

3.6 A 6-to-600MS/s Fully Dynamic Ringamp Pipelined ADC with Asynchronous Event-Driven Clocking in 16nm

Benjamin Hershberg, Barend van Liempd, Nereo Markulic, Jorge Lagos, Ewout Martens, Davide Dermit, Jan Craninckx

imec, Leuven, Belgium

At the upper end of achievable ADC operating speeds, clocking becomes a critical performance limiter. In "deep" pipelined ADCs that contain many stages, the clock tree constitutes a highly distributed network, with parasitics and mismatch creating skew between the different branches. Sufficient margin must be included in the timing generation such that all non-overlap and causal relationships are maintained. This leads to a difficult set of design tradeoffs in terms of power, speed, jitter, and reliability. Meanwhile, although residue amplifiers have traditionally dominated the power budget in deep pipelines, recent advances such as ring amplification have improved achievable efficiencies to the point that clocking is now the primary consumer in some cases [1,2].

It is already well known that asynchronous event-driven clocking techniques can provide speed and power advantages in SAR and pipelined-SAR ADCs. However, in deep pipelines this approach remains largely unexplored at the system level. In this paper, we propose a clocking paradigm where the conventional hierarchical clock tree is replaced by localized control units that interact with each other using event-driven triggers and handshake protocols. The result is a modular, correct-by-construction timing system that minimizes routing complexity and maximizes performance. The numerous advantages of the approach are combined with the efficiency of ring amplifiers to build a single-channel 11b, 60dB SNDR, 78dB SFDR pipelined ADC with fully dynamic power consumption that maintains better than 13fJ/conv-step Walden FoM from 6MS/s to 600MS/s.

Illustrated in Fig. 3.6.1, the single-channel ADC consists of seven 1.5b stages followed by a 1.5+3b backend. Each stage in the pipeline communicates with its direct neighbors through an event-driven protocol. This can be seen in the lower portion of Fig. 3.6.1, where a digital control unit in each stage sends and receives commands. The wires linking these control units together serve a dual-purpose: they provide both inter-stage communication and local line driving of circuits. This is more power and area efficient than hierarchical clocking where global distribution and local buffering are handled separately.

Figure 3.6.2 shows a simplified version of the event-driven timing that occurs during normal operation. It begins in the system reset state, where all stages are waiting to sample. The arrival of the *master clock* initiates a chain reaction. First, stage 1 samples the input and begins quantizing. When quantization is complete and stage 2 also indicates that it is ready, stage 1 amplifies the residue. After internal timing controls determine the residue is settled, *sample REQ* is sent to stage 2. Stage 2 then samples and returns an acknowledgement, allowing stage 1 to power down its ringamp and return to the *track* state. The inherent causality of this approach makes it correct-by-construction; safe and valid phase relationships are guaranteed.

Stage 2 now begins a similar exchange with stage 3, with a few notable differences. First, all stages except stage 1 use an early quantization technique that occurs in parallel with sampling. This removes the quantization phase from the critical timing path (explained later). Second, when stage 2 is ready to resume tracking, it will wait until the *begin* signal of stage 1 signal indicates that it is safe to reconnect. This prevents stage 2 from reconnecting to the output of stage 1 just as stage 1 samples, and avoids kickback related errors.

As illustrated in Fig. 3.6.1, the digital outputs of each stage propagate to the end of the pipeline for alignment and then shift back to stage 1 using the causally-related edges of *sample REQ*, where they can be re-synchronized with the master clock.

Event triggers can be generated with delay cells, derived directly from analog events, or a combination of both. For example, the sub-ADC comparators can self-detect completion, but also use a timeout delay to protect against metastability. Another example, as we introduce in Fig. 3.6.3, is a technique for sensing the end of a ringamp's slewing operation. Consider the binary value of the voltages at nodes A, B, C, and D. During slewing, the logic equation $AB'CD'$

will never equal '1', because $A=B$ and $C=D$. During settling, the dynamic formation of a dead-zone – in this case due to the CMOS resistor [2] – forces the four nodes apart such that $A=C='1'$ and $B=D='0'$ and the logic equation evaluates to '1'. Based on this principle, the simple detection circuit in Fig. 3.6.3 generates the *sub-ADC latch* signal of Figs. 3.6.1 and 3.6.2. This is used to implement early quantization [3], where the sub-ADC samples the residue of the previous stage early and quantizes it in parallel with continued residue settling. Our approach provides an optimal solution to the classic challenge of this technique, namely, when to sample. Sampling too soon may result in too much quantization error, and sampling too late will impede optimal settling due to disturbance of the amplifier output.

Triggering on analog events requires that the analog blocks themselves function correctly. This is not always the case during startup, when e.g. common-mode or differential saturation of the ringamp may occur. In such cases, slewing will never finish and the *sub-ADC latch* signal will never assert. The result is a deadlock. Fortunately, this can easily be detected (Fig. 3.6.2). Any deadlock will eventually propagate back to stage 1, and if a *master clock* edge arrives at stage 1 before it has begun sampling the input, we know that either: 1) the clock speed is too high for the given settings or 2) a deadlock has occurred. When this condition is detected, a "flush" command is issued that forces each stage into its *track* state. Analog blocks such as ringamp CMFB are designed so that repeatedly returning to this reset condition will result in correct bias convergence.

The internal processing speed of the pipeline is completely independent of the input clock. This has several important advantages. First, it automatically maximizes track time. Second, many critical leakage constraints are eliminated, significantly increasing the range of clock speeds that can be supported. For example, the MDAC switches can be sized for the maximum speed without concern of leakage corrupting the charge sampled onto the MDAC capacitors at low clock speeds. Third, reconfiguring the internal operation – even inserting or removing entire phases – is both trivial to implement and self-optimized for timing efficiency. Fourth, it eliminates the subtle but crucial tradeoff between input track time and sampling jitter in conventional clock trees that is particularly relevant at high speeds.

The ADC is fabricated in a 16nm technology, occupying an active area of 0.037mm² (110μm × 340μm), excluding decoupling (Fig. 3.6.7). It operates on a 0.85V supply with 50mV/800mV references. At 600MS/s with a Nyquist input it consumes 6.0mW (including clock buffer) and achieves 60.2dB SNDR, 78.3dB SFDR, 12.0fJ/conv-step FoM_w, and 167.2dB FoM_s. Measured performance is summarized in Figs. 3.6.4 and 3.6.5. All results use the same digital settings and stage gain coefficients (calculated off-chip). The fully dynamic, linear scaling of power with respect to clock frequency demonstrated in Fig. 3.6.4 makes this architecture an excellent candidate for reconfigurable applications such as multi-standard wireless. The supply sweep of Fig. 3.6.4 demonstrates the robustness of the ringamps to variation (to sweep down to 0.8V without saturating, the references are set to 50mV and 750mV for this test.)

A comparison with relevant state-of-the-art is provided in Fig. 3.6.6. This design achieves the best Walden FoM of any ADC above 410MS/s in [4], indicating that ringamp-based single-channel pipelined ADCs can offer a power-efficient alternative to multi-channel SAR architectures, without the overhead of interleave calibration. Furthermore, the versatility of event-driven control creates new possibilities for developing more efficient, reconfigurable, and robust solutions to classic challenges in pipelined ADCs.

References:

- [1] J. Lagos, et. al., "A 1Gsps, 12-bit, single-channel pipelined ADC with dead-zone-degenerated ring amplifiers," *IEEE CICC*, May 2018. (power breakdown by private communication)
- [2] J. Lagos, et. al., "A single-channel, 600MSps, 12bit, ringamp-based pipelined ADC in 28nm CMOS," *IEEE Symp. VLSI Circuits*, pp. 96-97, June 2017.
- [3] B.-M. Min, et. al., "A 69-mW 10-bit 80-MSample/s Pipelined CMOS ADC," *IEEE JSSC*, vol. 38, no. 12, pp. 2031-2039, Dec. 2003.
- [4] B. Murmann, "ADC Performance Survey 1997-2018," [Online]. Available: <http://web.stanford.edu/~murmman/adcsurvey.html>.

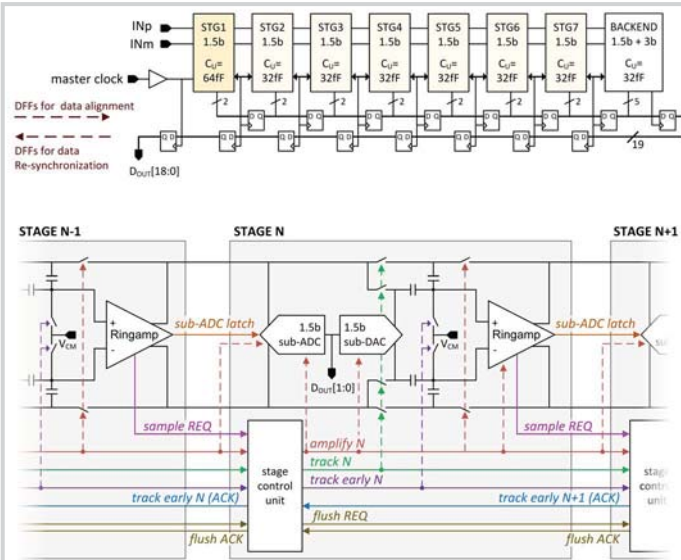


Figure 3.6.1: ADC top level and simplified MDAC with event-driven control.

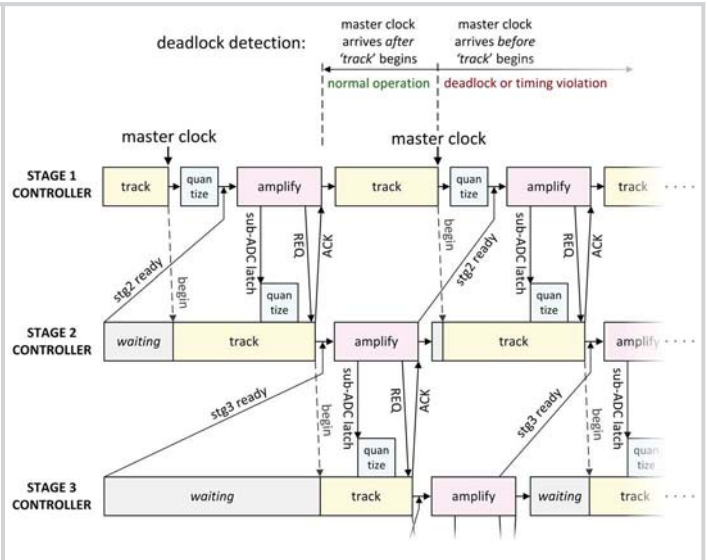


Figure 3.6.2: Timing diagram with simplified event-driven communication.

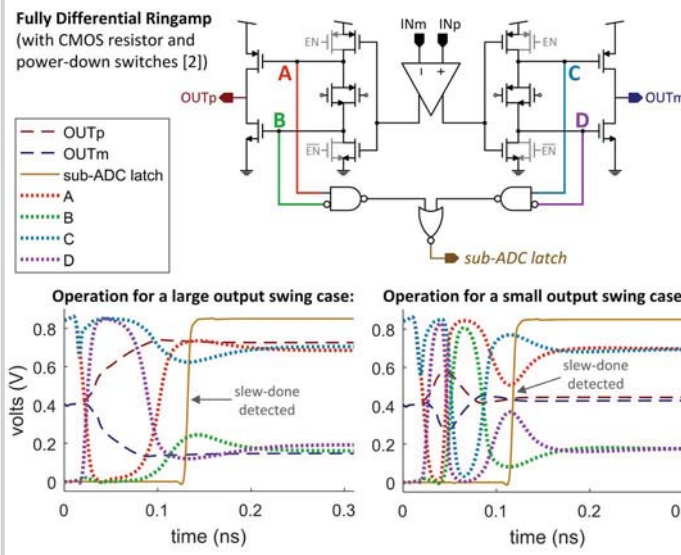


Figure 3.6.3: Simplified ringamp and two examples of slew-done detection.

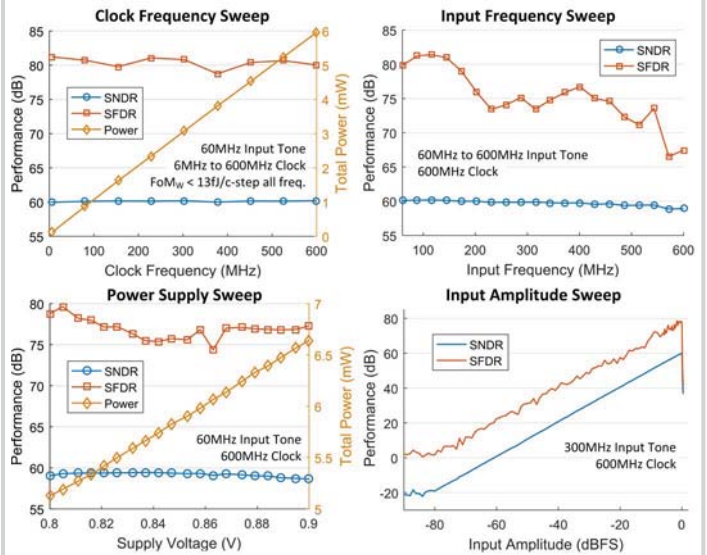


Figure 3.6.4: Characterization sweeps.

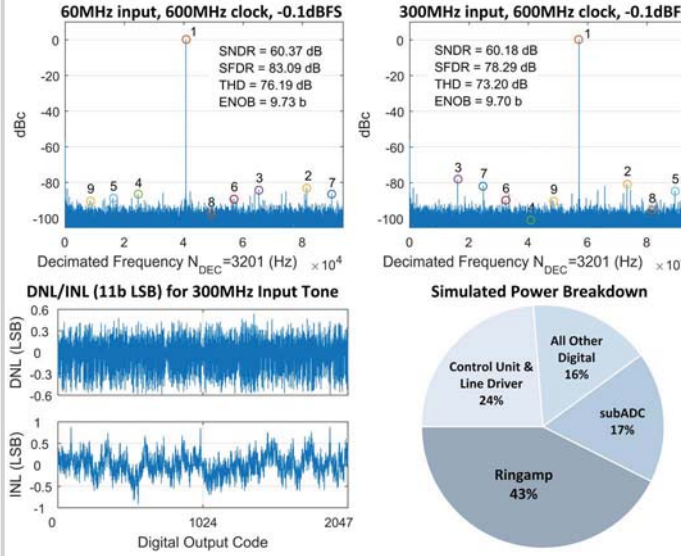


Figure 3.6.5: Measured dynamic and static performance.

Architecture	High Speed Ringamp SoTA			SoTA in [1] for SNDR > 56dB, Fs > 410MHz			
	This work	Lagos CICC 2018	Lagos VLSI 2017	Yenca ISSCC 2016	Nam VLSI 2016	Lien VLSI 2016	Sung ISSCC 2015
Pipeline	Pipeline	Pipeline	Pipeline	TI SAR ΔΣ	TI SAR ΔΣ	TI Pipe-SAR	TI FATI-SAR
Sampling rate (max) [MS/s]	600	1000	600	600	1600	800	1600
Number of channels	1	1	1	4	8	4	12
Per-channel rate [MS/s]	600	1000	600	150	200	200	133
Residue Amplifier	Ringamp	Ringamp	Ringamp	-	-	Opamp	-
Resolution [bit]	11	12	12	12	12	12	10
Technology [nm]	16	28	28	28	65	28	65
Supplies [V]	0.85	0.9	0.9	1.2, 1.5	1.1, 1.2	1	1.1
Input range [Vpp]	1.5	1.6	1.6	1.5	1.6	1.2	n.a.
ENOB LF input [bit]	9.7	9.2	9.4	9.8	11.6	10.0	9.2
ENOB Nyquist [bit]	9.7	9.1	9.1	9.3	10.6	9.8	9.0
SNDR LF input [dB]	60.4	57.1	58.1	60.7	71.8	62.2	57.2
SNDR Nyquist [dB]	60.2	56.6	56.3	58.0	65.3	60.8	56.1
SFDR LF input [dB]	83.1	74.6	67.5	77.5*	83.0	79.0*	68.0
SFDR Nyquist [dB]	78.3	73.1	69.2	66.0*	65.3	75.0	61.2
Power [mW]	6.0	24.8	14.2	26.5	37.7	14.6	17.3
FoM _{total} [9/c.step]	12.0	45.0	44.3	68.0	17.8	20.0	21.0
FoM _{Schreier} [dB]	167.2	159.6	159.5	158.5	168.6	165.2	162.8
Core area [mm ²]	0.037	0.540	0.620	0.042	0.900	0.093	0.360
Calibration	gain	gain	gain	offset, gain, mismatch	offset, gain, timing skew	offset, gain	offset

Figure 3.6.6: Comparison with relevant state-of-the-art.

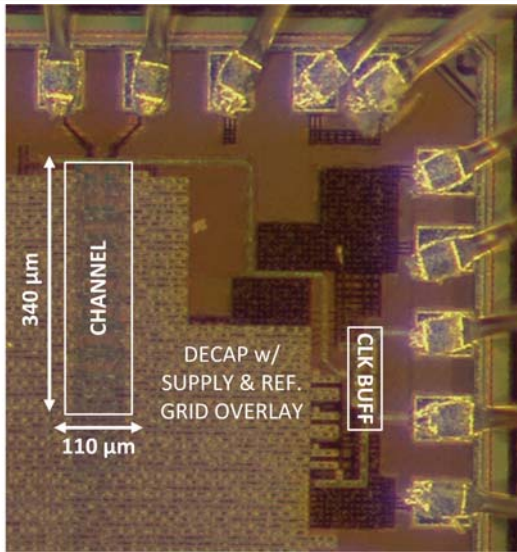


Figure 3.6.7: Chip Micrograph.