A re-assessment of EPIC telemetry rates

and

the implications of EPIC operating modes on the spectral quality of the data

L.Chiappetti - IFCTR
Issue 1- Tue, 26 Apr 1994

1 Introduction

This document reports details of the simulations done recently (and whose conclusions are reported in [Ref. 1]). The main aim of these simulations has been to establish a quantitative criterium for the limiting rates (maximum count rates allowed by pile-up, which does not sensibly affect the spectral quality of the data). The spectral quality of the data is assessed taking into account all elements of the EPIC chain (detector operating modes, on-board electronics & software modes, possible ground software corrections). Once the limiting rates are known, translation into bit rates is straightforward.
As a by-product, some considerations on the operating modes based purely on the spectral quality of the data are derived. In the future it is planned to make further simulations to assess the timing quality of the data.

This note supersedes [Ref. 2] of 25 Nov 92.

2 Scheme of the simulations

The simulation has been pursued using the latest version of the EPOS software which is described elsewhere in detail [Ref. 3].

The programs simulating the interaction of photons and charged particles in the CCD were adapted for EPOS using programs supplied by P.Sarra, based on original work by C.Pigot of SAp Saclay, and are virtually unchanged since [Ref. 2 25 Nov 92]

Essentially the route followed was the following :

For the simulation of the target, a general purpose randomizer program is able to produce a list of photons with energies (or arrival times) randomly distributed according to any given spectral (or temporal) distribution (described in an external program). For our purposes we have used Crab-like spectra of different intensities.

Simulation of the X-ray background has been neglected at this stage (see ref. [2] and references therein.)

The effects of the optics are simulated very crudely: the total number of photons is determined by the geometric collecting area (3000 cm^2); photons are also assigned random positions using a bivariate normal distribution (with σ=12.72 arcsec) as approximation of the on-axis PSF; finally the transmission of the mirrors (as supplied by B.Aschenbach via C.Erd [Ref. 4]) is applied to each photon according to its energy.

Filters and dead layers in the chip were neglected (but all photons below 0.1 keV were rejected).

In a few cases the presence of the grating box has been simulated via the random selection of 43% of the photons (irrespective of energy).

A subset of the photons falling on the representative chip positions (central chip only) was then extracted (converting from microns to pixels) and submitted to CCD detection simulation. The photon list in input to the CCD is considered the "*reference input file*".

Cosmic ray background files resulting from the CCD interaction simulation are defined at chip level (since there is no simulation of the shielding by part of the spacecraft or instrument structure, a single file may apply to all chip positions).
The CCD simulation is representative of the splitting of photons or particles into events.
An advantage of the simulation is that each event carries with it a number of flags recording its origin, the kind of splitting occurred, etc.

Files containing only target photons, or only cosmic ray events were processed separately, in order to disentangle the effects of the ExCE-ExDH algorithms.

For each reference file (in the case of the target the procedure was repeated for different source intensities) the post-detection file was passed through two or three steps (a simulation of the readout in each of the different operating modes, a simulation of the EMCE or EPCE and ExDH in the appropriate mode, a simulation of additional ground processing in the pn case), producing a final "*reconstructed photon file*".

The assessment of the spectral quality (as a function of mode and source intensity) was done comparing the reconstructed photon file with the "reference input file" as explained in 3.1.

2.1 Standard configurations

2.1.1 Chip geometrical arrangement

Only one chip was considered (although EPOS is capable of describing the geometrical arrangement of all chips in the focal plane), namely the central one for the MOS case, and one of the 4 central ones for the pn case. For more detail on the reference files used consult the EPOS documentation [Ref. 3].

The target has been located in the centre of the MOS chip, or close to a corner of the pn chip (for the burst mode care has been taken to locate the source opposite to the readout end).

2.1.2 Chip physical characteristics

Chip physical characteristics used are reported here (see [EPOS Ref.3] for complete parameter files). These values have been verified personally by members of the CCD teams.

MOS

```
70                                     ! depletion depth (microns)
30                                     ! field free depth (microns)
0  (bulk)                              ! substrate depth (microns)
10.                                    ! substrate diffusion lenght (microns)
0. (bulk)                              ! reflectivity of the field free boundary
2.0E12                                 ! impurity concentration (cm-3)
173.                                       ! CCD temperature (∞K)
```

pn

```
270.                                   ! depletion depth (microns)
0.                                     ! field free depth (microns)
0                                      ! substrate depth (microns) ignore epitaxial
10.                                    ! substrate diffusion lenght (microns) n/a
0.                                     ! reflectivity of the field free boundary
2.E12                                  ! impurity concentration (cm-3)
183.                                   ! CCD temperature (∞K)
```

2.1.3 Operating mode parameters

The definition of the operating modes is based on the latest definitions, which in part supersede the OMWG Report [Ref. 5] and take into account all latest changes (usage of frame store for all MOS chips, redefinition of MOS modes, changes in definition of the pn T and B modes).

The reference for the "new" MOS imaging modes (frame store over full field, window, refresh frame store) and for the new timing mode (formerly called Fast Window mode) is [Ref. 8].
Operating modes for the pn case are as described in [Ref. 5], except that the Burst mode has been redefined (as in [Ref. 9] , and that fig. 2.10 in [Ref. 5] (about the Timing mode) was incorrect and has to be replaced as in [Ref. 10].

For all readout simulations, a set of important parameters are the "clock tics".

∑ For the MOS cases three basic times have been used : row shift is 2 µs ; pixel shift is 1 µs ; pixel sampling is 5 (or 6 ?) µs. Overscan in x (312 pixels read instead of 300) has been neglected.

∑ For the pn case two basic times are necessary : a 20 µs row read and a 1 µs row shift.

The following configurations have been used in the simulation :

MOS Imaging, Frame store readout of entire chip via 1 node	(frame time = 2.163 s)
MOS Imaging, Frame store readout of entire chip via 2 nodes	(frame time = 1.0827 s)
MOS Imaging, 100x100 centered window via 1 node	(frame time = 0.1125 s)
MOS Imaging, Refresh frame store of entire chip via 2 nodes	(duty cycle of 5.76% of 1.0827 s)
MOS Timing mode, 54x54 window	(pixel time 0.87 ms)
pn Full Frame mode	(frame time = 0.048 s)
pn Large Window mode has NOT been simulated.	
pn small Window mode (48x30 window in corner)	(frame time = 4.625 ms)
pn Timing mode	(frame time = 0.39 ms)
pn Burst mode (NB duty cycle << 1)	(frame time = 4 ms)

2.1.4 Controller and EDH configurations

The simulation of the EMCE or EPCE and ExDH has been done via the following modules :

a) for the MOS Imaging mode (FS,W,RFS) one module simulates the EDU pattern recognition (described in [Ref. 6]), using as pattern library the canonical one (pag.15 of [Ref. 6]) and also the additional application of thresholds in the EMDH.

b) for the MOS T mode another module simulates cosmic ray rejection and split event reconstruction (to be allocated between EDU and EMDH) with an 1-d algorithm analogous to 2-d one used in the previous case.

c) for the pn full imaging modes (FF and Large W) one module simulates the EPCE application of thresholds and cosmic ray rejection (rejection of three entire rows centered on one pixel at the overflow charge level) per [Ref. 7]

d) for the pn W mode another module simulates the EPCE application of thresholds and cosmic ray rejection (rejection of entire frames containing one pixel at the overflow charge level) also per [Ref. 7]

e) for the pn T and B mode, a third module just simulates the EPCE application of a low and high thresholds [Ref. 7 and 5].

f) in the imaging cases c-d, and in the Burst mode (part of case e), a further module simulates split event reconstruction (as preliminary approximation using the same algorithm used for the MOS in (a)). It has to be decided whether this function has to be done on the ground or on-board (not doing it will increase the data rate by a factor 1.2 to 1.4). It is anyhow required to do it in the simulation in order to make a meaningful comparison between "reconstructed" and "input" data.

g) in the Timing mode (part of case e), a further module simulates split event reconstruction (as preliminary approximation using the same 1-d algorithm used for the MOS in (b)). The same considerations reported above for (f) apply.

2.1.5 Instrument level configurations

Contrary to the cases in [Ref. 2] the simulation has been limited to the central chip, and the differences in sky coverages of the outer chips (for background) have been neglected.
The following configurations might be considered representative of observational situations for the computation of count rates in one entire camera :

Chip with target	Other chips
MOS	
Imaging Frame Store mode, entire chip	Imaging Frame Store, entire chip (x6)
Imaging, window option	Imaging Frame Store, entire chip (x6)
Imaging, refresh frame store option	Imaging Frame Store, entire chip (x6)
Timing	Imaging Frame Store, entire chip (x6)
pn	
Full Frame	Full Frame (x11)
Large window (on 6 chips)	not used
Small window (on 1 chip)	Full Frame (modif. exp. time) (x11)
Timing	not used
Burst (on 1 or 2 chips, see text in 3.2.11)	not used

3 Pile-up and limiting event rates

3.1 Procedure

First I generated a reference photon list for a Crab-like spectrum for a 10 mCrab point source, exposure 100 s. This amounts to 27171 photons (i.e. 1mCrab = 27 cts/s before mirrors).

I propagated this photon list through a coarse simulation of optics (and optionally grating, no filters or dead layers so far) as described above in Section 2. I also converted photon positions from micron to pixels, and relocated source in appropriate position on the chip as described above in 2.1.1.
This leaves 13155 photons (1 mCrab = 13 cts/s after full mirror, before CCD).

These are the *reference photon input files* (they are separate for MOS and pn because of the different chip sizes).

Passed the reference input files through the CCD detection simulation. The output is saved as a *reference event file*.

The source intensity can be scaled without repeating all the above just by scaling the arrival times in the reference event file.
This way each test corresponds to a different total exposure time, but to a constant number of photons (therefore all spectra have the same statistics). An exception to this is the case when the flux is scaled to very low intensities (due to the way times are represented as a 32-bit integer number of microseconds, if the original 100 s interval is stretched beyond 2147 s only the photons falling in the latter interval are considered).

The reference event file is passed through the simulation of the readout modes, of the EMCE/EPCE and ExDH, and of the additional split event reconstruction. The output is the *reconstructed photon file*.

For each set of operating modes and intensities, an *input spectrum* P_i is generated from the reference photon input file, and a *reconstructed spectrum* is generated from the reconstructed photon file. The latter is a charge spectrum, which is resampled back to an energy spectrum on the same scale as the input spectrum. The reconstructed energy spectrum is called O_i .

Both spectra are accumulated as count histograms, so that the associated errors are Poissonian : $\sigma_{Pi} = \mathrm{sqrt}(P_i)$ and $\sigma_{Oi} = \mathrm{sqrt}(O_i)$.

The *residuals* are defined as $R_i = O_i - P_i$ and $\sigma_{Ri} = \mathrm{sqrt}(\sigma_{Oi}{}^2 + \sigma_{Pi}{}^2) = \mathrm{sqrt}(O_i + P_i)$.

If all photons in the input reference file which are split into more than one event are reconstructed back into one photon, one would expect the residuals to be *distributed around an expected value of zero*. The "noise" in the residuals will be due just to the effects of the energy resolution and of the low charge threshold.

If pile-up occurs, this results in negative residuals in the more intense part of the spectrum.
I have estimated the quality of the reconstruction in the following ways :

a) I have inspected visually a plot of the residuals. Some plots are shown below (note that, although the vertical scales are different for each plot, the total number of counts in each spectrum is constant because the exposure time is adjusted accordingly - with the exception noted above for very low intensities).

b) In the attempt to get a more quantitative evaluation I have computed the total residual and its error, and computed its significance S (in σ) :

$$R_{tot} = \mathrm{Isu}(_{i},,R_{i}\ ) \qquad \sigma_{\tau ot} = \mathrm{sqrt}\ \mathbf{B}\mathrm{bc}((\mathrm{Isu}(_{i},,\sigma_{Ri}{}^{2})) \qquad S = R_{tot} / \sigma_{\tau ot}$$

c) I have computed the χ^2 of the residuals around their expected value (i.e. zero), and computed the associated probability P (which should be 0 if the residual are consistent with zero, and 1 if they are radically different from zero) :

$$\chi^2 = \mathrm{Isu}(R_{i\neq 0},,(R_i/\sigma_i)^2) \qquad P(\chi^2\ |\nu) = 1 - Q(\chi^2\ |\nu)$$

where the number of degrees of freedom ν is the number of points with non-zero residuals.
However the number of points (and therefore the probability P) depends on the binning of the spectra. In practice I have used for each case two binnings, a fine one (bin width = 10 eV) and a coarse one (bin width = 100 eV).

d) the above quantitative criteria are preferred to the estimation of the *pile-up fraction* (as defined in [Ref.2]) being less or equal than 5%. In fact this considered only the *physical pileup events*, (i.e. those which actually fall in the same pixel) and neglected the additional *geometrical pileup events*, (i.e. when two photons give rise to events in adjacent pixels), which can be estimated by the pattern analysis performed by modules a-b-f (see above 2.1.4).
The difference between *recoverable* pileups (or *diagonal* events, when two single events are adjacent only in a corner) and irrecoverable pileups (when two single events are adjacent in the vertical or horizontal direction) has not been considered (although the simulations would allow to count and correct the recoverable ones).

In conclusion there is a rough agreement among all methods that the limiting rate can be put around the case S=5 (5-sigma significance of the overall residual).

Extensive examples are shown below in fig. 1-11.
In all such figures I plot on the same log-lin scale both the data and the residuals. The input spectrum is shown as solid crosses with error bars. The reconstructed spectrum is shown as a thicker histogram in gray (no errors are plotted in order not to overcrowd the graph). The residuals are shown as solid error bars around or below the horizontal line at y=0; the thicker curve in light gray is just a boxcar smoothing of the residuals in order to make more apparent the general trend.

As a comparison (since in practice one would rely on post-fit residuals to assess detection of line features) I have also done a purely numerical exercise (i.e. no CCD simulation at all), taking as reference spectrum P_i a perfect absorbed power law distribution, and as perturbed spectrum O_i, either the reference perturbed with random noise (which gives P=0 as expected), or the reference plus a broad line, again perturbed with random noise (which gives P=1 insofar the intensity of the line is sufficient for detection).

Source	bin size	residual R_{tot}	error $\sigma_{\tau ot}$	S	χ^2	χ^2/DoF	DoF	P
	(eV)	(counts)		(sigma)				Probab.
PL	10	-124	±144	0.74	722	0.73	984	0
	100				48.63	0.49	99	4.8e-6
PL + line	10	3017	±175	17.2	1283.3	1.30	984	1.0
	100				661.9	6.68	99	1.0
PL + line/2	10	1446	±171	8.5	856.3	0.87	984	1.3e-3
	100				215.5	2.17	99	1.0
PL + line/4	10	661	±168	4.0	744.3	0.75	984	0
	100				89.5	0.90	99	0.24

3.2 Results

3.2.1 MOS Imaging mode, default Frame Store option

The default option reads the entire chip in the Frame Store modality from one output node (the frame time is 2.163 s). The default EMCE imaging mode is used for split event reconstruction.

For this mode I present an extensive set of figures 1-3ab, showing at both binnings (10 and 100 eV) three cases : 0.1 mCrab (no pileup), 0.5 mCrab (borderline), 1 mCrab (excessive pileup).

The limiting rate for this mode is assumed to be at 0.5 mCrab. The quality factors corresponding to some test cases (including those plotted) are reported in table below.

Source	bin size	residual R_{tot}	error $\sigma_{\tau ot}$	S	χ^2	χ^2/DoF	DoF	P
(mCrab)	(eV)	(counts)		(sigma)				Probab.
0.1	10	-49	±74	0.7	500.3	0.71	706	0
	100				48.29	0.53	91	6.8e-5
0.5	10	-750	±160	4.7	698.2	0.79	889	5.4e-7
	100				110.4	1.12	98	0.81
1	10	-1391	±158	8.8	789.8	0.88	896	8.9e-3
	100				187.2	1.91	98	0.99
2	10	-2568	±154	17	1034.8	1.15	897	0.999
	100				431.8	4.4	98	1

The tests were also done in the case of grating obscuration. One would expect to scale the limiting rates by a factor 2.32 = 1/0.43, but it looks like one can get somewhat higher (see table here below and Fig. 4ab.)

Source	bin size	residual R_{tot}	error $\sigma_{\tau ot}$	S	χ^2	χ^2/DoF	DoF	P
(mCrab)	(eV)	(counts)		(sigma)				Probab.
0.5	10	-175	±105	1.6	592.8	0.74	805	0
+grating	100				53.3	0.54	97	9.1e-5
1	10	-304	±105	2.8	600.1	0.74	808	0
+grating	100				61.2	0.63	97	1.6e-3
2	10	-553	±104	5.3	640.9	0.79	811	2.7e-6
+grating	100				89.1	0.92	97	0.29
3	10	-775	±103	7.5	672.0	0.83	808	1.6e-4
+grating	100				122.8	1.26	97	0.96

3.2.2 MOS Imaging mode, 2-node Frame Store option

A possibility to enhance the readout speed is to use the Frame Store modality from two output nodes (giving a frame time of 1.0827 s). For the rest this is identical to the previous case.

As expected (see also quality factors in following table), the limiting rate for this mode is 1 mCrab (double than the previous case), which is shown in figure 5a (top).

Source	bin size	residual R_{tot}	error $\sigma_{\tau ot}$	S	χ^2	χ^2/DoF	DoF	P
(mCrab)	(eV)	(counts)		(sigma)				Probab.
0.5	10	-493	±161	2.7	667.4	0.75	891	0
	100				85.7	0.87	98	0.19
1	10	-742	±160	4.6	696.6	0.78	887	4.7e-7
	100				107	1.09	98	0.75
2	10	-1336	±158	8.4	782	0.87	889	4.2e-3
	100				174.3	1.78	98	1

3.2.3 MOS Imaging mode, window option

As an example of the window option I have used one readout node and a 100x100 window centered around the source (this gives a frame time of 0.1125 s). The size of the window is slightly larger than 3 HEW (81 pixels), such to include all events from the target (this makes simpler the comparison between input and reconstructed spectra) and also in principle allow some estimation of the background.

The limiting rate for this mode is of the order of 7 mCrab (shown in figure 5b, bottom).

Source	bin size	residual R_{tot}	error $\sigma_{\tau ot}$	S	χ^2	χ^2/DoF	DoF	P
(mCrab)	(eV)	(counts)		(sigma)				Probab.
7	10	-721	±160	4.5	683.9	0.77	888	5.9e-8
	100				98.34	1.00	98	0.52
10	10	-933	±160	5.8	703	0.79	890	9e-7
	100				118	1.2	98	0.91
20	10	-1538	±157	9.7	815	0.91	891	3e-2
	100				202.5	2.06	98	1

3.2.4 MOS Imaging mode, Refresh Frame Store option

In the Refresh Frame Store option of the imaging mode, the exposure time is programmed to be (much) less than the frame time (and data coming during the remaining part of the frame are discarded). As an example I have chosen to read the entire chip from 2 nodes, using an exposure time of 0.06 s (the cycle time being 1.8027 s), which gives a duty cycle of 5.76%.

The limiting rate under these conditions is of the order of 50 mCrab (figure 6a, top). The amount of pileup at 100 mCrab (figure 6b, bottom) is clearly not acceptable. Therefore this mode, in the conditions assumed as example, cannot be used to study very strong sources.

One possible way out would be (as suggested by C.Pigot) to decrease adaptively the duty cycle to follow the source intensity (e.g. exposure of 0.03 s, duty cycle 2.99% is shown in fig. 7a (top) for 100 mCrab, and in the bottom row - marked with an asterisk - of the following table).

Source	bin size	residual R_{tot}	error $\sigma_{\tau ot}$	S	χ^2	χ^2/DoF	DoF	P
(mCrab)	(eV)	(counts)		(sigma)				Probab.
10	10	-50	±38	1.3	204.4	0.25	808	0
	100				30.99	0.32	96	0
50	10	-146	±37	3.9	221	0.27	805	0
	100				47.8	0.50	96	9.4e-6
100	10	-242	±36	6.8	266	0.33	806	0
	100				95.8	0.99	96	0.51
100 (*)	10	-100	±26	3.8	185	0.23	802	0
	100				52	0.54	96	0

NB the comparison bewteen input and reconstructed spectrum is approximated since the input spectrum is derived from the totality of the data, and scaled down by the duty cycle.

3.2.5 MOS Timing mode

In the example considered for the Timing mode (formerly Fast Window) I have used a window of 54x54 pixels (2 HEW) around the source as recommended (the number of events falling out of the window is 324 out of 16698 and is neglected in the comparison between input and reconstructed data).
Split event reconstruction is done (note that the allocation of this task between EMCE and EMDH is not yet defined) using an 1-d algorithm analogous to the 2-d one used by the EMCE Imaging modes (i.e. a "pattern analysis" based on the 5x1 projections of the canonical patterns, which in the 1-d case leave just 1 monopixel, 2 bipixel and 1 tripixel patterns). The case of events split among consecutive frames is found negligible (≤ 0.06 %) and is ignored.

The limiting rate can be placed at about or less than 100 mCrab (see Fig. 7b, bottom).

Source	bin size	residual R_{tot}	error $\sigma_{\tau ot}$	S	χ^2	χ^2/DoF	DoF	P
(mCrab)	(eV)	(counts)		(sigma)				Probab.
10	10	-374	±161	2.3	686	0.76	893	5.9e-8
	100				91.8	0.92	99	0.31
50	10	-600	±160	3.7	697	0.77	894	2.4e-7
	100				100.04	1.01	99	0.55
100	10	-955	±159	6	740	0.82	898	3.8e-5
	100				131.2	1.32	99	0.98
200	10	-1570	±157	10	862	0.95	904	0.16
	100				237.6	2.40	99	1

Also in this case some tests were done in the case of grating obscuration. Here too one would expect limiting rates to be higher by a factor 2.32=1/0.43, but it looks like one cat get higher, e.g. up to 300 mCrab.

3.2.6 pn : introductory note

In order to assess the limiting rate in the pn case, one must produce a reconstructed photon file. The current information about the actions done by the EPCE is rather scarce, but it appears clear that EPCE will just do some cosmic ray rejection (see sections 2.1.4 and 4), and not split event reconstruction. Now, while it is true that the larger pn pixels should diminish splitting, one must also note that the perimeter of the pixel is larger, so there is an increased probability of a photon impinging close to a pixel boundary. Actually, for 13155 incident photons, one gets 16698 events in the MOS case, and 20839 events in the pn case. However most of the pn events are split just in two pixels, and in a large fraction of cases the charge content of one of them is below the low charge threshold (here assumed at 25 electrons). Nevertheless one is left with split events after the EPCE, and, even if one thinks viable to downlink them, it is important to estimate a reconstruction efficiency to assess the limiting rate.

Also one should note that the larger area of pixels increases the probability of physical pileup (two photons in same pixel).

I have used (in lack of better information) the same algorithm used in the MOS EDU with the same canonical pattern library. Note that this is applied on the output of EPCE, i.e. after thresholding ! It looks like that there is a number of excessive rejections, so that the algorithm probably requires some tuning (different pattern library, relaxation of guard ring clause ? another possibility is that the CCD detection simulation is unrealistic, but this looks unlikely by common agreement with CCD team members).

Therefore the results reported below are to be regarded as provisional.

For clarity I have reported summary tables which give the event budget at the various stages of reconstruction.

3.2.7 pn Full Frame mode

The limiting rate in this case appears surprisingly to be no higher than 2 mCrab (the cases of 2 and 5 mCrab are shown in fig. 8ab).

Source	bin size	residual R_{tot}	error $\sigma_{\tau ot}$	S	χ^2	χ^2/DoF	DoF	P
(mCrab)	(eV)	(counts)		(sigma)				Probab.
1	10	-380	±161	2.3	756.8	0.81	931	7.6e-6
	100				97.7	0.99	99	0.48
2	10	-794	±159	4.9	771.6	0.83	930	4.7e-5
	100				123.3	1.25	98	0.95
5	10	-1922	±156	12.3	926.5	0.99	939	0.40
	100				277.8	2.8	98	1

It appears that the effect is due to excessive pile-up as shown by the table below. As a representative example consider the following case of a single frame (for 10 mCrab intensity) : there are 9 photons which give rise to 13 events and are reconstructed as 5 photons as follow :

2 events are single (charges 195 and 382) and are preserverd

1 photon is split into similar parts (charge 90+49) and is reconstructed

1 photon is split into unequal parts (charge 923+1) and is taken as single (as 923)

2 photons (each of them split) give rise to a class-31 pileup (central pixels coincide) with charges (512+3+1+19 in a square) and is therefore reconstructed as one (as 512)

2 photons (split) give rise to a class-32 pileup (central pixel of one coincides with split pixel of other) with charges (317+215)

in addition a single photon with charge 190 falls in the pixel just above 215 giving rise to an L-shaped pattern.

therefore the entire set of 3 pixels (with highest charge in a corner of the L) is rejected as unknown pattern

Hence 2 photons are lost for physical pileup, 2 more for geometrical effects.

Source	IN	CCD	Readout		EPCE		Reconstruction	
mCrab	input photons	detected events	pileup[1]	read events	below low thr	taken events	rejected events[2]	recons. photons
10	13154	20839	701	20127	3966	16161	1284	9646
5	13154	20839	324	20504	4098	16406	676	11242
2	13154	20839	138	20693	4152	16541	294	12364
1	13154	20839	62	20765	4178	16587	132	12776

1: pileup counts only physical pileups

2: rejected events refer to those classified as "unknown patterns" or triggering the guard ring clause (not isolated).

3.2.8 pn Large Window mode

Not simulated yet.

3.2.9 pn Small Window mode

This mode presents a slight problem, i.e. a non-negligible smearing. Events falling in the window while the window is read out or stacked appear to have wrong coordinates and are therefore discarded. The fraction of events discarded may be considered as an increased dead time and is given by :

smear **=** **F**(readtime+stacktime,frametime)

In the case simulated, frametime=exposure+readtime+stacktime=4625 µs, readtime=450 µs, stacktime=170 µs, resulting in a 13.4% deadtime. This deadtime must be taken into account when comparing the input spectrum (deducted the dead time) with the reconstructed spectrum.

The limiting rate for this mode can be placed at 40 mCrab (see fig. 10a).

Source	IN	CCD	Readout		EPCE		Reconstruction	
mCrab	input phot^1	detected events^1	pileup	read events	below low thr	taken events	rejected events	recons. photons
10	11391	18046	53	19079	3813	15266	118	11786
25	11391	18046	142	19097	3827	15270	319	11284
40	11391	18046	252	18993	3805	15188	518	10727
50	11391	18046	320	18932	3777	15154	665	10403

1: These numbers are "predictions" based on the subtraction of 13.4% smearing fraction from the actual number of photons and events (reported in 3.2.8 above).

Note that there is an unexpected excess in the residuals at the low end which shows up also at low source intensities (compare fig. 9a, 9b, 10a) . This is probably an artifact of the reconstruction algorithm and must be investigated.

Source	bin size	residual R_{tot}	error $\sigma_{\tau ot}$	S	χ^2	χ^2/DoF	DoF	P
(mCrab)	(eV)	(counts)		(sigma)				Probab.
10	10	395 sic!	±152	2.6 sic!	697.4	0.76	920	0
	100				94.3	0.96	98	0.41
25	10	-113	±150	0.7	692.8	0.74	930	0
	100				92.8	0.93	99	0.34
40	10	-674	±149	4.5	707.9	0.76	930	0
	100				129.9	1.31	99	1
50	10	-997	±148	6.8	759.6	0.82	929	1.4e-5
	100				168.1	1.71	98	1

3.2.10 pn Timing mode

This mode is nominally capable of supporting very high intensities, up to 800 mCrab, as is shown by the table and by fig. 10b, bottom).

Source	bin size	residual R_{tot}	error $\sigma_{\tau ot}$	S	χ^2	χ^2/DoF	DoF	P
(mCrab)	(eV)	(counts)		(sigma)				Probab.
10	10	42	±162	0.3	756.2	0.80	946	1.4e-6
	100				89.6	0.89	100	0.24
400	10	-309	±161	1.9	772.6	0.81	945	1.2e-5
	100				103.2	1.03	100	0.6
800	10	-736	±160	4.6	816.4	0.86	946	9.3e-4
	100				140.8	1.40	100	0.995

Note that there is an unexpected excess in the residuals at the low end which shows up as a flat plateau at high source intensities.

The following table reports the usual event budget :

Source	IN	CCD	Readout		EPCE		Reconstruction	
	input	detected		read	outside	taken	rejected	recons.
mCrab	photons	events	pileup[1]	events	thresh	events	events	photons
10	13154	20839	4045	16792	1909	14883	0	13196
40	13154	20839	4041	16796	1911	14885	1	13195
400	13154	20839	4177	16651	1895+3	14753	58	12847
800	13154	20839	4337	16459	1842+4	14613	135	12423

[1] The majority of the indicated pileup events are "good" pileups (i.e. vertically split events are coadded by the readout algorithm - which integrates in 20-pixel-high slices), namely for 10 mCrab all, for 40 mCrab 4038, for 400 mCrab 3973 and for 800 mCrab 3935 are good.

3.2.11 pn Burst mode

This mode appears to have virtually no limiting rate (I have used sources up to 1.6 Crab without significant effects, see tables below and fig. 11ab).

Source	bin size	residual R_{tot}	error $\sigma_{\tau ot}$	S	χ^2	χ^2/DoF	DoF	P
(mCrab)	(eV)	(counts)		(sigma)				Probab.
10	10	-2	±45	0.04	241.2	0.29	821	0
	100				33.35	0.33	100	0
100	10	-43	±45	1	252.0	0.30	819	0
	100				40.2	0.40	99	0
400	10	-44	±45	1	242.7	0.29	822	0
1600	10	-87	±44	2	239.3	0.29	820	0
	100				43.6	0.43	100	1.7e-7

The event budget table is reported below. It is slightly different from the other modes due to the fact that the Burst mode has a duty cycle < 100%.

The following problems are also related with the usage of this mode : for a proper operation the source should be located at the "top" of the chip, i.e. farthest from the output register end. It has been actually proposed to locate the source *across* **two** chips read out in opposite directions (i.e. at Y=200), which should make use of the theoretical duty cycle of 5%. In this simulation the source was located at Y=185 (so that the PSF is contained entirely in a **single** chip, which is simpler to simulate), which gives an *effective duty cycle* empirically determined to 7.85%. The effective duty cycle is a function of position, and it has been estimated by counting the number of individual photons (giving rise to one or more events) accepted after readout and is reported in figure 12.

Source	IN	Readout			EPCE		Reconstruction	
mCrab	input phot[1]	rejected events[2]	pileup	read events	below low thr	taken events	rejected events	recons. photons
10	1033	19186	0	1648	339	1309	0	1031
100	(1007)	19232	0	1569	329	1240	0	990
400	(1013)	19077	2	1590	312	1278	5	989
1600	(1040)	18573	13	1671	358	1313	29	945

1: These numbers are "predictions" based on the measurement of the "effective duty cycle" of 7.85 % measured at low intensity, or (values in parentheses) the number of photons corresponding to the read events (for the actual number of photons and events see 3.2.8 above).

2: this is the number of events rejected in the inactive part of the duty cycle.

4 Cosmic ray rejection

4.1 Procedure

I have generated a reference file (at post-CCD level) with "medium solar activity" cosmic rays for 200 s on a representative chip (in the case of MOS it had to be done separately for the image (A) and store (B) sections, which were then merged). The characteristics of the reference files are given below.

These reference files are passed through the simulation of the readout modes and of the EMCE/EPCE as indicated above in 3.1. This way one gets an indication of the rejection efficiency by the on-board algorithm (in the case of MOS these algorithm also do the split event reconstruction; in the case of the pn there is no need or sense in applying the separate reconstruction program - possibly run on the ground - to these data).

4.2 Results

All data refer to a time interval of 200 s.

CCD and mode	IN input particles	CCD detected events	Readout		ExCE and/or ExDH		
			pileup[1]	read events[2]	rejected events[3]	outside thresh.[3]	residuale vents
MOS A	1862	33398					
MOS B	559	19313					
total		52711					
FS			8	33023	2376	266	2
W			0	919	70	6	2
RFS			0	1967	150	7	1
T			2178	1020	150	84	11
pn	931	4532					
FF			2	4513	2809	251+182	16
W (sm)			0	478	470	9+14	7
T			1840	2687	-	69+2369	249
B			0	2120	-	120+1608	392

1: pileup include physical pileups in the chip, as well as "good" pileups in the output register (for the Timing modes), and are anyhow not very meaningful in the case of pure cosmic ray fields.

2: the number of read events may notably differ from the number of events at CCD level because of various mode-dependent effects (e.g. smearing, presence of a window, duty cycle less than 1, etc.)

3: rejected events for EMCE+EMDH means the events discarded by the pattern recognition as unknown patterns or triggering the guard ring - surrounding events are implicitly discarded and are not counted. Events remaining with reconstructed energy above the high threshold are flagged as "outside thresholds" and are rejected later.
For pn in imaging modes rejected events are those at the overflow level - which cause further rejection in their surroundings. Events "outside threshold" are indicated separately for those below the low and above the high additional threshold.
For pn non-imaging modes, no rejection is applied besides the standard thresholding.
The low and high thresholds are fixed in both MOS and pn cases to 25 and 2750 electrons (the overflow threshold for pn is 6793 electrons), corresponding respectively to about 0.1, 10 and 25 keV.

The cosmic ray rejection efficiency of the EMCE (or EMCE-like for the Timing mode, where it is not yet decided whether EMCE or EMDH does it) algorithm is very good (it must be remembered that at the same time this algorithm does split event reconstruction). The residual load on EMDH to do rejection of events above the high threshold is quite limited.

The proposed EPCE algorithms for the imaging modes (see 2.1.4 c-d) is also reasonably efficient and is anyhow better than the plain thresholding (see 2.1.4 e) used for the other modes. As an indication usage of algorithm e in the FF or W case will let through respectively 825 and 103 events instead of 16 and 7.

In the case of the pn Timing and Burst modes the EPCE algorithm (2.1.4 e) is not very efficient (a residual rate of 1-2 event/s is left). It might be suggested to use an algorithm analogous to the one used in the Window mode (2.1.4 d), i.e. reject entire "frames" at the expense of a time-dependent dead time.

In the case of the Burst mode it must also be noted that the number of cosmic rays read out is quite large, notwithstanding the nominal duty cycle of 5%. In fact, as shown in 3.2.11 the effective duty cycle is a function of Y, being higher at lower Y, and in the case of a diffuse event field must be "averaged" all over the chip.

5 Split events

As a reference on event splitting, we report here some simplified statistics based on simulation with monochromatic lines (10000 photons uniformly distributed over the chip surface).

5.1 MOS

Line	E	No lower threshold			Lower threshold 25 e-		
	(eV)	Single	1-2	1-4	Single	1-2	1-4
C	282	97	100	100	99	100	100
O	523	94	99	100	98	100	100
F	677	91	99	100	97	100	100
Cu-L	928	87	98	100	94	99	100
Al	1483	74	97	~100	86	99	100
Si	1740	68	95	~100	81	98	100
P	2015	88	99	100	93	99	100
Ti	4510	62	92	99	75	97	~100
V	4952	57	89	98	71	95	~100
Fe	6403	41	76	92	56	86	96
Cu	8047	33	64	84	47	78	92

The table is made for analogy with the one reported in section 3.6 of [Ref. 11]. These results are reasonably similar with the experimental measurements (taking into account that the exact parameters of the measured chip are not reported in [Ref. 11], and that the simulated line is *perfectly* monochromatic).

5.2 pn

For reference I have done a similar, though more limited, run of checks for the pn :

Line	E	No lower threshold			Lower threshold 25 e-		
	(eV)	Single	1-2	1-4	Single	1-2	1-4
C	282	57	94	100	92	100	100
O	523						
F	677	53	93	100	80	100	100
Cu-L	928						
Al	1483	56	94	100	76	100	100
Si	1740						
P	2015	49	91	100	70	100	100
Ti	4510	55	94	100	71	100	100
V	4952						
Fe	6403	60	94	100	74	100	100
Cu	8047	65	96	100	76	100	100

6 Data and bit rates

6.1 Definitions

The overall bit rate from one EPIC camera can be considered as :

$$R = N_T * R_T + N_B * R_B$$

where N_T is the number of chips containing the target (this is usually 1, except for the case of the pn Large Window and possibly Burst modes) and N_B is the number of remaining chips being used (6 for MOS, 11 or none for pn), as given in section 2.1.5.
As an approximation it is assumed that the background rates are the same in all chips (contrary to [Ref. 2] where an attempt to consider the different sky coverage of outer chips was made).
The individual raw chip rates (neglecting header of frame overheads) are given by :

$$R_T = b * (S_T+S_C+S_B)$$
$$R_B = b * (S_C+S_B)$$

where b is the bit/event allocation of the chosen packing mode, and S_T, S_C and S_B are the event rates due to target, cosmic rays and X-ray background respectively.

The target event rates S_T can be scaled from the source rates I_T (in mCrab for a Crab-like spectrum), provided they are less than the limiting rates as follows :

$$S_T = d\ g\ k\ I_T$$

where d is the duty cycle (1 except for the MOS Refresh Frame Store, and for the pn Burst modes), g is 1 for the no-grating case and 0.43 for the grating case, and k is an empiric factor with values k=12.4 (MOS) or k=16.5 (pn).

The cosmic ray event rate S_C can be estimated from the simulations taking into account the various rejection methods used as function of the operating mode. The event rates are obtained dividing by a factor 200 the last column in the table in section 4.2

The X-ray background event rate S_B can be estimated *conservatively* taking the values in [Ref..2] (which refer to an effective area higher than the true one, and are therefore in excess), and neglecting the different sky coverage of the various chips, i.e. approximately 2 event/s for the MOS case and 1.3 events/s for the pn case.

6.2 Results

The various event rates, also as function of source intensity, are tabulated here below, separately for the MOS and pn cases (for the pn case no on-board reconstruction of split events is assumed; if done, it could reduce the rates by 20-40%).

I_T	g=	1	0.1	0.5	1	5	10	20	50	70	100
(mCrab)	g=	0.43	0.23	1.16	2.32	11.6	23.2	46.4	116	162	232
MOS Mode	S_B	S_C	S_T								
FS1	2	0.01	1.2	6.2	-	-	-	-	-	-	-
FS2	2	0.01	1.2	6.2	12.4	-	-	-	-	-	-
W	2	0.01	1.2	6.2	12.4	62	-	-	-	-	-
RFS	2	0.005	0.06	0.3	0.6	3.0	6.1	12.2	30.6	-	-
T	2	0.055	1.2	6.2	12.4	62	122	244	610	854	1220

I_T	g=	1	0.5	2	10	40	100	200	500	800	1600
(mCrab)	g=	0.43	1.16	4.6	23.2	92.8	232	464	1160	1856	3712
pn Mode	S_B	S_C	S_T								
FF	1.3	0.08	8.3	33	-	-	-	-	-	-	-
Large W	--	--									
Small W	1.3	0.035	8.3	33	165	660	-	-	-	-	-
T	1.3	1.245	8.3	33	165	660	1650	3300	8250	13200	-
B	1.3	1.96	0.65	2.6	13	52	130	259	648	1036	2072

The next table gives the total bit rate for an *entire MOS camera*, according to the configurations presented in 2.1.5 (i.e. one chip in any mode and all other chips in basic FS mode via 1 node). Two tentative possibilities of data packing are presented. In one case (indicated as D1 and D3) the original bit/event allocation is mantained, in the other (indicated as D2 and D4) the bit/event allocation is reduced (note that use of D4 in conjunction with timing mode is assumed here to go with D2 in conjunction with basic FS mode in outer chips).

I_T	g=	1	0.1	0.5	1	5	10	20	50	70	100
(mCrab)	g=	0.43	0.23	1.16	2.32	11.6	23.2	46.4	116	162	232
Mode	b		kbit/s								
FS1/D1	80		1.2	1.6	-	-	-	-	-	-	-
FS1/D2	40		0.6	0.79	-	-	-	-	-	-	-
FS2/D1	80		1.2	1.6	2.07	-	-	-	-	-	-
FS2/D2	40		0.6	0.79	1.03	-	-	-	-	-	-
W/D1	80		1.2	1.6	2.07	5.94	-	-	-	-	-
W/D2	40		0.6	0.79	1.03	2.97	-	-	-	-	-
RFS/D1	80		1.1	1.13	1.15	1.38	1.66	2.21	3.89	-	-
RFS/D2	40		0.55	0.56	0.58	0.69	0.83	1.11	1.94	-	-
T/D3	32		1.05	1.2	1.39	2.94	4.88	8.76	20.4	28.1	39.8
T/D4	22		0.54	0.65	0.78	1.85	3.18	5.84	13.8	19.2	27.2

The next table gives the total bit rate for an *entire pn camera*, according to the configurations presented in 2.1.5 (for T and B mode only one chip active; for the other modes the 11 "non-target" chips are assumed in FF mode; in particular for the Small Window case it is neglected the fact that the 11 FF chips should operate with a different exposure time than usual).

I_T	g= 1	0.5	2	10	40	100	200	500	800	1600
(mCrab)	g= 0.43	1.16	4.6	23.2	92.8	232	464	1160	1856	3712
Mode	b					kbit/s				
FF	32	0.78	1.55	-	-	-	-	-	-	-
Large W	32									
Small W	32	0.77	1.55	5.67	21.1	-	-	-	-	-
T	32	0.34	1.11	5.24	20.7	51.6	103.2	258.3	413	-
B	32	0.12	0.18	0.51	1.72	4.15	8.2	20.3	32.5	64.9

The above tables are also presented in graphical form in Fig. 13 and 14. When the table presents different possibilities of bit/event, the highest one has been chosen for plotting. Note also that in the case of imaging modes (like Frame Store by 1 and 2 nodes and Window for the MOS) which, when they are all allowed by the pile-up, give the same rate, only the "lowest ranking" mode is shown to avoid cluttering the plot.

6.3 Some considerations

The limiting rate for the MOS case is *rather low* (the only possibilities of studying sources stronger than 50 mCrab is the use of the timing mode, or of the refresh frame store option with an *unusually low* duty cycle - or by use of a window in refresh frame store, and in no case one can get to a source stronger than 100 mCrab). Actually those values should be doubled if one considers the presence of the grating in the beam. The limitation is therefore not so severe, if one considers the paucity of sources that strong.

The situation is different for the pn case (particularly if one considers the pn camera being used without grating in the beam), since the timing and burst mode allow to observe sources of *virtually unlimited strength*.

However, considering again the small number of sources stronger than 50-100-200 mCrab, one could wonder whether such strong sources should not be *at all excluded from the scientific goals* of EPIC.

In the MOS case one notes that all imaging mode options (even if one transmits the full 80 bit/event) always remain *well within* the "equal" allocation of 16 kbit/s per camera.
Only the timing mode (at the nominal allocation of 32 bit/event) exceeds this value for about 40 mCrab (without grating, 93 mCrab with grating). If one reduces the bit allocation to 22 bit/event (renouncing all spatial information) the equal allocation is exceeded at 55 mCrab (127 mCrab with grating).
One might think to operate in timing mode, direct transmission, for stronger sources by allowing the bit rate from one camera to exceed 16 kbit/s. This is indeed possible remaining within the overall EPIC allocation up to the MOS limiting rate. Of course it does mean that the other cameras shall be off.
Therefore any need for an **indirect mode** (on-board accumulation of spectra or time profiles) will be needed only (in conjunction with the timing mode) for strong sources if one still wants to use all three cameras at the same time.

In the pn case (without grating) one remains within the "equal" allocation of 16 kbit/s up to 30 mCrab (window and timing modes) or 400 mCrab (burst mode).
Note that the strongest sources which can be observed with the pn camera are well above the limiting rate for the MOS camera. One might therefore consider to have the MOS cameras off (or could one even think to mask out the target and transmit only the "serendipitous" part of the field of view ?), and to the dedicate the bulk of the EPIC telemetry to the pn camera. With 40 kbit/s one can reach 75 mCrab in timing mode or 1000 mCrab in burst mode.
Once again, the need for an **indirect mode** (on-board accumulation of spectra or time profiles) exists essentially only in the case of the timing mode when one wants to study sources stronger than 75 mCrab.

6.4 About headers and trailers

All the above considerations neglect all kind of overheads (which should be defined in detail at a later design stage), namely :

- packet telemetry overheads (inclusive of Source Packet Headers)
- overheads due to the content of the Data Field Header (both the parts mandated by ESA and the eventual additional information added by EPIC). This overhead of H bytes reduces the part of the Source Packet useful to contain event data. If each event requires b bits, the number of events in a Source Packet is :

  $$n = (512-H)*8 / b$$

  For the rest this overhead is independent on the event rate
- data are assumed to be transmitted separately by chip, while instead in the case of the MOS when reading out from 2 nodes, EMCE passes to EMDH data by node. This *has been neglected* (but it could affect the reasoning about frames below, typically one will have half the number of event/frames if the source is in the middle of the chip).
- overheads due to the handling of the frame headers and trailers present in the protocol between ExCE and ExDH. This overhead depends somehow on the event rate (namely on the number of event/frames) and is *higher for low source rates*.

  In a maximal case (both header and trailer present and transmitted to ground unchanged for any non-empty frame) it amounts to 2*b bits/frame (assuming both header and trailer use the same number of bits as 1 event, as in the MOS case).

  In a minimal case some redundant information (mode configuration) could be checked at ExDH level and not transmitted (or included once in a packet only in the Data Field Header), some other (e.g. counters) could be moved to some HK packets, and only a frame time tag remains to be transmitted for each frame. This time tag is likely to be at least 32 bit long.

  In addition a mechanism to separate in a clear way the events of a frame from those of the next must be considered. The simplest case (which however requires sequential interpretation of the packet for decoding) is to use one flag bit in the time tag itself. Another possibility would be to provide a counter of events for each frame.

Note that the number of events/frame is quite low (the highest values are, for a source at the limiting rate, 33 events/frame for the MOS RFS option and 8 events/frame for the pn Burst mode, but it is not unlikely that in most cases one has less than 1 event/frame, which essentially adds the header-trailer overhead *once for each event*).

The effect of this (assuming a 32 bit overhead) is to **double the bit rate** in the case of the MOS Timing mode and in all pn modes.

In fact the effect on the MOS timing mode depends on the *definition of "frame"* (I have used a frame time equal to the pixel exposure time, if a frame is defined as a larger number of exposures, e.g. 10, the effect becomes much smaller e.g. an overhead of 3.5 kbit/s for any source strength greater than 9 mCrab, and scaled in proportion for weaker sources; if the frame is 100 exposures the effect is 0.35 kbit/s for any source stronger than 0.9 mCrab, etc.). Remember that a time counter within the "frame" is present with each individual event.

In the case of the pn the effect is unavoidable due to the s*hort duration of the frames*. E.g for the window mode one has more than 1 event/frame for sources stronger than 15 mCrab (which gives an overhead of 6.7 kbit/s), for the Burst mode this occurs for sources stronger than 200 mCrab (overhead of 7.8 kbit/s), and for the Timing mode this occurs for sources stronger than 170 mCrab (with an overhead of 77 kbit/s **sic !** i.e. **virtually always doubled !**).

A possibility to overcome this would be either to reduce the number of bits allocated to the time tag (or frame counter).

Another possibility (feasible perhaps in the case of the timing mode) would be to define a "mega-frame" (multiple of the natural frame), to place a time tag only for the mega-frame, and to place a frame counter with a reduced number of bits with each event (in the case of the pn timing mode this is already provided in the Y-position field, which ranges 1 to 10, but could easily range 1 to 200, reducing the overhead to 4 kbit/s).

For the other pn modes the introduction of a mega-frame would require a redefinition of the individual event format (adding a "true frame" counter even with a limited number of bits).

As an example of the effect of the overheads, the following table reports the camera bit rates one will have, assuming 32 bit/frame overhead for all modes, and the frame times listed in section 2.1.3 (see here above for a *caveat* about timing modes). The overhead is accounted for only in the chip containing the target source.

I_T	g= 1	0.1	0.5	1	5	10	20	50	70	100
(mCrab)	g= 0.43	0.23	1.16	2.32	11.6	23.2	46.4	116	162	232
MOS Mode	b	kbit/s								
FS1/D1	80	1.21	1.60	-	-	-	-	-	-	-
FS2/D1	80	1.22	1.61	2.10	-	-	-	-	-	-
W/D1	80	1.23	1.78	2.35	6.22	-	-	-	-	-
RFS/D1	80	1.11	1.14	1.17	1.41	1.69	2.24	3.92	-	-
T/D3	32	1.08	1.39	1.78	4.83	8.69	16.4	39.4	54.8	75.7

I_T	g= 1	0.5	2	10	40	100	200	500	800	1600
(mCrab)	g= 0.43	1.16	4.6	23.2	92.8	232	464	1160	1856	3712
pn Mode	b	kbit/s								
FF	32	1.03	2.20	-	-	-	-	-	-	-
Large W	32									
Small W	32	1.01	2.50	10.4	27.9	-	-	-	-	-
T	32	0.57	2.02	9.80	39.0	97.3	183.3	338.2	492.7	-
B	32	0.14	0.26	0.92	3.36	8.25	16.0	28.2	40.3	72.7

The content of the tables is also plotted in Fig. 15 (for the case MOS obstructed by grating, and pn without grating).

7 References

[1] L.Chiappetti, Inputs about ExDH and EPIC system requirements, Issue 1 - 26 Apr 1994 - EPIC-IFC-TN-004

[2] L.Chiappetti, A preliminary assessment of EPIC telemetry rates, Draft 0.1 - 25 Nov 1992 (previous version of this document)

[3] L.Chiappetti, EPOS : EPIC Observation Simulations. Issue 2 - update in progress (to be issued) - EPIC-IFC-TN-006

[4] C.Erd, e-mail of 12 Nov 1993 ; files are available at IFCTR.

[5] L.Chiappetti (ed.), EPIC Operating Modes. Report of the OMWG. Issue 1.A - 31 May 1993 - EPIC-IFC-TN-001

[6] A.Bouere, C.Cara, M.Lortholary, C.Pigot, Description of the EDU architecture and operation, Issue 0.1 - Nov 1993

[7] E.Kendziorra, e-mail of 31 Jan 1994

[8] C.Pigot , Some comments on operating modes, Nov 93.

[9] E.Kendziorra, fax on "pn burst mode", 16 Sep 93

[10] E.Kendziorra, e-mail of 4 Oct 93 (exchange of correspondence with L.Chiappetti, 29 Sep and 8 Oct 93).

[11] A.Holland, Status of the Open Electrode MOS CCDs, 25 Nov 1993