Visible & Infrared Survey Telescope for Astronomy

IR CAMERA

Document Title: IR Camera Software User and Maintenance Manual

Document Number: VIS-MAN-ATC-06080-0020

Issue: 3.0

Date: 18 August 2008

Document Prepared By:	**S.M. Beard** **IR Camera Software Manager** **S.A. McLay** **Software Engineer**	**Signature and Date:**	
Document Approved By:	**Martin Caldwell** **IR Camera Systems Engineer**	**Signature and Date:**	
Document Released By:	**Kim Ward** **IR Camera Project Manager**	**Signature and Date:**	
Document Reviewed By:	**Gavin Dalton** **IR Camera Scientist**	**Signature and Date:**	

CHANGE RECORD

Issue	Date	Section(s) Affected	Description of Change/Change Request Reference/Remarks
1.0	12/11/03		FDR version
2.0	01/02/07	All	First Paranal release version.
3.0	18/08/08	All	Updated following VISTA commissioning at Paranal.

NOTIFICATION LIST

The following people should be notified by email that a new version of this document has been issued and is available on the IR Camera document database:

RAL:	Martin Caldwell
	Gavin Dalton
	David Terrett
ATC:	Steven Beard
VPO:	Malcolm Stewart
	Will Sutherland
QMU	Jim Emerson
ESO Paranal	Nick Kornweibel

TABLE OF CONTENTS

1 INTRODUCTION

1.1 Purpose

This document describes the installation, operation and maintenance of the VISTA IR Camera software. It provides an introduction for observers to the operation of the VISTA IR camera. It is based on the document template provided in [AD3]. The VISTA IR Camera is described in [RD1].

This document will also become the primary reference which Paranal software engineers will consult for software operations and maintenance after commissioning.

1.2 Scope

This document covers only the control software for the VISTA IR Camera (VIRCAM). The VISTA Data Flow pipeline is described elsewhere (e.g. [RD17], [RD18] and [RD19]), as are the Exposure Time Calculator, [RD16], and the VISTA Survey Area Definition Tool observation planning software (e.g. [RD14] and [RD15]). The operation of the VISTA telescope is described in [RD13].

This document includes a summary of how to operate the VISTA High Order Wavefront Sensing (HOWFS) software, but it does not describe the wavefront analysis process itself; nor does it describe the Low Order Wavefront Sensing system, which is controlled by the VISTA TCS An introduction to the VISTA wavefront sensing software may be found in [RD12] and [RD9] and references therein.

1.3 Applicable Documents

[AD1] *VISTA IR Camera Software Requirements*, VIS-SPE-ATC-06080-0010, Issue 2.2, 12 January 2004.

[AD2] *VISTA Instrument Software Requirements,* VIS-SPE-ATC-00150-0003, Issue 2.2, 25 July 2002.

[AD3] *VLT Software Template Instrument Software User & Maintenance Manual*, VLT-MAN-ESO-17240-1973, Issue 5, 13 January 2005.

1.4 Reference Documents

1.4.1 VISTA IR Documents

[RD1] *VISTA IR Camera System Description*, VIS-SPE-RAL-06013-0001, Issue 2.0, November 2003.

[RD2] *VISTA IR Camera System Block Diagram,* VIS-DES-RAL-06013-9001, Issue 1.5, 9 March 2005.

[RD3] *VISTA IR Filter Wheel Control,* VIS-MAN-ATC-06080-0020, Issue 2.2, 29 August 2005.

[RD4] *VISTA IR Camera Observation Software Design Description,* VIS-DES-ATC-06084-0001, Issue 3.4, 17 June 2005.

[RD5] *VISTA IR Camera Instrument Control Software Design Description,* VIS-DES-ATC-06083-0001, Issue 2.2, 19 November 2007.

[RD6] *VISTA IR Camera Low Order Wavefront Sensor Software Design Description,* VIS-DES-UOD-06048-0001, Issue 3.2, 27 April 2007.

[RD7] *VISTA IR Camera High Order Wavefront Sensor Software Design Description,* VIS-DES-UOD-06048-0002, Issue 3.5, 27 April 2007.

[RD8] *VISTA IR Camera Autoguider Software Design Description (LCU part),* VIS-DES-UOD-06048-0003, Issue 1.2, 27 April 2007.

[RD9] *VISTA IR Wavefront Sensing and Autoguiding Software Overview,* VIS-TRE-UOD-06048-0004, Issue 1.1, 26 June 2006.

[RD10] *Image Analysis Algorithm for VISTA Wavefront Sensing,* VIS-DES-UOD-06048-0005, Issue 1.0, 12 Nov. 2003.

[RD11] *VISTA IR Camera Software Acceptance Test Plan*, VIS-PLA-ATC-06087-0001, Issue 1.6, 19 November 2007.

1.4.2 VISTA TCS Documents

[RD12] *VISTA Active Optics and Guiding Workstation Software Design*, VIS-SPE-RAL-13030-0003, Issue 2.7, 28 March 2006.

[RD13] *VISTA Telescope Control System User Manual*, VIS-MAN-ATC-?????-????, TBD.

1.4.3 VISTA Observation Planning Documents

[RD14] *VISTA Requirements for Surveys: Planning, Scheduling and Progress*, VIS-SPE-QMU-20000-0007, Issue 1.0, 17 June 2004.

[RD15] *VISTA Survey Definition and Progress Tools: Functional Specification,* VIS-SPE-ATC-20500-0001, Issue 1.0, 17 November 2004.

[RD16] VISTA IR Camera Exposure Time Calculator Specification, VIS-SPE-IOA-20000-0009, Issue 1.0, 10 January 2005.

1.4.4 VISTA Data Flow Documents

[RD17] *VISTA IR Camera DFS System Impact,* VIS-SPE-20000-0001, Issue 1.2, 9 May 2005.

[RD18] *VISTA IR Camera Calibration Plan,* VIS-SPE-20000-0002, Issue 1.3, 13 December 2005.

[RD19] *VISTA IR Camera Data Reduction Specifications,* VIS-SPE-20000-0003, Issue 1.0, 15 December 2004.

1.4.5 General Documents

[RD20] *Lakeshore Model 218 Temperature Monitor User's Manual*, Lake Shore Cryotronics Inc. (http://www.lakeshore.com), Revision 1.8, 27 August 2002.

[RD21] *Lakeshore Model 332 Temperature Controller User's Manual*, Lake Shore Cryotronics Inc. (http://www.lakeshore.com), Revision 1.2, 27 August 2002.

[RD22] *Pfeiffer Vacuum TPG 256 A Operating Manual*, Pfeiffer Vacuum GmbH (http://pfeiffer-vacuum.de), BG 805 186 BE (9907).

1.4.6 ESO-VLT Documents

[RD23] *VLT Data Interface Control Document*, GEN-SPE-ESO-19400-0794, Issue 3.0, 1 February 2005.

[RD24] *VLT Software ICD between the VLT Control Software and the VLT Archive System*, VLT-ICD-ESO-17240-19400, Issue 2.0/6.

[RD25] *VLT Software ICD between Instrumentation Software and VLT Archive System*, VLT-ICD-ESO-17240-0415, Issue 1.0, 14 Sept. 1995.

[RD26] *VLT Software Programming Standards*, VLT-PRO-ESO-10000-0228, Issue 1.0, 10 March 1993.

[RD27] *VLT Software Basic Tools and Working Environment Guidelines*, VLT-MAN-ESO-17000-2972, Issue 2, 30 April 2004.

[RD28] *VLT Paranal Network/Computers/Consoles Description*, VLT-SPE-ESO-17100-3439, Issue 1, 3 March 2005.

[RD29] *VLT Instrument Software Specification*, VLT-SPE-ESO-17212-0001, Issue 5, 30 September 2005.

[RD30] *VLT INS Common Software Specification,* VLT-SPE-ESO-17240-0385, Issue 4, 13 January 2005.

[RD31] *VLT Software ICD between the VLT Control Software and the Observation Handling System*, VLT-ICD-ESO-17240-19200, Issue 1.3, 7 June 2000.

[RD32] *VLT Common Software Overview*, VLT-MAN-ESO-17200-0888, Issue 1.0, 17 August 1995.

[RD33] *VLT Problem Report and Change Request User Manual,* VLT-MAN-ESO-17200-0981, Issue 2.0 16 October 1998.

[RD34] *VLT Environment Common Configuration User Manual,* VLT-MAN-ESO-17210-0855, Issue 3.0, 31 October 1998.

[RD35] *VLT Central Control Software (CCS) User Manual,* VLT-MAN-ESO-17210-0619, Issue 3, 16 December 2005.

[RD36] *VLT Tools for Automated Testing (TAT) User Manual,* VLT-MAN-ESO-17200-0908, Issue 1.4, 15 February 2001.

[RD37] *VLT Configuration Management Module (CMM) User Manual*, VLT-MAN-ESO-17200-0780, Issue 2.0, 22 October 2001.

[RD38] *VLT LCU Common Software User Manual*, VLT-MAN-SBI-17210-0001, Issue 4, 21 December 2005.

[RD39] *VLT IRACE-DCS User Manual*, VLT-MAN-ESO-14100-1878, Issue 1.7, 25 January 2006.

[RD40] *VLT CCD Detectors Control Software — User Manual*, VLT-MAN-ESO-17240-0672, Issue 2, 25 September 1998.

[RD41] *VLT Common Software Scientific Linux 4.0 Installation Manual,* VLT-MAN-ESO-17200-2009, Issue 4, 7 March 2006

[RD42] *VLT Common Software Installation Manual*, VLT-MAN-ESO-17200-0642, Issue 5, 12 January 2006.

[RD43] *VLT Software Installation Tool for VLT Software Packages (pkgin) — User and Maintenance Manual*, VLT-MAN-ESO-17240-1913, Issue 5, 31 December 2005.

[RD44] *VLT Software Template Instrument Software User & Maintenance Manual*, VLT-MAN-ESO-17240-1973, Issue 5, 13 January 2006.

[RD45] *VLT High Level Operating Software (HOS) / Broker for Observation Blocks (BOB) User Manual,* VLT-MAN-ESO-17220-1332, Issue 5, 27 October 2005.

[RD46] *VLT INS Common Software / Base ICS (icb) User Manual*, VLT-MAN-ESO-17240-0934, Issue 6, 31 December 2005.

[RD47] *VLT INS Common Software / Base ICS Control Panel (icbpan) User Manual*, VLT-MAN-ESO-17240-2606, Issue 4, 9 December 2005.

[RD48] *VLT Software Base Observation Software Stub (BOSS) User Manual,* VLT-MAN-ESO-17240-2265, Issue 5, 8 December 2005.

[RD49] *VLT INS Common Software for Templates — User Manual*, VLT-MAN-ESO-17240-2240, Issue 5, 15 December 2005.

[RD50] *VLT INS Common Software Configuration Tool (ctoo) — User Manual*, VLT-MAN-ESO-17240-2235, Issue 5, 31 December 200.

[RD51] *VLT INS Common Software Startup Tool (stoo) — User Manual*, VLT-MAN-ESO-17240-2153, Issue 6, 28 August 2005.

[RD52] *VLT Motor Engineering Interface User Manual*, VLT-MAN-ESO-17210-0669, Issue 1.6, 2 October 1998.

[RD53] *VLT CCS Engineering Interface and Graphical Tools User Manual,* VLT-MAN-ESO-17210-3816, Issue 2.0, 12 February 2007.

1.5 Abbreviations and Acronyms

ACC	Access and Configuration Control
AD	Applicable Document
ADC	Analogue to Digital Converter
AG	Auto Guiding or Auto Guider
AIT	Assembly Integration and Test
aO	Active Optics
ASM	Astronomical Site Monitor
ATC	(UK) Astronomy Technology Centre
BOB	Broker for Observation Blocks
BOSS	Base Observation Software Stub
CCD	Charge Coupled Device
CCS	Central Control Software
ccsei	Central Control Software Engineering Interface
CDT	Command Definition Table
CIT	Command Interpreter Table
CLDC	(IRACE) Clock Converter and DC voltage generator
CMM	Configuration Management Module
COI	Co-Investigator
CS	Curvature Sensor/Sensing
ctoo	Configuration tool

CWP	Current Working Point
DB	Database
DCS	Detector Control Software
Dec	Declination
DET	DETector software package
DFE	Detector Front-end Electronics
DFS	Data Flow System
DHS	Data Handling Server
DIC	DICtionary
DICB	ESO Data Interface Control Board
DICD	Data Interface Control Document
DID	Data Interface Dictionary
DMA	Direct Memory Access
DR	Data Reduction
ECCS	Extended CCS
EMC	Electro-Magnetic Compatibility
EMI	Electro-Magnetic Interference
ESO	European Southern Observatory
ETC	Exposure Time Calculator
FDR	Final Design Review
FITS	Flexible Image Transport System
FOV	Field Of View
FPA	Focal Plane Array (thermal plate)
FWHM	Full Width at Half Maximum
GS	Guide Star
GUI	Graphical User Interface
HDU	Header Data Unit = FITS Header + Data Unit
HOS	High Level Operating Software (also referred to as "High Level Operational Software")
HOCS	High Order Curvature Sensor
HOWFS	High Order Wavefront Sensor
HW	Hardware
ICB	Base ICS
ICD	Interface Control Document
ICS	Instrument Control Software
ID	Identifier
INS	INstrument Software package
IR	Infrared
IRACE	Infrared Array Control Electronics
IRTD	IRACE Real Time Display
ISF	Instrument Summary File
IWS	Instrument Work Station
I/O	Input Output
LAN	Local Area network
LCC	LCU Common Software
lccei	LCU Common Software Engineering Interface
LCU	Local Control Unit (normally a VME/VxWorks system)
LOCS	Low Order Curvature Sensor
LOWFS	Low Order Wavefront Sensor
LSC	Lakeshore Controller
LSM	Lakeshore Monitor
M1	Primary mirror
M2	Secondary mirror
MCM	Motor Control Module
MIDAS	Munich Image Data Analysis System

motei	MOTor Engineering Interface
MS	Maintenance (and verification) Software
N/A	Not Applicable
NC	(IRACE) Number Cruncher
NFS	Network File System
OB	Observation Block
OB top	Optical Bench top
OBD	Observation Block Description (File)
OBS	OBServation
OLAC	On Line Archive Client
OLAS	On Line Archive System
OLDB	On Line Database
OS	Observation Software
OSLX	Objective SLX
P2PP	Phase II Proposal Preparation tool
PAF	(ESO-VLT) PArameter File
PCF	(ESO-VLT) Point Config File
PECS	Pluggable Environment Contribution System
PI	Principal Investigator
PID	Proportional Integral Derivative (controller)
PSF	Point Spread Function
QC	Quality Control
QC0	Quality Control level zero
QC1	Quality Control level one
QMU	Queen Mary University (London)
RA	Right Ascension
RAL	Rutherford Appleton Laboratory
RD	Reference Document
RMS	Root Mean Square
REF	Reference Setup File
RTAP	Real Time Application Platform (from Hewlett Packard)
RTD	Real Time Display
SADT	(VISTA) Survey Area Definition Tool
SEQ	(Template) Sequencer Script File
SHF	Short Hierarchical Format
SLX	Setup files and operation Logs handling
SPR	Software Problem Report
stoo	Startup tool
SW	Software
TAT	Tools for Automated Testing
TBC	To Be Confirmed
TBD	To Be Decided
TCCD	Technical CCD controller
TCL	Tool Command Language
TCP/IP	Transmission Control Protocol/ Internet Protocol
TCS	Telescope Control Software
TIF	Telescope Interface
TPL	TemPLate
TSF	Template Signature File
TSFX	Extended Template Signature File
UIF	(Portable) User Interface (Toolkit)
UK	United Kingdom
UKATC	United Kingdom Astronomy Technology Centre
UML	Unified Modelling Language

UT	(ESO-VLT) Unit Telescope (not to be confused with "Universal Time")
UTC	Universal Time (Coordinated)
vcc	VLT common configuration
VCCDB	VLT Common Configuration Database
VDFS	VISTA Data Flow System
VISTA	Visible and Infrared Survey Telescope for Astronomy
VLT	Very Large Telescope
VME	Versa Module Eurocard (a widely used computer bus)
VOLAC	VCS OLAC Client
VPO	VISTA Project Office (Management Organisation within the UKATC)
WAN	Wide Area Network
WCS	World Coordinate System
WFS	Wavefront Sensor
WS	(Unix) Work Station (and by implication Unix servers)

1.6 Glossary

Acquisition Template: A specific operations *template* used to position the telescope and instrument correctly for acquiring a target. The first template of an *Observation Block* involving target acquisition must be an acquisition template. See *template*.

Alias: An alternative (short) name for an ESO-VLT standard keyword.

Alias Conversion Table: A table containing the association between a Short-FITS keyword and its alias. See also "*Translation/Alias Table*".

aO star: An "active optics" reference star used by a VISTA IR wavefront sensing subsystem; either the High Order Wavefront Sensor (see "*HOWFS star*") or Low Order Wavefront Sensor (see "*LOWFS star*").

Active Optics (aO): The system responsible for maintaining the correct figure in the telescope optics and compensating for gravity vector and low frequency wind pressure etc. The active optics system can correct the M1 and M2 figures open-loop using lookup tables generated by the High Order Wavefront Sensor (HOWFS). Corrections are made every few seconds to the baseline from the lookup tables by measuring the wavefront error derived from the images of two stars with the Low Order Wavefront Sensors (LOWFSs).

Attribute: A data item contained inside a database *point* which is used to characterise its status. In object oriented terminology attributes correspond to "data members". See also "*point*".

Autoguider (AG): The telescope system responsible for generating telescope tracking corrections by measuring the drift in the centroid of a guide star image. The system can correct for telescope tracking errors and high frequency components such as seeing translation and wind shake.

Auxiliary Configuration File: A supplementary *configuration file* which describes the various elements and positions for each function of a particular instrument.

Bad Pixel Mask: A calibration frame mapping the location of bad pixels on each detector. The data processing pipeline uses this information to ignore data from bad pixels.

Bias frame: See "*Reset frame*".

Branch: The sub-tree of a hierarchical database identified by a unique root point. A branch is a logical collection of related points. See also "*point*".

Branch Configuration File: A file containing a description of a database branch. It is converted by the dbl tool into a *Point Configuration File*.

Calibration Frame: A frame used in the process of data reduction to remove the instrument signature from observations, or to provide astrometric or photometric calibration.

Cold Blocker (VISTA): A cold, opaque object inserted into the filter wheel of the infrared camera to prevent external light from reaching the detector. The DARK and SUNBLIND filters are used for this purpose. If available, the DARK filter is better suited as a cold blocker since it has a black surface and will stay colder.

Configuration file: A file containing a description of an application. There are types of configuration file: An *instrument configuration file* describes the current configuration of an instrument. A *reference configuration file* describes all the functions available for an instrument, and an *auxiliary configuration file* describes the various elements and positions for each instrument function.

Configuration set: A set of configuration files that provide the configuration for a specific ESO-VLT application or package.

Dark Frame: A calibration frame taken using a *cold blocker* filter with the same exposure time as the science observation it is intended to calibrate. The dark frame (DPR.TYPE=DARK) contains the integration of the detector dark current signal for the given exposure time, and can be subtracted from the science frame.

Data Flow System (ESO-VLT Software): The system that handles the flow of scientific data and information for the ESO-VLT. It includes subsystems for proposal handling, observation handling, science archiving, data pipeline and quality control. (Not to be confused with Yourdon/De Marco data flow diagrams).

Data Interface: A set of definitions that describe the contents of the VISTA or ESO-VLT data products.

Data Interface Dictionary (DID): A computer readable dictionary which defines all the terms used by the ESO Data Flow System to describe, for example, the meaning of keywords in the FITS header, configuration files and setup files.

Datum: (In the context of motor control) A reference point used to define a known location for a mechanism controlled by a stepper motor. Also known as the “index” or “reference point”.

Datuming: The action used by a mechanism to find the datum/index/reference point. Also known as “indexing”.

Detector Control Software (DCS): The control software responsible for sequencing the detector hardware, controlling a shutter (if any) and reading out data.

Detector Front-End Electronics (DFE): The electronics located near the detector and normally isolated from the instrument.

Detector setup file: A *setup file* containing a subset of setable parameters required for an exposure relating to the configuration of the detector.

Engineering User Interface (ESO-VLT software): This software module allows the user to send commands to and receive replies from any process on any node in the ESO-VLT software environment.

Error and Alarm System (ESO-VLT software): This software module allows the logging and display of error messages and the management of alarms.

Exposure: An exposure is the basic observation unit for the *Observation Software*. In the case of the VISTA IR camera, an exposure produces 16 raw data frames, which are merged into a single multi-extension FITS file. There may be more than one exposure described in an *Observation Block*. The exposure is the stored product of many (NDIT) individual *integrations* which have been co-added in the IRACE Data Acquisiton System. Each exposure is associated with an exposure time.

Flat-field Frame: A calibration frame containing an exposure of a uniform illumination, allowing the relative sensitivities of the detector pixels and any vignetting in the optical system to be calibrated.

Frame: A data unit which combines all relevant information with the scientific data from one *exposure*.

Filter (VISTA): *Either,* an individual filter made from suitably coated glass when referred to in the context of the instrument design; *or*, the name of an entire filter tray when referred to in the description of a science observation.

Filter tray (VISTA): A container into which an array of filters may be installed. For the VISTA IR science filters a tray contains a 4 x 4 array of filters — each covering a single detector. Once installed, the filters are fixed in place and the tray regarded as a single unit.

Filter slot (VISTA): A position on the filter wheel into which a filter tray may be installed. The slot numbers on the filter wheel are always the same, but the filter trays may be loaded and unloaded and moved to different slots.

Guide star: A star used by the VISTA IR autoguider subsystem.

HOWFS star: A star used by a VISTA IR High Order Wavefront Sensor.

Instrument Control System (ICS): The control system responsible for controlling the instrument hardware and sensing the instrument environment. (This is the equivalent of the Components Controller for a Gemini instrument, for example).

Instrument mode: A distinct instrument operating mode (such as "IMAGING" or "SPECTROSCOPY") used in configuring the Observation Software.

Instrument package: The collection of files shared between ESO/VLT instrument software and observation preparation software. Typically, the instrument package from each instrument is distributed to all users of P2PP.

Instrument path: A specific instrument light path (such as "INFRARED" or "OPTICAL") used in configuring the Observation Software.

Instrument setup file: A *setup file* containing a subset of setable parameters required for an exposure relating to the configuration of the instrument.

Instrument Software (INS): All the software associated with instrument control.

Instrument Summary File (ISF): A file containing a description of all the configurable elements of an instrument.

Instrument Workstation (IWS): A workstation which is assigned at startup to control instrument and/or detector LCUs.

Integration: A time interval during which a detector is collecting data. A simple snapshot, within the Data Acqusition System, of a specified elapsed time (DIT) in seconds. This elapsed time is known as the integration time. An *exposure* can be made from one or more integrations.

Integration data/Integration frame: The data resulting from a single detector integration. An example of *transient data*. One or more integration frames may be combined to make the *raw data* for a single exposure.

Intermediate filter (VISTA): A filter or filter tray with a smaller than normal size, positioned mid way between the usual slots on the filter wheel. These filters only cover a few of the science detectors but they enable engineering and calibration observations to be made. The HOWFS beam splitters are examples of an intermediate filter.

Jittering (VISTA): The process of taking a pattern of overlapping *exposures,* each shifted by a small *telescope movement* (<30 arcsec) from the reference position. Unlike a *microstep* the non-integral part of the shifts is any fractional number of pixels. Each position of a jitter pattern can contain a *microstep* pattern. Successive observations made while jittering can share the same guide and LOWFS stars, as long as the telescope movement is smaller than the field of view of the autoguider detector. Jittering is typically used by the IR Camera to make a sky-flat or to eliminate bad pixels. See also "*offsetting* and *micro-stepping"*.

Logging System (ESO-VLT software): This software module provides facilities to keep trace of important events occurring in the ESO-VLT software environment.

LOWFS star: A star used by a VISTA IR Low Order Wavefront Sensor. (Unlike the ESO-VLT, the VISTA IR Camera wavefront sensors use different stars to the one used by the autoguider — see "*Guide star*").

Message System (ESO-VLT software): This software module provides a homogeneous inter-process communication mechanism within the ESO-VLT environment.

Metadata: Additional information, such as the date, time, object name, telescope pointing information, etc. provided alongside the images generated by the camera to describe those images, facilitating the reduction and scientific interpretation of those images. Metadata is typically provided in the data header but might also be provided in auxiliary files such as an observation log.

Micro-stepping (VISTA): The process of taking a pattern of overlapping exposures, each shifted by a very *small telescope movement* (<3 arcsec) from the reference position. Unlike a *jitter* the non-integral part of the shifts are specified as 0.5 of a pixel, which allows the pixels in the series to be interleaved in an effort to increase sampling. A microstep pattern can be contained within each position of a *jitter* pattern. Micro-stepping is very similar to *jittering*, in that the same guide and LOWFS stars can be used. See also "*offsetting"*.

Module (ESO-VLT software): A module is a major subdivision of a software package (e.g. instrument control module, vci, or observation software module, vco). A module is divided into *units*.

Observation: A series of related *exposures* involving a single target/area.

Observation Block (OB): An Observation Block is the smallest schedulable observational unit for the ESO-VLT and VISTA. It contains all the information necessary to execute, in sequence and without interruption, a set of related *exposures* involving a single target (i.e. a single *telescope preset*). It contains one or more *template calls*; i.e. it describes which *templates* to call and the parameters to supply with each *template*. An Observation Block may include only one *acquisition template* giving target details, followed by one or more calibration or observation templates. Observation Blocks also contain scheduling and pipeline reduction requirements. A VISTA survey OB will normally contain one *acquisition template* followed by one *tile* template.

Observation Block Description (OBD): An ASCII file describing the contents of one or more *observation blocks*. The descriptions include an observation block identifier, all the template calls associated with each observation block and any other information stored in the observation block which needs to be included in the FITS header of the generated data.

Observation Frame: The data product (raw frame) containing the result of an observation. Different instrument modes normally produce different observation frames.

Observation Planning: The process of specifying a set of observations well in advance and submitting the specification as an observing programme.

Observation Scheduling: The process of examining all the outstanding *observation blocks* and converting them into an observing plan. Also the process of deciding, from current observing conditions, which *observation block* is the best one to be executed next.

Observation Sequence: A series of observations which are described with the *Sequencer* syntax.

Observation Software (OS): The software responsible for coordinating the telescope control system, instrument control system and detector control system.

Observing Programme: A programme of observations on various targets to answer a specific scientific or technical question, submitted by the same principal investigator. An Observing Programme can result in one or more *observation blocks*. For VISTA, an Observing Programme could be the specification of an entire survey.

Offsetting (VISTA): The process of taking several *exposures* at different telescope positions whose differences are larger than or comparable with the size of a detector. Each exposure will require a different set of guide and LOWFS stars. Offsets are used by the IR Camera to fill in the large gaps between detectors. See also "*jittering*" and "*tile*".

On-line Archive System (OLAS): A software system for automatically writing the data generated by an instrument to an on-line archive.

On-line Database (ESO-VLT software): This provides a mechanism to organise, store and share data within the ESO-VLT environment.

Package (ESO-VLT software): One of the major functional software packages of the ESO-VLT software, such as Telescope Control Software, Instrument Software, Remote Operations Software, High Level Operations Software and Central Control Software. A package is divided into *modules*.

Parameter File Format: The format of a Parameter File (PAF). Same as "*Short Hierarchical Format (SHF)*".

Partial setup file: A *setup file* containing only a subset of the setable parameters required for an exposure.

Pawprint (VISTA): A set of non-contiguous exposures from the 16 non-contiguous detectors of the VISTA IR Camera. Six pawprints at different telescope offsets need to be combined to make a contiguous *tile*, in which almost all the sky area covered has been observed at least twice.

Phase I: The first phase of an ESO-VLT observing project, in which a science case is proposed and telescope time sought from an allocation committee.

Phase II: The second phase of an ESO-VLT observing project, in which telescope time has been granted and the observing programme is planned in advance.

Phase II Proposal Preparation (P2PP): The P2PP system allows an observer to prepare *observation blocks*. The observer needs to select *templates*, define parameters associated with those *templates* and give additional parameters for scheduling and data reduction.

Pointing (VISTA): A set of one or more *exposures,* which can be spaced by a series of small *jitters* around a single telescope position using the same guide star(s).

Point: A basic unit of the hierarchical database structure. It may contain other *points* extending the structure or one or more *attributes* containing information.

Point Configuration File: A complete description of an RTAP or CCS-lite database.

Portable User Interface Toolkit (UIF, ESO-VLT software): This software module provides the basic tools to build a homogeneous man-machine interface on an ESO-VLT workstation.

QC Pipeline: A set of automatic data reduction procedures to remove detector and instrument signatures, with the aim of assessing data quality rather than scientific analysis.

Raw data/Raw frame: The term "raw data" or "raw frame" applies to data saved by an instrument which has not been processed by the *pipeline*. Each exposure can generate one or more (typically one) raw frames. These data are always included in the archive. See also "*reduced data*" and "*transient data*".

Readout data/Readout frame: The data resulting from a single readout of the detector (an example of *transient data*). One or more readout frames may need to be combined to generate a single *integration frame*.

Reduced data/Reduced frame: The term "reduced data" applies to data which has been processed automatically by the data reduction pipeline or manually by a data reduction package. (Applies to science data and calibration data). These data may be included in the archive See also "*raw data*" and "*transient data*".

Reference configuration file: A *configuration file* describing the complete set of functions available for a particular instrument.

Reference setup file: A *setup file* containing the complete set of definable parameters required for an exposure.

Reference star: A star used by the active optics system. See *aO star*.

Reset Frame: A calibration frame taken using a *cold blocker* filter with zero exposure time, giving a readout of the signal from each detector immediately after it has reset. The reset frame provides a zero point that can be subtracted from subsequent data frames. Also known as a *bias frame* (DPR.TYPE=BIAS).

Root database point: The database path describing the top level *point* in a branch.

Root directory: The top level directory containing a tree of ESO-VLT software files. "**VLTROOT**" contains the ESO-VLT common software, "**INTROOT**" contains the installed application software and "**INS_ROOT**" contains instrument configuration files and data. In addition, "**VLTDATA**" contains application configuration data and log files. "**MODROOT**" is sometimes used to point to the top level directory containing the current software module.

Run Number: An incremental number used to manage the observations for a science programme or made during the night.

Scan System (ESO-VLT software): This software module a bi-directional way to exchange data between workstations and to copy data from the LCUs to the workstations in the ESO-VLT environment.

Schedule Server: A process that runs within the *Observation Handling System* and provides schedule information (*observation blocks* to be executed) to the ESO-VLT Control Software upon request.

Scheduler (SCHED): A program which assists operations in implementing flexible scheduling.

Scheduling parameters server: A process that runs within the ESO-VLT Control Software and provides scheduling parameters (current weather conditions, current instrumental configuration, etc.) to the Observation Handling System upon request.

Scheduling resources: The resources needed by an Observation to make it schedulable (e.g. a specific telescope, instrument, instrument configuration and detector).

Sequence, Sequencer script: A set of commands in Sequencer language, generally intended to define and execute a series of related observations. These sequences are interpreted by the sequencer shell.

Sequencer: A software module used to define and execute a sequence of operations automatically and efficiently.

Service mode observing: An observing mode where observations are made by ESO staff according to a schedule and specific programme requirements. The PI may monitor the results but will not modify the schedule in real time.

Setup file: An ASCII file in *short hierarchical format* describing setup parameters for exactly one exposure. If a setup file contains all the setable parameters required for an exposure it is called a *reference setup file*, otherwise it is a *partial setup file*. An example of the latter is a *telescope setup file* with part of the necessary information to set up the telescope for a particular exposure. Other examples of partial setup file are *instrument setup files* and *detector setup files*.

Short Hierarchical Format (SHF): A format derived from Hierarchical FITS keywords, used for parameter files (setup files, configuration files etc.). Also referred to as “Short-FITS” format.

Sun Blind: The “sun blind” is a reflective filter designed to protect the camera from ambient light. In the VISTA IR Camera this filter is called SUNBLIND.

Target: The astronomical object or field to be acquired.

Telescope movement: In the context of VISTA, a small shift of the telescope to a new position relative to a reference position which defines the current target. (N.B. Small VLT telescope movements are referred to as an “offset”, but the VISTA TCS does not use the offset command — all movements, large and small, are handled with the setup command).

Telescope preset: A slew of the telescope to a new target.

Telescope setup file: A *setup file* containing a subset of setable parameters required for an exposure relating to the configuration of the telescope.

Template: Any instructions template (such as *Acquisition Template, Observation Template, Calibration Template* or *Technical Template*). Templates have input parameters described by a *template signature* and produce results that can serve as inputs to other templates (e.g. an acquisition template can generate a slit angle to be used in later templates). A template contains a sequence (*Sequencer script*) dealing with the setup and execution of one or more exposures. Templates are used to describe telescope, instrument and detector operations that are needed often. The exact behaviour of the execution of a template is determined by the values of its parameters.

The term “template” can also be used to refer to a stub which contains blank parts to be filled in (such as a file template or directory structure created by the getTemplate command, or an environment template used by TAT or pkgin, or the *Template Instrument Software*).

Template call: The name of a template to be executed, together with its parameter values (in SHF).

Template Instrument Software: A software control system for a fictitious instrument for the purpose of providing an example to help instrument software developers.

Template parameter GUI: Graphical user interface used to edit actual *template* parameters.

Template selector: A GUI which allows a user to select a particular *template* name.

Template server: A process that runs within the ESO-VLT Control Software and provides information about *templates* on request. Each instrument has a corresponding template server process.

Template signature file: This is a description (in short hierarchical format, SHF) of a *template* and its parameters. It contains information about the type and allowed ranges of the parameters, so that a trivial validity check can already be performed when the parameters are entered via the template parameter GUI.

Tile (VISTA): A filled image of sky fully sampled (by filling in the gaps in a pawprint) by combining multiple *pawprints*. Because of the detector spacing, a minimum of 6 pointed observations (with fixed offsets) are required for reasonably uniform coverage, which exposes each piece of sky, away from the edges of the tile, to at least 2 camera pixels. The QC pipeline does not combine *pawprints* into tiles. The separate *pawprints* in a tile require a different set of guide and LOWFS stars. Several tiles may be combined together to make a survey.

Time System: This software module provides Universal Time (UTC), allowing process synchronisation within one processor or between processors.

Top level source directory: The directory into which the "vcins" module has been installed and used to build the VIRCAM software directory tree with "pkginBuild".

Transient data: The term "transient data" refers to data that only exists for a finite time and is not normally saved permanently (except for engineering purposes). Examples include the integration data kept by IRACE before co-adding or the image data used by the autoguider. These data can be displayed while they exist, but are not normally included in the archive. See also "*raw data*" and "*reduced data*".

Translation/Alias Table: A table containing alternative names for ESO-VLT standard keywords. It can be used to translate short names into ESO-VLT standard parameter keywords.

Unit (ESO-VLT software): A group of logical or functionally related components.

Unit Telescope (UT): Each of the four main ESO-VLT telescopes: Antu, Kueyen, Melipal and Yepun.

1.7 Stylistic Conventions

The following styles are used within this document:

bold

in the text, for commands and filenames as they have to be typed.

italic

in the text, for parts that have to be substituted with the real content.

`teletype`

used for examples

`<name>`

in the examples for parts that have to be substituted with the real content.

bold and *italic* are also used to highlight words in the main text, and *"italic in quotes"* is used for document reference titles.

1.8 Naming Conventions

ESO-VLT standard naming conventions are applied, as described in [RD25].

1.9 Problem Reporting/Change Request

Problem reports and change requests should be made using the ESO-VLT SPR system, as described in [RD33]. The package name is VIRCAM.

2 OVERVIEW

2.1 Hardware Architecture

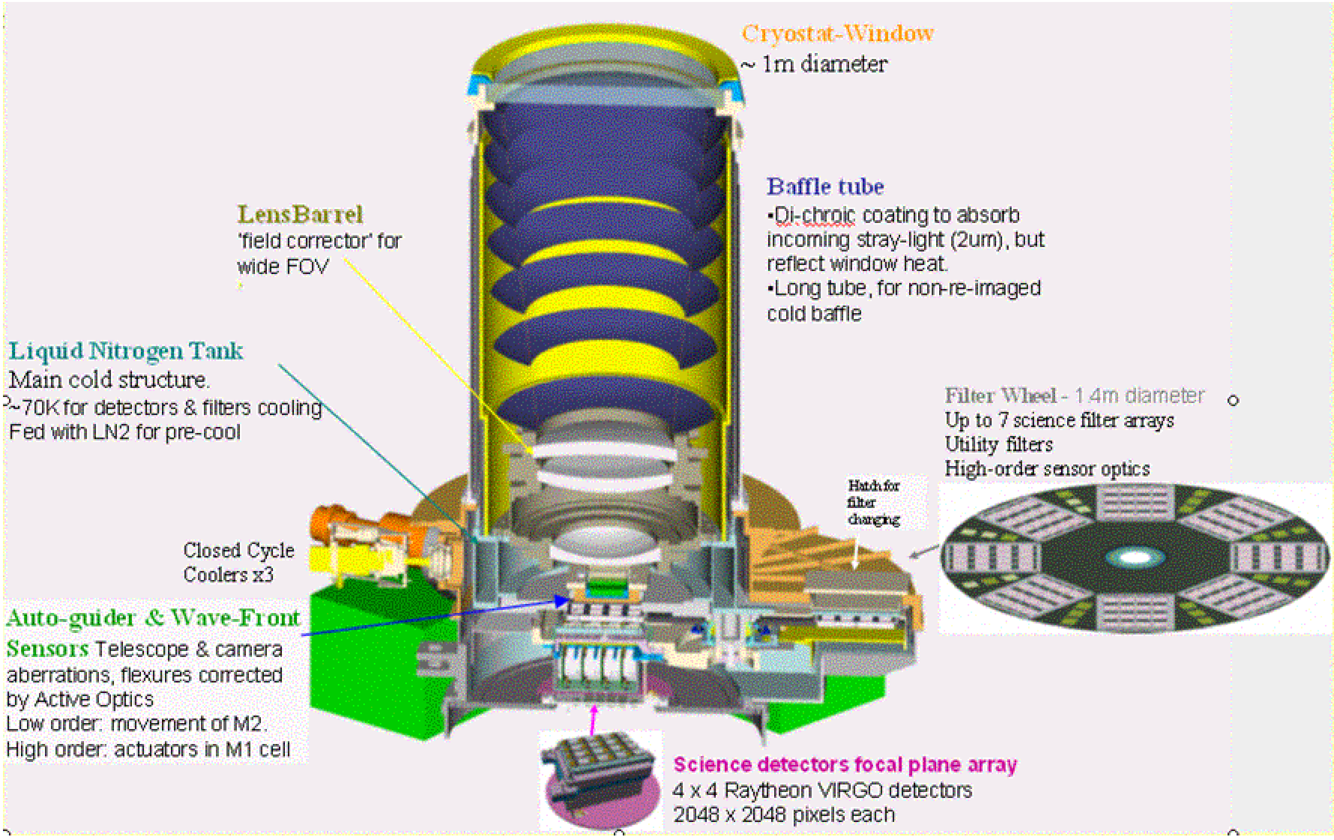


Figure 1 Overall Layout of the VISTA IR Camera

The VISTA IR Camera consists of the following major components, as shown in Figure 1:

- A cryostat to keep the contents cold and under vacuum. The cryostat is cooled with a set of closed-cycle cryo-coolers. The software monitors the vacuum pressure and the temperature at various key locations and controls a heater on the top of the cryostat tube, which prevents condensation forming on the window.
- Five electronic cabinets, thermally controlled with ESO-VLT cabinet cooling controllers.
- A filter wheel, designed to put a selected tray of science filters into the beam in front of the science detectors. The filter wheel is also fitted with smaller, "intermediate" filters which are positioned in the beam for calibration observations.
- The science detectors — a 4x4 array of 2048x2048 pixel VIRGO infrared detectors. An ESO-VLT IRACE controller [RD39] controls these detectors. The detectors are mounted on a focal plane plate whose temperature is managed by the VISTA IR camera software.

- A pair of wavefront sensor units, positioned on opposite sides of the focal plane, including:
  - Two 1024x2048 pixel frame transfer, visible light CCDs for the autoguider;
  - Two low-order wavefront sensor units, each consisting of two 2048x2048 visible light CCDs.

  All these wavefront sensor CCDs are controlled with ESO-VLT "technical CCD" (TCCD) controllers, [RD40], from the VISTA TCS, [RD13].

2.1.1 Sensors and Controllers

Figure 2 shows a block diagram of the instrument local control unit (LCU), showing the hardware used by the instrument sensors and controllers.

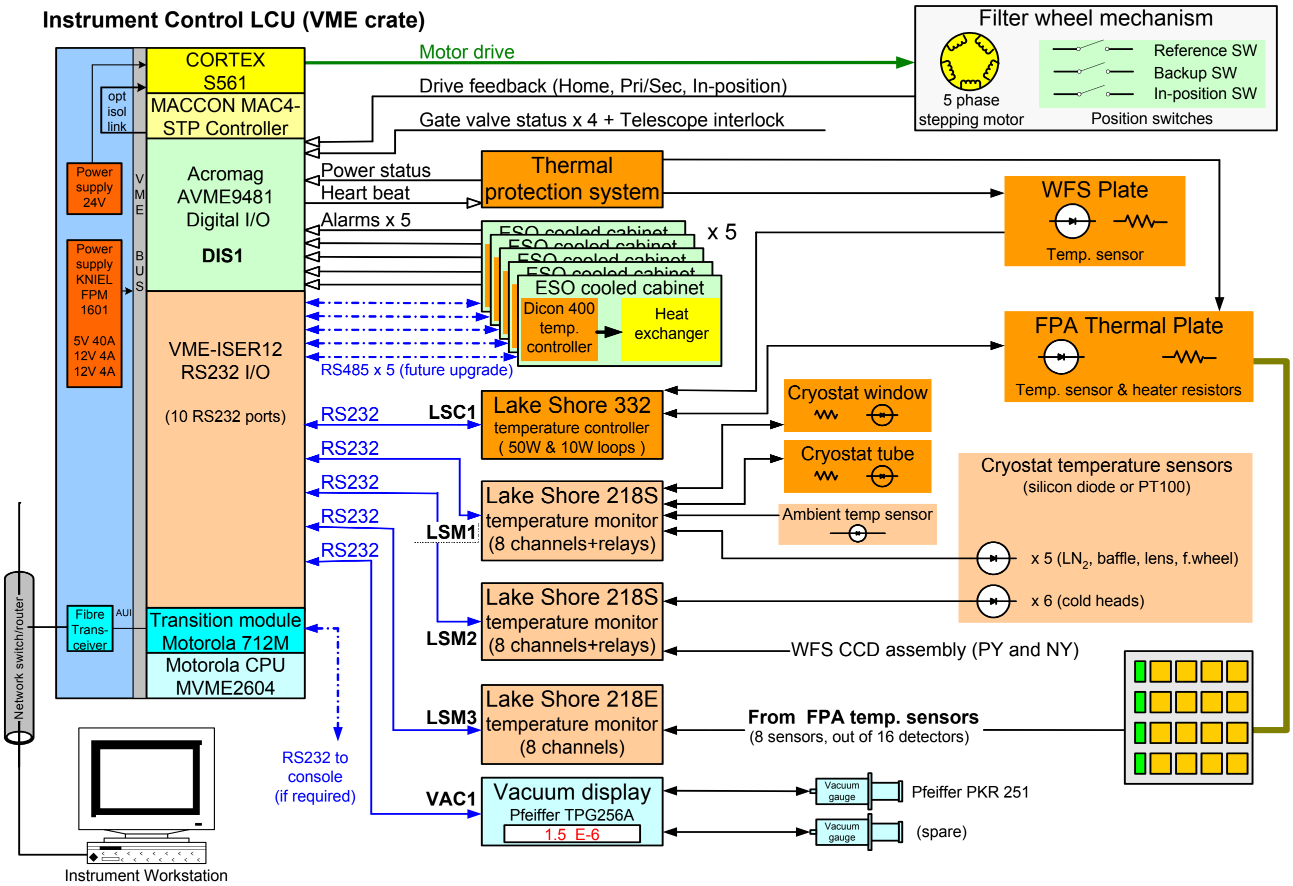


Figure 2 Block Diagram of Instrument LCU Connections

The instrument has the following sensors and controllers:

- 24 temperature sensors measuring:
  - 1 ambient air temperature;
  - 1 cryostat window cell temperature;
  - 1 cryostat top tube temperature;
  - 1 optical bench top temperature;

  - 1 baffle temperature;
  - 1 lens barrel temperature;
  - 2 filter wheel temperatures (shield and hub);
  - 6 cryo-cooler temperatures (1st and 2nd stages of the 3 cold heads);
  - 2 wavefront sensor assembly temperatures (PY and NY);
  - 8 science detector temperatures (there is a sensor on each of the 16 detectors, but only 8 are used at a time – the other 8 are spares).

- 3 temperature controllers controlling:
  - The cryostat window cell temperature (heaters switched on and off using relays);
  - The cryostat top tube temperature (heaters switched on and off using relays);
  - The focal plane array (FPA) thermal plate temperature, attached to which are the science detectors (fine heater control using PID feedback).

- 2 vacuum sensors measuring:
  - Cryostat vacuum;
  - Spare vacuum sensor – sometimes connected during engineering procedures.

- 5 ESO-VLT cabinet cooling controllers, each looking after a different electronics cabinet. These normally provide[1]:
  - 4 temperatures (coolant inlet temperature, coolant outlet temperature plus 2 internal cabinet temperatures);
  - 3 flow rates;
  - 1 alarm signal.

- 16 digital I/O signals giving
  - Status of filter reference (or home) switch
  - Status of filter in-position switch
  - Status of filter reference selector switch
  - VME mains power supply status
  - Thermal protection DC power supply status
  - Status of detector thermal protection heater
  - 4 signals giving the status of gate valves 1 and 2
  - 5 cabinet cooling controller alarm signals

  In addition there is one spare input and one spare output signal not currently connected to anything, which can be used for future upgrade.

NOTE: The Lakeshore 332 device is capable of controlling two temperature channels, but you will notice in Figure 2 that only the channel to the "FPA Thermal Plate" communicates both ways. The other channel only senses the temperature of the "WFS Plate", since the WFS

1 At present, standard ESO software is not available for the cabinet coolers used by VISTA (which use DICON Jumo temperature sensors), so only the alarm signals are used by the VISTA IR Camera software.

detector temperatures are controlled separately by the TCCD controller (managed by the VISTA CTS and not shown here).

2.1.2 Thermal control

The VISTA IR Camera system manages the temperature of the instrument components using the temperature sensing and heating devices listed in the previous section. There are two levels of control loops:

- Low level control: The Lakeshore devices installed in the VME crate will maintain their individual temperatures at their current set points as long as they are powered up, regardless of whether the LCU software or workstation software are running. When the instrument in its operational state at thermal equilibrium, these Lakeshore devices can maintain the detector and cryostat temperatures close to their operational values.

- High level control: When it is running and in the ONLINE state, the VIRCAM ICS server process (vciServer) on the instrument workstation monitors a collection of temperatures from the LCU and makes adjustments to the set points of the Lakeshore devices at regular intervals. These adjustments are important in the following situations:
    - When the filter wheel brings a filter from a warmer part of the instrument over the detectors. The high level control loop detects a warming of the detectors and reduces the Focal Plane Array set point to compensate.
    - When there are significant changes in the ambient temperature, the high level software adjusts the Lakeshore temperature settings to keep the camera window and tube within their desired range of the ambient temperature.
    - When the instrument is cooling down and has not yet reached thermal equilibrium. The high level control loop initially protects the detectors from any contaminants that may try to condense on them by keeping them warmer than the surrounding cryostat. At the end of the cooldown the software allows the detectors to cool to their target temperature and thermal equilibrium to be achieved.
    - When the instrument is warming up, the high level control loop warms the detectors to boil off any contaminants that may try to condense on them.

NOTE: Since the high level thermal control loop only operates when the ICS workstation and LCU software are both running, and the instrument software is in the ONLINE state (section 5.7 on page 96), *it is important to have the software running and ONLINE whenever the instrument is being cooled down or warmed up, at least until thermal equilibrium is achieved. The software should also be put in the ONLINE state well before operations begin (at least 4 hours in advance if possbile)*, to allow the software to correct the detector target temperature and cryostat window and tube target temperatures for any environmental changes that may have happened while in the STANDBY state.

The thermal control software operates the state machine shown in Figure 3 below, and detects whether the instrument is cooling down, warming up or is steady at ambient or operational temperature. The thermal states are:

AMBIENT

Temperatures are monitored and the state is changed if required.
There is no high level temperature control.

COOLDOWN

Temperatures are monitored and the state is changed if required.
The detectors are allowed to cool but are maintained at a temperature significantly warmer than the mean cryostat internal temperature for the majority of the cooldown. The the rate of change of temperature of the detectors and the temperature gradient between the detectors and focal plane array (FPA) plate are both kept within safe limits. When the cryostat has cooled below its target temperature, the detectors are allowed to cool to their target temperature until thermal equilibrium is achieved. The software ensures the detectors are never cooler than a safe limit below the optical bench top temperature.
The cryostat window is warmed to prevent condensation during cooldown.

OPERATIONAL

Temperatures are monitored and the state is changed if required.
The detector temperature is maintained at its target temperature by making small adjustments to the focal plane array temperature set point. The software ensures the detectors are never cooler than a safe limit below the optical bench top temperature.
The cryostat window and tube temperatures are kept within their operational limits, adjusting the set points to track changes to the ambient temperature.

WARMUP (also used during transit)

Temperatures are monitored and the state is changed if required.
The detectors are warmed but the rate of change of temperature and the temperature gradient between the detectors and focal plane array (FPA) plate are both kept within safe limits. Towards the end of the warmup, the detectors are maintained at a temperature significantly warmer than the mean cryostat internal temperature.
The cryostat window is warmed to prevent condensation during a transit.

The target temperatures and transition temperatures can be programmed by an instrument engineer by means of instrument configuration parameters, as described in section 6 on page 119. The parameters can be displayed by the panels shown in section 4.5 on page 62. For details of the sensor and control software see the *"ICS Software Design Description",* [RD4].

NOTE: The thermal control software operates slowly. When the software is first started, the software has to go through the state changes shown in Figure 3 below. It begins in AMBIENT mode. If the cryostat is cold it will move to COOLDOWN mode. If the cryostat is warming up it may take several minutes for the cold heads to warm by DELTA and the software to move to the WARMUP state. The thermal control screen, described in section 4.15.2 on page 84, can be used to speed up the state transitions when necessary.

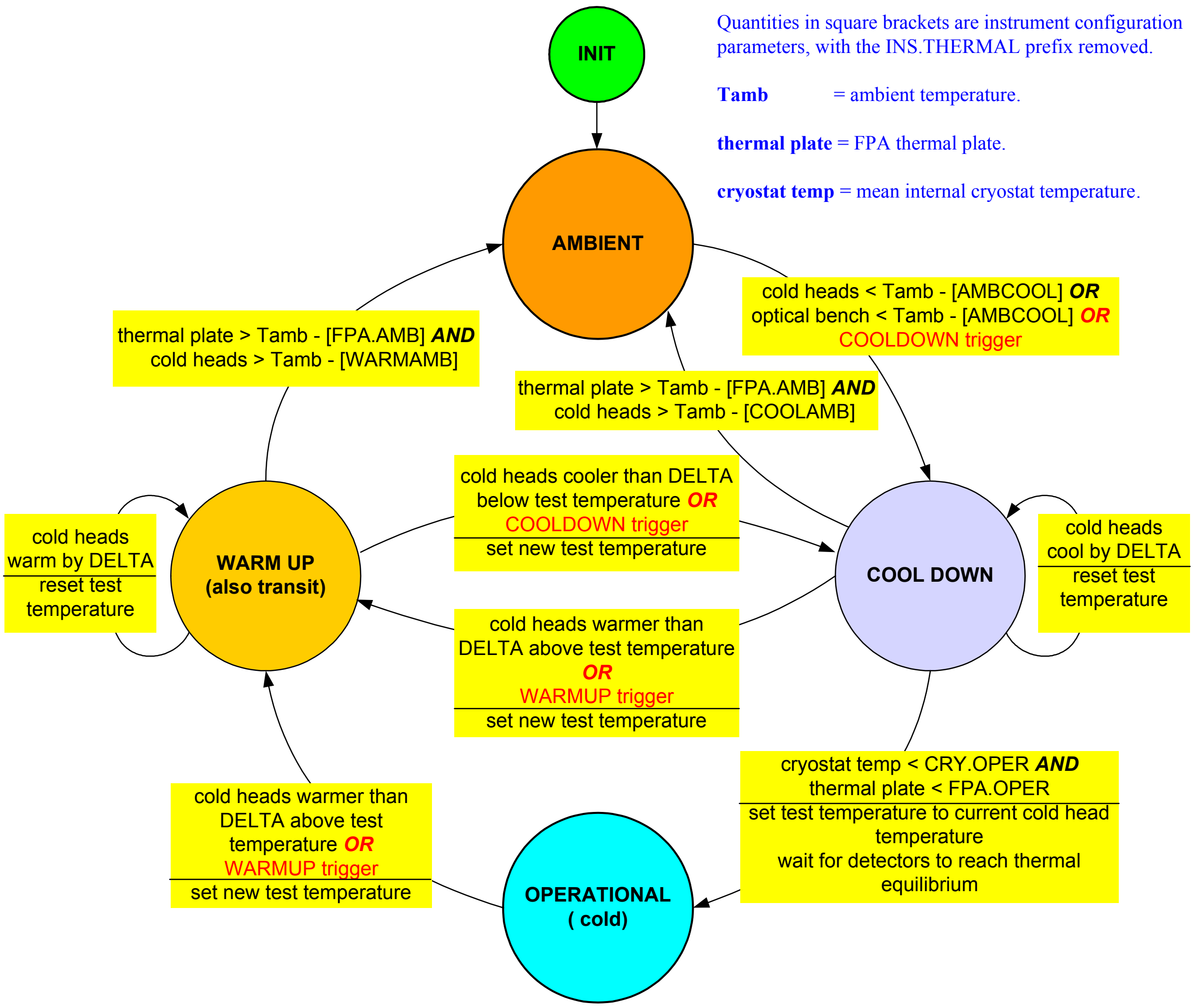


Figure 3 VIRCAM Thermal Control States

2.1.3 The Filter Wheel

The layout of the filter wheel is shown schematically in Figure 4 below. The wheel has 8 slots, 7 of which can be fitted with a "tray" of science filters, with one slot reserved for a tray of DARK filters. Each tray consists of a 4x4 array of filters designed to match the 4x4 array of science detectors. The wedge-shaped spaces in between the science filter trays can be populated with smaller "intermediate" filters. These filters only cover a subset of the science detectors and are designed for one-off calibration observations. These filters can be offset from the beam centre, and made to cover different detectors, by rotating the filter wheel slightly. The beam splitters for the high order wavefront sensor fit into these intermediate positions.

The wheel is 1.37 metres in diameter. It takes 210000 half-steps of the motor to rotate the wheel by one revolution, so a movement of one motor half-step moves the rim of the wheel by 20.5 microns. The filter wheel is driven at a maximum speed of 4 KHz, which will rotate the wheel through one revolution in about 53 seconds.

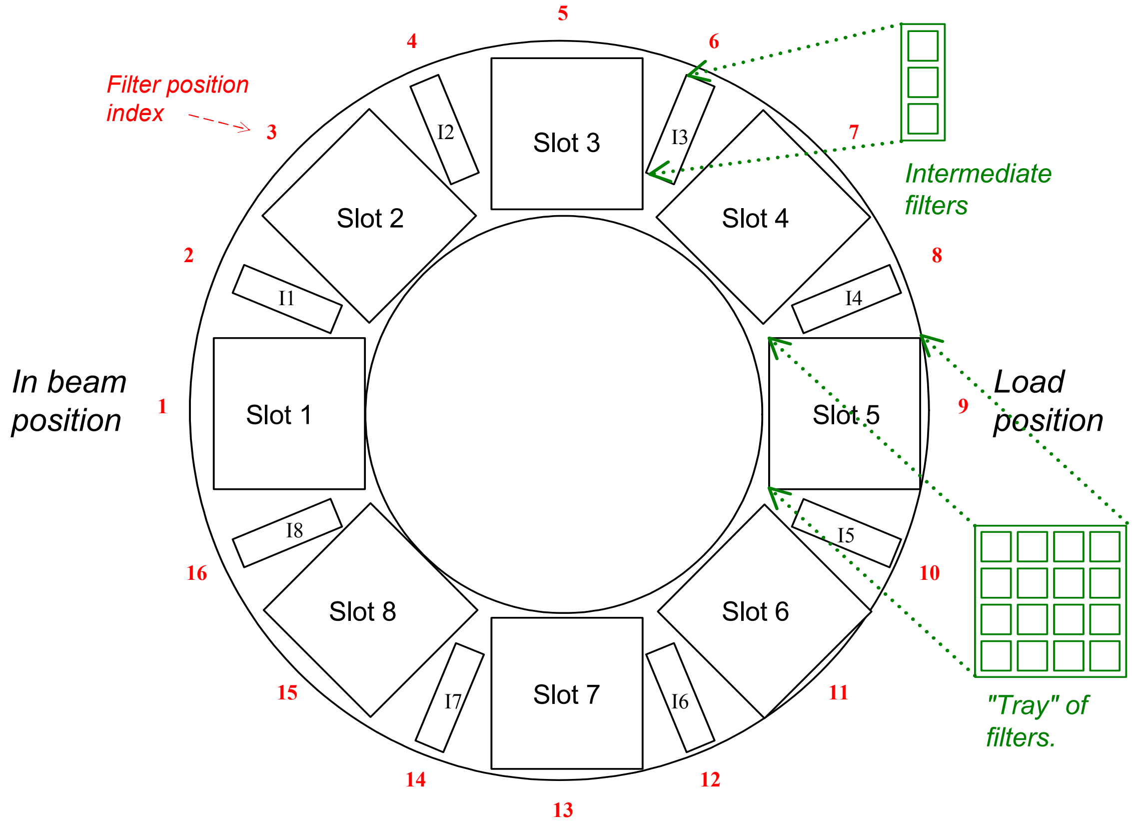


Figure 4 Layout of the VISTA IR Camera Filter Wheel

The following filters were loaded in the filter wheel (see vcmcfg/config/vcmcfgICS_filters.cfg), as of 10th July 2007:

Slot/Int	Filter
1	SUNBLIND
2	NB118
3	J
INT 3	HOWFS J beam splitter
4	Ks
5	H
6	reserved for Z'
7	Y
8	DARK1

Figure 5 below shows a schematic layout of the filter wheel drive. The wheel is driven with a stepper motor and positioned by counting the number of motor steps from a reference switch. A backup reference switch can be selected as a temporary measure to allow the wheel to keep working after a failure of the main reference switch until a switch replacement can be scheduled.

The wheel also has an in-position switch which is "active" whenever a tray of science filters is positioned in the beam. These in-position switches cover several hundred steps of travel, so an "active" switch does not guarantee that a filter is correctly positioned within its required tolerance. However, regular test procedures may be used to verify that the filter wheel motor is achieving the necessary accuracy (see section 6.3.4 on page 127).

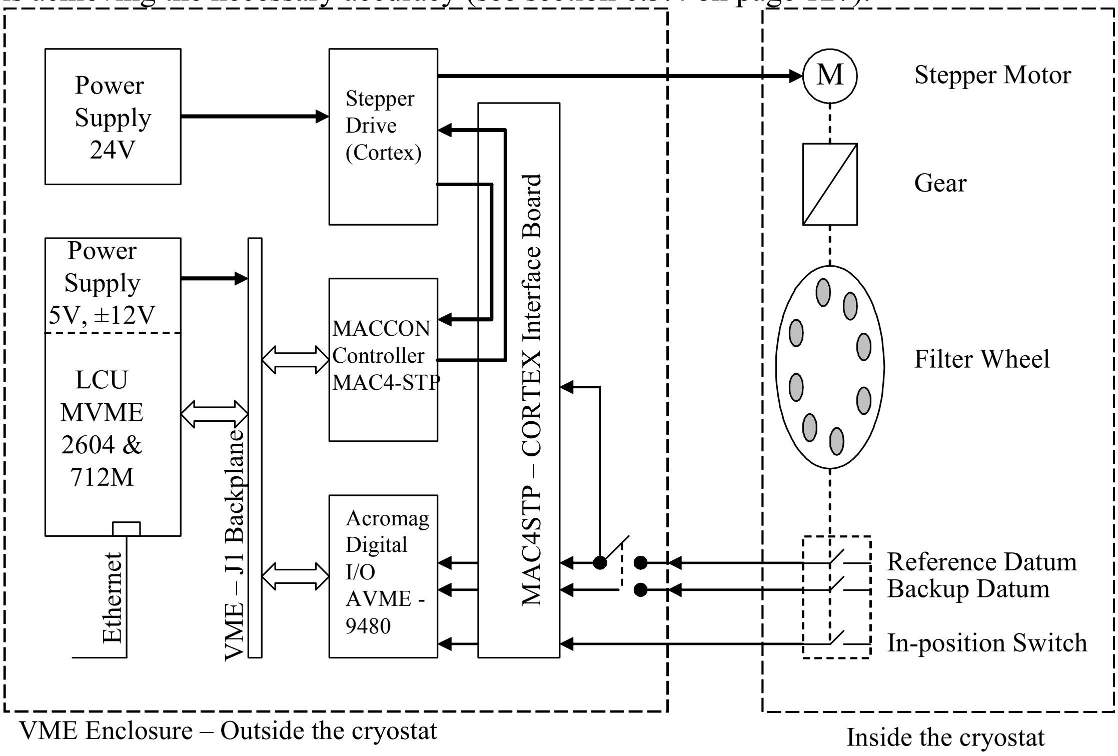


Figure 5 Layout of the Filter Wheel Drive Electronics

The VISTA filter wheel control software also has the ability to prevent the detectors being flashed with unnecessary ambient light by choosing a path which passes the least number of bright filters through the beam (by means of the INS.FILTER assembly, see section 5.5 on page 90). For details of the filter wheel control software see the *"ICS Software Design Description",* [RD4] and "*VISTA IR Filter Wheel Control*", [RD3].

2.1.4 The Science Detectors

The VISTA IR Camera has an a 4x4 array of 2048x2048 pixel VIRGO science detectors whose layout on the focal plane is shown in Figure 6 below, which also shows the instrument X and Y coordinates. The detectors are controlled using the ESO/VLT IRACE controller, [RD39], and are separated by 90% of their size parallel to the X axis and 42.5% of their size parallel to their Y axis. The labels on the detectors show the labels attached to their cables (in brackets, 1A to 4D) and the chip number used by the IRACE controller (1 to 16).

The VISTA telescope and IR camera optics together produce a on-axis plate scale on the camera focal plane of 17.0887 arcseconds/mm, with a focal length of 12.07m. The focal plane is distorted by pincushion distortion[2]. Each detector has a pixel size of 20µm, and the 2048x2048 pixels cover an area of 40.96mm x 40.96mm on the focal plane. The pincushion distortion (due to projection effects between the spherical sky and flat focal plane, and due to residual distortions in the optical system) makes the detectors further from the optical axis cover a smaller area on the sky. The mean pixel size across the whole focal plane is 0.339 arcseconds on the sky, and each detector covers a ~694 x 694 arcsecond^2 area of sky on average. The 16 detectors cover 274.432mm x 216.064mm on the focal plane, which gives a nominal field of view of 1.292° x 1.017° on the sky.

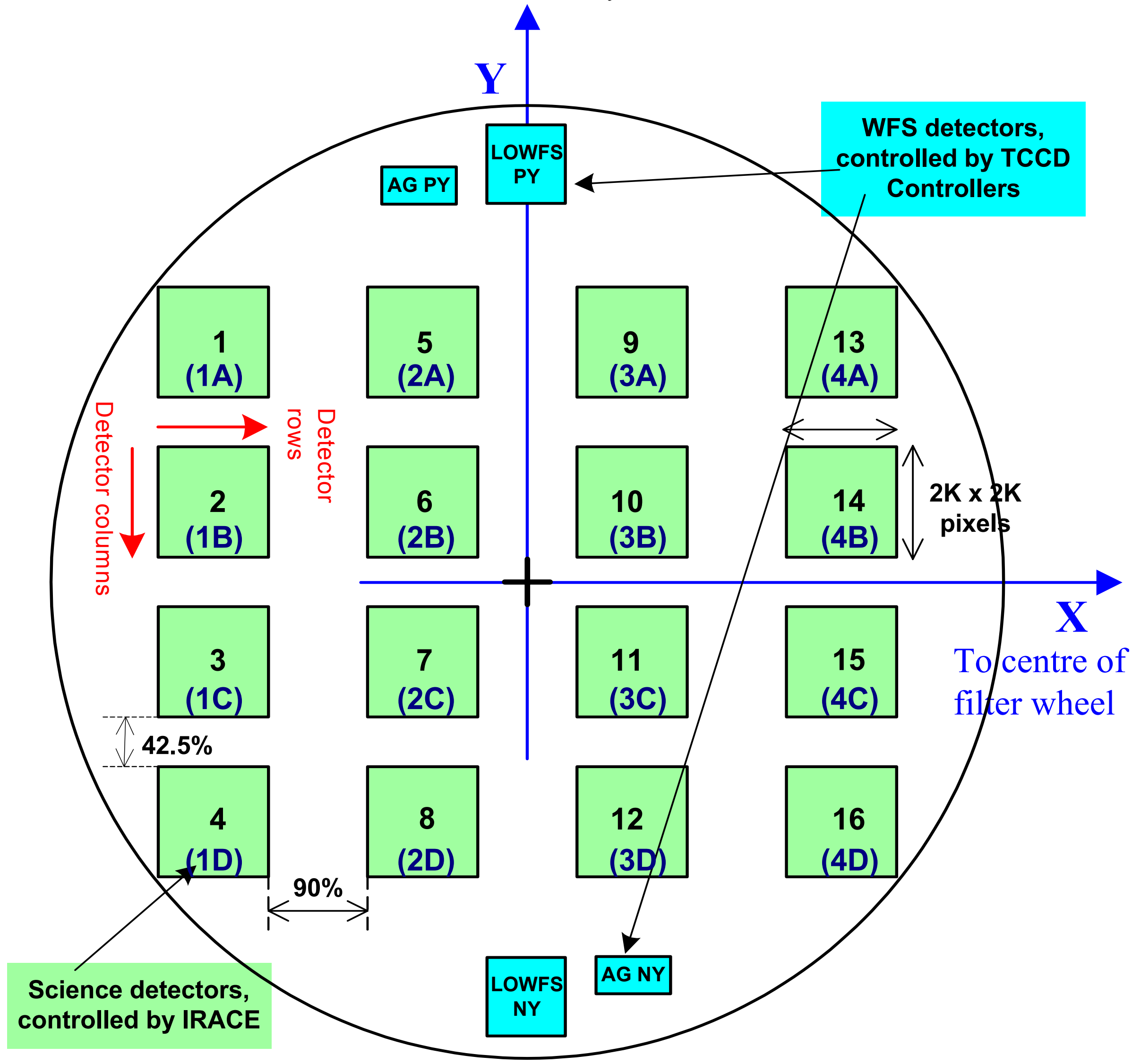


Figure 6 Layout of the VISTA IR Camera Focal Plane

[2] The distortion coefficient used by the slaPcd function in the VISTA TCS and VIRCAM software is +41.79705.

The detector chips installed in positions 1-16 (see vcdcfg/config/vircam.dcf) on installation at Paranal in 2007 were:

Position	Detector chip	Position	Detector chip	Position	Detector chip	Position	Detector chip
1	#35	5	#39	9	#45	13	#30
2	#22	6	#36	10	#47	14	#43
3	#23	7	#41	11	#33	15	#42
4	#44	8	#25	12	#46	16	#38

A block diagram of the IRACE system is shown in Figure 7 below. The IRACE system is controlled through two number cruncher workstations, wvcirc1 and wvcirc2, each of which communicates with a detector front end (DFE) electronics box which handles 8 detectors. Both of the number crunchers should be up and running for the full IRACE system to work.

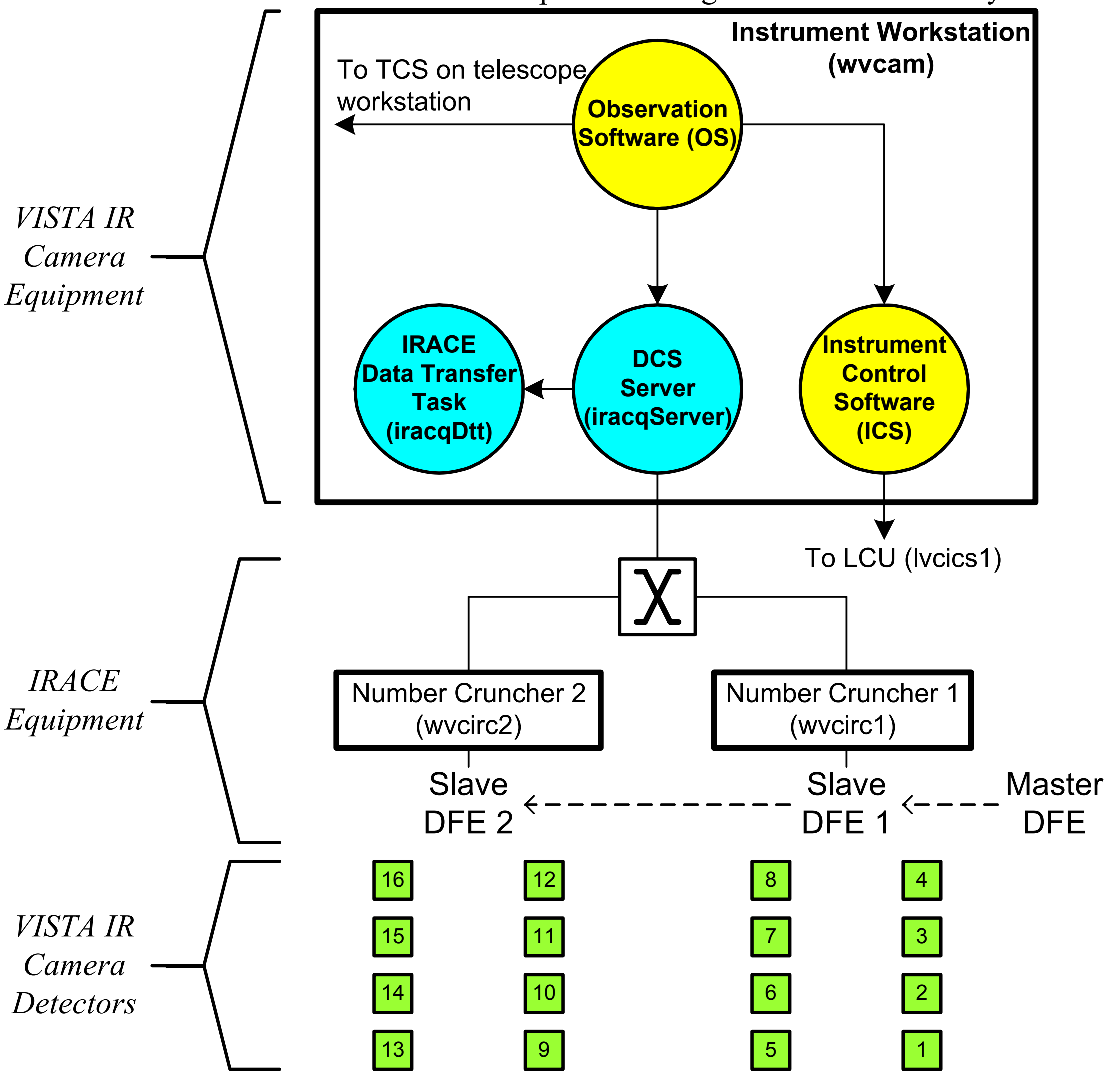


Figure 7 Simplified Block Diagram of the VISTA IRACE System

2.1.5 The Wavefront Sensors

Unlike most ESO-VLT instruments, the VISTA IR Camera has its own built-in wavefront sensors whose locations on the focal plane are shown in Figure 6 (see [RD9]). These locations are fixed, and the wavefront sensors do not have moveable probes. These wavefront sensors consist of:

- An autoguider (AG) which has two discrete 1024x2048 pixel CCD detectors at opposite sides of the focal plane (labelled “1” or “py” on the positive Y side and labelled “2” or “ny” on the negative Y side). The autoguider is designed to operate with a guide star image falling on *one* of the detectors (the other detector is not used for normal operations[3]). It may be thought of as a single autoguider whose target field consists of two discrete rectangles.

- Two low order wavefront sensors (LOWFS) at opposite sides of the focal plane (also labelled “1” or “py” and “2” or “ny”). Each LOWFS contains a beam splitter which sends pre-focus and post-focus images to two 2048x2048 pixel CCD detectors, as shown in Figure 8. So there are four LOWFS detectors altogether; two for each LOWFS.

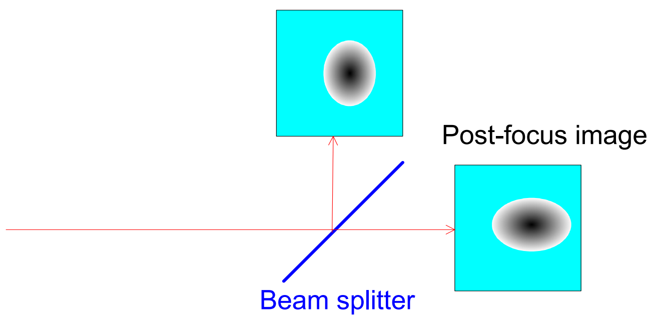


Figure 8 Schematic layout of LOWFS with 2 CCD detectors

- A high order wavefront sensor (HOWFS) which uses a beam splitter stored in an intermediate position on the filter wheel and extracts pre-focus and post-focus images from the science detector. There are two HOWFS beam splitters, each at a different radius from the centre of the filter wheel. Figure 9 below shows the arcs swept out when the filter wheel moves these beam splitters across the focal plane. The “+” symbols show the locations of the 8 pre-defined HOWFS measurement positions. The “HOJ” prefix in the names of these locations indicates that this particular beam splitter contains a filter in the “J” wavelength range. An “ex” or “ey” in the name of a HOWFS filter represents extreme x or y, “co” in the name means ‘corner` and “oa” means ‘on-axis’

[3] Some VISTA TCS calibration and commissioning procedures require an image to be made simultaneously with both AG detectors, but this is not a normal operating mode.

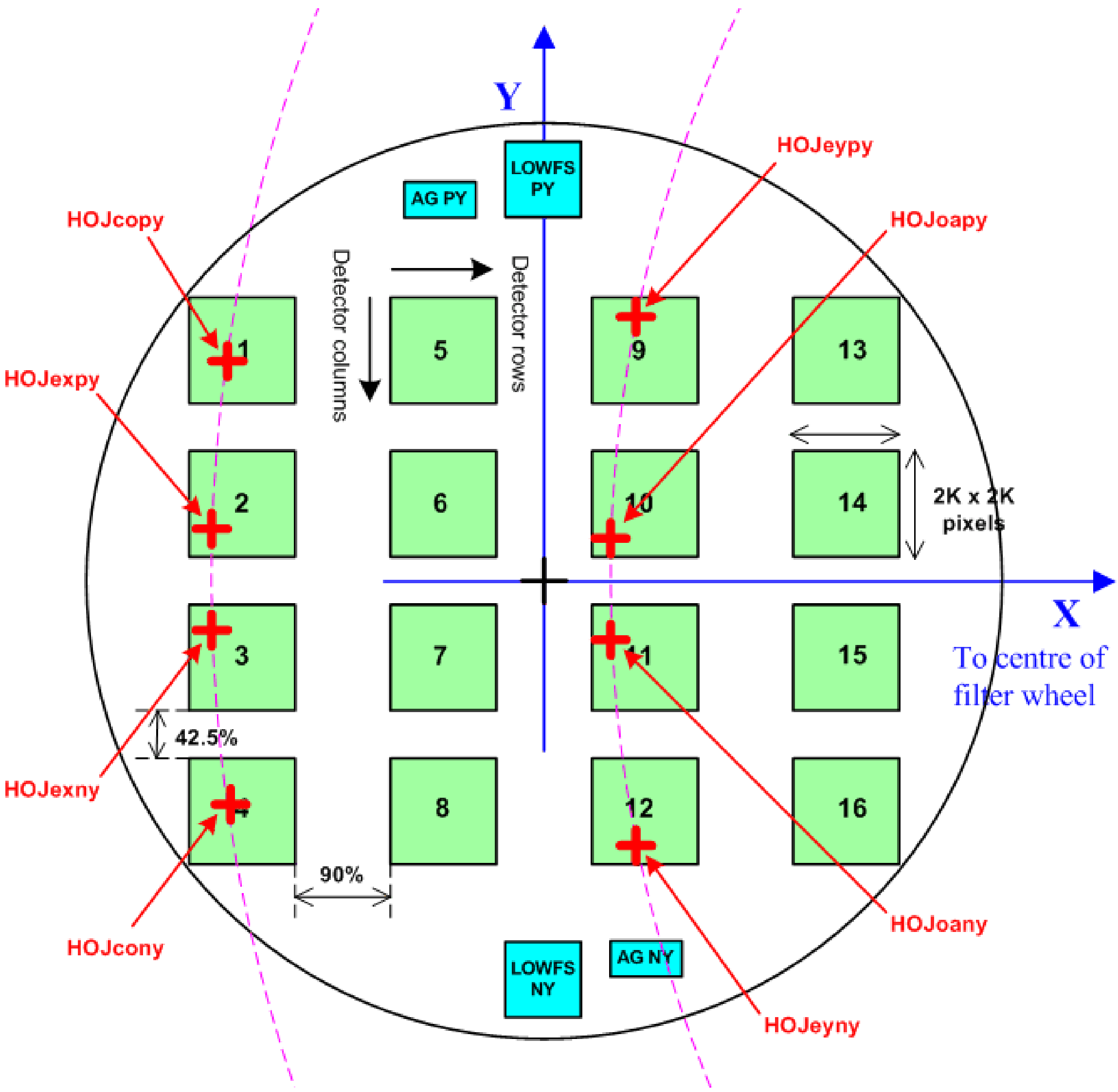


Figure 9 Location of HOWFS positions in the focal plane.

Autoguider and LOWFS observations are made concurrently with science observations[4]. HOWFS observations are made once or twice a night and are not concurrent with science observations (because they use the science detector).

Unlike a typical ESO-VLT exposure, which uses one guide star, each VISTA IR Camera exposure requires 3 stars — one star for the autoguider and two stars for the two LOWFS, as shown in Figure 10 below.

[4] Assuming the science exposure time is greater than the time taken for one closed loop LOWFS cycle.

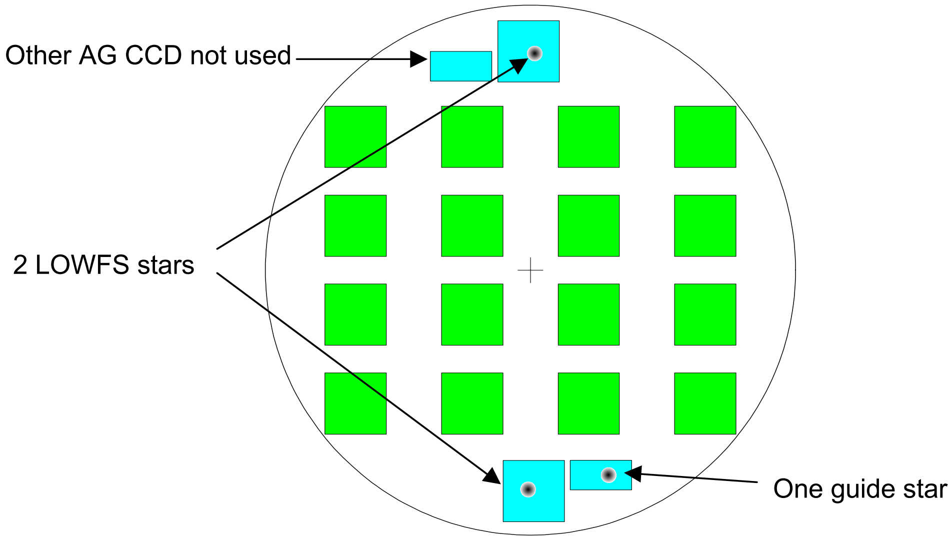


Figure 10 Each Pointing Requires 1 Guide Star and 2 LOWFS Stars

2.1.6 Computers

The VISTA IR Camera Software runs on the following computers (shown in Figure 11 below).

- Workstations
    - Instrument Workstation (wvcam); Linux workstation with high speed disks and Gigabit Ethernet interface.
    - Image analysis workstation(s) for LOWFS (wvtia1 and/or wvtia2). Linux workstations. *(OPTIONAL: This software can also run on the telescope workstation, wvttcs)*
- Local Control Units (LCUs)[5]
    - Instrument Control System LCU (lvcics1), with standard power PC processor and VME equipment shown in Figure 2.
    - Autoguider LCU 1 (lvtag1), with standard power PC processor and Technical CCD Controller (TCCD) hardware.
    - Autoguider LCU 2 (lvtag2), with standard power PC processor and Technical CCD Controller (TCCD) hardware.
    - LOWFS LCU 1 (lvtwfs1), with standard power PC processor and Technical CCD Controller (TCCD) hardware.
    - LOWFS LCU 2 (lvtwfs2), with high speed power PC processor and Technical CCD Controller (TCCD) hardware.
- Workstation/LCU hybrids
    - IRACE DCS number cruncher 1 (wvcirc1); Linux PC with Gigabit Ethernet interface.
    - IRACE DCS number cruncher 2 (wvcirc2); Linux PC with Gigabit Ethernet interface.

[5] Note that all but one of these LCUs are actually managed by the VISTA Telescope Control System.

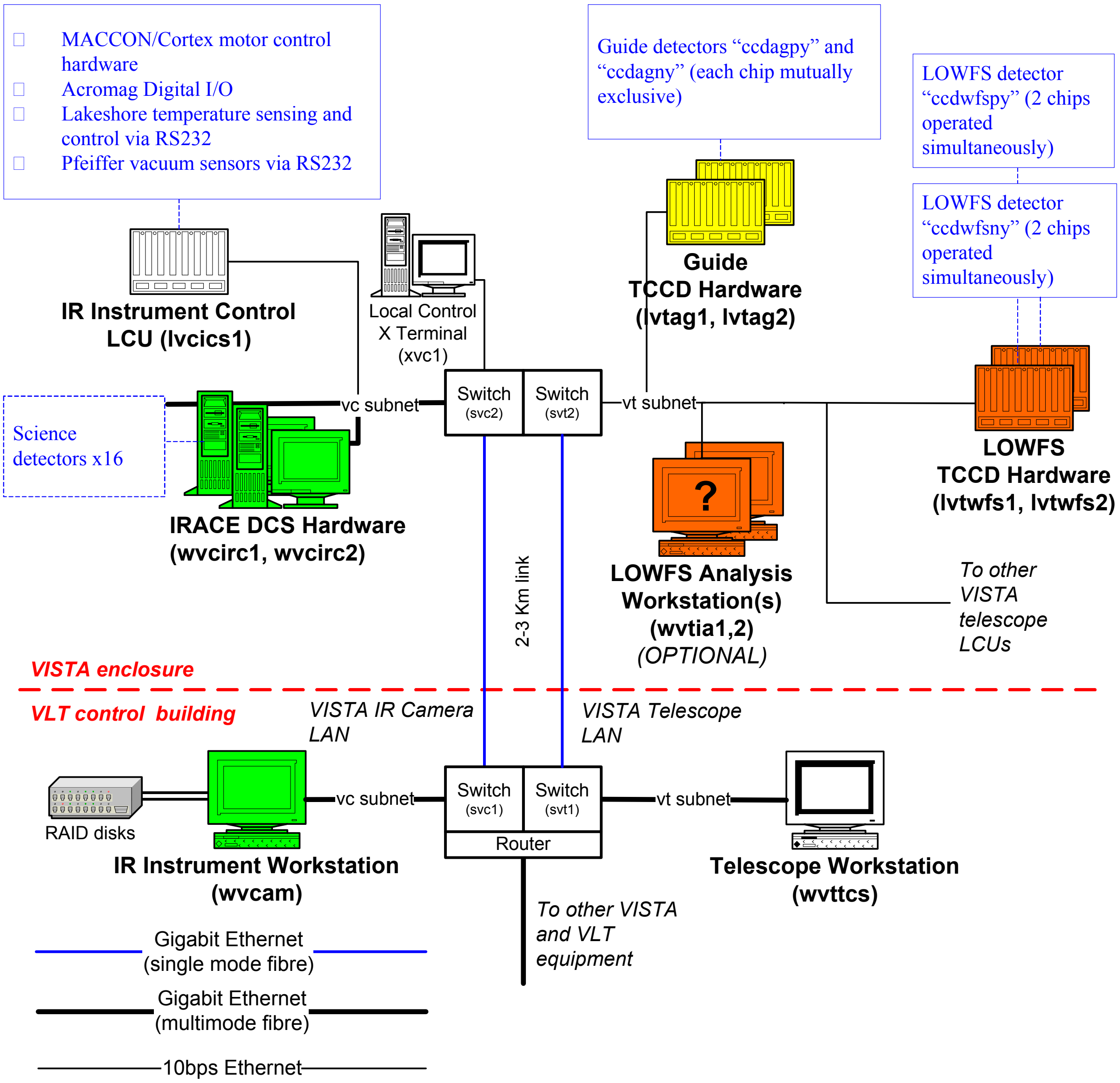


Figure 11 Layout of the VISTA IR Camera LAN

2.1.7 Layout of the VISTA IR Camera LAN

The layout of the VISTA IR LAN is shown in Figure 11 and is based on the standard ESO-VLT LAN layout specified in [RD28]. There are two subnets:

- The "vc" subnet is the VISTA IR Camera subnet, configured like a standard ESO-VLT instrument subnet.
- The "vt" subnet is the VISTA telescope subnet. The VISTA IR Camera is unusual because its autoguider and LOWFS components are regarded logically to be part of the telescope control system, even though they are physically located within the instrument.

2.1.8 Special connections

The VISTA IR Camera uses hardware which is part of the telescope control subnet, as shown in Figure 11 above.

2.2 Observing Strategy

VISTA is a survey telescope, so the VISTA IR Camera's primary function is to build up a map of a large area of the sky using several overlapping exposures. The basic unit of a survey is known as a "tile"; each tile being the smallest contiguous area of sky that the camera can image. A survey is made by tessellating the tiles together (with a small amount of overlap) to cover the required region of the sky (see [RD15]). The most efficient survey is made from a set of tiles which have the minimum overlap, and covering the sky in an efficient way can require some tiles (e.g. ones near the celestial poles) to be tilted with respect to their neighbours, like the ones shown in Figure 12, which ignores the curvature of the pawprint edges.

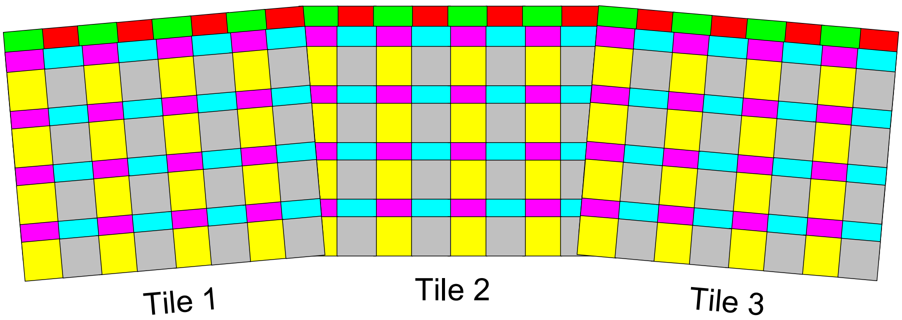


Figure 12 Three Tiles Making up a Simple Survey

2.2.1 Making tiles

A non-contiguous exposure made with the 16 detectors (section 2.1.4) is known as a "pawprint". The instrument needs to make a minimum of 6 overlapping "pawprints", with the telescope stepped by an amount in arcsec corresponding to ~0.95 detector widths in the X direction and an amount in arcsec corresponding to ~0.475 detector widths in the Y direction, to make a contiguous "tile", for example as shown in Figure 13 below (see [RD14] and [RD18]). The pincushion distortion is not shown in the figure.

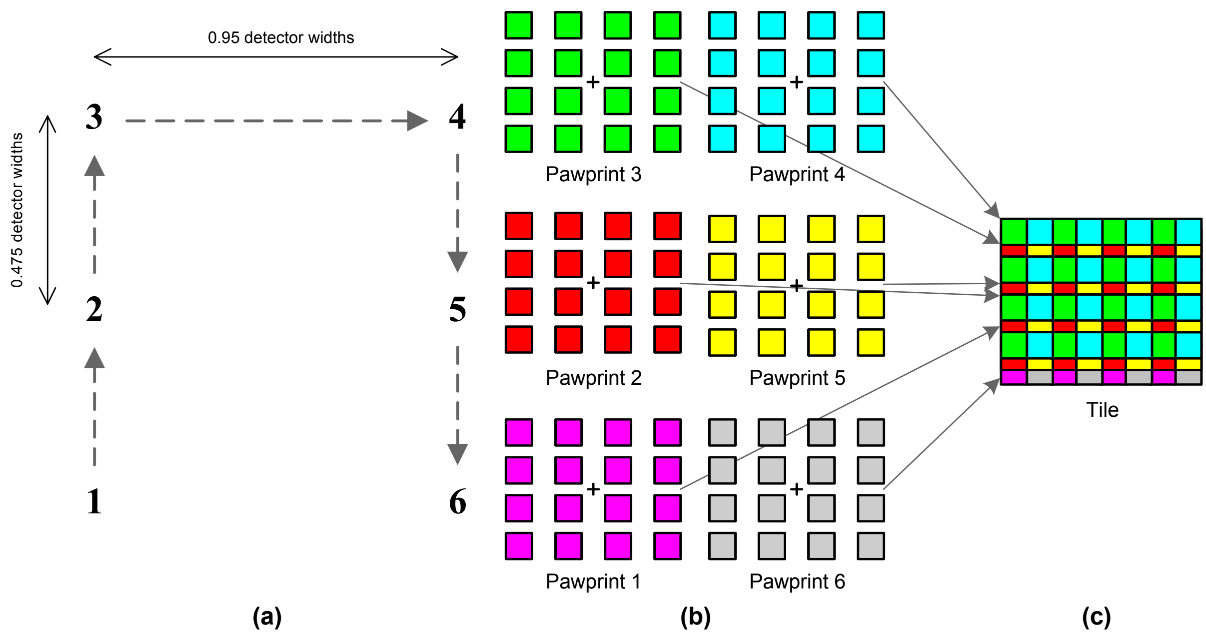


Figure 13 Six Overlapping Pawprints Make up One Contiguous Tile

Note that the telescope movements used to assemble a tile out of pawprints are made with respect to the X,Y coordinates in the camera focal plane, not with respect to celestial coordinates. So, unlike the layout of tiles shown in Figure 12, pawprints are not tilted with respect to their neighbours (unless such a tilt is specifically requested by the observer).

In addition, each individual pawprint can itself be made up from several observations made with smaller telescope movements (of <30 arcseconds in size). This sequence of movements made with the idea of co-adding the data to remove cosmic ray events and bad pixels and to smooth out regions of overlap is known as "jittering", as shown in Figure 14.

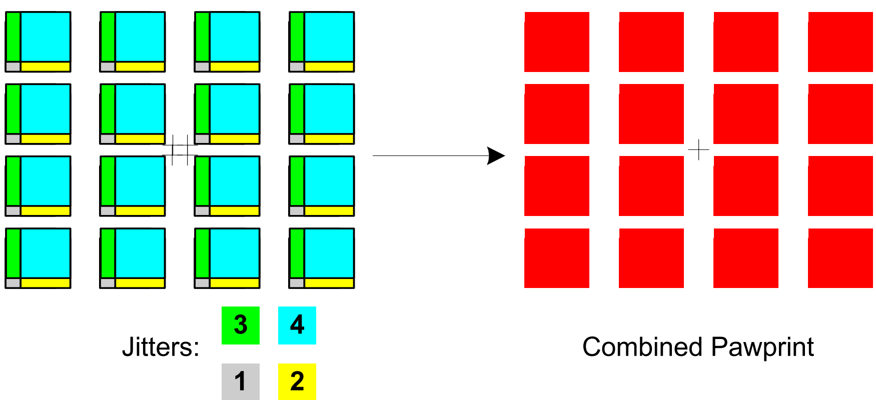


Figure 14 Combining Exposures with Jittering

If the exposures are combined with the idea of interleaving the pixels to improve the sampling in excellent seeing conditions the sub-pixel movements (which must be <3 arcsec and a half integer number of pixels) are referred to as "microstepping", as shown in Figure 15.

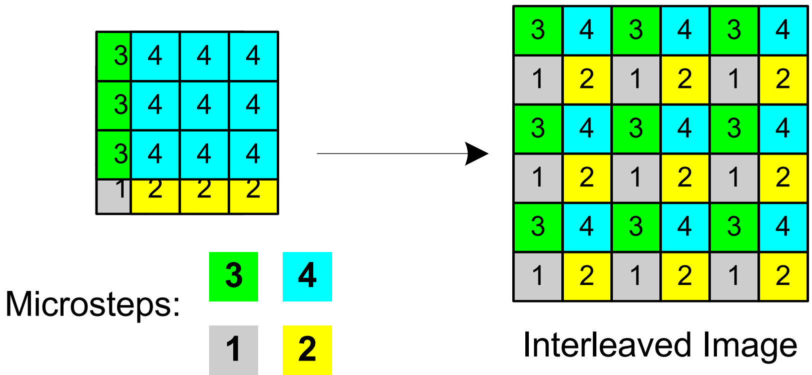


Figure 15 Combining Exposures with Microstepping

All jittering and microstep offsets are also made with respect to X,Y focal plane coordinates.

2.2.2 Observation parameters

A survey observation is made using an Observation Block containing two templates, which are supplied with the following observation parameters:

- An acquisition template containing:
  - The celestial coordinates of the tile reference position (normally its centre).
  - Any guide star and LOWFS stars to confirm acquisition of the reference field (optional).
  - Any science filter to position in the beam during the telescope movement (optional).

  See section 5.18.2.1 on page 116 and section 11.7.4.1 on page 216 for details.

- A tile template containing:
  - The name of a tile pattern giving the telescope offsets (with respect to the above reference position) for each of the pawprints making up the tile, plus an optional scaling factor. See section 5.14 on page 107.
  - A set of guide star and LOWFS stars for each pawprint position.
  - The name of any jitter pattern giving the telescope offsets (with respect to each pawprint position) used for jittering, plus an optional scaling factor (section 5.14).
  - The name of any microstep pattern giving the telescope offsets (with respect to each jitter position) used for microstepping (section 5.14).
  - A list of science filters to be used.
  - A nesting pattern used to sequence the above telescope offsets and science filters.
  - The exposure time, or a set of exposure times for each science filter.

  See section 5.18.2.3 on page 118 and section 11.7.6.3 on page 245 for details.

2.2.3 Survey efficiency

The most efficient survey will be one which:

- only seeks guide star confirmation from the operator when a failure is detected (i.e. AG.CONFIRM=F);
- either does not verify the reference position in the acquisition template using guide stars (AG.START=F in the acquisition template), or which begins each pawprint pattern with (0.0, 0.0) — i.e. makes the reference position the same as the first pawprint position;
- selects the first science filter within the acquisition template (so it is selected in parallel with the telescope movement).
- minimises the number of science filter movements;
- uses tile and jitter patterns which minimise the number of telescope movements;
- uses the minimum exposure time required for the job;
- schedules the Observation Blocks to minimise telescope movements.

Note that the VISTA telescope system does not make any distinction between large movements (known on the VLT as a "preset") and small movements (known on the VLT as "offsets"), so the telescope movements made by an acquisition template or by one of the observation templates are just as efficient. The data acquisition efficiency is not improved, for example, by combining tiles together in a single Observation Block (as long as the OBs are scheduled efficiently).

2.2.4 Observation preparation

Observations for a typical ESO/VLT instrument are prepared using the Phase 2 Proposal Tool (P2PP). The interface between ESO/VLT instrument software and this tool is described in [RD31]. ESO/VLT instrument software provides an instrument summary file and a set of template signatures describing the capabilities of the instrument. Observations are defined and executed using Observation Blocks. VISTA users need the ability to prepare automatically an entire survey over a defined area of the sky, including the automatic choice of the guide and LOWFS stars for each pawprint, [RD14]. A Survey Area Definition Tool (SADT) is provided for this purpose, [RD15].

Figure 16, below, shows the relationship between the VIRCAM software, BOB, P2PP and the SADT. The VIRCAM software makes available an "instrument package" for use by P2PP and SADT. (The instrument package is contained in the "vcotsf" software module — see section 2.3.1.)

The communication links shown in black are the ones used by any ESO/VLT instrument. The VIRCAM software uses the following additional communication links, shown in blue:

- The collection of tile, jitter and microstep patterns is provided in the instrument package as a set of parameter (PAF) files, as described in section 5.14 on page 107. The user may choose from the set of available patterns.

- P2PP communicates the guide and LOWFS stars chosen by the SADT for each pawprint using PAF files delivered using "SEQ.REF.FILEi" parameters of type "paramfile" and embedded within the tile template contained within an Observation Block. See [RD31] for a description of "paramfile" parameters, and see sections 5.18 and 11.7.6 for tile template details.
- When BOB receives an Observation Block containing "paramfile" parameters it recreates the original PAF files, which can then be read by the VIRCAM software.

NOTE: The ESO software requires that "paramfile" parameters are always provided with a value, and it is not possible to provide a default value. This means that all the "SEQ.REF.FILEi" parameters provided by a template must be used. Since it is possible for tile patterns to different numbers of pawprints, a different variation of the tile template signature is provided within the VIRCAM software for each possible number of pawprints (e.g. VIRCAM_img_obs_tile1, VIRCAM_img_obs_tile3 and VIRCAM_img_obs_tile6).

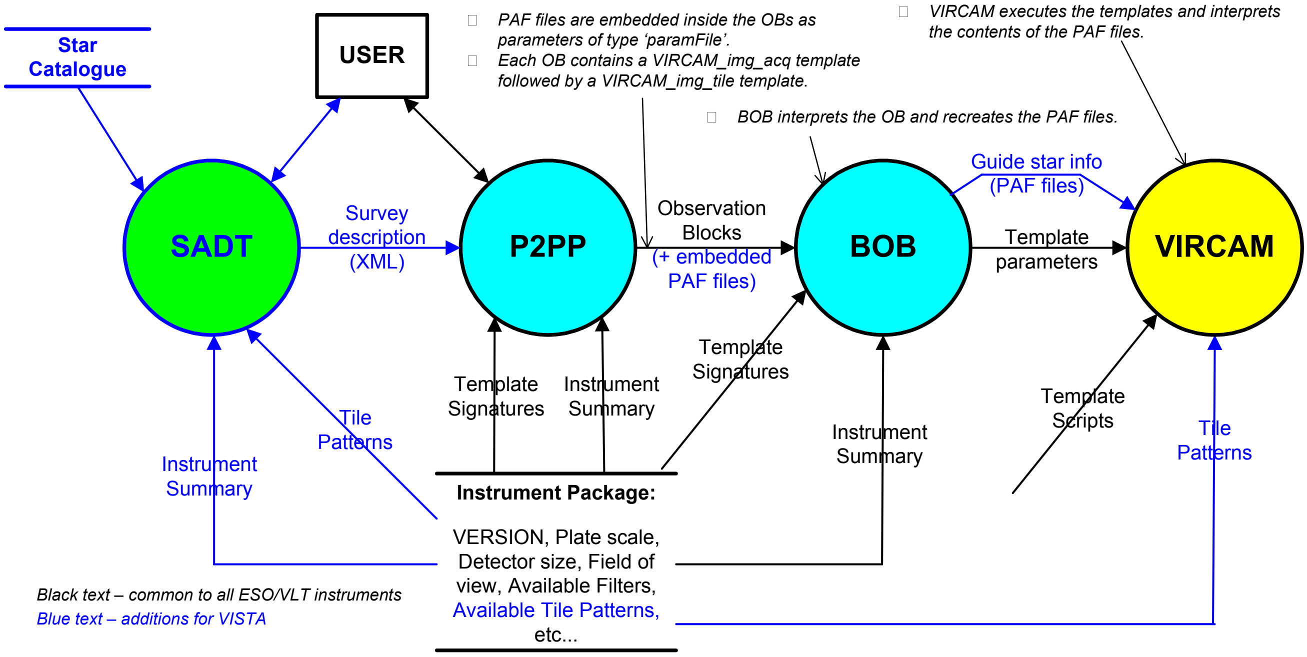


Figure 16 Relationship with Observation Preparation Tools

2.3 Software Architecture

The overall software architecture is shown in Figure 17 below. This is the same as the standard ESO/VLT architecture, [RD30], with the addition of the HOWFS image analysis process, which acts more like a local data reduction system than as an instrument subsystem, and is not controlled through the Observation Software.

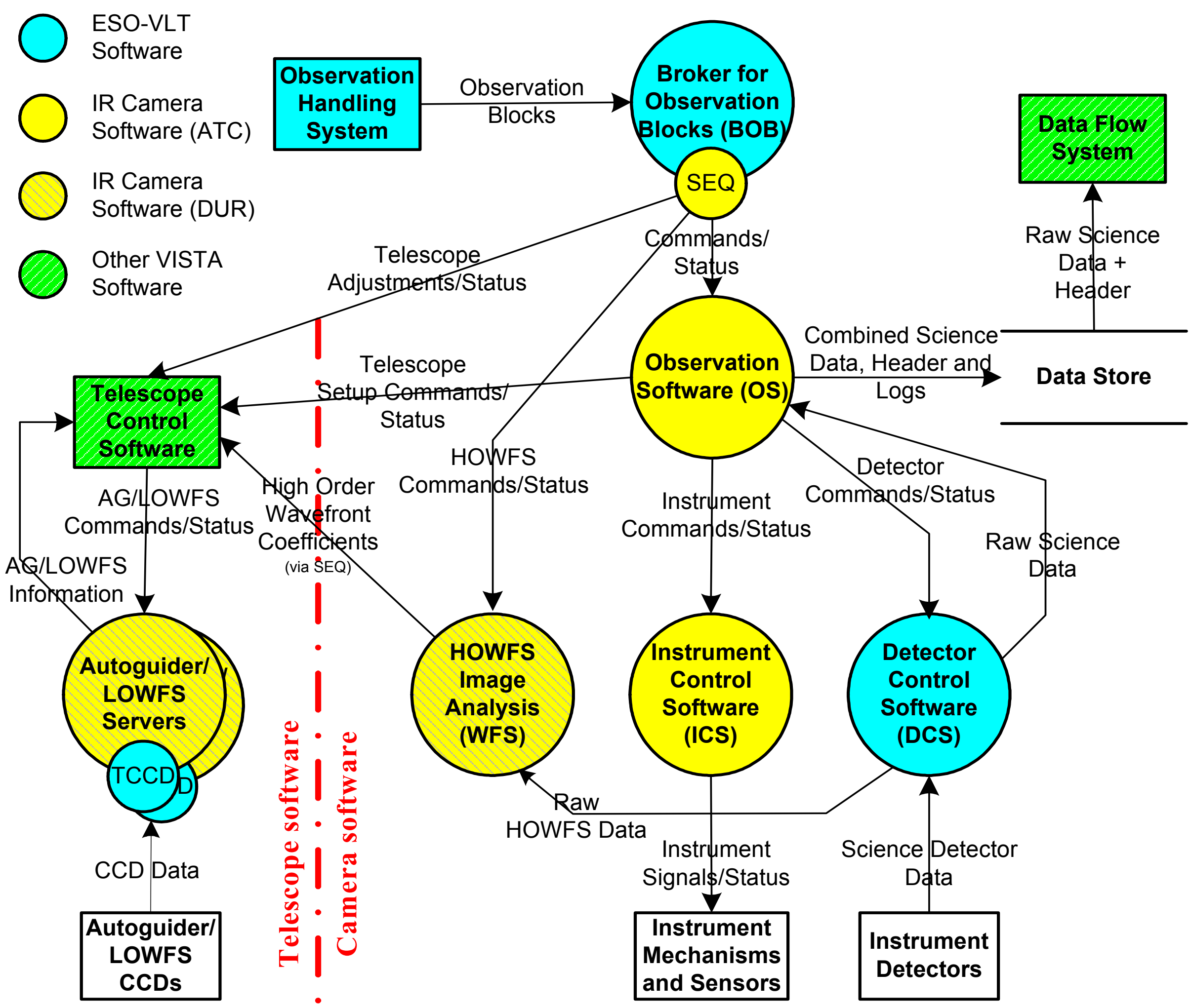


Figure 17 VISTA IR Camera (VIRCAM) Software Architecture

For more details of the software architecture, see the "*Observation Software Design Description*", [RD4].

2.3.1 Software Modules

The VISTA IR software is divided into the following software modules, which are similar to those used by the ESO-VLT Template Instrument, [AD2]. In the "Platform" column "WS" means the instrument workstation.

Module	INS Package	Platform	Description
vcins	N/A	WS	Software integration module.
dicVIRCAM	N/A	WS	Dictionary module.
ICS:			
vci	ICS	WS + ICS LCU	ICS workstation front-end and ICS control module..

vcipan	ICS	WS	ICS stand-alone GUI.
vcihb	ICS	ICS LCU	ICS heart beat device module.
vcilsc	ICS	ICS LCU	ICS Lakeshore 332 device module.
vcilsm	ICS	ICS LCU	ICS Lakeshore 218 device module.
vcitpg	ICS	ICS LCU	ICS Pfeiffer TPG device module.
OS:			
vco	OS	WS	OS control and server module.
vcopan	OS	WS	OS GUI.
vcoseq	OS	WS	Observation template scripts.
vcotsf	OS	WS	Instrument package, containing the observation template signature files and instrument summary file.
MS:			
vcmcfg	MS	WS	Instrument configuration files.
vcmseq	MS	WS	Maintenance template scripts.
vcmtsf	MS	WS	Maintenance template signature files and technical instrument summary file.
HOWFS:			
vchoia	HOWFS	WS	HOWFS image analysis module[6].
vtialib	HOWFS	WS	HOWFS image analysis library (shared with LOWFS software in the VISTA TCS).
vchpan	HOWFS	WS	HOWFS stand-alone GUI.
DCS:			
vcd	DCS	WS	VISTA DCS test scripts and engineering display module.
vcdacq	DCS	WS + IRACE	VISTA IRACE acquisition module.
vcdcfg	DCS	WS	VISTA IRACE configuration module.
vcrtd	DCS	WS	VISTA operations realtime data display module

The modules are laid out in a directory structure shown in Figure 18.

[6] The vchoia module is based on the LOWFS image analysis module, vtact, used in the VISTA TCS software.

VIRCAMSource

- dicVIRCAM — *VIRCAM dictionary module*
- vcins — *VIRCAM software integration module*
- DCS — *VIRCAM Detector Control Configuration Software*
  - vcd — *DCS test module*
  - vcdacq — *VISTA IRACE acquisition module*
  - vcdcfg — *VISTA IRACE configuration module*
  - vcrtd — *VISTA realtime display module*
- ICS — *VIRCAM Instrument Control Software*
  - vci — *ICS control and server module*
  - vcihb — *ICS heart beat special device module*
  - vcilsm — *ICS Lakeshore 218 special device module*
  - vcilsc — *ICS Lakeshore 332 special device module*
  - vcitpg — *ICS Pfeiffer TPG256A special device module*
  - vcipan — *ICS GUI module*
- MS — *VIRCAM Maintenance Software*
  - vcmcfg — *Instrument configuration module*
  - vcmseq — *MS sequencer scripts module*
  - vcmtsf — *MS template signature module*
- OS — *VIRCAM Observation Software*
  - vco — *OS control and server module*
  - vcopan — *OS GUI Module*
  - vcoseq — *OS sequencer scripts module*
  - vcotsf — *OS template signature module*
- HOWFS — *VIRCAM HOWFS WFS Software*
  - vchoia — *HOWFS image analysis module*
  - vtialib — *HOWFS image analysis library*
  - vchpan — *HOWFS GUI module*

Figure 18 VISTA IR Camera Software Modules and Source Layout

2.3.2 Environments

The VISTA IR software uses the following environments (see also Figure 11 on page 38).

2.3.2.1 VIRCAM Instrument Environments

Environment	Platform	Description
wvcam	Instrument Workstation (wvcam)	VIRCAM instrument workstation CCS environment
lvcics1	lvcics1 ICS LCU (lvcics1)	VIRCAM ICS LCU LCC environment
(none)	IRACE number cruncher 1 (wvcirc1)	IRACE number cruncher 1 software (“no CCS”)
(none)	IRACE number cruncher 2 (wvcirc2)	IRACE number cruncher 2 software (“no CCS”)

2.3.2.2 VISTA Telescope Control Environments

Environment	**Platform**	**Description**
wvttcs	Telescope Workstation (wvttcs)	VISTA telescope control CCS environment
wvt0tcs	Telescope Workstation (wvttcs) or Instrument Workstation (wvcam)	VISTA simulator telescope control CCS environment
lvtag1	Autoguider LCU 1 (lvtag1)	VISTA autoguider LCU 1 (py) LCC environment
lvtag2	Autoguider LCU 2 (lvtag2)	VISTA autoguider LCU 2 (ny) LCC environment
wvtia1	LOWFS image analysis workstation 1 (wvtia1)	VISTA LOWFS 1 (py) CCS environment
wvtia2	LOWFS image analysis workstation 2 (wvtia2), or 1 (wvtia1).	VISTA LOWFS 2 (ny) CCS environment
lvtwfs1	LOWFS LCU 1 (lvtwfs1)	VISTA LOWFS LCU 1 (py) LCC environment
lvtwfs2	LOWFS LCU 2 (lvtwfs2)	VISTA LOWFS LCU 2 (ny) LCC environment

2.3.3 Standards

The VISTA IR Camera software is based on the standard VLT packages:

- The overall software is based on the ESO-VLT Template Instrument, [AD2].
- The IR DCS uses the IRACE software, [RD39].
- The OS uses the BOSS software, [RD48].
- Templates are based on the “INS common software for templates” packages, [RD49].
- The ICS uses the “Base ICS” software, [RD46] and [RD47].
- Instrument software installation uses the pkgin package, [RD43].
- Instrument configuration uses the ctoo package, [RD50].
- Instrument startup/shutdown uses the stoo package, [RD51].
- The CMM package is used for configuration control, [RD37].

The software follows these VISTA standards:

- Software requirements, [AD2]

and these ESO-VLT standards:

- Software requirements, [RD27] and [RD29]
- Programming standards, [RD25].

The software testing facilities are compatible with:

- The “Tools for Automated Testing” package, [RD36].

3 INSTALLATION GUIDE

3.1 Requirements

3.1.1 Hardware

Section 2.1.6 describes and Figure 11 shows the computing hardware that must be available to run the complete VISTA IR software. The following configurations are possible.

3.1.1.1 Simulation mode on the workstation only

This configuration requires just the Instrument Workstation. The TCS, IRACE and instrument LCU are simulated.

Figure 19 Hardware required for workstation only simulation

3.1.1.2 "Hardware simulation mode" on the workstation and LCU only

This configuration requires the Instrument Workstation and ICS LCU, but the ICS LCU only requires a CPU card. Communication with the instrument LCUs is tested but all access to the instrument hardware is simulated.

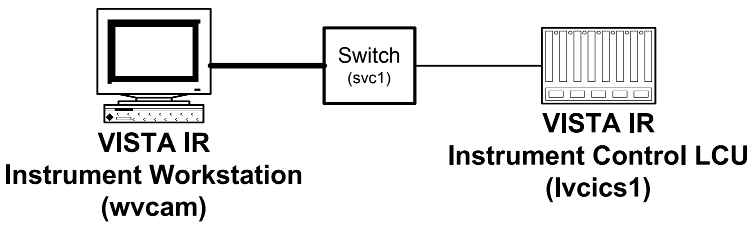


Figure 20 Hardware required for LCU hardware simulation mode

NOTE: It is possible to have some of the LCU hardware present and some of the hardware simulated, in which case the LCU can be configured to simulate the missing hardware. Some devices depend on each other, so the higher level software will not work if one device is simulated and another is not. In particular:

- The filter wheel software depends on the digital I/O device.
- The heartbeat software depends on the digital I/O device.
- The thermal control software requires temperatures read by all the Lakeshore devices, so all the devices must either be present or simulated.

3.1.1.3 Hardware for data throughput performance testing

This configuration requires the IRACE number crunchers, the Gigabit Ethernet network equipment, the Instrument Workstation and any fast disks needed to achieve the high speed data read/write performance. Optionally, a data handling workstation may also be added to test the performance of the onward link from the VLT online archiver (VOLAC).

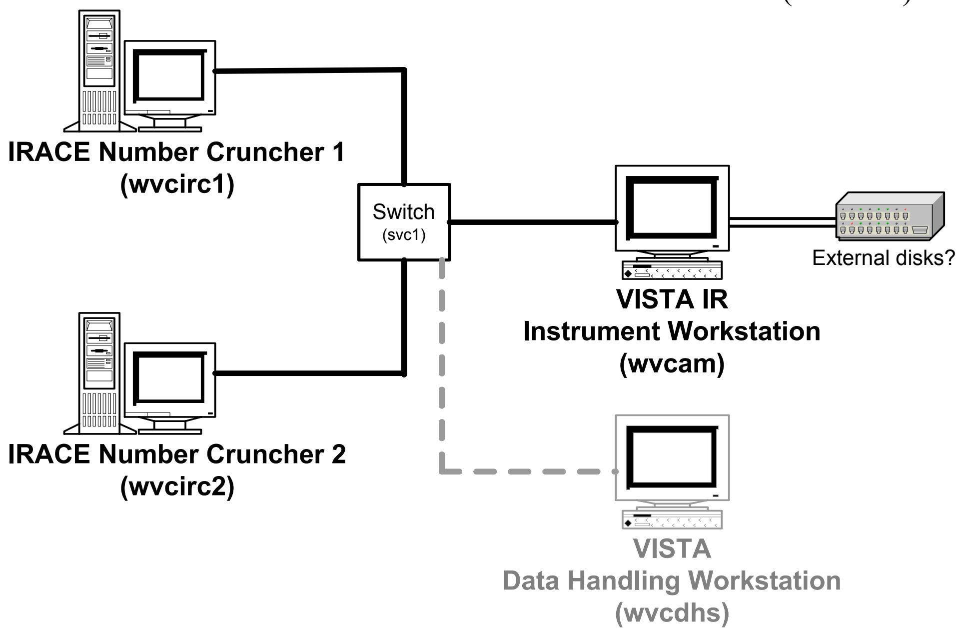


Figure 21 Hardware required for data throughout performance test

3.1.1.4 Hardware for instrument stand-alone mode

This configuration requires all the hardware present on the instrument “vc” subnet. The instrument software drives the complete instrument but communication with the TCS is simulated. (This mode is useful for controlling the instrument in the instrument prep. lab. without affecting the VISTA telescope).

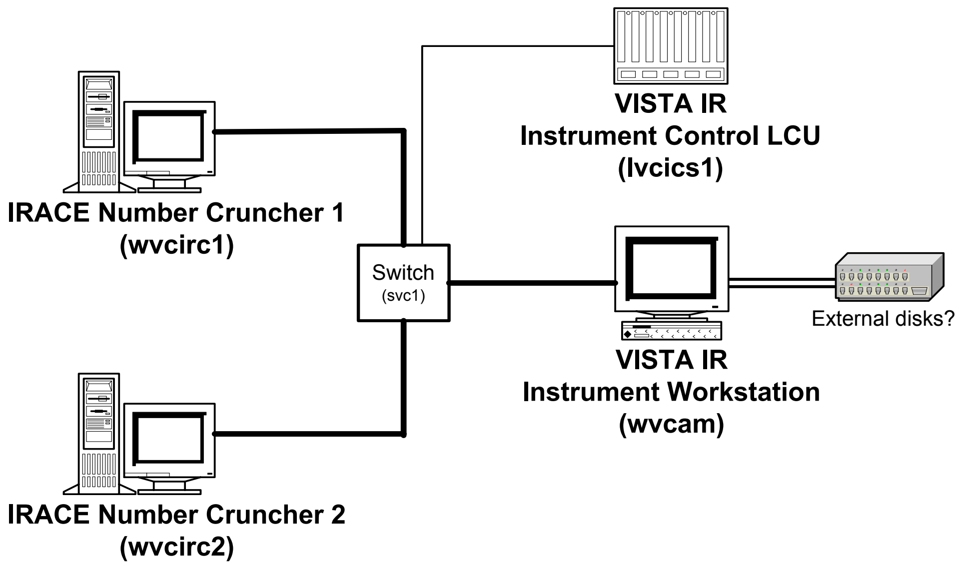


Figure 22 Hardware required for stand-alone testing

3.1.1.5 Complete hardware for instrument operation

This requires all the hardware shown in Figure 11 on page 38, i.e. everything shown in Figure 22, above, plus the telescope hardware.

3.1.2 Software

The operating systems installed on the VIRCAM computers must be compatible with the VLT software installation (see [RD41] for the operating system installation procedure).

The VLT software ("JAN2006" version or higher) must be installed on the instrument workstation and configured for CCSlite (see [RD41] and [RD42] for the software installation procedure). The VLT software must also be installed on the IRACE number crunchers and configured for "No CCS".

3.1.3 Environment variables

The VIRCAM software requires the following environment variables to be defined before the software can be installed. Check the definition files in the ~vcmgr/.pecs subdirectory and, if necessary, follow the PECS setup procedure described in [RD42].

Environment variable	Where declared	Comments
RTAPENV	PECS	The main workstation environment which must be declared as follows: `export RTAPENV=wvcam`
INTROOT	PECS	Instrument root directory, for example: `export INS_ROOT=/insroot/vcam`
INS_ROOT	PECS	Integration root directory, for example `export INTROOT=/introot/vcam`

See section 4.2 on page 57 for additional environment variables that will be needed to run the software. At this stage only the "PECS" environment variables are needed.

3.2 Installation Procedure

The installation procedure always starts with these steps:

- Log on as user "vcmgr" on the instrument workstation.
- Run the *vccShow* command (or the *vccEnv* utility) and check that the environments listed in section 2.3.2 are known and correctly configured in the ACC database.
- Make sure the installation environment variables are defined correctly (see section 3.1.3 above). If you are installing a new version of the VIRCAM software, make sure the $INTROOT and $INS_ROOT directories are empty.
- Create an empty source directory and cd into it:

  ```
  % mkdir $HOME/VIRCAMSource
  % cd $HOME/VIRCAMSource
  ```

  This directory will be referred to in this document as the "top level source directory"[7].
- Optional: Make sure the $VLTDATA/ENVIRONMENTS directory does not contain any wvcam or lvcics1 environments owned by someone else:

  ```
  % ls -l $VLTDATA/ENVIRONMENTS/wvcam*
  % ls -l $VLTDATA/ENVIRONMENTS/lvcics1*
  ```

  This check is only necessary on development workstations where more than one user may have been working on the environment. At Paranal, the environments should always be owned by "vcmgr".

[7] It is possible to make different directories to contain different versions of the software (for test purposes), but only one version may be installed into the INTROOT and INS_ROOT directories at a time. Always start with empty INTROOT and INS_ROOT directories before rebuilding a new version of the software.

The next step of the installation procedure recognises three targets, as described in the following table:

Target name	Environments used	Description
wvcam	wvcam lvcics1 wvttcs	The default. A normal installation on the instrument workstation. (Many ESO/VLT instruments declare this target as PARANAL.)
TEST	wvcam lvcics1 wvt0tcs	development and test installation on the instrument workstation, which uses the VISTA TCS simulator.
NC	(none)	An installation on an IRACE number cruncher.

The software may be built using the "pkginBuild" command described in [RD43]. If no "-target" option (or TARGET environment variable) is specified, the "wvcam" target is built by default; and if no "-env" option is specified, all the environments defined by the specified target are built (as listed above). Some building options are described below.

After the vcins module has been built and installed, the command "vcinsHelp" may be used to get further information. Don't miss the last installation step described in section 3.2.6 below.

3.2.1 Building the default configuration (workstation + LCU + TCS)

Extract and build all the VIRCAM modules, using the default "wvcam" target[8] and all the default environments.

```
% cmmCopy vcins
% pkginBuild vcins
```

The command "cmmCopy vcins <version>" can be used if a particular version of the VIRCAM software is needed. Use "cmmHistory vcins" for a description of the versions available. By default, the latest version is installed. HINT: pkginBuild does not generate much output, but an installation log is written to the file "./INSTALL/pkginBuild.log". A more detailed running commentary on the build can be viewed by typing

```
% tail -f ./INSTALL/pkginBuild.log
```

on another terminal (from within the top level source directory).

3.2.2 Building a stand-alone configuration (workstation + LCU but no TCS)

```
% cmmCopy vcins
% pkginBuild vcins -target TEST -env wvcam lvcics1
```

[8] Assuming there is no TARGET environment variable naming a different target.

3.2.3 Building a stand-alone configuration (workstation + LCU + simulated TCS)

```
% cmmCopy vcins
% pkginBuild vcins -target TEST
```

3.2.4 Building the workstation-only simulation configuration

```
% cmmCopy vcins
% pkginBuild vcins -target TEST -env wvcam
```

3.2.5 Building the TCS simulation configuration (no LCU)

```
% cmmCopy vcins
% pkginBuild vcins -target TEST -env wvcam wvt0tcs
```

3.2.6 Important note about first time software installation

After building the software for the first time, it is necessary to log out and log in again so that the environment variables defined automatically by the build procedure can be created properly. The same is true of any build procedure which may have modified an environment variable (e.g. after changing an instrument configuration parameter).

3.3 IRACE Software Installation

3.3.1 Installation of IRACE software on the instrument workstation

The IRACE modules are built and installed on the instrument workstation when the VIRCAM modules are built, as described above. No additional IRACE software installation procedure is needed.

3.3.2 Installation of IRACE software on the IRACE number crunchers

It is important that the IRACE modules installed on the instrument workstation and IRACE number crunchers are compatible. The IRACE number crunchers should be installed with the same release of the ESO/VLT software as installed on the instrument workstation. The number crunchers are normally installed with a subset of the ESO/VLT software called "NoCCS[9]". So, for example, if the instrument workstation is installed with "JAN2006 CCSlite" the number crunchers should be installed with "JAN2006 NoCCS".

The IRACE software on the number crunchers can be made compatible with a particular release of the VIRCAM software using a similar installation procedure which specifies a target of "NC", like this:

- Log on as user "vcmgr" on a number cruncher. There is no need to check environments because there aren't any.
- Make sure the INTROOT and INS_ROOT environment variables exist and point to some sensible directories.

[9] because CCS environments are not used on the IRACE number crunchers.

- Create an empty source directory and cd into it:

```
% mkdir $HOME/VIRCAMSource
% cd $HOME/VIRCAMSource
```

- Get the vcins module, retrieve the IRACE-dependent modules (optional) and build the NC target:

```
% cmmCopy[10] vcins
% pkginBuild[10] vcins -target NC -step RETRIEVE
% pkginBuild vcins -target NC -tostep BUILD_MOD
```

This procedure should build and install just the IRACE-dependent modules.

3.3.3 Important note about IRACE simulation

This section may be skipped if installing the VIRCAM software on the operational system at Paranal.

When the IRACE software is simulated without the number crunchers present (e.g. in workstation only simulation mode or in hardware simulation mode) it uses simulated acquisition processes running on the instrument workstation. These processes need to be granted root privileges so they may be run at a sufficiently high priority. The following additional steps are necessary after installing the software:

```
% cd $INTROOT/bin
% su root
% chown root virgo1 virgo2
% chmod a+s virgo1 virgo2
```

Failing to do this may cause “ring buffer overrun” errors from IRACE, when the processes can’t keep up.

A side effect of changing the ownership of these files is that the “pkginBuild” utility will complain at its CREATE_ROOTS stage that it can’t manipulate these files, with these error messages:

```
INTROOT:  using /introot/vcam: FAILURE.

ERROR: Script error: chmod: changing permissions of
`/introot/vcam/bin/virgo1':
```

[10] The cmmCopy command assumes the IRACE number crunchers are able to access the ESO/VLT CMM repository over the network. If this is not the case, execute the first two commands on the instrument workstation, copy all the retrieved modules to both IRACE number crunchers and then execute the third command on each IRACE number cruncher from the directory containing the modules.

```
Operation not permitted (retcode 1).
See error log file for details.
```

and will abandon the procedure just before building the environments. This feature can be worked around by repeating "pkginBuild" from the "BUILD_ENV" step, like this:

```
% pkginBuild vcins -env wvcam lvcics1
(stops with error message)
% pkginBuild vcins -env wvcam lvcics1 -fromstep BUILD_ENV
```

3.4 Checking the Installation

After installing and building the software, the following commands can be used to check that the CCS workstation (wvcam) and LCU (lvcics1) environments are running::

```
% vccEnvCheck -e wvcam
% vccEnvCheck -e lvcics1
```

If the workstation has just been rebooted, the following command is also useful for checking that the ACC database is correctly configured and the msql daemon has been successfully restarted (it also lists all the available environments):

```
% vccShow
```

A more thorough check of the installation can be carried out by following the procedures described in section 6.4 on page 129.

3.5 Checking the communication with the VISTA TCS environment

3.5.1 Command communication via the message system

The command

```
% echo $TCS_ENVNAME
```

will indicate which TCS environment the VIRCAM software is configured to interact with. It should display "wvttcs" for the real VISTA TCS or "wvt0tcs" for the VISTA TCS simulator. The status of the VISTA TCS environment may be checked with the command

```
% vccEnvCheck -e $TCS_ENVNAME
```

and the communication with the VISTA TCS may be tested with the command

```
% msgSend $TCS_ENVNAME vtifControl VERSION ""
```

If the TCS environment is running properly, the command should return a version number string.

3.5.2 Database updates via the scan link

The status of the scan link, which copies database information from the TCS environent to the wvcam environment, can be verified with the command

```
% scanei &
```

which brings up the scan manager. The display should show "ENABLED" next to the appropriate TCS environment and "ACTIVE" for the overall status. (The same panel can be used to verify the scan link from the LCU environment, lvcics1.)

4 Operator's Guide

4.1 Getting Help

The command

```
% vcinsHelp
```

will display a list of commands available. Where man pages are available, help may be obtained with the command "man <command>", for example

```
% man vcinsStart
```

Nearly every VIRCAM panel has an associated man page. Selecting "Help→Display man page" from any panel will display detailed information on how to use the panel. Some panels also contain other useful information and pictures in their "Help" menu.

4.2 Environment Variables

The VIRCAM software requires the following environment variables to be defined before the software can be executed.

Environment variable	Where declared	Comments
RTAPENV	PECS	The main workstation environment which must be declared as follows: `export RTAPENV=wvcam`
INTROOT	PECS	Instrument root directory. This must point to the directory where the instrument software has been installed, for example: `export INS_ROOT=/insroot/vcam`
INS_ROOT	PECS	Integration root directory. This must point to the directory where the instrument software has been installed, for example `export INTROOT=/introot/vcam`
DISPLAY	LOGIN	The location of the display device. This is normally defined automatically during the Linux login procedure.
TCS_ENVNAME	OSB	The name of the environment running the VISTA TCS (OCS.TEL.ENVNAME). For example "wvttcs".
TCSID	OSB	The name of the VISTA TCS (OCS.TEL.NAME). This should be "VISTA"[11].

[11] N.B. This definition of TCSID (as used by the startup tool [RD51]) differs from that used by the VISTA and VLT telescope software, where it is an integer used to select a real or simulated environment. The two definitions should not be confused.

OLAS_ID	OSB	Instrument ID as known to the VLT on-line archiver (INS.CON.ID). Should be "VIRCA".
TCOM_SRV	OSB	TCOM server port used by IRACE.
SDMA_HOST	OSB	Root name of IRACE number cruncher host. Should be "wvcirc".
SDMA_HOST1	VCINS	Name of IRACE number cruncher host 1 – wvcirc1
SDMA_HOST2	VCINS	Name of IRACE number cruncher host 2 – wvcirc2
SDMA_CMD1	VCINS	IRACE command port 1
SDMA_CMD2	VCINS	IRACE command port 2
SDMA_CMD3	VCINS	IRACE command port 3
SDMA_CMD4	VCINS	IRACE command port 4
SDMA_DATA1	VCINS	IRACE data port 1
SDMA_DATA2	VCINS	IRACE data port 2
SDMA_DATA3	VCINS	IRACE data port 3
SDMA_DATA4	VCINS	IRACE data port 4

The "PECS" environment variables are defined in files in the ~vc/.pecs subdirectory. RTAPENV, INTROOT and INS_ROOT *must* have the same definitions for the "vcmgr" and "vc" accounts. To correct these variables, edit the ".pecs" files or follow the PECS setup procedure described in [RD42].

The "OSB" and "VCINS" environment variables are defined automatically through the file "$INTROOT/config/vcins-misc-all.env", which (if the software has been installed correctly) should be executed automatically on login[12]. Make sure this happens. The "OSB" environment variables are the standard ones determined from any ESO/VLT instrument configuration, and may be viewed with the command:

```
% osbEnvSet VIRCAM
```

The variables labelled "VCINS" are VIRCAM-specific additions (due to the fact that VIRCAM uses more than one IRACE number cruncher), and are defined explicitly in the "vcins-misc-all.env" file. Try commands such as:

```
% env | grep SDMA
```

to check all the environment variables are declared.

4.3 System Startup

- Log on as user "vc" on the instrument workstation.
- Ensure the environment variables are defined correctly, as shown above.
- Optional: Ensure that all files are available with the correct permissions by typing

[12] You will be prompted to log out and log back in again after installing and building the software.

```
% vcinsCheckPermissions
```
[13]

- Start the VIRCAM software

```
% vcinsStartup &
```

This will bring up the startup configuration screen shown in Figure 23 (which shows the normal operational configuration). The panel may be used to select whether the TCS, IRACE or ICS subsystems are available, whether to run a subsystem in workstation simulation mode, and whether to start BOB, an alarm display or the HOWFS image analysis process. It may also be used to control which panels are started. The first three lights shown in the “Check list” should all be switched on (i.e. coloured green) if the instrument is to be used for normal operation. The filter wheel behaves normally when these three lights are on. The “FW intermediate stop” light should be off if survey speed is important, but this can be on during periods when filter wheel checking is regarded as more important (when this option is enabled the filter wheel makes intermediate stops to check that the in-position switch deactivates outside the science positions).

[13] This command is especially useful when running the software from the “vc” account for the first time, since testing the software from “vcmgr” can sometimes leave behind files with incorrect permissions.

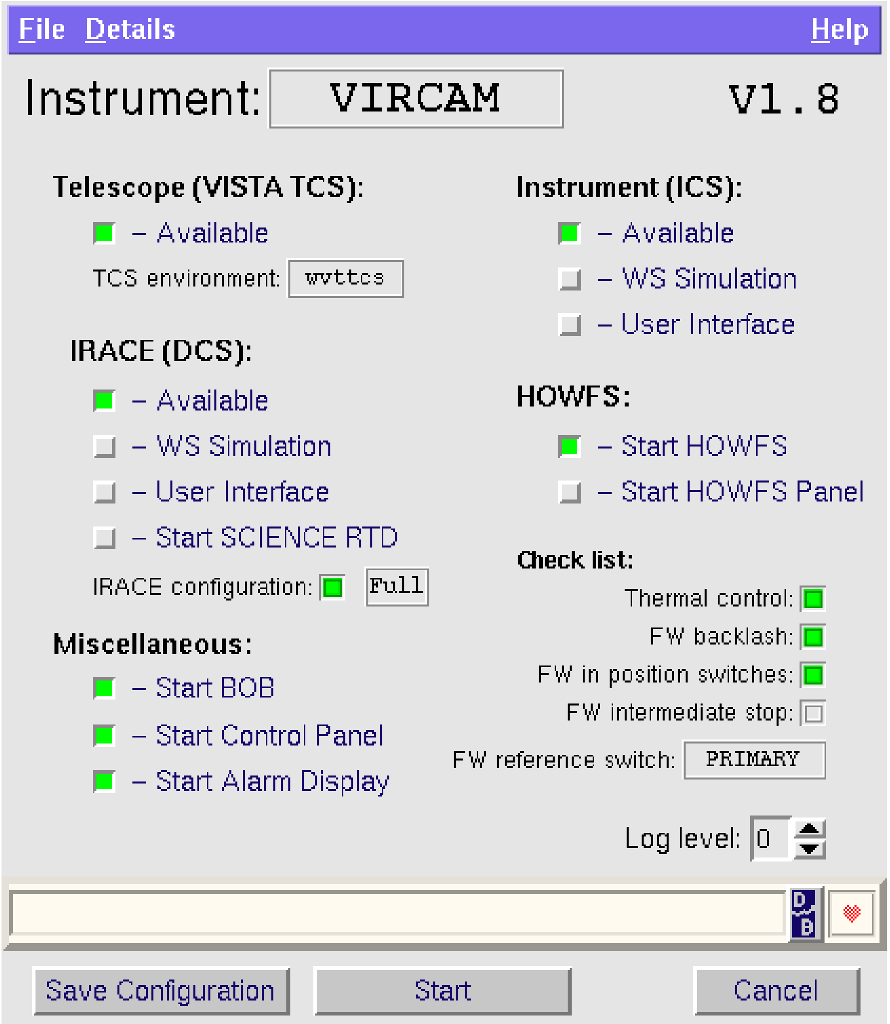


Figure 23 VIRCAM software startup panel

In general, green and grey lights on this panel are good; red lights indicate something wrong with the configuration, which needs to be corrected by an expert user (see below). Online help on this panel can be obtained by selecting "Help → Display man page".

Pressing the START button will start the VIRCAM software. Once configured, the command

```
% vcinsStart [-restart]
```

may be used to bring up the software without going through the initial configuration panel. (The optional "-restart" option may be used to force all processes to be restarted).

4.4 *Expert System Startup*

- Log on as user "vcmgr" on the instrument workstation (only vcmgr has access to the expert startup panels).
- Ensure the environment variables are defined correctly, as shown above.

- Start the VIRCAM software with the engineering/development alternative to vcinsStartup

```
% vcinsStartupDev &
```

This will bring up the expert startup configuration screen shown in Figure 24, below. This panel should *only* be used by experts. It can be used to define configurations where only parts of the hardware are available, or for engineering reasons switch off software functions that are normally used. For example, the "Use in-position switches" option can be switched off to continue operations when the filter wheel in-position switches are broken. The SECONDARY reference switch can be selected when the PRIMARY switch is broken. The panel can also be used to configure the software to use a simulated IRACE system or use the VISTA TCS simulator (which will require a software environment restart). Online help on this panel can also be obtained by selecting "Help → Display man page".

NOTE: Once the expert startup panel has been used to change the instrument configuration parameters, those parameter values become the defaults the next time the software is started – even for the "vc" user.

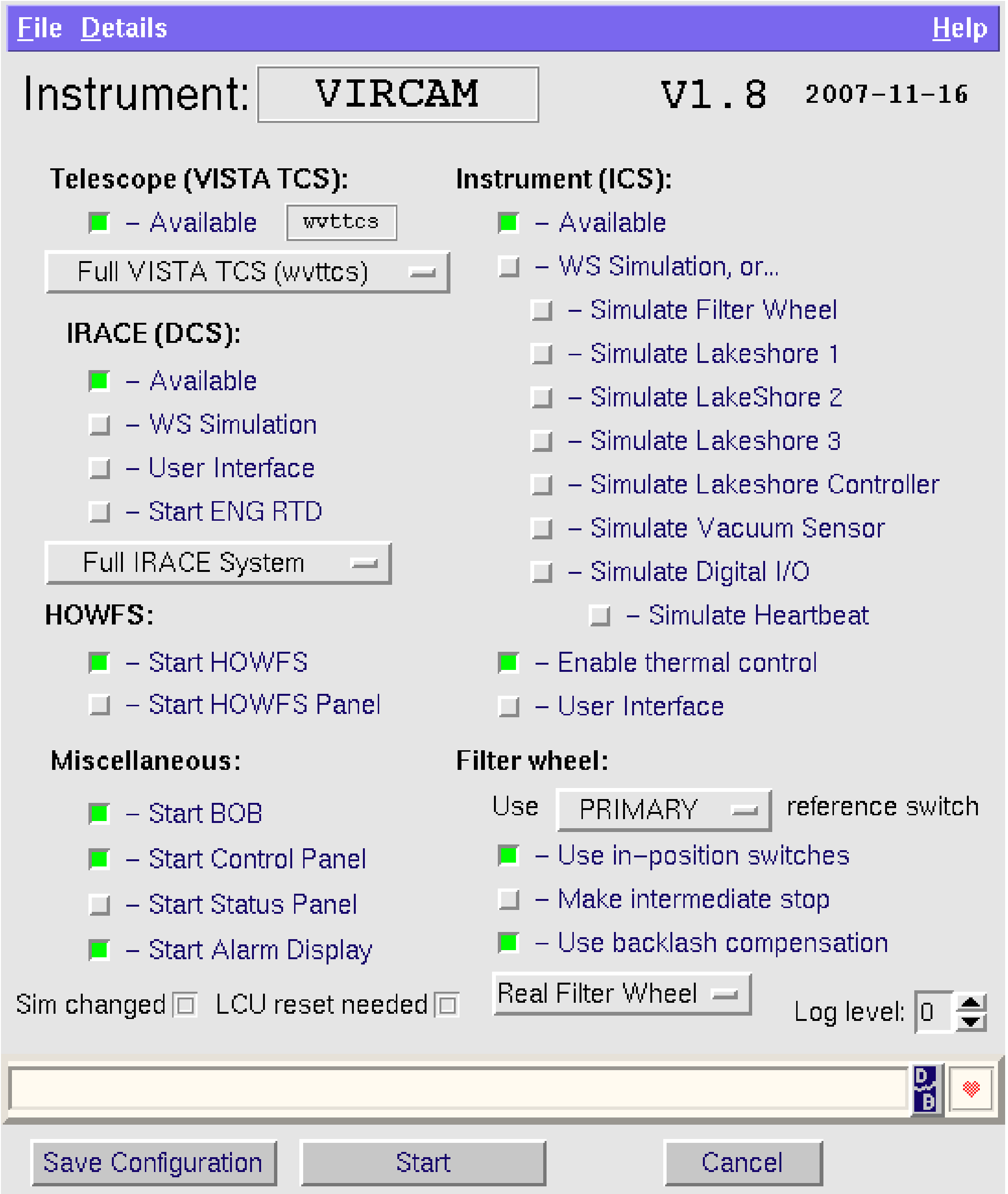
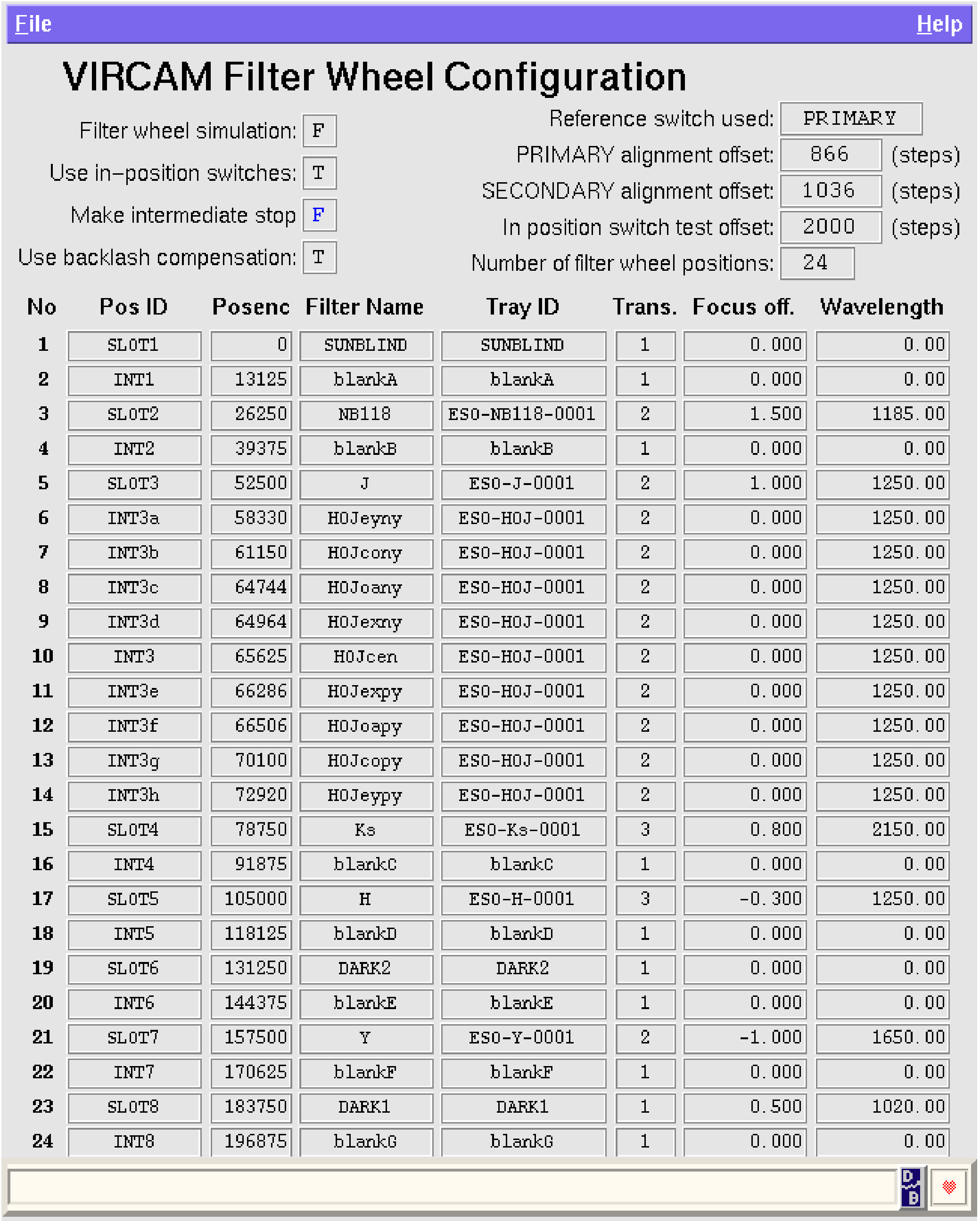


Figure 24 VIRCAM software expert startup panel

4.5 Configuration Display

Various panels are available to display the current VIRCAM configuration settings, which are described section 6 on page 119. These panels are also available through the "Details" menu of the vcinsStartup panel.


File Help

VIRCAM Filter Wheel Configuration

Filter wheel simulation: F
Use in-position switches: T
Make intermediate stop F
Use backlash compensation: T
Reference switch used: PRIMARY
PRIMARY alignment offset: 866 (steps)
SECONDARY alignment offset: 1036 (steps)
In position switch test offset: 2000 (steps)
Number of filter wheel positions: 24

No	Pos ID	Posenc	Filter Name	Tray ID	Trans.	Focus off.	Wavelength
1	SLOT1	0	SUNBLIND	SUNBLIND	1	0.000	0.00
2	INT1	13125	blankA	blankA	1	0.000	0.00
3	SLOT2	26250	NB118	ESO-NB118-0001	2	1.500	1185.00
4	INT2	39375	blankB	blankB	1	0.000	0.00
5	SLOT3	52500	J	ESO-J-0001	2	1.000	1250.00
6	INT3a	58330	HOJeyny	ESO-HOJ-0001	2	0.000	1250.00
7	INT3b	61150	HOJcony	ESO-HOJ-0001	2	0.000	1250.00
8	INT3c	64744	HOJoany	ESO-HOJ-0001	2	0.000	1250.00
9	INT3d	64964	HOJexny	ESO-HOJ-0001	2	0.000	1250.00
10	INT3	65625	HOJcen	ESO-HOJ-0001	2	0.000	1250.00
11	INT3e	66286	HOJexpy	ESO-HOJ-0001	2	0.000	1250.00
12	INT3f	66506	HOJoapy	ESO-HOJ-0001	2	0.000	1250.00
13	INT3g	70100	HOJcopy	ESO-HOJ-0001	2	0.000	1250.00
14	INT3h	72920	HOJeypy	ESO-HOJ-0001	2	0.000	1250.00
15	SLOT4	78750	Ks	ESO-Ks-0001	3	0.800	2150.00
16	INT4	91875	blankC	blankC	1	0.000	0.00
17	SLOT5	105000	H	ESO-H-0001	3	-0.300	1250.00
18	INT5	118125	blankD	blankD	1	0.000	0.00
19	SLOT6	131250	DARK2	DARK2	1	0.000	0.00
20	INT6	144375	blankE	blankE	1	0.000	0.00
21	SLOT7	157500	Y	ESO-Y-0001	2	-1.000	1650.00
22	INT7	170625	blankF	blankF	1	0.000	0.00
23	SLOT8	183750	DARK1	DARK1	1	0.500	1020.00
24	INT8	196875	blankG	blankG	1	0.000	0.00

Figure 25 VIRCAM filter wheel configuration display panel

The command

```
% vcinsFilterConfig &
```

brings up the filter wheel configuration display panel shown in Figure 25 above. This displays the location, name, ID and properties of all the filters currently installed in the instrument. Filters installed at a "SLOTn" position are the available science filters. The other filters installed at "INTn" positions are intermediate filters, which may have more than one position in the beam. The command

```
% vcinsThermalConfig &
```

brings up a panel which displays all the thermal control settings, as shown in Figure 26 below.

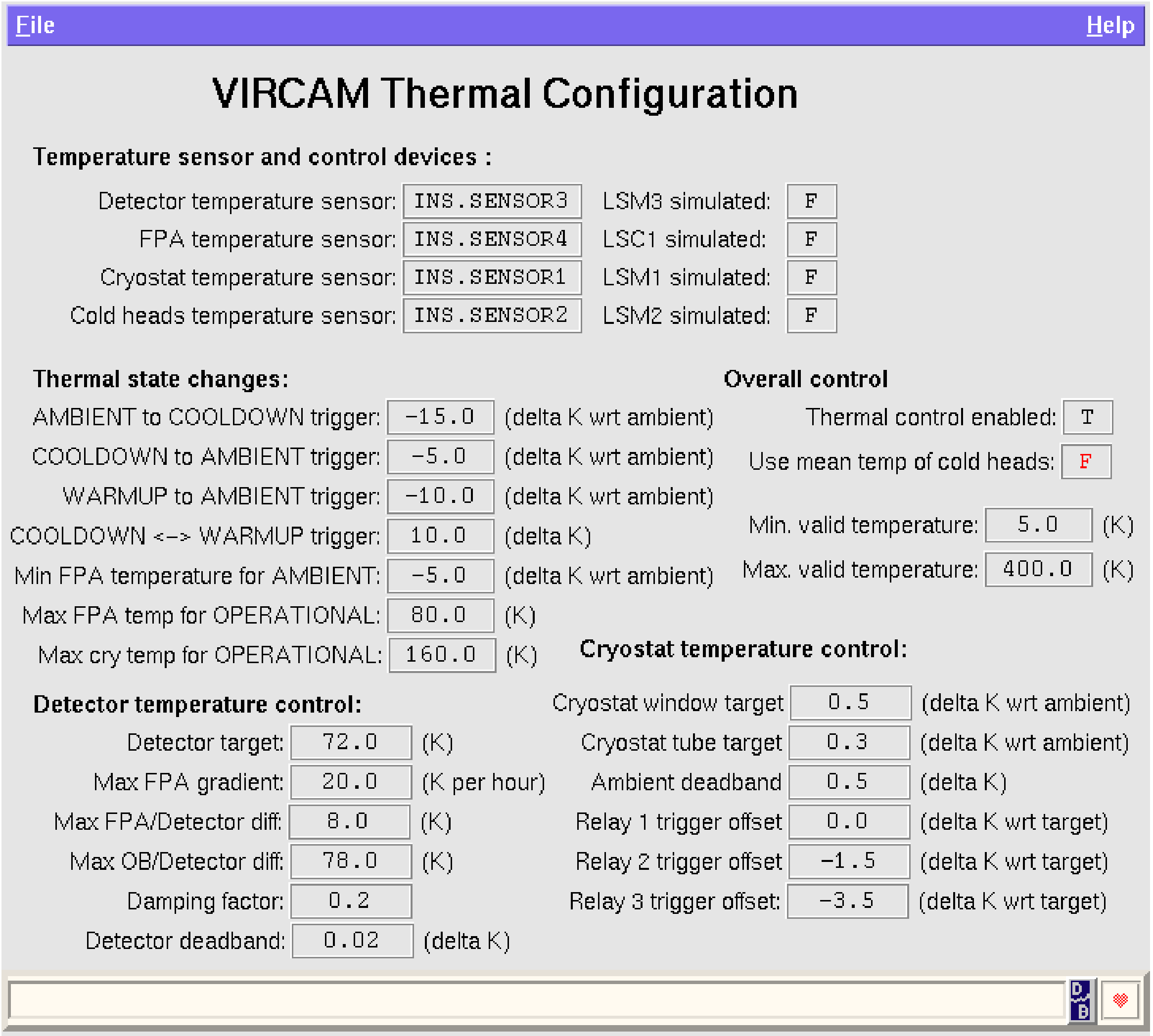


Figure 26 VIRCAM thermal configuration display panel

The command

```
% vcinsWcsConfig &
```

may be used to bring up a "World Coordinates" configuration display panel (not shown here).

4.6 *Beginning Operations*

When the VIRCAM software has started the top level Observation Software control panel should appear, as shown in Figure 27. Selecting "Help→Display man page" will give a detailed description of this panel.

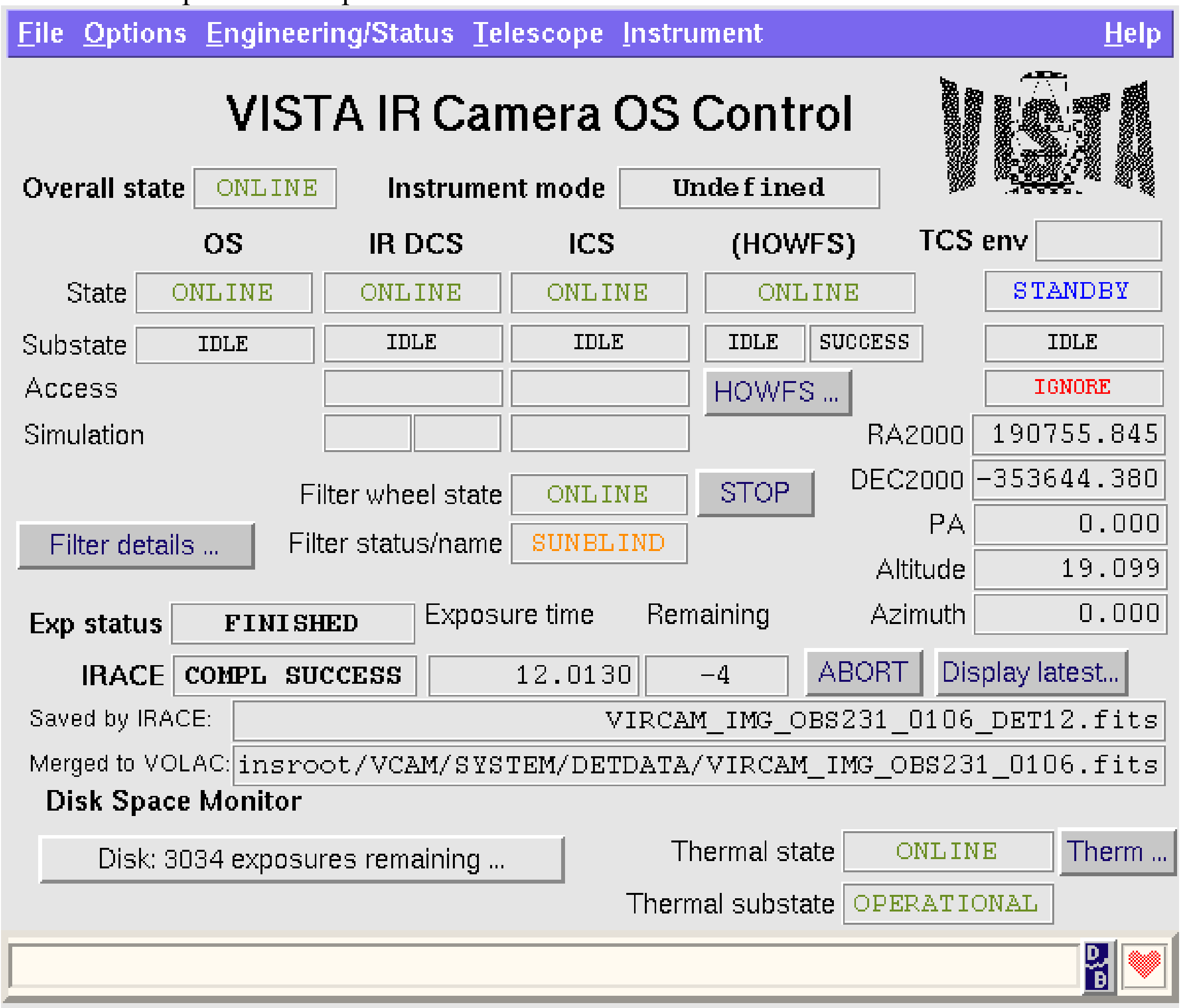


Figure 27 VIRCAM Observation Software Control Panel

Before operations can begin, the instrument must be in the ONLINE state. If it is not ONLINE, the instrument can be moved to the ONLINE state by selecting "Instrument → ONLINE" from the menu. The ONLINE request will only succeed if the VISTA TCS is also ONLINE (unless access to the VISTA TCS is disabled by means of the "Telescope → DISABLE" menu option). The ICS subsystem could take up to 3 minutes to switch to the ONLINE state if it needs to initialise the filter wheel by searching for its home switch.

Additional status panels may be displayed by pressing the button ending in "…" next to their summary. For example, the "Filter details…" button next to the filter summary brings up the filter wheel status panel shown in Figure 28 below, and the "Therm…" button next to the thermal state summary brings up the panel shown in Figure 37 on page 85.

Pressing the "Display latest…" button displays the latest complete data set, as named in the "Merged to VOLAC" field, with the science operations real-time display (section 4.7.2).

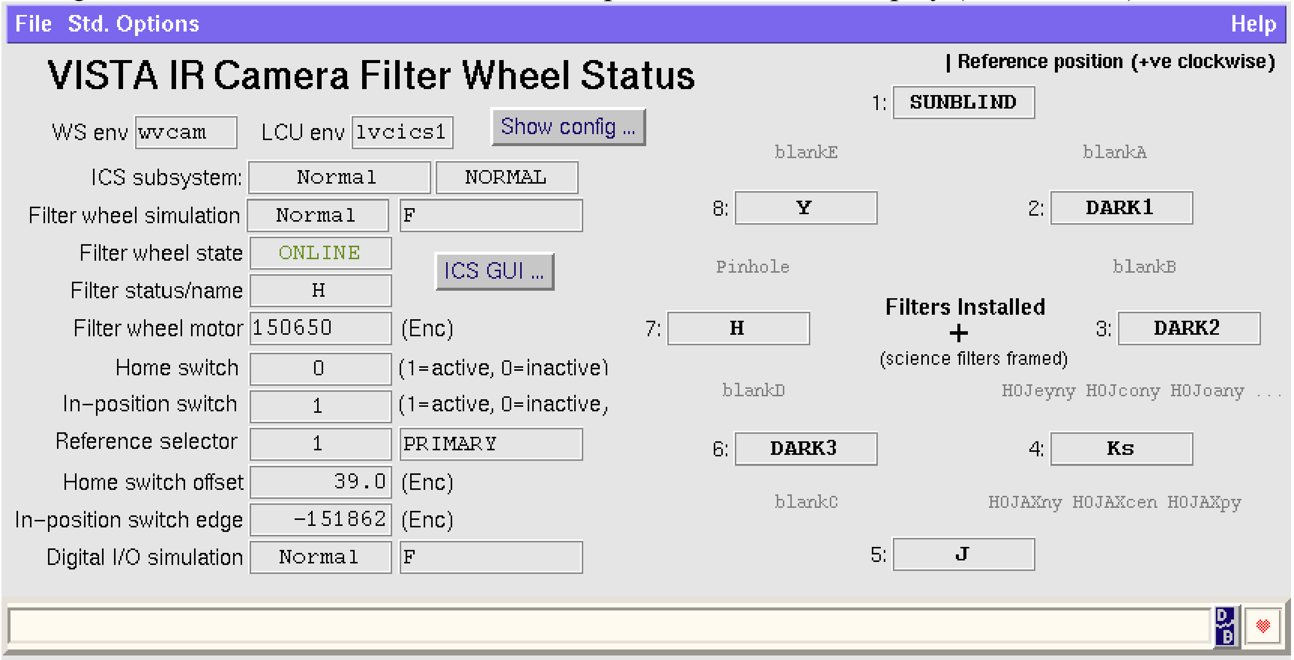


Figure 28 VIRCAM Filter wheel status panel

The filter wheel status panel shows, on the right hand side, the relative location of all the filters installed in the wheel.

Help may be obtained from these panels by accessing the "Help" menu on the right hand side. Several of the panels can display pictures illustrating the VIRCAM instrument, for example try the "Help → Templates" from the Observation Software control panel.

4.7 Real-time Data Display

The VIRCAM software will tend to be operated from a double-headed workstation with two display screens. A typical layout for the VIRCAM control screens is shown in Figure 29 below. The "bob" panel and instrument control panels are displayed on the left hand screen and the real-time data displays are displayed on the right hand screen. To generate this layout it is necessary to use a terminal window displayed on the right hand screen to launch the real-time displays.

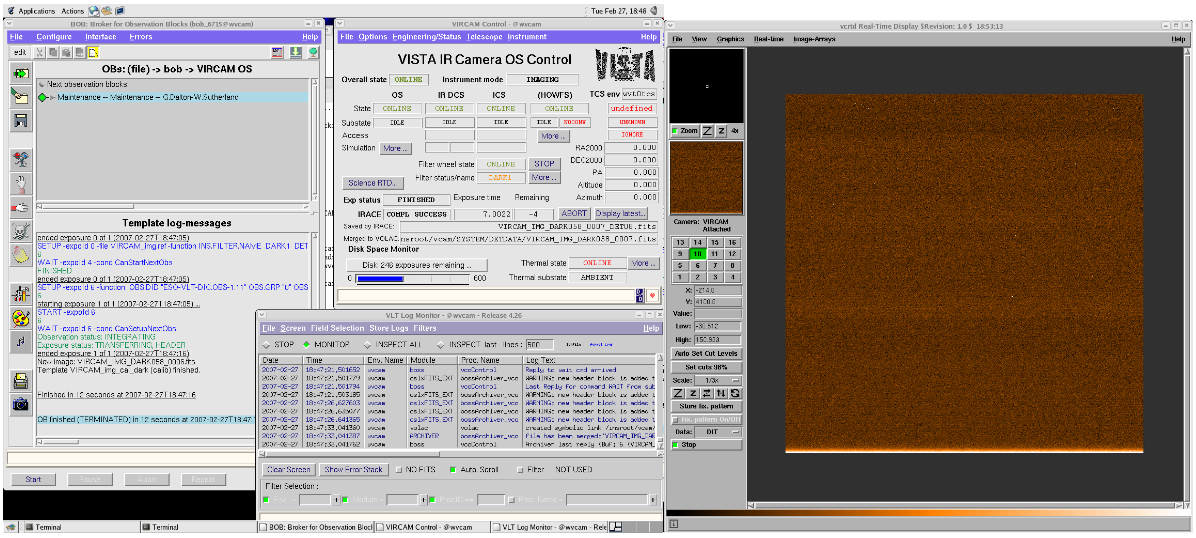


Figure 29 Typical VIRCAM Control Screen Layout

The VIRCAM software comes with two different kinds of real-time data display, as described in sections 4.7.1 and 4.7.2 below.

4.7.1 Engineering Real-Time Data Display

The engineering real-time data display can show all of the data being received by the IRACE data transfer task. The display may be started with the command:

```
% vcinsStart -panel IRTD_ALL
```

which results in the display shown in Figure 30 below. The display shown here consists of four separate windows, each displaying the data received on one of the four separate IRACE data channels. The channels can also be displayed individually with the commands:

```
% vcinsStart -panel IRTD_1
% vcinsStart -panel IRTD_2
% vcinsStart -panel IRTD_3
% vcinsStart -panel IRTD_4
```

This engineering display has the advantage of showing the data from all 16 detectors. However, a big disadvantage is that attaching monitoring processes to the IRACE data channels will slow them down, and the displays themselves will use up a substantial amount of system resources. *Do not use the engineering real-time display during observations where performance and data throughput are important.*

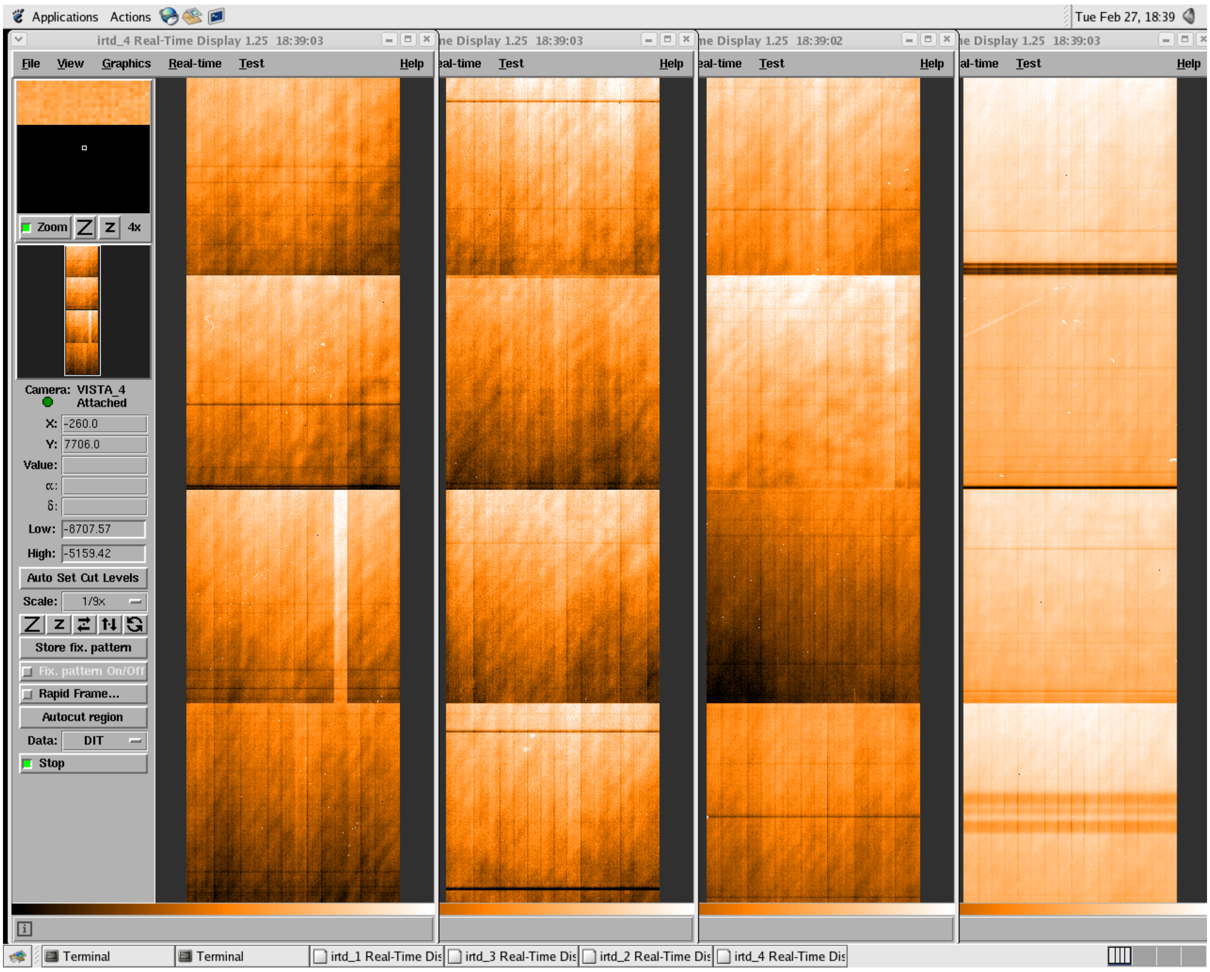


Figure 30 Engineering Real-Time Display for VIRCAM

4.7.2 Science Operations Real-Time Data Display

The science operations real-time data display is an alternative to the engineering real-time display which minimises the amount of system resources used. The display may be started up with either the command:

```
% vcinsStart -panel SCI_RTD
```

or just simply:

```
% vcrtd &
```

which results in the display shown in Figure 31 below. The operational real-time display saves resources by only displaying data from one detector at a time. The display can be changed to any detector by clicking on the detector selector widget on the left side of the panel.

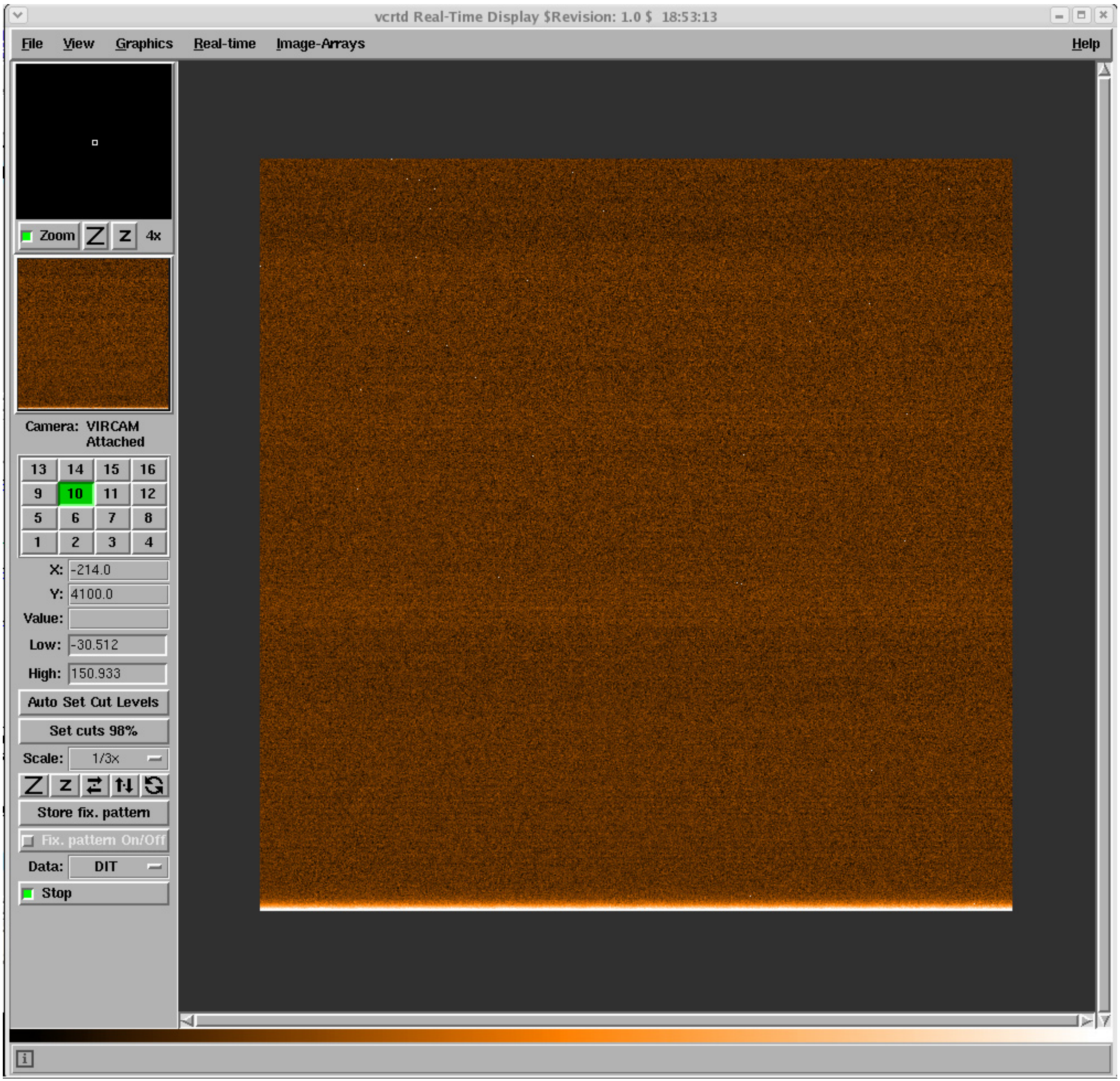


Figure 31 Operational Real-Time Display for VIRCAM

Science data or HOWFS data FITS files may be displayed on demand using the commands:

```
% vcrtd -file <filename> &
% vcrtd -howfs -file <filename> &
```

which results in a display rather like the one shown in Figure 32 below (which assumes the “Display as one image” button has been selected on the data display). All 16 detectors are shown, and the detectors are spaced out according to the World Coordinates contained in the data header. The latest observation may be displayed in this way by clicking the “Display latest…” button on the OS control screen after an observation has completed.

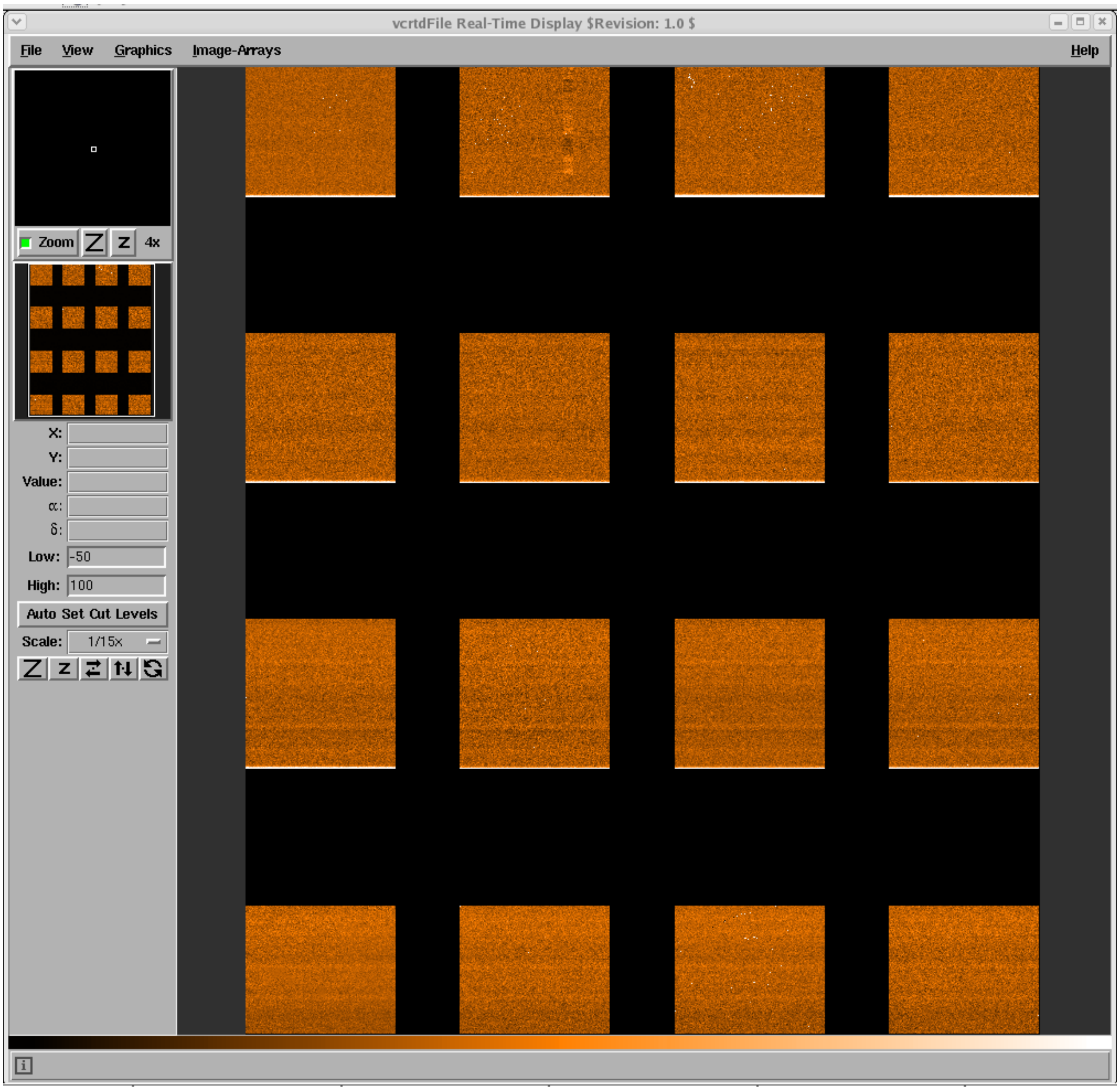


Figure 32 Displaying the contents of a VIRCAM FITS data file

NOTE: Only one copy of the vcrtd display may be running at any one time on a particular workstation (a quirk of the rtdb tool underlying vcrtd). If you wish to have many data displays open at a time, use plain rtd instead with the command:

```
% vcrtd &
```

4.8 Ending Operations

If the instrument is to be left idle and unattended for long periods the SUNBLIND filter must be selected, to protect the detectors from ambient light (the instrument has no shutter). The normal way to leave the VIRCAM software for long periods is to select

"Instrument → PARK" from the OS control panel menu (Figure 27). This will select the SUNBLIND filter and put the IRACE into a safe STANDBY state.

If no active thermal control is required during the idle period, the software should then be switched to the STANDBY state by selecting "Instrument → STANDBY" from the OS control panel menu (Figure 27). Filter wheel movements and active thermal control are both disabled in the STANDBY state. Disabling active thermal control means the high level software no longer sends new temperature targets to the LCU. The LCU will keep monitoring and maintaining the instrument temperature, but the temperature targets will be locked at the values they had when the instrument was last ONLINE.

NOTE: Switching to the STANDBY state is not recommended when the instrument is cooling down or warming up (as might happen during engineering or after a cooling system failure), since active thermal control is needed to manage the cool-down or warm-up of the detectors. See section 6.3.4 on page 127 for a description of the cool-down or warm-up procedure.

4.9 System Shutdown

The VIRCAM software may be stopped and shut down with the command.

```
% vcinsStop
```

This command will stop all running processes but it will not stop the software environments, so the LCU will continue to monitor temperatures. High level thermal control will stop, but the Lakeshore devices will continue to maintain temperatures at their last settings.

4.10 Partial startup/shutdown options

In general, the commands

```
% vcinsStart -proc XXX -panel YYY
% vcinsStop -proc XXX -panel YYY
```

may be used to start and stop any processes (XXX) or panels (YYY). A list of all known processes or panels may be obtained by specifying a process or panel known not to exist, e.g.

```
% vcinsStart -proc JUNK
% vcinsStart -panel JUNK
```

NOTE: A quirk of the vcinsStart command is that when a particular panel is already running somewhere else, the old panel will be shut down before the new one is started. So if a BOB panel is already running, the command:

```
% vcinsStart -panel BOB
```

will stop the existing BOB panel before starting a new one. However, starting BOB directly with the command

```
% bob &
```

will not stop any existing panels. Normally this behaviour is sensible, since it is undesirable to have multiple panels controlling the software. But if you really want multiple panels running at once, it is better to start the new panel using its direct startup command, as shown in the following table.

Panel	**"vcinsStart –panel" name**	**Direct startup command**	**Comments**
BOB	BOB	bob &	Direct start is often easier.
OS control	OS_CONTROL	vcopanControl &	
OS status	OS_STATUS	vcopanStatus &	
ICS control	ICS	vcipanControl &	Direct start not recommended. vcinsStart also ensures the correct processes are running.
DCS control	IRDCS	iracqCtrl &	Direct start not recommended. vcinsStart also ensures the correct processes are running.
Alarm manager	ALARM	alrmDisplay &	Direct start is often easier.
Log monitor	LOG	logMonitor &	Direct start is often easier.
OS engineering	OS_ENGINEERING	vcopanEngineering &	
Thermal control	THERMAL	vcipanThermalControl &	
HOWFS control	HOWFS	vchpanControl &	
Engineering real-time display	IRTD_ALL	vcdRtd &	
Science real-time display	SCI_RTD	vcrtd &	Direct start is often easier, but do not create more than one panel.

4.11 Observations With Templates

The VIRCAM instrument is operated by means of templates which are invoked from Observation Blocks in the same way as any other ESO/VLT instrument, [RD49]. Each Observation Block consists of an acquisition templet followed by one or more observation templates. When the instrument software is operated stand-alone, Observation Blocks may be loaded manually into BOB and executed using the procedure described in the *"Template Instrument User and Maintenance Manual",* [RD44], and "*BOB User Manual*", [RD45]. An important template for testing and verification purposes is "VIRCAM_gen_tec_SelfTest", which self-tests the instrument by executing every possible template.

VISTA is a survey telescope and VIRCAM is designed to accept Observation Blocks prepared in advance and queued by the VLT scheduling system. An observer can use one of two methods for preparing VIRCAM observations (see section 2.2.2 on page 41):

1. Observation Blocks may prepared in advance (from any VIRCAM templates) using P2PP on a separate workstation, just as any standard ESO/VLT instrument, [RD25].

2. A sky survey consisting of large numbers of tile observations (section 2.1.4 on page 32) may be prepared in advance automatically using the VISTA Survey Area Definition Tool (see [RD14]). The tool generates a file which is ingested by P2PP and used to generate the Observation Blocks. This option uses the tile template, "VIRCAM_img_obs_tile", which is the normal template used for science observations.

The VIRCAM templates are described in more detail in section 5.18 on page 113 and section 11.7 on page 204.

4.12 Wavefront Sensing

As mentioned in section 2.1.5, the VISTA IR camera contains wavefront sensors which communicate wavefront information to the VISTA TCS. The Low Order Wavefront Sensors (LOWFS) are controlled directly by the VISTA TCS, and their operation is described in [RD12]. The High Order Wavefront Sensor (HOWFS) software is invoked whenever the "VIRCAM_howfs_obs_exp" or "VIRCAM_howfs_obs_wfront" templates are executed[14] (see section 5.18.1 on page 115). These templates position the filter wheel to one of the HOWFS beam splitter positions (see Figure 9 on page 36) and then instruct the IRACE DCS to make an exposure windowed around the location of the pre-focal and post-focal images. An exposure is made and the data saved to a FITS file. This FITS file is then passed on to the HOWFS image analysis server, which processes the images and generates wavefront coefficients. In the "VIRCAM_howfs_obs_wfront" template, the wavefront coefficients are

[14] NOTE: These two templates will not execute properly when IRACE is operating in simulation mode. This is because the HOWFS image analysis software will reject the simulated IRACE data. It is possible to test the HOWFS templates in simulation mode using the "VIRCAM_howfs_tec_test" maintenance template, which substitutes a test file for the IRACE data (see section 8.10 on page 159).

then transmitted to the TCS to adjust the open-loop lookup tables used by the active optics software within the TCS. The complete process is summarised in the flow chart shown in Figure 33 below. Important things to know are:

- The primary purpose of the HOWFS is to make high order corrections to the open-loop lookup tables. These are only needed occasionally. Continuous, closed-loop adjustments to M2 are controlled by the LOWFS software.
- The HOWFS image analysis software does its own dark-subtraction and flat-fielding. Before analysing any images, calibration observations need to have been made using the same HOWFS filter and exposure time. The "VIRCAM_howfs_cal_dark" and "VIRCAM_howfs_cal_domeflat" templates may be used (during daytime calibration) to do this.
- HOWFS calibration data files are stored indefinitely in the $INS_ROOT/SYSTEM/HOWFSDATA directory. On-sky HOWFS exposures are stored in the same $INS_ROOT/SYSTEM/DETDATA directory as the science data, but soft links to the HOWFS data are written to HOWFSDATA. The HOWFSDATA directory should be tidied up manually at regular intervals.

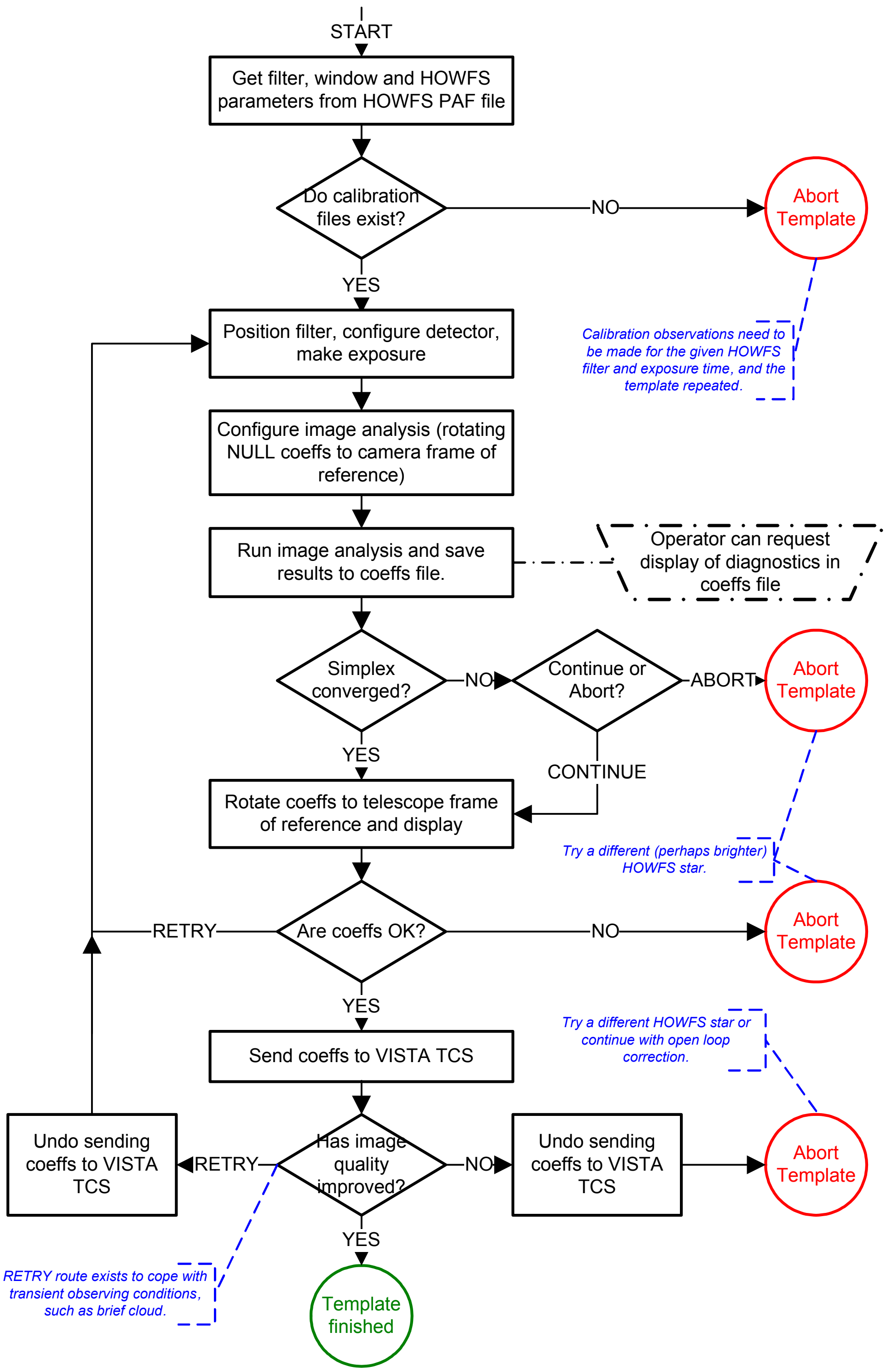


Figure 33 Flow Chart for the VIRCAM_howfs_obs_wfront template

- Each HOWFS position has a set of NULL coefficients giving the wavefront aberration expected at that off-axis position. These coefficients are rotated from the camera focal plane frame of reference to the camera detectors frame of reference before being used by the image analysis software. The NULL coefficients are subtracted from the fitted coefficients to give the corrections that need to be applied to the TCS lookup tables.

- Fitted coefficients are rotated from the camera detectoras frame of reference to the telescope M1 frame of reference (taking into account the Cassegrain rotator angle) before being transmitted to the VISTA TCS.
- When the NULL coefficients are subtracted, fitted coefficients will be zero when no adjustments are needed.

The progress of the image analysis software may be monitored from the HOWFS panel (Figure 34, below), which may be invoked from the "HOWFS…" button next to the HOWFS summary on the OS control panel (Figure 27 on page 65) or directly by the command

```
% vcinsStart -panel HOWFS
```

The panel displays the current HOWFS SETUP parameters at the top. The image data being analysed, the optional bad pixel mask, the dark frame or the flat-field frame may be displayed by pressing the "I…", "M…", "D…" or "F…" buttons respectively.

A summary of the progress of the simplex algorithm used for fitting the wavefront, [RD10], is available at the bottom left of the panel (press the "Show details…" button for more information). The image analysis algorithm consists of two loops:

- The inner loop executes the simplex algorithm and terminates when the fitting has converged, when the relative tolerance goes below the maximum limit shown.
- The outer loop, which makes at least two iterations, expands and repeats the simplex algorithm from its last position, and terminates when the fit converges on the same solution as the previous iteration, when the largest change in any coefficient is less than the maximum shown. The outer loop also terminates when the maximum repeat count is reached.

If the simplex algorithm exceeds the maximum iteration count or function evaluation count, the analysis is terminated with a "failed to converge" error. In principle, fitted wavefront coefficients are available when the inner loop has terminated. The outer loop exists to move the simplex algorithm away from false minima and encourage it to find the real best fit, (see [RD10] and the documents referenced within for details).

The simplex status will be "WORKING" when the algorithm is analysing a new set of data and will change to "SUCCESS" when it has successfully analysed the data. The wavefront coefficients (in the camera frame of reference) are displayed in mirror modes at the bottom right of the panel (pressing the "Zernikes…" button displays the same coefficients in terms of Zernikes). If the Simplex algorithm finishes with a "FAILED" status it has failed to converge. The ABORT and STOP buttons may be used if the Simplex algorithm is converging too slowly (although it is normal for the algorithm to take several minutes). See the documents "VISTA Wavefront Sensing Overview", [RD9], and "High Order Wavefront Sensor Software Design Description", [RD7], for more information.

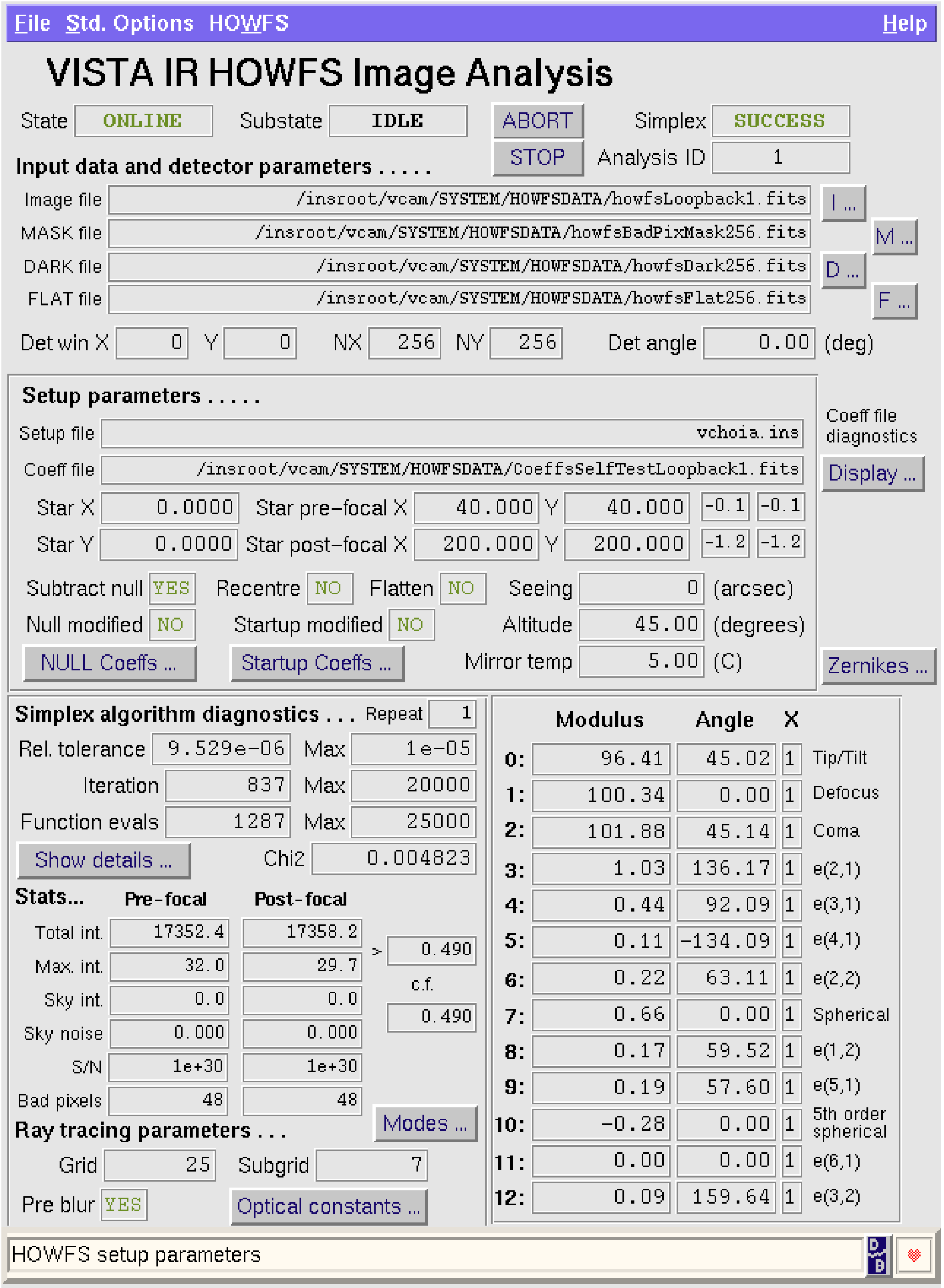


Figure 34 VIRCAM HOWFS Image Analysis Panel

The diagnostic images contained in the coefficients file can be displayed by pressing the "Display…" button next to the coefficients file name. The images can be viewed all at once, in the layout shown in Figure 35 below, by selecting "Display as one image" in the real time display utility. The six images shown are, from top to bottom:

- The original pre-focal and post-focal images, as extracted and windowed by the software.
- The theoretical pre-focal and post-focal images created from the best-fitting wavefront coefficients.
- The difference between the original pre-focal and post-focal images and the theoretical one.

If there has been a good fit, the original and theoretical images should look as similar as possible, and the difference images should contain mostly noise and unfittable artefacts such as the spider supporting M2.

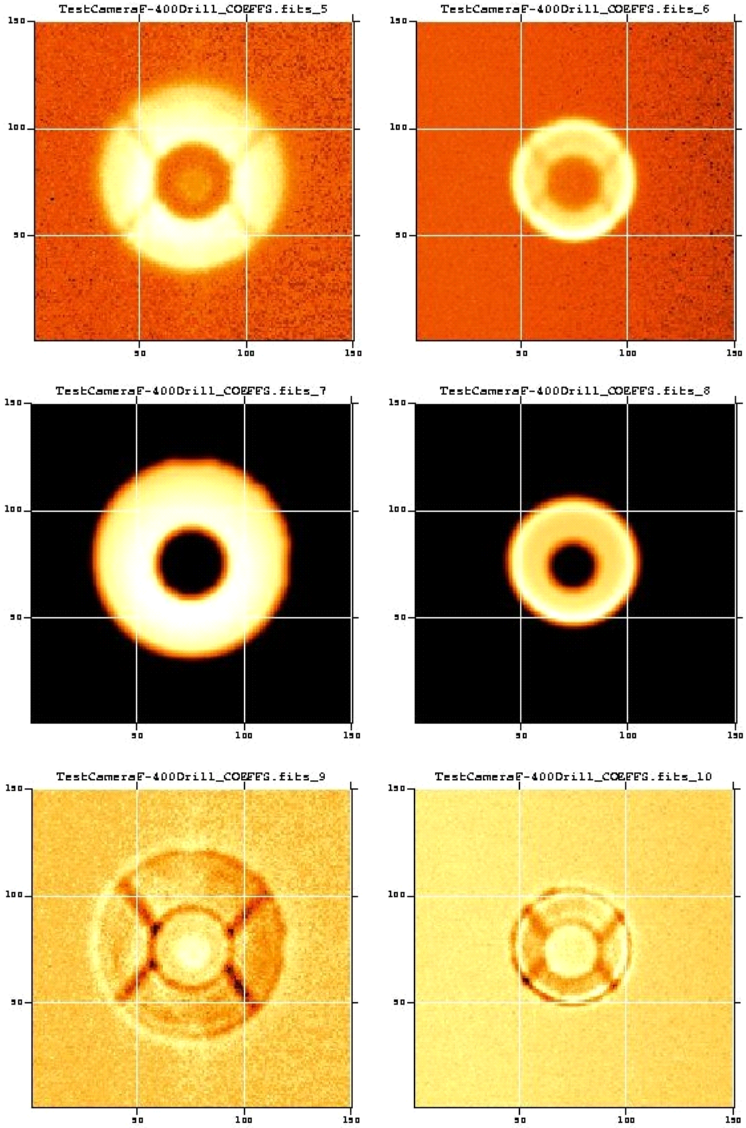


Figure 35 Typical diagnostic Images contained within the HOWFS coefficients file

4.13 Alarms

The VIRCAM instrument can report the following alarms. If it is not started automatically, the alarm manager panel may be brought up with the command

```
% vcinsStart –panel ALARM
```

A description of the problem, and the recommended corrective action may be obtained from the alarm manager panel by selecting "ACTIONS → Display help" from the menu. The operator actions are also summarised here.

Description	**Severity**	**Operator's action**
WFS thermal plate temperature reading out of range	Warning Serious	Check Lakeshore device LSM2 and report the problem. Operations may continue.
FPA thermal plate temperature reading out of range	Warning Serious	Check Lakeshore device LSM3 and report the problem. Watch the detector temperatures carefully and continue operations.
Detector XX temperature reading out of range	Warning	Check Lakeshore device LSM3. Check mean detector temperature. Data quality may suffer if the detectors are not within 0.03K of their ideal temperature.
	Serious	Abandon operations and call an engineer. Consider detector thermal protection procedure (see below) if detectors are too cold.
WFS XX temperature reading out of range	Warning	Check the TCCD temperature control system. WFS detectors are not as sensitive to temperature as the science detectors, so operations may continue.
	Serious	Call an engineer.
Cryo-cooler N Nth stage temp reading out of range	Warning Serious	Check Lakeshore device LSM2. This alarm is normal if the cryostat is being warmed up, cooled down, or is at ambient temperature. Operations can continue if two coolers are still working and internal temperatures are ok. Abort operations, and start detector thermal protection procedure (below) if several internal temperature alarms are activated or if all 3 coolers have failed. One or more coolers warming up unexpectedly can be a symptom of a compressor fault or a blockage in the helium circuit. Call an engineer.
Filter wheel hub temperature reading out of range	Warning	Check Lakeshore device LSM1. Check that the cryo-coolers are operating properly. Out of range readings are ok if the cryostat is being warmed up, cooled down, or is at ambient temperature. If several cryostat temperatures warm up report the
Filter wheel shield temperature reading out of	Warning	

range		problem and start detector thermal protection procedure (see below).
Lens temperature reading out of range	Warning	
Baffle temperature reading out of range	Warning	
Optical bench top temperature reading out of range	Warning	Check Lakeshore device LSM1. This alarm is ok if the Liquid Nitrogen tank is being flushed, or if the cryostat is at ambient temperature. If cooling down with Liquid Nitrogen, have you run out? This alarm is not serious unless accompanied by other cryostat temperature alarms (see above). An unusually warm OB temperature may prevent the detectors from reaching their target temperature.
Cryostat tube temperature reading out of range	Warning	Check Lakeshore device LSM1. Check the cryostat and window for condensation or icing. Consider closing the dome if the ambient temperature is close to dew point. Check the thermal control software and consider switching cryostat external heaters on manually (see below).
Window temperature reading out of range	Warning	
Ambient temperature reading out of range	Warning	
Detector protection heater on	Serious	Check digital I/O device DIS1. The detector protection heater has started warming the detectors. Report the problem. Abandon operations when detectors get too warm.
Cooling failure in cabinet N	Fatal	Check digital I/O device DIS1. **This alarm is FATAL** – some power will already have failed. Call an engineer. Shut down power to cabinets.
Thermal protection DC power failure	Serious	Check digital I/O device DIS1. Report the problem to an engineer. The alarm is not fatal (operations can continue) but problem must be corrected as soon as possible.
Mains power failure (UPS activated)	Serious	Check digital I/O device DIS1. *** **Stop any observations and select the SUNBLIND filter** to protect the detectors against ambient light**.** Save any information that would be lost during a power failure. Call an engineer. Operations may continue if power is restored before UPS runs out. If UPS power runs out, refer to "Recovering from a system reboot or power failure" in section 9.1 on page 161.
Vacuum gauge N reading too high	Warning	Check Pfeiffer device VAC1. Call an engineer. Switch on the cryo-pumps (manually) until the vacuum is restored. Check the cryostat for leaks at

		the next available opportunity.
	Serious	Call an engineer. A quick recovery is no longer possible. The cryostat must be attached to the service trolley and the vacuum re-established using roughing pumps. Check the cryostat for leaks as soon as possible.

4.13.1 Emergency procedures

See section 9 on page 161 for recovery from common failures.

4.13.1.1 Detector thermal protection procedure

The science detectors need to be protected in two ways:

1. Any change to their temperature must be limited to a certain maximum gradient, to protect them from thermal stress.
2. If the cryostat begins to warm up, the detectors need to be warmed so they are not the coldest objects in the cryostat, to prevent any out-gassed contaminants from condensing on the detectors.

During normal operation there is no problem. The detectors are maintained at their ideal temperature by gentle heating of the FPA thermal plate. However, if the cryostat starts warming because of a power failure or cooling system failure, the detectors need to be warmed. The thermal control software should do this automatically — when it detects an increase in cryocooler temperature the thermal state will change from OPERATIONAL to WARMUP. If the operator knows the cryostat is definitely warming up, or the detectors are definitely too cold, this procedure can be triggered early manually from the thermal status panel (vcipanThermalControl, see Figure 37) by selecting "ENGINEERING → WARMUP" from the menu.

Switching the coolers off when there is a large temperature difference between the detectors and the optical bench top can also cause thermal stress on the detectors. The software protects the detectors from this event by warming them automatically if the optical bench top gets too warm. This is a rare event which can happen only during an unusual cooling system failure.

In the event of a power failure or software failure, the VIRCAM LCU cabinet is equipped with some "detector thermal protection" electronics. This backup system monitors the power status signals and a heartbeat signal coming from the LCU. If the mains power fails or the heartbeat signal stops, the electronics applies a voltage to a detector heater. (This signal is sensed by the software, and is what causes the "Detector protection heater on" alarm). The backup thermal protection can be activated manually by putting the software into the OFF state (which stops the heartbeat signal). This is a useful way of protecting the detectors in the event of a Lakeshore device failure, but activating this backup protection should not be necessary if the software has successfully changed to the WARMUP state.

4.13.1.2 Cryostat window thermal protection

The cryostat window is protected from icing and condensation by three heaters which should come on progressively when the temperature falls below certain thresholds compared to ambient. Extra heating will be applied in COOLDOWN or WARMUP mode. If for any reason this protection software does not apply sufficient heating, the heaters may be turned on manually by pressing the "1" button against heater 3 on the thermal status panel (Figure 37 on page 85). Heaters 1 and 2 are triggered directly from the LCU and heater 4 is a spare for future expansion, and may not be wired up. If the software keeps switching heater 3 off and you want it left on permanently, you can take full manual control by shutting down the ICS software, disabling thermal control and then restarting the software, with these commands:

```
% msgSend "" vciControl STANDBY ""
% vcinsStop -proc ICS
% ctooConfigSet VIRCAM INS.THERMAL.ENABLE F
% vcinsStart -proc ICS
% msgSend "" vciControl ONLINE ""
```

Be aware that disabling thermal control will also disable the detector temperature control and leave the focal plane array set point constant at its last defined value.

4.13.1.3 Detector ambient light protection

Since the instrument has no shutter, the detectors need to be protected against overexposure to ambient light, as this can produce persistence effects which only decay on long timesales.

The detectors should be protected from ambient light whenever the instrument is idle by selecting the SUNBLIND filter. When VISTA is parked (by executing the PARK template or selecting "`Instrument→PARK`" from the OS control panel) the SUNBLIND filter is automatically selected. If any alarms indicate an imminent power failure or telescope control problem, the SUNBLIND filter must be selected immediately.

The filter wheel control software can protect the detectors by minimising the number of bright filters passed through the beam when the wheel is being rotated.

To protect the detectors from overexposure, the instrument software automatically sets the exposure time to a minimum whenever the instrument is not carrying out any template.

4.14 Data Files Location

All the data files generated by the VISTA IR Camera are stored in the $INS_ROOT directory in the following locations:

- Configuration files:
  $INS_ROOT/SYSTEM/COMMON/CONFIGFILES/

- IRACE configuration files
  $INS_ROOT/MISC/IRACE/<type>/
  where <type> is one of CLK, DSUP, SEQ, SYS, VOLTAGE, DET, MSUP, SSD and TEST (see the IRACE user manual [RD39] for details).

- FITS files containing science data and results of exposures:
  $INS_ROOT/SYSTEM/DETDATA/

- FITS files containing High Order Wavefront Sensor data and calibration files:
  $INS_ROOT/SYSTEM/HOWFSDATA/

- Setup files:
  $INS_ROOT/SYSTEM/COMMON/SETUPFILES/<type>/
  where <type> is one of
  REF — reference setup files (widely used by VIRCAM)
  INS — instrument setup files (used by HOWFS and self-test scripts)
  DET — detector setup files (not used by VIRCAM)
  TARG — target setup files (not used by VIRCAM)

- VISTA parameter files (including HOWFS location descriptions, tile, jitter and microstep setup files and twilight sky database):
  $INS_ROOT/SYSTEM/MISC/VISTA/

- Template signature files:
  $INS_ROOT/SYSTEM/COMMON/TEMPLATES/TSF/

- Observation block description files (starting directory for BOB):
  $INS_ROOT/ SYSTEM/COMMON/TEMPLATES/OSB/

4.15 Engineering

4.15.1 OS engineering panel

The OS engineering panel (see Figure 36 below) may be brought up by selecting "Engineering/Status → OS Eng GUI" from the menu of the OS control panel, or by invoking the panel directly with the command:

```
% vcinsStart -panel OS_ENGINEERING
```

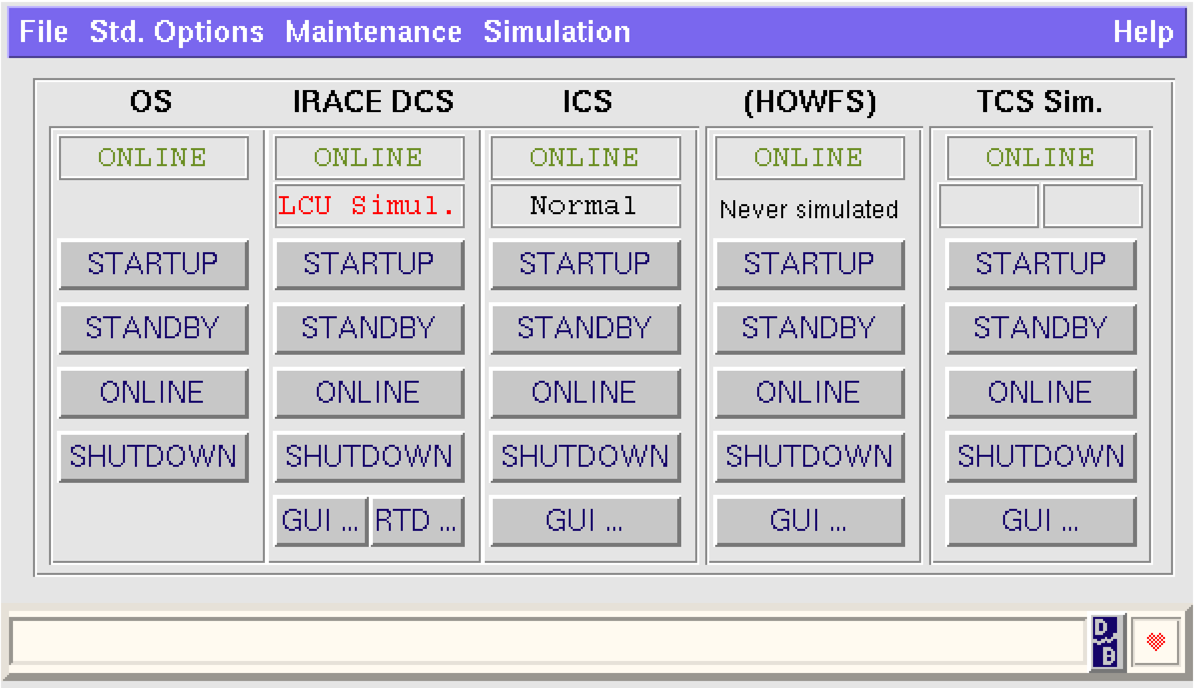


Figure 36 VIRCAM OS Engineering Panel

This panel may be used to adjust the state and bring up the GUI associated with an individual subsystem (e.g. to investigate a problem). The GUI for the IRACE system is described in [RD39], and the VISTA HOWFS GUI has been described in section 4.12. The other subsystem panels are described below. The "RTD" button brings up a real-time display selector panel, which may be used to select one of the real-time displays described in section 4.7.

4.15.2 Thermal Status Panel

The thermal status panel can be started by pressing the "More…" button next to the thermal state displayed on the OS control panel (Figure 27 on page 65), by selecting "Engineering/Status → OS Thermal control GUI" from the menu of the OS control panel, or by invoking the panel directly with the command:

```
% vcinsStart -panel THERMAL
```

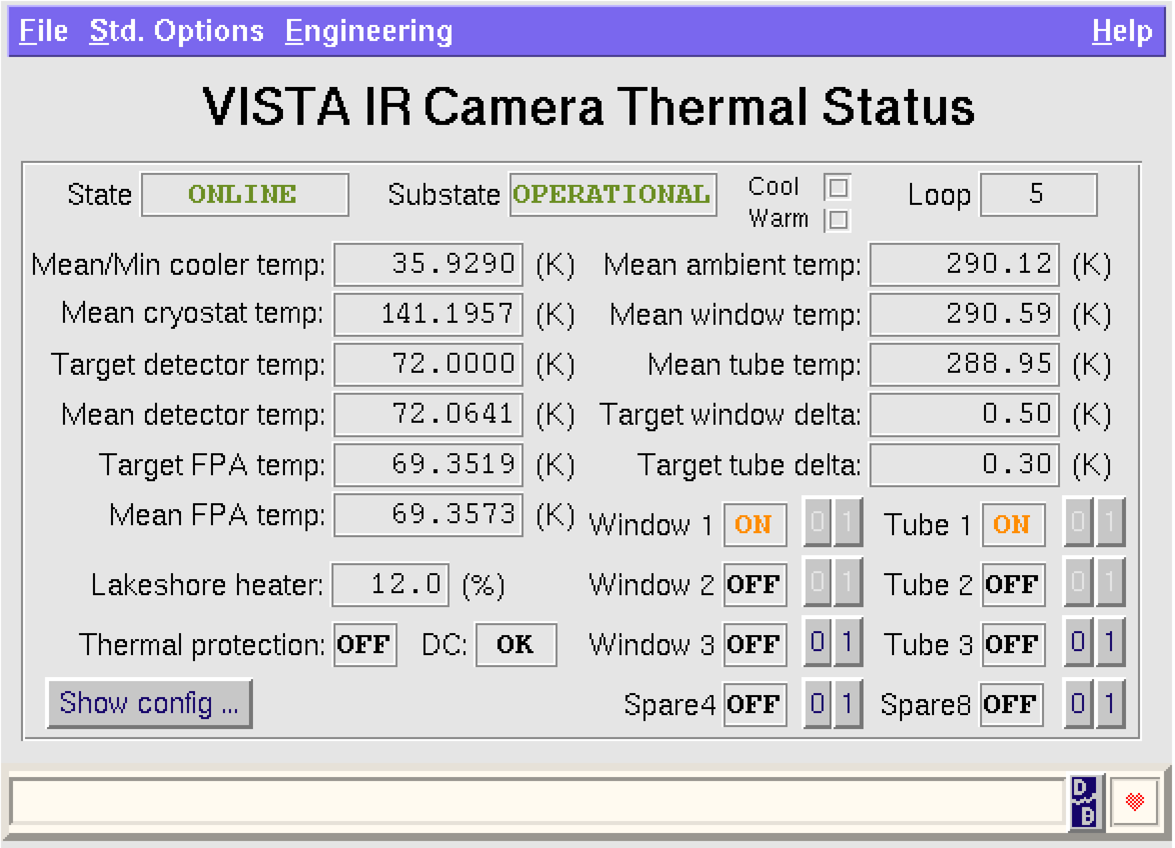


Figure 37 VIRCAM Thermal status panel

The panel shows the key temperatures and targets, and the status of the Lakeshore heater (on the FPA plate) and the cryostat window and tube heater relays. Select "Help→Display man page" for detailed information. The *State* box in the above panel shows the overall state of the thermal control software, which may be:

- **DISABLED**: Thermal control software has been disabled by setting THERMAL.ENABLE=F (see next section).
- **OFF**: Thermal control is OFF because the instrument is not in the *STANDBY* or *ONLINE* state, or because the LCU is offline.
- **ONLINE**: Thermal control software is enabled and *ONLINE*. This is the normal state when the instrument is operational.
- **INVALID**: Thermal control software is enabled and *ONLINE*, but some of the temperature readings are invalid. This typically happens when some of the sensor devices are simulated or are not working, and there is insufficient information for thermal control.

The *Substate* box shows the thermal state of the instrument, as derived from the history of temperature readings, which may be:

- **UNKNOWN**: Instrument thermal state is unknown.

- **AMBIENT**: Instrument is at ambient temperature.
- **COOLDOWN**: Instrument is being cooled down.
- **WARMUP**: Instrument is being warmed up.
- **OPERATIONAL**: Instrument is at operational temperatures. This is the normal state when the instrument is operational.

The "Engineering" menu contains entries that may be used to force a state change: Selecting "Engineering→COOLDOWN" will cause an AMBIENT→COOLDOWN or a WARMUP→COOLDOWN state change. Selecting "Engineering→WARMUP" will cause an OPERATIONAL→WARMUP or a COOLDOWN→WARMUP state change. It is prudent to force an AMBIENT→COOLDOWN transition to preheat the detectors an hour or so before cooling and prudent to force an OPERATIONAL→WARMUP to preheat the detectors an hour or so before a warmup. A COOLDOWN→WARMUP transition can be used to increase the detector heating before transit to the telescope (or to speed up the state transition if the software is restarted during a warmup).

NOTE: It may be several seconds before the software responds to a trigger request. This is normal because the software only checks its status every few seconds.

4.15.3 ICS engineering panel

The ICS engineering panel (see Figure 38 below) may be invoked from the OS panels or be started directly with the command:

```
% vcinsStart -panel ICS
```

Select "Help→Display man page" on the panel for detailed information.

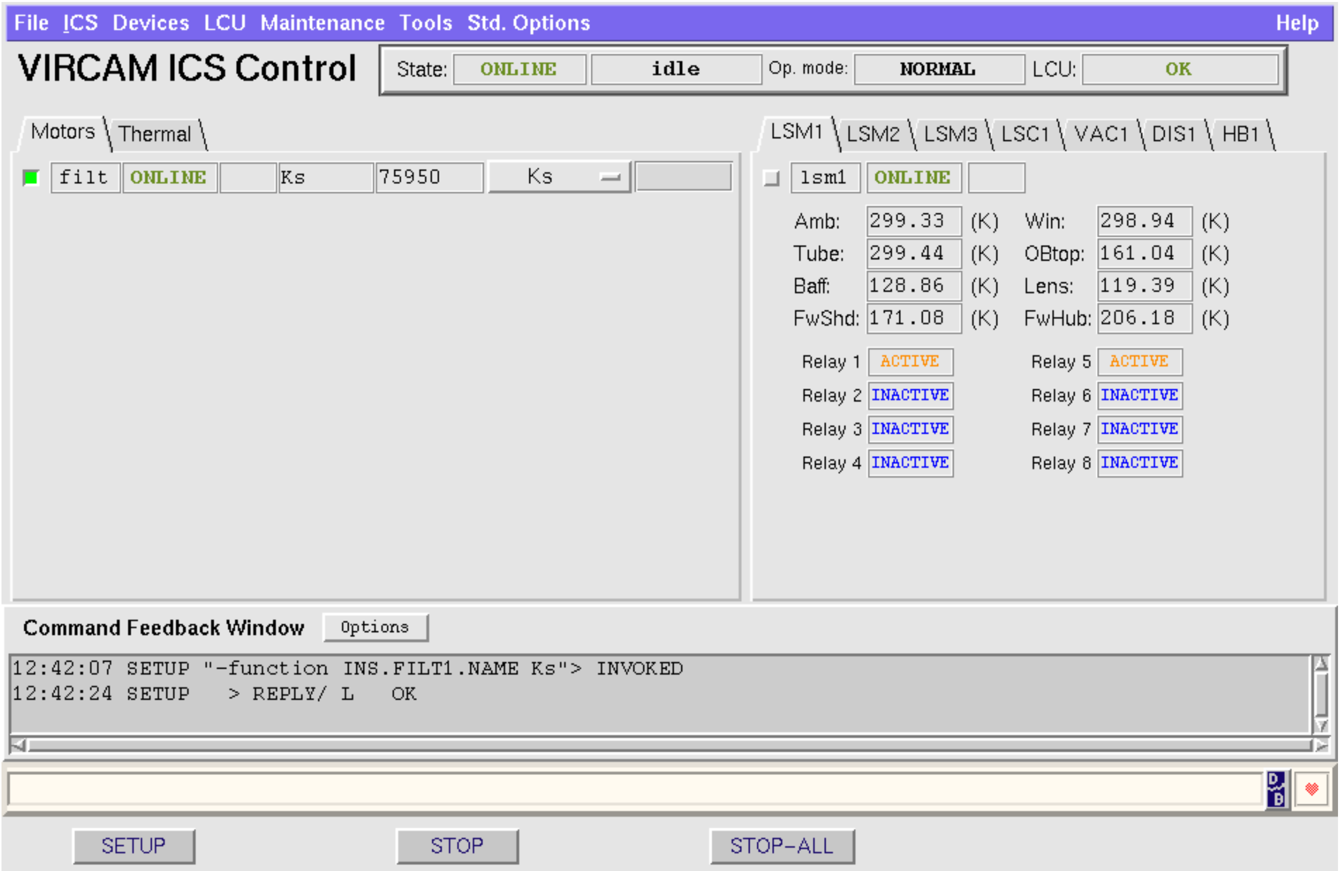


Figure 38 VIRCAM ICS Engineering Panel

This panel allows any filter to be selected or, by selecting "Enc" from the filter menu and entering a number in the right hand box, the filter wheel to be moved to any motor position. Be aware that moving the filter wheel from this panel will bypass the high level software which checks the in-position switches and avoids moving bright filters through the beam. The switches can be monitored manual by selecting the "DIS1" tab on the right hand pane. The ICS engineering panel can also be used to redatum the filter wheel if you suspect it has lost its position — select "ICS → Redatum" from the menu. The "Devices" menu can be used to control the state and simulation mode of an individual device, for example to restart a particular device if there is a problem. See the "ICS Software Design Description", [RD5], for details.

The "Tools" menu contains a collection of utilities for configuring or monitoring the instrument. In particular, "Tools→Plotting" can be used to bring up the ESO/VLT plotting tool, [RD53], and generate a strip chart showing various collections of instrument sensor readings.

4.15.4 VISTA TCS public panel (simulator only)

The VIRCAM software will only go into the ONLINE state if the VISTA TCS is already in the ONLINE state. This should be done from the VISTA TCS panels. However, in situations where the VISTA TCS is being represented by a simulator, the simulator can be controlled from a panel (shown in Figure 39) invoked from the OS control panel. If the TCS simulator is in the OFF state press the COLD-START button to invoke the STANDBY state. When in the STANDBY state, pressing the START button will change the TCS to the ONLINE state. For more details about the VISTA TCS, see the VISTA TCS documentation. As with many of the VIRCAM software panels, selecting "Help → Display man page" on the panel will bring up a man page.

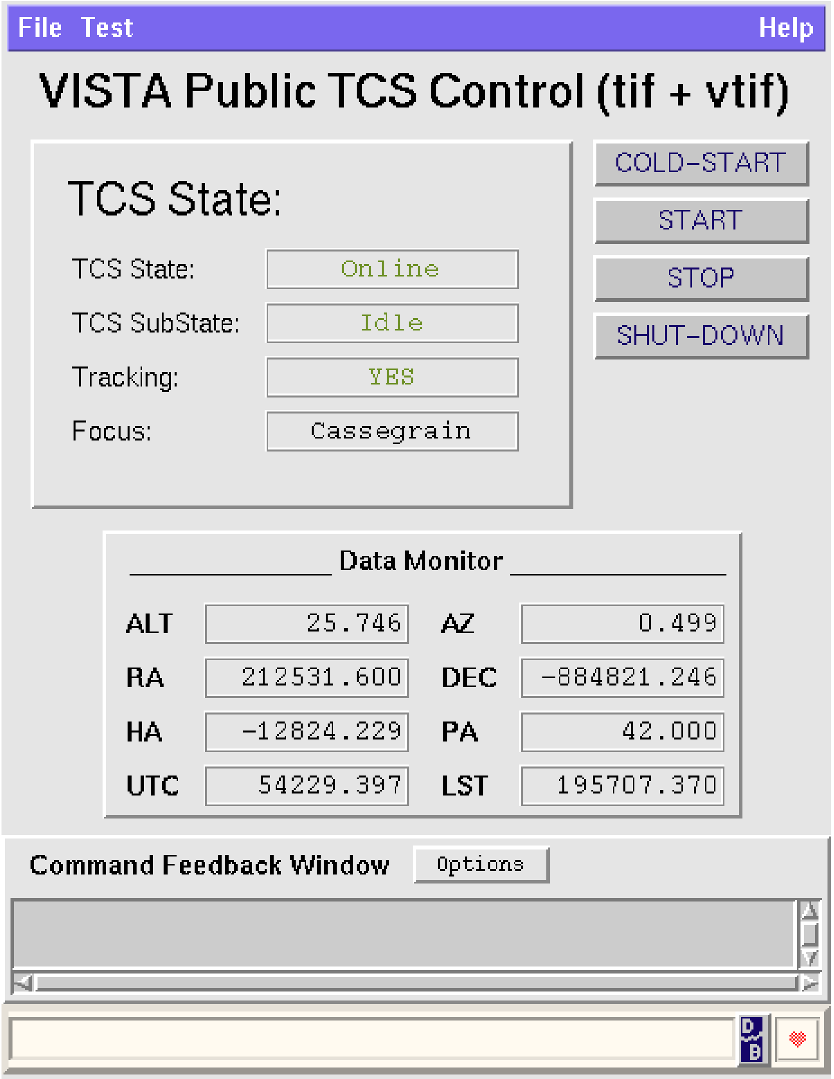


Figure 39 VISTA TCS simulator panel

5 PROGRAMMER'S GUIDE

5.1 *Instrument Modes*

The VISTA IR Camera has the following instrument modes:

- **IMAGING** — An observation is made with the science detectors and the data file is passed to VOLAC. This mode is used for most observations.
- **HOWFS** — An observation is made with the science detectors using a special intermediate filter and a small region of interest. The data file is passed to the HOWFS image analysis subsystem. The HOWFS coefficients are then passed to the TCS.

5.2 *Subsystems*

The following table shows the FITS prefix assigned to the various VISTA IR Camera subsystems:

Subsystem	**FITS Prefix**
OS	OCS
ICS	INS
IRDCS	DET
TCS	TEL
(HOWFS)	INS HOWFS

NOTE: The HOWFS is an on-line data reduction system rather than a subsystem, but it still uses a FITS prefix.

5.3 *ICS Software Devices*

ICS software devices are defined in the configuration file "vcmcfg/config/vcmcfgINS.cfg" and other files whose names are of the form "vcmcfgICS*.cfg" (see section 6.1 for a list of files). These files are installed into "$INS_ROOT/SYSTEM/COMMON/CONFIGFILES/" when the vcmcfg module is built.

The VIRCAM devices are all based on the templates and classes provided by the "*INS Common Software Base ICS*", [RD46]. The devices are defined as follows (see the block diagram in Figure 2 on page 26):

#	**Name**	**Description**	**Positions**	**Motor axis**	**FITS prefix**	**ICB class**
1	filt	Filter wheel	discrete	circular	INS.FILT1	icbMOT_FILTER
2	lsm1	Lakeshore 218	N/A	N/A	INS.SENSOR1 INS.TEMPi	Special device: vcils218 → ic0INS_SENSOR
3	lsm2	Lakeshore 218	N/A	N/A	INS.SENSOR2 INS.TEMPi	Special device: vcils218 → ic0INS_SENSOR

4	lsm3	Lakeshore 218	N/A	N/A	INS.SENSOR3 INS.TEMPi	Special device: vcils218 → ic0INS_SENSOR
5	lsc1	Lakeshore 332	N/A	N/A	INS.SENSOR4 INS.TEMPi	Special device: vcils332 → ic0INS_SENSOR
6	vac1	Pfeiffer TGP256	N/A	N/A	INS.SENSOR5 INS.PRESi	Special device: vcitgp256 → ic0INS_SENSOR
7	dis1	Digital Sensors	N/A	N/A	INS.SENSOR6 INS.SWi INS.SENSi	icbSEN_DIGITAL
8	hb1	Heart beat	N/A	N/A	INS.HB1	Special device: vcihb

All devices are on the same LCU. Some remarks:

- Device filt1 controls the filter wheel (the only moving part) through the MACCON motor controller device "/mcon0".
- Devices lsm1 and lsm1 control Lakeshore 218S temperature monitors and relays through an RS232 channel managed by the ISER-12 device "/iser0".
- Device lsm3 controls a Lakeshore 218E temperature monitor (which has no relays) through an RS232 channel managed by the ISER-12 device "/iser0"..
- Device lsc1 controls a Lakeshore 332 temperature controller through an RS232 channel managed by the ISER-12 device "/iser0".
- Device vac1 controls a Pfeiffer vacuum sensor through an RS232 channel managed by the ISER-12 device "/iser0".
- Device dis1 controls an Acromag digital I/O card through device "/acro0".
- Device hb1 generates a heartbeat signal which it communicates to the Acromag digital I/O card through device "/acro0".

The devices are described in detail in the "*Instrument Control Software Design Description*" document, [RD5].

5.4 *ICS Special Devices*

The Heartbeat device, Pfeiffer vacuum gauge, Lakeshore temperature monitor and Lakeshore temperature controller are special devices based on Base ICS device templates.

5.5 *ICS Assemblies*

ICS assemblies are defined in the configuration file "vcmcfg/config/vcmcfgINS.cfg" (installed in $INS_ROOT/SYSTEM/COMMON/CONFIGFILES/".

No.	Name	Description	Commands	Values
1	INS.SENSORS	All instrument sensors and environmental controllers.	STATUS	N/A

2	INS.INFRARED	All devices used to make infrared exposures.	STATUS	N/A
3	INS.MODE	Instrument mode	SETUP STATUS	See Section 5.1.
4	INS.PATH	Instrument light path.	EXPSTRT EXPEND STATUS	Only one possibility: INFRARED
5	INS.FILTER	Instrument filter assembly: moves named filter to beam, with automatic in-position switch check and avoidance of bright filters.	SETUP -function INS.FILTER.NAME	Name of filter.
6	INS.LOADER	Instrument filter loader: moves named filter to load position.	SETUP -function INS.LOADER.NAME	Name of filter.

5.6 Exposures

See the description in the “Template Instrument User and Maintenance Manual”, [AD3], for comparison.

5.6.1 Exposure types

The VISTA IR Camera only supports the exposure types defined by IRACE, [RD39].

5.6.2 Exposure ID

The VIRCAM OS software maintains an exposure ID in the manner described in the “BOSS User Manual”, [RD48]. The first SETUP command must be issued with the “-expoId 0” parameter, which then returns an exposure ID to be specified with all subsequent commands referring to that exposure.

5.6.3 Exposure status

The exposure status is managed by the VIRCAM OS software using the BOSS, [RD48], and may be found in the database attribute “<alias>vco:exposure.expStatus”. An exposure table summarising the status of the previous few exposures may also be found the database attribute “<alias>vco:exposure.currExposureTbl”. Any database attributes not displayed on the VIRCAM control panels may be queried using the “ccseiDb” database utility, [RD35].

5.6.4 Exposure parallelism

The VIRCAM software does not make parallel exposures in the sense defined by the BOSS software, [RD48]. There is only one detector controller[15] (IRACE). However, the software does rely on some parallel operation between the ICS, TCS, IRACE and OS subsystems to maximise the data acquisition performance:

- The bossArchiver process runs independently from the vcoControl OS server process and merges together the FITS files from completed exposures without holding up the data acquisition. The bossArchiver is capable of lagging several exposures behind because it operates from a queue (see [RD48]).
- VIRCAM templates do not wait for the data acquired by the template to be merged by bossArchiver. It is therefore possible for the data merging activity from one template to continue in parallel with the instrument configuration and/or telescope preset needed by the next template.
- Since all 16 detectors are read out simultaneously, the data acquired by the IRACE system is safe as soon as the data from the first detector is has been saved. IRACE normally generates its "end of exposure" event when the data from the last detector has been saved. The VIRCAM software configures IRACE to generate the event after the first detector data has been saved by specifying the keyword

  ```
  DET.EXP.WAITLAST F;         # Wait for last exposure data
  ```

  in the IRACE configuration file (vcmcfgIRDCS.cfg, see [RD39]). This "end of exposure" event is associated by BOSS with the "CanSetupNextObs" signal. However, since the VIRCAM software could reconfigure the TCS or instrument after this event, the BOSS needs to be configured to collect the FITS header earlier than usual. This is achieved by specifying the keyword

  ```
  OCS.DET1.OPTSEQ T;     # Enable early collection of header
  ```

  in the OS configuration file (vcmcfgOS.cfg).
- The VIRCAM software uses the "CanSetupNextObs" and "CanStartNextObs" options to allow the BOSS to complete an exposure in parallel with the preparation steps leading up to the next exposure. The "ObsEnd" option is only used by templates, such as those which generate HOWFS observations, which need to do something with the data.

The OS commands used to set up this subsystem parallelism are described in the next section.

[15] The TCCD controllers used by the VISTA TCS to make exposures with the Low Order Wavefront Sensors don't count because these detectors are not managed by the VIRCAM software.

5.6.5 Exposure life-cycle

The exposure life cycle is very similar to that described in the "*Template Instrument User and Maintenance Manual*", [AD3], and the "*BOSS User Manual*", [RD48], i.e.

- An exposure is defined with a SETUP command which specifies "-expoId 0" and specifies the INS.MODE parameter. The OS software forwards appropriate SETUP commands to the VIRCAM subsystems.
- The first SETUP command returns a new exposure ID, which is specified in all subsequent commands referring to this particular exposure.
- A "START –expoId <exposure ID>" is used to start the exposure. During the exposure:
    - The OS calls "tifGetFitsStart", which requests the VISTA TCS to provide the FITS information for the start of the exposure.
    - The OS sends the START command to the IRACE DCS.
    - The OS sends the EXPSTRT command to the ICS. (NOTE: The EXPSTRT command is normally sent before the START command, but this ordering has been modified for VIRCAM to allow the start of an exposure and collection of ICS FITS information to be done in parallel.)
    - The IRACE integrates and reads all 16 of the science detectors simultaneously (integration time specified by the DET.DIT parameter and number of integrations specified by the DET.NDIT parameter). If the exposure consists of more than one integration, the IRACE coadds the integrations on the number crunchers.
    - The IRACE transfers one data frame per detector to the instrument workstation. (There is always one frame per detector per exposure, regardless of how many integrations make up the exposure). The IRACE sends an "exposure completed" event to the OS when the *first* data frame has been transferred (since DET.EXP.WAITLAST=F).
    - While the IRACE is still writing the remaining 15 files, the OS calls "tifGetFitsEnd", which requests the VISTA TCS to provide the FITS information for the end of the exposure. (The collection of header information happens early because OCS.DET1.OPTSEQ=T).
    - The OS sends the EXPEND command to the ICS, followed by the command "STATUS –header –dumpFits". This makes the ICS collect FITS header information at the end of the exposure and dump its FITS header to a file.
    - In parallel, IRACE finishes writing the 16 data files.
- A WAIT command may be used to wait for the exposure to reach one of the following stages (as described in [RD48]):
    - CanSetupNextObs — a SETUP command can be used to change the instrument configuration and prepare for the next exposure.
    - CanStartNextObs — a START command can be used to make another exposure.
    - ObsEnd — the exposure has completed and the data files have been merged.
- At the end of an exposure the OS merges together the following files:

- Partial header file from the OS itself;
- Partial header file from the TCS;
- Partial header file from the ICS;
- 16 FITS files containing data from each of the 16 detectors;

to make one multi-extension FITS file (see section 5.6.7 below).

- The OS than informs the VLT online archiver (VOLAC) that a new file is available for archiving.

5.6.6 Exposure execution

The VIRCAM software generates a series of exposures efficiently by means of the following sequence of commands:

Set up the parameters which are common to all exposures in the sequence and obtain a new exposure ID:

```
SETUP -expoId 0 -file VIRCAM.ref -function INS.MODE IMAGING
                                           INS.FILT1 J
                                           DET.DIT 10.0
                                           DET.NDIT 1
                                           etc…
<expoId_this>       ← New exposure ID returned
```

Set up the parameters which need to be specified explicitly for each new exposure

```
SETUP -expoId <expoId_this> -function DPR.CATG SCIENCE
                                      DPR.TECH IMAGE
                                      DPR.TYPE OBJECT
                                      etc…
<expoId_this>       ← Same exposure ID returned
```

Execute the first exposure

```
START -expoId $expoId_this
OK
```

Save the exposure ID, `expoId_prev = expoId_this`

For each subsequent exposure from the second to the last

Wait for IRACE to be ready for the next exposure

```
WAIT -expoId $expoId_prev -cond CanStartNextObs
OK
```

Set up the parameters which need to be specified explicitly for each new exposure

```
SETUP -expoId <expoId_this> -function DPR.CATG SCIENCE
                                      DPR.TECH IMAGE
                                      DPR.TYPE OBJECT
                                      etc…
<expoId_this>       ← Same exposure ID returned
```

Execute the first exposure

```
START -expoId $expoId_this
OK
```

Save the exposure ID, **expoId_prev = expoId_this**

Go back and do the next exposure

The combination of SETUP, START, WAIT commands and the use of expoId_this and expoId_prev variables allows setup, data acquisition and data merging operations to be overlapped. The expoId_prev variable is remembered to that any subsequent exposure sequence can wait for the last exposure of the previous sequence. Each template waits for the data merging from any previous template to finish by executing the following command.

```
WAIT 0 -all
OK
```

5.6.7 Data merging

At the end of each exposure the OS software needs to merge together the partial header files created by the subsystems with the data files created by IRACE. By default, the OS merges the partial header files together with the *first* file created by IRACE to make a single FITS file. This behaviour is not appropriate for VIRCAM, so the configuration keyword

```
OCS.DET1.ARCMODE "MERGEALL";     # Enable merging of IRACE data
```

is defined in the OS configuration file (vcmcfgOS.cfg), which configures the OS to merge all of the 16 files created by IRACE to make a single multi-extension FITS file.

- The FITS headers from the OS itself, the VISTA TCS and the ICS are written to the top level FITS header.
- Each of the 16 files written by IRACE becomes a FITS “IMAGE” extension, and the header information written by IRACE becomes the extension header.
- The OS writes World Coordinates information for each detector chip to the appropriate extension header.
- The OS also copies the “EXPTIME”, “MJD-OBS”, “DATE-OBS”, “DET DIT”, “DET NCORRS NAME” and “DET NDIT” IRACE header keywords from the first extension header to the top level header, where they belong:

The merging process is summarised in Figure 40 below.

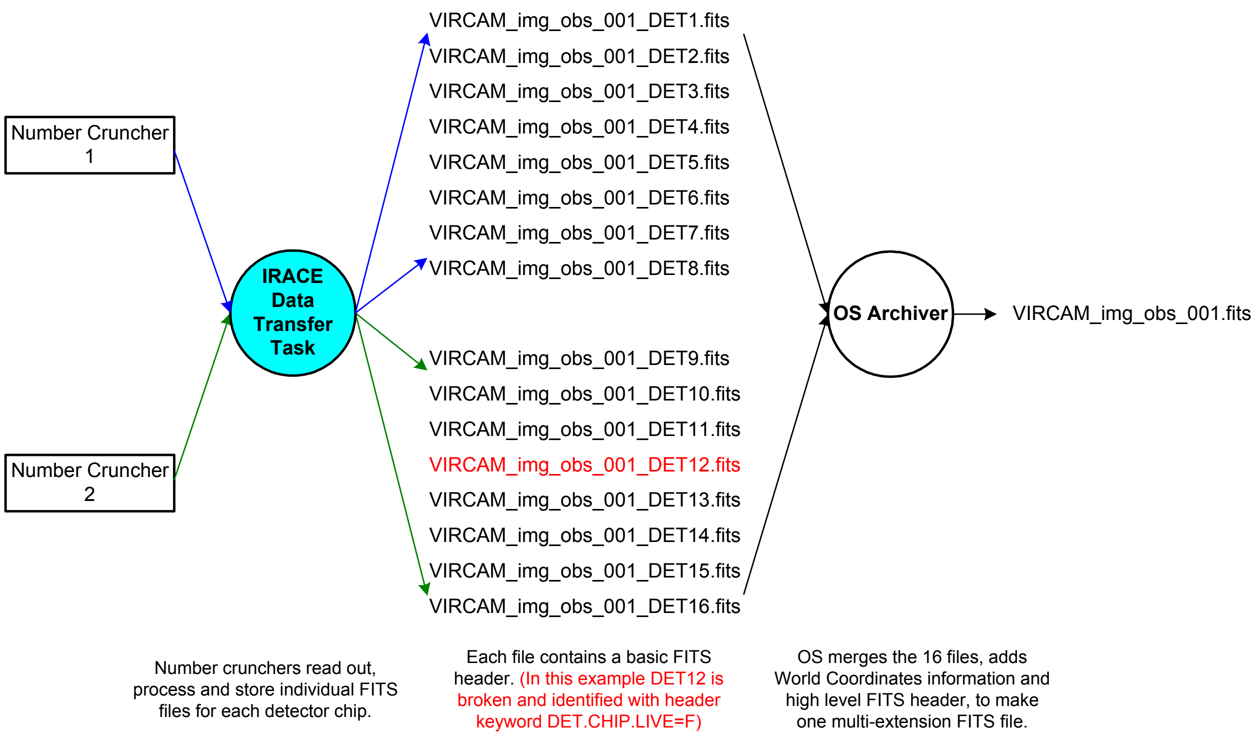


Figure 40 VIRCAM Data Merging

5.7 *Operational States*

The VISTA IR Camera uses the same states as any standard ESO/VLT instrument, as described in the *"Template Instrument User and Maintenance Manual",* [AD3]. In particular:

- The OFF or LOADED state means the VIRCAM software is not fully initialised.
- The STANDBY state means the VIRCAM software is "read-only". It will report the status of the instrument, but commands which change the instrument configuration (such as to move the filter wheel or change a temperature target) are disabled.
- The ONLINE state is the normal operational state.

The instrument (or any of its subsystems) may be reset with the command sequence OFF, ONLINE.

5.8 *Commands*

Commands are normally sent to the VIRCAM OS from the running template scripts, but for development, maintenance and debugging purposes, individual commands can be sent directly to the VIRCAM OS, ICS or DCS servers while they are running with these respective Linux shell commands:

```
msgSend "" vcoControl "<command>" "<parameters>"
msgSend "" vciControl "<command>" "<parameters>"
```

```
msgSend "" iracqServer "<command>" "<parameters>"
```

This command may be used to send commands directly to the HOWFS image analysis server:

```
msgSend "" vchoiaServer "<command>" "<parameters>"
```

5.8.1 Standard BOSS commands

The VIRCAM OS server process (vcoServer) obeys all the standard commands defined by BOSS. Useful commands for querying and debugging are:

STATE: Query the state of the instrument or subsystem.
Parameters:
-subSystem <string>
Subsystem whose state is to be queried (optional). If not given, query the whole instrument.
Returns a state description string.

STATUS: Query the values of certain instrument keywords.
Parameters:
-expoId <exposure id>
Exposure ID.
-function <keyword list>
List of one or more keywords to be displayed.
Returns a string containing list of "<keyword> <value>" pairs.

VERSION: Query the software version number of the VIRCAM OS.
Parameters:
-subSystem <string>
Subsystem to forward the command to.
Returns a software version description string.

For a description of other commands (such as SETUP, START and WAIT), see the *"BOSS User Manual"*, [RD48].

5.8.2 Special OS commands

The VIRCAM OS server recognises the following VIRCAM-specific commands. The commands are designed to be used by the template scripts, and should not normally be executed separately. The commands for converting spherical coordinates are used to calculate the (RA,Dec) offsets for the pawprints within a tile, or to find a suitable twilight sky field:

CHIPS: Define which detector chips are not vignetted (for HOWFS data).
Parameters:
-all

If present, all chips are valid.

-list <chip1>,<chip2>,…

List of unvignetted chips.

Returns OK.

NEWOFF: Start a new sequence of offsets.

Parameters:

-type <offset type>

Type of sequence (OFFSET, JITTER or USTEP).

-number <integer>

How many steps in the sequence will there be?

-id <integer>

ID of offset sequence pattern.

-reset

If keyword present, reset all offset sequence

Returns OK.

NEXTOFF: Declare the next step in an offset sequence (previously started with the NEWOFF command).

Parameters:

-type <offset type>

Type of sequence (OFFSET, JITTER or USTEP).

-xoff <real>

X offset of this step (arcseconds).

-yoff <real>

Y offset of this step (arcseconds).

-index <integer>

Index number of this step (optional). Increment index if not provided.

Returns OK.

DSEP: Calculate the angular distance between two points on a sphere.

Parameters:

-long1 <DDDMMSS.TTT>

Longitude/RA of first point in hours or degrees.

-lat1 <+/-DDDMMSS.TTT>

Latitude/Dec of first point in degrees.

-long2 <DDDMMSS.TTT>

Longitude/RA of second point in hours or degrees.

-lat2 <+/-DDDMMSS.TTT>

Latitude/Dec of second point in degrees.

-hours

If keyword provided, longitude coordinates are hours.

-noencode

If keyword provided, coordinates are plain numbers,

not sexagesimally encoded.
Returns the separation in degrees.

SKY2AAZ: Convert equatorial coordinates (alpha, delta) on the sky to telescope mount coordinates (altitude,azimuth).
Parameters:
-alpha <HHMMSS.TTT>
RA in hours.
-delta <+/-DDMMSS.TTT>
Declination in degrees.
-lst <HHMMSS.TTT>
Local sidereal time in hours.
-phi <+/-DDMMSS.TTT>
Latitude of the observatory in degrees.
Returns a string containing “Azimuth, Altitude”.

AAZ2SKY: Convert telescope mount coordinates (altitude, azimuth) to sky equatorial coordinates (alpha, delta).
Parameters:
-az <+/-DDD.TTT>
Azimuth in degrees.
-alt <DD.TTT>
Altitude in degrees.
-lst <HHMMSS.TTT>
Local sidereal time in hours.
-phi <+/-DDMMSS.TTT>
Latitude of the observatory in degrees.
Returns a string containing “RA, Dec”.

SKY2XY: Given the coordinates of the telescope axis (telalpha, teldelta the coordinates of a target (alpha, delta), compute the tangent plane X,Y coordinates of that target on the VISTA IR Camera focal plane.
Parameters:
-alpha <HHMMSS.TTT>
Target RA in hours.
-delta <+/-DDMMSS.TTT>
Target declination in degrees.
-telalpha <HHMMSS.TTT>
Telescope axis RA in hours.
-teldelta <+/-DDMMSS.TTT>
Telescope axis declination in degrees.
-telrot <+/-DDD.TTT>
Telescope rotator angle in degrees.
-scale <real>
Plate scale in arcseconds per mm

-disto <real>
Barrel (-ve) or pincushion (+ve) distortion factor
-xreflect
If keyword present, all X coordinates are reflected.
Returns a string containing "X, Y in mm".

XY2SKY: Given the coordinates of the telescope axis (telalpha, telapha) and the coordinates of a target on the focal plane (x,y), compute the tangent plane (alpha, delta) coordinates of that target on the sky.
Parameters:
-x <real>
Camera X coordinate of target in mm.
-y <real>
Camera Y coordinate of target in mm.
-telalpha <HHMMSS.TTT>
Telescope axis RA in hours.
-teldelta <+/-DDMMSS.TTT>
Telescope axis declination in degrees.
-telrot <+/-DDD.TTT>
Telescope rotator angle in degrees.
-scale <real>
Plate scale in arcseconds per X,Y unit
-disto <real>
Barrel (-ve) or pincushion (+ve) distortion factor
-xreflect
If keyword present, all X coordinates are reflected.
Returns a string containing target "RA, Dec".

XY2TP: Given the coordinates of the telescope axis (telalpha, telapha) and the coordinates of a target on the focal plane (x,y), compute the telescope pointing coordinates (telalpha, teldelta) of the tangent plane centre needed to place target at the particular X,Y coordinate.
Parameters:
-alpha <HHMMSS.TTT>
Target RA in hours.
-delta <+/-DDMMSS.TTT>
Target declination in degrees.
-x <real>
Camera X coordinate of target in mm.
-y <real>
Camera Y coordinate of target in mm.
-telrot <+/-DDD.TTT>
Telescope rotator angle in degrees.
-scale <real>
Plate scale in arcseconds per X,Y unit

-disto <real>
Barrel (-ve) or pincushion (+ve) distortion factor
-xreflect
If keyword present, all X coordinates are reflected.
Returns a string containing target "telRA, telDec".

SKY2DPA: Given the coordinates of the telescope axis (telalpha, telapha) and the coordinates of a target (alpha, delta), compute the difference in the parallactic angle between those two coordinates.
Parameters:
-alpha <HHMMSS.TTT>
Target RA in hours.
-delta <+/-DDMMSS.TTT>
Target declination in degrees.
-telalpha <HHMMSS.TTT>
Telescope axis RA in hours.
-teldelta <+/-DDMMSS.TTT>
Telescope axis declination in degrees.
Returns a string containing the parallactic angle difference in degrees.

SETWCS: Define the nominal World Coordinates for the next exposure.
Parameters:
-ctype1 <string>
Type of coordinate for axis1 (PIXEL, RA---ZPN or RA---TAN).
-ctype2 <string>
Type of coordinate for axis2 (PIXEL, RA---ZPN or RA---TAN).
-pscale <real>
Nominal VISTA IR plate scale in arcsec/pixel.
-pangle <real>
Camera position angle wrt telescope in degrees.
-xref <real>
X coordinate of reference pixel.
-yref <real>
Y coordinate of reference pixel.
Returns OK.

See the *"Observation Software Design Description",* [RD4], for details.

5.8.3 Special ICS commands

There are no special ICS commands. The ICS server process (vciServer) recognises the standard commands defined by the "Base ICS", as described in [RD46]. In particular, the ICS recognises the "STATE", "STATUS" and "VERSION" commands, which provide useful diagnostic information.

5.8.4 Special DCS commands

VIRCAM uses IRACE, which obeys all the commands described in the *"IRACE-DCS User Manual"*, [RD39]. There are no special DCS commands for VIRCAM.

5.8.5 Special HOWFS commands

The High Order Wavefront Sensor server process (vchoiaServer) obeys all the standard ESO/VLT state changing and querying commands, as described in section 5.8.1, but it also recognises the following VISTA-specific commands:

ANASTAR: Start image analysis (using the file name and parameters specified in one or more SETUP commands) and return a unique analysis ID.
Parameters:
None.
Returns analysis ID.

WAIT: Wait for analysis procedure to complete.
Parameters:
-id <integer>
Current analysis ID.
Returns OK.

CHECK: Check whether the analysis procedure is running.
Parameters:
None.
Returns analysis ID if busy or 0 if not busy.

See the *"High Order Wavefront Sensor Software Design Description",* [RD7], for details.

5.9 Tcl libraries

The VIRCAM templates defined in the vcoseq and vcmseq modules use the base classes provided by the *tpl* template library, [RD49], plus:

- vcoseqLib.tcl — General purpose sequencing functions.
- vcoseqDR.tcl — Data reduction functions, based on tplDR.
- vcoseqHOWFS.tcl — HOWFS sequencing, control and interface functions.
- vcoseqICS.tcl — ICS interface functions, based on tplICS.
- vcoseqIRACE.tcl — IRACE interface functions, based on tplIRACE.
- vcoseqOBS.tcl — Observation sequencing functions, based on tplOS.
- vcoseqTCS.tcl — TCS sequencing, control and interface functions, based on tplTCS.

5.10 Dictionaries

The VISTA IR Camera software uses the following dictionaries:

- ESO-VLT-DIC.VIRCAM_ICS for the keywords belonging to the ICS subsystem;

- ESO-VLT-DIC.VIRCAM_HOWFS for the keywords belonging to the HOWFS image analysis subsystem;
- ESO-VLT-DIC.VIRCAM_OS for the keywords belonging to the OS subsystem;
- ESO-VLT-DIC.VIRCAM_DCS for the keywords supplementary to IRACE dictionary used with the DCS subsystem;

plus the other dictionaries used by the Template Instrument, [AD3].

5.11 Alias files

None used.

5.12 Configuration files

After installation all configuration files used by the software are installed in the directories $INS_ROOT/SYSTEM/COMMON/CONFIGFILES and $INS_ROOT/SYSTEM/MISC/IRACE. See section 6 for more information on the individual files.

5.13 Setup files and keywords

5.13.1 OCS keywords

The standard OCS keywords used by the BOSS package are defined in the ESO-VLT-DIC.OSB dictionary. The VIRCAM software uses the following special setup keywords, as described in the dictionary ESO-VLT-DIC.VIRCAM_OS. The keywords are mainly used to provide additional FITS header information to aid the VISTA data processing pipeline:

Keyword	**Type**	**Description**
OCS.EXTENDED	string	Defines whether the object being observed is “extended”, i.e. does its data reduction require a separate sky frame?
OCS.WCS.CHIPi.CRPIXi	double	World coordinates. The pixel coordinate of the reference point (normally the intersection of the optical axis with the detector) for the given axis of the given chip, i.
OCS.WCS.CHIPi.SCALEi	double	World coordinates. The magnification of the plate scale for the given axis, as seen on a particular chip, i. 1.0 means no change.
OCS.WCS.CHIPi.ANGLE	double	World coordinates. The orientation of a particular chip i with respect to the focal plane X axis. Angle increases clockwise. 0.0 means a perfectly aligned chip.
OCS.WCS.CHIPi.PVi	double	World coordinates. Parameters used for the non-linear transformation of sky projection to pixel coordinates for a particular chip, i.
OCS.TARG.ALPHAOBJ	double	Right Ascension of target coordinate positioned at

		the pointing origin.
OCS.TARG.DELTAOBJ	double	Declination of target coordinate positioned at the pointing origin.
OCS.TARG.X	double	The X coordinate of the pointing origin on the VISTA camera focal plane in mm (where 0.0 is the field centre).
OCS.TARG.Y	double	The Y coordinate of the pointing origin on the VISTA camera focal plane in mm (where 0.0 is the field centre).
OCS.REQTIME	double	The integration time requested by the observer. The actual integration time (EXPTIME) by be slightly different because of detector control limits (e.g. minimum integration time, MINDIT).
OCS.RECIPE	string	Data reduction recipe to be used by the UK pipeline.
OCS.NOFFSETS	integer	Number of offset positions in a sequence.
OCS.OFFSTNUM	integer	The observation sequence number of the first exposure in the current offset sequence.
OCS.OFFST_ID	string	Unique string identifying the offset pattern (e.g. tile).
OCS.OFFSET_I	integer	Sequence number of offset in sequence, i.e. this is the OFFSET_Ith offset in the sequence
OCS.OFFSET_X	double	Current X offset from reference position in arcseconds.
OCS.OFFSET_Y	double	Current Y offset from reference position in arcseconds.
OCS.NJITTER	integer	Number of jitter positions in a sequence.
OCS.JITTRNUM	integer	The observation sequence number of the first exposure in the current jitter sequence.
OCS.JITTR_ID	string	Unique string identifying the jitter pattern.
OCS.JITTER_I	integer	Sequence number of jitter in sequence, i.e. this is the JITTER_Ith jitter in the sequence
OCS.JITTER_X	double	Current X offset of this jitter from reference position in arcseconds.
OCS.JITTER_Y	double	Current Y offset of this jitter from reference position in arcseconds.
OCS.NUSTEP	integer	Number of microstep positions in a sequence.
OCS.USTEPNUM	integer	The observation sequence number of the first exposure in the current microstep sequence.
OCS.USTEP_ID	string	Unique string identifying the microstep pattern.
OCS.USTEP_I	integer	Sequence number of jitter in sequence, i.e. this is the USTEP_Ith microstep in the sequence
OCS.USTEP_X	double	Current X offset of this microstep from reference

		position in arcseconds.
OCS.USTEP_Y	double	Current Y offset of this microstep from reference position in arcseconds.
OCS.NEXP	integer	The number of exposures made at each telescope dwell (i.e. the innermost exposure loop). This is different from TPL.NEXP, which gives the number of exposures in an entire template
OCS.EXPNO	integer	The exposures number within a dwell sequence, in the range 1 to OCS.NEXP.

The following OCS keywords exist for providing additional FITS header information for HOWFS exposures, and are not used for science exposures:

Keyword	Type	Description
OCS.HOWFS.STARPOS.X	double	The X coordinate of the HOWFS star image in the focal plane in mm (as would be seen without the HOWFS beam splitter).
OCS.HOWFS.STARPOS.Y	double	The Y coordinate of the HOWFS star image in the focal plane in mm (as would be seen without the HOWFS beam splitter).
OCS.HOWFS.PREIMG.X	double	Detector X co-ordinate of centre of pre-focal image in pixels.
OCS.HOWFS.PREIMG.Y	double	Detector Y co-ordinate of centre of pre-focal image in pixels.
OCS.HOWFS.POSTIMG.X	double	Detector X co-ordinate of centre of post-focal image in pixels.
OCS.HOWFS.POSTIMG.Y	double	Detector Y co-ordinate of centre of post-focal image in pixels.
OCS.HOWFS.OBSOFF.X	double	Fractional offset of the VISTA central obscuration (due to M2) in the X direction, as seen from the current HOWFS position.
OCS.HOWFS.OBSOFF.X	double	Fractional offset of the VISTA central obscuration (due to M2) in the X direction, as seen from the current HOWFS position.

5.13.2 HOWFS keywords

All the HOWFS setup keywords are described in [RD7]. The most important ones are listed in the following table:

Keyword	Type	Description
HOWFS.IMGFILE	string	Name of FITS file containing the pair of pre-focal and post-focal images to be analysed.
HOWFS.BADMASK	string	Name of FITS file containing bad pixel mask (blank if none).
HOWFS.DARKFILE	string	Name of FITS file containing a dark calibration image

HOWFS.FLATFILE	string	Name of FITS file containing a flat-field calibration image
HOWFS.COFILE	string	Name of FITS file in which to write measured coefficients, along with additional data
HOWFS.PATH	string	Directory path where the above files are kept.
HOWFS.OBSOFF.X	double	Offset of the central obscuration in the X direction in the pupil plane (wrt pupil radius).
HOWFS.OBSOFF.Y	double	Offset of the central obscuration in the Y direction in the pupil plane (wrt pupil radius).
HOWFS.STARPOS.X	double	X position of star in Focal Plane co-ordinates in mm.
HOWFS.STARPOS.Y	double	Y position of star in Focal Plane co-ordinates in mm.
HOWFS.PREIMG.X	double	Detector X co-ordinate of centre of pre-focal image in pixels.
HOWFS.PREIMG.Y	double	Detector Y co-ordinate of centre of pre-focal image in pixels.
HOWFS.POSTIMG.X	double	Detector X co-ordinate of centre of post-focal image in pixels.
HOWFS.POSTIMG.Y	double	Detector Y co-ordinate of centre of post-focal image in pixels.
HOWFS.RECENTRE	logical	Will each image be recentred to its measured centroid?
HOWFS.FLATTEN	logical	Will each image be flattened by subtracting the best-fitting flat surface?
HOWFS.PREBLUR	logical	Will images be blurred by the seeing estimate?
HOWFS.NULLSUB	logical	Will the null aberration be subtracted from results?
HOWFS.NULLABER. MODMODEi	double	Modulus of null aberration mode i in nm
HOWFS.NULLABER. ANGMODEi	double	Angle of null aberration mode i in degrees
HOWFS.STRTPNT. MODMODEi	double	Starting modulus of mode i for the simplex algorithm in nm
HOWFS.STRTPNT. ANGMODEi	double	Starting angle of mode i for the simplex algorithm in degrees
HOWFS.USEMODi	logical	Flag to indicate whether mode i is fitted.
HOWFS.MAXRTOL	double	Maximum relative tolerance ratio for simplex inner loop.
HOWFS.MAXITR	integer	Maximum iteration count for simplex algorithm.
HOWFS.MAXFUN	integer	Maximum number of function evaluations for simplex algorithm.
HOWFS.MAXREP	integer	Maximum number of repeats of simplex outer loop.

HOWFS.MAXCDIF	double	Maximum difference in metres between any wavefront coefficient to trigger a repeat in the outer simplex loop.

5.13.3 INS keywords

The standard INS keywords used by the Base ICS package are defined in the ESO-VLT-DIC.ICB_REF dictionary. The VIRCAM software uses the following special setup keywords, as described in the dictionary ESO-VLT-DIC.VIRCAM_ICS. The keywords are used by special assemblies and devices:

Keyword	Type	Description
INS.FILTER.NAME	string	The name of a filter to be placed in the beam using the INS.FILTER assembly, which tests the in-position switches and minimises the number of light filters swept through the beam when switching between two dark filters.
INS.LOADER	string	Invokes the INS.LOADER assembly, which puts the named filter into the loading bay.
INS.FILTER.STEP	integer	An internal setup keyword used by the INS.FILTER.NAME assembly to execute step N of the movement sequence (not to be used directly).
INS.LSMi.ALRMSPj	string	Define the lower and upper trigger points for Lakeshore 218S device i alarm channel j. The string is given in the form "lower limit<upper limit", e.g. "260.1<280.5".
INS.LSMi.RLYHIj	integer	Define the relay to be triggered by a high alarm on channel j on Lakeshore 218S device i.
INS.LSMi.RLYLOj	integer	Define the relay to be triggered by a high alarm on channel j on Lakeshore 218S device i.
INS.LSMi.RLYSTj	integer	Explicitly set relay j on Lakeshore 218S device i into a particular state (0=off, 1=on).
INS.LSCi.SETPj	double	Define a new temperature control set point for channel j of Lakeshore 332 device i.

5.13.4 DCS keywords

Standard IRACE DCS keywords are used, see [RD39].

5.14 Pattern files

The VISTA IR Camera defines a set of pattern files, which describe the tile patterns, jitter patterns and microstep patterns recognised by the VIRCAM software. These pattern files are delivered to ESO along with the instrument package (see section 2.2.2 on page 41) so that observation preparation tools can also see which patterns are available. The pattern files may be found in the config directory of the vcotsf module, and are installed in

$INS_ROOT/SYSTEM/MISC/VISTA when that module is built. There are 3 types of pattern file, described below:

5.14.1 Tile pattern files

Tile pattern files have names matching “VIRCAM_Tile<name>.paf”, where <name> is the name of the tile pattern identified with the “SEQ.TILE.ID” template parameter (see the description of the tile template in section 11.7.6.3 on page 245). A tile pattern file has a similar format to a template signature file. The important part of the file is the definition of 4 parameters; for example:

```
SEQ.TILE.NAME          "6 step n pattern";
SEQ.TILE.OFFSETX       "-0.475 -0.475 -0.475 0.475 0.475  0.475";
SEQ.TILE.OFFSETY       "-0.475  0.0    0.475 0.475 0.0   -0.475";
SEQ.TILE.OFFSETROT     " 0.0    0.0    0.0   0.0   0.0    0.0";
```

The SEQ.TILE.NAME gives the name of the pattern described by the file. The other 3 keywords give a list of X, Y and rotator offsets, which are read from left to right. This particular pattern (which is the same one shown in Figure 13 on page 40) translates to these 6 offsets:

Step	X offset	Y offset	Rotator offset
1	-0.475	-0.475	0.0
2	-0.475	0.0	0.0
3	-0.475	0.475	0.0
4	0.475	0.475	0.0
5	0.475	0.0	0.0
6	0.475	-0.475	0.0

The units of the offsets are detector widths[16], so these numbers translate directly into the 0.95 detector width steps in the X direction and the 0.475 detector width steps in the Y direction. All the offsets are relative to the coordinates of the target acquired during the last acquisition template (survey observation blocks normally consist of an acquisition template followed by a tile template).

Although the tile pattern gives offsets in detector widths, the offsets can be scaled by means of the SEQ.TILE.SCALE template parameter. Changing this parameter is not recommended unless done during the survey definition — setting it to a value less than 1.0 will increase the amount of overlap between the pawprints but require the tiles making up a survey to be spaced closer together.

The rotator offsets are in degrees and are almost always zero — the rotator offset exists for special purpose applications, such as tweaking the pawprint overlaps to avoid a bright star or avoid putting a detector defect on an object of interest.

[16] It has been suggested that the units be changed to arcseconds.

5.14.2 Jitter pattern files

Jitter pattern files have names matching "VIRCAM_Jitter<name>.paf", where <name> is the name of the jitter pattern identified with the "SEQ.JITTER.ID" template parameter (see the description of the pawprint and tile templates in sections 11.7.6.2 and 11.7.6.3 on page 245). A jitter pattern file is very similar to a tile pattern file, with 4 important parameters; for example:

```
SEQ.JITTER.NAME       "2x2 |\| pattern";
SEQ.JITTER.OFFSETX    "-10.0 -10.0   10.0  10.0";
SEQ.JITTER.OFFSETY    "-10.0  10.0  -10.0  10.0";
SEQ.JITTER.OFFSETROT  "  0.0   0.0    0.0   0.0";
```

The SEQ.JITTER.NAME gives the name of the pattern described by the file. The other 3 keywords give a list of X, Y and rotator offsets, which are read from left to right, just like the tile pattern. This particular pattern (which is similar to the one shown in Figure 14 on page 40) translates to these 4 offsets:

Step	X offset	Y offset	Rotator offset
1	-10.0	-10.0	0.0
2	-10.0	10.0	0.0
3	10.0	-10.0	0.0
4	10.0	10.0	0.0

All the offsets are relative to the coordinates of the current pawprint. The default units of the offsets are arcseconds, but they may be scaled by setting the SEQ.JITTER.SCALE template parameter. For example, setting this parameter to 2.0 will convert all the above offsets to 20.0 arcseconds. This scaling allows the jitter patterns to be applied to a variety of different situations.

The rotator offsets are in degrees and are almost always zero — the rotator offset exists for special purpose applications.

5.14.3 Microstep pattern files

Microstep pattern files have names matching "VIRCAM_Ustep<name>.paf", where <name> is the name of the microstep pattern identified with the "SEQ.USTEP.ID" template parameter (see the description of the pawprint and tile templates in sections 11.7.6.2 and 11.7.6.3 on page 245). They are very similar to jitter pattern files, with 4 important parameters; for example:

```
SEQ.USTEP.NAME       "2x2 pattern";
SEQ.USTEP.OFFSETX    "0.0 0.0 0.5 0.5";
SEQ.USTEP.OFFSETY    "0.0 0.5 0.5 0.0";
SEQ.USTEP.OFFSETROT  "0.0 0.0 0.0 0.0";
```

The SEQ.USTEP.NAME gives the name of the pattern described by the file. The other 3 keywords give a list of X, Y and rotator offsets, which are again read from left to right. This particular pattern (which is similar to the one shown in Figure 15 on page 41) translates to these 4 offsets:

Step	X offset	Y offset	Rotator offset
1	0.0	0.0	0.0
2	0.0	0.5	0.0
3	0.5	0.5	0.0
4	0.5	0.0	0.0

All the offsets are relative to the coordinates of the current jitter position. The default units of the offsets are detector pixels. Since scaling these offsets does not make sense. the SEQ.USCALE.SCALE template parameter value is locked at 1.0.

The rotator offsets exist for compatibility with the tile and jitter patterns, but it doesn't make sense to use values other than zero for microstepping.

5.15 FITS files

The VIRCAM software generates the following kinds of file in FITS format.

5.15.1 Science data

These data files (made in IMAGING mode) are written to disk on the instrument workstation in multi-extension FITS format adhering the standards defined in the ESO-VLT *"Data Interface Control Document"*, [RD23]. Each file contains a header plus 16 IMAGE extensions:

- A top-level FITS header.
- IMAGE extension containing data for detector 1.
- IMAGE extension containing data for detector 2.
- etc..
- IMAGE extension containing data for detector 15.
- IMAGE extension containing data for detector 16.

The files are written to $INS_ROOT/SYTEM/DETDATA. VOLAC is informed of each new complete file using the standard protocol, [RD31].

5.15.2 HOWFS data

Data files made in HOWFS mode have virtually the same format as science data, with the exception that only one of the 16 IMAGE extensions contains valid data. The top level FITS header contains a "VALIDEXT" keyword pointing to the extension containing the valid data. In addition, each extension contains a "DET CHIP VIGNETD" keyword which is "T" if the extension contains unvignetted data.

Just as with science data, the files are written to $INS_ROOT/SYTEM/DETDATA and VOLAC is informed of each new complete file. However, in addition, HOWFS calibration files are copied to $INS_ROOT/SYSTEM/HOWFSDATA, where they should remain until explicitly removed or overwritten (i.e. they are protected against the DETDATA directory being cleaned out). HOWFS calibration file names have the following names:

- VIRCAM_HOWFS_BIAS_<filter>.fits
  BIAS calibration for HOWFS filter <filter>.

- VIRCAM_HOWFS_DARK_<filter>_<exptime>.fits
  DARK calibration for HOWFS filter <filter> and exposure time <exptime>.

- VIRCAM_HOWFS_FLAT_<filter>.fits
  Flat-field calibration for HOWFS filter <filter>.

These files are necessary because the HOWFS image analysis software is an on-line data reduction system and requires access to suitable calibration files.

5.15.3 HOWFS coefficients files

The HOWFS image analysis software writes its results to multi-extension FITS files which contains:

- A top-level FITS header.
- ASCII table extension containing the fitted wavefront coefficients in mirror modes.
- ASCII table extension containing the null wavefront coefficients in mirror modes.
- ASCII table extension containing the starting wavefront coefficients in mirror modes.
- ASCII table extension containing the fitted wavefront coefficients in Zernikes.
- IMAGE extension containing the original pre-focal image
- IMAGE extension containing the original post-focal image
- IMAGE extension containing the ray traced pre-focal trial image
- IMAGE extension containing the ray traced post-focal trial image
- IMAGE extension containing the residuals for the pre-focal image
- IMAGE extension containing the residuals for the post-focal image

These files are written to $INS_ROOT/SYSTEM/HOWFSDATA and their existence is not reported to VOLAC. See [RD7] for details.

5.16 Public on-line database attributes

5.16.1 World coordinates

The world coordinates for the current exposure may be found in the following database attributes, for each detector <nn> from 01 to 16:

- <alias>VIRCAM:OS:wcs:detector<nn>.crtype1 — CTYPE1
- <alias>VIRCAM:OS:wcs:detector<nn>.crtype2 — CTYPE2

- <alias>VIRCAM:OS:wcs:detector<nn>.crval1 — CRVAL1
- <alias>VIRCAM:OS:wcs:detector<nn>.crval2 — CRVAL2
- <alias>VIRCAM:OS:wcs:detector<nn>.cdelt1 — CDELT1
- <alias>VIRCAM:OS:wcs:detector<nn>.cdelt2 — CDELT2
- <alias>VIRCAM:OS:wcs:detector<nn>.cd_1_1 — CD_1_1
- <alias>VIRCAM:OS:wcs:detector<nn>.cd_1_2 — CD_1_2
- <alias>VIRCAM:OS:wcs:detector<nn>.cd_2_1 — CD_2_1
- <alias>VIRCAM:OS:wcs:detector<nn>.cd_2_2 — CD_2_2

This information can be used by real-time display tools to lay out images from the 4x4 array of detectors on the screen.

5.16.2 Wavefront coefficients

The completion status of the last HOWFS image analysis procedure may be found in the following database attribute:

- <alias>VIRCAM:HOWFS:fit:simplexDiag.successFlag

This attribute will contain a 1 if the wavefront analysis converged or 0 if it did not converge.

The wavefront coefficients, in mirror modes, generated by the last HOWFS image analysis may be found in the following database vectors:

- <alias>VIRCAM:HOWFS:fit:coeffs.modulus — modulus of coeffs in nm
- <alias>VIRCAM:HOWFS:fit:coeffs.angle — angle of coeffs in degrees

The same information can be found expressed as Zernike coefficients in the following database vectors:

- <alias>VIRCAM:HOWFS:fit:zernike.modulus — modulus of coefficients in nm
- <alias>VIRCAM:HOWFS:fit:zernike.order — order of coefficients, i.e. Z<i>

5.17 Operational logs

The VIRCAM software writes the following operational log files:

- Messages reported by the VIRCAM template scripts are saved, via BOB, to the file $VLTDATA/tmp/bob_vcoControl.log.
- Messages logged by the HOWFS image analysis software are recorded to the file $VLTDATA/tmp/vchoiaServer.log
- General log messages are recorded in the file $VLTDATA/tmp/logFile and may be monitored with the "logMonitor" utility.
- CCS messages from the wvcam environment are logged to file $VLTDATA/ENVIRONMENTS/wvcam/ccs_log.

The VIRCAM software reports the following information as “FITS_LOG” events:

- Device initialisation;
- Filter wheel initialisation (datum) time and reference switch offset measurement.
- Change of thermal state.
- Mean science detector temperature;
- Mean Focal Plane Array (FPA) temperature;
- Mean ambient, cryostat window and cryostat tube temperatures.

The Base ICS historian records the following information in the VIRCAM ICS historian database “<alias>VIRCAM:ICS:HISTORIAN”:

- Slow history records, which record all the temperatures and vacuum pressures sensed by the instrument every 10 minutes in a circular buffer containing 50 days worth of measurements.
- Fast history records, which record the position of the filter wheel every second in a circular buffer containing an hour’s worth of measurements.

By default, only the slow history records are activated. The fast records may be activated using the historian manager, “hisConfigTool”. The historian records are used by the plotting utilities described in section 4.15.3 on page 86. The historian is configured in the file “vci/dbl/vciHISTORIAN.db”.

The VIRCAM software also includes utilities which can record sensor information to log files on demand. The command

```
% vciLogging
```

spawns 3 processes (vciLogTemperatures, vciLogVacuum and vciLogThermalControl), which record temperature and pressure readings every 10 minutes to log files whose names are of the form “$VLTDATA/tmp/<environment>_<date>_ccseiDb<description>.log”. By default, the logging stops after a week, but can be continued by repeating the above command. These utilities can be used to provide supplementary logging information for diagnostic or commissioning purposes (see section 8.5 on page 147).

5.18 Templates

The VIRCAM software uses the following templates, as summarised in Figure 41 below. The use for the templates is specified in the “VISTA *IR Camera Calibration Plan*”, [RD18]. There are a series of templates for each of the operating modes described in section 5.1.

HOWFS mode

- *VIRCAM_howfs_acq.tsf*
- *VIRCAM_howfs_acq_domescreen.tsf*
- VIRCAM_howfs_cal_reset.tsf
- VIRCAM_howfs_cal_dark.tsf
- VIRCAM_howfs_cal_domeflat.tsf
- VIRCAM_howfs_obs_exp.tsf
- VIRCAM_howfs_obs_wfront.tsf

IMAGING mode

- *VIRCAM_img_acq.tsf*
- *VIRCAM_img_acq_twilight.tsf*
- *VIRCAM_img_acq_domescreen.tsf*
- VIRCAM_img_cal_reset.tsf
- VIRCAM_img_cal_dark.tsf
  - VIRCAM_img_cal_darkcurrent.tsf
- VIRCAM_img_cal_domeflat.tsf
  - VIRCAM_img_cal_linearity.tsf
  - VIRCAM_img_cal_noisegain.tsf
- VIRCAM_img_cal_twiflat.tsf
- VIRCAM_img_cal_persistence.tsf
- VIRCAM_img_cal_std.tsf
- VIRCAM_img_obs_exp.tsf
  - VIRCAM_img_cal_crosstalk.tsf
  - VIRCAM_img_cal_illumination.tsf
- VIRCAM_img_obs_paw.tsf
  - VIRCAM_img_obs_tile.tsf
  - VIRCAM_img_obs_offsets.tsf

Figure 41 Hierarchy of VISTA IR Camera Templates

A much more detailed description of the purpose, parameters and action of each template may be found in section 11.7 on page 113. *NOTE: The VIRCAM software also provides a number of maintenance templates, which are described in section 8.10 on page 159.*

In each mode there are three kinds of templates:

- Acquisition templates (*shown in blue italic*), which define the operating mode and telescope target parameters. Each Observation Block begins with an acquisition template defining the primary target to which that Observation Block refers. Acquisition templates do not generate exposures.

- Calibration templates (shown in red), which obtain exposures necessary for calibrating observations in a particular instrument mode. A calibration template can result in one or more exposures being made.

- Observation templates (shown in black), which obtain the exposures necessary to make science observations. An observation template can result in one or more exposures being made.

Each Observation Block is expected to start with an acquisition template followed by one or more observation templates. Certain acquisition templates are designed to be paired with certain observation templates, as shown in the following table:

Observation template	Use with this acquisition template
VIRCAM_img_cal_reset VIRCAM_img_cal_dark VIRCAM_howfs_cal_reset VIRCAM_howfs_cal_dark	No acquisition template needed.
VIRCAM_img_cal_domeflat VIRCAM_howfs_cal_domeflat VIRCAM_img_cal_linearity	VIRCAM_img_acq_domeflat or VIRCAM_img_acq_altaz
VIRCAM_img_cal_twiflat	VIRCAM_img_acq_twilight or VIRCAM_img_acq_altaz
VIRCAM_howfs_obs_wfront VIRCAM_howfs_obs_exp	VIRCAM_howfs_acq
VIRCAM_img_obs_paw VIRCAM_img_obs_tile<N>	VIRCAM_img_acq

NOTE: Some observation templates recall the TCS position saved by the most recent acquisition template. An acquisition template should be executed before running a series of observation templates, to ensure they behave correctly.

5.18.1 HOWFS Templates

5.18.1.1 HOWFS acquisition templates

- **VIRCAM_howfs_acq.tsf** — Acquire a HOWFS source. This template sets the instrument into HOWFS mode and selects a HOWFS beam-splitting filter/position combination. It also points the telescope and arranges for light from the specified star to fall on the HOWFS beam splitter.

- **VIRCAM_howfs_acq_domescreen.tsf** — This template sets the instrument into HOWFS mode and selects a HOWFS beam-splitting filter. It also moves the telescope to point at the flat-field screen in the dome. Telescope tracking is turned off and the flat-field illumination source is switched on.

5.18.1.2 HOWFS calibration templates

- **VIRCAM_howfs_cal_reset.tsf** — This template makes several reset (aka BIAS) exposures with the detector windowed to match a variety of different HOWFS filter/position combinations. The DARK filter is selected.

- **VIRCAM_howfs_cal_dark.tsf** — This template makes several DARK exposures at a variety of different HOWFS filter/position combinations. The DARK filter is selected.

- **VIRCAM_howfs_cal_domeflat.tsf** — This template makes a flat-field exposure (or series of exposures) at a variety of different HOWFS filter/position combinations (assuming that the telescope is already pointing at the dome screen with the calibration source turned on).

5.18.1.3 HOWFS observation templates

- **VIRCAM_howfs_obs_exp.tsf** — This template makes a HOWFS wavefront measurement and saves the result to a coefficients file.

- **VIRCAM_howfs_obs_wfront.tsf** — This template makes a HOWFS wavefront measurement and saves the result to a coefficients file. In addition, a wavefront correction is calculated and sent to the VISTA TCS. (See the flow chart in Figure 33 on page 75.)

5.18.2 Imaging Templates

5.18.2.1 Acquisition templates

VIRCAM_img_acq.tsf — Acquire a science target. This template sets the instrument into IMAGING mode and selects a science filter. It also points the telescope to a new science target.

- **VIRCAM_img_acq_twilight.tsf** — This template selects a twilight sky field. It sets the instrument into IMAGING mode, selects a science filter and points the telescope to best twilight sky available (as read from the twilight sky database file $INS_ROOT/SYSTEM/MISC/VISTA/VIRCAM_twilight.paf).

- **VIRCAM_img_acq_domescreen.tsf** — This template sets the instrument into IMAGING mode and selects a science filter. It also moves the telescope to point at the flat-field screen in the dome. Telescope tracking is turned off and the flat-field illumination source is switched on

5.18.2.2 Calibration templates

- **VIRCAM_img_cal_reset.tsf** — This template makes one or more reset (aka BIAS) exposures with the DARK filter selected.

- **VIRCAM_img_cal_dark.tsf** — This template makes one or more DARK exposures at the specified exposure time with the DARK filter selected.

- **VIRCAM_img_cal_darkcurrent.tsf** — This template makes a series of DARK exposures (as in VIRCAM_img_cal_dark.tsf) but at a variety of different exposure times. The resulting data can be used to calibrate the dark current of the detector.

- **VIRCAM_img_cal_domeflat.tsf** — This template makes one or more flat-field exposures (assuming that the telescope is already pointing at the dome screen with the calibration source turned on)

- **VIRCAM_img_cal_linearity.tsf** — This template makes a series of flat-field exposures (as in VIRCAM_img_cal_domeflat.tsf) but at a variety of different exposure times, with corresponding darks. The resulting data can be used to determine the linearity of the detector response.

- **VIRCAM_img_cal_noisegain.tsf** — This template makes a series of DARK exposures followed by the same number of flat-field exposures, with corresponding integration times. The resulting data can be used to make a measurement of the detector readout noise and gain.

- **VIRCAM_img_cal_twiflat.tsf** — This template waits until the sky has reached a suitable brightness and then makes a series of twilight sky. It assumes the telescope is already pointing at the twilight sky.

- **VIRCAM_img_cal_persistence.tsf** — This template makes one exposure (of a bright source) with a selected science filter, followed by a series of DARK exposures. All exposures have the same integration time and number of integrations. The resulting sequence of exposures can be used to measure the image persistence.

- **VIRCAM_img_cal_crosstalk.tsf** — This template makes a series of exposures, with each exposure offset from the previous one by a sequence of small steps designed to place the image of a bright star on each of the 16 readout sectors on each detector. The resultant series of exposures can be used to detect any cross-talk between detector readout sectors. The template assumes that a bright, nearly saturated star has already been acquired on the first sector of the first detector.

- **VIRCAM_img_cal_illumination.tsf** — This template makes a series of exposures, with each exposure offset from the previous one by a sequence of small steps designed to place bright star at a regular grid of offset positions across each detector. The resultant series of exposures can be used to calibrate the illumination correction caused by scattering within the camera. The template assumes that a sparse field of bright stars has already been acquired at the first position.

- **VIRCAM_img_cal_std.tsf** — This template makes one “pawprint” observation of a field of photometric standards. Its implementation is identical to

VIRCAM_img_obs_paw.tsf, described below, except for the template name and "DPR TYPE" header keyword that end up in the resulting data.

5.18.2.3 Observation templates

VIRCAM_img_obs_exp.tsf — This template makes a series of exposures at the same target position with a single science filter. This simple template with no jittering or microstepping, useful for making a test or quick-look observation.

- **VIRCAM_img_obs_paw.tsf** — This template makes one "pawprint" observation (see section 2.1.4 on page 32 and [RD14]) using a selection of filter changes, jittering and microstep movements. It is assumed the telescope has already been positioned to the first target and is already locked onto guide stars.

  NOTE: The main difference between a pawprint template and a tile template containing a single pawprint is in the way the guide stars are processed. The pawprint template does not define any guide stars, since these are assumed already to have been set up by the acquisition template. The tile template with a single pawprint can have guide stars pre-chosen by the Survey Area Definition Tool.

- **VIRCAM_img_obs_tile<N>.tsf** — This template makes sufficient observations to generate a contiguous "tile", using a selection of pawprints, filter changes, jittering and microstep movements (see section 2.1.4). It is assumed the telescope has already been pointed to the first target. *This template is the one that will be used most frequently to build VISTA surveys.*

  NOTE: The VIRCAM_img_obs_tile template uses parameter (PAF) files to communicate the guide star information for each pawprint in the tile. These parameter files are communicated to the template through parameters named "SEQ.REF.FILEi", which are of type "paramfile". Since the ESO P2PP and BOB software requires that all "paramfile" keywords have a value, a different variation of the tile template signature file is provided for each possible number of pawprints (e.g. VIRCAM_img_obs_tile1, VIRCAM_img_obs_tile3 and VIRCAM_img_obs_tile6). Furthermore, when a tile template is imported directly into BOB for test purposes (without being generated by P2PP), the blank "paramfile" keywords can cause BOB to issue "missing PAF parameter" warnings. These warnings do not prevent the templates from running and being tested.

- **VIRCAM_img_obs_offsets.tsf** — This template makes a series of observations using a list of user-defined telescope offsets (suitable for making a one-off observation not covered by the pre-defined tile and jitter patterns). It is assumed the telescope has already been pointed to the first target.

6 Configuration

Instrument configuration parameters should be changed explicitly only by engineers responsible for the instrument maintenance. It is possible for an expert operator to modify some of these settings (for example if the filter wheel in-position switches stop working) using the "expert startup" panel described in section 4.4 on page 60.

6.1 Tabular Overview of Files

Here is a tabular overview of the location, purpose and access of all the files used by the VISTA IR Camera software.

File	Location	How accessed	Description
VIRCAM.isf	$INS_ROOT/SYSTEM /COMMON/CONFIGFILES $INTROOT/config	Read by P2PP and the VISTA Survey Area Definition Tool (SADT), and used by tpltoo to build OBs.	Instrument summary file
VIRCAM_tec.isf	$INS_ROOT/SYSTEM /COMMON/CONFIGFILES $INTROOT/config	Used by tpltoo to build OBs.	Instrument summary file — engineering version
VIRCAM_Tile*.paf	$INS_ROOT/SYSTEM /MISC/VISTA	Read by SDT and tile and pawprint templates.	Tile pattern descriptions
VIRCAM_Jitter*.paf	$INS_ROOT/SYSTEM /MISC/VISTA	Read by SDT and pawprint templates.	Jitter pattern descriptions
VIRCAM_Ustep*.paf	$INS_ROOT /SYSTEM/MISC/VISTA	Read by SDT and pawprint templates.	Microstep pattern descriptions
VIRCAM_twilight.paf	$INS_ROOT/SYSTEM /MISC/VISTA	Read by twilight sky template.	Twilight sky database
VIRCAM_HOWFS_*.paf	$INS_ROOT/SYSTEM /MISC/VISTA	Read by HOWFS templates.	HOWFS position database
VIRCAM.zip	$INTROOT/config	Read by P2PP and SDT.	Instrument summary package
vcmcfgCONFIG.cfg	$INS_ROOT/SYSTEM /COMMON/CONFIGFILES	Read by ctoo.	Main instrument configuration file.
vcmcfgINS.cfg	$INS_ROOT/SYSTEM /COMMON/CONFIGFILES	Read by ctoo.	Instrument configuration file.
vcmcfgINS_TEST.cfg			Instrument test configuration
vcmcfgICS_sensors.cfg			Instrument sensor configuration
vcmcfgICS_filters.cfg			Instrument filter configuration
vcmcfgICS_thermal.cfg			Instrument thermal control configuration
vcmcfgIRDCS.cfg	$INS_ROOT/SYSTEM /COMMON/CONFIGFILES	Read by IRACE software.	IRACE configuration file
vcmcfgOS.cfg	$INS_ROOT/SYSTEM /COMMON/CONFIGFILES	Read by ctoo.	Observation Software configuration file.

vcmcfgSTART.cfg	$INS_ROOT/SYSTEM /COMMON/CONFIGFILES	Read by ctoo and stoo.	Instrument startup configuration file.
VIRCAM_*.ref	$INS_ROOT/SYSTEM /COMMON/SETUPFILES/REF	Read by template sequencer scripts.	Reference setup files
VIRCAM_star*.ref	INS_ROOT/SYSTEM /COMMON/SETUPFILES/REF	Read by tile templates that require new guide stars.	Guide star reference setup files.
VIRCAM_*.tsf	$INS_ROOT/SYSTEM /COMMON/TEMPLATES/TSF	Read by P2PP and bob. Used by tpltoo to generate OBs.	Template signature files
VIRCAM_*.seq	$INS_ROOT/SYSTEM/ COMMON/TEMPLATES/SEQ	Executed by bob.	Sequencer scripts
VIRCAM_*.obd	$INS_ROOT/SYSTEM/ COMMON/TEMPLATES/SEQ	Written by P2PP or tpltoo. Read by bob.	Observation blocks
ESO-VLT-DIC.VIRCAM_*	$INS_ROOT/SYSTEM/ COMMON/Dictionary	Read by all software.	Dictionaries
lvcics1.scan	$INTROOT/config	Read by scan system during software build.	Scan link definitions
vcins-misc-all.env	$INTROOT/config	Invoked by osbEnvSet on system startup.	Bash/pecs environment variables.
vcins.bobrc	$INTROOT/config	Copied to home directory and read by bob.	Bob startup variables.
vc*.CDT	$INTROOT/CDT	Read by message system before sending commands.	Command definition tables

6.2 Changing instrument configuration parameters

The configuration of the running VIRCAM software is contained in the files stored in $INS_ROOT (as listed above). Most of the configuration files are created by the “vcmcfg” module. The files in $INS_ROOT are overwritten whenever the “vcmcfg” module is built. The IRACE configuration files are created by the “vcdcfg” module.

6.2.1 Temporary changes to instrument configuration parameters

The instrument configuration can be changed temporarily by editing the files stored in the $INS_ROOT/SYTEM/CONFIGFILES directory, or by using one of the configuration change panels supplied with the VIRCAM software: vcinsFilterConfigDev, vcinsThermalConfigDev or vcinsWcsConfigDev. Individual keywords can be queried or set with the commands (see [RD50]):

```
% ctooConfigGet VIRCAM <keyword>
% ctooConfigSet VIRCAM <keyword> <value>
```

If any LCU parameters are changed, the command

```
% icbConfigSet VIRCAM
```

ensures the ICS configuration file is regenerated (the configuration change panels will do this automatically). The new configuration parameters will be used by the VIRCAM software the next time it is reset with an OFF, ONLINE command sequence.

Any changes made in this way are temporary. The configuration settings will return to their default values the next time the "vcmcfg" module is rebuilt, like this

```
% cmmCopy vcmcfg
% cd vcmcfg/src
% make clean all man install
% icbConfigSet VIRCAM
```

In fact, this sequence of commands *must* be done to maintain proper configuration control if the temporary configuration changes are rejected or no longer required (see [AD3]).

6.2.2 A note about LCU device simulation

Some LCU configuration parameters, such as the simulation mode or availability of individual devices, will only change when the LCU is rebooted. (This can cause a problem is an attempt is made to restart the software after using the expert startup panel to change the simulation mode of individual LCU devices). It is possible to avoid having to reboot the LCU by changing the simulation mode explicitly by selecting all the simulated devices on the ICS engineering panel (Figure 38 on page 87) and using the "Devices → Simulate HW" menu to put them explicitly into simulation mode. A useful procedure is:

- Bring up the expert startup panel:
  ```
  % vcinsStartupDev
  ```
  Alter the simulation modes of the LCU devices and press the "Save config" button. Exit from the panel.

- Bring up the ICS engineering panel:
  ```
  % vcinsStart -panel ICS
  ```
  Modify the simulation modes of the LCU devices to match what you did on the expert startup panel by selecting the check boxes belonging to those devices and using "Devices → Simulate HW". To verify what you did, try "ICS → ONLINE" and make sure all the devices go ONLINE in the new simulation mode.

- Start the VIRCAM software
  ```
  % vcinsStart
  ```

6.2.3 Permanent changes to the instrument configuration parameters

Permanent changes can only be made to instrument configuration parameters by changing the files in the "vcmcfg" module and archiving it back to the CMM repository. This should *only* be done after the changes have been thoroughly tested. It is important to do this for good configuration management, and it is especially important to archive any changes made to the instrument configuration before reinstalling any new version of the operating system or common software on the instrument workstation.

The current configuration files stored in $INS_ROOT can automatically be archived to the vcmcfg module (after thorough testing) with the command:

```
% ctooConfigArchive VIRCAM
```

which sequences the "cmmModify" and "cmmArchive" commands automatically (see [RD37]). However, if you are familiar with the CMM utility it is preferable to run the commands manually and see exactly which files are being updated. A useful sequence is:

1. Rename or remove any existing copy of the vcmcfg module, then
   ```
   % cmmModify vcmcfg
   ```
2. Copy the relevant files (editing to record the changes in their history):
   ```
   % cp -i $INS_ROOT/SYSTEM/CONFIGFILES/*.cfg vcmcfg/config
   ```
3. Edit "vcmcfg/ChangeLog" and document the changes.
4. Check which files have changed:
   ```
   % cmmCheckForArchive vcmcfg
   ```
5. Finally, archive the module with a comment:
   ```
   % cmmArchive vcmcfg "<comment>"
   ```

As a final test, make a backup of the files within the $INS_ROOT/SYSTEM/CONFIGFILES directory, then retrieve the updated vcmcfg module from the CMM repository and rebuild it:

```
% cmmCopy vcmcfg
% cd vcmcfg/src
% make clean all man install
% icbConfigSet VIRCAM
```

The new instrument configuration should include all the changes you made.

6.3 VIRCAM configuration keywords

The VIRCAM subsystems are configured using standard BOSS, Base ICS and IRACE configuration keywords. The following sections describe the additional special keywords used by VIRCAM. Any of these keywords can be queried or set with the ctooConfigGet and ctooConfigSet commands, using the syntax shown in section 6.2.1.

6.3.1 Lakeshore and Pfeiffer sensor device keywords

Although they are programmed as special devices, the Lakeshore 218, Lakeshore 332 and Pfeiffer TPG256 devices are configured using exactly the same keywords as a standard "Base ICS" sensor device, [RD46].

6.3.2 Temperature control keywords

The following configuration keywords are used to configure the thermal control software. Most of them can be configured through the vcinsThermalConfigDev panel.

6.3.2.1 Basic thermal control keywords

These keywords define the basic properties of the thermal control devices and rarely need to be changed.

Keyword	Type	Description
INS.THERMAL.ENABLE	Boolean	Enable thermal control loops. Normally T, but setting this to F disables the high-level thermal control software and allows manual control of the Lakeshore devices.
INS.THERMAL.DBROOT	string	Database root point for the temperature sensing devices.
INS.THERMAL.DET.SENSOR	string	Name of sensor device for detector temperatures.
INS.THERMAL.FPA.SENSOR	string	Name of sensor device for FPA temperatures.
INS.THERMAL.CRY.SENSOR	string	Name of sensor device for ambient/cryostat temperatures.
INS.THERMAL.CLD.SENSOR	string	Name of sensor device for cold head temperatures.
INS.THERMAL.MIN	double	Minimum sensible temperature in K
INS.THERMAL.MAX	double	Maximum sensible temperature in K

6.3.2.2 State machine keywords

The following keywords control the state transitions of the thermal control software. The software begins in the AMBIENT state, with transitions to the other states governed by changes to the focal plane array and cold head temperatures, as defined by these keywords (the state transitions are shown in Figure 3 on page 30):

Keyword	Type	Description
INS.THERMAL.CLD.USEMEAN	Boolean	If T, use the mean of the cold head temperatures. If F, use the minimum temperature (in case one or more devices are not switched on).

INS.THERMAL.CLD.AMBCOOL	double	Max cold head temp (wrt ambient) for AMBIENT --> COOLDOWN
INS.THERMAL.CLD.COOLAMB	double	Min cold head temp (wrt ambient) for COOLDOWN --> AMBIENT
INS.THERMAL.CLD.WARMAMB	double	Min cold head temp (wrt ambient) for WARMUP --> AMBIENT
INS.THERMAL.CLD.DELTA	double	Cold head change for WARMUP <--> COOLDOWN
INS.THERMAL.FPA.AMB	double	Min FPA temp (wrt ambient) for --> AMBIENT
INS.THERMAL.FPA.OPER	double	Max FPA temp for --> OPERATIONAL

6.3.2.3 Detector temperature control keywords.

The following keywords configure the detector temperature control loop: DET.TARGET should be set to the final desired target temperature for the detectors (in K). FPA.MXGRD is the maximum allowed rate of change of detector temperature (in K per hour). The high level thermal control software will maintain FPA thermal plate temperature changes within this limit. A RAMP setting might be defined on the Lakeshore 332 controller (LSC1) as well. FPA.MXDIFF is the maximum allowed temperature difference between the detectors and the FPA thermal plate, which is required to minimise the spatial thermal gradient across the focal plane; this limit is relaxed slightly during WARMUP mode where the thermal gradient is less important than keeping the detectors warm. DET.DB is the detector deadband (in K); the thermal loop will normally not update the FPA target until the detector temperature has changed by this amount (which prevents the LCU being flooded with unnecessary updates).

Keyword	**Type**	**Description**
INS.THERMAL.DET.TARGET	double	Detector target temperature in K. This is the temperature for optimal performance of the science detectors.
INS.THERMAL.WFS.TARGET	double	WFS plate target temperature in K. *Not used, since the WFS temperature is controlled by the TCCD controller. Leave set to 0.0.*
INS.THERMAL.FPA.MXGRD	double	Maximum allowed detector temperature gradient in K per hour.
INS.THERMAL.FPA.MXDIFF	double	Maximum detector to FPA temperature difference in K.
INS.THERMAL.FPA.DAMPING	double	Damping factor applied to FPA temperature setting changes (must be less than 1).
INS.THERMAL.OB.MXDIFF	double	Maximum detector to optical bench top temperature difference in K.
INS.THERMAL.DET.DB	double	Detector temperature change deadband in K.

6.3.2.4 Cryostat temperature control keywords

The following keywords configure the cryostat temperature control loop. WIN.DELTA should be set to the desired cryostat window temperature with respect to ambient (e.g. 0.5 means "keep window 0.5 K above ambient"). TUB.DELTA should be set to the desired cryostat tube temperature with respect to ambient (e.g. 0.3 means "keep tube 0.3 K above ambient"). AMB.DB is the ambient deadband (in K); the thermal loop will not update the cryostat targets until the ambient temperature has changed by this amount (which prevents the LCU being flooded with unnecessary updates). The RLYOFFi keywords control the points at which the relays controlling the cryostat heaters switch on: Heater i switches on when cryostat temperature < ambient + delta + RLYOFFi.

Keyword	**Type**	**Description**
INS.THERMAL.WIN.DELTA	double	Cryostat window target temperature with respect to ambient in K.
INS.THERMAL.TUB.DELTA	double	Cryostat tube target temperature with respect to ambient in K.
INS.THERMAL.AMB.DB	double	Ambient temperature change deadband in K.
INS.THERMAL.RLYOFF1	double	Temperature offset for LS218S relay 1
INS.THERMAL.RLYOFF2	double	Temperature offset for LS218S relay 2
INS.THERMAL.RLYOFF3	double	Temperature offset for LS218S relay 3

6.3.3 Filter wheel configuration keywords

The filter wheel is configured in a very similar way to a standard “Base ICS” filter wheel, [RD46]. It is mainly configured using the standard “Base ICS” keywords, but there are some additional special keywords for defining the VIRCAM-specific aspects of the filter wheel, such as the use of the in-position switches. Most of the keywords can be configured using the vcinsFilterConfigDev panel.

6.3.3.1 Standard filter wheel keywords

Keyword	**Type**	**Description**
INS.FILT1.DEVNAME	string	Short name of the filter device (“filt”).
INS.FILT1.DEVDESC	string	Description of filter device (“Filter wheel”)
INS.FILT1.LCUID	integer	ID of LCU controlling device (always 1).
INS.FILT1.SWSIM	boolean	T if the device is software simulated.
INS.FILT1. INITTOUT	integer	Initialisation timeout in milliseconds.
INS.FILT1.TWOSTEP	boolean	T if the device uses two step movement (for backlash compensation). *Must be T for correct operation*.
INS.FILT1.POSNUM	integer	Number of named filter positions defined.
For each filter position <n>, where <n> = 1…POSNUM		
INS.FILT1.POSID<n>	string	Name of filter wheel slot at position <n>. VIRCAM uses a naming convention in

		which science filter position names begin with "SLOT" and intermediate position names begin with "INT".
INS.FILT1.ID<n>	string	ID of filter wheel slot at position <n>. In the VIRCAM software, the ID keywords must also be set to the names of the slots (because the INS.FILTER assembly uses "INS.FILT1.ID<n>" setup keywords). The true ID of a particular filter tray is specified by the TRAYID keyword (see below).
INS.FILT1.NAME<n>	string	Name of filter at position <n>, for example "J". VIRCAM uses a naming convention such that filters whose names begin with "DARK" and "SUNBLIND" are dark filters.

6.3.3.2 VIRCAM special filter wheel keywords

Keyword	**Type**	**Description**
INS.FILT1.REFSW	string	Reference switch currently being used (PRIMARY or SECONDARY). Normally PRIMARY but may be switched to SECONDARY if the primary reference switch breaks. *NOTE: If this parameter is changed the "PRIMARY/SECONDARY" selector switch on the back of the LCU must also be changed.*
INS.FILT1.ALIGNP	integer	Offset applied to all filters positions when the wheel is used with the PRIMARY reference switch [enc].
INS.FILT1.ALIGNS	integer	Offset applied to all filters positions when the wheel is used with the SECONDARY reference switch [enc].
INS.FILT1.INOFF	integer	In-position switch test 2 stage movement offset [enc]. This must be much larger than the width of an in-position switch.
INS.FILT1.USESW	boolean	If T, the in-position switch is checked after arriving at a science filter slot. The switch must be ACTIVE or the test fails. May be set to F to continue operations if the in-position switch breaks.
INS.FILT1.INTSW	boolean	If T, the filter wheel is also stopped at an intermediate position (determined by the INS.FILT1.INOFF value) and the in-position switch checked. The switch must be INACTIVE at this intermediate position or

		the test fails. Setting this parameter to T reduces survey efficiency but improves in-position switch error detection. This parameter is normally F unless a switch fault is suspected.
For each filter position <n>, where <n> = 1…POSNUM		
INS.FILT1.TRAYID<n>	string	Unique identifier for a particular filter tray installed at position <n>, for example "ESO-J-0001" (as opposed to the filter name, which just specifies the wavelength class of a filter). The unique identifier changes any time there is a significant change to a filter tray.
INS.FILT1.TRANS<n>	integer	Transmission of filter at position <n> (1=dark, 2=medium, 3=bright.
INS.FILT1.FOCUS<n>	double	Telescope focus offset needed by filter at position <n> in mm.
INS.FILT1.WLEN<n>	double	Effective wavelength of filter at position <n> in nanometres. Zero is used for filters with no preferred wavelength, such as a DARK filter.

6.3.4 Heart beat device configuration keywords

The special heart beat device is configured with the following keywords. There is no reason to change any of them unless the VME crate is rewired.

Keyword	Type	Description
INS.HB1.DEVNAME	string	Short name of the device ("hb1").
INS.HB1.DEVDESC	string	Description of device ("Heart Beat")
INS.HB1.LCUID	integer	ID of LCU controlling device (always 1).
INS.HB1.SWSIM	boolean	T if the device is software simulated (always F).
INS.HB1.FREQUENCY	double	Square wave frequency in Hz
INS.HB1.PIN	integer	Output pin on the acromag digital I/O device (0-63).

6.3.5 World Coordinates configuration keywords

The World Coordinates keywords can be configured using the vcinsWcsConfigDev panel. They are used to generate keywords in the FITS header. It is assumed the VISTA data flow pipeline will be able to use this information to identify astrometric standards and refine the World Coordinates.

The following keywords configure the offset of each detector chip with respect to the reference point on the focal plane, measured in terms of the coordinate of the reference point, in pixels, from each detector's origin (see the coordinate system in Figure 6 on page 33).

Keyword	**Type**	**Description**
For each detector chip <i>, where <i> = 1…16		
OCS.WCS.CHIP<i>.CRPIX1	double	X coordinate of reference point from chip i
OCS.WCS.CHIP<i>.CRPIX2	double	Y coordinate of reference point from chip i

The following keywords configure the transformation between pixel coordinate and focal plane coordinate on the sky for each detector. The skew of each chip is assumed to be negligible, but the detectors can have slightly different plate scales and orientations.

Keyword	**Type**	**Description**
For each detector chip <i>, where <i> = 1…16		
OCS.WCS.CHIP<i>.SCALE1	double	Magnification of chip i X axis from normal plate scale.
OCS.WCS.CHIP<i>.SCALE2	double	Magnification of chip i Y axis from normal plate scale.
OCS.WCS.CHIP1.ANGLE	double	Angular offset of chip 1 from perfect alignment

The following keywords configure the transformation between sky coordinate, projected onto the focal plane, and focal plane coordinate. The VISTA IR Camera software uses two alternative projections:

- Pixel coordinates, in which PV = (1,0,0,0,0) for all chips. This is used for flat-fields and other calibrations where sky coordinates have no meaning. In this case the PV matrix is omitted from the data.

- A ZPN tangent plane projection, in which the PV matrix represents the radial polynomial. The VISTA optics will have a dominant third order term. These are the coefficients defined below.

Keyword	**Type**	**Description**
For each detector chip <i>, where <i> = 1…16		
OCS.WCS.CHIP<i>.PV1	double	First order ZPN coefficient for chip i
OCS.WCS.CHIP<i>.PV2	double	Second order ZPN coefficient for chip i
OCS.WCS.CHIP<i>.PV3	double	Third order ZPN coefficient for chip i
OCS.WCS.CHIP<i>.PV4	double	Fourth order ZPN coefficient for chip i
OCS.WCS.CHIP<i>.PV5	double	Fifth order ZPN coefficient for chip i

6.4 Lakeshore and Pfeiffer Sensor Device Initialisation

As mentioned earlier, the Lakeshore and Pfeiffer devices are configured in exactly the same way as any standard "Base ICS" sensor device, [RD46]. The sensor device configuration is defined in the configuration file "vcmcfg/config/vcmcfgINS_sensors.cfg". Of special note are the INS.SENSORi.INITSj keywords, which contain the device initialisation strings. These strings contain the commands which are sent to the Lakeshore or Pfeiffer devices on initialisation (i.e. whenever the device state changes from OFF to ONLINE). The recognised device commands are listed in section 8.9 on page 153.

7 TESTING

The VIRCAM software provides several levels of testing utilities.

7.1 Minimal Instrument Self Test

The VIRCAM software will check the communication with and run a minor self-test on every instrument device when the software state changes from OFF to ONLINE. Therefore, a minor self test can be carried out (and should be carried out at least daily) by sending the command sequence OFF, ONLINE (after the software has been started with vcinsStart). The sensor devices self-test and switch to the ONLINE automatically after an LCU reboot. The instrument will not go into the ONLINE state if a problem is detected in any device.

7.2 Major Software Installation Test

All the VIRCAM subsystem software modules have a "test" subdirectory which contains facilities for self-testing the software contained within that module and all its dependent modules. These major self-test utilities can be used to check out instrument components in great detail. The complete software installation can be tested with the command:

```
% pkginTest vcins [-nobuild] –frommodule dicVIRCAM
```

The "-frommodule" option will skip the tests on any VLT software modules referenced by the vcins module. Use the "-nobuild" option if you have defined temporary configuration parameters (section 6.2.1) and don't want them overwritten when the vcmcfg module is rebuilt.

The pgkinTest command, [RD43], invokes the "Tools for Automated Testing" (TAT) utility, [RD36], and automatically tests every software module. An individual software module can be tested by invoking the command "tat" or "make test" from within its "test" subdirectory. pgkinTest also has the side effect of running some of the tests twice (e.g. from vci/src and vci/test). It generates very little output while running, since the output is interpreted by the TAT utility. NOTE: The "tat" facility needs a reference file to compare its results with – see [RD36].

This particular test is designed to check out the software installation before a major new release.

7.3 Major Instrument Self Test

The instrument subsystems (ICS, DCS, HOWFS and OS) can be self-tested more quickly and with more detailed output (than with the pkginTest described above) using the command:

```
% vcinsSelfTest
```

Care should be taken not to run the ICS tests too frequently, as the tests exercise the filter wheel more frequently than ordinary science observations and may reduce its operational lifetime.

7.3.1 Survey Observation Soak Test

The entire VISTA system (instrument subsystems plus telescope system and data acquisition system) can be soak-tested with the command:

```
% vcmseqSoakTest <nrepeats>
```

where <nrepeats> is the number of repeats required. The command runs a script which acquires <nrepeats> targets randomly positioned on the sky and makes a tile observation at each target. The resulting survey is extremely inefficient but exercises the camera to telescope interface as much as possible. Since all the targets are random, guide stars cannot be chosen in advance. The advance specification of guide stars can be tested with the VIRCAM_gen_tec_AcqTest.obd test Observation Block, described below.

7.3.2 Test Observation Blocks

The VIRCAM software provides the following test Observation Blocks, which are generated when the vcoseq and vcmseq modules are built. These Observation Blocks may be loaded into BOB and executed to test various aspects of the instrument system:

- **VIRCAM_gen_tec_SelfTest.obd** — This is the most important self-test OB. It checks the execution of all the non-technical templates defined for the instrument. It is very useful to run this test before making a new release of the VIRCAM software, or after installing a new version of the VLT common software.

  NOTE 1: The VIRCAM_img_obs_offsets, VIRCAM_img_obs_paw and VIRCAM_img_obs_tile templates are only tested with one instance of their parameters. Since those particular templates have a very large possible combination of parameters, they have their own self-test templates, described below.

  NOTE 2: The VIRCAM_howfs_obs_exp and VIRCAM_howfs_obs_wfront templates, which are executed by this self-test OB, will fail if the HOWFS image analysis process is given simulated data. This can be prevented by defining the "TAT_TEST" environment variable. This variable will also prevent the need to answer interactive prompts during the test.

- **VIRCAM_gen_tec_AcqTest.obd** — This OB tests the acquisition of a target, with the VIRCAM_img_acq.tsf template, using every common combination of VISTA TCS parameters (i.e. with or without guiding, with or without confirmation of the guide star, with and without active optics, etc…).

- **VIRCAM_gen_tec_CheckFilters.obd** — This OB tests that all the filters specified in a list can be selected, either in the order given or in random order. The mean filter exchange time is recorded.

- **VIRCAM_gen_tec_OffsetsTest.obd** — This OB executes the VIRCAM_img_obs_offsets.tsf template with a large number of common combinations of parameters.

- **VIRCAM_gen_tec_PawprintTest.obd** — This OB executes the VIRCAM_img_obs_paw.tsf template with a large number of common combinations of parameters.

- **VIRCAM_gen_tec_TileTest.obd** — This OB executes the VIRCAM_img_obs_tile<N>.tsf template with a large number of common combinations of parameters. Since the tile template is very important for survey operations, it is recommended that this test be run in addition to VIRCAM_gen_tec_SelfTest.obd.

  NOTE: The VIRCAM_img_obs_tile template uses parameter (PAF) files to communicate the guide star information for each pawprint in the tile. These parameter files are communicated to the template through parameters named "SEQ.REF.FILEi", which are of type "paramfile". When the tile template is included within a test OB, some of the "paramfile" keywords contain values which can cause BOB to issue "missing PAF parameter" warnings. These warnings do not prevent the templates from being tested.

- **VIRCAM_gen_tec_exp.obd** — This OB tests the data acquisition performance. A series of exposures (which could be tens or thousands) are made at the same instrument configuration, and the mean execution time per exposure is calculated and displayed. The template also tests the high speed interaction between the OS and DCS systems.

- **VIRCAM_howfs_tec_loopback.obd** — This OB tests the operation of the HOWFS image analysis software by generating and then analysing loopback data.

7.4 Individual Subsystem Self Test

7.4.1 ICS tests

The command

```
% vciTest
```

will run a complete self-test on the ICS subsystem. The filter wheel will be moved to every possible position, the in-position switch states will be verified, and the Lakeshore devices will be sent some example update commands.

7.4.2 DCS tests

The command

```
% vcdTest
```

will run a complete self test on the IRACE DCS software, as configured for VIRCAM. The DCS will be instructed to make exposures in all possible detector modes. The "VIRCAM_gen_tec_exp.obd test OB (described above) can also be used to test the DCS to OS interaction.

7.4.3 HOWFS tests

The command

```
% vchTest
```

will run a self-test on the HOWFS image analysis software. It will be requested to analyse some loopback data made from known wavefront aberrations. The test can be supplemented by the command

```
% vchSoakTest <nrepeats> <nabort>
```

which will run a soak test on the HOWFS image analysis software. The script will generate <nrepeats> sets of random test data, which the HOWFS software will be instructed to analyse. The process will be aborted every <nabort> times around the cycle. The test is designed to ensure the HOWFS software contains no time-dependent bugs or memory leaks.

7.4.4 OS tests

The command

```
% vcoTest
```

will test the VIRCAM OS by making a series of exposures, merging the data and checking the integrity of the FITS header. However, the "VIRCAM_gen_tec_exp.obd test OB (described above) actually runs a more comprehensive test. The command

```
% voseqTest [options]
```

will automatically start BOB and run one or more of the following test templates, depending on the options provided:

Option	Result	
-a0 (default)	Run	acquisition test,VIRCAM_gen_tec_AcqTest.obd
-a1	Do not run	
-s0	Run	self-test VIRCAM_gen_tec_SelfTest.obd
-s1 (default)	Do not run	
-t0	Run	tile test, VIRCAM_gen_tec_TileTest.obd
-t1 (default)	Do not run	
-p0 (default)	Run	pawprint test, VIRCAM_gen_tec_PawprintTest.obd
-p1	Do not run	
-o0 (default)	Run	offsets test, VIRCAM_gen_tec_OffsetsTest.obd
-o1	Do not run	

If no options are provided, the command runs the self-test and tile test templates. It also defines the “TAT_TEST” environment variable to prevent interactive prompts and prevent the HOWFS templates having problems with simulated data.

7.5 Filter Wheel Test and Diagnostic Scripts

There are several filter wheel test scripts that are designed to test the filter wheel mechanisms reliability and repeatability for positional accuracy. All test scripts are written in BASH shell programming language and must be called from a Linux command prompt. All test scripts write log files during their execution which are written in a comma separated value (.csv) file format that can be loaded into most spreadsheet applications. An overview of the test scripts is given in sections 7.5.1 onwards.

The following command must be used (after cd’ing to the top level source directory) before executing any of the filter wheel test scripts:

```
% vciFilterTestsSetup
```

An LCU reboot is recommended after executing this command. This command disables the two step movement normally used to remove backlash (by setting the INS.FILT1.TWOSTEP keyword to “F”). The test scripts use single step movements that do not work properly[17] when two step movement is enabled. The scripts are also designed so that backlash is not a factor in the results and will be cancelled out by the sequence of moves. The above command also defines the filter wheel motor as a linear device rather than a circular device, which gives the test scripts full control over the direction of movement.

After the testing with the filter wheel test scripts has finished the command

```
% vciFilterTestsRestore
```

[17] The scripts will work, but the extra movements (e.g. 500 steps to the right followed by 501 steps to the left) will produce jerky movements and a very large time penalty.

must be used (after cd'ing to the top level source directory) to re-enable the two step movement and restore the filter wheel's original settings. An LCU reboot is recommended after executing this command. Alternatively, the old configuration settings can be restored after an engineering session by retrieving and rebuilding a fresh copy of the vcmcfg module from the CMM repository (see section 9.1.2 on page 162). This will take longer but may be worth it to guarantee the configuration settings are restored. *Don't forget to restore the filter wheel to its proper settings and reboot the LCU before restarting normal operations. Failure to do this will result in filter wheel errors.*

7.5.1 Finding the Reference Position

The purpose of this script is to test the repeatability of the filter wheel while making several revolutions. It should find the edge of the reference switch in the same position after each revolution.

The script repeatedly searches for the reference (datum) position. It initialises by locating the edge of the reference switch and then moves just short of one full revolution and then performs a single step search for the same edge of the reference switch. The number of reference position searches and the direction they are carried out (clockwise or counter clockwise) are available as command line options. The step position at which the reference is found is written to the log file *reference_positions.csv*. An example of the script invocation is given below with command lines options.

```
% vciFindReference 1 10
```

The first command line argument determines the direction of the reference search, '0' for clockwise and '1' for counter clockwise. The second command line argument is the total number of times to perform the reference position search. The example given above would result in 10 reference position searches in the counter clockwise direction.

As an example, the log file output might look like this:

```
Test script vciFindReference results executed on …
CCW search
Sequence,Reference(Absolute),Reference(Offset)
1,-210000 ,0
2,-419998 ,2
3,-629998 ,2
4,-839998 ,2
5,-1049998,2
Test script vciFindReference results completed on …
```

In this case there were 5 iterations in a clockwise direction. The 3 numbers reported by the script are the iteration count, the motor step position where the edge of the reference switch has been detected, and the value of that location modulo 210000 (the number of steps per rotation). In the above test there was a slippage of 2 steps during the first revolution and then no further slippage. (Slippage can be caused by a variation in the switch mechanism as well as by actual steps lost by the motor, so anything within ~4 steps is within the measurement error. A systematic trend after several revolutions would indicate actual step loss by the motor.)

7.5.2 Counting steps to the reference position

The purpose of this script is to test for any slippage of the filter wheel when stationary. The filter wheel should be initialised and then moved to a location just short of the reference switch, for example like this:

```
% msgSend "" vciControl OFF ""
% msgSend "" vciControl ONLINE ""
% msgSend "" vciControl SETUP "-function INS.FILT1.ENC -209980"
```

The filter wheel is then left stationary and a slippage test carried out (e.g. by moving the telescope rotator). When the test has finished, the position of the reference switch can be determined with the script

```
% vciStepsToReference
```

The script performs a single step search to find the edge of the reference switch and reports its position. If the filter wheel hasn't slipped, the position should be close to -210000 steps.

7.5.3 Finding the In-position Bearings

The purpose of this script is to verify that the bearings are in the correct positions (so the in-position switch is *ACTIVE* when a filter tray is in the beam). It can also be used to look for movements of the bearing as a function of temperature, and as another way of checking the repeatability.

The script finds the edges of all of the bearings that activate the in-position switch. There are eight bearings placed at each of the filter tray positions. The bearing search is performed in three sweeps; the first is a coarse search to roughly find the bearing edges to an accuracy of 10 steps; the second is a fine search to an accuracy of 1 step and the final sweep is a another fine search to test repeatability. The script can be run to find the bearings in either the clockwise or counter clockwise direction. The direction is selected by supplying a command line argument. An example of the script invocation is given below.

```
% vciFindBearings 0
```

The command line argument '0' specifies a search in the clockwise direction but '1' can be entered for the counter clockwise direction. The log file "*bearing_positions.csv*" is written whilst the test script is executing containing information about the bearing positions.

As an example, the log file output might look like this:

```
Test script /introot/vcam/bin/vciFindBearings results executed on …
CCW search
```

```
Coarse search results:
Bearing,Near Edge,Far Edge,Width,Distance
1,-24370 ,-26260 ,1890,24400
2,-50610 ,-52470 ,1860,24350
3,-76900 ,-78700 ,1800,24430
4,-103120 ,-105000 ,1880,24420
5,-129370 ,-131230 ,1860,24370
6,-155600 ,-157440 ,1840,24370
7,-181880 ,-183700 ,1820,24440
8,-208100 ,-209970 ,1870,24400
First fine search results:
Bearing,Near Edge,Far Edge,Width,Distance
1,-234368 ,-236254 ,1886,24398
2,-260602 ,-262462 ,1860,24348
3,-286896 ,-288708 ,1812,24434
4,-313114 ,-314990 ,1876,24406
5,-339370 ,-341226 ,1856,24380
6,-365596 ,-367438 ,1842,24370
7,-391874 ,-393692 ,1818,24436
8,-418094 ,-419970 ,1876,24402
Second fine search results:
Bearing,Near Edge,Far Edge,Width,Distance
1,-444370 ,-446252 ,1882,24400
2,-470602 ,-472462 ,1860,24350
3,-496896 ,-498704 ,1808,24434
4,-523114 ,-524992 ,1878,24410
5,-549370 ,-551224 ,1854,24378
6,-575600 ,-577438 ,1838,24376
7,-601874 ,-603692 ,1818,24436
8,-628094 ,-629970 ,1876,24402
Test script /introot/vcam/bin/vciFindBearings results completed on …
```

In the above log the script has homed in on the edge of the bearings, giving a more accurate reading with each iteration. The positions shown in subsequent iterations need to be processed modulo 210000 to be compared. The bearings should all have roughly the same width (~1850 steps) and be equidistant (~26250 steps – width). Filter locations should be well away from the edge of any bearing.

7.5.4 Finding the Reference Position after a Sequence of Direction Changes

There are two scripts available whose purpose is to check the repeatability of the filter wheel after making a sequence of moves that specifically involve a change of direction at known positions of the filter wheel.

The two scripts repeatedly search for the reference (datum) position after performing sequence of moves that involve changes of direction. Initially a datum is performed and then a sequence of direction changes are made by moving backwards and forwards to specific positions of the filter wheel before moving just short of the reference position and performing a single step search for the reference position. The number of direction changes is a command line option as well as the number of times to repeat the test sequence. The step position at which the reference is found is written to the log file “*direction_reversal.csv*”. The two scripts perform the reference search in a particular direction, which are naturally clockwise and counter clockwise. An example of the invocation of these scripts is given below with command line options.

```
% vciTestReversalCW 2 5
% vciTestReversalCCW 4 10
```

The first command line argument specifies the number of directional changes to be performed. The second command line argument is the total number of times to perform the test sequence. The first example given above would result in 5 reference position searches in a clockwise direction each of which includes 2 directional changes. The second example given above would result in 10 reference position searches in a counter clockwise direction each of which includes 4 directional changes.

7.5.5 Finding the Reference Position after a Sequence of User Specified Positions

There are two scripts available whose purpose is to check the repeatability of the filter wheel after making a sequence of moves to positions specified by the user.

The two scripts repeatedly search for the reference (datum) position after performing a sequence of moves to positions specified by the user. Initially a datum is performed and then a sequence of moves to user specified positions before moving just short of the reference position and performing a single step search for the reference position. The sequence of positions are specified as command line options as well as the number of times to repeat the test sequence. The step position at which the reference is found is written to the log file "*user_positions.csv*". Each of the two scripts available perform the reference search in a particular direction, which are naturally clockwise and counter clockwise. An example of the invocation of these scripts is given below with command line options.

```
% vciFindUserCW 5 140000 70000 140000
% vciFindUserCCW 10 50000 100000 150000
```

The first command line argument specifies the number of times to perform the reference switch search. The following command line arguments are the sequence of positions to move to during each reference position search. The first example given above would result in 5 reference position searches in a clockwise direction each of which includes a sequence of moves to 140,000, 70,000, 140,000 step positions. The second example given above would result in 10 reference position searches in a counter clockwise direction each of which includes a sequence of moves to 50,000, 100,000, 150,000 step positions.

7.5.6 Finding Reference Position after Several Random Moves

The purpose of this script is to check the repeatability of the filter wheel after making several random moves in both clockwise and counter clockwise directions.

The script repeatedly searches for the reference (datum) position after performing several random moves. It initially performs a datum and then moves to a number of random positions before moving just short of the reference position and performing a single step search for the reference position. The number of random moves and time the search cycle is carried out are available as command line options. The step position at which the reference is found is written to the log file "*random_positions.csv*". An example of the script invocation is given below with command line options.

```
% vciFindRandom 3 10
```

The first command line argument specifies the number of random positions that are to be moved to. The second command line argument is the total number of times to perform the reference position search. The example given above would result in 10 reference position searches each of which includes 3 random moves.

7.5.7 Finding the Backlash Measurement

The purpose of this script is to measure the backlash distance of the filter wheel.

The script repeatedly alternates between clockwise and counter clockwise direction searches for the reference position. It initially performs a datum and then moves just short of one full revolution in the clockwise direction and performs a single step search for the reference position. It then moves just short of one full revolution in the counter clockwise direction and performs yet another single step search for the reference position. The two reference position searches in both directions constitute one test cycle and given that one full revolution of the filter wheel is 210,000 steps it is possible to work out the backlash distance from the step positions written to the log file "*backlash_positions.csv*". An example of the script invocation is given below with command line options.

```
% vciFindBacklash 5
```

The command line argument specifies the number of backlash measurement test cycles to perform.

7.6 Cryostat Thermal Control Test Scripts

The thermal control software is provided with the following scripts, that may be used to test the control software when some or all the devices are in simulation mode. The scripts

```
% vciSimDevicesOnline
% vciSimDevicesOffline
```

simulate the sensor devices going into the ONLINE or the LOADED state. The script

```
% vciSimSetAmbient value
```

pokes the ambient temperature reading with a particular value (only possible if the ambient temperature sensor is being simulated). The script

```
% vciSimSetCoolers val1 val2 val3
```

pokes the three second stage cooler temperature readings with particular values (only possible if the cooler temperature sensors are being simulated). The script

```
% vciSimSetDetectors val1 val2 val3 val4 val5 val6 val7 val8
```

pokes the eight science detector temperature readings with particular values (only possible if the science detector temperature sensors are being simulated). The script

```
% vciSimSetFPA value
```

pokes the focal plane array temperature reading with a particular value (only possible if the FPA temperature sensor is being simulated). The most important script is

```
% vciThermalTestSim
```

which takes the thermal control software through a simulated cooldown, operational and warmup sequence. The script only works when all the sensors, or the entire LCU environment, are in simulation mode.

8 MAINTENANCE

8.1 Software Support for Camera Maintenance Procedures

8.1.1 Exchanging filters

The software stores filter information in two different places:

- A list of the names and properties of every filter which may possibly be installed in the instrument is listed in the instrument description file, vcotsf/config/VIRCAM.isf (there is also a copy in vcmtsf/config/VIRCAM_tec.isf).
- A list of the names and properties of the filters currently installed in the instrument is contained in the filter configuration file, vcmcfg/config/vcmcfgICS_filters.cfg.

If one or more new filters need to be installed in the camera, or if one or more existing filters need to be exchanged for different ones, the preferred way of doing this is with the *VIRCAM_gen_tec_LoadFilters* template (see section 8.10 on page 159 and section 11.7.7.6 on page 258). An alternative more primitive way of accessing and loading filters is using the *vciLoader* and *vciLoadAll* scripts described in section 8.4 on page 146.

The *VIRCAM_gen_tec_LoadFilters* template can be used when the full VIRCAM software is running. The template moves a list of filters one at a time to the load position and prompts an engineer to enter details of the new filter installed. If the new filter is described in the instrument description file, its properties will be automatically recalled. The template writes a new vcmcfgICS_filters.cfg file, which can be used to update the software configuration permanently.

The *vciLoadAll* script can be used in situations when only the ICS software is running. It will move to all the filter wheel positions in turn and give the engineer the option of exchanging each one. This script does not use the instrument description information, so new filter properties have to be entered manually.

The *vciLoader* script is an even more primitive script which simply puts a named filter into the loading position.

8.1.2 Camera cooldown and transit to telescope

The ICS contains thermal control software which can automatically take care of the temperature of the camera components (as described in section 2.1.2 on page 28). However, when the camera is being cooled or warmed for maintenance, the following software procedures are useful:

8.1.2.1 Triggering a cooldown

The thermal control software will switch to COOLDOWN mode automatically when it senses the cryocoolers or optical bench top cooling down. However, it may take the software a few minutes to notice. It is better to give the software advance notice of a cooldown by using the "Engineering→COOLDOWN" menu option on the thermal control panel at least half an hour before the cooldown starts (Figure 26 on page 64). This allows the software to start pre-warming the detectors and the cryostat window before the cooldown begins.

8.1.2.2 Transit to telescope

It is prudent to switch the software into WARMUP mode for an hour or so before disconnecting the camera from its power supply and moving it. This can be done by selecting the "Engineering→WARMUP" menu option on the thermal control panel (Figure 26 on page 64) at least half an hour before the transit begins. This will cause the detectors to be warmed a little before the transit and will help prevent the detectors attracting contaminants. (The detectors must not be the coldest objects in the cryostat during initial cooldown, warmup and transit).

8.1.3 Camera warmup

8.1.3.1 Triggering a warmup

The thermal control software will switch to WARMUP mode automatically when it senses the cryocoolers warming up. However, it may take the software a few minutes to notice. It is better to give the software advance notice of a warmup by using the "Engineering→WARMUP" menu option on the thermal control panel (Figure 26 on page 64) at least an hour before the warmup begins. This allows the software to start pre-warming the detectors before the other components start warming up.

8.2 Updating the Detector Bad Pixel Mask

The software release includes a bad pixel map for the detectors. This map is a multi-extension FITS file which has the same format as VIRCAM science data, except the data arrays contain a "0" (representing a good pixel) or a "1" (representing a bad pixel). The file is contained in the detector configuration module, vcdcfg, as

vcdcfg/src/VIRCAM_BadPixelMask.fits.bz2

The file is compressed with "bzip2" to reduce its size in the CMM repository. This file is uncompressed and installed into the DETDATA directory when the vcdcfg module is built. The bad pixel mask can be updated with the following procedure (assuming a new mask has been created using a data processing utility):

- Get the vcdcfg module from the CMM repository (with cmmModify).
- Compress the new bad pixel mask using bzip2 and write it to the src directory in the vcdcfg module.

- Replace the module in the CMM repository (with cmmArchive).

The procedure should be carried out by a software engineer, and the new vcdcfg module incorporated into the next release of the VIRCAM software. See section 6.2.3 on page 122 to compare with the procedure for making permanent changes to the vcmcfg module.

8.3 Filter Wheel Motor Configuration

8.3.1 Using motei

The filter wheel motor control parameters (acceleration, speed, circular optimisation, etc…) are contained in the *vcmcfgFILT.dbcfg* configuration file, which is created and maintained using the motor engineering utility *motei*, [RD52]. This utility can be activated from ICS engineering panel (Figure 38) by selecting "Tools → Motors → motei" from the menu, or from a Linux command prompt using the command:

```
% motei &
```

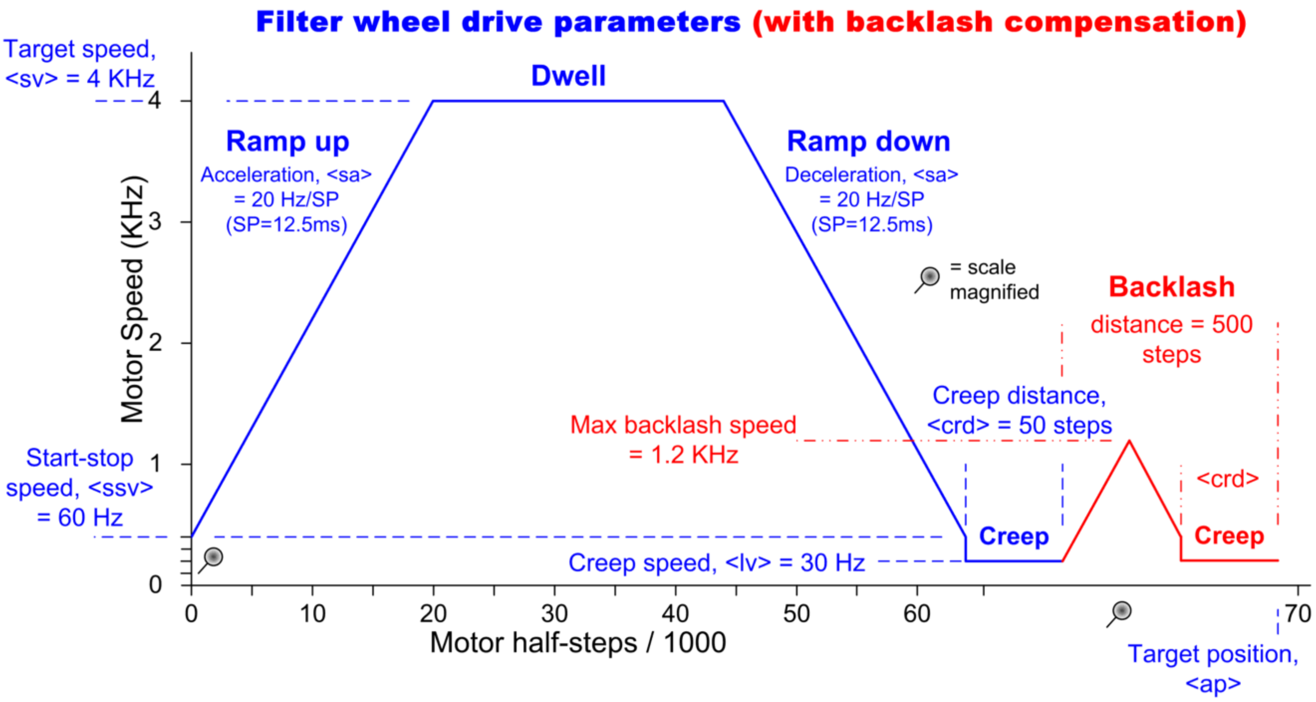


Figure 42 VISTA IR Camera Filter Wheel Motor Control Parameters

Figure 42 above illustrates the "motei" motor control parameters for the VISTA IR Camera filter wheel, and those same parameters are shown on the motei configuration panels in Figure 43 and Figure 44 below.

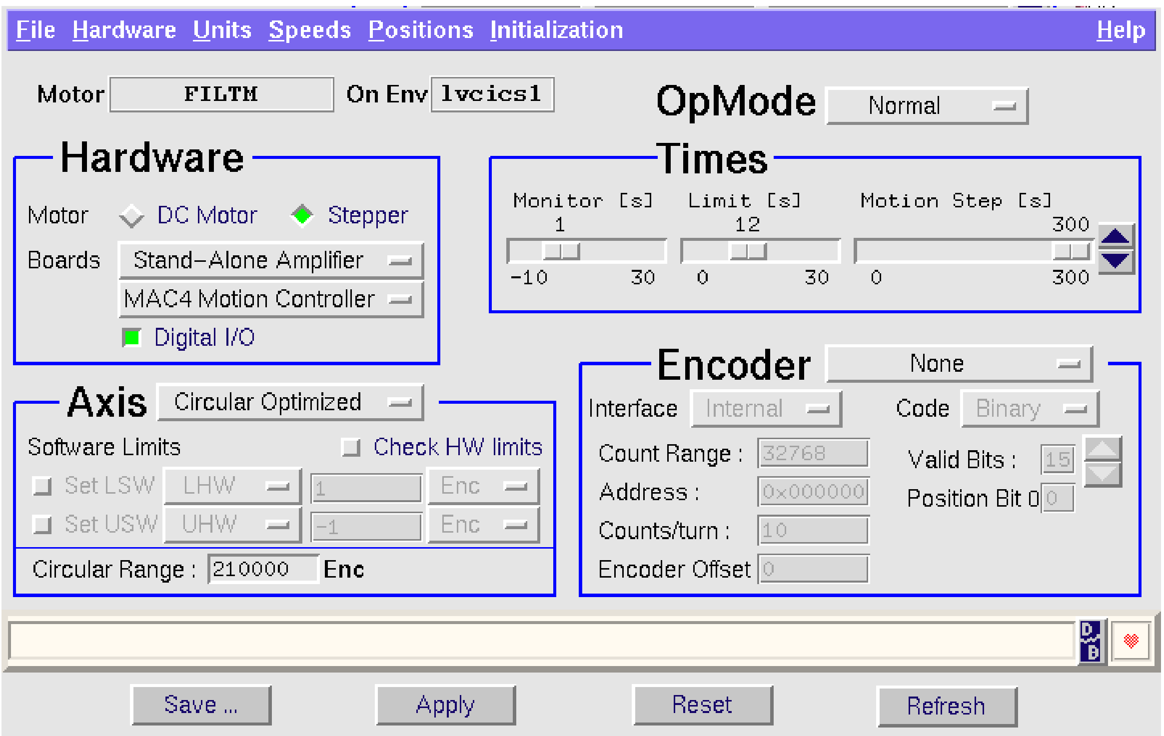


Figure 43 motei Configuration Panel for VIRCAM Filter Wheel

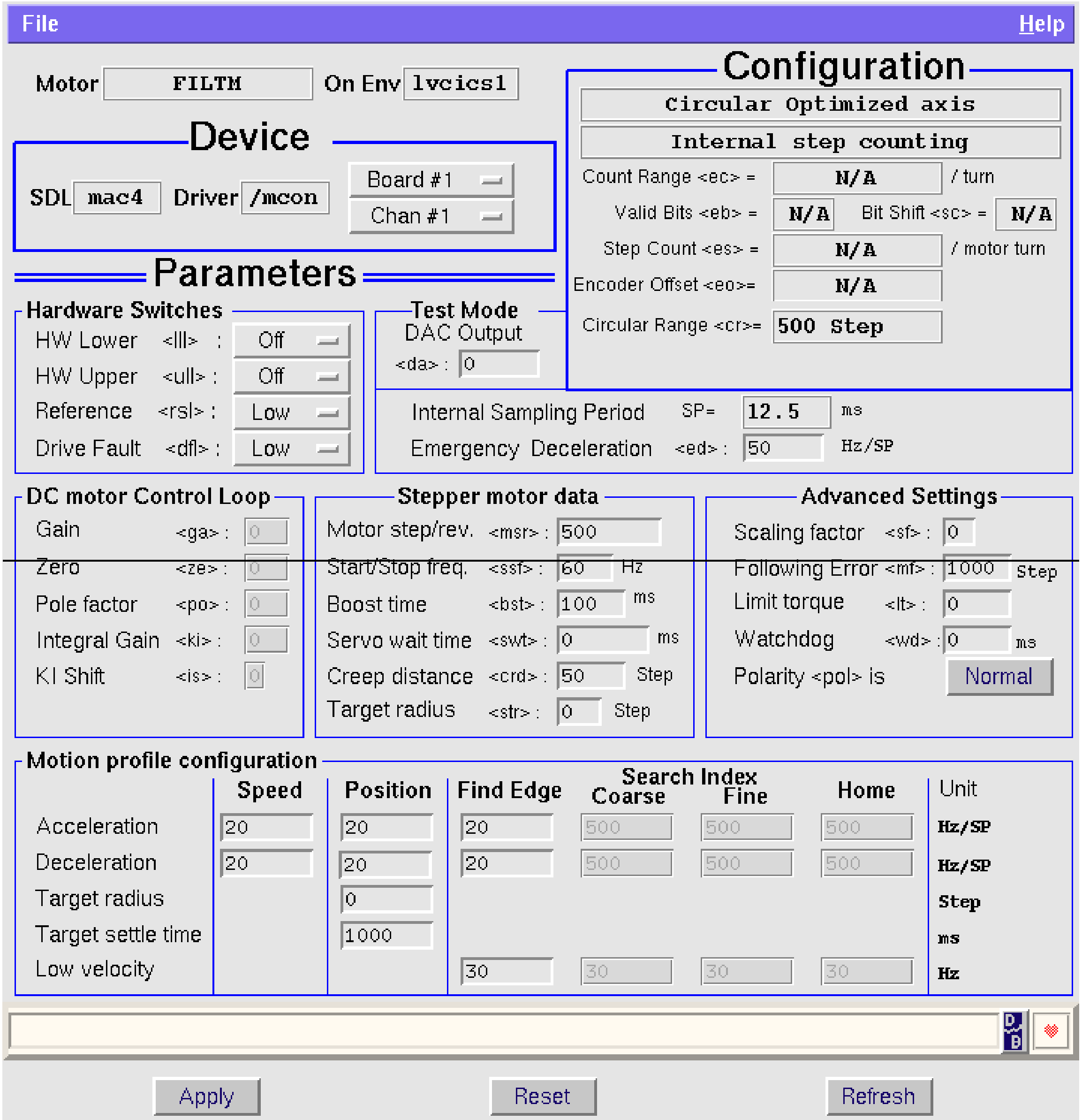


Figure 44 motei Controller Configuration panel for VIRCAM Filter Wheel

8.3.2 Using vciMakeFILTM

Some scripts were developed with the VIRCAM software to allow the commonly changed configuration parameters to be modified quickly without needing to start-up *motei* or have the LCU environment running. The filter wheel can be reconfigured on-the-fly using the *vciMakeFILTM* script (which must be executed from the top level source directory, just like the pkgin utilities). This script backs up the current filter wheel configuration file, "vcmcfg/config/vcmcfgFILTM.dbcfg", to "vcmcfg/config/vcmcfgFILTM.dbcfg_BACKUP" and then generates a new filter wheel configuration file from a template file. The new configuration file is then installed and the database is re-initialised. The new filter wheel configuration settings can be supplied as command line options for setting parameters such as acceleration, velocity, mechanism type, etc… The available command line options can be listed by using the help option as shown below.

```
% vciMakeFILTM -h
    -v nnn     default velocity (4000 steps/s)
    -b nnn     backlash velocity (1200 steps/s)
    -d nnn     datum velocity (2000 steps/s)
    -a nnn     normal acceleration (20 Hz/SP ?)
    -l         linear axis
    -c nnn     circular axis
    -o nnn     circular optimised axis (default with
               range 210000 steps)
    -s nnn     start stop velocity (60 steps/s)
    -f nnn     final velocity (30 steps/s)
    -t nnn     final travel distance (50 steps)
    -z nnntwo  step distance (500 steps)
    -h         this help message
```

There is no need to supply values for all of the command line options as the default values will be used unless overridden by a command line option. The default values are shown in the brackets when the help option is used to list the available command line options. If no command line options are supplied then all of the default values will be for reconfiguring the filter wheel.

The original filter wheel configuration that was backed up by *vciMakeFILTM* can be restored by calling the *vciRestoreFILTM* script.

8.3.2.1 A word of warning about vciMakeFILTM

The *vciMakeFILTM* script has been used successfully during the development of the VIRCAM software. However, it does make assumptions about the format of the *vcmcfgFILT.dbcfg* configuration file and may stop working if a new release of the ESO motor control software changes this format. The only ESO-recommended and future-safe way of configuring the filter wheel motor is through the motei utility.

8.4 Filter Wheel Maintenance Utilities

There are two filter wheel mechanism maintenance scripts available for aiding the process of loading filter trays. These scripts are *vciLoader* and *vciLoadAll*. The first of these, *vciLoader*, accepts a filter or slot name as an argument which it then moves into the filter load position so the filter tray is ready to be replaced by a new filter tray. A few examples of the script invocation are given below.

```
% vciLoader SLOT3
% vciLoader INT5
```

```
% vciLoader Ks
```

The second maintenance script, *vciLoadAll,* is more elaborate as it can used to sequentially load all of the filter trays and generate an updated filter wheel configuration file reflecting the new filter positions in the filter wheel. The user is prompted if they wish to move to each filter position in sequence. When the user selects a filter they are then prompted to enter the new filter's name, transmission, focus offset and wavelength. These configuration values are immediately updated in the configuration database and when all the filter are in place or the user chooses to enter the stop option at the command prompt they are then prompted if they wish to create a new updated configuration file. The new filter configuration file should be used by the user to replace the original file in the *vcmcfg* module's *config* directory. An example of the script invocation is given below.

```
% vciLoadAll
Move to filter z' ID <SLOT1> (y=yes, n=next filter, s=stop)? N
Move to filter DARKA ID <INT1> (y=yes, n=next filter, s=stop)? N
Move to filter Y ID <SLOT2> (y=yes, n=next filter, s=stop)? Y
Filter 3 name <Y>, enter new name (RETURN to skip)? H
Filter 3 name <H>, enter transmission (1=dark, 2=medium, 3=bright)? 3
Filter 3 name <H>, enter focus offset (mm)? -0.3
Filter 3 name <H>, enter wavelength (nm)? 1650
Move to filter DARKB ID <INT2> (y=yes, n=next filter, s=stop)? S
WARNING:
If you want you changes to the filter mechanism setup to become permanent
you MUST create a new vcmcfgICS_filter.cfg configuration file and use it to
replace the old one in MS/vcmcfg/config directory.
Create new filter configuration file (y=yes, n=no)? Y
Creating new filter slot configuration file vcmcfgICS_filters.cfg
Writing filter names
Writing filter transmissions
Writing filter focus offsets
Writing filter wavelengths
```

NOTE: There is also a maintenance template, VIRCAM_gen_tec_LoadFilters, to do this (see sections 8.10 and 11.7.7.6).

8.5 Maintenance Logging Facilities

8.5.1 Sensor data logging

Three commands are available, which spawn a process which uses the ccseiDbMonitor utility, [RD53], to record a log of regular temperature and pressure readings:

```
% vciLogTemperatures <maxRecords> <pollTime>
% vciLogVacuum <maxRecords> <pollTime>
% vciLogThermalControl <maxRecords> <pollTime>
```

vciLogTemperatures records all temperature readings, vciLogVacuum logs all vacuum readings and vciLogThermal logs thermal control information (such as the target temperatures and the mean temperatures used to make decisions). <maxRecords> is the maximum number of records to be recorded and <pollTime> is the time between each record. The following command can be used as a short cut to start all three processes:

```
% vciLogging <maxRecords> <pollTime>
```

The vciLogging command will stop any vciLogTemperatures, vciLogVacuum and vciLogThermalControl processes that are already running, before starting new ones. The records are written to log files whose names are of the form "$VLTDATA/tmp/<environment>_<date>_ccseiDb<description>.log". By default records are recorded every 10 minutes for 72 hours, although they can be continued indefinitely by reissuing the vciLogging command at regular intervals. However, the Base ICS historian is better for indefinite logging (section 5.17 on page 112).

8.5.2 Miscellaneous logging

The command

```
% vcinsLog <initials> <comment>
```

can be used at any time to write time-stamped messages into a file called "VIRCAM_LogFile_<year>-<month>.txt" within the user's home directory. This facility can be used to compile ad-hoc lab notes during commissioning or maintenance.

8.6 Using the ICS Engineering Panel

This ICS engineering panel has already been mentioned in section 4.15.3 on page 86. This section describes the usage of the panel in more detail. The ICS engineering panel has two notebooks (see Figure 38 and the other figures below):

- The left notebook contains two tabs *Motors* and *Thermal*. The *Motors* tab contains status information relating to all motor mechanisms. In the case of the VISTA IR Camera it contains only one filter wheel mechanism. The *Thermal* tab contains[18] status information for the cryostat thermal control.

- The right notebook contains seven tabs *LSM1, LSM2, LSM3, LSC1, VAC1*, *DIS1 and HB*. Each of these tabs corresponds to a particular sensor device. The *LSM1-3 devices* are the three Lakeshore 218 temperature monitor devices, *LSC1* is the Lakeshore 332 temperature controller device, *VAC1* is Pfeiffer TPG256 vacuum gauge device, *DIS1* is the digital I/O interface device and HB1 is the heartbeat device.

[18] NOTE: The *Thermal* tab is currently empty, but the same information can be obtained by selecting "Thermal" from the ICS menu.

At the top of the engineering panel there is a box containing the global status information of the complete LCU system. At the bottom there is window which relays command feedback and below that there is an information bar which provides further information about a device field when the user places the mouse pointer over it.

8.7 Changing LCU State (all devices)

When the LCU environment is started its initial global state will be *LOADED*. The same is true of the individual device states, with the exception of the sensor devices in the right notebook. The initial state of the sensor devices will be *ONLINE* as they have been specifically configured (via the "INS.CON.ONLINE" configuration keyword) to go *ONLINE* automatically during boot up phase of the LCU, so they are begin monitoring their sensors immediately. The user can change the global status of all LCU devices by selecting one of the status options from the *ICS* menu item as shown in Figure 45. The LCU states are the same as described in section 5.7 on page 96. Choosing the *ONLINE* state will bring the LCU global state *ONLINE* after each of the LCU devices have been brought online, which means they have been initialised in both software and hardware. As already mentioned, the sensor devices monitor their sensors when in the *ONLINE* state. The filter mechanism will carry out its software and hardware initialisation sequence when changed from the *LOADED* state to the *ONLINE* state. This includes a datum search procedure which involves physically moving the filter wheel. When *ONLINE* the LCU devices are fully operational and ready for use during observations. The user can switch the devices to a safe state such as *STANDBY* when they are not required or maintenance is being carried out.

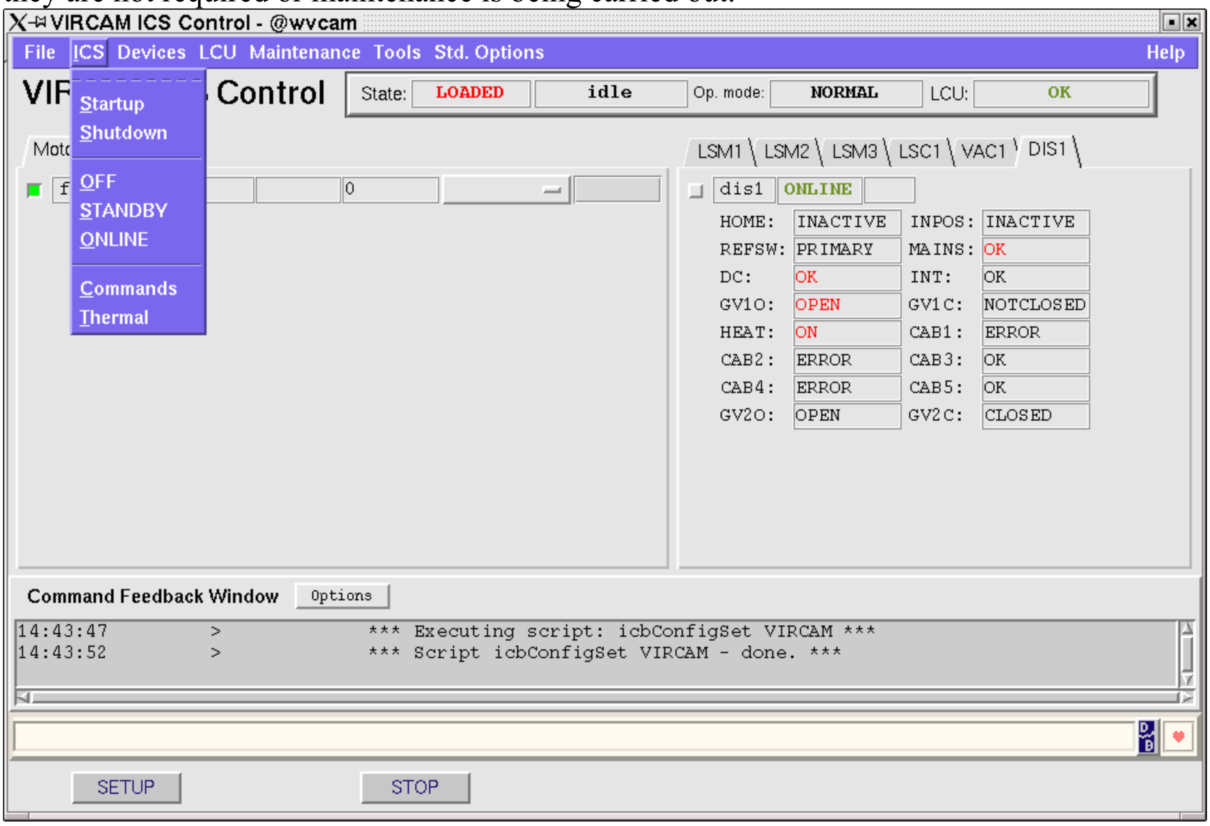


Figure 45 VIRCAM ICS Menu Options

8.8 Controlling Selected Devices

Each device in a notebook tab has a radio button on the far left. This button should be depressed if the user specifically wishes to control this device by sending commands. The commands that can be invoked on the selected devices are available in the *Devices* menu item shown in Figure 46. These commands include changing the state of the device(s) similar to those described in the preceding section. The “Select all devices” and “Deselect all devices” tabs can be used as a shortcut to set or unset all the device radio buttons. Starting with “Deselect all devices” is a way of ensuring only the devices you have specified will be included when a command or state button is pressed. (Some of the radio buttons are hidden behind the device tabs).

As well as the *Devices* menu there are also three buttons at the bottom of the panel with the command labels *SETUP*, *STOP* and *STOP-ALL*. SETUP can be used to configure a selected device, STOP will send a stop command to a selected device and STOP-ALL will stop all devices whether selected or not (useful as a software emergency stop). These commands are particularly useful when driving the filter wheel mechanism as described in the following section.

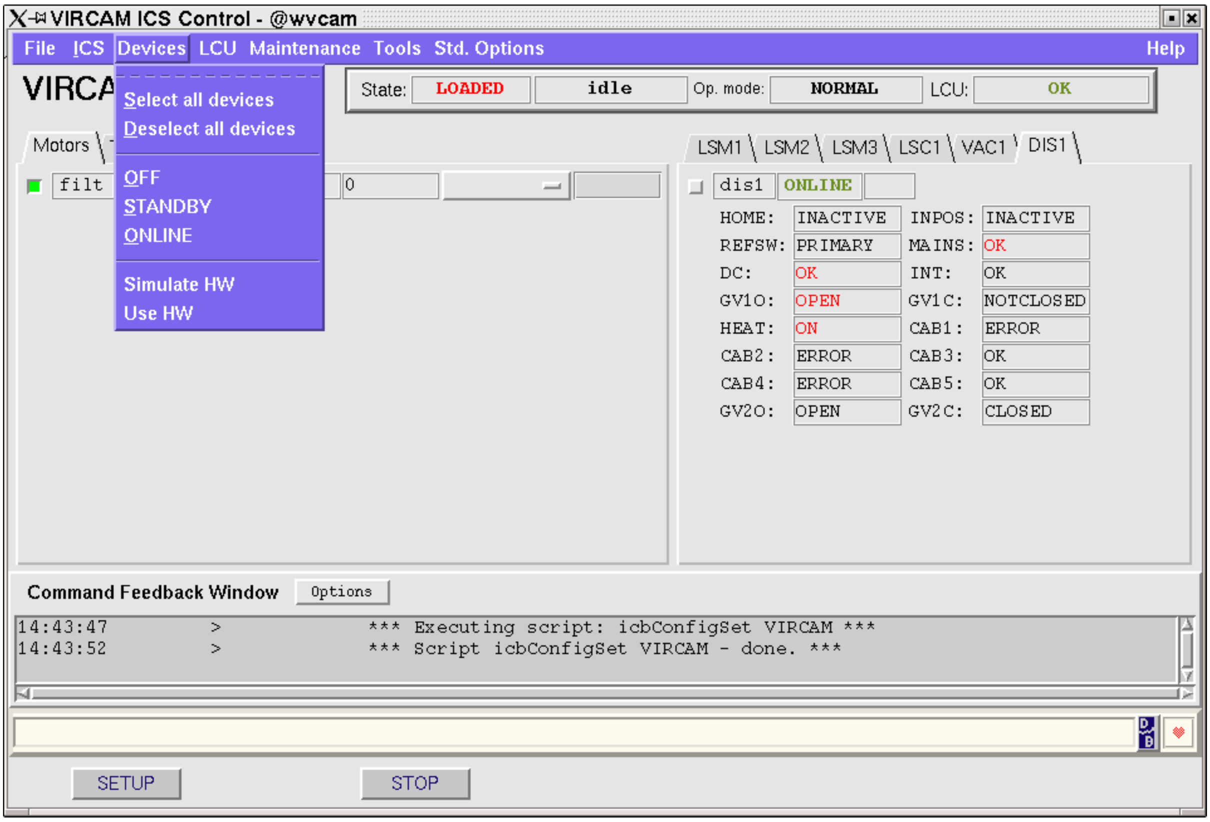


Figure 46 VIRCAM Devices Menu Options

8.8.1 Driving the Filter Wheel

The filter wheel device can be found under the *Motors* tab of the notebook on the left. There are a number of boxes containing status information and a couple of which allow the user to change the filter wheel settings. Reading from right to left these boxes are: device name, device state, operational mode, current filter position, current position (encoder units), new filter drop down menu and input box for relative or absolute positions (encoder units). This can be confirmed by placing the mouse pointer over each of the fields in turn and checking the information bar at the bottom of the panel. The last two are input fields where the user can select a new filter by name or choose to enter a new relative or absolute filter wheel position in encoder units. By clicking on the drop down menu the user has a list of the named filter wheel positions to choose from as shown in Figure 47. At the bottom of the menu are the options "enc" and "encrel" which the user can choose for entering a new absolute or relative position respectively. When one of these options is selected the user is then permitted to enter a value in the box to the right for the new position. After a new filter/position has been chosen the user must then click on the *SETUP* button at the bottom of the panel to initiate the move. The filter wheel move can be stopped by clicking on the *STOP-ALL* button (or clicking on the STOP button after selecting the filter device) also at the bottom of the panel. Please note that before the filter can be moved to a new position it must be in the online state, having completed its initialisation sequence.

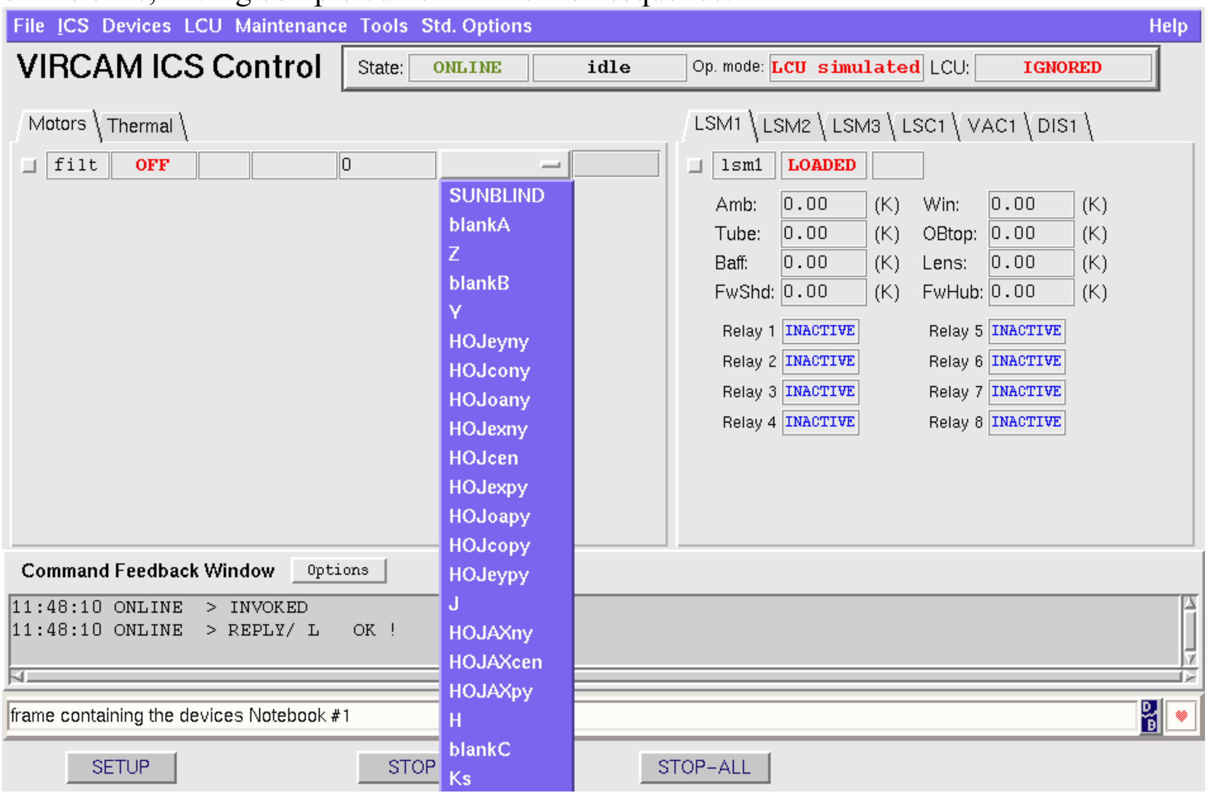


Figure 47 VIRCAM Filter wheel Drop Down Menu

8.8.2 Examining the Sensor Devices

The sensor devices can be found under the tabs of the notebook on the right. Examples are shown in Figure 48 and Figure 49. Each device has its own tab and the user can select a particular device by clicking on its tab. The sensor device tabs contain read-only status information, and the user cannot interact with any of them from this panel except for changing the state of the devices as described above. Each device tab has a number of boxes containing the following status information: device name, device state, operational mode and a number of boxes feeding back sensor readings. This can be confirmed by placing the mouse pointer over each of the fields in turn and checking the information bar at the bottom of the screen.

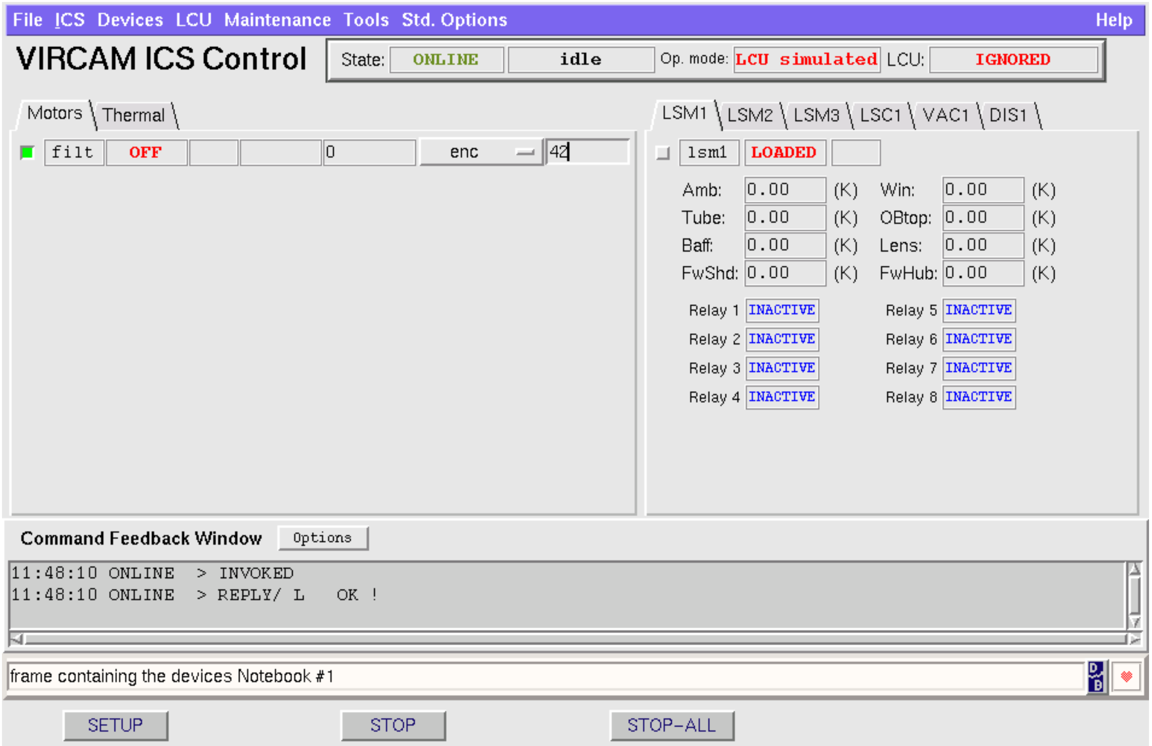


Figure 48 Lakeshore 218 Temperature Monitor Status

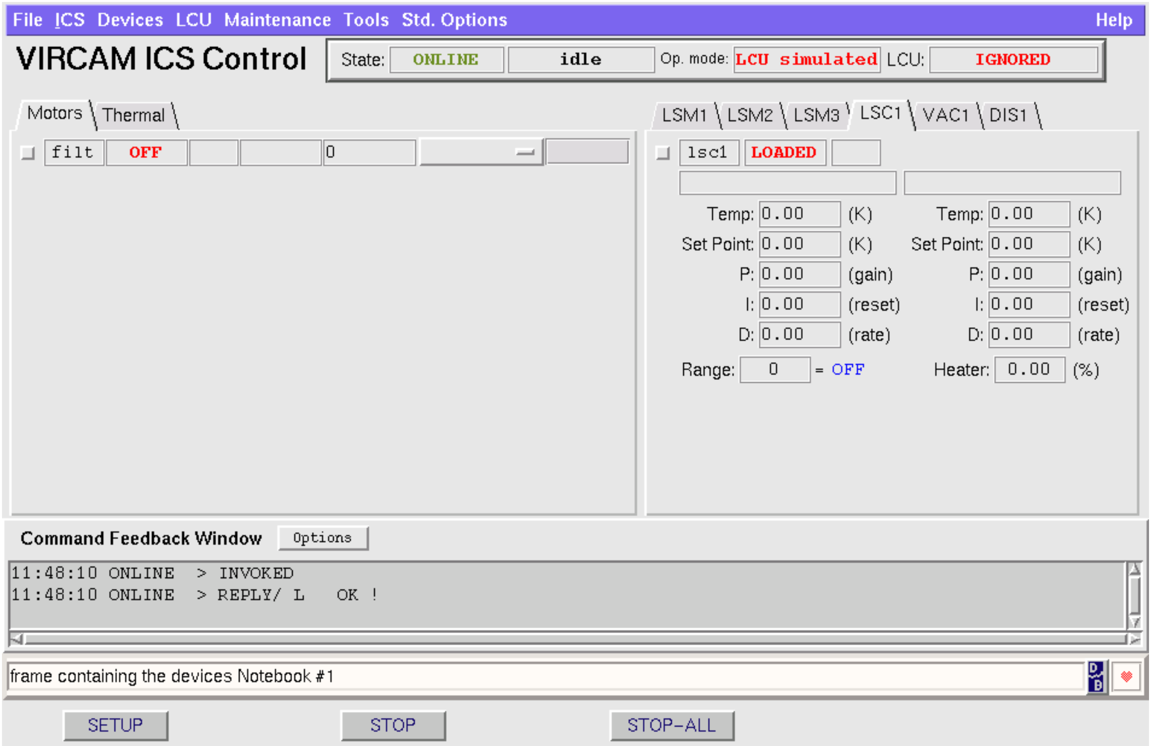


Figure 49 VIRCAM Lakeshore 332 Temperature Controller Status

8.9 *Lakeshore and Pfeiffer device diagnostic utilities*

If the ICS engineering screen does not provide sufficient information to diagnose a problem with a Lakeshore or Pfeiffer device, it is possible to interact with those devices directly from the LCU. To use these utilities, connect to the LCU console from a Linux prompt like this (assuming the LCU is switched on and booted):

```
% rlogin lvcics1
lvcics1->
      …
      …
lvcics1-> logout
%
```

The prompt "lvcics1->" shows commands entered from this LCU console. You *must* finish each session on the LCU with a "logout" command to free up the console for someone else.

8.9.1 Lakeshore 218 diagnostic utilities

Before communicating with the Lakeshore 218 devices, it is necessary to find out which file descriptors (fd) are associated with the three devices.

```
lvcics1-> vcilsmServerShow
```

```
vcilsm devices
--------------
index name      montask          fd       initialised
0     "lsm1"    0xe6f5150        45       Yes
1     "lsm2"    0xe6e4d30        46       Yes
2     "lsm3"    0xe6e4ba0        48       Yes
value = 2 = 0x2
lvcics1->
```

Any recognised Lakeshore 218 command can be sent to the device over the RS232 link with the following command. For example, here is a command to display the status byte of the LSM1 device:

```
lvcics1-> vcilsmHwConsole 45,"*STB?"
SUCCESS returned.
Reply = 000
value = 2 = 0x2
lvcics1->
```

The "Reply" value contains any reply from the command, in this case the value of the status byte.

8.9.1.1 Useful Lakeshore 218 commands

Here is a subset of useful Lakeshore 218 commands:

```
*   *CLS         - clear interface and terminate pending operations
    <no reply>

*   *ESE?        - query configuration of status reports in the
                   Standard Event Status Register
    <returns ESE bit field - see *ESR command>

*   *ESR?        - query Standard Event Status Register
    <returns standard event register as bit field:
       1 = Operation complete
       4 = QYE Query error - data loss due to queue overflow
       8 = DDE Device dependent error
      16 = EXE Execution error
      32 = CME Command error
     128 = PON Power on - power has been cycled>

*   *IDN?        - query identification
    <returns manufacturer,model,serial,firmware>

*   *RST         - reset instrument to power up setting
    <no reply>

*   *STB?        - query status byte
    <returns status byte as bit field:
       1 = new reading
       2 = (unused)
       4 = overload - input is SOVER, TOVER, SUNDER or TUNDER
```

```
       8 = alarm - there is an alarm condition
      16 = error - instrument error
      32 = ESB - bit set in standard event status register (use *ESR? for details)
      64 = SRQ - service request mode enabled
     128 = datalog done>

*    *TST?         - query result from power-up self-test
     <returns n: 0=no errors; 1=errors found>

*    MODE n        - set local mode (n=0), remote mode (n=1) or local lockout (n=2)

*    ALARM? x      - query alarm parameters for input x
     <returns on/off, source, high, low, deadband, latch-enable>

*    ALARMST? x    - query alarm status for input x
     <returns h,l: h=0/1 high activated/unactivated,
                   l=0/1 low activated/unactivated>

*    CRDG? x       - get Celsius reading from input x (where x=1...8)
     <returns reading>

*    KRDG? x       - get Kelvin reading from input x (where x=1...8)
     <returns reading>

*    DATETIME?     - query the date and time
     <returns date and time read from device>

*    AOUT? x       - get analogue output for data output x
     <returns value>

*    FILTER? x     - get filter parameters for input x
     <returns on/off, points, window filter parameters>

*    INPUT? x      - query input control parameter for input x
     <returns off/on>

*    MODE?         - query remote interface mode
     <returns 0=local; 1=remote; 2=remote+lockout>

*    RDGST? x      - query input status for input x
     <returns status bit field:
       16 = temp under range
       32 = temp over range
       64 = units under range
      128 = units over range>

*    RELAY? r      - query control parameters for relay r
     <returns mode (0=off;1=on;2=alarm), input(1-8),
                    alarm type (0=low;1=high;2=both)
```

Many more commands are detailed in the "*Lakeshore 218 Temperature Monitor User's Manual*", [RD20].

8.9.2 Lakeshore 332 diagnostic utilities

There is only one Lakeshore 332 device, so there is no need to find out which file descriptor this device is using. Any recognised Lakeshore 332 command can be sent to this device over the RS232 link with the following command. For example, here is the command to display the status byte of the LSC1 device:

```
lvcics1-> vcilscHwConsole"*STB?"
SUCCESS returned.
Reply = 000
value = 2 = 0x2
lvcics1->
```

The "Reply" value contains any reply from the command, in this case the value of the status byte.

8.9.2.1 Useful Lakeshore 332 commands

Here is a subset of useful Lakeshore 332 commands:

```
*    *CLS          - clear interface and terminate pending operations
     <no reply>

*    *ESE?         - query configuration of status reports in the
                     Standard Event Status Register
     <returns ESE bit field - see *ESR command>

*    *ESR?         - query Standard Event Status Register
     <returns standard event register as bit field:
        1 = Operation complete
        4 = QYE Query error - data loss due to queue overflow
        8 = DDE Device dependent error
       16 = EXE Execution error
       32 = CME Command error
      128 = PON Power on - power has been cycled>

*    *IDN?         - query identification
     <returns manufacturer,model,serial,firmware>

*    *RST          - reset instrument to power up setting
     <no reply>

*    *STB?         - query status byte
     <returns status byte as bit field:
        1 = new reading
        2 = (unused)
        4 = overload - input is SOVER, TOVER, SUNDER or TUNDER
        8 = alarm - there is an alarm condition
       16 = error - instrument error
       32 = ESB - bit set in standard event status register (use *ESR? for details)
       64 = SRQ - service request mode enabled
      128 = datalog done>

*    *TST?         - query result from power-up self-test
     <returns n: 0=no errors; 1=errors found>

*    MODE n        - set local mode (n=0), remote mode (n=1) or local lockout (n=2)

*    CRDG? x       - get Celsius reading from input x (where x=A or B)
     <returns reading>

*    KRDG? x       - get Kelvin reading from input x (where x=A or B)
     <returns reading>
```

```
*   DATETIME?     - query the date and time
    <returns date and time read from device>

*   HTR?          - query heater percentage for channel A
    <returns heater percentage for channel A>

*   AOUT?         - query analogue output for channel B
    <returns analogue output for channel B>

*   PID x, p, i, d - Set PID values for channel x
    <no reply>

*   PID? x        - query PID values for channel x
    <returns p, i, d>

*   RAMP x, o, r - Set ramp for channel x, on/off at rate r
    <no reply>

*   RAMP? x       - query ramp setting from channel x
    <returns o, r: o=0/1 ON/OFF; r=ramp in Kelvin/min>

*   RANGE r       - sets heater range value to r
    <no reply>

*   RANGE?        - query heater range value
    <returns range value>

*   SETP x, y     - write temperature set point for channel x to y
    <no reply>

    SETP? x       - query temperature set point for channel x
    <returns set point>
```

Many more commands are detailed in the *"Lakeshore 332 Temperature Controller User's Manual"*, [RD21].

8.9.3 Pfeiffer TGP256 diagnostic utilities

Any recognised Pfeiffer TPG command can be sent to this device over the RS232 link with the following command. For example, here is the command to display the error diagnostic code of the VAC1 device:

```
lvcics1-> vcitpgHwConsole "ERR"
SUCCESS returned.
Reply = 00000.00000
value = 2 = 0x2
lvcics1->
```

The "Reply" value contains any reply from the command, in this case the value of the error diagnostic code.

8.9.3.1 Useful Pfeiffer TPG256 commands

Here is a subset of useful Pfeiffer Vacuum Gauge commands. In general, missing out a command parameter turns the command into a query:

```
*    DCD ,n                   - change display to n decimal digits (2 or 3)
     DCD                      - query display digits
     <returns n>

*    ERR          - get error information
     <returns diagnostic code xxxxx.yyyyy - see below>

*    PNR                        - get program version
     <returns program version>

*    PRx          - read sensor X (where X=1...6)
     <returns sensor status, value
      status = 0 -> measurement data ok
               1 -> under range
               2 -> over range
               3 -> over range
               4 -> sensor off
               5 -> no sensor
               6 -> identification error>

*    SCx ,ons,offs,onv,offv   - configure sensor control
                                (x = A,B,C,D,E,F for sensors 1,2,3,4,5,6)
                                (ons = switch on controlling source for sensor;
                                       0,1,2,3,4,5 = sensor 1,2,3,4,6,6;
                                       6 = external control; 7 = manual)
                                (ons = switch off controlling source for sensor;
                                       0,1,2,3,4,5 = sensor 1,2,3,4,6,6;
                                       6 = external control; 7 = manual)
                                (onv = switch off value)
                                (offv = switch on value)
     SCx                      - query sensor control for sensor x
     <returns ons,offs,onv,offv>

*    SEN ,x,x,x,x,x,x         - switch sensors 1,2,3,4,5,6 on/off
                                (x=0:no change, x=1:off, x=2:on)
     SEN                      - query which sensors are on or off
     <returns x,x,x,x,x,x>

*    TEP                      - EPROM test (do not use this command a lot)
     <returns diagnostic code xxxxx.yyyyy - see below>

*    TID                      - sensor identification
     <returns id, id, id, ...>

*    TRA                      - RAM test (engineering)
     <returns diagnostic code xxxxx.yyyyy - see below>

*    PUC ,x,x,x,x,x,x         - Set underrange control for sensors 1,2,3,4,5,6
                                (x=0:deactivated; x=1:activated)
     PUC                      - Query underrange control
     <returns x,x,x,x,x,x>

*    UNI ,x                   - Set measurement unit (0=mbar, 1=Torr, 2=Pascal)
     UNI                      - Query measurement unit
     <returns x>
```

Many more commands are detailed in the *"Pfeiffer TPG256 Operator Manual"*, [RD22].

8.9.3.2 Pfeiffer TPG256 Diagnostic codes

The code is reported as a bit field

```
xxxxx.yyyyy

      yyyyy =     0 -> No error
                  1 -> Sensor 1: measurement error
                  2 -> Sensor 2: measurement error
                  4 -> Sensor 3: measurement error
                  8 -> Sensor 4: measurement error
                 16 -> Sensor 5: measurement error
                 32 -> Sensor 6: measurement error
                512 -> Sensor 1: identification error
               1024 -> Sensor 2: identification error
               2048 -> Sensor 3: identification error
               4096 -> Sensor 4: identification error
               8192 -> Sensor 5: identification error
              16384 -> Sensor 6: identification error

xxxxx =     0 -> no error
            1 -> Watchdog has reported
            2 -> Task fail error
            4 -> IDCX idle error
            8 -> Stack overflow error
           16 -> EPROM error
           32 -> RAM error
           64 -> EEPROM error
          128 -> Key error
         4096 -> Syntax error
         8192 -> Inadmissible parameter
        16384 -> No hardware
        32768 -> Fatal error
```

8.10 Maintenance Templates

In addition to the templates described in section 5.18 on page 113, the VIRCAM software provides a number of maintenance templates. These templates are not available to P2PP, and are designed to be run at the instrument by a maintenance engineer.

- **VIRCAM_gen_tec_SelfTest.tsf** — A template that tests the operation of the instrument by executing all the observation templates.

- **VIRCAM_gen_tec_CalibFilter.tsf** — A template that checks the accuracy and repeatability of the filter wheel by making an exposure, moving the filter wheel away and back again, and repeating the exposure.

- **VIRCAM_gen_tec_CheckFilters.tsf** — A template that checks the functioning of the filter wheel by selecting filters one at a time from a list of filters. The template reports the minimum, maximum and mean exchange time between the filters.

- **VIRCAM_img_acq_random.tsf** — This template acquires a target at a randomly-chosen altitude and azimuth within the telescope's limits. It can be used for soak-testing the VISTA telescope systems when it is not practical to choose a set of targets in advance.

- **VIRCAM_img_tec_FocusFilters.tsf** — This template derives the best telescope focus offset for a science filter, or list of science filters. Several exposures are made at different focus offsets, and a MIDAS task is used to derive the best focus offset for each filter.

- **VIRCAM_img_tec_LoadFilters.tsf** — This template takes an engineer through the procedure to load a series of filters. The template prompts the engineer to provide the name and properties of the new filters being installed and uses this information to build a new filter wheel configuration file.

- **VIRCAM_img_cal_labflats.tsf** — This template makes a series of flat-field observations, with corresponding darks, similar to VIRCAM_img_cal_linearity, except it is designed to use a laboratory light source rather than the flat-field screen in the enclosure. The template uses an arithmetic sequence of exposure times (given a minimum exposure time, increment and number of steps). Each flat-field exposure is interspersed with a flat-field made with a fixed exposure time, which can be used to check for any variations in the laboratory light source with time.

- **VIRCAM_gen_tec_StrayLight.tsf** — This template carries out an automatic stray light investigation by taking several exposures with the filter wheel offset from the central position by differing amounts. If any stray light pattern results from a reflection from a component mounted on the filter wheel, the reflection pattern will be seen to move. The same template can also be used to verify that the filter wheel is moving, to check the vignetting limits of the filter wheel, and to test the orientation of the detectors with respect to the filter wheel.

- **VIRCAM_gen_tec_exp.tsf** — This template makes a series of exposures designed to test the science detectors. It also times the sequence and reports the data acquisition performance statistics.

- **VIRCAM_howfs_tec_loopback.tsf** — This technical template generates a set of loopback data containing the NULL coefficients associated with each HOWFS filter. The filter parameters are obtained from VIRCAM_HOWFS*.paf files in the directory $INS_ROOT/$INS_USER/MISC/VISTA/VIRCAM_HOWFS. The loopback data files are written to the directory $INS_ROOT/$INS_USER/HOWFSDATA.

9 FAQ AND TROUBLESHOOTING

It is recommended that the logMonitor utility is always run alongside the software, to report the details of any errors encountered. The utility can be run on its own with the command

```
% logMonitor &
```

or can be started alongside other VIRCAM panels by adding "LOG" to the list of panels to be started, for example

```
% vcinsStart -panel ICS LOG
```

9.1 Recovering from a system reboot or power failure

9.1.1 Reboot or power failure affects workstation only

If a reboot or power failure affects only the instrument workstation, the instrument LCU will carry on maintaining the last requested temperature set points.

If the instrument workstation is rebooted, all the workstation environments will stop running. An msql daemon should automatically be restarted to manage the ACC database. The msql server and ACC database should first be checked with this command:

```
% vccShow
```

If this command fails, the msql daemon may need to be started manually. The procedure is described in the VLT common software installation documents, [RD41] and [RD42]. Next, the environments will need to be restarted and the scan links between the workstation environment, LCU and VISTA TCS need to be reenabled. First log in on the telescope workstation and (if it is not running) restart the VISTA TCS environment as described in the VISTA TCS documentation; then return to the instrument workstation, log in as account "vcmgr", and rebuild the environments with this command:

```
% pkginBuild vcins -fromstep BUILD_ENV
```

(typed in from the top level source directory – i.e. the one containing the "vcins" module). Make sure the LCU is switched on.

If the VIRCAM software was running when the reboot happened it will need to be restarted with the command:

```
% vcinsStart
```

The ONLINE command should be sent to put the software back into the ONLINE state.

9.1.2 Reboot or power failure affects LCU only

If the LCU reboots while the VIRCAM software is running, it should restart and reconnect automatically when the power is restored. During the reboot the LCU status reported on the ICS engineering screen will change to "Not OK". The status will return to "OK" when the LCU has been successfully rebooted. The LCU software finishes booting in the STANDBY state, and it will be necessary to send the ONLINE command to put the software back into the ONLINE state.

The detector temperature control device on the LCU (LSC1) goes through the following steps when the LCU reboots:

- The device is initialised and an arbitrary default set point of 67.5K is defined. This state lasts for a couple of seconds.
- The device reads its configuration parameters from the LCU database and executes its INIT commands. These commands will define a set point which is the same as the current FPA temperature.
- The LCU then waits for further commands from the workstation.

If no further commands are sent from the workstation, the LCU maintains the status quo. The workstation software should be put back into the ONLINE state to restore full control.

9.1.3 Reboot or power failure affects both workstation and LCU

In this situation the LCU will try and reboot automatically when the power is restored, but since the LCU needs to download its operating system from the workstation, and also needs to communicate with the workstation environment, it is unlikely that the reboot will succeed. The Lakeshore devices will not be initialised by the LCU and they will not switch on any heaters. If the LCU software does not boot, the detector thermal protection system will start to warm the detectors.

The same procedure described in section 9.1.1 above should be used to recover from a general power failure.

9.2 Configuration problems

If the filter wheel doesn't move to the correct position, the target temperatures look wrong, or some devices don't start up correctly, there may be a problem with the software configuration (especially if engineering work has taken place recently). The configuration can be restored to its original state by retrieving a fresh copy of the vcmcfg module from the CMM repository, like this

```
% cd MS
% mv vcmcfg vcmcfg-old
% cmmCopy vcmcfg
```

then by rebuilding the "vcmcfg" module and regenerating the LCU configuration database, like this:

```
% cd MS/vcmcfg/src
% make clean all man install
% icbConfigSet VIRCAM
```

The configuration files are installed in the $INS_ROOT/SYSTEM/COMMON/CONFIGFILES directory, and the LCU configuration database is stored in the file $VLTDATA/config/lvcics1.dbcfg. Make sure all these files are updated. After that, try the "vcinsStartupDev", "vcinsFilterConfig", "vcinsThermalConfig" and "vcinsWcsConfig" panels and make sure the configuration is as you expect it. (N.B. Do not start any of these panels from the vcmcfg/config directory. The panels will use the files in the current directory in preference to $INS_ROOT). Finally, reboot the LCU to make sure it reads the new configuration database.

If rebuilding the "vcmcfg" module doesn't work, the ultimate reset is to remove the old installation and rebuild the software from scratch. First make a backup copy of $INTROOT and $INS_ROOT, for example:

```
% cd /introot
% mv vcam vcam_<date>
% mkdir vcam

% cd /insroot
% mv vcam vcam_<date>
% mkdir vcam
```

If you genuinely don't care about the contents of $INTROOT and $INS_ROOT, this command

```
% vcinsDestroy
```

will remove the entire contents of $INTROOT and $INS_ROOT automatically , leaving only empty directories. *Think carefully before using vcinsDestroy — it also removes all the data from DETDATA. A safer alternative is just to destroy* the *$INTROOT directory with the commands*

```
% rm -rf $INTROOT; mkdir $INTROOT
```

OPTIONAL: If you suspect the software source itself has become corrupted, as new version can be created from the CMM repository like this (*make a backup copy of the directory if you want to keep any of it*):

```
% cd <top level source directory>
% rm -rf DCS HOWFS ICS MS OS dicVIRCAM vcins
% cmmCopy vcins
% pkginBuild vcins -step RETRIEVE
```

Then the following command can be used to rebuild the software (from the top level source directory):

```
% pkginBuild vcins
```

9.3 Problems at system startup

9.3.1 Login fails

1. Make sure the terminal you are attempting to log in from is connecting to the host wvcam.
2. Make sure you have entered the correct user name (vc) and password.
3. If login still fails, make sure the wvcam workstation is working correctly.

9.3.2 Software fails to start

1. Make sure the wvcam software environment exists and is running:

   ```
   % vccShow
   % vccEnvCheck -e wvcam
   ```

   If not, follow the restart procedure described in section 9.1.
2. Make sure you have logged in as user "vc".
3. Make sure the DISPLAY, INS_ROOT and INTROOT environment variables are defined correctly (see section 4.2 on page 57).
4. Try the command

   ```
   % vcinsCheckPermissions
   ```

   to make sure all the relevant files have the correct permissions.
5. If one particular subsystem is failing to start, check that subsystem is running in the correct simulation mode. Look at the "simulation" row on the OS control panel or bring up the simulation status panel with the command:

```
% vcopanSimStatus &
```

The simulation mode can be corrected with the "vcinsStartupDev" panel (see Figure 24 on page 62).

6. If the ICS is in the correct simulation mode but still fails to start up, is the LCU running? Check with commands such as these:

```
% vccEnvCheck -e lvcics1
% ping lvcics1
```

   If there is a problem with the LCU, check it is switched on or reboot it.
7. Check the startup configuration contained in $INS_ROOT/SYSTEM/COMMON/CONFIGFILES/vcmcfgSTART.cfg. If you suspect a more widespread configuration problem, follow the procedure in section 9.1.2.

9.3.3 Software fails to go ONLINE

1. Start the OS engineering panel

```
% vcinsStart -panel OS_ENGINEERING
```

   and check which subsystem has failed to go ONLINE.
2. If the ICS has failed to go ONLINE, the most likely cause is that one of the devices has not initialised properly. The ICS engineering panel, which may be started by pressing the "GUI" button in the ICS column of the OS engineering panel or started directly with the command

```
% vcinsStart -panel ICS
```

   can be used to investigate which device has failed. For filter wheel problems, see section 9.9 or for sensor problems see section 9.10, below.
3. If the IRACE has failed to go ONLINE, make sure the number cruncher workstations are up and running, make sure IRACE is configured in the correct simulation mode, and make sure the SDMA environment variables are correct (see section 4.2 on page 57 and section 9.7 below).
4. If the ICS and IRACE are both ONLINE but the OS still refuses to go ONLINE, the problem may be with the TCS subsystem. There are two possibilities:
   - The VISTA TCS is not required but it is not being ignored. Enter "Telescope → Disable" from the OS control panel and repeat the ONLINE command.
   - The VISTA TCS is required but it is not in the ONLINE state. If the VISTA TCS simulator is being used, press the "STARTUP" and "ONLINE" buttons on the TCS column of the OS Engineering panel to start it up and put it in the ONLINE state. If the VISTA TCS is not being simulated, consult the TCS documentation, [RD13], to find out why it is not ONLINE.

5. If the HOWFS system is not ONLINE this doesn't prevent the OS from going ONLINE, since the HOWFS is not a proper subsystem. The HOWFS may move to the STANDBY state when HOWFS observations are not being made. If the HOWFS system is in the LOADED state its process is not running. It may be restarted by pressing the STARTUP button on the HOWFS column of the OS engineering panel.

9.4 Problems when loading templates into BOB

9.4.1 The PAF.NAME keyword is not part of the PARFILE string.

To stop the messages *"The PAF.NAME keyword is not part of the PARFILE string. The format of the OBD is therefore not standard."* coming from BOB, type the following command:

```
% vcmseqNoP2PP
```

The command will need to be repeated each time the software is rebuilt (specifically the template modules vcotsf, vcoseq, vcmtsf or vcmseq).

The VIRCAM templates are designed to receive their observation blocks from P2PP. The tile templates use SEQ.REF.FILEi parameters (where i is a sequential integer) to define the names of the files containing the guide star information for each pawprint. P2PP sends this information by encoding the files into a specially formated string which BOB interprets as needing to be saved to a file. These files are communicated using parameters of type "paramfile". After this process has completed the VIRCAM software sees a file name in these parameters.

To test the VIRCAM software without P2PP, a set of files has to be provided manually and the names of these files stored in the SEQ.REF.FILEi parameters. When BOB sees these parameters it attempts to parse them as P2PP-encoded messages and generates the error messages. The `vcmseqNoP2PP` script converts all the SEQ.REF.FILEi parameters to type "string", which BOB is happy to deal with.

After using the `vcmseqNoP2PP` script, the VIRCAM software must be rebuilt before P2PP can be used with it again.

9.5 Problems when running exposures

For TCS, IRACE, HOWFS, filter wheel or sensor device problems seen while running exposures, see below.

9.6 TCS problems

9.6.1 Cannot sent command to the TCS or access tif

The TCS must be running and ONLINE. Check that the RA and Dec fields on the OS engineering screen are ok — if their background colour changes to grey then the TCS is not working, or the scan link from the TCS environment to wvcam has failed.

9.6.2 TCS reports "out of limit" error when presetting to a target

This is a common TCS error, but it is not serious. It means that the RA and Dec or PA you have chosen is outside the TCS limits (e.g. the object is outside the altitude limits). Choose another RA and Dec and try again (compare the RA with the sidereal time).

NOTE: The RA and Dec sent to the TCS are time dependent, so a test which succeeds with a particular RA and Dec may fail with this "out of limit" error a few hours later. To get around this problem, the VIRCAM self-test scripts use an RA and Dec within the small "always visible" zone around the south celestial pole.

9.6.3 TCS reports "No guide star in catalogue" error when presetting to a target.

This error happens when an acquisition template or tile template is being executed in which the VISTA TCS has been instructed to find its own guide stars, using the SETUP string

```
-function TEL.AG.START T TEL.AG.GUIDESTAR CATALOGUE
```

The TCS has moved to a new pointing and cannot find any suitable guide stars within its catalogue. This problem can be avoided by choosing guide stars in advance with the VISTA Survey Area Definition Tool, [RD15]. If you don't have any guide stars chosen in advance, the failed template can be repeated with one of the following adjustments:

- Try slightly different pointing coordinates. (This is pot luck, and if you are pointing near the Galactic pole it might not help).
- Set "`TEL.AG.CONFIRM T`" and use the VISTA TCS to choose a guide star manually using pixel coordinates.
- Send a "`SETUP -function TEL.AG.MINMAG <minmag> TEL.AG.MAXMAG <maxmag>`" command to the VISTA TCS to make it less fussy about the guide star magnitude range.
- If all else fails, change "`TEL.AG.START T`" to "`TEL.AG.START F`" and turn off autoguiding.

9.6.4 TCS reports "Guide star position is off chip"

This can happen when something has gone wrong with the guide star selection procedure and a guide star has been provided to the TCS which is outside both of the autoguider chips. In this situation autoguiding can be recovered by setting "`TEL.AG.CONFIRM T`" and using the VISTA TCS to choose a guide star manually. Failing that, the TCS can be made to choose a guide star for itself by specifying "TEL.AG.GUIDESTAR CATALOGUE" in the SETUP string.

9.7 *IRACE problems*

Refer to [RD39] for IRACE disgnostic procedures.

9.7.1 IRACE DCS will not go ONLINE

If the IRACE DCS will not go ONLINE, check the error message displayed on the logMonitor screen. The most common reason for a failure to go ONLINE is a problem with the IRACE configuration or the hardware setup.

- Make sure you have selected the correct IRACE configuration. If you intended to run IRACE in workstation simulation mode, make sure you have selected that option.
- The error message "IRACE error - unable to download bootfile" means there is a communication problem between the iracqServer process and the IRACE DFE electronics. Bring up the OS status panel with the command

```
% vcinsStart -panel OS_STATUS
```

and make sure the SDMA command and data port numbers are set correctly. Check that the number crunchers are up and running and verify that the fibres are connected between the number crunchers and DFE boxes — try swapping the fibres and see if that makes a difference.

9.7.2 Ring buffer overflow

If IRACE fails with a "ring buffer overflow" error, the IRACE acquisition processes are unable to keep up with the acquisition of the data. This is very unlikely to happen with the full IRACE system, although if it does check for any programs that could be hogging resources on the number cruncher workstations (wvcirc1 and wvcirc2).

It is much more likely that IRACE is running in simulation mode, and this problem is caused by the simulated acquisition processes on the instrument workstation being unable to keep up. These processes need to be installed with privileges in order to work properly. See the installation procedure described in section 3.3.3 on page 54.

9.7.3 IRACE error - exposure is still active

This error message from IRACE isn't a problem with IRACE itself. It is a side effect of an error happening in the exposure command sequence. The true problem will be reported a little earlier in the error log.

9.8 *HOWFS problems*

See [RD7].

9.8.1 File not found

If this error is seen in response to an ANASTAR command, it is likely that the HOWFS software has not been configured correctly, or one or more of the files to which the software is configured to use no longer exist. Bring up the vchpanControl panel (Figure 34) to see the names of the files involved.

If this error is seen during the execution of a HOWFS template, it is likely that some calibration files are missing. The VIRCAM_howfs_cal_dark and VIRCAM_howfs_cal_domeflat templates must be used to generate suitable calibration data before the VIRCAM_howfs_obs_wfront template can be used.

9.8.2 Image analysis takes a very long time

It is perfectly normal for the HOWFS image analysis to take several minutes. The normal execution time is around 5 minutes, but a difficult data set could take up to 30 minutes. Bring up the vchpanControl panel (Figure 34) and make sure the simplex relative tolerance is decreasing. If the iteration count is increasing and the relative tolerance shows no sign of decreasing, the analysis may be aborted using the ABORT button on the panel. If you are happy the relative tolerance is small enough, the analysis can be stopped, and the latest fit use to generate coefficients, using the STOP button.

If the image analysis server is in the BUSY substate but the iteration count is not changing, the process may have hung up. Try using the ABORT button. If the process fails to respond, try using "HOWFS → Shutdown". If the shutdown succeeds, restart the process using "HOWFS → Startup". If it fails, try stopping and restarting the wvcam environment.

9.8.3 Image analysis finishes but fails to converge

This problem may indicate poor data. Use the "I…" button to display the data being fitted. If a coefficients file is available, use the "Display…" button to view the diagnostic images contained within it. Try choosing a different (and possibly brighter) HOWFS star. If that fails, try using "HOWFS → FITVEC" to select fewer wavefront coefficients for fitting. Alternately, repeat the observation using a longer exposure time or a brighter star.

9.9 Filter wheel problems

9.9.1 Timeout during initialisation/datum operation

If the filter wheel software reports a timeout during its initialisation, then the filter wheel has been unable to find its reference switch within the allowed timeout. Assuming the timeout parameter has not been reduced below its default value, and this problem happens constantly, suspect a problem with the filter wheel's home switch. Try the following procedure:

1. Flip the reference switch selector on the back of the LCU from PRIMARY to SECONDARY.

2. Bring up the expert startup panel "vcinsStartupDev" and change the filter wheel reference switch selection from PRIMARY to SECONDARY. Then press the "Save configuration" button (do not start the software yet).

3. Since changing the reference switch is a major configuration change, make sure the LCU configuration database is updated with the commands:

   % vciConfigSet
   % icbConfigSet VIRCAM

   The vciConfigSet procedure will recalculate all the POSENC values to become offsets from the new reference switch.

4. Now update the LCU by using the commands OFF, ONLINE. If this doesn't work, try restarting the LCU environment with:

   % vccEnvStart –e lvcics1

The filter wheel should now initialise by looking for the secondary reference switch. The primary switch must be repaired at the earliest opportunity.

9.9.2 Reference/home switch configuration problems

The filter wheel has two reference switches — PRIMARY and SECONDARY. This allows the filter wheel to continue to be used when the primary switch has broken, but it can also lead the problems. The filter wheel will only work properly when the switch at the back of the LCU and its software configuration refer to the same switch. Suspect a mismatch if the filter wheel always misses its intended destination by the same number of steps (resulting in exactly the same vignetting pattern on each exposure). There are two ways of recovering from this problem:

- If the filter wheel is supposed to be using its PRIMARY reference switch, check the selector switch on the back of the LCU is set to PRIMARY. Then the safest thing to do is to rebuild the software configuration, as described in section 9.1.2.

- If the filter wheel is supposed to be using its SECONDARY reference switch (i.e. the primary one has already broken) , check the selector switch on the back of the LCU is set to SECONDARY. Then use the procedure described in section 9.9.1 to make sure the filter wheel configuration is also set to SECONDARY.

9.9.3 In-position switch problems

The error messages "*in-position switch INACTIVE, expecting ACTIVE*" and "*in-position switch ACTIVE, expecting INACTIVE*" mean the filter wheel in-position switch was not in the correct state when the filter wheel arrived at its destination. This error can have any of the following causes, with the most likely cause listed first:

1. The filter wheel configuration is incorrect. This will cause the filter wheel to drive to the wrong locations, where the in-position switch will not be in the expected state. To test for this possibility, examine the filter wheel configuration with the vcinsFilterConfig panel. Are the POSENC values as expected? To recover from this problem, rebuild the software configuration or rebuild the entire software using the procedure described in section 9.1.2.

2. The filter wheel has lost its position. To test for this possibility, try driving the filter wheel to various locations (see section 8.8.1) and watch what happens on the "DIS1" device panel. Does the in-position switch change state when the filter wheel moves but appears to change state in the wrong place? To recover from this problem, redatum the filter wheel using the "ICS → Redatum" procedure on the ICS engineering panel. If the filter wheel consistently resets itself to the same wrong position, suspect a mismatch between the filter wheel configuration and home switch (see above).

3. The in-position switch is not working. Suspect this possibility if the in-position switch does not appear to change state when the filter wheel moves. This problem can be worked around by disabling the in-position switch checks. Shut down the software with "vcinsStop". Bring up the expert startup panel "vcinsStartupDev" and uncheck the "Use in-position switches" button (which changes the INS.FILT1.USESW configuration parameter) and restart the software. The switch needs to be repaired at the next available opportunity.

4. The filter wheel is broken. This is bad. To test for this possibility try making some exposures, move the filter wheel a short distance (using the "Enc" option on the engineering screen, see section 8.8.1), then repeat the exposures. If the exposures look the same (i.e. filter wheel in the same place) then the filter wheel may be broken. Check that the motor control cable hasn't become detached. NOTE: It isn't possible to tell whether the filter wheel is moving by looking at the motor controller status. The motor steps display reports the pulses sent to the motor, not the position of the motor itself, and it will happily update even when the motor cable is detached. The in-position switches give positive feedback that the filter wheel really is moving.

9.10 Sensor device problems

9.10.1 Initialisation errors from the sensor devices

If any of the Lakeshore or Pfeiffer sensor devices fail to go ONLINE on startup they have not initialised properly. This error can have any of the following causes, with the most likely cause listed first:

1. The device in question is running in the wrong simulation mode (i.e. the device does not exist but is not in simulation mode). Use the "Devices → Simulation" option on

the ICS engineering screen to put the devices into their correct simulation mode, then try "ICS → ONLINE".

2. The device's cable has disconnected from the ISER card. Reconnect the cable and repeat the ONLINE command.

3. The device is powered off. Check the electronics cabinets and verify that all the sensor devices are switched on.

4. The device's configuration parameters have become corrupted. The configuration parameters include a set of "INIT" command strings which are executed when the device starts up (see section 6.4 on page 129). This problem can be corrected by rebuilding the configuration, as described in section 9.1.2.

5. The device has been configured manually into an incompatible state. This can happen if the device's configuration has been changed using its front panel. The VIRCAM software cannot communicate with a device that has been set into "local only" mode (i.e. that responds to the front panel only)[19]. If you suspect this problem, use the front panel to put the device into "local and remote" mode (the procedure is described in the documentation which came with the devices).

6. If there are none of the above problems, the device may be faulty. The software can continue running with a faulty device if that device is put into simulation mode. If there is a faulty Lakeshore device, disable the thermal control software (by setting configuration parameter "INS.THERMAL.ENABLE" to F) and restart the software. If possible, monitor the detector temperatures manually. If there is any doubt about the science detector temperature, start detector thermal protection procedures (section 4.13.1 on page 81).

Further diagnostics can be obtained by logging on to the LCU console and interacting directly with the devices, as shown in section 8.9 on page 153. It is especially useful to watch for messages on the LCU console when a device is put through an OFF,ONLINE command sequence.

9.10.1.1 Special note about Pfeiffer TPG256 device initialisation

The Pfeiffer TPG256 device has no "*reset*" command and cannot therefore be reset from the software. If the Pfeiffer device shows initialisation problems that cannot be overcome by the software, and the cable connection looks ok, try power cycling the device on its own.

9.10.2 Timeout errors from the Lakeshore devices

The occasional timeout error from a Lakeshore device is not a problem. The Lakeshores have the annoying habit of occasionally not replying to a command when they are busy. This can

[19] If this becomes a common problem, it is possible to lock out the Lakeshore front panels altogether by specifying "MODE 2" in the INIT configuration strings.

happen every few minutes without affecting the performance of the VISTA IR Camera. The thermal control software sends commands to the Lakeshore devices several times a minute, and if one command is missed because of a timeout, another command corrects the situation a few tens of seconds later.

If the commands to a Lakeshore device time out regularly every few seconds this is a more serious problem, since every command is failing. Again, further diagnostics can be obtained by interacting with the devices through the LCU, as shown in section 8.9 on page 153. Messages like this:

```
vcilsmHwRead (46): Nothing to read after 50 tries
```

mean that the named device (LSM2 in this case – the value in brackets is the file descriptor) has received a command but failed to send back a reply.

9.11 Thermal control problems

9.11.1 Thermal control software state will not go ONLINE

The ICS must be ONLINE before the thermal state will go ONLINE (try sending the ONLINE command). Assuming the ICS is ONLINE, the other reasons why the thermal control software will not go ONLINE are as follows:

- If the thermal state is DISABLED, thermal control has been disabled by setting the INS.THERMAL.ENABLED configuration parameter to “F”. Reset it back to “T”.
- If the thermal state is INVALID, one or more of the temperature readings are invalid. This suggests a problem with the Lakeshore sensor devices. Follow the suggestions in section 9.10.

9.11.2 Thermal control software is in the wrong substate

Has the software recently been restarted, or has the warmup or cooldown operation only just begun? The thermal control software has to go through the state transitions shown in Figure 3 on page 30. When restarted, the software begins in the AMBIENT state, moves to COOLDOWN when the coolers or optical bench are below a certain temperature threshold and then moves either to the OPERATIONAL state or the WARMUP state. The transition between COOLDOWN and WARMUP takes the longest because the software has to wait for the coolers to warm or cool by a defined temperature difference (INS.THERMAL.CLD.DELTA). This period lengthens towards the end of a warmup or cooldown when the cooler temperatures are levelling off. The software may take several minutes to move into the correct state. Transitions to the WARMUP or COOLDOWN state can be speeded up by setting the warmup or cooldown triggers, as described in section 4.15.2 on page 84.

9.11.3 Thermal control software will not respond to a warmup or cooldown trigger

Is the thermal control software in the ONLINE state and in the correct substate? The software will only respond to a COOLDOWN trigger when in the AMBIENT or WARMUP states, and will only respond to a WARMUP trigger in the OPERATIONAL or COOLDOWN states.

The thermal control software only checks the trigger flags every few seconds. Wait a minute and it should have moved into the correct substate. If it has not moved into the correct substate after a few minutes, consider taking manual control as described below.

9.11.4 Thermal control software changes state unexpectedly

If the thermal control software changes state unexpectedly there is likely to be an underlying hardware fault. Using the cooldown or warmup trigger will not solve the problem until the underlying hardware fault is resolved. The software monitors the cooler temperatures and the optical bench top temperature and changes state when these temperatures warm up or cool down. The FPA thermal plate, detector and mean cryostat temperature is used to decide whether the camera has reached the OPERATIONAL or AMBIENT state.

- If the software state changes unexpectedly from OPERATIONAL to WARMUP or COOLDOWN to WARMUP, check the status of the coolers. One or more of the coolers may have started to warm up. If this happens, check the state of the compressors. If one cooler fails it is possible to carry on without it. The software can be programmed to use the minimum cooler temperature by setting the CLD.USEMEAN parameter (using the vcinsThermalConfigDev panel).
- If the software changes unexpectedly from AMBIENT to COOLDOWN or WARMUP to COOLDOWN it is possible that someone has switched on a cooler or has begun to cool down the camera with liquid nitrogen. If this isn't true, check the cooler and optical bench top temperature readings — there may be a fault in the temperature sensors.
- If the software fails to reach the AMBIENT or OPERATIONAL state after several hours, check the FPA thermal plate, detector and mean cryostat temperature readings — there may be a fault in one of them.

Once the underlying hardware fault is resolved, the software can be returned to the correct state early using the cooldown or warmup trigger; otherwise it will detect the change in temperature itself after some time has elapsed.

9.11.5 Thermal control software is not heating/cooling the detectors as expected

The thermal control software operates on long timescales. It may be several minutes, or even a few hours, before you can see the effect of the software on the instrument temperature. The software may not be heating or cooling the detectors at the moment because of the built in protection mechanisms:

- The FPA target temperature is not allowed to ramp up or down faster than the limit defined in INS.THERMAL.FPA.MXGRD parameter and the RAMP command used to initialise the Lakeshore 332 device. The FPA target shown on the thermal control panel may not match that shown on the LSC1 device panel because the device's target temperature is still ramping to meet the new FPA target temperature.

- Compare the FPA target temperature with the current mean detector temperature. The FPA target and the detectors are not allowed to exceed the temperature difference defined in the INS.THERMAL.FPA.MXDIFF parameter.

The software is more likely to run into one of the protection limits if the temperatures have been controlled manually before restarting the software.

If you have waited a long time, have checked the long-term temperature trends and are certain that something is wrong with the thermal control, it is possible to take manual control by disabling thermal control with

```
% msgSend "" vciControl STANDBY ""
% vcinsStop -proc ICS
% ctooConfigSet VIRCAM INS.THERMAL.ENABLE F
% vcinsStart -proc ICS
% msgSend "" vciControl ONLINE ""
```

and then selecting "`Engineering→Manual temperature control`" from the thermal control panel. You can then update the LSC1 target temperature manually. It is also possible to define target temperatures on the front panel of the Lakeshore 332 device. (The RAMP setting still applies even to manual control.)

9.12 Plotting problems

The plot windows created from the "Tools→Plotting" menu of the ICS engineering panel (section 4.15.3 on page 86) use the plotServer process. The process is limited to 5 simultaneous plot windows. The plotServer process is sensitive to the creation, deletion and reconfiguratuion of the plot windows and it may eventually die or hang up. If the plot windows no longer appear, or stop updating, the plotServer may have died. The best way to recover from this problem is to restart the wvcam environment:

```
% vccEnvStop -e wvcam
% vccEnvStart -e wvcam
```

However, if an environment restart is not convenient, because you are taking data or don't want the collection of sensor data to be interrupted, the plotServer process can be restarted manually with the command

```
% plotServer -s 16000000 -n 5 &
```

9.13 Real-time display problems

The real-time display windows, which display the latest IRACE data, use the rtdServer process. This process can sometimes hang up in a similar manner to the plotServer process. . The best way to recover from this problem is to restart the wvcam environment:

```
% vccEnvStop -e wvcam
% vccEnvStart -e wvcam
```

Note also that only one vcrtd display can be running at any one time. To start multiple displays, use the rtd command instead.

9.14 Problems when shutting down

9.14.1 MIDAS processes are not stopped

If a twilight sky observation template, or one of the technical templates, are executed, the VIRCAM software may start a MIDAS background process, which creates the MIDAS message window. This process is not stopped when the "vcinsStop" command is executed. Instead, a message saying, "*The following MIDAS processes have been started, and are not normally stopped by vcinsStop*" is reported. This simply an informational message. The MIDAS message window can be left running, but it is wise to shut it down before logging out of the workstation. If the logout process ends up being suspended, stopping the MIDAS window will let you log out.

10 ERROR DEFINITIONS

Error definitions are contained within each software module in the "ERRORS" subdirectory. The help information associated with each error may be found inside files in the "ERRORS/HELP" subdirectory. The following sections summarise.

10.1 ICS errors

10.1.1 ICS server errors

10.1.1.1 vciERR_ALIGNP — Failed to read value of filter ALIGNP keyword.

Unable to read the value of the INS.FILT1.ALIGNP configuration keyword. Check it is set in the "vcmcfgINS.cfg" configuration file and defined in the dicVIRCAM_ICS dictionary. Its value should be visible on the "vcinsFilterConfig" panel.

Make sure the commands "icbConfigSet VIRCAM" and "vciConfigSet" have been executed since the last time the software configuration was changed.

If the keyword does exist, the software configuration may have become corrupted. Try rebuilding the vcmcfg module and using the command "icbConfigSet VIRCAM" to restore the software configuration to its default. Then reset the software with OFF, ONLINE.

Note: The VISTA IR filter wheel uses separate ALIGNP and ALIGNS keywords instead of ALIGN, since there are PRIMARY and SECONDARY reference switches.

10.1.1.2 vciERR_ALIGNS — Failed to read value of filter ALIGNS keyword.

Same as above, except ALIGNS is the problem keyword.

10.1.1.3 vciERR_CONFIG — Failed to read value of configuration keyword

Unable to read the specified configuration keyword. Check the keyword is set in the "vcmcfgINS*.cfg" or "vcmcfgICS*.cfg" configuration files and is defined in the dicVIRCAM_ICS dictionary.

If the keyword does exist, the software configuration may have become corrupted. Try rebuilding the vcmcfg module and using the command "icbConfigSet VIRCAM" to restore the software configuration to its default. Then reset the software with OFF, ONLINE.

10.1.1.4 vciERR_CONFIGN — Failed to read value of enumerated configuration keyword

Unable to read the specified enumerated configuration keyword. Check the keyword is set in the "vcmcfgINS*.cfg" or "vcmcfgICS*.cfg" configuration files and is defined in the dicVIRCAM_ICS dictionary.

Is the enumerated prefix for the keyword out of range? For example, are there 26 filter positions and the keyword INS.FILT1.POSENC27 being requested?

If the keyword does exist, the software configuration may have become corrupted. Try rebuilding the vcmcfg module and using the command "icbConfigSet VIRCAM" to restore the software configuration to its default. Then reset the software with OFF, ONLINE.

10.1.1.5 vciERR_FILTER_UNKNOWN — Filter name not recognised

The name of the requested filter is not recognised. Please check the filter name has been correctly entered. Otherwise check the "vcmcfgINS.cfg" and "vcmcfgICS_filters.cfg" configuration files to make sure such a filter is available and is installed in the instrument. Try the "vcinsFilterConfig" panel - is your filter mentioned? If the displayed configuration does not correctly describe what is installed in the filter wheel, the configuration may be corrected with the "vcinsFilterConfigDev" panel (the LCU will need to be reinitialised).

Any filters requested by templates should have been installed within the instrument (using the "vciLoader" script) before the observations are scheduled.

10.1.1.6 vciERR_INPOS_NOT_ACTIVE — In-position switch INACTIVE and expecting ACTIVE

The filter wheel has completed its second in-position switch test move and has found the in-position switch status is not ACTIVE, as it should have been.

Make sure the FILT1 and DIS1 devices are not in simulation mode.

Check the "vcinsFilterConfig" panel. Is the filter wheel configuration correct? This error might happen if the filter wheel is moving to the wrong place - do exposures look vignetted by the filter wheel?

Look at the "DIS1" device while the filter wheel is moving. Are the switches changing state? If so then suspect a configuration problem. If not then suspect a switch fault or a filter wheel fault. (Try moving the filter wheel a short distance and see if the vignetting pattern on exposures changes).

NOTE: The motor step count only counts the pulses being sent to the filter wheel motor. Only the switches and the exposure vignetting pattern give feedback on the actual motion of the filter wheel.

10.1.1.7 vciERR_INPOS_NOT_INACTIVE — In position switch ACTIVE and expecting INACTIVE

The filter wheel has completed its first in-position switch test move and has found the in-position switch status is not INACTIVE, as it should have been.

Make sure the FILT1 and DIS1 devices are not in simulation mode.

Check the "vcinsFilterConfig" panel. Is the filter wheel configuration correct? This error might happen if the filter wheel is moving to the wrong place - do exposures look vignetted by the filter wheel?

Look at the "DIS1" device while the filter wheel is moving. Are the switches changing state? If so then suspect a configuration problem. If not then suspect a switch fault or a filter wheel fault. (Try moving the filter wheel a short distance and see if the vignetting pattern on exposures changes).

NOTE: The motor step count only counts the pulses being sent to the filter wheel motor. Only the switches and the exposure vignetting pattern give feedback on the actual motion of the filter wheel.

10.1.1.8 vciERR_NOHELP — General purpose error message

A miscellaneous error message for which there is no additional help.

10.1.1.9 vciERR_REFSW_PRIMARY — Configuration mismatch

The filter wheel software is configured (with INS.FILT1.REFSW) to use the SECONDARY reference switch, but the digital I/O card is reporting that the LCU has the PRIMARY reference switch selected for input.

Please select the SECONDARY reference switch input on the back of LCU or change the INS.FILT1.REFSW value to "PRIMARY" in the "vcmcfgINS.cfg" configuration file. (NOTE: The SECONDARY switch is intended to be used only if the PRIMARY switch is broken).

This error might also happen if the digital I/O card is giving false readings. Did it initialise successfully?

10.1.1.10 vciERR_REFSW_SECONDARY — Configuration mismatch

The filter wheel software is configured (with INS.FILT1.REFSW) to use the PRIMARY reference switch, but the digital I/O card is reporting that the LCU has the SECONDARY reference switch selected for input.

Please select the PRIMARY reference switch input on the back of LCU or change the INS.FILT1.REFSW value to "SECONDARY" in the "vcmcfgINS.cfg" configuration file.

(NOTE: The SECONDARY switch is intended to be used only if the PRIMARY switch is broken).

This error might also happen if the digital I/O card is giving false readings. Did it initialise successfully?

10.1.1.11 vciERR_REPLY — Error reply from LCU

The LCU has rejected a command and returned the error message shown. Are the command parameters valid? Is the LCU environment in the correct state, and are all the devices functioning correctly? Check the LCU environment with the command "vccEnvCheck -e lvcics1". Try rebooting the LCU.

10.1.1.12 vciERR_SETUP_CMD — Failed to send SETUP command to LCU

The ICS server failed to send a SETUP command to the LCU environment. Is the LCU environment running (VME crate switched on) and in the ONLINE/IDLE state? Are the command parameters valid? Check the LCU environment with the command "vccEnvCheck –e lvcics1". Try rebooting the LCU.

10.1.1.13 vciERR_SETUP_FLT — Failed to set up command FILTER for LCU

The ICS server failed to define a message filter to manage the communication of the specified command to the LCU. Are the command and its parameters valid? This error might be caused by a programming error or a resource problem on the workstation.

Is the LCU environment running (VME crate switched on) and in the ONLINE/IDLE state? Check the LCU environment with the command "vccEnvCheck -e lvcics1". Try rebooting the LCU.

10.1.1.14 vciERR_TIMEOUT — Timeout while waiting for reply from LCU

A reply was not received within the defined timeout period after sending the specified command to the LCU environment. Is the timeout period too short?

An occasional failure might occur if the LCU is overworked (e.g. sensor polling frequency set too high). The Lakeshore devices will occasionally fail to reply to a command when they are busy or in the wrong state.

Frequent failures would suggest a more serious problem. Check the LCU environment with the command "vccEnvCheck –e lvcics1". Try rebooting the LCU.

10.1.2 Lakeshore 218 device errors

10.1.2.1 vcilsmERR_LSM — Failed to read status from Lakeshore 218 monitor

The software has failed to get a temperature reading from a Lakeshore 218 temperature monitor. This can be caused by any of the following problems:

1. An intermittent failure of the Lakeshore to respond to a command. (This is a feature of the Lakeshore device).
2. The RS232 cable has become disconnected.
3. The device has lost power or has developed a fault.
4. Someone has been playing with the keys on the front panel.

To recover from the four circumstances:

1. Repeat the status read. The Lakeshore should eventually respond with a reading.
2. Reconnect the RS232 cable. The device should work again.
3. Power on the device and/or execute an OFF/ONLINE command sequence to reset the device. If there are further problems the device may be faulty.
4. Go and shout at whoever messed about with the device, then proceed as (3). Make sure the device is not in local mode.

10.1.2.2 vcilsmERR_LSM_COMMS — Failed to communicate with Lakeshore 218 monitor

The software has failed to open an RS232 communication link with a Lakeshore 218 temperature monitor. To recover:

1. Check the RS232 cable is connected.
2. Check the Lakeshore 218 device is powered on.
3. Check the front panel of the device and make sure it has not been set into local mode.
4. Check the status of the ISER12 serial card (especially if more than one device has reported an RS232 communication error).
5. Has the software configured the correct baud rate, etc...?

10.1.2.3 vcilsmERR_LSM_INIT — Failed to initialize Lakeshore 218 monitor

The software has failed to initialise a Lakeshore 218 temperature monitor. The device has failed its self-test, or has responded with an error while attempting to execute one of the commands defined in the INIT configuration sequence. This error is common when a device is removed and the software restarted without setting the device into simulation mode. To recover:

1. Check the RS232 cable is connected properly and the device is powered up.
2. Check the vcmcfgICS_sensors.cfg configuration file and make sure the INIT commands defined for the "lsmx" devices are sensible.
3. Check the front panel and ensure the device has not been set into local mode.
4. Try repeating the OFF/ONLINE command sequence and, if that doesn't work, try power cycling the device. If there are repeated problems the device or the RS232 cable may be faulty.

10.1.2.4 vcilsmERR_LSM_RESET — Failed to reset Lakeshore 218 monitor

The software has failed to reset the Lakeshore 218 temperature monitor. To recover:

1. Check the RS232 cable is connected properly and the device is powered up.
2. Check the front panel and ensure the device has not been set into local mode.
3. Try power cycling the device and repeating the OFF/ONLINE command sequence. If there are repeated problems the device may be faulty.

10.1.2.5 vcilsmERR_LSM_SETUP — Failed to setup Lakeshore 218 monitor

The software has failed to update the alarm and relay set points for a Lakeshore 218S temperature monitor. An attempt to send the appropriate commands to the device has failed. To recover:

1. Try repeating the SETUP command. SETUP command failures are often intermittent (as the Lakeshore devices occasionally fail to reply to a command).
2. If the failures are continuous, check the RS232 cable is connected properly and the device is powered up. Do the alarms displayed on the device's front panel look sensible? Is the device configured to accept remote commands (it should not be in local mode)?
3. Connect to the LCU console and use the command:

```
vcilsmServerShow
```

to identify the file descriptor belonging to the device; then the commands:

```
vcilsmHwConsole fd, "ALARM? 1"
vcilsmHwConsole fd, "ALARMST? 1"
vcilsmHwConsole fd, "RELAY? 1"
vcilsmHwConsole fd, "RELAYST?"
```

to find out if the device can accept commands and to query the current alarm and relay configuration.

4. Try executing an OFF/ONLINE command sequence and repeat the SETUP. If that doesn't work, try power cycling the device. If there are repeated problems the device or the RS232 cable may be faulty.

10.1.2.6 vcilsmERR_LSM_TEST — Lakeshore 218 monitor self test failed

The Lakeshore 218 temperature monitor failed its self-test on startup. This error is common when a device is removed and the software restarted without setting the device into simulation mode. To recover:

1. Check the RS232 cable is connected properly to the device.

2. Check the Lakeshore 218 device is powered on.

3. Test the device from its front panel. If the device fails often it may be faulty. If the device is ok, check the quality of the RS232 cable.

10.1.3 Lakeshore 332 device errors

10.1.3.1 vcilscERR_LSC — Failed to read status from Lakeshore 332 controller

Same as for the Lakeshore 218.

10.1.3.2 vcilscERR_LSC_COMMS — Failed to communicate with Lakeshore 332 controller

Same as for the Lakeshore 218.

10.1.3.3 vcilscERR_LSC_INIT— Failed to initialize Lakeshore 332 controller

Same as for the Lakeshore 218.

10.1.3.4 vcilscERR_LSC_RESET— Failed to reset Lakeshore 332 controller

Same as for the Lakeshore 218.

10.1.3.5 vcilscERR_LSC_SETUP — Failed to setup Lakeshore 332 controller

The software has failed to update the set points for the Lakeshore 332 temperature controller. An attempt to send the appropriate commands to the device has failed. To recover:

1. Often this error is caused by an intermittent failure of the Lakeshore to reply to a command. If this is the case, repeat the command.

2. If there are repeated problems, check the RS232 cable is connected properly and the device is powered up. Do the set points displayed on the device's front panel look sensible? Is the device configured to accept remote commands (it should not be in local mode)?

3. Connect to the LCU console and use the commands

   ```
   vcilscHwConsole "SETP 1, 100"
   vcilscHwConsole "SETP 2, 100"
   ```

   to find out if the device can accept new set points manually.

4. Try executing an OFF/ONLINE command sequence and repeat the SETUP. If that doesn't work, try power cycling the device. If there are repeated problems the device or the RS232 cable may be faulty.

10.1.3.6 vcilscERR_LSC_TEST — Lakeshore 332 controller self test failed

Same as for the Lakeshore 218.

10.1.4 Pfeiffer TPG 256 device errors

10.1.4.1 vcitpgERR_TPG — Failed to read status from Pfeiffer vacuum gauge

The software has failed to get a vacuum pressure reading from the Pfeiffer TPG256 vacuum gauge. This can be caused by the following problems:

1. The RS232 cable has become disconnected.

2. The device has lost power or has developed a fault.

3. Someone has been playing with the keys on the front panel.

To recover from the three circumstances:

1. Reconnect the RS232 cable. The device should work again.

2. Power cycle the device to reset it, then execute an ONLINE command. If there are further problems the device may be faulty.

3. Go and shout at whoever messed about with the device, then proceed as (2).

10.1.4.2 vcitpgERR_TPG_COMMS — Failed to open communication to Pfeiffer vacuum gauge

The software has failed to open an RS232 communication link with the Pfeiffer TGP 256 vacuum gauge. To recover:

1. Check the RS232 cable is connected.

2. Check the Pfeiffer device is powered on. NOTE: The Pfeiffer device sometimes fails to recognise its RS232 link when it is powered up before the LCU. This problem can be corrected by power cycling the device while the LCU is powered up.

3. Check the status of the ISER12 serial card (especially if more than one device has reported an RS232 communication error).

4. Has the software configured the correct baud rate, etc...?

10.1.4.3 vcitpgERR_TPG_INIT — Failed to initialize Pfeiffer vacuum gauge

The software has failed to initialise the Pfeiffer TPG256 vacuum gauge. The device has failed its self-test, or has responded with an error while attempting to execute one of the commands defined in the INIT configuration sequence. This error is common when a device is removed and the software restarted without setting the device into simulation mode. To recover:

1. Check the RS232 cable is connected properly and the device is powered up. NOTE: The Pfeiffer device sometimes fails to recognise its RS232 link when it is powered up before the LCU. This problem can be corrected by power cycling the device while the LCU is powered up.

2. Check the vcmcfgICS_sensors.cfg configuration file and make sure the INIT commands defined for the "vac1" device are sensible.

3. Try repeating the OFF/ONLINE command sequence and, if that doesn't work, try power cycling the device. If there are repeated problems the device or the RS232 cable may be faulty.

10.1.4.4 vcitpgERR_TPG_SENSORS — Failed to switch on sensors for Pfeiffer TPG256 vacuum gauge

The Pfeiffer vacuum gauge reported an error when an attempt was made to switch on the sensors using the SEN command. To recover:

1. Check the serial connection to the device. If all else fails, the sensor may be switched on manually by connecting to the LCU console and typing the command

```
vcitpgHwConsole "SEN ,1,0,0,0,0,0"
```

10.1.4.5 vcitpgERR_TPG_TEST — Pfeiffer vacuum gauge has failed its self-test

The Pfeiffer TGP 256 vacuum gauge has failed its self-test. This error is common when a device is removed and the software restarted without setting the device into simulation mode. To recover:

1. Check the RS232 cable is connected.
2. Check the Pfeiffer device is powered on.
3. Test the device from its front panel. If the device fails often it may be faulty. If the device is ok, check the quality of the RS232 cable.

10.1.5 Heart Beat device errors

10.1.5.1 vcihbERR_PULSE — Heart Beat device failed to generate pulse

The heart beat device failed to generate a pulse signal on the digital I/O board. Please check digital I/O hardware and configuration.

NOTE: The heart beat device assumes the Acromag digital I/O device is available. If the digital I/O device is not present, the heart beat device *must* be used in simulation mode. If the digital I/O device is put into simulation mode, so must the heart beat device.

10.1.5.2 vcihbERR_PULSE_INIT — Failed to initialise heart beat device

```
The heart beat device failed to initialise. Please check the digital I/O
hardware and configuration.
```

NOTE: The heart beat device assumes the Acromag digital I/O device is available. If the digital I/O device is not present, the heart beat device *must* be used in simulation mode. If the digital I/O device is put into simulation mode, so must the heart beat device.

10.2 DCS errors

See the IRACE-DCS User Manual, [RD39].

10.3 OS errors

For BOSS errors see the BOSS user manual, [RD48].

10.3.1.1 vcoERR_ADD_KW — Failed to add FITS keyword to file

A FITS error has occurred while attempting to write the specified keyword to the specified file. Does the file exist? Is the file write-protected? Is the disk full?

10.3.1.2 vcoERR_BAD_NEWOFF — Invalid new offset sequence

An invalid offset sequence has been specified. The number of offsets must be a positive integer.

10.3.1.3 vcoERR_BAD_NEXTOFF — Failed to add new step to offset sequence

An attempt to add a new step to an offset sequence has been rejected. The offset sequence does not exist - a programming error.

10.3.1.4 vcoERR_BAD_SCALE — Bad plate scale

The plate scale value is invalid. This has been trapped to avoid a divide by zero.

10.3.1.5 vcoERR_POLE — Declination too close to the celestial pole

The telescope offset calculation has dropped into a "best effort" mode because the telescope pointing is too close to the celestial pole and dividing by cos(dec) would cause a divide by zero.

10.3.1.6 vcoERR_SLALIB — Error reported by slalib function

slalib is a collection of utilities for making astronomical calculations, such as coordinate conversion. The named function has returned an error code. The error message should also contain a brief description of the error.

The most likely cause of an error from slalib is faulty data input; for example requesting a tangent plane projection for an object more than 90 degrees from the tangent point.

10.3.1.7 vcoERR_TIF_QUERY — Failed to get parameter from TCS via tif

A call to the TCS tifGetByName function has failed. The most likely cause of this error is a problem with the TCS database. Is the TCS environment running? Are any items in the TCS database flagged as having bad quality (they might be shown with a dark grey background when displayed on a GUI)?

Make sure the TCS is available and ONLINE.

10.4 HOWFS errors

10.4.1.1 vchoiaERR_BADSIZE — Bad data size

The specified data frame has an illegal size.

10.4.1.2 vchoiaERR_FILE_LOAD — Image file could not be loaded

The specified image file could not be loaded. Is the file name valid? Does the file exist? Is the file readable? Is the file in the correct format?

- LOWFS files must contain two IMAGE extensions, separately containing the pre-focal and post-focal plane images.

- HOWFS files must contain one IMAGE extension, containing both the pre-focal and post-focal plane images.

10.4.1.3 vchoiaERR_FITS — FITS error

A FITS error has occurred. This is usually caused by a failure to read from or write to a FITS file. The image analysis software expects to receive a file in a specific format:

- LOWFS files must contain two IMAGE extensions, separately containing the pre-focal and post-focal plane images.

- HOWFS files must contain one IMAGE extension, containing both the pre-focal and post-focal plane images.

A file read error might have one of the following causes:

- The file does not exist. If necessary, recreate it.
- The exists but is protected against read access. Unprotect it.
- The file exists and is readable but is in the wrong format.

Choose a different file or create a new one.

A file write error most likely means the intended file already exists. Delete the file and try again.

10.4.1.4 vchoiaERR_FITS_GETKW — Failed to read keyword from FITS header

The specified keyword could not be read from the FITS header. Does the keyword exist? Is the keyword in the correct format? Does the file allow read access?

10.4.1.5 vchoiaERR_FITS_HDU_NUM — File contains wrong number of HDUs

The specified file does not contain the expected number of HDUs. See file format description above.

10.4.1.6 vchoiaERR_FITS_HDU_TYPE — Specified HDU is of wrong type

The specified HDU is not of the expected type. See file format description above.

10.4.1.7 vchoiaERR_FITS_OPEN — Failed to open FITS file

The specified FITS file could not be opened. See the FITS error description above.

10.4.1.8 vchoiaERR_FITS_SETKW — Failed to write keyword to FITS header

The specified keyword could not be written to the FITS header. Is the keyword specified in the dictionary? Is the keyword being written in the correct format? Is the file open for write access?

10.4.1.9 vchoiaERR_GET_ATTR — Database read error

An error has occurred while attempting to read from the online database. This kind of error can happen during development and testing where the software contains a mistake in the name of a database attribute. If a database error happens during normal operation, something more serious has gone wrong. Possible causes are:

1. The environment to which the database belongs is no longer running. Check the environment with vccEnvCheck. If necessary, check the named attribute exists in the online database using ccseiDb.

2. The database manager has stopped running or has run into problems. The best way out of this problem is to stop and restart the environment.

10.4.1.10 vchoiaERR_NOT_IDLE — Server not IDLE

Only the CHECK, STOP, ABORT, STATE, STATUS and EXIT commands may be sent to vchoiaServer while it is BUSY. All other commands can only be executed when IDLE.

Wait for the image analysis to finish and try again.

10.4.1.11 vchoiaERR_NOT_ONLINE — Server is not ONLINE

The vchoiaServer must be in the ONLINE state before it can execute any setup or image analysis commands.

Send the ONLINE command and try again.

10.4.1.12 vchoiaERR_SET_ATTR — Database write error

An error has occurred while attempting to read from the online database. This kind of error can happen during development and testing where the software contains a mistake in the name of a database attribute. If a database error happens during normal operation, something more serious has gone wrong. Possible causes are:

1. The environment to which the database belongs is no longer running. Check the environment with vccEnvCheck. If necessary, check the named attribute exists in the online database using ccseiDb.

2. The database manager has stopped running or has run into problems. The best way out of this problem is to stop and restart the environment.

10.4.1.13 vchoiaERR_THREAD_CANCEL — Failed to cancel thread

The pthread_cancel() function used to abort the image analysis thread has returned a failure. The most likely cause is that the vchoiaServer has run out of system resources.

This error is fatal. The vchoiaServer must be shut down and restarted.

10.4.1.14 vchoiaERR_THREAD_CREATE — Failed to create thread

The pthread_create() function used to create the image analysis thread has returned a failure. The most likely cause is that the vchoiaServer has run out of system resources.

This error is fatal. The vchoiaServer must be shut down and restarted.

10.4.1.15 vchoiaERR_WINDOW — Coordinate outside window

A supplied coordinate is outside the allowed window.

11 Reference

11.1 Programs

11.1.1 Command definition tables

- ./DCS/vcrtd/CDT/vcrtd.cdt
    - Command Definition Table for the VISTA Camera real-time display server.
- ./HOWFS/vchoia/CDT/vchoiaServer.cdt
- ./ICS/vcilsm/CDT/vcilsmServer.cdt
    - Command definition table for the "vcilsmServer" Lakeshore 218 device server process — copies the "ic0devServer" command definition table.
- ./ICS/vcilsc/CDT/vcilscServer.cdt
    - Command definition table for the "vcilscServer" Lakeshore 332 device server process — copies the "ic0devServer" command definition table.
- ./ICS/vcitpg/CDT/vcitpgServer.cdt
    - Command definition table for the "vcitpgServer" Pfeiffer TPG256 device server process — copies the "ic0devServer" command definition table.
- ./ICS/vcihb/CDT/vcihbServer.cdt
    - Command definition table for the "vcihbServer" heart beat device server process — copies the "ic0devServer" command definition table.
- ./ICS/vci/CDT/vciControl.cdt
    - Command Definition Table for the "vciServer" ICS server process — copies the "icbControl" command definition table.
- ./ICS/vci/CDT/vciSimControl.cdt
    - Command Definition Table for the "vciSimServer" simulated ICS server process — copies the "ic0lcuServer" command definition table.
- ./OS/vco/CDT/vcoControl.cdt
    - Command Definition Table for the "vcoServer" OS server process — based on the "osbControl" command definition table but adds VISTA-specific commands described in section 5.8 on page 96.

11.1.2 Servers

- ./HOWFS/vchoia/src/vchoiaServer.C
    - Main process for HOWFS command server.
- ./ICS/vci/src/vciControl.C
    - Main process for ICS command server. See also vciSERVER.C.
- ./ICS/vcilsm/src/vcilsmServer.c
    - Main process for Lakeshore 218 device server.
- ./ICS /vcilsc/src/vcilscServer.c
    - Main process for Lakeshore 332 device server.
- ./ICS /vcitpg/src/vcitpgServer.c

- Main process for Pfeiffer TPG256 device server.
- ./ICS /vcihb/src/vcihbServer.c
  - Main process for hearty beat device server.
- ./OS/vco/src/vcoControl.C
  - Main process for OS command server. See also vcoSERVER.C.

11.1.3 Special device drivers

- ./ICS/vcilsm/src/vcilsmHw.c
  - Contains functions for communicating with one or more Lakeshore 218 devices over their RS232 links. (Note the use of the "port" data structure and "fd" file descriptor to identify each individual device).
- ./ICS/vcilsc/src/vcilscHw.c
  - Contains functions for communicating with the Lakeshore 332 device over the RS232 link.
- ./ICS/vcitpg/src/vcitpgHw.c
  - Contains functions for communicating with the Pfeiffer TPG256 device over the RS232 link.

11.2 Scripts

11.2.1 Startup and shutdown scripts

See section 4 on page 57.

- vcinsStartup
- vcinsStartupDev
- vcinsStart
- vcinsStop

11.2.2 Installation scripts

- vcinsInstallHook
  - Called by pkginBuild after the "CREATE_SCAN" step. Creates the HOWFS data directory and installs IRACE and TCS configuration files.
- vcinsLinuxHook
  - Obsolete script only needed for use with the APR2004 release.
- vcinsRootHook
  - Called by pkginBuild after the "CREATE_ROOTS" step. Ensures the "$INS_ROOT/SYSTEM/MISC/VISTA" directory exists.

11.2.3 Test scripts

- vcinsSelfTest
  - Self-test the entire VIRCAM instrument software. See section 7 on page 130.

11.2.4 Utility scripts

- vcinsCheckConfig
  - Development script used to check the uniqueness of all the MODULE keywords in the pkgin configuration files.
- vcinsCheckPermissions
  - Check the existence and permissions of the most important VIRCAM files prior to starting the software.
- vcinsCmmLast
  - Displays the latest version of all VIRCAM modules in the CMM repository.
- vcinsConfigSet
  - A short cut for “icbConfigSet VIRCAM”.
- vcinsDestroy
  - Destroy the contents of the $INTROOT and $INS_ROOT directories and replace with empty directories.
- vcinsEnvSet
  - Display the environment variables needed by the VIRCAM software to be defined at login.
- vcinsHelp
  - Display help information.
- vcinsSavePanel
  - Save a panel to a file containing a GIF image.
- vcinsVersionCheck
  - Send the VERSION command to all the VIRCAM command servers.

11.2.5 OS test scripts

- vco/test/vcoTest
  - Self-test the VIRCAM OS software. See section 7 on page 130.
- vco/test/vcoTestArchiver
  - Test the VIRCAM bossArchiver by sending it a specified ARF file.
- vco/test/vcoTestExposure
  - Make and merge an exposure will full debugging and logging enabled.
- vco/test/vcoTestPerformance
  - Test the performance of the VIRCAM OS and BOSS by making a large number of exposures.

11.2.6 OS utility scripts

- vcoseq/test/checkfits
  - Check that all the FITS files in a given list have the same number of header keywords.

11.2.7 ICS test scripts

- vci/test/vciTest

  - Self-test the VIRCAM instrument devices and ICS software. See section 7 on page 130.
- vci/test/vciTestINS_FILTER
  - High level filter assembly tests executed by vciTest.
- vci/test/vciTestLakeshoreSetup
  - Test the Lakeshore devices by sending them several consecutive SETUP commands.
- vci/test/vciTestReversalCCW
  - Filter wheel test script. See section 7.5 on page 134.
- vci/test/vciTestReversalCW
  - Filter wheel test script. See section 7.5 on page 134.
- vci/test/vciTestVelocities
  - Filter wheel test script. See section 7.5 on page 134.
- vci/test/vciThermalTestSim
  - Self-test the thermal control software in simulation mode.

11.2.8 ICS utility scripts

- vci/src/vciLogging
  - Start logging ICS sensor data (using vciLogTemperatures, vciLogThermalControl and vciLogVacuum).
- vci/test/vciCheckSwitch
  - Check the state of the filter wheel in-position switch.
- vci/test/vciDivide1000
  - Divide a number by 1000.
- vci/test/vciFilterFunc
  - Functions used by the filter wheel test scripts.
- vci/test/vciFindBacklash
  - Filter wheel backlash measurement script. See section 7.5 on page 134.
- vci/test/vciFindBearings
  - Filter wheel in-position bearings location script. See section 7.5 on page 134.
- vci/test/vciFindRandom
  - Filter wheel test script — goes to random positions. See section 7.5 on page 134.
- vci/test/vciFindReference
  - Filter wheel test script — counts number of steps in a revolution. See section 7.5 on page 134.
- vci/test/vciFindUserCCW
  - Filter wheel test script — goes to user defined positions. See section 7.5 on page 134.
- vci/test/vciFindUserCW
  - Filter wheel test script — goes to user defined positions. See section 7.5 on page 134.
- vci/test/vciLoadAll
  - Moves each filter in turn to the load position. See section 7.5 on page 134.

- vci/test/vciLoader
    - Moves a filter to the load position. See section 7.5 on page 134.
- vci/test/vciMakeFILTM
    - Generates a new vcinsFILTM.dbcfg motor configuration file from the specified drive parameters. See section 7.5 on page 134.
- vci/test/vciRemainder
- vci/test/vciRestoreFILTM
    - Restores the previous vcinsFILTM.dbcfg motor configuration file used before the last vciMakeFILTM command. See section 7.5 on page 134.
- vci/test/vciSetAmbient
    - Write an ambient temperature value into the sensor LCU database.
- vci/test/vciSetCoolers
    - Write cooler temperature values into the sensor LCU database.
- vci/test/vciSetCryostat
    - Write cryostat temperature values into the sensor LCU database.
- vci/test/vciSetDetectors
    - Write detector temperature values into the sensor LCU database.
- vci/test/vciSetFPA
    - Write a focal plane array temperature value into the sensor LCU database.
- vci/test/vciSimDevicesOffline
    - Simulate all sensor devices being offline by writing state values to the workstation database.
- vci/test/vciSimDevicesOnline
    - Simulate all sensor devices being online by writing state values to the workstation database.
- vci/test/vciSimSetAmbient
    - Write an ambient temperature value into the sensor workstation database.
- vci/test/vciSimSetCoolers
    - Write cooler temperature values into the sensor workstation database.
- vci/test/vciSimSetCryostat
    - Write cryostat temperature values into the sensor workstation database.
- vci/test/vciSimSetDetectors
    - Write detector temperature values into the sensor workstation database.
- vci/test/vciSimSetFPA
    - Write a focal plane array temperature value into the sensor workstation database.
- vci/test/vciTwoStep
    - Sets the INS.FILT1.TWOSTEP configuration keyword to T or F.
- vcmcfg/config/MakeRealFW
    - Invokes vciMakeFILTM to set motor configuration parameters for the real filter wheel.
- vcmcfg/config/MakeTestFW
    - Invokes vciMakeFILTM to set motor configuration parameters for the UKATC test filter wheel.

11.2.9 DCS test scripts

- vcd/test/vcdTest
    - Self-test the VIRCAM DCS software. See section 7 on page 130.

11.2.10 HOWFS test scripts

- vchoia/test/vchTest
    - Self-test the VIRCAM HOWFS software by processing specific loop-back data. See section 7 on page 130.
- vchoia/test/vchSoakTest
    - Soak test the VIRCAM HOWFS software by processing randomly-generated loop-back data. See section 7 on page 130.
- vchoia/test/vchTestBlurring
    - Test the effectiveness of the HOWFS internal blurring algorithm by generating test data with different amounts of blurring and analysing it with the same or a different blurring factor.
- vchoia/test/vchTestCameraA
    - Analyses the "A" set of data from the VISTA test camera.
- vchoia/test/vchTestCameraB
    - Analyses the "B" set of data from the VISTA test camera.
- vchoia/test/vchTestCameraF
    - Analyses the "F" set of data from the VISTA test camera.
- vchoia/test/vchTestCameraH
    - Analyses the "H" set of data from the VISTA test camera.
- vchoia/test/vchTestCameraR
    - Analyses the "R" set of data from the VISTA test camera.
- vcoseq/test/howfsAnastarTest
    - Test the HOWFS server by sending it a rapid sequence of ANASTAR and WAIT commands.
- vcoseq/test/howfsStateTest
    - Test the HOWFS server by sending it a rapid sequence of ONLINE, STANDBY and OFF commands.

11.2.11 HOWFS utility scripts

- vchoia/test/vchCmdAfterWait
    - Send a command to the HOWFS server after a specified time delay.
- vchoia/test/vchGenerateData
    - Generate test data with specified Zernike coefficients.
- vchoia/test/vchGenerateInFocusImages
    - Generate a series of in-focus images with varying degrees of seeing blur, to test the blurring algorithm.
- vchoia/test/vchMemCheck
    - Run the HOWFS server under watchful eye of the "valgrind" memory leak checker.

11.3 Include Files

All include files are private to VIRCAM.

11.4 Tcl Libraries

- vcoseq/src/vcoseqDR.tcl
  - Library of data reduction utilities, inherited from tplDR, [RD49].
- vcoseq/src/vcoseqHOWFS.tcl
  - Library of HOWFS server communication utilities.
- vcoseq/src/vcoseqICS.tcl
  - Library of ICS server communication utilities, inherited from tplICS, [RD49].
- vcoseq/src/vcoseqIRACE.tcl
  - Library of IRACE server communication utilities, inherited from tplIRACE, [RD49].
- vcoseq/src/vcoseqLib.tcl
  - Library of general purpose, high-level template script functions.
- vcoseq/src/vcoseqOBS.tcl
  - Library of OS server communication and observation control utilities, inherited from tplOBS, [RD49]..
- vcoseq/src/vcoseqTCS.tcl
  - Library of ICS server communication utilities, inherited from tplTCS, [RD49].

11.5 Configuration Files

11.5.1 OS

The following VIRCAM configuration files are installed in $INS_ROOT/SYSTEM/COMMON/CONFIGFILES. See the file “vcmcfg/config/README” for more information.

- vcmcfg/config/vcmcfgCONFIG.cfg
  - Master configuration file — includes all others.
- vcmcfg/config/vcmcfgFILTM.dbcfg
  - Filter wheel motor control configuration file, written by the motei utility (or by the vciMakeFILTM utility script).
- vcmcfg/config/vcmcfgFILTM.dbcfg_template
  - Template filter wheel motor control configuration file, used by the vciMakeFILTM utility script.
- vcmcfg/config/vcmcfgICS_filters.cfg
  - List of filters installed in the instrument. May be updated when the filters within the instrument are changed.
- vcmcfg/config/vcmcfgICS_sensors.cfg
  - Sensor configuration keywords.
- vcmcfg/config/vcmcfgICS_thermal.cfg
  - Thermal control configuration keywords.

- vcmcfg/config/vcmcfgINS.cfg
  - Top-level ICS configuration keywords, including the permanent configuration settings for the filter wheel.
- vcmcfg/config/vcmcfgINS_TEST.cfg
  - List of commands and responses for self-testing the ICS.
- vcmcfg/config/vcmcfgIRDCS.cfg
  - IRACE/DCS configuration keywords for the full IRACE system.
- vcmcfg/config/vcmcfgIRDCS_WSSIM.cfg
  - IRACE/DCS configuration keywords for defining IRACE in workstation simulation mode.
- vcmcfg/config/vcmcfgIcsSelfTest_1.ins
  - Setup file used by the ICS self-test script.
- vcmcfg/config/vcmcfgOS.cfg
  - Top-level OS configuration keywords.
- vcmcfg/config/vcmcfgOS_WCS.cfg
  - World coordinate system keywords, describing the layout of the detectors in the VIRCAM focal plane.
- vcmcfg/config/vcmcfgSTART.cfg
  - Default startup configuration keywords, defining the subsystems and panels to be started. These values can be redefined with the vcinsStartup or vcinsStartupDev panels.

11.5.2 DCS

The following IRACE configuration files are installed in $INS_ROOT/SYSTEM/MISC/IRACE.

11.5.2.1 DCS Configuration files

See the *IRACE-DCS User Manual*, [RD39], for more details.

- vcdcfg/config/RTDB.cfg
- vcdcfg/config/irtdCtrl.cfg
- vcdcfg/config/irtdMenu.cfg
- vcdcfg/config/rtdbMenu.cfg
  - Configuration files describing the plugins, menus and control widgets used by the rtdb real-time display tool. There are similar files in vcrtd/config.
- vcdcfg/config/vircam.cfg
  - System configuration file for the full IRACE system.
- vcdcfg/config/vircam.clk
  - IRACE clock sequence file for Raytheon 2048x2048 Virgo detectors.
- vcdcfg/config/vircam.dcf
  - Detector configuration file for the full IRACE system.
- vcdcfg/config/vircam.dsup
  - Default parameter setup for IRACE.
- vcdcfg/config/vircam1.v

- vcdcfg/config/vircam2.v
  - IRACE voltage files for Raytheon 2048x2048 VIRGO detector.
- vcdcfg/config/vircamSim.cfg
  - System configuration file for the workstation-simulated IRACE system (with 512x512 pixel data).
- vcdcfg/config/vircamSim.dcf
  - System configuration file for the workstation-simulated IRACE system (with 1024x1024 pixel data).

11.5.2.2 DCS Sequence files

- vcdcfg/src/VirgoDblCor_seq.tcl
  - IRACE sequencer program for an exposure made in standard double correlated sampling (Double) mode (using separate RESET and READ operations) with VISTA Virgo detectors, i.e.

```
Reset   Read1  Integrate      Read2
     +       +                    +
    /|      /|                   /|
   / |     / |   DET.DIT        / |
  /  |    /<-|---------------->/  |
 /   |   /   |                /   |
+----+-+-----+---------------+-----+
     ^ ^
     DET.DITDELAY
```

- vcdcfg/src/VirgoDblCorReadReset_seq.tcl
  - IRACE sequencer program for exposures made in FAST double correlated sampling mode (which uses the combined READ+RESET operation) with VISTA Virgo detectors, i.e.

```
   Reset    Read Integrate 1 ReadReset Read Integrate 2 ReadReset etc..
     +       +                    +       +                    +
    /|      /|                   /|      /|                   /|
   / |     / |   DET.DIT        / |     / |   DET.DIT        / |
  /  |    /<-|---------------->/  |    /<-|---------------->/  |
 /   |   /   |                /   |   /   |                /   |
+----+-+-----+---------------+-----+-+-----+--------------+-----+ etc..
     ^ ^                          ^ ^
     DET.DITDELAY                 DET.DITDELAY
```

- vcdcfg/src/VirgoUnCor_seq.tcl
  - IRACE sequencer program for exposures made in uncorrelated sampling (Uncorr) mode (with a single READ) with VISTA Virgo detectors, i.e.

```
Reset   Integrate          Read
     +                         +
    /|                        /|
   / |                       / |
  /  |       DET.DIT        /  |
 /   | <------------------>/   |
+----+-+------------------+----+
     ^ ^
     DET.DITDELAY
```

- vcdcfg/src/VirgoUnCorReadReset_seq.tcl
  - IRACE sequencer program for exposures made in uncorrelated sampling (Uncorr) mode (with a single READRESET) with VISTA Virgo detectors, i.e.

```
Reset   Integrate          ReadReset
     +                         +
    /|                        /|
   / |                       / |
  /  |       DET.DIT        /  |
 /   | <------------------>/   |
+----+-+------------------+----+
     ^ ^
     DET.DITDELAY
```

- vcdcfg/src/VirgoDelay_ssd.tcl
  - Sequence file which implements a time delay with a Raytheon Virgo detector.
- vcdcfg/src/VirgoFrame_ssd.tcl
  - Sequence file which implements a frame READ operation with a Raytheon Virgo detector.
- vcdcfg/src/VirgoFrameReadReset_ssd.tcl
  - Sequence file which implements a frame READ+RESET operation with a Raytheon Virgo detector.
- vcdcfg/src/VirgoReset_ssd.tcl
  - Sequence file which implements a RESET operation with a Raytheon Virgo detector.

11.6 Setup Files

11.6.1 REF files

The following files are installed in $INS_ROOT/SYSTEM/COMMON/SETUPFILES/REF.

- VIRCAM_gen.ref
  - Reference setup file for any general purpose maintenance template. Sets INS.MODE=IMAGING.

- VIRCAM_howfs.ref
    - Reference setup file for HOWFS mode. Sets INS.MODE=HOWFS.
- VIRCAM_howfs_cal_reset.ref
    - Reference setup file for bias observations made in HOWFS mode. Sets INS.MODE=HOWFS, DET.DIT=0.0, DET.NDIT=1.
- VIRCAM_img.ref
    - Reference setup file for IMAGING mode. Sets INS.MODE=IMAGING.
- VIRCAM_img_cal_reset.ref
    - Reference setup file for bias observations made in IMAGING mode. Sets INS.MODE=IMAGING, DET.DIT=0.0, DET.NDIT=1.
- VIRCAM_star1.ref
- VIRCAM_star2.ref
- VIRCAM_star3.ref
- VIRCAM_star4.ref
    - Example files containing guide star candidates for a VISTA pawprint.

11.6.2 PAF files

The following files are installed in $INS_ROOT/SYSTEM/MISC/VISTA.

- VIRCAM_HOWFS_HOJAXcen.paf
    - Parameters describing the properties of the HOJAXcen HOWFS filter (used for AIT only).
- VIRCAM_HOWFS_HOJAXny.paf
    - Parameters describing the properties of the HOJAXny HOWFS filter (used for AIT only).
- VIRCAM_HOWFS_HOJAXpy.paf
    - Parameters describing the properties of the HOJAXpy HOWFS filter (used for AIT only).
- VIRCAM_HOWFS_HOJcen.paf
    - Parameters describing the properties of the HOJcen HOWFS filter.
- VIRCAM_HOWFS_HOJcony.paf
    - Parameters describing the properties of the HOJcony HOWFS filter.
- VIRCAM_HOWFS_HOJcopy.paf
    - Parameters describing the properties of the HOJcopy HOWFS filter.
- VIRCAM_HOWFS_HOJexny.paf
    - Parameters describing the properties of the HOJexny HOWFS filter.
- VIRCAM_HOWFS_HOJexpy.paf
    - Parameters describing the properties of the HOJexpy HOWFS filter.
- VIRCAM_HOWFS_HOJeyny.paf
    - Parameters describing the properties of the HOJeyny HOWFS filter.
- VIRCAM_HOWFS_HOJeypy.paf
    - Parameters describing the properties of the HOJeypy HOWFS filter.
- VIRCAM_HOWFS_HOJoany.paf

- Parameters describing the properties of the HOJoany HOWFS filter.
- VIRCAM_HOWFS_HOJoapy.paf
  - Parameters describing the properties of the HOJoapy HOWFS filter.
- VIRCAM_HOWFS_TestCamera.paf
  - Parameters describing the properties of the HOWFS data obtained from the VISTA test camera (for engineering and testing only).

- VIRCAM_Jitter2d.paf
- VIRCAM_Jitter2u.paf
- VIRCAM_Jitter2x2.paf
- VIRCAM_Jitter3d.paf
- VIRCAM_Jitter3u.paf
- VIRCAM_Jitter3x3.paf
- VIRCAM_Jitter5p.paf
- VIRCAM_Jitter5x.paf
- VIRCAM_Jitter5x5.paf
  - Files describing jitter patterns which may be used by the VIRCAM_img_obs_paw or VIRCAM_img_obs_tile<N> templates (see also section 11.8).

- VIRCAM_JitterXTalk.paf
- VIRCAM_JitterXTalk_ALT.paf
  - Jitter patterns used by the VIRCAM_img_acq_crosstalk template.

- VIRCAM_Tile1_00.paf
- VIRCAM_Tile3nx.paf
- VIRCAM_Tile3px.paf
- VIRCAM_Tile6n.paf
- VIRCAM_Tile6s.paf
- VIRCAM_Tile6ss.paf
- VIRCAM_Tile6u.paf
- VIRCAM_Tile6z.paf
- VIRCAM_Tile6zz.paf
- VIRCAM_TileXTalk.paf
  - Files describing jitter patterns which may be used by the VIRCAM_img_obs_tile<N> template (see also section 11.8). The value of "N" must match the number of pawprints within the pattern (contained within the pattern file name, with the exception of VIRCAM_TileXTalk which defines 16 pawprints).

- VIRCAM_Ustep2.paf
- VIRCAM_Ustep2x2.paf
  - Files describing microstep patterns which may be used by the VIRCAM_img_obs_paw or VIRCAM_img_obs_tile<N> templates (see also section 11.8).

- VIRCAM_twilight.paf
    - The database of VIRCAM twilight sky fields.

11.7 Templates

11.7.1 HOWFS acquisition templates

11.7.1.1 VIRCAM_howfs_acq

```
Description
-----------
Acquire a HOWFS source. This template sets the instrument into HOWFS
mode and selects a HOWFS beam-splitting filter/position
combination. (Each HOWFS has a number of preset positions in the focal
plane where it can be used). It also points the telescope to a HOWFS
standard star (using a "preset"), specifying the sky coordinates of
the star, the required (X,Y) in the instrument focal plane and the
position angle of the rotator. If not specified, the position angle
defaults to orient the instrument Y axis to the north and X axis to
the west. Any position angle specified refers to the position angle at
the pointing centre (i.e. the meridian line of the TEL.TARG.ALPHA
should intersect the column of pixels at TEL.TARG.X at angle
TEL.ROT.OFFANGLE). If autoguiding and active optics correction are
required one guide star and two aO stars are specified.

Parameters
----------
Name                    Default          Description
----                    -------          -----------
Fixed:
INS.MODE                HOWFS            Instrument mode (FIXED).

Compulsory:
INS.FILTER.NAME          (none)          Name of HOWFS filter (must be
                                         contained in (FILTERS_HOWFS).
TEL.TARG.ALPHA           (none)          Target RA
TEL.TARG.DELTA           (none)          Target DEC
TEL.AG.START                 T           Observe with autoguiding (T or F)
TEL.AO.START                 T           Observe with aO (T or F)

Optional:
INS.SCIFILT.NAME         (none)          Name of science filter to be
                                         used to acquire the star.
TEL.TWEAK               T                Allow manual position adjustment
TEL.ROT.ENABLED         T                Rotator preset enabled
TEL.ROT.OFFANGLE        0.0              Camera sky position angle
                                         (-180 to +180 degrees).
TEL.TARG.EQUINOX        2000.0           Target equinox
TEL.TARG.ADDVELALPHA    0                Target drift in RA
TEL.TARG.ADDVELDELTA    0                Target drift in Dec
TEL.TARG.PMA            0                Target proper motion in RA
TEL.TARG.PMD            0                Target proper motion in Dec
TEL.TARG.EPOCH          2000             Target epoch
TEL.TARG.EPOCHSYSTEM    J                Target epoch system
```

```
TEL.AG.GUIDESTAR        SETUPFILE       Where to find guide stars
                                        (NONE, SETUPFILE, CATALOGUE)
TEL.AG.CONFIRM          F               Confirm each new guide star?
TEL.GS1.ALPHA           (none)          RA of guide star candidate 1
TEL.GS1.DELTA           (none)          Dec of guide star candidate 1
TEL.GS1.MAG             (none)          Magnitude of guide star candidate 1
 "   "   "                "                "       "       "
TEL.GS5.ALPHA           (none)          RA of guide star candidate 5
TEL.GS5.DELTA           (none)          Dec of guide star candidate 5
TEL.GS5.MAG             (none)          Magitude of guide star candidate 5

TEL.AO.AOSTARA          SETUPFILE       Where to find active optics star A
                                        (NONE, SETUPFILE, CATALOGUE)
TEL.AO.AOSTARB          SETUPFILE       Where to find active optics star B
                                        (NONE, SETUPFILE, CATALOGUE)
TEL.AO.CONFIRM          F               Confirm active optics before
starting?
TEL.AO.PRIORITY         HIGH            Active optics priority (LOW, NORMAL
                                        or HIGH)
TEL.AOSA1.ALPHA         (none)          RA of aO star A candidate 1
TEL.AOSA1.DELTA         (none)          Dec of aO star A candidate 1
TEL.AOSA1.MAG           (none)          Magnitude of aO star A candidate 1
 "   "   "                "                "       "       "
TEL.AOSA5.ALPHA         0.0             RA of aO star A candidate 5
TEL.AOSA5.DELTA         0.0             Dec of aO star A candidate 5
TEL.AOSA5.MAG           25.0            Magnitude of aO star A candidate 5
TEL.AOSB1.ALPHA         (none)          RA of aO star B candidate 1
TEL.AOSB1.DELTA         (none)          Dec of aO star B candidate 1
TEL.AOSB1.MAG           (none)          Magnitude of aO star B candidate 1
 "   "   "                "                "       "       "
TEL.AOSB5.ALPHA         0.0             RA of aO star B candidate 5
TEL.AOSB5.DELTA         0.0             Dec of aO star B candidate 5
TEL.AOSB5.MAG           25.0            Magnitude of aO star B candidate 5
```

```
Sequence
--------
If telescope interface is disabled then
   Pop up a warning box.
End if
Get parameters for HOWFS filter.
In parallel:
1) Set instrument mode to HOWFS.
   If INS.SCIFILT.NAME is defined then
      Put science filter into beam.
   Else
      Put HOWFS filter into beam.
   End if
2) Preset telescope to target.
   Adjust telescope focus for HOWFS filter.
   Calculate X,Y of pointing origin corresponding to HOWFS filter.
   Offset telescope to X,Y of pointing origin.
   {
      If autoguiding is enabled then
         If AG.CONFIRM is TRUE then
            Prompt operator to confirm autoguiding.
         End if
         Wait for autoguiding to start
      End if
      If active optics are enabled and AO.PRIORITY is > 0 then
         Wait for active optics to start
      End if
   }
Save the current target coordinates for future observation templates.
Display the expected location of the HOWFS star on the detectors.
If TEL.TWEAK is T then
   Prompt for the observer to tweak the telescope position manually.
   If not skipped then
      Update the saved target coordinates.
   End if
End if
__oOo__
```

11.7.1.2 VIRCAM_howfs_acq_domescreen

```
Description
-----------
This template sets the instrument into HOWFS mode and selects a HOWFS
beam-splitting filter. It also moves the telescope to point at the
flat-field screen in the dome (using a "preset"). Telescope tracking
is turned off and the required illumination level is defined. The
flat-field illumination source is switched on and allowed to
stabilize.

Parameters
----------
Name                    Default        Description
----                    -------        -----------
Fixed:
INS.MODE                HOWFS          Instrument mode (FIXED).
TEL.TARG.NAME           DOMEFLAT       Telescope target name (FIXED).
TEL.AG.START            F              No autoguiding (FIXED).
TEL.AO.START            F              No closed loop aO (FIXED).
TEL.ECS.FLATOFF         F              Do not switch flat-field off when
                                       finished (FIXED)?

Compulsory:
INS.FILTER.NAME         (none)         Name of HOWFS filter (must be
                                       contained in (FILTERS_HOWFS).
TEL.ECS.FFREQ           1              Telescope flat-field level (0-7).

Sequence
--------
If telescope interface is disabled then
   Pop up a warning box.
End if
In parallel:
1) Set instrument mode to HOWFS.
   Select HOWFS filter.
2) Preset telescope to DOMEFLAT (no tracking or autoguiding)
   Set flat-field illumination to TEL.ECS.FFREQ
Ask operator to confirm that ambient lights are turned off.
__oOo__
```

11.7.2 HOWFS calibration templates

11.7.2.1 VIRCAM_howfs_cal_dark

```
Description
-----------
This template makes one or more DARK exposures suitable for
calibrating HOWFS observations. The dark filter is selected and a
sequence of exposures made by the detector controller using the window
parameters needed for each of the specified HOWFS filters.

Prerequirements
---------------
None.

Data Products
-------------
DPR.CATG                TECHNICAL
DPR.TECH                IMAGE
DPR.TYPE                DARK

Parameters
----------
Name                    Default          Description
----                    -------          -----------
Fixed:
INS.FILTER.DARK         SUNBLIND         Name of DARK filter (FIXED)

Compulsory:
INS.FILTER.NAME         FILTERS_HOWFS    List of HOWFS filters (each must
                                         be in FILTERS_HOWFS)
DET1.DIT                10.0             Either: Detector integration time
                                         (seconds)
                                         Or: List of detector integration
                                         times corresponding to list of
                                         HOWFS filters.
Optional:
DET1.NDIT               1                Either: Number of detector
                                         integrations per exposure.
                                         Or: List of detector integrations
                                         per exposure corresponding to list
                                         of HOWFS filters.
DET1.NCORRS.NAME        Double           Detector readout mode (must be
                                         in NCORRS_RANGE)
SEQ.NEXPO               1                Number of DARK exposures for
                                         each HOWFS filter.
```

```
Sequence
--------
Select DARK filter.
For each HOWFS filter
   Select detector window corresponding to HOWFS filter
      (without actually selecting the filter).
   Select detector chip corresponding to HOWFS filter.
   Set detector exposure time and readout mode.
   {
      For exposure = 1 to SEQ.NEXPO
         Set WCS parameters to "pixel coordinates".
         Make exposure
      Next exposure
   }
   Store DARK calibration as
HOWFSDATA/VIRCAM_HOWFS_DARK_<filter>_<DIT>.fits
Next HOWFS filter
__oOo__
```

11.7.2.2 VIRCAM_howfs_cal_domeflat

```
Description
-----------
This template makes a flat-field exposure (or series of exposures)
suitable for calibrating HOWFS observations. A series of flat-field
exposures are made with the specified list of HOWFS filters, using the
detector window parameters associated with each filter.

Prerequirements
---------------
Telescope already pointing at flat-field target
(VIRCAM_howfs_acq_domescreen) with calibration source switched on.

Data Products
-------------
DPR.CATG                TECHNICAL
DPR.TECH                IMAGE
DPR.TYPE                FLAT,LAMP
```

```
Parameters
----------
Name                    Default         Description
----                    -------         -----------
Compulsory:
INS.FILTER.NAME         FILTERS_HOWFS   List of HOWFS filters (each must
                                        be in FILTERS_HOWFS)
DET1.DIT                10.0            Either: Detector integration time
                                        (seconds)
                                        Or: List of detector integration
                                        times corresponding to list of
                                        HOWFS filters.

Optional:
DET1.NDIT               1               Either: Number of detector
                                        integrations per exposure.
                                        Or: List of detector integrations
                                        per exposure corresponding to list
                                        of HOWFS filters.
DET1.NCORRS.NAME        Double          Detector readout mode (must be
                                        in NCORRS_RANGE)
TEL.ECS.FFREQ           1               Telescope flat-field level (0-7)
TEL.ECS.FLATOFF         T               Switch flat-field off when
finished?
SEQ.NEXPO               1               Number of flat-field exposures at
                                        each HOWFS filter.

Sequence
--------
For each HOWFS filter
   Select HOWFS filter.
   Adjust telescope focus for HOWFS filter.
   Select detector window corresponding to HOWFS filter.
   Select detector chip corresponding to HOWFS filter.
   Set detector exposure time and readout mode.
   {
      For exposure = 1 to SEQ.NEXPO
         Set WCS parameters to "pixel coordinates".
         Make exposure
      Next exposure
   }
   Store FLAT as HOWFSDATA/VIRCAM_HOWFS_FLAT_<filter>.fits
Next HOWFS filter
If TEL.ECS.FLATOFF is T then
   Switch off flat-field illumination
End if
__oOo__
```

11.7.2.3 VIRCAM_howfs_cal_reset

```
Description
-----------
This template makes one or more reset (aka BIAS) exposures suitable
for calibrating HOWFS observations. The dark filter is selected and a
reset/read (uncorrelated) sequence executed by the detector controller
using the window parameters needed for each of the specified HOWFS
filters.

Prerequirements
---------------
None.

Data Products
-------------
DPR.CATG                TECHNICAL
DPR.TECH                IMAGE
DPR.TYPE                BIAS

Parameters
----------
Name                    Default         Description
----                    -------         -----------
Fixed:
INS.FILTER.DARK         SUNBLIND        Name of DARK filter (FIXED)
DET1.DIT                0               Detector integration time
                                        (seconds)
DET1.NCORRS.NAME        Uncorr          Detector readout mode (FIXED)
DET1.NDIT               1               Number of detector integrations
                                        per exposure (FIXED).

Compulsory:
INS.FILTER.NAME         FILTERS_HOWFS   List of HOWFS filters (each must be
                                        in FILTERS_HOWFS)
SEQ.NEXPO               1               Number of reset exposures for
                                        each HOWFS filter.

Sequence
--------
Select DARK filter.
For each HOWFS filter
   Select detector window corresponding to HOWFS filter
      (without actually selecting the filter).
   Select detector chip corresponding to HOWFS filter.
   Set zero detector exposure time and select Uncorr readout mode.
   {
      For exposure = 1 to SEQ.NEXPO
         Set WCS parameters to "pixel coordinates".
         Make exposure
      Next exposure
   }
   Store BIAS calibration as HOWFSDATA/VIRCAM_HOWFS_BIAS_<filter>.fits
Next HOWFS filter
```

```
__oOo__
```

11.7.3 HOWFS observation templates

11.7.3.1 VIRCAM_howfs_obs_exp

```
Description
-----------
This is a multi-purposes template capable of making a HOWFS
observation and analysing it, or analysing an existing file of HOWFS
data. If the template is requested to make a new exposure, the
detector controller is configured with a suitable readout window and
the HOWFS beam-splitter filter is selected and positioned over the
required detector. A HOWFS observation is then made. If the temmplate
is not requested to make a new exposure, it uses a file name provided
by the user. The HOWFS image analysis process is then started. The
derived coefficients are stored in a file, which may be used to
generate the active optics lookup tables for the TCS. The template can
wait for the analysis to complete or can be requested to finish
straight away and leave the analysis running in the background,
leaving VISTA to make other observations.

Prerequirements
---------------
Telescope already pointing at reference target (VIRCAM_howfs_acq).
Suitable HOWFS calibration observations must be available.

Data Products
-------------
DPR.CATG                ACQUISITION
DPR.TECH                IMAGE
DPR.TYPE                OBJECT,PSF-CALIBRATOR

Parameters
----------
Name                    Default         Description
----                    -------         -----------
Compulsory:
DET1.DIT                5.0             Detector integration time
                                        (seconds)
INS.FILTER.NAME         (none)          Name of HOWFS filter (must be
                                        in FILTERS_HOWFS)
INS.HOWFS.EXPOSE        T               Make new exposure?
INS.HOWFS.PROCESS       T               Process the data?
INS.HOWFS.IAWAIT        T               Wait for image image analysis
                                        to finish?
INS.HOWFS.IMGFILE       (none)          If INS.HOWFS.EXPOSE=F, name of
                                        file to be analysed.
INS.HOWFS.COFILE        ""              File in which to store coefficients
                                        (if blank use exposure file
                                        name).
```

```
Optional:
DET1.NDIT               6               Number of detector integrations
                                        per exposure.
DET1.NCORRS.NAME        Double          Detector readout mode (must be
                                        in NCORRS_RANGE)
SEQ.NEXPO               1               Number of exposures (only the
                                        last exposure is analyzed).
INS.HOWFS.MASKFILE      VIRCAM_BadPixelMask.fits
                                        Name of bad pixel mask file
                                        (none if empty string).
INS.HOWFS.DARKFILE      (none)          Name of DARK calibration file
                                        (derived from filter and exp
                              time if not specified)
INS.HOWFS.FLATFILE      (none)          Name of flat-field calibration
                                        file (derived from filter if
                                        not specified).
INS.HOWFS.NULLSUB       T               Subtract null abberations?
INS.HOWFS.PREBLUR       T               Blur images before analysis?
INS.HOWFS.DESLOPE       F               Remove slope from image background?
INS.HOWFS.RECENTRE      T               Recentre images?
INS.HOWFS.MAXRTOL       1.0e-6          Maximum relative tolerance
INS.HOWFS.MAXITR        20000           Maximum iterations
INS.HOWFS.MAXFUN        25000           Maximum function calls
INS.HOWFS.MAXREP        10              Maximum repeats
INS.HOWFS.MAXCDIF       5.0e-9          Maximum coefficient difference (nm)


Sequence
--------
Check whether HOWFS server is busy and ensure it is ONLINE.
Get parameters for HOWFS filter.
If INS.HOWFS.EXPOSE=T then
   Select HOWFS filter.
   Adjust telescope focus for HOWFS filter.
   Select detector window corresponding to HOWFS filter.
   Select detector chip corresponding to HOWFS filter.
   Calculate X,Y of pointing origin corresponding to HOWFS filter.
   Obtain NULL coefficients and rotate into camera detector frame.
   Setup HOWFS image analysis with
X,Y,OBSOFF,NULLSUB,PREBLUR,MAXITR,MAXRTOL and
      HOWFS.DARKFILE = HOWFSDATA/VIRCAM_DARK_<filter>_<DIT>.fits
      HOWFS.FLATFILE = HOWFSDATA/VIRCAM_HOWFS_<filter>.fits
   Set detector exposure time and readout mode.
   Calculate dwell time (NEXPO * (DIT * NDIT + calctime)) and inform TCS.
   For exposure = 1 to SEQ.NEXPO
      Get WCS information from TCS.
      Set WCS parameters.
      Make exposure
   Next exposure
Else
   Obtain NULL coefficients and rotate into camera detector frame.
   Setup HOWFS image analysis with
X,Y,OBSOFF,NULLSUB,PREBLUR,MAXITR,MAXRTOL
      HOWFS.DARKFILE and HOWFS.FLATFILE
End if
```

```
If INS.HOWFS.PROCESS=T then
   Setup HOWFS image analysis with IMGFILE and COFILE keywords
   Calculate wavefront coefficients and store in INS.HOWFS.COFILE
   If HOWFS.IAWAIT flag is T then
      Wait for image analysis to finish
      Rotate coefficients into TCS M1 coordinate frame.
      Log wavefront coefficients
      Display statistics
   End if
End if
__oOo__
```

11.7.3.2 VIRCAM_howfs_obs_wfront

```
Description
-----------
This template makes a HOWFS wavefront measurement suitable for
determining the current residual from the active optics lookup
tables. It uses the same procedure as VIRCAM_howfs_obs_exp, with the
addition that the derived coefficients are forwarded to the TCS when
the analysis is finished. The procedure can be repeated more than once
to check that the wavefront residuals get smaller each time.

Prerequirements
---------------
Telescope already pointing at reference target (VIRCAM_howfs_acq)
Suitable HOWFS calibration observations must be available.

Data Products
-------------
DPR.CATG                ACQUISITION
DPR.TECH                IMAGE
DPR.TYPE                OBJECT,PSF-CALIBRATOR

Parameters
----------
Name                    Default         Description
----                    -------         -----------
Fixed:
SEQ.NEXPO               1               Number of exposures (fixed at
                                        1 because only the last exposure
                                        is analyzed).

Compulsory:
DET1.DIT                5.0             Detector integration time
                                        (seconds)
INS.FILTER.NAME         (none)          Name of HOWFS filter (must be
                                        in FILTERS_HOWFS)
```

```
Optional:
DET1.NDIT              6                Number of detector integrations
                                        per exposure.
DET1.NCORRS.NAME       Double           Detector readout mode (must be
                                        in NCORRS_RANGE)
INS.HOWFS.MASKFILE     VIRCAM_BadPixelMask.fits
                                        Name of bad pixel mask file
                                        (none if empty string).
INS.HOWFS.NULLSUB      T                Subtract null abberations?
INS.HOWFS.PREBLUR      T                Blur images before analysis?
INS.HOWFS.DESLOPE      F                Remove slope from image background?
INS.HOWFS.RECENTRE     T                Recentre images?
INS.HOWFS.MAXRTOL      1.0e-6           Maximum relative tolerance
INS.HOWFS.MAXITR       20000            Maximum iterations
INS.HOWFS.MAXFUN       25000            Maximum function calls
INS.HOWFS.MAXREP       10               Maximum repeats
INS.HOWFS.MAXCDIF      5.0e-9           Maximum coefficient difference (nm)
INS.HOWFS.COFILE       ""               File in which to store coefficients
                                        (if blank use exposure file
                                        name).
INS.HOWFS.INCM1ZP      T                Adjust TCS M1 zero points?
INS.HOWFS.INCM2ZP      T                Adjust TCS M2 zero points?

Sequence
--------
Check whether HOWFS server is busy and ensure it is ONLINE.
Get parameters for HOWFS filter.
Select HOWFS filter.
Adjust telescope focus for HOWFS filter.
Select detector window corresponding to HOWFS filter.
Select detector chip corresponding to HOWFS filter.
Calculate X,Y of pointing origin corresponding to HOWFS filter.
Obtain NULL coefficients and rotate into camera detector frame.
Setup HOWFS image analysis with X,Y,OBSOFF,NULLSUB,PREBLUR,MAXITR,MAXRTOL
and
   HOWFS.DARKFILE = HOWFSDATA/VIRCAM_DARK_<filter>_<DIT>.fits
   HOWFS.FLATFILE = HOWFSDATA/VIRCAM_HOWFS_<filter>.fits
Set detector exposure time and readout mode.
Calculate dwell time (DIT * NDIT + calctime) and inform TCS.
Repeat
   Get WCS information from TCS.
   Set WCS parameters.
   Make exposure
   Setup HOWFS image analysis with IMGFILE and COFILE keywords
   Calculate wavefront coefficients and store in INS.HOWFS.COFILE
   Wait for image analysis to finish
   Rotate coefficients into TCS M1 coordinate frame.
   Log wavefront coefficients
   Display statistics
   Ask operator to verify wavefront coefficients
   If operator says "OK" then
      If HOWFS.INCM2ZP is T then update TCS M2 lookup table.
      If HOWFS.INCM1ZP is T then update TCS M1 lookup table.
      If the TCS image quality has not improved then
         Undo the lookup table changes.
```

```
      End if
   End if
Until aborted or operator is satisfied with image quality
__oOo__
```

11.7.4 Imaging Acquisition Templates

11.7.4.1 VIRCAM_img_acq

```
Description
-----------
This template acquires a science target. It sets the instrument into
IMAGING mode and (if one has been specified) selects a science
filter. It also points the telescope to a new target (using a
"preset"). If not specified, the position angle defaults to orient the
instrument Y axis to the north and X axis to the west. The pointing
centre is the rotator centre unless specified otherwise in the
optional (X,Y) parameters. Any position angle specified refers to the
position angle at the pointing centre (i.e. the meridian line of the
TEL.TARG.ALPHA should intersect the column of pixels at TEL.TARG.X at
angle TEL.ROT.OFFANGLE). If autoguiding and active optics correction
are required one guide star and two aO stars are specified.

Parameters
----------
Name                    Default         Description
----                    -------         -----------
Fixed:
INS.MODE                IMAGING         Instrument mode (FIXED).

Compulsory:
INS.FILTER.NAME         (none)          Name of science filter (must be
                                        contained in FILTERS_SCI, or
                                        can be an empty string).
TEL.TARG.ALPHA          (none)          Target RA
TEL.TARG.DELTA          (none)          Target DEC
TEL.AG.START            T               Observe with autoguiding (T or F)
TEL.AO.START            T               Observe with aO (T or F)

Optional:
TEL.TWEAK               F               Allow manual position adjustment
TEL.ROT.ENABLED         T               Rotator preset enabled
TEL.ROT.OFFANGLE        0.0             Camera sky position angle
                                        (-180 to +180 degrees).
TEL.TARG.EQUINOX        2000.0          Target equinox
TEL.TARG.ADDVELALPHA    0.0             Target drift in RA (optional)
TEL.TARG.ADDVELDELTA    0.0             Target drift in Dec (optional)
TEL.TARG.PMA            0.0             Target proper motion in RA
                                        (optional)
TEL.TARG.PMD            0.0             Target proper motion in Dec
                                        (optional)
TEL.TARG.EPOCH          2000            Target epoch
TEL.TARG.EPOCHSYSTEM    J               Target epoch system
```

```
TEL.TARG.X              0.0             Pointing origin X in mm (optional)
TEL.TARG.Y              0.0             Pointing origin Y in mm (optional)

TEL.AG.GUIDESTAR        SETUPFILE       Where to find guide stars
                                        (NONE, SETUPFILE, CATALOGUE)
TEL.AG.CONFIRM          F               Confirm each new guide star?
TEL.GS1.ALPHA           (none)          RA of guide star candidate 1
TEL.GS1.DELTA           (none)          Dec of guide star candidate 1
TEL.GS1.MAG             (none)          Magnitude of guide star candidate 1
 "    "    "              "                 "       "       "
TEL.GS5.ALPHA           0.0             RA of guide star candidate 5
TEL.GS5.DELTA           0.0             Dec of guide star candidate 5
TEL.GS5.MAG             25.0            Magnitude of guide star candidate 5

TEL.AO.AOSTARA          SETUPFILE       Where to find active optics star A
                                        (NONE, SETUPFILE, CATALOGUE)
TEL.AO.AOSTARB          SETUPFILE       Where to find active optics star B
                                        (NONE, SETUPFILE, CATALOGUE)
TEL.AO.CONFIRM          F               Confirm active optics before
starting?
TEL.AO.PRIORITY         NORMAL          Active optics priority (LOW, NORMAL
                                        or HIGH)
TEL.AOSA1.ALPHA         (none)          RA of aO star A candidate 1
TEL.AOSA1.DELTA         (none)          Dec of aO star A candidate 1
TEL.AOSA1.MAG           (none)          Magnitude of aO star A candidate 1
 "    "    "              "                 "       "       "
TEL.AOSA5.ALPHA         0.0             RA of aO star A candidate 5
TEL.AOSA5.DELTA         0.0             Dec of aO star A candidate 5
TEL.AOSA5.MAG           25.0            Magnitude of aO star A candidate 5
TEL.AOSB1.ALPHA         (none)          RA of aO star B candidate 1
TEL.AOSB1.DELTA         (none)          Dec of aO star B candidate 1
TEL.AOSB1.MAG           (none)          Magnitude of aO star B candidate 1
 "    "    "              "                 "       "       "
TEL.AOSB5.ALPHA         0.0             RA of aO star B candidate 5
TEL.AOSB5.DELTA         0.0             Dec of aO star B candidate 5
TEL.AOSB5.MAG           25.0            Magnitude of aO star B candidate 5
```

```
Sequence
---------
If telescope interface is disabled then
   Pop up a warning box.
End if
If pointing origin is not (0,0) then
   Adjust telescope coordinates to bring target to pointing origin.
End if
In parallel:
1) Set instrument mode to IMAGING.
   If science filter has been specified
      Select science filter.
   End if
2) Preset telescope to target.
   If science filter has been specified
      Adjust telescope focus for science filter.
   End if
   {
      If autoguiding is enabled then
         If AG.CONFIRM is TRUE then
            Prompt operator to confirm autoguiding.
         End if
         Wait for autoguiding to start
      End if
      If active optics are enabled and AO.PRIORITY is > 0 then
         Wait for active optics to start
      End if
   }
Save the current target coordinates for future observation templates.
If TEL.TWEAK is T then
   Prompt for the observer to tweak the telescope position manually.
   If not skipped then
      Update the saved target coordinates.
   End if
End if
__oOo__
```

11.7.4.2 VIRCAM_img_acq_altaz

```
Description
-----------
This template acquires a target in alt-azimuth coordinates. It sets
the instrument into IMAGING mode and (if one has been specified)
selects a science filter. It also points the telescope to a new
alt-azimuth target (using a "preset").  If not specified, the position
angle defaults to orient the instrument Y axis to the north and X axis
to the west. The pointing centre is the rotator centre unless
specified otherwise in the optional (X,Y) parameters. Once the target
is acquired, the telescope is tracked but there is no autoguiding or
closed loop active optics. The rotator is tracked in Alt-Az.

Parameters
----------
Name                    Default         Description
----                    -------         -----------
Fixed:
INS.MODE                IMAGING         Instrument mode (FIXED).
TEL.AG.START            F               No autoguiding (FIXED).
TEL.AO.START            F               No closed loop aO (FIXED).
TEL.ROT.ALTAZTRACK      T               Track rotator on alt-az axis
                                        instead of RA,Dec
Compulsory:
INS.FILTER.NAME         (none)          Name of science filter (must be
                                        contained in FILTERS_SCI, or
                                        can be blank).
TEL.TARG.AZ             (none)          Target azimuth
TEL.TARG.ALT            (none)          Target altitude

Optional:
TEL.ROT.ENABLED         T               Rotator preset enabled
TEL.ROT.OFFANGLE        0.0             Camera sky position angle
                                        (-180 to +180 degrees).
TEL.TARG.X              0.0             Pointing origin X in mm (optional)
TEL.TARG.Y              0.0             Pointing origin Y in mm (optional)

Sequence
--------
If telescope interface is disabled then
   Pop up a warning box.
End if
If pointing origin is not (0,0) then
   Adjust telescope coordinates to bring target to pointing origin.
End if
In parallel:
1) Set instrument mode to IMAGING.
   If science filter has been specified
      Select science filter.
   End if
2) Preset telescope to target.
   If science filter has been specified
      Adjust telescope focus for science filter.
```

```
   End if
__oOo__
```

11.7.4.3 VIRCAM_img_acq_domescreen

```
Description
-----------
This template sets the instrument into IMAGING mode and (if one has
been specified) selects a science filter. It also moves the telescope
to point at the flat-field screen in the dome (using a "preset").
Telescope tracking and guiding are switched off and the required
flat-field illumination level defined. The flat-field illumination
source is switched on and allowed to stabilise.

Parameters
----------
Name                    Default         Description
----                    -------         -----------
Fixed:
INS.MODE                IMAGING         Instrument mode (FIXED).
TEL.TARG.NAME           DOMEFLAT        Telescope target name (FIXED).
TEL.AG.START            F               No autoguiding (FIXED).
TEL.AO.START            F               No closed loop aO (FIXED).
TEL.ECS.FLATOFF         F               Do not switch flat-field off
                                        when finished (FIXED).

Compulsory:
INS.FILTER.NAME         (none)          Name of science filter (must be
                                        contained in FILTERS_SCI, or
                                        can be blank).
TEL.ECS.FFREQ           1               Telescope flat-field level (1-7).

Sequence
--------
If telescope interface is disabled then
   Pop up a warning box.
End if
In parallel:
1) Set instrument mode to IMAGING.
   If science filter has been specified
      Select science filter.
   End if
2) Preset telescope to DOMEFLAT (no tracking or autoguiding)
   Set flat-field illumination to TEL.ECS.FFREQ
Ask operator to confirm that ambient lights are turned off.
__oOo__
```

11.7.4.4 VIRCAM_img_twilight

```
Description
-----------
This template is used to select a twilight sky field. It sets the
instrument into IMAGING mode and (if one has been specified) selects a
science filter and points the telescope to the twilight sky. The
twilight sky coordinates are obtained by searching a database of
twilight sky coordinates and choosing one closest to the desired
Altitude and Azimuth (while avoiding the Moon and keeping within the
Altitude and Azimuth limits, which are different for morning and
evening twilight). The telescope is tracked but guiding is switched
off. If not specified, the position angle defaults to orient the
instrument Y axis to the north and X axis to the west.

Parameters
----------
Name                    Default         Description
----                    -------         -----------
Fixed:
INS.MODE                IMAGING         Instrument mode (FIXED).
TEL.AG.START            F               No autoguiding (FIXED)
TEL.AO.START            F               No closed loop aO (FIXED)

Compulsory:
INS.FILTER.NAME         (none)          Name of science filter (must be
                                        contained in FILTERS_SCI, or
                                        can be blank).

Optional:
TEL.TWILIGHT.ALT        60.0            Desired target altitude for
                                        twilight sky observation.
TEL.TWILIGHT.AZ          0.0            Desired target azimuth for
                                        twilight sky observation.
TEL.ROT.OFFANGLE         0.0            Camera sky position angle
                                        (-180 to +180 degrees).
TEL.TWILIGHT.ALTMIN     25.0            Minimum target altitude for
                                        twilight sky observation.
TEL.TWILIGHT.ALTMAX     85.0            Maximum target altitude for
                                        twilight sky observation.
TEL.TWILIGHT.AM.AZMIN  260.0            Minimum target azimuth for
                                        morning twilight sky observation
                                        (for sun avoidance).
TEL.TWILIGHT.AM.AZMAX  360.0            Maximum target azimuth for
                                        morning twilight sky observation
                                        (for sun avoidance).
TEL.TWILIGHT.PM.AZMIN    0.0            Minimum target azimuth for
                                        evening twilight sky observation
                                        (for sun avoidance).
TEL.TWILIGHT.PM.AZMAX  100.0            Maximum target azimuth for
                                        evening twilight sky observation
                                        (for sun avoidance).
TEL.MOON.AVOID          20.0            Moon avoidance in degrees.
```

```
Sequence
--------
If telescope interface is disabled then
   Pop up a warning box.
End if
Set instrument mode to IMAGING.
If science filter has been specified
   Select science filter.
End if
Repeat
   Query local time, local sidereal time, observatory latitude,longitude
   and moon RA,Dec from TCS
   Search database to find coordinates of a twilight sky closest to
   TEL.TWILIGHT.ALT,TEL.TWILIGHT.AZ, within the given range of ALT and AZ,
   but avoiding the Moon (Ra,Dec) by TEL.MOON.AVOID degrees.
   Preset telescope to (Ra,Dec) of selected twilight sky coordinates,
   disabling autoguiding and active optics.
   If science filter has been specified
      Adjust telescope focus for science filter.
   End if
   Wait for operator to adjust and confirm twilight sky position.
Until operator confirms twilight sky successfully acquired.
__oOo__
```

11.7.5 Imaging calibration templates

11.7.5.1 VIRCAM_img_cal_crosstalk

```
Description
-----------
This template makes a series of exposures, with each exposure offset
from the previous one by a sequence of mesosteps (i.e. offsets of
intermediate size between a jitter and a tile) designed to place the
image of a bright star on each of the 16 readout sectors on each
detector. The resultant series of exposures can be used to detect any
cross-talk between detector readout sectors. The template assumes that
a bright, nearly saturated  star has already been acquired on the
first sector of the first detector. Autoguiding is not needed.

Prerequirements
---------------
Telescope already pointing at reference target, with the image falling
on the first readout sector of the first detector
(VIRCAM_img_acq_crosstalk).

Data Products
-------------
DPR.CATG                CALIB
DPR.TECH                IMAGE,JITTER
DPR.TYPE                OBJECT,CROSSTALK

Parameters
----------
Name                    Default         Description
----                    -------         -----------
Fixed:
SEQ.NESTING             FPJME           Nesting pattern
SEQ.USTEP.ID            Single          Microstep ID (no microstepping)
SEQ.JITTER.SCALE        1.0             Jitter scale multiplier

Compulsory:
INS.FILTER.NAME          (none)         Science filter (must be in
                                        FILTERS_SCI)

SEQ.TILE.ID             TileXTalk       Tile pattern ID (default is a
                                        sequence of 16 steps to offset
                                        the target to the same place
                                        on each detector).
SEQ.REF.FILES            (none)         List of reference setup files
                                        to define AG and aO stars (not
                                        important).
SEQ.JITTER.ID           JitterXTalk     Jitter pattern ID (default is
                                        a sequence of 16 mesosteps to
                                        offset the target to each
                                        readout sector on a single
                                        detector.
DET1.DIT                10.0            Detector integration time
```

```
                                                (seconds)

Optional:
OCS.EXTENDED              F                 T if object is extended
DET1.NDIT                 1                 Number of detector integrations
                                            per exposure.
DET1.NCORRS.NAME          Double            Detector readout mode (must be
                                            in NCORRS_RANGE)
SEQ.NEXPO                 1                 Number of exposures.

Sequence
--------
Same sequence as VIRCAM_img_obs_tile, with different data product
keywords (and different tile and jitter patterns, as above).
__oOo__
```

11.7.5.2 VIRCAM_img_cal_dark

```
Description
-----------
This template makes one or more DARK exposures. The dark filter is
selected and exposures are made at the same exposure time and
integration time as the science observation they are intended to
calibrate.

Prerequirements
---------------
None.

Data Products
-------------
DPR.CATG                  CALIB
DPR.TECH                  IMAGE
DPR.TYPE                  DARK

Parameters
----------
Name                      Default           Description
----                      -------           -----------
Fixed:
INS.FILTER.DARK           SUNBLIND          Name of DARK filter (FIXED)

Compulsory:
DET1.DIT                  10.0              Detector integration time
                                            (seconds)

Optional:
DET1.NDIT                 1                 Number of detector integrations
                                            per exposure.
DET1.NCORRS.NAME          Double            Detector readout mode (must be
                                            in NCORRS_RANGE)
SEQ.NEXPO                 1                 Number of DARK exposures.

Sequence
```

```
--------
Select DARK filter.
Set detector exposure time and readout mode.
{
   For exposure = 1 to SEQ.NEXPO
      Set WCS parameters to "pixel coordinates".
      Make exposure
   Next exposure
}
__oOo__
```

11.7.5.3 VIRCAM_img_cal_darkcurrent

```
Description
-----------
This template makes a series of DARK exposures (as in
VIRCAM_img_cal_dark) but at a variety of different exposure times. The
resulting data can be used to determine the detector dark current.

NOTE: This template is the same as VIRCAM_img_cal_dark with a list of
exposure times.

Prerequirements
---------------
None.

Data Products
-------------
DPR.CATG                CALIB
DPR.TECH                IMAGE
DPR.TYPE                DARK,DARKCURRENT

Parameters
----------
Name                    Default         Description
----                    -------         -----------
Fixed:
INS.FILTER.DARK         SUNBLIND        Name of DARK filter (FIXED)

Compulsory:
DET1.DIT                10,20,30,40,50  List of detector integration
                                        times (seconds)

Optional:
DET1.NDIT               1               Number of detector integrations
                                        per exposure.
DET1.NCORRS.NAME        Double          Detector readout mode (must be
                                        in NCORRS_RANGE)
SEQ.NEXPO               1               Number of DARK exposures at
                                        each exposure time.
```

```
Sequence
--------
Select DARK filter.
For each exposure time
   Set detector exposure time and readout mode.
   {
      For exposure = 1 to SEQ.NEXPO
         Set WCS parameters to "pixel coordinates".
         Make exposure
      Next exposure
   }
Next exposure time
__oOo__
```

11.7.5.4 VIRCAM_img_cal_domeflat

```
Description
-----------
This template makes a flat-field exposure (or series of exposures)
suitable for calibrating an IMAGING observation. It assumes the
VIRCAM_img_acq_domescreen template has been executed and the telescope
is already pointing at the dome screen with the calibration source
turned on. At the end of the template the flat-field calibration
source may optionally be switched off.

When a list of science filters is specified, it may be necessary to
associate a different exposure time with each filter (because the
filters have different transmissions). It is possible to do this by
specifying the DET.DIT parameter as a list of exposure times. The list
must be exactly the same length as the list of science filters. If
DET.DIT is given a single value, that exposure time will be applied to
all filters. For example

   DET.DIT           1.0;
   INS.FILTER.NAME   "H J Ks";

will expose with H, J and Ks filters, each with an exposure time of
1.0 seconds, and

   DET.DIT           "1.0 2.0 3.0";
   INS.FILTER.NAME   "H J Ks";

will expose with H filter for 1.0 seconds, the J filter for 2.0
seconds and the Ks filter for 3.0 seconds. The combination

   DET.DIT           "1.0 2.0";
   INS.FILTER.NAME   "H J Ks";

is illegal because the lists have different lengths.
```

```
Prerequirements
---------------
Telescope already pointing at flat-field target
(VIRCAM_img_acq_domescreen) with calibration source switched on.

Data Products
-------------
DPR.CATG                CALIB
DPR.TECH                IMAGE
DPR.TYPE                FLAT,LAMP

Parameters
----------
Name                    Default          Description
----                    -------          -----------
Compulsory:
INS.FILTER.NAME         (none)           List of science filters (must
                                         be in FILTERS_SCI)
DET1.DIT                10.0             Either: Detector integration time
                                         (seconds)
                                         Or: List of detector integration
                                         times corresponding to list of
                                         science filters.

Optional:
DET1.NDIT               1                Either: Number of detector
                                         integrations per exposure.
                                         Or: List of detector integrations
                                         per exposure corresponding to list
                                         of science filters.
DET1.NCORRS.NAME        Double           Detector readout mode (must be
                                         in NCORRS_RANGE)
SEQ.NEXPO               1                Number of exposures.
TEL.ECS.FFREQ           1                Telescope flat-field level (0-7)
TEL.ECS.FLATOFF         T                Switch flat-field off when
                                         finished?
SEQ.NEXPO               1                Number of flat-field exposures.

Sequence
--------
Set detector readout mode.
For each science filter
   Select science filter
   Set detector exposure time for science filter.
   Adjust telescope focus for science filter.
   {
      For exposure = 1 to SEQ.NEXPO
         Set WCS parameters to "pixel coordinates".
         Make exposure
      Next exposure
   }
Next science filter
If required, switch off flat-field source.
__oOo__
```

11.7.5.5 VIRCAM_img_cal_illumination

```
Description
-----------
This template makes a series of exposures, with each exposure offset
from the previous one by a sequence of mesosteps (i.e. offsets of
intermediate size between a jitter and a tile) designed to place
bright star at a regular grid of offset positions across each
detector. The resultant series of exposures can be used to calibrate
the illumination correction caused by scattering within the
camera. The template assumes that a sparse field of bright stars has
already been acquired at the first mesostep position.

Prerequirements
---------------
Telescope already pointing at reference target (VIRCAM_img_acq)

Data Products
-------------
DPR.CATG                 CALIB
DPR.TECH                 IMAGE,JITTER
DPR.TYPE                 STD,ILLUMINATION

Parameters
----------
Name                     Default          Description
----                     -------          -----------
Compulsory:
DET1.DIT                 10.0             Detector integration time
                                          (seconds)
INS.FILTER.NAME          (none)           Science filter (must be in
                                          FILTERS_SCI)
SEQ.OFFSETALPHA          ( 30 arcsec) List of mesostep RA offsets
SEQ.OFFSETDELTA          ( 30 arcsec) List of mesostep Dec offsets
SEQ.REF.FILES            (none)           List of reference setup files
                                          containing telescope SETUP
                                          keywords defining AG and AO stars
                                          for each mesostep.

Optional:
DET1.NDIT                1                Number of detector integrations
                                          per exposure.
DET1.NCORRS.NAME         Double           Detector readout mode (must be
                                          in NCORRS_RANGE)
SEQ.NEXPO                1                Number of exposures.
SEQ.OFFSETROT            0.0              List of rotator offsets

Each SEQ.REF.FILES file contains
--------------------------------
Compulsory keywords:
TEL.AG.START             T                Observe with autoguiding
TEL.AO.START             T                Observe with aO (T or F)
```

```
Optional keywords (only if TEL.AG.START=T or TEL.AO.START=T):
TEL.AG.CONFIRM            F                Confirm each new guide star?
TEL.GS1.ALPHA             (none)           RA of guide star candidate 1
TEL.GS1.DELTA             (none)           Dec of guide star candidate 1
TEL.GS1.MAG               (none)           Magnitude of guide star
                                           candidate 1
TEL.AO.PRIORITY           1                Active optics priority (0, 1 or 2)
TEL.AOSA1.ALPHA           (none)           RA of aO star A candidate 1
TEL.AOSA1.DELTA           (none)           Dec of aO star A candidate 1
TEL.AOSA1.MAG             (none)           Magnitude of aO star A
                                           candidate 1
TEL.AOSB1.ALPHA           (none)           RA of aO star B candidate 1
TEL.AOSB1.DELTA           (none)           Dec of aO star B candidate 1
TEL.AOSB1.MAG             (none)           Magnitude of aO star B
                                           candidate 1

Sequence
--------
Same as VIRCAM_img_obs_offsets, with different data product keywords
(and a different default list of offsets).
__oOo__
```

11.7.5.6 VIRCAM_img_cal_linearity

```
Description
-----------
This template makes a series of DARK exposures (as
VIRCAM_img_cal_darkcurrent) followed by the same number of flat-field
exposures (as VIRCAM_img_cal_domeflat) but at a variety of different
exposure times. The Nth dark exposure should have exactly the same
integration time and number of integrations as the Nth flat-field
exposure. The resulting data can be used to determine the linearity of
the detector response. At the end of the template the flat-field
calibration source may optionally be switched off.

This template shares the same VIRCAM_img_cal_darkflat sequence script
as the VIRCAM_img_cal_noisegain template. The main difference is that
VIRCAM_img_cal_linearity uses a list of different integration times,
whereas VIRCAM_img_cal_noisegain uses the same integration time.

Prerequirements
---------------
Telescope already pointing at flat-field target
(VIRCAM_img_acq_domescreen) with calibration source switched on.

Data Products
-------------
DPR.CATG                  CALIB
DPR.TECH                  IMAGE
DPR.TYPE                  DARK,LINEARITY (or DARK,CHECK)
                          FLAT,LAMP,LINEARITY (or FLAT,LAMP,CHECK)
```

```
Parameters
----------
Name                    Default         Description
----                    -------         -----------
Compulsory:
INS.FILTER.NAME         (none)          Science filter (must be in
                                        FILTERS)
DET1.DIT                5,10,15,20      List of detector integration
                                        times (seconds)

Optional:
DET1.NDIT               1               Number of detector integrations
                                        per exposure.
DET1.NCORRS.NAME        Double          Detector readout mode (must be
                                        in NCORRS_RANGE)
SEQ.ADDFIXED            F               Add fixed exposures to calibrate
                                        light source?
SEQ.DET.DITFIXED        7.5             Exposure time for fixed exposures.
SEQ.NEXPO               1               Number of repeats per exposure
                                        time.
TEL.ECS.FLATOFF         T               Switch flat-field off when
                                        finished?

Sequence (VIRCAM_img_cal_darkflat)
----------------------------------
Same as VIRCAM_img_cal_noisegain.
__oOo__
```

11.7.5.7 VIRCAM_img_cal_noisegain

```
Description
-----------
This template makes a series of (typically two) dark exposures (as
VIRCAM_img_cal_darkcurrent) followed by the same number of flat-field
exposures (as VIRCAM_img_cal_domeflat). All the observations have the
same exposure time. The resulting data can be used to make a
measurement of the detector readout noise and gain. At the end of the
template the flat-field calibration source may optionally be switched
off.

This template shares the same VIRCAM_img_cal_darkflat sequence script
as the VIRCAM_img_cal_linearity template. The main difference is that
VIRCAM_img_cal_linearity uses a list of different integration times,
whereas VIRCAM_img_cal_noisegain uses the same integration time.

Prerequirements
---------------
Telescope already pointing at flat-field target
(VIRCAM_img_acq_domescreen) with calibration source switched on.
```

```
Data Products
-------------
DPR.CATG                CALIB
DPR.TECH                IMAGE
DPR.TYPE                DARK,GAIN (or DARK,CHECK)
                        FLAT,LAMP,GAIN (or FLAT,LAMP,CHECK)

Parameters
----------
Name                    Default         Description
----                    -------         -----------
Compulsory:
INS.FILTER.NAME         (none)          Science filter (must be in
                                        FILTERS_SCI)
DET1.DIT                10              Detector integration time
                                        (seconds)
SEQ.ADDFIXED            F               Add fixed exposures to calibrate
                                        light source?
SEQ.DET.DITFIXED        7.5             Exposure time for fixed exposures.
SEQ.NEXPO               2               Number of exposures each for DARK
                                        and FLAT.

Optional:
DET1.NDIT               1               Number of detector integrations
                                        per exposure.
DET1.NCORRS.NAME        Double          Detector readout mode (must be
                                        in NCORRS_RANGE)
TEL.ECS.FLATOFF         T               Switch flat-field off when
                                        finished?

Sequence (VIRCAM_img_cal_darkflat)
----------------------------------
Select DARK filter.
For each exposure time
   Set detector exposure time and readout mode.
   For exposure = 1 to SEQ.NEXPO
      Set WCS parameters to "pixel coordinates".
      Make DARK exposure
   Next exposure
Next exposure time
If fixed exposure required
   Make fixed DARK exposure (DARK,CHECK)
Endif

Select science filter
Adjust telescope focus for science filter.
For each exposure time
   Set detector exposure time and readout mode.
   {
      For exposure = 1 to SEQ.NEXPO
         Set WCS parameters to "pixel coordinates".
         Make FLAT exposure
      Next exposure
   }
```

```
   If fixed exposure required
      Make fixed FLAT exposure (DARK,CHECK)
   Endif
Next exposure time
If required, switch off flat-field source.
__oOo__
```

11.7.5.8 VIRCAM_img_persistence

```
Description
-----------
This template makes one exposure with a selected science filter,
followed by a series of dark exposures. All exposures have the same
integration time and number of integrations. The resulting sequence of
exposures can be used to measure the image persistence. The template
assumes that a field has been acquired containing a bright, nearly
saturated star.

Prerequirements
---------------
Telescope already pointing at reference target (VIRCAM_img_acq)

Data Products
-------------
DPR.CATG                CALIB
DPR.TECH                IMAGE
DPR.TYPE                OBJECT,PERSISTENCE      for first exposure
                        DARK,PERSISTENCE        for subsequent exposures


Parameters
----------
Name                    Default         Description
----                    -------         -----------
Fixed:
INS.FILTER.DARK         SUNBLIND        Name of DARK filter (FIXED)

Compulsory:
DET1.DIT                10.0            Detector integration time (seconds)
INS.FILTER.NAME         (none)          Science filter (must be in FILTERS)

Optional:
DET1.NDIT               1               Number of detector integrations
                                        per exposure.
DET1.NCORRS.NAME        Double          Detector readout mode (must be
                                        in NCORRS_RANGE)
SEQ.NEXPO               10              Number of DARK exposures.
```

```
Sequence
--------
Select science filter
Adjust telescope focus for science filter.
Set detector exposure time and readout mode.
Calculate dwell time ((NEXPO+1) * DIT * NDIT) and inform TCS.
Get WCS information from TCS.
{
   Set WCS parameters.
   Make exposure
}
Select DARK filter.
{
   For exposure = 1 to SEQ.NEXPO
      Set WCS parameters to "pixel coordinates".
      Make exposure
   Next exposure
}
__oOo__
```

11.7.5.9 VIRCAM_img_cal_reset

```
Description
-----------
This template makes a reset (aka BIAS) exposure with the shortest
possible exposure time. The dark filter is selected and a reset/read
(uncorrelated) sequence executed by the detector controller.

Prerequirements
---------------
None.

Data Products
-------------
DPR.CATG                CALIB
DPR.TECH                IMAGE
DPR.TYPE                BIAS

Parameters
----------
Name                    Default         Description
----                    -------         -----------
Fixed:
INS.FILTER.DARK         SUNBLIND        Name of DARK filter (FIXED)
DET1.DIT                0               Detector integration time
                                        (seconds) (FIXED)
DET1.NDIT               1               Number of detector integrations
                                        per exposure (FIXED).
DET1.NCORRS.NAME        Uncorr          Detector readout mode (FIXED)
```

```
Optional:
SEQ.NEXPO                1               Number of BIAS exposures.

Sequence
--------
Select DARK filter.
Set zero detector exposure time and select Uncorr readout mode.
{
   For exposure = 1 to SEQ.NEXPO
      Set WCS parameters to "pixel coordinates".
      Make exposure
   Next exposure
}
__oOo__
```

11.7.5.10 VIRCAM_img_cal_std

```
Description
-----------
This template makes one "pawprint" observation of a field of
photometric standards. Its implementation is identical to
VIRCAM_img_obs_paw, described below, except for the template name and
"DPR TYPE" header keyword that end up in the resulting data.

Prerequirements
---------------
Telescope already pointing at reference target (VIRCAM_img_acq)

Data Products
-------------
DPR.CATG                 CALIB
DPR.TECH                 IMAGE,JITTER
DPR.TYPE                 STD,FLUX

Parameters
----------
Same parameters as VIRCAM_img_obs_paw, without OCS.EXTENDED.

Sequence
--------
Same sequence as VIRCAM_img_obs_paw, with different data product
keywords.
__oOo__
```

11.7.5.11 VIRCAM_img_cal_twiflat

```
Description
-----------
This template makes a series of exposures sufficient to make a
twilight sky flat-field suitable for calibrating an IMAGING
observation. It assumes the VIRCAM_img_acq_twilight template has been
executed and the telescope is already pointing at the twilight
sky. The template makes a series of test exposures, first in
uncorrelated mode to check for saturation and then in correlated mode,
and waits until the sky background level reaches a desired
target. Then a twilight sky observation is made. Several exposures can
be made at different telescope offsets, so that star images can be
averaged out when the data are processed.

Prerequirements
---------------
Telescope already pointing at twilight sky (VIRCAM_img_twilight)

Data Products
-------------
DPR.CATG                TECHNICAL - for test exposures
                        CALIB     - for twilight sky calibration exposures
DPR.TECH                IMAGE
DPR.TYPE                FLAT,TWILIGHT

Parameters
----------
Name                    Default         Description
----                    -------         -----------
Compulsory:
INS.FILTER.NAME         (none)          Science filter or list of
                                        science filters (must be in
                                        FILTERS_SCI).
DET1.DITMIN               1.0           Shortest allowable detector
                                        integration time (seconds)
DET1.DITMAX              60.0           Longest allowable detector
                                        integration time (seconds)
DET1.EXPLEVEL           10000           Required mean exposure level.
SEQ.OFFSETALPHA         -60.0 -60.0  60.0 60.0
                                        List of RA offsets
SEQ.OFFSETDELTA         -60.0  60.0 -30.0 90.0
                                        List of Dec offsets
SEQ.OFFSETROT             0.0   0.0   0.0  0.0
                                        List of rotator offsets

Optional:
DET1.NDIT               1               Number of detector integrations
                                        per exposure.
DET1.NCORRS.NAME        Double          Detector readout mode (must be
                                        in NCORRS_RANGE)
SEQ.NEXPO               1               Number of exposures.
```

```
Sequence
--------
Get local time (morning or evening).
For each science filter in the list
   Select science filter
   Adjust telescope focus for science filter.

   Set detector exposure time to DET1.DITMIN.
   Set detector to uncorrelated mode.
   Set WCS parameters to "pixel coordinates".
   Make exposure (DPR CATG = TECHNICAL).
   Calculate the mean exposure level.
   If uncorrelated exposure is saturated then
      If local time is before midnight then
         Repeat
            Get WCS information from TCS.
            Set WCS parameters.
            Make exposure (DPR CATG = TECHNICAL).
            Calculate the mean exposure level.
         Until exposure is no longer saturated.
      Else
         Abort - morning sky already saturated.
      Endif
   End if

   Set detector exposure time to DET1.DITMIN.
   Set detector to correlated mode.
   Set WCS parameters to "pixel coordinates".
   Make exposure (DPR CATG = TECHNICAL).
   Calculate the mean exposure level.
   If actual exposure level > desired exposure level then
      If local time is before midnight then
         Repeat
            Get WCS information from TCS.
            Set WCS parameters.
            Make exposure (DPR CATG = TECHNICAL).
            Calculate the mean exposure level.
         Until exposure level <= desired exposure level
         Set DET1.DIT = DET1.DITMIN
      Else
         Abort - morning sky already too bright.
      Endif
   Else
      Calculate exposure time required to obtain desired level.
      If calculated exposure time > DET1.DITMAX then
         If local time is after midnight then
            Repeat
               Get WCS information from TCS.
               Set WCS parameters.
               Make exposure (DPR CATG = TECHNICAL).
               Calculate the mean exposure level.
               Calculate exposure time required to obtain desired level.
            Until required exposure time <= DET1.DITMAX
            Set DET1.DIT = calculated exposure time.
         Else
```

```
            Abort - evening sky already too dark.
         Endif
      Else
         Set DET1.DIT to calculated exposure time
      Endif
   Endif

   Set detector exposure time to calculated value.
   {
      Set SEQ.NOFFSETS to minimum size of SEQ.OFFSETALPHA and
      SEQ.OFFSETDELTA lists.
      For each offset, i
         Move telescope to SEQ.OFFSETALPHA[i],SEQ.OFFSETDELTA[i]
                           SEQ.OFFSETROT[i]
         Get WCS information from TCS.
         For exposure = 1 to SEQ.NEXPO
            Make exposure (DET CATG = CALIB)
         Next exposure
      Next offset
   }
Next science filter
__oOo__
```

11.7.6 Imaging observation templates

11.7.6.1 VIRCAM_img_obs_offsets

```
Description
-----------
This template makes a series of observations using a list of
user-defined telescope offsets (suitable for making a one-off, ad hoc
observation not covered by the pre-defined tile and jitter
patterns). It is assumed the telescope has already been pointed to the
null target with the VIRCAM_img_acq template. A list of position angle
offsets may be specified, which default to zero and are assumed to be
offsets with respect to the position angle specified in the
VIRCAM_img_acq template. Unlike the tile template, guide stars cannot
be chosen in advance with the VISTA SADT. The default behaviour is to
choose new guide stars for each offset on the fly. The detector
controller is configured with the required readout and exposure time
parameters and the specified number of exposures made at each of the
specified telescope offsets.

When a list of science filters is specified, it may be necessary to
associate a different exposure time with each filter (because the
filters have different transmissions). It is possible to do this by
specifying the DET.DIT parameter as a list of exposure times. The list
must be exactly the same length as the list of science filters.  If
DET.DIT is given a single value, that exposure time will be applied to
all filters. For example

   DET.DIT            1.0;
   INS.FILTER.NAME    "H J Ks";

will expose with H, J and Ks filters, each with an exposure time of
1.0 seconds, and

   DET.DIT            "1.0 2.0 3.0";
   INS.FILTER.NAME    "H J Ks";

will expose with H filter for 1.0 seconds, the J filter for 2.0
seconds and the Ks filter for 3.0 seconds. The combination

   DET.DIT            "1.0 2.0";
   INS.FILTER.NAME    "H J Ks";

is illegal because the lists have different lengths.

Prerequirements
---------------
Telescope already pointing at reference target (VIRCAM_img_acq).
```

```
Data Products
-------------
DPR.CATG                SCIENCE
DPR.TECH                IMAGE,JITTER
DPR.TYPE                OBJECT          When OCS.EXTENDED is F
                        OBJECT,EXTENDED When OCS.EXTENDED is T

Parameters
----------
Name                    Default         Description
----                    -------         -----------
Compulsory:
DET1.DIT                10.0            Either: Detector integration time
                                        (seconds)
                                        Or: List of detector integration
                                        times corresponding to list of
                                        science filters.
DET1.NDIT               1               Either: Number of detector
                                        integrations per exposure.
                                        Or: List of detector integrations
                                        per exposure corresponding to list
                                        of science filters.
INS.FILTER.NAME         (none)          List of science filter (each
                                        must be in FILTERS_SCI)
SEQ.OFFSETALPHA         (none)          List of RA offsets
SEQ.OFFSETDELTA         (none)          List of Dec offsets
SEQ.OFFSETROT           (none)          List of rotator offsets
SEQ.REF.FILEi           "CAT"           Pointer to PAF file containing
                                        telescope SETUP keywords defining
                                        AG and AO stars for offset i.
                                        (Empty string means no file
                                        available - keep existing
                              stars. CAT means choose stars
                              on the fly).
Optional:
DET1.NCORRS.NAME        Double          Detector readout mode (must be
                                        in NCORRS_RANGE)
SEQ.NEXPO               1               Number of exposures per offset.
OCS.EXTENDED            F               T if object is extended
```

```
The SEQ.REF.FILEi keyword may point to a PAF file containing
------------------------------------------------------------
Compulsory keywords:
PAF.NAME etc..         (file name)     Header of PAF file (see bob and OHS
                                       documentation).
TPL.FILE.DIRNAME       $INS_ROOT/$INS_USER/MISC/VISTA
                                       Folder to contain PAF file.

TEL.AG.START           T               Observe with autoguiding (T or F)
TEL.AO.START           T               Observe with aO (T or F)

Minimum star candidates required when TEL.AG.START=T or TEL.AO.START=T:
TEL.GS1.ALPHA          (none)          RA of guide star candidate 1
TEL.GS1.DELTA          (none)          Dec of guide star candidate 1
TEL.GS1.MAG            (none)          Magnitude of guide star
                                       candidate 1
TEL.AOSA1.ALPHA        (none)          RA of aO star A (PY) candidate 1
TEL.AOSA1.DELTA        (none)          Dec of aO star A (PY) candidate 1
TEL.AOSA1.MAG          (none)          Magnitude of aO star A (PY)
                                       candidate 1
TEL.AOSB1.ALPHA        (none)          RA of aO star B (NY) candidate 1
TEL.AOSB1.DELTA        (none)          Dec of aO star B (NY) candidate 1
TEL.AOSB1.MAG          (none)          Magnitude of aO star B (NY)
                                       candidate 1

Optional keywords:
TEL.AG.CONFIRM         F               Confirm each new guide star?
TEL.AO.PRIORITY        NORMAL          Active optics priority (LOW, NORMAL
                                       or HIGH)

Optional additional star candidates:
TEL.GSi.ALPHA          (none)          RA of guide star candidate i
                                       (i=2-5)
TEL.GSi.DELTA          (none)          Dec of guide star candidate i
                                       (i=2-5)
TEL.GSi.MAG            (none)          Magnitude of guide star candidate
                                       i (i=2-5)
TEL.AOSAi.ALPHA        (none)          RA of aO star A (PY) candidate i
                                       (i=2-5)
TEL.AOSAi.DELTA        (none)          Dec of aO star A (PY) candidate i
                                       (i=2-5)
TEL.AOSAi.MAG          (none)          Magnitude of aO star A (PY)
                                       candidate i (i=2-5)
TEL.AOSBi.ALPHA        (none)          RA of aO star B (NY) candidate
                                       i (i=2-5)
TEL.AOSBi.DELTA        (none)          Dec of aO star B (NY) candidate
                                       i (i=2-5)
TEL.AOSBi.MAG          (none)          Magnitude of aO star B (NY)
                                       candidate i (i=2-5)
```

```
Sequence
--------
Recall telescope pointing reference from last acquisition template.
Set detector readout mode.
For each science filter
   Select science filter
   Set detector exposure time for science filter.
   Determine telescope focus for science filter.
   Set SEQ.NOFFSETS to minimum size of SEQ.OFFSETALPHA and
   SEQ.OFFSETDELTA lists.
   For each offset, i
      Move telescope to SEQ.OFFSETALPHA[i],SEQ.OFFSETDELTA[i]
                        SEQ.OFFSETROT[i]
      {
         If SEQ.REF.FILES[i] is not a blank or null string then
            If SEQ.REF.FILE[i] is set to "CAT" then
               Select stars on the fly from online catalogue
            Else If SEQ.REF.FILE[i]) file exists then
               Send SETUP -file SEQ.REF.FILES[i] to TCS
            Else
               Issue warning and define new guide star setup
               parameters to select stars on the fly from
               online catalogue.
            Endif
         Else
            (Keep previously defined stars).
         Endif
      }
      {
         If TEL.AG.START is TRUE then
            If AG.CONFIRM is TRUE then
               Prompt operator to confirm autoguiding.
            End if
            Wait for autoguiding to start
         End if
         If TEL.AO.START is TRUE and AO.PRIORITY > 0 then
            Wait for active optics to start
         End if
      }
      Get WCS information from TCS.
      Calculate dwell time (NEXPO * DIT * NDIT) and inform TCS.
      {
         For each exposure
            Define header keywords:
               NOFFSETS, OFFSET_ID, OFFSET_I, OFFSET_X, OFFSET_Y
            Set WCS parameters.
            Make exposure
         Next exposure
      }
   Next offset
Next science filter
__oOo__
```

11.7.6.2 VIRCAM_img_obs_paw

```
Description
-----------
This template makes one "pawprint" observation using a selection of
filter changes, jittering and microstepping movements. It is assumed
the telescope has already been positioned at the target, and the
camera position angle defined, using the VIRCAM_img_acq template. The
detector controller is configured with the required readout and
exposure time parameters and the following sequence executed:

FJME     Step through science filters in outer loop. At each science
         filter execute a jitter pattern (if specified), and within
         each jitter pattern execute a microstep pattern (if
         specified). The jitter and microstep patterns are made using
         the camera position angle specified during the VIRCAM_img_acq
         template, unless a new position angle is specified here.

When a list of science filters is specified, it may be necessary to
associate a different exposure time with each filter (because the
filters have different transmissions). It is possible to do this by
specifying the DET.DIT parameter as a list of exposure times. The list
must be exactly the same length as the list of science filters. If
DET.DIT is given a single value, that exposure time will be applied to
all filters. For example

   DET.DIT            1.0;
   INS.FILTER.NAME    "H J Ks";

will expose with H, J and Ks filters, each with an exposure time of
1.0 seconds, and

   DET.DIT            "1.0 2.0 3.0";
   INS.FILTER.NAME    "H J Ks";

will expose with H filter for 1.0 seconds, the J filter for 2.0
seconds and the Ks filter for 3.0 seconds. The combination

   DET.DIT            "1.0 2.0";
   INS.FILTER.NAME    "H J Ks";

is illegal because the lists have different lengths.

Prerequirements
---------------
Telescope already pointing at reference target (VIRCAM_img_acq).
Autoguiding and active optics already started.

Data Products
-------------
DPR.CATG                 SCIENCE
DPR.TECH                 IMAGE,JITTER
DPR.TYPE                 OBJECT          When OCS.EXTENDED is F
                         OBJECT,EXTENDED When OCS.EXTENDED is T
```

```
Parameters
----------
Name                    Default         Description
----                    -------         -----------
Fixed:
SEQ.NESTING             PFJME           Nesting pattern (FIXED)
SEQ.USTEP.SCALE         1.0             Microstep scale multiplier (FIXED)

Compulsory:
DET1.DIT                10.0            Either: Detector integration time
                                        (seconds)
                                        Or: List of detector integration
                                        times corresponding to list of
                                        science filters.
INS.FILTER.NAME         (none)          List of science filter (each
                                        must be in FILTERS_SCI)
SEQ.JITTER.ID           Single          Jitter pattern ID (must be in
                                        JITTER_RANGE)
SEQ.JITTER.SCALE        1.0             Jitter scale multiplier
SEQ.USTEP.ID            Single          Microstep ID (must be in
                                        USTEP_RANGE)
OCS.EXTENDED            F               T if object is extended

Optional:
DET1.NDIT               1               Either: Number of detector
                                        integrations per exposure.
                                        Or: List of detector integrations
                                        per exposure corresponding to list
                                        of science filters.
DET1.NCORRS.NAME        Double          Detector readout mode (must be
                                        in NCORRS_RANGE)
SEQ.NEXPO               1               Number of exposures.


SEQ.JITTER.ID points to a PAF file which contains
-------------------------------------------------
SEQ.JITTER.NAME         (No default)    Name of jitter pattern
SEQ.JITTER.OFFSETX      (No default)    List of X offsets for jitter
                                        pattern
SEQ.JITTER.OFFSETY      (No default)    List of Y offsets for jitter
                                        pattern
SEQ.JITTER.OFFSETROT    0.0             (Optional) List of rotator offsets
                                        for jitter pattern.
(The number of jitter offsets is derived from the length of the
shortest list).
```

```
SEQ.USTEP.ID points to a PAF file which contains
------------------------------------------------
SEQ.USTEP.NAME          (No default)    Name of microstep pattern
SEQ.USTEP.OFFSETX       (No default)    List of X offsets for
                                        microstep pattern
SEQ.USTEP.OFFSETY       (No default)    List of Y offsets for
                                        microstep pattern
SEQ.USTEP.OFFSETROT     0.0             (Optional) List of rotator
                                        offsets for microstep pattern.
(The number of microstep offsets is derived from the length of the
shortest list).

Sequence
--------
Recall telescope pointing reference from last acquisition template.
Set detector readout mode.
For each science filter
   Select science filter
   Set detector exposure time for science filter.
   Determine telescope focus for science filter.
   For each jitter offset
      For each microstep offset.
         Convert (X,Y,ROT) offset into (ALPA,DELTA,ROT) offset
         Offset telescope to jitter and microstep offset
         Get WCS information from TCS.
         Calculate dwell time (NEXPO * DIT * NDIT) and inform TCS.
         {
            For each exposure
               Define header keywords:
                  NJITTER, JITTRNUM, JITTR_ID, JITTER_I, JITTER_X,
JITTER_Y,
                  NUSTEP, USTEPNUM, USTEP_ID, USTEP_I, USTEP_X, USTEP_Y
               Set WCS parameters.
               Make exposure
            Next exposure
         }
      Next microstep
   Next jitter
Next science filter
__oOo__
```

11.7.6.3 VIRCAM_img_obs_tile<N>

```
Description
-----------
This template makes sufficient observations to generate a contiguous
"tile", using a selection of pawprints, filter changes, jittering and
microstepping movements. There are separate variations of the tile
template signature depending on the number of pawprints in the tile
pattern (e.g. VIRCAM_img_obs_tile1, VIRCAM_img_obs_tile3 and
VIRCAM_img_obs_tile6) but each shares exactly the same template
sequencer script. The detector controller is configured with the
required readout and exposure time parameters and one of the following
sequences executed:

FPJME     Construct the tile from a series of pawprints, repeating
          each pawprint with a different science filter. Within each
          pawprint execute a jitter pattern (if specified), and within
          each jitter pattern execute a microstep pattern (if
          specified).

PFJME     Construct the tile from a series of pawprints. Within each
          pawprint execute a jitter pattern, only this time repeat
          each jitter with a different science filter before moving on
          to the next. Within each jitter execute a microstep pattern
          (if specified).

FJPME     Construct the tile from a pawprint and jitter pattern such
          that one jitter observation is made from each pawprint in
          turn. Within each jitter pattern there can be a microstep
          pattern. The whole sequence may be repeated with different
          science filters.

Each time a new pawprint is selected, the TCS is provided with a new
guide star and a new pair of aO stars, taken from the list provided
with the template. The pawprint, jitter and microstep patterns are
made using the camera position angle specified during the
VIRCAM_img_acq template, unless a new position angle is specified
here.

When a list of science filters is specified, it may be necessary to
associate a different exposure time with each filter (because the
filters have different transmissions). It is possible to do this by
specifying the DET.DIT parameter as a list of exposure times. The list
must be exactly the same length as the list of science filters.  If
DET.DIT is given a single value, that exposure time will be applied to
all filters. For example

   DET.DIT           1.0;
   INS.FILTER.NAME   "H J Ks";

will expose with H, J and Ks filters, each with an exposure time of
1.0 seconds, and

   DET.DIT           "1.0 2.0 3.0";
```

```
   INS.FILTER.NAME    "H J Ks";

will expose with H filter for 1.0 seconds, the J filter for 2.0
seconds and the Ks filter for 3.0 seconds. The combination

   DET.DIT            "1.0 2.0";
   INS.FILTER.NAME    "H J Ks";

is illegal because the lists have different lengths.

NOTE: You may see some warnings when loading this template into bob
for testing without P2PP. This is because the SEQ.REF.FILEi parameters
are only recognised properly by bob when defined by P2PP. The template
should still work.

Prerequirements
---------------
Telescope already pointing at reference target (VIRCAM_img_acq).
NOTE: It is not necessary to enable autoguiding for the reference
target. Autoguiding will be started when the first pawprint is
observed.

Data Products
-------------
DPR.CATG                 SCIENCE
DPR.TECH                 IMAGE,JITTER
DPR.TYPE                 OBJECT          When OCS.EXTENDED is F
                         OBJECT,EXTENDED When OCS.EXTENDED is T

Parameters
----------
Name                     Default          Description
----                     -------          -----------
Fixed:
SEQ.USTEP.SCALE          1.0              Microstep scale multiplier

Compulsory:
DET1.DIT                 10.0             Either: Detector integration time
                                          (seconds)
                                          Or: List of detector integration
                                          times corresponding to list of
                                          science filters.
INS.FILTER.NAME          (none)           List of science filter (each
                                          must be in FILTERS_SCI)
SEQ.NESTING              FPJME            Nesting pattern (must be in
                                          NESTING_RANGE)
SEQ.TILE.ID              Standard         Tile pattern ID
SEQ.REF.FILEi            (no default)     Pointer to PAF file containing
                                          telescope SETUP keywords defining
                                          AG and AO stars for pawprint i.
                                          (i = 1,N).
SEQ.JITTER.ID            Single           Jitter pattern ID (must be in
                                          JITTER_RANGE)
SEQ.JITTER.SCALE         1.0              Jitter scale multiplier
SEQ.USTEP.ID             Single           Microstep ID (must be in
```

```
                                                USTEP_RANGE)

Optional:
OCS.EXTENDED             F                T if object is extended
DET1.NDIT                1                Either: Number of detector
                                          integrations per exposure.
                                          Or: List of detector integrations
                                          per exposure corresponding to list
                                          of science filters.
DET1.NCORRS.NAME         Double           Detector readout mode (must be in
                                          NCORRS_RANGE)
SEQ.NEXPO                1                Number of exposures.


Each SEQ.REF.FILEi keyword points to a PAF file containing
----------------------------------------------------------
Compulsory keywords:
PAF.NAME etc..           (file name)      Header of PAF file (see bob and OHS
                                          documentation).
TPL.FILE.DIRNAME         $INS_ROOT/$INS_USER/MISC/VISTA
                                          Folder to contain PAF file.

TEL.AG.START             T                Observe with autoguiding (T or F)
TEL.AO.START             T                Observe with aO (T or F)

Minimum star candidates required when TEL.AG.START=T or TEL.AO.START=T:
TEL.GS1.ALPHA            (none)           RA of guide star candidate 1
TEL.GS1.DELTA            (none)           Dec of guide star candidate 1
TEL.GS1.MAG              (none)           Magnitude of guide star candidate 1
TEL.AOSA1.ALPHA          (none)           RA of aO star A (PY) candidate 1
TEL.AOSA1.DELTA          (none)           Dec of aO star A (PY) candidate 1
TEL.AOSA1.MAG            (none)           Magnitude of aO star A (PY)
                                          candidate 1
TEL.AOSB1.ALPHA          (none)           RA of aO star B (NY) candidate 1
TEL.AOSB1.DELTA          (none)           Dec of aO star B (NY) candidate 1
TEL.AOSB1.MAG            (none)           Magnitude of aO star B (NY)
                                          candidate 1

Optional keywords:
TEL.AG.CONFIRM           F                Confirm each new guide star?
TEL.AO.PRIORITY          NORMAL           Active optics priority (LOW, NORMAL
                                          or HIGH)

Optional additional star candidates:
TEL.GSi.ALPHA            (none)           RA of guide star candidate i
                                          (i=2-5)
TEL.GSi.DELTA            (none)           Dec of guide star candidate i
                                          (i=2-5)
TEL.GSi.MAG              (none)           Magnitude of guide star candidate
                                          i (i=2-5)
TEL.AOSAi.ALPHA          (none)           RA of aO star A (PY) candidate i
                                          (i=2-5)
TEL.AOSAi.DELTA          (none)           Dec of aO star A (PY) candidate i
                                          (i=2-5)
TEL.AOSAi.MAG            (none)           Magnitude of aO star A (PY)
```

```
                                        candidate i (i=2-5)
TEL.AOSBi.ALPHA         (none)          RA of aO star B (NY) candidate i
                                        (i=2-5)
TEL.AOSBi.DELTA         (none)          Dec of aO star B (NY) candidate i
                                        (i=2-5)
TEL.AOSBi.MAG           (none)          Magnitude of aO star B (NY)
                                        candidate i (i=2-5)

SEQ.TILE.ID points to a PAF file which contains
-----------------------------------------------
SEQ.TILE.NAME           (No default)    Name of tile pattern
SEQ.TILE.OFFSETX        (No default)    List of X offsets for tile pattern
SEQ.TILE.OFFSETY        (No default)    List of Y offsets for tile pattern
SEQ.TILE.OFFSETROT     0.0              (Optional) List of rotator offsets
                                        for tile pattern
(The number of pawprints is derived from the length of the shortest
list).

SEQ.JITTER.ID points to a PAF file which contains
-------------------------------------------------
SEQ.JITTER.NAME         (No default)    Name of jitter pattern
SEQ.JITTER.OFFSETX      (No default)    List of X offsets for jitter
                                        pattern
SEQ.JITTER.OFFSETY      (No default)    List of Y offsets for jitter
                                        pattern
SEQ.JITTER.OFFSETROT   0.0              (Optional) List of rotator offsets
                                        for jitter pattern
(The number of jitter offsets is derived from the length of the
shortest list).

SEQ.USTEP.ID points to a PAF file which contains
------------------------------------------------
SEQ.USTEP.NAME          (No default)    Name of microstep pattern
SEQ.USTEP.OFFSETX       (No default)    List of X offsets for
                                        microstep pattern
SEQ.USTEP.OFFSETY       (No default)    List of Y offsets for
                                        microstep pattern
SEQ.USTEP.OFFSETROT    0.0              (Optional) List of rotator
                                        offsets for microstep pattern
(The number of microstep offsets is derived from the length of the
shortest list).
```

```
Sequence
--------
Recall telescope pointing reference from last acquisition template.
Set detector readout mode.
If SEQ.NESTING is FPJME then
   For each science filter
      Select science filter
      Set detector exposure time for science filter.
      Determine telescope focus for science filter.
      For each pawprint
         {
            If SEQ.REF.FILE($pawprint) is not a blank or null string then
               If SEQ.REF.FILE($pawprint) file exists then
                  Define new guide star setup parameters from
                  SEQ.REF.FILE($pawprint).
               Else
                  Issue warning and define new guide star setup
                  parameters to select stars on the fly from
                  online catalogue.
               Endif
            Else
               (Keep previously defined stars).
            Endif
         }
         For each jitter offset
            For each microstep offset.
               Convert (X,Y,ROT) offset into (ALPA,DELTA,ROT) offset
               Offset telescope to pawprint, jitter and microstep offset
               If new guide stars are available then
                  {
                     If TEL.AG.START is TRUE then
                        If AG.CONFIRM is TRUE then
                           Prompt operator to confirm autoguiding.
                        End if
                        Wait for autoguiding to start
                     End if
                     If TEL.AO.START is TRUE and AO.PRIORITY > 0 then
                        Wait for active optics to start
                     End if
                  }
               Endif
               Get WCS information from TCS.
               Calculate dwell time (NEXPO * DIT * NDIT) and inform TCS.
               {
                  For each exposure
                     Define header keywords:
                        TILE_ID, TILE_I, TILENUM
                        NJITTER, JITTRNUM, JITTR_ID, JITTER_I, JITTER_X,
JITTER_Y,
                        NUSTEP, USTEPNUM, USTEP_ID, USTEP_I, USTEP_X,
USTEP_Y
                     Set WCS parameters.
                     Make exposure
                  Next exposure
               }
```

```
            Next microstep
         Next jitter
      Next pawprint
   Next science filter

Else if SEQ.NESTING is PFJME then
   For each pawprint
      {
         If SEQ.REF.FILE($pawprint) is not a blank or null string then
            If SEQ.REF.FILE($pawprint) file exists then
               Define new guide star setup parameters from
               SEQ.REF.FILE($pawprint).
            Else
               Issue warning and define new guide star setup
               parameters to select stars on the fly from
               online catalogue.
            Endif
         Else
            (Keep previously defined stars).
         Endif
      }
      For each science filter
         Select science filter
         Set detector exposure time for science filter.
         Determine telescope focus for science filter.
         For each jitter offset
            For each microstep offset.
               Convert (X,Y,ROT) offset into (ALPA,DELTA,ROT) offset
               Offset telescope to pawprint, jitter and microstep offset
               If new guide stars are available then
                  {
                     If TEL.AG.START is TRUE then
                        If AG.CONFIRM is TRUE then
                           Prompt operator to confirm autoguiding.
                        End if
                        Wait for autoguiding to start
                     End if
                     If TEL.AO.START is TRUE and AO.PRIORITY > 0 then
                        Wait for active optics to start
                     End if
                  }
               Endif
               Get WCS information from TCS.
               Calculate dwell time (NEXPO * DIT * NDIT) and inform TCS.
               {
                  For each exposure
                     Define header keywords:
                        TILE_ID, TILE_I, TILENUM
                        NJITTER, JITTRNUM, JITTR_ID, JITTER_I, JITTER_X,
JITTER_Y,
                        NUSTEP, USTEPNUM, USTEP_ID, USTEP_I, USTEP_X,
USTEP_Y
                     Set WCS parameters.
                     Make exposure
                  Next exposure
```

```
              }
            Next microstep
         Next jitter
      Next science filter
   Next pawprint

Else if SEQ.NESTING is FJPME then
   For each science filter
      Select science filter
      Set detector exposure time for science filter.
      Determine telescope focus for science filter.
      For each jitter offset
         For each pawprint
            {
               If SEQ.REF.FILE($pawprint) is not a blank or null string
then
                  If SEQ.REF.FILE($pawprint) file exists then
                     Define new guide star setup parameters from
                     SEQ.REF.FILE($pawprint).
                  Else
                     Issue warning and define new guide star setup
                     parameters to select stars on the fly from
                     online catalogue.
                  Endif
               Else
                  (Keep previously defined stars).
               Endif
            }
            For each microstep offset.
               Convert (X,Y,ROT) offset into (ALPA,DELTA,ROT) offset
               Offset telescope to jitter, pawprint and microstep offset
               If new guide stars are available then
                  {
                     If TEL.AG.START is TRUE then
                        If AG.CONFIRM is TRUE then
                           Prompt operator to confirm autoguiding.
                        End if
                        Wait for autoguiding to start
                     End if
                     If TEL.AO.START is TRUE and AO.PRIORITY > 0 then
                        Wait for active optics to start
                     End if
                  }
               Endif

               Get WCS information from TCS.
               Calculate dwell time (NEXPO * DIT * NDIT) and inform TCS.
               {
                  For each exposure
                     Define header keywords:
                        TILE_ID, TILE_I, TILENUM
                        NJITTER, JITTRNUM, JITTR_ID, JITTER_I, JITTER_X,
JITTER_Y,
                        NUSTEP, USTEPNUM, USTEP_ID, USTEP_I, USTEP_X,
USTEP_Y
```

```
                    Set WCS parameters.
                    Make exposure
                 Next exposure
              }
          Next microstep
       Next pawprint
    Next jitter
 Next science filter

End if
```

__oOo__

11.7.7 Technical templates

11.7.7.1 VIRCAM_img_acq_random

```
Description
-----------
This template is used to select a random sky field. It sets the
instrument into IMAGING mode and (if one has been specified) selects a
science filter and points the telescope to a random part of the sky
within the defined altitude and azimuth range.

Parameters
----------
Name                    Default          Description
----                    -------          -----------
Fixed:
INS.MODE                IMAGING          Instrument mode (FIXED).
TEL.AG.GUIDESTAR        CATALOGUE        Guide star must be chosen on the
                                         fly
TEL.AO.AOSTARA          CATALOGUE        aO star must be chosen on the fly
TEL.AO.AOSTARB          CATALOGUE        aO star must be chosen on the fly

Compulsory:
INS.FILTER.NAME         (none)           Name of science filter (must be
                                         contained in FILTERS_SCI, or
                                         can be blank).
TEL.AG.START            T                Observe with autoguiding (T or F)
TEL.AO.START            T                Observe with aO (T or F)

Optional:
TEL.ALTMIN              24.0             Minimum target altitude.
TEL.ALTMAX              85.0             Maximum target altitude.
TEL.MOON.AVOID          20.0             Moon avoidance in degrees.

TEL.ROT.ENABLED         T                Rotator preset enabled
TEL.ROT.OFFANGLE        0.0              Camera sky position angle (0-360
                                         degrees).
TEL.TARG.EQUINOX        2000.0           Target equinox
TEL.TARG.ADDVELALPHA    0.0              Target drift in RA (optional)
TEL.TARG.ADDVELDELTA    0.0              Target drift in Dec (optional)
TEL.TARG.PMA            0.0              Target proper motion in RA
                                         (optional)
TEL.TARG.PMD            0.0              Target proper motion in Dec
                                         (optional)
TEL.TARG.EPOCH          2000             Target epoch
TEL.TARG.EPOCHSYSTEM    J                Target epoch system
TEL.TARG.X              0.0              Pointing origin X in mm (optional)
TEL.TARG.Y              0.0              Pointing origin Y in mm (optional)

TEL.AG.CONFIRM          F                Confirm each new guide star?
TEL.AO.CONFIRM          F                Confirm active optics before
                                         starting?
TEL.AO.PRIORITY         NORMAL           Active optics priority (LOW, NORMAL
```

```
                                        or HIGH)

Sequence
--------
Set instrument mode to IMAGING.
If science filter has been specified
   Select science filter.
End if
Query local time, local sidereal time, observatory latitude,longitude
and moon RA,Dec from TCS
Generate a random (azimuth,altitude) and convert it into (RA,Dec)
Preset telescope to (Ra,Dec).
If science filter has been specified
   Adjust telescope focus for science filter.
End if
If autoguiding is enabled then
   If AG.CONFIRM is TRUE then
      Prompt operator to confirm autoguiding.
   End if
   Wait for autoguiding to start
End if
If active optics are enabled and AO.PRIORITY is > 0 then
   Wait for active optics to start
End if
__oOo__
```

11.7.7.2 VIRCAM_gen_tec_StrayLight

```
Description
-----------
This template carries out an automatic stray light investigation by
taking several exposures with the filter wheel offset from the central
position by differing amounts. If any stray light pattern results from
a reflection from a component mounted on the filter wheel, the
reflection will be seen to move.

The same template can also be used to verify that the filter wheel is
moving, to check the vignetting limits of the filter wheel, and to
test the orientation of the detectors with respect to the filter
wheel.

Prerequirements
---------------
A suitable target (dome screen or star field) has been acquired.

Data Products
-------------
DPR.CATG                TEST
DPR.TECH                IMAGE,FILTOFFSET
DPR.TYPE                STD,STRAYLIGHT
```

```
Parameters
----------
Name                    Default         Description
----                    -------         -----------
Compulsory:
INS.FILTER.NAME         H               Filter to be used
                                        (must be in FILTERS).
DET.DIT                 10.0            Detector integration time in
                                        seconds.

Optional:
INS.FILTER.OFFSET       -2000 -1000 0 1000 2000
                                        List of filter wheel offsets
                                        in motor steps.
DET1.NDIT               1               Number of detector integrations
                                        per exposure.
SEQ.NEXPO               1               Number of exposures at each
                                        filter wheel offset.

Sequence
--------
Select science filter
Adjust telescope focus for science filter
Make exposure
For each filter wheel offset
   Move filter wheel to offset
   Make exposure
Next offset
__oOo__
```

11.7.7.3 VIRCAM_gen_tec_CalibFilter

```
Description
-----------
This template checks the repeatability of the filter wheel by making
an exposure with a particular filter, moving the filter wheel by a
given offset then returning it to the same filter, then making a
second exposure. Any offset between the images in the two exposures
gives a measure of the filter wheel repeatability.

Prerequirements
---------------
A suitable target (dome screen or star field) has been acquired.

Data Products
-------------
DPR.CATG                TEST
DPR.TECH                IMAGE,FILTOFFSET
DPR.TYPE                STD,FILTCALIB
```

```
Parameters
----------
Name                    Default         Description
----                    -------         -----------
Compulsory:
INS.FILTER.NAME         (none)          Filter to be used
                                        (must be in FILTERS)
DET.DIT                 10.0            Detector integration time in
                                        seconds.

Optional:
INS.FILTER.OFFSET       40000           Filter wheel offset in motor steps
SEQ.WAIT                2               Time to wait after offsetting
                                        filter wheel.
DET1.NDIT               1               Number of detector integrations
                                        per exposure.
SEQ.NEXPO               1               Number of pairs of exposures.

Sequence
--------
Select science filter
Adjust telescope focus for science filter
For each pair of exposures (SEQ.NEXPO)
   Make exposure
   Move filter wheel by INS.FILTER.OFFSET steps
   Wait for SEQ.WAIT seconds
   Move filter wheel by -INS.FILTER.OFFSET steps
   Make exposure
Next pair of exposures
__oOo__
```

11.7.7.4 VIRCAM_gen_tec_CheckFilters

```
Description
-----------
This template checks the functioning of the filter wheel by selecting
a list of science filters.
NOTE: This template only gives a rough idea of filter wheel
performance. There are more detailed test scripts within the vci
module.

Prerequirements
---------------
None.

Data Products
-------------
None. No data generated.
```

```
Parameters
----------
Name                    Default          Description
----                    -------          -----------
Optional:
INS.FILTER.NAME         FILTERS          List of filters (each
                                         must be in FILTERS)
SEQ.REPEATS             1                Number of repetitions.
SEQ.RANDOMIZE           T                Randomize filter sequence?
SEQ.WAIT                5                Time to wait at each filter
                                         (in seconds)

Sequence
--------
For each repetition (SEQ.REPEATS)
   If SEQ.RANDOMIZE is T then
      Shuffle filter list into random order.
   Endif
   For each filter in the list
      If the filter is installed in the instrument then
         Select science filter (and check switches).
         Report length of movement and filter exchange time.
         Wait SEQ.WAIT seconds.
      End if
   Next filter
Next repetition
Report fastest and slowest filter exchange time.
__oOo__
```

11.7.7.5 VIRCAM_gen_tec_FocusFilters

```
Description
-----------
This template derives the best telescope focus offset for a science filter,
or list of science filters. Several exposures are made at different focus
offsets, and a MIDAS task is used to derive the best focus offset for
each filter.

Prerequirements
---------------
Telescope already pointing at a field of well-separated stars
(VIRCAM_img_acq)

Data Products
-------------
DPR.CATG                TEST
DPR.TECH                IMAGE,TEL-THROUGH
DPR.TYPE                OBJECT,FOCUS
```

```
Parameters
----------
Name                     Default          Description
----                     -------          -----------
Compulsory:
INS.FILTER.NAME          (none)           List of science filters (each
                                          must be in FILTERS_SCI)
DET.DIT                  10.0             Detector integration time in
                                          seconds.
TEL.INITFOCUS           -5.0              Initial telescope focus offset
                                          in mm.
TEL.FOCUSSTEP            1.0              Telescope focus step in mm.
SEQ.NEXPO                10               Number of focus exposures.

Optional:
DET1.NDIT                1                Number of detector integrations
                                          per exposure.
DET1.NCORRS.NAME         Double           Detector readout mode (must be
                                          in NCORRS_RANGE)
DET1.CHIPS               1..16            List of unvignetted chips
DET1.WIN.STARTX          768              Starting column for window in
                                          pixels
DET1.WIN.STARTY          768              Starting row for window in pixels
DET1.WIN.NX              512              X size of window in pixels
DET1.WIN.NY              512              Y size of window in pixels

Sequence
--------
For each science filter
   Select science filter
   For each telescope focus offset
      Adjust telescope focus offset
      Set detector exposure time, window and readout mode.
      Set WCS parameters to "pixel coordinates".
      Make exposure.
   Next focus offset
   Calculate the best focus offset for this filter.
Next science filter
__oOo__
```

11.7.7.6 VIRCAM_gen_tec_LoadFilters

```
Description
-----------
This template takes an engineer through the procedure to load a series
of filters. The template prompts the engineer to provide the name and
properties of the new filters being installed and uses this
information to build a new filter wheel configuration file.

Prerequirements
---------------
Camera oriented so that access to the filter load hatch is possible.
A means of physically disabling the filter wheel while working should
be in place.
```

```
Data Products
-------------
None. No data generated.

Parameters
----------
Name                    Default         Description
----                    -------         -----------
Compulsory:
INS.FILTER.LOAD         FILTERS         List of filters to be replaced
                                        (each must be in FILTERS).

Sequence
--------
For each filter in the list

   If the filter is installed in the instrument then
      Move the filter to the load position.
      Put filter wheel in STANDBY state to disable movement commands.
      Prompt the engineer to exchange the filter (recommending that
      the motor be disabled for safety).
      Prompt the engineer for the new filter name, focus, wavelength and
      transmission and update the instrument configuration.
      Put filter wheel back into the ONLINE state.
   End if
Next filter
__oOo__
```

11.7.7.7 VIRCAM_gen_tec_exp

```
Description
-----------
This template makes a series of exposures designed to test the science
detectors. It also times the sequence and reports performance statistics.

Prerequirements
---------------
Telescope already pointing at target, if a target is needed.

Data Products
-------------
DPR.CATG                TEST
DPR.TECH                IMAGE
DPR.TYPE                TEST

Parameters
----------
Name                    Default         Description
----                    -------         -----------
Compulsory:
DET1.DIT                10.0            Detector integration time (seconds)
INS.FILTER.NAME         (none)          Filter to be used (must be in
                                        FILTERS)
```

```
Optional:
DET1.NDIT              1              Number of detector integrations
                                      per exposure.
DET1.NCORRS.NAME       Double         Detector readout mode (must be
                                      in NCORRS_RANGE).
DET1.CHIPS             {1 2 3.. 16}   Space separated list of unvignetted
                                      chips.
DET1.WIN.STARTX        1              Starting column for window in
                                      pixels
DET1.WIN.STARTY        1              Starting row for window in pixels
DET1.WIN.NX            2048           X size of window in pixels
DET1.WIN.NY            2048           Y size of window in pixels
SEQ.NEXPO              1              Number of exposures.

Sequence
--------
Select filter.
Set detector exposure time, window and readout mode.
Identify unvignetted detector chips.
Start timer
   For exposure = 1 to SEQ.NEXPO
      Set WCS parameters to "pixel coordinates".
      Make exposure.
   Next exposure
Report elapsed time and calculate overheads.
__oOo__
```

11.7.7.8 VIRCAM_howfs_tec_loopback

```
Description
-----------
This technical template generates a set of loopback data containing
the NULL coefficients associated with each HOWFS filter. The filter
parameters are obtained from VIRCAM_HOWFS*.paf files in the directory
$INS_ROOT/$INS_USER/MISC/VISTA/VIRCAM_HOWFS.

The loopback data files are written to the directory
$INS_ROOT/$INS_USER/HOWFSDATA.

Prerequirements
---------------
None.

Data Products
-------------
Not applicable for HOWFS loopback data.
```

```
Parameters
----------
Name                    Default          Description
----                    -------          -----------
Compulsory:
INS.FILTER.NAME         FILTERS_HOWFS    List of HOWFS filters (must be
                                         in FILTERS_HOWFS)

Optional:
None.

Sequence
--------
Check whether HOWFS server is busy and ensure it is ONLINE.
For each HOWFS filter
   Configure the HOWFS image analysis server for this filter.
   Instruct HOWFS image analysis to generate loop-back data.
Next filter
Put the HOWFS server into the STANDBY state.
__oOo__
```

11.8 Pattern Files

The following pattern files are delivered with the VIRCAM software, but there is no reason why more patterns can't be added as and when needed.

11.8.1 Tile patterns

11.8.1.1 VIRCAM_Tile3nx

```
SEQ.TILE.NAME           "3 step nx (negative x) pattern";
SEQ.TILE.OFFSETX        "-0.475 -0.475 -0.475";
SEQ.TILE.OFFSETY        "-0.475  0.0    0.475";
SEQ.TILE.OFFSETROT      " 0.0    0.0    0.0  ";
```

11.8.1.2 VIRCAM_Tile3px

```
SEQ.TILE.NAME           "3 step px (positive x) pattern";
SEQ.TILE.OFFSETX        " 0.475  0.475  0.475";
SEQ.TILE.OFFSETY        "-0.475  0.0    0.475";
SEQ.TILE.OFFSETROT      " 0.0    0.0    0.0  ";
```

11.8.1.3 VIRCAM_Tile6n

```
SEQ.TILE.NAME           "6 step n pattern";
SEQ.TILE.OFFSETX        "-0.475 -0.475 -0.475 0.475 0.475  0.475";
SEQ.TILE.OFFSETY        "-0.475  0.0    0.475 0.475 0.0   -0.475";
SEQ.TILE.OFFSETROT      " 0.0    0.0    0.0   0.0   0.0    0.0";
```

11.8.1.4 VIRCAM_Tile6s

```
SEQ.TILE.NAME           "6 step large S pattern";
SEQ.TILE.OFFSETX        " 0.475 -0.475 -0.475 0.475 0.475 -0.475";
SEQ.TILE.OFFSETY        " 0.475  0.475  0.0   0.0  -0.475 -0.475";
SEQ.TILE.OFFSETROT      " 0.0    0.0    0.0   0.0   0.0    0.0";
```

11.8.1.5 VIRCAM_Tile6ss

```
SEQ.TILE.NAME           "6 step ss zig zag pattern";
SEQ.TILE.OFFSETX        "-0.475  0.475 -0.475 0.475 -0.475  0.475";
SEQ.TILE.OFFSETY        "-0.475 -0.475  0.0   0.0    0.475  0.475";
SEQ.TILE.OFFSETROT      " 0.0    0.0    0.0   0.0    0.0    0.0";
```

11.8.1.6 VIRCAM_Tile6u

```
SEQ.TILE.NAME           "6 step u pattern";
SEQ.TILE.OFFSETX        "-0.475 -0.475 -0.475  0.475 0.475 0.475";
SEQ.TILE.OFFSETY        " 0.475  0.0   -0.475 -0.475 0.0   0.475";
SEQ.TILE.OFFSETROT      " 0.0    0.0    0.0    0.0   0.0   0.0";
```

11.8.1.7 VIRCAM_Tile6z

```
SEQ.TILE.NAME           "6 step large Z pattern";
SEQ.TILE.OFFSETX        "-0.475  0.475  0.475 -0.475 -0.475   0.475";
SEQ.TILE.OFFSETY        " 0.475  0.475  0.0    0.0   -0.475  -0.475";
```

```
SEQ.TILE.OFFSETROT      " 0.0    0.0    0.0    0.0    0.0     0.0";
```

11.8.1.8 VIRCAM_Tile6zz

```
SEQ.TILE.NAME           "6 step zz zig zag pattern";
SEQ.TILE.OFFSETX        " 0.475 -0.475  0.475 -0.475 0.475 -0.475";
SEQ.TILE.OFFSETY        "-0.475 -0.475  0.0    0.0   0.475  0.475";
SEQ.TILE.OFFSETROT      " 0.0    0.0    0.0    0.0   0.0    0.0";
```

11.8.1.9 VIRCAM_TileXTalk

A special tile pattern used by the VIRCAM_img_cal_crosstalk template. This file defines a special tile pattern (also known as a dither pattern) for the VISTA IR Camera, which defines the offsets needed to place a star in the same position on each of the 16 detectors.

```
SEQ.TILE.NAME           "Cross talk pattern";
SEQ.TILE.OFFSETX        "0.0 1.9 3.8 5.7 0.0   1.9   3.8   5.7   0.0  1.9
3.8  5.7  0.0   1.9   3.8   5.7";
SEQ.TILE.OFFSETY        "0.0 0.0 0.0 0.0 1.425 1.425 1.425 1.425 2.85 2.85
2.85 2.85 4.275 4.275 4.275 4.275";
SEQ.TILE.OFFSETROT      "0.0 0.0 0.0 0.0 0.0   0.0   0.0   0.0   0.0  0.0
0.0  0.0  0.0   0.0   0.0   0.0";
```

11.8.2 Jitter patterns

11.8.2.1 VIRCAM_Jitter2d

```
SEQ.JITTER.NAME        "2 step / down pattern";
SEQ.JITTER.OFFSETX     "10.0 -10.0";
SEQ.JITTER.OFFSETY     "10.0 -10.0";
SEQ.JITTER.OFFSETROT   " 0.0   0.0";
```

11.8.2.2 VIRCAM_Jitter2u

```
SEQ.JITTER.NAME        "2 step / up pattern";
SEQ.JITTER.OFFSETX     "-10.0  10.0";
SEQ.JITTER.OFFSETY     "-10.0  10.0";
SEQ.JITTER.OFFSETROT   "  0.0   0.0";
```

11.8.2.3 VIRCAM_Jitter2x2

```
SEQ.JITTER.NAME        "2x2 |\| pattern";
SEQ.JITTER.OFFSETX     "-10.0 -10.0   10.0  10.0";
SEQ.JITTER.OFFSETY     "-10.0  10.0  -10.0  10.0";
SEQ.JITTER.OFFSETROT   "  0.0   0.0    0.0   0.0";
```

11.8.2.4 VIRCAM_Jitter3d

```
SEQ.JITTER.NAME        "3 step / down pattern";
SEQ.JITTER.OFFSETX     "10.0   0.0 -10.0";
SEQ.JITTER.OFFSETY     "10.0   0.0 -10.0";
SEQ.JITTER.OFFSETROT   " 0.0   0.0   0.0";
```

11.8.2.5 VIRCAM_Jitter3u

```
SEQ.JITTER.NAME        "3 step / up pattern";
SEQ.JITTER.OFFSETX     "-10.0   0.0  10.0";
SEQ.JITTER.OFFSETY     "-10.0   0.0  10.0";
SEQ.JITTER.OFFSETROT   "  0.0   0.0   0.0";
```

11.8.2.6 VIRCAM_Jitter3x3

```
SEQ.JITTER.NAME        "3x3 [] + pattern";
SEQ.JITTER.OFFSETX     " 0.0 -10.0 -10.0  10.0  10.0 -10.0   0.0  10.0
0.0";
SEQ.JITTER.OFFSETY     " 0.0 -10.0  10.0  10.0 -10.0   0.0  10.0   0.0
-10.0";
SEQ.JITTER.OFFSETROT   " 0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0
0.0";
```

11.8.2.7 VIRCAM_Jitter5p

```
SEQ.JITTER.NAME        "5 step + pattern";
SEQ.JITTER.OFFSETX     " 0.0 -10.0   0.0  10.0   0.0";
SEQ.JITTER.OFFSETY     " 0.0   0.0  10.0   0.0 -10.0";
SEQ.JITTER.OFFSETROT   " 0.0   0.0   0.0   0.0   0.0";
```

11.8.2.8 VIRCAM_Jitter5x

```
SEQ.JITTER.NAME        "5 step X pattern";
SEQ.JITTER.OFFSETX     "0.0 -10.0 -10.0  10.0 -10.0";
SEQ.JITTER.OFFSETY     "0.0 -10.0  10.0  10.0 -10.0";
SEQ.JITTER.OFFSETROT   "0.0   0.0   0.0   0.0   0.0";
```

11.8.2.9 VIRCAM_Jitter5x5

```
SEQ.JITTER.NAME        "5x5 spiral pattern";
SEQ.JITTER.OFFSETX     "0 -10 -10   0  10  10  10   0 -10 -20 -20 -20 -20
-10   0 10 20 20 20  20  20  10   0 -10 -20";
SEQ.JITTER.OFFSETY     "0   0 -10  10  10   0 -10 -10 -10 -10   0  10  20
20 20 20 20 10   0 -10 -20 -20 -20 -20 -20";
SEQ.JITTER.OFFSETROT   "0   0   0   0   0   0   0   0   0   0   0   0   0
0  0  0  0  0  0   0   0   0   0   0";
```

11.8.2.10 VIRCAM_JitterXTalk

A special jitter pattern used by the VIRCAM_img_cal_crosstalk template. This file defines a special jitter pattern which defines the sequence of offsets needed to place the image of a star on each of the 16 readout sectors of a single detector.

```
SEQ.JITTER.NAME        "Cross talk jitter pattern";
SEQ.JITTER.OFFSETX     "0.0  38.4 76.8 115.2 153.6 192.0 230.4 268.8 307.2
345.6 384.0 422.4 460.8 499.2 537.6 576.0";
SEQ.JITTER.OFFSETY     "0.0   0.0  0.0   0.0   0.0   0.0   0.0   0.0   0.0
0.0   0.0   0.0   0.0   0.0   0.0   0.0";
SEQ.JITTER.OFFSETROT   "0.0   0.0  0.0   0.0   0.0   0.0   0.0   0.0   0.0
0.0   0.0   0.0   0.0   0.0   0.0   0.0";
```

11.8.3 Microstep patterns

11.8.3.1 VIRCAM_Ustep2

```
SEQ.USTEP.NAME        "2 step / pattern";
SEQ.USTEP.OFFSETX     "0.0 0.5";
SEQ.USTEP.OFFSETY     "0.0 0.5";
SEQ.USTEP.OFFSETROT   "0.0 0.0";
```

11.8.3.2 VIRCAM_Ustep2x2

```
SEQ.USTEP.NAME        "2x2 pattern";
SEQ.USTEP.OFFSETX     "0.0 0.0 0.5 0.5";
SEQ.USTEP.OFFSETY     "0.0 0.5 0.5 0.0";
SEQ.USTEP.OFFSETROT   "0.0 0.0 0.0 0.0";
```

11.9 FITS Files

Important header entries and items specific to VISTA are highlighted.

11.9.1 Example of top level FITS header

```
SIMPLE  =                    T / Standard FITS (NOST-100-2.0)
BITPIX  =                    8 / # of bits per pix value
NAXIS   =                    0 / # of axes in data array
EXTEND  =                    T / FITS Extension may be present
ORIGIN  = 'ESO     '                    / European Southern Observatory
DATE    = '2007-01-16T17:52:40'         / Date this file was written
TELESCOP= 'VISTA   '                    / ESO Telescope Name
INSTRUME= 'VIRCAM  '                    / Instrument used.
OBJECT  = 'Bit of the sky'     / Original target.
RA      =           320.221083          / 21:20:53.0 RA (J2000) pointing (deg)
DEC     =            -88.71108          / -88:42:39.8 DEC (J2000) pointing (deg
EQUINOX =                2000.          / Standard FK5 (years)
RADECSYS= 'FK5     '                    / Coordinate reference frame
EXPTIME =           10.0000000          / Integration time
MJD-OBS =       54116.74490925          / Obs start
DATE-OBS= '2007-01-16T17:52:40.1590'    / Observing date
UTC     =            64353.682          / 17:52:33.682 UTC at start (sec)
LST     =            75247.396          / 20:54:07.396 LST at start (sec)
PI-COI  = 'G.Dalton-W.Sutherland' / PI-COI name.
OBSERVER= 'S.Beard-S.McLay'    / Name of observer.
ORIGFILE= 'VIRCAM_IMG_OBS016_0002.fits' / Original File Name
COMMENT VISTA IR Camera OS $Revision: 0.29 $
HIERARCH ESO ADA ABSROT END  =       0.00000 / Abs rot angle at exp end (deg)
HIERARCH ESO DET DIT         =    10.0000000 / Integration Time
HIERARCH ESO DET NCORRS NAME = 'Double  '    / Read-Out Mode Name
HIERARCH ESO DET NDIT        =             1 / # of Sub-Integrations
HIERARCH ESO DPR CATG        = 'SCIENCE '    / Observation category
HIERARCH ESO DPR TECH        = 'IMAGE,JITTER' / Observation technique
HIERARCH ESO DPR TYPE        = 'OBJECT,EXTENDED' / Observation type
HIERARCH ESO INS DATE        = '2007-01-09' / Instrument release date (yyyy-mm-dd)
HIERARCH ESO INS FILT1 DATE  = '2007-01-16T15:20:09' / Filter index time
HIERARCH ESO INS FILT1 ENC   =        100850 / Filter wheel abs position [Enc]
HIERARCH ESO INS FILT1 ERROR =          22.0 / Filter home switch offset [Enc]
HIERARCH ESO INS FILT1 FOCUS =        -0.300 / Filter focus offset [mm]
HIERARCH ESO INS FILT1 ID    = 'SLOT5   '    / Filter slot name
HIERARCH ESO INS FILT1 NAME  = 'J       '    / Filter name
HIERARCH ESO INS FILT1 NO    =            19 / Filter wheel position index
HIERARCH ESO INS FILT1 POSEDGE=      -102296 / In-position switch edge [Enc]
HIERARCH ESO INS FILT1 TRAYID= 'ESO_J_0001' / Filter tray ID
HIERARCH ESO INS FILT1 WLEN  =      1250.000 / Filter effective wavelength [nm]
HIERARCH ESO INS HB1 SWSIM   =             F / If T, heart beat device simulated
HIERARCH ESO INS ID          = 'VIRCAM/1.56' / Instrument ID
HIERARCH ESO INS LSC1 OK     =             T / If T, controller is operational
HIERARCH ESO INS LSC1 SWSIM  =             F / If T, lakeshore ctrllr simulated
HIERARCH ESO INS LSM1 OK     =             T / If T, controller is operational
HIERARCH ESO INS LSM1 SWSIM  =             F / If T, lakeshore monitor simulated
HIERARCH ESO INS LSM2 OK     =             T / If T, controller is operational
HIERARCH ESO INS LSM2 SWSIM  =             T / If T, lakeshore monitor simulated
HIERARCH ESO INS LSM3 OK     =             T / If T, controller is operational
HIERARCH ESO INS LSM3 SWSIM  =             T / If T, lakeshore monitor simulated
HIERARCH ESO INS PRES1 ID    = 'Vac1    '    / Pressure sensor type
HIERARCH ESO INS PRES1 NAME  = 'Vacuum gauge 1' / Pressure sensor name
HIERARCH ESO INS PRES1 UNIT  = 'mbar    '    / Pressure unit
```

```
HIERARCH ESO INS PRES1 VAL   =         0.001 / Pressure [mbar]
HIERARCH ESO INS SW1 ID      = 'INPOS    '    / Switch ID
HIERARCH ESO INS SW1 NAME    = 'Filter In-position Switch' / Switch name
HIERARCH ESO INS SW1 STATUS  = 'ACTIVE   '    / Switch status
HIERARCH ESO INS SW2 ID      = 'REFSW    '    / Switch ID
HIERARCH ESO INS SW2 NAME    = 'Filter Reference Select' / Switch name
HIERARCH ESO INS SW2 STATUS  = 'PRIMARY '    / Switch status
HIERARCH ESO INS SW3 ID      = 'HOME     '    / Switch ID
HIERARCH ESO INS SW3 NAME    = 'Filter Reference Switch' / Switch name
HIERARCH ESO INS SW3 STATUS  = 'INACTIVE'    / Switch status
HIERARCH ESO INS TEMP1 ID    = 'Amb      '    / Temperature sensor type
HIERARCH ESO INS TEMP1 NAME  = 'Ambient temperature' / Temperature sensor name
HIERARCH ESO INS TEMP1 UNIT  = 'K        '    / Temperature unit
HIERARCH ESO INS TEMP1 VAL   =       299.320 / Temperature [K]
HIERARCH ESO INS TEMP10 ID   = 'CC1_2    '    / Temperature sensor type
HIERARCH ESO INS TEMP10 NAME = 'Cryo cooler 1 2nd' / Temperature sensor name
HIERARCH ESO INS TEMP10 UNIT = 'K        '    / Temperature unit
HIERARCH ESO INS TEMP10 VAL  =        30.000 / Temperature [K]
HIERARCH ESO INS TEMP12 ID   = 'CC2_2    '    / Temperature sensor type
HIERARCH ESO INS TEMP12 NAME = 'Cryo cooler 2 2nd' / Temperature sensor name
HIERARCH ESO INS TEMP12 UNIT = 'K        '    / Temperature unit
HIERARCH ESO INS TEMP12 VAL  =        30.000 / Temperature [K]
HIERARCH ESO INS TEMP14 ID   = 'CC3_2    '    / Temperature sensor type
HIERARCH ESO INS TEMP14 NAME = 'Cryo cooler 3 2nd' / Temperature sensor name
HIERARCH ESO INS TEMP14 UNIT = 'K        '    / Temperature unit
HIERARCH ESO INS TEMP14 VAL  =        30.000 / Temperature [K]
HIERARCH ESO INS TEMP15 ID   = 'WFS1     '    / Temperature sensor type
HIERARCH ESO INS TEMP15 NAME = 'WFS CCD assembly PY' / Temperature sensor name
HIERARCH ESO INS TEMP15 UNIT = 'K        '    / Temperature unit
HIERARCH ESO INS TEMP15 VAL  =       100.000 / Temperature [K]
HIERARCH ESO INS TEMP16 ID   = 'WFS2     '    / Temperature sensor type
HIERARCH ESO INS TEMP16 NAME = 'WFS CCD assembly NY' / Temperature sensor name
HIERARCH ESO INS TEMP16 UNIT = 'K        '    / Temperature unit
HIERARCH ESO INS TEMP16 VAL  =       100.000 / Temperature [K]
HIERARCH ESO INS TEMP17 ID   = 'Dt1AB    '    / Temperature sensor type
HIERARCH ESO INS TEMP17 NAME = 'Science detector 1AB' / Temperature sensor name
HIERARCH ESO INS TEMP17 UNIT = 'K        '    / Temperature unit
HIERARCH ESO INS TEMP17 VAL  =        72.020 / Temperature [K]
HIERARCH ESO INS TEMP18 ID   = 'Dt1CD    '    / Temperature sensor type
HIERARCH ESO INS TEMP18 NAME = 'Science detector 1CD' / Temperature sensor name
HIERARCH ESO INS TEMP18 UNIT = 'K        '    / Temperature unit
HIERARCH ESO INS TEMP18 VAL  =        72.020 / Temperature [K]
HIERARCH ESO INS TEMP19 ID   = 'Dt2BA    '    / Temperature sensor type
HIERARCH ESO INS TEMP19 NAME = 'Science detector 2BA' / Temperature sensor name
HIERARCH ESO INS TEMP19 UNIT = 'K        '    / Temperature unit
HIERARCH ESO INS TEMP19 VAL  =         72.020 / Temperature [K]
HIERARCH ESO INS TEMP2 ID    = 'Win      '    / Temperature sensor type
HIERARCH ESO INS TEMP2 NAME  = 'Cryostat window cell' / Temperature sensor name
HIERARCH ESO INS TEMP2 UNIT  = 'K        '    / Temperature unit
HIERARCH ESO INS TEMP2 VAL   =       298.940 / Temperature [K]
HIERARCH ESO INS TEMP20 ID   = 'Dt2DC    '    / Temperature sensor type
HIERARCH ESO INS TEMP20 NAME = 'Science detector 2DC' / Temperature sensor name
HIERARCH ESO INS TEMP20 UNIT = 'K        '    / Temperature unit
HIERARCH ESO INS TEMP20 VAL  =        72.020 / Temperature [K]
HIERARCH ESO INS TEMP21 ID   = 'Dt3AB    '    / Temperature sensor type
HIERARCH ESO INS TEMP21 NAME = 'Science detector 3AB' / Temperature sensor name
HIERARCH ESO INS TEMP21 UNIT = 'K        '    / Temperature unit
HIERARCH ESO INS TEMP21 VAL  =        72.020 / Temperature [K]
HIERARCH ESO INS TEMP22 ID   = 'Dt3CD    '    / Temperature sensor type
HIERARCH ESO INS TEMP22 NAME = 'Science detector 3CD' / Temperature sensor name
HIERARCH ESO INS TEMP22 UNIT = 'K        '    / Temperature unit
```

```
HIERARCH ESO INS TEMP22 VAL  =       72.020 / Temperature [K]
HIERARCH ESO INS TEMP23 ID   = 'Dt4BA   '   / Temperature sensor type
HIERARCH ESO INS TEMP23 NAME = 'Science detector 4BA' / Temperature sensor name
HIERARCH ESO INS TEMP23 UNIT = 'K       '   / Temperature unit
HIERARCH ESO INS TEMP23 VAL  =       72.020 / Temperature [K]
HIERARCH ESO INS TEMP24 ID   = 'Dt4DC   '   / Temperature sensor type
HIERARCH ESO INS TEMP24 NAME = 'Science detector 4DC' / Temperature sensor name
HIERARCH ESO INS TEMP24 UNIT = 'K       '   / Temperature unit
HIERARCH ESO INS TEMP24 VAL  =       72.020 / Temperature [K]
HIERARCH ESO INS TEMP25 ID   = 'FPA     '   / Temperature sensor type
HIERARCH ESO INS TEMP25 NAME = 'FPA thermal plate' / Temperature sensor name
HIERARCH ESO INS TEMP25 UNIT = 'K       '   / Temperature unit
HIERARCH ESO INS TEMP25 VAL  =       80.780 / Temperature [K]
HIERARCH ESO INS TEMP26 ID   = 'WFSpl   '   / Temperature sensor type
HIERARCH ESO INS TEMP26 NAME = 'WFS plate'  / Temperature sensor name
HIERARCH ESO INS TEMP26 UNIT = 'K       '   / Temperature unit
HIERARCH ESO INS TEMP26 VAL  =      110.930 / Temperature [K]
HIERARCH ESO INS TEMP3 ID    = 'Tube    '   / Temperature sensor type
HIERARCH ESO INS TEMP3 NAME  = 'Cryostat tube' / Temperature sensor name
HIERARCH ESO INS TEMP3 UNIT  = 'K       '   / Temperature unit
HIERARCH ESO INS TEMP3 VAL   =      299.430 / Temperature [K]
HIERARCH ESO INS TEMP4 ID    = 'OBtop   '   / Temperature sensor type
HIERARCH ESO INS TEMP4 NAME  = 'Optical Bench Top' / Temperature sensor name
HIERARCH ESO INS TEMP4 UNIT  = 'K       '   / Temperature unit
HIERARCH ESO INS TEMP4 VAL   =      121.030 / Temperature [K]
HIERARCH ESO INS TEMP5 ID    = 'Baff    '   / Temperature sensor type
HIERARCH ESO INS TEMP5 NAME  = 'Baffle  '   / Temperature sensor name
HIERARCH ESO INS TEMP5 UNIT  = 'K       '   / Temperature unit
HIERARCH ESO INS TEMP5 VAL   =      128.850 / Temperature [K]
HIERARCH ESO INS TEMP6 ID    = 'Lens    '   / Temperature sensor type
HIERARCH ESO INS TEMP6 NAME  = 'Lens barrel' / Temperature sensor name
HIERARCH ESO INS TEMP6 UNIT  = 'K       '   / Temperature unit
HIERARCH ESO INS TEMP6 VAL   =      119.380 / Temperature [K]
HIERARCH ESO INS TEMP7 ID    = 'FwShd   '   / Temperature sensor type
HIERARCH ESO INS TEMP7 NAME  = 'Filter wheel shield' / Temperature sensor name
HIERARCH ESO INS TEMP7 UNIT  = 'K       '   / Temperature unit
HIERARCH ESO INS TEMP7 VAL   =      101.070 / Temperature [K]
HIERARCH ESO INS TEMP8 ID    = 'FwHub   '   / Temperature sensor type
HIERARCH ESO INS TEMP8 NAME  = 'Filter wheel hub' / Temperature sensor name
HIERARCH ESO INS TEMP8 UNIT  = 'K       '   / Temperature unit
HIERARCH ESO INS TEMP8 VAL   =      106.170 / Temperature [K]
HIERARCH ESO INS THERMAL AMB MEAN=   297.76 / Ambient temperature [K]
HIERARCH ESO INS THERMAL CLD MEAN=    30.00 / Cold head temperature [K]
HIERARCH ESO INS THERMAL DET MEAN=    72.02 / Detector mean temperature [K]
HIERARCH ESO INS THERMAL DET TARGET=  72.00 / Detector target temperature [K]
HIERARCH ESO INS THERMAL ENABLE=          T / If T, thermal control enabled
HIERARCH ESO INS THERMAL FPA MEAN=    80.43 / Focal plane array temperature [K]
HIERARCH ESO INS THERMAL TUB MEAN=   299.10 / Tube temperature [K]
HIERARCH ESO INS THERMAL WIN MEAN=   298.42 / Window temperature [K]
HIERARCH ESO INS VAC1 OK     =            T / If T, controller is operational
HIERARCH ESO INS VAC1 SWSIM  =            T / If T, vacuum sensor simulated
HIERARCH ESO OBS DID         = 'ESO-VLT-DIC.OBS-1.11' / OBS Dictionary
HIERARCH ESO OBS GRP         = '0       '   / linked blocks
HIERARCH ESO OBS ID          =           -1 / Observation block ID
HIERARCH ESO OBS NAME        = 'Maintenance' / OB name
HIERARCH ESO OBS OBSERVER    = 'S.Beard-S.McLay' / Observer Name
HIERARCH ESO OBS PI-COI ID   =            0 / ESO internal PI-COI ID
HIERARCH ESO OBS PI-COI NAME = 'G.Dalton-W.Sutherland' / PI-COI name
HIERARCH ESO OBS PROG ID     = 'Maintenance' / ESO program identification
HIERARCH ESO OBS START       = '2007-01-16T17:51:35' / OB start time
HIERARCH ESO OBS TARG NAME   = 'Bit of the sky' / OB target name
```

```
HIERARCH ESO OBS TPLNO       =              1 / Template number within OB
HIERARCH ESO OCS DET1 IMGNAME= 'VIRCAM_IMG_OBS' / Data File Name.
HIERARCH ESO OCS EXPNO       =              1 / Exposure number of dwell
HIERARCH ESO OCS NEXP        =              1 / Number of exposures per dwell
HIERARCH ESO OCS RECIPE      = 'OFFSETSKY'   / Data reduction recipe to be used
HIERARCH ESO OCS REQTIME     =         10.000 / Requested integration time [s]
HIERARCH ESO OCS SADT AOSA1 ID= 'S2301200159' / AO star A ID
HIERARCH ESO OCS SADT AOSA2 ID= 'S2301200157' / AO star A ID
HIERARCH ESO OCS SADT AOSA3 ID= 'S230120056672' / AO star A ID
HIERARCH ESO OCS SADT AOSA4 ID= 'S230120056327' / AO star A ID
HIERARCH ESO OCS SADT AOSA5 ID= 'S230120055492' / AO star A ID
HIERARCH ESO OCS SADT AOSB1 ID= 'S230320034067' / AO star B ID
HIERARCH ESO OCS SADT AOSB2 ID= 'S230320033628' / AO star B ID
HIERARCH ESO OCS SADT AOSB3 ID= 'S230320031521' / AO star B ID
HIERARCH ESO OCS SADT AOSB4 ID= 'S230320030062' / AO star B ID
HIERARCH ESO OCS SADT AOSB5 ID= 'S230320031153' / AO star B ID
HIERARCH ESO OCS SADT CAT ID = 'GSC-2 at ESO' / Guide star catalogue used
HIERARCH ESO OCS SADT GS1 ID = 'S2303200224' / Guide star ID
HIERARCH ESO OCS SADT GS2 ID = 'S2303200220' / Guide star ID
HIERARCH ESO OCS SADT GS3 ID = 'S2303200227' / Guide star ID
HIERARCH ESO OCS SADT GS4 ID = 'S230320031516' / Guide star ID
HIERARCH ESO OCS SADT GS5 ID = 'S230120052235' / Guide star ID
HIERARCH ESO OCS SADT IP ID  = 'SADT v1.02, Obsolete VIRCAM IP' / Creator sw
HIERARCH ESO OCS TARG ALPHAOBJ=  211940.138 / RA of target object [HHMMSS.TTT]
HIERARCH ESO OCS TARG DELTAOBJ= -885029.437 / Dec of target object [DDMMSS.TTT]
HIERARCH ESO OCS TARG X      =           0.00 / Pointing origin X coord [mm]
HIERARCH ESO OCS TARG Y      =           0.00 / Pointing origin Y coord [mm]
HIERARCH ESO TEL ABSROT START=        0.000 / Abs rotator angle at start
HIERARCH ESO TEL AIRM END    =        0.000 / Airmass at end
HIERARCH ESO TEL AIRM START  =        0.000 / Airmass at start
HIERARCH ESO TEL ALT         =       25.943 / Alt angle at start (deg)
HIERARCH ESO TEL AMBI FWHM END=        0.00 / Observatory Seeing queried from A
HIERARCH ESO TEL AMBI FWHM START=      0.00 / Observatory Seeing queried from A
HIERARCH ESO TEL AMBI PRES END=      750.00 / Observatory ambient air pressure
HIERARCH ESO TEL AMBI PRES START=    750.00 / Observatory ambient air pressure
HIERARCH ESO TEL AMBI RHUM   =            12. / Observatory ambient relative hum
HIERARCH ESO TEL AMBI TAU0   =     0.000000 / Average coherence time
HIERARCH ESO TEL AMBI TEMP   =        10.00 / Observatory ambient temperature q
HIERARCH ESO TEL AMBI WINDDIR=             0. / Observatory ambient wind directi
HIERARCH ESO TEL AMBI WINDSP =        10.00 / Observatory ambient wind speed qu
HIERARCH ESO TEL AO ALT      =     0.000000 / Altitude of last closed loop aO
HIERARCH ESO TEL AO DATE     = '        '   / Last closed loop aO
HIERARCH ESO TEL AO M1 DATE  = '2007-01-16T17:52:41' / Last M1 update
HIERARCH ESO TEL AO M2 DATE  = '2007-01-16T17:52:40' / Last M2 update
HIERARCH ESO TEL AO MODES    =              0 / Which aO modes corrected closed l
HIERARCH ESO TEL AZ          =      359.803 / Az angle at start (deg) S=0,W=90
HIERARCH ESO TEL DATE        = 'not set '   / TCS installation date
HIERARCH ESO TEL DID         = 'ESO-VLT-DIC.TCS-01.00' / Data dictionary for TE
HIERARCH ESO TEL DID1        = 'ESO-VLT-DIC.VTCS-0.2' / Additional data dict. f
HIERARCH ESO TEL DOME STATUS = 'FULLY-OPEN' / Dome status
HIERARCH ESO TEL ECS FLATFIELD=             0 / Flat field level
HIERARCH ESO TEL ECS MOONSCR =         0.00 / Moon screen position
HIERARCH ESO TEL ECS VENT1   =         0.00 / State of vent i
HIERARCH ESO TEL ECS VENT2   =         0.00 / State of vent i
HIERARCH ESO TEL ECS VENT3   =         0.00 / State of vent i
HIERARCH ESO TEL ECS WINDSCR =         0.00 / Wind screen position
HIERARCH ESO TEL FOCU ID     = 'CA      '   / Telescope focus station ID
HIERARCH ESO TEL FOCU VALUE  =        0.000 / M2 setting (mm)
HIERARCH ESO TEL GEOELEV     =         2530. / Elevation above sea level (m)
HIERARCH ESO TEL GEOLAT      =     -24.6157 / Tel geo latitude (+=North) (deg)
HIERARCH ESO TEL GEOLON      =     -70.3976 / Tel geo longitude (+=East) (deg)
```

```
HIERARCH ESO TEL GUID STATUS = 'OFF     '   / Status of autoguider
HIERARCH ESO TEL ID          = 'v 0.59  '   / TCS version number
HIERARCH ESO TEL M2 ACENTRE  =         0.00 / M2 centring alpha
HIERARCH ESO TEL M2 ATILT    =         0.00 / M2 tilt alpha
HIERARCH ESO TEL M2 BCENTRE  =         0.00 / M2 centring beta
HIERARCH ESO TEL M2 BTILT    =         0.00 / M2 tilt beta
HIERARCH ESO TEL M2 Z        =      0.00000 / Focussing position of M2 in Z coo
HIERARCH ESO TEL MOON DEC    =      0.00000 / 00:00:00.0 DEC (J2000) (deg)
HIERARCH ESO TEL MOON RA     =     0.000000 / 00:00:00.0 RA (J2000) (deg)
HIERARCH ESO TEL OPER        = 'Operator name not set' / Telescope Operator
HIERARCH ESO TEL PARANG END  =        0.000 / Parallactic angle at end (deg)
HIERARCH ESO TEL PARANG START=        0.000 / Parallactic angle at start (deg)
HIERARCH ESO TEL POSANG      =       42.000 / Rot position angle at start
HIERARCH ESO TEL TARG ALPHA  =   212053.060 / Alpha coordinate for the target
HIERARCH ESO TEL TARG COORDTYPE= 'M       ' / Coordinate type (M=mean A=apparen
HIERARCH ESO TEL TARG DELTA  =  -884239.893 / Delta coordinate for the target
HIERARCH ESO TEL TARG EPOCH  =     2000.000 / Epoch
HIERARCH ESO TEL TARG EPOCHSYSTEM= 'J       ' / Epoch system (default J=Julian)
HIERARCH ESO TEL TARG EQUINOX=     2000.000 / Equinox
HIERARCH ESO TEL TARG PARALLAX=       0.000 / Parallax
HIERARCH ESO TEL TARG PMA    =     0.000000 / Proper Motion Alpha
HIERARCH ESO TEL TARG PMD    =     0.000000 / Proper motion Delta
HIERARCH ESO TEL TARG RADVEL =        0.000 / Radial velocity
HIERARCH ESO TEL TH M1 TEMP  =         0.00 / M1 superficial temperature
HIERARCH ESO TEL TH STR TEMP =         0.00 / Telescope structure temperature
HIERARCH ESO TEL TRAK STATUS = 'NORMAL  '   / Tracking status
HIERARCH ESO TPL DID         = 'ESO-VLT-DIC.TPL-1.9' / Data dictionary for TPL
HIERARCH ESO TPL EXPNO       =            2 / Exposure number within template
HIERARCH ESO TPL ID          = 'VIRCAM_img_obs_tile' / Template signature ID
HIERARCH ESO TPL NAME        = 'VIRCAM tile observation' / Template name
HIERARCH ESO TPL NEXP        =           36 / Number of exposures within templa
HIERARCH ESO TPL PRESEQ      = 'VIRCAM_img_obs_tile.seq' / Sequencer script
HIERARCH ESO TPL START       = '2007-01-16T17:51:35' / TPL start time
HIERARCH ESO TPL VERSION     = '$Revision: 0.55 $' / Version of the template
JITTER_I=                    2          / Sequence number of jitter
JITTER_X=                0.000          / X offset in jitter pattern [arcsec]
JITTER_Y=                0.000          / Y offset in jitter pattern [arcsec]
JITTRNUM=                    1          / Value of 1st OBSNUM in jitter seq
JITTR_ID= 'Jitter3u'                    / Name of jitter pattern
NJITTER =                    3          / Number of jitter positions
NOFFSETS=                    6          / Number of offset positions
NUSTEP  =                    1          / Number of microstep positions
OBSNUM  =                    2          / Observation number
OFFSET_I=                    1          / Sequence number of offset
OFFSET_X=             -332.500          / X offset [arcsec]
OFFSET_Y=              332.500          / Y offset [arcsec]
OFFSTNUM=                    1          / First OBSNUM in offset sequence
OFFST_ID= 'Tile6z  '                    / Name of offset pattern
RECIPE  = 'OFFSETSKY'                   / Data reduction recipe to be used
REQTIME =               10.000          / Requested integration time [s]
USTEPNUM=                    2          / Value of 1st OBSNUM in ustep seq
USTEP_I =                    1          / Sequence number of ustep
USTEP_ID= 'Single  '                    / Name of ustep pattern
USTEP_X =                0.000          / X offset in ustep pattern [arcsec]
USTEP_Y =                0.000          / Y offset in ustep pattern [arcsec]
END
```

11.9.2 Example of FITS IMAGE extension header

```
XTENSION= 'IMAGE   '           / IMAGE extension
BITPIX  =                   32 / # of bits per pix value
```

```
NAXIS   =                    2 / # of axes in data array
NAXIS1  =                 1024 / # of pixels in axis1
NAXIS2  =                 1024 / # of pixels in axis2
PCOUNT  =                    0 / number of random group parameters
GCOUNT  =                    1 / number of random groups
EXTNAME = 'DET1.CHIP1'         / Extension name
EXTVER  = 1 / Extension version
ORIGIN  = 'ESO     ' / European Southern Observatory
DATE    = '2007-01-16T17:52:50.6323' / Date the file was written
EXPTIME =           10.0000000 / Integration time
MJD-OBS =       54116.74490925 / Obs start 2007-01-16T17:52:40.159
DATE-OBS= '2007-01-16T17:52:40.1590' / Observing date
CTYPE1  = 'RA---ZPN'           / Coordinate/projection type
CTYPE2  = 'DEC--ZPN'           / Coordinate/projection type
CRVAL1  =                   0. / Coordinate value at ref pixel
CRVAL2  =                   0. / Coordinate value at ref pixel
CRPIX1  =               5401.6 / Pixel coordinate at ref point
CRPIX2  =               6860.8 / Pixel coordinate at ref point
ORIGFILE= 'VIRCAM_IMG_OBS016_0002_DET01.fits' / Original File Name
CD1_1   = -6.3530233681516E-05 / WCS transform matrix element
CD1_2   = 7.05574725967148E-05 / WCS transform matrix element
CD2_1   = -7.05574725967148E-05 / WCS transform matrix element
CD2_2   =  6.3530233681516E-05 / WCS transform matrix element
HIERARCH ESO DET CHIP ID      = 'ESO-Virgo35' / Detector ID
HIERARCH ESO DET CHIP LIVE    = F / Detector live or broken
HIERARCH ESO DET CHIP NAME    = 'Virgo-Sim' / Detector name
HIERARCH ESO DET CHIP NO      = 1 / Unique Detector Number
HIERARCH ESO DET CHIP NX      = 1024 / Pixels in X
HIERARCH ESO DET CHIP NY      = 1024 / Pixels in Y
HIERARCH ESO DET CHIP PXSPACE= 1.850e-05 / Pixel-Pixel Spacing
HIERARCH ESO DET CHIP TYPE    = 'IR' / The Type of Det Chip
HIERARCH ESO DET CHIP VIGNETD = F / Detector chip vignetted?
HIERARCH ESO DET CHIP X       = 1 / Detector position x-axis
HIERARCH ESO DET CHIP Y       = 4 / Detector position y-axis
HIERARCH ESO DET CHOP FREQ    = 0 / Chopping Frequency
HIERARCH ESO DET CON OPMODE   = 'SIMULATION' / Operational Mode
HIERARCH ESO DET DID          = 'ESO-VLT-DIC.IRACE-1.38' / Dictionary Name and Re
HIERARCH ESO DET DIT          = 10.0000000 / Integration Time
HIERARCH ESO DET DITDELAY     = 0.000 / Pause Between DITs
HIERARCH ESO DET EXP NAME     = 'VIRCAM_IMG_OBS016_0002' / Exposure Name
HIERARCH ESO DET EXP NO       = 1461 / Exposure number
HIERARCH ESO DET EXP UTC      = '2007-01-16T17:52:50.6323' / File Creation Time
HIERARCH ESO DET FILE CUBE ST= F / Data Cube On/Off
HIERARCH ESO DET FRAM NO      = 1 / Frame number
HIERARCH ESO DET FRAM TYPE    = 'INT' / Frame type
HIERARCH ESO DET FRAM UTC     = '2007-01-16T17:52:50.4185' / Time Recv Frame
HIERARCH ESO DET IRACE ADC1 DELAY= 7 / ADC Delay Adjustment
HIERARCH ESO DET IRACE ADC1 ENABLE= 1 / Enable ADC Board (0/1)
HIERARCH ESO DET IRACE ADC1 FILTER1= 0 / ADC Filter1 Adjustment
HIERARCH ESO DET IRACE ADC1 FILTER2= 0 / ADC Filter2 Adjustment
HIERARCH ESO DET IRACE ADC1 HEADER= 1 / Header of ADC Board
HIERARCH ESO DET IRACE ADC1 NAME= 'VISTA-AQ-GRP' / Name for ADC Board
HIERARCH ESO DET IRACE ADC10 DELAY= 0 / ADC Delay Adjustment
HIERARCH ESO DET IRACE ADC10 ENABLE= 1 / Enable ADC Board (0/1)
HIERARCH ESO DET IRACE ADC10 FILTER1= 0 / ADC Filter1 Adjustment
HIERARCH ESO DET IRACE ADC10 FILTER2= 0 / ADC Filter2 Adjustment
HIERARCH ESO DET IRACE ADC10 HEADER= 1 / Header of ADC Board
HIERARCH ESO DET IRACE ADC10 NAME= 'VISTA-AQ-GRP' / Name for ADC Board
HIERARCH ESO DET IRACE ADC11 DELAY= 0 / ADC Delay Adjustment
HIERARCH ESO DET IRACE ADC11 ENABLE= 1 / Enable ADC Board (0/1)
HIERARCH ESO DET IRACE ADC11 FILTER1= 0 / ADC Filter1 Adjustment
```

```
HIERARCH ESO DET IRACE ADC11 FILTER2= 0 / ADC Filter2 Adjustment
HIERARCH ESO DET IRACE ADC11 HEADER= 1 / Header of ADC Board
HIERARCH ESO DET IRACE ADC11 NAME= 'VISTA-AQ-GRP' / Name for ADC Board
HIERARCH ESO DET IRACE ADC12 DELAY= 0 / ADC Delay Adjustment
HIERARCH ESO DET IRACE ADC12 ENABLE= 1 / Enable ADC Board (0/1)
HIERARCH ESO DET IRACE ADC12 FILTER1= 0 / ADC Filter1 Adjustment
HIERARCH ESO DET IRACE ADC12 FILTER2= 0 / ADC Filter2 Adjustment
HIERARCH ESO DET IRACE ADC12 HEADER= 1 / Header of ADC Board
HIERARCH ESO DET IRACE ADC12 NAME= 'VISTA-AQ-GRP' / Name for ADC Board
HIERARCH ESO DET IRACE ADC13 DELAY= 0 / ADC Delay Adjustment
HIERARCH ESO DET IRACE ADC13 ENABLE= 1 / Enable ADC Board (0/1)
HIERARCH ESO DET IRACE ADC13 FILTER1= 0 / ADC Filter1 Adjustment
HIERARCH ESO DET IRACE ADC13 FILTER2= 0 / ADC Filter2 Adjustment
HIERARCH ESO DET IRACE ADC13 HEADER= 1 / Header of ADC Board
HIERARCH ESO DET IRACE ADC13 NAME= 'VISTA-AQ-GRP' / Name for ADC Board
HIERARCH ESO DET IRACE ADC14 DELAY= 0 / ADC Delay Adjustment
HIERARCH ESO DET IRACE ADC14 ENABLE= 1 / Enable ADC Board (0/1)
HIERARCH ESO DET IRACE ADC14 FILTER1= 0 / ADC Filter1 Adjustment
HIERARCH ESO DET IRACE ADC14 FILTER2= 0 / ADC Filter2 Adjustment
HIERARCH ESO DET IRACE ADC14 HEADER= 1 / Header of ADC Board
HIERARCH ESO DET IRACE ADC14 NAME= 'VISTA-AQ-GRP' / Name for ADC Board
HIERARCH ESO DET IRACE ADC15 DELAY= 0 / ADC Delay Adjustment
HIERARCH ESO DET IRACE ADC15 ENABLE= 1 / Enable ADC Board (0/1)
HIERARCH ESO DET IRACE ADC15 FILTER1= 0 / ADC Filter1 Adjustment
HIERARCH ESO DET IRACE ADC15 FILTER2= 0 / ADC Filter2 Adjustment
HIERARCH ESO DET IRACE ADC15 HEADER= 1 / Header of ADC Board
HIERARCH ESO DET IRACE ADC15 NAME= 'VISTA-AQ-GRP' / Name for ADC Board
HIERARCH ESO DET IRACE ADC16 DELAY= 0 / ADC Delay Adjustment
HIERARCH ESO DET IRACE ADC16 ENABLE= 1 / Enable ADC Board (0/1)
HIERARCH ESO DET IRACE ADC16 FILTER1= 0 / ADC Filter1 Adjustment
HIERARCH ESO DET IRACE ADC16 FILTER2= 0 / ADC Filter2 Adjustment
HIERARCH ESO DET IRACE ADC16 HEADER= 1 / Header of ADC Board
HIERARCH ESO DET IRACE ADC16 NAME= 'VISTA-AQ-GRP' / Name for ADC Board
HIERARCH ESO DET IRACE ADC2 DELAY= 7 / ADC Delay Adjustment
HIERARCH ESO DET IRACE ADC2 ENABLE= 1 / Enable ADC Board (0/1)
HIERARCH ESO DET IRACE ADC2 FILTER1= 0 / ADC Filter1 Adjustment
HIERARCH ESO DET IRACE ADC2 FILTER2= 0 / ADC Filter2 Adjustment
HIERARCH ESO DET IRACE ADC2 HEADER= 1 / Header of ADC Board
HIERARCH ESO DET IRACE ADC2 NAME= 'VISTA-AQ-GRP' / Name for ADC Board
HIERARCH ESO DET IRACE ADC3 DELAY= 0 / ADC Delay Adjustment
HIERARCH ESO DET IRACE ADC3 ENABLE= 1 / Enable ADC Board (0/1)
HIERARCH ESO DET IRACE ADC3 FILTER1= 0 / ADC Filter1 Adjustment
HIERARCH ESO DET IRACE ADC3 FILTER2= 0 / ADC Filter2 Adjustment
HIERARCH ESO DET IRACE ADC3 HEADER= 1 / Header of ADC Board
HIERARCH ESO DET IRACE ADC3 NAME= 'VISTA-AQ-GRP' / Name for ADC Board
HIERARCH ESO DET IRACE ADC4 DELAY= 0 / ADC Delay Adjustment
HIERARCH ESO DET IRACE ADC4 ENABLE= 1 / Enable ADC Board (0/1)
HIERARCH ESO DET IRACE ADC4 FILTER1= 0 / ADC Filter1 Adjustment
HIERARCH ESO DET IRACE ADC4 FILTER2= 0 / ADC Filter2 Adjustment
HIERARCH ESO DET IRACE ADC4 HEADER= 1 / Header of ADC Board
HIERARCH ESO DET IRACE ADC4 NAME= 'VISTA-AQ-GRP' / Name for ADC Board
HIERARCH ESO DET IRACE ADC5 DELAY= 0 / ADC Delay Adjustment
HIERARCH ESO DET IRACE ADC5 ENABLE= 1 / Enable ADC Board (0/1)
HIERARCH ESO DET IRACE ADC5 FILTER1= 0 / ADC Filter1 Adjustment
HIERARCH ESO DET IRACE ADC5 FILTER2= 0 / ADC Filter2 Adjustment
HIERARCH ESO DET IRACE ADC5 HEADER= 1 / Header of ADC Board
HIERARCH ESO DET IRACE ADC5 NAME= 'VISTA-AQ-GRP' / Name for ADC Board
HIERARCH ESO DET IRACE ADC6 DELAY= 0 / ADC Delay Adjustment
HIERARCH ESO DET IRACE ADC6 ENABLE= 1 / Enable ADC Board (0/1)
HIERARCH ESO DET IRACE ADC6 FILTER1= 0 / ADC Filter1 Adjustment
HIERARCH ESO DET IRACE ADC6 FILTER2= 0 / ADC Filter2 Adjustment
```

```
HIERARCH ESO DET IRACE ADC6 HEADER= 1 / Header of ADC Board
HIERARCH ESO DET IRACE ADC6 NAME= 'VISTA-AQ-GRP' / Name for ADC Board
HIERARCH ESO DET IRACE ADC7 DELAY= 0 / ADC Delay Adjustment
HIERARCH ESO DET IRACE ADC7 ENABLE= 1 / Enable ADC Board (0/1)
HIERARCH ESO DET IRACE ADC7 FILTER1= 0 / ADC Filter1 Adjustment
HIERARCH ESO DET IRACE ADC7 FILTER2= 0 / ADC Filter2 Adjustment
HIERARCH ESO DET IRACE ADC7 HEADER= 1 / Header of ADC Board
HIERARCH ESO DET IRACE ADC7 NAME= 'VISTA-AQ-GRP' / Name for ADC Board
HIERARCH ESO DET IRACE ADC8 DELAY= 0 / ADC Delay Adjustment
HIERARCH ESO DET IRACE ADC8 ENABLE= 1 / Enable ADC Board (0/1)
HIERARCH ESO DET IRACE ADC8 FILTER1= 0 / ADC Filter1 Adjustment
HIERARCH ESO DET IRACE ADC8 FILTER2= 0 / ADC Filter2 Adjustment
HIERARCH ESO DET IRACE ADC8 HEADER= 1 / Header of ADC Board
HIERARCH ESO DET IRACE ADC8 NAME= 'VISTA-AQ-GRP' / Name for ADC Board
HIERARCH ESO DET IRACE ADC9 DELAY= 0 / ADC Delay Adjustment
HIERARCH ESO DET IRACE ADC9 ENABLE= 1 / Enable ADC Board (0/1)
HIERARCH ESO DET IRACE ADC9 FILTER1= 0 / ADC Filter1 Adjustment
HIERARCH ESO DET IRACE ADC9 FILTER2= 0 / ADC Filter2 Adjustment
HIERARCH ESO DET IRACE ADC9 HEADER= 1 / Header of ADC Board
HIERARCH ESO DET IRACE ADC9 NAME= 'VISTA-AQ-GRP' / Name for ADC Board
HIERARCH ESO DET IRACE SEQCONT= F / Sequencer Continuous Mode
HIERARCH ESO DET MINDIT      = 0.4006000 / Minimum DIT
HIERARCH ESO DET MODE NAME   = '' / DCS Detector Mode
HIERARCH ESO DET NCORRS      = 2 / Read-Out Mode
HIERARCH ESO DET NCORRS NAME = 'Double' / Read-Out Mode Name
HIERARCH ESO DET NDIT        = 1 / # of Sub-Integrations
HIERARCH ESO DET NDITSKIP    = 0 / DITs skipped at 1st.INT
HIERARCH ESO DET RSPEED      = 2 / Read-Speed Factor
HIERARCH ESO DET RSPEEDADD   = 0 / Read-Speed Add
HIERARCH ESO DET VOLT1 CLKHI1= 5.0000 / Set Value High-Clock
HIERARCH ESO DET VOLT1 CLKHI10= 5.0000 / Set Value High-Clock
HIERARCH ESO DET VOLT1 CLKHI11= 5.0000 / Set Value High-Clock
HIERARCH ESO DET VOLT1 CLKHI12= 5.0000 / Set Value High-Clock
HIERARCH ESO DET VOLT1 CLKHI13= 0.0000 / Set Value High-Clock
HIERARCH ESO DET VOLT1 CLKHI14= 0.0000 / Set Value High-Clock
HIERARCH ESO DET VOLT1 CLKHI15= 0.0000 / Set Value High-Clock
HIERARCH ESO DET VOLT1 CLKHI16= 0.0000 / Set Value High-Clock
HIERARCH ESO DET VOLT1 CLKHI2= 5.0000 / Set Value High-Clock
HIERARCH ESO DET VOLT1 CLKHI3= 5.0000 / Set Value High-Clock
HIERARCH ESO DET VOLT1 CLKHI4= 5.0000 / Set Value High-Clock
HIERARCH ESO DET VOLT1 CLKHI5= 5.0000 / Set Value High-Clock
HIERARCH ESO DET VOLT1 CLKHI6= 5.0000 / Set Value High-Clock
HIERARCH ESO DET VOLT1 CLKHI7= 5.0000 / Set Value High-Clock
HIERARCH ESO DET VOLT1 CLKHI8= 5.0000 / Set Value High-Clock
HIERARCH ESO DET VOLT1 CLKHI9= 5.0000 / Set Value High-Clock
HIERARCH ESO DET VOLT1 CLKHINM1= 'clk1Hi LSYNC' / Name of High-Clock
HIERARCH ESO DET VOLT1 CLKHINM10= 'clk10Hi READ' / Name of High-Clock
HIERARCH ESO DET VOLT1 CLKHINM11= 'clk11Hi VDD' / Name of High-Clock
HIERARCH ESO DET VOLT1 CLKHINM12= 'clock12Hi LRST' / Name of High-Clock
HIERARCH ESO DET VOLT1 CLKHINM13= 'clock13Hi' / Name of High-Clock
HIERARCH ESO DET VOLT1 CLKHINM14= 'clock14Hi' / Name of High-Clock
HIERARCH ESO DET VOLT1 CLKHINM15= 'clock15Hi' / Name of High-Clock
HIERARCH ESO DET VOLT1 CLKHINM16= 'clock16Hi' / Name of High-Clock
HIERARCH ESO DET VOLT1 CLKHINM2= 'clk2Hi CLK1' / Name of High-Clock
HIERARCH ESO DET VOLT1 CLKHINM3= 'clk3Hi CLKB1' / Name of High-Clock
HIERARCH ESO DET VOLT1 CLKHINM4= 'clk4Hi CLK2' / Name of High-Clock
HIERARCH ESO DET VOLT1 CLKHINM5= 'clk5Hi CLKB2' / Name of High-Clock
HIERARCH ESO DET VOLT1 CLKHINM6= 'clk6Hi FSYNC' / Name of High-Clock
HIERARCH ESO DET VOLT1 CLKHINM7= 'clk7Hi VCLK' / Name of High-Clock
HIERARCH ESO DET VOLT1 CLKHINM8= 'clk8Hi RESET' / Name of High-Clock
HIERARCH ESO DET VOLT1 CLKHINM9= 'clk9Hi RESETEN' / Name of High-Clock
```

```
HIERARCH ESO DET VOLT1 CLKHIT1= 5.0000 / Tel Value High-Clock
HIERARCH ESO DET VOLT1 CLKHIT10= 5.0000 / Tel Value High-Clock
HIERARCH ESO DET VOLT1 CLKHIT11= 5.0000 / Tel Value High-Clock
HIERARCH ESO DET VOLT1 CLKHIT12= 5.0000 / Tel Value High-Clock
HIERARCH ESO DET VOLT1 CLKHIT13= 0.0000 / Tel Value High-Clock
HIERARCH ESO DET VOLT1 CLKHIT14= 0.0000 / Tel Value High-Clock
HIERARCH ESO DET VOLT1 CLKHIT15= 0.0000 / Tel Value High-Clock
HIERARCH ESO DET VOLT1 CLKHIT16= 0.0000 / Tel Value High-Clock
HIERARCH ESO DET VOLT1 CLKHIT2= 5.0000 / Tel Value High-Clock
HIERARCH ESO DET VOLT1 CLKHIT3= 5.0000 / Tel Value High-Clock
HIERARCH ESO DET VOLT1 CLKHIT4= 5.0000 / Tel Value High-Clock
HIERARCH ESO DET VOLT1 CLKHIT5= 5.0000 / Tel Value High-Clock
HIERARCH ESO DET VOLT1 CLKHIT6= 5.0000 / Tel Value High-Clock
HIERARCH ESO DET VOLT1 CLKHIT7= 5.0000 / Tel Value High-Clock
HIERARCH ESO DET VOLT1 CLKHIT8= 5.0000 / Tel Value High-Clock
HIERARCH ESO DET VOLT1 CLKHIT9= 5.0000 / Tel Value High-Clock
HIERARCH ESO DET VOLT1 CLKLO1= 0.0000 / Set value Low-Clock
HIERARCH ESO DET VOLT1 CLKLO10= 0.0000 / Set value Low-Clock
HIERARCH ESO DET VOLT1 CLKLO11= 5.0000 / Set value Low-Clock
HIERARCH ESO DET VOLT1 CLKLO12= 5.0000 / Set value Low-Clock
HIERARCH ESO DET VOLT1 CLKLO13= 0.0000 / Set value Low-Clock
HIERARCH ESO DET VOLT1 CLKLO14= 0.0000 / Set value Low-Clock
HIERARCH ESO DET VOLT1 CLKLO15= 0.0000 / Set value Low-Clock
HIERARCH ESO DET VOLT1 CLKLO16= 0.0000 / Set value Low-Clock
HIERARCH ESO DET VOLT1 CLKLO2= 0.0000 / Set value Low-Clock
HIERARCH ESO DET VOLT1 CLKLO3= 0.0000 / Set value Low-Clock
HIERARCH ESO DET VOLT1 CLKLO4= 0.0000 / Set value Low-Clock
HIERARCH ESO DET VOLT1 CLKLO5= 0.0000 / Set value Low-Clock
HIERARCH ESO DET VOLT1 CLKLO6= 0.0000 / Set value Low-Clock
HIERARCH ESO DET VOLT1 CLKLO7= 0.0000 / Set value Low-Clock
HIERARCH ESO DET VOLT1 CLKLO8= 0.0000 / Set value Low-Clock
HIERARCH ESO DET VOLT1 CLKLO9= 5.0000 / Set value Low-Clock
HIERARCH ESO DET VOLT1 CLKLONM1= 'clk1Lo LSYNC' / Name of Low-Clock
HIERARCH ESO DET VOLT1 CLKLONM10= 'clk10Lo READ' / Name of Low-Clock
HIERARCH ESO DET VOLT1 CLKLONM11= 'clk11Lo VDD' / Name of Low-Clock
HIERARCH ESO DET VOLT1 CLKLONM12= 'clock12Lo LRST' / Name of Low-Clock
HIERARCH ESO DET VOLT1 CLKLONM13= 'clock13Lo' / Name of Low-Clock
HIERARCH ESO DET VOLT1 CLKLONM14= 'clock14Lo' / Name of Low-Clock
HIERARCH ESO DET VOLT1 CLKLONM15= 'clock15Lo' / Name of Low-Clock
HIERARCH ESO DET VOLT1 CLKLONM16= 'clock16Lo' / Name of Low-Clock
HIERARCH ESO DET VOLT1 CLKLONM2= 'clk2Lo CLK1' / Name of Low-Clock
HIERARCH ESO DET VOLT1 CLKLONM3= 'clk3Lo CLKB1' / Name of Low-Clock
HIERARCH ESO DET VOLT1 CLKLONM4= 'clk4Lo CLK2' / Name of Low-Clock
HIERARCH ESO DET VOLT1 CLKLONM5= 'clk5Lo CLKB2' / Name of Low-Clock
HIERARCH ESO DET VOLT1 CLKLONM6= 'clk6Lo FSYNC' / Name of Low-Clock
HIERARCH ESO DET VOLT1 CLKLONM7= 'clk7Lo VCLK' / Name of Low-Clock
HIERARCH ESO DET VOLT1 CLKLONM8= 'clk8Lo RESET' / Name of Low-Clock
HIERARCH ESO DET VOLT1 CLKLONM9= 'clk9Lo RESETEN' / Name of Low-Clock
HIERARCH ESO DET VOLT1 CLKLOT1= 0.0000 / Tel Value Low-Clock
HIERARCH ESO DET VOLT1 CLKLOT10= 0.0000 / Tel Value Low-Clock
HIERARCH ESO DET VOLT1 CLKLOT11= 5.0000 / Tel Value Low-Clock
HIERARCH ESO DET VOLT1 CLKLOT12= 5.0000 / Tel Value Low-Clock
HIERARCH ESO DET VOLT1 CLKLOT13= 0.0000 / Tel Value Low-Clock
HIERARCH ESO DET VOLT1 CLKLOT14= 0.0000 / Tel Value Low-Clock
HIERARCH ESO DET VOLT1 CLKLOT15= 0.0000 / Tel Value Low-Clock
HIERARCH ESO DET VOLT1 CLKLOT16= 0.0000 / Tel Value Low-Clock
HIERARCH ESO DET VOLT1 CLKLOT2= 0.0000 / Tel Value Low-Clock
HIERARCH ESO DET VOLT1 CLKLOT3= 0.0000 / Tel Value Low-Clock
HIERARCH ESO DET VOLT1 CLKLOT4= 0.0000 / Tel Value Low-Clock
HIERARCH ESO DET VOLT1 CLKLOT5= 0.0000 / Tel Value Low-Clock
HIERARCH ESO DET VOLT1 CLKLOT6= 0.0000 / Tel Value Low-Clock
```

```
HIERARCH ESO DET VOLT1 CLKLOT7= 0.0000 / Tel Value Low-Clock
HIERARCH ESO DET VOLT1 CLKLOT8= 0.0000 / Tel Value Low-Clock
HIERARCH ESO DET VOLT1 CLKLOT9= 5.0000 / Tel Value Low-Clock
HIERARCH ESO DET VOLT1 DC1    = 4.0000 / Set value DC-Voltage
HIERARCH ESO DET VOLT1 DC10   = 5.0000 / Set value DC-Voltage
HIERARCH ESO DET VOLT1 DC11   = 5.9560 / Set value DC-Voltage
HIERARCH ESO DET VOLT1 DC12   = 0.0000 / Set value DC-Voltage
HIERARCH ESO DET VOLT1 DC13   = 0.0000 / Set value DC-Voltage
HIERARCH ESO DET VOLT1 DC14   = 0.0000 / Set value DC-Voltage
HIERARCH ESO DET VOLT1 DC15   = 0.0000 / Set value DC-Voltage
HIERARCH ESO DET VOLT1 DC2    = 5.4000 / Set value DC-Voltage
HIERARCH ESO DET VOLT1 DC3    = 5.0000 / Set value DC-Voltage
HIERARCH ESO DET VOLT1 DC4    = 5.0000 / Set value DC-Voltage
HIERARCH ESO DET VOLT1 DC5    = 5.0000 / Set value DC-Voltage
HIERARCH ESO DET VOLT1 DC6    = 5.0000 / Set value DC-Voltage
HIERARCH ESO DET VOLT1 DC7    = 5.0000 / Set value DC-Voltage
HIERARCH ESO DET VOLT1 DC8    = 5.0000 / Set value DC-Voltage
HIERARCH ESO DET VOLT1 DC9    = 5.0000 / Set value DC-Voltage
HIERARCH ESO DET VOLT1 DCNM1 = 'DC1 VRESET1-2-3-4' / Name of DC-voltage
HIERARCH ESO DET VOLT1 DCNM10= 'DC10 VLOAD4' / Name of DC-voltage
HIERARCH ESO DET VOLT1 DCNM11= 'DC11 Reference RevB' / Name of DC-voltage
HIERARCH ESO DET VOLT1 DCNM12= 'DC12' / Name of DC-voltage
HIERARCH ESO DET VOLT1 DCNM13= 'DC13' / Name of DC-voltage
HIERARCH ESO DET VOLT1 DCNM14= 'DC14' / Name of DC-voltage
HIERARCH ESO DET VOLT1 DCNM15= 'DC15' / Name of DC-voltage
HIERARCH ESO DET VOLT1 DCNM2 = 'DC2 Reference' / Name of DC-voltage
HIERARCH ESO DET VOLT1 DCNM3 = 'DC3 BIASGATE' / Name of DC-voltage
HIERARCH ESO DET VOLT1 DCNM4 = 'DC4 BIASPOWER' / Name of DC-voltage
HIERARCH ESO DET VOLT1 DCNM5 = 'DC5 VDDA' / Name of DC-voltage
HIERARCH ESO DET VOLT1 DCNM6 = 'DC6 DRAIN' / Name of DC-voltage
HIERARCH ESO DET VOLT1 DCNM7 = 'DC7 VLOAD1' / Name of DC-voltage
HIERARCH ESO DET VOLT1 DCNM8 = 'DC8 VLOAD2' / Name of DC-voltage
HIERARCH ESO DET VOLT1 DCNM9 = 'DC9 VLOAD3' / Name of DC-voltage
HIERARCH ESO DET VOLT1 DCTA1 = 3.9990 / Tel Value 1 for DC
HIERARCH ESO DET VOLT1 DCTA10= 5.0000 / Tel Value 1 for DC
HIERARCH ESO DET VOLT1 DCTA11= 5.9521 / Tel Value 1 for DC
HIERARCH ESO DET VOLT1 DCTA12= 0.0000 / Tel Value 1 for DC
HIERARCH ESO DET VOLT1 DCTA13= 0.0000 / Tel Value 1 for DC
HIERARCH ESO DET VOLT1 DCTA14= 0.0000 / Tel Value 1 for DC
HIERARCH ESO DET VOLT1 DCTA15= 0.0000 / Tel Value 1 for DC
HIERARCH ESO DET VOLT1 DCTA2 = 5.3955 / Tel Value 1 for DC
HIERARCH ESO DET VOLT1 DCTA3 = 5.0000 / Tel Value 1 for DC
HIERARCH ESO DET VOLT1 DCTA4 = 5.0000 / Tel Value 1 for DC
HIERARCH ESO DET VOLT1 DCTA5 = 5.0000 / Tel Value 1 for DC
HIERARCH ESO DET VOLT1 DCTA6 = 5.0000 / Tel Value 1 for DC
HIERARCH ESO DET VOLT1 DCTA7 = 5.0000 / Tel Value 1 for DC
HIERARCH ESO DET VOLT1 DCTA8 = 5.0000 / Tel Value 1 for DC
HIERARCH ESO DET VOLT1 DCTA9 = 5.0000 / Tel Value 1 for DC
HIERARCH ESO DET VOLT1 DCTB1 = 3.9990 / Tel Value 2 for DC
HIERARCH ESO DET VOLT1 DCTB10= 5.0000 / Tel Value 2 for DC
HIERARCH ESO DET VOLT1 DCTB11= 5.9521 / Tel Value 2 for DC
HIERARCH ESO DET VOLT1 DCTB12= 0.0000 / Tel Value 2 for DC
HIERARCH ESO DET VOLT1 DCTB13= 0.0000 / Tel Value 2 for DC
HIERARCH ESO DET VOLT1 DCTB14= 0.0000 / Tel Value 2 for DC
HIERARCH ESO DET VOLT1 DCTB15= 0.0000 / Tel Value 2 for DC
HIERARCH ESO DET VOLT1 DCTB2 = 5.3955 / Tel Value 2 for DC
HIERARCH ESO DET VOLT1 DCTB3 = 5.0000 / Tel Value 2 for DC
HIERARCH ESO DET VOLT1 DCTB4 = 5.0000 / Tel Value 2 for DC
HIERARCH ESO DET VOLT1 DCTB5 = 5.0000 / Tel Value 2 for DC
HIERARCH ESO DET VOLT1 DCTB6 = 5.0000 / Tel Value 2 for DC
HIERARCH ESO DET VOLT1 DCTB7 = 5.0000 / Tel Value 2 for DC
```

```
HIERARCH ESO DET VOLT1 DCTB8 = 5.0000 / Tel Value 2 for DC
HIERARCH ESO DET VOLT1 DCTB9 = 5.0000 / Tel Value 2 for DC
HIERARCH ESO DET WIN NX      = 1024 / # of Pixels in X
HIERARCH ESO DET WIN NY      = 1024 / # of Pixels in Y
HIERARCH ESO DET WIN STARTX  = 1 / Lower left X ref
HIERARCH ESO DET WIN STARTY  = 1 / Lower left Y ref
HIERARCH ESO DET WIN TYPE    = 0 / Win-Type: 0=SW/1=HW
INHERIT = T / Extension inherits primary header
PV2_1    =                    1. / WCS parameter value term
PV2_2    =                    0. / WCS parameter value term
PV2_3    =                   42. / WCS parameter value term
PV2_4    =                    0. / WCS parameter value term
PV2_5    =                    0. / WCS parameter value term
END
```

11.10 Log Files

Some examples of log files created during commissioning can be inserted here.

11.11 Panels

For more information on any of these panels use the command

```
% man <panel_name>
```

In many of these panels, this information can also be displayed by selecting "Help → Display man page" on the panel menu bar.

11.11.1Configuration panels

- ./vcins/src/vcinsStartup.pan
- ./vcins/src/vcinsStartupDev.pan
  - Normal and expert startup panels.
- ./vcins/src/vcinsFilterConfig.pan
- ./vcins/src/vcinsFilterConfigDev.pan
  - Filter wheel configuration display and expert adjustment panels.
- ./vcins/src/vcinsThermalConfig.pan
- ./vcins/src/vcinsThermalConfigDev.pan
  - Thermal control configuration display and expert adjustment panels.
- ./vcins/src/vcinsWcsConfig.pan
- ./vcins/src/vcinsWcsConfigDev.pan
- ./vcins/src/vcinsWcsScale.pan
- ./vcins/src/vcinsWcsScaleDev.pan
  - World Coordinates configuration display and expert adjustment panels.
- ./vcins/src/vcinsHwAvail.pan
- ./vcins/src/vcinsHwAvailDev.pan
  - Hardware availability display and expert adjustment panels.

11.11.2 ICS panels

- ./ICS/vcipan/src/vcipanControl.pan
    - Top level ICS engineering panel.
- ./ICS/vcipan/src/vcipanThermalControl.pan
    - Thermal control panel
- ./ICS/vcipan/src/vcipanLs332Manual.pan
    - Lakeshore 332 temperature set point manual control panel.
- ./ICS/vcipan/src/vcipanDeviceHealth.pan
    - ICS device health panel.

11.11.3 HOWFS panels

- ./HOWFS/vchpan/src/vchpanControl.pan
    - Top level HOWFS control panel
- ./HOWFS/vchpan/src/vchpanCalc.pan
    - HOWFS filter parameter calculation panel. (Used to generate the numbers that go into the HOWFS PAF files described in section 11.6.2.)
- ./HOWFS/vchpan/src/vchpanNullCoeffs.pan
    - HOWFS null coefficients display panel.
- ./HOWFS/vchpan/src/vchpanStartupCoeffs.pan
    - HOWFS startup coefficients display panel.
- ./HOWFS/vchpan/src/vchpanOptical.pan
    - HOWFS optical constants display panel.
- ./HOWFS/vchpan/src/vchpanModes.pan
    - HOWFS wavefront modes display panel.
- ./HOWFS/vchpan/src/vchpanZernikes.pan
    - HOWFS Zernike coefficients display panel.
- ./HOWFS/vchpan/src/vchpanSimplexDetails.pan
    - HOWFS simplex details display panel.
- ./HOWFS/vchpan/src/vchpanSimplexStripChart1.pan
    - Strip chart showing the change in simplex relative tolerance over time, for the simplex inner loop.
- ./HOWFS/vchpan/src/vchpanSimplexStripChart2.pan
    - Strip chart showing the change in coefficient difference over time, for the simplex outer loop.

11.11.4 OS panels

- ./OS/vcopan/src/vcopanControl.pan
    - Top level OS control panel
- ./OS/vcopan/src/vcopanEngineering.pan
    - OS engineering panel
- ./OS/vcopan/src/vcopanStatus.pan
    - Top level OS status panel
- ./OS/vcopan/src/vcopanSimStatus.pan

    - Device simulation status panel
- ./OS/vcopan/src/vcopanFilterStatus.pan
    - Filter wheel status panel
- ./OS/vcopan/src/vcopanIrdcsRtd.pan
    - Real-time data display selector panel

11.12 Error Files

All of the following files contain the descriptions for the errors listed in section 10 on page 177.

- ./ICS/vcilsm/ERRORS/vcilsm_ERRORS
- ./ICS/vcilsm/ERRORS/vcilsmERRORS.IDX
- ./ICS/vcilsc/ERRORS/vcilsc_ERRORS
- ./ICS/vcilsc/ERRORS/vcilscERRORS.IDX
- ./ICS/vcitpg/ERRORS/vcitpg_ERRORS
- ./ICS/vcitpg/ERRORS/vcitpgERRORS.IDX
- ./ICS/vcihb/ERRORS/vcihb_ERRORS
- ./ICS/vcihb/ERRORS/vcihbERRORS.IDX
- ./ICS/vci/ERRORS/vci_ERRORS
- ./ICS/vci/ERRORS/vciERRORS.IDX
- ./HOWFS/vtialib/ERRORS/vtialib_ERRORS
- ./HOWFS/vtialib/ERRORS/vtialibERRORS.IDX
- ./HOWFS/vchoia/ERRORS/vchoia_ERRORS
- ./HOWFS/vchoia/ERRORS/vchoiaERRORS.IDX
- ./OS/vco/ERRORS/vco_ERRORS
- ./OS/vco/ERRORS/vcoERRORS.IDX

__oOo__