MIT Kavli Institute **Chandra X-Ray Center**

MEMORANDUM
September 28, 2016

To: Jonathan McDowell, SDS Group Leader
From: Glenn E. Allen, SDS
Subject: `acis_process_events` spec
Revision: 4.16
URL: http://space.mit.edu/CXC/docs/docs.html#ape
File: /nfs/inconceivable/d0/sds/specs/acis_process_events/ape_spec_4.16.tex

1 acis_process_events

This spec, which is incomplete, describes some of processing steps for continuous-clocking mode data.

1.1 Description

1.2 Input

1.3 Output

1.4 Parameters

1.5 Processing

1.5.1 Error checking

The following steps are performed once prior to the processing of the data for each input ACIS event.

1. `obsfile`:
   (a) Validation:
      i. If
      $$\texttt{obsfile} \quad = \quad \text{NONE}, \tag{1}$$
      then `obsfile` is changed to "none".
      ii. Existence:
      If
      $$\texttt{obsfile} \quad \neq \quad \text{none} \tag{2}$$
      and the `obsfile` does not exist, then `obsfile` is changed to "none" and `acis_process_events` produces a warning message.

iii. Permission:
If

$$\texttt{obsfile} \neq \text{none} \tag{3}$$

and the file permissions do not allow the `obsfile` to be read, then `obsfile` is changed to "none" and `acis_process_events` produces a warning message.

iv. `OBS_MODE`:
If

$$\texttt{obsfile} \neq \text{none} \tag{4}$$

then

A. If the `obsfile` does not include the keyword `OBS_MODE`, then $\texttt{OBS\_MODE}_{\text{in}}$ is set to "none".

B. If the `obsfile` includes the keyword `OBS_MODE` then $\texttt{OBS\_MODE}_{\text{in}}$ is set to `OBS_MODE`.

C. If

$$\texttt{OBS\_MODE}_{\text{in}} = \text{POINTING}, \tag{5}$$

then $\texttt{OBS\_MODE}_{\text{in}}$ is set to "pointing".

D. If

$$\texttt{OBS\_MODE}_{\text{in}} = \text{SECONDARY}, \tag{6}$$

then $\texttt{OBS\_MODE}_{\text{in}}$ is set to "secondary".

E. If

$$\texttt{OBS\_MODE}_{\text{in}} \neq \text{none and} \tag{7}$$

$$\texttt{OBS\_MODE}_{\text{in}} \neq \text{pointing and} \tag{8}$$

$$\texttt{OBS\_MODE}_{\text{in}} \neq \text{secondary}, \tag{9}$$

then $\texttt{OBS\_MODE}_{\text{in}}$ is set to "none" and `acis_process_events` produces a warning message.

2. `acaofffile`:

(a) Validation:
If

$$\texttt{OBS\_MODE}_{\text{in}} = \text{pointing}, \tag{10}$$

then

i. If

$$\texttt{acaofffile} = \text{NONE}, \tag{11}$$

then `acaofffile` is set to "none."

ii. Setting:
If

$$\texttt{acaofffile} = \text{none}, \tag{12}$$

then `acis_process_events` produces a warning message.

iii. Existence:
If

$$\texttt{acaofffile} \neq \text{none} \tag{13}$$

and the acaofffile does not exist, then acis_process_events produces a warning and acaofffile is set to "none."

iv. Permission:
If the acaofffile exists and the file permissions do not allow it to be read, then acis_process_events produces a warning message and acaofffile is set to "none."

v. CONTENT:
If the acaofffile does not have an HDU h_{acaoff} with the keyword

$$\texttt{CONTENT} = \text{ASPSOL or} \tag{14}$$

$$\texttt{CONTENT} = \text{OBCSOL,} \tag{15}$$

then acis_process_events produces a warning message and acaofffile is set to "none."

vi. Keyword:
If HDU h_{acaoff} of the acaofffile does not include the keyword TSTART, then acis_process_events produces a warning message and acaofffile is set to "none."

vii. Columns:
If HDU h_{acaoff} of the acaofffile does not include the columns TIME, RA, DEC, and ROLL then acis_process_events produces a warning message and acaofffile is set to "none." Hereafter, these columns are referred to as $\texttt{TIME}_{\text{acaoff}}$, $\texttt{RA}_{\text{acaoff}}$, $\texttt{DEC}_{\text{acaoff}}$, and $\texttt{ROLL}_{\text{acaoff}}$.

viii. Sequential:
If more than one valid acaofffile is specified and the the values TSTART are not in increasing order, then acis_process_events produces a warning message and acaofffile is set to "none."

3. alignmentfile:

(a) Validation:
If

$$\texttt{OBS\_MODE}_{\text{in}} = \text{pointing,} \tag{16}$$

then

i. If

$$\texttt{alignmentfile} = \text{NONE,} \tag{17}$$

then alignmentfile is changed to "none."

ii. Setting:
If

$$\texttt{alignmentfile} = \text{none,} \tag{18}$$

then acis_process_events produces a warning message.

iii. Existence:
If

$$\texttt{alignmentfile} \neq \text{none} \tag{19}$$

and the alignmentfile does not exist, then acis_process_events produces a warning message and alignmentfile is set to "none.".

iv. Permission:
If the `alignmentfile` exists and the file permissions do not allow it to be read, then `acis_process_events` produces a warning message and `alignmentfile` is set to "none.".

v. `CONTENT`:
If the `alignmentfile` does not have an HDU $h_{\text{alignment}}$ with the keyword

$$\texttt{CONTENT} = \text{ASPSOL or} \quad (20)$$
$$\texttt{CONTENT} = \text{OBCSOL,} \quad (21)$$

then `acis_process_events` produces a warning message and `alignmentfile` is set to "none.".

vi. Keyword:
If HDU $h_{\text{alignment}}$ of the `alignmentfile` does not include the keyword `TSTART`, then `acis_process_events` produces a warning message and `alignmentfile` is set to "none.".

vii. Columns:
If HDU $h_{\text{alignment}}$ of the `alignmentfile` does not include the columns `DY`, `DZ`, and `DTHETA` then `acis_process_events` produces a warning message and `alignmentfile` is set to "none.".

viii. Sequential:
If more than one valid `alignmentfile` is specified and the values `TSTART` are not in increasing order, then `acis_process_events` produces a warning message and `alignmentfile` is set to "none.".

4. `infile`:

(a) Existence:
If the `infile` does not exist, then `acis_process_events` exits with an error message.

(b) Permission:
If the `infile` exists and the file permissions do not allow it to be read, then `acis_process_events` exits with an error message.

(c) Validation:

i. `OBS_MODE`:
If $\texttt{OBS\_MODE}_{\text{in}}$ = none and HDU h_{in} of the `infile` includes the keyword `OBS_MODE`, then

A. $\texttt{OBS\_MODE}_{\text{in}}$ is set to `OBS_MODE`.

B. If

$$\texttt{OBS\_MODE}_{\text{in}} = \text{POINTING,} \quad (22)$$

then $\texttt{OBS\_MODE}_{\text{in}}$ is set to "pointing".

C. If

$$\texttt{OBS\_MODE}_{\text{in}} = \text{SECONDARY,} \quad (23)$$

then $\texttt{OBS\_MODE}_{\text{in}}$ is set to "secondary".

D. If

$$\texttt{OBS\_MODE}_{\text{in}} \neq \text{none and} \quad (24)$$
$$\texttt{OBS\_MODE}_{\text{in}} \neq \text{pointing and} \quad (25)$$
$$\texttt{OBS\_MODE}_{\text{in}} \neq \text{secondary,} \quad (26)$$

then $\texttt{OBS\_MODE}_{\text{in}}$ is set to "none" and `acis_process_events` produces a warning message.

ii. DATAMODE:
The DATAMODE is read from the HDU $h_{\rm in}$ keyword of the same name. If the HDU $h_{\rm in}$ does not include the keyword DATAMODE or if

$$\begin{aligned}
\texttt{DATAMODE} &\neq \text{CC33_FAINT and} && (27)\\
\texttt{DATAMODE} &\neq \text{CC33_GRADED and} && (28)\\
\texttt{DATAMODE} &\neq \text{FAINT and} && (29)\\
\texttt{DATAMODE} &\neq \text{FAINT_BIAS and} && (30)\\
\texttt{DATAMODE} &\neq \text{GRADED and} && (31)\\
\texttt{DATAMODE} &\neq \text{VFAINT,} && (32)
\end{aligned}$$

then acis_process_events exits with an error message. Hereafter, the value of this keyword is referred to as $\texttt{DATAMODE}_{\rm in}$.

iii. CONTENT:
If the infile does not have an HDU $h_{\rm in}$ with the keyword

$$\begin{aligned}
\texttt{CONTENT} &= \text{EVT0 or} && (33)\\
\texttt{CONTENT} &= \text{EVT1 or} && (34)\\
\texttt{CONTENT} &= \text{TGEVT1 or} && (35)\\
\texttt{CONTENT} &= \text{EVT2,} && (36)
\end{aligned}$$

then acis_process_events exits with an error message. Hereafter, the value of this keyword is referred to as $\texttt{CONTENT}_{\rm in}$.

iv. TIME:
If

$$\begin{aligned}
\texttt{DATAMODE}_{\rm in} &= \text{CC33_FAINT or} && (37)\\
\texttt{DATAMODE}_{\rm in} &= \text{CC33_GRADED} && (38)
\end{aligned}$$

and HDU $h_{\rm in}$ of the infile does not include the column TIME, then acis_process_events exits with an error message. Hereafter, this column is referred to as $\texttt{TIME}_{\rm in}$.

v. TIME_RO:
If

$$\begin{aligned}
\texttt{DATAMODE}_{\rm in} &= \text{CC33_FAINT or} && (39)\\
\texttt{DATAMODE}_{\rm in} &= \text{CC33_GRADED} && (40)
\end{aligned}$$

and

$$\begin{aligned}
\texttt{CONTENT}_{\rm in} &= \text{EVT1 or} && (41)\\
\texttt{CONTENT}_{\rm in} &= \text{TGEVT1 or} && (42)\\
\texttt{CONTENT}_{\rm in} &= \text{EVT2} && (43)
\end{aligned}$$

and HDU $h_{\rm in}$ of the infile does not include the column TIME_RO, then acis_process_events exits with an error message. Hereafter, this column is referred to as $\texttt{TIME\_RO}_{\rm in}$.

vi. CCD_ID:
A. If

$$\texttt{CONTENT}_{\rm in} = \text{EVT0} \qquad (44)$$

and HDU $h_{\rm in}$ of the infile does not include the keyword CCD_ID, then acis_process_events exits with an error message. Hereafter, this keyword is referred to as $\texttt{CCD\_ID}_{\rm in}$.

B. If

$$\texttt{CONTENT}_{\rm in} = \text{EVT1 or} \tag{45}$$
$$\texttt{CONTENT}_{\rm in} = \text{TGEVT1 or} \tag{46}$$
$$\texttt{CONTENT}_{\rm in} = \text{EVT2} \tag{47}$$

and HDU $h_{\rm in}$ of the `infile` does not include the column `CCD_ID`, then `acis_process_events` exits with an error message. Hereafter, this column is referred to as $\texttt{CCD\_ID}_{\rm in}$.

vii. `CCDX`:

A. If

$$\texttt{CONTENT}_{\rm in} = \text{EVT0} \tag{48}$$

and HDU $h_{\rm in}$ of the `infile` does not include the column `CCDX` and does not include the column `CHIPX`, then `acis_process_events` exits with an error message. Hereafter, these columns are referred to as $\texttt{CCDX}_{\rm in}$ and $\texttt{CHIPX}_{\rm in}$, respectively.

viii. `CHIPX`:

A. If

$$\texttt{CONTENT}_{\rm in} = \text{EVT1 or} \tag{49}$$
$$\texttt{CONTENT}_{\rm in} = \text{TGEVT1 or} \tag{50}$$
$$\texttt{CONTENT}_{\rm in} = \text{EVT2} \tag{51}$$

and HDU $h_{\rm in}$ of the `infile` does not include the column `CHIPX`, then `acis_process_events` exits with an error message. Hereafter, this column is referred to as $\texttt{CHIPX}_{\rm in}$.

ix. `CCDY`:

A. If

$$\texttt{CONTENT}_{\rm in} = \text{EVT0} \tag{52}$$

and

$$\texttt{DATAMODE}_{\rm in} = \text{FAINT or} \tag{53}$$
$$\texttt{DATAMODE}_{\rm in} = \text{FAINT\_BIAS or} \tag{54}$$
$$\texttt{DATAMODE}_{\rm in} = \text{GRADED or} \tag{55}$$
$$\texttt{DATAMODE}_{\rm in} = \text{VFAINT} \tag{56}$$

and HDU $h_{\rm in}$ of the `infile` does not include the column `CCDY` and does not include the column `CHIPY`, then `acis_process_events` exits with an error message. Hereafter, these columns are referred to as $\texttt{CCDY}_{\rm in}$ and $\texttt{CHIPY}_{\rm in}$, respectively.

x. `TROW`:

A. If

$$\texttt{CONTENT}_{\rm in} = \text{EVT0} \tag{57}$$

and

$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_FAINT or} \tag{58}$$
$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_GRADED} \tag{59}$$

and HDU $h_{\rm in}$ of the `infile` does not include the column `TROW` and does not include the column `CHIPY`, then `acis_process_events` exits with an error message. Hereafter, these columns are referred to as $\texttt{TROW}_{\rm in}$ and $\texttt{CHIPY}_{\rm in}$, respectively.

xi. CHIPY:

A. If

$$\texttt{CONTENT}_{\rm in} = \text{EVT1 or} \tag{60}$$
$$\texttt{CONTENT}_{\rm in} = \text{TGEVT1 or} \tag{61}$$
$$\texttt{CONTENT}_{\rm in} = \text{EVT2} \tag{62}$$

and HDU $h_{\rm in}$ of the infile does not include the column CHIPY, then acis_process_events exits with an error message. Hereafter, this column is referred to as $\texttt{CHIPY}_{\rm in}$.

xii. TIMEDEL:

If

$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_FAINT or} \tag{63}$$
$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_GRADED} \tag{64}$$

and HDU $h_{\rm in}$ of the infile does not include the keyword TIMEDEL, then acis_process_events exits with an error message. Hereafter this keyword is referred to as $\texttt{TIMEDEL}_{\rm in}$.

xiii. RA_TARG, DEC_TARG, RA_NOM, DEC_NOM, RA_PNT, DEC_PNT, CHIPY_TG, and TG_M:

If

$$\texttt{OBS\_MODE}_{\rm in} = \text{pointing} \tag{65}$$

and

$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_FAINT or} \tag{66}$$
$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_GRADED,} \tag{67}$$

then

A. RA_TARG, DEC_TARG, RA_NOM, DEC_NOM, RA_PNT, DEC_PNT:
If HDU $h_{\rm in}$ of the infile does not include the keywords RA_TARG, DEC_TARG, RA_NOM, DEC_NOM, RA_PNT, and DEC_PNT, then acis_process_events exits with an error message. Hereafter these keywords are referred to as $\texttt{RA\_TARG}_{\rm in}$, $\texttt{DEC\_TARG}_{\rm in}$, $\texttt{RA\_NOM}_{\rm in}$, $\texttt{DEC\_NOM}_{\rm in}$, $\texttt{RA\_PNT}_{\rm in}$, and $\texttt{DEC\_PNT}_{\rm in}$, respectively.

B. CHIPY_TG and TG_M:
If

$$\texttt{CONTENT}_{\rm in} = \text{TGEVT1} \tag{68}$$

and HDU $h_{\rm in}$ of the infile does not include the columns CHIPY_TGand TG_M, then acis_process_events exits with an error message. Hereafter these columns are referred to as $\texttt{CHIPY\_TG}_{\rm in}$ and $\texttt{TG\_M}_{\rm in}$, respectively.

xiv. TGAINCOR:

If HDU $h_{\rm in}$ of the infile does not include the keyword TGAINCOR, then this keyword is set to zero (i.e. FALSE). Hereafter, this keyword is referred to as $\texttt{TGAINCOR}_{\rm in}$.

5. stop:

(a) Lowercase:
The parameter string is converted to contain only lower case letters.

(b) Validation:

i. Setting:
If

$$\begin{aligned} \texttt{stop} &\neq \text{none and} \quad (69)\\ \texttt{stop} &\neq \text{chip and} \quad (70)\\ \texttt{stop} &\neq \text{tdet and} \quad (71)\\ \texttt{stop} &\neq \text{det and} \quad (72)\\ \texttt{stop} &\neq \text{tan and} \quad (73)\\ \texttt{stop} &\neq \text{sky,} \quad (74) \end{aligned}$$

then `stop` is changed to "none" and `acis_process_events` produces a warning message.

ii. `OBS_MODE`:
If

$$\begin{aligned} \texttt{OBS\_MODE}_{\text{in}} &\neq \text{pointing and} \quad (75)\\ \texttt{stop} &\neq \text{none and} \quad (76)\\ \texttt{stop} &\neq \text{chip and} \quad (77)\\ \texttt{stop} &\neq \text{tdet,} \quad (78) \end{aligned}$$

then `stop` is changed to "tdet" and `acis_process_events` produces a warning message.

iii. `acaofffile`:
If

$$\begin{aligned} \texttt{OBS\_MODE}_{\text{in}} &= \text{pointing and} \quad (79)\\ \texttt{acaofffile} &= \text{none and} \quad (80)\\ \texttt{stop} &\neq \text{none and} \quad (81)\\ \texttt{stop} &\neq \text{chip and} \quad (82)\\ \texttt{stop} &\neq \text{tdet,} \quad (83) \end{aligned}$$

then `acis_process_events` produces a warning message and `stop` is changed to "none."

iv. `alignmentfile`:
If

$$\begin{aligned} \texttt{OBS\_MODE}_{\text{in}} &= \text{pointing and} \quad (84)\\ \texttt{alignmentfile} &= \text{none and} \quad (85)\\ \texttt{stop} &\neq \text{none and} \quad (86)\\ \texttt{stop} &\neq \text{chip and} \quad (87)\\ \texttt{stop} &\neq \text{tdet,} \quad (88) \end{aligned}$$

then `acis_process_events` produces a warning message and `stop` is changed to "none."

6. `doevtgrade`:

(a) Lowercase:
The parameter string is converted to contain only lower case letters.

(b) Validation:

i. Setting:
If

$$\begin{aligned} \texttt{doevtgrade} &\neq \text{yes and} \quad (89)\\ \texttt{doevtgrade} &\neq \text{no,} \quad (90) \end{aligned}$$

then `acis_process_events` exits with an error message.

ii. Timed mode:
If

$$\begin{aligned}
\texttt{DATAMODE}_{\rm in} &= \text{FAINT or} && (91)\\
\texttt{DATAMODE}_{\rm in} &= \text{FAINT_BIAS or} && (92)\\
\texttt{DATAMODE}_{\rm in} &= \text{GRADED or} && (93)\\
\texttt{DATAMODE}_{\rm in} &= \text{VFAINT} && (94)
\end{aligned}$$

and HDU $h_{\rm in}$ of the `infile` does not include the column `PHAS`, then `doevtgrade` is changed to "no" and `acis_process_events` produces a warning message.

iii. Continuous clocking mode:
If

$$\begin{aligned}
\texttt{DATAMODE}_{\rm in} &= \text{CC33_GRADED or} && (95)\\
\texttt{DATAMODE}_{\rm in} &= \text{GRADED} && (96)
\end{aligned}$$

and HDU $h_{\rm in}$ of the `infile` does not include the column `PHA_RO`, then `doevtgrade` is changed to "no" and `acis_process_events` produces a warning message.

7. `apply_cti`:

(a) Lowercase:
The parameter string is converted to contain only lower case letters.

(b) Validation:

i. Setting:
If

$$\begin{aligned}
\texttt{apply_cti} &\neq \text{yes and} && (97)\\
\texttt{apply_cti} &\neq \text{no,} && (98)
\end{aligned}$$

then `acis_process_events` exits with an error message.

ii. Timed mode:
If

$$\begin{aligned}
\texttt{DATAMODE}_{\rm in} &= \text{FAINT or} && (99)\\
\texttt{DATAMODE}_{\rm in} &= \text{FAINT_BIAS or} && (100)\\
\texttt{DATAMODE}_{\rm in} &= \text{GRADED or} && (101)\\
\texttt{DATAMODE}_{\rm in} &= \text{VFAINT} && (102)
\end{aligned}$$

and

$$\texttt{apply\_cti} = \text{yes} \qquad (103)$$

and the `infile` does not include the columns `CCD_ID`, `CHIPX`, `CHIPY`, and `PHAS`, then `apply_cti` is changed to "no" and `acis_process_events` produces a warning message.

iii. Continuous clocking mode:
If

$$\begin{aligned}
\texttt{DATAMODE}_{\rm in} &= \text{CC33_GRADED or} && (104)\\
\texttt{DATAMODE}_{\rm in} &= \text{GRADED} && (105)
\end{aligned}$$

and

$$\texttt{apply\_cti} = \text{yes} \qquad (106)$$

and the `infile` does not include the columns `CCD_ID`, `CHIPX`, `CHIPY`, `PHA_RO`, and `FLTGRADE`, then `apply_cti` is changed to "no" and `acis_process_events` produces a warning message.

iv. `doevtgrade`:
If

$$\texttt{apply\_cti} = \text{yes and} \tag{107}$$
$$\texttt{doevtgrade} = \text{no,} \tag{108}$$

then `apply_cti` is changed to “no” and `acis_process_events` produces a warning message.

8. `apply_tgain`:

(a) Lowercase:
The parameter string is converted to contain only lower case letters.

(b) Validation:

i. Setting:
If

$$\texttt{apply\_tgain} \neq \text{yes and} \tag{109}$$
$$\texttt{apply\_tgain} \neq \text{no,} \tag{110}$$

then `acis_process_events` exits with an error message.

ii. `TIME`:
If

$$\texttt{apply\_tgain} = \text{yes} \tag{111}$$

and the `infile` does not include the column `TIME`, then `apply_tgain` is changed to “no” and `acis_process_events` produces a warning message.

iii. `doevtgrade`:
If

$$\texttt{apply\_tgain} = \text{yes and} \tag{112}$$
$$\texttt{doevtgrade} = \text{no,} \tag{113}$$

then `apply_tgain` is changed to “no” and `acis_process_events` produces a warning message.

9. `gradefile`:

(a) Validation:

i. If

$$\texttt{gradefile} = \text{NONE,} \tag{114}$$

then `gradefile` is changed to “none.”

ii. Existence:
If

$$\texttt{gradefile} \neq \text{none} \tag{115}$$

and the `gradefile` does not exist, then `gradefile` is changed to “none” and `acis_process_events` produces a warning message.

iii. Permission:
If

$$\texttt{gradefile} \neq \text{none} \tag{116}$$

and the file permissions do not allow it to be read, then `gradefile` is changed to "none" and `acis_process_events` produces a warning message.

iv. HDU:
If the `gradefile` does not have an HDU that includes the keyword `CBD10001` and where the keyword includes $\texttt{DATAMODE}_{\text{in}}$, then `gradefile` is changed to "none" and `acis_process_events` produces a warning message. Hereafter this HDU is referred to as h_{grade}.

v. Columns:
If

$$\texttt{gradefile} \neq \text{none} \tag{117}$$

and HDU h_{grade} of the `gradefile` does not include the columns `FLTGRADE` and `GRADE`, then `gradefile` is changed to "none" and `acis_process_events` produces a warning message. Hereafter these columns are referred to as $\texttt{FLTGRADE}_{\text{grade}}$ and $\texttt{GRADE}_{\text{grade}}$, respectively.

10. `grade_image_file`:

(a) Validation:
If

$$\texttt{grade\_image\_file} \neq \text{caldb and} \tag{118}$$
$$\texttt{grade\_image\_file} \neq \text{CALDB and} \tag{119}$$
$$\texttt{apply\_cti} = \text{yes} \tag{120}$$

and

$$\texttt{DATAMODE} = \text{CC33\_GRADED or} \tag{121}$$
$$\texttt{DATAMODE} = \text{GRADED,} \tag{122}$$

then

i. Existence:
If the `grade_image_file` does not exist, then `apply_cti` is changed to "no" and `acis_process_events` produces a warning message.

ii. Permission:
If the `grade_image_file` exists and the file permissions do not allow it to be read, then `apply_cti` is changed to "no" and `acis_process_events` produces a warning message.

iii. `CONTENT`:
If the `grade_image_file` does not have one or more HDU h_{grdimg} with the keyword

$$\texttt{CONTENT} = \text{GRADED\_CTI,} \tag{123}$$

then `apply_cti` is changed to "no" and `acis_process_events` produces a warning message.

iv. Columns:
If HDU h_{grdimg} of the `grade_image_file` does not include the columns `FLTGRADE`, `GRDIMG`, and `ESCL`, then `apply_cti` is changed to "no" and `acis_process_events` produces a warning message.

11. `badpixfile`:

(a) Validation:

i. If

$$\texttt{badpixfile} = \text{NONE}, \quad (124)$$

then `badpixfile` is changed to "none."

ii. Existence:
If

$$\texttt{badpixfile} \neq \text{none} \quad (125)$$

and the `badpixfile` does not exist, then `badpixfile` is changed to "none" and `acis_process_events` produces a warning message.

iii. Permission:
If

$$\texttt{badpixfile} \neq \text{none} \quad (126)$$

and the file permissions do not allow it to be read, then `badpixfile` is changed to "none" and `acis_process_events` produces a warning message.

iv. `CONTENT`:
If

$$\texttt{badpixfile} \neq \text{none} \quad (127)$$

and the `badpixfile` does not have one or more HDU h_{badpix} with the keyword

$$\texttt{CONTENT} = \text{BADPIX or} \quad (128)$$
$$\texttt{CONTENT} = \text{CDB\_ACIS\_BADPIX}, \quad (129)$$

then `badpixfile` is changed to "none" and `acis_process_events` produces a warning message.

v. Keyword:
If

$$\texttt{badpixfile} \neq \text{none} \quad (130)$$

and the HDU(s) h_{badpix} of the `badpixfile` do not include the keyword `CCD_ID`, then `badpixfile` is changed to "none" and `acis_process_events` produces a warning message. Hereafter this keyword is referred to as $\texttt{CCD\_ID}_{\text{badpix}}$.

vi. Columns:
If

$$\texttt{badpixfile} \neq \text{none} \quad (131)$$

and the HDU(s) h_{badpix} of the `badpixfile` do not include the columns `CHIPX`, `CHIPY`, `TIME`, `TIME_STOP`, and `STATUS`, then `badpixfile` is changed to "none" and `acis_process_events` produces a warning message. Hereafter these columns are referred to as $\texttt{CHIPX}_{\text{badpix}}$, $\texttt{CHIPY}_{\text{badpix}}$, $\texttt{TIME}_{\text{badpix}}$, $\texttt{TIME\_STOP}_{\text{badpix}}$, and $\texttt{STATUS}_{\text{badpix}}$, respectively.

12. `ctifile`:

(a) Validation:
If

$$\texttt{ctifile} \neq \text{caldb}, \quad (132)$$
$$\texttt{ctifile} \neq \text{CALDB, and} \quad (133)$$
$$\texttt{apply\_cti} = \text{yes}, \quad (134)$$

then

i. Existence:
If the `ctifile` does not exist, then `apply_cti` is changed to “no” and `acis_process_events` produces a warning message.

ii. Permission:
If the `ctifile` exists and the file permissions do not allow it to be read, then `apply_cti` is changed to “no” and `acis_process_events` produces a warning message.

iii. `CONTENT`:
If the `ctifile` does not have one or more HDU h_{cti} with the keyword

$$\texttt{CONTENT} = \text{CDB\_ACIS\_CTI}, \quad (135)$$

then `apply_cti` is changed to “no” and `acis_process_events` produces a warning message.

iv. Columns:
If HDU h_{cti} of the `ctifile` does not include the columns `CCD_ID`, `CHIPX_LO`, `CHIPX_HI`, `CHIPY_LO`, `CHIPY_HI`, `PHA`, `VOLUME_X`, `VOLUME_Y`, `FRCTRLX`, `FRCTRLY`, `TCTIX`, and `TCTIY`, then `apply_cti` is changed to “no” and `acis_process_events` produces a warning message.

13. `tgainfile`:

(a) Validation:
If

$$\texttt{tgainfile} \neq \text{caldb}, \quad (136)$$
$$\texttt{tgainfile} \neq \text{CALDB, and} \quad (137)$$
$$\texttt{apply\_tgain} = \text{yes}, \quad (138)$$

then

i. Existence:
If the `tgainfile` does not exist, then `apply_tgain` is changed to “no” and `acis_process_events` produces a warning message.

ii. Permission:
If the `tgainfile` exists and the file permissions do not allow it to be read, then `apply_tgain` is changed to “no” and `acis_process_events` produces a warning message.

iii. `CONTENT`:
If the `tgainfile` does not have one or more HDU h_{tgain} with the keyword

$$\texttt{CONTENT} = \text{CDB\_ACIS\_TGAIN}, \quad (139)$$

then `apply_tgain` is changed to “no” and `acis_process_events` produces a warning message.

iv. Columns:
If HDU h_{tgain} of the `tgainfile` does not include the columns `CCD_ID`, `CHIPX_LO`, `CHIPX_HI`, `CHIPY_LO`, `CHIPY_HI`, `NPOINTS`, `PHA`, `DELTPHA1`, and `DELTPHA2`, then `apply_tgain` is changed to “no” and `acis_process_events` produces a warning message.

v. Keywords:
If HDU h_{tgain} of the `tgainfile` does not include the keywords `EPOCH1` and `EPOCH2`, then `apply_tgain` is changed to "no" and `acis_process_events` produces a warning message.

14. `mtlfile`:

(a) Lowercase:
If

$$\texttt{mtlfile} = \text{NONE}, \tag{140}$$

then change `mtlfile` to "none".

(b) Validation:
If

$$\texttt{mtlfile} \neq \text{none and} \tag{141}$$
$$\texttt{apply\_cti} = \text{yes}, \tag{142}$$

then

i. Existence:
If the `mtlfile` does not exist, then `mtlfile` is changed to "none" and `acis_process_events` produces a warning message.

ii. Permission:
If the `mtlfile` exists and the file permissions do not allow it to be read, then `mtlfile` is changed to "none" and `acis_process_events` produces a warning message.

iii. `CONTENT`:
If the `mtlfile` does not have one or more HDU h_{mtl} with the keyword

$$\texttt{CONTENT} = \text{mtl}, \tag{143}$$

then `mtlfile` is changed to "none" and `acis_process_events` produces a warning message.

iv. Columns:
If HDU h_{mtl} of the `mtlfile` does not include the columns `TIME` and `FP_TEMP`, then `mtlfile` is changed to "none" and `acis_process_events` produces a warning message.

15. `clobber`:

(a) Lowercase:
The parameter string is converted to contain only lower case letters.

(b) Validation:

i. Setting:
If

$$\texttt{clobber} \neq \text{yes and} \tag{144}$$
$$\texttt{clobber} \neq \text{no}, \tag{145}$$

then `clobber` is changed to "no" and `acis_process_events` produces a warning message.

ii. Permission:
If

$$\texttt{clobber} = \text{yes} \tag{146}$$

and the `outfile` exists and the file permissions of the `outfile` do not allow it to be overwritten, then `acis_process_events` exits with an error message.

iii. Don't overwrite:
If

$$\texttt{clobber} = \text{no} \tag{147}$$

and the `outfile` exists, then `acis_process_events` exits with an error message.

16. `pix_adj`:

(a) Lowercase:
The parameter string is converted to contain only lower case letters.

(b) Validation:

i. Setting:
If

$$\texttt{pix\_adj} \neq \text{centroid and} \tag{148}$$
$$\texttt{pix\_adj} \neq \text{edser and} \tag{149}$$
$$\texttt{pix\_adj} \neq \text{none and} \tag{150}$$
$$\texttt{pix\_adj} \neq \text{randomize,} \tag{151}$$

then `pix_adj` is changed to "none" and `acis_process_events` produces a warning message.

ii. `OBS_MODE`:
If

$$\texttt{OBS\_MODE}_{\text{in}} \neq \text{pointing and} \tag{152}$$
$$\texttt{pix\_adj} \neq \text{none,} \tag{153}$$

then `pix_adj` is changed to "none" and `acis_process_events` produces a warning message.

iii. `acaofffile`:
If

$$\texttt{pix\_adj} \neq \text{none} \tag{154}$$

and

$$\texttt{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \tag{155}$$
$$\texttt{DATAMODE}_{\text{in}} = \text{CC33\_GRADED} \tag{156}$$

and

$$\texttt{acaofffile} = \text{none,} \tag{157}$$

then `pix_adj` is changed to "none" and `acis_process_events` produces a warning message.

iv. `alignmentfile`:
If

$$\texttt{pix\_adj} \neq \text{none} \tag{158}$$

and

$$\texttt{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \tag{159}$$
$$\texttt{DATAMODE}_{\text{in}} = \text{CC33\_GRADED} \tag{160}$$

and

$$\texttt{alignmentfile} = \text{none,} \tag{161}$$

then `pix_adj` is changed to "none" and `acis_process_events` produces a warning message.

v. `stop`:
If

$$\texttt{pix\_adj} = \text{centroid or} \quad (162)$$
$$\texttt{pix\_adj} = \text{edser or} \quad (163)$$
$$\texttt{pix\_adj} = \text{randomize} \quad (164)$$

and

$$\texttt{stop} \neq \text{sky}, \quad (165)$$

then `pix_adj` is changed to "none" and `acis_process_events` produces a warning message.

vi. `PHAS`:
If

$$\texttt{pix\_adj} = \text{centroid} \quad (166)$$

and the `infile` does not include the column `PHAS`, then `pix_adj` is changed to "none" and `acis_process_events` produces a warning message.

vii. `FLTGRADE`:
If

$$\texttt{pix\_adj} = \text{edser} \quad (167)$$

and

$$\texttt{DATAMODE} = \text{CC33\_GRADED or} \quad (168)$$
$$\texttt{DATAMODE} = \text{GRADED} \quad (169)$$

and the `infile` does not include the column `FLTGRADE`, then `pix_adj` is changed to "none" and `acis_process_events` produces a warning message.

17. `subpixfile`:

(a) If

$$\texttt{pix\_adj} = \text{edser}, \quad (170)$$

then

i. Existence:
If the `subpixfile` does not exist, then `pix_adj` is changed to "none" and `acis_process_events` produces a warning message.

ii. Permission:
If the `subpixfile` exists and the file permissions do not allow it to be read, then `pix_adj` is changed to "none" and `acis_process_events` produces a warning message.

iii. Validation:

A. `CONTENT`:
If the `subpixfile` does not have one or more HDU h_{subpix} with the keyword

$$\texttt{CONTENT} = \text{AXAF\_SUBPIX}, \quad (171)$$

then `pix_adj` is changed to "none" and `acis_process_events` produces a warning message.

B. Keyword:
If the HDUs $h_{\rm subpix}$ of the `subpixfile` do not include the keyword `CCD_ID`, then `pix_adj` is changed to "none" and `acis_process_events` produces a warning message.

C. Columns:
If the HDUs $h_{\rm subpix}$ of the `subpixfile` do not include binary tables with the columns `FLTGRADE`, `NPOINTS`, `ENERGY`, `CHIPX_OFFSET`, and `CHIPY_OFFSET`, then `pix_adj` is changed to "none" and `acis_process_events` produces a warning message. Hereafter these columns are referred to as $\texttt{FLTGRADE}_{\rm subpix}$, $\texttt{NPOINTS}_{\rm subpix}$, $\texttt{ENERGY}_{\rm subpix}$, $\texttt{CHIPX\_OFFSET}_{\rm subpix}$, and $\texttt{CHIPY\_OFFSET}_{\rm subpix}$, respectively.

1.5.2 Initializations

1. Focal-point CCD:
If

$$\texttt{OBS\_MODE}_{\rm in} = \text{pointing} \tag{172}$$

and

$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_FAINT or} \tag{173}$$
$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_GRADED,} \tag{174}$$

then the values of $\texttt{RA\_PNT}_{\rm in}$ and $\texttt{DEC\_PNT}_{\rm in}$ are used to determine the `CCD_ID` associated with the focal point. Hereafter this value is referred to as $\texttt{CCD\_ID}_{\rm focus}$.*

2. $\texttt{TIME}_{\rm min}$ and $\texttt{TIME}_{\rm max}$:
If

$$\texttt{OBS\_MODE}_{\rm in} = \text{pointing and} \tag{175}$$
$$\texttt{acaofffile} \neq \text{none and} \tag{176}$$
$$\texttt{alignmentfile} \neq \text{none,} \tag{177}$$

then the `acaofffile` data are processed to determine the earliest and latest times for which there is aspect information:

$$\texttt{TIME}_{\rm min} = \min\left(\texttt{TIME}_{\rm acaoff}\right) \text{ and} \tag{178}$$
$$\texttt{TIME}_{\rm max} = \max\left(\texttt{TIME}_{\rm acaoff}\right). \tag{179}$$

3. $\texttt{CHIPX\_TARG}_{\rm acaoff,med}$, $\texttt{CHIPY\_TARG}_{\rm acaoff,med}$:
If

$$\texttt{OBS\_MODE}_{\rm in} = \text{pointing} \tag{180}$$

and

$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_FAINT or} \tag{181}$$
$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_GRADED} \tag{182}$$

and

$$\texttt{acaofffile} \neq \text{none and} \tag{183}$$
$$\texttt{alignmentfile} \neq \text{none,} \tag{184}$$

*The focal point is the location associated with the optical axis in the absence of dither. This location should not be confused with the aim point, which is the location illuminated by an undithered point source provided that the source is not offset from the target location.

then the values of $\texttt{CHIPX\_TARG}_{\mathrm{acaoff}}$ and $\texttt{CHIPY\_TARG}_{\mathrm{acaoff}}$ are computed from the values of RA_TARG and DEC_TARG using the orientation of the telescope (i.e. RA, DEC, and ROLL) and the orientation of the SIM (i.e. DY, DZ, and DTHETA) and the TIMEs in the acaofffile. The values of $\texttt{CHIPX\_TARG}_{\mathrm{acaoff}}$ and $\texttt{CHIPY\_TARG}_{\mathrm{acaoff}}$ are processed to obtain the median values:

$$\texttt{CHIPX\_TARG}_{\mathrm{acaoff,med}} = \mathrm{median}(\texttt{CHIPX\_TARG}_{\mathrm{acaoff}}) \text{ and} \tag{185}$$

$$\texttt{CHIPY\_TARG}_{\mathrm{acaoff,med}} = \mathrm{median}(\texttt{CHIPY\_TARG}_{\mathrm{acaoff}}). \tag{186}$$

1.5.3 Loop over events

The following steps are performed, in sequence, for each event.

1. STATUS:

   (a) Exists:
   If HDU h_{in} of the infile includes a 32-bit column named STATUS, then

   i. The values of the bits for an event are read from the infile.

   ii. The value of STATUS[k] is set to zero for bits $k = 1$–5, 14, 16–19, and 23 (of 0–31), bits that can be set by acis_process_events.

   iii. If

   $$\texttt{doevtgrade} = \text{yes,} \tag{187}$$

   then the value of STATUS[20], the other bit that can be set by acis_process_events, is set to zero.

   (b) Does not exist:
   If HDU h_{in} does not include a 32-bit column named STATUS, then

   i. A set of 32 bits are allocated for the event.

   ii. The values of the 32 bits are initialized to zero.

2. EXPNO:

   (a) Read:

   i. If HDU h_{in} of the infile includes the EXPNO, hereafter referred to as $\texttt{EXPNO}_{\mathrm{in}}$, then

   $$\texttt{EXPNO} = \texttt{EXPNO}_{\mathrm{in}}. \tag{188}$$

   ii. If HDU h_{in} of the infile does not include the EXPNO, then

   $$\texttt{EXPNO} = \text{NULL.} \tag{189}$$

   (b) Validation:
   If

   $$\texttt{EXPNO} \neq \text{NULL} \tag{190}$$

   and

   $$\texttt{EXPNO} < 0 \text{ or} \tag{191}$$

   $$\texttt{EXPNO} \geq 10^8, \tag{192}$$

   then acis_process_events produces a warning upon completion with a count of the total number of events for which equation 190 and either equation 191 or 192 is true. These conditions should not occur.

3. CCD_ID:

   (a) Read:
   The value of CCD_ID for an event is given by $\texttt{CCD\_ID}_{\rm in}$.

   (b) Validation:
   If

$$\begin{aligned} \texttt{CCD\_ID} &< 0 \text{ or} \quad (193) \\ \texttt{CCD\_ID} &> 9, \quad (194) \end{aligned}$$

   then acis_process_events exits with an error message because CCD_ID-dependent computations could fail if the value of CCD_ID is unphysical.

4. CHIPX:

   (a) Read:
      i. If the infile includes the column CHIPX, then the value of CHIPX for an event is given by

$$\texttt{CHIPX} = \texttt{CHIPX}_{\rm in}. \quad (195)$$

      ii. If the infile does not include the column CHIPX, then the value of CHIPX for an event is given by

$$\texttt{CHIPX} = \texttt{CCDX}_{\rm in} + 1. \quad (196)$$

5. CHIPX_ADJ:

   (a) Initialize:

$$\texttt{CHIPX\_ADJ} = \texttt{CHIPX}. \quad (197)$$

   (b) Validation:
      i. Unphysical:
      If

$$\texttt{CHIPX\_ADJ} < 1, \quad (198)$$

      then

$$\texttt{CHIPX\_ADJ} = 1. \quad (199)$$

      If

$$\texttt{CHIPX\_ADJ} > 1024, \quad (200)$$

      then

$$\texttt{CHIPX\_ADJ} = 1024. \quad (201)$$

      ii. Unexpected:
      If

$$\begin{aligned} \texttt{CHIPX\_ADJ} &= 1 \text{ or} \quad (202) \\ \texttt{CHIPX\_ADJ} &= 1024, \quad (203) \end{aligned}$$

      then acis_process_events sets STATUS bit 0 (of 0–31) to one and increments a counter for the number of unexpected CHIPX values. These values should not occur.

6. NODE_ID:

   (a) Calculate:
   The NODE_ID of an event is given by

$$\texttt{NODE\_ID} = \mathrm{int}\left(\frac{\texttt{CHIPX\_ADJ} - 1}{256}\right), \tag{204}$$

   where "int" means the integer portion of (i.e. truncate or round down) the quantity in parentheses.

7. CHIPY:

   (a) Read:

      i. If the infile includes the column CHIPY, then the value of CHIPY for an event is given by

$$\texttt{CHIPY} = \texttt{CHIPY}_{\mathrm{in}}. \tag{205}$$

      ii. If the infile does not include the column CHIPY and

$$\texttt{DATAMODE}_{\mathrm{in}} = \text{FAINT or} \tag{206}$$
$$\texttt{DATAMODE}_{\mathrm{in}} = \text{FAINT\_BIAS or} \tag{207}$$
$$\texttt{DATAMODE}_{\mathrm{in}} = \text{GRADED or} \tag{208}$$
$$\texttt{DATAMODE}_{\mathrm{in}} = \text{VFAINT}, \tag{209}$$

      then the value of CHIPY for an event is given by

$$\texttt{CHIPY} = \texttt{CCDY}_{\mathrm{in}} + 1. \tag{210}$$

      iii. If the infile does not include the column CHIPY and

$$\texttt{DATAMODE}_{\mathrm{in}} = \text{CC33\_FAINT or} \tag{211}$$
$$\texttt{DATAMODE}_{\mathrm{in}} = \text{CC33\_GRADED}, \tag{212}$$

      then the value of CHIPY for an event is given by

$$\texttt{CHIPY} = \texttt{TROW}_{\mathrm{in}} + 1. \tag{213}$$

8. CHIPY_ADJ:

   (a) Initialize:

$$\texttt{CHIPY\_ADJ} = \texttt{CHIPY}. \tag{214}$$

   (b) Validation:

      i. Unphysical:

         A. Minimum:
         If

$$\texttt{CHIPY\_ADJ} < 1, \tag{215}$$

         then

$$\texttt{CHIPY\_ADJ} = 1 \tag{216}$$

         and acis_process_events sets STATUS bit 0 (of 0–31) to one and increments a counter for the number of unexpected CHIPY values. These values should not occur.

B. Timed exposure mode maximum:
If

$$\texttt{DATAMODE}_{\rm in} = \text{FAINT or} \quad (217)$$
$$\texttt{DATAMODE}_{\rm in} = \text{FAINT\_BIAS or} \quad (218)$$
$$\texttt{DATAMODE}_{\rm in} = \text{GRADED or} \quad (219)$$
$$\texttt{DATAMODE}_{\rm in} = \text{VFAINT} \quad (220)$$

and

$$\texttt{CHIPY\_ADJ} > 1024, \quad (221)$$

then

$$\texttt{CHIPY\_ADJ} = 1024 \quad (222)$$

and `acis_process_events` sets `STATUS` bit 0 (of 0–31) to one and increments a counter for the number of unexpected `CHIPY` values. These values should not occur.

C. Continuous-clocking mode maximum:
If

$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_FAINT or} \quad (223)$$
$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_GRADED} \quad (224)$$

and

$$\texttt{CHIPY\_ADJ} > 512, \quad (225)$$

then

$$\texttt{CHIPY\_ADJ} = 512 \quad (226)$$

and `acis_process_events` sets `STATUS` bit 0 (of 0–31) to one and increments a counter for the number of unexpected `CHIPY` values. These values should not occur.

ii. Unexpected:

A. VFAINT:
If

$$\texttt{DATAMODE}_{\rm in} = \text{VFAINT} \quad (227)$$

and

$$\texttt{CHIPY\_ADJ} = 2 \text{ or} \quad (228)$$
$$\texttt{CHIPY\_ADJ} = 1023, \quad (229)$$

then `acis_process_events` sets `STATUS` bit 0 (of 0–31) to one and increments a counter for the number of unexpected `CHIPY` values. These values should not occur.

9. `TIME_RO`:

(a) Timed exposure mode:
If

$$\texttt{DATAMODE} = \text{FAINT or} \quad (230)$$
$$\texttt{DATAMODE} = \text{FAINT\_BIAS or} \quad (231)$$
$$\texttt{DATAMODE} = \text{GRADED or} \quad (232)$$
$$\texttt{DATAMODE} = \text{VFAINT}, \quad (233)$$

then

$$\texttt{TIME\_RO} = \text{NaN}. \tag{234}$$

(b) Continuous-clocking mode:
If

$$\texttt{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \tag{235}$$
$$\texttt{DATAMODE}_{\text{in}} = \text{CC33\_GRADED}, \tag{236}$$

then

i. Read:

A. Level 0:
If

$$\texttt{CONTENT}_{\text{in}} = \text{EVT0}, \tag{237}$$

then the value of TIME_RO for an event is given by $\texttt{TIME}_{\text{in}}$.

B. Level 1, 1.5, or 2:
If

$$\texttt{CONTENT}_{\text{in}} = \text{EVT1 or} \tag{238}$$
$$\texttt{CONTENT}_{\text{in}} = \text{TGEVT1 or} \tag{239}$$
$$\texttt{CONTENT}_{\text{in}} = \text{EVT2} \tag{240}$$

and

$$\texttt{TIME\_RO}_{\text{in}} > 0, \tag{241}$$

then

$$\texttt{TIME\_RO} = \texttt{TIME\_RO}_{\text{in}}. \tag{242}$$

If

$$\texttt{CONTENT}_{\text{in}} = \text{EVT1 or} \tag{243}$$
$$\texttt{CONTENT}_{\text{in}} = \text{TGEVT1 or} \tag{244}$$
$$\texttt{CONTENT}_{\text{in}} = \text{EVT2} \tag{245}$$

and

$$\texttt{TIME\_RO}_{\text{in}} = 0, \tag{246}$$

then

$$\texttt{TIME\_RO} = \texttt{TIME}_{\text{in}}. \tag{247}$$

ii. Validation:
If

$$\texttt{TIME\_RO} < 0 \text{ or} \tag{248}$$
$$\texttt{TIME\_RO} \geq 3 \times 10^9, \tag{249}$$

then acis_process_events produces a warning upon completion with a count of the total number of events for which either equation 248 or 249 is true. These conditions should not occur.

10. $\texttt{CHIPX\_TARG}_{\rm evt}$, $\texttt{CHIPY\_TARG}_{\rm evt}$, and $\texttt{CHIPY\_TARG}_{\rm eff}$:
The coordinate $\texttt{CHIPY\_TARG}_{\rm eff}$ is used to compute the coordinates X, Y, and SKY_1D.

(a) Approximate:
If

$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_FAINT or} \tag{250}$$
$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_GRADED} \tag{251}$$

and

$$\texttt{acaofffile} \neq \text{none,} \tag{252}$$

then:

i. Initial values:
The values of $\texttt{CHIPX\_TARG}_{\rm evt}$ and $\texttt{CHIPY\_TARG}_{\rm evt}$ are computed from the values of RA_TARG and DEC_TARG using the value of $\texttt{CCD\_ID}_{\rm focus}$ and using the orientation of the telescope (i.e. RA, DEC, and ROLL) and the orientation of the SIM (i.e. DY, DZ, and DTHETA) at the time given by $\texttt{TIME\_RO} - (\texttt{CHIPY\_TARG}_{\rm acaoff,med} + 1028) \times \texttt{TIMEDEL}$. The value of $\texttt{CHIPY\_TARG}_{\rm evt}$ can be negative.

ii. ACIS-I0 or -I2:
If

$$\texttt{CCD\_ID}_{\rm focus} = 0 \text{ or} \tag{253}$$
$$\texttt{CCD\_ID}_{\rm focus} = 2, \tag{254}$$

then

A. If

$$\texttt{CCD\_ID} = 0 \text{ or} \tag{255}$$
$$\texttt{CCD\_ID} = 2, \tag{256}$$

then

$$\texttt{CHIPY\_TARG}_{\rm eff} = \texttt{CHIPY\_TARG}_{\rm evt}. \tag{257}$$

B. If

$$\texttt{CCD\_ID} = 1 \text{ or} \tag{258}$$
$$\texttt{CCD\_ID} = 3, \tag{259}$$

then

$$\texttt{CHIPY\_TARG}_{\rm eff} = 512 - (\texttt{CHIPY\_TARG}_{\rm evt} - \texttt{CHIPY\_TARG}_{\rm acaoff,med}). \tag{260}$$

C. If

$$\texttt{CCD\_ID} \geq 4, \tag{261}$$

then

$$\texttt{CHIPY\_TARG}_{\rm eff} = 512 - (\texttt{CHIPX\_TARG}_{\rm evt} - \texttt{CHIPX\_TARG}_{\rm acaoff,med}). \tag{262}$$

iii. ACIS-I1 or -I3:
If

$$\texttt{CCD\_ID}_{\rm focus} = 1 \text{ or} \tag{263}$$
$$\texttt{CCD\_ID}_{\rm focus} = 3, \tag{264}$$

then

A. If

$$\texttt{CCD\_ID} = 0 \text{ or} \tag{265}$$
$$\texttt{CCD\_ID} = 2, \tag{266}$$

then

$$\texttt{CHIPY\_TARG}_{\rm eff} = 512 - (\texttt{CHIPY\_TARG}_{\rm evt} - \texttt{CHIPY\_TARG}_{\rm acaoff,med}). \tag{267}$$

B. If

$$\texttt{CCD\_ID} = 1 \text{ or} \tag{268}$$
$$\texttt{CCD\_ID} = 3, \tag{269}$$

then

$$\texttt{CHIPY\_TARG}_{\rm eff} = \texttt{CHIPY\_TARG}_{\rm evt}. \tag{270}$$

C. If

$$\texttt{CCD\_ID} \geq 4, \tag{271}$$

then

$$\texttt{CHIPY\_TARG}_{\rm eff} = 512 + (\texttt{CHIPX\_TARG}_{\rm evt} - \texttt{CHIPX\_TARG}_{\rm acaoff,med}). \tag{272}$$

iv. ACIS-S:
If

$$\texttt{CCD\_ID}_{\rm focus} \geq 4, \tag{273}$$

then

A. If

$$\texttt{CCD\_ID} = 0 \text{ or} \tag{274}$$
$$\texttt{CCD\_ID} = 2, \tag{275}$$

then

$$\texttt{CHIPY\_TARG}_{\rm eff} = 512 + (\texttt{CHIPX\_TARG}_{\rm evt} - \texttt{CHIPX\_TARG}_{\rm acaoff,med}). \tag{276}$$

B. If

$$\texttt{CCD\_ID} = 1 \text{ or} \tag{277}$$
$$\texttt{CCD\_ID} = 3, \tag{278}$$

then

$$\texttt{CHIPY\_TARG}_{\rm eff} = 512 - (\texttt{CHIPX\_TARG}_{\rm evt} - \texttt{CHIPX\_TARG}_{\rm acaoff,med}). \tag{279}$$

C. If

$$\texttt{CCD\_ID} \geq 4, \tag{280}$$

then

$$\texttt{CHIPY\_TARG}_{\mathrm{eff}} = \texttt{CHIPY\_TARG}_{\mathrm{evt}}. \tag{281}$$

(b) Validation:
If

$$\texttt{CHIPY\_TARG}_{\mathrm{eff}} < -256 \text{ or} \tag{282}$$
$$\texttt{CHIPY\_TARG}_{\mathrm{eff}} \geq 1280, \tag{283}$$

then `acis_process_events` produces a warning message.

11. `TG_M`:

(a) Continuous-clocking mode with gratings:
If

$$\texttt{OBS\_MODE}_{\mathrm{in}} = \text{pointing} \tag{284}$$

and

$$\texttt{DATAMODE}_{\mathrm{in}} = \text{CC33\_FAINT or} \tag{285}$$
$$\texttt{DATAMODE}_{\mathrm{in}} = \text{CC33\_GRADED} \tag{286}$$

and

$$\texttt{CONTENT}_{\mathrm{in}} = \text{TGEVT1}, \tag{287}$$

then

i. Read:
The value of `TG_M` for an event is given by $\texttt{TG\_M}_{\mathrm{in}}$.

ii. Validation:

A. If

$$\texttt{TG\_M} < -99, \tag{288}$$

then

$$\texttt{TG\_M} = -99 \tag{289}$$

and `acis_process_events` produces a warning upon completion with a count of the total number of events for which this condition is true. These values should not occur.

B. If

$$\texttt{TG\_M} > 99, \tag{290}$$

then

$$\texttt{TG\_M} = 99 \tag{291}$$

and `acis_process_events` produces a warning upon completion with a count of the total number of events for which this condition is true. These values should not occur.

12. `CHIPY_TG`:

(a) Continuous-clocking mode with gratings:
If

$$\texttt{OBS\_MODE}_{\rm in} = \text{pointing} \tag{292}$$

and

$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_FAINT or} \tag{293}$$
$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_GRADED} \tag{294}$$

and

$$\texttt{CONTENT}_{\rm in} = \text{TGEVT1,} \tag{295}$$

then

i. Read:
The value of `CHIPY_TG` for an event is given by $\texttt{CHIPY\_TG}_{\rm in}$.

ii. Validation:

A. If

$$\texttt{TG\_M} > -99 \text{ and} \tag{296}$$
$$\texttt{TG\_M} < 99 \text{ and} \tag{297}$$
$$\texttt{CHIPY\_TG} \neq \text{NaN} \tag{298}$$

and

$$\texttt{CHIPY\_TG} \leq 0 \text{ or} \tag{299}$$
$$\texttt{CHIPY\_TG} \geq 1025, \tag{300}$$

then `acis_process_events` exits with an error message because `CHIPY_TG`-dependent computations could fail if the value of `CHIPY_TG` is unphysical.

B. If

$$\texttt{TG\_M} > -99 \text{ and} \tag{301}$$
$$\texttt{TG\_M} < 99 \text{ and} \tag{302}$$
$$\texttt{CHIPY\_TG} \neq \text{NaN and} \tag{303}$$
$$\texttt{CHIPY\_TG} < 1, \tag{304}$$

then

$$\texttt{CHIPY\_TG} = 1. \tag{305}$$

C. If

$$\texttt{TG\_M} > -99 \text{ and} \tag{306}$$
$$\texttt{TG\_M} < 99 \text{ and} \tag{307}$$
$$\texttt{CHIPY\_TG} \neq \text{NaN and} \tag{308}$$
$$\texttt{CHIPY\_TG} > 1024, \tag{309}$$

then

$$\texttt{CHIPY\_TG} = 1024. \tag{310}$$

13. TIME and CHIPY_ADJ:
For continuous-clocking mode observations, the value of CHIPY_ADJ is used to compute the TIME, pulse heights, and the coordinates TDETX, TDETY, DETX, and DETY.

(a) Calculate:

i. Timed exposure mode:
If

$$\begin{aligned} \texttt{DATAMODE}_{\text{in}} &= \text{FAINT or} && (311)\\ \texttt{DATAMODE}_{\text{in}} &= \text{FAINT\_BIAS or} && (312)\\ \texttt{DATAMODE}_{\text{in}} &= \text{GRADED or} && (313)\\ \texttt{DATAMODE}_{\text{in}} &= \text{VFAINT}, && (314) \end{aligned}$$

then

$$\texttt{TIME} = \texttt{TIME}_{\text{in}}. \quad (315)$$

ii. Continuous-clocking mode:
If

$$\begin{aligned} \texttt{DATAMODE}_{\text{in}} &= \text{CC33\_FAINT or} && (316)\\ \texttt{DATAMODE}_{\text{in}} &= \text{CC33\_GRADED} && (317) \end{aligned}$$

then

A. Set

$$\begin{aligned} \texttt{CHIPY\_ADJ} &= 512 \text{ and} && (318)\\ \texttt{TIME}' &= \texttt{TIME\_RO} - (\texttt{CHIPY\_ADJ} + 1028) \times \texttt{TIMEDEL}. && (319) \end{aligned}$$

B. If

$$\texttt{acaofffile} \neq \text{none}, \quad (320)$$

then

$$\begin{aligned} \texttt{CHIPY\_ADJ}' &= \texttt{CHIPY\_TARG}_{\text{eff}} \text{ and} && (321)\\ \texttt{TIME}' &= \texttt{TIME\_RO} - (\texttt{CHIPY\_ADJ}' + 1028) \times \texttt{TIMEDEL}, && (322) \end{aligned}$$

and CHIPY_ADJ is computed from the values of RA_TARG and DEC_TARG using the orientation of the telescope (i.e. RA, DEC, and ROLL) and the orientation of the SIM (i.e. DY, DZ, and DTHETA) at the time TIME′. At this step, the value of CHIPY_ADJ can be negative.

C. If

$$\begin{aligned} \texttt{CONTENT}_{\text{in}} &= \text{TGEVT1 and} && (323)\\ \texttt{TG\_M} &> -99 \text{ and} && (324)\\ \texttt{TG\_M} &< 99 \text{ and} && (325)\\ \texttt{CHIPY\_TG} &\neq \text{NaN}, && (326) \end{aligned}$$

then

$$\texttt{CHIPY\_ADJ} = \texttt{CHIPY\_TG}. \quad (327)$$

D. If

$$\texttt{CONTENT}_{\rm in} = \text{TGEVT1} \quad (328)$$

and

$$\texttt{TG\_M} = -99 \text{ or} \quad (329)$$
$$\texttt{TG\_M} = 99, \quad (330)$$

then

$$\texttt{CHIPY\_ADJ} = 512. \quad (331)$$

E. If

$$\texttt{acaofffile} \neq \text{none} \quad (332)$$

and

$$\texttt{TIME}' < \texttt{TIME}_{\rm min} \text{ or} \quad (333)$$
$$\texttt{TIME}' \geq \texttt{TIME}_{\rm max}, \quad (334)$$

then

$$\texttt{CHIPY\_ADJ} = 512. \quad (335)$$

F. If

$$\texttt{acaofffile} \neq \text{none} \quad (336)$$

and

$$\texttt{TIME}' < \texttt{TIME}_{\rm min} \text{ or} \quad (337)$$
$$\texttt{TIME}' \geq \texttt{TIME}_{\rm max}, \quad (338)$$

and

$$\texttt{CCD\_ID}_{\rm focus} = 0 \text{ or} \quad (339)$$
$$\texttt{CCD\_ID}_{\rm focus} = 2 \quad (340)$$

and

$$\texttt{CCD\_ID} = 0 \text{ or} \quad (341)$$
$$\texttt{CCD\_ID} = 2, \quad (342)$$

then

$$\texttt{CHIPY\_ADJ} = \texttt{CHIPY\_TARG}_{\rm acaoff,med}. \quad (343)$$

G. If

$$\texttt{acaofffile} \neq \text{none} \quad (344)$$

and

$$\texttt{TIME}' < \texttt{TIME}_{\rm min} \text{ or} \quad (345)$$
$$\texttt{TIME}' \geq \texttt{TIME}_{\rm max}, \quad (346)$$

and

$$\texttt{CCD\_ID}_{\rm focus} = 1 \text{ or} \tag{347}$$
$$\texttt{CCD\_ID}_{\rm focus} = 3 \tag{348}$$

and

$$\texttt{CCD\_ID} = 1 \text{ or} \tag{349}$$
$$\texttt{CCD\_ID} = 3, \tag{350}$$

then

$$\texttt{CHIPY\_ADJ} = \texttt{CHIPY\_TARG}_{\rm acaoff,med}. \tag{351}$$

H. If

$$\texttt{acaofffile} \neq \text{none} \tag{352}$$

and

$$\texttt{TIME}' < \texttt{TIME}_{\rm min} \text{ or} \tag{353}$$
$$\texttt{TIME}' \geq \texttt{TIME}_{\rm max}, \tag{354}$$

and

$$\texttt{CCD\_ID}_{\rm focus} \geq 4 \text{ and} \tag{355}$$
$$\texttt{CCD\_ID} \geq 4, \tag{356}$$

then

$$\texttt{CHIPY\_ADJ} = \texttt{CHIPY\_TARG}_{\rm acaoff,med}. \tag{357}$$

I. If

$$\texttt{acaofffile} = \text{none} \tag{358}$$

and

$$\texttt{CONTENT}_{\rm in} \neq \text{TGEVT1}, \tag{359}$$

then

$$\texttt{CHIPY\_ADJ} = 512. \tag{360}$$

J. If

$$\texttt{CCD\_ID}_{\rm focus} \leq 3 \text{ and} \tag{361}$$
$$\texttt{CCD\_ID} \geq 4, \tag{362}$$

then

$$\texttt{CHIPY\_ADJ} = 512. \tag{363}$$

K. If

$$\texttt{CCD\_ID}_{\rm focus} \geq 4 \text{ and} \tag{364}$$
$$\texttt{CCD\_ID} \leq 3, \tag{365}$$

then

$$\texttt{CHIPY\_ADJ} = 512. \tag{366}$$

L. If

$$\texttt{OBS\_MODE}_{\rm in} = \text{secondary}, \tag{367}$$

then

$$\texttt{CHIPY\_ADJ} = 512. \tag{368}$$

M. If

$$\texttt{CHIPY\_ADJ} < 0.5 \text{ or} \tag{369}$$

$$\texttt{CHIPY\_ADJ} \geq 1024.5 \tag{370}$$

then

$$\texttt{CHIPY\_ADJ} = 512. \tag{371}$$

N. Set

$$\texttt{TIME} = \texttt{TIME\_RO} - (\texttt{CHIPY\_ADJ} + 1028) \times \texttt{TIMEDEL}. \tag{372}$$

In continuous-clocking mode, the coordinate CHIPY_ADJ is used to compute the time, the pulse heights, and the coordinates (except for X, Y, and SKY_1D).

(b) Validation:

i. If

$$\texttt{TIME} < 0 \text{ or} \tag{373}$$

$$\texttt{TIME} \geq 3 \times 10^9, \tag{374}$$

then acis_process_events produces a warning upon completion with a count of the total number of events for which one or the other of these conditions is true. These values should not occur.

ii. If

$$\texttt{CHIPY\_ADJ} < 0.5 \text{ or} \tag{375}$$

$$\texttt{CHIPY\_ADJ} \geq 1024.5, \tag{376}$$

then acis_process_events exits with an error message because CHIPY_ADJ-dependent computations could fail if the value of CHIPY_ADJ is unphysical.

14. Bad pixel:

(a) If

$$\texttt{badpixfile} \neq \text{none} \tag{377}$$

and the badpixfile includes a valid HDU $h_{\rm badpix}$ where $\texttt{CCD\_ID}_{\rm badpix} = \texttt{CCD\_ID}$, then the HDU $h_{\rm badpix}$ is searched as follows to determine if the event should have one or more STATUS bits set to one.

i. If $\texttt{DATAMODE}_{\rm in}$ = CC33_FAINT or $\texttt{DATAMODE}_{\rm in}$ = CC33_GRADED and there are one or more row r in HDU $h_{\rm badpix}$ where

$$\texttt{CHIPX} \geq \texttt{CHIPX}_{{\rm badpix},r}[0] \text{ and} \tag{378}$$

$$\texttt{CHIPX} \leq \texttt{CHIPX}_{{\rm badpix},r}[1] \text{ and} \tag{379}$$

$$\texttt{TIME} \geq \texttt{TIME}_{{\rm badpix},r} \text{ and} \tag{380}$$

$$\texttt{TIME} < \texttt{TIME\_STOP}_{{\rm badpix},r} \tag{381}$$

and

$$\texttt{STATUS}_{\text{badpix},r}[5] = 1 \text{ or} \quad (382)$$
$$\texttt{STATUS}_{\text{badpix},r}[6] = 1 \text{ or} \quad (383)$$
$$\texttt{STATUS}_{\text{badpix},r}[9] = 1, \quad (384)$$

then

$$\texttt{STATUS}[0] = 1 \quad (385)$$

for the event. Here $\texttt{CCD\_ID}_{\text{badpix}}$ is the value of the keyword CCD_ID in HDU h_{badpix} of the badpixfile, $\texttt{CHIPX}_{\text{badpix},r}[0]$ and $\texttt{CHIPX}_{\text{badpix},r}[1]$ are the first and second values in the vector column named CHIPX of row r of HDU h_{badpix} of the badpixfile, and $\texttt{TIME}_{\text{badpix},r}$ and $\texttt{TIME\_STOP}_{\text{badpix},r}$ are the values in the columns named TIME and TIME_STOP, respectively, of row r of HDU h_{badpix} of the badpixfile.

ii. If $\texttt{DATAMODE}_{\text{in}}$ = CC33_FAINT or $\texttt{DATAMODE}_{\text{in}}$ = CC33_GRADED and there are one or more row r in HDU h_{badpix} where

$$\texttt{CHIPX} \geq \texttt{CHIPX}_{\text{badpix},r}[0] \text{ and} \quad (386)$$
$$\texttt{CHIPX} \leq \texttt{CHIPX}_{\text{badpix},r}[1] \text{ and} \quad (387)$$
$$\texttt{TIME} \geq \texttt{TIME}_{\text{badpix},r} \text{ and} \quad (388)$$
$$\texttt{TIME} < \texttt{TIME\_STOP}_{\text{badpix},r} \quad (389)$$

and

$$\texttt{STATUS}_{\text{badpix},r}[0] = 1 \text{ or} \quad (390)$$
$$\texttt{STATUS}_{\text{badpix},r}[1] = 1 \text{ or} \quad (391)$$
$$\texttt{STATUS}_{\text{badpix},r}[7] = 1 \text{ or} \quad (392)$$
$$\texttt{STATUS}_{\text{badpix},r}[11] = 1 \text{ or} \quad (393)$$
$$\texttt{STATUS}_{\text{badpix},r}[12] = 1 \text{ or} \quad (394)$$
$$\texttt{STATUS}_{\text{badpix},r}[13] = 1 \text{ or} \quad (395)$$
$$\texttt{STATUS}_{\text{badpix},r}[14] = 1 \text{ or} \quad (396)$$
$$\texttt{STATUS}_{\text{badpix},r}[16] = 1, \quad (397)$$

then

$$\texttt{STATUS}[4] = 1 \quad (398)$$

for the event.

iii. If $\texttt{DATAMODE}_{\text{in}}$ = CC33_FAINT or $\texttt{DATAMODE}_{\text{in}}$ = CC33_GRADED and there are one or more row r in HDU h_{badpix} where

$$\texttt{CHIPX} \geq \texttt{CHIPX}_{\text{badpix},r}[0] \text{ and} \quad (399)$$
$$\texttt{CHIPX} \leq \texttt{CHIPX}_{\text{badpix},r}[1] \text{ and} \quad (400)$$
$$\texttt{TIME} \geq \texttt{TIME}_{\text{badpix},r} \text{ and} \quad (401)$$
$$\texttt{TIME} < \texttt{TIME\_STOP}_{\text{badpix},r} \quad (402)$$

and

$$\texttt{STATUS}_{\text{badpix},r}[8] = 1 \text{ or} \quad (403)$$
$$\texttt{STATUS}_{\text{badpix},r}[10] = 1, \quad (404)$$

then

$$\texttt{STATUS}[5] = 1 \quad (405)$$

for the event.

iv. If $\texttt{DATAMODE}_{\rm in}$ = CC33_FAINT or $\texttt{DATAMODE}_{\rm in}$ = CC33_GRADED and there are one or more row r in HDU $h_{\rm badpix}$ where

$$\begin{aligned} \texttt{CHIPX} &\geq \texttt{CHIPX}_{{\rm badpix},r}[0] \text{ and} && (406)\\ \texttt{CHIPX} &\leq \texttt{CHIPX}_{{\rm badpix},r}[1] \text{ and} && (407)\\ \texttt{TIME} &\geq \texttt{TIME}_{{\rm badpix},r} \text{ and} && (408)\\ \texttt{TIME} &< \texttt{TIME\_STOP}_{{\rm badpix},r} && (409) \end{aligned}$$

and

$$\texttt{STATUS}_{{\rm badpix},r}[3] = 1, \quad (410)$$

then

$$\texttt{STATUS}[6] = 1 \quad (411)$$

for the event.

v. If $\texttt{DATAMODE}_{\rm in}$ = CC33_FAINT or $\texttt{DATAMODE}_{\rm in}$ = CC33_GRADED and there are one or more row r in HDU $h_{\rm badpix}$ where

$$\begin{aligned} \texttt{CHIPX} &\geq \texttt{CHIPX}_{{\rm badpix},r}[0] \text{ and} && (412)\\ \texttt{CHIPX} &\leq \texttt{CHIPX}_{{\rm badpix},r}[1] \text{ and} && (413)\\ \texttt{TIME} &\geq \texttt{TIME}_{{\rm badpix},r} \text{ and} && (414)\\ \texttt{TIME} &< \texttt{TIME\_STOP}_{{\rm badpix},r} && (415) \end{aligned}$$

and

$$\begin{aligned} \texttt{STATUS}_{{\rm badpix},r}[2] &= 1 \text{ or} && (416)\\ \texttt{STATUS}_{{\rm badpix},r}[4] &= 1, && (417) \end{aligned}$$

then

$$\texttt{STATUS}[8] = 1 \quad (418)$$

for the event.

vi. If $\texttt{DATAMODE}_{\rm in}$ = CC33_FAINT or $\texttt{DATAMODE}_{\rm in}$ = CC33_GRADED and there are one or more row r in HDU $h_{\rm badpix}$ where

$$\begin{aligned} \texttt{CHIPX} &\geq \texttt{CHIPX}_{{\rm badpix},r}[0] \text{ and} && (419)\\ \texttt{CHIPX} &\leq \texttt{CHIPX}_{{\rm badpix},r}[1] \text{ and} && (420)\\ \texttt{TIME} &\geq \texttt{TIME}_{{\rm badpix},r} \text{ and} && (421)\\ \texttt{TIME} &< \texttt{TIME\_STOP}_{{\rm badpix},r} && (422) \end{aligned}$$

and

$$\texttt{STATUS}_{{\rm badpix},r}[15] = 1, \quad (423)$$

then

$$\texttt{STATUS}[16] = 1 \quad (424)$$

for the event.

vii. In summary, the mapping between a bad-pixel STATUS bit and the corresponding event STATUS bit is listed in Table 1.

15. PHAS:

Table 1: Bad-pixel to event STATUS bit mapping

Bad-pixel STATUS bit	Event STATUS bit
0	4
1	4
2	8
3	6
4	8
5	0
6	0
7	4
8	5
9	0
10	5
11	4
12	4
13	4
14	4
15	16
16	4

(a) If HDU 1 of the `infile` includes the column PHAS, then

i. the values of PHAS for an event are read from the `infile`.

ii. If PHAS[4] < the split threshold, then STATUS[k] = 1 for bit $k = 1$ (of 0–31).

iii. If PHAS[4] < PHAS[j] for one or more $j = 0$–3, then STATUS[k] = 1 for bit $k = 1$.

iv. If PHAS[4] $\leq$ PHAS[j] for one or more $j = 5$–8, then STATUS[k] = 1 for bit $k = 1$.

v. If PHAS[j] > 4095 for one or more $j = 0$–8, then STATUS[k] = 1 for bit $k = 2$.

16. FLTGRADE_RO:

(a) If HDU h_{in} of the `infile` includes the column PHAS and

$$\texttt{doevtgrade} = \text{yes}, \quad (425)$$

then

$$\texttt{FLTGRADE\_RO} = c_{\mathrm{fro}}[0] + 2c_{\mathrm{fro}}[1] + 4c_{\mathrm{fro}}[2] + 8c_{\mathrm{fro}}[3] + \quad (426)$$
$$16c_{\mathrm{fro}}[5] + 32c_{\mathrm{fro}}[6] + 64c_{\mathrm{fro}}[7] + 128c_{\mathrm{fro}}[8]. \quad (427)$$

The values of $c_{\mathrm{fro}}[j]$ are determined as follows.

i. Each value of $c_{\mathrm{fro}}[j]$ is initialized such that

$$c_{\mathrm{fro}}[j] = 1. \quad (428)$$

ii. If

$$\texttt{PHAS}[j] < \text{split threshold or} \quad (429)$$
$$\texttt{PHAS}[j] > 4095, \quad (430)$$

then

$$c_{\mathrm{fro}}[j] = 0. \quad (431)$$

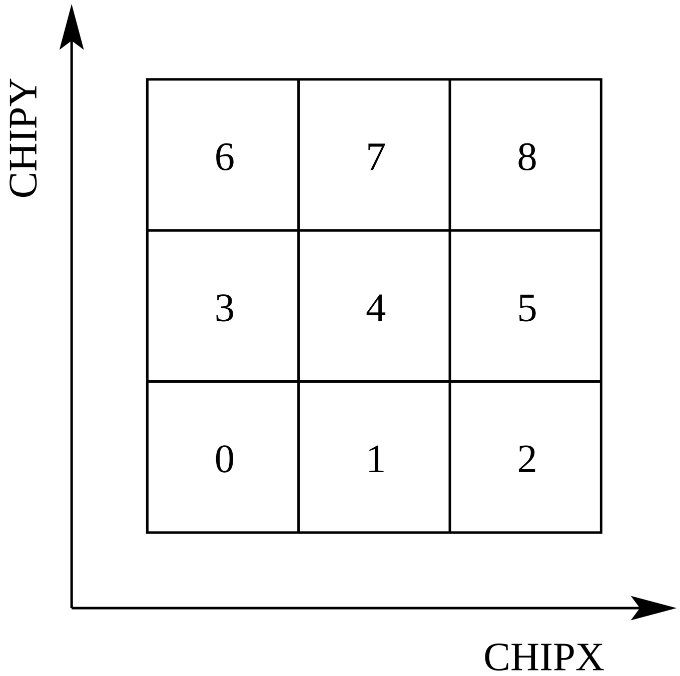

Figure 1: The relative CHIPX and CHIPY coordinates of the nine elements $j = 0$–8 of a 3 pixel × 3 pixel event island PHAS[j] or PHAS_ADJ[j].

iii. If

$$j \geq 0 \text{ and} \quad (432)$$
$$j \leq 3 \text{ and} \quad (433)$$
$$\texttt{PHAS}[j] > \texttt{PHAS}[4], \quad (434)$$

then

$$c_{\rm fro}[j] = 0. \quad (435)$$

iv. If

$$j \geq 5 \text{ and} \quad (436)$$
$$j \leq 8 \text{ and} \quad (437)$$
$$\texttt{PHAS}[j] \geq \texttt{PHAS}[4], \quad (438)$$

then

$$c_{\rm fro}[j] = 0. \quad (439)$$

The elements $j = 0$–8 of PHAS are depicted in Figure 1.

(b) If HDU $h_{\rm in}$ of the infile does not include the column PHAS or

$$\texttt{doevtgrade} = \text{no}, \quad (440)$$

then

$$\texttt{FLTGRADE\_RO} = \text{NULL}. \quad (441)$$

17. GRADE_RO:

(a) If

$$\texttt{FLTGRADE\_RO} \neq \text{NULL and} \quad (442)$$
$$\texttt{gradefile} \neq \text{none and} \quad (443)$$

HDU h_{grade} of the gradefile has a row r, where

$$\texttt{FLTGRADE}_{\text{grade}}[r] = \texttt{FLTGRADE\_RO}, \quad (444)$$

then

$$\texttt{GRADE\_RO} = \texttt{GRADE}_{\text{grade}}[r]. \quad (445)$$

(b) If

$$\texttt{FLTGRADE\_RO} = \text{NULL or} \quad (446)$$
$$\texttt{gradefile} = \text{none or} \quad (447)$$

HDU h_{grade} of the gradefile does not have a row r, where

$$\texttt{FLTGRADE}_{\text{grade}}[r] = \texttt{FLTGRADE\_RO}, \quad (448)$$

then

$$\texttt{GRADE\_RO} = \text{NULL}. \quad (449)$$

18. PHA_RO:

(a) If HDU h_{in} of the infile includes the column PHAS and

$$\texttt{doevtgrade} = \text{yes and} \quad (450)$$
$$\texttt{GRADE\_RO} \neq \text{NULL}, \quad (451)$$

then

$$\texttt{PHA\_RO} = \sum_{j=0}^{8} c_{\text{pro}}[j]\texttt{PHAS}[j]. \quad (452)$$

The values of $c_{\text{pro}}[j]$ are determined as follows.

i. Each value of $c_{\text{pro}}[j]$ is initialized such that

$$c_{\text{pro}}[j] = 1. \quad (453)$$

ii. If

$$\texttt{PHAS}[j] < \text{split threshold}, \quad (454)$$

then

$$c_{\text{pro}}[j] = 0. \quad (455)$$

iii. If

$$j \geq 0 \text{ and} \quad (456)$$
$$j \leq 3 \text{ and} \quad (457)$$
$$\texttt{PHAS}[j] > \texttt{PHAS}[4], \quad (458)$$

then

$$c_{\text{pro}}[j] = 0. \quad (459)$$

iv. If

$$j \geq 5 \text{ and} \quad (460)$$
$$j \leq 8 \text{ and} \quad (461)$$
$$\texttt{PHAS}[j] \geq \texttt{PHAS}[4], \quad (462)$$

then

$$c_{\text{pro}}[j] = 0. \quad (463)$$

v. If

$$\texttt{CORNERS} = -1, \quad (464)$$

then

$$c_{\text{pro}}[0] = 0 \text{ and} \quad (465)$$
$$c_{\text{pro}}[2] = 0 \text{ and} \quad (466)$$
$$c_{\text{pro}}[6] = 0 \text{ and} \quad (467)$$
$$c_{\text{pro}}[8] = 0. \quad (468)$$

vi. If

$$\texttt{CORNERS} = 0, \quad (469)$$

then there are no additional constraints on $c_{\text{pro}}[0]$, $c_{\text{pro}}[2]$, $c_{\text{pro}}[6]$, and $c_{\text{pro}}[8]$.

vii. If

$$\texttt{CORNERS} = 1, \quad (470)$$

then

A. If

$$c_{\text{pro}}[1] = 0 \text{ and} \quad (471)$$
$$c_{\text{pro}}[3] = 0, \quad (472)$$

then

$$c_{\text{pro}}[0] = 0. \quad (473)$$

B. If

$$c_{\text{pro}}[1] = 0 \text{ and} \quad (474)$$
$$c_{\text{pro}}[5] = 0, \quad (475)$$

then

$$c_{\text{pro}}[2] = 0. \quad (476)$$

C. If

$$c_{\text{pro}}[3] = 0 \text{ and} \quad (477)$$
$$c_{\text{pro}}[7] = 0, \quad (478)$$

then

$$c_{\text{pro}}[6] = 0. \quad (479)$$

D. If

$$c_{\text{pro}}[5] = 0 \text{ and} \quad (480)$$
$$c_{\text{pro}}[7] = 0, \quad (481)$$

then

$$c_{\text{pro}}[8] = 0. \quad (482)$$

viii. If

$$\texttt{CORNERS} = 2, \quad (483)$$

then

A. If

$$c_{\text{pro}}[1] = 0 \text{ or} \quad (484)$$
$$c_{\text{pro}}[3] = 0 \text{ or} \quad (485)$$
$$\texttt{GRADE\_RO} \neq 6, \quad (486)$$

then

$$c_{\text{pro}}[0] = 0. \quad (487)$$

B. If

$$c_{\text{pro}}[1] = 0 \text{ or} \quad (488)$$
$$c_{\text{pro}}[5] = 0 \text{ or} \quad (489)$$
$$\texttt{GRADE\_RO} \neq 6, \quad (490)$$

then

$$c_{\text{pro}}[2] = 0. \quad (491)$$

C. If

$$c_{\text{pro}}[3] = 0 \text{ or} \quad (492)$$
$$c_{\text{pro}}[7] = 0 \text{ or} \quad (493)$$
$$\texttt{GRADE\_RO} \neq 6, \quad (494)$$

then

$$c_{\text{pro}}[6] = 0. \quad (495)$$

D. If

$$c_{\rm pro}[5] = 0 \text{ or} \tag{496}$$
$$c_{\rm pro}[7] = 0 \text{ or} \tag{497}$$
$$\texttt{GRADE\_RO} \neq 6, \tag{498}$$

then

$$c_{\rm pro}[8] = 0. \tag{499}$$

The elements $j = 0$–8 of PHAS are depicted in Figure 1.

(b) If HDU $h_{\rm in}$ of the `infile` does not include the column PHAS or

$$\texttt{doevtgrade} = \text{no or} \tag{500}$$
$$\texttt{GRADE\_RO} = \text{NULL}, \tag{501}$$

then

i. If HDU $h_{\rm in}$ of the `infile` includes the column PHA_RO, then

$$\texttt{PHA\_RO} = \texttt{PHA\_RO}_{\rm in}. \tag{502}$$

ii. If HDU $h_{\rm in}$ of the `infile` does not include the column PHA_RO, then

A. If HDU $h_{\rm in}$ of the `infile` includes the column PHA and HDU $h_{\rm in}$ of the `infile` includes the keyword TGAINCOR and

$$\texttt{TGAINCOR} = 0, \tag{503}$$

then

$$\texttt{PHA\_RO} = \texttt{PHA}_{\rm in}. \tag{504}$$

B. If HDU $h_{\rm in}$ of the `infile` includes the column PHA and HDU $h_{\rm in}$ of the `infile` includes the keyword TGAINCOR and

$$\texttt{TGAINCOR} \neq 0, \tag{505}$$

then

$$\texttt{PHA\_RO} = \text{NULL}. \tag{506}$$

C. If HDU $h_{\rm in}$ of the `infile` includes the column PHA and HDU $h_{\rm in}$ of the `infile` does not include the keyword TGAINCOR then

$$\texttt{PHA\_RO} = \texttt{PHA}_{\rm in}. \tag{507}$$

D. If HDU $h_{\rm in}$ of the `infile` does not include the column PHA, then

$$\texttt{PHA\_RO} = \text{NULL}. \tag{508}$$

19. PHAS_ADJ:

(a) Non-graded:
If

$$\texttt{apply\_cti} = \text{yes} \tag{509}$$

and if HDU $h_{\rm in}$ of the `infile` includes the column PHAS, then the CTI-adjusted pulse heights are computed as follows.

i. The real-valued arrays for the serial CTI adjustment Δ_x, the parallel CTI adjustment Δ_y, and the adjusted pulse heights PHAS_ADJ are initialized such that

$$\begin{aligned} \Delta_x[j] &= 0, & (510)\\ \Delta_y[j] &= 0, \text{ and} & (511)\\ \texttt{PHAS\_ADJ}[j] &= \texttt{PHAS}[j] & (512) \end{aligned}$$

for every element $j = 0$–8, where the starting point for the adjusted pulse heights are the unadjusted pulse heights PHAS. Note that the values of the unadjusted pulse heights PHAS remain unchanged to ensure that it is possible to remove the CTI adjustment or to reapply the adjustment if the algorithm or calibration data are modified.

ii. The CTI iteration counter n is initialized such that

$$n = 1. \quad (513)$$

iii. The temporary variables Δ'_x, Δ'_y, and PHAS_ADJ$'$ are set such that

$$\begin{aligned} \Delta'_x[j] &= \Delta_x[j], & (514)\\ \Delta'_y[j] &= \Delta_y[j], \text{ and} & (515)\\ \texttt{PHAS\_ADJ}'[j] &= \texttt{PHAS\_ADJ}[j] & (516) \end{aligned}$$

for each element j.

iv. A. If there is a serial CTI trap-density map in the ctifile for CCD_ID and NODE_ID = 0 or 2, then the values of Δ_x are given by

$$\begin{aligned}
\Delta_x[0] &= c_x[0]s_x\rho_x[0]V_x[0], & (517)\\
\Delta_x[1] &= c_x[1]s_x\rho_x[1]V_x[1] - c'_x[0]s_x\rho_x[0]V_x[0], & (518)\\
\Delta_x[2] &= c_x[2]s_x\rho_x[2]V_x[2] - c'_x[1]s_x\rho_x[1]V_x[1], & (519)\\
\Delta_x[3] &= c_x[3]s_x\rho_x[3]V_x[3], & (520)\\
\Delta_x[4] &= c_x[4]s_x\rho_x[4]V_x[4] - c'_x[3]s_x\rho_x[3]V_x[3], & (521)\\
\Delta_x[5] &= c_x[5]s_x\rho_x[5]V_x[5] - c'_x[4]s_x\rho_x[4]V_x[4], & (522)\\
\Delta_x[6] &= c_x[6]s_x\rho_x[6]V_x[6], & (523)\\
\Delta_x[7] &= c_x[7]s_x\rho_x[7]V_x[7] - c'_x[6]s_x\rho_x[6]V_x[6], \text{ and} & (524)\\
\Delta_x[8] &= c_x[8]s_x\rho_x[8]V_x[8] - c'_x[7]s_x\rho_x[7]V_x[7], & (525)
\end{aligned}$$

where

$$c_x[j] = \begin{cases} 0 & \begin{cases} \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] < \text{ split threshold} \\ \quad \text{(for all } j\text{),} \end{cases} \\ \texttt{FRCTRLX} & \begin{cases} \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \geq \text{ split threshold and} \\ \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] < \\ \quad \texttt{PHAS}[j-1] + \Delta'_x[j-1] + \Delta'_y[j-1] \\ \quad \text{(for } j = 1,2,4,5,7,8\text{),} \end{cases} \\ 1 & \begin{cases} \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \geq \text{ split threshold} \\ \quad \text{(for } j = 0,3,6\text{)} \\ \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \geq \text{ split threshold and} \\ \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \geq \\ \quad \texttt{PHAS}[j-1] + \Delta'_x[j-1] + \Delta'_y[j-1] \\ \quad \text{(for } j = 1,2,4,5,7,8\text{),} \end{cases} \end{cases}$$

$$c'_x[j] = \begin{cases} 0 & \begin{cases} \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] < \text{ split threshold or} \\ \texttt{PHAS}[j+1] + \Delta'_x[j+1] + \Delta'_y[j+1] < \text{ split threshold or} \\ j \to \texttt{CHIPX} = 1, 256, 513, \text{ or } 768 \\ \quad (\text{for } j = 0,1,3,4,6,7), \end{cases} \\ \texttt{FRCTRLX} & \begin{cases} \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] > \\ \quad \texttt{PHAS}[j+1] + \Delta'_x[j+1] + \Delta'_y[j+1] \text{ and} \\ \texttt{PHAS}[j+1] + \Delta'_x[j+1] + \Delta'_y[j+1] \geq \text{ split threshold} \\ \quad (\text{for } j = 0,1,3,4,6,7), \end{cases} \\ 1 & \begin{cases} \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \leq \\ \quad \texttt{PHAS}[j+1] + \Delta'_x[j+1] + \Delta'_y[j+1] \text{ and} \\ \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \geq \text{ split threshold} \\ \quad (\text{for } j = 0,1,3,4,6,7), \end{cases} \end{cases}$$

$$s_x = \text{a temperature dependent scaling factor :}$$

$$\begin{cases} 1 + \texttt{TCTIX}\,(T - \texttt{FP\_TEMP0})\,, & \text{if } \texttt{mtlfile} \neq \text{“none”} \\ 1, & \text{if } \texttt{mtlfile} = \text{“none”} \end{cases} \tag{526}$$

$$\texttt{TCTIX} = \text{the } \texttt{CCD\_ID} \text{ dependent value in the column } \texttt{TCTIX} \text{ of the } \texttt{ctifile}, \tag{527}$$

$$T = \left(\frac{t' - t'_k}{t'_{k+1} - t'_k}\right)(\texttt{FP\_TEMP}_{k+1} - \texttt{FP\_TEMP}_k) + \texttt{FP\_TEMP}_k, \tag{528}$$

$$\{\ T \text{ is the time dependent focal plane temperature,}$$

$$t' = t + \texttt{TIMEDEL}_{\text{in}}\,(\texttt{TIMEPIXR}_{\text{evt}} - 0.5)\,, \tag{529}$$

$$\begin{cases} t \text{ is the } \texttt{TIME} \text{ of the event,} \\ \texttt{TIMEPIXR}_{\text{evt}} \text{ is a keyword in the } \texttt{infile}, \end{cases}$$

$$t'_k = \texttt{TIME}_k + \texttt{TIMEDEL}_{\text{mtl}}\,(\texttt{TIMEPIXR}_{\text{mtl}} - 0.5)\,, \tag{530}$$

$$\begin{cases} \texttt{TIME}_k \text{ is the } k^{\text{th}} \text{ element of the column } \texttt{TIME} \text{ in the } \texttt{mtlfile}, \\ t'_k \leq t', \\ \text{If } t' < t'_k \text{ for } k = 0, \text{ then } k = 0, \\ \texttt{FP\_TEMP}_k \text{ is the } k^{\text{th}} \text{ element of the column } \texttt{FP\_TEMP} \text{ in the } \texttt{mtlfile}, \\ \texttt{TIMEDEL}_{\text{mtl}} \text{ is a keyword in the } \texttt{mtlfile}, \\ \texttt{TIMEPIXR}_{\text{mtl}} \text{ is a keyword in the } \texttt{mtlfile}, \end{cases}$$

$$t'_{k+1} = \texttt{TIME}_{k+1} + \texttt{TIMEDEL}_{\text{mtl}}\,(\texttt{TIMEPIXR}_{\text{mtl}} - 0.5)\,, \tag{531}$$

$$\begin{cases} \texttt{TIME}_{k+1} \text{ is the } (k+1)^{\text{th}} \text{ element of the column } \texttt{TIME} \text{ in the } \texttt{mtlfile}, \\ t'_{k+1} > t', \\ \text{If } t' > t'_k \text{ for } k = n, \text{ where } n \text{ is the last element, then } k = n, \\ \texttt{FP\_TEMP}_{k+1} \text{ is the } (k+1)^{\text{th}} \text{ element of the column } \texttt{FP\_TEMP} \text{ in the} \\ \quad \texttt{mtlfile}, \end{cases}$$

$$\texttt{FP\_TEMP0} = \text{a keyword in the } \texttt{ctifile}, \tag{532}$$

$$\rho_x[j] = \text{serial trap density,} \tag{533}$$

$$\begin{cases} \rho_x[j] \text{ depends upon the } \texttt{CCD\_ID} \text{ and upon the nint}(\texttt{CHIPX\_ADJ}) \text{ and nint}(\texttt{CHIPY\_ADJ}) \\ \quad \text{coordinates associated with element } j \text{ of } \texttt{PHAS\_ADJ}[j] \text{ (see Fig. 1),} \end{cases}$$

$$V_x[j] = \left(\frac{\texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] - \texttt{PHA}_l}{\texttt{PHA}_{l+1} - \texttt{PHA}_l}\right)(\texttt{VOLUME\_X}_{l+1} - \texttt{VOLUME\_X}_l) + \texttt{VOLUME\_X}_l, \tag{534}$$

$$\left\{\begin{array}{l}
\texttt{PHA}_l \text{ is the } l^{\text{th}} \text{ element of the column } \texttt{PHA} \text{ in the } \texttt{ctifile}, \\
\texttt{PHA}_l \text{ (and } \texttt{PHA}_{l+1}) \text{ are } \texttt{CCD_ID} \text{ depdendent}, \\
\texttt{PHA}_l \leq \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j], \\
\text{If } \texttt{PHA}_l > \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \text{ for } l = 0, \text{ then } l = 0, \\
\texttt{PHA}_{l+1} \text{ is the } (l+1)^{\text{th}} \text{ element of the column } \texttt{PHA} \text{ in the } \texttt{ctifile}, \\
\texttt{PHA}_{l+1} > \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j], \\
\text{If } \texttt{PHA}_{l+1} \leq \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \text{ for } l = n, \text{ where } n \text{ is the last} \\
\quad \text{element, then } l = n, \\
\texttt{VOLUME_X}_l \text{ is the } l^{\text{th}} \text{ element of the column } \texttt{VOLUME_X} \text{ in the } \texttt{ctifile}, \\
\texttt{VOLUME_X}_l, \text{ which is } \texttt{CCD_ID} \text{ depdendent, is associated with } \texttt{PHA}_l, \\
\texttt{VOLUME_X}_{l+1} \text{ is the } (l+1)^{\text{th}} \text{ element of the column } \texttt{VOLUME_X} \text{ in the} \\
\quad \texttt{ctifile}, \text{ and} \\
\texttt{VOLUME_X}_{l+1}, \text{ which is } \texttt{CCD_ID} \text{ depdendent, is associated with } \texttt{PHA}_{l+1}
\end{array}\right.$$

B. If there is a serial CTI trap-density map in the `ctifile` for `CCD_ID` and `NODE_ID` = 1 or 3, then the values of Δ_x are given by

$$\Delta_x[0] = c_x[0]s_x\rho_x[0]V_x[0] - c'_x[1]s_x\rho_x[1]V_x[1], \quad (535)$$
$$\Delta_x[1] = c_x[1]s_x\rho_x[1]V_x[1] - c'_x[2]s_x\rho_x[2]V_x[2], \quad (536)$$
$$\Delta_x[2] = c_x[2]s_x\rho_x[2]V_x[2], \quad (537)$$
$$\Delta_x[3] = c_x[3]s_x\rho_x[3]V_x[3] - c'_x[4]s_x\rho_x[4]V_x[4], \quad (538)$$
$$\Delta_x[4] = c_x[4]s_x\rho_x[4]V_x[4] - c'_x[5]s_x\rho_x[5]V_x[5], \quad (539)$$
$$\Delta_x[5] = c_x[5]s_x\rho_x[5]V_x[5], \quad (540)$$
$$\Delta_x[6] = c_x[6]s_x\rho_x[6]V_x[6] - c'_x[7]s_x\rho_x[7]V_x[7], \quad (541)$$
$$\Delta_x[7] = c_x[7]s_x\rho_x[7]V_x[7] - c'_x[8]s_x\rho_x[8]V_x[8], \text{ and} \quad (542)$$
$$\Delta_x[8] = c_x[8]s_x\rho_x[8]V_x[8], \quad (543)$$

where

$$c_x[j] = \left\{\begin{array}{ll}
0 & \left\{\begin{array}{l} \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] < \text{ split threshold} \\ \quad \text{(for all } j), \end{array}\right. \\
\texttt{FRCTRLX} & \left\{\begin{array}{l} \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \geq \text{ split threshold and} \\ \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] < \\ \quad \texttt{PHAS}[j+1] + \Delta'_x[j+1] + \Delta'_y[j+1] \\ \quad \text{(for } j = 0, 1, 3, 4, 6, 7), \end{array}\right. \\
1 & \left\{\begin{array}{l} \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \geq \text{ split threshold} \\ \quad \text{(for } j = 2, 5, 8) \\ \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \geq \text{ split threshold and} \\ \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \geq \\ \quad \texttt{PHAS}[j+1] + \Delta'_x[j+1] + \Delta'_y[j+1] \\ \quad \text{(for } j = 0, 1, 3, 4, 6, 7), \end{array}\right.
\end{array}\right.$$

$$c'_x[j] = \begin{cases} 0 & \begin{cases} \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] < \text{ split threshold or} \\ \texttt{PHAS}[j-1] + \Delta'_x[j-1] + \Delta'_y[j-1] < \text{ split threshold or} \\ j \rightarrow \texttt{CHIPX} = 257, 512, 769, \text{ or } 1024 \\ \quad \text{(for } j = 1,2,4,5,7,8), \end{cases} \\ \texttt{FRCTRLX} & \begin{cases} \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] > \\ \quad \texttt{PHAS}[j-1] + \Delta'_x[j-1] + \Delta'_y[j-1] \text{ and} \\ \texttt{PHAS}[j-1] + \Delta'_x[j-1] + \Delta'_y[j-1] \geq \text{ split threshold} \\ \quad \text{(for } j = 1,2,4,5,7,8), \end{cases} \\ 1 & \begin{cases} \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \leq \\ \quad \texttt{PHAS}[j-1] + \Delta'_x[j-1] + \Delta'_y[j-1] \text{ and} \\ \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \geq \text{ split threshold} \\ \quad \text{(for } j = 1,2,4,5,7,8), \end{cases} \end{cases}$$

and s_x, T, t', t'_k, t'_{k+1}, $\rho_x[j]$, and $V_x[j]$ are given by equations. 526, 528, 529, 530, 531, 533, and 534, respectively.

v. If there is a parallel CTI trap-density map in the `ctifile` for `CCD_ID`, then the values of Δ_y are given by

$$\Delta_y[0] = c_y[0] s_y \rho_y[0] V_y[0], \tag{544}$$
$$\Delta_y[1] = c_y[1] s_y \rho_y[1] V_y[1], \tag{545}$$
$$\Delta_y[2] = c_y[2] s_y \rho_y[2] V_y[2], \tag{546}$$
$$\Delta_y[3] = c_y[3] s_y \rho_y[3] V_y[3] - c'_y[0] s_y \rho_y[0] V_y[0], \tag{547}$$
$$\Delta_y[4] = c_y[4] s_y \rho_y[4] V_y[4] - c'_y[1] s_y \rho_y[1] V_y[1], \tag{548}$$
$$\Delta_y[5] = c_y[5] s_y \rho_y[5] V_y[5] - c'_y[2] s_y \rho_y[2] V_y[2], \tag{549}$$
$$\Delta_y[6] = c_y[6] s_y \rho_y[6] V_y[6] - c'_y[3] s_y \rho_y[3] V_y[3], \tag{550}$$
$$\Delta_y[7] = c_y[7] s_y \rho_y[7] V_y[7] - c'_y[4] s_y \rho_y[4] V_y[4], \text{ and} \tag{551}$$
$$\Delta_y[8] = c_y[8] s_y \rho_y[8] V_y[8] - c'_y[5] s_y \rho_y[5] V_y[5], \tag{552}$$

where

$$c_y[j] = \begin{cases} 0 & \begin{cases} \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] < \text{ split threshold} \\ \quad \text{(for all } j), \end{cases} \\ \texttt{FRCTRLY} & \begin{cases} \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \geq \text{ split threshold and} \\ \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] < \\ \quad \texttt{PHAS}[j-3] + \Delta'_x[j-3] + \Delta'_y[j-3] \\ \quad \text{(for } j = 3,4,5,6,7,8), \end{cases} \\ 1 & \begin{cases} \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \geq \text{ split threshold} \\ \quad \text{(for } j = 0,1,2) \\ \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \geq \text{ split threshold and} \\ \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \geq \\ \quad \texttt{PHAS}[j-3] + \Delta'_x[j-3] + \Delta'_y[j-3] \\ \quad \text{(for } j = 3,4,5,6,7,8), \end{cases} \end{cases}$$

$$c'_y[j] = \begin{cases} 0 & \begin{cases} \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] < \text{ split threshold or} \\ \texttt{PHAS}[j+3] + \Delta'_x[j+3] + \Delta'_y[j+3] < \text{ split threshold or} \\ j \rightarrow \texttt{CHIPY} = 1 \text{or } 1024 \\ \quad (\text{for } j = 1, 2, 3, 4, 5), \end{cases} \\ \texttt{FRCTRLY} & \begin{cases} \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] > \\ \quad \texttt{PHAS}[j+3] + \Delta'_x[j+3] + \Delta'_y[j+3] \text{ and} \\ \texttt{PHAS}[j+3] + \Delta'_x[j+3] + \Delta'_y[j+3] \geq \text{ split threshold} \\ \quad (\text{for } j = 0, 1, 2, 3, 4, 5), \end{cases} \\ 1 & \begin{cases} \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \leq \\ \quad \texttt{PHAS}[j+3] + \Delta'_x[j+3] + \Delta'_y[j+3] \text{ and} \\ \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \geq \text{ split threshold} \\ \quad (\text{for } j = 0, 1, 2, 3, 4, 5), \end{cases} \end{cases}$$

$$s_y = \text{a temperature dependent scaling factor :}$$
$$\begin{cases} 1 + \texttt{TCTIY}\,(T - \texttt{FP\_TEMPO}), & \text{if } \texttt{mtlfile} \neq \text{“none”} \\ 1, & \text{if } \texttt{mtlfile} = \text{“none”} \end{cases} \tag{553}$$

$$\texttt{TCTIY} = \text{the } \texttt{CCD\_ID} \text{ dependent value in the column } \texttt{TCTIY} \text{ of the } \texttt{ctifile}, \tag{554}$$

$$\rho_y[j] = \text{parallel trap density,} \tag{555}$$
$$\begin{cases} \rho_y[j] \text{ depends upon the } \texttt{CCD\_ID} \text{ and upon the nint}(\texttt{CHIPX\_ADJ}) \text{ and nint}(\texttt{CHIPY\_ADJ}) \\ \quad \text{coordinates associated with element } j \text{ of } \texttt{PHAS\_ADJ}[j] \text{ (see Fig. 1),} \end{cases}$$

$$V_y[j] = \left(\frac{\texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] - \texttt{PHA}_l}{\texttt{PHA}_{l+1} - \texttt{PHA}_l}\right)(\texttt{VOLUME\_Y}_{l+1} - \texttt{VOLUME\_Y}_l) + \texttt{VOLUME\_Y}_l, \tag{556}$$

$$\begin{cases} \texttt{PHA}_l \text{ is the } l^{\text{th}} \text{ element of the column } \texttt{PHA} \text{ in the } \texttt{ctifile}, \\ \texttt{PHA}_l \text{ (and } \texttt{PHA}_{l+1}) \text{ are } \texttt{CCD\_ID} \text{ depdendent,} \\ \texttt{PHA}_l \leq \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j], \\ \text{If } \texttt{PHA}_l > \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \text{ for } l = 0, \text{ then } l = 0, \\ \texttt{PHA}_{l+1} \text{ is the } (l+1)^{\text{th}} \text{ element of the column } \texttt{PHA} \text{ in the } \texttt{ctifile}, \\ \texttt{PHA}_{l+1} > \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j], \\ \text{If } \texttt{PHA}_{l+1} \leq \texttt{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \text{ for } l = n, \text{ where } n \text{ is the last} \\ \quad \text{element, then } l = n, \\ \texttt{VOLUME\_Y}_l \text{ is the } l^{\text{th}} \text{ element of the column } \texttt{VOLUME\_Y} \text{ in the } \texttt{ctifile}, \\ \texttt{VOLUME\_Y}_l, \text{ which is } \texttt{CCD\_ID} \text{ depdendent, is associated with } \texttt{PHA}_l, \\ \texttt{VOLUME\_Y}_{l+1} \text{ is the } (l+1)^{\text{th}} \text{ element of the column } \texttt{VOLUME\_Y} \text{ in the} \\ \quad \texttt{ctifile}, \\ \texttt{VOLUME\_Y}_{l+1}, \text{ which is } \texttt{CCD\_ID} \text{ depdendent, is associated with } \texttt{PHA}_{l+1}, \end{cases}$$

and T, t', t'_k, and t'_{k+1}, are given by equations. 528, 529, 530, and 531, respectively.

vi. The CTI-adjusted pulse heights

$$\texttt{PHAS\_ADJ}[j] = \texttt{PHAS}[j] + \Delta_x[j] + \Delta_y[j] \tag{557}$$

for all j.

vii. A. If

$$|\texttt{PHAS\_ADJ}'[j] - \texttt{PHAS\_ADJ}[j]| < \texttt{cticonverge} \text{ (for all } j) \text{ and} \tag{558}$$
$$n \leq \texttt{max\_cti\_iter}, \tag{559}$$

then the computation of PHAS_ADJ is complete for the event.

B. If

$$|\texttt{PHAS\_ADJ}'[j] - \texttt{PHAS\_ADJ}[j]| \geq \texttt{cticonverge} \text{ (for one or more } j) \text{ and} \tag{560}$$
$$n < \texttt{max\_cti\_iter}, \tag{561}$$

then $n = n + 1$ and steps 1.5.19(a)iii–1.5.19(a)vii are repeated.

C. If

$$|\texttt{PHAS\_ADJ}'[j] - \texttt{PHAS\_ADJ}[j]| \geq \texttt{cticonverge} \text{ (for one or more } j\text{) and} \quad (562)$$
$$n \geq \texttt{max\_cti\_iter}, \quad (563)$$

then no additional iterations are performed, the values of PHAS_ADJ[j] from the most recent iteration are used as are, and STATUS[k] = 1 for bit $k = 20$ (of 0–31) to indicate that the CTI adjustment did not converge.

20. FLTGRADE:

(a) If HDU $h_{\rm in}$ of the `infile` includes the column PHAS and

$$\texttt{apply\_cti} = \text{yes and} \quad (564)$$
$$\texttt{doevtgrade} = \text{yes,} \quad (565)$$

then

$$\texttt{FLTGRADE} = c_{\rm f}[0] + 2c_{\rm f}[1] + 4c_{\rm f}[2] + 8c_{\rm f}[3] + \quad (566)$$
$$16c_{\rm f}[5] + 32c_{\rm f}[6] + 64c_{\rm f}[7] + 128c_{\rm f}[8]. \quad (567)$$

The values of $c_{\rm f}[j]$ are determined as follows.

i. Each value of $c_{\rm f}[j]$ is initialized such that

$$c_{\rm f}[j] = 1. \quad (568)$$

ii. If

$$\texttt{PHAS\_ADJ}[j] < \text{split threshold,} \quad (569)$$

then

$$c_{\rm f}[j] = 0. \quad (570)$$

The elements $j = 0$–8 of PHAS_ADJ are depicted in Figure 1.

(b) If HDU $h_{\rm in}$ of the `infile` includes the column PHAS and

$$\texttt{apply\_cti} = \text{no and} \quad (571)$$
$$\texttt{doevtgrade} = \text{yes,} \quad (572)$$

then

$$\texttt{FLTGRADE} = \texttt{FLTGRADE\_RO}. \quad (573)$$

(c) If HDU $h_{\rm in}$ of the `infile` does not include the column PHAS or

$$\texttt{doevtgrade} = \text{no,} \quad (574)$$

then

i. If HDU $h_{\rm in}$ of the `infile` includes the column FLTGRADE, then

$$\texttt{FLTGRADE} = \texttt{FLTGRADE}_{\rm in}. \quad (575)$$

ii. If HDU $h_{\rm in}$ of the `infile` does not include the column `FLTGRADE`, then

$$\texttt{FLTGRADE} = \text{NULL}. \tag{576}$$

21. `GRADE`:

(a) If

$$\texttt{FLTGRADE} \neq \text{NULL and} \tag{577}$$
$$\texttt{gradefile} \neq \text{none and} \tag{578}$$

HDU $h_{\rm grade}$ of the `gradefile` has a row r, where

$$\texttt{FLTGRADE}_{\rm grade}[r] = \texttt{FLTGRADE}, \tag{579}$$

then

$$\texttt{GRADE} = \texttt{GRADE}_{\rm grade}[r]. \tag{580}$$

(b) If

$$\texttt{FLTGRADE} = \text{NULL or} \tag{581}$$
$$\texttt{gradefile} = \text{none or} \tag{582}$$

HDU $h_{\rm grade}$ of the `gradefile` does not have a row r, where

$$\texttt{FLTGRADE}_{\rm grade}[r] = \texttt{FLTGRADE}, \tag{583}$$

then

$$\texttt{GRADE} = \text{NULL}. \tag{584}$$

22. `PHA`:

(a) Timed mode, CTI adjusted, not NULL:
If

$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_FAINT or} \tag{585}$$
$$\texttt{DATAMODE}_{\rm in} = \text{FAINT or} \tag{586}$$
$$\texttt{DATAMODE}_{\rm in} = \text{FAINT\_BIAS or} \tag{587}$$
$$\texttt{DATAMODE}_{\rm in} = \text{VFAINT}, \tag{588}$$

and

$$\texttt{doevtgrade} = \text{yes, and} \tag{589}$$
$$\texttt{apply\_cti} = \text{yes}, \tag{590}$$

and

$$\texttt{CORNERS} \neq 2 \text{ or} \tag{591}$$
$$\texttt{GRADE} \neq \text{NULL}, \tag{592}$$

then

$$\texttt{PHA} = \sum_{j=0}^{8} c_{\rm p}[j]\,\texttt{PHAS\_ADJ}[j]. \tag{593}$$

The elements $j = 0$–8 of `PHAS_ADJ` are depicted in Figure 1. The values of $c_{\rm p}[j]$ are determined as follows.

i. The values are initialized such that

$$c_{\mathrm{p}}[j] \quad = \quad 1 \tag{594}$$

for $j = 0$–8.

ii. If

$$\texttt{PHAS\_ADJ}[j] \quad < \quad \text{split threshold}, \tag{595}$$

then

$$c_{\mathrm{p}}[j] \quad = \quad 0. \tag{596}$$

iii. If

$$\texttt{CORNERS} \quad = \quad -1, \tag{597}$$

then

$$c_{\mathrm{p}}[0] = c_{\mathrm{p}}[2] = c_{\mathrm{p}}[6] = c_{\mathrm{p}}[8] = 0. \tag{598}$$

iv. If

$$\texttt{CORNERS} \quad = \quad 0, \tag{599}$$

then there are no additional constraints on $c_{\mathrm{p}}[0]$, $c_{\mathrm{p}}[2]$, $c_{\mathrm{p}}[6]$, and $c_{\mathrm{p}}[8]$.

v. If

$$\texttt{CORNERS} \quad = \quad 1, \tag{600}$$

then

A. If

$$c_{\mathrm{p}}[1] = c_{\mathrm{p}}[3] = 0, \tag{601}$$

then

$$c_{\mathrm{p}}[0] \quad = \quad 0. \tag{602}$$

B. If

$$c_{\mathrm{p}}[1] = c_{\mathrm{p}}[5] = 0, \tag{603}$$

then

$$c_{\mathrm{p}}[2] \quad = \quad 0. \tag{604}$$

C. If

$$c_{\mathrm{p}}[3] = c_{\mathrm{p}}[7] = 0, \tag{605}$$

then

$$c_{\mathrm{p}}[6] \quad = \quad 0. \tag{606}$$

D. If

$$c_\text{p}[5] = c_\text{p}[7] = 0, \tag{607}$$

then

$$c_\text{p}[8] = 0. \tag{608}$$

vi. If

$$\texttt{CORNERS} = 2, \tag{609}$$

then

A. If

$$c_\text{p}[1] = 0 \text{ or} \tag{610}$$
$$c_\text{p}[3] = 0 \text{ or} \tag{611}$$
$$\texttt{GRADE} \neq 6, \tag{612}$$

then

$$c_\text{p}[0] = 0. \tag{613}$$

B. If

$$c_\text{p}[1] = 0 \text{ or} \tag{614}$$
$$c_\text{p}[5] = 0 \text{ or} \tag{615}$$
$$\texttt{GRADE} \neq 6, \tag{616}$$

then

$$c_\text{p}[2] = 0. \tag{617}$$

C. If

$$c_\text{p}[3] = 0 \text{ or} \tag{618}$$
$$c_\text{p}[7] = 0 \text{ or} \tag{619}$$
$$\texttt{GRADE} \neq 6, \tag{620}$$

then

$$c_\text{p}[6] = 0. \tag{621}$$

D. If

$$c_\text{p}[5] = 0 \text{ or} \tag{622}$$
$$c_\text{p}[7] = 0 \text{ or} \tag{623}$$
$$\texttt{GRADE} \neq 6, \tag{624}$$

then

$$c_\text{p}[8] = 0. \tag{625}$$

(b) Timed mode, CTI adjusted, NULL:
If

$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_FAINT or} \tag{626}$$
$$\texttt{DATAMODE}_{\rm in} = \text{FAINT or} \tag{627}$$
$$\texttt{DATAMODE}_{\rm in} = \text{FAINT\_BIAS or} \tag{628}$$
$$\texttt{DATAMODE}_{\rm in} = \text{VFAINT,} \tag{629}$$

and

$$\texttt{doevtgrade} = \text{yes, and} \tag{630}$$
$$\texttt{apply\_cti} = \text{yes,} \tag{631}$$

and

$$\texttt{CORNERS} = 2 \text{ and} \tag{632}$$
$$\texttt{GRADE} = \text{NULL,} \tag{633}$$

then

$$\texttt{PHA} = \text{NULL.} \tag{634}$$

(c) Timed mode, not CTI adjusted, not NULL:
If

$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_FAINT or} \tag{635}$$
$$\texttt{DATAMODE}_{\rm in} = \text{FAINT or} \tag{636}$$
$$\texttt{DATAMODE}_{\rm in} = \text{FAINT\_BIAS or} \tag{637}$$
$$\texttt{DATAMODE}_{\rm in} = \text{VFAINT,} \tag{638}$$

and

$$\texttt{doevtgrade} = \text{yes, and} \tag{639}$$
$$\texttt{apply\_cti} = \text{no,} \tag{640}$$

and

$$\texttt{CORNERS} \neq 2 \text{ or} \tag{641}$$
$$\texttt{GRADE} \neq \text{NULL,} \tag{642}$$

then

$$\texttt{PHA} = \sum_{j=0}^{8} c_{\rm p}[j]\,\texttt{PHAS}[j]. \tag{643}$$

The elements $j = 0$–8 of PHAS are depicted in Figure 1. The values of $c_{\rm p}[j]$ are determined as follows.

i. The values are initialized such that

$$c_{\rm p}[j] = 1 \tag{644}$$

for $j = 0$–8.

ii. If

$$\texttt{PHAS}[j] \quad < \quad \text{split threshold}, \tag{645}$$

then

$$c_{\rm p}[j] \quad = \quad 0. \tag{646}$$

iii. If

$$\texttt{CORNERS} \quad = \quad -1, \tag{647}$$

then

$$c_{\rm p}[0] = c_{\rm p}[2] = c_{\rm p}[6] = c_{\rm p}[8] = 0. \tag{648}$$

iv. If

$$\texttt{CORNERS} \quad = \quad 0, \tag{649}$$

then there are no additional constraints on $c_{\rm p}[0]$, $c_{\rm p}[2]$, $c_{\rm p}[6]$, and $c_{\rm p}[8]$.

v. If

$$\texttt{CORNERS} \quad = \quad 1, \tag{650}$$

then

A. If

$$c_{\rm p}[1] = c_{\rm p}[3] = 0, \tag{651}$$

then

$$c_{\rm p}[0] \quad = \quad 0. \tag{652}$$

B. If

$$c_{\rm p}[1] = c_{\rm p}[5] = 0, \tag{653}$$

then

$$c_{\rm p}[2] \quad = \quad 0. \tag{654}$$

C. If

$$c_{\rm p}[3] = c_{\rm p}[7] = 0, \tag{655}$$

then

$$c_{\rm p}[6] \quad = \quad 0. \tag{656}$$

D. If

$$c_{\rm p}[5] = c_{\rm p}[7] = 0, \tag{657}$$

then

$$c_{\rm p}[8] \quad = \quad 0. \tag{658}$$

vi. If

$$\texttt{CORNERS} = 2, \tag{659}$$

then

A. If

$$c_{\rm p}[1] = 0 \text{ or} \tag{660}$$
$$c_{\rm p}[3] = 0 \text{ or} \tag{661}$$
$$\texttt{GRADE} \neq 6, \tag{662}$$

then

$$c_{\rm p}[0] = 0. \tag{663}$$

B. If

$$c_{\rm p}[1] = 0 \text{ or} \tag{664}$$
$$c_{\rm p}[5] = 0 \text{ or} \tag{665}$$
$$\texttt{GRADE} \neq 6, \tag{666}$$

then

$$c_{\rm p}[2] = 0. \tag{667}$$

C. If

$$c_{\rm p}[3] = 0 \text{ or} \tag{668}$$
$$c_{\rm p}[7] = 0 \text{ or} \tag{669}$$
$$\texttt{GRADE} \neq 6, \tag{670}$$

then

$$c_{\rm p}[6] = 0. \tag{671}$$

D. If

$$c_{\rm p}[5] = 0 \text{ or} \tag{672}$$
$$c_{\rm p}[7] = 0 \text{ or} \tag{673}$$
$$\texttt{GRADE} \neq 6, \tag{674}$$

then

$$c_{\rm p}[8] = 0. \tag{675}$$

(d) Timed mode, not CTI adjusted, NULL:
If

$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_FAINT or} \tag{676}$$
$$\texttt{DATAMODE}_{\rm in} = \text{FAINT or} \tag{677}$$
$$\texttt{DATAMODE}_{\rm in} = \text{FAINT\_BIAS or} \tag{678}$$
$$\texttt{DATAMODE}_{\rm in} = \text{VFAINT}, \tag{679}$$

and

$$\texttt{doevtgrade} = \text{yes, and} \tag{680}$$
$$\texttt{apply\_cti} = \text{no,} \tag{681}$$

and

$$\texttt{CORNERS} = 2 \text{ and} \tag{682}$$
$$\texttt{GRADE} = \text{NULL,} \tag{683}$$

then

$$\texttt{PHA} = \text{NULL.} \tag{684}$$

(e) Graded mode, CTI adjusted:
If

$$\texttt{DATAMODE}_{\text{in}} = \text{CC33\_GRADED or} \tag{685}$$
$$\texttt{DATAMODE}_{\text{in}} = \text{GRADED,} \tag{686}$$

and

$$\texttt{doevtgrade} = \text{yes, and} \tag{687}$$
$$\texttt{apply\_cti} = \text{yes,} \tag{688}$$

then

i. The initial pulse height adjustment

$$\Delta_y = s_y \rho_y V_y, \tag{689}$$

where s_y is given by equation 553, ρ_y is the parallel trap density associated with CCD_ID, CHIPX_ADJ, and CHIPY_ADJ, and V_y is given by

$$V_y = \left(\frac{\texttt{PHA\_RO} - \texttt{PHA}_l}{\texttt{PHA}_{l+1} - \texttt{PHA}_l}\right)(\texttt{VOLUME\_Y}_{l+1} - \texttt{VOLUME\_Y}_l) + \texttt{VOLUME\_Y}_l \tag{690}$$

(see equation 556).

ii. The approximate energy associated with $\texttt{PHA\_RO} + \Delta_y$ is given by

$$\texttt{ENERGY}_{\text{approx}} = \left(\frac{\texttt{PHA\_RO} + \Delta_y - \texttt{PHA}_{\text{gain},r}[k]}{\texttt{PHA}_{\text{gain},r}[k+1] - \texttt{PHA}_{\text{gain},r}[k]}\right)(\texttt{ENERGY}_{\text{gain},r}[k+1] - \texttt{ENERGY}_{\text{gain},r}[k]) + \texttt{ENERGY}_{\text{gain},r}[k] \tag{691}$$

(see equation 745).

iii. The HDU h of the grade_image_file is identified such that

$$\texttt{CCD\_ID} = \texttt{CCD\_ID}_{\text{grdimg},h}, \tag{692}$$

where $\texttt{CCD\_ID}_{\text{grdimg},h}$ is the value of the keyword CCD_ID in HDU h of the grade_image_file.

iv. The variable REF_EN is set equal to the value of the keyword of the same name in HDU h of the grade_image_file.

v. The variable REF_EN is converted to units of eV.

vi. The row r of HDU h of the `grade_image_file` is identified such that

$$\texttt{FLTGRADE} = \texttt{FLTGRADE}_{\mathrm{grdimg},h,r}, \quad (693)$$

where $\texttt{FLTGRADE}_{\mathrm{grdimg},h}$ is the column FLTGRADE in HDU h of the `grade_image_file`.

vii. The variable

$$\texttt{ESCL} = \texttt{ESCL}_{\mathrm{grdimg},h,r}, \quad (694)$$

where $\texttt{ESCL}_{\mathrm{grdimg},h}$ is the column ESCL in HDU h of the `grade_image_file`.

viii. The array

$$\texttt{GRDIMG}[j] = \texttt{GRDIMG}_{\mathrm{grdimg},h,r}[j], \quad (695)$$

for $j = 0$–8, where $\texttt{GRDIMG}_{\mathrm{grdimg},h}$ is the column named GRDIMG in HDU h of the `grade_image_file`.

ix. If

$$\texttt{GRDIMG}[j] < 0, \quad (696)$$

then

$$\texttt{GRDIMG}[j] = 0. \quad (697)$$

x. The fraction of the charge in the outer eight pixels of a 3 pixel × 3 pixel event island at the energy REF_EN is given by

$$f_{\mathrm{grdimg,ref}} = 1 - \texttt{GRDIMG}[4]. \quad (698)$$

xi. If

$$f_{\mathrm{grdimg,ref}} < 0, \quad (699)$$

then

$$f_{\mathrm{grdimg,ref}} = 0. \quad (700)$$

xii. At the approximate energy $\texttt{ENERGY}_{\mathrm{approx}}$, this fraction is

$$f_{\mathrm{grdimg,approx}} = f_{\mathrm{grdimg,ref}} \left(\frac{\texttt{ENERGY}_{\mathrm{approx}}}{\texttt{REF\_EN}} \right)^{\texttt{ESCL}}. \quad (701)$$

xiii. Perform the following steps (1.5.3.22(e)xiiiA–1.5.3.22(e)xiiiH) three times.

A. For $j = 0$–3 and $j = 5$–8 (i.e. for the outer eight pixels of a 3 pixel × 3 pixel event island),

$$\texttt{PHAS\_ADJ}[j] = \left(\frac{f_{\mathrm{grdimg,approx}}}{f_{\mathrm{grdimg,ref}}} \right) \texttt{GRDIMG}[j] \left(\texttt{PHA\_RO} + \Delta_y \right). \quad (702)$$

B. For $j = 4$,

$$\texttt{PHAS\_ADJ}[4] = \left(1 - f_{\mathrm{grdimg,approx}}\right) \left(\texttt{PHA\_RO} + \Delta_y \right). \quad (703)$$

C. For $j = 0$–8, if

$$\texttt{PHAS\_ADJ}[j] \geq \text{split threshold}, \quad (704)$$

then

$$V_y[j] = \left(\frac{\texttt{PHAS\_ADJ}[j] - \texttt{PHA}_l}{\texttt{PHA}_{l+1} - \texttt{PHA}_l} \right) \left(\texttt{VOLUME\_Y}_{l+1} - \texttt{VOLUME\_Y}_l \right) + \texttt{VOLUME\_Y}_l \quad (705)$$

(see equation 556).

D. For $j = 0$–8, if

$$\texttt{PHAS\_ADJ}[j] \quad < \quad \text{split threshold}, \tag{706}$$

then

$$V_y[j] \quad = \quad 0. \tag{707}$$

E. For $j = 0$–2,

$$V_y[j+6] \quad = \quad V_y[j+6] - V_y[j+3] - V_y[j]. \tag{708}$$

F. For $j = 0$–2,

$$V_y[j+3] \quad = \quad V_y[j+3] - V_y[j]. \tag{709}$$

G. For $j = 0$–8, if

$$V_y[j] \quad < \quad 0, \tag{710}$$

then

$$V_y[j] \quad = \quad 0. \tag{711}$$

H. The iterated value of the pulse height adjustment

$$\Delta_y \quad = \quad \sum_{j=0}^{8} s_y \rho_y[j] V_y[j], \tag{712}$$

where s_y is given by equation 553, $\rho_y[0] = \rho_y[3] = \rho_y[6]$ is the parallel trap density associated with CCD_ID, CHIPX_ADJ-1, and CHIPY_ADJ, $\rho_y[1] = \rho_y[4] = \rho_y[7]$ is the parallel trap density associated with CCD_ID, CHIPX_ADJ, and CHIPY_ADJ, $\rho_y[2] = \rho_y[5] = \rho_y[8]$ is the parallel trap density associated with CCD_ID, CHIPX_ADJ$+1$, and CHIPY_ADJ, and $V_y[j]$ is given by equation 705.

xiv. The CTI adjusted pulse height

$$\texttt{PHA} \quad = \quad \text{nint}\left(\texttt{PHA\_RO} + \Delta_y\right), \tag{713}$$

where $\text{nint}(\texttt{PHA\_RO} + \Delta_y)$ indicates that $(\texttt{PHA\_RO} + \Delta_y)$ is rounded to the nearest integer.

(f) Graded mode, not CTI adjusted:
If

$$\texttt{DATAMODE}_{\text{in}} \quad = \quad \text{CC33\_GRADED or} \tag{714}$$

$$\texttt{DATAMODE}_{\text{in}} \quad = \quad \text{GRADED}, \tag{715}$$

and

$$\texttt{doevtgrade} \quad = \quad \text{yes, and} \tag{716}$$

$$\texttt{apply\_cti} \quad = \quad \text{no}, \tag{717}$$

then

$$\texttt{PHA} \quad = \quad \texttt{PHA\_RO}_{\text{in}}. \tag{718}$$

(g) $\texttt{doevtgrade} = \text{no}$, PHA exists:
If

$$\texttt{doevtgrade} = \text{no} \tag{719}$$

and HDU h_{in} of the infile includes the column PHA, then

$$\texttt{PHA} = \texttt{PHA}_{\text{in}}. \tag{720}$$

(h) $\texttt{doevtgrade} = \text{no}$, PHA missing:
If

$$\texttt{doevtgrade} = \text{no} \tag{721}$$

and HDU h_{in} of the infile does not include the column PHA, then

$$\texttt{PHA} = \text{NULL}. \tag{722}$$

(i) If

$$\texttt{PHA} \neq \text{NULL and} \tag{723}$$

$$\texttt{PHA} \geq 32767, \tag{724}$$

then $\texttt{STATUS}[k] = 1$ for bit $k = 3$ (of 0–31).

23. Time-dependent gain:

(a) If

$$\texttt{apply\_tgain} = \text{yes}, \tag{725}$$

then

$$\texttt{PHA} = \texttt{PHA} - \text{int}\left[\left(\frac{\texttt{TIME} - \texttt{EPOCH1}}{\texttt{EPOCH2} - \texttt{EPOCH1}}\right)(\delta_2 - \delta_1) + \delta_1 - \epsilon\right], \tag{726}$$

where

$$\text{int} = \text{the integer portion of (i.e. truncate or round down)}, \tag{727}$$

$$\texttt{TIME} = \text{the time of the event}, \tag{728}$$

$$\texttt{EPOCH1} = \text{a keyword in the } \texttt{tgainfile}, \tag{729}$$

$$\texttt{EPOCH2} = \text{a keyword in the } \texttt{tgainfile}, \tag{730}$$

$$\delta_1 = \left(\frac{\texttt{PHA} - \texttt{PHA}_m[r]}{\texttt{PHA}_{m+1}[r] - \texttt{PHA}_m[r]}\right)(\texttt{DELTPHA1}_{m+1}[r] - \texttt{DELTPHA1}_m[r]) + \tag{731}$$

$$\texttt{DELTPHA1}_m[r], \tag{732}$$

$$\left\{\begin{array}{l} r \text{ is the row of the } \texttt{tgainfile} \text{ where} \\ \left\{\begin{array}{l} \texttt{CCD\_ID}[r] = \texttt{CCD\_ID}, \\ \texttt{CHIPX\_LO}[r] \leq \text{nint}(\texttt{CHIPX\_ADJ}), \\ \texttt{CHIPX\_HI}[r] \geq \text{nint}(\texttt{CHIPX\_ADJ}), \\ \texttt{CHIPY\_LO}[r] \leq \text{nint}(\texttt{CHIPY\_ADJ}), \text{ and} \\ \texttt{CHIPY\_HI}[r] \geq \text{nint}(\texttt{CHIPY\_ADJ}). \end{array}\right. \\ m \text{ is the element of row } r \text{ where} \\ \left\{\begin{array}{l} \texttt{PHA}_m[r] \leq \texttt{PHA} \text{ and} \\ \texttt{PHA}_{m+1}[r] > \texttt{PHA}. \\ \text{If } \texttt{PHA} < \texttt{PHA}_m[r] \text{ for } m = 0, \text{ then } m = 0. \\ \text{If } \texttt{PHA} \geq \texttt{PHA}_m[r] \text{ for } m = M \text{ and } M \text{ is the last element of } \texttt{PHA}[r], \\ \quad \text{then } m = M - 1. \end{array}\right. \\ \text{The } \texttt{tgainfile} \text{ includes a binary table with columns named} \\ \quad \texttt{CCD\_ID}, \texttt{CHIPX\_LO}, \texttt{CHIPX\_HI}, \texttt{CHIPY\_LO}, \texttt{CHIPY\_HI}, \texttt{PHA}, \texttt{DELTPHA1}, \text{ and} \\ \quad \texttt{DELTPHA2}. \end{array}\right. \tag{733}$$

$$\delta_2 = \left(\frac{\texttt{PHA} - \texttt{PHA}_m[r]}{\texttt{PHA}_{m+1}[r] - \texttt{PHA}_m[r]}\right)(\texttt{DELTPHA2}_{m+1}[r] - \texttt{DELTPHA2}_m[r]) + \quad (734)$$
$$\texttt{DELTPHA2}_m[r], \quad (735)$$
$$\epsilon = \text{is a uniform random deviate in the range } [0, 1), \quad (736)$$
$$\{ \text{ If } \texttt{rand\_pha} = \text{no, then } \epsilon = 0. \quad (737)$$

(b) If

$$\texttt{PHA} \neq \text{NULL and} \quad (738)$$
$$\texttt{PHA} \geq 32767, \quad (739)$$

then $\texttt{STATUS}[k] = 1$ for bit $k = 3$ (of 0–31).

24. CORN_PHA:

(a) If

$$\texttt{DATAMODE}_{\text{in}} = \text{CC33\_GRADED}, \quad (740)$$

then the value of CORN_PHA is read from the infile.

25. ENERGY:

(a) If the parameter calculate_pi = yes and the parameter gainfile is specified and PHA > 0, then

i. The row r in the gainfile is identified such that

$$\texttt{CCD\_ID} = \texttt{CCD\_ID}_{\text{gain},r}, \quad (741)$$
$$\texttt{CHIPX\_MIN}_{\text{gain},r} \leq \text{nint}(\texttt{CHIPX\_ADJ}) \leq \texttt{CHIPX\_MAX}_{\text{gain},r}, \quad \text{and} \quad (742)$$
$$\texttt{CHIPY\_MIN}_{\text{gain},r} \leq \text{nint}(\texttt{CHIPY\_ADJ}) \leq \texttt{CHIPY\_MAX}_{\text{gain},r}, \quad (743)$$

where $\texttt{CCD\_ID}_{\text{gain}}$, $\texttt{CHIPX\_MIN}_{\text{gain}}$, $\texttt{CHIPX\_MAX}_{\text{gain}}$, $\texttt{CHIPY\_MIN}_{\text{gain}}$, and $\texttt{CHIPY\_MAX}_{\text{gain}}$ are columns in the gainfile.

ii. A uniform random deviate Δp is computed over the interval from $[-0.5, +0.5)$.

iii. The element k of row r of $\texttt{PHA}_{\text{gain}}$ is identified such that

$$\texttt{PHA}_{\text{gain},r}[k] \leq (\texttt{PHA} + \Delta p) < \texttt{PHA}_{\text{gain},r}[k+1], \quad (744)$$

where $\texttt{PHA}_{\text{gain}}$ is a vector column in the gainfile. If $\texttt{PHA} + \Delta p < \texttt{PHA}_{\text{gain},r}[0]$, then $k = 0$. If $\texttt{PHA}_{\text{gain},r}[\texttt{NPOINTS} - 2] \leq \texttt{PHA} + \Delta p$, then $k = \texttt{NPOINTS} - 2$, where NPOINTS is a column in the gainfile.

iv. The ENERGY of an event is computed from the PHA of the event:

$$\texttt{ENERGY} = \left(\frac{\texttt{PHA} + \Delta p - \texttt{PHA}_{\text{gain},r}[k]}{\texttt{PHA}_{\text{gain},r}[k+1] - \texttt{PHA}_{\text{gain},r}[k]}\right)(\texttt{ENERGY}_{\text{gain},r}[k+1] - \texttt{ENERGY}_{\text{gain},r}[k]) +$$
$$\texttt{ENERGY}_{\text{gain},r}[k], \quad (745)$$

where $\texttt{ENERGY}_{\text{gain}}$ is a vector column in the gainfile.

v. If ENERGY < 0, then ENERGY = 0.

(b) If the parameter calculate_pi = yes and the parameter gainfile is specified and PHA ≤ 0, then ENERGY = 0.

(c) If the parameter calculate_pi = no or if the parameter gainfile is not specified, then

i. If the infile includes the ENERGY of an event, then the ENERGY of the event is equal to the ENERGY in the infile.

ii. If the `infile` does not include the `ENERGY` of an event, then `ENERGY` $= 0$.

26. `PI`:

(a) If

$$\texttt{calculate\_pi} = \text{yes}, \tag{746}$$

then

i.

$$\texttt{PI} = \text{int}\left(\frac{\texttt{ENERGY}}{\texttt{pi\_bin\_width}}\right) + 1, \tag{747}$$

where "int" indicates the integer portion of what is in parentheses (i.e. the value is truncated or rounded down).

ii. If

$$\texttt{PI} < 1, \tag{748}$$

then `PI` $= 1$.

iii. If

$$\texttt{PI} > \texttt{pi\_num\_bins}, \tag{749}$$

then `PI` = `pi_num_bins`.

(b) If

$$\texttt{calculate\_pi} = \text{no} \tag{750}$$

and the `infile` includes the value of `PI` for an event, then the value of `PI` is read from the `infile`.

27. `pix_adj`:

(a) centroid:

i. If

$$\texttt{pix\_adj} = \text{centroid} \tag{751}$$

and

$$\begin{aligned} \texttt{DATAMODE}_{\text{in}} &= \text{CC33\_FAINT or} & (752)\\ \texttt{DATAMODE}_{\text{in}} &= \text{FAINT or} & (753)\\ \texttt{DATAMODE}_{\text{in}} &= \text{FAINT\_BIAS or} & (754)\\ \texttt{DATAMODE}_{\text{in}} &= \text{VFAINT}, & (755) \end{aligned}$$

then

$$\begin{aligned} \texttt{CHIPX\_ADJ} &= \texttt{CHIPX\_ADJ} - w'[0] + w'[2] - w'[3] + w'[5] - w'[6] + w'[8] \text{ and} & (756)\\ \texttt{CHIPY\_ADJ} &= \texttt{CHIPY\_ADJ} - w'[0] - w'[1] - w'[2] + w'[6] + w'[7] + w'[8], & (757) \end{aligned}$$

where

$$w'[j] = \frac{w[j]}{\sum_{j=0}^{8} w[j]}, \tag{758}$$

$$w[j] = \begin{cases} p[j] & \text{if the pixel is valid} \\ 0 & \text{if the pixel is invalid,} \end{cases} \tag{759}$$

$$p[j] = \begin{cases} \texttt{PHAS\_ADJ}[j] & \text{if } \texttt{apply\_cti} = \text{yes} \\ \texttt{PHAS}[j] & \text{if } \texttt{apply\_cti} = \text{no,} \end{cases} \tag{760}$$

and the pixel is invalid if

$$
\begin{aligned}
\beta[j] &= 0 \text{ or} && (761)\\
\texttt{STATUS}[0] &= 1 \text{ or} && (762)\\
\texttt{STATUS}[1] &= 1 \text{ or} && (763)\\
\texttt{STATUS}[2] &= 1 \text{ or} && (764)\\
\texttt{STATUS}[3] &= 1 \text{ or} && (765)\\
\texttt{STATUS}[4] &= 1 \text{ or} && (766)\\
\texttt{STATUS}[11] &= 1 \text{ or} && (767)\\
\texttt{STATUS}[13] &= 1 \text{ or} && (768)\\
\texttt{STATUS}[14] &= 1 \text{ or} && (769)\\
\texttt{STATUS}[15] &= 1 \text{ or} && (770)\\
\texttt{STATUS}[16] &= 1. && (771)
\end{aligned}
$$

ii. If

$$
\begin{aligned}
\texttt{pix\_adj} &= \text{centroid and} && (772)\\
\texttt{DATAMODE}_{\text{in}} &= \text{CC33\_FAINT,} && (773)
\end{aligned}
$$

then

$$\texttt{TIME} = \texttt{TIME} + (w'[0] + w'[1] + w'[2] - w'[6] - w'[7] - w'[8]) \times \texttt{TIMEDEL}_{\text{in}}. \quad (774)$$

Note that it is possible for the centroid algorithm to yield adjustments to CHIPX_ADJ and/or CHIPY_ADJ that are greater than half a pixel. However, the adjustment cannot equal or exceed one pixel.

(b) edser:

i. If

$$\texttt{pix\_adj} = \text{edser} \quad (775)$$

and

$$
\begin{aligned}
\texttt{DATAMODE}_{\text{in}} &= \text{CC33\_FAINT or} && (776)\\
\texttt{DATAMODE}_{\text{in}} &= \text{CC33\_GRADED or} && (777)\\
\texttt{DATAMODE}_{\text{in}} &= \text{FAINT or} && (778)\\
\texttt{DATAMODE}_{\text{in}} &= \text{FAINT\_BIAS or} && (779)\\
\texttt{DATAMODE}_{\text{in}} &= \text{GRADED or} && (780)\\
\texttt{DATAMODE}_{\text{in}} &= \text{VFAINT,} && (781)
\end{aligned}
$$

then

A. If

$$
\begin{aligned}
\texttt{ENERGY} &\neq \text{NaN and} && (782)\\
\texttt{ENERGY} &> 0, && (783)
\end{aligned}
$$

then

$$\texttt{CHIPX\_ADJ} = \texttt{CHIPX\_ADJ} + \left(\frac{\texttt{ENERGY} - E[k]}{E[k+1] - E[k]}\right)(\Delta X[k+1] - \Delta X[k]) + \Delta X[k] \quad (784)$$

and

$$\texttt{CHIPY\_ADJ} = \texttt{CHIPY\_ADJ} + \left(\frac{\texttt{ENERGY} - E[k]}{E[k+1] - E[k]}\right)(\Delta Y[k+1] - \Delta Y[k]) + \Delta Y[k], \quad (785)$$

where $E[k]$ and $E[k+1]$, $\Delta X[k]$ and $\Delta X[k+1]$, and $\Delta Y[k]$ and $\Delta Y[k+1]$ are the k and $(k+1)^{th}$ elements of the vector columns $\texttt{ENERGY}_{\rm subpix}$, $\texttt{CHIPX\_OFFSET}_{\rm subpix}$, and $\texttt{CHIPY\_OFFSET}_{\rm subpix}$, respectively. These columns are in the HDU of the `subpixfile` where the value of the keyword `CCD_ID` is equal to the value of the `CCD_ID` of the event. The appropriate row of these columns is the one where $\texttt{FLTGRADE}_{\rm subpix} = \texttt{FLTGRADE}$. The values of k are the ones where

$$\texttt{ENERGY} \geq E[k] \text{ and} \quad (786)$$
$$\texttt{ENERGY} < E[k+1]. \quad (787)$$

Note that if

$$\texttt{ENERGY} \leq E[0], \quad (788)$$

then $k = 0$. Similarly, if

$$\texttt{ENERGY} \geq E[\texttt{NPOINTS}_{\rm subpix} - 2], \quad (789)$$

then $k = \texttt{NPOINTS}_{\rm subpix} - 2$.

B. If

$$\texttt{ENERGY} = \text{NaN or} \quad (790)$$
$$\texttt{ENERGY} \leq 0, \quad (791)$$

then the `CHIPX_ADJ` and `CHIPY_ADJ` coordinates are not modified.

ii. If

$$\texttt{pix\_adj} = \text{edser} \quad (792)$$

and

$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_FAINT or} \quad (793)$$
$$\texttt{DATAMODE}_{\rm in} = \text{CC33\_GRADED}, \quad (794)$$

then

A. If

$$\texttt{ENERGY} \neq \text{NaN and} \quad (795)$$
$$\texttt{ENERGY} > 0, \quad (796)$$

then

$$\texttt{TIME} = \texttt{TIME} - \left(\left(\frac{\texttt{ENERGY} - E[k]}{E[k+1] - E[k]}\right)(\Delta Y[k+1] - \Delta Y[k]) + \Delta Y[k]\right) \times \texttt{TIMEDEL}_{\rm in} \quad (797)$$

B. If

$$\texttt{ENERGY} = \text{NaN or} \quad (798)$$
$$\texttt{ENERGY} \leq 0, \quad (799)$$

then the `TIME` is not modified.

(c) none:
If

$$\texttt{pix\_adj} = \text{none}, \tag{800}$$

then

$$\texttt{CHIPX\_ADJ} = \texttt{CHIPX\_ADJ} \text{ and} \tag{801}$$
$$\texttt{CHIPY\_ADJ} = \texttt{CHIPY\_ADJ} \text{ and} \tag{802}$$
$$\texttt{TIME} = \texttt{TIME}. \tag{803}$$

No sub-pixel adjustments are applied to the values of CHIPX_ADJ and CHIPY_ADJ (for timed exposure mode) or CHIPX_ADJ and TIME (for continuous-clocking mode).

(d) randomize:

i. If

$$\texttt{pix\_adj} = \text{randomize}, \tag{804}$$

then

$$\texttt{CHIPX\_ADJ} = \texttt{CHIPX\_ADJ} + \epsilon_x \text{ and} \tag{805}$$
$$\texttt{CHIPY\_ADJ} = \texttt{CHIPY\_ADJ} + \epsilon_y, \tag{806}$$

where ϵ_x and ϵ_y are a uniform random deviates in the range $[-0.5, +0.5)$ pixel.

ii. If

$$\texttt{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \tag{807}$$
$$\texttt{DATAMODE}_{\text{in}} = \text{CC33\_GRADED}, \tag{808}$$

then

$$\texttt{TIME} = \texttt{TIME} - \epsilon_y \times \texttt{TIMEDEL}_{\text{in}}. \tag{809}$$

(e) If

$$\texttt{CHIPX\_ADJ} < 0.5, \tag{810}$$

then

$$\texttt{CHIPX\_ADJ} = 1. \tag{811}$$

(f) If

$$\texttt{CHIPX\_ADJ} \geq 1024.5, \tag{812}$$

then

$$\texttt{CHIPX\_ADJ} = 1024. \tag{813}$$

(g) If

$$\texttt{CHIPY\_ADJ} < 0.5, \tag{814}$$

then

$$\texttt{CHIPY\_ADJ} = 1. \tag{815}$$

(h) If

$$\texttt{CHIPY\_ADJ} \geq 1024.5, \tag{816}$$

then

$$\texttt{CHIPY\_ADJ} = 1024. \tag{817}$$

28. X and Y:

(a) If

$$\texttt{stop} = \text{sky}, \tag{818}$$

then

i. If

$$\texttt{DATAMODE}_{\text{in}} = \text{FAINT or} \tag{819}$$
$$\texttt{DATAMODE}_{\text{in}} = \text{FAINT\_BIAS or} \tag{820}$$
$$\texttt{DATAMODE}_{\text{in}} = \text{GRADED or} \tag{821}$$
$$\texttt{DATAMODE}_{\text{in}} = \text{VFAINT}, \tag{822}$$

then the values of X and Y are computed using the real-valued coordinates CHIPX_ADJ and CHIPY_ADJ and the orientation of the telescope (i.e. RA, DEC, and ROLL) at the time TIME.

ii. If

$$\texttt{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \tag{823}$$
$$\texttt{DATAMODE}_{\text{in}} = \text{CC33\_GRADED}, \tag{824}$$

then the values of X and Y are computed using the real-valued coordinates CHIPX_ADJ and $\texttt{CHIPY\_TARG}_{\text{eff}}$ and the orientation of the telescope (i.e. RA, DEC, and ROLL) at the time TIME.

iii. If

$$\texttt{acaofffile} \neq \text{none} \tag{825}$$

and

$$\texttt{CONTENT}_{\text{in}} \neq \text{TGEVT1} \tag{826}$$

and

$$\texttt{TIME} < \texttt{TIME}_{\text{min}} \text{ or} \tag{827}$$
$$\texttt{TIME} \geq \texttt{TIME}_{\text{max}}, \tag{828}$$

then

$$\texttt{X} = \text{NaN and} \tag{829}$$
$$\texttt{Y} = \text{NaN}. \tag{830}$$

iv. If

$$\texttt{acaofffile} \neq \text{none} \tag{831}$$

and

$$\begin{aligned}\texttt{CONTENT}_{\text{in}} &= \text{TGEVT1 and} &(832)\\ \texttt{CHIPY\_TG} &= \text{NaN,} &(833)\end{aligned}$$

then

$$\begin{aligned}\texttt{X} &= \text{NaN and} &(834)\\ \texttt{Y} &= \text{NaN.} &(835)\end{aligned}$$

29. SKY_1D:

(a) If

$$\texttt{stop} = \text{sky} \quad (836)$$

and

$$\begin{aligned}\texttt{DATAMODE}_{\text{in}} &= \text{CC33\_FAINT or} &(837)\\ \texttt{DATAMODE}_{\text{in}} &= \text{CC33\_GRADED,} &(838)\end{aligned}$$

then

i. The value of SKY_1D is computed using the real-valued coordinates CHIPX_ADJ and $\texttt{CHIPY\_TARG}_{\text{eff}}$ and the orientation of the telescope (i.e. RA, DEC, and ROLL) at the time TIME.

ii. If

$$\texttt{acaofffile} \neq \text{none} \quad (839)$$

and

$$\texttt{CONTENT}_{\text{in}} \neq \text{TGEVT1} \quad (840)$$

and

$$\begin{aligned}\texttt{TIME} &< \texttt{TIME}_{\text{min}} \text{ or} &(841)\\ \texttt{TIME} &\geq \texttt{TIME}_{\text{max}}, &(842)\end{aligned}$$

then

$$\texttt{SKY\_1D} = \text{NaN.} \quad (843)$$

iii. If

$$\texttt{acaofffile} \neq \text{none} \quad (844)$$

and

$$\begin{aligned}\texttt{CONTENT}_{\text{in}} &= \text{TGEVT1 and} &(845)\\ \texttt{CHIPY\_TG} &= \text{NaN,} &(846)\end{aligned}$$

then

$$\texttt{SKY\_1D} = \text{NaN.} \quad (847)$$

30. DETX and DETY:

(a) If

$$\texttt{stop} = \text{det or} \tag{848}$$
$$\texttt{stop} = \text{tan or} \tag{849}$$
$$\texttt{stop} = \text{sky}, \tag{850}$$

then

i. The values of DETX and DETY are computed using the real-valued coordinates CHIPX_ADJ and CHIPY_ADJ and the orientation of the SIM (i.e. DY, DZ, and DTHETA) at the time TIME.

ii. If

$$\texttt{acaofffile} \neq \text{none} \tag{851}$$

and

$$\texttt{CONTENT}_{\text{in}} \neq \text{TGEVT1} \tag{852}$$

and

$$\texttt{TIME} < \texttt{TIME}_{\text{min}} \text{ or} \tag{853}$$
$$\texttt{TIME} \geq \texttt{TIME}_{\text{max}}, \tag{854}$$

then

$$\texttt{DETX} = \text{NaN and} \tag{855}$$
$$\texttt{DETY} = \text{NaN}. \tag{856}$$

iii. If

$$\texttt{acaofffile} \neq \text{none} \tag{857}$$

and

$$\texttt{CONTENT}_{\text{in}} = \text{TGEVT1 and} \tag{858}$$
$$\texttt{CHIPY\_TG} = \text{NaN}, \tag{859}$$

then

$$\texttt{DETX} = \text{NaN and} \tag{860}$$
$$\texttt{DETY} = \text{NaN}. \tag{861}$$

31. TDETX and TDETY:

(a) If

$$\texttt{stop} = \text{tdet or} \tag{862}$$
$$\texttt{stop} = \text{det or} \tag{863}$$
$$\texttt{stop} = \text{tan or} \tag{864}$$
$$\texttt{stop} = \text{sky} \tag{865}$$

then

i. The values of TDETX and TDETY are computed using the values of nint(CHIPX_ADJ) and nint(CHIPY_ADJ). Here, “nint” indicates that the real-valued coordinate is rounded to the nearest integer.

ii. If

$$\texttt{acaofffile} \neq \text{none} \tag{866}$$

and

$$\texttt{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \tag{867}$$
$$\texttt{DATAMODE}_{\text{in}} = \text{CC33\_GRADED} \tag{868}$$

and

$$\texttt{CONTENT}_{\text{in}} \neq \text{TGEVT1} \tag{869}$$

and

$$\texttt{TIME} < \texttt{TIME}_{\text{min}} \text{ or} \tag{870}$$
$$\texttt{TIME} \geq \texttt{TIME}_{\text{max}}, \tag{871}$$

then

$$\texttt{TDETX} = \text{NULL and} \tag{872}$$
$$\texttt{TDETY} = \text{NULL.} \tag{873}$$

iii. If

$$\texttt{acaofffile} \neq \text{none} \tag{874}$$

and

$$\texttt{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \tag{875}$$
$$\texttt{DATAMODE}_{\text{in}} = \text{CC33\_GRADED} \tag{876}$$

and

$$\texttt{CONTENT}_{\text{in}} = \text{TGEVT1 and} \tag{877}$$
$$\texttt{CHIPY\_TG} = \text{NaN}, \tag{878}$$

then

$$\texttt{TDETX} = \text{NULL and} \tag{879}$$
$$\texttt{TDETY} = \text{NULL.} \tag{880}$$

1.5.4 Write `outfile`

1. Unexpected `CHIPX` values:
   If the counter associated with equations 202 and 203 is greater than zero, then `acis_process_events` produces a warning message that includes the number of events that have unexpected `CHIPX` values.

2. Unexpected `CHIPY` values:
   If the counter associated with equations 228 and 229 is greater than zero, then `acis_process_events` produces a warning message that includes the number of events that have unexpected `CHIPY` values.

3. `PIX_ADJ`:

(a) If

$$\texttt{pix\_adj} = \text{centroid}, \tag{881}$$

then

$$\texttt{PIX\_ADJ} = \text{CENTROID}. \tag{882}$$

(b) If

$$\texttt{pix\_adj} = \text{edser}, \tag{883}$$

then

$$\texttt{PIX\_ADJ} = \text{EDSER}. \tag{884}$$

(c) If

$$\texttt{pix\_adj} = \text{none}, \tag{885}$$

then

$$\texttt{PIX\_ADJ} = \text{NONE}. \tag{886}$$

(d) If

$$\texttt{pix\_adj} = \text{randomize}, \tag{887}$$

then

$$\texttt{PIX\_ADJ} = \text{RANDOMIZE}. \tag{888}$$

4. `RAND_SKY`:

(a) If

$$\texttt{pix\_adj} = \text{centroid}, \tag{889}$$

then

$$\texttt{RAND\_SKY} = 0.0. \tag{890}$$

(b) If

$$\texttt{pix\_adj} = \text{edser}, \tag{891}$$

then

$$\texttt{RAND\_SKY} = 0.0. \tag{892}$$

(c) If

$$\texttt{pix\_adj} = \text{none}, \tag{893}$$

then

$$\texttt{RAND\_SKY} = 0.0. \tag{894}$$

(d) If

$$\texttt{pix\_adj} = \text{randomize,} \quad (895)$$

then

$$\texttt{RAND\_SKY} = 0.5. \quad (896)$$

5. TIME_ADJ:

(a) Timed-exposure mode:
If

$$\texttt{DATAMODE}_{\text{in}} = \text{FAINT or} \quad (897)$$
$$\texttt{DATAMODE}_{\text{in}} = \text{FAINT\_BIAS or} \quad (898)$$
$$\texttt{DATAMODE}_{\text{in}} = \text{GRADED or} \quad (899)$$
$$\texttt{DATAMODE}_{\text{in}} = \text{VFAINT,} \quad (900)$$

then

$$\texttt{TIME\_ADJ} = \text{NONE.} \quad (901)$$

(b) Continuous-clocking mode:

i. Set

$$\texttt{TIME\_ADJ} = \text{MIDCHIP.} \quad (902)$$

ii. If

$$\texttt{OBS\_MODE}_{\text{in}} = \text{pointing and} \quad (903)$$
$$\texttt{acaofffile} \neq \text{none} \quad (904)$$

then

$$\texttt{TIME\_ADJ} = \text{TARGET.} \quad (905)$$

iii. If

$$\texttt{OBS\_MODE}_{\text{in}} = \text{pointing and} \quad (906)$$
$$\texttt{CONTENT}_{\text{in}} = \text{TGEVT1,} \quad (907)$$

then

$$\texttt{TIME\_ADJ} = \text{GRATING.} \quad (908)$$

2 TBD

- Add the graded mode cti adjustment.
- Include all timed exposure mode processing.
- Should CONTENTs other than EVT0, EVT1, TGEVT1, and EVT2 be included?
- Should CONTENT = EVT2 be dropped?
- Should DATAMODEs other than CC33_FAINT, CC33_GRADED, FAINT, FAINT_BIAS, GRADED, and VFAINT be included?
- Are the β in PHA_RO the same as the β in PHA?
- Should something be done about SKY_1D?