XMM-Newton

Calibration Access and Data Handbook

XMM-PS-GM-20 Issue 4.0

Written by:
B. Altieri, B. Chen, C. Erd, C. Gabriel, P. Gondoin,
M. Guainazzi, U. Lammers, D. Lumb, M.G.F Kirsch, R. Much,
A.M.T. Pollock, R. Saxton, M.J.S. Smith, M. Stuhlinger, A. Talavera
and G. Vacanti

May 24, 2011

List of Changes

Changes from Issue 3.2 to Issue 4.0

Major revision of RGS related CCF components (Section 4.4).

In addition, update of the EPIC Redistribution Cal and CCF related components (Sections 3.3.22, 3.3.23, 4.3.27 and 4.3.28).

Changes from Issue 3.1 to Issue 3.2

section 3.3.9: modified MOS CTI modelling

section 4.3.9: added description for the CTI_COLUMN extension in the EMOS_CTI CCF

section 4.3.10: modified description of the LONG_TERM_CTI table in the EPN_CTI CCF; added description for the EFF_GAIN, QBOXTEMP_GAIN and RATE_DEPENDENT_CTI tables in the EPN_CTI CCF.

section 4.3.29: added description for the MASTER_OFFSET_TABLE_INDEX, NOISE_MAP_DUMMY, MEDIAN_MAP_DUMMY, MASTER_OFFSET_MAP and MASTER_MAP_DUMMY tables in the EPN_REJECT CCF.

Changes from Issue 3.0 to Issue 3.1

Updates to most OM related CCF component sections.

Minor editorial corrections in the EPIC related CCF components EPN_ADUCONV and EPN_BADPIX sections.

Changes from Issue 2.2 to Issue 3.0

Major revision of EPIC related CCF components and CAL routine sections.

Changes from Issue 2.1 to Issue 2.2

1. Functional Changes

   Section Mirrors

   - updated description of CCF XPSF (section 4.2.2; additional table); this is used in function `CAL_getEncircledEnergy` (section 3.2.2; new formula) and as a new EXTENDED mode in function `CAL_getPSFmap`(section 3.2.1)

   Section EPIC

   Section RGS

- updated description of CCF EXAFS (section 4.4.12; file did on exist before); this is used in function `CAL_getCCDQuantumEfficiency` (3.4.13)
  Impact on `rgsrmfgen`: This function was not used so far.
- New table `RGA_SELFVIGNCORR` in CCF `QuantumEf` (section 4.4.17) to correct RGA self-vignetting function with empirical values.
  Impact on `rgsrmfgen`: Change of efficiency.
- added a new table `RGA_EFFAREACORR` to CCF `QuantumEf` (section 4.4.11), and a new function `CAL_getRGSEffAreaCorr` (section 3.4.35).
  Impact on `rgsrmfgen`: Change of efficiency.
- addition of a empirical correction factors to the RGA reflection coefficients in CCF `QuantumEf` (section 4.4.17) which is applied by `CAL_getRGAQuantumEfficiency` (section 3.4.27).
  Impact on `rgsrmfgen`: Change of efficiency.
- added function `CAL_getCalSrcRegions` (section 3.4.14) and table `CALSOURCEREGIONS` in CCF `CalSourceData`.
  Impact on `rgsrmfgen`: Background rejection. Was not used before

Section OM

2. Editorial Changes

General

Section Mirrors

Section EPIC

Section RGS

- added the names of the various RGA related tables specified in CCF `QuantumEf` (section 4.4.17), for clarification
- made definition of self-vignetting function (section 3.4.28) more accurate.
- Added comment about restriction of parameters for HTR mode for CTI correction (section 3.4.11).
- added comment that function `CAL_getCalSourceData` (section 3.4.15) and parameterized intensity images in CCF `CalSourceData` are not yet available.
- added comment to RGS CrossPsf CCF section about meaning of `GAUSS_NORM`/`LORENTZ_NORM` columns

Section OM

Changes from Issue 2.0 to Issue 2.1

1. Functional Changes

Section Mirrors

- in `CALgetEncircledEnergy` (section 3.2.2): changed calling interface (new input CCF)
  Impact on `arfgen`: This function was not used so far.
- in `XEncirEn` (section 4.2.1): new Encircled Energy CCF
  Impact on `arfgen`: This file was not used so far.

Section EPIC

- in CAL_getBadPixelList (section 3.3.8) changed calling interface to return magnitude of bad pixel generation in PN, and allow some locations to be mode dependent

Section RGS

- in CAL_rgsGetScatter (section 3.4.20): changed calling interface (input and output)
  Impact on rgsrmfgen: This function was not used so far.
- new functions CAL_rgsGetLAScatter (section 3.4.21) and CAL_rgsGetSAScatter (section 3.4.22).
  Impact on rgsrmfgen: This function was not used so far.
- new functions CAL_getRFCdefocus (section 3.4.24) and CAL_getLSFdefocusDist (section 3.4.25).
  Impact on rgsrmfgen.
- new function CAL_getGratBow (section 3.4.32) and new table BOWS in CCF RGS_LineSpreadFunc (section 4.4.15).
  Impact on rgsrmfgen.
- change of definition of function CAL_getRFCdefocus (section 3.4.24)
  Impact on rgsrmfgen: This function was not used so far.
- documentation and change of name of scaling paramter in CAL_getRGAFigure (section 3.4.31) and CCF LineSpreadFunc, table FIGURE (section 4.4.15).
- change of CCD offset calibration, which is now a function of observation time: CAL_offsetCorrect (section 3.4.9) and CCF ADUConv (section 4.4.4)
  Impact on: rgsevents, rgsregions, rgsrmfgen; see releasenote of CCF ADUCONV XMM-CCF-REL-68
- adding broadeningDistribution (section 3.4.26)
  Impact on rgsrmfgen.

Section OM

2. Editorial Changes

General

Section Mirrors

Section EPIC

Section RGS

- in CAL_rgsGetScatter (section 3.4.20)
  - an explanation of the the meaning of β_{m} is added
  - correction: σ_{large} is a function of wavelength, not l_{large}
  - correction of error in exponent of Eq. (13)
  - correction of error in Eq's (18) & (19): removal of factor R
  - changed typo of input parameter from NFWHM to NHWHM to reflect actual implementation
- in CAL_rgsgetXRTFigure (section 3.4.33) changed typo of input parameter from NFWHM to NHWHM to reflect actual implementation

- in `CAL_rgsgetXRTFigure` (section 3.4.33) added missing factor $\frac{1}{\pi}$ in definition of Lorentz function.
- in `CAL_rgsCrossPSF` (section 3.4.23) added missing factor $\frac{1}{\pi}$ in definition of Lorentz function.
- in the description of the contents of CCF `Quantumef`, in section 4.4.17, text was changed to better clarify the contents of layer thickness data.

Section OM

- corrected typo: `COINCIDENCE` table driving empirical linearity correction is located in constituent `PHOTTONAT` (*not*`COLORTRANS`)
- corrected several typos and misspellings in all OM sections

Changes from Issue 1.1 to Issue 2.0

1. Functional Changes

Section Mirrors

Section EPIC

Section RGS

- CTI correction changed from just proportional to distance to also proportional to pulse-height.
  Impact `rgsevents`: The interface needs to be activated explicitly. The task so far used the generic CTI correction by the CAL.
- new function `CAL_rgsGetScatterPars` (section 3.4.19) to provide access to the paremeters of the RGS scatter functions.
  Impact `rgsrmfgen`: These parameters were so far hard coded inside the task.
- new function `CAL_rgsGetLAScatterRoughness` (section 3.4.18) to provide the large angle scatter parameters that are tabulated as a function of wavelength.
  Impact `rgsrmfgen`: These parameters were so far hard coded inside the task.
- change of layout of CCF `LineSpreadFunc`, table `LASCAT` (previously unused).
  Impact `rgsrmfgen`: These parameters were so far hard coded inside the task.
- change of description and first implementation of `CAL_getRGAFigure` (section 3.4.31) and change of structure of CCF `LineSpreadFunc`, table `FIGURE` (section 4.4.15; previously unused) to provide the necessary parameters.
  Impact `rgsrmfgen`: These parameters were so far hard coded inside the task.
- adding function `CAL_rgsgetXRTFigure` (section 3.4.33) and related addition of table `BETAPSF` in CCF `XRT_XPSF` (section 4.2.2).
  Impact `rgsrmfgen`: These parameters were so far hard coded inside the task.
- modification of function `CAL_rgsCrossPSF` (section 3.4.23) to allow for superposition of several Gaussian and Lorentzians. This changes the structure of the CCF `CrossPSF`.

Impact rgsregion, rgsrmfgen: Only change of calibrations, no interface change.

- change CAL_rgsGetScatter (section 3.4.20) due to change of scatter tabulation above. Additionally this function is now implemented in the CAL (was not before).
  Impact on rgsrmfgen.
- change sign of Equation 28. This has affects on the effective area of RGS and of the effective area of MOS for off-axis sources.
  Impact on , rgsrmfgen, arfgen.
- new implementation of partial event floor of CCD redistribution in CAL_getRedistribution (section 3.4.12). This requires also additional parameters in CCF RGS_REDIST (section 4.4.18).
- new function CAL_getRFCdefocus (section 3.4.24).
  Impact on rgsrmfgen.
- new function CAL_getGratBow (section 3.4.32) and new table BOWS in CCF RGS_LineSpreadFunc (section 4.4.15).
  Impact on rgsrmfgen.

Section OM

- CAL_omGetColorTrans (section 3.5.7): functional form of colour transformation equations is changed. Magnitudes of the standard system now occur on the left hand side of the colour equations.
  Branching of the colour equations is supported to describe colour transformations for objects with different metalicities. A new optional input parameter is used to select the different branches. The new input parameter requires a change in the CCF structure of the CCF OM_COLORTRANS (section 4.5.4).
  Impact ommag: the task has to support a new optional parameter.
- CAL_omPhotoNatural (section 3.5.14): the function is changed to support usage of the theoretical and the empirical linearity correction. The correction is selected via a new optional parameter. The empirical linearity correction is used per default.
  A new extension is required in the CCF OM_COLORTRANS to describe the empirical linearity curve (section 4.5.4).
- CAL_getAperRadius (section 3.5.16): The new routine returns the used aperture radius in the photometric calibration and checks for consistency between the aperture radii specified in the CCFs PSF1DRB and COLORTRANS.
  The storage of the aperture radius used in the calibration requires one additional keyword per filter in the CCFs OM_COLORTRANS (section 4.5.4) and OM_PSF1DRB (section 4.5.10).

2. Editorial Changes

General the definition of the coordinate systems that are being used by the CAL has been added (section 2)

Section Mirrors

Section EPIC

Section RGS

- split description of CCF LineSpreadFunc (section 4.4.15) per table that is provided in the CCF.

Section OM

- changed font of input parameters in sections *Calling Parameters* from bold face to typewriter where applicable.

Changes from Issue 1.0 to Issue 1.1

Section Mirrors

- updated the description of the PSF parametrization in section 3.2.1 in line with CCF version XRT1_XPSF_0006, XRT2_XPSF_0002 and XRT3_XPSF_0002 and later.
  editorial change only.

Section EPIC

- Editorial updates to improve description for SAS public release
- MOS_CTI updated for in-orbit damage
- Highlighted descriptions not yet implemented
- Description of background event file changed
- More detailed description of CCF file components

Section RGS

- expanded the description of RGS off-axis calculation in section 3.4.29

Section OM

Changes from Draft 0.4 to Issue 1.0

Section Mirror

- The CALgetPSFmap routine has been updated
- The X-ray telescopes effective area calibration files now only contain an on-axis effective area table and a table of vignetting factors at different energies and off-axis angles. The vignetting effect by the RGA is taken into account separately by an RGA calibration file.

Section EPIC

- Major changes

Section RGS

- typo corrected in formula for dark frame evaluation in 3.4.8 and formula now symmetric for a_i & b_i
- `CAL_getRGAIntercept` has a dependence on energy now

- the former RGS function CAL_getEffectiveArea is now renamed to CAL_getRGSEffectiveArea to avoid mis-understandings with the corresponding mirror related function
- CAL_getRGSEffectiveArea now also depends on BETA
- new function CAL_getRGAObscure
- additional extensions in CCF QuantumEf
- in CCF HKParmInt changes of column names of AUXPAR%PARM_ID to AUXPAR%PARM_NAME
- CCD redistribution uses the CCF REDIST, and not CCDrmf; this change only affects this document.

Section OM

- change algorithm and call in CAL_omPhotoNatural
- illumination template of internal lamp added to CCF file PixToPixSens
- CAL_omLEDTemplate added

Changes from Draft 0.3 to Draft 0.4

Section EPIC

- Major changes throughout the EPIC section

Section RGS

- corrections of errors in CAL_getCCDRedistribution

Section OM

- Major changes throughout the OM section

Changes from Draft 0.2 to Draft 0.3

Section Mirror

- description of CAL_getPSFmap routine updated
- description of CAL_getEncircledEnergy routine updated
- description of CAL_getEffectiveArea routine updated
- description of CAL_getVignettingFactor routine updated
- description of XPSF CCF file updated
- description of XAreaEf CCF file updated

Section EPIC initial version included

Section RGS

- SHAPE added in CAL_offsetCorrect
- changed parameter interface of CAL_rgsCrossPSF
- description of CAL_getHKwindows, CAL_getAUXwindows and CCF HKParmInt added

- description of CCF Contam added
- all unit alignment data are now stored in CCF LinCoord
- description of CCF QuantumEf added
- CAL_rgsGetLSF added
- RGA efficiencies added (CCF QuantumEf) & CAL_getRGAQuantumEfficiency
- CAL_rgsGetScatter added to describe the scattering of the LSF & description in CCF LineSpreadFunc
- CAL_getRGAVign added
- CAL_getRGAIntercept
- functions for figure properties of RGA (CAL_getRGAFigure)
- CAL_rgsgetLSF added

Section OM initial version included

Changes from Draft 0.1 to Draft 0.2

Section Mirror

- Mirror Related CCF Components added

Section RGS

- DPP version number added to CCF ModeParam
- modifications of the contents of CCF ClockPatterns
- CAL_rgsCTIcorrect updated to contain the correct functional form of the CTI correction; this also affects CCF CTI.
- CAL_rgsGetEvThresh updated
- description of CAL_getCCDRedistribution added
- cross-reference tables for maintenance of CCF constituents added

Changes from Draft 0.0 to Draft 0.1

- CAL_beta2wavelength & CAL_energy2wavelength deleted
  this functionality will be part of the package rgslib
- CAL_offsetCorrect to take integer PHA as input and perform randomization inside by using the randomization interval as stored in CCF ADUConf
- CAL_chipX2rawX added

Contents

1 Introduction

This CAL HB also serves as a reference documentation for users of XMM data. It describes in detail what calibration data/algorithms are employed in the pipeline/interactive processing of observation data.

1.1 Scope of the Document

This document contains a high level description of the calibration access layer (CAL) and of the constituents of the current calibration files (CCF).

1.2 Documentation

[1] C. de Vries. A simple CCD pulse height model function. RGS-SRON-CAL-ME-98/010, Space Research Organization of the Netherlands, November 1998. http://ws13.sron.nl:8080/xmmdoc/effective_area/rgs-sron-cal-me-98_010.ps.

[2] C. de Vries. Modified CCD pulse height model including partial events tail. RGS-SRON-CAL-ME-00/cv8, Space Research Organization of the Netherlands, January 2001. http://ws13.sron.nl:8080/xmmdoc/effective_area/rgs-sron-cal-me-00_cv8.ps.

[3] J. Cottam. Final Model of the RGS Grating Efficiency. RGS-COL-CAL-99005, Columbia, May 1999.

[4] A. Rasmussen. Tuining and Assessment of the RGS LSF model appropriate for Response Matrix Generation. RGS-COL-CAL-01002, Columbia, March 2001. http://xmm.astro.columbia.edu/cal_files/cal01002.ps.

[5] ESA. Interface Control Document for the XMM Current Calibration File. XMM-SOC-ICD-0005-SSD, ESA/SSD, Dec 1999. Issue 3.4.

2 Coordinate Definitions used by the CAL

2.1 Spatial coordinate systems

The CAL makes use of the following spatial coordinate systems

- PIXCOORD0: Telemetry-related CCD pixel coordinate system
  A pair of two integer numbers $telX/telY$ specifies a pixel position on the currently defined CCD chip (state variable $ccdChipId$) which is read out through one of two

nodes (state variable *ccdNodeId*). Pixels outside the actual image area of the chip (over/underscan pixels) are taken into account and are counted as ordinary pixels. The pixel closest to the readout node is designated as (0/0). If a readout window is defined PIXCOORD0 coordinates are relative to the lower left corner of that window.

- PIXCOORD1: Node-oriented CCD pixel coordinate system
  A pair of two integer numbers - referred to as $rawX/rawY$ in the following - specifies a pixel position on the currently defined CCD chip (state variable *ccdChipId*) which is read out through one of two readout nodes (state variable *ccdNodeId*). The pixel coordinate system is, thus, **node-oriented**. Example: Pixel position (123/456) on the central EMOS1 chip readout through the primary node corresponds to pixel $(600 - 123 + 1 = 478/456)$ on the same chip read-out through the redundant node. CCD pixel coordinates are in the range $[1, max_x] \times [1, max_y]$ where max_x, max_y are the number of available pixels in X/Y for the current instrument.

  In the case of OM the notion of raw pixels is identical to *centroided* pixels. The size of the detector in centroided pixels is 2048×2048.

  Over- and/or underscan pixels are no longer considered. PIXCOORD1 is no longer relative to any defined readout window.

- CHIPCOORD: Node-free CCD pixel coordinate system The CHIPCOORD system is identical to the PIXCOORD1 system except for the removal of the node-orientation. All pixel coordinates are referred to the primary/first readout node.

- CAMCOORD1/CAMCOORD2: Camera reference systems
  A triple of three real numbers (*xmm*/*ymm*/*zmm*) specifies a physical location in the camera system in units of mm from the origin of a right-handed, Cartesian reference system. The origin in CAMCOORD1 is the geometrical center of the camera (center of central chip for MOS/RGS/OM and center of chip wafer for EPN), in CAMCOORD2 it is the point where the optical axis of the telescope in front of the the camera intersects the focal plane. For RGS1 the CAMCOORD1 and CAMCOORD2 systems are identical.

  The alignments of the CAMCOORD1/2 $+X$ and $+Y$ axes with respect to the PIXCOORD1 (see Sect. 2.1) and the SACCOORD (see Sect. 2.1) frames for all cameras are as specified in the following table:

Camera unit(s)	alignment of CAMCOORD1/2 axes in PIXCOORD1 frame	$+X_{CC1/2}$ aligned with axis in SACCOORD frame	$+Y_{CC1/2}$ aligned with axis in SACCOORD frame
EMOS1/2	$+X/+Y$ axes aligned with the respective axes of primary node of central CCD	$+Y_{SC}/+Z_{SC}$	$-Z_{SC}/+Y_{SC}$
EPN	$+X/+Y$ axes aligned with respective PIXCOORD1 axes of CCDs 7–12 in quadrants 2/3	$-Z_{SC}$	$-Y_{SC}$
RGS1/2	$+X/+Y$ axes aligned with $+X/-Y$ PIXCOORD1 axes	$+Z_{SC}$ rotated by 7.333°	$+Y_{SC}$
OM	$+X/+Y$ axes aligned with $-X/+Y$ PIXCOORD1 axes	$+Z_{SC}$	$+Y_{SC}$

- SACCOORD: Spacecraft reference system
  A triple of three real numbers ($xmm/ymm/zmm$) specifies a physical location in the satellite in units of mm from the origin of the right-handed, Cartesian spacecraft reference system which is defined as follows:
  - The origin is located in the center of the circle inscribing the optical axes of the three mirror modules in the plane of the mirror support platform which faces the focal plane assembly.
  - The $+X$ axis is perpendicular to the mirror support platform, pointing positively towards the focal plane assembly, i.e., along the direction of incoming X-rays. Please note that the $-X$ axis corresponds to the viewing direction of the telescope (not taking the effect of boresight misalignment into account).
  - The $+Z$ is defined by the point where the optical axis of mirror module 3 (which is in front of the EPN camera) intersects the mirror support platform and the origin. $+Z$ is pointing positively away from mirror module 3 (and the optical monitor) and is perpendicular to the solar panels.
  - The $+Y$ axis completes the right-handed orthogonal coordinate frame.
- TELCOORD: Telescope coordinate system
  A right-handed Cartesian reference system whose +X axis is defined by the optical axis of the telescope in front of the currently set instrument (state variable *instrument*) along the direction of incoming photons. The +Y and +Z axes are aligned with the respective axes of the SACCOORD (see Sect. 2.1) frame. The origin lies in the mirror support platform, so, in front of the telescope when looking from the sky along the +X axis.

  A pair of two real angles (θ, ϕ) represents a position in the +Y/+Z plane. The θ angle signifies the angular deviation from the X-axis, i.e., $\theta = 0$ is the on-axis case corresponding to $Y = Z = 0$. ϕ is the azimuthal angle measured in the mathematically positive sense (anti-clockwise) from the +Z-axis when looking along the +X axis. Please note: If the TELCOORD system is translated to the focal plane the mirror inversion effect has to be considered.
  The units for θ and ϕ are arcsecs and radians respectively.

Example: $(600, \pi/2)$ specifies a point being 10 arcmin off the optical axis with an azimuth of +90 degrees away from the spacecraft +Z-axis, i.e., on the spacecraft +Y-axis. In the CAMCOORD2 (see Sect. 2.1) system of EMOS1 this approximatley refers to the mid-point of CCD 6 and for EMOS2 is refers to the mid-point between CCDs 7 and 2 respectively.

- ROWCOORD: Rowland coordinate system
  ROWCOORD is a RGS-specific coordinate system where the two angles β and χ designate an event position in the dispersion and cross-dispersion direction respectively. $(\beta/\chi) = (3.228°/0)$ are the ROWCOORD coordinates of the central pixel of CCD 5 (M-point). β increases positively in the direction of increasing dispersion. χ increases in the direction of the CAMCOORD1 (see Sect. 2.1) $+Y$ axis. Please note: β/χ designate the non-aspect corrected dispersion/cross-dispersion angles in the frame of the RGA. β/χ are defined with respect to the RGA-G point and the center of the mirror module respectively.

Please note: A pixel location in the CHIPCOORD system is given by two integer numbers. When converting any such CHIPCOORD pixel location to the CAMCOORD1 frame the corresponding physical location is that of the *center* of the pixel.

The following figures illustrate the alignment of the PIXCOORD1/CAMCOORD1/CAMCOORD2 and SACCOORD reference system with respect to each other for the six cameras. The last figure is a schematic overview of all available coordinate systems in the CAL and the provided paths of conversions between them.

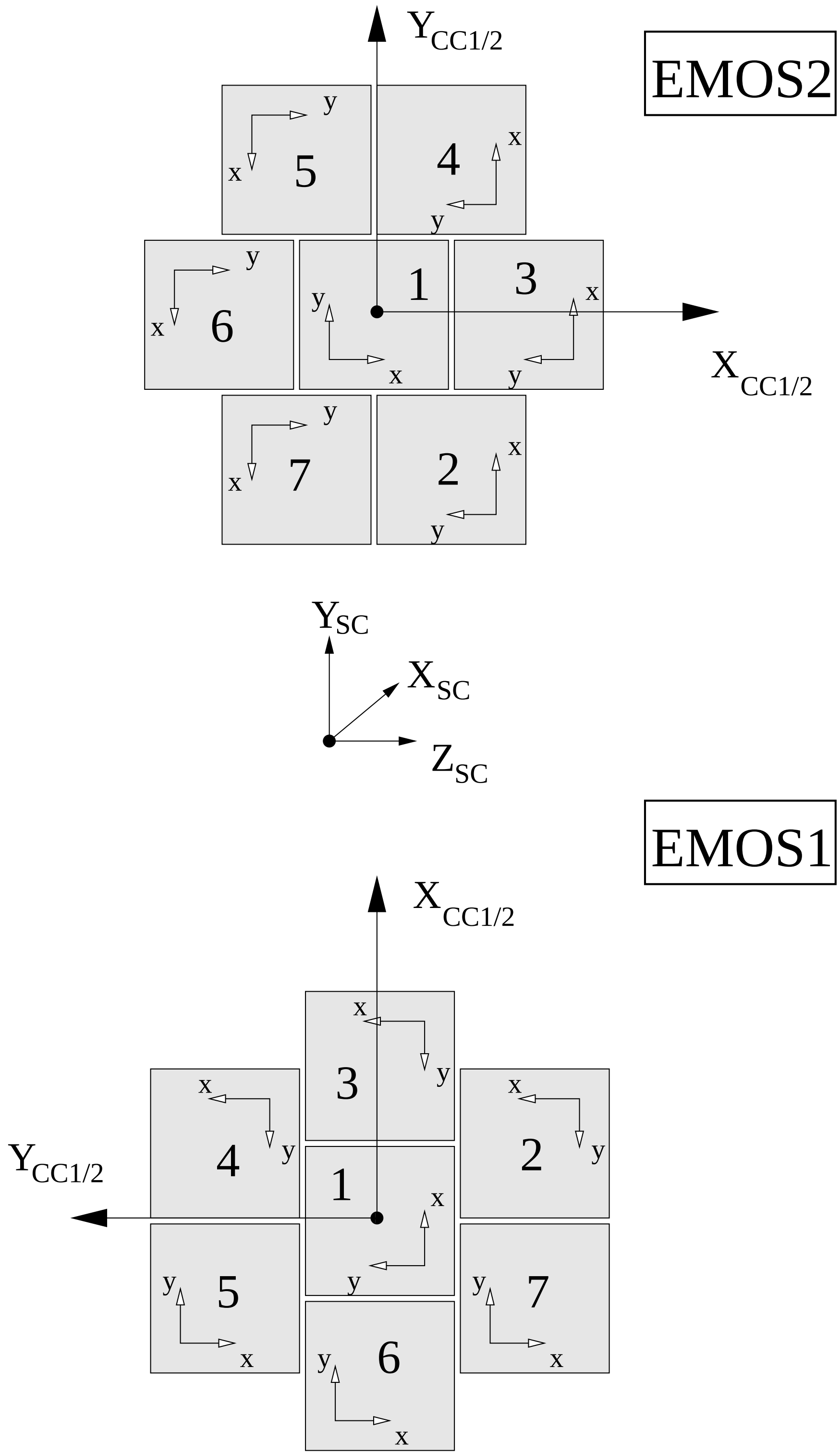

Figure 1: Alignment of the CAMCOORD1/2 frames with respect to the PIXCOORD1 and SACCORD frames for the EMOS1 and EMOS2 cameras. Please note that the two CAMCOORD frames are identical because the telescopes' optical axes intersect the cameras in their centers.

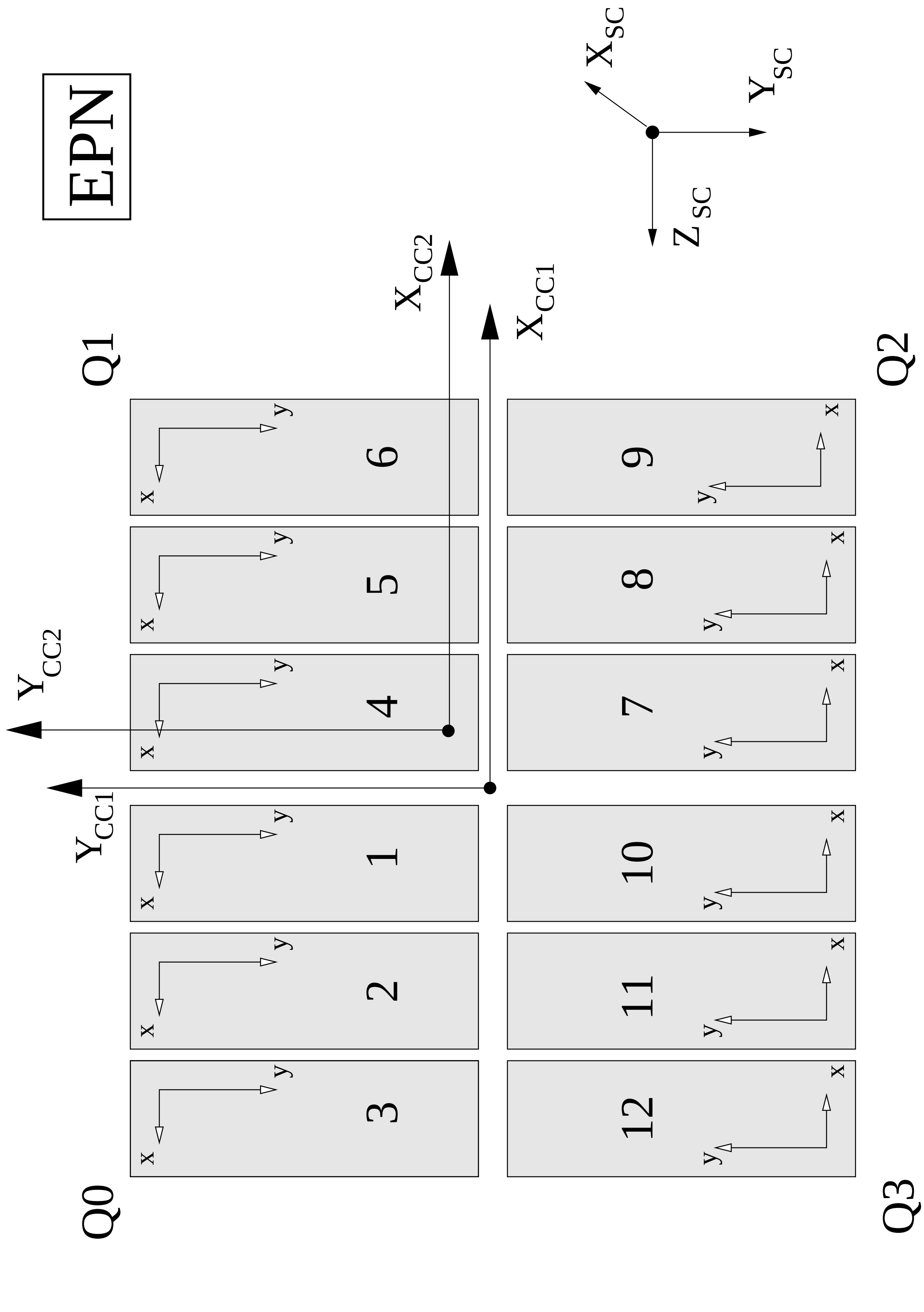

Figure 2: Alignment of the CAMCOORD1/2 frames with respect to the PIXCOORD1 and SACCORD frames for the EPN camera. Please note that the CAMCOORD2 frame is offset from the CAMCOORD1 frame by about (4mm/2mm) in the $-Z$ and $-Y$ SACCOORD direction respectively. The optical axis of mirror module 3 in front of EPN intersects the camera plane in the origin of CAMCOORD2.

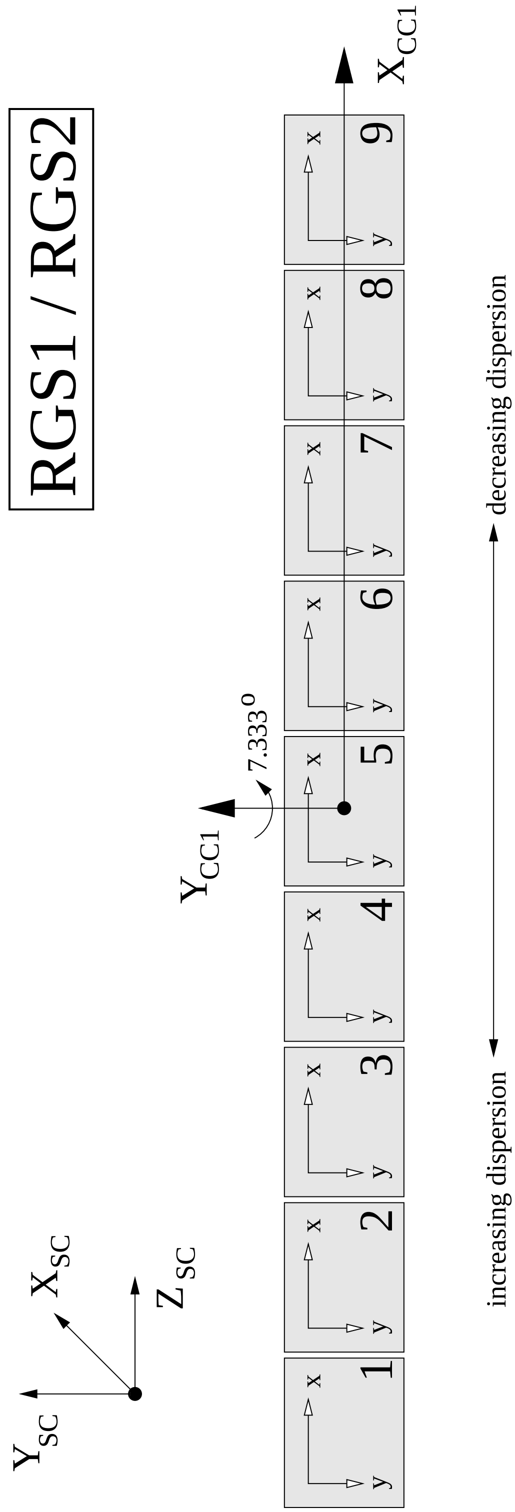

Figure 3: Alignment of the CAMCOORD1 frame with respect to the PIXCOORD1 and SACCORD frames for the RGS camera units. Please note that strictly speaking CAMCOORD2 does not exists for RGS, however, it is formally equivalent to CAMCOORD1. The CAMCOORD1 X/Y plane is rotated by 7.333° with respect to the SACCOORD Y/Z plane around the $+Y_{CC1}$ axis. Increasing values of the CAMCOORD1 X coordinate correspond to decreasing dispersion, i.e., increasing energy (decreasing wavelength).

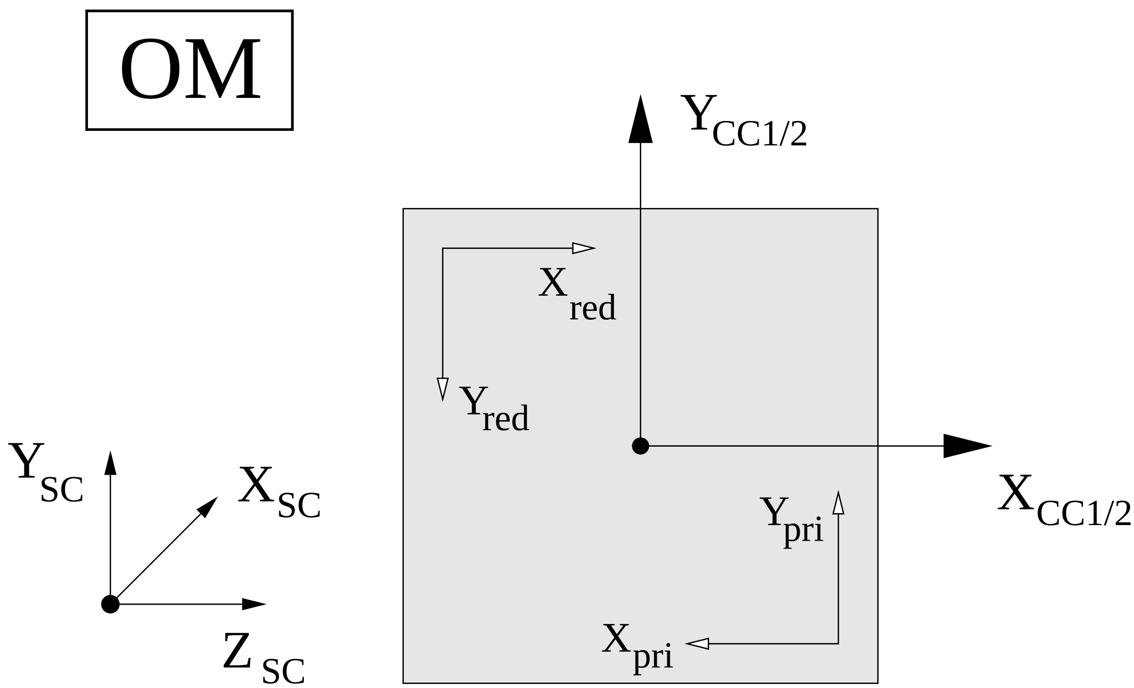

Figure 4: Alignment of the CAMCOORD1/2 frames with respect to the PIXCOORD1 and SACCORD frames for the OM camera.

2.2 Time coordinate system (TIMCOORD)

Several calibration entities are time dependent, i.e., change on a time-scale given by the typical duration of an observation (∼ksecs). Each corresponding CAL subroutine receives a time argument which specifies a point in time represented as ellapsed number of seconds since the fixed mission reference time
`1998-01-01T00:00:00.00 TT = 1997-12-31T23:58.56.816 UTC`

3 Routines of the Calibration Access Layer

This section does not intend to substitute the detailed CAL API Guide, it rather is an integral part of the overall description of the CAL. During the development it is used as input to the definition of the CAL.

It is TBD in many cases, whether certain variables will be implemented as input arguments, or CAL State Variables. A final decision will be left to the implementation, and the documentation of the CAL should be consulted.

Possible users are strongly advised not to refer directly to the CCF constituents, as in most of the cases the actual calibration quantity is produced by the appropriate CAL function, in combination with data from one or more CCF constituents. For this reason a utility called `calview` is provided as part of the SAS, in order to enable the user to investigate the calibration quantities in a transparent manner.

3.1 Generic CAL Routines

3.1.1 CAL_random

3.1.1.1 Procedure

Define the range of random numbers to be returned by the Random Number Generator used by the CAL.

3.1.1.2 Calling Parameters

3.1.1.2.1 Input

`min, max` `CAL_random` returns a random number in the interval [`min,max`]

3.1.1.2.2 Output

none

3.1.1.3 List of CCF Components used

none

3.1.2 CAL_integerToReal

3.1.2.1 Procedure

Convert a vector with integer number into a corresponding real-valued vector thereby performing randomization (see also section 3.1.1).

3.1.2.2 Calling Parameters

3.1.2.2.1 Input

`in` integer vector

`min, max` interval for randomization by `CAL_random`

3.1.2.2.2 Output

`out` real vector corresponding to input

3.1.2.3 List of CCF Components used

none

3.2 Mirror Related CAL Routines

3.2.1 CALgetPSFmap

3.2.1.1 Procedure

Returns the mirror point-spread-function sampled on an equidistant, square spatial grid centered around a specified point in the focal plane. The state variable `AccuracyLevel` determines the scheme to be used to construct the PSF:

Accuracy Level	scheme	description
LOW	analytic	the PSF is constructed using a single bi-dimensional Gaussian fit involving energy dependent parameters which are tabulated in the CCF. This mode is a fastest way to compute the PSF but with a very limited accuracy. In particular, it shall not be used for encircled energy calculations.
MEDIUM	library	the PSF is constructed from a library of PSF images provided in the CCF. This is meant to be most direct way to access the PSF. A set of images is extracted from the CCF. Interpolations in energy and field angle are performed. Resampling of the extrapolated image is performed to account for mismatches (e.g rotation) between the requested spatial grid and the one at which the PSF image are stored in the CCF.
HIGH	analytic	the PSF is constructed using multi-Gaussian fits involving parameters which are tabulated in the CCF. This mode involve a high number of fitting paramteters which provide an accurate description of the core and wings of the PSF. The validity range is limited to energies lower than 5 keV and field angles smaller than 7 arcmin.
EXTENDED	analytic	the PSF is constructed using a single one-dimensional King profile. The parameters of the King function are tabulated in terms of energy and off-axis angle in the CCF. This method gives good accuracy but is only relevant for the one-dimensional case. The parameters of the King function have been measured for energies in the full XMM band pass and at off-axis angles upto 12 arc minutes. However, there are gaps in the applicability which are described in Ghizzardi, S., 2001, EPIC-MCT-TN-011.

The analytical expression use to describe the PSF in the low accuracy mode is as follows.

$$PSF(y,z) = 1/(2\pi P_2 P_3 p^2)\exp\left[-1/2\left\{((y-P_4)/P_2)^2 + ((z-P_5)/P_3)^2\right\}\right]$$

where y and z are expressed in pixels units. p is the pixel size expressed in mm. The analytical expression use to describe the PSF in the high accuracy mode is as follows:

$$
\begin{aligned}
PSF(y,z) &= 1/(Np^2)(\\
& A_1 \exp\left[-\left\{(y-A_2)^2/A_4 + (z-A_3)^2/A_5\right\}\right](1 + A_6\cos(3\phi + A_7)) + \\
& B_1 \exp\left[-\left\{(y-B_2)^2/B_4 + (z-B_3)^2/B_5\right\}\right](1 + B_6\cos(3\phi + B_7)) + \\
& C_1 \exp\left[-\left\{(y-C_2)^2/C_4 + (z-C_3)^2/C_5\right\}\right](1 + C_6\cos(16\phi + C_7)))
\end{aligned}
$$

and

$$\phi = \arctan((y - A_2)/(z - A_3))$$

N is a normalisation factor and p is the pixel size expressed in mm. The PSF is provided in a local reference frame where the y and z dimensions are expressed in pixels. The y axis points in the direction opposite to the center of the field of view. The normalization of the PSF is such that

$$
\begin{aligned}
1 &= \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} dy\, dz\; PSF(y,z) \\
&\approx p^2 \sum_{i,j}^{n_i,n_j} PSF(y_i, z_i)
\end{aligned}
$$

The analytical expression used to describe the PSF in the extended accuracy mode is as follows:

$$PSF(r) = A\left\{\frac{1}{\left[1 + (\frac{r}{r_c})^2\right]^\alpha}\right\}$$

The PSF images provided in the library are sampled on a grid of `(Y_EXTENT,Z_EXTENT)` pixels with `Y_PIXSZ, Z_PIXSZ` pixel size. These parameters are specified in the CCF components. They are the results of a trade-off between simulation times and simulation accuracy defined by field of view and oversampling. The fitting routines are then applied to these sampled images. Hence, the (y, z) coordinates are expressed in sampling pixel units.

The CAL provides the PSF in a physical reference frame $(y_{\mathrm{mm}}, z_{\mathrm{mm}})$ in units of mm which is centerd on the PSF barycenter and whose axes are aligned with the CAL TELCOORD frame. The conversion $(y, z) \longrightarrow (y_{\mathrm{mm}}, z_{\mathrm{mm}})$ is given by:

$$
\begin{aligned}
y_{\mathrm{mm}} &= (y' + F')\cos(\phi) + z'\sin(\phi) + \delta y \\
z_{\mathrm{mm}} &= (-y' + F')\sin(\phi) + z'\cos(\phi) + \delta z
\end{aligned}
$$

with

$$
\begin{aligned}
y' &= p(y - \frac{\mathrm{Y\_EXTENT} - 1}{2}) \\
z' &= p(z - \frac{\mathrm{Z\_EXTENT} - 1}{2})
\end{aligned}
$$

and

$$F' = (F + \delta F)\tan(\theta)$$

F, θ, ϕ are the focal length, the off-axis angle and azimuth of the source position with respect to the EPIC boresight. p is the sampling pixel size defined in mm in the CCF. $\delta f(\theta)$, δy and δz are correction factors which account for the mirror module field curvature, EPIC defocus, mirror module tilt with respect to boresight and EPIC decenter with respect to the average focal point of the mirror modules. Initially these correcting factors will be set to 0.

3.2.1.2 Calling Parameters

```
subroutine CAL_getPSFmap(E, theta, phi, PSFmap)
```

E: energy in eV at which the PSF shall be generated.

theta: field angle in arcsrc at which the PSF shall be generated.

phi: field azimuth in radians at which the PSF shall be generated.

3.2.1.3 Output

PSFmap: two-dimensional array containing the selected part of the PSF at the requested azimuth and field angle. The PSF map is sampled with either EPIC MOS (0.040 mm) or EPIC pn (0.140 mm) pixel size. The PSF map cover a field of view of (Y-EXTENT x Y-PIXSZ, Z-EXTENT x Z-PIXSZ) i.e initially 10.24 x 10.24 mm or 4.7 by 4.7 arcmin (256 x 256 EPIC MOS pixels or 73 by 73 EPIC pn pixels). The integral of the PSFmap is normalized to 1. The undersampling of the PSF by the pn pixels is taken into account by retrieving the EPIC pn alignment parameters vs the telescope.

3.2.1.4 List of CCF Components used:

For the SciSim CCF, one of the following CCF components shall be used depending on the selected telescope.

```
XRT1_XPSF_????_SCISIM.CCF
XRT2_XPSF_????_SCISIM.CCF
XRT3_XPSF_????_SCISIM.CCF
```

For the XMM CCF, one of the following CCF components shall be used depending on the selected telescope.

```
XRT1_XPSF_????_SCISIM.CCF
XRT2_XPSF_????_SCISIM.CCF
XRT3_XPSF_????_SCISIM.CCF
```

3.2.2 CALgetEncircledEnergy

3.2.2.1 Procedure

This routine computes the value of the PSF enclosed energy fraction (EE) according to the input arguments described below. The encircled energy (EE) depends on the energy, on the position of the object within the EPIC FOV and on the radius and position of the circular window in which it has to be calculated.

The encircled energy function is calculated by integrating the PSF, described as a single King function, out to the input radius by:

$$EE(R) = \frac{1 - \frac{1}{\left[1+(\frac{R}{r_c})^2\right]^{\alpha-1}}}{1 - \frac{1}{\left[1+(\frac{5'}{r_c})^2\right]^{\alpha-1}}}$$

Where r_c is the core radius in arcseconds and α is the slope.

The parameters of the King function have been measured for the three mirror modules and described in terms of photon energy and off-axis angle by the formulae:

$$r_{\rm c} = a + b1 \cdot E + b2 \cdot E^2 + c \cdot \theta + d \cdot E \cdot \theta$$

$$\alpha = x + y1 \cdot E + y2 \cdot E^2 + z \cdot \theta + w \cdot E \cdot \theta$$

From these formulae the parameters r_c and α have been tabulated as a function of energy and θ and stored in the CCF components XRT?_XPSF. This routine uses spline fitting to interpolate between these tabulated values.

3.2.2.2 Calling Parameters

```
subroutine CAL_getEncircledEnergy(E, theta, Radius)
```

E: energy in eV for which the enclosed energy value will be computed

theta: field angle in arcsec which defines the position of the PSF barycenter and the center of the circular extraction window.

Radius: radius specification in mm of the window in which the enclosed energy has to be calculated. It is assumed that the extraction window is circular and perfectly centered on the PSF barycenter.

3.2.2.3 Output:

value of the PSF enclosed energy fraction (EE) in the circular window with the specified radius and off-axis position.

3.2.2.4 List of CCF Components used

One of the following CCF components shall be used depending on the selected telescope.

```
XRT1_XPSF_????.CCF
XRT2_XPSF_????.CCF
XRT3_XPSF_????.CCF
```

3.2.3 CALgetEffectiveArea

3.2.3.1 Procedure

Return the telescope effective area in square cm averaged over the given energy band and at a specified position in the telescope field of view. Vignetting by the X-ray baffles and by the reflection grating spectrometer is taken into account. The state variable AccuracyLevel determines the scheme to be used to calculate the effective area:

Accuracy Level	scheme	description
NORMAL	library	the effective area at any position in the field of view is calculated from an on-axis effective table and from a set of vignetting tables provided in the CCFs. The vignetting tables are normalized to one on-axis and are provided as a function of the off-axis angle for energies ranging from 100 eV to 15 keV. Interpolations in energy and field angle are performed to provide the value of the vignetting function at the requested energy and field position. This value is then multiplied by the on-axis effective area tabulated in the CCF. For the MOS camera, the results is then multiplied by the vignetting law of the reflection grating assembly.

3.2.3.2 Calling Parameters

```
subroutine CAL_getEffectiveArea(theta, phi, Elo, Ehi, area)
          use types
          real(kind=double), intent(in)                  :: theta, phi
          real(kind=double), intent(in)                  :: Elo, Ehi
          real(kind=double)                              :: area
end subroutine
```

theta: off-axis angle position in the TELCOORD reference system at which the effective area curve is requested.

phi: azimut angle position in the TELCOORD reference system at which the effective area curve is requested.

Elo, Ehi: energy range at which the effective area in square cm will be calculated.

3.2.3.3 Output:

area: effective area value in square cm average over the selected energy range at the selected source position.

3.2.3.4 List of CCF Components used

`MISCDATA`
`XareaEf`

3.2.4 CALgetVignettingFactor

3.2.4.1 Procedure

This routine computes the mirror vignetting factor for a particular energy band at a specific position in the focal plane as the ratio between the mirror effective area values for that position and on-axis. Thus, calling the routine with theta = 0 will return 1.0 for all azimuthal angles and energies. Vignetting by the X-ray baffles and by the reflection grating spectrometer is taken into account. The state variable AccuracyLevel determines the scheme to be used to calculate the effective area:

Accuracy Level	scheme	description
NORMAL	library	the vignetting factor at any position in the field of view is calculated from a library of vignetting tables provided in the CCF. Interpolations in energy and field angle are performed to provide the value of the vignetting function at the requested energy and field position.

3.2.4.2 Calling Parameters

```
subroutine CAL_VignettingFactor(theta, phi, Elo, Ehi, factor)
```

theta: off-axis angle position in the FOCCOORD reference system at which the effective area curve is requested.

phi: azimut angle position in the FOCCOORD reference system at which the effective area curve is requested.

Elo, Ehi: energy band at which the effective area will be calculated.

3.2.4.3 Output:

factor: computed vignetting factor in the range $[0, 1]$ corresponding to the input arguments.

3.2.4.4 List of CCF Components used

XareaEf

3.3 EPIC Related CAL Routines

3.3.1 CAL_getHKwindows

3.3.1.1 Procedure

Provides HK parameter windows for definition of Good Time Intervals. The parameter names stored in CCF `HKParmInt`, are returned together with their upper and lower validity limits as arrays.

3.3.1.2 Calling Parameters

3.3.1.2.1 Input

`time` event time co-ordinates in the TIMCOORD time frame (see CAL documentation, section 5.2.4.5.4)

3.3.1.2.2 Output

`PARAMETER` (array) name of parameter to be checked

`SELECT` (array) `selectlib` expression that describes the valid range

3.3.1.3 Used CAL State Variables

instrument, ccdChipId

3.3.1.4 List of CCF Components used

- `HKParmInt`

3.3.2 CAL_getAUXwindows

3.3.2.1 Procedure

Provides auxiliary science parameter windows for definition of GTI's. For example if the `NABOVE` keyword (which describes number of events above threshold in a frame) were found to be a suitable filter to catch enhanced background, the limits would be defined in the *auxparlist* extension `HKParmInt`, and returned together with their upper and lower validity limits as arrays.

3.3.2.2 Calling Parameters

3.3.2.2.1 Input

`time` event time coordinates in the TIMCOORD time frame (see CAL documentation, section 5.2.4.5.4)

3.3.2.2.2 Output

`PARAMETER` (array) name of parameter to be checked

`SELECT` (array) `selectlib` expression that describes the valid range

3.3.2.3 Used CAL State Variables

instrument, ccdChipId

3.3.2.4 List of CCF Components used

- `HKParmInt`

3.3.3 CAL_getBoresightMatrix

Bore-sight misalignment matrix

3.3.3.1 Procedure

A transformation from instrument reference system (CAMCOORD2) to the spacecraft reference system (SACOORD) by using the the Euler angles given in CCF `BoreSight`.

The time dependency prepares for future use, when parameterizations of time dependent misalignments (e.g. due to relaxations after slew and/or perigee passages) become available.

3.3.3.2 Calling Parameters

3.3.3.2.1 Input

`time` event time coordinates in the TIMCOORD time frame (see CAL documentation, section 5.2.4.5.4)

3.3.3.2.2 Output

- misalignment matrix

3.3.3.3 Used CAL State Variables

instrument, temperature (TBC which H/K parameters), observationStartTime

3.3.3.4 List of CCF Components used

- `BoreSight`

3.3.4 CAL_rawXY2mm

pixel coordinates to physical position

3.3.4.1 Procedure

CCF `LinCoord` holds the positions of distinct reference points of the CCD's (at reference temperature) and the directions in which the counting of `rawX` and `rawY` increases. The size of each pixel is stored in CCF `MiscData`, and also a thermal expansion coefficient.

Using these inputs, together with (`rawX,rawY`), the location of each pixel can be given in focal plane coordinates as a function of temperature. The shift of CCD's due to thermal contraction/expansion is calculated such that each CCD is fixed with its center on the cold bench, and the pixel-size is determined from the thermal expansion coefficient of Si.

3.3.4.2 Calling Parameters

3.3.4.2.1 Input

`rawX, rawY` pixel coordinates

`randomize` flag whether the output should be randomized within this pixel

3.3.4.2.2 Output

`Xmm, Ymm, Zmm` pixel position in mm

3.3.4.3 Used CAL State Variables

instrument, ccdChipId, ccdNodeId, ccdModeId, temperature

3.3.4.4 List of CCF Components used

- `LinCoord`
- `MiscData`

3.3.5 CAL_mm2rawXY

inverse of the above; similar interface.

3.3.6 CAL_rawX2chipX

3.3.6.1 Procedure

Conversion of `rawX` pixel indices to chip oriented coordinate system. The values are completely dependent on the CCD clocking mode, so the data are stored in ModeParams

3.3.6.2 Calling Parameters

3.3.6.2.1 Input

`rawX` pixel coordinates

3.3.6.2.2 Output

`chipX` chip oriented pixel index system

3.3.6.3 Used CAL State Variables

ccdNodeId, ccdModeId

3.3.6.4 List of CCF Components used

- `ModeParams`

3.3.7 CAL_chipX2rawX

3.3.7.1 Procedure

Conversion of chip oriented pixel indices to node oriented coordinate system.

3.3.7.2 Calling Parameters

3.3.7.2.1 Input

`chipX` chip oriented pixel index system

3.3.7.2.2 Output

`rawX` pixel coordinates

3.3.7.3 Used CAL State Variables

ccdNodeId, ccdModeId

3.3.7.4 List of CCF Components used

- `ModeParams`

3.3.8 CAL_getBadPixelList & CAL_getBadPixelMap

`CAL_getBadPixelList` — obtain list of bad pixels
`CAL_getBadPixelMap` — bad pixel list as 2D map

3.3.8.1 Procedure

Bad pixels effectively remove useful focal plane area from an accumulation. The calculation of exposure maps, arfgen etc. need to have this data passed to them in the form of a map or a list.

A secondary use (currently only forseen for PN,and now implemented as a placeholder) is to determine the effect of bad pixels on the CTI correction: The actual *amount* of signal generated in a bad pixel is required as well. This is handled with an additional column that contains a value which is a proxy for the number of electrons generated per second so that a mode dependence of frame time can be accommodated. (An open issue is how implement newly discovered bad pixels for this point.) Also unique for PN is an extension which includes an *offset* value for selected columns. This is the offset applied for "warm" columns to reduce telemetry load, and which must then be removed in ground processing to get the correct energy scale.

The routine extracts from the *BadPix* CCF file integer arrays for `RAWX, RAWY, type` and `yextent` and a logical array showing whether the pixel has been previously identified and up-linked to the instrument for removal from the data stream. For the PN only (*not yet implemented*) a final real array is extracted containing the generation rates per second of each defect. This is converted via the frame time from CALgetModeParameters into PHA units (so that the data can be used in the CTI correction routine where necessary)

An example of use could be that the user needs to know what fraction of an extended region selected to make an Ancillary Response File (ARF) might be subject to bad pixels. In such a case *ARFGEN* would enquire via. size() of the array returned versus size() of the ARF region to correct for effective area. For point sources the weighted contribution of the location of a bad pixel must be removed from the encircled energy calculation.

Finally, in some modes of the instruments there is no up-loading of the bad pixel tables, so that the file has to be made mode dependent, via. the use of a column called MODE_N/A. This is also used in cases where a mode dependency in incorrect databases between on-board versions can be handled by the file.

We now intend to make the CCF file contain only the up-linked pixels, so as to avoid confusion. To detect low level bad pixels present in the observers' data sets, they should re-run the pipeline to locate new defects, and flag out the locations. These defects will be at an occurrence level ≤1%.

3.3.8.2 Calling Parameters

3.3.8.2.1 Input

`TYPE` type of bad pixels that is requested from CCF (see CCF `BadPix`, section 4.4.5 for a description of allowed values) - optional

`rawX0/rawY0` optional starting location of the readout window of the mode

`rawXsize/rawYsize` optional size of the readout window of the mode

3.3.8.2.2 Output

The following are returned from the BadPix file to create a list (CAL_getBadPixelList) or 2-d map of any bad-pixel type (CAL_getBadPixelMap)

`rawX/rawY` array of locations of bad pixels in PIXCOORD

`TYPE` list of bad pixel types as given by the CCF component `BadPix`

`YEXTENT` length of the bad pixel occurrence along the CCD column

`UPLINKED` boolean array for whether the defect is already uplinked to the spacecraft

`N_SAT_IMP` for the PN case the placeholder value returned for the bad pixel generation rate

`MODE_N/A` a numerical list of instrument readout nodes for which any bad pixel is not applicable

For the list, the user can subsequently filter on status, type and magnitude to obtain a list of pixels which are within any selection range

For the map, can likwise filter before producing a map (for example to be used as mask) of defective pixels

3.3.8.3 Used CAL State Variables

instrument, ccdChipId, ccdNodeId, mode, temperature (TBC)

3.3.8.4 List of CCF Components used

- `BadPix`

3.3.9 CAL_mosCTIcorrection

3.3.9.1 Procedure

The current MOS procedure envisages a less complicated correction mechanism than for the PN. However as the physical principles are the same it is plausible that MOS routine may need revising at some future date to more nearly approximate the PN.

The CTI correction applies a scalar correction to the event list set, on an event by event basis, in order to correct for loss of charge which has occurred in reading out the CCD array.

Note that the *EMENERGY* task just requires the PHA values be corrected. There is no flag or other data carried forward to the RMF-generating routine for example, to account for a "noise" factor in the correction process.

We expect to correct for a fixed amount of charge loss, and a fractional amount of charge loss, which depends on the number of row and column transfers completed. The early analysis of ground calibration data shows some possible detailed effects that need to be accounted for. For example:

- optical background - with the OPEN filter position, a significantly more efficient charge transfer resulted. This must be due to the filling of traps by the continually circulating small level charges generated by optical photons.
- a count - rate effect where the CTI may depend on the filling of traps by precursor events. Although this is explicitly handled for the PN camera we envisage in the MOS only to use some local count rate criteria to modify the CTI values. This is not yet implemented.
- There seem to be (rare) cases where a localised trap creates a step-wise increase in charge loss. If the CCF components accurately give a fixed loss component per column, this can mimic loss of extra charge from some point in the serial register (although then a node dependency is required), and also this accounts for possible spatially varying charge loss factors in peripheral columns for example. At present the data do not seem to warrant a description of additional step-wise loss occurring at an individual row.

Note that the intended use of the correction occurs in the pipeline at *EMENERGY*, so that the rawX,rawY attributes are related to the co-ordinates range (1,600), whereas if the same routine were to be used to correct diagnostic data, then a conversion is needed. This would be invoked on looking for the PUT_XY flag setting. At the moment the latter case is hardly conceived as needing some CTI correction therefore ignored and assume a future SEPARATE CAL routine might be needed. Use in *EMENERGY* also implies we can only accurately correct *BEFORE* a modification for a background light contamination has occurred, and also that in principle we should correct ALL E_i components of an event, and not just the central pixel value.

Additionally we note that in principle CTI may change with mode due to the differences in clock timings employed, so that a mode dependence in the CAL state and the CCF files must be carried (TBC from ground data).

Note in the current context of applying the correction to a whole array there is not a concept of handling a frame-dependent count rate dependence to modify the CTE values. The best method for doing this should be established. For example should *EMENERGY* actually ingest a frame file, and calculate from that some count rate estimate - not easy if the dependence is really column dependent.

Algorithm:
A detailed in-flight trend-analysis of the MOS Charge Transfer Inefficiency (CTI), using the two main emission lines (Mn and Al) from the internal calibration sources, spanning three years of operations has shown that :

(1) at a given energy the energy loss per transfer due to CTI (Charge Transfer Inefficiency) is a linear function of time, in time intervals between major discontinuities due to solar flares or change of CCD operating temperature.

(2) the energy losses per transfer scale as a power law function of energy, by comparing the time evolution (degradation) of the CTI at Mn (6keV) and Al (1.5 keV) energies.

Assuming that the energy losses at other energies can be extrapolated from these two energies (Al and Mn emission lines from the internal calibration source), the MOS CTI has been successfully modeled and coded as follows in the CAL CtiCorrector.cc with ALGOID = 2 (in the latest set of CCFs) for a pixel at CCD coordinates (RawX,RawY) :

$eOut = eIn + rawY * ctiY + rawX * ctiX - offset(rawX, rawY)$ with :

$ctiX = (serial[0] + serial[1] * deltaT) * eIn^{(serial[2])}$

$ctiY = (parall[0] + parall[1] * deltaT) * eIn^{(parall[2])}$

where :

- eIn is the uncorrected PHA value.
- eOut is the CTI-corrected PHA value.
- offset(rawX,rawY) are energy offsets of individual column segments.
- deltaT is the time difference between launch and the observation date, expressed in seconds.
- all other serial[i] and parall[i] coefficients are CCD dependent and stored in the CCF MOS CTI extensions CTI_EXTENDED. from version 8 onwards, using ALGOID = 2. (Previous CCF versions use ALGOID = 0 and 1)

The transfer losses are dominated by the parallel CTI. Its typical power law dependence with energy (PHA) of the order of 0.65 for all 14 CCDs (parallel[2] coefficients) at the CCD operating temperature of -100C, but seems to have dropped to lower values (0.5 or lower), since the detectors were cooled down to -120C in revolution 533.

The elapsed time since launch is computed from the date indicated in the FITS header keyword REF_DATE of the first binary extension.

The plots of MOS parallel and serial CTI trends since launch for all CCDs at Mn and Al energies are available on the internal web at:
`http://xmm.vilspa.esa.es/ xmmdoc/MOS/mos_cti.html`

3.3.9.2 Calling Parameters

3.3.9.2.1 Input

rawX, rawY - arrays with CCD pixel co-ordinates in the PIXCOORD reference frame

E - arrays with uncorrected energies referring to the raw measured signal in ADU's. Note from above I suspect we need to convert all E1,E2,E3,E4 values (TBC) because the CTI correction may be proportionally different in the smaller peripheral charge cloud packets than in a combined event.

3.3.9.2.2 Output

PHA - PHA corrected values corresponding to the input E values.

3.3.9.3 Used CAL State Variables

instrument,ccdChipID,ccdNodeId, temperature, ccdModeId

observationStartTime - we assume that CTI will change with time due to increasing radiation damage. We assume that this may be characterisable as a function of time, so that each component CTIn will have an associated dCTIn/dt, so that based on a reference date in the CCF file

3.3.9.4 List of CCF Components used

mosCTI

3.3.10 CAL_getmosOffset

3.3.10.1 Procedure

Returns a default offset map when the offset is not calculated on-board. Based on ground or in-flight diagnostic data, some default image files will be stored. With accuracyLevel = LOW, the routine just returns the offset map. With accuracyLevel=HIGH, the routine returns the deviations from the local row and column average.

3.3.10.2 Calling Parameters

3.3.10.2.1 Input

rawX, rawY

rawX,rawY - pixel location for which the offset value is required

accuracyLevel

3.3.10.2.2 Output

offset deviation of pixel signal from the predicted row/column offset values for the requested CCD location, or the actual offset value. .

3.3.10.3 Used CAL State Variables

instrument, ccdChipId, FilterId, ccdModeId, ccdNodeId

3.3.10.4 List of CCF Components used

DarkFrame

3.3.11 CAL_mosgainCorrect

Converts PHA values to PI via. a gain correction.

3.3.11.1 Procedure

The function converts an array of PHA values into a common PI scale, by accounting for gain of the current CCD readout node. The second extension of the ADUconv file contains offset and gain coefficient values that allow the PI channels to be calculated in terms of approximate eV, according to

```
PI = a + b*PHA + c*PHA*PHA
```

3.3.11.2 Calling Parameters

3.3.11.2.1 Input

PHA (int) - PHA values to be corrected (needs to be converted into real), as determined from the E_i values by task $EMEVENTS$.

PATTERN - pattern type is used to invoke slightly modified coefficients. If no pattern is invoked the default for single pixel coefficients is used. Otherwise, the pixel size identifier for 1, 2,3 or 4 pixel events is found according to integer arithmetic 1+(PATTERN+3)/4

3.3.11.2.2 Output

PI (float) - gain corrected PI

3.3.11.3 Used CAL State Variables

instrument, ccdChipID, ccdNodeID,FPAtemperature,observationStartTime,mode

Notes:

1. FPAtemperature or EMAE temperatures in principle may change the gain of the pre-amplifiers, but TBD the temperature should be stable At the moment we have gathered evidence that the temperature does change and modifies the gain according to a component of the form *dGain/dtemp*. the detailed analysis of this feature is still underway
2. Mode may in principle cause a change in gain if the readout amplifier sampling sequence changes. It is TBD if we cope with changes in the mode via an expanding list of modes, or via. a time dependent association between logical RPS mode name and readout sequence
3. observationStartTime - in addition to the above there will be the possibility of secular changes in gain (radiation effects?) requiring a modification in assumed gain values

The current implementation has no account of these items.

3.3.11.4 List of CCF Components used

ADUconv

3.3.12 CAL_getEventPatterns

Returns the current set of event patterns that are used by the EPIC MOS event detection unit. These data are used in SAS for emulation of this on-board processing in tasks such as *emdiag*, which operate on diagnostic data only. In addition the data can be used to reject cosmic rays on the basis of pattern type.

3.3.12.1 Procedure

The routine simple makes a transparent read to return the following data:

1. The *eduThreshold*, which is the value in ADU's of the threshold currently applied to events. Because there may be 2 different ADU threshold (for inner and outer CCDs respectively) per mode, then each extension will have two attributes describing this
2. The patterns data which are a 3-d array comprising the index of pattern identifying number, and a 5×5 array describing the significance of each pixel within the patterm

Note that the outer CCDs should remain in a Full Window mode of readout, so that except for any change in their EDUThreshold values with mode, then only the pattern IDs of the central CCD are expected to change with mode (and currently only for the FAST mode). For completness and possible future developnent there is a separate extension per MODE for the patterns library of this central CCD.

3.3.12.2 Calling Parameters

3.3.12.2.1 Input

arg

3.3.12.2.2 Output

eduThreshold , which is the value in ADU's of the threshold currently applied to events. Because there may be 2 different ADU threshold (for inner and outer CCDs respectively) per mode, then each extension will have two attributes describing this

patterns - data which are a 3-d array comprising the index of pattern identifying number, and a 5×5 array describing the significance of each pixel within the pattern

3.3.12.3 Used CAL State Variables

instrument, ccdChipID, modeID

3.3.12.4 List of CCF Components used

PatternLib

3.3.13 CAL_getFilterTransmission

3.3.13.1 Procedure

The task returns the X-ray transmission factor for the chosen filter. The nominal function is expected to return a vector of energy values, by consulting the FilterTransX file for the appropriate extension and interpolating the between transmission values stored for neighbouring energy points.

The file is made with a coarse (default only one) set of spatially sampled points.

3.3.13.2 Calling Parameters

3.3.13.2.1 Input

Energy,theta,phi

Energy The array of requested energy values for the energy bins in which the transmisison is desired (in keV)

theta array specifying position in focal plane in TELCOORD

phi array specifying position in focal plane in TELCOORD

3.3.13.2.2 Output

Transmission

Transmission - an array of real values between 0.0 and 1.0 describing the fraction of photons transmitted by the filter at the requested energy bins, and spatial location

3.3.13.3 Used CAL State Variables

FilterID The name of the filter requested. This can be one of `OPEN`, `THICK`, `THINA`, `THINB`, `MEDIUM` and maps to the column name storing the required transmission values

instrument

3.3.13.4 List of CCF Components used

FilterTransX

3.3.14 CAL_getQuantumEfficiency

3.3.14.1 Procedure

The task returns the X-ray detection efficiency for the chosen area on the CCD. The nominal function is expected to return a vector of values for energy bins, by consulting the QuantumEff file for the appropriate spatial limits and interpolating the between efficiency values stored for neighbouring energy points. A sub-set of pattern types can be selected (for example single pixel events, bi-pixels etc.). These can read from individual columns. (TBD is one column for every pattern or collected in sub-groups: i.e. in the MOS case, patterns 1-4 are bi-pixel events, but they could be stored combined as one column, or the CAL can implement the intelligence to sum up the 4 columns. *Ad interim* we use a separate column in the CCF per individual pattern number.)

3.3.14.2 Calling Parameters

3.3.14.2.1 Input

Energy, Raw_X, Raw_Y,PatternNos.

Energy - the array of energy values for the energy bins at which the efficiency is required. The Energy column of the CCF file will be checked to provide data to interpolate to the exact energies required

Raw_X - location within the CCD at which the efficiency is requested (in case there is spatial variation)

Raw_Y - location within the CCD at which the efficiency is requested (in case there is spatial variation)

PatternNos - list of pattern identifiers that comprise the total sub-class of events required

3.3.14.2.2 Output

Efficiency - An array of detection efficiency, real values between 0.0 and 1.0 for all events which should be detectable at the energy bins, spatial location and event pattern sub class required

3.3.14.3 Used CAL State Variables

ccdChipID - the identifier for the CCD - each CCD stored as a separate extension?

ccdNodeID - the identifier for the CCD node

instrument

3.3.14.4 List of CCF Components used

QuantumEff

3.3.15 CAL_getLargePatternSize

3.3.15.1 Procedure

The task returns the average pattern size of pixels with a pattern size greater than 12 for a given input energy or energy range. This value may be useful in the computation of pile-up effects.

The average size is calculated from the formula:

$$Size = C_0 + C_1.E + C_2.E^2 + C_3.E^3 + C_4.E^4 \quad (1)$$

where E is photon energy in keV and C_n are coefficients read from the QuantumEff CCF.

3.3.15.2 Calling Parameters

3.3.15.2.1 Input

Energy or Energy range.

Energy - photon energy in eV (not in keV) or a low and high photon energy in eV over which to integrate.

3.3.15.2.2 Output

averagePatternSize - the average size of events with patterns greater than 12.

3.3.15.3 Used CAL State Variables

ccdChipID - the identifier for the CCD.

instrument

3.3.15.4 List of CCF Components used

QuantumEff

3.3.16 CAL_getContamination

3.3.16.1 Procedure

The task returns the X-ray transmission factor for the suspected contamination layer. For each energy bin the transmission is obtained by interpolating the between transmission values stored for neighbouring energy points. **Not yet implemented**

By invoking the time, the predicted depth of contamination layer is first calculated from some TBD coefficients, stored in a depth extension. *Ad interim* we assume a linear growth from a reference point of form:

```
Depth = Growth  .  (Time - Reference Time)
```

The `attenuation` extension then has linear attenuation coefficients versus energy appropriate for the witches brew building up on the CCDs. From the depth value, a transmission factor is thus obtained.

3.3.16.2 Calling Parameters

3.3.16.2.1 Input

Energy

Energy The requested energy bin in which the transmisison is desired (in keV)

3.3.16.2.2 Output

Transmission

Transmission - a real value between 0.0 and 1.0 describing the fraction of photons transmitted by the contamination at the requested energy bin

3.3.16.3 Used CAL State Variables

instrument, ccdChipID, observationStartTime

3.3.16.4 List of CCF Components used

Contam

3.3.17 CAL_pnCtiCorrect

3.3.17.1 Procedure

This routine corrects for a list of EPIC PN events the energy values for the effect of CTI. The effect is position dependent, i.e., the routine receives spatial event coordinates in addition to the uncorrected energies. The routines operates on real-valued arrays, i.e., integer-valued PHA channels from ODF data sets must be converted to corresponding real arrays with routine.

At the moment this procedure implements complex code supplied by the PI team, and does not make use of normal CAL and CCF calls. Time is used to reference any changing component of CT with radiation damage.

3.3.17.2 Calling Parameters

3.3.17.2.1 Input

rawX, rawY - the rawY value increases towards the centre of the PN array, and thus charge is increasingly lost with distance. There is a column to column dependency of charge loss, and this is invoked with the rawX parameter

frame - needed to keep track of the possibility of precursor events in the same column and CCD readout frame as the subject event.

eIn array with uncorrected energies of offset and gain - corrected ADU

3.3.17.2.2 Output

eOut - array with computed CTI-corrected energy values [ADU]

3.3.17.3 Used CAL State Variables

instrument, ccdChipId, temperature, observationStartTime

3.3.18 CAL_pnGainCorrect

3.3.18.1 Procedure

Uses hardware-PI-supplied code to fix the gain of the ADU channels. The gain depends on the readout node, which identified directly with rawX.

3.3.18.2 Calling Parameters

3.3.18.2.1 Input

rawX array with X-pixel coordinates in the PIXCOORD1 reference frame; this is used to determine the readout node

eIn - input array with PHA channels to correct for gain

eTot - array with amplitude values proportional to the totally released charge of the events

3.3.18.2.2 Output

eOut - array with gain corrected PI channels

3.3.18.3 Used CAL State Variables

instrument, ccdChipId, temperature

3.3.18.4 List of CCF Components used

ADUConv

3.3.19 CAL_pnReEmissionThreshold

3.3.19.1 Procedure

A hardware-PI-supplied code which computes the re-emission threshold value of an event following a detected main event. Checks for events following along with higher value of rawY.

3.3.19.2 Calling Parameters

3.3.19.2.1 Input

energy - energy value of the main event in ADU.

rawY - Y-pixel coordinate of the main event in the PIXCOORD1 reference frame

3.3.19.2.2 Output

amplitude - computed amplitude value of the re-emission threshold corresponding to the input parameters and the currently set temperature.

3.3.19.3 Used CAL State Variables

instrument, temperature

3.3.19.4 List of CCF Components used

ADUconv

3.3.20 CAL_getBackgroundMap

3.3.20.1 Procedure

This returns an event list of the background data, as a generic task.

For EPIC this CAL routine should be used only in cases where the background estimated from source-free regions of an exposure cannot be used (e.g. very extended sources, small windows extent). The background is dependent on a large number of factors, especially user-defined data selections, such as energy range, event selections etc..

Since launch it has been noted that the background depends to a significant degree on the effect of soft proton flares. These flares are variable in intensity and spectrum, so a generic background file is unlikely to have a high fidelity reproduction of *actual* background experienced.

The approach suggested ad interim is to have large event files extracted from quiet intervals.

3.3.20.2 Used CAL State Variables

ccdCHIPID,instrument,time

3.3.20.3 List of CCF Components used

bckgd

3.3.21 CAL_getEbounds

returns the correspondence between PI channels and energy ranges. The raw PHA channels are converted to approximate energy values in the form of PI channels. However the response matrix distributions are provided also in PI channels, but typically in a fixed set of energy ranges. TBD, these might be semi-log spaced to account for the changing CCD energy resolution and to the reduced photon statistical accuracy at the highest energies. Therefore the upper and lower energy limits per response bin are to be converted between PI channel and energy.

3.3.21.1 Procedure

A direct call to the Redist file is used to extract the Ebounds extension, which retrieves min and max energy ranges per channel.

3.3.21.2 Calling Parameters

3.3.21.2.1 Input

arg

3.3.21.2.2 Output

channels - a 1-d array containing the PI channels in increasing order

E_{min} 1-d array containing lower energy bounds corresponding to PI channel numbers. $E_{min(i)}$ is th lower bound of PI channel i

E_{max} 1-d array containing upper energy bounds corresponding to PI channel numbers. $E_{max(i)}$ is the upper bound of PI channel i

3.3.21.3 Used CAL State Variables

3.3.21.4 List of CCF Components used

Redist

3.3.22 CAL_getEpicMOSRedistribution

Returns the EPIC MOS redistribution function. The processing described below is coded in the CALMOSALGO package which is called from the C++ CAL Atom, CanonicalRedist.

3.3.22.1 Procedure

The described procedure returns the one-dimensional response distribution in PI channels, based on an input energy. The redistribution function of the MOS cameras comprises a main peak (roughly centred on the input photon energy); a lower-energy secondary peak, whose relative magnitude is inversely correlated with the photon energy, and a noise term which extends down to the detection threshold energy. Details of the current calculation are given in the CCF release note, CAL-SRN-272.

CanonicalRedist retrieves a set of parameters from the appropriate MOS redistribution CCF and passes these to the CALMOSALGO initialisation routine *CALMOSALGO_initRmfRow*. The routine *CALMOSALGO_rmfRow* may then be used to calculate the response, in each PI channel bin, for a given input energy and event pattern selection.

The redistribution parameters have been calibrated as a function of:

Observation date - A gradual decrease in the instrumental resolution with time has been observed, with a discrete jump at the epoch of CCD cooling (Rev. 532). Responses have been divided into steps, with a number of separate, time-dependent, CCF elements, relating to certain ranges of revolutions.

Spatial region - On the central CCD (1), a patch has developed, centred on the boresight, where the resolution has become degraded. Three spatial regions have been defined, as an on-patch, wings-of-patch and off-patch selection.

Observing mode

Event grade - The instrumental resolution is most sharp when the electrons generated by an incoming photon are collected in a single pixel. It degrades progressively as events are spread over multiple pixels. A separate calibration is available for single-pixel events (pattern=0) and multi-pattern events (patterns 0–12).

3.3.22.2 Calling Parameters

3.3.22.2.1 Input

Energy - the photon energy (eV)

Pattern - event patterns

3.3.22.2.2 Output

response - a one dimensional array containing the response in each PI channel bin

3.3.22.3 Used CAL State Variables

Instrument

3.3.22.4 List of CCF Components used

Redist

3.3.23 CAL_getEpicPnRedistribution

Returns the EPIC-pn redistribution function. The processing described below is coded in the CALPNALGO package which is called from the C++ CAL Atom, CanonicalRedist.

3.3.23.1 Procedure

The described procedure returns the one-dimensional response distribution in PI channels, based on an input energy.

CanonicalRedist retrieves a set of parameters from the EPIC-pn redistribution CCF and passes these to the *CALPNALGO_resp* routine. This routine then calculates the response, in each PI channel bin, for an input photon energy, an event pattern selection and a RAW-Y (row) position on the CCD.

The redistribution parameters have been calibrated as a function of:

Observing mode - The shape of the response has a significant dependence on the observational mode.

Event grade - The instrumental resolution is most sharp when the electrons generated by an incoming photon are collected in a single pixel. It degrades progressively as events are spread over multiple pixels. A separate calibration is available for single-pixel events (pattern=0) and double-pixel events (patterns 1–4).

chipY - The act of transferring charge across the pixels to the readout introduces an extra spread in the detector response. This works such that pixels close to the camera centre have poorer resolution than those towards the edge of the field-of-view. The calibration is based on the chip-Y coordinate.

3.3.23.2 Calling Parameters

3.3.23.2.1 Input

Energies - an array of photon energies (eV)

Pattern - event patterns

chipY - the row number on the CCD (from 1-199).

3.3.23.2.2 Output

response - a one dimensional array containing the response in each PI channel bin

3.3.23.3 Used CAL State Variables

Instrument

3.3.23.4 List of CCF Components used

Redist

3.3.24 CAL_getCalSourceSpec

Not yet implemented

3.3.24.1 Procedure

The intensity distribution map will be stored in CCF `CalSourceData` (sections 4.3.7 and 4.3.8), in its first extension. A second extension will hold the energy spectrum.

The intensities of the calibration sources will be corrected for the half life of the source, via. a function of the form $I=I_0\exp(-\tau t)$. Where, τ ($\log_e/T_{1/2}$) is an attribute in the CCF file, and time t is the elased time between observationStartTime and reference time which is also an attribute in the CCF file.

The function returns a energy spectrum in the range from `E_LOW` to `E_HIGH` for a given CCD, and 0-12 pattern selection, for a duration of exposure .

3.3.24.2 Calling Parameters

3.3.24.2.1 Input

`E_LOW` lower limit of energy range

`E_HIGH` upper limit of energy range

`T_exp` exposure duration

3.3.24.2.2 Output

`ENERGY` (array): the centers of the energy bins

`Counts` (array): the counts per energy bin

3.3.24.3 Used CAL State Variables

instrument, ccdChipId, ccdNodeId, observationStartTime

3.3.24.4 List of CCF Components used

- `CalSourceData`

3.3.25 CAL_getCalSourceMap

Not yet implemented

3.3.25.1 Procedure

The intensity distribution map will be stored in CCF `CalSourceData` (sections 4.3.7 and 4.3.8), in its first extension. A second extension will hold the energy spectrum.

The intensities of the calibration sources will be corrected for the half life of the source, via. a function of the form I=I_0exp-(τt). Where, τ ($\log_e/T_{1/2}$) is an attribute in the CCF file, and time t is the elased time between observationStartTime and reference time which is also an attribute in the CCF file.

The function returns an intensity map for a given CCD, and 0-12 pattern selection.

3.3.25.2 Calling Parameters

3.3.25.2.1 Input

Time - exposure duration

3.3.25.2.2 Output

image - an integer array at CCD pixel resolution of number of monopixel events expected in the exposure

3.3.25.3 Used CAL State Variables

instrument, ccdChipId, ccdNodeId, observationStartTime

3.3.25.4 List of CCF Components used

- `CalSourceData`

3.3.26 CAL_getPSFsmear

Not yet implemented

3.3.26.1 Procedure

Modifies the PSF map, by applying the effect of EPIC charge transfer smearing. This should be called *BEFORE* any pile-up correction (see 3.3.31) in order to ensure that the correct fraction of INPUT detected events are re-distributed.

The routine consults the ModeParams files for a smearing fraction, and takes the integrated surface brightness in each CCD column, and reduces the brightness according to this fraction, then redistributes the smeared fraction evenly over the column.

3.3.26.2 Calling Parameters

3.3.26.2.1 Input

MapIn - input PSF image array (from CAL_PSFMap for example)

3.3.26.2.2 Output

MapOut - output PSF image array

3.3.26.3 Used CAL State Variables

instrument, ccdModeId,

3.3.26.4 List of CCF Components used

ModeParams

3.3.27 CAL_getFOV

3.3.27.1 Procedure

Returns an array for pixels in or out of the field of view. These pixels can be shadowed by the edge of filter wheel, by the staggered ranks of MOS CCDs, PN rows declared inactive etc.

3.3.27.2 Calling Parameters

3.3.27.2.1 Input

3.3.27.2.2 Output

-

3.3.27.3 Used CAL State Variables

instrument

3.3.27.4 List of CCF Components used

lincoord (fov extension)

3.3.28 CAL_pnMedianMap

3.3.28.1 Procedure

Returns a median map to show the noise distribution at low-energy.

3.3.28.2 Calling Parameters

3.3.28.2.1 Input

3.3.28.2.2 Output

`MEDIANMAP` (image): the noise at each pixel

3.3.28.3 Used CAL State Variables

instrument, ccdChipID, modeID

3.3.28.4 List of CCF Components used

- `REJECT`

3.3.29 CAL_pnOffsetCorrValues

3.3.29.1 Procedure

Returns correction values as a function of photon energy.

3.3.29.2 Calling Parameters

3.3.29.2.1 Input

3.3.29.2.2 Output

`HLO` the low bounds of the energy bins

`HLO` the upper bounds of the energy bins

`CORRVALS` the correction values

3.3.29.3 Used CAL State Variables

instrument, ccdChipID, modeID

3.3.29.4 List of CCF Components used

- `REJECT`

3.3.30 CAL_pnNoiseMap

3.3.30.1 Procedure

Returns a noise map as a function of RAW-X position.

3.3.30.2 Calling Parameters

3.3.30.2.1 Input

3.3.30.2.2 Output

`RAWX` array of RAW-X positions for which noise images are defined.

`NoiseMap` A noise image for each RAW-X position.

3.3.30.3 Used CAL State Variables

instrument, ccdChipID, modeID

3.3.30.4 List of CCF Components used

- REJECT

3.3.31 CAL_PSFPileUp

Not yet implemented

3.3.31.1 Procedure

Returns a modified PSF map based on the effect of core suppression due to pile-up. The suppression is mildly energy and pattern dependent, but for the usage of supplying a PSF brightness distribution (for checking if a source is truly point-like, or as an aid to determining if the pileup is sufficent to give concern for spectral analysis), then the approximations employed herein may be sufficient.

Supplying a total input events/point source/second is the preferred activation input. This is converted to input events/point source/CCD frame according to the frame time in ModeParams. The input PSFimage is normalised to unity, and from this the localised pixel-by-pixel count rate per frame is established.

The calculation depends on the pattern types selected, with progressively more complex arithmetic for the larger patterns (see J Ballet *Ast Ap Suppl Ser* v 135 pp371-381).

The output intensity (μ) in each case is modelled as follows, where λ is the input intensity per pixel :

Mono-Pixels

$\mu_1 = (1 - \exp(-\alpha_1 \lambda)) exp(-\gamma_1 \lambda)$
where
$\gamma_1 = 9 + 3\alpha_2 + 6\alpha_3 + 7\alpha_4$
and the α_i are the fractional pattern occurrences taken from the pattern library CCF, but α_2 corresponds to patterns 1,2,3 and 4 , α_3 corresponds to patterns 5,6,7 and 8 and α_4 corresponds to patterns 9,10,11 and 12.

Bi-Pixels

$\mu_2 = 2(p_2 + (1 - p_2)p_1^2) exp(-(\gamma_2 - 2\alpha_1 - 0.5\alpha_2)\lambda)$
where
$\gamma_2 = 12 + 3.5\alpha_2 + 7\alpha_3 + 8\alpha_4$
$p_1 = 1 - exp(-\alpha_1 \lambda)$
$p_2 = 1 - exp(-\alpha_2 \lambda/2)$

Tri-Pixels

$\mu_3 = 4(1 - (1 - T_3)exp(-\alpha_3\lambda/4))exp(-\gamma_3\lambda)$
where
$T_3 = p_1^3/3exp(-\alpha_2\lambda) + 0.5p_1p_2exp(-\alpha_2\lambda/2) + 0.75p_2^2$
$\gamma_3 = 15 + 4\alpha_2 + 7.75\alpha_3 + 9\alpha_4 - 3\alpha_1 - \alpha_2 - 0.25\alpha_3$

Quad-Pixels
$\mu_4 = [1 - exp(-\alpha_4\lambda)(1 - Q_3)]exp(-\lambda(\gamma_4 - 1 - 3\alpha_1 - \alpha_2))$
where
$Q_3 = 1 - (1 - Q_2)exp(-\alpha_3\lambda) - 4(1 - exp(-\alpha_3\lambda/4))exp(-\lambda(\alpha_1 + \alpha_2 + 0.75\alpha_3))$
$Q_2 = p_1^4exp(-2\alpha_2\lambda) + 4p_1^2p_2exp(-3\alpha_2\lambda/2) + 4p_1p_2^2exp(-\alpha_2\lambda) + p_2^2(2 - p_2^2)$
$\gamma_4 = 16 + 4\alpha_2 + 8\alpha_3 + 9\alpha_4$

3.3.31.2 Calling Parameters

3.3.31.2.1 Input

MapIn - input PSF image map in CCD pixel units (including readout smearing effects if any). It must be centered correctly to sub-pixel location, especially if the PN pile-up is to be predicted correctly

rate_in - input counts/second expected in the point source

Patterns - list of pattern numbers used. A valid input range is 1,2,3,4 where the number represents the maximum sized pattern type used (i.e. if mono-, bi- and tri- pixel patterns are to be used, then the number 3 is input)

3.3.31.2.2 Output

MapOut - modified PSF image map

rate_out - summed PSF image map intensity times the input rate

3.3.31.3 Used CAL State Variables

instrument,ccdModeId

3.3.31.4 List of CCF Components used

ModeParams,EventPatterns,

3.4 RGS Related CAL Routines

Many calibration quantities depend on basic instrument operation parameters such as the temperature of the focal plane. It is to be expected that items such as temperatures will be changed over the coarse of the mission, but this will only occur at certain infrequent points in time. Such a change of operation parameters influences the instrument calibrations in many areas, and in a complex way. The decision of change of operation parameters is triggered by a certain degradation of instrument performance. Given the complexity of a possible parameterization, together with the somewhat unpredictable degradation of performance, and the infrequent changes of parameter settings, does not justify to fully parameterize this dependence as part of the CCF and the CAL. Therefore these changes will be incorporated with updates of both CCF and CAL functions.

The consistency, however, of choice of operating parameters (available from telemetry) and validity of CCF files will be checked by the CAL functions that are accessing the relevant CCF constituent. In order to indicate the parameters that should be checked, they will also be listed in the following per CAL function as input parameters or state variables, but with the appended attribute *(check)*.

As an example, the instrumental bad pixel table depends on operation temperature of the CCD's. The CCF `BadPix` contains only data for one temperature, which is given in a keyword in its header. The CAL functions `CAL_getBadPixelList` & `CAL_getBadPixelMap` use the CAL state variable `temperature` only to check that the current operating temperature is consistent with the CCF constituent `BadPix`. No interpolation is attempted either by `CAL_getBadPixelList` nor by `CAL_getBadPixelMap`. In case of inconsistent temperatures, the user will be informed to obtain an updated version of the CCF in question.

Since the measured temperatures (as obtained from the telemetry) will not necessarily have the same value as the nominal validity temperature which is given in the headers of the respective CCF constituents, a range is defined. Per temperature (the CCD temperature `temperature`, and the temperature of the electronics `FPA_temp`) an allowed range is specified in CCF `MiscData`. This will be used to check the actual temperature versus the one that is required for a certain CCF constituent.

The following possible dependencies of the instrument calibrations are currently neglected as they are at this point in time not considered to cause important effects:

- differential non-linearity of ADC used for the CCD pulse height digitization
- effects of fixed pattern noise during CCD readout
- effects of mode dependent DPP processing (SER, HER, SES)on other items than quantum efficiency

3.4.1 CAL_getBoresightMatrix

Bore-sight misalignment matrix

3.4.1.1 Procedure

A transformation from instrument reference system (CAMCOORD2) to the spacecraft reference system (SACOORD) by using the the Euler angles given in CCF `BoreSight`.

The time dependency prepares for future use, when parameterizations of time dependent misalignments (e.g. due to relaxations after slew and/or perigee passages) become available.

3.4.1.2 Calling Parameters

3.4.1.2.1 Input

`time` event time coordinates in the TIMCOORD time frame (see section 2.2)

3.4.1.2.2 Output

- misalignment matrix

3.4.1.3 Used CAL State Variables

instrument, temperature (check), observationStartTime

3.4.1.4 List of CCF Components used

- `BoreSight`

3.4.2 CAL_rawXY2mm

pixel coordinates to physical position

3.4.2.1 Procedure

CCF LinCoord holds the positions of distinct reference points of the CCD's (at reference temperature) and the directions in which the counting of rawX and rawY increases. The size of each pixel is stored in CCF MiscData, and also a thermal expansion coefficient.

Using these inputs, together with (rawX,rawY), the location of each pixel can be given in focal plane coordinates as a function of temperature. The shift of CCD's due to thermal contraction/expansion is calculated such that each CCD is fixed with its center on the cold bench (thermal expansion of Al), and the pixel-size is determined from the thermal expansion coefficient of Si.

3.4.2.2 Calling Parameters

3.4.2.2.1 Input

rawX, rawY pixel coordinates

NODEID (optional array): if present, it has the same length as the rawX, rawY pairs, and it overwrites the state variable ccdNodeId

randomize flag whether the output should be randomized within this pixel

3.4.2.2.2 Output

Xmm, Ymm, Zmm pixel position in mm

3.4.2.3 Used CAL State Variables

instrument, ccdChipId, ccdNodeId, ccdModeId, OCB, temperature

3.4.2.4 List of CCF Components used

- LinCoord
- MiscData

3.4.3 CAL_mm2rawXY

inverse of the above; similar interface.

3.4.4 CAL_rgsRawPixCorr

3.4.4.1 Procedure

Corrects `rawX` & `rawY` as obtained from the telemetry for windowing and pre-scan, such that they are consistent with the convention used within the SAS, where `rawX` & `rawY` start counting from 1 for the first physical pixel on the CCD. Consequently pre-scan pixels have values of `rawX` < `1`.

Note that nominally the commanding will be such that the first physical bin on the CCD has coordinates (0,0) in the telemetry. If a window is applied by re-programming the clock sequence generator, also supply the correct coordinates will be loaded to the DPP such that the on-board hot pixel lookup table does not have to be re-loaded.

The physical size of the first bin depends on the OCB. E.g. for ODB = 3×3, the fist bin contains only 2 pixels.

3.4.4.2 Calling Parameters

3.4.4.2.1 Input

`rawX, rawY` pixel coordinates

`WINDOWX0, WINDOWY0` windowing information as obtained from telemetry (and stored in the header keywords of the event file of the ODF)

`CSG_ID` clock sequence ID (stored as `CCDCSG` in the ODF)

3.4.4.2.2 Output

`rawX, rawY` corrected pixel coordinates according to the SAS definition

3.4.4.3 Used CAL State Variables

instrument, ccdChipId, ccdModeId (check), OCB (check)

3.4.4.4 List of CCF Components used

- `ClockPatterns` to get the pre-scanning pixels according to `CSG_ID`

3.4.5 CAL_rawX2chipX

3.4.5.1 Procedure

Conversion of `rawX` pixel indices to chip oriented coordinate system.

3.4.5.2 Calling Parameters

3.4.5.2.1 Input

`rawX` pixel coordinates

`NODEID` (optional array): if present, it has the same length as the **`rawX, rawY`** pairs, and it overwrites the state variable ccdNodeId

`WINDOWX0, WINDOWDX` windowing information along X as obtained from telemetry (and stored in the header keywords of the event file of the ODF)

`CSG_ID` clock sequence ID (stored as `CCDCSG` in the ODF)

3.4.5.2.2 Output

`chipX` chip oriented pixel index system

3.4.5.3 Used CAL State Variables

ccdNodeId, ccdModeId

3.4.5.4 List of CCF Components used

- `ClockPatterns`
- `MiscData`

3.4.6 CAL_chipX2rawX

3.4.6.1 Procedure

Conversion of chip oriented pixel indices to node oriented coordinate system.

3.4.6.2 Calling Parameters

3.4.6.2.1 Input

`chipX` chip oriented pixel index system

`WINDOWX0, WINDOWDX` windowing information along X as obtained from telemetry (and stored in the header keywords of the event file of the ODF)

`CSG_ID` clock sequence ID (stored as `CCDCSG` in the ODF)

3.4.6.2.2 Output

`rawX` pixel coordinates

`NODEID` (optional array): if present, it has the same length as the `rawX, rawY` pairs, and it overwrites the state variable ccdNodeId

3.4.6.3 Used CAL State Variables

ccdNodeId, ccdModeId

3.4.6.4 List of CCF Components used

- `ClockPatterns`
- `MiscData`

3.4.7 CAL_getBadPixelList & CAL_getBadPixelMap

`CAL_getBadPixelList` — obtain list of bad pixels
`CAL_getBadPixelMap` — bad pixel list as 2D map

3.4.7.1 Procedure

3.4.7.2 Calling Parameters

3.4.7.2.1 Input

`TYPE` type of bad pixels that is requested from CCF (see CCF `BadPix`, section 4.4.5 for a description of allowed values)

Nothing specific, depends on efficient interface to tasks.

3.4.7.2.2 Output

- list of locations of bad pixels in PIXCOORD
- list of bad pixel types as given by the CCF component `BadPix`

3.4.7.3 Used CAL State Variables

instrument, ccdChipId, ccdNodeId, OCB (check), temperature (check)

3.4.7.4 List of CCF Components used

- `BadPix`

3.4.8 CAL_rgsGetDarkFrame

Get dark frame map.

3.4.8.1 Procedure

The dark frame map contains all non-uniformities across the CCD surface. It will be shifted additively by `CAL_offsetCorrect` with the offset available either from CCF `ADUConv` or as the output of `rgsoffsetcalc`.

The dark frame will be stored in parameterized form in CCF `DarkFrame` on a per node basis. The functional form is

$$f = a_0 \times e^{a_1(\mathtt{rawX}-1)} + a_2 \times (\mathtt{rawX} - 1) + \frac{(a_3 + b_3)}{2} + b_0 \times e^{b_1(\mathtt{rawY}-1)} + b_2 \times (\mathtt{rawY} - 1)$$

with parameters a_0 to a_3 and b_0 to b_3 that are available from CCF `DarkFrame`.[1]

Modifications of the dark frame due to light leaks from imperfections of the Al-filter will be included.

This routine is mainly used by `CAL_gainCorrect` to obtain the local offset at the position `rawX` & `rawY`.

[1] The values of `rawX` and `rawY` need to be reduced by 1, as the parameterization was performed on ground calibration data, that had a different pixel index notation than is used by the SAS.

3.4.8.2 Calling Parameters

3.4.8.2.1 Input

`rawX, rawY` pixel coordinates: these are used for the offset correction with the dark frame by using `CAL_rgsGetDarkFrame`

`CSG_ID` clock sequence ID (stored as `CCDCSG` in the ODF); a lookup in CCF `ClockPatterns` will be performed to find the associated value of `CSG_Name`, which is then in turn required for the lookup of the index in CCF `DarkFrame`.

3.4.8.2.2 Output

`DARKCUR` dark current at location (`rawX`,`rawY`)

3.4.8.3 Used CAL State Variables

instrument, ccdChipId, ccdNodeId, ccdModeId, OCB (check), temperature (check)

3.4.8.4 List of CCF Components used

- `DarkFrame`
- `ClockPatterns` for the mapping between `CSG_ID` and `CSG_Name` (`CSG_Name` is used for the identifications of the CSG program parameters in CCF `DarkFrame`.)

3.4.9 CAL_offsetCorrect

Subtraction of offset from PHA value

3.4.9.1 Procedure

First the integer PHA input value will be converted to float with randomization depending on whether the calibration state variable `randomize`. The randomization interval will be taken from the keywords `E_RAND_MIN` and `E_RAND_MAX` that are available in CCF `ADUConf`.

The local offset will be subtracted from the PHA value. Offset subtraction is performed by additively shifting the parameterized dark frame (CCF `DarkFrame`, section 4.4.10) with the value stored in `OFFSET` in the second extension of CCF `ADUConv` (section 4.4.4), or with the result of `rgsoffsetcalc`, if available.

The offset is provided by the CCF `ADUCONV` as a parameterization depending on the start of observation time T:

$$\texttt{offset} = \texttt{OFFSET\_SLOPE} \times (T - \texttt{REFTIME}) + \texttt{OFFSET\_CONST} \ ,$$

with `OFFSET_SLOPE`,`REFTIME` and `OFFSET_CONST` being available from the CCF.

Alternatively, the offset can be calculated and interpolated using the queue memory data (if available) with `rgsoffsetcalc`, otherwise the data as stored in the CCF can be used as fall back.

The formula for the surface independent offset correction is

$$PHA0 = PHA - (GRADE) * offset$$

Notes

- For the time being, no temperature dependence will be included, however consistency will be checked between the current temperature and the temperature for which CCF `ADUConv` is valid. If the temperatures are inconsistent, a warning will be issued.
- Offset variations as function of position on the CCD are included in dark frame map (which might additionally contain optical load and electronic noise, once processing of `rgsoffsetcalc` is extended to include this).

3.4.9.2 Calling Parameters

3.4.9.2.1 Input

`rawX, rawY` pixel coordinates: these are used for the offset correction with the dark frame by using `CAL_rgsGetDarkFrame`

`NODEID` (optional array): if present, it has the same length as the `rawX, rawY` pairs, and it overwrites the state variable ccdNodeId

`PHA` (integer): PHA

`time` event time coordinates in the TIMCOORD time frame (see section 2.2)

`SHAPE` number of pixels combined during on-board split event reconstruction

`FPA_temp` temperature of the focal plane assembly; this affects the temperature of the readout electronics.

`CSG_ID` clock sequence ID (stored as `CCDCSG` in the ODF); this will be passed on to `CAL_rgsGetDarkFrame`

Notes

- Aging of the electronics will be implemented by updating of the CCF constituents `ADUConv` & `DarkFrame`
- This CAL routine may also require a dependency on the satellite pointing direction, should a known offset distribution be required to be calculated as a function of SUN aspect angle and/or other bright objects in the FoV (planets).

3.4.9.2.2 Output

`PHAO` (float): offset subtracted PHA value

3.4.9.3 Used CAL State Variables

instrument, ccdChipId, ccdNodeId, OCB (check), temperature (check), observationStartTime, randomize

3.4.9.4 List of CCF Components used

- `ADUConv`
- `DarkFrame` through access of `CAL_rgsGetDarkFrame`

3.4.10 CAL_gainCorrect

Conversion of PHA values to PI (gain correction)

3.4.10.1 Procedure

Function to convert an array of PHA channels into PI channels. This conversion takes into account the momentary principal gain values of the current CCD readout node.

By using the GAIN as stored in the second extension of `ADUConv`, the PHA channels will be scaled to PI channels. The range of the PI channels will be such that they are roughly corresponding to energies in eV.

Therefore the basic formula is

$$PI = PHA/GAIN \ .$$

3.4.10.2 Calling Parameters

3.4.10.2.1 Input

`PHA` (float): PHA value to be converted

`NODE` (array, optional): contains the node-identifier for the array of `PHA` values. If provided it overwrites the value of the state variable ccdNodeId.

`time` event time coordinates in the TIMCOORD time frame (see section 2.2)

`FPA_temp` temperature of the focal plane assembly; this affects the temperature of the readout electronics.

Notes

- Instabilities of the gain due to temperature variations of the electronics are not included at the moment, as the temperature of the FPA is expected to be sufficiently stable.
- For the time being, no temperature dependence will be included, however consistency will be checked between the current temperature and the temperature for which CCF `ADUConv` is valid. If the temperatures are inconsistent, a warning will be issued.

3.4.10.2.2 Output

`PI` (float): gain corrected PI

3.4.10.3 Used CAL State Variables

instrument, ccdChipId, ccdNodeId, OCB (check), temperature (check), observationStartTime

3.4.10.4 List of CCF Components used

- `ADUConv`

3.4.11 CAL_rgsCTIcorrect

CTI correction of energy data (RGS)

3.4.11.1 Procedure

In order to preserve the calibrational accuracy of the CTI correction, it should be performed before event recombination. This is due to the way the current values of the CTI correction were derived from analysis of ground calibration data.

This model of CTI correction assumes pulse-height proportional charge loss as a function of distance from the readout node.

The CTI correction of the parallel shifts is carried out first, followed by the CTI correction of the serial register.

The CTI correction for the parallel transfers has a dependency on the column number (`DETX`), due to lower CTE closer to the sides of the CCD's. The CTI is performed according to

$$PI'_{\text{corr}} = PI \times [1 + y \times YCTI(x)]$$

where x and y are the distances of the location of the event from the respective readout node in CCD pixels, starting from $(x, y) = (1, 1)$.

The CTI correction for the serial register is

$$PI_{\text{corr}} = PI'_{\text{corr}} \times [1 + x \times XCTI]$$

with the same convention as before.

$XCTI$ is stored per readout node in CCF `CTI`, and YCTI is stored per column (`DETX`-coordinate).

No temperature dependence will be included, however consistency will be checked between the current temperature and the temperature for which CCF `CTI` is valid. If the temperatures are inconsistent, a warning will be issued.

Note, that in due time the history of the events (column wise) may have to be included: e.g. to apply the CTI correction only to the first pixel of an multi-pixel event in a certain column and readout.

3.4.11.2 Calling Parameters

3.4.11.2.1 Input

`rawX, rawY` pixel coordinates

`NODEID` (optional array): if present, it has the same length as the `rawX, rawY` pairs, and it overwrites the state variable ccdNodeId

`PI` (float): PI value of pixel

`SHAPE` shape code of event as determined during split event reconstruction; note that the shape codes are relative to the readout node, and thus the same shape code when displayed on the CCD is mirrored from one node to the other.
This is not used for HTR mode.

`SEPARATION` the distance in terms of number of transfers to the event preceding the current one in the same column.
This is not used for HTR mode.

Notes

- Degradation of the CTI with time will most likely require both: updates to this function, and a new issue of CCF `CTI`

3.4.11.2.2 Output

`PHA` (float): computed CTI-corrected PHA

3.4.11.3 Used CAL State Variables

instrument, ccdChipId, ccdNodeId, ccdModeId, OCB (check), temperature (check)

3.4.11.4 List of CCF Components used

- `CTI`

3.4.12 CAL_getRedistribution

return PI response for an input energy

3.4.12.1 Procedure

The redistribution function of the CCD's is parameterized per CCD and per node. The model is described in [1] with an update described in [2]. It combines the X-ray absorption probability in silicon with an empirical parameterization of the generated charge signal, which is then folded by a Gaussian for noise representation.

The probability P for absorption of a photon in silicon at the depth x is given by

$$\frac{\mathrm{d}P}{\mathrm{d}x} = \frac{1}{\tau}\, e^{-x/\tau} \ , \tag{2}$$

with the mean absorption length in silicon τ. For a given photon of energy E (in eV), and absorption at x, the collected charge Q (in eV) is parameterized with the empirical model

$$Q = \begin{cases} 0 & \text{for } Q < T \\ T + (E-T)\left(1 - e^{-(x/b)^{1/3}}\right) & \text{for } Q \geq T \end{cases} \tag{3}$$

with a threshold for charge detection $T = 50\,\text{eV}$, and $b = 20.95\,\text{nm}$ being a parameter that defines the scale of the collected charge.

The charge probability density then is

$$\frac{\mathrm{d}P}{\mathrm{d}Q} = \frac{\mathrm{d}P}{\mathrm{d}x}\frac{\mathrm{d}x}{\mathrm{d}Q} \ . \tag{4}$$

From (3) follows that

$$\mathrm{d}Q = (E-T)\ e^{-(x/b)^{1/3}} \left(\frac{x}{b}\right)^{-2/3} \frac{1}{3b}\ \mathrm{d}x\ , \tag{5}$$

and hence

$$\frac{\mathrm{d}P}{\mathrm{d}Q} = \frac{1}{\tau}\ e^{-x/\tau} \frac{3b}{E-T}\ e^{(x/b)^{1/3}} \left(\frac{x}{b}\right)^{2/3} \tag{6}$$

Reforming (3) into

$$\left(\frac{x}{b}\right)^{1/3} = \ln\frac{E-T}{E-Q}\ , \tag{7}$$

allows to eliminate x from (6), and yields

$$\frac{\mathrm{d}P}{\mathrm{d}Q} = \frac{3b}{\tau}\ \frac{1}{E-Q}\ \ln^2\frac{E-T}{E-Q}\ e^{-\frac{b}{\tau}\ln^3\frac{E-T}{E-Q}}\ , \tag{8}$$

which gives the response probability of an ideal CCD to an incident photon of energy E with the parameters b and T which are specified in the CCF `REDIST`.

To this function a partial event tail is added that has a constant probability density for all charges less than the incident energy, and zero above:

$$\frac{\mathrm{d}I(t,Q,E)}{\mathrm{d}Q} = \begin{cases} \frac{\mathrm{d}P}{\mathrm{d}Q} + t & \text{for } Q < E \\ \frac{\mathrm{d}P}{\mathrm{d}Q} & \text{for } Q > E \end{cases} \tag{9}$$

t is the differential amplitude of the total fraction of partial events P_{part}. It is $t = t(Q) =$ `const` and its value is defined from

$$P_{\text{part}} = \int_0^E t\,\mathrm{d}Q = t\ E\ . \tag{10}$$

$P_{\rm part}$ is parameterized as

$$P_{\rm part} = g \times \left[1 - e^{-c(E-d)}\right] \times \frac{\tau(E_{\rm ref})}{\tau(E)} \tag{11}$$

with the parameters c, d, g and $E_{\rm ref}$, which are specified in the CCF `REDIST`.

Finally $\mathrm{d}I(t,Q,E)/\mathrm{d}Q$ is convolved with a Gaussian to represent the Fano-noise and the amplifier noise. The σ (in eV) of this Gaussian can be written as

$$\sigma = \sqrt{FEw + sa^2w^2}\,, \tag{12}$$

with Fano factor F, mean pair generation energy w (in eV/e$^-$), average event size s (in pixels), and amplifier noise a (in electrons).

It has to be noted that the result of the this function, as far as described so far, is in units of energy. In order to be able to compare with data, which are in units of PI, the output of this function has to be converted to PI, taking the relationship between the definition of PI and energy into account.

Note that the Si escape peak is not included in the current model.

3.4.12.2 Calling Parameters

3.4.12.2.1 Input

`rawX, rawY` pixel coordinates

`ENERGY` energy for which the redistribution is requested

`REJTHRESH` value of the rejection threshold (currently unused)

`FPA_temp` temperature of the focal plane assembly; this affects the temperature of the readout electronics.

3.4.12.2.2 Output

`PI` (array) channels of redistribution function

`INTENS` (array) normalized probability density per PI channel

3.4.12.3 Used CAL State Variables

instrument, ccdChipId, ccdNodeId, ccdModeId, OCB (check), temperature (check)

3.4.12.4 List of CCF Components used

- REDIST

3.4.13 CAL_getCCDQuantumEfficiency

CCD quantum efficiency

3.4.13.1 Procedure

This will provide quantum efficiencies for CCD's. The QE will be parameterized.

This will only be the QE of the CCD, excluding additional effects e.g. such as contaminations.

The values calculated from the Henke tables, are amended over a limited range in energy by the absorption values provided in the CCF EXAFS (section 4.4.12) with a finer energy grid.

3.4.13.2 Calling Parameters

3.4.13.2.1 Input

rawX, rawY pixel coordinates

ENERGY energy for which the QE is requested

SHAPE shape code of event as determined during split event reconstruction

ACCTHRESH acceptance threshold; this is only used as a check for consistency with the version of CCF QuantumEf

3.4.13.2.2 Output

CCD_QE quantum efficiency

3.4.13.3 Used CAL State Variables

instrument, ccdChipId, ccdNodeId, ccdModeId, OCB (check)

3.4.13.4 List of CCF Components used

- QuantumEf

- EXAFS

3.4.14 CAL_getCalSrcRegions

3.4.14.1 Procedure

Returns parameters to be used as exclusion regions of the internal calibration source events.

3.4.14.2 Calling Parameters

3.4.14.2.1 Input

`CCDID` CCD identifier

`NODE` node number

3.4.14.2.2 Output

`XLOW` lower limit of extent of exclusion region in RAWX

`XHIGH` upper limit of extent of exclusion region in RAWX

`YLOW` lower limit of extent of exclusion region in RAWY

`YHIGH` upper limit of extent of exclusion region in RAWY

`ENERGY` energy of emitted line

3.4.14.3 Used CAL State Variables

instrument, OCB

3.4.14.4 List of CCF Components used

- `CalSourceData`

3.4.15 CAL_getCalSourceData

3.4.15.1 Procedure

Note: This function is not yet available.

The intensity distribution will be fitted with TBD functions (splines, etc) and the parameters will be stored in CCF `CalSourceData`. A second extension will hold the energy spectrum for the high energy component (TBC).

The intensities of the calibration sources will be corrected for the half life of the source.

The function returns a energy spectrum in the range from `E_LOW` to `E_HIGH` for a given position on the CCD.

This function is used for the background determination after definition of a order selection cut.

3.4.15.2 Calling Parameters

3.4.15.2.1 Input

`rawX, rawY` pixel coordinates

`E_LOW` lower limit of energy range

`E_HIGH` upper limit of energy range

`NBINS` number of bins

3.4.15.2.2 Output

`ENERGY` (array): the centers of the energy bins

`INTENSITY` (array): the intensities per energy bin

3.4.15.3 Used CAL State Variables

instrument, ccdChipId, ccdNodeId, observationStartTime

3.4.15.4 List of CCF Components used

- `CalSourceData`

3.4.16 CAL_getContaminationData

Provides contamination data per material and absorption efficiency.

3.4.16.1 Procedure

Provides the absorption efficiency of a layer of contaminants. `CAL_getAbsCoeff` is used to obtain the absorption efficiency curves from CCF `AbsCoefs`.

3.4.16.2 Calling Parameters

3.4.16.2.1 Input

`rawX, rawY` pixel coordinates

`ENERGY` energy for which modified QE is requested

`time` event time coordinates in the TIMCOORD time frame (see section 2.2)

3.4.16.2.2 Output

`material` (array) type of atom or compounds

`THICK` (array) thickness of layer

`ABSORB_EFF` total absorption efficiency by all contaminating materials

3.4.16.3 Used CAL State Variables

instrument, ccdChipId, ccdNodeId

3.4.16.4 List of CCF Components used

- `Contam`
- `AbsCoefs` through call to `CAL_getAbsCoeff`

3.4.17 CAL_getEventSize

3.4.17.1 Procedure

Obtains the fractional intensity of a given event size. This is used for monitoring purposes, but also for the determination of the effective area correction when filtering on event size is performed.

3.4.17.2 Calling Parameters

3.4.17.2.1 Input

`SIZE` size of events for which data is requested

`FPA_temp` temperature of the focal plane assembly; this affects the temperature of the readout electronics.

3.4.17.2.2 Output

`INTENS` relative intensity of specified event size

3.4.17.3 Used CAL State Variables

instrument, ccdChipId, ccdNodeId, ccdModeId, OCB (check), temperature (check), observationStartTime

3.4.17.4 List of CCF Components used

- `EventSizeDist`

3.4.18 CAL_rgsGetLAScatterRoughness

Access to the roughness parameter of the large angle scatter.

3.4.18.1 Procedure

Interpolation of the tabulated values as a function of wavelength.

3.4.18.2 Calling Parameters

3.4.18.2.1 Input

`Wavelength` wavelength at which the scatter parameter is sought

3.4.18.2.2 Output

`RMS` interpolated micro-roughness parameter at `Wavelength`

3.4.18.3 Used CAL State Variables

instrument

3.4.18.4 List of CCF Components used

- `LineSpreadFunc`, table `LASCAT`

3.4.19 CAL_rgsGetScatterPars

Access to the RGS scatter parameters.

3.4.19.1 Procedure

Access to parameters stored in the CCF.

3.4.19.2 Calling Parameters

3.4.19.2.1 Input

none

3.4.19.2.2 Output

`A, B` values of the parameters a_i & b_i that are part of Equation (14) in section 3.4.20.

`RMS_SMALL` surface roughness of small angle scatter

`CORLEN_SMALL` correlation length of small angle scatter

`RMS_LARGE` surface roughness of large angle scatter; this is provided for completeness only. With the current knwoledge of the calibration, the values provided by function `rgsGetLAScatterRoughness` (section 3.4.18) should be used.

`CORLEN_LARGE` correlation length of large angle scatter

3.4.19.3 Used CAL State Variables

instrument

3.4.19.4 List of CCF Components used

- `LineSpreadFunc`, table `SCATTER`

3.4.20 CAL_rgsGetScatter

The scattering contribution of the RGS line-spread-function.

3.4.20.1 Procedure

This description is a copy of in [3]. The following definitions are used:

$k = 2\pi/\lambda$, where λ is the wavelength at which the LSF is to be evaluated

σ_s is the r.m.s. surface roughness for two scatter models; s is either *small* or *large* for small angle scatter, and large angle scatter, respectively. This is available from CCF `LineSpreadFunc`. Note that the surface roughness for the large angle scatter is parameterized as a function of wavelength λ and its value is obtained with `CAL_rgsGetLAScatterRoughness` (section 3.4.18)

l_s is the correlation length of the scatter components; s is either *small* or *large* for small angle scatter, and large angle scatter, respectively. This is available from CCF `LineSpreadFunc`.

$\beta_{\rm m}$ is the center angle of the unscattered light distribution

The fraction of scattered light for small/large angle scatter is

$$F_{\rm small,large} = 1 - e^{-[k\sigma_{\rm small,large}(\sin\alpha+\sin\beta_{\rm m})]^2}\ , \tag{13}$$

with a correction factor of the LSF probability density $P(\beta)$ for large angle scattering of

$$R = \frac{a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4}{b_0 + b_1 x + b_2 x^2 + b_3 x^3 + b_4 x^4} \tag{14}$$

where a_i & b_i are stored in CCF `LineSpreadFunc`, and x is defined by

$$x = \frac{1}{k\, l_{\rm large} \sin^2\beta_{\rm m}}\ .$$

The power spectral densities are defined with respect to p, with

$$p = k(\cos\beta_{\rm m} - \cos\beta) \ ,$$

and finally the density functions for small and large angle scatter are

$$W_{\rm small}(p) = \frac{1}{\sqrt{\pi}}\, l_{\rm small}\, \sigma^2_{\rm small}\, e^{-l^2_{\rm small}\, p^2} \tag{15}$$

$$W_{\rm large}(p) = \frac{1}{\pi}\, l_{\rm large}\, \sigma^2_{\rm large}\, \frac{1}{1 + l^2_{\rm large}\, p^2} \ . \tag{16}$$

The intensity of the LSF for one bin in β that has a range from $\beta - \Delta$ to $\beta + \Delta$ is then given by

$$LSF(\beta) = F_{\rm large}\, \frac{R \cdot \sin\beta_{\rm m}\, k^3}{(k\sigma_{\rm large}(\sin\alpha + \sin\beta_{\rm m}))^2} \int_{\beta-\Delta}^{\beta+\Delta} \frac{(\sin\alpha + \sin\beta)^4}{(\sin\alpha + \sin\beta_{\rm m})^2}\, W_{\rm large}\ d\beta + \tag{17}$$

$$(1 - F_{\rm large})F_{\rm small}\, \frac{\sin\beta_{\rm m}\, k^3}{(k\sigma_{\rm small}(\sin\alpha + \sin\beta_{\rm m}))^2} \int_{\beta-\Delta}^{\beta+\Delta} \frac{(\sin\alpha + \sin\beta)^4}{(\sin\alpha + \sin\beta_{\rm m})^2}\, W_{\rm small}\ d\beta + \tag{18}$$

$$(1 - F_{\rm large})(1 - F_{\rm small}) \int_{\beta-\Delta}^{\beta+\Delta} \delta(\beta - \beta_{\rm m})\ d\beta \ , \tag{19}$$

where the total fraction of non-scattered light is given by $(1 - F_{\rm large})(1 - F_{\rm small})$. The function that is implemented in the CAL does not include the last term (the δ-function), as is only the unscattered light and its contribution is taken care of by the calling task (`rgsrmfgen`).

In order to obtain the LSF, this needs to be convolved with the PSF of the mirror, which acts as a perturbation of α. This is not performed by this function.

3.4.20.2 Calling Parameters

3.4.20.2.1 Input

`ENERGY` energy at which the scatter has to be evaluated

`ORDER` order of diffraction

`ALPHA` angle of incidence on the central grating α

`NHWHM` width over which the scatter distribution is to be evaluated in units of FWHM

`BETA_WIDTH` width of channels for which the distribution shall be calculated

3.4.20.2.2 Output

`SCATTER` (array) the scatter probability distribution
the distribution will be evaluated such that `BETA_M` (which is derived from (`ENERGY`, `ORDER`, `ALPHA`) is at the center of the central bin.

`CENTER_BIN` the bin number of the central bin

3.4.20.3 Used CAL State Variables

instrument

3.4.20.4 List of CCF Components used

- `LineSpreadFunc`

3.4.21 CAL_rgsGetLAScatter

The large angle scatter distribution of the RGS line-spread-function.

3.4.21.1 Procedure

The procedure is identical as outlined in section 3.4.20, but only the large angle scatter distribution is returned.

Only Equations (13) (14) (16) (17) are considered.

3.4.21.2 Calling Parameters

Identical to those specified in section 3.4.20.

3.4.21.3 Used CAL State Variables

instrument

3.4.21.4 List of CCF Components used

- `LineSpreadFunc`

3.4.22 CAL_rgsGetSAScatter

The small angle scatter distribution of the RGS line-spread-function.

3.4.22.1 Procedure

The procedure is identical as outlined in section 3.4.20, but only the large small scatter distribution is returned.

Only Equations (13) (15) (18) are considered.

3.4.22.2 Calling Parameters

Identical to those specified in section 3.4.20.

3.4.22.3 Used CAL State Variables

instrument

3.4.22.4 List of CCF Components used

- `LineSpreadFunc`

3.4.23 CAL_rgsCrossPSF

the PSF in cross-dispersion direction

3.4.23.1 Procedure

The shape of the dispersed image in the cross-dispersion direction is parameterized as a function of dispersion angle β. The parameterization is a superposition of Gaussian and a Lorentzians.

This parameterization is independent of the source position in the FOV along the dispersion direction. In the cross-dispersion direction, the location of the dispersed image is shifted according to

$$\Delta XDISP = L \times \tan\theta \ ,$$

where θ is the component of the off-axis angle in the cross-dispersion direction, and L is the distance between RGA and RFC.

The parameterizations are evaluated at each bin in β by integration over the entire width of the bin.

The functions have the following forms. The Gaussian function is parameterized by

$$G(XDSP) = a(\beta)\ e^{-\frac{1}{2}\left(\frac{XDSP-\mu(\beta)}{\sigma(\beta)}\right)^2} \ ,$$

and the Lorentzian is parameterized by

$$L(XDSP) = \frac{1}{\pi}\frac{b(\beta)\ w(\beta)/2}{(XDSP-\nu(\beta))^2+\frac{w^2(\beta)}{4}} \ .$$

3.4.23.2 Calling Parameters

3.4.23.2.1 Input

`BETA` dispersion angle at center of bin

`XOFFAX` cross-dispersion off-axis angle of celestial target

`XDISP` (array) cross-dispersion angular direction on the RFC dispersed image at which the density shall be calculated

`XDSP_WID` The width of the bin centered at `XDSP`. This will be used as boundaries to calculate the integral of the probability density.

3.4.23.2.2 Output

`PROB` probability of signal

3.4.23.3 Used CAL State Variables

instrument

3.4.23.4 List of CCF Components used

- `CrossPSF`

3.4.24 CAL_getRFCdefocus

Distance between a CCD pixel on the RFC and the Rowland-circle [4].

3.4.24.1 Procedure

For this calculation the grating assembly is approximated by an imaginary grating placed at the center of the grating assembly at point $\vec{G}$.

The angle of incidence on the facets of the gratings

$$\gamma_0 = \alpha_0 + \delta \ ,$$

where δ is the blaze angle, which is available from CCF `MiscData`, item `BLAZE_ANGLE`, and α_0 is the angle of incidence measured from the grating surface for an on axis source. α_0 is available from CCF `MiscData`, item `INCIDENCE_ANGLE`, or from CCF `Lincoord`, item `G_ALPHA`.

Since the RGA may be rotated around $\mathrm{R_y}$, the actual angle of incidence for an on-axis source is

$$\alpha'_0 = \alpha_0 + G_{\mathrm{Ry}} \ .$$

G_{Ry} is stored in CCF `Lincoord`, item `G_RY`.

With an off-axis source at given angular pair (θ, ϕ) in the `TELCOORD` reference frame, the component along the RGS dispersion direction is

$$\phi' = -\theta \cos\phi \ ,$$

and the angle of incidence on the gratings is given by

$$\alpha = \alpha'_0 + \frac{F}{2R\cos\gamma_0}\,\phi' \ , \qquad (20)$$

and The distance between $\vec{G}$ and a virtual focus of the telescope L is

$$L = 2R\,\cos\left(\gamma_0 + \frac{F}{2R\cos\gamma_0}\,\phi'\right) \ . \qquad (21)$$

The hardware related dispersion angle β is calculated from the ray from point $\vec{G}$ to the detector coordinates of the event $\vec{P}(DETX, DETY)$, as

$$\beta = \tan^{-1}\left[\frac{-\left(\vec{P}-\vec{G}\right)_{\mathrm{z}}}{\left(\vec{P}-\vec{G}\right)_{\mathrm{x}}}\right] - \alpha'_0 \qquad (22)$$

with the radius of the Rowland circle R available from CCF `MiscData` item `ROWLAND`.

And finally the defocus value Δx is defined as X-component of the distance between $\vec{P}$ and the intersection of the ray $\bar{GP}$ with the Rowland circle, and is calculated by

$$\begin{aligned}\Delta x = \quad & \left(\vec{P} - \vec{G}\right)_x - 2R\cos(\alpha_0' + \beta - \gamma_0)\,\cos(\alpha_0' + \beta) \\ & - [G_x - (F' - L)] \times \tfrac{\sin\alpha}{\sin\beta} + \Delta x_{\rm offs}\end{aligned} \tag{23}$$

The first component is the distance between detector bin and mid-point of the grating assembly along the `TELCOORD` X-axis; the second item is the distance between point G and the intersection of the dispersed ray with the Rowland circle; the third component is a correction term for a shift of the Rowland circle from it nominal geometry due to a shift of point G during the integrations. F is the focal length of the telescope, which is available from CCF `MiscData`, item `FOCAL_LENGTH`, and

$$F' = F\cos\phi' \tag{24}$$

is the X-component of the shift of the focus due to off-axis sources.

$\Delta x_{\rm offs}$ is a linear offset that is taken from the CCF `LINESPREADFUNC`, table `FIGURE`. This is a convenient quantity to vary the defocus, without a distortion of the wavelength scale.

3.4.24.2 Calling Parameters

3.4.24.2.1 Input

`DETX, DETY` pixel coordinates

`THETA, PHI` off-axis angles of source position in TELCOORD reference frame

3.4.24.2.2 Output

`DEFOCUS` X-component of the distance between detection bin and the Rowland circle.

3.4.24.3 Used CAL State Variables

instrument

3.4.24.4 List of CCF Components used

- `LinCoord`, `MiscData`

3.4.25 CAL_getLSFdefocusDist

A distribution of the broadening of the LSF due to defocusing of the RFC (as described by `CAL_getRFCdefocus`, section 3.4.24).

3.4.25.1 Procedure

These formulae are derived from the description in [4]. The distribution is computed according to the inner (R_1) and outer (R_2) radii of the XRT assemblies, so that the contribution appears like the projected telescope aperture for finite defocus distance Δx (which is provided by `CAL_getRFCdefocus`).

$$
\begin{aligned}
D(\Delta\beta) = \frac{2FL}{\pi R_2 \left[1 - \left(\frac{R_1}{R_2}\right)^2\right] \Delta x} \quad \times & \\
\left[\sqrt{1 - \left(\frac{FL\Delta\beta}{R_2\Delta x}\right)^2} \quad \times \quad \Theta\left(\left|\frac{R_2\Delta x}{FL}\right| - |\Delta\beta|\right) - \right. & \\
\left. \frac{R_1}{R_2}\sqrt{1 - \left(\frac{FL\Delta\beta}{R_1\Delta x}\right)^2} \quad \times \quad \Theta\left(\left|\frac{R_1\Delta x}{FL}\right| - |\Delta\beta|\right)\right] , &
\end{aligned}
\tag{25}
$$

with $F = F'$, which is the result from Eq. (24), and L as in Eq. (21), $\Theta(x)$ is the Heaviside-step function with its derivative $\delta(x)$, and $\Delta\beta_i = \beta_i - \beta_\mathrm{m}$. Eq. (25) needs to be integrated for each bin of $\Delta\beta_i$, and thus the probability for bin i is

$$
\begin{aligned}
P_i \quad = \quad & Q(R_1) \times \Theta\left(\left|\frac{R_2\Delta x_i}{FL}\right| - |\Delta\beta_i|\right)\Bigg|_{\Delta\beta_i}^{\Delta\beta_{i+1}} - \int_{\Delta\beta_i}^{\Delta\beta_{i+1}} Q(R_1)\,\delta\left(\left|\frac{R_2\Delta x_i}{FL}\right| - |\Delta\beta_i|\right) \mathrm{d}\Delta\beta_i \\
- \quad & G(R_2) \times \Theta\left(\left|\frac{R_1\Delta x_i}{FL}\right| - |\Delta\beta_i|\right)\Bigg|_{\Delta\beta_i}^{\Delta\beta_{i+1}} + \int_{\Delta\beta_i}^{\Delta\beta_{i+1}} Q(R_2)\,\delta\left(\left|\frac{R_1\Delta x_i}{FL}\right| - |\Delta\beta_i|\right) \mathrm{d}\Delta\beta_i
\end{aligned}
\tag{26}
$$

with

$$
Q(R) = \frac{-1}{\pi\Delta x(R_1^2 - R_2^2)}\left[\Delta\beta RFL\sqrt{\frac{(R\Delta x)^2 - (FL\Delta\beta)^2}{(R\Delta x)^2}} + (R\Delta x)^2 \arcsin\left(\frac{FL\Delta\beta}{R\Delta x}\right)\right] \tag{27}
$$

and $\Delta x_i = \frac{1}{2}\left[\Delta x(\Delta\beta_1) + \Delta x(\Delta\beta_2)\right]$. P_i is filled such that the first moment of P_i is 0.

3.4.25.2 Calling Parameters

3.4.25.2.1 Input

`ENERGY` energy at which the defocus distribution has to be evaluated

`ORDER` order of diffraction

`ALPHA` angle of incidence on the central grating α

`NFWHM` width over which the scatter distribution is to be evaluated in units of FWHM (FWHM is numerically evaluated by the CAL)

`BETA_WIDTH` width of channels for which the distribution shall be calculated

`THETA, PHI` off-axis angles of source position in TELCOORD reference frame

3.4.25.2.2 Output

`LSFDEFOCUS` (array) the probability distribution of the defocussed LSF is be evaluated such that `BETA_M` (which is derived from (`ENERGY`, `ORDER`, `ALPHA`) is at the center of the central bin.

`CENTER_BIN` the bin number of the central bin

3.4.25.3 Used CAL State Variables

instrument

3.4.25.4 List of CCF Components used

- `LinCoord`, `MiscData`

3.4.26 broadeningDistribution

The purpose of this distribution is to broaden the LSF to account for the finite width of the (incident photon) energy bins. It is used by `rgsrmfgen`.

3.4.26.1 Procedure

A probability is calculated for a given wavelength, dispersion angle and order, according to

$$p = \Delta\beta \times \frac{d}{m\lambda} \times \sin\beta_{\mathrm{m}} \ ,$$

where $\Delta\beta$ is the width of the spectral channels in dispersion angles, d is the groove spacing of the gratings, m is the spectral order, λ is the wavelength, and β_{m} is the dispersion angle for unscattered light.

The function returns a flat distribution of $1/p$ channels, with constant probability per channel of p.

3.4.26.2 Calling Parameters

3.4.26.2.1 Input

`BETA_WIDTH` width of channels for which the distribution shall be calculated

`LAMBDA` wavelength in Å

`BETA_M` is the dispersion angle corresponding to `LAMBDA` for unscattered light

`ORDER` spectral order

3.4.26.2.2 Output

`BroadeningDist` Broadening distribution to be used in construction of LSF through convolution of various components.

3.4.26.3 Used CAL State Variables

instrument

3.4.26.4 List of CCF Components used

- `MiscData`

3.4.27 CAL_getRGAQuantumEfficiency

RGA grating reflection efficiency

3.4.27.1 Procedure

This will provide reflection efficiencies for RGA grating plates. This is stored in form of a lookup table and a second table with an empirical correction.

3.4.27.2 Calling Parameters

3.4.27.2.1 Input

`ENERGY` incident X-ray energy

`ALPHA` angle of incidence α

`ORDER` reflection order

3.4.27.2.2 Output

`RGA_QE` reflection efficiency

3.4.27.3 Used CAL State Variables

None.

3.4.27.4 List of CCF Components used

- `QuantumEf`

3.4.28 CAL_getRGAVign

Provides the transmission due to self vignetting of the RGA as a function of dispersion angle β. This is used for an analytical evaluation of the effective area. Due to the close stacking of the grating plates inside the RGA, light at large dispersion angles cannot leave the RGA assembly, and therefore, effectively the active surface of the grating plates is vignetted by the back of the neighboring plate.

3.4.28.1 Procedure

This vignetting of the RGA is effective for $\beta > \beta_{\mathrm{vig}}$. The transmission of the RGA has the following parameterization:

$$F(\beta) = \begin{cases} 1 & , \text{ for } \beta < \beta_{\mathrm{vig}} \\ \frac{\beta_{\mathrm{vig}}}{\tan\beta} & , \text{ for } \beta > \beta_{\mathrm{vig}} \end{cases}$$

Emiprical Corrections of these values are provided in the table `RGA_SELFVIGNCORR` in CCF `QuantumEf` (section 4.4.17). After interpolation of this table, the analytical function be multiplied my these correction factors.

3.4.28.2 Calling Parameters

3.4.28.2.1 Input

`BETA` dispersion angle

3.4.28.2.2 Output

`VIGNET` transmission of the RGA

3.4.28.3 Used CAL State Variables

instrument

3.4.28.4 List of CCF Components used

- `QuantumEf`

3.4.29 CAL_getRGAIntercept

Provides the fraction of light that the RGA intercepts of the total light that exits the telescope. This is the fraction that is collected by the surface of the gratings, which is used for the dispersion.

Note that the transmission to EPIC is **not** the complement of this function, as in both cases also the passive volume of the RGA support structure has to be taken into account. The fraction of light transmitted to EPIC is available in `CAL_getRGAObscure`.

3.4.29.1 Procedure

The fraction of intercepted light equals η for on-axis source positions and is tabulated in `QuantumEf` (see section 4.4.17). The function interpolates between the energy points tabulated.

For off-axis sources this parameterization is scaled with the angle of incidence for on-axis sources α_0 according to

$$\eta' = \eta \frac{\alpha}{\alpha_0} \ .$$

η is dependent on energy due to the fact that the effective area of the mirror shells is a function of energy (smaller shells have higher effective area at higher energies).

α is calculated from the dispersion component of the off-axis position of the source in the following way: using the TELCOORD coordinate system, the position of the source in the field-of-view is given by angles ϕ & θ. It follows then

$$\alpha = \alpha_0 - \frac{F}{L}\,\theta\;\cos\phi\;, \tag{28}$$

where F is the focal length of the respective mirror module, and L is the distance between RGS focal camera (RFC) and Reflection Grating Array (RGA).

3.4.29.2 Calling Parameters

3.4.29.2.1 Input

`ALPHA` angle of incidence on the gratings

`ENERGY` energy of incident photons

3.4.29.2.2 Output

`ETA1` fraction of intercepted light as defined above as η'

3.4.29.3 Used CAL State Variables

instrument

3.4.29.4 List of CCF Components used

- `QuantumEf`
- `LinCoord`

3.4.30 CAL_getRGAObscure

Provides the fraction of light that is blocked by the RGA from the beam to the EPIC focus. This is different than $1 -$ `CAL_getRGAIntercept`, as the RGA support structure has to be considered too.

3.4.30.1 Procedure

The functional form is identical to that in `CAL_getRGAIntercept` (section 3.4.29), except that the tabulation of κ is used for the efficiency. κ is scaled for off-axis sources identically to function `CAL_getRGAIntercept`, by

$$\kappa' = \kappa \frac{\alpha}{\alpha_0} .$$

κ is tabulated in CCF `QuantumEf` for on-axis sources, and is interpolated for different off-axis values by this equation. The remainder is identical to function `CAL_getRGAIntercept`.

3.4.30.2 Calling Parameters

3.4.30.2.1 Input

`ALPHA` angle of incidence on the gratings

`ENERGY` energy of incident photons

3.4.30.2.2 Output

`KAPPA1` fraction of intercepted light as defined above as κ'

3.4.30.3 Used CAL State Variables

instrument

3.4.30.4 List of CCF Components used

- `QuantumEf`
- `LinCoord`

3.4.31 CAL_getRGAFigure

The figure error from the RGA grating mis-alignment distribution.

3.4.31.1 Procedure

The grating mis-alignment is parameterized with a set of Gaussian functions, whose parameters are stored in the CCF. The corresponding probability density is according to

$$p(\beta) = \sum_{i=1}^{N} \left[a_i \; e^{-\frac{1}{2}\left(\frac{\beta - f(1+C)\mu_i}{f(1+C)\sigma_i}\right)^2} \right]$$

The parameters a_i, μ_i, σ_i are stored in the CCF. N is the number of parameters available in the CCF. C is the chromatic magnification that scales the width of the grating mis-alignment distribution as a function of angle of incidence and diffraction angle. It is defined as

$$C = \frac{\sin\alpha}{\sin\beta_{\mathrm{m}}}$$

and is an input argument.

Additionally to the chromatic magnification, the alignment distribution is homogeniously scaled by a factor f, which is stored in the CCF.

The average center of all Gaussians is calculated by

$$a'_i = a_i \; \sigma_i \; \sqrt{2\pi}$$

$$\mu_0 = \frac{\sum a'_i \cdot \mu_i}{\sum a'_i} \; .$$

The function returns the probability distribution $P(j)$ which calculated from $p(\beta)$ by integration per channel j

$$P(j) = \sum_{i=1}^{N} a'_i \; \int_{\beta_j}^{\beta_{j+1}} e^{-\frac{1}{2}\left(\frac{\beta - [(1+fC)(\mu_i - \mu_0) + \mu_0]}{(1+fC)\sigma_i}\right)^2} \, \mathrm{d}\beta \; .$$

The distribution will be calculated such that the peak is at the center of the bins.

3.4.31.2 Calling Parameters

3.4.31.2.1 Input

`NSIGMA` width over which the distribution is to be evaluated in units of SIGMA of the sum of the Gaussians

`BETA_WIDTH` width of bins

`CHROMAG` chromatic magnification factor C

3.4.31.2.2 Output

`P` Probablity distribution $P(j)$ as described above which is centered around the average of the Gaussians.

3.4.31.3 Used CAL State Variables

instrument

3.4.31.4 List of CCF Components used

- `LineSpreadFunc`, table `FIGURE`

3.4.32 CAL_getGratBows

The contribution of the grating non-flatness bows to the figure error of the RGA distribution.

3.4.32.1 Procedure

The grating non-flatness (bow) orthoginal to the dispersion plane is parameterized with a set of Gaussian functions, whose parameters are stored in the CCF. The corresponding probability density is calculated in an identical way as specified for function `CAL_getRGAFigure` (section 3.4.31).

3.4.32.2 Calling Parameters

3.4.32.2.1 Input

`NSIGMA` width over which the distribution is to be evaluated in units of SIGMA of the sum of the Gaussians

`BETA_WIDTH` width of bins

`CHROMAG` chromatic magnification factor C

3.4.32.2.2 Output

`P` Probablity distribution $P(j)$ as described above which is centered around the average of the Gaussians.

3.4.32.3 Used CAL State Variables

instrument

3.4.32.4 List of CCF Components used

- `LineSpreadFunc`, table `BOWS`

3.4.33 CAL_rgsgetXRTFigure

The contribution of the figure properties of the telescope to the LSF of RGS.

3.4.33.1 Procedure

This provides the shape of the XRT along the dispersion direction (integrated in cross-dispersion).

This function can have two modes of operations:

1. Numerical integration of the parameterized PSF (see HIGH accuracy level described in section 3.2.1)
   This parameterization is numerically integrated along the cross-dispersion direction.
2. Dedicated parameterization
   A dedicated set of parameters is provided that describes the PSF when integrated on the cross-dispersion direction. Gaussian and Lorentzian functions and a linear combination are allowed. The Gaussian parameterization has the form

$$G_i(\beta) = a_i \; e^{-\frac{1}{2}\left(\frac{\mu_i+\beta-[C(\mu_i-\mu_0)]}{C\sigma_i}\right)^2} ,$$

   with the parameters a_i, μ_i, and σ_i provided by the CCF per Gaussian. Lorentzian parameterizations have the form

$$L_i(\beta) = \frac{1}{\pi}\frac{b_i \; C \; w_i/2}{(\beta - [\nu_i + C(\nu_i - \mu_0)])^2 + \frac{C^2 w_i^2}{4}} ,$$

   with the parameters b_i, nu_i, and w_i provided by the CCF per Lorentzian.

   The functions are selected by a flag in the CCF. The parameters of the centers (μ and ν) are given with respect to a common center to allow for an assymetric parameterizations. The common center μ_0 is calculated by

$$a_i' = a_i \; \sigma_i \; \sqrt{2\pi}$$

$$\mu_{\rm G} = \frac{\sum a_i' \cdot \mu_i}{\sum a_i'}$$

$$\mu_\mathrm{L} = \frac{\sum b_i \cdot \nu_i}{\sum b_i}$$

$$\mu_0 = \frac{\mu_\mathrm{G} \sum a'_i + \mu_\mathrm{L} \sum b_i}{\sum a'_i + \sum b_i} \ .$$

The value of the input argument `CENTER_CHAN` is added to μ_0, in order to provide the PSF correctly centered.

The width of the PSF of the telescope scales as a function of angle of incidence on the gratings and diffration angle of the RGA by the chromatic magnification function C. The chromatic magnification which is an imaging effect by the gratings that scales the width of the PSF in the dispersion direction as a function of angle of incidence and diffraction angle. It is defined as

$$C = \frac{\sin \alpha}{\sin \beta_\mathrm{m}}$$

and is an input argument.

The function returns the probability distribution $P(j)$ which calculated from $p(\beta)$ by integration per channel j

$$P(j) = \sum_{i=1}^{N_\mathrm{Gauss}} \left(a_i \ \sigma_i \ \sqrt{2\pi}\right) \int_{\beta_j}^{\beta_{j+1}} G_i(\beta) \ \mathrm{d}\beta + \sum_{i=1}^{N_\mathrm{Lorentz}} b_i \int_{\beta_j}^{\beta_{j+1}} L_i(\beta) \ \mathrm{d}\beta \ .$$

Note: The CAL inverts the parameterization that is in `XRT_XPSF` along the dispersion axis, because the RGA acts as a mirorr in this respect.

3.4.33.2 Calling Parameters

3.4.33.2.1 Input

`NHWHM` width over which the XRT PSF is to be evaluated in units of FWHM of the sum of the distributions

`BETA_WIDTH` width of channels for which the distribution shall be calculated

`OFFSET` offset of the center of the distribution as a fraction of `BETA_WIDTH`

`CHROMAG` chromatic magnification factor C

`TYPE` type of function; either numerical or parameterized

3.4.33.2.2 Output

`P` Probablity distribution $P(j)$ as described above which is centered around `CENTER_CHAN`.

3.4.33.3 Used CAL State Variables

instrument

3.4.33.4 List of CCF Components used

- XRT_XPSF

3.4.34 CAL_rgsgetLSF

3.4.34.1 Procedure

Calculates the LSF of RGS for one energy of incidence. This is a convolution of

$$\texttt{CAL\_rgsGetScatter} \otimes \texttt{CAL\_getRGAVign} \otimes \texttt{CAL\_getRGAFigure} \otimes \texttt{CAL\_getPSFmap}$$

3.4.34.2 Calling Parameters

3.4.34.2.1 Input

`ENERGY` energy at which the LSF has to be evaluated

`ORDER` order of diffraction

`ALPHA` angle of incidence on the central grating α

`BETA` (array) dispersion angle bins (float) at which the LSF has to be evaluated

`BETA_WID` (array) width of dispersion angle bins at which the LSF has to be evaluated

3.4.34.2.2 Output

`LSF` (array) LSF as a function of dispersion angle (in units of mm^2

3.4.34.3 Used CAL State Variables

instrument

3.4.34.4 List of CCF Components used

- `LineSpreadFunc`
- `QuantumEf`

3.4.35 CAL_getRGSEffAreaCorr

3.4.35.1 Procedure

This is a tabulated correction of the effective area, which has to be applied after all other efficiency related correction factors. This is obtained by interpolation of the table `RGA_EFFACORR`.

3.4.35.2 Calling Parameters

3.4.35.2.1 Input

`ENERGY` energy at which a correction of the effective area needs to be obtained.

3.4.35.2.2 Output

`FACTOR` value of the correction.

3.4.35.3 Used CAL State Variables

instrument

3.4.35.4 List of CCF Components used

- `QuantumEf`

3.4.36 CAL_getRGSEffectiveArea

Total effective area curve of RGS.

3.4.36.1 Procedure

This is the product of the following functions:

$$
\begin{aligned}
A_{\mathrm{eff}} = \; & \texttt{CAL\_getEffectiveArea} \\
& \times \texttt{CAL\_getRGAIntercept} \\
& \times \texttt{CAL\_getRGAQuantumEfficiency} \\
& \times \texttt{CAL\_getRGAVign} \\
& \times \texttt{CAL\_rgsCrossPSF} \times f_{\mathrm{xdspsel}} \\
& \times \texttt{CAL\_rgsgetLSF} \times f_{\mathrm{ordersel}} \\
& \times \texttt{CAL\_getRGSEffAreaCorr}
\end{aligned}
$$

3.4.36.2 Calling Parameters

3.4.36.2.1 Input

ENERGY energy at which the effective area will be calculated.

theta off-axis angle position in the FOCCOORD reference system at which the effective area curve is requested.

phi azimuth angle position in the FOCCOORD reference system at which the effective area curve is requested.

`ORDER` reflection order

`BETA` (array) dispersion angle bins (float) at which the LSF has to be evaluated

`F_ORDERSEL` fraction of range of the CCD redistribution that has to be considered for the order selection; this is numerical input field that describes the selection efficiency that was use to generate the order selection region.

`F_XDSPSEL` fraction of range of the cross-dispersion LSF that has to be considered for the order selection; this is numerical input field that describes the selection efficiency that was use to generate the image selection region.

`REGION_ID` the ID of selection regions (order selection & cross-dispersion selection; TBC)

3.4.36.2.2 Output

`A_EFF` effective area for given energy and order

3.4.36.3 Used CAL State Variables

instrument, observationStartTime

3.4.36.4 List of CCF Components used

- QUANTUMEF
- XAREAEF

3.4.37 CAL_rgsGetEvThresh

3.4.37.1 Procedure

Provides a list of thresholds per CCD.

For GRADE-one events (single pixel events) REJTHRESH is applied such that any pixel with PI below this threshold is strictly ignored by **rgsevents**, and will not be further processed. ACCTHRESH is used such that any event with no pixels at or above this threshold will be marked with the BELOW_ACCEPTANCE rejection flag.

These thresholds REJTHRESH & ACCTHRESH are not directly applicable to pixels with grade larger than one. In order to apply them in such cases a set of three empirically determined charge-splitting ratios is included in QSPLITR2/3/4, one for each of the grades between two and four. The charge-splitting ratio is the average proportion of the total energy belonging to the pixel with the largest individual energy in a reconstructed event. This ratio is used to adjust the threshold to account for the extra energy in pixels from on-board reconstruction.

For GRADE-one events (single pixel events), QSPLITR1 = 1. For GRADE > 1 events, the thresholds are scaled according to

$$\mathtt{REJ\_THRESH}(\mathtt{GRADE},\mathtt{NODE}) = \mathtt{REJTHRESH}/\mathtt{QSPLITR}(\mathtt{GRADE}-1)\ ,$$

and

$$\mathtt{ACC\_THRESH}(\mathtt{GRADE},\mathtt{NODE}) = \mathtt{ACCTHRESH}/\mathtt{QSPLITR}(\mathtt{GRADE}-1)\ .$$

GRADE ranges from 1–4, and QSPLITR2/3/4 ≤ 1.

3.4.37.2 Calling Parameters

3.4.37.2.1 Input

None.

3.4.37.2.2 Output

REJ_THRESH (array) of the form (GRADE, NODE) (FORTRAN syntax).

ACC_THRESH (array) of the form (GRADE, NODE) (FORTRAN syntax).

3.4.37.3 Used CAL State Variables

instrument, ccdChipId, ccdNodeId, ccdModeId (check), OCB (check), temperature (check)

3.4.37.4 List of CCF Components used

- ADUConv

3.4.38 CAL_getHKwindows

Provides HK parameter windows for definition of GTI's.

3.4.38.1 Procedure

The parameter names that stored in CCF HKParmInt, are returned together with their upper and lower validity limits as arrays.

3.4.38.2 Calling Parameters

3.4.38.2.1 Input

time event time coordinates in the TIMCOORD time frame (see section 2.2)

3.4.38.2.2 Output

PARAMETER (array) name of parameter to be checked

SELECT (array) selectlib expression that describes the valid range

3.4.38.3 Used CAL State Variables

instrument, ccdChipId

3.4.38.4 List of CCF Components used

- HKParmInt

3.4.39 CAL_getAUXwindows

Provides auxiliary science parameter windows for definition of GTI's.

3.4.39.1 Procedure

The parameter names that stored in CCF HKParmInt, are returned together with their upper and lower validity limits as arrays.

3.4.39.2 Calling Parameters

3.4.39.2.1 Input

time event time coordinates in the TIMCOORD time frame (see section 2.2)

3.4.39.2.2 Output

PARAMETER (array) name of parameter to be checked

SELECT (array) selectlib expression that describes the valid range

3.4.39.3 Used CAL State Variables

instrument, ccdChipId

3.4.39.4 List of CCF Components used

- HKParmInt

3.5 OM Related CAL Routines

3.5.1 CAL_omGetDegradationCoeffs

CAL_omGetDegradationCoeffs - returns the coefficients to compute the correction factor for time dependent sensitivity degradation.

3.5.1.1 Procedure

The OM CCF component OM_PHOTONAT is read and the coefficients for input filter are extracted from its DEGRADATION fits extension.

3.5.1.2 Calling Parameters

3.5.1.2.1 Input

3.5.1.2.2 Output

`degradationCoeffs` polynomial coeffcients to compute correction factor

3.5.1.3 Used CAL State Variables

FilterId

3.5.1.4 List of CCF Components used

PHOTTONAT

3.5.2 CAL_omGetFluxConvFactors

CAL_omGetFluxConvFactors - returns the conversion factors to compute the flux in a given filter from the corresponding count rate.

3.5.2.1 Procedure

The OM CCF component OM_COLORTRANS is read and an the conversion factors are extracted from the fits keywords. There are conversion factors based in standard white dwarves and also for AB system.

3.5.2.2 Calling Parameters

3.5.2.2.1 Input

3.5.2.2.2 Output

`OmFluxConvF` count rate to flux conversion factors

3.5.2.3 Used CAL State Variables

none

3.5.2.4 List of CCF Components used

COLORTRANS

3.5.3 CAL_omGetGrismFlux

CAL_omGetGrismFlux - returns the flux calibration for an OM grism.

3.5.3.1 Procedure

The OM CCF component OM_GRISMCAL is read and an array of wavelengths, fluxes and flux errors is returned.

3.5.3.2 Calling Parameters

3.5.3.2.1 Input

3.5.3.2.2 Output

`lambda` real array of wavelengths

`isf` real array containing the Inverse Sensitivity Function

`isfError` real array containing the error on the ISF

3.5.3.3 Used CAL State Variables

FilterId

3.5.3.4 List of CCF Components used

GRISMCAL

3.5.4 CAL_omGetGrismWavelength

CAL_omGetGrismWavelength - returns the wavelength calibration for an OM grism.

3.5.4.1 Procedure

The OM CCF component OM_GRISMCAL is read and an array of wavelengths is returned.

3.5.4.2 Calling Parameters

3.5.4.2.1 Input

3.5.4.2.2 Output

`lambda` real array of wavelengths

3.5.4.3 Used CAL State Variables

FilterId

3.5.4.4 List of CCF Components used

GRISMCAL

3.5.5 CAL_omDistortion

CAL_omDistortion - returns the offset from the linear grid position at a specified detector location

3.5.5.1 Procedure

The routine returns the offset (deltaX, deltaY) from the linear grid position for a specified detector location. The linear grid position (linX, linY) is defined by the angular distance to the OM boresight (i.e. pixel 1024.5,1024.5 in the V-filter), divided by the platescale (section 3.5.8). The input may consist of a list of positions. The offset can be calculated in two different ways. Either the offset is obtained by interpolation of the coarsely mapped distortion or by direct evaluation of an up to 7th order polynomial. The way of calculation is selected by setting the input parameter calc_mode.

- If calc_mode is set to 0, then the distortion vector is calculated by interpolation of the distortion map.
- If calc_mode is set to 1, then the distortion vector is calculated by evaluation of the polynomial.

calc_mode is set to 0:
the distortion (deltaX, deltaY) at the detector location (rawX, rawY) is derived from the distortion map stored in the extension FILTER-FilterId (e.g. FILTER-U, FILTER-V, FILTER-GRISM1 etc.).
As the distortion map is only coarsly sampled an interpolation is required to derive the distortion at the requested position (rawX, rawY).

- select correct extension using the FilterId
- extract the n closest records around the specified detector position (rawX, rawY) e.g. (rawX$_1$,rawY$_1$,RAWX_OFF$_1$,RAWY_OFF$_1$),
  (rawX$_2$,rawY$_2$,RAWX_OFF$_2$,RAWY_OFF$_2$), ...,
  (rawX$_n$,rawY$_n$,RAWX_OFF$_n$,RAWY_OFF$_n$).
  with

  $$(rawX_1 - rawX)^2 + (rawY_1 - rawY)^2 \leq ...$$
  $$... \leq (rawX_n - rawX)^2 + (rawY_n - rawY)^2 \leq \text{any other entry}$$

- interpolate the extracted distortion points (RAWX_OFF$_i$,RAWY_OFF$_i$) and calculate the distortion (deltaX,deltaY) at the requested postion (rawX, rawY)

calc_mode is set to 1:
the distortion (deltaX,deltaY) at the location (rawX, rawY) is computed using an up to 7th

order polynomial. The coefficients of the polynomials are stored in the columns XPOLYCOEF, YPOLYCOEF of the extension POLYNOM_MAP. Unused coefficients are set to zero in the CCF file, e.g. if only a second order polynomial is used, then all higher order coefficients are zero. There is one set of coefficients (XPOLYCOEF, YPOLYCOEF) per filter element in the POLYNOM_MAP extension. The required binary table entry is identified by matching the value of the column FILTER_ID with the State Variable FilterID.

$$\begin{aligned}
&deltaX = \\
&ax_0 + ax_1 Y + ax_2 Y^2 + ax_3 Y^3 + ax_4 Y^4 + ax_5 Y^5 + ax_6 Y^6 + ax_7 Y^7 + \\
&ax_8 X + ax_9 XY + ax_{10} XY^2 + ax_{11} XY^3 + ax_{12} XY^4 + ax_{13} XY^5 + ax_{14} XY^6 + \\
&ax_{15} X^2 + ax_{16} X^2 Y + ax_{17} X^2 Y^2 + ax_{18} X^2 Y^3 + ax_{19} X^2 Y^4 + ax_{20} X^2 Y^5 + \\
&ax_{21} X^3 + ax_{22} X^3 Y + ax_{23} X^3 Y^2 + ax_{24} X^3 Y^3 + ax_{25} X^3 Y^4 + \\
&ax_{26} X^4 + ax_{27} X^4 Y + ax_{28} X^4 Y^2 + ax_{29} X^4 Y^3 + \\
&ax_{30} X^5 + ax_{31} X^5 Y + ax_{32} X^5 Y^2 + \\
&ax_{33} X^6 + ax_{34} X^6 Y + \\
&ax_{35} X^7 \\
&deltaYF = \\
&ay_0 + ay_1 Y + ay_2 Y^2 + ay_3 Y^3 + ay_4 Y^4 + ay_5 Y^5 + ay_6 Y^6 + ay_7 Y^7 + \\
&ay_8 X + ay_9 XY + ay_{10} XY^2 + ay_{11} XY^3 + ay_{12} XY^4 + ay_{13} XY^5 + ay_{14} XY^6 + \\
&ay_{15} X^2 + ay_{16} X^2 Y + ay_{17} X^2 Y^2 + ay_{18} X^2 Y^3 + ay_{19} X^2 Y^4 + ay_{20} X^2 Y^5 + \\
&ay_{21} X^3 + ay_{22} X^3 Y + ay_{23} X^3 Y^2 + ay_{24} X^3 Y^3 + ay_{25} X^3 Y^4 + \\
&ay_{26} X^4 + ay_{27} X^4 Y + ay_{28} X^4 Y^2 + ay_{29} X^4 Y^3 + \\
&ay_{30} X^5 + ay_{31} X^5 Y + ay_{32} X^5 Y^2 + \\
&ay_{33} X^6 + ay_{34} X^6 Y + \\
&ay_{35} X^7
\end{aligned} \tag{29}$$

where

deltaX, deltaY	offset in x- and ydirection from linear grid. The offset is defined as difference (true - linear) position
X	offset from boresight in x-direction ($rawX - 1024.5$)
Y	offset from boresight in y-direction ($rawY - 1024.5$)
ax_{nn}, ay_{nn}	coefficients of the 7th order polynomial stored in the columns XPOLYCOEF, YPOLYCOEF respectively.

Interpretation of offset vector by SAS: The offset vector (deltaX, deltaY) are used by the SAS to calculate the angular distance of a position to the boresight in the following way. First the linear offset (linX, linY) from the boresight (in units of pixel) is calculated by the following equation:

$$\begin{pmatrix} linX \\ linY \end{pmatrix} = \begin{pmatrix} rawX - deltaX \\ rawY - deltaY \end{pmatrix} \tag{30}$$

Note that the offsets (deltaX, deltaY) are subtracted from the detector coordinates (rawX, rawY), because the offsets are defined as the difference between the measured and the expected (i.e. linear grid) position. The angular separation from the boresight is defined as the linear offset (linX, linY) divided by the platescale (section 3.5.8).

3.5.5.2 Calling Parameters

3.5.5.2.1 Input

`rawX, rawY` list with subpixel coordinates in the PIXCOORD system. Real numbers.

`calc_mode` parameter indicating the way of calculating the distorion. Integer number with value either 0 or 1.

`n` optional input parameter defining the number of neighbouring entries to be considered for interpolation process. Integer number (≥ 4).
Default value is 4, i.e. only the four closest detector positions listed in the CCF extension are used in the 2-dimensional interpolation.

3.5.5.2.2 Output

`deltaX, deltaY` calculated X/Y offset in subpixels

`sigmaX, sigmaY` uncertainty associated with calculates offset deltaX/deltaY (in subpixels)

3.5.5.3 Used CAL State Variables

FilterId

3.5.5.4 List of CCF Components used

ASTROMET

3.5.6 CAL_omInverseDistortion

CAL_omInverseDistortion - returns the linear position from the distorted coordinates at a specified detector location

3.5.6.1 Procedure

The routine returns the linear position (deltaX, deltaY) from the distorted grid position for a specified detector location. The input may consist of a list of positions. The offset can be calculate in two different ways. Either the offset is obtained by interpolation of the coarsely mapped distortion or by direct evaluation of an up to 7th order polynomial. The way of calculation is selected by setting the input parameter calc_mode.

- If calc_mode is set to 0, then the distortion vector is calculated by interpolation the distortion map.
- If calc_mode is set to 1, then the distortion vector is calculated by evaluation of the polynomial.

calc_mode is set to 0:
the linear position (deltaX, deltaY) at the detector location (rawX, rawY) is derived from the distortion map stored in the extension FILTER-FilterId (e.g. FILTER-U, FILTER-V, FILTER-GRISM1 etc.).
As the distortion map is only coarsly sampled an interpolation is required to derive the distortion at the requested position in the same way as for Cal_omDistortion.

calc_mode is set to 1:
the linear position (deltaX,deltaY) at the location (rawX, rawY) is computed using an up to 7th order polynomial. The coefficients of the polynomials are stored in the columns XPOLYCOEF, YPOLYCOEF of the extension POLYNOM_MAP2. The processing is identical to that of Cal_omDistortion.

Note that the offsets (deltaX, deltaY) are subtracted from the detector coordinates (rawX, rawY), because the offsets are defined as the difference between the measured and the expected (i.e. linear grid) position.

3.5.6.2 Calling Parameters

3.5.6.2.1 Input

`rawX, rawY` list with subpixel coordinates in the PIXCOORD system. Real numbers.

`calc_mode` parameter indicating the way of calculating the disttorion. Integer number with value either 0 or 1.

`n` optional input parameter defining the number of neighbouring entries to be considered for interpolation process. Integer number (≥ 4).
Default value is 4, i.e. only the four closest detector positions listed in the CCF extension are used in the 2-dimensional interpolation.

3.5.6.2.2 Output

`deltaX, deltaY` calculated X/Y offset in subpixels

`sigmaX, sigmaY` uncertainty associated with calculates offset deltaX/deltaY (in subpixels)

3.5.6.3 Used CAL State Variables

FilterId

3.5.6.4 List of CCF Components used

ASTROMET

3.5.7 CAL_omGetColorTransform

CAL_omGetColorTransform - convert a colour and a magnitude from the natural into a standard system

3.5.7.1 Procedure

Colour transformations are supplied only for the three visual bandpass filter (U, B, V). Due to time constraints or due to bright source restrictions the normally used colour terms may not be available and non-standard colour transformations have to be used instead. For that purpose non-standard colour transformations are supported, which make use of the UV filters. Colour transformations making use of the UVM2 filter are not considered at all because of the strong dependence of the UVM2 brightness on interstellar absorption.
The accuracy achieved by a colour transformation differs between the various transformations. Therefore a priority is assigned to the various transformations, which should be followed by the calling task: if possible, the colour transformation with the highest priorities should be used first.
The following table shows the supported colour transformations sorted according to their priority (1 is highest). Only the listed combinations are possible.

FILTERID1	FILTERID2	priority
B	V	1
U	V	2
UVW1	V	3
UVW2	V	4
U	B	1
UVW1	B	2
UVW2	B	3
UVW1	U	1

The colour indices of the correction terms (mag_1-mag_2) are always defined as (shorter wavelength passband - longer wavelength passband), e.g. (u-b), (b-v) or (uvw1-u) etc.

A new keyword ALGOID is used in the CCF to associate parameters sets with different algorithm. An ALGOID keyword value of 1 corresponds to the transformation described in this version (2.0 or later) of the calibration handbook. An ALGOID keyword value of 0 corresponds to transformation as described in calibration handbook version 1.1 or earlier.

The CAL provides the following functionality (assuming the keyword ALGOID has a value of 1):

- verify that one of the input filter identifiers is either U, B or V.
- check whether the input is provided with `filter1` being at a shorter wavelength than `filter2`.

- compute the colour term (mag1-mag2) in the instrumental system
- select the applicable colour transformation
  - the transformations of the correct colours are selected:
    match the input filter identifier `filter1` and `filter2` to the filter identifier in the CCF column FILTERID1 and FILTERID2
  - the correct branch of the colour transformation is selected:
    match the value of the optional parameter `branch` (if not set the default value 0 is used) to the value in the column BRANCH of the CCF COLOURMAG extension.
  - the correct interval of the colour transformation is selected:
    the parameters of the colour interval are selected so that the colour term (mag1-mag2) is within the validity range of the colour transformation (as defined in equation 31).
- transform colour term (mag1-mag2) into standard system according to equation 33
- transform magnitude mag2 into standard system according to equation 34

The colour transformation can only be applied when a transformation exists between the specified filters `filter1` and `filter2` with the colour (mag1-mag2) falling within the validity range of the colour transformation equation and if the specified transformation branch `branch` is listed in the COLORMAG extension of the CCF ColorTrans component.

The colour in the instrumental system (mag1-mag2) must fall within the validity range of the colour transformation:

$$r_0 \leq (\texttt{mag1} - \texttt{mag2}) < r1 \tag{31}$$

where

r_0, r_1	defines the validity range of the colour transformation as defined in the column TRAFOLIMIT of the ColorTrans CCF component.
`mag1-mag2`	the colour index calculated from the two input magnitudes in the instrumental system.

and the parameters of the colour equation must be defined for the requested branch (specified by the optional input parameter `branch`) of the colour-colour diagram:

$$\texttt{branch} = branch \tag{32}$$

where

branch	value in column BRANCH in the COLORMAG extension of the CCF ColorTrans component.
`branch`	value of optional input parameter

The transformation of the colour term (mag1-mag2) into a standard system is described by the equation

$$(MAG1 - MAG2) = \sum_{i=0,9} k_i \cdot (\texttt{mag1} - \texttt{mag2})^i \tag{33}$$

where

k_i	are the parameters stored in the TRAFOP1 column of the ColorTrans CCF component.
`mag1-mag2`	the colour index calculated from the two input magnitudes in the instrumental system

The transformation of the instrumental magnitude mag2 into the magnitude MAG2 of a standard photometric system is described by the equation:

$$MAG2 = \sum_{i=0,n} (l_i \cdot (\mathtt{mag1} - \mathtt{mag2})^i) + \mathtt{mag2} \tag{34}$$

where

l_i	are the parameters stored in the TRAFOP2 column of the ColorTrans CCF component.
`mag1-mag2`	the colour index calculated from the two input magnitudes in the instrumental system
`mag2`	brightness in the instrumental system of `filter2`
MAG2	brightness in the standard system of `filter2`

3.5.7.2 Calling Parameters

3.5.7.2.1 Input

`filter1` filter associated with the magnitude `mag1`. The filter identifier `filter1` must be at a shorter wavelength as the filter defined by `filter2`

`filter2` filter associated with the magnitude `mag2`. The value must be one of the three filters: 'U', 'B' 'V'.

`mag1` zero point corrected magnitude derived from the deadtime corrected count rate measured in the passband `filter1`. This is the brightness which is used to calculate the colour (mag1-mag2) in the instrumental system (units: magnitude) .

`mag2` zero point corrected magnitude derived from the deadtime corrected count rate measured in the passband `filter2`. This is the brightness which is transformed into the standard system (units: magnitude).

`branch` optional integer parameter, used to select a colour transformation of a required branch in the Hertzsprung Russel diagram (equivalent to stars with different metallicities). The default value (=0) selects the transformation for main sequence stars.

3.5.7.2.2 Output

`stdcol12` colour derived from the input magnitudes `mag1` and `mag2` transformed into the standard system (units: magnitude)

`estcol12` uncertainty associated with output colour `stdcol12` (units: magnitude) `stdmag2` input magnitude `mag2` corrected to its standard system (units: magnitude)

`estdmag2` uncertainty associated with output magnitude `stdmag2` (units: magnitude)

3.5.7.3 Used CAL State Variables

none

3.5.7.4 List of CCF Components used

COLORTRANS

3.5.7.5 List of CAL Routines used

The whole procedure outlined before is done by using several routines:

`CAL_omGetColorTransformator` controls the pointers for the next routines

`CAL_omColorTransBranches` finds the available branches in the H-R diagram

`CAL_omColorTransValidityRanges` finds the number of limits for the selected transformation

`CAL_omColorTransValidityRange` finds the values of the limits

`CAL_omStandardColor` calculates the color index from two input magnitudes in the standard system

`CAL_omStandardMagnitude` transforms an input magnitude value from the instrumental into the standard system

3.5.8 CAL_omGetPlateScale

CAL_omGetPlateScale - obtain platescale for the specified filter

3.5.8.1 Procedure

The routine returns the platescale of the specified filter. The platescale is stored as keyword PLTSCALE in the the distortion map extension associated with a filter. The extension name is FILTER-FilterId (e.g. FILTER-U, FILTER-V, FILTER-GRISM1 etc.) The units of the platescale value are arcsec/pixel. The platescale defines the size of the pixel (1025,1025), which is the central OM pixel.

3.5.8.2 Calling Parameters

3.5.8.2.1 Input

`lambda` nominal wavelength of filter; currently dummy parameter

3.5.8.2.2 Output

`plateScale` Intrinsic plate scale of the specified filter [arcsec/subpixel]

3.5.8.3 Used CAL State Variables

FilterId

3.5.8.4 List of CCF Components used

ASTROMET

3.5.9 CAL_omGetPSFmap

CAL_omGetPSFmap - returns a mapped OM PSF for the specified filter

3.5.9.1 Procedure

This routine is used by the SAS.
The PSF is stored as a radial distribution in the CCF component PSF1DRB. The routine reads the PSF from the PSF1DRB CCF component, where the PSF with the closest CFRR to the input countFrameRateRatio is selected. The radial width of the PSF is stretched by the factor $f_{stretch}$ by multiplying the radial bin limits with $f_{stretch}$:

$$RMIN \rightarrow RMIN \cdot f_{stretch} RMAX \rightarrow RMAX \cdot f_{stretch}$$

The stretching factor $f_{stretch}$ is calculated for the specified field position (rawX, rawY) from a 3rd order polynomial.

$$f_{stretch} = a_0 + a_1 Y + a_2 Y^2 + a_3 Y^3 a_4 X + a_5 XY + a_6 XY^2 + a_7 X^2 + a_8 X^2 Y + a_9 X^3 \qquad (35)$$

where $X = (rawX - 1024.5)^i andY = (rawY - 1024.5)$.
The 10 polynomial coefficients $a_0, ..., a_9$ are listed as keywords in the extension headers:

keyword	variable name in equ. 35
PARX0Y0	a_0
PARX0Y1	a_1
PARX0Y2	a_2
PARX0Y3	a_3
PARX1Y0	a_4
PARX1Y1	a_5
PARX1Y2	a_6
PARX2Y0	a_7
PARX2Y1	a_8
PARX3Y0	a_9

The PSF map is generated from the radial distribution in the following steps:

1. identify PSF1DRB extension associated with FilterId
2. read polynomial coefficients in header
3. calculate the stretching factor f_{strech} for the specified field position (rawX, rawY)
4. read binary table extension

5. apply stretching factor f_{strech}
6. initialize PSF map as specified by input variables:
(nPixelsNegX+nPixelsPosX+1, nPixelsNegY+nPixelsPosY+1)
7. for each pixel of the PSF map
   - calculate the minimum distance r_{min} of the pixel to the PSF centre (rawX, rawY)
   - calculate the maximum distance r_{max} of the pixel to the PSF centre (rawX, rawY)
   - calculate the normalization factor f_{norm}:

$$f_{norm} = \frac{1}{[(RMAX - RMIN)^2 \cdot \pi] \cdot xRes \cdot yRes}$$

   - set the PSF pixel value to the interpolated intensity value at the radius $r = r_{min} + (r_{max} - r_{min})/2$
   - normalize PSF pixel value with the normalization factor f_{norm}
8. normalize PSF map to 1

3.5.9.2 Calling Parameters

3.5.9.2.1 Input

`rawX, rawY` centre of the PSF map specified in the PIXCOORD reference system (real)

`nPixelsNegX, nPixelsPosX, nPixelsNegY, nPixelsPosY` parameters to define the extent of the PSF map in the PIXCOORD system (integer). The map will cover the detector range from:
(rawX - nPixelsNegX:rawX+nPixelsPosX, rawY - nPixelsNegY:rawY+nPixelsPosY)

`xRes, yRes` resolution of PSF map (integer) (units: PSF bins per subpixel *(xRes, yRes ≥1)*). The output array will be of the dimension:
*((nPixelsNegX+nPixelsPosX+1)*xRes,(nPixelsNegY+nPixelsPosY+1)*yRes)*

`countFrameRateRatio` This parameter is equivalent to the average count rate in the region of r=4 subpixel (TBC) around the source. The parameter is used to select the PSF with the appropriate shape, as the shape changes due to coinciding events with increasing count rate. Units are counts/subpixel/CCD-frame.

3.5.9.2.2 Output

`PSFmap` 2-dimensional array containing the OM PSF for the specified filter at the specified position.

`binWidth` optional output parameter, which specified the width of a radial PSF bin in the CCF component PSF1DRB (in units of centroided pixel).

3.5.9.3 Used CAL State Variables

FilterId

3.5.9.4 List of CCF Components used

PSF1DRB

3.5.9.5 List of CAL Routines used

The whole procedure outlined before is done by using several routines:

`CAL_omGetPSF` controls the pointers for the next routines

`CAL_omPsfEncircledEnergy` computes the fraction of the energy contained in a circle of a given radius centered on the PSF centroid

`CAL_omPsfCircleRadius` computes the radius of a circle centered on the PSF barycentre containing a given fraction of the total energy

3.5.10 CAL_omLargeSenseVariation

CAL_omLargeSenseVariation - returns the flatfield component describing large scale variations of the instrument response

3.5.10.1 Procedure

The OM flatfield response is decomposed into two components, the small scale variation from subpixel to subpixel and the large scale variations with a scaling length above several CCD pixels. The overall flatfield is obtained by multiplying the two flatfield components. The flat field correction is applied to a science image by multiplication, thusthe inverse normalized sensitivity is returned by the CAL.
The large scale variations are mainly introduced by filter defects and contamination either on the filter or on the detector window. CAL_omLargeSenseVariation returns the flatfield component describing the large scale variations of the combination of detector plus specified filter. In case of the grism the zero and first order image are treated as different elements. Different extension of the CCF component LargeScaleSens are accessed for the various filter elements.

The coordinate system used in the CAL call is the PIXCOORD system. The image maps in the LrgScaleSens CCF components are sampled at a coarse resolution. The position (CRVAL1 and CRVAL2) and the address (CRPIX1 and CRPIX2) of the reference bin and the increment (CDELT1 and CDELT2) of the image array have to be interpreted when reading the extensions in the LrgScaleSens component. The position of the pixel (x,y) in an LrgScaleSens image extension corresponds to the position (CRVAL1+(x-CRPIX1)*CDELT1,CRVAL2+(y-CRPIX2)*CDELT2) in the PIXCOORD system.
The CAL has to interpolate the data to get from the coarser gridded map in the CCF file to a high resolution image required by the calling routine. Care must be taken at the edge of the FOV in the interpolation process. CAL_omLargeSenseVariation determines the large scale flat field response in the following way:

- determine filter setting
- read image array in Filter-*FilterId* extension of the LargeScaleSens CCF component
- read keywords AVERAGE, DAVERAGE and PIXERR and set them to the output variables ffavgDet(1:2) and ffavgPix respectively.
- extract area from image array as required by input parameters
  - interpret the positional keywords CRVALn, CRVALn and CDELTn
  - extract slightly enlarge area if required by interpolation
- interpolate extracted area
- extract output area from interpolated image

3.5.10.2 Calling Parameters

3.5.10.2.1 Input

`x0, y0` position of lower left corner of requested flatfield image in the PIXCOORD system. Validity range for both values is [1,2048] for both parameter. (Integer)

`xSize, ySize` size of requested flatfield image in x- and y-direction. Units are subpixel. The maximum size is 2048 for both parameter. (Integer)

`lambda` nominal wavelength of the flatfield. This parameter is currently not used by the algorithm. (real)

3.5.10.2.2 Output

`senseMap` Two-dimensional (real) array: inverse sensitivity map.
i.e. the flatfield correction is applied to a science image by multiplying with the senseMap.

`ffavgPix` optional output, average uncertainty of a single pixel value in units of percent (real)

`ffavgDet` optional output, two element real vector.
flatfield value averaged over all good pixel of entire detector and its associated uncertainty.

3.5.10.3 Used CAL State Variables

FilterId

3.5.10.4 List of CCF Components used

LARGESCALESENS

3.5.11 CAL_omLEDTemplate

CAL_omLEDTemplate - returns the LED illumination pattern on the requested position on the detector as 2-dimensional map.

3.5.11.1 Procedure

The extension LEDTEMPLATE of the OM CCF component PixToPixSens contains the normalized inverse illumination amplitude of each detector position. The array has a size of 2048x2048 elements. Bright areas in the internal flatfield data have assigned a value less than 1.0 and faint areas have a value of larger than 1.0. The SAS task applies the correction by multiplication of the returned array.

The function accesses the CCF LEDTEMPLATE map and returns an image with the requested size and resolution at the requested detector position. This functionality is identical to the functionality of CAL_omPixelSenseVariation (cf. section 3.5.15)

3.5.11.2 Calling Parameters

3.5.11.2.1 Input

`rawX0, rawY0` position of lower left corner of requested flatfield image in the PIXCOORD system. Validity range for both values is [1,2048] for both parameter. (Integer)

`rawXsize, rawYsize` extent of requested flatfield window on detector in x- and y-direction. Units are subpixel. The maximum size is 2048 for both parameter. (Integer)

3.5.11.2.2 Output

`LEDmap` Two-dimensional (real) array with extent rawXsize in x-direction and rawYsize in y-direction describing the normalized shape of the LED illumination pattern.

3.5.11.3 Used CAL State Variables

instrument

3.5.11.4 List of CCF Components used

PIXTOPIXSENS

3.5.12 CAL_omPhotoMagnitude

CAL_omPhotoMagnitude Converts count rates into magnitudes of the natural system

3.5.12.1 Procedure

The routine converts a corrected (flatfielded, background, deadtime) count rate into magnitude values of the natural system by applying the zero point correction. The input count rate has already been corrected for instrumental effects (as output from routine `CAL_PhotoNatural` (see sect. 3.5.14) and for background contributions. The definition of the zero point is retrieved from the extension COLORMAG of the CCF component `ColorTrans`. The zero-points of the different filters are stored as real number in the keywords ZPT*FilterID*:

ZPTU	zero point of U-filter
ZPTB	zero point of B-filter
ZPTV	zero point of V-filter
ZPTUW1	zero point of UW1-filter
ZPTUM2	zero point of UM2-filter
ZPTUW2	zero point of UW2-filter
ZPTMAGNI	zero point of magnifier
ZPTWHITE	zero point of white light-filter

The count rates conversion is made with the assumption that the full PSF is sampled. In case of partial PSF sampling the magnitudes have to be corrected by the calling task.

The transformation of the input count rate s_{nat} into the natural system m_{nat} (in units of magnitude) is described by the equation:

$$m_{nat} = zpt0 - 2.5 \log s_{nat} \tag{36}$$

where $zpt0$ is the zero point of the used filter.

The original routine has been aliased as `CAL_omNaturalMagnitude`.

3.5.12.2 Calling Parameters

3.5.12.2.1 Input

`naturalRate` a corrected count rate in the natural system as output by routine `CAL_PhotoNatural`

`naturalRateErrorMinus,naturalRateErrorPlus` absolute asymmetrical errors (plus/minus) associated with *magnitude* (optional arguments)

3.5.12.2.2 Output

`magnitude` a magnitude value in the natural system corresponding to the input count rate `magnitudeErrorMinus,magnitudeErrorPlus` absolute asymmetrical errors (plus/minus) associated with *magnitude* (optional arguments)

3.5.12.3 Used CAL State Variables

FilterId

3.5.12.4 List of CCF Components used

COLORTRANS

3.5.13 CAL_omGetFrameParameters

CAL_omGetFrameParameters Returns OM-CCD parameters needed for frametime calculation.

3.5.13.1 Procedure

The routine access the PhottoNat CCF file and reads the parameters in the field PARAM00 and DPARM00 of the PHOTTONAT extension, which characterise the OM CCD. The parameters of the binary field PARAM00 match to the CAL output parameters as follows:

CCF name	output name	description
PARAM00(1)	NHP	Number of horizontal pixels
PARAM00(2)	NVPS	Number of vertical pixels in store
PARAM00(3)	VTRANS	Vertical transfer time
PARAM00(4)	HTRANS	Horizontal transfer time
PARAM00(5)	NVPI	Number of vertical pixels in image
PARAM00(6)	V2H	Vertical to horizontal switch time
PARAM00(7)	H2V	Horizontal to vertical switch time
PARAM00(8)	OFFSET	Redundant CCD y-offset
PARAM00(9)	0.	not used
PARAM00(10)	0.	not used

DPARAM00(1:10) contain the associated uncertainties of the PARAM00 variables.

3.5.13.2 Calling Parameters

3.5.13.2.1 Input

None

3.5.13.2.2 Output

`NHP` Number of horizontal pixels (units: pixel/row)

`NVPS` Number of vertical pixels (units: pixel/column)

`VTRANS` Vertical transfer time (units: sec)

`HTRANS` Horizontal transfer time (units: sec)

`NVPI` Number of vertical pixels in image (units: pixel/column)

`V2H` Vertical to horizontal switch time (units: sec)

`H2V` Horizontal to vertical switch time (units: sec)

`OFFSET` Redundant CCD y-offset (units: pixel)

3.5.13.3 Used CAL State Variables

none

3.5.13.4 List of CCF Components used

PHOTTONAT

3.5.14 CAL_omPhotoNatural

CAL_omPhotoNatural returns OM raw counts corrected for instrumental effects

3.5.14.1 Procedure

The routine converts a raw input count rate from the instrumental into the natural system. The raw input count rate is the total (background and source) count rate extracted within the aperture defined by the radius used in the photometric calibration analysis. This aperture radius can be queried by CAL_omGetApertureRadius.

The routine corrects the raw count rate for CCD deadtime and detector non-linearity.

The detector non-linearity correction (coincidence loss correction) is not applied when the source extent flag is set. The count rates are corrected for deadtime after the coincidence loss correction was applied.

It is possible to select between two different versions of the linearity (=coincidence loss) correction. The user can select between a theoretical coincidence loss correction and an empirical linearity correction using the optional input parameter `linearity_type`. The empirical linearity correction is derived by default.
While the theoretical linearity correction is a hard coded formula (see pseudo code below), the empirical linearity applies an additional parameterized correction function to the theoretical formula. The coefficients of the empirical linearity correction function are stored in the extension `COINCIDENCE` of the tt PHOTTONAT CCF constituent.

The empirical coincidence correction multiplies the theoretical coincidence correction formula with an additional parameterized function f(x), where x is the count to framerate ratio:

$$f(x) = \sum_{i=1}^{n} a_i \cdot x^{(i-1)} \tag{37}$$

where

- x is the count to framerate ratio, i.e. `cts_detected * frametime`
- a_i parameters stored in column PLINFUNC of extension `COINCIDENCE` of `PhotToNat` CCF file
- n number of elements stored in column PLINFUNC of extension COINCIDENCE

```
;
function f_omraw2nat,eflag,frametime,deadfrac,cts_detected,$
 linearity_type=linearity_type
;
; apply deadtime and coincidence correction to the raw OM count rates
; 16/12/97 initial version
; 24/02/98 rename variables to make code more intuitive
; 03/02/00 swap coincidence loss and deadtime correction
```

```
;          set eflag to excluded extended sources from coincidence loss
;          correction


; 19/01/01 prepare for empirical linearity correction
; 12/02/01 include empirical linearity correction
;


;INPUT:
; eflag          logical to flag extended sources
; cts_detected   measured photon rate per second
; deadfrac       frametransfer time expressed as fraction of frametime
; frametime      CCD frametime in units of seconds


; optional INPUT:
; linearity_type  optional integer parameter selecting type of
;                 linearity correction (0=Igor's empirical, 1=theoretical)


;OUTPUT:
; photons_in      in-falling photon rate (not corrected for phd-fraction)

if eflag then  cts=cts_detected $

else begin

 ; correct for coincidence losses
 if cts_detected lt 1./frametime then begin



    ; calculate theoretical correction, which is needed anyhow
    cts=alog(1.-cts_detected*frametime)/frametime*(-1.)

    if linearity_type eq '0' then begin

      ; use empirical correction!!!
      ; apply additional correction term: Igor's parameterized function
      ; parameters are stored in column PLINFUNC of extension COINCIDENCE
      ; in PhotToNat CCF file
     x=cts_detected*frametime
     f_x=0.D0
     idim=n_elements(a_i)
     for i=0,idim-1 do f_x= f_x + a_i[i]* x^i
     cts=cts*f_x

    endif else if linearity_type eq '1' then begin
```

```
     ; no further correction for theoretical curve required
     cts=cts

    endelse
  endif else begin

  ; more counts than frames, either source is extended or very very bright

  cts=infinity ; (i.e. largest possible number*(1-deadfrac) )
  print,'warning: count rate gt frametime, rate set to largest possible number'


 endelse

endelse

   ;correct for frametransfer time
   photons_in=cts/(1.-deadfrac)


return,photons_in
end
```

A correction for the global loss of detector efficiency is applied optionally using the value of the PHDFRAC keyword in the PhotToNat CCF file. The value stored in PHDFRAC describes the fraction of detected events as measured at the time of the most recent photometric calibration update. The correction for the average detector ageing or change in the threshold setting is achieved by:

$$photons = photons_{in}/(1 - eventLoss) * PHDFRAC \tag{38}$$

3.5.14.2 Calling Parameters

3.5.14.2.1 Input

EFlag flag (logical) indicating whether a source is extended or not; flag set to true if source is extended.

rawRate uncorrected OM count rate (real) to correct; units are: counts/Aperture/sec. **Note** that the counts are raw counts (source+background).

frameTime CCD frame time in units of ms (real)

deadFrac deadtime caused by frametransfer expressed as fraction of CCD frame time (real)

eventLoss fraction of events (real) below detection threshold in the most recent PHD (optional argument) `rawRateErrorMinus,rawRateErrorPlus` absolute asymmetrical errors (plus/minus) associated with rawRate (real)

`linearity_type` optional integer parameter with default value 0;
0=Igor's empirical linearity correction is applied,
1=theoretical linearity correction is applied.

3.5.14.2.2 Output

`naturalRate` count rate corrected for instrumental effects; units: cts/sec (real)

`naturalRateErrorMinus, naturalRateErrorPlus` absolute asymmetrical errors (plus/minus) associated with computed natural Rate (real) (optional arguments)

3.5.14.3 Used CAL State Variables

FilterId (TBC)

3.5.14.4 List of CCF Components used

PHOTTONAT

3.5.15 CAL_omPixelSenseVariation

CAL_omPixelSenseVariation returns a pixel to pixel sensitivity map

3.5.15.1 Procedure

This function accesses the small scale variation CCF flatfield map (PixToPixSens) and returns an image with the requested size, for the requested detector position and at the maximum detector resolution.

The CCF PixToPixSens file comprises of a 2048x2048 image array. This file is read. The image section specified by the input parameters (x0,y0,xsize,ysize) are extracted and returned.

The algorithm assumes that the PixToPixSens file is at full detector resolution (2048x2048 pixel). A correction algorithm to adjust for different CFRR will initially not be implemented. The CAL directly extracts the pixel to pixel variation map from the image extension PIXTOPIXSENS.

In a later implementation of the CAL routine either a TBD correction algorithm will be implemented to correct for the difference in the CFRR or the image extension with the CFRR keyword value closest to the input parameter *countRateFrameRatio* is selected.

The optional output variables ffavgPix and ffavgDet(1:2) are set to the values of the keywords PIXERR, AVERAGE and DAVERAGE respectively.

3.5.15.2 Calling Parameters

3.5.15.2.1 Input

`x0,y0` coordinates of lower left corner of requested window in units of subpixel

`xsize,ysize` extent of requested window in units of subpixel

`countRateFrameRatio` average ratio of count to framerate in requested image
units are: [counts/subpixel/frame]

3.5.15.2.2 Output

`senseMap` two-dimensional normalized sensitivity map (real). The returned image is at maximum detector resolution.

`ffavgPix` optional output, average uncertainty of a single pixel value in units of percent (real)

`ffavgDet` optional output, two element real vector.
flatfield value averaged over all good pixel of entire detector and its associated uncertainty.

3.5.15.3 Used CAL State Variables

FilterId

3.5.15.4 List of CCF Components used

PIXTOPIXSENS

3.5.16 CAL_omGetApertureRadius

CAL_omGetAperRadius - returns the aperture radius used in the photometric calibration of the specified filter. The consistency of the aperture radii specified in the CCF components PSF1DRB and COLORTRANS is verified.

3.5.16.1 Procedure

The routines reads the aperture radius keywords of the specified filter (set by CAL State Variable FilterId) from the two CCF components PSF1DRB and COLORTRANS.
The aperture radii of the CCF PSF1DRB file are listed in the keywords `APERTURE` of the PSF extensions of the relevant filters PSF-*FilterId* (cf. section 4.5.10). The aperture radii used during the zero-point definition and the colour transformation are contained in the keywords `APE-`*FilterId* in the binary extension COLORMAG of the CCF component COLORTRANS (cf. section 4.5.4).

The keyword values (`APERTURE`) and (`APE-`*FilterId*) orginating from the PSF-*FilterId* extension of the PSF1DRB file and the COLORMAG extension of the COLORTRANS file are compared and in case of an agreement the keyword value is returned. In case of disagreement the routine returns an error.
The call is currently not supported for the grisms.

3.5.16.2 Calling Parameters

3.5.16.2.1 Input

None

3.5.16.2.2 Output

`aperture_radius` real number. The aperture radius used to compile the CCF files PSF1DRB and COLORTRANS in the photometric calibration of the specified filter.

3.5.16.3 Used CAL State Variables

FilterId

3.5.16.4 List of CCF Components used

PSF1DRB and COLORTRANS

3.5.17 CAL_getHKwindows

CAL_getHKwindows - returns limits of HKD to define GTI

3.5.17.1 Procedure

The OM CCF component HKParmInt is read and a character array containing HK window definitions are returned in the character array windowList. An optional input parameter hkparam can be set to the name of a specific HKD (e.g. “VOLTAGE1”). In this case only the limits of the specified HKD are returned, e.g. “VOLTAGE1>= 12.5 && VOLTAGE <=23.4”. The name and the limits of the returned HK parameter can be used as input to the task TABGTIGEN, which produces the GTIFILE.

3.5.17.2 Calling Parameters

3.5.17.2.1 Input

`hkparam` optional input parameter defining the name of a specific HK parameter. If this parameter is used, only the limits of the specified parameter are returned.

3.5.17.2.2 Output

`windowList` character array with HK window(s) definitions; suitable to be used as input to TABGTIGEN.

3.5.17.3 Used CAL State Variables

instrument

3.5.17.4 List of CCF Components used

HKPARMINT

type	CCF description	CAL encoding
turn on channel	T	10
hot CCD pixel	H	11
edge emission	E	12
dead CCD pixel	D	13
dead pore	P	14
bad CCD pixel	B	15
bad for unknown reason	U	16
area of reduced sensitivity (reduction of sensitivity in steps of 10%)	1-9	1 to 9

Table 1: Encoding of OM detector defects

3.5.18 CAL_getBadPixelMap

CAL_getBadPixelMap - returns bad pixel list as 2-dimensional map

3.5.18.1 Procedure

The OM CCF component BadPix is read and the entries of bad pixels falling into the specified area are extracted. A value of zero is assigned to defect free pixels in the output map. Bad pixels are marked in the output map by non-zero values. The value assigned to a bad pixel encodes the type of defect. The mapping between CCF description and the integer encoding used in the output badpixel map is as following:

3.5.18.2 Calling Parameters

3.5.18.2.1 Input

`rawX0, rawY0` coordinates of the lower left corner of a window in the PIXCOORD reference system (units: subpixel).

`rawXsize, rawYsize` extent of requested window on detector in x- and y-direction (units: subpixel).

3.5.18.2.2 Output

`map` two-dimensional array with extent rawXsize in x-direction and rawYsize in y-direction. The map value is zero for pixels without defect, otherwise an integer value is assigned, which encodes the type of defect.

3.5.18.3 Used CAL State Variables

instrument

3.5.18.4 List of CCF Components used

BADPIX

3.5.19 CAL_getBoresightMatrix

CAL_getBoresightMatrix - returns OM values of the Boresight misalignment matrix for astrometry computations

3.5.19.1 Procedure

A transformation from instrument reference system (CAMCOORD2) to the spacecraft reference system (SACOORD) by using the the Euler angles given in CCF `BoreSight`.

The time dependency prepares for future use, when parameterizations of time dependent misalignments (e.g. due to relaxations after slew and/or perigee passages) become available.

The original routine is aliased as `getBoresightMatrix`

3.5.19.2 Calling Parameters

3.5.19.2.1 Input

`time` event time coordinates in the TIMCOORD time frame (see section 2.2)

3.5.19.2.2 Output

misalignment matrix

3.5.19.3 Used CAL State Variables

instrument, temperature (check), observationStartTime

3.5.19.4 List of CCF Components used

BORESIGHT

3.5.19.5 List of CAL Routines used

A series of routines may be used to handle the boresight information:

`CAL_getBoresight` controls calls to other routines

`CAL_withSacCoordOrientation` handles the orientation of the SACCOORD coordinate frame

`CAL_projectOntoTelCoord` projects a list of sky positions onto the TELCOORD reference frame

`CAL_toEulerAngles` converts boresight misalignment data into 3-2-1 Euler angles

`CAL_toDirectionCosineMatrix1` obtains instrument misalignment as direction cosine matrix

4 CCF Components

This section contains descriptions of the structure of CCF constituents. These descriptions will however only be to the extent that all items that are relevant for the instrumental calibrations will be listed. Overall keywords that are required for the proper archiving and management of the CCF will be omitted, as they are to be detailed in [5].

The coordinates used inside the CCF files will be consistent with the coordinate conventions as is used by the SAS (see CAL documentation for a description). Consequently all coordinates specified in terms of `RAWX` & `RAWY` start at 1, which depending on the instrument, may be different than in telemetry.

The calibration team of the SOC is also providing additional CCF components where required in conjunction with data that were generated with SciSim. This is only a temporary solution in order to get calibrated test data on a short time scale for testing SAS task. The goal for the long run is to update SciSim such that it produces data similar to the instruments' performances, and thus eventually no special CCF components will be needed.

These CCF components are generated on a need basis and are identical in structure to the real life components. The difference between the instances of a real life CCF component and its SciSim applicable counterpart is that the `CATEGORY` keyword either is set to `XMMCCF`, or to `SCISIMCCF`, respectively.

As far as the CAL is concerned there is only one CCF, i.e., the one the user points at with the `SAS_CCFDIR` environment variable. The calibration index file (CIF) in that directory will tell the CAL how to find the various constituents it needs. However, this flexibility can be used at the CIF-building stage. The task cifbuild can locate CCF constituents along a specified search path (see cifbuild description, GV note). So, assuming the constituents of scisimccf/xmmccf are located in different directories the former does not need to contain the entire set if the latter does. In other words, scisimccf does only need to contain the scisim-specific parts (which cifbuild will pick if scisimccf comes BEFORE xmmccf in the CCF search path).

4.1 General CCF Components

4.2 Mirror Related CCF Components

The mirror calibration files describe the X-ray response of the three XMM mirror modules. As a baseline, there will be one CCF constituent per telescope. They can be distinguished with the header xrt1, xrt2 or xrt3.

4.2.1 XEncirEn

4.2.1.1 Description

The files provide the encircled energy function within the X-ray point spread function of each of the three mirror modules. The encircled energy functions are tabulated at discrete energies for an on-axis position of the source. The circular windows in which the encircled energy is calculated are assumed to be perfectly centered on the PSF.

- The table section provides the on-axis encircled energy fraction as a function of the radius of the circular window in mm. The tables are used by the CCF access layer to interpolate the on-axis encircled energy function of one mirror module to any energy.
- Scaling parameters (conv, cor1 and cor2) are provided in the headers of the XRT_XENCIREN CCF files. They are used to correct the on-axis encircled energy fraction for off-axis position of the extraction window. There units are arcsec/mm, arcsec^{-1} and arcsec^{-2}, respectively.

4.2.1.2 Generation from Ground Calibration Data

The initial CCF have been generated from batch simulation of the telescope performance using the scisim configuration files calibrated against the on-ground calibration tests.

4.2.1.3 Generation from In-Orbit Data

In-orbit measurements of the PSF have been used to validate the CCF.

NB: These files are obsolete. The encircled energy is now calculated by integrating a King function whose parameters are given in the XPSF CCF files.

4.2.2 XPSF

4.2.2.1 Description

The files describe the PSFs of the flight mirror modules xrt1, xrt2 or xrt3 vs field angle, energy and field azimuth. The xrt1, xrt2 or xrt3 telescopes are respectively associated with the MOS1, MOS2 and PN EPIC cameras. Each file will consist of a header, a table section and an image section.

- an alignment correction table is provided. It contains correction factors which account for the mirror module field curvature, EPIC defocus, mirror module tilt with respect to the boresight and EPIC decenter with respect to the average focal point of the mirror modules. Initially these correcting factors will be set to 0.

- a second table section contain a three dimensional table of 6 fitting coefficients provided as a function of 6 energies and 6 field angles. These coefficient will be used by the CCF access layer to generate analytical PSF descriptions using a simple bi-dimensional Gaussian fit.

- a third table section contain a table of 21 fitting coefficients. These coefficient will be used by the CCF access layer to generate analytical PSF descriptions using the bi-dimensional multi-Gaussian fit. The validity range of this analytical fit is limited to the central 10 arcmin of the field of view and to energies lower than 4.5 keV.

- the image section contains a collection of 36 FITS images obtained at 6 energies and 6 field angles. Theses images can be interpolated by the CCF access layer to any energy, field angle or field azimuth.

- table `KING_PARAMS` provides the parameters of the King function tabulated against Energy and off-axis angle.

  `ENERGY` Photon energy in eV.

  `THETA` Off-axis angle in radians.

  `PARAMS` Two parameters, core radius and slope for each ENERGY/THETA pair.

- table `BETAPSF` provides a set of parameters of the PSF as a function of RGS dispersion coordinate (Z-coordinate in `SACOORD`, being integrated in cross-dispersion. This is required for the RGS response generation. The table has the following columns:

  `TYPE` a flag on which function these parameters act; either 'G' or 'L' for *Gaussian* of *Lorentzian* parameterization, respectively.

  `NORM` Amplitude of Gaussian and normalization of Lorentzian, as described by parameters a and b in section 3.4.33, respectively.

  `WIDTH` either Gaussian σ, or Lorentzian FWHM, as described by parameters σ and w in section 3.4.33, respectively.

  `CENTER` Center of the distribution with respect to zero, as described by parameters μ and ν in section 3.4.33, respectively.

  Note: Inversion due to the mirroring effect of the gratings takes place as part of the CAL function `CAL_rgsgetXRTFigure` (see section 3.4.33).

4.2.2.2 Generation from Ground Calibration Data

The initial CCF will be generated from batch simulation of the telescope performance using the scisim configuration files calibrated against the on-ground calibration tests. Since these tests were performed with a finite source distance, this configuration file shall be updated and calibrated using in-orbit observations.

4.2.2.3 Generation from In-Orbit Data

The multi-Gaussian fit is based on in-orbit observations of bright p[oint sources. In particular, it accounts for the triangular shape of the FM2 telescope.

The King function parameters have been calculated from fits to the radial profiles of bright point sources observed in-orbit.

4.2.3 XAreaEf

4.2.3.1 Description

The files describe the x-ray effective area and vignetting function of each of the three mirror modules vs field angle and energy. Vignetting by the RGA is not taken into account. The files will each consist of a header and 2 table sections.

- The first table section provides the on-axis effective area as a function of energy between 0.1 to 20 keV in variable step of 0.1 keV to 0.001 keV close to Au absorption edges. This table is used by the CCF access layer to interpolate the on-axis effective area of one mirror module to any energy.
- The second table has 11 different lines for 11 different energies ranging from 100 eV to 15000 eV. Each line provide a set of 27 vignetting factors for off-axis angle ranging from 0 degree to 0.26 degrees per step of 0.01 degree.

4.2.3.2 Generation from Ground Calibration Data

The initial on-ground calibration file will be generated by the science simulator (scisim) using the mirror model files which have been calibrated from the X-ray pencil beam, partial aperture and full aperture tests of the mirror modules at the CSL and PANTER facilities.

4.2.3.3 Generation from In-Orbit Data

It is not foreseen to perform absolute in-orbit effective area measurements. However the vignetting function of the telescope has been verified on constant X-ray sources such as cluster of galaxies and supernovae remnants. Also, contamination effect if any during launch and telescope tube outgazing phase could induce CK and OK absorption edges as well as a modification of the mirror reflectivity close to the Au M absorption edges. Such effect were not detected in the spectra of lineless BL LAC objects. Bright BL LAC object with hard spectra have been used for that purpose in order to obtain a high enough ratio signal to noise in the 2.2 to 3.4 keV range of the Au M edges.

4.3 EPIC Related CCF Components

4.3.1 EMOS_ADUCONV

4.3.1.1 Description

The CCF contains three table extensions : `ADUCOEFF`, `OFFSET_GAIN` and `ENERGY_COMBINE`.

`ADUCOEFF`: Obsolete, it was supposed to be used by the operational Payload Monitoring Subsystem (PMS) for a rough approximation of energy to PHA conversion.

Attributes: None.

Columns:

`CCD_ID`: CCD index number (1–7).
`NODE_ID`: 0/1 prime/redundant node.
`OFFSET`: One global offset value per CCD.
`GAIN`: One global gain value per CCD.

`OFFSET_GAIN`: With respect to the SAS, the detailed conversion is provided by the coefficients of this second extension. Three polynomial coefficients for the gain conversions are given for the four different pattern types (mono- (0), bi- (1-4), tri- (5-8) and quad-pixels (9-12)) and for the seven CCDs: a constant term (OFFSET), linear term (GAIN) and a quadratic term (POLY_TERM).

The polynomial coefficents are at present repeated for the redundant node, but not used in in-orbit operations, so pending necessity to recalibrate the latter.

Accessed by:

- `CAL_gainCorrect()`:
  p=0: $eOut = OFFSET[\_n] + GAIN[\_n] \times eIn + POLY\_TERM[\_n] \times eIN^2$
  p=1: $eOut = OFFSET1[\_n]+GAIN1[\_n]\times eIn+POLY\_TERM1[\_n]\times eIN^2$
  p=2: $eOut = OFFSET2[\_n]+GAIN2[\_n]\times eIn+POLY\_TERM2[\_n]\times eIN^2$
  p=3: $eOut = OFFSET3[\_n]+GAIN3[\_n]\times eIn+POLY\_TERM3[\_n]\times eIN^2$
  where _n is the node and p in the range [0,3] is a pattern index computed from event pattern passed to CAL_gainCorrect() as input argument :

Pattern ID	Pattern index
0	0
1,2,3,4	1
5,6,7,8	2
9,10,11,12	3
13 ...	3

- `CAL_offsetCorrect()`: $eOut = eIn - OFFSET[\_node]$. An alternative MOS offset correction could be obtained via CAL_getMOSoffsets(), which makes use of EMOS1/2_DARKFRAME.CCF.

- `CAL_piToeV()`: $eOut = KEV0 + eIn \times KEV\_P\_PI$.

Attributes:

`KEV0`: Constant in linear PI to keV conversion.

`KEV_P_PI`: Slope of linear PI to keV conversion.

Columns:

`CCD_ID`: CCD index number (1–7).

`OFFSET`: vector column of length 2; one offset value for pattern index 0 events, one value per node per chip.

`GAIN`: vector column of length 2; one gain value for pattern index 0 events, one value per node per chip.

`POLY_TERM`: vector column of length 2; quadratic term for pattern index 0 events, one value per node per chip in gain correction.

`OFFSET1/2/3`: vector column of length 2; one offset value for pattern index 1/2/3 events, one value per node per chip.

`GAIN1/2/3`: vector column of length 2; one gain value for pattern index 1/2/3 events, one value per node per chip.

`POLY_TERM1/2/3`: vector column of length 2; quadratic term for pattern index 1/2/3 events, one value per node per chip in gain correction.

`ENERGY_COMBINE`: Accessed by :

- `CAL_mosPhaBuild (in package emsalib)`. This routine combines the four MOS event energies ENERGY_1/2/3/4 into a single PHA value according to an algorithm described in the documentation of package emsalib. The table contains for every event pattern (0-31) a set of four coefficients a1=ENERGY_PARAM[0], a2=ENERGY_PARAM[1], a3=ENERGY_PARAM[2], a4=ENERGY_PARAM[3]; depending on the sign of a4, one of the two schemes is selected :
  - for $a4 > 0$: $PHA = a1 \times ENERGY\_1 + a2 \times ENERGY\_2 + a3 \times ENERGY\_3 + a4 \times ENERGY\_4 - BKG$, where BKG is a local weighted background value.
  - for $a4 < 0$ the formula is more complicated; see emsaplib documentation for more details.

Attributes: None.

Columns:

`PATTERN_ID`: event pattern ID in valid range [0, 31]

`ENERGY_PARAM`: vector column of length 4; four parameters (a1/a2/a3/a4) per PATTERN_ID

4.3.1.2 Generation from Ground Calibration Data

From the Orsay data, events were selected on a pattern by pattern basis, and a further selection made based on whether anomalous exceding of E3 thresholds. A polynomial fit of combined PHA values against energy was made, and the fit inverted to provide the CCF coefficients.

See document EPIC-LUX-RE-166, by S Sembay and P Bennie.

4.3.1.3 Generation from In-Orbit Data

Maintenance of the CCF components relies mainly on observing the emission lines of the internal calibration source and adjusting the coefficient in several time periods (the same intervals as the CTI) since launch.

Further analysis is made by checking spectral residuals near fluorescent background lines and absorption edges.

Note that from issue 12 of these CCFs onwards all quadratic terms have been set to zero, i.e. a linear relationship is assumed between the charge detected from an event and the energy of the detected X-ray hence :

$$E_{eV} = gain \times E_{charge} + offset$$

4.3.2 EPN_ADUCONV

4.3.2.1 Description

This CCF contains the following five extensions: `ADUCOEFF`, `OFFSET_GAIN`, `GAIN-HIGH`, `REEMISSION` and `REEMISSION1`.

Keywords to describe the contents of the CCF:

`BPT_CODE`: Code of the bad pixel table (BPT) version, which includes the additional column offset values, described in the `OFFSET_GAIN` extension. Owing to changes in the uploaded BPT and periods of operational changes, the number and value of additional column offsets has altered frequently from observation to observation. These changes are reflected in the corresponding CCFs. In view of the resulting large number of issues, a procedure has been implemented in the SAS to check whether the correct CCF is used for a given observation. This is done by correlating the uploaded BPT version, which is uniquely coded through the pn HK parameters F1629, F1630 and F1628, with the associated CCF header keyword BPT_CODE. A warning is issued if the two do not tally. The coding of the BPT version has been implemented since revolution 293 and the corresponding BPT_CODE keyword will be present in EPN_ADUCONV CCF issues as of 0108.

The CCF contains the following tables:

ADUCOEFF: Obsolete.

Attributes: None.

Columns:

CCD_ID: CCD index number (1–12).

NODE_ID: Node (0 for pn).

OFFSET: One global offset value per CCD.

GAIN: One global gain value per CCD.

OFFSET_GAIN: Accessed by:

- CAL_offsetCorrect(): There are two versions of this table with ALGOID=0 (or missing ALGOID) and ALGOID=1:

  ALGOID=0: $eOut = eIn - OFFSET[\_node]$

  ALGOID=1: $eOut = eIn - (OFFSET\_CONST[\_node] + deltaT * OFFSET\_SLOPE[\_node])$

  $deltaT = \_observationStartTime - REFTIME$
- CAL_gainCorrect(): [on _accuracyLevel == LOW]

  $eOut = eIn/GAIN[\_node]$
- CAL_piToeV():

  $eOut = KEV0 +$
  $eIn * KEV\_P\_PI$

Attributes:

REFTIME: Reference time for ALGOID=1 offset correction; format *yyyy-mm-ddThh:mm:ss.*

KEV0: Constant in linear PI to keV conversion.

KEV_P_PI: Slope of linear PI to keV conversion.

Columns:

CCD_ID: CCD index number (1–12).

GAIN: Vector column of length 64; one gain value per node per chip.

OFFSET: Vector column of length 64; one offset value per node per chip.

OFFSET_CONST: Vector column of length 64; constant in linear offset value calculation (only present for ALGOID=1).

OFFSET_SLOPE: Vector column of length 64; slope in linear offset value calculation (only present for ALGOID=1).

GAIN_HIGH: Accessed by CAL_pnGainCorrect(), implements I/F to the detailed pn gain model code in the calpnalgo package.

Attributes: FINEPAR1, FINEPAR2 and FINEPAR3: three constant coefficients needed in the gain model.

Columns:

GAIN_PAR: Vector column of length $768 = 12 * 64$; 10 rows with various parameters of the gain model code in calpnalgo. The correspondence between row numbers and code variables is: row 1–4: $pp0/pp1/pp2/pp3$; row 7: aa; row 9: bb; the rest are dummy values.

REEMISSION: Accessed by `CAL_pnReEmissionThreshold()`.
$thresh = A[0] + A[1] * EkeV + (B[0] + B[1] * EkeV) * (rawY - RAWY0)$ $EkeV = 1000 * Eadu * ADU\_P\_EV$, where $Eadu$ and $rawY$ are inputs to the routine.

Attributes:

ADU_P_EV: ADU units per eV.

RAWY0: Reference Y position.

Columns:

A: Vector of length 2; re-emission parameter.

B: Vector of length 2; re-emission parameter.

REEMISSION1: Obsolete. Was initially implemented as a more complex form of the re-emission threshold; the code is no longer in the CAL.

Attributes: None.

Columns:

C_K: Parameter for old re-emission model; obsolete.

D_K: Parameter for old re-emission model; obsolete.

E_K: Parameter for old re-emission model; obsolete.

F_K: Parameter for old re-emission model; obsolete.

4.3.2.2 Generation from Ground Calibration Data

Ground calibration data obtained from extensive measurements of X-ray emission lines and synchrotron lines (mainly flat fields). The gains of the individual readout nodes are derived by comparing the positions of the CTE corrected calibration lines (in ADU) with their energies.

4.3.2.3 Generation from In-Orbit Data

Monitoring of the gain through emission line centroiding using calclosed data and suitable celestial targets such as SNRs.

4.3.3 EMOS_BACKGROUND

4.3.3.1 Description

Currently not used in the SAS, but may be in the future. Is a placeholder that was supposed to have either some real background spectra or some coefficients with which to generate background spectra. It is currently not used, however might by used for a future task that takes as input some proxies like the RA & Dec, the ROSAT broad band count rates for the target location, the corner count rates etc. and generates a best guess background file.

One extension containing calibrated event list of low background data, with brighter point sources removed.

The CCF contains the following tables:

`PARTICLE` : Accessed by :

- `CAL_getParticleBackground()`: Obtain background particle spectrum in PI channel space according to the model :
  $bg = PAR[0] + PAR[1] \times PI[Hz/mm^2/chan]$

Attributes:

`EV_2_PI`: eV per PI channel (unused)

Columns:

`PAR`: model parameters; two-row scalar columns

`X_RAY_DIFFUSE` : Accessed by :

- `CAL_getDiffuseXBackground()`: Returns a ready made diffuse X-ray spectrum in PI space from the data in the SPECTRUM column modified by a second order polynomial:
  $spec = SPECTRUM \times (PAR[0] + PAR[1] \times r + PAR[2] \times r^2)$
  where r = radial offset from on-axis [mm]; is input to routine.

Attributes:

`EV_2_PI`: eV per PI channel (unused)

Columns:

`FILTER_ID`: meant to be the filter ID but contains currently an invalid entry and is ignored by the CAL

`SPECTRUM`: unmodulated PI background PI spectrum

`X_RAY_DIFFUSE_MODULATION` : Accessed by :

- `CAL_getDiffuseXBackground()`: see above.

Attributes: none.

Columns:

`PAR`: 3-row scalar column with modulation parameters (see above).

4.3.3.2 Generation from In-Orbit Data

Probably derived from early Cal/PV data by D. Lumb.

4.3.4 EPN_BACKGROUND

4.3.4.1 Description

Currently not used in the SAS, but may be in the future. Is a placeholder that was supposed to have either some real background spectra or some coefficients with which to generate background spectra. It is currently not used, however might by used for a future task that takes as input some proxies like the RA & Dec, the ROSAT broad band count rates for the target location, the corner count rates etc. and generates a best guess background file.

4.3.5 EMOS_BADPIX

4.3.5.1 Description

Provides the list of bad (hot/dead/flickering) CCD pixels, either uploaded on-board or characterized on-ground.

The CCF contains the following tables:

`BADPIX` Accessed by :

- `CAL_getBadPixelMap()`:
- `CAL_getBadPixelList()`: Provides the list of bad (hot/dead/flickering) CCD pixels.

Attributes: None.

Columns:

`CCD_ID` **:** the CCD index number (1–7)
`NODE_ID` **:** node name (either 0 or 1 for MOS)
`RAWX` **:** PIXCOORD1, start of bad pixel region in `RAWX`
`RAWY` **:** PIXCOORD2, start of bad pixel region in `RAWY`
`YLENGTH` **:** length of bad column (`YLENGTH` = 1 indicates a single bad pixel)
`DESCRIPTION` **:** type of the bad pixel listed (see below)
`MODE_N/A` **:** mode for which bad pixels are not applicable

Note that `RAWX` & `RAWY` will be in absolute coordinates on the CCD chip, and are thus independent of any windowing information.

A bad CCD pixel can be (value of `DESCRIPTION`):

- *dead*: no signal is ever detected in this pixel. All pixels that are parallel transferred through a dead pixel are dead as well. (type value equals 'd' if uplinked, equals 'D' if stored in CCF only)

- *hot*: a pixel that contains a signal almost at every readout of the CCD. The charge content of such a pixel is nearly constant, and is dependent on the CCD bias and readout parameters. (type value equals 'h' if uplinked, equals 'H' if stored in CCF only)
- *flickering*: the characteristics of flickering pixels depend on the CCD bias and read-out parameters, as well as on the average charge that is transferred through them during readout. (type value equals 'f' if uplinked, equals 'F' if stored in CCF only)

4.3.5.2 Generation from Ground Calibration Data

Listswere provided by EPIC from unit level ground calibrations.

4.3.5.3 Generation from In-Orbit Data

Bad pixel lists are derived mostly from the internal calibration exposures, acquired at regular intervals since the beginning of the mission via dedicated automatic monitoring scripts at the SOC.

Exposures without on-board bad pixel masking, acquired at much lower frequency are also used to detect monitor defects from micrometeoroid impacts: high energetic hot pixels with a very high recurrence frequency, that create small dead area (3x3 or 3x4 pixels).

Diagnostic and science exposures can also be used to identify bad pixels. Diagnostic exposures can also be used to confirm the presence of previously identified and up-linked bad pixels.

The integration of the data over each exposure can be used to identify hot or flickering pixels (pixels with repeated occurrence in excess of the local Poisson probability distribution of X-ray events), for example running SAS task *badpixfind* or *embadpixfind*.

4.3.6 EPN_BADPIX

4.3.6.1 Description

This file contains a single extension, `BADPIX`, describing the locations and several characteristics of the pn CCD bad pixels. These are pixels which exhibit a signal in a high proportion of readout frames and/or anomalously high signals. Depending on operating mode, most of the bad pixel locations are uploaded and their signals blanked out by the on-board electronics. In addition, in order to reduce noise, several rows of pixels at the read-out end of the CCDs are similarly blanked out.

Keywords to describe the contents of the CCF:

`BPT_CODE`: Code of the bad pixel table (BPT) version described in the CCF. Owing to the occurrence of new bad pixels and periods of operational changes, the number and location of blanked-out bad pixels included in the BPT has altered frequently from observation to observation. These changes are reflected in the corresponding CCFs. In view of the resulting large number of issues, a procedure has been implemented in the SAS to check whether the correct CCF is used for a given observation. This is done by correlating the uploaded BPT version, uniquely coded through the pn HK parameters F1629, F1630 and F1628, with the associated CCF header keyword BPT_CODE. A warning is issued if the two do not tally. The BPT coding has been implemented since revolution 293 and the corresponding BPT_CODE keyword will be present in EPN_BADPIX CCF issues as of 0093.

The CCF contains one table:

`BADPIX`: This table is accessed by `CAL_getBadPixelMap()`, `CAL_getBadPixelList()` and `CAL_pnCtiCorrect()`.

Attributes: None.

Columns:

`CCD_ID`: CCD index number (1–12).

`NODE_ID`: Node ID (0 for pn).

`RAWX`: Start of bad pixel region in `RAWX` (absolute coordinates on chip, thus independent of any windowing information).

`RAWY`: Start of bad pixel region in `RAWY` (absolute coordinates on chip, thus independent of any windowing information).

`YLENGTH`: Length of bad column (`YLENGTH` = 1 indicates a single bad pixel).

`DESCRIPTION`: Type of the bad pixel listed. A bad CCD pixel can be:

- *dead*: no signal is ever detected in this type of pixel. All pixels that are transferred through a dead pixel are dead as well. (Type value equals 'd' if up-linked, 'D' if stored in CCF only);
- *hot*: a pixel that contains a signal almost at every readout of the CCD. The charge content of such a pixel is nearly constant, and is dependent on the CCD bias and readout parameters. (Type value equals 'h' if up-linked, 'H' if stored in CCF only);
- *flickering*: the characteristics of flickering pixels depend on the CCD bias and readout parameters, as well as on the average charge that is transferred through them during readout. (Type value equals 'f' if up-linked, equals 'F' if stored in CCF only).

`MODE_N/A`: Mode(s) for which bad pixels are not applicable.

`N_SAT_IMP`: Hot pixel brightness.

4.3.6.2 Generation from Ground Calibration Data

The initial bad pixel list was provided by the EPIC team from unit level ground calibrations.

4.3.6.3 Generation from In-Orbit Data

Diagnostic exposures, science exposures and offset maps can be used to identify bad pixels and to confirm the presence of previously identified and up-linked bad pixels.

The integration of the data over each exposure can be used to identify hot or flickering pixels (pixels with repeated occurrence in excess of the local Poisson probability distribution of X-ray events), for example running the SAS task *badpixfind*.

4.3.7 EMOS_CALSOURCEDATA

4.3.7.1 Description

Not Used. It was meant to hold data about the calibration sources. One extension per CCD containing intensity distribution as events / second in RAWX/RAWY coordinates One extension per CCD containing spectra in PI units per CCD additional keywords provide a reference time and half life constants to correct for time–dependent count rate.

The CCF contains one table:

`CALSOURCEREGIONS:`

Attributes: None.

Columns:

- `CCD_ID`: CCD index number (1–7).
- `N_PEAKS`: number of peaks in cal source spectrum.
- `PHA_SEARCH_LO`: vector column of length N_PEAKS; lower PHA channel bound to bracket the peak.
- `PHA_SEARCH_HI`: vector column of length N_PEAKS; upper PHA channel bound to bracket the peak.
- `PHA_CALC_RANGE`: ???
- `EXPECTED_COUNT_RATE`: expected count rate from peak (units?) .

4.3.8 EPN_CALSOURCEDATA

4.3.8.1 Description

Not Used. This CCF was a placeholder to contain the intensity distribution versus energy for the calibration sources, and also some half life data so that one could compare eventually the ratios of fluxes to determine if there were any spatially dependent contamination layers. It is unlikely ever to be used.

4.3.9 EMOS_CTI

4.3.9.1 Description

A new CCF structure has been defined to support the new CTI correction algorithm ($ALGOID = 2$) used by the CAL CtiCorrector.cc

In extension 'CTI_EXTENDED' there are three columns for the serial CTI (CTI_X columns) and three columns por the parallel CTI (CTI_Y columns) that describes the CTI degradation per CCD, node and mode (same correction for both nodes and all modes) :

- the constant term (coefficients [0] in 3.3.9 equations)
- the linear degradation rate (coefficients [1] in 3.3.9 equations)
- the power law index (coefficients [2] in 3.3.9 equations)

In extension 'CTI_COLUMN' we can take account for step-like changes within the column offsets. Ignoring these steps would lead to an over- or understimation of the real CTI losses. The table contains offset values for all columns, in case no correction is needed, the offset value is zero with RAWY_START=0 and YLENGTH=600.

For each major period where the CTI trend displayed different degradation rates, there is one CCF per MOS and period.

Note that the extension 'CTI' is obsolete and not maintained anymore. This approximate set of CTI parameters was originally intended for QLA (and/or for SciSim ?).

The CCF contains three tables:

`CTI` : This table is not used by the CAL.

Attributes: None.

Columns:

CCD_ID: CCD index number (1–7).

CTI_X: single scalar fractional charge loss in serial direction.

CTI_Y: single scalar fractional charge loss in parallel direction.

CTI_EXTENDED: This table is accessd by CAL_mosCTIcorrection()

Attributes:

REF_DATE: reference date in ISO format yyyy-mm-ddYhh:mm::ss

CTX_RATE: serial CTI rate change coefficient for ALGOID=0 only, so not used anymore.

CTY_RATE: parallel CTI rate change coefficient for ALGOID=0 only, so not used anymore.

Columns:

MODE_ID: numeric ID of the mode.

CCD_ID: CCD index number (1–7).

NODE_ID: ID of the readout node [0, 1].

CTI_FRAME_TRANSFER: (only in ALGOID=0 CCFs) the column is not used by the CAL

D_CTI_FRAME_TRANSFER: (only in ALGOID=0 CCFs) absolute error of CTI_FRAME_TRANSFER values; column not used by the CAL.

CTI_X: fractional charge loss parameter in serial direction; vector column with size 2/4/3 corresponding to ALGOID=0/1/2

D_SERIAL_CTI: (only in ALGOID=0 CCFs) absolute error of CTI_X values; column not used by the CAL.

CTI_Y: fractional charge loss parameter in parallel direction; vector column with size 2/4/3 corresponding to ALGOID=0/1/2

D_CTI_PARALLEL: (only in ALGOID=0 CCFs) absolute error of CTI_Y values; column not used by the CAL.

CTI_COLUMN: This table is accessed by CAL_mosCTIcorrection()

Attributes: None.

Columns:

CCD_ID: CCD number [1-7].

NODE_ID: CCD readout direction (node identifier) [0,1].

RAWX: CCD column number RAW-X [0-599].

RAWY_START: RAW-Y start position of a column segment within a column [0-599].

YLENGTH: Length of the column segment [1-600].

OFFSET: The offset of the column segment in energy (ADU).

4.3.9.2 Generation from ground data

Using various monochromatic emission line stimulations, the values of PHA output as a function of spatial location were calculated, and fitted to the charge loss measured per pixel (average).

These values derived from ground-based data were used in the initial CCFs, making use of ALGOID=0, and refined in ALGOID=1, where a first attempt to model the in-flight time-dependent CTI degradation rate was made (SAS v5.3)

4.3.9.3 Generation from in-flight data

Since SAS v5.4 and the use of the new CTI correction algorithm (ALGOID=2) that applies well for both the $-100°C$ and the $-120°C$ CCD operating temperature a complete new set of parameters has been derived from in-flight trend-analysis and implemented as described above.

4.3.10 EPN_CTI

4.3.10.1 Description

The pn CTI CCF holds all information to perform the correction for charge transfer loss for every detected event. The corrections depend on the specific readout of the CCDs for each observation mode.

The CCF contains the following tables:

`CTI` The table is not used by the CAL; probably only by QLA but probably neither because the values appear all dummy at present

Attributes: None.

Columns:

- `CCD_ID`: CCD number
- `CTI_X`: single scalar fractional charge loss in serial direction
- `CTI_Y`: single scalar fractional charge loss in parallel direction

`CTI_EXTENDED` The table is used/accessed by: CAL_ctiCorrect() Generic CTI correction algorithm: $eOut = \frac{eIn}{1-CTI\_X^{RAWX}(1-CTI\_Y)^{RAWY}}$

Attributes: None.

Columns:

- `MODE_ID`: numeric ID of the mode.
- `CCD_ID`: CCD number [1, 12]
- `NODE_ID`: ID of the readout node;. in the case of EPN this is identical to RAWX-1
- `CTI_FRAME_TRANSFER`: the column is not used by the CAL; can be dropped in future version without consequences

D_CTI_FRAME_TRANDFER: absolute error of CTI_FRAME_TRANSFER values; column not used by the CAL; can be dropped in future version

CTI_X: fractional charge loss in RAWX direction; makes no sense for EPN but needs to be kept for commonality with EMOS

D_SERIAL_CTI: absolute error of CTI_X values; column not used by the CAL; can be dropped in future versions

CTI_Y: fractional charge loss per transfer in RAWY direction

D_CTI_PARALLEL: absolute error of CTI_Y values; column not used by the CAL; can be dropped in future versions

HOT_PIXELS The table is used/accessed by: CAL_pnCtiCorrect() [only up to including cal-3.117 (SAS-5.2)] Table containing the list of hot pixels (not-uplinked bad pixels of type 'H') to be passed to the high-accuracy CTI correction algorithm in calpnalgo. As of cal-3.118 this information is read from EPN_BADPIX:BADPIX - table is thus obsolete and can be dropped in future versions of this CCF

Attributes: None.

Columns:

QUAD_ID: hardware quadrant ID [0, 3]

CCD_ID: hardware CCD number [0, 2]

RAWX: PIXCOORD0 X-coordinate [0, 63]

RAWY: PIXCOORD0 Y-coordinate [0, 199]

YLENGTH: same as EPN_BADPIX:BADPIX:YLENGTH

N_SAT_IMP: same as EPN_BADPIX:BADPIX:N_SAT_IMP

DISCARDED_PIXELS The table is used/accessed by: CAL_pnCtiCorrect() [only up to including cal-3.117 (SAS-5.2)] Table containing the list of discarded pixels (uplinked hot pixels (type 'h') which are not telemetered by the event analyzer logic, i.e. appear as logically dead) to be passed to the high-accuracy CTI correction algorithm in calpnalgo. As of cal-3.118 this information is read from EPN_BADPIX:BADPIX - table is thus obsolete and can be dropped in future versions of this CCF

Attributes: None.

Columns:

QUAD_ID: hardware quadrant ID [0, 3]

CCD_ID: hardware CCD number [0, 2]

RAWX: PIXCOORD0 X-coordinate [0, 63]

RAWY: PIXCOORD0 Y-coordinate [0, 199]

QUAD_ID: hardware quadrant number [0, 3]

CCD_ID: hardware CCD number [0, 2]

CTI-HIGH The table is used/accessed by: CAL_pnCtiCorrect() Main table to drive the high-accuracy EPN CTI correction algorithm in calpnalgo; Contains four sets of parameters per CCD column; numbers come from hardware group at MPE

Attributes:

EPERADU: number of released electron per ADU value (needed by calpnalgo)

KEVPER_E: energy in keV per released electron (needed by calpnalgo)

Columns:

CTI_PAR: vector column of length 768=12*64 with CTI model parameter; there are four rows corresponding to four main parameter sets

CTI-HIGH_ADD_PAR The table is used/accessed by: CAL_pnCtiCorrect() Table containing additional parameter needed by the CTI correction in calpnalgo

Attributes: None.

Columns:

Y_SHIFT: shift in Y-direction (vector column of length 6 corresponding to modes)

SCALE: scale factor for CTI correction (only != 1 for TI mode)

MODE_FACTOR: mode-dependent factor used in CTI correction (vector column length 6)

EFF_PAR: four parameter values controlling CTI correction in eFF mode

LWSW_PAR: 16 parameter values controlling CTI correction in LW/SW mode

LW_PAR: 16 parameter values controlling CTI correction in LW mode only (to be used in preference to the LWSW_PAR values).

SW_PAR: 16 parameter values controlling CTI correction in SW mode only (to be used in preference to the LWSW_PAR values).

CCD_OFSETS The table is used/accessed by: CAL_pnAdditionalGainCtiCorrect() Holds single scalar offset values used by EPN gain correction in calpnalgo

Attributes: None.

Columns:

CCD_OFFSETS: scalar offset values - one row per CCD

TEMP_CTI The table is used/accessed by: CAL_pnAdditionalGainCtiCorrect() Parameter controlling temperature-dependent stage of overall gain correction; correction is linear in T: tgain = OFFSET + T*SLOPE

Attributes:

ADU_REF: Mn K-alpha reference energy in ADU for T-dependent gain correction

Columns:

OFFSET: offset of linear T-gain correction

SLOPE: slope of linear T-gain correction

ORSAY_GAIN The table is used/accessed by: CAL_pnAdditionalGainCtiCorrect() Parameters controlling Orsay gain scaling stage of overall gain correction

Attributes: None.

Columns:

ORSAY_GAIN: scalar gain scaling values; one row per CCD

LONG_TERM_CTI The table is used/accessed by: CAL_pnAdditionalGainCtiCorrect() Parameter controlling long-term CTE degradation correction which is modeled as a gain scaling in calpnalgo

Attributes:

`ALGOID`: algorithm identifier.

Columns:

`MODE_ID`: numeric ID of the mode.

`CCD_ID`: CCD number [1, 12].

`T_COEFF`: time dependence of long term CTI; four-element vector column.

`E_COEFF`: energy dependence of long term CTI; three-element vector column.

`SHIFT`: shift parameter.

`TIMING_GAIN` The table is used/accessed by: CAL_pnGainTimingCorrect Table with parameters controlling fine-adjustment of the gain in TI mode

Attributes: None.

Columns:

`TIMING_GAIN`: scalar columns with four parameters (one per row)

`BURST_GAIN` The table is used/accessed by: CAL_pnGainBurstCorrect Table with parameters controlling fine-adjustment of the gain in BU mode

Attributes: None.

Columns:

`BURST_GAIN`: scalar columns with four parameters (one per row)

`EFF_GAIN` The table is used/accessed by: TBD. Table with the parameters of the gain tuning function which is used to adjust the EFF gain to that of FF.

Attributes:

`ALGOID`: algorithm identifier.

Columns:

`CCD_ID`: CCD number [1, 12].

`EFF_GAIN`: fine tuning of the Gain/CTI in EFF Mode; four-element vector column.

`QBOXTEMP_GAIN` The table is used/accessed by: TBD Table with parameters for the correction of the peak gain depending on the quadrant box temperatures.

Attributes:

`ADU_REF`: Mn K-alpha reference channel (ADU) for temperature dependent gain correction.

Columns:

`MODE_ID`: numeric ID of the mode.

`CCD_ID`: CCD number [1, 12].

`SLOPE`: temperature dependence of the gain correction.

`C_REF`: correction factor to adjust peak position.

`RATE_DEPENDENT_CTI` The table is used/accessed by: CAL_pnAdditionalGainCtiCorrect(). The parameters control the rate-dependent CTI correction in pn Fast Modes.

Attributes: None.

Columns:

`MODE_ID`**:** Numeric ID of the mode.

`A_i`**:** coefficients of the rate-dependent gain correction: $A_0 \times X^{A_1} + A_2$, where X is the number of shifted electrons per pixel per second.

4.3.10.2 Generation from Ground Data

Various calibration measurements have been performed at the PANTER test facility (Munich), and the Lure synchrotron at Orsay. Using synchrotron beams, standard x-ray tubes and fluorescence tubes charge transfer efficiency was measured at different energies and the behavior with energy was modeled using the physical parameters of the CCDs.

4.3.10.3 Generation from In-Orbit Data

In orbit CTI can be monitored through line centroiding of calclosed data and suitable celestial targets. Various observations including line rich SNR observations have been used to verify and improve the ground based calibration.

4.3.11 EMOS_DARKFRAME

4.3.11.1 Description

This CCF contains the fixed-offset vector tables for each CCD and each mode, as well as dark frame images of each of the 7 CCDs.

The CCF contains the following 12 extension tables:

`OFFSET_CCD1` : Accessed by :

- `CAL_getMOSoffsets()`: Routine to retrieve the row and column offsets for CCD 1 as given in the table.

Attributes: None.

Columns:

`MODE_ID` **:** numeric ID of the mode.

`COL_X0` **:** scalar value; column start in PIXCOORD0 coordinates for that mode

`ROW_Y0` **:** scalar value; row start in PIXCOORD0 coordinates for that mode

`ROW_LEN` **:** row length in pixels for that mode

`COL_LEN` **:** column length in pixel for that mode

ROW_OFFSET : vector column (type I) length 610 with ADC offset values: one per **column**.

COLUMN_OFFSET : vector column (type I) length 602 with ADC offset values: one per **row**.

OFFSET_CCD2...7 : same as for OFFSET_CCD1 for CCD2 to 7.(see above).

DARKFRAME_CCD1 : Accessed by :

- CAL_getMOSdarkFrameMap(): Retrieves full 600x600 dark frame array from CCF (in PIXCOOR1 coordinates)

Attributes: None.

Columns:

DARKFRAME_CCD2...7 : same as for DARKFRAME_CCD1 for CCD2 to 7.(see above).

4.3.11.2 Generation from ground data

Dark frames were derived from offset and median filtering of diagnostic files.

4.3.11.3 Generation from in-flight data

Fixed-offset tables are generated from background maps derived from E3/E4 energies of the internal calibration measurements. Median filterin of diagnostic files is also used to check the offset levels.

4.3.12 EPN_DARKFRAME

4.3.12.1 Description

Currently not used by the CAL. This CCF contains dark frame images of each of the 12 CCDs.

The CCF contains the following 12 tables:

DARKFRAME-1: Dark frame image of CCD1. Contains dummy values. Is currently not used by the CAL.

Attributes: None

DARKFRAME-2 **to** DARKFRAME-12: *Ditto* for CCDs 2–12.

4.3.13 EMOS_EFFICIENCY

4.3.13.1 Description

Not used. This CCF was meant to provide EMOS/RGA transmission efficiency data, and contains dummy data. But it is not used by the CAL and hence should be dropped.

The CCF contains one table:

`EFFICIENCY` : Not used.

Attributes: None.

Columns:

`FILTER_ID` **:** valid name of filter .
`EMOS_RGS_RATIO` **:** EMOS vs. RGS detection efficiency???

4.3.13.2 Generation from ground data

From offset and median filtering of diagnostic files

4.3.13.3 Generation from in-flight data

As ground generation

4.3.14 EPN_EFFICIENCY

4.3.14.1 Description

Not used. Contains dummy data.

4.3.15 EMOS_FILTERTRANSX

4.3.15.1 Description

There is one extension per filter, and one extension containing the common energy bins data. The filter extensions include a region expression, to cover cases where we know there are spatial variations, and the transmission values corresponding to each energy bin datum.

The CCF contains the following tables:

`FILTER-THIN1`: Thin1 filter transmission values. Accessed by :

- `CAL_getFilterTransmission()`: Retrieve filter transmission data as a function of energy for the THIN1 filter; the function exists in two flavors the second of which integrates the efficiency curve over a specified energy range

Attributes: None.

Columns:

`REGION`: The description of the region in which the transmission values are valid compliant with the selectlib spatial selection (in TELCOORD coordinates)

`TRANSMISSION`: vector column with transmission efficiency values in the range [0, 1]; index corresponds to energy axis bins in table EBINS.

`FILTER-THIN2`: Thin2 filter transmission values. Accessed by :

- `CAL_getFilterTransmission()`: Retrieve filter transmission data as a function of energy for the THIN1 filter; the function exists in two flavors the second of which integrates the efficiency curve over a specified energy range

Attributes: None.

Columns:

`REGION`: The description of the region in which the transmission values are valid.

`TRANSMISSION`: vector column with transmission efficiency values in the range [0, 1]; index corresponds to energy axis bins in table EBINS.

`FILTER-MEDIUM`: Medium filter transmission values. Accessed by :

- `CAL_getFilterTransmission()`: Retrieve filter transmission data as a function of energy for the THIN1 filter; the function exists in two flavors the second of which integrates the efficiency curve over a specified energy range

Attributes: None.

Columns:

`REGION`: The description of the region in which the transmission values are valid.

`TRANSMISSION`: vector column with transmission efficiency values in the range [0, 1]; index corresponds to energy axis bins in table EBINS.

`FILTER-THICK`: Thick filter transmission values. Accessed by :

- `CAL_getFilterTransmission()`: Retrieve filter transmission data as a function of energy for the THIN1 filter; the function exists in two flavors the second of which integrates the efficiency curve over a specified energy range

Attributes: None.

Columns:

`REGION`: The description of the region in which the transmission values are valid.

`TRANSMISSION`: vector column with transmission efficiency values in the range [0, 1]; index corresponds to energy axis bins in table EBINS.

`FILTER-OPEN`: Open filter transmission values. Accessed by :

- `CAL_getFilterTransmission()`: Retrieve filter transmission data as a function of energy for the THIN1 filter; the function exists in two flavors the second of which integrates the efficiency curve over a specified energy range

Attributes: None.

Columns:

`REGION`: The description of the region in which the transmission values are valid.

`TRANSMISSION`: vector column with transmission efficiency values in the range [0, 1]; index corresponds to energy axis bins in table EBINS.

`EBINS`: Defines the common energy bin structure. Accessed by :

- `CAL_getFilterTransmission()`

Attributes: None.

Columns:

`ENERGY`: vector column with energy values in eV defining energy axis on which transmission data are defined; length must match length of TRANSMISSION column in preceding tables.

4.3.15.2 Generation from Ground Calibration Data

Measurements at the synchrotron facilities, guided by fitting to analytical model of the filter material thicknesses

4.3.15.3 Generation from In-Orbit Data

Modifications based on known residuals at filter material absorption edges (namely carbon and aluminium)

4.3.16 EPN_FILTERTRANSX

4.3.16.1 Description

Describes the pn filter transmission *vs* energy. The CCF contains one extension of transmission values per filter, and one extension containing the common energy bins. The filter extensions include a region expression to cover cases where there are known spatial variations.

The CCF contains the following tables:

`FILTER-THIN1`**:** Thin1 filter transmission values.

Attributes: None.

Columns:

`REGION`**:** The description of the region in which the transmission values are valid.

`TRANSMISSION`**:** Variable length vector containing the transmission values corresponding to each energy bin datum.

`FILTER-THIN2`**:** Thin2 filter transmission values.

Attributes: None.

Columns:

`REGION`**:** The description of the region in which the transmission values are valid.

`TRANSMISSION`**:** Variable length vector containing the transmission values corresponding to each energy bin datum.

`FILTER-MEDIUM`**:** Medium filter transmission values.

Attributes: None.

Columns:

`REGION`**:** The description of the region in which the transmission values are valid.

`TRANSMISSION`**:** Variable length vector containing the transmission values corresponding to each energy bin datum.

`FILTER-THICK`**:** Thick filter transmission values.

Attributes: None.

Columns:

`REGION`**:** The description of the region in which the transmission values are valid.

`TRANSMISSION`**:** Variable length vector containing the transmission values corresponding to each energy bin datum.

`FILTER-OPEN`**:** Open filter transmission values.

Attributes: None.

Columns:

`REGION`: The description of the region in which the transmission values are valid.

`TRANSMISSION`: Variable length vector containing the transmission values corresponding to each energy bin datum.

`EBINS`: Defines the common energy bin structure.

Attributes: None.

Columns:

`ENERGY`: Variable length vector containing the energy bin values.

4.3.16.2 Generation from Ground Calibration Data

Measurements at synchrotron facilities (BESY, HASY, Orsay) guided by fitting to an analytical model based on filter composition and material thicknesses.

4.3.16.3 Generation from In-Orbit Data

Modifications of overall transmission values and those at the filter material absorption edges (C, Al, Sn).

4.3.17 EMOS_HKPARMINT

4.3.17.1 Description

Three extensions, with two listing the ranges for GTI definition, for each of several parameters that are believed to affect science quality. The third extension provides descriptive translation from alphanumeric housekeeping parameter to a description recognisable by advanced carbon-based life-form .

The CCF contains the following tables:

`HKPARMINT` : The table is used/accessed by:

- `CAL_getHKwindows()` : Table with HK parameter validity data; data outside the given validity intervals may indicate degraded science quality and ought to be screened in the pipeline reduction by means of GTI filtering.

Attributes: None.

Columns:

`PARM_NAME`: 5-character short ID of parameter as found in the ODF HK data (e.g. E1253)

`PARM_RANGE`: validity interval of the parameter in the form of a valid selectlib expression; '@' characters may be used instead of the parameter name; e.g. @ in [-125.0:-115.0].

`TRANSLATION` : The table is used/accessed by:

- `CAL_getHKwindows()` : Table to associate short (5 char) names with full, more meaningful ones; e.g. 'E1253' − > 'H_FPLANNORRANTEM'; the longer names may be used in SAS tasks directly (e.g. 'hkgtigen')

Attributes: None.

Columns:

`SHORT_NAME`: five-character short name of parameters

`LONG_NAME`: corresponding long name

`AUXPAR` : This table is not used by the CAL at the moment; it was meant to provide validity intervals for auxiliary data.

Attributes: None.

Columns:

`PARM_NAME`: parameter name (as in the AUX ODF tables)

`PARM_RANGE`: validity interval of the parameter in the form of a valid selectlib expression; '@' characters may be used instead of the parameter name; e.g. @ in [-125.0:-115.0]

`PARM_DESC`: description of parameter and associated information about validity expression

`PARM_UNIT`: units of parameter

4.3.17.2 Generation from In-Orbit Data

The list of relevant parameters is maintained with operational experience and their validity range updated according to the domain where scientific quality is believed to be unaffected when parameters are within this range. In its last versions the only parameters left for the MOSs are three focal plane temperatures.

4.3.18 EPN_HKPARMINT

4.3.18.1 Description

Three extensions, with two listing the ranges for GTI definition, for each of several parameters that are believed to affect science quality. The third extension provides descriptive translation from alphanumeric housekeeping parameter to a description recognisable by advanced carbon-based life-form .

The CCF contains the following tables:

`HKPARMINT` **:** The table is used/accessed by:

- `CAL_getHKwindows()` : Table with HK parameter validity data; data outside the given validity intervals may indicate degraded science quality and ought to be screened in the pipeline reduction by means of GTI filtering

Attributes: None.

Columns:

`PARM_NAME`**:** 5-character short ID of parameter as found in the ODF HK data (e.g. F1123)

`PARM_RANGE`**:** validity interval of the parameter in the form of a valid selectlib expression; '@' characters may be used instead of the parameter name; e.g. @ in [-91.0:-89.0]

`TRANSLATION` **:** The table is used/accessed by: CAL_getHKwindows() Table to associate short (5 char) names with full, more meaningful ones; e.g. 'F1129' $->$ 'H_CE_TTMPS'; the longer names may be used in SAS tasks directly (e.g. 'hkgtigen')

Attributes: None.

Columns:

`SHORT_NAME`**:** five-character short name of parameters

`LONG_NAME`**:** corresponding long name

`AUXPAR` **:** The table is not used by the CAL at the moment. It was meant to provide validity intervals for auxiliary data

Attributes: None.

Columns:

`PARM_NAME`**:** parameter name (as in the AUX ODF tables)

`PARM_RANGE`**:** validity interval of the parameter in the form of a valid selectlib expression; '@' characters may be used instead of the parameter name; e.g. @ in [-91.0:-89.0]

`PARM_DESC`**:** description of parameter and associated information about validity expression

`PARM_UNIT`**:** units of parameter

4.3.19 EMOS_LINCORD

4.3.19.1 Description

It contains columns for CCDID, nodeID and reference pixel location (nominally the middle of 300,300 in RAWX,RAWY). Their distance from the camera reference position in x,y and z are given in mm, while the rotation of each CCD is in radians. A second extension provides the description of the active field of view in the form of region expressions.

The CCF contains the following tables:

LINCORD : Table with data driving coordinate conversion routines; all geometry-related data about the camera system are in the CCF; this CCF is not MOS-specific. The table is used/accessed by:

- CAL_rawXY2mm()
- CAL_pixCoord0ToPixCoord1()
- CAL_pixCoord1ToChipCoord()
- CAL_chipCoordToPixCoord1()
- CAL_chipCoordToCamCoord1()
- CAL_camCoord1ToChipCoord()
- CAL_camCoord1ToCamCoord2()
- CAL_camCoord2ToCamCoord1()
- CAL_camCoord2ToSacCoord()
- CAL_sacCoordToCamCoord2()
- CAL_sacCoordToRowCoord()
- CAL_rowCoordToSacCoord()
- CAL_camCoord2ToTelCoord()
- CAL_telCoordToCamCoord2()

Attributes:

MM_PX_X: [mm] pixel size in X
MM_PX_Y: [mm] pixel size in Y
MM_PX_LY: [mm] Y-size of last pixel row
CC12_TX: [mm] CAMCOORD1− >2 translation in X
CC12_TY: [mm] CAMCOORD1− >2 translation in Y
CC12_TZ: [mm] CAMCOORD1− >2 translation in Z
R_CHIP_T: [K] reference chip temperature
R_CAM_T: [K] reference camera temperature
EXP_CHIP: [1/K] therm. expansion coeff. for chip (Si)
EXP_CAM: [1/K] therm. expansion coeff. for camera (Si)
CC2_TX: [mm] CAMCOORD2 origin in SACCOORD - X
CC2_TY: [mm] CAMCOORD2 origin in SACCOORD - Y
CC2_TZ: [mm] CAMCOORD2 origin in SACCOORD - Z
CC2_RPH: [deg] CAMCOORD2 orien. in SACCORD - PHI
CC2_RTH: [deg] CAMCOORD2 orien. in SACCCORD - THETA
CC2_RPS: [deg] CAMCOORD2 orien. in SACCCORD - PSI
ARC_MM_X: [arcsec/mm] plate scale in X
ARC_MM_Y: [arcsec/mm] plate scale in Y
MM_TX: [mm] origin of telescope frame in SACCOORD - X
MM_TY: [mm] origin of telescope frame in SACCOORD - Y
MM_TZ: [mm] origin of telescope frame in SACCOORD - Z

Columns:

CCD_ID: CCD number [1, 7]

NODE_ID: node id [0, 1]

X0: X-coordinate of center of PIXCOORD1 pixel (RAWX0/RAWY0) in CAMCOORD1 system

Y0: Y-coordinate of center of PIXCOORD1 pixel (RAWX0/RAWY0) in CAMCOORD1 system

Z0: Z-coordinate of center of PIXCOORD1 pixel (RAWX0/RAWY0) in CAMCOORD1 system

DX0: absolute uncertainty associated with X0 (currently unused)

DY0: absolute uncertainty associated with X0 (currently unused)

DZ0: absolute uncertainty associated with X0 (currently unused)

EULER_PHI: first angle (in 3-1-3 Euler rotation sequence) to align PIXCOORD1 with CAMCOORD1 frame

EULER_THETA: second angle (in 3-1-3 Euler rotation sequence) to align PIXCOORD1 with CAMCOORD1 frame

EULER_PSI: third angle (in 3-1-3 Euler rotation sequence) to align PIXCOORD1 with CAMCOORD1 frame

D_EULER_PHI: absolute uncertainty associated with EULER_PHI (currently unused)

D_EULER_THETA: absolute uncertainty associated with EULER_THETA (currently unused)

D_EULER_PSI: absolute uncertainty associated with EULER_PSI (currently unused)

RAWX0: PIXCOORD1 X-coordinate of reference pixel whose coordinates are given in (X0,Y0,Z0)

RAWY0: PIXCOORD1 Y-coordinate of reference pixel whose coordinates are given in (X0,Y0,Z0)

FOV : The table is used/accessed by: CAL_getFOVmap() Table with parameters defining the Field-Of-View

Attributes: None.

Columns:

CCD_ID: CCD number [1, 12]

REGION: valid selectlib expression defining the FOV in the CHIPCOORD system

4.3.19.2 Generation from Ground Data

Derived from assembly metrology data, also measurements of alignment device at Orsay synchrotron. (Note these sets were not in agreement, and the focal planes were reconstructed, so in-orbit data have been used)

4.3.19.3 Generation from In-Orbit Data

Using stellar clusters as the OMC2/3 field or NGC2516 and serendipitous field sources, source positions in RAWX,RAW Y are cross-sorrelated with catalogues to determine positions of the CCDs

4.3.20 EPN_LINCORD

4.3.20.1 Description

It contains columns for CCDID, nodeID and reference pixel location. Their distance from the camera reference position in x,y and z are given in mm, while the rotation of each CCD is in radians. A second extension provides the description of the active field of view in the form of region expressions.

The CCF contains the following tables:

`LINCORD` : Table with data driving coordinate conversion routines; all geometry-related data about the camera system are in the CCF; this CCF is not EPN-specific The table is used/accessed by:

- `CAL_rawXY2mm()`
- `CAL_pixCoord0ToPixCoord1()`
- `CAL_pixCoord1ToChipCoord()`
- `CAL_chipCoordToPixCoord1()`
- `CAL_chipCoordToCamCoord1()`
- `CAL_camCoord1ToChipCoord()`
- `CAL_camCoord1ToCamCoord2()`
- `CAL_camCoord2ToCamCoord1()`
- `CAL_camCoord2ToSacCoord()`
- `CAL_sacCoordToCamCoord2()`
- `CAL_sacCoordToRowCoord()`
- `CAL_rowCoordToSacCoord()`
- `CAL_camCoord2ToTelCoord()`
- `CAL_telCoordToCamCoord2()`

Attributes:

`MM_PX_X`: [mm] pixel size in X
`MM_PX_Y`: [mm] pixel size in Y
`MM_PX_LY`: [mm] Y-size of last pixel row
`CC12_TX`: [mm] CAMCOORD1− >2 translation in X

CC12_TY: [mm] CAMCOORD1− >2 translation in Y

CC12_TZ: [mm] CAMCOORD1− >2 translation in Z

R_CHIP_T: [K] reference chip temperature

R_CAM_T: [K] reference camera temperature

EXP_CHIP: [1/K] therm. expansion coeff. for chip (Si)

EXP_CAM: [1/K] therm. expansion coeff. for camera (Si)

CC2_TX: [mm] CAMCOORD2 origin in SACCOORD - X

CC2_TY: [mm] CAMCOORD2 origin in SACCOORD - Y

CC2_TZ: [mm] CAMCOORD2 origin in SACCOORD - Z

CC2_RPH: [deg] CAMCOORD2 orien. in SACCORD - PHI

CC2_RTH: [deg] CAMCOORD2 orien. in SACCCORD - THETA

CC2_RPS: [deg] CAMCOORD2 orien. in SACCCORD - PSI

ARC_MM_X: [arcsec/mm] plate scale in X

ARC_MM_Y: [arcsec/mm] plate scale in Y

MM_TX: [mm] origin of telescope frame in SACCOORD - X

MM_TY: [mm] origin of telescope frame in SACCOORD - Y

MM_TZ: [mm] origin of telescope frame in SACCOORD - Z

Columns:

CCD_ID: CCD number [1, 12]

NODE_ID: node id [0, 1]; always 0 for EPN

X0: X-coordinate of center of PIXCOORD1 pixel (RAWX0/RAWY0) in CAMCOORD1 system

Y0: Y-coordinate of center of PIXCOORD1 pixel (RAWX0/RAWY0) in CAMCOORD1 system

Z0: Z-coordinate of center of PIXCOORD1 pixel (RAWX0/RAWY0) in CAMCOORD1 system

DX0: absolute uncertainty associated with X0 (currently unused)

DY0: absolute uncertainty associated with X0 (currently unused)

DZ0: absolute uncertainty associated with X0 (currently unused)

EULER_PHI: first angle (in 3-1-3 Euler rotation sequence) to align PIXCOORD1 with CAMCOORD1 frame

EULER_THETA: second angle (in 3-1-3 Euler rotation sequence) to align PIXCOORD1 with CAMCOORD1 frame

EULER_PSI: third angle (in 3-1-3 Euler rotation sequence) to align PIXCOORD1 with CAMCOORD1 frame

D_EULER_PHI: absolute uncertainty associated with EULER_PHI (currently unused)

D_EULER_THETA: absolute uncertainty associated with EULER_THETA (currently unused)

D_EULER_PSI: absolute uncertainty associated with EULER_PSI (currently unused)

RAWX0: PIXCOORD1 X-coordinate of reference pixel whose coordinates are given in (X0,Y0,Z0)

RAWY0: PIXCOORD1 Y-coordinate of reference pixel whose coordinates are given in (X0,Y0,Z0)

FOV The table is used/accessed by: CAL_getFOVmap() Table with parameters defining the Field-Of-View

Attributes: None.

Columns:

CCD_ID: CCD number [1, 12]

REGION: valid selectlib expression defining the FOV in the CHIPCOORD system

4.3.20.2 Generation from Ground Data

Derived from assembly metrology data, also measurements of alignment device at Orsay synchrotron. (Note these sets were not in agreement, and the focal planes were reconstructed, so in-orbit data have been used)

4.3.20.3 Generation from In-Orbit Data

Using stellar clusters as the OMC2/3 field or NGC2516 and serendipitous field sources, source positions in RAWX,RAW Y are cross-sorrelated with catalogues to determine positions of the CCDs

4.3.21 EMOS_MODEPARAM

4.3.21.1 Description

One extension providing the list by instrument mode of important *fixed* parameters such as readout dead time, frame times, live times, window sizes, etc ... It allows to correct for mode dependent factors.

The CCF contains one table:

MODEPARAM: table with characteristic parameters of the various observing modes, accessed by :

- CAL_getModeParameters():

Attributes:

UNDER_X: Column under scan in pixels.

OVER_X: Column over scan in pixels.

UNDER_Y: Row under scan in pixels.

OVER_Y: Row over scan in pixels.

Columns:

MODE_ID: numeric Mode identifier.

INTEGRATION_T: Integration time (ms).

SHIFT_T: Shift time (ms).

FRAME_T: Frame time (ms).

CYCLE_T: Cycle time (ms).

TIME_RES: Time resolution (ms).

DUTY_CYCLE: Duty cycle.

SMEARED_FRAC: Smeared fraction.

PTSRCE_PILEUP: Pile up count rate limit for point sources.

LIMITING_FLUX: Maximum flux based on point source pile up considerations.

WINDOWX0: X-coord. (PIXCOORD0) of lower left corner of readout window.

WINDOWY0: Y-coord. (PIXCOORD0) of lower left corner of readout window.

WINDOWDX: Width of readout window in X direction (pixels).

WINDOWDY: Width of readout window in Y direction (pixels).

4.3.22 EPN_MODEPARAM

4.3.22.1 Description

Consists of one table containing the characteristic parameters of the various observing modes. It allows to correct for mode dependent factors such as effective live time, out of time fractions, etc... , accessed by :

- CAL_getModeParameters():

The CCF contains one table:

MODEPARAM:

Attributes:

UNDER_X: Column under scan in pixels.

OVER_X: Column over scan in pixels.

UNDER_Y: Row under scan in pixels.

OVER_Y: Row over scan in pixels.

Columns:

MODE_ID: Mode identifier.

INTEGRATION_T: Integration time (ms).

SHIFT_T: Shift time (ms).

FRAME_T: Frame time (ms).

CYCLE_T: Cycle time (ms).

TIME_RES: Time resolution (ms).

DUTY_CYCLE: Duty cycle.

SMEARED_FRAC: Smeared fraction.

PTSRCE_PILEUP: Pile up count rate limit for point sources.

LIMITING_FLUX: Maximum flux based on point source pile up considerations.

WINDOWX0: X-coord. (PIXCOORD0) of lower left corner of readout window.

WINDOWY0: Y-coord. (PIXCOORD0) of lower left corner of readout window.

WINDOWDX: Width of readout window in X direction (pixels).

WINDOWDY: Width of readout window in Y direction (pixels).

4.3.23 EMOS_PATTERNLIB

4.3.23.1 Description

Table with data defining the event pattern; each table corresponds to one observing mode.

The CCF contains the following tables:

PrimeFullWindow: accessed by :

- CAL_getEventPatterns():

Attributes: None

Columns:

PATTERN_ID: numerical ID of the pattern; range [0, 31].

PATTERN: 2-D vector columns with pattern data; 1 stands for pixel above threshold

PrimePartialW2...6: Same for indicated modes.

PrimePartialRFS: Same for RFS mode.

FastUncompressed: same for FastUncompressed (5 patters only).

FastCompressed: same for FastCompressed (5 patters only).

4.3.24 EPN_PATTERNLIB

4.3.24.1 Description

Contains one table per observing mode containing the data defining the event pattern.

The CCF contains the following tables:

`PrimeFullWindow`: accessed by :

- `CAL_getEventPatterns()`:

Attributes: None.

Columns:

`PATTERN_ID`: Numerical ID of the pattern.

`PATTERN`: 2-D vector columns with pattern data; 1 stands for pixel above threshold.

`PrimeFullWindowExtended`: *Ditto.*

`PrimeLargeWindow`: *Ditto.*

`PrimeSmallWindow`: *Ditto.*

`FastTiming`: *Ditto.*

`FastBurst`: *Ditto.*

4.3.25 EMOS_QUANTUMEFF

4.3.25.1 Description

Tables with energy-/position- and pattern-dependent quantum efficiencies; CCD number s encoded in the table name; pattern number is encoded in column name (QE_0, QE_1, etc.); position information is given in column REGION

Keywords to describe the contents of the CCF:

The CCF contains the following tables:

`QE_CCDn`: The quantum efficiency (QE) of the MOS detectors is given, per CCD, in the extensions QE_CCDn (where n=1 (central CCD) to 7). It is described as a function of event pattern type in columns QE_nn (where nn=0, single pixel events, - 31). The energy grid for these arrays in given in the extension EBINS. These extensions also hold keywords to describe the ratio of the events which have patterns 13–25 to those which have larger pattern numbers (available as MISCDATA). Keywords are also present which allow the average size of the large patterns to be calculated as a function of photon energy.

Attributes:

`LRG_1325`: Ratio of large events to Pattern 13-25 events

`LRG_SIZ0`: Coefficient 0 for calculation of average large pattern size

`LRG_SIZ1`: Coefficient 1

`LRG_SIZ2`: Coefficient 2

`LRG_SIZ3`: Coefficient 3

`LRG_SIZ4`: Coefficient 4

Columns:

`REGION`: A spatial region to allow intra-chip QE to be expressed.

`QE_nn`: The QE per pattern.

`EBINS`: The energy grid for the QE_CCDn arrays.

Attributes: None.

Columns:

`ENERGY`: Photon energy (eV).

`QE_TOTAL`: The total QE as a function of energy.

Attributes: None.

Columns:

`QE_TOTAL`: Total QE.

`FRACTION_CHANNEL` The fraction of the total QE which is contained in single, double, triple or quadruple pixel events, as a function of the channels tabulated in CHBINS_FRACTION.

Attributes: None.

Columns:

`MODE_ID`: The observing modes

`THRESH`: The threshold at which events are cut-off in channels.

`REGION`: A spatial selection

`FRAC_S`: The fraction contained in single pixels

`FRAC_D`: The fraction contained in double-pixel events

`FRAC_T`: The fraction contained in triples

`FRAC_Q`: The fraction contained in quadruple-pixel events

`FRAC_SD`: The fraction contained in single+double pixel events

`FRAC_SDTQ`: The fraction contained in single+double+triple+quads

`FRACTION_ENERGY` The fraction of the total QE which is contained in single, double, triple or quadruple pixel events, as a function of the energies tabulated in EBINS_FRACTION.

The effective area file generator, arfgen, calculates the detector QE by multiplying QE_TOTAL by FRACTION_ENERGY.

Attributes: None.

Columns:

`MODE_ID`: The observing modes

THRESH: The threshold at which events are cut-off in channels.

REGION: A spatial selection

FRAC_S: The fraction contained in single pixels in energy space

FRAC_D: The fraction contained in double-pixel events

FRAC_T: The fraction contained in triples

FRAC_Q: The fraction contained in quadruple-pixel events

FRAC_SD: The fraction contained in single+double pixel events

FRAC_SDTQ: The fraction contained in single+double+triple+quads

EBINS_FRACTION The energy grid for the FRACTION_ENERGY arrays.

Attributes: None.

Columns:

ENERGY: Photon energy (eV).

CHBINS_FRACTION The channel grid for the FRACTION_CHANNEL arrays.

Attributes: None.

Columns:

PI_CHAN: PI channel number (channels).

4.3.25.2 Generation from Ground Calibration Data

Derived from synchrotron measurements against reference detectors and interpolations against physical models.

4.3.25.3 Generation from In-Orbit Data

Spectral residuals guide the necessity to adapt the ground-based knowledge. This may be robust around absorption edges but generally it is difficult to disentangle gain and RMF effects.

4.3.26 EPN_QUANTUMEFF

4.3.26.1 Description

The PN quantum efficiency file has the same format as the MOS files. The only difference is in the CCD extensions where there are only two pattern dependent columns: QE_0 giving the QE in single-pixel events as a function of energy and QE_1 which gives the QE in double-pixel events. The latter is the sum of the QE in patterns 1–4.

4.3.26.2 Generation from Ground Data

See EMOS_QUANTUMEF.

4.3.26.3 Generation from In-Orbit Data

See EMOS_QUANTUMEF.

4.3.27 EMOS_REDIST

4.3.27.1 Description

The redistribution parameters for the MOS cameras.

The CCF contains the following tables:

`CCD_REDISTRIBUTION-n` The redistribution parameters for the MOS detectors are given here per CCD. NB: The response of the MOS detectors has evolved as the satellite has aged. A separate REDIST file is available for each epoch where the response is seen to be different.

Attributes: None.

Columns:

- `MODE_ID`: The observing mode.
- `REGION`: The central CCD of each MOS detector has developed a patch where the energy resolution is worse than the average. This column describes each region.
- `THRESHOLD` : The event cut-off threshold (eV).
- `GRADE`: The pattern type [1=singles (pattern 0 only); 2=singles,doubles,triples and quads (patterns 0-12)].
- `PARAM`: The redistribution parameters.

`PHA_EBOUNDS` The energy bounds of PHA channels.

Attributes: None.

Columns:

- `CHANNEL`: The PHA channel number.
- `E_MIN`: Minimum energy (eV).
- `E_MAX`: Maximum energy (eV).

`BINNEDPI_EBOUNDS` The energy bounds of the PI channels, used to determine the equivalent energies of spectral channels in the redistribution matrix construction.

Attributes: None.

Columns:

`CHANNEL`**:** The PI channel number.

`E_MIN`**:** Minimum energy (eV).

`E_MAX`**:** Maximum energy (eV).

`EBINS` The energy bins used when contructing the redistribution matrix.

Attributes: None.

Columns:

`CHANNEL`**:** The channel number.

`E_MIN`**:** Minimum energy (eV).

`E_MAX`**:** Maximum energy (eV).

4.3.27.2 Generation from Ground Calibration Data

The initial energy response of the MOS CCDs was measured on ground.

4.3.27.3 Generation from In-Orbit Data

Many calibration observations of celestial targets have been used to measure the response function of the MOS detectors as a function of observing date. The response has evolved, particularly at low-energies, and is currently modelled by an empirical formula designed to give consistent results between the EPIC-pn, EPIC-MOS and RGS cameras.

4.3.28 EPN_REDIST

4.3.28.1 Description

The file contains energy bound information and absorption coefficients used in the analytical model that constructs the response matrix.

The CCF contains the following tables:

`EBOUNDS` The energy bounds of PI channels (not used ?).

Attributes: None.

Columns:

`CHANNEL`**:** The PI channel number.

`E_MIN`**:** Minimum energy (eV).

`E_MAX`**:** Maximum energy (eV).

PHA_EBOUNDS The energy bounds of PHA channels (not used ?).

Attributes: None.

Columns:

CHANNEL: The PHA channel number.
E_MIN: Minimum energy (eV).
E_MAX: Maximum energy (eV).

BINNEDPI_EBOUNDS The energy bounds of the PI channels, used to determine the equivalent energies of spectral channels in the redistribution matrix construction.

Attributes: None.

Columns:

CHANNEL: The PI channel number.
E_MIN: Minimum energy (eV).
E_MAX: Maximum energy (eV).

EBINS The energy bins used when contructing the redistribution matrix.

Attributes: None.

Columns:

CHANNEL: The channel number.
E_MIN: Minimum energy (eV).
E_MAX: Maximum energy (eV).

ABSORPTION **Attributes:** None.

Columns:

COMPOUND: The name of the material.
ENERGY: Photon energy (eV).
INV_LENGTH: Absorption coefficient (1/cm).

NOISE_PARAMS **Attributes:** None.

Columns:

MODE_ID: The observing mode.
PATTERN: Event patterns (1/2).
NOISE_PARAMETER: Noise parameters.

PARTEVENT_PARAMS **Attributes:** None.

Columns:

TAU_IN: Tau noise parameter for each energy.
S_IN: S noise parameter for each energy.
T_IN: T noise parameter for each energy.

THRESH_LOSS **Attributes:** None.

Columns:

MODE_ID: The observing mode.
PATTERN: Event patterns (1/2).
THRESH_LOSS: Threshold loss for each energy.

4.3.28.2 Generation from Ground Data

4.3.28.3 Generation from In-Orbit Data

4.3.29 EPN_REJECT

4.3.29.1 Description

The file contains median maps, noise maps and an offset correction table used in low energy noise reduction of the PN.

The CCF contains the following tables:

`NOISE_MAP_INDEX` An index which gives the name of the extension containing the correct noise map for a given mode, ccd and rawx value.

Attributes: None.

Columns:

- `MODE`: The observing mode
- `CCD`: The CCD number
- `RAWX`: X-range specification
- `POINTER`: Name of the noise map extension
- `EXPOSURE`: Exposure time of the map (secs)

`MEDIAN_MAP_INDEX` An index to the median map extensions.

Attributes: None.

Columns:

- `MODE`: The observing mode
- `CCD`: The CCD number
- `POINTER`: Name of the median map extension

`MASTER_OFFSET_TABLE_INDEX` Index to master offset maps.

Attributes: None.

Columns:

- `MODE`: The observing mode
- `CCD`: The CCD number
- `POINTER`: Pointer to master offset maps

`CORRECTION_VALUES` The correction values for each energy.

Attributes: None.

Columns:

MODE: The observing mode
CCD: The CCD number
H_LOW: Minimum energy (keV)
H_HIGH: Maximum energy (keV)
COR_VAL: Correction value

NOISE_MAP_mm_nn The noise map for ccd nn and mode mm, consisting of a single image.

Attributes: None.

MEDIAN_MAP_mm_nn The median map for ccd nn and mode mm, consisting of a single image.

Attributes: None.

NOISE_MAP_DUMMY Dummy noise map, consisting of a single image.

Attributes: None.

MEDIAN_MAP_DUMMY Dummy median map, consisting of a single image.

Attributes: None.

MASTER_MAP_mm_nn Master offset map for ccd nn and mode mm, consisting of a single image.

Attributes: None.

MASTER_MAP_DUMMY Dummy master offset map, consisting of a single image.

Attributes: None.

4.3.29.2 Generation from Ground Data

4.3.29.3 Generation from In-Orbit Data

The **reference map** for computing the offset shifts was derived from exposures of the Lockman hole (high statistics due to long exposure times, only faint X–ray sources in the FOV), obtained in revolutions 344, 349, 522, 523, 525, 526, 527, 528, 544, 548.

4.3.30 EMOS_TIMECORR

4.3.30.1 Description

This CCF provides frame times per CCD per mode.

The CCF contains the following tables:

FRAMETIME: Table with nominal CCD frame times [ms] as a function of mode, accessed by :

- `CAL_getFrameTime()`:

Attributes: None.

Columns:

`MODE_ID` **:** numerical ID of observing mode.

`FRAME_TIME`**:** frame integration time [ms]; this is a vector columns of length 7 - *should be converted to scalar column (retain only first element) in subsequent CCF issues; has no impact on CAL.*

4.3.31 EPN_TIMECORR

4.3.31.1 Description

This CCF provides frame times and time correlations per mode.

The CCF contains the following tables:

`FRAMETIME`**:** Table is not used and can be dropped in future versions of this CCF (unless used by QLA).

Attributes: None.

Columns:

`MODE_ID`**:** Numerical ID of observing mode (see 4.3.32).

`FRAME_TIME`**:** Frame integration time [ms].

`TIMECORR`**:** Used/accessed by CAL_correctTime(). Table with data controlling fine-time adjustments to event times as described in "Time resolution capabilities of the XMM EPIC pn-CCD in different readout modes", M. Kuster et al., Eq. (1)-(6) + (7) + (10).

Attributes: None.

Columns:

`MODE_ID`**:** Numerical ID of observing mode (see 4.3.32).

`FRAME_TIME`**:** Vector column of length 3 with parameters of fine-time adjustments.

Modes of EMOS1 & 2	MODE_ID
PrimeFullWindow	0
PrimePartialRFS	1
PrimePartialW2	2
PrimePartialW3	3
PrimePartialW4	4
PrimePartialW5	5
PrimePartialW6	6
FastUncompressed	7
FastCompressed	8
OffsetVariance	9
CcdDiagnostic	10
ExtraHeatingAnnealing	11
ExtraHeatingDeicing	12
ExtraHeatingDecontam	13
InFligtTest	14
Default mode is: PrimeFullWindow	

Science data modes of EMOS1 & 2	ID
Imaging	0
Timing	1
ReducedImaging	2
CompressedTiming	3

4.3.31.2 Generation from Ground Data

4.3.31.3 Generation from In-Orbit Data

4.3.32 Appendix: List of valid Mode/Filter IDs

4.4 RGS CCF Components

While some relevant RGS calibration data are given in the general-purpose CCF components

- `XMM_ABSCOEFF`
- `XMM_BORESIGHT`
- `XMM_MISCDATA`
- `XMM_SPECQUAL`
- `XRT[12]_XPSF`

Modes of EPN	MODE_ID
PrimeFullWindow	0
PrimeFullWindowExtended	1
PrimeLargeWindow	2
PrimeSmallWindow	3
FastTiming	4
FastBurst	5
PrimeFullOffset	6
PrimeLargeOffset	7
Offset	8
Noise	9
Diagnostic	10
ExtraHeatingDeicing	11
ExtraHeatingDecontam	12
InFligtTest	13
Default mode is: PrimeFullWindow	

Science data modes of EPN	ID
Imaging	0
Timing	1
Burst	2

most are stored in files named `RGS[12]_dddddd_vvvv.CCF`, where `dddddd` shows the type of data and `vvvv` is the 4-digit version number. There is one CCF file each of identical structure for RGS1 and RGS2 as shown by the filename and the corresponding `INSTRUMENT` header keyword. Some information concerning the maintenance of RGS CCF files is given below, though more details are available in the release notes that accompany new file releases and other technical notes. Maintenance work is often done with a combination of SAS and IDL tools, many of which were written by Cor de Vries at SRON Utrecht. Efforts are underway to centralise these procedures at ESAC. The purpose of CCF files is for use during data reduction by the SAS in the production of RGS event lists from which the spectra of sources are extracted, accompanied by response files. The RGS SAS tasks that use individual CCFs are shown in the table below.

Filters of EMOS1/2 & PN	FILTER_ID
Open	0
Closed	1
Thin1	2
Thin2	3
Medium	4
Thick	5
CalOpen	6
CalClosed	7
CalThin1	8
CalThin2	9
CalMedium	10
CalThick	11
Default filter is: Open	

Ground station name	ID	longitude	lattitude	elevation
Villafranca-1	5	356.048deg	40.4426deg	656.455m
Villafranca-2	13	356.047deg	40.4456deg	664.432m
Villafranca-3	14	356.048deg	40.4433deg	650.833m
Perth	22	115.885deg	-31.8025deg	22.16m
Kourou	21	307.195deg	5.25144deg	-14.558m
Santiago	41	289.334deg	-33.151deg	729m

Use of CCF components by RGS SAS tasks

	hkgtigen	rgssources	rgsframes	rgsenergy	rgsbadpix	rgsevents	rgsangles	rgsfilter	rgsregions	rgsspectrum	rgsrmfgen
XMM_ABSCOEFF									Y		Y
XMM_BORESIGHT		Y									
XMM_MISCDATA		Y								Y	
XRT[12]_XPSF											Y
RGS[12]_ADUCONV				Y	Y	Y					Y
RGS[12]_BADPIX			Y		Y						
RGS[12]_CLOCKPATTERNS											
RGS[12]_COOLPIX											
RGS[12]_CROSSPSF									Y		Y
RGS[12]_CTI				Y							
RGS[12]_DARKFRAME				Y	Y	Y					
RGS[12]_EFFAREACORR											Y
RGS[12]_EXAFS											Y
RGS[12]_HKPARMINT	Y										
RGS[12]_LINCOORD		Y		Y		Y	Y	Y	Y	Y	Y
RGS[12]_LINESPREADFUNC											Y
RGS[12]_MODEPARAM			Y								
RGS[12]_QUANTUMEF											Y
RGS[12]_REDIST									Y		Y
RGS[12]_TEMPLATEBCKGND											

4.4.1 XMM_MISCDATA

Purpose

Provide general purpose data for RGS and other XMM instruments.

Functionality

Table of data for named parameters.

Ground calibration

Important RGS parameters concerning instrument geometry were measured to high accuracy before launch.

Maintenance procedures

Only rare changes are to be expected on a case-by-case basis.

References

`http://xmm.esac.esa.int/docs/documents/CAL-SRN-0134-1-0.ps.gz`
`http://xmm.esac.esa.int/docs/documents/CAL-SRN-0065-1-0.ps.gz`

CCF Structure

XMM_MISCDATA_vvvv.CCF

	parameter	value	units
RGS1	CCD_XPIXELS	1024	pixel
RGS1	CCD_YPIXELS	384	pixel
RGS1	ROWLAND	3352.64	mm
RGS1	GRAT_LINE_DENS	15489.455	
RGS1	GRAT_LINE_SLOPE	4.6197	Angstrom/mm
RGS1	GRAT_LINE_OFFS	15491	Angstrom
RGS1	GRAT_XSIZE	200.0	
RGS1	GRAT_Y1SIZE	100.406	mm
RGS1	GRAT_Y2SIZE	97.206	mm
RGS1	INCIDENCE_ANGLE	1.576191	
RGS1	BLAZE_ANGLE	0.6988771	deg
RGS1	REFERENCE_TEMP	193.15	K
RGS1	DELTA_TCCD	10.0	K
RGS1	DELTA_TFPA	20.0	K
RGS1	REJECT_THR_FACT	4.0	
RGS1	ACCEPT_THR_FACT	5.0	
RGS1	DPPVERS	0	
RGS1	AREA_EFF	50.0	cm^2
RGS1	ARCSEC_PER_MM_X	27.0	arcsec/mm
RGS1	ARCSEC_PER_MM_Y	27.0	arcsec/mm
RGS1	CCD_WIDTH	27.648	mm
RGS1	CCD_CENTER	0.	mm
RGS1	MM_PER_PIXEL_X	0.027	mm
RGS1	MM_PER_PIXEL_Y	0.027	mm
RGS1	OPTICS_X	0.	pixel
RGS1	OPTICS_Y	0.	pixel
RGS1	PIXEL_SIZE	0.081	mm

XMM_MISCDATA_vvvv.CCF

	parameter	value	units
RGS2	CCD_XPIXELS	1024	pixel
RGS2	CCD_YPIXELS	384	pixel
RGS2	ROWLAND	3355.645	mm
RGS2	GRAT_LINE_DENS	15489.455	
RGS2	GRAT_LINE_SLOPE	4.6197	Angstrom/mm
RGS2	GRAT_LINE_OFFS	15491	Angstrom
RGS2	GRAT_XSIZE	200.0	mm
RGS2	GRAT_Y1SIZE	100.406	mm
RGS2	GRAT_Y2SIZE	97.206	mm
RGS2	INCIDENCE_ANGLE	1.576191	deg
RGS2	BLAZE_ANGLE	0.6988771	deg
RGS2	REFERENCE_TEMP	193.15	K
RGS2	DELTA_TCCD	10.0	K
RGS2	DELTA_TFPA	20.0	K
RGS2	REJECT_THR_FACT	4.0	
RGS2	ACCEPT_THR_FACT	5.0	
RGS2	DPPVERS	0	
RGS2	AREA_EFF	50.0	cm^2
RGS2	ARCSEC_PER_MM_X	27.0	
RGS2	ARCSEC_PER_MM_Y	27.0	
RGS2	CCD_WIDTH	27.648	mm
RGS2	CCD_CENTER	0.	mm
RGS2	MM_PER_PIXEL_X	0.027	mm
RGS2	MM_PER_PIXEL_Y	0.027	mm
RGS2	OPTICS_X	0.	pixel
RGS2	OPTICS_Y	0.	pixel
RGS2	PIXEL_SIZE	0.081	mm
XMM	SI_THERMAL_EXP	2.0E-06	mm/K

4.4.2 XMM_BORESIGHT

Purpose

Define XMM-Newton focal plane geometry.

Functionality

Table of Euler angles with one row for each of the XMM-Newton instruments.

Ground calibration

From alignment data measured during instrument integration.

Maintenance procedures

Soon after launch, measurements of narrow lines in coronal sources were used to correct the Euler angles and it is to be expected that this exercise will be repeated.

References

`http://xmm.esac.esa.int/docs/documents/CAL-SRN-0017-1-0.ps.gz`

CCF Structure

XMM_BORESIGHT_vvvv.CCF

BORESIGHT Binary Table Columns

```
INSTRUMENT_ID
EULER_PHI (rad)
EULER_THETA (rad)
EULER_PSI (rad)
D_EULER_PHI (rad)
D_EULER_THETA (rad)
D_EULER_PSI (rad)
```

4.4.3 XRT_XPSF

Purpose

Describe the XRT PSF projected onto the RGS dispersion axis input onto the RGS gratings.

Functionality

This CCF contains many telescope PSF extensions, only one of which is directly relevant to the RGS. The RGS projected XRT PSF is modelled through a combination of Gaussian and Lorenzian functions.

Ground calibration

None. Initial values were calculated by integrating the analytical functions used at launch to approximate the telescope PSFs.

Maintenance procedures

RGS1 and MOS1 share XRT1. RGS2 and MOS2 share XRT2. Any revision of the MOS PSFs stored elsewhere in the CCFs should be used to calculate the corresponding new RGS projection.

References

http://xmm.esac.esa.int/docs/documents/CAL-SRN-0063-1-1.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0043-1-0.ps.gz

CCF Structure

XRT[12]_XPSF_vvvv.CCF

BETAPSF Binary Table Columns

```
TYPE
NORM
CENTER (rad)
WIDTH (rad)
```

4.4.4 RGS_ADUCONV

Purpose

Calculate PI event energy from its PHA value .

Functionality

OFFSET and GAIN are needed to calculating the energy of each detected RGS event. Early in the mission, a single OFFSET value was used in the SAS for each CCD node before the introduction of dynamic values calculated for individual pixels from diagnostic data averaged over three XMM revolutions and supplied with the ODF. The structure of one of these files, whose names follow the scheme ${rev}_${ObsID}_R[12]X00000OFX.FIT, is shown below.

The static values stored in this CCF are kept in reserve. The table OFFSET_GAIN contains individual values of gain and offset for each CCD readout node. Although it has not yet proved necessary, OFFSET values are formally parameterized as a linear function of observation time with the relevant parameters stored in the columns OFFSET_CONST and OFFSET_SLOPE to be used in conjunction with the value of the keyword REFTIME.

The columns REJTHRESH and ACCTHRESH hold the thresholds in units of PI that are applied during event reconstruction in rgsevents. These thresholds are scaled for event GRADE > 1 by using the values of the keywords QSPLITR2/3/4, which are empirically determined charge-splitting ratios for grades between two and four.

The table ADUCOEFF is for QLA use only where no CTI corrections are performed. The CCF data are compensated for this. The formula used is

$$E = (PHA - OFFSET) \times GAIN$$

Ground calibration

Initial values were established from the data collected during the calibration of the CCDs at SRON and the calibration of the RGS at PANTER.

Maintenance procedures

Every two years, GAIN and CTI are calibrated jointly using strong lines of known wavelength in Capella.

References

http://xmm.esac.esa.int/docs/documents/CAL-SRN-0240-1-1.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0237-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0214-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0140-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0132-1-1.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0131-1-1.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0068-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0044-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0011-1-0.ps.gz

CCF Structure

RGS[12]_ADUCONV_vvvv.CCF

Primary Header CCF Keywords

```
KEV_P_PI= 1.00000000000000E+00 / [keV/chan] slope for linear PI->keV conversion
KEV0    = 1.00000000000000E+00 / [keV] offset for linear PI->keV conversion
```

ADUCOEFF Binary Table Columns

```
CCD_ID
NODE_ID
OFFSET (chan)
GAIN (eV/chan)
```

OFFSET_GAIN Binary Header CCF Keywords

```
RAND_MIN= 0.00000000000000E+00 / lower limit for PHA randomization
RAND_MAX= 1.00000000000000E+00 / upper limit for PHA randomization
QSPLITR2= 1.00000000000000E+00 / charge split ratio for 2-pixel events
QSPLITR3= 1.00000000000000E+00 / charge split ratio for 3-pixel events
QSPLITR4= 1.00000000000000E+00 / charge split ratio for 4-pixel events
ALGOID  =                    1 / CAL algorithm identifier
REFTIME = '2001-11-05T20:00:00' / reference for parameterization of offset
```

OFFSET_GAIN Binary Table Columns

```
CCD_ID
OFFSET_CONST (chan)
OFFSET_SLOPE (chan/s)
GAIN (chan/eV)
GAIN_ERROR (chan/eV)
REJTHRESH (chan)
ACCTHRESH (chan)
```

ODF Structure

${rev}_${ObsID}_R[12]X00000OFX.FIT

OFX Image extensions

CCD1_offset_c
CCD1_offset_d
CCD2_offset_c
CCD2_offset_d
CCD3_offset_c
CCD3_offset_d
CCD4_offset_c
CCD4_offset_d
CCD5_offset_c
CCD5_offset_d
CCD6_offset_c
CCD6_offset_d
CCD7_offset_c
CCD7_offset_d
CCD8_offset_c
CCD8_offset_d
CCD9_offset_c
CCD9_offset_d

4.4.5 RGS_BADPIX

Purpose

Identify CCD pixel and column defects.

Functionality

Bad parts of the CCD surfaces are identified by their starting coordinates `RAWX,RAWY` and length `YLENGTH`$\geq$ 1 and fall into different categories `DESCRIPTION` in lower case for uplinked quantities or upper case in the CCF only

`d|D` A dead pixel gives no data and kills other pixels transfered through it.

`h|H` A hot pixel delivers a high, nearly constant, signal independent of X-ray illumination almost every readout.

`f|F` A flickering pixel depends on both the CCD settings and the throughput of charge.

`p|P` A pin-hole defect in the aluminium light shield.

Ground calibration

An initial set of CCD defects was identified before launch.

Maintenance procedures

References

http://xmm.esac.esa.int/docs/documents/CAL-SRN-0239-1-1.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0226-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0130-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0094-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0087-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0083-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0075-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0011-1-0.ps.gz

CCF Structure

RGS[12]_BADPIX_vvvv.CCF

BADPIX Binary Header CCF Keywords

```
TEMPERAT= 1.63150000000000E+02 / [K] Temperature of CCDs
OCB     =                    3 / On-chip binning factor
```

BADPIX Binary Table Columns

```
CCD_ID
NODE_ID
RAWX (pixel)
RAWY (pixel)
YLENGTH (pixel)
DESCRIPTION
```

BADPIX1 Binary Header CCF Keywords

```
TEMPERAT= 1.63150000000000E+02 / [K] Temperature of CCDs
OCB     =                    1 / On-chip binning factor
```

BADPIX1 Binary Table Columns

```
CCD_ID
NODE_ID
RAWX (pixel)
RAWY (pixel)
YLENGTH (pixel)
DESCRIPTION
```

4.4.6 RGS_CLOCKPATTERNS

Purpose

Parameters of clock patterns used in reading out CCDs.

Functionality

Binary table of relevant data.

Ground calibration

Not applicable.

Maintenance procedures

New clock sequences are provided when required by specialist instrument engineers.

References

`http://xmm.esac.esa.int/docs/documents/CAL-SRN-0015-1-0.ps.gz`

CCF Structure

RGS[12]_CLOCKPATTERNS_vvvv.CCF

ClockPatterns Binary Table Columns

```
CSG_ID
CSG_NAME
MODE_ID
OCB
NODES
SLOSHING
PRESCAN (pixel)
OVERSCAN (pixel)
FRAME_TRANSF_TIME (ms)
TOTAL_READOUT (ms)
SHIFT_AND_READ (ms)
WINDOWX0 (pixel)
WINDOWX (pixel)
WINDOWY0 (pixel)
WINDOWY (pixel)
DESCRIPTION
```

4.4.7 RGS_COOLPIX

Purpose

Identification of the class of so-called cool pixels.

Functionality

Stacking many spectra of smooth continuum sources over the course of the mission has revealed that the average response of some columns is significantly lower than those of immediate neighbours, leading to their designation as cool. These small defects are only detectable in spectra of high statistical weight but could be mistaken for weak absorption lines of cosmic origin. The SAS can be instructed to discard these cool pixels in the same way as it does more obviously bad pixels.

Ground calibration

None.

Maintenance procedures

As the behaviour of bad surface can change, the identification of cool pixels should periodically be reexamined.

References

`http://xmm.esac.esa.int/docs/documents/CAL-SRN-0218-1-0.ps.gz`

CCF Structure

RGS[12]_COOLPIX_vvvv.CCF

COOLPIX Binary Header CCF Keywords

```
OCB     =                    3 / On-chip binning factor
```

COOLPIX Binary Table Columns

```
CCD_ID
NODE_ID
RAWX (pixel)
RAWY (pixel)
YLENGTH (pixel)
DESCRIPTION
```

4.4.8 RGS_CROSSPSF

Purpose

Describe the width of the point-source event distribution in the cross-dispersion direction as a function of dispersion angle to enable extraction of spectra.

Functionality

The distribution is decomposed into a combination of Gaussian and Lorentzian distributions.

Ground calibration

Initial values from PANTER data and comparisons with ray-trace models.

Maintenance procedures

In-flight observations of point-sources like Mkn421 that illuminate the whole detector provide measurements of high statistical weight to enable accurate models to be derived at all dispersion angles.

References

`http://xmm.esac.esa.int/docs/documents/CAL-SRN-0142-1-0.ps.gz`
`http://xmm.esac.esa.int/docs/documents/CAL-SRN-0114-1-0.ps.gz`
`http://xmm.esac.esa.int/docs/documents/CAL-SRN-0045-1-0.ps.gz`
`http://xmm.esac.esa.int/docs/documents/CAL-SRN-0014-1-0.ps.gz`

CCF Structure

RGS[12]_CROSSPSF_vvvv.CCF

CROSSPSF Binary Table Columns

```
BETA (rad)
GAUSS_NORM
GAUSS_CENTER (rad)
GAUSS_SIGMA (rad)
LORENTZ_NORM
LORENTZ_CENTER (rad)
LORENTZ_WIDTH (rad)
```

4.4.9 RGS_CTI

Purpose

Corrections are required to compensate for the loss of charge that occurs when events are transferred through the CCD to the readout.

Functionality

Calibration data are required for both serial, X, and parallel, Y, transfers.

Ground calibration

Intial values were provided at launch from laboratory measurements.

Maintenance procedures

The detector CTI characteristics are subject to change and require assessment on a regular basis in flight. The combination of bright emission lines in Capella and the smooth continuum of Mkn421 provide the means for this, usually in conjunction with gain measurements. Once every two years, Mkn421 is observed offset by ±2 armin in the cross-dispersion direction near the edges of the detectors at ±2.5 armin for purposes of monitoring parallel transfers.

References

http://xmm.esac.esa.int/docs/documents/CAL-SRN-0240-1-1.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0237-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0214-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0140-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0132-1-1.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0131-1-1.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0068-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0044-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0011-1-0.ps.gz

CCF Structure

RGS[12]_CTI_vvvv.CCF

CTI Binary Table Columns

```
CCD_ID
CTI_X
CTI_Y
```

CTI_EXTENDED Binary Header CCF Keywords

```
ALGOID  =                    1 / argorithm identifyer
```

CTI_EXTENDED Binary Table Columns

```
CCD_ID
MODE_ID
NODE_ID
CTI_X
CTI_Y
```

XCTI Binary Header CCF Keywords

```
ALGOID  =                    1 / argorithm identifyer
SCALE   = 1.00000000000000E+00 / muliplicative scale factor of CTI values
```

XCTI Binary Table Columns

```
CCD_ID
NODE_ID
CTI_X
CTI_X_ERR
```

CTIY1 Binary Header CCF Keywords

```
ALGOID  =                    1 / argorithm identifyer
SCALE   = 1.00000000000000E+00 / muliplicative scale factor of CTI values
```

CTIY1 Binary Table Columns

YCOL
CTI_Y
CTI_Y_ERROR

CTIY2 Binary Header CCF Keywords

```
ALGOID  =                    1 / argorithm identifyer
SCALE   = 1.00000000000000E+00 / muliplicative scale factor of CTI values
```

CTIY2 Binary Table Columns

YCOL
CTI_Y
CTI_Y_ERROR

CTIY3 Binary Header CCF Keywords

```
ALGOID  =                    1 / argorithm identifyer
SCALE   = 1.00000000000000E+00 / muliplicative scale factor of CTI values
```

CTIY3 Binary Table Columns

YCOL
CTI_Y
CTI_Y_ERROR

CTIY4 Binary Header CCF Keywords

```
ALGOID  =                    1 / argorithm identifyer
SCALE   = 1.00000000000000E+00 / muliplicative scale factor of CTI values
```

CTIY4 Binary Table Columns

```
YCOL
CTI_Y
CTI_Y_ERROR
```

CTIY5 Binary Header CCF Keywords

```
ALGOID  =                    1 / argorithm identifyer
SCALE   = 1.00000000000000E+00 / muliplicative scale factor of CTI values
```

CTIY5 Binary Table Columns

```
YCOL
CTI_Y
CTI_Y_ERROR
```

CTIY6 Binary Header CCF Keywords

```
ALGOID  =                    1 / argorithm identifyer
SCALE   = 1.00000000000000E+00 / muliplicative scale factor of CTI values
```

CTIY6 Binary Table Columns

```
YCOL
CTI_Y
CTI_Y_ERROR
```

CTIY7 Binary Header CCF Keywords

```
ALGOID  =                    1 / argorithm identifyer
SCALE   = 1.00000000000000E+00 / muliplicative scale factor of CTI values
```

CTIY7 Binary Table Columns

```
YCOL
CTI_Y
CTI_Y_ERROR
```

CTIY8 Binary Header CCF Keywords

```
ALGOID  =                    1 / argorithm identifyer
SCALE   = 1.00000000000000E+00 / muliplicative scale factor of CTI values
```

CTIY8 Binary Table Columns

```
YCOL
CTI_Y
CTI_Y_ERROR
```

CTIY9 Binary Header CCF Keywords

```
ALGOID  =                    1 / argorithm identifyer
SCALE   = 1.00000000000000E+00 / muliplicative scale factor of CTI values
```

CTIY9 Binary Table Columns

```
YCOL
CTI_Y
CTI_Y_ERROR
```

4.4.10 RGS_DARKFRAME

Purpose

Parameterisation of the dark signal per pixel for use in conjunction with offset values in correcting observed pulse heights.

Functionality

Used in conjunction with offset values to correct the detected charge of events..

Ground calibration

An initial set of parameters was derived from diagnostic data during ground measurements.

Maintenance procedures

Diagnostic data accumulated in flight could be used to check the validity of darkframe data although the relevance of such a procedure should be considered now that dynamical offets values are calculated from diagnostic data. In any case, changes are expected to be very small.

References

`http://xmm.esac.esa.int/docs/documents/CAL-SRN-0173-1-1.ps.gz`
`http://xmm.esac.esa.int/docs/documents/CAL-SRN-0033-1-0.ps.gz`

CCF Structure

RGS[12]_DARKFRAME_vvvv.CCF

DARKFRAME Binary Header CCF Keywords

```
TEMP    = 1.93400000000000E+02 / [K] Temperature of focal plane
```

DARKFRAME Binary Table Columns

```
CCD_ID
NODE_ID
PHA_POS (chan)
PHA_SIG (chan)
PHA_N (chan)
PHA_3 (chan)
PHA_4 (chan)
A_0 (chan)
A_1 (1/pixel)
A_2 (chan/pixel)
A_3 (chan)
X_CHI
B_0 (chan)
B_1 (1/pixel)
B_2 (chan/pixel)
B_3 (chan/pixel)
Y_CHI
```

4.4.11 RGS_EFFAREACORR

Purpose

Supply empirical corrections to the physical model of the RGS instrument reponse for two separate purposes:

- ensure independent agreement with of RGS models with known cosmic spectra
- improve consistency of simultaneous models of RGS and EPIC-pn spectra

The first of these is a purely RGS mechanism that forms part of the independent calibration of the effective area of the spectrometers and thus forms part of the effective area model delivered to observers by the SAS in RGS response matrices. The second mechanism, known as RGS-pn rectification, combines possible contributions of different origin including any deficiencies in the calibration of either instrument, data analysis procedures or spectral models and thus is not a part of the RGS calibration model. Rather, it provides a convenient mechanism for spectral analysis via RGS matrices modified for this particular purpose.

Functionality

Effective-area correction factors and rectification factors are tabulated against energy in the `EFFAREACORR` and `RECTIFICATION` extensions, repsectively. The relatively small number of elements in the tabulation compared with the number of RGS spectra elements has been determined by the smoothness of the functions concerned. The values at intermediate energies required for RMFs are determined by interpolation in the SAS.

Ground calibration

None.

Maintenance procedures

`EFFAREACORR` correction factors are assumed to result from two effects

- wavelength-dependent effective area correction;
- absorption by contamination on the CCD surfaces.

that are analysed with a combination of observations of power-law spectra, particularly of Mkn421, and sources assumed to be constant, namely RXJ1856.6-3754 and the Vela PWN, in which the observed changes are due to changes in absorption by the contamination layer. The power-law reference source, currently Mkn421, and the contamination reference source, RXJ1856.6-3754, are both observed twice per year, the Vela PWN once per year as a contamination control.

`EFFAREACORR` RGS-pn rectification factors result from more complex mechanisms and therefore need careful consideration before use. The factors are derived from spectral analysis of a number of sources in the XCal archive.

References

http://xmm.esac.esa.int/docs/documents/CAL-SRN-0269-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0262-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0238-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0216-1-0.ps.gz

CCF Structure

RGS[12]_EFFAREACORR_vvvv.CCF

EFFAREACORR Binary Header CCF Keywords

```
EMIN    = 3.20000000000000E-01 / [keV] Minimum Energy
EMAX    = 2.50000000000000E+00 / [keV] Maximum Energy
```

EFFAREACORR Binary Table Columns

```
MJD
ENERGY (keV)
FACTOR
```

RECTIFICATION Binary Header CCF Keywords

```
EMIN    = 3.20000000000000E-01 / [keV] Minimum Energy
EMAX    = 2.50000000000000E+00 / [keV] Maximum Energy
```

RECTIFICATION Binary Table Columns

```
MJD
ENERGY (keV)
FACTOR
```

4.4.12 RGS_EXAFS

Purpose

EXAFS means Extended X-ray Absorption Fine Structure. This CCF describes the detailed absorption lengths of different layers of CCDs in the vicinity of absoption edges to facilitate accurate calculation of quantum efficiencies. Away from the edges, the well-known Henke coefficients are used. There are two notable RGS instrumental absorption features by oxygen and fluorine. Of these oxygen is more important and clearly visible in RGS spectra. Its absorption is modelled as due to a layer of water ice.

Functionality

Absorption lengths for H_2O and MgF_2 are tabulated as a function of energy around the relevant edges.

Ground calibration

None.

Maintenance procedures

As data are accumulated throughout the mission, more accurate descriptions of the absorption features is possible using, for example, the smooth continuum spectrum of Mkn421.

References

http://xmm.esac.esa.int/docs/documents/CAL-SRN-0212-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0171-1-1.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0143-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0108-1-0.ps.gz

CCF Structure

RGS[12]_EXAFS_vvvv.CCF

H2O-P-CCD1 Binary Header CCF Keywords

```
CCDID   =                    1 / CCD Identifier
```

H2O-P-CCD1 Binary Table Columns

```
ENERGY (eV)
MABSLEN (cm)
```

H2O-P-CCD2 Binary Header CCF Keywords

```
CCDID   =                    2 / CCD Identifier
```

H2O-P-CCD2 Binary Table Columns

```
ENERGY (eV)
MABSLEN (cm)
```

H2O-P-CCD3 Binary Header CCF Keywords

```
CCDID   =                    3 / CCD Identifier
```

H2O-P-CCD3 Binary Table Columns

```
ENERGY (eV)
MABSLEN (cm)
```

H2O-P-CCD4 Binary Header CCF Keywords

```
CCDID   =                    4 / CCD Identifier
```

H2O-P-CCD4 Binary Table Columns

```
ENERGY (eV)
MABSLEN (cm)
```

H2O-P-CCD5 Binary Header CCF Keywords

```
CCDID   =                    5 / CCD Identifier
```

H2O-P-CCD5 Binary Table Columns

```
ENERGY (eV)
MABSLEN (cm)
```

H2O-P-CCD6 Binary Header CCF Keywords

```
CCDID   =                    6 / CCD Identifier
```

H2O-P-CCD6 Binary Table Columns

```
ENERGY (eV)
MABSLEN (cm)
```

H2O-P-CCD7 Binary Header CCF Keywords

```
CCDID   =                    7 / CCD Identifier
```

H2O-P-CCD7 Binary Table Columns

```
ENERGY (eV)
MABSLEN (cm)
```

H2O-P-CCD8 Binary Header CCF Keywords

```
CCDID   =                    8 / CCD Identifier
```

H2O-P-CCD8 Binary Table Columns

```
ENERGY (eV)
MABSLEN (cm)
```

H2O-P-CCD9 Binary Header CCF Keywords

```
CCDID   =                    9 / CCD Identifier
```

H2O-P-CCD9 Binary Table Columns

```
ENERGY (eV)
MABSLEN (cm)
```

MgF2-P-CCD1 Binary Header CCF Keywords

```
CCDID   =                    1 / CCD Identifier
```

MgF2-P-CCD1 Binary Table Columns

```
ENERGY (eV)
MABSLEN (cm)
```

MgF2-P-CCD2 Binary Header CCF Keywords

```
CCDID   =                    2 / CCD Identifier
```

MgF2-P-CCD2 Binary Table Columns

```
ENERGY (eV)
MABSLEN (cm)
```

MgF2-P-CCD3 Binary Header CCF Keywords

```
CCDID   =                    3 / CCD Identifier
```

MgF2-P-CCD3 Binary Table Columns

```
ENERGY (eV)
MABSLEN (cm)
```

MgF2-P-CCD4 Binary Header CCF Keywords

```
CCDID   =                    4 / CCD Identifier
```

MgF2-P-CCD4 Binary Table Columns

```
ENERGY (eV)
MABSLEN (cm)
```

MgF2-P-CCD5 Binary Header CCF Keywords

```
CCDID   =                    5 / CCD Identifier
```

MgF2-P-CCD5 Binary Table Columns

```
ENERGY (eV)
MABSLEN (cm)
```

MgF2-P-CCD6 Binary Header CCF Keywords

```
CCDID   =                    6 / CCD Identifier
```

MgF2-P-CCD6 Binary Table Columns

```
ENERGY (eV)
MABSLEN (cm)
```

MgF2-P-CCD7 Binary Header CCF Keywords

```
CCDID   =                    7 / CCD Identifier
```

MgF2-P-CCD7 Binary Table Columns

```
ENERGY (eV)
MABSLEN (cm)
```

MgF2-P-CCD8 Binary Header CCF Keywords

```
CCDID   =                    8 / CCD Identifier
```

MgF2-P-CCD8 Binary Table Columns

```
ENERGY (eV)
MABSLEN (cm)
```

MgF2-P-CCD9 Binary Header CCF Keywords

```
CCDID   =                    9 / CCD Identifier
```

MgF2-P-CCD9 Binary Table Columns

```
ENERGY (eV)
MABSLEN (cm)
```

4.4.13 RGS_HKPARMINT

Purpose

Define ranges of acceptable values for housekeeping, HK, and auxiliary data for the generation by the SAS of good time intervals, or GTIs, for selecting RGS data of good quality.

Functionality

There are 3 tables. The `HKPARMINT` table gives the names of HK parameters as defined in telemetry and acceptable ranges of the subset of HK parameters that are checked. The telemetry names are functional labels not connected with any physical meaning. The `TRANSLATION` table explains the meaning of a complete set of HK parameters including those in the `HKPARMINT` table. The `AUXPAR` table combines similar information for quantities related to on-board data processing supplied in the ODF files *R[12]*AUX.FIT.

Ground calibration

Initial values supplied from ground calibrations and failure analysis.

Maintenance procedures

Modifications of allowed parameters ranges are expected following analysis of any failures of the SAS to generate RGS GTIs.

References

http://xmm.esac.esa.int/docs/documents/CAL-SRN-0134-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0111-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0107-1-1.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0106-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0085-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0051-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0050-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0013-1-0.ps.gz

CCF Structure

RGS[12]_HKPARMINT_vvvv.CCF

HKPARMINT Binary Table Columns

`PARM_NAME`
`PARM_RANGE`

TRANSLATION Binary Table Columns

`SHORT_NAME`
`LONG_NAME`

AUXPAR Binary Table Columns

`PARM_NAME`
`PARM_RANGE`
`PARM_DESC`
`PARM_UNIT`

4.4.14 RGS_LINCOORD

Purpose

Geometry of the RGS grating and detector systems.

Functionality

As the RGS is a dispersive instrument, where the wavelength of an event is determined by the position at which it strikes the detector following grating reflection, accurate knowledge of the instrumental geometry is fundamental. Grating geometry is held in header keywords; and the geometry of individual CCD detectors in the binary table.

Ground calibration

RGS geometry measured during instrument integration.

Maintenance procedures

Geometry of the whole system of gratings and detectors is subject to verification through comparison of measured wavelengths of lines with laboratory values. Further checks on CCD alignment are expected using continuity of accumulated data across chip boundaries.

References

`http://xmm.esac.esa.int/docs/documents/CAL-SRN-0081-1-0.ps.gz`
`http://xmm.esac.esa.int/docs/documents/CAL-SRN-0017-1-0.ps.gz`

CCF Structure

RGS[12]_LINCOORD_vvvv.CCF

LINCOORD Binary Header CCF Keywords

```
MM_PX_X = 2.70000000000000E-02 / [mm] pixel size in X
MM_PX_Y = 2.70000000000000E-02 / [mm] pixel size in Y
MM_PX_LY= 2.70000000000000E-02 / [mm] Y-size of last pixel row
CC12_TX = 0.00000000000000E+00 / [mm] CAMCOORD1->2 translation in X
CC12_TY = 0.00000000000000E+00 / [mm] CAMCOORD1->2 translation in Y
CC12_TZ = 0.00000000000000E+00 / [mm] CAMCOORD1->2 translation in Z
R_CHIP_T= 2.93000000000000E+02 / [K] reference chip temperature of alignment dat
R_CAM_T = 2.93000000000000E+02 / [K] reference camera temperature of alignment d
EXP_CHIP= 1.00000000000000E-06 / [1/K] therm. expansion coeff. for chip (Si)
EXP_CAM = 2.00000000000000E-05 / [1/K] therm. expansion coeff. for camera (Al)
CC2_TX  = 7.47178000000000E+03 / [mm] CAMCOORD2 origin in telescope frame - X
CC2_TY  = 0.00000000000000E+00 / [mm] CAMCOORD2 origin in telescope frame - Y
CC2_TZ  = -5.62630000000000E+02 / [mm] CAMCOORD2 origin in telescope frame - Z
CC2_RPH = -9.00000000000000E+01 / [deg] CAMCOORD2 orien. in SACCORD - PHI
CC2_RTH = 8.26670000000000E+01 / [deg] CAMCOORD2 orien. in SACCCORD - THETA
CC2_RPS = 9.00000000000000E+01 / [deg] CAMCOORD2 orien. in SACCCORD - PSI
G_TX    = 7.95540000000000E+02 / [mm] X coordinate of G-point in SACCOORD
G_TY    = 0.00000000000000E+00 / [mm] Y coordinate of G-point in SACCOORD
G_TZ    = 0.00000000000000E+00 / [mm] Z coordinate of G-point in SACCOORD
G_RX    = 0.00000000000000E+00 / [deg] Rotation of RGA around X
G_RY    = 0.00000000000000E+00 / [deg] Rotation of RGA around Y
G_RZ    = 0.00000000000000E+00 / [deg] Rotation of RGA around Z
G_ALPHA = 1.57619100000000E+00 / [deg] nominal on-axis grating angle of incidenc
ARC_MM_X= 2.73330000000000E+01 / [arcsec/mm] plate scale in X
ARC_MM_Y= 2.73330000000000E+01 / [arcsec/mm] plate scale in Y
MM_TX   = 0.00000000000000E+00 / [mm] origin of telescope frame in SACCOORD - X
MM_TY   = -4.56830000000000E+02 / [mm] origin of telescope frame in SACCOORD - Y
MM_TZ   = 2.63750000000000E+02 / [mm] origin of telescope frame in SACCOORD - Z
```

LINCOORD Binary Table Columns

CCD_ID
NODE_ID
X0 (mm)
Y0 (mm)
Z0 (mm)
D_X0 (mm)
D_Y0 (mm)

D_Z0 (mm)
EULER_PHI (rad)
EULER_THETA (rad)
EULER_PSI (rad)
D_EULER_PHI (rad)
D_EULER_THETA (rad)
D_EULER_PSI (rad)
RAWX0 (pixel)
RAWY0 (pixel)

4.4.15 RGS_LINESPREADFUNC

Purpose

Describe the instrinsic delta-function response of the RGAs, known as the line spread function or LSF.

Functionality

The LSF results from the convolution of several physical components starting from the angular distribution of light incident on the gratings from the mirrors given in `XRT1_XPSF` and `XRT2_XPSF`. The specific RGA components described here are

`FIGURE` Misalignment distribution of the set of elements in the grating assembly.

`BOWS` Non-flatness distribution of the set of elements in the grating assembly.

`SCATTER` Small-angle scatter from grating surface.

`LASCAT` Large-angle scatter from grating surface.

Ground calibration

An accurate set of values was supplied from pre-launch measurements.

Maintenance procedures

The CCF model LSF is nevertheless subject to verification in-flight by comparison of 1st and 2nd order data ideally of narrow-lined sources with spectral models in XSPEC for example, calculated using RGS response matrices calculated by the SAS.

References

http://xmm.esac.esa.int/docs/documents/CAL-SRN-0088-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0065-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0046-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0014-1-0.ps.gz

CCF Structure

RGS[12]_LINESPREADFUNC_vvvv.CCF

FIGURE Binary Header CCF Keywords

```
SCALE   = 1.40000000000000E+00 / grating misalignment scale factor
DXOFFS  = 0.00000000000000E+00 / [mm] offset of RFC defocus
RGA_HEW = 7.77777800000000E-04 / [deg] Grating mis-alignment HEW
ALGOID  =                    1 / argorithm identifyer
```

FIGURE Binary Table Columns

```
AMPL
CENTER (rad)
SIGMA (rad)
```

SCATTER Binary Table Columns

```
A
B
RMS (angstrom)
CORLEN (um)
```

LASCAT Binary Header CCF Keywords

```
ALGOID  =                    1 / argorithm identifyer
```

LASCAT Binary Table Columns

```
WAVELENG (Angstrom)
RMS (Angstrom)
```

BOWS Binary Table Columns

```
AMPL
CENTER (rad)
SIGMA (rad)
```

4.4.16 RGS_MODEPARAM

Purpose

Provide summary parameters of the read-out operation of instrument modes.

Functionality

These data, most not used by the SAS, record information related to CCD readout operations described in `RGS[12]_CLOCKPATTERNS_vvvv.CCF`. On-board software methods encapsulated in the `DPPVERS` keyword and do determine the corresponding SAS methods.

Ground calibration

Initial values described the on-board methods at launch.

Maintenance procedures

Changes to on-board software require a new CCF.

References

http://xmm.esac.esa.int/docs/documents/CAL-SRN-0015-1-0.ps.gz

CCF Structure

RGS[12]_MODEPARAM_vvvv.CCF

MODEPARAM Binary Header CCF Keywords

```
DPPVERS = 1.90000000000000E+01 / version of DPP code
```

MODEPARAM Binary Table Columns

```
MODE_ID
PAR_TRANS_TIME (ms)
SER_TRANS_TIME (ms)
PIXEL_READOUT_TIME (ms)
```

4.4.17 RGS_QUANTUMEF

Purpose

This is one of the most important RGS calibration files, whose separate extensions describe different aspects of the physical model of the ensemble of gratings and CCD detectors that form an RGS instrument:

`CCD_DESC` Thickness of CCD layers of passive material.

`SIn` Total thickness of CCDs $1 \leq n \leq 9$ including `CCD_DESC` dead layers.

`RGA_EFF`

`RGA_INTERCEPT`

`RGA_OBSCURATE`

`RGA_EFFCORR`

`RGA_SELFVIGNCORR`

`RGA_EFFAREACORR`

Functionality

This set of data provides the basis of the physical model of the RGS effective area derived before and immediately after launch. It provides the starting point for the further empirical effective area corrections calculated from the response of the RGS instruments to cosmic sources of known spectra given in `RGS[12]_EFFAREACORR_vvvv.CCF`.

Ground calibration

The complete physical model of the RGS devised before launch on the basis of ground measurements was adjusted during the early calibration phase.

Maintenance procedures

Now that the `RGS[12]_EFFAREACORR_vvvv.CCF` is in place, `QUANTUMEF` components are not expected to change.

References

http://xmm.esac.esa.int/docs/documents/CAL-SRN-0215-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0176-1-1.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0143-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0108-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0071-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0012-1-0.ps.gz

CCF Structure

RGS[12]_QUANTUMEF_vvvv.CCF

CCD_DESC Binary Table Columns

```
CCDID
EEV_ID
D_SI (nm)
D_AL (nm)
D_MGF2 (nm)
D_SIO2 (nm)
D_H2O (nm)
D_ALO2 (nm)
SENS_M1
SENS_M2
SENS_M3
SENS_M4
SENS_M5
```

SI1 Binary Header CCF Keywords

```
MTHICK  = 2.66000000000000E+01 / [micron] Middle thickness
```

SI2 Binary Header CCF Keywords

```
MTHICK  = 3.12000000000000E+01 / [micron] Middle thickness
```

SI3 Binary Header CCF Keywords

```
MTHICK  = 2.72000000000000E+01 / [micron] Middle thickness
```

SI4 Binary Header CCF Keywords

```
MTHICK  = 2.71000000000000E+01 / [micron] Middle thickness
```

SI5 Binary Header CCF Keywords

```
MTHICK  = 3.09000000000000E+01 / [micron] Middle thickness
```

SI5 Binary Table Columns

```
CHIPX (pixel)
CHIPY (pixel)
FRINGES
```

SI6 Binary Header CCF Keywords

```
MTHICK  = 3.12000000000000E+01 / [micron] Middle thickness
```

SI6 Binary Table Columns

```
CHIPX (pixel)
CHIPY (pixel)
FRINGES
```

SI7 Binary Header CCF Keywords

```
MTHICK  = 3.13000000000000E+01 / [micron] Middle thickness
```

SI8 Binary Header CCF Keywords

```
MTHICK  = 2.73000000000000E+01 / [micron] Middle thickness
```

SI8 Binary Table Columns

```
CHIPX (pixel)
CHIPY (pixel)
FRINGES
```

SI9 Binary Header CCF Keywords

```
MTHICK  = 3.34000000000000E+01 / [micron] Middle thickness
```

SI9 Binary Table Columns

```
CHIPX (pixel)
CHIPY (pixel)
FRINGES
```

RGA_EFF Binary Header CCF Keywords

```
BETA_VIG= 3.03800000000000E+00 / [deg] onset of self vignetting in BETA
```

RGA_EFF Binary Table Columns

```
ALPHA (deg)
LAMBDA (angstrom)
EFF0
EFF1
EFF2
EFF3
EFF4
EFF5
```

RGA_INTERCEPT Binary Table Columns

```
ENERGY (eV)
FRACTION
```

RGA_OBSCURATE Binary Table Columns

```
ENERGY (eV)
FRACTION
```

RGA_EFFCORR Binary Table Columns

```
LAMBDA (angstrom)
CORR0
CORR1
CORR2
CORR3
CORR4
CORR5
```

RGA_SELFVIGNCORR Binary Table Columns

```
BETA (rad)
FACTOR
```

RGA_EFFAREACORR Binary Table Columns

```
ENERGY (eV)
FACTOR
```

4.4.18 RGS_REDIST

Purpose

Parameterisation of CCD energy response to monochromatic radiation, known as redistribution.

Functionality

The energy sensitivity of the CCD detectors allows the separation of spectral orders in the RGS. The redistribution functions coded here allow definition of calibrated selection regions for the accumulation of spectra.

Ground calibration

An initial set of parameters was derived from data on narrow lines. Due to the absence of consistent data for all CCDs, where variations of the response could be verified in detail, all CCDs were given the same set of parameters.

Maintenance procedures

In flight, the energy response of each CCD is verified using strong lines of known wavelength in suitable objects such as Capella. Two other lines are available from the in-flight calibration sources.

References

`http://xmm.esac.esa.int/docs/documents/CAL-SRN-0067-1-0.ps.gz`
`http://xmm.esac.esa.int/docs/documents/CAL-SRN-0011-1-0.ps.gz`

CCF Structure

RGS[12]_REDIST_vvvv.CCF

EBOUNDS Binary Table Columns

```
CHANNEL (channel)
E_MIN (eV)
E_MAX (eV)
```

CCD_REDISTRIBUTION Binary Header CCF Keywords

```
FANO    = 1.30000000000000E-01 / Fano factor
SI_THICK= 2.50000000000000E-05 / [m] CCD thickness
SCALEFAC= 1.04781000000000E-08 / [m] Model scale factor
E_RMS   = 5.40000000000000E+00 / [e] Readout noise
EH_E    = 3.65000000000000E+00 / [eV] e-h pair creation energy
EVT_SIZE= 1.40000000000000E+00 / [pixel] Average event size (for OCB=1)
E_THR   = 5.00000000000000E+01 / [eV] Energy threshold
PARTNORM= 3.60000000000000E-02 / Const factor in normalization of partial events
PARTDECY= 1.83258150000000E-03 / [1/eV] Const factor in normalization of partial
PARTOFFS= 0.00000000000000E+00 / [eV] Offset of exponential in normalization of pa
PARTEREF= 1.90000000000000E+03 / [eV] Energy of reference absorption length in n
```

4.4.19 RGS_TEMPLATEBCKGND

Purpose

Provide source-free rate-dependent background spectra.

Functionality

Background spectrum templates are accumulated in intervals of background rate. A background spectrum for an individual observation is then synthesised from a combination of the templates weighted according to the variations of the observed background rate.

Ground calibration

None.

Maintenance procedures

XMM makes regular observations of X-ray sources that are too weak for detection with the RGS, whose data are then dominated by the background. Observations of this type are the raw material for the `TEMPLATEBCKGND` CCFs. No assessment has yet been made of background variability.

Selection of observations of empty RGS files

Suitable observations should be longer than about 20 ksec, cover all the considered time period, and have different background levels, in particular the highest ones, so the sample should include observations taken at the beginning and end of the science window. It is helpful to consider objects for which no RGS spectrum is expected or to look at the RGS monitor pages.

Each observation must be examined to check that there is no significant RGS signal. This can be done by inspection of

- image and banana images in the PPS
- cross-dispersion distribution of RGS events with SAS or IDL

SAS processing

SAS processing with the latest software and CCF files is performed as follows

- Generation of filtered event files, source lists and extracted spectra for ONAXIS coordinates

```
rgsproc xpsfincl=100 pdistincl=95 finalstage=2:angles
rgssources srclist=P0123450101RXS099SRCLI_0000.FIT changeprime=yes \
           primelabel='ONAXIS'
rgsproc entrystage=2:angles finalstage=4:spectra
```

- Generation of background light curves

```
evselect table=P0123450101RXS099EVENLI0000.FIT \
         expression=((XDSP_CORR<-3.E-4)||(XDSP_CORR>3.E-4))&&(CCDNR==9) \
         filtertype=expression rateset=rateRXS099.fits \
         timecolumn=TIME timebinsize=100 maketimecolumn=yes \
         tmakeratecolumn=yes withrateset=yes
```

- Generation of GTIs for each of the 16 pre-defined background levels
  - if no there are no data for some levels their GTIs will be empty

```
tabgtigen table=rateRXS099.fits gtiset=rR1S004_Y.YY.fits \
          expression='RATE > Y.YY && RATE <= Z.ZZ' \
          timecolumn='TIME'
```

- Generation of filtered event files for each of the 16 pre-defined background levels

```
rgsfilter mergedset=P0123450101RXS099EVENLI0000.FIT \
          evlist=RXS099_Y.YY.FIT \
          auxgtitables=rRXS099_Y.YY.fits \
          withcombmap=no
```

- Extraction of spectra from each of the filtered event files for
  - 1st and 2nd order
  - PI extraction regions 90 and 95%

```
rgsspectrum evlist=RXS099_Y.YY.FIT srclist=RX_SRCLI_XX.FIT \
            bkgcorrect=no spectrumset=RXS099SPEC_Y.YY.FIT \
            withbkgset=no source=2 order=1
```

Combination of the selected observations to make the templates and build CCFs

This step is done mostly interactively and uses some basic ad-hoc IDL procedures, mainly for testing and checking of the consistency of the results.Once the observations have been selected and the count spectra extracted, the available spectra of each background level are accumulated. The combined spectrum is built as follows

Add `COUNTS`

Average `BACKSCAL`

Average `AREASCAL`

Average `QUALITY` such that `QUALITY`=0—1

Exposure times are added and the total is written in the `EXPOSURE` keyword of the header of the output FITS file. Finally, the CCFs are built from these templates.

References

http://xmm.esac.esa.int/docs/documents/CAL-SRN-0261-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0250-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0229-1-1.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0217-1-0.ps.gz
http://xmm.esac.esa.int/docs/documents/CAL-SRN-0179-1-1.ps.gz

CCF Structure

RGS[12]_TEMPLATEBCKGND_vvvv.CCF

X100_P090_1_0.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 9.90640859375000E+04 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 0.00000000000000E+00 / Background level
```

X100_P090_1_0.00 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_1_0.01 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.52006843750000E+05 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 1.00000000000000E-02 / Background level
```

X100_P090_1_0.01 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_1_0.02 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 4.10720093750000E+05 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 2.00000000000000E-02 / Background level
```

X100_P090_1_0.02 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_1_0.04 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 3.43383250000000E+05 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 4.00000000000000E-02 / Background level
```

X100_P090_1_0.04 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_1_0.06 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.75616515625000E+05 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 6.00000000000000E-02 / Background level
```

X100_P090_1_0.06 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_1_0.08 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 9.06328437500000E+04 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 8.00000000000000E-02 / Background level
```

X100_P090_1_0.08 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_1_0.10 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.80498343750000E+05 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 1.00000000000000E-01 / Background level
```

X100_P090_1_0.10 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
```

```
AREASCAL
BACKSCAL
```

X100_P090_1_0.20 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.25412320312500E+05 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 2.00000000000000E-01 / Background level
```

X100_P090_1_0.20 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_1_0.40 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 6.82629453125000E+04 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 4.00000000000000E-01 / Background level
```

X100_P090_1_0.40 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_1_0.60 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 3.95844375000000E+04 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 6.00000000000000E-01 / Background level
```

X100_P090_1_0.60 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_1_0.80 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 2.84314257812500E+04 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 8.00000000000000E-01 / Background level
```

X100_P090_1_0.80 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_1_1.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 7.07511562500000E+04 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 1.00000000000000E+00 / Background level
```

X100_P090_1_1.00 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_1_2.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 3.62098828125000E+04 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 2.00000000000000E+00 / Background level
```

X100_P090_1_2.00 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_1_4.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.04597470703125E+04 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 4.00000000000000E+00 / Background level
```

X100_P090_1_4.00 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
```

AREASCAL
BACKSCAL

X100_P090_1_6.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.04823798828125E+04 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 6.00000000000000E+00 / Background level
```

X100_P090_1_6.00 Binary Table Columns

CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL

X100_P090_1_8.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.60749938964844E+03 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 8.00000000000000E+00 / Background level
```

X100_P090_1_8.00 Binary Table Columns

CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL

X100_P090_2_0.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 9.90640859375000E+04 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 0.00000000000000E+00 / Background level
```

X100_P090_2_0.00 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_2_0.01 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.52006843750000E+05 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 1.00000000000000E-02 / Background level
```

X100_P090_2_0.01 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_2_0.02 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 4.10720093750000E+05 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 2.00000000000000E-02 / Background level
```

X100_P090_2_0.02 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_2_0.04 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 3.43383250000000E+05 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 4.00000000000000E-02 / Background level
```

X100_P090_2_0.04 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_2_0.06 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.75616515625000E+05 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 6.00000000000000E-02 / Background level
```

X100_P090_2_0.06 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
```

AREASCAL
BACKSCAL

X100_P090_2_0.08 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 9.06328437500000E+04 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 8.00000000000000E-02 / Background level
```

X100_P090_2_0.08 Binary Table Columns

CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL

X100_P090_2_0.10 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.80498343750000E+05 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 1.00000000000000E-01 / Background level
```

X100_P090_2_0.10 Binary Table Columns

CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL

X100_P090_2_0.20 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.25412320312500E+05 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 2.00000000000000E-01 / Background level
```

X100_P090_2_0.20 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_2_0.40 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 6.82629453125000E+04 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 4.00000000000000E-01 / Background level
```

X100_P090_2_0.40 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_2_0.60 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 3.95844375000000E+04 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 6.00000000000000E-01 / Background level
```

X100_P090_2_0.60 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_2_0.80 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 2.84314257812500E+04 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 8.00000000000000E-01 / Background level
```

X100_P090_2_0.80 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_2_1.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 7.07511562500000E+04 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 1.00000000000000E+00 / Background level
```

X100_P090_2_1.00 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
```

```
AREASCAL
BACKSCAL
```

X100_P090_2_2.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 3.62098828125000E+04 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 2.00000000000000E+00 / Background level
```

X100_P090_2_2.00 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_2_4.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.04597470703125E+04 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 4.00000000000000E+00 / Background level
```

X100_P090_2_4.00 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P090_2_6.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.04823798828125E+04 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 6.00000000000000E+00 / Background level
```

X100_P090_2_6.00 Binary Table Columns

CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL

X100_P090_2_8.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.00000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.60749938964844E+03 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 8.00000000000000E+00 / Background level
```

X100_P090_2_8.00 Binary Table Columns

CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL

X100_P095_1_0.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 9.90640859375000E+04 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 0.00000000000000E+00 / Background level
```

X100_P095_1_0.00 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P095_1_0.01 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.52006843750000E+05 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 1.00000000000000E-02 / Background level
```

X100_P095_1_0.01 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P095_1_0.02 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 4.10720093750000E+05 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 2.00000000000000E-02 / Background level
```

X100_P095_1_0.02 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
```

AREASCAL
BACKSCAL

X100_P095_1_0.04 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 3.43383250000000E+05 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 4.00000000000000E-02 / Background level
```

X100_P095_1_0.04 Binary Table Columns

CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL

X100_P095_1_0.06 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.75616515625000E+05 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 6.00000000000000E-02 / Background level
```

X100_P095_1_0.06 Binary Table Columns

CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL

X100_P095_1_0.08 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 9.06328437500000E+04 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 8.00000000000000E-02 / Background level
```

X100_P095_1_0.08 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P095_1_0.10 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.80498343750000E+05 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 1.00000000000000E-01 / Background level
```

X100_P095_1_0.10 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P095_1_0.20 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.25412320312500E+05 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 2.00000000000000E-01 / Background level
```

X100_P095_1_0.20 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P095_1_0.40 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 6.82629453125000E+04 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 4.00000000000000E-01 / Background level
```

X100_P095_1_0.40 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P095_1_0.60 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 3.95844375000000E+04 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 6.00000000000000E-01 / Background level
```

X100_P095_1_0.60 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
```

```
AREASCAL
BACKSCAL
```

X100_P095_1_0.80 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 2.84314257812500E+04 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 8.00000000000000E-01 / Background level
```

X100_P095_1_0.80 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P095_1_1.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 7.07511562500000E+04 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 1.00000000000000E+00 / Background level
```

X100_P095_1_1.00 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P095_1_2.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 3.62098828125000E+04 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 2.00000000000000E+00 / Background level
```

X100_P095_1_2.00 Binary Table Columns

CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL

X100_P095_1_4.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.04597470703125E+04 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 4.00000000000000E+00 / Background level
```

X100_P095_1_4.00 Binary Table Columns

CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL

X100_P095_1_6.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.04823798828125E+04 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 6.00000000000000E+00 / Background level
```

X100_P095_1_6.00 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P095_1_8.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.60749938964844E+03 / [s] Maximum effective integration time
ORDER   =                    1 / Grating order
LEVEL   = 8.00000000000000E+00 / Background level
```

X100_P095_1_8.00 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P095_2_0.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 9.90640859375000E+04 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 0.00000000000000E+00 / Background level
```

X100_P095_2_0.00 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
```

```
AREASCAL
BACKSCAL
```

X100_P095_2_0.01 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.52006843750000E+05 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 1.00000000000000E-02 / Background level
```

X100_P095_2_0.01 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P095_2_0.02 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 4.10720093750000E+05 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 2.00000000000000E-02 / Background level
```

X100_P095_2_0.02 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P095_2_0.04 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 3.43383250000000E+05 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 4.00000000000000E-02 / Background level
```

X100_P095_2_0.04 Binary Table Columns

CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL

X100_P095_2_0.06 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.75616515625000E+05 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 6.00000000000000E-02 / Background level
```

X100_P095_2_0.06 Binary Table Columns

CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL

X100_P095_2_0.08 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 9.06328437500000E+04 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 8.00000000000000E-02 / Background level
```

X100_P095_2_0.08 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P095_2_0.10 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.80498343750000E+05 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 1.00000000000000E-01 / Background level
```

X100_P095_2_0.10 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P095_2_0.20 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.25412320312500E+05 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 2.00000000000000E-01 / Background level
```

X100_P095_2_0.20 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
```

AREASCAL
BACKSCAL

X100_P095_2_0.40 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 6.82629453125000E+04 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 4.00000000000000E-01 / Background level
```

X100_P095_2_0.40 Binary Table Columns

CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL

X100_P095_2_0.60 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 3.95844375000000E+04 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 6.00000000000000E-01 / Background level
```

X100_P095_2_0.60 Binary Table Columns

CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL

X100_P095_2_0.80 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 2.84314257812500E+04 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 8.00000000000000E-01 / Background level
```

X100_P095_2_0.80 Binary Table Columns

CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL

X100_P095_2_1.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 7.07511562500000E+04 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 1.00000000000000E+00 / Background level
```

X100_P095_2_1.00 Binary Table Columns

CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL

X100_P095_2_2.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 3.62098828125000E+04 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 2.00000000000000E+00 / Background level
```

X100_P095_2_2.00 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P095_2_4.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.04597470703125E+04 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 4.00000000000000E+00 / Background level
```

X100_P095_2_4.00 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL
```

X100_P095_2_6.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.04823798828125E+04 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 6.00000000000000E+00 / Background level
```

X100_P095_2_6.00 Binary Table Columns

```
CHANNEL
COUNTS
QUALITY
```

AREASCAL
BACKSCAL

X100_P095_2_8.00 Binary Header CCF Keywords

```
ALGOID  =                    1 / algorithm identifier
XPSFFRA = 1.00000000000000E+00 / PSF fraction used in cross-dispersion direction
PIPSFFRA= 9.50000000000000E-01 / PSF fraction used in PI direction
EXPOSURE= 1.60749938964844E+03 / [s] Maximum effective integration time
ORDER   =                    2 / Grating order
LEVEL   = 8.00000000000000E+00 / Background level
```

X100_P095_2_8.00 Binary Table Columns

CHANNEL
COUNTS
QUALITY
AREASCAL
BACKSCAL

4.5 OM Related CCF Components

4.5.1 ASTROMET

4.5.1.1 Description

This CCF component is used by QLA and the SAS. A copy of this CCF component is also used by SGS tools.
There are twelve extension - one for each filter - named FILTER-FilterId. Two extensions are required for each grism to describe the displacement of the zero orders with respect to the direct image obtained with the lenticular filters.

Two additional extensions contain the polynomial coefficients describing the distortion for all the filters.

FILTER-U
FILTER-B
FILTER-V
FILTER-UVW1
FILTER-UVM2
FILTER-UVW2
FILTER-WHITE
FILTER-MAGNI
FILTER-GRISM1
FILTER-GRISM2
FILTER-GRISM10
FILTER-GRISM20
POLYNOM_MAP
POLYNOM_MAP2

Each extension FILTER-FilterId holds a binary table sampling the detector distortion of a specific filter at a coarse resolution ($\sim$ 1 datapoint every 25 subpixel). Intermediate detector position have to be obtained by interpolation. The distortion at the position RAWX, RAWY is described as the difference between the measured image position on the detector and the position assuming a linear grid. The platescale of the linear grid is the value of the keyword PLTSCALE (units: arcsec/pixel). The format of the binary tables follows the description of the CCF ICD [5].

The distortion can alternatively be obtained using two polynomials describing the offset in x- and y-direction. Analysis of calibration data showed that the detector distortion can be parameterized with an up to 7th order polynomial. The 36 parameters a_0, ..., a_{35} (cf. section 3.5.5) needed to calculate the distortion in x- and y-direction at any detector position (equivalent to RAWX_OFF and RAWY_OFF) are stored in the columns XPOLYCOEF, YPOLYCOEF of the extension POLYNOM_MAP. There is one set of coefficients per filter element in the POLYNOM_MAP extension. The filter element is found in the column FILTER_ID. The required binary table entry is identified by matching the value of the column FILTER_ID with the State Variable FilterID. The reverse transformation from distorted to flat image is available from the polynomial stored in the extension POLYNOM_MAP2. This has an identical format to the POLYNOM_MAP block.

The format of the AstroMet file is as follows

Binary table: FILTER-FilterID
Twelve binary extensions, one per FilterID

Binary table: POLYNOM_MAP

column name	column type	column unit	comment
FILTER_ID	9A	not applicable	filter ID
PLTSCALE	E	arcsec/subpixel	platescale of filter specified in FILTER_ID
XPOLYCOEF	36E	not applicable	coefficients of x-distortion polynomial in ascending order
YPOLYCOEF	36E	not applicable	coefficients of y-distortion polynomial in ascending order

Binary table: POLYNOM_MAP2

column name	column type	column unit	comment
FILTER_ID	9A	not applicable	filter ID
PLTSCALE	E	arcsec/subpixel	platescale of filter specified in FILTER_ID
XPOLYCOEF	36E	not applicable	coefficients of x-distortion polynomial in ascending order
YPOLYCOEF	36E	not applicable	coefficients of y-distortion polynomial in ascending order

4.5.1.2 Generation from Ground Calibration Data

The distortion was first derived from CSL pinhole measurements. Double pinholes were excluded from this analysis. The pinhole positions were scaled, rotated and shifted until the best match between pinhole position and measured image was achieved. A pinhole starmask image in the correct position was generated (starmask_new.fits). The adopted parameters were:

$$\begin{aligned} \text{rotation angle} &= 62^\circ \text{ clockwise} \\ \text{xshift} &= 196 \text{ -0.41669 pixels} \\ \text{yshift} &= \text{-34 -0.08849 pixels} \end{aligned}$$

The platescale was assumed as 0.499844 arcsec/pixel in the generation of the pinhole image, which is based on the following measurements on the OM FM:

$$\begin{aligned} \text{Pixel Size} &= 9.2675\text{x10-4 cm} \\ \text{Focal Length} &= 382.431 \text{ cm} \end{aligned}$$

Pinholes starmask positions and centroids of the measured images were paired, and the difference between measured and expected positions is written to a file (routine DISTORT.PRO). Finally the offsets are parameterized as function of field position, more precisely as function of the distance to the boresight position, i.e. (x-1024.5). The offsets were fitted with an up to 7th order polynomial (cf. section 3.5.5) using the routine DISTORT_FORM.PRO
The analysis was performed by Alice Breeveld (MSSL).

4.5.1.3 Generation from In-Orbit Data

Analysis of in-orbit data closely follow the CSL data analysis. Stars with known astrometric positions are matched with their image centroids. Objects from different observations can be used.
The photometric calibration fields are here especially useful, as they contain a list of objects with accurately known position. The measured position, the expected position and the offset between the measured and expected positions are calculated and written into a file. An equivalent of the IDL-routine DISTORT_FORM.PRO can be used to analysis the data:

- select catalogued positions for different observations/pointings (e.g. positions of objects in photometric sequence)
- preselect fit input sample on brightness range and position reliability
- fit data with polynomial
- calculate residuals
- sort residuals on position
- average residual within resolution element of CCF file
- force residual of V-filter at center to zero
- polynomial fit to averaged residuals

The new distortion map was generated from in-flight observations of the G153 field. The U filter was used and the positions of 813 stars were measured. The old map was based on a LMC field with only 230 sources. Since the main source of distortion is the detector fiber taper, this map is applied to all filters, including two OM grisms. The platescale is a natural by-product of this analysis. In orbit the platescale was found to be 0.4765 arcsec/subpixel for the V-filter.

The distortion map for the grisms, which describes the displacement of the zero orders in the grism image with respect to the direct image was measured in the same way. Observations of the Sco X-1 field in revs. 402 and 688 using the V filter and both grisms have been used to measure the positions of the sources and their respective zero orders. Then a map has been computed following the same method described for the geometric distortion.

4.5.2 BADPIX

4.5.2.1 Description

This CCF component is used by QLA and the SAS.
The format of the binary tables follows the description of the CCF ICD [5].

The possible types of bad pixels and the mapping between CCF description and the integer encoding used in the output badpixel map is described in section 3.5.18.

4.5.2.2 Generation from Ground Calibration Data

In the CSL images there is clear evidence of edge emission. The area of edge emission has been flagged as bad pixels by

- co-adding all CSL images not degraded by the electronic problem
  file: rcslsummed.fits.gz
- computation of the average pixel value in the lower and upper left corner. (Note that the median in both areas was zero).
- flagging pixels as bad with an pixel content of greater equal two times the average value.

Pinhole images are almost free of any background and are not suitable to identify further defects.

Because of the problem in the OM electronics neither flatfield nor darkframe images were acquired at CSL. Although such measurements are available for the intensifier, these measurements are not directly usable, because they provide relative positional information only. I.e. these measurements are suitable to quantify defects, but not to derive their exact position. The CCD used in these measurements had an arbitrary location in respect to the intensifier. Two images were taken with the CCD rotated by 90 degree to distinguish between intensifier and CCD defects.

The relative position of identified defects to each other can be derived. Study is ongoing whether these defects can be found in the CSL image. In this case these pixels will be added to the badpix file.

4.5.2.3 Generation from In-Orbit Data

Several composite flatfields (superflats) have been generated from all Eng. 4 flatfield observations. After removing the LED pattern, these superflats where used to identify OM bad pixels. These badpixels are real physical pixels, thus include 8x8 subpixels. In addition, some observations of extented sources, such as comets, were also used to identify low sensitivity pixels.

Significant deviations from the median or average values were spotted and flagged as bad pixels. The sensitivity values associated with the badpixel are suitable for a rough quality flag but not good enough for accurate correction. This CCF is used by SAS to identify the badpixel area. The "1" in the QFLAG column in the OM source list file means that there are one or more bad pixels within the photometry aperture.

The signature of the different bad pixel types is listed below:

type of defect	recognition
turn on channels	positive deviation in darkframe
hot CCD pixel	8x8 subpixel positive deviation in full frame image
edge emission	positive deviation in full frame
dead CCD pixel	8x8 subpixel negative deviation in slew flatfield
dead pore	negative deviation in slew flatfield
bad CCD pixel	any 8x8 subpixel deviation in slew flatfield
bad for unknown reason	any deviation
area of reduced sensitivity	negative deviation in slew flatfield

4.5.3 BORESIGHT

4.5.3.1 Description

This CCF component is used by QLA and the SAS. A copy of this CCF component is also used by the PHS tools.

For OM this CCF describes the missalignment between the OM optical axes and the star tracker. It is used in mission planning to predict the position of Field Acquisition Stars and targets.

SAS uses this CCF in the computation of astronomical coordinates of the detected sources.

The reference point for the boresight in the OM detector is defined as the lower left edge of pixel (1025,1025) in the V-filter. (Here it is assumed that the pixels are numbered starting with 1.)

4.5.3.2 Generation from Ground Calibration Data

4.5.3.3 Generation from In-Orbit Data

Offsets measured during Field Acquisition in 1500 observations are used to compute new Euler angles using the SAS task newsiam.

4.5.4 COLORTRANS

4.5.4.1 Description

The COLORTRANS CCF component is used by the SAS.
COLORTRANS contains one extension named `COLORMAG`.

It lists the coefficients defining the colour transformations, the zero points for the different filters and the aperture radii used in the photometric calibration analysis.

In the COLORMAG extension the instrument zero points of non-dispersive filter elements are stored as header keywords as follows: the zero point of the filter FilterID is stored in the keyword ZPT*FilterID*, e.g. the zero point of the the U-filter is stored in ZPTU.

The AB system zero point of the filter FilterID is stored in the keyword ABM0*FilterID*, e.g. the AB zero point of the the U-filter is stored in ABM0U.

The flux conversion factor derived from White Dwarfs of the filter FilterID is stored in the keyword FCF*FilterID*.

The flux conversion factor in the AB system of the filter FilterID is stored in the keyword ABF0*FilterID*.

The aperture radii applied to extract the source counts in the photometric calibration analysis are listed for the different filter elements in the keywords APE*FilterId* (e.g. APEU for the U-filter). The listed radii define the aperture widths of the different filter wheel elements for which the zero points are valid and for which the colour transformations are applicable.

The coefficients of the colour transformation are stored in the binary table of the COLORMAG extension.
The columns FILTERID1 and FILTERID2 identify the type of colour used in the transformation. FILTERID1 holds the identifier of the shorter wavelength compared with FILTERID2, e.g. FILTERID1=B and FILTERID2=V. The sequence of filters sorted from short to long wavelength is: UVW2, UVM2, UVW1, U, B, V.
The validity range of a specific colour transformation are stored in the column TRAFOLIMIT. The lower limit is stored as the first and the upper limit as the second element in the TRAFOLIMIT column. Note that the validity ranges of the colour transformations must not overlap, i.e. $(b-v)_1 < (b-v)_2 < ... < (b-v)_n$
Different colour transformations can be defined in the same colour interval. The different branches of the colour transformation are identified using an integer number which is stored in the column BRANCH. The application of the different transformations is explained in the CCF release note (i.e. which value of `BRANCH` corresponds to which metallicity). However generally the value 0 in the BRANCH column is reserved for the colour transformation of the main sequence stars.
The coefficients to convert the colour index (mag1-mag2) into the colour index of the standard system (MAG1-MAG2) are stored in the column TRAFOP1. The coefficients to calculate the standard magnitude MAG2 from the colour index (mag1-mag2) and the brightness (mag2) are

stored in the column TRAFOP2. The uncertainties of the coefficients are stored in the columns TRAFOP1E and TRAFOP2E respectively.
Currently the colour transformations are described with a quadratic functions of the colour index. Therefore only three parameters per column are used. The calibration file can keep up to 10 parameters and the not used coefficients are set to zero.

The binary table COLORMAG has the following format:

Binary table: COLORMAG

keyword name	keyword type	keyword unit	description
ZPTU	E	magnitude	zero point of U-filter
ZPTB	E	magnitude	zero point of B-filter
ZPTV	E	magnitude	zero point of V-filter
ZPTUVW1	E	magnitude	zero point of UVW1-filter
ZPTUVM2	E	magnitude	zero point of UVM2-filter
ZPTUVW2	E	magnitude	zero point of UVW2-filter
ZPTMAGNI	E	magnitude	zero point of MAGNI-filter
ZPTWHITE	E	magnitude	zero point of WHITE-filter
ABM0U	E	magnitude	AB zero point of U-filter
ABM0B	E	magnitude	AB zero point of B-filter
ABM0V	E	magnitude	AB zero point of V-filter
ABM0UVW1	E	magnitude	AB zero point of UVW1-filter
ABM0UVM2	E	magnitude	AB zero point of UVM2-filter
ABM0UVW2	E	magnitude	AB zero point of UVW2-filter
ABF0U	E	erg/cm^2/A/count	AB Flux conversion factor of U
ABF0B	E	erg/cm^2/A/count	AB Flux conversion factor of B
ABF0V	E	erg/cm^2/A/count	AB Flux conversion factor of V
ABF0UVW1	E	erg/cm^2/A/count	AB Flux conversion factor of UVW1
ABF0UVM2	E	erg/cm^2/A/count	AB Flux conversion factor of UVM2
ABF0UVW2	E	erg/cm^2/A/count	AB Flux conversion factor of UVW2
FCFU	E	erg/cm^2/A/count	Flux conversion factor of U
FCFB	E	erg/cm^2/A/count	Flux conversion factor of B
FCFV	E	erg/cm^2/A/count	Flux conversion factor of V
FCFUVW1	E	erg/cm^2/A/count	Flux conversion factor of UVW1
FCFUVM2	E	erg/cm^2/A/count	Flux conversion factor of UVM2
FCFUVW2	E	erg/cm^2/A/count	Flux conversion factor of UVW2
APEU	E	pixel	$r_{aperture}$ of photometric calib. U-filter
APEB	E	pixel	$r_{aperture}$ of photometric calib. B-filter
APEV	E	pixel	$r_{aperture}$ of photometric calib. V-filter
APEUVW1	E	pixel	$r_{aperture}$ of photometric calib. UVW1-filter
APEUVM2	E	pixel	$r_{aperture}$ of photometric calib. UVM2-filter
APEUVW2	E	pixel	$r_{aperture}$ of photometric calib. UVW2-filter
APEMAGNI	E	pixel	$r_{aperture}$ of photometric calib. MAGNI-filter
APEWHITE	E	pixel	$r_{aperture}$ of photometric calib. WHITE-filter
ALGOID	I	n.a.	parameter to select type of colour transformation

column name	column type	column unit	comment
FILTERID1	9A	n.a.	short wavelength filter id
FILTERID2	9A	n.a.	long wavelength filter id
TRAFOLIMIT	2E	magnitude	upper/lower limit of validity range of colour transformation
TRAFOP1	10E	n.a.	coefficients of colour transformation (mag1-mag2) into (MAG1-MAG2)
TRAFOP1E	10E	n.a.	uncertainties of parameters in TRAFOP1
TRAFOP2	10E	n.a.	coefficients of magnitude transformation (mag1-mag2) into MAG2
TRAFOP2E	10E	n.a.	uncertainties of parameters in TRAFOP2
BRANCH	I	n.a.	identifier of branch of colour transformation

Each record holds the transformation coefficients related to one filter combination of a certain colour interval and one branch, e.g. the record with FILTERID1='U', FILTERID2='B', TRAFOLIMIT=[0.0, 0.5[and BRANCH=0 holds the coefficients of the colour equations to transform main sequence stars with an instrumental u-b colour between 0.0 mag and 0.5 mag into the standard photometric (U-B) colour and B-magnitude. (The transformation into filter magnitudes generally has a higher uncertainty than the transforming of colours).

The binary table is designed to allow colour transformations for any filter combinations. However it is intended to use colour tranformations only to correct the magnitudes measured in the optical filters, i.e. U, B, V, although the colours between and UV- and optical measurement may be used for this purpose, e.g. UVW2-B colour to correct a B-magnitude.

4.5.4.2 Generation from Ground Calibration Data

During the ground calibration campaign no absolute photometric measurements were made at CSL. Therefore the CCF component was initially filled based on simulated data by SCISIM. A detailed description can be found in XMM-PS-TN-33 and only a short summary is given here.

OM count rates of different spectral types in different passbands were simulated with OSIM. The OM count rates are converted into OM magnitudes applying an zero point correction of 18.5 mag. The expected standard magnitudes of the various spectral types are taken from Zombeck. The pre-launch colour transformation coefficients and the associated errors are calculated using the methode described by Harris et al. (PASP 1981, **93** 507). Harris' recommendation is followed and the colour transformation is defined as quadratic function of the colour terms. The equations are solved by a multilinear regression.

4.5.4.3 Generation from In-Orbit Data

The transformations from the OM instrumental system to the standard Johnson's system were established based on in-flight observations. Several fields have been observed from the ground with the standard Johnson UBV filters and with the XMM-Newton OM. The fitting was limited to stars with less than 20% coincidence loss, and 5% statistical errors. 363 cross-identified stars have been used to make the color transformation.

The Color-transformations for UV filters are based on the simulations because we have not got enough calibration observations for UV filters. A calibration field with many blue stars (B-V < 0.4) will be observed, then we will update the UV color transformation based on in-flight observations.

We have derived the flux conversion factors from OM observations of seven spectrophotometric standard stars (white dwarfs: Hz4, GD50, GD153, LBB227, BPM16274, Hz2 and G93-48). For each filter, if you multiply the countrates (cts/s) from SAS by these conversion factors, you will get the flux (erg/cm^2/s/A).

The AB system for OM has been derived from simulations based in the results obtained from the same white dwarfs.

These flux conversion factors have been added as keywords in the COLORMAG extension of the CCF as defined above.

A series of calibration observations have been requested to refine the conversion factors used in the count rate to flux calculations according to spectral type. These observations will provide data for a longer term strategy to introduce colour correction terms that can be applied to fluxes calculated from count rates received by two or more filters.

4.5.5 GRISMCAL

4.5.5.1 Description

This CCF contains the wavelength and flux calibrations for the two OM grisms, GRISM1 (or UV grism) and GRISM2 (or Visible grism). It is used by the SAS.

The wavelength calibration is provided in two tables, one per grism, containing the value of the wavelength corresponding to each pixel in the extracted spectrum, counted from the centroid of the spectrum zero order in the undistorted and rotated image.

The flux calibration is also provided in two tables, one per grism. They define the Inverse Sensitivity Function (ISF) in three columns containing wavelength, value of the ISF and its percentage error. The ISF's are defined in the intervals 1800 - 3600 A for GRISM1, and 3000 - 6000 A for GRISM2. The sampling of the ISF's is 40 A.

When applying the flux calibration, the ISF has to be interpolated for each wavelength in the extracted net spectrum, within the calibrated range. Points outside this range are zeroed.

A header keyword ALGOID has been introduced to prevent the usage of the CCF Version 4 (the latest one) with SAS versions earlier than 7.

The format of the Grism calibration CCF component is described in detail below.

Binary table: FLUX_GRISM1

column name	column type	column unit	comment
WAVELENGTH	E	Angstrom	
ISF	E	erg/cm2/count	Inverse Sensitivity Function
ISF_error	E	%	Error on ISF

Binary table: FLUX_GRISM2

column name	column type	column unit	comment
WAVELENGTH	E	Angstrom	
ISF	E	erg/cm2/count	Inverse Sensitivity Function
ISF_error	E	%	Error on ISF

Binary table: WAVELENGTH_GRISM1

column name	column type	column unit	comment
WAVELENGTH	E	Angstrom	

Binary table: WAVELENGTH_GRISM2

column name	column type	column unit	comment
WAVELENGTH	E	Angstrom	

4.5.5.2 Generation from Ground Calibration Data

No calibration was performed on the ground for any of the grisms. This CCF has been defined and produced from in-orbit data.

4.5.5.3 Generation from In-Orbit Data

4.5.5.3.1 Wavelength calibration

The calibration is based in the identification of spectral features in the observed spectra of F-type stars for the UV grism and BPM 16274 for the Visible one.

A polynomial (degree 2 for UV and 1 for Visible) is used to fit the laboratory wavelengths as a function of the pixel number counted from the centroid of the zero order. The CCF table is obtained by applying the polynomial to a monotonic array representing the pixels array in the extracted spectrum.

4.5.5.3.2 Flux calibration

Spectroscopic Standard stars, mainly white dwarfs, were observed with both OM grisms. The ISF is obtained by dividing the extracted spectrum, in count rate versus wavelength, by the standard flux of the star obtained from HST Calspec database. The results obtained from several stars are then averaged and the standard deviation is taken as the error in the ISF.

Due to the presence on spectral lines, mainly in the Visible grism, the spectra of the standard stars are interpolated and re-sampled prior to the division.

The first ISF's have been obtained from several observations with OM of the selected stars. Two of them have been included in the XMM-Newton Routine Calibration Plan, to be observed periodically to study variations with time which will be incorporated into the flux calibration.

4.5.6 HKPARMINT

4.5.6.1 Description

This CCF component is used by the SAS.

Binary table: HKParmInt

keyword	keyword type	keyword unit	keyword value
HKD_NAME	XA	not applicable	
LIMIT_LOW	E	-	varies
LIMIT_HIGH	E	-	varies
UNIT	20A	not applicable	varies

4.5.6.2 Generation from Ground Calibration Data

The data names and the associated limits are taken from the TM database.

4.5.6.3 Generation from In-Orbit Data

An update of the limits will emerge from the analysis of the instrument performance as function HKD values.

4.5.7 LARGESCALESENS

The analysis of calibration observations of fields observed in different positions of the detector has demonstrated that large sensitivity variations do not exist in OM data, or they are smaller than the errors of the photometry (a few percent). Therefore this calibration file has been set to unity.

Something similar occurs with pixel to pixel variations (see 4.5.9).

We keep the following description as a guideline for eventual needs that might arise in the future.

4.5.7.1 Description

This CCF component is used by the SAS. The LargeScaleSens file describes the large scale response uniformity of the combination filter and detector. The large scale flat field correction is applied to an image by multiplication, thus the inverse normalized sensitivity is stored in the CCF component.

The LargeScaleSens file maps the relative sensitivity variations over longer spatial scale for the combination of a filter element and the detector. The FOV is mapped at a coarser grid. An interpolation of the large-scale map is required in order to obtain a map at a finer grid.

The LargeScaleSens file consists of 12 image extensions. Generally there is one extension associated with each filter element, except for the grism where there is one extension for the zero and first order respectively. In total there are 12 image extension contained in the LargeScaleSens file.

FILTER-U	large scale response of U-filter
FILTER-B	large scale response of B-filter
FILTER-V	large scale response of B-filter
FILTER-UVW1	large scale response of UVW1-filter
FILTER-UVM2	large scale response of UVM2-filter
FILTER-UVW2	large scale response of UVW2-filter
FILTER-WHITE	large scale response of WHITE-filter
FILTER-MAGNI	large scale response of MAGNI-filter
FILTER-GRISM1	large scale response of GRISM1 1st order
FILTER-GRISM2	large scale response of GRISM2 1st order
FILTER-GRISM10	large scale response of GRISM1 0th order
FILTER-GRISM20	large scale response of GRISM2 0th order

The format of each extensions is as listed below:

Image: FILTER-*FilterId*

keyword	keyword type	keyword unit	keyword value	comment
BITPIX		not applicable	-32	
NAXIS	I	not applicable	2	
NAXIS1	I	not applicable	may vary	
NAXIS2	I	not applicable	may vary	
CTYPE1	S	not applicable	rawx	
CRVAL1	E	pixel	may vary	
CDELT1	E	pixel	may vary	
CTYPE2	S	not applicable	rawy	
CRVAL2	E	pixel	may vary	
CDELT2	E	pixel	may vary	
AVGERAGE	E	not applicable	may vary	avg. flatfield value of good pixel
DAVGERAGE	E	not applicable	may vary	1σ uncertainty of AVERAGE
PIXERR	E	%	may vary	avg. % of 1σ uncerainty of a single pixel value

4.5.7.2 Generation from Ground Calibration Data

No flat flatfield data at system level are available from ground calibration. Therefore it was decided to set the LargeScaleSens file uniformly to one.

4.5.7.3 Generation from In-Orbit Data

The basic idea is to obtain a supersky flatfield by combining a large number of relatively object-free exposures of random fields. A similar approach has been used to obtain WFPC superflats (Ratanunga et al. 1994, APJ, 108, 2362). Most suitable datasets are data from default configurations, because the entire detector area is covered. The detailed analysis steps are

- fpn correction
- source identification
- cut out sources
- normalize images by average counts per pixel and stack images
- add stack of images
  - weighting with square root of avg. counts per pixel
  - take into account number of exposure per pixel
- fit bicubic spline surface to obtain new large scale flatfield

The LargeScaleSens files can locally be validated using celestial objects in calibration offset pointings.

4.5.8 PHOTTONAT

4.5.8.1 Description

This file is used by QLA and the SAS. A copy of this CCF component is also used by the PHS tools.

`PhotToNat` contains three tables, viz. `PHOTTONAT` `COINCIDENCE` and `DEGRADATION`.

The `PHOTTONAT` table contains coefficients used to calculate the MIC CCD framerate for a given detector configuration (stored in the binary field PARAM00). In the extension header the fraction of events above the detection threshold is stored in the keyword PHDFRAC together with the associated event detection threshold (keyword: THRESH).
The value of PHDFRAC can be used to correct the detected counts for the global loss of detector efficiency (ageing). The pair of values THRESH and PHDFRAC are the settings at the time the OM photometric calibration was performed (cf. CCF file ColorTrans, section 4.5.4

The parameters characterising the clocking of the OM CCD are stored in the 10 element field PARAM00 of the extension PHOTONAT. The sequence witin PARAM00 and the field values are

description	value	unit	variable in SAS routine frametime_mod
total horizontal clocks for one row	407	H rows	NHP
total vertical clocks for a frame transfer	288	V rows	NVPS
time required for a shift by one pixel			
– in vertical direction	6.E-7	sec	VTRANS
– in horizontal direction	1.E-7	sec	HTRANS
number of rows in image area	290	V rows	NVPI
vertical to horizontal switch time	2.E-7	sec	V2H*1.E-9
horizontal to vertical switch time	1.E-7	sec	H2V*1.E-9
start of logical CCD window in hardware CCD	15	row	ROFFSET
the remaining 2 parameters are not used at the moment	0.	n.a.	

The logical CCD window of the prime channel has its starting address at (52,27), while the redundant is at (64,15).

The format of the binary table in the PHOTTONAT extension follows the description of the CCF ICD [5]. The following two additional keywords are present in the extension header:

keyword	keyword type	keyword unit	keyword value
PHDFRAC	E	not applicable	may vary
THRESH	I	ADU	may vary

The table `COINCIDENCE` holds the coefficients used to parameterize the empirical coincidence loss curve. There is no filter dependence of the parameters. The binary table has the following format:

Binary table: COINCIDENCE

column name	column type	column unit	comment
PLINFUNC	xE	n.a.	set of coefficients for linearity curve parameterization

The table `DEGRADATION` contains the coefficients of the time dependent sensitivity degradation correction. Since sensitivity degradation is due in part to the photocathode, it is wavelength dependent and therefore there is one set of coefficients for each one of the OM lenticular filters.

The current correction is a linear function of the MJD of observation. However the table has place for a future cuadratic function. The result of the function is a correction factor to be multiplied by the coincidence loss corrected counts. The binary table has the following format:

Binary table: DEGRADATION

column name	column type	column unit	comment
FilterID (U B V UVW1 UVM2 UVW2)	xE	n.a.	set of coefficients for correction function

4.5.8.2 Generation from Ground Calibration Data

The keyword THRESH is the setting of the event detection threshold used by the BPE to recognize events. THRESH describes the threshold for which the fraction of detected events PHDFRAC was calculated. The fraction of detected events is derived from the OM engineering data subtype 6 (Intensifier characteristics).
Given a event detection threshold THRESH the fraction of detected events is approximated using e.g. the IDL functions gaussfit and gauss_pdf:

```
pro phd_ana, name,ll_fit,ul_fit,i_thresh,PHDFRAC
 ; INPUT:
 ; name     string name of datafile
 ; ll_fit   integer lower limit of fit
 ; ul_fit   integer upper limit of fit
 ; i_thresh integer event detection threshold
 ; OUTPUT:
```

```
 ; PHDFRAC  real   fraction of detected events

 ; read data
 rdtable,name,x,y

 ; gaussian fit
 yfit=gaussfit(x(ll_fit:ul_fit),y(ll_fit:ul_fit),a1,nterm=3)

 ; calculate fraction of detected events
 PHDFRAC= 1.-gauss_pdf((i_thresh-a1(1))/a1(2))

return
end
```

The CCD parameters stored in the field PARAM00 were already derived from laboratory measurements at MSSL and are listed above. No further update is intended.

4.5.8.3 Generation from In-Orbit Data

The degradation correction coefficients have been calculated by a linear fitting to the measured counts (coincidence loss corrected) of the stars BPM 16274, Hz 2 and GD 153 observed periodically since the beginning of the mission. The fit is done for each filter as a function of the MJD of observation, using the data of all stars. These stars are observed once per year (BPM 16274 twice) to monitor the variation of sensitivity.

4.5.9 PIXTOPIXSENS

The analysis of calibration observations of targets and fields also observed from the ground has shown us that pixel to pixel sensitivity variations (flat field correction) if they exist are small, of the same order than the errors of the photometry (a few percent). Therefore PixToPixSens calibration file has been set to unity.

A similar conclusion was pointed out for LargeScaleSens (see 4.5.7)

We keep the following description as a guideline for eventual needs that might arise in the future.

4.5.9.1 Description

This CCF component is used by the SAS. The OM flatfield response is decomposed into two components, spatial large scale and small scale sensitivity variations.

The PixToPixSens file describes the spatially high frequency throughput and sensitivity non-uniformities. In addition PixToPixSens describes the illumination pattern of the internal calibration lamps seen by the detector.
The high frequency sensitivity variations are introduced by the detector. Non-uniformities in the filter transmissions won't show up in small scale structures, as the filter wheel elements are out of focus.

The PixToPixSens file describes the inverse detector sensitivity normalized to the average sensitivity. Values less than one indicate pixels with a sensitivity higher than the average and values greater than one indicate pixels with a lower sensitivity than the average.

PixToPixSens consists of two image extension. The Image extension has a size of 2048x2048 pixel and is at full detector resolution. The extension name is PIXTOPIXSENS. The keyword CFRR specifies the average count rate (units: count/CCD frame) in the flatfield exposures used to generate the image extension.

The image extension LEDTEMPLATE holds the description of the LED illumination pattern. The illumination pattern is described as the inverse illumination level by a full resolution map (2048x2048 subpixel). Values less than one indicate pixels with a stronger illumination by the LEDs (higher than the average) and values greater than one indicate pixels with a lower illumination than the average.

The image extension in the pixtopixsens CCF component has the following structure PIXTOPIXSENS.
Image extension: PIXTOPIXSENS

keyword name	keyword type	keyword unit	keyword value	comment
NAXIS	Integer	n.a.	2	number of data axes
NAXIS1	Integer	n.a.	2048	length of data axis 1
NAXIS2	Integer	n.a.	2048	length of data axis 2
CFRR	Real	cts/fullframe/sec	varies	average CFRR
AVERAGE	Real	n.a.	varies	average sensitivity of good pixels
DAVERAGE	Real	n.a.	varies	standard deviation on AVERAGE
PIXERR	Real	percent	varies	average standard deviation on single pixel value

The second image binary extension in the pixtopixsens CCF component is LEDTEMPLATE with the structure as follows.

Image extension: LEDTEMPLATE

keyword name	keyword type	keyword unit	keyword value	comment
NAXIS	Integer	n.a.	2	number of data axes
NAXIS1	Integer	n.a.	2048	length of data axis 1
NAXIS2	Integer	n.a.	2048	length of data axis 2
AVERAGE	Real	n.a.	varies	avg. correction for nonflat illumination
DAVERAGE	Real	n.a.	varies	standard deviation on AVERAGE
PIXERR	Real	percent	varies	average single pixel deviation

4.5.9.2 Generation from Ground Calibration Data

The PixToPixSens fill is generated from internal calibration exposures producing fullframe, high resolution engineering mode data (DD_ENG subtype 4). However so far no representative flatfield data have been acquired and the initial image extension is set to unity.

4.5.9.3 Generation from In-Orbit Data

Ideally the in-orbit PixToPixSens file will be generated from internal calibration measurements using the fullframe high resolution engineering mode. However due to their large data volume the fullframe high resolution data won't be available frequently enough to base the in-flight generation of the PixToPixSens file on the high resolution data. Therefore a different strategy has to be adopted.

The SAS task *omflatgen* will be used to compute the PixToPixSens files from low resolution internal calibration exposures (tracking off, BPE on), which are predominately acquired during slews. The analysis can be outlined as

1. Multiple exposures may be co-added to achieve high enough statistics.
2. removal/flagging of bad pixels
3. calculation of average CCD frametime in the used exposures
4. calculation of average CFRR,
   i.e. the average counts per pixel per CCD-frame
5. fpn?
6. check dark counts and exclude bad pixel?
7. calculation of the local normalization mask(?)/value for each 8x8area/pixel
   e.g. by calculating the average within a sliding box of TBD width.
8. local image normalization
   by dividing the pixel content by the local average value
9. eventual exclusion of large excursion regions and recalculation of step 4 and 5
10. global normalization
    e.g. by division with the average of the local normalization values or by normalization to values of a reference area/pixel.

If the non uniformity of the calibration lamp illumination is known with sufficient accuracy, the inverse of the illumination pattern can be be used for uniformity correction and the normalization steps can be omitted in the data analysis. The non-uniformity of the calibration lamp illumination can be derived from a comparison of the smoothed high resolution flatfield with the large scale flatfield.

In case not sufficient high resolution flatfield data are available the low resolution data must be expanded into high resolution PixToPixSens data.
This can be achieved in three different ways, depending on the availability of data.

- If high resolution data are available and provide sufficient statistics, these data are used to calculate the PixToPixSens file. A pixel expansion is not required.
- If high resolution data are available, but the data do not provide sufficient statistics or they are out of date, each pixel of the low resolution data is expanded into four subpixels by applying a weighting factor to each quadrant of the old pixel. The weighting factor is derived from full resolution images.
- If no high resolution data are available or for the initial calibration procedure low resolution data are expanded into full resolution data applying a weighting factor of 1.0 to each quadrant of the old pixel.

4.5.10 PSF1DRB

4.5.10.1 Description

This CCF component is used by the SAS. The radial PSF distribution of the filter FilterId is tabulated in the extension PSF-*FilterId*. Azimuthal symmetry is assumed.
There is one extension per non-dispersive filter element and two extensions for each grisms. Thus in total there are 12 other binary table extension in the PSF1DRB CCF component, with the names:

PSF1DRB extension name	comment
PSF-U	radial PSF of U-filter
PSF-B	radial PSF of B-filter
PSF-V	radial PSF of V-filter
PSF-UVW1	radial PSF of UVW1-filter
PSF-UVM2	radial PSF of UVM2-filter
PSF-UVW2	radial PSF of UVW2-filter
PSF-WHITE	radial PSF of WHTIE-filter
PSF-MAGNI	radial PSF of magnifier
PSF-GRISM1	radial PSF of UV grism 1st order (TBC)
PSF-GRISM2	radial PSF of visible grism 1st order (TBC)
PSF-GRISM10	radial PSF of UV grism 0th order (TBC)
PSF-GRISM20	radial PSF of visible grism 0th order (TBC)

In the photometric calibration analysis a fixed aperture for each filter is used to extract the source counts. The value of the cumulative PSF is normalized to 1.0 at this fixed aperture radius. The aperture radius is listed for each filter elements in the keyword APERTURE. The integrated flux within the radius APERTURE. is defined to contain 100% of the source flux. The aperture radii in the analysis of UV data are generally larger than in the visible. Therefore the aperture radius is listed separately for each filter in the keyword `APERTURE` of the filter extensions.

The stretching of the PSF as function of field position is described as function of the distance r between the detector location (x,y) and the boresight, i.e. as function of

$$r = \sqrt{(x - 1024.5)^2 + (y - 1024.5)^2}$$

. The functional form is a third order polynomial in r (cf. section 3.5.9). The parameters of the third order polynomial are stored as keywords in the extension headers.

In the binary extensions the radial PSF at the central detector position of the specified filter is stored as integrated intensity (column INTENSITY) between the radial bins RMIN and RMAX. The PSF is stored for different count to framerate ratios (column CFRR).

The binary extensions have the follow structure:

Binary table: PSF-*FilterId*

keyword name	keyword type	keyword unit	keyword value	comment comment
APERTURE	E	pixel	varies	aperture radius of photometric calibration
PARX0Y0	E	not applicable	varies	
PARX1Y0	E	not applicable	varies	
PARX2Y0	E	not applicable	varies	
PARX3Y0	E	not applicable	varies	
PARX0Y1	E	not applicable	varies	
PARX1Y1	E	not applicable	varies	
PARX2Y1	E	not applicable	varies	
PARX0Y2	E	not applicable	varies	
PARX1Y2	E	not applicable	varies	
PARX0Y3	E	not applicable	varies	

column name	column type	column unit	comment
CFRR	E	counts/frame/resolution element	count to framerate ratio
RMIN	20E	subpixel	lower radius limit of bin
RMAX	20E	subpixel	upper radius limit of bin
INTENSITY	20E	not applicable	intensity in [RMIN,RMAX]

4.5.10.2 Generation from Ground Calibration Data

The radial distributions were generated from the CSL pinhole images. The accuracy of the derived distributions is hampered by the uneven illumination of the detector and the change in PSF width originating in the CSL collimator.
The analysis was performed in the following steps.

- localizing sources with daofind (noao.digiphot.daophot)
- generate radial profile with pradprof (cl.plot)
- normalize radial profiles
- average radial profile in central detector area

4.5.10.3 Generation from In-Orbit Data

A direct measurement of the curve of growth of the PSF is limited by the small number of appropriate, isolated stars and the large scatter caused by coincidence loss and straylight. Thus we have used Daophot for our PSF analysis. This allows us to fit the same function to all the stars in a field of view, thus giving a set of good average PSFs for a range of count to framerate ratios(CFRR). This CCF has been achieved from in-flight data for all filters and is already incorporated into the SAS.

4.5.11 DARKFRAME

4.5.11.1 Description

This CCF component is not used by the SAS. The format of the image extension follows the description of the CCF ICD [5].

The following additional keywords are present in the Image extension DARKFRAME:
Additional Keywords in Image: DARKFRAME

column name	column type	column unit	comment
PIXERR	E	%	average percentage of the 1σ uncertainty on a single pixel dark count rate
AVERAGE	E	1/s/pixel	dark count rate averaged over entire detector
DAVERAGE	E	1/s/pixel	1σ uncertainty of keyword AVERAGE

4.5.11.2 Generation from Ground Calibration Data

No darkframe image was taken with the FM detector. The available darkframe images of the FM intensifier do not provide any absolute position. Therefore a uniform darkframe has to be assumed with the dark count rate set to the average dark current measured for the specific intensifier tube.

4.5.11.3 Generation from In-Orbit Data

Darkframes are calculated from flatfield data with the filter wheel in blocked position and the LEDs switched off. Darkframe images of several exposures are co-added to achieve good statistics. The co-added image is divided through the sum of the exposure durations to obtain the dark count rate. The average statistical uncertainty of each pixel content and the average dark count rate together with its uncertainty are stored as keywords.

4.5.12 QUICKMAG

4.5.12.1 Description

This CCF component is used by QLA. A copy of this CCF component is also used by the PHS tools.
There is one binary extension per filter element. The formats of the binary tables follow the description of the CCF ICD [5].

4.5.12.2 Generation from Ground Calibration Data

Data are filled with SCISIM. The file lists the photon rate as function of brightness [mag] for different spectral types.

4.5.12.3 Generation from In-Orbit Data

Data are filled with SCISIM after adjustment of SCISIM to in-orbit performance.

4.5.13 DIFFUSEGALA

4.5.13.1 Description

A copy of this CCF component is used by the PHS tools.

The DiffuseGala file consists of three extensions:

DGL_MAP The mapping of the diffuse galactic emission in galactic coordinates. The map is in S10 units. When reading the Image extension DGL_MAP, the keywords CRVALn, CRPIXn and CDELTn have to be interpreted.

DGL_SPEC The assumed spectrum of the DGL.

DGL_COUNTS The expected count rates (cts/s/subpixel) in the different filter elements.

The format of the DiffuseGala CCF component is described in detail below.

Image: DGL_MAP

keyword name	keyword type	keyword unit	keyword value	comment
BITPIX		not applicable	-32	
NAXIS	I	not applicable	2	
NAXIS1	I	not applicable	varies	
NAXIS2	I	not applicable	varies	
CTYPE1	S	not applicable	gal. longitude	
CRVAL1	E	degree	varies	reference longitude
CDELT1	E	degree	varies	increment of longitude
CRPIX1	I	not applicable	varies	address of reference pixel
CTYPE2	S	not applicable	gal. latitude	
CRVAL2	E	degree		reference latitude
CDELT2	E	degree		increment of latitude
CRPIX2	I	not applicable	varies	address of reference pixel
BUNIT	S	S10(B)	may vary	Units of image

Binary table: DGL_SPEC

keyword	keyword type	keyword unit	keyword value
REF_INTENS	E	S10(B)	may vary
column name	column type	column unit	comment
LAMBDA	E	nm	wavelength
FLUX	E	$W/m^2/nm/deg^2$	DGL flux for reference intensity

Binary table: DGL_COUNTS

keyword	keyword type	keyword unit	keyword value
REF_INTENS	E	S10(B)	may vary
column name	column type	column unit	comment
FILTER_ID	9A	not applicable	filter ID
COUNTS	E	cts/sec/subpixel	expected flux per pixel at the reference intensity

4.5.13.2 Generation from Ground Calibration Data

A spatial and spectral information of the diffuse galactic light emission is taken from literature:

- Lillie and Witt (1976).
- Gondhalekar and Wilson, A&A,1975,38,329
- Mattila, A&A,1980,39,53
- Bahcall & Soneira ApJS, 1984, 55, 67
- Leinert et al. (A&AS, 1998, 127, 1)

The spatial distribution of the DGL is contained in the DGL_MAP extension. The galactic coordinate system is used.
The spectrum is compiled from literature and written into the extension DGL_SPEC. The intensity of the spectrum is scaled to the reference intensity. The photon rate are estimated by folding the DGL spectrum with OSIM. The binary table DGL_CTS lists the expected photon rates for the different filters at the reference intensity.

4.5.13.3 Generation from In-Orbit Data

It is not expected to update these CCF files on the short term, neither is there a dedicated calibration activity. However on the longterm it will be possible to estimate the diffuse background components (DGL and Zodiacal light). Background levels of individual science observations can be stored as function of galactic longitude and latitude (l,b), the helioecliptic coordinates $\lambda - \lambda_{\odot}, \beta$, and filter passband. The two background components can be disentangled in this 5-dimensional data-space.

4.5.14 ZODIACAL

4.5.14.1 Description

A copy of this CCF component is used by the PHS tools.

4.5.14.2 Generation from Ground Calibration Data

A spatial and spectral information of the zodiacal light emission is taken from literature:

- Levasseur-Regourd & Dumont (A&A, 1980, 84, 277)
- Leinert et al. (A&AS, 1998, 127, 1)

The spatial distribution of the zodiacal light is contained in the ZODIACAL_MAP extension. Helioecliptic coordinates are used.
The zodiacal light spectrum (essentially solar) is compiled from literature and written into the extension ZODIACAL_SPEC. The intensity of the spectrum is scaled to the reference intensity. The photon rates are estimated by folding the zodiacal light spectrum with the OM response. The binary table ZODIACAL_CTS lists the expected photon rates for the different filters at the reference intensity.

Image extension: ZODICAL_MAP

keyword name	keyword type	keyword value	comment
BITPIX	integer	-32	number of bits per data pixel
NAXIS	integer	2	number of data axes
NAXIS1	integer	may vary	length of data axis 1
NAXIS2	integer	may vary	length of data axis 2
BUNIT	string	S10(V)	Units of the image
CTYPE1	string	'ecliptic longitude / [deg]'	name of x-axis
CTYPE2	string	'ecliptic latitude / [deg]'	name of y-axis
CRVAL1	real	may vary	[degree] x-coordinate of reference bin
CRVAL2	real	may vary	[degree] y-coordinate of reference bin
CRPIX1	integer	1	[bin] address of reference bin x-direction
CRPIX2	integer	1	[bin] address of reference bin y-direction
CDELT1	real	may vary	[degree] increment in x-direction
CDELT2	real	may vary	[degree] increment in y-direction

Binary table: ZODICAL_SPEC

keyword name	keyword type	keyword value	comment
REFINTEN	real	may vary	reference intensity of spectrum [S10(V)]

column name	column type	column unit	comment
LAMBDA	E	nm	wavelength of bin
FLUX	E	$W/m^2/nm/deg^2$	zodiacal light flux at reference intensity

Binary table: ZODICAL_COUNTS

keyword name	keyword type	keyword value	comment
REFINTEN	real	may vary	reference intensity of count rates [S10(V)]

column name	column type	column unit	comment
FILTER_ID	9A	-	filter identifier
COUNTS	E	photons/sec/subpixel	photon rate at reference intensity

4.5.14.3 Generation from In-Orbit Data

It is not expected to update these CCF files on the short term, neither is there a dedicated calibration activity. However on the longterm it will be possible to estimate the diffuse background components, namely DGL and zodiacal light (cf. section 4.5.13).

5 Applications of CCF — Cross Reference Tables

5.1 Cross References to In-Flight Calibration Targets

Target Name	**Primary Calibration Goal**	**Secondary Goal**	**CCF Files Verified**
Hyades	PSF	Alignment	Astromet, Boresight, LinCoord PSFpileup, XEncirEn, XPSF
MS0419.3+1943 MS0737.9+7441 MS0922.9+7459 S5 0716+71	Effective area near Au edge	Contamination	FilTransX, QuantumEf, XAreaEf
A 496	Vignetting function		XAreaEF
LMC 30 Dor	Vignetting function	Plate scale	LinCoord, PSFpileup, XAreaEf
NGC 6475	PSF	Plate scale	PSFpileup, XEncirEn, XPSF
X LMC X-2	X-ray straylight		XSLAreaEf, XSLDist
α Cen β Cen	Visible straylight		FilterTransV, VPSF, VSLAreaEf

Table 2: Cross reference of mirror CCF components to In-Flight Calibration Plan.

Target Name	Primary Calibration Goal	Secondary Goal	CCF Files Verified
Internal Source	Gain and CTI	Contamination	ADUConv, Redist CalSourceData Contam, CTI EventSizeDist
NGC2516	EPIC Boresight trend analysis	PSF and focal plane scale	LinCoord XPSF
LMC X-3	PN PSF vs pile-up	MOS PSF wings	XPSF,PSFPileUp RedistPileUp
PKS0312-770	MOS PSF	PN PSF	XPSF
Terzan 2	High energy effective area	Co-ordination with SAX	QuantumEf XAreaEf
3C58	Effective area vs energy	Y axis vignetting	QuantumEf XAreaEf
G21.5-09	Effective area vs energy	Z axis vignetting	QuantumEf XAreaEf
N132D	Gain	CTI	ADUConv, CTI, Redist
1ES0102-72	Gain	CTI & AXAF cross check	ADUConv, CTI, Redist XAreaEf
HZ43 CAL83 CAL87	C band filter transm. O band filter transm. M band filter transm.	Contamin. Contamin. Contamin.	,FilterTransX,Contam,XAreaEf FilterTransX,Contam, XAreaEf FilterTransX,Contam,XAreaEf
PG0136 RXJ1856 PG1658	Contamination Contamination Contamination	Gain Gain Gain	Contam, ADOConv,XAreaEf Contam, ADOConv,XAreaEf Contam, ADOConv,XAreaEf
PG1634 Mkn205 PKS0558	Pile-UP Pile-UP Pile-UP	Mode dependence Mode dependence Mode dependence	PSFPileUp,RedistPileUp PSFPileUp,RedistPileUp PSFPileUp,RedistPileUp
Crab Nebula	X-ray Stray light	Fast timing	XSLAreaEf,XSLDist TimeCorr
PSR0540 EXO0748-676	Timing Absolute timing		TimeCorr TimeCorr
Coma		Flat Field	XAreaEf,QuantumEf
α Pic	Vignetting		XAreaEf,VAreaEf
Various stars	Filter stray light		VAreaEf, FilterTransV
All			Background,BadPix EventSizeDist,EXAFS HKParmInt,ModeParam

Table 3: Cross reference of EPIC CCF components to In-Flight Calibration Plan.

CCF Component	**Calibration Target/ Source**	Comment
LinCoord	NGC2516	Checked also for telescope ageing and other trends
XPSF	LMC X-3 PKS0312-770	For PN, but produces pile-up in MOS, so we use 2 sources
PSFPileUp	LMC X-3 PG1634	Checked with various source
ADUConv	N132D Internal	Internal source has few lines
CTI	1ES0102-72 N132D	Via. multiple pointings
QuantumEf	Terzan 2 G21.5-09 3C58	Various power law spectra observed in other observatories. Check for features attributable to CCD efficiency discrepancies
XAreaEf and Redist	Terzan 2 G21.5-09 3C58 HZ43 Cal83 Cal87	As for Quantum efficiency but more general effective area calibration Additional sources for limited energy regions in soft band
FilterTransX	HZ43 Cal83 Cal87	Particular features of effective area near filter edges checked by these targets
EXAFS	3C58 G21.5-09	Power law spectra and the mirror contamination targets used to detect EPIC EXAFS features
RedistPileUp	PG1634 Mkn205 PKS0558	Via. Mode dependence measurements measure changes in spectra with count rate
EventSizeDist	All	Monitored with PCS
Contam	Various	Soft sources checked for edge features
TimeCorr	PSR0540 Crab Pulsar	
XSLAreaEf XSLDist	Crab Nebula "	Off axis leak Spatial distribution
Background	All	Trend analysis of source excised regions
BadPix	All	Monitored with PCS
HKParmInt ModeParam	All All	Developed by experience Only needed if modes change - observe standard candles
VAreaEf FilTransV	Various Stars "	Observe in diagnostic mode

Table 4: Cross reference of EPIC Inflight Calibration Targets to CCF components

Target Name	**Primary Calibration Goal**	**Secondary Goal**	**CCF Files Verified**
Internal Source	CCD gain, resolution	contamination, CTI	ADUConv, CCDrmf, Contam, CTI
Capella	wavelength scale	resolution, PSF	BoreSight, CrossPSF, LinCoord
HR 1099	wavelength scale	resolution, PSF	BoreSight, CrossPSF, LinCoord
YY Men	wavelength scale	resolution, PSF	BoreSight, CrossPSF, LinCoord
λ And	wavelength scale	resolution, PSF	BoreSight, CrossPSF, LinCoord
AB Dor	wavelength scale	resolution, PSF	BoreSight, CrossPSF, LinCoord
α Cen	wavelength scale	resolution, PSF	BoreSight, CrossPSF, LinCoord
3C273	eff. area		QuantumEf, MiscData, EXAFS
PKS2155	eff. area		QuantumEf, MiscData, EXAFS
Mkn 421	eff. area		QuantumEf, MiscData, EXAFS
GX13+1	eff. area		QuantumEf, MiscData, EXAFS
PSR0540-69	eff. area		QuantumEf, MiscData, EXAFS
Sco X-1	eff. area, CTI	checkout, geometry	BoreSight, CrossPSF, CTI, LinCoord, QuantumEf
Canopus	straylight		FilterTransV
Diag. images	dark current	bad pixels	DarkFrame

Table 5: Cross reference of RGS CCF components to In-Flight Calibration Plan.

CCF Component	**Calibration Target/ Source**	Comment
BoreSight	Capella HR 1099 YY Men λ And AB Dor α Cen Sco X-1	
MiscData	3C273 Mkn 421 GX13+1 PSR0540-69	
ADUConv	Internal Source	
Background	—	combination of weak source observations
BadPix	—	combination of all exposures
CalSourceData	—	ground calibrations
CCDrmf	Internal Source	
ClockPatterns	—	on-board s/w parameters
Contam CrossPSF	Internal Source Capella HR 1099 YY Men λ And AB Dor α Cen Sco X-1	
CTI	Internal Source Sco X-1	
DarkCurrent	—	left empty — unused
DarkFrame	—	collection of diag. images
EventSizeDist	—	from combination of observations
EXAFS	3C273 Mkn 421 GX13+1 PSR0540-69	
FilterTransV	Canopus	
HKParmInt	—	h/k limits to be used for analysis
LinCoord	Capella HR 1099 YY Men λ And AB Dor α Cen Sco X-1	
LineSpreadFunc	—	left empty — unused
ModeParam	—	on-board s/w parameters
PSFPileUp	—	left empty — unused
QuantumEf	3C273 Mkn 421 GX13+1 PSR0540-69	

	Sco X-1	
Redist	—	unused; constructed by `rgsrmfgen`
RedistPileUp	—	unused; constructed by `rgsrmfgen`
WavelengthScale	—	unused; constructed by `rgsrmfgen`

Table 6: Cross reference of RGS CCF components to Inflight Calibration Targets.

Target Name	Calibration Goal	CCF Files
BPM16274	time sensitivity degradation λ-scale Vis_grism photometric stability	PhotToNat Grismcal
GD153, Hz 2	spectrophotometry time sensitivity degradation	Grismcal PhotToNat
GD 153, Hz 2, GD 50, Hz 4, G 93-48, LTT9491, LBB 227	flux conversion zero point definition	ColorTrans ColorTrans
HD 221996, HD 224317, HD 13499 field	λ-scale UV_grism	Grismcal
G 153 field	Geometric distortion PSF U, B and V filters	Astromet PSF1DRB
LH9-LH10 field	PSF UVW1 and UVM2 filters	PSF1DRB
PSR0540-69.3 field	PSF UVW2 filter	PSF1DRB
Sco X-1 field	Grism zero order position	Astromet
G 153 field, HD 5980 field, M 67 field	UBV colour transformation	ColorTrans
EXO 0748-67 field	Large scale Sensitivity variation PSF optical filters	PixToPixSens PSF1DRB
SA95-42	Large scale Sensitivity variation photometric accuracy	PixToPixSens
Internal calibration	event detection setting HV setting image Cosmetics dark count rate	PhotToNat PhotToNat BadPix DarkFrame

Table 7: Cross reference of In-Flight Calibration Targets to OM CCF components

5.2 Cross References between CCF Components and CAL Functions

The list of files is based on [5].

CCF Component	Sections
XPSF	`CAL_rgsgetXRTFigure` (3.4.33)

Table 8: Cross reference of use of telescope related CCF's.

CCF Component	Sections	Status
Boresight	CAL_getBoresightMatrix 3.3.3	
Redist	CAL_getRedist 3.3.22 / 3.3.23 CAL_getEbounds 3.3.21	
BadPix	CAL_getBadPixelList & CAL_getBadPixelMap 3.3.8	
LinCoord	CAL_rawXY2mm CAL_rawmm2XY	
QuantumEf	CAL_getQuantumEfficiency 3.3.14	
FilterTransX	CAL_getFilterTransmission 3.3.13	
PatternLib	CAL_getEventPatterns 3.3.12	
CTI	CAL_mosCTIcorrection 3.3.9 CAL_pnCtiCorrect 3.3.17	
ADUconv	CAL_mosgainCorrect 3.3.11 CAL_pngainCorrect 3.3.18	
Background	CAL_getBackgroundMap,3.3.20	
CalSource	Cal_getCalSourceSpec 3.3.24	
Contam	CAL_getContaminationdata 3.3.24	
DarkFrame	CAL_getmosOffset 3.3.10	
HKParmInt ModeParm	CAL_getHKWindows CAL_getModeParameters	
PSFPileUp	CAL_getPSFMap 3.3.31	

Table 9: Cross reference of use of EPIC related CCF's.

CCF Component	CAL Function
BoreSight	`CAL_getBoresightMatrix` (3.4.1)
MiscData	`CAL_rawXY2mm` (3.4.2) `CAL_mm2rawXY` (3.4.3) `CAL_rawX2chipX` (3.4.5) `CAL_chipX2rawX` (3.4.6) `CAL_getContaminationData` (3.4.16) `CAL_getRFCdefocus` (3.4.24) `CAL_getLSFdefocusDist` (3.4.25) `broadeningDistribution` (3.4.26)
ADUConv	`CAL_offsetCorrect` (3.4.9) `CAL_gainCorrect` (3.4.10) `CAL_rgsgetEvThresh` (3.4.37
Background	after launch
BadPix	`CAL_getBadPixelList` (3.4.7) `CAL_getBadPixelMap` (3.4.7)
CalSourceData	`CAL_getCalSrcRegions` (3.4.14) `CAL_getCalSourceData`(3.4.15)
CCDrmf	left empty — unused
ClockPatterns	`CAL_rgsRawPixCorr` (3.4.4)

	CAL_rawX2chipX (3.4.5) CAL_chipX2rawX (3.4.6) CAL_rgsGetDarkFrame (3.4.8)
Contam	CAL_getContaminationData (3.4.16)
CrossPSF	CAL_rgsCrossPSF (3.4.23)
CTI	CAL_rgsCTIcorrect (3.4.11)
DarkCurrent	left empty
DarkFrame	CAL_rgsGetDarkFrame (3.4.8) CAL_offsetCorrect (3.4.9)
EventSizeDist	CAL_getEventSize (3.4.17)
EXAFS	CAL_getCCDQuantumEfficiency (3.4.13)
FilterTransV	left empty; after calibrations
HKParmInt	CAL_getHKwindows (3.4.38)
LinCoord	CAL_rawXY2mm (3.4.2) CAL_mm2rawXY (3.4.3) CAL_getRGAIntercept (3.4.29) CAL_getRFCdefocus (3.4.24) CAL_getLSFdefocusDist (3.4.25)
LineSpreadFunc	CAL_rgsGetScatter (3.4.20) CAL_rgsGetScatterPars (3.4.19) CAL_rgsGetLAScatterRoughness3.4.18 CAL_rgsGetLAScatter3.4.21 CAL_rgsGetSAScatter3.4.22 CAL_getRGAFigure (3.4.31) CAL_geGratBows3.4.32
ModeParam	used by QLA
PSFPileUp	left empty
QuantumEf	CAL_getCCDQuantumEfficiency (3.4.13) CAL_getRGAQuantumEfficiency (3.4.27) CAL_getRGAVign (3.4.28) CAL_getRGAIntercept (3.4.29) CAL_getRGSEffAreaCorr (3.4.35)
Redist	CAL_getRedistribution (3.4.12)
RedistPileUp	left empty
WavelengthScale	left empty — unused

Table 10: Cross reference of use of RGS related CCF's.

CAL_	**CCF Component**	**SAS task**
CAL_omGetDegradationCoeffs	PHOTTONAT	omlcbuild mssllib
CAL_omGetFluxConvFactors	COLORTRANS	
CAL_omGetGrismFlux	GRISMCAL	omgrism
CAL_omGetGrismWavelength	GRISMCAL	omgrism
CAL_omDistortion	ASTROMET	omatt omgprep omgsource omsource

		mssllib
CAL_omInverseDistortion	ASTROMET	mssllib
CAL_omGetColorTransform CAL_omGetColorTransformator CAL_omColorTransBranches CAL_omColorTransValidityRanges CAL_omColorTransValidityRange CAL_omStandardColor CAL_omStandardMagnitude	COLORTRANS	omsrclistcomb omsrclistcomb omsrclistcomb
CAL_omGetPlateScale	ASTROMET	omatt omdetect omdrifthist omgprep omgsource omphotom omsource omsrclistcomb
CAL_omGetPSFmap CAL_omGetPSF CAL_omPsfEncircledEnergy CAL_omPsfCircleRadius	PSF1DRB	omlcbuild mssllib omlcbuild,ommag
CAL_omLargeSenseVariation	LARGESCALESENS	
CAL_omLEDTemplate	PIXTOPIXSENS	
CAL_omPhotoMagnitude CAL_omNaturalMagnitude (alias)	COLORTRANS	omlcbuild ommag omsrclistcomb mssllib
CAL_omGetFrameParameters	PHOTTONAT	mssllib
CAL_omPhotoNatural	PHOTTONAT	omlcbuild ommag mssllib
CAL_omPixelSenseVariation	PIXTOPIXSENS	omflatgen
CAL_omGetAperRadius	PSF1DRB COLORTRANS	omlcbuild ommag omphotom omsource mssllib
CAL_getHKwindows	HKPARMINT	
CAL_getBadPixelMap	BADPIX	omcosflag omfastflat
CAL_getBoresightMatrix CAL_getBoresight CAL_withSacCoordOrientation CAL_projectOntoTelCoord CAL_toEulerAngles CAL_toDirectionCosineMatrix1	BORESIGHT	omatt omgprep omgsource omsource

Table 11: Cross reference of use of OM related CCF's by CAL call and corresponding SAS task.