# ACIS Software IP&CL Structure Definition Notes

MIT 36-53204.0204 Rev. N

## 1.0 Purpose

This document describes the conventions used in defining the ACIS Software IP&CL Structures definitions, and to provide an initial list of enumerated code values used within the structures.

This version of the IP&CL software structures are stored under *patches/ipcl* CVS Tag "release-N". To obtain a copy of the source (MIT only):

```
setenv CVSROOT /nfs/acis/h3/acisfs/configcntl
cvs export -r release-N patches/ipcl
```

## 2.0 References

**TABLE 1. Reference Documents**

| Part Number | Version | Title |
|---|---|---|
| MIT 36-01103 | G | ACIS Science Instrument Software Requirements Specification |
| MIT 36-02104 | C | DPA Hardware Specification and System Description |
| MIT 36-02203 | A | Focal Plane to Detector Housing Interface Control Document |
| MIT 36-02205 | C | DPA to DEA Interface Control Document |

## 3.0 Naming Conventions

In general, all record and field names contain one or more words, separated, by a single space, with the first letter in each word capitalized, and the remaining letters of each word in lower-case. The following abbreviations have been used in the naming of the record and field names:

**1d** .....- 1 Dimensional (1D)

**2d** .....- 2 Dimensional (2D)

**Arg**.....- Argument

**Bep**.....- Back End Processor (BEP)

**Cc**.....- Continuous Clocking Mode

**Ccd**.....- Charge Coupled Device (CCD)

**Config** .....- Configuration

**Dea**.....- Detector Electronics Assembly (DEA)

**Fep**.....- Front End Processor (FEP)

**Id** .....- Identifier

**Oc**.....- Overclock

**Pram**.....- DEA Program RAM (PRAM)

**Sram**.....- DEA Sequencer RAM (SRAM)

**Sw** .....- Software

**Te** .....- Timed Exposure Mode

**WD**.....- Watchdog Timer

# 4.0 Packet Sizes

All command packets contain between 3 and 256 16-bit words (inclusive). All telemetry packets will contain between 2 and 1023 32-bit words (inclusive).

# 5.0 Constants

This section describes constant codes used with the IP&CL structures. These codes are preliminary, and will be final upon release of the instrument software and hardware (some constants are defined by the DEA and DPA hardware design). These constants are listed below in the form of “C” programming enumerated types, or pre-processor definitions. Unless otherwise specified, the value of an enumerated symbol is the value of the preceding symbol plus 1. The first symbol in the enumerated list has a value of 0.

## 5.1 Command Opcodes

The following is the current list of command opcodes use by the instrument software. These values are used in the “Command Opcode” field of all command packets. These codes are provided in “acis_h/interface.h”

```
enum CmdOpcode
{
    CMDOP_UNUSED,              // Unused Opcode

    // ---- Load from Uplink Commands ----
    CMDOP_START_UPLOAD,        // Start Load From Uplink
    CMDOP_CONTINUE_UPLOAD,     // Continue Load From Uplink

    // ---- Memory Read/Write/Execute Commands ----
    CMDOP_READ_BEP,            // Read BEP Memory
   // BEP Write and Execute are now near the end of the table

    CMDOP_READ_FEP,            // Read FEP Memory
    CMDOP_WRITE_FEP,           // Write FEP Memory
```

```
 CMDOP_EXEC_FEP,           // Execute FEP Memory

 CMDOP_READ_PRAM,          // Read DEA PRAM
// Write PRAM is now near the end of the table

 CMDOP_READ_SRAM,          // Read DEA SRAM

 // ---- Load Parameter Block Commands ----
 CMDOP_LOAD_TE,            // Load Timed Exposure Block
 CMDOP_LOAD_CC,            // Load Continuous Clocking Block
 CMDOP_LOAD_2D,            // Load 2D Window List
 CMDOP_LOAD_1D,            // Load 1D Window List
 CMDOP_LOAD_DEA,           // Load DEA Housekeeping Block

 // ---- Start/Stop Run Commands ----
 CMDOP_START_TE,           // Start Timed Exposure Run
 CMDOP_BIAS_TE,            // Start Timed Exposure Bias Run

 CMDOP_START_CC,           // Start Continuous Clocking Run
 CMDOP_BIAS_CC,            // Start Continuous Clocking Bias Run

 CMDOP_START_DEA,          // Start DEA Housekeeping Run

 CMDOP_STOP_SCIENCE,       // Stop Science Run
 CMDOP_STOP_DEA,           // Stop DEA Housekeeping Run

 // ---- Patch Commands ----
 CMDOP_ADD_PATCH,          // Add Patches
 CMDOP_REMOVE_PATCH,       // Remove Patches

 // ---- System Configuration/Bad Pixel List Commands ----
 CMDOP_ADD_BAD_PIXEL,      // Add Bad Pixel
 CMDOP_RESET_BAD_PIXEL,    // Reset Bad Pixel List
 CMDOP_DUMP_BAD_PIXELS,    // Dump Bad Pixel List

 CMDOP_ADD_BAD_TE_COL,     // Add Bad Timed Exposure Column
 CMDOP_RESET_BAD_TE_COL,   // Reset Bad Timed Exposure Column List
 CMDOP_DUMP_BAD_TE_COL,    // Dump Bad Timed Exposure Column List

 CMDOP_ADD_BAD_CC_COL,     // Add Bad Cont. Clocking Column
 CMDOP_RESET_BAD_CC_COL,   // Reset Bad Cont. Clocking Column List
 CMDOP_DUMP_BAD_CC_COL,    // Dump Bad Cont. Clocking Column List

 CMDOP_CHANGE_SYS_ENTRY,   // Change System Configuration Settings
 CMDOP_DUMP_SYS_CONFIG,    // Dump System Configuration Settings

CMDOP_DUMP_PATCHLIST,      // Dump PatchList
CMDOP_DUMP_HUFFMAN,        // Dump Huffman Table
CMDOP_DUMP_TE_SLOTS,       // Dump Timed Exposure Block Slots
CMDOP_DUMP_CC_SLOTS,       // Dump Continuous Clocking Block Slots
CMDOP_DUMP_2D_SLOTS,       // Dump 2D Window Parameter Block Slots
CMDOP_DUMP_1D_SLOTS,       // Dump 1D Window Parameter Block Slots
CMDOP_DUMP_DEA_SLOTS,      // Dump DEA Housekeeping Block Slots

// ---- Two-bit difference codes ----
CMDOP_WRITE_BEP = 0xc0,    // Write BEP Memory
CMDOP_EXEC_BEP = 0xc3,     // Execute BEP Memory

CMDOP_WRITE_PRAM = 0xcc,   // Write DEA PRAM
```

```
    CMDOP_WRITE_SRAM = 0xf0,   // Write DEA SRAM

    // ---- Miscellaneous Codes ----
    CMDOP_COUNT,               // Total # of potential Command Opcodes
    CMDOP_LASTID = CMDOP_COUNT - 1, // Value of last opcode

    CMDOP_INVALID = 0xffff     // Code for an invalid opcode
};
```

[see Command Header: Command Opcode]

## 5.2 Command Execution Result Codes

The following is the current list of command result use by the instrument software when echoing the command to telemetry. These values are used in the “Result” field of “Command Echo” telemetry packets. These codes are provided in “acis_h/interface.h”

```
enum CmdResult
{
   CMDRESULT_UNUSED,          // Unused response code

   CMDRESULT_OK,              // Command Successfully Dispatched
   CMDRESULT_NO_HANDLER,      // No handler for command opcode
   CMDRESULT_BUSY,            // Target of command is busy
   CMDRESULT_BAD_ARGUMENT,    // Command contains a bad parameter
   CMDRESULT_CORRUPT_DEFAULT, // Parameter Blk corrupt, use default
   CMDRESULT_CORRUPT_IDLE,    // Parameter Blk/Default corrupt, no run
   CMDRESULT_TABLE_FULL,      // Pixel/Column/Patch Table Full
   CMDRESULT_TABLE_EMPTY,     // Patch Table Empty
   CMDRESULT_INVALID_PKT,     // Command Packet Header corrupt
   CMDRESULT_BOARD_OFF,       // Selected Board has no power
   CMDRESULT_BOARD_RESET,     // Select Board is Reset
   CMDRESULT_STORE_ERROR,     // Error while storing parameter block

   CMDRESULT_INHIBITED,       // Run was inhibited prior to stop/start
   CMDRESULT_CLOBBERED,       // Operation aborted by new operation

   CMDRESULT_ITEM_CLIPPED,    // Input value was clipped to limit

   CMDRESULT_COUNT,           // Total number of result codes
   CMDRESULT_LASTID = CMDRESULT_COUNT - 1
};
```

[see Command Echo: Result]

## 5.3 Telemetry Format Tags

The following is the current list of telemetry format tags. These appear in the “Format Tag” field of the header portion of all telemetry packets. These codes are provided in “acis_h/interface.h”

```
enum TlmFormatTag
{
    TTAG_UNUSED,              // Unused format tag

    // ---- Memory Command Responses ----
    TTAG_READ_BEP,            // Read Back End Memory
    TTAG_READ_FEP,            // Read Front End Memory
    TTAG_READ_SRAM,           // Read SRAM
    TTAG_READ_PRAM,           // Read Pram
    TTAG_EXEC_BEP,            // Execute Back End Subroutine
    TTAG_EXEC_FEP,            // Execute Front End Subroutine

    // ---- Command Echoes ----
    TTAG_CMD_ECHO,            // Echoed command

    // ---- Miscellaneous Housekeeping ----
    TTAG_STARTUP,             // Startup Message
    TTAG_FATAL,               // Fatal Error Message
    TTAG_SW_HOUSE,            // Software Housekeeping
    TTAG_DEA_HOUSE,           // DEA Housekeeping

    // ---- Parameter Dumps ----
    TTAG_DUMP_TE,             // Timed Exposure Parameters + 2D Window
    TTAG_DUMP_CC,             // Cont. Clocking Parameters + 1D Windows

    // ---- Miscellaneous Science ----
    TTAG_SCI_TE_BIAS,         // Timed Exposure Bias Map
    TTAG_SCI_REPORT,          // Science Run Report

    // ---- Timed Exposure Science ----
    TTAG_SCI_TE_REC_RAW,      // Timed Exposure Raw Mode Exposure Hdr
    TTAG_SCI_TE_DAT_RAW,      // Timed Exposure Raw Mode Data
    TTAG_SCI_TE_REC_HIST,     // Timed Exposure Histogram Mode Exp. Hdr
    TTAG_SCI_TE_DAT_HIST,     // Timed Exposure Histogram Mode Data
    TTAG_SCI_TE_REC_FAINT,    // Timed Exposure Faint Mode Exp. Hdr
    TTAG_SCI_TE_DAT_FAINT,    // Timed Exposure Faint Mode Data
    TTAG_SCI_TE_REC_FAINTB,   // Timed Exposure Faint-Bias Mode Exp. Hdr
    TTAG_SCI_TE_DAT_FAINTB,   // Timed Exposure Faint-Bias Mode Data
    TTAG_SCI_TE_REC_GRADED,   // Timed Exposure Graded Mode Exp. Hdr
    TTAG_SCI_TE_DAT_GRADED,   // Timed Exposure Graded Mode Data

    // ---- Continuous Clocking Science ----
    TTAG_SCI_CC_REC_RAW,      // Continuous Clk Raw Mode Exposure Hdr
    TTAG_SCI_CC_DAT_RAW,      // Continuous Clk Raw Mode Data
    TTAG_SCI_CC_REC_FAINT,    // Continuous Clk Faint Mode Exp. Hdr
    TTAG_SCI_CC_DAT_FAINT,    // Continuous Clk Faint Mode Data
    TTAG_SCI_CC_REC_GRADED,   // Continuous Clk Graded Mode Exp. Hdr
    TTAG_SCI_CC_DAT_GRADED,   // Continuous Clk Graded Mode Data

    TTAG_SCI_CC_BIAS,         // Continuous Clocking Bias Map
    TTAG_SCI_BIAS_ERROR,      // Science Bias Error Data Packet
```

```
    TTAG_DUMP_SYS_CONFIG,     // Dump of System Configuration Table
    TTAG_DUMP_BAD_PIXEL,      // Dump of Bad Pixel Map
    TTAG_DUMP_BAD_TE_COL,     // Dump of Bad Timed Exposure Columns
    TTAG_DUMP_BAD_CC_COL,     // Dump of Bad Cont. Clocking Columns
    TTAG_DUMP_PATCHES,        // Dump of Patch List
    TTAG_DUMP_HUFFMAN,        /* Dump of Huffman Tables */
    TTAG_DUMP_TE_SLOTS,       /* Dump of Timed Exposure Slots */
    TTAG_DUMP_CC_SLOTS,       /* Dump of Cont. Clocking Slots */
    TTAG_DUMP_2D_SLOTS,       /* Dump of 2D Window Slots */
    TTAG_DUMP_1D_SLOTS,       /* Dump of 1D Window Slots */
    TTAG_DUMP_DEA_SLOTS,      /* Dump of DEA Housekeeping Slots */

    TTAG_FILL_PATTERN = 45,   /* Tag received if hw fill pattern used */

    TTAG_SCI_TE_DAT_FAINT_5x5 /* Timed Exposure Faint 5x5 Data */
    TTAG_SCI_TE_REC_FAINT_5x5,/* Timed Exposure Faint 5x5 Exp. Hdr */

    /* ---- NEW PATCHED TAGS ---- */
    TTAG_SCI_TE_DAT_EV_HIST,    /* 3x3 Event Histogram Data */
    TTAG_SCI_TE_REC_EV_HIST,    /* 3x3 Event Histogram Exposure Record */

    TTAG_SCI_PATCHED_BIAS_ERROR = 50, /* Patched Bias Error Packet */

    TTAG_SCI_CC_DAT_FAINT3x3,    /* Continuous Clocking 3x3 Mode Data */
    TTAG_SCI_CC_REC_FAINT3x3,    /* Continuous Clocking 3x3 Mode Record */

    TTAG_SCI_CC_DAT_GRADED3x3,   /* Continuous Clock Graded 3x3 Mode Data */
    TTAG_SCI_CC_REC_GRADED3x3,   /* Continuous Clock Graded 3x3 Mode Record */

    TTAG_SCI_TE_DAT_CTI1,        /* Timed Exposure with CTI correction Data */
    TTAG_SCI_TE_REC_CTI1,        /* Timed Exposure with CTI correction Hdr */

    // ---- Miscellaneous Codes ----
    TTAG_RESERVED = 0x3f,     /* Reserved for Maintenance Use */

    TTAG_COUNT,               /* Number of Format Tags */
    TTAG_LASTID = TTAG_COUNT-1
};
```

[see Telemetry Header: Format Tag]

## 5.4 Software Bi-level Discrete Telemetry (LED) Codes

The following is the current list of software bi-level discrete telemetry codes (LED codes) (NOTE: The remaining bi-levels used by the BEP hardware are not described in this section). These values appear within the bi-level discrete telemetry in a TBD location (by TRW) of the engineering portion of the telemetry frames. These codes are provided in “acis_h/interface.h”

```
enum LedState
{
 LED_WD_SCIENCE_A,         // WD Science Active - Blink State A
 LED_WD_SCIENCE_B,         // WD Science Active - Blink State B
 LED_WD_IDLE_A,            // WD Idle - Blink State A
 LED_WD_IDLE_B,            // WD Idle - Blink State B
 LED_RUN_SCIENCE_A,        // Science Active - Blink State A
```

```
 LED_RUN_SCIENCE_B,        // Science Active - Blink State B
 LED_RUN_IDLE_A,           // Idle - Blink State A
 LED_RUN_IDLE_B,           // Idle - Blink State B
 LED_RUN_STARTUP,          // Initializing loaded code
 LED_RUN_PATCH,            // About to patch loaded code
 LED_BOOT_UPLINK_EXECUTE,  // About to execute loaded code
 LED_BOOT_UPLINK_COPY,     // Copying packets from uplink FIFO
 LED_BOOT_UPLINK_WAIT,     // Waiting for initial load from uplink pkt.
 LED_BOOT_SPARE1,          // Spare (was About to execute copied code)
 LED_BOOT_SPARE2,          // Spare (was Copying ROM into Bep RAM)
 LED_BOOT_RESET            // Bep was just reset
};
```

## 5.5 CCD Identifiers

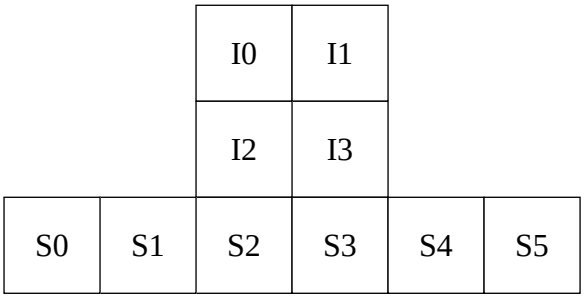
The following values are used to identify a CCD. These codes are provided in “acis_h/interface.h”

```
enum CcdId
{
    CCD_I0,         // Imaging CCD I0
    CCD_I1,         // Imaging CCD I1
    CCD_I2,         // Imaging CCD I2
    CCD_I3,         // Imaging CCD I3

    CCD_S0,         // Spectroscopy CCD S0
    CCD_S1,         // Spectroscopy CCD S1
    CCD_S2,         // Spectroscopy CCD S2
    CCD_S3,         // Spectroscopy CCD S3
    CCD_S4,         // Spectroscopy CCD S4
    CCD_S5,         // Spectroscopy CCD S5

   CCD_DESELECT,    // Code used to indicate no CCD selection
   CCD_COUNT = CCD_DESELECT,// Number of selectable CCD Codes
   CCD_LASTID = CCD_COUNT - 1
};
```

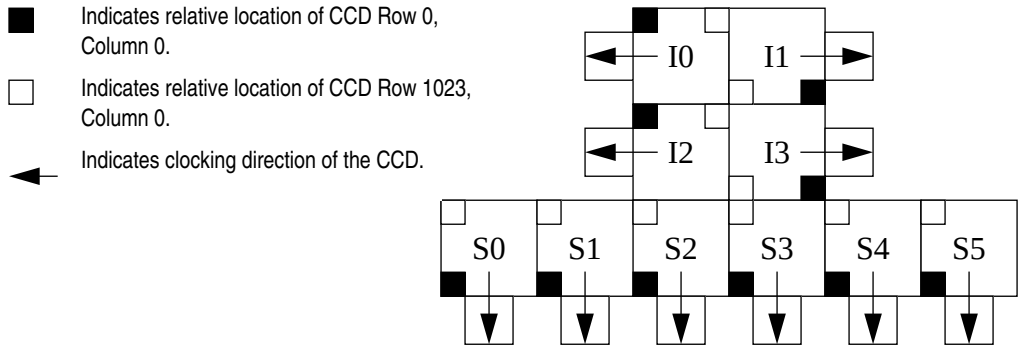
For reference purposes only, the following diagram illustrates the relative layout of the indicated CCDs in the focal plane assembly, as viewed from the High Resolution Mirror Assembly (HRMA). This information was obtained from the “Focal Plane to Detector Housing Interface Control Document”, MIT 36-02203, Rev. A.




## 5.6 CCD Row and Column Position Definition

The following illustration defines the CCD Row and Column positions used by the ACIS Software. Row 0 is defined to be the CCD imaging row closest to the Framestore. Column

0 is defined to be the left-most imaging column (i.e. the first to be clocked out of output node A).




## 5.7 Event Grade Code Definition

The following illustration defines the CCD Grade Code bit-definitions for a 3x3 event, where the row and column positions are indicated relative to the reported center of the event.

**3x3 Event**

| row | | | | |
|---|---|---|---|---|
| | 1 | 5 | 6 | 7 |
| | 0 | 3 | | 4 |
| | -1 | 0 | 1 | 2 |
| | | -1 | 0 | 1 |

**column**

**Grade Code Bits**

| msb | | | | | | | lsb |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

The following illustration defines the CCD Grade Code bit-definitions for a 1x3 event, where the column positions are indicated relative to the reported center of the event.


## 5.8 Huffman Compression Table Format

The on-board Huffman table array consists of a 32-word index table followed by a set of Huffman compression tables. Each word in the table is indexed by table slot id, and indicates the offset of the corresponding Huffman Compression table after the index table (i.e. a value of 0 indicates that the Huffman table immediately follows the index table, a value of 200 indicates that the table is 200 32-bit words after the index table). If an index table word is 0xffffffff, there is no Huffman table referenced by corresponding slot.

The format of each Huffman table is as follows:

| 32-bit Words | Item | |
|---|---|---|
| 1 | Table Identifier | |
| 1 | Low Limit | |
| 1 | Table Size | |
| 1 | Truncation Code | |
| 1 | Bad Bias Code | |
| 1 | Bad Pixel Code | |
| Table Size | Huffman Code Entries | |
| | The Truncation Code, Bad Bias Code, Bad Pixel Code and Huffman Code entries have the following format: | |
| | **Bits** | **Item** |
| | 5 | Code Bit Length |
| | 32 - (Code Bit Length + 5) | Pad |
| | Code Bit Length | Huffman Code |

## 5.9 DEA PRAM/SRAM Load Format

The DEA PRAM/SRAM load format is as follows:

| 16-bit Words | Item | |
|---|---|---|
| 1 | Sequence Type (always 0) | |
| 1 | Section Count | |
| Variable | Sections | |
| | where the format of each section is as follows: | |
| | **16-bit Words** | **Item** |
| | 1 | CCD Controller Select |
| | | where: |
| | | bit 0 selects CCD I0 (0 - no load, 1 - load) |
| | | bit 1 selects CCD I1 |
| | | bit 2 selects CCD I2 |
| | | bit 3 selects CCD I3 |
| | | bit 4 selects CCD S0 |
| | | bit 5 selects CCD S1 |
| | | bit 6 selects CCD S2 |
| | | bit 7 selects CCD S3 |
| | | bit 8 selects CCD S4 |
| | | bit 9 selects CCD S5 |
| | | If bit 15 is 0, load the data into SRAM |
| | | If bit 15 is 1, load the data into PRAM |
| | 1 | Index into SRAM or PRAM |
| | 1 | Word Count |
| | Word Count | Data to Load into SRAM or PRAM |

## 5.10 FEP Load Format

The Front End Processor load format is as follows:

| 16-bit Words | Item |
|---|---|
| 1 | Program Type (always 0) |
| 1 | Section Count |
| 1 | Execute Address (low order 16-bits) |
| | Execute Address (high order 16-bits) |
| 1 | Command Mailbox Address (low order 16-bits) |
| 1 | Command Mailbox Address (high order 16-bits) |
| 1 | Ring Buffer Address (low order 16-bits) |
| 1 | Ring Buffer Address (high order 16-bits) |

| 16-bit Words | Item |
|---|---|
| Variable | Sections |

where the format of each section is as follows:

| 16-bit Words | Item |
|---|---|
| 1 | Section Type (always 0) |
| 1 | Load Address (low order 16-bits) |
| 1 | Load Address (high order 16-bits) |
| 1 | Number of 16-bit words to load (MUST BE EVEN) |
| Word Count | Data to Load |

## 5.11 FEP Identifiers

The following are used to identify specific Front End Processors. These codes are provided in “acis_h/interface.h”

```
enum FepId
{
    FEP_0,          // FEP Slot 0 Identifier
    FEP_1,          // FEP Slot 1 Identifier
    FEP_2,          // FEP Slot 2 Identifier
    FEP_3,          // FEP Slot 3 Identifier
    FEP_4,          // FEP Slot 4 Identifier
    FEP_5,          // FEP Slot 5 Identifier

   FEP_COUNT,       // Number of FEP Ids
   FEP_LASTID = FEP_COUNT - 1
};
```

## 5.12 Video Chain Identifiers

The following codes are used to identify a video chain from a single CCD. These codes are provided in “acis_h/interface.h”

```
enum QuadId
{
    ONODE_A,        // Output Node/Video Chain A
    ONODE_B,        // Output Node/Video Chain B
    ONODE_C,        // Output Node/Video Chain C
    ONODE_D         // Output Node/Video Chain D

   ONODE_COUNT,     // Number of Output Node Ids
   ONODE_LASTID = ONODE_COUNT - 1
};
```

## 5.13 Output Register Clocking Modes

The following codes are used to indicate how to clock a CCD’s output register. These codes are provided in “acis_h/interface.h”

```
enum QuadMode
{
   QUAD_FULL,       // Clock charge to all four output nodes
   QUAD_DIAG,       // Clock charge away from all four output nodes
   QUAD_AC,         // Clock charge toward output nodes A and C
   QUAD_BD,         // Clock charge toward output nodes B and D

   QUAD_COUNT,      // Number of Quadrant Clocking Modes
   QUAD_LASTID = QUAD_COUNT - 1
};
```

[see Load Cc Block: Output Register Mode, Load Te Block: Output Register Mode]

## 5.14 DEA Query Identifiers

The following lists the currently defined DEA Housekeeping Query identifiers. These codes are provided in “acis_h/interface.h”.

```
enum DeaQueryCntlId
{
  /* ---- Interface Board Queries ---- */
  DEAHOUSE_CNTL_BASE = 0,         /* Base Query for controller board */
                                  /* Relay Positions */
  DEAHOUSE_CNTL_RELAY=DEAHOUSE_CNTL_BASE,

  DEAHOUSE_CNTL_ADC_BASE,          /* ADC Base Query */
                                   /* DPA Thermistor 1 - BEP PC Board */
  DEAHOUSE_CNTL_ADC_TMP_BEP_PCB=DEAHOUSE_CNTL_ADC_BASE,
  DEAHOUSE_CNTL_ADC_TMP_BEP_OSC,    /* DPA Thermistor 2 - BEP Oscillator */
  DEAHOUSE_CNTL_ADC_TMP_FEP0_MONG,  /* DPA Thermistor 3 - FEP 0 Mongoose */
  DEAHOUSE_CNTL_ADC_TMP_FEP0_PCB,   /* DPA Thermistor 4 - FEP 0 PC Board */
  DEAHOUSE_CNTL_ADC_TMP_FEP0_ACTEL, /* DPA Thermistor 5 - FEP 0 ACTEL */
  DEAHOUSE_CNTL_ADC_TMP_FEP0_RAM,   /* DPA Thermistor 6 - FEP 0 RAM */
  DEAHOUSE_CNTL_ADC_TMP_FEP0_FB,    /* DPA Thermistor 7 - FEP 0 Frame Buf. */
  DEAHOUSE_CNTL_ADC_TMP_FEP1_MONG,  /* DPA Thermistor 8 - FEP 1 Mongoose */
  DEAHOUSE_CNTL_ADC_TMP_FEP1_PCB,   /* DPA Thermistor 9 - FEP 1 PC Board */
  DEAHOUSE_CNTL_ADC_TMP_FEP1_ACTEL, /* DPA Thermistor 10- FEP 1 ACTEL */
  DEAHOUSE_CNTL_ADC_TMP_FEP1_RAM,   /* DPA Thermistor 11- FEP 1 RAM */
  DEAHOUSE_CNTL_ADC_TMP_FEP1_FB,    /* DPA Thermistor 12- FEP 1 Frame Buf. */
  DEAHOUSE_CNTL_ADC_SUBAHK,      /* DEA Video Board ADC */
  DEAHOUSE_CNTL_ADC_SPARE1,      /* Spare - Unused */
  DEAHOUSE_CNTL_ADC_FPTEMP_12,      /* Spare - Focal Plane Temp. Board 12 */
  DEAHOUSE_CNTL_ADC_FPTEMP_11,      /* Spare - Focal Plane Temp. Board 11 */
  DEAHOUSE_CNTL_ADC_DPAGNDREF1,     /* DPA Ground Reference 1 */
  DEAHOUSE_CNTL_ADC_DPA5VHKA,     /* DPA 5V Housekeeping A */
  DEAHOUSE_CNTL_ADC_DPAGNDREF2,     /* DPA Ground Reference 2 */
  DEAHOUSE_CNTL_ADC_DPA5VHKB,     /* DPA 5V Housekeeping B */
  DEAHOUSE_CNTL_ADC_UNUSED1,      /* Unused */
  DEAHOUSE_CNTL_ADC_UNUSED2,      /* Unused */
  DEAHOUSE_CNTL_ADC_UNUSED3,      /* Unused */
  DEAHOUSE_CNTL_ADC_UNUSED4,      /* Unused */
  DEAHOUSE_CNTL_ADC_DEA28VDCA,      /* PSMC A DEA 28V DC */
  DEAHOUSE_CNTL_ADC_DEA24VDCA,      /* PSMC A DEA 24V DC */
  DEAHOUSE_CNTL_ADC_DEAM15VDCA,     /* PSMC A DEA -15.5V */
  DEAHOUSE_CNTL_ADC_DEAP15VDCA,     /* PSMC A DEA +15.5V */
  DEAHOUSE_CNTL_ADC_DEAM6VDCA,      /* PSMC A DEA -6V DC */
  DEAHOUSE_CNTL_ADC_DEAP6VDCA,      /* PSMC A DEA +6V DC */
  DEAHOUSE_CNTL_ADC_RAD_PCB_A,      /* Relative Dose Rad. Monitor Side A */
  DEAHOUSE_CNTL_ADC_GND_1,       /* Interface Ground Reference */
  DEAHOUSE_CNTL_ADC_DEA28VDCB,      /* PSMC B DEA 28V DC */
  DEAHOUSE_CNTL_ADC_DEA24VDCB,      /* PSMC B DEA 24V DC */
  DEAHOUSE_CNTL_ADC_DEAM15VDCB,     /* PSMC B DEA -15.5V DC */
  DEAHOUSE_CNTL_ADC_DEAP15VDCB,     /* PSMC B DEA +15.5V DC */
  DEAHOUSE_CNTL_ADC_DEAM6VDCB,      /* PSMC B DEA -6V DC */
  DEAHOUSE_CNTL_ADC_DEAP6VDCB,      /* PSMC B DEA +6V DC */
  DEAHOUSE_CNTL_ADC_RAD_PCB_B,      /* Relavtive Dose Rad. Monitor Side B */
  DEAHOUSE_CNTL_ADC_GND_2,       /* Ground */

  DEAHOUSE_CNTL_ADC_END = DEAHOUSE_CNTL_ADC_GND_2,
  DEAHOUSE_CNTL_END = DEAHOUSE_CNTL_ADC_END,
};
```

```
enum DeaQueryCcdId
{
    // ---- CCD Controller Queries ----
    DEAHOUSE_CCD_BASE = 0,       // Base Query for CCD Queries
                               // Register 0 Sequencer Control
    DEAHOUSE_CCD_REG_0 = DEAHOUSE_CCD_BASE,
    DEAHOUSE_CCD_REG_1,        // Register 1 Video ADC Control
    DEAHOUSE_CCD_REG_2,        // Register 2
    DEAHOUSE_CCD_REG_3,        // Register 3

    DEAHOUSE_CCD_ADC_BASE = 0x80, // Base Index for ADC Registers
                               // Image Array Parallel +
    DEAHOUSE_CCD_PIA_P = DEAHOUSE_CCD_ADC_BASE,
    DEAHOUSE_CCD_PIA_M,        // Image Array Parallel -
    DEAHOUSE_CCD_PFS_P,        // Framestore Parallel +
    DEAHOUSE_CCD_PFS_M,        // Framestore Parallel -
    DEAHOUSE_CCD_S_P,          // Serial Register +
    DEAHOUSE_CCD_S_M,          // Serial Register -
    DEAHOUSE_CCD_R_P,          // Reset Gate +
    DEAHOUSE_CCD_R_M,          // Reset Gate -
    DEAHOUSE_CCD_OG,          // Output Gate Bias Level
    DEAHOUSE_CCD_SCP,          // Scupper
    DEAHOUSE_CCD_RD,           // Reset Diode
    DEAHOUSE_CCD_DR0,          // Drain Output Channel A
    DEAHOUSE_CCD_DR1,          // Drain Output Channel B
    DEAHOUSE_CCD_DR2,          // Drain Output Channel C
    DEAHOUSE_CCD_DR3,          // Drain Output Channel D
    DEAHOUSE_CCD_SPARE,           // Spare Housekeeping Channel
    DEAHOUSE_CCD_TEMP_BOARD,    // Board Temperature (RTD4)
    DEAHOUSE_CCD_TEMP_SRAM,     // SRAM Temperature (RTD3)
    DEAHOUSE_CCD_TEMP_ADC,      // ADC Temperature (RTD2)
    DEAHOUSE_CCD_TEMP_ACTEL,    // Gate Array Temperature (RTD1)

    DEAHOUSE_CCD_END = DEAHOUSE_CCD_TEMP_ACTEL // Last CCD Query
};
```

[see Query Dea Housekeeping: Query Id]

These identifiers are based on the Rev. C release of the DPA to DEA ICD (MIT 36-02205).

Queries which fail, either due to a time-out or because the queried board is not powered, will be indicated in the telemetered housekeeping packet as having a value of 0xffff in the queried data:

```
#define DEAHOUSE_VALUE_INVALID (0xffff)/* Result of Failed Query */
```

## 5.15 System Configuration Item Identifiers

The following lists the currently defined System Configuration Item Identifiers. These codes are provided in “acis_h/interface.h”

```
enum SystemSettings
{
    SYSSET_DEA_POWER,             // DEA CCD-controller Power Settings
    SYSSET_FEP_POWER              // FEP Power Settings
```

```
// ---- DEA Interface Board Settings ----
 SYSSET_CNTL_BASE,            // Base settings for controller board
                                // Master clock during science
 SYSSET_CNTL_MASTER_CLK = SYSSET_CNTL_BASE,
 SYSSET_CNTL_FOCAL_TEMP,       // Focal Plane Temperature
 SYSSET_CNTL_BAKE_TEMP,        // BakeOut Temperature
 SYSSET_CNTL_BAKE_ENABLE,      // BakeOut Enable[1]
 SYSSET_CNTL_LED_ENABLE,       // LED Enable
 SYSSET_CNTL_HOUSE_HOLD,       // Hold Housekeeping Address
 SYSSET_CNTL_SIGNAL_PATH,      // Signal Path Selection
 SYSSET_CNTL_CMDCLOCK_DISABLE, // Command Clock Enable Select
 SYSSET_CNTL_CMDDATA_DISABLE, // Command Data Enable Select
 SYSSET_CNTL_RELAY_SET_0,      // DEA Board Relay Selections
 SYSSET_CNTL_RELAY_SET_1,
 SYSSET_CNTL_RELAY_SET_2,
 SYSSET_CNTL_RELAY_SET_3,
 SYSSET_CNTL_RELAY_SET_4,
 SYSSET_CNTL_END = SYSSET_CNTL_RELAY_SET_4,
 SYSSET_CNTL_COUNT = SYSSET_CNTL_END - SYSSET_CNTL_BASE + 1,

 // ---- DEA CCD Controller Register Settings ----
 SYSSET_CCD_BASE = SYSSET_CNTL_END+1, // Base Settings for CCDs

                                // Sequencer Offset (for all CCDs)
 SYSSET_CCD_SEQ_OFFSET = SYSSET_CCD_BASE,
 SYSSET_CCD_ADC_OFFSET,        // Video ADC Offset
 SYSSET_CCD_VIDEO_ENABLE,      // Video Channel Enable Mask
 SYSSET_CCD_HOLD_HOUSE,         // Hold Housekeeping Address
 SYSSET_CCD_BJD,               // Back-Junction Diode Enable
 SYSSET_CCD_HIGH_SPEED_TAP,    // High-speed tap disable

 // ---- DEA CCD Controller Digital-To-Analog Converter Settings ---
 SYSSET_DAC_BASE,               // Delimit Start of DAC codes
 SYSSET_DAC_PIA_P = SYSSET_DAC_BASE, // Image Array Parallel +
 SYSSET_DAC_PIA_MP,            // Image Array Parallel -+
 SYSSET_DAC_PIA_M,             // Image Array Parallel -

 SYSSET_DAC_PFS_P,             // Framestore Parallel +
 SYSSET_DAC_PFS_MP,            // Framestore Parallel -+
 SYSSET_DAC_PFS_M,             // Framestore Parallel -

 SYSSET_DAC_S_P,               // Serial Register +
 SYSSET_DAC_S_M,               // Serial Register -

 SYSSET_DAC_R_P,               // Reset Gate +
 SYSSET_DAC_R_MP,              // Reset Gate -+
 SYSSET_DAC_R_M,               // Reset Gate -

 SYSSET_DAC_SCP,               // Scupper

 SYSSET_DAC_OG_P,              // Output Gate +
 SYSSET_DAC_OG_M,              // Output Gate -

 SYSSET_DAC_RD,                // Reset Diode
```

1. Bakeout Enable is now prevented from being used without a small patch to the limit table. Attempts to set this field to any value other than zero will result in a CMDRESULT_ITEM_CLIPPED result code.

```
        SYSSET_DAC_DR0,              // Drain Output (A)
        SYSSET_DAC_DR1,              // Drain Output (B)
        SYSSET_DAC_DR2,              // Drain Output (C)
        SYSSET_DAC_DR3,              // Drain Output (D)

        SYSSET_DAC_A_OFF,            // A channel offset
        SYSSET_DAC_B_OFF,            // B channel offset
        SYSSET_DAC_C_OFF,            // C channel offset
        SYSSET_DAC_D_OFF,            // D channel offset

        SYSSET_DAC_SPARE,            // Spare DAC Channel
        SYSSET_DAC_END = SYSSET_DAC_SPARE, // Last DAC Setting
        SYSSET_DAC_COUNT = SYSSET_DAC_END - SYSSET_DAC_BASE + 1,
        SYSSET_CCD_END = SYSSET_DAC_END,   // Last CCD Controller Setting

        SYSSET_NSETTINGS = SYSSET_CCD_BASE +
                       ((SYSSET_CCD_END - SYSSET_CCD_BASE + 1)*10),
        SYSSET_COUNT
    };
    // The above settings are based on the Rev. C release
    // of the DPA to DEA ICD (MIT 36-02205)
```

[see Config Setting: Item Id]

The bit-fields of the DEA and FEP power settings correspond to the CCD and FEP identifier codes, respectively. For example, bit 0 of the SETTING_DEA_POWER field corresponds to CCD Controller I0 (CCD_I0), bit 1 corresponds to CCD Controller I1, etc. A “1” indicates that the corresponding board is to be turned on, and “0” indicates that the board should be off.

The settings from SYSSET_CCD_BASE through SYSSET_CCD_END index the settings for CCD I0. The arrays of settings for the remaining CCDs immediately follow SYSSET_CCD_END, and are indexed in CCD Id order.

## 5.16 Software Housekeeping Statistic Codes

The following Software Housekeeping sub-codes are used to report BEP filter statistics when the "reportgrade1" patch is active.

```
enum SwFilterId {
    SW_FILTER_NONE,         /* events unfiltered */
    SW_FILTER_EVENT,        /* events filtered by energy */
    SW_FILTER_GRADE1,       /* events filtered by SW_GRADE_CODE1 */
    SW_FILTER_GRADE2,       /* events filtered by SW_GRADE_CODE2 */
    SW_FILTER_GRADE3,       /* events filtered by SW_GRADE_CODE3 */
    SW_FILTER_GRADE4,       /* events filtered by SW_GRADE_CODE4 */
    SW_FILTER_GRADE5,       /* events filtered by SW_GRADE_CODE5 */
    SW_FILTER_OTHER,        /* events filtered by other grade */
    SW_FILTER_WIN,          /* events filtered by window */
    SW_FILTER_COUNT
};

#define SW_FILTER_SIZE      (FEP_COUNT*SW_FILTER_COUNT)
```

The following grade codes are reported in software housekeeping when the "reportgrade1" patch is active.

```
enum SwSpecialGrade {
    SW_GRADE_CODE1  = 24,
    SW_GRADE_CODE2  = 66,
    SW_GRADE_CODE3  = 107,
    SW_GRADE_CODE4  = 214,
    SW_GRADE_CODE5  = 255
};
```

The following list the currently defined Software Housekeeping codes. These codes are provided in “acis_h/interface.h”

```
enum SwStatistic
{
   SWSTAT_VERSION,            /* ACIS Software Version Number */

   SWSTAT_SWHOUSE_RANGE,      /* Housekeeping report beyond end of list */
   SWSTAT_SWHOUSE_SKIPPED,    /* Dropped Software Housekeeping Statistic */

   SWSTAT_TIMERCB_INVOKE,     /* Timer Interrupt Callback invocations */

   SWSTAT_FEPLOCK_TIMEOUT,    /* Fep Wait: timed out */
   SWSTAT_FEPLOCK_POWEROFF,   /* Fep Wait: no power */
   SWSTAT_FEPLOCK_RESET,      /* Fep Wait: is reset */
   SWSTAT_FEPLOCK_NOIO,       /* Fep Wait: No mailbox/ringbuffer */

   SWSTAT_FEPREPLY_TIMEOUT,   /* Fep Reply: timed out */
   SWSTAT_FEPREPLY_POWEROFF,  /* Fep Reply: no power */
   SWSTAT_FEPREPLY_RESET,     /* Fep Reply: is reset */
   SWSTAT_FEPREPLY_NOIO,      /* Fep Reply: No mailbox/ringbuffer */

   SWSTAT_SCI_STOPRUN,        /* Science Run Stop Invoked */
   SWSTAT_SCI_STOPRUN_IDLE,   /* Stopped when already idle */
   SWSTAT_SCI_STOPRUN_RSTOP,  /* Stop Request issued to mode */

   SWSTAT_SCI_STARTRUN,       /* Science Run Start Invoked */
   SWSTAT_SCI_STARTRUN_BUSY,  /* Start when not idle */
   SWSTAT_SCI_STARTRUN_RUNNING,/* Start aborted previous run */
   SWSTAT_SCI_STARTRUN_RSTOP, /* Start requested stop to prv. mode */

   SWSTAT_SCI_EXPSTART_ZERO_EXPNUM,/* Exposure Number from FEP is 0 */
   SWSTAT_SCI_EXPEND_EXPNUM,  /* Ending Exposure Number did not match cur */
   SWSTAT_SCI_EXPSTART_NOEND, /* Prev. Exposure missing end. */

   SWSTAT_INTR_FEPBUS,        /* FEP Bus Error Timeout [Arg: Bad Vaddr] */

   SWSTAT_TE_SHORT_DUMP_TLM,  /* Timed Exposure Dump tlm pkt too small */
   SWSTAT_2D_SHORT_DUMP_TLM,  /* TE 2-D Windows Dump tlm pkt too small */
   SWSTAT_TE_BAD_FEP_MODE,    /* Unrecognized Timed Exp. FEP Mode */
   SWSTAT_TE_BAD_BEP_MODE,    /* Unrecognized Timed Exp. BEP Mode */

   SWSTAT_CCD_NULL_SETTING,   /* CCD Setting Ptr NULL [Arg: setting id] */

   SWSTAT_CMDECHO_NULL,       /* Cmd Echo passed NULL Pkt ptr */
   SWSTAT_CMDECHO_MISMATCH,   /* Cmd Echo pkt != curpkt [Arg: curpkt] */
   SWSTAT_CMDECHO_BADLENGTH,  /* Cmd Echo cmd too long [Arg: cmd data cnt] */
```

```
SWSTAT_CMDECHO_TRUNCATE,   /* Cmd Echo truncated [Arg: cmd data cnt] */
SWSTAT_CMDECHO_DROPPED,    /* Cmd Echo Dropped [Arg: cmd pkt id] */

SWSTAT_CMDMAN_INVALID,     /* Invalid Cmd Pkt [Arg: First word read] */
SWSTAT_CMDMAN_ERRCALLED,   /* # calls to CmdMan::handleError() */
SWSTAT_CMDMAN_ERRRETRY,    /* # retries in CmdMan::handleError() */
SWSTAT_CMDMAN_HANDLED,     /* # calls to CmdMan::handleCommand() */
SWSTAT_CMDMAN_BADLENGTH,   /* CmdMan Bad Pkt Length [Arg: Cmd Length] */

SWSTAT_DEAMAN_PRAMWRITE,   /* Bad PRAM Write Address [Arg: PRAM Index] */
SWSTAT_DEAMAN_PRAMREAD,    /* Bad PRAM Read Address [Arg: PRAM Index] */
SWSTAT_DEAMAN_SRAMWRITE,   /* Bad SRAM Write Address [Arg: SRAM Index] */
SWSTAT_DEAMAN_SRAMREAD,    /* Bad SRAM Read Address [Arg: SRAM Index] */
SWSTAT_DEAMAN_BADCNTLREG,  /* Bad DEA Cntl Reg [Arg: Reg Index] */

SWSTAT_PHHIST_BADQUAD,     /* PH Histogram Bad Quad Mode [Arg: mode] */
SWSTAT_PIX1X3_CORRUPTROW,  /* Pixel1x3 Bad Row [Arg: row] */
SWSTAT_PIX1X3_CORRUPTCOL,  /* Pixel1x3 Bad Column [Arg: col] */
SWSTAT_PMTEHIST_BADQUAD,   /* PM Te Hist Bad Quad Mode [Arg: mode] */

SWSTAT_PIX3X3_CORRUPTROW,  /* Pixel3x3 Bad Row [Arg: row] */
SWSTAT_PIX3X3_CORRUPTCOL,  /* Pixel3x3 Bad Column [Arg: col] */

SWSTAT_FEPMAN_RINGRDINDX,  /* FepMan Corrupt Rd Index [Arg: readIndex] */
SWSTAT_FEPMAN_RINGWRINDX,  /* FepMan Corrupt Wr Index [Arg: writeIndex] */

SWSTAT_PM_BADRECTYPE,      /* ProcessMode Bad Record Type [Arg: type] */

SWSTAT_DEACCD_LOADINVALID, /* CCD Cntl Start on invalid load [Arg: 0] */
SWSTAT_DEABOARD_ERROR,     /* DEA Error [Arg: (slot << 16) | errcode] */

SWSTAT_FEPCMD_MBOXSTATE,   /* FEP Mailbox not empty [Arg: mbox state] */

SWSTAT_FEP_READMEM,        /* FEP Read Memory Called [Arg: fepid] */
SWSTAT_FEP_WRITEMEM,       /* FEP Write Memory Called [Arg: fepid] */
SWSTAT_FEP_EXECMEM,        /* FEP Execute Memory Called [Arg: fepid] */
SWSTAT_FEP_STARTBIAS,      /* FEP Start Bias [Arg: none] */
SWSTAT_FEP_STOP,           /* FEP Stop Issued [Arg: abortFlag] */
SWSTAT_FEP_STARTDATA,      /* FEP Start Data Process [Arg: requestType] */
SWSTAT_FEP_QUERY,          /* FEP Query [Arg: fepid] */

SWSTAT_SMPROC_RSTOP,       /* Sci Mode Data Proc Stop Rqst [Arg: none] */
SWSTAT_SMWAITBIAS_ABORT,   /* Sci Mode Bias Wait Abort [Arg: caught] */
SWSTAT_SMWAITEVENT_CAUGHT, /* Sci Mode Event Wait Signal [Arg: caught] */
SWSTAT_SMWAITEVENT_ABORT,  /* Sci Mode Event Wait Abort [Arg: caught] */
SWSTAT_SMRABORT,           /* Sci Mode Request Abort [Arg: reason] */

SWSTAT_SCI_DUMPFAILED,     /* Sci Manager Dump Failed [Arg: none] */
SWSTAT_SCI_SETUPFAILED,    /* Sci Manager Setup Failed [Arg: none] */
SWSTAT_SCI_DEADUMPFAILED,  /* Sci Manager DEA Dump Failed [Arg: none] */
SWSTAT_SCI_DEACHECKFAILED, /* Sci Manager DEA Check Failed [Arg: none] */
SWSTAT_SCI_BIASFAILED,     /* Sci Manager Bias Failed [Arg: none] */
SWSTAT_SCI_DATACOMPLETE,   /* Sci Manager Data Run Complete [Arg: none] */
SWSTAT_SCI_BIASCOMPLETE,   /* Sci Manager Bias Run Complete [Arg: none] */

SWSTAT_SCI_INHIBIT_ON,     /* Sci Man Inhibit On [Arg: prv state] */
SWSTAT_SCI_INHIBIT_OFF,    /* Sci Man Inhibit Off [Arg: prv state] */

SWSTAT_FEPMAN_POWERON,     /* FEP Manager Power On [Arg: fepid] */
```

```
SWSTAT_FEPMAN_POWEROFF,    /* FEP Manager Power Off [Arg: fepid] */
SWSTAT_FEPMAN_STARTLOAD,   /* FEP Manager Start Prog. Load [Arg: fepid] */
SWSTAT_FEPMAN_ENDLOAD,     /* FEP Manager End Prog. Load [Arg: fepid] */
SWSTAT_DEACCD_POWERON,     /* DEA Ccd Cntl Power On [Arg: board] */
SWSTAT_DEACCD_POWEROFF,    /* DEA Ccd Cntl Power Off [Arg: board] */

SWSTAT_SCI_EXPSTART_FEPTIME,/* FEP Timestamp corrupted [Arg: fepTime] */

SWSTAT_FEPREPLY_BADTYPE,   /* FEP Reply Bad Type [Arg: fepid] */
SWSTAT_FEPREC_POWEROFF,    /* FEP Read Record No Power [Arg: fepid] */
SWSTAT_FEPREC_RESET,       /* FEP Read Record Reset [Arg: fepid] */
SWSTAT_FEPCFG_NACK,        /* FEP Config Nack [Arg: fepid] */
SWSTAT_FEPDIST_NACK,       /* FEP Distribute Cmd Nack [Arg: fepid] */

SWSTAT_SYSCFG_IN_CLIP,     /* SysCfg clipped stored item [Arg: item] */

 SWSTAT_SCI_JITTERFAILED,  /* Sci Man Jitter DAC operation failed */
 SWSTAT_SMWAITTRICKLE_ABORT, /* Sci Mode Bias Trickle Abort [Arg:caught]*/

SWSTAT_FILT_FEP_0_NONE,    /* FEP_0 events unfiltered [Arg: expNum] */
SWSTAT_FILT_BASE = SWSTAT_FILT_FEP_0_NONE, /* Sci Mode Event Stats */
SWSTAT_FILT_FEP_0_EVENT,   /* FEP_0 events filtered by energy */
SWSTAT_FILT_FEP_0_GRADE1,  /* FEP_0 events filtered by SW_GRADE_CODE1 */
SWSTAT_FILR_FEP_0_GRADE2,  /* FEP_0 events filtered by SW_GRADE_CODE2 */
SWSTAT_FILT_FEP_0_GRADE3,  /* FEP_0 events filtered by SW_GRADE_CODE3 */
SWSTAT_FILT_FEP_0_GRADE4,  /* FEP_0 events filtered by SW_GRADE_CODE4 */
SWSTAT_FILT_FEP_0_GRADE5,  /* FEP_0 events filtered by SW_GRADE_CODE5 */
SWSTAT_FILT_FEP_0_OTHER,   /* FEP_0 events filtered by other grade */
SWSTAT_FILT_FEP_0_WIN,     /* FEP_0 events filtered by window */

SWSTAT_FILT_FEP_1_NONE,    /* FEP_1 events unfiltered [Arg: expNum] */
SWSTAT_FILT_FEP_1_EVENT,   /* FEP_1 events filtered by energy */
SWSTAT_FILT_FEP_1_GRADE1,  /* FEP_1 events filtered by SW_GRADE_CODE1 */
SWSTAT_FILT_FEP_1_GRADE2,  /* FEP_1 events filtered by SW_GRADE_CODE2 */
SWSTAT_FILT_FEP_1_GRADE3,  /* FEP_1 events filtered by SW_GRADE_CODE3 */
SWSTAT_FILT_FEP_1_GRADE4,  /* FEP_1 events filtered by SW_GRADE_CODE4 */
SWSTAT_FILT_FEP_1_GRADE5,  /* FEP_1 events filtered by SW_GRADE_CODE5 */
SWSTAT_FILT_FEP_1_OTHER,   /* FEP_1 events filtered by other grade */
SWSTAT_FILT_FEP_1_WIN,     /* FEP_1 events filtered by window */

SWSTAT_FILT_FEP_2_NONE,    /* FEP_2 events unfiltered [Arg: expNum] */
SWSTAT_FILT_FEP_2_EVENT,   /* FEP_2 events filtered by energy */
SWSTAT_FILT_FEP_2_GRADE1,  /* FEP_2 events filtered by SW_GRADE_CODE1 */
SWSTAT_FILT_FEP_2_GRADE2,  /* FEP_2 events filtered by SW_GRADE_CODE2 */
SWSTAT_FILT_FEP_2_GRADE3,  /* FEP_2 events filtered by SW_GRADE_CODE3 */
SWSTAT_FILT_FEP_2_GRADE4,  /* FEP_2 events filtered by SW_GRADE_CODE4 */
SWSTAT_FILT_FEP_2_GRADE5,  /* FEP_2 events filtered by SW_GRADE_CODE5 */
SWSTAT_FILT_FEP_2_OTHER,   /* FEP_2 events filtered by other grade */
SWSTAT_FILT_FEP_2_WIN,     /* FEP_2 events filtered by window */

SWSTAT_FILT_FEP_3_NONE,    /* FEP_3 events unfiltered [Arg: expNum] */
SWSTAT_FILT_FEP_3_EVENT,   /* FEP_3 events filtered by energy */
SWSTAT_FILT_FEP_3_GRADE1,  /* FEP_3 events filtered by SW_GRADE_CODE1 */
SWSTAT_FILT_FEP_3_GRADE2,  /* FEP_3 events filtered by SW_GRADE_CODE2 */
SWSTAT_FILT_FEP_3_GRADE3,  /* FEP_3 events filtered by SW_GRADE_CODE3 */
SWSTAT_FILT_FEP_3_GRADE4,  /* FEP_3 events filtered by SW_GRADE_CODE4 */
SWSTAT_FILT_FEP_3_GRADE5,  /* FEP_3 events filtered by SW_GRADE_CODE5 */
SWSTAT_FILT_FEP_3_OTHER,   /* FEP_3 events filtered by other grade */
SWSTAT_FILT_FEP_3_WIN,     /* FEP_3 events filtered by window */
```

```
    SWSTAT_FILT_FEP_4_NONE,    /* FEP_4 events unfiltered [Arg: expNum] */
    SWSTAT_FILT_FEP_4_EVENT,   /* FEP_4 events filtered by energy */
    SWSTAT_FILT_FEP_4_GRADE1,  /* FEP_4 events filtered by SW_GRADE_CODE1 */
    SWSTAT_FILT_FEP_4_GRADE2,  /* FEP_4 events filtered by SW_GRADE_CODE2 */
    SWSTAT_FILT_FEP_4_GRADE3,  /* FEP_4 events filtered by SW_GRADE_CODE3 */
    SWSTAT_FILT_FEP_4_GRADE4,  /* FEP_4 events filtered by SW_GRADE_CODE4 */
    SWSTAT_FILT_FEP_4_GRADE5,  /* FEP_4 events filtered by SW_GRADE_CODE5 */
    SWSTAT_FILT_FEP_4_OTHER,   /* FEP_4 events filtered by other grade */
    SWSTAT_FILT_FEP_4_WIN,     /* FEP_4 events filtered by window */

    SWSTAT_FILT_FEP_5_NONE,    /* FEP_5 events unfiltered [Arg: expNum] */
    SWSTAT_FILT_FEP_5_EVENT,   /* FEP_5 events filtered by energy */
    SWSTAT_FILT_FEP_5_GRADE1,  /* FEP_5 events filtered by SW_GRADE_CODE1 */
    SWSTAT_FILT_FEP_5_GRADE2,  /* FEP_5 events filtered by SW_GRADE_CODE2 */
    SWSTAT_FILT_FEP_5_GRADE3,  /* FEP_5 events filtered by SW_GRADE_CODE3 */
    SWSTAT_FILT_FEP_5_GRADE4,  /* FEP_5 events filtered by SW_GRADE_CODE4 */
    SWSTAT_FILT_FEP_5_GRADE5,  /* FEP_5 events filtered by SW_GRADE_CODE5 */
    SWSTAT_FILT_FEP_5_OTHER,   /* FEP_5 events filtered by other grade */
    SWSTAT_FILT_FEP_5_WIN,     /* FEP_5 events filtered by window */

    SWSTAT_COUNT,
    SWSTAT_LAST = SWSTAT_COUNT - 1 /* Last slot is sent but unused */
};
```

[see Sw Housekeeping: Statistics]

## 5.17 Fatal Error Codes

The following lists the currently defined fatal error codes. These codes are provided in “acis_h/interface.h”.

```
enum FatalCode
{
    FATAL_UNKNOWN = 0,         /* Unknown Fatal Error */
    FATAL_RTXERROR,            /* Nucleus RTX generated fatal error */
    FATAL_EXCEPTION,           /* Processor Exception *
    FATAL_INTERRUPT,           /* Unexpected Interrupt Cause */
    FATAL_FEPDEVICE_BAD_FEPID,/* Bad FEP Id */
    FATAL_TASK_EXIT,           /* Task returned [Arg: Task Ptr] */
    FATAL_RTX_RETURNED,        /* Nucleus RTX Returned */
    FATAL_INTR_FEP_BUS_ERROR, /* FEP Bus Error [Arg: Bad Vaddr] */

    FATAL_LAST,                /* Last slot unused */
    FATAL_COUNT
};
```

[see Fatal Message: Fatal Code, Bep Startup Message: Last Fatal Code]

## 5.18 Bias Algorithm Selection Codes

The following define the Bias Algorithm selection codes for Timed Exposure Mode and Continuous Clocking Mode. The determination of which algorithm is used and which parameters affect the computation is derived from the parameter values.

### 5.18.1 Timed Exposure

```
typedef enum {
 FEP_NO_BIAS,          /* none */
 FEP_BIAS_1,           /* algorithm #1:Whole Frame Mode */
 FEP_BIAS_2            /* algorithm #2:Strip Mode */
} fepBiasType;
```

The following table (copied from the “FEP Timed Exposure Bias Calibration” section of the Detailed Design Specification (MIT 36-53226)) indicates how the parameters from the Timed Exposure Parameter Block (see “Load Te Block” in the IP&CL Structures definitions) affect the bias computation for each mode:

**TABLE 2. Timed Exposure Bias Parameter Usage**

| **Field Name** | **“Whole-Frame” Mode** | **“Strip” Mode** |
|---|---|---|
| Bias Algorithm Id | FEP_BIAS_1 | FEP_BIAS_2 |
| Bias Arg 0 | Number of conditioning exposures (PHASE2) | Number of exposures per pixel |
| Bias Arg 1 | Number of approximation-to-mean exposures (PHASE3), including the conditioning exposures listed in Bias Arg 0. | =0 to use ***mean***<br>=1 to use ***fractile***<br>=2 to use ***medmean*** |
| Bias Arg 2 | Rejection threshold for low-pixel elimination (immediately prior to PHASE3) | For ***mean*** and ***medmean***, specifies σ rejection criterion. For ***fractile***, index of sorted pixel array. |
| Bias Arg 3 | Threshold for event rejection (PHASE3) | Specifies how many of the largest samples are to be removed from the pixel array before applying the ***mean***, ***medmean***, or ***fractile*** algorithm |
| Bias Arg 4 | Rejection threshold for approximation-to-mean | Specifies how many of the smallest samples are to be removed from the pixel array before applying the ***mean***, ***medmean***, or ***fractile*** algorithm |

[see Load Te Block: Bias Algorithm Id, Load Te Block: Bias Arg]

### 5.18.2 Continuous Clocking

The following table (copied and adjusted from the “FEP Continuous Clocking Bias Calibration” section of the Detailed Design Specification (MIT 36-53227)) indicates how the parameters from the Continuous Clocking Parameter Block (see “Load Cc Block” in the IP&CL Structures definitions) affect the bias computation

**TABLE 3. Continuous Clocking Bias Parameter Usage**

| Field Name | Description |
|---|---|
| Bias Algorithm Id | =0 to use the Iterated Mean algorithm, ***mean***<br>=1 to use the Fractile algorithm, ***fractile*** |
| Bias Rejection | For ***mean***, specifies σ rejection criterion. For ***fractile***, index of sorted pixel array. |

[see Load Cc Block: Bias Algorithm Id, Load Cc Block: Bias Rejection]

## 5.19 FEP Science Report Error Codes

The following list the FEP error codes supplied by the Science Run report. The first set of codes are defined by the FEP/BEP interface, and are supplied by the FEP in response to a command or action initiated by the BEP. The second set are defined by the BEP to report power, reset conditions, or I/O error conditions when attempting to access a FEP. All of the definitions are provided in “acis_h/interface.h”

```
typedef enum {
 FEP_CMD_NOERR= 0,            /* no errors detected */
 FEP_CMD_ERR_NO_RUN,          /* no command currently running */
 FEP_CMD_ERR_UNK_CMD,         /* unknown command type */
 FEP_CMD_ERR_PARM_LEN,        /* parameter block too long */
 FEP_CMD_ERR_PARM_TYPE,       /* unknown parameter block type */
 FEP_CMD_ERR_QUAD_CODE,       /* unknown quadrant code */
 FEP_CMD_ERR_BIAS_TYPE,       /* unknown bias type code */
 FEP_CMD_ERR_BIAS_PARM0,      /* bad bias parm 0 */
 FEP_CMD_ERR_NROWS,           /* bad number of rows */
 FEP_CMD_ERR_NCOLS,           /* bad number of columns */
 FEP_CMD_ERR_NOCLK,           /* bad number of overclocks */
 FEP_CMD_ERR_NHIST,           /* bad histogram exposure count */
 FEP_CMD_ERR_NO_PARM,         /* no parameter block loaded */
 FEP_CMD_ERR_BAD_CMD,         /* illegal secondary command */
 FEP_CMD_ERR_NO_BIAS,         /* no bias map stored */
} fepCmdRetCode;

enum FepIoErrors
{
 FEP_ERR_LOCK_TIMEOUT   = 0x80, // Timeout on FEP lock
 FEP_ERR_NO_POWER       = 0x81, // FEP has no power
 FEP_ERR_IS_RESET       = 0x82, // FEP is reset
 FEP_ERR_NO_CMDRING     = 0x83, // FEP program has no command mailbox
 FEP_ERR_REPLY_TIMEOUT  = 0x84, // FEP reply timed-out
 FEP_ERR_BAD_REPLY_TYPE = 0x85, // FEP produced bad reply
 FEP_ERR_BAD_MBOX_STATE = 0x86  // FEP mailbox state invalid
};
```

[see Science Report: Fep Error Codes]

## 5.20 FEP Science Mode Codes

The following define the FEP mode codes for Timed Exposure and Continuous Clocking Mode. These definitions are provided in “acis_h/interface.h”.

### 5.20.1 Timed Exposure

```
enum TeFepMode
{
    FEP_TE_MODE_RAW,    // Raw Mode
    FEP_TE_MODE_HIST,   // Histogram Mode
    FEP_TE_MODE_EV3x3,  // 3x3 Event Detection Mode
    FEP_TE_MODE_EV5x5,  // 5x5 Event Detection Mode
    FEP_TE_MODE_CTI1,   // 5x5 CTI Reporting Mode
    FEP_TE_MODE_CTI2,   // 3x3 CTI Reporting Mode

    FEP_TE_MODE_COUNT,
    FEP_TE_LASTMODE = FEP_TE_MODE_COUNT - 1
};
```

[see Load Te Block: Fep Mode]

### 5.20.2 Continuous Clocking

```
enum CcFepMode
{
    FEP_CC_MODE_RAW,    // Raw Mode
    FEP_CC_MODE_EV1x3,  // 1x3 Event Detection Mode
    FEP_CC_MODE_EV3x3.  // 3x3 Event Detection Mode

    FEP_CC_MODE_COUNT,
    FEP_CC_LASTMODE = FEP_CC_MODE_COUNT - 1
};
```

[see Load Cc Block: Fep Mode]

## 5.21 BEP Packing Mode Codes

The following define the BEP Event List Packing Codes for Timed Exposure and Continuous Clocking Mode. These definitions are provided in “acis_h/interface.h”.

### 5.21.1 Timed Exposure

```
enum TeBepMode
{
    BEP_TE_MODE_FAINT,         // 3x3 Faint Mode Event Telemetry
    BEP_TE_MODE_FAINTBIAS,     // 3x3 Faint with Bias Event Telemetry
    BEP_TE_MODE_GRADED,        // 3x3 Graded Event Telemetry
    BEP_TE_MODE_EVHIST,        // 3x3 Event Histogram Telemetry

    BEP_TE_MODE_COUNT,
    BEP_TE_LASTMODE = BEP_TE_MODE_COUNT - 1
};
```

[see Load Te Block: Bep Packing Mode]

### 5.21.2 Continuous Clocking

```
enum CcBepMode
{
   BEP_CC_MODE_FAINT,         // 1x3 Faint Mode Event Telemetry
   BEP_CC_MODE_GRADED,        // 1x3 Graded Event Telemetry

   BEP_CC_MODE_COUNT,
   BEP_CC_LASTMODE = BEP_CC_MODE_COUNT - 1
};
```

[see Load Cc Block: Bep Packing Mode]

## 5.22 Science Mode Termination Codes

The following list the science mode termination reason codes. These codes are provided in “acis_h/interface.h”

```
enum SmTerminationCode
{
 SMTERM_UNUSED,            /* Unused */
 SMTERM_STOPCMD,           /* Commanded to Stop i.e. normal term.*/
 SMTERM_BIASDONE,          /* Bias-only Run completed */
 SMTERM_RADMON,            /* Radiation Monitor was asserted */
 SMTERM_CLOBBERED,         /* Clobbered by another start command */

 SMTERM_FEP_BIAS_START,    /* FEP Bias Processing did not start */
 SMTERM_FEP_DATA_START,    /* FEP Data Processing did not start */
 SMTERM_CCD_BIAS_START,    /* Cmd. start clock. CCDs for bias failed */
 SMTERM_CCD_DATA_START,    /* Cmd. start clock. CCDs for data failed */
 SMTERM_CCD_BIAS_STOP,     /* Cmd. stop clock. CCDs for bias failed */

 SMTERM_PROC_PARM_INVALID, /* Processing Parameter out of range */
 SMTERM_DEA_PARM_INVALID,  /* DEA Parameter out of range */
 SMTERM_FEP_PARM_INVALID,  /* FEP Parameter out of range */
 SMTERM_FEP_CONFIG_ERROR,  /* FEP Configuration Error */

 SMTERM_DEA_IO_ERROR,      /* I/O errors, or no CCD controllers on */
 SMTERM_FEP_IO_ERROR,      /* I/O errors, or no FEPs are on */

 SMTERM_UNSPECIFIED,       /* Reason is unspecified */
};
```

[see Science Report: Termination Code]

## 5.23 Miscellaneous FEP Constants

The following list some miscellaneous constants and limits used by the Front End Processor software:

```
#define BIAS_BAD    (0xffe)/* bias parity error value */
#define PIXEL_BAD   (0xfff)/* pixel in bad pixel map */
#define MAX_NOCLK   (30)     /* maximum overclocks per output node */
#define MAX_NOCLKR  (16)    /* max raw overclocks per output node */
#define MAX_NROWS   (1024)  /* maximum number of pixel rows */
#define MAX_NCOLS   (1024)  /* maximum number of pixel columns */
#define MAX_STRIPS  (64)   /* maximum bparm[0] in strip mode */
```

```
#define CCLK_NROWS  (512)   /* number of pixel rows in CClk mode */
#define PIXEL_MASK   (0xfff)/* valid pixel and bias bits */
#define INITSKIP     2        /* number of science exposures to skip */
```

## 5.24 Miscellaneous BEP Constants

The following list some miscellaneous constants and limits used by the Back End Processor software:

```
/* ------------------------------------------------------------------------
 * BEP Graded Mode Special Codes
 * ------------------------------------------------------------------------ */

enum GradedMeanCode
{
    CORNER_MEAN_LOW = -4096,  /* Graded Mode Corner Mean below -4095 */
    CORNER_MEAN_MISSING = 4095/* Graded Mode No Valid Corner Pixels */
};

[see Event Te Graded:Corner Mean]
```

## 5.25 Bias Parity Errors in Te5x5 Mode

A feature in the implementation of Timed Exposure 5x5 mode within the FEP can generate bogus bias parity errors. With the *digestbiaserror* patch installed, these can be readily identified within *patchDataBiasError* packets from the following unique combination of field values:

```
biasErrors = {
    row              = 0
    column           = 1022
    evenPixelFlags   = 15
    oddPixelFlags    = 0
}
```

All *patchDataBiasError* elements with this combination of values should be discarded.

# 6.0 Miscellaneous Notes

## 6.1 Telemetry Fill Pattern

When the ACIS Back End Processor (BEP) has no telemetry packets to send when queried by the spacecraft for data, the BEP hardware places a constant byte value into the telemetry stream. This value is **0xb7** (hexadecimal).

When the instrument is not powered, the telemetry hardware logic tends to float to a logical 1, usually producing a fill byte of **0xff** (don't count on this).

## 6.2 System Configuration Table Limits

Some of the numerically possible analog settings in the System Configuration Table (see Section 5.15 ) can cause voltages that exceed the tested limits of the CCDs to be applied to the CCDs. In order to prevent this, the instrument contains a table of maximum value settings for each System Configuration Table item. If a command is received to change a setting which exceeds its limit value, the value will be clipped to its maximum value. The following limits are applied to their corresponding settings:

**TABLE 4. System Configuration Setting Limits**

| Item Identifier | Voltage Limit | DAC Limit | Item Description |
|---|---|---|---|
| SYSSET_DAC_PIA_P, | 12.775 V | 255 | Image Array Parallel + |
| SYSSET_DAC_PIA_MP, | 12.775 V | 255 | Image Array Parallel -+ |
| SYSSET_DAC_PIA_M, | -7.025V | 140 | Image Array Parallel - |
| SYSSET_DAC_PFS_P, | 12.775 V | 255 | Framestore Parallel + |
| SYSSET_DAC_PFS_MP, | 12.775 V | 255 | Framestore Parallel -+ |
| SYSSET_DAC_PFS_M, | -7.025V | 140 | Framestore Parallel - |
| SYSSET_DAC_S_P, | 12.775 V | 255 | Serial Register + |
| SYSSET_DAC_S_M, | -7.025V | 140 | Serial Register - |
| SYSSET_DAC_R_P, | 12.775 V | 255 | Reset Gate + |
| SYSSET_DAC_R_MP, | 12.775 V | 255 | Reset Gate -+ |
| SYSSET_DAC_R_M, | -7.025V | 140 | Reset Gate - |
| SYSSET_DAC_SCP, | 12.775 V | 255 | Scupper |
| SYSSET_DAC_OG_P, | 12.775 V | 255 | Output Gate + |
| SYSSET_DAC_OG_M, | -7.025V | 140 | Output Gate - |
| SYSSET_DAC_RD, | 11.7V | 233 | Reset Diode |
| SYSSET_DAC_DR0, | 20.6V (8.9V) | 177 | Drain Output (A) |
| SYSSET_DAC_DR1, | 20.6V (8.9V) | 177 | Drain Output (B) |
| SYSSET_DAC_DR2, | 20.6V (8.9V) | 177 | Drain Output (C) |
| SYSSET_DAC_DR3, | 20.6V (8.9V) | 177 | Drain Output (D) |

# 7.0 Table Description

The ACIS software structures are described using a single table. Each column in the table describes a particular attribute or property of a software record or field belonging to a record. Each row in the table describes a field of a record. By convention, columns which describe a record are included only in the first field of the record.

## 7.1 Bit and Byte Order

Within ACIS, all data is packed least-significant bit first. For example, two 12-bit words are bit-packed as follows, where *w0* corresponds to the first word to pack, *w1* corresponds to the second, and *b0* through *b11* indicate the bits of the corresponding word (*b0* corresponds to the least-significant bit, and *b11* corresponds to the most-significant bit of the 12-bit value):

**TABLE 5. ACIS Internal Bit Order**

| Byte Location | Bit 7 (msb) | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 (lsb) |
|---|---|---|---|---|---|---|---|---|
| 0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 |
| | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
| 1 | w1 | w1 | w1 | w1 | w0 | w0 | w0 | w0 |
| | b3 | b2 | b1 | b0 | b11 | b10 | b9 | b8 |
| 2 | w1 | w1 | w1 | w1 | w1 | w1 | w1 | w1 |
| | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 |

## 7.2 Column Definitions

The column definitions are as follows:

**TABLE 6. ACIS Software IP&CL Structures Table Column Definitions**

| Column Number | Name | Description |
|---|---|---|
| A | Record Description | The first field of each record contains a description of the record in this column. |
| B | Record Name | This is the name of the record being described. This must be filled in for each field in the table. |
| C | Field Number | This is the order of the field within the record. |
| D | Subfield Number | Unused by ACIS |
| E | Owner | This field indicates the owner of the field. For ACIS, this field always contains the string “ACIS” |
| F | Field Name | This is the name of the field. |

**TABLE 6. ACIS Software IP&CL Structures Table Column Definitions**

| Column Number | Name | Description |
|---|---|---|
| G | Mnemonic | This is intended to be the 8-character mnemonic of the field. Currently, only the last two characters are meaningful. “CI” indicates that the fields belong to a command packet record, “XF” indicates that they belong to a telemetry packet record, and “IT” indicate that the fields belong to an internal structure definition, usable within both command and telemetry packet records. |
| H | Type | This indicates the data type of the field. This may contain the following:<br>bit - The field is a bit-field, whose length is determined by the dimension column (see column I). If field’s Start Range (see Column O) is negative, the value of the field is represented using two’s complement, otherwise, the field is treated as unsigned.<br>uint[8,16,32] - The field is an 8, 16, or 32 bit unsigned integer.<br>int[8,16,32] - The field is an 8, 16, or 32 bit 2’s complement signed integer<br>“record name” - The field is a structure whose’s type is defined by the referenced record name. |
| I | Dimension | This is used to define an array of elements, or to specify the number of bits in a bit-field. This may be defined as a constant number, or as a formula |
| J | Max. Dimension | Unused by ACIS |
| K | Variable | Unused by ACIS |
| L | Field Text | This describes the contents of the field |
| M | Size in Bytes | Unused by ACIS |
| N | Total Size | Unused by ACIS |
| O | Start Range | This is the lowest value that can be contained to the field. |
| P | End Range | This is the largest value that can be contained in the field. |
| Q | Units | This describes the units of the field. This is currently unspecified by ACIS. |
| R | Value | This specifies the value of the field. This field is unused by ACIS. The corresponding information is provided using the Start Range and End Range fields. |
| S | Format | Unused by ACIS |
| T | Question | Unused by ACIS |
| U | Alignment | This specifies the bit-alignment requirement of the field. For fields defined as an array or record, this specifies the alignment requirement of the first element in the array, or first field in the referenced record. |
| V | Command Timeout | This defines the minimum time to allow for a command record to be handled by ACIS. This item is specified for the first field of command records only. |

**TABLE 6. ACIS Software IP&CL Structures Table Column Definitions**

| Column Number | Name | Description |
|---|---|---|
| W | Tag | This is used by ACIS to assist in implementing the described format. It specifies the symbolic Command Opcode, or Telemetry Format Tag for command and telemetry packet records, respectively. |
| X | Allocator | This is used by ACIS to assist its code-generator. This specifies the telemetry packet buffer allocator to use for the corresponding telemetry packet record. It is only specified for telemetry packets. |
| Y | Implementation Key | This is used by ACIS to assist its code-generator. |

## 7.3 Dimension Formula Description

The dimension field in the table structure may express a constant integer value, or a function of one or more fields in the record description. The dimension formula may use the following operators:

**+** .....- Addition

**-** .....- Subtraction

***** .....- Multiplication

**DIV** .....- Integer division (truncated)

**bitoffset**.....- The bit-position of the field within the record

**bitsize(***record name or type***)**.....- The number of bits occupied by the referenced record

Typically, the ACIS table definitions use a constant integer value for the dimension, however, some dimensions are coupled to the length of the containing command packet or telemetry packet.

Since the Command Packet Length field describes the total number of 16-bit words in a command packet, command packets which contain a variable number of elements at the end of the packet usually use the following form to describe the number of elements:

**(**Command Length ***** 16 **- bitoffset) DIV bitsize(***field's data type***)**

Since the Telemetry Packet Length field describes the total number of 32-bit words in the packet. Telemetry packets which contain a variable number of elements at the end, usually use the following form:

**(**Telemetry Length ***** 32 **- bitoffset) DIV bitsize(***field's data type***)**

In order to shorten the formula text in the simplified format descriptions, the following abbreviations are used:

Command Length.....CL

Telemetry Length .....TL

bitoffset .....OFF

bitsize .....SIZE