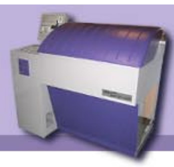
Python Installation Guide v5.0



Preface

Version 5.0	November 2006	(for software version 5.0.0, incorporating the Python Console v1.0.1)
Version 4.2	July 2005	(for software version 4.0.0 SP1/2)
Version 4.1	December 2004	(for software version 4.0.0 SP1)
Version 4.0	September 2004	(for software version 4.0.0)
Version 3.4	March 2004	

HighWater's contact information

HighWater Designs Limited
1-6 St. George's Business Park
Alstone Lane
Cheltenham
Gloucestershire
UK
GL51 8HF

Telephone: +44 (0) 1242 542100

Fax: +44 (0) 1242 251600

Email: sales@highwater.co.uk

Web: www.highwater.co.uk

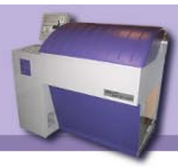
HighWater support department

Telephone: +44 (0) 1242 542102

Email: support@highwater.co.uk

Web: www.4hws.co.uk

Contents



1. Introduction

This chapter includes the following sections:

- 1.1, Before you begin (p9).
- 1.2, Introduction to this manual (p9).

1.1 Before you begin

Warning: You must read this manual before starting the installation and, in particular, read the safety information in the next chapter.

Please read the following information before you begin:

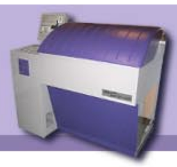
- **This manual refers to version 5.0.0 and later of the Python software (incorporating the first release of the Python Console — v1.0.1).**
- Ensure that the room the Python will be situated in has the necessary space for access to the Python engine. We recommend a 60cm access space on both the left and right-hand sides (40cm minimum) and 90cm in front of the Python engine. Please see Appendix B for more brightroom information.
- Read the plate and working environment information in Appendix A and Appendix B. These appendices tell you what plates can be used with the Python system and the conditions that the Python system should be operated in.
- This manual assumes that an approved, fully functional plate processor is available for development of the plates.
- Please carefully read all warnings in this manual and follow any given instructions.

1.2 Introduction to this manual

This manual provides full installation instructions for the Python CtP system and covers:

- **Safety notices and requirements** — this information is very important and you must ensure that you fully understand and abide by the safety requirements before starting the installation.
- **Unpacking and moving the Python engine.**
- **Connecting the workstation to the Python engine.**
- **Connecting the Python engine to a power supply.**
- **Attaching the dongle to the workstation and enabling the Python software.**
- **Loading and unloading plates on the Python tilt-table.**
- **Producing test plates and carrying out calibration and quality checks.**
- **Plate information.**
- **Brightroom layout and specification.**
- **Safety label information.**
- **Test results.**

This guide does not fully cover setting up and using the Torrent RIP and other Python software. Once you have installed and tested the Python engine, you should refer to chapter 11 of this manual and also to the **Python User Guide** for more details on setting up the software for the user.



2. Safety information

Warning: All engineers must read this section and abide by the safety requirements when they start work on the Python engine. You should also read the supplied Python Safety Information manual.

This section includes the following information:

2.1 Safety information

Python, under normal operation, is a **CLASS 1** embedded laser product and is inherently safe for operators because they cannot operate the laser if the cover is open. Service engineers will be able to override the safety interlock system on the cover and may be exposed to the laser, which is a **CLASS 3B** product. This chapter details all the necessary precautions you should take when working on the Python engine.

All installation/service engineers must be trained and approved by HighWater Designs Limited, and must follow all safety procedures defined in this document.

HIGHWATER DESIGNS LIMITED WILL ACCEPT NO RESPONSIBILITY FOR ANY DAMAGE TO EQUIPMENT OR INJURY TO THE PERSON CAUSED BY NON-APPROVED ENGINEERS OR BY FAILURE TO FOLLOW THESE SAFETY PROCEDURES.

2.1.1 Safety warning

Installation and servicing should be carried out with the interlock system intact, wherever possible. Safety interlocks should only be overridden where this is absolutely necessary as part of the recommended procedures.

Access to the internal mechanisms is gained by removing the back panel. This, and all other panels, must be replaced and the mains lead reconnected before leaving the equipment.

With the safety interlocks overridden, the machine covers in place and the lid open, beware of the 'pinch' risk from moving mechanical parts. Take care when working around the tilt-table area with the machine in this state.

2.2 Working safely

2.2.1 Protecting yourself from injury

To protect yourself from injury:

- Make sure that nothing, especially your clothes, gets trapped in Python's lid.
- Follow all safety warnings and instructions given in this manual and in the Python Test Tool software.
- The edges of a plate are sharp so take great care when handling plates and, in particular, keep the edges of the plate away from your face.

2.2.2 Protecting the Python drum and carriage from damage

To protect the Python drum and carriage from damage:

- Do not put any object other than a plate on the tilt-table. When the imaging process starts, anything left on the tilt-table will fall down into Python's drum. This could damage Python's drum or carriage, or the plate.

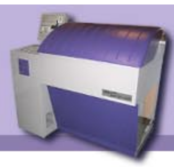
2.2.3 Safe plate handling

Plates must be handled with care. Engineers need to be aware of the following guidelines:

- Plate edges are sharp, particularly the corners. Remove plates from their packaging carefully and keep plate edges and corners away from your face.
- Handle plates at their edges.
- Contact with human skin can damage the plate's emulsion surface.
- Keep the interleave paper on the emulsion side of the plate during handling, but remove it before placing the plate onto the tilt-table.
- Always put the plate onto the tilt-table emulsion side up.

2.3 Safety labels

Please refer to Appendix C on page 109 for details of the safety labels on the Python engine.



2.4 Laser precautions

Python has a single laser source for the plate expose beam: a blue-violet laser diode. It is rated at 60mW (some early machines used a 30mW or 5mW diode) and emits a visible light beam at a nominal wavelength of 405nm. This laser falls into **CLASS 3B**, which means that direct viewing of the laser beam is always hazardous, as is viewing direct reflections. Diffuse reflections are normally safe.

Python is designed as a **CLASS 1** embedded laser product and is, therefore, safe for the operator. The **CLASS 3B** ratings apply as soon as an installation/service engineer removes the safety interlocks. This can only be done with a special magnetic key which allows the laser shutter to be opened and the expose carriage to be moved with the Python lid in the up position, or when the back panel is removed for servicing and the machine is powered on.

Warning: CLASS 3B lasers are dangerous and, if the beam is viewed incorrectly, may cause irreversible retinal damage or blindness. Therefore, you must, where appropriate:

1. **Ensure that you, and those working with you, have adequate eye protection against the laser.**
2. **Ensure that no unauthorized personnel have access to the work area when the laser is exposed.**
3. **Ensure that no laser light escapes from the work area.**
4. **Ensure that the room is clear of unauthorised personnel while you are working on Python.**
5. **Ensure that appropriate warning signs are displayed on all doors leading to the work area.**
6. **Leave Python in a safe state when it is left unattended.**
7. **Ensure that the safety interlock bypass key is removed from the unit when installation/servicing is complete or when you leave the work area.**

   At no time should the safety interlock bypass key be fitted to the machine whilst the engineer is not present in the work area.
8. **Keep the bypass key in a secure place.**
9. **Replace and secure all panels before leaving the Python work area.**

---

3. Unpacking and moving the Python system

This chapter shows you how to unpack the Python system from its crate and move it into position in the brightroom. It contains the following sections:

- 3.1, Tool requirements (p15).
- 3.2, Crate contents at delivery (p15).
- 3.3, Unpacking the Python system (p16).
- 3.4, Moving the Python engine (p20).
- 3.5, Removing the transit bracket (p21).
- 3.6, Bridge/processor connection (p22).

3.1 Tool requirements

The installation procedure does not require any specialised equipment. However, the following tools are recommended for unpacking, installation and testing:

- Set of metric Allen keys (for example, 1.5, 2.0, 2.5, 3, 4, 5, 6, 7, 8, 9 and 10mm).
- Set of metric spanners (for example, 4, 5, 5.5, 6, 7, 8, 9, 10, 12, 13 and 14mm).
- Screwdrivers: flat head and cross point (pozidrive).
- Plate densitometer for density calibration checks.
- Steel metric ruler (1 metre long).
- Magnifying glasses (x10 and x100).

3.2 Crate contents at delivery

A full Python system is shipped in a single crate containing:

- The Python CtP engine. The computer system, monitor, keyboard and mouse are shipped in situ inside the Python PC enclosure with packing material around them (this should be removed during installation).
- A box containing the Python software and cables.

Note: If the customer has ordered a two-computer Python system, the second computer will be shipped in its own box, separate from the main crate.

The main crate dimensions are approximately 161 x 110 x 142cm (w x d x h).

The total weight of Python in its crate is approximately 410kg. Python, uncrated and fully assembled, weighs 300kg.

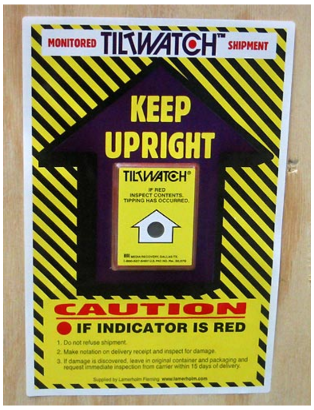
Warning: You must take extreme care during lifting and manoeuvring operations, and make sure that the appropriate lifting equipment and an adequate number of people are used to move and install the Python system.

3.3 Unpacking the Python system

3.3.1 Examining the shock indicators

Before shipping, a number of 'shock' indicators are attached to the Python crate and to the wrapped Python engine. These indicators tell you if the system has been subjected to excessively rough handling during transit from the factory to the customer site.

There is a 'tilt' indicator on both the inside and outside of the crate. On the outside of the crate there may also be a 'Shockwatch' indicator:



Tilt indicator

Shockwatch indicator

When these indicators are activated they turn red showing that the crate has been subjected to excessive tipping and/or an impact of some sort. However, this does not necessarily mean that the Python engine has been damaged.

There are a number of Shockwatch indicators on the Python engine itself — normally 25g, 50g and 75g — to indicate the force of any impact the Python engine has been subjected to. When these indicators are activated they turn red, in which case some damage may have occurred to the Python engine.

If any of the tilt or Shockwatch indicators have been activated (that is, if they have turned red) then follow the procedure set out on the label, thus:

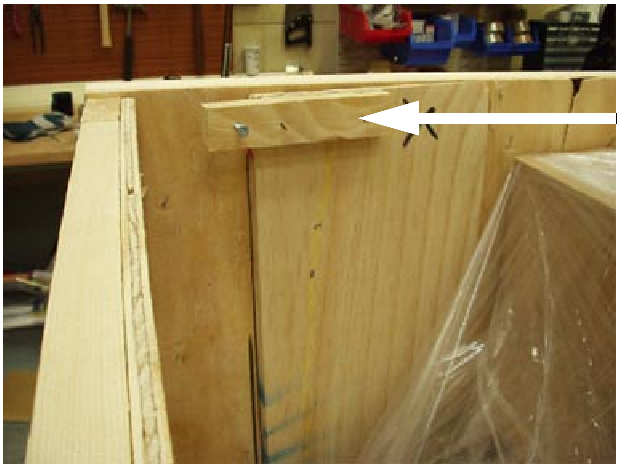
1. **Do not refuse shipment of the Python system**.
2. Make a note on the delivery receipt that the indicator(s) have been activated and inspect the Python system for any obvious damage.
3. If any damage is discovered, leave the Python system in its original container and packaging and then request immediate inspection from the carrier within 15 days of delivery (or 3 days international). Contact HighWater Designs (or other supplier) to further discuss any problems.

3.3.2 Unpacking the Python system

When you are ready to unpack the Python system:

1. Site the crate in a suitable space such that there will be enough room:
   - To remove the crate lid and place it on the floor next to the Python.
   - To move the Python off the pallet from the side marked 'Ramp This End'.

2. Remove the screws securing the top to the four sides then take the crate lid off. (Later, you will invert the lid to use as a ramp for Python to be wheeled off the pallet base.)
3. Locate the crate side on which the ramp floor piece has been attached. In the photo below, it is located on the left-hand crate side panel:



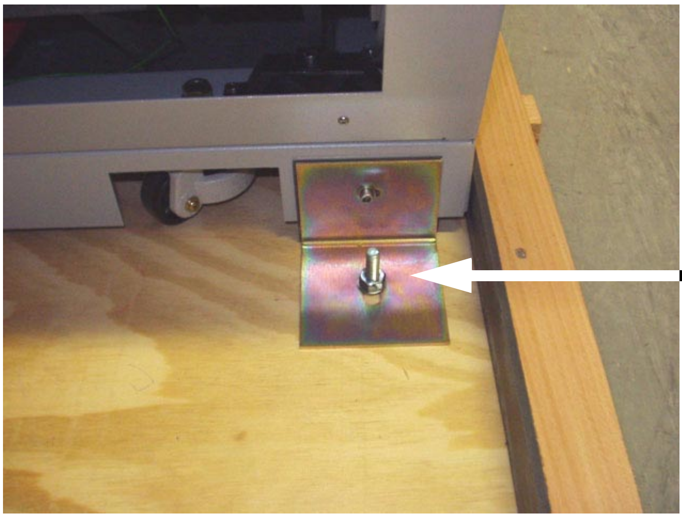
Remove the ramp floor piece

4. Take this side panel off and remove the ramp floor piece by removing the screws marked with a cross and put them to one side.
5. Remove the other three side panels to leave Python on its pallet.
6. Move the four side panels away from the work area.
7. Carefully remove the clear plastic wrapping material from around Python.

8. Remove the separately packed panels, along with their packaging, and the boxes containing the computer system peripherals and put these safely to one side.
9. Place the inverted crate lid onto the floor to the left of the pallet (when looking from the front of the Python). This will act as the ramp. Move the ramp up against and level with the pallet:

Crate lid inverted

10. Remove the metal bracket from the front-right of the Python by removing the securing screws:

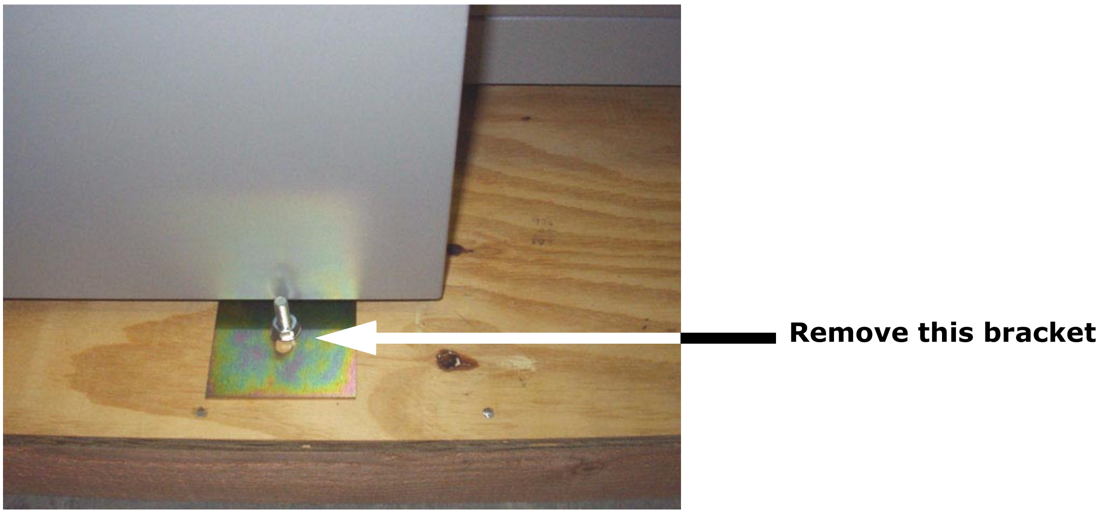


Remove this bracket

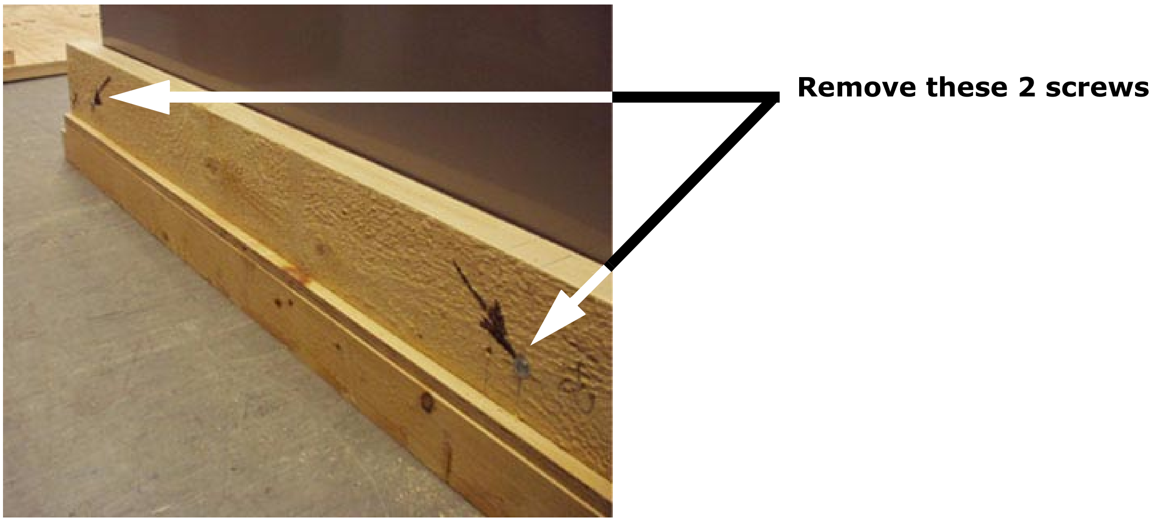
11. Remove the ramp-end cross piece by removing the securing screw and bolt head screw.

Note: Keep the bolt head screw for the next stage.

12. Remove the metal bracket from the front-left of the Python by removing the securing screw:



13. Attach the ramp floor piece to the other end of the ramp using the bolt head screw removed from the ramp-end cross piece.
14. Remove the two screws marked with a cross that secure the machine to the wood block running the length of the pallet:



15. Remove the wooden block.

The Python engine is now uncrated and can be moved into position, as described next.

Note: If the customer has ordered a two-computer Python system, also unpack the second computer from its box.

3.4 Moving the Python engine

Now that the Python engine is uncrated:

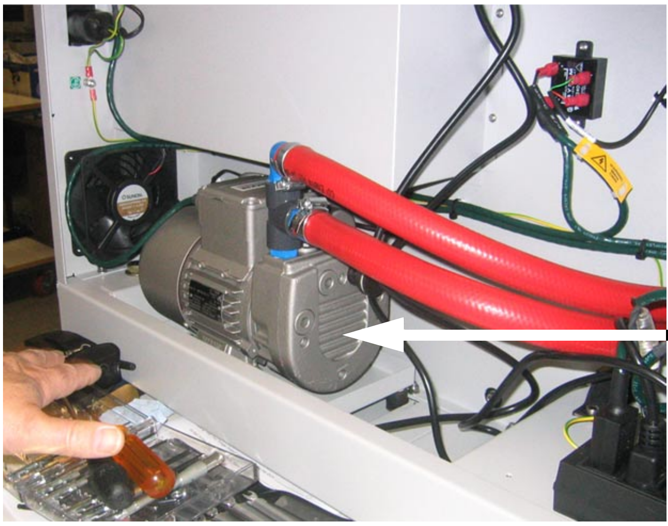
1. Unlock the machine's wheels by rotating the four orange-coloured cogs such that the floor mounting pads are raised up fully. Python is now free to move.
2. Wheel Python off its pallet and manoeuvre it into position in the brightroom.

   Note: Ensure that there is enough access space around the Python engine. We recommend a 60cm space on both the left and right-hand sides (40cm minimum) and 90cm in front of the Python engine. Please see Appendix B for more access and brightroom information.
3. When Python is correctly positioned, lift the wheels by rotating the orange toothed cog just above each wheel. This will lower a pad to the floor. Python will sit on these four pads.

   Note: Levelling the Python will be necessary if the floor is very uneven (evidence of this is uneven seating of the Python lid when closed and the lid switch will not operate correctly). If the Python engine requires extra levelling, then re-site the machine to a more even piece of flooring, if possible (often this should suffice). If you cannot rectify the problem call a support provider or HighWater Designs.

Once the Python engine is in the correct position, continue with the installation as follows:

1. Clean the outer surfaces of the Python and clean the area inside the lid.
2. Remove the left-hand end panel by loosening the 8 securing screws.
3. Remove any packing that may be supporting the vacuum pump and place it on the tray in the cut-out in the Python casing:



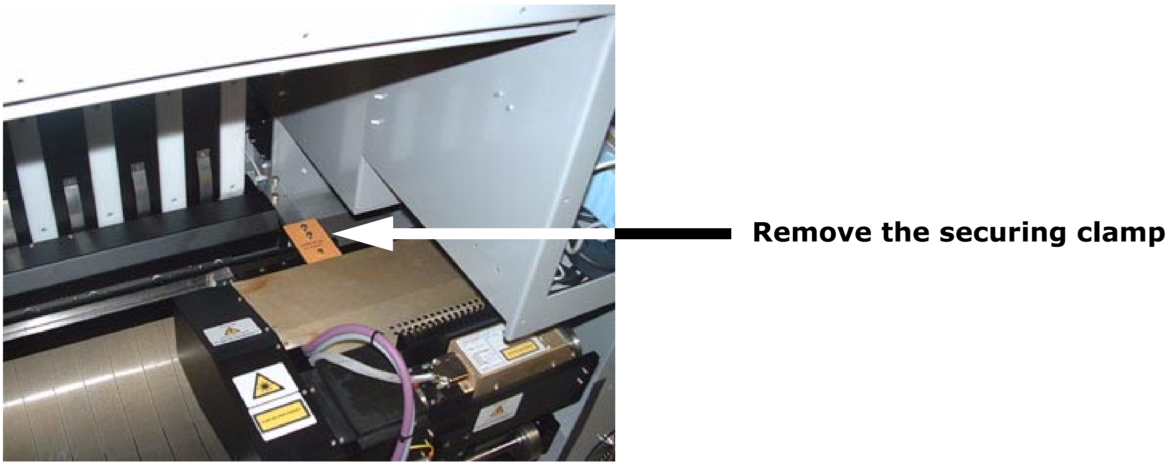
3.5 Removing the transit bracket

The Python engine is shipped with the optics carriage clamped and the tilt-table hinged downwards. Removing the transit bracket allows both these assemblies to be unclamped and prepared for when the power is switched on.

Warning: DO NOT switch Python on while the carriage transit bracket is in place.

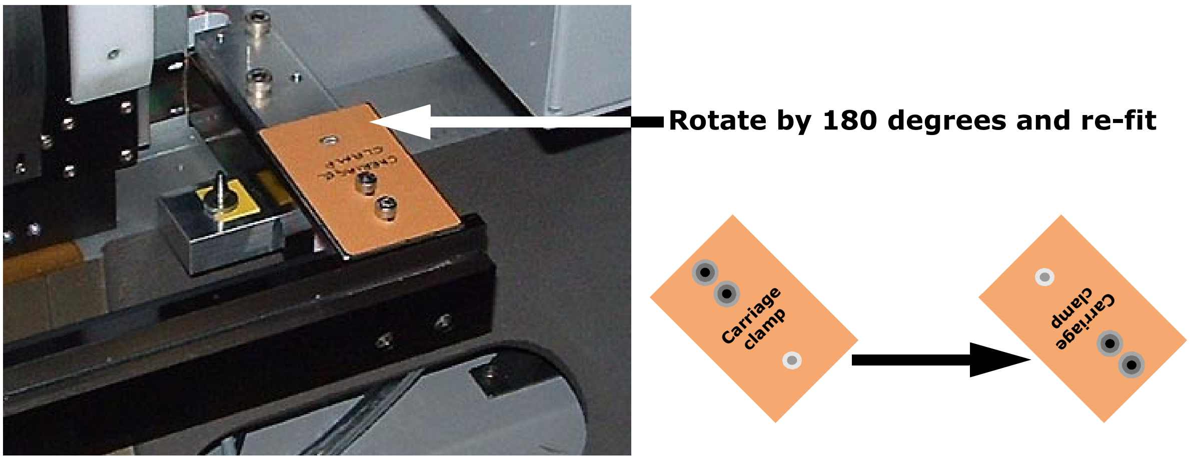
To remove the transit bracket:

1. Remove the back panel of the Python engine by releasing the 24 screws that secure it to its frame.
2. Locate and remove the orange coloured optics carriage securing clamp:



Note: The screws are two different sizes.

3. Rotate the clamp by 180 degrees, as shown below, and fit it back onto the drum chassis, where it can be kept safe. Replace and tighten the two larger screws:



4. Replace the smaller screw into its original hole and tighten it.

5. Now, carefully check the tilt-table and drum for debris, and vacuum clean both thoroughly, if necessary.

   Note: When Python is switched on and a test image is sent to the engine, the carriage automatically finds its start position and the tilt-table/crossbar is driven to its loading position, ready to accept a plate (providing the cover is closed).

3.6 Bridge/processor connection

If a bridge/processor connection is required, then also carry out the following modifications to the Python engine:

1. Remove the cover from the letterbox opening on the rear of the Python by removing the four securing screws:

Remove this cover

2. Once the cover is removed, site the Python, bridge and processor accordingly.
3. Next, loosen the two screws on either side of the plate exit ramp and adjust the ramp's angle to ensure the smooth movement of the plate from the tilt-table to the bridge/processor.

4. Connecting up the Python system

After unpacking the Python system and moving it into place, you next need to connect the workstation to the Python engine, and then connect the system to a power supply. This chapter contains the following sections:

- 4.1, Connecting the workstation to the Python engine (p23).
- 4.2, Connecting the Python engine to a power supply (p25).

4.1 Connecting the workstation to the Python engine

Unpack the workstation and connect it to the Python engine, as follows:

1. The Python workstation has been shipped inside the Python assembly. It is fixed in place with packing material, which should be removed at this stage.
2. Unpack the monitor, keyboard, mouse, cables and optional barcode reader, that were removed from the crate earlier, and connect them to the workstation.

   Note: Connect the barcode reader in-line with the keyboard.
3. The computer system is shipped with the internal data connections (the CAN bus) already made:

CAN bus cable

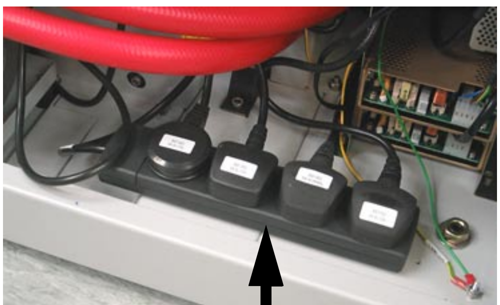
4. Check that the CAN bus cable from the PC is connected to the 9-way D-type connector marked ‘PC’:

9-way D-type connector

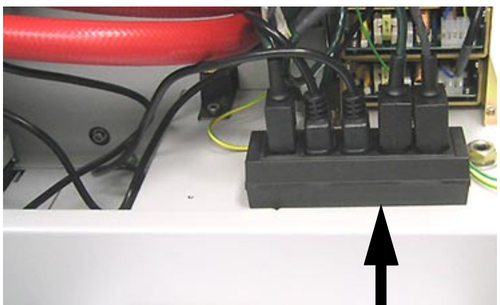
5. Position the monitor, keyboard, mouse and optional barcode reader on top of the Python user operating area to the left of the machine:

6. Route the cables through the circular cable access port and connect them to the workstation.

7. Check that all the mains plugs are plugged into the distribution board (this is situated next to the vacuum pump):

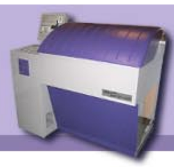


240V distribution board



110V distribution board

Note: There are separate plugs for the PC monitor, PC workstation, vacuum pump and Python power supply.



8. Connect the customer's CAT5 network cable to the workstation.
9. If the customer ordered a two-computer Python system, connect the second computer (running the Torrent RIP) into the network.

   Note: You can connect this computer anywhere on the network.

4.2 Connecting the Python engine to a power supply

Warning: Before powering up the Python system, make sure that the transit bracket has been removed, as described in section 3.5 on page 21.

This equipment MUST be earthed.

Python requires a single power outlet, accessible at all times and located as close as possible to the Python system. Python is rated at 230 volts 6 amps, or 110 volts 10 amps, and has a maximum power consumption of 1265W.

4.2.1 Connecting the Python system to a power supply

To connect the Python system to a power supply:

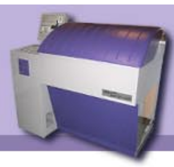
1. Connect the supplied power cable to the power inlet on the back of the Python cabinet.
2. Check that there are no loose cables, and that all the cables are plugged in firmly.

If the local mains power supply is 110V, the PC's Power Supply Unit (PSU) will have a label marked either 'FULL RANGE' or '110V' (depending on supplier). This label will be absent for 230V operation.

For the vacuum pump to be configured for 110V working, a '110V' label will be mounted horizontally on the top of the unit. Also, the UL file number (E123665) will be visible on a vertically and centrally mounted metal plate.

4.2.2 Disconnecting power completely from the Python

To disconnect power completely from the Python engine, remove the AC power cord from the wall outlet.



5. Starting up and shutting down the Python system

This chapter includes the following sections:

5.1 Starting up the Python system and logging on

To start up the Python system and log on:

1. Switch on the Python platesetter using the switch on the back. The system, including the front-end computer, will boot up automatically.
2. The Python software, which runs under Windows 2000, has two logins:

Login	User name	Password
Engineer This login lets you run the Python Test Tool software (in engineer mode), which you use to perform standalone tests on Python’s sub-systems and carry out the quality/calibration checks.	**engineer**	**highwater**
Customer/end-user login	**python**	*no password*

3. Log in as **engineer** with the password **highwater**.

   Warning: Do not disclose the engineer login password to anyone, especially end-users, as this login provides access to test facilities, which (if used improperly) could cause personal injury or irreparable damage to the Python platesetter.

4. The Python Test Tool software will launch automatically.

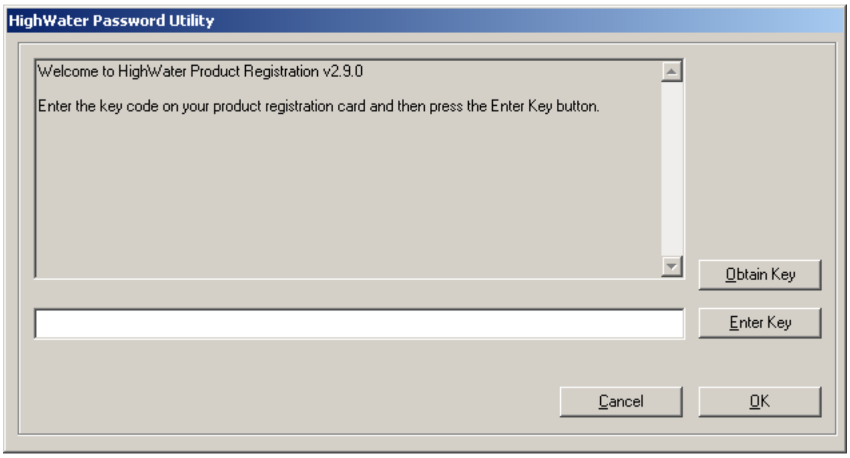
   Note: If the Python Test Tool does not launch, you may need to enable it first, as described in the next section.

5.2 Enabling the Python software

All the Python software that the user requires has been pre-installed onto the Python workstation. Before running the Python software for the first time, however, it needs to be enabled using HighWater's Password Utility program. To do this:

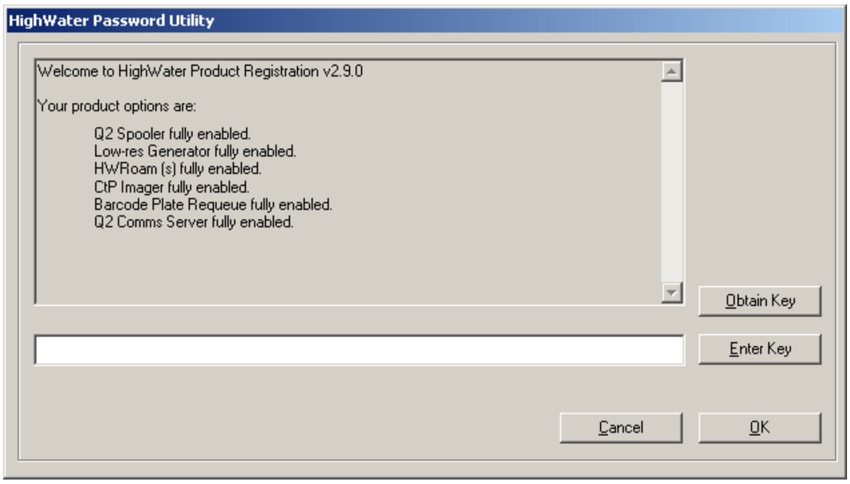
1. If you have not done so already, attach the supplied dongle to the USB port of the machine. You must do this **now**, before you launch the HighWater Password Utility program.

2. When the dongle is attached to the machine, select **Start > Programs > HighWater Designs > Q2vx.x.x > Register Q2** to display the following dialog:



3. Type the supplied keycode (either the sequence of numbers or words) into the white text box at the bottom, then click on the **Enter Key** button.

4. If the keycode is valid, you will see the enabled software applications listed in the main window, for example:



5. Click on the **OK** button to exit the HighWater Password Utility program.

You will now be able to run the Python software applications.

5.3 Shutting down the Python system

Later, when you are ready to close down the Python system:

1. Close down all running applications.
2. Shut down the Windows operating system by selecting **Shutdown** from the **Start** menu.
3. When the computer system has closed down, Python can be switched off at the back of the machine.

6. Initial checks and setup on the Python engine

Before going through the plate loading sequence and carrying out the plate tests, you need to carry out some initial checks on the Python engine, which are described in this chapter.

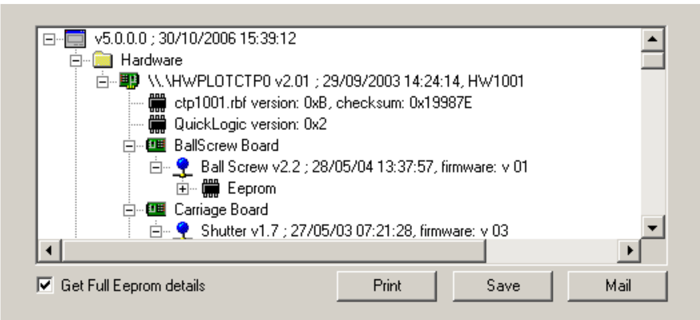
This chapter covers:

- 6.1, Checking the Python system components (p31).
- 6.2, Cleaning the Python tilt-table and drum (p33).
- 6.3, Enabling image clipping (p36).
- 6.4, Setting up email (p37).

6.1 Checking the Python system components

Before you can run through the cleaning procedure and installation tests you should make sure that all the system components are operating correctly. To do this:

1. Launch the Python Test Tool by selecting **Start > Programs > HighWater Designs > Python vx.x.x > Python Test Tool**.
2. In the Python Test Tool, click on the **System Tests** tab and select **System Status** from the drop-down menu.
3. Click the **Run Test** button. After a short time (usually less than 10 seconds), the results of the test are displayed, for example:



4. Each of the Python engine's sub-systems is shown with its corresponding icon:

	An application program. This is a file with the .exe extension.
	A folder. This may not necessarily exist on the computer as it may be used to head a given category, such as 'hardware' or 'software'.
	This indicates hardware in the host (front-end) PC. This is usually the controller board for the CTP engine. There may be more than one, depending on how many CTP engines are connected to.
	This indicates hardware inside the CTP engine. Each board in the CTP engine may carry one or more CAN devices.
	This indicates a memory device or other programmable device within the system.
	This indicates a CAN bus component within the CTP engine. If this icon is displayed, the CAN component was successfully probed for status and version information.
	This indicates a CAN bus component within the CTP engine. If this icon is displayed the CAN component was probed for status and version information but an error was encountered.
	This indicates a component within the CTP engine, which may be a software device or some other device which is not on the CAN bus. If this icon is displayed the component was successfully probed for status and version information.
	This indicates a component within the CTP engine, which may be a software device or some other device which is not on the CAN bus. If this icon is displayed the component was probed for status and version information but an error was encountered.
	This indicates the serial COMPORT (usually COM1) connection on the CTP engine. It may have other devices connected to it, which will also be indicated with this symbol.
	This indicates that a serial device was found but an error was encountered.
	This icon represents an application, driver or a dll file. (These are files with the .exe, .sys or .dll extension.)

Note: Depending on the Python system version some components, such as the Plate Width System (PWS), are not present and, therefore, are not displayed in the results panel.

5. Scroll down and check through the list of sub-systems. If any icon is crossed through in red (for example ) it is not responding and needs to be investigated. In the first instance, check the cable connections at the bulkhead.

Note: If you cannot identify the cause of the problem, please call a support engineer.

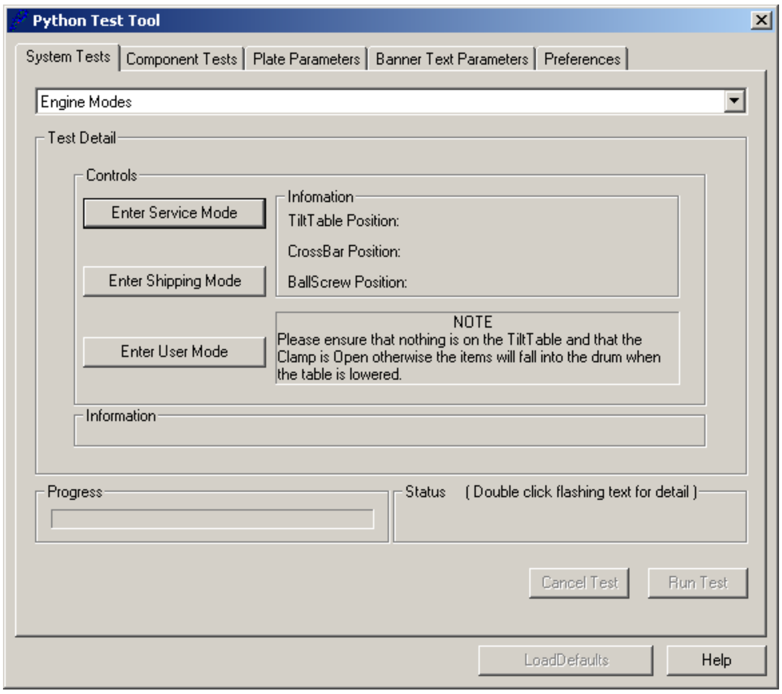
6.2 Cleaning the Python tilt-table and drum

Before loading a plate into Python, the tilt-table and drum must be cleaned because loose debris may have settled on their surfaces during transit or unpacking.

6.2.1 Cleaning the tilt-table

In order to clean Python's tilt-table, you must use the Python Test Tool to position the tilt-table in its horizontal position:

1. Make sure Python's lid is closed.
2. In the Python Test Tool, select the **System Tests** tab then select **Engine Modes** from the pull-down menu to display the following dialog:



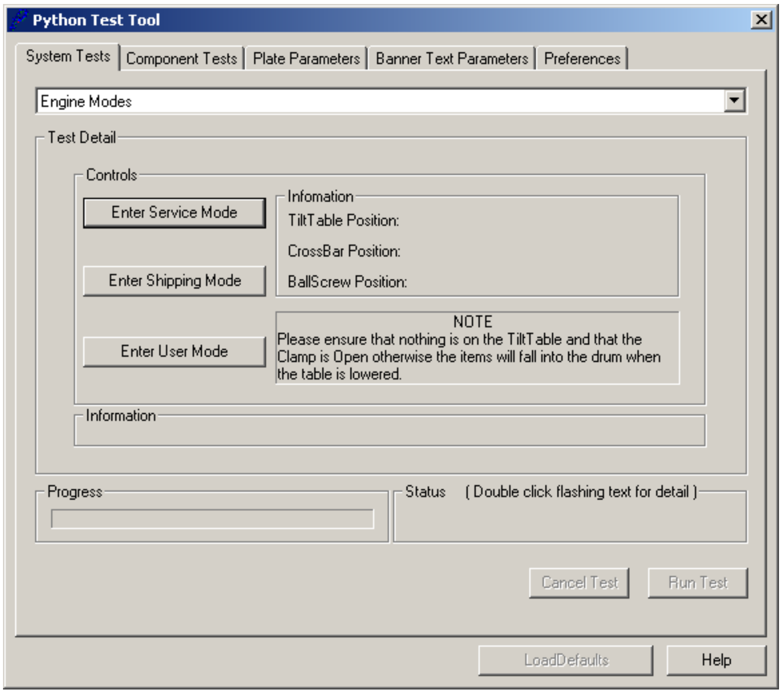
3. Click on the **Enter User Mode** button.
4. The carriage will move to the right-hand side of the drum, the crossbar will move up the tilt-table and the tilt-table will be raised to its horizontal position.
5. You should now clean the tilt-table and surrounding area carefully and thoroughly using a vacuum cleaner hose.

6.2.2 Cleaning the Python drum

To clean the Python drum, you use the Python Test Tool to put the Python engine into 'service mode'. This parks the optics carriage in a safe position and moves the tilt-table to its vertical position so that as much of the drum as possible is exposed. You can then clean the drum with a vacuum cleaner hose. To do this:

1. Make sure Python's lid is closed.
2. In the Python Test Tool, click on the **System Tests** tab.

3. From the drop-down menu select **Engine Modes** to display the following dialog:

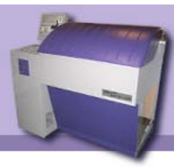


4. Click on the **Enter Service Mode** button. The system will 'park' the tilt-table in a safe, vertical position, and move the carriage to the far left-hand side of the drum.

5. Ensure that the internal drum and the area at the rear of the tilt-table are clear of debris. Use a vacuum cleaner hose, if necessary.

Warning: When you have access to the inside of the Python engine, take great care not to knock the carriage or other sensitive parts of the Python engine as this could result in damage.

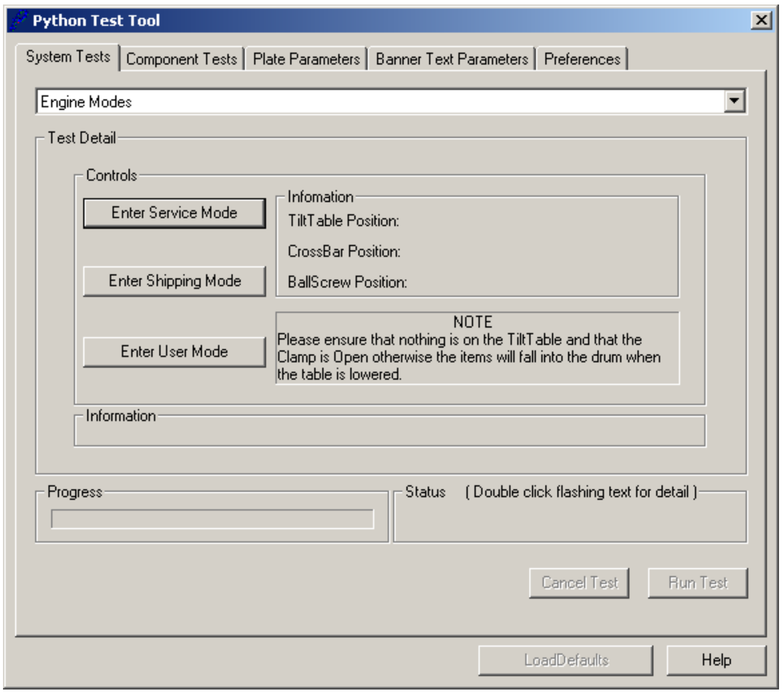
The instructions in the next section show you how to move the tilt-table back to its horizontal position.



6.2.3 Raising the tilt-table to its horizontal position

After you have cleaned the Python engine, the tilt-table must be raised back to its horizontal position so that normal operation can be resumed. To do this:

1. Make sure Python's lid is closed.
2. Select the **System Tests** tab. From the drop-down menu, select **Engine Modes** to display the following dialog:



3. Click on the **Enter User Mode** button. This restores the Python engine to its default status (the tilt-table is raised to its horizontal position and the crossbar moves up the tilt-table) ready to start imaging plates.

The next chapter shows you how to load a plate onto Python's tilt-table.

6.3 Enabling image clipping

From v5.0.0 engine software onwards, Python provides an 'image clipping' mode, which allows users to image plate-sized jobs.

Usually, when a plate is imaged, the user specifies that the image will start at a specific offset from the front of the plate in order that the front of the image is not 'lost' over the clamped area of the plate (as shown in diagram 1 below). However, if the user wishes to expose plate-sized jobs, they can achieve this using 'image clipping'. This positions the start of the image with no offset from the front of the plate (that is, at 0mm) and, hence, the start of the image gets clipped off (as shown in diagram 2 below) because it falls within the unimageable area of the plate.

Diagram 1. Image position without clipping
The image starts at an offset

Diagram 2. Image position with clipping
The image starts at 0mm offset

Notes: The distance that gets clipped off is defined by the **Plate in clamp** distance in the Python Test Tool's Engine Calibrations.

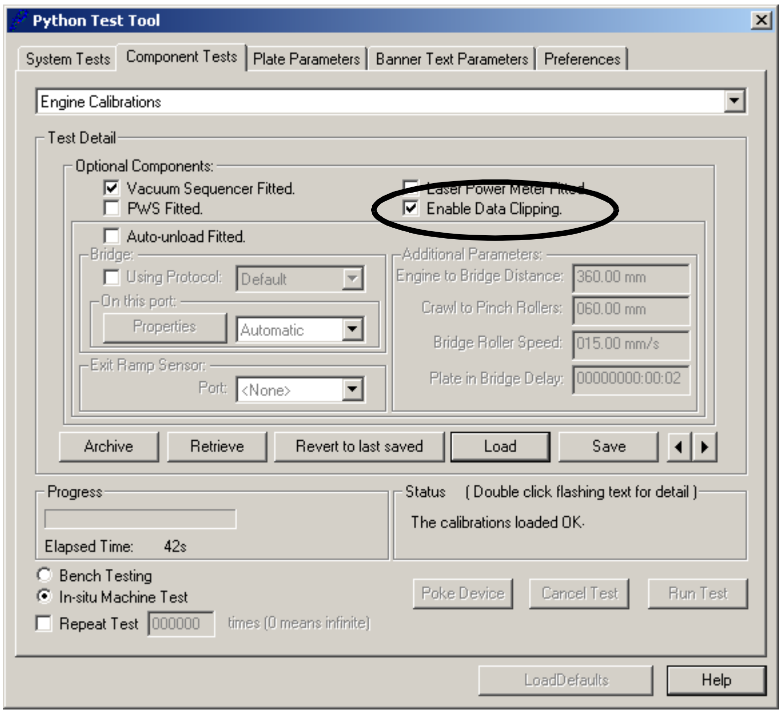
Image clipping is carried out by PythonImager and not the Python Test Tool. The Python Test Tool's Image Output test does not implement the image clipping feature and will, therefore, always output images as if the image clipping feature is turned off.

When image clipping is enabled, the minimum front gap setting in the Python Layout Tool is automatically set to **0**mm. We recommended that all layout files used on systems with image clipping enabled, specify no front gap (that is, 0mm) and it is left to the page makeup/imposition software to generate plate sized jobs and position the content within the job appropriately (that is, the front gap in the Python Layout Tool should no longer be used to position the job on the plate).

To enable image clipping:

1. Launch the Python Test Tool, if it is not already running.
2. Select the **Component Tests** tab, then select the **Engine Calibrations** test from the pull-down menu.
3. Click on the **Load** button to load the current engine calibration data from the machine.
4. Click on the **Next** arrow ▸ twice to reach the Optional Components page of the Engine Calibrations data.

5. Enable image clipping by checking the **Enable Data Clipping** box:



6. Save the engine calibrations by clicking on the **Save** button.

7. Next, you should archive the engine calibrations. For instructions on how to do this, please refer to section 16.2 of the **Python User Guide**.

8. Close the Python Test Tool when you have finished using it.

6.4 Setting up email

You should now set up the email package that is pre-installed on the machine.

7. Loading a plate onto the tilt-table

After cleaning the Python tilt-table and drum you are ready to start the plate tests. Before you do these, however, you should load a plate onto the Python tilt-table to make sure that the clamping mechanism is working properly.

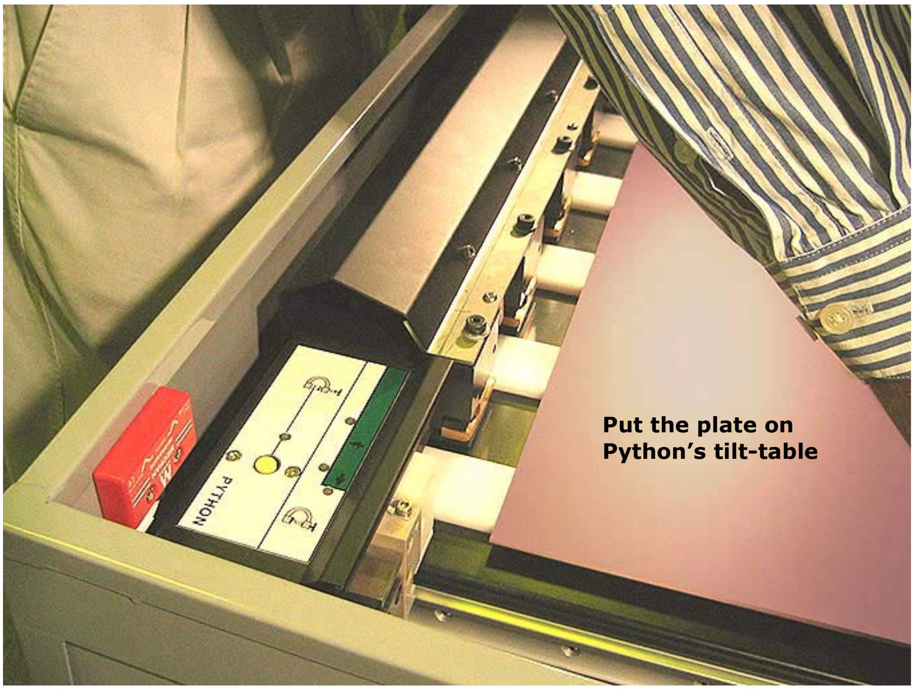
This chapter covers the following:

Warning: Please make sure that you are familiar with plate handling procedures, as described in section 2.2 on page 12.

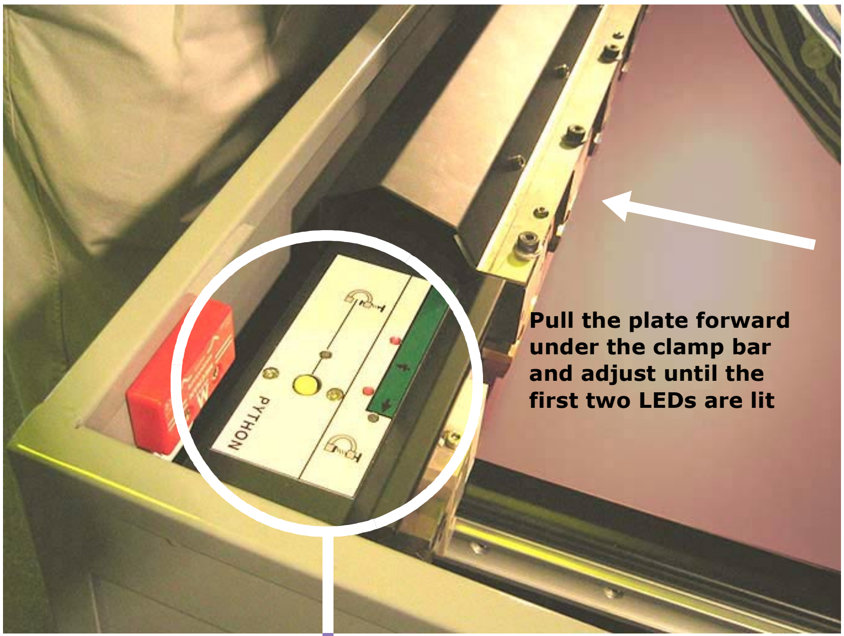
7.1 Loading a plate onto the Python tilt-table

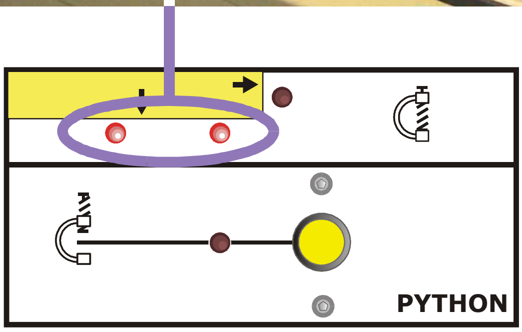
To load a plate onto the Python tilt-table:

1. Put the plate on the Python tilt-table (carefully hold the plate at its edges with your fingertips):

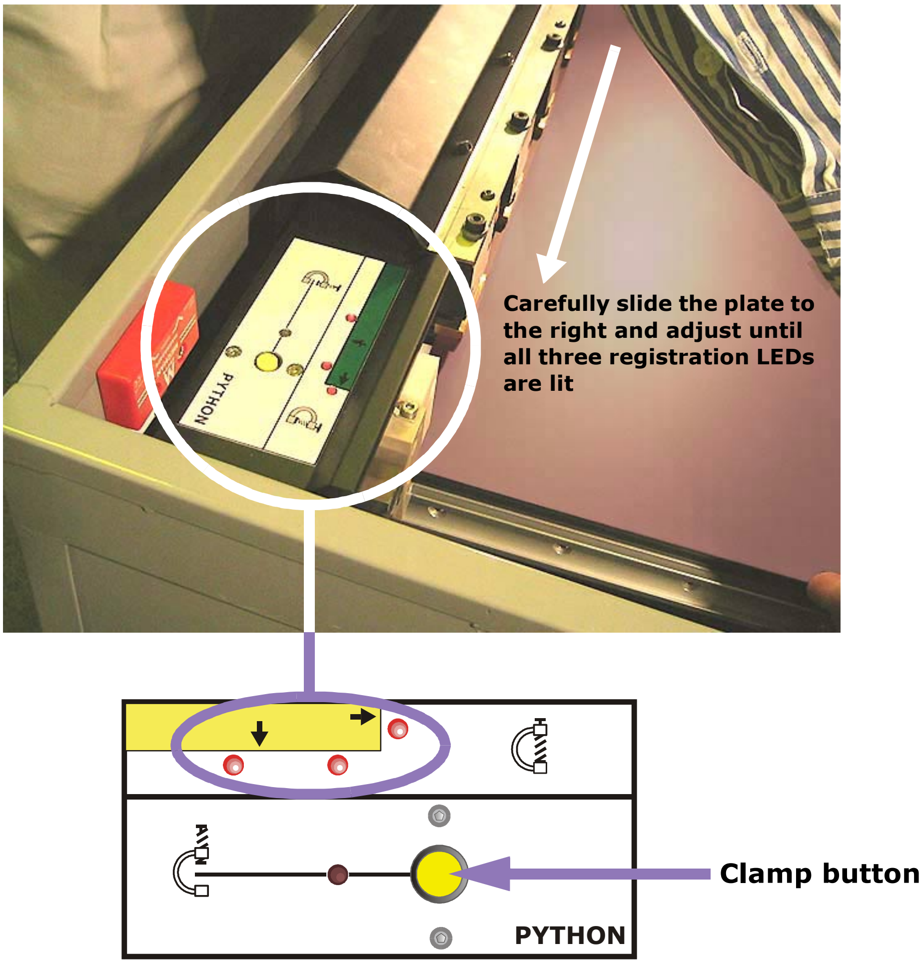


2. Slide the plate forwards and under the clamp until the first two registration LEDs on the crossbar light up, as shown below:





3. Carefully, slide the plate to the right until all three registration LEDs are lit. Make sure that both the front registration LEDs stay on:



4. When all three registration LEDs are lit (that is, when front and side registration is maintained) hold the plate securely until the clamp automatically closes to hold the plate in position.

5. Once the lid is closed, Python is ready to image a plate.

   Note: If, during clamping, one or more of the LEDs go off, press the clamp button to release the plate, move the plate away from the contacts and wait for the clamp to re-arm. Repeat the plate loading process (steps 2–4).

7.1.1 To remove the plate

To remove the plate, open Python’s lid and press the yellow clamp button to unclamp the plate. Carefully pull the plate out from under the clamp.

7.1.2 When you have finished

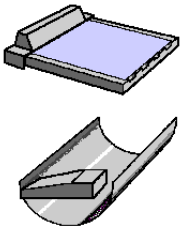
By now, you should have successfully loaded the plate. If you are unfamiliar with the plate loading process then you should practise a few times until you feel more confident.

The next section describes what happens when Python images a plate. The following three chapters describe how to create a plate definition using the Python Layout Tool, how to set up Python’s plate auto-unload feature and how to carry out the plate tests.

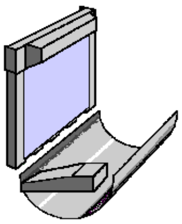
7.2 The Python plate load sequence

When the plate is loaded and a software command is given (from the Python Test Tool or the Python Console) to the Python engine to start imaging, the following plate load/image/unload sequence is carried out:

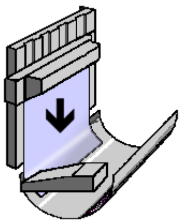
1. The plate is loaded onto the tilt-table and clamped:



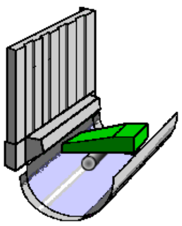
2. The optics carriage moves to its right-hand limit switch and the spinner runs up to speed.
3. The tilt-table drops down to a vertical position:



4. The crossbar moves down the tilt-table and inserts the plate into the drum:

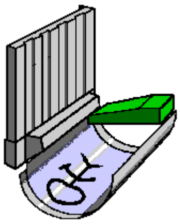


5. Sensors detect the width of the plate and the carriage moves to the centre of the plate.
6. The vacuum and pressure roller are applied to hold the plate in position in the drum:

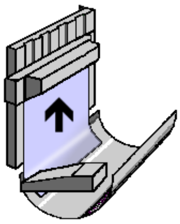


7. The carriage moves back to the right-hand side of the drum.
8. The laser shutter opens.

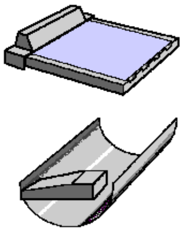
9. The carriage traverses across the plate at 6mm per second, generating the image as it travels.



10. On job completion, the vacuum is turned off and the crossbar raised up, removing the plate from the drum.



11. The tilt-table is tilted back to its horizontal position and the plate can either be removed by the user or it is automatically unloaded through the rear cover (if the plate auto-unload feature is available and enabled).



8. Creating the plate definitions

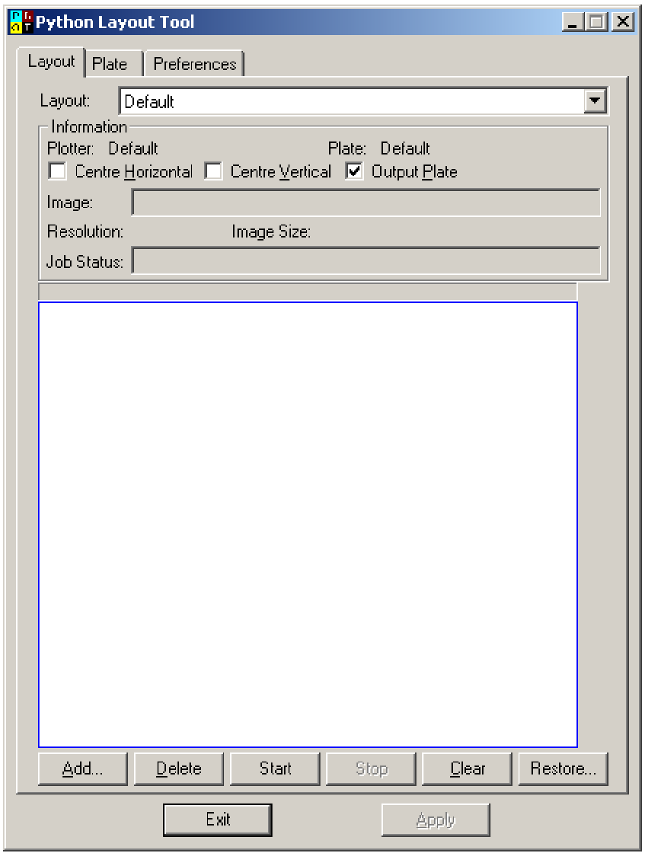
Before you can carry out the plate tests described in chapter 10 you need to create a plate definition for every plate that the customer will be using and that you will be testing. The plate definition contains information about the plate size, thickness, etc. This chapter shows you how to create the plate definitions using the Python Layout Tool.

This chapter contains the following sections:

8.1 Launching the Python Layout Tool

On the Python workstation, launch the Python Layout Tool by selecting **Start > Programs > HighWater Designs > Python v5.x.x > Python Layout Tool**.

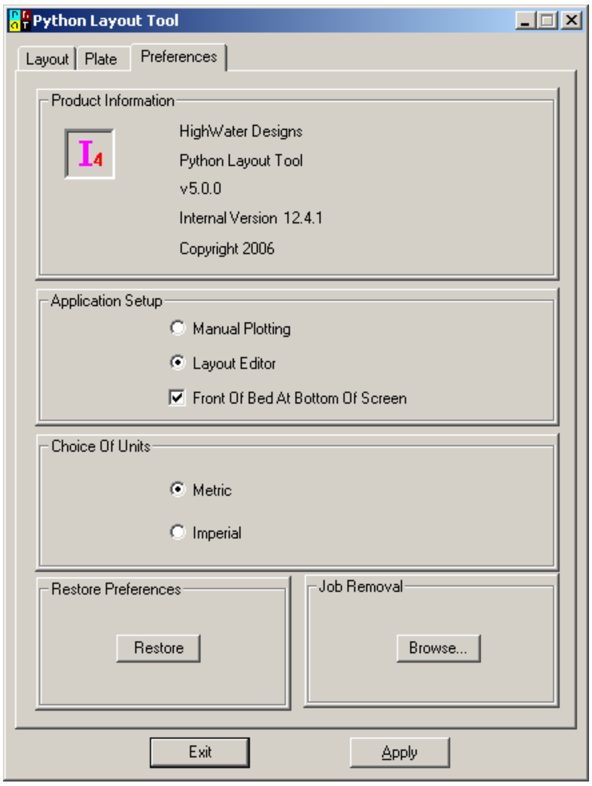
The Python Layout Tool software launches:



8.2 Setting the preferences

First, you need to set the preferences. To do this:

1. Click on the **Preferences** tab to view the following pane:



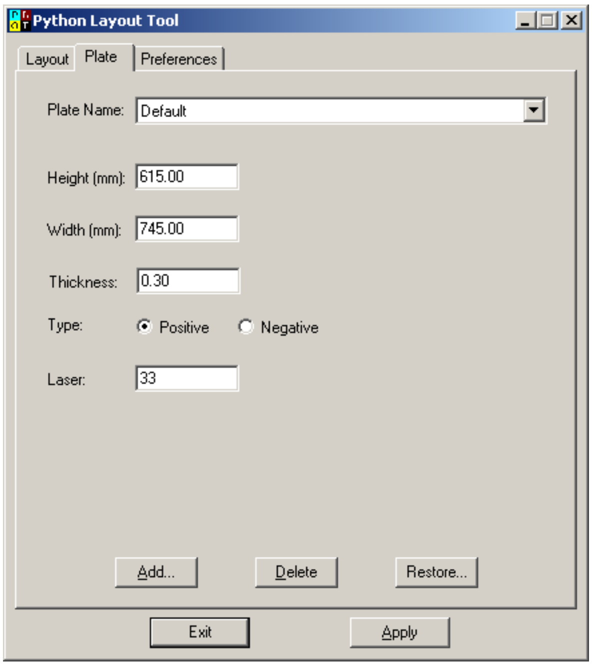
2. In the **Choice Of Units** panel, select either **Metric** to display measurements in millimetres, or **Imperial** to display measurements in inches.

3. Click on the **Apply** button to apply any changes.

8.3 Creating a plate definition

To create a new plate definition:

1. Click on the **Plate** tab to display the following dialog:

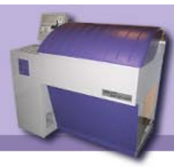


2. Make a note of the current **Plate Name** in the menu at the top (you will need to re-select this when you have finished creating plates).
3. Click on the **Add...** button, type a new name for the plate then click on **OK**.
   Note: Make sure that the plate name is meaningful, for example, **Komori20-459x530**.
4. Type in the **Height** and **Width** of the plate.
   Note: The height is the dimension from front to back of the Python tilt-table. The width is the dimension from left to right of the Python tilt-table.
5. Type in the **Thickness** of the plate (usually **0.3** or **0.15** mm).
6. Select **Positive** or **Negative** for the plate **Type**.
   Note: Refer to the plate information in section A.1 on page 101 if you are not sure if the plate is positive or negative.
7. Leave the **Laser** power setting at the default value.
   Note: The correct laser power will be determined later when you carry out the plate tests (described in chapter 10). When you have determined the correct laser power setting for each plate you will return here to the Python Layout Tool and enter this information for each plate.
8. When you have finished, click on the **Apply** button to apply the changes.
9. Now, create a plate definition for **all** other plates that the customer will be using and that you will be testing by following steps 3–7 again.
10. When you have finished creating all the required plate definitions, select the **Plate Name** that was originally displayed when you started the session (see step 2 above) and

click on the **Apply** button. This ensures that the settings of the current layout (those selected on the Layout pane) are not altered.

11. When you have finished, close the Python Layout Tool.

Next, you will set up the plate auto-unload feature, if required, as described in chapter 9. Otherwise, go to chapter 10, which details the plate tests that need to be carried out.



9. Automatic plate unload

The Python front-end software (from version 4.0.0 onwards) supports automatic unload of the imaged plate through the rear of the machine, as described in this chapter.

This chapter contains the following sections:

9.1 Before you begin

Before configuring the Python system for automatic plate unload, ensure that the Python hardware is capable of supporting plate auto-unload. (If you are not sure about this, please check with an authorized Python dealer.)

Warnings: Enabling automatic plate unload on a Python that does not have the necessary unload hardware components may damage the Python machine.

To prevent potential damage to the system, HighWater Designs strongly recommends the use of a bridge/processor with a supported communication link. (Contact a Python dealer for the latest list of supported bridges/processors and communication links.)

To use the plate auto-unload function the Python system must be physically linked to a supported plate unload bridge or processor (as described in section 3.6). If the bridge/processor can communicate with the Python system, this communication link should be used for the reasons set out below:

- To eliminate the possibility of incorrectly timed plate unload and the potentially damaging consequences of this.
- It can pass information to the unload device about the next plate due to be unloaded (for example, plate thickness, which some unload devices need to know in order to process the plate correctly).
- It allows for an intelligent plate unload whereby the Python system can check the status of the bridge/processor device before unloading, and it can delay plate unload until the device indicates it is ready to accept the plate.

The Python system can also, optionally, use RS232-style serial communications via a serial port on the Python controller PC (installed inside the Python chassis) to connect to and communicate with a bridge/processor.

9.2 Overview of the plate auto-unload operation

Automatic plate unload is achieved by the Python system automatically unclamping the plate and driving the crossbar (thereby pushing the plate) towards the rear of the machine. An exit slot (including a plate support ramp) allows the plate to be pushed out of the rear of the machine. The distance the crossbar travels (and, hence, the distance the plate is pushed out) and the speed at which the crossbar travels are parameters that can be configured in the Python system.

The following table shows the operational differences between manual and automatic plate unload operation.

Python expose operation with manual (user) plate unload	Python expose operation with automatic plate unload
The user loads a plate into the clamp. ↓ The table is tilted to its vertical position and the crossbar pushes the plate into the drum. ↓ The vacuum is enabled. ↓ The image is exposed onto the plate. ↓ The vacuum is disabled then the crossbar pulls the (still clamped) plate out of the drum. ↓ The table is tilted back to its horizontal position. ↓	The user loads a plate into the clamp. ↓ The table is tilted to its vertical position and the crossbar pushes the plate into the drum. ↓ The vacuum is enabled. ↓ The image is exposed onto the plate. ↓ The vacuum is disabled then the crossbar pulls the (still clamped) plate out of the drum. ↓ The table is tilted back to its horizontal position. ↓
The user unclamps and manually removes the plate. ↓ The user loads a new plate into the clamp.	The plate is automatically unclamped. ↓ The crossbar pushes the plate across the horizontal table towards the rear plate exit slot in the Python, at full speed, for the configured distance. ↓ The crossbar (optionally) decelerates to the configured speed and travels a further configured distance towards the exit slot. ↓ After a configured delay, the crossbar retracts back to the front of the tilt-table in preparation for the loading of a new plate. ↓ The user loads a new plate into the clamp.

The automatic plate unload operation is controlled by a number of parameters, which must be configured by the installation engineer when installing the system or when re-configuring the system for automatic plate unload operation.

The available parameters are:

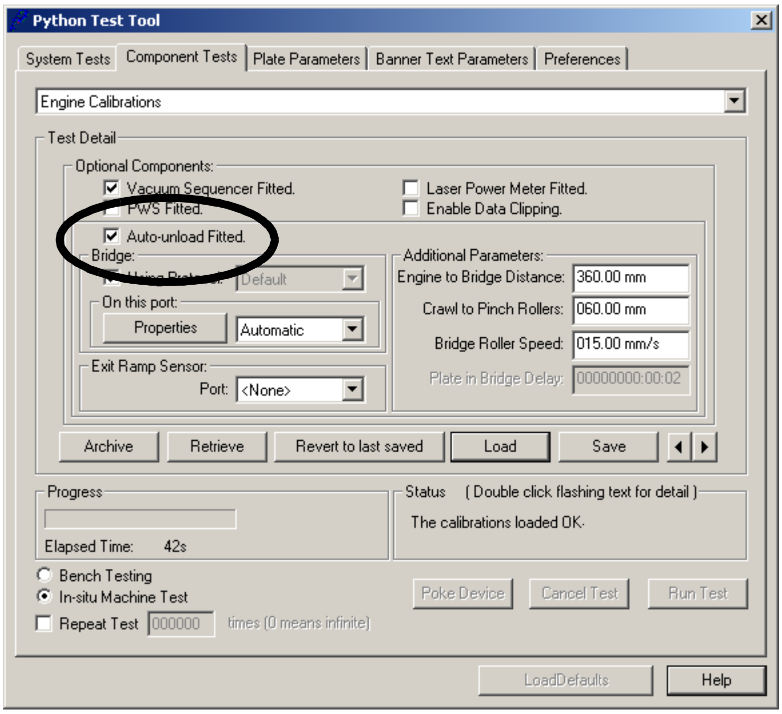
- **Auto-unload enable/disable**
- **Bridge/processor communication enable/disable**
  Contact a dealer for the latest list of supported bridge/processor communication links.
- **Bridge/processor communication protocol type**
- **Bridge/processor communication protocol serial port**
- **Bridge/processor communication protocol serial port settings**
- **Exit ramp sensor port setting**
- **Distance to push plate (first stage) at full speed**
  In the Python Test Tool, this is referred to as 'Engine to Bridge Distance'.
- **Distance to push plate (second stage) at user-defined speed**
  In the Python Test Tool, this is referred to as 'Crawl to Pinch Rollers'.
- **User-defined speed for second stage**
  In the Python Test Tool, this is referred to as 'Bridge Roller Speed'.

These parameters are configured in the Python Test Tool's **Engine Calibrations** dialog (as described in sections 9.3–9.5).

9.3 Selecting plate auto-unload

To configure the Python system to automatically unload a plate onto a bridge/processor after exposure:

1. In the Python Test Tool, select the **Component Tests** tab, then select the **Engine Calibrations** test from the pull-down menu.
2. Click on the **Load** button to load the current engine calibration data from the machine.
3. Click on the **Next** arrow twice to reach the Optional Components page of the Engine Calibrations data.
4. Enable plate auto-unload operation by checking the **Auto-unload Fitted** box:



5. Now, you need to configure the software depending on whether you have a communication link:
   - If you have a communication link then go to section 9.4 on page 53.
   - If you **do not** have a communication link then go to section 9.5 on page 56.

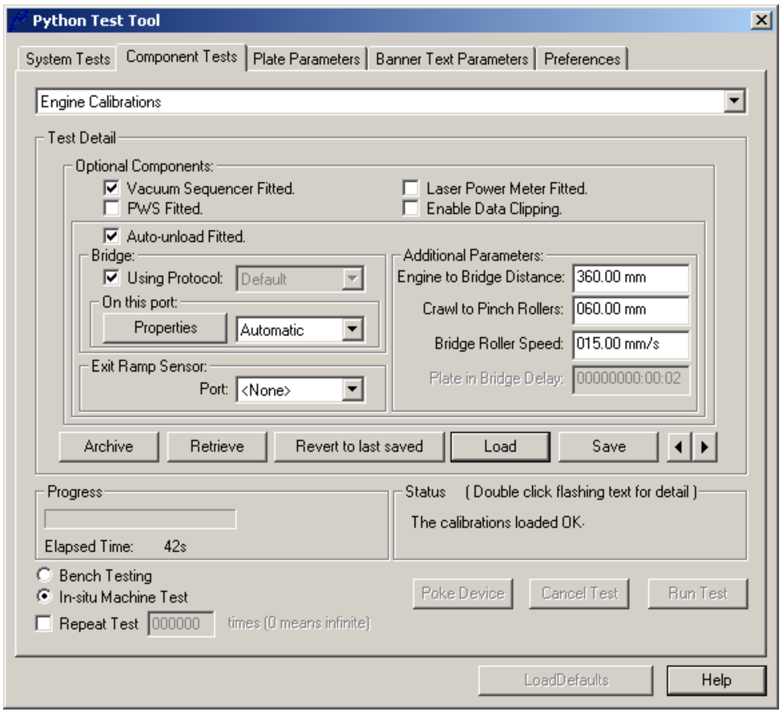
Warning: In order to avoid potential damage to the Python system, HighWater Designs recommends that you use a bridge or processor with a communication link supported by the Python system, if possible. Please see the warning in section 9.5 for more information about this.

9.4 Configuring a bridge or processor with a communication link

Note: If there is no communication link (or no supported communication link) to the bridge/processor then go straight to section 9.5 on page 56.

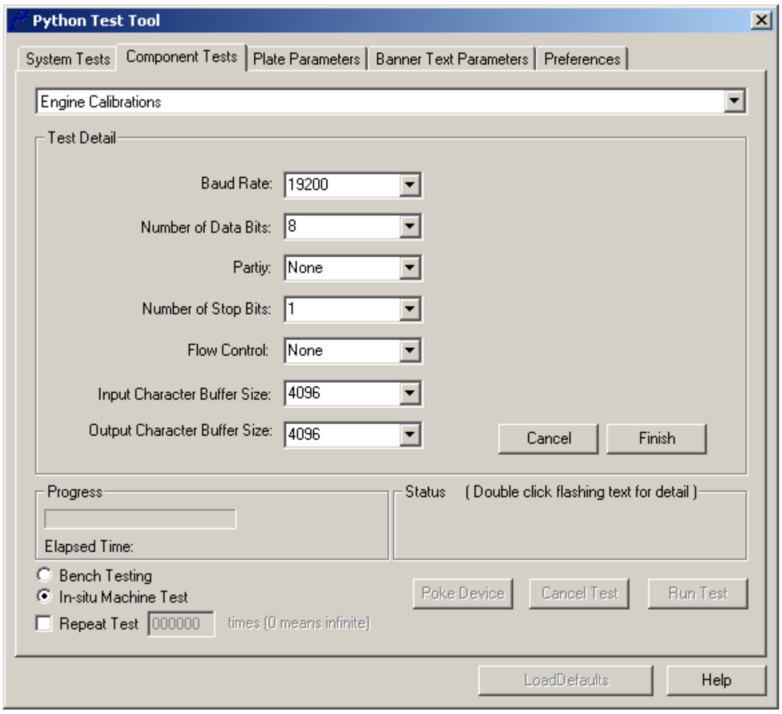
If you are using a bridge or processor with a communication link supported by the Python system, then set it up as follows:

1. Go to the Optional Components page of the Engine Calibrations dialog (see section 9.3 for details of how to get to this page):



2. Enable the communication link to Python by checking the **Using Protocol** box.
3. Select the appropriate bridge protocol from the drop-down list (currently, only the **Default** option is available).
4. Select **Automatic** from the **On this Port** drop-down list.
   Note: The **Automatic** setting instructs the Python software to check all available serial ports in turn (starting at the lowest numbered port) for the presence of the bridge/processor, and to automatically select and use the port on which it first finds the bridge/processor.
5. In the Exit Ramp Sensor panel, select **<None>** for the **Port** setting.

6. To configure the communication parameters for the port, click on the **Properties** button. The following dialog is displayed:



7. Set the port properties, as appropriate.

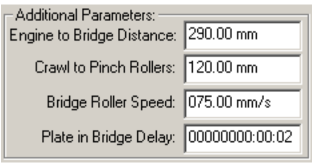
   Note: Refer to the bridge/processor documentation for details on how these port settings should be configured.

8. When you have finished, click on the **Finish** button.

The following sub-section shows you how to set the Additional Parameters.

9.4.1 Additional parameters (bridge protocol enabled)

On the Engine Calibrations Optional Components page (shown on the previous dialog), set the Additional Parameters panel, as described next:



1. In the **Engine to Bridge Distance** field, specify the distance from the rear edge of the Python tilt table (when horizontal) to a point just beyond (5–10mm) the plate input sensor of the attached bridge/processor.

2. In the **Crawl to Pinch Rollers** field, specify the distance from the plate input sensors on the bridge/processor to a point at least 10mm beyond the bridge/processor input pinch rollers.

3. In the **Bridge Roller Speed** field, specify a speed that is slightly slower than the bridge/processor input pinch roller speed.

Warning: It is vital that this speed is slower, otherwise Python will push the plate too quickly into the bridge/processor pinch rollers and the plate, along with the Python, bridge and/or processor, may be damaged.

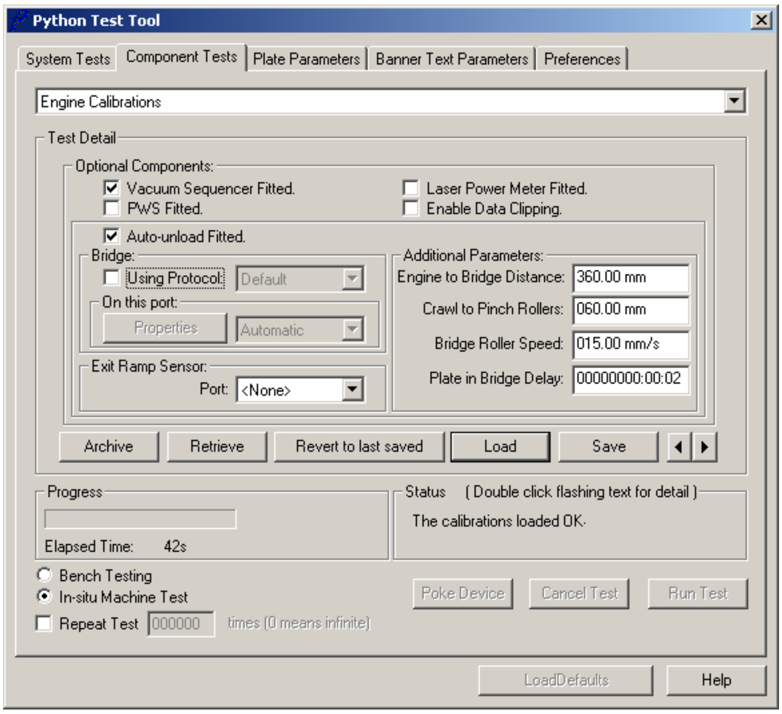
4. The **Plate in Bridge Delay** field is not required since Python can check when the plate has been removed from the system by the unload device via the communication link.
5. When you have finished configuring the automatic plate unload parameters, click on the **Save** button to save the engine calibration data back to the engine, thereby applying the changes.
6. Now, go straight to section 9.6 which shows you how to test the plate auto-unload operation.

9.5 Configuring a bridge or processor with no communication link

Warning: Using the Python system with no communication link to the bridge/processor, as described in this section, is potentially dangerous because Python will automatically unload the imaged plate from the tilt-table even if the bridge/processor is not ready to receive the plate. This, potentially, could damage the input pinch rollers and Python's registration pins. In addition, the plate may fall into Python's drum where damage may occur. Therefore, we do NOT recommend using automatic plate unload without a communication link. If you do, however, you must ALWAYS ensure that the processor is ready to receive another plate before you load a plate onto the Python tilt-table. Also, you MUST instruct the user to do this.

If you are using a bridge or processor that does not have a communication link or the link is not currently supported by the Python software, then configure the software as follows:

1. Go to the Optional Components page of the Engine Calibrations dialog (see section 9.3 for details of how to get to this page):



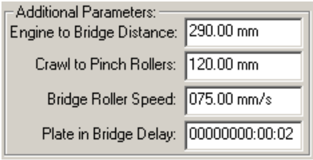
2. Disable the communication link to Python by unchecking the **Using Protocol** box.

3. In the Exit Ramp Sensor panel, select **<None>** for the **Port** setting.

4. Once the plate has been imaged, the unload operation will eject the plate from the rear of the Python using only the parameters specified in the Additional Parameters panel, as described next.

9.5.1 Additional parameters (bridge protocol disabled)

On the Engine Calibrations Optional Components page (shown on the previous dialog), set the Additional Parameters panel, as described next:



1. In the **Engine to Bridge Distance** field, specify the distance from the rear edge of the Python tilt table (when horizontal) to a point just beyond (5–10mm) the plate input sensor of the attached bridge/processor **or** if no plate input sensor is fitted, then specify the distance from the rear edge of the Python tilt-table to a point at least 20mm before the input pinch rollers of the bridge/processor.
2. In the **Crawl to Pinch Rollers** field, specify the distance from the plate input sensors on the bridge/processor to a point at least 10mm beyond the bridge/processor input pinch rollers.
3. In the **Bridge Roller Speed** field, specify a speed that is slightly slower than the bridge/processor input pinch roller speed.

   Warning: It is vital that this speed is slower, otherwise the Python will push the plate too quickly into the bridge/processor pinch rollers and the plate, along with the Python, bridge and/or processor, may be damaged.

4. In the **Plate in Bridge Delay** field, specify the amount of time required for the bridge/processor to pull the plate fully clear of the Python system. The Python software will ensure that, after all crossbar plate push motions have completed, there is a delay of at least this length of time before the crossbar is retracted back to the front of the tilt-table in preparation for the next plate to be loaded.
5. When you have finished configuring the automatic plate unload parameters, click on the **Save** button to save the engine calibration data back to the engine, thereby applying the changes.

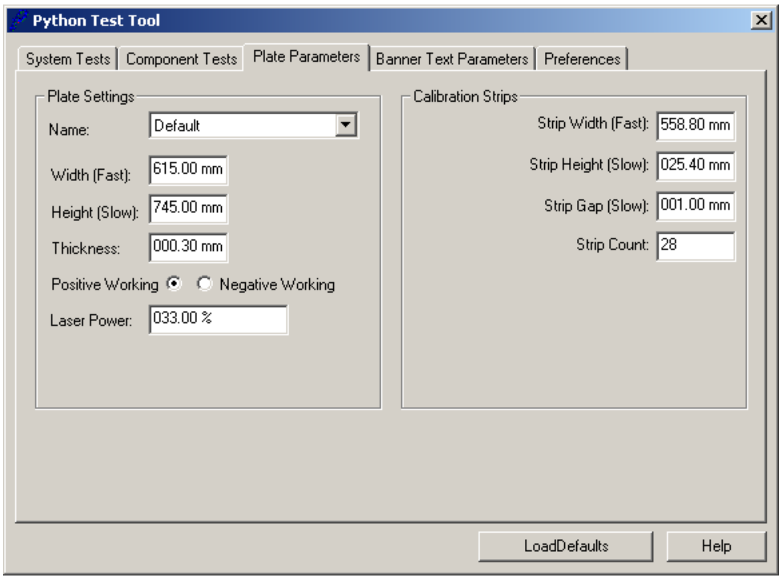
The next section shows you how to test the plate auto-unload operation.

9.6 Testing the plate auto-unload operation

Warning: Before attempting to test the plate auto-unload function, please make sure that the plate exit ramp has been set up correctly, as described in section 3.6.

Having set up the plate auto-unload operation, you now need to test that it works correctly. To do this:

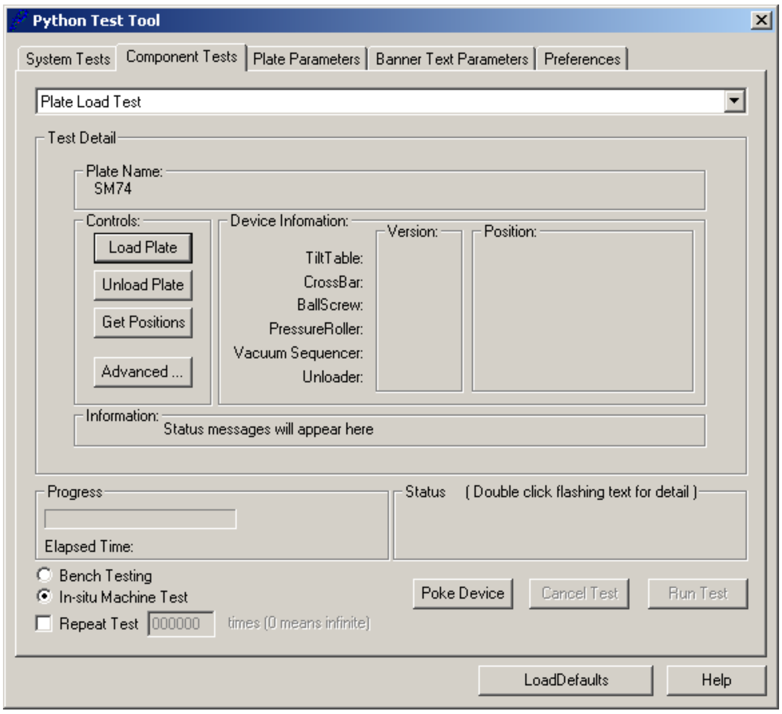
1. In the Python Test Tool, select the **Plate Parameters** tab to display the following dialog:



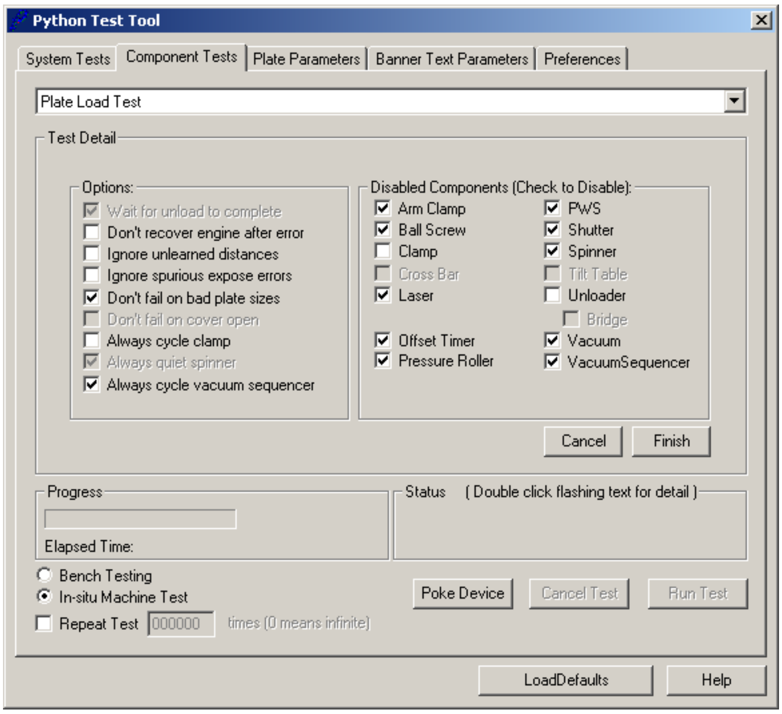
2. From the **Name** pull-down menu, select the plate that matches the dimensions of the plate to be used for the unload test.

   Note: The plate names available on this dialog are those that have been created in the Python Layout Tool. If the required plate is not available you can either create it in the Python Layout Tool (this is covered in chapter 8) or you can alter the plate values on this dialog to suit your requirements. If you do this, any new values you enter will persist even when you re-launch the Python Test Tool program. To restore the values to their initial values (that is, those set in the Python Layout Tool), re-select the plate **Name** from the pull-down menu.

3. Select the **Component Tests** tab and select **Plate Load Test** from the pull-down menu to display the following dialog:



4. Click on the **Advanced...** button and set the options as follows:



5. Click on the **Finish** button to return to the main Plate Load Test dialog.

6. Load a plate into the clamp, as described in section 7.1 on page 39.

   Note: Since the system is not imaging a plate, the automatic clamp arming software will not be active. Therefore, you will need to manually arm the clamp prior to loading the plate.

7. Click on the **Unload Plate** button to execute the plate unload test.

   Warning: If, at any time, the plate hits an obstruction, open the Python cover to stop the unload operation immediately.

8. If the unload parameters require adjustment, go back to section 9.4 or section 9.5, as appropriate, and make the necessary engine calibration data changes, then perform the unload test again.

   Note: You should carry out the auto-unload test on all plate sizes the user intends to use.

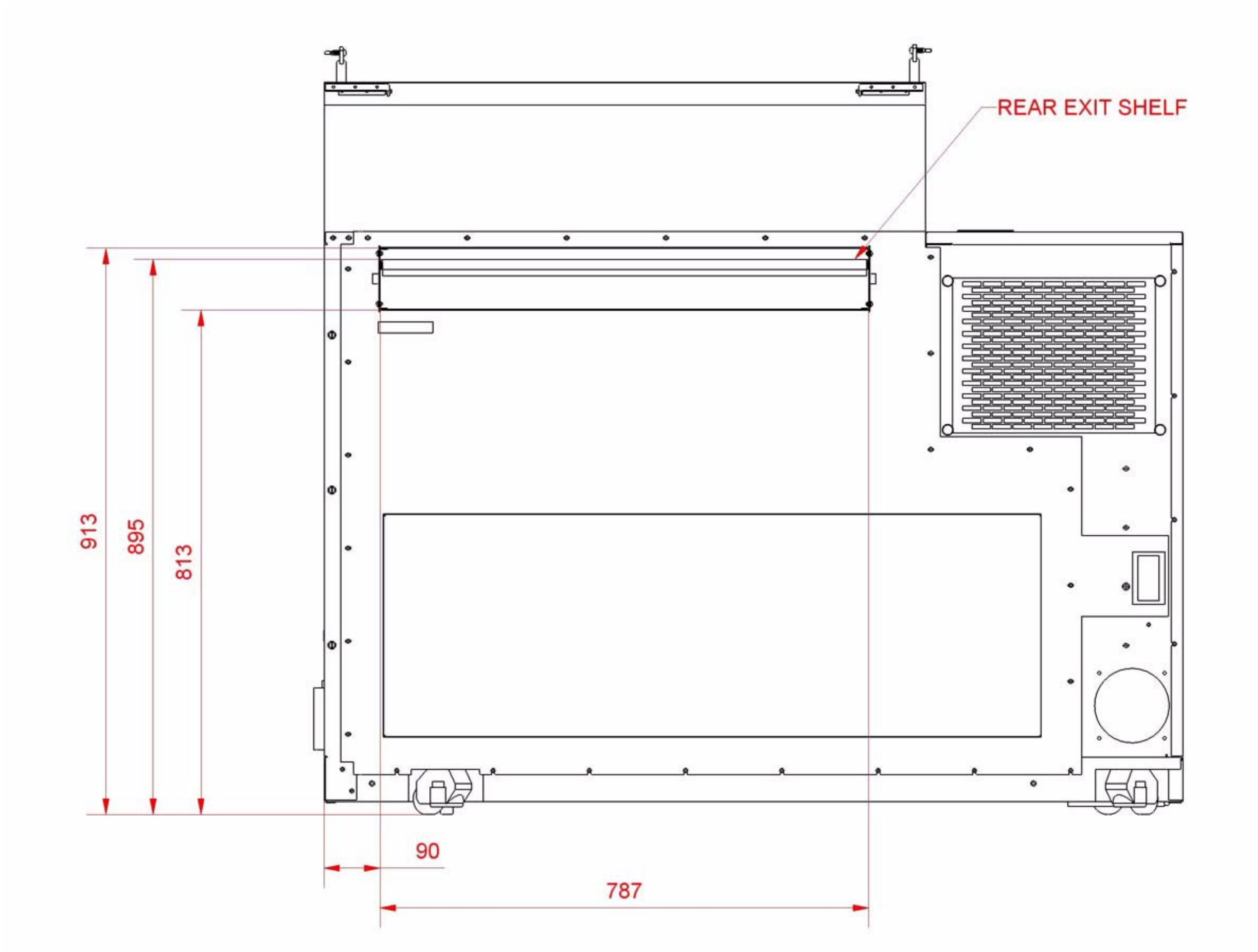
9.6.1 Testing plate auto-unload in the Python Console

Having completed the plate auto-unload tests using the Python Test Tool, you should later use the Python Console to test full image output with plate auto-unload for every plate size that the customer is using or plans to use.

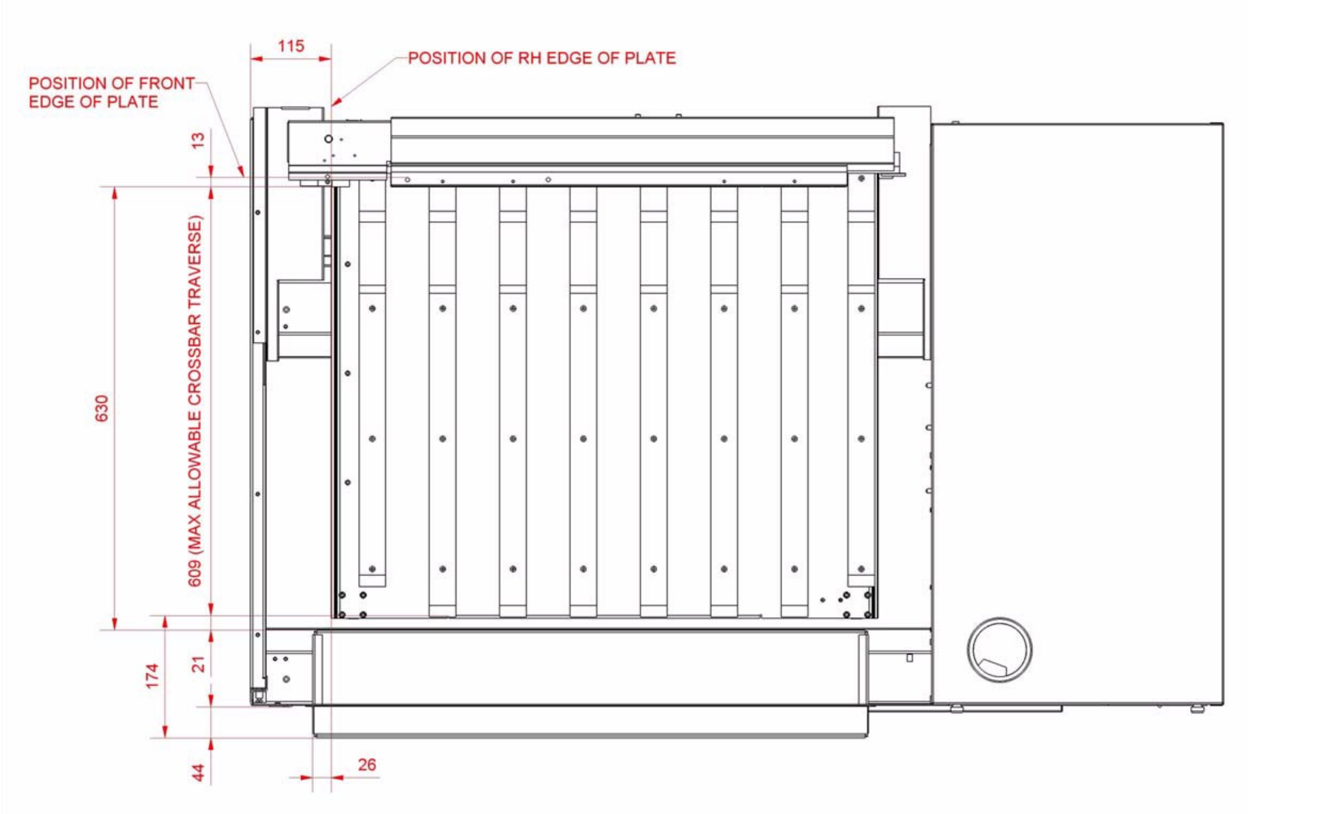
9.7 Physical dimensions and limits

The following three diagrams provide detailed dimensions relating to Python's automatic plate unload.

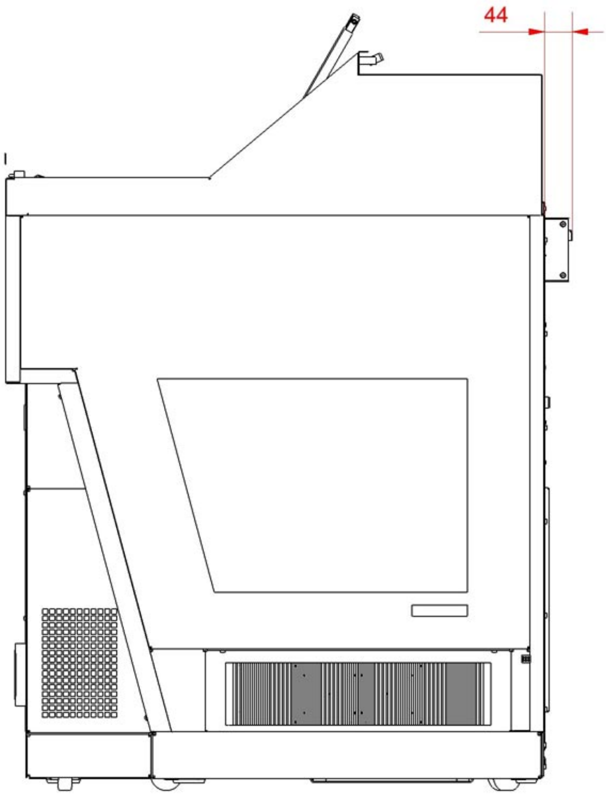
Rear view



Top-down view



Side view

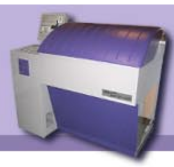


Based on the above dimensions and the implied machine and operational tolerances, the maximum distance that an unloaded plate can protrude from the rear of the Python system (measured from the rear edge of the plate exit slot) is:

Plate dimension (around the drum) - 187mm (+/- 5mm)

So, for example, assuming maximum crossbar travel, the amount of plate protruding from the rear of the Python for the minimum and maximum plate sizes would be as follows:

Plate size	Amount protruding
400 x 335 x 0.15mm	208 to 218mm
615 x 745 x 0.3mm	423 to 433mm



10. The plate tests

After transit and installation of the Python engine you need to check that all Python's operating parts are still correctly aligned and that the engine is outputting plates satisfactorily. This chapter takes you through the various tests that need to be carried out.

This detailed chapter is intended mainly for engineers who are not familiar with the plate tests. More experienced engineers can work solely from Appendix D.

This chapter includes the following sections:

Warning: If you are not familiar with the plate tests then read this whole chapter carefully.

10.1 Requirements for the tests

Before you start the tests, please make sure that you have the following:

Requirement	Notes	✔
Plate definition(s)	You need to set up a 'plate definition' for each plate the customer will be using and that you will be testing. Creating the plate definitions is covered in chapter 8. **If you have not already created the required plate definitions, please do so now.**	
Brightroom/darkroom and processor	These are required for outputting and processing the plates required for the plate tests. Ensure that the lighting is suitable for the plates being used (see section B.1 on page 105 for more information).	
Plates	• You will need a minimum of 4 unexposed, good quality plates for the tests. Make sure the plates are compatible with the Python system (refer to section A.1 on page 101 for more details). • Unless otherwise stated, you should use the largest available plate size for the tests (maximum 745 x 615mm, 0.3mm thick). It is best to use square-edged plates, although this is not absolutely essential. **Warning: If plates do not have square edges some measurements may be affected, so you should take this into account when assessing the test results. The following sections have more information about this, where relevant. A simple check of plate squareness is to place one plate against another, emulsion side to emulsion side, and check if the edges line up.**	
1 metre precision steel ruler	This should have as fine rules as possible.	
Magnifying eye glass(es): one with a vernier scale	Ideally, you should have two eyeglasses: **x10** and **x100** magnification.	
Good lighting conditions	The plates you output will contain some very small/fine detail (text, lines, patterns, etc.). Therefore, you should examine the plates under good lighting conditions.	
A large, flat, clean surface	The plates need to remain flat and clean while you are taking measurements.	
Knowledge of vernier scales	For one of the tests, you are required to read a vernier scale.	
Plate densitometer	This is required for measuring the density patches off the plates.	

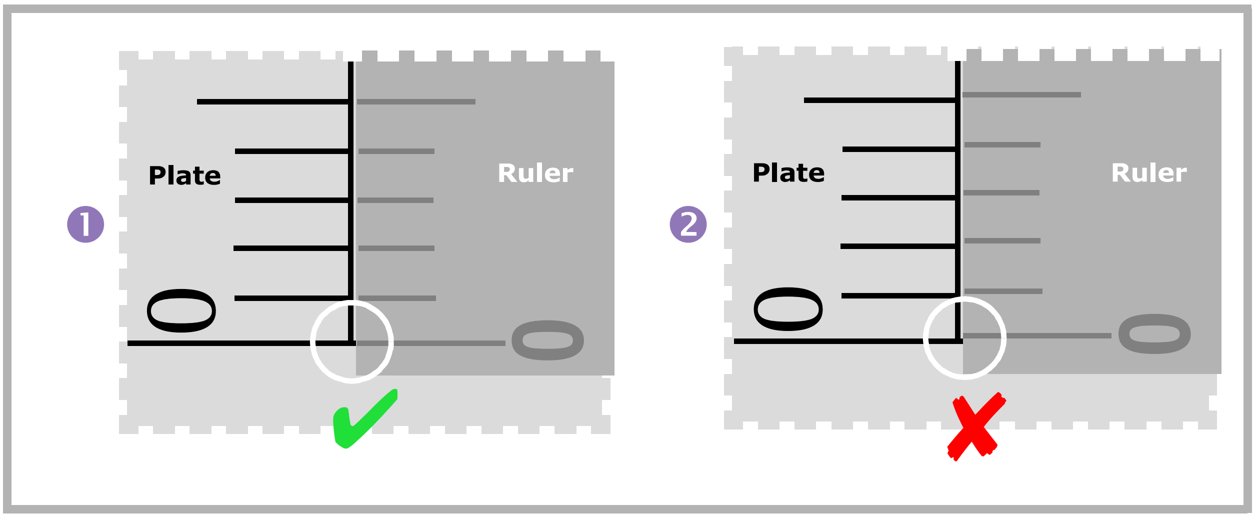
10.2 Guidelines on taking measurements

10.2.1 Measuring the rulers on the plates

Warning: It is vital that you ACCURATELY measure the rulers (or other objects) on the plates, as described below.

Make sure that you line up the 0cm (or other) mark on the steel ruler precisely with the 0cm mark on the plate, as shown in ❶ below. In ❷ below, the rulers have **not** been accurately aligned so this will introduce an error into any measurements taken.

Note: When aligning the rulers, make sure that your eye is directly above the 0 marks (or other marks that you are measuring).



When taking measurements, use a magnifying glass with a vernier scale, if possible. When you are experienced at carrying out these tests you may be able to do them accurately enough 'by eye' but, if in doubt, always use a magnifying glass and scale.

10.2.2 Making visual checks

Some of the tests require that you make visual checks of objects on the plate (for example, the laser focus checks) rather than taking measurements with a ruler. Initially, you may find it difficult to assess the quality of the objects until you have gained more experience. Where possible, this manual gives examples of good and bad output to help you judge the quality of the plate.

10.3 Recording the test results

When you have finished each test, you should record the test results in Appendix D and also follow any instructions for what to do if a test fails.

Warning: Generally, if any of the tests fail, you should call a service engineer. You are not permitted to make any adjustment or repair to the Python system yourself unless you have the express permission from HighWater Designs Limited to do so. If in any doubt, call a service engineer.

10.4 The plate tests

The following sections of this manual take you through the various tests that are required. Four of the sections show you how to output the test plates, and their subsequent sections show you what test(s) to carry out on those plates:

Warning: The order in which you carry out some of the plate tests is important. Please follow any instructions you are given about this.

The beginning of each test section contains important information about the plates and tests. Please read this information carefully and follow any given instructions.

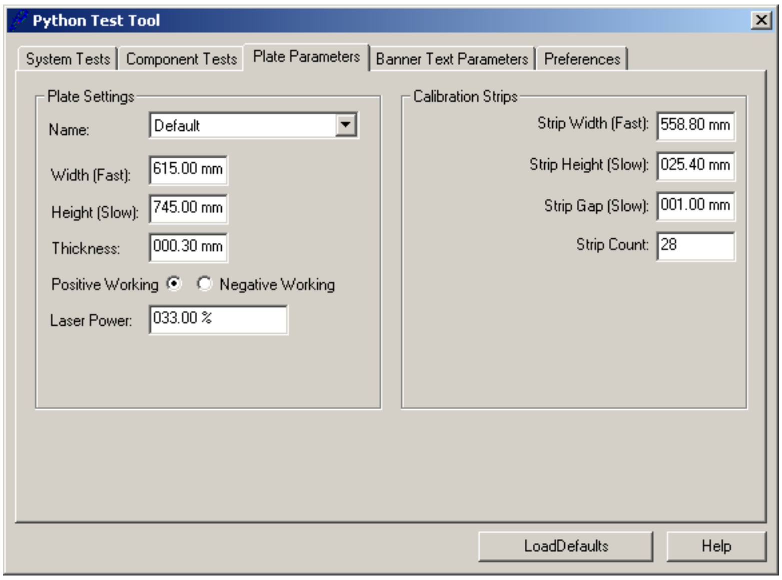
The following sections contain diagrams of plates or parts of the plates that you will be checking. These diagrams are intended for GUIDANCE ONLY and, therefore, may not match the details/scaling of the plate exactly.

Plate 1: Calibration Strips

Plate Details	
Plate name	Calibration Strips
Use plate type	Use the largest plate available (maximum 745 x 615mm, 0.3mm thick)
Job location on disk	D:\Python\Engine v5.x.x\calibration jobs\CalStrip_175lscr.TIF **or** D:\Python\Engine v5.x.x\calibration jobs\CalStrip_200lscr.TIF
Used in test	Test 1A – Laser Power on page 71
Notes	**You must output this plate and carry out the Laser Power test on it before carrying out any of the other tests.** **If you do not use the largest plate size (745 x 615mm) then, potentially, the plate may be too small to allow the power to increase sufficiently to achieve the required %dot values. In this case, output the plate again but increase the 'Start Power' so that a maximum power of 100% can be achieved.**

To output the Calibrations Strips job:

1. In the Python Test Tool, select the **Plate Parameters** tab:

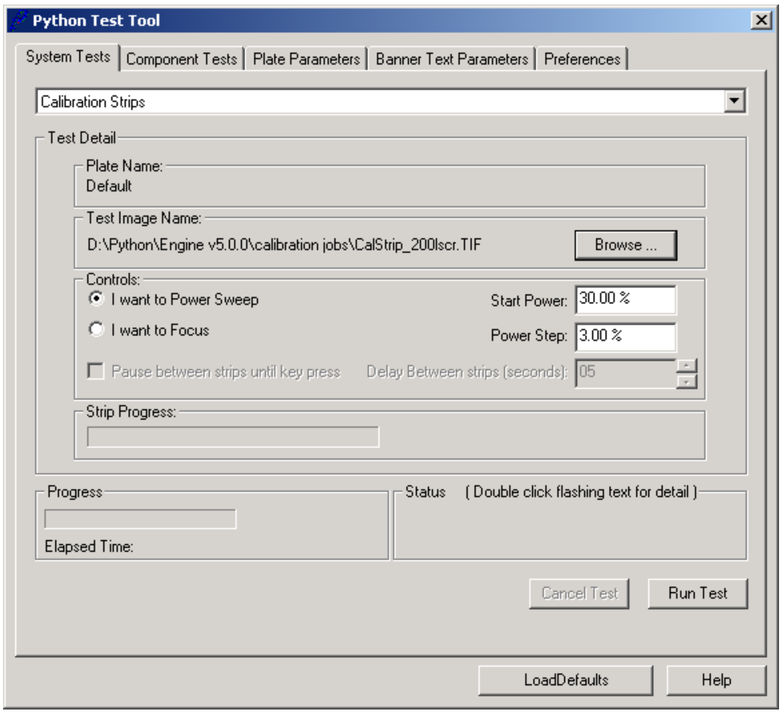


2. Select the required plate from the **Name** menu and check that the plate settings (**Width**, **Height, Thickness**, etc.) are correct.

   Notes: The plate names available on this dialog were created in the Python Layout Tool. If the required plate is not available you can either create it in the Python Layout Tool (this is covered in chapter 8) or you can alter the plate values on this dialog to suit your requirements. If you do this, the values you enter will remain even when you re-launch the Python Test Tool program. To restore the values to their initial values (that is, those set in the Python Layout Tool), re-select the plate **Name** from the pull-down menu.

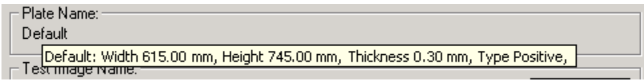
3. Do not adjust the Calibration Strips parameters.

4. Next, click on the **System Tests** tab and select **Calibration Strips** from the pull-down menu at the top of the pane. The following dialog is displayed:



5. Check that the Plate Name and details are correct.

   Note: To view the plate details, place the mouse cursor over the plate name.



6. Click on the **Browse...** button and select the **CalStrip_175lscr.TIF** or **CalStrip_200lscr.TIF** file, depending on the screen ruling required, then click on **Open**.

   Note: The default location of these files is D:\Python\Engine v5.x.x\calibration jobs.

7. In the Controls panel, select the **I want to Power Sweep** radio button.

8. Enter **30.00 %** for the **Start Power** and **3.00 %** for the **Power Step**.

   Note: If you are not using a full-size plate then set the **Start Power** to **45.00 %**.

9. Click on the **Run Test** button and, when prompted, load the plate onto the Python tilt-table then close Python's lid.

10. The Calibration Strips job is output to the plate.

    Note: As the job is output, you will see the Strip Progress bar move from 0-100 as each strip is output, and the (overall) Progress bar will move, more slowly, from 0-100 to indicate the progress of the full set of calibration strips for this test.

11. When the job has been output, remove the plate from Python and process it.

Next, you will carry out the Laser Power test on this plate, as described in the following section.

Test 1A – Laser Power

Test Details	
Test purpose	This test determines the correct laser power for the plate
Test plate/job	Plate 1: Calibration Strips
Notes	You **must** carry out the laser power test before outputting any other plates as the results of this test are required for all following tests. Ideally, you should perform a laser power test for each plate thickness used and also on plates from different manufacturers.
Acceptable tolerance	**±10% on factory setting** In any case, the laser power should not go above 80%.
If this test fails	Immediately call a service engineer for advice.

Getting the factory set laser power

Before carrying out the laser power test you need to obtain the factory set laser power value. This is printed on a label situated at the front left-hand side of the area within Python's cover.

Note: The baffle piece to the side does restrict vision slightly and the tilt-table needs to be tilted in order to view it.

The label contains details of the various power levels to be used with the plate types that are supported by Python, together with any requirement for a filter, for example:

Laser Power Settings for Plate Types		**Laser Power Measurements**			
Silver plate (2µJ/cm2 with ND filter)	**30%**	100%	**147** uW*	**17.2** mW	**223** uJ/cm2
Agfa N91v (40µJ/cm2) 3.0mW (26uW*)	**Need 0.4D**	50%	**75** uW*	**8.7** mW	**114** uJ/cm2
Fuji LP-NV (90µJ/cm2) 6.9mW (60uW*)	**39%**	25%	**40** uW*	**4.6** mW	**60** uJ/cm2
Lastra LV2 (130µJ/cm2) 10mW (85uW*)	**57%**	* LaserStar power meter dynamic measurement			
Py **381**Ld **316** **60mW**					

In this label example, the 'Need 0.4D' comment indicates that a filter needs to be fitted for the Agfa N91v plate. Please contact the supplier for more information about filters.

Note: If the power levels label on the Python machine is not present then you should carry out the following test but without the reference value.

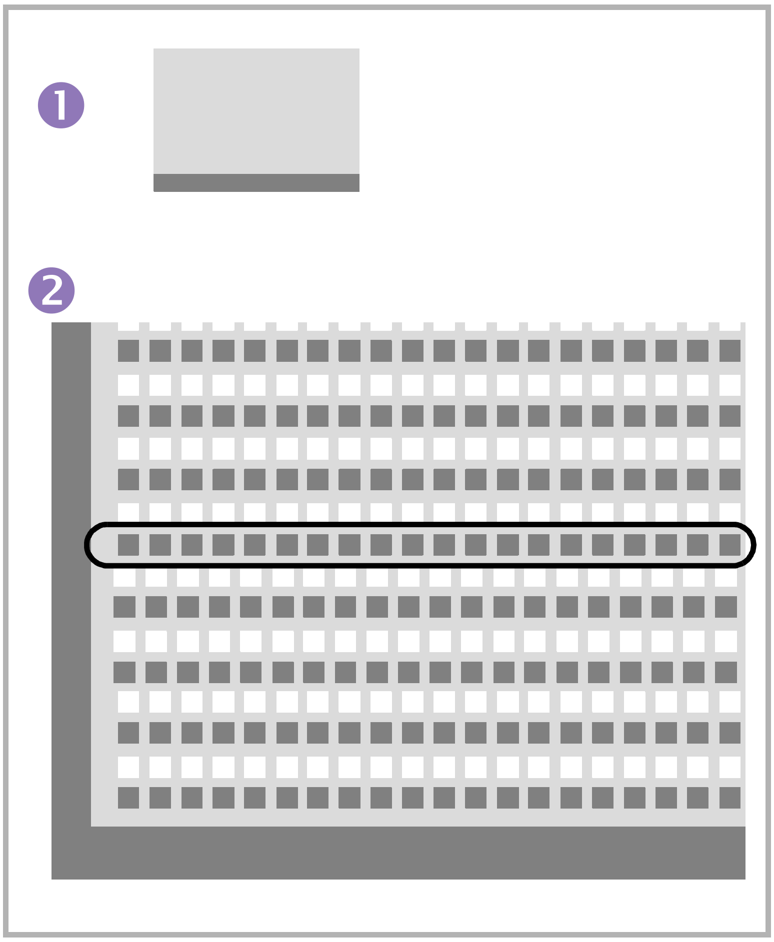
Determining the laser power for the plate

To determine the correct laser power setting for the plate:

1. Place the Calibration Strips job in front of you with the grip edge at the bottom, as shown in ❶.
2. Select any one row of 50% patches to measure, as shown in ❷ (it does not matter which row you choose).
3. Using the plate densitometer, and starting from either the left or right-hand side of the plate, measure each 50% patch in the selected row until you get a reading of approximately 52% for a silver positive-working plate.

   Note: For correct readings for other types of plate please contact a Python dealer.

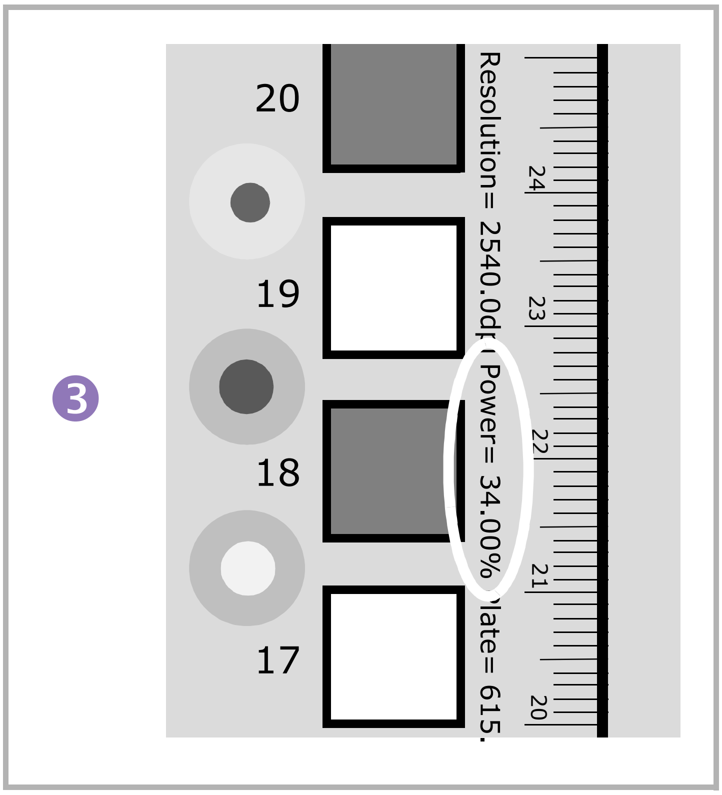
Continued overleaf...



4. Running down the right-hand edge of the correct patch is a vertical line of text, which includes the date, plate size, job name and resolution, as shown in ❸.

   This text also contains a laser power value (in this example, 34.00%), which is the correct setting for this plate.

   Note: If necessary, keep reading up/down the line of text until you reach the **Power** information.



5. For a test pass, this laser power reading should be within ±10% of the laser power setting on the label.

   For example, the laser power setting in the table on page 71 was 30%, so a power value between 20% and 40% for this plate would constitute a test pass.

   Note: If there is no power level label on the Python machine, then you should carry out this test without the reference values.

6. When you have finished, make a note of both the factory set laser power and the laser power that you have measured on the plate in “Test 1A – Laser Power” on page 118.

Note: If you are using different plate thicknesses and/or plates from other manufacturers, then carry out this laser power test on all those plates: output the Calibration Strips job on each plate type, as described on page 69, then carry out the laser power test on each plate.

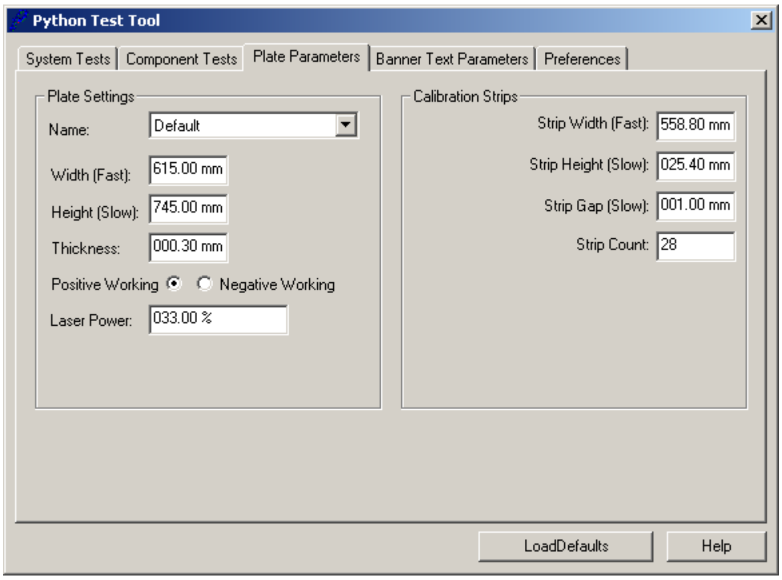
When you have determined the correct laser power setting for each plate you should return to the Python Layout Tool and enter this information for each plate definition.

Plate 2: Scaling

Plate Details	
Plate name	Scaling
Use plate type	Use the largest plate available (maximum 745 x 615mm, 0.3mm thick)
Job location on disk	D:\Python\Engine v5.x.x\calibration jobs\scaling\Scaling6_175.tif
Used in tests	Test 2A – Carriage Reference Setup on page 77 Test 2B – Fast Axis Scaling on page 79 Test 2C – Drum Reference on page 80 Test 2D – Plate Registration Skew on page 82 Test 2E – Focus and Spot Shape on page 83 Test 2F – Beam Bow on page 85
Notes	**Do not output this job until you have output the Calibration Strips job and determined the correct laser power for the plate (as described on page 69).** **Immediately after you have output the Scaling job to plate and BEFORE you start doing the tests on it, you should start outputting plate 3, the 'Scaling with Slur' job (see page 87), because of the amount of time it takes to output this job.**

To output the Scaling job:

1. In the Python Test Tool, select the **Plate Parameters** tab:

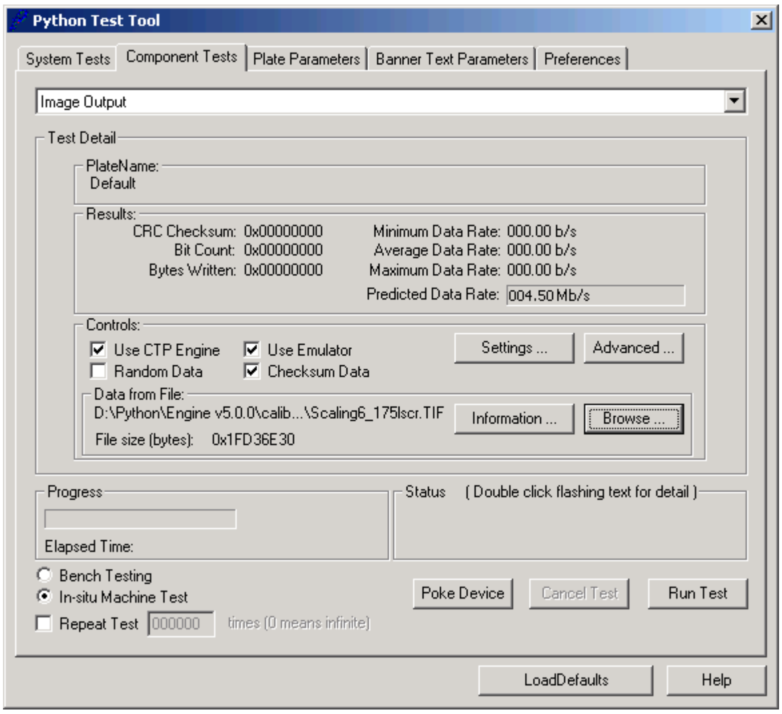


2. Select the required plate from the **Name** menu and check that the plate settings (**Width**, **Height, Thickness**, etc.) are correct.

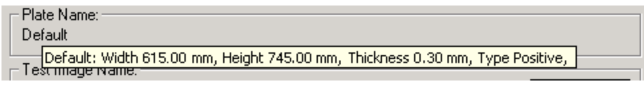
   Notes: The plate names available on this dialog are those that have been created in the Python Layout Tool. If the required plate is not available you can either create it in the Python Layout Tool (this is covered in chapter 8) or you can alter the plate values on this dialog to suit your requirements. If you do this, the values you enter will remain even when you re-launch the Python Test Tool program. To restore the values to their initial values (that is, those set in the Python Layout Tool), re-select the plate **Name** from the pull-down menu.

Any changes you make to the plate details here will not overwrite the plate details in the Python Layout Tool.

3. Do not adjust the Calibration Strips parameters.
4. Select the **Component Tests** tab then select **Image Output** from the drop-down menu to display the following:

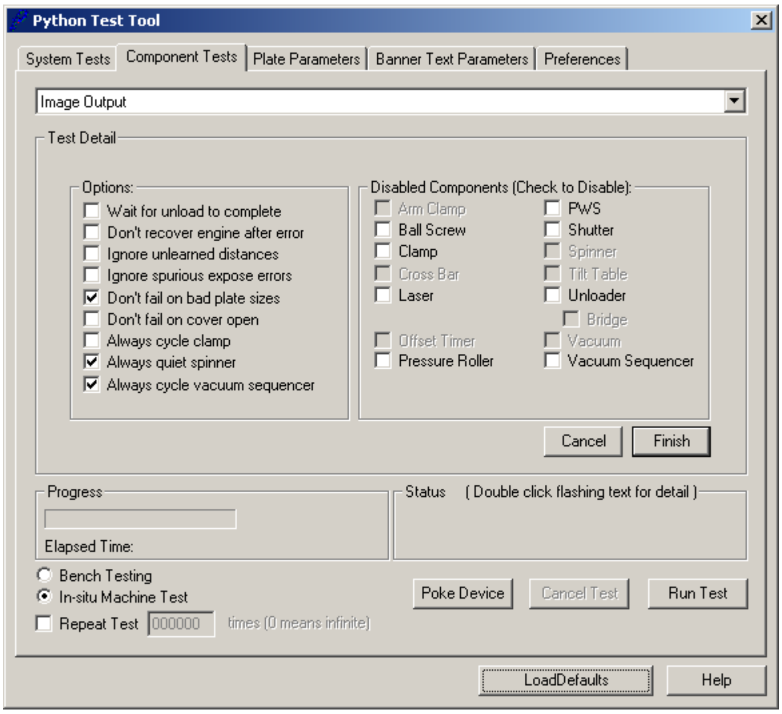


5. Check that the Plate Name and details are correct.
Note: To view the plate details, place the mouse cursor over the plate name:



6. Make sure that the **Use CTP Engine** and **Use Emulator** boxes are checked.
7. Make sure that the **Random Data** box is unchecked.
8. Click on the **Browse...** button to display the Open dialog.
9. From the **D:\Python\Engine v5.x.x\calibration jobs\scaling** folder select the **Scaling6_175lscr.TIF** file then click on the **Open** button.

10. Click on the **Advanced...** button to display the following dialog:

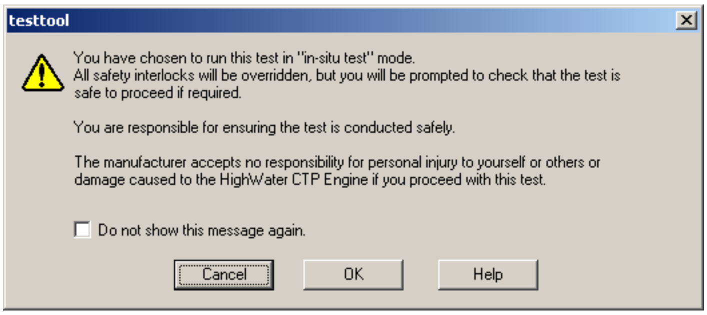


11. Click on the **Load Defaults** button.

12. Click on **Finish** to return to the previous dialog.

13. On the **Component Tests** tab, click on the **Run Test** button.

14. The following warning message will appear:



Warning: This dialog contains important safety information. Click on OK **to continue or on** Cancel **if you do not wish to proceed with the test.**

15. When prompted, load a plate and when it has been imaged, process it.

The following sections detail the tests you need to carry out on this plate.

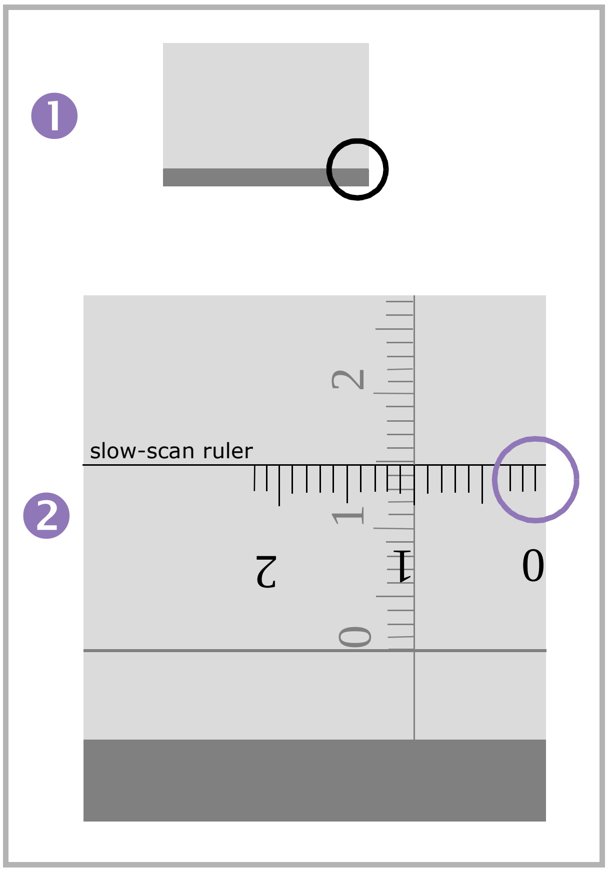
Test 2A – Carriage Reference Setup

Test Details	
Test purpose	Checks the image start position in the slow-scan direction
Test plate/job	Plate 2: Scaling job
Notes	None
Acceptable tolerance	**0–0.25mm**. A result between 0.75 and 1mm is a test pass.
If this test fails	Continue with the remaining tests but call a service engineer when you have finished all tests.

To check the Carriage Reference Setup:

1. Place the Scaling job in front of you with the grip edge at the bottom, as shown in ❶.
2. In the bottom right-hand corner of the plate you will see two rulers: one along the fast-scan and one along the slow-scan direction, as shown in ❷.

Continued overleaf...



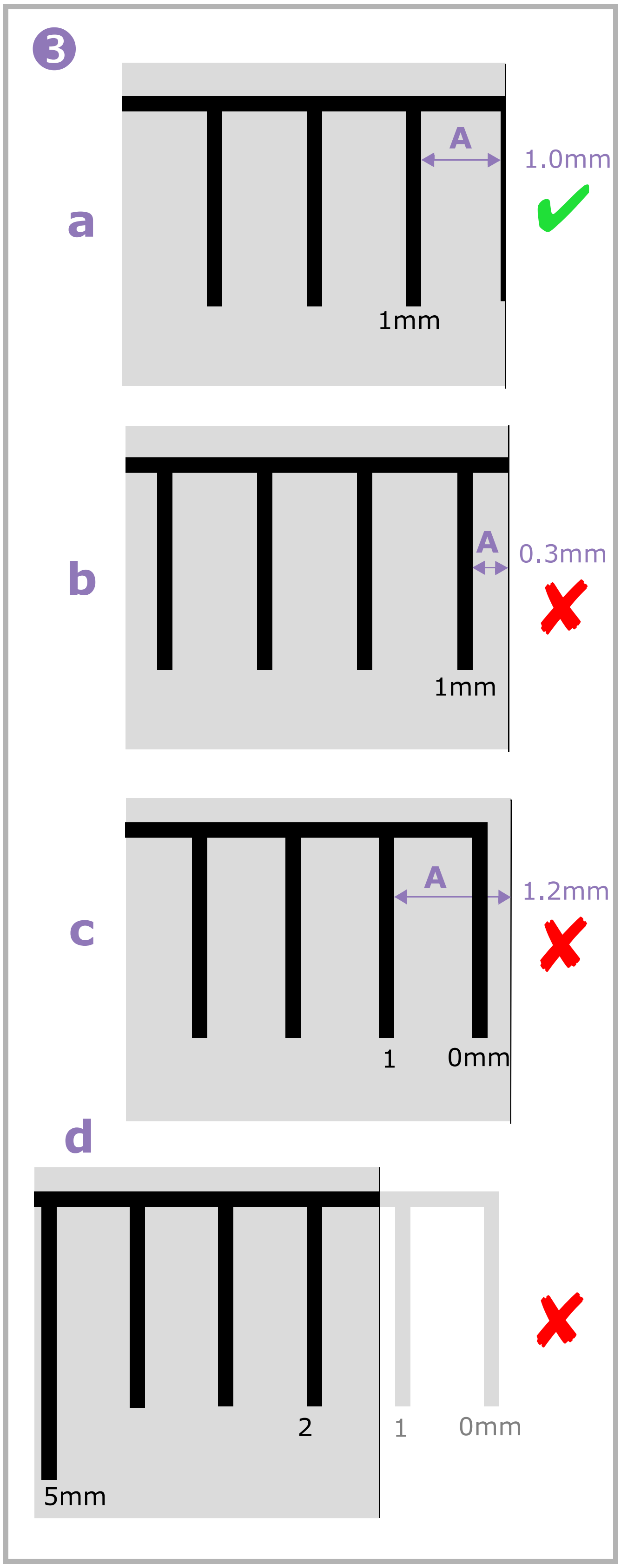
3. On the slow-scan ruler measure the distance between the right edge of the plate and the 1mm mark — that is, the distance **A**, as shown in **❸a**.

   Note: Make sure that there is a 1mm mark. If the first visible mark is the 2mm (or more) mark, as shown in **❸d**, then the test has failed.

   A tolerance of +0–0.25mm is acceptable for this test, so the width of **A** should be between 0.75 and 1mm for a test pass.

   Note: In the examples shown on the right, **❸b**, **c** and **d** all constitute a test fail because the measurement from the edge of the plate to the 1mm mark is not between 0.75 and 1mm, or because there is no 1mm mark on the plate.

4. When you have finished, record the results in "Test 2A – Carriage Reference Setup" on page 119.



Test 2B – Fast Axis Scaling

Test Details	
Test purpose	This test checks that the image height (that is, in the fast-scan direction) is accurate.
Test plate/job	Plate 2: Scaling job
Notes	If you have not used the largest plate size then the ruler will not run to 58cm, in which case you should measure to the furthest centimetre mark (for example, 46cm).
Acceptable tolerance	**±0.1mm**
If this test fails	Continue with the remaining tests but call a service engineer when you have finished all tests.

To check the Fast Axis Scaling:

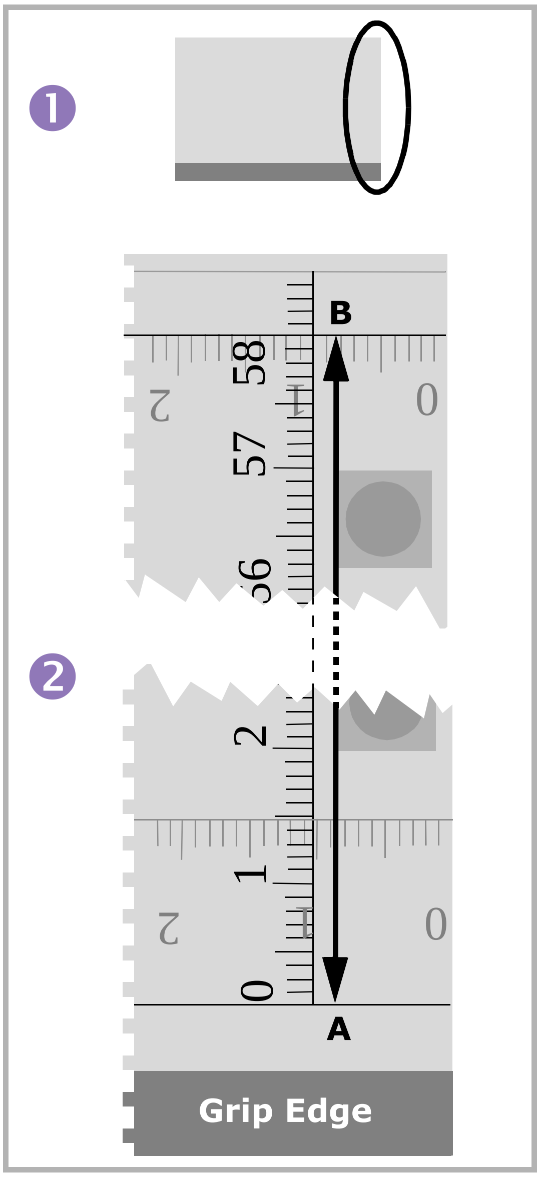
1. Place the Scaling job in front of you with the grip edge at the bottom, as shown in ❶.
2. Running down the right-hand side of the plate you will see a ruler labelled from 0 to 58cm, as shown in ❷.

   Note: If you have not used the maximum size plate, the ruler may not be as long as 58cm.
3. Place the 1 metre steel ruler alongside the ruler on the plate.

   Note: Align the 0cm mark on the steel rule **exactly** with the 0cm mark on the plate, as described in section 10.2 on page 67.
4. **Very carefully** measure the distance between the 0cm and 58cm marks on the plate (or other chosen mark, if the ruler is not 58cm long), as shown in ❷, between points **A** and **B**.

   Note: It may help to use a x10 eye glass to get a more accurate ruler measurement.
5. At the top of the plate where the 58cm (or other) mark lies, carefully read off the value on the steel rule.

   For a pass, this measurement should read 58cm ±0.1mm (or the length you have measured ±0.1mm).
6. When you have finished, record the test results in "Test 2B – Fast Axis Scaling" on page 120.

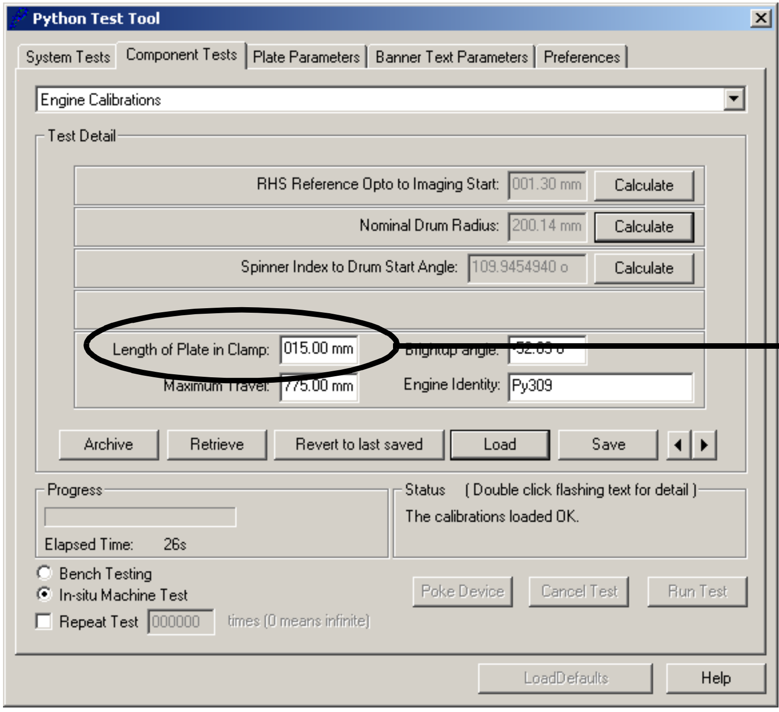


Test 2C – Drum Reference

Test Details	
Test purpose	This test checks the fast-scan start position of the laser relative to the front edge of the drum
Test plate/job	Plate 2: Scaling job
Notes	None
Acceptable tolerance	**±0.1mm**
If this test fails	Continue with the remaining tests but call a service engineer when you have finished all tests.

First, you need to get the drum reference value for the Python engine. To do this:

1. In the Python Test Tool, click on the **Component Tests** tab then select **Engine Calibrations** from the pull-down menu.
2. Click on the **Load** button to get the information about the Python engine.
3. Make a note of the **Length of Plate in Clamp** value (you will compare this to the actual measurement taken from the plate):



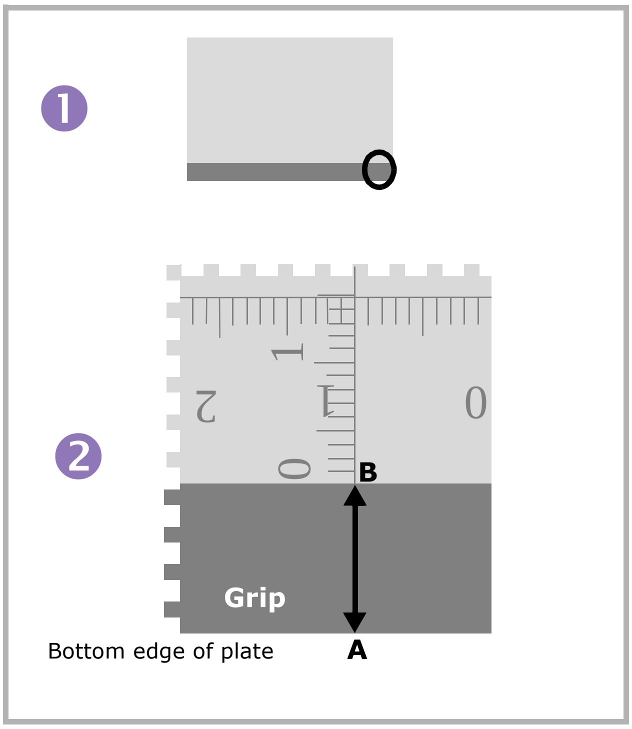
'Length of Plate in Clamp' value

Note: If you don't see the above dialog, click on the arrow buttons to cycle through the available screens until you get to this one.

To measure the actual 'Length of Plate in Clamp' value on the plate:

1. Place the Scaling job in front of you with the grip edge at the bottom, as shown in ❶.
2. In the bottom right-hand corner of the plate, **very carefully** measure the distance between the clamped edge of the plate and the 0mm mark on the plate, as shown in ❷, between points **A** and **B**.

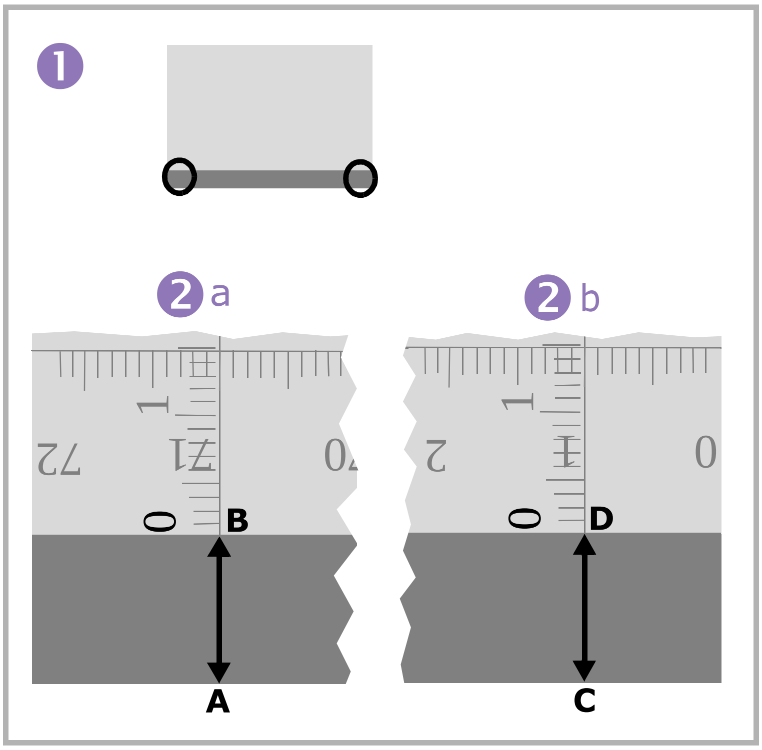
   For a pass, this measurement should be accurate to within ±0.1mm. For example, if the 'Length of Plate in Clamp' value obtained from the Python Test Tool is 15mm, then an actual measurement between 14.9 and 15.1mm is acceptable.
3. When you have finished, record the test results in "Test 2C – Drum Reference" on page 121.



Test 2D – Plate Registration Skew

Test Details	
Test purpose	This test checks that the plate is sitting squarely in the clamp
Test plate/job	Plate 2: Scaling job
Notes	None
Acceptable tolerance	**±0.1mm**
If this test fails	Continue with the remaining tests but call a service engineer when you have finished all tests.

To check the Plate Registration Skew:



1. Place the Scaling job in front of you with the grip edge at the bottom, as shown in ❶.
2. Near the left-hand edge of the plate, **very carefully** measure the distance between the bottom edge of the plate and the 0cm mark, as shown in ❷**a**, between points **A** and **B**.
3. Near the right-hand edge of the plate, **very carefully** measure the distance between the bottom edge of the plate and the 0cm mark, as shown in ❷**b**, between points **C** and **D**.

   For a pass, the measurement **AB** should be within ±0.1mm of measurement **CD**. For example, if **AB** is exactly 15mm, then **CD** must be between 14.9 and 15.1mm.
4. When you have finished, record the test results in "Test 2D – Plate Registration Skew" on page 122.

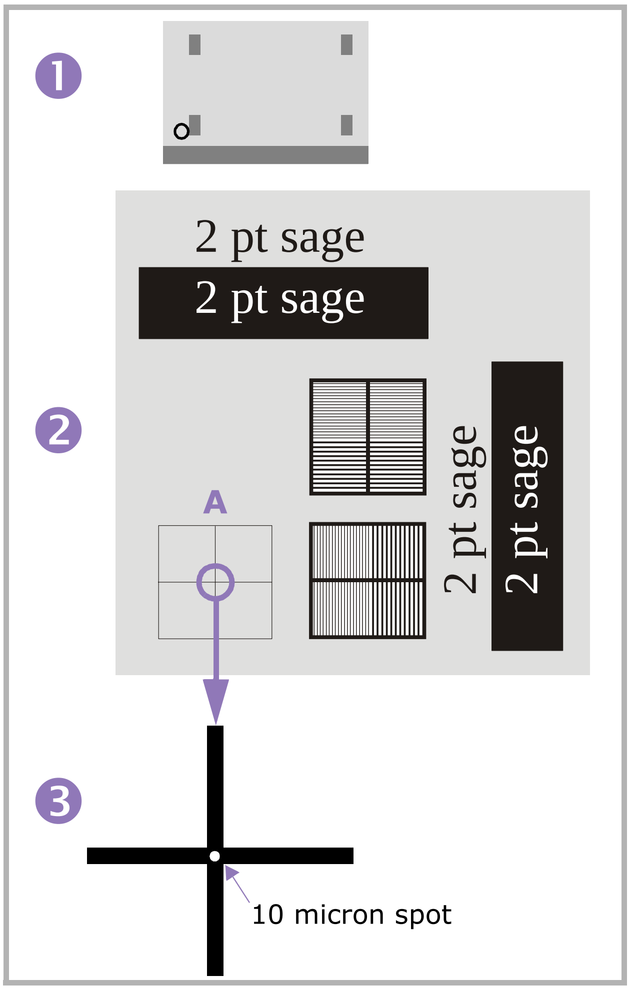
Test 2E – Focus and Spot Shape

Test Details	
Test purpose	These checks are used to judge the laser focus
Test plate/job	Plate 2: Scaling job
Notes	You will not take any measurements in this section. Instead, you will be making a visual assessment of the laser's focus.
Acceptable results	• The 10 micron spot should be visible. • Lines should be sharp and easily distinguishable from each other. • There should be a visible (but not severe) step at the 50% point in the vignettes. • The 10-40µm lines should be of equal width and there should be no areas where the lines disappear.
If this test fails	Continue with the remaining tests but call a service engineer when you have finished all tests.

To carry out the Focus and Spot Shape checks:

1. Place the Scaling job in front of you with the grip edge at the bottom, as shown in ❶.
2. Locate one of the four black rectangles, as shown in ❶. To the left of this rectangle you will see some boxes and 2pt text, as shown in ❷.
3. In the middle of box A, where the two lines intersect, there should be a visible 10 micron spot, as shown in ❸.

Continued overleaf...



4. Locate the 2 large vignettes on the plate, as shown in 4.

   On a correctly set and focused laser there should be a noticeable (but not severe) visible step at the 50% point.
   Note: This occurs when the corners of the Euclidean dot start to bleed into each other.

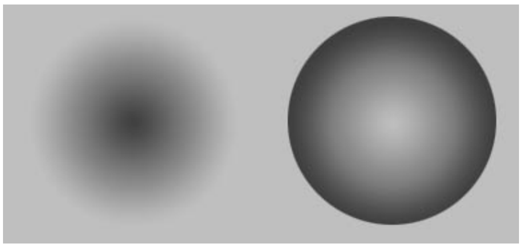
   If there is no apparent step at 50%, this may indicate poor laser focus.

5. Locate the concentric circles and 40µm lines, as shown in 5 and 6.

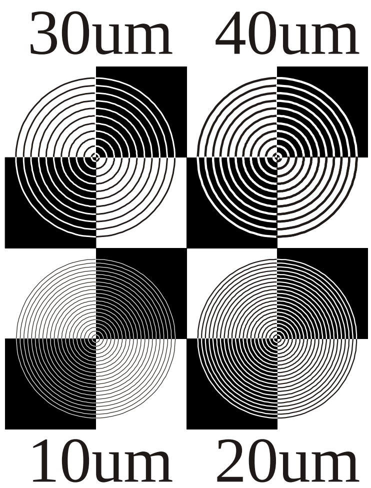
   The spot quality is affected by the laser switching times. At the correct power, a typical Python laser and optics should be capable of resolving 10µm lines and dots. Ideally, the white and black lines should be of equal thickness in both the slow and fast axes, as shown in 5a and 6a.

6. When you have finished, record the test results in "Test 2E – Focus and Spot Shape" on page 123.

4

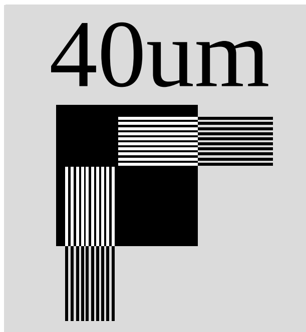


5

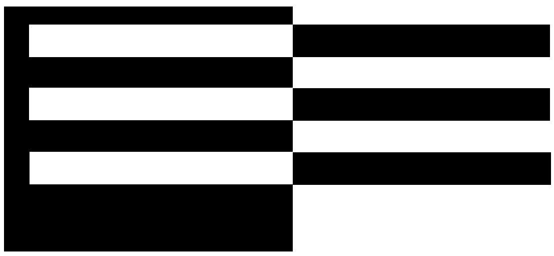


5 a

6



6 a



Test 2F – Beam Bow

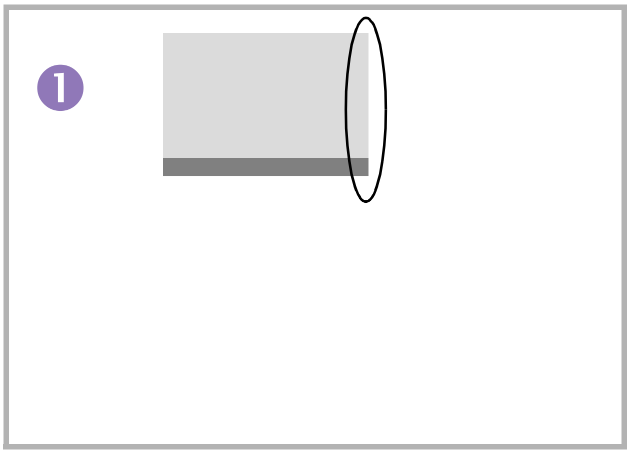
Test Details	
Test purpose	This test checks if the laser scans absolutely perpendicular to the drum surface
Test plate/job	Plate 2: Scaling job
Notes	You cannot carry out this test accurately if the plate you are working with does not have square edges.
Acceptable tolerance	**±0.1mm**
If this test fails	Continue with the remaining tests but call a service engineer when you have finished all tests.

To carry out the Beam Bow check:

1. Place the Scaling job in front of you with the grip edge at the bottom, as shown in ❶.
2. Running down the right-hand side of the plate you will see a ruler labelled from 0 to 58cm, as shown in ❷ overleaf.

   Note: If you have not used the maximum size plate, the ruler may not be as long as 58cm.

Continued overleaf...



3. You need to take three measurements from the right-hand edge of the plate to this ruler, as shown in ❷:
   - **A** Take one measurement at the bottom of the plate, near the 0cm mark.
   - **B** Take one measurement roughly in the middle of the plate, near the 29cm mark (or other mark if you did not use a full size plate).
   - **C** Take one measurement at the top of the plate, near the 58cm mark (or other mark if you did not use a full size plate).
4. If there is no significant bowing of the beam these three measurements **A**, **B** and **C**, should be within ±0.1mm of each other, as shown in ❸**a**. This constitutes a test pass.
5. If the beam is bowed then the measurements will not be within ±0.1mm of each other, as shown in ❸**b**.

   Notes: The beam bow has been exaggerated in this diagram. Typically, you would not be able to tell 'by eye' that the beam is bowed.

   If the plate you used was not square, this should be taken into account as it will affect the measurements taken.
6. When you have finished, record the results in "Test 2F – Beam Bow" on page 124.

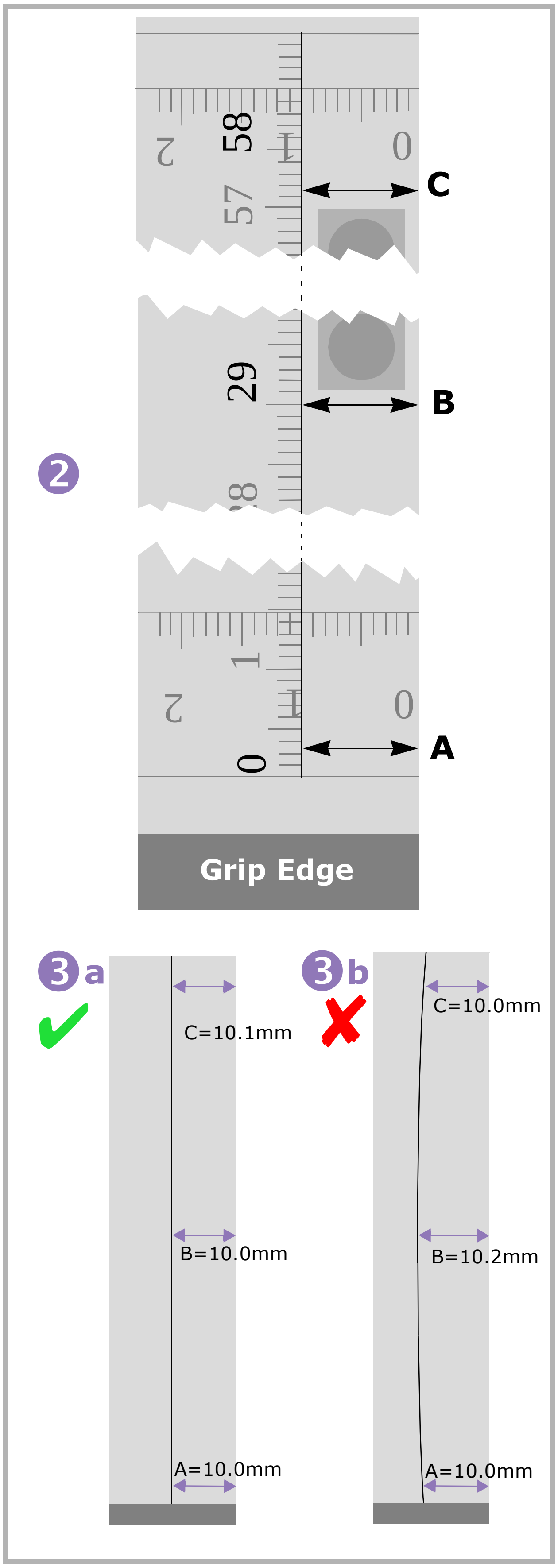
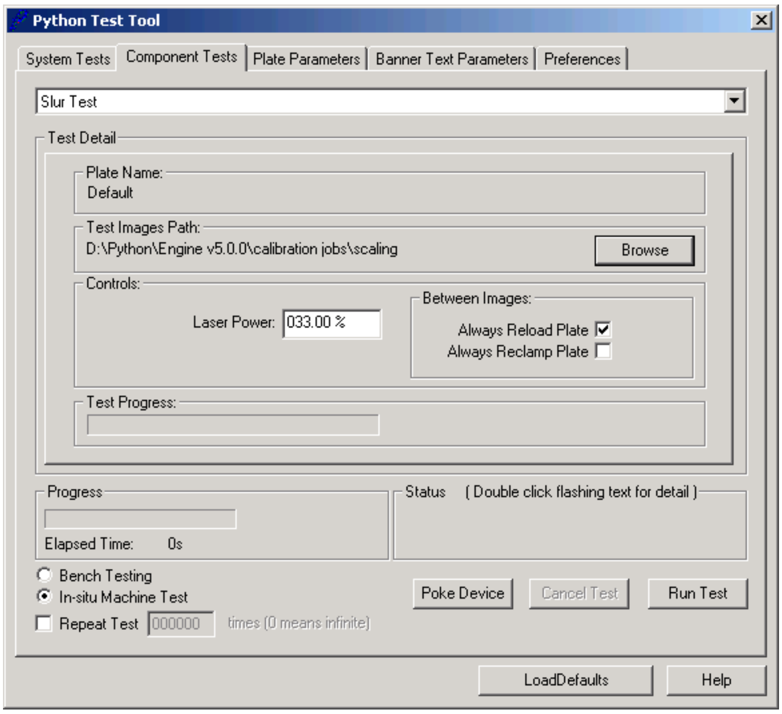


Plate 3: Scaling with Slur

Plate Details	
Plate name	Scaling with Slur
Use plate type	Use the largest plate available (maximum 745 x 615mm, 0.3mm thick)
Job location on disk	D:\Python\Engine v5.x.x\calibration jobs\scaling
Used in test	Test 3A – Plate Registration Repeatability on page 89
Notes	**This plate has a number of separate jobs imaged onto it. It takes approximately 30-40 minutes to image and process this plate so you should output it while you are carrying out the tests on plate 2, the Scaling job.** **Do not output this job until after you have output the Calibration Strips job and determined the correct laser power setting for the plates (as described on page 69).**

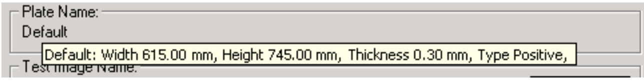
To output the Scaling with Slur job:

1. In the Python Test Tool, select the **Component Tests** tab.
2. Select **Slur Test** from the pull-down menu to display the following window:



3. Check that the plate name and details are correct.

   Note: To view the plate details, place the mouse cursor over the plate name:

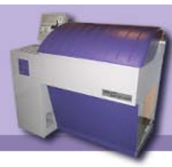
   

4. In the Test Images Path panel, click on the **Browse** button to open the Browse for Folder dialog.
5. Locate the **D:\Python\Engine v5.x.x\calibration jobs\scaling** folder then click on the **Open** button.
6. In the Controls panel, type in the correct **Laser Power** setting for the plate (this is the value that was determined in the laser power test on page 71).
7. In the Between Images panel, check the **Always Reload Plate** box but make sure the **Always Reclamp Plate** box is left unchecked.

   Note: Setting these options will ensure that the plate is unloaded then re-loaded into the drum for each image output, as required for this test, but it will not require you to unclamp and reclamp the plate between each output.
8. Click on the **Run Test** button.
9. When prompted to do so, load a plate onto the Python tilt-table and close the lid.
10. When the plate has been imaged, process it.

    Note: This plate takes approximately 30-40 minutes to image as a number of separate jobs are output onto one plate.

The next section describes the test to carry out on this plate.



Test 3A – Plate Registration Repeatability

Test Details	
Test purpose	The test plate has 8 separate jobs superimposed onto each other and it is used to check the accuracy and repeatability of the registration
Test plate/job	Plate 3: Scaling with Slur
Notes	You must be able to read vernier scales for this test
Acceptable tolerance	**±0.05mm (50μm) in each axis**
If this test fails	Continue with the remaining tests but call a service engineer when you have finished all tests.

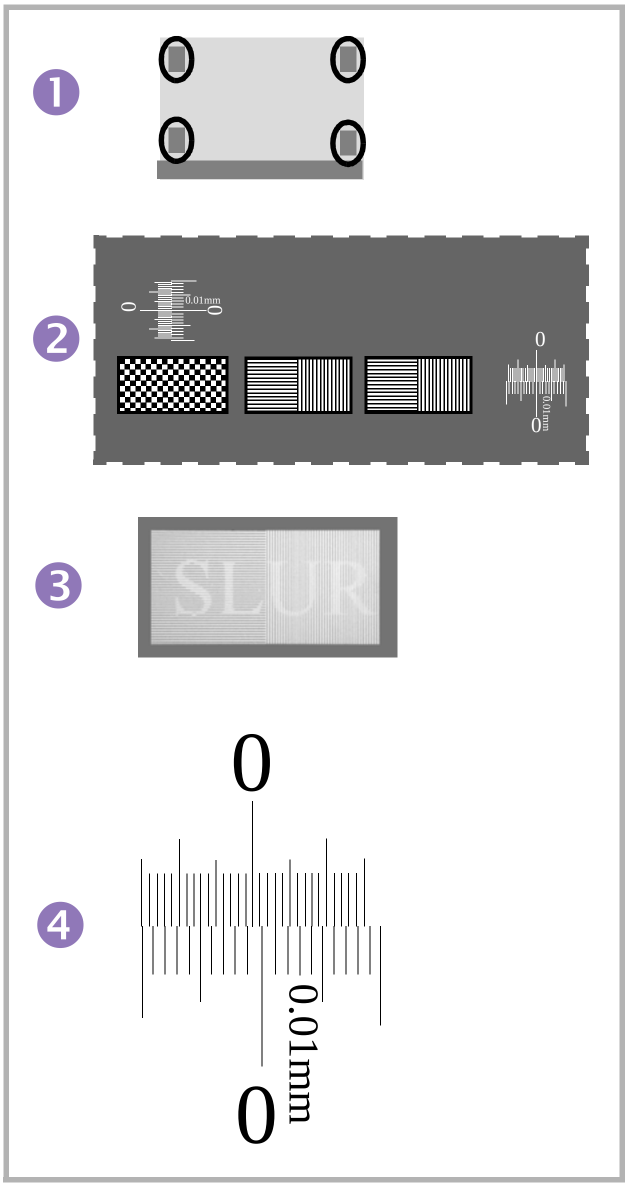
To carry out the Plate Registration Repeatability test:

1. Place the Scaling with Slur job in front of you with the grip edge at the bottom, as shown in ❶.
2. Locate one of the four black rectangles shown in ❶. In this rectangle you will see boxes of 50% tints made up from a 45 degree screen, fine horizontal lines and fine vertical lines. Next to these boxes you will see small vernier scales in the slow and fast-scan directions, as shown in ❷.
3. In the 50% tint boxes, any variation in repeatability will result in the word SLUR appearing in the tints, as shown in ❸.

   Note: Typically, you *will* see the word SLUR appear in the tints. However, this does not necessarily mean that there is a repeatability problem with the plate.
4. Next, look at the vernier scales, shown in ❹, to determine the repeatability error.

   The fine scale on the vernier is in 0.1mm (or 100μm) units, and the coarse scale is calibrated to show a difference of 0.01mm.

   In the example shown in ❹, the 0 on the coarse scale just passes the first tick right of the zero on the fine scale. So the measure is greater than 100μm. The first tick on the coarse scale, right



of its 0, lines up exactly with the tick on the fine scale, so you can add 10µm to determine that the repeatability error is 110µm (which, for this test, would constitute a fail).

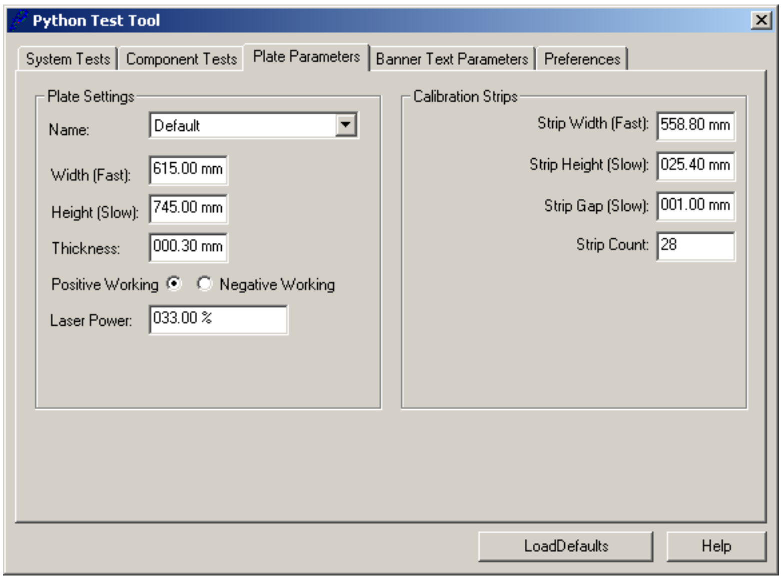
5. Find the repeatability error for both a vernier in the fast-scan and slow-scan direction.
6. Also, you should check one of the vernier scales in **each** of the four black rectangles shown in ➊. Widely differing measurements across the plate may indicate a repeatability or vacuum hold-down problem.
7. When you have finished, record the test results in “Test 3A – Plate Registration Repeatability” on page 125.

Plate 4: Full50

Plate Details	
Plate name	Full50
Use plate type	This job should be output on **every** plate size and thickness to be used by the customer. (The Flat Tint test, described on page 94, can highlight problems with vacuum hold down of the plate, which may vary with different sized plates.)
Job location on disk	D:\Python\Engine v5.x.x\calibration jobs\Full50v2_175lscr.TIF
Used in test	Test 4A – Flat Tint on page 94
Notes	**Do not output this job until you have output the Calibration Strips job and determined the correct laser power setting for the plates (as described on page 69).**

To output the Full50 plate:

1. In the Python Test Tool, select the **Plate Parameters** tab:



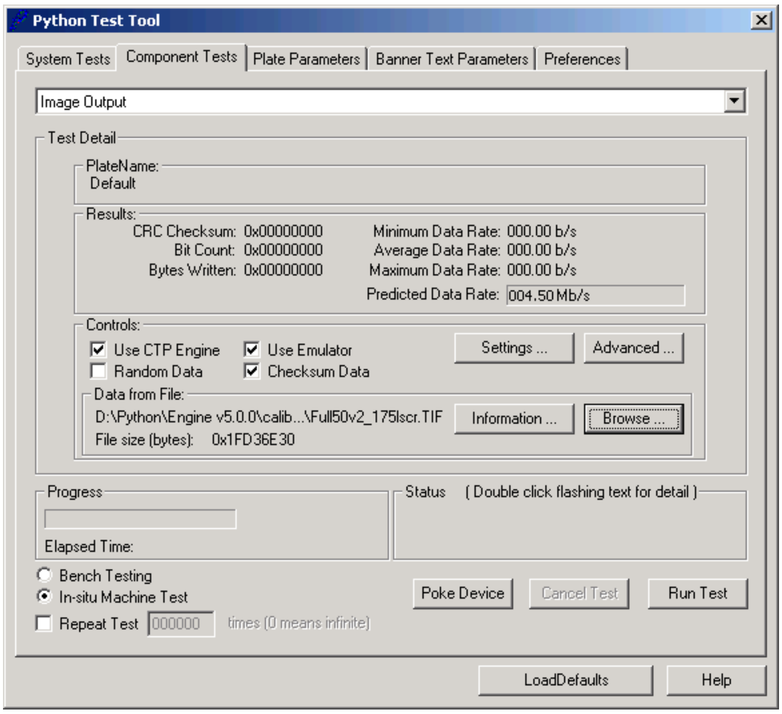
2. Select the required plate from the **Name** menu and check that the plate characteristics (**Width**, **Height, Thickness**, etc.) are correct.

   Notes: The plate names available on this dialog were created in the Python Layout Tool. If the required plate is not available you can either create it in the Python Layout Tool (this is covered in chapter 8) or you can alter the plate values on this dialog to suit your requirements. If you do this, the values you enter will remain even when you re-launch the Python Test Tool program. To restore the values to their initial values (that is, those set in the Python Layout Tool), re-select the plate **Name** from the pull-down menu.

   Any changes you make here to the plate details will not overwrite the plate details in the Python Layout Tool.

3. Do not adjust the Calibration Strips parameters.

4. Next, select the **Component Tests** tab and, from the drop-down menu, select **Image output** to display the following:

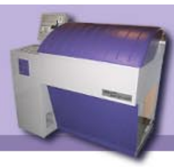


5. Make sure that the **Use CTP Engine** and **Use Emulator** options are checked.

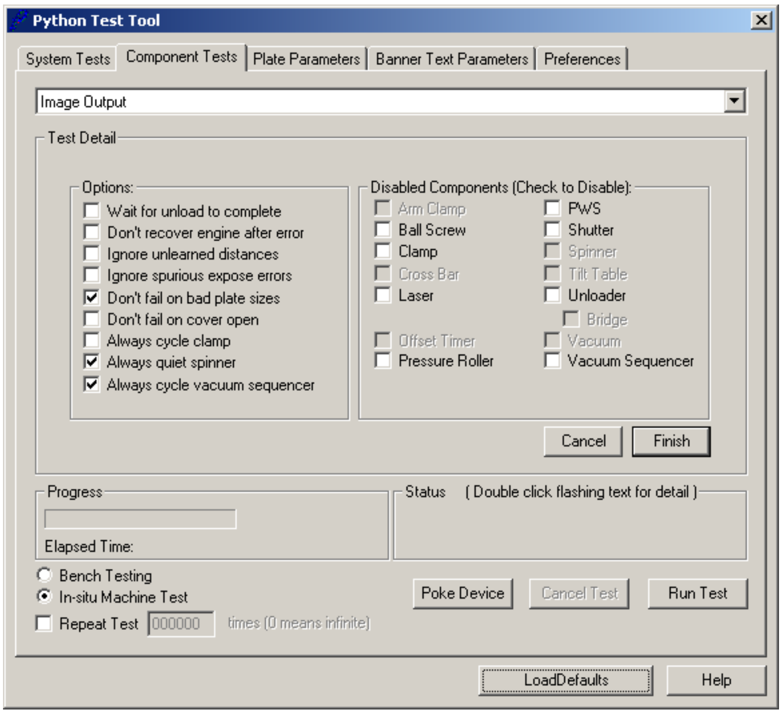
6. Make sure that the **Random Data** option is unchecked.

7. Click on the **Browse...** button to display the Open dialog.

8. From the **D:\Python\Engine v5.0.0\calibration jobs** folder select the **Full50v2_175lscr.TIF** file then click on the **Open** button.



9. Click on the **Advanced...** button to display the following dialog:

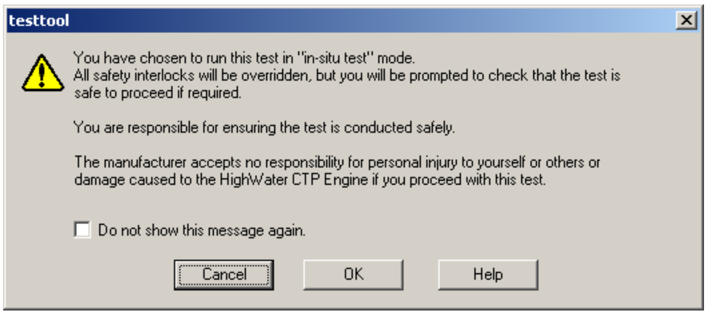


10. Click on the **Load Defaults** button.

11. Click on **Finish** to return to the previous dialog.

12. On the **Component Tests** tab, click on the **Run Test** button.

13. The following warning message will appear:



Warning: This dialog contains important safety information. Click on OK **to continue or on** Cancel **if you do not wish to proceed with the test.**

14. Load a plate and, when it has been imaged, process it.

The following section details the test you need to carry out on this plate.

Test 4A – Flat Tint

Test Details	
Test purpose	The flat tint test checks for any variation in the flat tint across the plate
Test plate/job	Plate 4: Full50
Notes	None
Acceptable tolerance	**No banding in the slow or fast axes** **±2% variation across the plate** **52% (or other) measurement across plate, depending on plate type**
If this test fails	Call a service engineer.

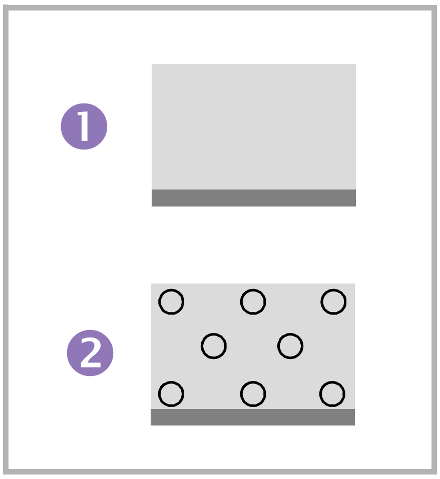
To carry out the Flat Tint test:

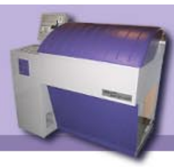
1. Place the Full50 job in front of you with the grip edge at the bottom, as shown in ❶.
2. There are 3 checks you should make on this plate:
   i. Look for a noticeable 2mm banding in the slow or fast-scan direction. This occurs if the carriage is not traversing smoothly across the plate.

   ii. At various locations on the plate (typically, those shown in ❷) use the densitometer to take measurements. All values should be within 2% of each other. If they are not, this may indicate a problem with the vacuum hold-down.

   iii. The measurements taken in (ii) should measure 52% for a positive-working silver plate to a tolerance of ±2% (for correct readings for other types of plate please contact a Python dealer).

   If the measurements are not the required value (depending on plate type) then repeat the laser power check (on page 71) and output the Full50 job again using the revised laser power.
3. When you have finished, record the test results in "Test 4A – Flat Tint" on page 126.





11. Setting up the Python software

Once the Python system has been installed and checked you need to set up the HighWater software applications that enable the user to output jobs to the Python engine. This section gives an overview of setting up the software for the user — full instructions can be found in the in the **Python User Guide**.

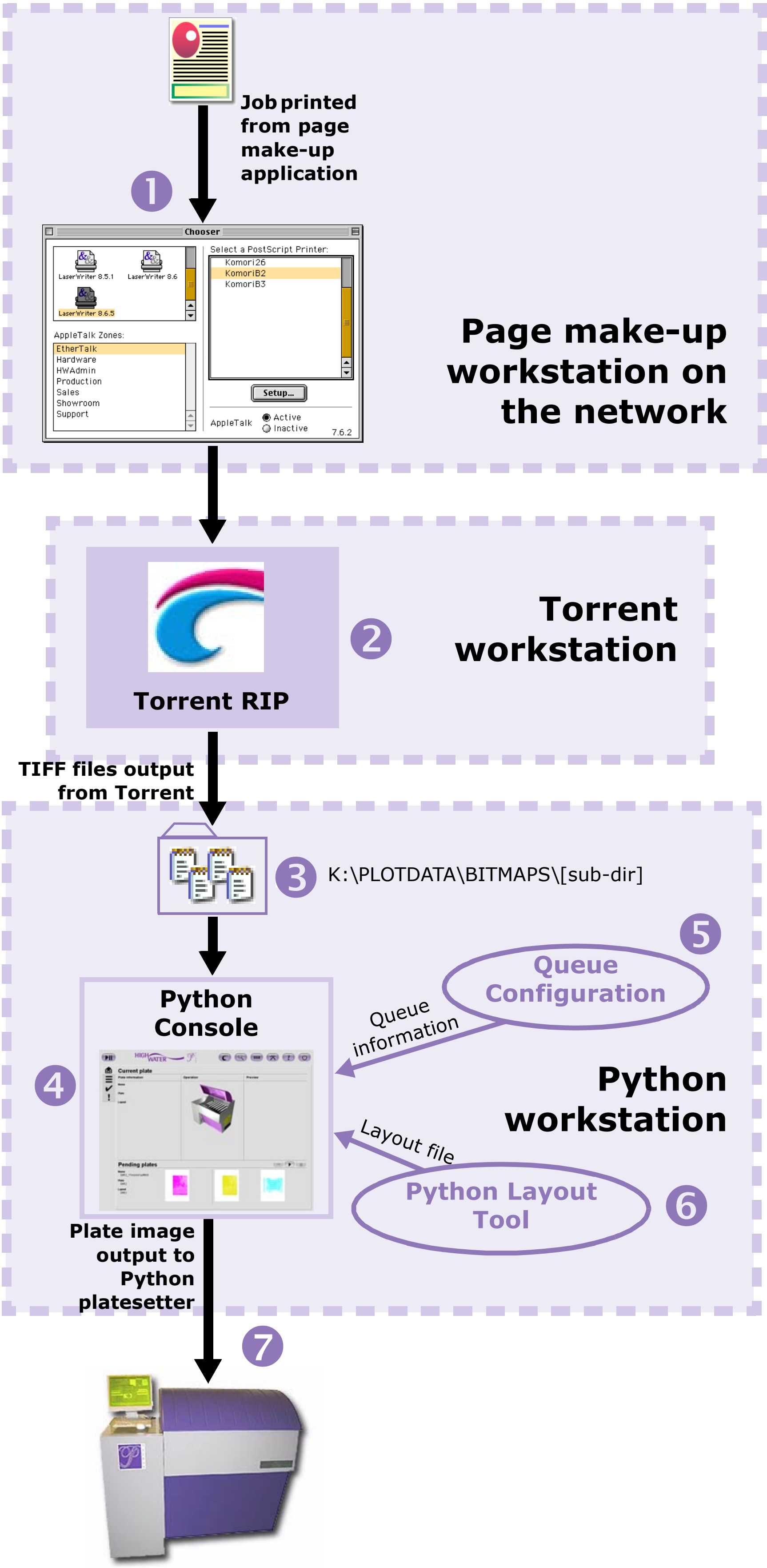
This chapter contains the following sections:

- 11.1, Job flow through the Python system (p95).
- 11.2, Setting up the software (p97).

11.1 Job flow through the Python system

The diagram on the following page shows the typical job flow through the Python system.

Note: The diagram shows the Python system with the two-computer configuration (that is, the Torrent RIP running on a separate workstation on the network). Some users may have a one-computer system configuration (that is, with Torrent running on the Python PC).



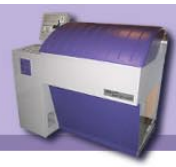
1
Printing the job
You print a job to Torrent from a page make-up workstation (either Mac or PC) on the network, via a Python print queue.

2 3
The Torrent RIP
Jobs are processed by the Torrent RIP and output as TIFF bitmap files to a sub-directory of K:\PLOTDATA\BITMAPS on the Python PC.

4
Python Console
These bitmap TIFF files are moved into the appropriate queue in the Python Console, ready for processing.

Note: The Python Console gets the queue names and other processing details from the Queue Configuration application (5), and the plate and job information from the appropriate Layout file created in the Python Layout Tool (6).

7
Python platesetter
Finally, the plate image is output to the Python platesetter.



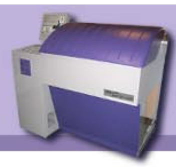
11.2 Setting up the software

Setting up the various Python-related software applications for the user is summarised in the table below and the particular chapter or section to refer to in the **Python User Guide** for further instructions is given.

Note: v4.0.0 and later Python systems ship with a set of pre-configured plate definitions, layout definitions, ICF files and Q2 queues. Therefore, it should only be necessary to delete from or add to these, as required, and step 1 specified below (creation of the Imager and Completed queues) should not be necessary unless the pre-configured settings do not exist.

	TASK	See Python User Guide
1	**Create the Imager and Completed queues** The Imager queue feeds jobs (coming from the TicketMaker queues) to the Python platesetter. Once jobs have been processed, they are moved to the Completed queue where they can be re-output, if required. **Note:** Usually, this step will not be required as the Imager and Completed queues should already exist on the new Python system. If not, then create these queues.	**Appendix A** **Appendix B**
2	**Determine how many workflows are required** The path of each job through the system is known as a ‘workflow’. A separate workflow is required for each different combination of settings: plate size; plate thickness; plate type (negative or positive); plate borders.	**Chapter 10**
FOR EACH REQUIRED WORKFLOW:		
3	**Choose a name for the workflow** Each workflow requires a unique name, usually based on the press model.	**Section 10.3**
4	**Create a page setup and input queue in the Torrent RIP** In the Torrent RIP, a page setup defines the settings (resolution, page size, etc.) to be applied to jobs sent to Torrent. The input queue lets you print jobs to Torrent from across the network.	**Chapter 11**
5	**Create a BITMAPS directory** TIFF files created by the Torrent RIP are saved here.	**Section 11.1**
6	**Create a Layout file in the Python Layout Tool** Layout files contain the plate and job position information that the Python Console needs for outputting jobs to the Python platesetter.	**Chapter 13**
7	**Create a queue using the Queue Configuration application** You create queues for the Python Console. Each queue specifies details such as where to look for jobs to process and job settings.	**Chapter 14**
8	**Optionally set up the Barcode Plate Requeue option** This option puts a barcode onto imaged plates so that the user can quickly locate jobs for re-making plates, if required.	**Appendix C**

9	**Create the relevant print queue on the page-make up workstation(s)** You need to set up 'virtual' printers on each page make-up workstation that the customer will be printing from. These printers allow the customer to print jobs directly to the Torrent RIP.	**Chapter 19**
10	**Calibrate the Torrent page setup** Calibrating the Torrent page setup is necessary to ensure output quality.	**Chapter 12**
WHEN YOU HAVE FINISHED:		
11	**Save the Python system settings** When you have finished creating the new workflow(s) you should save all the settings you have made.	**Section 16.2**



12. Handover to the customer

By now, you should have finished installing, testing and setting up the Python system. Assuming that everything is in satisfactory working order, you need to check that the system is made safe before final handover to the customer.

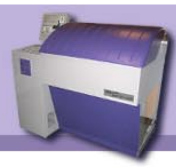
This chapter includes the following section:

- 12.1, Checking the system before handover to the customer (p99).

12.1 Checking the system before handover to the customer

When all the installation and plate tests have been satisfactorily completed and the software has been set up and tested, the Python engine must be checked before handover to the customer. Please carry out all the following checks:

Make the following checks:	✓
All the Python engine's covers are **securely** in place if they have been removed or loosened during the installation: • The carriage's optic covers. • The rear, left and right-hand side panels. • The hinged front panel.	
The Python engine's drum surface and tilt-table are clean and free of any debris. (If necessary, follow the cleaning procedure in section 6.2 on page 33.)	
There are no loose connections and all cables are correctly routed.	
The lid interlock override switch has been removed.	
The user: • Can load/unload a plate. • Can successfully print jobs to the Python system from a page make-up workstation. • Is aware of the potential for the Python engine to become damaged if care is not taken when loading plates if auto-unload is enabled with **no** communication link.	
The user has been instructed on the health and safety aspects of using the Python system and has read the supplied Python Safety Information manual.	



Appendix A: Plate specifications

This appendix gives plate information for the Python system including:

- A.1, Plate types (p101).
- A.2, Plate sizes (p102).
- A.3, Plate exposure times (p103).

Note: Please refer to section 2.2.3 on page 12 for safe plate handling guidelines.

A.1 Plate types

The plates used in the Python engine must be compatible with its laser expose system. Python has been tested with the following plate types:

Plate type	Pos/neg	Filter fitted	Plate type
Agfa Lithostar N91v	Negative	Yes*	Photopolymer
Fuji LP-NV	Negative	No	Photopolymer
Lastra LV2	Negative	No	Photopolymer
Silver plate	Positive	Yes	Silver

*This is a 0.4D filter.

Notes: The laser expose power must be determined for each of these plate types (this is covered in “Test 1A – Laser Power” on page 71).

Section B.1 lists suitable lighting for each plate type.

A.1.1 Positive and negative working plates

Plates can be exposed on Python in two different ways:

- **Negative working plates**. The laser exposes the image area which is to be printed on the plate. The remaining area is unexposed. When the plate is put through the processor, the unexposed area is etched away to leave bare plate.
- **Positive working plates**. The laser exposes the area which is not to be printed on the plate. When the plate is put through the processor, the exposed area is etched away to leave bare plate. This means that all positive working plates must be exposed up to the edges, regardless of the size of the image on the plate.

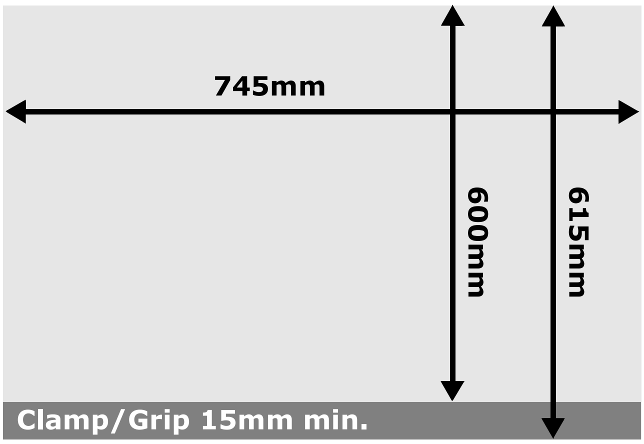
In either case, the physical appearance of the plate to go on the press will be the same (positive image, right reading), apart from the unexposed grip area.

A.2 Plate sizes

Python has been designed to support 4-up A4 layout. Plates between 0.15 and 0.3mm thickness are supported. Correct configuration of the plate thickness is vital as this affects the exact spinner speed in order to ensure correct sizing around the drum.

A.2.1 Maximum plate size

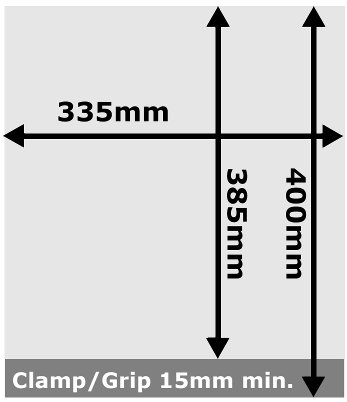
The maximum plate size is 745mm (along the grip edge) by 615mm. The minimum grip edge clamp depth is 15mm, which is not imaged so, typically, the maximum image area is 600 x 745mm, as shown below. Python can accommodate SM72 (724 x 615mm) and SM74 (745 x 605mm) plates.



Maximum plate size

A.2.2 Minimum plate size

The minimum plate size is 335mm (along the grip edge) by 400mm. The minimum grip edge clamp depth is 15mm, which is not imaged so, typically, the maximum image area of the smallest allowed plate is 335 x 385mm, as shown below. Python can accommodate QM46 (340 x 505mm) and GTO52 (510 x 605mm) plates.



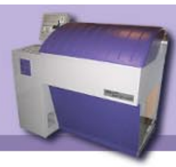
Minimum plate size

A.3 Plate exposure times

Python exposes at a fixed rate of 600 scan lines per second. The laser expose head traverses at 6mm per second. The plate handling time is approximately 1 minute, regardless of plate size.

The following table shows plate exposure times (not including plate load/unload time) for various plate sizes:

Plate size (mm)	Exposure Time
745 x 605	2 minutes 4 seconds
510 x 400	1 minute 25 seconds
340 x 505	57 seconds



Appendix B: Brightroom layout and environment

This appendix contains the following information:

- B.1, Required operating conditions (p105).
- B.2, Access space around the Python engine (p106).
- B.3, Typical brightroom layout (p107).

B.1 Required operating conditions

Python should be operated in a clean (office-like) environment. The atmosphere must be clear of contaminants that could cause harmful deposits on optical surfaces.

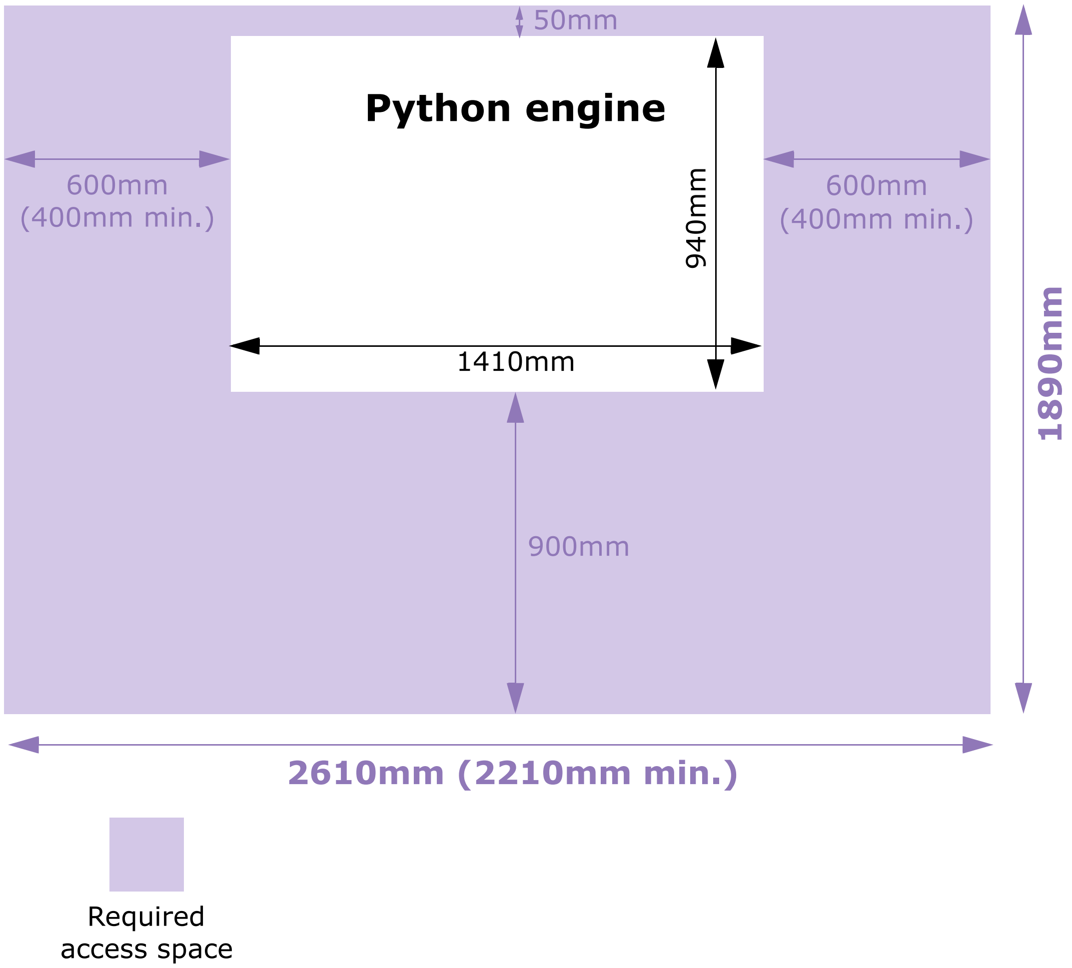
Python operates at a temperature range of 15–25°C and in a relative humidity of 10 to 80% non-condensing. Air conditioning is required for most installations.

For loading and unloading of plates, suitable lighting is required to prevent fogging, for example:

Plate	Lighting
Agfa Lithostar N91v	Osram L36 W62 and EncapSulite V50
Fuji LP-NV	G30
Lastra LV2	Encapsulite YG10
Silver plate	Encapsulite YG10

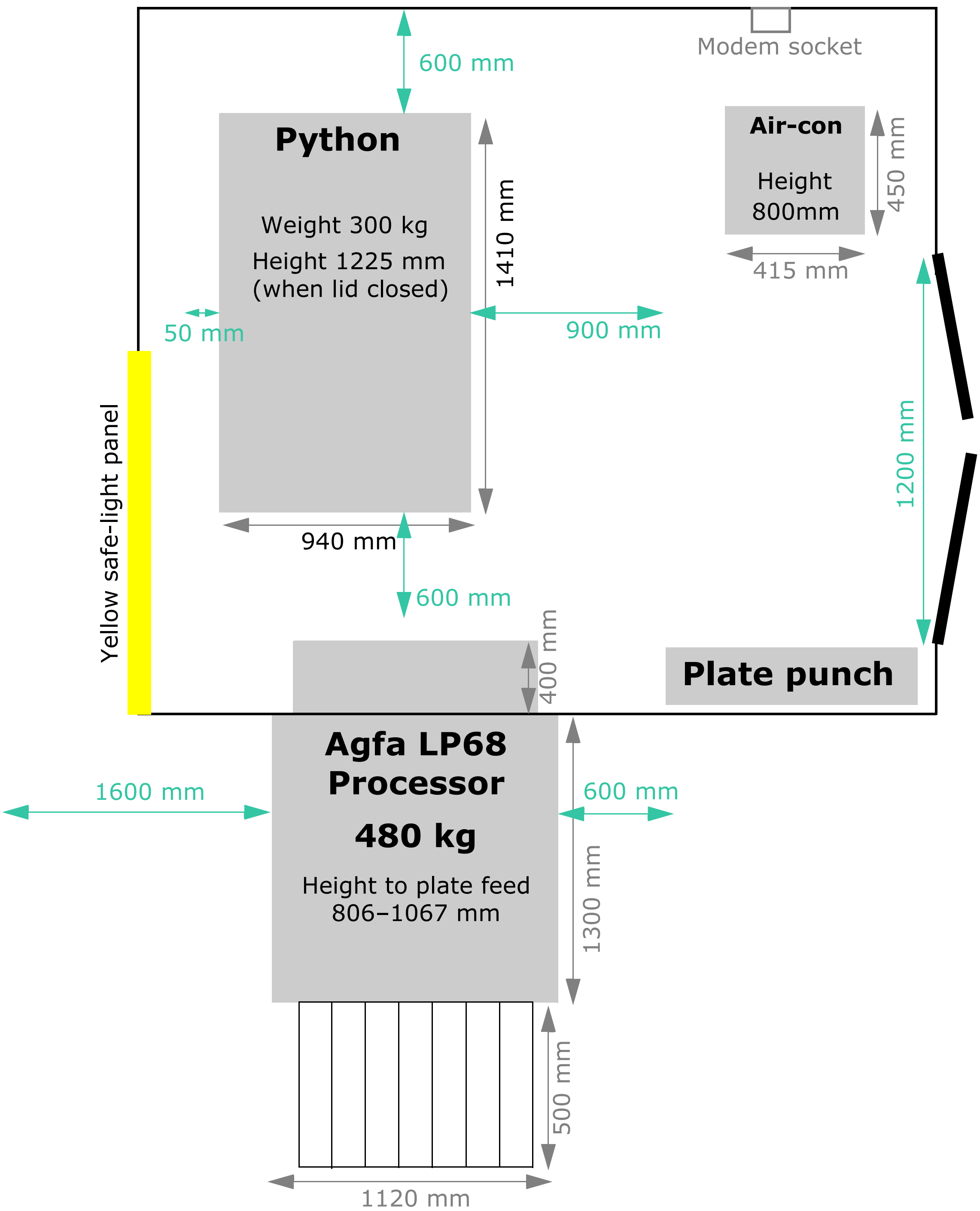
B.2 Access space around the Python engine

The following diagram shows the recommended and, where appropriate, minimum required access space around the Python engine in the brightroom:



B.3 Typical brightroom layout

The following diagram shows a typical brightroom layout for the Python engine:



Appendix C: Python safety labels

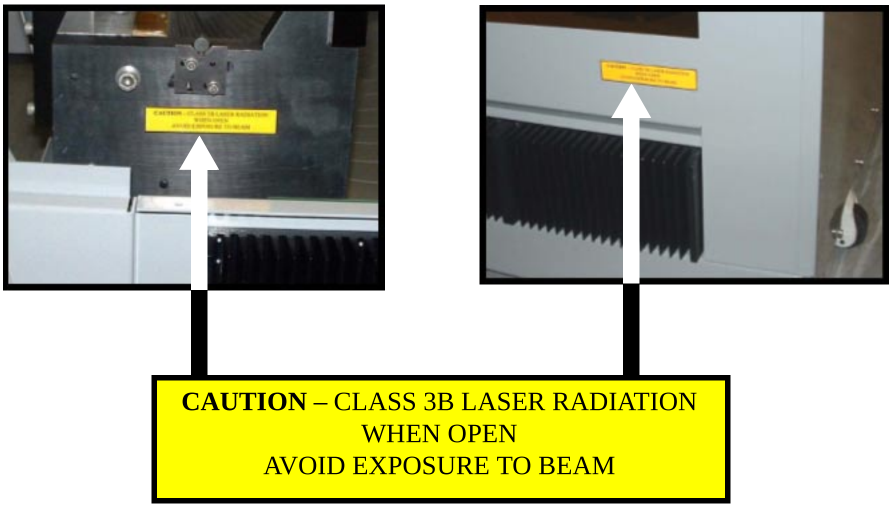
This appendix shows the labels that are present on the Python engine:

- C.1, On Python’s lid (inside and outside) (p109).
- C.2, On the right-hand side panel (inside and outside) (p110).
- C.3, On the rear panel (inside and outside) (p110).
- C.4, On the underside of the tilt-table (p111).
- C.5, On the carriage (p112).
- C.6, Product ID and certification labels (EN and IEC versions) (p113).
- C.7, MET label (p114).
- C.8, Other warning labels (p115).

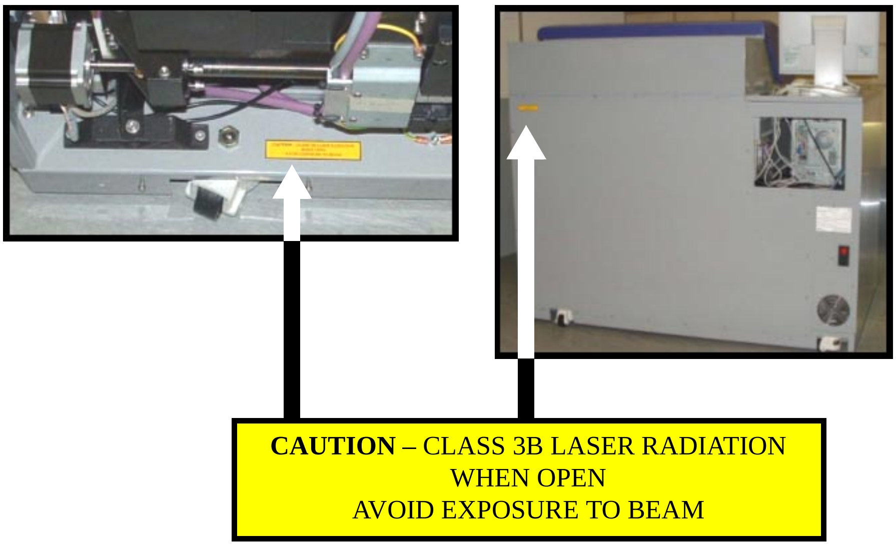
C.1 On Python’s lid (inside and outside)

CAUTION – CLASS 3B LASER RADIATION
WHEN OPEN AND INTERLOCKS DEFEATED
AVOID EXPOSURE TO BEAM

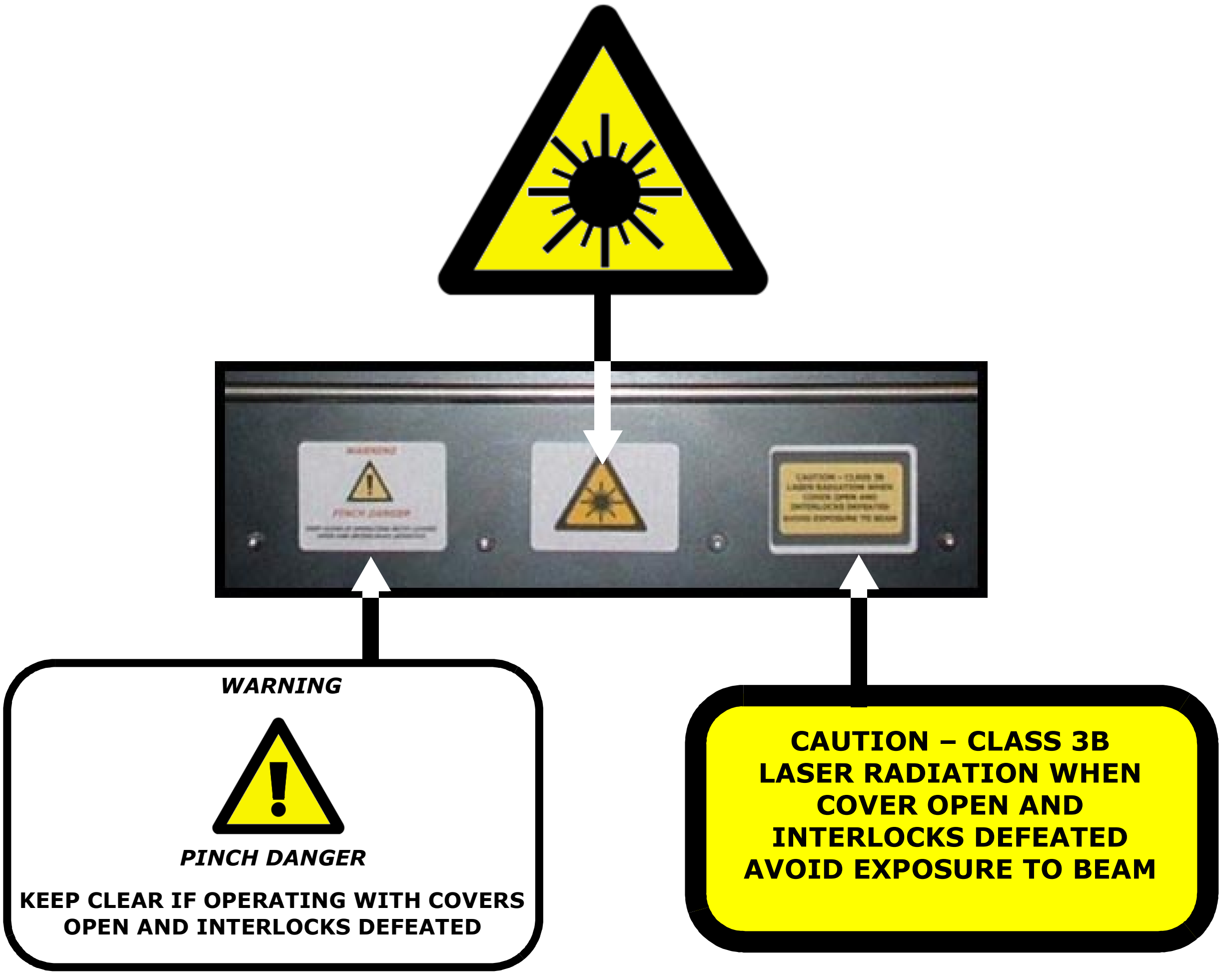
C.2 On the right-hand side panel (inside and outside)



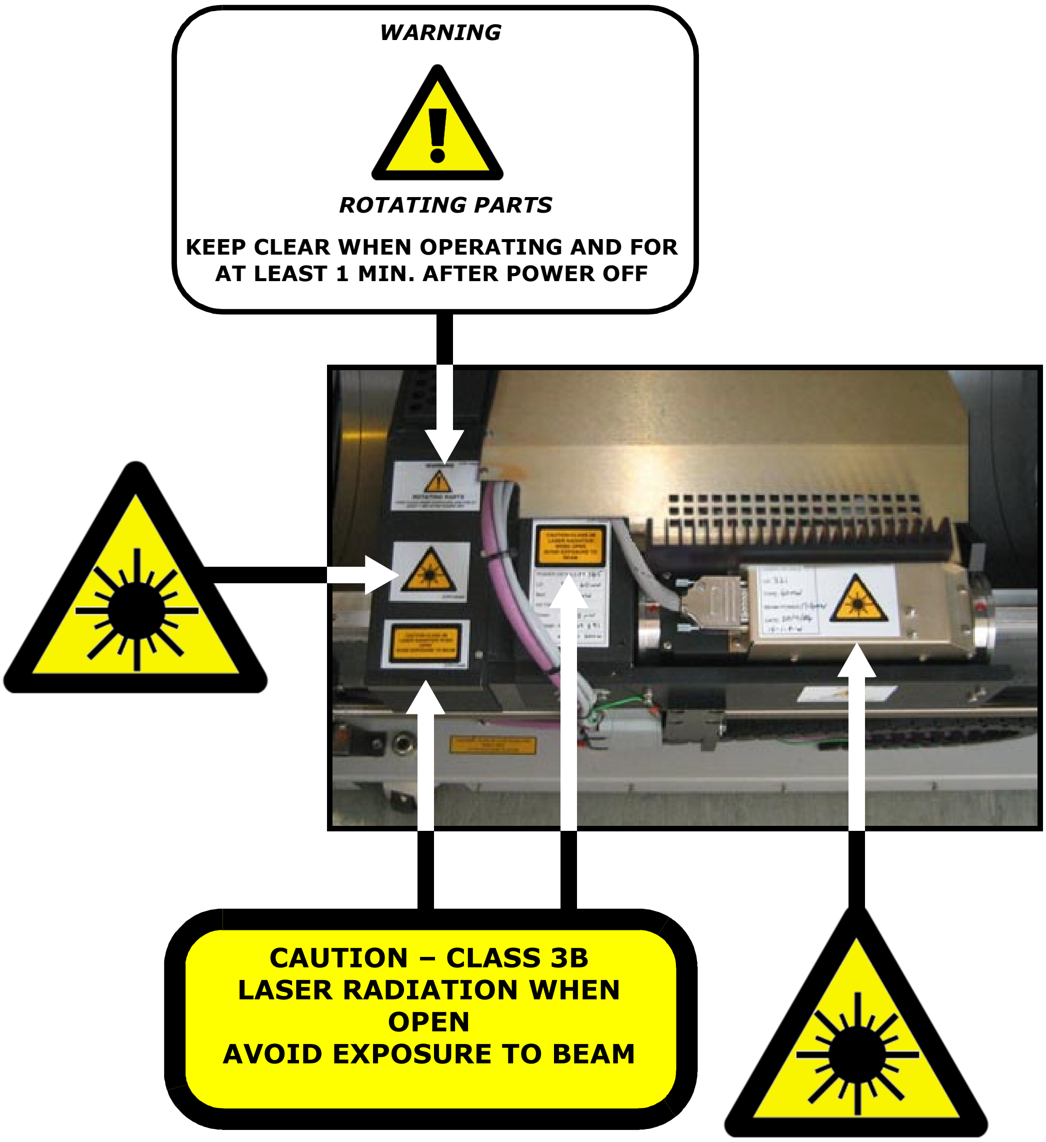
C.3 On the rear panel (inside and outside)



C.4 On the underside of the tilt-table



C.5 On the carriage



C.6 Product ID and certification labels (EN and IEC versions)

Python 74BV30EN

Computer to Plate Equipment

CE

Serial Number	PY 589
Date of Manufacture	October 2006

Rating 230V
50Hz
6 Amps
Fuse in IEC Inlet

CAUTION For continued protection against risk of fire, replace only with the same type and rating of fuse
ATTENTION Pour ne pas compromettre la protection contre les risques d'incendie, remplacer par un fusible de même type et des mêmes caractéristiques nominales.

HighWater Designs Ltd
Head Office: St George's Business Park, Alstone Lane, Cheltenham, Glos, GL51 8HF
Tel +44 (0) 1242-542100, Fax +44 (0) 1242-251600
http://www.highwater.co.uk, Email: Info@highwater.co.uk

See installation instructions before connecting to supply
Voir la notice d'installation avant de raccorder au reseau
Fuse 230V – T6.3AH250V

Complies with
BS EN 60825-1:1994+A1&A2

Python 74-BV30-IEC

Computer to Plate Equipment

SERIAL NUMBER	**PY 588**
Date of Manufacture	**October 2006**

Rating 110V
60Hz
10 Amps
Fuse in IEC Inlet

CAUTION: For continued protection against risk of fire, replace only with the same type and rating of fuse

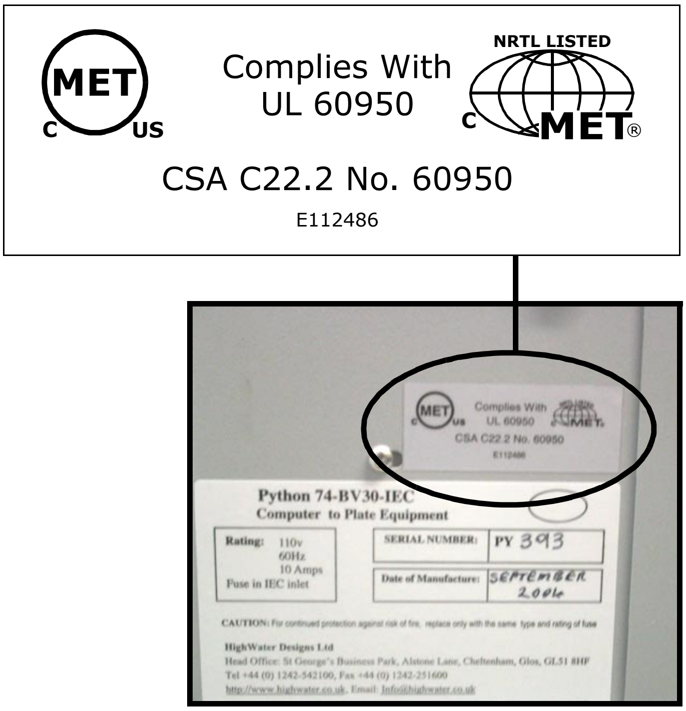
HighWater Designs Ltd
Head Office: St George's Business Park, Alstone Lane, Cheltenham, Glos, GL51 8HF
Tel +44 (0) 1242-542100, Fax +44 (0) 1242-251600
http://www.highwater.co.uk, Email: Info@highwater.co.uk

Complies with: BS EN 60825-1:1994 + A1 and A2 and with 21 CFR 1040.10 and 1040.11

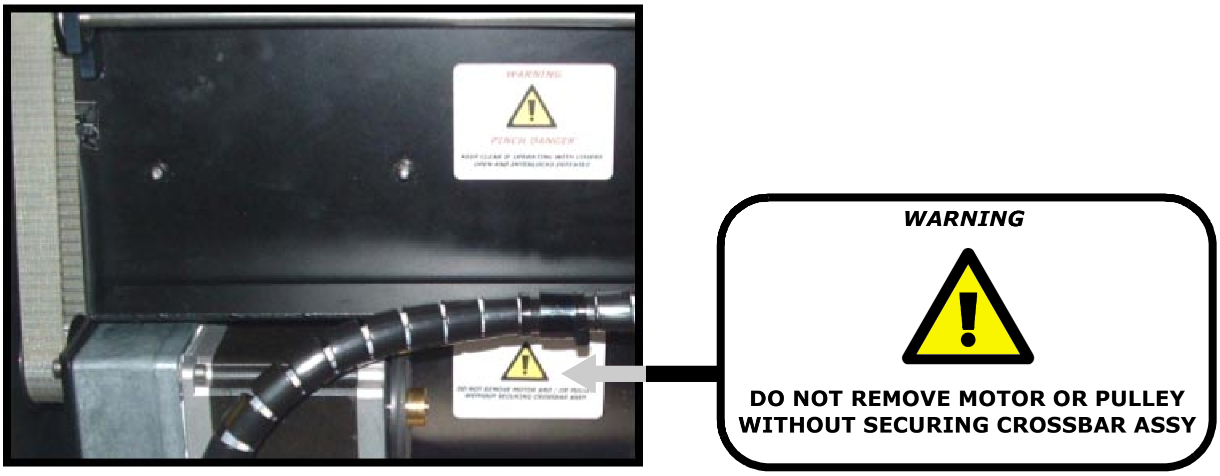
Fuse 110V – T10AH110V

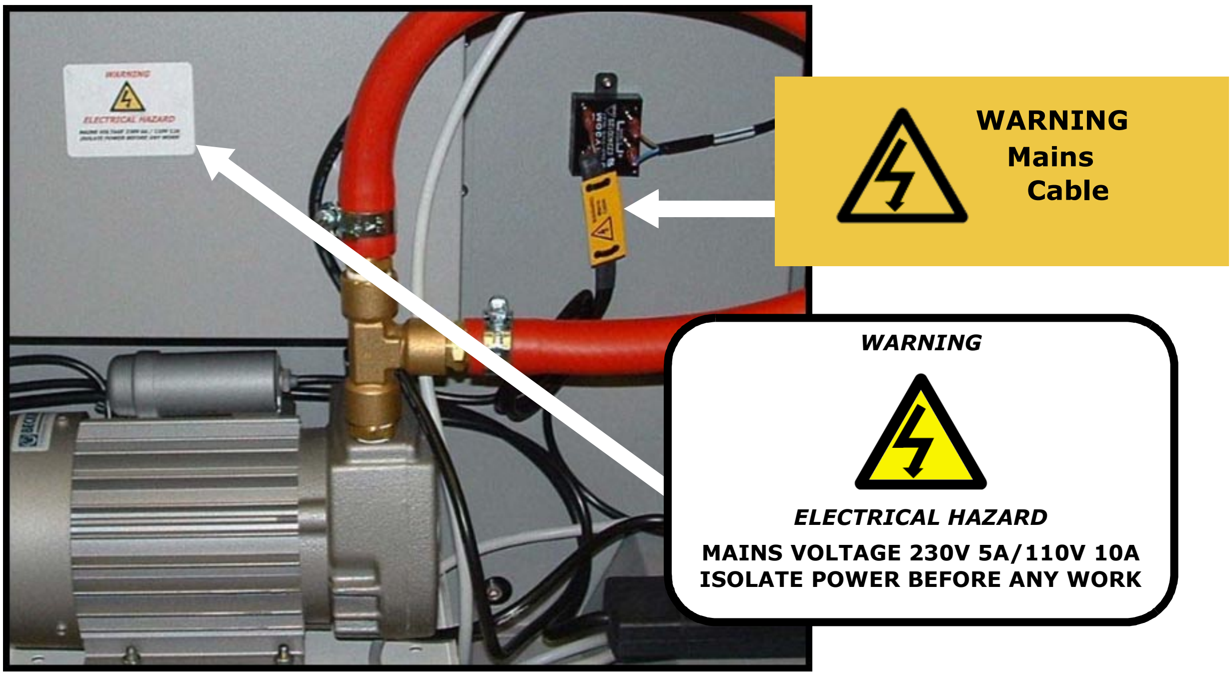
C.7 MET label

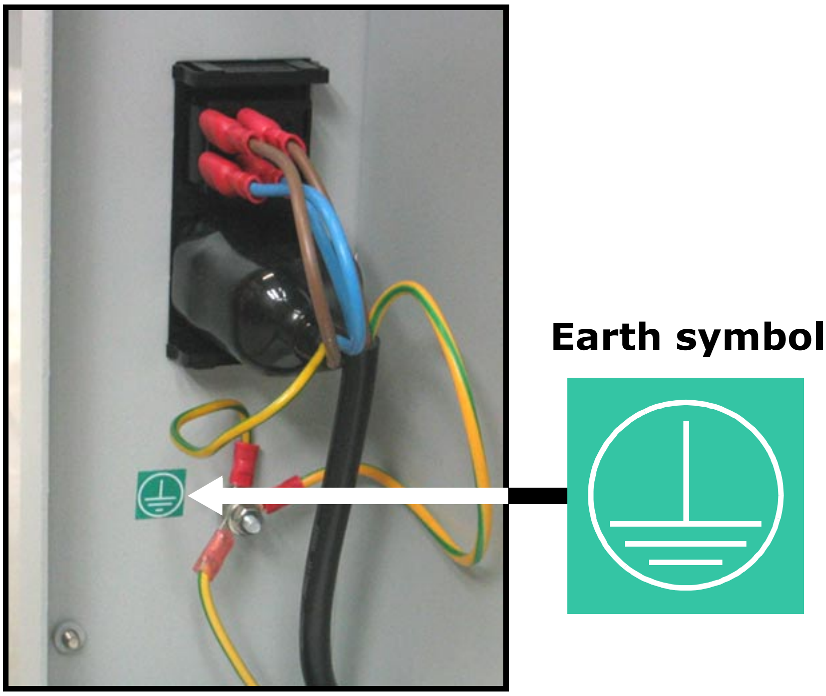
This label is present on 110V machines:

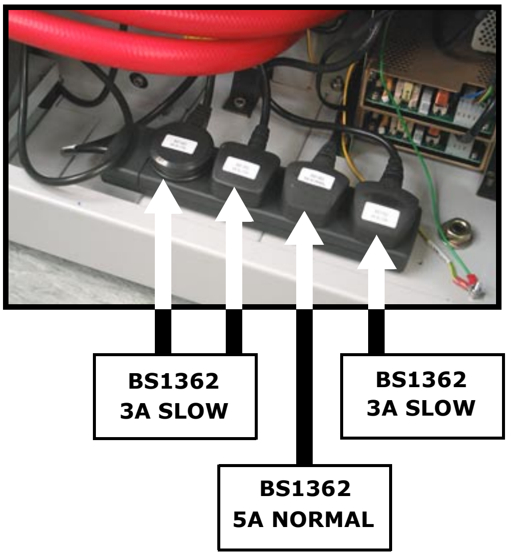


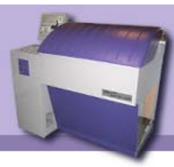
C.8 Other warning labels






Earth symbol


BS1362
3A SLOW
BS1362
3A SLOW
BS1362
5A NORMAL



Appendix D: Plate test results

Use the following pages to record the results of the plate tests carried out in chapter 10. Where appropriate, follow any advice given for test failures.

This chapter contains the following test results pages:

- Test 1A – Laser Power (p118).
- Test 2A – Carriage Reference Setup (p119).
- Test 2B – Fast Axis Scaling (p120).
- Test 2C – Drum Reference (p121).
- Test 2D – Plate Registration Skew (p122).
- Test 2E – Focus and Spot Shape (p123).
- Test 2F – Beam Bow (p124).
- Test 3A – Plate Registration Repeatability (p125).
- Test 4A – Flat Tint (p126).

Note: Photocopy this chapter if more results pages are required for additional plates.

Test 1A – Laser Power

Test	1A. Laser Power (page 71)
Test Description	Determines the correct laser power for the plate
Uses plate	Plate 1: Calibration Strips
Test tolerance	± 10% on factory setting
If this test fails	Immediately call a support engineer for advice

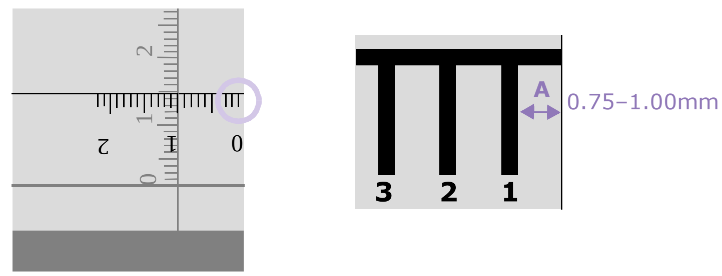
Test results

Plate name	Size (mm)	Thickness (mm)	+/– working	Factory set power	Power	Pass/Fail
Example plate	745 x 615 mm	0.3mm	+	30%	34.00%	Pass

Engineer's comments

Date	Comments

Test 2A – Carriage Reference Setup

Test	2A. Carriage Reference Setup (page 77)
Test Description	Check the image start position in the slow scan direction 
Uses plate	Plate 2: Scaling
Test tolerance	+0–0.25mm (a result between 0.75 and 1.00mm is a test pass)
If this test fails	Call a service engineer when you have finished all tests

Test results

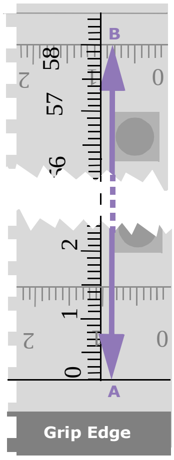
	Example	Plate 1	Plate 2	Plate 3	Plate 4	Plate 5	Plate 6
Plate name/size	745x615						
Measurement A	0.8mm						
Pass/Fail	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline

Engineer's comments

Date	Comments

Test 2B – Fast Axis Scaling

Test	2B. Fast Axis Scaling (page 79)
Test Description	Checks the accuracy of the image height



Uses plate	Plate 2: Scaling
Test tolerance	±0.1mm
If this test fails	Call a service engineer when you have finished all tests

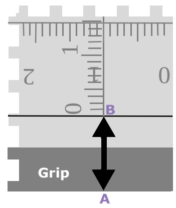
Test Results

	Example	Plate 1	Plate 2	Plate 3	Plate 4	Plate 5	Plate 6
Plate name/size	745x615						
Measurement on plate	58cm						
Actual measurement	58.2cm						
Pass/Fail	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline			

Engineer’s comments

Date	Comments

Test 2C – Drum Reference

Test	2C. Drum Reference (page 80)
Test Description	Checks the ‘length of plate in clamp’ value 
Uses plate	Plate 2: Scaling
Test tolerance	±0.1mm
If this test fails	Call a service engineer when you have finished all tests

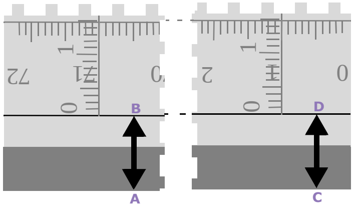
Test Results

	Example	Plate 1	Plate 2	Plate 3	Plate 4	Plate 5	Plate 6
Plate name/size	745x615						
Drum reference	15mm						
Actual measurement AB	15.05cm						
Pass/Fail	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline			

Engineer’s comments

Date	Comments

Test 2D – Plate Registration Skew

Test	2D. Plate Registration Skew (page 82)
Test Description	Checks that the plate is sitting squarely in the clamp 
Uses plate	Plate 2: Scaling
Test tolerance	±0.1mm
If this test fails	Call a service engineer when you have finished all tests

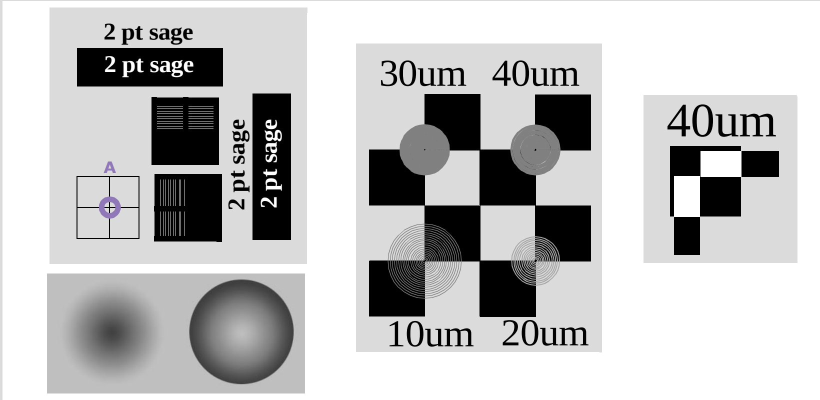
Test Results

	Example	Plate 1	Plate 2	Plate 3	Plate 4	Plate 5	Plate 6
Plate name/size	745x615						
Measurement AB	20.0mm						
Measurement CD	20.1+mm						
Difference in measurement	0.1+						
Pass/Fail	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline			

Engineer’s comments

Date	Comments

Test 2E – Focus and Spot Shape

Test	2E. Focus and Spot Shape (page 83)
Test Description	Checks the laser focus 
Uses plate	Plate 2: Scaling
Test tolerance	Check for sharp lines and text, a visible step at the 50% point in vignettes, and lines of equal thickness in the concentric circles and 40um lines
If this test fails	Call a service engineer when you have finished all tests

Test Results

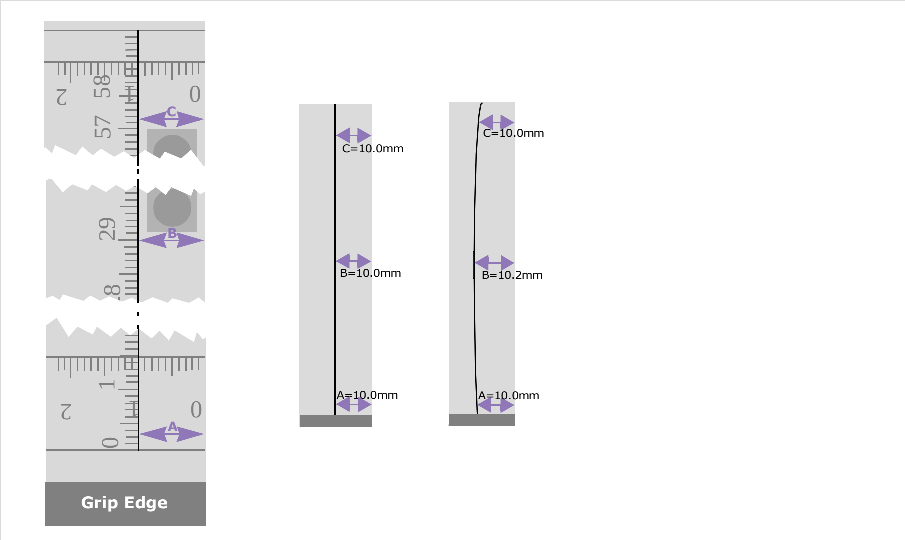
	Example	Plate 1	Plate 2	Plate 3	Plate 4	Plate 5	Plate 6
Plate name/size	745x615						
Laser Focus	Good OK Poor	Good OK Poor	Good OK Poor	Good OK Poor	Good OK Poor	Good OK Poor	Good OK Poor
Pass/Fail	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline

Engineer's comments

Date	Comments

Test 2F – Beam Bow

Test	2F. Beam Bow (page 85)
Test Description	Checks that the laser scans perpendicular to the drum's surface



Uses plate	Plate 2: Scaling
Test tolerance	±0.1mm (take into account non-square plates)
If this test fails	Call a service engineer when you have finished all tests

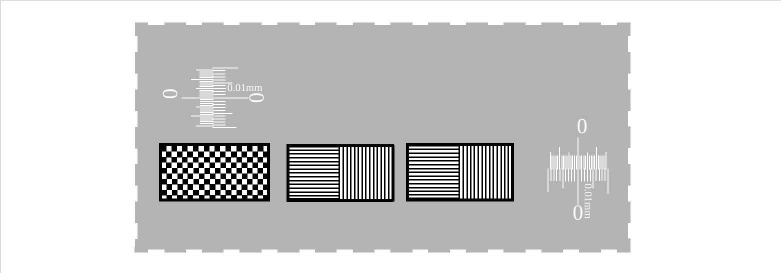
Test Results

	Example	Plate 1	Plate 2	Plate 3	Plate 4	Plate 5	Plate 6
Plate name/size	745x615						
Difference in measurement ABC	0.2						
Pass/Fail	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline			

Engineer's comments

Date	Comments

Test 3A – Plate Registration Repeatability

Test	3A. Plate Registration Repeatability (page 89)
Test Description	Checks the accuracy and repeatability of the plate registration 
Uses plate	Plate 3: Scaling With Slur
Test tolerance	In 50% boxes: check 'severity' of the word slur On Vernier scale: ± 0.05mm (50µm) in each axis
If this test fails	Call a service engineer when you have finished all tests

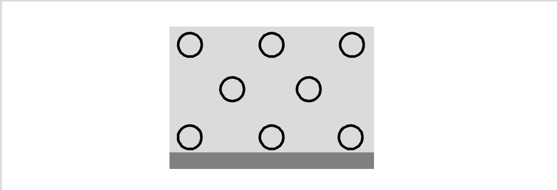
Test Results

	Example	Plate 1	Plate 2	Plate 3	Plate 4	Plate 5	Plate 6
Plate name/size	745x615						
Slur severity	Mild <u>Moderate</u> Severe	Minor Moderate Severe	Minor Moderate Severe	Minor Moderate Severe	Minor Moderate Severe	Minor Moderate Severe	Minor Moderate Severe
Vernier measurement	0.05+mm						
Pass/Fail	Pass Fail <u>Borderline</u>	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline

Engineer's comments

Date	Comments

Test 4A – Flat Tint

Test	4A. Flat Tint (page 94)
Test Description	Checks for variation in the flat tint across the plate 
Uses plate	Plate 4: Full50
Test tolerance	• No banding across plate • Continuity: All measurements taken are within 2% of each other • Laser power: Measurements are 52% for silver plates (or other measurement obtained from dealer)
If this test fails	Call a service engineer when you have finished all tests

Test Results

	Example	Plate 1	Plate 2	Plate 3	Plate 4	Plate 5	Plate 6
Plate name/size	745x615						
Banding	No Yes	No Yes	No Yes	No Yes	No Yes	No Yes	No Yes
All measurements within 2% of each other?	No Yes	No Yes	No Yes	No Yes	No Yes	No Yes	No Yes
All measurements 52% (or other)	No Yes	No Yes	No Yes	No Yes	No Yes	No Yes	No Yes
Pass/Fail	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline	Pass Fail Borderline

Engineer's comments

Date	Comments