

# CIPS On-Board Image Data Processing Design Note

## Purpose

This design note fully describes how the CIPS instrument hardware and software will process image data taken by its four CCDs. It is meant to be *the* definitive guide, used by software and hardware engineers in the CIPS design as well as instrument scientists who will be interpreting the data.

## Background

Each CCD is a square 4 megapixel device, with 2048 pixels on each side. Photons impinging on the CCD first pass through an image intensifier. The aperture on the intensifier does not quite cover the entire CCD, instead covering only an “Active Region” that is 1388 x 1516 pixels in size. The CIPS instrument will be capable of telemetering to the spacecraft bus the data from the entire CCD or any smaller region on the CCD. CCD image data is binned and compressed in a series of steps to meet the AIM data bandwidth requirements (limitations).

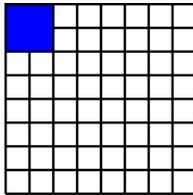
Images are synchronously taken with any (or all) of the CCDs at a periodic cadence of about 41 seconds. In its nominal operating mode, some number of pictures (about 24) will be taken with the suite of CCDs, followed by a few (roughly 8) with only one camera. Images are taken starting at the same latitude (about 80 degrees prior to reaching the pole). The transition to single camera mode (or ‘Mapping Mode’) occurs in the neighborhood of 30-50 degrees latitude, depending on the time of year.

It is important to note that each step in the binning and compression process described herein is optional and can be enabled or disabled by command. Other features that can be commanded include the image size, time delta between images, image collection start time (latitude), number of images to be taken, and the cameras to be used for image collection.

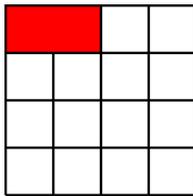
## Detail

### ***CCD To Camera FPGA***

Image data is binned 2x2 (that is, 2 rows by 2 columns) on the CCD, and the result from one binning of 4 pixels is called a ‘Super Pixel’. Data is digitized by a 14-bit ADC when it is read by the Camera FPGA. As data is being read out from the CCDs by the Camera FPGAs, column binning is performed (that is, 1x2 binning) forming an ‘Intermediate Pixel’ and adding a carry bit. Each Intermediate Pixel contains the data from 2 Super Pixels or 8 CCD pixels. Thus, in the next step of the process, 15-bit data is transferred.



**Figure 1: Sample CCD Region Showing a Super Pixel (blue)**



**Figure 2: Sample CCD Region Showing an Intermediate Pixel (red) After On-Chip CCD Binning**

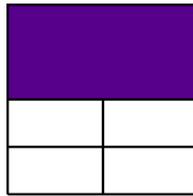
### ***Camera FPGA to Processor FPGA to SDRAM***

Pixel data is next transferred to the Processor FPGA. Packets consisting of one scanline (row) are sent by the Camera FPGA to the Processor FPGA. Each packet contains the scanline number, which the Processor FPGA uses to ensure the line is written to the correct location in SDRAM, thereby ensuring that missing or erroneous scanlines do not cause errors in the rest of the image.

Transfer error status information is stored by the Processor FPGA in SDRAM along with the scanline. Transfer errors are detected by the Processor FPGA on a per-pixel basis so that an error in the transfer of one pixel does not propagate through the rest of the scanline. Packet transfer error flags consist simply of a logical ‘OR’ of the pixel error bits; thus, data transfer errors can only be resolved to the scanline level. In addition to these error flags, the Processor FPGA also calculates and stores in SDRAM a packet checksum, which allows error checking to continue through the next phase of the image transfer.

## SDRAM to Software to Spacecraft Bus

Once in SDRAM, software running on the UTMC UT69R000 microcontroller has access to image data. For simplicity in the FPGA and software design, it has been decided that SDRAM access will be limited to one device at a time, i.e. either data is being transferred by the Camera FPGA to SDRAM, or the microcontroller is pulling data from SDRAM. This will be controlled by software (through a sequence or ground command). When the CIPS software has access to SDRAM, scanline packets are pulled in pairs from memory. Packet error flags are saved for later use, and packet checksums are verified. If errors are detected in either place, image processing will continue, and resultant science CCSDS telemetry packets will be flagged appropriately. Scanlines are pulled in pairs so that the image data may be binned 2x2. In this process, ‘Science Pixels’ are created. Each Science Pixel contained the binned contents of 4 Intermediate Pixels, 8 Super Pixels, or 32 CCD pixels. Due to the binning process, Science Pixels will require 17 bits of storage.



**Figure 3: Sample CCD Region Showing a Science Pixel (purple) after FPGA and On-Chip Binning**

Once the binning process has been completed and two large scanlines have been transformed into one smaller scanline, the scanline data is compressed using a ‘Root 2N’ lossy compression scheme. In this algorithm, each binned Science Pixel data value (N) is compressed to a new value (C) using less bits per pixel than the uncompressed data according to the following equation. Compressed values are rounded such that the resultant value, when decompressed, most closely resembles the original data value.

### Equation 1

$$C = \text{round}(\sqrt{2 * N})$$

After compression, pixel data are stored with maximum efficiency (no unused bits). These compressed scanline data are combined with the scanline packet error status flags stored earlier and are buffered in a CCSDS telemetry packet for later downlink. Each of these CCSDS packets will contain 1010 bytes of image and image status data, regardless of where this data falls with regards to scanline or image boundary. Image data will, however, be separated by camera. The only other exception to this rule is when the last image in an observation sequence has been processed. Once complete, CCSDS telemetry packets are then handed over to the Mil-Std-1553 bus controller at the next available slot in the bus schedule.

A summary of the daily image data output by CIPS is shown in Table 6 below.

## Software Compression Options

Root 2N compression allows data to be compressed in a (slightly) lossy manner. For the nominal binning scheme on CIPS, this means that data will be reduced in size from 17 to 10 bits per pixel. Since bandwidth is at a premium on the AIM mission, there are some compression options which should be considered. Some of these options involve trades between lesser bandwidth and the chance of slightly higher data loss, others trade slightly improved resolution in mid-range pixel values for slightly increased complexity in CIPS flight and data processing software.

## Nominal Root 2N Compression

In the nominal compression scheme, some optimization is done to preserve mid-range data numbers. The highest possible data number for a Science Pixel (17 bits) is:

$$N_{\max} = 2^{17} - 1 = 131071$$

For this value, the resultant value for C using Equation 1 is 512. Thus, for all values of N, C will range from 0 to 512. Since C, as a 10-bit number, can have a value up to 1023 ( $C_{\max}$ ), it makes sense to introduce an offset equal to the difference between  $C_{\max}$  and the largest possible calculated value of C. Then, Equation 1 becomes

### Equation 2

$$C = f(N) = \text{round}(\sqrt{2 * N}) + \delta + 1$$

where

### Equations 3

$$\alpha = f(N_{\max})$$

$$\delta = C_{\max} - \alpha - 1$$

Now, for all values of N, C will range from  $\delta$  to  $C_{\max}$ , and it becomes possible to avoid compressing values of N that range from 0 to  $\delta$ . Thus, the compression algorithm uses Equations 3 along with Equation 4 below.

### Equation 4

$$C = f(N) = \begin{cases} \text{round}(\sqrt{2 * N}) + \delta + 1 & \text{for } N > \delta \\ N & \text{for } N \leq \delta \end{cases}$$

Decompression, then, simply consists of the inverse of Equation 4, again using  $\delta$  from Equations 3 as the discontinuity point.

### Equation 5

$$U = f^{-1}(C) = \begin{cases} \frac{(C - \delta - 1)^2}{2} & \text{for } C > \delta \\ C & \text{for } C \leq \delta \end{cases}$$

where U represents the decompressed value (and is not necessarily equal to N due to the lossy compression).

For a 17-bit N, the parameters for this compression scheme are as shown in the table below.

**Table 1: Nominal Root 2N Compression Parameters for 17-bit N**

Parameter	$N_{\max}$	$C_{\max}$	$\alpha$	$\delta$
Value	131071	1023	512	510

## Optimized Root 2N Compression

This algorithm improves slightly upon the Nominal scheme just presented in order to increase the size of the bottom-range region in which data numbers are not compressed. This stems from the fact that in the Nominal algorithm, there exist some values of C that cannot be obtained. For example, the value of C jumps from 510 for  $N = \delta = 510$  to 543 when  $N = \delta + 1$ . It follows that there exists some value,  $\gamma$ , that can serve as a better discontinuity point than  $\delta$ . The value of  $\gamma$  can be shown to be related to the root of a polynomial and is fixed for a given number of bits in N.<sup>1</sup>

Thus, repeating Equation 4 and Equation 5 with the new cutoff values, the following functions are defined and used by the appropriate software element when compressing and decompressing image data:

### Equation 6

$$C = f(N) = \begin{cases} \text{round}(\sqrt{2 * N}) + \delta + 1 & \text{for } N > \gamma \\ N & \text{for } N \leq \gamma \end{cases}$$

<sup>1</sup> Left as an exercise for the reader. The author decided not to try to tackle this problem and instead found  $\gamma$  by trial and error.

**Equation 7**

$$U = f^{-1}(C) = \begin{cases} \frac{(C - \delta - 1)^2}{2} & \text{for } C > \gamma \\ C & \text{for } C \leq \gamma \end{cases}$$

and the parameters used in this scheme are shown in the following table.

**Table 2: Optimized Root 2N Compression Parameters for 17-bit N**

Parameter	$N_{\max}$	$C_{\max}$	$\alpha$	$\delta$	$\gamma$
Value	131071	1023	512	510	543

### High End Dropoff Root 2N Compression

This compression scheme allows 17 bits of Science Pixel data to be reduced in size to 9 bits instead of the nominal 10. This is done by setting a cut off value that Science Pixel values cannot exceed. If C in Equation 1 were a 9 bit number, then its maximum value would be 511. Thus, the maximum value of N that still produces a 9 bit compressed number is:

$$N_{\max} = \frac{(511.5)^2}{2} = 130816$$

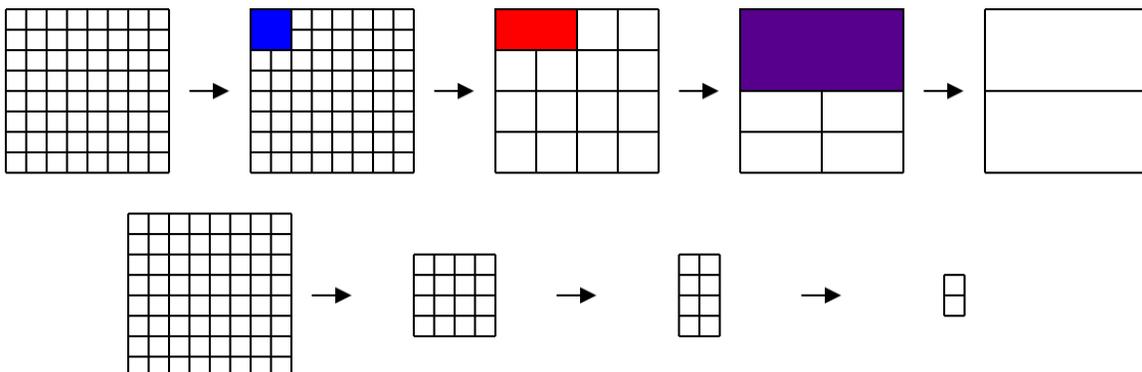
This means that, for a 17 bit N (maximum value: 131071), the highest 255 data numbers (0.195%) cannot be reduced to 9 bits and will result in an error in on-board processing (resultant C would be set to 511 or 0?). This would probably be acceptable since we know that this means that the average pixel for the 32 CCD pixels represented by this Science Pixel was 99.8% saturated. Note that in this scheme, there is no additional optimization that can take place since there is no bottom-range region in which data numbers (N) do not have to be compressed. If it is desirable that bottom-range numbers not be compressed to avoid compression losses, then a lower number for  $N_{\max}$  can be selected. However, it takes a significant decrease in  $N_{\max}$  to raise the value of  $\gamma$  appreciably (i.e.  $N_{\max}$  of 90111, which cuts off the top 31.25% of the full range of N, generates a  $\gamma$  of only 99).

### On-Board Compression Determination

This option consists of adding logic to flight software that varies the compression scheme employed depending on the binning being performed. If any of the binning steps are omitted, the CCD image data no longer occupies 17 bits. A compression scheme from 17 bits to 10 bits can be abandoned for a 16->9 (or 15->9 or 14->8) scheme (this simply changes the values of  $N_{\max}$ ,  $C_{\max}$ ,  $\alpha$ ,  $\delta$ , and  $\gamma$ ). The compression scheme used to obtain final (downlinked) science data would be indicated in each CCSDS telemetry packet.

### Summary

A pictorial summary of the binning done on CCD pixels to produce Science Pixels is shown in the image below. In this Image, the top row shows a region of CCD pixels being binned 2x2 on-chip (blue), 2x1 in the Camera FPGA (red), then 2x2 in software (purple). The top row is meant to show the coverage in terms of CCD pixels per Science Pixel (32 CCD pixels per Science Pixel), as well as at each step along the transformation. The second row in this figure shows how a region of CCD pixels is reduced in size during this transformation.



**Figure 4: CCD Image Binning Progression (Summary, fixed pixel size shown in lower row)**

The table below helps summarize the on-board data processing.

**Table 3: Binning / Compression Effects on Image Size**

Element	CCD	Camera FPGA	Software Binning	Software Compression
Binning / Compression	2x2	1x2	2x2	Root(2N)
Maximum Post-Binning Size (pixels)	1024x1024	1024x512	512x256	512x256
Projected Post Binning Size (pixels)	694x758	694x379	347x190	347x190
Post-Binning Bits Per Pixel	14	15	17	9*

Finally, all of this discussion would be moot without considering the final output of the data processing scheme, which is in the form of CCSDS packets. The reason this is critically important is because that is where we have to meet requirements (1.1 Gbit / day max). Rather than try to put all of this information into words, given the above summary, the calculations made within the 'CIPS Telemetry Packets' worksheet should be sufficient to show the effect of these various transformations on the output data rate. The important information contained within this worksheet is summarized below.

The following table shows the header information for all CIPS science telemetry packets. Note that each camera has a unique APID in the CCSDS primary header.

**Table 4: CCSDS Science Packet Header**

Item Name	Size (bits)	Offset (bits)
CCSDS Primary Header	48	0
CCSDS Secondary Header (time)	64	48
Image Exposure Time (seconds)	32	112
Image Exposure Time (sub-sec)	32	144
Start Pixel	24	176
Bits / Pixel	8	200
Image Columns	16	208
Image Rows	16	224
Compression Scheme	5	240
Software Binning Performed	1	245
Camera Binning Performed	1	246
Camera Mode (TDI/Snapshot)	1	247
Exposure Time	16	248
TDI Row Count	16	264
Packet Version	8	280
CRC	16	288
<b>Total Size</b>	<b>304</b>	

Within each packet, there will be multiple scanlines or portions of scanlines, depending on where scanlines fall relative to packet boundaries. Each scanline will be preceded by a scanline header containing the FPGA transfer error flags and checksum status flags. This format is shown in the table below.

**Table 5: Scanline Header**

Item Name	Size (bits)	Offset (bits)
Transfer Error Flags	5	0
Checksum Error Flags	3	5
<b>Total Size</b>	<b>8</b>	

On the AIM spacecraft, the maximum size of CCSDS telemetry packets is 1024 bytes, of which 1000 bytes are available for data (14 bytes of CCSDS headers are used). Thus, after the Science Packet Header, 976 bytes of image data can be downlinked in a single packet. This is enough room for 4.15 scanlines 189 pixels in length at 10 bit resolution (4.61 at 9 bit resolution). Assuming full binning and compression, the following numbers can be derived for CIPS bandwidth:

**Table 6: Daily Image Data Comparison**

<b>Bit Resolution</b>	<b>10</b>	<b>9</b>
Scanlines / Image (pixels)	189	
Columns / Image (pixels)	347	
Scanline Header Size (bits)	8	
Compressed Image Size (Kbits)	642	578
Mapping Mode Images / Observation	8	
Full Camera Suite Images / Observation	26	
Total Image Data / Observation (Mbits)	70	63
Scanlines / Packet	4	4
Data / Packet (bits)	7560	6804
Packets / Observation	9739	8767
Packet Header Size (bits)	304	
Packet Header Overhead (Mbits)	2.824	2.542
Observations / Day	15	
<b>Total Image Data / Day (Gbits)</b>	<b>1.070</b>	<b>0.963</b>

It is important to note that there are CIPS housekeeping data that also need to be accounted for. The current estimate for CIPS housekeeping data is 20.4 Mbits (0.0199 Gbits) per day.