RD53A
PARIS CHIP

Le misure fatte da Regina Castrovilli

Stefania Spagnolo

May 4th, 2020

CONFIGURAZIONE - REGINA

- Abbiamo ***modificato*** ed eseguito ***std_digitalscan.json*** con le seguenti configurazioni:
  - Analisi e Histogrammer uguali
  - Loops:
    - Rd53aMaskLoop ()
      - Rd53aParameterLoop
        - Rd53aCoreColLoop (tutte e 50 le colonne di core)
          - Rd53aTriggerLoop (100 trigger per pixel) `"edgeMode": true`

```
{
  "config": {
    "max": 200,
    "min": 200,
    "step": 5,
    "parameter":"InjVcalDiff"
  },
  "loopAction": "Rd53aParameterLoop"
},
```

Un unico valore alla volta di InjVcalDiff

  - Prescan:
    - "InjEnDig": 0,
    - "InjAnaMode": 0,
    - "LatencyConfig": 58,
    - "GlobalPulseRt": 16384,
    - "SyncVth": 500,
    - "LinVth" : 500,
    - "DiffVth1": 500

CONFIGURAZIONE - *STD_DIGITALSCAN.JSON*

https://github.com/Yarr/Yarr/blob/master/configs/scans/rd53a/std_digitalscan.json

- Name: DigitalScan
  - Analisi:
    - OccupancyAnalysis, L1Analysis
  - Histogrammer:
    - OccupancyMap, TotMap, Tot2Map, L1Dist, HitsPerEvent, DataArchiver
  - Loops:
    - Rd53aMaskLoop (1 pixel per core, 64 [max permesso alla volta], step1)
      - Rd53aCoreColLoop (tutte e 50 le colonne di core)
        - Rd53aTriggerLoop (100 trigger per pixel)
- Prescan:
  - "InjEnDig": 1,
  - "InjAnaMode": 0,
  - "LatencyConfig": 58,
  - "GlobalPulseRt": 16384,
  - "SyncVth": 500,
  - "LinVth" : 500,
  - "DiffVth1": 500

```
"edgeMode": true
```

```
51          {
52            "config": {
53              "max": 50,
54              "min": 0,
55              "step": 1,
56              "nSteps": 5
57            },
58            "loopAction": "Rd53aCoreColLoop"
59          },
```

```
{
  "config": {
    "max": 64,
    "min": 0,
    "step": 1
  },
  "loopAction": "Rd53aMaskLoop"
},
```

IL LOOP DI MASCHERA/CORECOL

```
{
  "config": {
    "max": 64,
    "min": 0,
    "step": 1
  },
  "loopAction": "Rd53aMaskLoop"
},
```

```
51          {
52            "config": {
53              "max": 50,
54              "min": 0,
55              "step": 1,
56              "nSteps": 5
57            },
58            "loopAction": "Rd53aCoreColLoop"
59          },
```

Pixel “letti” nell’iter. 0 su mask e 0 su corecol

Pixel “letti” nell’iter. 10 su mask e 0 su corecol

sono coinvolte le corecol 0,5, … 45

Pixel “letti” nell’iter. 0 su mask e 1 su corecol

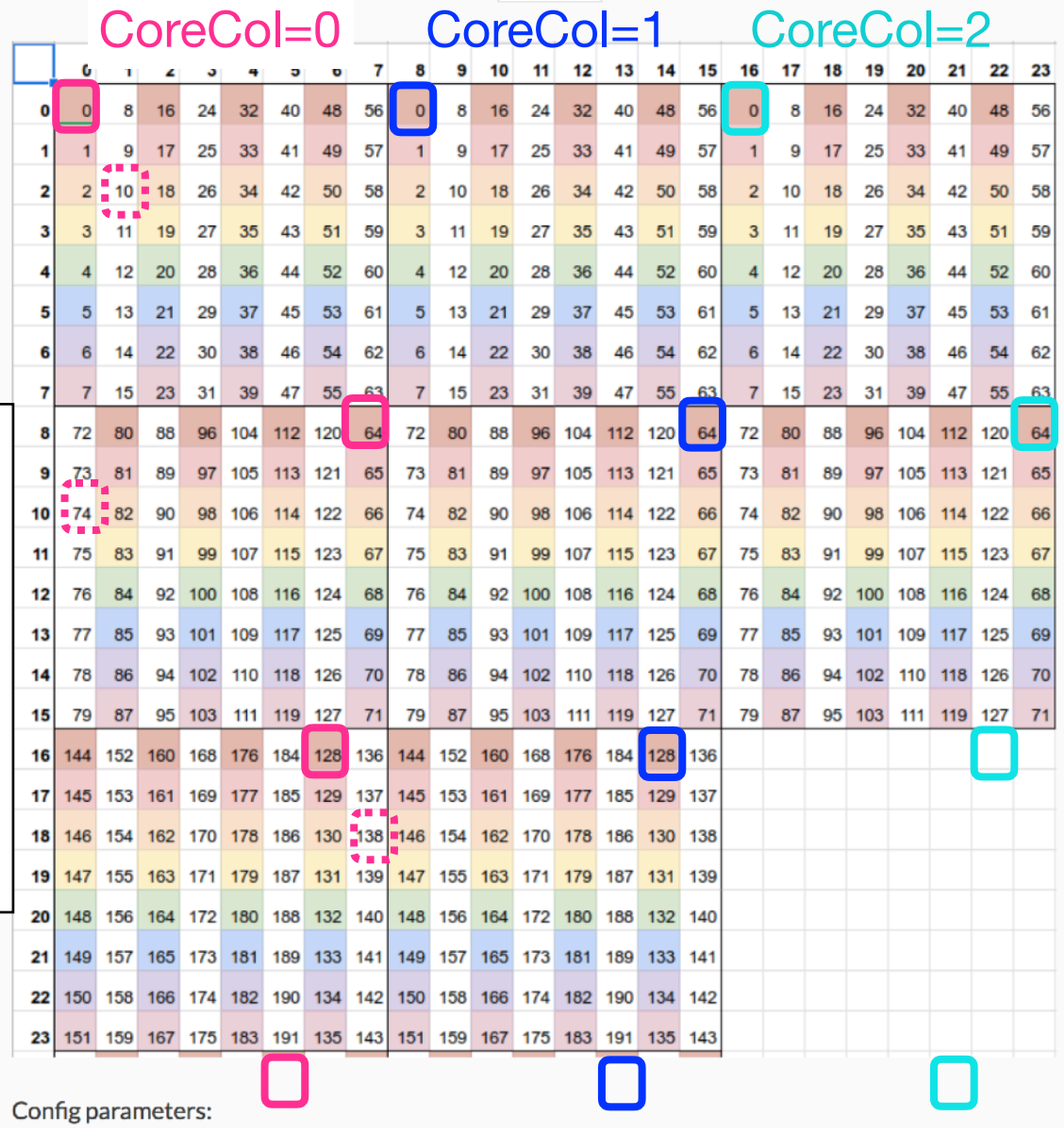
sono coinvolte le corecol 1,6, … 46

Rd53aMaskLoop

Loops over pixels. All pixels in one core column are serialized on the following fashion.

```
unsigned serial = (core_row*64)+((col+(core_row%8))%8)*8+row%8;
```

The maximum of the loops defines how many pixels should be activated at one time. E.g. if the max is 64 that means every 64th pixel (1 pixel per core) and requires 64 steps to loop over all pixels. A pattern of enabled pixels in each mask step for `max = 16` can be found here



Config parameters:

- max `<int>` : number of mask stages
- min `<int>` : mask stage to start with
- step `<int>` : step size of mask stage

IL LOOP DI MASCHERA/CORECOL

```
{
  "config": {
    "max": 64,
    "min": 0,
    "step": 1
  },
  "loopAction": "Rd53aMaskLoop"
},
```

La prima maschera parte da indice 0 (primo pixel = riga zero e colonna minima nella CoreColl) si saltano 64 pixel (nella numerazione seriale codificata); la seconda maschera parte dal pixel separato di 1 (step) nella sequenza ed e' costruita con la stessa logica (pixels: 1, 65, 129, ecc). Step 64 => si legge un pixel per core in una maschera.

```
51        {
52          "config": {
              "max": 50,
              "min": 0,
              "step": 1,
              "nSteps": 5
            },
            "loopAction": "Rd53aCoreColLoop"
59        },
```

Si legge da CoreCol=0, a step di 5 (nstep): Quindi, in questo caso, in una iterazione si leggono pixel (secondo la maschera) in10 CoreCol (0, 5, 10, …); all'iterazione successiva si legge a partire da CoreCol=1 (lo step di CoreCo e' 1, non si salta nessuna colonna di core), Max indica il numero d'ordine della colonna di core piu' alta che si vuole leggere.

Pixel "letti" nell'iter. 0 su mask e iter. 0 su coreco *sono coinvolte le corecol 0,5, … 45*

Pixel "letti" nell'iter. 10 su mask e 0 su corecol *sono coinvolte le corecol 0,5, … 45*

Rd53aMaskLoop

Loops over pixels. All pixels in one core column are serialized on the following fashion.

```
unsigned serial = (core_row*64)+((col+(core_row%8))%8)*8+row%8;
```

The maximum of the loops defines h … ed at one time. E.g. if the max is 64 that means every 64th pixel (1 pi … s to loop over all pixels. A pattern of enabled pixels in each ma … d here

CoreCol=0

CoreCol=5,10,15, … 45

	0	1	2	3	4	5	6	7
0	0	8	16	24	32	40	48	56
1	1	9	17	25	33	41	49	57
2	2	10	18	26	34	42	50	58
3	3	11	19	27	35	43	51	59
4	4	12	20	28	36	44	52	60
5	5	13	21	29	37	45	53	61
6	6	14	22	30	38	46	54	62
7	7	15	23	31	39	47	55	63
8	72	80	88	96	104	112	120	64
9	73	81	89	97	105	113	121	65
10	74	82	90	98	106	114	122	66
11	75	83	91	99	107	115	123	67
12	76	84	92	100	108	116	124	68
13	77	85	93	101	109	117	125	69
14	78	86	94	102	110	118	126	70
15	79	87	95	103	111	119	127	71
16	144	152	160	168	176	184	128	136
17	145	153	161	169	177	185	129	137
18	146	154	162	170	178	186	130	138
19	147	155	163	171	179	187	131	139
20	148	156	164	172	180	188	132	140
21	149	157	165	173	181	189	133	141
22	150	158	166	174	182	190	134	142
23	151	159	167	175	183	191	135	143

	16	17	18	19	20	21	22	23
0	0	8	16	24	32	40	48	56
1	1	9	17	25	33	41	49	57
2	2	10	18	26	34	42	50	58
3	3	11	19	27	35	43	51	59
4	4	12	20	28	36	44	52	60
5	5	13	21	29	37	45	53	61
6	6	14	22	30	38	46	54	62
7	7	15	23	31	39	47	55	63
8	72	80	88	96	104	112	120	64
9	73	81	89	97	105	113	121	65
10	74	82	90	98	106	114	122	66
11	75	83	91	99	107	115	123	67
12	76	84	92	100	108	116	124	68
13	77	85	93	101	109	117	125	69
14	78	86	94	102	110	118	126	70
15	79	87	95	103	111	119	127	71

Config parameters:

- max `<int>` : number of mask stages
- min `<int>` : mask stage to start with
- step `<int>` : step size of mask stage

CONFIGURAZIONE - REGINA

- Abbiamo ***modificato*** ed eseguito ***std_digitalscan.json*** con le seguenti configurazioni:
  - Analisi e Histogrammer uguali
  - Loops:
    - Rd53aMaskLoop ()
      - Rd53aParameterLoop
        - Rd53aCoreColLoop (tutte e 50 le colonne di core)
          - Rd53aTriggerLoop (100 trigger per pixel) `"edgeMode": true`

```
{
  "config": {
    "max": 200,
    "min": 200,
    "step": 5,
    "parameter":"InjVcalDiff"
  },
  "loopAction": "Rd53aParameterLoop"
},
```

Un unico valore alla volta di InjVcalDiff

- Prescan:
  - "InjEnDig": 0,
  - "InjAnaMode": 0,
  - "LatencyConfig": 58,
  - "GlobalPulseRt": 16384,
  - "SyncVth": 500,
  - "LinVth" : 500,
  - "DiffVth1": 500

Dubbi sul setup *Regina*

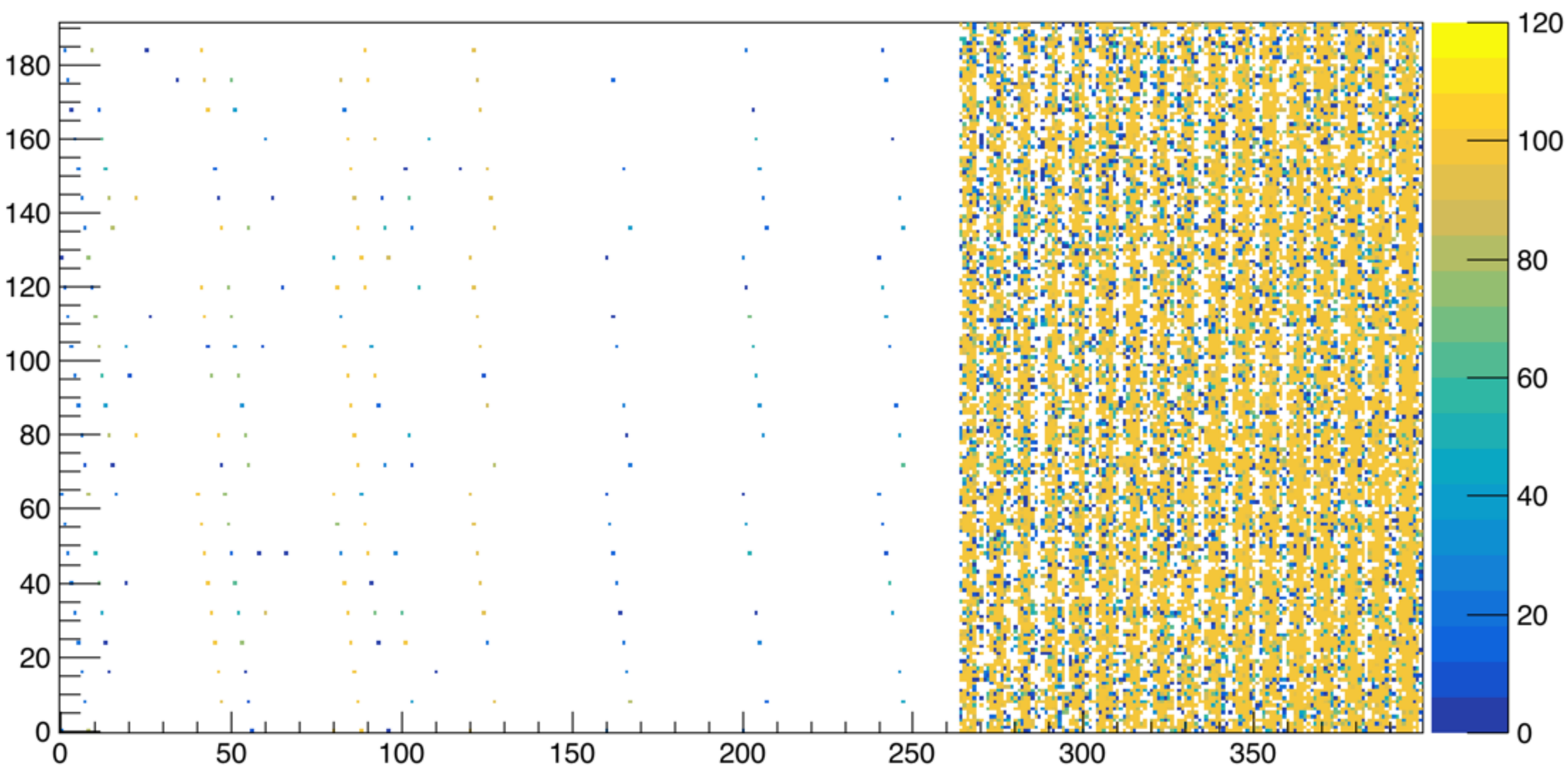
Qual è l'impatto di:
InjEnDig = 0 -> Timon: no digital injection, then analogue inj ?
InjAnaMode = 0 -> ??? Never asked
"edgeMode": true -> Timon: it must be false for analogue injection

I DATI

- Vcal (DAQ parameter InjVCaldiff):
  - 250, 500, 600, 750, 900, 1000, 1200, 1500, 3000
- 1 unita’ di Vcal = 10e

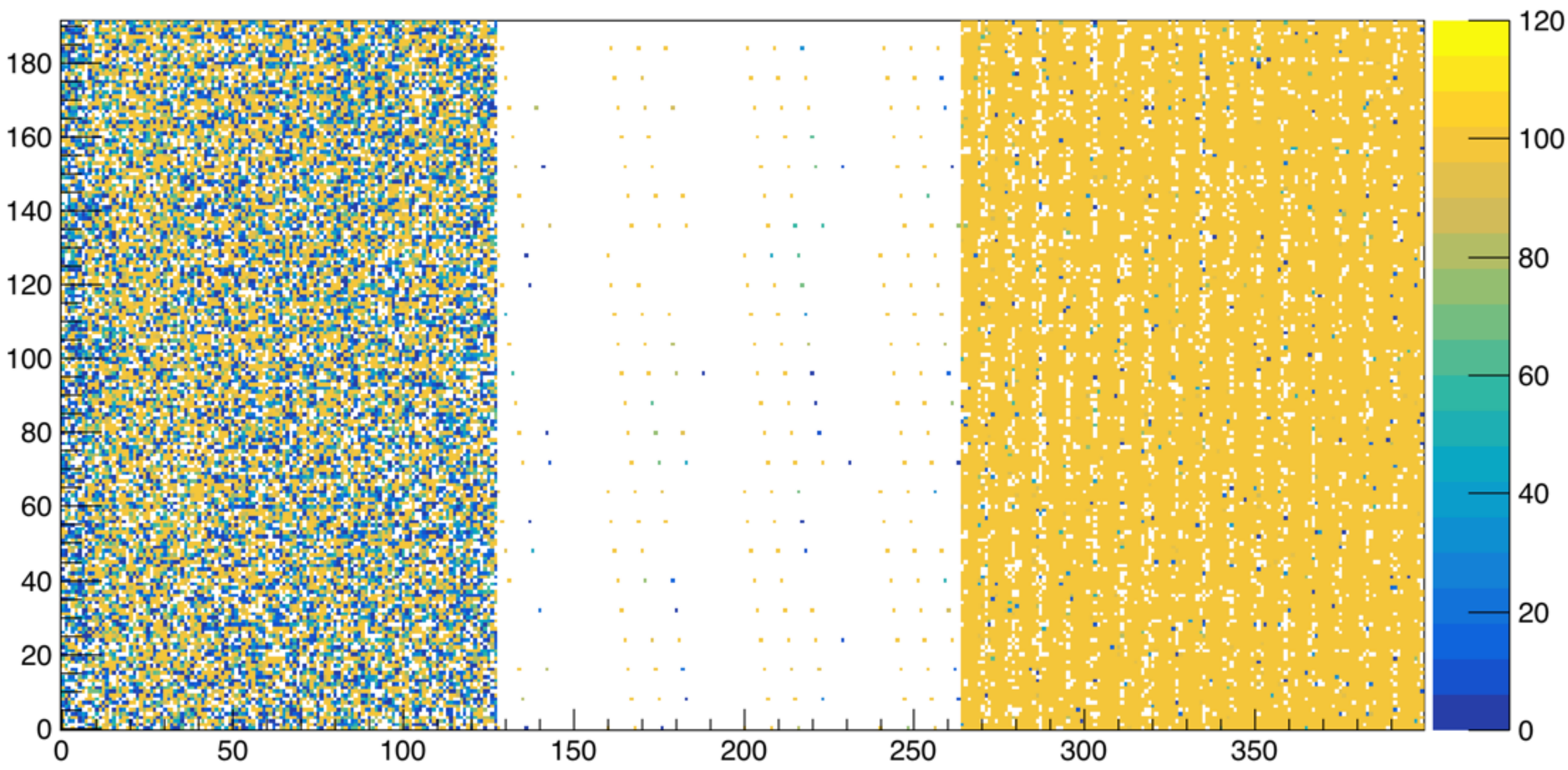
InjVCalDiff = 250 Occupancy Map

Occupancy map



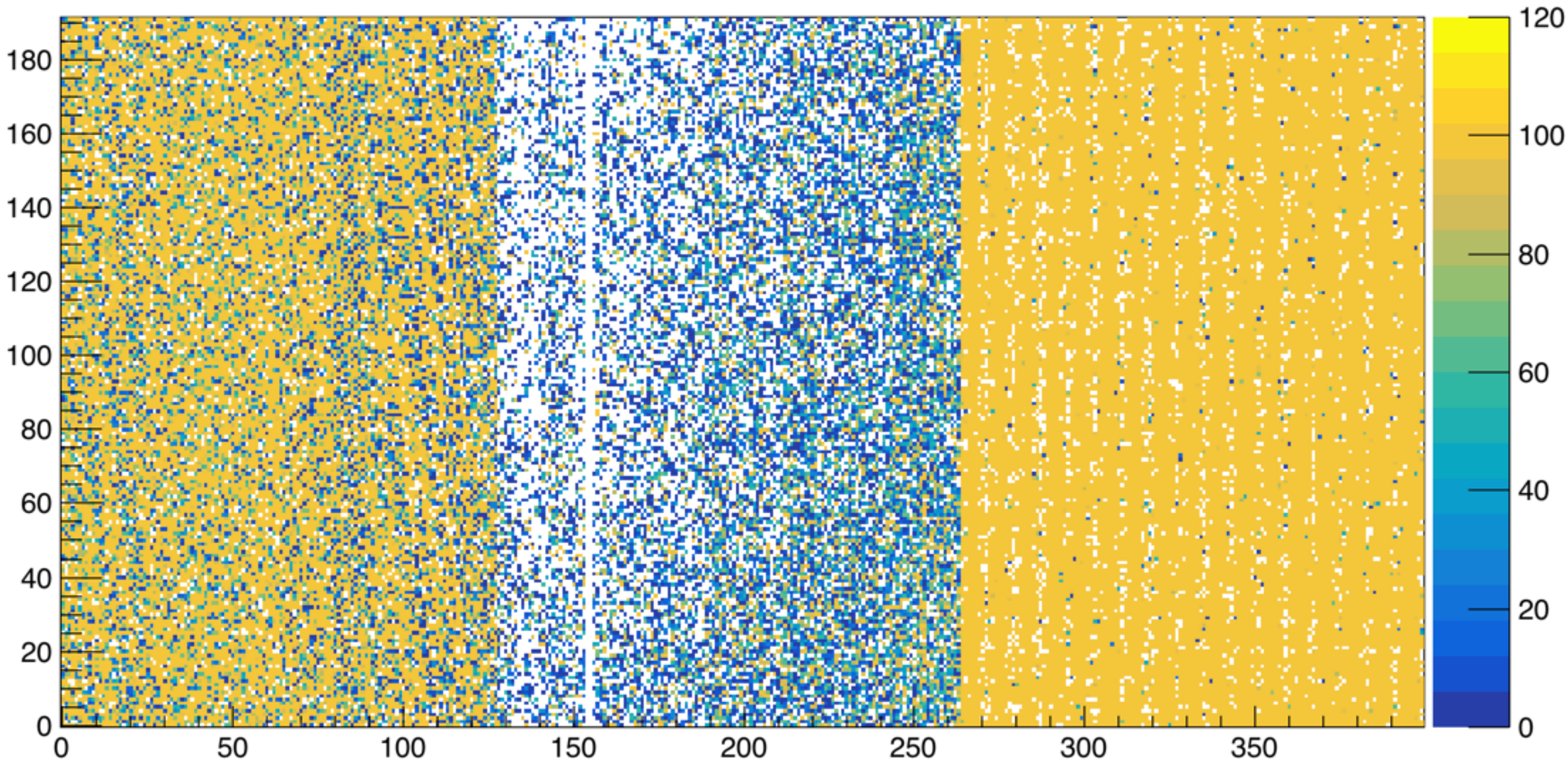
INJVCALDIFF = 500 OCCUPANCY MAP

Occupancy map



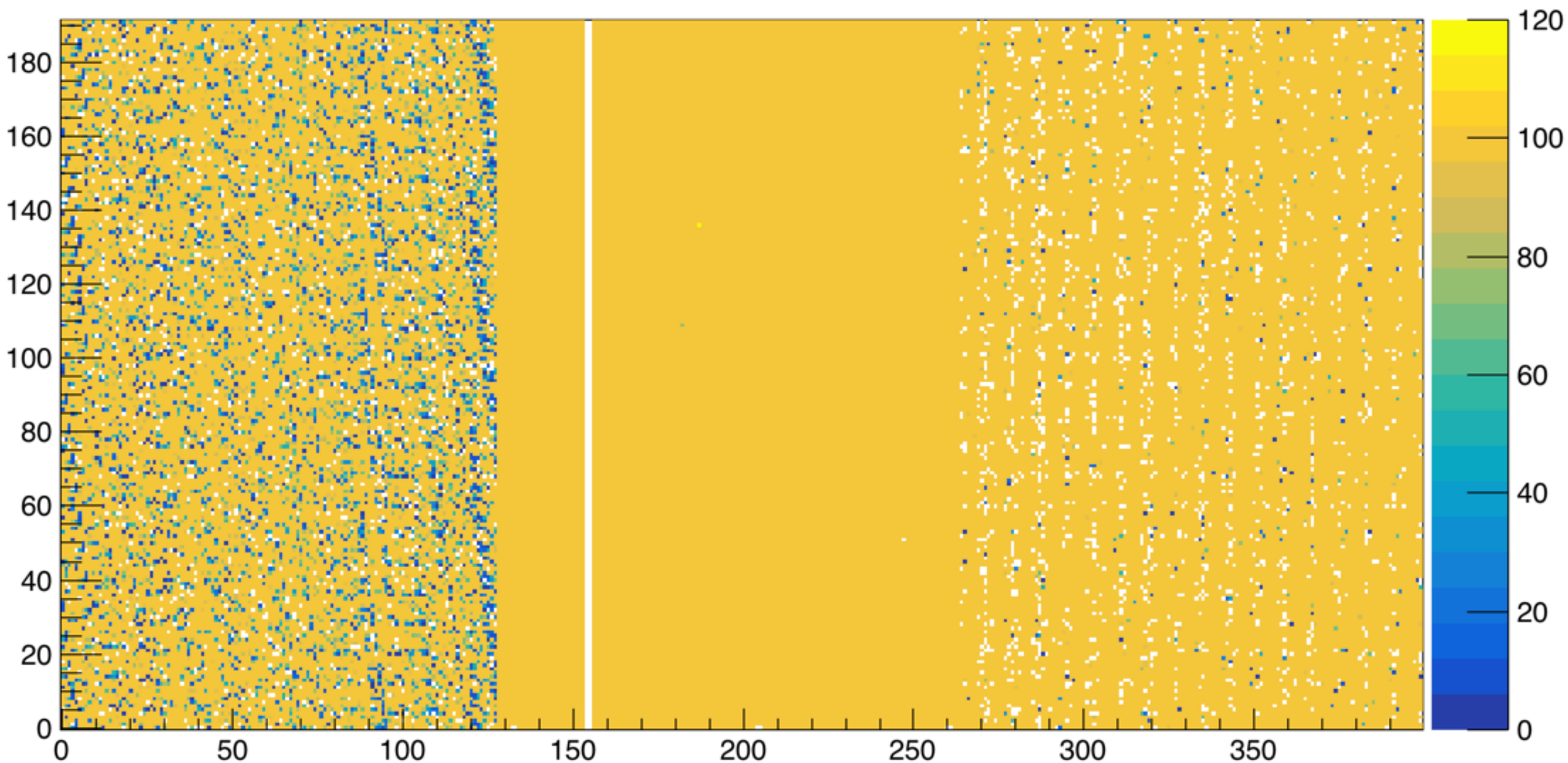
InjVCalDiff = 600 Occupancy Map

Occupancy map



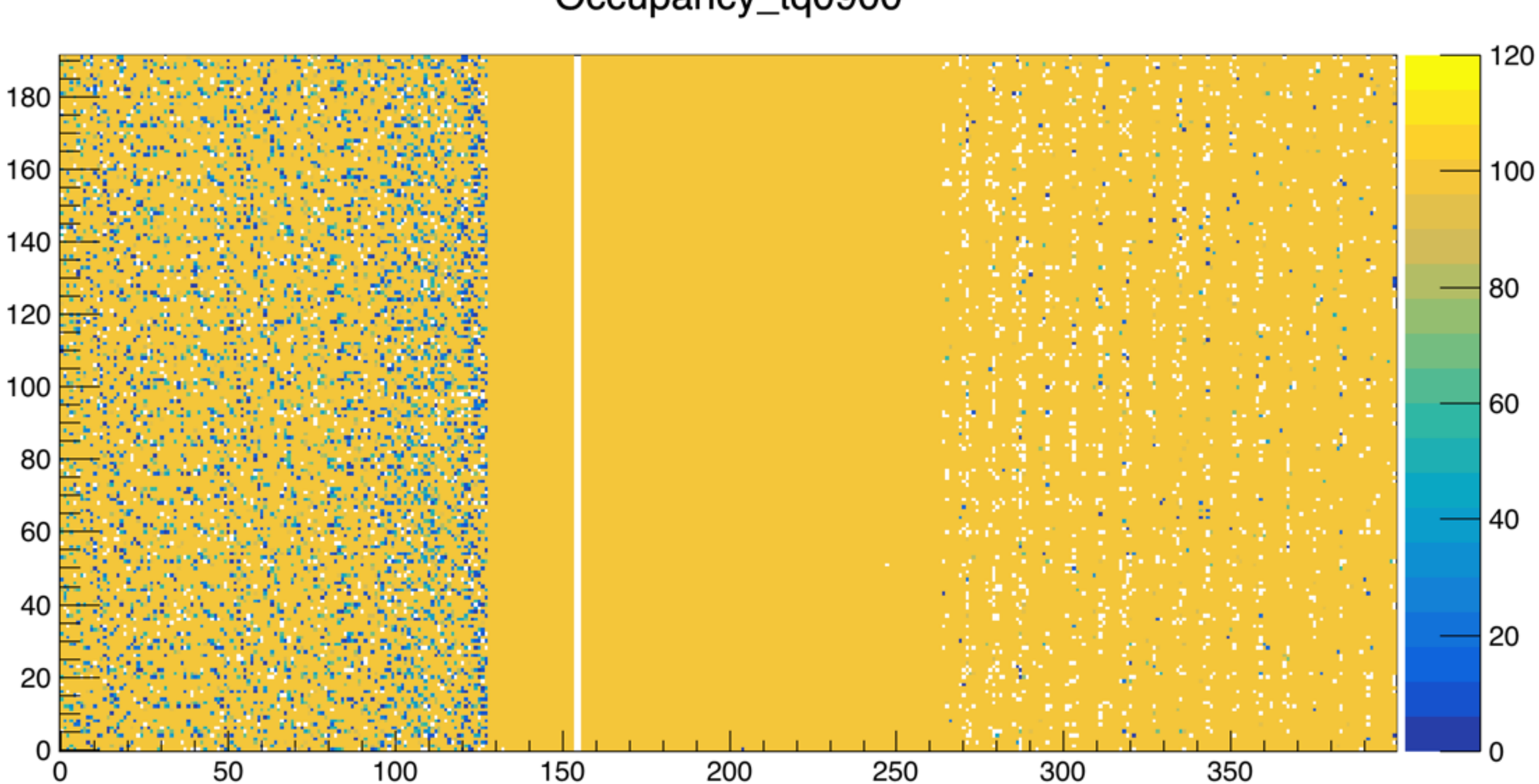
InjVCalDiff = 750 Occupancy Map

Occupancy map



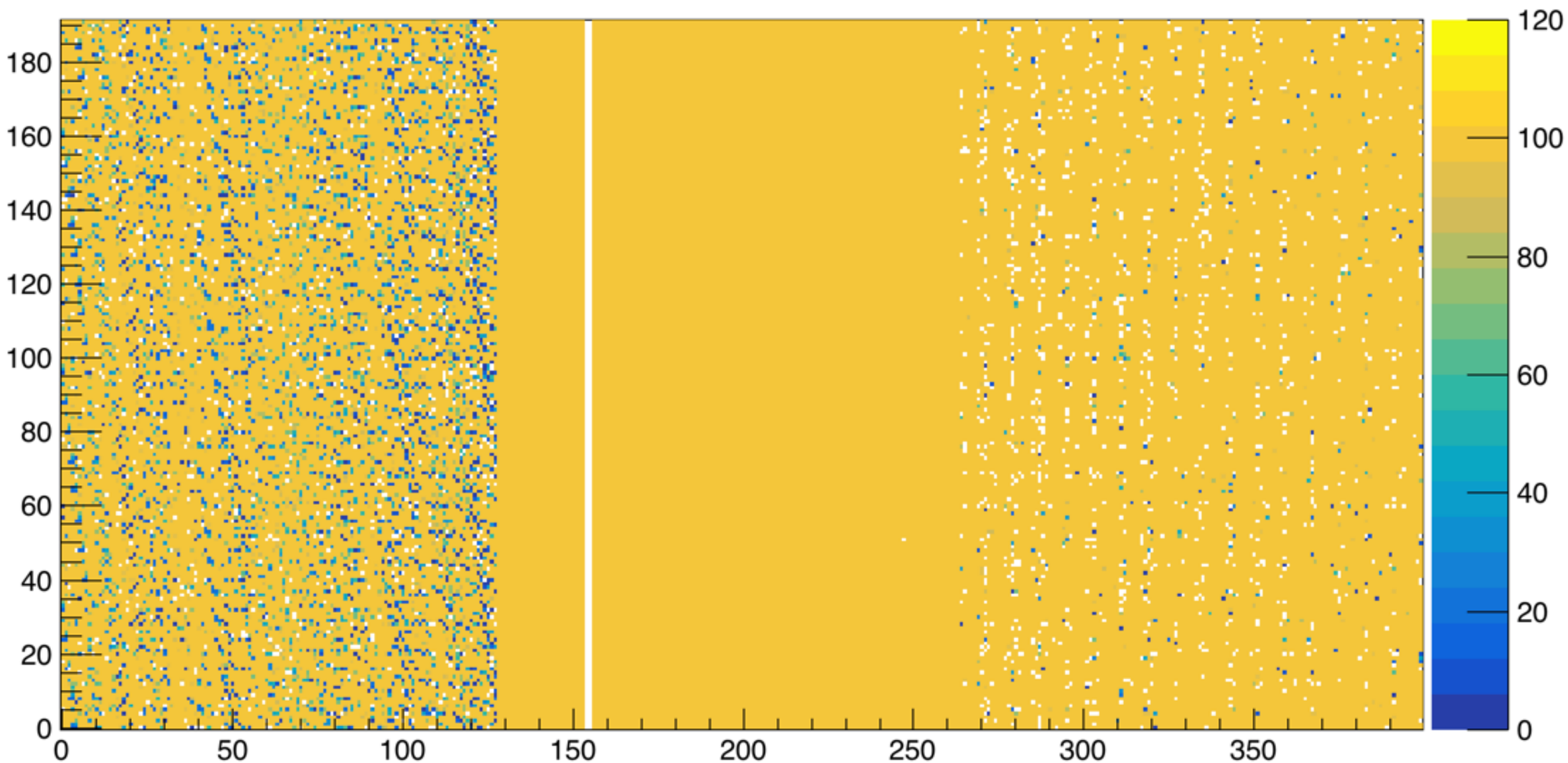
InjVCalDiff = 900 Occupancy Map

Occupancy map



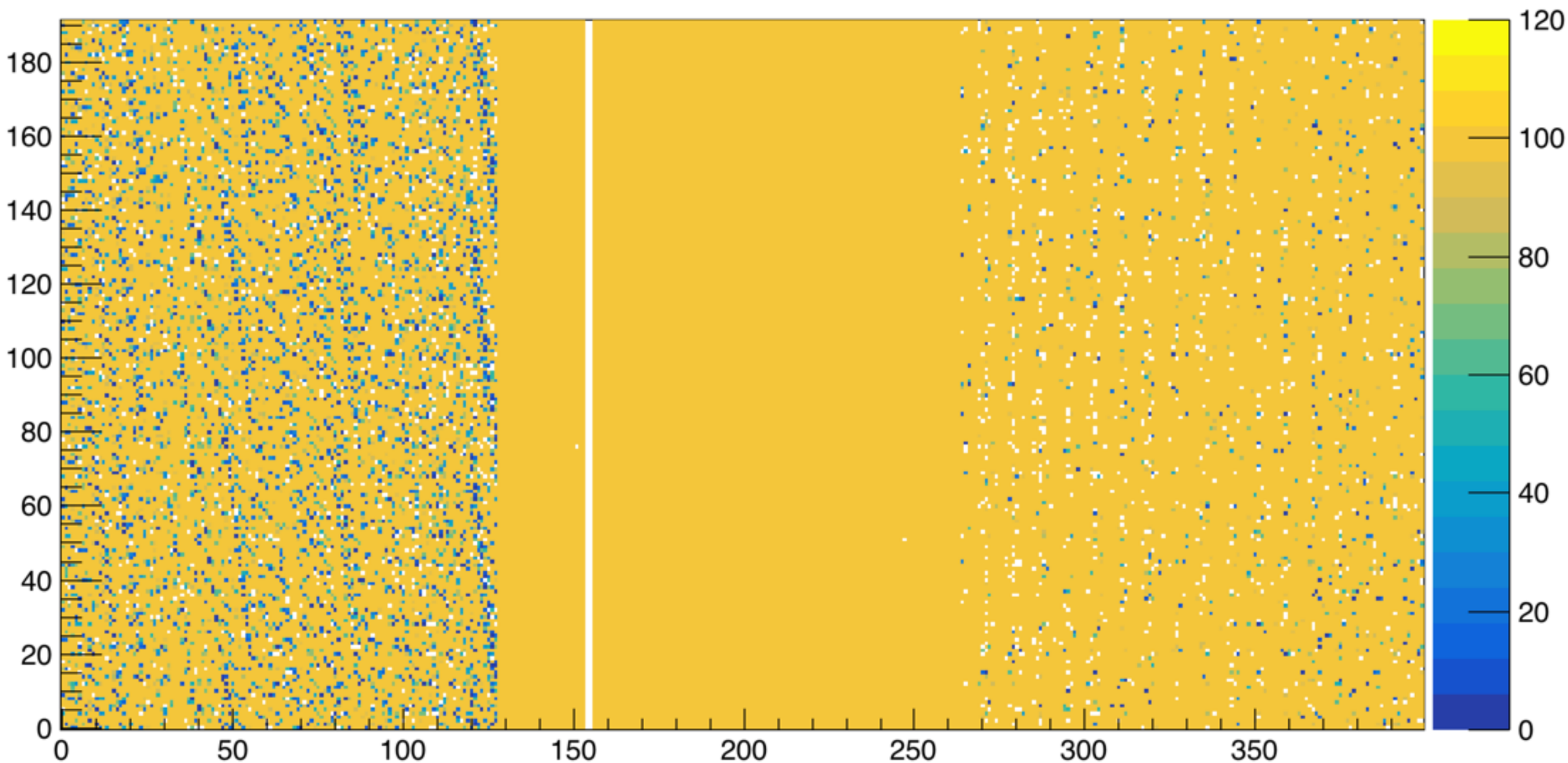
InjVCalDiff = 1000 Occupancy Map

Occupancy map



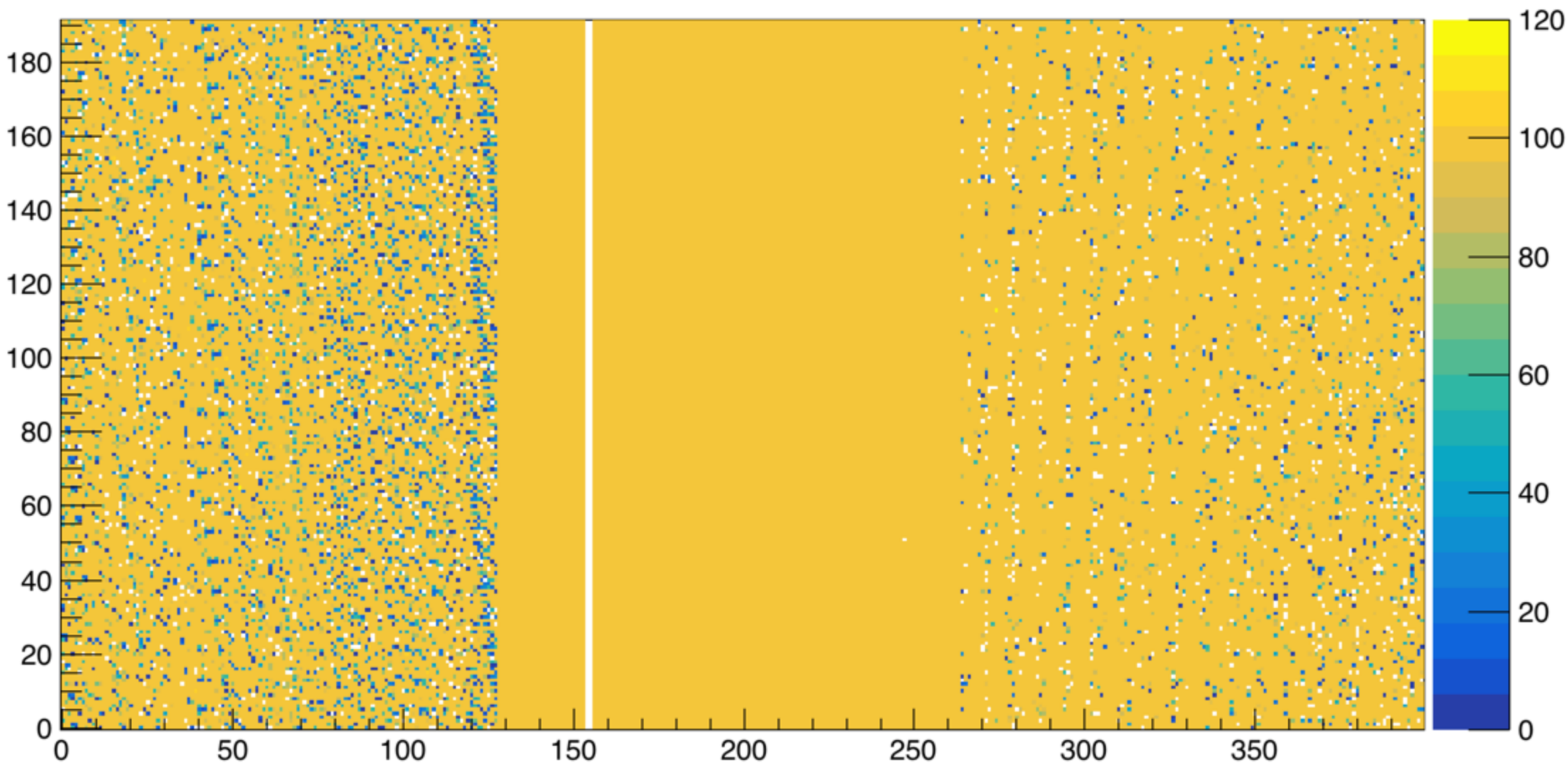
INJVCALDIFF = 1200 OCCUPANCY MAP

Occupancy map



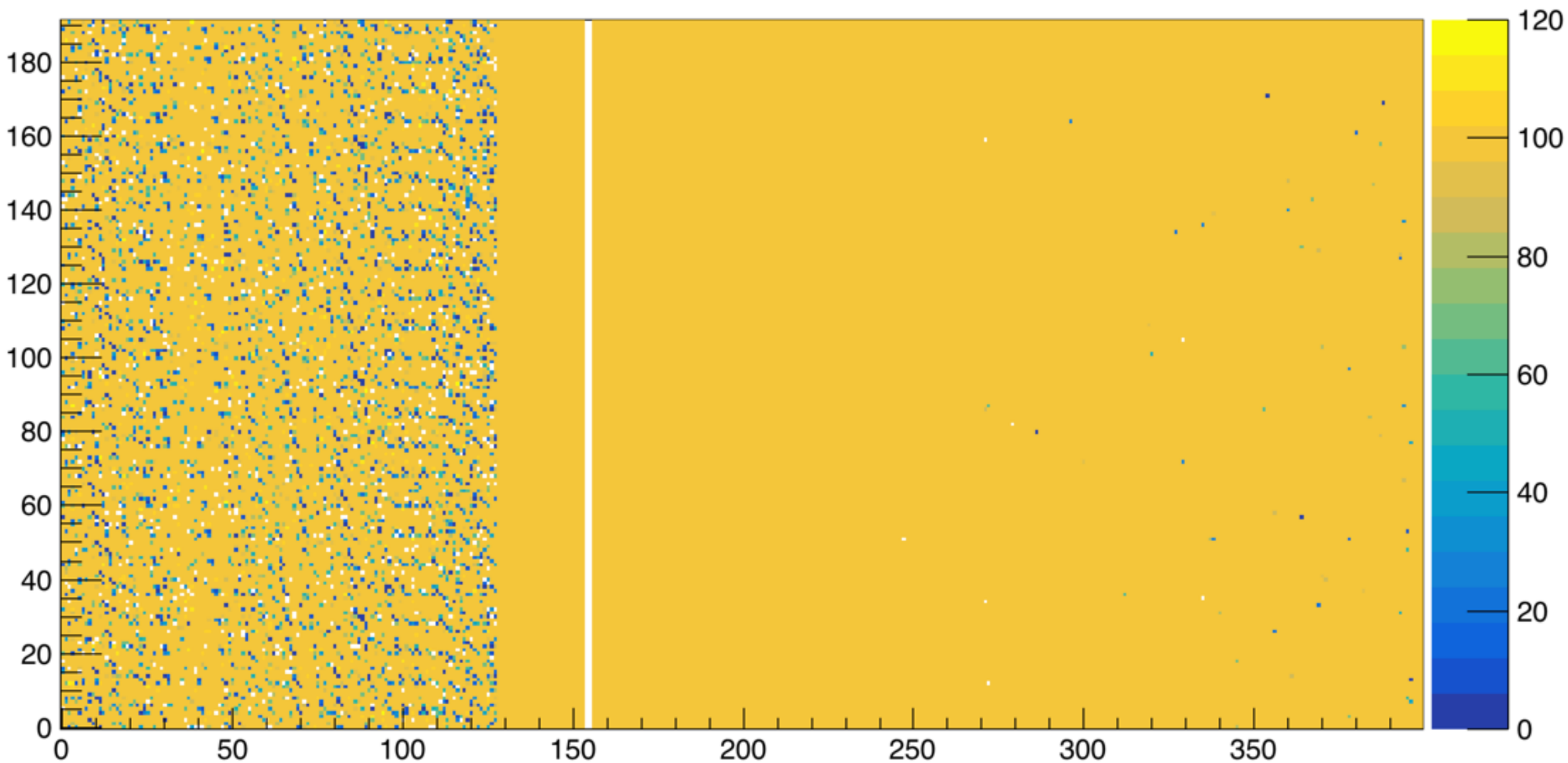
INJVCALDIFF = 1500 OCCUPANCY MAP

Occupancy map

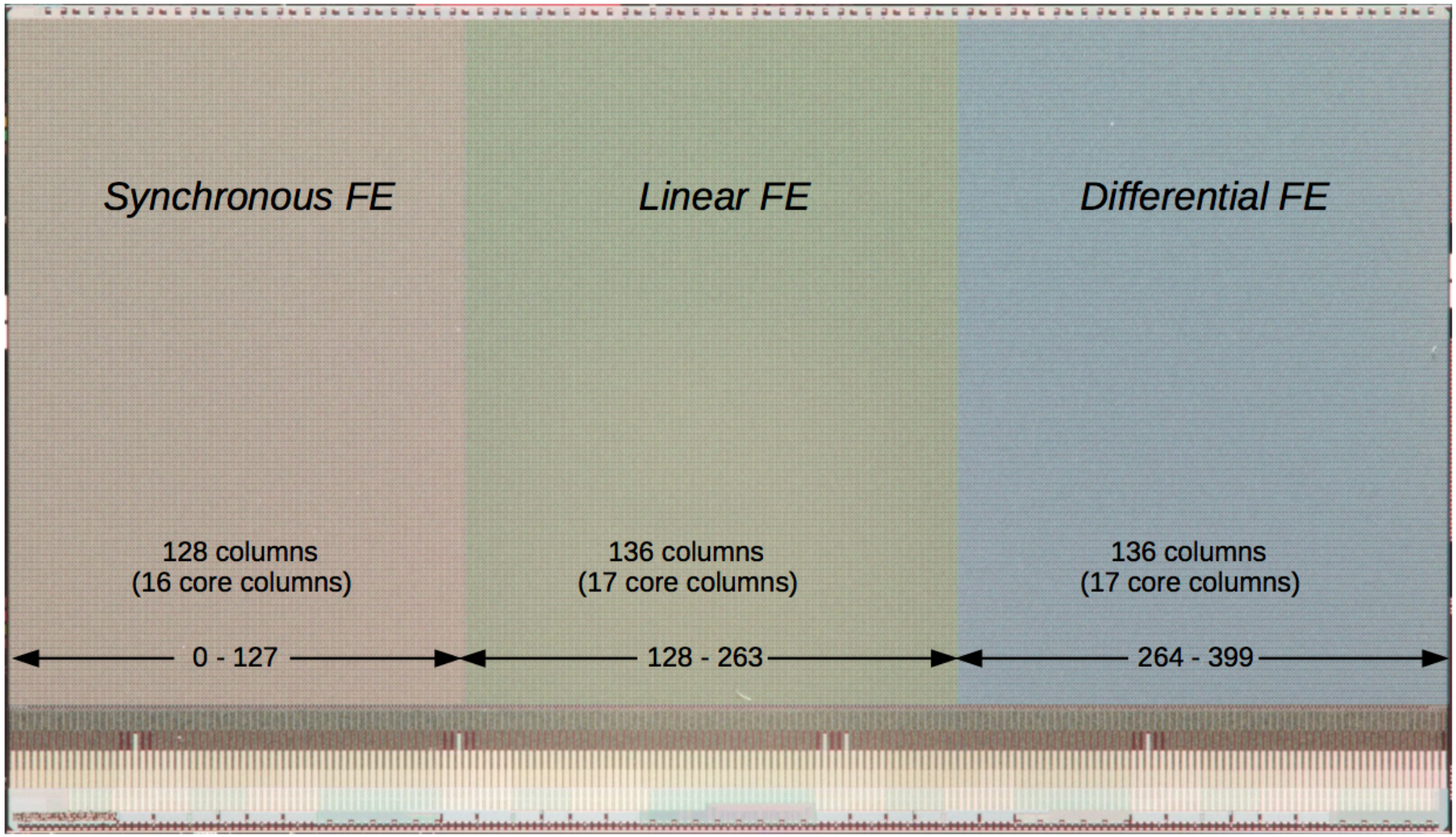


InjVCalDiff = 3000 Occupancy Map

Occupancy map



RD53A RECAP


Synchronous FE
Linear FE
Differential FE
128 columns
(16 core columns)
136 columns
(17 core columns)
136 columns
(17 core columns)
0 - 127
128 - 263
264 - 399

RD53A RECAP

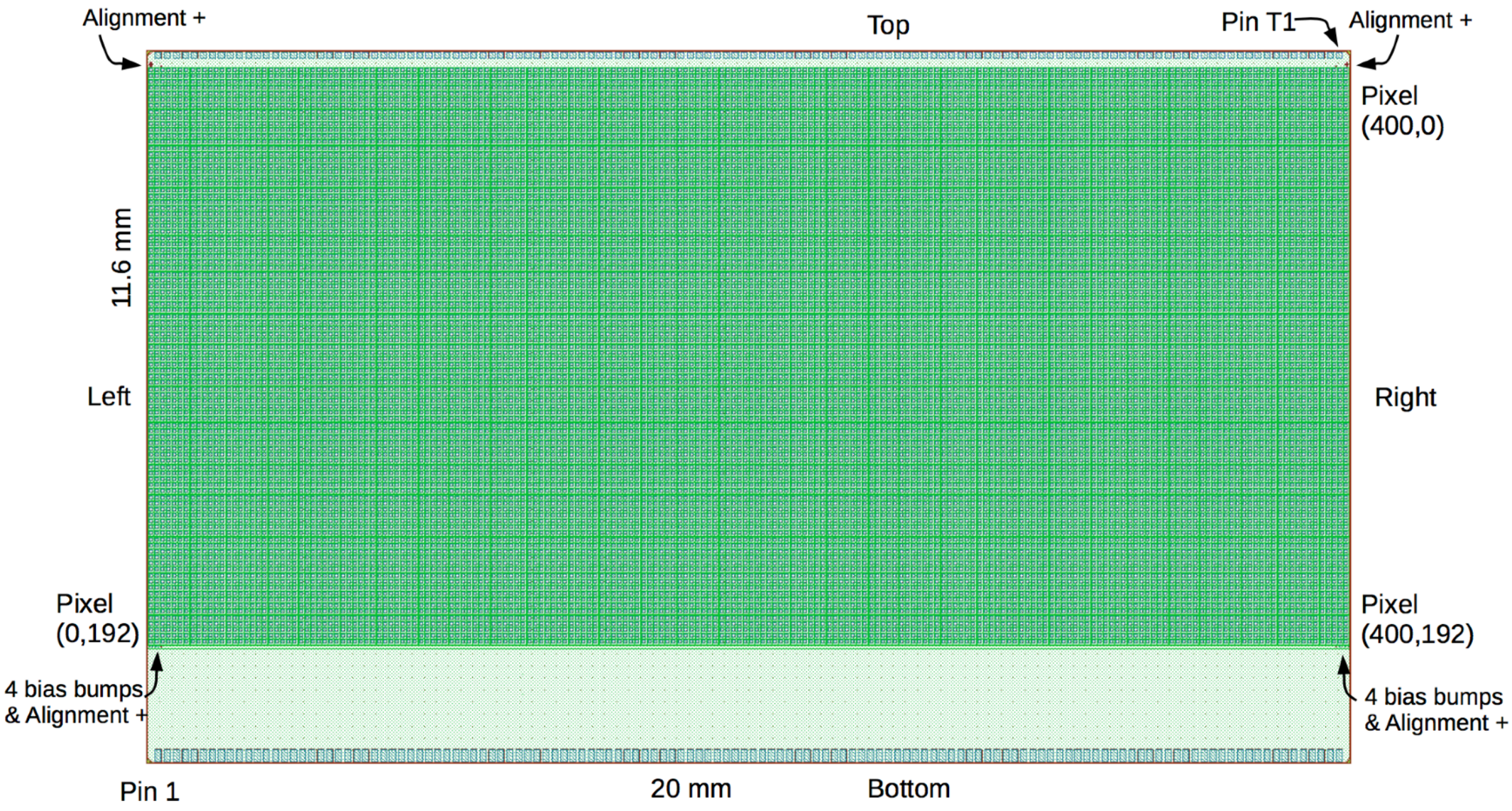


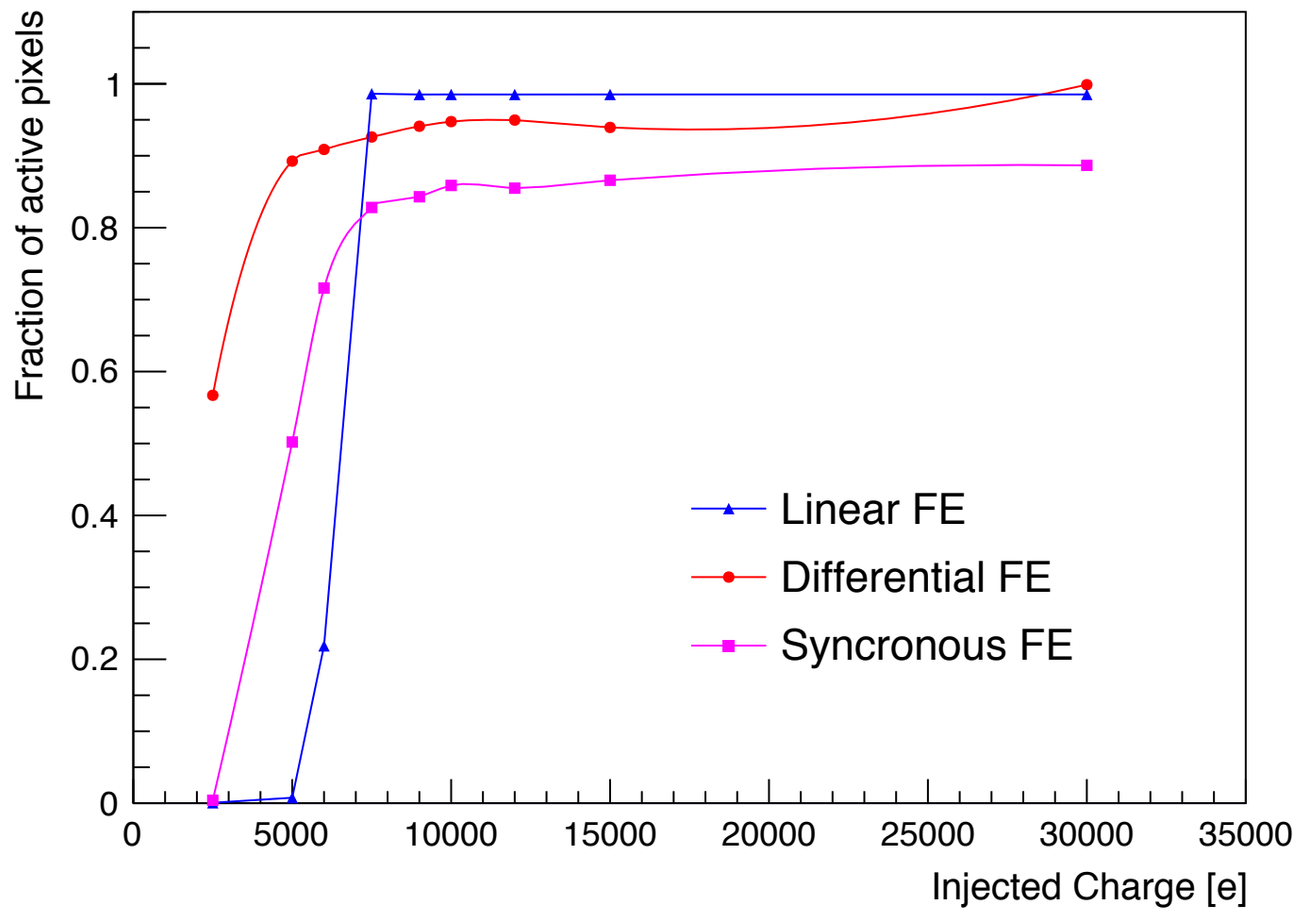
Figure 1: Top view of RD53A layout. The width is 20mm and 400 pixels, and the height is 11.6mm and 192 pixels.

AVERAGE EFFICIENCY

- Nhit/Npixel
- Nhit(ToT>0)/Npixel)

- Synchronous: col<128. ——> 24576
- Linear: 128 < col < 264——> 26112
- Differential: 264 < col < 400 ——> 26112

N_hit/C = > errore -> sqrt(N_hit)/C

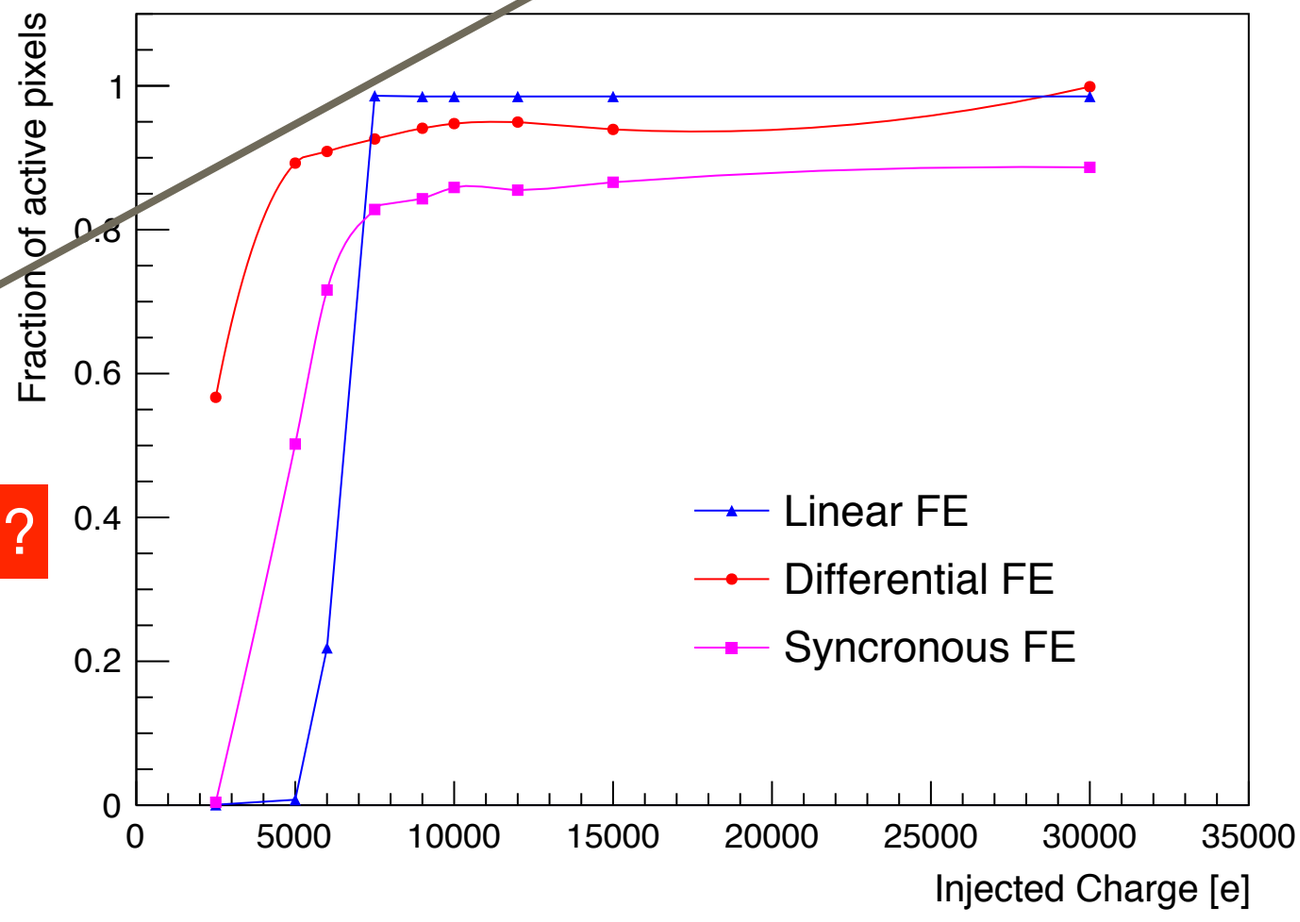


AVERAGE EFFICIENCY

- Nhit/Npixel
- Nhit(ToT>0)/Npixel)

- Synchronous: col<128. ——> 24576
- Linear: 128 < col < 264——> 26112
- Differential: 264 < col < 400 ——> 26112

N_hit/C = > errore -> sqrt(N_hit)/C



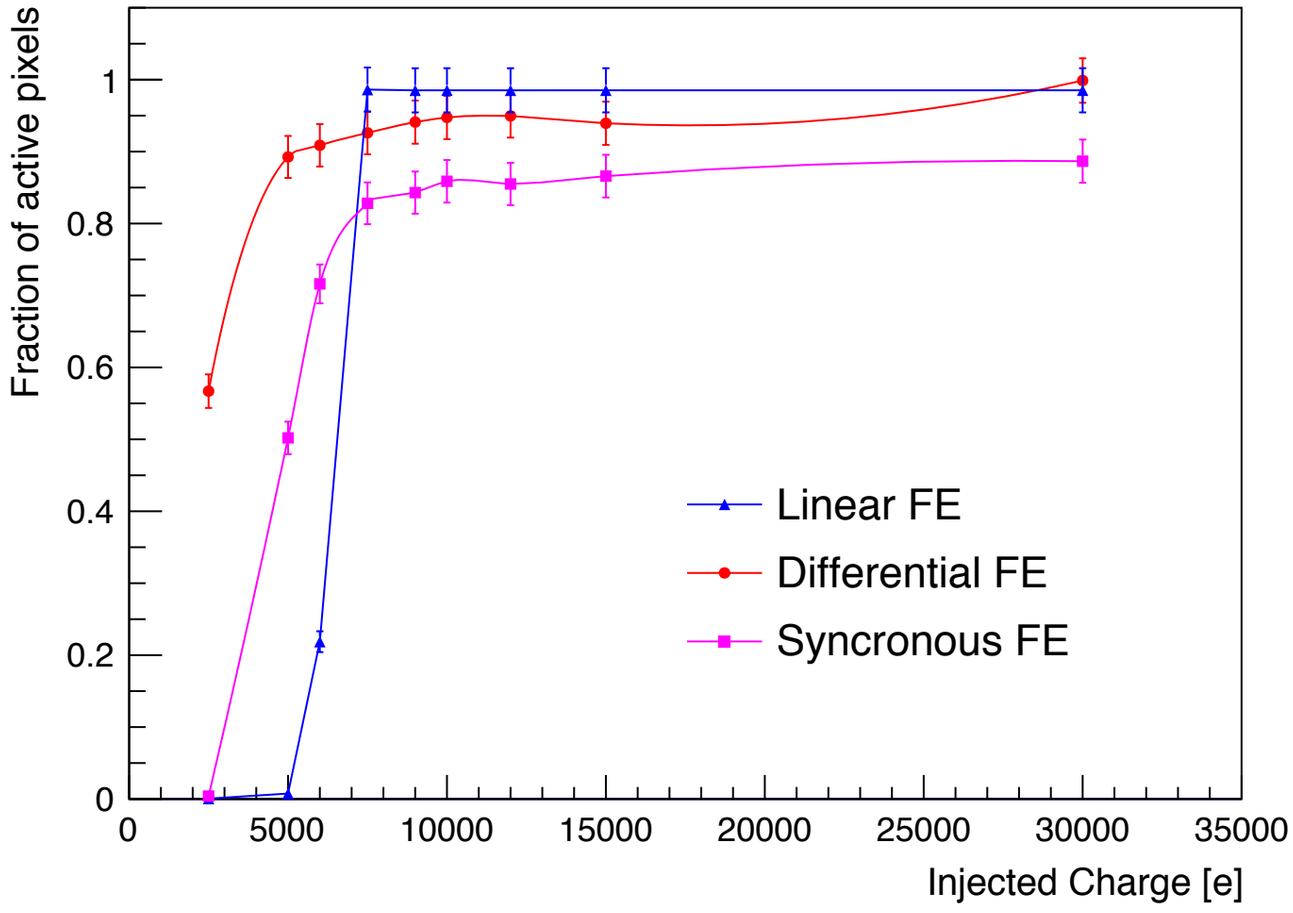
Altre proposte ??

AVERAGE EFFICIENCY

- Nhit/Npixel
- Nhit(ToT>0)/Npixel)

- Synchronous: col<128. ——> 24576
- Linear: 128 < col < 264——> 26112
- Differential: 264 < col < 400 ——> 26112

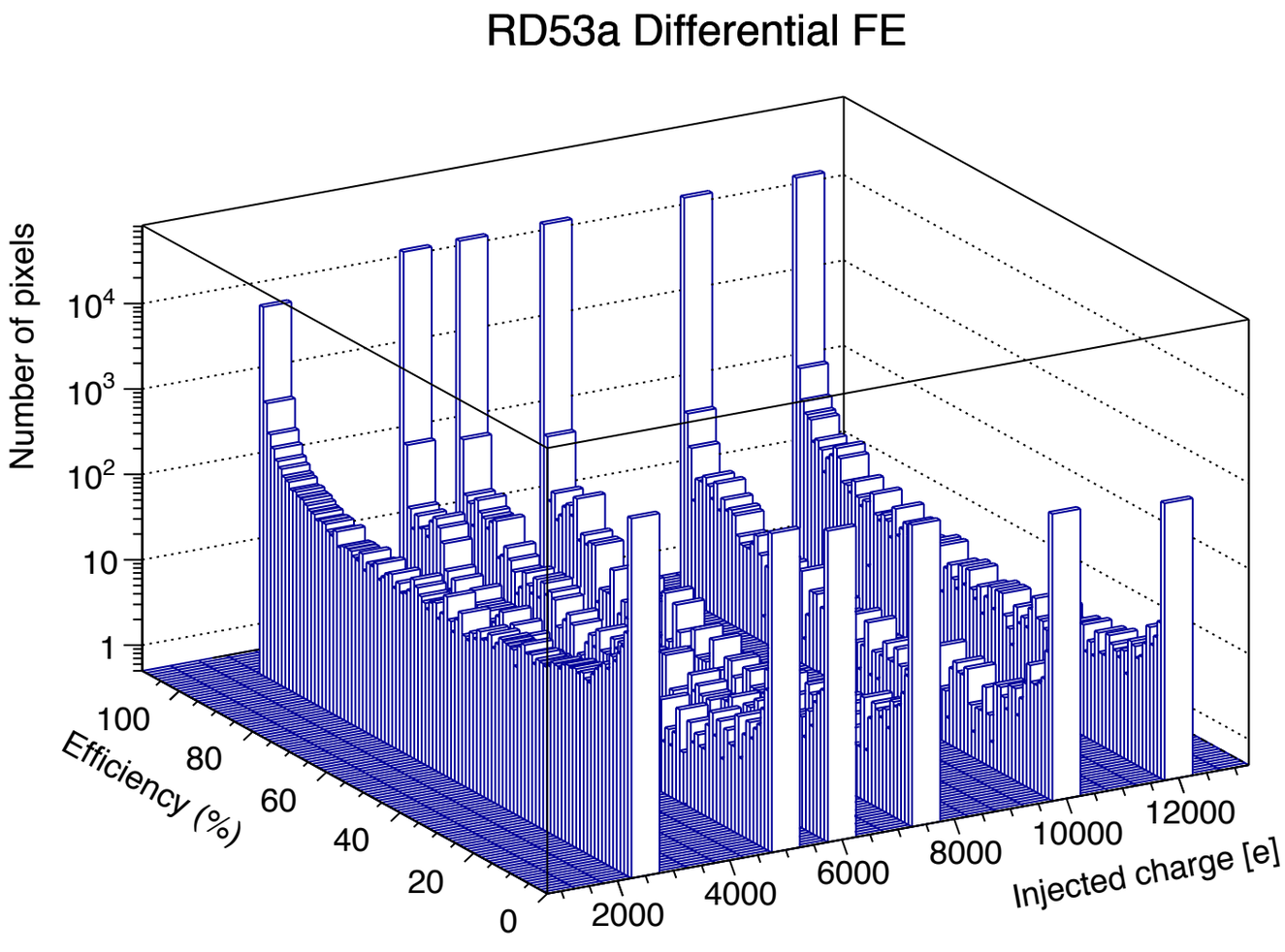
N_hit/C = > errore -> 50 x sqrt(N_hit)/C



Per visualizzare l'errore

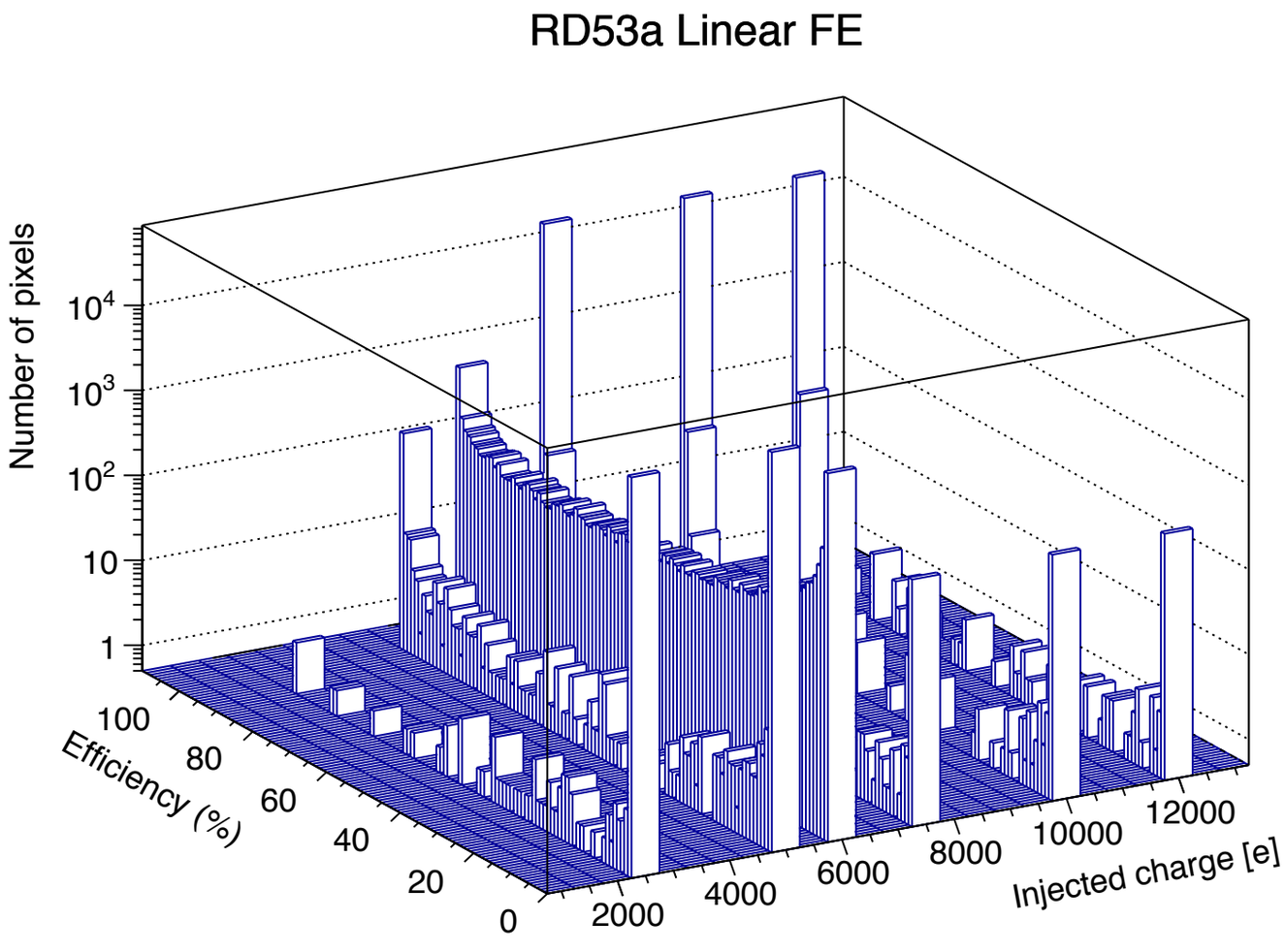
EFFICIENCY DISTRIBUTIONS

- For each “Occupancy map” (i.e. for each injected charge)
- loop over pixels (col and row), Fill this histogram with the observed pixel occupancy (max 100, as for the trigger loop); => Occ = 100 -> efficiency 100%



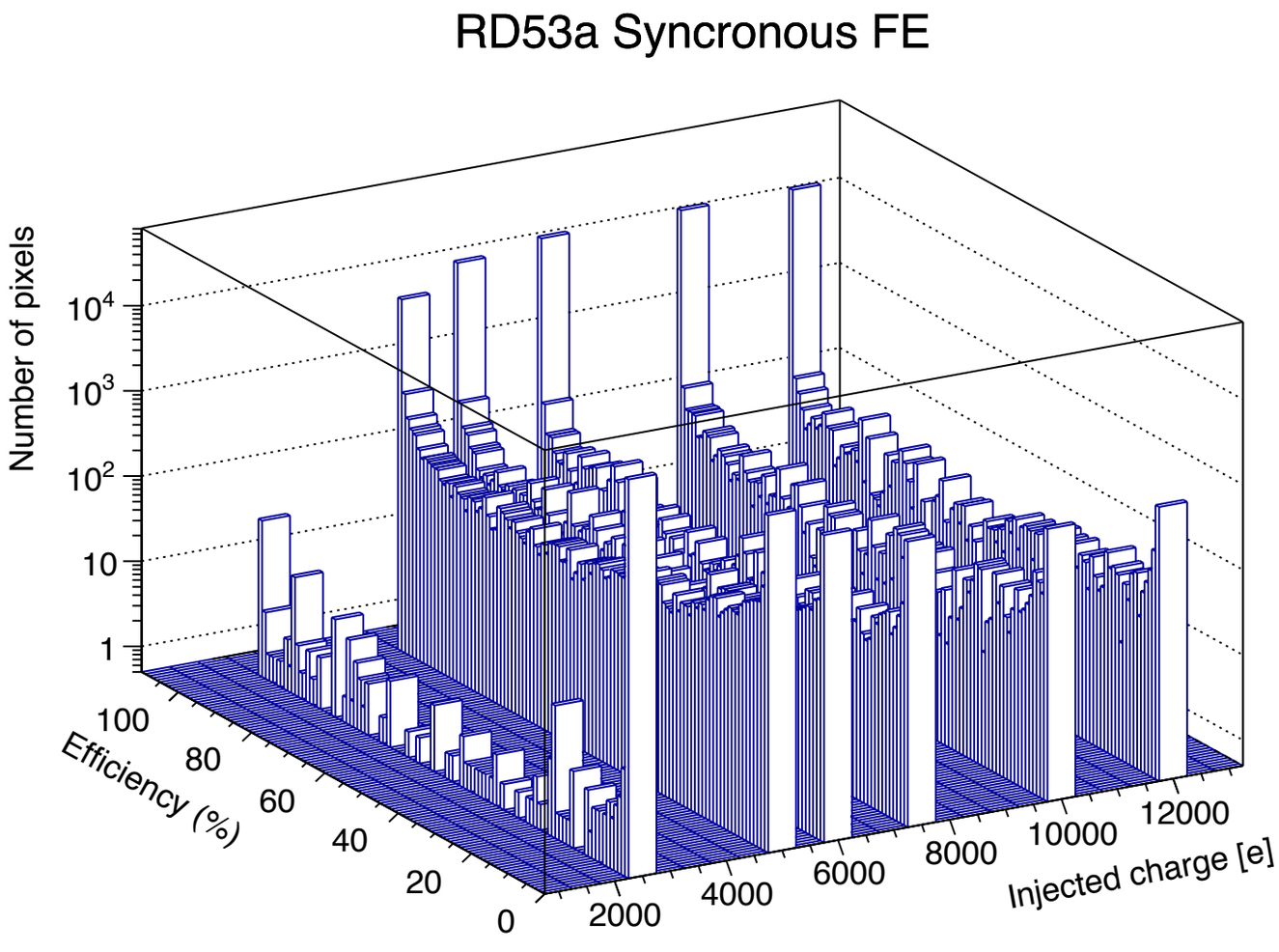
EFFICIENCY DISTRIBUTIONS

- For each “Occupancy map” (i.e. for each injected charge)
- loop over pixels (col and row), Fill this histogram with the observed pixel occupancy (max 100, as for the trigger loop); => Occ = 100 -> efficiency 100%



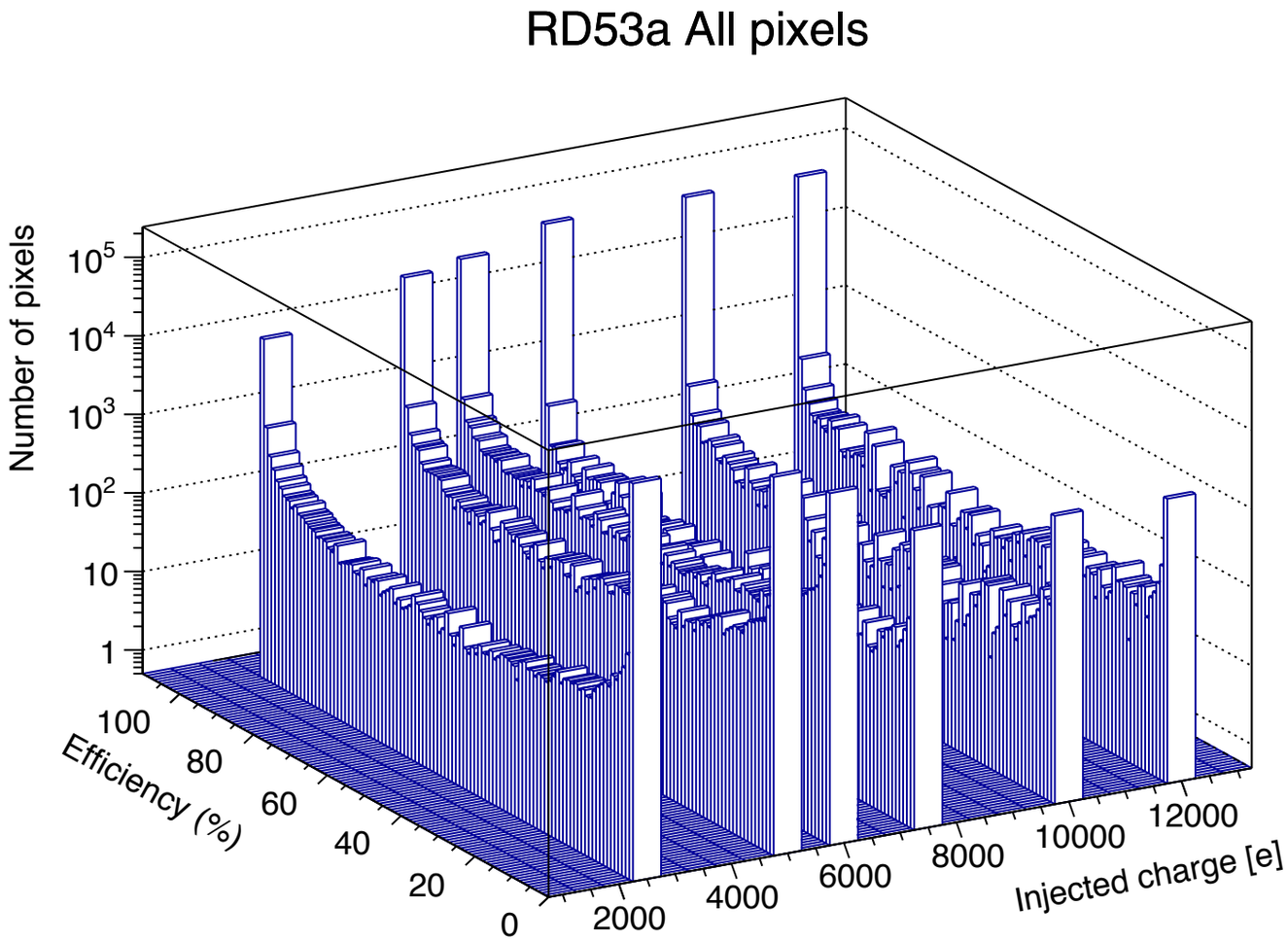
EFFICIENCY DISTRIBUTIONS

- For each “Occupancy map” (i.e. for each injected charge)
- loop over pixels (col and row), Fill this histogram with the observed pixel occupancy (max 100, as for the trigger loop); => Occ = 100 -> efficiency 100%



EFFICIENCY DISTRIBUTIONS

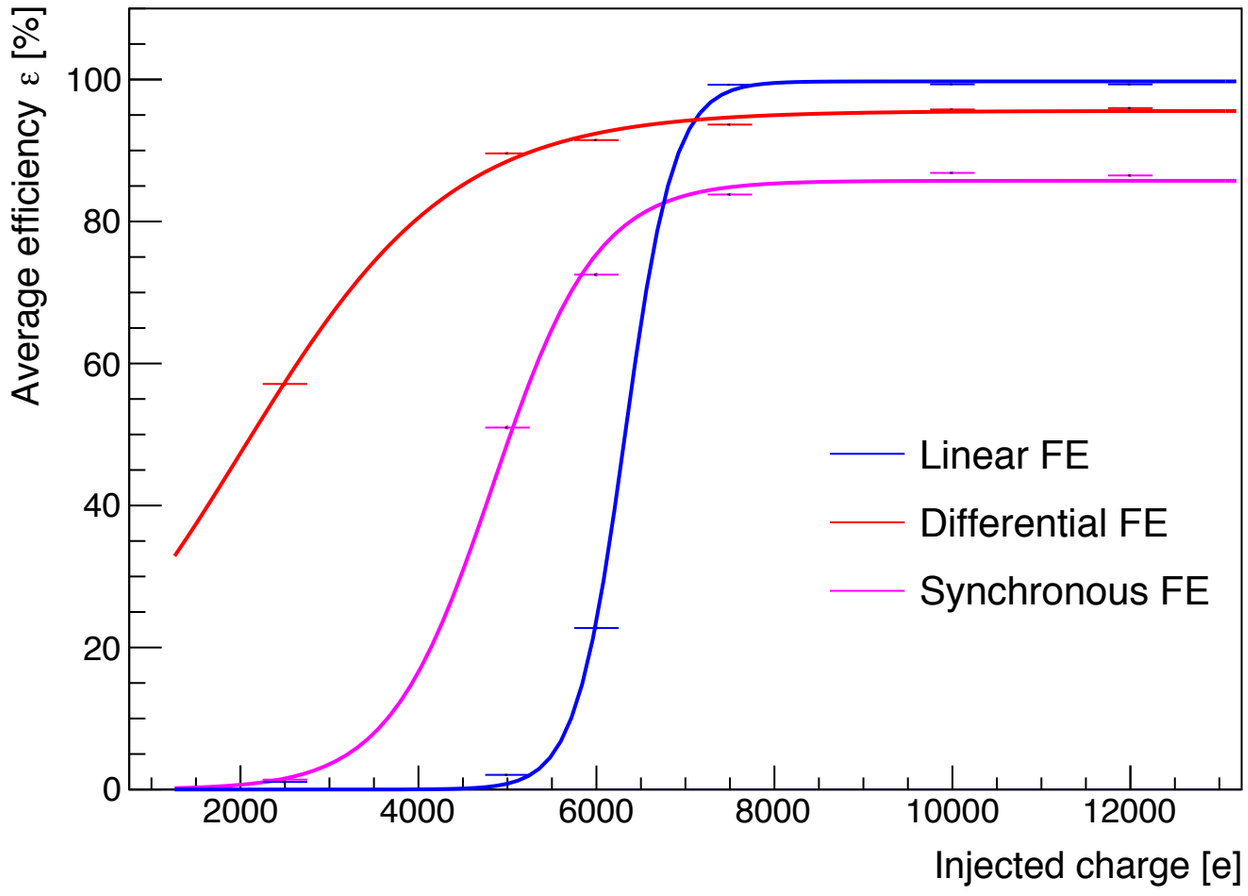
- For each “Occupancy map” (i.e. for each injected charge)
- loop over pixels (col and row), Fill this histogram with the observed pixel occupancy (max 100, as for the trigger loop); => Occ = 100 -> efficiency 100%



AVERAGE EFFICIENCY FROM DISTRIBUTIONS

- From the average and rms of the projections

```
Double_t plat=par[0]/( 1.+ pow(81.,  (par[1]-x[0])/par[2] ) );
```



FCN=10533.7 FROM MIGRAD STATUS=CONVER
EDM=1.57466e-07 STRATEGY= 1

EXT PARAMETER NO.	NAME	VALUE	ERROR	STEP SIZE
1	p0	9.97340e+01	5.16827e-02	2.29561
2	p1	6.32023e+03	3.96570e+00	1.46244
3	p2	1.21007e+03	1.26303e+01	4.55443

FCN=117.996 FROM MIGRAD STATUS=CONVER
EDM=2.24153e-08 STRATEGY= 1

EXT PARAMETER NO.	NAME	VALUE	ERROR	STEP SIZE
1	p0	9.55714e+01	8.93423e-02	3.64764
2	p1	2.02110e+03	1.81752e+01	7.01909
3	p2	5.18248e+03	7.90153e+01	2.52737

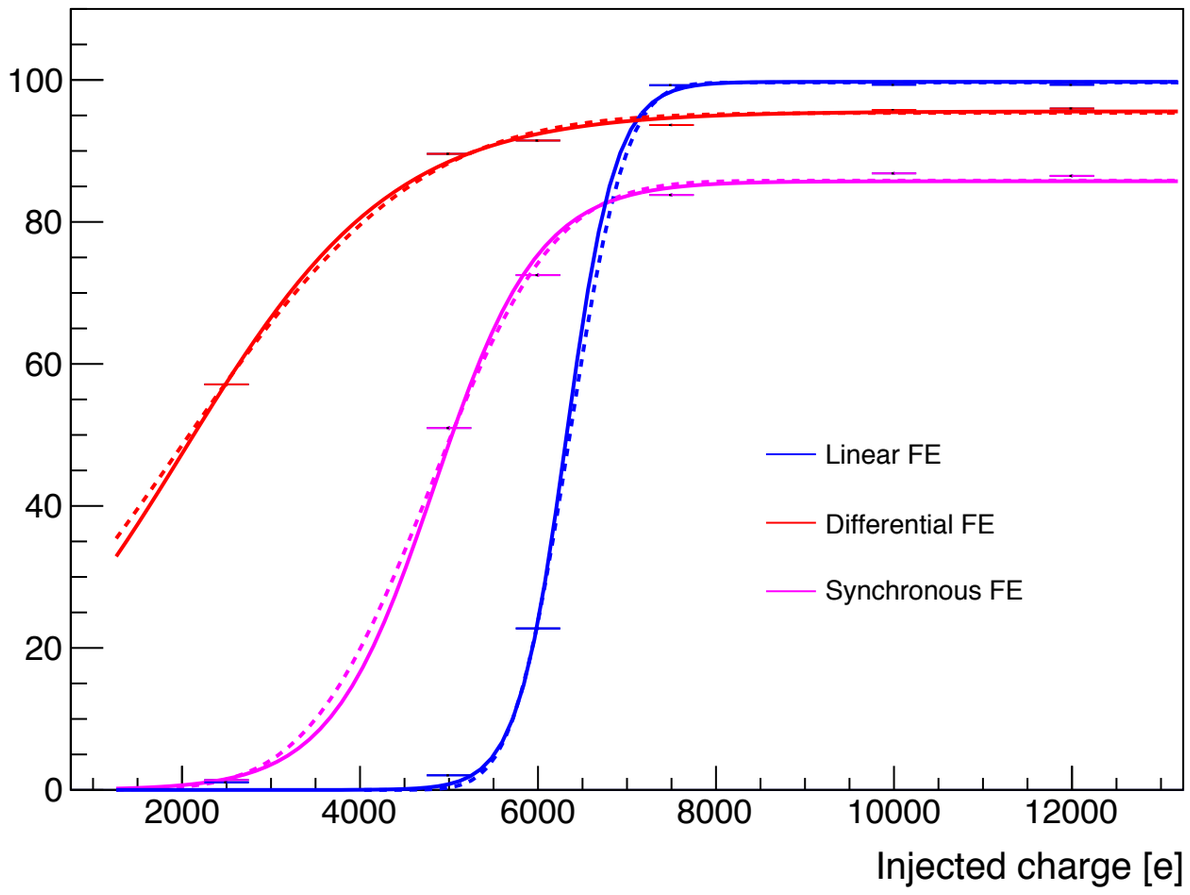
FCN=284.325 FROM MIGRAD STATUS=CONVER
EDM=9.50667e-13 STRATEGY= 1

EXT PARAMETER NO.	NAME	VALUE	ERROR	STEP SIZE
1	p0	8.57336e+01	1.17860e-01	8.69179
2	p1	4.83692e+03	7.01110e+00	5.09568
3	p2	2.57085e+03	1.46014e+01	1.12315

AVERAGE EFFICIENCY FROM DISTRIBUTIONS

- From the average and rms of the projections
- ***Tratteggiati i fit con la FC di una gaussiana***

```
Double_t myerffunction(Double_t *x, Double_t *par)
{
  Double_t z = (x[0]-par[1])/(sqrt(2.)*par[2]);
  Double_t plat=par[0]*(1.+erf(z));
  return plat;
}
```



FCN=950.825 FROM MIGRAD STATUS=CONVERGED
EDM=2.83729e-09 STRATEGY= 1 ERRO

EXT PARAMETER			STEP	FIRST
NO.	NAME	VALUE	ERROR	SIZE	DER
1	p0	4.98213e+01	2.67073e-02	3.44709e-04	
2	p1	6.35500e+03	4.01111e+00	4.68978e-02	
3	p2	4.98673e+02	5.34922e+00	6.01888e-02	

Chi2 = 950.825 nDoF = 2

FCN=198.997 FROM MIGRAD STATUS=CONVERGED
EDM=1.79204e-11 STRATEGY= 1 ERRO

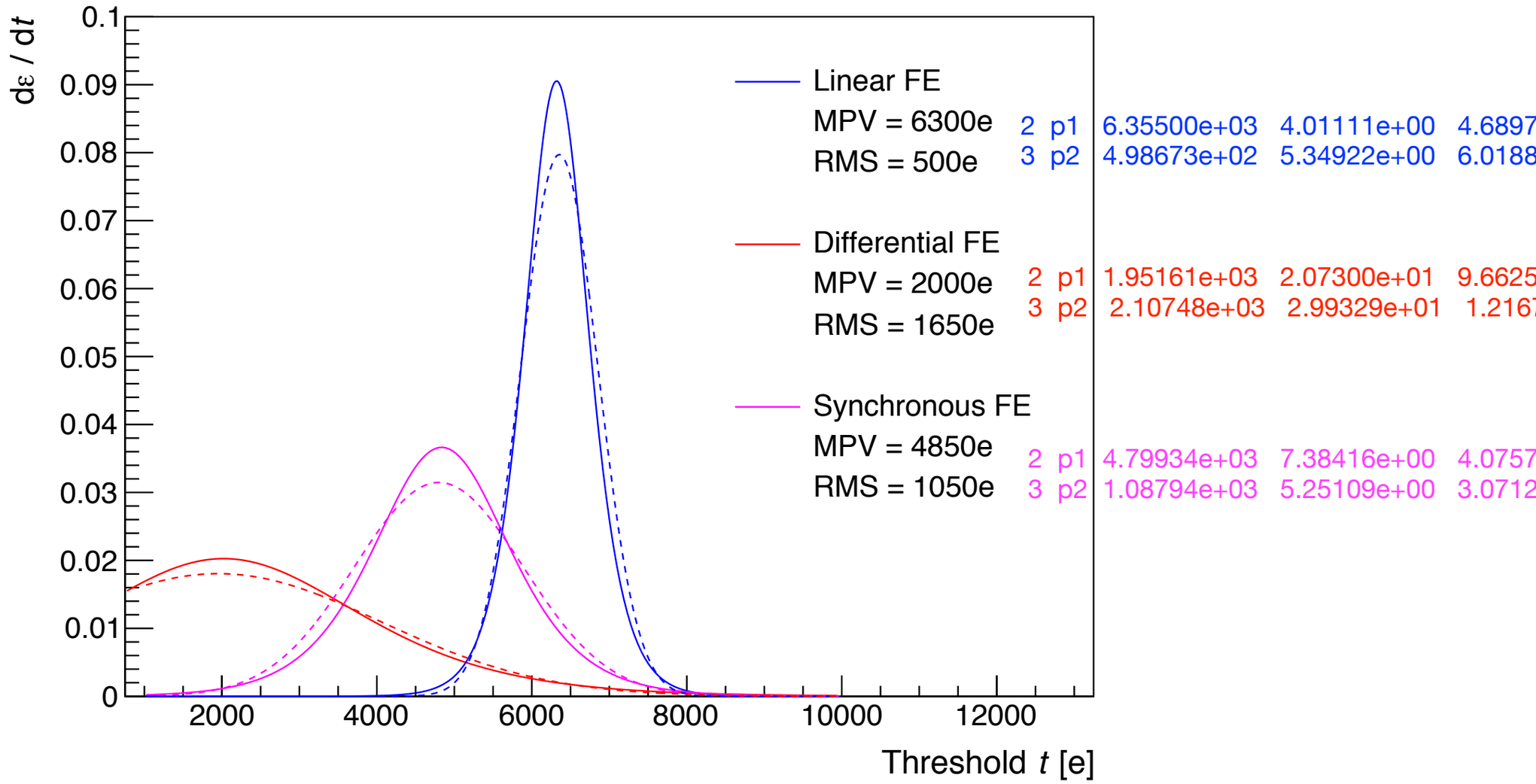
EXT PARAMETER			STEP	FIRST
NO.	NAME	VALUE	ERROR	SIZE	DER
1	p0	4.76779e+01	4.25649e-02	2.35915e-04	
2	p1	1.95161e+03	2.07300e+01	9.66250e-02	
3	p2	2.10748e+03	2.99329e+01	1.21675e-01	

FCN=182 FROM MIGRAD STATUS=CONVERGED 10
EDM=4.96975e-09 STRATEGY= 1 ERRO

EXT PARAMETER			STEP	FIRST
NO.	NAME	VALUE	ERROR	SIZE	DER
1	p0	4.29114e+01	5.93462e-02	3.48381e-04	
2	p1	4.79934e+03	7.38416e+00	4.07576e-02	
3	p2	1.08794e+03	5.25109e+00	3.07128e-02	

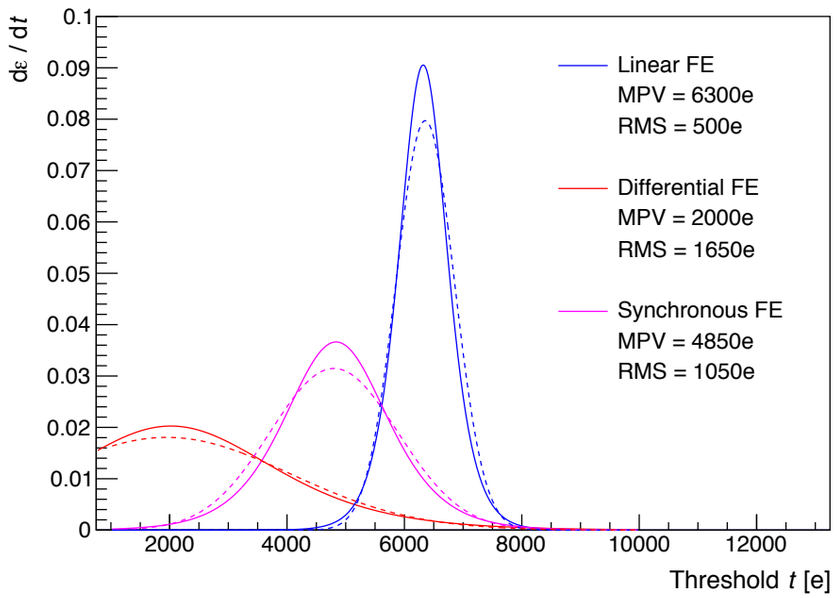
DISPERSIONE DI SOGLIA

- From the derivative of the fitting functions
  - Linea continua: Fermi
  - Linea tratteggiata: gaussian FC



DISPERSIONE DI SOGLIA

Regina: no tuning



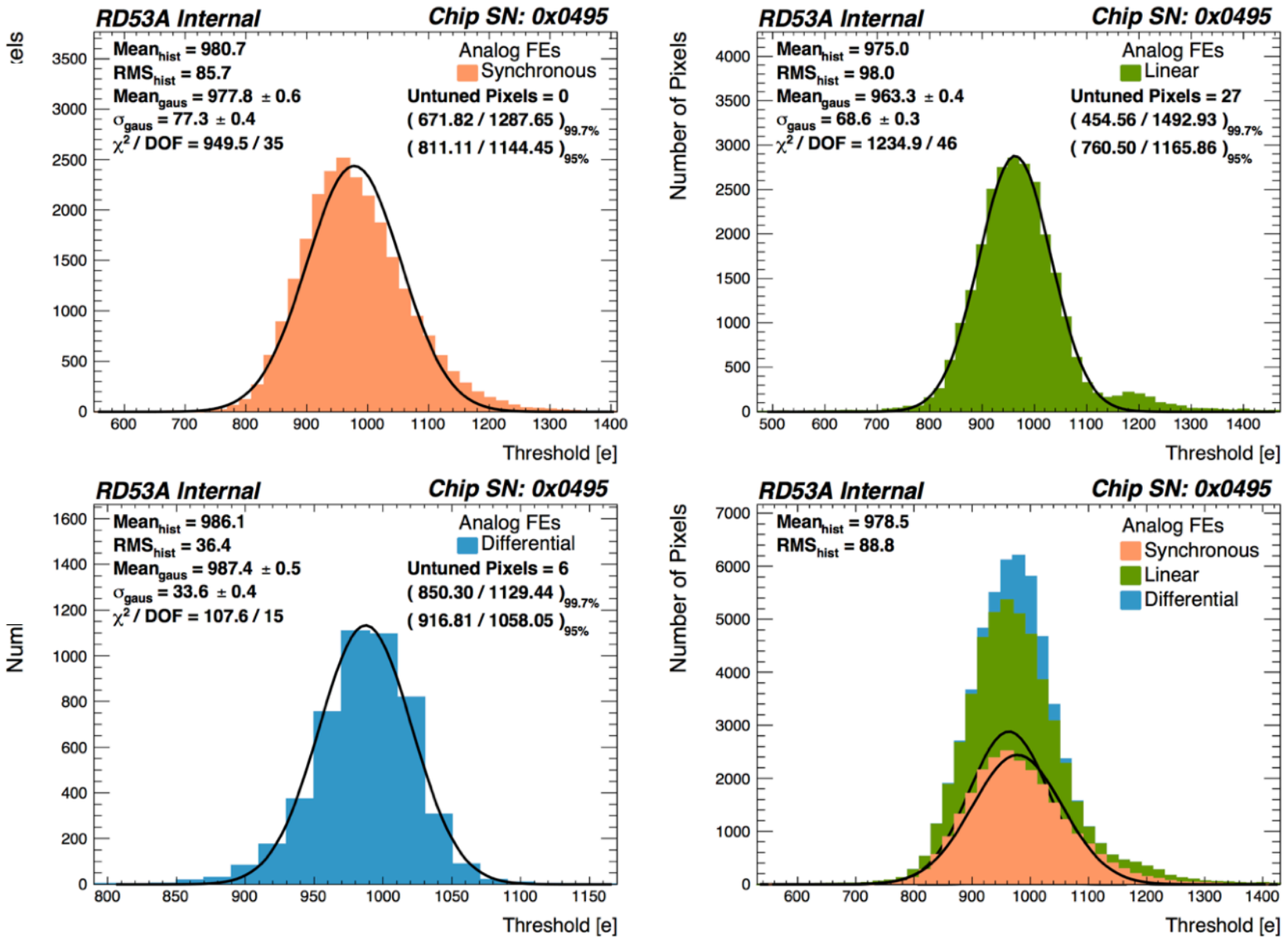
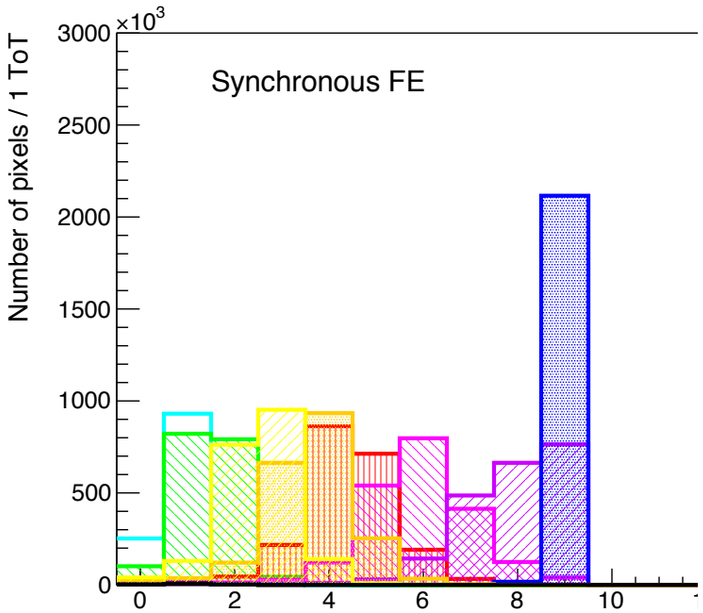
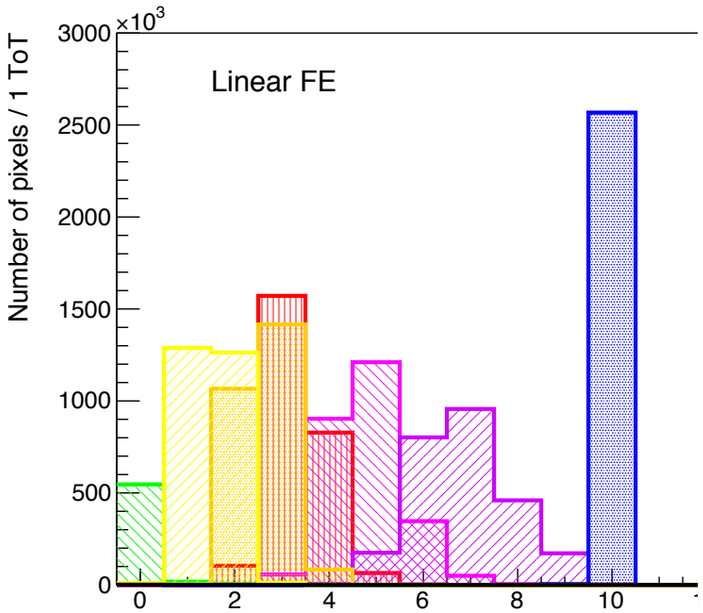


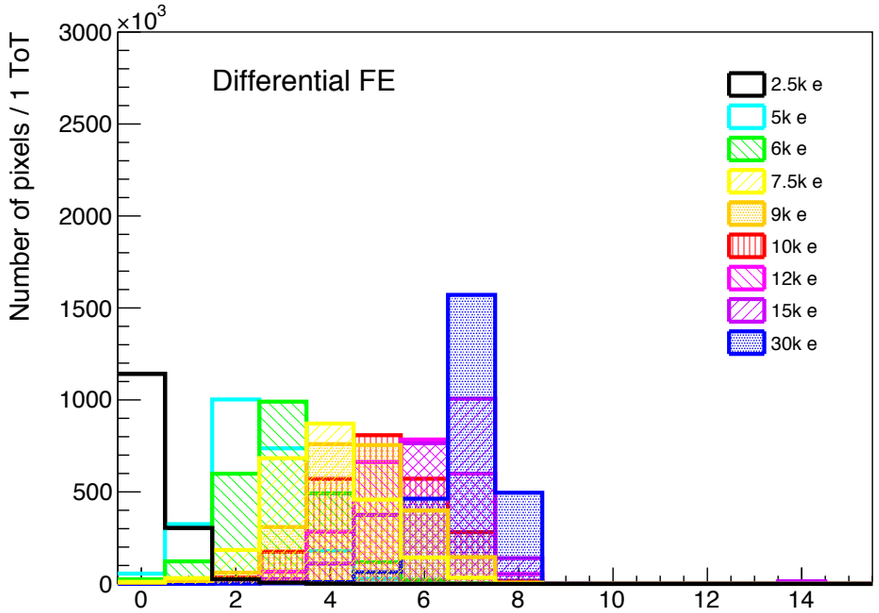
Figure 8: Threshold scan: threshold distributions.

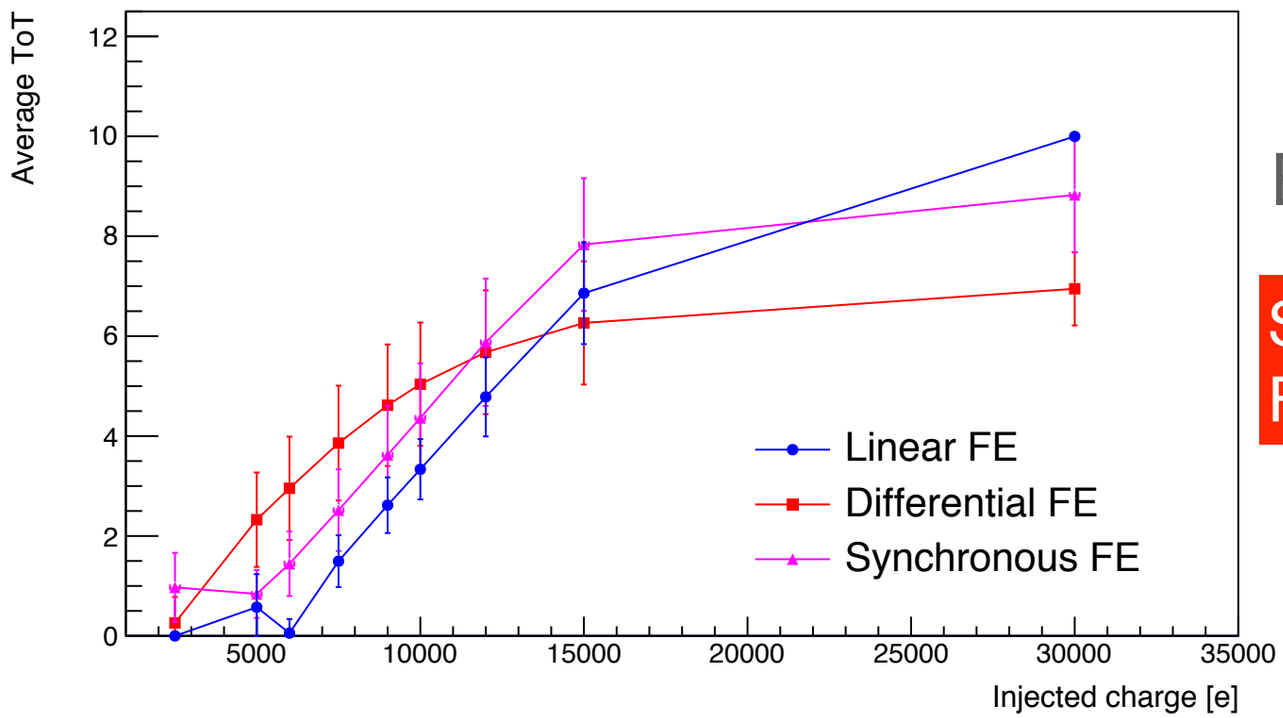
Traveling chip: post tuning

ToT DISTRIBUTIONS









Error -> RMS

Saturazione ?
Forte non linearita' ?

CONFIGURAZIONE - *STD_ANALOGSCAN.JSON*

https://github.com/Yarr/Yarr/blob/master/configs/scans/rd53a/std_analogscan.json

- Name: AnalogScan
  - Analisi:
    - OccupancyAnalysis, L1Analysis
  - Histogrammer:
    - OccupancyMap, TotMap, Tot2Map, L1Dist, HitsPerEvent, DataArchiver
  - Loops:
    - Rd53aMaskLoop (1 pixel per core, 64 [max permesso alla volta], step1)
      - Rd53aCoreColLoop (tutte e 50 le colonne di core)
        - Rd53aTriggerLoop (100 trigger per pixel) `"edgeMode": NOT SET -> false`
- Prescan:
  - "InjEnDig": 0,
  - "LatencyConfig": 48,
  - "GlobalPulseRt": 16384,
  - "InjVcalDiff": 0,
  - "InjVcalHigh": 2500,
  - "InjVcalMed": 500

CONFIGURAZIONE - *STD_THRESHOLDSCAN.JSON*

https://github.com/Yarr/Yarr/blob/master/configs/scans/rd53a/std_thresholdscan.json

- Name: ThresholdScan
  - Analisi:
    - ScurveFitter, OccupancyAnalysis
  - Histogrammer:
    - OccupancyMap, TotMap, Tot2Map, L1Dist, HitsPer
  - Loops:
    - Rd53aMaskLoop
      - Rd53aParameterLoop su InjVcalDiff
        - Rd53aCoreColLoop (tutte e 50 le colonne di core)
          - Rd53aTriggerLoop (100 trigger per pixel)
  - Prescan:
    - "InjEnDig": 0,
    - "LatencyConfig": 48,
    - "GlobalPulseRt": 16384

```
{
  "config": {
    "max": 300,
    "min": 50,
    "step": 5,
    "parameter":"InjVcalDiff"
  },
  "loopAction": "Rd53aParameterLoop"
},
```

"edgeMode": NOT SET -> false

CONFIGURAZIONE - *DIFF_TUNE_GLOBALTHRESHOLD.JSON*

- Name: DiffGlobalThresholdTune
  - Analisi:
    - OccGlobalThresholdTune
  - Histogrammer:
    - OccupancyMap, TotMap, Tot2Map, L1Dist, HitsPerEvent
  - Loops:
    - Rd53aGlobalFeedback
      - Rd53aMaskLoop (from 0 to 64, step 8)
        - Rd53aCoreColLoop (colonne 33-50 [diff], step 1, nSteps 17)
          - Rd53aTriggerLoop (100 trigger per pixel)
  - Prescan:
    - "InjEnDig": 0,
    - "LatencyConfig": 48,
    - "GlobalPulseRt": 16384

```
{ "config": {
    "max": 500,
    "min": 0,
    "step": 16,
    "parameter": "DiffVth1",
    "pixelRegs": [0, 1]

  },
  "loopAction": "Rd53aGlobalFeedback"
},
```

CONFIGURAZIONE - *DIFF_TUNE_PIXELTHRESHOLD.JSON*

- Name: StdPixelThresholdTune
  - Analisi:
    - OccGlobalThresholdTune
  - Histogrammer:
    - OccupancyMap, TotMap, Tot2Map, L1Dist, HitsPerEvent
  - Loops:
    - Rd53aPixelFeedback
      - Rd53aMaskLoop (from 0 to 64, step 8)
        - Rd53aCoreColLoop (colonne 33-50 [diff], step 1, nSteps 2)
          - Rd53aTriggerLoop (100 trigger per pixel)
  - Prescan:
    - "InjEnDig": 0,
    - "LatencyConfig": 48,
    - "GlobalPulseRt": 0,
    - "EnCoreColSync": 0,
    - "EnCoreColLin1": 0,
    - "EnCoreColLin2": 0

```
{
    "config": {
        "max": 15,
        "min": -15,
        "steps": [8,4,2,1,1],
        "tuneDiff": true,
        "tuneLin": false,
        "resetTdac": true
    },
    "loopAction": "Rd53aPixelFeedback"
```

CONFIGURAZIONE - *DIFF_TUNE_FINEPIXELTHRESHOLD.JSON*

- Name: FinePixelTuning
  - Analisi:
    - Tune ScurveFitter, OccupancyAnalysis
  - Histogrammer:
    - OccupancyMap, TotMap, Tot2Map, L1Dist, HitsPerEvent
  - Loops:
    - Rd53aPixelFeedback
      - Rd53aMaskLoop (from 0 to 64, step 8)
        - *Rd53aParameterLoop (InjVcalDiff, 50-150, step 5)*
          - Rd53aCoreColLoop (colonne 33-50 [diff], step 1, nSteps 17)
            - Rd53aTriggerLoop (100 trigger per pixel)
- Prescan:
  - "InjEnDig": 0,
  - "LatencyConfig": 48,
  - "GlobalPulseRt": 0,
  - "EnCoreColSync": 0,
  - "EnCoreColLin1": 0,
  - "EnCoreColLin2": 0

```
"config": {
    "max": 15,
    "min": -15,
    "steps": [1,1,1],
    "tuneDiff": true,
    "tuneLin": false,
    "resetTdac": false
},
"loopAction": "Rd53aPixelFeedback"
```

CONFIGURAZIONE - *DIFF_TUNE_GLOBALPREAMP.JSON*

- Name: DiffGlobalPreampTune
  - Analisi:
    - TotAnalysis, OccupancyAnalysis, L1Analysis
  - Histogrammer:
    - OccupancyMap, TotMap, Tot2Map, L1Dist, HitsPerEven
  - Loops:
    - Rd53aPixelFeedback
      - Rd53aMaskLoop (from 0 to 64, step 1)
        - Rd53aCoreColLoop (colonne 33-50 [diff], step 1, nSteps 17)
          - Rd53aTriggerLoop (100 trigger per pixel)
  - Prescan:

```
{ "config": {
    "max": 64,
    "min": 0,
    "step": 32,
    "parameter": "DiffVff"
  },
  "loopAction": "Rd53aGlobalFeedback"
```

```
"InjEnDig": 0,
"LatencyConfig": 48,
"GlobalPulseRt": 0,
"SyncVth":500,
"LinVth": 500,
"EnCoreColLin1": 0,
"EnCoreColLin2": 0,
"EnCoreColSync": 0
```