indiIT
individual IT solutions and consulting

Results of Code Review of OSIRIS high-level on-board Software (OSIRIS Library)

Revision: 0.7

IIT-OSI-OCL-0002

2009-06-15

Prepared by
Tim Wittrock

Inhaltsverzeichnis

1 Scope

1.1 Change record

Issue	Date	Change Ref. / Reason (AI, RID, e-mail etc.)	Status N New D Deleted M Modified	Change	Description of Change
0.1	2009-03-02		N	all	Initial version
0.2	2009-04-07	Review progress	M	2.2	continued
			N	2.3	added
			N	3	started
0.3	2009-04-15	Review progress	M	3	completed
			N	2.2.5	started
0.4	2009-04-16	Review progress	M	2.2.5	completed
			N	2.2.9	added
0.5	2009-04-24	Review progress	M	2.2.9.7, 2.2.9.11 - 2.2.9.17	added
			N	2.6	Summary added
			N	2.3.3	Shutter optimization demon check added
0.6	2009-06-03	Review progress	N	2.7	File coverage summary added
			N	2.2.8.2 - 2.2.8.20, 2.2.10, 2.2.13 - 2.2.15, 2.3.4	added
			N	4	Examination of semaphores / resources inserted
			M	2.2.1.3, 2.2.1.4, 2.2.3.2, 2.2.3.5, 2.2.17	updated
0.7	2009-06-15	Review progress	N	2.2.11, 2.2.12, 2.2.17 - 2.2.23, 2.3.2, 2.3.5, 2.4	Review added.
			M	2.2.24, 2.6, 2.7	Updated
			M	4	Some annotations added

1.2 Purpose of this document

This document reflects the results of the high-level on-board software (the so-called OSIRIS library) code review for Rosetta/OSIRIS, mainly version 2.1 Patchlevel 2 Release 1 (date 2008-07-07). Some sections reflect the review of the software development branch (snapshots of 2009-02-24 and 2009-05-15), for parts which will be replaced on-board for sure. It lists the UDPs and their potential risks and shows fault trees of the most important parts of the OSIRIS library and demons.

Most of the listed risks are not critical, but particularly with regard to the ability to detect errors as soon as possible

(before it might result in a significant malfunction), even uncritical errors should not be ignored as they might indicate problems at other parts of the system in an early state.

1.3 Applicable Documents

[AD1] Results of Code Review and Fault-Tree Analysis of OSIRIS low-level on-board Software, indi-IT GmbH, IIT-OSI-OCL-0001

[AD2] RTL code review, IDA 2008, presented in OSIRIS S/W Review Meeting at IDA (RO-RIS-MPAE-MN-114), 2009-01-23

2 Potential Code Flaws

The reviewed source code is distributed over several files, which will be addressed in the following sub-sections. Each file contains different functions. Dependencies (like function calls or synchronization via events or semaphores) and potential flaws will be listed for each of these functions separately.

Compliant	Checked properties which are correct are just mentioned.
Minor critical	**Issues which do not influence the correct execution but affect the visibility of minor errors (like timeouts or uncritical but out-of-range values) mainly are marked yellow. These should be revised.**
Critical	**Bugs which might influence the correct execution are marked red. These need to be revised.**

Especially functions which are accessible via RTL should check all arguments and not rely on enumeration types. The result of these checks should be returned to the calling UDP.

2.1 Preface

2.1.1 Boolean Datatypes

Although the OCL compiler should support boolean data types, all boolean values are stored in a pseudo-boolean enumeration data type in the high level software. Therefore a couple of type checks will not be active (like using integer values as boolean). Another consequence is that functions might not rely on the limitation of boolean input parameters to the values 0 and 1.

Due to a bug in the OCL compiler up to version 4.16 it is recommended to keep the current state instead of switching to real OCL built-in boolean data type (the OCL bug is non-critical, since it does not causes wrong UDP code but rejects some boolean expressions which would be valid for a correct OCL compiler).

The current high-level code makes extensive use of expressions like “if (unsigned integer && bool)”. This should be avoided, because it not only would cause “using numeric value as boolean”-warnings (if activated), but it might also cause (quite surprisingly) larger code than “if ((unsigned integer > 0) && bool)” (for signed integer values the code size is the same for both implementations).

		Integer as boolean	Cleaned version	Optimized version
Signed integer	**Source code**	if (I && b) {}	if ((I != 0) && b) {}	if ((((unsigned long) I) > 0) && b) {}
	Resulting token code	LDI 5 6 0 TESTI 6 6 0 AND 6 3 7 FLGI 7 0 0 JMPZ 1 1 0	EQVI4 0 5 6 NOTB 6 7 0 AND 7 3 6 FLGI 6 0 0 JMPZ 1 1 0	ULSV 0 5 6 AND 6 3 7 FLGI 7 0 0 JMPZ 1 1 0
Unsigned integer	**Source code**	if (ul && b) {}	if ((ul > 0) && b)	-
	Resulting token code	LDI 4 6 0 TESTI 6 6 0 AND 6 3 7 FLGI 7 0 0 JMPZ 1 1 0	ULSV 0 4 6 AND 6 3 7 FLGI 7 0 0 JMPZ 1 1 0	

Many locations with room for improvement (when a debugged OCL compiler is available) are shown in this list:

Warning: OsirisLib/TMHandler.ocl at line 96: Using numeric value as boolean ()
Warning: OsirisLib/TMHandler.ocl at line 142: Using numeric value as boolean ()
Warning: OsirisLib/TMHandler.ocl at line 144: Using numeric value as boolean ()

Warning: OsirisLib/TMHandler.ocl at line 548: Using numeric value as boolean ()
Warning: OsirisLib/TMHandler.ocl at line 906: Using numeric value as boolean ()
Warning: OsirisLib/TMHandler.ocl at line 1234: Using numeric value as boolean (heater_block_counter)
Warning: OsirisLib/TMHandler.ocl at line 1267: Using numeric value as boolean (lock_count)
Warning: OsirisLib/ProcessHandler.ocl at line 54: Using numeric value as boolean
Warning: OsirisLib/ProcessHandler.ocl at line 106: Using numeric value as boolean (lock_count)
Warning: OsirisLib/ProcessHandler.ocl at line 149: Using numeric value as boolean
Warning: OsirisLib/ProcessHandler.ocl at line 153: Using numeric value as boolean
Warning: OsirisLib/ProcessHandler.ocl at line 166: Using numeric value as boolean (1)
Warning: OsirisLib/ProcessHandler.ocl at line 169: Using numeric value as boolean
Warning: OsirisLib/ProcessHandler.ocl at line 320: Using numeric value as boolean ()
Warning: OsirisLib/ProcessHandler.ocl at line 366: Using numeric value as boolean
Warning: OsirisLib/ProcessHandler.ocl at line 515: Using numeric value as boolean ()
Warning: OsirisLib/ProcessHandler.ocl at line 735: Using numeric value as boolean ()
Warning: OsirisLib/IMManager.ocl at line 31: Using numeric value as boolean ()
Warning: OsirisLib/IMManager.ocl at line 40: Using numeric value as boolean ()
Warning: OsirisLib/IMManager.ocl at line 514: Using numeric value as boolean
Warning: OsirisLib/IMManager.ocl at line 517: Using numeric value as boolean
Warning: OsirisLib/IMManager.ocl at line 550: Using numeric value as boolean
Warning: OsirisLib/IMManager.ocl at line 550: Using numeric value as boolean
Warning: OsirisLib/IMManager.ocl at line 556: Using numeric value as boolean ()
Warning: OsirisLib/IMManager.ocl at line 709: Using numeric value as boolean ()
Warning: OsirisLib/IMManager.ocl at line 930: Using numeric value as boolean
Warning: OsirisLib/MMBManager.ocl at line 92: Using numeric value as boolean (1)
Warning: OsirisLib/MMBManager.ocl at line 173: Using numeric value as boolean (1)
Warning: OsirisLib/MMBManager.ocl at line 233: Using numeric value as boolean (1)
Warning: OsirisLib/MMBManager.ocl at line 435: Using numeric value as boolean (remain)
Warning: OsirisLib/MMBManager.ocl at line 480: Using numeric value as boolean
Warning: OsirisLib/MMBManager.ocl at line 777: Using numeric value as boolean ()
Warning: OsirisLib/MMBManager.ocl at line 779: Using numeric value as boolean ()
Warning: OsirisLib/MMBManager.ocl at line 1151: Using numeric value as boolean
Warning: OsirisLib/MMBManager.ocl at line 1151: Using numeric value as boolean
Warning: OsirisLib/MMBManager.ocl at line 1160: Using numeric value as boolean ()
Warning: OsirisLib/MMBManager.ocl at line 1288: Using numeric value as boolean ()
Warning: OsirisLib/MMBManager.ocl at line 1415: Using numeric value as boolean
Warning: OsirisLib/MMBManager.ocl at line 1415: Using numeric value as boolean
Warning: OsirisLib/MMBManager.ocl at line 1421: Using numeric value as boolean ()
Warning: OsirisLib/MMBManager.ocl at line 1427: Using numeric value as boolean ()
Warning: OsirisLib/MMBManager.ocl at line 1702: Using numeric value as boolean ()
Warning: OsirisLib/MMBManager.ocl at line 1704: Using numeric value as boolean ()
Warning: OsirisLib/MMBManager.ocl at line 1895: Using numeric value as boolean ()
Warning: OsirisLib/MMBManager.ocl at line 1963: Using numeric value as boolean ()
Warning: OsirisLib/MMBManager.ocl at line 2046: Using numeric value as boolean ()
Warning: OsirisLib/MMBManager.ocl at line 2497: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 359: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 455: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 516: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 559: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 584: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 651: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 871: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 886: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 909: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 1109: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 1116: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 1123: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 1130: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 1709: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 1713: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 1717: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 1721: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 1725: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 1729: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 1733: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 1737: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 1741: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 1746: Using numeric value as boolean ()

Warning: OsirisLib/ParameterTable.ocl at line 1751: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 1765: Using numeric value as boolean ()
Warning: OsirisLib/ParameterTable.ocl at line 1780: Using numeric value as boolean ()
Warning: OsirisLib/ImageLib.ocl at line 490: Using numeric value as boolean (uEnBin)
Warning: OsirisLib/StreamedImageProcessing.ocl at line 353: Using numeric value as boolean (uEnBin)
Warning: OsirisLib/StreamedImageProcessing.ocl at line 396: Using numeric value as boolean
Warning: OsirisLib/StreamedImageProcessing.ocl at line 396: Using numeric value as boolean
Warning: OsirisLib/StreamedImageProcessing.ocl at line 410: Using numeric value as boolean
Warning: OsirisLib/StreamedImageProcessing.ocl at line 410: Using numeric value as boolean
Warning: OsirisLib/StreamedImageProcessing.ocl at line 437: Using numeric value as boolean
Warning: OsirisLib/StreamedImageProcessing.ocl at line 437: Using numeric value as boolean
Warning: OsirisLib/StreamedImageProcessing.ocl at line 449: Using numeric value as boolean (binsize)
Warning: OsirisLib/StreamedImageProcessing.ocl at line 472: Using numeric value as boolean (uEnBin)
Warning: OsirisLib/StreamedImageProcessing.ocl at line 523: Using numeric value as boolean (rcount)
Warning: OsirisLib/StreamedImageProcessing.ocl at line 527: Using numeric value as boolean (ec)
Warning: OsirisLib/StreamedImageProcessing.ocl at line 554: Using numeric value as boolean ()
Warning: OsirisLib/StreamedImageProcessing.ocl at line 566: Using numeric value as boolean ()
Warning: OsirisLib/StreamedImageProcessing.ocl at line 571: Using numeric value as boolean ()
Warning: OsirisLib/StreamedImageProcessing.ocl at line 576: Using numeric value as boolean ()
Warning: OsirisLib/StreamedImageProcessing.ocl at line 599: Using numeric value as boolean ()
Warning: OsirisLib/StreamedImageProcessing.ocl at line 618: Using numeric value as boolean
Warning: OsirisLib/StreamedImageProcessing.ocl at line 618: Using numeric value as boolean
Warning: OsirisLib/StreamedImageProcessing.ocl at line 765: Using numeric value as boolean
Warning: OsirisLib/StreamedImageProcessing.ocl at line 765: Using numeric value as boolean
Warning: OsirisLib/StreamedImageProcessing.ocl at line 838: Using numeric value as boolean ()
Warning: OsirisLib/StreamedImageProcessing.ocl at line 853: Using numeric value as boolean
Warning: OsirisLib/StreamedImageProcessing.ocl at line 858: Using numeric value as boolean
Warning: OsirisLib/StreamedImageProcessing.ocl at line 895: Using numeric value as boolean ()
Warning: OsirisLib/StreamedImageProcessing.ocl at line 925: Using numeric value as boolean ()
Warning: OsirisLib/StreamedImageProcessing.ocl at line 965: Using numeric value as boolean ()
Warning: OsirisLib/StreamedImageProcessing.ocl at line 968: Using numeric value as boolean (uEnBin)
Warning: OsirisLib/StreamedImageProcessing.ocl at line 972: Using numeric value as boolean ()
Warning: OsirisLib/StreamedImageProcessing.ocl at line 989: Using numeric value as boolean
Warning: OsirisLib/StreamedImageProcessing.ocl at line 989: Using numeric value as boolean
Warning: OsirisLib/StreamedImageProcessing.ocl at line 1003: Using numeric value as boolean
Warning: OsirisLib/StreamedImageProcessing.ocl at line 1008: Using numeric value as boolean
Warning: OsirisLib/StreamedImageProcessing.ocl at line 1013: Using numeric value as boolean
Warning: OsirisLib/StreamedImageProcessing.ocl at line 1087: Using numeric value as boolean ()
Warning: OsirisLib/StreamedImageProcessing.ocl at line 1089: Using numeric value as boolean ()
Warning: OsirisLib/StreamedImageProcessing.ocl at line 1095: Using numeric value as boolean ()
Warning: OsirisLib/HKCheck.ocl at line 333: Using numeric value as boolean
Warning: OsirisLib/HKCheck.ocl at line 333: Using numeric value as boolean
Warning: OsirisLib/HKCheck.ocl at line 334: Using numeric value as boolean
Warning: OsirisLib/HKCheck.ocl at line 334: Using numeric value as boolean
Warning: OsirisLib/HKCheck.ocl at line 335: Using numeric value as boolean
Warning: OsirisLib/HKCheck.ocl at line 335: Using numeric value as boolean
Warning: OsirisLib/HKCheck.ocl at line 336: Using numeric value as boolean
Warning: OsirisLib/HKCheck.ocl at line 336: Using numeric value as boolean
Warning: OsirisLib/ShutterHandler.ocl at line 322: Using numeric value as boolean
Warning: OsirisLib/ShutterHandler.ocl at line 358: Using numeric value as boolean
Warning: OsirisLib/ShutterHandler.ocl at line 375: Using numeric value as boolean (PulseCount)
Warning: OsirisLib/ShutterHandler.ocl at line 379: Using numeric value as boolean (PulseCount)
Warning: OsirisLib/ShutterHandler.ocl at line 583: Using numeric value as boolean ()
Warning: OsirisLib/ShutterHandler.ocl at line 1070: Using numeric value as boolean (RtlBoxcarFilter)
Warning: OsirisLib/ShutterHandler.ocl at line 1077: Using numeric value as boolean (RtlBwFilter)
Warning: OsirisLib/ShutterHandler.ocl at line 1224: Using numeric value as boolean (code)
Warning: OsirisLib/ShutterHandler.ocl at line 1463: Using numeric value as boolean ()
Warning: OsirisLib/ShutterHandler.ocl at line 1684: Using numeric value as boolean (RtlLinRegVector)
Warning: OsirisLib/ShutterHandler.ocl at line 1753: Using numeric value as boolean ()
Warning: OsirisLib/ShutterHandler.ocl at line 2123: Using numeric value as boolean (PulseCount)
Warning: OsirisLib/ShutterHandler.ocl at line 2127: Using numeric value as boolean (PulseCount)
Warning: OsirisLib/ShutterOptimize.ocl at line 35: Using numeric value as boolean
Warning: OsirisLib/ShutterOptimize.ocl at line 35: Using numeric value as boolean
Warning: OsirisLib/ShutterOptimize.ocl at line 35: Using numeric value as boolean
Warning: OsirisLib/ShutterOptimize.ocl at line 914: Using numeric value as boolean (count)
Warning: OsirisLib/ShutterErrorHandling.ocl at line 204: Using numeric value as boolean ()
Warning: OsirisLib/ShutterErrorHandling.ocl at line 339: Using numeric value as boolean ()

```
Warning: OsirisLib/ShutterErrorHandling.ocl at line 454: Using numeric value as boolean (she_error_history[])
Warning: OsirisLib/ShutterErrorHandling.ocl at line 458: Using numeric value as boolean ()
Warning: OsirisLib/MCBHandler.ocl at line 1122: Using numeric value as boolean
Warning: OsirisLib/MCBHandler.ocl at line 1127: Using numeric value as boolean
Warning: OsirisLib/CRBHandler.ocl at line 958: Using numeric value as boolean ()
Warning: OsirisLib/CRBHandler.ocl at line 1034: Using numeric value as boolean ()
Warning: OsirisLib/CRBHandler.ocl at line 1041: Using numeric value as boolean ()
Warning: OsirisLib/CRBHandler.ocl at line 1069: Using numeric value as boolean ()
Warning: OsirisLib/CRBHandler.ocl at line 1076: Using numeric value as boolean ()
Warning: OsirisLib/CRBHandler.ocl at line 1103: Using numeric value as boolean
Warning: OsirisLib/CRBHandler.ocl at line 1103: Using numeric value as boolean
Warning: OsirisLib/CRBHandler.ocl at line 1156: Using numeric value as boolean ()
Warning: OsirisLib/CRBHandler.ocl at line 1308: Using numeric value as boolean
Warning: OsirisLib/CRBHandler.ocl at line 1322: Using numeric value as boolean
Warning: OsirisLib/CRBHandler.ocl at line 1557: Using numeric value as boolean ()
Warning: OsirisLib/CRBHandler.ocl at line 1897: Using numeric value as boolean
Warning: OsirisLib/CRBHandler.ocl at line 1921: Using numeric value as boolean
Warning: OsirisLib/CRBHandler.ocl at line 2070: Using numeric value as boolean
Warning: OsirisLib/CRBHandler.ocl at line 2093: Using numeric value as boolean
Warning: OsirisLib/UserLib.ocl at line 57: Using numeric value as boolean
Warning: OsirisLib/UserLib.ocl at line 61: Using numeric value as boolean
Warning: OsirisLib/UserLib.ocl at line 61: Using numeric value as boolean
Warning: OsirisLib/UserLib.ocl at line 62: Using numeric value as boolean
Warning: OsirisLib/UserLib.ocl at line 65: Using numeric value as boolean ()
Warning: OsirisLib/UserLib.ocl at line 423: Using numeric value as boolean ()
Warning: OsirisLib/UserLib.ocl at line 477: Using numeric value as boolean ()
Warning: OsirisLib/UserLib.ocl at line 760: Using numeric value as boolean
Warning: OsirisLib/UserLib.ocl at line 760: Using numeric value as boolean
Warning: OsirisLib/UserLib.ocl at line 821: Using numeric value as boolean ()
Warning: OsirisLib/UserLib.ocl at line 871: Using numeric value as boolean ()
Warning: OsirisLib/TC.ocl at line 1171: Using numeric value as boolean ()
Warning: OsirisLib/TC.ocl at line 1229: Using numeric value as boolean ()
Warning: OsirisLib/TC.ocl at line 1231: Using numeric value as boolean ()
Warning: OsirisLib/TC.ocl at line 1234: Using numeric value as boolean ()
Warning: OsirisLib/TC.ocl at line 1294: Using numeric value as boolean ()
Warning: OsirisLib/TC.ocl at line 1309: Using numeric value as boolean (time_between_exposures)
Warning: OsirisLib/ShutterCalibrate.ocl at line 196: Using numeric value as boolean ()
Warning: OsirisLib/ShutterCalibrate.ocl at line 197: Using numeric value as boolean ()
Warning: OsirisLib/ShutterCalibrate.ocl at line 202: Using numeric value as boolean ()
Warning: OsirisLib/ShutterCalibrate.ocl at line 203: Using numeric value as boolean ()
Warning: OsirisLib/ShutterCalibrate.ocl at line 207: Using numeric value as boolean
Warning: OsirisLib/ShutterCalibrate.ocl at line 207: Using numeric value as boolean
Warning: OsirisLib/ShutterCalibrate.ocl at line 238: Using numeric value as boolean ()
Warning: OsirisLib/ShutterCalibrate.ocl at line 268: Using numeric value as boolean ()
Warning: OsirisLib/ShutterCalibrate.ocl at line 305: Using numeric value as boolean (arg1)
Warning: OsirisLib/ShutterCalibrate.ocl at line 307: Using numeric value as boolean (arg2)
```

2.1.2 Enumeration Datatypes

Operations involving enumeration datatypes are checked by the OCL compiler for correct usage of the enumeration, but the compiler does only warn about potential (or actual) misuses, it does not treat them as errors. Furthermore it is possible to suppress these warnings completely. The code review revealed a couple of functions which rely on correct usage of enumeration types (otherwise they might cause array limitation exceedings for example). For safety reasons it would be necessary to check enumeration parameters for valid values, but for speed reasons these runtime checks should be left out.

Both, the current OSIRIS library and the development branch as well, have been compiled for this code review with all enumeration warnings activated to ensure that no misuses occur in the UDP code. No enumeration misuses were found in the current on-board software (only boolean type warnings were detected as mentioned above), which means that the missing value checks are definitely no significant risk for the execution of the OSIRIS library at the moment. Nevertheless, these missing checks have been mentioned in the following documentation, because they could cause problems in future versions of the UDP code, if enumeration misuses are added, as shown by the following sub-section, which shows warnings of the development branch.

2.1.2.1 Compiler Warnings

The following warnings are produced by the OCL compiler during the compilation of the development branch snapshot of 2009-05-15:

Warning: OsirisLib/OsiTypes.h at line 162: Useless operation (0+)
Warning: OsirisLib/ProcessHandler.ocl at line 158: Ineffective code (constant as condition)
Warning: OsirisLib/MMBManager.ocl at line 62: Ineffective code (constant as condition)
Warning: OsirisLib/MMBManager.ocl at line 143: Ineffective code (constant as condition)
Warning: OsirisLib/MMBManager.ocl at line 203: Ineffective code (constant as condition)
Warning: OsirisLib/MMBManager.ocl at line 1658: Useless operation (-0)
Warning: OsirisLib/MMBManager.ocl at line 1716: Useless operation (0+)
Warning: OsirisLib/PCMHandler.ocl at line 515: Assignment of different enumeration types
Warning: OsirisLib/PCMHandler.ocl at line 515: Assignment to enum may exceed destination range
Warning: OsirisLib/PCMHandler.ocl at line 521: Assignment of different enumeration types
Warning: OsirisLib/PCMHandler.ocl at line 521: Assignment to enum may exceed destination range
Warning: OsirisLib/PCMHandler.ocl at line 529: Assignment of different enumeration types
Warning: OsirisLib/PCMHandler.ocl at line 529: Assignment to enum may exceed destination range
Warning: OsirisLib/PCMHandler.ocl at line 531: Assignment of different enumeration types
Warning: OsirisLib/PCMHandler.ocl at line 531: Assignment to enum may exceed destination range
Warning: OsirisLib/PCMHandler.ocl at line 645: Assignment of different enumeration types
Warning: OsirisLib/PCMHandler.ocl at line 645: Assignment to enum may exceed destination range
Warning: OsirisLib/PCMHandler.ocl at line 651: Assignment of different enumeration types
Warning: OsirisLib/PCMHandler.ocl at line 651: Assignment to enum may exceed destination range
Warning: OsirisLib/PCMHandler.ocl at line 658: Assignment of different enumeration types
Warning: OsirisLib/PCMHandler.ocl at line 658: Assignment to enum may exceed destination range
Warning: OsirisLib/PCMHandler.ocl at line 717: Assignment of different enumeration types
Warning: OsirisLib/PCMHandler.ocl at line 717: Assignment to enum may exceed destination range
Warning: OsirisLib/PCMHandler.ocl at line 719: Assignment of different enumeration types
Warning: OsirisLib/PCMHandler.ocl at line 719: Assignment to enum may exceed destination range

(These warnings are caused by using a hkInterface constant as eBool type parameter in the call to IFControl)

Warning: OsirisLib/ShutterHandler.ocl at line 380: Useless operation (+0)
Warning: OsirisLib/ShutterHandler.ocl at line 1806: Assignment to enum may exceed destination range

(This warning is caused by a immediate 0 verbosity mode parameter for SendMessage)

Warning: OsirisLib/ShutterHandler.ocl at line 2154: Useless operation (+0)
Warning: OsirisLib/ShutterHandler.ocl at line 2156: Useless operation (+0)
Warning: OsirisLib/UserLib.ocl at line 960: Useless operation (0+)
Warning: OsirisLib/UserLib.ocl at line 982: Useless operation (0+)
Warning: OsirisLib/UserLib.ocl at line 1018: Useless operation (0+)
Warning: OsirisLib/UserLib.ocl at line 1044: Useless operation (0+)
Warning: demons/DownlinkManager.ocl at line 126: Useless operation (0+)
Warning: demons/DownlinkManager.ocl at line 159: Useless operation (0+)
Warning: demons/SHMOptimize.ocl at line 157: Useless operation (+0)
Warning: demons/SHMOptimize.ocl at line 159: Useless operation (+0)

2.2 OsirisLib

IsValidImage is declared twice in part1_prototypes.h

2.2.1 MMBManager.ocl

2.2.1.1 MMBStat

Dependencies	Calls	ReleaseResAccess, GetTime, AcquireResAccess, mmb_IsManagerOK, IsMMBOn
	Called by	MMBManagerInit

	Synchronization	Acquires and releases RES_MMB_MAN
Error handling	Input checking	-
	Return codes	FALSE on error
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Loop limited to max block numbers
	Memory access	-
	Blocking	-

2.2.1.2 SelectMMB

Dependencies	Calls	SendMessage
	Called by	TCSelectMainConfig, TCSelectRedConfig
	Synchronization	-
Error handling	Input checking	**No. NOT IMPLEMENTED at all**
	Return codes	Always TRUE
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.3 MMBIFWrite

unnecessary 'safe' declaration

Dependencies	Calls	AcquireResAccess, IsResourceOwnedByUDP
	Called by	MMBInitHandleMemory, IMBlockToMMB, mmb_WriteFAT, FlushStreamWriter, SetImageHeaderToMMB, TCInitPlainfileRead, SetImageControlHeaderToMMB
	Synchronization	Acquires RES_MMB_WRITE_DIB0 or RES_MMB_WRITE_DIB1, has to be released by caller
Error handling	Input checking	-
	Return codes	Acquired DIB
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Endless loop possible if no DIB accessible.
	Memory access	-
	Blocking	-

2.2.1.4 MMBIFRead

unnecessary 'safe' declaration

Dependencies	Calls	AcquireResAccess, CalcTimeDiff, GetTime, IsResourceOwnedByUDP, ReleaseResAccess
	Called by	ExtractPulsesFromImage, GetImageHeaderFromMMB, GetImageontrolHeader-

		FromMMB, MMBBlockToIM, MMBPoweredOnSize, mmb_ReadMemManagerQuiet, MMBMemoryDump, TCPlainfileRecovery
	Synchronization	Acquires RES_MMB_READ_DIB0 or RES_MMB_READ_DIB1, RES_MMB_WRITE_DIB0 or RES_MMB_WRITE_DIB1. Resource has to be released by calling function.
Error handling	Input checking	**transfer_size is not used at all**
	Return codes	Acquired DIB
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Possibly endless loop if no DIB is accessible
	Memory access	-
	Blocking	-

2.2.1.5 mmb_IsManagerOK

Dependencies	Calls	-
	Called by	Many functions
	Synchronization	-
Error handling	Input checking	-
	Return codes	FALSE if manager is not ok
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.6 mmb_ReadMemManagerQuiet

Dependencies	Calls	FromMMB, IsMMBOn, MMBIFRead, ReleaseResAccess, SendMessage
	Called by	mmb_ReadMemManager, MMBManagerInit
	Synchronization	-
Error handling	Input checking	No, verbosity level is forwarded directly
	Return codes	FALSE on error **result is always FALSE if FAT1 is inconsistent (“ok” is not reset to TRUE for the FAT2 check)**
	Timeouts	-
	Aborts	If MMB is off, with error code
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.7 mmb_ReadMemManager

Dependencies	Calls	IsMMBOn, mmb_ReadMemManagerQuiet, SendMessage

	Called by	-
	Synchronization	-
Error handling	Input checking	-
	Return codes	FALSE on error
	Timeouts	-
	Aborts	With TM on error
Potential flaws	Loops	5 times to try loading
	Memory access	-
	Blocking	-

2.2.1.8 mmb_WriteFAT

Dependencies	Calls	IsMMBOn, MMBIFWrite, ReleaseResAccess, ToMMB
	Called by	-
	Synchronization	-
Error handling	Input checking	-
	Return codes	-
	Timeouts	-
	Aborts	**if MMB is switched of, without notification**
Potential flaws	Loops	Limited to size of mmbManager / 256
	Memory access	-
	Blocking	-

2.2.1.9 mmb_FindUnusedMemNode

Dependencies	Calls	-
	Called by	MMBAlloc, MMBShrink, MMBSplit
	Synchronization	-
Error handling	Input checking	-
	Return codes	ILLEGAL_HANDLE if no free block available
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Limited to max number of nodes
	Memory access	-
	Blocking	-

2.2.1.10 mmb_AddMemList

Dependencies	Calls	-
	Called by	-
	Synchronization	-
Error handling	Input checking	yes

	Return codes	FALSE on illegal input
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.11 mmb_InsertBeforeMemList

Dependencies	Calls	-
	Called by	MMBAlloc
	Synchronization	-
Error handling	Input checking	No, validity ensured by caller
	Return codes	**Always TRUE**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.12 mmb_InsertAfterMemList

Dependencies	Calls	-
	Called by	MMBShrink, MMBSplit
	Synchronization	-
Error handling	Input checking	No, validity ensured by callers
	Return codes	**Always TRUE**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.13 mmb_RemoveFromMemList

Dependencies	Calls	mmb_ConsolidateMemory
	Called by	-
	Synchronization	-
Error handling	Input checking	No, validity is guaranteed by caller
	Return codes	**Always TRUE**
	Timeouts	-
	Aborts	-

Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.14 mmb_ConsolidateMemory

Dependencies	Calls	mmb_RemoveFromMemList
	Called by	MMBFree
	Synchronization	-
Error handling	Input checking	yes
	Return codes	Ignores result of mmb_RemoveFromMemList
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Possibly endless loop if nodes are circular chained
	Memory access	-
	Blocking	-

2.2.1.15 mmb_FindTempBlock

Dependencies	Calls	-
	Called by	MMBAlloc
	Synchronization	-
Error handling	Input checking	-
	Return codes	ILLEGAL_HANDLE if no temp block of sufficient size is found
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Possibly endless loop if nodes are circular chained
	Memory access	-
	Blocking	-

2.2.1.16 mmb_FindFreeBlock

Dependencies	Calls	-
	Called by	MMBAlloc
	Synchronization	-
Error handling	Input checking	-
	Return codes	ILLEGAL_HANDLE if no free block is found
	Timeouts	-
	Aborts	If found block is not more than 50% larger than required size
Potential flaws	Loops	Possibly endless loop if nodes are circular chained
	Memory access	-
	Blocking	-

2.2.1.17 IsMMBOn

Dependencies	Calls	DPUReadHK
	Called by	Many functions
	Synchronization	-
Error handling	Input checking	-
	Return codes	FALSE if MMB is off
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.18 MMBManagerReset

Dependencies	Calls	AcquireResAccess, IsMMBOn, MMBPoweredOnSize, ReleaseResAccess, SendMessage
	Called by	MMBManagerInit
	Synchronization	Acquiring and releasing RES_MMB_MAN
Error handling	Input checking	-
	Return codes	-
	Timeouts	-
	Aborts	TM message on abort
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.19 MMBManagerInit

Dependencies	Calls	SendMessage, mmb_ReadMemManagerQuiet, MMBManagerReset, MMBStat, IsMMBOn
	Called by	StartMMB
	Synchronization	
Error handling	Input checking	-
	Return codes	FALSE on error ignoring return code of MMBStat (should be no problem here)
	Timeouts	-
	Aborts	TM messages on aborts
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.20 StartMMB

Dependencies	Calls	SendMessage, MMBConf, TableCheck, IsMMBOn, MMBSwitch, MMBManagerInit
	Called by	-
	Synchronization	-
Error handling	Input checking	-
	Return codes	FALSE on error
	Timeouts	Waiting 3 minutes for the MMB switching on
	Aborts	On error with error code
Potential flaws	Loops	With fix number of max. 40 runs.
	Memory access	-
	Blocking	-

2.2.1.21 StopMMB

Dependencies	Calls	IsMMBOn, MMBSwitch, SendMessage
	Called by	-
	Synchronization	- **sleeps for 1 second after setting mmb_is_initiated to FALSE, probably waiting for all possibly pending accesses to be finished. Would it be possible to acquire RES_MMB_MAN for safely syncing?**
Error handling	Input checking	-
	Return codes	FALSE if MMB is still on after 1 second.
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.22 MMBOptimizeMemBlockLocation

Dependencies	Calls	SendMessageArg, IMFree, MMBFree, IMBlockToMMB, MMBBlockToIM
	Called by	MMBDefragment
	Synchronization	-
Error handling	Input checking	Yes.
	Return codes	-1 on error, 1 on move, 0 on unchanged (comment) **“MMB_OPT_CHANGED_HANDLE” message if IMBlockToMMB returns “ILLEGAL_HANDLE” (in this case, the data of this block is lost, too)** Return value is (other than commented) “TRUE” (1) whenever the new handle is not “ILLEGAL_HANDLE”, regardless whether the block has been moved or not (this is a comment error only).
	Timeouts	-
	Aborts	On illegal input with error code -1

Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.23 MMBDefragment

Dependencies	Calls	MMBOptimizeMemBlockLocation, AcquireResAccess, ReleaseResAccess
	Called by	-
	Synchronization	Acquiring and releasing RES_MMB_MAN
Error handling	Input checking	-
	Return codes	**Always TRUE**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Possibly endless loop on circular chain of nodes (indicates MMB manager corruption). **“iteration” is incremented but never used.**
	Memory access	-
	Blocking	RES_MMB_MAN might kept blocked on endless loop

2.2.1.24 MMBAlloc

Dependencies	Calls	mmb_IsManagerOK, MMBFree, mmb_FindFreeBlock, mmb_FindtempBlock, mmb_FindUnusedMemNode, mmb_InsertBeforeMemList, AcquireResAccess, ReleaseResAccess, SendMessageArg, SendMessage, SendArray
	Called by	CRBAllocAcquireMem, IMBlockToMMB, MMBNamedAlloc
	Synchronization	Acquiring and releasing RES_MMB_MAN
Error handling	Input checking	yes
	Return codes	ILLEGAL_HANDLE on error
	Timeouts	-
	Aborts	**Not reverting the change of the free mem block's size (hRet) on abort after failing mmb_InsertBeforeMemList might cause MMB manager inconsistency (called function cannot fail and returns always true, branch can be removed completely)**
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.25 MMBFree

Dependencies	Calls	SendMessage, mmb_ConsolidateMemory, ReleaseResAccess, mmb_IsManagerOK, AcquireResAccess, SendMessageArg
	Called by	Many functions
	Synchronization	Acquiring and releasing RES_MMB_MAN
Error handling	Input checking	yes
	Return codes	FALSE on error

	Timeouts	-
	Aborts	On error, with error code
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.26 MMBShrink

Dependencies	Calls	AcquieresAccess, mmb_FindUnusedMemNode, mmb_InsertAfterMemList, mmb_IsManagerOK, ReleaseResAccess, SendMessage, SendMessageArg
	Called by	CloseStreamWriter
	Synchronization	Acquiring and releasing RES_MMB_MAN
Error handling	Input checking	yes
	Return codes	FALSE on error
	Timeouts	-
	Aborts	On error with error code
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.27 MMBSplit

Dependencies	Calls	AcquieresAccess, mmb_FindUnusedMemNode, mmb_InsertAfterMemList, mmb_IsManagerOK, ReleaseResAccess, SendMessage
	Called by	TCPlainfileRecovery
	Synchronization	Acquiring and releasing RES_MMB_MAN
Error handling	Input checking	Yes (flags not checked but masked).
	Return codes	FALSE on error
	Timeouts	-
	Aborts	On error with error code
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.28 MMBGetQueue

Dependencies	Calls	-
	Called by	DeleteImagesFromQueue, MoveImagesToQueue
	Synchronization	-
Error handling	Input checking	yes
	Return codes	FALSE on error
	Timeouts	-
	Aborts	-

Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.29 MMBSetQueue

Dependencies	Calls	AcquireResAccess, mmb_IsManagerOK, ReleaseResAccess, SendMessage
	Called by	MoveImagesToQueue, QueueImage
	Synchronization	Acquires and releases RES_MMB_MAN
Error handling	Input checking	yes
	Return codes	FALSE on error
	Timeouts	-
	Aborts	On error with error code
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.30 MMBSetFlag

Dependencies	Calls	AcquireResAccess, mmb_IsManagerOK, ReleaseResAccess, SendMessage
	Called by	MoveImagesToQueue, QueueImage
	Synchronization	Acquires and releases RES_MMB_MAN
Error handling	Input checking	**“flag” not checked**
	Return codes	FALSE on error
	Timeouts	-
	Aborts	On error with error code
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.31 MMBIsFlagSet

Dependencies	Calls	SendMessage
	Called by	DeleteImagesFromQueue, MoveImagesToQueue
	Synchronization	-
Error handling	Input checking	yes
	Return codes	FALSE on error
	Timeouts	-
	Aborts	On error with error code
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.32 MMBNamedFind

Dependencies	Calls	AcquireresAccess, ReleaseResAccess, mmb_IsManagerOK
	Called by	MMBNamedAlloc, MMBNamedFree, TCInitPlainfileRead, MMBSetHandleName, TCPlainfileRecovery
	Synchronization	Acquiring and releasing RES_MMB_MAN
Error handling	Input checking	yes
	Return codes	ILLEGAL_HANDLE on error
	Timeouts	-
	Aborts	On error with error code
Potential flaws	Loops	Possibly endless loop if MMB chain is circular (indicates MMB Manager corruption)
	Memory access	-
	Blocking	-

2.2.1.33 MMBSetHandleName

Dependencies	Calls	AcquireresAccess, ReleaseResAccess, mmb_IsManagerOK, SendMessage
	Called by	-
	Synchronization	Acquires and releases RES_MMB_MAN
Error handling	Input checking	yes
	Return codes	FALSE on error. **Variable “ok” can be removed (is not touched anyway) and constant TRUE used instead.**
	Timeouts	-
	Aborts	On invalid input or unprepared environment with error code.
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.34 MMBNamedAlloc

Dependencies	Calls	SendMessage, MMBNamedFind, MMBAlloc
	Called by	-
	Synchronization	-
Error handling	Input checking	yes
	Return codes	ILLEGAL_HANDLE on error
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.35 MMBNamedFree

Dependencies	Calls	SendMessage, MMBFree, MMBNamedFind
	Called by	-
	Synchronization	-
Error handling	Input checking	No, forwarded to MMBNamedFInd
	Return codes	FALSE on error
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.36 MMBGetHandleSize

Dependencies	Calls	mmb_IsManagerOK, SendMessage
	Called by	DeleteImagesFromQueue, MoveImagesToQueue, MMBBlockToIM, OpenStream-Writer, MMBInitHandleMemory, TCPlainfileRecovery
	Synchronization	-
Error handling	Input checking	yes
	Return codes	0 on error
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.37 MMBGetHandleName

Dependencies	Calls	mmb_IsManagerOK, SendMessage
	Called by	-
	Synchronization	-
Error handling	Input checking	yes
	Return codes	FALSE on invalid or unnamed handle
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.38 MMBIsHandleValid

Dependencies	Calls	mmb_IsManagerOK

	Called by	MMBBlockToIM
	Synchronization	-
Error handling	Input checking	yes
	Return codes	FALSE if invalid
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.39 MMBLargestBlock

Dependencies	Calls	AcquireResAccess, mmb_IsManagerOK, ReleaseResAccess
	Called by	-
	Synchronization	Acquires and releases RES_MMB_MAN
Error handling	Input checking	-
	Return codes	0 on error
	Timeouts	-
	Aborts	On MMB manager not OK
Potential flaws	Loops	**Possibly endless loop if circular MMB node chain (indicates MMB manager corruption)**
	Memory access	-
	Blocking	RES_MMB_MAN keeps locked until the loop finishes

2.2.1.40 MMBLock

Dependencies	Calls	AcquireResAccess, SendMessageArg, mmb_IsManagerOK, ReleaseResAccess, SendMessage
	Called by	Many functions
	Synchronization	Locks and releases RES_MMB_MAN **First checks are executed before RES_MMB_MAN is locked – preconditions might change afterwards (at least the allocate-state)**
Error handling	Input checking	yes
	Return codes	**0 (FALSE) if MMB Manager is not OK,** 0xFFFFFFFF (NULL) if input is invalid
	Timeouts	-
	Aborts	Locks and releases RES_MMB_MAN
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.41 MMBUnlock

Dependencies	Calls	AcquireResAccess, SendMessageArg, mmb_IsManagerOK, ReleaseResAccess,

		SendMessage
	Called by	Many functions
	Synchronization	Locks and releases RES_MMB_MAN
Error handling	Input checking	Yes
	Return codes	FALSE on error
	Timeouts	-
	Aborts	On error
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.42 MMBFirstHandleOfType

Dependencies	Calls	-
	Called by	DeleteImagesFromQueue, MoveImagesToQueue
	Synchronization	-
Error handling	Input checking	Indirect check for valid handle, not checking for valid type specifier.
	Return codes	ILLEGAL_HANDLE on error
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Max. mmbManager.header.last_used runs
	Memory access	-
	Blocking	-

2.2.1.43 MMBNextHandleOfType

Dependencies	Calls	-
	Called by	DeleteImagesFromQueue, MoveImagesToQueue
	Synchronization	-
Error handling	Input checking	Indirect check for valid handle, not checking for valid type specifier.
	Return codes	ILLEGAL_HANDLE on error
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Max. mmbManager.header.last_used runs
	Memory access	-
	Blocking	-

2.2.1.44 MMBGetHandleFlags

Dependencies	Calls	-
	Called by	MMBBlockToIM
	Synchronization	-

Error handling	Input checking	Yes.
	Return codes	0 on illegal input
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.45 MMBRangeCheck

Dependencies	Calls	mmb_IsManagerOK, SendMessage
	Called by	-
	Synchronization	-
Error handling	Input checking	yes
	Return codes	FALSE on error
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.46 MMBPoweredOnSize

Dependencies	Calls	MMBIFRead, ReleaseResAccess, MMBStatus
	Called by	MMBManagerReset
	Synchronization	Aquires and releases resource
Error handling	Input checking	-
	Return codes	MMB size in words
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.1.47 MMBInitHandleMemory

Dependencies	Calls	IMAlloc, MMBIFWrite, IMFree, MMBUnlock, MMBLock, MMBGetHandleSize, ToMMB, ReleaseResAccess
	Called by	CRBAllocAcquireMem
	Synchronization	Locks MMB, **does not unlock MMB if IMAlloc fails** Acquires DIB access and releases it.
Error handling	Input checking	Forwards input directly to MMBLock

	Return codes	Returns FALSE on error.
	Timeouts	-
	Aborts	On error.
Potential flaws	Loops	Potentially endless loop if no DIB is accessible
	Memory access	ok
	Blocking	**MMB might keep locked if function aborts after failing IMAlloc.**

2.2.1.48 IsMMBConsistent

Dependencies	Calls	AcquireResAccess, ReleaseResAccess, IsMMBOn, SendMessage
	Called by	-
	Synchronization	Resource Access
Error handling	Input checking	No, interpreting all other than 0 as TRUE
	Return codes	Returns false on error. Ignores return code of AcquireResAccess (is always true, since it would wait endlessly otherwise)
	Timeouts	Waiting without timeout
	Aborts	On error detection with return code FALSE
Potential flaws	Loops	Limited to number of MMB blocks
	Memory access	ok
	Blocking	Releasing all acquired resources

2.2.2 IMManager.ocl

Why can't IM handles be locked like MMB?

2.2.2.1 im_FindUnusedMemNode

Dependencies	Calls	-
	Called by	IMAlloc, IMShrink
	Synchronization	-
Error handling	Input checking	-
	Return codes	ILLEGAL_HANDLE on error
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	Limited to max number of nodes
	Blocking	-

2.2.2.2 im_AddMemList

Dependencies	Calls	-
	Called by	-
	Synchronization	-
Error handling	Input checking	yes

	Return codes	FALSE on illegal input
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.2.3 im_InsertBeforeMemList

Dependencies	Calls	-
	Called by	IMAlloc
	Synchronization	-
Error handling	Input checking	Not checking, caller ensures validity
	Return codes	**Always TRUE**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.2.4 im_InsertAfterMemList

Dependencies	Calls	-
	Called by	IMShrink
	Synchronization	-
Error handling	Input checking	No, caller ensures validity
	Return codes	**Always TRUE**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.2.5 im_RemoveFromMemList

Dependencies	Calls	-
	Called by	im_ConsolidateMemory
	Synchronization	-
Error handling	Input checking	No, validity guaranteed by caller
	Return codes	**Always TRUE**
	Timeouts	-

	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.2.6 im_FindFreeBlock

Dependencies	Calls	-
	Called by	IMAlloc
	Synchronization	-
Error handling	Input checking	No, size could be checked for reasonable value.
	Return codes	ILLEGAL_HANDLE on error
	Timeouts	-
	Aborts	-
Potential flaws	Loops	**Possible endless loop if nodes are circular chained**
	Memory access	-
	Blocking	-

2.2.2.7 im_ConsolidateMemory

Dependencies	Calls	im_RemoveFromMemList
	Called by	IMFree
	Synchronization	-
Error handling	Input checking	yes
	Return codes	-
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.2.8 IMManagerInit

Dependencies	Calls	-
	Called by	OsirisLibInit
	Synchronization	-
Error handling	Input checking	-
	Return codes	Always TRUE (local variable “ok” can be removed and replaced by constant TRUE, or return type can be set to void)
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Fix number of repetitions (IM_MAX_BLOCK_COUNT)

	Memory access	-
	Blocking	-

2.2.2.9 IMAlloc

Dependencies	Calls	AcquireResAccess, im_FindFreeBlock, im_FindUnusedMemNode, im_InsertBeforeMemList, ReleaseResAccess, SendMessage
	Called by	Several functions
	Synchronization	Acquiring and releasing RES_IM_MAN
Error handling	Input checking	no.
	Return codes	ILLEGAL_HANDLE on error, in-out parameter “address” remains unchainged
	Timeouts	-
	Aborts	**Not reverting the change of the free mem block's size (hRet) on abort after failing im_InsertBeforeMemList might cause IM manager inconsistency (called function cannot fail and returns always true, branch can be removed completely)**
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.2.10 IMAddressOf

Dependencies	Calls	-
	Called by	StreamIMImageTo, CRBReorderImage, GetImageHeaderFromIM, GetImageControlHeaderFromIM, AutoExpose, SetImageControlHeaderToIM, IMBlockToMMB, SendImageFromIMToPlainFile, SetImageHeaderToIM
	Synchronization	-
Error handling	Input checking	yes
	Return codes	NULL on error
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.2.11 IMFree

Dependencies	Calls	AcquireResAccess, ReleaseResAccess, SendMessageArg, im_ConsolidateMemory, SendMessage
	Called by	A lot of functions
	Synchronization	Acquiring and releasing RES_IM_MAN
Error handling	Input checking	yes
	Return codes	FALSE on error
	Timeouts	-
	Aborts	On error with error code

Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.2.12 IMShrink

Dependencies	Calls	AcquireResAccess, SendMessageArg, im_FindUnusedMemNode, im_InsertAfter-MemList, ReleaseResAccess, SendMessage
	Called by	-
	Synchronization	Acquiring and releasing RES_IM_MAN
Error handling	Input checking	yes
	Return codes	FALSE on error
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.2.13 IMGetHandleSize

Dependencies	Calls	SendMessage
	Called by	CRBReorderImage, IMBlockToMMB, SendImageFromIMToPlainFile
	Synchronization	-
Error handling	Input checking	yes
	Return codes	0 on error
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.2.14 IMGetHandleFlags

Dependencies	Calls	-
	Called by	IMBlockToMMB
	Synchronization	-
Error handling	Input checking	Yes. **Not checking for INVALID_HANDLE (currently defined 0xFFFFFFFF, which is actually > IM_MAX_BLOCK_COUNT, but it is checked separately in other functions)**
	Return codes	The flags
	Timeouts	-
	Aborts	-

Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.2.15 IMIsHandleValid

Dependencies	Calls	-
	Called by	IMBlockToMMB
	Synchronization	-
Error handling	Input checking	yes
	Return codes	FALSE on error
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.2.16 IMLargestBlock

Dependencies	Calls	-
	Called by	-
	Synchronization	-
Error handling	Input checking	-
	Return codes	0 on error
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Potentially endless loop if nodes are circularly chained
	Memory access	-
	Blocking	-

2.2.2.17 IMReleaseHandlesOwnedByVM

Dependencies	Calls	SendMessageArg, ReleaseResAccess, AcquireResAccess, IMFree, im_Consolidate-Memory
	Called by	HandleUDPMgrAbort
	Synchronization	Acquiring and releasing RES_IM_MAN
Error handling	Input checking	Not effective, checking “vm >= 0” (inside of the loop) does not make sense for UNS32
	Return codes	**Return value is missing**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Potentially endless loop if nodes are circularly chained
	Memory access	-

	Blocking	-

2.2.3 ProcessHandler.ocl

Opposite to the other sections, this section does not review the current flight software but the development branch (snapshot of 20090224) of the on-board software, because some very important changes will take place in the synchronization. The current flight software is known to be vulnerable to illegal concurrent data access (due to lack of reliable locking mechanisms), which will be fixed by the new version which refers to OS semaphores for locking. On the one hand this has a huge advantage for access locking, on the other hand this semaphore handling might also cause more serious blocking than the former mechanism, which makes it necessary to check correct usage carefully.

It has to be kept in mind that abortion of a UDP which holds a semaphore will block this semaphore – it is not released automatically. To minimize this risk, semaphores should be hold as short as possible (which is actually the case in the checked UDPs).

2.2.3.1 ProcessHandlerInit

Dependencies	Calls	-
	Called by	OsirisLibInit
	Synchronization	-
Error handling	Input checking	-
	Return codes	**Always TRUE**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Fix number of repetitions
	Memory access	-
	Blocking	-

2.2.3.2 IsResourceOwnedByUDP

'safe' declaration seems to be unnecessary for this function

Dependencies	Calls	-
	Called by	MMBIFRead, MMBIFWrite
	Synchronization	No, not necessary, since return value depends on settings made by the calling UDP only. Concurrent access is possible without problems.
Error handling	Input checking	No, enumeration input is used as index for an array.
	Return codes	FALSE if resource is not owned by the current UDP
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	Possibly array limits exceeding
	Blocking	-

2.2.3.3 QueryResAccess

Dependencies	Calls	-
	Called by	HKLimitCheckUnit, IsUnitOn, MCBControler, MCBMoveMechanismDual
	Synchronization	-

Error handling	Input checking	No, illegal array access possible
	Return codes	FALSE if resource is already in use
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	Possibly array limits exceeding
	Blocking	-

2.2.3.4 AcquireResAccess

Dependencies	Calls	CritSecBegin, CritSecEnd
	Called by	-
	Synchronization	Enters and leaves RES_CRIT_SECTION. **"ControlArray.res_acquire_flag[vm]" is write-accessed outside of the critical section which might cause concurrent access. The flag seems not to be used (read) at all and should be removed.**
Error handling	Input checking	**No, "resource" might be out of range (used as array index)**
	Return codes	FALSE if resource could not be acquired
	Timeouts	-
	Aborts	-
Potential flaws	Loops	**Checking for successful resource locking is not necessary any more, since acquisition is locked reliably.**
	Memory access	Possibly array limits exceeding
	Blocking	-

2.2.3.5 ReleaseResAccess

Dependencies	Calls	CritSecBegin, CritSecEnd, SendMessageArg
	Called by	-
	Synchronization	Enters and leaves RES_CRIT_SECTION.
Error handling	Input checking	**No, "resource" might be out of range (used as array index)**
	Return codes	-
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	**Resource might be freed from other machines with TM error message – is this intended?**

2.2.3.6 SendOCLEvent

Dependencies	Calls	MsgPutW
	Called by	SendDemonRequest
	Synchronization	Sends message MAIL_OCL

Error handling	Input checking	no.
	Return codes	-
	Timeouts	Waiting without timeout for message reception by any destination
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	If no thread listens for MAIL_OCL

2.2.3.7 MarkDemonRunning

Dependencies	Calls	GetTime, SendMessageArg
	Called by	MarkDemonAlive
	Synchronization	-
Error handling	Input checking	No, relies on enumeration type check: input is used directly as array index
	Return codes	FALSE if demon is already marked “running”
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	Possibly array limitation violation
	Blocking	-

2.2.3.8 MarkDemonAlive

Dependencies	Calls	GetTime, MarkDemonRunning, SendMessageArg
	Called by	CheckForDemonRequest
	Synchronization	-
Error handling	Input checking	No, relies on enumeration type check: input is used directly as array index
	Return codes	-
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	Possibly array limitation violation
	Blocking	-

2.2.3.9 MarkDemonTerminated

Dependencies	Calls	-
	Called by	-
	Synchronization	-
Error handling	Input checking	No, relies on enumeration type check: input is used directly as array index not checking whether the demon is active or not (this might be not necessary, but would be symmetric to MarkDemonRunning)
	Return codes	-

	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	Possibly array limitation violation
	Blocking	-

2.2.3.10 MarkDemonPaused

Dependencies	Calls	-
	Called by	-
	Synchronization	-
Error handling	Input checking	No, relies on enumeration type check: input is used directly as array index not checking whether the demon is already paused or not (this might be not necessary, but would be symmetric to MarkDemonRunning)
	Return codes	-
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	Possibly array limitation violation
	Blocking	-

2.2.3.11 MarkDemonResumed

Dependencies	Calls	-
	Called by	-
	Synchronization	-
Error handling	Input checking	No, relies on enumeration type check: input is used directly as array index not checking whether the demon is paused or not (this might be not necessary, but would be symmetric to MarkDemonRunning)
	Return codes	-
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	Possibly array limitation violation
	Blocking	-

2.2.3.12 SendDemonRequest

Dependencies	Calls	SendMessage, SendMessageArg, SendOCLEvent
	Called by	CalibrateShutter, DemonPing
	Synchronization	Sends OCL events OCLCMD_REQUEST_OCLEVENT_STOP and OCLCMD_REQUEST_OCLEVENT_PING
Error handling	Input checking	Not checking demon ID, which is used as index, relies on enumeration type check.
	Return codes	FALSE on error

	Timeouts	600 times ~200ms waiting for acknowledge flag to be set, returning FALSE if timeout occurs, sending TM message in this case, too.
	Aborts	TM message of missing implementation for PROC_REQUEST_PAUSE and PROC_REQUEST_RESUME of DEMON_OCL_EVENT – return value is even in these cases TRUE. Will never occur. No message if OCL event demon is not running (not symmetric to handling of other demons). Will never occur.
Potential flaws	Loops	600 times waiting for ACK flag to be set by demon. Possibly the demon reacts exactly after the last wait – nevertheless failure notices are generated in that case. The request flags should be reset before the ACK flag is reset (probably the ACK flag should be checked before, after a short additional wait, just to be sure).
	Memory access	Array limitation violation possible
	Blocking	-

2.2.3.13 PostDemonRequest

Dependencies	Calls	SendMessageArg
	Called by	-
	Synchronization	-
Error handling	Input checking	No, request is not even masked to valid range
	Return codes	-
	Timeouts	-
	Aborts	TM if demon is not running
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.3.14 CheckForDemonRequest

Dependencies	Calls	MarkDemonAlive
	Called by	-
	Synchronization	-
Error handling	Input checking	Not checking demon ID, which is used as index, relies on enumeration type check.
	Return codes	Request ID
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.3.15 DemonPing

Dependencies	Calls	SendDemonRequest
	Called by	-
	Synchronization	-

Error handling	Input checking	No, forwarding demon ID unchecked, relies on enumeration type check.
	Return codes	Transfers the return value of SendDemonRequest immediately
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.3.16 IsDemonOk

Dependencies	Calls	CalcTimeDiff, GetTime
	Called by	-
	Synchronization	-
Error handling	Input checking	Not checking demon ID, which is used as index.
	Return codes	FALSE if demon does not respond on requests
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.4 StreamedImageProcessing.ocl

2.2.4.1 OpenStreamWriter

Dependencies	Calls	AcquireResAccess, IsMMBOn, MMBLock, MMBGetHandleSize, MMBUnlock, IMAlloc,
	Called by	StreamIMImageTo
	Synchronization	Syncs to RES_SCIENCE_LINK
Error handling	Input checking	Yes.
	Return codes	Ignored return code of AcquireResAccess, which is always TRUE (it won't return otherwise). Returns FALSE on error
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	Locks RES_SCIENCE_LINK

2.2.4.2 FlushStreamWriter

Dependencies	Calls	SendMessage, MMBIFWrite, ToMMB, ReleaseResAccess
	Called by	CloseStreamWriter, WriteToStream
	Synchronization	Syncs to RES_MMB_WRITE_DIB0 or RES_MMB_WRITE_DIB1

Error handling	Input checking	Yes.
	Return codes	FALSE on buffer overflow
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	(Locking RES_MMB_WRITE_DIB0 or RES_MMB_WRITE_DIB1 in called function MMBI-FWrite) Releasing RES_MMB_WRITE_DIB0 or RES_MMB_WRITE_DIB1

2.2.4.3 CloseStreamWriter

Dependencies	Calls	ReleaseResAccess, FlushStreamWriter, MMBUnlock, IMFree, MMBShrink
	Called by	StreamIMImageTo
	Synchronization	Synced to RES_SCIENCE_LINK
Error handling	Input checking	-
	Return codes	**Ignores result of MMBUnlock, IMFree, MMBShrink**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	Releases RES_SCIENCE_LINK or unlocks MMB

2.2.4.4 WriteToStream

Dependencies	Calls	SendScienceDataFromIM, SendMessageArg, FlushStreamWriter,
	Called by	RunStreamProcessor, StreamHandleSegment, StreamIMImageTo
	Synchronization	-
Error handling	Input checking	Input valid by design.
	Return codes	**All callers ignore the return value**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Loop over all elements of data.
	Memory access	-
	Blocking	-

2.2.4.5 InitProcessingWorkspace

Dependencies	Calls	IMAlloc
	Called by	StreamProcessSection
	Synchronization	-
Error handling	Input checking	No, memory might leak if it is called with reserved workspace.
	Return codes	Returns FALSE if IMAlloc fails

	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.4.6 CloseProcessingWorkspace

Dependencies	Calls	IMFree
	Called by	StreamProcessSection
	Synchronization	-
Error handling	Input checking	Yes.
	Return codes	Returns FALSE if input is invalid **ignores return code of IMFree**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.4.7 CalcCCDSectionCoords

Dependencies	Calls	GetPixels, GetLines
	Called by	InitSegmentHeader
	Synchronization	-
Error handling	Input checking	No, relies on enumeration check by OCL compiler. Default branch in switch would be safe.
	Return codes	Returns FALSE if resulting rectangle is 0
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.4.8 StreamFilterBadPixels

Dependencies	Calls	ImgRemoveBad, SendMessageArg
	Called by	RunStreamProcessor
	Synchronization	-
Error handling	Input checking	Mixing sizeof(bad_pixels.table) and BAD_PIXEL_TABLE_SIZE for size specification of table size (it is equal, so it is okay)
	Return codes	Returning TRUE if bad pixels have been removed, FALSE if unchanged
	Timeouts	-

	Aborts	-
Potential flaws	Loops	Loop over bad pixel table, limited to table size.
	Memory access	-
	Blocking	-

2.2.4.9 RunStreamProcessor

Dependencies	Calls	GetTime, StreamFilterBadPixels, ImgConv, ImgSqrt, ImgBin, SendMessage, SendMessageArg, Split16, Encode, CalcTimeDiff, PackSegmentHeader, WriteToStream
	Called by	StreamProcessSection
	Synchronization	-
Error handling	Input checking	Checking where necessary, relies on validity of I.mage data
	Return codes	Ignores return code of ImgConv and ImgBin (would indicate inconsistent image memory) **returns always TRUE, declaration of "BOOL ok = TRUE;" not necessary**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	While-loop to check bin_size, limited to 16 runs Nested for-loop, limited to 25 runs
	Memory access	-
	Blocking	-

2.2.4.10 StreamProcessSection

Dependencies	Calls	InitProcessingWorkspace, SendMessage, ImgCrop, RunStreamProcessor, CloseProcessingWorkspace
	Called by	StreamHandleSegment
	Synchronization	-
Error handling	Input checking	Partially. Unchecked parameters are validated by caller.
	Return codes	Ignores return code of ImgCrop (uncritical, error code here would indicate a very improbable major inconsistency in image data which occurred during this function's runtime) Ignores return code of CloseProcessingWorkspace (which could only fail if worspace is invalid, which will not occur at this point) Returns error code (FALSE) if StreamProcessSection fails, is checked by caller.
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Potentially virtually endless loops if mastersh.rect is unreasonable large (which implies that the image memory has been corrupted whilst the image was stored there)
	Memory access	-
	Blocking	-

2.2.4.11 InitSegmentHeader

Dependencies	Calls	CalcCCDSectionCoords

	Called by	StreamHandleSegment
	Synchronization	-
Error handling	Input checking	Validity of parameters ensured by caller
	Return codes	Returns FALSE if CalcCCDSectionCoords fails **Return code ignored by StreamHandleSegment**
	Timeouts	-
	Aborts	Returns FALSE if CalcCCDSectionCoords fails
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.4.12 StreamHandleSegment

Dependencies	Calls	InitSegmentHeader, PackSegmentHeader, WriteToStream, ImgBright, StreamProcessSection
	Called by	StreamIMImageTo
	Synchronization	-
Error handling	Input checking	ch is forwarded to InitSegmentHeader, StreamProcessSection. ch.comp_value is checked for CPV_SEND_IMAGE_RAW and/or CPV_MARK_THUMBNAIL set or not. Forwards “writer” to WriteToStream and StreamProcessSection without check, checks for “writer.destination == eStreamToTM” only (all other desination values are supposed to be “eStreamToMMB” sh is supposed to be correct, caller gets it directly from memory (although image memory is vulnerable it seems not to be reasonable to check, since images are stored in MMB most of the time). Correctness of pData depends on integrity of image memory. Valid segment_count is guaranteed by caller.
	Return codes	**No notification about failing InitSegmentHeader** **Ignores return value of WriteToStream and ImgBright** FALSE on error.
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	Illegal memory access possible on invalid pData (on corrupted IM)
	Blocking	-

2.2.4.13 StreamIMImageTo

Dependencies	Calls	CRBReorderImage, IMAddressOf, IsValidImage, SendMessage, OpenStreamWriter, PackImageHeader, WriteToStream, AcquireResAccess, ReleaseResAccess, StreamHandleSegment, SendEvent, SetImageHeaderToMMB, CloseStreamWriter
	Called by	StreamImageTo
	Synchronization	- (uses AcquireResAccess and ReleaseResAccess to prevent concurrent access to ressources)
Error handling	Input checking	Forwards “hImageSource” without checking to CRBReorderImage and IMAddressOf (might be ILLEGAL_HANDLE), but these functions check for validity.

		Forwards “destination” without checking to OpenStreamWriter (assumes all other than eStreamToTM to be eStreamToMMB; later the cases eStreamToTM and eStreamToMMB are checked separately without default or error case – low criticality, unless the enumeration warnings of the OCL compiler are not deactivated. Forwards “hImageDest” without checking to OpenStreamWriter and SetImageHeaderToMMB; both check this value (the latter also raises a TM message). **PackImageHeader's parameter “ih” contains unchecked “comp_type” and “comp_value” (the rest of the structure can be considered correct, since it is read from image directly)**
	Return codes	**Ignors return code of PackImageHeader, WriteToStream, AcquireResAccess, SetImageHeaderToMMB and CloseStreamWriter** Returns FALSE on error.
	Timeouts	-
	Aborts	If CRBReorderImage, IMAddressOf, IsValidImage or OpenStreamWriter fail.
Potential flaws	Loops	Possibly virtually endless loop if image header contains negative control.segment_count. Loop relies on validity of header data which resides in (vulnerable) image memory.
	Memory access	Illegal imMemory access might occur on illegal segment_count (which might caused by corrupted image memory). This would be detected at runtime and abort the UDP.
	Blocking	-

2.2.4.14 StreamImageTo

Dependencies	Calls	SendMessageArg, MMBBlockToIM, StreamIMImageTo, IMFree, GetImageControlHeaderFromMMB, SetImageControlHeaderToMMB
	Called by	DLProcessQueue, DLHandleThumbnails, AcquireDualImage, AcquireImageStored, AcquireImageLowPower, TCCompressStoredImage, TCSendStoredImage, SendImageFromMMB, SendImageFromIM, TestAFT
	Synchronization	-
Error handling	Input checking	ILLEGAL_HANDLE might be forwarded via “hIM” to StreamIMImageTo from parameter “hImageSource” if image source is not MMB (it is checked on lower level) mem_route is checked for eMemMMB only. **“destination”, “hImageDest”, “comp_type” and “comp_value” are forwarded to StreamIMImageTo without checking**
	Return codes	**Return code of IMFree ignored.** Return code of GetImageControlHeaderFromMMB and SetImageControlHeaderToMMB ignored, which should always work at this point, since the image has been written immediately before. **AcquireImageLowPower, AcquireImageStored, AcquireDualImage and TestAFT ignore the returned value.**
	Timeouts	-
	Aborts	If MMBBlockToIM fails, with error code (FALSE)
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.5 ImageLib.ocl

2.2.5.1 InitImageControlHeader

Dependencies	Calls	TopLevelStamp
	Called by	CRBAcquirePrepare
	Synchronization	-
Error handling	Input checking	Argument is output parameter only
	Return codes	-
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.5.2 IsValidImage

Dependencies	Calls	_memcmp
	Called by	TCPlainfileRecovery, QueueImage, GetImageControlHeaderFromMMB, SetImageHeaderToMMB, StreamIMImageTo, GetImageHeaderFromIM, GetImageHeaderFromMMB, SetImageHeaderToIM
	Synchronization	-
Error handling	Input checking	-
	Return codes	OK
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.5.3 GetImageControlHeaderFromMMB

Dependencies	Calls	FromMMB, IsMMBOn, IsValidImage, MMBIFRead, ReleaseResAccess, MMBLock, MMBUnlock
	Called by	DLProcessQueue, TCPlainfileRecovery, AcquireDualImage, QueueImage, StreamImageTo, AcquireImageStored, DeleteImagesFromQueue, MoveImagesToQueue, ExtractPulsesFromImage
	Synchronization	-
Error handling	Input checking	Indirectly by called function
	Return codes	Ignores return code of MMBUnlock, **Return value is ignored by DLHandleImage, TCPlainfileRecovery, AcquireDualImage, QueueImage, StreamImageTo, AcquireImageStored, DeleteImagesFromQueue, MoveImagesToQueue, ExtractPulsesFromImage**
	Timeouts	-
	Aborts	If MMB is not reachable, with FALSE

Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.5.4 SetImageControlHeaderToMMB

Dependencies	Calls	IsMMBOn, MMBIFWrite, MMBLock, MMBUnlock, ReleaseResAccess, ToMMB
	Called by	AcquireDualImage, QueueImage, DLProcessQueue, StreamImageTo, AcquireImageStored
	Synchronization	-
Error handling	Input checking	Does not check for valid image header.
	Return codes	Ignores return code of MMBUnlock, **return value ignored by all callers.**
	Timeouts	-
	Aborts	If MMB is not reachable, with FALSE
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.5.5 SetImageControlHeaderToIM

Dependencies	Calls	IMAddressOf
	Called by	AcquireDualImage, AcquireImageLowPower, AcquireImageStored
	Synchronization	-
Error handling	Input checking	Indirectly by called function
	Return codes	Return value ignored by all callers
	Timeouts	-
	Aborts	On illegal handle, with FALSE.
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.5.6 GetImageControlHeaderFromIM

Dependencies	Calls	IMAddressOf
	Called by	AcquireImageStored, AcquireImageLowPower, AutoExpose, AcquireDualImage, CRBReorderImage, DLProcessQueue
	Synchronization	-
Error handling	Input checking	Indirectly by called function
	Return codes	**Return value ignored by DLProcessQueue, AcquireDualImage, AutoExpose, AcquireImageLowPower, AcquireImageStored**
	Timeouts	-
	Aborts	On illegal handle, with FALSE.

Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.5.7 GetImageHeaderFromMMB

Dependencies	Calls	FromMMB, IsMMBOn, IsValidImage, MMBIFRead, ReleaseResAccess, MMBLock, MMBUnlock
	Called by	ExtractPulsesFromImage
	Synchronization	-
Error handling	Input checking	Indirectly by called function
	Return codes	Ignores return value of MMBUnlock (error is extremely improbable). **Return code ignored by ExtractPulsesFromImage**
	Timeouts	-
	Aborts	If MMB is not accessible, with FALSE
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.5.8 SetImageHeaderToMMB

Dependencies	Calls	IsMMBOn, IsValidImage, MMBIFWrite, MMBLock, MMBUnlock, ReleaseResAccess, ToMMB
	Called by	StreamIMImageTo
	Synchronization	-
Error handling	Input checking	Indirectly by called function
	Return codes	Ignores return value of MMBUnlock (error is extremely improbable). **Return code ignored by StreamIMImageTo**
	Timeouts	-
	Aborts	If MMB is not accessible, with FALSE
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.5.9 GetImageHeaderFromIM

Dependencies	Calls	IMAddressOf, IsValidImage
	Called by	ExtractPulsesFromImage
	Synchronization	-
Error handling	Input checking	Indirectly by called function
	Return codes	OK
	Timeouts	-
	Aborts	If provided handle is not accessible

Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.5.10 SetImageHeaderToIM

Dependencies	Calls	IMAddressOf, IsValidImage
	Called by	-
	Synchronization	-
Error handling	Input checking	Indirectly by called function
	Return codes	OK
	Timeouts	-
	Aborts	If provided handle is not accessible
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.5.11 CRBReorderImage

Dependencies	Calls	GetImageControlHeaderFromIM, IMAddressOf, GetLines, GetPixels, GetShutter, GetSize, IMGetHandleSize, SendMessage, Plausible, Reorder
	Called by	ExtractPulsesFromImage, StreamIMImageTo
	Synchronization	-
Error handling	Input checking	Yes, indirectly by sub-function called at the beginning of CRBOrderImage
	Return codes	FALSE on error, checked by callers
	Timeouts	-
	Aborts	On illegal handle, with error code FALSE
Potential flaws	Loops	-
	Memory access	Source and Target overlap (but source starts on lower address, which is required by Reorder) Modified header might be written back at the end of the 'if' branch instead of using a second 'if'
	Blocking	-

2.2.5.12 CalculateHistogram

Dependencies	Calls	ImgHist
	Called by	AutoExpose
	Synchronization	-
Error handling	Input checking	-
	Return codes	Return value ignored by AutoExpose, but value is always TRUE.
	Timeouts	-
	Aborts	-

Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.5.13 AcquireHKSnapshot

Dependencies	Calls	GetHKPacketData, SendMessage
	Called by	CRBAcquireFinalize
	Synchronization	-
Error handling	Input checking	Relies on enumeration type check, does not check
	Return codes	**Return code is ignored by CRBAcquireFinalize**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.6 ShutterHandler.ocl

The code review of this section refers to the development branch of the on-board software which is not used in space yet (snapshot of 20090224).

2.2.6.1 ShutterAcquireTestImage

Dependencies	Calls	IsUnitOn, CheckShutterTemperature, GetShutter, SendMessage, SendArray, ShutterBuildTestCmd, CRBToRAM, GetSize, ShutterConfigSHE, ShutterDelayForChange, ShutterSaveFiringTimestamp, ShutterSplitBladePulses, Split16, SendMessageArg, AcquireResAccess, ReleaseResAccess
	Called by	ShutterPerformeHomepush, ShutterClose, ShutterOpen, ShutterForceClosed, ShutterTestFunctionality, ShutterTuneAB, ShutterTuneMotion, ShutterTuneImpactVelOff, ShutterTuneFiltering
	Synchronization	Acquires and releases RES_NAC_CRB/RES_WAC_CRB
Error handling	Input checking	Relies on enumeration type eDIB, might fail if camera parameter is invalid and NAC is switched off.
	Return codes	FALSE on error, **no TM if NAC/WAC is not switched on, but message if temperature check fails.** **Ignores the return value of ShutterConfigSHE (which forwards the return value of CRBShutter)** **Ignores the return value of CRBToRAM**
	Timeouts	Waiting without timeout for resource access.
	Aborts	On unit switched off, with error code.
Potential flaws	Loops	Max. 4 repetitions
	Memory access	-
	Blocking	-

2.2.6.2 ShutterForceClosed

Dependencies	Calls	AcquireResAccess, ReleaseresAccess, IsShutterOpen, ShutterAcquireTestImage, ShutterBuildUnlockProfile, SendMessageArg
	Called by	ShutterClose, ShutterTestFunctionality, ShutterTuneFiltering, ShutterTuneTravel, ShutterTuneImpactVelOff, ShutterTuneMotion, ShutterTuneImpact
	Synchronization	Acquires and releases RES_NAC_CRB/RES_WAC_CRB
Error handling	Input checking	No, relies on enumeration type (for camera)
	Return codes	FALSE on error
	Timeouts	-
	Aborts	If lower and upper limits are reached with FALSE and message
Potential flaws	Loops	**Why is delta_vel alternating?**
	Memory access	-
	Blocking	-

2.2.6.3 ShutterConfigSHE

Dependencies	Calls	Many functions
	Called by	Even more functions
	Synchronization	Acquires and releases RES_NAC_CRB/RES_WAC_CRB
Error handling	Input checking	Partially. No shutter brake handling for WAC – is the hardware different? Answer: Yes.
	Return codes	Return value of CRBShutter
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.6.4 ShutterDelayForCharge

Dependencies	Calls	CalcTimeDiff, GetTime, SendMessageArg
	Called by	ShutterAcquireTestImage, CRBAcquirePrepare
	Synchronization	-
Error handling	Input checking	-
	Return codes	**Not sleeping at all if sleep_time is >= 13000, but no message about it**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.6.5 ShutterSaveFiringTimestamp

Dependencies	Calls	GetTime
	Called by	ShutterAcquiretestImage, CRBAcquirePrepare, CRBInitiateShutter
	Synchronization	-
Error handling	Input checking	-
	Return codes	-
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.6.6 ShutterInitWorkspace

Dependencies	Calls	IMAlloc
	Called by	Many functions
	Synchronization	-
Error handling	Input checking	No, relies on enumeration type
	Return codes	ILLEGAL_HANDLE on error
	Timeouts	-
	Aborts	If IMAlloc fails, with return code ILLEGAL_HANDLE
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.6.7 CheckShutterTemperature

Dependencies	Calls	CRBReadHK, HKGetCalibratedNAC, HKGetCalibratedWAC
	Called by	CRBCheckAcquirePreCon, CRBInitiateShutter, ShutterAcquireTestImage
	Synchronization	-
Error handling	Input checking	No, relies on enumeration type
	Return codes	FALSE if limit check fails **Ignores return value of CRBReadHK**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.6.8 ShutterBuildTestCmd

Dependencies	Calls	GetDuration

	Called by	ShutterAcquireTestImage
	Synchronization	-
Error handling	Input checking	No, just puts the values in the command
	Return codes	-
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.6.9 ShutterEnableNACBrake

Dependencies	Calls	SendMessageArg, ReleaseResAccess, IMFree, AcquireResAccess, IMAlloc, CRB-BuildCommand, CRBShutter, CRBToRAM, GetSize
	Called by	ShutterEnableNACBrake
	Synchronization	Acquiring and releasing RES_NAC_CRB
Error handling	Input checking	-
	Return codes	FALSE if IMAlloc fails
	Timeouts	-
	Aborts	On failing IMAlloc
Potential flaws	Loops	-
	Memory access	Allocating and freeing a buffer
	Blocking	-

2.2.6.10 SleepChargeTime

Dependencies	Calls	-
	Called by	ShutterErrorRecovery
	Synchronization	-
Error handling	Input checking	Only partially (zero sleep time), might cause very long sleep if parameter 'count' is too large (uncritical, since all calls use constant 1 or 2)
	Return codes	-
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.6.11 GetShutterErrorType

Dependencies	Calls	-
	Called by	ShutterErrorRecovery, IsShutterOpen
	Synchronization	-

Error handling	Input checking	-
	Return codes	Shutter error A, B, C or none.
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.6.12 IsShutterOpen

Dependencies	Calls	CRBShutter, ShutterConfigSHE, GetShutterErrorType
	Called by	ShutterClose, ShutterOpen, ShutterForceClosed, ShutterTuneAB, ShutterTestFunctionality, ShutterTuneMotion
	Synchronization	-
Error handling	Input checking	No, forwarding the parameter directly.
	Return codes	FALSE if shutter is closed; checking error D, too
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.6.13 ShutterOpen

Dependencies	Calls	Many functions
	Called by	CRBSetAcquireMode, ShutterTestFunctionality, ShutterTuneImpactVelOff, ShutterTuneAB, ShutterTuneFiltering
	Synchronization	Acquiring and releasing RES_NAC_CRB/RES_WAC_CRB
Error handling	Input checking	No, relies on enumeration type, uses parameter as array index
	Return codes	FALSE if shutter could not be opened, also TM message
	Timeouts	-
	Aborts	If already open with TRUE and message. FALSE and message if the desired unit is not powered on.
Potential flaws	Loops	Loop limited to max. 6 repetitions
	Memory access	Allocating (indirectly) and freeing workspace, 'camera' parameter might be out of range
	Blocking	-

2.2.6.14 ShutterClose

Dependencies	Calls	Many functions
	Called by	CRBSetAcquireMode, ShutterErrorRecovery, ShutterTuneAB, ShutterTuneMotion, ShutterTuneMotion, ShutterTuneImpactVelOff, ShutterTuneFiltering
	Synchronization	Acquiring and releasing RES_NAC_CRB/RES_WAC_CRB

Error handling	Input checking	No, relying on enumeration types
	Return codes	FALSE and message if the desired unit is not powered on, TRUE and message if shutter is already closed
	Timeouts	-
	Aborts	If already open with TRUE and message. FALSE and message if the desired unit is not powered on.
Potential flaws	Loops	-
	Memory access	Allocating (indirectly) and freeing workspace, 'camera' parameter might be out of range
	Blocking	**SleepChargeTime has been commented out but not finally removed**

2.2.6.15 RealizeAccelerationProfile

Dependencies	Calls	BessellFilter, RealizeLine, RtlBoxcarFilter, RtlBwFilter, SendArray, SendMessageArg
	Called by	BuildReferenceVelocityProfile, ShutterBuildUnlockProfile
	Synchronization	-
Error handling	Input checking	Partially. No message if “time” is smaller than “accel”, but taking minimum (current implementation could be realized by “UNS32 count = sizeof (time) <? sizeof (accel);”, too)
	Return codes	FALSE on error
	Timeouts	-
	Aborts	On error with FALSE
Potential flaws	Loops	All loops will end after a constant max. number of repetitions
	Memory access	-
	Blocking	-

2.2.6.16 ShutterCalcRefProfileTiming

Dependencies	Calls	SendMessageArg
	Called by	BuildReferenceVelocityProfile, ShutterTuneImpactVelOff
	Synchronization	-
Error handling	Input checking	If bp.fDecelPowerCorrFactor is negative, it is set to 1 without further messaging. Are values between 1.0 and 0.0 allowed? Answer: Yes.
	Return codes	Always TRUE, but TM message if profile generation failed
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.6.17 BuildReferenceVelocityProfile

Dependencies	Calls	RealizeAccelerationProfile, ShutterCalcRefProfileTiming

	Called by	BuildShutterPowerProfile
	Synchronization	-
Error handling	Input checking	No, relies on enumeration type
	Return codes	FALSE if RealizeAccelerationProfile fails
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.6.18 BuildShutterPowerProfile

Dependencies	Calls	BuildReferenceVelocityProfile
	Called by	ShutterBuildUnlockProfile, ShutterTestFunctionality, ShutterTuneAB, RealizeShutterProfile, ShutterTuneMotion, ShutterTuneTravel
	Synchronization	-
Error handling	Input checking	Partially, relies on enumeration type for range checking
	Return codes	FALSE on error
	Timeouts	-
	Aborts	On illegal input, with FALSE
Potential flaws	Loops	Max. repetitions limited by constants
	Memory access	-
	Blocking	-

2.2.6.19 RealizeShutterProfile

Dependencies	Calls	BuildShutterPowerProfile, ShutterInitWorkspace, CRBInitiateShutter, IMFree, IsUnitOn, SendMessage
	Called by	CRBInitiateShutter, ShutterTuneFiltering, CalibrateShutter, ShutterTuneImpact, ShutterTuneTravel, ShutterTuneMotion
	Synchronization	-
Error handling	Input checking	No, does not check for NAC or WAC
	Return codes	FALSE on error
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Potentially endless recursion CRBInitiateShutter ↔ RealizeShutterProfile, should never occur since RealizeShutterProfile creates a non-zero power profile.
	Memory access	-
	Blocking	-

2.2.6.20 ShutterSplitBladePulses

Dependencies	Calls	-
	Called by	ShutterPerformHomepush, ShutterAcquireTestImage

	Synchronization	-
Error handling	Input checking	-
	Return codes	**Always TRUE**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Limited to number of array elements
	Memory access	-
	Blocking	-

2.2.6.21 ExtractPulsesFromImage

Dependencies	Calls	Many functions
	Called by	-
	Synchronization	Releases access to resource acquired by MMBIFRead
Error handling	Input checking	No
	Return codes	FALSE on error
	Timeouts	-
	Aborts	On error with FALSE
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.6.22 ShutterPulseToVelocity

Dependencies	Calls	-
	Called by	CalculateShutterMotionStat
	Synchronization	-
Error handling	Input checking	Partially
	Return codes	FALSE if pulse_input is too small
	Timeouts	-
	Aborts	on illegal input, with FALSE
Potential flaws	Loops	Limited to size of input array
	Memory access	-
	Blocking	-

2.2.6.23 CalculateShutterStatFromMotion

Dependencies	Calls	RtlLinRegVector
	Called by	CalculateShutterMotionStat
	Synchronization	-
Error handling	Input checking	Yes, checks sizes of input buffers
	Return codes	FALSE on error

	Timeouts	-
	Aborts	If buffer sizes of time and vel do not fit or if RtlLinRegVector fail, with FALSE
Potential flaws	Loops	Limited to size of input buffer
	Memory access	-
	Blocking	-

2.2.6.24 CalculateShutterMotionStat

Dependencies	Calls	CalculateShutterStatFromMotion, SendMessage, SendMessageArg, ShutterPulseToVelocity
	Called by	ShutterTuneSingleBladeMotion, ShutterTuneFiltering, ShutterTuneImpactVelOff
	Synchronization	-
Error handling	Input checking	Yes, but relies on enumeration type.
	Return codes	FALSE on error
	Timeouts	-
	Aborts	On error, with FALSE (illegal blade or failing calculation of shutter status from motion) or with FALSE and TM message (insufficient shutter pulses in provided shutter workspace, failing conversion of raw pulse data into velocity data or blade impact detection).
Potential flaws	Loops	Loop limited to number of pulses
	Memory access	-
	Blocking	-

2.2.6.25 ShutterBuildUnlockProfile

Dependencies	Calls	BuildShutterPowerProfile, RealizeAccellerationProfile
	Called by	ShutterForcedClosed
	Synchronization	-
Error handling	Input checking	**Parameter camera is unused**
	Return codes	Always TRUE **Ignoring result of RealizeAccellerationProfile**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.6.26 ShutterBuildHomepushProfile

Dependencies	Calls	RealizeLine, RtlBoxcarFilter
	Called by	ShutterPerformHomepush
	Synchronization	-
Error handling	Input checking	No, relies on enumeration type check.

	Return codes	Always TRUE
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Nested loop, repetitions limited to 2 * HOMEPUSH_POINT_COINT (= 2*6) and 2 * SINGLE_BLADE_PROFILE_SIZE (= 2*128)
	Memory access	-
	Blocking	-

2.2.6.27 ShutterPeformHomepush

Dependencies	Calls	Many functions
	Called by	CRBAcquireFinalize, ShutterErrorRecovery, ShutterHomepushNAC, Shutter-HomepushWAC
	Synchronization	Acquires and releases RES_NAC_CRB/RES_WAC_CRB
Error handling	Input checking	Relies on enumeration type check
	Return codes	FALSE on error, **ignored by all callers**
	Timeouts	-
	Aborts	On failed build of HomepushProfile, with FALSE. No messages
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.6.28 ShutterHomepushNAC / ShutterHomepushWAC

Dependencies	Calls	ShutterPeformHomepush
	Called by	-
	Synchronization	-
Error handling	Input checking	-
	Return codes	**Ignores the return value of ShutterPeformHomepush**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.6.29 ShutterIsMotionOkForTuning

Dependencies	Calls	SendMessageArg
	Called by	ShutterTuneSingleBladeMotion
	Synchronization	-
Error handling	Input checking	No, relies on enumeration type check
	Return codes	FALSE if motion is not ok with TM message

	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.6.30 ShutterTuneSingleBladeMotion

Dependencies	Calls	CalculateShutterMotionStat, SendMessageArg, ShutterIsMotionOkForTuning, UpdatePID
	Called by	ShutterTestFunctionality, ShutterTuneMotion
	Synchronization	-
Error handling	Input checking	Relies on enumeration type checking
	Return codes	FALSE on error, with TM message (“return FALSE;” could be left out in “else if” branches)
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.6.31 ShutterTestFunctionality

Dependencies	Calls	IsShutterOpen, ShutterAcquireTestImage, ShutterForceClosed, BuildShutterPowerProfile, ShutterTuneSingleBladeMotion
	Called by	ShutterTestParam
	Synchronization	-
Error handling	Input checking	Relies on enumeration type check
	Return codes	FT_TEST_FAILURE if preconditions for test are not as required, FT_NOT_WORKING if shutter is not working as expected, FT_WORKING if blade is working
	Timeouts	-
	Aborts	If preconditions are not ok, with error code.
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.7 ShutterErrorHandling.ocl

The shutter error handling will be replaced by an updated version, so the UDPs from the development branch (snapshot of 20090224) have been checked.

2.2.7.1 EnterShutterSafemode

Dependencies	Calls	ShutterConfigSHE, SendMessageArg
	Called by	ShutterErrorRecovery, ShutterFunctionalLimitCheck

	Synchronization	-
Error handling	Input checking	Relies on enumeration type check
	Return codes	Ignores return code of ShutterConfigSHE (which is the return code of CRBShutter)
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.7.2 ShutterErrorRecovery

Dependencies	Calls	A lot of functions
	Called by	CRBAcquirePrepare, CRBAcquireFinalize, CRBInitiateShutter
	Synchronization	Acquires and releases RES_NAC_CRB/RES_WAC_CRB **No locking of RuntimeTable.shutter_state which could be (very improbably) set by CRBInitiateShutter concurrently.**
Error handling	Input checking	No, relies on enumeration type check
	Return codes	FALSE if recovery failed and safemode is entered TRUE if successfully recovered
	Timeouts	-
	Aborts	With FALSE and TM message if parameter table is not okay or shutter in safemode, with TRUE if shutter error recovery is already active
Potential flaws	Loops	Limited to 32 repetitions
	Memory access	-
	Blocking	Might block later error recoveries, if this one is aborted during execution.

Why is “as_exposure” checked two times in line 172ff, 260ff, 320ff and 384ff? Could be only one conditional jump.

Why shutter homepush after successful recovery of type A (line 239)?

Recovery does not handle changing error types during recovery (recovery might fail even if the original error has been recovered)

RuntimeTable.she_error_history is not modified for error type D.

2.2.7.3 ShutterFunctionalLimitCheck

Dependencies	Calls	EnableShutterSafemode
	Called by	-
	Synchronization	-
Error handling	Input checking	-
	Return codes	-
	Timeouts	-
	Aborts	**Does not enter safemode if uMinExpTimeForMotionOpt < LONG_EXP_TIME_FOR_OPT, no notification about this.**

Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.8 CRBHandler.ocl

These checks reviewed the file from the development branch state of 20090515.CRBSetAmplifier

Dependencies	Calls	TableCheck, SendMessage
	Called by	CRBSetUp, TestAFT, test_acquire_darks_and_bias, test_cal_lamp, test_dual, test_take_image
	Synchronization	-
Error handling	Input checking	Yes
	Return codes	FALSE on error, **ignored by all callers except CRBSetUp**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.8.1 CRBSetGain

Dependencies	Calls	TableCheck, SendMessage
	Called by	test_take_image, test_dual, test_cal_lamp, test_acquire_darks_and_bias, TestAFT, CRBSetUp
	Synchronization	-
Error handling	Input checking	Yes.
	Return codes	FALSE on error.
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.8.2 CRBSetADC

Dependencies	Calls	SendMessage, TableCheck
	Called by	CRBSetUp
	Synchronization	-
Error handling	Input checking	Yes, **negative values of “adc” (enum, signed integer) are not detected (other functions rely on enumeration type – why not here?)**
	Return codes	FALSE on error
	Timeouts	-

	Aborts	FALSE and error message on error
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.8.3 CRBSetBinningMode

Dependencies	Calls	SendMessage, TableCheck
	Called by	CRBSetUp
	Synchronization	-
Error handling	Input checking	Yes, **binning_mode is not checked for negative values (other functions rely on enumeration type – why not here?)**
	Return codes	FALSE on error
	Timeouts	-
	Aborts	On error, with FALSE and message
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.8.4 CRBSetWindow

Dependencies	Calls	SendMessage, TableCheck
	Called by	AutoExpose, CRBSetUp
	Synchronization	-
Error handling	Input checking	yes
	Return codes	FALSE on error, **but returns TRUE even if one or more coordinate limits are violated – is this the intend?**
	Timeouts	-
	Aborts	On illegal camera or parameter table, with message and FALSE
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.8.5 CRBSetFullframe

Dependencies	Calls	SendMessage, TableCheck
	Called by	CRBSetUp
	Synchronization	-
Error handling	Input checking	FALSE on error
	Return codes	-
	Timeouts	-

	Aborts	On illegal camera or parameter table, with message and FALSE
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.8.6 CRBSetAcquireMode

Dependencies	Calls	AcquireResAccess, ReleaseResAccess, SendMessage, ShutterClose, ShutterConfig-SHE, ShutterOpen, TableCheck
	Called by	TCNACOff, TCWACOff
	Synchronization	Aquires and releases RES_NAC_CRB or RES_WAC_CRB
Error handling	Input checking	yes
	Return codes	FALSE on error
	Timeouts	-
	Aborts	On error
Potential flaws	Loops	-
	Memory access	-
	Blocking	**Not unlocking RES_NAC_CRB or RES_WAC_CRB on illegal shutter mode**

2.2.8.7 CRBSetUp

Dependencies	Calls	CRBSetADC, CRBSetAmplifier, CRBSetBinningMode, CRBSetFullframe, CRBSetGain, CRBSetWindow, SendMessage
	Called by	CalibrateShutter
	Synchronization	-
Error handling	Input checking	Yes.
	Return codes	FALSE on error
	Timeouts	-
	Aborts	On error
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.8.8 CRBRequestExtra

Dependencies	Calls	SendMessage, TableCheck
	Called by	-
	Synchronization	-
Error handling	Input checking	Yes. BOOL values are compared to 0 (not FALSE)
	Return codes	FALSE on error
	Timeouts	-
	Aborts	On error, with FALSE

Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.8.9 CRBSetShutterConfig

Dependencies	Calls	CRBInitiateShutter, IsSwitchOn
	Called by	-
	Synchronization	-
Error handling	Input checking	**No, camera switch is set to WAC for all other than NAC (other functions generate error in that case)**
	Return codes	**Always TRUE**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.8.10 CRBInitiateShutter

Dependencies	Calls	_memcmp, CheckShutterTemperature, CRBSync, IsShutterPowerOn, RealizeShutterProfile, SendMessage, SendMessageArg, ShutterConfigSHE, ShutterErrorRecovery, ShutterSaveFiringTimestamp, TableCheck
	Called by	CalibrateShutter, CRBResetShutter, CRBSetShutterConfig, PowerShutterOn, RealizeShutterProfile, and ShutterTuneImpactVelOff
	Synchronization	-
Error handling	Input checking	yes
	Return codes	FALSE on error
	Timeouts	-
	Aborts	On illegal input or error with FALSE
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.8.11 CRBResetShutter

Dependencies	Calls	CRBInitiateShutter
	Called by	-
	Synchronization	-
Error handling	Input checking	**error_code and force_reset are unused**
	Return codes	-
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-

	Memory access	-
	Blocking	-

2.2.8.12 CRBBuildCommand

Frame transfer mode support needs to be updated, if required.

Dependencies	Calls	GetDuration, GetTime, SendMessage, SendMessageArg, TableCheck
	Called by	CRBAcquirePrepare, CRBSetCCDHeaterPower, PowerCRBOn, ShutterEnableNACBrake, TCAcquireDualImage, TCAcquireImage, TCFMonitorObservation, TCMonitorObservation
	Synchronization	-
Error handling	Input checking	Yes.
	Return codes	FALSE on error
	Timeouts	-
	Aborts	On error with FALSE
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.8.13 CRBSetCCDHeaterPower

Dependencies	Calls	CRBBuildCommand, CRBImage
	Called by	-
	Synchronization	-
Error handling	Input checking	No, parameters are propagated to called functions
	Return codes	Ignores return values of called functions, **returns always TRUE**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	**camera is used as array index, relies on enumeration check for validity.**
	Blocking	-

2.2.8.14 CRBCheckAcquirePreCon

Is call to CRBSync really necessary if IsUnitOn is FALSE? IsUnitOn calls CRBSync by itself already.

Dependencies	Calls	CheckShutterTemperature, CRBSync, IsMMBOn, IsSwitchOn, IsUnitOn, SendMessage, SendMessageArg
	Called by	CRBAcquirePrepare
	Synchronization	-
Error handling	Input checking	No
	Return codes	FALSE on error
	Timeouts	-
	Aborts	On error, with message and FALSE

Potential flaws	Loops	Max. 5 times
	Memory access	**camera is used as array index, relies on enumeration check for validity.**
	Blocking	-

2.2.8.15 CRBAllocAcquireMem

Dependencies	Calls	IMAlloc, MMBAlloc, MMBInitHandleMemory, MMBLock, SendMessage, ToMMB
	Called by	CRBAcquirePrepare
	Synchronization	-
Error handling	Input checking	No, all other than eMemMMB are handled as eMemIM
	Return codes	FALSE on error
	Timeouts	-
	Aborts	On error, with message and FALSE
Potential flaws	Loops	In simulator mode only, fixed to 63 repetitions
	Memory access	-
	Blocking	MMBLock is canceled by a call to MMBFree, if e.g. shutter error recovery fails (CRBAcquirePrepare returns FALSE, therefore CRBAcquireFinalize will not be called to call the MMBUnlock function)

2.2.8.16 CRBAcquirePrepare

Dependencies	Calls	CRBAllocAcquireMem, CRBBuildCommand, CRBCheckAcquirePreCon, CRBSync, GetSize, GetTime, IMFree, InitImageControlHeader, MMBFree, SendMessage, SendMessageArg, ShutterDelayForCharge, ShutterErrorRecovery, ShutterPeform-Homepush, ShutterSaveFiringTimestamp, TableCheck
	Called by	CRBAcquireDualImage, CRBAcquireImageToIM, CRBAcquireImageToMMB
	Synchronization	-
Error handling	Input checking	No, but parameter “camera” is checked by subfunction before it is used as array index
	Return codes	FALSE on error; ignores return value of ShutterPeformHomepush
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Virtually endless loop possible if ParameterTable.shutter.shutter_params[camera].uPreHomepushCount is very large
	Memory access	-
	Blocking	-

2.2.8.17 CRBAcquireFinalize

Dependencies	Calls	AcquireHKSnapshot, AcquireResAccess, CRBSync, FromMMB, GetTime, MMBUnlock, ReleaseResAccess, SendMessageArg, ShutterConfigSHE, ShutterError-Recovery, ShutterPeformHomepush, ToMMB
	Called by	CRBAcquireDualImage, CRBAcquireImageToIM, and CRBAcquireImageToMMB
	Synchronization	Acquires and releases RES_LAST_IMAGE

Error handling	Input checking	no
	Return codes	**Always TRUE** **ignores return codes of AcquireHKSnapshot, ShutterConfigSHE and ShutterPeformHomepush**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	**Parameter “camera” is used as array index without checking, relying on enumeration type check.**
	Blocking	-

2.2.8.18 CRBAcquireImageToIM

Dependencies	Calls	AcquireResAccess, CalcTimeDiff, CRBAcquireFinalize, CRBAcquirePrepare, CRBToRAM, GetDuration, GetTime, IMFree, ReleaseResAccess, SendMessage, SendMessageArg, ShutterConfigSHE
	Called by	AcquireImageLowPower, AcquireImageStored, AutoExpose, CRBAcquireTestImageToIM, and CRBAcquireWithoutShutterToIM
	Synchronization	Aquires and releases RES_NAC_CRB/RES_WAC_CRB
Error handling	Input checking	Yes, **but first use of parameter “camera” as array index is before the check.**
	Return codes	Ignores return value of ShutterConfigSHE
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.8.19 CRBAcquireImageToMMB

Dependencies	Calls	AcquireResAccess, CalcTimeDiff, CRBAcquireFinalize, CRBAcquirePrepare, CRBToMMB, GetDuration, GetTime, MMBFree, ReleaseResAccess, SendMessage, SendMessageArg, ShutterConfigSHE
	Called by	AcquireImageStored
	Synchronization	Acquires and releases RES_NAC_CRB/RES_WAC_CRB and RES_MMB_WRITE_DIB0/RES_MMB_WRITE_DIB1
Error handling	Input checking	Yes, **but first use of parameter “camera” as array index is before the check.**
	Return codes	Ignores return value of ShutterConfigSHE
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.8.20 CRBAcquireDualImage

Dependencies	Calls	AcquireResAccess, CRBAcquireFinalize, CRBAcquirePrepare, CRBDual, GetTime, IMFree, MMBFree, ReleaseResAccess, SendMessage, SendMessageArg, Shutter-ConfigSHE
	Called by	AcquireDualImage
	Synchronization	Acquires and releases RES_NAC_CRB and RES_WAC_CRB as well as RES_MMB_WRITE_DIB0 and/or RES_MMB_WRITE_DIB1
Error handling	Input checking	No, relies on enumeration type check.
	Return codes	FALSE on error
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.9 ParameterTable.ocl

2.2.9.1 TableNVRAMAddress

Dependencies	Calls	-
	Called by	TableWriteLifetimeHeader, TableReadLifetimeHeader, TableGetHWSpecifiHeader, TableInitHWSpecifiHeader, TableInit, TableExit
	Synchronization	-
Error handling	Input checking	Yes.
	Return codes	NULL on error
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.9.2 WriteToNVRAM

Dependencies	Calls	_memcmp, FromNVRAM, IMFree, IMAlloc, ToNVRAM, SendMessageArg
	Called by	TableExit, TableWriteLifetimeHeader
	Synchronization	-
Error handling	Input checking	No.
	Return codes	FALSE on failing IMAlloc Ignores return value of IMFree (FALSE is extremely improbable, since this handle has been allocated in the same function a few lines before)
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Possibly endless loop if provided parameter “segment_size” is 0 (but both callers use constant value 64)

	Memory access	-
	Blocking	-

2.2.9.3 TableGetHWSpecifiHeader

Dependencies	Calls	CalcChecksum, FromNVRAM, SendMessage, TableNVRAMAddress
	Called by	IsMainDPU, TableInitHWSpecificHeader, TableResetRuntimeParams
	Synchronization	-
Error handling	Input checking	-
	Return codes	FALSE if checksum is not OK **Return value ignored by TableResetRuntimeParams**
	Timeouts	-
	Aborts	With FALSE on failing TableNVRAMAddress (which can never happen...)
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.9.4 TableInitHWSpecifiHeader

Dependencies	Calls	CalcChecksum, TableGetHWSpecifiHeader, TableNVRAMAddress, ToNVRAM
	Called by	-
	Synchronization	-
Error handling	Input checking	no.
	Return codes	Result of TableGetHWSpecifiHeader
	Timeouts	-
	Aborts	With FALSE on failing TableNVRAMAddress (which can never happen...)
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.9.5 TableReadLifetimeHeader

Dependencies	Calls	CalcChecksum, FromNVRAM, SendMessage, TableNVRAMAccess
	Called by	SendLifetimeCounters
	Synchronization	-
Error handling	Input checking	-
	Return codes	FALSE, if default table is returned instead of loaded one. **Return value is ignored by SendLifetimeCounters**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.9.6 TableWriteLifetimeHeader

Dependencies	Calls	CalcChecksum, TableNVRAMAccess, WriteToNVRAM
	Called by	SendLifetimeCounters
	Synchronization	-
Error handling	Input checking	Yes, returning FALSE on outdated table version
	Return codes	**Return value is ignored by SendLifetimeCounters**
	Timeouts	-
	Aborts	With FALSE if outdated
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.9.7 TableResetSegment

Dependencies	Calls	IsMainDPU, TableResetRuntimeParams, MMBManagerReset, IMManagerInit, ProcessHandlerInit
	Called by	TableReset
	Synchronization	-
Error handling	Input checking	-
	Return codes	Ignoring return code of ProcessHandlerInit and IMManagerInit (both return always TRUE)
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.9.8 TablePatchValue

Dependencies	Calls	SendMessage, SendMessageArg
	Called by	-
	Synchronization	
Error handling	Input checking	Yes, default branch in switch for illegal input, generating TM message
	Return codes	-
	Timeouts	-
	Aborts	On outdated table version
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.9.9 TableReset

Dependencies	Calls	SendMessage, TableResetSegment

	Called by	TableInit
	Synchronization	-
Error handling	Input checking	-
	Return codes	-
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.9.10 TableCheck

Dependencies	Calls	TableInit, SendMessage
	Called by	A lot of functions
	Synchronization	-
Error handling	Input checking	-
	Return codes	FALSE on error, with TM message. Ignores return value of TableInit (which is always TRUE)
	Timeouts	-
	Aborts	Executes auto reset of table on error
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.9.11 TableCheckQuiet

Dependencies	Calls	-
	Called by	SendImageHandlingMessage, SendArray
	Synchronization	-
Error handling	Input checking	-
	Return codes	FALSE on error
	Timeouts	-
	Aborts	Does not auto reset the table, like TableCheck (see 2.2.9.10 above) does
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.9.12 TableResetRuntimeParams

Dependencies	Calls	GetTime, SendMessage, TableGetHWSpecifiHeader
	Called by	TableInit
	Synchronization	-

Error handling	Input checking	-
	Return codes	-
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Fix number of repetitions (ID_COUNT == 82, which is also the number of array elements)
	Memory access	-
	Blocking	-

2.2.9.13 TableInit

Dependencies	Calls	CalcChecksum, FromNVRAM, TableReset, TableResetRuntimeParams, TableSendSegment
	Called by	OsirisLibInit, TableCheck
	Synchronization	-
Error handling	Input checking	-
	Return codes	**Always TRUE,** but indicates possible errors in NVRAM copy via TM message Return value ignored by TableCheck
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Fix number of repetitions (ID_COUNT == 82, which is also the number of array elements)
	Memory access	-
	Blocking	-

2.2.9.14 TableExit

Dependencies	Calls	SendLifetimeCounters, WriteToNVRAM, _memcmp, TableCheck, TableSendSegment, SendMessage, TableNVRAMAddress, FromNVRAM
	Called by	OsirisLibExit, OsirisRequestShutdown
	Synchronization	-
Error handling	Input checking	-
	Return codes	**Always TRUE, even if saving failed (but at least TM error message is generated)**
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.9.15 TableSendSegment

Dependencies	Calls	IMFree, IMAlloc, SendArray, AcquireResAccess, ReleaseResAccess
	Called by	SendParameterTable, ShutterTuneTravel, ShutterTuneImpact, TableInit, TableExit

	Synchronization	With RES_MMB_MAN or RES_IM_MAN
Error handling	Input checking	ok
	Return codes	-
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Recursive call
	Memory access	-
	Blocking	-

2.2.9.16 SendParameterTable

Dependencies	Calls	TableSendSegment
	Called by	-
	Synchronization	-
Error handling	Input checking	Input directly forwarded to called function
	Return codes	-
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.2.9.17 SendLifetimeCounters

Dependencies	Calls	TableWriteLifetimeHeader, TableReadLifetimeHeader, SendArray
	Called by	TableExit
	Synchronization	-
Error handling	Input checking	-
	Return codes	-
	Timeouts	-
	Aborts	-
Potential flaws	Loops	Number of repetitions limited to table size
	Memory access	-
	Blocking	-

2.2.10 MCBHandler.ocl

The checked version is a snapshot of 20090515.

MCBReadHK is not always surrounded by resource acquisition in this file.

2.2.10.1 CheckFWMTemperature

OK

2.2.10.2 CheckFDMTemperature

OK

2.2.10.3 SelectMCB

OK

2.2.10.4 MCBGetControler

Might hang if resource is blocked (permanently) by another task just after the query. This is very improbable, but maybe waiting for this resource with timeout could avoid this.

2.2.10.5 MCBReleaseControler

OK

2.2.10.6 MCBHardwareDirection

Error code is identical to valid return values.

TCMCBMoveMechanism does not check validity of logical_direction which is forwarded up to MCBHardwareDirection

2.2.10.7 WriteMCBRegister

No default branch to handle illegal input parameters (all callers actually use valid input data in the current version)

2.2.10.8 MCBPerformZeroStep

OK

2.2.10.9 MCBMoveMechanism

Error code of MCBHardwareDirection is not checked for error code

2.2.10.10 MCBMoveMechanismDual

Error code of MCBHardwareDirection is not checked for error code

RuntimeTable update in error case – should this be only if MCBMove2 is called?

2.2.10.11 MCBUpdateFWMPosition

Return value is not checked by MCBMoveFilterWheelLow and MCBInitFilterWheel

2.2.10.12 MCBInitFilterWheel

Loop over constant number of repititions – number should be a #define or similar for better readability.

No call to MCBZeroStepMode(controler, ZEROSTEP_LOWPOWER); in return-branch in for-loop

2.2.10.13 MCBMoveFilterWheelLow

'return FALSE' is missing for filterwheel #2 unknown-position-branch

2.2.10.14 MCBMoveFilterWheel

Keep-keep-handling after temperature check TBC

2.2.10.15 MCBGetFDMRegion

OK.

2.2.10.16 MCBDoorGotoSwitch

Might be obsolete?

2.2.10.17 MCBOpenDoor

OK

2.2.10.18 MCBCloseDoor

OK, if REGION_ROTATING is really the desired end position

2.2.10.19 MCBLockDoor

Might be obsolete?

2.2.11 HKCheck.ocl

Main review done by Richard Moissl. Checked files are from the development branch 20090515.

2.2.11.1 HKGenericLimitAction

UNIT_DPU not handled.

2.2.11.2 HKCheckForRequiredAction

OK.

2.2.11.3 HKLimitCheckUnit

Return value is not checked by HKLimitCheck.

2.2.12 HKHandler.ocl

Main check done by Richard Moissl. Reviewed files are from development branch 20090515.

2.2.12.1 HKGetEncoder

OK.

2.2.12.2 HKGetPhase

OK.

2.2.12.3 HKEncoderToPos

Returns 0 and FWM_UNKNOWN (= 0), might cause problems if FWM_UNKNOWN is changed in later development.

2.2.12.4 HKGetPosition

OK.

2.2.12.5 HKSensorADCToPhysical

OK.

2.2.12.6 HKSensorPhysicalToADCByte

OK.

2.2.12.7 ADCValueByID_PCM

OK.

2.2.12.8 ADCValueByID_MCB

OK.

2.2.12.9 ADCValueByID_NAC

OK.

2.2.12.10 ADCValueByID_WAC

OK.

2.2.12.11 HKGetCalibratedPCM

OK.

2.2.12.12 HKGetCalibratedMCB

OK.

2.2.12.13 HKGetCalibratedNAC

OK.

2.2.12.14 HKGetCalibratedWAC

OK.

2.2.12.15 HKGetSensor

OK.

2.2.13 TMHandler.ocl

The checked UDPs are taken from the development branch of 20090515.

2.2.13.1 SendImageHandlingMessage

Parameter “queue” is not checked for validity ($< 2^{16}$), which might overwrite the event ID.

2.2.13.2 SendEvent

Relies on enumeration type check for event ID.

2.2.13.3 SendMessage

OK

2.2.13.4 SendMessageArg

OK

2.2.13.5 SendMessageArgF

OK

2.2.13.6 IsScienceLinkOk

OK

2.2.13.7 IsPlainFileLinkOk

OK

2.2.13.8 RequestRTUTransfer

Might block for 20 seconds (lead in comment seems to be outdated) if specified block size is too large.

2.2.13.9 SendScienceDataFromIM

OK, but declaration and assignment “block_size = 1024” could be moved outside the loop (performance).

2.2.13.10 SendMemoryDump

OK. Acquires and releases RES_SCIENCE_LINK.

2.2.13.11 SendArrayViaIF

Variable “size” not used except for initialization of “remain”.

Acquires and releases RES_SCIENCE_LINK.

RTU blocks might be lost without notification if RequestRTUTransfer fails.

2.2.13.12 SendArray

OK

2.2.13.13 SendImageFromIMToPlainFile

Acquires and releases RES_SCIENCE_LINK.

Comment says “lock the memory block”, but the called function IMAddressOf does not lock anything. Might memory be changed or freed meanwhile?

2.2.13.14 SendPlainfileClose

OK

2.2.13.15 IMMemoryDump

Returns always TRUE.

2.2.13.16 MMBMemoryDump

Acquires and releases RES_SCIENCE_LINK.

Uses undefined value of “res” for resource acquisition. “res” is changed afterwards and then released. That means a random resource might remain locked.

Acquires indirectly and releases RES_MMB_READ_DIB0, RES_MMB_READ_DIB1, RES_MMB_WRITE_DIB0 or RES_MMB_WRITE_DIB1.

2.2.13.17 NVRAMMemoryDump

Acquires and releases RES_SCIENCE_LINK.

2.2.13.18 QueueToBit

OK

2.2.13.19 IsDownlinkEnabledFromQueue

OK

2.2.13.20 IsStorageEnabledInQueue

OK

2.2.13.21 QueueImage

OK

2.2.13.22 PackImageHeader

Contains a known bug which is kept for the moment for compatibility reasons (see comments inside the function). The offset of the last element is 1 long too low.

Maybe the header array should use a structure or union data type consisting of elements of dedicated subtypes to avoid this type of errors.

2.2.13.23 PackSegmentHeader

OK

2.2.13.24 GetHKPacketData

No default branch for packet_sid.

Acquires and releases RES_NAC_CRB, RES_WAC_CRB, RES_PCM or RES_MCB.

Does not fill packet buffer if packet buffer size is smaller than snapshot requires, but return value is always the required size ("min_size").

2.2.13.25 SendUDPHKPacket

Checksum calculation is just summing up – should this replaced by more sophisticated methods?

2.2.13.26 UDPLevelHKRate

No default branch, relies on enumeration type check.

2.2.14 TC.ocl

The checked UDPs are taken from the development branch of 20090515.

2.2.14.1 load_shutter

Empty function, seems to be unused.

2.2.14.2 TCNACOn / TCWACOn

Ignores return values of both called functions (no abort), no message about unsuccessful call is returned (is generated in called function already).

2.2.14.3 TCNACOff / TCWACOff

Ignores return values of both called functions (no abort), no message about unsuccessful call is returned (is generated in called function already).

2.2.14.4 TCSelectMainConfig / TCSelectRedConfig

Ignores return values of all called functions.

2.2.14.5 TCMainMCB / TCRedMCB

Ignore return value of called function.

2.2.14.6 TCMainPCM / TCRedPCM

Ignore return value of called function.

2.2.14.7 TCStartHeaters

Ignores return value of called function, which means a possibly invalid table would not be reported to ground at this time.

2.2.14.8 TCRequestDPU_ID

OK.

2.2.14.9 TCSendStoredImage

OK.

2.2.14.10 TCAcquireImage

Relies on enumeration type check, uses parameter "camera" as array index.

2.2.14.11 TCAcquireDualImage

OK.

```
sleep(time_nac > time_wac ? time_nac : time_wac);
```

could also be written as

```
sleep(time_nac >? time_wac);
```

2.2.14.12 TCMCBMoveMechanism

OK, single-step input is recognized and reported to ground.

2.2.14.13 TCCompressStoredImage

OK.

2.2.14.14 TCInitPlainfileRead

This function seems not to be protected against (improbable but possible) concurrent execution (handle might be freed or altered after MMBNamedAlloc, before MMBLock is called), result of MMBLock is not checked for failure code.
Return value of MMBFree is not checked – freeing might fail if concurrent plainfile access is active.

"size" alignment could use bit-arithmetic instead of integer arithmetic (but this is not necessary, and this is no time critical section):

```
size = (size + 63) & ~64;
```

instead of

```
if((size % 64) != 0) size = (size/64)*64 + 64;
```

2.2.14.15 TCPlainfileRecovery

Return value of MMBLock is not checked – locking might fail if handle is altered/locked by concurrent process.
Size of handle might also change between call to MMBGetHandleSize and MMBLock.

2.2.14.16 TCQueueConfig

Result of DeleteImagesFromQueue and MoveImagesToQueue are ignored.

2.2.14.17 TCParseMonitorInput

Function is marked "ToDo".

2.2.14.18 TCMonitorObservation

Function is marked "ToDo" and seems to be incompatible to the calling function TCFExecute (no arguments provided).

Return values of ShutterTuneMotion and CRBBuildCommand ignored.

2.2.14.19 TCShutterConfig

OK.

2.2.15 UserLib.ocl

The checked UDPs are taken from the development branch of 20090515.

2.2.15.1 AutoExpose

"camera" is not checked and used as array index.

Ignores return value of GetImageControlHeaderFromIM – but it cannot fail here.

2.2.15.2 AcquireImageLowPower

Ignores return value of PowerCRBOff, PowerShutterOff and PowerCRBOn and PowerShutterOn

2.2.15.3 AcquireImageStored

First check “if (ok == TRUE &&...” could be simplified, because “ok” is always TRUE at this point. Call to AutoExpose could be moved into the next “if(ok == TRUE)” to avoid double checking “ok”

Ignoring return value of GetImageControlHeaderFromIM, SetImageControlHeaderToIM and IMFree, which cannot fail.

But ignoring also return value of StreamImageTo – this maybe might return FALSE.

2.2.15.4 AcquireDark

OK.

2.2.15.5 AcquireBias

OK.

2.2.15.6 AcquireDualImage

Since AutoExpose does not change the ok flag, both calls to this function could be put into only one “if (ok == TRUE)” together with taking the image.

Ignoring return value of GetImageControlHeaderFromIM, SetImageControlHeaderToIM and IMFree, which cannot fail.

Ignores the return values of GetImageControlHeaderFromMMB, SetImageControlHeaderFromMMB and QueueImage, which might fail e.g. if MMB is switched off.

2.2.15.7 DeleteImagesFromQueue

Ignores return values of MMBGetQueue and GetImageControlHeaderFromMMB, which could fail e.g. if the MMB is switched off.

Returns random BOOL (“return” missing)

2.2.15.8 MoveImagesToQueue

Ignores return values of MMBGetQueue, MMBSetQueue, MMBSetFlag and GetImageControlHeaderFromMMB, which could fail e.g. if the MMB is switched off.

Returns random BOOL (“return” missing)

2.2.15.9 AcquireImage

OK.

2.2.16 MathUtil.ocl

The checked UDPs are taken from the development branch of 20090515.

2.2.16.1 BesselIFilter

Not implemented yet

2.2.16.2 RealizeLine

OK.

2.2.16.3 GaussRandomNumber

OK.

2.2.16.4 TestRandomAbove

OK.

2.2.16.5 GaussRandomWalk

OK.

2.2.17 mem_util.ocl

Files from the development branch 20090515 were reviewed.

2.2.17.1 MMBBlockToIM

IM is not locked against concurrent execution.

2.2.17.2 IMBlockToMMB

Not checking return code of MMBLock

Comment says “lock the source handle”, but the IM handle is not locked. There is no locking against concurrent execution of this function which might cause problems.

2.2.18 misc.ocl

Files from the development branch 20090515 were reviewed.

2.2.18.1 CalcTimeDiff

Conversion to “double” seems to be useless (or even worse), because “double” is realized as 32-bits floating point value. That means that there are no bits gained for higher precision by this cast, but it might cause loss of lower bits.

This function is declared “safe”, but it is not called by “critical” or other “safe” functions.

2.2.18.2 CalcChecksum

No default-branch with error message (no error code possible).

2.2.18.3 PCMGet28V_Current

OK.

2.2.18.4 MCBConvertId

OK.

2.2.18.5 IsMainDPU

OK.

2.2.19 OsirisInit.ocl

Reviewed files are from the development branch 20090515.

2.2.19.1 OsirisLibInit

OK.

2.2.19.2 OsirisLibExit

Could be coded more efficiently...

2.2.20 Obsolete.ocl

Reviewed files are from the development branch 20090515. None of the functions in this file is called.

2.2.20.1 SendImageFromIM

OK.

2.2.20.2 SendImageFromMMB

OK.

2.2.20.3 CRBAcquireWithoutShutterToIM

OK.

2.2.20.4 CRBAcquireTestImageToIM

OK.

2.2.21 PCMHandler.ocl

Main check done by Irene Büttner. Checked files are from the development branch 20090515.

2.2.21.1 IsUnitOn

Outdated information might be used if resource is not available.

2.2.21.2 IsSwitchOn

OK.

2.2.21.3 PowerSwitchOnOff

OK.

2.2.21.4 SelectPCM

OK.

2.2.21.5 PCMSelectS1

Sleep time could use a named constant instead of immediate value.

2.2.21.6 ConfigPCM

Returns always TRUE.

2.2.21.7 PCMOn

Using a hkInterface constant as eBool type parameter in the call to IFControl.

Sleep times could use named constants instead of immediate values.

2.2.21.8 PCMOff

Using a hkInterface constant as eBool type parameter in the call to IFControl.

Sleep times could use named constants instead of immediate values.

2.2.21.9 PowerMCBOn

Using a hkInterface constant as eBool type parameter in the call to IFControl.

Sleep times could use named constants instead of immediate values.

2.2.21.10 PowerMCBOff

Sleep time could use a named constant instead of immediate value.

2.2.21.11 PowerMCBMotorsOn

OK.

2.2.21.12 PowerMCBMotorsRedOn

OK.

2.2.21.13 PowerMCBMotorsOff

OK.

2.2.21.14 PowerCRBOn

OK.

2.2.21.15 PowerCRBOff

OK.

2.2.21.16 PowerShutterOn

OK.

2.2.21.17 PowerShutterOff

OK.

2.2.21.18 PowerCycleShutter

OK.

2.2.21.19 IsShutterPowerOn

OK.

2.2.21.20 CRBSetLampState

OK.

2.2.21.21 SelectHeaters

OK.

2.2.21.22 PowerHeaters

OK.

2.2.21.23 SetHeaterBasePower

Could use named constants for limits.

2.2.21.24 PCMLockHeaterAccess

Not checking for overrun (extremely improbable, since it would require 2^{32}-1 calls.

2.2.21.25 PCMUnlockHeaterAccess

OK.

2.2.21.26 PCMWaitForHeaterAccess

OK, but does not lock heater access immediately, so another process might catch the lock afterwards.

2.2.21.27 PCMSwitchMonitor

Monitor of failsafe switches is missing (as commented).

2.2.21.28 RequestOsirisShutdown

Ignores return values of PowerShutterOff, PowerCRBOff, PowerMCBMotorsOff, CRBSetLampState and TableExit .

2.2.22 ShutterCalibrate.ocl

Reviewed files are from the development branch 20090515.

2.2.22.1 CalibrateShutter

Ignores return values of SendDemonRequest, CRBSetUp, CRBSetLampState, ShutterConfigSHE, TCAcquireImage, CRBInitiateShutter, ShutterTuneMotion, ShutterTuneTravel, ShutterTuneImpact, ShutterTuneImpactVelOff, ShutterTuneFiltering, ShutterOptimizeMaster and RealizeShutterProfile.

2.2.23 ShutterOptimize.ocl

2.2.23.1 ShutterTuneAB

Returns TRUE on illegal workspace and does not send an error message – is this intended?

2.2.23.2 ShutterTuneFiltering

OK.

2.2.23.3 ShutterTuneMotion

OK.

2.2.23.4 ShutterTestParam

OK.

2.2.23.5 ShutterSearchFunctionalInterval

Endless “while”-loops if “sp.fMinStep” is 0 (input parameter sp.fMinStep is not checked).

2.2.23.6 ShutterTuneImpact

OK, but the first branch uses the local variable “ok” while the second branch returns directly – this could be harmonized.

2.2.23.7 ShutterTuneTravel

Second branch uses new local variable 'ok' – it could use the variable from the upper scope instead.

2.2.23.8 ShutterTuneImpactVelOff

Ignores return value of CRBInitiateShutter.

2.2.23.9 ShutterOptimizeMaster

OK.

2.2.23.10 ShutterAutoOptimize

OK.

2.2.24 Non-reviewed Files

OsiInit.ocl,

2.3 Demons

2.3.1 DownlinkManager.ocl

2.3.1.1 DLGetIF

Dependencies	Calls	DPUReadHK, PostDemonRequest, SendMessage
	Called by	DLHandleThumbnails, DLProcessQueue, DownlinkManagerDemon
	Synchronization	-

Error handling	Input checking	-
	Return codes	dlNone on error, DemonRequest for pause (if flagged)
	Timeouts	-
	Aborts	On error, with error code
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.3.1.2 DLSafeMMBFree

Dependencies	Calls	MMBFree, AcquireResAccess, ReleaseResAccess
	Called by	DLProcessQueue
	Synchronization	Acquiring and releasing RES_LAST_IMAGE. **At this level it would be possible that RuntimeTable.handle_last_image[] is set to the handle which will be freed after the check. In that case (which is probably never reached, since the handle will not be freed before it has been noted in this variable) the RuntimeTable.handle_last_image[] would point to an invalid handle.**
Error handling	Input checking	Not checking for ILLEGAL_HANDLE; worst case would be wasteful execution of ressouce-aquisition. The calling function won't call it with ILLEGAL_HANDLE input, anyway.
	Return codes	-
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	If this UDP is aborted from the outside, RES_LAST_IMAGE might left blocked.

2.3.1.3 DLFindImageToHandle

Dependencies	Calls	IsMMBOn, MMBFirstHandleOfType, MMBGetQueue, MMBIsFlagSet, MMBNextHandleOfType
	Called by	DLProcessQueue
	Synchronization	-
Error handling	Input checking	No, relies on validity of enumeration type.
	Return codes	ILLEGAL_HANDLE on error
	Timeouts	-
	Aborts	Returning ILLEGAL_HANDLE if MMB is not ready
Potential flaws	Loops	Possibly endless loop, if internal chain is corrupted (even then it is improbable)
	Memory access	-
	Blocking	-

2.3.1.4 DLHandleThumbnails

Dependencies	Calls	StreamImageTo, QueueImage, MMBSetQueue, MMBFree, MMBAlloc, MMBGetH-

		andleSize, SendMessageArg, DLGetIF, MMBGetQueue
	Called by	DLProcessQueue
	Synchronization	-
Error handling	Input checking	
	Return codes	-
	Timeouts	-
	Aborts	Generates messages if thumbnail could not be created for non-TM destination, **no message on illegal input handle or MMBGetQueue failing (which is extremely improbable)**
Potential flaws	Loops	-
	Memory access	Allocates MMB memory (frees it on error)
	Blocking	-

2.3.1.5 DLProcessQueue

Inefficient code (calculation of 'can_transmit' could be removed and condition used directly, also the thumbnail handling and normal processing could be done in only one “if(){}”-block

Dependencies	Calls	GetTime, CalcTimeDiff, IsDownlinkEnabledFromQueue, SendMessageArg, SendImageHandlingMessage, QueueImage, GetImageControlHeaderFromMMB, SetImageControlHeaderToMMB, StreamImageTo, MMBBlockToIM,MMBGetHandleSize, DLHandleThumbnails, MMBSetFlag, DLSafeMMBFree, MMBOptimizeMemBlockLocation, SendImageFromIMToPlainFile, IMFree, IsMMBOn, GetImageControlHeaderFromIM, DLGetIF
	Called by	DownlinkManagerDemon
	Synchronization	-
Error handling	Input checking	No, relies on enumeration type
	Return codes	FALSE on error (e.g. non-fitting interface, no block to process was found, no block in IM, no transmission) **Ignores return code of GetImageControlHeaderFromMMB and SetImageControlHeaderFromMMB (might occur if handle cannot be locked – improbable)** **Ignores return code of GetImageControlHeaderFromIM (can occur only if input handle points to NULL very improbable)** **Ignores return code of QueueImage (might occur if MMB block is not a valid image)** **Ignores return code of MMBSetFlag, IMFree (would indicate internal inconsitency in MMBManager)** **Ignores return code of MMBOptimizeMemBlockLocation (could indicate internal inconsistency of MMB manager)**
	Timeouts	-
	Aborts	Returning FALSE MMB is not ready
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.3.1.6 DownlinkManagerDemon

Dependencies	Calls	CheckForDemonRequest, MarkDemonRunning, SendMessageArg, DLProcessQueue, DLGetIF, IsMMBOn, SendMessage, DemonPing, MarkDemonPaused,

		MarkDemonResumed, MarkDemonTerminated
	Called by	-
	Synchronization	-
Error handling	Input checking	-
	Return codes	Ignoring return code of MarkDemonRunning (could indicate second demon running, which might have been started slightly after the first check – very improbable)
	Timeouts	-
	Aborts	On STOP request with TM message
Potential flaws	Loops	Intentional endless loop (unless flag is reset via demon request)
	Memory access	-
	Blocking	-

2.3.1.7 DownlinkManager

Dependencies	Calls	DownlinkManagerDemon (staring in a new virtual machine)
	Called by	-
	Synchronization	-
Error handling	Input checking	-
	Return codes	-
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.3.2 HKMonitor.ocl

Reviewed files are from the development branch 20090515.

2.3.2.1 Service19Monitor

On initial violation count exceeding only WAC filter wheel is moved – is this correct?

2.3.2.2 HKLimitCheck

Return value of HKLimitCheckUnit is not evaluated, but limit violation is already handled inside.

2.3.2.3 HKMonitorDemon

OK.

2.3.2.4 HKMonitor

OK.

2.3.3 SHMOptimize.ocl

2.3.3.1 IterateTVRandomWalk

Dependencies	Calls	GaussRandomWalk, TestRandomAbove, SendMessageArg

	Called by	ProcessSHMOpt
	Synchronization	-
Error handling	Input checking	No, relies on enumeration type checks
	Return codes	-
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.3.3.2 LoadSHMOptData

Dependencies	Calls	ReleaseResAccess, AcquireResAccess, ExtractPulsesFromImage
	Called by	SHMOptimizeDemon
	Synchronization	Acquiring and releasing RES_LAST_IMAGE
Error handling	Input checking	Relies on enumeration type check
	Return codes	FALSE if no image and/or if no pulse data were found
	Timeouts	Waiting without timeout
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.3.3.3 ProcessSHMOpt

Dependencies	Calls	SendArray, SendUDPHKPacket, ShutterSplitBladePulses, IterateTVRandomWalk ShutterTuneSingleBladeMotion, SendMessageArg, BuildShutterPowerProfile, ShutterConfigSHE, SendMessage
	Called by	SHMOptimizeDemon
	Synchronization	-
Error handling	Input checking	Relies on enumeration type check
	Return codes	FALSE on error, **ignored by SHMOptimizeDemon** Check of ShutterSplitBladePulses could be removed (return value is always TRUE) **ignores return value of ShutterConfigSHE (from CRBShutter)**
	Timeouts	-
	Aborts	With FALSE on error
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.3.3.4 SHMOptimizeDemon

Dependencies	Calls	CheckForDemonRequest, DemonPing, MarkDemonRunning, GetTime, Calc-

		TimeDiff, ProcessSHMOpt, SendMessageArg, IMFree, LoadSHMOptData, MarkDemonPaused, MarkDemonResumed, MarkDemonTerminated, ShutterInitWorkspace TableCheck, SendMessage
	Called by	-
	Synchronization	-
Error handling	Input checking	-
	Return codes	Ignores return value of MarkDemonRunning **no default branch handling illegal demon requests** Ignores return value of ProcessSHMOpt
	Timeouts	-
	Aborts	If parameter table is invalid, or if demon is already active and responding or if shutter workspace allocation failed with TM message
Potential flaws	Loops	Intentionally endless loop
	Memory access	-
	Blocking	-

2.3.3.5 SHMOptimize

Dependencies	Calls	Starts SHMOptimizeDemon
	Called by	-
	Synchronization	-
Error handling	Input checking	-
	Return codes	-
	Timeouts	-
	Aborts	-
Potential flaws	Loops	-
	Memory access	-
	Blocking	-

2.3.4 OCLEvent.ocl

Reviewed files are from the development branch 20090515.

2.3.4.1 OsirisToSafeMode

OK.

2.3.4.2 HandleService19

OK.

2.3.4.3 HandleSpuriousPCMEvent

Use enums/defines instead of absolute values

ePSEActionIgnore is not handled correctly (“do nothing”) in switch – reaches default branch which sends REQUEST_LCL_OFF_EVENT.

2.3.4.4 HandleUDPMgrAbort

Attention: Format of UDP abort message is under change!

Loops could use 'break' after VM is found (starting could use switch case).

Return value of 'start' is not checked. Demon restart may fail due to several reasons most probable: all interpreters busy)

2.3.4.5 OCLEventDemon

MsgGetW should be replaced by MsgGetWT to allow UDP manager reset to work properly (repeat while timed out).
Return value of “start("OsirisToSafeMode");” is not checked.
Successful restart should be checked somehow (might fail e.g. due to low memory or all token interpreters busy).

2.3.4.6 OCLEvent

OK

2.3.5 ThermalControl.ocl

Reviewed files are from the development branch 20090515.

2.3.5.1 InitThermalControl

OK.

2.3.5.2 CalcBalance

Could use “tmp” instead of re-calculating the sum of cycles.

2.3.5.3 PowerHeatersT

Return value of PCMHeaters is ignored.

Cycles are set to the heater table first and then the upper limit is checked – this could be reordered.

2.3.5.4 CalcBalanceDuty

OK.

2.3.5.5 ThermalControlDemon

During startup it is checked that MCB is switched on – if not, MCB_OFF is reported. If this happens too often inside the loop, DIB_CRASH is reported (but the loop is continued with an error message every repetition, anyway).

Why depends this demon on the activity of DEMON_OCL_EVENT?

2.3.5.6 ThermalControl

OK.

2.3.6 Unchecked files

AnnealControl.ocl

2.4 SwitchOn

Reviewed files are from the development branch 20090515.

2.4.1 AutoStart.ocl

2.4.1.1 AutoStart

OK.

2.4.2 SwitchMCBOn.ocl

2.4.2.1 SwitchMCBOn

In simulator mode the result of PowerMCBOn is overwritten – if it does not matter for simulator mode, it could be moved into the else branch.
Does not check for successful start command

2.4.3 SwitchOff.ocl

2.4.3.1 SwitchOff

OK.

2.4.4 SwitchPCMOn.ocl

2.4.4.1 SwitchPCMOnOK.

2.5 Tests

The test files (test_fwm.ocl, test_heaters.ocl and test_sync.ocl) have not been reviewed, because they are not part of the on-board software.

2.6 Summary

2.6.1 Found Errors

- CRBHandler.ocl
  - CRBSetAcquireMode: Does not unlock resource on illegal shutter mode.
- IMManager.ocl
  - IMReleaseHandlesOwnedByVM: Return value is missing
- MCBHandler.ocl
  - MCBMoveFilterWheelLow: 'return FALSE' is missing for filterwheel #2 unknown-position-branch
- mem_util.ocl
  - not checking return code of MMBLock
- MMBManager.ocl
  - mmb_ReadMemManagerQuiet does not handle FAT2 copy correctly in error case “FAT1 invalid”
  - MMBInitHandleMemory: does not unlock MMB if IMAlloc fails
  - MMBLock: returning 0 (FALSE) if MMB Manager is not OK (0xFFFFFFFF (NULL) would be correct)
  - SelectMMB is not implemented
- OCLEvent.ocl
  - HandleSpuriousPCMEvent: ePSEActionIgnore is not handled correctly (“do nothing”) in switch – reaches default branch which sends REQUEST_LCL_OFF_EVENT.
- TMHandler.ocl

- MMBMemoryDump: Uses undefined value of “res” for resource acquisition, which is not released.

2.6.2 Potentially Obscured Runtime-Errors

- CRBHandler.ocl
  - CRBSetAmplifier: return value ignored by callers
- CRBHandler.ocl
  - CRBAcquireFinalize: returns always TRUE
  - CRBSetCCDHeaterPower and
  - CRBSetWindow: returns TRUE even if one or more coordinate limits are violated
- DownlinkManager.ocl
  - DLHandleThumbnails: no message on illegal input handle or MMBGetQueue failing (which is extremely improbable)
  - DLProcessQueue: ignores some return values
- ImageLib.ocl
  - AcquireHKSnapshot, GetImageControlHeaderFromIM, GetImageControlHeaderFromMMB, GetImageHeaderFromMMB, SetImageControlHeaderToMMB, SetImageHeaderToMMB: return value is ignored by some callers
- MCBHandler.ocl
  - MCBHardwareDirection: Error code is identical to valid return values
- MMBManager.ocl
  - mmb_WriteFAT: no notification on abort if MMB is switched off.
  - MMBSetFlag: “flag” is not checked.
- OCLEvent.ocl
  - OCLEventDemon: Return value of “start("OsirisToSafeMode");” is not checked. Successful restart should be checked somehow (might fail e.g. due to low memory or all token interpreters busy).
- ParameterTable.ocl
  - TableExit: return value is always TRUE, even if saving failed (but at least TM error message is generated)
  - TableGetHWSpecifiHeader, TableReadLifetimeHeader, TableWriteLifetimeHeader: return value ignored by callers
- SHMOptimize.ocl
  - ProcessSHMOpt: ignores return value of ShutterConfigSHE (from CRBShutter)
  - SHMOptimizeDemon: no default branch handling illegal demon requests
- ShutteErrorHandler.ocl
  - ShutterErrorRecovery: RuntimeTable.she_error_history is not modified for error type D.
  - ShutterFunctionalLimitCheck: Does not enter safemode if uMinExpTimeForMotionOpt < LONG_EXP_TIME_FOR_OPT, no notification about this.
- ShutterCalibrate.ocl
  - CalibrateShutter: Ignores return values of SendDemonRequest, CRBSetUp, CRBSetLampState, ShutterConfigSHE, TCAcquireImage, CRBInitiateShutter, ShutterTuneMotion, ShutterTuneTravel, ShutterTuneImpact, ShutterTuneImpactVelOff, ShutterTuneFiltering, ShutterOptimizeMaster and RealizeShutterProfile.
- ShutterHandler.ocl
  - CheckShutterTemperature, ShutterAcquireTestImage, ShutterBuildUnlockProfile, ShutterHomepushNAC,

ShutterHomepushWAC: ignore some return values.

- ShutterAcquireTestImage: no TM if NAC/WAC is not switched on, but other errors are signaled via TM.
- ShutterBuildUnlockProfile: Ignoring result of RealizeAccellerationProfile
- ShutterCalcRefProfileTiming: If bp.fDecelPowerCorrFactor is negative, it is set to 1 without further messaging.
- ShutterDelayForCharge: Not sleeping at all if sleep_time is >= 13000, but no message about it
- ShutterHomepushNAC, ShutterHomepushWAC: Ignores the return value of ShutterPeformHomepush
- ShutterPeformHomepush: return value ignored by callers

- ShutterOptimize.ocl
  - ShutterTuneAB: Returns TRUE on illegal workspace and does not send an error message.
- StreamedImageProcessing.ocl
  - CloseProcessingWorkspace, CloseStreamWriter, StreamImageTo, StreamIMImageTo: ignore some return values
  - InitSegmentHeader, StreamImageTo, WriteToStream: result is ignored by callers
  - StreamHandleSegment: No notification about failing InitSegmentHeader, ignores return value of WriteToStream and ImgBright
- SwitchMCBOn.ocl
  - SwitchMCBOn: Does not check for successful start
- TC.ocl
  - TCInitPlainfileRead: result of MMBLock is not checked for failure code. Return value of MMBFree is not checked – freeing might fail if concurrent plainfile access is active.
  - TCPlainfileRecovery: Return value of MMBLock is not checked
- ThermalControl.ocl
  - PowerHeatersT: Return value of PCMHeaters is ignored.
- TMHandler.ocl
  - GetHKPacketData: return value is always the required size (“min_size”), even if buffer is not filled.
  - SendArrayViaIF: RTU blocks might be lost without notification if RequestRTUTransfer fails.
- UserLib.ocl
  - DeleteImagesFromQueue, MoveImagesToQueue: return missing, random boolean value is returned

2.6.3 Potential Runtime Errors

- CRBHandler.ocl
  - CRBAcquireFinalize: ignores return codes of AcquireHKSnapshot, ShutterConfigSHE and ShutterPeform-Homepush, unchecked parameter “camera” is used as array index
  - CRBAcquireImageToIM, CRBAcquireImageToMMB: first use of parameter “camera” as array index is before the check
  - CRBBuildCommand: Frame transfer mode support needs to be updated, if required.
  - CRBSetCCDHeaterPower, CRBCheckAcquirePreCon: unchecked parameter “camera” is used as array index
- HKMonitor.ocl
  - Service19Monitor: On initial violation count exceeding only WAC filter wheel is moved – is this correct?
- mem_util.ocl
  - MMBBlockToIM: IM is not locked against concurrent execution.
- MMBManager.ocl
  - MMBOptimizeMemBlockLocation: Block is lost if IMBlockToMMB fails (might delayed freeing hMMB avoid this?)
- OCLEvent.ocl
  - HandleUDPMgrAbort: Attention: Format of UDP abort message is under change!
  - OCLEventDemon: MsgGetW might block UDP manager reset. Return value of “start("OsirisToSafeMode");” is not checked.
- ShutterErrorHandling.ocl
  - ShutterErrorRecovery: RuntimeTable.she_error_history is not modified for error type D.
- ShutterOptimize.ocl
  - ShutterSearchFunctionalInterval: Endless “while”-loops if “sp.fMinStep” is 0 (input parameter sp.fMinStep is not checked).
- StreamedImageProcessing.ocl
  - StreamIMImageTo: PackImageHeader's parameter “ih” contains unchecked “comp_type” and “comp_value”
- TC.ocl
  - TCInitPlainfileRead: not protected against concurrent execution
  - TCParseMonitorInput, TCMonitorObservation: marked “to do”
  - TCPlainfileRecovery: Size of handle might change between call to MMBGetHandleSize and MMBLock on concurrent access.
- TMHandler.ocl
  - PackImageHeader: Contains a known bug which is kept for the moment for compatibility reasons (see comments inside the function). The offset of the last element is 1 long too low.

2.6.4 Minor Annotations

Several functions rely on enumerations, which limit the valid range of variables. As the OCL compiler might be configured to ignore misuses of enumeration types, this is a potential risk. But the OSIRIS library has been compiled with all warnings activated and showed no enumeration warning, so relying on enumerations can be considered safe. For this reason, these 'unchecked input' annotations are not repeated in this summary.

- CRBHandler.ocl
  - CRBAcquireFinalize, CRBSetCCDHeaterPower, CRBSetShutterConfig: Return always TRUE
- DownlinkManager.ocl
  - DLProcessQueue: contains inefficient code
  - DLSafeMMBFree: RuntimeTable.handle_last_image[] is not fully protected
- DownlinkManager.ocl
  - DLSafeMMBFree: it would be possible that RuntimeTable.handle_last_image[] is set to the handle which will be freed after the check. In that case the RuntimeTable.handle_last_image[] would point to an invalid handle.
- IMManager.ocl
  - im_FindFreeBlock: Possible endless loop if nodes are circular chained (this is very improbable, major data corruption has to occur first)
  - im_InsertAfterMemList, im_InsertBeforeMemList, im_RemoveFromMemList: return value is always TRUE, the interface might be changed to void return value
  - IMAlloc: Not reverting the change of the free mem block's size (hRet) on abort after failing im_InsertBeforeMemList might cause IM manager inconsistency (called function cannot fail and returns always true, branch can be removed completely)
  - IMGetHandleFlags: Not checking for INVALID_HANDLE (currently defined 0xFFFFFFFF, which is actually > IM_MAX_BLOCK_COUNT, but it is checked separately in other functions)
- MathUtil.ocl
  - BessellFilter: not implemented
- MCBHandler.ocl
  - MCBDoorGotoSwitch, MCBLockDoor: might be obsolete
- mem_util.ocl
  - IMBlockToMMB: Comment says “lock the source handle”, but the IM handle is not locked. There is no locking against concurrent execution of this function which might cause problems.
- Misc.ocl
  - CalcChecksum: no checksumming on illegal checksumAlgorithm-ID, but no message about this.
- MMBManager.ocl
  - mmb_InsertAfterMemList, mmb_InsertBeforeMemList, mmb_RemoveFromMemList, MMBDefragment: return value is always TRUE, the interface might be changed to void return value
  - MMBAlloc: Not reverting the change of the free mem block's size (hRet) on abort after failing mmb_InsertBeforeMemList might cause MMB manager inconsistency (called function cannot fail and returns always true, branch can be removed completely)
  - MMBDefragment: “iteration” is incremented but never really used.
  - MMBIFRead does not need argument transfer_size. It is declared 'safe' but not called by other 'safe' or 'critical' UDPs
  - MMBIFWrite is declared 'safe' but not called by other 'safe' or 'critical' UDPs
  - MMBLargestBlock: Possibly endless loop if circular MMB node chain (this is very improbable, since it would occur only if MMB manager is corrupted)
  - MMBLock: First checks are executed before RES_MMB_MAN is locked – preconditions might change afterwards (at least the allocate-state)
  - MMBOptimizeMemBlockLocation: second message if IMBlockToMMB fails could be removed

- MMBSetHandleName: Variable “ok” can be removed (is not touched anyway) and constant TRUE used instead.

- OCLEvent.ocl
  - HandleUDPMgrAbort: Format of UDP abort message is under change!
  - OCLEventDemon: MsgGetW should be replaced by MsgGetWT to allow UDP manager reset to work properly (repeat while timed out).
- ParameterTable.ocl
  - TableInit: returns always TRUE
- PCMHandler.ocl
  - PCMOn, PCMOff, PowerMCBOn: Using a hkInterface constant as eBool type parameter in the call to IFControl.
  - PCMSwitchMonitor: Monitor of failsafe switches is missing (as commented).
- ProcessHandler.ocl
  - AcquireResAccess, ReleaseResAccess: “resource” might be out of range (used as array index)
  - AcquireResAccess: “ControlArray.res_acquire_flag[vm]” is write-accessed outside of the critical section which might cause concurrent access. The flag seems not to be used (read) at all and should be removed.
  - AcquireResAccess: Checking for successful resource locking is not necessary any more, since acquisition is locked reliably.
  - IsResourceOwnedByUDP: 'safe' declaration seems to be unnecessary.
  - ProcessHandlerInit: return value is always TRUE, the interface might be changed to void return value
- ShutterErrorHandling.ocl
  - ShutterErrorRecovery: No locking of RuntimeTable.shutter_state which could be (very improbably) set by CRBInitiateShutter concurrently.
- ShutterHandler.ocl
  - ShutterSplitBladePulses: Return value always TRUE
- StreamedImageProcessing.ocl
  - RunStreamProcessor: returns always TRUE, declaration of “BOOL ok = TRUE;” not necessary
  - StreamIMImageTo: PackImageHeader's parameter “ih” contains unchecked “comp_type” and “comp_value” (the rest of the structure can be considered correct, since it is read from image directly)
- SwitchMCBOn.ocl
  - In simulator mode the result of PowerMCBOn is overwritten – if it does not matter for simulator mode, it could be moved into the else branch.
- TC.ocl
  - load_shutter: empty, seems to be unused
- TC.ocl
  - TCMonitorObservation: Function is marked “ToDo”.
  - TCParseMonitorInput: Function is marked “ToDo”.
- TMHandler.ocl
  - PackImageHeader: Contains a known bug which is kept for the moment for compatibility reasons
- UserLib.ocl

- DeleteImagesFromQueue and
- MoveImagesToQueue: returns random BOOL value (which is currently ignored by calling function)

2.6.4.1 Questions

- MMBManager.ocl
  - StopMMB: sleeps for 1 second after setting mmb_is_initiated to FALSE, probably waiting for all possibly pending accesses to be finished. Would it be possible to acquire RES_MMB_MAN for safely syncing?
- ShutterHandler.ocl
  - ShutterCalcRefProfileTiming: If bp.fDecelPowerCorrFactor is negative, it is set to 1 without further messaging. Are values between 1.0 and 0.0 allowed? Answer: Yes.
  - ShutterConfigSHE: No shutter brake handling for WAC – is the hardware different? Answer: Yes.
  - ShutterForceClosed: Why is delta_vel alternating?
- ShutterErrorHandling.ocl
  - ShutterErrorRecovery:
    - Why is “as_exposure” checked two times in line 172ff, 260ff, 320ff and 384ff? Could be only one conditional jump.
    - Why shutter homepush after successful recovery of type A (line 239)?
    - Recovery does not handle changing error types during recovery (recovery might fail even if the original error has been recovered)
  - ShutterFunctionalLimitCheck: Does not enter safemode if uMinExpTimeForMotionOpt < LONG_EXP_TIME_FOR_OPT, no notification about this.
- IMManager.ocl
  - Why can't IM handles be locked like MMB?
- ProcessHandler.ocl
  - ReleaseResAccess: Resource might be freed from other machines with TM error message – is this intended?
- MCBHandler.ocl
  - MCBCloseDoor: is REGION_ROTATING really the desired end position?

2.7 File Coverage

The following table lists all files of the high-level software and shows the current check status (✔: fully checked, **O**: partially checked, ✘: unchecked):

Module	File	Check Status	Size (lines)	Annotation
OsirisLib	**CRBHandler**	✔	2352	Development branch 20090515
	HKCheck	✔	447	Development branch 20090515
	HKHandler	✔	696	Development branch 20090515
	ImageLib	✔	608	
	IMManager	✔	1041	
	MathUtil	✔	69	Development branch 20090515
	MCBHandler	✔	1842	Development branch 20090515
	mem_util	✔	190	Development branch 20090515

Module	File	Check Status	Size (lines)	Annotation
	misc	✔	171	Development branch 20090515
	MMBManager	✔	2926	
	Obsolete	✔	130	Development branch 20090515
	OsirisInit	✔	106	Development branch 20090515
	ParameterTable	✔	1943	
	PCMHandler	✔	1737	
	ProcessHandler	✔	738	Development branch 20090224
	ShutterCalibrate	✔	362	Development branch 20090515
	ShutterErrorHandling	✔	566	Development branch 20090224
	ShutterHandler	✔	2646	Development branch 20090224
	ShutterOptimize	✔	1072	Development branch 20090515
	StreamedImageProcessing	✔	1245	
	TC	✔	1325	Development branch 20090515
	TMHandler	✔	1446	Development branch 20090515
	UserLib	✔	1103	Development branch 20090515
demons	AnnealControl	✘	341	
	DownlinkManager	✔	553	
	HKMonitor	✔	380	Development branch 20090515
	OCLEvent	✔	419	Development branch 20090515
	SHMOptimize	✔	464	
	ThermalControl	✔	644	Development branch 20090515
SwitchOn	AutoStart	✔	44	Development branch 20090515
	SwitchMCBOn	✔	112	Development branch 20090515
	SwitchOff	✔	59	Development branch 20090515
	SwitchPCMOn	✔	125	Development branch 20090515

All files have also been checked for their correct usage of RTL functions (see chapter 3).

3 RTL Usage in UDPs

This chapter checks the high-level use of RTL functions in UDPs. It examines the utilization of return values and possibly error codes (e.g. timeouts) on base of the high-level software version 2.1 Patchlevel 2 Release 1 (date 2008-07-07) with the change requests of [AD2] in mind (that means that some calls to RTL functions, which are currently not returning any error code, will be checked for high-level error handling, too).

3.1 Osiris

3.1.1 GetTime

This function does not provide a return value.

3.1.2 SendData

Caller	Annotation
SendScienceDataFromIM	These functions do not pay attention for an return value of SendData (because this function has no return value yet)
MMBMemoryDump	
SendArrayViaIF	
SendUDPHKPacket	
SendImageHandlingMessage	
SendMessageArg	
SendMessage	
SendEvent	
SendPlainfileClose	

3.1.3 GetNextSeqCounter

This function seems to be unused.

3.1.4 TmFlush

This function seems to be unused.

3.1.5 TmGetFree

Caller	Annotation
RequestRTUTransfer	Return value is used.

3.2 OsiDrv

3.2.1 LinkStart3

Function seems to be unused.

3.2.2 LinkReset3

Function seems to be unused.

3.2.3 DIBInit

Function seems to be unused.

3.2.4 DIBConfig

Function seems to be unused.

3.2.5 DIBStatus

Function seems to be unused.

3.2.6 DIBPulse

Function seems to be unused.

3.2.7 DIBClock

Function seems to be unused.

3.2.8 ToNVRAM

Caller	Annotation
TableInitHWSpecificHeader	No check for error codes yet, because ToNVRAM does not return any error code yet.
WriteToNVRAM	

3.2.9 FromNVRAM

Caller	Annotation
NVRAMMemoryDump	No check for error codes yet, because FromNVRAM does not return any error code yet.
TableGetHWSpecificHeader	
TableInit	
WriteToNVRAM	
TableReadLifetimeHeader	
TableExit	

3.3 OsiAsm

3.3.1 Swap16

Function seems to be unused.

3.3.2 Swap32

Function seems to be unused.

3.3.3 Split16

Caller	Annotation
ExtractPulsesFromImage	No check for error codes, because FromNVRAM does not return any error code yet – error case will never occur, as long as input parameters are valid (which is ensured by the callers)
ShutterAcqureTestImage	
ShutterPerformHomepush	
RunStreamProcessor	

3.3.4 Join16

Function seems to be unused.

3.3.5 Join16S

Function seems to be unused.

3.3.6 Not32

Function seems to be unused.

3.3.7 ToPM

Function seems to be unused.

3.3.8 FromPM

Function seems to be unused.

3.4 OsiImage

3.4.1 Plausible

Caller	Annotation
CRBReorderImage	ok.

3.4.2 GetLines

Caller	Annotation
CalcCCDSectionCoords	ok
CRBReorderImage	ok

3.4.3 GetPixels

Caller	Annotation
CalcCCDSectionCoords	ok
CRBReorderImage	ok

3.4.4 GetSize

Caller	Annotation
CRBAcquirePrepare	ok
ShutterEnableNACBrake	**Not checking for 0 size (= error indicator)**
ShutterAcquireTestImage	**Not checking for 0 size (= error indicator)**
CRBReorderImage	**Not checking for 0 size (= error indicator)**

3.4.5 GetDuration

Caller	Annotation
TCAcquireDualImage	ok.
CRBAcquireImageToMMB	
CRBAcquireImageToIM	

CRBBuildCommand	
ShutterBuildTestCmd	
TCAcquireImage	
TCFMonitorObservation	
TCMonitorObservation	

3.4.6 GetShutter

Caller	Annotation
CRBReorderImage	ok.
ExtractPulsesFromImage	
ShutterAcquireTestImage	
ShutterPeformHomepush	

3.4.7 Reorder

Caller	Annotation
CRBReorderImage	ok

3.4.8 CRBToRAM

Caller	Annotation
ShutterAcquireTestImage	**Ignore return value**
ShutterEnableNACBrake	
CRBAcquireImageToIM	ok

3.4.9 CRBToMMB

Caller	Annotation
CRBAcquireImageToMMB	ok

3.4.10 CRBDual

Caller	Annotation
CRBAcquireDualImage	ok

3.4.11 _memcmp

Caller	Annotation
CRBInitiateShutter	ok
IsValidImage	
WriteToNVRAM	
TableExit	

3.5 OsiMMB

3.5.1 MMBSwitch

Caller	Annotation
StartMMB	**Ignore the return value**
StopMMB	

3.5.2 MMBConf

OK.

3.5.3 MMBStatus

Caller	Annotation
MMBPoweredOnSize	OK

3.5.4 ToMMB

Caller	Annotation
CRBAcquireFinalize	No return value yet to be checked.
CRBAllocAcquireMem	
MMBInitHandleMemory	
FlushStreamWriter	
SetImageHeaderToMMB	
IMBlockToMMB	
mmb_WriteFAT	

3.5.5 FromMMB

Caller	Annotation
CRBAcquireFinalize	No return value yet to be checked.
TCPlainfileRecovery	
GetImageControlHeader-FromMMB	
MMBBlockToIM	
GetImageHeaderFromMMB	
ExtractPulsesFromImage	
mmb_ReadMemManagerQuiet	
MMBMemoryDump	

3.5.6 FileToMMB

Caller	Annotation
TCInitPlainfileRead	**Ignores return value**

3.5.7 MMBOCLSize

Seems to be unused.

3.5.8 MMBToOCL

Seems to be unused.

3.5.9 MMBOCLCheck

Seems to be unused.

3.5.10 FileStatus

Caller	Annotation
TCPlainfileRecovery	OK.

3.6 OsiUnit

3.6.1 CRBSend

Seems to be unused.

3.6.2 CRBReset

OK.

3.6.3 CRBSync

Caller	Annotation
CRBAcquirePrepare	OK.
CRBAcquireFinalize	OK.
ShutterErrorRecovery	OK.
CRBInitiateShutter	OK.
CalibrateShutter	OK.
CRBCheckAcquirePreCon	**Ignores return value**
IsUnitOn	**Ignores return value**

3.6.4 CRBImage

Caller	Annotation
CRBSetCCDHeaterPower	**Ignore return value**
PowerCRBOn	

3.6.5 CRBExec

Seems to be unused.

3.6.6 CRBReadHK

Caller	Annotation
CheckShutterTemperature	**Ignore return value**
PowerCRBOff	

HKGetSensor	
HKLimitCheckUnit	

3.6.7 CRBHkSize

Caller	Annotation
GetHKPacketData	OK

3.6.8 CRBReadHk

Caller	Annotation
GetHKPacketData	**Ignores return value**

3.6.9 CRBShutter

Caller	Annotation
IsShutterOpen	OK
ShutterEnableNACBrake	**Ignores return value**
ShutterConfigSHE	OK

3.6.10 PCMSend

Seems to be unused.

3.6.11 PCMReset

OK.

3.6.12 PCMSwitch

Caller	Annotation
PCMSelectS1	**Ignore return value**
CRBSetLampState	
PowerSwitchOnOff	

3.6.13 PCMSwitch2

Caller	Annotation
PowerMCBOff	**Ignore return value**
PowerMCBOn	

3.6.14 PCMPeekRAM

Seems to be unused.

3.6.15 PCMPokeRAM

Seems to be unused.

3.6.16 PCMPeekIO

Seems to be unused.

3.6.17 PCMPokeIO

Seems to be unused.

3.6.18 PCMPeekSFR

Seems to be unused.

3.6.19 PCMPokeSFR

Seems to be unused.

3.6.20 PCMReadHK

Caller	Annotation
HKGetSensor	**Ignore return value**
PCMGet28V_Current	
IsSwitchOn	
HKLimitCheckUnit	
IsUnitOn	
SwitchPCMOn	
PCMSelectS1	
ConfigPCM	

3.6.21 PCMHkSize

Caller	Annotation
GetHKPacketData	OK.

3.6.22 PCMReadHk

Caller	Annotation
GetHKPacketData	**Ignore return value**

3.6.23 PCMReadUpperL

Seem to be unused.

3.6.24 PCMReadLowerL

Seem to be unused.

3.6.25 PCMReadAction

Seem to be unused.

3.6.26 PCMReadHeaters

Caller	Annotation
GetHKPacketData	**Ignores return value**
MCBInitFilterWheel	
PCMLockHeaterAccess	

3.6.27 PCMReadAnneal

Seems to be unused.

3.6.28 PCMPrimaryL

Caller	Annotation
ConfigPCM	**Ignores return value**

3.6.29 PCMUpperL

Caller	Annotation
ConfigPCM	**Ignores return value**

3.6.30 PCMLowerL

Caller	Annotation
ConfigPCM	**Ignores return value**

3.6.31 PCMAction

Caller	Annotation
ConfigPCM	**Ignores return value**

3.6.32 PCMHeaters

Caller	Annotation
MCBInitFilterWheel	**Ignore return value**
PCMLockHeaterAccess	
PCMUnlockHeaterAccess	
PowerHeaters	
PowerHeatersT	

3.6.33 PCMAnneal

Caller	Annotation
AnnealControl	**Ignore return value**
Service19Monitor	

3.6.34 PCMPowerDown

Caller	Annotation
PCMOff	**Ignore return value**
PCMOn	

3.6.35 MCBSend

Seems to be unused.

3.6.36 MCBReset

OK.

3.6.37 MCBGetRegister

Seems to be unused.

3.6.38 MCBSetRegister

Caller	Annotation
WriteMCBRegister	**Ignores return value**

3.6.39 MCBMove2

Caller	Annotation
MCBMoveMechanismDual	OK.

3.6.40 MCBReadHK

Caller	Annotation
CalcAnnealDuty	**Ignore return value**
InitAnnealControl	
CalcBalanceDuty	
InitThermalControl	
MCBMoveFilterWheel	
MCBMoveFilterWheelLow	
CheckFWMTemperature	
MCBInitFilterWheel	
MCBGetFDMRegion	
CheckFDMTemperature	
HKLimitCheckUnit	
IsUnitOn	
HKGetSensor	
SwitchMCBOn	

3.6.41 MCBHkSize

Caller	Annotation
GetHKPacketData	OK.

3.6.42 MCBReadHk

Caller	Annotation
GetHKPacketData	**Ignores return value**

3.6.43 MCBMove

Caller	Annotation
MCBInitFilterWheel	**Ignore return value**
MCBPerformZeroStep	

MCBMoveMechanism	OK

3.6.44 MCBPhase

Caller	Annotation
MCBMoveMechanism	**Ignores return value**
MCBMoveMechanismDual	
MCBPerformZeroStep	
MCBInitFilterWheel	

3.6.45 MCBZeroStepMode

Caller	Annotation
MCBPerformZeroStep	**Ignore return value**
MCBInitFilterWheel	

3.6.46 DPUReadHK

Caller	Annotation
AutoStart	OK (return value is void at the moment)
IsMMBOn	
DLGetIF	
IsPlainFileLinkOk	
SendScienceDataFromIM	
IsScienceLinkOk	
IsUnitOn	
PowerMCBOn	
PCMOn	
PCMGet28V_Current	

3.6.47 DPUHkSize

Caller	Annotation
GetHKPacketData	OK

3.6.48 DPUReadHk

Caller	Annotation
GetHKPacketData	OK (return value is void at the moment)

3.6.49 DPUReadPCMAlarm

Seems to be unused.

3.6.50 HKStatus

Caller	Annotation
PCMOff	OK

PCMOn	

3.6.51 HKControl

Caller	Annotation
OsirisLibInit	OK (return value is void at the moment)
PCMOff	
PCMOn	
UDPLevelHKRate	

3.6.52 IFControl

Caller	Annotation
PCMOff	OK (return value is void at the moment)
PowerMCBOn	
PCMOn	

3.6.53 TMControl

OK.

3.7 OsiVirt

3.7.1 MsgPutW

Caller	Annotation
SendOCLEvent	**Ignores return value**

3.7.2 MsgPutWT

Seems to be unused.

3.7.3 MsgGetW

Caller	Annotation
OCLEventDemon	**Ignores return value**

3.7.4 MsgGetWT

Seems to be unused.

3.8 OclMan

3.8.1 CritSecBegin

Caller	Annotation
AcquireResAccess	OK, return value is void
ReleaseResAccess	

3.8.2 CritSecEnd

Caller	Annotation
AcquireResAccess	OK, return value is void
ReleaseResAccess	

3.8.3 LoadPOP

Caller	Annotation
AutoStart	**Ignores return value**

3.9 OclInt

3.9.1 TopLevelStamp

Caller	Annotation
InitImageControlHeader	OK

3.10 OsiWave

3.10.1 Encode

Caller	Annotation
RunStreamProcessor	OK

3.10.2 Decode

Seems to be unused.

3.11 OsilLib

3.11.1 ImgAdd

Seems to be unused.

3.11.2 ImgSub

Seems to be unused.

3.11.3 ImgMin

Seems to be unused.

3.11.4 ImgMax

Seems to be unused.

3.11.5 ImgMult

Seems to be unused.

3.11.6 ImgSqrt

Caller	Annotation
RunStreamProcessor	OK (currently void return)

3.11.7 ImgMean

Seems to be unused.

3.11.8 ImgStdDev

Seems to be unused.

3.11.9 ImgHist

Caller	Annotation
CalculateHistogram	OK (currently void return)

3.11.10 ImgShift

Seems to be unused.

3.11.11 ImgCrop

Caller	Annotation
StreamProcessSection	**Ignores return value**

3.11.12 ImgConv

Caller	Annotation
RunStreamProcessor	**Ignores return value**

3.11.13 ImgBin

Caller	Annotation
RunStreamProcessor	**Ignores return value**

3.11.14 ImgBright

Caller	Annotation
StreamHandleSegment	**Ignores return value**

3.11.15 ImgRemoveBad

Caller	Annotation
StreamFilterBadPixels	OK

3.11.16 RtlFFT

Seems to be unused.

3.11.17 RtlInvFFT

Seems to be unused.

3.11.18 RtlBwFilter

Caller	Annotation
RealizeAccelerationProfile	OK

3.11.19 RtlBoxcarFilter

Caller	Annotation
RealizeAccelerationProfile	OK
ShutterBuildHomepushProfile	Ignores return value

3.11.20 RtlLinRegVector

Caller	Annotation
CalculateShutterStatFromMotion	OK

4 Synchronization

This chapter shows dependencies between functions, which are not caused by direct function calls but by synchronization tools.

4.1 Semaphores

Since on-board software version 8.10 beta X the OCL system supports the use of the operating system's semaphores for reliable mutual exclusion. Only one of the two provided semaphores is used by the high-level software – and only for very short periods. Therefore no problems are expected here.

Resource	Begin	End
S_OCL0	ReleaseResAccess, AcquireResAccess	ReleaseResAccess, AcquireResAccess
S_OCL1	-	-

4.2 Resources

Resources are implemented on high level. They are used to lock hard- or software resources against concurrent access. As long as a resource is blocked, other tasks will have to wait for the resource to be released. This might last for some (significant) time. The following table lists all implemented resources, the functions which actually acquire the resource, the top level functions which might use the resource indirectly and it lists the functions which release the resources as well as the top-level functions which might use the releasing function. The top-level functions correspond to separate tasks, so this list might show dependencies between tasks.

Resource	Acquired by		Released by	
	Function	Top-Level Caller	Function	Top-Level Caller
RES_FWM_NAC_GROUP	AcquireDualImage AcquireImageStored AcquireImageLowPower	TCAcquireDualImage, AcquireBias, AcquireDark, AcqureImage, NACAcquireQueueDemon, WACAcquireQueueDemon, CalibrateShutter, TCFMonitorObservation, TCFExecute	AcquireDualImage AcquireImageStored AcquireImageLowPower	TCAcquireDualImage, AcquireBias, AcquireDark, AcqureImage, NACAcquireQueueDemon, WACAcquireQueueDemon, CalibrateShutter, TCFMonitorObservation, TCFExecute
RES_FWM_WAC_GROUP	AcquireDualImage AcquireImageStored AcquireImageLowPower	TCAcquireDualImage, AcquireBias, AcquireDark, AcqureImage, NACAcquireQueueDemon, WACAcquireQueueDemon, CalibrateShutter, TCFMonitorObservation, TCFExecute	AcquireDualImage AcquireImageStored AcquireImageLowPower	TCAcquireDualImage, AcquireBias, AcquireDark, AcqureImage, NACAcquireQueueDemon, WACAcquireQueueDemon, CalibrateShutter, TCFMonitorObservation, TCFExecute
RES_ILLEGAL	-		-	
RES_IM_MAN	TableSendSegment IMReleaseHandlesOwnedByVM IMShrink IMFree IMAlloc	AcquireBias, AcquireDark, AcquireImage, AnnealControl, AutoStart, CalibrateShutter, CRBAcquireTestImageToIM, CRBAcquireWithoutShutterToIM, CRBRequestExtra, CRBResetShutter, CRBSetCCDHeaterPower, CRBSetShutterConfig, DownlinkManagerDemon, HKMonitorDemon, IMShrink, MMBDefragment, NACAcquireQueueDemon, OCLEventDemon, OsirisToSafeMode, SendImageFromIM, SendImageFromMMB, SendParamet-	TableSendSegment IMReleaseHandlesOwnedByVM IMShrink IMFree IMAlloc	AcquireBias, AcquireDark, AcquireImage, AnnealControl, AutoStart, CalibrateShutter, CRBAcquireTestImageToIM, CRBAcquireWithoutShutterToIM, CRBRequestExtra, CRBResetShutter, CRBSetCCDHeaterPower, CRBSetShutterConfig, DownlinkManagerDemon, HKMonitorDemon, IMShrink, MMBDefragment, NACAcquireQueueDemon, OCLEventDemon, OsirisToSafeMode, SendImageFromIM, SendImageFromMMB, SendParamet-

Resource	Acquired by		Released by	
	Function	Top-Level Caller	Function	Top-Level Caller
		erTable, SHMOptimizeDemon, ShutterAutoOptimize, ShutterHomepushNAC, ShutterHomepushWAC, SwitchMCBOn, SwitchOff, SwitchPCMOn, TCAcquireDualImage, TCCompressStoredImage, TCFExecute, TCFMonitorObservation, TCMainPCM, TCRedPCM, TCSelectMainConfig, TCSelectRedConfig, TCSendStoredImage, TCStartHeaters, ThermalControlDemon, WACAcquireQueueDemon		erTable, SHMOptimizeDemon, ShutterAutoOptimize, ShutterHomepushNAC, ShutterHomepushWAC, SwitchMCBOn, SwitchOff, SwitchPCMOn, TCAcquireDualImage, TCCompressStoredImage, TCFExecute, TCFMonitorObservation, TCMainPCM, TCRedPCM, TCSelectMainConfig, TCSelectRedConfig, TCSendStoredImage, TCStartHeaters, ThermalControlDemon, WACAcquireQueueDemon
RES_LAST_IMAGE	CRBAcquireFinalize DLSafeMMBFree LoadSHMOptData StreamIMImageTo		CRBAcquireFinalize DLSafeMMBFree LoadSHMOptData StreamIMImageTo	
RES_MCB	GetHKPacketData HKGetSensor HKLimitCheckUnit PowerMCBOff PowerMCBOn		GetHKPacketData HKGetSensor HKLimitCheckUnit PowerMCBOff PowerMCBOn	
RES_MCB_A	MCBGetControler MCBMoveMechanismDual	AcquireBias, AcquireImage, AcqureDark, CalibrateShutter, HKMonitorDemon, MCBLockDoor, NACAquireQueueDemon, OsirisToSafeMode, SwitchMCBOn, SwitchOff, TCAcquireDualImage, TCFExecute, TCFMonitorObservation, TCMCBMoveMechanism, WACAcquireQueueDemon	MCBGetControler MCBMoveMechanismDual	AcquireBias, AcquireImage, AcqureDark, CalibrateShutter, HKMonitorDemon, MCBLockDoor, NACAquireQueueDemon, OsirisToSafeMode, SwitchMCBOn, SwitchOff, TCAcquireDualImage, TCFExecute, TCFMonitorObservation, TCMCBMoveMechanism, WACAcquireQueueDemon
RES_MCB_B	MCBGetControler MCBMoveMechanismDual	AcquireBias, AcquireImage, AcqureDark, CalibrateShutter, HKMonitorDemon, MCBLockDoor, NACAquireQueueDemon, OsirisToSafeMode, SwitchMCBOn, SwitchOff, TCAcquireDualImage, TCFExecute, TCFMonitorObservation, TCMCBMoveMechanism, WACAcquireQueueDemon	MCBGetControler MCBMoveMechanismDual	AcquireBias, AcquireImage, AcqureDark, CalibrateShutter, HKMonitorDemon, MCBLockDoor, NACAquireQueueDemon, OsirisToSafeMode, SwitchMCBOn, SwitchOff, TCAcquireDualImage, TCFExecute, TCFMonitorObservation, TCMCBMoveMechanism, WACAcquireQueueDemon
RES_MMB_MAN	IsMMBConsistent MMBAlloc MMBDefragment MMBFree MMBLargestBlock MMBLock MMBManagerReset MMBNamedFind MMBSetFlag MMBSetHandleName MMBSetQueue MMBShrink		IsMMBConsistent MMBAlloc MMBDefragment MMBFree MMBLargestBlock MMBLock MMBManagerReset MMBNamedFind MMBSetFlag MMBSetHandleName MMBSetQueue MMBShrink	

Resource	Acquired by		Released by	
	Function	Top-Level Caller	Function	Top-Level Caller
	MMBSplit MMBStat MMBUnlock TableSendSegment		MMBSplit MMBStat MMBUnlock TableSendSegment	
RES_MMB_READ_DIB0	MMBIFRead	AcquireBias, AcquireDark, AcquireImage, AutoStart, CalibrateShutter, DownlinkManagerDemon, MMBDefragment, NACAcquireQueueDemon, SendImageFromIM, SendImageFromMMB, SHMOptimizeDemon, TCAcquireDualImage, TCCompressStoredImage, TCFExecute, TCFMonitorObservation, TCPlainfileRecovery, TCQueueConfig, TCSendStoredImage, WACAcquireQueueDemon	ExtractPulsesFromImage GetImageControlHeaderFromMMB GetImageHeaderFromMMB mmb_ReadMemManagerQuiet MMBBlockToIM MMBIFRead MMBMemoryDump MMBPoweredOnSize TCPlainfileRecovery	
RES_MMB_READ_DIB1	MMBIFRead	AcquireBias, AcquireDark, AcquireImage, AutoStart, CalibrateShutter, DownlinkManagerDemon, MMBDefragment, NACAcquireQueueDemon, SendImageFromIM, SendImageFromMMB, SHMOptimizeDemon, TCAcquireDualImage, TCCompressStoredImage, TCFExecute, TCFMonitorObservation, TCPlainfileRecovery, TCQueueConfig, TCSendStoredImage, WACAcquireQueueDemon	ExtractPulsesFromImage GetImageControlHeaderFromMMB GetImageHeaderFromMMB mmb_ReadMemManagerQuiet MMBBlockToIM MMBIFRead MMBMemoryDump MMBPoweredOnSize TCPlainfileRecovery	
RES_MMB_WRITE_DIB0	MMBIFRead MMBIFWrite CRBAcquireDualImage CRBAcquireImageToMMB	AcquireBias, AcquireDark, AcquireImage, AutoStart, CalibrateShutter, DownlinkManagerDemon, HKMonitorDemon, MMBDefragment, NACAcquireQueueDemon, SendImageFromIM, SendImageFromMMB, SHMOptimizeDemon, TCAcquireDualImage, TCCompressStoredImage, TCFExecute, TCFInit, TCFMonitorObservation, TCInitPlainfileRead, TCPlainfileRecovery, TCQueueConfig, TCSendStoredImage, WACAcquireQueueDemon	TCPlainfileRecovery TCInitPlainfileRead FlushStreamWriter ExtractPulsesFromImage MMBInitHandleMemory, MMBPoweredOnSize, mmb_WriteFAT, mmb_ReadMemManagerQuiet, IMBlockToMMB, MMBBlockToIM, SetImageHeaderToMMB, GetImageHeaderFromMMB, SetImageControlHeaderToMMB, GetImageControlHeaderFromMMB, CRBAcquireDualImage, CRBAcquireImageToMMB, MMBMemoryDump	
RES_MMB_WRITE_DIB1	MMBIFRead MMBIFWrite	AcquireBias, AcquireDark, AcquireImage, AutoStart, CalibrateShutter, Down-	TCPlainfileRecovery TCInitPlainfileRead	

Resource	Acquired by		Released by	
	Function	Top-Level Caller	Function	Top-Level Caller
	CRBAcquireDualImage CRBAcquireImageToMMB	linkManagerDemon, HK-MonitorDemon, MMBDefragment, NACAcquireQueueDemon, SendImageFromIM, SendImageFromMMB, SHMOptimizeDemon, TCAcquireDualImage, TCCompressStoredImage, TCFExecute, TCFInit, TCFMonitorObservation, TCInitPlainfileRead, TCPlainfileRecovery, TCQueueConfig, TCSendStoredImage, WACAcquireQueueDemon	FlushStreamWriter ExtractPulsesFromImage MMBInitHandleMemory MMBPoweredOnSize mmb_WriteFAT mmb_ReadMemManagerQuiet IMBlockToMMB MMBBlockToIM SetImageHeaderToMMB GetImageHeaderFromMMB SetImageControlHeaderToMMB GetImageControlHeaderFromMMB CRBAcquireDualImage CRBAcquireImageToMMB MMBMemoryDump	
RES_NAC_ACQUIRE_QUEUE				
RES_NAC_CRB	GetHKPacketData ShutterPeformHomepush ShutterForceClosed ShutterClose ShutterOpen ShutterConfigSHE ShutterEnableNACBrake ShutterAcquireTestImage ShutterErrorRecovery PowerCycleShutter PowerShutterOff PowerShutterOn PowerCRBOff PowerCRBOn HKGetSensor HKLimitCheckUnit CRBAcquireDualImage CRBAcquireImageToMMB CRBAcquireImageToIM CRBSetAcquireMode		GetHKPacketData ShutterPeformHomepush ShutterForceClosed ShutterClose ShutterOpen ShutterConfigSHE ShutterEnableNACBrake ShutterAcquireTestImage ShutterErrorRecovery PowerCycleShutter PowerShutterOff PowerShutterOn PowerCRBOff PowerCRBOn HKGetSensor HKLimitCheckUnit CRBAcquireDualImage CRBAcquireImageToMMB CRBAcquireImageToIM CRBSetAcquireMode	
RES_NAC_FDM	MCBLockDoor MCBCloseDoor MCBOpenDoor MCBDoorGotoSwitch		MCBLockDoor MCBCloseDoor MCBOpenDoor MCBDoorGotoSwitch	
RES_NAC_FWM	MCBMoveFilterWheel		MCBMoveFilterWheel	
RES_PCM	AnnealControl PowerHeatersT GetHKPacketData PCMUnlockHeaterAccess PCMLockHeaterAccess PowerHeaters PowerMCBOff PowerMCBOn PCMOff		AnnealControl PowerHeatersT GetHKPacketData PCMUnlockHeaterAccess PCMLockHeaterAccess PowerHeaters PowerMCBOff PowerMCBOn PCMOff	

Resource	Acquired by		Released by	
	Function	Top-Level Caller	Function	Top-Level Caller
	PCMOn ConfigPCM PowerSwitchOnOff IsSwitchOn MCBInitFilterWheel HKGetSensor HKLimitCheckUnit		PCMOn ConfigPCM PowerSwitchOnOff IsSwitchOn MCBInitFilterWheel HKGetSensor HKLimitCheckUnit	
RES_RTU				
RES_SCIENCE_LINK	NVRAMMemoryDump MMBMemoryDump SendImageFromIMToPlain-File SendArrayViaIF SendMemoryDump OpenStreamWriter		NVRAMMemoryDump MMBMemoryDump SendImageFromIMToPlain-File SendArrayViaIF SendMemoryDump CloseStreamWriter	
RES_USER_3				
RES_WAC_ACQUIRE_QUEUE				
RES_WAC_CRB	GetHKPacketData ShutterPeformHomepush ShutterForceClosed ShutterClose ShutterOpen ShutterConfigSHE ShutterAcquireTestImage ShutterErrorRecovery PowerCycleShutter PowerShutterOff PowerShutterOn PowerCRBOff PowerCRBOn HKGetSensor HKLimitCheckUnit CRBAcquireDualImage CRBAcquireImageToMMB CRBAcquireImageToIM CRBSetAcquireMode		GetHKPacketData ShutterPeformHomepush ShutterForceClosed ShutterClose ShutterOpen ShutterConfigSHE ShutterAcquireTestImage ShutterErrorRecovery PowerCycleShutter PowerShutterOff PowerShutterOn PowerCRBOff PowerCRBOn HKGetSensor HKLimitCheckUnit CRBAcquireDualImage CRBAcquireImageToMMB CRBAcquireImageToIM CRBSetAcquireMode	
RES_WAC_FDM	MCBLockDoor MCBCloseDoor MCBOpenDoor MCBDoorGotoSwitch		MCBLockDoor MCBCloseDoor MCBOpenDoor MCBDoorGotoSwitch	
RES_WAC_FWM	MCBMoveFilterWheel		MCBMoveFilterWheel	
Undefined resource	**MMBMemoryDump**			

As one can see, there are a lot of synchronization dependencies – already at top-level. There might be even more, if low level dependencies are also taken into account.