Characterizing ExOPlanet Satellite (CHEOPS)

CHEOPS-UBE-INST-TN-109

Issue 1 Revision 0

Recipe for the implementation of the non-linearity correction in the IFSW

Mandatory Cover Page Attributes:

Document Title:	Recipe for the implementation of the non-linearity correction in the IFSW
Origin Name:	Physikalisches Institut, Universität Bern, Andrea Fortier
Date of Issue:	22. Nov. 2018
WBS code:	N/A
Package code:	N/A

CHEOPS Cover Page Attributes:

Old document reference:	
Restrictions:	None
(additional attributes)	

Distribution List

Public	[ ]
CHEOPS (Through Project Manager)	[X]
Industry: specify in detail (Company Name, point of contact)	[ ]

Approval Sheet

	Name	Signature	Date
Prepared by:	Andrea Fortier		
Verified by:	*Roland Ottensamer* *Adrien Deline*		
Approved by:			
Approved by:			

Document Change Record

Issue	Rev.	Date	Pages affected	Modification	DCR	Initials
1	0	22/11/2018	all	(new document)	N/A	AFO

DCR = document change request number

Table of Contents

List of documents

Applicable documents:	
AD1	

Reference documents:	
RD1	**CHEOPS-UBE-MA-LI-005-i0.1_List of Abbreviations**
RD2	**CHEOPS-UGE-SYS-PR-019, i3.2; Data reduction procedures for the on-ground payload calibration**

1 Introduction

1.1 Scope of the Document

This document aims at providing a guideline for the implementation of the CCD non-linearity correction (NLC) in the IFSW.

Bear in mind that on-board resources are limited and therefore, the possibilities are much more restricted from what could be achieved with an implementation on the ground.

The need for NLC on-board was suggested due to image stacking. Unstacked images will not suffer from any manipulation on board, and will be sent to the ground untouched. However, for short exposure times, window images will be stacked. It is not decided yet if in this case, to preserve a high photometric accuracy, images have to be corrected for non-linearity deviations before performing the stacking. While a decision on this point will be taken after the In-Orbit Commissioning, the IFSW has to be already prepared to perform the non-linearity correction in case its use is requested. The steps to be followed by the IFSW to perform this correction are explained in this document.

1.2 Abbreviations and Acronyms

For general abbreviation, refer to CHEOPS' List of Abbreviations (RD01).

Bias	Bias correction
G	Gain
IFSW	In Flight SW
NL	Non-Linearity
NLC	Non-Linearity Correction
T_{CCD}	CCD Temperature (one of the HK parameters, present also in images' header)
VSS	Bias voltage VSS (one of the HK parameters, present also in images' header)
VOD	Bias voltage VOD (one of the HK parameters, present also in images' header)
VRD	Bias voltage VRD (one of the HK parameters, present also in images' header)
VOG	Bias voltage VOG (one of the HK parameters, present also in images' header)
x	Pixel value in e-
x'	Pixel value in e- after NLC
y	Pixel value in ADU
y'	Pixel value in ADU

2 Non-linearity correction

2.1 In theory, what are the steps to perform the NLC?

Theoretically, without any boundary conditions concerning computational resources or SW architecture, the steps to correct the images for non-linearity are:

1. **Determine the bias correction $Bias$ that should be applied to the image**
2. **Calculate the gain G.**

   The gain depends on the CCD temperature and on the four bias voltages:

   $$G = G(T_{CCD}, VSS, VOD, VRD, VOG)$$

   Equation 1

   G is a fourth-degree polynomial that depends on the bias voltages and the temperature of the CCD (this last dependence is linear). The polynomial can be found in RD2, tables 4-3 and 4-4 (nominal and redundant channel). **Note that the polynomial function form varies depending on the electronic channel for read-out. Therefore, the gain calculation depends on the channel used for the observation.**
3. **Convert pixel values from ADU to e-:**

   Let's y be the pixel value in ADU and x is the pixel value in e-. The relationship between them is given by:

   $$x = \frac{y - Bias}{G}$$

   Equation 2
4. **Proceed to do the non-linearity correction:**

   The non-linearity correction function was derived after the calibration of the instrument (see RD2 for details).

   The proposed functional form is:

   $$x' = NLC(x) = Ax^2 + Bx + C$$

   Equation 3

   where x' is the pixel value in e- after the non-linearity correction.

   The NLC is represented by a quadratic spline with 9 intervals. In each interval the NLC has the form of a is a second-degree polynomial, as shown in Eq.3:

   $$x' = A_m x^2 + B_m x + C_m,$$

where the coefficients A_m, B_m, C_m depend on the number of electrons in the pixel ($m = 1, \dots, 10$). These coefficients can be related to the ones listed in Tables 4-5 and 4-6 in RD2 in the following way (tables are also in the annex):

$$A_m = a_m$$

$$B_m = b_m - 2a_m k_m$$

$$C_m = c_m - b_m k_m + a_m k_m^2$$

Equation 4

Note that k_m refers to knot$_m$ in table 4-5.

After this step, all pixels in the image are corrected for non-linearity and expressed in e-.

2.2 How is the NLC implemented on-board?

To perform the NLC on board we have to bear in mind that:

- The information available to estimate parameters of one image can only come from information within the image. This means that past and future information related to other images is not available.
- The number of operations should be minimized.
- Images should be delivered to the ground in ADU.

I. **Determine the bias correction $Bias$ that should be applied to the image**

   The $Bias$ to correct the image will be estimated as the median of all pixels contained in the side margin LOS. The estimated bias used for the correction will be stored to be delivered to ground. Therefore, the compression entity will have the computed bias for each image.

II. **Determine the gain correction G that should be applied to the image**

   In the header of each image the information to calculate the gain is stored ($T_{CCD}, VSS, VOD, VRD, VOG$). This will be used to calculate the gain corresponding to the image as expressed in Eq. 1 and Tables 4-3 and 4-4 (RD2). Image headers are all sent to ground, i.e. all headers are part of the compression entity.

III. **Define each interval of the quadratic spline for the NLC (that is originally given in e-) in ADUs:**

   $$(k'_m, k'_{m+1})\ in\ ADU = (k_m G + Bias, k_{m+1} G + Bias)$$

   Equation 5

This should be done to be able to determine, for each pixel (in ADU) the interval $(m, m+1)$ from where the parameters A_m, B_m ,C_m (and therefore also the parameters k_m, a_m, b_m, c_m) should be taken for the spline interpolation.

Note that the definition of the splinter intervals in ADU was done on ground (because of the way the allocated variables in the IFSW were defined) and for that the nominal gain was used to calculate each k_m. This should not pose a big problem, as only the value of the knots are affected. Due to the continuity and smoothness of the splines, the error made by calculating the knots with a fixed value for the gain should be negligible.

IV. Calculate the 10 sets of three parameters A_m, B_m ,C_m using Eqs. 4 and tables 4-5 and 4-6 of RD-01.

V. Correct for bias, gain and NL in one step (input in ADU, output in e-).

Here the idea is to minimize the number of operations to be performed in the NLC procedure.

Then:

a- Given the number of counts in the pixel (in ADU) select the interval m using Eq. 5 and the corresponding parameters A_m, B_m ,C_m for the spline interpolation.

b- Calculate A'_m ,B'_m ,C'_m as follows:

$$
\begin{aligned}
A'_m &= \frac{A_m}{G^2} \\
B'_m &= \frac{B_m}{G} - 2\frac{A_m Bias}{G^2} \\
C'_m &= C_m - \frac{B_m Bias}{G} + \frac{A_m Bias^2}{G^2}
\end{aligned}
$$

Equation 6

c- Perform the NLC on the pixel:

$$x' = A'_m y^2 + B'_m y + C'_m$$

Equation 7

➔ The result is the value of the pixel in e-

VI. Return to ADU

The image should be provided in ADU to ground. Then after doing the NLC we need to go back to ADU. This means that:

$$y' = x' G + Bias$$

Equation 8

for each pixel.

2.2.1 Discussion about converting back images to ADU after performing the NLC

After the NLC (step V.) we end up with an image in e-. This image needs to be converted back to ADU before stacking (Eq. 8). The problem is that, if we use the same gain and bias as the ones determined in steps I and II, these values are only valid for the image we are correcting. Meaning that the following image will be corrected for NL using slightly different values for the gain and bias. Therefore, all images that are stacked will have been corrected and transformed back to ADU individually, using for each of them a slightly different value for the gain and bias. When the stack image arrives to ground, there is no way to "undo" this process. The gain and bias correction will be done using only one value and therefore the result of the NLC (that is, the image in e-) will not be recovered.

One way to avoid this is the following. After doing the NLC for each image, the transformation to ADU can be done using a fix value for the gain and a fix value for the bias (e.g. the values obtained during the calibration campaign). This fix values will be the same for all the images that where corrected by non-linearity (independently of the visit). As this value will be known by the ground, they can recover exactly the vital information (i.e. NL corrected image in e-) by subtracting the bias and dividing by the gain.

Therefore, the procedure on board to correct for NL and output an image in ADU will be the following:

Given a value for the gain, G_0, and a value for the bias, $Bias_0$, the result of Eq. 7 (in e-) can be converted back to ADU using Eq. 8:

$$\begin{aligned}
y' &= x' G_0 + Bias_0 = (A'_m y^2 + B'_m y + C'_m) G_0 + Bias_0 = \\
&= (A'_m G_0) y^2 + (B'_m G_0) y + (C'_m G_0 + Bias_0) = \\
&= A''_m y^2 + B''_m y + C''_m
\end{aligned}$$

Equation 9

where:

$$A''_m = A'_m G_0 = A_m \frac{G_0}{G^2}$$

$$B''_m = B'_m G_0 = B_m \frac{G_0}{G} - 2 A_m \frac{Bias G_0}{G^2}$$

$$C''_m = C'_m G_0 + Bias_0 = C_m G_0 + Bias_0 - B_m \frac{Bias G_0}{G} + A_m \frac{Bias^2 G_0}{G^2}$$

Equation 10

Therefore, now Equation 9 replaces Eq. 7 and 8, providing the result of the NLC in ADU.

As mentioned before, the conversion back to ADU is done using a fix value for the gain and the bias. This means that when n images are stacked into one, the ADU values of each pixel relates to the electron values as follows:

$$y'_{stack} = (x'_1 + \cdots + x'_n)G_0 + nBias_0$$

Equation 11

This operation is fully reversible on ground, and the result of the NLC, i.e. the pixel information in e- $(x'_1 + \cdots + x'_n)$ can be recovered using the inverse of Eq. 11. Moreover, the headers of each individual image will be sent to the ground, so the actual value of the gain that was used to do the non-linearity correction will be known.

3 Annex 1: Where do Eqs. 6 and 7 come from?

We will derive a polynomial that using the bias and gain values estimated in I and II, will perform the NLC.

Introducing Eq. 2 in Eq. 3 we get:

$$x' = NLC(x) = NLC\left(\frac{y - Bias}{G}\right) = A\left(\frac{y - Bias}{G}\right)^2 + B\left(\frac{y - Bias}{G}\right) + C$$

Equation 12

Eq. 5 can be expressed as a second order polynomial:

$$x' = A'y^2 + B'y + C'$$

Equation 13

where A', B', C' are function of $A, B, C,\ Bias, G$:

$$A' = \frac{A}{G^2}$$

$$B' = \frac{B}{G} - 2\frac{ABias}{G^2}$$

$$C' = C - \frac{BBias}{G} + \frac{ABias^2}{G^2}$$

Equation 14

4 Annex 2

Table for the polynomial fit to the system gain.

The system gain is calculated following:

$$gain(V_{SS}, V_{OD}, V_{RD}, V_{OG}, T_{CCD}) = gain_{nominal} \times \left(1 + \sum_i Coefficient_i \times Term_i\right)$$

The terms and coefficients are in the following table, shown here for illustration purposes. Please note that, even though most of the terms are common to the two channels, some are different, which gives a total of 15 different terms in the two tables.

Table 1 Coefficients of the polynomial fit to get the system gain of the FM2 CCD from the CCD temperature [°C] and the bias voltages [V], given for the nominal and the redundant channel.

IASW DP entry	Term	Coeff. NOM	Coeff. RED
NLCCOEFF_D[0]	R_{SS}	8.8	8.8
NLCCOEFF_D[1]	R_{OD-SS}	22.0	22.0
NLCCOEFF_D[2]	R_{RD-SS}	9.0	9.0
NLCCOEFF_D[3]	R_{OG-SS}	5.75	5.75
NLCCOEFF_D[4]	$V_{SS} - R_{SS}$	– 2.8951E-02	– 2.9119E-02
NLCCOEFF_D[5]	$V_{OD} - V_{SS} - R_{OD-SS}$	3.0388E-02	3.0827E-02
NLCCOEFF_D[6]	$(V_{OD} - V_{SS} - R_{OD-SS})^2$	– 8.376E-03	– 4.140E-03
NLCCOEFF_D[7]	$V_{RD} - V_{SS} - R_{RD-SS}$	5.288E-03	7.773E-03
NLCCOEFF_D[8]	$(V_{RD} - V_{SS} - R_{RD-SS})^2$	– 6.852E-03	– 3.414E-03
NLCCOEFF_D[9]	$V_{OG} - V_{SS} + R_{OG-SS}$	– 1.1206E-02	– 8.437E-03
NLCCOEFF_D[10]	$(V_{OG} - V_{SS} + R_{OG-SS})^2$	0	6.44E-03
NLCCOEFF_D[11]	$(V_{OG} - V_{SS} + R_{OG-SS})^3$	1.640E-03	4.076E-03

NLCCOEFF_D[12]	$(V_{OD} - V_{SS} - R_{OD-SS}) \times (V_{RD} - V_{SS} - R_{RD-SS})$	1.3978E-02	8.502E-03
NLCCOEFF_D[13]	$(V_{OD} - V_{SS} - R_{OD-SS})^2 \times (V_{RD} - V_{SS} - R_{RD-SS})$	0	– 8.07E-03
NLCCOEFF_D[14]	$(V_{OD} - V_{SS} - R_{OD-SS}) \times (V_{RD} - V_{SS} - R_{RD-SS})^2$	6.612E-03	0
NLCCOEFF_D[15]	$(V_{OD} - V_{SS} - R_{OD-SS})^2 \times (V_{RD} - V_{SS} - R_{RD-SS})^2$	4.35E-03	– 5.10E-03
NLCCOEFF_D[16]	$(V_{RD} - V_{SS} - R_{RD-SS}) \times (V_{OG} - V_{SS} + R_{OG-SS})$	0	1.766E-03
NLCCOEFF_D[17]	$(V_{RD} - V_{SS} - R_{RD-SS})^2 \times (V_{OG} - V_{SS} + R_{OG-SS})$	0	2.171E-03
NLCCOEFF_D[18]	$(V_{RD} - V_{SS} - R_{RD-SS}) \times (V_{OG} - V_{SS} + R_{OG-SS})^2$	0	0
NLCCOEFF_D[19]	$(V_{RD} - V_{SS} - R_{RD-SS})^2 \times (V_{OG} - V_{SS} + R_{OG-SS})^2$	0	0
NLCCOEFF_D[20]	$(V_{OD} - V_{SS} - R_{OD-SS}) \times (V_{OG} - V_{SS} + R_{OG-SS})$	0	0
NLCCOEFF_D[21]	$(V_{OD} - V_{SS} - R_{OD-SS})^2 \times (V_{OG} - V_{SS} + R_{OG-SS})$	0	0
NLCCOEFF_D[22]	$(V_{OD} - V_{SS} - R_{OD-SS}) \times (V_{OG} - V_{SS} + R_{OG-SS})^2$	0	0
NLCCOEFF_D[23]	$(V_{OD} - V_{SS} - R_{OD-SS})^2 \times (V_{OG} - V_{SS} + R_{OG-SS})^2$	0	0
NLCCOEFF_D[24]	$T_{CCD} + 40$	– 1.106E-03	– 9.37E-04
NLCCOEFF_D[25]	Spare	0	0
NLCCOEFF_D[26]	Spare	0	0
NLCCOEFF_D[27]	Spare	0	0

Table to calculate the non-linearity correction.

The non-linearity correction is given by the following formula:

$$e^-_{lin} = a_m(e^-_{read} - knot_m)^2 + b_m(e^-_{read} - knot_m) + c_m$$

The parameters for the spline interpolation are in Table 4-5 of RD2 (read out frequency 230 kHz), and shown below for illustration purposes.

m	$knot_m$	a_m	b_m	c_m
1	0.0	-1.94482918345E-07	0.997736728997	0.0
2	7103.16429219	-2.54714606839E-10	0.994973840755	7077.27528186
3	13877.9456658	6.19551571033E-08	0.994970389483	13817.9938346
4	27963.1392963	8.15233959021E-08	0.996715690252	27844.6358768

5	62360.172491	8.41841447793E-08	1.00232401616	62225.1534463
6	80978.3482555	5.78852949964E-08	1.00545872657	80915.7794468
7	96220.4327926	2.39255611544E-07	1.00722331169	96254.5143336
8	114402.799912	2.13949699613E-05	1.01592377842	114647.315898
9	120304.91174	0.0012188125695	1.26847478895	121388.7038
10	121297.344431	-1.29277857111e-05	3.68765366499	123848.015749
11	122622.236656	-	-	-

The parameters for the spline interpolation are in Table 4-6 of RD2 (read out frequency 100 kHz), and shown below for illustration purposes.

m	$knot_m$	a_m	b_m	c_m
1	0.0	6.46419703187e-08	0.992065872118	0.0
2	28266.655961	1.02887151948e-07	0.995720296789	28094.0338803
3	62972.3354519	8.88435582923e-08	1.00286183383	62775.1093109
4	97466.9096729	9.32695548994e-08	1.00899107526	97474.1140581
5	111205.493458	7.87357596327e-06	1.01155385845	111353.82699
6	117732.116977	0.000200238893613	1.11432959057	118291.247449
7	121460.487946	-	-	-

<End of document>