nSoft User's Guide

Machine Tool Generation 3 (MTGen3)

December 2011 Edition

Software Operation Manual

Sections

Table of Contents

1 Introduction

This document describes the operation of Machine Tool, Generation III (MTGen3). MTGen3 provides the operator interface for nScrypt dispensing machines. Manual motion controls, machine status information, process management controls, and various editors are provided through the user interface. Users define a project-driven process where a hierarchy of areas, set within the machine's range of motion, are used to determine where work should occur. During processing, spatial coordinates are calculated based on the areas defined and the calibration of the attached devices to ensure positional accuracy and repeatability. The software also provides a variety of automated processes used for orientation, dispensing, and data collection, which are all performed relative to the user defined areas.

This manual is organized by the type of automated processes grouped by their intent. An overview of dispensing is provided, followed by instructions on project creation and architecture, definitions and usage of automated process, and then calibration. The purpose of this document is to provide training and to assist users in learning the concepts involved when using the MtGen3 software. Please refer to the machine's maintenance manual for any hardware-related questions.

1.1 Installation

MTGen3 is typically installed in the file path C:/Transfer/Installs/MtGen3/MtGen3.exe. Contact nScrypt support if the application is not in this location and cannot be located. A shortcut may be created on the desktop for easy access.

1.2 Software Launch and Login

At the login page, select your operator type and enter your password. Most systems use the default operator "Administrator" and default password "admin". Based on the operator type, two modes of usage are available: **Application** and **Machine Settings**. Normal operation runs in **Application** mode and initial setup and calibration runs in **Machine Settings**.

1.2.1 User Levels

The software supports different user levels, providing and hiding features for each level:

- **Administrator** level is intended for the owners and managers of the machine.
- **Engineer** level is for personnel in charge of machine setup, maintenance, and process development.
- **Operator** level is for users who are in charge of running and monitoring a developed process.
- **Developer** level is for developers running various feature testing.

User passwords are managed and maintained in:
C:/Transfer/Installs/MtGen3/Config/ApplicationSettings.xml

1.2.2 Hardware Start

To operate the hardware, pull out the e-stop button and press the reset button to activate the system. In Application mode, click **Fault Acknowledge** then click **Reset** to ready the axes for motion. Test the axes operation by using the software's manual motion controls. Move the XY Motion Speed slider between 5 and 30 mm/s. Press the X and Y jog buttons to move the axis. This is done to verify that the stages are operational.

1.3 Quick Start

This section assumes that the machine has been configured and calibrated and that the valve body and tip are installed on the pumps as well. Step 1 creates a basic project for printing with no orientation adjustments. If there is a project already created, it may be loaded and Step 1 may be skipped.

<u>Step 1: Create or Load a Project</u>

Load a Print Project:

1) Select File -> Open
2) Navigate to the Projects directory and select a project.
3) Open the (*.mtp) file within the desired project folder.

--Or--

Create a New Project:

1) Select **File -> New** to create a new project.
2) Name the project and click **Save**.
3) Right-click **Substrate** in the project tree and select **Add Sub Piece**.
4) Select WorkPiece (**WKP**) from the WorkPiece creator.
5) To create a new work piece, ensure **Load from File** is unchecked and click **Next >>**.
6) Name the work piece (for example, "plastic sheet" or "my widget").
7) Ensure **Open WorkPiece Editor when finished is checked** and click **Finished.**
8) Name the work piece for the project tree display.
9) Enter the **Width**, **Length**, and **Height** of the substrate. (The width and length will be used for orientation. If orientation is not needed, specifying 0 for the width and length is acceptable.)

10) Click **Apply Changes** and click the close X.

11) Expand the newly added WorkPiece by clicking the '+' next to its name.

12) Right-click **Jobs**, and select **Add Job**.

13) Select the PrintScriptJob (**PSJ**), ensure **Load from File** is unchecked and click next.

14) Name the job file and click **Save**.

15) Ensure **Open Job Editor when finished is checked** and click **Finished.**

16) Set the PrintFilePath to the desired script file (.txt).

17) Name the job using JobName. This name appears in the project tree display.

18) Add a dispense gap in millimeters to avoid scraping the surface.

19) Click **Apply Changes**, select **Yes**, reload script changes, and close the editor.

Step 2 : Create or Load a Project

1) Click the **Gizmos** tab on the main display.

2) Select a device in the Gizmo Selection drop down.

3) Drive the selected device to the origin of the substrate.

4) Right-click on the WorkPiece in the Project tree and select **Teach Origin**

5) Load material for your printing device.

6) Set the closed position. (See section 2.5.2 Valve Control.)

7) Set the desired **Material Feed Pressure**. (See section 2.5.2 Pressure.)

8) Click the GO button to run the project.

9) Use the STOP button to stop the project and close the valves.

2 Manual Operation

2.1 Motion Control

WARNING: Use extreme caution (use lower speeds) when operating the machine manually to avoid crashes which will result in severe and costly equipment damage.

The X, Y, and Z axes are controlled using the manual motion controls in the XYZ Controller. Click-and-hold style buttons are used to move the axes at the specified axes motion speeds. Sliders are provided to adjust the movement speed. The default speed is set to the minimum of 0.1 units/s. Reasonable movement speeds are 10 to 30 units/s for X and Y and 1 to 10 units/s for Z. These speeds are only used for the manual motion controls.

The Move to Position (absolute) area allows users to move to a known or saved location. The **M** button for each axis will match the current position of the machine on that axis. Use the **M** button to store the current axis position, and use the **Go** button to send the machine from its current location to the stored position on that axis. Multiple axes can be moved to their given positions by using the **Go to Position** button. The check boxes determine which axes will be moved when this button is pressed. Values may be typed into the **Move to Position (absolute)** text boxes instead of matching the current position.

2.2 System Lighting

To adjust the system lighting, from the top tool bar select **Controls -> System Lights** and change the corresponding sliders to adjust the illumination. The **Enable Lights** check box will turn all enabled lights on or off. The **Light Enabled** check box under each light slider tells the lights to individually turn on/off. Check/uncheck the **Enable Lights** checkbox on the right to activate the controls.

2.3 Monitors

Systems may have up to three screens at the operator console. All systems are provided with a computer screen (the **screen**) and some systems are provided with one or two **monitors** for observing the process *in situ*.

Note that the process view cameras viewed on the monitors and the fiducial find camera is viewed in MTGen3 in the **Camera** gizmo tab. See section 2.6 Target Camera.

2.4 Gizmo Selection

Devices registered to the tool plate in the Machine Settings are selected from a drop down list on the **Gizmos** tab. Selecting a gizmo allows access to its manual controls.

2.5 SmartPump™

SmartPump™ micro-dispensing pumps are adjustable in two ways. Material feed pressure supplies precise air pressure to the back of the material syringe, thus pushing material into the valve body. The valve body allows material flow to be very precisely controlled by adjusting the position of the valve rod between open and closed.

2.5.1 Pressure Control

Material feed pressure may be adjusted from 1 to 100 psi using the **Material Feed** slider or value box. Use the **Enabled** check box to turn the pressure on and off. To conserve compressed air, turn off the pressure when stepping away from the system. Typical pressures may be between 5 and 30 psi.

Spray pressure is configured only on certain systems with high air flow sprayers. A separate air pressure regulator is used on these systems, a separate control is available on the SmartPump™ gizmo control.

2.5.2 Valve Control

For the valve, 0.0 mm is the maximum closed position and 4.0 mm is the maximum open position. A typical closed position may be 1.200 mm. However, the closed position must be calibrated each time a material with a different viscosity is loaded or the lower o-ring is changed to compensate for small changes in the operation of the valve.

The valve opening (the difference between the open and closed positions) is adjusted in the script and works in conjunction with the feed pressure to set the desired flow rate, which (along with the printing speed) determines line width. A typical valve opening may be 0.500 mm written into the script which assuming a closed position at 1.200 mm would equate to a valve open position of 1.700 mm as displayed the SmartPump™ gizmo control.

Manual valve operation exercise:

First time users should perform these steps without a pen tip. Moving the valve rod to its limits can damage the tip and/or the motor for the pump.

1. Turn on the machine as described in section 1.2.2 Hardware Start.
2. If the Status indicates OFF, click **Fault** Acknowledge, then click **Reset on the Manual Motion Controls**.
3. If the Status indicates HOME, click **HOME** within the gizmo control area. Homing may take 5 to 30 seconds.
4. Set **Valve Speed** between 1 to 4 units/s.
5. Use the **Open** and **Close** buttons to move the valve.

Saving Close position for dispense:

1. Move the valve rod to a position greater than 0 with the **Open** button.
2. Open the **Closed Position Config** tab.
3. Click **Match Current Position.**
4. Click **Save Position**
5. To close the valve later, return to the **Closed Position Config** tab, click **Match Saved Position**, and then click **Goto Position**.

Saving Open position for dispense (Optional):

6. Open the **Open Position Config** tab.
7. Type the desired valve open position (not the valve opening) into the edit box.
8. Click **Save Position**.
9. To open the valve later (for example, to flow some material before beginning), return to the **Open Position Config** tab, click **Match Saved Position**, and then click **Goto Position**.

NOTE: The Closed Position Config tab defines the closed position for printing. When **Save Position** is clicked, the **Current Saved Position** is saved to the machine settings to be recalled during project execution. However, the Open position is not used. The valve commands in the script generated for the print will instead define how much the valve will open when printing. The valve commands specify valve moves relative to the pump's Closed position.

2.5.3 Actuation

On systems with multiple SmartPumps™, it is necessary for the tool plate to lower a pump to "Ready for Use" position. All other pumps not in use are raised. This is called *actuation*. Pumps automatically actuate up and down during a process but always finish in the up position. It may be easier to clean or replace the tip with the pump in the down position. To manually actuate a pump, select the desired pump from the **Gizmo List** and click the **Actuate** button at the bottom of the **gizmo control** area.

2.6 Target Camera

The Target Camera is a down-facing camera which focuses on substrates for post-print inspection and locating fiducial marks to accurately align prints.

Without a project running, the Target Camera may be used for inspection by selecting **MainCamera** from the **Gizmo List**, clicking the **Live View** button, adjusting the Z axis for focusing, and moving the X and Y axes.

If a **FiducialFindJob** is included in the project, this camera's image viewer will be used to locate the fiducial on the surface. See section 6 Fiducials for details on using fiducials.

2.7 Laser Displacement Sensor

The laser displacement sensor is used for conformal (non-flat) printing on curved, uneven, or unlevel surfaces or on surfaces whose height is unknown. While a print project is running, the laser displacement sensor is used in several different ways (see section 7 Conformal Printing).

Under manual control, while a print project is not running, the sensor reading may be viewed in real time. Some systems may have several laser displacement sensors. From the Gizmo List, select **PrimaryLaserSensor** or **SecondaryLaserSensor**. To start reading, observe the **displacement sensor reading** in the selected laser's measurement display. The X, Y, and Z stages may be operated to obtain manual laser measurements across the substrate.

2.8 UV Curing Lamp

The UV curing lamp is used to cure certain photopolymer materials using focused ultraviolet light. While a print project is running, the UV curing lamp is used by **PrintScriptJobs** (see section 5). Under manual control, while a print project is not running, the lamp may be used by clicking the **Enable** checkbox, adjusting the **intensity** slider and operating the X, Y, and Z axes.

2.9 Laser

The laser is used to cure certain thermally-sensitive or evaporatively-cured materials using focused infrared laser light.

WARNING: This laser is an extremely dangerous accessory and should **ONLY** be operated with all safeguards in place, including (at a minimum) IR filtering window glass and automatic shutdown switches when the doors are opened.

While a print project is running, the laser is used by **PrintScriptJobs** (see section 5). Under manual control, while a print project is not running, the laser may be used by clicking the **Enable** checkbox, adjusting the intensity slider, and operating the X, Y, and Z stages.

WARNING: Do not leave the laser focused on one spot for too long, as this will start a fire.

2.10 Process View Camera Trolley

A trolley (motor stage) is provided to move the process view camera left and right on quad-head tool plates to observe each pump on the monitor separately. This trolley may be operated from each SmartPump™ gizmo control by clicking **Move Process View Trolley Here** or by using the dedicated Process View Camera Trolley Gizmo tab.

In the Gizmo tab, the camera may be commanded to any of the pump positions, and the view of the tip may be observed on the monitor.

2.11 Vision Pit Elevator

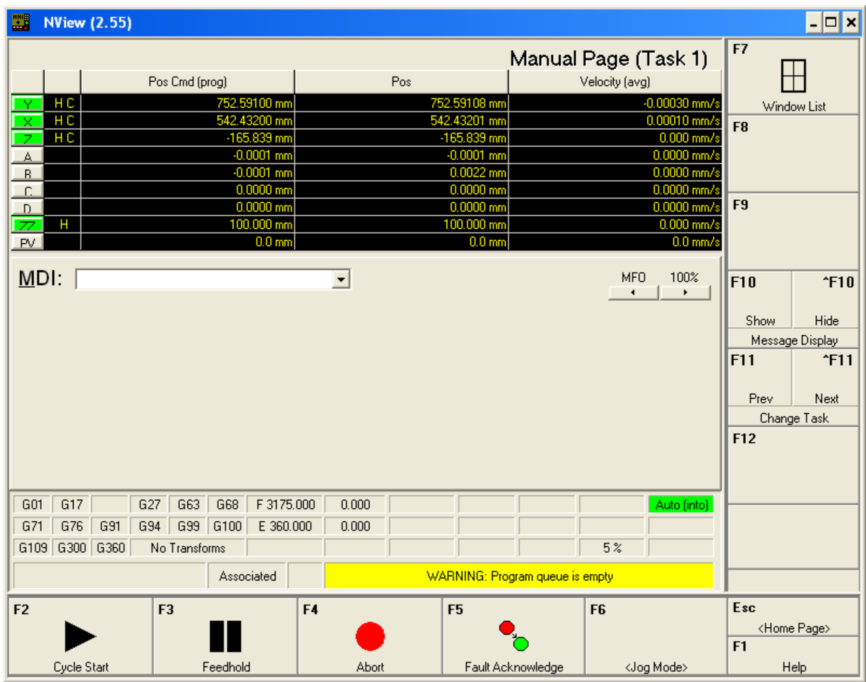
1) Open the **NView** software

2) Select **Manual Mode**

3) Go to **MDI Mode (F6)** if **Jog Mode (F6)** is currently selected. The window should appear as shown below.

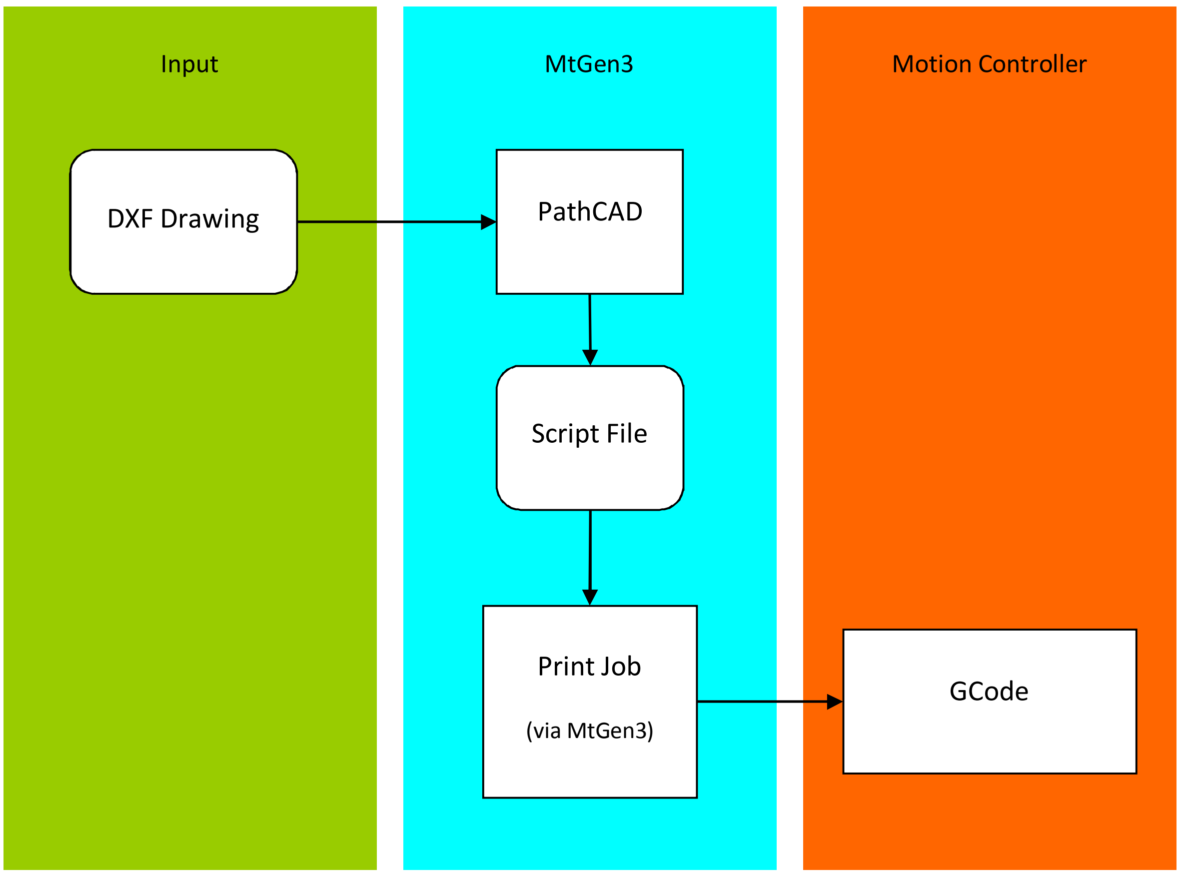
4) In the **MDI** command area, enter the two commands to set the vision pit to the calibration position.

```
HOME ZZ
G1 ZZ 100
```



3 Printing

Printing instruction files (script files) typically start out as a line drawing in a DXF file. DXF files may be edited in or exported from many different CAD programs such as DraftSight, SolidWorks, and Cadence. PathCAD (PCAD) is provided by nScrypt to allow ASCII-type DXF files to be imported, edited, and converted to textual script files. PCAD may also be used to make new drawings. In either case, PCAD is used to convert drawings to textual script files. These script files are loaded into MTGen3 via **print jobs** and executed on the machine within the **project** framework. A typical file conversion flow is shown below.



Please see 12.2 Appendix 2: Software for an overview of software which may be useful for obtaining script files.

3.1 A Word on Nomenclature

Line width	Narrow / Wide	XY width of lines
Line thickness	Thick / Thin	Z height of line off deck

3.2 Parameters

<table>
<tr><th>Parameter</th><th>Command</th><th>Adjustable</th></tr>
<tr><td>Speed</td><td><code>speed</code></td><td>In the script</td></tr>
<tr><td colspan="3">Higher speed narrows lines but may cause material to lose adhesion with the surface and cause blotching. Blotching may be reduced by reducing the dispense gap. Units: mm/s.</td></tr>
<tr><td>Dispense gap</td><td>none</td><td>In the print job</td></tr>
<tr><td colspan="3">Reducing the dispense gap produces cleaner, straighter lines and sharper corners but widens lines. Printing too close to the substrate may cause crashes if the surface is not flat enough. Increasing dispense gap produces thicker lines but may cause blotching. Units: mm.</td></tr>
<tr><td>Pressure</td><td><code>pressure</code></td><td>In the script</td></tr>
<tr><td colspan="3">Lower pressure allows narrower lines but may result in blotching at higher speeds. Higher pressure increases reliability but also increases line width and thickness however this may be mitigated with higher speed. Higher pressure may reduce drying and clogging in volatile materials. Units: psi.</td></tr>
<tr><td>Valve opening</td><td><code>trigvalverel</code></td><td>In the script</td></tr>
<tr><td colspan="3">Valve opening operates similarly to pressure. Set the full open pressure first then use the valve opening to tune the precise flow rate. Units: mm.</td></tr>
<tr><td>Open position</td><td colspan="2">Use the valve opening plus closed position or test using gizmo control</td></tr>
<tr><td colspan="3">The valve position where the desired flow rate is achieved. This is used to benchmark the valve opening by subtracting the open position from the closed position to obtain the open position.</td></tr>
<tr><td>Opening speed</td><td><code>trigvalverel</code></td><td>In the script</td></tr>
<tr><td colspan="3">Slower opening speeds reduce line-start blotching but may take a long time to begin flowing. Quicker opening speeds mitigate this but may cause spattering. Units: mm/s.</td></tr>
<tr><td>Trigger Wait (Trig. Wait)</td><td><code>trigwait</code></td><td>In the script</td></tr>
<tr><td colspan="3">Delays the XY motion to allow the material to begin flowing out the tip. Used after <code>trigvalverel</code> command. Units: seconds.</td></tr>
<tr><td>Closed position</td><td>none</td><td>In the gizmo control</td></tr>
<tr><td colspan="3">This should be set as close to the boundary between on and off as possible. Closing too far will introduce inaccuracies in the valve opening. Additionally, air may be sucked back into the pen tip causing blotching upon opening. Units: mm.</td></tr>
<tr><td>Closing speed</td><td><code>valverel</code></td><td>In the script</td></tr>
<tr><td colspan="3">Slower closing speeds reduce air suck back and faster closing speeds reduce end of line blotching. Units: mm/s.</td></tr>
</table>

3.3 Scripts

A script is a text file containing a list of commands which is interpreted and converted into GCode at run time. A script file contains path information for motion and various other commands to control the machine automatically.

Scripts may either be exported from PathCAD or written manually depending on the complexity of the script. When exporting from PathCAD scripts will still need to be edited, adjusting certain commands during the development process.

A sample script is provided here. It may be used as a template for manually writing scripts for the machine. Each command is described to give a starting point for tuning up basic printing parameters.

Line	Command	Comment
1.	`relative`	XYZ commands are relative.
2.	`speed 5`	Must specify speed first.
3.	`move 0 0 10`	**Safe Move Pick-up.** May be used to move to start location if not starting print line at the origin.
4.	`move 0 0 -10`	**Safe Move Put-down**. Move down to substrate.
5.	`speed 10`	Print speed.
6.	`trigvalverel 0.55 8`	**Valve opening**: 0.55 mm, Open Speed: 8 mm/s
7.	`trigwait 0.12`	Delays motion to allow material to begin exiting tip.
8.	`move 10 0 0`	Prints a line 10 mm long in the X direction.
9.	`valverel 0 8`	**Valve close**: Closes the valve at 8 mm/s. The value 0 indicates no offset from the close position and the valve will fully close.
10.	`speed 50`	**Fly speed**. The line transfer speed.
11.	`move 0 0 10`	**Pick-up**. Machine moves up by 10 mm.
12.	`move 0 1 0`	Fly. Machine moves by 1 mm in the Y direction.
13.	`move 0 0 -10`	**Put down**. Machine moves down by 10 mm to the surface.
14.	`speed 10`	*Beginning of next print block. Repeat these commands to create a pattern of lines.*
15.	`trigvalverel 0.55 8`	
16.	`trigwait 0.12`	
17.	`move 10 0 0`	
18.	`valverel 0 8`	

The relative command at the beginning of a script is not required, but it provides clarity. Motion commands in a script are all relative to a given starting position by default. Each move is cumulative. This means each move is pre-calculated to an absolute coordinate by adding the move to the previous calculated position during the script's interpretation.

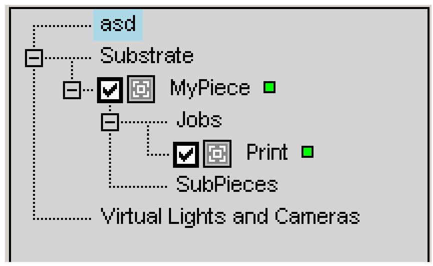
NOTE: Absolute coordinates can be specified in a script with the absolute command. A script may toggle between the two modes of positioning as needed. All move commands that follow will be calculated appropriately based on the last positioning mode specified. **Absolute positioning bypasses the automatic positioning calculations and should be used carefully.**

Safe Move Pick-up and Put-down.

During the initial positioning, safe moves are used to determine where to begin from the starting position. The safe move is also used to position the machine near the substrate for verification and preparing initial commands for the system to process before continuing the rest of the script. **These commands should not be omitted when using relative scripts and are not generated by PathCAD.** The Safe Move Pick-up will stop the specified device a given distance from the substrate based on the Z parameter of the command. The safe move Put-down will bring the device to the substrate based on the Z parameter as well. The Z component for the Put-down command should always be the inverse of the Pick-up.

4 The Project

A project is a unified repository of all pieces and jobs organized into a hierarchy to define a specific application. The project is displayed as a tree structure for ease of management. Projects may be saved and reloaded. They may be used to document progress. Below is a sample project tree view consisting of a **Substrate** definition, one **Work Piece** called "MyPiece," and one **Print Script Job** called "Print".

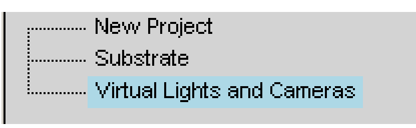


4.1 Categories

Project Title – the name of the project is defined on creation. The title of the project also denotes the project directory from which this project will access information.

Substrate – The substrate is defined by **Work Pieces** and their subcomponents. Together, these define how the machine will perform its user defined tasks. For more information, see section 4.3 Introduction to Work Pieces.

Virtual World – The virtual world is a set of controls that modify the appearance and settings within the **3D world display**. *Note: This section of the application is currently under development.*



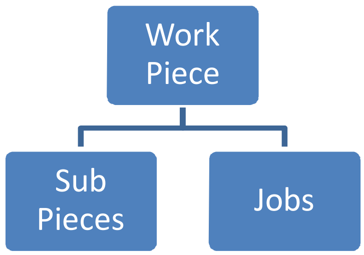
Project Title: "New Project"
Substrate Name: "Substrate"
Virtual World Text: "Virtual Lights and Cameras"

4.2 Elements within a Project

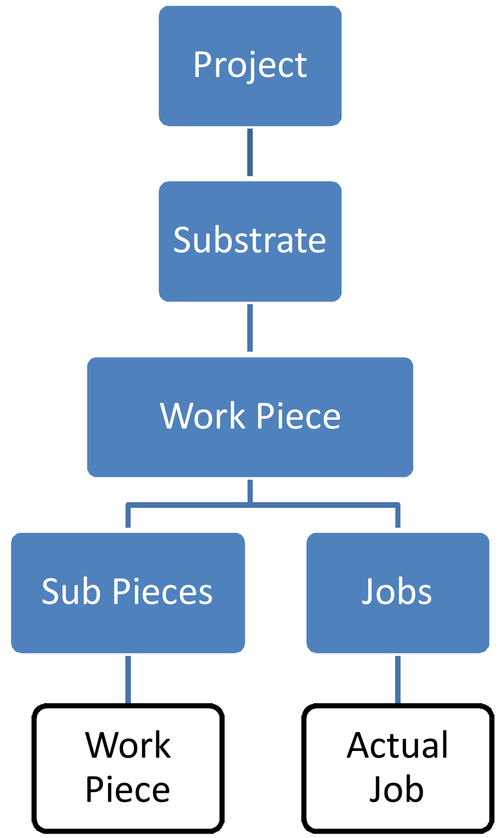
The architecture of the project is formed as the substrate is defined by creating a hierarchy of containers called WorkPieces. Processes called Jobs are attached to the WorkPiece where that job should be performed. WorkPieces attached to WorkPieces are referred to as SubPieces.

The top of the project tree is always a substrate definition with one or more work pieces.

4.2.1 Fundamental Building Block



4.2.2 Example Hierarchy



4.3 Introduction to Work Pieces

A **WorkPiece** (**WKP**) is an area in which the machine will perform **Jobs**. The WorkPiece is defined by the dimensions and positioning of a physical substrate or part placed within the machine. A WorkPiece at the top of the hierarchy contains its position and orientation information in a local matrix relative to the machine's **world coordinate system**. Jobs and SubPieces are maintained and processed relative to their parent WorkPiece.

As a project is executed, a process list is formed. This list contains WorkPieces which are ready to be processed. All WorkPieces attached to the Substrate branch of the project are added to the list first. As they are processed their children are added. For a group of SubPieces to be added, all jobs must be completed without error for the current WorkPiece. If one job fails, the current WorkPiece is failed and updated in the project tree with a red status icon next to its name. When a WorkPiece fails, none of its SubPieces are added, and the next piece in the list is processed.

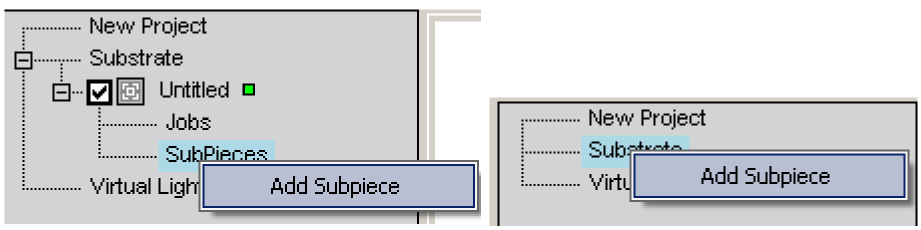
4.3.1 Work Piece Group

A **Work Piece Group** (**WPG**) is a wizard-driven template used to generate an array of SubPieces. This is useful when a grid of identical patterns must be printed throughout the substrate. Please see section 4.5 Creating Work Piece Groups below for details on creating and using work piece groups.

4.3.2 Creating a Work Piece

To add a work piece:

1) Right-click the **Substrate** node within the project or the **SubPieces** node under a work piece to activate a drop down menu. You may need to use the [+] expander button to reveal the SubPieces node for a work piece.



2) Select **Add SubPiece** to launch the **Work Piece Creator**.

3) Select **WKP** for work piece**.**

4) Choose the creation options from the check boxes below the job selection area. To get started, ensure **Load from File** is unchecked.

5) Click **Next >>**.

6) Selecting **Open editor when finished** will launch the job editor when you click the **Finished** button.

7) Click Finished.

<u>WorkPiece and Job Creation Options</u>

Load from File

The **Load from File** option allows data to be loaded from a previously created Job or WorkPiece file. Templates can be created by saving a copy of a WorkPiece or job file to another folder. The default template directory for WorkPieces is the Substrates folder. For Jobs, the BoardFiles directory is the default template directory. When "Load from File" is checked, the creator will open the default template directory to find a file to use. The file selected is copied and placed into the project's scripts directory.

File Referencing

Sometimes multiple components all need the same data and need to be treated the same. If the selected file already exists in the scripts directory of the project, the file is referenced and loaded from multiple components within the project. This means that changing one Job or WorkPiece changes all other jobs or WorkPieces referencing the same file. For Jobs, this affects all editable parameters except the display name in the project tree. However, for WorkPieces changes in the referenced files only affect the dimensions, not the local position or display name in the project tree.

Save as Separate File

When using Load from File, the file name stays the same as the file copied. To avoid referencing, the selected file will be copied to the scripts directory with a new name.

4.3.3 Teach Origin

To position objects throughout the world coordinate system, the lower left corner (or Origin) of the WorkPieces at the top of the hierarchy needs to be taught. In a hierarchy, all SubPieces should maintain their fixed offsets based on schematics. Positional error and inaccuracies may be introduced by trying to teach origin on SubPieces. Therefore, teaching origin on a SubPiece is not recommended. The position of the WorkPiece is calculated based on the calibration of the current device selected in the gizmo selection.

To teach origin:

1) Select a device in the gizmo selection dropdown list.
2) Move that device to the lower left corner of the WorkPiece.
3) Right click the **WorkPiece** in the project tree.
4) Click **Teach Origin**.

4.4 Understanding Positioning

4.4.1 Positioning Work Pieces

The work piece is positioned by the center of its dimensions. The WorkPiece is then moved by its translation parameter into position. The origin of a WorkPiece is calculated to find the top, lower left corner of the area defined by the dimensions. All SubPieces are translated with an offset from the parent WorkPiece's calculated origin. Upon creation, a SubPiece's starting position is centered on the calculated origin of its parent WorkPiece. This structure of dependency allows easy placement of multiple sub pieces throughout the substrate with little management.

To define the substrate when positioning WorkPieces, the software assumes the initial measurements are perfect with no rotation accommodations. This is why it is beneficial to enter accurate measurements from a drawing or schematic.

Important: When adding a sub piece to a work piece, the *center* of the child work piece is located at the *origin* of the parent work piece. The child can then be placed into the parent using its local offsets. The position of a subpiece is not confined to the dimensions of its parent's WorkPiece.

Setting the length and width dimensions of a work piece to zero defines a single point. This is done when no rotation adjustment is required. WorkPiece heights accumulate through the hierarchy. Each WorkPiece is placed on the floor. Its height is the distance from the top of the WorkPiece to the floor. A SubPiece's height is the distance from the top of the SubPiece to the top of its parent WorkPiece.

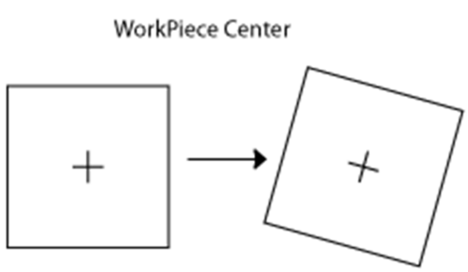
4.4.1.a Positioning Jobs

The positioning of a job is based on the work piece origin. Some jobs have offset values which are used to translate the job's starting position from the WorkPiece origin. This allows positional tweaking while the main definition of the substrate is maintained through the hierarchy of WorkPieces.

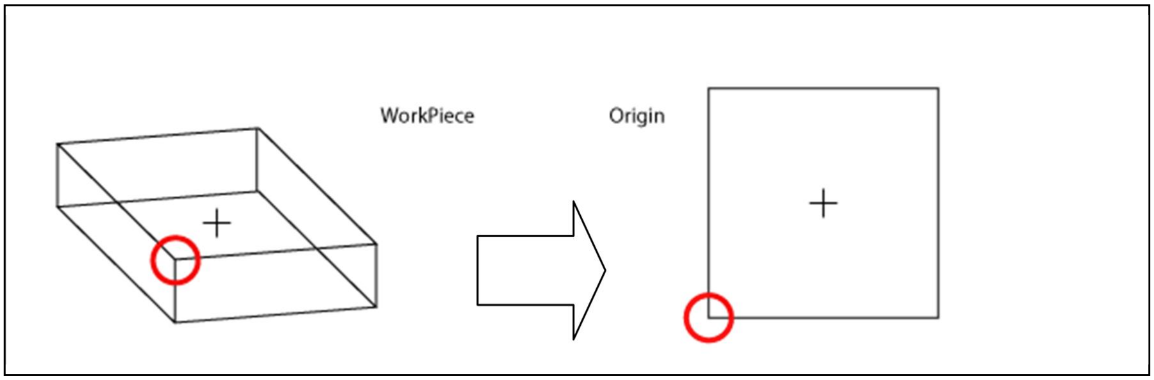
Important: When adding a job to a work piece, the *origin* of the script file is located at the *origin* of the parent work piece. Script printing paths need not exist within the dimensions of the WorkPiece.

4.4.2 Rotation and Translation Offsets

Work pieces are positioned and rotated about their center.



All sub-components of a work piece are positioned relative to their parent work piece's origin. This means that the origin of the parent work piece becomes the (0, 0) local coordinate for all attachments. Visually, the work piece origin is always the lower left corner at the top of the work piece. The top surface is determined by the user-defined height.



4.5 Creating Work Piece Groups

A **Work Piece Group** (**WPG**) automatically creates a WorkPiece containing multiple SubPieces in an array. This saves time when creating a grid of identical areas. Creating a Work Piece Group requires a premade WorkPiece or template for duplication.

1. Create a regular work piece (call this the source work piece) within the project as it needs to be configured. This will be used by the group for duplication. Once the work piece group has been created, the regular work piece may be deleted from its original location in the project.

2. Add a **work piece group** to the project by right-clicking the **Substrate** or **SubPieces** in the project tree.

3. Select **WorkPiecegroup** from the WorkPiece creator.

4. Ensure “Load From File” is unchecked and click **Next**.

5. Name the parent WorkPiece file and click **Save**.

6. Enter the Dimensions of the Parent WorkPiece. (This step is optional and can be edited again later in the wizard.)

7. Click apply changes and close the WorkPiece editor.

8. At the **Work Piece Grid Creation Page**, use the file path selector to select the template WorkPiece which is to be used for duplication.

9. Enter the number rows in the X direction.

10. Enter the number columns in the Y direction.

11. Enter the spacing between work piece centers into **Centroid Spacing X and Y** by adding the dimensions of the template with the desired spacing between SubPieces. (For Example: A template WorkPiece’s dimensions specify 20 mm in width and 20 mm length. The desired gap between SubPieces is 3 mm in X and Y. Centroid Spacing X and Y are set to the value of 23.)

12. Use the Parent WorkPiece button or the Child WorkPiece button to open the WorkPiece editor for adjustments to the dimensions. Editing translation information is ignored for now as the local offsets are calculated by the wizard.

13. The starting X and Y offsets are used to adjust the position of the entire array based on the Parent WorkPiece's origin. Use the display to verify the resulting placement.

14. Click **Next**.

15. Uncheck "Open WorkPiece Editor" and click **Finished**.

16. The project tree should now have a WorkPiece with a list of subpieces named by their row and column location.

4.6 Introduction to Jobs

A **Job** is an editable automated process set to perform a specific task. A job sometimes modifies and adjusts its parent WorkPiece, collects information about the substrate, or executes a script. The order in which jobs need to be added to each work piece depends on the substrate and desired results.

All jobs are processed in the order they appear in the hierarchy from top to bottom. A job must complete its function without failure in order for the next job to be processed. Job failure results in the failure of its WorkPiece.

Jobs can be edited while the project is being executed but will not affect the currently running job. Once a job has started, it will run with the current settings it found in its reference file before execution. Jobs are forced to reload their referenced file to obtain any adjustments made by the user.

Generic instructions for creating jobs can be found in section 4.6.2 Creating a Job. Each of the following sections of this document discusses job-type specifics.

4.6.1 Caution against External Editing

WARNING: Project editing should be done within the software's user interface. Although it is not recommended, advanced users may wish to edit the project files using an XML or text editor. This is not advised because *incorrect XML syntax may cause the project to become corrupt and unloadable.*

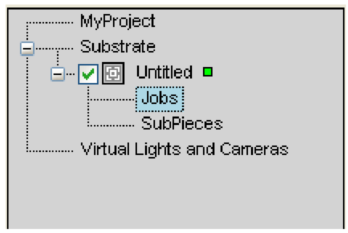
All jobs and work pieces referenced within the project must be located in the project's scripts directory. This directory is automatically generated when the project is created. The user may manually modify a project's XML file as long as the XML syntax remains intact. A project can only be modified if it is saved on the same drive the application is saved on. If a project is saved on a networked drive, it must be copied to the same drive the application is saved on before modifying. *Failure to do this may cause issues with loading or modifying a project.* The Projects folder in the application directory is the default directory which projects save to.

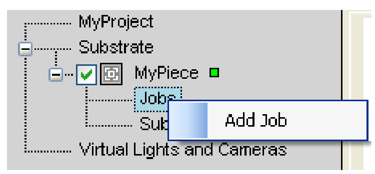
WARNING: Defining inaccurate measurements in either jobs or work pieces creates varying degrees of positional error. Always use accurate measurements to prevent collision with the physical substrate or the floor of the machine, which will result in severe and costly equipment damage.

4.6.2 Creating a Job

1) Expand a WorkPiece to reveal the Jobs branch in the project tree.

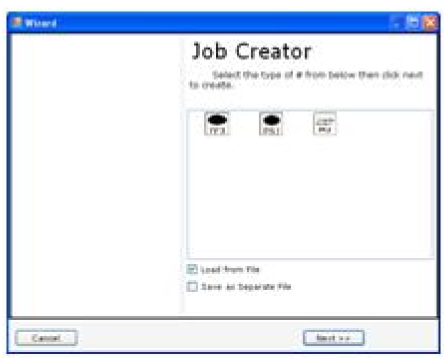
2) Right-click on the **Jobs** branch. Select **Add Job** from the pop-up menu to activate the **Job Creator Wizard**.





3) Click one of the job icons to select an MPJ (manual print job) and ensure **Load from File** is unchecked.

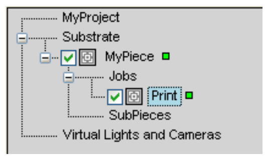
4) Click **Next** to create the job.



5) Check the Open Job editor when finished to customize the job parameters once the **Finished** button is clicked. (For editing see the Editing a Job section.)

6) Apply changes to the job and close the Job editor window.

7) The created job appears under the **Jobs** branch of the work piece once it has been added to the project tree.



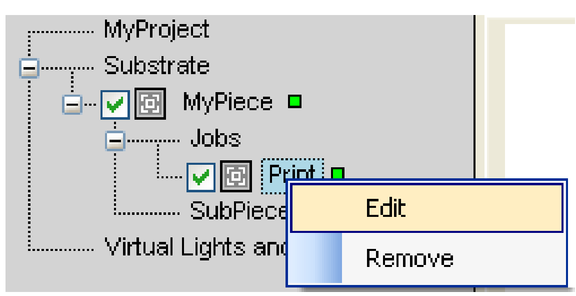
4.6.3 Editing a Job

Each job has its own set of parameters which may be modified to define its process. Parameters are used to specify how the job should be performed to fit the needs of the application. Details of job features are given in the following sections of this manual.

The job editor can be accessed in two ways. When creating a new job, the Job Creation Wizard's final page has a check box titled **Open Job editor when finished**. This opens the job editor automatically once the **Finished** button is clicked.

If the job has already exists in the project, the job editor can be opened by right clicking on the job and choosing **Edit** from the job menu as shown below.

1) Right click on a Job in the Project View to activate the Job Menu.
2) Click **Edit** to launch the job editor.



Job Editor Rules:

- Changes to a job are not accepted until **Apply Changes** is clicked.
- If the editor window is closed before the changes are applied, the changes will be lost.
- Applying changes to a job that is referenced multiple times in the project or from several projects will cause all the referenced jobs to use the modified file.
- Some job types (PSJ, FFJ, etc.) can use library files. To modify or save a library file, use the top section of the job editor titled **Library File Select**.

4.6.4 Library File Select

Library files allow frequently used files to be saved to a common location as templates where multiple projects may refer to them. The section at the top of the job editor allows the user to save, update, or reload a library file into the job.

Option	**Function**
Template File Path Selector	Displays the current selected template file used for this Job.
Reload	Updates the current Job data based on the template file. All data from the template file is copied. The Job data is overwritten.
Save	Saves the current data to the template file.
Save As	Saves a new template with the current data from the Job.

4.6.5 Available Jobs

Acronym	Title	Relevant Section
MPJ	Manual print job	5.1 Manual Print Jobs
PSJ	Print script job	0 Print Script Jobs
APJ	Automatic print job	5.4 Automatic Print Job
CPJ	Custom Print Job	5.5 Custom Print Jobs
FFJ	Fiducial find job	6 Fiducials
PCJ	Planar correction job / Four point correction	7.3 Planar Correction
PMJ	Path mapping job	7.5 Path Mapping
GMJ	Grid mapping job	7.6 Grid Mapping
RSJ	Rotational script job	8 Rotary Stages
CPJ	Custom print job	5.5 Custom Print Jobs
APJ	Auto print job	5.4 Automatic Print Job

4.7 Job Position and Orientation and Scripts

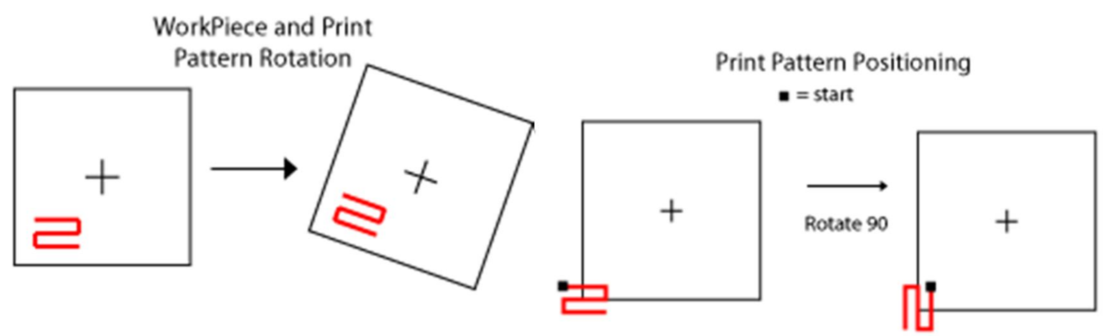
A Job's starting position is located at the origin of its parent WorkPiece by default. Local X and Y offsets in specific jobs are used to change the start position relative to the WorkPiece's origin. For Jobs with scripts, the relative coordinates generated by a script are also positioned based on the WorkPiece's origin.

NOTE: The origin of the script when it is exported determines the script's initial position about the WorkPiece origin without adjustment. For example, a drawing where origin is the center of the pattern will center the script about the WorkPiece's origin as shown in section 4.7.1.

4.7.1 Rotation

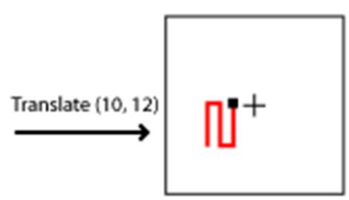
Orientation information is used to calculate rotation of a WorkPiece. In the hierarchy, a WorkPiece's rotation and translation changes the position of its origin. A Job's positioning and orientation is local to its parent WorkPiece; therefore adjustments are calculated automatically.

Jobs with scripts may have orientation information to rotate only the script coordinates. In this case, the scripts's centroid is calculated and is used to manipulate the points within the scripts own coordinate system. Afterwards, the script is then modified by the WorkPiece's orientation information for correct positioning.



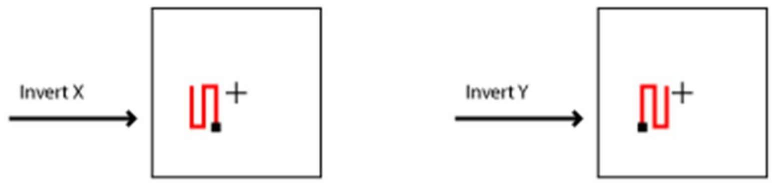
4.7.2 Translation

The calculated centroid of the script is used to translate all of the script points into position based on a starting XY offset.



4.7.3 Inversion (Flipping Side–For–Side)

The calculated centroid of the script is used to flip the X and Y coordinates of the script. The Z component of each script position remains the same.



4.8 Project Trouble Shooting

4.8.1 Offset

Most positioning problems occur due to improper offsets or failure to teach the origin of a WorkPiece with a calibrated device. If a job begins sending a device to an unexpected location, the best response is to press the **Stop** button in the software. For emergencies, pressing the **E-Stop** will also stop the axes' current motion, but no cleanup is performed as the motors can no longer be controlled. In many cases the machine will not move at all and fail a job because the next intended location was calculated as being out-of-bounds. This is referred to by the motion controller as a Soft Limit Error. The soft limits prevent any axis from moving to a location the system already knows it cannot reach before motion occurs. The same occurs if any of the points in the script file are translated as out-of-bounds of the machine limits. The fault will occur before the script begins running. Once the machine is stopped, the project should be checked for any positional inaccuracies before contacting nScrypt support.

4.8.2 Crashes and Dispense Gap

Specifying a proper dispense gap is important. A dispense gap of zero tells the Print Pattern to occur on the calculated surface of the substrate. This surface height is based on the height of the parent work piece. A negative dispense gap tells the Print Pattern to occur below the calculated surface. *This is not recommended, as it can cause crashes.*

If the desired height cannot be reached or if the specified dispense gap appears incorrect on the camera display, check the height of the parent work piece or try recalibrating the machine's devices. Make sure to redefine the floor as well.

4.8.3 Height Correction

If the height of the substrate varies, try assigning a **planar correction job** or **grid map job** just prior to the Print Script Job in the process tree. Some surfaces may be less level or flat than they appear.

One way to define the height of the substrate is by using the laser sensor. Bring the laser sensor to the floor and zero where the laser read is zero. Record the current Z position. Next raise the laser onto the top of the substrate and make the laser read zero. Subtract floor value from the substrate value to find the height of the substrate.

Substrate height = Substrate position Z - Floor position Z

Other methods of determining height, such as using external tools like calipers or a ruler, are also good methods. The more accurate the specified height is, the more accurate the dispense gap will be. Having an estimated height and using the Single Point Correction option when using a print script job will also work.

When using a grid map, all points in the script file must be contained within the scanned area. Scanning an area too small or adding rotation in the print job causes the print job to abort just before it runs in the process tree.

Calipers or fine rulers may be used to determine the substrate thickness. The more accurate the specified thickness is, the more accurate the dispense gap will be. For example, if a dispense gap of 50 +/- 10 µm is needed, the substrate thickness must be measured to within 10 µm minus any other sources of error (such as calibration errors).

5 Print Jobs

A **Print Job** is a process which converts a script file into a tool path. The device commands in the script are converted based on the device held in the gizmo slot targeted by this Job. Only one gizmo slot and one script file may be used per print job.

The Print Job loads the script at the time of execution and uses the hierarchy to position and orient the pattern. The original script remains unmodified, allowing a single pattern attached to multiple print jobs to be performed in different places on the substrate.

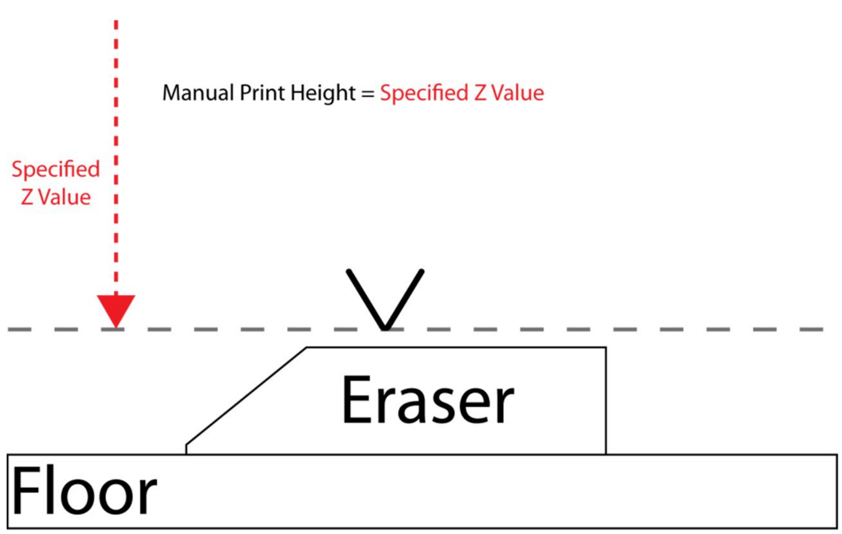
There are four types of print jobs: Manual Print Jobs (**MPJs**), Print Script Jobs (**PSJs**), Automatic Print Jobs (**APJs**), and Custom Print Jobs. Manual Print Jobs execute a target script with an absolute Z position. No height adjustments or calculations are used. Print Script Jobs use various height calculations to adjust a target script. Automatic Print Jobs use various forms of collected data to generate a script file to perform. Custom Print Jobs work like Print Script Jobs but support the use of plugins.

5.1 Manual Print Jobs

Manual print jobs (MPJs) are designed for quick and simple printing when height information is not needed. Manual Print Jobs print at the user-defined Z position specified in the MPJ job editor.

User-Defined Height in Job Editor

Manual Print Jobs do not support conformal printing or height adjustments. The distance from the top of the substrate should be calculated with a gap in mind. Normal practice for this job is to carefully move the target device to the top of the substrate. Record the Z position and save that as the Z position for the Manual Print Job. (For dispensing, move the Z axis up by a desired offset, which should be the same distance as the dispense gap.)



5.2 Print Script Jobs

Print Script Jobs (**PSJs**) are the most common job for dispensing. Print scripts apply a number of calculations to the script file to determine final positioning and height calculations. Conformal printing is achieved when a print script uses a grid map to adjust the heights of all the script points. For simpler height adjustments, the Print Script Job uses the laser as an initial safety test to determine and adjust the height of its parent WorkPiece before the script is processed. When performing a Single Point Correction, the Primary Laser on the system is always used.

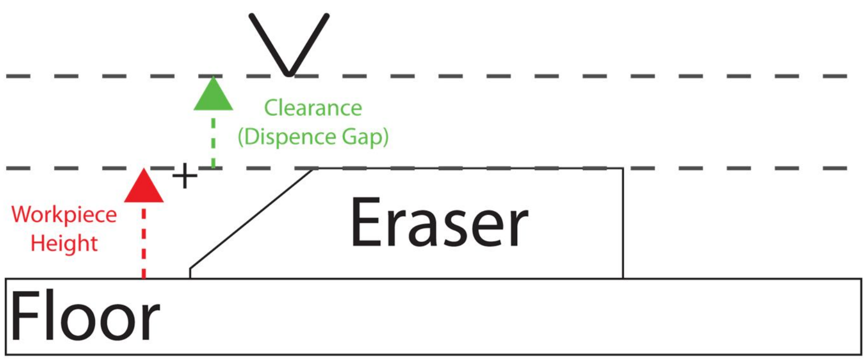
5.2.1 Case 1 – Standard Mode

Print script jobs print at a height which is computed as follows:

Machine Floor + Total Height + Dispense Gap

This print height computation may be visualized as shown in the illustration below.

Print Height = Floor + Workpiece Height + Clearance (Dispence Gap)



5.2.2 Case 2 – Laser Displacement Height Testing

If Single Point Correction is used, the print script job <u>samples</u> at a height which is computed as follows:

Machine Floor + Total Height

and <u>prints</u> at a height which is computed as follows:

Machine Floor + Total Height + Displacement Laser Reading + Dispense Gap

5.2.3 Case 3 – Laser Displacement Scanning

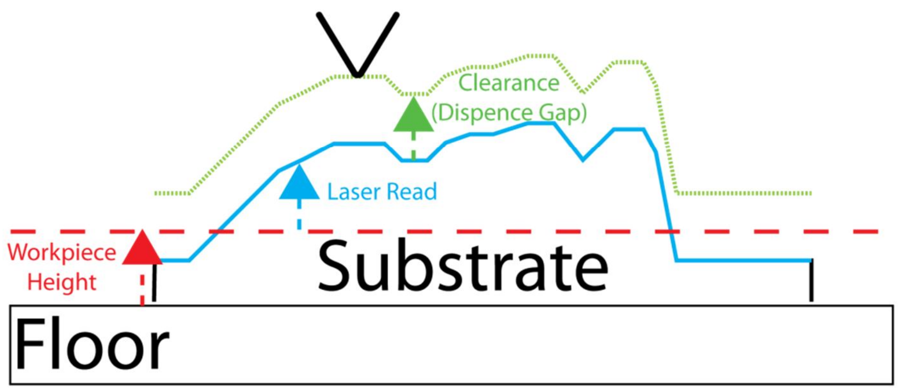
If a grid map or four-point correction are used before the print job, the print script job prints at a height which is computed as follows:

Machine Floor + Total Height + Each Script Point Z + Projected Collision Height at Script Point XY + Dispense Gap

NOTE: Calibration is critical for print script jobs. It is calibration which allows the laser sensor to be used to take surface measurements and then follow with SmartPump™ or other gizmo at the correct height.

The illustration below shows how laser displacement scanning operates.

Line Scan Print Height = Floor + Workpiece Height + Laser Read + Clearance (Dispence Gap)



5.3 Editing a Print Script Job

Parameter	Description
Print File Path	Selects which script file to use.
Gizmo Slot To Use	Selects which gizmo to use. The offset from calibration is used to position the correct gizmo. This is why calibration is critical.
Perform Height Test	This optional parameter will use the primary laser to sample the actual height of the substrate. The sample XY location is always at the origin + the starting XY offset of the print script job. (See section 7.2 Single Point Correction for more information.)
Job Name	This is the name which appears in the project tree display.
Clearance / Dispense Gap	This value is added to the height of the work piece to define how far away from the substrate the pattern should be performed.
Print Resolution	During grid map jobs, the print pattern is broken into a set of line segments based on the resolution. (See section 7.6 Grid Mapping for more information.)
Rotation	Specifies how many degrees to rotate the pattern by. The rotation is performed about the calculated centroid of the pattern.
Invert X	Flips the pattern 180 degrees along the Y-axis about the calculated centroid of the pattern.
Invert Y	Flips the pattern 180 degrees along the X-axis about the calculated centroid of the pattern.
Start Offset X	Translates the center position of the pattern from the work piece origin along the X-axis.
Start Offset Y	Translates the center position of the pattern from the work piece origin along the Y-axis.

To finalize changes when using the editor, click **Apply Changes**.

5.3.1 Clickable Fields in Print Jobs

Clicking the modification area for the **EditScript** parameter reveals the browser button. Clicking the browser button opens the script in the default text editor.

Clicking the modification area for the **PrintFilePath** parameter reveals the browser button. Clicking the browser button launches the file browser. Select the script file to assign to the print job.

To finalize changes when using the editor, click **Apply Changes**.

5.4 Automatic Print Job

An Automatic Print Job (APJ) uses Print Link Enabled Jobs to print basic geometry based on various forms of data collected from the substrate. Path and positioning information is retrieved to automatically generate as script with the printing parameters specified in this Job.

5.5 Custom Print Jobs

Custom Print Jobs (**CPJs**) are a modified version of a PSJ which support outside adjustments to data via plug-ins. A plug-in is created by the user adhering to a specific interface and selected to use for this job. Processing data is passed to the plug-in for modification and then passed back to this job for further processing.

See Plug-ins section for details on supported functionality.

6 Fiducials

A Fiducial is any repeatable point on a substrate used to adjust position and orientation information. This adjustment is made based on where the fiducial was expected to be versus where the fiducial was actually found. The difference is calculated to produce an adjustment rotation and translation for a WorkPiece.

Measurements from fiducials provide the best results for positioning. Drawings are often based on fiducial marks; offset information from fiducials can often be translated directly to the jobs and scripts of the project.

A fiducial is always found at an expected offset from the WorkPiece origin. The software uses the Target Camera calibration to position itself over the expected position and take a picture. The image is then processed to find the exact position of the fiducial. If found, the fiducial's offset from the center of the image is used to calculate the final position of the fiducial.

As part of automation, fiducial recognition is processed as pass, fail, or user corrected. A vision process is used to determine if the fiducial can be recognized. The vision processes used for specific fiducials are tailored in a vision process outside of MtGen3.

6.1 Correction Method

A vector is computed between the expected coordinates of two fiducials. A second vector is computed between the actual coordinates of the fiducials. The offset between the centers of each vector is found and added to the WorkPiece's translation. The rotation angle is then computed as the angle between these two vectors after translating the expected vector to the actual vector. The resulting angle is applied as a rotation to the current WorkPiece. **Fiducials are not used for scaling**.

6.2 Introduction to Automated Fiducial Find Modes

Three fiducial find modes are available to determine how to respond to fiducial recognition during automation. The modes of response are listed below:

1) **Full inspection** mode is for processes which require verification by the user regardless of whether the fiducial is found or not. Settings file keyword: **FullInspection**.

2) **Fault-only** mode allows user correction only when a fiducial fails to be recognized. **This is the default mode for all new installations.** Settings file keyword: **FaultOnly**.

3) **No-inspection** mode eliminates the ability for a user to correct or verify the fiducial position. Settings file keyword: **NoInspection**.

During modes which allow user correction, the processing of a project waits for user feedback to continue. An image is displayed in the manual controls for the Target Camera found under the Gizmo Select menu. A fiducial is corrected by clicking on the image display where the center of the fiducial is located and clicking **Submit Correctio**n. This will automatically pass the fiducial with the new adjustment. Clicking the **Continue** button will run the current state of the fiducial find process without correction.

If a fiducial is not located within the bounds of the image, the Target Camera can be switched to continuous mode by pressing the **Live Image** button. The image will update automatically as the axes are moved. When the fiducial is located, the user can click on the center of the image and the current machine position with the offset to the center of the image will be used to calculate the adjustment.

NOTE: If a fiducial is not found within the bounds of the image, this is a good indication there may be positional error defined within the project.

Modifying the inspection mode to be used:

1) Navigate to the Application Directory.

2) Open the ApplicationSettings.xml file in the Config directory.

3) Enter the keyword between the inspection mode tags:
<FiducialFindInspectionMode>**(add keyword here)**</ FiducialFindInspectionMode>.

6.3 Using Fiducial Find Jobs

Before a fiducial find job may be used, a Cognex Vision Pro Persistence file (*.vpp) containing at least one cog job must be created and targeted in the machine settings. The first cogjob in this file is always used to define which camera on the system is the Target Camera when running in application mode. If this is not set, the manual controls of the Target Camera will not provide an image. All of the cogjobs created in the vpp file are selectable within the fiducial find job. This allows support for various styles of fiducials within and between projects.

Automatic fiducial detection requires the tuning of a cogjob to recognize a specific fiducial. User correction can override the need for tuning fiducial recognition. Please refer to section 9.5 Machine Vision Setup for Cognex setup basics.

Creating a Fiducial Find Job:

1) Add a fiducial find job to the project.
2) Open the job editor upon creation or from the project tree.
3) Enter the job name. This is the name which appears in the project tree display.
4) Expand the Fiducials section [+]. [0] is the first fiducial and [1] is the second.
5) Enter the local offsets. These offsets are the XY positions from the WorkPiece origin.
6) Select the desired CogJob from VisionProcessName dropdown. The names in this list are the same as the CogJob names in the Fiducial Find vpp file targeted in the machine settings.
7) Click the **Apply Changes**.
8) Save the project.

Submitting a user correction:

1) Select the Main Camera from the gizmo selection.
2) Ensure the axes are ready and click **Go**.
3) When the Target Camera reaches the first fiducial, analyze the updated image in the display.
4) Click on the image in the window where the center of the fiducial mark is located. This step is optional if the fiducial is already centered in the image.
5) Click **Submit Correction** to accept the correction.
6) Repeat steps 3-5 for the second fiducial.

7 Conformal Printing

Systems equipped with a laser displacement sensor can be used to obtain height information from surfaces which are uneven, warped, unlevel, or tilted. Different methods are used to apply height adjustments. The methods for performing height adjustments are as follows:

1. Single-Point Correction
2. Four-Point Correction
3. Line Scan
4. Path Scan
5. Grid Mapping

7.1 Scanning Principles

The laser displacement sensor Emitter emits a thin beam of light straight down towards the floor. The reflection of the beam is read by the laser displacement sensor's Receiver. Distance is determined by the laser displacement sensor and sent to the motion controller as an analog voltage. The voltage is converted into a measurement based on the machine settings. Using the laser displacement sensor effectively ensures height accuracy when running a project. For more detailed information about how the laser displacement sensor works, refer to the laser displacement manual.

Obtaining a good surface scan is dependent on some major factors: scan resolution, laser settings, how the laser is mounted, and the range of the laser.

7.1.1 Laser Ranges

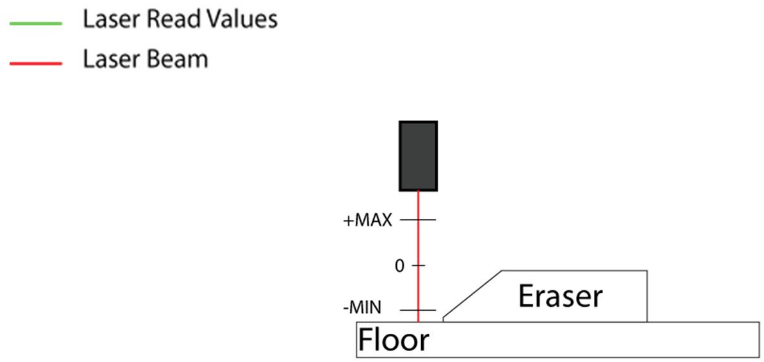
Different laser models have different ranges as shown below.

Model	Zero-Centered	Total Range
LK-081	+/- 15 mm	30 mm
LK-G Series	+/- 10 mm	20 mm
LT-900 Series	+/- 5 mm	10 mm

When the laser obtains values that are out of range, the reading is converted to the laser's max range value. This prevents crashing due to erroneous laser reads.

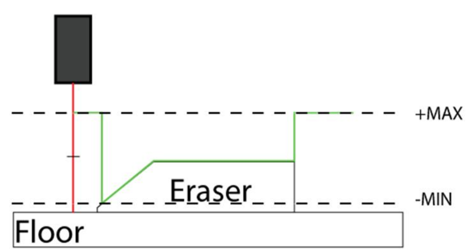
7.1.2 Example 1

In the diagrams below, an eraser was placed on the floor of the system, ready to be scanned.



Notice that the laser's range does not reach the floor. This means the floor is actually read as an error value. The laser is positioned at a height so that when the laser is moved across the eraser, the surface is within the laser's read range. The Max and Min read range are determined by the laser setting and the type of laser used. The center of the range is always zero.

7.1.3 Example 2



The green line represents the final results of the scan. The contour of the eraser is followed as long as the reading value stays within tolerance. Holes, excess material, dirt, or any other abnormalities to the surface can cause read errors. Any error values will be converted to the max value so that the pen tip does not crash into the substrate.

7.2 Single Point Correction

Single point correction is an option within print script jobs. Set the **PerformHeightTest** option in the PSJ to **True** to use this feature. A one-time height adjustment to the WorkPiece will be performed for the print job alone to follow. This height adjustment **does not** affect following jobs or Subpieces attached to the parent WorkPiece. The task is performed by bringing the laser displacement sensor to the calculated Starting XY coordinate of the Print Job and then to the calculated height of the WorkPiece based on the laser displacement sensor's calibration. The value that is received from the laser displacement sensor is used as the height adjustment which is temporarily added to the Z value of the WorkPiece's translation.

7.3 Planar Correction or Four-Point Correction

A planar correction job (**PCJ**) is also called four-point correction because this job takes a displacement reading at four predefined points just inboard of the edge of the substrate. These points are used to compute a flat plane which corrects for non-level substrates. Technically only three points are needed to sample a flat plane however a fourth point is taken so as to verify the quality of the other three points. If the fourth point is not within a tolerance of where it should be on the plane defined by the first three points then the job will fail out its work piece.

7.4 Line Scan (**Print Link Enabled**)

A line scan job (**LSJ**) scans a line, from a point to an offset from that point, and records the values to a Process Data file. A line scan's basic function is to collect the process data for analysis and verification. For example, line scan information can be recorded, printed, and then recorded again. The difference between these two scans represents the thickness of the dispensed material.

<u>Print Link Enabled:</u>

To use this option, set the **PrintLinkEnabled** value in the Job editor to "true". The line printed with the line scan information through the Auto Print Job uses the highest point recorded to adjust the height of the print. Printing this way does not contour the substrate.

7.5 Path Mapping

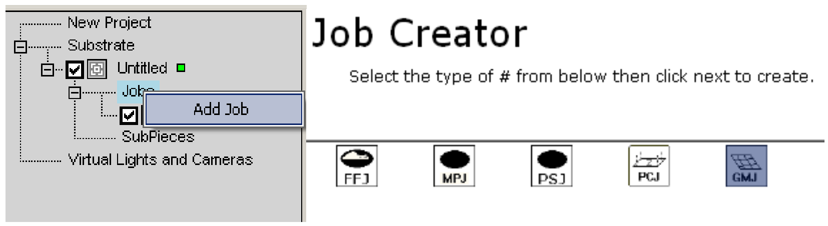
A path map job (**PMJ**) uses a script file to collect height information along the script's generated tool path. This method of scanning adjusts the tool path, based on a resolution, to contour the scanned object. Path Map scripts are intended for one-time use. Each time a path map is run, it must rescan the tool path before adjustments can be made. Sometimes this may be much more efficient than a full grid map for patterns which are linearly long but are not complex (such as serpentine patterns).

7.6 Grid Mapping

A grid map scans a surface along a serpentine pattern calculated by its settings. The data collected along the path is cleaned and triangulated into a mesh for collision detection. A grid map takes the longest of all the methods to scan but captures the most detail. Grid scans are optimized to be reusable throughout the project as long as the substrate remains in the same position at all times. If the substrate is moved, a new grid scan must be performed to ensure accuracy. Depending on the substrate, sometimes it is more efficient to scan small sections of the substrate rather than the entire substrate.

7.6.1 Adding a Grid Map Job

To use the grid map in your project, add a grid map job (**GMJ**) to a work piece. A grid map job will automatically scan or load grid map files for the current WorkPiece. Grid Maps are stored in the Map Data folder of the current project directory. The file name for the scan data is the same name as the Grid Map Job. When processing, the Map Data directory will be searched for a file with the same name as the Grid Map job. If no file is found, a new scan is performed.

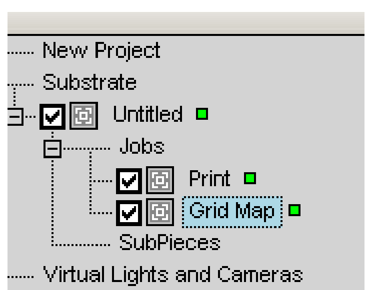


Please note:

- All tool paths generated by the scripts in print jobs that follow grid map jobs in a project will be adjusted in the Z-axis by the scanned data.
- Although the grid scan XY data is relative to its work piece, if the substrate is moved in the machine it is recommended that a new scan be taken to ensure accuracy.

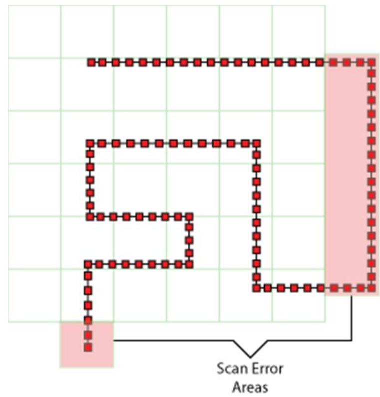
7.6.2 Ordering Matters

Adding the grid map job after the print does not affect the print; however, it may be useful for volumetric calculation. If volumetric calculation is required, add a grid map before and after a print job and save the results to two separate files. Once the project is completed, these grid maps may be compared to determine total print volume.



7.6.3 Scan Size

Make sure the scan area is large enough and is positioned to encompass the script points. The machine checks the script points against the grid map during project execution after scanning and just before printing begins. If any points in the script are outside of the scanned area, the print script job will fail before moving to the surface to begin printing. I.E. the machine will not execute a partial script if some of the points are outside the scanned area. See the illustration below.



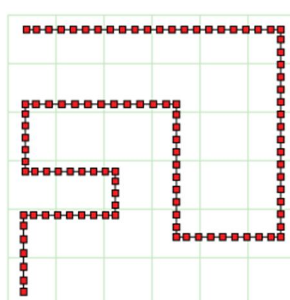
Rotation does not affect grid map jobs. If the grid map job is only just as large as the pattern, the pattern may exceed the grid map area. Printing along the edges of the scan area is not suggested. A path point can easily be calculated outside of the scan area due to minor inaccuracies caused by calibration.

7.6.4 Scan Resolutions

It is important to understand the two different resolutions which are used with grid mapping.

1. **Scan resolution** is the spacing used to generate a grid to clean the data collected from the scan. The scan resolution is also used to generate the scan path. This resolution is set by the **Scan Resolution** parameter in the grid map job editor.
2. **Print resolution** is frequency in which the script file's generated tool path is divided. This resolution is set by the Print Resolution parameter in the Print Script Job.

In the illustration below, the print has been split by its print resolution. All points exist within bounds of the grid map, so all calculated heights are satisfied by the grid. At each small red block the height of the generated tool path will be adjusted to conform approximately to the surface. The spacing between the larger blocks is the scan resolution.

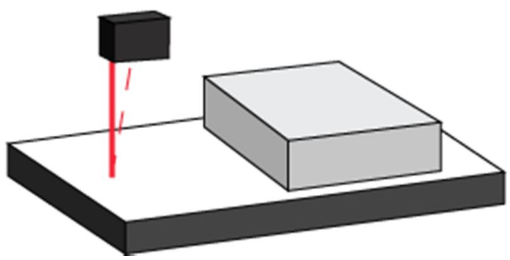


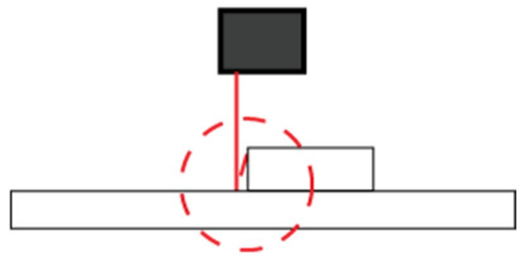
This grid is generated by the **Scan Resolution** parameter in the grid map job editor. The second is the resolution to which lines are broken up from the script file. For instance, for a scan resolution of 2 mm and a script with a 10 mm line, the 10 mm line must be broken into segments, and the starting and ending coordinates of each segment must be adjusted as per the triangle sheet. The grid on which the segments are computed is set by the **Resolution** parameter in the grid map job editor and is typically equal to the scan resolution. The resolution is commonly equal to the scan resolution.

7.6.5 Optimal Scan Resolution

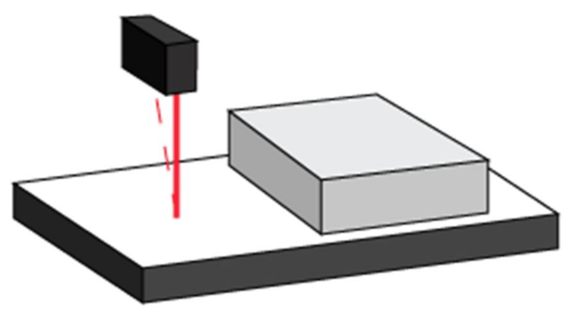
The type of surface should determine the resolution of the scan. The more variation in the surface, the higher the scan resolution should be so that fine details are recognized.

Scan perpendicular to the laser mounting. On most systems, the laser is mounted along the Y axis, so left- right scanning is best (PosXPosY).





NOTE: The laser needs to receive the reflection of its beam in order to read height data. In the illustrations above, the orientation of the laser causes blocking, or shadowing, when approaching a right angle.



Some substrate geometry may have walls or angles which are difficult to scan. The images above show the best way to get usable results from such geometry with a single laser.

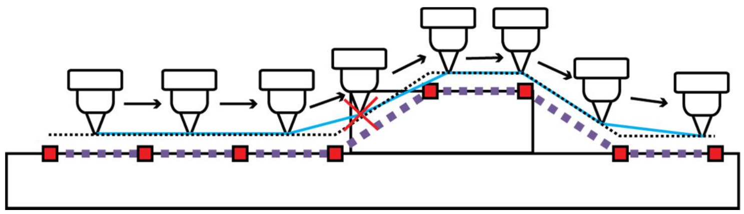
In the example above, the laser is mounted along the Y axis, which means the angle of reflection is on the Y axis. By scanning along the X axis, to the right, the leftmost wall of the obstruction is approached. The angle of reflection is parallel to the wall being scanned. This mitigates any kind of shadowing. Scanning along the X axis is the most optimal in this case. The only wall which will cause shadowing is the front wall of the obstruction.

7.6.6 Scan Resolution Examples

In the following example illustrations, the following applies.

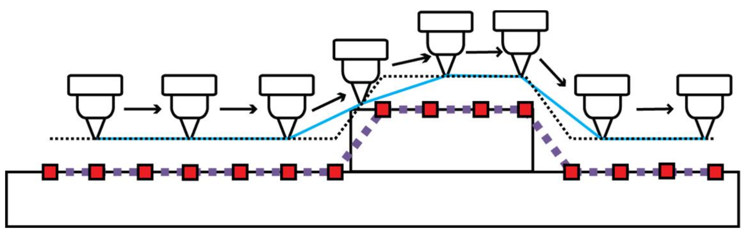
- The **red dots** represent the Scan Resolution.
- The **smaller purple dots** represent the plane of the triangle generated based on the scan resolution.
- The light blue line represents the path the tip of the pen tip takes.
- The tips represent the Print Resolution.
- The fine dotted line represents the projection of the scan data based on the dispense gap of a print. This illustrates the difference between the scan data's adjustment and the final tool path adjustment based on the print resolution.

7.6.6.a Example 3



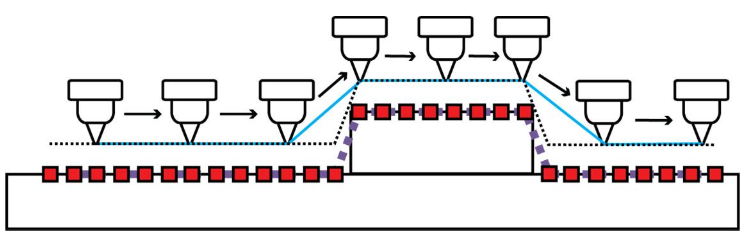
In this example, too low a Scan Resolution was used, so the pen tip collides with the corner of the block.

7.6.6.b Example 4



In this example a higher resolution was used, but the pen tip barely misses the corner of the block and prints at too low a dispense gap.

7.6.6.c Example 5



In this example, the scan resolution was increased to avoid crashing into the corner of the block.

7.6.7 Characteristics

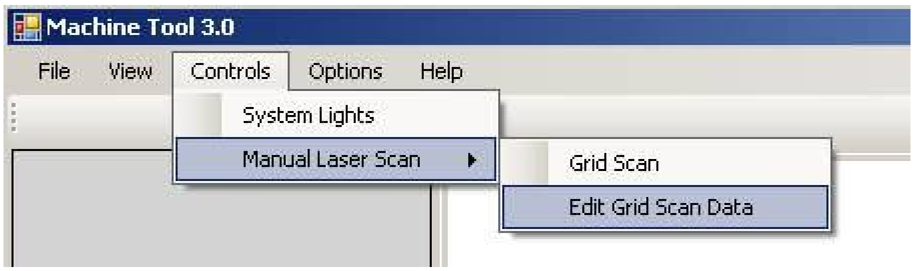
- WorkPiece rotations caused by Fiducial Find Jobs and Planar Correction Jobs are not applied to the generated scan path.

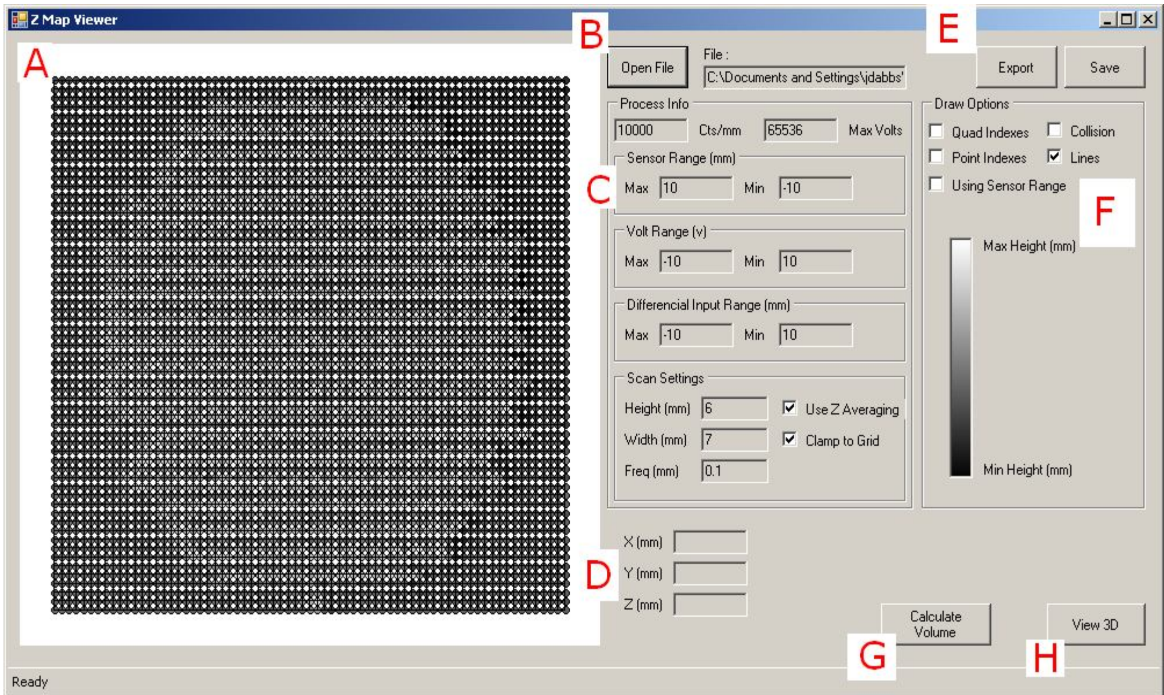
7.6.8 Grid Map Settings

Parameter	Edit	Description
Starting Direction	yes	Determines the starting direction of the scan path. The grid will always be a rectangular shape. The resulting scan path follows a serpentine pattern.
Columns	yes	The number of grid units which will be generated along the X-axis.
Rows	yes	The number of grid units which will be generated along the Y-axis.
Width	no	The size of the scan across the X-axis.
Length	no	The size of the scan across the Y-Axis.
Sample Count	no	The total number of expected scan points which will be generated by the scan data.
Primary Laser Name	yes	Selects the first laser to scan with.
Save Clean File	yes	Saves a (*.czg) to the project's Grid Map folder. If a previous file is found, the user is asked if a new grid scan should be made. Answering no will load the previous scanned grid map. If a clean file is available, the clean file will be reloaded. This saves time by eliminating the need to rescan and clean.
Notify On Complete	yes	When a grid map is completed, this option will allow the process to be paused without continuing the remaining jobs. This allows the user to view and edit the scanned data before continuing.
Local Offset	yes	This value allows the adjustment of the positional offset from origin of the Work piece the job is attached to.
Resolution X & Y	yes	This value determines the spacing between points in the grid along the X-Axis and Y-Axis. Unit grids are currently the only kind of grid which can be loaded. Therefore, these two values must be the same.
Travel Speed	yes	The speed of the scan.

7.6.9 The Scanned Data Viewer

Launch the Scan Data Viewer (see image below) to open, view, edit, and save scan data. This editor is used to inspect and edit the scanned data when a new scan is performed. When using the Notify On Complete option, the machine will pause. The scan data may be modified by the viewer, but the changes will not be recognized by the project until a grid map job loads the corrected file.





Section	Name	Description
A	2D View	Shows the grid from the top down. The shade of each dot in the grid is determined by its height value. Clicking on this view will give you a height point in Position Solver.
B	Open File	Launches the file browser to select a scan data file to load. (*.czg) is a clean file and (*.zmg) is a raw data file.
C	Process Info & Scan Settings	These are the settings that were specified when the scan was performed. (See cleaning options below.)
D	Position Solver	Displays the clicked position in the 2D View in actual units.
E	Export & Save	*Save* will save the current data as a (*.czg). *Export* saves the file as a (*.x) file.
F	Draw Options	Changes the appearance of 2D View. (See Draw Options below.)
G	Calculate Volume	Calculates the volume of the current scan data and displays the result in a message box. **Note:** Ctrl+c will copy the results within the message box.
H	View 3D	Launches the 3D viewer, which is used to view and modify the grid map data.

7.6.10 Cleaning Options

If these options are disabled, the closest point to the generated collision area's center per grid position is used instead.

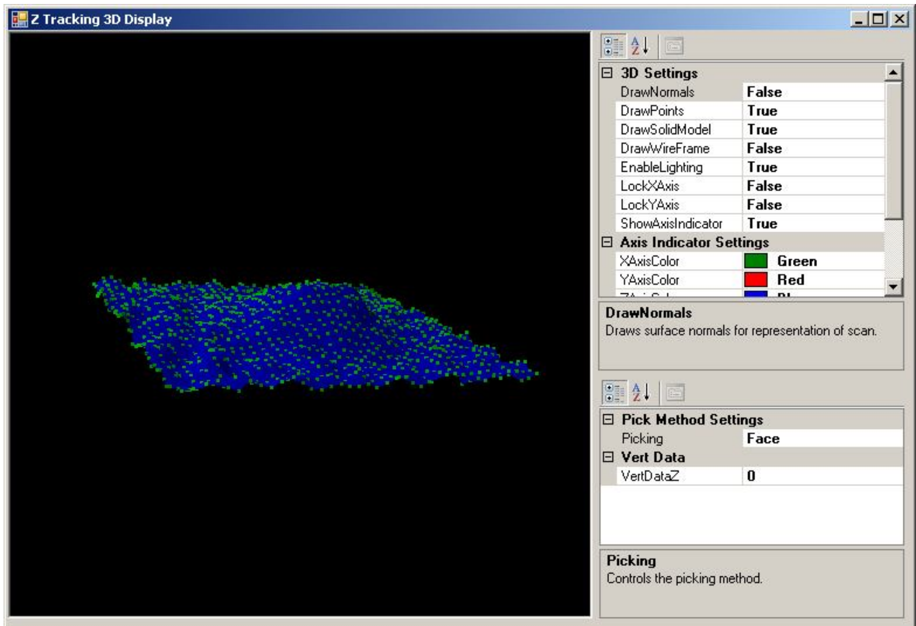
Cleaning Option	Function
Use Z Averaging	Performs calculations to get the best determined height value for a given position in the map.
Clamp to Grid	Ensures all points are clamped to the grid during the cleaning process.

7.6.11 Draw Options

Cleaning Option	Function
Quad Indexes	Draws the index number for each grid unit.
Point Indexes	Draws the index number for each point.
Using Sensor Range	Draws the shade of each point based on the sensor range. If this is disabled, the Min and Max height values for the entire grid are used instead.
Collision	Draws the location of each collision area. The diameter of each collision area is the same size as the scan resolution.
Lines	Draws the lines for each triangle in the grid.

7.6.12 Editing a Grid Map

Launch the Scan Data Viewer, load a grid map, and click **View 3D**.



In this view, you can view and modify the scan data in 3D. The X & Y coordinates of each vertex are locked to ensure the integrity of the grid. Errors in the grid can be adjusted to prevent problems when using this map during printing.

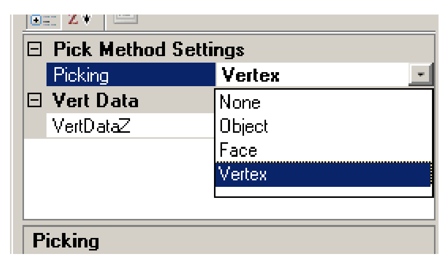
When finished with the 3D viewer, click **close**. A prompt will ask you to commit the changes. Clicking **yes** updates the data in the 2D view. To save the changed data, click the **Save** button once the application has returned to the 2D view.

7.6.12.a Rotate view

To rotate the view, right-click and drag on the 3D scan view area.

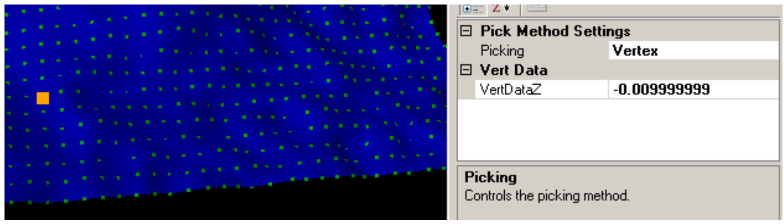
7.6.12.b Select Vertex

To select a vertex for editing, in the lower right options area, select **Vertex** mode from the **Picking** dropdown list. This allows the selection of vertices by clicking on the desired vertex.



7.6.12.c Modify vertex value

To adjust data, select a vertex in the 3D view to see its current height value. The height value can be edited using the **VertDataZ** parameter in the lower right options area.

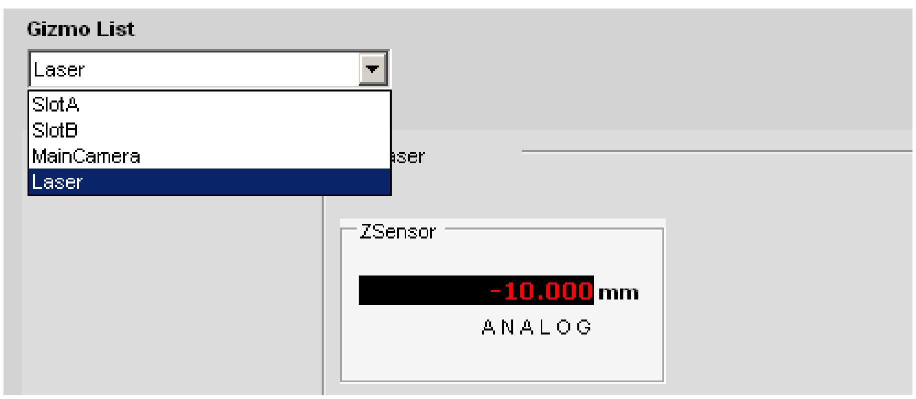


7.6.13 Manual Grid Mapping

A grid map can be generated manually by using the manual laser scan option.

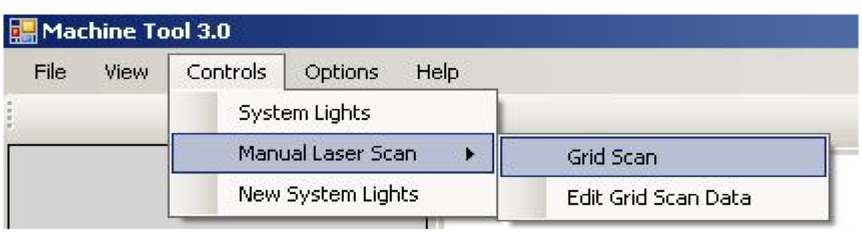
1) Using the XY motion controls, move the machine to the starting position of your scan.

2) To monitor the current laser reading, select the laser gizmo control.



3) Move the machine position down in Z to select the scan height. At no time should the substrate drop below the minimum or rise above the maximum range of the laser. The position of the Z axis will remain constant throughout the scan.

4) To access the manual grid scan, navigate to **Controls -> Manual Laser Scan -> Grid Scan** as shown below.



5) Enter the scan parameters and click **Start Scan**. The scan parameters are described below.

7.6.14 Manual Grid Map Settings

Parameter	Edit	Description
Output File Select	yes	Specifies the file path the scan data will be saved to.
Starting Direction	yes	Determines the starting direction of the scan path. The grid will always be a rectangular shape. The resulting scan path follows a serpentine pattern.
Width	no	The size of the scan across the X-axis.
Length	no	The size of the scan across the Y-Axis.
Columns	yes	The number of grid units which will be generated along the X-axis.
Rows	yes	The number of grid units which will be generated along the Y-axis.
Resolution X & Y	yes	This value determines the spacing between points in the grid along the X-Axis and Y-Axis. Unit grids are currently the only kind of grid which can be loaded. Therefore, these two values must be the same.
Sample Count	no	The total number of expected scan points which will be generated by the scan data.
Primary Laser Name	yes	Selects the first laser to scan with.
Secondary Laser Name	yes	If two lasers are mounted, this option becomes available. (Otherwise, this is disabled.)
Travel Speed	yes	The speed of the scan.
Planar Scan	yes	Not used.

7.6.15 File Definitions

Type	Extension	Description
Z Map Grid	(*.zmg)	Non-cleaned data file containing all points collected along the scan path of the laser. This file needs to be cleaned and processed before its data can be used.
Clean Z Map Grid	(*.czg)	Contains only the final values used to generate a grid. Using this type of file is quicker than loading a normal Z Map Grid.

8 Rotary Stages

Rotational script jobs (RSJs) are designed for systems with rotary stages. A rotary script job rotates the stage by a predefined amount when this type of job runs. This job will not rotate the substrate while printing; rotating while printing must be performed within a script.

9 Calibration

9.1 Why is calibration necessary?

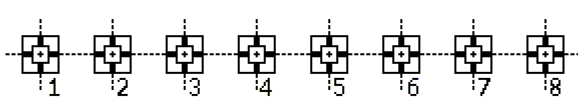
Calibration is vital to maintain accuracy and avoid damaging the machine. Devices attached to the machine's tool plate are referred to as Gizmos. SmartPumps™, laser displacement sensors, the target camera, and UV curing lamps are a few of the devices which are listed as Gizmos. Each of the gizmos arranged on the tool plate are offset from a master device at fixed calibrated distances. The master device is always the gizmo attached to Slot A on the tool plate. Given a world coordinate, the calibrated offsets determine where the machine will move a gizmo into position. This allows any calibrated gizmo to be brought to the same position where work is to be performed.

Calibration is achieved by bringing all of the gizmos to a single position so their offsets may be calculated. There are two options for performing calibration: Manual or Automatic. Manual calibration uses manual motion controls to move each of the devices to a registration point on a physical object, such as a **calibration eraser**. Automated calibration uses a vision pit with cameras aligned to a conical which is the registration point. In either mode, each gizmo is precisely positioned at the target so that the world coordinates of each gizmo may be recorded. Based on these world coordinates, the machine computes offsets. Camera access used during calibration is linked to the vpp file assigned to the Calibration Vision File parameter of the machine settings.

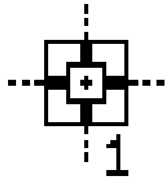
9.2 Manual Calibration

9.2.1 Fabricating a Calibration Eraser

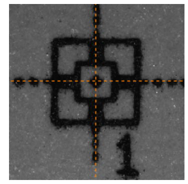
A calibration eraser is a soft physical object with one or multiple target(s) to which each gizmo is brought to for calibration. A target is printed on paper and double-stick taped to the eraser. Several targets are printed, on a single eraser since targets are occasionally marred during calibration. An example target is shown below.



When printing the targets, make the center dot as small as possible to achieve accurate positioning. Do this by setting the print dpi to 600 and using a high DPI image file. Unfortunately, Microsoft Paint is unable to modify the dpi from its default of 96 dpi and will not work for drawing a calibration target. Contact nScrypt to obtain a suitable calibration target image file.



Target



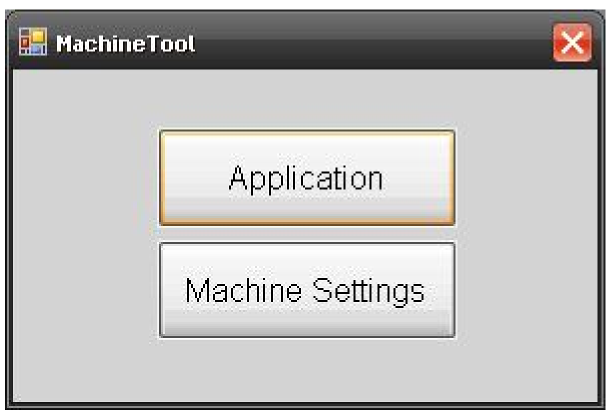
Camera View

The target shown on the right was printed using a high-speed office laser printer at 600 dpi. Each ink jet pixel is about 50 µm, and the Target camera reads 9.9 µm / pixel. This resolution is sufficient for most applications, but a higher resolution would be needed for certain projects, especially those using 12.5 µm or 25 µm tips.

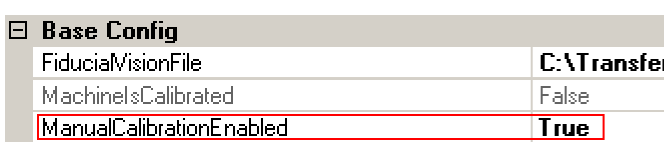
9.2.2 Launch Manual Calibration

1) Either open or reopen the software to bring up the mode dialog.

2) Launch **Machine Settings** from the dialog box.

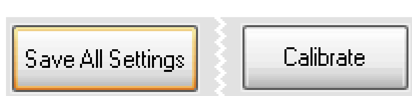


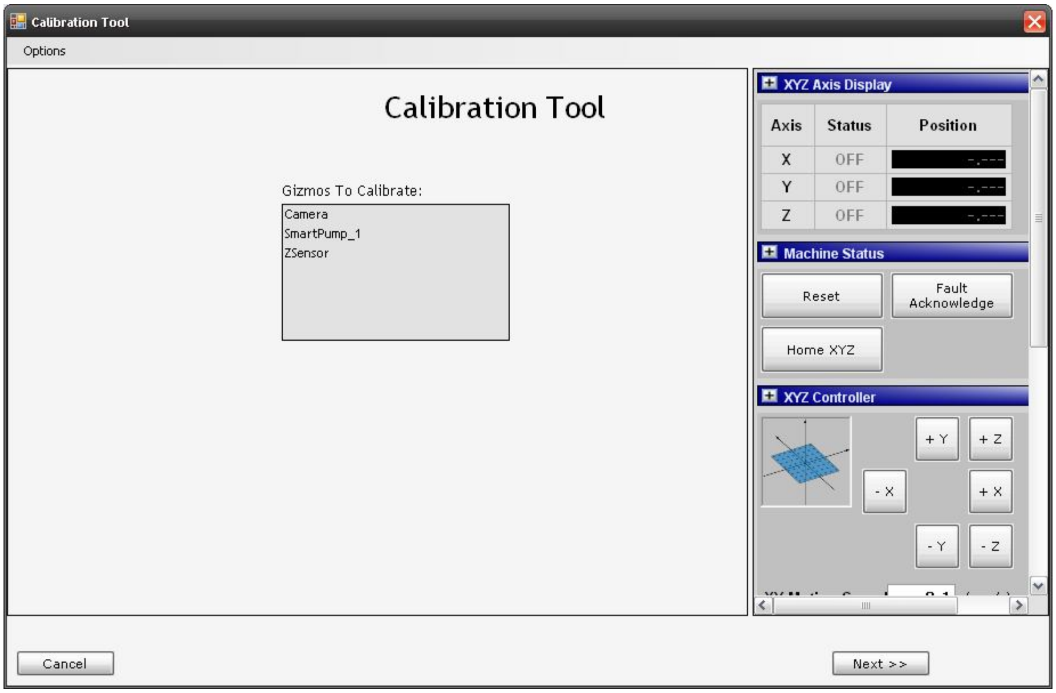
3) In the settings area, set the **ManualCalibrationEnabled** property to **True**.



4) Click **Save All Settings** to enable calibration.

5) Click **Calibrate** to launch the calibration wizard.

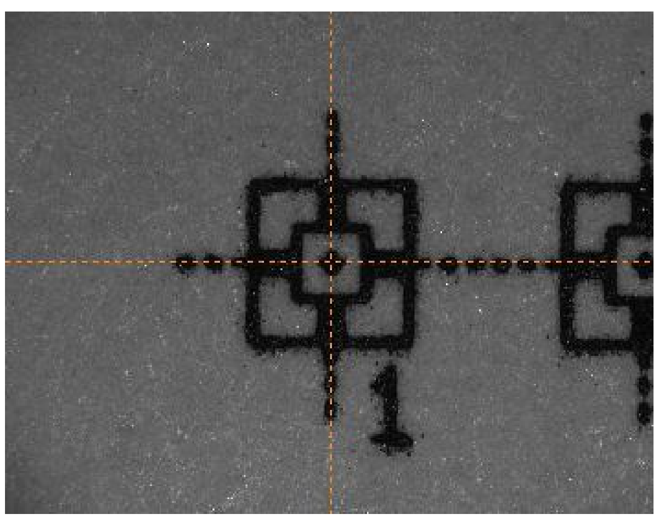




The **Gizmos To Calibrate** list shows which gizmos will be calibrated. The gizmos to be calibrated may be edited in **Machine Settings**. The following sections cover all gizmos which may be calibrated. (Your system may not have all of the following gizmos.)

6) If necessary, pull out the e-stop button and push the reset button; then click **Fault Acknowledge** then **Reset** to activate the machine.

9.2.3 Target Camera



1) Using the motion controls on the right, bring the camera near or over the calibration target. Be aware of the other gizmos on the machine to avoid crashes.

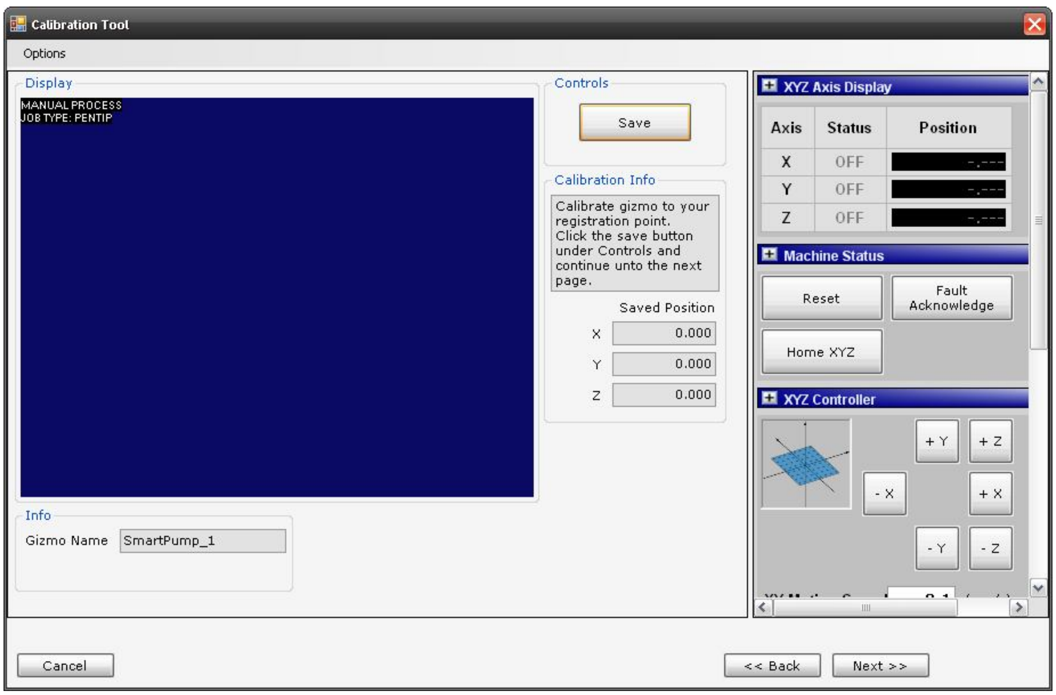
2) Using the **Options -> System Lighting** menu item at the top, adjust the Target camera's ring light for optimum illumination. Full or near full intensity is typically best.

3) Using the motion controls, bring the camera's crosshairs over the calibration target and use the Z stage to optimize the focus.

4) Once camera is targeted, click **Save** to save the system coordinates to the **Saved Position** boxes.

5) Click **Next**.

9.2.4 SmartPump™



6) SmartPumps™ are calibrated in order starting from Slot A / SmartPump™, which is the rightmost pump on multi-pump machines. On machines with actuation, only the current SmartPump™ to be calibrated will be in the down position.

7) Using the motion controls on the right, bring the SmartPump™ near the calibration target. Be aware of the other gizmos on the machine to avoid crashes.

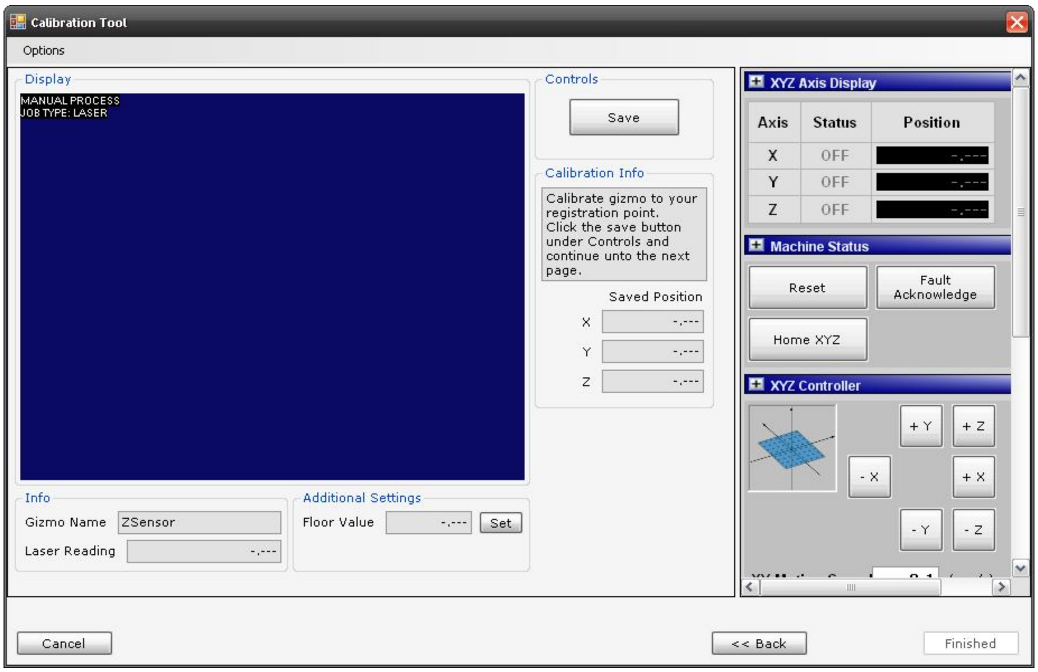
8) Using the **Options -> System Lighting** menu item at the top, adjust the process view lighting for optimum illumination as viewed on the monitor above the computer screen.

9) Using the motion controls, bring the pen tip to the calibration target in X, Y, and Z. On systems with X and Y cameras, be sure to use both cameras to verify targeting. In this case, set the Z speed to 0.1 mm/s and allow the pen tip to slightly indent the target then pull back from the target in order to verify accurate Z height adjustment. Once in place, the tip should be touching the surface and neither indentation nor a gap should be visible between the tip and the target.

10) Once the SmartPump™ is calibrated, click **Save** to save the system coordinates to the **Saved Position** boxes.

11) Click **Next**.

9.2.5 Laser Displacement Sensor



NOTE: In the following steps, the term **zero the laser** refers to adjusting the Z stage until the **Laser Reading** displays 0.000 +/- 0.002 mm (2 µm).

1) Using the motion controls on the right, bring the laser displacement sensor to the calibration target until the laser begins reading. Be aware of other gizmos on the machine to avoid crashes.

2) Adjust the X and Y stages until the laser dot is in the center of the target.

3) Once the red laser dot is visible on the calibration target, **zero the laser**.

4) Dimming the laser beam will aid accuracy.

   a. Open **Windows Start Menu -> Keyence Applications -> LK Navigator**

   b. Use the option to **Read Controller Settings**.

   c. On the main page, set **ABLE(P)** from **AUTO** to **MANUAL 1-2**

   d. Click **Send settings to controller**.

   e. The LK Navigator software need not be closed.

5) Adjust the X and Y stages until the laser is precisely targeted. Do not adjust the Z stage while the laser is dimmed.

6) Return the laser to **ABLE: AUTO** mode using the above steps.

7) Zero the laser again now that the laser is accurately positioned in X and Y.

8) Click **Save**.

9) Move the laser to an area within the machine where substrates are to be placed. This area is termed the **floor** and may be a vacuum chuck, aluminum block, or a custom part fixture.

10) Zero the laser to calibrate the floor.

11) Click **Set** next to the **Floor Value** box.

WARNING: Do not click Save again! If save is inadvertently clicked, recalibrate the laser to the calibration target.

12) Click **Next**.

Once all of the gizmos are calibrated click the Finished button to commit the changes to the machine settings. The Save as Master Calibration dialog which appears updates the offsets of the gizmos for Auto Calibration.

9.3 Automated Calibration

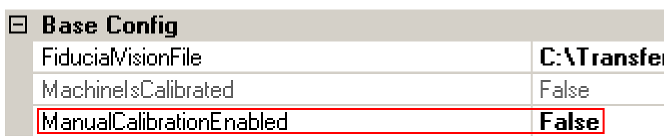
Automated calibration, or **auto cal**, is available on machines with a vision pit. The vision pit consists of two orthogonal cameras trained horizontally on a pen tip-like ceramic conical. Using these two cameras and vision software, each gizmo is automatically calibrated to the conical.

9.3.1 Launching Automated Calibration

1) Either open or reopen the software to bring up the mode dialog.

2) Launch **Machine Settings** from the dialog box.

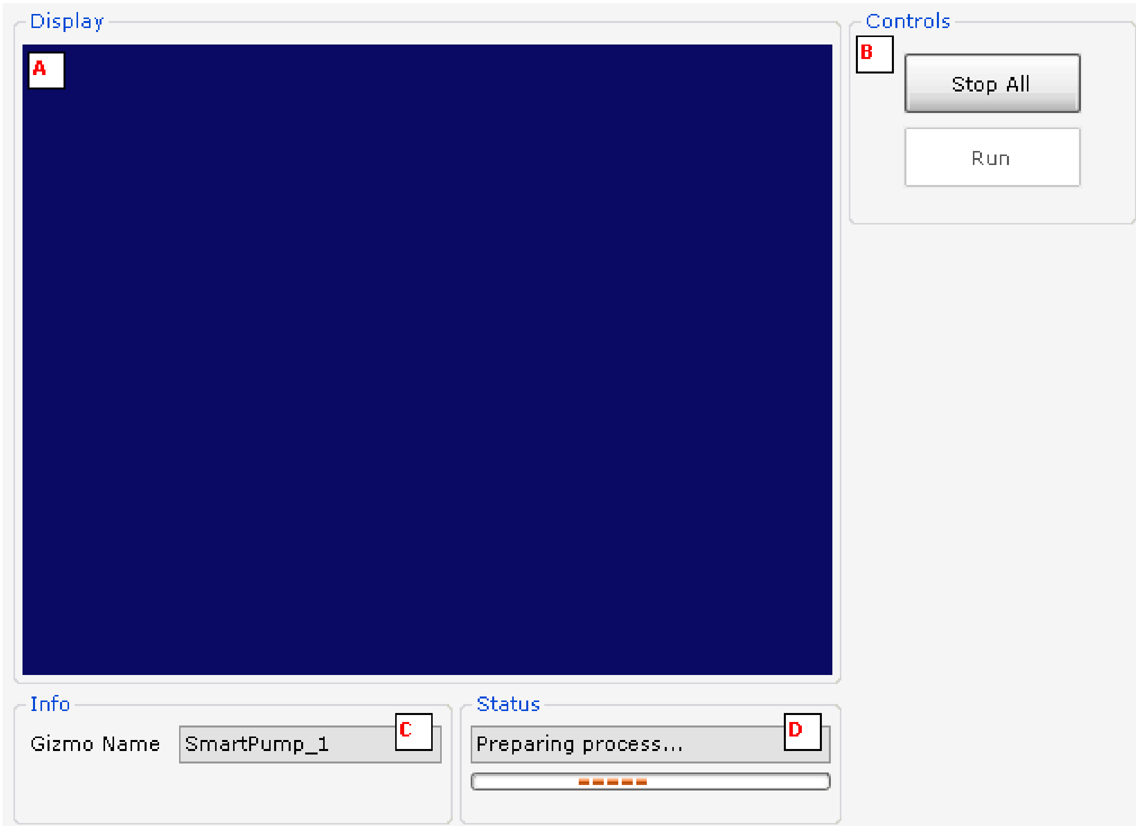
3) On the initial tab, set the **ManualCalibrationEnabled** property to **False**.



4) Click **Save All Settings** to enable calibration.

5) Click **Calibrate** to launch the calibration wizard.

9.3.2 The Auto Cal Page

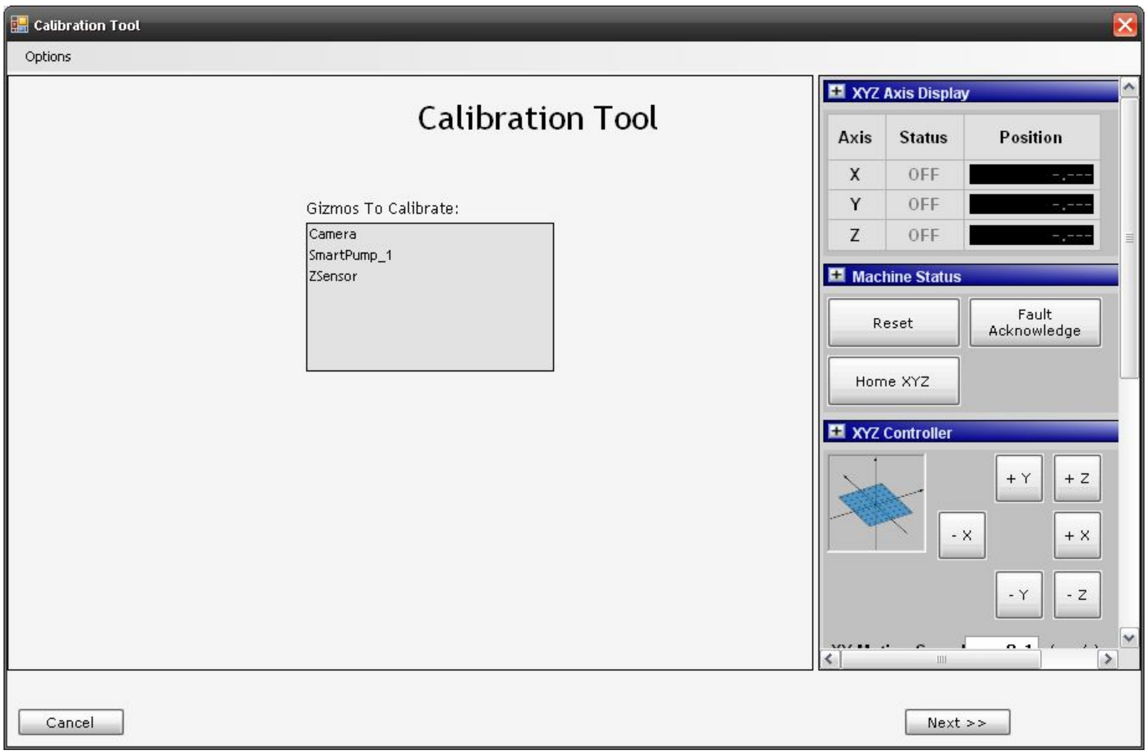


Section	Name	Description
A	Display	This is where the corresponding camera image is displayed.
B	Stop All Run	Stops current process and any motion tasks. Runs process from its initial state.
C	Gizmo Name	The name of the gizmo currently being calibrated is displayed here.
D	Status	The process state is displayed here.

9.3.3 Performing Automated Calibration

The **Gizmos To Calibrate** list shows which gizmos will be calibrated. The gizmos to be calibrated may be edited in **Machine Settings**. The following sections cover all gizmos which may be calibrated by Auto Calibration. (Your system may not have all of the following gizmos.)

1) If necessary, pull out the e-stop button and push the reset button; then click **Fault Acknowledge** then **Reset** to activate the machine.



9.3.4 Conical

The machine automatically brings each gizmo to the conical using a master calibration found in the AutoCalibrationSettings.xml file located in the Application Directory/Vision folder. The master calibration should never need to be adjusted as long as the gizmos are not changed.

NOTE: If positioning issues occur, carefully perform a manual calibration to the top of the conical and record the positions to the AutoCalicbrationSettings.xml file. If positioning problems with auto calibration persist, contact nScrypt support.

Once a gizmo is brought to its master calibration position, various routines are used to fine-tune its final calibrated position.

9.3.4.a Lighting

The vision pit ring light should be adjusted for nominal brightness on the X and Y vision pit cameras. Select **Options -> System Lighting** to adjust the ring light brightness until the conical is clearly visible.

9.3.4.b Conical Marker

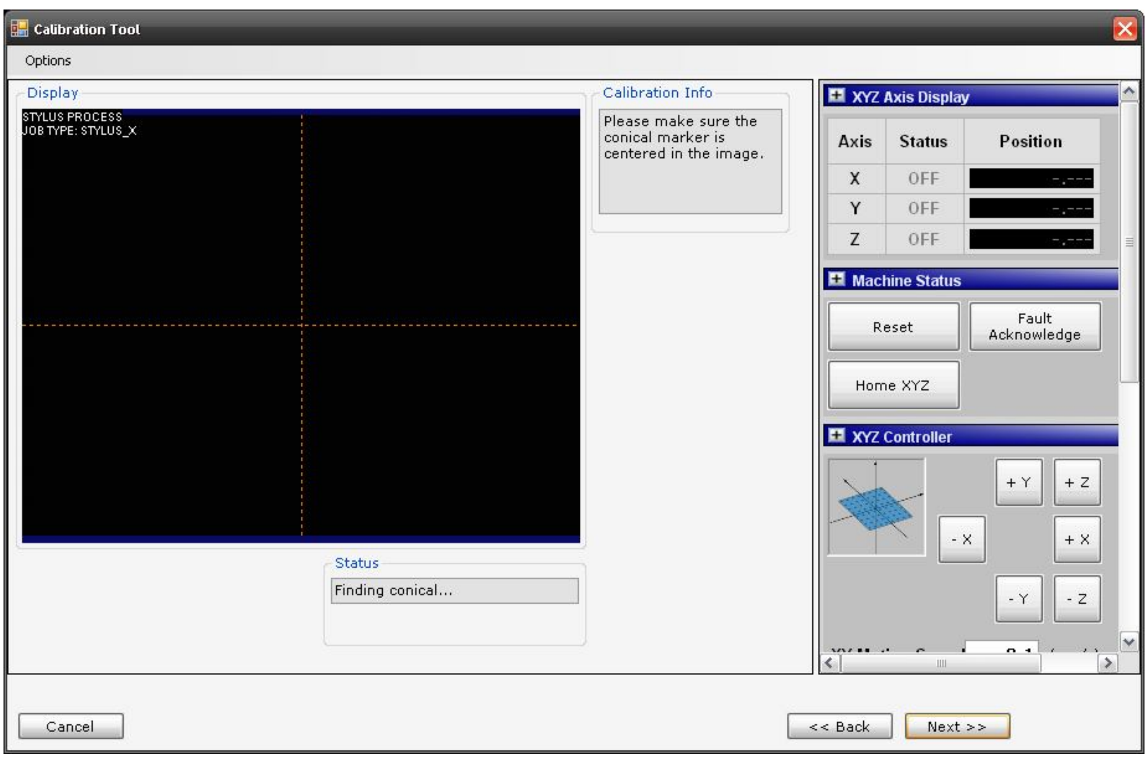
When detecting the conical, or stylus, the conical marker is the small crosshair hovering above the tip in the display of the Calibration Tool. This location is the intersection point of two lines detected as the angled sides of the conical. These two lines and the point of intersection are calculated by running the stylus detection CogJobs from the calibration vpp file.

Tune the stylus cogjobs in the Calibration vpp if:

- ☐ The conical marker does not appear.
- ☐ The conical marker flickers with delays longer than a second.
- ☐ The conical marker's position sporadically moves throughout the image.

If any of the above symptoms are observed, do not continue Auto calibration.

The main page where the system identifies the conical is shown below.



9.3.4.c Camera Alignment

The conical marker should align with the main crosshairs. If it does not, adjust the vision pit cameras until the crosshairs are aligned. This is done using the set screws on the camera mount. If positioning the conical marker to the crosshairs cannot be achieved, contact nScrypt support.

9.3.5 Target Camera

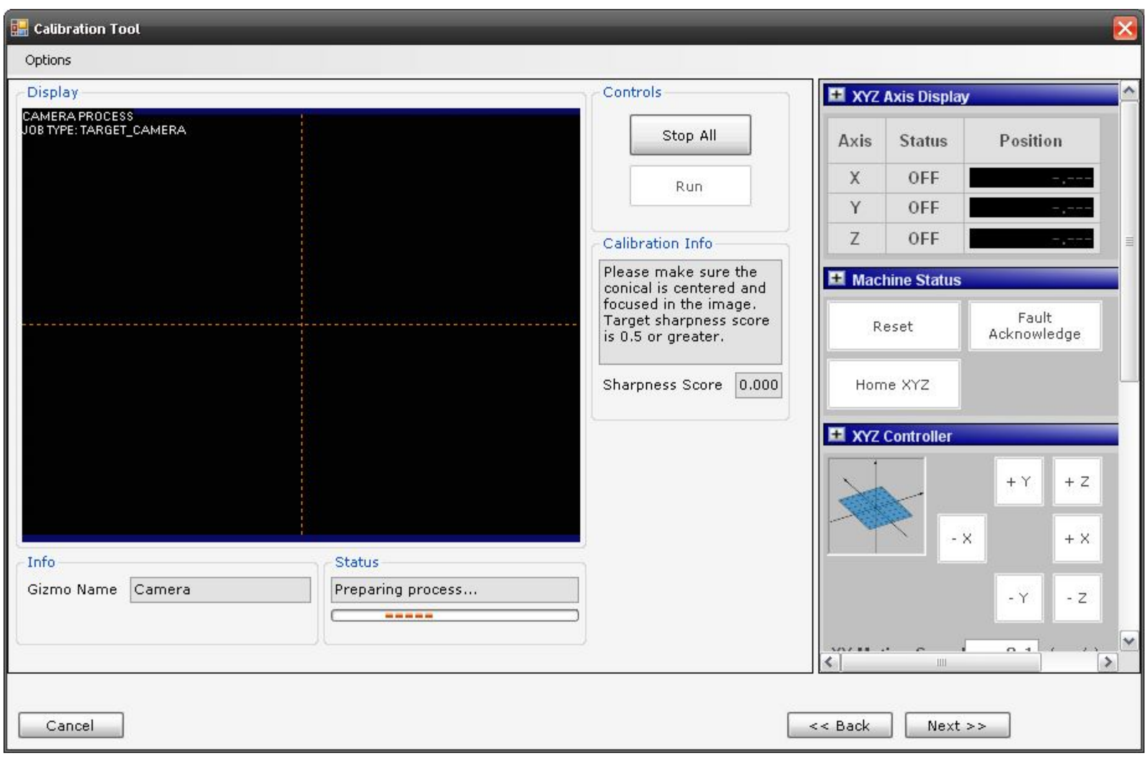
1) The target camera is moved to its master calibration location.

2) If the conical is not found, manual mode is enabled.

3) Center the conical tip in the crosshairs and adjust the Z-stage to focus on the tip of the conical.

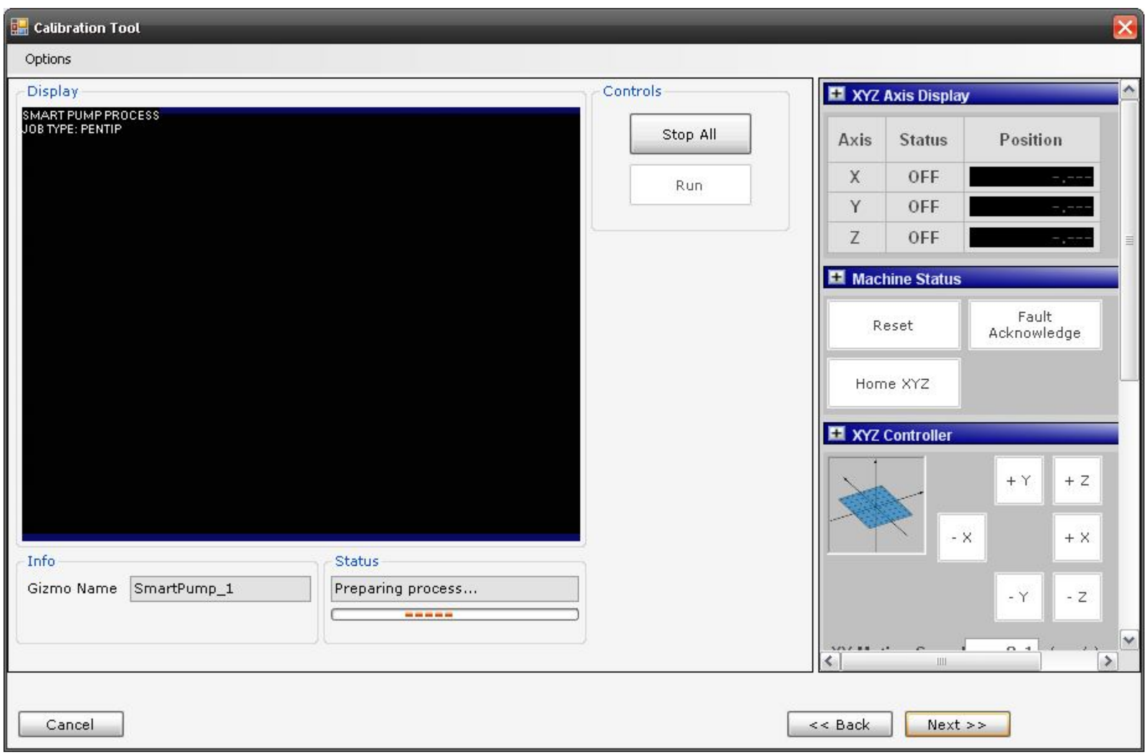
4) A sharpness score, used to determine whether or not the image is focused, can be viewed in the **Calibration Info** group box. This score must meet or exceed the value set by the SHARPNESS_TOLERANCE parameter in the AutoCalibrationSettings.xml file.

5) Click the **Run** button to continue calibration.



9.3.6 SmartPump™

1) Each pump on the system is moved to its master calibration location.
2) For each pump, the machine adjusts using the X camera.
3) For each pump, the machine adjusts using the Y camera.
4) The crosshairs for the pump and conical are aligned.
5) The calibration tool will ask if the pump looks calibrated. Click **No**.
6) Use the conical and the pen tip controls on the right to define the bottom of the pen tip and the top of the conical.
7) Click **Continue** to move the tip down to the conical.
8) When the distance between the tip and the conical are satisfactory, click **Next**.

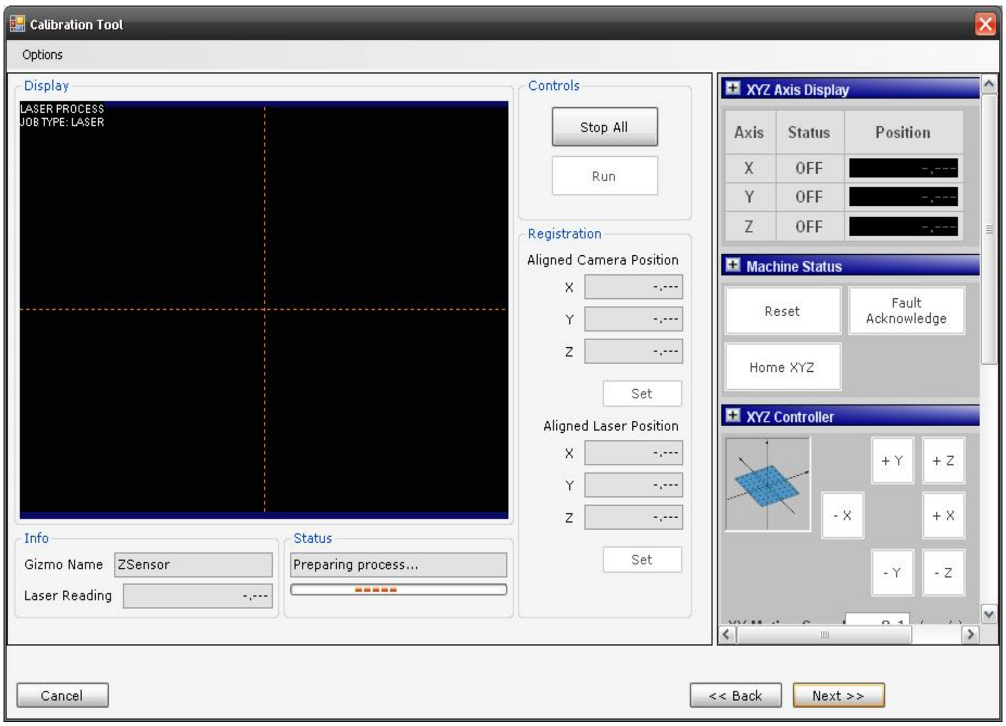


9.3.7 Laser Displacement Sensor

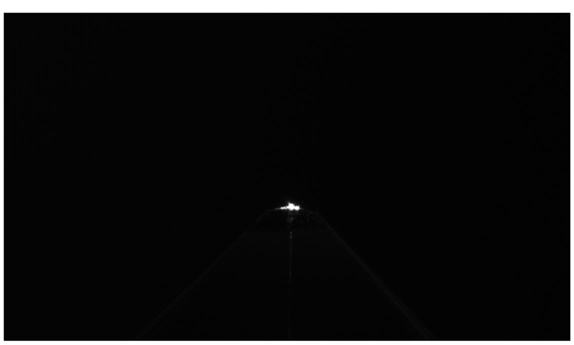
1) The machine brings the laser displacement sensor to the floor. (This step is skipped for the secondary laser.)

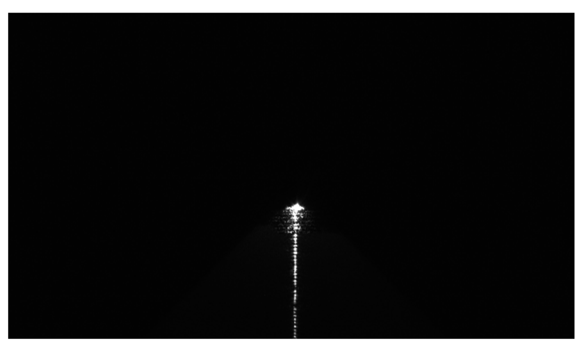
2) The machine brings the laser displacement sensor to its master calibration location at the top of the conical.

3) The laser must be within (+/- 0.050 mm) to be registered.



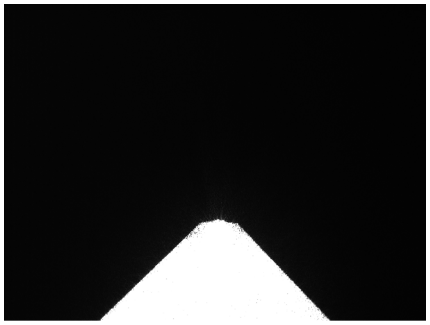
9.3.8 Proper Laser Alignment



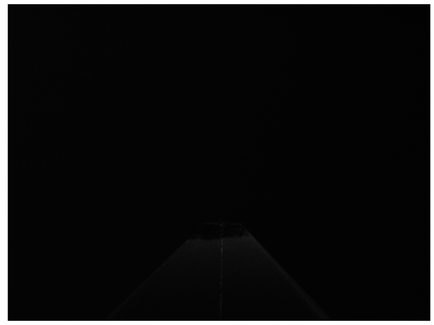


The laser is at the center of the conical when the X and Y position is moved at 0.1 mm/s and the laser reading does not vary more than +/- 10 µm. The result looks like a dome or a straight line going down the center of the conical in both X and Y View.

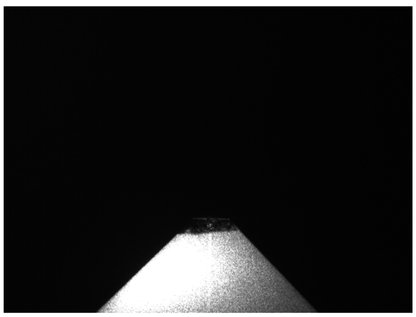
9.3.9 Improper Laser Alignment



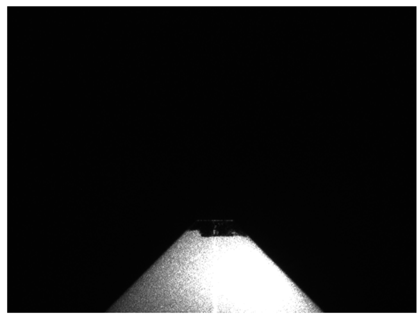
Out of read range.



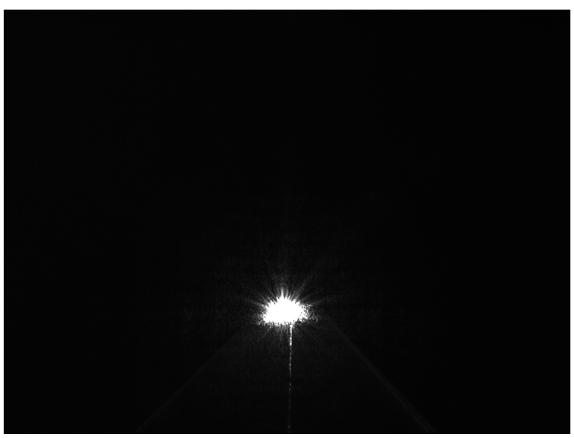
Position not on conical or position is behind the conical from camera view.



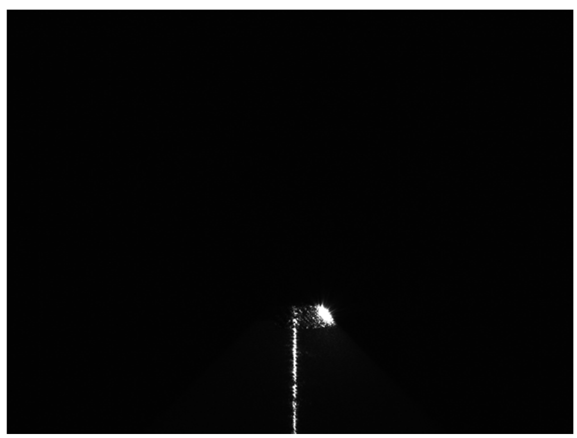
Position is off of the side on the ceramic.



The ceramic will diffuse the laser spot.



Position is off center from the top of the conical.



Corners of the conical may cause deflection.

9.4 Semi-Automated Calibration

Semi-automated calibration (semi-auto cal) works similarly to automated calibration except only one axis camera is used to observe the conical. Since two orthogonal cameras are needed, the **Process View** camera, which may be viewed by the user on the monitor over the computer screen, is used as the other camera. While performing semi-auto cal, the user must adjust the tip location to the conical for the axis perpendicular to the process view camera.

9.5 Machine Vision Setup

MtGen3 uses Cognex Vision Pro Persistence (vpp) files to communicate with the cameras to obtain images and measurement data. The Application vpp and the Calibration vpp which are shipped with the system are the default vision files needed for camera control. These files are assigned in the machine settings and may be replaced and/or modified to fit the user's needs.

The Application.vpp file is used for all camera functionality when the software is in Application mode. This file contains CogJobs, which support access to the Target camera and provide measurement data for Fiducial Find Jobs.

The Calibration.vpp file is used for all camera functionality when running the Calibration Tool. This file contains CogJobs which support conical recognition, Target camera centering, and pen tip alignment.

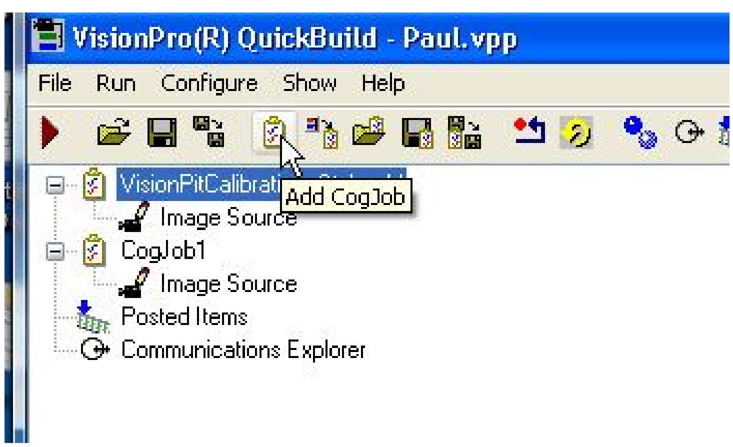
To create a new vpp file for calibration:

1) Open the VisionPro QuickBuild software.

2) Create a new **QuickBuild** application. Click File -> New QuickBuild Application…

3) Save the QuickBuild application to the vision folder C:\Transfer\installs\mtgen3\Vision.

4) Add each required CogJob listed below. Refer to the screenshot below. At this point, simply create "blank" cog jobs and name them. Instructions for configuring each will follow.



a. *"VisionPitCalibration_Stylus_X"* uses the Y camera to find the sides of the conical

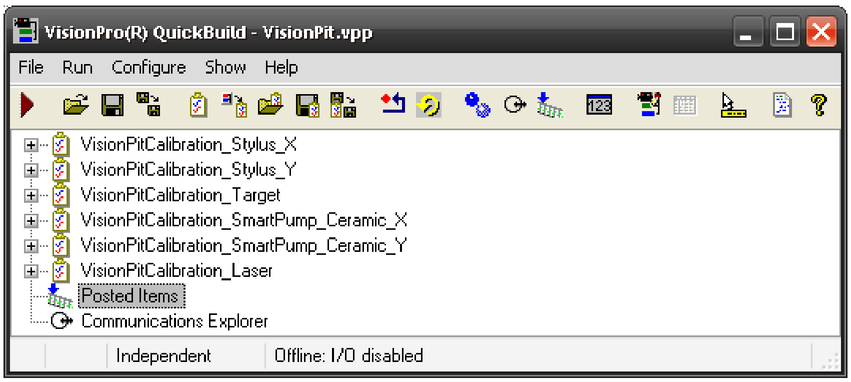
b. "VisionPitCalibration_Stylus_Y" uses the X camera to find the sides of the conical

c. *"VisionPitCalibration_Target"* uses the target camera to find the top of the conical

d. *"VisionPitCalibration_SmartPump_(tip type)_X"* uses the Y camera to align the tip to the conical.

e. *"VisionPitCalibration_SmartPump_(tip type)_Y"* uses the X camera to align the tip to the conical.

f. *"VisionPitCalibration_Laser"* uses either X or Y camera to view a laser displacement sensor alignment.



5) Set the image source for each cog job.

   a. Double-click the cog job to edit it.

   b. Click **Camera**.

   c. Set the **Image Acquisition Device/Frame Grabber** to the correct GigE camera.

   d. Set the **Video Formats** to **Generic GigEVision**.

   e. Click **Initialize Acquisition**.

   f. Click the small black camera button in the upper-left of the window to start the live display.

   g. Configure the camera settings and adjust appropriately.

      i. Exposure: 15 ms. Increasing the exposure brightens the image.

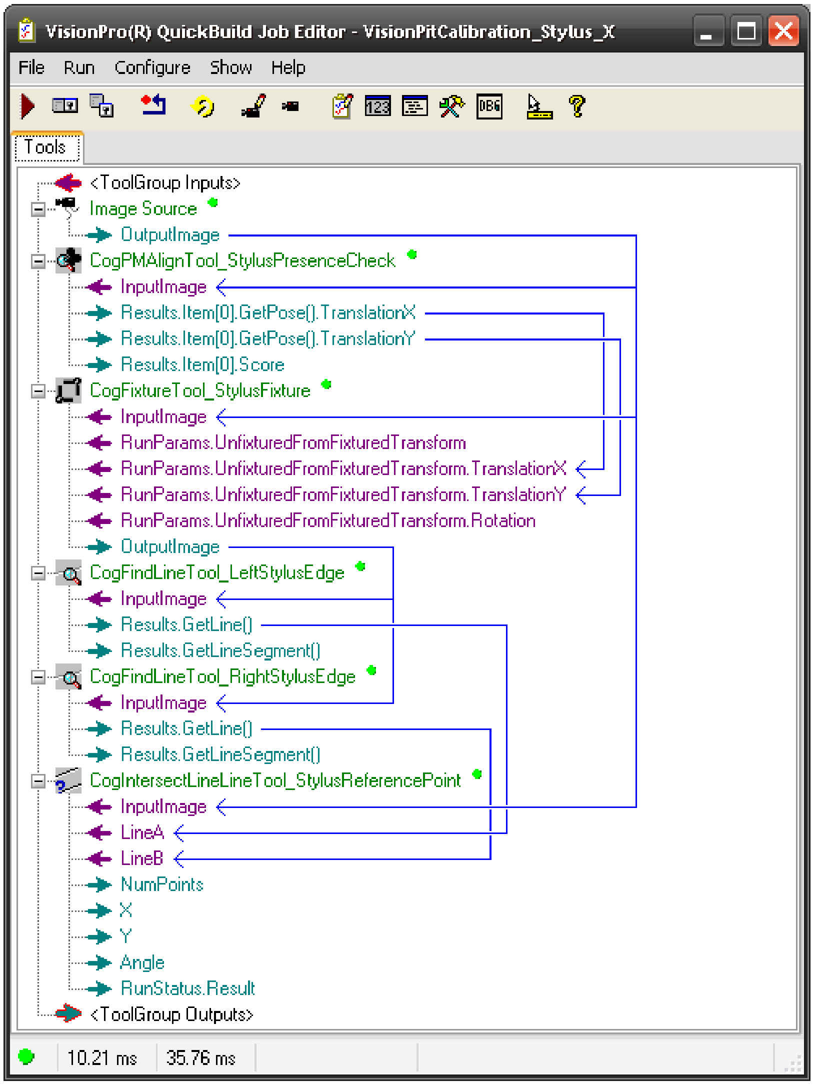
      ii. Brightness: 0.

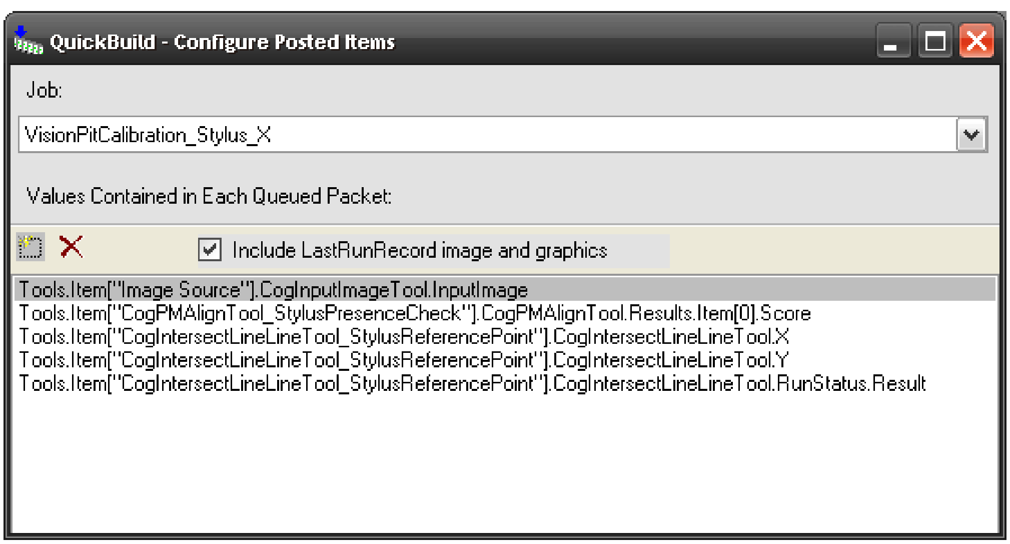
      iii. Contrast: 0.

      iv. Timeout: Checked, 10000 ms.

   h. Click close to return to the **QuickBuild** application editor.

9.5.1 Stylus Job Setup

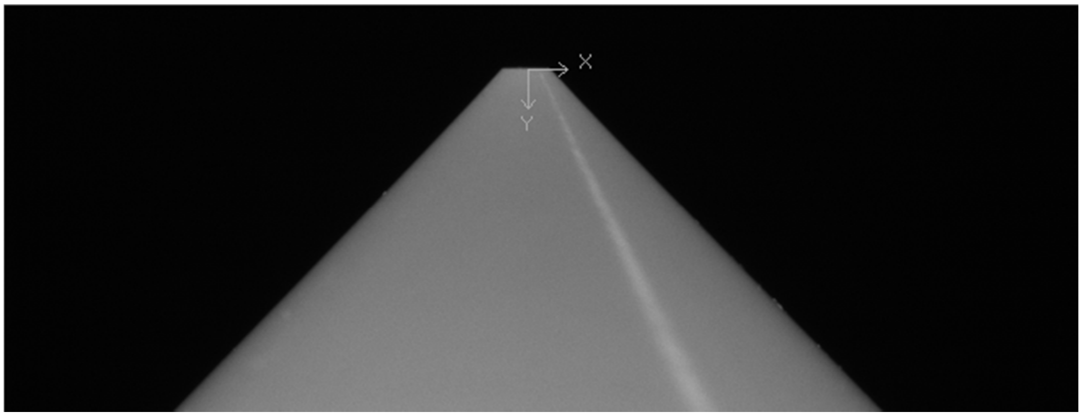




1) Add **OutputImage** from **Image Source** to **Posted Items**.

2) Add a **CogPMAlignTool** and rename it as **CogPMAlignTool_StylusPresenceCheck**.

   a) Link **OutputImage** from **Image Source** to **InputImage**.
   b) Click Train Image. The image should appear as shown below.



   c) Add “**Results.Item[0].Score**” to Posted Items.

3) Add a **CogFixtureTool** and rename it as **CogFixtureTool_StylusFixture**.

   a) Link **Image Source** to **InputImage**.

   b) Link **Results.Item[0].GetPose().TranslationX** from **CogPMAlignTool_StylusPresenceCheck** to **RunParams.UnfixturedFromFixturedTransform.TranslationX**.

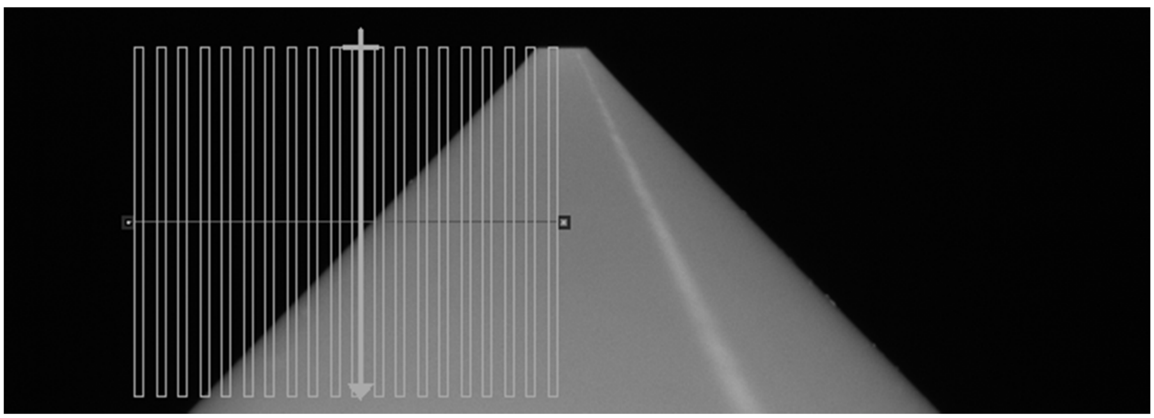
   c) Link **Results.Item[0].GetPose().TranslationY** from **CogPMAlignTool_StylusPresenceCheck** to **RunParams.UnfixturedFromFixturedTransform.TranslationY**.

   d) Configure the tool.

      i) Settings

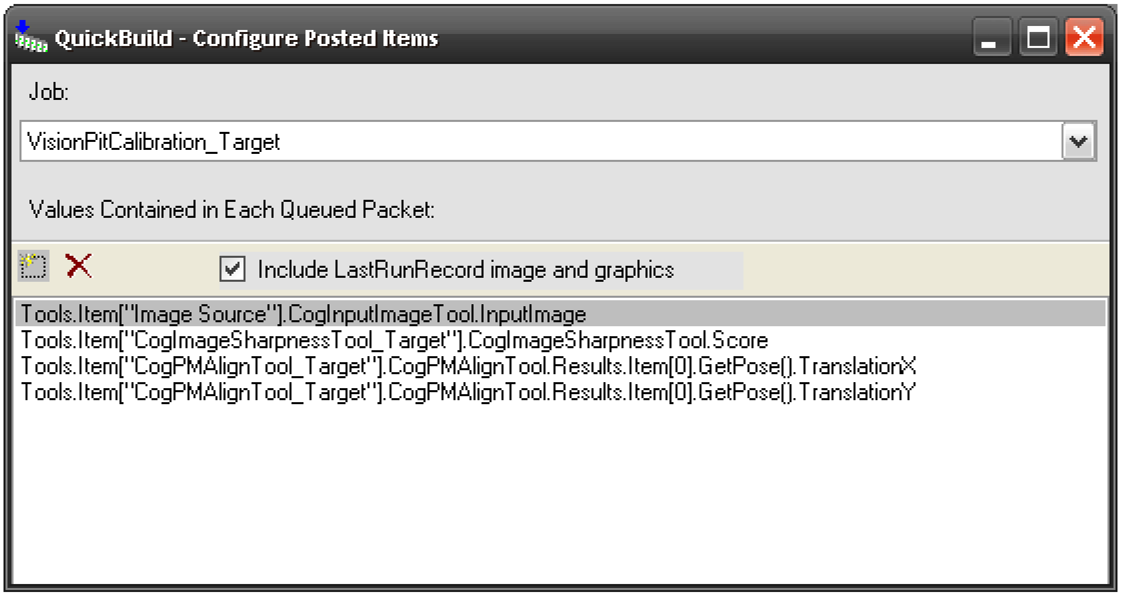
         (1) Fixtured Space -> Name: **Fixture_Stylus**

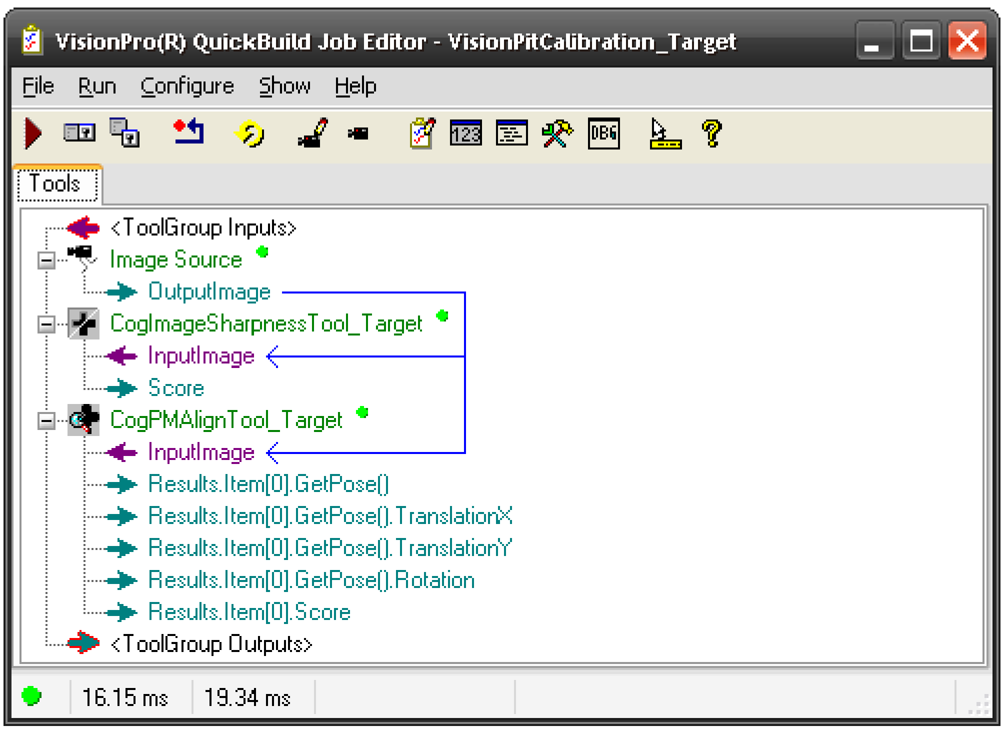
4) Add a **CogFindLineTool** and rename it as **CogFindLineTool_LeftStylusEdge**.
   a) Link **OutputImage** from **CogFixtureTool_StylusFixture** to **InputImage**.
   b) Configure the tool.
      i) Settings
         (1) Number of Calipers: 20
         (2) Search Length: 200
         (3) Projection Length: 5
         (4) Number to Ignore: 2
         (5) Selected Space Name: **@\Fixture_Stylus**
      ii) Caliper Settings
         (1) Edge 0 Polarity -> **Dark to Light**
         (2) Filter Half Size Pixels: 1

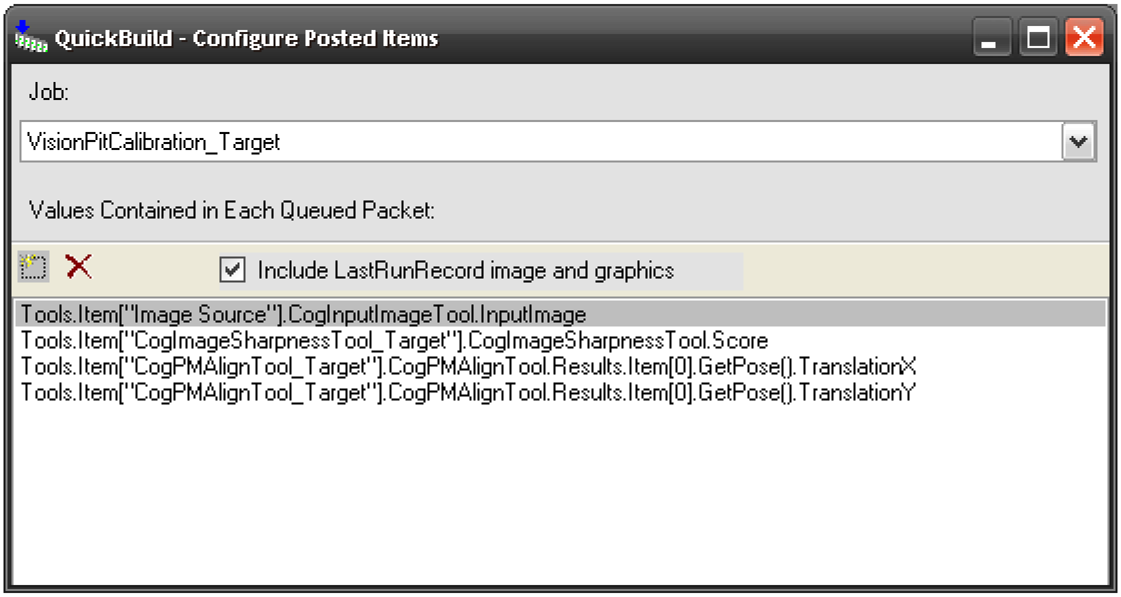


5) Add a **CogFindLineTool** and rename it as **CogFindLineTool_RightStylusEdge**.
   a) Repeat Step 4.
6) Add a **CogIntersectLineLineTool** and rename it as **CogIntersectLineLineTool_StylusReferencePoint**.
   a) Link **OutputImage** from **Image Source** to **InputImage**.
   b) Link **Results.GetLine()** from **CogFindLineTool_LeftStylusEdge** to **LineA**.
   c) Link **Results.GetLine()** from **CogFindLineTool_RightStylusEdge** to **LineB**.
   d) Add **X** to Posted Items.
   e) Add **Y** to Posted Items.
7) Add **RunStatus.Result** as an Output and add to Posted Items.

9.5.2 Target Camera Job Setup







1) Add **OutputImage** from **Image Source** to Posted Items.

2) Add a **CogImageSharpnessTool** and rename it as **CogImageSharpnessTool_Target**.

   a) Link **OutputImage** from **Image Source** to **InputImage**.

   b) Navigate to settings and configure as follows:

      (1) Mode: Autocorrelation

      (2) Region Shape: CogCircle

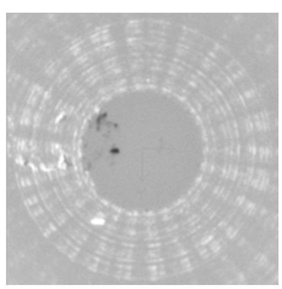
      (3) Radius: 80

   c) Add **Score** to posted items.   Score should be **0.3** or higher.

3) Add a **CogPMAlignTool** and rename it as CogPMAlignTool_Target.

   a) Link **OutputImage** from **Image Source** to **InputImage**.

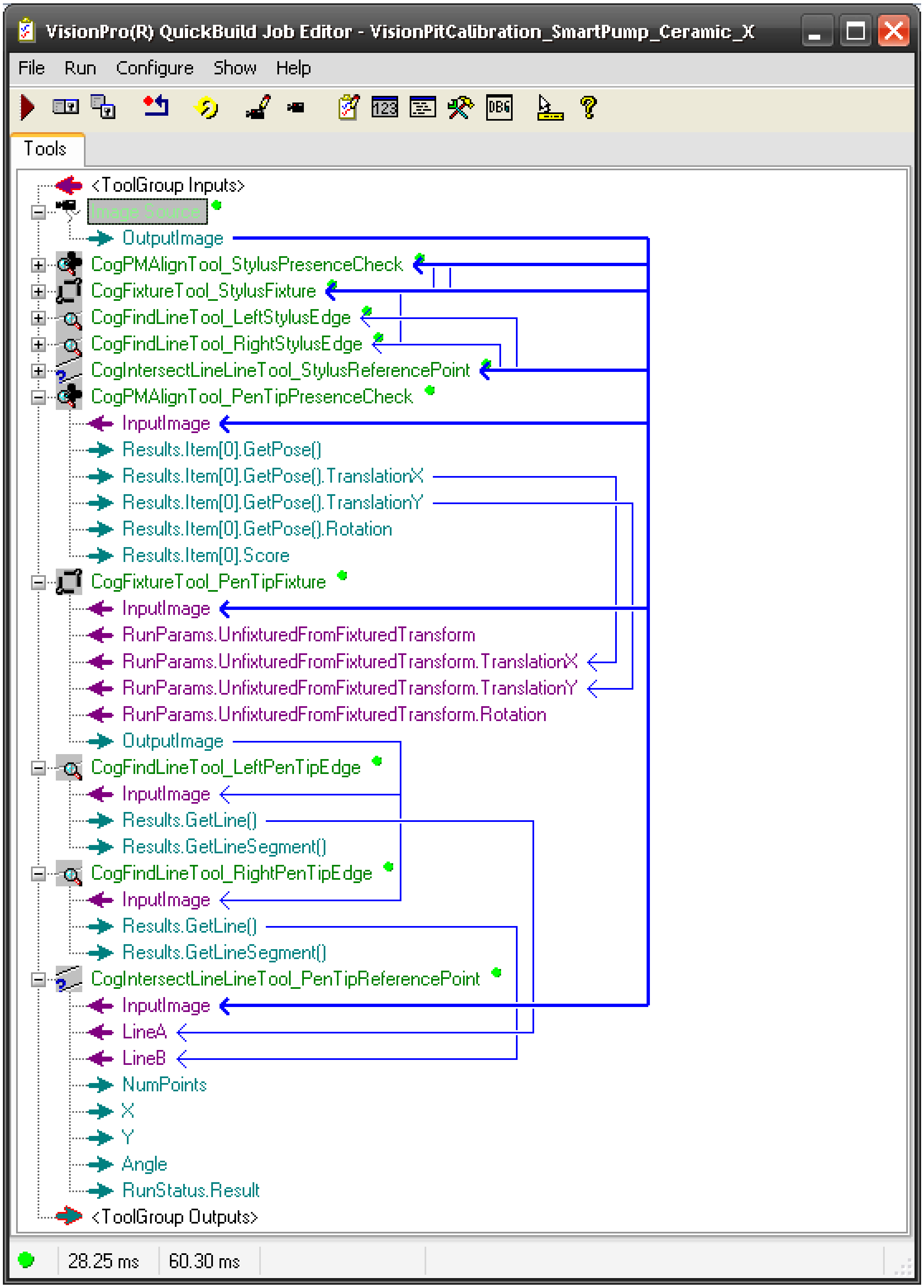
   b) Train the image. See the reference image below.

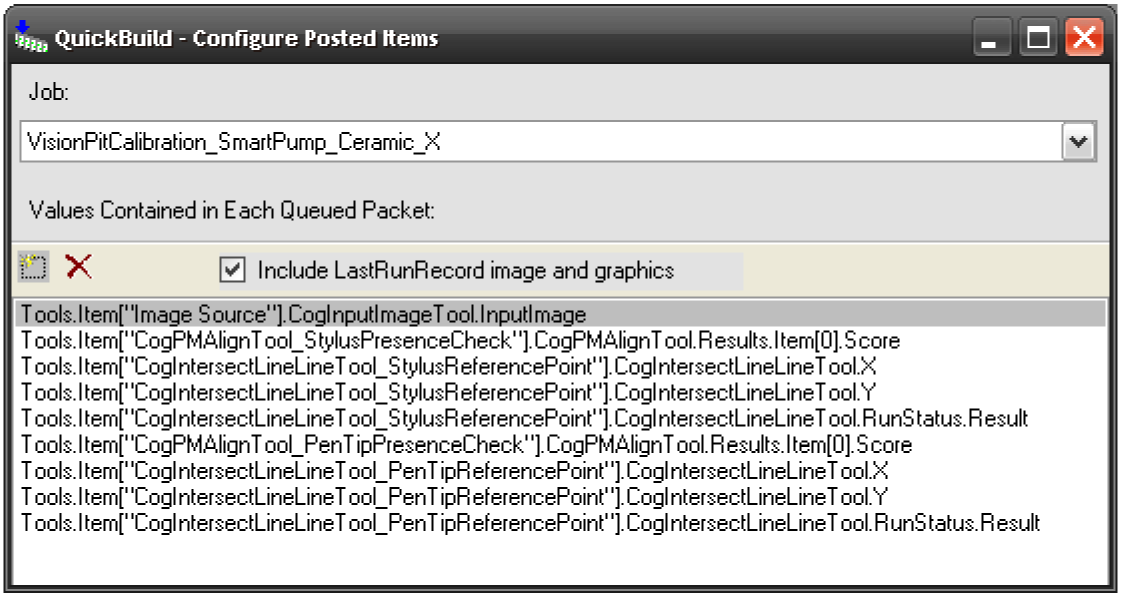


   c) Add **Results.Item[0].GetPose().TranslationX** to Posted Items.

   d) Add **Results.Item[0].GetPose().TranslationY** to Posted Items.

9.5.3 Pump Job Setup





1) Add **OutputImage** from **Image Source** to Posted Items.
2) Repeat Stylus job setup.
3) Add a **CogPMAlignTool** and rename it "**CogPMAlignTool_PenTipPresenceCheck**.
   a) Link **OutputImage** from Image Source to **InputImage**.
   b) Click train image.

   c) Add **Results.Item[0].Score** to posted items.
4) Add a **CogFixtureTool** and rename it as **CogFixtureTool_PenTipFixture**.
   a) Link **Image Source** to **InputImage**.
   b) Link **Results.Item[0].GetPose().TranslationX** from **CogPMAlignTool_PenTipPresenceCheck** to **RunParams.UnfixturedFromFixturedTransform.TranslationX**.
   c) Link **Results.Item[0].GetPose().TranslationY** from **CogPMAlignTool_PenTipPresenceCheck** to **RunParams.UnfixturedFromFixturedTransform.TranslationY**.
   d) Configure the tool.
      i) Settings
         (1) **Fixtured Space -> Name: Fixture_PenTip**
5) Add a **CogFindLineTool** and rename it as **CogFindLineTool_LeftPenTipEdge**.
   a) Link **OutputImage** from **CogFixtureTool_StylusFixture** to **InputImage**.
   b) Configure the tool.

i) Settings

(1) Number of Calipers: 25

(2) Search Length: 255

(3) Projection Length: 6

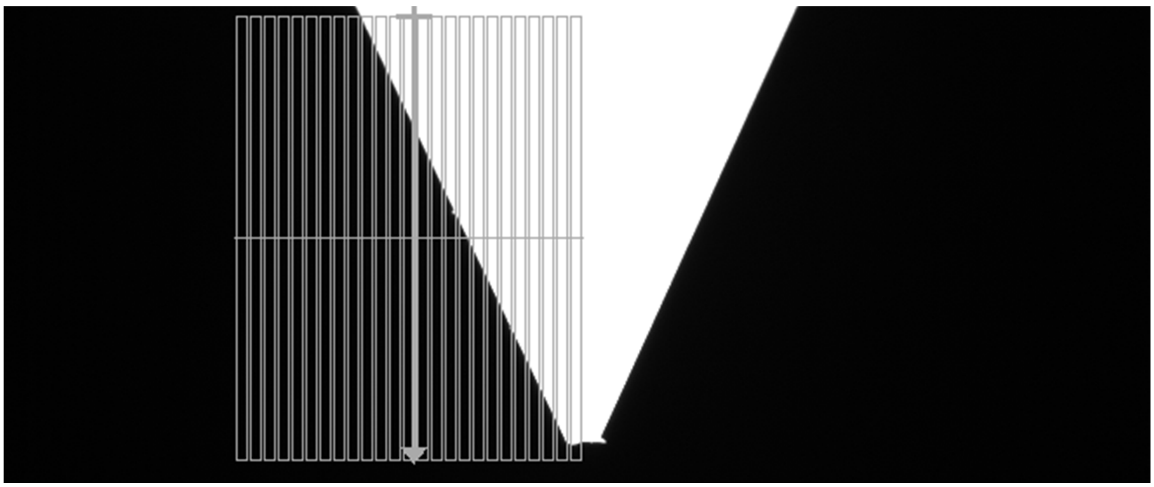
(4) Number to Ignore: 3

(5) Selected Space Name: **@\Fixture_PenTip**

ii) Caliper Settings

(1) Edge 0 Polarity -> **Light to Dark**

(2) Filter Half Size Pixels: 1



6) Add a **CogFindLineTool** and rename it as **CogFindLineTool_RightPenTipEdge**.

   a) Repeat Step 5.

7) Add a **CogIntersectLineLineTool** and rename it as **CogIntersectLineLineTool_PenTipReferencePoint**.

8) Link **OutputImage** from **Image Source** to **InputImage**.

9) Link **Results.GetLine()** from **CogFindLineTool_LeftPenTipEdge** to **LineA**.

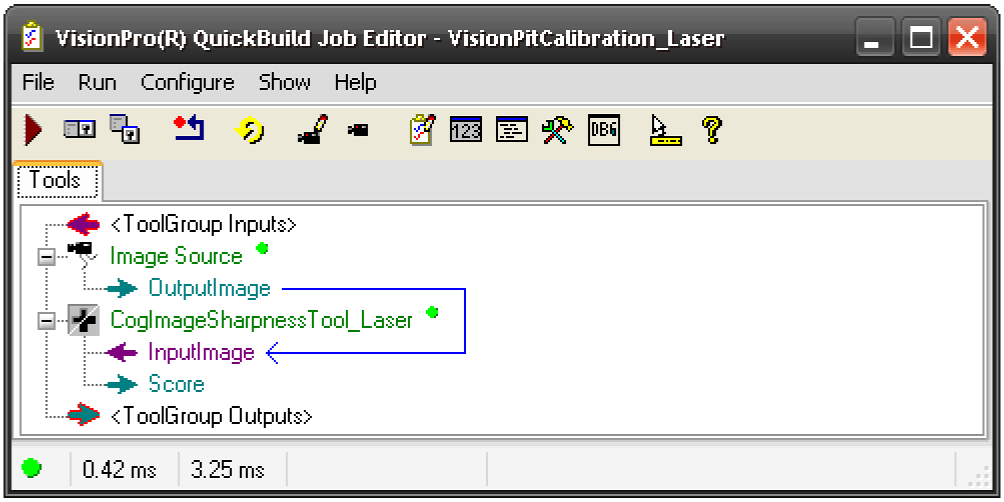
10) Link **Results.GetLine()** from **CogFindLineTool_RightPenTipEdge** to **LineB**.

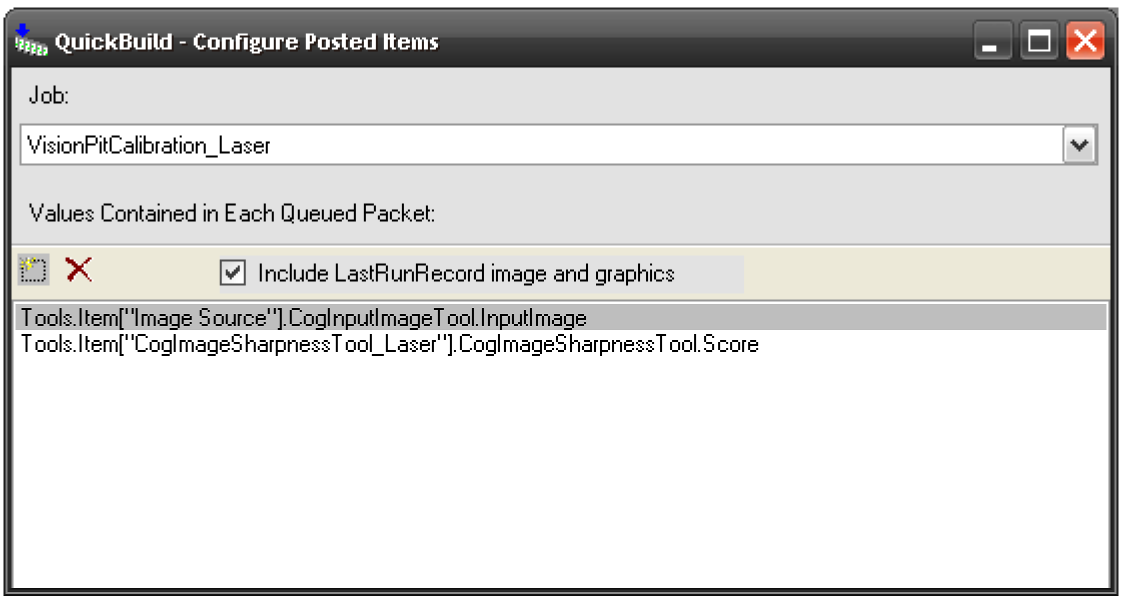
11) Add **X** to Posted Items.

12) Add **Y** to Posted Items.

13) Add **RunStatus.Result** as an **Output** and add to Posted Items.

9.5.4 Laser Job Setup





1) Add **OutputImage** from **Image Source** to Posted Items.
2) Add a **CogImageSharpnessTool** and rename it as **CogImageSharpnessTool_Laser**.
   a) Link **OutputImage** from **Image Source** to **InputImage**.
   b) Configure the tool.
      i) Settings
         (1) Mode -> **Autocorrelation**
   e) Add **Score** to Posted Items. Score should be 0.3 or higher.

10 Machine Settings

WARNING: Improper initial setup will result in problematic operation and may result in severe and costly equipment damage. Installation and initial setup of MTGen3 should only be performed by trained personnel.

In the machine settings area, machine managers can reconfigure the system. These settings are normally only adjusted when the hardware configuration of the machine changes (i.e., gizmos or accessories are added or removed).

10.1 Launching Machine Settings

1) Either open or reopen the software to bring up the mode dialog.
2) Launch **Machine Settings** from the dialog.

10.2 Saving New Machine Settings

Click **Save All Settings** then click **Close**.

10.3 Base Configuration (Base Config)

CalibrationLaserEquation – **ABSOLUTE** – Do not change.

CalibrationVisionFile – The location of the Vision Pro Persistence (*.vpp) file to be used during calibration. This vpp should only contain CogJobs for calibration.

FiducialVisionFile – The location of the Vision Pro Persistence (*.vpp) to be used in Application Mode. This vpp contains CogJobs for the Target camera to capture images and return measurements for automated fiducial detection.

MachineIsCalibrated – This value is **False** when new software is installed or when calibration fails. Once successful calibration is saved, **True** is displayed.

ManualCalibrationEnabled – Set this to true to perform manual calibration when the Calibrate button is clicked. Set this to false to enable automated calibration when the Calibrate button is clicked.

TextEditorFilePath – Allows users to change which text editor is used to edit script files from the print job editor. The default editor is NotePad.

WorldFloorValue – This floor value is obtained during calibration and is used to compute offsets between each gizmo, the substrate, and the floor (printing surface) in the machine.

BaseIOSettingsColl – Not used.

10.4 Gizmo Installation (Installed Gizmos)

The **gizmo selection list** on the left shows all available gizmos in your software version. When a new hardware component (gizmo) is added to the machine, that gizmo must be selected from the gizmo selection list on the left, and the **[>]** button must be clicked to add it to the **Installed Gizmos** list on the right. Use the **[<]** button to remove gizmos. To complete the installation of a new gizmo, continue to the slot configuration tab.

10.5 Slot Configuration (Slot Config)

Once a new gizmo is added to the machine, it must be assigned to an appropriate slot. Gizmos may only be assigned to their corresponding slot. SmartPumps™ may be assigned to any of the four slots: SlotA, SlotB, SlotC, SlotD

To assign a newly installed gizmo, under the appropriate slot click the **GizmoName: None** box to open the available gizmos list. Select the new gizmo from the list.

10.6 Main Camera (Target Camera)

CanActuate False – Not used.

ACTUATE – Not used.

AxisName – Not used.

AmplifierIndex – Not used.

HeadPosition – Not used.

Name – This parameter is set by which gizmo is assigned to this slot and cannot be changed.

GizmoType – This parameter is set by which gizmo is assigned to this slot and cannot be changed.

X0, Y0, Z0 – These are the positions obtained during calibration and are used to compute offsets between the stored gizmo and the master device.

MicronPerPixel – The resolution of the camera measured in microns per pixel. The specifications of the camera are used to determine this parameter. The default value is 9.9 um/pixel.

Magnification – The multiplier used to determine magnification specified on the side of the lens attached to the camera. The default value is 1.0.

MMPerPixel – Computed millimeter per pixel value based on the magnification and camera resolution.

GizmoConfigIO – Not used.

IsCalibrated – This value is **False** when new software is installed or when calibration fails. Once successful calibration is saved, **True** is displayed.

10.7 SmartPump™ Slots

CanActuate – Enables the slot to actuate the device down into is ready position. See the system documentation to find which slots on the tool plate use actuation.

ACTUATE – The digital output command used to actuate a device down into its ready position. Syntax for this command: $DO(Bit Index)_(Axis Name). See system documentation for the correct bit and axis used to actuate a given slot.

X Camera Cogjob Name – The name of the CogJob used to align the tip along the X axis found in the current Calibration vpp. This parameter is only used on systems with auto calibration.

Y Camera Cogjog Name – The name of the CogJob used to align the tip along the Y axis found in the current Calibration vpp. This parameter is only used on systems with auto calibration.

AxisName – This parameter cannot be adjusted (i.e., SlotA must always use axis A).

AmplifierIndex – This zero-referenced value indicates the number which corresponds to the axis index. The axis index is found in the system parameter file minus 1.

HeadPosition – This parameter cannot be adjusted. (i.e., Slot A is always head position 1).

Name – This parameter is set by which gizmo is assigned to this slot and cannot be changed.

GizmoType– This parameter is set by which gizmo is assigned to this slot and cannot be changed.

X0, Y0, Z0 – These are the positions obtained during calibration and are used to compute offsets between the stored gizmo and the master device.

ON / OFF – The digital output command used to **toggle material feed pressure on and off**. Syntax for this command: $DO(Bit Index)_(Axis Name). See system documentation for the correct bit and axis used to enable the material feed pressure for the given slot.

SET – The analog output command used to **adjust material feed pressure**. Syntax for this command: $AO(Bit Index)_(Axis Name). See system documentation for the correct bit and axis used to set the material feed pressure for the given slot.

SWITCH – The digital output command used to activate the pneumatic switch for material feed pressure. Syntax for this command: $DO(Bit Index)_(Axis Name). See system documentation for the correct bit and axis used to activate the pneumatic switch for the material feed pressure, if applicable.

GizmoConfigIO – Not used.

IsCalibrated – This value is **False** when new software is installed or when calibration fails. Once successful calibration is saved, **True** is displayed.

10.8 Laser Displacement Sensors (Primary/SecondaryLaserSensor)

CanActuate False – Not used.

ACTUATE – Not used.

AxisName – Not used.

AmplifierIndex – Not used.

HeadPosition – Not used.

Name – This parameter is set by which gizmo is assigned to this slot and cannot be changed.

GizmoType– This parameter is set by which gizmo is assigned to this slot and cannot be changed.

X0, Y0, Z0 – These are the positions obtained during calibration and are used to compute offsets between the stored gizmo and the master device.

GizmoConfigIO – Sets which analog input is use to **read the value from the displacement sensor**. To set this to analog input 1 on axis 2 (zero-referenced axis index), set **IOPort** to 2 and **IOBit** to 1.

IsCalibrated – This value is **False** when new software is installed or when calibration fails. Once successful calibration is saved, **True** is displayed.

10.9 UV Light

CanActuate False – Not used.

ACTUATE – Not used.

AxisName Not used.

AmplifierIndex – Not used.

HeadPosition – Not used.

Name – This parameter is set by which gizmo is assigned to this slot and cannot be changed.

GizmoType– This parameter is set by which gizmo is assigned to this slot and cannot be changed.

X0, Y0, Z0 –These are the positions obtained during calibration and are used to compute offsets between the stored gizmo and the master device.

ON / OFF – The digital output command used to **toggle the UV light on and off**. Syntax for this command: $DO(Bit Index)_(Axis Name). See system documentation for the correct bit and axis used to enable the UV cure light for the given slot.

SET – The analog output command used to **adjust the UV light intensity**. Syntax for this command: $AO(Bit Index)_(Axis Name). See system documentation for the correct bit and axis used to adjust the UV light intensity for the given slot.

GizmoConfigIO – Not used.

IsCalibrated – This value is **False** when new software is installed or when calibration fails. Once successful calibration is saved, **True** is displayed.

10.10 Process View Camera Trolley (Camera Trolley)

CanActuate False – Not used.

ACTUATE – Not used.

AxisName PV – This parameter cannot be adjusted. The PV axis must be used.

AmplifierIndex – This zero-referenced value indicates the number which corresponds to the axis index. The axis index is found in the system parameter file minus 1.

HeadPosition – Not used.

Name – This parameter is set by which gizmo is assigned to this slot and cannot be changed.

GizmoType– This parameter is set by which gizmo is assigned to this slot and cannot be changed.

X0, Y0, Z0 –These are the positions obtained during calibration and are used to compute offsets between the stored gizmo and the master device.

GizmoConfigIO – Not used.

IsCalibrated – This value is **False** when new software is installed or when calibration fails. Once successful calibration is saved, **True** is displayed.

10.11 IR Curing Laser (Aux Laser)

CanActuate False – Not used.

ACTUATE – Not used.

AxisName – Not used.

AmplifierIndex – Not used.

HeadPosition – Not used.

Name – This parameter is set by which gizmo is assigned to this slot and cannot be changed.

GizmoType– This parameter is set by which gizmo is assigned to this slot and cannot be changed.

X0, Y0, Z0 – Do not adjust these parameters. These are the positions obtained during calibration and are used to compute offsets between each gizmo on the machine.

Latch – The digital output command used to **latch the intensity setting**. Syntax for this command: $DO(Bit Index)_(Axis Name). See system documentation for the correct bit and axis used to latch the current intensity for the stored gizmo.

Master Oscillator – The digital output command used to **toggle the maser oscillator**. Syntax for this command: $DO(Bit Index)_(Axis Name). See system documentation for the correct bit and axis used to toggle the master oscillator for the stored gizmo.

Power Amplifier – The digital output command used to **toggle the power amplifier**. Syntax for this command: $DO(Bit Index)_(Axis Name). See system documentation for the correct bit and axis used to toggle power amplifier for the stored gizmo.

Pilot Light – The digital output command used to **toggle the red pilot light**. Syntax for this command: $DO(Bit Index)_(Axis Name). See system documentation for the correct bit and axis used to toggle the red pilot light for the stored gizmo.

Starting Intensity Bit – The digital output command for **the first power intensity bit**. Syntax for this command: $DO(Bit Index)_(Axis Name). See system documentation for the correct bit and axis for the first power intensity bit of the stored gizmo.

Intensity Bit Count 8 – The number of bits used in the power intensity bit mask. This value is assigned by the gizmo and cannot be changed.

GizmoConfigIO – Not used.

IsCalibrated – This value is **False** when new software is installed or when calibration fails. Once successful calibration is saved, **True** is displayed.

10.12 Lighting Control

CCS2Channel False – Not used.

CCS4Channel False – Not used.

Moritex1Channel False – Not used.

Moritex3Channel False – Not used.

LightingControllerType – Drop-down menu to select which lighting controller is installed on the system.

Lighting I/O – Not used.

11 Glossary

Actuation: uses pneumatics to move gizmos into a ready position.

APJ – See section 5.4 Automatic Print Job.

Auto cal: Automated calibration using a vision pit.

Auto print job – See section 5.4 Automatic Print Job.

Automatic print job – See section 5.4 Automatic Print Job.

Calibration eraser: Used for manual calibration, an eraser with a small target printed on it where each gizmo is referenced.

Cog job: A Cognex Vision Pro process used for vision detection. CogJobs are used for automated calibration to identify the gizmo position with respect to the conical and for fiducial find jobs to identify the actual fiducial location on the substrate.

CPJ – See section 5.5 Custom Print Jobs.

Custom Print Job – See section 5.5 Custom Print Jobs.

Dispense gap – The distance from the top of the substrate to the selected gizmo such as a pen tip.

FFJ – See section 6 Fiducials.

Target camera – A down-facing camera which focuses on substrates for post-print inspection and locating fiducial marks to accurately align prints.

Fiducial find job – See section 6 Fiducials.

Fiducial marks – Also called fiducials. Small crosshairs or other marks on substrates used for aligning future print layers.

Four point correction – See section 7.3 Planar Correction.

Gizmo – Any device which is attached to the machine's toolplate which performs a particular function.

GMJ – See section 7.6 Grid Mapping.

Grid mapping job – See section 7.6 Grid Mapping.

Height – the distance from the floor along the Z axis.

Job - A set of tasks grouped into a single procedure. A job holds processing information and modifies the work piece it is attached to.

Length – the dimension of an object along the Y axis; the length of a printed line.

Library file – See section 4.6.4 Library File Select.

Line thickness – The Z height of a printed line of material. See **line width** for comparison.

Line width – The X or Y width of a printed line of material. See **line thickness** for comparison.

Manual print job – See section 5.1 Manual Print Jobs

MPJ – See section 5.1 Manual Print Jobs

MTGen3: Machine Tools Generation 3. nScrypt software for machine control.

Path mapping job – See section 7.5 Path Mapping.

PCJ – See section 7.3 Planar Correction.

Planar correction job – See section 7.3 Planar Correction.

PMJ – See section 7.5 Path Mapping.

Print script job – See section 0

Print Script Jobs.

PSJ – See section 0

Print Script Jobs.

Rotational script job – See section 8 Rotary Stages.

RSJ – See section 8 Rotary Stages.

Script or script file – Text file with ".txt" extension containing printing pattern and parameters.

Slot – A location on the tool plate where a gizmo is attached.

Thickness – May refer to substrate thickness or line thickness.

Tool Plate – A tool plate is a fixture with slots where gizmos are mounted.

Vision pit – A hardware system composed of a conical and two orthogonal cameras used for calibration. The vision pit may be mounted to a motorized elevation platform or used as a portable deck for manual installation in the machine.

Width – the dimension of an object along the X axis; the width of a printed line.

Z map – The set of data obtained from the laser displacement sensor. The z map represents an actual conformal surface and may be used to modify the path of a gizmo so as to follow that surface accurately.

12 Appendices

12.1 Appendix 1: Hardware

The Aerotech A3200 motion controller is used to operate the mechanical stages, motors, pumps, and system accessories. The motion control system is connected to the onboard computer via 1394 Firewire. This connection facilitates manual operation and script downloading.

Due to the wide array of options available, please contact nScrypt for detailed information on the various hardware components of the machine.

12.2 Appendix 2: Software

12.2.1.a MTGen3

MTGen3 provides the operator interface for nScrypt dispensing machines. Features include: automation, process management and development, interfacing with the motion controller, data collection, system configuration, and the calibration of system devices.

12.2.1.b PathCAD

PathCAD software is provided to allow graphical script authoring or importing from DXF files. PathCAD allows the user to order the lines and set the print direction of each line. PathCAD exports the drawing file as a series of movement commands and automatically adds additional commands to turn on and off the flow of material.

12.2.1.c Keyence LK Navigator

LK Navigator is used to tune and configure the laser displacement sensor.

12.2.1.d Cognex VisionPro QuickBuild

Cognex's vision application management tool used to create and configure image processing procedures.

12.2.1.e Microsoft Notepad

Notepad is a text editor useful for taking notes, saving coordinate values temporarily, and editing script files.

12.2.1.f AeroTech NView

NView is Aerotech's operator interface to the motion controller. NView is used for system hardware testing, diagnostics, and manual control of the motion stages.