Algorithm for thresholds calculation in Pixirad-1

**Step1: find the actual threshold Vth1 one has to apply to the 'low' discriminator ( in Volt) for the freely set energy threshold Eth1 ( in keV)**

using the relation

1) Eth1= a(Vth1)2 + bVth1

where a and b were carefully determined by calibration with several energies at the synchrotron

a= 36.6 (keV/Volt2) b= 39.3 (keV/Volt)

**Step2: find the best couple (VTHMAXbest, % best) which best realizes the Vth1 value calculated above**.

using the relation

2) Vth1= [(VTHMAX \* 0.000781)-0.6]\*%

Best couple means the one with the highest VTHMAX, but lower than 2200 and higher than 800.

There are 32 possible VTHMAX fractions (%) , with % spanning from 0 to 1 according to the following block-wise linear steps

Block1 0, 0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1

Block2 0.12, 0.14, 0.16, 0.18, 0.20, 0.22, 0.24, 0.26, 0.28

Block 3 0.32, 0.36, 0.40, 0.44, 0.48, 0.52, 0.56, 0.60

Block4 0.7, 0.8, 0.9, 1.0

The finite detector energy resolution and the VTHMAX resolution of 1 count make reasonable to assume for Vth1 a precision of 0.001 V( better than 100eV)

**Step3: translate the best % ( %best) value in the best DAC\_counts value**

**(DAC\_countsbest)**

Inside the ASIC there are 4, 5 bit DACs. The fractions above are internally set using the externally set values of DAC\_counts according to this map :

fractions 0-0.1 are linearly set by DAC \_counts 0-10

fractions 0.12-0.28 are linearly set by DAC \_counts 10-19

fractions 0.32-0.6 are linearly set by DAC \_counts 20-27

fractions 0.7-1 are linearly set by DAC \_counts 28-31

The couple ( VTHMAXbest, DAC\_countsbest) implements the threshold Eth1 for the 'low counter '

**Step4: find the common 32 allowed threshold values Vth2,3,4 (in Volt) for counters 2,3,4 , once the common VTHMAXbest value has been fixed at Step2**

using the relation

3) Vth2,3,4allowed= [(VTHMAXbest \* 0.000781)-0.6]\*%

and the 32 possible values of %, according to the blocks above

**Step5: find the 32 allowed energy thresholds Eth2,3,4allowed**  in keV corresponding to the 32 allowed Vth2,3,4allowed

using the relation

4) Eth2,3,4allowed= a(Vth2,3,4allowed)2 + bVth2,3,4allowed

**Step6: find among the Eth2,3,4allowed values, the one closest to the wanted threshold values Eth2,3,4set and the related %2,3,4closest**

using the relation

5) |min(Eth2,3,4allowed-Eth2,3,4set)|

**Step7: translate  the %2,3,4closest value for which in step 6 the best approximation was found, in DAC\_counts 2,3,4closest**

using the same mapping as in Step3. The couple ( VTHMAXbest, DAC\_counts2,3,4closest)

implements the best approximation to Eth2,3,4 set