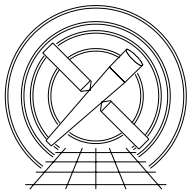

MIT Kavli Institute **Chandra X-Ray Center**

MEMORANDUM
July 15, 2014

To:	Jonathan McDowell, SDS Group Leader
From:	Glenn E. Allen, SDS
Subject:	`acis_build_chip_gti`
Revision:	2.1
URL:	http://space.mit.edu/CXC/docs/docs.html#gti
File:	/inconceivable/d0/SDS/specs/acis_build_chip_gti/acis_build_chip_gti_2.1.tex

1 `acis_build_chip_gti`

1.1 Description

For an ACIS CCD, a good-time interval (GTI) is a continuous set of frames during which the detector operated nominally. The GTIs can differ from one CCD to another if more than one is used for an observation.

1.2 Input

1. A Level 1 exposure-statistics file (acis*stat1.fits)
2. A Level 1 event-data file (acis*evt1.fits)
   (This file is also the output file.)
3. A parameter-block file (acisf*pbk0.fits)
4. One or more Level 1 aspect-solution file(s) (pcad*asol1.fits)
5. A parameter file that includes a list of the CALDB files used by pixlib
6. The `CCD_ID` of the aim-point CCD

1.3 Output

The input Level 1 event-data file is modified to include

- several exposure-related keywords and
- one GTI HDU for each valid CCD.

1.4 Parameters

1. `infile`,f,a,“”,,,“Name of input exposure-statistics file”
2. `outfile`,f,a,“”,,,“Name of event-data file to be modified”
3. `pbkfile`,f,a,“”,,,“Name of input parameter-block file”
4. `aspfile`,f,a,“”,,,“Name(s) of input aspect-solution file(s)”
5. `geompar`,f,h,“geom”,,,“Name of input pixlib parameter file”
6. `nominalchip`,s,h,“default”,default|0|1|2|3|4|5|6|7|8|9,,“Aim-point CCD_ID”
7. `verbose`,i,h,0,,,
8. `mode`,s,h,“ql”,,,

1.5 Processing

1. Error checking:
   (a) `infile`:
      i. If the `infile` does not exist, then `acis_build_chip_gti` exits with an error message.
      ii. If the `infile` exists, but the file permissions do not allow it to be read, then `acis_build_chip_gti` exits with an error message.
      iii. If the `infile` does not have an HDU $h_{\rm in}$ with the keyword
      $$\texttt{CONTENT} = \text{EXPSTATS}, \tag{1}$$
      then `acis_build_chip_gti` exits with an error message.
      iv. If HDU $h_{\rm in}$ does not include the header keywords
      - `FLSHTIME`,
      - `TIMEDEL`, and
      - `TIMEPIXR`,

      then `acis_build_chip_gti` exits with an error message. Hereafter the values of these keywords are referred to as $\texttt{FLSHTIME}_{\rm in}$, $\texttt{TIMEDEL}_{\rm in}$, and $\texttt{TIMEPIXR}_{\rm in}$, respectively.
      v. If HDU $h_{\rm in}$ does not include a binary table with the columns
      - `CCD_ID`,
      - `EVTSENT`,
      - `EXPNO`, and
      - `TIME`,

      then `acis_build_chip_gti` exits with an error message. Hereafter these columns are referred to as $\texttt{CCD\_ID}_{\rm in}$, $\texttt{EVTSENT}_{\rm in}$, $\texttt{EXPNO}_{\rm in}$, and $\texttt{TIME}_{\rm in}$, respectively.
   (b) `outfile`:
      i. If the `outfile` does not exist, then `acis_build_chip_gti` exits with an error message.
      ii. If the `outfile` exists, but the file permissions do not allow it to be read and modified, then `acis_build_chip_gti` exits with an error message.
      iii. If the `outfile` does not have an HDU $h_{\rm out}$ with the keyword
      $$\texttt{CONTENT} = \text{EVT1}, \tag{2}$$
      then `acis_build_chip_gti` exits with an error message.

iv. If HDU $h_{\rm out}$ of the `outfile` does not include the header keyword
- `DATAMODE`,

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

v. If

$$\texttt{DATAMODE}_{\rm out} \neq \text{CC33\_FAINT and} \quad (3)$$
$$\texttt{DATAMODE}_{\rm out} \neq \text{CC33\_GRADED and} \quad (4)$$
$$\texttt{DATAMODE}_{\rm out} \neq \text{FAINT and} \quad (5)$$
$$\texttt{DATAMODE}_{\rm out} \neq \text{FAINT\_BIAS and} \quad (6)$$
$$\texttt{DATAMODE}_{\rm out} \neq \text{GRADED and} \quad (7)$$
$$\texttt{DATAMODE}_{\rm out} \neq \text{VFAINT}, \quad (8)$$

then `acis_build_chip_gti` exits with an error message.

vi. If

$$\texttt{DATAMODE}_{\rm out} = \text{CC33\_FAINT or} \quad (9)$$
$$\texttt{DATAMODE}_{\rm out} = \text{CC33\_GRADED}, \quad (10)$$

and if HDU $h_{\rm out}$ does not include the keywords `RA_TARG`and `DEC_TARG`, then `acis_build_chip_gti` exits with an error message. Hereafter the values of these keywords are referred to as $\texttt{RA\_TARG}_{\rm out}$, and $\texttt{DEC\_TARG}_{\rm out}$, respectively.

vii. If

$$\texttt{DATAMODE}_{\rm out} = \text{CC33\_FAINT or} \quad (11)$$
$$\texttt{DATAMODE}_{\rm out} = \text{CC33\_GRADED}, \quad (12)$$

if HDU $h_{\rm out}$ includes the keyword `HDUCLAS3`, and if `HDUCLAS3` = CC_CORRECTED, then $t_{\rm adj}$ = yes. Otherwise, $t_{\rm adj}$ = no.

viii. If HDU $h_{\rm out}$ does not include a binary table with the columns
- `CCD_ID` and
- `EXPNO`,

then `acis_build_chip_gti` exits with an error message. Hereafter these columns are referred to as $\texttt{CCD\_ID}_{\rm out}$ and $\texttt{EXPNO}_{\rm out}$, respectively.

(c) `pbkfile`:

i. If the `pbkfile` does not exist, then `acis_build_chip_gti` exits with an error message.

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

iii. If the `pbkfile` does not have an HDU $h_{\rm pbk}$ with the keyword

$$\texttt{CONTENT} = \text{PBK}, \quad (13)$$

then `acis_build_chip_gti` exits with an error message.

(d) `aspfile`:

i. If the `aspfile` does not exist, then `acis_build_chip_gti` exits with an error message.

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

iii. If the `aspfile` does not have an HDU $h_{\rm asp}$ that has the keyword

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

then `acis_build_chip_gti` exits with an error message.

iv. If HDU $h_{\rm asp}$ does not include a binary table with the columns

- `DEC`,
- `DTHETA`,
- `DY`,
- `DZ`,
- `RA`,
- `ROLL`, and
- `TIME`,

then `acis_build_chip_gti` exits with an error message.

(e) `nominalchip`:

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

ii. If

$$\begin{aligned}
\texttt{nominalchip} &\neq \text{default and} && (15)\\
\texttt{nominalchip} &\neq 0 \text{ and} && (16)\\
\texttt{nominalchip} &\neq 1 \text{ and} && (17)\\
\texttt{nominalchip} &\neq 2 \text{ and} && (18)\\
\texttt{nominalchip} &\neq 3 \text{ and} && (19)\\
\texttt{nominalchip} &\neq 4 \text{ and} && (20)\\
\texttt{nominalchip} &\neq 5 \text{ and} && (21)\\
\texttt{nominalchip} &\neq 6 \text{ and} && (22)\\
\texttt{nominalchip} &\neq 7 \text{ and} && (23)\\
\texttt{nominalchip} &\neq 8 \text{ and} && (24)\\
\texttt{nominalchip} &\neq 9, && (25)
\end{aligned}$$

then `acis_build_chip_gti` exits with an error message.

iii. If

$$\begin{aligned}
\texttt{nominalchip} &= 0 \text{ or} && (26)\\
\texttt{nominalchip} &= 1 \text{ or} && (27)\\
\texttt{nominalchip} &= 2 \text{ or} && (28)\\
\texttt{nominalchip} &= 3 \text{ or} && (29)\\
\texttt{nominalchip} &= 4 \text{ or} && (30)\\
\texttt{nominalchip} &= 5 \text{ or} && (31)\\
\texttt{nominalchip} &= 6 \text{ or} && (32)\\
\texttt{nominalchip} &= 7 \text{ or} && (33)\\
\texttt{nominalchip} &= 8 \text{ or} && (34)\\
\texttt{nominalchip} &= 9, && (35)
\end{aligned}$$

then $c_{\rm aim}$ is the integer equivalent of the string `nominalchip`.

iv. If `nominalchip` = default, then the value of $c_{\rm aim}$ is determined from the input data.

2. DTCOR:
The value of DTCOR, which does not depend on the CCD_ID, is given by the following expressions.

(a) If

$$\texttt{DATAMODE}_{\rm out} = \text{FAINT or} \tag{36}$$
$$\texttt{DATAMODE}_{\rm out} = \text{FAINT\_BIAS or} \tag{37}$$
$$\texttt{DATAMODE}_{\rm out} = \text{GRADED or} \tag{38}$$
$$\texttt{DATAMODE}_{\rm out} = \text{VFAINT,} \tag{39}$$

then

$$\texttt{DTCOR} = \frac{\texttt{TIMEDEL}_{\rm in} - 0.04104}{\texttt{TIMEDEL}_{\rm in} + \texttt{FLSHTIME}_{\rm in}}. \tag{40}$$

DTCOR includes $\texttt{FLSHTIME}_{\rm in}$ because it is included in the GTIs.

(b) If

$$\texttt{DATAMODE}_{\rm out} = \text{CC33\_FAINT or} \tag{41}$$
$$\texttt{DATAMODE}_{\rm out} = \text{CC33\_GRADED,} \tag{42}$$

then

$$\texttt{DTCOR} = 0.99609375 \tag{43}$$

(i.e. 510/512).

3. $\texttt{EXPNO}_{\rm min}$:
The value of $\texttt{EXPNO}_{\rm min}$ is determined as follows.

(a) If

$$\texttt{DATAMODE}_{\rm out} = \text{FAINT or} \tag{44}$$
$$\texttt{DATAMODE}_{\rm out} = \text{FAINT\_BIAS or} \tag{45}$$
$$\texttt{DATAMODE}_{\rm out} = \text{GRADED or} \tag{46}$$
$$\texttt{DATAMODE}_{\rm out} = \text{VFAINT,} \tag{47}$$

then $\texttt{EXPNO}_{\rm min} = 3$.

(b) If

$$\texttt{DATAMODE}_{\rm out} = \text{CC33\_FAINT or} \tag{48}$$
$$\texttt{DATAMODE}_{\rm out} = \text{CC33\_GRADED,} \tag{49}$$

then $\texttt{EXPNO}_{\rm min} = 4$.

4. Valid CCD_IDs:
The set of valid CCD_IDs is identified. Here, CCD_ID c is defined as valid if there are one or more rows (i.e. frames) in HDU $h_{\rm in}$ of the infile where $\texttt{CCD\_ID}_{\rm in} = c$. If there are no valid CCD_IDs, then acis_build_chip_gti exits with an error message. If $c_{\rm aim}$ is not a valid CCD_ID, then acis_build_chip_gti exits with an error message.

5. For each valid CCD_ID:
The following steps are performed for each valid CCD_ID c.

(a) Identify the valid EXPNOs:
If

$$\begin{aligned} \texttt{CCD\_ID}_{\rm in}[r_{\rm in}] &= c, && (50)\\ \texttt{EXPNO}_{\rm in}[r_{\rm in}] &\geq \texttt{EXPNO}_{\rm min}, && (51)\\ \texttt{EVTSENT}_{\rm in}[r_{\rm in}] &> 0, \text{ and} && (52)\\ \texttt{EVTSENT}_{\rm in}[r_{\rm in}] &= N_{\rm out}, && (53) \end{aligned}$$

then $\texttt{EXPNO}_{\rm in}[r_{\rm in}]$ is valid. Here $N_{\rm out}$, which is greater than or equal to zero, is the number of rows (i.e. events) where

$$\begin{aligned} \texttt{CCD\_ID}_{\rm out}[r_{\rm out}] &= c \text{ and} && (54)\\ \texttt{EXPNO}_{\rm out}[r_{\rm out}] &= \texttt{EXPNO}_{\rm in}[r_{\rm in}] && (55) \end{aligned}$$

and $r_{\rm in}$ and $r_{\rm out}$ are rows in HDUs $h_{\rm in}$ and $h_{\rm out}$ of the `infile` and `outfile`, respectively. If there are no valid EXPNOs for CCD_ID c, then `acis_build_chip_gti` exits with an error message.

(b) GTI START and STOP times:
Each consecutive set of one or more valid EXPNOs from $\texttt{EXPNO}_i$ to $\texttt{EXPNO}_j$ for CCD_ID c is a GTI for the CCD. The GTI START and STOP times associated with this set of EXPNOs is given by the following expressions.

i. If

$$\begin{aligned} \texttt{DATAMODE}_{\rm out} &= \text{FAINT or} && (56)\\ \texttt{DATAMODE}_{\rm out} &= \text{FAINT\_BIAS or} && (57)\\ \texttt{DATAMODE}_{\rm out} &= \text{GRADED or} && (58)\\ \texttt{DATAMODE}_{\rm out} &= \text{VFAINT}, && (59) \end{aligned}$$

then

$$\begin{aligned} \texttt{START} &= \texttt{TIME}_{\rm in}[r_{{\rm in},i}] - \texttt{TIMEPIXR}_{\rm in} \times \texttt{TIMEDEL}_{\rm in} - \texttt{FLSHTIME}_{\rm in} \text{ and} && (60)\\ \texttt{STOP} &= \texttt{TIME}_{\rm in}[r_{{\rm in},j}] + (1 - \texttt{TIMEPIXR}_{\rm in}) \times \texttt{TIMEDEL}_{\rm in}, && (61) \end{aligned}$$

where the row $r_{{\rm in},i}$ is the one where

$$\begin{aligned} \texttt{CCD\_ID}_{\rm in}[r_{{\rm in},i}] &= c \text{ and} && (62)\\ \texttt{EXPNO}_{\rm in}[r_{{\rm in},i}] &= \texttt{EXPNO}_i && (63) \end{aligned}$$

and the row $r_{{\rm in},j}$ is the one where

$$\begin{aligned} \texttt{CCD\_ID}_{\rm in}[r_{{\rm in},j}] &= c \text{ and} && (64)\\ \texttt{EXPNO}_{\rm in}[r_{{\rm in},j}] &= \texttt{EXPNO}_j. && (65) \end{aligned}$$

The GTIs include the $\texttt{FLSHTIME}_{\rm in}$ because DTCOR includes it.

ii. If

$$\begin{aligned} \texttt{DATAMODE}_{\rm out} &= \text{CC33\_FAINT or} && (66)\\ \texttt{DATAMODE}_{\rm out} &= \text{CC33\_GRADED} && (67) \end{aligned}$$

and if $t_{\rm adj} = \text{yes}$, then

$$\begin{aligned} \texttt{START} &= \texttt{TIME}_{\rm in}[r_{{\rm in},i}] - \texttt{TIMEPIXR}_{\rm in} \times \texttt{TIMEDEL}_{\rm in} \text{ and} && (68)\\ \texttt{STOP} &= \texttt{TIME}_{\rm in}[r_{{\rm in},j}] + (1 - \texttt{TIMEPIXR}_{\rm in}) \times \texttt{TIMEDEL}_{\rm in}. && (69) \end{aligned}$$

iii. If

$$\texttt{DATAMODE}_{\rm out} = \text{CC33\_FAINT or} \tag{70}$$
$$\texttt{DATAMODE}_{\rm out} = \text{CC33\_GRADED} \tag{71}$$

and if $t_{\rm adj} = \text{no}$, then

$$\begin{aligned}\texttt{START} = {} & \texttt{TIME}_{\rm in}[r_{{\rm in},i}] - \texttt{TIMEPIXR}_{\rm in} \times \texttt{TIMEDEL}_{\rm in} + \\ & (\texttt{CHIPY\_TARG}[\texttt{TIME}_{\rm in}[r_{{\rm in},i}]] + 1028) \times 0.00285 \text{ and} \end{aligned} \tag{72}$$
$$\begin{aligned}\texttt{STOP} = {} & \texttt{TIME}_{\rm in}[r_{{\rm in},j}] + (1 - \texttt{TIMEPIXR}_{\rm in}) \times \texttt{TIMEDEL}_{\rm in} + \\ & (\texttt{CHIPY\_TARG}[\texttt{TIME}_{\rm in}[r_{{\rm in},j}]] + 1028) \times 0.00285, \end{aligned} \tag{73}$$

where $\texttt{CHIPY\_TARG}[\texttt{TIME}_{\rm in}[r_{{\rm in},i}]]$ and $\texttt{CHIPY\_TARG}[\texttt{TIME}_{\rm in}[r_{{\rm in},j}]]$ are the CHIPY locations of the sky coordinates $\texttt{RA\_TARG}_{\rm out}$ and $\texttt{DEC\_TARG}_{\rm out}$ at the times $\texttt{TIME}_{\rm in}[r_{{\rm in},i}]$ and $\texttt{TIME}_{\rm in}[r_{{\rm in},j}]$, respectively. The `aspfile`, `geompar`, and pixlib are used to determine the values of $\texttt{CHIPY\_TARG}[\texttt{TIME}_{\rm in}[r_{{\rm in},i}]]$ and $\texttt{CHIPY\_TARG}[\texttt{TIME}_{\rm in}[r_{{\rm in},j}]]$.

(c) ONTIMEc:
The value of ONTIMEc is given by

$$\texttt{ONTIME}c = \sum_k \left(\texttt{STOP}_k - \texttt{START}_k\right), \tag{74}$$

where $\texttt{START}_k$ and $\texttt{STOP}_k$ are the START and STOP times, respectively, of the $k^{\rm th}$ GTI for CCD_ID c.

(d) LIVTIMEc:
The value of LIVTIMEc is given by

$$\texttt{LIVTIME}c = \texttt{ONTIME}c \times \texttt{DTCOR}. \tag{75}$$

(e) EXPOSURc:
The value of EXPOSURc is given by

$$\texttt{EXPOSUR}c = \texttt{ONTIME}c \times \texttt{DTCOR} \tag{76}$$

(i.e. EXPOSURc = LIVTIMEc).

6. Aim-point CCD:
If the CCD_ID $c = c_{\rm aim}$, then

$$\texttt{ONTIME} = \texttt{ONTIME}c, \tag{77}$$
$$\texttt{LIVETIME} = \texttt{LIVTIME}c, \text{ and} \tag{78}$$
$$\texttt{EXPOSURE} = \texttt{EXPOSUR}c. \tag{79}$$

7. Write output:

(a) The HDU $h_{\rm out}$ of the `outfile` is modified to include the keywords

- DTCOR,
- EXPOSURE,
- EXPOSURc (one such keyword for each valid CCD_ID),
- LIVETIME,
- LIVTIMEc (one such keyword for each valid CCD_ID),
- ONTIME, and

- ONTIME*c* (one such keyword for each valid CCD_ID),

(b) The outfile is modified to include a GTI HDU for each valid CCD. Each one of these HDUs includes a binary table with the columns

- START and
- STOP.

These columns include one row for each GTI for the CCD.

1.6 TBD

- Are there DATAMODEs other than CC33_FAINT, CC33_GRADED, FAINT, FAINT_BIAS, GRADED, and VFAINT that should be included?
- How is the aim-point CCD_ID determined? DETNAM? SIM_Z? $(X, Y) = (4096.5, 4096.5)$?
- Is there a systematic offset for the mission-timeline GTIs with respect to the GTIs produced by acis_build_chip_gti?
- How can the mission-timeline GTIs be merged with the acis_build_chip_gti GTIs so that they are aligned with the frame boundaries?