

If a "Default connection" is specified in "Preferences" dialog, each applet will start by opening this target. If this functionality is used, and the controller does not respond, it could take a long time for LCC to become responsive when an applet is started.

When using multiple applets with same target, care must be taken with applets which update data regularly. It is not recommended running more that one applet which updates regularly, since data requests can overlap, and corrupted data could be displayed.

## 8.2 TERMINAL EMULATOR

The "Terminal" applet can be used in day-to-day operation and maintenance of the controller. If multiple controllers are connected to different serial ports (e.g. a robot on COM1 and a prealigners on COM2), terminal applets control both controllers.



Commands can be entered on the terminal by keyboard, or they can be pasted using standard Microsoft Windows® shortcut Ctrl-V. Ctrl-C shortcut should not be used, since this shortcut is interpreted by the connected controller, and can cause errors. Use right-click, Copy command, or Edit | Copy menu item.

All commands sent, and all responses received, can be logged to a local disk file using Logging function. Logging is enabled from File | Logging | Start function, and is disabled from File | Logging | Stop function. Local disk file is updated regularly, but it is normal for last couple of lines to be missing from the file during logging.

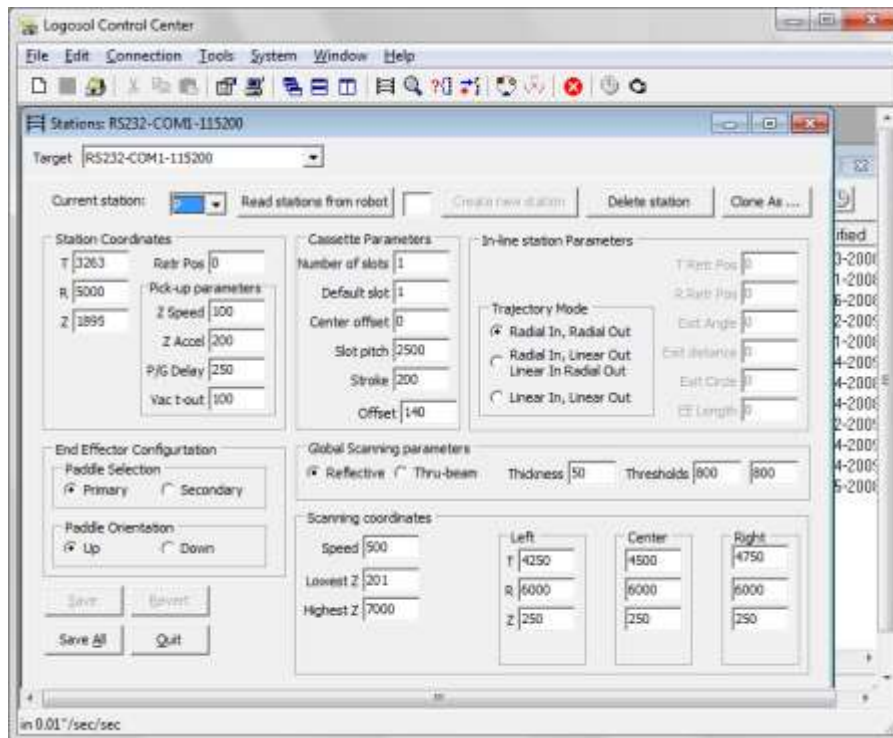


### 8.3 STATIONS EDITOR

This applet provides for following functions:

- Editing of “live” stations residing into robot controller, or stations.ini file
- Direct editing of station coordinates, retract positions, inline parameters and wafer mapping parameters
- Creating new and deleting of existing stations, making copy of stations
- Saving stations back to the robot, and to robot permanent memory

This applet is invoked from Tools | Station menu, or by clicking the “Stations” icon on the toolbar. Depending of number of the stations currently defined, and communication speed, this applet could take relatively long time to respond after target is changed. Screen layout is adapted to actual robot configuration (e.g. dual-arm, track-based etc).



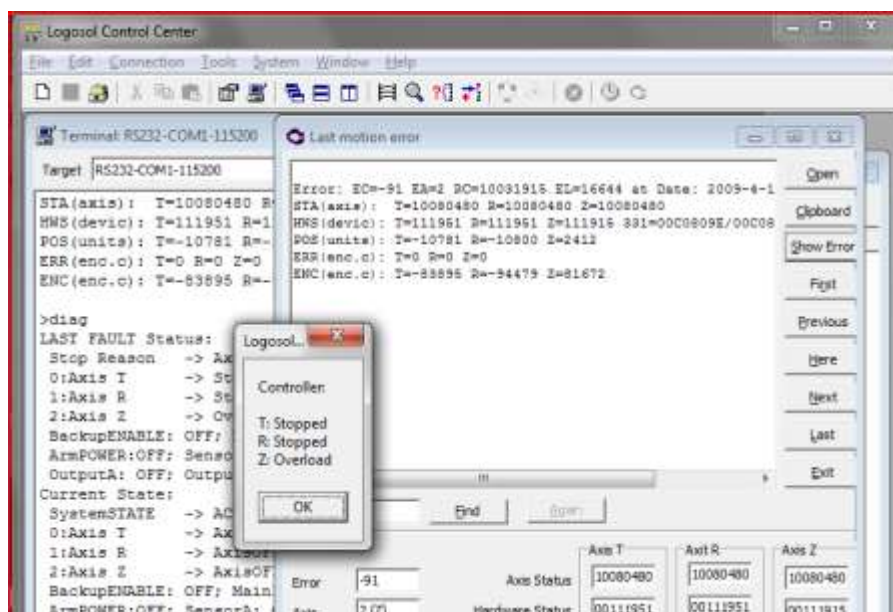
If station enumeration command **ENS\_** is supported on the controller, stations are read from the controller as needed. If this command is not supported (on some really old robot firmware), or “Read stations from robot” is clicked, all stations are first pulled up from the controller. “Read stations from robot” re-reads causes execution of **RSP\_** robot command, and any changes made to stations and not saved will be lost.

“Save” button saves currently edited station only, while “Save All” saves back all stations to the robot, and executes **SSP\_** command on the controller to save them permanently.

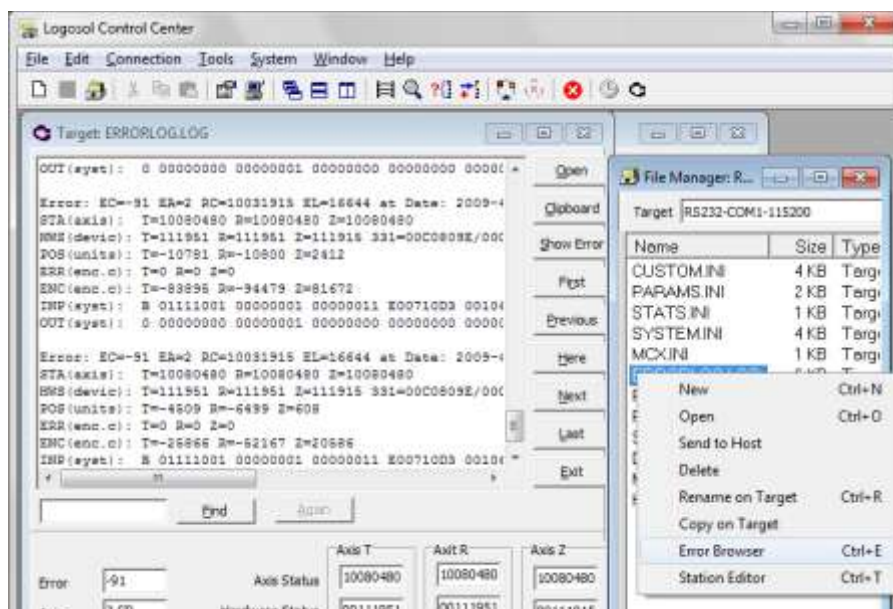
“Copy station” button requests a name of a station (new or existing) where current station parameters are to be copied to. If the destination station already exists, a confirmation will be requested.

### 8.4 ERROR LOG BROWSER

One of most difficult tasks in day-to-day operation of the controllers is determining the reason for motion stop. **DIAG** command can help while diagnosing last error, but it is of no help when an error log has been downloaded and cause for stop is being looked for.

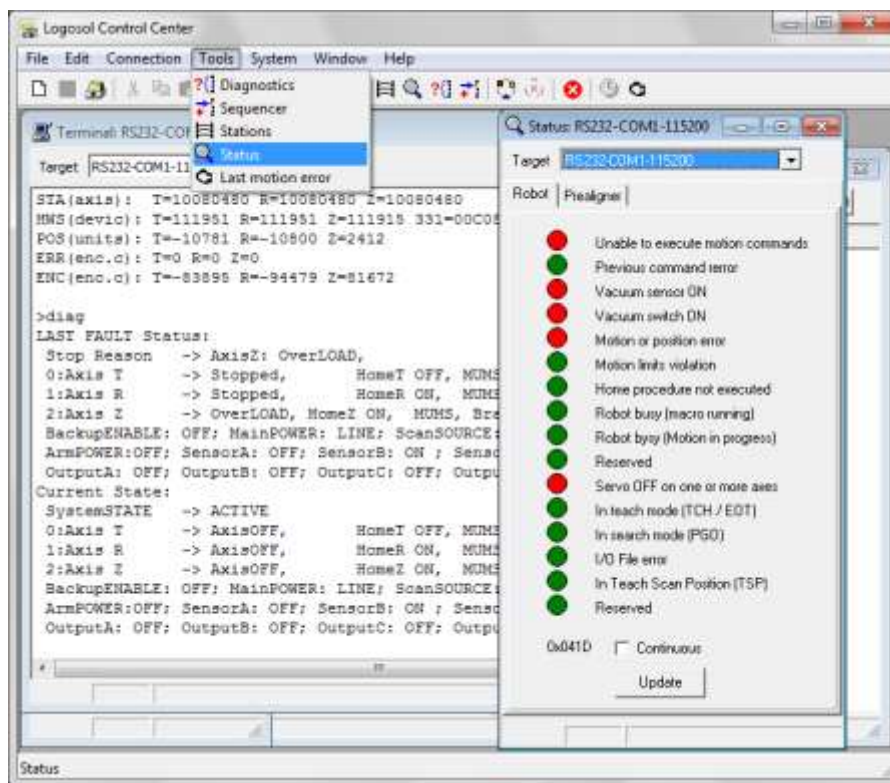


When executed while connected to a controller, Error Browser applet will show result from **EST** command. It can be executed also from File manager applet, by loading local or controller' errorlog.log file. In that case, buttons on the right can be used to navigate through the errors. Placing the cursor over a status field will show text description of that status, while pressing "Show Error" button will open a window with error description.



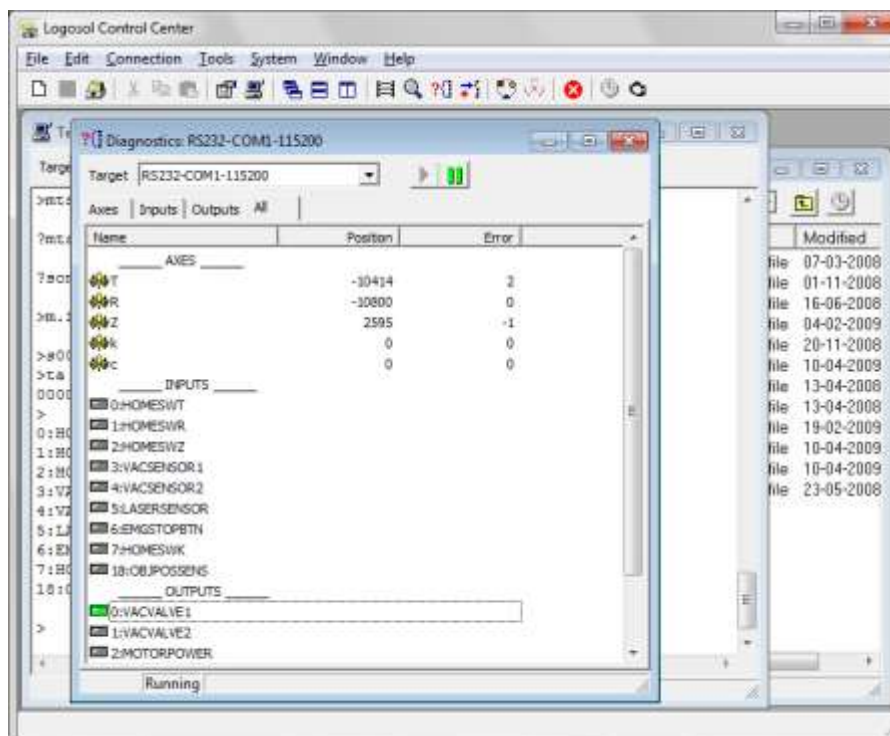
## 8.5 CURRENT SYSTEM STATUS

Current controller status (as reported by **STA** command) can be obtained from System Applet, by clicking on Tools | Status.



Selecting "Continuous" checkbox will cause LCC to request status information from the controller on regular interval, as defined in "Preferences" dialog. Clicking on "Update" button will request a single update.

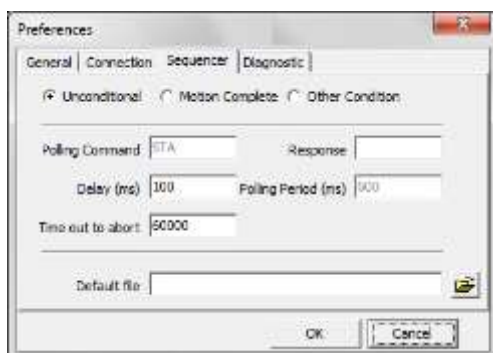
Current axes positions, position errors, input and output states can be obtained from Diagnostics applet, by clicking on Tools | Diagnostics menu item. Double-clicking of an output will toggle its state from ON to OFF and vice-versa.



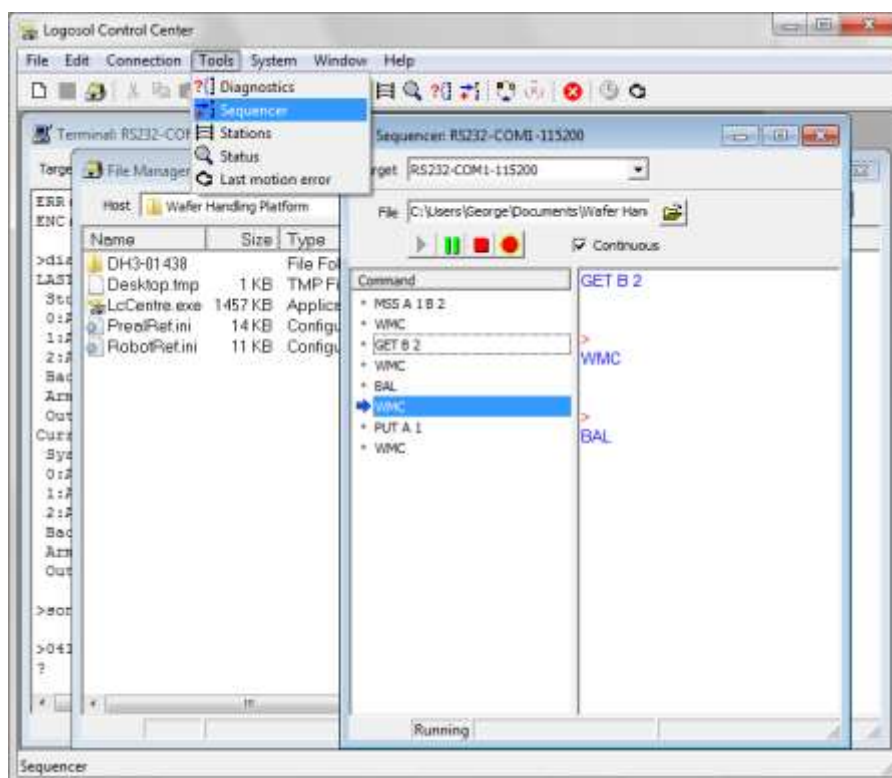
Data displayed by Diagnostics applet is updated continuously on regular interval, unless "Pause" button is pressed. System state is updated next time "Run" button is pressed.

## 8.6 COMMANDS SEQUENCER

A simple commands sequencer is provided, allowing for continuous command execution by the controller. Through a setting in "Preferences" dialog box, the sequencer can be set to execute next command immediately (that is, as soon as the prompt has been received), to wait for motion complete, or to wait for specific command to return specific response. The time between commands, the time between polling commands, and the time out to abort the sequence, and polling command can be configured as well.



In unconditional mode, commands are executed immediately. This can cause problem in designing a sequence of commands which do not finish with return of the prompt (e.g. move commands **MVA/MVR**, wafer handling commands **GET/PUT/MSS**, etc). In that case, "Wait for motion complete" command **WMC** can be executed after command which needs time to be executed.



"Start" button starts the sequencer, "Pause" buttons stops it, or advances it one step. "Stop" button stops the sequencer (but leaves any background motion intact), while "Abort" button stops the sequences, and issues **STP\_** command to target controller.

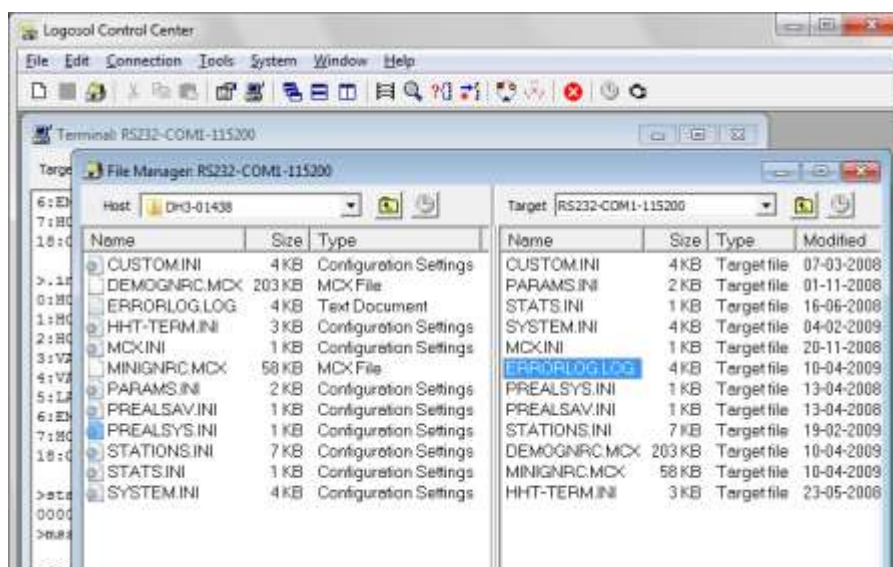


## 8.7 FILE MANAGEMENT

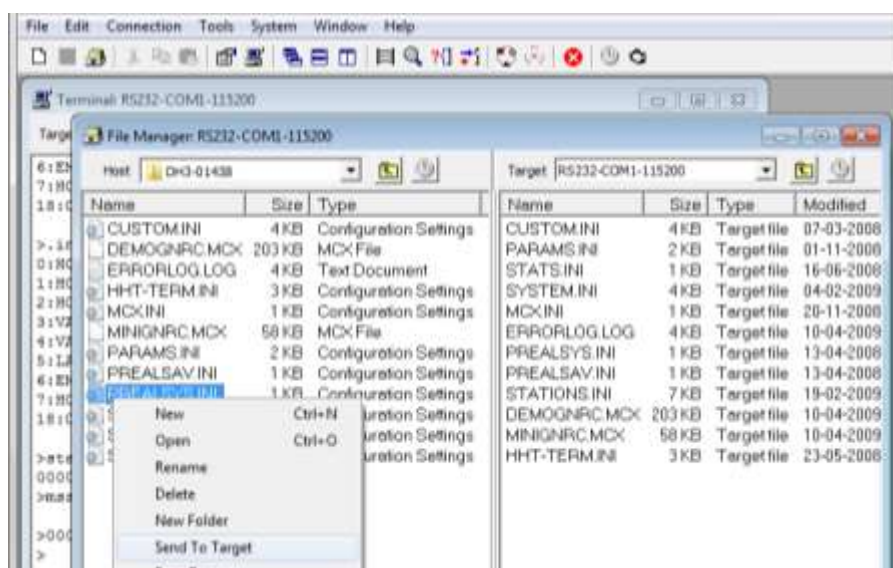
Each controller keeps the information in MS-DOS-like file system. For cases where direct access to files is needed, "File Manager" applet can give access to controller' files, as well as easy access to other applets working with files.

In order to get access to file system, controller must be placed in "Command line interpreter" mode. This is done by issuing ".quit" command in terminal window, or by confirming the target communication in "File manager" applet. This mode differs from standard controller operation by the prompt displayed, "C:\>". Although this prompt resembles command prompt of MS-DOS® operating system, or the one available in NT-class operating systems, the controller does not run neither of these operating systems. Full list of available commands in this mode are given in relevant documentation.

Before entering "Command line" mode on the controller, one must make sure that this is a safe operation, since the controller servo loop will be disabled.



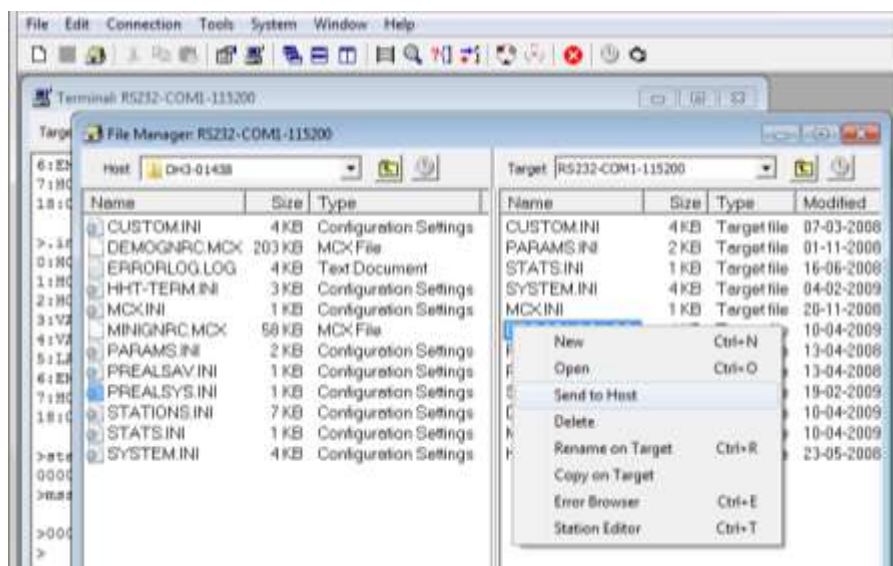
A file can be transferred from host PC to the target by drag-and-drop, or by right-clicking on its name and selecting "Send to Target", or from File | Send to | Target" menu option. Several files can be selected for transfer in single operation by Ctrl-click-ing on them.



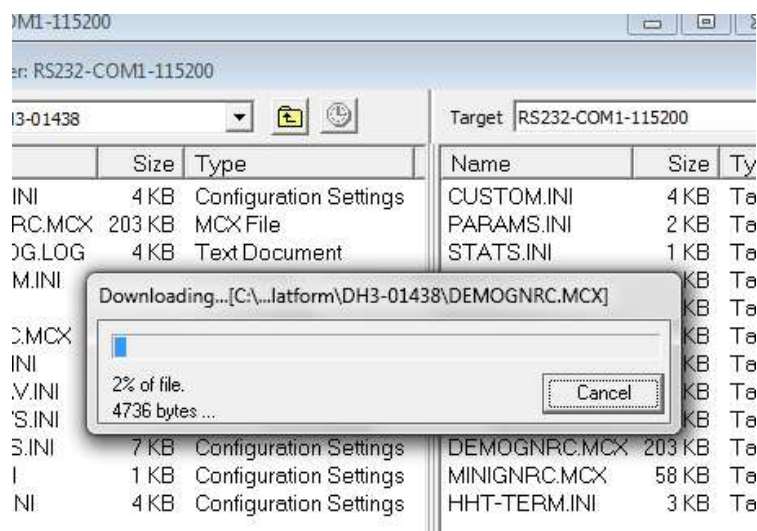
If the file to be transferred has a long file name (that is, file name is more than 8 character, or file extension is more than 3 characters), a warning message will be displayed. If file with the same name exist on the target, a confirmation dialog will be displayed.

Same techniques can be used to transfer files from the target to the host.

In order to transfer files from the host PC to WHR:



The file transfer uses X-modem file transfer protocol, which transfers files using block size of 128 bytes. Therefore, all transferred files will have sizes multiple of 128. Text files will have end-of-file character, with ASCII code 26, used as filler. This character is recognized as end-of-file, and displayed as a square on most text editors, including LCC built-in text editor.



Depending on file size and communication speed, transfer time may vary. A progress indicator shows each file being transferred, and file transfer can be cancelled by pressing <Cancel> button.

Double-clicking of a file on the target will initiate a transfer of this file to a temporary file, then opening this file into text editor applet. When this applet is closed, the file will be saved to the target.

If the file to be opened has an extension of **.INI**, it can be opened as stations parameter file into "Station Editor" applet by selecting "Station editor" from the right-click menu. Similarly, a file with an extension of **.LOG** can be opened into "Error browser" applet by selecting "Error browser" from the right-click menu.

File manager can be used also to rename and copy files on host PC, or onto target.