



**VIRMOS**

CNR - ISTITUTO DI FISICA COSMICA G. OCCHIALINI

REF : VLT-MAN-VIRG-14634-0001

INT ref.

Issue 3

Rév. 1

Date: 28/03/2003

page 1

# VIRMOS

## Mask Manufacturing Unit

### Operator Manual

*Authors :* **L.Chiappetti, G.Conti,  
D. Maccagni, E.Mattaini**

*Approved by :* **O. LeFèvre, D.Maccagni**



## EVOLUTION PAGE

*Note: text items changed between issue 3.0 and 3.1 are indicated in this font and/or colour in both the printed and HTML version, with a bar on the margin in the printed version and on coloured background (for the entire page) in the HTML version.*

Issue	Rev.	Paragr.	Date	Observations
0	0	all	19/10/1999	Internal issue continuously updated
1	0	all	29/03/2000	Draft issue for PAE
2	0	all	29/06/2000	Delivery of MMU to Paranal
2	1	2, 5, 5.6, 6.1.8.1, 7.1, 7.3, 7.3.1, 7.3.3, 7.4, 7.6, 7.8, 7.8.1, 7.8.2, 7.9.2, 8.1.2.11, 8.1.5.11, 8.1.5.28, 8.2, 9.2.3	15/01/2002	VIMOS commissioning 1
3	0	7.3, 7.3.1, 7.3.2, 7.3.3, 7.4, 7.8.1, 7.9.3, 8.1.3.2, 8.1.3.5, 8.1.3.6, 8.1.5.12, 8.2, 9.1.1, 9.1.2	12/04/2002	Second VIMOS commissioning
3	1	6.1.1, 6.1.2, 6.1.3, 6.1.4, 6.1.5, 6.1.6, 6.1.8.1, 6.1.10, 6.1.10.2, 6.1.10.7, 6.1.10.8, 6.2, 7.3.2, 7.4, 8.1.1.2, 8.1.3.6, 8.1.3.8, 8.2, 8.3.2, 8.3.3, 8.3.4, 8.3.5, 8.3.6, 8.3.9	28/03/2003	Delivery of MHS v 6 and Cut Manager v2 and ...
		1, 3, 5.5.2, 5.3, 5.4, 5.6, 6.1.7, 6.1.8.2, 6.1.9, 7.2, 7.3, 7.3.3, 7.10, 8.1.4.3		... removal of references to NIRMOS



## 1. Introduction

The present document (**Operator Manual**) is the first part of the **VIRMOS Mask Manufacturing Unit (MMU) operation and maintenance manual**. The MMU is the complex of hardware and software developed by the VIRMOS Consortium to make the multi-slit masks for the VIMOS *and NIRMOS* spectrographs. An executive summary of the MMU functions is presented in the MMU Global Description document [\[ref. 11\]](#)

This first part collects the information needed to the operator for routine operations, and namely, after some common formal parts (sections 2-4) : a description of the MMU system (in section 5), a description of the MMU software functions inclusive of their GUIs (in section 6), a step-by-step description of the standard operations (in section 7), a description of the contingency recovery procedures, of all possible error messages and of standard recovery actions (in section 8), and a description of the routine quality control and maintenance activities (in section 9).

A companion document (part 2 [Maintenance Manual](#)) will include technical information not necessary for routine operations, inclusive of full software description and extraordinary maintenance activities.

The present document will be delivered in hardcopy (and computer readable form if required) as well as a set of HTML files for easier on-line consultation.

## 2. Applicable and Reference Documents

[1] ESO, *Interface Control Document between the VIRMOS Control Software and the Observation Handling Subsystem*. VLT-ICD-ESO-14600-2030 issue 1.0 14 Feb 2000

[2] ESO, *Data Interface Control Document*. GEN-SPE-ESO-19400-794 issue 1.1 25 Nov 1997

[3] VIRMOS Consortium, *VIMOS Mask Conversion Software User and Maintenance Manual*. VLT-MAN-VIRG-14616-0134 issue 1.2 09 Sep 2001 (to be updated, latest version [available on line at IFCTR](#))

[4] Gerber System Corp. Customer Support Dept., *Gerber RS 274X Format Users' Guide*. Part no. 414-100-014 Rev B 10 Oct 1997

[5] Adobe Systems Inc., *Postscript Language Reference Manual*, second edition,. ISBN 0-201-18127-4

[6] LPKF Laser & Electronics GmbH, *LPKF CircuiCam StencilLaser version 3.0 Manual*, English Edition, 20 May 1998

[7] LPKF Laser & Electronics GmbH, *LPKF BoardMaster version 3.0 Manual*, English version, Sep 1997

[8] LPKF CAD/CAM Systems AG, *LPKF StencilLaser Manual typ 600x600 DC*, English version 2.0, 1998

[9] Taylor Hobson Ltd., *Operator's Handbook for FORM TALYSURF plus*. 112/2458, no date indicated

[10] ANTIL Srl, *Robot ARS1 Manuale d'uso e manutenzione*, Versione M.USO.ARS1, no date indicated, in italian, with 2 annexes

[11] VIRMOS Consortium, *VIRMOS Mask Manufacturing Unit Global Description*. VLT-LIS-VIRG-14634-00001 issue 1.4 22 Mar 2000

[12] VIRMOS Consortium, *VIRMOS Mask Manufacturing Unit Preliminary Acceptance Europe Test Results* issue 1.2 27 Apr 2000

[13] VIRMOS Consortium, *VIRMOS Mask Manufacturing Unit Optimization of the cutting parameters Slit edge quality reference measurements*. VLT-LIS-VIRG-14634-00005 issue 1.1 10 Apr 2000



## 3. Acronyms

ADM	Aperture Definition file in Millimeter
DLL	Dynamic Load Library
DRS	Data Reduction Software (on IWS)
EEPROM	Electrically Erasable Programmable Read Only Memory
ERD	Emergency Repair Disk
FAT	File Allocation Table (filesystem)
GUI	Graphical User Interface
IC	Instrument Cabinet
ICT	Instrument Cabinet Table
IWS	Instrument WorkStation
LCD	Liquid Crystal Display
LED	Light Emitting Diode
LRU	Line Replaceable Unit
MCS	Mask Conversion Software (on IWS : vmcs for VIMOS, <del>mmcs for NIRMOS</del> )
MDJ	Mask Discard Job
MDT	Mask Discard Termination report
MHCU	Mask Handling Control Unit
MHS	Mask Handling Software
MIJ	Mask Insertion Job
MIT	Mask Insertion Termination report
MMCU	Mask Manufacturing Control Unit
MMJ	Mask Manufacturing Job
MMT	Mask Manufacturing Termination report
MMU	Mask Manufacturing Unit
MSF	Machine Slit File
NIRMOS	Near InfraRed Multi Object Spectrograph
NTFS	NT File System
SC	Storage Cabinet
SCT	Storage Cabinet Table
VB	Visual Basic
VIMOS	Visible Multi Object Spectrograph

## 4. Stylistic conventions

The following notation conventions are used throughout this document :

- A *fixed spacing font* is used to identify computer related items like command names or run strings, or file names. In lack of an other explicit notation, this font indicates items to be supplied **in input** by the user and to be typed literally as they appear in this document.
- A *fixed spacing italic font* is used to identify variable command arguments, or variable parts of file names, which have to be replaced by user-supplied values.
- the same font indicates also eventual computer **output**.
- A(n Helvetica) font like e.g. **Control Panel => Display** indicates a cascade through Win NT menus, or panel/icons, or other system items.
- The same font in **SMALL CAPS** indicates boxes, buttons and other elements of our own GUIs.
- An underlined action like put serial port 2 to off indicates actions to be explicitly taken.
- **Bold** and *italic* are freely used to emphasize particular items.

The above description refers to the printed version of this document. For the HTML version use the look of the present page with your browser as reference.



## 5. System description

The items constituting the MMU configuration which will be delivered with the VIMOS instrument are the following.

It.	n.	Item	Manufactured	Procured	Notes
1		mask cutting machine		LPKF	
	a	LPKF Stencil Laser machine (model 600x600 DC)	LPKF	LPKF	
	b	electronics control rack SL 25-1 (includes water pump for laser cooling circuit)	LPKF	LPKF	
	c	cooling unit (chiller) MLA-S 7 N	KKT Kraus	LPKF	
	d	pressure duplicator Maximator	Schmidt, Kranz & Co.	LPKF	
	e	entrance filter complex (4 filters)	EWO	IFCTR	added
	f	dryer filter KM 10 (mounted with filter complex)	Zander	IFCTR	added
	g	additional springs and pressure regulator for clamping unit	---	IFCTR	added
	h	vacuum exhauster PP DC	TBH	LPKF	
	i	water piping	---	LPKF	
	j	air piping	---	LPFK	
	k	power and electric signal cabling	---	LPFK	
	l	Stencil Laser hardware documentation [ref. 8] with additional documentation for parts 1c, 1d, 1e, 1h and 2g.	LPKF	LPKF	
	m	standard spare part kit	LPKF	LPKF	
2		MMCU computer	Dell	IFCTR	
	a	tower rack w/ CPU, 3x2 GB disk , 2 serial, 1 parallel i/o ports, Ethernet port	Dell	Dell	
	b	17 inch colour monitor	Dell	Dell	
	c	SK100 REW keyboard	Dell	Dell	
	d	Microsoft mouse	Dell	Dell	
	e	Win NT Workstation 4.0 SP 3 operating system	Microsoft	Dell	
	f	system hardware and software documentation and installation media	Dell	Dell	
	g	Heidenhain serial card and cables connecting with LPKF control rack	Heidenhain	LPKF	installed
	h	additional card and serial cables connecting to LPKF control rack	LPKF	LPKF	installed
	i	twisted pair network cable	---	IFCTR	
	j	BoardMaster   StencilMaster 3.x software	LPKF	LPKF	
	k	BoardMaster documentation [ref. 7] and installation media	LPKF	LPKF	
	l	Cut Manager software	IFCTR	IFCTR	
3	a	VIMOS mask Storage Cabinets	IFCTR	IFCTR	
	b	bar code reader #1 (search)	Datalogic	IFCTR	
	c	bar code reader #2 (spare)	Datalogic	IFCTR	delivered Feb 02
	d	ceiling support for bar code cables	IFCTR	IFCTR	with cable 5i
4		Instrument Cabinet exchange unit (MHS robot)	ANTIL	IFCTR	
	a	Instrument Cabinet holder (ICs proper are not part of MMU and will be delivered by OAC)	IFCTR	IFCTR	
	b	robotized Z-displacement system	ANTIL	IFCTR	part of robot
	c	control electronics	ANTIL	IFCTR	part of robot
	d	mask stand (with mask clamp)	IFCTR	IFCTR	installed
	e	bar code reader #3 (on mask stand)	Datalogic	IFCTR	



It.	n.	Item	Manufactured	Procured	Notes
	f	internal power and electric signal cabling	ANTIL	IFCTR	
	g	bar code reader power supply unit (white box)	IFCTR	IFCTR	
	h	RS422/RS332 serial interface converter ICC 11	ANTIL	IFCTR	attached
	i	ANTIL documentation [ref. 10]. (in Italian)	ANTIL	IFCTR	
5		MHCU computer	Dell	IFCTR	
	a-f	standard items as per item 2a-f	vv	Dell	
	g	additional serial card with 2 ports	National Instruments	IFCTR	installed
	h	serial cable connecting ports 1 with stand bar code reader (includes power supply to reader)	IFCTR	IFCTR	
	i	serial cable connecting port 2 with SC robot bar code reader (includes power supply to reader)	IFCTR	IFCTR	mounted in 3d, spare available
	j	serial cable connecting port 5 with IC robot controller	IFCTR	IFCTR	via converter 4h
	k	twisted pair network cable	---	IFCTR	
	l	CircuitCam 3.x software	LPKF	LPKF	
	m	CircuitCam hardware key	LPKF	LPKF	to be plugged in parallel port
	n	CircuitCam documentation [ref. 6].and installation media	LPKF	LPKF	
	o	MHS software	IFCTR	IFCTR	
	p	disk mirroring software	IFCTR	IFCTR	also on MMCU
	q	pair of loudspeakers with audio cable	---	IFCTR	
6		TalySurf roughness meter	Taylor-Hobson	Taylor-Hobson	
	a	accessory kit	Taylor-Hobson	Taylor-Hobson	
	b	hardware and software documentation [ref. 9].	Taylor-Hobson	Taylor-Hobson	
7		spare computer	Dell	IFCTR	
	a-f	standard items as per item 2a-f	vv	Dell	
	g	additional serial card with 2 ports	National Instruments	IFCTR	installed
	h	serial cable for connection with roughness meter	IFCTR	IFCTR	
	i	FTS+ roughness meter communication software	Taylor-Hobson	Taylor-Hobson	
	j	Mountain roughness meter analysis software	Taylor-Hobson	Taylor-Hobson	
	k	Taylor-Hobson s/w hardware key	Taylor-Hobson	Taylor-Hobson	to be plugged in parallel port
	l	Visual Basic 6.0 development environment	Microsoft	IFCTR	
	m	Visual Basic documentation and installation media	Microsoft	IFCTR	
	n	bar code reader configuration software	Datalogic	IFCTR	

Consumables necessary for mask production (invar sheets) and for Stencil Laser operation (deionized water, additives etc.) are not listed here, but in the maintenance section [9.2](#)

Robot spare parts brought to Paranal during G.Conti's visit in July 2001 are listed in [Maintenance Manual section 12.3](#)

## 5.1 LPKF mask cutting machine

The StencilLaser cutting machine main body will be delivered as a mounted unit with the assistance of LPKF personnel. It will be connected with the remaining sub-units as described below.

It shall be noted that the default configuration of the cutting machine has been altered at IFCTR to reduce the force applied by the clamping frame to the Invar sheet when stretching (in order to minimize geometric distortions between the production of the masks under stretching and their usage when released). A set of springs has been applied instead of the air pressure system originally provided, in addition a pressure regulator has been added if one wishes to operate the



original system at reduced pressure. Another alteration introduced at IFCTR has been the move of the exhaustor control switches close to the laser head for operator convenience.

A protective tube will host a beam of cables and water pipes (primary laser cooling system) and connect the control rack to the cutting machine.

The water pipes for the secondary laser cooling system coming from the control rack will be connected to the chiller. The pipes are labelled with arrows (in => and out <=) on the chiller side, and must be connected with the inlets labelled `Eintritt` and `Austritt` on the chiller.

One tube will connect the StencilLaser outlet with the exhaustor, and allow removal of fumes and cutting residuals. Two electric cables (one for the exhaustor switch, and the other for the safety lamp which turn on when the filter needs cleaning) will also connect the StencilLaser machine with the exhaustor (which can therefore be commanded from near the laser head).

The compressed air flow is as follows : the pipe coming from the main supply enters the filter complex and passes through the first stage (2 filters), then goes into the dryer. From the dryer it goes to the pressure duplicator, and from here comes back into the second filtering stage (2 filters). Finally the pipe forks and enters the StencilLaser separately for the air pads and for the laser head blowing.

The dryer needs to be powered. It receives power from a supply common with the exhaustor, whose switch has a lamp indicating when it is on.

The data communication flow is as follows : the control rack will be connected by four cables to the ports in the backplane of the MMCU computer (see figure in [Maintenance Manual section 7.13](#)):

- one cable (labelled `PC SL25-1`) will be connected to the COM2 serial port (on left hand side of the backplane )
- two cables (labelled `PC X-1` and `PC X-2`) will be connected to the ports (respectively top and bottom) in the two port LPKF-Heidenhain card (on the rightmost side of the backplane)
- one cable (labelled `PC/MBUS`) will be connected to the 25-pin port in the other LPKF card (on the right hand side of the backplane)

## 5.2 Storage Cabinets

The VIMOS SCs will be delivered installed on a trolley (table mounted on wheels), and enclosed within a plexiglas protection. Each SC consists of 100 slots also made of plexiglas. Two quadrants are accessed on one side and the other two of the other side of the trolley. The four quadrant SCs are identified by a colour coding (which is used by the software to refer to the various quarants).

Quadrant	VIMOS colour
1	Blue
2	Red
3	Yellow
4	Green

One barcode reader will be installed on a sliding support which moves on either side of the table supporting the SCs. There will be two supports, one on the quadrant 1&2 side, and one on the quadrant 3&4 side. The barcode reader will be connected via appropriate cabling to a serial port of the MHCU computer. The cable will be suspended to a ceiling rotatable hanger, so that the reader can be toggled between the two supports without unplugging.

The identification stand used during storage is the mask stand, part of the MHS robot (see [5.3](#))



### 5.3 Instrument Cabinet exchange unit (MHS robot)

The VIRMOS IC exchange unit consists of an IC holder box (a trolley with transparent plexiglass walls and with drawers in which the ICs must be loaded) and a robotized front end (Z-displacement system) with associated control electronics.

The IC holder will be capable of holding the 4 VIMOS ~~or NIRMOS~~ ICs in its 4 drawers. The delivery and description of the ICs (part of the instrument) are outside the scope of the present document.

The Z-displacement system will move the IC holder up (once loaded onto the platform and secured in position by the three ball pins) and down in order to locate a given mask slot in front of the mask stand. A mask clamp will allow to insert or extract a mask in/from the ICs. A barcode reader will be installed on the mask stand. The robot and the bar code reader will have each one its own serial (and power) cable connection to the MHCU computer (the power supply for the barcode readers is hosted in a separate "white box" while the interface converter for the robot serial connection is applied externally to the main robot control electronics box and derives its power from it)

The Z-displacement system is equipped with a photo cell to detect obstructions in the moving path, and of a proximity switch used to make the reference (home) position. An emergency stop switch allows to manually interrupt the movement (*without* switching the machine off) in case of need.

### 5.4 MMU computer configuration

The MMU computers consist of 3 (three) identical PC-tower Optiplex GX1 computers by Dell ([www.dell.com](http://www.dell.com)) with factory installed Windows NT 4.0 SP3. Each computer consists of a CPU rack with 6 GB system disk, a monitor, keyboard, mouse, related cabling, s/w disks and manuals. We describe here the roles of the different computers while the steps taken to configure their h/w and s/w are described in the [Maintenance Manual](#)

- **MMCU** : Mask Manufacturing Control Unit, hosting two dedicated cards supplied by LPKF and connected to the LPKF stencil laser machine control rack. This machine runs IFCTR Cut Manager software and LPKF StencilMaster software.
- **MHCU** : Mask Handling Control Unit, hosting one National Instrument serial card, extending the number of I/O ports to 4, connected to the SC and IC bar code readers and to the IC robot. This machine runs IFCTR MHS software and LPKF CircuitCAM software. This machine also runs the ftp server accessed by the VIMOS ~~and NIRMOS~~ IWSs to submit jobs and retrieve reports.
- **Spare** : the spare computer is hardware configured as the MHCU. It is normally in cold redundancy, and is occasionally used to operate the roughness meter. The Microsoft Visual Basic development environment for the IFCTR software is also installed here, as well as the barcode reader configuration utility.

The IFCTR disk mirroring software runs on MMCU and MHCU to keep their data disk content synchronised.

### 5.5 Roughness meter

The Taylor Hobson Form Talysurf Plus roughness meter will be delivered installed on a granite stand with a special vice manufactured by IFCTR to hold the measuring samples. The standard accessory kit (inclusive of measuring knives and of the calibration sphere) will be delivered in the kit metal case.

The relevant software will be installed on the spare computer, to which a serial cable connection to port 1 must be done for data acquisition (the roughness meter operates also in standalone mode and does not need the connection if one does not want to use the software). An hardware key is necessary to use the software and must be plugged in the parallel port of the spare computer





## 5.6 MMU operation concepts

The entire MMU system allows to perform the following operations necessary for the execution of VIMOS ~~and NIRMOS~~ observations :

- To manufacture on the stencil laser machine the masks requested by the MCS via a Mask Manufacturing Job order (MMJ), and to store the manufactured masks in the Storage Cabinet (SC).
- To load the masks requested by the MCS via a Mask Insert Job order (MIJ) into the Instrument Cabinet (IC) which will then be mounted at the telescope
- To get rid of masks (and related files) no longer needed as requested by the MCS via a Mask Discard Job order (MDJ)
- To manage and recover contingency conditions which may occur during the above tasks.

All operations (except contingency recovery and maintenance operations managed autonomously by the MMU operator) are executed after receipt of a job order from MCS (in form of a file transferred from IWS to a staging area of the MHCU computer), and terminate with the return of a termination report to MCS (depositing it in the same staging area on the MHCU). MCS will also have access to the tables (SCT and ICT) describing the current content of the SC and IC. In all operations MMU will only know masks by their individual six-digit mask id (which is also engraved on the actual masks in form of a bar code), and will normally handle entire mask sets. A mask set is defined as a set of four or less masks (with the same last 5 digits in the id) used for the same observation in the active VIMOS ~~or NIRMOS~~ quadrants.

The placement of slits on a mask will be described by a MSF file produced by MCS. Usage of the stencil laser machine requires conversion of this file in other proprietary format, resulting in the following data flow (for an executive summary with flow diagrams refer to the MMU Global Description document [\[ref. 11\]](#)).

- MSF files are transmitted by MCS in the framework of a MMJ.
- The MSF file is converted in a CAD format (Gerber format) by the MHS software running on MHCU. The Gerber file will also contain the description of the contour of the mask, the reference and handling holes and the bar code. Items in a Gerber files are grouped into *layers* (refer to [Maintenance Manual section 11.5.3](#) for more details).
- The Gerber file is converted in a proprietary format (LMD file) by the LPKF CircuitCam software. This will associate to each layer a different logical tool (laser power, frequency, speed), and associate it to a different working *phase*.
- The LMD files are used by the LPKF StencilMaster program running under our Cut Manager software on MMCU to manufacture the mask. Manufacturing is done in two phases : in the first (CuttingSide) one cuts all apertures, in the second (Outline) one cuts the contour allowing the mask to be removed from the machine. The actual association of physical parameters (laser power, frequency, speed) to logical tools is handled by StencilMaster.
- Masks are then stored (one mask set at a time) in the SC under assistance of the MHS software.
- Upon receipt of a MIJ and when the ICs are physically removed from the telescope and mounted on the IC robot controlled by MHCU, masks no longer needed are unloaded from IC and put back into SC ; then requested masks are loaded from SC into IC. All these functions are assisted by MHS on MHCU. The IC can then be returned to the telescope.
- Whenever a mask has been used for its observations, or is anyhow expired because of celestial constraints, MCS requests to discard it via an MDJ. This is processed by MHS on MHCU. As a result the masks are physically removed from SC, and all relevant LMD files deleted. This also means it is no longer possible to reproduce a copy of the mask (insofar the LMD file is kept on MMCU, it is possible to manufacture it again in case of damages).



## 6. Command reference

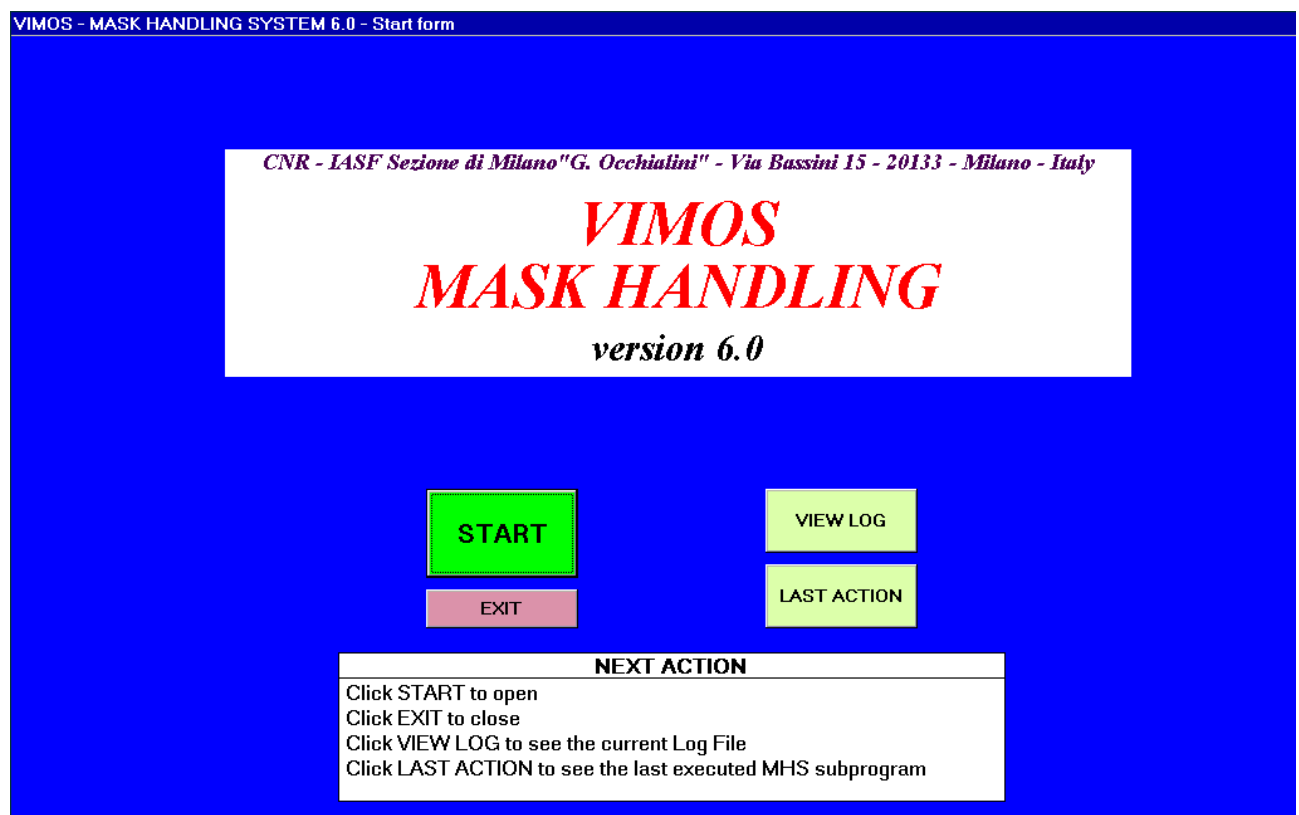
This section includes a description of the GUI (inclusive of screen dumps) and associated functions, and leads the operator through typical steps.

### 6.1 MHS

The Mask Handling Subsystem software runs on the computer designated as MHCUC.

#### 6.1.1 Initial page

The Mask Handling Subsystem is started clicking on the MHS icon (or using the start menu shortcut, or the following DOS-level command C:\Mhssw\mhs). As a result the following *start* form appears on the screen.



As usual the “NEXT ACTION” box contains a tip for the next action to be undertaken. The meaning of the various buttons and the default course of actions are as follows :

- the *program now deals with the VIMOS instrument only, therefore* click **START** to proceed to the next step (see section 6.1.5)
- clicking on **EXIT** will ask confirmation to exit MHS
- The **VIEW LOG** and **LAST ACTION** buttons give access to the relevant displays (these can be accessed also by the majority of the other windows) described respectively in sections 6.1.2 and 6.1.4

#### 6.1.2 View log

The **VIEW LOG** button (when available) will present a window like the following. This is an informative, readonly window. The only possible course of actions is to click **CLOSE** to return to the previous screen. A scrollbar allows to scroll through a long log file. The format of the log file is described in Maintenance Manual ([section 11.5.7.](#)).



**Mask Handling System Log**

VIMOS mhcu log file of the day: 2003-03-20 [CLOSE]

```

11:17:20>/UNFORESEEN: bar code not responding [mhcuV]
11:17:20>/RECOVERY: forcing program termination [mhcuV]
11:17:23>-START MHS STORE / mask store function started [mhcuV]
11:20:00>-START MHS CONVERT / file convert function started [mhcuV]
11:20:00>-READ MMJ / copied order 2001-01-22T23_43_35.000-014.mmj [mhcuV]
11:20:00>-READ MSF / copied 100004.msf file to d:\mhs\vimos\msf\ [mhcuV]
11:20:00>-READ MSF / copied 200004.msf file to d:\mhs\vimos\msf\ [mhcuV]
11:20:00>-READ MSF / copied 300004.msf file to d:\mhs\vimos\msf\ [mhcuV]
11:20:00>-READ MSF / copied 400004.msf file to d:\mhs\vimos\msf\ [mhcuV]
11:20:03>-WRITE GERBER / created file 1_100001.gbr in d:\mhs\vimos\gbr\ [mhcuV]
11:20:03>-WRITE GERBER / created file 1_100004.gbr in d:\mhs\vimos\gbr\ [mhcuV]
11:20:03>-WRITE GERBER / created file 2_200004.gbr in d:\mhs\vimos\gbr\ [mhcuV]
11:20:03>-WRITE GERBER / created file 3_300004.gbr in d:\mhs\vimos\gbr\ [mhcuV]
11:20:03>-WRITE GERBER / created file 4_400004.gbr in d:\mhs\vimos\gbr\ [mhcuV]
11:20:52>-PAUSE MHS CONVERT / mask convert function suspended by operator [mhcuV]
11:20:52>-RESUME MHS CONVERT / convert resuming from suspend or from crash [mhcuV]
11:21:05>-WRITE LMD / file 1_100004.lmd stored in d:\mhs\vimos\lmd\ [mhcuV]
11:22:19>-WRITE LMD / file 2_200004.lmd stored in d:\mhs\vimos\lmd\ [mhcuV]
11:22:24>-WRITE LMD / file 3_300004.lmd stored in d:\mhs\vimos\lmd\ [mhcuV]
11:22:37>-WRITE LMD / file 4_400004.lmd stored in d:\mhs\vimos\lmd\ [mhcuV]
11:22:44>-WRITE LMD / file 1_100004.lmd copied to MMCU [mhcuV]
11:22:44>-WRITE LMD / file 2_200004.lmd copied to MMCU [mhcuV]
11:22:44>-WRITE LMD / file 3_300004.lmd copied to MMCU [mhcuV]
11:22:44>-WRITE LMD / file 4_400004.lmd copied to MMCU [mhcuV]
11:22:44>-STOP MHS CONVERT / file convert function terminated [mhcuV]

```

### 6.1.3 View orders

The **VIEW ORDERS** button (when available) will present a window like the following. This is an informative, readonly window. The only possible course of actions (as instructed in the "NEXT ACTION" box) is to click **CLOSE** to return to the previous screen.

**LIST OF ORDERS**

VIMOS [CLOSE]

MASK MANUFACTURE ORDERS	MASK DISCARD ORDERS	MASK INSERT ORDERS
2001-01-22T23_43_35.000-010 2001-01-22T23_43_35.000-011 2001-01-22T23_43_35.000-012 2001-01-22T23_43_35.000-013	2000-02-05T23_43_35.000-036 2000-02-05T23_43_35.000-037	2000-02-05T23_43_35.000-020 null

**NEXT ACTION**  
List of received orders to be executed. CLICK CLOSE.

The window indicates the instrument *name*, and the job files submitted by the MCS and present in the staging area, divided in MMJ (corresponding to Mask Manufacturing Orders), MDJ (corresponding to Mask Discard Orders) and MIJ (Mask Insertion Orders). The names of the files correspond to the ESO ICD naming convention (ref. [1]) i.e. ISO date-and-time and sequence number, except that the hh:mm:ss string is expressed in the form hh\_mm\_ss as required by the Windows NT operating system.



### 6.1.4 Last action

The **LAST ACTION** button (when available) will present a window like the following. This is an informative, readonly window. The only possible course of actions is to click **CLOSE** to return to the previous screen.

MHS - LAST PROGRAM RUN									
<b>LAST MHS PROGRAM RUN</b>									
instrument:	VIMOS								
sub-program:	CONVERT								
on order:	2001-01-22T23_43_35.000-014.mmj								
	resumed on: 2003-03-20 11:20:52								
	stopped on: 2003-03-20 11:22:44								
<table border="1"> <thead> <tr> <th colspan="2">suspended orders to be completed</th> </tr> </thead> <tbody> <tr> <td>manufacture</td> <td>2001-01-22T23_43_35.000-014.mmj</td> </tr> <tr> <td>insert</td> <td></td> </tr> <tr> <td>discard</td> <td></td> </tr> </tbody> </table>		suspended orders to be completed		manufacture	2001-01-22T23_43_35.000-014.mmj	insert		discard	
suspended orders to be completed									
manufacture	2001-01-22T23_43_35.000-014.mmj								
insert									
discard									
<div style="border: 1px solid black; padding: 5px; display: inline-block;">CLOSE</div>									

The top form reports the instrument *name*, the name of the last program run (i.e. one of the main functions in the main task bar described in section 6.1.5), the relevant order (job file) if applicable, the time the function was started (or resumed) and the time the function was stopped (or suspended) as applicable.

The bottom *form* reports the “*suspended*” or “pending” orders (job files), where *this* means they are present in the MHS work area (they have been transferred from the Staging area and are no longer visible to MCS, appearing to it as “in progress”). *In particular MIJ and MDJ can only be suspended, while an MMJ can also be pending when the convert phase is terminated and the store phase is pending.*

### 6.1.5 Main task bar

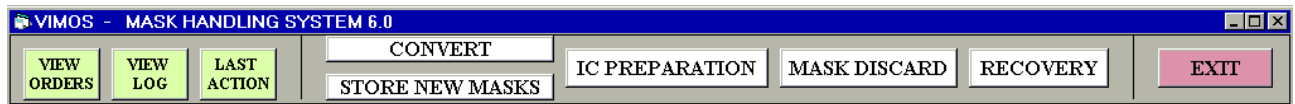
The **MAIN TASK BAR** is the basic interface to MHS functions *after the start form*. It presents the name of the instrument in the title bar, and a sequence of buttons, divided into three groups :

- Query buttons (always active unless the entire task bar is disabled )
  - **VIEW ORDERS** : gives access to the window described in section 6.1.3
  - **VIEW LOG** gives access to the MHS log as described in section 6.1.2
  - **LAST ACTION** gives access to the window described in section 6.1.4
- 
- Function buttons (only one or some of them can be active at any one time to lead the operator through the various phases)
  - **CONVERT** gives access to the file conversion stage (first stage) of the mask manufacturing cycle (see section 6.1.6)
  - **STORE NEW MASKS** gives access to the store function (last stage of the mask manufacturing cycle; see section 6.1.7)



- **IC PREPARATION** leads the operator through the two stages of the mask insertion cycle (see section 6.1.8)
- **MASK DISCARD** gives access to the discard function (mask discard cycle, see section 6.1.9)
- **RECOVERY** gives access to a front end page switching among the various recovery functions (see section 6.1.10)
- 
- The **EXIT** button

The main task bar is always present in the top part of the screen, while the rest of the screen is occupied by function-specific windows.



The following colour coding convention is in use for buttons :

- Query buttons, already active (unless the entire task bar is disabled, which occurs during critical intervals like robot movements and barcode reader activation), have an yellow background
- They become green as soon as activated, and this button remains so while active, and until a new function is activated (to remind the operator of the last function operated)
- The **EXIT** button has a red background

## 6.1.6 Convert

The *convert form* appears in the lower part of the screen as soon as the **CONVERT** button is pressed. It is divided into three rows, a button row, a "**NEXT ACTION**" hint box and a number of information boxes. The form shown here has the look typical of the *end* of the conversion. If a store session is already active on a MMJ order, the program won't start (a **POPUP** will ask for user acknowledgment and exit).

The button row contains the instrument *name*, and the following buttons :

- **SELECT AND RETRIEVE** button : this is active at beginning to call the order selection form, and transfer all associated files from the Staging area to the MHS work area. Once the transfer is completed the button is inactivated and the next one becomes active. If no MMJ orders exist, or if their transmission from MCS appears to be incomplete, a **POPUP** will ask to acknowledge program termination. If an MSF file is missing it is skipped after a **POPUP** has requested acknowledgment.
- **TRANSLATE** button : is used to start a loop converting all MSF files into Gerber format files. Once the conversion is completed the button is inactivated and the next one becomes active. Files which give conversion errors are skipped after a **POPUP** has requested acknowledgment.
- **SUBMIT** button : is used to start a loop (described below) which submits one Gerber file at a time to the CircuitCam program (described in 6.4) for generation of the LMD format files. If there are no Gerber files at all, a **POPUP** will ask to acknowledge program termination.
- There is also a yellow toggle button on top of the previous one, which alternates between **BATCH MODE** (default) and **INTERACTIVE MODE** of operation of CircuitCam.
- **END** button : this can be used at any time to suspend or terminate the convert function. A suspension occurs whenever a termination is requested and the three phases (retrieval, translation and LMD generation) have not been completed.



VIMOS - MASK HANDLING SYSTEM - [CONVERT FROM msf TO lmd FILES]

VIEW ORDERS | VIEW LOG | LAST ACTION | **CONVERT** | STORE NEW MASKS | IC PREPARATION | MASK DISCARD | RECOVERY | EXIT

VIMOS | SELECT the mask manufacturing order and RETRIEVE machine slit files | TRANSLATE machine slit files to Gerber files | BATCH MODE | SUBMIT Gerber files to CircuitCam | END

NEXT ACTION  
click END

MASK MANUFACTURING ORDER	machine slit files	Gerber files	LPKF cutting files	# of free slots in Storage Cabinets															
2000-01-10T23_43_35.000-220	8 files	8 files	8 files	SC1	SC2	SC3	SC4												
8 masks	8 files	8 files	8 files	0	0	0	0												
1_100220 2_200220 3_300220 4_400220 5_100221 6_200221 7_300221 8_400221	1_100220.msf 2_200220.msf 3_300220.msf 4_400220.msf 5_100221.msf 6_200221.msf 7_300221.msf 8_400221.msf	1_100220.gbr 2_200220.gbr 3_300220.gbr 4_400220.gbr 5_100221.gbr 6_200221.gbr 7_300221.gbr 8_400221.gbr	1_100220.LMD 2_200220.LMD 3_300220.LMD 4_400220.LMD 5_100221.LMD 6_200221.LMD 7_300221.LMD 8_400221.LMD	<table border="1"> <tr> <th colspan="4">masks to be manufactured</th> </tr> <tr> <th>quadr 1</th> <th>quadr 2</th> <th>quadr 3</th> <th>quadr 4</th> </tr> <tr> <td>2</td> <td>2</td> <td>2</td> <td>2</td> </tr> </table>				masks to be manufactured				quadr 1	quadr 2	quadr 3	quadr 4	2	2	2	2
masks to be manufactured																			
quadr 1	quadr 2	quadr 3	quadr 4																
2	2	2	2																

CRASH

The information boxes are arranged in columns from left to right as follows :

- *Mask Manufacturing Order* : one box contains the name of the current MMJ file (when loaded), another one the number of masks in the order, and a list box the mask identifier of all masks.
- *Machine slit files* : one box with the number of MSF files actually retrieved, and a list box with their names.
- *Gerber files* : one box with the number of Gerber files successfully converted and a list box with their names.
- *LPKF cutting files* : one box with the number of LMD files successfully generated, and a list box with their names.
- *# of free slots* : 4 boxes with the number of free slots (out of 100 available) in each quadrant of the SC to allow to check there is sufficient space for the new masks.
- *masks to be manufactured* : 4 boxes with the number of MSFs validly read broken down by quadrant

The **CRASH** button is for debugging purposes and will be removed.

The manufacturing *order selection form* is a typical example of order selection form. It is a modal form taking focus until dismissed. It presents the user with a list box of the MMJ files submitted by MCS in the Staging area. A “**NEXT ACTION**” hint box suggests the typical action which is to select the first order and click **OK**. One can extraordinarily select another order, or use the **CANCEL** button to acknowledge abort of the selection (which will cause termination of the convert program).



**MANUFACTURING ORDER SELECTION**

**VIMOS**

**MASK MANUFACTURING ORDERS**

2000-01-10T23_43_35.000-457
2000-01-10T23_44_12.000-458
2000-01-10T23_45_18.000-459
2000-01-10T23_46_14.000-460

**OK**

**CANCEL**

**NEXT ACTION**

Select the manufacturing order (default is the first one) and click OK

The LMD file generation via CircuitCam (in either batch or interactive mode) is assisted by a loop which uses the following popup menu form, which shows in two list boxes the preexisting Gerber files and the already created LMD files. One button (variously labelled **START**, **CONTINUE**, or **END**) allows to call CircuitCam on the next Gerber file or to terminate, while a **SUSPEND** button allows to suspend the entire convert function. As usual follow instructions in the "NEXT ACTION" hint box.

**CircuitCam - .lmd file generation**

**VIMOS**

Gerber files	LMD files
1_100004.gbr	1_100004.lmd
2_200004.gbr	
3_300004.gbr	
4_400004.gbr	

**CONTINUE**

**SUSPEND**

**NEXT ACTION**

Click CONTINUE for next .lmd file generation.

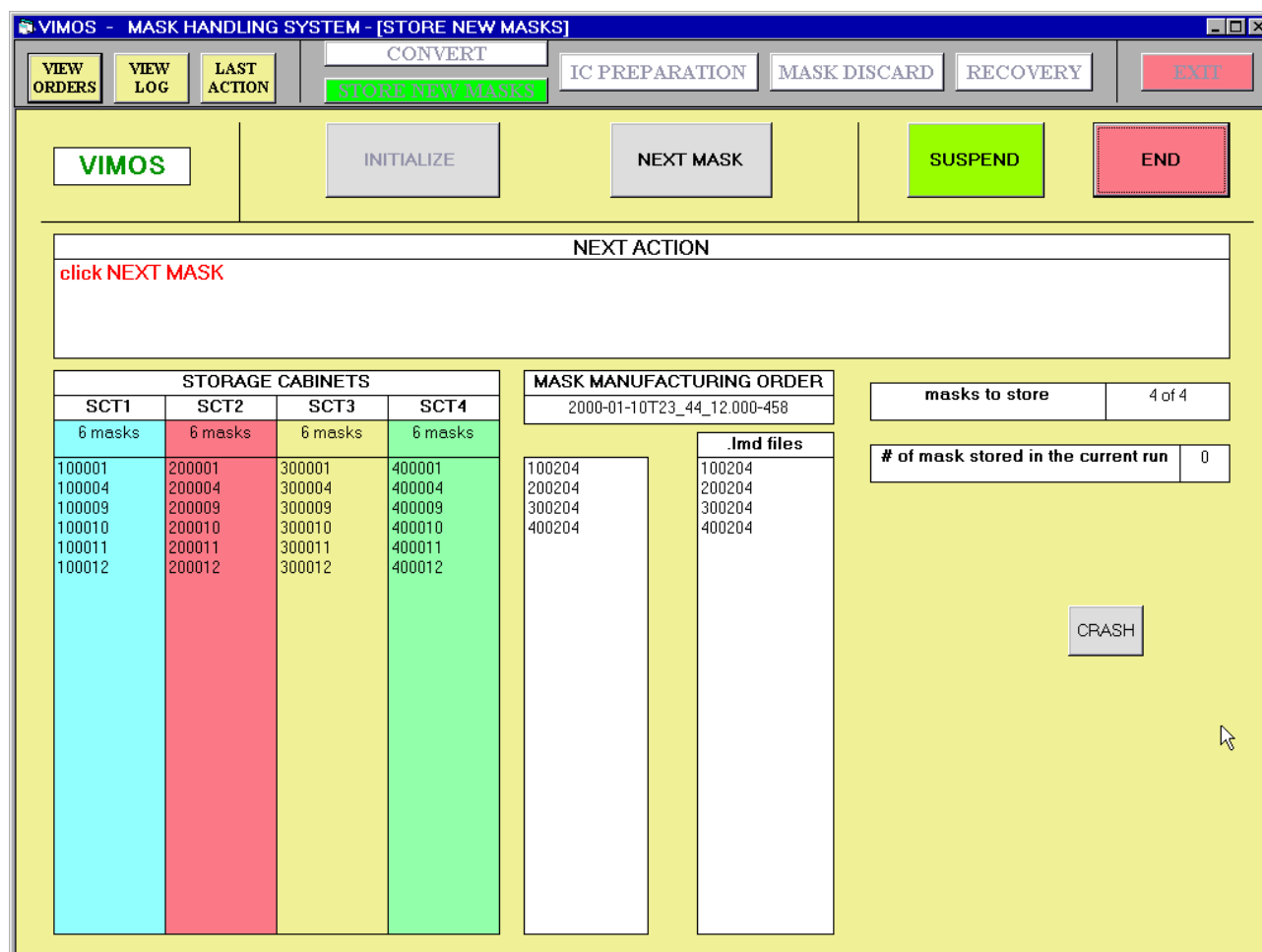


### 6.1.7 Store

The *store form* appears in the lower part of the screen as soon as the **STORE** button is pressed. It is divided into three rows, a button row, a “**NEXT ACTION**” hint box and a number of information boxes. The form shown here has the look typical *just after initialization*. This phase cannot start if *convert* has not been terminated (in this case a **POPUP** will ask acknowledgment and exit).

The button row contains the instrument *name*, and the following buttons :

- **INITIALIZE** button : this is active at beginning to set-up the connection with the SC barcode reader, and verify the presence of the files prepared by the previous *convert* stage. Absence of files to store will cause program termination after acknowledgment.
- **NEXT MASK** button : is used to start the procedure to store a (newly manufactured) mask in the SC. The 4 masks of a mask set are typically stored one immediately after the other.
- **SUSPEND** button : it shall be used to exit, after confirmation, from *store* in all cases the MHS has to perform more urgent functions, but the processing of the masks in the current MMJ is not yet finished..
- **END** button : this must be used only for the **definitive termination** of a MMJ, since it will delete the order and all associated files, and make available the MMT report to MCS. If not all masks in the current MMJ have been stored, the operator will be notified in the “**NEXT ACTION**” area and asked a double confirmation via two **POPUPS**. A single **POPUP** for confirmation occurs in the other cases.



The information boxes are arranged in columns from left to right as follows :

- *Storage Cabinet* : 4 colour coded list boxes with the identifiers of all masks in the SCT (and their number). Updated at any new mask storage.





- *Mask manufacturing order* one box with the name of the current MMJ file, a list box with the mask identifiers (which will be 'ticked' when stored), and another one with the relevant LMD files.
- *masks to store* : one box giving, in the form  $n$  of  $p$  the indication of how many ( $n$ ) valid LMDs are present on MMCU for manufacturing with respect to the total number of masks ( $p$ ) in the MMJ order.
- *# of masks stored* : one box with the total number of masks stored (inclusive of those stored before a crash or a suspension of the store function).

The **CRASH** button is for debugging purposes and will be removed.



## 6.1.8 IC preparation

The processing of a mask insertion order is a two stage process. First all masks **not** required to be inserted in the IC have to be *unloaded* into the SC. Then the masks not already present in the IC have to be *loaded* from the SC. The **IC PREPARATION** function will lead in a software assisted way through the two stages as appropriate. Typically one will go through the *unload* and *load* phases in order, but one may 'jump in' the procedure according to the content of MIJ and IC, and to the previous course of action.

### 6.1.8.1 Unload

The *IC unload form* appears in the lower part of the screen as soon as the **IC PREPARATION** button is pressed and there is no pending MIJ, or if *unload* is not terminated. It is divided into three rows, a button row, a "NEXT ACTION" hint box and a number of information boxes. The form shown here has the look typical *just after initialization*.

The screenshot shows the VIMOS MASK HANDLING SYSTEM interface. At the top, there is a title bar and a menu bar with buttons: VIEW ORDERS, VIEW LOG, LAST ACTION, CONVERT, STORE NEW MASKS, IC PREPARATION (highlighted), MASK DISCARD, RECOVERY, and EXIT. Below the menu bar, there are four main buttons: VIMOS, RETRIEVE ICTs, SCTs and MASK INSERTION ORDER, ICs UNLOADING, and END. A 'NEXT ACTION' box contains the instruction: 'click ICs UNLOADING to begin the UNLOAD procedure.' Below this, there are three data tables: MASK INSERTION ORDER, INSTRUMENT CABINET TABLES, and STORAGE CABINET TABLES. The MASK INSERTION ORDER table shows 4 masks: 100110, 200110, 300110, and 400110. The INSTRUMENT CABINET TABLES table has 15 rows and 4 columns (ICT1, ICT2, ICT3, ICT4). The STORAGE CABINET TABLES table has 6 rows and 4 columns (SCT1, SCT2, SCT4, SCT4). At the bottom, there are two status boxes: ROBOT STATUS (Robot ready to receive a command) and ROBOT CURRENT POSITION (UNDEFINED).

MASK INSERTION ORDER	
2000-01-10T23_43_35.000-400	
4 masks	
100110	
200110	
300110	
400110	

INSTRUMENT CABINET TABLES				
#	ICT1	ICT2	ICT3	ICT4
1	100211	200211	300211	400211
2	empty	empty	empty	empty
3	empty	empty	empty	empty
4	empty	empty	empty	empty
5	empty	empty	empty	empty
6	empty	empty	empty	empty
7	empty	empty	empty	empty
8	empty	empty	empty	empty
9	empty	empty	empty	empty
10	empty	empty	empty	empty
11	empty	empty	empty	empty
12	empty	empty	empty	empty
13	empty	empty	empty	empty
14	empty	empty	empty	empty
15	empty	empty	empty	empty

STORAGE CABINET TABLES			
SCT1	SCT2	SCT4	SCT4
100001	200001	300001	400001
100110	200110	300110	400110
100120	200120	300120	400120
100210	200210	300210	400210
100211	200211	300211	400211

masks to unload			
100211	200211	300211	400211

ROBOT STATUS	
Robot ready to receive a command	

ROBOT CURRENT POSITION	
UNDEFINED	

The button row contains the instrument *name*, and the following buttons :

- **RETRIEVE** button : active at beginning to call the order selection form, and get the MIJ from the Staging area to the MHS work area, it also sets-up the connection with the (SC and stand) barcode readers and with the MHS robot, and loads the SCT and ICT content. *During this phase it also prompts the operator to know if the ICs in use belong to the prime or spare set via a IC configuration form.* Then the button is inactivated and the next one becomes active.
- **IC UNLOADING** button : is used to start a loop which leads the operator through the unloading of masks from the IC and their storage back into the SC . Unloading occurs one quadrant at a time.
- **END** button : this can be used at any time to suspend or terminate the *unload* function. The function will be suspended if not all masks have been unloaded. In this case a resume of the IC preparation will offer the



opportunity to unload the missing masks. Termination may instead pass control to the load function (typically), or directly generate a MIT report for MCS and delete the MIJ if there are no further masks to load into ICs.

- The **CRASH** button is for debugging purposes and will be removed.

The information boxes are arranged in columns from left to right as follows :

- *Mask Insertion Order* : one box contains the name of the current MIJ file (when loaded), another one the number of masks in the order, and a list box the mask identifier of all masks. The identifiers are accompanied by a mark which can be ok if the mask is already in IC, <= if the mask has to be moved from SC to IC, and ?? if the mask is not recorded in the SCT or ICT tables (error).
- *Instrument Cabinet* : 4 top colour coded list boxes with the content of the 15 slots in each quadrant ICT (where a slot may be also empty), plus 4 (bottom) colour coded list boxes with the list of masks to be unloaded, built dynamically as a complement from the MIJ and ICT content
- *Storage Cabinet* : 4 colour coded list boxes with the identifiers of all masks in the SCT (and their number). Updated at any new mask storage.
- *Robot status and position*, 2 boxes containing the decoded value of the last status readout, and the last commanded Z-displacement unit position (one of undefined, home, load or quadrant  $q$  slot  $n$ ) *inclusive of a reminder of the cabinet set defined in the IC configuration form*

The insertion order selection form is similar in appearance and usage to the MIJ order selection form. Only it usually contains only one MIJ supplied from MCS (the presence of more than one requires a direct inquiry with the MCS operator to decide the correct priorities). In addition it **always** contains the null MIJ, which is used to fully unload the ICs from any mask they contain, whenever an empty IC is required for maintenance. The null MIJ cannot be deleted, but can be selected at any time, even if other orders are present, to perform autonomous maintenance .

**INSERTION ORDER SELECTION**

VIMOS

**MASK INSERTION ORDERS**

2000-01-10T23\_60\_10.000-800

2000-01-10T23\_60\_10.000-801

2000-01-10T23\_60\_10.000-802

2000-01-10T23\_60\_10.000-803

2000-01-10T23\_60\_10.000-804

2000-01-10T23\_60\_10.000-805

null

OK

CANCEL

**NEXT ACTION**

7 insertion orders are present. Only one can be executed.  
Verify what is to be done, select it and click OK

The IC configuration form shows up in the form reported below. It is compulsory for the operator to verify if the ICs in use for each quadrant belong to the original set (A) or the new spare set (B). If the last recorded configuration shown on the screen (in the example ABBA) matches the current one just press the NO CHANGE button. If instead it does not correspond click on MODIFY : this enables editing of the configuration via the round toggle buttons A and B (for each quadrant). When finished click the STORE button, which has replaced NO CHANGE, to proceed .



For verification the IC set one quadrant IC belongs to will appear in the robot current position box when the robot has reached a given IC slot, and will also be available via the View Table function (see 6.1.10.7).

**INSTRUMENT CABINET CONFIGURATION**

**VIMOS**

Current INSTRUMENT CABINET configuration

<b>Q1</b> <input checked="" type="radio"/> A (original) <input type="radio"/> B (spare)	<b>Q2</b> <input type="radio"/> A (original) <input checked="" type="radio"/> B (spare)	<b>Q3</b> <input type="radio"/> A (original) <input checked="" type="radio"/> B (spare)	<b>Q4</b> <input checked="" type="radio"/> A (original) <input type="radio"/> B (spare)
---	---	---	---

MODIFY

NO CHANGE

**NEXT ACTION**

Two sets of Instrument Cabinets are available: A (original) and B (spare)  
 The Instrument Cabinets currently in use are shown as selected  
 Click **MODIFY** to change or **NO CHANGE** to confirm the current configuration

### 6.1.8.2 Load

The IC load form appears in the lower part of the screen when the unload phase is terminated or directly (after acknowledgment) when the IC PREPARATION button is pressed and there are no masks to be unloaded. It is divided into three rows, a button row, a "NEXT ACTION" hint box and a number of information boxes. The form shown here has the look typical just after the automatic initialization (set-up of the connection with the SC and IC barcode readers and the MHS robot, and loading of the SCT and ICT content).

The button row contains the instrument name, and the following buttons :

- **NEXT MASK** button : is used to start the procedure to search a mask in the SC and load it into the IC. Loading occurs one quadrant at a time.
- **MASK FOUND** : it is normally inactive. It is activated as soon as the barcode reader locates the mask in the SC. The operator must press it to acknowledge this in order to continue.
- **MASK NOT FOUND** : it is normally inactive. It is activated as soon one presses **NEXT MASK**, and allows the operator to interrupt a search. It is disabled as soon as the barcode reader locates the mask in the SC.
- **END** button : this is typically used for the **definitive termination** of a MIJ, when it will delete all associated files, and make available the MIT report to MCS. If not all IC quadrants and slots have been processed (either successfully or not), the load function is instead suspended.

The information boxes are arranged in columns from left to right as follows :

- **Mask Insertion Order** : one box contains the name of the current MIJ file (when loaded), another one the number of masks in the order, and a list box the mask identifier of all masks. An x will mark masks already in IC after the unload phase. A + will mark any mask loaded in IC during this phase.



- *Storage Cabinet* : 4 colour coded list boxes with the identifiers of all masks in the SCT (and their number). Updated at any mask removal.
- *Instrument Cabinet* : 4 colour coded list boxes with the content of the 15 slots in each quadrant ICT, updated at each mask loading.
- *Robot status and position*, 2 boxes containing the decoded value of the last status readout, and the last commanded Z-displacement unit position (one of undefined, home, load or quadrant  $q$  slot  $n$ )

**VIMOS - MASK HANDLING SYSTEM - [INSTRUMENT CABINET LOAD]**

VIEW ORDERS | VIEW LOG | LAST ACTION | CONVERT | STORE NEW MASKS | **IC PREPARATION** | MASK DISCARD | RECOVERY | EXIT

VIMOS | NEXT MASK | MASK FOUND | MASK NOT FOUND | END

**NEXT ACTION**  
Click NEXT MASK

MASK INSERTION ORDER		STORAGE CABINET TABLES				INSTRUMENT CABINET TABLES				
2000-01-10T23_43_35.000-410		SCT1	SCT2	SCT3	SCT4	#	ICT1	ICT2	ICT3	ICT4
4 masks		100001	200001	300001	400001	1	empty	empty	empty	empty
100210		100110	200110	300110	400110	2	empty	empty	empty	empty
200210		100120	200120	300120	400120	3	empty	empty	empty	empty
300210		100210	200210	300210	400210	4	empty	empty	empty	empty
400210		100211	200211	300211	400211	5	empty	empty	empty	empty
						6	empty	empty	empty	empty
						7	empty	empty	empty	empty
						8	empty	empty	empty	empty
						9	empty	empty	empty	empty
						10	empty	empty	empty	empty
						11	empty	empty	empty	empty
						12	empty	empty	empty	empty
						13	empty	empty	empty	empty
						14	empty	empty	empty	empty
						15	empty	empty	empty	empty

CRASH

**ROBOT STATUS**  
Robot ready to receive a command

**ROBOT CURRENT POSITION**  
UNDEFINED

The **CRASH** button is for debugging purposes and will be removed.



## 6.1.9 Discard

The *discard form* appears in the lower part of the screen as soon as the **DISCARD** button is pressed. It is divided into three rows, a button row, a “**NEXT ACTION**” hint box and a number of information boxes. The form shown here has the look typical *in the middle of the procedure*.

MASK DISCARD ORDER		STORAGE CABINETS			
2000-01-10T23_50_07.000-500		SCT1	SCT2	SCT3	SCT4
8 masks		100004	200004	300004	400004
100002		100009	200009	300009	400009
100003 x		100010	200010	300010	400010
200002		100011	200011	300011	400011
200003 x		100012	200012	300012	400012
300002		100104	200104	300104	400104
300003		100105	200105	300105	400105
400002		100110	200110	300110	400110
400003		100204	200204	300204	400204

The button row contains the instrument *name*, and the following buttons :

- **RETRIEVE** button : this is active at beginning to call the discard order selection form (not shown, similar to the manufacturing order selection form), and get the MDJ from the Staging area to the MHS work area, it also sets-up the connection with the SC barcode reader, and loads the SCT content. Once all this is completed the button is inactivated and the next one becomes active.
- **SEARCH MASK** button : is used to start the procedure to search a mask in the SC for discard. Discarding occurs one quadrant at a time.
- **MASK FOUND** : it is normally inactive. It is activated as soon as the barcode reader locates the mask in the SC. The operator must press it to acknowledge this in order to continue.
- **ABORT SEARCH** : it is normally inactive. It is activated as soon one presses **SEARCH MASK**, and allows the operator to interrupt a search. It is disabled as soon as the barcode reader locates the mask in the SC.
- **END** button : this must be used only for the **definitive termination** of a MDJ, since it will delete all associated files, and make available the MDT report to MCS. If not all masks in the current MDJ have been processed, the operator will be asked twice for confirmation, otherwise only once. If this button is pressed before an MDJ has been retrieved the discard function is aborted.

The information boxes are arranged in columns from left to right as follows :

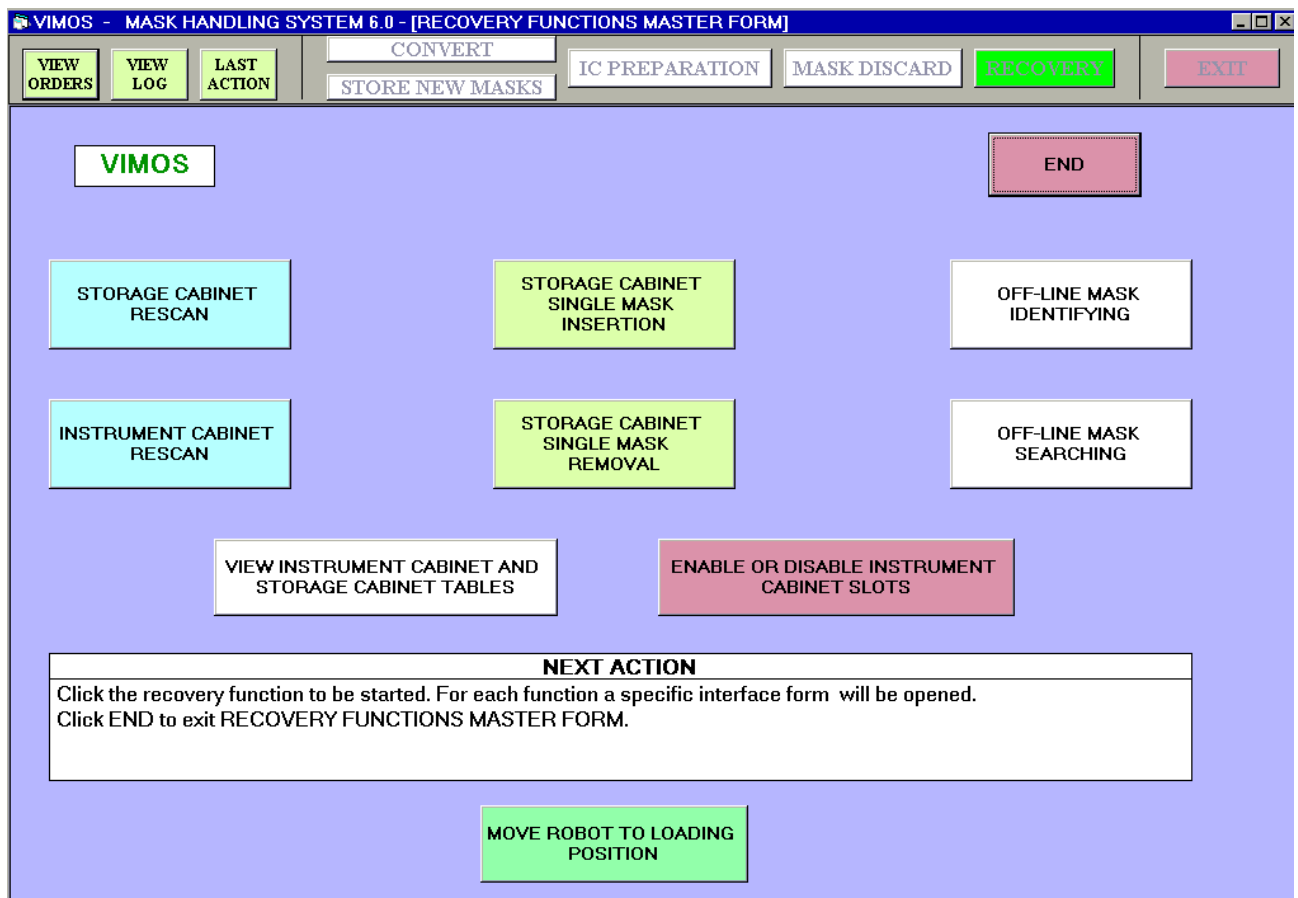


- *MaskDiscard Order* : one box contains the name of the current MDJ file (when loaded), another one the number of masks in the order, and a list box the mask identifier of all masks. An **x** will mark masks already discarded.
- *Storage Cabinet* : 4 colour coded list boxes with the identifiers of all masks in the SCT (and their number). Updated at any mask removal.

The **CRASH** button is for debugging purposes and will be removed.

## 6.1.10 Recovery

The *recovery function master form* appears in the lower part of the screen as soon as the **RECOVERY** button is pressed. It contains the “**NEXT ACTION**” hint box, the instrument *name*, an **END** button, and 6+2+1 function buttons :



- **STORAGE CABINET RESCAN** invokes the Rescan recovery function (see section 6.1.10.1)
- **INSTRUMENT CABINET RESCAN** invokes the RescanIC recovery function (see section 6.1.10.2)
- Storage Cabinet **SINGLE MASK INSERTION** invokes the Insert recovery function (see section 6.1.10.3), equivalent to a manual Store of a single mask
- Storage Cabinet **SINGLE MASK REMOVAL** invokes the Remove recovery function (see section 6.1.10.4), equivalent to a manual Discard of a single mask
- Off-line **MASK IDENTIFYING** invokes the Identify function (see section 6.1.10.5) used to read the bar code of a single mask
- Off-line **MASK SEARCHING** invokes the Search function (see section 6.1.10.6) allowing to locate a named mask in the SC
- *VIEW INSTRUMENT CABINET AND STORAGE CABINET TABLES* invokes the View Table function see section 6.1.10.7) allowing to verify the content of ICT and SCT without any further action
- *ENABLE OR DISABLE INSTRUMENT CABINET SLOT* invokes the Enable or Disable function see section 6.1.10.8) allowing to declare a given IC slot as out of use or to restore it to normal use



- An additional button **MOVE ROBOT TO LOADING POSITION** is used to unload the IC holder from the MHS robot (better, to move the robot platform to the unload position, allowing to remove the IC holder). This is normally done by the IC preparation or RescanIC functions, thus this function is provided only for contingencies.

**6.1.10.1 RescanSC**

The *Storage Cabinet rescan form* has the following look, and allows to reconstruct the content of the SCT based on the list of masks actually stored in the SC. Its appearance and usage should be self explanatory. The form shown here has the look *after* one quadrant has been processed and before the scan of the second is started. It contains **VIEW LOG** and **LAST ACTION** buttons since the main task bar is inactive during a modal form. It contains an **INITIALIZE** button to connect the SC barcode reader. The **SCAN START** button begins the rescan loop. An additional button **WRONG MASK FOUND** is normally invisible and appears only when it is necessary to search and remove a mask stored in the wrong quadrant. The **END** button returns to the recovery main form.

The screenshot shows the 'STORAGE CABINETS RESCAN' interface. At the top, there are buttons for 'VIMOS', 'VIEW LOG', 'LAST ACTION', 'INITIALIZE', 'SCAN START', and 'END'. Below these is a 'NEXT ACTION' field. The main area contains a table with columns for SC1, SC2, SC3, and SC4. Two rows show 'number of masks currently inserted in SC' and 'number of masks identified in right SC', both with a value of 5. A list of mask codes (100110, 100210, 100120, 100211, 100001) is displayed under SC1. A dialog box titled 'SCAN SC MESSAGE Sc02' is open, asking the user to count masks and enter a value between 0 and 100.

	SC1	SC2	SC3	SC4
number of masks currently inserted in SC	5 masks			
number of masks identified in right SC	5 masks			
Mask Codes	100110 100210 100120 100211 100001			

SC#	mask code

**SCAN SC MESSAGE Sc02**

Please count the masks currently inserted in the VIMOS Storage Cabinet 2 and type the number in the input box. Cancel will abort the program.

value between 0 and 100

OK Cancel

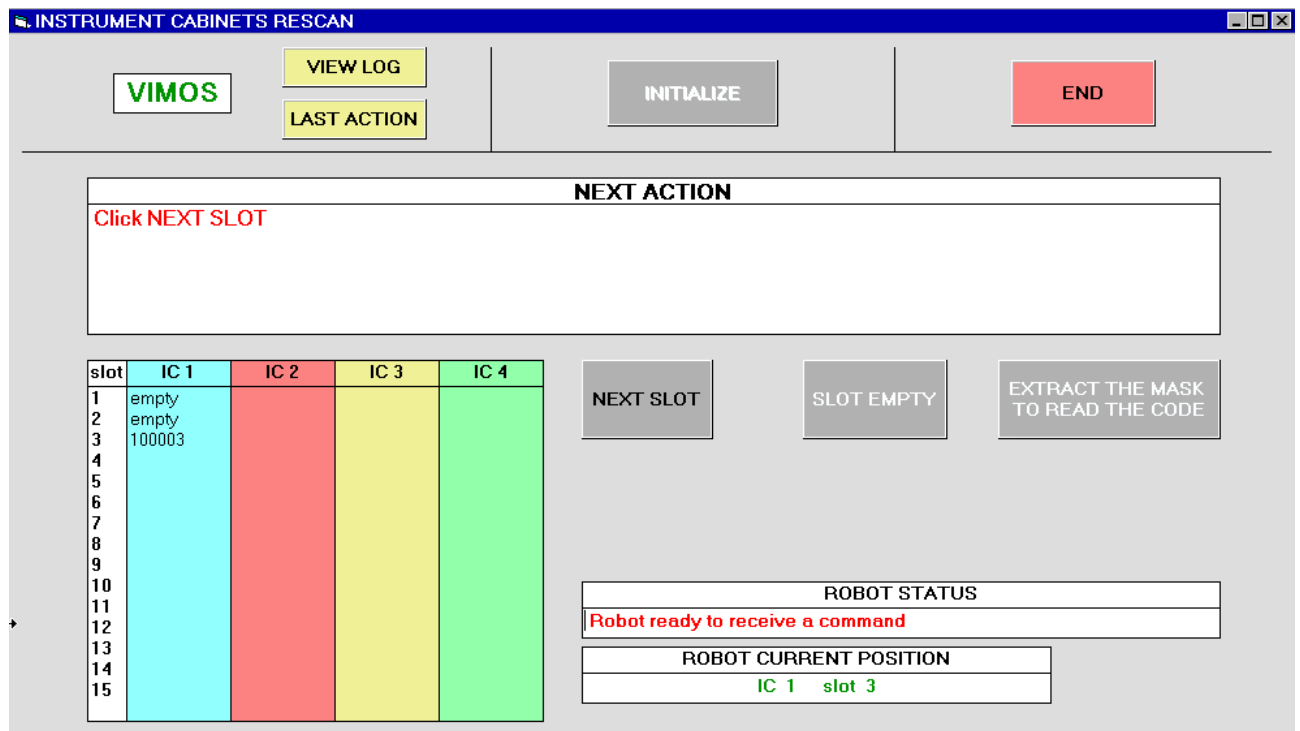




**6.1.10.2 RescanIC**

The *Instrument Cabinet rescan form* has the following look, and allows to reconstruct the content of the ICT based on the list of masks actually stored in the ICs. Its appearance and usage should be self explanatory. The form shown here has the look after a few slots of the first quadrant have been scanned. It contains **VIEW LOG** and **LAST ACTION** buttons since the main task bar is inactive during a modal form. It contains an **INITIALIZE** button to connect the stand barcode reader and the robot. The **FIRST|NEXT SLOT** button moves the robot to the next IC slot. Two additional buttons, **SLOT EMPTY** and **EXTRACT**, allow to tell whether the slot is empty or contains a mask (in which case its identifier is read by the barcode reader). The **END** button returns to the recovery main form.

The **INITIALIZE** button also prompts the operator to know if the ICs in use belong to the original or spare set via a *IC configuration form* identical to the one described in 6.1.8.1.





**6.1.10.3 Insert**

The *insert recovery modal form* has the following look, and allows to store manually one or more masks in the SC. Its appearance and usage should be self explanatory. It contains **VIEW LOG** and **LAST ACTION** buttons since the main task bar is inactive during a modal form. It contains an **INITIALIZE** button to connect the SC barcode reader and load the SCT content. The **INSERT** button starts a loop by which the operator can store one (or more) masks. The **END** button returns to the recovery main form.

STORAGE CABINET SINGLE MASK INSERTION

VIMOS

VIEW LOG

INITIALIZE

INSERT

END

LAST ACTION

NEXT ACTION

Click END

9 masks	9 masks	9 masks	10 masks
SCT1	SCT2	SCT3	SCT4
100204	200204	300204	400001
100104	200104	300104	400204
100004	200004	300004	400104
100105	200105	300105	400004
100110	200110	300110	400105
100009	200009	300009	400110
100010	200010	300010	400009
100011	200011	300011	400010
100012	200012	300012	400011
			400012



**6.1.10.4 Remove**

The *remove recovery modal form* has the following look, and allow to discard manually one or more masks in the SC. Its appearance and usage should be self explanatory. It contains **VIEW LOG** and **LAST ACTION** buttons since the main task bar is inactive during a modal form. It contains an **INITIALIZE** button to connect the SC barcode reader and load the SCT content. The **REMOVE** function starts a loop by which the operator can discard one (or more) masks. The **MASK FOUND** and **MASK NOT FOUND** buttons are used as in usual SC searches (see 6.1.9 or 6.1.8.2). The **END** button returns to the recovery main form.

STORAGE CABINET SINGLE MASK REMOVAL

VIMOS

VIEW LOG  
LAST ACTION

INITIALIZE

REMOVE

MASK FOUND

MASK NOT FOUND

END

NEXT ACTION

Click REMOVE

9 masks	9 masks	9 masks	10 masks
SCT1	SCT2	SCT3	SCT4
100204	200204	300204	400001
100104	200104	300104	400204
100004	200004	300004	400104
100105	200105	300105	400004
100110	200110	300110	400105
100009	200009	300009	400110
100010	200010	300010	400009
100011	200011	300011	400010
100012	200012	300012	400011
			400012

The identifier of the mask to be removed is passed by the user via a popup like this

SINGLE MASK REMOVAL

Type the bar code of the mask to remove and click OK  
Cancel or null entry will exit the program

OK

Cancel

400001



### 6.1.10.5 Search

The *search recovery modal form* has the following look, and allow to locate manually one or more masks in the SC for maintenance purpose. Its appearance and usage should be self explanatory. It contains an **INITIALIZE** button to connect the SC barcode reader. The **SEARCH** function starts a loop using the barcode reader. The **MASK FOUND** and **STOP SEARCH** buttons are used as in usual SC searches (see 6.1.9 or 6.1.8.2 or 6.1.10.4). The **END** button returns to the recovery main form.

**OFF-LINE SINGLE MASK SEARCH**

VIMOS INITIALIZE SEARCH MASK FOUND STOP SEARCH END

**NEXT ACTION**  
Click SEARCH to search another mask  
Click END to exit the program

MASK TO FIND  
100204

The mask 100204 has been found

The identifier of the mask to be searched is passed by the user via a popup similar to the one described in 6.1.10.4. Its name and the status of the search appear in two boxes in the lower part of the form.



### 6.1.10.6 Identify

The *identify recovery modal form* has the following look, and allow to read manually the barcode identifier of one or more masks inserting it in the MHS robot decoding stand (note that the Z-displacement unit is *not used*). Its appearance and usage should be self explanatory. It contains an **INITIALIZE** button to connect the mask stand barcode reader The **IDENTIFY** function puts the barcode reader on wait until a mask is inserted (the resulting identifier is shown in a box in the lower part of the form). The **END** button returns to the recovery main form.

OFF-LINE MASK IDENTIFICATION

VIMOS INITIALIZE IDENTIFY END

NEXT ACTION

OFF-LINE SINGLE MASK IDENTIFICATION

Do you want to identify another mask?

Yes No

MASK BAR CODE READING

300001

The form shown here shows also the popup which asks to continue the loop through more masks.



### 6.1.10.7 View Table

The View Table recovery modal form has the following look, and allows to inspect the current content of the ICT and SCT (as well as a reminder of the cabinet set to which IC belongs, as defined by the last invocation of the IC configuration form), without the need to invoke an order processing function or another recovery function. This is more a query function than a proper recovery action, but it has been felt appropriate to make it accessible from the Recovery form since it can be used easily to verify the result of other recovery actions (e.g. in the example picture one can verify which IC slots have been declared out of use).

The only possible course of actions (as instructed in the "NEXT ACTION" box) is to click **CLOSE** to return to the recovery main form.

INSTRUMENT CABINET AND STORAGE CABINET TABLES

VIMOS

CLOSE

INSTRUMENT CABINETS				
slot	Q1	Q2	Q3	Q4
	A(original)	B(spare)	B(spare)	A(original)
1	100101	empty	300101	400101
2	empty	200101	empty	empty
3	empty	empty	empty	empty
4	empty	empty	empty	empty
5	empty	empty	empty	empty
6	empty	empty	empty	empty
7	empty	empty	empty	empty
8	empty	empty	empty	empty
9	empty	empty	empty	empty
10	empty	empty	empty	empty
11	empty	empty	empty	empty
12	empty	empty	empty	empty
13	empty	empty	empty	empty
14	empty	empty	empty	empty
15	empty	empty	empty	empty

STORAGE CABINETS			
Q1	Q2	Q3	Q4
100001	200001	300001	400001

**NEXT ACTION**

Click **CLOSE**

### 6.1.10.8 Enable or Disable IC slots

The Enable or Disable recovery modal form has the following look, and allows to declare one or more IC slots as "out of use" or to restore them to normal usage. Out of use slots will be ignored during IC mask loading.

When one invokes the form one sees the current situation and can just click **CLOSE** to return to the recovery main form without any change.

Otherwise one can click on one of the two active buttons, namely **DISABLE** slots to toggle any number of IC slots from empty to out of use, or **ENABLE SLOTS** to toggle any number of IC slots from out of use to empty.

One then just clicks on any number of slots in the ICT lists, and presses the active **STORE** button to commit the toggle.



At any time one can click **CLOSE** to return to the recovery main form, aborting the changes if no **STORE** was performed yet...

Note that only empty slots can be declared out of use, and only out of use slots can be enabled (otherwise an error will be generated any time one tries to operate on a slot containing a mask). In practice this means that slots to be declared out of use shall have been physically emptied before by any appropriate MIJ.

**Instrument Cabinet Slots ENABLE / DISABLE**

**VIMOS** **CLOSE**

INSTRUMENT CABINETS					NEXT ACTION
slot	Q1	Q2	Q3	Q4	
1	100001	200001	300001	400001	Select the slots to be DISABLED (entry = out_of_use) <b>NO SELECTION = NO CHANGE</b> <b>and click STORE</b>
2	empty	empty	empty	empty	
3	empty	empty	empty	empty	
4	empty	empty	empty	empty	
5	empty	empty	empty	empty	
6	empty	empty	empty	empty	
7	empty	empty	empty	empty	
8	empty	empty	empty	empty	
9	empty	empty	empty	empty	
10	empty	empty	empty	empty	
11	empty	empty	empty	empty	
12	empty	empty	empty	empty	
13	empty	empty	empty	empty	
14	empty	empty	empty	empty	
15	empty	empty	empty	empty	

**DISABLE SLOTS**
**STORE**  
**ENABLE SLOTS**
**STORE**

Note that performing an IC rescan will not handle out of use slots (since they should be physically empty, they are likely to be flagged as such). Therefore if one wants to disable some IC slots after an IC rescan (which has mapped the physical content to the ICTs), one shall invoke the Enable or Disable function.

## 6.2 Cut Manager

The Cut Manager software runs on the computer designated as MMCU.

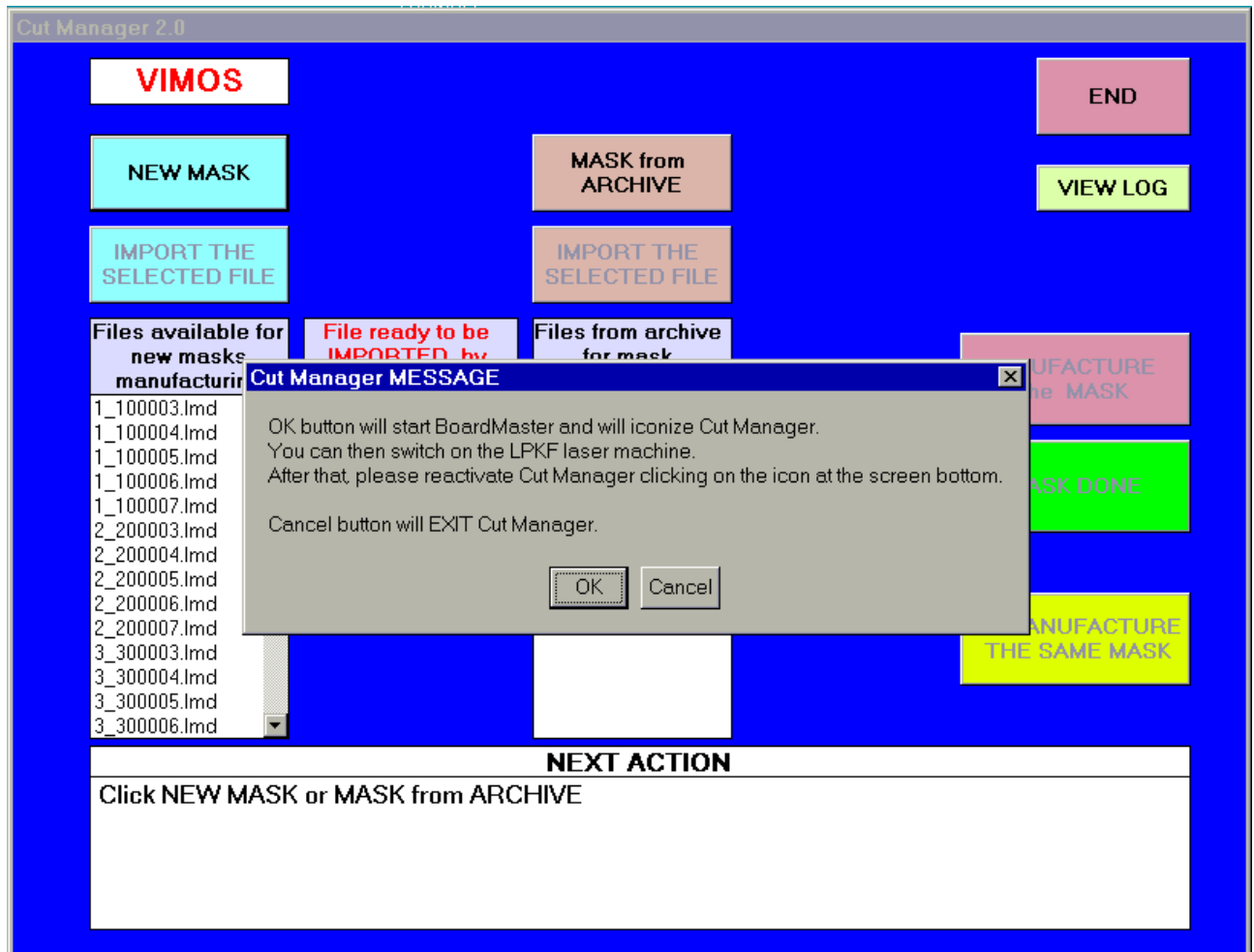
The Cut Manager is started clicking on the relevant icon (or using the start menu shortcut or the following DOS-level command C:\Mmusw\cut). As a result a *start form* (not shown, very similar to the one described in 6.1.1, except that the **LAST ACTION** button is missing) appears.

The *start form*, accompanied by a *LPKF switch-on popup* appears on the screen as shown here. Note it does not occupy the entire screen (it is not maximized) because the program requires the Windows NT task bar to interact with StencilMaster. The contingency case when an LMD file is already present in the current directory (after a crash or suspend ?) is treated in section 8.1.3.8.

The Cut Manager works in strict conjunction with StencilMaster, the program controlling the LPKF machine (this program makes critical uses of the communication ports, and therefore need to exist in a single instance. **Never** operate it manually when Cut Manager is running). When you click **OK** StencilMaster will be started (as described in 6.5)



unless it is already running (in which case you have to acknowledge a warning message), then the Cut Manager main form will be iconized.



Once the laser machine is on, re-activate Cut Manager clicking on its entry on the Windows NT task bar at the bottom of the screen. This will show again the main form, which is described here. Besides the usual “NEXT ACTION” hint box, the form includes a section with three list boxes and a few selection buttons, and a column on the right with action buttons.

In the selection section there are the following items (left to right, top to bottom)

- box indicating the instrument name
- **NEW MASK** button. This is active when no mask is selected for manufacturing. Click *this button to activate* the incoming folder list, and then click *on one of the files to select it*.
- **IMPORT THE SELECTED FILE** button. This is activated by the previous selection of a new mask. Click on it to transfer the file in the Current work area for manufacture via StencilMaster.
- *Incoming folder list box with the list of ‘new’ LMD files available for manufacturing. These are the files just produced by the MHS convert function (see 6.1.6) and corresponding to the mask in the current MMJ order. Note that the mask identifiers are prefixed with a sequence number indicating the preferred order of manufacturing. This list box is activated by the NEW MASK button*
- *Files ready list box. Contains the name of the LMD file present (if any) in the Current work area and ready for manufacture via StencilMaster. The presence of a file here activates the action button MANUFACTURE.*
- **MASK FROM ARCHIVE** button. This also is active when no mask is selected for manufacturing. *this button to activate* the archive folder list, and then click *on one of the files* to select it, if you want to re-produce a mask whose LMD files is already in the archive.





- **IMPORT THE SELECTED FILE** button. This is activated by the previous selection of a archive mask. Click on it to transfer the file in the Current work area for manufacture via StencilMaster.
- **Archive folder** list box with the list of all LMD files in the archive. The LMDs for mask which have already been manufactured are kept in the archive until a discard order (MDJ) requests to delete them.

The screenshot shows the 'Cut Manager 2.0' interface with a blue background. At the top left is the 'VIMOS' logo. Below it are buttons for 'NEW MASK' (cyan), 'IMPORT THE SELECTED FILE' (cyan), and 'MASK from ARCHIVE' (yellow). To the right are 'END' (orange) and 'VIEW LOG' (yellow) buttons. There are three file lists: 'Files available for new masks for manufacturing' (listing 2\_200003.lmd, 3\_300003.lmd, 4\_400003.lmd), 'File ready to be IMPORTED by BoardMaster' (listing 1\_100003.lmd), and 'Files from archive for mask re-manufacturing' (listing 100003.lmd through 400007.lmd). On the right side, there are 'MANUFACTURE the MASK' (orange), 'MASK DONE' (green), and 'RE-MANUFACTURE THE SAME MASK' (yellow) buttons. At the bottom, a 'NEXT ACTION' box contains the following text: 'The file 1\_100003.lmd is ready to be imported in BoardMaster. Click MANUFACTURE the MASK. Cut Manager will be iconized and the control passed to BoardMaster. After manufacturing termination, please reactivate Cut Manager.'

Once a mask (either new or from the archive) has been selected for manufacturing (its LMD is in the current area and its name appears in the Files ready list box) one can use the function buttons :

- **END** button (always active) : causes termination of the Cut Manager program. A separate manual action is required to terminate StencilMaster and switch off the laser machine as described in section 7.6.
- **VIEW LOG** button (always active) : is used to inspect the log file (not shown), similarly to what described in 6.1.2)
- **MANUFACTURE** button (activated when a mask LMD file is ready for manufacturing). Clicking on it will iconize the main form, and therefore transfer control to the underlying StencilMaster screen. Proceed to manufacture the mask as described in section 7.3.2, and when finished call back the main form selecting Cut Manager clicking on its entry on the Windows NT task bar. Then use one of the following two buttons.
- **REMANUFACTURE** button : (active when a mask has been manufactured and not yet dismissed, i.e. immediately after return from StencilMaster). If the mask just manufactured is defective or damaged and must be fabricated again the operator must click this button and follow instructions.
- **MASK DONE** button : (active when a mask has been manufactured and not yet dismissed). If the mask just manufactured is OK, click on this button to dismiss its LMD file. The LMD file will be moved into the archive if not already there.



## 6.3 Mirroring software

The mirroring program (running on the machines designated as MMCU and MHCU) will make a backup copy of the entire work area of a particular machine on a different machine, to ease recovery in case of failures as described in and in [Maintenance Manual section 12.5](#), namely as follows :

- on MHCU the D drive will be mirrored to \\mmcu\Dclone, the D drive residing on MMCU, usually mapped as the G drive
- on MMCU the E drive will be mirrored to \\mhcu\Eclone, the E drive residing on MHCU, usually mapped as the H drive

The mirroring program has no GUI, and can be invoked manually from the start menu shortcut or from a DOS command window (as C:\Mirror\mirroring), but usually will be invoked (a)periodically in an automatic way by Administrator via the AT command. Normally this is done automatically at boot time and requires no operator intervention.

The program has no arguments, it will retrieve/build the necessary information (names of master and slave drives) from a machine-dependent configuration file as described in [Maintenance Manual section 11.4.4](#). The program will ensure that any file or directory existing on the master drive is copied to the slave drive, that any file updated on the master will be updated also on the slave, and that any file or directory deleted on the master will be deleted on the slave.

In order to start the mirroring program for periodic schedule, follow these instructions :

- [log on as Administrator](#) (both on mmcu and mhcu)
- From [Settings](#) => [Control Panel](#) => [Services](#) check that the [Schedule](#) service is started (if not start it)
- From a DOS window issue the at command to check that the mirroring program is in the periodic schedule list (normally it should be so automatically even after a reboot)
- if not, [invoke the C:\Mirror\mirrorsetup.bat](#) batch to insert it in the schedule list. Such batch is a sequence of spaced commands of the form : at hh:mm /every:day mirroring The batch contains commands to run the mirroring program every 15 minutes (at times different on MMCU and MHCU), however most of such commands will be commented out. Remove the rem comment to enable running at a particular time. Note that the batch contains also a net time command to synchronize the clock of MMCU to that of MHCU (the latter is assumed master, since is the one communicating with the IWS).

In order to verify that the mirroring program is functioning properly you can (also as Operator) verify the content of the log file C:\Mirror\mirror.log. This file will record the time mirroring run last time, and any eventual file copy or delete operation, as well as any unforeseen error. There will also be a mirror.bak backup of the previous log. *Only the two last runs will be logged.* The log file will also contain the name of the account under which the program ran, and any info about mounting remote disks. Peculiar (unforeseen) errors are likely to provoke a popup to appear (this is allowed by the /interactive flag in C:\Mirror\mirrorsetup.bat).

## 6.4 CircuitCam

The LPKF CircuitCAM software runs on the computer designated as MHCU and will normally be used only in batch mode, or anyhow as called by the MHS convert function (see section [6.1.6](#)).

The batch mode is the default operation mode in convert, it can also be invoked in a DOS window via the command C:\Lpkf31\Ccam\cam file.gbr or from the Explorer using drag-and-drop of a Gerber file icon onto the CircuitCam desktop icon. In this cases the program converts file.gbr into file.LMD (in the same directory) without any user interaction and without any error checking.

The interactive mode is invoked clicking on the desktop CircuitCam icon, in a DOS window if no file argument is passed or from convert when the interactive toggle button is enabled. In the latter case the operator will use exclusively the button sequence described in the procedure per section [7.3.1](#). Usage of CircuitCAM for "special" masks is described with the relevant procedures in section [7.7](#).

For the rest the reader is referred to the LPKF online Help Files and supplied [LPKF documentation \[ref. 6\]](#).

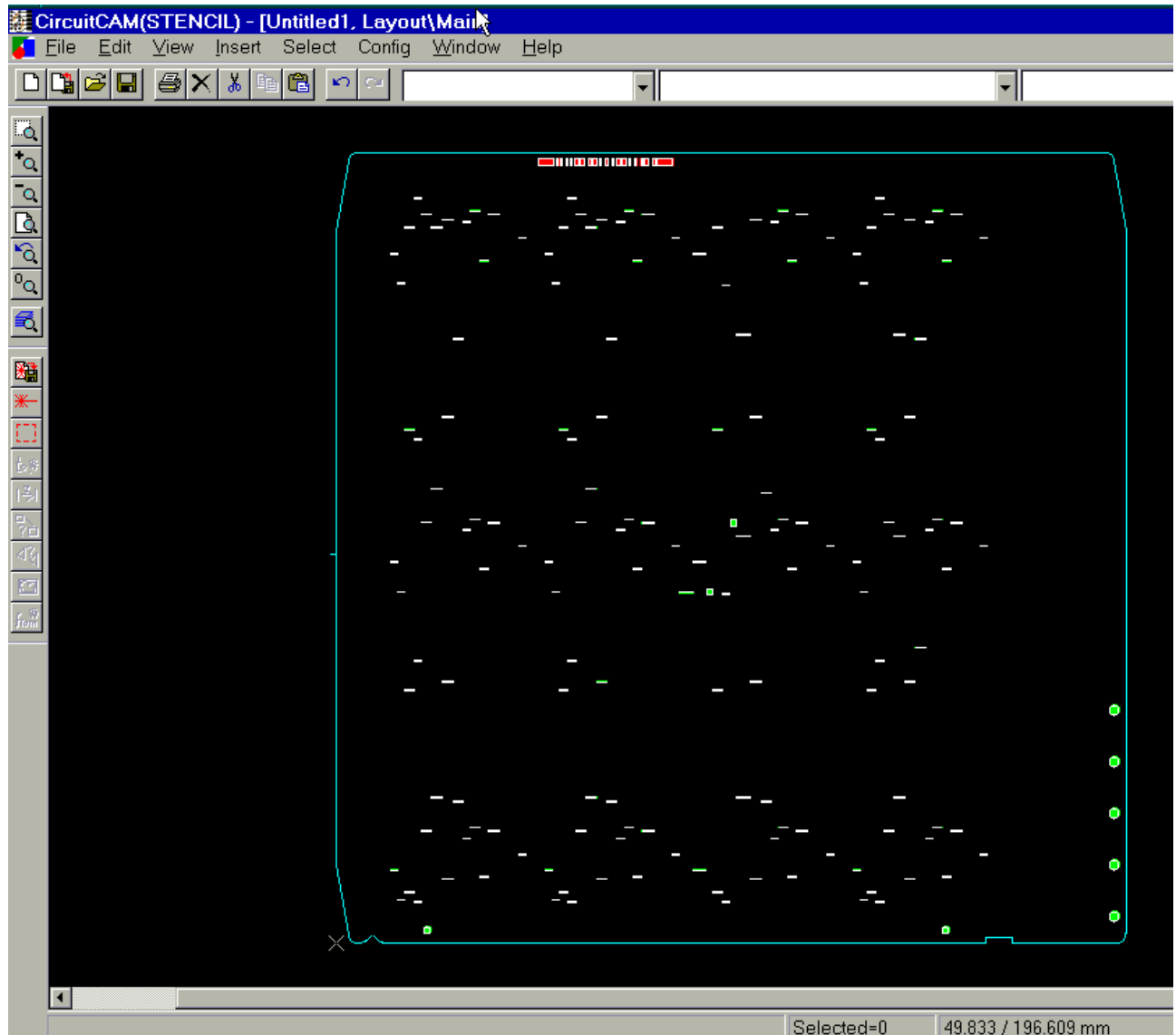
The CircuitCam screen looks as follows :

There is a menu bar, one horizontal row of buttons and other widgets, one vertical row of buttons divided in two panels, a main graphic area, and a bottom bar. For items not described here refer to the spot help, on line help or LPKF documentation .



The functions normally used are in the [File](#) or [Window](#) menus, plus the [Import](#) button (second in the first horizontal row), and the [Export](#) and [Stencil calculation](#) buttons (the top two in the second panel in the vertical row).

The graphical display is initially blank. Once a Gerber file has been loaded it shows the apertures and paths with a simple colour coding (green for apertures in the SLITS layer, red for apertures in the layer named BARCODE, cyan for the CONTOUR layer). Apertures are displayed as filled areas. The concept of layer is described in section [5.6](#) and in [Maintenance Manual section 11.5.3](#)



The [Stencil calculation](#) button computes the cutting path for any aperture or **closed** path. The resulting contour is drawn in white, and corresponds to the actual path of the laser beam which will be written in the LMD file. The above figure corresponds to this stage (just before export to LMD). This is a beam radius *inside* the aperture contour. Moreover for apertures the cut begins *at a point inside* the aperture (in a part which will fall off, so that the initial burn will not damage the aperture contour ; for paths not defined as aperture it is **care of the programmer** writing the Gerber file to ensure a similar arrangement).

## 6.5 StencilMaster

The LPKF StencilMaster software (replacing an earlier version called BoardMaster) runs on the computer designated as MMCU and will normally be used as called by the CutManager (see section [6.2](#)). The operator will use exclusively the



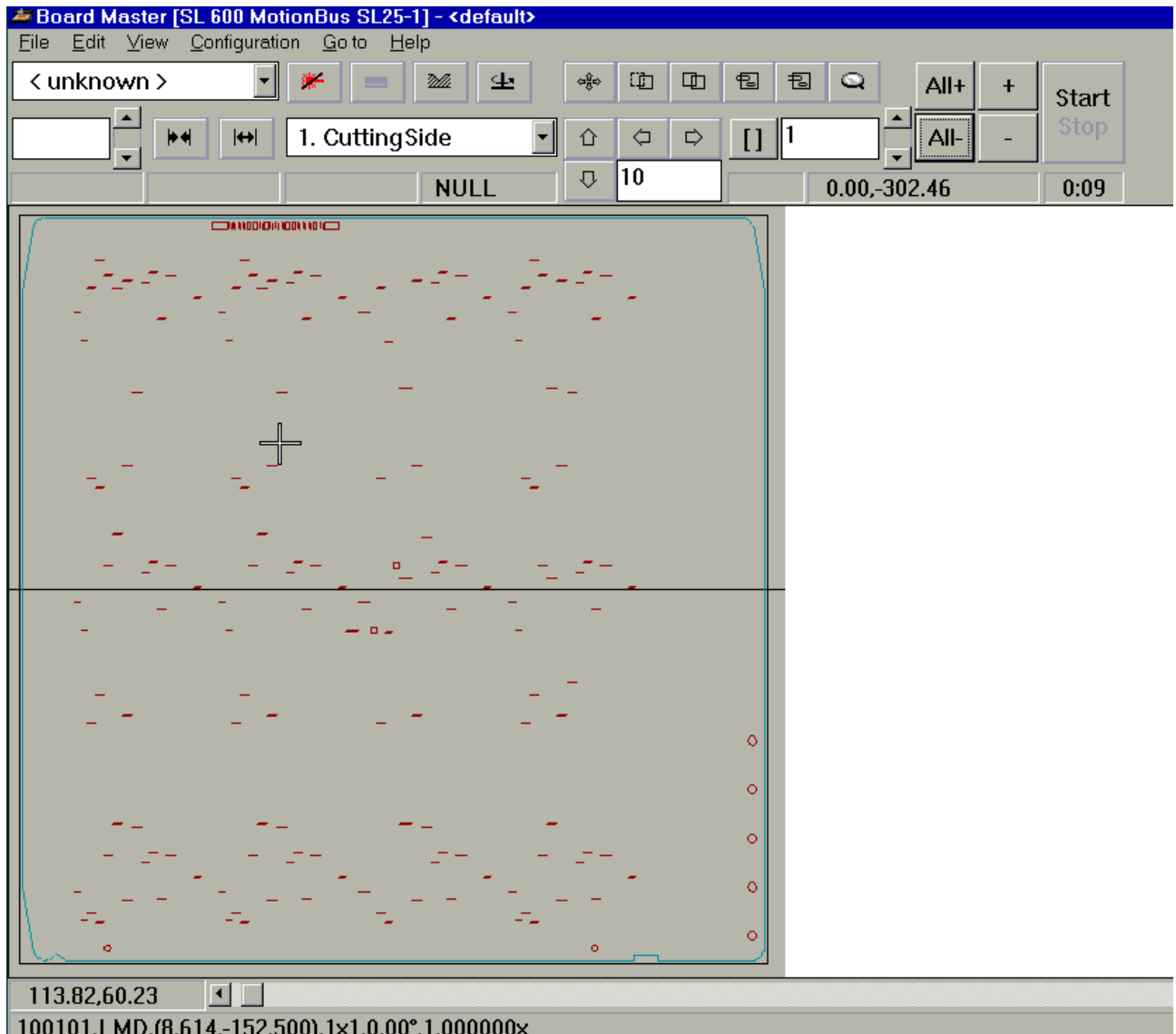
button sequence described in the procedures per section 7.2, 7.3.2 and 7.6. Usage of StencilMaster for "special" masks is described with the relevant procedures in in section 7.7.  
For the rest the reader is referred to the LPKF online Help Files and supplied [LPKF documentation \[ref. 7\]](#).

The StencilMaster startup presents a first "Start StencilMaster" popup. Reply **OK** only when instructed to do so in section 7.2. Then reply **OK** to the second "Move head to reference point" popup.  
If you want to run StencilMaster *without* switching on the LPKF laser machine, you must reply **Cancel** to the "Start StencilMaster" popup, then twice repeat the sequence of click on **Stop**, then **Yes** in the resulting popups. The program will then run without communication with the machine (this is useful only to inspect LMD files and should be **otherwise avoided**)

The StencilMaster screen looks as follows :

There is a menu bar, three rows of buttons and other widgets, a main graphic area, and a bottom bar. For most items (not described here) there is a spot help moving the mouse on them.  
The look presented here is with a standard mask loaded from an LMD file (the zoom has been arranged to show it all

using the [View=>Zoom](#) options and the  button)





The buttons and widgets normally used are the [File](#) menu, the buttons in the second row used to stretch and release the invar sheet, the menu in the second row used to select the Cutting Side or Outline phase, the [All+](#) and [Start](#) buttons, and the [Configuration=>Calibrate](#) item.

The graphical display shows the apertures to be cut in the mask (it might be necessary to use a high zoom to see tiny features like small holes, because of the limited resolution of the display). Apertures are usually displayed in a dark red colour (when unselected). They change colour when selected with the [All+](#) button (or one of the other selection buttons if a peculiar procedure requires manual cutting of individual apertures). The [Start](#) button will start the *cutting only of the selected apertures*. While they are cut, they change colour again (note that the order of cutting selected apertures is chosen by the program outside of operator control)

## 6.6 Roughness meter

The Taylor-Hobson roughness meter attached to the spare computer and its relevant software used for quality control is described in the routine maintenance section (see section [9.1](#))

## 6.7 Barcode reader configuration

The utility software used to configure the EEPROM of the barcode readers is described in the Maintenance Manual ([section 9.4](#)). Once initialized to the selected reading mode, each barcode reader retains its configuration, so this software shall be used only in case a new reader has to be installed.



## 7. Standard Procedures

This section describes the standard operational procedures of the MMU in the normal order in which they occur.

### 7.1 Logon

All ordinary operations are to be performed logging in on the MHCU and MMCU computers under the username operator, password paranal. Software maintenance operations (see [Maintenance Manual](#)) are to be performed under the username administrator password garching. For convenience the background colour of the desktop of the administrator account is *bright red*, so that the user is continuously warned that is using a reserved account. It is advised to change such passwords at ESO convenience, e.g. after commissioning.

Before logging in on either MHCU or MMCU as operator, **make sure that the other computer** (MMCU resp. MHCU) **is switched on and connected to the network** : correct usage of the software assumes disk sharing between the two computers, and provides to map (mount) the relevant disks during logon.

### 7.2 Laser switch on

The full procedure for the switch on of the laser cutting machine is the following :

- make sure you are logged in on the MMCU computer as operator
- make sure the granite surface and the metal working plane of the laser machine are clean and eventually wash them with an appropriate alcohol-based detergent for hard stone surfaces, and wait until it is dry
- make sure the chiller is on (or switch it on)
- make sure the erogation of pressurized air from your local compressor is functioning
- switch on the dryer filter (red, using the lighted switch which will also power - but *not* switch on - the exhauster)
- make sure the pressure duplicator manometer reads a value between 15 and 16 bar
- for normal operation on the MMCU start the Cut Manager program clicking on the relevant icon *and then clicking START*); see section 6.2
- once you click **OK** on the **CUT MANAGER MESSAGE POPUP**, this will start StencilMaster (as described in section 6.2) . For the manual manufacturing of "special masks" (see section 7.7) start directly StencilMaster clicking on its icon. StencilMaster will in both cases display a "switch machine on" popup. **Do not click OK now but proceed through the following steps.**
- open the front door of the LPKF control rack
- turn on the main switch : the rack LCD display will briefly show a "self-test 1" message : wait for its completion
- when the rack display instructs it is OK to proceed, press the SYS ON button
- wait for the "self-test 2" message to show completion :
- when the rack display instructs it is OK to proceed, turn the key on, and also press the shutter open button
- when the LCD display says "waiting for head down", you can click OK in the StencilMaster popup.
- A further popup will say "machine head will move to reference point" : click OK
- There may be a delay of a few minutes before StencilMaster allows interaction with the mouse, even after the main StencilMaster screen has appeared. Once you gain control (i.e. when the cursor changes from a crosshair to an arrow), you can proceed to use the laser machine for cutting either under assistance of the Cut Manager or via direct use of StencilMaster
- Check the digital manometer for the air pads (located behind the granite base of the StencilLaser machine) : it should read a value between 5.6 and 5.7 bar.
- make sure that the vacuum exhauster is on before starting actual cutting (you may turn it off when you are not cutting). For convenience the exhauster switch is located close to the laser head.

### 7.3 Manufacturing Cycle

The full manufacturing cycle is the processing of a MMJ. It starts with the retrieval of the MMJ delivered by MCS (and of the relevant MSFs), and terminates when all masks referred in the order have been manufactured and safely stored in the correct SC (and storage has been recorded in the SCT file). The manufacturing cycle goes through the following steps :



- “MHS convert” (on the MHCU, the operator retrieves the MMJ order and associated MSF, and converts the latter to Gerber files and finally to LMD files stored on the MMCU)
- usage of Cut Manager on the MMCU to assist the actual mask manufacturing
- “MHS store” (on the MHCU, to assist the storage of masks in the SCs, and finally make the MMT report available to MCS)


The conversion step is run once before the other two for all masks listed in a MMJ. The manufacturing and storage steps typically proceed in parallel (one produces the masks for a mask set and stores them, either one at a time or all together, then proceeds to produce the eventual next mask set in the next MMJ).


### 7.3.1 MHS convert


The conversion process starts with the retrieval of one MMJ delivered by MCS, which becomes active. It cannot proceed if there are no MMJs, and one cannot process a new MMJ if the store phase of the last active MMJ is not completed.

The MMJ is an order, a list of masks to be manufactured. This step usually advances manually but very quickly through the various steps :

- make sure you are logged in on the MHCU computer as operator
- if you have not yet done so, start the MHS subsystem clicking on its icon, *and then on START*; see section 6.1.1.
- on the top bar, click on the CONVERT button (refer to section 6.1.6)
- click the SELECT AND RETRIEVE button, and use the order selection form to select the MMJ to be processed. This stage is bypassed if a pending convert session on a previously retrieved MMJ is resumed. The program terminates here if there are no MMJ in the staging area.
- if an MMJ is found, the relevant files (the MMJ itself and the MSFs, one per mask) are moved from the staging area to the MHS work area, and the MMJ becomes active
- The MSFs are checked, each one in turn, for completeness (all MSFs listed in the MMJ must exist) and for syntax. An error at this stage will cause the omission of further processing for that mask, but the overall convert process continues.
- click on the TRANSLATE button to convert all MSFs in Gerber format files. Here too erroneous files will be skipped.
- click on the SUBMIT button to convert Gerber format files into the LPKF proprietary LMD format. This conversion is performed by the LPKF CircuitCam software. The latest release of CircuitCam is able to process Gerber files in a batch mode, but we include a description also of the old procedure where one instance of CircuitCam is spawned to assist with *the manual conversion of one file at a time* as follows
  1. a panel with the list of Gerber and LMD files is shown and you click on the **START** (first time only) or **CONTINUE** button (or just press return if the button is selected)
  2. CircuitCam starts and, if default **BATCH MODE** is selected, silently does its job, and **jumps to step 9 below**. Otherwise, if **INTERACTIVE MODE** is selected, it starts and occupies the screen. In such case proceed from the next step.

3. From the second (horizontal) row menubar select the Import button 
4. you are presented an **Open** popup **a single file** just select it and click Open (or doubleclick). You are presented an **Import** popup for Gerber-X files, accept it clicking OK
5. if the Gerber file looks complete and you do not get any error message, you can proceed, otherwise skip to step 8; you may also inspect the CircuitCam report screen using **Window=>Report** (for unexplained reasons this will show only the first 113 apertures); and use **Window=>Untitled** to come back

6. click on the stencil calculation icon (second from top in the second vertical panel on the left)  to perform the calculation of the cutting path

7. click on the export icon (just above the previous)  > to create the LMD file, current version present a **Save as CAM** popup, press **CANCEL** to create only the LMD and acknowledge the resulting popup
8. terminate CircuitCam selecting **File=>Exit** and replying **No** to the "Save changes" popup.



9. if the LMD files is succesfully created, MHS will present an acknowledgment **POPUP** (click OK or just press return) and let you proceed to step 1 for the next mask, otherwise a **POPUP** asks to retry conversion or to skip this mask
- at any time you can, if wished, suspend `convert` either clicking the main **END** button, or the **SUSPEND** button in the Gerber/LMD file panel. In this case next time the Gerber-LMD conversion will skip, notifying you, all files already converted.
  - when the LMD generation is complete, MHS will tell so in the appropriate message area : follow instructions, and clicking on the main **END** button will move all LMDs belonging to a complete mask set to the appropriate disk on the MMCU. If a mask set is incomplete (some of the MSFs in the order have given an error at any stage of the conversion) no LMDs are copied to MMCU, and all relevant files are deleted, so that no mask of the mask set will be manufactured. Since current policy allows only one mask set per MMJ, de facto this means that an incomplete mask set will generate an unsuccessful MMT report without need to proceed to manufacturing and storage
  - once the `convert` stage is terminated you may start the `store` stage on MHCU, and at the same time move to MMCU for mask manufacturing.




### 7.3.2 Cut Manager assisted mask manufacturing

Mask manufacturing is a manual operation controlled by the StencilMaster program and assisted by the Cut Manager program. The StencilMaster program is in full control of the laser cutting machine, **it must be used to switch it on and off**, and only a single instance of it can be active at any time. StencilMaster is used standalone for the manual manufacturing of special masks.

The function of the Cut Manager is just to assist to keep track of which masks have to be manufactured. The LMD files generated by the convert session on the current MMJ are stored in an “*incoming*” area on the MMCU. Cut Manager moves one of them at a time in a “*current*” work area, and passes control to StencilMaster for manufacturing. Once you declare manufacturing is finished, the LMD file is moved to an “*archive*” area, where it is available for reproduction of the same mask until discard. Cut Manager allows also to assist mask reproduction, moving existing LMD files from the archive to the current area.

Typically one manufactures a mask of a mask set, and stores it in the SCs either immediately after cutting, or while cutting the next mask (to do this the store function of MHS must be active on MHCU as described in 7.3.3). In practice this occurs through the following steps.

1. Make sure the laser machine has been switched on as described in section 7.2, and the Cut Manager and StencilMaster programs are running
2. Click on CutManager icon in the Windows NT task bar on the bottom of the screen
3. In CutManager select the source of LMD files (the incoming area for new masks) clicking on the **NEW MASK** button (this will also refresh the list for any files converted in the meantime), select the first mask (according to the sequence numbering) and transfer it to the current area clicking **IMPORT THE SELECTED FILE** then click on **MANUFACTURE** to pass control to StencilMaster
4. In StencilMaster select File =>Import =>LMD (from the **File** menu) : this will show an "import Project" popup with a single file in the *current* area, select and open it clicking on **OK** (the content of the LMD file will appear on the screen)

5. Use the **button**  (second in the second row) to release the clamping unit on the machine
6. insert a new invar sheet in the clamping unit until the marks
7. lock the clamping cams on both sides
8. Use the **button**  (third in the second row) to stretch the material
9. Peel the material: use the  button to advance the clamping frame towards you and position the laser head clearly out of the way : click on the button, then click in the uppermost part of the frame showing the work area to move. Then with a cutter cut the protective film as close as possible to the borders of the clamping frame, then pull





off the film, pulling as parallel as you can to the Invar surface, from one corner in the X direction towards the opposite border

10. **make sure** the **current phase menu** (fourth item in second row) **indicates the CuttingSide working phase**
11. click on the ALL+ button (all slits and barcode will be selected and change colour from dark red to white)
12. make sure the vacuum exhaustor is on
13. click on START to start cutting
14. the display on the control rack shows a short self test, and then changes to indicate the power and other settings. Note the power does not immediately reach the operating value. The machine will then start moving and cutting. A typical VIRMOS mask will take 6-10 minutes to be cut.
15. while cutting you can use the eyepiece to check the operation of the laser and the cleanliness of the nozzle
16. once the phase is over, click on **OK** to acknowledge (the popup might appear a few seconds before the machine actually stops working)
17. Select the Outline phase in the **current phase menu**
18. click on the ALL+ button
19. click on START to start cutting the contour
20. walk to the machine and support the mask while the contour is being cut, eventually using the provided magnets, then remove the mask
21. put the mask aside in some intermediate place (typically on the barcode stand) awaiting for storage (if one has time one can store it immediately, see step 5 of **7.3.3**)
22. walk to the MMCU and acknowledge the end of the phase (**OK** response to popup)
23. Use the **second button** in the second row button to release the material (as in 5)
24. remove the Invar and throw it away
25. click on File =>New=>default (and answer **Cancel** to the **StencilMaster Save Popup**) to clear the screen
26. **Select** the **CuttingSide phase** in the **current phase menu (to be ready for the next mask !)**
27. return control to the Cut Manager, clicking on its icon in the Windows NT task bar on the bottom of the screen
28. if the mask is successfully manufactured, click on MASK DONE to move the LMD to archive. If you want to play safe you may store it (see step 5 of **7.3.3**) before clicking on **MASK DONE**
29. if you have not yet stored the mask and want to save time by concurrent operation proceed to step 3 to manufacture a new mask, and while it is being cut proceed to store the previous one on the MHCU (see next section **7.3.3**)

It is possible to speed up slightly the operations (after the first time) mounting a fresh Invar sheet at the same time as soon as one has removed and thrown away the used one (e.g. after step 22 do 25,26,27,28,3,4,23,24, then 5 onwards).

It is recommended to remove the mask as soon as it has being cut (step 20). It is not advisable instead to defer removing the mask after the release of the Invar (step 23) because the mask may get damaged while the machine moves to the centre for Invar release.

If a mask is damaged during storage, it has to be remanufactured. The way to do this depends on the moment in the manufacturing sequence when the mask is stored :

- if the storage is attempted immediately after cut (step 21), its LMD is still loaded on StencilMaster. To remanufacture it execute steps 22 to 24 to remove the Invar, and then jump to step 5 to cut it again.
- if the storage is attempted at the end of the loop (step 28), its LMD is no longer in StencilMaster, but it is still in the current area. One shall click on **REMANUFACTURE** (instead of **MASK DONE**) then proceed to step 3 as usual to reload and cut it again
- if the storage is attempted while the next one is being cut, its LMD is in the archive. In such case dispose of the mask being cut, then reproduce the previous one clicking on **MASK FROM ARCHIVE** (instead of **NEW MASK**) at step 3 and then proceed as usual. You should know the mask id (without sequence number prefix) : the fact that the list in the archive panel is sorted shall make easier to locate it.



### 7.3.3 MHS store

The mask storage procedure runs in parallel to the mask manufacturing

1. if you have not yet done so, start the MHS subsystem clicking on its icon, *and then on START*; see section 6.1.1.
2. on the top bar, click on the STORE button (refer to section 6.1.7). You can leave MHS in store mode as long as you wish. If you ever need to use MHS for other tasks you can **SUSPEND** store (it will be *resumed* the next time you click on **STORE**)
3. if you are entering store for the first time, or if otherwise instructed in the “NEXT ACTION” hint box, click INITIALIZE to connect the bar code reader.
4. For each mask set aside after manufacturing, click on NEXT MASK
5. pick up the mask set aside and place it in the MHS robot mask stand
6. as soon as the bar code has been read and validated a **POPUP** (labelled S08) appears
7. put the mask in a free slot in the respective SC quadrant (as instructed : note the **NEXT ACTION** hint box background colour changes for a quick visual reference of the quadrant !) taking care to place the barcode onto the white marked sections
8. reply **OK** to the above mentioned **POPUP**
9. go back to step 4
10. once all mask sets have been manufactured and stored, click **END**. This generates the MMT report for MCS, gets rid of the MMJ order and all associated files (but the LMD files, which are kept archived on MMCU until discarded) and thus terminates the manufacturing cycle for this mask set. If there are MMJs pending for other mask sets start a new manufacturing cycle.

## 7.4 Load/Unload Cycle

The load/unload (IC preparation) cycle is the processing of a MIJ. It starts with the retrieval of the MIJ delivered by MCS, and terminates when all masks required for the observations have been loaded in the ICs; contextually unneeded masks will be moved back from IC into SC, while the ICT and SCT files will keep track of where masks are stored. The full cycle goes through the following steps :

- “MHS unload” (on the MHCU, the operator retrieves the MIJ then is assisted in unloading all not needed masks in IC back into SC; masks needed and already in IC remain there)
- “MHS load” (on the MHCU, the operator is assisted in loading into IC any needed mask not already there)

The two steps are handled in sequence by a single MHS function (IC preparation, see section 6.1.8). The process starts with the retrieval of one MIJ delivered by MCS. Of course it cannot proceed if there are no MIJs as well as if the ICs are not physically available. Usually there is a single MIJ at a time : if more are present, the operator can select one, but must get in personal contact with the MCS astronomer to know which is the right one. The operator is however always able to select autonomously the null order to empty the IC for maintenance purposes. The MIJ is a list of masks to be loaded in the IC for the next observing night(s) : the list of masks to be unloaded is built dynamically as complement.

The following is a description of the cycle :

1. make sure the ICs have been dismantled from the telescope and are available to you
2. make sure the ICs are correctly inserted and firmly clamped in the IC holder : check the colour codes, remember that quadrant 1 IC is the topmost, and that in each quadrant slot 1 is on the top and slot 15 of the bottom of the IC when inserted in the relevant IC holder drawer.
3. make sure you are logged in on the MHCU computer as operator
4. *if you have not yet done so, start the MHS subsystem clicking on its icon, and then on START*
5. on the top bar, click on the IC PREPARATION button (refer to section 6.1.8)
6. make sure the MHS robot is powered on, and has been moved to the home position (if it is not so the software will lead you through the right stages)
7. click the RETRIEVE button in the unload form, and use the order selection form to select the (usually unique) MIJ to be processed. This stage is bypassed if a pending unload/load session on a previously retrieved MIJ is resumed. The program terminates here if there are no MIJ in the staging area. It can also be conditionally terminated in case of errors (masks in the order not found in SCT or ICT, see 8.1.5.28). *During this stage you will also be asked to*



*check (for each quadrant) to which IC set (original, A or spare, B) the ICs in use belong via the IC configuration form*

8. the program will move the robot Z-displacement unit platform to the loading position (if requested and acknowledged answering **POPUPs** U04 and U05), mount the IC holder trolley on the platform (if not already mounted) and check it is firmly secured to it, then reply to the **POPUP** U06 and the robot will move to the home position
9. if instructed to do so (at least one mask to be unloaded) click on **IC UNLOADING**
10. the program moves the MHS robot Z-displacement system so that the mask stand is in correspondence of the next mask to be unloaded (it operates one quadrant at a time)
11. use the mask clamp to extract the mask into the mask stand (first push the open clamp to grab the mask in the IC slot, then pull towards you to extract the mask fully), and acknowledge **POPUP** U07
12. the bar code of the mask is read and verified
13. pick up the mask and insert it in a free slot in the right SC quadrant (as instructed : note the **NEXT ACTION** hint box background colour changes for a quick reference of the quadrant). Note no further identification is made in the SC.
14. acknowledge the correct storage replying **YES** in reply to **POPUP** U11
15. if there are more masks to be unloaded the program goes back to step 10
16. if instructed to do so (no more masks to unload) click **END**, otherwise it will proceed from step 27
17. the load form appears after acknowledgment
18. if instructed to do so (at least one mask to be loaded) click **NEXT MASK**
19. while robot moves walk to the SC and search the mask in the right SC quadrant (note the **NEXT ACTION** hint box background colour changes for a quick visual reference of the quadrant) moving the SC bar code reader
20. when the wished mask is located, the program beeps insofar the bar code reader laser beam is on the mask
21. pick up the mask from the SC and acknowledge clicking **MASK FOUND**
22. put the mask in the MHS robot mask stand for identification and reply **YES** to the **POPUP** L05
23. the program has already moved the MHS robot Z-displacement unit to put the mask stand in correspondence of the first free slot in the right IC quadrant
24. use the mask clamp in closed position to push the mask and insert it fully into the IC
25. acknowledge the correct insertion replying **YES** in reply to **POPUP** L08
26. if there are more masks to be loaded the program instructs to go back to step 15
27. if instructed to do so (all masks loaded) click **END**. This generates the MIT report for MCS, gets rid of the MIJ order and thus terminates the unload/load cycle
28. the Z-displacement system will move to allow unloading the IC holder trolley, wait for the movement to complete and take out the IC holder.
29. the ICs are ready to be picked up and mounted on the telescope

Note that, while the motion of the robot always occurs from quadrant 1 to quadrant 4 and from slot 1 to slot 15, at present the loading of the ICs tries, as far as possible, to insert a mask every other slot, leaving one free slot in between (which is possible only if less than 8 masks are required to be ultimately in the IC). In practice the program computes the list of preferred slots at the beginning of the load phase (leaving alone the slots already occupied) and then puts the mask in such slot in order of increasing slot number to optimize robot motion.

*Note also that the ICT tables refer to whatever ICs are physically in use, irrespective of the fact they belong to the original (A) or spare (B) set. The IC configuration form is used only to tell the system which set is in use for the purpose of robot positioning (there are two sets of position tables but not of ICTs !). Therefore it is recommended that, before changing the IC configuration, the ICs becoming not used are emptied. To make sure ICTs and SCTs are properly updated, this should be done using an appropriate insertion order, like the resident null order (see 8.3.7), or a null MIJ generated at mask tracker level*

## 7.5 Discarding Cycle

The discarding cycle is the processing of a MDJ. It starts with the retrieval of the MDJ delivered by MCS, and terminates when all masks listed in the order have been physically discarded (and the relevant LMD files eliminated, and the SCT updated). The cycle is managed by a single computer assisted MHS function (discard, see section 6.1.9). The process starts with the retrieval of one MDJ delivered by MCS. It cannot proceed if there are no MDJs. The MDJ is a list of masks to be discarded. Once discarded (removed from SC and thrown away) a mask cannot be referred by any



further MIJ, nor can be fabricated again (the LMD file on MMCU is deleted). The following is a description of the cycle.

1. make sure you are logged in on the MHCU computer as operator
2. if you have not yet done so, start the MHS subsystem clicking on its icon, *and click START*, see section 6.1.1.
3. on the top bar, click on the DISCARD button (refer to section 6.1.9)
4. click the RETRIEVE button in the discard form, and use the order selection form to select the MDJ to be processed. This stage is bypassed if a pending discard session on a previously retrieved MDJ is resumed. The program terminates here if there are no MDJ in the staging area.
5. if instructed to do so click SEARCH MASK
6. search the mask in the right SC quadrant (note the **NEXT ACTION** hint box background colour changes for a quick visual reference of the quadrant) moving the bar code reader
7. when the wished mask is located, the program beeps insofar the bar code reader laser beam is on the mask
8. acknowledge clicking **MASK FOUND** and pick up the mask from the SC
9. throw away the mask in the waste basket
10. acknowledge discarding (in order to update the SCT and delete the LMD file)
11. if there are more masks to be discarded the program instructs to go back to step 5
12. if instructed to do so (all masks discarded) click END. This generates the MDT report for MCS, gets rid of the MDJ order and thus terminates the discarding cycle

## 7.6 Laser switch off

The correct procedure for the switch off of the laser cutting machine is the following :

- if you are running StencilMaster via Cut Manager, terminate Cut Manager clicking **END** and confirming **OK**
- make sure there are no invar sheets mounted on the machine
- terminate StencilMaster using **File => Exit**
- reply OK to the first **popup** and the cutting head moves to the corner
- reply Cancel to the eventual "**StencilMaster Save**" popup
- open the front door of the LPKF control rack
- press the shutter close button
- turn the key off
- **wait at least 3 minutes** for the laser lamp to cool
- disregard the message on the LCD (relevant to a switch on) but press the SYS OFF button
- turn off the main switch
- switch off the vacuum exhauster (if not already off)
- switch off the chiller (if you do not plan to reuse the machine in a short time)
- switch off the dryer filter
- turn off compressed air erogation



## 7.7 Special masks

There are masks and other items which periodically (or on demand) can be manufactured “autonomously” on MMU without explicit request (order) from MCS. This includes the cases described in the following two sections.

## 7.8 maintenance masks

Permanent masks are used for periodic maintenance and calibration of the instrument (e.g. alignments, routine calibrations, etc.). Although these masks are not used for normal scientific observations, they might be anyhow identified by reserved bar codes. Some of these masks will be used for periodic calibrations (hereafter named *calibration masks*) and will be loaded into ICs under MCS request (MIJ) as if they were normal masks. Some other masks will be used only extraordinarily during instrument integration, or for re-alignment of the instrument when dismantled (hereafter named *engineering masks*). All these masks might never be discarded : one specimen of them could be kept in SC or elsewhere (when not deployed in the IC), but in case of wear it might need to be replaced by a new copy. The ultimate policy will be finalised after commissioning.

### 7.8.1 Definition of permanent masks

At the time of writing the following kinds of permanent masks are defined and eventually present in SC (quadrant  $q=1..4$ ). Mask id' s below 10 are reserved to engineering masks, mask id' s between 10 and 99 are initially reserved to calibration masks.

- file `eng_black-q.lmd`, mask id `q00000`, a black (blank) mask with no apertures
- file `eng_05s_cross-q.lmd`, mask id `q00001`, alignment cross on the optical axis
- file `eng_05s_horslit-q.lmd`, mask id `q00002`, a very long horizontal slit (divided in sections)
- file `eng_1s_horslit-q.lmd`, mask id `q00003`, similar to the previous, different slit height
- file `eng_01m_image-q.lmd`, mask id `q00004`, a pattern of 16 0.1 mm holes for image quality
- file `eng_01m_omni-q.lmd`, mask id `q00005`, a matrix of 0.1 mm holes plus additional patterns
- file `eng_05s_omni-q.lmd`, mask id `q00006`, similar to previous but 0.5 arcsec holes
- file `eng_05s_dotcross-q.lmd`, mask id `q00007`, a cross pattern of holes along the optical axes
- file `cal_05s_pinhole-q.lmd`, mask id `q00010`, the mask to CCD alignment matrix (latest version)
- file `cal_05s_LRstartrace-q.lmd`, mask id `q00011`, the Low Resolution grism star trace mask (idem)
- file `cal_05s_MRstartrace-q.lmd`, mask id `q00012`, the Medium Resolution grism star trace mask (idem)
- file `cal_05s_HRstartrace-q.lmd`, mask id `q00013`, the High Resolution grism star trace mask (idem)
- file `cal_mos_flux-q.lmd`, mask id `q00014`, slits of different height for spectrophotometric calibration
- file `cal_2pix_Mshut-q.lmd`, mask id `q00015`, a set of slits in the centre for mask shutter calibration

All these masks are defined by a Gerber file (stored on MHCU in `D:\Mhs\Vimos\Archive\Gerber`), and a corresponding LMD file (working copy stored in the Permanent archive on the MMCU, backup copy in `D:\Mhs\Vimos\Archive\LMD`). No operator intervention is normally required to maintain such files. In the near future calibration masks referred to in operational templates will not be permanent in this sense and will be manufactured on the basis of MMJs and MSFs (and numbered) as normal masks, and later discarded via MDJs as normal masks.

In the case it will be necessary to define new kinds of permanent masks, a new set of Gerber files must be created manually with an editor, and converted into LMD format by *manual* usage of CircuitCAM (which means either interactive usage, or manual invocation of CircuitCam with a file argument, or drag-and drop operation, see 6.4). The description of this task is outside the scope of the present document. (we note however that, in order to ease the generation of bar codes, we have found convenient the usage of *dummy orders*, like the ones stored in the subdirectories of `D:\Mhs\Vimos\Archive\maintenance_orders`). **We intend to accompany each maintenance mask with a data sheet describing how it was generated.** The idea is to generate dummy MMJs and MSFs with the desired mask id, and any geometrical characteristics which can be described as a slit. The dummy orders can then be moved *manually* to the staging area and submitted to `convert`. Once the first phase of the conversion (the generation of the Gerber files) has been done, one might suspend `convert` and manually edit the Gerber files to insert any further feature (note that MHS supports, for maintenance purposes, also not rectangular slits, like the step-and-repeat holes used in



cal\_pinhole-g.lmd or round apertures like for eng\_image-g.lmd, thus in such cases no manual editing is necessary). These features can be verified in CircuitCam - invoked by convert in interactive mode - and new ones, like lettering, added within CircuitCam, which will finally produce the LMD files. The LMD files on MMCU shall be moved *manually* from the Incoming to the Permanent directory, and any MMT report *manually* deleted.

The convention that permanent mask file names end with -g (hyphen and quadrant number) is an OS requirement for calibration masks. DRS analysis of such masks require the existence of ADM files (which are normally created by MCS for standard observations). An utility to convert Gerber to ADM files (vmmcsGbr2adm) exists inside the VIRMOS MCS (vmmcs, see [section 10.1 of \[ref.3\]](#)). The usage of dummy orders (necessary to maintain engineering masks) will be necessary even for calibration masks which are not permanent to "bootstrap" the ADP/ADM files (from which MCS will later generate MSFs) from Gerber files.

## 7.8.2 (Re-)production of permanent masks

In the case a permanent mask is damaged or worn out, it will be necessary to produce a new copy. In this case the operator shall proceed as follows :

- Remove the worn out calibration mask from SC using the Recovery Remove function (see [6.1.10.4](#)), or throw away the worn out engineering mask.
- Manually produce a new mask from the LMD file in the Permanent archive using directly StencilMaster (CutManager cannot handle this archive). In any case while in the Open... popup of StencilMaster one must explicitly select the Permanent subdirectory and the relevant LMD file
- Manual production has to be done with special precautions (see below)
- Insert the new calibration mask in SC using the Recovery Insert function (see [6.1.10.3](#)), or store new engineering mask in reserved place elsewhere.

Special masks may involve the cutting of large pieces of Invar out of the raw sheets. These pieces, if not removed, may mechanically obstruct the entrance of the vacuum extractor, and cause damage to the entire mask. Therefore it is necessary to remove manually each piece once cut. Therefore each piece must be selected and cut individually within StencilMaster. The procedure and preferred order is described below for each kind of mask which will require special handling.

- The list of masks requiring this sort of special handling is currently empty

## 7.8.3 Deployment of permanent masks

Permanent calibration masks are loaded in ICs via standard (MCS generated) MIJ as any normal mask.

Permanent engineering masks are not stored in SC, are not managed by the MHS software, and will be inserted manually in the IC when the instrument is put off line for dismounting and extraordinary maintenance.

## 7.9 Samples for quality control

Quality samples need to be cut periodically out of standard Invar sheets. These samples are not masks, and are not identified by a barcode. They will never be used at the telescope, but just measured with the roughness meter in order to assess if a standard set of quality parameters are within required limits. Once measured, the samples will be kept or thrown away according to policy to be established by ESO Typically one might produce more than one specimen for some kind of samples.

### 7.9.1 Definition of quality samples

At the moment the following kind(s) of quality samples are defined

- file test\_rug (contains a 3x3 matrix of 35 mm square sample *and* one slit sample)
- file test\_lab (contains the numeric labels for the above samples)

All these samples are defined by a Gerber file (stored on MHCU in D:\Mhs\Vimos\Archive\Gerber), and a corresponding LMD file (stored in the Permanent archive on the MMCU where eventually one LMD contains more samples). No operator intervention is normally required to maintain such files. The file test\_lab has no Gerber equivalent (the LMD has been manually generated within CircuitCam).

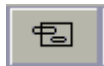


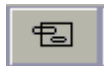
In the case it will be necessary to define new kinds of samples, new Gerber files must be created manually with an editor, and converted into LMD format by manual usage of CircuitCAM (which means either interactive usage, or manual invocation of CircuitCam with a file argument, or drag-and drop operation, see 6.4). CircuitCAM (via appropriate programming of the Gerber file) or StencilMaster (combining repeated instances of the same LMD project at different positions eventually in a JOB file) can be used to place multiple instances of the same sample at different machine positions into an LMD file. The description of this task is outside the scope or the present document.

## 7.9.2 Production of quality samples

The production of quality samples involves the cutting of many sizeable pieces of Invar out of the raw sheets. These pieces, if not removed, may mechanically obstruct the entrance of the vacuum extractor, and cause damage to the entire sheet. Therefore it is necessary to remove manually each sample once cut. Therefore each sample must be selected and cut individually within StencilMaster. The procedure and preferred order is described below. It is assumed that an Invar sheet is mounted for cutting as described in section 7.3.2.

- within StencilMaster import file `test_lab.lmd` from the Permanent archive, make sure you are in the CuttingSide working phase, select all items (**ALL+**) and press **START** to cut the numbers. Do not cut the contour. Do not remove the Invar sheet.
- clear the screen selecting **File=>New=>default** and import file `test_rug.lmd`, then on the same Invar sheet, cut manually one at a time each of the 9 samples as follows



- with the  button (4th button of the second group of 6 i.e. "enable selection of *elements* crossing...") select one of the samples clicking on a corner
  - click on the + button to select it for manufacturing
  - press **START** to cut it
  - then carefully remove the sample with a magnet and set it aside for later measurements. Be careful not to touch the sides of the cut with the magnet, nor to let them touch the edges of the cut. The 9 samples will be labelled from A1 to C3 because of the previous cut of the labels.
- when the 9 roughness samples have been cut, use one of the selection buttons to select all the slits in the slit sample, then click on the + button to select them for cutting
  - press **START** to cut the 6 slits
  - then select the Outline phase and press the **ALL+** button to select the contour of the slit sample for cutting
  - press **START** to cut the contour of the slit sample, and remove the sample with a magnet
  - finally remove the Invar sheet.

In addition to the standard quality samples there is also a "comb" pattern which can be cut for a quick assessment of the cut quality when it is necessary to adjust the micrometer for the height of the laser head. The relevant file can be found in `E:\Vimos\Special_masks\mu300.job` on MMCU, and is accessed in StencilMaster via the **File=>Open** (since it is a job and not an LMD one shall not use **File=>Import**). It can then be cut as normal, and inspected by eye for the presence of burns.

Routine quality inspection of normal masks shall also be performed by eye looking at the presence of obvious irregularities in the contour shape due to improper clamping or support of the Invar sheets while cutting.

## 7.9.3 Usage of quality samples

You have 9 *roughness samples*, for which you can make a roughness measurement on each side for a total of 36 measurements. The numbering of the sides is illustrated in the figure below. If you follow the procedure described below, you will collect the result quite easily. However the measurement itself and the data transfer to the spare computer will take about 3 minutes per measurement.

There may be reasons to make less than 36 measurements. If one measures only the top side of each sample (total 9), i.e. those numbered 0,4,8...32, it will be sure that the edge is not damaged nor will collect dirt from the bottom of the vice. If one measures only the top and bottom sides (total 18, those labelled with even numbers), it will not be affected by



possible distortions in the vertical sides due to the large size of the piece moving while being cut off : such distortions are not representative of the real slit case, and may affect the waviness parameters).

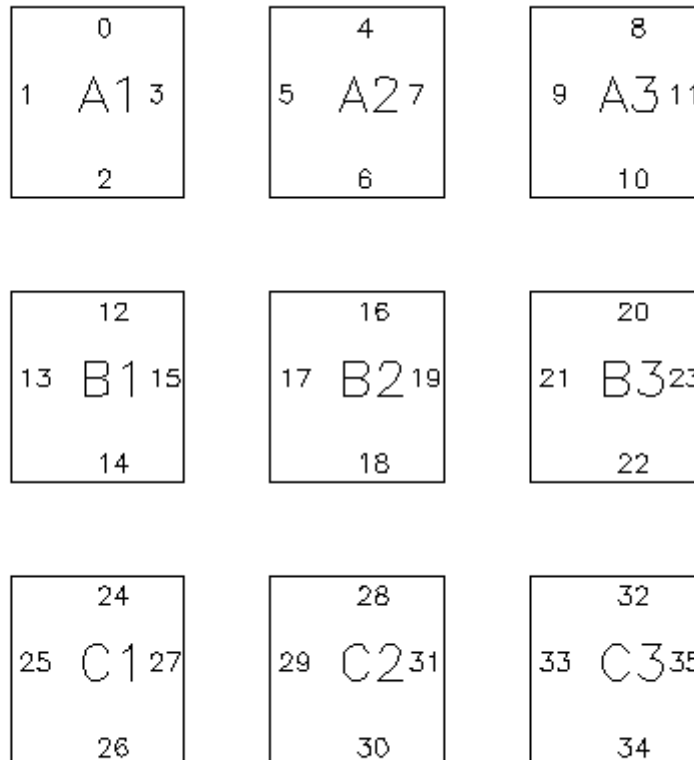
In all cases one makes less than 36 measurements, one may either redefine a new continuous numbering sequence (different from the one shown in figure), or make sure to increment manually the file index in a way compliant to the figure.

1. set up the roughness meter as described in 9.1.1
2. make sure the serial cable from the roughness meter is connected to serial port 1 of the spare computer
3. start the Taylor-Hobson file transfer program FTSP as described in 9.1.3
4. check the following parameters in the setup (only two of them must be reset, all the other shall be normally left alone)
  - NUMBER OF SERIAL PORT: 1
  - SELF-INCREMENT FOR FILE TALYPROFILE (self-increment of TalyProfile file number) : YES
  - CURRENT INDEX SELF-INCREMENT FOR FILE TALYPROFILE: must be reset to 0 before each run (otherwise it remembers the last one of the last run)
  - NAME OF FILE TALYPROFILE : insert a recognizable prefix xxxx before each run (e.g. linked to the date of the measurement)
  - PATH FOR FILE TALYPROFILE: the default is C:\Roughness\ (mind the final slash), could be pointed to a subdirectory specific of each measurement before start if desired.
5. on the roughness meter insert one sample in the vice: the suggested order is to start with the top side of sample A1, then turn the sample clockwise at each measurement, until you have done all four sides, then move to the next mask. This way the files associated with the sample measurements will be as shown in the following figure. Otherwise do only the top side of each sample (or at least use the top side values for quality assessment)
6. Position the measuring knife at the origin close to the left hand extremum of the sample, and lower the knife in position for the start of the measure (as described in 9.1.2)
7. Optionally in the FTSP Setup adjust the **CURRENT INDEX SELF-INCREMENT FOR FILE TALYPROFILE**, if you are not doing all 36 measurements.
8. In the FTSP program click on Measure & Analysis : the program will enter in wait mode
9. On the roughness meter start the measure pressing the main action button ; follow the measure on the Liquid Crystal Display (LCD). Once the measure is done, the data dump will automatically start
10. While the data dump is in progress (as shown by the LCD and by the **STATUS WINDOW** of FTSP), you can raise the measuring knife and prepare the sample for the next measurement (but not start it). Be careful not to touch the stylus, otherwise the data dump may abort.
11. When the dump is over, FTSP will show a rough plot and write the data to a binary file named `xxxxnn.pro`, where `xxxx` is the prefix you have chosen in step 4 above and `nn` is the measurement sequence number (0 to 35, see figure). FTSP will then start the Mountain analysis program and display a plot with the rough profile. It is important that you **do not interrupt** the process before this stage
12. At this stage you can analyse the `xxxxnn.pro` file immediately, or **EXIT** the Mountain program (and defer the analysis of all 36 files after the acquisition).
13. Once you have exited Mountains, you can proceed to step 6 for the next measurement. You may return the knife to the exact starting position semi-automatically using the procedure described in 9.1.2.

The analysis procedure within Mountains requires the following steps :

- Wait for the rough plot of the `xxxxnn.pro` file to appear (or load the file with **OPEN STUDIABLE** if running in deferred mode)
- In the **FILE** menu select APPLY TEMPLATE and choose template `template.mnt`
- this will replace the rough data plot with a detrended (linearized) plot of the rough data, and a smoothed plot resulting from the waviness analysis. In addition a panel will show a number of parameters resulting from the analysis, while another panel shows the file identification.
- Since this analysis can be redone from the original file at any time, it is not necessary to save this document (use **NEW DOCUMENT** to clear the page) once you have printed and filed it, or copied the relevant parameter values to a log file, according to the policy to be established by ESO.





The reference values for roughness parameters are reported in the specifications as quoted in the VIMOS PAE report [\[ref. 12\]](#) and the accompanying reference measurements [\[ref. 13\]](#)

You also have one slit sample, with 6 slits of different width. in the horizontal direction, and 6 in the vertical direction. The slit widths (0.2, 0.5, 1.0, 1.2, 1.5 and 2.0 mm) shall be verified comparing them with the provided reference sample. The tools for this comparison (microscope or profile projector) will be provided by ESO

## 7.10 Contingency (foreign) masks

The production and handling of masks for instruments other than VIMOS *and NIRMOS*, as well as the manufacturing of items not described above, is outside the scope of the present document. These operations can be undertaken if Gerber files are provided in input to CircuitCAM, manually converted into LMD files (which means either interactive usage of CircuitCam, or manual invocation of CircuitCam with a file argument, or drag-and-drop operation, see [6.4](#)), these files manually moved to MMCU, and supplied in input to a manual session of StencilMaster. No other support is within the scope of the present document.

In order to support the above functions the following Contingency directories are provided.

- D:\Staging\Contingency (on MHCU) is a repository for Gerber files supplied by the equivalent of MCS
- D:\Mhs\Contingency (on MHCU) is intended as work area for CircuitCam (move here files from the staging area in order to process them)
- E:\Contingency (on MMCU) is intended as work area dor StencilMaster (move here LMD files produced by CircuitCam)
- For the convenience of FORS2 the equivalent of the above directories (D:\Staging\Fors2, D:\Mhs\Fors2 and E:\Fors2) are provided separately.



The maintenance of files in the above directories, and the eventual arrangement in subdirectories is outside of the scope of the present document.

In the case the Gerber files are produced elsewhere (e.g. on another instrument IWS) they must be transferred into the D:\Staging\Contingency area on MHCU using the dedicated ftp Contingency account (with password contingencyftp), or respectively D:\Staging\Fors2 using the dedicated ftp Fors2 account (with password fors2ftp),.

## 7.11 Disk mirroring

The mirroring utility described in 6.3 will normally be running automatically without the need of operator intervention, and will ensure that the working disks of MHCU and MMCU are periodically cloned to the equivalent disk of the other machine. The times when the mirroring program runs are controlled by the `mirrorsetup.bat` file described in 6.3, which also describes how to verify correct running checking the log files, and how to run the mirroring utility manually.



## 8. Contingency Recovery Procedures

This section lists all conditions we are aware of which may occur during the standard procedures, then gives a comprehensive table of all popup messages generated by MHS and Cut Manager (there is no comprehensive list of the messages in the "NEXT ACTION" hint box because of their self-explanatory nature) and finally describes the usage of the recovery functions provided by MHS.

### 8.1 Conditions

This section lists all recognized contingency conditions, grouped according to the main procedural phases listed in section 7. For each condition we fill in a template like the following :

- **Condition** : verbose description of the condition.
- **Signalled by** : usually reference to a **POPUP** message, with cross reference to text listed in section 8.2.
- **Action**: explicit procedure to be undertaken by operator

#### 8.1.1 Contingencies during initialization (logon and switch on)

The following is a partial list of known or recognized contingency conditions during the phases without usage of the MHS or Cut Manager programs.

##### 8.1.1.1 contingencies during computer boot or logon

A full description of contingencies which may occur during boot or logon is outside of the scope of the present document. The following condition (related to our specific arrangement) is worth mentioning.

- **Condition** : cannot map Dclone or Eclone shares during logon as operator. Probably one computer is disconnected from the network.
- **Signalled by** : Win NT system message (popup)
- **Action**: Check that MMCU and MHCU are both connected to the network, and that their network interface is working (e.g. using ping from a DOS window).Once they are connected, log off and log on again.

##### 8.1.1.2 contingencies starting MHS or CutManager

- **Condition** : operator has pressed **EXIT** to exit *from the start form*.
- **Signalled by** : A **MHS POPUP** or a **CUT MANAGER POPUP**
- **Action**: Acknowledge to exit.

##### 8.1.1.3 hardware errors (LPKF rack)

The reference document in case of hardware errors on LPKF machine (which cannot be predicted *a priori*) is represented by the LPKF documentation (LPKF documentation [ref. 8]). We do not provide therefore an exhaustive list of error conditions. We note that such errors appear on the LCD display of the LPKF control rack (which also provides function keys on its front panel to view past errors : use the sequence **F1 SHOW F2 HISTORY ERROR** and then the arrow keys to scroll through the error list of 100 elements). We quote here only a couple of errors encountered by us a few times :

- **Condition** : water conductivity problem in the cooling circuit.
- **Signalled by** : A **LCD** message "Water conductivity too high"
- **Action**: This message may occur if the machine has been off for a long time (in this case just wait and retry a few times to switch on) or in the case the deionized water tank needs refilling.
  
- **Condition** : temperature problem.
- **Signalled by** : A **LCD** message "Temperature out of range"
- **Action**: This is a transient message, which will disappear in some time. Anyhow verify the chiller is on



#### 8.1.1.4 StencilMaster errors

The reference document in case of StencilMaster errors (which cannot be predicted *a priori*) is represented by the LPKF documentation ([LPKF documentation \[ref. 7\]](#)). We do not provide therefore an exhaustive list of error conditions. We only note that a series of messages can appear if one is using StencilMaster *without* the LPKF machine being switched on (this is a *legal* procedure to inspect LMD files and is described in [6.5](#)), and a "Cannot set the power" transient message sometimes occurs at the beginning of the cut.

We note also that, if one of the StencilMaster popups (like "Waiting to transmit") is interrupted pressing the relevant **STOP** button (and acknowledging the subsequent warning popup about the possibility of leaving the system in an unstable state) while the machine is actually on, it is then recommended to stop any cutting phase, exit StencilMaster and switch off and on the machine. Otherwise sooner or later an error message can appear on the rack LCD.

In such case one shall use the front panel to reset the error (**ERROR** and **RESET**) and follow the instructions on the LCD (note that the reset procedure will put the system in SYS OFF status !)

### 8.1.2 Contingencies during MHS convert phase

The following are the recognized contingency conditions during the conversion phase of the manufacturing cycle.

#### 8.1.2.1 no MMJ found in staging area or MMJ selection cancelled

This is actually a couple of separate conditions, both of which are flagged as "null runs" (this means they are not recorded in the log file, nor in the last action file, as nothing happened) :

- **Condition** : no MMJ is present in staging area.
- **Signalled by** : **CONVERT MESSAGE POPUP C02**
- **Action**: this is an informative message. No further action required. Click OK to acknowledge and the convert program will be terminated
  
- **Condition** : operator did not select any MMJ present in staging area.
- **Signalled by** : operator pressed **CANCEL** in the order selection form (described in [6.1.6](#))
- **Action**: No action. The convert program will be silently terminated

#### 8.1.2.2 more than one MMJ found in staging area

- **Condition** : more than one MMJ found in staging area
- **Signalled by** : not signalled, just see the standard order selection form (described in [6.1.6](#))
- **Action**: this is not an error ; default action is to select the first job in chronological order, however it is possible to select any job for processing

#### 8.1.2.3 MMJ waiting for store

- **Condition** : one MMJ in work area, already processed by convert, but waiting for store.
- **Signalled by** : **CONVERT MESSAGE POPUP C01**
- **Action**: error, convert not allowed to start (this is considered a null run as described in [8.1.2.1](#)). Click OK to acknowledge and exit, then complete the store phase

#### 8.1.2.4 crash during convert

If a computer, system or MHS crash occurs while running convert (or any other MHS phase), the checkpointing of data to disk inclusive of usage of flag files like `producecrash.txt` will allow program to restart from the correct point once situation is cleared. The same applies to the case the program is suspended by operator.

- **Condition** : one MMJ in work area, already partially processed by convert.
- **Signalled by** : informative message in "**NEXT ACTION**" hint box
- **Action**: standard recovery, `convert` will start with "resume from crashed session" then operator has to continue as instructed



#### 8.1.2.5 No MMJ or null MMJ in MMJ directory

This case includes two rare possibilities :

- **Condition** : MMJ directory exists and contains MSFs, but contains no MMJ order.
- **Signalled by** : **CONVERT MESSAGE POPUP C03**
- **Action**: presumably the MMJ is currently being transmitted by MCS. Click OK to acknowledge and terminate convert and retry again after a short time. If conditions persists for unreasonable long time, operator shall get in contact with MCS operator and check if network connection hung
  
- **Condition** : the MMJ order is null (i.e. refers to zero masks!).
- **Signalled by** : **CONVERT MESSAGE POPUP C04**
- **Action**: Click OK to acknowledge and generate a valid MMT (the order is formally fulfilled)

#### 8.1.2.6 Missing MSF

- **Condition** : no MSF found for mask id qnnnnn. Or any Visual Basic error while copying a MSF from staging to work area. The text of the VB error will be reported by the popup message text.
- **Signalled by** : **CONVERT MESSAGE POPUP C05**
- **Action**: error, click OK to acknowledge, requires no further action, MHS will flag appropriately and skip this mask

#### 8.1.2.7 duplicated MSF

- **Condition** : MSF found in staging area but one with same mask id already exists in work area.
- **Signalled by** : not signalled
- **Action**: error, this should not be possible, file will anyhow always be overwritten silently

#### 8.1.2.8 MSF to Gerber conversion error

- **Condition** : error during conversion of MSF to Gerber format (MSF corrupt or syntactically wrong : this might be trapped as a VB i/o error like a premature end of file, or by a *limited* analysis of the MSF syntax : for the details of this see program code, it covers however cases like an illegal quadrant identifier, a mismatch between the mask identifier and the quadrant, a missing slit coordinate).
- **Signalled by** : **CONVERT MESSAGE POPUP C06**
- **Action**: error, click OK to acknowledge, requires no further action, MHS will flag appropriately and skip this mask

#### 8.1.2.9 Gerber to LMD conversion error

- **Condition** : error during conversion of Gerber to LMD format (CircuitCam in interactive mode reports error or visual inspection error and operator has not saved the LMD file; no error checking is done using CircuitCam in default batch mode which might create invalid or empty LMD files)
- **Signalled by** : once operator has quit CircuitCam, **CONVERT MESSAGE POPUP C09**
- **Action**: error, operator, noticing this error, will exit CircuitCam without saving LMD ; convert will notice the absence of the LMD and present the above mentioned **POPUP**, offering a possibility to retry conversion (the same message can occur if the operator has forgotten to save the LMD file). If this possibility is renounced (replying **NO**) MHS will flag appropriately and skip this mask. Note that if CircuitCam in batch mode creates an invalid LMD file this cannot be noticed at this stage, but only by visual inspection with StencilMaster during manufacturing !

#### 8.1.2.10 less LMD files present than MSFs

- **Condition** : there are less LMD files present than the number of corresponding MSFs
- **Signalled by** : implicitly signalled by the LMD file generation popup (described in 6.1.6)
- **Action**: warning, this typically occurs when resuming from a crashed session. The **POPUP** will lead the operator through the correct procedure, i.e. if this was not due to one of the previous errors, repeat the Gerber to LMD conversion stage (will do missing ones). If any LMD will be skipped for valid reasons ( 8.1.2.9) this will ultimately lead to the condition described in 8.1.2.11.



### 8.1.2.11 *Incomplete set of LMD files*

- **Condition** : there are less than the number of LMD files requested in the order (usually 4) present for one mask set due to previous errors
- **Signalled by** : **CONVERT MESSAGE POPUP C13**
- **Action**: error, convert (routine MaskSetCopy called by Produce End) will not copy to MMCU any of the LMD for that mask set, and will delete any intermediate file, after operator acknowledgment. The fact will be duly logged. The manufacturing of this particular mask set will therefore be skipped in toto. If all mask sets are affected by this error, the condition described in 8.1.2.13 will be triggered, which, since MMJ orders contain exactly one mask set, means it will always be triggered.

### 8.1.2.12 *Errors inside MMJ*

This may include the following conditions :

- **Condition** : the order name in the MMJ header does not match the MMJ file name
- **Signalled by** : not signalled, the wrong header will be corrected automatically using the MMJ file name for any future purpose
- **Action**: no operator action required
  
- **Condition** : the number of masks in the MMJ header does not match the number of MSF
- **Signalled by** : not signalled directly, may trigger other errors
- **Action**: The effect on this may vary. If the number in the header is greater than the MSFs actually present on, one obtains an end-of-file message (C05) treated as in 8.1.2.6 (although the error is not on an MSF and the message make fictitious reference to an unnamed MSF; a spurious message C13 may be generated at the end of convert). If the number is non zero and less than the MSFs actually present, the extra MSFs are not copied and ignored. This is also likely to generate a condition of incomplete mask set (as in 8.1.2.11). If the number is zero but there are MSFs present, the order is handled as a null MMJ (see 8.1.2.5) and the MSFs are ignored and deleted

### 8.1.2.13 *no LMDs at all present at end of conversion*

This may occur as a consequence of several errors of the previous kinds and give rise to one of the following conditions. In all cases an MMT report will be automatically sent (no manufacturing or storage is possible, the processing of the MMJ order terminates at this stage):

- **Condition** : no MSF was found
- **Signalled by** : **CONVERT MESSAGE POPUP C14**
- **Action**: convert is aborted before the translation-to-Gerber phase, after operator acknowledgment
  
- **Condition** : no Gerber file was generated
- **Signalled by** : **CONVERT MESSAGE POPUP C07**
- **Action**: convert is aborted before invoking CircuitCam, after operator acknowledgment.
  
- **Condition** : no LMD file was generated nor copied to MMCU
- **Signalled by** : **CONVERT MESSAGE POPUP C15**
- **Action**: convert is terminated after operator acknowledgment. This is considered a normal termination, however an MMT report is sent at this stage.

## 8.1.3 *Contingencies during mask manufacturing phase*

The following are the recognized contingency conditions during the manufacturing phase of the manufacturing cycle.

### 8.1.3.1 *mask manufacturing termination*

- **Condition** : mask manufactured normally (masks 1-3 of a mask set)
- **Signalled by** : operator has to keep track of which masks have been manufactured using the quadrant and sequence numbers
- **Action**: not an error, normal condition, operator shall store the mask to the intermediate repository and proceed manufacturing the next mask



### 8.1.3.2 mask set manufacturing termination

- **Condition** : mask manufactured normally (last of a mask set)
- **Signalled by** : operator has to keep track of which masks have been manufactured using the quadrant and sequence numbers
- **Action**: not an error, normal condition, operator shall store any mask of the set not yet stored to the SC

### 8.1.3.3 mask dropped after manufacturing

- **Condition** : just manufactured mask dropped while storing to intermediate repository
- **Signalled by** : no signal given to operator, it' s the operator which has to tell Cut Manager that the mask is being remanufactured using the **REMANUFACTURE** button
- **Action**: error, operator shall reproduce (manufacture again) mask immediately (or asap if manufacturing of the next one is already in progress)

### 8.1.3.4 laser machine failure before start of a mask set

- **Condition** : cutting machine requires routine intervention or has recoverable failure before all masks of a set are produced. A recoverable problem is one which can be fixed on the spot in a reasonably short time.
- **Signalled by** : depends
- **Action**: warning, operator shall delay production and storage until problem fixed

### 8.1.3.5 laser machine irrecoverable failure before start of a mask set

- **Condition** cutting machine has irrecoverable failure before all masks of a set are produced. An irrecoverable failure requires external intervention, replacement on unavailable spare parts, or other action which cannot be concluded within the current day shift.
- **Signalled by** : depends
- **Action**: error, operator shall go to MHCU running the store program and terminate it (this issues a termination report to MCS). store will remove all files of the forthcoming mask set (and of any other undone mask set) from MHCU and MMCU (inclusive of the LMD files in the MMCU Incoming area)

### 8.1.3.6 laser machine irrecoverable failure during production of a mask set

- **Condition** cutting machine has irrecoverable failure before production of a new mask set is started. See previous section for definition of an irrecoverable failure.
- **Signalled by** : depends
- **Action**: error, operator shall throw away the masks of the current mask set still in intermediate repository then go to MHCU running the store program and terminate it (this issues a termination report to MCS). store will remove all files of the current unfinished mask set (and any other undone mask set) from MHCU and MMCU (inclusive of the LMD files in the MMCU Incoming area). *However the LMD files of any mask which was pending for manufacture will not be removed until the condition described in 8.1.3.8 will be triggered: it is suggested for this to exit and restart Cut Manager and reply CANCEL to the relevant POPUP as instructed in 8.1.3.8*

### 8.1.3.7 hardware errors

The reference document in case of hardware errors on LPKF machine (which cannot be predicted *a priori*) is represented by the LPKF documentation (LPKF documentation [ref. 8]). We do not provide therefore an exhaustive list of error conditions. We note only two possible actions.

- In case of problems occurring during a cut, like excessive sparks from the cut area, Invar shavings obstructing the cut and not removed by the exhauster, excessive vibration of the air pad support, machine head hitting the end stops, the first course of action shall be to **STOP** the current cutting phase using the appropriate StencilMaster button.
- If this is not sufficient and an *immediate* stop is desired, the emergency stop button on the LPKF rack must be pressed, then one must exit StencilMaster and restart (do not forget to pull out the emergency stop button !)
- In case of hit of the end stops (end stage error) the machine will in general attempt autonomously to reconstruct its reference system doing a slow scan. If it is desired to make a recalibration, refer to the procedure described in **9.2**



- In particular we notice that a spurious hit of the end stops can occur, even if the machine is not moving at all and with the head located in the centre, if the pressure of the air pads decreases below the safe value (see digital manometer on the back of the granite stand). In this particular case the machine cannot move anymore. It is necessary to exit StencilMaster and turn it off. When switched on, it will slowly move to the reference position and "bump" into it to reconstruct its reference. It is *strongly advised* to re-make a calibration at this time.

#### 8.1.3.8 LMD file already in working area

- **Condition** :Cut Manager starts and finds an LMD file in its current directory (presumably left by a crash)
- **Signalled by** : **CUT MANAGER** message POPUP
- **Action**: the popup will offer the possibility of starting manufacturing the LMD file (*you should select this option if the mask has not been manufactured at all*, pressing **RETRY**) or to press **CANCEL** to remove it and perform a normal start (*you should select this option only if the mask cannot be made at all because of an irrecoverable LPKF machine fault*). *In the case the mask had been manufactured, but one did not press MASK DONE (to save the LMD file in the Archive folder) the correct procedure is to press RETRY, do not do anything in StencilMaster, deiconize Cut Manager, and press MASK DONE*

### 8.1.4 Contingencies during storage phase

The following are the recognized contingency conditions during the storage phase of the manufacturing cycle.

#### 8.1.4.1 timeout during connection to bar code reader

- **Condition** : MHS cannot establish connection with the SC bar code reader.
- **Signalled by** : **BAR CODE POPUP** B01
- **Action**: the popup will ask operator to check barcode reader is connected to port (check serial cable at both ends) and switched on (check power cable plugged in and LED on SC barcode reader). Fasten any loose cable and, if everything looks OK, press **RETRY**. In case of repeated failures or clear hardware failure press **CANCEL** to exit program, and refer to Maintenance Manual section 12.5 (bar code or serial card failure).

#### 8.1.4.2 mask dropped before identification

- **Condition** : mask dropped before identification (while moving from MMU to intermediate storage, see 8.1.3.3, or from intermediate storage to bar code reader)
- **Signalled by** : operator will notice
- **Action**: error, if operator has already clicked **NEXT MASK**, he shall break the operation pressing the **SUSPEND** button. Then operator shall reproduce (manufacture again) mask asap (and then resubmit to store)

#### 8.1.4.3 mask not belonging to MMJ

- **Condition** : bar code reader found a mask id which is invalid or not present in MMJ
- **Signalled by** : **STORE MESSAGE POPUP** S04 (invalid bar code, *i.e. not in the VIMOS range*) or S05 (valid bar code, but does not belong to current order)
- **Action**: error, operator shall acknowledge, put mask aside, and skip rest of store operation, then proceed to another mask

#### 8.1.4.4 mask listed as already stored

This may happen in three cases.

- **Condition** : bar code reader found a mask id for a mask already listed in SCT and seemingly stored in this session (eventually verify mask is physically there in SC)
- **Signalled by** : **STORE MESSAGE POPUP** S06
- **Action** error, this is seemingly a duplicated mask, if you just forgot to store the previous mask, do it now, otherwise if you really produced another mask, throw duplicated mask away, and proceed to next one





- **Condition** : bar code reader found a mask id for a mask already listed in SCT (but mask is not physically there !)
- **Signalled by** : not signalled (except as by manual verification in consequence of the previous error).
- **Action** unlikely to occur, current procedure does not foresee this verification, handle as previous case (no need to run the recovery Rescan SC function)
  
- **Condition** : bar code reader found a mask id for a mask already listed in SCT and seemingly stored *prior* to this session (must be verified that mask is physically there in SC)
- **Signalled by** : **STORE MESSAGE POPUP S07**
- **Action** : the acknowledgment will terminate (abort) store. Operator must understand the cause of this situation before deciding if :
  1. there is no such mask in SC and the current one can be stored (eventually using the recovery insert function see 6.1.10.3);
  2. there is an old (wrong) mask in SC which must be discarded (using the recovery remove function see 6.1.10.4) before storing the current one ;
  3. there is an irrecoverable mismatch between SCT and SC content, which requires to run the recovery rescan function see 6.1.10.1)

#### 8.1.4.5 No room in storage cabinet

This may happen in a few rare cases (it should be prevented by controls at MCS level) :

- **Condition** : SCT says there is no room in SC (and it is true)
- **Signalled by** : not signalled directly. Operator may notice it in advance since the screen will show 100 masks present in a SC. Operator will definitely notice when storing, since there is no room he cannot successfully store.
- **Action** : error, operator shall stop everything (acting as described in 8.1.4.6, and specifically replying **NO** to both popups S08 and S09 , then suspend or terminate store) and put mask aside, then get in contact with MCS operator to sort out the situation.
  
- **Condition** : SCT says there is no room in SC (and it is *not* true)
- **Signalled by** : not signalled directly. Operator may notice it in advance since the screen will show 100 masks present in a SC, but may also not notice and insert the mask in a free slot.
- **Action** : error, if operator notices in advance he shall stop everything (and act as described in the previous condition). Otherwise if a mask is inserted in the SC, but the SCT is wrong (incorrectly 100% full), the mask will not be recorded in the SCT. The program continues without problems, but a mismatch situation requiring a recovery rescan procedure (see 6.1.10.1) is likely to arise some time later.
  
- **Condition** : SCT vs SC mismatch (SCT says there is room, but physically there is no room)
- **Signalled by** : operator will notice when attempting to store
- **Action** : error, operator shall stop everything (i.e. acting as described in 8.1.4.6, and specifically replying **NO** to both popups S08 and S09 , then suspend or terminate store) and run recovery rescan procedure (see 6.1.10.1).

#### 8.1.4.6 mask dropped after identification

- **Condition** : mask dropped after identification but before storage (while moving from bar code reader to SC)
- **Signalled by** : after operator has replied **NO** to the **STORE MESSAGE POPUP S08** will see another **STORE MESSAGE POPUP S09**
- **Action** : error, operator shall reproduce mask asap and then resubmit to store. He has the option of replying **NO** to skip the current mask and continue storing (while remanufacturing the mask again some time later) or to reply **YES** to put store in standby waiting for the mask to be remanufactured immediately.

#### 8.1.4.7 mask successfully stored

- **Condition** : mask successfully stored
- **Signalled by** : operator replies **YES** to the **STORE MESSAGE POPUP S08**
- **Action** : this is normal, proceed to next mask (MHS will not normally know about mask sets but usage of intermediate repository will ensure only entire mask sets are processed).



#### **8.1.4.8 MHS required for other tasks**

- **Condition** : MHS required for other tasks
- **Signalled by** : oral message
- **Action** : operator shall suspend store, perform other tasks (IC preparation or discard) and resume store

#### **8.1.4.9 crash during store**

If a computer, system or MHS crash occurs while running store (or any other MHS phase), the checkpointing of data to disk inclusive of usage of flag files like `storecrash.txt` and `temp_rep.txt` will allow program to restart once situation is cleared

#### **8.1.4.10 normal termination**

- **Condition** : storage terminated normally by operator pressing **END** (all masks of all mask sets)
- **Signalled by** **STORE MESSAGE POPUP S12**
- **Action**: a reply of OK will send MMT report and terminate

#### **8.1.4.11 premature termination before initialization**

- **Condition** : processing terminated by operator pressing **END** before connecting to bar code reader and retrieving job related data (for the first time in a store session)
- **Signalled by**: **STORE MESSAGE POPUP S12**
- **Action**: null run, no action (will *not* send MMT report)

#### **8.1.4.12 premature termination just after initialization**

- **Condition** processing terminated explicitly by operator pressing **END** when no mask have been stored yet and cannot be stored because of an irrecoverable cutting machine failure. This covers also an end before connecting to bar code reader if store is resuming from a suspended session.
- **Signalled by**: external condition
- **Action**: handled as case [8.1.4.14](#) below (will send an MMT report but requires no recovery))

#### **8.1.4.13 premature termination with incomplete mask set stored**

- **Condition** processing terminated by operator, last mask set stored is incomplete because of an irrecoverable cutting machine failure
- **Signalled by**: external condition, then after operator pressed **END** via **STORE MESSAGE POPUP S10** and **S11**
- **Action**: error, anyhow send MMT; files referring to undone masks will be removed by MHS. Operator has then to run recovery remove procedure (see [6.1.10.4](#)) to take masks of the incomplete set out the SC

#### **8.1.4.14 premature termination with entire mask sets missing**

- **Condition** processing terminated by operator, There are entire mask sets missing because of an irrecoverable cutting machine failure
- **Signalled by**: external condition, then after operator pressed **END** via **STORE MESSAGE POPUP S10** and **S11**
- **Action**: error, anyhow send MMT; files referring to undone masks will be removed by MHS. No further action required

Note that in this case (as well in the one described in [8.1.4.13](#)) the files referring to undone masks are the working copies residing on MHCUC. The LMD files stored on MMCUC will not be deleted : they can be deleted only by the discard function.

#### **8.1.4.15 no masks to store**

- **Condition** initialization found no masks to store (convert was not run)
- **Signalled by**: **STORE MESSAGE POPUP S03**
- **Action**: acknowledge to force termination (of course without MMT report)



#### 8.1.4.16 *convert pending on current MMJ*

- **Condition** store cannot start because convert has not yet finished processing the current MMJ
- **Signalled by**: **STORE MESSAGE POPUP S01**
- **Action**: acknowledge to terminate as a null run

### 8.1.5 Contingencies during IC preparation phase

The following are the recognized contingency conditions during the IC preparation cycle.

#### 8.1.5.1 *timeout during connection to bar code reader*

- **Condition** : MHS cannot establish connection with the SC or IC bar code reader.
- **Signalled by** : **BAR CODE MESSAGE POPUP B01 or B02**
- **Action**: the popup will ask operator to check one of the two barcode readers is connected to port (check serial cable at both ends) and switched on (check power cable plugged in and LED on SC or IC barcode reader). Fasten any loose cable and, if everything looks OK, press RETRY. In case of repeated failures or clear hardware failure press **CANCEL** to exit program, and refer to Maintenance Manual section 12.4 or 12.5 (bar code or serial card failure).

#### 8.1.5.2 *timeout during connection to MHS robot*

- **Condition** : MHS cannot establish connection with the MHS robot controller.
- **Signalled by** : **ROBOT MESSAGE POPUP R01**
- **Action**: the popup will ask operator to check the robot controller is connected to port (check serial cable at both ends) and switched on (check safety lamp labelled 220V is on and not burnt, and switch turned on) and to check the serial cable connector adapter (check LED is on). Fasten any loose cable and, if everything looks OK, press RETRY. In case of repeated failures or clear hardware failure press **CANCEL** to exit program, and consider it a robot failure.

#### 8.1.5.3 *robot is homing*

- **Condition** : The MHS robot Z-displacement unit is moving to the reference position (home) .
- **Signalled by** : **ROBOT MESSAGE POPUP R10**
- **Action**: the Z-displacement unit need to set its home position as soon after it has been switched on. If the program (via a status read) detects home has not yet been set, will do so, after the operator has pressed **OK** for acknowledgment. This is not an error, but part of routine initialization.

#### 8.1.5.4 *robot error while homing*

The MHS robot Z-displacement unit is commanded to move to the reference position (home) and the command is either not acknowledged or a status request gives an error . This can occur in four different conditions :

- **Condition 1**: The home command is either not acknowledged or the serial connection times out
- **Signalled by** : **ROBOT MESSAGE POPUP R02 to R06**. The robot status text area will contain a specific indication of the error (a command timeout error, or a status timeout error, or a command syntax error). See Maintenance Manual section 11.1.1.11 and in particular 11.1.1.20 for details on robot errors.
- **Action**: only option is **CANCEL**. This aborts the MHS IC preparation as described in [8.1.5.26](#) (o immediately in the case of IC rescan). You can attempt to resume it after checking the connection as described in [8.1.5.2](#). Timeout errors probably indicate a communication problem (loose cable or power) occurred after the initial test. A syntax error (corresponding to a NAK sent by the controller in response to a command) may indicate robot controller trouble, or a software error. Try to cycle robot power off and on, or to stop and restart MHS.
- **Condition 2** : Recoverable error (typically corresponding to robot error codes 4 and 5) : either the photo cell signals an obstruction on the path of the Z-displacement unit, or the emergency stop button has been pushed. More rarely it may correspond to the robot being busy (robot error code 3) when not expected, or to two commands sent at same time (robot error code 9)
- **Signalled by** : **ROBOT MESSAGE POPUP R07** or **R11** The robot status text area will contain a specific indication of the error



- **Action:** options are **RETRY** or **CANCEL**. The latter sends a controller reset command, then aborts the MHS IC preparation as described in [8.1.5.26](#). The former request you to remove the error condition (push masks in the IC or in the mask stand so they do not obstruct the photo cell, or remove other objects ; or respectively pull the emergency stop button out) *then press* **RETRY**. This will send a reset command and attempt to retry the homing operation. The reset command is anyhow unnecessary and not sent in case of robot error code 3.
- **Condition 3 :** Recoverable error : home position lost (the no-home status, robot error code 1, remains after sending the home command).
- **Signalled by :** **ROBOT MESSAGE POPUP R08** The robot status text area will also contain indication of the error
- **Action:** options are **RETRY** or **CANCEL**. The latter aborts the MHS IC preparation as described in [8.1.5.26](#). The former request you to attempt to remove the error condition (check the proximity switch is not obstructed and operating) *then press* **RETRY**. This will retry directly the homing operation.
- **Condition 4:** Unrecoverable error : an hardware error (robot error codes 6-10) has been detected.
- **Signalled by :** **ROBOT MESSAGE POPUP R09** The robot status text area will also contain indication of the error
- **Action:** only option is **CANCEL**. This aborts the MHS IC preparation as described in [8.1.5.26](#) You can then verify the functioning of the Z-displacement unit (eventually cycling power off and on) and resume the IC preparation

### 8.1.5.5 robot error while initializing

The MHS robot Z-displacement unit is commanded to set the acceleration and speed (which is done at the beginning) and the command is either not acknowledged or a status request gives an error . This can occur in three cases :

- **Condition :** The command is either not acknowledged or the serial connection times out
- **Signalled by :** **ROBOT MESSAGE POPUP R02** See [8.1.5.4 condition 1](#) for details.
- **Action:** See actions for condition 1 in [8.1.5.4](#).
- **Condition :** Recoverable error (typically robot error codes 4 and 5) : either the photo cell signals an obstruction on the path of the Z-displacement unit, or the emergency stop button has been pushed.
- **Signalled by :** **ROBOT MESSAGE POPUP R07** or **R11** See [8.1.5.4 condition 2](#) for details. . Note that error code 1 ([8.1.5.4 condition 3](#)) is not handled as an error since the home setting is independent on the speed/acceleration settings
- **Action:** options are **RETRY** or **CANCEL** as for [8.1.5.4 condition 2](#). The latter aborts the MHS IC preparation as described in [8.1.5.26](#). The former request you to remove the error condition (push masks in the IC or in the mask stand so they do not obstruct the photo cell, or remove other objects ; or respectively pull the emergency stop button out) *then press* **RETRY**. This will send a reset command and attempt to retry the set speed or acceleration operation
- **Condition :** Unrecoverable error : an hardware error (robot error codes 6-10) has been detected.
- **Signalled by :** **ROBOT MESSAGE POPUP R09** See [8.1.5.4 condition 4](#) for details.
- **Action:** See actions for condition 4 in [8.1.5.4](#).

### 8.1.5.6 robot needs to be loaded

- **Condition :** The MHS robot Z-displacement unit must be loaded with the IC holder, and therefore has to move to the load position. Once the position is reached it requires operator intervention.
- **Signalled by :** **UNLOAD/LOAD MESSAGE POPUP U04** or **L01** or **Ic01**.
- **Action:** This is not an error but part of standard operations. The operator shall reply **YES** to confirm whether the IC holder is already loaded or **NO** if a load is requested. If the holder has to be loaded, the operator has to acknowledge another **POPUP U05** or **L02** or **Ic02** and to wait for the Z-displacement unit to reach the load (lowest) position, and has to physically load the IC holder onto the platform, until the notches in the holder bottom are secured to three balls on the platform pins. Once this is done press **OK** in the confirmation **POPUP U06** or **L03** or **Ic03** to continue.. The program will automatically command a new home (with the platform loaded) before continuing as in [8.1.5.3](#).



### 8.1.5.7 robot error while moving

The MHS robot Z-displacement unit is commanded to move to any position and the command is either not acknowledged or a status request gives an error . This can occur in four cases :

- **Condition** : The move command is either not acknowledged or the serial connection times out
- **Signalled by** : **ROBOT MESSAGE POPUP R02** See [8.1.5.4 condition 1](#) for details.
- **Action**: See actions for condition 1 in See [8.1.5.4](#).
  
- **Condition** : Recoverable error (robot error codes 4 and 5) : either the photo cell signals an obstruction on the path of the Z-displacement unit, or the emergency stop button has been pushed.
- **Signalled by** : **ROBOT MESSAGE POPUP R07** or **R11** See [8.1.5.4 condition 2](#) for details.
- **Action**: options are **RETRY** or **CANCEL** as for [8.1.5.4 condition 2](#). The latter aborts the MHS IC preparation as described in [8.1.5.26](#). The former request you to remove the error condition (push masks in the IC or in the mask stand so they do not obstruct the photo cell, or remove other objects ; or respectively pull the emergency stop button out) *then* press **RETRY**. This will send a reset command and attempt to retry the movement
  
- **Condition** : Recoverable error : home position lost (robot error code 1).
- **Signalled by** : **ROBOT MESSAGE POPUP R08** See [8.1.5.4 condition 3](#) for details.
- **Action**: options are **RETRY** or **CANCEL**. The latter aborts the MHS IC preparation as described in [8.1.5.26](#). The former request you to attempt to remove the error condition (check the proximity switch is not obstructed and operating) *then* press **RETRY**. This will perform a new homing operation then attempt to retry the movement.
  
- **Condition** : Unrecoverable error : an hardware error (robot error codes 6-10) has been detected.
- **Signalled by** : **UNLOAD/LOAD MESSAGE POPUP R09** See [8.1.5.4 condition 4](#) for details
- **Action**: See actions for condition 4 in [8.1.5.4](#).

### 8.1.5.8 robot moving

- **Condition** : The MHS robot Z-displacement unit is commanded to move to any position. Movement takes some time (during which a robot status code 3 is generated).
- **Signalled by** : messages in the **NEXT ACTION** area and in the robot status text area.
- **Action**: This is standard operation. Operator shall wait until a movement completed (machine ready) message appears in the robot status area.

### 8.1.5.9 more than one MIJ present in staging area

- **Condition** : more than one MIJ is present when only one is normally expected (at start of unload phase).
- **Signalled by** : not signalled, just see the standard order selection form (described in [6.1.8.1](#))
- **Action**: this is an anomaly, normally only one MIJ (plus the null order always present) must be present, the form allows to select one of many, but we cannot give here a rule about which will be the correct one : direct inquiry with the MCS operator is likely to be needed to select the correct priority.

Note that the null order is always there, hence is not signalled. If there is no other order, and, as typical you do *not* wish to empty completely the IC executing the null order, press **CANCEL** to exit the unload program.

### 8.1.5.10 no MIJ found in staging area

- **Condition** : no MIJ is present in staging area.
- **Signalled by** : **UNLOAD MESSAGE POPUP U02**
- **Action**: this is an informative message. No further action required. Click **OK** to acknowledge and the unload program will be terminated. In practice this will never happen, since the null order is always there (see [8.1.5.9](#))



#### **8.1.5.11 mask already in IC**

- **Condition** : a mask already in IC is requested to stay there in the MIJ.
- **Signalled by** : this is not signalled directly by the unload program (but a text area will show the complementary list of masks to be unloaded, and the masks in the order already in the IC will be flagged with an `ok` in the order listing text area). This will also not be signalled directly by the load program (which however will mark with an `x` in the order listing text area the masks already in IC)
- **Action**: No further action required

#### **8.1.5.12 mask cannot be unloaded**

This may occur in three conditions when attempting to unload a mask from an IC slot into the mask stand, but this is not possible for the following reasons.

- **Condition** : The slot is empty (when a mask was expected) because telescope staff has removed it
- **Signalled by** : `UNLOAD MESSAGE POPUP U08` after operator reply of `NO` to the `UNLOAD MESSAGE POPUP U07`
- **Action**: Reply `YES` to the `POPUP`. The mask will be deleted from the ICT and not recorded into the SCT and the IC preparation continues. If the mask is required again in the future, it has to be reproduced from the archive (with the same priority as for the case in [8.1.5.15](#)).
- **Condition** : The slot is empty (when a mask was expected) for an unknown reasons
- **Signalled by** : `UNLOAD MESSAGE POPUP U08` after operator reply of `NO` to the `UNLOAD MESSAGE POPUP U07`
- **Action**: Reply `NO` to the `POPUP`. Soon afterwards the unload program is suspended. The case must be handled as a mask id mismatch (see [8.1.5.14](#)), i.e. running a `recovery rescanIC` procedure (see [6.1.10.2](#)) and then resuming IC preparation to retry.
- **Condition** : The robot stopped in an intermediate position and the mask clamp is not aligned with an IC slot
- **Signalled by** : `UNLOAD MESSAGE POPUP U08` after operator reply of `NO` to the `UNLOAD MESSAGE POPUP U07`
- **Action**: Reply `NO` to the `POPUP`. Soon afterwards the unload program is suspended. After IC preparation has been suspended, `cycle` robot power off and on, and `resume` IC preparation. If problem persists declare a robot failure.

#### **8.1.5.13 mask unloaded successfully**

- **Condition** : mask moved from IC to SC
- **Signalled by** : An operator reply of `YES` to the `UNLOAD MESSAGE POPUP U07`
- **Action**: this is part of the routine operations

#### **8.1.5.14 mask id mismatch**

- **Condition** : The robot bar code reader found an id different from the one expected for that IC slot on the basis of ICT (this covers also the case mentioned above of no mask present in IC where one expected from ICT).
- **Signalled by** : `UNLOAD MESSAGE POPUP U09`
- **Action**: error, operator: `re-insert` mask in IC; after an acknowledge the unload program will automatically be suspended. `Run recovery rescanIC` procedure (see [6.1.10.2](#)) eventually try first to `cycle` robot power off and on, and `resume` IC preparation to force homing the robot and retry.

#### **8.1.5.15 mask dropped while moving from IC to SC**

- **Condition** : mask dropped while moving from IC to SC.
- **Signalled by** : An operator reply of `NO` to the `UNLOAD MESSAGE POPUP U11`
- **Action**: error, operator: `reproduce` asap when unload/load finished then `store` it using `recovery insert` procedure (see [6.1.10.3](#)). In the meanwhile `continue` with unloading, after acknowledging the next `POPUP U12`. Note the mask will not be listed neither in ICT nor in SCT until re-manufactured. Note that this is not a sever error: the masks to be unloaded are not required in the IC for the next observing nights (and possibly will never be required again), hence the IC preparation has priority on the mask reproduction.



#### 8.1.5.16 **no room in SC**

- **Condition** : there is no room in SC for moving a mask from IC to SC.
- **Signalled by** : this is not signalled directly, operator may notice it in advance since the screen will show 100 masks present in a SC. Operator will definitely notice when storing, since there is no room he cannot successfully store
- **Action**: error, operator : if store is not possible (SC physically full), try to put the mask aside, reply **NO** to the **UNLOAD MESSAGE POPUP U11** as in 8.1.5.15 and defer storage at the end of IC preparation, using recovery insert procedure (see 6.1.10.3), when it is likely that more space will exist. Report the fact to the IWS operator (this is likely to be an anomaly in MCS). If instead store is possible (there is room in the SC, but the SCT incorrectly says there are 100 masks stored, the the mask will not be recorded in the SCT. The program continues without problems, but a mismatch situation requiring a recovery rescan procedure (see 6.1.10.1) is likely to arise some time later.

#### 8.1.5.17 **MIJ empty or requires no loading**

- **Condition** : MIJ empty (MIJ not present in load phase is impossible), or anyhow MIJ does not require to load any additional mask.
- **Signalled by** : **UNLOAD MESSAGE POPUP U17** will report that the load phase is unnecessary
- **Action**: info, acknowledge then no further action (an empty MIJ can be used to unload all masks from ICs). Note no MIT is generated in case of *autonomous* null order (the one called `null.mij`), while it is generated for any order coming from MCS and completed just after the unload phase.

The above mentioned **POPUP U17** appears after the robot has moved the platform to the unload position (see 8.1.5.25) and the first **POPUP U15** has been shown.

Note that if at the end of the unload phase there are (as usual) masks to be loaded, operator will see first **POPUP U15** then **POPUP U18**

#### 8.1.5.18 **mask not found in SCT**

- **Condition** : required mask not found in SCT (at load time).
- **Signalled by** : pressing **NOT FOUND** button while searching (currently not signalled by program)
- **Action**: error, omit insertion and skip to next mask. currently this is not handled differently from the next case (i.e. screen there is no preliminary check in SCT to prevent search), the operator notices the difference just looking at the SCT on the screen.

#### 8.1.5.19 **mask physically not found in SC**

- **Condition** : required mask physically not found in SC (apparent SCT vs SC mismatch)
- **Signalled by** : pressing **NOT FOUND** button while searching, generates **LOAD MESSAGE POPUP L10**
- **Action**: error, operator: stop SC search mode loop pressing **NOT FOUND** button, verify simple problems (masks well pushed into SC, etc.) and in response to **POPUP L10** either press RETRY to search again (eventually in other quadrants), otherwise press CANCEL to skip loading of this mask. If problem persists (on more masks) press END to stop everything (abort loading) and run recovery rescan procedure (see 6.1.10.1)

#### 8.1.5.20 **mask dropped while moving from SC to bar code reader**

- **Condition** : mask dropped while moving from SC to bar code reader
- **Signalled by** : reply **NO** to **LOAD MESSAGE POPUP L05**
- **Action**: error, operator : if there is time enough for immediate mask reproduction, reply **NO** to next popup **L06** to defer loading (this loops back showing again **POPUP L05**) , reproduce mask, then put it immediately into IC and proceed (at this time SCT will be updated) If immediate reproduction is not possible, reply **YES** to next popup to declare loading of this particular mask as unsuccessful and continue (at this time mask will be removed from SCT). When possible reproduce mask and use recovery insert procedure (see 6.1.10.3) to store it. Note that this *is a critical condition* : if a mask is not loaded in the IC, the observation will *not* be possible, therefore one must do any effort to reproduce damaged masks asap.



#### **8.1.5.21 unexpected mask id found by robot bar code reader**

- **Condition** : mismatch in mask id : unexpected mask id found by robot bar code reader
- **Signalled by** : **LOAD MESSAGE POPUP L07**
- **Action**: warning, probably wrong mask extracted from SC : put mask back in SC and search again (s/w assisted : follow instructions in "NEXT ACTION" hint box)

#### **8.1.5.22 mask cannot be loaded**

- **Condition** : The program asks to load a mask into IC from the mask stand, but this is not possible.
- **Signalled by** : **LOAD MESSAGE POPUP L09** as a consequence of a reply **NO** to **POPUP L08**
- **Action**: This may occur either because the IC slot is already full with a mask (when instead is expected to be empty) or because the robot stopped in an intermediate position and the mask clamp is not aligned with an IC slot. Both cases abort IC preparation. The first case is a mask id mismatch and requires to run the `recovery rescan IC` function, then resume IC preparation, while the second shall be handled as for condition 3 of **8.1.5.12**

#### **8.1.5.23 mask damaged moving from mask stand to IC slot**

- **Condition** : mask damaged moving from mask stand to IC slot
- **Signalled by** : this is not signalled by software
- **Action**: a damage while inserting is unlikely and might occur as a consequence of errors in **8.1.5.22**. Reproduction of the damaged mask shall be handled similarly to what described in **8.1.5.20** i.e. while program is aborted the operator may reproduce the mask, use `recovery insert` to store it in SC, and resume IC preparation.

#### **8.1.5.24 no room in IC**

- **Condition** : no room in IC
- **Signalled by** : not signalled directly by software, however if the ICT indicates the IC is full, all further masks in the MIJ are automatically and silently skipped.
- **Action**: error, should not be possible because of previous checks at MCS level, but anyhow requires no action : a MIT report telling that the masks were not loaded will be sent at the end.

#### **8.1.5.25 unloading IC holder**

- **Condition** : The MHS robot Z-displacement unit shall move to the unload (lowest) position to allow the operator to take out the IC holder. .
- **Signalled by** : **UNLOAD/LOAD MESSAGE POPUP U13** and **L04** or **Ic04**
- **Action**: This is not an error but part of standard operations, such movement is automatically done as final operation of the Load function (or Unload when load is not necessary), or of the IC Rescan function. **Press OK** to acknowledge, and then remove the IC holder from the platform once this has reached the unload position.

#### **8.1.5.26 handling abort conditions**

- **Condition** : A robot recoverable error has been solved pressing the **CANCEL** (abort) button, or a robot unrecoverable error has been acknowledged causing an abort. .
- **Signalled by** : one of above errors **8.1.5.4**, **8.1.5.5**, or **8.1.5.7** via **UNLOAD/LOAD MESSAGE POPUP U16** and **L13**  
Note that in case of IC Rescan, this is aborted unconditionally.
- **Action**:. the **POPUP** will propose a termination request with the idea to either **RETRY** IC preparation (in which case it will be actually suspended, then the operator has to try to clear the error condition, and if successful resume IC preparation) or to immediately **CANCEL** definitively (abort) IC preparation and generate a MIT report.

#### **8.1.5.27 unloaded mask already recorded in SCT**

- **Condition** : a mask unloaded from IC is recorded in SCT as already stored in SC.
- **Signalled by** : **UNLOAD MESSAGE POPUP U10**
- **Action**: error, operator : put mask back in IC and press **OK** then suspend unload, run recovery remove to remove the duplicated mask (or eventually `recovery rescan SC` to clear a spurious duplication), and resume IC preparation





### 8.1.5.28 *masks cannot be located in ICT and SCT*

- **Condition** : one or more masks listed in MIJ is not recorded in ICT nor in SCT.
- **Signalled by** : **UNLOAD MESSAGE POPUP U19** In addition such masks will be flagged by a ?? marker (an ok marker indicates masks in IC, a <= marker masks in SC) The message will appear once at the beginning of the unload phase and will not give details about individual masks (use the mentioned flags)
- **Action**: error, operator can choose to: press **RETRY** to suspend unload, run recovery rescan IC and recovery rescan SC to update the content of ICT and SCT and finally resume IC preparation hoping that the situation has been cleared ; alternately the operator can press **CANCEL** to proceed with unloading and loading, and be ready to encounter further errors later (typically the condition described in [8.1.5.19](#))

## 8.1.6 Contingencies during discard phase

The following are the recognized contingency conditions during the mask discard cycle.

### 8.1.6.1 *timeout during connection to bar code reader*

- **Condition** : MHS cannot establish connection with the SC bar code reader.
- **Signalled by** : **BAR CODE MESSAGE POPUP B01**
- **Action**: the popup will ask operator to check barcode reader is connected to port (check serial cable at both ends) and switched on (check power cable plugged in and LED on SC barcode reader). Fasten any loose cable and, if everything looks OK, press **RETRY**. In case of repeated failures or clear hardware failure press **CANCEL** to exit program, and refer to Maintenance Manual section 12.5 (bar code or serial card failure).

### 8.1.6.2 *no MDJ found in staging area or null MDJ*

One has two cases :

- **Condition** : no MDJ is present in staging area.
- **Signalled by** : **DISCARD MESSAGE POPUP D01**
- **Action**: this is an informative message. No further action required. Click OK to acknowledge and the discard program will be terminated
  
- **Condition** : the MDJ order is null (i.e. refers to zero masks!).
- **Signalled by** : **CONVERT MESSAGE POPUP D02**
- **Action**: Click OK to acknowledge and generate a valid MDT (the order is formally fulfilled)

### 8.1.6.3 *mask still in IC*

- **Condition** : mask is not found in SCT (because it appears from ICT it is still loaded in the IC)
- **Signalled by** : **DISCARD MESSAGE POPUP D03**
- **Action**: Click OK to acknowledge and the discard program will be skip discarding of this mask (and report it as unsuccessful in the termination report).

### 8.1.6.4 *mask not found in SCT because it is nowhere (dropped and not yet re-made)*

- **Condition** : mask is not found in SCT nor in ICT (presumably it was dropped and not yet been re-manufactured )
- **Signalled by** : **DISCARD MESSAGE POPUP D04**
- **Action**: Click OK to acknowledge. Since the mask has to be discarded anyhow, the relevant LMD file will be deleted, and the discard considered successful.

### 8.1.6.5 *mask physically not found in SC (SCT vs SC mismatch)*

- **Condition** : mask cannot be physically found in the SC (although listed in SCT : there is a mismatch between the content of the table and of the relevant cabinet ! )
- **Signalled by** : **DISCARD MESSAGE POPUP D06** after operator has aborted the search in the SC pressing the **MASK NOT FOUND** button.
- **Action**: Verify simple problems (like masks not well pushed into SC) and click **YES** button to retry search (eventually in other quadrants if mask misplaced). If problem persists click **NO**. This will cause **DISCARD MESSAGE POPUP D07** to appear. In order to remove entry from SCT and delete LMD associated to given mask



anyhow operator shall reply **OK**. This will suspend the discard function. One must then run the recovery rescan SC procedure (see [6.1.10.4](#)) on the named mask, and then re-enter discard. If operator replies **CANCEL** instead, discard will skip processing of the missing mask (considered *not* discarded) and continue.

#### **8.1.6.6 LMD file for given mask does not exist**

- **Condition** : there is no LMD file corresponding to the mask to be discarded (already accidentally deleted).
- **Signalled by** : condition is not signalled
- **Action**: program silently proceeds with discarding (considered successful)

#### **8.1.6.7 mask dropped**

- **Condition** : mask is dropped while moving it from SC to Waste Basket.
- **Signalled by** : condition is not explicitly signalled
- **Action**: this is not a problem, throw away the damaged mask (it has to be discarded anyhow) and when prompted by **DISCARD MESSAGE POPUP D05** to acknowledge mask has been discarded, click **OK** to proceed with deletion of relevant files and updating of the SCT.

#### **8.1.6.8 not all masks discarded**

- **Condition** : not all masks listed in MDJ have been discarded
- **Signalled by** : **DISCARD MESSAGE POPUP D09** after operator has pressed the **END** button
- **Action**: the operator has an option to cancel the termination and re-enter the program to complete discarding. If he clicks **OK** and decides to really terminate, a further **DISCARD MESSAGE POPUP D10** reminds that a termination report will be sent and requires further confirmation. Note that a cancel action in both popups will re-enter the discard function. There is *no way* to willingly suspend it (except in case of errors like [8.1.6.5](#)).



## 8.2 List of popup messages

The following tables list all popup messages. The first column corresponds to the label in the title bar of the popup (which usually contains also a sequence number). The second column gives the message text. The third column makes cross reference to the appropriate subsection in 8.1, which describes how to deal with the anomalous condition (there is no cross reference for normal conditions which just require an acknowledgment).

The first set of tables refers to the MHS program.

First the general messages which can be generated by any MHS function.

Popup heading	Message text	Action
MHS MESSAGE	Click OK to confirm EXIT from Mask Handling System	<a href="#">8.1.1.2</a>
ROBOT MESSAGE R01	robot not responding. Verify the power supply, the cables and click Retry. If the problem remains click Cancel to exit the program.	(ack) <a href="#">8.1.5.2</a>
ROBOT MESSAGE R02	Robot does not respond. The program will be aborted	<a href="#">8.1.5.4 condition 1</a> see also <a href="#">8.1.5.5</a> and <a href="#">8.1.5.7</a>
ROBOT MESSAGE R03	Robot handshaking error. The program will be aborted	as previous
ROBOT MESSAGE R04	Robot does not respond. The program will be aborted	as previous
ROBOT MESSAGE R05	Robot command syntax error. The program will be aborted	as previous
ROBOT MESSAGE R06	Robot handshaking error. The program will be aborted	as previous
ROBOT MESSAGE R07	Soft recoverable error detected Pull the emergency button or remove the photo cell obstruction. Click RETRY to reset and to repeat the last command or CANCEL to abort the program.	<a href="#">8.1.5.4 condition 2</a> see also <a href="#">8.1.5.5</a> and <a href="#">8.1.5.7</a>
ROBOT MESSAGE R08	Home position lost or not acquired Check proximity switch. Click RETRY to re-home and then repeat the last command or CANCEL to abort the program.	<a href="#">8.1.5.4 condition 3</a> see also <a href="#">8.1.5.7</a>
ROBOT MESSAGE R09	Robot hardware error. The program will be aborted	<a href="#">8.1.5.4 condition 4</a> see also <a href="#">8.1.5.5</a> and <a href="#">8.1.5.7</a>
ROBOT MESSAGE R10	PAY ATTENTION ! The robot will move to the HOME position	<a href="#">8.1.5.3</a>
ROBOT MESSAGE R11	Robot busy processing a command Click RETRY to repeat the last command or CANCEL to abort the program.	<a href="#">8.1.5.4 condition 2</a> also <a href="#">8.1.5.5</a> and <a href="#">8.1.5.7</a>
BAR CODE 1 MESSAGE B01	The stand bar code reader does not respond. Verify the power supply, the cables and click Retry. If the problem remains click Cancel to exit the program.	<a href="#">8.1.4.1</a> <a href="#">8.1.5.1</a> <a href="#">8.1.6.1</a>
BAR CODE 2 MESSAGE B02	The SC bar code reader does not respond. Verify the power supply, the cables and click Retry. If the problem remains click Cancel to exit the program.	<a href="#">8.1.5.1</a>

Then the messages generated by the main functions (convert, store, unload, load, discard).



Popup heading	Message text	Action
CONVERT MESSAGE C01	The CONVERT program cannot start. You have to complete the STORE of the masks produced with the previous order	8.1.2.3
CONVERT MESSAGE C02	No manufacturing order. The program will be terminated.	8.1.2.1
CONVERT MESSAGE C03	The orders folder does not contain the .mmj file. FTP in progress? Try again later.	8.1.2.5 condition 1
CONVERT MESSAGE C04	The order <i>mmj_name</i> is a null order. A null report will be generated and the CONVERT program will be terminated	8.1.2.5 condition 2
CONVERT MESSAGE C05	ERROR: <i>Errmsg</i> on file <i>qnnnnn</i> .MSF. The file will be skipped	8.1.2.6 see also 8.1.2.12
ERROR RECOVERY C06	ERROR: <i>Errmsg</i> in file <i>qnnnnn</i> .MSF Gerber conversion. File skipped.	8.1.2.8
.lmd GENERATION MESSAGE C07	No .gbr file to convert into .lmd. The program will be terminated	(ack) 8.1.2.13
.lmd file generation C08	To CONTINUE the program execution CLICK OK	ack
CircuitCam -.lmd GENERATION MESSAGE C09	The file <i>qnnnnn.gbr</i> has not been converted. Retry ? CLICK NO to skip the file YES to try again	8.1.2.9
CONVERT MESSAGE C10	CONVERT program terminated. click OK	ack
CONVERT MESSAGE C11	CONVERT program suspended. click OK	ack
CONVERT MESSAGE C12	Exiting CONVERT program	ack
CONVERT MESSAGE C13	Mask set incomplete. No LMD file of this mask set will be copied to MMCU	8.1.2.11
CONVERT MESSAGE C14	No .msf file to translate into .gbr. The program will be terminated	(ack) 8.1.2.13
CONVERT MESSAGE C15	No mask sets to manufacture The program will be terminated	(ack) 8.1.2.13

Popup heading	Message text	Action
STORE MESSAGE S01	CONVERT not yet terminated	8.1.4.16
STORE MESSAGE S02	STORE function suspended. Click OK to confirm	confirm
STORE MESSAGE S03	No new masks to store	8.1.4.15
STORE MESSAGE S04	<i>qnnnnn</i> is an illegal code number. Skipped.	8.1.4.3
STORE MESSAGE S05	The mask <i>qnnnnn</i> does not belong to the pending manufacturing order <i>mmj_name</i> Do not store this mask in the SCs NOW. Skipped.	8.1.4.3
STORE MESSAGE S06	The mask <i>qnnnnn</i> appears already stored in this session. Please verify: if you have not yet stored the mask to SC, move it NOW if you have produced a mask duplicate, discard it.	8.1.4.4
STORE MESSAGE S07	ERROR - MISMATCH The mask <i>qnnnnn</i> is already listed in the Storage Cabinet Tables. Before any action (STORE, DISCARD or SC RESCAN) please try to understand why.	8.1.4.4
STORE MESSAGE S08	HAVE YOU STORED the mask <i>qnnnnn</i> in the <i>instrument col</i> cabinet (ST <i>quadr</i> ) ?	ack ( 8.1.4.7)
STORE MESSAGE S09	Do you want to re-manufacture it immediately ? Click YES to retry storing now (in the meanwhile got to MMCU and make it) Click NO to store the next mask	8.1.4.6
STORE MESSAGE S10	STORE NEW MASKS program will be terminated and the manufacturing report will be sent. Click OK to confirm	8.1.4.13 8.1.4.14



Popup heading	Message text	Action
STORE MESSAGE S11	Are you really sure ? click OK to confirm only if no more masks can be manufactured	8.1.4.13 8.1.4.14
STORE MESSAGE S12	STORE NEW MASKS program terminated. click OK to confirm	8.1.4.10

Popup heading	Message text	Action
UNLOAD MESSAGE U01	Exiting IC PREPARATION sequence	ack
UNLOAD MESSAGE U02	No mask insertion order. The program will be terminated.	8.1.5.10
UNLOAD MESSAGE U03	The current Mask Insertion Order does not require mask unloading from ICs.	ack
UNLOAD MESSAGE U04	The IC holder is already loaded on the robot lift?	8.1.5.6
UNLOAD MESSAGE U05	PAY ATTENTION ! The robot will move to the LOADING position	Ack ( 8.1.5.6)
UNLOAD MESSAGE U06	When the IC holder is loaded on the robot lift click OK.	Ack ( 8.1.5.6)
UNLOAD MESSAGE U07	Have you successfully extracted the mask ?	8.1.5.12 8.1.5.13
UNLOAD MESSAGE U08	Has the slot been emptied manually by purpose at the telescope ? YES will skip unloading this mask and will remove it from ICT. NO (also in case of robot malfunction) will suspend UNLOAD to allow you to run RECOVERY/RESCAN	8.1.5.12
UNLOAD MESSAGE U09	WRONG MASK - You have extracted the mask <i>qnnnnn</i> instead of <i>qmmmmm</i> Insert again the mask in the IC. The unloading will be suspended. A rescan of the ICs is necessary	8.1.5.14
UNLOAD MESSAGE U10	The mask <i>qnnnnn</i> is already in <i>instrument SCT quadr</i> IC unloading must be suspended and RECOVERY / REMOVE must be run Insert again the mask in the IC and click OK.	8.1.5.27
UNLOAD MESSAGE U11	Have you stored the mask <i>qnnnnn</i> in the <i>col cabinet SC quadr</i> ?	8.1.5.15
UNLOAD MESSAGE U12	If the mask <i>qnnnnn</i> is damaged, discard it. A new one must be manufactured and stored in SC with recovery / single mask insertion	8.1.5.15
UNLOAD MESSAGE U13	PAY ATTENTION ! The robot will move to the UNLOADING position	8.1.5.25
UNLOAD MESSAGE U14	The ICs UNLOAD program is not terminated. Clicking OK the program will be suspended and you can restart it later to complete unloading. Click Cancel to continue unloading.	Confirm suspend or continue
UNLOAD MESSAGE U15	IC UNLOAD program terminated. click OK to confirm	Ack
UNLOAD MESSAGE U16	A robot error occurred. Click RETRY to suspend the program. You can than switch off/on the robot and restart IC PREPARATION. Click CANCEL to end the program. The report will be sent.	8.1.5.26
UNLOAD MESSAGE U17	The current Mask Insertion Order does not require mask loading into ICs. ICTs, SCTs and MIT report will be available to MCS and IC PREPARATION terminated.	8.1.5.17



Popup heading	Message text	Action
UNLOAD MESSAGE U18	The current Mask Insertion Order requires mask loading in the ICs. The LOAD program will be started	Ack
UNLOAD MESSAGE U19	Some of the masks requested by the insertion order are neither in SCTs nor in ICTs. Click RETRY to suspend IC PREPARATION for IC and SC rescan. Click CANCEL to continue, but errors will be generated later.	8.1.5.28

Popup heading	Message text	Action
LOAD MESSAGE L01	The IC holder is already loaded on the robot lift?	8.1.5.6
LOAD MESSAGE L02	PAY ATTENTION ! The robot will move to the LOADING position	ack 8.1.5.6
LOAD MESSAGE L03	When the IC holder is loaded on the robot lift click OK.	ack 8.1.5.6
LOAD MESSAGE L04	PAY ATTENTION ! The robot will move to the UNLOADING position	8.1.5.25
LOAD MESSAGE L05	Have you safety moved the mask to the robot stand ?	8.1.5.20
LOAD MESSAGE L06	If the mask <i>qnnnnn</i> is damaged, it must be remanufactured. Do you confirm that cannot be loaded?	8.1.5.20
LOAD MESSAGE L07	You are loading a wrong mask ( <i>qnnnnn</i> ) Move it back to the SC and click OK.	8.1.5.21
LOAD MESSAGE L08	Have you inserted the mask ?	8.1.5.22 8.1.5.23
LOAD MESSAGE L09	The LOAD program will be aborted. If the slot already contains a mask RECOVERY / RESCAN IC must be used If the robot is not in the right position try to switch it off and on and restart IC PREPARATION In any case you have to move back the mask to the Storage Cabinet	8.1.5.22
LOAD MESSAGE L10	Are you sure that the mask <i>qnnnnn</i> is not in SC? Search also in other quadrants. Click Retry to search again or Cancel to skip the mask loading.	8.1.5.19
LOAD MESSAGE L11	The ICs LOAD program is not terminated. Click OK to suspend the IC PREPARATION. You can restart it later. Click Cancel to continue loading.	ack
LOAD MESSAGE L12	IC LOAD program terminated. click OK to confirm	ack
LOAD MESSAGE L13	A robot error occurred Click RETRY to suspend the program. You can than switch off/on the robot and restart IC PREPARATION. Click CANCEL to end the program. The report will be sent.	8.1.5.26

Popup heading	Message text	Action
DISCARD MESSAGE D01	No discard order. The program will be terminated.	8.1.6.2
DISCARD MESSAGE D02	The order <i>mdj_name</i> is a null order. A null report will be generated and the DISCARD program will be terminated	8.1.6.2



Popup heading	Message text	Action
DISCARD MESSAGE D03	The mask <i>qnnnnn</i> is still in the instrument cabinet <i>quadr</i> Discard is impossible. Click OK	8.1.6.3
DISCARD MESSAGE D04	The mask <i>qnnnnn</i> is neither in SCs nor in ICs. Mask already discarded ? The discard is considered successfull. Click OK	8.1.6.4
DISCARD MESSAGE D05	HAVE YOU DISCARDED THE MASK <i>qnnnnn</i> ?	8.1.6.7
DISCARD MESSAGE D06	Please verify the position of the masks in the Storage Cabinets. Retry ?	8.1.6.5
DISCARD MESSAGE D07	Mask physically not found. Click OK to suspend the DISCARD program and to run RECOVERY / RESCAN SC procedure Click CANCEL to continue discarding the other masks of the order	8.1.6.5
DISCARD MESSAGE D08	Mask discard program aborted before retrieving the order	ack
DISCARD MESSAGE D09	Not all the masks of the order have been discarded. Click OK to end the program	8.1.6.8
DISCARD MESSAGE D10	Do you confirm? The discard report will be sent.	8.1.6.8
DISCARD MESSAGE D11	Mask discard program terminated. click OK to confirm	ack
DISCARD MESSAGE D12	Exiting DISCARD program	ack



Then the messages generated by the recovery functions.

<b>Popup heading</b>	<b>Message text</b>	<b>Action</b>
ENABLE-DISABLE MESSAGE Ed01	One of the slots you have selected contains a mask code. You cannot modify this entry. Click Retry or Cancel to CLOSE	ack 8.3.9

<b>Popup heading</b>	<b>Message text</b>	<b>Action</b>
OFF-LINE SINGLE MASK IDENTIFICATION Id01	Illegal bar code. Ignore this mask.	ack 8.3.6
<del>OFF-LINE SINGLE MASK IDENTIFICATION Id02</del>	Message deleted	
OFF-LINE SINGLE MASK IDENTIFICATION Id03	Do you want to identify another mask?	Option to loop Y N see 8.3.6

<b>Popup heading</b>	<b>Message text</b>	<b>Action</b>
INSERT MESSAGE In01	Illegal mask name. Do not move the mask to the Storage Cabinet. Insert ignored.	ack
<del>INSERT MESSAGE In02</del>	Message deleted	
INSERT MESSAGE In03	The mask <i>qnnnnn</i> is already in the SC <i>quadr</i> This mask will be ignored.	ack
INSERT MESSAGE In04	HAVE YOU STORED the mask <i>qnnnnn</i> in the <i>instrument col</i> cabinet (ST <i>quadr</i> ) ? OK will update the <i>instrument</i> Storage Cabinet Tables Cancel will ...	8.3.3
INSERT MESSAGE In05	Do you want to insert another mask?	Option to loop Y N
INSERT MESSAGE In06	INSERT program terminated. Click OK	ack

<b>Popup heading</b>	<b>Message text</b>	<b>Action</b>
REMOVE MESSAGE Re01	Type the bar code of the mask to remove and click OK Cancel or null entry will exit the program	requires user to type in a mask identifier
REMOVE MESSAGE Re02	You have typed an invalid code	ack
REMOVE MESSAGE Re03	Illegal mask name. Remove ignored.	ack
<del>REMOVE MESSAGE Re04</del>	Message deleted.	
REMOVE MESSAGE Re05	The mask <i>qnnnnn</i> is not in the Storage Cabinet <i>Tables</i> <i>Remove</i> ignored.	ack
REMOVE MESSAGE Re06	Do you want to remove another mask?	Option to loop Y N
REMOVE MESSAGE Re07	HAVE YOU DISCARDED THE MASK <i>qnnnnn</i> ? OK will update <i>the</i> Storage Cabinet Tables Cancel will ignore the mask <i>removal</i>	8.3.4
REMOVE MESSAGE Re08	REMOVE program terminated. Click OK	ack

<b>Popup heading</b>	<b>Message text</b>	<b>Action</b>
IC RESCAN MESSAGE Ic01	The IC holder is already loaded on the robot lift?	ack ( 8.1.5.6)
IC RESCAN MESSAGE Ic02	PAY ATTENTION ! The robot will move to the LOADING position	8.1.5.6
IC RESCAN MESSAGE Ic03	Load the IC holder on the robot lift and click OK	ack ( 8.1.5.6)
IC RESCAN MESSAGE Ic04	PAY ATTENTION ! The robot will move to the UNLOADING position	8.1.5.25
IC RESCAN MESSAGE Ic05	Reinsert the mask <i>qnnnnn</i> into IC <i>quadr</i> slot <i>slot</i> and click OK	ack ( 8.3.2)
IC RESCAN MESSAGE Ic06	The ICs SCAN program is not terminated.	8.3.2





Popup heading	Message text	Action
	Clicking OK the program will be aborted and the IC Tables will not be updated. Click Cancel to continue IC Scan.	
IC RESCAN MESSAGE Ic07	IC SCAN program terminated. Click OK to update the IC Tables. Click Cancel to exit without IC Tables updating.	8.3.2
IC RESCAN MESSAGE Ic08	WARNING! this is a quadrant <i>ident</i> mask in the WRONG <i>quadr</i> Do not reinsert the mask and the slot will be considered empty.	ack ( 8.3.2)
<i>IC RESCAN MESSAGE Ic09</i>	<i>Illegal mask bar code</i> <i>Do not reinsert the mask and the slot will be considered empty.</i>	<i>ack ( 8.3.2)</i>

Popup heading	Message text	Action
SCAN SC MESSAGE Sc01	Position the bar code reader to scan ONLY the Storage Cabinet <i>i</i> and click OK	ack
SCAN SC MESSAGE Sc02	Please count the masks currently inserted in the <i>instrument</i> Storage Cabinet <i>i</i> and type the number in the input box. Cancel will abort the program	requires user to type in a number between 0 and 100
SCAN SC MESSAGE Sc03	You have to type a number	ack
SCAN SC MESSAGE Sc04	You have to type a number between 0 and 100	ack
SCAN SC MESSAGE Sc05	The mask <i>qnnnnn</i> is stored in the wrong Storage Cabinet. Click OK to search and to remove it.	ack
SCAN SC MESSAGE Sc06	The scan of the Storage Cabinet <i>i</i> is terminated. Click OK to continue.	ack
SCAN SC MESSAGE Sc07	All the masks of the <i>instrument</i> Storage Cabinets have been identified Click OK to update the Storage Cabinet Tables Click Cancel to exit without tables updating	8.3.1
SCAN SC MESSAGE Sc08	The storage cabinet scan is not terminated. The program will be aborted.	ack
SCAN SC MESSAGE Sc09	Extract the mask <i>qnnnnn</i> from the Storage Cabinet Later you have to run the RECOVERY/INSERT program to store the mask in the right SC When done click OK.	8.3.1

Popup heading	Message text	Action
OFF-LINE MASK SEARCH INPUT Se01	Type the code of the mask you wish to search and click OK Cancel or null entry will terminate the program	requires user to type in a mask identifier
OFF-LINE MASK SEARCH MESSAGE Se02	You have typed an invalid <i>bar</i> code	ack
OFF-LINE MASK SEARCH MESSAGE Se03	<i>You have typed an illegal bar code</i>	ack
<del>OFF-LINE MASK SEARCH MESSAGE Se04</del>	<del><i>Message deleted</i></del>	
OFF-LINE MASK SEARCH MESSAGE Se05	The mask <i>qnnnnn</i> is not in the <i>instrument</i> Storage Cabinet Tables. Search anyway ?	8.3.5
OFF-LINE MASK SEARCH MESSAGE Se06	Do you want to search another mask?	Option to loop Y N

The final table refers to the Cut Manager program.



Popup heading	Message text	Action
Cut Manager MESSAGE	Click OK to confirm EXIT from Cut Manager. Then you can close StencilMaster and switch off the LPKF laser machine or operate manually StencilMaster.	confirm
Cut Manager MESSAGE	OK button will start StencilMaster <i>or will prompt a message if already running</i> Cut Manager <b>WILL BE ICONIZED</b> . <i>(Afterwards if the LPKF laser machine is OFF switch it ON).</i> <i>Then to give back the contro to Cut Manager clickon the icon at the screen bottom.</i> Cancel button will EXIT Cut manager.	6.2
<i>Cut Manager MESSAGE</i>	<i>Stencil Master is already running</i>	<i>ack</i>
Cut Manager MESSAGE	The Current folder contains the file <i>name . lmd</i> Click Cancel to remove this file <i>if the mask has already been manufactured</i> or Retry to manufacture it.	8.1.3.8
Cut Manager MESSAGE	Click OK to confirm EXIT from Cut manager.	8.1.1.2

The disk mirroring program has no interactive error messages since it runs in the background.

### 8.3 Usage of standard recovery procedures

We give here a description of the usage of the recovery functions provided by MHS. Of course further contingencies may occur during their usage, but they will in general be self-explanatory and/or similar to those which occur during standard operations (e.g. the case of establishing the serial connection to a bar code reader or to the robot, wrong or invalid mask identifier etc.).

Note that currently the recovery function may update the local SCT and ICT but will **not** update the copies of the SCT and ICT in the staging area (i.e. *those known to MCS*) except as a consequence of the execution of a (MMJ, MIJ or MDJ) order. Therefore if the normal processing of an order is interrupted (suspended), a recovery function invoked, and the processing of the order resumed, MCS will retrieve the actual SCT and ICT together with the termination report. If instead the recovery function is run after the order processing has been terminated (normally or abnormally), the actual SCT and ICT will be available to MCS *only after the next order will be processed*.

#### 8.3.1 Storage cabinet rescan

This function must be invoked (via the form described in 6.1.10.1) whenever there is a reason to believe that the content of the SCT is not aligned with the actual content of the SC. The result of the action will be logged to the ops log file and to the last action file. The typical course of action will be the following :

- operator is invited to press the INITIALIZE button, which will connect the SC barcode reader
- operator has to press the SCAN START button to begin the procedure, which will do one quadrant at a time (it can be aborted at any time pressing END or by other means described below)
  - operator must go to the SC and position the bar code reader in front of the right quadrant, he should also place the dark screen in front of the masks of the adjacent quadrant to limit the readout only to one quadrant, then press OK in the POPUP Sc01. It is **important** that the reader is in the *right quadrant before* one presses **OK**
  - operator shall count the number of masks present in the quadrant and type the number in the POPUP Sc02. If the quadrant is empty type 0 (zero) and the program will proceed to the next quadrant.
  - if the number is invalid (POPUP Sc03 ) or out of range (POPUP Sc04 ) the procedure restarts from Sc01 after acknowledgment. A reply of CANCEL to Sc02 allows to abort the entire rescan.
  - the operator moves slowly the bar code reader to scan all masks in the quadrant, until an audible beep is generated. This should normally be POPUP Sc06, which signals the end of a complete scan of a quadrant.



Possible contingencies at this stage are described further below. Normally an acknowledgment at this stage goes back to Sc01 for the next quadrant.

- when all four quadrants are done, **POPUP Sc07** appears. The operator has the choice to press **OK** to update the SCT on disk, or **CANCEL** not to it.
- after this press **END** to terminate

At any time the operator may (typically pressing **END**) request to abort the SC rescan. The **POPUP Sc08** allows to acknowledge. In case of abort SCT are not updated.

There are two possible contingencies during the scanning with the bar code reader :

1. the first occurs when a *misplaced mask* is encountered (i.e. a mask belonging to another quadrant). In this case **POPUP Sc05** appears, and the **WRONG MASK FOUND** button is made visible. Operator is instructed to search the offending mask, when found he shall press the **WRONG MASK FOUND** button, extract the mask, set it aside, and later run recovery insert to store it in the right quadrant (as instructed by **POPUP Sc09**)
2. the second occurs when the *search loop can't be terminated* Typically this occurs because the operator *has counted wrongly* the number of mask (it might also be due to two stored masks having the same mask id - they won't be recognised since the same mask can be scanned more than once - but this is *highly unlikely* because of the protections against this eventuality in the store and insert functions). Only possible action is to press **END** to abort SC rescan, and restart afresh.

### 8.3.2 Instrument cabinet rescan

This function must be invoked (via the form described in 6.1.10.2) whenever there is a reason to believe that the content of the ICT is not aligned with the actual content of the IC. The result of the action will be logged to the ops log file and to the last action file. The typical course of action will be the following :

- operator is invited to press the **INITIALIZE** button, which will connect the robot and stand barcode reader. *During this stage the operator is also be asked to check (for each quadrant) to which IC set (original, A or spare, B) the ICs in use belong via the IC configuration form*
- operator has to load the IC holder onto the Z-displacement platform (if not already loaded) exactly as in the case of IC preparation (see 7.4, steps 7 and 8 and related contingencies in 8.1.5)
- operator has to press the **FIRST SLOT** button to move the first slot of the first IC in front of the decoding stand
  - if the slot is empty, operator presses the **SLOT EMPTY** button
  - otherwise the operator presses the **EXTRACT** button to activate the barcode reader, and extracts the mask
  - if the mask is recognised as belonging to the correct quadrant, a confirmation **POPUP Ic05** appears, inviting the operator to reinsert the mask in the IC, so that the mask is recorded in the new ICT
  - if the mask is recognised as wrong (*i.e. not belonging to the expected quadrant or having an otherwise illegal barcode*), a confirmation **POPUP respectively Ic08 or Ic09** appears, inviting the operator to set the mask aside (it has to be later stored in the SC using recovery insert) while the IC slot will be considered empty
- operator has to press the **NEXT SLOT** button to loop processing the next slot
- when all slots of all ICs have been scanned, operator is invited to press the **END** button to terminate. The **POPUP Ic07** appears. A reply of **OK** will terminate updating the ICT with the new ICT content. A reply of **CANCEL** exits without updating the real ICT.

At any time the operator may press **END** to abort the IC rescan. A The **POPUP Ic06** allows to confirm abort (**OK**) without updating the ICT, or to **CANCEL** abort and continue with the next slot.

*Note that IC rescan will identify each IC slot as either empty or containing a mask, i.e. reconstruct the ICT table with the physical content of the ICs. It will ignore any slot declared out of use (see 8.3.9). It will therefore be necessary to run the Enable or Disable IC slots recovery function (see 6.1.10.8) after IC rescan if one wishes to declare some slots as out of use.*



### 8.3.3 Inserting (storing) a single mask

This function must be invoked (via the form described in 6.1.10.3) whenever there is the need to store off line one or more masks in the SC (updating the SCT). Off-line means here without a corresponding MMJ request. In addition to usual checks, this function will verify that the mask is identified by a valid barcode for *VIMOS*.

This function uses the MHS robot decoding stand (but does not use the robot).

This function and its results are logged to the ops log file, and to the last action file. The typical course of action will be the following :

- operator is invited to press the **INITIALIZE** button, which will connect the stand barcode reader
- operator presses **INSERT** to switch on the bar code reader
- operator puts the mask on the stand
- the value of the barcode is displayed on the screen. In case of errors an additional **POPUP** warning (In01) may appear if the mask *has a "wrong" barcode (i.e. one not conforming with VIMOS convention)*, or the **POPUP** warning In03 appears if a mask with the same id is already recorded as stored in the SCT (a recovery rescan SC is recommended in such case). In both case the mask is ignored after acknowledgment.
- Otherwise the operator shall insert the mask in a free slot of the right SC as instructed, and reply **YES** to the **POPUP** In04. At this time the mask is recorded in the SCT. If the mask is damaged or for any reason not stored reply **NO** and it will not be recorded in the SCT.
- The **POPUP** In05 offers the operator the possibility to loop back and inserting another mask
- Otherwise operator presses **END** to terminate after confirmation

### 8.3.4 Removing (discarding) a single mask

This function must be invoked (via the form described in 6.1.10.4) whenever there is the need to discard off line one or more masks in the SC (updating the SCT). Off-line means here without a corresponding MDJ request. This function shall be used whenever one wants to take a mask out of the SC and to ensure the SCT is up to date (one shall never take a mask out manually). In addition to usual checks, this function will verify that the mask is identified by a valid barcode for *VIMOS*. Note that the relevant LMD file is not deleted by this function (but only by execution of a full discard order).

This function uses the SC bar code reader

This function and its results are logged to the ops log file, and to the last action file. The typical course of action will be the following :

- operator is invited to press the **INITIALIZE** button, which will connect the SC barcode reader
- operator presses **REMOVE**
- operator types in the required mask id in the popup Re01 (an action of **CANCEL** at this stage exits as a null run)
- the mask id is verified. In case of an invalid bar code (typing error, message Re02) the previous popup is shown requiring to supply a valid one
- Other errors involve an additional **POPUP** warning (Re03) if the mask *has a "wrong" barcode (i.e. one not conforming with the VIMOS convention)*, or the **POPUP** warning Re05 when the mask is not recorded as stored in the SCT. In all these cases the request will be ignored.
- Otherwise the operator shall go to the SC and search the mask as in 7.5 steps 6-9,
- When the mask is found the operator presses the **FOUND** button and is shown the **POPUP** Re07 to confirm mask discard. After an answer of **YES** the mask is removed in the SCT. If the mask is not discarded, put it back into SC and reply **NO** and it will not be removed from the SCT.
- If the mask cannot be found the operator presses the **NOT FOUND** button. The mask is considered discarded and removed anyhow from SCT.
- The **POPUP** Re06 offers the operator the possibility to loop back and removing another mask
- Otherwise operator presses **END** to terminate after confirmation

### 8.3.5 Searching a mask in SC

This function must be invoked (via the form described in 6.1.10.5) whenever there is the need to locate one or more masks in the SC without taking it off. This function has no correspondence with standard functions and is provided for maintenance purposes. In addition to usual checks, this function will verify that the mask is identified by a valid barcode for *VIMOS*.



This function uses the SC bar code reader

This function and its results are not logged to the ops log file, nor to the last action file (this is considered a fully off-line function). The typical course of action will be the following :

- operator is invited to press the **INITIALIZE** button, which will connect the SC barcode reader
- operator presses **SEARCH**
- operator types in the required mask id in the popup Se01 (an action of **CANCEL** at this stage exits)
- the mask id is verified. In case of an invalid bar code (typing error, message Se02) or of an additional **POPUP** warning (Se03) if the mask *has a "wrong" barcode (i.e one not conforming with VIMOS convention)* the request will be ignored.
- If the mask is not recorded in SCT, the operator is asked by **POPUP** Se05 whether he wants to search it anyway in SC (reply **YES**) or ignore the request (reply **NO**)
- Otherwise the operator shall go to the SC and search the mask as in 7.5 steps 6-9,
- When the mask is found the operator presses the **FOUND** button and the result is shown on the screen
- If the mask cannot be found the operator presses the **STOP SEARCH** button.
- The **POPUP** Se06 offers the operator the possibility to loop back and searching another mask
- Otherwise operator presses **END** to terminate

### 8.3.6 Identifying a mask

This function must be invoked (via the form described in 6.1.10.6) whenever there is the need to identify one or more masks which are not in the SC without storing them. This function has no correspondence with standard functions and is provided for maintenance purposes. In addition to usual checks, this function will verify that the mask is identified by a valid barcode for **VIMOS**.

This function uses the MHS robot decoding stand (but does not use the robot).

This function and its results are not logged to the ops log file, nor to the last action file (this is considered a fully off-line function). The typical course of action will be the following :

- operator is invited to press the **INITIALIZE** button, which will connect the stand barcode reader
- operator presses **IDENTIFY** to switch on the bar code reader
- operator puts the mask on the stand
- the value of the barcode is displayed on the screen. An additional **POPUP** warning (Id01) may appear if the mask *has a "wrong" barcode (i.e. one not conforming to the VIMOS convention)*
- The **POPUP** Id03 offers the operator the possibility to loop back and identifying another mask
- Otherwise operator presses **END** to terminate

### 8.3.7 Emptying the IC

In all cases it will be necessary to empty completely the IC for maintenance, autonomously with respect to MCS, this can be done using the standard IC preparation procedure described in section 7.4, but selecting the special MIJ called null.mij. This is a permanent file held on MHCUC, which will never be deleted, and will not generate a report to MCS.

### 8.3.8 Unloading the IC holder

In all cases where (IC preparation or IC rescan) processing is interrupted and the IC holder is still mounted on the MHS robot Z-displacement platform, it will be necessary to move the platform to the lowest position in order to unload the IC holder. This can be done with a dedicated button in the main recovery form (see 6.1.10).

Note that the robot will make two movements, first to the home position, and then to the unload position. Additionally the communication will be tested, and any soft errors will be cleared before moving.

Note also that this function can be used as a quick way to unload the IC holder after any interruption of other programs using the robot, but of course *cannot* be used to lower the platform in case the communication with the robot or the robot operation is impossible for severe hardware reasons.

There are *no contingency popups* foreseen for this recovery action. There is just a message indicating completion of the operation in the **NEXT ACTION** text area, or an error message in case of time out of the serial connection to the robot before sending the commands (a time out in between is not handled).



### 8.3.9 Declaring an IC slot out of use

*This function must be invoked (via the form described in 6.1.10.8) whenever there is the need to declare that one or more IC slots will not be used for mask insertion (because of any hardware problem). This is called "disabling" a slot or declaring it as "out of use".*

*The same function can be invoked to restore a slot back in use, which is called "enabling" a slot .*

*The procedure to disable one or more slots is the following*

- *click the DISABLE SLOTS button, which activates the ICT lists for editing*
- *select any number of slots (provided they are empty) clicking on them*
- *click the active (red) STORE button to commit the change to the ICT on disk*
- *click the CLOSE button*

*The procedure to enable one or more slots is the following*

- *click the ENABLE SLOTS button, which activates the ICT lists for editing*
- *select any number of slots (provided they are out of use) clicking on them*
- *click the active (green) STORE button to commit the change to the ICT on disk*
- *click the CLOSE button*

*In both cases an error is generated if one attempts to change state to a slot containing a mask (or recorded as such in the ICT). In practice this means it is recommended that slots to be declared out of use shall be emptied in a way known to MHS. This means that taking the mask out of the slot is not enough. One should either run IC Rescan (see 8.3.2) if the slot is already empty but the ICT does not reflect this, or empty the entire cabinet with an appropriate insertion order, like the resident null order (see 8.3.7), or a null MIJ generated at mask tracker level.*

*Remember that slots declared out of use are handled as such. They are ignored for what concerns inserting a mask, but are considered as they were empty by the algorithm which decides the filling sequence (which tries to leave free slots around a slot containing a mask whenever possible). This is a further reason to ensure out of use slots are physically empty.*

## 9. Routine Maintenance

This section describes the quality control activities to be periodically performed with the roughness meter, and the other routine maintenance activities to be done periodically. All remaining (extraordinary) maintenance activities are described in the [Maintenance Manual](#)

### 9.1 Roughness Meter (quality control)

We describe here only the steps necessary for the actual quality check procedures (see 7.9) used. A full description about the usage of the roughness meter and associated software is outside of the scope of the present document : make reference to [Taylor-Hobson manual \[ref. 9\]](#). for any further information.

#### 9.1.1 Instrument configuration

The roughness meter has a Liquid Crystal Display (LCD) with four lines. There are three couples of buttons on the right hand side of the LCD, which are used to assign values to the the three top lines (the fourth, bottom line is used to display status values) advancing or backspacing through a menu. There are four arrow keys (below the LCD) which are used to move the measuring knife. There is one main "action" key (on the left hand side below the LCD) whose action depends on what is selected on the LCD (but usually starts a measure).

The roughness meter can perform a measure in standalone mode (acquire a profile, filter it and compute some parameters), or transmit the data to the spare computer. In such case one should transmit raw data and do the filtering



and parameter computation via software on the spare computer . We describe here the recommended setup, as it shall be programmed on the LCD :

- Line 1 : SETUP
- Line 2 : ASSESSMENT MODE
- Line 3 : UNFILTERED
  
- Line 1 : SETUP
- Line 2 : FORM OPTIONS
- Line 3 : DATUM
  
- Line 1 : SETUP
- Line 2 : DATA DUMP
- Line 3 : PROFILE ONLY
  
- Line 1 : SETUP
- Line 2 : TRAVERSE LENGTH
- Line 3 : select here the length of the run in mm (suggested 25.4 mm)
  
- Line 1 : AUTO
- Line 2 : DATA DUMP
- Line 3 : AUTO ON
  
- Line 1 : AUTO
- Line 2 : AUTO REVERSE
- Line 3 : AUTO Off

The latter options means that the data dump to the spare computer starts automatically once the measure is over, while the knife will not return to the starting position automatically. This has been disabled because the knife may wear returning (at a speed higher than during measures) while in contact with the sample. One shall always return the knife when in raised position. A procedure to return it to the correct position in semi-automatic way is given below in 9.1.2.

### 9.1.2 Execution of measures

To execute a roughness measure place a sample in the vice. Handle samples very carefully, and measure them asap to avoid dust accumulation. Make sure that the maximum vertical excursion over the sample length (traverse length parameter set up above) is less than 2 mm (otherwise an overrun condition will occur). Also make sure that the measuring knife is well centered over the sample.

- make sure the line 1 of the LCD shows : SETUP
- use the screw to raise the measuring knife
- use the arrow key to move the measuring knife to the beginning of the measuring sample
- use the screw to lower slowly the measuring knife until it touches the sample (bottom line of LCD changes from \*\*\*\* to a numeric value) and regulate it so that the LCD shows an height as close as possible to 0 mm (a value between  $\pm 100$  mm is acceptable)
- press the main action button to start the measure

The LCD will show the percentage of measure in progress while the knife is moving. Then the roughness meter computes the parameters (only unfiltered parameters in the default set up), and then the data dump will start. The knife will revert to the starting position if you follow this procedure :

- click twice on the right-hand side button (arrows down) in the top row so that the line 1 of the LCD shows : AUTO
- use the screw to raise the measuring knife



- replace the sample
- press the main action button to return the knife to the starting position
- click once on the left-hand side button (arrows up) in the top row so that the line 1 of the LCD shows : **SETUP** to be ready for the next measure

### 9.1.3 Data acquisition

The data acquisition from the roughness meter to the spare computer occurs over a serial line, and is managed by a front end program called FTSP supplied by Taylor-Hobson Italia. This program can retrieve and save the data, and do some rough plotting and analysis. It can also pass control to the more sophisticated general purpose Mountain program.



The program is started clicking on the icon. The menu bar contains a **Setup** button which is used to change a few parameters in the configuration (modify only those described in 7.9.3). The main panel has three buttons on the left hand side, labelled **Measure**, **Analysis** and **Measure & Analysis**. The former starts an acquisition, the second transfers the data to the Mountain program, and the latter combines the two steps (recommended).

Once the acquisition is started, the program waits for the data dump to be started on the roughness meter side. This occurs automatically at the end of a measure. Hence the acquisition shall be active before the data dump starts. In case of dump failure, check cables and program, then restart the acquisition, and either repeat the measure, or dump the latest measure on the roughness meter side as follows

- use the LCD to obtain the following setting
  - Line 1 : **AUTO**
  - Line 2 : **DATA DUMP**
  - Line 3 : **AUTO OFF**
- press the main action key to start the dump
- when the dump is over set Line 3 back to **AUTO ON** , then Line 1 back to **SETUP**

Note that FTSP allows, among other functions, also to save the raw profile in ASCII format (.ftp extension)

### 9.1.4 Data analysis

The Mountain program is a general purpose profile analysis program, which can also be started manually, but which we assume is started by the FTSP program. It can read a number of file formats (in our case it will read the .pro files written by FTSP) and perform graphical and numerical analysis.

Once one has done a satisfactory sequence of operations on a file, and wants to repeat them systematically on other files, it is sufficient to save the current document as an .mnt file, and later apply it as a template.

For full details refer to the Mountains manual. We describe here only the main functions used to set up our default template.

The **File** menu contains an **Open Studiable** item which will show all .pro files in the current directory (C:/Roughness). This can be used to load a rough profile in a fresh page. There is also a **New document** item to clear the document, and an **Apply Template** item, plus other self explanatory items (e.g. **Print**, **Save**).

Note that applying a template works only on the first profile shown, and may cause the disappearance of such profile (if its raw plot is not part of the template).

There are **Operator** and **Studies** menus. They cause a box containing a plot or some text to appear on the screen. The box can be repositioned with the mouse, and its layout (font, colour) changed with various buttons. In our template we have applied :





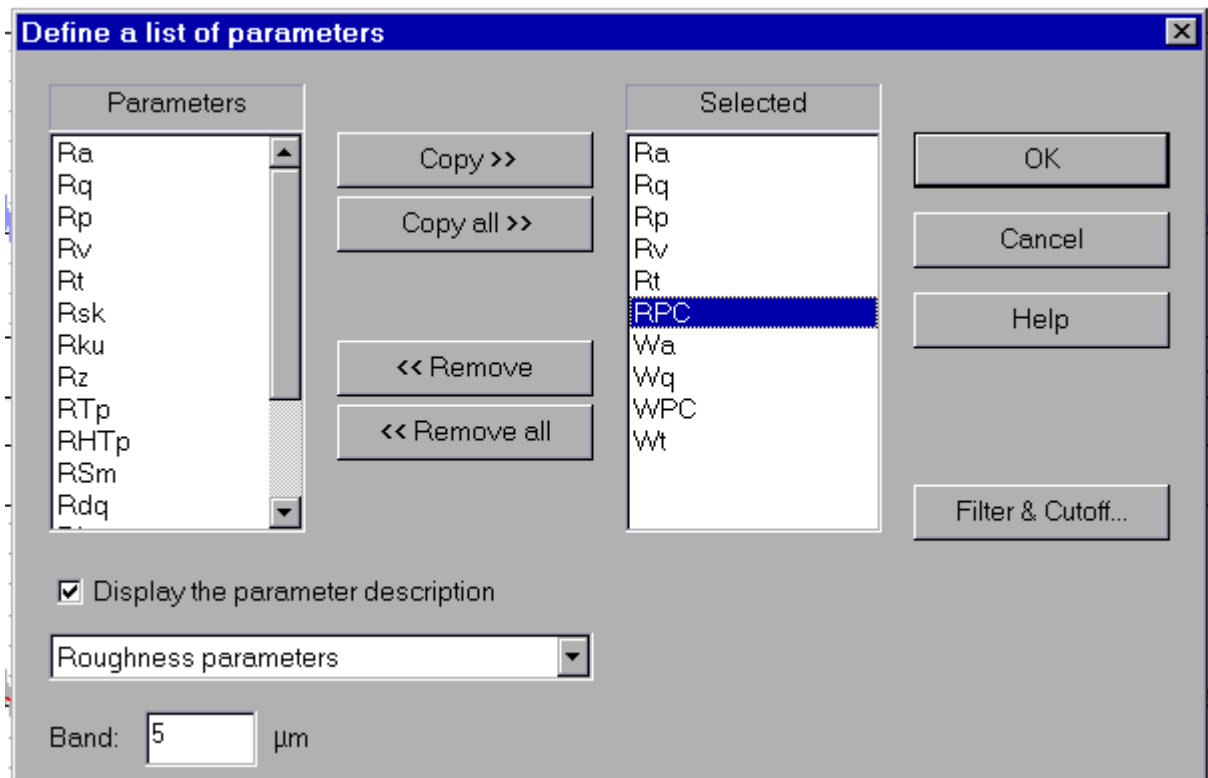
- the [Levelling](#) operator (which removes a linear trend, replacing the raw profile with a rectified one)
- the [Roughness and Waviness Profile](#) study which adds a smoothed plot
- the [Parameters](#) study, which prints a list of parameters. This list can be edited clicking on the frame, and then on



the menu bar button which resembles a pocket calculator. This will show a list of parameters in various categories, which can be added/removed to/from the list of parameters computed and shown. Note that some parameters (like the RPC and WPC peak counts) require further tuning. In such case click on the parameter in the [Selected](#) item list to obtain a further dialogue (this is shown in the figure below).

The meaning of the parameters is explained by the parameter description shown on the printouts. For more details refer to the Taylor-Hobson manuals. Note anyhow what follows :

- The key roughness parameters are Ra and the RMS Rq
- The Rt parameter may be affected by spurious (positive) peaks due to dirt (which will go away after suitable cleaning) or by (negative) notches due to improper handling of the samples.
- The Wt waviness parameter may be larger for measurement showing a curvature, which might be the case if the piece is distorted. In such case the roughness parameters around the curved profile will still be significant.
- All parameters allow individual setting of the filter and cutoff parameters (the ESO recommended value is a gaussian filter with a cutoff of 1 VIMOS pixel, which is about 0.12 mm). This is done selecting one parameter at a time in the Selected (right) pane (or more parameters with shift-Click to use the same cutoff), and clicking on the [Filter&Cutoff](#) button thus enabled.
- Some parameters (like RPC shown in figure) require the specification of a bandwidth (which has to be entered in the Band box which appears). The value to be entered is the *full* range (e.g. 5  $\mu\text{m}$  corresponding to +/- 2.5  $\mu\text{m}$  on the printout)
- The filter setting can be done also for the [Roughness and Waviness Profile](#) study, clicking on the rightmost button of the study.
- Recommended values will be stored in the standard template to be applied





Note that the Mountain program can, among several other functions, also be used to obtain an ASCII dump of any profile. Just select the particular profile clicking on it, and from the **File** menu select **Save Profile**, then use the UA2 format.

## 9.2 LPKF machine maintenance

The key reference for the routine maintenance of the StencilLaser machine is chapter 8 of [LPKF documentation \[ref. 8\]](#)., in particular section 8.2 therein gives a detailed schedule, followed by a description of the various tasks. We provide here only a revised checklist (ordered by frequency of intervention and integrated with non-LPKF items), plus some limited supplementary information (see [9.2.3](#)).

Action/Item	Frequency and detail of operation plus reference to <a href="#">LPKF doc. [ref. 8]</a>
	<i>daily</i>
Granite surface cleaning	daily before switch on (section 8.3.1)
Nozzle cleaning	clean with a drill point daily (section 8.5.1)
Compressed air fine filters check	(filters on back of laser machine) discharge daily just before switch off (section 8.3.2)
Pressure duplicator filter check	discharge daily
Dryer filter check	discharge daily
Exhauster extraction funnel check	clean and check daily (section 8.4.2)
	<i>weekly</i>
StencilMaster calibration	run about weekly (or whenever machine is off for a long time or whenever room temperature changes significantly) : see section. <a href="#">9.2.1</a> of this manual
Pressure duplicator tank check	once weekly open valve below tank and check there is no water
Clamping cams and bars greasing	grease weekly (section 8.3.4 and 8.3.5)
Felt contact ring check	check weekly and replace if worn (section 8.5.6)
Slider disc check	clean and check weekly (section 8.4.1) Note that the brass disc has been replaced by a Teflon one which should not wear significantly
	<i>when necessary</i>
Calibration of cutting optics	when material thickness is changed, or when slider disc or felt contact ring is replaced ? (see section 7.9 if ref. [8] for full procedure, plus complementary notes in <a href="#">9.2.2</a> here below)
Deionized water refill	(tank inside back of electronics rack) : refill when necessary (e.g. LCD message)
Chiller water refill	refill with distilled water when necessary (see level indicator outside)
Exhauster filter cleaning	clean with compressed air jet when safety (red) lamp (near exhauster switch close to laser head support) is lighted on. Replace when it cannot be cleaned any more
Laser lamp replacement	replace every 400 h as instructed (see 8.6)
Slider disc replacement	replace when worn (estimate for brass disc is about 500 h, see section 8.4.1; the Teflon disc should last much longer)
	<i>3-monthly</i>
Protecting lens inspection	clean and inspect every 500 h of operation or 3 months elapsed and replace if worn (see 8.5.2-3-4 for <i>full</i> procedure)
Focusing optics inspection	clean and inspect every 500 h of operation or 3 months elapsed (together with previous item, see also 8.5.5)
Exhauster filter replacement	replace every 500 h of operation or 3 months elapsed
Chiller fins cleaning	clean the fins of the air cooled condenser from dust and dirt (see KKT documentation). The fins are located behind the grid on the chiller side opposite to the filter (labelled "keep this side clean")
Chiller filter replacement	replace cartridge (LPKF suggests every 500 h of operation or 3 months elapsed, however with distilled water it is expected to be necessary much seldom, only when there is a significant pressure difference between the two manometers at entrance and exit from the filter)
Air filters (EWO) replacement	replace cartridges when pressure below 0.6 bar (manometer in red area, part numbers in EWO documentation)



Action/Item	Frequency and detail of operation plus reference to <u>LPKF doc. [ref. 8]</u>
	<i>long term</i>
Ballscrew spindles greasing	grease every 500-1000 h (section 8.3.3)
Exhauster extraction funnel replacement	replace when worn (about 1000 h, see section 8.4.2) Note this part has to be ordered separately (see section 11.5 of <u>LPKF doc. [ref. 8]</u> )
Nozzle replacement	replace every 1000 h
Particle filter (primary cooling circuit) replacement	(filter inside back of electronics rack) replace cartridge every 1000 h of operation or 6 month elapsed (section 8.7.1)
Pump chamber seal ring (laser source) replacement	replace every 2000 h of operation or 1 year elapsed (section 8.6.4)
Dryer filter replacement	every 2500 h of operation or 1 year elapsed verify presence of impurities in drying material or in drying tank. The drying material lifetime is typically 8000 h. For further info refer to Zander documentation.



## 9.2.1 Calibration of the positioning system

This calibration is done under control of the StencilMaster software, and updates the tables used to compensate the commanded motions of the platform of the StencilLaser machine for known deviations of the ballscrew spindles, using the readout of the Heidenhain rulers.

- the calibration of the positioning system shall be done weekly (or after the machine has been kept off for several days or whenever there is a substantial change in room temperature) and is done using either the [Configuration=>Calibrate](#) item (this item is normally ticked to indicate a set of calibration files is in use) or the [Configuration=>Measure](#) item (which will be entered anyhow). Press the [Calibrate](#) button, then reply [OK](#) to the popup to command an automatic calibration scan of both axes and update all necessary files, and exit with [OK](#).
- The [Configuration=>Measure](#) item allows to follow the calibration with the readout of the Heidenhain rules. It can also be entered at any time (and left with [Cancel](#) without doing anything) to see the readout of the current position (this should be in the range from 0 to 322 mm for X and from 0 to 604 mm for Y, since coordinates are relative to the reference point in the bottom left corner : if values are totally out of range, as it might occurs when the end stops are hit, or after a long switch off, it is advised to re-do a calibration) .
- after a succesful calibration there will be three files (measure\*.\*) updated in C:\LPKF30\Smaster32. Of these the two txt files are self-explanatory ASCII files which can be archived elsewhere and eventually plotted if it is desired to keep an history of the calibrations for quality control purposes. These activities are however outside of the scope of the present document.

## 9.2.2 Calibration of cutting optics

This operation is an adjustment of the focus of the laser head in order to preserve the correct quality of the cut, and has to be done whenever the head is dismounted, or the slider or felt rings are replaced. The basic procedure is described in section 7.9 of [LPKF documentation \[ref. 8\]](#), however a finer procedure is recommended for optimization of the cut (roughness) quality :

- first perform the preliminary calibration according to LPKF documentation
- then cut the test pattern in file C:\LPKF30\Data\mu300.job (job files must be read in StencilMaster with the [File=>Open](#) button) : if the cut is regular without burns proceed, otherwise repeat the preliminary calibration
- move the micrometer up and down by 0.05-0.06 mm, and for each position (central, up and down) cut one sample of the type described in section [7.9](#)
- measure the roughness of the samples as described in [7.9.3](#) and choose the optimal position (eventually adjust the micrometer to an intermediate position and repeat roughness measurement for a new sample)

## 9.2.3 Supplementary information

- the content of the standard spare kit plus additional spare parts is given below (additional spare parts to be ordered separately are listed in section 11.5 of [LPKF documentation \[ref. 8\]](#)) This list reflects the situation at July 2000 and is not updated for actions done during the installation in August 2000 or later by LPKF or ESO personnel
  - 4 laser lamps
  - 2 spare lamp contacts
  - laser focusing unit and protective "lens"
  - 1 pair of goggles (must be weared when laser cover is removed)
  - paper for cleaning optical parts
  - 1 steel nozzle (plus an additional 1)
  - drill points for nozzle cleaning
  - 3 felt rings (plus 1 additional felt and 1 brass ring) for exhauster
  - 1 O-ring
  - assorted tools
  - particle filter cartridge for primary cooling circuit filter (filter unit inside control rack)
  - spare filter unit (blue case) for chiller (contains one spare cartridge)
  - 1 additional filter cartridge for chiller filter
  - complete set of "rapid" taps for chiller
  - 2 380 V industrial plugs



- the consumables necessary for operations are
  - Invar sheets : a supply for about 2 years of mask production will be delivered
  - de-ionized water for the primary cooling circuit (small tank inside LPFK rack) : will *not* be delivered and has to be supplied in situ at Paranal
  - distilled water for the secondary cooling circuit (in external chiller, about 60 l) : will *not* be delivered and has to be supplied in situ at Paranal
  - anti-frost additive to be added to the secondary cooling circuit water (this is an *una tantum* addition when filling the chiller tank for the first time, will *not* be delivered and has to be supplied in situ at Paranal )
  - grease (grease with Li powder will be delivered with and for the IC robot, see [9.3](#), but can be used also for the StencilLaser motion system ballscrew spindles)
- the replacement of the lamp will be signalled by a StencilMaster message and has to be performed according to [section 8.6.2. of ref.8](#). The completion of the operation has to be signalled to StencilMaster clicking on [Configuration=>Laser subsystem=> New lamp](#). The current lamp usage time is shown by the LCD on the control rack using the following sequence of front panel buttons : **F1 SHOW**, **F4 MORE**, **F1 OP.HOURS**. The numeric counter close to to the SYS ON button indicates the total on time instead.
- besides the filling of the cooling water in the internal cooling circuit (tank in LPKF rack) according to [section 8.2. of ref. 8](#) also the water in the external cooler shall be kept under control
- besides the routine checks on the standard air filters according to [section 8.3.2. of ref. 8](#), particular care shall be given to water contamination, for the prevention of which a dryer has been added to the filter sets. All filters shall be anyhow discharged at switch off or on. Make sure no water is present : **presence of water in the air circuit will cause serious damage to the optical system !** Filter cartridges shall be replaced when the filter manometer is on the red.
- An alarm lamp near the laser head (close to the exhauster button) indicates when the filter inside the exhauster has to be replaced.

### 9.3 IC robot and SC maintenance

The SC is a passive piece of furniture which requires no particular maintenance. In order to ensure cleanliness of the masks it is however recommended to keep the doors of the SC closed whenever a quadrant is not in use.

The IC holder is also a passive piece of furniture which requires no particular maintenance.

The Z-displacement unit of the MHS robot shall require no particular maintenance (refer to [Antil documentation \[ref. 10\]](#)), except the greasing every 6 months of the sliding guides. A grease with Li powder shall be used, and shall be inserted in the provided units at the lower end of the sliding guides.

Keep the barcode reader exit windows clean.



## 9.4 Computer maintenance

The computer hardware does not require any unusual maintenance besides normal custom. Refer to Maintenance Manual section 12 for LRU redundancy policy

The only maintenance activities related to data files can be the *removal of old logs* (which are the daily ops log files located respectively in directory D: on MHCU and E: on MMCU, plus the ever growing StencilMaster log in C:\Lpkf30\Smaster32\Smaster.log on MMCU formerly in C:\Lpkf30\Bmaster\Bmaster.log) and the *backup of sensitive files* **only** in the case they are modified with respect to the delivered version. A non-exhaustive list includes :

- the configuration file D:\Mhs\instrument\configure.txt on MHCU
- the files in directories D:\Mhs\instrument\archive, calib, contours, holes, refpins and robot on MHCU
- the files in directory E:\instrument\Permanent on MMCU
- the .bat files in C:\Mirror on MMCU and MHCU
- the C:\Lpkf31\New\_templates\default.cat file on MHCU (see Maintenance Manual section 8.1)
- the C:\Lpkf30\Smaster32\sl600dcs.mch, stencil.phs and stencil.bls files on MMCU (see Maintenance Manual section 8.2)
- the \*.cfg files in C:\Barcode\Winhost on spare (see Maintenance Manual section 9.4)

## 9.5 Software maintenance

All software is delivered pre-installed on the three computers as appropriate. In addition to this the original distribution CDs or floppies are delivered (for third party software) and a custom CD will be delivered (for IFCTR software and with a backup version of all modifications and additions to the third party software). Re-installation shall be necessary **only** in case of major computer failures (requiring hardware reinstallation) : in such case make reference to the Maintenance Manual.

No other maintenance action is foreseen.