LPSC GRENOBLE

for

CALICE Collaboration

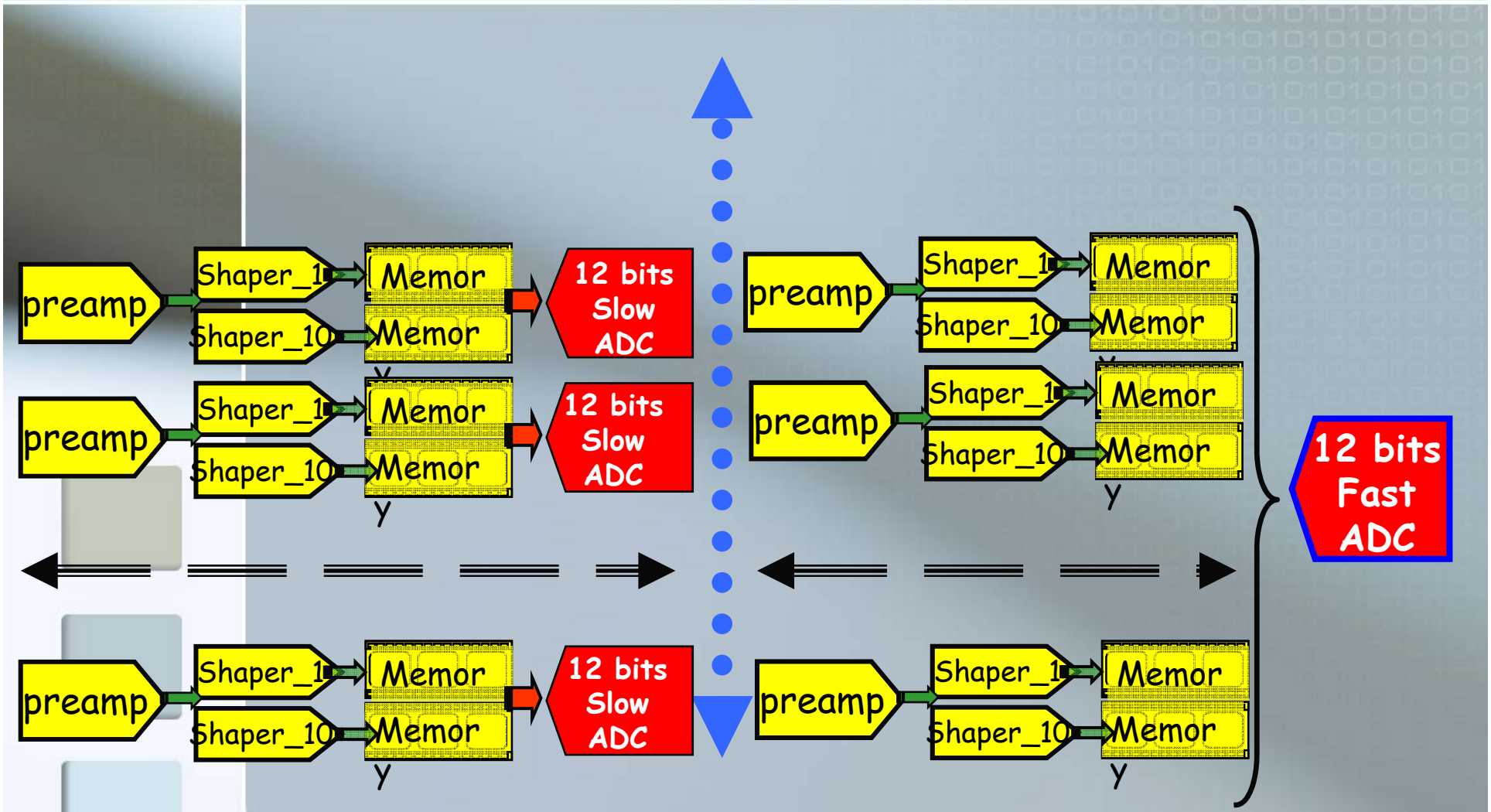
A low power 12-bits up to 30-MS/s pipelined ADC *(F.Rarbi; D. Dzahini)*

16 bits DAC for the CALIBRATION

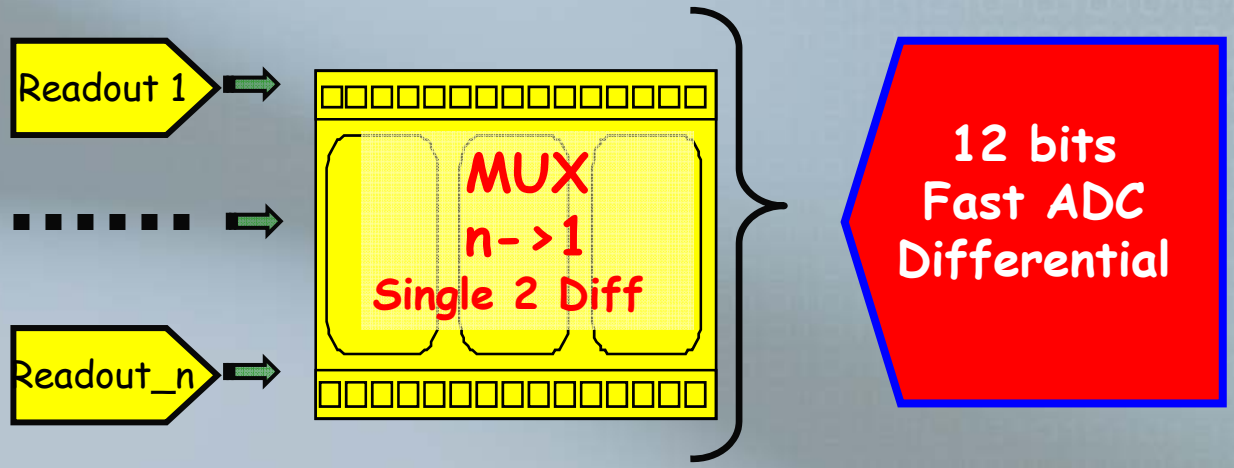
Laurent Gallin Martel / Olivier Rossetto /D. Dzahini

Slow versus fast digitizer for CALICE

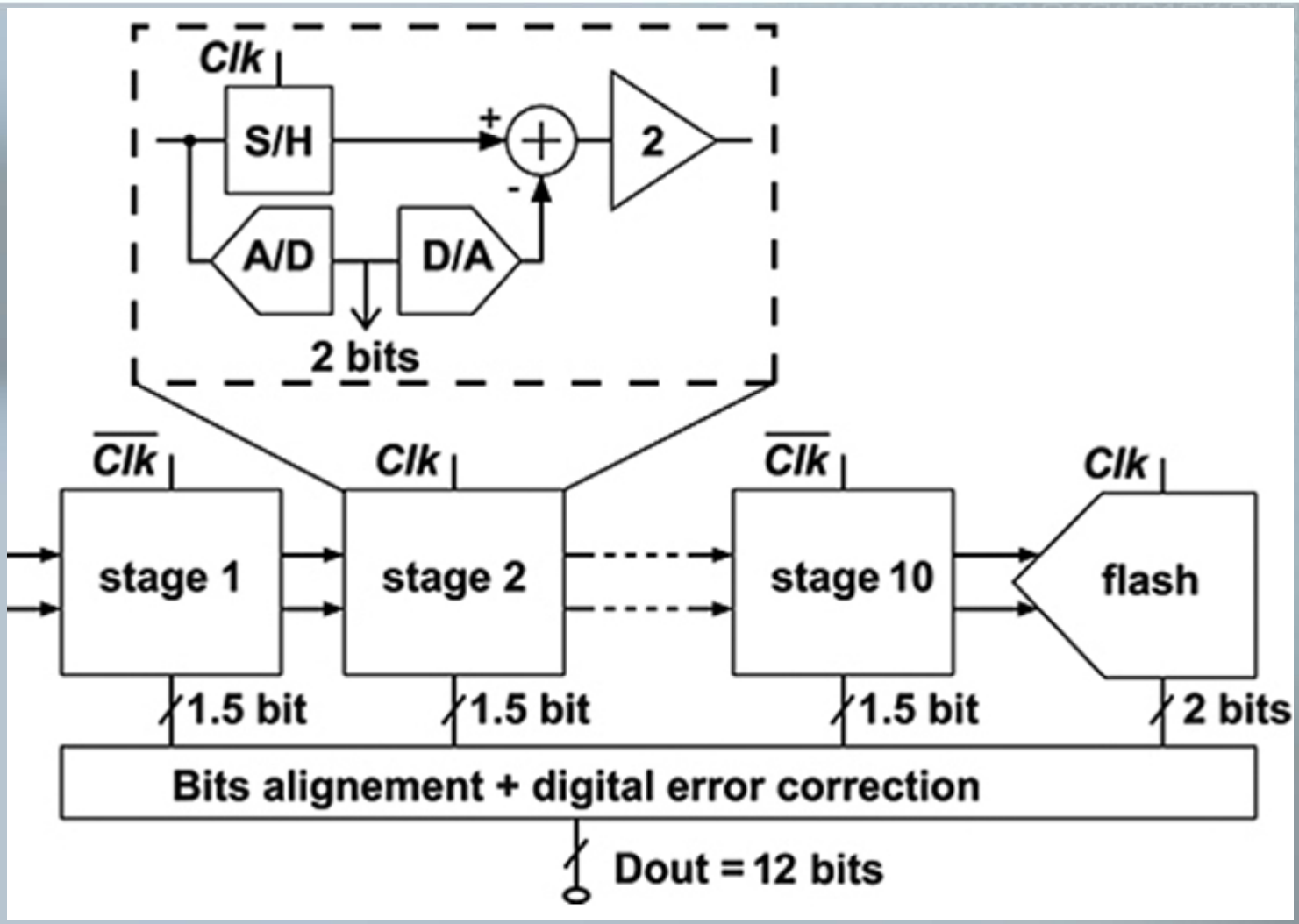
Which one will optimize better: Power / Clock noise / cross talk



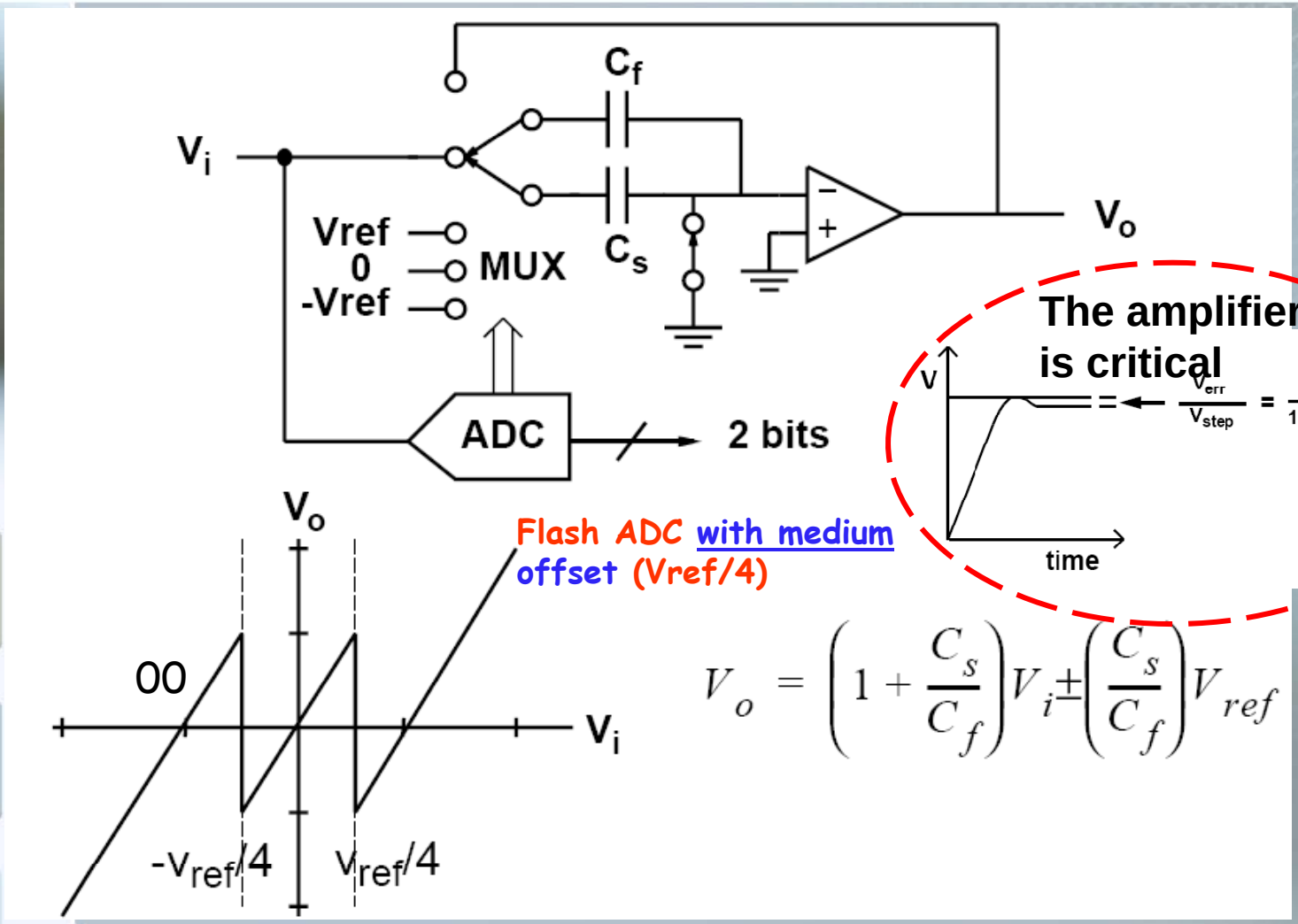
Fast MUX + High speed ADC proposal



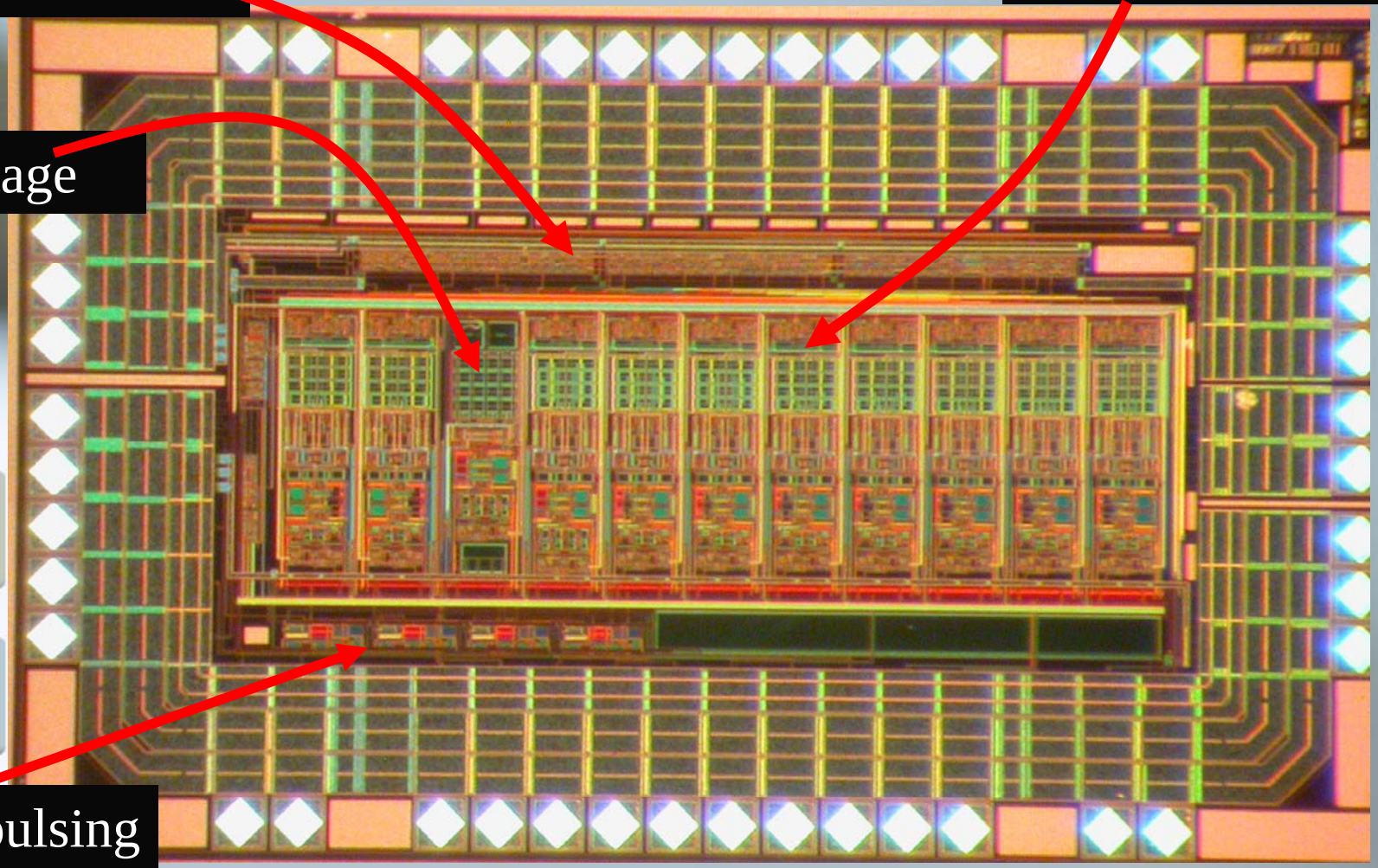
ADC Pipeline



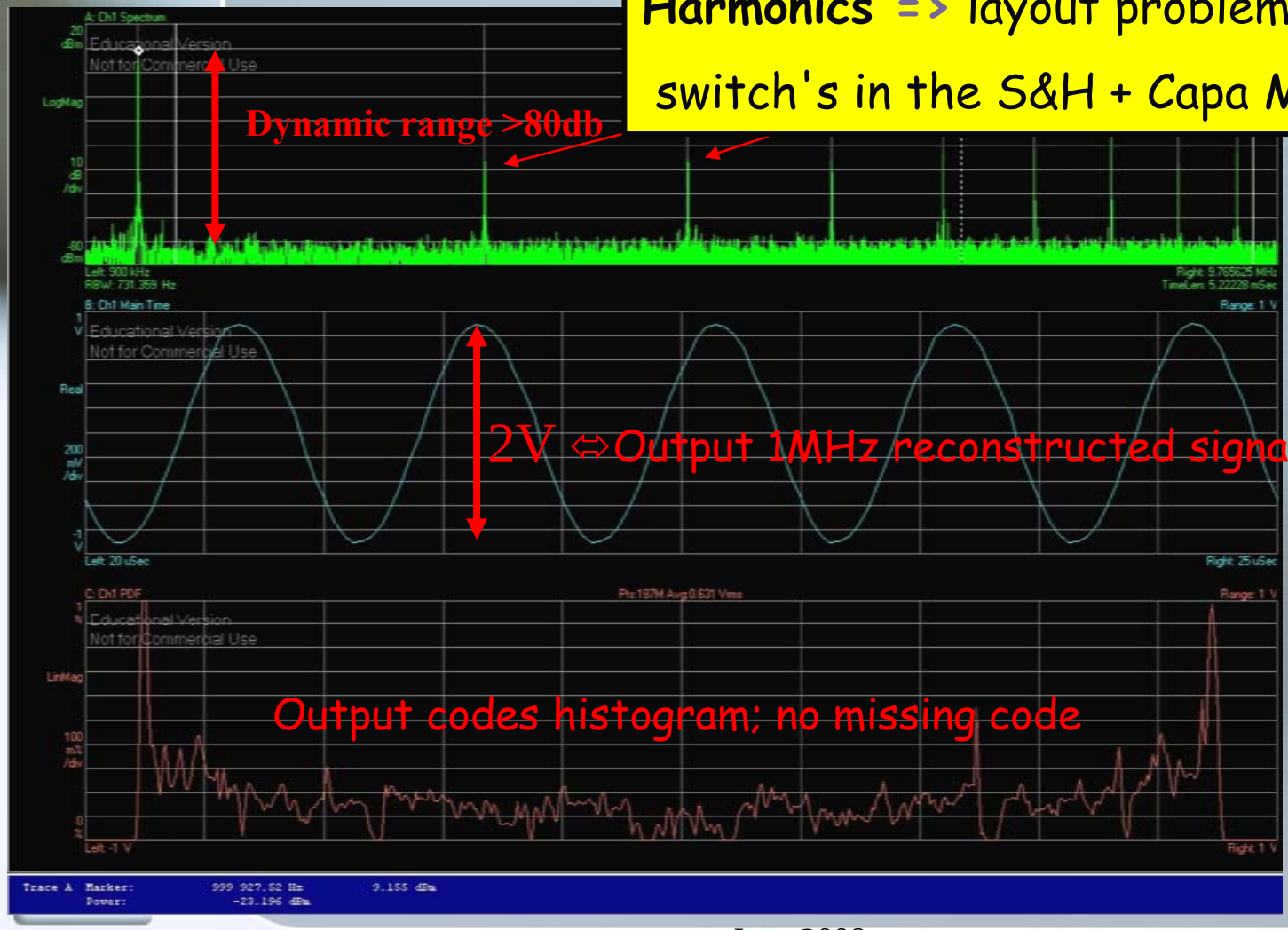
MDAC Function: Multiplier & DAC



Die of the «prototype n-2» Oct.07: S&H+ADC

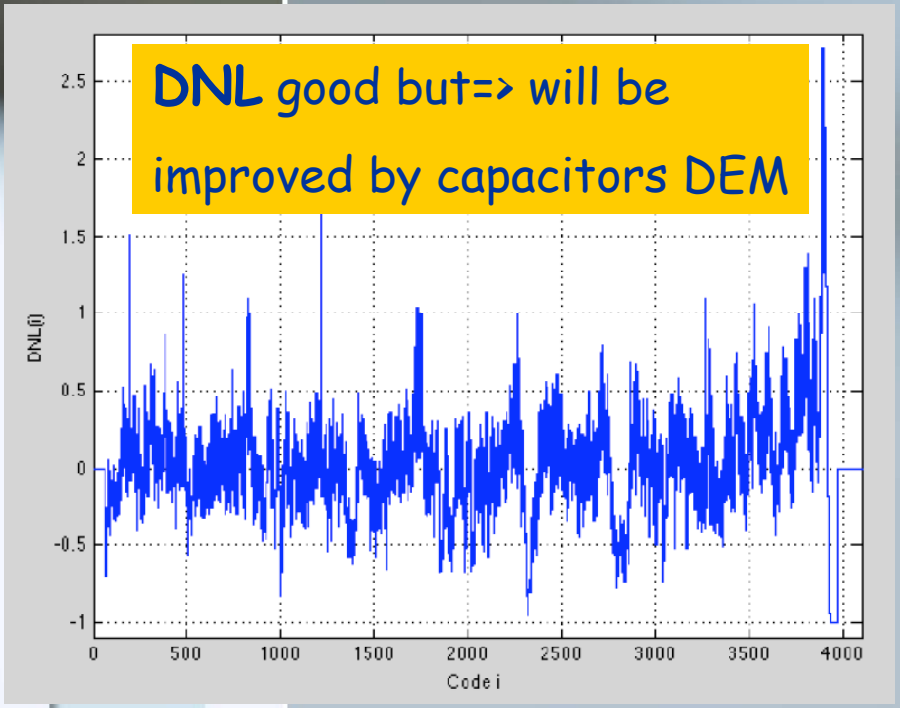


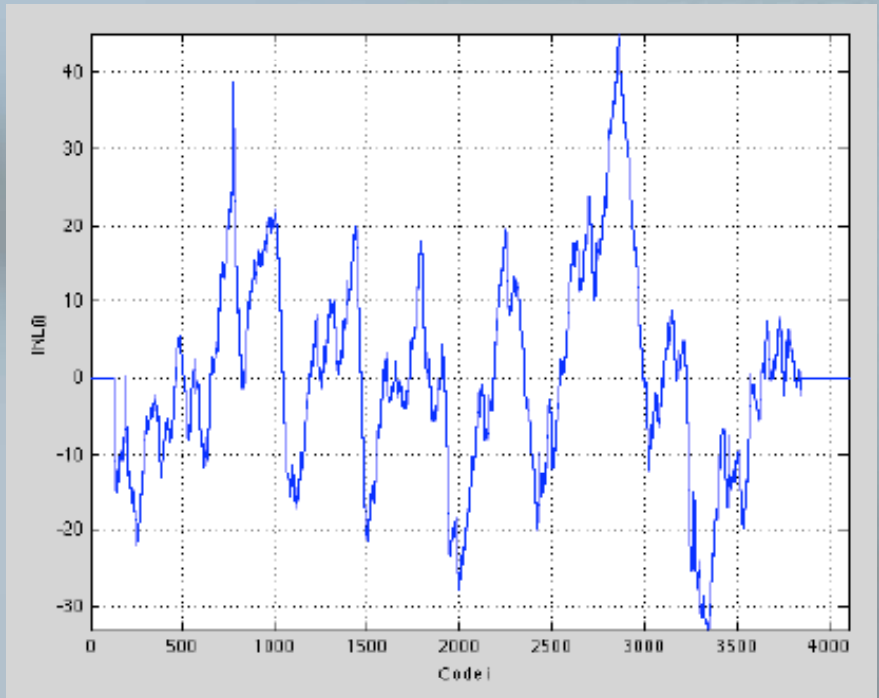
Spectral response to an input sinus:1Mhz, 2V pp



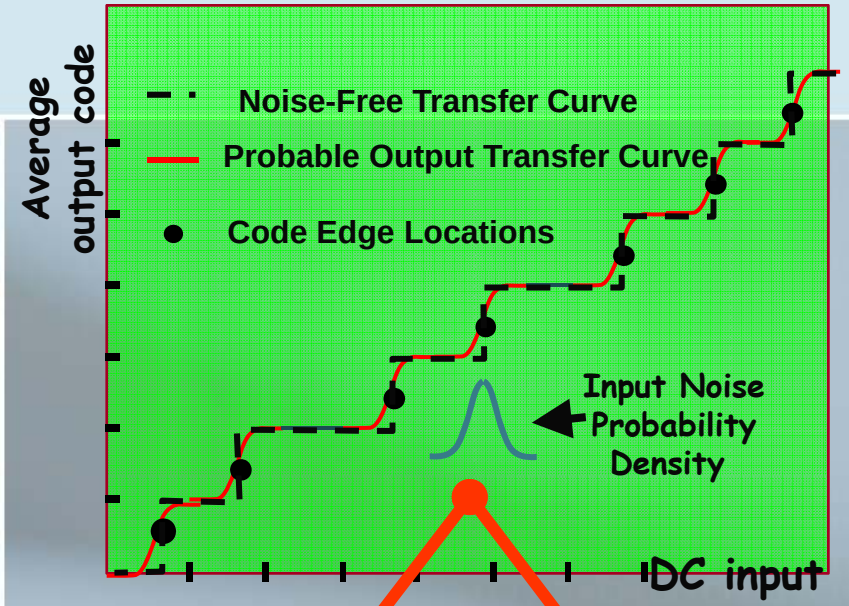
Linearity results extracted from the histogram method (1MHz)

The Clock noise in S&H (layout problem) and capacitors Mismatch leads to distortions and **high INL**;

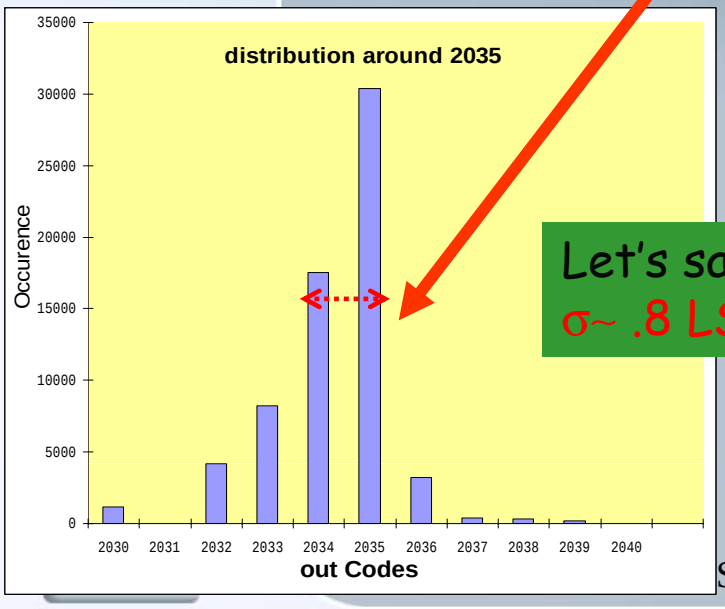




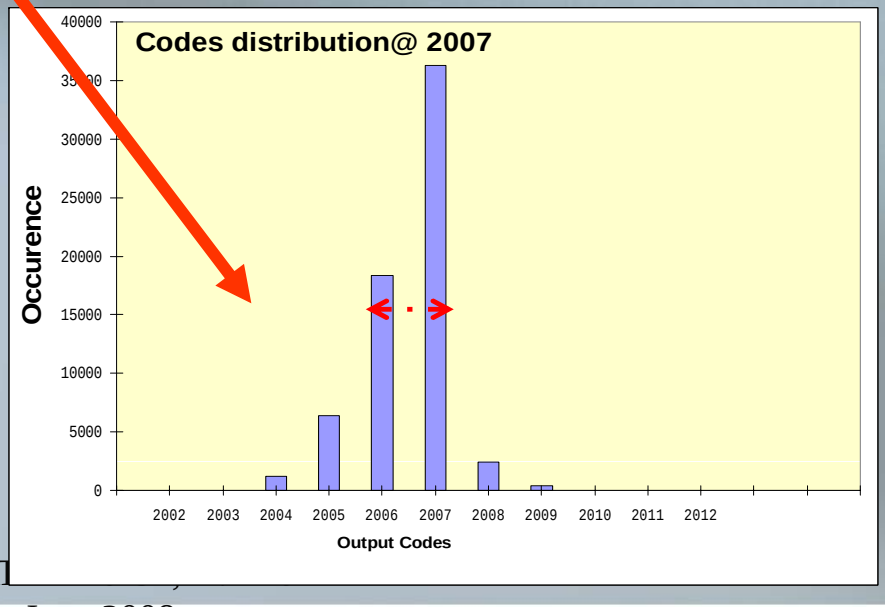
Code-Edge Noise results @25Mhz



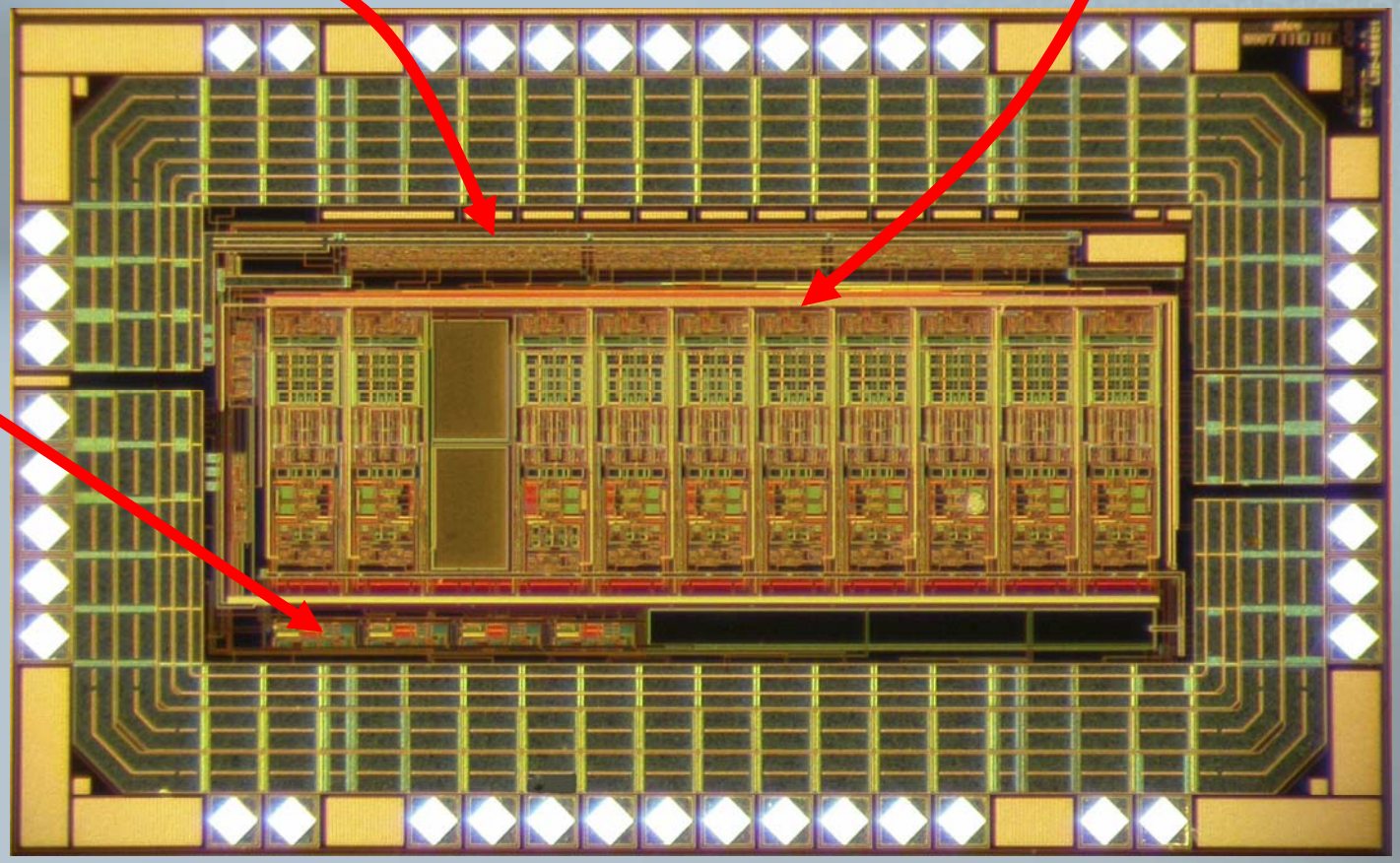
σ~ .8 LSB for SH+ADC
Means .8/√2 ≈ .6 LSB for ADC alone



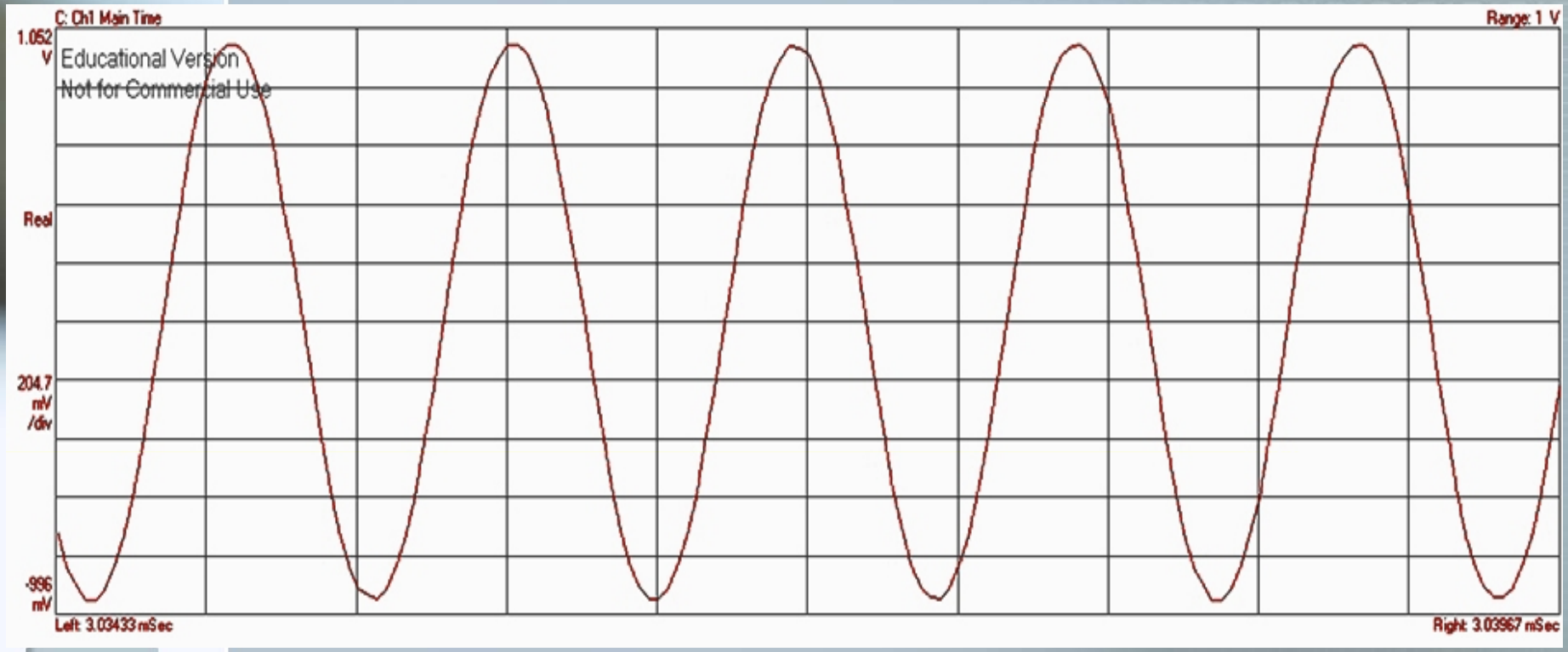
Let's say :
σ~ .8 LSB



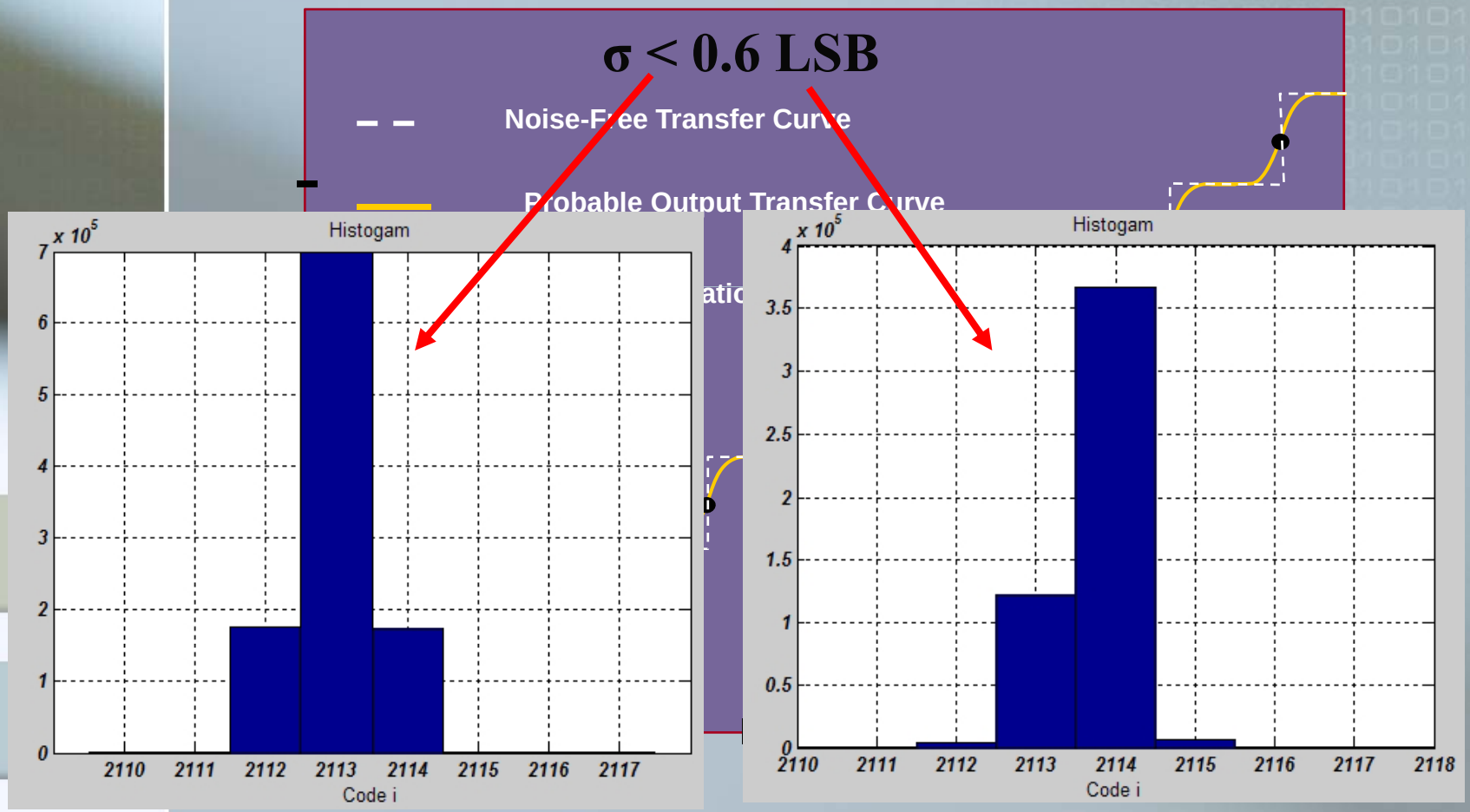
ADC 12b Without S&Hold, 30MHz&35mW : April 08=> prototype n-1



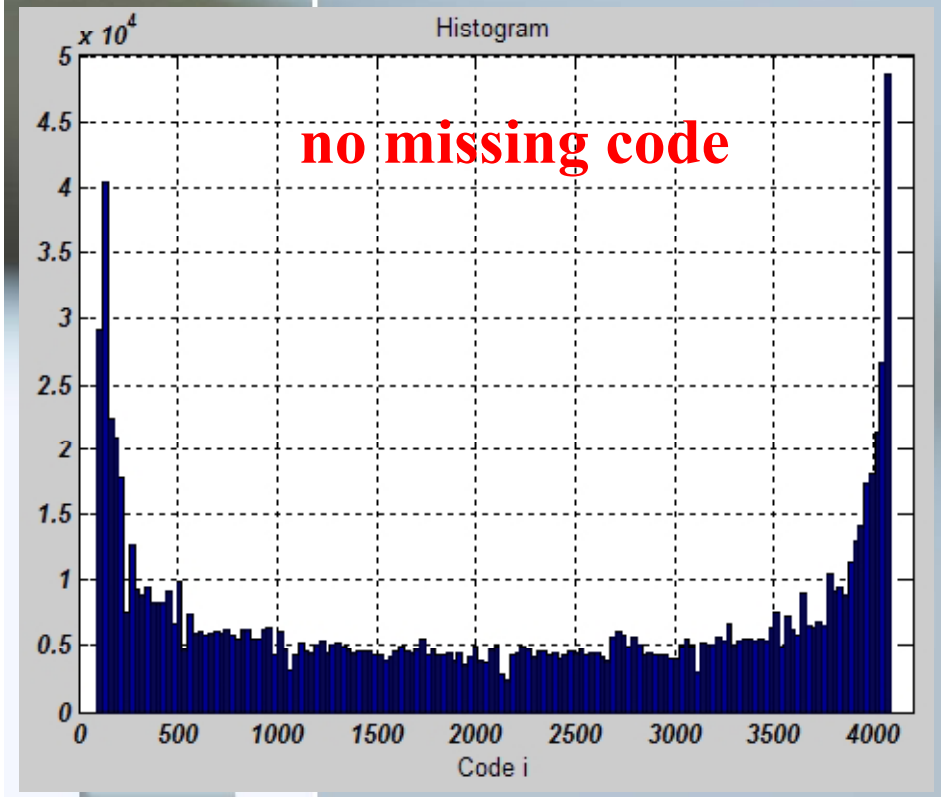
2 Vpp Dynamic range at 30MHz ADC Pipeline

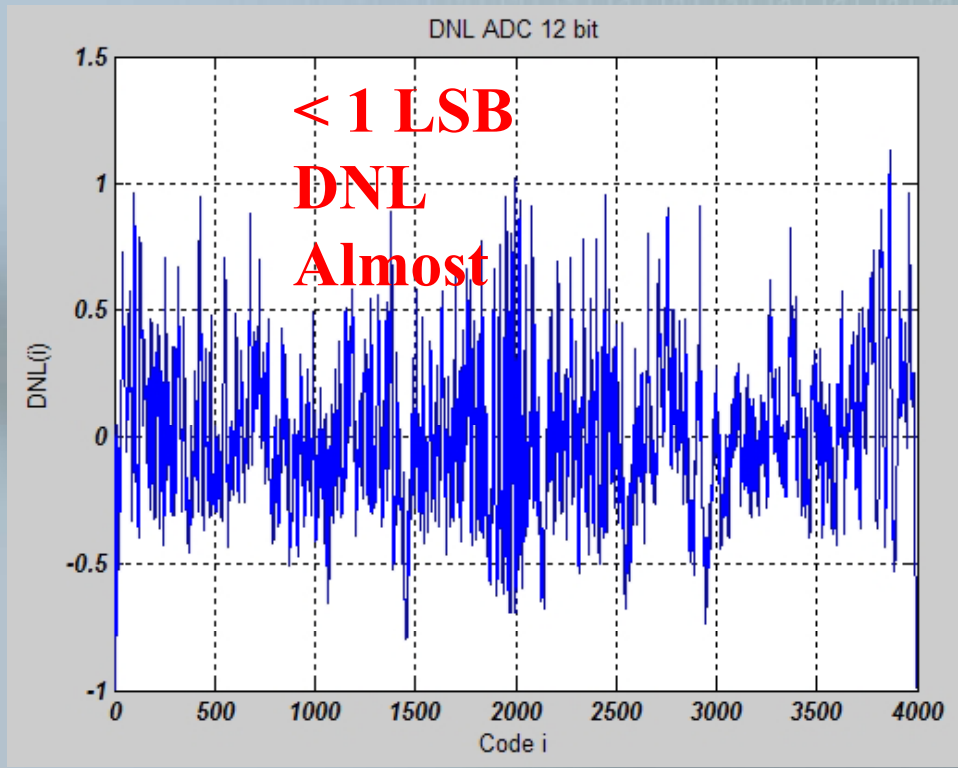


Noise distribution at 30MHz

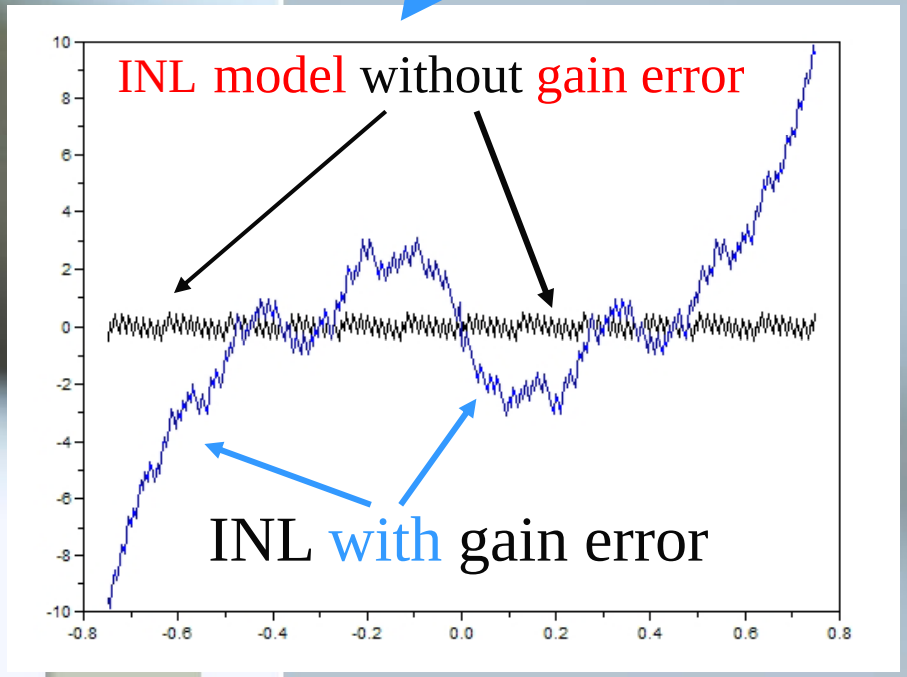


Missing code and DNL at 30MHz on a 1MHz sinus

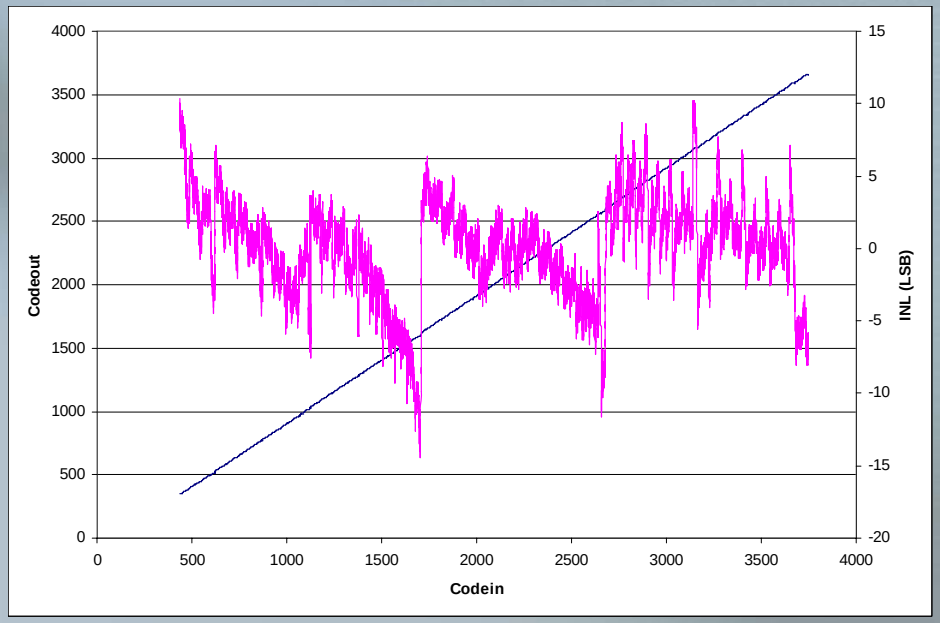




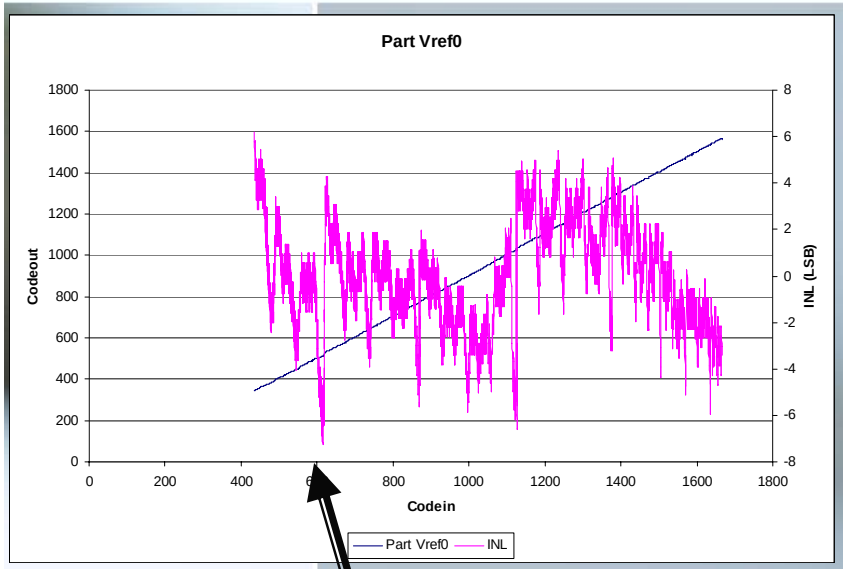
INL from ADC Model with and without gain error versus testing results

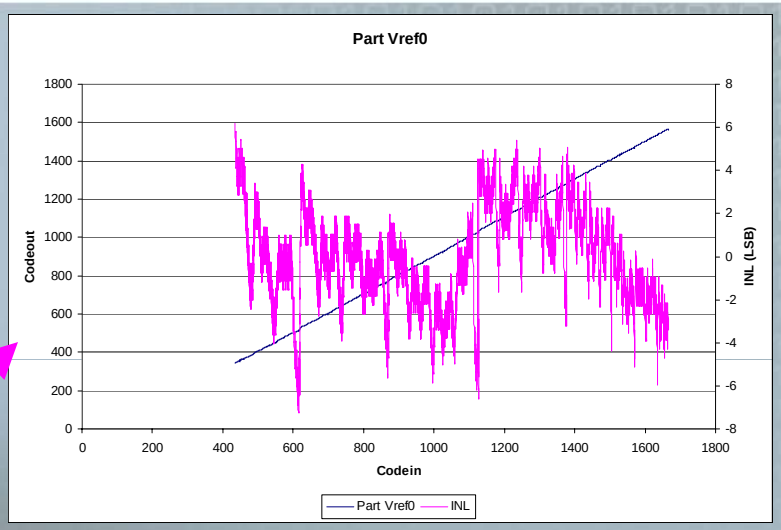


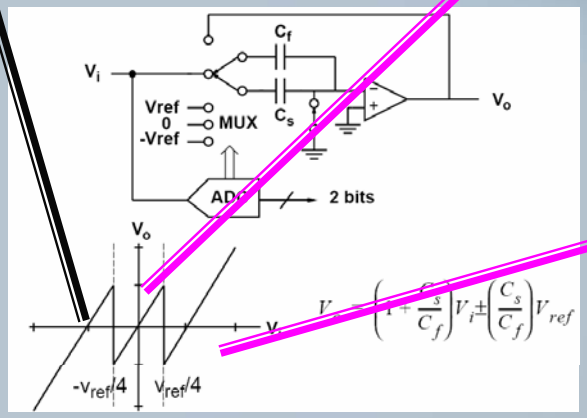
This shape (in **3 sections**) Corresponds to mismatch From the first stage.

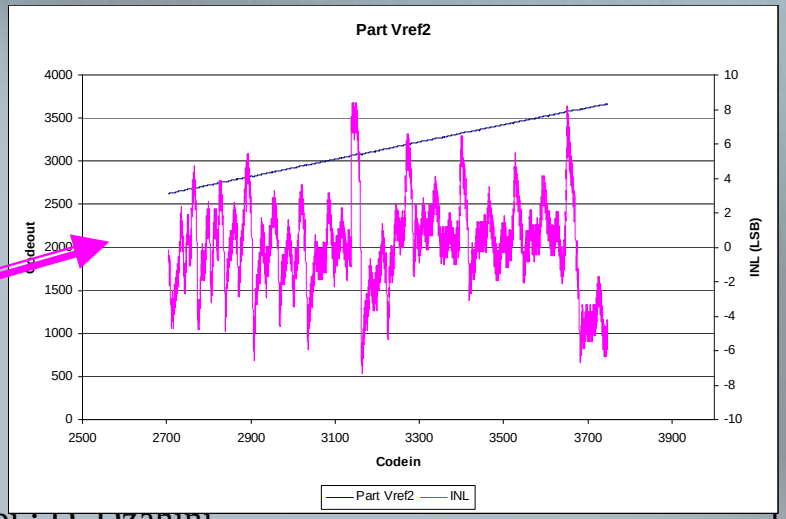


INL testing results in each section of the first MDAC stage





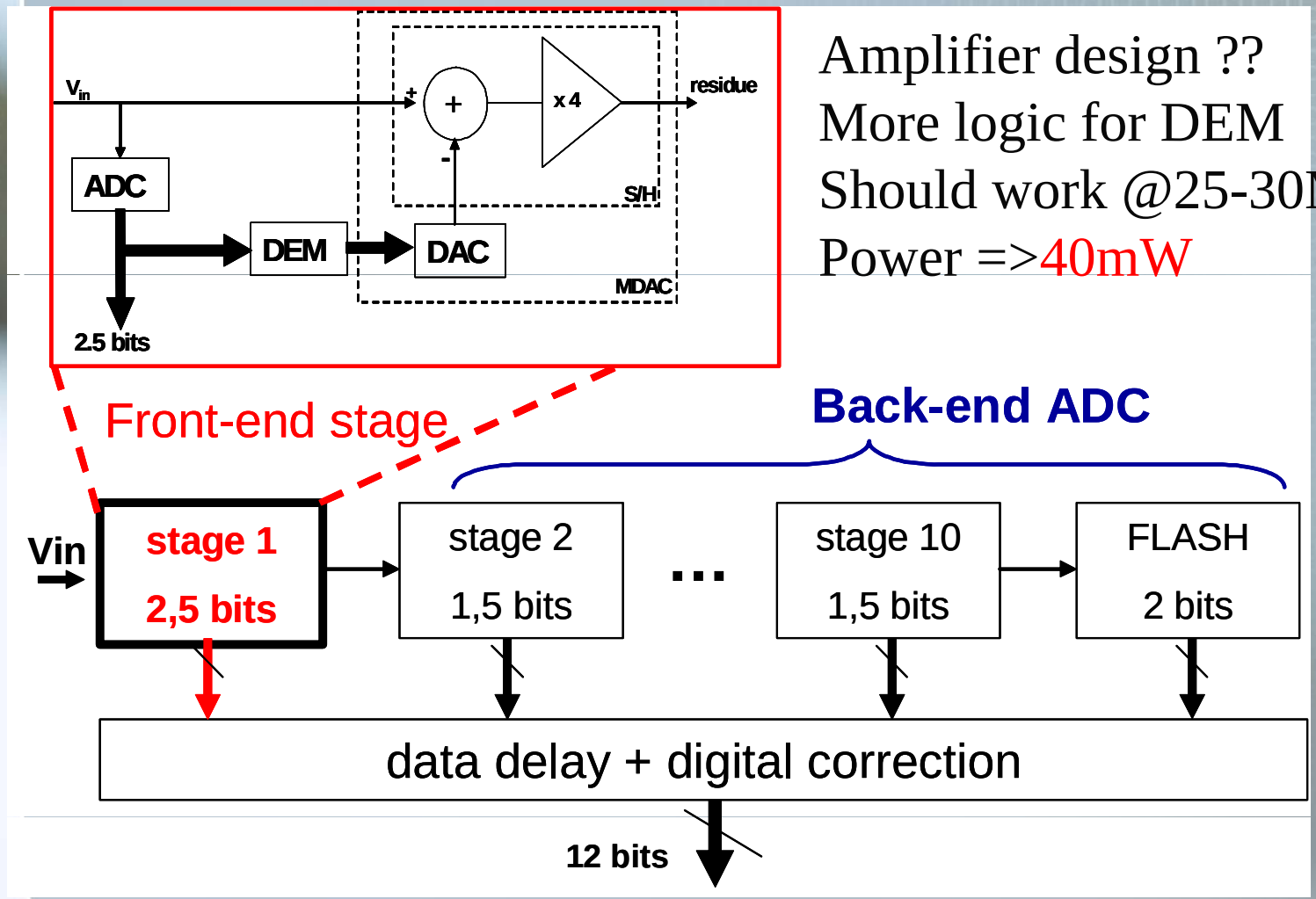




Trends for INL reduction in 12 bits

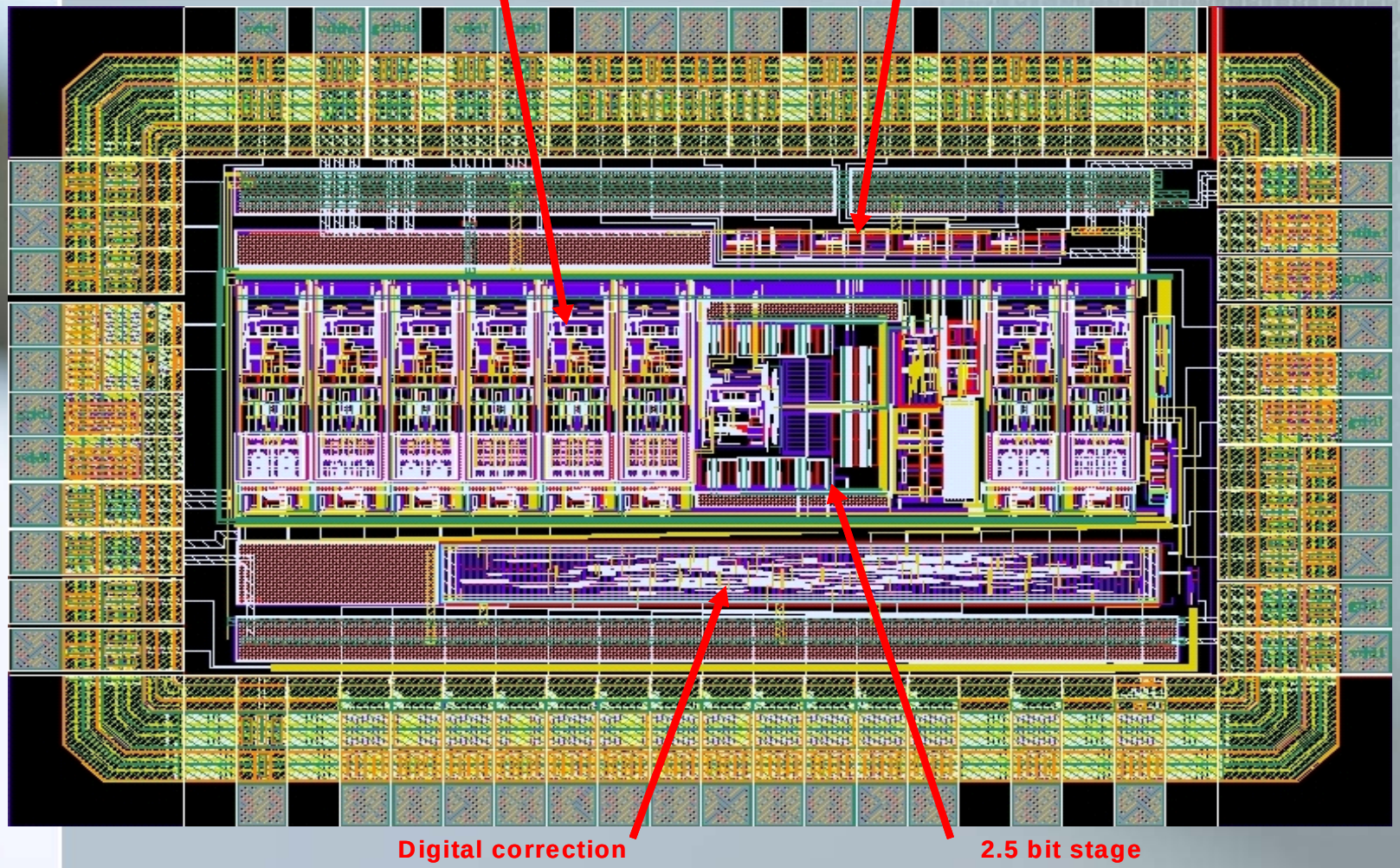
- Our Design reaches the limits for Cpoly matching in AMS CMOS process.
- Possible improvements using Cmim capa in SiGe.
- Increase the number of bits +DEM in the first stage=> Done.
- Digital gain correction algorithm very soon.
- Analog gain setting ??? *I do not like it.*

Prototype "n" submitted April 08

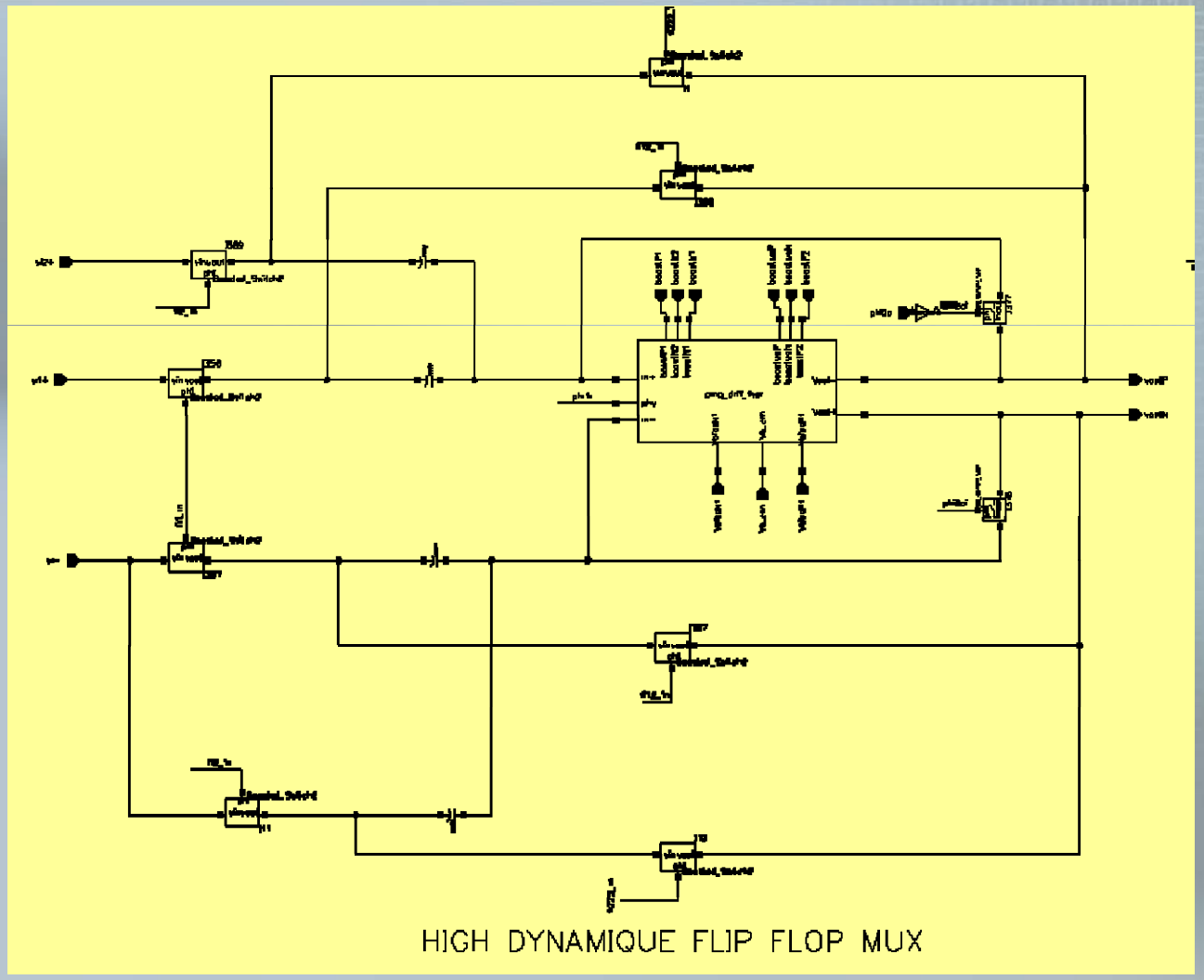


Amplifier design ??
More logic for DEM
Should work @25-30Mhz
Power =>40mW

Prototype April 08 (4mm² ou .9*1.9 mm² (coeur))

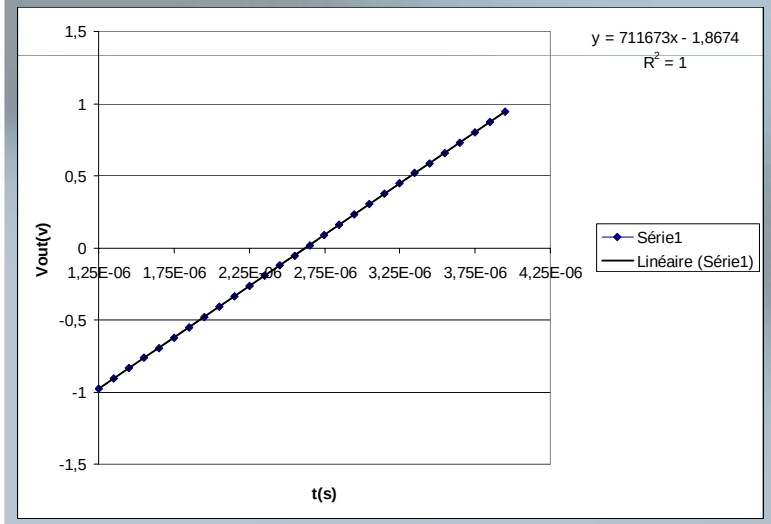


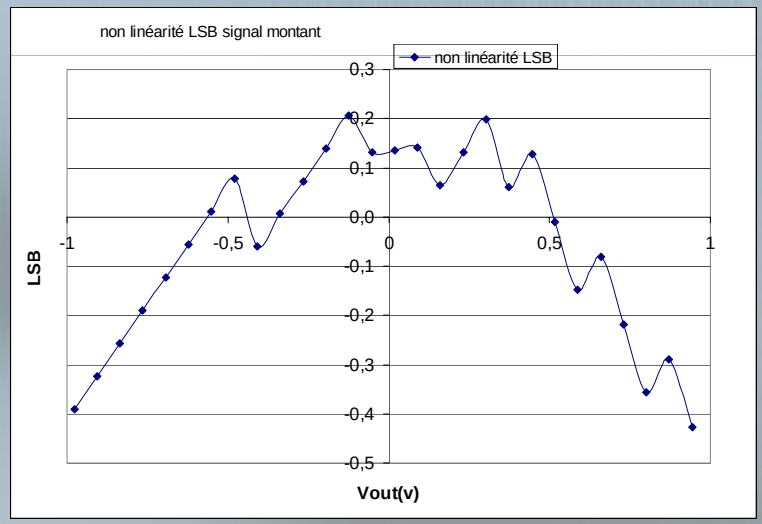
ANALOG FLIP FLOP MUX (12 bits)
Next spteps



HIGH DYNAMIQUE FLIP FLOP MUX

Beginning of simulations





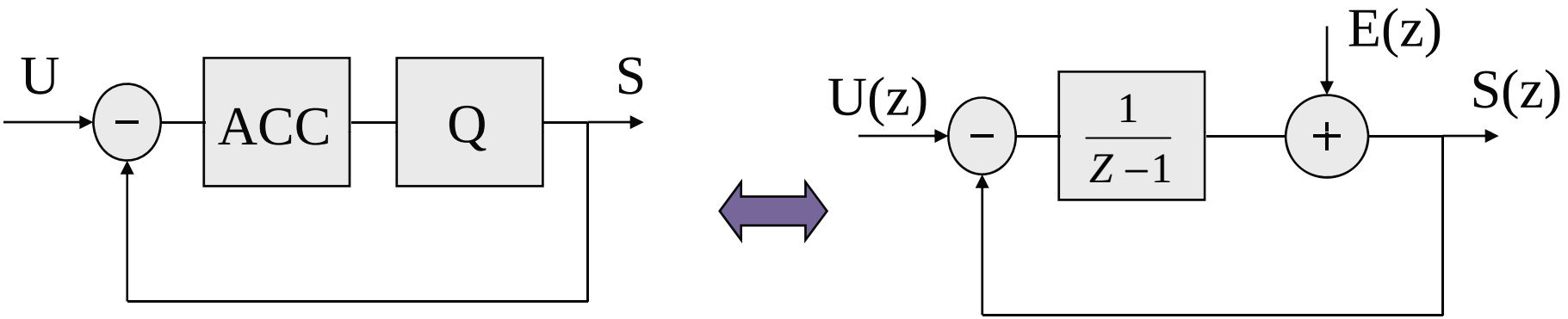
16 bits DAC for the CALIBRATION

Laurent Gallin Martel / Olivier Rossetto /D. Dzahini

Modulation Sigma Delta

Modulateur ‘’single stage’’ du premier ordre

E(z) : quantization noise



ACC somme les erreurs de quantification et agit comme un filtre passe haut du point de vue du bruit.

C’est le nombre d’intégrateurs qui détermine l’ordre du modulateur.

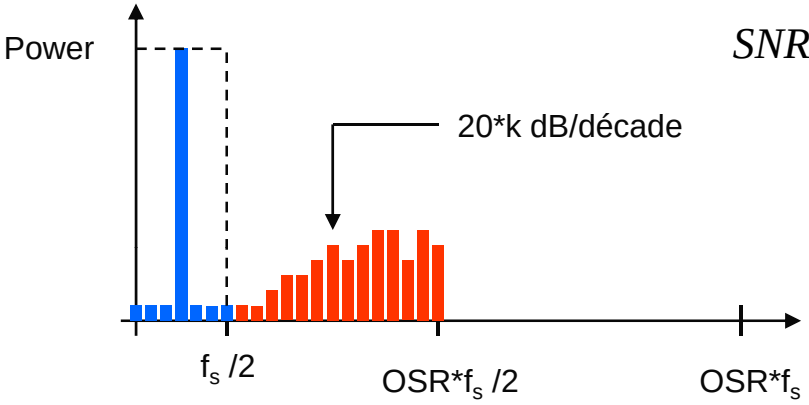
Le quantificateur peut avoir de 2 à m niveaux de quantification.

$$\begin{cases} S(z) = z^{-1}U(z) + (1 - z^{-1})E(z) \\ STF = Z^{-1} \\ NTF = 1 - Z^{-1} \text{ (high pass filter)} \end{cases}$$

laurent.gallin-martel@lpsc.in2p3.fr

Modulation Sigma Delta

SNR d'un convertisseur Sigma Delta

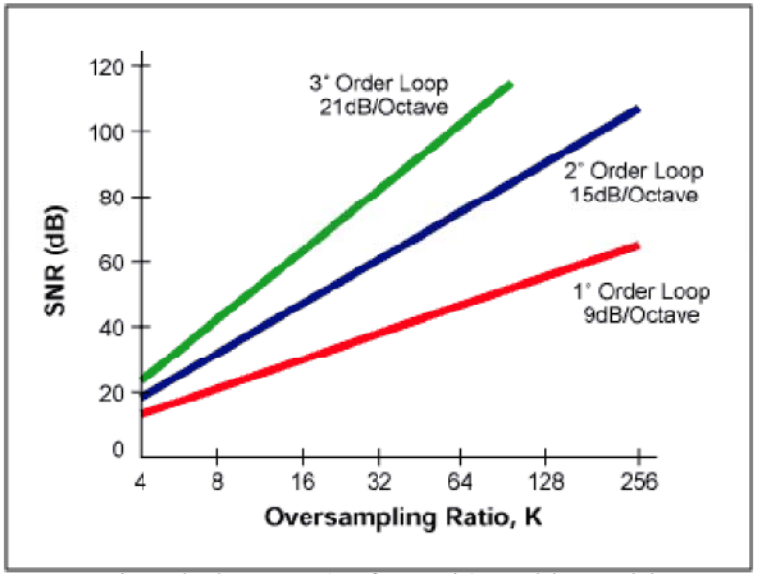


$$SNR_{dB} = 6.02N + 1.76 + 10\log\left(\frac{2k+1}{\pi^{2k}}\right) + 10(2k+1)\log(OSR)$$

- N : nombre de bits du quantificateur
- k : ordre du modulateur
- OSR : taux de sur échantillonnage

SNR d'un modulateur 1 bit

http://www.maxim-ic.com/appnotes.cfm/an_pk/1870

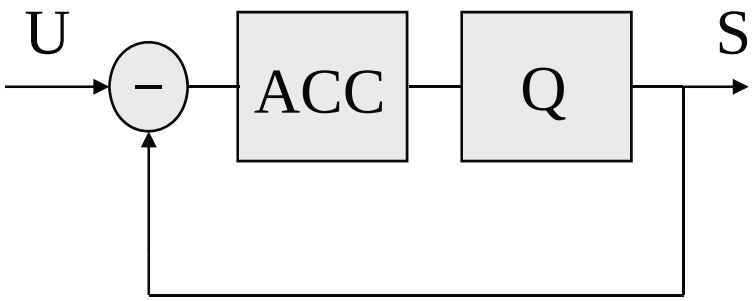


L.Gallin Marte

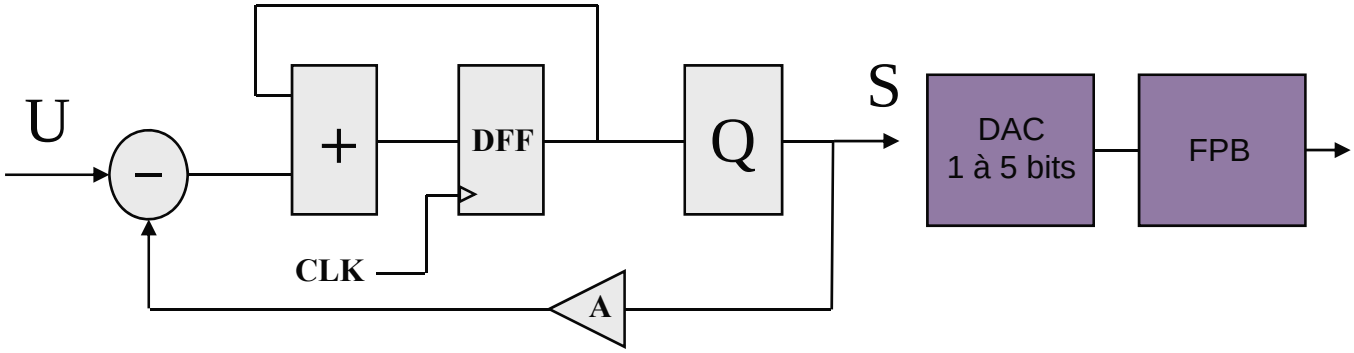
laurent.gallin-martel@lpsc.in2p3.fr

Modulation sigma delta

Modulateur ‘’single stage’’ du premier ordre – Implémentation pour un DAC



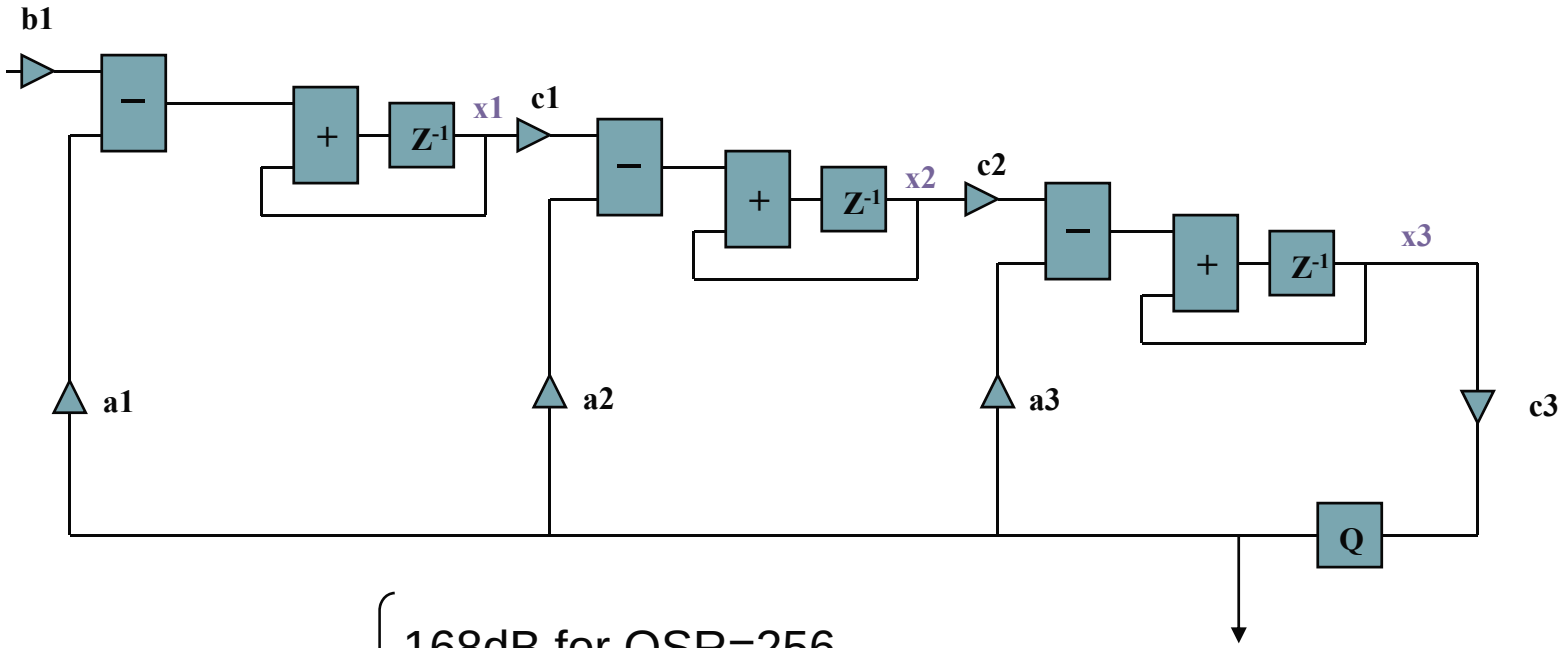
Quantificateur à m niveaux => nb de bits n = log(m)/log(2)
=> de 2 à 32 niveaux de quantification



laurent.gallin-martel@lpsc.in2p3.fr

DAC sigma delta 16 bits

3e ordre – 9 niveaux de quantification



SNR théorique :

- 168dB for OSR=256
- 147dB for OSR=128
- 126dB for OSR=64

La toolbox permet de déterminer la valeur des coefficients
ainsi que le taille des bus xn pour atteindre (approcher) le SNR théorique.

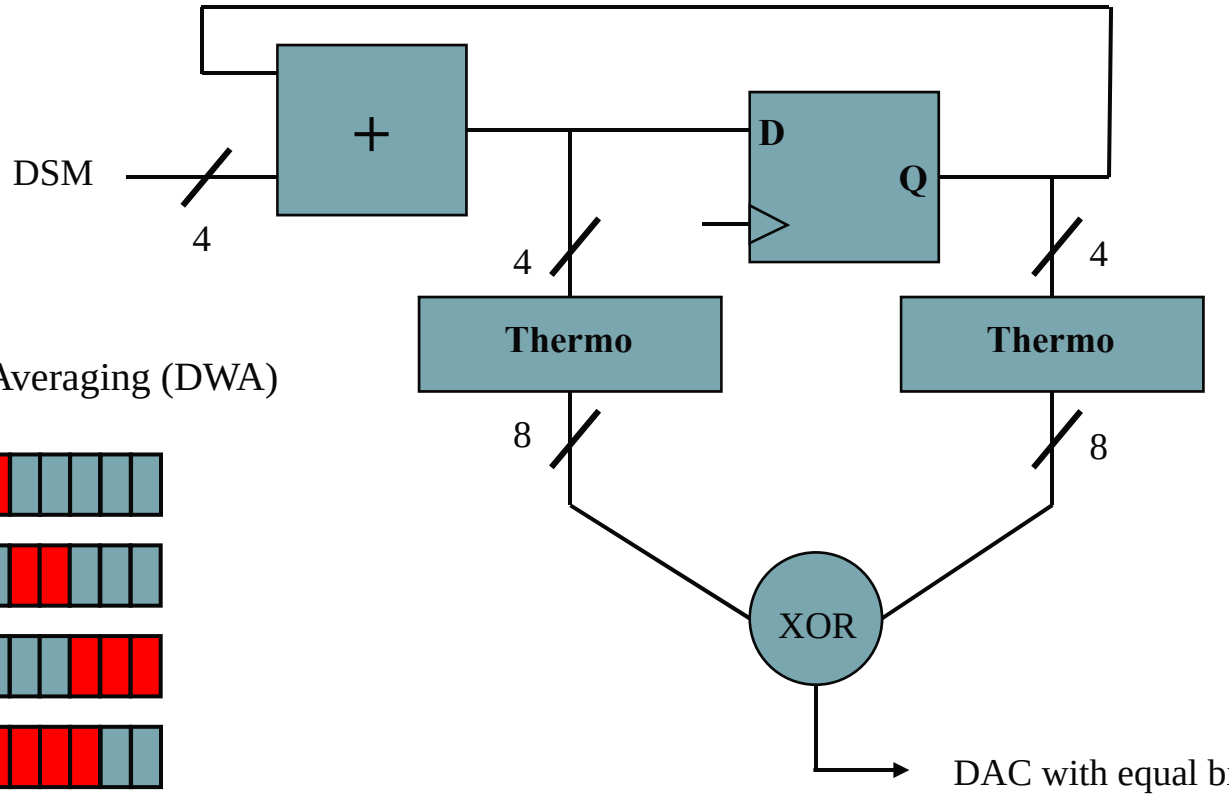
Une fois le DAC mis au point avec la toolbox on peut le simuler avec
n'importe quel langage

```
S=c3*x3;
if (S>4) S=4;
if (S<-4) S=-4;
x3=x3 + c2*x2 - a3*S;
x2=x2 + c1*x1 - a2*S;
x1=x1 + U - a1*v;
```

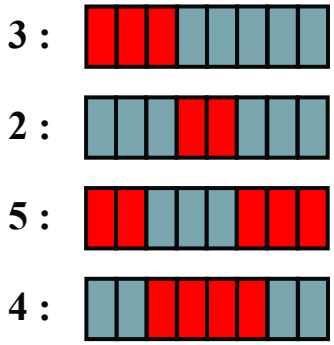

laurent.gallin-martel@lpsc.in2p3.fr

DAC sigma delta 16 bits

Dynamic Element Matching



Data Weighted Averaging (DWA)



Random re-mapping turns elements mismatch into white noise.

DWA re-mapping shapes DAC errors noise.

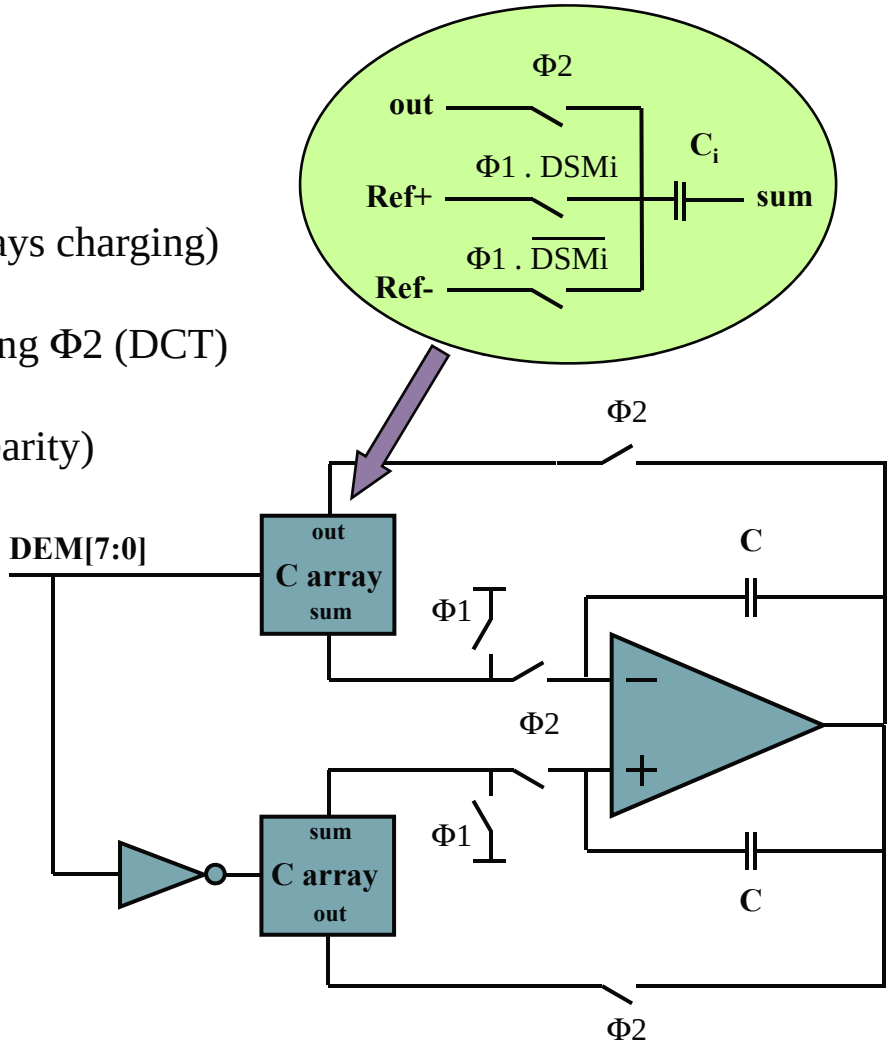
laurent.gallin-martel@lpsc.in2p3.fr

DAC sigma delta 16 bits

DAC 9 niveaux

Based on Direct Charge Transfert (DCT) :

- DAC operation is performed during Φ1 (capacitor arrays charging)
- Charges are shared with the feedback capacitor C during Φ2 (DCT)
- Ref voltage is limited by the OTA dynamic range (linearity)



Low pass DCT :

$$H(z) = \frac{1}{1 + a - a \cdot z^{-1}}$$

$$a = \frac{C}{\sum c_i}$$

laurent.gallin-martel@lpsc.in2p3.fr

DAC sigma delta 16 bits

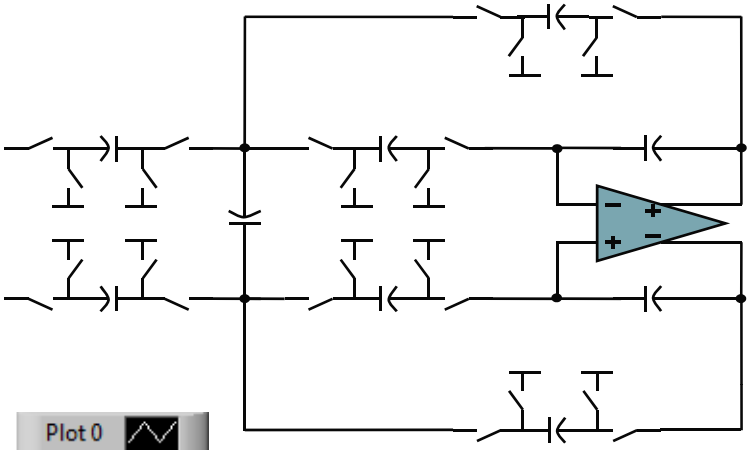
Filtre passe bas du 4e ordre (capa com)

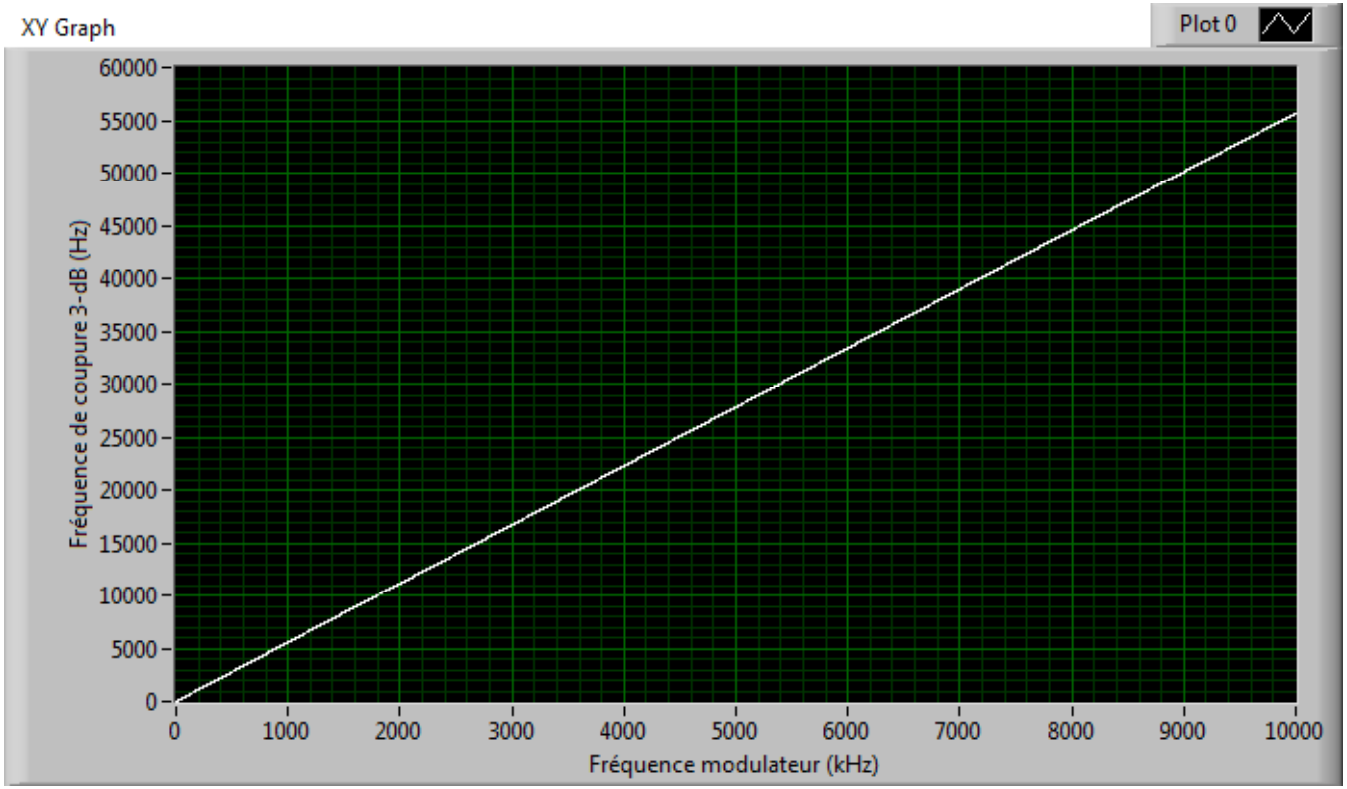
Architecture Multi FeedBack (MFB)

2 filtres du second ordre (identiques) cascadés

Simulations trop longues avec SPECTRE (> à 1 semaine !!!)

2 x =>

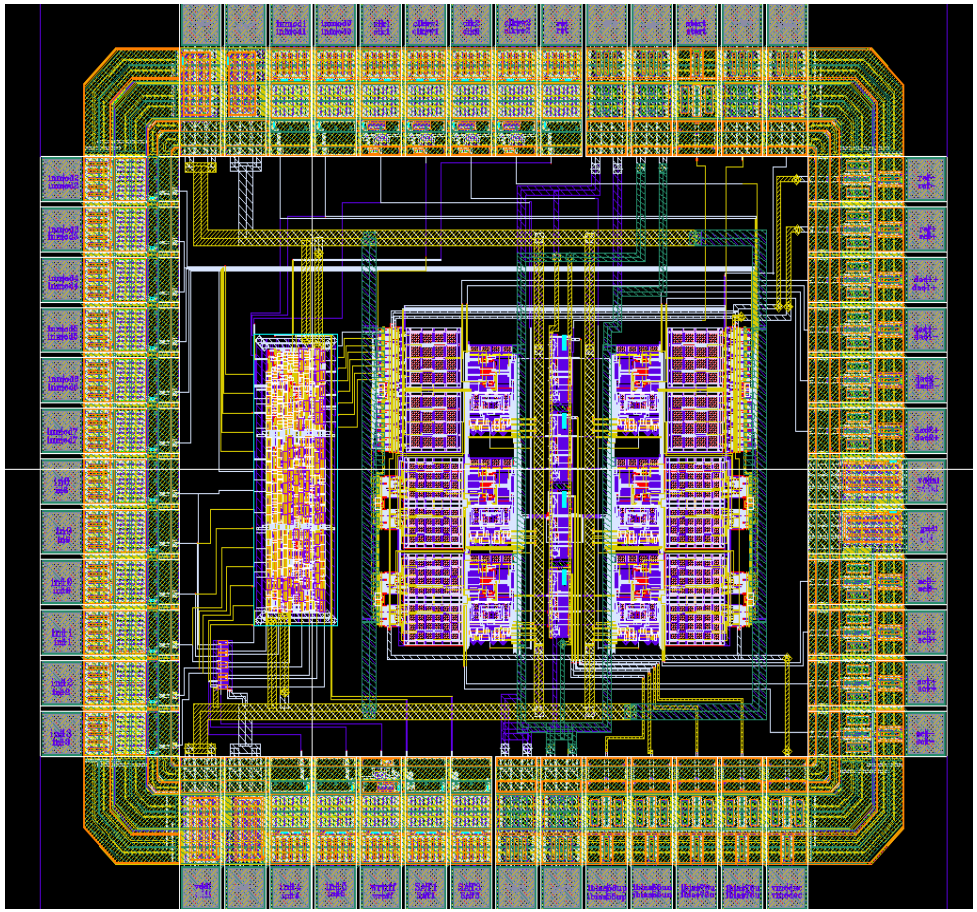




laurent.gallin-martel@lpsc.in2p3.fr

DAC sigma delta 16 bits

Layout



AMS CMOS 0.35µ

Surface puce : 2000 x 2200 µm^2

Surface DSM : 650 x 200 µm^2

Surface 1 analogique : 700 x 500 µm^2

Boîtier PLCC52

Le chip contient un DAC 16 bits complet et une deuxième partie analogique (DAC 3 bits + FPB) qui peut être connectée à modulateur externe (FPGA) .

laurent.gallin-martel@lpsc.in2p3.fr

DAC sigma delta 16 bits

Caractéristiques (provisoires)

La carte de test utilise un ADC 16 bits (AD7621) qui présente un SNRmax de 90dB

Le bruit RMS obtenu avec cette carte est de 0.8 LSB => $20\log\left(\frac{32768}{0.8*\sqrt{2}}\right)=89dB$

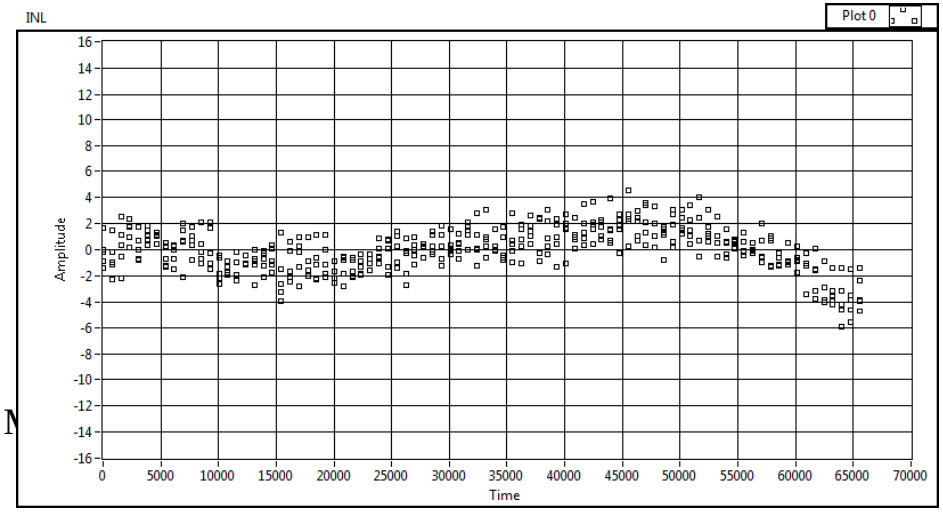
Le SNR obtenu pour le DAC 16 bits est de :	70dB (bruit RMS # 5 LSB)	pour Fdsm = 3 MHz, Fsinus = 12kHz
THD :	85dB	
INL :	+/- 3LSB	
Consommation partie numérique :	70µW @ 250kHz (simulation)	
Consommation partie analogique :	11mW (simulation)	

Il reste à mesurer la DNL et surtout à comprendre pourquoi le SNR est si mauvais.

Pour un OSR de 250 : SNR théorique # 160dB !!!

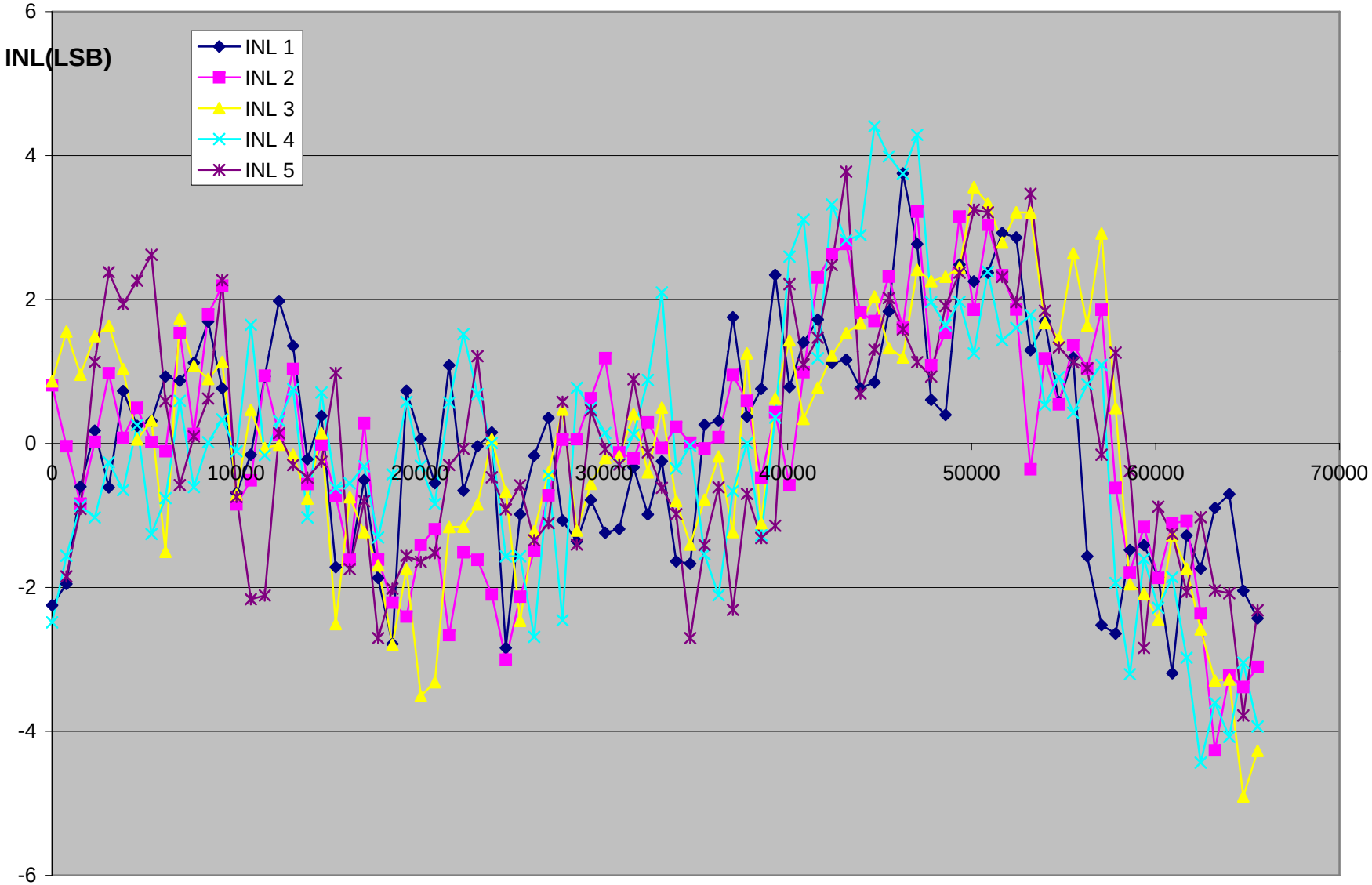
D'autres DSM peuvent être testés dans le FPGA

=> Encore beaucoup de travail



laurent.gallin-martel@lpsc.in2p3.fr

Testing results2



OUTPUT Codes rms noise