MIT
Center for Space Research **Chandra X-Ray Center**

MEMORANDUM
September 4, 2002

To:	Martin Elvis, SDS Group Leader
From:	Glenn Allen, SDS, for the CTI task group
Subject:	Adjusting ACIS Event Data to Compensate for CTI
Revision:	4.7
URL:	
File:	

The ACIS instrument teams at PSU and MIT have shown that a significant improvement in the energy resolution of existing ACIS event data can be obtained by compensating for some of the effects of the parallel charge-transfer inefficiency (CTI) of the front-side–illuminated CCDs and for the parallel and serial CTI of the back-side–illuminated CCDs. To achieve this improvement, charge is added to each 3×3 pixel event island and the charge within the event island is redistributed. The amount of charge added to each event is based on an estimate of the average amount of charge that is lost as charge packets are clocked across the charge traps on the ACIS detectors. The average amount of charge lost depends on the density of charge traps on the detector, the location of the event on the CCD, and the number of traps that have already been filled by "precursor events" in the same CCD column (i.e. on how many empty traps an event must cross to get to the read out). We propose to modify acis_process_events to use the algorithm described in section 1.4 to compute the CTI-adjusted values of PHAS. These adjusted values are contained in a new column called PHAS_ADJ. The original, unadjusted values are retained in the column PHAS. Once the values of PHAS_ADJ are computed, the values of GRADE, FLTGRADE, PHA, ENERGY, and PI are computed using the column PHAS_ADJ in the usual fashion. At the present, the calibration data is only appropriate for data obtained using some of the front-side–illuminated CCDs with frame times of about 3.2 s. Current FEF, GAIN, OSIP, QE, and QEU CALDB files are inappropriate for data adjusted using this technique. New CALDB files should be produced for CTI-adjusted data.

1 Changes to acis_process_events

1.1 Additional Parameters

apply_cti,b,a,"no","no","yes","Apply CTI adjustment?"
ctifile,s,h,"CALDB",,,"ACIS CTI file (NONE — none — CALDB — <filename>)"
max_cti_iter,i,h,15,1,20,"Maximum number of iterations for the CTI adjustment of each event"
cti_converge,r,h,0.1,0.1,1.0,"The convergence criterion for each CTI-adjusted pixel in adu"

When Catherine Grant tested the PSU CTI-adjustment tool, she found that the median number of iterations required to satisfy a convergence criterion of 0.1 adu is four. No event required more than ten iterations. Therefore, a default maximum of fifteen iterations should be sufficient to estimate the distribution of the charge deposited in the detector. We suggest that the maximum number of iterations be limited to the range of 1–20. The default convergence criterion is set to 0.1 adu because this is the default value used for the PSU CTI-adjustment tool. Since the utility of using values smaller than 0.1 adu is questionable, the minimum value for the parameter cti_converge is 0.1 adu. A maximum value of 1 adu seems reasonable for the convergence parameter.

Until the instrument team and SDS are satisfied with the algorithm and until new CALDB files are produced and tested, the CTI adjustment should not be applied to new data or reprocessed data by default.

1.2 Additional Input

The tool acis_process_events must read the CTI ARD file in addition to the other files it already reads. The format of the CTI ARD file is summarized in section 2.

1.3 Additional Output

If apply_cti = "yes," the output event file has the same format as the present Level 1 or 2 ACIS event files except that the file includes the additional column "PHAS_ADJ" and the additional keyword, "CTI_FILE." The new column contains the real (not integer) CTI-adjusted values of the charge distribution in the event island. The column named PHAS contains the original, unadjusted values of PHAS (i.e. the bias and overclock-subtracted values of PHAS obtained before any CTI adjustment is performed). The keyword CTI_FILE contains the name of the CTI ARD file used to process the data. If apply_cti = "no," no CTI adjustment is applied, the column PHAS_ADJ is not created, the column PHAS contains the unadjusted values of PHAS, and CTI_FILE = "NONE."

1.4 Processing

The tool should produce an error if apply_cti = "yes" and the input event-data file has READMODE $\neq$ TIMED or DATAMODE $\neq$ FAINT, FAINT_BIAS, or VFAINT. The tool should produce a warning that the columns GRADE, FLTGRADE, PHA, ENERGY, and PI will not contain CTI-adjusted information if apply_cti = "yes" and doevtgrade = "no." The tool should produce a warning if TIMEDEL is not in the range specified by the corresponding calibration-boundary ("CBD") keyword.

If apply_cti = "yes," the column PHAS_ADJ is created and populated with the values of the CTI-adjusted contents of PHAS. The column PHAS contains the unadjusted PHAS information. The original PHAS information is retained to ensure that users can analyze the unadjusted distribution of charge in the event island and to provide the input required if users want to rerun acis_process_events with an updated CTI ARD file at a later time. The algorithm used to compute the CTI-adjusted values of PHAS_ADJ is described later in this section. After the values of PHAS_ADJ are computed, the values of GRADE, FLTGRADE, PHA, ENERGY, and PI are computed as usual from the column PHAS_ADJ (not PHAS). The column PHA has integer values whether the CTI-adjustment is applied or not. The values of the columns GRADE, FLTGRADE, and PHA are computed if and only if doevtgrade = "yes" (and apply_cti = "yes"). The values of the columns ENERGY and PI are computed if and only if calculate_pi = "yes" (and apply_cti = "yes").

The columns PHAS_ADJ and PHAS are arrays. For TIMED FAINT-mode observations, the relative CHIPX and CHIPY coordinates associated with PHAS_ADJ$[i, j]$ are distributed as shown in figure 1 with i and $j \in [1,3]$. If a TIMED FAINT-mode event occurs at $(\mathrm{CHIPX}, \mathrm{CHIPY}) = (960, 571)$, then PHAS_ADJ$[1, 2]$ corresponds to the CCD coordinates (959,571) and PHAS_ADJ$[2, 3]$ corresponds to the CCD coordinates (960,572). The algorithm described below applies only to a 3×3 pixel event island. For TIMED VFAINT-mode observations, i and $j \in [1, 5]$ and the appropriate 3×3 pixel region to use is the central nine pixels of the 5×5 pixel event island (i.e. the region associated with PHAS$[i, j]$, where i and $j \in [2, 4]$ instead of i and $j \in [1, 5]$). The outer sixteen pixels of the 5×5 pixel event island are not modified by the CTI-adjustment algorithm. These sixteen pixels have identically the same values in the columns PHAS_ADJ and PHAS.

If apply_cti = "yes," the following steps describe how to apply the CTI adjustment. The adjustment is applied to every event in the input ACIS event-data file. For an event detected on CCD_ID = n at a location CHIPX = x_{CCD}, and CHIPY = y_{CCD},

1. Copy the contents of PHAS to PHAS_ADJ.
2. Copy the values of PHAS_ADJ (not PHAS) to PHAS_TMP.
3. For the bottom row of the 3×3 pixel event island, compute the effects of parallel CTI. Steps 3–5 should be performed only if the appropriate parallel-CTI trap-density map exists for CCD_ID $= n$. The CTI ARD may not contain some parallel-CTI trap-density maps if the effects of parallel CTI are not fully calibrated. The bottom row of PHAS$[i, j]$ is given by $i \in [1, 3], j = 1$ (fig. 1). If PHAS_ADJ$[i, 1] \geq$ the split threshold, then

$$\begin{aligned} \text{DIFF\_Y}[i,1] &= \text{DELTPHAY}_{i,1} + \\ &\quad \text{PHAS}[i,1] - \text{PHAS\_ADJ}[i,1]. \end{aligned} \tag{1}$$

The quantity DIFF_Y$[i, j]$ is an estimate of the amount of charge that should be added to pixel $[i, j]$ of the event island. The quantity DELTPHAY$_{i,j}$ represents the amount of charge lost from pixel $[i, j]$ due to parallel CTI and is a function of the CCD used, the location of an event on the CCD, and the charge deposited on the CCD.

The value of DELTPHAY$_{i,j}$ used in equation 1 is computed as follows.

   i. Find the row m in HDU 1 of the appropriate CTI ARD file that satisfies the conditions

$$\text{CCD\_ID}_m = n, \tag{2}$$

$$\text{CHIPXMIN}_m \leq x_{\text{CCD}} \leq \text{CHIPXMAX}_m, \text{ and} \tag{3}$$

$$\text{CHIPYMIN}_m \leq y_{\text{CCD}} \leq \text{CHIPYMAX}_m, \tag{4}$$

   where CCD_ID, CHIPXMIN, CHIPXMAX, CHIPYMIN, and CHIPYMAX are the names of columns in the CTI ARD file (see sec. 2).

   ii. For the row that satisfies these conditions, find the two non-zero (real!) values PHA$_k$ and PHA$_{k+1}$ such that

$$0 < \text{PHA}_k \leq \text{PHAS\_ADJ}[i,j] < \text{PHA}_{k+1}, \tag{5}$$

   where PHA$_k$ and PHA$_{k+1}$ are elements of the column PHA in the CTI ARD file.

   iii. Compute the effective "VOLUME" occupied by the charge in pixel $[i, j]$ using the linear interpolation

$$\begin{aligned} \text{VOLUME}_{i,j} &= \frac{\text{PHAS\_ADJ}[i,j] - \text{PHA}_k}{\text{PHA}_{k+1} - \text{PHA}_k} \times \left(\text{VOLUME\_Y}_{k+1} - \text{VOLUME\_Y}_k\right) + \\ &\quad \text{VOLUME\_Y}_k, \end{aligned} \tag{6}$$

   where VOLUME_Y$_k$ and VOLUME_Y$_{k+1}$ are the k^{th} and $(k+1)^{\text{th}}$ elements of the column VOLUME_Y in the CTI ARD file (see sec. 2). This formula is valid if and only if $1 \leq k <$ NPOINTS (i.e. if PHA$_1 \leq$ PHAS_ADJ$[i, j] <$ PHA$_{\text{NPOINTS}}$, where PHA$_1$ and PHA$_{\text{NPOINTS}}$ are the smallest and largest values of the vector PHA for row m, respectively). If $0 <$ PHAS_ADJ$[i, j] <$ PHA$_1$, then use the linear extrapolation

$$\begin{aligned} \text{VOLUME}_{i,j} &= \frac{\text{PHAS\_ADJ}[i,j] - \text{PHA}_1}{\text{PHA}_2 - \text{PHA}_1} \times \left(\text{VOLUME\_Y}_2 - \text{VOLUME\_Y}_1\right) + \\ &\quad \text{VOLUME\_Y}_1. \end{aligned} \tag{7}$$

   If PHAS_ADJ$[i, j] \geq$ PHA$_{\text{NPOINTS}}$, then use the linear extrapolation

$$\begin{aligned} \text{VOLUME}_{i,j} &= \frac{\text{PHAS\_ADJ}[i,j] - \text{PHA}_{\text{NPOINTS}-1}}{\text{PHA}_{\text{NPOINTS}} - \text{PHA}_{\text{NPOINTS}-1}} \times \\ &\quad \left(\text{VOLUME\_Y}_{\text{NPOINTS}} - \text{VOLUME\_Y}_{\text{NPOINTS}-1}\right) + \\ &\quad \text{VOLUME\_Y}_{\text{NPOINTS}-1}. \end{aligned} \tag{8}$$

   If PHAS_ADJ$[i, j] \leq 0$, then VOLUME$_{i,j} = 0$.

iv. Find the HDU in the CTI ARD file that has the parallel-CTI trap-density map for CCD_ID = n (see sec. 2). Set TRAPDENS[i,j] equal to the value of the map at the position (CHIPX, CHIPY) = (x_i, y_i), where $x_i = x_{\rm CCD} - 1$, $x_{\rm CCD}$, and $x_{\rm CCD} + 1$ for i = 1, 2, and 3 and $y_i = y_{\rm CCD} - 1$, $y_{\rm CCD}$, and $y_{\rm CCD} + 1$ for j = 1, 2, and 3.

v. Compute the value of DELTPHAY$_{i,j}$:

$$\mathrm{DELTPHAY}_{i,j} \;=\; \mathrm{TRAPDENS}_{i,j} \times \mathrm{VOLUME}_{i,j}. \tag{9}$$

4. For the middle and top rows of the 3×3 pixel event island, compute the effects of parallel CTI. The middle and top rows of PHAS[i,j] are given by $i \in [1,3]$, $j \in [2,3]$ (fig. 1). If PHAS_ADJ[i,j] $\geq$ the split threshold $>$ PHAS_ADJ[$i,j-1$], then

$$\mathrm{DIFF\_Y}[i,j] \;=\; \mathrm{DELTPHAY}_{i,j} + \mathrm{PHAS}[i,j] - \mathrm{PHAS\_ADJ}[i,j]. \tag{10}$$

If PHAS_ADJ[i,j] $\geq$ PHAS_ADJ[$i,j-1$] $\geq$ the split threshold, then

$$\mathrm{DIFF\_Y}[i,j] \;=\; (\mathrm{DELTPHAY}_{i,j} - \mathrm{DELTPHAY}_{i,j-1}) + \mathrm{PHAS}[i,j] - \mathrm{PHAS\_ADJ}[i,j]. \tag{11}$$

If PHAS_ADJ[$i,j-1$] $>$ PHAS_ADJ[i,j] $\geq$ the split threshold, then

$$\mathrm{DIFF\_Y}[i,j] \;=\; \mathrm{FRCTRLY}n \times (\mathrm{DELTPHAY}_{i,j} - \mathrm{DELTPHAY}_{i,j-1}) + \mathrm{PHAS}[i,j] - \mathrm{PHAS\_ADJ}[i,j], \tag{12}$$

where FRCTRLYn is the name of a keyword in the CTI ARD file (see sec. 2) and represents the fraction of the trapped charge that is released in the first pixel following the pixel which lost the charge.

5. Use the estimate of the effects of parallel CTI to adjust the 3×3 pixel event island PHAS_ADJ. For i and $j \in [1,3]$,

$$\mathrm{PHAS\_ADJ}[i,j] = \mathrm{PHAS\_ADJ}[i,j] + \mathrm{DIFF\_Y}[i,j]. \tag{13}$$

Pixels whose amount of charge $<$ the split threshold are not modified. These pixels have the same values in the columns PHAS and PHAS_ADJ.

6. For the column of a 3×3 pixel event island that is closest to the serial read-out node, compute the effects of serial CTI. Steps 6–8 should be performed only if the appropriate serial-CTI trap-density map exists for CCD_ID = n. The CTI ARD may not contain some serial-CTI trap-density maps if the effects of serial CTI are not fully calibrated.

i. For NODE_ID = 0, $i = 1$ and $j \in [1,3]$. If PHAS_ADJ[$1,j$] $\geq$ the split threshold, then

$$\mathrm{DIFF\_X}[1,j] \;=\; \mathrm{DELTPHAX}_{1,j} + \mathrm{PHAS}[1,j] - \mathrm{PHAS\_ADJ}[1,j]. \tag{14}$$

ii. For NODE_ID = 1, $i = 3$ and $j \in [1,3]$. If PHAS_ADJ[$3,j$] $\geq$ the split threshold, then

$$\mathrm{DIFF\_X}[3,j] \;=\; \mathrm{DELTPHAX}_{3,j} + \mathrm{PHAS}[3,j] - \mathrm{PHAS\_ADJ}[3,j]. \tag{15}$$

iii. For NODE_ID = 2, $i = 1$ and $j \in [1,3]$. If PHAS_ADJ[$1,j$] $\geq$ the split threshold, then

$$\mathrm{DIFF\_X}[1,j] \;=\; \mathrm{DELTPHAX}_{1,j} + \mathrm{PHAS}[1,j] - \mathrm{PHAS\_ADJ}[1,j]. \tag{16}$$

iv. For NODE_ID $= 3$, $i = 3$ and $j \in [1,3]$. If PHAS_ADJ$[3,j] \geq$ the split threshold, then

$$\begin{aligned} \text{DIFF\_X}[3,j] &= \text{DELTPHAX}_{3,j} + \\ &\quad \text{PHAS}[3,j] - \text{PHAS\_ADJ}[3,j]. \end{aligned} \tag{17}$$

The quantity DIFF_X$[i,j]$ is an estimate of the amount of charge that should be added to pixel $[i,j]$. The quantity $\text{DELTPHAX}_{i,j}$ represents the amount of charge lost from pixel $[i,j]$ due to the effects of serial CTI and is a function of the CCD used, the location of an event on the CCD, and the charge deposited on the CCD. This quantity is computed using the same linear interpolation (and extrapolation) method used to compute $\text{DELTPHAY}_{i,j}$, where VOLUME_Y is replaced with VOLUME_X and the parallel-CTI trap-density map for CCD_ID $= n$ is replaced with the serial-CTI trap-density map for the CCD.

7. For the two columns of a 3×3 pixel event island that are farthest from the serial read-out node, compute the effects of serial CTI.

i. For NODE_ID $= 0$, $i \in [2,3]$ and $j \in [1,3]$. If PHAS_ADJ$[i,j] \geq$ PHAS_ADJ$[i-1,j] \geq$ the split threshold, then

$$\begin{aligned} \text{DIFF\_X}[i,j] &= (\text{DELTPHAX}_{i,j} - \text{DELTPHAX}_{i-1,j}) + \\ &\quad \text{PHAS}[i,j] - \text{PHAS\_ADJ}[i,j]. \end{aligned} \tag{18}$$

If PHAS_ADJ$[i,j] \geq$ the split threshold and PHAS_ADJ$[i,j] <$ PHAS_ADJ$[i-1,j]$, then

$$\begin{aligned} \text{DIFF\_X}[i,j] &= \text{FRCTRLX}n \times (\text{DELTPHAX}_{i,j} - \text{DELTPHAX}_{i-1,j}) + \\ &\quad \text{PHAS}[i,j] - \text{PHAS\_ADJ}[i,j]. \end{aligned} \tag{19}$$

ii. For NODE_ID $= 1$, $i \in [1,2]$ and $j \in [1,3]$. If PHAS_ADJ$[i,j] \geq$ PHAS_ADJ$[i+1,j] \geq$ the split threshold, then

$$\begin{aligned} \text{DIFF\_X}[i,j] &= (\text{DELTPHAX}_{i,j} - \text{DELTPHAX}_{i+1,j}) + \\ &\quad \text{PHAS}[i,j] - \text{PHAS\_ADJ}[i,j]. \end{aligned} \tag{20}$$

If PHAS_ADJ$[i,j] \geq$ the split threshold and PHAS_ADJ$[i,j] <$ PHAS_ADJ$[i+1,j]$, then

$$\begin{aligned} \text{DIFF\_X}[i,j] &= \text{FRCTRLX}n \times (\text{DELTPHAX}_{i,j} - \text{DELTPHAX}_{i+1,j}) + \\ &\quad \text{PHAS}[i,j] - \text{PHAS\_ADJ}[i,j]. \end{aligned} \tag{21}$$

iii. For NODE_ID $= 2$, $i \in [2,3]$ and $j \in [1,3]$. If PHAS_ADJ$[i,j] \geq$ PHAS_ADJ$[i-1,j] \geq$ the split threshold, then

$$\begin{aligned} \text{DIFF\_X}[i,j] &= (\text{DELTPHAX}_{i,j} - \text{DELTPHAX}_{i-1,j}) + \\ &\quad \text{PHAS}[i,j] - \text{PHAS\_ADJ}[i,j]. \end{aligned} \tag{22}$$

If PHAS_ADJ$[i,j] \geq$ the split threshold and PHAS_ADJ$[i,j] <$ PHAS_ADJ$[i-1,j]$, then

$$\begin{aligned} \text{DIFF\_X}[i,j] &= \text{FRCTRLX}n \times (\text{DELTPHAX}_{i,j} - \text{DELTPHAX}_{i-1,j}) + \\ &\quad \text{PHAS}[i,j] - \text{PHAS\_ADJ}[i,j]. \end{aligned} \tag{23}$$

iv. For NODE_ID $= 3$, $i \in [1,2]$ and $j \in [1,3]$. If PHAS_ADJ$[i,j] \geq$ PHAS_ADJ$[i+1,j] \geq$ the split threshold, then

$$\begin{aligned} \text{DIFF\_X}[i,j] &= (\text{DELTPHAX}_{i,j} - \text{DELTPHAX}_{i+1,j}) + \\ &\quad \text{PHAS}[i,j] - \text{PHAS\_ADJ}[i,j]. \end{aligned} \tag{24}$$

If PHAS_ADJ$[i, j] \geq$ the split threshold and PHAS_ADJ$[i, j] <$ PHAS_ADJ$[i+1, j]$, then

$$\text{DIFF\_X}[i,j] = \text{FRCTRLX}n \times (\text{DELTPHAX}_{i,j} - \text{DELTPHAX}_{i+1,j}) + \text{PHAS}[i,j] - \text{PHAS\_ADJ}[i,j], \quad (25)$$

where FRCTRLXn is the name of a keyword in the CTI ARD file (see sec. 2) and represents the fraction of the trapped charge that is released in the first pixel following the pixel which lost the charge.

8. Use the estimate of the effects of serial CTI to adjust the 3×3 pixel event island PHAS_ADJ. For i and $j \in [1, 3]$,

$$\text{PHAS\_ADJ}[i,j] = \text{PHAS\_ADJ}[i,j] + \text{DIFF\_X}[i,j]. \quad (26)$$

Pixels whose amount of charge < the split threshold are not modified. These pixels have the same values in the columns PHAS and PHAS_ADJ.

9. Repeat steps 2–8 until the absolute values of PHAS_ADJ$[i, j]$ − PHAS_TMP$[i, j]$ are less than the value of the parameter cti_converge for all pixels in the 3×3 pixel event island (i.e. until the changes in each of the values of PHAS_ADJ$[i, j]$ from one iteration to the next are less than the value of the parameter cti_converge). The maximum number of times that steps 2–8 should be performed is specified by the parameter max_cti_iter. If the CTI-adjustment process does not converge after max_cti_iter iterations, set the values of PHAS_ADJ to be the values obtained during the last iteration and set STATUS bit 20 (of bits 0–31) equal to one.

10. If doevtgrade = "yes," compute the values of GRADE, FLTGRADE, and PHA as usual. If apply_cti = "yes," use the values of PHAS_ADJ instead of the values of PHAS to compute the values of GRADE, FLTGRADE, and PHA. Otherwise, use the values of PHAS to compute the values of GRADE, FLTGRADE, and PHA. The values of PHA are integers in either case.

11. If calculate_pi = "yes," compute the values of ENERGY and PI as usual.

12. Write out the results. Write out the column PHAS_ADJ only if apply_cti = "yes."

2 New CTI ARD

Since the effects of CTI are temperature dependent, a different CTI ARD file is required for each of the different focal-plane temperatures. The CTI ARD files have the following structure.

2.1 HDU 1 Keywords

- FRCTRLXn
- FRCTRLYn

The keywords FRCTRLXn and FRCTRLYn represent the fraction of the trapped charge that is released in the first pixel following the pixel which lost the charge.

2.2 HDU 1 Columns

- CCD_ID
- CHIPXMIN
- CHIPXMAX
- CHIPYMIN
- CHIPYMAX

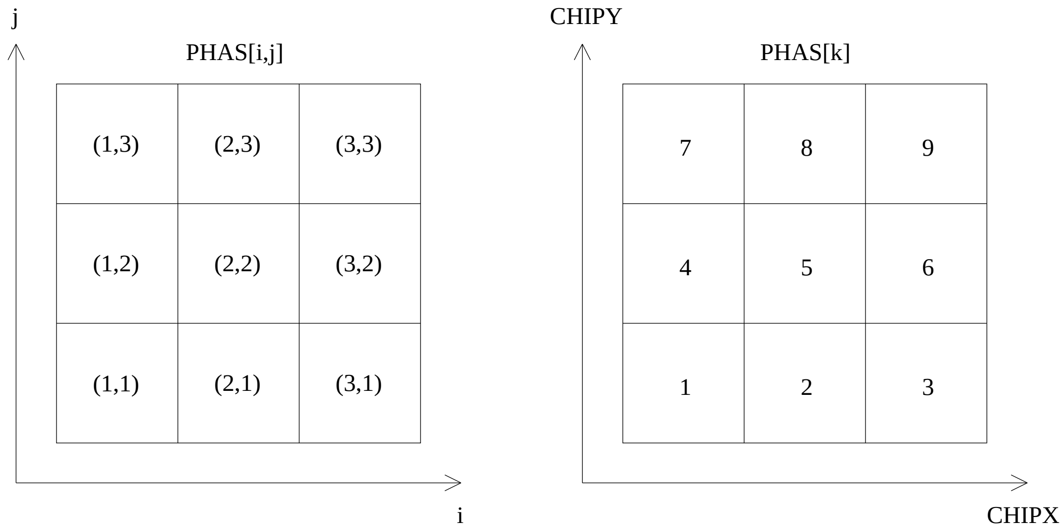

Figure 1: The relative CHIPX and CHIPY coordinates of the nine elements of a 3×3 ACIS event island (i.e. the nine elements of PHAS and PHAS_ADJ for TIMED FAINT-mode events).

- NPOINTS
- PHA (a vector with NPOINTS elements)
- VOLUME_X (a vector with NPOINTS elements)
- VOLUME_Y (a vector with NPOINTS elements)

The columns CCD_ID, CHIPXMIN, CHIPXMAX, CHIPYMIN, and CHIPYMAX are used to define a complete set of spatially-separate regions on the ACIS CCDs. The pulse-height–dependent effects of serial and parallel CTI are tabulated for each region in the columns PHA, VOLUME_X, and VOLUME_Y. These three columns are vector columns. Each row of HDU 1 corresponds to one region of an ACIS CCD and has NPOINTS PHA, VOLUME_X, and VOLUME_Y elements.

2.3 HDUs 2–21

Each one of these twenty extensions contains either a parallel or serial trap-density map for one of the ten ACIS CCDs. The serial and parallel-CTI trap-density maps are recorded separately for each CCD. The trap density of each of the 1024×1024 pixels in a map is the mean trap density for the pixel multiplied by the CHIPY coordinate of the pixel (i.e. the integrated trap density along the column associated with the pixel). The trap densities are stored as two-byte integers with appropriate keywords BZERO and BSCALE in the headers of each extension. (The trap density = BZERO + BSCALE $\times$ the value of the trap density in the image.) The use of two-byte integers instead of four-byte real numbers helps reduce the size of the ARD file.

2.4 Size of File

The CTI ARD files are relatively large. Each row of HDU 1 has six two-byte integers and three four-byte real vectors with NPOINTS elements each. If the CTI ARD contains information for one region (the entire CCD) on each of the ten ACIS CCDs and if NPOINTS = 100, the binary table of HDU 1 comprises $10 \times 1 \times (6 \times 2 + 3 \times 4 \times 100)$ bytes = 12.1 kb of information. The size of each of the twenty trap-density maps is 2.1 Mb (i.e. 1024×1024 pixels $\times$ 2 bytes/pixel). Unless the number of regions per CCD becomes much larger than one, the size of HDU 1 is much smaller than the size of the trap-density maps and the size of one CTI ARD file is 42 Mb (i.e. 20 hdus $\times$ 2.1 Mb/hdu).

URL: http://space.mit.edu/~gea/docs/memo_cti_correction_4.7.ps