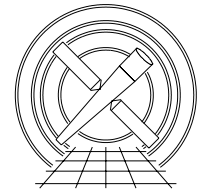




MIT Kavli Institute



Chandra X-Ray Center

## MEMORANDUM

May 21, 2015

**To:** Jonathan McDowell, SDS Group Leader  
**From:** Glenn E. Allen, SDS  
**Subject:** `acis_process_events` spec  
**Revision:** 3.25  
**URL:** <http://space.mit.edu/CXC/docs/docs.html#ape>  
**File:** `/nfs/inconceivable/d0/sds/specs/acis_process_events/ape_spec_3.25.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. Existence:

If

`obsfile`  $\neq$  `none` and (1)

`obsfile`  $\neq$  `NONE` (2)

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

ii. Permission:

If

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

$$\text{obsfile} \neq \text{NONE} \tag{4}$$

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.

iii. OBS\_MODE:

If

$$\text{obsfile} \neq \text{none} \text{ and} \tag{5}$$

$$\text{obsfile} \neq \text{NONE}, \tag{6}$$

then

A. If the `obsfile` does not include the keyword `obs_mode`, then `OBS_MODE` is set to “none”.

B. If the `obsfile` includes the keyword `obs_mode` and

$$\text{obs\_mode} = \text{pointing} \text{ or} \tag{7}$$

$$\text{obs\_mode} = \text{POINTING} \text{ or} \tag{8}$$

$$\text{obs\_mode} = \text{secondary} \text{ or} \tag{9}$$

$$\text{obs\_mode} = \text{SECONDARY}, \tag{10}$$

then `OBS_MODE` is set to the value of `obs_mode`. Hereafter this keyword is referred to as `OBS_MODE`.

C. If the `obsfile` includes the keyword `obs_mode` and

$$\text{obs\_mode} \neq \text{pointing} \text{ and} \tag{11}$$

$$\text{obs\_mode} \neq \text{POINTING} \text{ and} \tag{12}$$

$$\text{obs\_mode} \neq \text{secondary} \text{ and} \tag{13}$$

$$\text{obs\_mode} \neq \text{SECONDARY}, \tag{14}$$

then `OBS_MODE` is set to “none”.

2. infile:

(a) Existence:

If the `infile` does not exist, then `acis_process_events` exits with an error message.

(b) Permission:

If the `infile` exists, but 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 `OBS_MODE = none`, then

A. The `OBS_MODE` is read from the HDU `hin` keyword of the same name. Hereafter this keyword is referred to as `OBS_MODE`.

B. If the HDU `hin` does not include the keyword `OBS_MODE`, then `OBS_MODE` is set to “none” and `acis_process_events` produces a warning message.

C. If the HDU  $h_{in}$  includes the keyword `OBS_MODE` and

$$\text{obs\_mode} \neq \text{pointing and} \quad (15)$$

$$\text{obs\_mode} \neq \text{POINTING and} \quad (16)$$

$$\text{obs\_mode} \neq \text{secondary and} \quad (17)$$

$$\text{obs\_mode} \neq \text{SECONDARY,} \quad (18)$$

then `OBS_MODE` is set to “none” and `acis_process_events` produces a warning message.

ii. `DATAMODE`:

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

$$\text{DATAMODE} \neq \text{CC33\_FAINT and} \quad (19)$$

$$\text{DATAMODE} \neq \text{CC33\_GRADED and} \quad (20)$$

$$\text{DATAMODE} \neq \text{FAINT and} \quad (21)$$

$$\text{DATAMODE} \neq \text{FAINT\_BIAS and} \quad (22)$$

$$\text{DATAMODE} \neq \text{GRADED and} \quad (23)$$

$$\text{DATAMODE} \neq \text{VF\_FAINT,} \quad (24)$$

then `acis_process_events` exits with an error message. Hereafter, the value of this keyword is referred to as `DATAMODEin`.

iii. `CONTENT`:

If the `infile` does not have an HDU  $h_{in}$  with the keyword

$$\text{CONTENT} = \text{EVT0 or} \quad (25)$$

$$\text{CONTENT} = \text{EVT1 or} \quad (26)$$

$$\text{CONTENT} = \text{TGEVT1 or} \quad (27)$$

$$\text{CONTENT} = \text{EVT2,} \quad (28)$$

then `acis_process_events` exits with an error message. Hereafter, the value of this keyword is referred to as `CONTENTin`.

iv. `TIME`:

If HDU  $h_{in}$  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 `TIMEin`.

v. `TIME_R0`:

If

$$\text{DATAMODE}_{in} = \text{CC33\_FAINT or} \quad (29)$$

$$\text{DATAMODE}_{in} = \text{CC33\_GRADED,} \quad (30)$$

if

$$\text{CONTENT}_{in} = \text{EVT1 or} \quad (31)$$

$$\text{CONTENT}_{in} = \text{TGEVT1 or} \quad (32)$$

$$\text{CONTENT}_{in} = \text{EVT2,} \quad (33)$$

and if HDU  $h_{in}$  of the `infile` does not include the column `TIME_R0`, then `acis_process_events` exits with an error message. Hereafter, this column is referred to as `TIME_R0in`.

vi. `EXPNO`:

If HDU  $h_{in}$  the `infile` does not include the column `EXPNO`, then `acis_process_events` exits with an error message. Hereafter, this column is referred to as `EXPNOin`.

vii. CCD\_ID:

A. If

$$\text{CONTENT}_{\text{in}} = \text{EVT0} \quad (34)$$

and if HDU  $h_{\text{in}}$  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 `CCD_IDin`.

B. If

$$\text{CONTENT}_{\text{in}} = \text{EVT1 or} \quad (35)$$

$$\text{CONTENT}_{\text{in}} = \text{TGEVT1 or} \quad (36)$$

$$\text{CONTENT}_{\text{in}} = \text{EVT2} \quad (37)$$

and if HDU  $h_{\text{in}}$  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 `CCD_IDin`.

viii. CCDX:

A. If

$$\text{CONTENT}_{\text{in}} = \text{EVT0} \quad (38)$$

and if HDU  $h_{\text{in}}$  of the `infile` does not include the column `CCDX`, then `acis_process_events` exits with an error message. Hereafter, this column is referred to as `CCDXin`.

ix. CHIPX:

A. If

$$\text{CONTENT}_{\text{in}} = \text{EVT1 or} \quad (39)$$

$$\text{CONTENT}_{\text{in}} = \text{TGEVT1 or} \quad (40)$$

$$\text{CONTENT}_{\text{in}} = \text{EVT2} \quad (41)$$

and if HDU  $h_{\text{in}}$  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 `CHIPXin`.

x. CCDY:

A. If

$$\text{CONTENT}_{\text{in}} = \text{EVT0}, \quad (42)$$

if

$$\text{DATAMODE}_{\text{in}} = \text{FAINT or} \quad (43)$$

$$\text{DATAMODE}_{\text{in}} = \text{FAINT\_BIAS or} \quad (44)$$

$$\text{DATAMODE}_{\text{in}} = \text{GRADED or} \quad (45)$$

$$\text{DATAMODE}_{\text{in}} = \text{VFAINT}, \quad (46)$$

and if HDU  $h_{\text{in}}$  of the `infile` does not include the column `CCDY`, then `acis_process_events` exits with an error message. Hereafter, this column is referred to as `CCDYin`.

xi. TROW:

A. If

$$\text{CONTENT}_{\text{in}} = \text{EVT0}, \quad (47)$$

if

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \quad (48)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED} \quad (49)$$

and if HDU  $h_{\text{in}}$  of the `infile` does not include the column `TROW`, then `acis_process_events` exits with an error message. Hereafter, this column is referred to as `TROWin`.

xii. CHIPY:

A. If

$$\text{CONTENT}_{\text{in}} = \text{EVT1 or} \quad (50)$$

$$\text{CONTENT}_{\text{in}} = \text{TGEVT1 or} \quad (51)$$

$$\text{CONTENT}_{\text{in}} = \text{EVT2} \quad (52)$$

and if HDU  $h_{\text{in}}$  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 `CHIPYin`.

xiii. TIMEDEL:

If

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \quad (53)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED} \quad (54)$$

and HDU  $h_{\text{in}}$  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 `TIMEDELin`.

xiv. `RA_TARG`, `DEC_TARG`, `RA_NOM`, `DEC_NOM`, `RA_PNT`, `DEC_PNT`, `CHIPY_TG`, `CHIPY_Z0`, and `TG_M`:

If

$$\text{OBS\_MODE} = \text{pointing or} \quad (55)$$

$$\text{OBS\_MODE} = \text{POINTING} \quad (56)$$

and

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \quad (57)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED}, \quad (58)$$

then

A. `RA_TARG`, `DEC_TARG`, `RA_NOM`, `DEC_NOM`, `RA_PNT`, `DEC_PNT`:

If HDU  $h_{\text{in}}$  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 `RA_TARGin`, `DEC_TARGin`, `RA_NOMin`, `DEC_NOMin`, `RA_PNTin`, and `DEC_PNTin`, respectively.

B. `CHIPY_TG`, `CHIPY_Z0`, and `TG_M`:

If

$$\text{CONTENT}_{\text{in}} = \text{TGEVT1} \quad (59)$$

and if HDU  $h_{\text{in}}$  of the `infile` does not include the columns `CHIPY_TG`, `CHIPY_Z0`, and `TG_M`, then `acis_process_events` exits with an error message. Hereafter these columns are referred to as `CHIPY_TGin`, `CHIPY_Z0in`, and `TG_Min`, respectively.

3. `stop`:

(a) Lowercase:

The parameter string is converted to contain only lower case letters.

(b) Validation:

i. Setting:

If

$$\text{stop} \neq \text{none and} \quad (60)$$

`stop`  $\neq$  `chip` and (61)

`stop`  $\neq$  `tdet` and (62)

`stop`  $\neq$  `det` and (63)

`stop`  $\neq$  `tan` and (64)

`stop`  $\neq$  `sky`, (65)

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

ii. `OBS_MODE`:

If

`OBS_MODE`  $\neq$  `pointing` and (66)

`OBS_MODE`  $\neq$  `POINTING` (67)

and

`stop`  $\neq$  `none` and (68)

`stop`  $\neq$  `chip` and (69)

`stop`  $\neq$  `tdet`, (70)

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

4. `acaofffile`:

(a) Validation for CC mode:

If

`OBS_MODE` = `pointing` or (71)

`OBS_MODE` = `POINTING` (72)

and

`DATAMODEin` = `CC33_FAINT` or (73)

`DATAMODEin` = `CC33_GRADED`, (74)

then

i. Setting:

If

`acaofffile` = `none` or (75)

`acaofffile` = `NONE`, (76)

then `acis_process_events` exits with an error message.

ii. Existence:

If the `acaofffile` does not exist, then `acis_process_events` exits with an error message.

iii. Permission:

If the `acaofffile` exists, but the file permissions do not allow it to be read, then `acis_process_events` exits with an error message.

iv. `CONTENT`:

If the `acaofffile` does not have an HDU `hacaoff` with the keyword

`CONTENT` = `ASPSOL`, (77)

then `acis_process_events` exits with an error message.

- v. Keyword:  
If HDU  $h_{\text{acaoff}}$  of the `acaofffile` does not include the keyword `TSTART`, then `acis_process_events` exits with an error message.
- vi. Columns:  
If HDU  $h_{\text{acaoff}}$  of the `acaofffile` does not include the columns `TIME`, `RA`, `DEC`, and `ROLL` then `acis_process_events` exits with an error message. Hereafter, these columns are referred to as  $\text{TIME}_{\text{acaoff}}$ ,  $\text{RA}_{\text{acaoff}}$ ,  $\text{DEC}_{\text{acaoff}}$ , and  $\text{ROLL}_{\text{acaoff}}$ .
- vii. Sequential:  
If more than one valid `acaofffile` is specified, but the the values `TSTART` are not in increasing order, then `acis_process_events` exits with an error message.

5. `doevtgrade`:

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

$$\text{doevtgrade} \neq \text{yes and} \tag{78}$$

$$\text{doevtgrade} \neq \text{no}, \tag{79}$$

then `acis_process_events` exits with an error message.

6. `apply_cti`:

- (a) Lowercase:  
The parameter string is converted to contain only lower case letters.
- (b) Validation:
  - i. Setting:  
If

$$\text{apply\_cti} \neq \text{yes and} \tag{80}$$

$$\text{apply\_cti} \neq \text{no}, \tag{81}$$

then `acis_process_events` exits with an error message.

- ii. `PHAS`:  
If

$$\text{apply\_cti} = \text{yes} \tag{82}$$

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

- iii. `doevtgrade`:  
If

$$\text{apply\_cti} = \text{yes and} \tag{83}$$

$$\text{doevtgrade} = \text{no}, \tag{84}$$

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

7. `alignmentfile`:

(a) Validation for CC mode:

If

$$\text{OBS\_MODE} = \text{pointing or} \quad (85)$$

$$\text{OBS\_MODE} = \text{POINTING} \quad (86)$$

and

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \quad (87)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED}, \quad (88)$$

then

i. Setting:

If

$$\text{alignmentfile} = \text{none or} \quad (89)$$

$$\text{alignmentfile} = \text{NONE}, \quad (90)$$

then `acis_process_events` exits with an error message.

ii. Existence:

If the `alignmentfile` does not exist, then `acis_process_events` exits with an error message.

iii. Permission:

If the `alignmentfile` exists, but the file permissions do not allow it to be read, then `acis-process_events` exits with an error message.

iv. CONTENT:

If the `alignmentfile` does not have an HDU `halignment` with the keyword

$$\text{CONTENT} = \text{ASPSOL}, \quad (91)$$

then `acis_process_events` exits with an error message.

v. Keyword:

If HDU `halignment` of the `alignmentfile` does not include the keyword `TSTART`, then `acis-process_events` exits with an error message.

vi. Columns:

If HDU `halignment` of the `alignmentfile` does not include the columns `DY`, `DZ`, and `DTHETA` then `acis_process_events` exits with an error message.

vii. Sequential:

If more than one valid `alignmentfile` is specified, but the the values `TSTART` are not in increasing order, then `acis_process_events` exits with an error message.

8. `badpixfile`:

(a) Validation:

i. Existence:

If

$$\text{badpixfile} \neq \text{none and} \quad (92)$$

$$\text{badpixfile} \neq \text{NONE} \quad (93)$$

and the `badpixfile` does not exist, then `badpixfile` is changed to “none” and `acis-process_events` produces a warning message.



ii. Permission:

If

$$\text{badpixfile} \neq \text{none} \text{ and} \tag{94}$$

$$\text{badpixfile} \neq \text{NONE} \tag{95}$$

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.

iii. CONTENT:

If

$$\text{badpixfile} \neq \text{none} \text{ and} \tag{96}$$

$$\text{badpixfile} \neq \text{NONE} \text{ and} \tag{97}$$

and the `badpixfile` does not have one or more HDUs  $h_{\text{badpix}}$  with the keyword

$$\text{CONTENT} = \text{BADPIX} \text{ or} \tag{98}$$

$$\text{CONTENT} = \text{CDB\_ACIS\_BADPIX}, \tag{99}$$

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

iv. Keyword:

If

$$\text{badpixfile} \neq \text{none} \text{ and} \tag{100}$$

$$\text{badpixfile} \neq \text{NONE} \tag{101}$$

and the HDU(s)  $h_{\text{badpix}}$  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 `CCD_IDbadpix`.

v. Columns:

If

$$\text{badpixfile} \neq \text{none} \text{ and} \tag{102}$$

$$\text{badpixfile} \neq \text{NONE} \tag{103}$$

and the HDU(s)  $h_{\text{badpix}}$  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 `CHIPXbadpix`, `CHIPYbadpix`, `TIMEbadpix`, `TIME_STOPbadpix`, and `STATUSbadpix`, respectively.

9. `ctifile`:

(a) Validation:

If

$$\text{ctifile} \neq \text{caldb} \text{ and} \tag{104}$$

$$\text{ctifile} \neq \text{CALDB}, \tag{105}$$

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, but 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 HDUs  $h_{cti}$  with the keyword

$$\text{CONTENT} = \text{CDB\_ACIS\_CTI}, \quad (106)$$

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

iv. Columns:

If the first such HDU 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.

10. `clobber`:

(a) Lowercase:

The parameter string is converted to contain only lower case letters.

(b) Validation:

i. Setting:

If

$$\text{clobber} \neq \text{yes} \text{ and} \quad (107)$$

$$\text{clobber} \neq \text{no}, \quad (108)$$

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

ii. Permission:

If

$$\text{clobber} = \text{yes} \quad (109)$$

and the `outfile` exists, but 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

$$\text{clobber} = \text{no} \quad (110)$$

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

11. `pix_adj`:

(a) Lowercase:

The parameter string is converted to contain only lower case letters.

(b) Validation:

i. Setting:

If

$$\text{pix\_adj} \neq \text{centroid} \text{ and} \quad (111)$$

$$\text{pix\_adj} \neq \text{edser} \text{ and} \quad (112)$$

$$\text{pix\_adj} \neq \text{none} \text{ and} \quad (113)$$

$$\text{pix\_adj} \neq \text{randomize}, \quad (114)$$

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

ii. OBS\_MODE:

If

$$\text{OBS\_MODE} \neq \text{pointing and} \quad (115)$$

$$\text{OBS\_MODE} \neq \text{POINTING} \quad (116)$$

and

$$\text{pix\_adj} \neq \text{none}, \quad (117)$$

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

iii. stop:

If

$$\text{pix\_adj} = \text{centroid or} \quad (118)$$

$$\text{pix\_adj} = \text{edser or} \quad (119)$$

$$\text{pix\_adj} = \text{randomize} \quad (120)$$

and if

$$\text{stop} \neq \text{sky}, \quad (121)$$

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

iv. PHAS:

If

$$\text{pix\_adj} = \text{centroid} \quad (122)$$

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

v. FLTGRADE:

If

$$\text{pix\_adj} = \text{edser} \quad (123)$$

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

12. `subpixfile`:

(a) If

$$\text{pix\_adj} = \text{edser}, \quad (124)$$

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, but 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 HDUs  $h_{\text{subpix}}$  with the keyword

$$\text{CONTENT} = \text{AXAF\_SUBPIX}, \quad (125)$$

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

B. Keyword:

If the HDUs  $h_{\text{subpix}}$  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  $h_{\text{subpix}}$  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 `FLTGRADEsubpix`, `NPOINTSsubpix`, `ENERGYsubpix`, `CHIPX_OFFSETsubpix`, and `CHIPY_OFFSETsubpix`, respectively.

### 1.5.2 Initializations

1. Focal-point CCD:

If

$$\text{OBS\_MODE} = \text{pointing or} \quad (126)$$

$$\text{OBS\_MODE} = \text{POINTING} \quad (127)$$

and

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \quad (128)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED}, \quad (129)$$

then the values of `RA_PNTin` and `DEC_PNTin` are used to determine the `CCD_ID` associated with the focal point. Hereafter this value is referred to as `CCD_IDfocus`.\*

2. Zeroth-order coordinates:

If

$$\text{OBS\_MODE} = \text{pointing or} \quad (130)$$

$$\text{OBS\_MODE} = \text{POINTING} \quad (131)$$

and

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \quad (132)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED} \quad (133)$$

and

$$\text{CONTENT}_{\text{in}} = \text{TGEVT1}, \quad (134)$$

then the `CHIPY_ZOin` coordinates are processed to obtain the minimum, median, and maximum values:

$$\text{CHIPY\_ZO}_{\text{min}} = \text{minimum}(\text{CHIPY\_ZO}_{\text{in}}), \quad (135)$$

$$\text{CHIPY\_ZO}_{\text{med}} = \text{median}(\text{CHIPY\_ZO}_{\text{in}}), \text{ and} \quad (136)$$

$$\text{CHIPY\_ZO}_{\text{max}} = \text{maximum}(\text{CHIPY\_ZO}_{\text{in}}). \quad (137)$$

---

\*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.

Only events in the good-time intervals are included in the computation of the values of  $\text{CHIPY\_Z0}_{\min}$ ,  $\text{CHIPY\_Z0}_{\text{med}}$ , and  $\text{CHIPY\_Z0}_{\max}$ .

### 3. `acaofffile`:

If

$$\text{OBS\_MODE} = \text{pointing or} \quad (138)$$

$$\text{OBS\_MODE} = \text{POINTING} \quad (139)$$

and

$$\text{DATAMODE} = \text{CC33\_FAINT or} \quad (140)$$

$$\text{DATAMODE} = \text{CC33\_GRADED}, \quad (141)$$

then

#### (a) $\text{TIME}_{\min}$ , $\text{TIME}_{\max}$ , $\text{RA}_c$ , and $\text{DEC}_c$ :

The `acaofffile` data are processed to determine the earliest and latest times for which there is aspect information and to determine the right ascension and declination coordinates near the center of the dither pattern:

$$\text{TIME}_{\min} = \min(\text{TIME}_{\text{acaoff}}), \quad (142)$$

$$\text{TIME}_{\max} = \max(\text{TIME}_{\text{acaoff}}), \quad (143)$$

$$\text{RA}_c = \text{median}(\text{RA}_{\text{acaoff}}), \text{ and} \quad (144)$$

$$\text{DEC}_c = \text{median}(\text{DEC}_{\text{acaoff}}). \quad (145)$$

#### (b) $\text{TIME}_c$ :

The `acaofffile` data are processed to determine the time  $\text{TIME}_c$  at which the quantity

$$\cos(\text{DEC}_{\text{acaoff}}) \cos(\text{DEC}_c) \cos(\text{RA}_{\text{acaoff}} - \text{RA}_c) + \sin(\text{DEC}_{\text{acaoff}}) \sin(\text{DEC}_c) \quad (146)$$

is maximized (i.e. the time at which the telescope is pointed the closest to  $(\text{RA}_c, \text{DEC}_c)$ ).

#### (c) $\text{RA\_ADJ}_I$ , $\text{DEC\_ADJ}_I$ , $\text{RA\_ADJ}_S$ , $\text{DEC\_ADJ}_S$ :

The effective values of RA and DEC are computed for the ACIS-I and ACIS-S arrays. These coordinates are used to determine the values of  $\text{TIME}$  and  $\text{CHIPY\_ADJ}$ .

##### i. ACIS-I aim point:

For the ACIS-I array, the values of  $\text{RA\_ADJ}_I$  and  $\text{DEC\_ADJ}_I$  are initialized assuming that the source is at the ACIS-I aim point [i.e. that  $(\text{TIME}, \text{CCD\_ID}, \text{CHIPX}, \text{CHIPY}) = (\text{TIME}_c, 3, 965, 963)^\dagger$ ].

##### ii. ACIS-S aim point:

For the ACIS-S array, the values of  $\text{RA\_ADJ}_S$  and  $\text{DEC\_ADJ}_S$  are initialized assuming that the source is at the ACIS-S aim point [i.e. that  $(\text{TIME}, \text{CCD\_ID}, \text{CHIPX}, \text{CHIPY}) = (\text{TIME}_c, 7, 227, 509)^\ddagger$ ].

##### iii. Target location:

For the CCD at the focal point (i.e.  $\text{CCD\_ID}_{\text{focus}}$ ), the values of  $\text{CHIPY}$  are computed for each row of the `acaofffile`, assuming that the source is at the location specified by  $\text{RA\_TARG}_{\text{in}}$  and  $\text{DEC\_TARG}_{\text{in}}$ . These values of  $\text{CHIPY}$  are referred to as  $\text{CHIPY\_TARG}$ . If

$$\text{median}(\text{CHIPY\_TARG}) \geq 16.5 \text{ and} \quad (147)$$

$$\text{median}(\text{CHIPY\_TARG}) < 1008.5, \quad (148)$$

then

---

<sup>†</sup>As described in the Proposers' Observatory Guide, the location of the aim point on the ACIS-I array has drifted with time. The location used here is within a few dozen pixels of the actual aim point, provided the default `SIM_Y` and `SIM_Z` offsets are used.

<sup>‡</sup>Again, the location used here is within a few dozen pixels of the actual aim point, provided the default `SIM_Y` and `SIM_Z` offsets are used.

A. ACIS-I:

If

$$\text{CCD\_ID}_{\text{focus}} \geq 0 \text{ and} \quad (149)$$

$$\text{CCD\_ID}_{\text{focus}} \leq 3, \quad (150)$$

then

$$\text{RA\_ADJ}_I = \text{RA\_TARG}_{\text{in}} \text{ and} \quad (151)$$

$$\text{DEC\_ADJ}_I = \text{DEC\_TARG}_{\text{in}}. \quad (152)$$

B. ACIS-S:

If

$$\text{CCD\_ID}_{\text{focus}} \geq 4 \text{ and} \quad (153)$$

$$\text{CCD\_ID}_{\text{focus}} \leq 9, \quad (154)$$

then

$$\text{RA\_ADJ}_S = \text{RA\_TARG}_{\text{in}} \text{ and} \quad (155)$$

$$\text{DEC\_ADJ}_S = \text{DEC\_TARG}_{\text{in}}. \quad (156)$$

### 1.5.3 Loop over events

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

1. STATUS:

(a) Exists:

If HDU  $h_{\text{in}}$  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  $k = 1-5, 14, 16-19, \text{ and } 23$  (of 0-31), bits that can be set by `acis_process_events`.
- iii. If

$$\text{doevtgrade} = \text{yes}, \quad (157)$$

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  $h_{\text{in}}$  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:

The value of `EXPNO` for an event is given by `EXPNOin`.

(b) Validation:

If

$$\text{EXPNO} < 0 \text{ or} \quad (158)$$

$$\text{EXPNO} \geq 10^8, \quad (159)$$

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.

### 3. CCD\_ID:

- (a) Read:  
The value of CCD\_ID for an event is given by  $\text{CCD\_ID}_{\text{in}}$ .
- (b) Validation:  
If

$$\text{CCD\_ID} < 0 \text{ or} \tag{160}$$

$$\text{CCD\_ID} > 9, \tag{161}$$

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. Level 0:  
If

$$\text{CONTENT}_{\text{in}} = \text{EVT0}, \tag{162}$$

then the value of CHIPX for an event is given by

$$\text{CHIPX} = \text{CCDX}_{\text{in}} + 1. \tag{163}$$

- ii. Level 1, 1.5, or 2:  
If

$$\text{CONTENT}_{\text{in}} = \text{EVT1}, \text{ or} \tag{164}$$

$$\text{CONTENT}_{\text{in}} = \text{TGEVT1}, \text{ or} \tag{165}$$

$$\text{CONTENT}_{\text{in}} = \text{EVT2}, \tag{166}$$

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

- (b) Validation:
  - i. Unphysical:  
If

$$\text{CHIPX} < 1 \text{ or} \tag{167}$$

$$\text{CHIPX} > 1024, \tag{168}$$

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

- ii. Unexpected:  
If

$$\text{CHIPX} = 1 \text{ or} \tag{169}$$

$$\text{CHIPX} = 1024, \tag{170}$$

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. Although these values are not unphysical, they should not occur.

### 5. NODE\_ID:

(a) Calculate:

The NODE\_ID of an event is given by

$$\text{NODE\_ID} = \text{int} \left( \frac{\text{CHIPX} - 1}{256} \right), \quad (171)$$

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

## 6. CHIPY:

(a) Read:

i. Level 0:

If

$$\text{CONTENT}_{\text{in}} = \text{EVT0}, \quad (172)$$

then

A. TE mode:

If

$$\text{DATAMODE}_{\text{in}} = \text{FAINT or} \quad (173)$$

$$\text{DATAMODE}_{\text{in}} = \text{FAINT\_BIAS or} \quad (174)$$

$$\text{DATAMODE}_{\text{in}} = \text{GRADED or} \quad (175)$$

$$\text{DATAMODE}_{\text{in}} = \text{VFAINT}, \quad (176)$$

then the value of CHIPY for an event is given by

$$\text{CHIPY} = \text{CCDY}_{\text{in}} + 1. \quad (177)$$

B. CC mode:

If

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \quad (178)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED}, \quad (179)$$

then the value of CHIPY for an event is given by

$$\text{CHIPY} = \text{TROW}_{\text{in}} + 1. \quad (180)$$

ii. Level 1, 1.5, or 2:

If

$$\text{CONTENT}_{\text{in}} = \text{EVT1 or} \quad (181)$$

$$\text{CONTENT}_{\text{in}} = \text{TGEVT1 or} \quad (182)$$

$$\text{CONTENT}_{\text{in}} = \text{EVT2}, \quad (183)$$

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

(b) Validation:

i. Unphysical:

A. TE mode:

If

$$\text{DATAMODE}_{\text{in}} = \text{FAINT or} \quad (184)$$

$$\text{DATAMODE}_{\text{in}} = \text{FAINT\_BIAS or} \quad (185)$$

$$\text{DATAMODE}_{\text{in}} = \text{GRADED or} \quad (186)$$

$$\text{DATAMODE}_{\text{in}} = \text{VFAINT} \quad (187)$$



and if

$$\text{CHIPY} < 1 \text{ or} \tag{188}$$

$$\text{CHIPY} > 1024, \tag{189}$$

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

B. CC mode:

If

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT} \text{ or} \tag{190}$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED} \tag{191}$$

and if

$$\text{CHIPY} < 1 \text{ or} \tag{192}$$

$$\text{CHIPY} > 512, \tag{193}$$

then `acis_process_events` exits with an error message because the `CHIPY` value is out of range and `CHIPY`-dependent computations could fail if the value of `CHIPY` is unphysical (especially if it is less than 1).

ii. Unexpected:

A. FAINT, FAINT\_BIAS, or GRADED:

If

$$\text{DATAMODE}_{\text{in}} = \text{FAINT} \text{ or} \tag{194}$$

$$\text{DATAMODE}_{\text{in}} = \text{FAINT\_BIAS} \text{ or} \tag{195}$$

$$\text{DATAMODE}_{\text{in}} = \text{GRADED} \tag{196}$$

and if

$$\text{CHIPY} = 1 \text{ or} \tag{197}$$

$$\text{CHIPY} = 1024, \tag{198}$$

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. Although these values are not unphysical, they should not occur.

B. VFAINT:

If

$$\text{DATAMODE}_{\text{in}} = \text{VFAINT} \tag{199}$$

and if

$$\text{CHIPY} = 1 \text{ or} \tag{200}$$

$$\text{CHIPY} = 2 \text{ or} \tag{201}$$

$$\text{CHIPY} = 1023 \text{ or} \tag{202}$$

$$\text{CHIPY} = 1024, \tag{203}$$

then `acis_process_events` produces a warning upon completion with a count of the total number of events for which one or another of these conditions is true. Although these values are not unphysical, they should not occur.

C. CC33\_FAINT or CC33\_GRADED:

If

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \quad (204)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED} \quad (205)$$

and if

$$\text{CHIPY} = 1 \text{ or} \quad (206)$$

$$\text{CHIPY} = 512, \quad (207)$$

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. Although these values are not unphysical, they should not occur.

7. TG\_M:

(a) CC mode with gratings:

If

$$\text{OBS\_MODE} = \text{pointing or} \quad (208)$$

$$\text{OBS\_MODE} = \text{POINTING} \quad (209)$$

and

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \quad (210)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED} \quad (211)$$

and

$$\text{CONTENT}_{\text{in}} = \text{TGEVT1}, \quad (212)$$

then

i. Read:

The value of `TG_M` for an event is given by `TG_Min`.

ii. Validation:

A. If

$$\text{TG\_M} < -99, \quad (213)$$

then

$$\text{TG\_M} = -99 \quad (214)$$

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

$$\text{TG\_M} > 99, \quad (215)$$

then

$$\text{TG\_M} = 99 \quad (216)$$

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.

8. CHIPY\_TG:

(a) CC mode with gratings:

If

$$\text{OBS\_MODE} = \text{pointing or} \quad (217)$$

$$\text{OBS\_MODE} = \text{POINTING} \quad (218)$$

and

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \quad (219)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED} \quad (220)$$

and

$$\text{CONTENT}_{\text{in}} = \text{TGEVT1}, \quad (221)$$

then

i. Read:

The value of CHIPY\_TG for an event is given by  $\text{CHIPY\_TG}_{\text{in}}$ .

ii. Validation:

A. If

$$\text{TG\_M} > -99 \text{ and} \quad (222)$$

$$\text{TG\_M} < 99, \quad (223)$$

and if

$$\text{CHIPY\_TG} \leq 0 \text{ or} \quad (224)$$

$$\text{CHIPY\_TG} \geq 1025, \quad (225)$$

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

$$\text{TG\_M} > -99, \quad (226)$$

$$\text{TG\_M} < 99, \text{ and} \quad (227)$$

$$\text{CHIPY\_TG} < 1, \quad (228)$$

then

$$\text{CHIPY\_TG} = 1. \quad (229)$$

C. If

$$\text{TG\_M} > -99, \quad (230)$$

$$\text{TG\_M} < 99, \text{ and} \quad (231)$$

$$\text{CHIPY\_TG} > 1024, \quad (232)$$

then

$$\text{CHIPY\_TG} = 1024. \quad (233)$$

9. CHIPY\_Z0:

(a) CC mode with gratings:

If

$$\text{OBS\_MODE} = \text{pointing or} \quad (234)$$

$$\text{OBS\_MODE} = \text{POINTING} \quad (235)$$

and

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \quad (236)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED} \quad (237)$$

and

$$\text{CONTENT}_{\text{in}} = \text{TGEVT1}, \quad (238)$$

then

i. Read:

The value of CHIPY\_Z0 for an event is given by  $\text{CHIPY\_Z0}_{\text{in}}$ .

10. TIME\_RO:

(a) CC mode:

If

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \quad (239)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED}, \quad (240)$$

then

i. Read:

A. Level 0:

If

$$\text{CONTENT}_{\text{in}} = \text{EVT0}, \quad (241)$$

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

B. Level 1, 1.5, or 2:

If

$$\text{CONTENT}_{\text{in}} = \text{EVT1, or} \quad (242)$$

$$\text{CONTENT}_{\text{in}} = \text{TGEVT1, or} \quad (243)$$

$$\text{CONTENT}_{\text{in}} = \text{EVT2}, \quad (244)$$

and

$$\text{TIME\_RO}_{\text{in}} > 0, \quad (245)$$

then

$$\text{TIME\_RO} = \text{TIME\_RO}_{\text{in}}. \quad (246)$$

If

$$\text{CONTENT}_{\text{in}} = \text{EVT1, or} \quad (247)$$

$$\text{CONTENT}_{\text{in}} = \text{TGEVT1, or} \quad (248)$$

$$\text{CONTENT}_{\text{in}} = \text{EVT2}, \quad (249)$$

and

$$\text{TIME\_RO}_{\text{in}} = 0, \quad (250)$$

then

$$\text{TIME\_RO} = \text{TIME}_{\text{in}}. \quad (251)$$

ii. Validation:

If

$$\text{TIME\_RO} < 0 \text{ or} \quad (252)$$

$$\text{TIME\_RO} \geq 3 \times 10^9, \quad (253)$$

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.

11. `TIME` and `CHIPY_ADJ`:

(a) Read or calculate:

i. TE mode:

If

$$\text{DATAMODE}_{\text{in}} = \text{FAINT} \text{ or} \quad (254)$$

$$\text{DATAMODE}_{\text{in}} = \text{FAINT\_BIAS} \text{ or} \quad (255)$$

$$\text{DATAMODE}_{\text{in}} = \text{GRADED} \text{ or} \quad (256)$$

$$\text{DATAMODE}_{\text{in}} = \text{VFAINT}, \quad (257)$$

then

$$\text{TIME} = \text{TIME}_{\text{in}} \text{ and} \quad (258)$$

$$\text{CHIPY\_ADJ} = \text{CHIPY}. \quad (259)$$

ii. Pointing CC mode without grating data:

If

$$\text{OBS\_MODE} = \text{pointing} \text{ or} \quad (260)$$

$$\text{OBS\_MODE} = \text{POINTING} \quad (261)$$

and

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT} \text{ or} \quad (262)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED} \quad (263)$$

and

$$\text{CONTENT}_{\text{in}} = \text{EVT0} \text{ or} \quad (264)$$

$$\text{CONTENT}_{\text{in}} = \text{EVT1} \text{ or} \quad (265)$$

$$\text{CONTENT}_{\text{in}} = \text{EVT2}, \quad (266)$$

then

A.  $\text{TIME}'$ :

The approximate time of arrival

$$\text{TIME}' = \text{TIME\_RO} - (512 + 1028) \times \text{TIMEDEL}_{\text{in}}. \quad (267)$$

B.  $\text{CHIPY\_ADJ}'$ :

If

$$\text{CCD\_ID}_{\text{focus}} \geq 0 \text{ and} \quad (268)$$

$$\text{CCD\_ID}_{\text{focus}} \leq 3, \quad (269)$$

then  $\text{CHIPY\_ADJ}'$  (the approximate value of  $\text{CHIPY\_ADJ}$ ) is given by the  $\text{CHIPY}$  location (on the focal-point CCD) of the coordinates  $\text{RA\_ADJ}_I$  and  $\text{DEC\_ADJ}_I$  at the time  $\text{TIME}'$ . If  $\text{TIME}' < \text{TIME}_{\text{min}}$  or  $\text{TIME}' > \text{TIME}_{\text{max}}$ , then  $\text{TIME}_c$  is used instead of  $\text{TIME}'$ . If

$$\text{CCD\_ID}_{\text{focus}} \geq 4 \text{ and} \quad (270)$$

$$\text{CCD\_ID}_{\text{focus}} \leq 9, \quad (271)$$

then  $\text{CHIPY\_ADJ}'$  is given by the  $\text{CHIPY}$  location (on the focal-point CCD) of the coordinates  $\text{RA\_ADJ}_S$  and  $\text{DEC\_ADJ}_S$  at the time  $\text{TIME}'$ . If  $\text{TIME}' < \text{TIME}_{\text{min}}$  or  $\text{TIME}' > \text{TIME}_{\text{max}}$ , then  $\text{TIME}_c$  is used instead of  $\text{TIME}'$ .

C.  $\text{TIME}$ :

The value of  $\text{CHIPY\_ADJ}'$  is used to obtain a better estimate of the time of arrival

$$\text{TIME} = \text{TIME\_RO} - (\text{CHIPY\_ADJ}' + 1028) \times \text{TIMEDEL}_{\text{in}}. \quad (272)$$

D.  $\text{CHIPY\_ADJ}$ :

If

$$\text{CCD\_ID}_{\text{focus}} \geq 0 \text{ and} \quad (273)$$

$$\text{CCD\_ID}_{\text{focus}} \leq 3, \quad (274)$$

then the value of  $\text{CHIPY\_ADJ}$  is given by the  $\text{CHIPY}$  location (on the focal-point CCD) of the coordinates  $\text{RA\_ADJ}_I$  and  $\text{DEC\_ADJ}_I$  at the time  $\text{TIME}$ . If  $\text{TIME} < \text{TIME}_{\text{min}}$  or  $\text{TIME} > \text{TIME}_{\text{max}}$ , then  $\text{TIME}_c$  is used instead of  $\text{TIME}$ . If

$$\text{CCD\_ID}_{\text{focus}} \geq 4 \text{ and} \quad (275)$$

$$\text{CCD\_ID}_{\text{focus}} \leq 9, \quad (276)$$

then the value of  $\text{CHIPY\_ADJ}$  is given by the  $\text{CHIPY}$  location (on the focal-point CCD) of the coordinates  $\text{RA\_ADJ}_S$  and  $\text{DEC\_ADJ}_S$  at the time  $\text{TIME}$ . If  $\text{TIME} < \text{TIME}_{\text{min}}$  or  $\text{TIME} > \text{TIME}_{\text{max}}$ , then  $\text{TIME}_c$  is used instead of  $\text{TIME}$ .

iii. Pointing CC mode with ACIS-S grating data:

If

$$\text{OBS\_MODE} = \text{pointing} \text{ or} \quad (277)$$

$$\text{OBS\_MODE} = \text{POINTING} \quad (278)$$

and

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT} \text{ or} \quad (279)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED} \quad (280)$$

and

$$\text{CONTENT}_{\text{in}} = \text{TGEVT1} \quad (281)$$

and

$$\text{CCD\_ID}_{\text{focus}} \geq 4 \text{ and} \quad (282)$$

$$\text{CCD\_ID}_{\text{focus}} \leq 9, \quad (283)$$

then

A. Source events in GTIs:

If

$$\text{TG\_M} > -99 \text{ and} \quad (284)$$

$$\text{TG\_M} < 99 \quad (285)$$

and  $\text{TIME\_RO} - (\text{CHIPY\_Z0} + 1028) \times \text{TIMEDEL}_{\text{in}}$  is in a good-time interval, then

$$\text{CHIPY\_ADJ} = \text{CHIPY\_TG} \text{ and} \quad (286)$$

$$\text{TIME} = \text{TIME\_RO} - (\text{CHIPY\_ADJ} + 1028) \times \text{TIMEDEL}_{\text{in}}. \quad (287)$$

B. Source events not in GTIs:

If

$$\text{TG\_M} > -99 \text{ and} \quad (288)$$

$$\text{TG\_M} < 99 \quad (289)$$

and  $\text{TIME\_RO} - (\text{CHIPY\_Z0} + 1028) \times \text{TIMEDEL}_{\text{in}}$  is not in a good-time interval, then

$$\text{CHIPY\_ADJ} = \text{CHIPY\_Z0}_{\text{med}} \text{ and} \quad (290)$$

$$\text{TIME} = \text{TIME\_RO} - (\text{CHIPY\_ADJ} + 1028) \times \text{TIMEDEL}_{\text{in}}. \quad (291)$$

C. Background events with zeroth order on the array in GTIs:

If

$$\text{TG\_M} = -99 \text{ or} \quad (292)$$

$$\text{TG\_M} = 99 \quad (293)$$

and

$$\text{CHIPY\_Z0}_{\text{min}} \geq 0.5 \text{ and} \quad (294)$$

$$\text{CHIPY\_Z0}_{\text{max}} < 1024.5 \quad (295)$$

and  $\text{TIME\_RO} - (\text{CHIPY\_Z0} + 1028) \times \text{TIMEDEL}_{\text{in}}$  is in a good-time interval, then

$$\text{CHIPY\_ADJ} = \text{CHIPY\_Z0} \text{ and} \quad (296)$$

$$\text{TIME} = \text{TIME\_RO} - (\text{CHIPY\_ADJ} + 1028) \times \text{TIMEDEL}_{\text{in}}. \quad (297)$$

D. Background events with zeroth order on the array not in GTIs:

If

$$\text{TG\_M} = -99 \text{ or} \quad (298)$$

$$\text{TG\_M} = 99 \quad (299)$$

and

$$\text{CHIPY\_Z0}_{\min} \geq 0.5 \text{ and} \quad (300)$$

$$\text{CHIPY\_Z0}_{\max} < 1024.5 \quad (301)$$

and  $\text{TIME\_RO} - (\text{CHIPY\_Z0} + 1028) \times \text{TIMEDEL}_{\text{in}}$  is not in a good-time interval, then

$$\text{CHIPY\_ADJ} = \text{CHIPY\_Z0}_{\text{med}} \text{ and} \quad (302)$$

$$\text{TIME} = \text{TIME\_RO} - (\text{CHIPY\_ADJ} + 1028) \times \text{TIMEDEL}_{\text{in}}. \quad (303)$$

E. Background events with zeroth order off the array in GTIs:

If

$$\text{TG\_M} = -99 \text{ or} \quad (304)$$

$$\text{TG\_M} = 99 \quad (305)$$

and

$$\text{CHIPY\_Z0}_{\max} < 0.5 \text{ or} \quad (306)$$

$$\text{CHIPY\_Z0}_{\min} \geq 1024.5 \quad (307)$$

and  $\text{TIME\_RO} - (\text{CHIPY\_Z0} + 1028) \times \text{TIMEDEL}_{\text{in}}$  is in a good-time interval, then

$$\text{CHIPY\_ADJ} = 512 + (\text{CHIPY\_Z0} - \text{CHIPY\_Z0}_{\text{med}}) \text{ and} \quad (308)$$

$$\text{TIME} = \text{TIME\_RO} - (\text{CHIPY\_ADJ} + 1028) \times \text{TIMEDEL}_{\text{in}}. \quad (309)$$

F. Background events with zeroth order off the array not in GTIs:

If

$$\text{TG\_M} = -99 \text{ or} \quad (310)$$

$$\text{TG\_M} = 99 \quad (311)$$

and

$$\text{CHIPY\_Z0}_{\max} < 0.5 \text{ or} \quad (312)$$

$$\text{CHIPY\_Z0}_{\min} \geq 1024.5 \quad (313)$$

and  $\text{TIME\_RO} - (\text{CHIPY\_Z0} + 1028) \times \text{TIMEDEL}_{\text{in}}$  is not in a good-time interval, then

$$\text{CHIPY\_ADJ} = 512 \text{ and} \quad (314)$$

$$\text{TIME} = \text{TIME\_RO} - (\text{CHIPY\_ADJ} + 1028) \times \text{TIMEDEL}_{\text{in}}. \quad (315)$$

iv. Pointing CC mode with ACIS-I grating data:

If

$$\text{OBS\_MODE} = \text{pointing} \text{ or} \quad (316)$$

$$\text{OBS\_MODE} = \text{POINTING} \quad (317)$$

and

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT} \text{ or} \quad (318)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED} \quad (319)$$



and

$$\text{CONTENT}_{\text{in}} = \text{TGEVT1} \quad (320)$$

and

$$\text{CCD\_ID}_{\text{focus}} \geq 0 \text{ and} \quad (321)$$

$$\text{CCD\_ID}_{\text{focus}} \leq 3, \quad (322)$$

then

A.  $\text{TIME}'$ :

The approximate time of arrival

$$\text{TIME}' = \text{TIME\_RO} - (512 + 1028) \times \text{TIMEDEL}_{\text{in}}. \quad (323)$$

B.  $\text{CHIPY\_ADJ}'$ :

$\text{CHIPY\_ADJ}'$  (the approximate value of  $\text{CHIPY\_ADJ}$ ) is given by the  $\text{CHIPY}$  location (on the focal-point CCD) of the coordinates  $\text{RA\_ADJ}_1$  and  $\text{DEC\_ADJ}_1$  at the time  $\text{TIME}'$ . If  $\text{TIME}'$  is not in a good-time interval, then  $\text{TIME}_c$  is used instead of  $\text{TIME}'$ .

C.  $\text{TIME}$ :

The value of  $\text{CHIPY\_ADJ}'$  is used to obtain a better estimate of the time of arrival

$$\text{TIME} = \text{TIME\_RO} - (\text{CHIPY\_ADJ}' + 1028) \times \text{TIMEDEL}_{\text{in}}. \quad (324)$$

D.  $\text{CHIPY\_ADJ}$ :

The value of  $\text{CHIPY\_ADJ}$  is given by the  $\text{CHIPY}$  location (on the focal-point CCD) of the coordinates  $\text{RA\_ADJ}_1$  and  $\text{DEC\_ADJ}_1$  at the time  $\text{TIME}$ . If  $\text{TIME}$  is not in a good-time interval, then  $\text{TIME}_c$  is used instead of  $\text{TIME}$ .

v. Secondary CC mode:

If

$$\text{OBS\_MODE} \neq \text{pointing and} \quad (325)$$

$$\text{OBS\_MODE} \neq \text{POINTING}, \quad (326)$$

then

A.  $\text{TIME}$ :

$$\text{TIME} = \text{TIME\_RO} - (512 + 1028) \times \text{TIMEDEL}_{\text{in}}. \quad (327)$$

B.  $\text{CHIPY\_ADJ}$ :

$$\text{CHIPY\_ADJ} = 512. \quad (328)$$

(b) Validation:

i. If

$$\text{TIME} < 0 \text{ or} \quad (329)$$

$$\text{TIME} \geq 3 \times 10^9, \quad (330)$$

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

$$\text{CHIPY\_ADJ} < 0.5 \text{ or} \quad (331)$$

$$\text{CHIPY\_ADJ} \geq 1024.5, \quad (332)$$

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

12. Bad pixel:

(a) If

$$\text{badpixfile} \neq \text{none and} \quad (333)$$

$$\text{badpixfile} \neq \text{NONE} \quad (334)$$

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

i. If  $\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT}$  or  $\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED}$  and if there are one or more rows  $r$  in HDU  $h_{\text{badpix}}$  where

$$\text{CHIPX} \geq \text{CHIPX}_{\text{badpix},r}[0], \quad (335)$$

$$\text{CHIPX} \leq \text{CHIPX}_{\text{badpix},r}[1], \quad (336)$$

$$\text{TIME} \geq \text{TIME}_{\text{badpix},r}, \quad (337)$$

$$\text{TIME} < \text{TIME\_STOP}_{\text{badpix},r} \quad (338)$$

and where

$$\text{STATUS}_{\text{badpix},r}[5] = 1 \text{ or} \quad (339)$$

$$\text{STATUS}_{\text{badpix},r}[6] = 1 \text{ or} \quad (340)$$

$$\text{STATUS}_{\text{badpix},r}[9] = 1, \quad (341)$$

then

$$\text{STATUS}[0] = 1 \quad (342)$$

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

ii. If  $\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT}$  or  $\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED}$  and if there are one or more rows  $r$  in HDU  $h_{\text{badpix}}$  where

$$\text{CHIPX} \geq \text{CHIPX}_{\text{badpix},r}[0], \quad (343)$$

$$\text{CHIPX} \leq \text{CHIPX}_{\text{badpix},r}[1], \quad (344)$$

$$\text{TIME} \geq \text{TIME}_{\text{badpix},r}, \quad (345)$$

$$\text{TIME} < \text{TIME\_STOP}_{\text{badpix},r} \quad (346)$$

and where

$$\text{STATUS}_{\text{badpix},r}[0] = 1 \text{ or} \quad (347)$$

$$\text{STATUS}_{\text{badpix},r}[1] = 1 \text{ or} \quad (348)$$

$$\text{STATUS}_{\text{badpix},r}[7] = 1 \text{ or} \quad (349)$$

$$\text{STATUS}_{\text{badpix},r}[11] = 1 \text{ or} \quad (350)$$

$$\text{STATUS}_{\text{badpix},r}[12] = 1 \text{ or} \quad (351)$$

$$\text{STATUS}_{\text{badpix},r}[13] = 1 \text{ or} \quad (352)$$

$$\text{STATUS}_{\text{badpix},r}[14] = 1 \text{ or} \quad (353)$$

$$\text{STATUS}_{\text{badpix},r}[16] = 1, \quad (354)$$

then

$$\text{STATUS}[4] = 1 \quad (355)$$

for the event.

- iii. If  $\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT}$  or  $\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED}$  and if there are one or more rows  $r$  in HDU  $h_{\text{badpix}}$  where

$$\text{CHIPX} \geq \text{CHIPX}_{\text{badpix},r}[0], \quad (356)$$

$$\text{CHIPX} \leq \text{CHIPX}_{\text{badpix},r}[1], \quad (357)$$

$$\text{TIME} \geq \text{TIME}_{\text{badpix},r}, \quad (358)$$

$$\text{TIME} < \text{TIME\_STOP}_{\text{badpix},r} \quad (359)$$

and where

$$\text{STATUS}_{\text{badpix},r}[8] = 1 \text{ or} \quad (360)$$

$$\text{STATUS}_{\text{badpix},r}[10] = 1, \quad (361)$$

then

$$\text{STATUS}[5] = 1 \quad (362)$$

for the event.

- iv. If  $\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT}$  or  $\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED}$  and if there are one or more rows  $r$  in HDU  $h_{\text{badpix}}$  where

$$\text{CHIPX} \geq \text{CHIPX}_{\text{badpix},r}[0], \quad (363)$$

$$\text{CHIPX} \leq \text{CHIPX}_{\text{badpix},r}[1], \quad (364)$$

$$\text{TIME} \geq \text{TIME}_{\text{badpix},r}, \quad (365)$$

$$\text{TIME} < \text{TIME\_STOP}_{\text{badpix},r} \quad (366)$$

and where

$$\text{STATUS}_{\text{badpix},r}[3] = 1, \quad (367)$$

then

$$\text{STATUS}[6] = 1 \quad (368)$$

for the event.

- v. If  $\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT}$  or  $\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED}$  and if there are one or more rows  $r$  in HDU  $h_{\text{badpix}}$  where

$$\text{CHIPX} \geq \text{CHIPX}_{\text{badpix},r}[0], \quad (369)$$

$$\text{CHIPX} \leq \text{CHIPX}_{\text{badpix},r}[1], \quad (370)$$

$$\text{TIME} \geq \text{TIME}_{\text{badpix},r}, \quad (371)$$

$$\text{TIME} < \text{TIME\_STOP}_{\text{badpix},r} \quad (372)$$

and where

$$\text{STATUS}_{\text{badpix},r}[2] = 1 \text{ or} \quad (373)$$

$$\text{STATUS}_{\text{badpix},r}[4] = 1, \quad (374)$$

then

$$\text{STATUS}[8] = 1 \quad (375)$$

for the event.

- vi. If  $\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT}$  or  $\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED}$  and if there are one or more rows  $r$  in HDU  $h_{\text{badpix}}$  where

$$\text{CHIPX} \geq \text{CHIPX}_{\text{badpix},r}[0], \quad (376)$$

$$\text{CHIPX} \leq \text{CHIPX}_{\text{badpix},r}[1], \quad (377)$$

$$\text{TIME} \geq \text{TIME}_{\text{badpix},r}, \quad (378)$$

$$\text{TIME} < \text{TIME\_STOP}_{\text{badpix},r} \quad (379)$$

and where

$$\text{STATUS}_{\text{badpix},r}[15] = 1, \quad (380)$$

then

$$\text{STATUS}[16] = 1 \quad (381)$$

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.

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

### 13. PHAS:

- (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  $\text{PHAS}[4] < \text{split threshold}$ , then  $\text{STATUS}[k] = 1$  for bit  $k = 1$ .
- iii. If  $\text{PHAS}[4] \leq \text{PHAS}[j]$ , for one or more  $j = 0-3$  or  $5-8$ , then  $\text{STATUS}[k] = 1$  for bit  $k = 1$ .
- iv. If  $\text{PHAS}[j] > 4095$ , for one or more  $j = 0-8$ , then  $\text{STATUS}[k] = 1$  for bit  $k = 2$ .

14. PHAS\_ADJ:

- (a) If HDU 1 of the `infile` includes  $\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT}$ , if the parameter `apply_cti` = yes, and if the `ctifile` and `mtlfile` are specified, then the CTI-adjusted pulse heights are computed as follows.

- i. The real-valued arrays for the serial CTI adjustment  $\Delta_x$ , the parallel CTI adjustment  $\Delta_y$ , and the adjusted pulse heights PHAS\_ADJ are initialized such that

$$\Delta_x[j] = 0, \quad (382)$$

$$\Delta_y[j] = 0, \text{ and} \quad (383)$$

$$\text{PHAS\_ADJ}[j] = \text{PHAS}[j] \quad (384)$$

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

$$n = 1. \quad (385)$$

- iii. The temporary variables  $\Delta'_x$ ,  $\Delta'_y$ , and PHAS\_ADJ' are set such that

$$\Delta'_x[j] = \Delta_x[j], \quad (386)$$

$$\Delta'_y[j] = \Delta_y[j], \text{ and} \quad (387)$$

$$\text{PHAS\_ADJ}'[j] = \text{PHAS\_ADJ}[j] \quad (388)$$

for each element  $j$ .

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

$$\Delta_x[0] = c_x[0]s_x\rho_x[0]V_x[0], \quad (389)$$

$$\Delta_x[1] = c_x[1]s_x\rho_x[1]V_x[1] - c'_x[0]s_x\rho_x[0]V_x[0], \quad (390)$$

$$\Delta_x[2] = c_x[2]s_x\rho_x[2]V_x[2] - c'_x[1]s_x\rho_x[1]V_x[1], \quad (391)$$

$$\Delta_x[3] = c_x[3]s_x\rho_x[3]V_x[3], \quad (392)$$

$$\Delta_x[4] = c_x[4]s_x\rho_x[4]V_x[4] - c'_x[3]s_x\rho_x[3]V_x[3], \quad (393)$$

$$\Delta_x[5] = c_x[5]s_x\rho_x[5]V_x[5] - c'_x[4]s_x\rho_x[4]V_x[4], \quad (394)$$

$$\Delta_x[6] = c_x[6]s_x\rho_x[6]V_x[6], \quad (395)$$

$$\Delta_x[7] = c_x[7]s_x\rho_x[7]V_x[7] - c'_x[6]s_x\rho_x[6]V_x[6], \text{ and} \quad (396)$$

$$\Delta_x[8] = c_x[8]s_x\rho_x[8]V_x[8] - c'_x[7]s_x\rho_x[7]V_x[7]. \quad (397)$$

where

$$\begin{aligned}
c_x[j] &= \begin{cases} 0 & \left\{ \begin{array}{l} \text{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] < \text{split threshold} \\ \text{(for all } j), \\ \text{FRCTRLX} \left\{ \begin{array}{l} \text{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \geq \text{split threshold and} \\ \text{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] < \\ \text{PHAS}[j-1] + \Delta'_x[j-1] + \Delta'_y[j-1] \\ \text{(for } j = 1, 2, 4, 5, 7, 8), \\ \text{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \geq \text{split threshold} \\ \text{(for } j = 0, 3, 6) \end{array} \right. \\ 1 & \left\{ \begin{array}{l} \text{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \geq \text{split threshold and} \\ \text{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \geq \\ \text{PHAS}[j-1] + \Delta'_x[j-1] + \Delta'_y[j-1] \\ \text{(for } j = 1, 2, 4, 5, 7, 8), \end{array} \right. \end{cases} \\
c'_x[j] &= \begin{cases} 0 & \left\{ \begin{array}{l} \text{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] < \text{split threshold or} \\ \text{PHAS}[j+1] + \Delta'_x[j+1] + \Delta'_y[j+1] < \text{split threshold or} \\ j \rightarrow \text{CHIPX} = 1, 256, 513, \text{ or } 768 \\ \text{(for } j = 0, 1, 3, 4, 6, 7), \\ \text{FRCTRLX} \left\{ \begin{array}{l} \text{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] > \\ \text{PHAS}[j+1] + \Delta'_x[j+1] + \Delta'_y[j+1] \text{ and} \\ \text{PHAS}[j+1] + \Delta'_x[j+1] + \Delta'_y[j+1] \geq \text{split threshold} \\ \text{(for } j = 0, 1, 3, 4, 6, 7), \\ 1 & \left\{ \begin{array}{l} \text{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \leq \\ \text{PHAS}[j+1] + \Delta'_x[j+1] + \Delta'_y[j+1] \text{ and} \\ \text{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] \geq \text{split threshold} \\ \text{(for } j = 0, 1, 3, 4, 6, 7), \end{array} \right. \end{cases} \\
s_x &= 1 + \text{TCTIX} (T - \text{FP\_TEMPO}), \tag{398}
\end{aligned}$$

$\left\{ \begin{array}{l} s_x \text{ is a temperature dependent scaling factor,} \\ \text{TCTIX is the CCD\_ID dependent value in the column TCTIX of the} \\ \text{ctifile,} \\ \text{FP\_TEMPO is the name of a keyword in the ctifile,} \end{array} \right.$

$$T = \left( \frac{t' - t'_k}{t'_{k+1} - t'_k} \right) (\text{FP\_TEMP}_{k+1} - \text{FP\_TEMP}_k) + \text{FP\_TEMP}_k, \tag{399}$$

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

$$t' = t + \text{TIMEDEL}_{\text{in}} (\text{TIMEPIXR}_{\text{evt}} - 0.5), \tag{400}$$

$\left\{ \begin{array}{l} t \text{ is the TIME of the event,} \\ \text{TIMEPIXR}_{\text{evt}} \text{ is a keyword in the infile,} \end{array} \right.$

$$t'_k = \text{TIME}_k + \text{TIMEDEL}_{\text{mtl}} (\text{TIMEPIXR}_{\text{mtl}} - 0.5), \tag{401}$$

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

$$t'_{k+1} = \text{TIME}_{k+1} + \text{TIMEDEL}_{\text{mtl}} (\text{TIMEPIXR}_{\text{mtl}} - 0.5), \tag{402}$$

$$\rho_x[j] = \begin{cases} \text{TIME}_{k+1} \text{ is the } (k+1)^{\text{th}} \text{ element of the column TIME in the mtlfile,} \\ t'_{k+1} > t', \\ \text{If } t' > t'_k \text{ for } k = n, \text{ where } n \text{ is the last element, then } k = n, \\ \text{FP\_TEMP}_{k+1} \text{ is the } (k+1)^{\text{th}} \text{ element of the column FP\_TEMP in the} \\ \text{mtlfile,} \end{cases} \quad (403)$$

$$V_x[j] = \begin{cases} \rho_x[j] \text{ depends upon the CCD\_ID and upon the CHIPX and nint(CHIPY\_ADJ)} \\ \text{coordinates associated with element } j \text{ of PHAS\_ADJ}[j] \text{ (see Fig. 1),} \\ \left( \frac{\text{PHAS}[j] + \Delta'_x[j] + \Delta'_y[j] - \text{PHA}_l}{\text{PHA}_{l+1} - \text{PHA}_l} \right) (\text{VOLUME\_X}_{l+1} - \text{VOLUME\_X}_l) + \\ \text{VOLUME\_X}_l, \end{cases} \quad (404)$$

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

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

$$\Delta_x[0] = c_x[0]s_x\rho_x[0]V_x[0] - c'_x[1]s_x\rho_x[1]V_x[1], \quad (405)$$

$$\Delta_x[1] = c_x[1]s_x\rho_x[1]V_x[1] - c'_x[2]s_x\rho_x[2]V_x[2], \quad (406)$$

$$\Delta_x[2] = c_x[2]s_x\rho_x[2]V_x[2], \quad (407)$$

$$\Delta_x[3] = c_x[3]s_x\rho_x[3]V_x[3] - c'_x[4]s_x\rho_x[4]V_x[4], \quad (408)$$

$$\Delta_x[4] = c_x[4]s_x\rho_x[4]V_x[4] - c'_x[5]s_x\rho_x[5]V_x[5], \quad (409)$$

$$\Delta_x[5] = c_x[5]s_x\rho_x[5]V_x[5], \quad (410)$$

$$\Delta_x[6] = c_x[6]s_x\rho_x[6]V_x[6] - c'_x[7]s_x\rho_x[7]V_x[7], \quad (411)$$

$$\Delta_x[7] = c_x[7]s_x\rho_x[7]V_x[7] - c'_x[8]s_x\rho_x[8]V_x[8], \text{ and} \quad (412)$$

$$\Delta_x[8] = c_x[8]s_x\rho_x[8]V_x[8]. \quad (413)$$

where

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

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

and  $s_x$ ,  $T$ ,  $t'$ ,  $t'_k$ ,  $t'_{k+1}$ ,  $\rho_x[j]$ , and  $V_x[j]$  are given by equations. 398, 399, 400, 401, 402, 403, and 404, respectively.

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

$$\Delta_y[0] = c_y[0]s_y\rho_y[0]V_y[0], \quad (414)$$

$$\Delta_y[1] = c_y[1]s_y\rho_y[1]V_y[1], \quad (415)$$

$$\Delta_y[2] = c_y[2]s_y\rho_y[2]V_y[2], \quad (416)$$

$$\Delta_y[3] = c_y[3]s_y\rho_y[3]V_y[3] - c'_y[0]s_y\rho_y[0]V_y[0], \quad (417)$$

$$\Delta_y[4] = c_y[4]s_y\rho_y[4]V_y[4] - c'_y[1]s_y\rho_y[1]V_y[1], \quad (418)$$

$$\Delta_y[5] = c_y[5]s_y\rho_y[5]V_y[5] - c'_y[2]s_y\rho_y[2]V_y[2], \quad (419)$$

$$\Delta_y[6] = c_y[6]s_y\rho_y[6]V_y[6] - c'_y[3]s_y\rho_y[3]V_y[3], \quad (420)$$

$$\Delta_y[7] = c_y[7]s_y\rho_y[7]V_y[7] - c'_y[4]s_y\rho_y[4]V_y[4], \quad (421)$$

$$\Delta_y[8] = c_y[8]s_y\rho_y[8]V_y[8] - c'_y[5]s_y\rho_y[5]V_y[5], \quad (422)$$





and  $T$ ,  $t'$ ,  $t'_k$ , and  $t'_{k+1}$ , are given by equations. 399, 400, 401, and 402, respectively.  
vi. The CTI-adjusted pulse heights

$$\text{PHAS\_ADJ}[j] = \text{PHAS}[j] + \Delta_x[j] + \Delta_y[j] \quad (426)$$

for all  $j$ .

vii. A. If

$$|\text{PHAS\_ADJ}'[j] - \text{PHAS\_ADJ}[j]| < \text{cticonverge} \text{ (for all } j \text{) and} \quad (427)$$

$$n \leq \text{max\_cti\_iter}, \quad (428)$$

then the computation of PHAS\_ADJ is complete for the event.

B. If

$$|\text{PHAS\_ADJ}'[j] - \text{PHAS\_ADJ}[j]| \geq \text{cticonverge} \text{ (for one or more } j \text{) and} \quad (429)$$

$$n < \text{max\_cti\_iter}, \quad (430)$$

then  $n = n + 1$  and steps 1.5.14(a)iii–1.5.14(a)vii are repeated.

C. If

$$|\text{PHAS\_ADJ}'[j] - \text{PHAS\_ADJ}[j]| \geq \text{cticonverge} \text{ (for one or more } j \text{) and} \quad (431)$$

$$n \geq \text{max\_cti\_iter}, \quad (432)$$

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  $k = 20$  to indicate that the CTI adjustment did not converge.

#### 15. FLTGRADE:

(a) If

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT} \text{ and} \quad (433)$$

$$\text{apply\_cti} = \text{yes}, \quad (434)$$

then

$$\text{FLTGRADE} = c_f[0] + 2c_f[1] + 4c_f[2] + 8c_f[3] + 16c_f[5] + 32c_f[6] + 64c_f[7] + 128c_f[8], \quad (435)$$

where

$$c_f[j] = \begin{cases} 0 & \text{if PHAS\_ADJ}[j] < \text{split threshold} \\ 1 & \text{otherwise,} \end{cases} \quad (436)$$

and the elements  $j = 0-3$  and  $5-8$  of PHAS\_ADJ are depicted in Figure 1.

(b) If

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT} \text{ and} \quad (437)$$

$$\text{apply\_cti} = \text{no}, \quad (438)$$

then

$$\text{FLTGRADE} = c_f[0] + 2c_f[1] + 4c_f[2] + 8c_f[3] + 16c_f[5] + 32c_f[6] + 64c_f[7] + 128c_f[8], \quad (439)$$

where

$$c_f[j] = \begin{cases} 0 & \text{if PHAS}[j] < \text{split threshold} \\ 0 & \text{if PHAS}[j] > 4095 \\ 0 & \text{if PHAS}[j] > \text{PHAS}[4] \text{ for } j = 0-3 \\ 0 & \text{if PHAS}[j] \geq \text{PHAS}[4] \text{ for } j = 5-8 \\ 1 & \text{otherwise.} \end{cases} \quad (440)$$

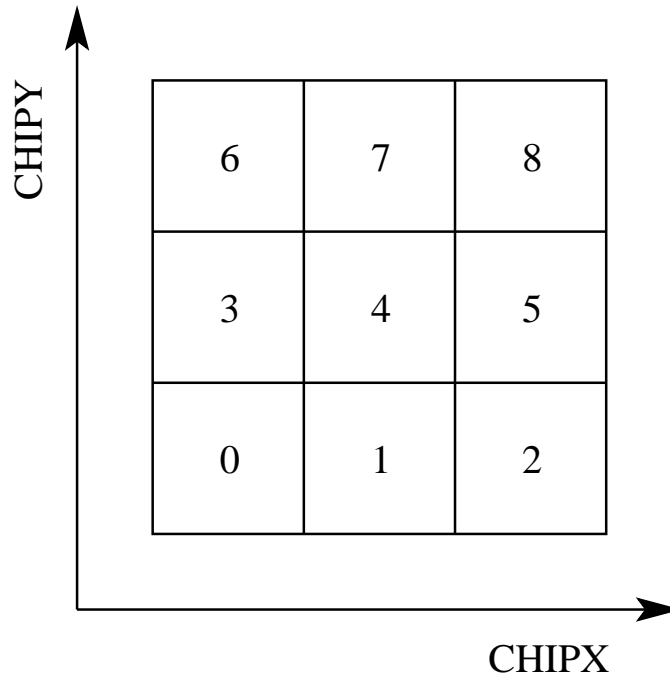


Figure 1: The relative CHIPX and CHIPY coordinates of the nine elements  $j = 0-8$  of a 3 pixel  $\times$  3 pixel event island PHAS[ $j$ ] or PHAS\_ADJ[ $j$ ].

(c) If

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED}, \quad (441)$$

then the FLTGRADE of an event is equal to the value of FLTGRADE for the event in the `infile`.

16. GRADE:

(a) If the `grade` file is specified, then the GRADE of an event is determined from the FLTGRADE of the event as follows.

- i. The appropriate HDU of the `grade` file is identified. This HDU is the one where the header keyword CBD10001 includes the `DATAMODEin` of HDU 1 of the `infile`.
- ii. The row  $i$  of the appropriate HDU of the `grade` file is identified. This row is the one where

$$\text{FLTGRADE}_{\text{grade},i} = \text{FLTGRADE}, \quad (442)$$

where `FLTGRADEgrade` is a column in the `grade` file.

iii. The GRADE of the event is given by

$$\text{GRADE} = \text{GRADE}_{\text{grade},i}, \quad (443)$$

where `GRADEgrade` is a column in the `grade` file.

17. PHA\_RO:

(a) If

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT}, \quad (444)$$

then

$$\text{PHA\_RO} = \sum_{j=0}^8 \beta[j]p[j], \quad (445)$$

where

i.

$$p[j] = \text{PHAS}[j]. \quad (446)$$

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

iii.

$$\beta[j] = 0 \quad \text{if} \quad p[j] < \text{split threshold}. \quad (447)$$

iv.

$$\beta[j] = 0 \quad \text{if} \quad \begin{cases} p[j] > p[4] & (\text{for } j = 0-3) \\ p[j] \geq p[4] & (\text{for } j = 5-8) \end{cases} \quad (448)$$

v. If CORNERS = -1, then

$$\beta[0] = \beta[2] = \beta[6] = \beta[8] = 0. \quad (449)$$

vi. If CORNERS = 0, then there are no additional constraints on  $\beta[0]$ ,  $\beta[2]$ ,  $\beta[6]$ , and  $\beta[8]$ .

vii. If CORNERS = 1, then

$$\beta[0] = 0 \quad \text{if} \quad \beta[1] = 0 \quad \text{and} \quad \beta[3] = 0. \quad (450)$$

$$\beta[2] = 0 \quad \text{if} \quad \beta[1] = 0 \quad \text{and} \quad \beta[5] = 0. \quad (451)$$

$$\beta[6] = 0 \quad \text{if} \quad \beta[3] = 0 \quad \text{and} \quad \beta[7] = 0. \quad (452)$$

$$\beta[8] = 0 \quad \text{if} \quad \beta[5] = 0 \quad \text{and} \quad \beta[7] = 0. \quad (453)$$

viii. If CORNERS = 2, then

$$\beta[0] = 0 \quad \text{if} \quad \beta[1] = 0 \quad \text{or} \quad \beta[3] = 0 \quad \text{or} \quad \text{GRADE} \neq 6. \quad (454)$$

$$\beta[2] = 0 \quad \text{if} \quad \beta[1] = 0 \quad \text{or} \quad \beta[5] = 0 \quad \text{or} \quad \text{GRADE} \neq 6. \quad (455)$$

$$\beta[6] = 0 \quad \text{if} \quad \beta[3] = 0 \quad \text{or} \quad \beta[7] = 0 \quad \text{or} \quad \text{GRADE} \neq 6. \quad (456)$$

$$\beta[8] = 0 \quad \text{if} \quad \beta[5] = 0 \quad \text{or} \quad \beta[7] = 0 \quad \text{or} \quad \text{GRADE} \neq 6. \quad (457)$$

(b) If

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED}, \quad (458)$$

then

i. If

$$\text{CONTENT}_{\text{in}} = \text{EVT0}, \quad (459)$$

then the value of PHA\_RO for the event is the value of PHA in the `infile`.

ii. If

$$\text{CONTENT}_{\text{in}} = \text{EVT1}, \text{ TGEVT1}, \text{ or EVT2}, \quad (460)$$

then the value of PHA\_RO for the event is the value of PHA\_RO in the `infile`.

18. PHA, including time-dependent gain:

(a) If

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT}, \quad (461)$$

then

$$\text{PHA} = \sum_{j=0}^8 \beta[j]p[j], \quad (462)$$

where

i.

$$p[j] = \begin{cases} \text{PHAS\_ADJ}[j] & \text{if } \text{apply\_cti} = \text{yes} \\ \text{PHAS}[j] & \text{if } \text{apply\_cti} = \text{no} \end{cases} \quad (463)$$

ii. The elements  $j = 0-8$  of PHAS\_ADJ (or PHAS) are depicted in Figure 1.

iii.

$$\beta[j] = 0 \quad \text{if } p[j] < \text{split threshold}. \quad (464)$$

iv. If the CTI adjustment is not performed, then

$$\beta[j] = 0 \quad \text{if } \begin{cases} p[j] > p[4] & (\text{for } j = 0-3) \\ p[j] \geq p[4] & (\text{for } j = 5-8) \end{cases} \quad (465)$$

v. If CORNERS = -1, then

$$\beta[0] = \beta[2] = \beta[6] = \beta[8] = 0. \quad (466)$$

vi. If CORNERS = 0, then there are no additional constraints on  $\beta[0]$ ,  $\beta[2]$ ,  $\beta[6]$ , and  $\beta[8]$ .

vii. If CORNERS = 1, then

$$\beta[0] = 0 \quad \text{if } \beta[1] = 0 \text{ and } \beta[3] = 0. \quad (467)$$

$$\beta[2] = 0 \quad \text{if } \beta[1] = 0 \text{ and } \beta[5] = 0. \quad (468)$$

$$\beta[6] = 0 \quad \text{if } \beta[3] = 0 \text{ and } \beta[7] = 0. \quad (469)$$

$$\beta[8] = 0 \quad \text{if } \beta[5] = 0 \text{ and } \beta[7] = 0. \quad (470)$$

viii. If CORNERS = 2, then

$$\beta[0] = 0 \quad \text{if } \beta[1] = 0 \text{ or } \beta[3] = 0 \text{ or } \text{GRADE} \neq 6. \quad (471)$$

$$\beta[2] = 0 \quad \text{if } \beta[1] = 0 \text{ or } \beta[5] = 0 \text{ or } \text{GRADE} \neq 6. \quad (472)$$

$$\beta[6] = 0 \quad \text{if } \beta[3] = 0 \text{ or } \beta[7] = 0 \text{ or } \text{GRADE} \neq 6. \quad (473)$$

$$\beta[8] = 0 \quad \text{if } \beta[5] = 0 \text{ or } \beta[7] = 0 \text{ or } \text{GRADE} \neq 6. \quad (474)$$

(b) If

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED}, \quad (475)$$

then the value of PHA for the event is read from the `infile`.

(c) If

$$\text{apply\_tgain} = \text{yes}, \quad (476)$$

then

$$\text{PHA} = \text{PHA} - \text{int} \left[ \left( \frac{\text{TIME} - \text{EPOCH1}}{\text{EPOCH2} - \text{EPOCH1}} \right) (\delta_2 - \delta_1) + \delta_1 - \epsilon \right], \quad (477)$$

where

$$\text{int} = \text{the integer portion of (i.e. truncate or round down)}, \quad (478)$$

$$\text{TIME} = \text{the time of the event}, \quad (479)$$

$$\text{EPOCH1} = \text{a keyword in the } \text{tgainfile}, \quad (480)$$

$$\text{EPOCH2} = \text{a keyword in the } \text{tgainfile}, \quad (481)$$

$$\delta_1 = \left( \frac{\text{PHA} - \text{PHA}_m[r]}{\text{PHA}_{m+1}[r] - \text{PHA}_m[r]} \right) (\text{DELTPHA1}_{m+1}[r] - \text{DELTPHA1}_m[r]) + \quad (482)$$

$$\text{DELTPHA1}_m[r], \quad (483)$$

$$\left\{ \begin{array}{l}
r \text{ is the row of the } \mathbf{tgainfile} \text{ where} \\
\left\{ \begin{array}{l}
\text{CCD\_ID}[r] = \text{CCD\_ID}, \\
\text{CHIPX\_LO}[r] \leq \text{CHIPX}, \\
\text{CHIPX\_HI}[r] \geq \text{CHIPX}, \\
\text{CHIPY\_LO}[r] \leq \text{nint}(\text{CHIPY\_ADJ}), \text{ and} \\
\text{CHIPY\_HI}[r] \geq \text{nint}(\text{CHIPY\_ADJ}).
\end{array} \right. \\
m \text{ is the element of row } r \text{ where} \\
\left\{ \begin{array}{l}
\text{PHA}_m[r] \leq \text{PHA} \text{ and} \\
\text{PHA}_{m+1}[r] > \text{PHA}. \\
\text{If } \text{PHA} < \text{PHA}_m[r] \text{ for } m = 0, \text{ then } m = 0. \\
\text{If } \text{PHA} \geq \text{PHA}_m[r] \text{ for } m = M \text{ and } M \text{ is the last element of } \text{PHA}[r], \\
\text{then } m = M - 1.
\end{array} \right. \\
\text{The } \mathbf{tgainfile} \text{ includes a binary table with columns named} \\
\text{CCD\_ID, CHIPX\_LO, CHIPX\_HI, CHIPY\_LO, CHIPY\_HI, PHA, DELTPHA1, and} \\
\text{DELTPHA2.}
\end{array} \right. \quad (484)$$

$$\delta_2 = \left( \frac{\text{PHA} - \text{PHA}_m[r]}{\text{PHA}_{m+1}[r] - \text{PHA}_m[r]} \right) (\text{DELTPHA2}_{m+1}[r] - \text{DELTPHA2}_m[r]) + \quad (485)$$

$$\text{DELTPHA2}_m[r], \quad (486)$$

$$\epsilon = \text{is a uniform random deviate in the range } [0, 1), \quad (487)$$

$$\left\{ \begin{array}{l}
\text{If } \mathbf{rand\_pha} = \text{no}, \text{ then } \epsilon = 0.
\end{array} \right. \quad (488)$$

(d) If

$$\text{PHA} \geq 32767, \quad (489)$$

then  $\text{STATUS}[k] = 1$  for bit  $k = 3$ .

#### 19. CORN\_PHA:

(a) If

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED}, \quad (490)$$

then the value of CORN\_PHA is read from the **infile**.

#### 20. ENERGY:

(a) If the parameter **calculate\_pi** = yes, if the parameter **gainfile** is specified, and if  $\text{PHA} > 0$ , then

i. The row  $i$  in the **gainfile** is identified such that

$$\text{CCD\_ID} = \text{CCD\_ID}_{\text{gain},i}, \quad (491)$$

$$\text{CHIPX\_MIN}_{\text{gain},i} \leq \text{CHIPX} \leq \text{CHIPX\_MAX}_{\text{gain},i}, \quad \text{and} \quad (492)$$

$$\text{CHIPY\_MIN}_{\text{gain},i} \leq \text{nint}(\text{CHIPY\_ADJ}) \leq \text{CHIPY\_MAX}_{\text{gain},i}, \quad (493)$$

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

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

iii. The element  $j$  of row  $i$  of  $\text{PHA}_{\text{gain}}$  is identified such that

$$\text{PHA}_{\text{gain},i}[j] \leq (\text{PHA} + \Delta p) < \text{PHA}_{\text{gain},i}[j + 1], \quad (494)$$

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

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

$$\text{ENERGY} = \left( \frac{\text{PHA} + \Delta p - \text{PHA}_{\text{gain},i}[j]}{\text{PHA}_{\text{gain},i}[j+1] - \text{PHA}_{\text{gain},i}[j]} \right) (\text{ENERGY}_{\text{gain},i}[j+1] - \text{ENERGY}_{\text{gain},i}[j]) + \text{ENERGY}_{\text{gain},i}[j], \quad (495)$$

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

v. If  $\text{ENERGY} < 0$ , then  $\text{ENERGY} = 0$ .

(b) If the parameter `calculate_pi` = yes, if the parameter `gainfile` is specified, and if  $\text{PHA} \leq 0$ , then  $\text{ENERGY} = 0$ .

(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  $\text{ENERGY} = 0$ .

21. PI:

(a) If

$$\text{calculate\_pi} = \text{yes}, \quad (496)$$

then

i.

$$\text{PI} = \text{int} \left( \frac{\text{ENERGY}}{\text{pi\_bin\_width}} \right) + 1, \quad (497)$$

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

ii. If

$$\text{PI} < 1, \quad (498)$$

then  $\text{PI} = 1$ .

iii. If

$$\text{PI} > \text{pi\_num\_bins}, \quad (499)$$

then  $\text{PI} = \text{pi\_num\_bins}$ .

(b) If

$$\text{calculate\_pi} = \text{no} \quad (500)$$

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

22. `pix_adj`:

(a) centroid:

If

$$\text{pix\_adj} = \text{centroid}, \quad (501)$$

then

$$\text{CHIPX\_ADJ} = \text{CHIPX\_ADJ} - w'[0] + w'[2] - w'[3] + w'[5] - w'[6] + w'[8] \quad (502)$$

and if

$$\text{DATAMODE} = \text{CC33\_FAINT} \text{ or} \quad (503)$$

$$\text{DATAMODE} = \text{CC33\_GRADED} \text{ or} \quad (504)$$

$$\text{DATAMODE} = \text{FAINT} \text{ or} \quad (505)$$

$$\text{DATAMODE} = \text{FAINT\_BIAS} \text{ or} \quad (506)$$

$$\text{DATAMODE} = \text{GRADED} \text{ or} \quad (507)$$

$$\text{DATAMODE} = \text{VFAINT}, \quad (508)$$

then

$$\text{CHIPY\_ADJ} = \text{CHIPY\_ADJ} - w'[0] - w'[1] - w'[2] + w'[6] + w'[7] + w'[8] \quad (509)$$

and if

$$\text{DATAMODE} = \text{CC33\_FAINT} \text{ or} \quad (510)$$

$$\text{DATAMODE} = \text{CC33\_GRADED}, \quad (511)$$

then

$$\text{TIME} = \text{TIME} + (w'[0] + w'[1] + w'[2] - w'[6] - w'[7] - w'[8]) \times \text{TIMEDEL}_{\text{in}}, \quad (512)$$

where

$$w'[j] = \frac{w[j]}{\sum_{j=0}^8 w[j]}, \quad (513)$$

$$w[j] = \begin{cases} p[j] & \text{if the pixel is valid} \\ 0 & \text{if the pixel is invalid,} \end{cases} \quad (514)$$

$$p[j] = \begin{cases} \text{PHAS\_ADJ}[j] & \text{if apply\_cti} = \text{yes} \\ \text{PHAS}[j] & \text{if apply\_cti} = \text{no,} \end{cases} \quad (515)$$

and the pixel is invalid if

$$\beta[j] = 0 \text{ or} \quad (516)$$

$$\text{STATUS}[0] = 1 \text{ or} \quad (517)$$

$$\text{STATUS}[1] = 1 \text{ or} \quad (518)$$

$$\text{STATUS}[2] = 1 \text{ or} \quad (519)$$

$$\text{STATUS}[3] = 1 \text{ or} \quad (520)$$

$$\text{STATUS}[4] = 1 \text{ or} \quad (521)$$

$$\text{STATUS}[11] = 1 \text{ or} \quad (522)$$

$$\text{STATUS}[13] = 1 \text{ or} \quad (523)$$

$$\text{STATUS}[14] = 1 \text{ or} \quad (524)$$

$$\text{STATUS}[15] = 1 \text{ or} \quad (525)$$

$$\text{STATUS}[16] = 1. \quad (526)$$

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



(b) edser:

If

$$\text{pix\_adj} = \text{edser}, \quad (527)$$

then

$$\text{CHIPX\_ADJ} = \text{CHIPX\_ADJ} + \left( \frac{\text{ENERGY} - E[k]}{E[k+1] - E[k]} \right) (\Delta X[k+1] - \Delta X[k]) + \Delta X[k] \quad (528)$$

and if

$$\text{DATAMODE} = \text{CC33\_FAINT} \text{ or} \quad (529)$$

$$\text{DATAMODE} = \text{CC33\_GRADED} \text{ or} \quad (530)$$

$$\text{DATAMODE} = \text{FAINT} \text{ or} \quad (531)$$

$$\text{DATAMODE} = \text{FAINT\_BIAS} \text{ or} \quad (532)$$

$$\text{DATAMODE} = \text{GRADED} \text{ or} \quad (533)$$

$$\text{DATAMODE} = \text{VF AINT}, \quad (534)$$

then

$$\text{CHIPY\_ADJ} = \text{CHIPY\_ADJ} + \left( \frac{\text{ENERGY} - E[k]}{E[k+1] - E[k]} \right) (\Delta Y[k+1] - \Delta Y[k]) + \Delta Y[k] \quad (535)$$

and if

$$\text{DATAMODE} = \text{CC33\_FAINT} \text{ or} \quad (536)$$

$$\text{DATAMODE} = \text{CC33\_GRADED}, \quad (537)$$

then

$$\text{TIME} = \text{TIME} - \left( \left( \frac{\text{ENERGY} - E[k]}{E[k+1] - E[k]} \right) (\Delta Y[k+1] - \Delta Y[k]) + \Delta Y[k] \right) \times \text{TIMEDEL}_{\text{in}}, \quad (538)$$

where  $E[k]$  and  $E[k+1]$ ,  $X[k]$  and  $X[k+1]$ , and  $Y[k]$  and  $Y[k+1]$  are the  $k$  and  $(k+1)^{\text{th}}$  elements of the vector columns  $\text{ENERGY}_{\text{subpix}}$ ,  $\text{CHIPX\_OFFSET}_{\text{subpix}}$ , and  $\text{CHIPY\_OFFSET}_{\text{subpix}}$ , 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  $\text{FLTGRADE}_{\text{subpix}} = \text{FLTGRADE}$ . The values of  $k$  are the ones where

$$\text{ENERGY} \geq E[k] \text{ and} \quad (539)$$

$$\text{ENERGY} < E[k+1]. \quad (540)$$

Note that if

$$\text{ENERGY} \leq E[0], \quad (541)$$

then  $k = 0$ . Similarly, if

$$\text{ENERGY} \geq E[\text{NPOINTS}_{\text{subpix}} - 2], \quad (542)$$

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

(c) none:

If

$$\text{pix\_adj} = \text{none}, \quad (543)$$

then the values of `CHIPX_ADJ` and `CHIPY_ADJ` remain unchanged.

(d) randomize:

If

$$\text{pix\_adj} = \text{randomize}, \quad (544)$$

then

$$\text{CHIPX\_ADJ} = \text{CHIPX\_ADJ} + \epsilon_x \quad (545)$$

and if

$$\text{DATAMODE} = \text{CC33\_FAINT} \text{ or} \quad (546)$$

$$\text{DATAMODE} = \text{CC33\_GRADED} \text{ or} \quad (547)$$

$$\text{DATAMODE} = \text{FAINT} \text{ or} \quad (548)$$

$$\text{DATAMODE} = \text{FAINT\_BIAS} \text{ or} \quad (549)$$

$$\text{DATAMODE} = \text{GRADED} \text{ or} \quad (550)$$

$$\text{DATAMODE} = \text{VFAINT}, \quad (551)$$

then

$$\text{CHIPY\_ADJ} = \text{CHIPY\_ADJ} + \epsilon_y \quad (552)$$

and if

$$\text{DATAMODE} = \text{CC33\_FAINT} \text{ or} \quad (553)$$

$$\text{DATAMODE} = \text{CC33\_GRADED}, \quad (554)$$

then

$$\text{TIME} = \text{TIME} - \epsilon_y \times \text{TIMEDEL}_{\text{in}}, \quad (555)$$

where  $\epsilon_x$  and  $\epsilon_y$  are uniform random deviates in the range  $[-0.5, +0.5)$  pixel.

(e) If

$$\text{CHIPX\_ADJ} < 0.5, \quad (556)$$

then

$$\text{CHIPX\_ADJ} = 1. \quad (557)$$

(f) If

$$\text{CHIPX\_ADJ} \geq 1024.5, \quad (558)$$

then

$$\text{CHIPX\_ADJ} = 1024. \quad (559)$$

(g) If

$$\text{CHIPY\_ADJ} < 0.5, \quad (560)$$

then

$$\text{CHIPY\_ADJ} = 1. \quad (561)$$

(h) If

$$\text{CHIPY\_ADJ} \geq 1024.5, \quad (562)$$

then

$$\text{CHIPY\_ADJ} = 1024. \quad (563)$$

23. TDETX and TDETY:

(a) If

$$\text{stop} = \text{tdet or} \quad (564)$$

$$\text{stop} = \text{det or} \quad (565)$$

$$\text{stop} = \text{tan or} \quad (566)$$

$$\text{stop} = \text{sky}, \quad (567)$$

then

i. If

$$\text{CONTENT} = \text{EVT0 or} \quad (568)$$

$$\text{CONTENT} = \text{EVT1 or} \quad (569)$$

$$\text{CONTENT} = \text{EVT2}, \quad (570)$$

then 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 before the computation of the TDET coordinate.

ii. If

$$\text{CONTENT} = \text{TGEVT1}, \quad (571)$$

then the values of TDETX and TDETY are computed using the values of `nint(CHIPX_ADJ)` and `nint(CHIPY_ZO)`.<sup>§</sup>

24. DETX and DETY:

(a) If

$$\text{stop} = \text{det or} \quad (572)$$

$$\text{stop} = \text{tan or} \quad (573)$$

$$\text{stop} = \text{sky}, \quad (574)$$

then

i. If

$$\text{CONTENT} = \text{EVT0 or} \quad (575)$$

$$\text{CONTENT} = \text{EVT1 or} \quad (576)$$

$$\text{CONTENT} = \text{EVT2}, \quad (577)$$

then the values of DETX and DETY are computed using the real-valued coordinates `CHIPX_ADJ` and `CHIPY_ADJ`.

---

<sup>§</sup>As requested, the TDETX, TDETY, DETX, DETY, X, and Y coordinates for gratings observations are computed using the `CHIPY` location of zeroth order, not the `CHIPY` location of the gratings arms.

ii. If

$$\text{CONTENT} = \text{TGEVT1}, \quad (578)$$

then the values of DETX and DETY are computed using the real-valued coordinates CHIPX\_ADJ and CHIPY\_Z0.<sup>§</sup>

25. X and Y:

(a) If

$$\text{stop} = \text{sky}, \quad (579)$$

then

i. If

$$\text{CONTENT} = \text{EVT0 or} \quad (580)$$

$$\text{CONTENT} = \text{EVT1 or} \quad (581)$$

$$\text{CONTENT} = \text{EVT2}, \quad (582)$$

the values of X and Y are computed using the real-valued coordinates CHIPX\_ADJ and CHIPY\_ADJ.<sup>§</sup>

ii. If

$$\text{CONTENT} = \text{TGEVT1}, \quad (583)$$

the values of X and Y are computed using the real-valued coordinates CHIPX\_ADJ and CHIPY\_Z0.

26. SKY\_1D:

(a) If

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \quad (584)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED} \quad (585)$$

and if

$$\text{stop} = \text{sky}, \quad (586)$$

then the value of SKY\_1D is computed.

#### 1.5.4 Write outfile

1. PIX\_ADJ:

(a) If

$$\text{pix\_adj} = \text{centroid}, \quad (587)$$

then

$$\text{PIX\_ADJ} = \text{CENTROID}, \quad (588)$$

(b) If

$$\text{pix\_adj} = \text{edser}, \quad (589)$$

then

$$\text{PIX\_ADJ} = \text{EDSER}, \quad (590)$$

(c) If

$$\text{pix\_adj} = \text{none}, \quad (591)$$

then

$$\text{PIX\_ADJ} = \text{NONE}, \quad (592)$$

(d) If

$$\text{pix\_adj} = \text{randomize}, \quad (593)$$

then

$$\text{PIX\_ADJ} = \text{RANDOMIZE}, \quad (594)$$

2. RAND\_SKY:

(a) If

$$\text{pix\_adj} = \text{centroid}, \quad (595)$$

then

$$\text{RAND\_SKY} = 0.0, \quad (596)$$

(b) If

$$\text{pix\_adj} = \text{edser}, \quad (597)$$

then

$$\text{RAND\_SKY} = 0.0, \quad (598)$$

(c) If

$$\text{pix\_adj} = \text{none}, \quad (599)$$

then

$$\text{RAND\_SKY} = 0.0, \quad (600)$$

(d) If

$$\text{pix\_adj} = \text{randomize}, \quad (601)$$

then

$$\text{RAND\_SKY} = 0.5, \quad (602)$$

3. TIME\_ADJ:

(a) TE mode:  
If

$$\text{DATAMODE}_{\text{in}} = \text{FAINT or} \quad (603)$$

$$\text{DATAMODE}_{\text{in}} = \text{FAINT\_BIAS or} \quad (604)$$

$$\text{DATAMODE}_{\text{in}} = \text{GRADED or} \quad (605)$$

$$\text{DATAMODE}_{\text{in}} = \text{VFAINT}, \quad (606)$$

then

$$\text{TIME\_ADJ} = \text{NONE}. \quad (607)$$

(b) Pointing CC mode without grating data:

i. If

$$\text{OBS\_MODE} = \text{pointing or} \quad (608)$$

$$\text{OBS\_MODE} = \text{POINTING} \quad (609)$$

and

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \quad (610)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED} \quad (611)$$

and

$$\text{CONTENT}_{\text{in}} = \text{EVT0 or} \quad (612)$$

$$\text{CONTENT}_{\text{in}} = \text{EVT1 or} \quad (613)$$

$$\text{CONTENT}_{\text{in}} = \text{EVT2} \quad (614)$$

and

$$\text{CCD\_ID}_{\text{focus}} \geq 0 \text{ and} \quad (615)$$

$$\text{CCD\_ID}_{\text{focus}} \leq 3 \quad (616)$$

and

$$\cos(\text{DEC\_ADJ}_I) \cos(\text{DEC\_TARG}_{\text{in}}) \cos(\text{RA\_ADJ}_I - \text{RA\_TARG}_{\text{in}}) + \quad (617)$$

$$\sin(\text{DEC\_ADJ}_I) \sin(\text{DEC\_TARG}_{\text{in}}) < \quad (618)$$

$$4.855 \times 10^{-11}, \quad (619)$$

then

$$\text{TIME\_ADJ} = \text{TARGET}. \quad (620)$$

ii. If

$$\text{OBS\_MODE} = \text{pointing or} \quad (621)$$

$$\text{OBS\_MODE} = \text{POINTING} \quad (622)$$

and

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \quad (623)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED} \quad (624)$$

and

$$\text{CONTENT}_{\text{in}} = \text{EVT0 or} \quad (625)$$

$$\text{CONTENT}_{\text{in}} = \text{EVT1 or} \quad (626)$$

$$\text{CONTENT}_{\text{in}} = \text{EVT2} \quad (627)$$

and

$$\text{CCD\_ID}_{\text{focus}} \geq 4 \text{ and} \quad (628)$$

$$\text{CCD\_ID}_{\text{focus}} \leq 9 \quad (629)$$

and

$$\cos(\text{DEC\_ADJ}_S) \cos(\text{DEC\_TARG}_{in}) \cos(\text{RA\_ADJ}_S - \text{RA\_TARG}_{in}) + \quad (630)$$

$$\sin(\text{DEC\_ADJ}_S) \sin(\text{DEC\_TARG}_{in}) < \quad (631)$$

$$4.855 \times 10^{-11}, \quad (632)$$

then

$$\text{TIME\_ADJ} = \text{TARGET}. \quad (633)$$

iii. If

$$\text{OBS\_MODE} = \text{pointing or} \quad (634)$$

$$\text{OBS\_MODE} = \text{POINTING} \quad (635)$$

and

$$\text{DATAMODE}_{in} = \text{CC33\_FAINT or} \quad (636)$$

$$\text{DATAMODE}_{in} = \text{CC33\_GRADED} \quad (637)$$

and

$$\text{CONTENT}_{in} = \text{EVT0 or} \quad (638)$$

$$\text{CONTENT}_{in} = \text{EVT1 or} \quad (639)$$

$$\text{CONTENT}_{in} = \text{EVT2} \quad (640)$$

and

$$\text{CCD\_ID}_{focus} \geq 0 \text{ and} \quad (641)$$

$$\text{CCD\_ID}_{focus} \leq 3 \quad (642)$$

and

$$\cos(\text{DEC\_ADJ}_I) \cos(\text{DEC\_TARG}_{in}) \cos(\text{RA\_ADJ}_I - \text{RA\_TARG}_{in}) + \quad (643)$$

$$\sin(\text{DEC\_ADJ}_I) \sin(\text{DEC\_TARG}_{in}) \geq \quad (644)$$

$$4.855 \times 10^{-11}, \quad (645)$$

then

$$\text{TIME\_ADJ} = \text{AIMPOINT}. \quad (646)$$

iv. If

$$\text{OBS\_MODE} = \text{pointing or} \quad (647)$$

$$\text{OBS\_MODE} = \text{POINTING} \quad (648)$$

and

$$\text{DATAMODE}_{in} = \text{CC33\_FAINT or} \quad (649)$$

$$\text{DATAMODE}_{in} = \text{CC33\_GRADED} \quad (650)$$

and

$$\text{CONTENT}_{in} = \text{EVT0 or} \quad (651)$$

$$\text{CONTENT}_{in} = \text{EVT1 or} \quad (652)$$

$$\text{CONTENT}_{in} = \text{EVT2} \quad (653)$$

and

$$\text{CCD\_ID}_{\text{focus}} \geq 4 \text{ and} \quad (654)$$

$$\text{CCD\_ID}_{\text{focus}} \leq 9 \quad (655)$$

and

$$\cos(\text{DEC\_ADJ}_S) \cos(\text{DEC\_TARG}_{\text{in}}) \cos(\text{RA\_ADJ}_S - \text{RA\_TARG}_{\text{in}}) + \quad (656)$$

$$\sin(\text{DEC\_ADJ}_S) \sin(\text{DEC\_TARG}_{\text{in}}) \geq \quad (657)$$

$$4.855 \times 10^{-11}, \quad (658)$$

then

$$\text{TIME\_ADJ} = \text{AIMPOINT}. \quad (659)$$

(c) Pointing CC mode with ACIS-S grating data:

If

$$\text{OBS\_MODE} = \text{pointing or} \quad (660)$$

$$\text{OBS\_MODE} = \text{POINTING} \quad (661)$$

and

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \quad (662)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED} \quad (663)$$

and

$$\text{CONTENT}_{\text{in}} = \text{TGEVT1} \quad (664)$$

and

$$\text{CCD\_ID}_{\text{focus}} \geq 4 \text{ and} \quad (665)$$

$$\text{CCD\_ID}_{\text{focus}} \leq 9, \quad (666)$$

then

$$\text{TIME\_ADJ} = \text{GRATING}. \quad (667)$$

(d) Pointing CC mode with ACIS-I grating data:

i. If

$$\text{OBS\_MODE} = \text{pointing or} \quad (668)$$

$$\text{OBS\_MODE} = \text{POINTING} \quad (669)$$

and

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \quad (670)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED} \quad (671)$$

and

$$\text{CONTENT}_{\text{in}} = \text{TGEVT1} \quad (672)$$



and

$$\text{CCD\_ID}_{\text{focus}} \geq 0 \text{ and} \quad (673)$$

$$\text{CCD\_ID}_{\text{focus}} \leq 3 \quad (674)$$

and

$$\cos(\text{DEC\_ADJ}_I) \cos(\text{DEC\_TARG}_{\text{in}}) \cos(\text{RA\_ADJ}_I - \text{RA\_TARG}_{\text{in}}) + \quad (675)$$

$$\sin(\text{DEC\_ADJ}_I) \sin(\text{DEC\_TARG}_{\text{in}}) < \quad (676)$$

$$4.855 \times 10^{-11}, \quad (677)$$

then

$$\text{TIME\_ADJ} = \text{TARGET}. \quad (678)$$

ii. If

$$\text{OBS\_MODE} = \text{pointing or} \quad (679)$$

$$\text{OBS\_MODE} = \text{POINTING} \quad (680)$$

and

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \quad (681)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED} \quad (682)$$

and

$$\text{CONTENT}_{\text{in}} = \text{TGEVT1} \quad (683)$$

and

$$\text{CCD\_ID}_{\text{focus}} \geq 0 \text{ and} \quad (684)$$

$$\text{CCD\_ID}_{\text{focus}} \leq 3 \quad (685)$$

and

$$\cos(\text{DEC\_ADJ}_I) \cos(\text{DEC\_TARG}_{\text{in}}) \cos(\text{RA\_ADJ}_I - \text{RA\_TARG}_{\text{in}}) + \quad (686)$$

$$\sin(\text{DEC\_ADJ}_I) \sin(\text{DEC\_TARG}_{\text{in}}) \geq \quad (687)$$

$$4.855 \times 10^{-11}, \quad (688)$$

then

$$\text{TIME\_ADJ} = \text{AIMPOINT}. \quad (689)$$

(e) Secondary CC mode:

If

$$\text{OBS\_MODE} \neq \text{pointing and} \quad (690)$$

$$\text{OBS\_MODE} \neq \text{POINTING} \quad (691)$$

and

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_FAINT or} \quad (692)$$

$$\text{DATAMODE}_{\text{in}} = \text{CC33\_GRADED}, \quad (693)$$

then

$$\text{TIME\_ADJ} = \text{MIDCHIP}. \quad (694)$$

## 2 TBD

- Complete the spec to include all of the timed exposure mode processing.
- Complete sections 1.1, 1.2, 1.3, and 1.4.
- 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 RA\_TARG, DEC\_TARG, RA\_NOM, DEC\_NOM, and TIMEDEL keywords in the output of afe (need obsfile sometimes)?
- What if TIME\_RO is not in the infile (output of afe? EVT2 files)?
- What if a small fraction of the values of CHIPY\_TARG are off the chip due to bad aspect?
- Make sure that the STATUS bits are unset and reset properly.
- What about aoff and soff files instead of asol files?
- Are the  $\beta$  in PHA\_RO the same as the  $\beta$  in PHA?
- Should something be done about SKY\_1D?