Connected Consumer Technology

Silicon & Software Systems

Digitally Enhancing Dynamic Linearity in High-Resolution A/D Converters

João Goes, Silicon & Software Systems (S3)

NMI Event, Dublin, 22nd. October 2009

Outline

- Why digitally assisted analogue circuits?
- Why digitally-assisted (DA) ADCs?
- Existing DA self-calibration solutions
- Why on-chip thermal noise is a good stimulus?
- Histogram-based algorithm
- A/D system architecture
- Required analog blocks to assist calibration
- Digital circuitry in FPGA
- Integrated prototypes in 90nm Logic CMOS
- Measured results
- Conclusions

Why digitally-assisted analogue circuits?

Nanometre; Digital shrinks – Analogue doesn't really – digital complexity comes almost at virtually zero incremental cost.

High performance Analog circuits are getting more difficult to design with scaling-down of CMOS technology:

- **The continuous reduction of the channel length: velocity of saturation, output resistance and intrinsic gain reduction;**
- **Open-loop amplifier gains over PVT, increasingly sensible to degradation;**
- **Reduced power supply voltage (Breakdown Voltages and Supplies decrease faster than V_T).**

In other hand, digital is achieving higher performances with technology scaling:

- **Faster (f_Ts are increasing);**
- **Smaller area for the same function; or more functions for the same area;**
- **Less power dissipation (for the same Fs);**

<u>Examples @ S3:</u> PLL VCO Calibration; RC Filter Calibration; Radio LO Feed-through Calibration; IIP2 calibration; IQ Mismatch Calibration; <u>ADCs</u>.

Why digitally-assisted (DA) ADCs?

- Accuracy of digital algorithms and time references can be used to compensate for the variations inherent in IC processing;

- This extra digital processing power can be used in the assistance of analog circuits by means of errors/distortion measurements followed by injecting correction/calibration countermeasures;

- Due to the digital output nature of ADC's and remembering that the correction/calibration codes are primarily digitals ones, these type of circuits are intrinsically well suited for DA Analog;

- In particular, in ADCs, total power dissipation and area are mainly imposed by effective resolution (ENOB), conversion-rate, noise (SNR) and matching & linearity (THD) specs. If digital assistance is used to improve matching & linearity in ADCs, this term gets out of the constraints list.

- Hence, small capacitance values and/or transistors aspect ratios can be used and this translates directly into significant power and area savings!

Existing self-calibration solutions

- **Dynamic Linearity of CMOS processes limited to 10-bits. To reach ENOBs above 10b either trimming or self-calibration has to be used;**
- **Trimming is too expensive and analogue self-calibration techniques are too sophisticated and complex (hard to implement);**
- **Digital domain calibration techniques are easier to implement and do not require fancy analogue circuitry. However, they require extra redundancy (meaning additional power dissipation) and several significant modifications in the ADC;**
- **How to digitally calibrate an ADC without requiring any modifications in it?**
  - **Stimulate the ADC with an appropriate analogue stimulus and apply, in the digital domain, an histogram-based algorithm to get the calibrating-codes.**
- **Is it possible to calibrate an ADC without modifying it (*i.e.* treat it as a "*black-box*")?**
  - **Yes, as long as we can predict where to find the main errors.**

Why on-chip thermal noise is a good stimulus?

- **Thermal (Gaussian noise) should be the best one because is based on a physical and random mechanism (sine waves are not pure enough, ramp signals are not linear enough).**

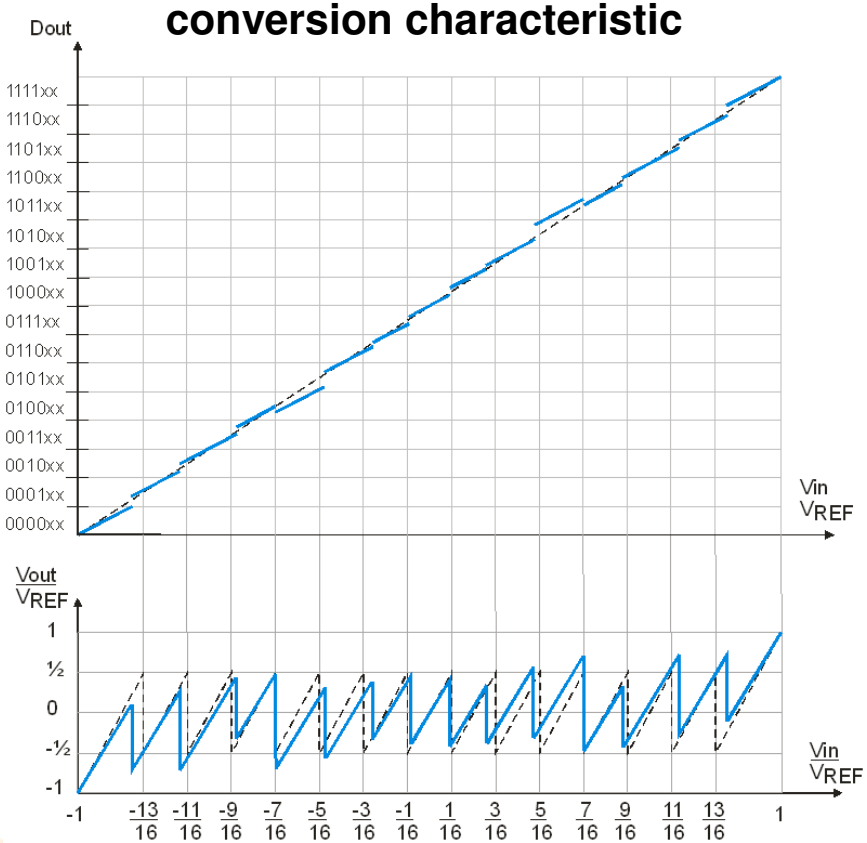
- **Some other advantages of using thermal noise:**
  - **Wide-band thermal noise is relatively easy to generate on-chip using a high closed-loop gain amplifier and input resistors as main noise sources;**
  - **The accuracy requirements of the standard deviation of the noise are relaxed (e.g. a 10-bit accurate σ is enough to calibrate a 13-bit ADC);**
  - **The input noise stimulus has not to be necessarily white as long as it remains Gaussian.**

- **How to deal with the uncertainties due to noise during the extraction of the calibrating codes?**
  - **Make use of histogram-based (cumulative) statistical methods.**

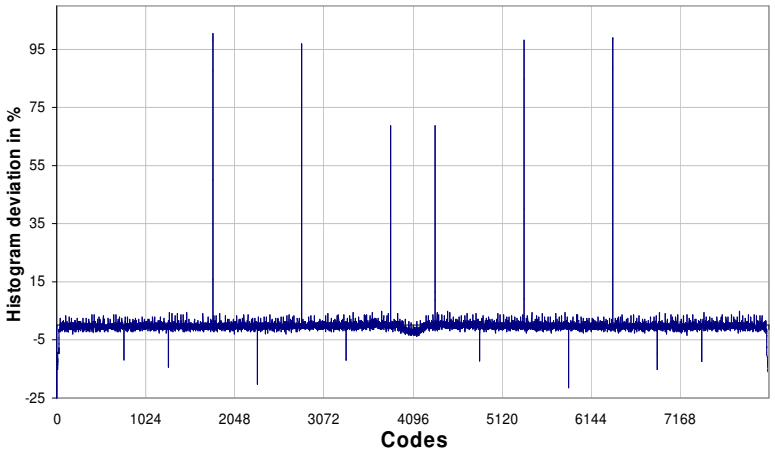
Histogram-based algorithm

Case-study: A 13-bit Pipeline ADC architecture with a 3.5-bit front-end stage (<u>but easily extendable to multi-stage algorithmic ADCs and multi-step flash ADCs</u>):

- **A 3.5-bit front-end stage produces 15 misaligned segments in the A/D conversion characteristic**



14 shifts in the conversion characteristic correspond to 14 spikes in the normalised output histogram



Histogram-based algorithm

STEP 1 (Calibration of the offset of the ADC): In this step the inputs are zeroed and 32 input samples are digitized and averaged; the averaged value is then used in the subsequent calibration steps as the offset of the ADC;

STEP 2 (Calibration of the σ of the GNG): the GNG noise is applied to the ADC through a PGA and the number of occurrences, for a total of 2^{26} samples, is counted in code intervals [89; 600] and [7592; 8103]; the PGA gain is adjusted by a SAA in order to obtain the correct value for the amplitude of the noise;

STEP 3 (Calculating histogram deviations): For a large number of samples, $NS = 2^{30}$ and a *binwidth*, $Bw = \pm 64$ codes, in our case, $D(i)$ is:

$$D(i) = [H(i)\ P^{-1}(i) - NS\ ].Bw/NS$$

where $P(i)$, stored in a ROM with 14 positions, is the probability of the bins in the ideal histogram obtained from a Gaussian distribution function with $\mu = 4096$ and $\sigma = 2048$

Histogram-based algorithm:

STEP 4 (finding the 15 calibrating codes): 14 values of $D(i)$ are used to calculate the deviations from the ideal transfer characteristic and the 15 calibrating codes. By exploring symmetry of the FE stage characteristic (optional), the 15 calibrating codes can be obtained as follows:

$$\text{calcode}(1) = -\text{GE} = -0.5 \cdot \sum_{i=1}^{7} \left[D(i) + D(16-i) \right]$$

$$\text{calcode}(i) = \text{calcode}(i-1) + 0.5 \cdot \left[D(i-1) + D(17-i) \right], \quad 2 \le i \le 7$$

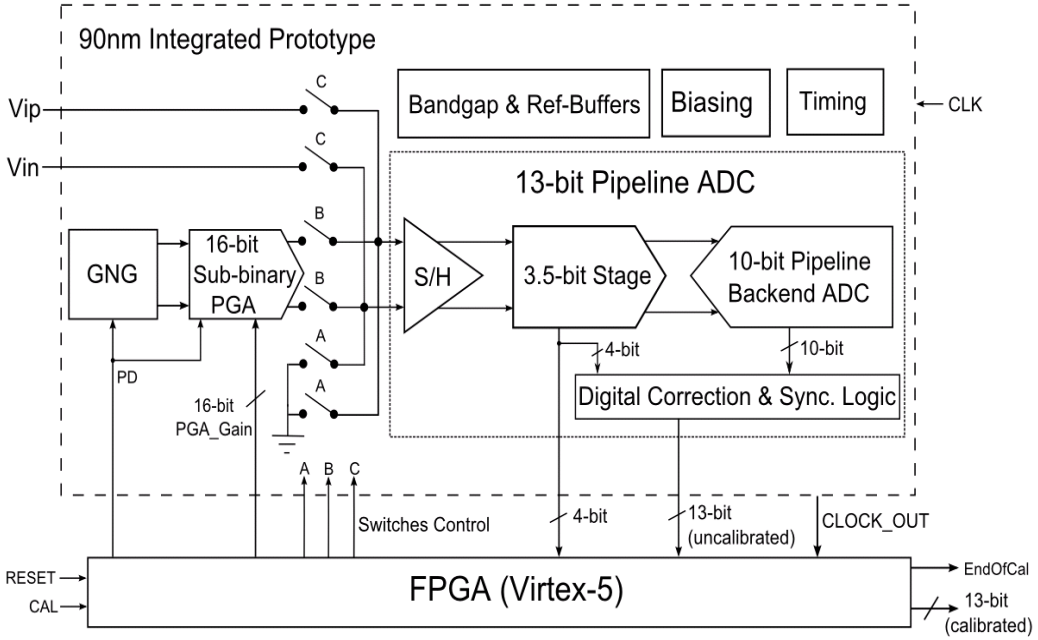
$$\text{calcode}(i) = -\text{calcode}(16-i), \quad 9 \le i \le 15; \quad \text{calcode}(8) = 0.$$

The resulting 15 calibrating codes are stored in a memory, which during normal conversion mode is addressed by the 4-bit output of the front-end quantizer, and the 15 output calibrating codes are added to the un-calibrated 13-bit output.

No multipliers are required since only elementary operations are used, Such as additions, subtractions, multiplications by constants and divisions by powers of 2

A/D system architecture

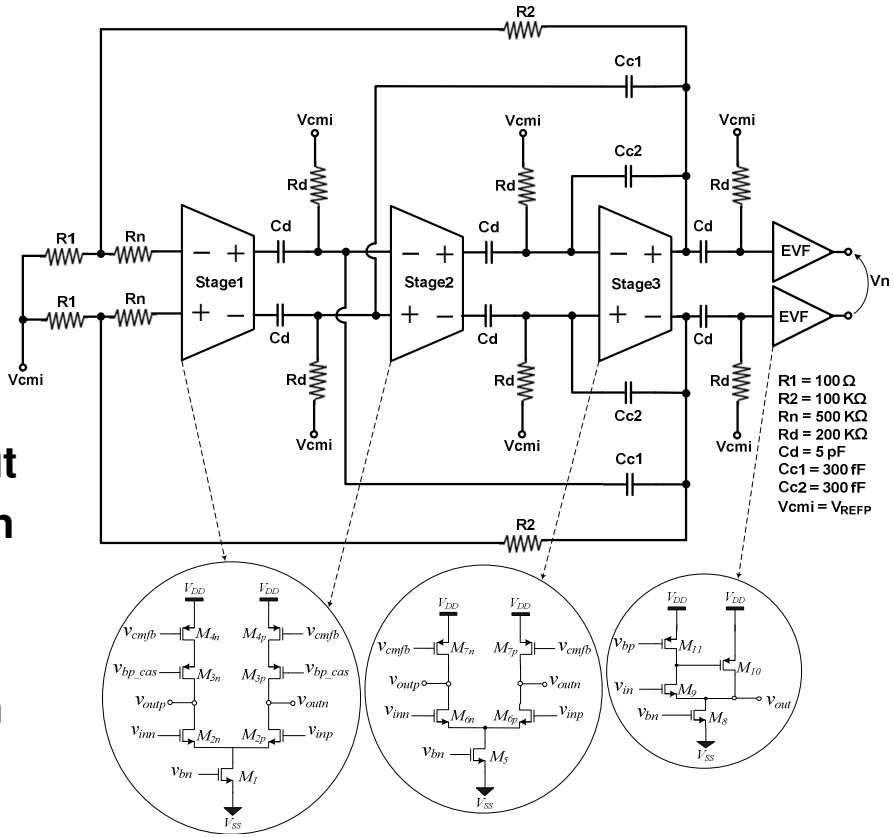
- A 13-bit pipeline ADC is associated to a noise generator (GNG) and to a 16-bit PGA.
- For flexibility in testing, an FPGA implements the calibration algorithm.
- The 13-bit ADC employs a front-end S/H followed by a 3.5-bit pipeline stage and by a 1.5-bit/stage 10-bit back-end pipeline ADC. The 14 output bits are digitally synchronized, and 13 bits are available after digital correction.



Required analog blocks to assist calibration: GNG

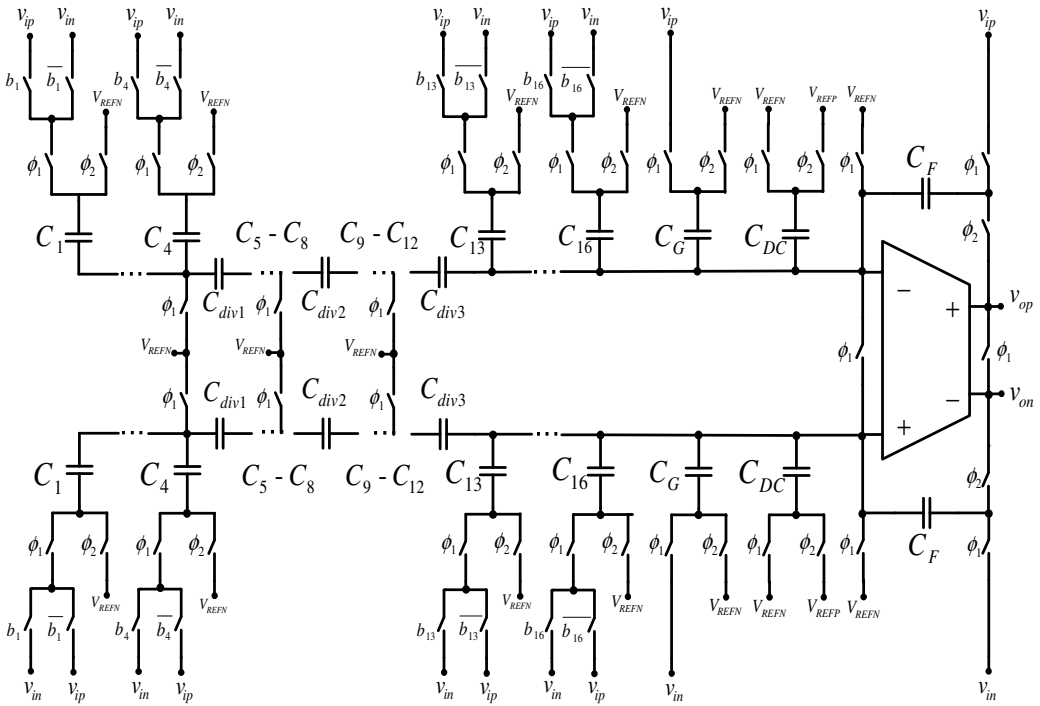
– The GNG circuit consists of a 3-stage nested-Miller compensated OTA followed by two enhanced voltage-followers (EVF) that drive the PGA.

– The thermal noise of the large input resistors (*Rn*) is amplified with a gain equal to *R2*/*R1*. To remove accumulated offset and 1/*f* noise, the 3-stages are AC coupled through *Cd* and *Rd*.



Required analog blocks to assist calibration: PGA

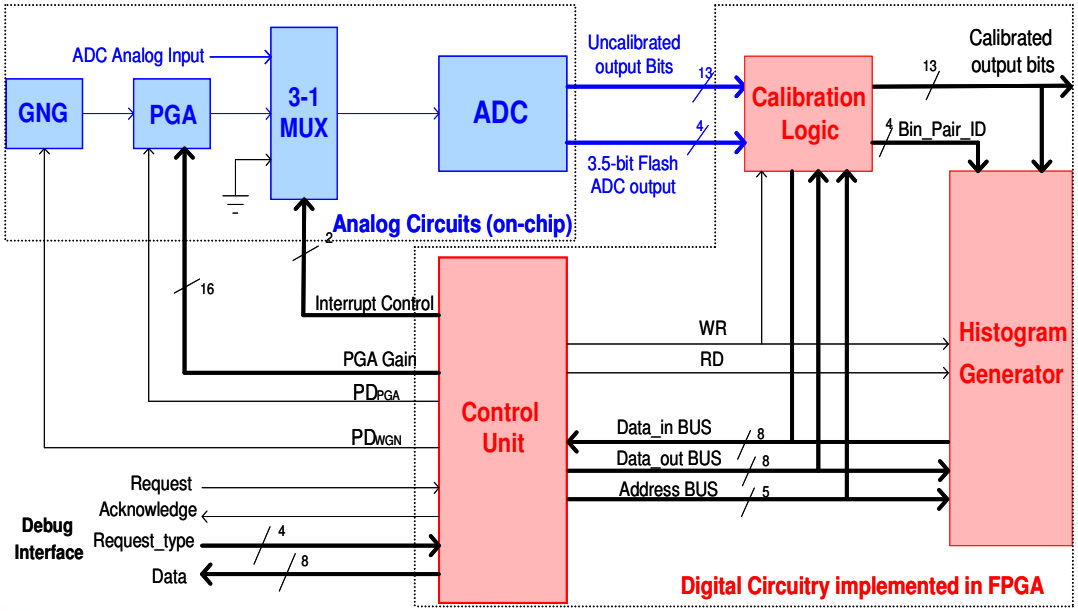
- Since the noise standard-deviation is PVT dependent, an SC 16-bit sub-binary PGA adjusts the σ to half the differential reference voltage with 10-bit accuracy;
- The average programming step was set to 1.7 (sub-binary), which ensures convergence of a successive-approximation algorithm.



Digital circuitry in FPGA

– The digital circuitry required to assist the algorithm consists of a <u>control unit</u>, a <u>histogram generator</u> and <u>some calibration logic blocks</u>.

A Virtex-5 FPGA was used to operate at 80 MHz to enable real time silicon verification in conjunction with the ADC chip. Since the digital circuitry is small, one of the smallest FPGAs of this family (e.g. a Virtex 2) could have been used. The hardware synthesis has also been performed for a Xilinx Spartan 3 FPGA, but it was not able to meet the 80 MHz requirement.

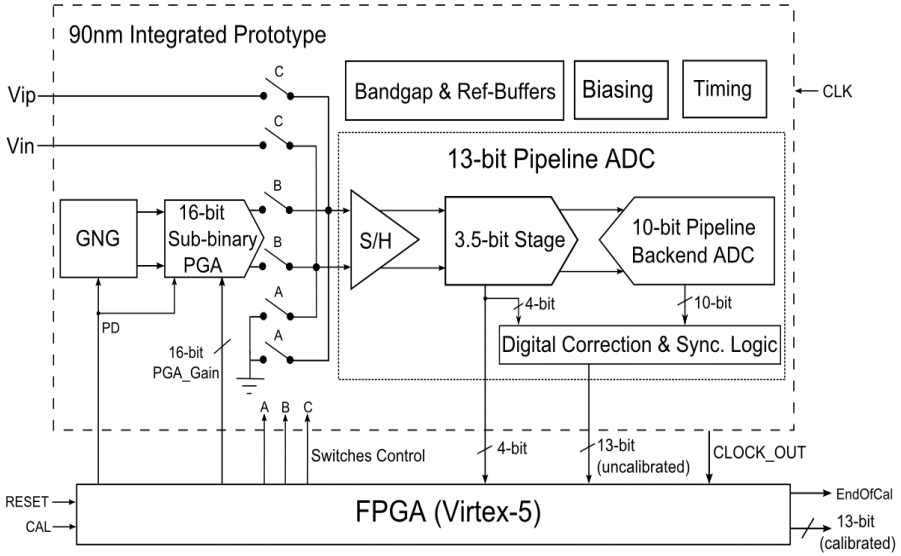


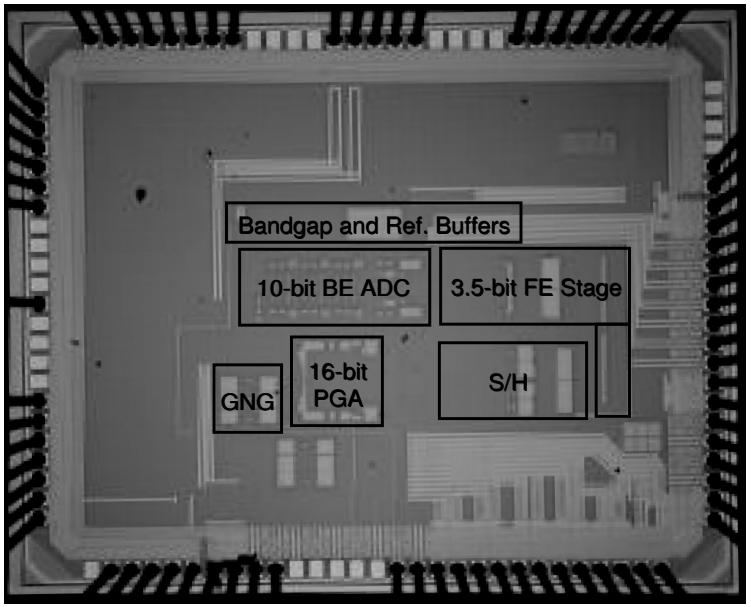
Integrated prototypes in 90nm Logic CMOS

An ADC prototype IC was fabricated in a pure 90 nm 1P8M CMOS logic process, in which, only six metal layers were used.

Die area (analogue circuitry) < 0.88 mm^2.

Extra Power for GNG + PGA ~ 20 mW (powered-down after calibration)



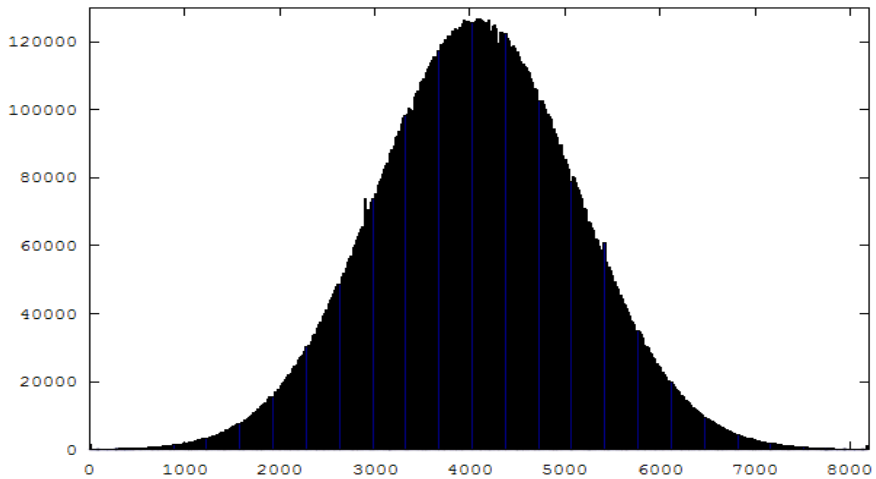


Measured results (1): GNG block

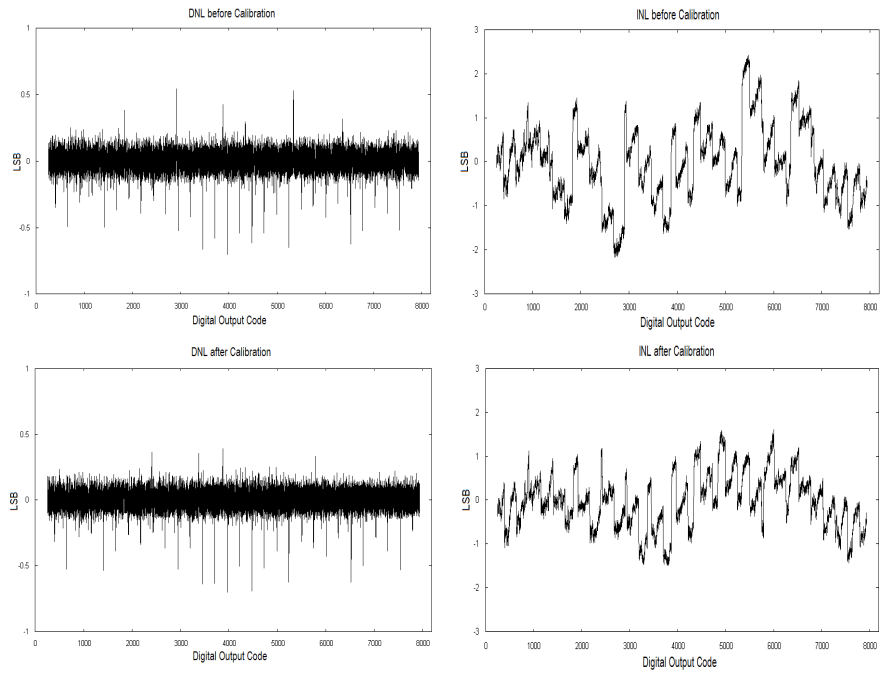
– The 13-bit ADC was used to digitalize the GNG noise at 80 MS/s, 20.8 million samples were collected at the outputs of the ADC and an histogram was computed. After statistical analysis, we found that both, the asymmetry and kurtosis coefficients are very close to the expected ideal values (zero).

MEASURED PERFORMANCE SUMARY OF THE GNG BLOCK.

Parameter	Measured Results	
	Fs = 80 MS/s	Unit
Mean	4148.0	LSB
Offset	52.0	LSB
Standard-deviation, σ	1052.0	LSB
Asymmetry coefficient	0.0000516	-
Kurtosis coefficient	-0.0018050	-



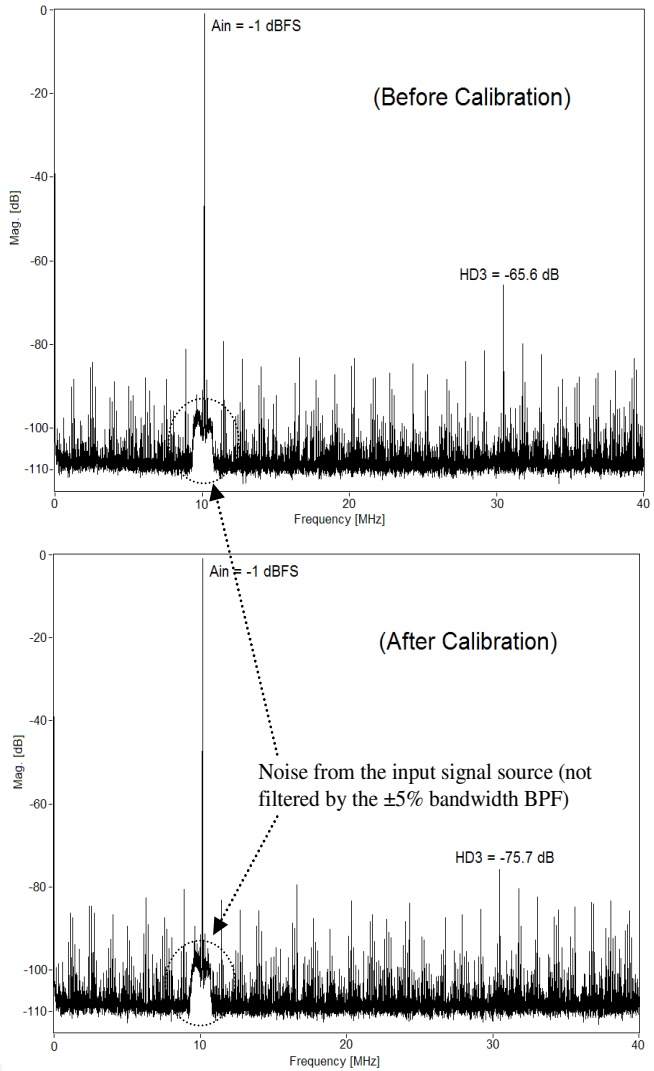
Measured results (2): DNL/INL @ 80 MS/s



– Calibration improves both characteristics and, after calibration, INL is always bounded to ±1.5 LSB. This limit is mainly imposed by the second stage in the pipelined chain (that corresponds to the first stage of the 10-bit back-end ADC) which has an 1.5-bit resolution and it is not calibrated.

Measured results (3): FFTs @ 80 MS/s

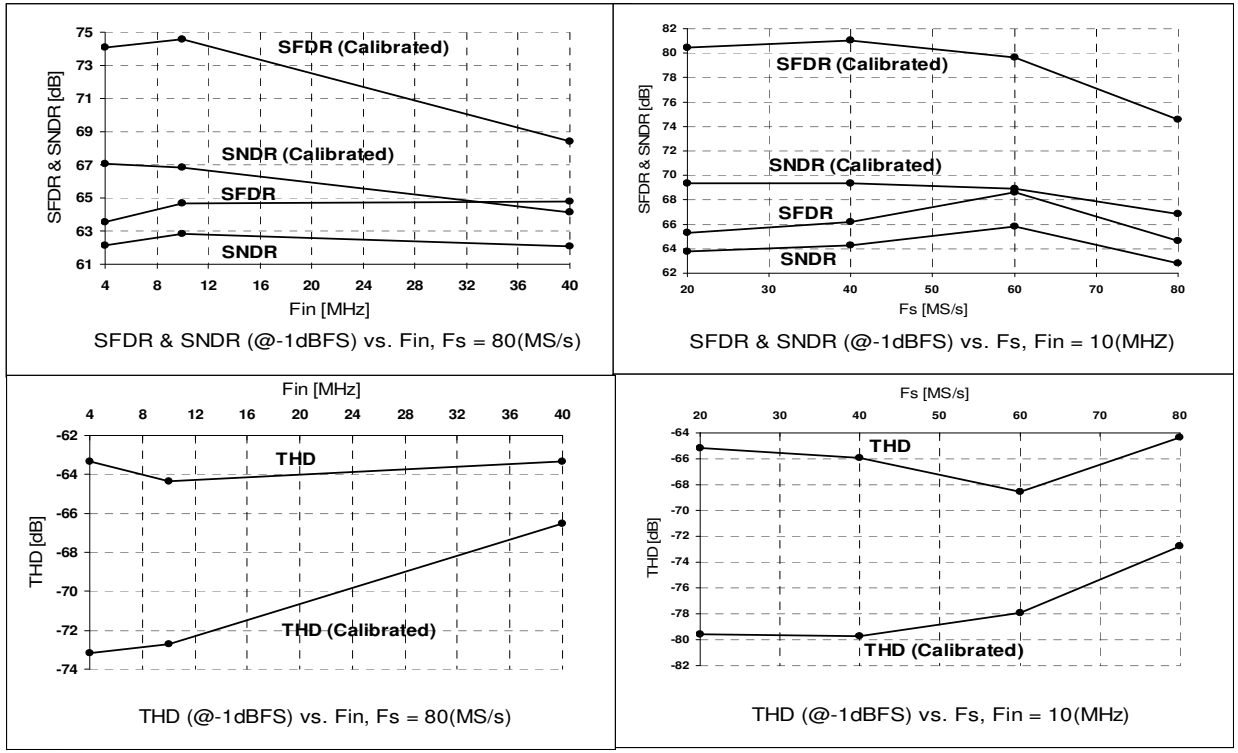
– FFT measurements demonstrate that SFDR (mainly dominated by HD3) is improved by 10 dB to 74.6 dB and THD is improved by 8.4 dB to -72.7 dB.



Measured results (4): SFDR, SNDR and THD

– SFDR, SNDR and THD are significantly improved after calibration for different values of *fin* and *FS*.

– The largest improvements, corresponding to over 14.8 dB in SFDR (to 81 dB) and 13.8 dB in THD (to -79.7 dB), were obtained at 40 MS/s.



Measured results (5): key measurements summary

	Before Calibration	After Calibration
Technology	90nm 1P 6M CMOS Logic	
Power dissipation @ 1.2V @ 80MS/s (ADC and reference buffers) @ 40MS/s (ADC and reference buffers)	85 mW 44 mW	
Die area	0.88mm^2	
Signal Range	1 Vp-p (differential)	
DNL (LSB @ 13-bit)	± 0.9 LSB	± 0.7 LSB
INL (LSB @ 13-bit)	± 4 LSB	± 1.5 LSB
SNR @ 40MS/s @ f_{in} = 10MHz SNR @ 80MS/s @ f_{in} = 10MHz	69.3dB 68.1dB	69.8dB 68.1dB
SFDR @ 40MS/s @ f_{in} = 10MHz @-1dBFS SFDR @ 80MS/s @ f_{in} = 10MHz @-1dBFS	66.2dB 64.7dB	81.0dB 74.6dB
THD @ 40MS/s @ f_{in} = 10MHz @-1dBFS THD @ 80MS/s @ f_{in} = 10MHz @-1dBFS	-65.9dB -64.3dB	-79.7dB -72.7dB
SNDR @ 40MS/s @ f_{in} = 10MHz @-1dBFS SNDR @ 80MS/s @ f_{in} = 10MHz @-1dBFS	64.3dB 62.8dB	69.4dB 66.8dB
ENOB @ 40MS/s @ f_{in} = 10MHz @-1dBFS (bits) ENOB @ 80MS/s @ f_{in} = 10MHz @-1dBFS (bits)	10.4 10.1	11.2 10.8
Maximum calibration time @80MS/s	27 seconds	

Conclusions

- A 13-bit 80 MS/s self-calibrated pipeline ADC employing a digitally-assisted histogram-based self-calibration technique using on-chip thermal noise was described;

- The self-calibrated ADC occupies less than 0.9 mm^2 and consumes only 85 mW at 1.2 V supply (@ 80MS/s and after PD of GNG and PGA);

- The largest improvements corresponding to over 14.8 dB in SFDR (to 81 dB) and 13.8 dB in THD (to -79.7 dB), were obtained at 40 MS/s resulting in a peak Figure-of-Merit of 467 fJ/conv.-step, including reference buffers.

- The proposed algorithm has a simple digital implementation and and it does not require any modifications in the ADC under calibration.

Acknowledgements

We would like to thank Dr. Boris Glass (from ESTEC-ESA) for his helpful feedback during the design and tests of the ADC,

and colleagues Cormac O'Sullivan and Mark Barry at S3 for their help on setting up this presentation.

Thank You.