The JSIM software package

Version 1.00

N.J. Westergaard
DSRI, Copenhagen

December 16, 1999

Contents

1 INTRODUCTION

This JSIM software package is written in C and it is meant to be ISDC compatible and it includes DAL, DAL3GEN, PIL, RIL, and COMMON functions. The package consists of two parts. The first set of executables sets up the data structure which is required by the photon generator. All executables use the DOL (Data Object Location) of a DAL group as an entry point to all other required data structures taking advantage of the DAL grouping structure.
The two JEM-X instruments are treated as two distinct instruments referred to as JMX1 and JMX2, but the executables will deal with both in the same run. The generic name JEMX will seldom be used as an instrument name but will be used in connection with the physical properties of various materials.

2 GENERAL OUTLINE

The package consists of a number of programs (executables) to be executed in a specific order. Each executable has its own subdirectory under '`/home/njw/jsim/`' where all the relevant files for the compilation and execution have been stored (source codes, include files, .par files, and Makefile).
The directory '`/home/njw/jsim/run/`' holds the executables and the programs can be run from there.
JSIM_AUX is the name for a set of programs to set up a grouping structure when the ISDC programs are not used. All data structures are set up but not necessarily filled out, the pointing directions defined for the dithering pattern and the X-ray sources are extracted from a specified catalog. A photon generator executable is also supplied.
JSIM is the name for the package of programs that will simulate the photons in the instruments, their detection, the behavior of the data-taking in the pre-defined instrumental modes, and the image formation.

2.1 Notice on DAL_CURRENT_DIR

The DAL library uses an environment variable 'DAL_CURRENT_DIR' as an indicator where to create new files.

When a FITS file (a data structure) is to be created using DAL the function 'DALobjectCreate' can be used.

Not using a template file: The NAME parameter must not contain any '/' but only a base filename. DAL will append the name extension '.fits' automatically. The file will created in the directory where DAL_CURRENT_DIR points.

Using a template file: Here there are two options: If DAL_CURRENT_DIR has been defined then the NAME parameter must not contain any '/' but only a base filename. DAL will append the name extension '.fits' automatically. The file will created in the directory where DAL_CURRENT_DIR points. On the other hand if DAL_CURRENT_DIR is *not* defined then a full path can be specified.

In this package only **JSIMsetup** uses DAL_CURRENT_DIR.

2.2 Strategy with respect to .par-files

Fopr the compatibility with FTOOLS programs the environment variable PFILES defined the .par-files must be present both in e.g.

`/home/njw/jsim/JSIMSciWinList`

and in

/home/njw/jsim/par/

This can be achieved using soft links.
/home/njw/jsim/par/JSIMSciWinList.par is a link to the physical file
/home/njw/jsim/JSIMSciWinList/JSIMSciWinList.par
As an example PFILES can look like this:

/home/njw/jsim/par:/home/njw/pfiles;/home/njw/jsim/par:/sas3a/ftools/SunOS_4.1_sparc/bin

in which case both the JSIM programs and the FTOOLS programs can run.

3 DATA GROUPING STRUCTURE

The files are organized in a way that is meant to be compatible with the DAL, DAL3GEN, and DAL3JEMX data access routines. All data from a simulated observation is stored in a grouping structure where the DOL to its top is used as input for all executables. In that way it is possible to access any data related to the observation without any further reference to files and directories.
The complete structure after running **JSIMsetup** or the ISDC equivalent is shown in the following diagram. Only JEMX relevant data structures are given for shortness.

```
GRP(GNRL-OG.1-GRP or other)
   |
   |
   |--- SWG(GNRL-SWG9-GRP, or RRRR00010001)
   |         |
   |         |--- AUXL-ATTI-HIS
   |         |--- OSIM-PDEF-LST
   |         |--- OSIM-JEMX-PHO
   |         |--- OSIM-SRCL-GRP
   |         |      |
   |         |      |--- OSIM-JEMX-SRC
   |         |
   |         |--- JMX1-EVNT-LST
   |         |--- JMX1-FULL-COR
   |         |--- JMX1-REST-COR
   |         |--- JMX1-SHAD-INT
   |         |--- JMX1-RSTI-INT
   |         |--- JMX1-SKYC-INT
   |         |--- JMX1-SKY.-INT
   |         |--- JMX2-EVNT-LST
   |         |--- JMX2-FULL-COR
   |         |--- JMX2-REST-COR
   |         |--- JMX2-SHAD-INT
   |         |--- JMX2-RSTI-INT
   |         |--- JMX2-SKYC-INT
   |         |--- JMX2-SKY.-INT
   |
      .
```

```
        .   (repeats for as many SWGs as required)
        .
```

4 JSIM_AUX PROGRAMS

All files created during a simulation are put in the directory pointed to by DAL_CURRENT_DIR.

4.1 JSIMsetup

This program defines the simulation conditions i.e. the basic position on the sky (RA and DEC), the dithering pattern (or single pointing), the duration of each pointing (Science Window). It creates the complete group structure for the OG or higher level group such as an ORG. The top grouping table (GRP) is an Observation Group (OG) in the current version since this complies with ISDC defined OSIM groups. The requested number of SWGs are defined, each in a separate file. All children of the SWGs are set up but not filled by this program, so that the following programs just have to find the data structures. All files are created in the directory defined by the environment variable: DAL_CURRENT_DIR.
JSIMsetup also updates the file 'OSIM-OBS.-LST' which holds list of simulations performed. If it does not already exist in DAL_CURRENT_DIR it will be created. This file (datastructure) is not part of the grouping structure.

Parameters:

observationName	Name of new observation
JEMXunit	JEMX instrument unit number (1 or 2)
RA	Right ascension in decimal degrees
DEC	Declination in decimal degrees
ROLL	Roll angle in decimal degrees
exposureTime	Exposure time in s of Science Window
ditherCode	Code for dithering pattern (0, 1, or 2)
verbosity	Level of debugging output

Functions:

JSIMGLitoan see section 10

JSIMGLreverse see section 10

JSIMGLgetNextAutoName see section 10

4.2 JSIMSciWinList

Produces the data structure that contains information about the science windows involved in the dithering pattern with appropriate treatment of coordinates. Currently only staring mode (one single pointing), hexagonal mode (7 pointings), and square mode (25 pointings) are implemented. The data structure OSIM-PDEF-LST is updated accordingly with RA and DEC of pointing directions.

Parameters:

DOL_of_GRP	DOL of top level group
verbosity	Level of debugging output

Functions:

JSIMitoan see section 10

matMult see section 10

zero2pi see section 10

rotationMatrix produces a rotation matrix for the coordinate transformation between the satellite coordinate system and the celestial (Equatorial) system.

yRot rotates an angle around the Y-axis

zRot rotates an angle around the Z-axis

makeDitherPattern produces a series of pointings distributed according to INTEGRAL's proposed basic dithering patterns.

JSIMreverse see section 10

4.3 JSIMSourceList

Reads the RASS-BSC as well as an auxiliary catalogue AUXL-CAT and fills out the data structure OSIM-JEMX-SRC with a table of sources for the entire observation, i.e. for all the SWGs. **JSIMSourceList** steps through the SWGs and compiles a list of sources in all FOVs (Field Of View) while avoiding duplications.

Parameters:

DOL_of_GRP	DOL of top level group
basicSourceCat	Basic input source catalogue
auxSourceCat	Aux. input source catalogue
FOVradius	Radius of FOV in degrees
verbosity	Level of debugging output

Functions:

matVmult see section 10

matprint see section 10

JSIMitoan see section 10

JSIMreverse see section 10

4.4 JSIMGeneratePhotons

Produces a FITS file for each SWG with photon information, OSIM-JEMX-PHO, based on the source list generated by **JSIMSourceList**. Also photons from the diffuse extragalactic X-ray background are included.
All photons hitting a basic rectangular area just covering both JEMX masks are generated based on the X-ray source spectra and positions.
Due to the very large files produced with photon information, a UNIX pipe system has been introduced that will transmit the photon information directly to the progam **JSIMraytrace**. This option is selected by a parameter, in which case the following raytracing program, **JSIMraytrace**, is started up automatically with the pipe option.

Parameters:

DOL_of_GRP	DOL of top level group
JEMXdata	File with grouping table for JEMX instr. data
radiusFOV	Radius of FOV in degrees
verbosity	Level of debugging output
usePipe	Is piping to the raytracer required ?

Functions:

stowePhoton receives the photon data and writes them to the photon table. It allocates a memory buffer for all the photon data for a given number of photons. The first call must be an initialization (control == 0), the following calls updates (control == 1) and at last a call to write the buffers to the file and free the memory (control == -1).

getAccum accumulates an energy spectrum.

flux calculates the photon spectrum at a specified energy.

csect returns interstellar atomic absorption cross-section.

getJEMXParmReal retrieve a JEMX parameter value

LinLinInterpol see section 10

getSx integrates the flux between 2 and 10 keV and returns the result in $\mathrm{erg\,s^{-1}\,cm^{-2}}$.

5 JSIM PROGRAMS

5.1 JSIMraytrace

Goes through all SWGs in an observation group and reads the photon table in each SWG or receives the data via an anonymous (UNIX) pipe and produces two tables of events, JMX1-EVNT-LST and JMX2-EVNT-LST, after raytracing through the JEM-X instrument for all the SWGs.
If the use of a UNIX pipe has been requested it is started in a condition awaiting for data from **JSIMGeneratePhotons** or any other executable designed for the pipe connection such as the Southampton '**os_photgen**' photon generator.

Parameters:

DOL_of_GRP	DOL of OG or higher level group
JEMXdata	File with grouping table for JEMX instr. data
verbosity	Level of debugging output
usePipe	Is piping to the raytracer required ?

Functions:

JSIMRTretrievePhoton reads the photon information one at a time but based on buffers of fixed size to minimize the number of I/O operations.

JSIMRTstoweEvent puts the event data into the event table based on buffers of fixed size to minimize the number of I/O operations.

JSIMGLgetHexCell return the indices for the hexagonal cell to which a given position belongs.

JSIMGLgetJEMXParmReal retrieve a JEMX parameter value.

JSIMGLgetJEMXArrShort retrieve a 2D array of short integers from JEM-X instrumental data.

5.2 JSIMModeConversion

Reads the event table and produces a table of events, JMX1-MMMM-COR, with proper modifications as inflicted by the operational mode chosen for JEM–X for all the SWGs. The four-letter code, MMMM, is one of the four JEMX data modes: FULL, REST, SPEC, or SPTI. Both JEM–X units are handled.
The operational mode is transmitted by the data structures, but these can be overwritten by setting in the parameter file (or argument input).

Parameters:

DOL_of_GRP	DOL of observation or higher level group
JEMXmode	JEMX operational mode: FULL REST SPEC SPTI
verbosity	Level of debugging output

Functions:

retrieveInEvent reads the event information one at a time but based on buffers of fixed size to minimize the number of I/O operations.

JMstoweEventFULL puts the event data into the event table based on buffers of fixed size to minimize the number of I/O operations according to the JEM–X FULL mode.

JMstoweEventREST puts the event data into the event table based on buffers of fixed size to minimize the number of I/O operations according to the JEM–X REST mode.

JSIMGreyFilter applies the grey filter action, i.e. removes events in a stochastic manner when the countrate is too high.

getGreyFilterlevel calculates the grey filter level ($1 \cdots 32$) based on the count rate.

5.3 JSIMTimeEnergyShadowgram

Reads the data structures with corrected data and produces shadowgrams in JMX1-SHAD-INT and JMX2-SHAD-INT. Eventselection based on pulseheights has been implemented whereas the selection by time is not ready yet.

Parameters:

DOL_of_GRP	DOL of observation or higher level group
JMX1mode	JEMX1 operational mode: FULL REST SPEC SPTI
JMX2mode	JEMX2 operational mode: FULL REST SPEC SPTI
verbosity	Level of debugging output
PIlo	Lowest PI value to be included
PIhi	Highest PI value to be included

Functions:

getHexCell return the indices for the hexagonal cell to which a given position belongs.

5.4 JSIMBasicImageReconstruction

Reads the data structures with shadowgrams and produces restored images: JMX1-RSTI-INT and JMX2-RSTI-INT.

Parameters:

DOL_of_GRP	DOL of observation or higher level group
JEMXdata	DOL of JEMX data group
verbosity	Level of debugging output
actualSciWin	Number of SW to make image reconstruction for
actualJEMXunit	Number of JEMX unit to make image reconstruction for

Functions:

JConvolve makes the convolution of the shadowgram with the mask pattern.

getJEMXArrShort retrieve a 2D array of short integers from JEM-X instrumental data.

5.5 JSIMmakeSkyImage

Converts the reconstructed 'skew' images to either an ordinary image in Cartesian coordinates (parameter selectType = "image", JMX1-SKYC-INT and JMX2-SKYC-INT) or to an image in celestial coordinates (parameter selectType = "sky", JMX1-SKY.-INT and JMX2-SKY.-INT).

Parameters:

DOL_of_GRP	DOL of observation or higher level group
verbosity	Level of debugging output
numPixelSubDiv	Number of pixel subdivisions
selectType	select ordinary "image" or "sky"

Functions:

crossProd see section 10

6 AUXILIARY PROGRAMS

6.1 jdata2fits

is an IDL program to convert the information about JEMX from S-format (in file */home/njw/jsim/dat/jemx1_data.sformat*) to FITS format as keyword values in file */home/njw/jsim/dat/jemx1_data.fits*.

6.2 newgroup

is a C-program in */home/njw/jsim/DataOrg/* that creates a FITS file with a grouping table as its first (and only) extension. A member to be included is also required. This member is described by the file name and the extension number in the file.

6.3 addmem

is a C-program in */home/njw/jsim/DataOrg/* that adds a member data element (FITS extension) to an existing grouping table e.g. created with program **newgroup**, see above.

6.4 sfile2fits

is a Perl script (found in */home/njw/perl_scripts*) that converts an ASCII file in S-format to a FITS file with the table as a binary table extension. It prompts for the names of the columns as well as the units.

6.5 phdu2imgext

is a C-program in */home/njw/jsim/DataOrg/* that converts a (integer) array in the primary HDU to an image extension in a FITS file.

6.6 JSIMaddSrcAuxCat

is a C-program in */home/njw/jsim/JSIMaddSrcAuxCat/* that adds a celestial X–ray source to the auxiliary source datalogue. The necessary information is interactively requested by the program.

6.7 JSIMdelSrcAuxCat

is a C-program in */home/njw/jsim/JSIMdelSrcAuxCat/* that removes a celestial X–ray source from the auxiliary source datalogue. The necessary information is interactively requested by the program.

6.8 ASCII2FITS_RASS-BSC

is a C-program in */home/njw/jsim/ASCII2FITS_RASS-BSC/* that reads the ROSAT All Sky Survey Bright Source Catalogue (W. Voges *et al.*, 1999, Astronomy and Astrophysics, Main Journal (MPE preprint 467, July 1999)) in ASCII format and produces a FITS binary table with basic source information such as column density and power-law index derived from hardness ratios and the assumption of a power-law spectrum. The source intensity in the range 2 - 10 keV is calculated on the basis of the reported broad band count rate and the derived spectral parameters.
The resulting catalogue file is used as input to **JSIMSourceList**.

7 OTHER FILES

/home/njw/jsim/set_environment is used for setting up the environment variable 'DAL_CURRENT_DIR' before running JSIM by the UNIX command '**source**'.

/home/njw/jsim/dat/RASS-CAT.fits holds most important information of the ROSAT All Sky Survey Bright Sources Catalogue.

/home/njw/jsim/dat/AUXL-CAT.fits holds other sources mainly those input for test purposes and those not correctly represented in the ROSAT catalogue. Updated by the use of
JSIMaddSrcAuxCat.

/home/njw/jsim/dat/jemx1_data.sformat Basic data describing JEMX in S-format. Used by JSIMGeneratePhotons and JSIMraytrace

/home/njw/jsim/dat/jemx_data.fits Basic data describing JEMX in FITS. It is a grouping table that proints to more specific information (see section 12). Used by **JSIMGeneratePhotons** and **JSIMraytrace**.

8 FLOW OF PROGRAMS

1. cd `/home/njw/jsim/run/`
   Edit *set_environment* and 'source' it.
2. Run **JSIMsetup** to set the OG up, and define the basic pointing, exposure times, dithering pattern.
3. Run **JSIMSciWinList** to define the S/C attitude of each pointing.
4. Run **JSIMSourceList** to define the catalogue sources for all science window FOVs.
5. Run **JSIMGeneratePhotons** to make the tables with input photons or setup a pipe to **JSIMraytrace** in which case the next step is skipped.
6. Run **JSIMraytrace** to trace the photons and make the event tables.
7. Run **JSIMModeConversion** to prepare output products according to the JEM–X mode and grey filter.
8. Run **JSIMTimeEnergyShadowgram** to produce the shadowgrams.
9. Run **JSIMBasicImageReconstruction** for the image reconstruction and produce 'skew' images.
10. Run **JSIMmakeSkyImage** to transform the images to a usual coordinate system.

The macro (or script) */home/njw/jsim/run/gojsim* executes all these steps.

9 DATA STRUCTURES

OSIM-PDEF-LST DAL table. Spacecraft attitude information for entire observation. Filled by **JSIMSciWinList**.

OSIM-JEMX-SRC DAL table. Excerpt of source catalogue of sources in all FOVs for a particular observation. Filled by **JSIMSourceList**.

OSIM-JEMX-PHO DAL table. Table containing photon information for specific science window.

JMX1-EVNT-LST DAL table. Table containing event information for a science window and for JEM–X–1.

JMX2-EVNT-LST DAL table. Table containing event information for a science window and for JEM–X–2.

JMX1-FULL-COR Table with event information similar to the ISDC 'JMXi-FULL-COR'

JMX2-FULL-COR Table with event information similar to the ISDC 'JMXi-FULL-COR'

JMX1-SHAD-INT DAL image. Integer array containing shadowgram with time and energy selection.

JMX2-SHAD-INT DAL image. Integer array containing shadowgram with time and energy selection.

JMX1-RSTI-INT reconstructed image in 'skew' coordinates.

JMX2-RSTI-INT reconstructed image in 'skew' coordinates.

JMX1-SKYC-INT The image transformed to Cartesian coordinates.

JMX2-SKYC-INT The image transformed to Cartesian coordinates.

JMX1-SKY.-INT The image transformed to celestial WCS coordinates with proper FITS keywords for image analysis.

JMX2-SKY.-INT The image transformed to celestial WCS coordinates with proper FITS keywords for image analysis.

10 GENERAL FUNCTIONS

The source codes are found in the directory
/home/njw/jsim/GeneralLibrary

JSIMGLitoan is a function to return a number as a string with a specified number of characters.

JSIMGLreverse reverses the order of a character string.

matMult multiplies two 3×3 matrices together.

matVmult multiplies a vector with a 3×3 matrix.

matPrint makes a nice print of a 3×3 matrix.

zero2Pi adjusts a number (an angle) to the range between zero (included) to $2pi$ (excluded).

crossProd evaluates the vectorial cross-product of two 3-element vectors.

11 ERROR CODES

The variable 'status' is returned from all functions except very low level ones. The value is 'ISDC_OK' if no formal error has occur, and has a negative value otherwise. 'status' is also an input argument and if its value is not 'ISDC_OK' the only action of the the functions will be to reflect the value of 'status'. Range of values are reserved for various libraries and programs and for JSIM the values are:

Not approved by ISDC:	
JSIMsetup	$-28600 \longrightarrow -28619$
JSIMSciWinList	$-28620 \longrightarrow -28639$
JSIMSourceList	$-28640 \longrightarrow -28659$
JSIMGeneratePhotons	$-28660 \longrightarrow -28679$
Approved by ISDC:	
JSIMraytrace	$-18600 \longrightarrow -18639$
JSIMModeConversion	$-18640 \longrightarrow -18659$
JSIMTimeEnergyShadowgram	$-18660 \longrightarrow -18679$
JSIMBasicImageReconstruction	$-18680 \longrightarrow -18699$
JSIMmakeSkyImage	$-18700 \longrightarrow -18719$
Reserved for later use	$-18720 \longrightarrow -18799$

12 JEMX PARAMETER ORGANIZATION

All parameters connected to the physical JEM-X instrument are organized in a grouping table. The file that holds this table is

/home/njw/jsim/dat/jems_data.fits

The particular keyword values ('attributes' in DAL parlance) can be obtained via a call of **getJEMXParmReal** e.g.:

status = getJEMXParmReal("RAD_MASK", "JMX1-MASK-MOD", JEMXdata, &value, status);

where the character string 'JEMXdata' is the DOL of the grouping table.
The various components are described below. All tables are functions of energy. The data structures in parenthsis '()' have not yet been implemented.

JMXi-DETE-MOD (binary table) with 'intrinsic' FWHM of the detector position resolution.

JMXi-MASK-MOD (image extension) with mask pattern. Positional and rotational information in keywords. Basic exoskeleton numerical information in keywords.

JMXi-QEFF-MOD (binary table) with quantum efficiency seen from 'above' collimator on-axis.

(JMXi-WTRN-MOD) (binary table) with window transmission.

(JMXi-COLL-MOD) (image extension) with collimator footprint on detector window and other information in keywords.

(JMXi-CVIG-MOD) (binary table) with collimator vignetting for a series of azimuth angles.

JEMX-ATXE-MOD (binary table) with attenuation length of Xe in 5 atm.

JEMX-ATAL-MOD (binary table) with attenuation length of Al.

JEMX-ATCU-MOD (binary table) with attenuation length of Cu.

JEMX-ATMO-MOD (binary table) with attenuation length of Mo.

(JEMX-ATW.-MOD) (binary table) with attenuation length of W.

JMX1-DETE-MOD

```
*********************************************
XTENSION= 'BINTABLE'           / binary table extension
BITPIX  =                    8 / 8-bit bytes
NAXIS   =                    2 / 2-dimensional binary table
NAXIS1  =                    8 / width of table in bytes
NAXIS2  =                   10 / number of rows in table
PCOUNT  =                    0 / size of special data area
GCOUNT  =                    1 / one data group (required keyword)
TFIELDS =                    2 / number of fields in each row
TTYPE1  = 'ENERGY  '           / label for field   1
TFORM1  = 'E       '           / data format of field: 4-byte REAL
TUNIT1  = 'keV     '           / physical unit of field
TTYPE2  = 'FWHM    '           / label for field   2
TFORM2  = 'E       '           / data format of field: 4-byte REAL
TUNIT2  = 'mm      '           / physical unit of field
SFILENAM= '/home/njw/jemx_sim/dat/msgc_poserr.dat'
COMMENT  //
COMMENT  // A simple guess for the position accuracy (FWHM in mm) of the
COMMENT  // JEMX MSGC as a function of energy in keV.
COMMENT  //
COMMENT  // Energy   FWHM
COMMENT  //
HISTORY   This FITS file was created by the FCREATE task.
HISTORY   fcreate3.0d at 10/5/99 10:31:11
EXTNAME = 'JMX1-DETE-MOD'      / Specific data on JEMX-1
GRPID1  =                   -1 / EXTVER of Group containing this HDU
GRPLC1  = 'file:///home/njw/jsim/dat/jemx_data.fits' / URL of file containing Gr
COMMENT   This table holds intrinsic position resolution
COMMENT   due to electronic noise and the MS grid
RADMASK =              267.500 / [mm] Mask radius
RADDET  =              125.000 / [mm] Detector radius
HDRIFT  =               55.000 / [mm] Height of drift layer
***** END OF FILE *****
```

JMX1-MASK-MOD

```
*********************************************
XTENSION= 'IMAGE   '           / IMAGE extension
BITPIX  =                   16 / number of bits per data pixel
NAXIS   =                    2 / number of data axes
NAXIS1  =                  200 / length of data axis 1
NAXIS2  =                  200 / length of data axis 2
PCOUNT  =                    0 / required keyword; must = 0
GCOUNT  =                    1 / required keyword; must = 1
EXTNAME = 'JMX1-MASK-MOD'      / Data structure with JEMX mask information
GRPID1  =                   -1 / EXTVER of Group containing this HDU
GRPLC1  = 'file:///home/njw/jsim/dat/jemx_data.fits' / URL of file containing Gr
NU      =                22501 / Unit pattern size
R       =                 1133 / Multiplier
EXPAND  =                1.030 / Expanding factor
RAD_MASK=                267.5 / Radius of mask (mm)
RAD_UPAT=                267.5 / Radius of unit pattern (mm)
RHO     =                1.650 / Half size of unit cell (mm)
TIMSIZ  =                  200 / Matrix size of pattern
RMIN_RG1=              45.0000 / [mm]
RMAX_RG1=              48.0000 / [mm]
RMIN_RG2=              114.900 / [mm]
RMAX_RG2=              117.100 / [mm]
RMIN_RG3=              210.000 / [mm]
RMAX_RG3=              212.200 / [mm]
X_BT1_01=              49.5000 / [mm]
Y_BT1_01=              0.00000 / [mm]
X_BT1_03=              24.7500 / [mm]
Y_BT1_03=              42.8680 / [mm]
X_BT1_05=             -24.7500 / [mm]
Y_BT1_05=              42.8680 / [mm]
X_BT1_07=             -49.5000 / [mm]
Y_BT1_07=              0.00000 / [mm]
X_BT1_09=             -24.7500 / [mm]
Y_BT1_09=             -42.8680 / [mm]
X_BT1_11=              24.7500 / [mm]
Y_BT1_11=             -42.8680 / [mm]
R_BT1   =              4.50000 / [mm]
X_BT2_01=              116.000 / [mm]
Y_BT2_01=              0.00000 / [mm]
X_BT2_03=              58.0000 / [mm]
Y_BT2_03=              100.459 / [mm]
X_BT2_05=             -58.0000 / [mm]
Y_BT2_05=              100.459 / [mm]
X_BT2_07=             -116.000 / [mm]
Y_BT2_07=              0.00000 / [mm]
X_BT2_09=             -58.0000 / [mm]
Y_BT2_09=             -100.459 / [mm]
X_BT2_11=              58.0000 / [mm]
Y_BT2_11=             -100.459 / [mm]
R_BT2   =              4.50000 / [mm]
X_S_01_1=              49.5000 / [mm]
Y_S_01_1=              0.00000 / [mm]
X_S_01_2=              277.000 / [mm]
Y_S_01_2=              0.00000 / [mm]
X_S_02_1=              101.412 / [mm]
Y_S_02_1=              58.5500 / [mm]
X_S_02_2=              181.865 / [mm]
Y_S_02_2=              105.000 / [mm]
X_S_03_1=              24.7500 / [mm]
Y_S_03_1=              42.8680 / [mm]
X_S_03_2=              138.500 / [mm]
Y_S_03_2=              239.889 / [mm]
```

```
X_S_04_1=            0.00000 / [mm]
Y_S_04_1=            117.100 / [mm]
X_S_04_2=            0.00000 / [mm]
Y_S_04_2=            210.000 / [mm]
X_S_05_1=           -24.7500 / [mm]
Y_S_05_1=            42.8680 / [mm]
X_S_05_2=           -138.500 / [mm]
Y_S_05_2=            239.889 / [mm]
X_S_06_1=           -101.412 / [mm]
Y_S_06_1=            58.5500 / [mm]
X_S_06_2=           -181.865 / [mm]
Y_S_06_2=            105.000 / [mm]
X_S_07_1=           -49.5000 / [mm]
Y_S_07_1=            0.00000 / [mm]
X_S_07_2=           -277.000 / [mm]
Y_S_07_2=            0.00000 / [mm]
X_S_08_1=           -101.412 / [mm]
Y_S_08_1=           -58.5500 / [mm]
X_S_08_2=           -181.865 / [mm]
Y_S_08_2=           -105.000 / [mm]
X_S_09_1=           -24.7500 / [mm]
Y_S_09_1=           -42.8680 / [mm]
X_S_09_2=           -138.500 / [mm]
Y_S_09_2=           -239.889 / [mm]
X_S_10_1=            0.00000 / [mm]
Y_S_10_1=           -117.100 / [mm]
X_S_10_2=            0.00000 / [mm]
Y_S_10_2=           -210.000 / [mm]
X_S_11_1=            24.7500 / [mm]
Y_S_11_1=           -42.8680 / [mm]
X_S_11_2=            138.500 / [mm]
Y_S_11_2=           -239.889 / [mm]
X_S_12_1=            101.412 / [mm]
Y_S_12_1=           -58.5500 / [mm]
X_S_12_2=            181.865 / [mm]
Y_S_12_2=           -105.000 / [mm]
W_RED_SP=            2.20000 / [mm]
W_GRN_SP=            2.20000 / [mm]
***** END OF FILE *****
```

JMX1-QEFF-MOD

```
********************************************
XTENSION= 'BINTABLE'           / binary table extension
BITPIX  =                    8 / 8-bit bytes
NAXIS   =                    2 / 2-dimensional binary table
NAXIS1  =                    8 / width of table in bytes
NAXIS2  =                  229 / number of rows in table
PCOUNT  =                    0 / size of special data area
GCOUNT  =                    1 / one data group (required keyword)
TFIELDS =                    2 / number of fields in each row
TTYPE1  = 'ENERGY  '           / label for field   1
TFORM1  = 'E       '           / data format of field: 4-byte REAL
TUNIT1  = 'keV     '           / physical unit of field
TTYPE2  = 'QUANTEFF'           / label for field   2
TFORM2  = 'E       '           / data format of field: 4-byte REAL
COMMENT  *
COMMENT  * QE for JEM-X as of 950823 detector configuration
COMMENT  * without fluorescence gating
COMMENT  * The collimator is included with a transmission of 82%
COMMENT  *
COMMENT  * The result of Carl's MonteCarlo calculation and his
COMMENT  * IDL plotting routine 'pljemxqe.pro'
COMMENT  *
HISTORY   This FITS file was created by the FCREATE task.
HISTORY   fcreate3.0d at 25/2/99 12:8:50.
EXTNAME = 'JMX1-QEFF-MOD'      / Quantum efficiency for JEMX_1
GRPID1  =                   -1 / EXTVER of Group containing this HDU
GRPLC1  = 'file:///home/njw/jsim/dat/jemx_data.fits' / URL of file containing Gr
***** END OF FILE *****
```

JEMX-ATXE-MOD

```
*********************************************
XTENSION= 'BINTABLE'           / binary table extension
BITPIX  =                    8 / 8-bit bytes
NAXIS   =                    2 / 2-dimensional binary table
NAXIS1  =                    8 / width of table in bytes
NAXIS2  =                  411 / number of rows in table
PCOUNT  =                    0 / size of special data area
GCOUNT  =                    1 / one data group (required keyword)
TFIELDS =                    2 / number of fields in each row
TTYPE1  = 'Energy  '           / label for field   1
TFORM1  = 'E       '           / data format of field: 4-byte REAL
TUNIT1  = 'keV     '           / physical unit of field
TTYPE2  = 'att_len '           / label for field   2
TFORM2  = 'E       '           / data format of field: 4-byte REAL
TUNIT2  = '/mm     '           / physical unit of field
SFILENAM= '/home/njw/jemx_sim/dat/att_len_xe.dat'
COMMENT  //
COMMENT  // Attenuation lengths for Xe (5 atm) in cm
COMMENT  // E(keV)   len(um)
HISTORY   This FITS file was created by the FCREATE task.
HISTORY   fcreate3.0d at 1/5/98 16:26:23
EXTNAME = 'JEMX-ATXE-MOD'      / Attenuation of Xrays in Xenon
GRPID1  =                   -1 / EXTVER of Group containing this HDU
GRPLC1  = 'file:///home/njw/jsim/dat/jemx_data.fits' / URL of file containing Gr
***** END OF FILE *****
```

JEMX-ATAL-MOD

```
*********************************************
XTENSION= 'BINTABLE'           / binary table extension
BITPIX  =                    8 / 8-bit bytes
NAXIS   =                    2 / 2-dimensional binary table
NAXIS1  =                    8 / width of table in bytes
NAXIS2  =                   67 / number of rows in table
PCOUNT  =                    0 / size of special data area
GCOUNT  =                    1 / one data group (required keyword)
TFIELDS =                    2 / number of fields in each row
TTYPE1  = 'Energy  '           / label for field   1
TFORM1  = 'E       '           / data format of field: 4-byte REAL
TUNIT1  = 'keV     '           / physical unit of field
TTYPE2  = 'att_len '           / label for field   2
TFORM2  = 'E       '           / data format of field: 4-byte REAL
TUNIT2  = '/mm     '           / physical unit of field
SFILENAM= '/home/njw/jemx_sim/dat/att_len_al.dat'
COMMENT  //
COMMENT  // Attenuation length in Aluminium in /mm
COMMENT  // Derived from 'abs_coeff.al'
COMMENT  //
HISTORY   This FITS file was created by the FCREATE task.
HISTORY   fcreate3.0d at 1/5/98 16:23:2.
EXTNAME = 'JEMX-ATAL-MOD'      / Attenuation of Xrays in Aluminium
GRPID1  =                   -1 / EXTVER of Group containing this HDU
GRPLC1  = 'file:///home/njw/jsim/dat/jemx_data.fits' / URL of file containing Gr
***** END OF FILE *****
```

JEMX-ATCU-MOD

```
*********************************************
XTENSION= 'BINTABLE'           / binary table extension
BITPIX  =                    8 / 8-bit bytes
NAXIS   =                    2 / 2-dimensional binary table
NAXIS1  =                    8 / width of table in bytes
NAXIS2  =                   70 / number of rows in table
PCOUNT  =                    0 / size of special data area
GCOUNT  =                    1 / one data group (required keyword)
TFIELDS =                    2 / number of fields in each row
TTYPE1  = 'Energy  '           / label for field   1
TFORM1  = 'E       '           / data format of field: 4-byte REAL
TUNIT1  = 'keV     '           / physical unit of field
TTYPE2  = 'att_len '           / label for field   2
TFORM2  = 'E       '           / data format of field: 4-byte REAL
TUNIT2  = '/mm     '           / physical unit of field
SFILENAM= '/home/njw/jemx_sim/dat/att_len_cu.dat'
COMMENT  //
COMMENT  // Attenuation length in Copper in /mm
COMMENT  // Derived from 'abs_coeff.cu'
COMMENT  //
HISTORY   This FITS file was created by the FCREATE task.
HISTORY   fcreate3.0d at 1/5/98 16:24:33
EXTNAME = 'JEMX-ATCU-MOD'      / Attenuation of Xrays in Copper
GRPID1  =                   -1 / EXTVER of Group containing this HDU
GRPLC1  = 'file:///home/njw/jsim/dat/jemx_data.fits' / URL of file containing Gr
***** END OF FILE *****
```

JEMX-ATMO-MOD

```
*********************************************
XTENSION= 'BINTABLE'           / binary table extension
BITPIX  =                    8 / 8-bit bytes
NAXIS   =                    2 / 2-dimensional binary table
NAXIS1  =                    8 / width of table in bytes
NAXIS2  =                   74 / number of rows in table
PCOUNT  =                    0 / size of special data area
GCOUNT  =                    1 / one data group (required keyword)
TFIELDS =                    2 / number of fields in each row
TTYPE1  = 'Energy  '           / label for field   1
TFORM1  = 'E       '           / data format of field: 4-byte REAL
TUNIT1  = 'keV     '           / physical unit of field
TTYPE2  = 'att_len '           / label for field   2
TFORM2  = 'E       '           / data format of field: 4-byte REAL
TUNIT2  = '/mm     '           / physical unit of field
SFILENAM= '/home/njw/jemx_sim/dat/att_len_mo.dat'
COMMENT  //
COMMENT  // Attenuation length in Molybdenum in /mm
COMMENT  // Derived from 'abs_coeff.mo'
COMMENT  //
HISTORY   This FITS file was created by the FCREATE task.
HISTORY   fcreate3.0d at 1/5/98 16:25:19
EXTNAME = 'JEMX-ATMO-MOD'      / Attenuation of Xrays in Molybdenum
GRPID1  =                   -1 / EXTVER of Group containing this HDU
GRPLC1  = 'file:///home/njw/jsim/dat/jemx_data.fits' / URL of file containing Gr
***** END OF FILE *****
```

13 Makefiles

A Makefile has been constructed for each executable. An example (for JSIMraytrace) is reproduced below. The choice of the 'cc' compiler is recommended for working conditions. The 'gcc' compiler gives some very useful comments on the code e.g. about variables never used and ANSI C compatibility.

```
# Makefile for JRaytrace
#
# Author: Carol Anne Oxborrow, DSRI Copenhagen.
# oxborrow@dsri.dk
#
# Version: 1.0
#
# Date: 18/2/1999
#
# Function: To perform compilation and library building for
# JEMX ISSW executables that use ISDC libraries,
# common, dal, ril, pil and also cfitsio
# Change Log:
# 1999/02/18 First release corresponds to first installation of
# ISDC libraries on tenma disk t.
# 1999/05/06 Update by N.J. Westergaard
#

# Name of your executable:
PROGRAM_NAME = jraytrace
EXECUTABLE_NAME = JRaytrace

# Name and version of individual function being compiled for library:
# (Used in ‘make function’ only)
FUNCTION_NAME = getjemxarrshort

# Directory containing your own /lib and /include directories, which contain
# your personal headers and library (libJExecutable.a). These are the
# defaults if you are working on the DSRI system
ISSW_DIR = /t/jemx/isdc/issw

# Directory contain the installation of the ISDC libraries and headers:
# Do not change this if you’re running of the DSRI system
ISDC_DIR = /t/jemx/isdc

# Directory naming conventions used on the DSRI system:
LIBRARY = lib
INCLUDE = include
RANLIB = ranlib
RM = /bin/rm

# Put all your function file names here (not executables)
SOURCE_FILES = getexposuretime.c \
jsimitoan.c \
jsimreverse.c \
retrievephoton.c \
```

```
stoweevent.c \
getjemxarrshort.c \
getnextautoname.c \
gethexcell.c \
getjemxparmreal.c


# Put all the names of the respectives function object files here
OBJECT_FILES = getexposuretime.o \
jsimitoan.o \
jsimreverse.o \
retrievephoton.o \
stoweevent.o \
getjemxarrshort.o \
getnextautoname.o \
gethexcell.o \
getjemxparmreal.o

# This target recompiles your current executable and links it to the ISDC libraries
# and the executable's own library and header. The compiled program is then run.

executable: ${PROGRAM_NAME}.c  ${PROGRAM_NAME}.h
cp ${PROGRAM_NAME}.h ${ISSW_DIR}/${INCLUDE}
cc  -g \
 -I${ISDC_DIR}/${INCLUDE} -I${ISSW_DIR}/${INCLUDE} ${PROGRAM_NAME}.c \
 -L${ISDC_DIR}/${LIBRARY} -L${ISSW_DIR}/${LIBRARY} -lsocket  -l${PROGRAM_NAME} \
 -lnsl -lm -ldal3gen -lcommon -lril -lpil -ldal -lcfitsio
mv a.out ${EXECUTABLE_NAME}
chmod +x ${EXECUTABLE_NAME}
# ${EXECUTABLE_NAME}


# This target takes a single function for the executable, compiles it and adds
#it to (or replaces it in) the executable's library.

function: ${FUNCTION_NAME}${FUNCTION_VERSION}.c
cp ${PROGRAM_NAME}.h ${ISSW_DIR}/${INCLUDE}
cc  -c -g   -I${ISDC_DIR}/${INCLUDE} -I${ISSW_DIR}/${INCLUDE} ${FUNCTION_NAME}.c
ar -r ${PROGRAM_NAME}.a ${FUNCTION_NAME}${FUNCTION_VERSION}.o
${RANLIB} ${PROGRAM_NAME}.a
ar -t ${PROGRAM_NAME}.a
cp ${PROGRAM_NAME}.a  ${ISSW_DIR}/${LIBRARY}/lib${PROGRAM_NAME}.a
echo Updated version of library installed in ${ISSW_DIR}/${LIBRARY}

# This target compiles individual function object code without adding it to
# a library

object-files:
cp ${PROGRAM_NAME}.h ${ISSW_DIR}/${INCLUDE}
cc  -c -g   -I${ISDC_DIR}/${INCLUDE} -I${ISSW_DIR}/${INCLUDE} ${SOURCE_FILES}

# This target makes an all-new version of the executable's library, moving the
```

```
# previous version to <Library_Name>.a.old

library:
cp ${PROGRAM_NAME}.h ${ISSW_DIR}/${INCLUDE}
cc  -c -g   -I${ISDC_DIR}/${INCLUDE} -I${ISSW_DIR}/${INCLUDE} ${SOURCE_FILES}
ar -r ${PROGRAM_NAME}.a ${OBJECT_FILES}
${RANLIB} ${PROGRAM_NAME}.a
ar -t ${PROGRAM_NAME}.a
cp ${PROGRAM_NAME}.a  ${ISSW_DIR}/${LIBRARY}/lib${PROGRAM_NAME}.a
echo All-new version of library installed in ${ISSW_DIR}/${LIBRARY}
clean:
rm *.o

clean-libraries:
rm ${ISSW_DIR}/${LIBRARY}/lib${PROGRAM_NAME}.a
rm ${PROGRAM_NAME}.a
```

14 COMMENTS

JSIMraytrace(990507): there may be problems when the number of photons is a multiple of the MAX_BUFFER_SIZE.

JSIMmakeSkyImage(990701): there is a problem with an image rotation of 120 deg probably due to the pattern symmetry. Where does it occur?

15 Miscellanous information

At DSRI the Makefile information is:

ISSW_DIR = /t/jemx/isdc/issw
ISDC_DIR = /t/jemx/isdc

At ISDC one should write:

ISSW_DIR = (any directory where you have write access)
ISDC_DIR = /isdc/dev

Index