WL LabMaster™ (for Pegasus) 9.0

Integrated Probing Environment

USER GUIDE

Wentworth Laboratories

Version 9.0 Issue 1
Last revised October 06 by Eddie Scrase/AST

Part # 2081-08219

Contents

Figures

1 Setting Up LabMaster

1.1 Welcome

Welcome to Wentworth Labs' WL LabMaster integrated probing environment. Because LabMaster is uncluttered in design and runs under the Microsoft® Windows™ graphical user interface, it is easy to learn and use.

This manual assumes the user is familiar with the Windows environment. If you have not worked with Windows in the past, you should familiarize yourself with the graphical interface before you start to use LabMaster. You can use the tutorial that comes with Microsoft Windows to introduce yourself to Windows.

1.2 What You Need to Run LabMaster

The following list specifies the minimum software and hardware requirements your computer needs to run LabMaster under Microsoft Windows successfully.

LabMaster requires:

- Microsoft Windows XP SP2 or Windows Vista.
- A personal computer (PC) with at least the recommended processor and minimum RAM for the version Windows you are running.
- A mouse, CD-ROM drive, and a minimum of two free PC bus slots.
- A free serial port if LabMaster is to be used with a Wentworth Pegasus prober fitted with a RS232 interface.
- A National Instruments PCI-GPIB card installed if LabMaster is used with a Wentworth Pegasus or AWP prober fitted with a GPIB interface.
- If overlay video is required, an AGP FlashPoint 4DX board.

1.3 Installation Order

The time needed to install the hardware and LabMaster software should take less than one hour. The steps needed to install and run LabMaster successfully are described below.

1. Install the FlashPoint 4XL board (Sections 1.4)
2. Install the National Instruments PCI-GPIB card (Section 1.5).
3. Connect the cables to the appropriate equipment (Section 1.6).
4. Install the LabMaster Software (Section 1.7).
5. See Chapter 2 for instructions on how to run LabMaster.

1.4 Installing the FlashPoint 4XL board

The FlashPoint 4XL board enables real-time video from the microscope camera to be displayed in the Microsoft Windows environment. True colour (24 bits per pixel), full-motion video can be shown in any VGA graphics mode up to 1280x1024 resolution. By using the LabMaster software the user can manipulate and capture a single frame of video, and can save the captured video frame to a file in a variety of image formats. LabMaster or other Windows programs can recall the saved image at a later time. See Chapter 4 for more information on the LabMaster Video Window and manipulating video images.

This section describes how to configure the FlashPoint 4XL board for the Wentworth LabMaster application. The device drivers and software needed to run the FlashPoint 4XL board is found on the FlashPoint installation CD-ROM.

NOTE Since the computer's cover needs to be removed for the installation of the FlashPoint 4XL board, you may want to install at the same time any National Instruments PCI-GPIB boards that are required.

1.4.1 FlashPoint 4XL Setup Procedure

The FlashPoint 4XL User Manual contains detailed instructions on how to install your FlashPoint board, so this section will only give an overview of installation procedure. ***Important:*** *Please read all of this section before attempting to install your FlashPoint board, because it contains LabMaster specific comments.*

The FlashPoint 4XL board replaces your existing AGP or PCI graphics board. Installing the FlashPoint board is a three part procedure:

1. First the FlashPoint hardware is installed by following the steps in the **Hardware Installation** section of the Installation chapter in the FlashPoint Manual.

2. Then the FlashPoint device driver is installed by following the steps in the **Installing the Display Driver under Window XP** section of the Installation chapter in the FlashPoint Manual

3. Finally the FlashPoint software is installed by following the steps in the **Installing FlashPoint 4XL software under Windows XP** section of the Installation chapter in the FlashPoint Manual.

1.4.2 Trouble shooting the FlashPoint 4XL board

- Verify that the camera is powered up and has been correctly connected to the FlashPoint board.

- Verify that the FlashPoint board has been correctly installed following the instructions in the FlashPoint Manual.

- Verify that the FlashPoint device driver has been correctly installed following the instructions in the FlashPoint Manual.

- Verify that the FlashPoint software has been correctly installed following the instructions in the FlashPoint Manual. This can be verified by displaying live video using the FlashPoint demonstration program. The program can be run using the following steps:

  1. From the Windows Start Menu, select the Programs icon.
  2. From the Programs menu, select the FlashPoint 4XL icon.

3. From the FlashPoint 4XL menu, select the FPG3D32 icon.

If there is a problem that could not be solved by the steps outlined in this section call Wentworth Laboratories. Please have the following information ready before you call.

- Computer brand, model, processor, and RAM.
- Version of Windows you are using.
- The current Windows VGA resolution. The current VGA resolution can be found in the LabMaster System Information dialog box by selecting the **Help | System Information...** menu item.
- Computer Configuration: amount of memory, and list any hardware installed in the computer (LAN's, I/O cards, etc.)
- Symptoms of problem, including the graphics mode.
- Steps already taken to try to solve the problem.

1.5 Installing the National Instruments PCI-GPIB Board

The LabMaster application is designed to interface with a National Instruments PCI-GPIB (IEEE-488.2) board. The GPIB board is needed only if LabMaster is to control external test equipment using a GPIB interface. This section describes the PCI-GPIB board configuration needed when interfacing with the LabMaster program.

All software needed to use the PCI-GPIB board with LabMaster is included on the National Instrument CD-ROM. For information on your PCI-GPIB board and the GPIB device drivers, see the National Instruments documentation that was provided with your PCI-GPIB board.

1.5.1 PCI-GPIB board Installation Procedure

Hardware needed to install the PCI-GPIB board

- A National Instruments PCI-GPIB board.
- A GPIB cable.

Software needed to configure the PCI-GPIB board

- The Wentworth LabMaster program is used to select a GPIB address for a device such as the AWP prober.
- The National Instruments GPIB Config. program is used to configure the hardware parameters of the PCI-GPIB board.

- The National Instruments GPIB Test program is used to test a newly installed PCI-GPIB board.

Selecting the PCI-GPIB board's I/O address

The factory default I/O address for the PCI-GPIB board is 0x2C0 hex. Check to determine that this space is not already used by other equipment installed in your computer. If any equipment in your computer uses this I/O address space, change the base I/O address of either the PCI-GPIB board or the other device. If the PCI-GPIB board's I/O address is changed, then a corresponding change needs to be made using the GPIB Config. program. See the National Instruments PCI-GPIB documentation for more information on how to change the I/O address.

To Install the PCI-GPIB board:

1. Turn off the computer.
2. Remove the cover.
3. Plug the PCI-GPIB board into an unused PCI slot.
4. Replace the cover.
5. Connect the GPIB cable between the PCI-GPIB board and the Pegasus.
6. Turn the computer back on.

Using the GPIB Config. Program

GPIB Config. is an interactive program from National Instruments that is used to modify the configuration parameters for your PCI-GPIB board and the GPIB devices connected to it. Any changes made by GPIB Config. are stored in the Windows directory in the GPIB.INI file. If you have changed the default address of the GPIB board, you will have to make a corresponding change using the GPIB Config. program. See the National Instruments documentation for more information.

Using GPIB Test to check your PCI-GPIB board

When the PCI-GPIB board has been installed and configured, the GPIB Test program can be used to determine if the PCI-GPIB board has been setup correctly. This program requires that there be no cables connected to the PCI-GPIB board. See the National Instruments documentation for more information on using the GPIB Test program.

1.6 RS232 and GPIB Connections

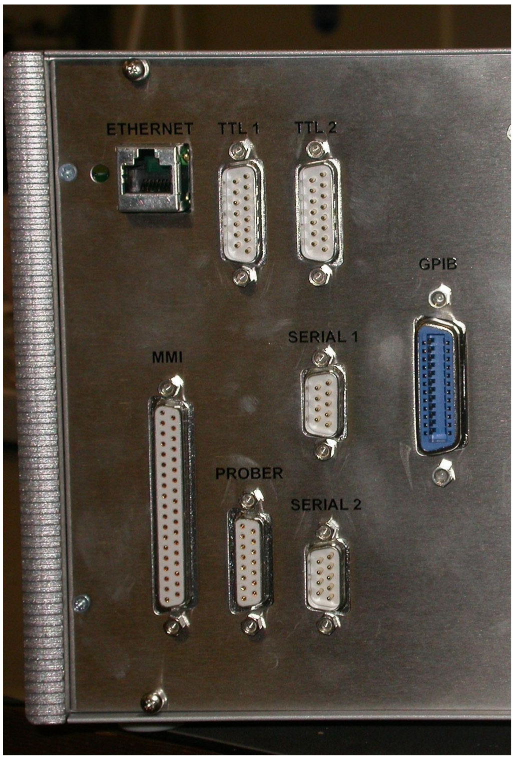


Figure 1-2 LabMaster PC Cable Connections

The Pegasus probers have RS232 and GPIB (IEEE 488.2) inputs that allows LabMaster to control its functions and movement (depending on the Pegasus accessories fitted). The GPIB board in the PC is connected to the prober's GPIB input using a GPIB cable. The RS232 port on the PC is connected to the prober's Serial 1 port using a standard 9 pin RS232 cable, see Figure 1-2.

The Prober's GPIB address is selected using the GPIB sub-menu of the Remote menu. See the Pegasus User's Manual for more information on its GPIB interface. The GPIB address must be entered into LabMaster using the Prober's Hardware Setup dialog box. Ensure you set the Prober to "INF" mode, with Text Terminator 10 (LF), and set the prober to remote mode using the Remote soft key before you start the LabMaster program.

1.7 Installing the LabMaster Software

The LabMaster software is installed by using the Wentworth Setup program located on the CD-ROM. The Wentworth Setup program checks for the minimum system requirements to run LabMaster and warns you if your system does not meet the requirements. Setup also requires you to enter a user name, a company name, and a prober serial number during the setup process. This information is used for the LabMaster software registration.

To set up LabMaster:

1. Insert the LabMaster CD-ROM in the CD drive, and close the drive door.
2. Close any open applications so that only Windows is running.
3. In the Windows Start Menu, choose the **Run...** menu item. The Run dialog box opens.
4. In the Command Line box, type **d:\setup** (replacing d: with the CD-ROM drive letter) and then choose the OK button.

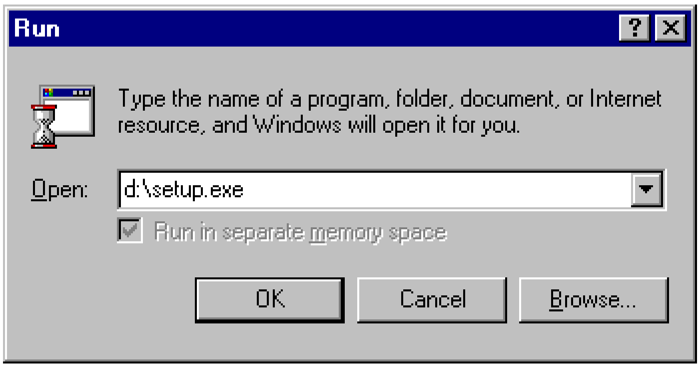


Figure 1-1 Run Dialog Box

5. Follow the instructions on the Screen. LabMaster installation should take less than 5 minutes.
6. After Setup has finished installing LabMaster, a new Wentworth Laboratories sub-menu will be added to the Program sub-menu of the Windows Start Menu, with icons depicting the programs provided with LabMaster. In addition, a LabMaster icon will be added to the top of the Start Menu.

2 LabMaster Basics

2.1 Introduction

LabMaster is a very user friendly application and is easy to learn and use. This chapter introduces you to the LabMaster program, and will provide you with basics needed to setup and run it. The following chapters describe LabMaster REXX Programming, Video functions, Wafer Map window, and using Dynamic Data Exchange (DDE) with LabMaster. This information is also located in the LabMaster on-line help. The LabMaster on-line help feature provides the user with detailed instructions on how to use the LabMaster application.

2.2 Before Starting the LabMaster Application

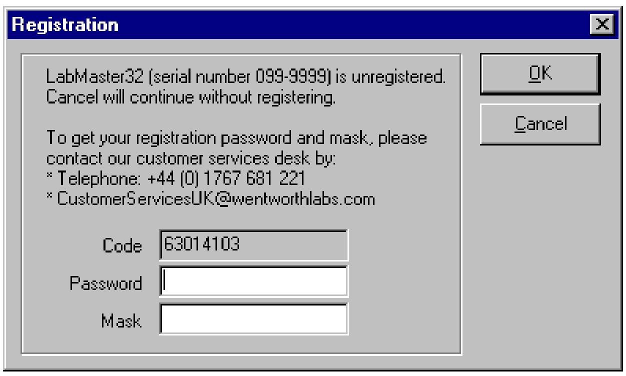
Before the LabMaster program is run, check that the required boards (for example, FlashPoint 4XL, GPIB) are installed and configured correctly.

Check that the Pegasus prober's Mode is set to INF Mode and the EOT Character is set to 10 using either the RS232 Menu or the GPIB Menu (depending on whether a serial or a GPIB link is being used). If a serial link is being used to connect the prober with the LabMaster PC, also check that the prober's Communication Parameters are set to 38400,7,E (recommended) using the RS232 Menu. Finally, the prober must be put into remote mode using the Remote soft-key. Please refer to the Pegasus Operating Manual for more details.

2.3 LabMaster Registration

2.3.1 Full Installation

The full version of the LabMaster application requires that you register the application to work with a given prober. Before registration, the LabMaster application will only allow you to work with any prober for a period of 2 days (or 20 activations). Each time the application is run, the registration dialog box will be displayed:



To register your LabMaster installation, simply contact our service desk with the following information:

1. The serial number of LabMaster (displayed in the dialog box above)
2. The registration code (also displayed in the dialog box above)
3. The type of prober you will be using
4. The serial number of the prober (label on the back of the prober's stage or controller)

Our service desk will then supply you with a registration password and mask. When these are entered in this dialog, the application will then run without displaying this dialog box again. If you need to re-register (for example, if you change the prober you are working with), the registration dialog box can always be accessed from the about dialog box.

2.3.2 Evaluation Installation

The evaluation installation of the LabMaster application will allow you to work with any prober for a period of 30 days (or 300 activations). Each time the application is run, the number of days remaining before expiry will be displayed:

A full license may be purchased at any time, and a registration password and mask used to turn the evaluation installation into a full installation. After the evaluation period has expired, the LabMaster application will not run until a full license is purchased.

2.4 Starting the LabMaster Application

The LabMaster application is started by double clicking the left mouse button on the LabMaster icon in the Wentworth program group. Once you have done this LabMaster will begin to initialise. If you receive any video error messages during the LabMaster initialisation and/or the colour magenta is shown in the LabMaster Video window, exit LabMaster and use the Video Config. program to align the current VGA mode. Once LabMaster has initialised, the LabMaster main window will be displayed with the real-time Video window and the Device Toolbar.

If this is the first time LabMaster has been run, the Device Toolbar will contain only one button as shown in Figure 2-1. This button is the Stop button and it is used to stop any device's movement. The Hardware Setup dialog box, which is activated by selecting the **Setup | Hardware...** menu item, is used to add devices to the Toolbar.

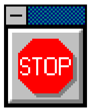


Figure 2-1 Initial LabMaster Device Toolbar

If the Hardware Setup dialog box has already been used to configure LabMaster for a Pegasus prober with a motorised platform, a PMM, six CAPs, and a Temptronic 3010 ThermoChuck®, the Toolbar will contain the device buttons as shown in Figure 2-2.

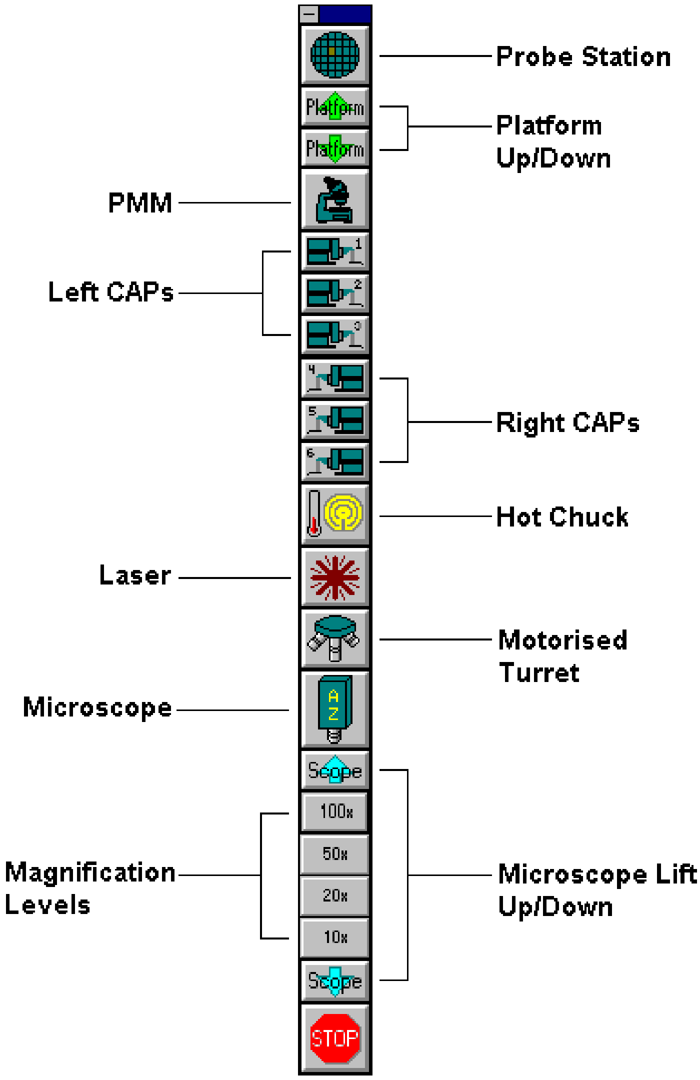
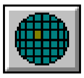
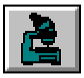
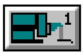
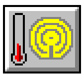
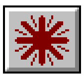
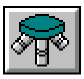
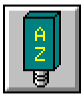


Figure 2-2 Fully Configured LabMaster Toolbar

Select	To
	If the Probe Station button is selected with the left mouse button, it will become the active master device. The Motion Control Dialog Box and the Probe Station Status Window will be displayed. If a PMM and/or CAP(s) are active, they will be automatically de-selected. If the right mouse button is clicked while the cursor is over this button, the Probe Station Pop-up Dialog Box will be displayed.
	If the Platform Up button is depressed, LabMaster will raise the prober's platform to the user definable upper limit.
	If the Platform Down button is depressed, LabMaster will lower the prober's platform to the user definable lower limit.
	If the PMM button is selected with the left mouse button, it will become the active master device and the PMM Status Window and the Motion Control Dialog Box will be displayed. If a Probe Station is active, it will be automatically de-selected. If one or more CAPs are active, the CAP's motion will track the motion of the PMM, which will become the master. If the right mouse button is clicked while the cursor is over the button, the PMM Pop-up Dialog Box will be displayed
	If a CAP button is selected with the left mouse button, it will become the active master device and the CAP Status Window and the Motion Control Dialog Box will be displayed. If a Probe Station is active, it will be automatically de-selected. If a PMM and/or other CAPs are active, the PMM and/or CAP's motion will track the motion of the master CAP. If the right mouse button is clicked while the cursor is over the button, the CAP Pop-up Dialog Box will be displayed
	If the TP03000 Series hot chuck button is selected with the left mouse button, the TP03000 Series Status Window will be displayed. If the right mouse button is clicked while the cursor is over the button, the TP03000 Series Pop-up Dialog Box will be displayed
	If the New Wave Laser button is selected with the left mouse button, the Laser Control Dialog Box will be displayed. This dialog box is used to control the functions of the Laser via RS-232. The right mouse button has no effect for this button.
	If the Motorized Turret button is selected with the left mouse button, the Motorized Turret Control Dialog Box will be displayed. This dialog box is used to control the functions of the Motorized Turret via RS-232. The right mouse button has no effect for this button.

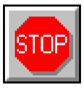


If the A-ZOOM Microscope button is selected with the left mouse button, the A-ZOOM Control Dialog Box will be displayed. This dialog box is used to control the functions of the A-ZOOM via RS-232. The right mouse button has no effect for this toolbar button.

If the Scope Up button is depressed, LabMaster will raise the microscope lift (either the motorized quick-lift or the 7" focus block) all the way to the top. In systems configured with a 7" focus block, the current position is saved for Scope Down. The Scope Up and Scope Down buttons are placed on the toolbar using the **Setup | Options…** menu item.

If one of the Magnification Level buttons is depressed, LabMaster will perform Auto-Par Centering/Focusing and use the new magnification value in the same manner as it does under the **Setup | Magnification…** menu item. The button representing the current Magnification Level will be indicated by a darker outline. The Magnification Levels can be placed on the toolbar for easier access using the **Setup | Options…** menu item.

If the Scope Down button is depressed, LabMaster will lower the microscope lift all the way to the bottom for the motorized quick-lift or back to its previous position for the 7" focus block. The Scope Up and Scope Down buttons are placed on the toolbar using the **Setup | Options**… menu item.



If the Stop button is depressed it will stop the motion of all active devices except the Pegasus probers. The movement on Pegasus probers can only be stopped by pressing the Emegency Stop button or by switching the controller off.

2.5 Hardware Setup Dialog Box

Most devices can only be controlled when present on the Toolbar. The exception is a Manual Probe with Contact Sense, (which is controlled through the Manual Probe Setup dialog box). The Hardware Setup dialog box is used to add or remove devices from the Toolbar. It can also be used to modify a device's hardware setup parameters.

To Add a Device:

1. Select the **Setup | Hardware...** menu item. The Hardware Setup dialog box will appear.

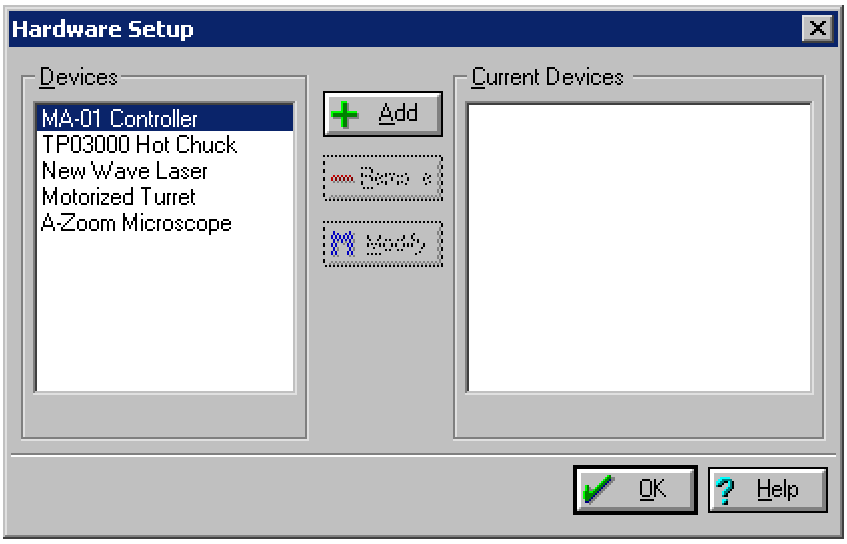


Figure 2-3 Hardware Setup Dialog Box

2. Use the **Devices** list box to select the device you wish to add to the Toolbar, and then press the **Add** button. The selected device's hardware setup dialog box will be displayed.

3. In the device's hardware setup dialog box, enter the name that you wish to call the device. The name will be used reference the device in messages, menu items, and dialog boxes displayed by LabMaster for that device.

4. Enter the device's serial port, or GPIB address as appropriate. The GPIB address must match the Pegasus prober or Temptronic 3010's GPIB address and must be in the range from "DEV1" to "DEV16".

5. Depress the **OK** button. The LabMaster program will reinitialise and the device's button will be added to the toolbar.

2.6 Moving Devices

There are four ways to move a device using LabMaster. The device may be moved using the arrow buttons in the Motion Control window, using the keyboard arrow keys, using the mouse and the Video window, and by using the Wafer Map window.

To Move a Device:

1. Select the device's button on the Toolbar. The device will be enabled and the device's motion control window and status window will be displayed. If more than one device is selected on the toolbar, the last device selected will be the *master* and the previously selected devices will be *slaved* to the master. When the master is moved the slaves will track the master device's motion.

2. Use the slow, medium, and fast arrow buttons on the Motion Control window to move the device in the X, Y or Z direction. The magnitudes of the slow, medium, and fast speeds can be entered using the device's setup dialog box.

3. Use the arrow keys on the keyboard to move the device in the X or Y axis, when the Motion Control window has input focus (i.e. its caption bar is highlighted). The "Page Up" and "Page Down" keys are used to move the device in the Z-axis. The speed that the device will move when a keyboard arrow key is depressed is indicated in the Motion Control window as "slow", "medium", or "fast". The "+" and "-" keys can be used to increase and decrease this speed. "Num Lock" must be off to use the keypad arrow keys.

4. If the right mouse button is clicked in the Video window, the device will move to the position where the mouse button was clicked. The LabMaster Video window must be correctly calibrated to the current microscope magnification before a device can be moved using the Video window. See Chapter 4 for more information.

5. The Wafer Map window can also be used for device navigation. See Chapter 5 for more information.

2.7 Pegasus Probe Stations

The Pegasus prober is Wentworth's current generation of automatic production probe station.

LabMaster controls the Pegasus prober via either an RS232 interface or a GPIB (IEEE-488.2) interface using the National Instruments PCI-GPIB board. If the GPIB interface is being used, the PCI-GPIB board must be installed and configured, and the Pegasus' GPIB device address must match the one selected using the Pegasus Hardware Setup dialog box. Also the prober must be placed in external mode by using the remote soft key on the prober.

2.7.1 Pegasus Motion Control Window

The Pegasus Motion Control window is used to control the motion of the prober's chuck. The arrow buttons are used to index the prober the distance specified by the index step values entered using the prober’s Setup dialog box.

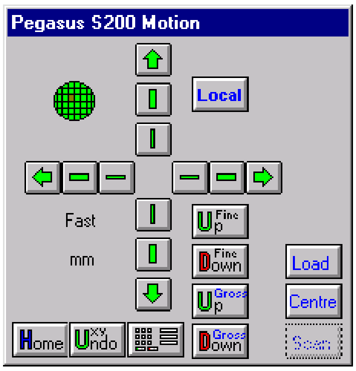
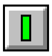
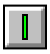
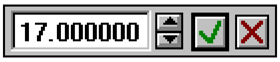
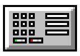
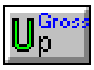


Figure 2-4 Pegasus Motion Control Window

Select		To
	Fast	Move at the FAST velocity. The fast, medium, and slow velocities for the X, Y, and Z axis can be set using the device's setup dialog box. If the Index Mode is active, the chuck will index by the large index size.
	Medium	Move at the MEDIUM velocity. If the Index Mode is active, the chuck will index by the medium index size.
	Slow	Move at the SLOW velocity. If the Index Mode is active, the chuck will Index by the small index size.
		By clicking the RIGHT mouse button on one of the above arrow buttons, the Change Speed Pop-up window will be displayed. The current speed for that button is displayed in mm/sec. Enter a new speed for the arrow button and then press the  button or the  button to cancel.
	Undo XY	Display a message box asking if the last XY move should be undone. Answering **YES** will cause the device to be moved back to the previous position. This simple 1-level undo function will allow moves in the X and/or Y axis that are initiated from the Motion Control Window or the Video Window be 'undone'. Moves from a Wafer Map Window or a REXX window cannot be 'undone'. Whenever new devices are selected on the Toolbar, the undo function is reset.
	Goto Home	Move to the User Home position. If the **Setup | Options | Confirm Goto User Home** menu item is checked, then a message box will be displayed asking the user to confirm the move. Answering **YES** will cause the chuck to move to the user selected home position.
	Numerical Entry	Display the Numerical Entry dialog box. This dialog box allows the user to move the chuck to an absolute location.
	Chuck Up	Cause the chuck to move to the Up position. The chuck is moved up the specified chuck drop distance. If the chuck is already in the up position this button will have no effect. The chuck drop distance is entered in the Pegasus Setup dialog box

	Chuck Down	Cause the chuck to move to the Down position. The chuck is moved down the specified chuck drop distance. If the chuck is already in the down position this button will have no effect. The chuck drop distance is entered in the Pegasus Setup dialog box.
	Gross Chuck Up	Cause the chuck to move to the Gross Up position. If the chuck is already in the gross up position this button will have no effect. This button is present only if a Pegasus prober is selected on the Toolbar.
	Gross Chuck Down	Cause the chuck to move to the Gross Down position. If the chuck is already in the gross down position this button will have no effect. This button is present only if a Pegasus prober is selected on the Toolbar.
 .	Local	Cause the Pegasus prober to revert to local control. The Pegasus prober can then be controlled using the keypad and joystick attached to the prober. This button is present only if an Pegasus prober is selected on the Toolbar.
 .	Load	Cause the Pegasus prober to move to the wafer loading position, and switch off the vacuum. The user can then unload the current wafer from the chuck, and load the next wafer. This button is present only if an Pegasus prober is selected on the Toolbar.
	Centre	Cause the Pegasus prober to move to the Centre position. The vacuum is put on before the chuck is moved, and it is raised to the Gross Lifted position after the move. This button is present only if an Pegasus prober is selected on the Toolbar.
	Align	Cause the Pegasus prober to enter the Align procedure, in order to allow the user to theta align the wafer. On manual theta chucks, this button enters the scan alignment procedure. On automatic theta chucks, this button enters the two-point alignment procedure.
	Goto Cruise	Move CAP to its cruising height. The Cruise height is entered in the CAP Setup dialog box.

	Goto Pretouch	Move CAP to its pretouch height. The Pretouch height is entered in the CAP Setup dialog box.

2.7.2 Pegasus Pop-up Dialog Box

The Pegasus Pop-up dialog box contains frequently used functions for the Pegasus. It can be activated via the standard two methods. The first method displays the Pop-up dialog box for the current master device by double clicking the left mouse button anywhere in the Video window. The master device is the last device selected on the Toolbar. The second method displays the Pop-up dialog box by clicking the right mouse button on the device's Toolbar button.

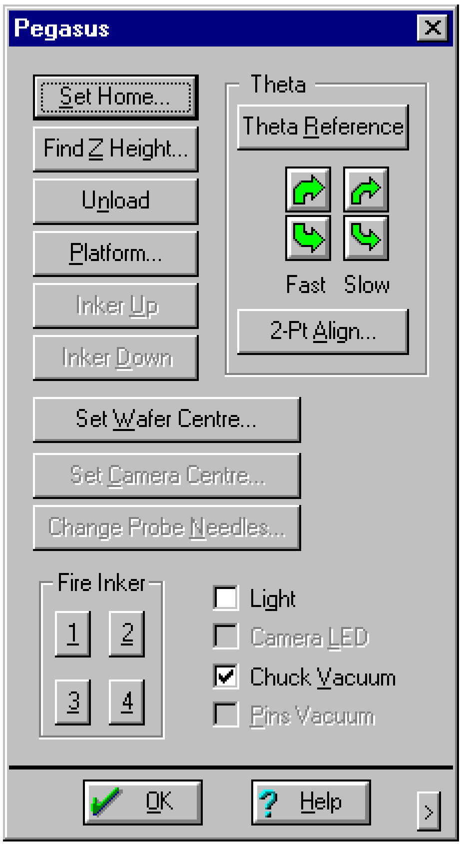


Figure 2-5 Pegasus Pop-up Dialog Box

The functions provided by the Pegasus Pop-up dialog box are:

Select	To
Set Home...	Display a message box asking if the current chuck position is to be set as the user selected home position. Answering **YES** will set the current position as 0.00000, 0.00000.

Find Z Height...	Display a message box asking if the chuck Z height is to be found. Answering **YES** will enable the user to re-establish the Z height of the chuck on the system.
Unload	Pressing this button will cause the chuck to move to the unload position.
Platform	Pressing this button will display the Pegasus Platform Dialog Box.
Inker Up	Pressing this button will cause the inker to be lifted, on a DSP prober.
Inker Down	Pressing this button will cause the inker to be put to the down position, on a DSP prober.
Set Wafer Centre	Pressing this button will cause the present position to be set as the wafer centre position (centre of circular probing zone), if it is a valid position.
Set Wafer Centre	Pressing this button will cause the present position to be set as the wafer centre position (centre of circular probing zone), if it is a valid position.
Change Probe Needles	Pressing this button will prompt the user to change the probe needles on a DSP prober.
Fire Inker	Fire the selected inker (1, 2, 3 or 4).
Theta:	Adjust the chuck theta position in the clockwise or counter-clockwise directions by using the slow and fast theta direction buttons.
Theta Reference	Reference the chuck in the theta axis.
2-Pt Align	Display the Pegasus' Two-Point Theta Align dialog box. After two points are selected on the wafer LabMaster will automatically move the chuck in theta to correct for theta misalignment. The two points can either be on the X axis or on the Y axis.
Light	Checking this check box turns the overhead lamp on or off.

Camera LED	Checking this check box turns the machine vision camera LEDs on or off.
Chuck Vacuum	Checking this check box turns the chuck vacuum on or off.
Pins Vacuum	Checking this check box turns the chuck vacuum on or off.

2.7.3 Changing the Pegasus' Setup Parameters

Pegasus' setup parameters can be altered by using the Pegasus Setup dialog box. The Pegasus Setup dialog box is displayed by selecting Pegasus' name under the Setup menu. Once the setup dialog box is visible, you can alter such items as the index step values, and Z overdrive. When an item in the Pegasus Setup dialog box is altered it is stored on the disk so that it can be recalled the next time LabMaster is run. Below is the Pegasus Setup dialog box.

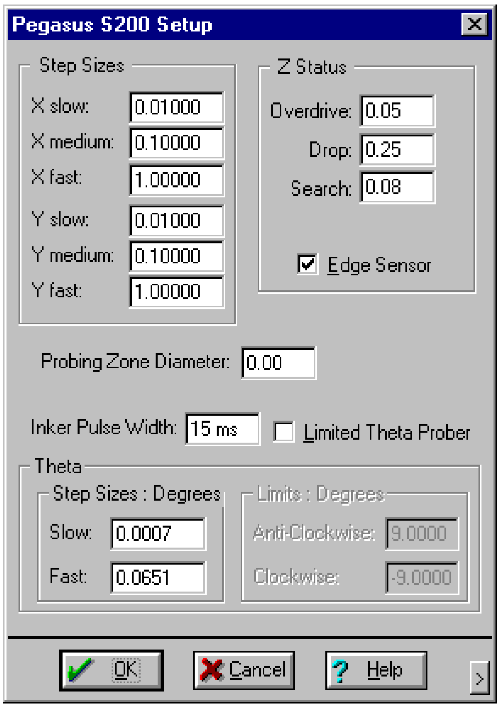


Figure 2-6 Pegasus Setup Dialog Box

The Pegasus Setup dialog box is used to configure the following device parameters of Pegasus.

Select	To
Index Step Sizes	Set the index step sizes for the Slow, Medium, and Fast arrow buttons on the Motion Control dialog box. Pegasus will index the specified step size when an arrow button is pressed

Theta Index Step Sizes Set the Theta axis index step sizes for the Slow, and Fast arrow buttons on the Pop-up dialog box. Pegasus will index the specified step size when an arrow button is pressed

Z Overdrive Set the Z Overdrive for the chuck. The chuck will move the Z Overdrive distance past the position that the edge sensor makes contact.

Z Drop Set the Z drop distance for the chuck. If the chuck is not already in the down position, it will drop the Z drop distance from the current chuck position when the chuck is moved in X or Y or the 

button is pressed.

Z Search Set the Z Search Window for the chuck. The chuck will move the Z Drop + Z Search distance to look for the edge sensor making contact, before applying the Z Overtravel.

Edge Sensor Checking this check box means that chuck up movements will search for the edge sensor making contact (using the Z Search parameter). If the check box is unchecked then chuck up movements will not look for the edge sensor, but will move by Z Drop + Z Overtravel.

Probing Zone Diameter Set the diameter of the circular probing zone. A value of zero indicates that the circular probing zone is not being used (so Pegasus stage movement is only limited by stage size).

Inker Pulse Width Set the amount of time which the inkers are fired for, which determines the size of the ink mark.

Limited Theta Prober This check box should be checked if the probe station chuck is an **ATC** with limited theta movement. If the check box is unchecked then the **ATC** chuck has unlimited theta movement (applies to ATC chucks only).

Theta Limits Set the limits of theta movement in Anti-Clockwise and Clockwise directions (Applies to limited theta ATC chucks only).

2.7.4 Pegasus Platform Dialog Box

The Platform dialog box is used to adjust the Pegasus platform's position and is also used to set the platform's safety limits. This dialog box is displayed either by selecting the **Platform...** button on the Pegasus Pop-up dialog box, or by right-clicking on either platform button on the Device Toolbar.

The first time a platform movement button is pressed, a warning message box is displayed asking if there is room to move the platform. If the platform is not initialised, the platform will be initialised first.

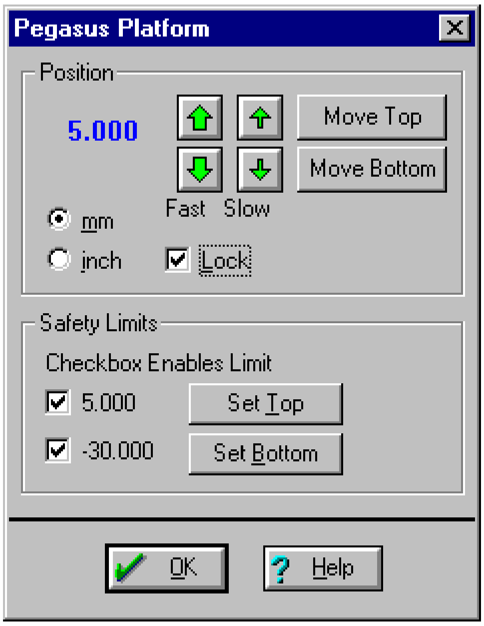


Figure 2-7 Pegasus Platform Dialog Box

The functions provided by the Pegasus Platform dialog box are:

Select	To
Position	Adjust the platform position in the upward or downward direction by using the slow and fast platform direction buttons. The position is displayed in either inches or mm.
Move Top	Move the platform position to the top safety limit. The position is displayed in either inches or mm.
Move Bottom	Move the platform position to the bottom safety limit. The position is displayed in either inches or mm.
Lock	Lock the platform's position at its current location. No platform movement will be allowed until it is unlocked. The displayed platform position will turn **blue** as a reminder that it is locked.

Set Top Set the current platform position as the top safety limit. If the top check box is checked, and the platform is moved, it will stop when it encounters this safety limit, and the displayed platform position will turn yellow. If the check box is unchecked, the displayed platform position will turn **red** when the safety limit is passed.

Set Bottom Set the current platform position as the bottom safety limit. If the bottom check box is checked, and the platform is moved, it will stop when it encounters this safety limit, and the displayed platform position will turn yellow. If the check box is unchecked, the displayed platform position will turn **red** when the safety limit is passed.

2.8 Computer Assisted Probe (CAP)

The CAP Computer Assisted Probe has a step size of 0.002 micron. LabMaster controls the CAP's X-Y-Z movements using the Wentworth PC-Based Stepper Motor Controller boards and Motor Driver Box.

2.8.1 CAP Motion Control Window

The CAP Motion Control window is used to control the motion of the CAP's X, Y, and Z-axes. The arrow keys are used to move the CAP at the speeds specified by the velocity values which are determined by right clicking on the Fast, Medium and Slow motion buttons and entering the speed required.

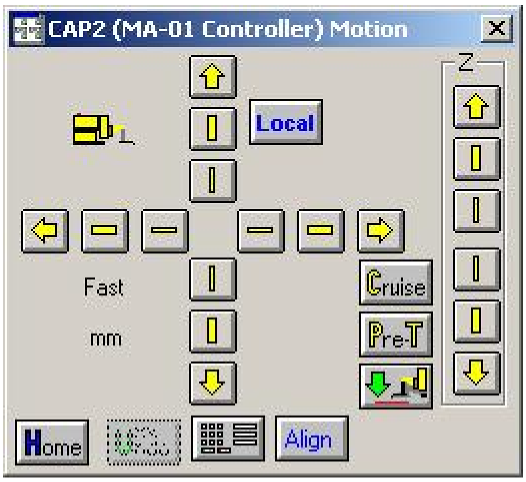


Figure 2-8 CAP Motion Control Window

See Pegasus Motion Control Window, for a functional description of the Motion Control window buttons.

2.8.2 Changing the CAP's Setup Parameters

The CAP's Motion Control Window is used to alter the Setup Parameters, i.e. the X, Y and Z velocities, contact height, pre-touch height and cruising height. When a setup parameter is altered it is stored on the disk so that it can be recalled the next time LabMaster is run.

To select the CAP motion control window click on the required icon.

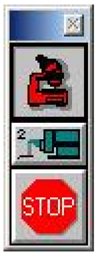


Figure 2-9 Device Selection Window

The CAP motion controller window is used to configure the following parameters:

Select	To
PreTouch Height	Set the PreTouch Height for a probe. The PreTouch Height is used only when a REXX program is running to stop the probe tip at the specified Pretouch height above the wafer. The probe tip can then be lowered using the Motion Control window to make contact with the Device Under Test (DUT). The PreTouch Height can be activated or deactivated by selecting the **PreTouch** check box.
Cruising Height	Set the Cruising Height for a probe. When a CAP is moved in the X or Y axis, the probe will move to the Cruising Height before any movement in X or Y begins. The Cruising Height feature can be activated or deactivated by selecting the **Cruise** check box. **NOTE:** The probe will not rise to the Cruising Height when the ***Slow*** arrow button is depressed in the Motion Control Window.
X, Y Velocities	Set the Velocities for the Slow, Medium, and Fast X and Y arrow buttons. The velocity values are in mm per second. When a Slow, Medium, or Fast arrow button is depressed on the Motion Control Window the device(s) selected on the Toolbar will move at the specified velocity. The

fast velocity is used when a CAP is moved to an absolute position.

NOTE: The X-Y axis resonant velocities are between 0.05 and 0.1 mm/sec. To avoid unwanted vibration, these speeds should be avoided.

Z Velocities

Set the Velocities for the Slow, Medium, and Fast Z arrow buttons. The velocity values are in mm per second. When a Z-axis button is pressed the selected probe will move at the specified velocity.

NOTE: The Z axis resonant velocities are between 0.05 and 0.1 mm/sec. To avoid unwanted vibration, these speeds should be avoided.

Set Machine Home...

Set the current position as the CAP's machine home position. A message box will be displayed asking if the CAP is at its machine home position. The machine home position is the centre of travel in each axis. If the **YES** button is pressed the current position will be set as the machine home position. Since there are no limit switches in the CAP to limit its travel, the machine home position will be used to calculate the software limits used to limit the X-Y-Z travel of the CAP.

NOTE: The travel limits will be +/-12mm in X and Y and +/-3mm in Z. These reduce the full range of motion of the CAP slightly so that the CAP is less likely to run into its mechanical limits of travel.

Align

Display the CAP's Align dialog box. This dialog box is used to perform the 3-point alignment for the selected CAP.

2.8.3 CAP 3-Point Alignment Procedure

The 3-Point Alignment Procedure is used to compensate for the misalignment of a CAP to the Device Under Test (DUT). The 3-Point Alignment Procedure can compensate for misalignment in X and Y as well as in Z by defining a new coordinate system for the CAP that is identical to the DUT's coordinate system. The new coordinate system is calculated using three points entered using the Align dialog box. When a CAP is aligned, all three axes can move together to compensate for any misalignment.

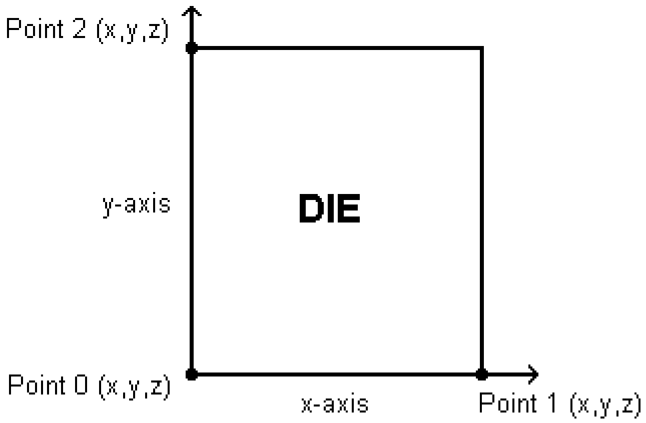


Figure 2-10 CAP 3-Point Alignment

To Perform a CAP 3-Point Alignment:

1. Position the CAP on the platform at the desired location.
2. Select the **Align** button in the motion control window. This will display the Align dialog box.

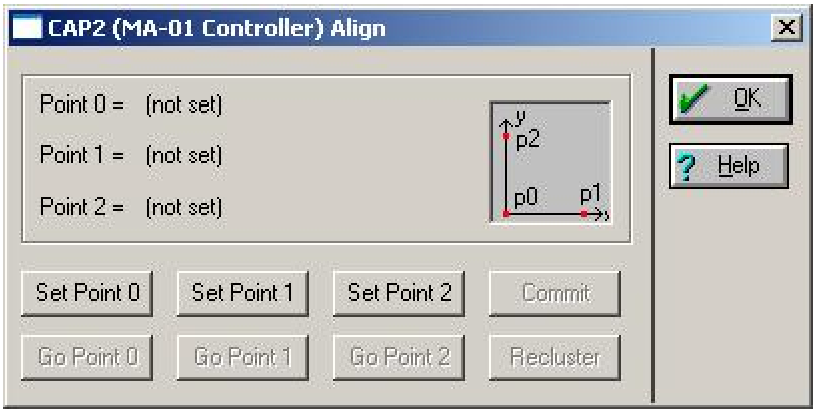


Figure 2-11 CAP Align Dialog Box

3. Using the Motion Control window, position the probe needle at the lower left corner of the die. Move the probe down in Z so that it just touches the wafer.
4. Depress the **Set Point 0** button. The current position is entered as the new origin of the CAP coordinate system.

5. Move the CAP along the edge of the die in the X-axis for approximately 5 mm, adjusting the Y position as needed to keep the needle tip in line with the edge of the die. Move the probe down in Z so that it just touches the wafer.

6. Depress the **Set Point 1** button. The current position and the position of point 0 are used to calculate the new X-axis vector.

7. Depress the **Go Point 0** button to return to point 0 and Move the CAP along the edge of the die in the Y axis for approximately 5 mm, adjusting the X position as needed to keep the needle tip in line with the edge of the die. Move the probe down in Z so that it just touches the wafer.

8. Depress the **Set Point 2** button. The current position and the position of point 0 are used to calculate the new Y-axis vector.

9. Depress the **OK** button to save the current alignment information.

Figure 2-12 CAP-4000 Status Dialog Box

NOTE	**If the probe station's chuck is moved the CAP may no longer be aligned properly to the DUT in the Z-axis.**

To Cancel the CAP alignment:

1. Depress the **Recluster** button followed by the **OK** button.

2. The alignment information for the CAP is reset to the previous values (if they exist).

2.9 Programmable Microscope Mounts (PMM)

The PMM is Wentworth's programmable microscope mount that provides 100mm of travel in the X and Y axis with a step size of 0.01 micron.

LabMaster controls the PMM's movement using the Wentworth PC-Based Stepper Motor Controller boards and Motor Driver Box. An optional attachment for motorized Z focus is also available. The PMM Hardware Setup dialog box is used to select the X-Y or X-Y-Z controllable microscope mounts.

2.9.1 PMM Motion Control Window

The PMM Motion Control window is used to control the motion of the PMM's X, Y axes and the Z-axis if the microscope is equipped with a Wentworth motorized focusing attachment. The arrow buttons are used to move the PMM at the speeds specified by the velocity values which are entered by right clicking on the High, Medium and Slow buttons and entering the speeds required. The PMM's X-Y-Z Motion Control window is pictured below.

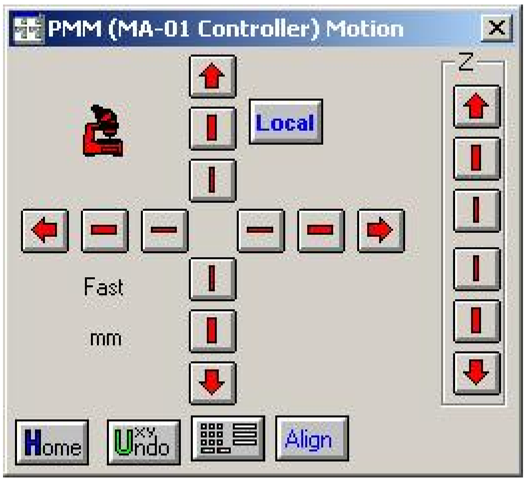


Figure 2-13 PMM Motion Control Window

See Pegasus Motion Control Window, for a functional description of the Motion Control window buttons.

2.9.2 Changing the PMM's Setup Parameters

The PMM's setup parameters are altered by using the PMM Motion Control Window. When the Motion Control Window is visible, you can alter such items as the X, Y, and Z velocities and find the machine home position. When an item in the PMM Motion Control Window is altered it is stored on the disk and so that it can be recalled the next time LabMaster is run.

The PMM Motion Control Window is used to configure the following device parameters of the PMM.

Select	To
X Y Motion buttons	Set the velocities for the Slow, Medium, and Fast X and Y axis arrow buttons. The velocity values are in mm per second. When a Slow, Medium, or Fast arrow button is depressed on the Motion Control dialog box the device(s) selected on the Toolbar will move at the specified velocity. The

fast velocity is used when a PMM is moved to an absolute position.

NOTE: The X-Y axis resonant velocities are between 0.05 and 0.1 mm/sec. To avoid unwanted vibration, these speeds should be avoided.

Z Motion Buttons Set the velocities for the Slow, Medium, and Fast Z axis arrow buttons. The velocity values are in mm per second.

NOTE: The Z axis resonant velocities are between 0.3 and 0.6 mm/sec. To avoid unwanted vibration, these speeds should be avoided.

Find Machine Home... Find the PMM's machine home position. A message box will be displayed asking if the microscope has been raised from the stage. If the **YES** button is pressed the PMM will find the limits of travel in the X and Y axis and then move to the machine home position.

NOTE: At the end of a successful Find Machine Home procedure a message box will be displayed asking if the PMM should be moved back to its original XY starting position. If the **YES** button is pressed the PMM will be moved back to that location.

Align PMM... Display the PMM's Align dialog box. This dialog box is used to perform the 3-point alignment for the PMM. See the CAP-4000 Align procedure for more information.

2.9.3 PMM 3-Point Alignment Procedure

The 3-Point Alignment Procedure is used to compensate for the misalignment of the PMM to the Device Under Test (DUT). The 3-Point Alignment Procedure can compensate for misalignment in X and Y as well as in Z by defining a new coordinate system for the PMM that is identical to the DUT's coordinate system. The new coordinate system is calculated using three points entered using the Align dialog box. When a PMM is aligned, all three axes can move together to compensate for any misalignment. Because of backlash in the focusing mechanism of the microscope the correction in Z under high magnification may not be accurate.

Select the PMM Align Dialog Box and follow the instructions in Section 2.8.3 to perform the 3-Point alignment for the PMM.

2.10 Temptronic 3010 ThermoChuck®

If the prober is equipped with a hot chuck model TP03010B ThermoChuck® System made by Temptronic Corporation, it can be controlled via GPIB using the National Instruments PCI-GPIB board and LabMaster. Before the hot chuck can be used with LabMaster, the PCI-GPIB board must be installed and configured, and the hot chuck's GPIB address must match the one selected using the Temptronic 3010 Hardware Setup dialog box. See the Temptronic 3010 Operator's Manual for more information on the operation of the ThermoChuck®.

2.10.1 Temptronic 3010 Pop-up Dialog Box

The Temptronic 3010 Pop-up dialog box contains frequently used functions for the hot chuck. It can be activated by clicking the right mouse button on the device's Toolbar button.

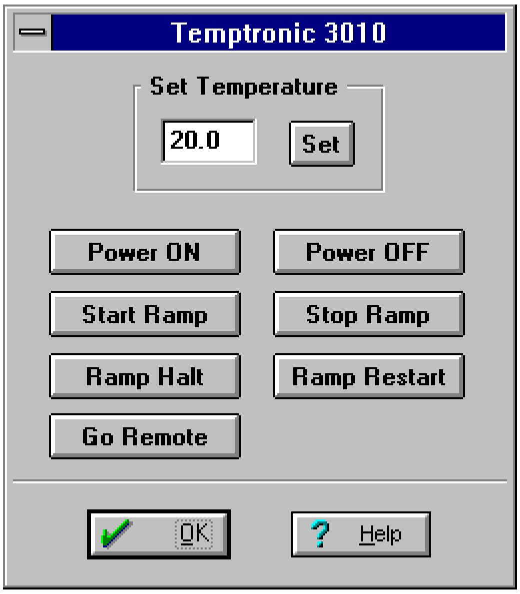


Figure 2-14 Temptronic 3010 Pop-up Dialog Box

The functions provided by the Temptronic 3010 Pop-up dialog box are:

Select	To
Set Temperature	Enter a hot chuck temperature. Press the **Set** button to cause the hot chuck to go to this temperature. **AT TEMP** will be displayed in the Status window and on the 3010's control panel when the hot chuck has reached the temperature.
Power ON	Turn the power to the hot chuck on after the **Power OFF** button was pressed.

Power OFF	Turn off power to the hot chuck. **NOTE:** Only power to the hot chuck itself is removed, the hot chuck's control box will remain on.
Start Ramp	To start the automatic mode. **ACTIVE** will be displayed in the Status window when the automatic mode is running. Automatic mode is used to cycle the DUT through a series of temperatures for specified duration's. The setpoint temperatures, window temperatures, ramp times, and soak times for automatic mode are entered using the Temptronic 3010 Setup dialog box. The 3010's control panel will display the current temperature, setpoint, etc.
Stop Ramp	To stop the automatic mode. **ACTIVE** will be removed from the Status dialog box.
Ramp Halt	To pause the automatic mode after it has been started. **PAUSED** will be displayed in the Status dialog box.
Ramp Restart	To restart the paused automatic mode. **PAUSED** will be removed from the Status dialog box.
Go Remote	To set the Temptronic 3010 to GPIB remote operation. **REMOTE** will be displayed in the Status dialog box and on the 3010's control panel. The 3010 must be in remote for LabMaster to communicate with it properly. It is automatically set to remote when LabMaster is started.

2.10.2 Changing the Hot Chuck's Setup Parameters

The Temptronic hot chuck's automatic mode setup parameters can be altered by using the Temptronic 3010 Setup dialog box. The Setup dialog box is displayed by selecting the hot chuck's name under the Setup menu. Once the setup dialog box is visible, you can alter such items as the setpoint temperatures, window temperatures, ramp times, soak times, beginning ramp number, and the number of cycles.

You can set up to five setpoint temperatures with different window values, ramp times, and soak times for a specified number of cycles. Enter "0.0" for the temperature, window temperature, ramp time, and soak time for the set points you do not wish to use. The automatic mode can be started at the ramp number specified by pressing the **Start Ramp** button in the Temptronic 3010 Pop-up dialog box.

When an item in the Setup dialog box is altered it is downloaded to the Temptronic 3010 using the GPIB interface. It is also stored on the disk so that it can be recalled the next time LabMaster is run. Below is the Temptronic Setup dialog box.

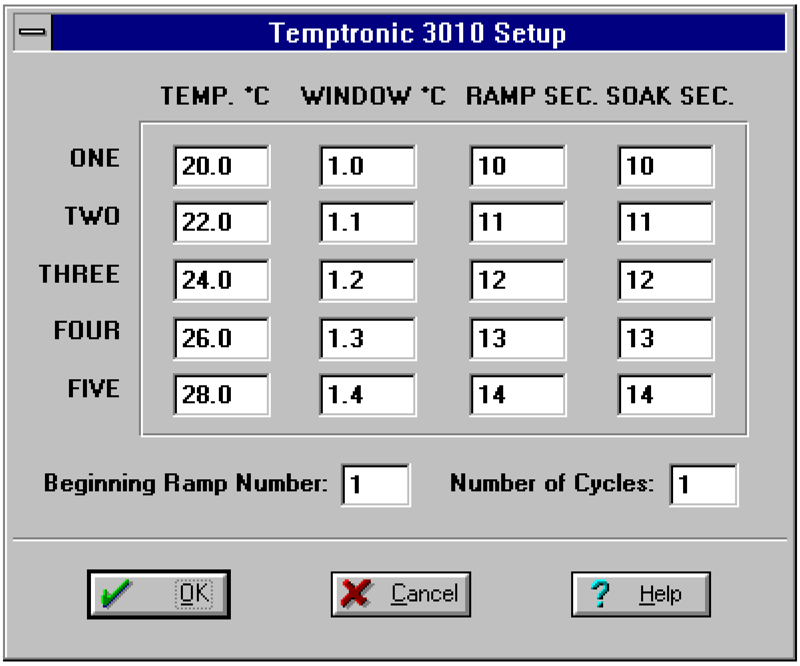


Figure 2-15 Temptronic 3010 Setup Dialog Box

The Temptronic 3010 Setup dialog box is used to configure the following device parameters of the hot chuck.

Select	To
TEMP °C	Enter the temperatures for the different set points. Temperatures must be between in the range of your Temptronic hot chuck.
WINDOW °C	Enter the window temperatures for the different set points. Temperatures must be between 0.1 to 9.9°C.
RAMP SEC.	Enter the ramp times for the different set points. Ramp times must be between 0 to 9999 seconds.
SOAK SEC.	Enter the soak times for the different set points. Soak times must be between 0 to 9999 seconds.
Beginning Ramp Number	Enter the ramp number (1 to 5) you wish the automatic mode to begin.

Number of Cycles	Enter the number of times you wish the automatic mode to cycle through the five set points.

2.11 Temperature Compensation Setup Dialog Box

This dialog box is used to enter the temperature and coefficient of thermal expansion values for the device under test (DUT). This information is used by LabMaster to compensate for the expansion or contraction of the DUT due to temperature variations. For example, when the wafer is heated it will expand by its coefficient of thermal expansion for each degree Celsius increase in temperature. LabMaster needs to take this expansion into account so that the distances LabMaster moves devices will be accurate for the higher temperature. The Temperature Compensation Setup dialog box is displayed by selecting the **Setup | Temperature...** menu item.

If LabMaster is configured for a hot chuck, you can check the **Auto update using Hot Chuck temperature checkbox** to automatically update the device temperature as the hot chuck temperature changes.

NOTE	A temperature of **20°C** (room temperature) should be entered for the Device Temperature if no thermal compensation is desired.

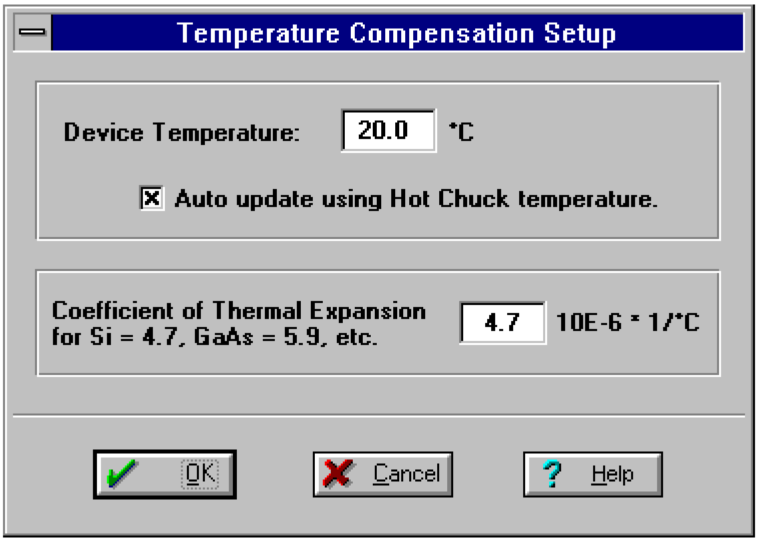
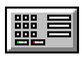


Figure 2-16 Temperature Compensation Setup Dialog Box

2.12 Numerical Entry Dialog Box

The Numerical Entry dialog box is used to move the device(s) selected on the Toolbar to an absolute location. The dialog box is displayed by selecting the  button in the Motion Control window.

When the Numerical Entry dialog box gains input focus (i.e. the dialog box caption bar is highlighted) the current position of the master device is displayed in the axis edit text windows. The current position can be altered using the keyboard or by using mouse and the numeric buttons. Once the correct position is entered, pressing the **GO** button will send the master device to the specified location. If there are any slaves they will track the master's movement. Depressing the **CANCEL** button will remove the Numerical Entry dialog box.

Up to ten positions and position descriptions can be stored per device and recalled using the **STO** and **RCL** buttons respectively. If the Numerical Entry dialog box is removed from the screen, the positions stored for that device will be saved and can be recalled the next time the dialog box is displayed. The stored device positions are not saved when you exit LabMaster.

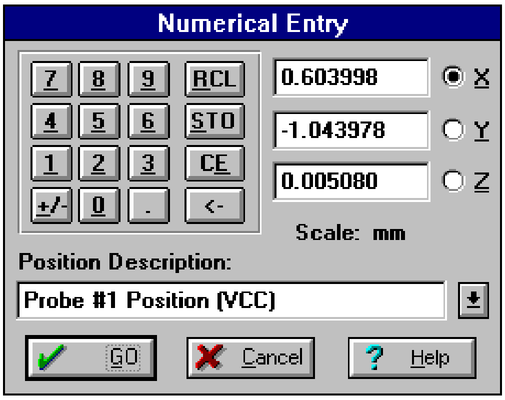


Figure 2-17 Numerical Entry Dialog Box

To STORE a position displayed in the axis' edit text windows:

1. Enter the X, Y, Z position and an optional Position Description.

2. Depress the **STO** button followed by a **0-9** numeric button.

3. The device's position is now stored in the location selected.

To RECALL a previously stored device position:

1. Depress the **RCL** button followed by the numeric button of the position to recall or select the description of the position using the Position Description drop down list box.

2. The recalled position is displayed in the axis edit text windows and the description of the position is displayed in the Position Description edit field.

3. Depress the **GO** button to move the device to the recalled position.

To CLEAR a numerical value displayed in an axis edit text window:

1. Select the edit text window you wish to clear by clicking the mouse on the **X**, **Y**, or **Z** edit text window or radio button.

2. Depress the **CE** button to clear the numerical value in the edit text window.

2.13 New Wave Laser

LabMaster can control the EZLaze, QuikLaze, MiniLase, LCS III, and LCS II lasers from New Wave Research that are purchased with the optional RS232 interface. The Laser must be powered on and placed in 'Remote' mode.

2.13.1 Laser Control Dialog Box

The Laser Control dialog box contains controls used to control the functions of the New Wave Laser if it is selected on the Toolbar. The dialog box is divided into three pages: Trigger, Wavelength, and Marker. The laser controls are described below.

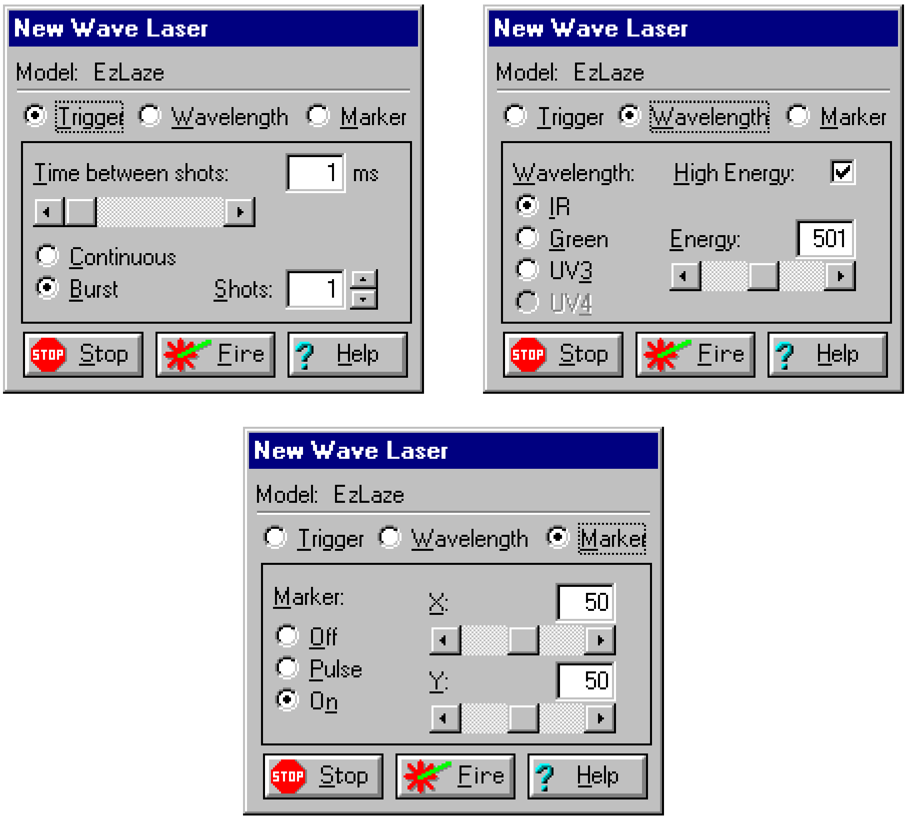


Figure 2-18 Laser Control Dialog Box

Select	To
(Trigger page)	
Burst / Continuous	Set the laser to pulse in **single burst** mode when the fire button is pressed (using the number of shots in burst and time between shots settings) or to begin firing **continuously** at 1 Hz.
Time Between Shots	Set the amount of time in milliseconds that the laser will pause between shots in a burst when pulsed in single burst mode **(0 - 995)**. If a specific value is typed into the edit box, the time between shots will not be changed until the 'Enter' key is pressed.
Shots	Set the number of shots that the laser will fire when pulsed in single burst mode **(1 - 999)**. If a specific value is typed into the edit box, the number of shots in

	a burst will not be changed until the 'Enter' key is pressed.
(Wavelength page)	
Wavelength	Set the wavelength to **GREEN**, **IR** (Infrared), **UV3** (first Ultraviolet wavelength), or **UV4** (second Ultraviolet wavelength). LabMaster determines the current and the available wavelength settings based on the Laser Type selected in the Laser Modify Dialog Box. This function is also available from REXX.
High Energy	Changes the energy range for which the laser's attenuator is operating to high if checked or to low if not checked. This function is also available from REXX.
Energy	Adjust the power level on the laser's attenuator **(0 - 999)**. This function is also available from REXX. If a specific value is typed into the edit box, the attenuator setting will not be changed until the 'Enter' key is pressed.
(Marker page)	
Marker	Set the laser's spot marker to **ON**, **PULSE**, or **OFF**.
X	Adjust the size of the spot marker in the x-axis **(0 - 100%)**.
Y	Adjust the size of the spot marker in the y-axis **(0 - 100%)**.
(All pages)	
Fire	Fire the laser (single burst mode). This function is also available from REXX.
Stop	Abort a command.

2.14 TnP Motorized Turret

LabMaster can now control the TnP Motorized Turret through an optional RS-232 interface. LabMaster's Motorized Turret interface functions similarly to the control box attachment that comes with the turret. LabMaster adds an additional safety feature, however, and raises the microscope before changing the objective on the turret.

2.14.1 Motorized Turret Control Dialog Box

The Motorized Turret Control dialog box is used to control the functions of the TnP Motorized Turret from LabMaster. This dialog box can be displayed by clicking the left mouse button on the Motorized Turret's Toolbar button. To close the dialog box, click the Toolbar button again.

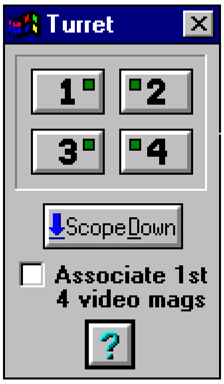


Figure 2-19 Motorized Turret Control Dialog Box

Select	To
1, 2, 3, 4	Select an objective on the turret. LabMaster will raise the microscope before rotating the turret.
Scope Down	Lower the microscope to its original position after a turret rotation is complete.
Associate 1st 4 video mags	Link the four objective positions on the Motorized Turret to LabMaster's first four video magnification levels. This will allow LabMaster to automatically switch the current magnification level as the turret is rotated.

2.15 A-ZOOM Microscope

LabMaster can control the Ready Products A-ZOOM Microscope through an optional RS-232 interface. LabMaster's A-ZOOM Microscope interface functions similarly to the control box attachment that comes with the A-ZOOM. Most of the A-ZOOM functions can also be controlled through REXX programming.

2.15.1 A-ZOOM Microscope Control Dialog Box

The A-ZOOM Microscope Control dialog box is used to control the functions of the A-ZOOM from LabMaster. This dialog box can be displayed by clicking the left mouse button on the A-ZOOM Microscope Toolbar button. To close the dialog box, click the Toolbar button again.

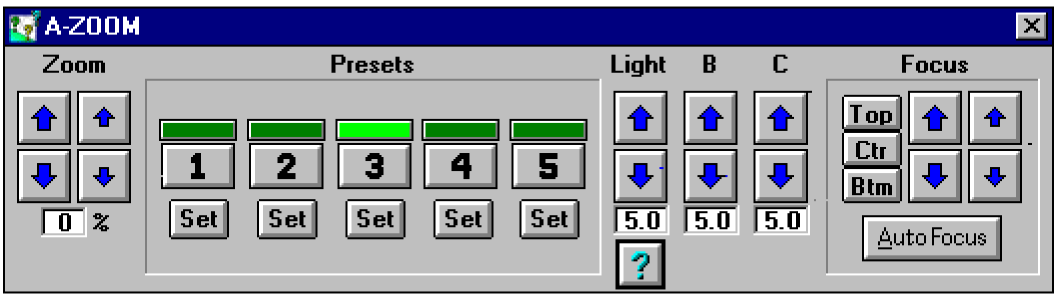
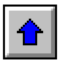
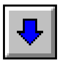
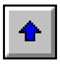
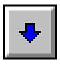
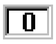


Figure 2-20 A-ZOOM Microscope Control Dialog Box

Select	To
	Increase the setting for the labelled control by 10 for the Zoom & optional Focus or by .5 for the Lights (including the optional B & C lights).
	Decrease the setting for the labelled control by 10 for the Zoom & optional Focus or by .5 for the Lights (including the optional B & C lights).
	Increase the setting for the labeled control by 1.
	Decrease the setting for the labelled control by 1.
	Set the associated control to a specific value. The change will not occur until the 'Enter' key is pressed.
1, 2, 3, 4, 5	Select the corresponding Preset and move the A-ZOOM to the stored Zoom and Light settings. The light above each Preset will illuminate when the A-ZOOM is at the current Zoom and Light settings for that particular Preset.
Set	Save the current Zoom and Light settings as the corresponding Preset.
Top, Ctr, Btm	Move the optional A-ZOOM focus assembly to the top, centre, or bottom position.
Auto Focus	Pulse the optional A-ZOOM auto-focus output low for 50ms.

2.16 Setup Options

The **Setup | Options** menu item, generates a dialog box containing several items that let the operation of LabMaster be customized.

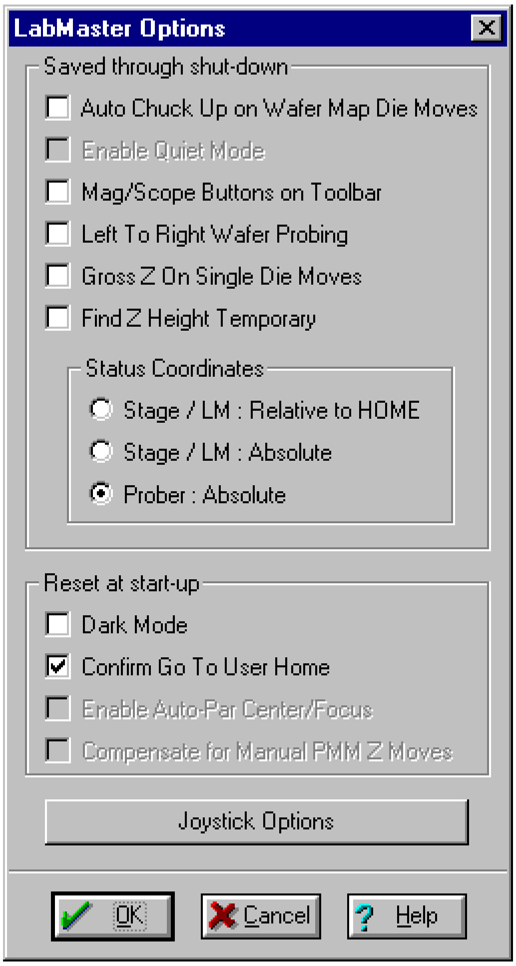


Figure 2-21 Setup | Options Menu Items

2.16.1 Auto ChuckUp On Wafer Map Die Moves

When **Auto ChuckUp On Wafer Map Die Moves** is checked, the chuck will automatically be moved up after a die move is executed. Die moves are the NextDie, PreviousDie, FirstDie, LastDie, and HomeDie commands as well as when the right mouse button is used in the Wafer Map Zoom or Top view windows to select a die to move the chuck to.

2.16.2 Enable Quiet Mode

When **Enable Quiet Mode** is checked, display a check box in the Motion Control Dialog Box which can be used to take the system in and out of Quiet Mode. When the system is in Quiet Mode, the power to the motor drivers is turned off in order to reduce the noise generated by the stepper motors and facilitate low-level testing. This setting is remembered through a LabMaster shutdown.

2.16.3 Mag / Scope Buttons on Toolbar

When **Mag / Scope Buttons on Toolbar** is checked, display **Scope Up** & **Scope Down** buttons, and the magnification level selections found under the Setup | Magnification menu item on the toolbar for easier access. The button representing the current magnification level will be indicated by a darker outline. This setting is remembered through a LabMaster shutdown.

2.16.4 Left To Right Wafer Probing

When **Left To Right Wafer Probing** is checked, causes the chuck to only probe die on the Wafer Map from left to right. This setting is remembered through a LabMaster shutdown.

2.16.5 Gross Z On Single Die Moves

When **Gross Z On Single Die Moves** is checked, causes the chuck to automatically move gross down and up between Wafer Map die moves, when the prober is set up to do Single Die running. This setting is remembered through a LabMaster shutdown.

2.16.6 Find Z Height Temporary

When **Find Z Height Temporary** is checked, causes the "**Find Z Height"** procedure to NOT save the new Gross Height on the prober. This setting is remembered through a LabMaster shutdown.

2.16.7 Status Coordinates

The setting selected for **Status Coordinates** decides how X and Y axis positions are shown in the Status Window. The alternative settings are:-

LabMaster Stage position relative to the HOME position
LabMaster Stage position absolute
Prober position absolute

This setting is remembered through a LabMaster shutdown.

2.16.8 Confirm Goto User Home

When **Confirm Goto User Home** is checked the user will be asked to verify that they want to move to the User Home position when they click on the Home button in the Motion Control dialog box. This setting is intentionally not remembered when LabMaster shuts down. Each time LabMaster starts executing this parameter will be set so that confirmations are required.

2.16.9 Enable Auto-Par Center / Focus

When **Enable Auto-Par Center / Focus** is checked, enables the Auto-Par Center/Focus function for the PMM. After the Auto-Par Center/Focus Calibration Procedure has been completed, LabMaster will automatically move the PMM to adjust for differences in the centres and the focus points of the objectives on the microscope. Each time LabMaster starts executing, this option will be set to active. This option is only available in systems that have a PMM.

2.16.10 Compensate for Manual PMM Z Moves

When **Compensate for Manual PMM Z Moves** is checked, verify PMM absolute Z moves. LabMaster will ask the user if the PMM has been moved manually before executing an absolute move in the PMM Z axis (i.e. moves initiated from the Numeric Entry Dialog Box). If the user answers **Yes**, LabMaster will then re-establish the Z Machine Home position before actually making the move. This option is only available when the PMM Z axis is controllable and has been built with hardware limits. This setting is intentionally not remembered when LabMaster shuts down. Each time LabMaster starts executing, this option will be set to active. This option is only available in systems that have a PMM.

2.16.11 Joystick Options

The **Joystick Options** item, displays the Joystick Setup Dialog Box. This dialog box is used to select a function for the user-defined joystick button and to set the joystick enable time. This option is only available in systems that have a joystick.

3 Video Window

3.1 Introduction

The LabMaster Video window displays real-time video from the camera attached to the microscope by using a FlashPoint 4XL card as the video card, thereby making a second monitor unnecessary.

The FlashPoint card is capable of displaying real-time video by overlaying the real-time video wherever it sees a certain colour "key". LabMaster uses the colour magenta as the colour key to display the real-time video from the FlashPoint board. If another window containing the colour magenta is placed on top of the real-time video window, the real-time video will be shown where that colour appears.

Any image shown in the LabMaster Video window can be saved to disk in a variety of image formats, or copied to the Windows clipboard for pasting into other Windows applications. ***Important****: This must be done using LabMaster, because all Windows utilities (for example, the Print Screen key) will only copy the magenta colour key, and not the video image.*

Calibrating the Video window to the current microscope magnification allows devices to be moved by using the real-time video image and the mouse, and structures on the wafer to be measured using the Video Ruler.

3.2 Calibrating the Video Window

The Video window Calibration procedure is used to calculate the relative pixel size for a given microscope magnification value. The calculated pixel size is used by the Video Ruler to calculate the size of objects that are measured in the Video window, and is also used to calculate the distance to move a device(s) when moved using the mouse and the Video window.

Each microscope objective should be calibrated using the Video Calibration procedure described below. Each time a microscope objective is changed the corresponding magnification value should be selected via the **Setup | Magnification** menu item so that the correct calculated pixel size is used.

If LabMaster is controlling an A-ZOOM Microscope, the user can choose to have LabMaster automatically adjust the video calibration data to compensate for the current zoom level. This automatic adjustment works best when the user has performed the video calibration procedure at a higher zoom factor.

If LabMaster is controlling a Motorized Turret, the user can choose to have LabMaster associate the first four magnification values with the four objective positions on the turret. This will allow LabMaster to automatically switch the magnification when the turret is rotated.

To Calibrate the Video window for a known magnification:

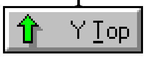
1. Select the **Video | Calibrate...** menu item. The Video Calibrate dialog box will be displayed and all active devices will be deactivated on the Toolbar

2. Using the Motion Control dialog box align an object or line on the wafer with the left inside edge of the Video Calibrate dialog box. Depress the 

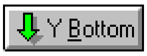
button.

3. Using the Motion Control dialog box align the same object or line on the wafer with the right inside edge of the Video Calibrate dialog box. Depress the 

button.

4. Using the Motion Control dialog box align an object or line on the wafer with the top inside edge of the Video Calibrate dialog box. Depress the 

button.

5. Using the Motion Control dialog box align the same object or line on the wafer with the bottom inside edge of the Video Calibrate dialog box. Depress the 

button.

6. Depress the **OK** button. The Magnification dialog box will be displayed.

7. Enter the current microscope magnification value and depress the **OK** button. LabMaster will allow up to six characters/numbers to identify the magnification level. If LabMaster is controlling an A-ZOOM Microscope and you would like LabMaster to automatically adjust the video calibration data to compensate for the current zoom level, check the automatic adjustment check box. If this box is checked, the word '**Auto**' will appear after the magnification value in the video related LabMaster menu items. Depress the **OK** button.

8. The Video window is now calibrated for this microscope magnification.

3.3 Using the Video Ruler

Once the Video window has been calibrated for the current microscope objective using the Video Calibrate procedure described above, the Ruler can be used to measure items in the Video window. The video calibration procedure is used to calculate the display's pixel size relative to the microscope magnification for use in calculating the dimension of items measured with the Ruler. The accuracy of the measured value depends the magnification value and on how well the Video Calibration procedure was performed.

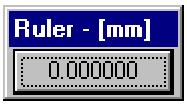


Figure 3-1 Video Ruler Window

To measure items displayed in the Video window:

1. Select the **Video | Ruler...** menu item. The Ruler dialog box will be displayed and the cursor will change to the ruler cursor. The **Video | Ruler...** menu item will now be checked.

2. Depress the button in the Ruler dialog box that displays the measured value to select between the millimetre, inch, micron, or mil measurement modes. The measurement mode is indicated in the Ruler dialog box caption bar.

3. Depress the left mouse button to select the starting measurement point.

4. While continuing to hold down the left mouse button, drag the mouse to the end measurement point.

5. Release the left mouse button. The measured value will be displayed on the button in the Ruler dialog box.

To turn off the Video Ruler:

1. Select the checked **Video | Ruler...** menu item.

3.4 Device Movement using the Video Window

The active device(s) can be moved via the mouse by using the Video window if the Video window Calibration procedure has been previously performed for the current microscope magnification.

To move a prober, or PMM via the Video window:

1. Select the device to be moved on the Toolbar.

2. Click the **right** mouse button on the Video image where the device is to be moved.

3. When the right mouse button is released, the device will moved so that the specified location will appear in the centre of the video image.

To move a CAP via the Video window:

1. Select the CAP to be moved on the Toolbar.

2. Click the **right** mouse button on the probe tip.

3. While continuing to hold the right button down, drag the mouse to the location on the Video image where the probe tip is to be moved.

4. When the right mouse button is released, the selected CAP will move to the specified location on the video image.

3.5 Manipulating the Video Image

The image displayed in the Video can be manipulated in many ways. This allows for the storage of images on the computer's disk and the copying of images into other Windows programs. The Video window provides a powerful failure analysis tool that allows for the pasting of images of probe card scrub marks or wafer defects into applications such as Microsoft Word for Windows for use in documentation. The various ways to manipulate the real-time Video image are described below.

3.5.1 Freezing the Video Window

Freezing the real-time Video window will permit the cutting or copying of a selected area of the video image to the Windows Clipboard or to a file. See Section 4.5.4 "Cutting, Copying, and Pasting Images" for more information.

To freeze the real-time Video window:

1. Select the **Video | Freeze** menu item.

2. The video will now be frozen and a check mark will be displayed next to the **Video | Freeze** menu item.

To unfreeze the Video window:

1. Select the **Video | Freeze** menu item.

2. The Video window will now display video in real-time and the check mark will be removed from the **Video | Freeze** menu item.

3.5.2 Saving Images to Disk

LabMaster has the capability of saving images displayed in the real-time Video window to a file in various image formats. If the video image is frozen a section of the video image can be selected and saved to a file.

NOTE	Many image formats require nearly a megabyte of disk space to store an entire full screen image. Use one of the compressed image formats to reduce the size of the stored image. See the Video File Formats section for a description of the supported video formats.

To select a portion of video and save it to a file:

1. If the video ruler is ON, select the **Video | Ruler...** menu item to remove the Ruler.

2. If the video image is not already frozen, select the **Video | Freeze** menu item to freeze the real-time video. The video will now be frozen and the cursor will change to video selection cursor.

3. Press the left mouse button to select the beginning point of the selecting rectangle.

4. While continuing to hold down the left mouse button, drag the mouse to size the rectangle.

5. Release the left mouse button when the rectangle is the appropriate size. The rectangle encloses the area of video that will be saved.

6. Select the **File | Save Image...** menu item. The Video File Save dialog box will be displayed.

7. Select the desired image format in the Video File Save dialog box.

8. Enter the filename and depress the **OK** button.

9. The image selected will be saved as the filename specified.

To save the entire video image to a file:

1. If there is an area previously selected on the Video window, select the **Video | Freeze** menu item to unfreeze the real-time video. This will reset the selected area back to the full Video window size.

2. Select the **File | Save Image...** menu item. The Video File Save dialog box will be displayed.

3. Select the desired image format in the Video File Save dialog box.

4. Enter the filename and depress the **OK** button.

5. The image selected will be saved as the filename specified.

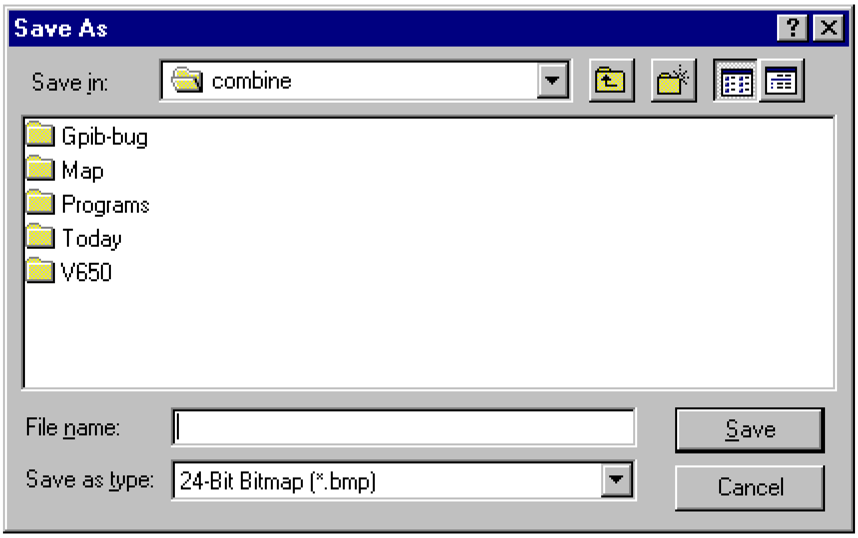


Figure 3-2 Video File Save Dialog Box

3.5.3 Loading Images From Disk

LabMaster has the capability of loading still images stored in various image formats and displaying these images in the Video window.

To load an image file:

1. Select the **Video | Open Image...** menu item. The Video Load Image dialog box will be displayed.

2. Select the filename and image format of the image to be loaded and depress the **OK** button.

3. The image will be displayed in the Video window. To return to real-time video select the **Video | Freeze** menu item to unfreeze the video.

To load an image file onto a selected area of the video image:

1. If the video ruler is ON, select the **Video | Ruler...** menu item to remove the Ruler.

2. If the video image is not already frozen, select the **Video | Freeze** menu item to freeze the real-time video. The video will now be frozen and the cursor will change to video selection cursor.

3. Press the left mouse button to select the beginning point of the selecting rectangle.

4. While continuing to hold down the left mouse button, drag the mouse to size the rectangle.

5. Release the left mouse button when the rectangle is the appropriate size. The rectangle encloses the area where the video will be loaded.

6. Select the **File | Open Image...** menu item. The Video Load Image dialog box will be displayed.

7. Select the filename and image format of the image to be loaded and depress the **OK** button.

8. The image will be displayed in the selected area of the Video window. To return to real-time video select the **Video | Freeze** menu item to unfreeze the video.

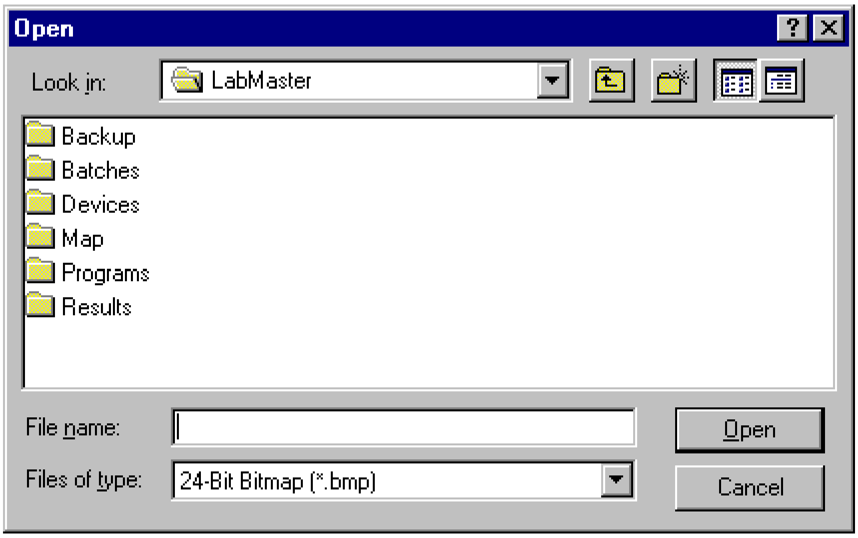


Figure 3-3 Video File Open Dialog Box

3.5.4 Cutting, Copying, and Pasting Images

LabMaster is capable of copying images to the Windows Clipboard for use by other Windows applications or for later use by LabMaster. Images are sent to the Windows Clipboard in the Windows 24-bit colour bitmap format. Only Windows 24-bit colour bitmaps can be pasted into the LabMaster's Video window. A full size video image will require nearly one megabyte of memory for use by the Clipboard for image storage.

To Copy a *portion* of the video image to the clipboard:

1. If the Ruler is ON, select the **Video | Ruler...** menu item to remove the Ruler.

2. If the video image is not already frozen, select the **Video | Freeze** menu item to freeze the real-time video. The video will now be frozen and the cursor will change to video selection cursor.

3. Press the left mouse button to select the beginning point of the selecting rectangle.

4. While continuing to hold down the left mouse button, drag the mouse to size the rectangle.

5. Release the left mouse button when the rectangle is the appropriate size. The rectangle encloses the area of video that will be saved.

6. Select the **Edit | Copy** menu item.

7. The image selected is copied to the Windows clipboard.

To Copy the *entire* video image to the clipboard:

1. If the video image is frozen, select the **Video | Freeze** menu item to unfreeze the real-time video. This will remove the video selection rectangle if visible.

2. Select the **Edit | Copy** menu item.

3. The image selected is copied to the Windows clipboard.

To Paste a 24-bit image from the clipboard onto the *entire* Video window:

1. If the video image is frozen, select the **Video | Freeze** menu item to unfreeze the real-time video. This will remove the video selection rectangle if visible.

2. Select the **Edit | Paste** menu item.

3. The image selected is displayed in the Video window.

To Paste a 24-bit image from the clipboard onto a selected area of the video image:

1. If the video ruler is ON, select the **Video | Ruler...** menu item to remove the Ruler.

2. If the video image is not already frozen, select the **Video | Freeze** menu item to freeze the real-time video. The video will now be frozen and the cursor will change to video selection cursor.

3. Press the left mouse button to select the beginning point of the selecting rectangle.

4. While continuing to hold down the left mouse button, move the mouse to size the rectangle.

5. Release the left mouse button when the rectangle is the appropriate size. The rectangle encloses the area where the video will be pasted

6. Select the **Edit | Paste** menu item.

7. The image is pasted in the area selected.

3.6 Video Setup Dialog Box

The Video window Setup dialog box is used to alter the display parameters of the real-time video in the Video window. The **Brightness**, **Saturation**, **Contrast**, **Hue**, and **Sharpness** of the video can be altered. The real-time video can be displayed in black and white or colour by checking or un-checking the **B&W check box**.

Use the scroll bars to select the Brightness, Saturation, Contrast, Hue, and Sharpness level from 0 to 100 of the video.

If the **OK** button is depressed on the Video window Setup dialog box the changes made to the real-time video will be saved to disk. If the **CANCEL** button is depressed the changes made to the real-time video are ignored.

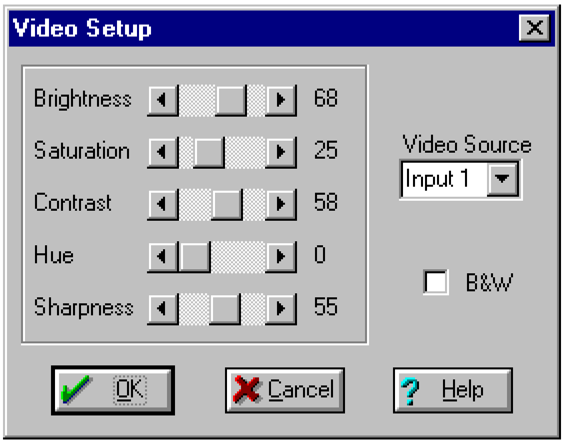


Figure 3-4 Video File Open Dialog Box

3.7 Video File Formats

LabMaster is capable of grabbing a single frame of real-time video and then saving it to a file in one of the following formats:

Windows 24-bit colour bitmap:

This is the 24-bit/pixel colour Windows bitmap format. Other Windows programs cannot display it properly unless a 24-bit colour VGA card is installed. LabMaster can display this format.

Windows 256 colour bitmap:

When saved in this format, the 24-bit true-colour image displayed by LabMaster will be converted to the Window's 256-colour format. Some colour information will be lost due to the conversion from 24-bits to 8-bits per pixel. LabMaster cannot display this format.

TIFF:

This is a 24-bit/pixel colour format that can be saved and recalled by LabMaster.

Targa:

This is a 24-bit/pixel colour format that can be saved and recalled by LabMaster.

JPEG:

This is the Joint Photographic Experts Group (JPEG) compressed file format. When an image is saved in this format, the LabMaster 24-bit colour image will be compressed according to the JPEG standard. A very high degree of compression can be obtained using this format, with only a small loss in image quality. LabMaster can display this format.

4 Wafer Map

4.1 Introduction

The LabMaster Wafer Map window is a powerful failure analysis tool that can be used for device navigation and positioning, and for displaying and storing die-binning information. By using the Wafer Map window for device navigation and movement, the user can quickly position the chuck to any die on the wafer and position the PMM and CAPs to any position within that die. Devices are moved "graphically" using the mouse to position the device at its new location.

Before you can use the Wafer Map window, you must have first purchased a version of LabMaster that has the optional wafer map capability. Then a new wafer map must be created or a previously stored wafer map must be loaded from disk. See Section 5.3 Creating a New Wafer Map for more information. Wafer maps are stored on disk in a tab-delimited text format that can easily be imported into Microsoft Excel, Lotus 123, or other applications that accept tab-delimited input.

4.2 Wafer Map Window

The Wafer Map Window is activated by selecting the File **| Wafer Map** menu item. The Wafer Map window is divided into three child windows and a toolbar. Each child window gives the user a different graphical representation of the wafer map (See Figure 5-1).

The Top View window gives the user an overall depiction of the wafer map. Using this window and the mouse, the area viewed in the Zoom View window can be selected and the chuck can be sent to any position on the wafer.

The Zoom View window gives the user a detailed view of a portion of the wafer map. This window allows you move the chuck to any die location by using the mouse. You are also able to add and delete die, mark die to be tested, and alter die specific information easily by using the Zoom View window.

The Die View window shows the positions of any PMM or CAPs within the current die. These devices can be moved by "dragging" the device to a new location using the mouse.

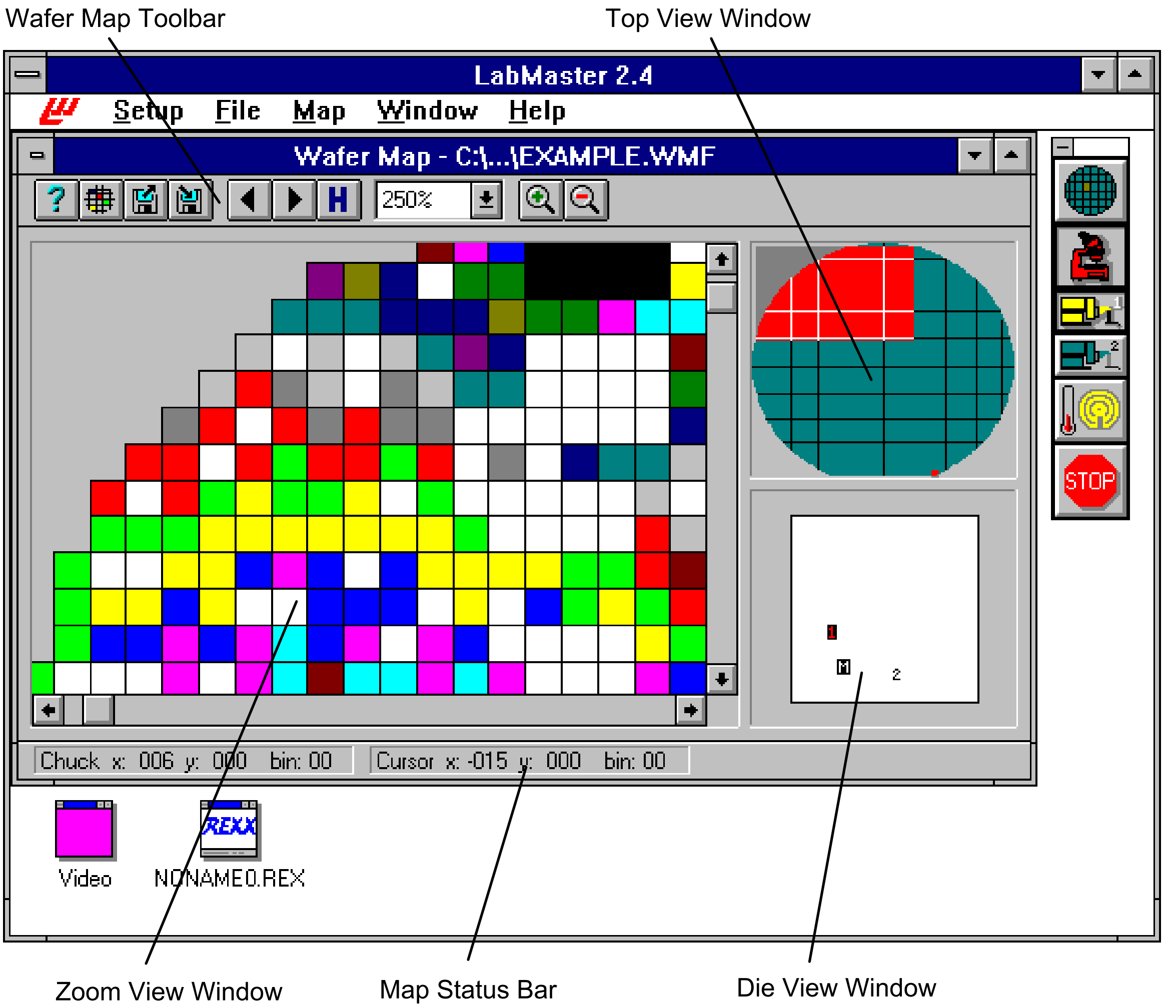


Figure 4-1 Video File Open Dialog Box

4.2.1 Wafer Map Toolbar

The Wafer Map toolbar contains buttons for frequently used Wafer Map functions.

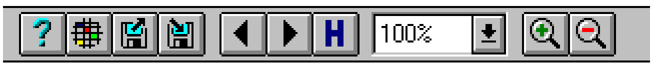
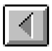


Figure 4-2 Wafer Map Toolbar

Select		To
	Help	Display the Wafer Map help contents.
	Map Setup	Display the Wafer Map Setup dialog box. If a wafer map is not currently loaded, a beep will sound if the button is selected.

	Open	Display the File Open dialog box, allowing the selection of a wafer map to be loaded.
	Save	Save the currently loaded wafer map.
	Previous Die	Move the probe station's chuck to the previous test die location. If there is no previous test die a beep will sound if the button is selected.
	Next Die	Move the chuck to the next test die location. If there is no next test die a beep will sound if the button is selected.
	Home Die	Move the chuck to the home die location.
	Zoom Percent	Select either a 100%, 200%, 300%, 400%, 500% zoom factor or the previous selected zoom percentage.
	Zoom In	Increase the zoom factor of the Zoom View window by 50%.
	Zoom Out	Decrease the zoom factor of the Zoom View window by 50%.

4.2.2 Top View Child Window

The Top View window shows the top view of the wafer map and a rectangle that depicts the portion of the map that is shown in the Zoom View window. This rectangle can be repositioned by dragging it to a new location using the left mouse button. Once this is done, the Zoom window will be updated with the new zoomed area indicated by the rectangle.

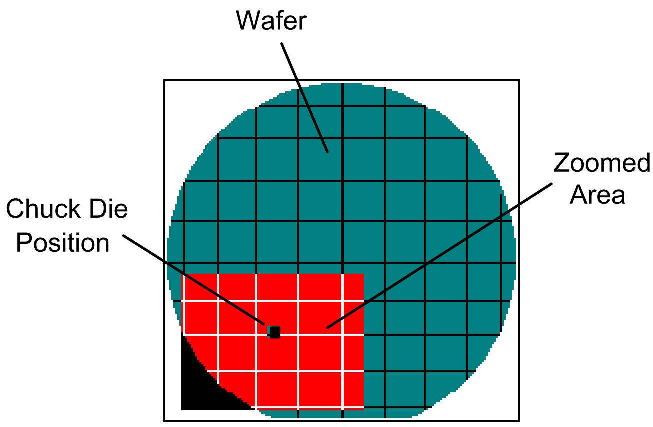


Figure 4-3 Top View Child Window

To move the chuck to a point on the wafer map:

- Click the right mouse button on the area in the Top View window you wish the chuck to move to. The probe station's chuck will move to die closest to this location. The current position of the chuck is indicated by a flashing point updated when the chuck is moved.

4.2.3 Zoom View Child Window

The Zoom View window shows an enlarged area of a wafer map. The zoom factor in percent is shown in the toolbar, along with buttons to increase or decrease the zoom magnitude. The zoom magnitude can range from 100% to 9950%. The zoomed area can be moved to a different area on the wafer map by using the scroll bars on the bottom and right of the Zoom View window, or by using the Top View window.

Each die in the Zoom View window is filled with the colour assigned to the bin number for that die. Colours can be assigned to each bin number by using the Colour Setup dialog box, which is displayed by pressing the **Bin Colours...** button in the Wafer Map Setup or Die Setup dialog boxes. Test structures and partial die are collared black. Skip die are marked with an 'X', ink-only and forced-ink die are marked with a '/', sample die are marked with an 'S', and alignment die are marked with a '+'.

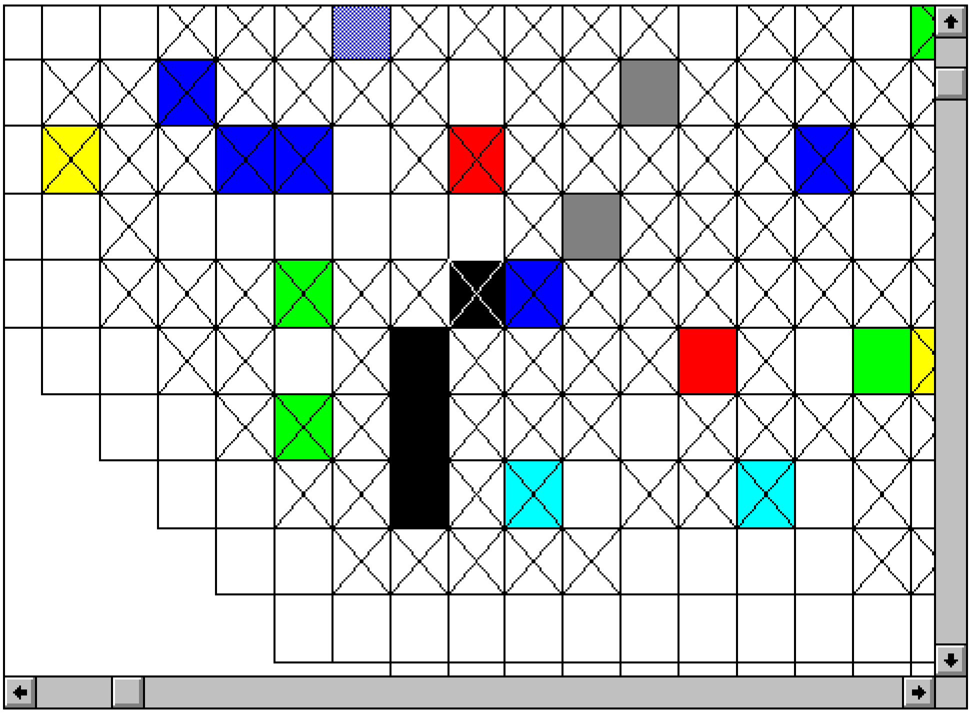


Figure 4-4 Zoom View Child Window

To display the Die Setup dialog box:

- The Die Setup dialog box can be displayed for any die by double clicking the left mouse button on that die.

To mark a die as a skip or test die:

- A die can be marked as a die to be skipped when testing by holding SHIFT key down and clicking with the left mouse button on that die, or by checking the Skip Die check box in the Die Setup dialog box.

To add or remove a die:

- A die can be removed or added to the wafer map by holding down the CTRL key and clicking the left mouse button on the die to be removed or the area where you wish a die to be added. You are limited to the number of die you can add to the wafer map by the number of rows and columns entered when the wafer map was created.

To mark a die as a ink-only die:

- A die can be marked as a ink-only die by holding SHIFT key down and clicking with the right mouse button on that die, or by checking the Ink-only Die check box in the Die Setup dialog box.

To mark a die as a sample die:

- A die can be marked as a sample die by holding CTRL key down and clicking with the right mouse button on that die, or by checking the Sample Die check box in the Die Setup dialog box.

To assign a die a bin number:

- A die can be assigned the bin number that was last selected using the Die Setup dialog box by holding down the CTRL and SHIFT keys simultaneously and clicking the left mouse button on the die.

To move the chuck to a die location:

- The chuck can be moved to a die by clicking the right mouse button on any die shown in the Zoom View window. The current die location of the chuck is indicated by the flashing die in the Zoom View and Top View windows and is updated as the chuck is moved.

4.2.4 Die View Child Window

The Die View window is collared the bin colour for the current die where the chuck is positioned. This window also has up to seven small device child windows that show the position of the PMM and CAPs relative to the bottom left corner of the die. Each of these little device windows contains either the letter "M" indicating the microscope (PMM), or a number indicating the CAP number. The number or letter for the device is shown in reverse video if the device is selected on the LabMaster Device Toolbar. If the device is the master it will be shown with a red number or letter, and if the device is a slave it will be shown with a white number or letter.

The Die View window assumes that the PMM and CAPs have had their user home positions set at the lower left hand corner of the die. If this has not been done, then the Die View window will not accurately depict the positions of the PMM or CAPs.

NOTE	If the device child windows are overlapping or are on top of each other, the left mouse button can be used to bring the windows on the bottom to the top.

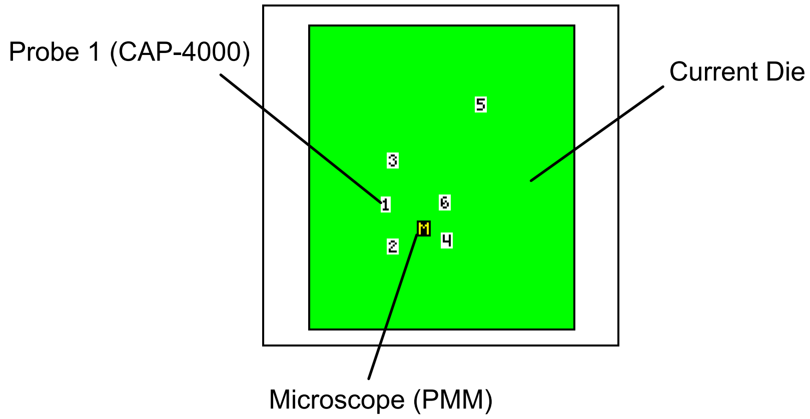


Figure 4-5 Die View Child Window

To move the master and slaves to a new location on the die:

- Using the right mouse button, click on the master device and drag it to its new location on the die. The master device will move to the selected location with the slave(s) tracking the master.

To move a slave to a new location on the die:

- Using the right mouse button, click on the slave and drag it to its new location. Only the slave selected will move to the new location.

To move just the master to a new location:

- While holding down the SHIFT key, use the right mouse button to drag the master to its new location. Only the master will move to the selected location.

To move a device that is not a master or slave:

- Using the right mouse button, click on the device and drag it to its new location. Only the device selected will move to the new location. A device that is neither a master nor slave is one that has not been selected on the LabMaster Device Toolbar.

4.3 Creating a New Wafer Map

Before you can enter a wafer map, you need to know the number of rows and columns of the wafer, the number of die in each row and their offset from the left side of the wafer, and the x and y die sizes. Once you have this information, the wafer map can be created by selecting the **File | New...** menu item.

There are three methods of creating a wafer map. The first has LabMaster create a default circular wafer map to which die can then be added or removed. The second is used to create a wafer map by entering the number of columns and their offset from the left side of the wafer for each row of the wafer map. The third method is to create a square wafer map and use the mouse to delete the dies that are not part of the wafer map. Examples depicting the later two methods are shown in Sections 4.3.1 below.

Follow the steps below for entering the data needed to create the wafer map.

1. Select the **File | New...** menu item. The Wafer Map Setup Dialog Box (Figure 4-10) is displayed.
2. Enter the wafer's name, size, flat/notch location, comment text, X and Y die sizes, and the X and Y street sizes.
3. Depress the **OK** button. The Wafer Map Setup Dialog Box is removed and LabMaster will then ask if it should create a default circular wafer map.

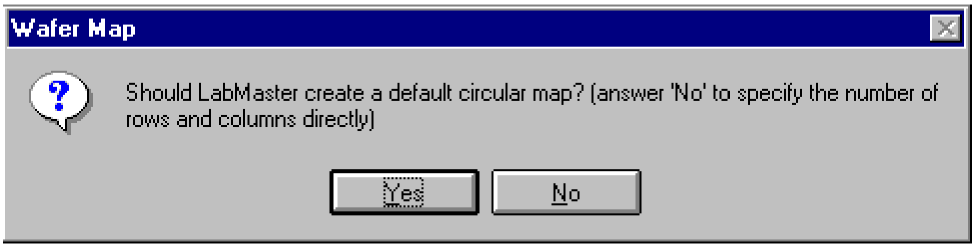


Figure 4-6 Wafer Map Default Circular Map Dialog Box

4. If 'Yes' is chosen, LabMaster will use the wafer size, x and y die sizes, and the x and y street sizes to generate a default circular wafer map. Die can be added or removed from the LabMaster generated map by holding down the CTRL key and using the left mouse button in the Zoom View Window. With the CTRL key held down click on the locations where die should be added or on the die that should be removed.
5. If 'No' is chosen, then LabMaster will prompt for the number of rows and columns for the wafer map using the Wafer Map Row and Column Entry dialog.

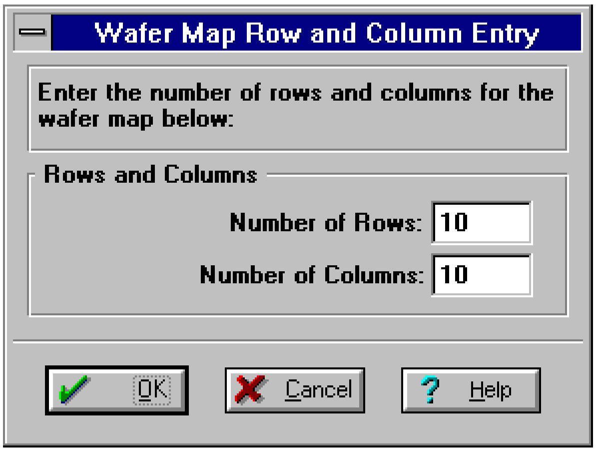


Figure 4-7 Wafer Map Row and Column Entry Dialog Box

6. Enter the number of rows and columns for the wafer map. The number of rows and columns must be consistent with the wafer and die sizes entered in the Wafer Map Setup dialog box. There can be one partial die at each side of the wafer map. Press the **OK** button. The dialog box is removed, and the Wafer Map Define Wafer dialog box is displayed.

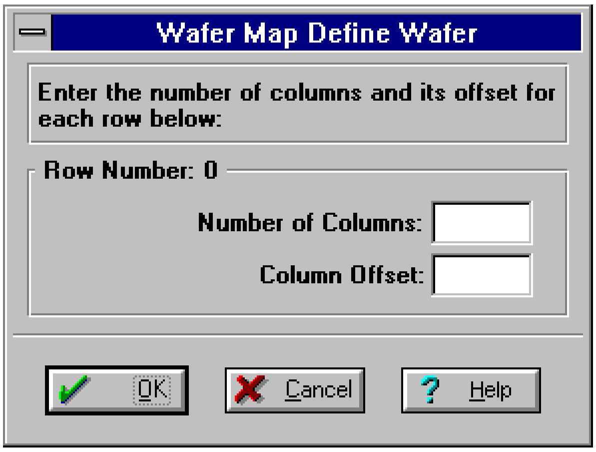


Figure 4-8 Wafer Map Define Wafer Dialog Box

7. Enter the number of columns in the first row (Row number: 0) and the offset of the first column from the left side of the wafer map. Press the **OK** button. Now enter the data for the second row and press the **OK** button. Repeat this for each row of the wafer map. When the last row is entered, the dialog box is removed and the Wafer Map window is updated to display the new wafer map.

4.3.1 New Wafer Map Examples

Example 1: Follow the steps below to enter the 5 X 5 wafer map shown in figure below.

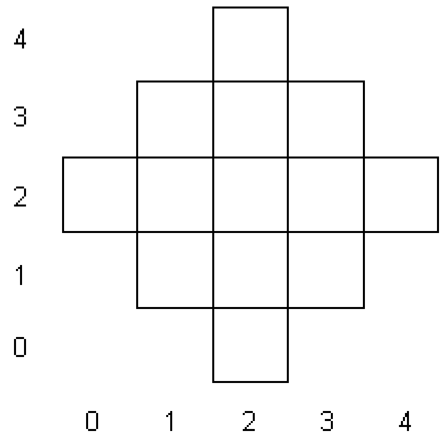


Figure 4-9 Wafer Map Example Wafer

1. If the current scale is not in mm, change the scale to mm by selecting the **Setup | Scale** menu item.

2. Select the **File | New...** menu item. The Wafer Map Setup dialog box (Figure 4-10) is displayed.

3. Enter the "Example1" as the wafer's name, "50" mm as the wafer size, "0°" as the flat/notch location, "9.9" mm as the X and Y die sizes, and "0.1" mm as the X and Y street sizes.

4. Press the **OK** button. The Wafer Map Setup dialog box is removed and the Wafer Map Row and Column Entry dialog is displayed.

5. Enter "5" for the number of rows and columns of the wafer map and press the **OK** button. The dialog box is removed, and the Wafer Map Define Wafer dialog box is displayed.

6. For row number 0 enter "1" for the number of columns for that row and "2" as the offset from the left side of the wafer. Press the **OK** button.

7. For row number 1 enter "3" for the number of columns for that row and "1" as the offset from the left side of the wafer. Press the **OK** button.

8. For row number 2 enter "5" for the number of columns for that row and "0" as the offset from the left side of the wafer. Press the **OK** button.

9. For row number 3 enter "3" for the number of columns for that row and "1" as the offset from the left side of the wafer. Press the **OK** button.

10. For row number 4 enter "1" for the number of columns for that row and "2" as the offset from the left side of the wafer. Press the **OK** button.

11. The Wafer Map Define Wafer dialog box is removed and the Wafer Map window is updated to display the new wafer map.

Example 2: Another method of entering a wafer map is to create a 5 X 5 square wafer map and use the mouse to delete the die that don't belong. Follow the steps below to create the same wafer map as in Example 1 using this method.

1. If the current scale is not in mm, change the scale to mm by selecting the **Setup | Scale** menu item.

2. Select the **File | New...** menu item. The Wafer Map Setup dialog box (Figure 4-10) is displayed.

3. Enter the "Example2" as the wafer's name, "50.0" mm as the wafer size, "0° " as the flat/notch location, "9.9" mm as the X and Y die sizes, and "0.1" mm as the X and Y street sizes.

4. Press the **OK** button. The Wafer Map Setup dialog box is removed and the Wafer Map Row and Column Entry dialog is displayed.

5. Enter "5" for the number of rows and columns of the wafer map and press the **OK** button. The dialog box is removed, and the Wafer Map Define Wafer dialog box is displayed.

6. For row number 0 enter "5" for the number of columns for that row and leave the offset blank. Press and hold down the **ENTER** key. Rows 0 through 4 will be automatically filled with 5 columns each. The Wafer Map Define Wafer dialog box will be removed and the 5 X 5 square wafer map will be displayed.

7. Hold down the CTRL key and use the left mouse button in the Zoom View window to delete the extra die that don't belong on the wafer map.

4.4 Wafer Map Setup Dialog Box

The Wafer Map Setup dialog box is used to enter the general information that defines a wafer map. It is displayed by selecting the **File | New...** menu item, the **Map | Setup...** menu item, or by pressing the setup button on the Wafer Map Toolbar. The information displayed within the Wafer Map Setup dialog box can be changed at any time. If any values are changed the wafer map displayed in the Wafer Map window will update according to the new information.

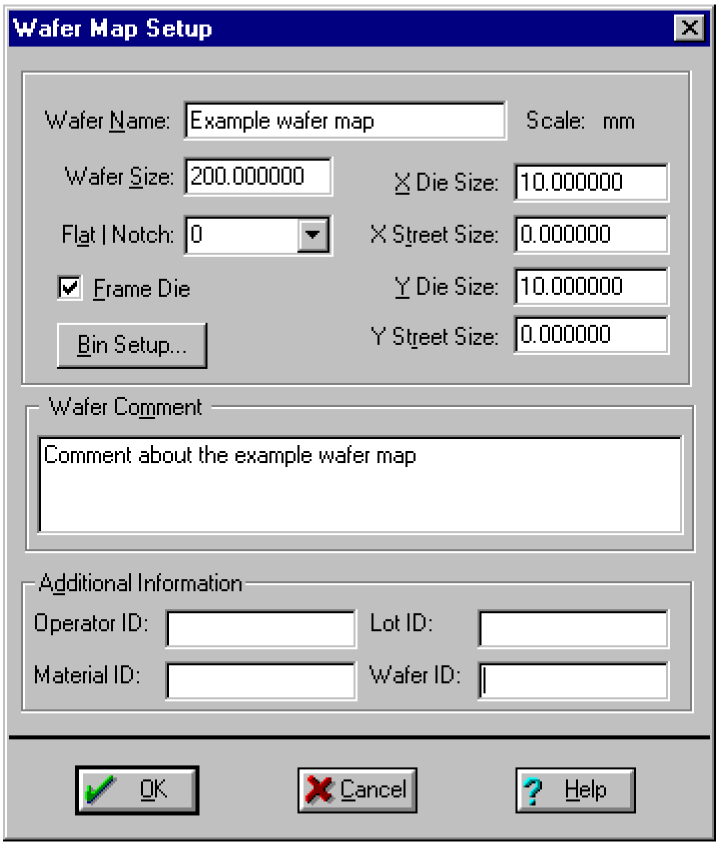


Figure 4-10 Wafer Map Setup Dialog Box

Select	To
Wafer Name:	Enter the name of the wafer map. The name is limited to 128 characters.
Wafer Size:	Enter the size of the wafer. The wafer size must be greater than 0 and less than or equal to 205 mm.
Flat location:	Enter the flat/notch location of the wafer map. 0° , 90°, 180°, and 270° are the possible flat/notch locations.
Frame Die	Draw a frame around each die on the wafer map in the Zoom View window.
X Die Size:	Enter the X-axis die size. The X-axis die size is limited to 50 mm.
X Street Size:	Enter the optional X-axis street size. This value will be added to the X die size to get the X-axis die-to-die spacing. The X-axis street size is limited to 50 mm.

Y Die Size:	Enter the Y-axis die size. This value will be added to the Y die size to get the Y-axis die-to-die spacing. The Y-axis die size is limited to 50 mm.
Y Street Size:	Enter the optional Y-axis street size. The Y-axis street size is limited to 50 mm.
Wafer Comment:	Enter an optional comment for the wafer. The comment text is limited to 256 characters.
Operator ID:	The Operator ID is intended to be automatically filled in by production software. It is limited to 127 characters in length.
Material ID:	The Material ID is intended to be automatically filled in by production software. It is limited to 127 characters in length.
Lot ID:	The Lot ID is intended to be automatically filled in by production software. It is limited to 127 characters in length.
Wafer ID:	The Wafer ID is intended to be automatically filled in by production software. It is limited to 127 characters in length.

4.5 Die Setup Dialog Box

The Die Setup Dialog Box is used to alter the setup information for the die on the wafer map. It is displayed by double clicking the left mouse button on any die in the Zoom View window. The coordinates of the die that was selected are displayed in the caption bar of the dialog box.

This dialog box is also used to select which die are to be tested on the wafer map by using the **Test All...**, **Test 25%...**, **Test 50%...**, **Test None...**, and **Test Bin #...** buttons. These buttons allow you to easily select which die you wish to test on the wafer map.

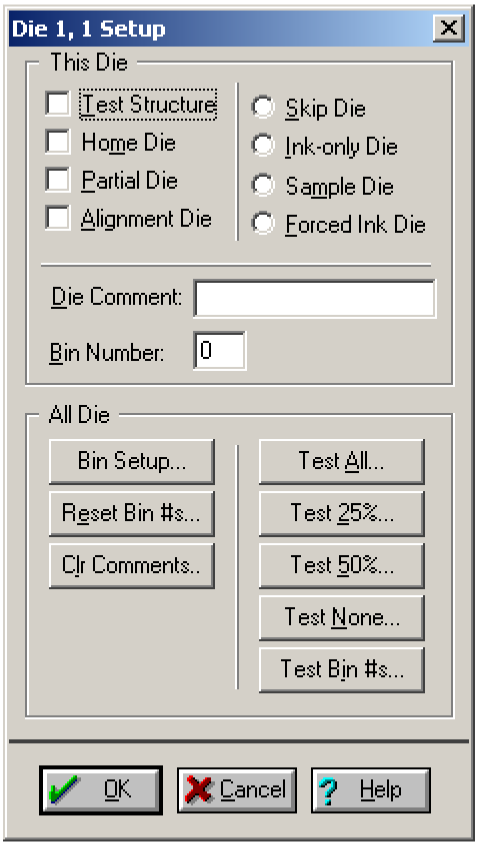


Figure 4-11 Die Setup Window

Select	To
Test Structure	Mark the die selected as a test structure. A test structure will be collared black on the wafer map.
Home Die	To select the current die as the home die. When the probe station's chuck is at the 0.000000, 0.000000 position it is at the home die location.
Partial Die	Mark the die selected as a partial die. A partial die will be collared black on the wafer map.
Alignment Die	Mark the die selected as an alignment die. A alignment die are marked with a '+' on the wafer map.
Skip Die	Mark the die selected as a die to be skipped when testing and inking. All skip die are marked with an 'X'. Die that are not skip die or ink-only die are considered "test" die. The Next Die and Previous Die toolbar buttons, and the NextDie( ) and PrevDie( ) REXX commands can be used to move from one test die to be another.

Ink-only Die Mark the die selected as a die to be skipped when testing and automatically inked when inking. All ink-only die are marked with a ‘/’. Die that are not skip die or ink-only die are considered "test" die. The Next Die and Previous Die toolbar buttons, and the NextDie( ) and PrevDie( ) REXX commands can be used to move from one test die to be another.

Sample Die Mark the die selected as a sample die. A alignment die are marked with a ‘S’ on the wafer map.

Forced Ink Die Forced Ink Die can only be created by double clicking each device and setting it as Forced Ink Die using the Map Setup dialog. The wafer map shows Forced Ink Die as ‘/’ which is also the indicator for Ink-Only Die. Forced Ink Die are inked automatically before the inking pass begins. Similar to Ink-only die Forced Ink Die are not tested.

Die Comment Enter a comment about the current die. The comment text is limited to 17 characters. For example, this can be used to store the filename of the raw test data for a die.

Bin Number: Enter the bin number for the selected die. A bin number can be in the range of 0 to 49, with a zero indicating a null bin number or no bin number assigned. The die will be collared the colour assigned to that bin number. The colours for each bin number can be changed by selecting the **Bin Colours...** button which displays the Colour Setup dialog box.

Bin Setup... Display the Bin Setup dialog box. This dialog box is used to assign colours to each bin number.

Reset Bin #s... Display a message box asking if all bin numbers on the wafer map should be set to 0. If **YES** is selected all bin numbers are reset.

Clr Comments.. Display a message box asking if all of the die comments are to be reset to null. If **YES** is selected all comments are cleared.

Test All.. Display a message box asking if all die are to be marked for testing. If **YES** is selected all the die on the wafer map will be marked for testing not including test structures, ink-only die, or partial die. All die on the wafer map are drawn without an "X" to indicate that they are die to be tested.

Test 25%... Display a message box asking if 25% of the die are to be marked as test die. If **YES** is selected 25% of the die on the wafer map are marked as die to be tested.

Test 50%... Display a message box asking if 50% of the die are to be marked as test die. If **YES** is selected 50% of the die on the wafer map are marked as die to be tested.

Test None... Display a message box asking if all die are to be marked as *not* to be tested. If **YES** is selected all die on the wafer map will be marked with an "X". This button is used to clear the wafer map of any die to be tested.

Test Bin #... Displays the Select Test Die dialog box. This dialog box allows you to select the bin number(s) of the die you wish to be tested.

4.6 Bin Setup Dialog Box

The Bin Setup dialog box is used to select the colours that are assigned to each bin number. It is displayed by selecting the **Bin Colours...** button in either the Die Setup dialog box or the Wafer Map Setup dialog box.

After selecting the bin number in the bin number list box, you can use the scroll bars to select the colour you wish to assign to the selected bin number. An optional comment can be entered for each bin number, which can be useful in describing whether the bin number indicates a pass or fail, etc.

When a new wafer map is initially created, bin number 0 is initialised to white, bin numbers 1 - 16 are initialised to the 16 primary Windows colours, and the rest are initialised to white.

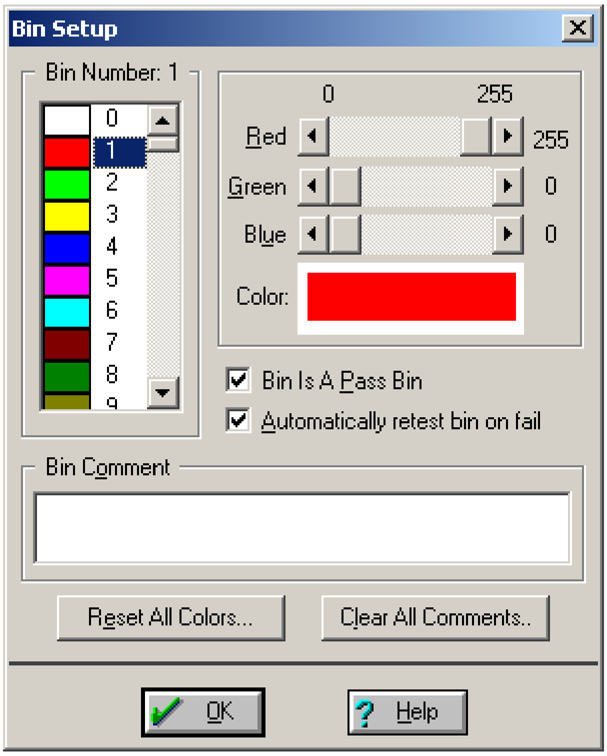


Figure 4-12 Colour Setup Dialog Box

Select	To
Reset Colours...	Display a message box asking if the first 16 bin colours are to be reset to the 16 primary Windows colours and the rest of the bin colours to white. Selecting **YES** will reset all bin colours.
Clr Comments..	Displays a message box asking if all bin comments should be cleared. Selecting **YES** will clear them.

4.7 Select Test Die Dialog Box

The Select Test Die dialog box is used to select which die are to be tested depending on their bin numbers. It is displayed by selecting the **Test Bin #...** button in the Die Setup dialog box. The dialog box contains a list box that displays the bin colours, bin numbers, and the user selected bin comments entered using the Colour Setup dialog box.

By using the standard Windows combinations of the left mouse button and the CTRL and SHIFT keys, you can highlight the bin numbers in the list box you wish to be marked as test die. All other die on the wafer map will be marked (with an X) as skip die. This dialog box can be very useful if a test needs to be repeated for single or several bin numbers.

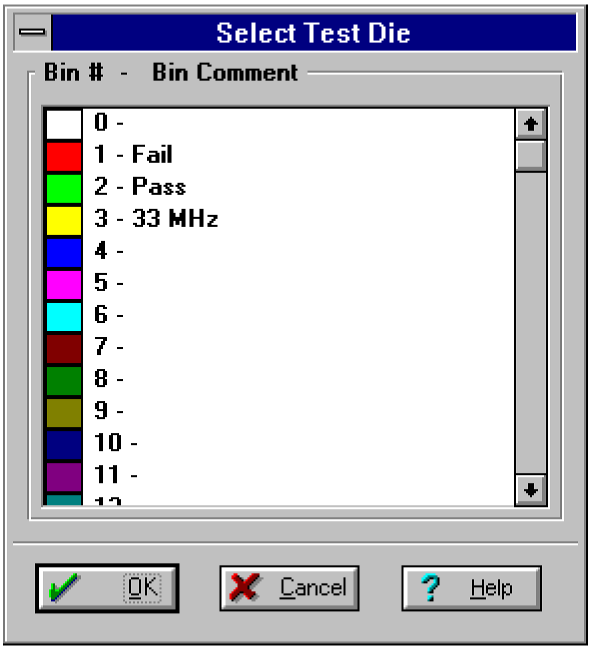


Figure 4-13 Select Test Die Dialog Box

You can use a mouse to select a bin number or group of bin numbers using the following methods...

To select a single bin number:

Click on the on the bin number.

To select two or more items in sequence

1. Click on the bin number you want to select.
2. Press and hold down the SHIFT key while you click on the last bin number in the group.

To select two or more items out of sequence

Press and hold down CTRL while you click on each bin number.

To cancel a selection

Press and hold down CTRL while you click on the selected bin number.

4.8 Setting the Home Die on the Wafer Map

This section describes how to align the home die on the wafer map to the home die on the actual wafer. The procedure below assumes that a wafer map has already been created and is currently loaded.

To Set the Home Die on the Wafer Map:

1. Double click with the left mouse button on the die you want to be the home die. This will bring up the Die Setup dialog box for that die.

2. Check the **Home Die** check box and press the **OK** button. This die is now the 0,0 die which is the 0.0, 0.0 X-Y home position of the prober.

3. Move the chuck to the home die on the actual wafer by using the arrow buttons in the Motion Control window.

4. If you are using a probe card, align the probe card to this die.

5. With the right mouse button, click on the prober button in the toolbar to bring up the devices' Pop-up dialog box.

6. Select the **Set Home...** button, enter 0.0, 0.0 as the X-Y user home position of the chuck, and press the **OK** button.

7. The die position shown on the wafer map will now match the die position of the probe card over the actual die on the wafer.

5 REXX Programming

5.1 Introduction

REXX is widely regarded for its ease of learning and use, and its user extendibility. Having been originally conceived as a user command language by Mike Cowlishaw of IBM, REXX is an ideal "macro and scripting" language for applications. LabMaster has extended the REXX language to include functions that control device movements, device settings, and GPIB communication. IBM has standardized REXX as the Systems Application Architecture (SAA) compliant command language for its mainframe, mid-range and personal computer systems. REXX has recently been accepted by ANSI for standardization.

REXX is a procedural language that allows programs and algorithms to be written in a clear and structured way. REXX was designed for ease of use by computer professionals and by "casual" general users. To be easy to use, a language must easily manipulate the object types you commonly deal with: names, addresses, text messages, words, numbers, etc. REXX features make manipulation of these object types easy. As a command language REXX is designed to be independent of the environment it operates within (e.g. an application or the operating system), and to have commands of that environment embedded within REXX programs.

LabMaster is able to provide the REXX programming ability because it interfaces with Personal REXX 3.0 for Windows, also referred to as WinREXX, developed by Quercus Systems. The LabMaster REXX interface is provided by Quercus Systems through Dynamic Link Libraries (DLLs) that are included on the LabMaster installation disks. LabMaster and Personal REXX 3.0 implement version 4.0 of the REXX language. If you would like to develop and run REXX programs outside of LabMaster, you are encouraged to install the WinREXX disks provided by Quercus Systems that were included in your purchase of LabMaster.

For learning REXX we suggest *Programming In REXX* by Charles Daney published by McGraw-Hill, ISBN 0-07-015305-1. For a full description of the REXX language refer to *The REXX Language - A Practical Approach to Programming* by M.F. Cowlishaw. Another excellent source is the *IBM Operating System/2 Procedures Language 2/REXX User's Guide* (S10G-6269 and S10G-6268). Also the documentation included with the Personal REXX for Windows (WinREXX) software is an excellent source for examples and learning REXX.

5.2 Creating, Editing, and Running REXX Programs

REXX programs can be created and edited in LabMaster by selecting the **File | Program | New Window** menu item. This will bring up a LabMaster Text Editor Multiple Document Interface (MDI) window, which is a fully functional text editor with file opening, saving, and printing capability. Multiple Edit windows can be open at any one time, allowing you to cut and paste text from one window to another. When an Edit window has input focus, i.e. its caption bar is highlighted; the menu at the top of LabMaster will change to depict the menu items available for the Edit window. Figure 5.1 shows the LabMaster Edit window and the associated menu items.

At the bottom of the Edit window is a status bar which displays information about the REXX program or text shown in the window. The status bar consists of an area for displaying messages, the current line number and column position of the insertion point, as well as the status of the CAPS LOCK and NUM LOCK keys. When the **Program | Run...** menu item is selected, the program loaded in the active Edit window will begin executing, and "Running..." will be displayed in that Edit window's status bar's message area.

At the top of the edit window is the toolbar which contains buttons for frequently used functions. The toolbar can be turned on and off by selecting the **Program | Toolbar** menu item.

NOTE	If you click the right mouse button on a REXX command or function, and then select help from the pop-up menu, help will be displayed for that command or function.

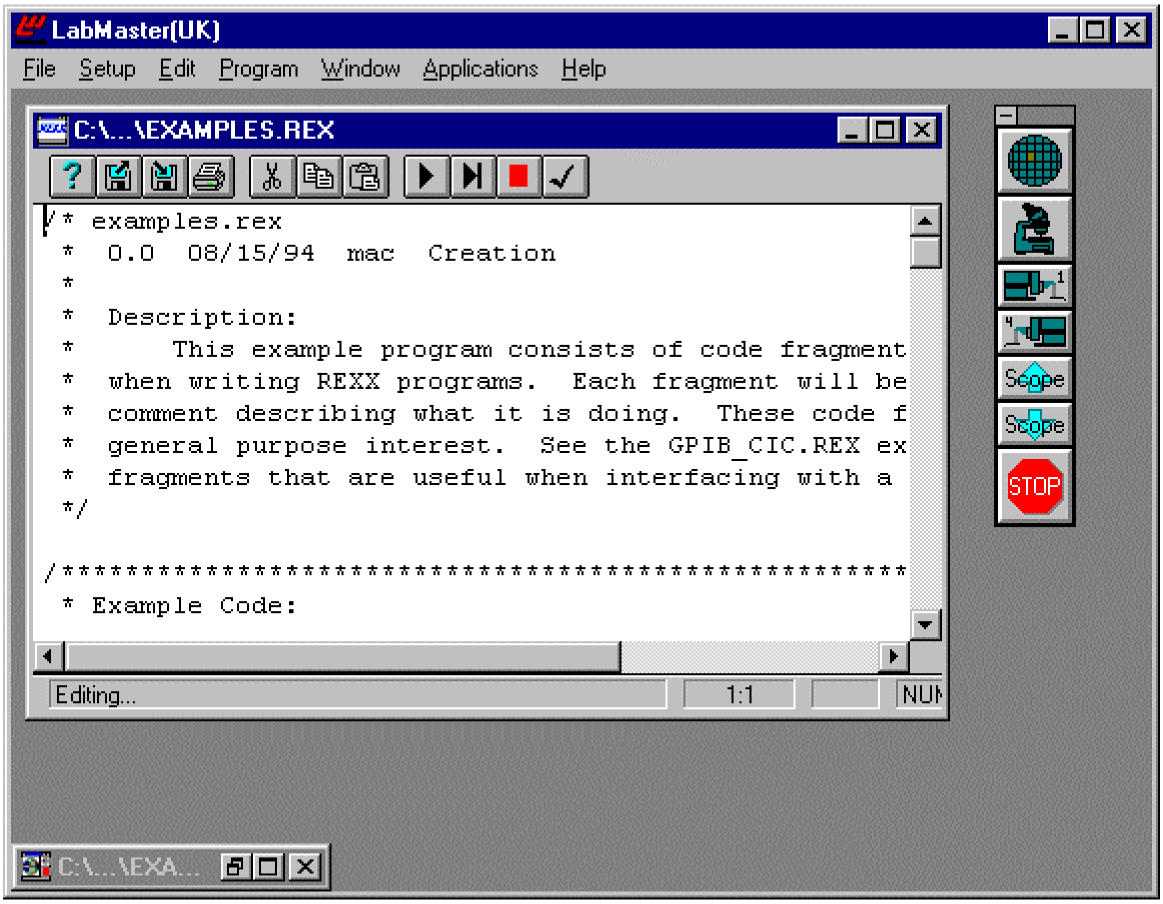


Figure 5-1 LabMaster Edit Text Window

5.2.1 Edit Window Toolbar

The Edit window toolbar contains buttons for frequently used functions.

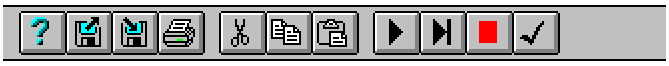
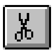
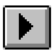
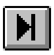
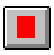


Figure 5-2 Edit Window Toolbar

Select		To
	Help	Display the Edit window help contents.
	Open	Display the File Open dialog box, allowing the selection of a text file to be loaded.
	Save	Saves the currently loaded text file.
	Print	Print the contents of the Edit window to the active printer.
	Cut	Cut the selected text to the Clipboard.
	Copy	Copy the selected text to the Clipboard.
	Paste	Paste the text located in the Clipboard at the current insertion point.
	Run	Run the REXX program loaded into the edit window.
	Trace	Trace through the REXX program loaded into the edit window.
	Stop	Stop the currently running REXX program.
	Syntax Check	Perform a syntax check of the REXX program loaded in the edit window.

5.3 Learn Dialog Box

The Learn Mode dialog box is used to record LabMaster movements and device settings into a LabMaster Edit Text Window. To enter Learn mode select the **Program | Learn...** menu item which will activate the Learn Mode dialog box.

The LabMaster Learn Mode was designed to provide you with a way of recording device positions and setup parameters into an Edit window for later execution. For example, if you needed to remember a sequence of four locations for a CAP within a particular die, you would perform the following steps:

1. Enter Learn Mode by selecting the **Program | Learn...** menu item.

2. Move the CAP to the first location, select the Position Saves check box for the CAP of interest in the Learn Program Mode dialog box, and then Press the **Probes & Microscope** button. The CAP position will be recorded into the Edit window at the current insertion point.

3. Repeat Step 2 to record the three other CAP positions.

4. Press the **OK** button to exit Learn Program Mode.

5. Now you can run the commands just recorded into the Edit window, by selecting the **Program | Run...** menu item. The CAP will then move to the previously recorded positions.

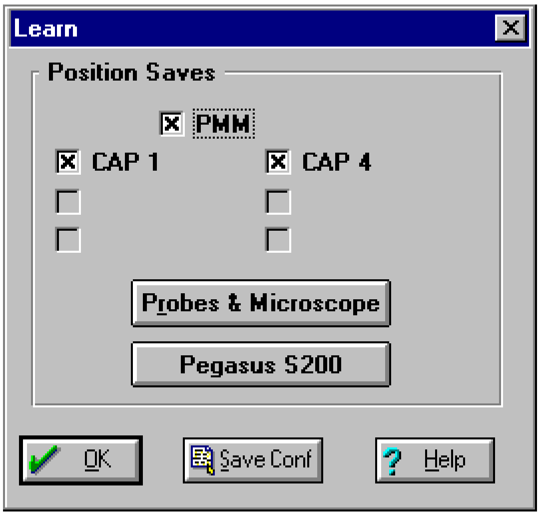


Figure 5-3 Learn Mode Dialog Box

The figure above shows what the Learn Program Mode dialog box would look like for a LabMaster system configured for a Pegasus S200, a PMM, and two CAPs.

NOTE	If you make changes to a device's setup parameters or change the scale while in learn mode, it will be automatically recorded into the program.

5.3.1 Learn Mode Example

This section describes the different buttons and controls in the Learn Program Mode dialog box, and gives an example of what is recorded into the program for each one. It describes what functions are recorded into a program for a LabMaster system containing a Pegasus prober, PMM, and a single CAP. A full description of each command can be found in Section 5.6 "LabMaster REXX functions"

When Learn Program mode is activated it automatically saves the current scale (millimetres or inches) as the first entry in the program. For example, after Learn Program mode is entered by selecting the **Program | Learn...** menu item and the Learn Mode dialog box is displayed, the program will contain the following function:

```
Set( "SCALE", "MM" )
```

The SET function is used to set the various internal LabMaster parameters such as the current scale, device velocities, etc.

By selecting the appropriate Position Saves checkboxes for the devices you wish to record the positions of, and then selecting the **Probes & Microscope** button, a Move command will be entered into the program for the selected PMM and CAP(s). The probe station's chuck position is recorded separately by selecting the **Pegasus** Position Saves button. If the **Probes & Microscope** button is pressed with the Microscope, and CAP 1 checkboxes checked, followed by the **Prober** button, the following functions will be entered into the program:

```
Move( "PMM", 1.000000, -5.000000, 0.000000 )
Move( "CAP1", 1.000000, -5.000000, 0.000000 )
Move( "PS", -10.000000, -10.000000 )
```

When the program is run and the functions are executed, the Microscope (PMM), CAP 1, and Pegasus prober (PS) will move to the positions specified in the MOVE functions.

The **Save Config.** button is used to record each device's setup parameters into the program. This will ensure that the same device parameters are used when the program is run at a later time. It is also useful for recording the device settings for each person that will be using LabMaster. By running the program containing these settings, each user can restore the device settings such as device speeds, step sizes, etc. to the way they like them. The **Save Config**. button causes the current LabMaster device settings to be entered in the program as show below:

```
Set( "SCALE", "MM" )
Set( "DARK_MODE", "OFF" )
Set( "PMM", "XY_SLOW", "0.00100" )
Set( "PMM", "XY_MEDIUM", "0.05000" )
Set( "PMM", "XY_FAST", "1.00000" )
Set( "PMM", "Z_SLOW", "0.00100" )
Set( "PMM", "Z_MEDIUM", "0.01000" )
Set( "PMM", "Z_FAST", "0.10000" )
Set( "PS", "VACUUM", "ON" )
Set( "PS", "Z_OVERDRIVE", "0.0000" )
Set( "PS", "Z_DROP", "0.0000" )
Set( "PS", "X_STEP_SMALL", "0.0010" )
Set( "PS", "X_STEP_MED", "0.0100" )
Set( "PS", "X_STEP_LARGE", "0.1000" )
Set( "PS", "Y_STEP_SMALL, "00010" )
Set( "PS", "Y_STEP_MED", "0.0100" )
Set( "PS", "Y_STEP_LARGE", "0.1000" )
Set( "PS", "Z_STEP_SMALL", "0.0010" )
Set( "PS", "Z_STEP_MED", "0.0100" )
Set( "PS", "Z_STEP_LARGE", "0.1000" )
Set( "PS", "Z_SLOW", "0.00100" )
Set( "PS", "Z_MEDIUM", "0.01000" )
Set( "PS", "Z_FAST", "0.10000" )
Set( "CAP1", "CRUISE", "OFF" )
```

```
Set( "CAP1", "CRUISE_HEIGHT", "0.005000" )
Set( "CAP1", "PRETOUCH", "OFF" )
Set( "CAP1", "PRETOUCH_HEIGHT", "0.005000" )
Set( "CAP1", "XY_SLOW", "0.00100" )
Set( "CAP1", "XY_MEDIUM", "0.05000" )
Set( "CAP1", "XY_FAST", "1.00000" )
Set( "CAP1", "Z_SLOW", "0.00100" )
Set( "CAP1", "Z_MEDIUM", "0.01000" )
Set( "CAP1", "Z_FAST", "0.10000" )
```

The **Help** button displays the LabMaster help file information on the Learn Mode dialog box.

5.4 Using REXX

This section gives a very brief introduction to the REXX language, it is not intended as a complete description of the language. For a complete description see one of the sources listed in Section 5.1, and LabMaster's extensive on-line help for a description of the REXX functions and commands.

5.4.1 Introduction to REXX

A REXX program is made up of a series of clauses. Clauses are made up of a sequence of blanks and tokens. You can place multiple clauses on one line by separating them with semicolons. You can continue very long clauses on the next line by using a comma (','). REXX is not case sensitive; therefore commands, variables, etc. can be either upper or lower case.

REXX does not make the distinction between numbers and strings. It essentially treats all variables as strings and conversions between data types are invisible to the user.

The best way to learn REXX is to study the REXX programs in the PROGRAMS subdirectory or the sample programs that come on the Personal REXX for Windows (WinREXX) disks. These programs demonstrate many of the REXX commands. Also the sources listed in Section 5.1 are an excellent source for learning REXX.

5.4.2 REXX Example 1

The following is an example of a simple REXX program.

```
/*     Example 1    */

say "Enter your age:"
pull age
say "Enter your first and Last name:"
parse pull first last
months = age * 12
say "Your name is" first last
say "You are" months "months old."
```

The first line of the example program shows how comments must be formatted in a REXX program. They must begin with "/*" and end with "*/".

The second line demonstrates the REXX SAY command. This command writes the specified text to the REXX I/O window (Figure 5-4). The I/O window is created the first time a REXX program uses a command that requires I/O or when there is a REXX error message to be displayed.

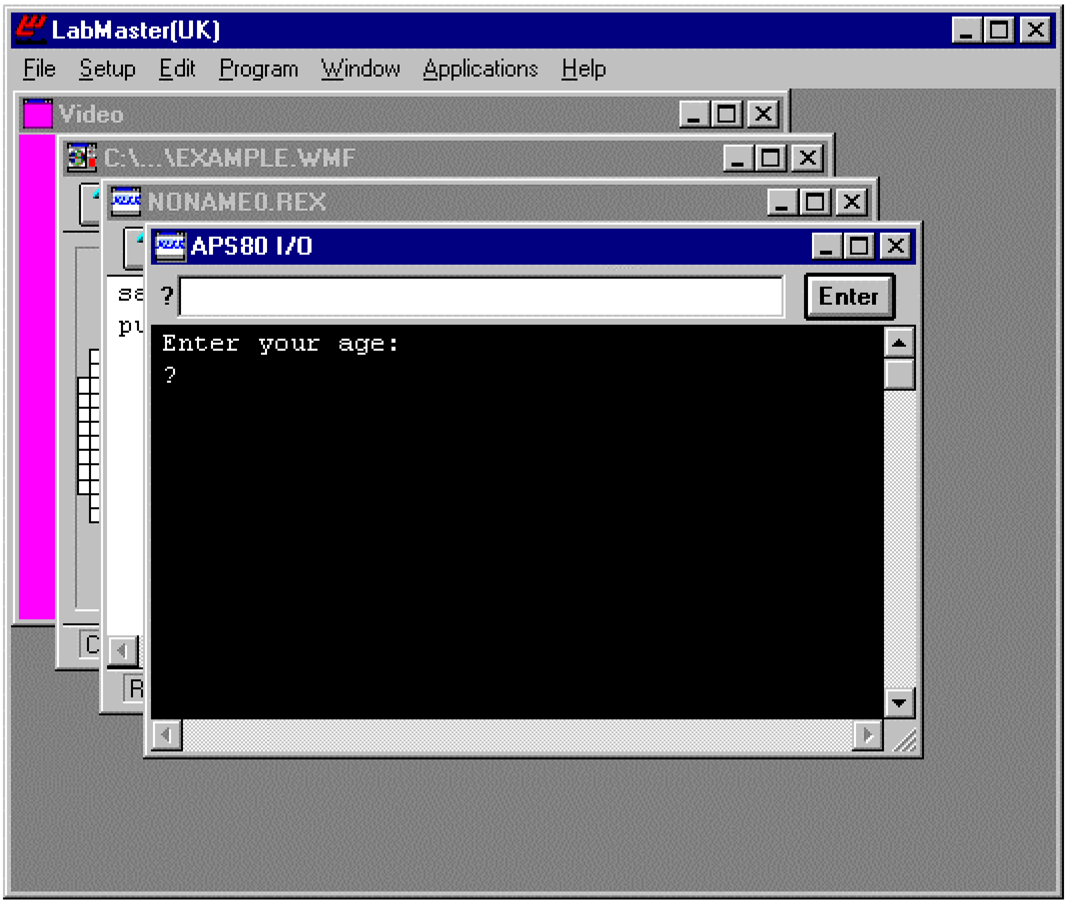


Figure 5-4 REXX I/O Window

The third line demonstrates the PULL command. This command causes the input bar to be displayed at the top of the I/O window, which allows the user to enter text. The text is then stored in the variable named after the PULL command.

The fifth line demonstrates a very simple use of the PARSE instruction. The PARSE instruction is used for assigning data to variables, and is one of the most powerful and unique aspects of the REXX language. The PARSE PULL instruction waits for the user to enter a response, and then assigns it to the variables "first" and "last", preserving the case of the letters. In this case, PARSE PULL uses blanks to determine how to split the user's entry. The logic for splitting apart the user's entry into two separate words and then assigning them to variables, which would require several steps in many other languages, is handled in one step by REXX's PARSE.

The sixth line calculates the number of months from a person’s age.

The last lines demonstrate that the SAY command can contain multiple items.

5.4.3 REXX Example 2 - Wafer Map Test

The following program demonstrates how to use the LabMaster REXX functions to move the chuck to each test die on a wafer map. The first thing the program does is to set the current scale to millimetres. The second function loads a wafer map called "example.wmf" located in the MAP subdirectory. The third function asks for the number of die that are marked to be tested on the wafer map and stores this number in the variable *iNumTestDie*. The fourth function moves the chuck to the first die to be tested. The last set of lines contain a loop that moves the chuck to each test die using the NextDie( ) function and sets each die's bin number to random number (1-10) using the SetDieBinNum( ) and random( ) functions. See the LabMaster on-line help for a complete description of each function.

```
/*      Example 2, Wafer Map Test Program    */

SET( "SCALE", "MM" )
LoadMap( GetRexxDirectory() || "\map\example.wmf" )

iNumTestDie = GetNumTestDie( )

FirstDie()

do iNumTestDie - 1
    ChuckUp( )
    SetDieBinNum( random(1,10) )
    NextDie( )
end
```

NOTE With the addition of some REXX GPIB function calls to test equipment to start a test and to get the test result, this simple program can be easily modified to test an entire wafer.

A fancier version of the above program called **RANDOM.REX** is distributed with LabMaster. It is located in the PROGRAMS subdirectory.

5.4.4 REXX Example 3 - Setting Bin Numbers

The following program **SET_BIN.REX** is distributed with LabMaster. It shows how to pull row & column bin number information out of an ASCII file and load it into a Wafer Map. It is located in the PROGRAMS subdirectory.

```
/*  set_bin.rex
 *  0.0  08/15/94  mac  Creation.
 *
 *  Description:
 *      Example REXX program showing how to pull row & column bin number
 *  information out of an ASCII file and load it into a Wafer Map.  The input
 *  file will be a space or tab delimited ASCII file.  The user will be
 *  prompted for the name of the input file.  The user will be prompted to
 *  load the wafer map file into which the bin number data should be placed.
 *      This example assumes that the production prober that originally
 *  collected the bin number data uses the same Home die as is defined for
 *  the LabMaster wafer map and that the production prober numbers the rows
 *  and columns in the same way.  If not, then the row and column numbers read
 *  in from the file would need to be remapped to their LabMaster equivalents.
 */

/*trace ?i*/

  /* Get name of this program for use in I/O with user */
  parse source Environment ExecType ProgName
```

```
/* Loop until we get a valid Input File Name from user */
ValidFileName = 0  /*false*/
do until ValidFileName
    msg = "Enter name of Production Prober Bin Number file:"
    InputFile = PromptBox( msg, ProgName, "" )
    if InputFile = "" then do
    /* User selected Cancel, end program */
    Exit
    end
    /* Check to make sure file exists (i.e. it has >0 lines) */
    if ( LINES( InputFile ) \= 0 ) then do
    ValidFileName = 1
    end
    else do
    msg = "Could not find file:" InputFile
        MessageBox( msg, ProgName )
        end
end /*looping for valid input file*/

/* Get User to load the cooresponding LabMaster Wafer Map */
msg = "Load Wafer Map file for Wafer:" InputFile
msg2 = "Hit Enter after its been loaded."
say msg
say msg2
pull ans   /* will wait for Enter to be hit */
/* Note that the say & pull instructions use the I/O window.  When the I/O
 * window is open, the user can still interact with the rest of LabMaster.
 * If we had used a PromptBox() or MessageBox() function, then the user
 * would have not been able to interact with the rest of LabMaster.
 */

/* Loop through all lines in the input file.  */
do while LINES( InputFile ) \= 0
    line = LINEIN( InputFile )   /* read in 1 line */
    /* Split line into row, column, and bin_number.  Anything after the bin
     * number we aren't interested so we'll ignore it. */
    parse var line row column bin_num any_extra_stuff
    /* Restrict bin numbers to the 0-49 range that LabMaster accepts */
    if ( bin_num > 49 ) then do
        bin_num = 49
        end
    /* Actually set the bin number in the wafer map */
    error_code = SetDieBinNum( row, column, bin_num )
    if ( error_code \= 0 ) then do
    msg = "Error:" error_code "returned from SetDieBinNum()"
        MessageBox( msg, ProgName )
        end
end /*of looping thru input file*/
```

5.4.5 REXX Example 4 - Useful code fragments

The following program **EXAMPLES.REX** is distributed with LabMaster. It contains examples of code fragments that can be useful when writing programs in REXX. It is located in the PROGRAMS subdirectory.

```
/* examples.rex
 *  0.0  08/15/94  mac  Creation
 *
 *  Description:
 *      This example program consists of code fragments that are useful
 *  when writing REXX programs.  Each fragment will be introduced with a
 *  comment describing what it is doing.  These code fragments are all of
 *  general purpose interest.  See the GPIB_CIC.REX example for code
 *  fragments that are useful when interfacing with a device over the GPIB
 *  bus.
 */

/***********************************************************************
 * Example Code:
 *     This code fragment shows how to prompt the user for the name of a
 * file that already exists.
```

```
**********************************************************************/
 /* Loop until we get a valid Input File Name from user */
 ValidFileName = 0  /*false*/
 do until ValidFileName
     msg = "Enter name of Production Prober Bin Number file:"
     InputFile = PromptBox( msg, ProgName, "" )
     if InputFile = "" then do
     /* User selected Cancel, end program */
     Exit
     end
     /* Check to make sure file exists (i.e. it has >0 lines) */
     if ( LINES( InputFile ) \= 0 ) then do
     ValidFileName = 1
     end
     else do
     msg = "Could not find file:" InputFile
         MessageBox( msg, ProgName )
         end
 end /*looping for valid input file*/


/**********************************************************************
 * Example Code:
 *     This code fragment shows how to prompt the user for the name of a
 * file that you want to use for output.  If a file with the user input name
 * already exists, then the user will be prompted about deleting the file.
 **********************************************************************/
 /* Loop to Get Output File Name from user */
 ValidFileName = 0  /*false*/
 do until ValidFileName
     msg = "Enter desired name for Output file:"
     OutputFile = PromptBox( msg, ProgName, "output.txt")
     /* If file already exists, ask user to confirm that we should erase the
      * old version. If we did not erase the old file the new information
      * would just be added to the end of the existing file. */
     if ( LINES( OutputFile ) \= 0 ) then do
     msg = "Overwrite existing file:" OutputFile "?"
     Answer = QuestionBox( msg, ProgName )
     if ( Answer == "Yes" ) then do
         /* Erase file */
         DOSDEL( OutputFile )
         ValidFileName = 1
         end
     end
     else do
     ValidFileName = 1
     end
 end /*looping for valid file name*/


/**********************************************************************
 * Example Code:
 *     Appending Carriage Returns and Line Feeds to a string.  REXX does
 * not use the 'C' language style "\r" and "\n" special notation.  Instead
 * these special characters must be entered as hexadecimal values.  When you
 * need to concatentate strings without REXX automatically putting a space
 * between the strings, the "||" operator must be used.
 **********************************************************************/
 CR = '0D'x    /* Carriage Return is hex 0x0D or decimal 13 */
 LF = '0A'x    /* Line Feed is hex 0x0A or decimal 10 */
 command = "FireLaser"
 command_with_terminators = command || CR || LF


/**********************************************************************
 * Example Code:
 *     This fragment shows how to get position information about a device,
 * format the numeric values, and then write the information out to a file
 * on the hard disk.
 **********************************************************************/
   DefectLogFile = "defect.log"
   /* Get the position of the Programmable Microscope Mount from LabMaster */
   Device = "PMM"
   x_mm = GetXPosition( Device )
   y_mm = GetYPosition( Device )
```

```
   /* Use the FORMAT function to format the numbers so that the values are
    * only displayed with .1 micron resolution. */
   x_microns = FORMAT( x_mm * 1000, , 1, , )
   y_microns = FORMAT( y_mm * 1000, , 1, , )
   /* Create the line to be added to the log file.  The DATE and TIME
    * functions are used to provide the current date and time. In REXX to
    * continue an expression onto a second line, a comma "," is used at the
    * end of the line to be continued as shown below. */
   line = "Defect found at:" x_microns "," y_microns "microns on:" ,
          DATE() TIME( C )
   LINEOUT( DefectLogFile, line )
   /* Following is an example of what a line added to the "defect.log" file
    * looks like:
    *   Defect found at: 15.9 , 22.4 microns on: 18 Aug 1994 8:39am
    */


/*********************************************************************
 * Example Code:
 *     When working with lots of different files, you can sometimes run into
 * the DOS limit on the number of files that can be open at once.  Within
 * REXX, any files that are automatically openned will be closed when the REXX
 * program finishes.  When it is necessary to explicitly close a file
 * within a REXX program, the LINEOUT function can be used as shown below.
 *********************************************************************/
   FileName = "output.txt"
   LINEOUT( FileName, "When you write out to the file it remains open" )
   LINEOUT( FileName )  /* a LINEOUT call with just the filename as an
                         * argument will close the file */

/*********************************************************************
 * Example Code:
 *     This fragment will demonstrate some ways to parse input data (parsing
 * data breaks it down into its components.  This input data could be
 * test results returned over the GPIB from a piece of equipment or it
 * could be data input by the user, or it could be data read in from a
 * file on the hard disk, etc.  For more information on parsing,
 * see the various REXX references refered to in the LabMaster manual.
 * Another good source is section 5.12 "More about PARSE" in the Personal REXX
 * User's Guide that is part of the Personal REXX package that is distributed
 * with LabMaster.
 *********************************************************************/
   /* In the examples that follow, the input data will be row number,
    * column number, bin number, and a comment that are delimited
    * (i.e. formated) in different ways. */
   /* Space delimited input */
   input = "3 5 13 Failed at 33MHz"
   parse var input row col bin_num comment
   say "r=" row "c=" col "bin=" bin_num "Comment is:" comment
   /* Comma delimited input */
   input = "4,6,11,Passed at 33MHz"
   parse value input with row "," col "," bin_num "," comment
   say "r=" row "c=" col "bin=" bin_num "Comment is:" comment
   /* Tab delimited input (a tab is '09'x) */
   input = "4     7      10     Floating point failed"
   parse value input with row '09'x col '09'x bin_num '09'x comment
   say "r=" row "c=" col "bin=" bin_num "Comment is:" comment


/*********************************************************************
 * Example Code:
 *     This fragment will demonstrate how a function might be used so
 * that a block of code that made a measurement would not to be repeated
 * in multiple places in the code.  This fragment might be part of a program
 * that uses a CAP with Contact Sense to probe various locations within a
 * die.  At each location at GPIB controllable DVM would be used to make
 * a voltage measurment.  The program keeps track of the maximum voltage
 * it measures and the location at which it was measured.
 *********************************************************************/
   /* code at beginning of program would have general gpib overhead as
    * appears in gpib_cic.rex example.  Variable gpib_ud would be assigned
    * the result of the ibfind() call.  Code at beginning would also take
    * care of setup issues such as getting user to move CAP and PMM to
    * a reference position and set that location their user homes, setting
    * the CAP Cruise height and enabling its use, etc
```

```
     */
    CAP = "CAP5"
    /* init variables used to keep track of our max voltage & its location */
    max_voltage = -1.
    max_volt_x  = -1.
    max_volt_y  = -1.
    /* Move to each location we need to test */
    x = 0.050
    y = 0.050
    Move( CAP, x, y, )
    Move( "PMM", x, y, )  /* have scope follow CAP so we can watch */
    MakeMeasurement( CAP, x, y )
    x = 0.050
    y = 0.150
    Move( CAP, x, y, )
    Move( "PMM", x, y, )
    MakeMeasurement( CAP, x, y )
    x = 0.050
    y = 0.200
    Move( CAP, x, y, )
    Move( "PMM", x, y, )
    MakeMeasurement( CAP, x, y )
    x = 0.150
    y = 0.200
    Move( CAP, x, y, )
    Move( "PMM", x, y, )
    MakeMeasurement( CAP, x, y )
    x = 0.250
    y = 0.200
    Move( CAP, x, y, )
    Move( "PMM", x, y, )
    MakeMeasurement( CAP, x, y )
    /* All done. Tell user about the maximum */
    say "Max Volts of:" max_voltage "detected at x,y=" max_volt_x max_volt_y
    exit  /* end of main program */

/*-----Function used by above program---------------------------------*/
/* MakeMeasurement: this function will move the CAP into contact with the
 *   wafer and get the DVM to measure the voltage.  It will update our
 *   global variables that keep track of the maximum voltage and its location.
 *       The 'expose' keyword is used so that the variables gpib_ud,
 *   max_voltage, max_volt_x, and max_volt_y are considered 'global' and
 *   code inside MakeMeasurement can use them and set them.
 * Inputs:
 *   CAPName - string that identifies which CAP-4000 we are working with
 *   x_location - our current x location
 *   y_location - our current y location
 * Outputs:  function returns the voltage value that was measured
 */
MakeMeasurement: procedure expose gpib_ud max_voltage max_volt_x max_volt_y
    CAPName  = arg(1)
    x_location = arg(2)
    y_location = arg(3)
    /* Move the CAP down in Z to make contact with the Bus line */
    Set( CAPName, "MODE", "CONTACT" )
    ContactSearch( CAPName )
    /* Tell the DVM to measure the voltage and read back the result */
    command = "MV"  /* measure voltage */
    ibwrt( gpib_ud, command, length(command) )
    voltage = ibrd( gpib_ud, 128 )
    /* Output the results to the I/O window */
    say "Voltage at x,y:" x_location y_location "is" voltage "volts"
    /* keep track of our maximum voltage */
    if ( voltage > max_voltage ) then do
    max_voltage = voltage
    max_volt_x  = x_location
    max_volt_y  = y_location
    end
return voltage   /* functions must return a value */
```

5.5 Tracing and Debugging

When you have written a program, it can be run step-by-step by selecting the **Program | Trace** menu item. The REXX I/O window will be displayed allowing you to interactively step through the program by repetitively pressing the **Enter** button on the keyboard or in the I/O window.

Programs can also be traced or debugged by using the TRACE command at the point in your program you wish to begin tracing. See one of the sources listed in Section 5.1 or the LabMaster on-line help for more information on the TRACE command.

5.6 LabMaster REXX Functions

Here is a list of the REXX functions that are unique to LabMaster. For comprehensive details on each of the functions listed here, see LabMaster's on-line help.

5.6.1 Generic Device Functions

These functions handle generic device procedures.

Function	Description
Get( )	Retrieves the value of various system parameters. The parameters that can be retrieved are the Z axis parameters associated with the PS device.
GetXCentrePosition( )	Retrieves the X-axis centre of rotation of the given device, which currently must be "PS" (Probe Station).
GetXPosition( )	Retrieves the X-axis position of the device specified.
GetYCentrePosition( )	Retrieves the Y-axis centre of rotation of the given device, which currently must be "PS" (Probe Station).
GetYPosition( )	Retrieves the Y-axis position of the device specified.
GetZPosition( )	Retrieves the Z-axis position of the device specified.
Move( )	Allows one device to be moved to a new position specified in coordinates relative to the Home position.
MoveRel( )	Allows one device to be moved to a new position specified as a relative move from the current position.
SendString( )	Sends the given message string to the given device, which currently must be "PS" (Probe Station).
Set( )	Lets the user change the value of various system parameters.
SetXYCentrePosition( )	Saves the X and Y position of the centre of rotation of the probe station chuck, in Stage coordinates.

5.6.2 CAP Functions

These functions handle the CAP (including contact sense) specific procedures.

Function	Description
AutoOffset( )	Automatically adjusts the offset voltage for a Contact Sense board associated with a particular CAP.
AutoThreshold( )	Automatically adjusts the threshold voltage for a Contact Sense board associated with a particular CAP.
ContactSearch( )	Starts the specified CAP moving down in the Z direction, looking for device contact.
MoveCruise( )	Allows the specified CAP to be moved to its cruise height.
MovePretouch( )	Allows the specified CAP to be moved to its pre-touch height.

5.6.3 Laser Functions

These functions handle laser specific procedures.

Function	Description
FireLaser( )	Fires the laser.

5.6.4 Microscope Functions

These functions handle microscope specific procedures.

Function	Description
GetAZoom( )	Get an AZoom microscope parameter.
ScopeDown( )	Moves the microscope lift to its bottom limit.
ScopeUp( )	Moves the microscope lift to its upper limit.

5.6.5 Platform Functions

These functions handle platform specific procedures.

Function	Description
PlatformDown( )	Moves the platform to its bottom limit.
PlatformReference( )	References the platform.
PlatformUp( )	Moves the platform to its upper limit.

5.6.6 Prober Functions

These functions handle prober specific procedures.

Function	Description
CheckCoordsValid( )	Checks that a move is valid for circular probing zone on the prober.
ChuckDown( )	Moves the probe stations chuck to the “Fine Down” position.
ChuckUp( )	Moves the probe stations chuck to the “Fine Up” position.
Clamp( )	Clamp control on a DSP with a motorised clamp.
GetCameraCentre( )	Gets the X or Y camera centre position.
GetClampPosition( )	Gets the position of the clamp on a DSP with a motorised clamp.
GetIncrement( )	Gets the X or Y axis increment (die pitch) for circular probing zone on the prober.
GetLoadPins( )	Retrieves the current position of the chuck load pins.
GetProberInfo( )	Retrieves information about the prober set-up.
GetStatus( )	Retrieves the current prober status.
GetThetaPosition( )	Retrieves the theta position of the prober’s chuck.
GetThetaRelHome( )	Retrieves the theta position of the prober’s chuck, relative to the theta HOME position.
GetWaferCentre( )	Gets the X or Y axis wafer centre position of circular probing zone on the prober.
GetXHomePosition( )	Retrieves the X-axis HOME position of the prober.
GetXPositionStage( )	Retrieves the X axis position of the probe station’s chuck, in Stage coordinates.
GetYHomePosition( )	Retrieves the Y-axis HOME position of the prober.
GetYPositionStage( )	Retrieves the Y axis position of the probe station’s chuck, in Stage coordinates.
GrossChuckDown( )	Moves the probe station chuck to the “Gross Down” Z axis position.
GrossChuckUp( )	Moves the probe station chuck to the “Gross Up” Z axis position.
InitialiseProber( )	Initialises the prober for use in LabMaster system.
IsProberConnected( )	Determine if the prober is connected to the system.
LoadPinsDown( )	Moves the load pins on the prober’s chuck to the down position (i.e. load pins withdrawn).
LoadPinsEnable( )	Enables/Disables the load pins the prober’s chuck.
LoadPinsUp( )	Moves the load pins on the prober’s chuck to the up position (i.e. load pins extended).
LocalControl( )	Puts the prober in to local control, so that the operator can control it locally from the keyboard and display unit.
MoveArea( )	Move to a prober area (probing centre, camera centre, manual load/unload, or robot load/unload).
MoveCentre( )	Move the prober’s stage to the centre position. This is the centre of the circular probing zone, if the circular probing facility is used.
MoveHome( )	Moves the chuck to the Home position.
MoveStage( )	Moves the chuck to the given stage coordinates.

Function	Description
MoveTheta( )	Moves the chuck to the given theta position.
MoveThetaRelHome( )	Moves the chuck to the give theta position, relative to the theta HOME position.
MoveUnload( )	Allows the chuck to be moved to its load/unload position.
SetCameraCentre( )	Sets the camera centre position.
SetIncrement( )	Saves the X or Y axis increment (die pitch) for circular probing zone on the prober.
SetWaferCentre( )	Sets the probing centre position.
SetXYHomePosition( )	Saves the X and Y position of the Home Die, in Stage coordinates.
ThetaReference( )	Drives the prober Automatic Theta Chuck in the theta axis, to the chucks hardware switch reference position.

5.6.7 Prober I/O Functions

These functions handle prober input/output procedures.

Function	Description
ClearOutputLine( )	Clears an output line on the prober.
GetInputLine( )	Retrieves the state of a input line on the prober.
FastTTL( )	Perform TTL testing on a die, using the prober firmware. *This function should be used for test times of one minute or less (see also the TestTTL( ) function below).*
Ink( )	Fires the specified inker on the probe station.
SecondaryVacuum( )	Switches the secondary (pins) vacuum on the prober to the specified state.
SetOutputLine( )	Sets an output line on the prober.
SignalEOW( )	Sends the TTL end of wafer signal.
TestTTL( )	Perform TTL testing on a die. *This function should be used for test times of over one minute (see also the FastTTL( ) function above).*
Vacuum( )	Switches the primary (chuck) vacuum on the prober to the specified state.

5.6.8 DDE Functions

These functions handle Dynamic Data Exchange capabilities.

Function	Description
DDEExecute( )	Execute a command, function, or macro in the server application.
DDEGetLastError( )	Retrieve the last DDE error generated by the DDEInitiate( ), DDEExecute( ), DDEPoke( ), DDERequest( ), DDETerminate( ) and DDETerminateAll( ) functions.
DDEInitiate( )	Initiates a channel with the given DDE ***topic*** within the application specified in ***app_name***.
DDEPoke( )	Send information (***data***) from a client application to server application.
DDERequest( )	Request information from a server application.
DDETerminate( )	Closes the channel opened using the DDEInitiate( ) function.
DDETerminateAll( )	Closes all open DDE channels opened using the DDEInitiate( ) function.
GetDDETimeout( )	Retrieve the timeout value set for DDE operations.
SetDDETimeout( )	Allows the setting of the timeout period for DDE transactions.
StartRamp( )	Allows the DDE Client application to start one device moving in the specified direction at the specified speed.
StopRamp( )	Allows the DDE Client application to stop the motion that was started with the previous StartRamp() command.

5.6.9 Mathematical Functions

These functions handle mathematical calculations.

Function	Description
RxArcTan( )	Finds the angle in radians where the tangent is the supplied parameter.
RxArcTan2( )	Finds the angle in radians where x and y are the supplied parameters.
RxCos( )	Calculates the Cosine of the angle supplied in radians.
RxSin( )	Calculates the Sine of the angle supplied in radians.
RxSqrt( )	Calculates the square root of the number given.

5.6.10 Programming Functions

These are REXX programming support functions.

Function	Description
CreateSavedVariable( )	Creates a global variable, who's value is saved to file when LabMaster exits and then restored when LabMaster is rerun.
CreateVariable ( )	Creates a global variable, who's value is lost when LabMaster exits.
GetLastReturnValue( )	Gets the last value returned from a *LabMaster* REXX function.
GetScale( )	Retrieves the current system scale units (“**IN”** or “**MM**”).
GetSWVersion( )	Gets LabMaster's name and version number.
GetVariable( )	Gets the value of a global variable.
InitLocal( )	Initialises a variable with a value from an initialisation file.
LabmasterMode( )	Sets the LabMaster system in to the given mode. The ***mode*** can be either **PRODUCTION** or **ANALYTICAL**.
RxFuncAttachDLL( )	Required by some DLLs before RxFuncAdd can be used.
SetLastReturnValue( )	Sets the last value returned from a *LabMaster* REXX function.
SetVariable( )	Sets the value of a global variable.

5.6.11 RS232 Functions

These functions handle RS232 ports.

Function	Description
RS232CheckForMessageIn( )	Checks whether there is a message waiting on the given RS232 port.
RS232Close( )	Closes the given RS232 port.
RS232Open( )	Opens the given RS232 port, to run at the given baud rate.
RS232Read( )	Reads a message from the given RS232 port.
RS232Write( )	Sends the given message to the given RS232 port.

5.6.12 User Interface Functions

These functions handle the user interface.

Function	Description
ActionBox( )	Generates a modeless dialog box, with **OK** and/or **Cancel** buttons.
ActionBoxDestroy( )	Destroys the modeless dialog box generated with ActionBox( ).
ApplicationMenu( )	Enables/disables LabMaster's main menu.
ControlPanelClose( )	Closes the control panel dialog box opened with ControlPanelOpen( ).
ControlPanelOpen( )	Opens a control panel dialog box, which contains 10 menu buttons.
DeviceToolbar( )	Allows the user to display or hide the Device Toolbar or its buttons.
ExtraProberFunctions( )	Allow the REXX program to run extra functions on the prober, using a floating dialog box.
GetFileName( )	Allows selection of a file name using a File Open dialog.
Group( )	Allows the user to specify the devices that are active in the Toolbar.
JoystickButtons( )	Reads the current state of the four joystick buttons.
JoystickPosition( )	Reads the current position of the joystick.
MotionControl( )	Displays the motion control dialog box.
ReadPassword( )	Allows the supervisor to input a password, in order to enter a password protected function.
StatusBar( )	Allows the LabMaster four panel status bar to be shown or hidden.
StatusBarPanel( )	Writes text to a panel in the LabMaster status bar.
StatusWindow( )	Allows the LabMaster status window to be shown or hidden.

5.6.13 Video Functions

These functions handle the video window.

Function	Description
VideoSelectCamera( )	Selects the given camera for the video window.

5.6.14 Wafer Map Functions

These functions handle Wafer Map data and procedures.

Function	Description
FirstDie( )	Moves the chuck from the current position to the first test die location.
GenerateWaferMap( )	Allows a wafer map to be defined and generated, using the given ***filename***.
GetBinComment( )	Retrieves the comment text associated with the bin number specified.
GetDieBinNum( )	Retrieves the bin number of the die at the given die position.
GetDieComment( )	Retrieves the Comment Text of the die at the given die position.
GetDieFromCentre( )	Calculate the offset from the centre of the wafer to the given die.
GetDieHome( )	Retrieves the absolute die position of the home die.
GetFirstDiePosition( )	Retrieves the absolute die position of the first die.
GetFlatLocation( )	Retrieves flat or notch location of the wafer shown in the wafer map.
GetNumDie( )	Retrieves the number of die on the wafer shown in the wafer map.
GetNumTestDie( )	Retrieves the number of die marked as die to be tested on the wafer map.
GetWaferDetails( )	Retrieves wafer data to allow wafer to be automatically probed.
GetWaferName( )	Retrieves name of the wafer.
GetWaferParameter( )	Retrieves the specified parameter of the current wafer map.
GetXDiePosition( )	Retrieves the chuck's X-axis position in die coordinates on the wafer map.
GetXDieSize( )	Retrieves the wafer map's X-axis die size.
GetXStreetSize( )	Retrieves the wafer map's X-axis street size.
GetYDiePosition( )	Retrieves the chuck's Y-axis position in die coordinates on the wafer map.
GetYDieSize( )	Retrieves the wafer map's Y-axis die size.
GetYStreetSize( )	Retrieves the wafer map's Y-axis street size.
IsDie( )	Determines if a position on the wafer map represents a die.
IsDieFlagSet( )	Determines if a die has a particular attribute (ink-only die, partial die, skip die, or test structure).
IsPassBin( )	Determines if a bin is a pass bin.
LastDie( )	Moves the chuck from the current position to the last die to

Function	Description
	be tested on the wafer map.
LoadMap( )	Loads the wafer map specified.
MoveAbsDie( )	Moves the probe station chuck, to put the specified die to the required position.
MoveDie( )	Allows the chuck to be moved to a new position specified in die coordinates relative to the Home die.
MoveDieRel( )	Allows the chuck to be moved to a new position specified as a relative move from the current position in die coordinates.
NextDie( )	Moves the chuck from the current position to the next die to be tested on the wafer map.
PrevDie( )	Moves the chuck from the current position to the previous test die on the wafer map.
SaveMap( )	Saves the currently loaded wafer map to a file specified.
SetDieBinNum( )	Sets the bin number for any die on a loaded wafer map.
SetDieComment( )	Sets the die's comment text for any die on the wafer map.
SetDieFlag( )	Sets a die attribute (ink-only die, partial die, skip die, or test structure).
SetDieHome( )	Sets the home die position.
SetDieStepping ( )	Sets the x and y step size for when multi-die testing is being performed.
SetWaferParameter( )	Sets the specified parameter of the current wafer map.
TestingRightToLeft( )	Returns the current movement direction.

5.6.15 Machine Vision Functions

These functions handle the Machine Vision System, which is used to do Pattern Recognition. These functions are unique to LabMaster with PR.

Function	Description
FindAllPatterns( )	Used to find pattern matches for both templates, using the vision system.
FindBlobs( )	Used to find one or more blobs.
FindPattern( )	Attempts to find a pattern match for the template given.
GetPRCameraView( )	Retrieves the current camera’s field of view.
GetPRParameter( )	Reads a Vision system parameter.
GetPRXXCal( )	Retrieves the X-axis’ X calibration value for the Vision system.
GetPRXYCal( )	Retrieves the X-axis’ Y calibration value for the Vision system.
GetPRYXCal( )	Retrieves the Y-axis’ X calibration value for the Vision system.
GetPRYYCal( )	Retrieves the Y-axis’ Y calibration value for the Vision system.
LearnPattern( )	Enables the user to learn a pattern on the wafer, using the Vision system.
PRInitialise ( )	Starts up the Vision system.
ProbesBoxSet( )	Sets up to display a rectangle with the coordinates given, on the LabMaster video screen.
PRSelectCamera( )	Selects the given camera for the Vision system.
PRTerminate ( )	Shuts down the Vision system.
RestorePattern( )	Restores the pattern data given on the Vision system.
SetPRParameter( )	Write a Vision system parameter.
SetPRXXCal( )	Saves the X-axis’ X calibration value for the Vision system.
SetPRXYCal( )	Saves the X-axis’ Y calibration value for the Vision system.
SetPRYXCal( )	Saves the Y-axis’ X calibration value for the Vision system.
SetPRYYCal( )	Saves the Y-axis’ Y calibration value for the Vision system.
ShowProbesBox( )	Allows a rectangle to be drawn or erased, on the LabMaster video screen.
VisionBarcodeRead( )	Reads a ID from a barcode.
VisionGetArea( )	Used to prompt the user to enter a rectangular area.
VisionLiveDisplay( )	Switches the Live Video display to the specified state (ON or OFF).
VisionOCRRead( )	Reads a ID from an OCR.
VisionSnap( )	Used to switch on or off, a facility of the Vision system in LabMaster to automatically snap a camera image after the stage has been moved.

5.7 GPIB REXX Functions

The LabMaster GPIB REXX functions allow you to send and receive data and commands using a National Instruments (NI) GPIB interface card. The GPIB REXX functions are located in a Dynamic Link Library (DLL) called LMGPIB.DLL. This DLL is used to translate the REXX GPIB functions into NI GPIB function calls to the NI GPIB.DLL. This section assumes that you have experience in programming with the NI GPIB functions.

The REXX GPIB functions are almost identical to the NI GPIB functions, therefore the NI GPIB documentation can be used. The REXX GPIB functions all begin with "GPIB" to differentiate them from the NI GPIB functions. Any differences between the LabMaster implementation of the NI GPIB function and NI implementation, are shown below.

5.7.1 Using the GPIB Functions

To use the functions contained in the LMGPIB.DLL, they must first be added to the program using the RxFuncAdd( ) command:

RxFuncAdd(***ExternalFunctionName, DLLName, EntryPoint*** **)**

ExternalFunctionName	The function name as it will be used in the REXX program.
DLLName	DLL file name which contains the function.
EntryPoint	DLL Export Name of the 'C' function to call in DLLName.

Returns: RxFuncAdd returns 0 if it is successful.

For example to use the NI ibfind( ) function within a REXX program, you must include the following line at the beginning of your REXX program:

```
RxFuncAdd( "ibfind", "LMGPIB.DLL", "GPIBfind" )
```

See the sample programs located in the PROGRAMS subdirectory for more examples on how to use the REXX GPIB functions.

5.7.2 LabMaster GPIB Slave Mode - Sample Program

The following program, **GPIB.REX**, demonstrates the use of the REXX GPIB commands. This program is located in the PROGRAMS subdirectory and is used to put LabMaster in a GPIB remote mode so that it acts like a GPIB device. REXX commands can then be sent over GPIB and will be executed by this REXX program. After each command is sent by the controller to LabMaster, the response from this command must be read back before another command is sent.

This program also demonstrates some of the advanced REXX programming items such as exception handlers, and the PARSE and INTERPRET commands.

```
/****************************************************************/
/*    GPIB.REX       Version 1.1       04/28/93 jlg            */
/*                                                             */
/*    LabMaster REXX program to make LabMaster act as a GPIB   */
/*    device (i.e. not the system controller).                 */
/*                                                             */
/*    Make sure that you use the National Instruments          */
/*    GPIBconFig. program to disable GPIB0 as the system        */
/*    controller.                                              */
/*                                                             */
/*    For this program to function correctly, each time you    */
/*    send LabMaster a REXX command over GPIB, you must read   */
/*    back the result or you will not be able to send another  */
/*    command.                                                 */
/****************************************************************/

/*    Set up exception handlers                                */
/*    Needed so that we can gracefully handle bad commands     */
/*      received over GPIB, and someone hitting the STOP       */
/*      button on the Toolbar                                  */
call on halt
signal on syntax

/*    Add GPIB functions we need located in the LMGPIB.DLL     */
RxFuncAdd( "iblines", "LMGPIB.DLL", "GPIBlines" )
RxFuncAdd( "ibfind",  "LMGPIB.DLL", "GPIBfind" )
RxFuncAdd( "ibrd",    "LMGPIB.DLL", "GPIBrd" )
RxFuncAdd( "ibwrt",   "LMGPIB.DLL", "GPIBwrt" )
RxFuncAdd( "ibcmd",   "LMGPIB.DLL", "GPIBcmd" )
RxFuncAdd( "ibtmo",   "LMGPIB.DLL", "GPIBtmo" )

/*       GPIB constants for ibsta and wait mask                */
ERR   =   '8000'x  /* Error detected                           */
TIMO  =   '4000'x  /* Timeout                                  */
END   =   '2000'x  /* EOI or EOS detected                      */
SRQI  =   '1000'x  /* SRQ detected by CIC                      */
RQS   =   '0800'x  /* Device needs service                     */
SPOLL =   '0400'x  /* Board has been serially polled           */
EVENT =   '0200'x  /* An event has occurred                    */
CMPL  =   '0100'x  /* I/O completed                            */
LOK   =   '0080'x  /* Local lockout state                      */
REM   =   '0040'x  /* Remote state                             */
CIC   =   '0020'x  /* Controller-in-Charge                     */
ATN   =   '0010'x  /* Attention asserted                       */
TACS  =   '0008'x  /* Talker active                            */
LACS  =   '0004'x  /* Listener active                          */
DTAS  =   '0002'x  /* Device trigger state                     */
DCAS  =   '0001'x  /* Device clear state                       */

/*    Get the GPIB0 address since we are going to make board   */
/*      level GPIB calls.                                      */
brd0 = ibfind( "GPIB0" )

/*    Set time-out to 10ms so that we don't stop Windows for a */
/*      long period.                                           */
ibtmo( brd0, 9 )    /* 9 = 10ms */

bcommand = 0       /* = 1 if we have received a command        */
ret_value = '0'    /* the return value from that command       */

begin:
say "Waiting for GPIB input..."

/*     Enter the GPIB polling loop                             */
do forever
    /*    Get the GPIB status variable                         */
    ireturn = iblines( brd0 )
    /*    Parse the return value into ibsta and ilines         */
    parse var ireturn ibsta ilines

    /* if the listen line is set read in command if we         */
```

```
    /* haven't already read in a valid command                  */
    if ( bitand( d2c(ibsta), LACS, 0 ) = LACS ) then do
        if bcommand = 0 then do
            bcommand = 1
            say "listen..."
            command = ibrd( brd0, 127 )
            /* Remove CR at end of command if it exist      */
            cr_pos = lastpos( 'A'x, command,  )
            if ( cr_pos <> 0 ) then
                command = left( command, cr_pos - 1, )

            say command        /* echo the command              */

            /* Use the interpret command to execute the         */
            /*  REXX function received via GPIB, the            */
            /*  return value from this command is saved         */
            /*  in ret_value                                    */
            execute_command = "ret_value =" command
            interpret execute_command
        end
    end

    /* if the talker line is set write out command result,      */
    /* if a command was received                                */
    if ( bitand( d2c(ibsta), TACS, 0 ) = TACS ) then do
        if bcommand = 1 then do
            bcommand = 0
            say "Talk..."
            ibwrt( brd0, ret_value, length(ret_value) )
            say ret_value
        end
    end
end /* do forever */

/*********************************************************************/
/*    syntax exception handler - called if this REXX program can not */
/*     execute a command, i.e. bad command sent to interpret function.  */
/*********************************************************************/
syntax:
    say 'REXX error' rc '('errortext(rc)||,
        ') occurred in line' sigl'.'
    say '=====>' command
    ret_value = "ERROR"
    signal on syntax
    signal begin

/*********************************************************************/
/*    Halt exception handler                                         */
/*    Called when someone hits the stop button.                      */
/*********************************************************************/
halt:
    reply = questionbox( "Do you wish to exit LabMaster GPIB slave mode?",,
                         "LabMaster GPIB" )
    if reply = 'No' then do
    call on halt
        return
           end
    else do
    say "GPIB link terminated."
    exit
    end
```

5.7.3 LabMaster GPIB CIC Mode - Sample Program

The following program, **GPIB_CIC.REX**, shows a framework for interfacing LabMaster with an external device via the GPIB bus. This program is located in the PROGRAMS subdirectory.

```
/*********************************************************************
 *  GPIB_CIC.REX
```

```
 *
 *  Example REXX program showing the framework for interfacing LabMaster
 *  with an external GPIB device via a National Instruments PCI-GPIB card.
 *  In this case the computer running LabMaster is the Controller-In-Charge
 *  (CIC) of the GPIB bus.
 *
 *  1.0  01/25/94  mac  Creation.
 ***********************************************************************/

/* Un-comment the following line to trace the program's execution      */
/*trace ?i */

/*  set debug to TRUE if you want GPIB commands echoed to I/O window    */
debug = "FALSE"

/*  Add external functions located in the LMGPIB.DLL.                  */
RxFuncAdd( "ibfind",  "LMGPIB.DLL", "GPIBfind" )
RxFuncAdd( "ibwrt",   "LMGPIB.DLL", "GPIBwrt" )
RxFuncAdd( "ibrd",    "LMGPIB.DLL", "GPIBrd" )

/*  Set 'dev' to the GPIB name selected using the GPIBconfig program   */
/*  Typically this is just 'DEV' followed by the address number.        */
dev = "DEV3"
addr = ibfind( dev )
call debugsay "ibfind = " addr

/*  Report an error if ibfind() failed                                 */
if ( addr < 0 ) then do
    messagebox( "ERROR: Could not find" dev "using ibfind", "LabMaster" )
    say "ERROR: Could not find" dev "using ibfind"
    exit
    end


/*  Example of how to send a command to the device                     */
command = "Example GPIB Command"
ibwrt( addr, command, length(command) )
call debugsay "ibwrt(" addr "," command ")"


/*  Example of how to read data from the device                        */
data = ibrd( addr, 128 )
call debugsay "ibrd(" addr ") = " data

exit   /*** End of Main Program ****************************************/


/***********************************************************************/
/*  debugsay subroutine                                                 */
/*    Used to write debug information to the I/O window */
/***********************************************************************/
debugsay:
    if debug = "TRUE" then
           say arg(1) arg(2) arg(3) arg(4) arg(5) arg(6) arg(7)
return ""
```

5.7.4 IEEE 488.1 (GPIB) REXX Function Prototypes

See the National Instruments *NI-488.2 Software Reference Manual for MS-DOS*, May 1992, NI Part Number 320282-01, for a complete description of the following 488.1 and 488.2 GPIB functions.

The following functions are located in the LMGPIB.DLL and all begin with "GPIB" instead of the "ib" used by NI. For example, the REXX GPIBfind( ) function is equivalent to the NI ibfind( ) function.

GPIBbna(ud, bname)

GPIBcac(ud, v)

GPIBclr(ud)

GPIBcmd(ud, cmd, cnt)

GPIBcmda(ud, cmd, cnt)

GPIBconfig(ud, option, value)

GPIBdev(boardindex, pad, sad, tmo, eot, eos)

GPIBdma(ud, v)

GPIBeos(ud, v)

GPIBeot(ud, v)

ud = GPIBfind(udname)

GPIBgts(ud, v)

GPIBist(ud, v)

clines = GPIBlines(ud)
NOTE: clines = "ibsta iblines"
Use the REXX PARSE command to parse clines into ibsta and iblines
example: "parse clines ibsta iblines".

listen = GPIBln(ud, pad, sad)

GPIBloc(ud)

GPIBonl(ud, v)

GPIBpad(ud, v)

GPIBpct(ud)

GPIBppc(ud, v)

rd = GPIBrd(ud, cnt)

rd = GPIBrda(ud, cnt)

GPIBrdf(ud, flname)

ppr = GPIBrpp(ud)

GPIBrsc(ud, v)

spr = GPIBrsp(ud)

GPIBrsv(ud, v)

GPIBsad(ud, v)

GPIBsic(ud)

GPIBsre(ud, v)

GPIBstop(ud)

GPIBtmo(ud, v)

GPIBtrap(mask, mode)

GPIBtrg(ud)

GPIBwait(ud, mask)

GPIBwrt(ud, wrt, cnt)

GPIBwrta(ud, wrt, cnt)

GPIBwrtf(ud, flname)

5.7.5 IEEE 488.2 (GPIB) REXX Function Prototypes

GPIBDevClear(board, address)

GPIBPassControl(board, address)

result = GPIBPPoll(board)

GPIBPPollConfig(board, address, dataline, sense)

data = GPIBRcvRespMsg(board, count, termination)

result = GPIBReadStatusByte(board, address)

data = GPIBReceive(board, address, count, termination)

GPIBReceiveSetup(board, address)

GPIBSend(board, address, data, count, eotmode)

GPIBSendCmds(board, commands, count)

GPIBSendDataBytes(board, data, count, eotmode)

GPIBSendIFC(board)

GPIBSendLLO(board)

result = GPIBTestSRQ(board)

GPIBTrigger(board, address)

result = GPIBWaitSRQ(board)

6 Dynamic Data Exchange Interface

6.1 Introduction

This chapter is intended for experienced Windows users who wish to write Windows applications to control and communicate with LabMaster remotely.

Dynamic Data Exchange (DDE) allows Windows applications to exchange data on a real-time basis. Two Windows programs can engage in a DDE conversation by posting messages to each other. In a DDE conversation one program is called the "server" and the other the "client". A DDE server application contains information or performs functions that may be useful to other windows applications. A DDE client program initiates the conversation with the server to retrieve information from or execute functions in the DDE server application. LabMaster is a DDE server application. LabMaster can also act as the client by using the DDE REXX commands, see Chapter 5 for a description of these functions.

By using DDE, Windows client applications can execute most of the REXX functions within LabMaster versions 2.0 and higher. These client applications can also request device position and wafer map information from LabMaster. DDE allows the user to create programs or macros written in another Windows application such as Microsoft Word, Microsoft Excel, National Instruments LabView for Windows, Microsoft Visual BASIC, HP BASIC for Windows, and many others to control test equipment as well as LabMaster. For more information see the documentation provided with client application that you use to communicate with LabMaster.

6.2 Initiating a DDE Conversation with LabMaster

The Initiate function within a client application opens a DDE channel between a client and server. A Windows program will use its Initiate function to start a DDE conversation with LabMaster. The Initiate function has two parameters, the Server Application Name and the Topic. The Server Application Name is the server program executable filename without the .EXE extension. The Topic identifies the item within the server application that you wish to communicate with.

The Initiate function returns a channel number for the Topic. This channel number is used as a parameter in all other DDE function calls used to communicate with that Topic. An error is returned if the server application is not currently running or an invalid Topic was specified.

The LabMaster Server Application Name is "LM". The valid LabMaster Topics are specified below.

Topic	Purpose
System	Used to execute LabMaster REXX functions.
Position	Used to request LabMaster device position information
Map	Used to request Wafer Map information

An application uses the Terminate function to close a DDE channel. The only parameter to the Terminate function is the DDE channel number returned from the Initiate function. An error is returned if the channel number is not valid. All channels should be closed in this manner when they are no longer being used.

6.3 Executing a REXX Command

The DDE Execute function sends commands to a server application. The Execute function has two parameters, the Channel Number and the Execute String. The Channel Number is the value returned from the Initiate command. An error is returned if the Channel Number is not valid, or if any errors occur when the server executes the command in the Execute String. The Execute function does not return data to the client application, therefore REXX commands that return information should not be used (e.g. GetXPosition( )). See Chapter 5 for more information on REXX.

To execute a LabMaster REXX command first initiate a DDE conversation with the LabMaster topic "System". Use the client application's Execute function using the channel number returned from the Initiate function and the REXX command to be executed. Below is example Microsoft Visual Basic code that initiates a conversation with LabMaster and display the version of LabMaster.

Example:

```
Sub Main()
 Channel = DDEInitiate("LM", "System")
 DDEExecute Channel, "GetSWVersion()"
 Version = DDERequest(Channel, "LastReturnValue")
 MsgBox Version, , "GetSWVersion()"
 DDETerminate Channel
End Sub
```

6.4 Requesting Information From LabMaster

The DDE Request function retrieves information from a server application. The two parameters to the Request function are the Channel Number and the Item. The Channel Number is the value returned by the Initiate function, and the Item is the data being requested. The server application returns the data requested. An error is returned if the Channel Number is invalid, the Item is not valid, or the server cannot return the data.

Windows applications can request either LabMaster device position information or Wafer Map information. The items that can be requested from LabMaster for the Map and Position topics are listed below.

System Topic Items:

Item	Result
LastReturnValue	Returns the last returned value from a LabMaster REXX function.

Position Topic Items:

Item	Result
PS	Returns the probe station chuck's x, y, and z position separated by commas.
PMM	Returns the PMM's x, y, and z position separated by commas.
CAP1	Returns CAP one's x, y, and z position separated by commas
CAP2	Returns CAP two's x, y, and z position separated by commas
CAP3	Returns CAP three's x, y, and z position separated by commas
CAP4	Returns CAP four's x, y, and z position separated by commas
CAP5	Returns CAP five's x, y, and z position separated by commas
CAP6	Returns CAP six's x, y, and z position separated by commas
Theta	Returns the theta position of the prober's chuck in radians.

Map Topic Items:

Item	Result
NumTestDie	Returns the number of die that are marked as test die.
NumDie	Returns the total number of die on the wafer map.
BinNumber	Returns the bin number of the current die.
BinComment	Returns the bin comment of the current die.
IsSkipDie	Returns "TRUE" if the current die is a die to be skipped when testing, and "FALSE" otherwise.
IsDieTestStructure	Returns "TRUE" if the current die is a test structure, and "FALSE" otherwise.
IsDiePartial	Returns "TRUE" if the current die is a Partial die, and "FALSE" otherwise.
IsInkOnlyDie	Returns "TRUE" if the current die is an Ink-only die, and "FALSE" otherwise.
IsDie	Returns "TRUE" if the current chuck position is at a die located on the wafer map, and "FALSE" otherwise
DieComment	Returns the current die's comment text.
FlatPosition	Returns one of the following: "None", "0", "90", "180", "270" indicating the flat or notch location of the wafer on the chuck.
XDieSize	Returns the die's X size in millimeters.
YDieSize	Returns the die's Y size in millimeters.
XStreet	Returns the X axis street size between the die in millimeters.

YStreet	Returns the Y axis street size between the die in millimeters.
XDiePosition	Returns the X axis die position of the chuck, relative to the home die location.
YDiePosition	Returns the Y axis die position of the chuck, relative to the home die location.
XDieAbsPosFromLeft	Returns the X axis absolute die position of the chuck relative to the left side of the wafer. I.E. the left most column is absolute position 1 and the values increase as you move right across the wafer.
XDieAbsPosFromRight	Returns the X axis absolute die position of the chuck relative to the right side of the wafer. I.E. the right most column is absolute position 1 and the values increase as you move left across the wafer.
YDieAbsPosFromTop	Returns the Yaxis absolute die position of the chuck relative to the top of the wafer. I.E. the top most row is absolute position 1 and the values increase as you move down the wafer.
YDieAbsPosFromBot	Returns the Yaxis absolute die position of the chuck relative to the bottom of the wafer. I.E. the bottom most row is absolute position 1 and the values increase as you move up the wafer.

To request information from LabMaster, first initiate a DDE conversation with the LabMaster topic "Map", "System", or "Position". Use the client application's Request function with the channel number returned from the Initiate function and the item of the data requested. Below is example Visual Basic code that initiates a conversation with LabMaster's Position and Map topics to retrieve some device position information and wafer map data.

Example:

```
Sub Main()
 ProgChannel = DDEInitiate("LM", "System")
 PosChannel = DDEInitiate("LM", "Position")
 MapChannel = DDEInitiate("LM", "Map")
 Position = DDERequest(PosChannel, "PS")
 MsgBox Position, , "Position"
 DDEExecute ProgChannel, "LoadMap('Example.wmf')"
 NumTestDie = DDERequest(MapChannel, "NumTestDie")
 MsgBox NumTestDie, , "NumTestDie"
 DDETerminate ProgChannel
 DDETerminate PosChannel
 DDETerminate MapChannel
End Sub
```

6.5 LabMaster DDE Status Dialog Box

The DDE Status dialog box is used to display information regarding the status of any DDE connections that applications may have with LabMaster. It can only be activated remotely by using the clients DDE execute command with the LabMaster's "System" channel number and "ECHO ON" as the execute string. It can be deactivated by using "ECHO OFF" as the execute string, or by pressing the **Cancel** button on the dialog box.

The DDE Status dialog box displays the DDE command last received from the client application, the status of that command, and the number of DDE channel connections with client applications. Pressing the **Abort...** button will cause the next DDE command to fail causing a fail message to be sent to the client application aborting the DDE connection.

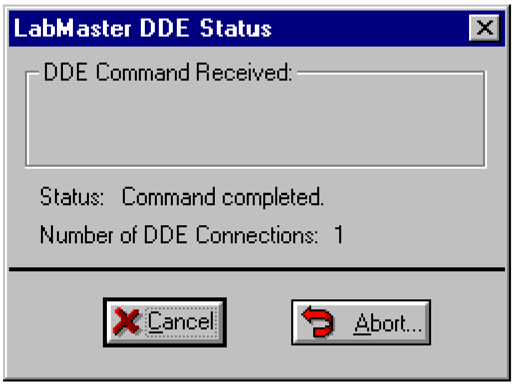


Figure 6-1 LabMaster DDE Status Dialog Box

Below is an example of Visual Basic code that turns the LabMaster DDE Status dialog box on and then off.

Example:

```
Sub Main()
 Channel = DDEInitiate("LM", "System")
 DDEExecute Channel, "ECHO ON"
 MsgBox "Press OK to close the Status Dialog"
 DDEExecute Channel, "ECHO OFF"
 DDETerminate Channel
End Sub
```

Appendix A - LabMaster Quick Start Guide

This appendix describes the basic LabMaster procedures and is intended for new users who wish to get up and running on LabMaster quickly. For a more in-depth discussion on LabMaster see Chapter 2 or LabMaster's on-line help. The procedures below assume that LabMaster has already been properly installed and configured.

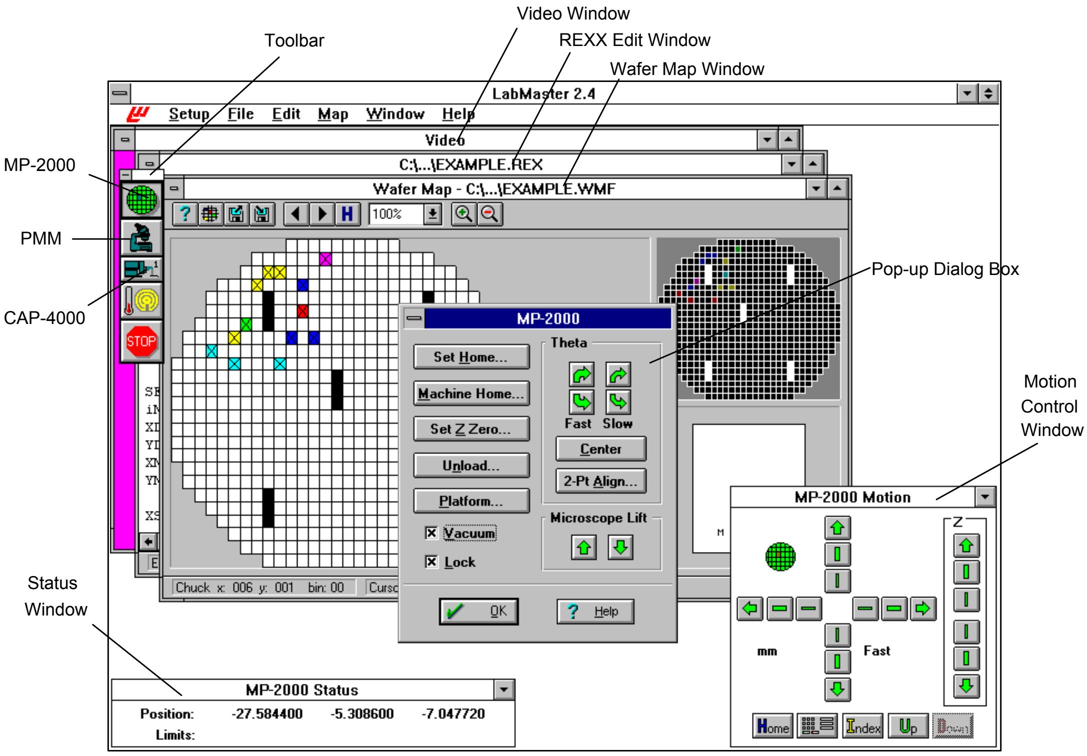


Figure A-0-1 LabMaster Main Window

Starting LabMaster

1. Turn on the computer and then power up the Pegasus Controller. The Pegasus Controller is used to drive the stepper motors in the Pegasus Prober, the Programmable Microscope Mount (PMM), and the (CAP) Computer Assisted Probe(s).
2. Start LabMaster by double-clicking on the **LabMaster** icon in the Wentworth program group.

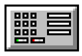
3. When LabMaster has finished initializing, the real-time Video Window will be displayed along with the LabMaster Device Toolbar. The Video window can be re-sized and moved to a new location at any time.

Moving Devices

- Devices that you wish to move must first be selected on the LabMaster Toolbar. The Toolbar contains a graphical representation of the devices that LabMaster controls. Use the **left** mouse button to select the button of the device that is to be moved. The selected devices on the Toolbar are shown in different colors to indicate that they are active.

- If you select more than one device, both devices will move at the same time. The last device selected is called the ***master*** and the devices previously selected are called ***slaves***. If you move the master, the slaves will follow the master's movements.

- When you select a device on the Toolbar, the device's Motion Control window and Status window will be displayed. The Motion Control window is used to move the device and the Status window shows the device's X-Y-Z position and limit switch status.

- Devices can be moved to an absolute location by selecting the  button on the device's Motion Control window. This button displays the Numerical Entry dialog box, which allows for the entry of an absolute device position. Pressing the **OK** button will send the device to the specified location.

- The **STOP** button on the Toolbar is used to stop a device's motion. This button **will not** stop an MP-1100 probe station, the MP-1100's stop button on the MP-1100's keyboard must be used instead.

- See Chapter 3 for information on how to move devices using the Video window.

- See Chapter 4 for information on moving devices using the Wafer Map window.

Displaying a Devices' Pop-up Dialog Box

- A device's Pop-up dialog box contains frequently used functions for that device. For example the Prober's Pop-up dialog box is used to adjust the chuck's theta position, set a user defined home position, move the chuck to the unload position, turn the chuck's lock and vacuum on and off, etc.

- To display a device's Pop-up dialog box, use the **right** mouse button and click on the devices' button in the LabMaster Toolbar.

- The Pop-up dialog box for the master device can also be displayed by double-clicking the left mouse button in the Video window.

Displaying a Devices' Setup Dialog Box

- A Setup dialog box is used to change the device's setup parameters. For example the Prober's Setup dialog box is used to set the speed the stage is moved, the index mode step sizes, find the machine home position, etc.

- To display a device's Setup dialog box, Select the devices name under the Setup menu. To remove the dialog box, select the **OK** button.

Appendix B - Mouse Quick Reference

Window	Mouse Button	Key Held Down	Result
Video	Left		If the video is frozen the area of the cut and paste rectangle is sized by dragging the mouse. If the Ruler is active the distance measured by the ruler is marked by dragging the mouse.
	Right		If the ptober or PMM is selected on the toolbar that device is moved to the point selected by the mouse on the video window. To move a CAP, select the CAP on the Toolbar, then select the tip of the probe and drag to the new location of the probe.
	Left-double click		Displays the pop-up dialog box for the master device.
Toolbar	Left		Activates or deactivates the device that is selected
	Right		Displays the pop-up dialog box for the device selected
Map Top View	Left		Used to drag the zoomed area rectangle to a new location.
	Right		Moves the chuck to the area selected.
Map Zoom View	Left-double click		Activates the Die Setup dialog box for the die selected.
	Left	SHIFT	Marks or unmarks a die as a skip die.
	Left	CTRL	Adds or removes a die from the wafer map.
	Left	CTRL + SHIFT	Assign a die the bin number last selected using the Die Setup dialog box.
	Right		Moves the chuck to the die selected.
	Right	SHIFT	Marks or unmarks a die as an ink-only die.
Map Die View	Left		Used to cycle through the device windows when they are on top of each other.
	Right		Used to drag a device window to a new location on the die.
REXX Edit Window	Right		Displays the pop-up context menu.

Appendix C - Wafer Map File Format

This appendix describes the Wentworth wafer-map file format. It is a tab-delimited text file format, allowing wafer map files to be imported into a spreadsheet like Microsoft Excel, or viewed by a text editor like Notepad. The following format must be strictly followed and must include all information listed in the Contents column. [TAB] is used to indicate a tab character (ASCII 8) and [EOL] is used to indicate a carriage return character (ASCII 13) followed by a line feed character (ASCII 10).

Line #	Contents	Comment
1	`FILE_VER[TAB]`**`310`**`[EOL]`	The wafer-map file format version (currently 3.1.0).
2	`WaferName[TAB]`***`szWaferName`***`[EOL]`	***`szWaferName`*** must be less than 127 characters long.
3	`WaferComment[TAB]`***`szWaferComment`***`[EOL]`	***`szWaferComment`*** must be less than 255 characters long.
4	`Scale[TAB]`***`iScale`***`[EOL]`	***`iScale`*** = 0 for mm, 1 for inches. This is the displayed scale, the wafer-map stores in distances in nanometers (nm).
5	`FlatAngle[TAB]`***`iFlatAngle`***`[EOL]`	***`iFlatAngle`*** = angle of the wafer flat/notch in degrees, clockwise from the bottom of the wafer. Set to –1 if no flat/notch is present.
6	`WaferSize[TAB]`***`nmWaferSize`***`[EOL]`	***`nmWaferSize`*** = wafer size in nm.
7	`XDie[TAB]`***`nmXDie`***`[EOL]`	***`nmXDie`*** = x die size in nm.
8	`YDie[TAB]`***`nmYDie`***`[EOL]`	***`nmYDie`*** = y die size in nm.
9	`XStreet[TAB]`***`nmXStreet`***`[EOL]`	***`nmXStreet`*** = x street size in nm.
10	`YStreet[TAB]`***`nmYStreet`***`[EOL]`	***`nmYStreet`*** = y street size in nm.
11	`NumberRow[TAB]`***`iNumRows`***`[EOL]`	***`iNumRows`*** = total number of wafer map rows.
12	`NumberCol[TAB]`***`iNumColumns`***`[EOL]`	***`iNumColumns`*** = total number of wafer map columns.
13	`XHome[TAB]`***`iDieXHome`***`[EOL]`	***`iDieXHome`*** = die x home position relative to bottom left of wafer map.
14	`YHome[TAB]`***`iDieYHome`***`[EOL]`	***`iDieYHome`*** = die y home position relative to bottom left of wafer map.
15	`OperatorID[TAB]`***`szOperatorID`***`[EOL]`	***`szOperatorID`*** must be less than 50 characters long.
16	`MaterialID[TAB]`***`szMaterialID`***`[EOL]`	***`szMaterialID`*** must be less than 50 characters long.
17	`LotID[TAB]`***`szLotID`***`[EOL]`	***`szLotID`*** must be less than 50 characters long.
18	`WaferID[TAB]`***`szWaferID`***`[EOL]`	***`szWaferID`*** must be less than 50 characters long.
19	`NumberBins[TAB]`***`iNumBins`***`[EOL]`	***`iNumBins`*** = number of pass/fail bins (currently 128). This number does not include the untested bin (bin 0).
20	`Application[TAB]`***`szApplication`***`[EOL]`	***`szAppliciation`*** = the name and version number of the application that created or modified this file (eg, LabMaster(UK) 8.5). Must be less than 100 characters long.

21	`*****Bin Info*****[EOL]`[1]	

22	`Number[TAB]Red[TAB]`
	`Green[TAB]Blue[TAB]`
	`Flags[TAB]Comment[EOL]`	

(The following line is repeated for each bin; for bins 0 to iNumBins)

23 - 151	***iBinNumber***[TAB]	***iBinNumber*** = 0 to iNumBins.
	iRed[TAB]	***iRed*** = 0 to 255.
	iGreen[TAB]	***iGreen*** = 0 to 255.
	iBlue[TAB]	***iBlue*** = 0 to 255.
	uFlags[TAB]	**Bit 0** = 1 if a pass bin or 0 if a fail bin **Bit 1** = 1 if a retestable bin
	szBinComment[EOL]	***szBinComment*** must be less than 127 characters long.

152	`*****Die Info*****[EOL]`[2]	

153	`Column[TAB]Row[TAB]`
	`BinNum[TAB]Die[TAB]`
	`Flags[TAB]Comment[EOL]`	

(The following line is repeated for each die in the wafer-map; for columns 0 to iNumColumns-1, rows 0 to iNumRows-1)

155 -	***iColumn***[TAB]	***iColumn*** = 0 to iNumColumns-1.
	iRow[TAB]	***iRow*** = 0 to iNumRows-1.
	iBinNumber[TAB]	***iBinNumber*** = 0 to iNumBins.
	bDie[TAB]	***bDie*** = 1 if it is a die, 0 otherwise.
	uFlags[TAB]	**Bit 0** = 1 if it is a partial die **Bit 1** = 1 if it is an ink-only die **Bit 2** = 1 if it is a sample die **Bit 3** = 1 if it is a forced ink die **Bit 4** = 1 if it is a test structure **Bit 5** = 1 if it is a skip die **Bit 6** = 1 if it is a alignment die
	szDieComment[EOL]	***szDieComment*** must be less than 17 characters long.

[1] Future versions of the wafer-map file format may add extra lines to the previous section. Therefore we strongly recommend that any software written to read the wafer-map file scans for this bin header line, rather than assuming it will be on a particular line.
[2] Future versions of the wafer-map file format may change the number of bins. Therefore we strongly recommend that any software written to read the wafer-map file scans for this die header line, rather than assuming it will be on a particular line.

Appendix D - Probing Procedures

This appendix describes the basic LabMaster procedures needed to use a Pegasus probe station with a probe card and manipulator. The procedures below assume that LabMaster has already been properly installed and configured for a Pegasus Prober, Programmable Microscope Mount (PMM), and Computer Assisted Probe (CAP). For a more in-depth discussion on LabMaster see Chapter 2 or LabMaster's on-line help.

LabMaster and Pegasus initialisation

1. Start the PC and Pegasus controller. Initialise the Pegasus Prober from it's keypad

2. Double-click on the LabMaster icon in the Wentworth program group to start the LabMaster program initialisation.

    a. **LabMaster will prompt you before the Prober searches for its machine home position.**

    b. **After the Pober has found its machine home position, LabMaster has finished its initialisation.**

3. Raise the microscope to its upper limit.

    a. **Click the right mouse button on the Pober button in the Toolbar to display the Pober Pop-up dialog box.**

    b. **Use the microscope lift buttons to raise the microscope.**

4. Find the PMM's machine home position.

    a. **Select the** Setup | PMM... **menu item to display the PMM Setup dialog box, and then press the** Find Machine Home... **button to have the PMM search for its machine home position**

5. Raise the platform to its upper limit.

    a. **Click the right mouse button on the Prober button in the Toolbar to display the Prober Pop-up dialog box.**

    b. **Select the Platform button to display the Platform dialog box and use the up button to raise the platform.**

6. Lower the chuck to its bottom limit.

    a. **Select the Prober button in the Toolbar.**

    b. **Use the arrow keys in the Motion Control Window to move the chuck to its bottom limit.**

7. Move the chuck to the unload position.

8. Load the wafer, and turn on the vacuum.

9. Move the chuck back to the center of its X and Y axis.

   a. **Click the right mouse button on the probe station button in the Device Selection Toolbar to display the Prober Pop-up dialog box. Press the** Machine Home... **button to move the chuck to the machine home position.**

10. Install the probe card in the holder and clamp down.

11. Lower the platform so that the probe tips are approximately 2.5 mm (0.1 in) above the wafer.

12. Select the 2X objective and lower the microscope. Focus the microscope on the probe tips.

13. Raise the chuck until the wafer comes into focus but do not make contact with the wafer.

14. Increase magnification to 10X and re-focus.

15. Align the wafer using the 2-Point Theta adjustment (either the X axis or the Y axis can be used for this. The X axis is used in the steps outlined below).

   a. **Click the right mouse button on the Prober button in the Toolbar to display the Prober Pop-up dialog box.**

   b. **Select the** 2-Pt Align... **button to display the 2-Point Theta dialog box.**

   c. **Align the cross hair in the center of the Video window with a X-axis scribe line on one side of the wafer and select the** Point A **button.**

   d. **Slowly move the stage in the X-axis direction for approximately 50 mm (2 in).**

   e. **Adjust the Y-axis to reposition the cross hair on the scribe line and press the** Point B **button.**

   f. **Press the** Adjust Theta... **button to correct for the theta error.**

   g. **Repeat at higher magnifications for increased accuracy.**

16. Align the bond pads under the appropriate probe tips while using the 2X objective to get a rough alignment.

17. Adjust the theta on the probe card holder with the adjustment bar provided so that the probe tips are parallel with the bond pads.

18. Increase to the 10X objective and insure that all of the probe tips are centered. Lock down the PCH so that it will not rotate.

19. Move the chuck up so that the probes just make contact with the pads. Make sure that the chuck is in its UP state by viewing the Prober Status window.

20. Set the chuck drop distance to approximately 0.25 mm (10 mil) and the overdrive distance to approximately 0.075 mm (3 mil).

    a. **Select the** Setup | **Prober**... **menu item to display the Prober Setup dialog box.**

    b. **Enter the chuck drop and overdrive distances.**

21. The **UP** and **DOWN** buttons in the Motion Control window can now be used to overdrive your probes on each die.

22. Set the Prober X and Y large step sizes to the die-to-die stepping distances. When the Index button is selected in the Motion Control window, the X and Y axis fast arrow buttons can now be used to index die-to-die.

    a. **Select the** Setup | **Prober**... **menu item to display the MP-2000 Setup dialog box.**

    b. **Enter the X and Y axis index sizes.**

CAP Setup

1. Select the CAP's button on the Toolbar.

2. Using the arrow button in the Motion Control window, position the CAP's X-Y-Z axis' white alignment dots so that they are in the center of each axis.

3. Set this position as the CAP's machine home position.

    a. **Select the** Setup | CAP... **menu item to display the CAPSetup dialog box.**

    b. **Press the** Set Machine Home... **button.**

4. Insert the probe tip into the coax front end. The probe tip should extend about 6 to 9 mm (1/4 to 3/8 inch) depending on the probe card depth.

5. Focus the highest power objective on the device. Since the highest power objective has the shortest working distance, using this objective will help to insure that the objectives and probe tip will not make contact with each other.

NOTE: Mititoyo objective working distances:

2X	34 mm	1.34 in
10X	33 mm	1.30 in
20X	20 mm	0.79 in
50XHR	13 mm	0.51 in
100X	6 mm	0.23 in

6. Carefully bend the CAP's front end assembly down into the opening of the probe card and position it underneath the objective without touching the device.

7. Raise the microscope and reduce the magnification.

   a. **Click the right mouse button on the Prober button in the Toolbar to display the Prober Pop-up dialog box.**

   b. **Use the microscope lift buttons to raise the microscope.**

8. If needed, follow the CAP's 3-point Align procedure in Chapter 2.

Questions?

If you have any questions or comments about the LabMaster software or any Wentworth product, please direct your inquiry to your local sales representative or Wentworth offices:

Wentworth Laboratories, Inc.
500 Federal Road
Brookfield, CT 06804-2098
Telephone: +1 203-775-0448
Fax: +1 203-775-8172

Wentworth Laboratories, Inc.
1046 Morse Ave.
Sunnyvale, CA 94089
Telephone: +1 408-745-1644
Fax: +1 408-744-0348

Wentworth Laboratories, Ltd.
Sunderland Road
Sandy Beds, SG19, 1RB
ENGLAND
Telephone: +44 (0) 767-681-221
Fax: +44 (0) 767-691-951

Wentworth Japan Co. Ltd.
Plumfield Bldg. 4F
1-7-5 Miyamaedaira
Miyamae-ku,
Kawasaki 216
JAPAN
Telephone: +81 (0) 44-857-0411
Fax: +81 (0) 44-857-1577

Web Site
http://www.wentworthlabs.com

Comments

To enable us to make improvements in our equipment, software, and documentation we would ***greatly*** appreciate any comments you may have.

If you have any comments or suggestions please complete the following form and mail or fax to:

Wentworth Laboratories, Ltd.
Sunderland Road
Sandy Beds, SG19, 1RB
ENGLAND
Telephone: +44 (0) 767-681-221
Fax: +44 (0) 767-691-951

You can also contact Wentworth Laboratories on info@wentworthlabs.com.

Name ____

Title ____

Company ____

Address ____

____

City ____________________ **State** ______ **Zip Code** ________

Country ____________________ **Date** ____________________

Please rate LabMaster on the following items using a scale of 1 to 10, with 10 being the best:

LabMaster in general ______
Ease of use ______
LabMaster's REXX Interface ______
LabMaster's Wafer Mapping ______
User Manual in general ______
Readability of User's Manual ______
Contents of User's Manual ______

Comments: ____

____

____

____