CCD CHARACTERIZATION LAB

Satyapriya Das (SC20B159)

AIM:-

A) Understanding of the workings of a CCD

B) Characterization of the IIST KAF0401E CCD in the SBIG ST-7XE Camera

1. Read Noise
2. Dark Current
3. Linearity
4. Gain
5. Charge Transfer Efficiency

C) Analyzing data using IRAF

In this analysis we will characterize the CCD by finding its key properties of it such as Read noise, Dark current, linearity and gain.

Theory:

Read Noise:

It is the inherent noise introduced by the electronics used to read the signal from each pixel. It arises from the amplifier and analog-to-digital converter and sets the baseline noise level for the CCD.

Dark Current:

This is the generation of electron-hole pairs within the CCD due to random thermal motion, even in the absence of light. It increases with temperature and can overwhelm the signal at room temperature; hence it requires cooling for faint object detection. While the average dark current can be subtracted, its fluctuations contribute to the overall noise.

Linearity:

CCDs exhibit a linear relationship between the number of incident photons and the output signal over a wide range, making them ideal for quantitative measurements. This allows calibration using brighter objects for observations of faint ones. However, linearity breaks down at high signal levels, especially when the Analog-to-digital converter (ADC) reaches its full capacity.

Procedure:

In all the cases mentioned, except where the LED was used for constant illumination. The shutter was kept closed for read noise (0.12 sec, minimum shutter time) and dark current measurement. Also, since I am using wsl, if I try to give some other name it shows a segmentation error hence, I made copies of the files after each step and overwritten the existing files for the subsequent steps.

1. Rename:

ls *.fit > list1 is done, and then using 'rename' the extension is changed from fit to fits. In epar rename, writing this will change the extension to fits in all files simultaneously.

i. files = @list1
ii. newname = fits
iii. field = extn

2. Cosmic Ray Removal:

It is done using the module 'cosmicrays' from (noao,imred, crutil). It will remove cosmic rays from all files.

3. Master Bias:

ls bias*.fits > biasframes ; to make list of bias frames. Then using imstat and giving @biasframes as input, we get the statistics of the bias frames. Using imcombine , all the biasframes at operating temperature were combined.

4. Bias Subtraction:

ls *.fits > list2 is made where manually I removed the bias files. Then the master bias is subtracted from them using imarith. Using epar imarith

(a) operand1 = @list2 (b) op = - (c) operand2 = masterbias as input .

5. The above procedure is followed for read noise, dark current, gain and linearity respectively.

Analysis

Bias Frames:

List of input images	NPIX	MEAN	STDEV	MIN	MAX	Temperature (Celsius)
Masterbias.fits	390150	105.31	5.256	89	137	0
bias_-9_temp.fits	390150	104.60	7.200	74	139	-9
bias_-5_temp.fits	390150	104.10	7.193	72	154	-5
bias_5_temp.fits	390150	105.20	7.217	78	142	5
bias_10_temp.fits	390150	105.80	7.182	74	168	10
bias_15_temp.fits	390150	106.70	7.222	78	142	15

Table 1: List of Bias Frames

Read Noise:

List of input images	NPIX	MEAN	STDEV	MIN	MAX	Time (sec)
rn_0.5s_0_temp_avg.fits	390150	0.04728	4.296	-23.92	18.92	0.5
rn_1s_0_temp_avg.fits	390150	2.11300	4.300	-29.58	81.17	1.0
rn_3s_0_temp_avg.fits	390150	0.64340	4.309	-29.92	25.58	3.0

Table 2.1: Average Read Noise

List of input images	NPIX	MEAN	STDEV	MIN	MAX	Time (sec)
read_noise_0.5s_0_temp.fits	390150	0.7628	6.282	-30.25	29.25	0.5
read_noise_0.5s_0_temp_1.fits	390150	-0.7286	6.283	-36.25	27.00	0.5
read_noise_0. 5s_0_temp_2.fits	390150	0.1076	6.298	-30.25	28.00	0.5
read_noise_1s_0_temp.fits	390150	2.2100	6.304	-28.25	227.50	1.0
read-noise_1s_0_temp_1.fits	390150	2.0780	6.284	-33.25	36.75	1.0
read_noise_1s_0_temp_2. fits	390150	2.0530	6.288	-29.25	30.25	1.0
read_noise_3s_0_temp. fits	390150	1.4840	6.300	-30.50	37.50	3.0
read_noise_3s_0_temp_1.fits	390150	-0.9680	6.302	-33.50	80.25	3.0
read_noise_3s_0_temp_2. fits	39150	1.3530	6.299	-31.25	39.25	3.0

Table 2.2: Read Noise frames

Dark Current:

List of input images	NPIX	MEAN	STDEV	MIN	MAX	Time (sec)
dark_current_5s_0_temp. fits	390150	-0.2168	6.311	-30.50	82.5	5.0
dark_current_10s_0_temp. fits	390150	0.1999	6.330	-29.25	61.0	10.0
dark_current_20s_0_temp.fits	390150	0.4711	6.659	-31.25	563.0	20.0
dark_current_50s_0_temp.fits	390150	3.9490	7.257	-23.50	1188.0	50.0
dark_current_100s_0_temp. fits	390150	3.7090	6.586	-26.25	283.8	100.0
dark_current_200s_0_temp. fits	390150	5.0920	7.159	-25.25	747.5	200.0
dark_current_500s_0_temp.fits	390150	15.7000	9.758	-15.00	1418.0	500.0

Table 3.1: Variation with Time

List of input images	NPIX	MEAN	STDEV	MIN	MAX	Temperature (Celsius)
var_dark_current_100s_-9_temp.fits	390150	1.776	8.168	-37.00	321.00	-9.0
var_dark_current_100s_-5_temp. fits	390150	-0.410	6.552	-31.25	217.25	-5.0
var_dark_current_100s_0_temp. fits	390150	3.153	6.699	-28.25	445.20	0.0
var_dark_current_100s_5_temp.fits	390150	4.329	7.571	-30.75	1164.75	5.0
var_dark_current_100s_10_temp. fits	390150	19.920	8.893	-31.00	618.00	10.0
var_dark_current_100s_15_temp.fits	390150	40.120	9.941	-17.00	941.00	15.0

Table 3.2: Variation with Temperature

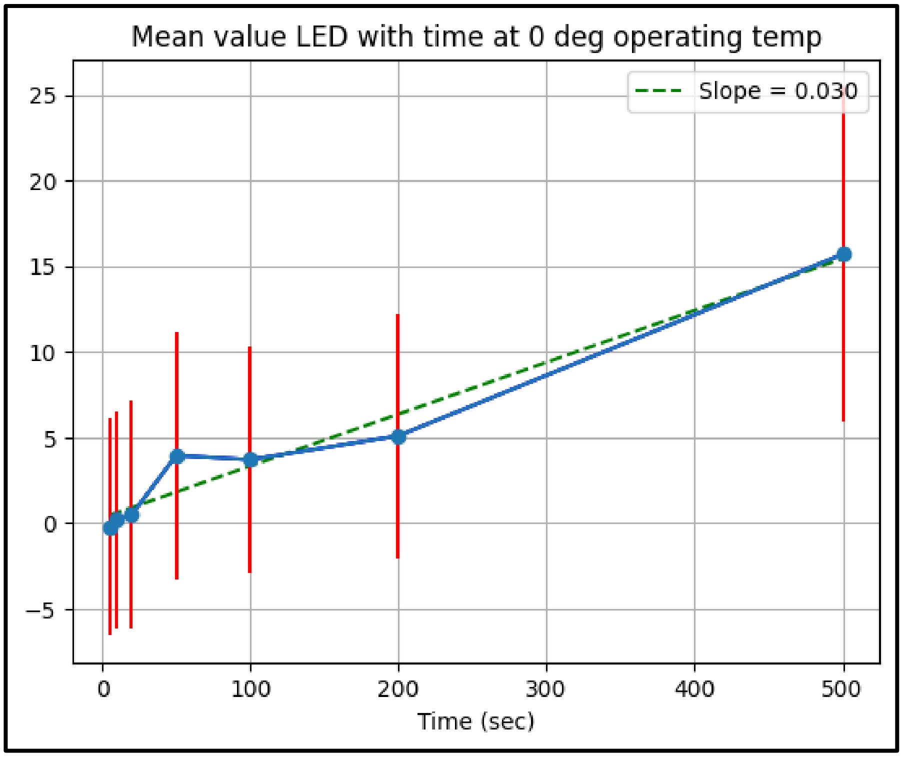

Fig 1.1: Variation of dark current vs time

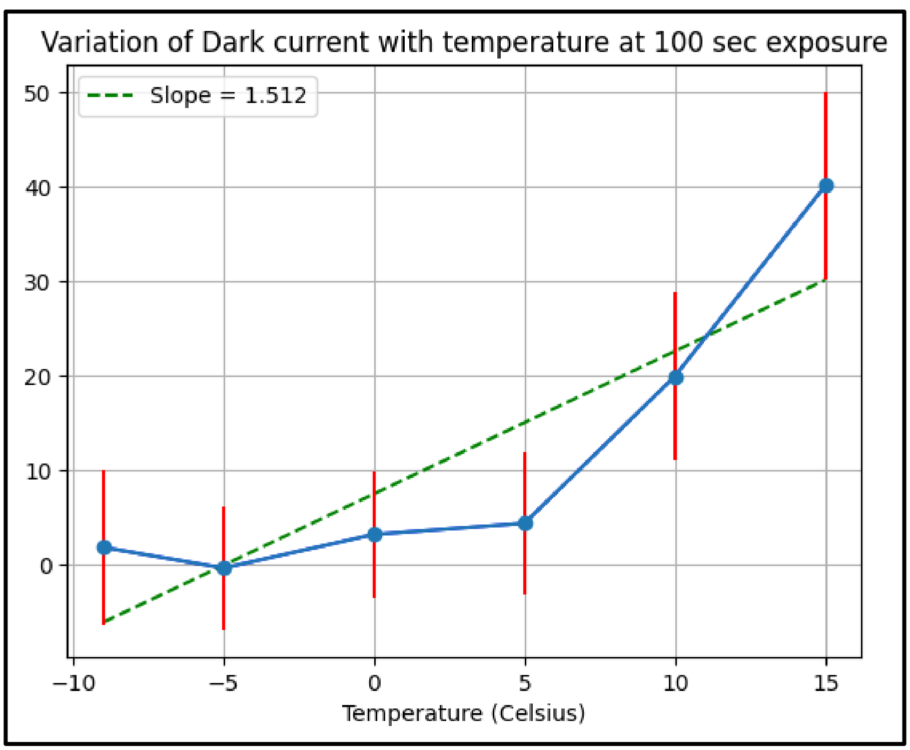

Fig 1.2: Variation of dark current at operating temperature

Linearity:

List of input images	NPIX	MEAN	STDEV	MIN	MAX	Time (sec)
led-ls_0_temp.fits	390150	1367.0	125.9	953.8	1628.0	1.0
led-3s_0_temp.Fits	390150	3876.0	352.6	2805.0	4490.0	3.0
led_5s_O_temp.f its	390150	6270.0	569.3	4612.0	7239.0	5.0
led-8s_0_temp.Fits	390150	9763.0	883.8	7229.0	11189.0	8.0
led-10s_0_temp.fits	390150	11524.0	1042.0	8588.0	13178.0	10.0
led-13s_0_temp.fits	390150	14126.0	1278.0	10504.0	16189.0	13.0
led-15s_0_teop.fits	390150	15806.0	1430.0	11766.0	18196.0	15.0
led_18s_0_temp.fits	390150	18341.0	1658.0	13676.0	20960.0	18.0
led-20s_0_temp.fits	390150	19085.0	1726.0	14271.0	21945.0	20.0
led_23s_0_tenp.fits	390150	18571.0	1674.0	13847.0	21240.0	23.0
led-25s_0_temp.Fits	390150	16904.0	1526.0	12576.0	19373.0	25.0
led-30s_0_temp.fits	390150	18551.0	1675.0	13804.0	21226.0	30.0
led_35s_0_temp. fits	390150	20547.0	1854.0	15270.0	23593.0	35.0
led-40s_0_temp.Fits	390150	22221.0	2004.0	16560.0	25386.0	40.0
led_45s_0_temp. fits	390150	22856.0	2061.0	16941.0	26250.0	45.0
led_50s_0_temp.Fits	390150	23673.0	2133.0	17662.0	27087.0	50.0
led-55s_0_temp.Fits	390150	24402.0	2198.0	18145.0	27977.0	55.0
led_60s_0_temp.fits	390150	24939.0	2245.0	18557.0	28497.0	60.0
led_65s_0_temp. fits	390150	25462.0	2292.0	18979.0	29065.0	65.0
led_70s_0_temp.fits	390150	26804.0	2411.0	19944.0	30586.0	70.0
led-90s_0_temp.Fits	390150	35421.0	3289.0	26095.0	40654.0	90.0
led-100s_0_temp.fits	390150	33940.0	3153.0	25003.0	38833.0	100.0
led_120s_0_temp.fits	390150	31005.0	2886.0	22836.0	35470.0	120.0

Table 4: LED data after bias subtraction and cosmic ray removed with time.

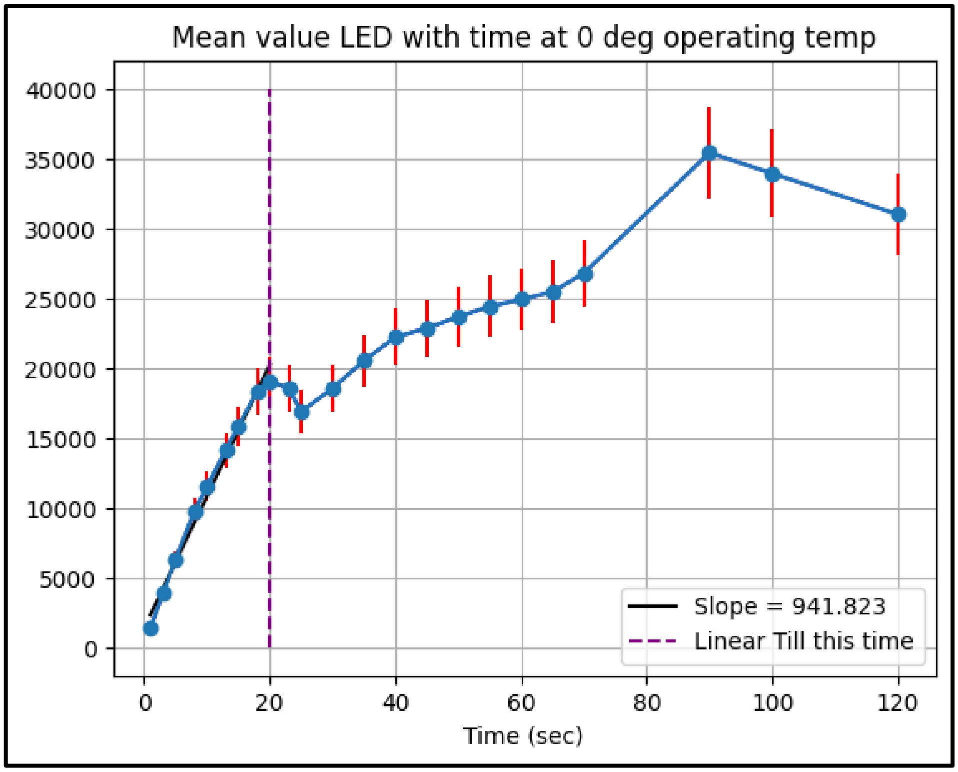

Fig 2: Mean Value of LED with time at operating temperature.

Gain:

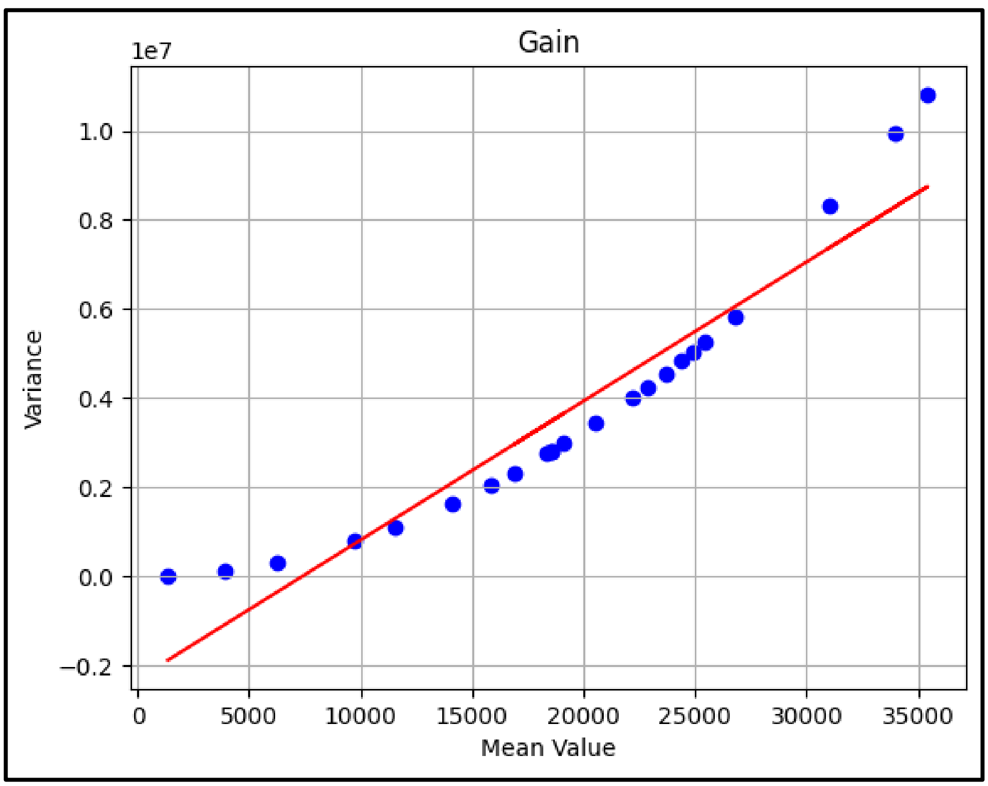

Fig 3.1: Gain Calculation with full image

Variance/Mean = 1/g = 312.1288 : **g = 0.032**

Since there was some issue with the LED, the illumination over the LED was not constant and a gradient can be seen. To deal with the issue, a segment of the image is taken for analysis using python.

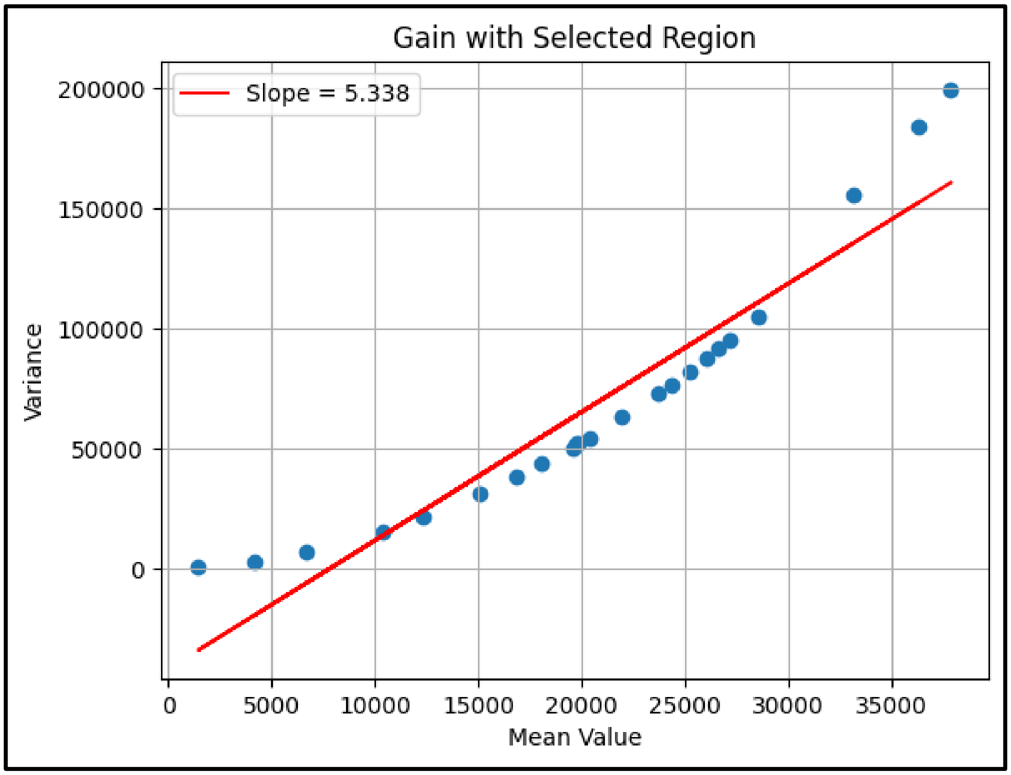

Fig 3.2: Gain Calculation with selected portion of image

Variance/Mean = 1/g =5.338 : **g = 0.187**

Charge Transfer Efficiency

Cosmic ray selection is done by reading the bias-subtracted dark frame of 500s. The CTE is calculated from the value and the one in the next pixel. To get some ideal values, only those are selected where the value is greater than 15000.

```python
import astropy.io.fits as fits
import glob

directory_path = "D://IIST//pdf books iist//IIST//SEMESTER 8//Lab//CCD"
# bias = "D://IIST//pdf books iist//IIST//SEMESTER 8//Lab//bias_mean_combine.fits"

data_bias = np.array(fits.open(bias)[0].data)
file_names = glob.glob(directory_path + '/*.fits')

# Open each file
for file_name in file_names:
    hdulist = fits.open(file_name)
    data = np.array(hdulist[0].data)
    print("mean: ", np.mean(data),'\n',"std: " ,np.std(data),'\n'
    "minimum:", np.min(data),'\n',"maximum:", np.max(data),'\n')

    coord_list = np.argwhere(data>15000)
    for coord in coord_list:
        if data[coord[0],coord[1]+1]>0:
            print(f"Cosmic Val {data[coord[0],coord[1]]} at {coord} and next val {data[coord[0],coord[1]+1]}
                  CTE = {coord[1]/(coord[1]+(data[coord[0],coord[1]+1]/data[coord[0],coord[1]])):.6f}\n")
    # Close the file
    hdulist.close()
```

Code: To select the cosmic ray sources and find CTE

PIX COORD	PIX VALUE	NEXT PIX VALUE	CTE
{182,162}	16433.75	61.75	0.999977
{220,215}	43226.75	142.50	0.999985
{345,198}	31169.00	111.75	0.999982
{381,633}	53613.25	249.00	0.999993
{493,484}	24678.25	97.50	0.999992

Table 5: CTE Value calculations

Result:

The following is the summary of this analysis:

- **Operating temperature:** 0 Degree Celsius
- **Dark Current:** 0.03 DN/s
- **Gain:** 0.187
- **CTE:** 0.99998