Pipeline ADC PN-Dither LMS Background Calibration: FPGA and ASIC Digital Back-End Implementation

XINGYU WANG
MASTER´S THESIS
DEPARTMENT OF ELECTRICAL AND INFORMATION TECHNOLOGY
FACULTY OF ENGINEERING | LTH | LUND UNIVERSITY

Pipeline ADC PN-Dither LMS Background Calibration: FPGA and ASIC Digital Back-End Implementation

Xingyu Wang

Department of Electrical and Information Technology
Faculty of Engineering, LTH, Lund University

Supervisor: Baktash Behmanesh and Wenbo Li

Examiner: Pietro Andreani

June 12, 2026

Printed in Sweden
Tryckeriet i E-huset, Lund

Abstract

This thesis presents a complete digital back-end implementation of a PN-dither least-mean-squares (LMS) background calibration scheme for a 12-bit, 500 MS/s pipeline analog-to-digital converter (ADC). The target converter, designed by a parallel analog project, comprises four 3-bit stages and a 4-bit final flash stage; its actual inter-stage gains depart substantially from their nominal values, reducing the effective number of bits (ENOB) of the uncalibrated output to 4.19 bits. The back-end injects a small pseudo-noise dither at the residue path of each calibrated stage and uses a one-tap LMS update to track the gain errors in real time. A division-free reconstruction datapath, obtained by algebraic rearrangement of the standard reconstruction equation, recovers the calibrated output using only multiplications and additions. A parametric study of the gain-accumulator bit width identifies a *stochastic-quantisation regime* that informs the choice of a 24-bit accumulator (S3.20 format), trading 0.26 bits of ENOB for a 53 % reduction in look-up table count relative to a full-precision reference.

The same Verilog source has been carried through both an FPGA verification flow and an ASIC physical-implementation flow. On a Xilinx Artix-7 XC7A100T the design closes timing at 58.8 MHz with 721 LUTs, 577 flip-flops, 11 DSP48 blocks, and 113 mW of total power. On a 65 nm CMOS process the placed-and-routed layout occupies 0.031 mm^2 and dissipates 2.07 mW under default activity—a factor of 55× lower than the FPGA target. The achieved steady-state ENOB of 7.20 bits exceeds the project objective of 7 bits and demonstrates that a heavily resource-optimised digital calibration back-end can match the spectral performance of a full-precision baseline in production-silicon form.

Popular Science Summary

Modern wireless networks, medical imaging devices, and self-driving cars all rely on a small but critical electronic component called an analog-to-digital converter (ADC), which turns continuously varying signals into the streams of numbers that a computer can process. Like a ruler whose markings are slightly bent, real ADCs are never perfect: imperfections in the analog circuitry, drift with temperature, and aging cause some bits of the output to be wrong. The converter studied in this thesis has a 12-bit hardware design, but with the uncorrected analog errors it delivers an effective resolution of only about four bits.

This thesis builds a digital self-correction circuit that sits alongside the analog ADC and quietly fixes its imperfections while the converter is running. A tiny known random nudge is added into the analog signal; by watching where that nudge ends up in the digital output, the circuit deduces how large each analog error is and updates a small set of correction numbers in the background—using a long-established adaptive-filter algorithm that also powers noise-cancelling headphones. The same hardware description has been implemented both on a reconfigurable prototyping chip (FPGA) and as a custom integrated circuit in a mainstream 65 nm silicon process. The custom-silicon version occupies an area about the size of a sand grain, dissipates only two milliwatts of power—more than fifty times less than the FPGA prototype—and recovers more than seven bits of effective resolution from a converter that, uncorrected, manages only four.

List of Abbreviations

Abbreviation	Meaning
ADC	Analog-to-Digital Converter
ASIC	Application-Specific Integrated Circuit
CDC	Clock Domain Crossing
CMOS	Complementary Metal-Oxide-Semiconductor
CORDIC	COordinate Rotation DIgital Computer
CSA	Carry-Save Adder
CTS	Clock Tree Synthesis
DFF	D-type Flip-Flop
DRC	Design Rule Check
DSP	Digital Signal Processing (block)
ECO	Engineering Change Order
ENOB	Effective Number Of Bits
FF	Flip-Flop
FoM	Figure of Merit
FPGA	Field-Programmable Gate Array
FSM	Finite-State Machine
GDS	Graphic Database System
IC	Integrated Circuit
I/O	Input/Output
IOB	Input/Output Block (Xilinx)
LEF	Library Exchange Format
LFSR	Linear-Feedback Shift Register
LMS	Least Mean Squares
LP	Low Power (process variant)
LSB	Least Significant Bit
LUT	Look-Up Table
LVS	Layout-Versus-Schematic
MMMC	Multi-Mode Multi-Corner
PDK	Process Design Kit
PnR	Place-and-Route
PN	Pseudo-Noise (sequence)

Abbreviation	Meaning
PT	PrimeTime (static timing analysis tool)
PVT	Process / Voltage / Temperature
RC	Resistance-Capacitance (extraction)
RTL	Register-Transfer Level
SDC	Synopsys Design Constraints
SDF	Standard Delay Format
SFDR	Spurious-Free Dynamic Range
SNDR	Signal-to-Noise-and-Distortion Ratio
SPEF	Standard Parasitic Exchange Format
STA	Static Timing Analysis
TCL	Tool Command Language
VCD	Value Change Dump
WNS	Worst Negative Slack

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Modern signal-processing systems rely on analog-to-digital converters (ADCs) at every interface between the continuous physical world and the digital domain. The performance of the ADC frequently determines the achievable resolution, speed, and energy efficiency of the entire system. In wireless communications, the migration from 4G LTE to 5G NR and the anticipated 6G standards has pushed channel bandwidths from tens of megahertz into the gigahertz range, requiring sample rates of hundreds of megasamples per second at resolutions of 10–12 bits to support higher-order modulation schemes [**?**]. In automotive radar, light detection and ranging (LiDAR), and medical imaging, ADCs with comparable dynamic range are equally indispensable [**?**].

Among the architectural families, the pipeline ADC has long held a privileged position in the moderate-to-high resolution, medium-speed segment of roughly 10–16 bits at 10 MS/s to 500 MS/s [**?**, **?**]. By distributing the conversion across N successive stages, each containing a coarse sub-ADC, a multiplying digital-to-analog converter (MDAC), and a residue amplifier, the pipeline architecture achieves throughput-area scaling that is favourable to nanometre CMOS technologies, and accepts the introduction of digital redundancy at each stage as a low-cost way to absorb comparator offsets.

Two long-running technology trends have nevertheless eroded the analog performance margin traditionally available to pipeline designers. Supply-voltage scaling restricts the available signal swing, forcing residue amplifiers to operate with lower open-loop gain than was customary in older nodes [**?**]. In parallel, demand for higher-resolution converters in cost-sensitive applications drives capacitor-matching requirements beyond what passive layout matching can guarantee. Surveys of state-of-the-art designs document a corresponding shift in the prevailing figure of merit: ADC entries near the front of the FoM envelope today almost without exception rely on some form of digital assistance to compensate for analog non-idealities at run time [**?**, **?**].

This thesis treats one component of a complete digitally-assisted pipeline ADC. The companion analog front-end—a 12-bit 500 MS/s five-stage pipeline ADC—was designed in a separate effort; the work reported here is the digital calibration back-end that converts the raw per-stage codes from the analog front-end into a

calibrated 12-bit output, compensates for inter-stage gain errors in real time, and is realised in two complementary implementation forms: an FPGA verification target and a standard-cell ASIC layout suitable for system-on-chip integration.

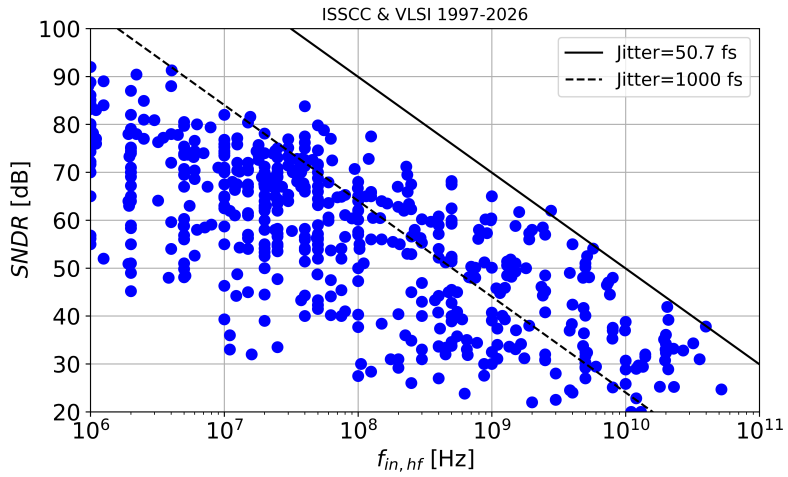


Figure 1.1: State-of-the-art ADC figure-of-merit landscape, redrawn from Murmann's annual ADC survey [**?**]; pipeline converters with digital calibration dominate the high-FoM frontier.

1.2 Why Digital Calibration?

Real-world pipeline ADCs deviate from the ideal in several reproducible ways. The residue amplifier at each stage achieves only a finite open-loop gain, causing the inter-stage gain factor G_k to fall below its nominal value. The capacitor array (CDAC) within the MDAC exhibits matching errors at the level of 0.1 % to 1 % even after careful layout, which translate directly into stage gain errors. Comparator offsets in the flash sub-ADCs are largely absorbed by the 0.5 bit redundancy at each stage, but the residual offset still consumes a portion of the signal headroom. All of these errors drift with temperature and supply voltage.

Transistor-level simulation of the target analog front-end shows the magnitude of the problem. With the nominal inter-stage gains of $(4, 4, 4, 4)$ replaced by the actual values (3.5, 2.8, 2.3, 2.2) extracted from analog simulation, the uncalibrated converter retains an effective resolution of only about four bits—far short of its 12-bit code width. Its output spectrum is dominated by gain-error harmonics sitting on an elevated noise floor; the before- and after-calibration spectra are examined quantitatively in Chapter 6. A digital back-end is therefore essential to the converter's usability.

Calibration techniques fall conventionally into two families. *Foreground* calibration interrupts normal conversion, injects a known test signal, computes the correction off-line, and stores the result. Although simple, foreground schemes are

incompatible with continuous-duty applications and cannot track drift after the initial calibration. *Background* calibration estimates the analog parameters continuously while the converter remains in service, and is therefore preferred whenever the converter must operate without interruption.

Among the background techniques in the literature, pseudo-noise (PN) dither combined with a least-mean-squares (LMS) adaptive filter offers an especially clean solution [**?**, **?**]. A small pseudo-random binary perturbation is injected at the residue amplifier input of each stage prior to quantisation and removed digitally in the reconstruction chain. Because the dither sequence is statistically orthogonal both to the input signal and to the dithers of other stages, cross-correlating the reconstructed output with the dither yields an unbiased estimate of the stage's gain error. The LMS update rule converts that estimate into a one-tap gradient descent on a gain register, requiring per sample only a multiplication, a subtraction, and an accumulation. The arithmetic burden falls entirely on the digital back-end; the analog front-end is augmented only by a single 1-bit dither switch per calibrated stage.

PN-dither LMS calibration is therefore well matched to the present design objective: a digital back-end that is small enough to share silicon area with the analog front-end, generic enough to be reused across pipeline ADCs with similar architecture, and continuous enough to track process and temperature drift over the converter's service life.

1.3 Problem Statement and Scope

The target converter is a five-stage pipeline ADC composed of four 3-bit stages followed by a final 4-bit flash stage, designed for a nominal sample rate of 500 MS/s and a 12-bit output. With each 3-bit stage providing one bit of redundancy through code overlap and the final stage producing four bits without redundancy, the unprocessed pipeline yields a 12-bit raw word; the digital back-end emits a 12-bit calibrated output. Inter-stage gain errors of the magnitude reported in Section 1.2 reduce ENOB to roughly four bits without calibration; the design objective for the back-end is to recover at least 7 bits of ENOB through PN-dither LMS calibration.

This thesis addresses three implementation phases:

- **Phase 1 — System modelling.** A bit-true MATLAB model of the calibration loop is developed. Parameter sweeps establish the dither amplitude, loop gain, and fixed-point bit widths, and a floating-point reference establishes the upper bound on achievable ENOB (7.7 bits).
- **Phase 2 — RTL implementation and FPGA verification.** The fixed-point model is translated to synthesisable Verilog, integrated with a self-checking testbench, and closed at 58.8 MHz on a Xilinx Artix-7 XC7A100T target.
- **Phase 3 — ASIC digital back-end.** The verified RTL is taken through Cadence Genus synthesis, Cadence Innovus place-and-route, and post-layout static timing analysis in a 65 nm CMOS standard-cell library, producing a placed-and-routed layout of 0.031 mm^2.

The scope is explicitly bounded as follows. The analog front-end—residue amplifiers, CDACs, comparators, sample-and-hold network, and reference buffers—is handled by a separate analog design effort and is treated here as a black box characterised by its measured stage gains. Tape-out, silicon validation, and field testing of the digital back-end lie outside the present scope; the work terminates at post-layout static timing sign-off and gate-level functional simulation against the FPGA reference.

1.4 Thesis Outline

Chapter 2 establishes the theoretical foundation: the pipeline ADC operating principle, the spectral signature of stage gain error, the role of PN dither injection, the LMS update rule, and the algebraic equivalence between reciprocal and forward-multiplication reconstruction. Chapter 3 presents the calibration architecture in detail and develops the rationale behind each design choice—loop gain, fixed-point bit widths, the decision to omit post smoothing of the gain estimate, and the structure of the reconstruction multiplier. Chapter 4 reports the FPGA implementation and the timing-closure narrative on the Xilinx Artix-7 target. Chapter 5 carries the verified RTL through the ASIC digital back-end flow on the 65 nm CMOS target, comprising synthesis, place-and-route, post-layout static timing analysis, and gate-level simulation. Chapter 6 consolidates the simulation results from both implementation platforms, including the bit-width Pareto study and a post-layout-versus-RTL spectral comparison. Chapter 7 summarises the contributions and outlines directions for future work.

Chapter 2

Pipeline ADC and PN-Dither LMS Calibration

2.1 Pipeline ADC Operation

The pipeline ADC realises an N-bit conversion by cascading several resolution-limited stages along a common signal path. Each intermediate stage quantises the signal coarsely into a b_k-bit code, regenerates the corresponding analog level, and amplifies the residue—the difference between the actual input and the regenerated value—for use by the next stage. The final stage is a single flash quantiser without residue amplification. Codes from all stages are aligned in time and combined in the digital domain to produce the final N-bit output.

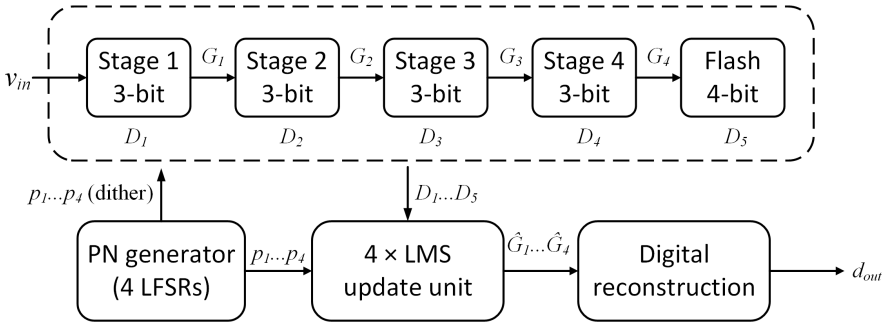


Figure 2.1: System overview of the digitally-assisted pipeline ADC, with the analog front-end (left of the dashed boundary) treated as a black box and the digital back-end (right) as the subject of this thesis.

The converter studied in this thesis comprises five stages: four 3-bit stages followed by a final 4-bit flash, yielding a 12-bit raw word. Each 3-bit stage quantises the signal into $2^3 = 8$ levels but uses only $2^3 - 1 = 7$ output codes; the redundant code provides 0.5 bit of margin that absorbs comparator offset and prevents the residue amplifier from saturating at sub-range boundaries.

Within stage k, the input voltage $v_{\mathrm{in},k}$ is sampled and passed in parallel to the

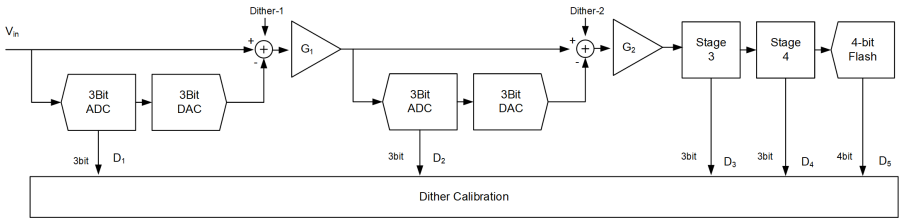


Figure 2.2: Internal datapath of one 3-bit pipeline stage: the sampled input is quantised by the flash sub-ADC, reconstructed and subtracted by the DAC, and the residue is amplified by the stage gain G_k.

flash sub-ADC and the residue path. The flash decision $D_k \in \{0, 1, \ldots, 2^{b_k} - 1\}$ indexes a reference level $D_k \cdot V_{\mathrm{ref}}/2^{b_k-1}$, which the internal DAC subtracts from the sampled signal; the residue amplifier then multiplies the difference by the stage gain G_k:

$$v_{\mathrm{res},k} \;=\; G_k\!\left(v_{\mathrm{in},k} \;-\; D_k \cdot \frac{V_{\mathrm{ref}}}{2^{b_k-1}} \right). \tag{2.1}$$

In a nominally designed stage, $G_k = 2^{b_k-1}$ so that the residue spans the same full-scale range as the original input. For the target converter, the nominal 3-bit stage gain is therefore $G_k = 4$, and the final 4-bit flash stage carries no residue amplifier at all.

The complete digital output is reconstructed by accumulating the contributions of all stages with appropriate scaling. With $N_s = 5$ total stages and inter-stage gains G_1, G_2, G_3, G_4, the reconstructed sample takes the form

$$D_{\mathrm{out}} \;=\; D_1 \;+\; \frac{D_2}{G_1} \;+\; \frac{D_3}{G_1 G_2} \;+\; \frac{D_4}{G_1 G_2 G_3} \;+\; \frac{D_5}{G_1 G_2 G_3 G_4}, \tag{2.2}$$

where each D_k is interpreted as a signed quantity centred on its mid-range value. Successive terms contribute smaller fractions of the output, mirroring the descending significance of bits in a binary word.

The 4-bit final stage employs $2^4 = 16$ comparators and produces a 4-bit code directly, without residue amplification. Compared with extending the chain by another 3-bit stage, the larger fan-out of the final flash yields a finer quantisation step at the bottom of the pipeline and—as analysed in Section 2.7—halves the geometric bias seen by the LMS calibration loop.

2.2 Stage Gain Error and Its Spectral Signature

The reconstruction in Equation (2.2) assumes that each G_k takes exactly its nominal value. In practice this assumption is violated by three reproducible mechanisms:

- **Finite open-loop gain** of the residue amplifier. A closed-loop residue amplifier built around an op-amp of finite open-loop gain A_0 realises slightly

less than its nominal gain, quantified in Equations (2.3)–(2.5) below. Heavy supply-voltage scaling in nanometre CMOS makes high open-loop gain progressively harder to achieve [**?**].

- **Capacitor matching error** in the multiplying DAC. The flip-around MDAC realises the gain through a ratio of sampling to feedback capacitors; mismatch between the unit capacitors perturbs this ratio at the level of 0.1 % to 1 % even after careful common-centroid layout.
- **Drift** of both effects with temperature and supply voltage, which means even a foreground-calibrated converter cannot retain its initial accuracy in operation.

The first of these mechanisms can be made quantitative. A switched-capacitor residue amplifier with feedback factor β_k, closed around an op-amp of open-loop gain A_0, has the standard finite-gain closed-loop response

$$G_k^{\text{actual}} \;=\; \frac{A_0}{1 + A_0\,\beta_k}, \tag{2.3}$$

which approaches the ideal value $G_k^{\text{nom}} = 1/\beta_k$—set by the sampling-to-feedback capacitor ratio of the MDAC—as $A_0 \to \infty$. Substituting $\beta_k = 1/G_k^{\text{nom}}$ and dividing the numerator and denominator of Equation (2.3) by $A_0\beta_k$ expresses the realised gain as a fraction of its nominal value,

$$G_k^{\text{actual}} \;=\; \frac{G_k^{\text{nom}}}{1 + \dfrac{1}{A_0\,\beta_k}}. \tag{2.4}$$

For a well-designed stage the loop gain satisfies $A_0\beta_k \gg 1$, so a first-order expansion $1/(1+x) \approx 1 - x$ of Equation (2.4) gives

$$G_k^{\text{actual}} \;\approx\; G_k^{\text{nom}}\left(1 - \frac{1}{A_0\beta_k}\right) \;=\; G_k^{\text{nom}}\left(1 - \frac{G_k^{\text{nom}}}{A_0}\right), \tag{2.5}$$

where the last equality again uses the flip-around-MDAC relation $\beta_k \approx 1/G_k^{\text{nom}}$. The realised gain therefore falls short of nominal by a fractional amount $1/(A_0\beta_k) = G_k^{\text{nom}}/A_0$ that grows with the nominal stage gain and shrinks with the available open-loop gain.

Collecting this finite-gain term together with the capacitor-matching and drift contributions into a single fractional error ε_k, the combined effect on stage k is written

$$G_k^{\text{actual}} \;=\; G_k^{\text{nom}}\,(1 + \varepsilon_k), \tag{2.6}$$

where ε_k may amount to several tens of percent in a practical pipeline implementation; for the finite-gain contribution alone, $\varepsilon_k \approx -1/(A_0\beta_k)$. For the target analog front-end, transistor-level simulation yields actual stage gains of (3.5, 2.8, 2.3, 2.2) against the nominal value of 4.0, i.e. fractional errors ε_k of -12.5%, -30.0%, -42.5%, and -45.0% across the four stages. The error increases with stage index because the same passive matching budget covers a progressively smaller signal

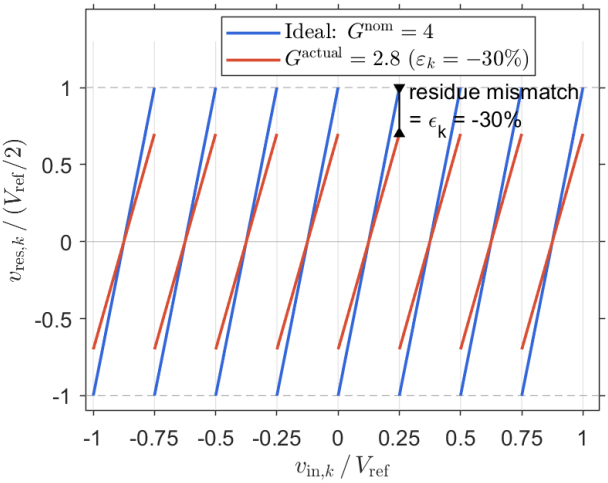


Figure 2.3: Residue transfer of one 3-bit pipeline stage for the ideal case ($G^{\text{nom}} = 4$) and for stage 2's actual gain ($G^{\text{actual}} = 2.8$, $\varepsilon = -30\%$), showing the "dead band" at every sub-range boundary.

swing, and because the per-stage amplifier follows the same open-loop gain target throughout the chain.

Two consequences follow from Equation (2.6). First, when $G_k^{\text{actual}} < G_k^{\text{nom}}$ the residue at the sub-range boundary $v_{\text{in},k} = D_k \cdot V_{\text{ref}}/2^{b_k-1}$ falls below the full-scale input expected by the next stage. The downstream stage observes a compressed input range, and its codes therefore under-represent the true residue. Across the cascade, this compression accumulates: the deeper the stage, the larger the relative contribution to the final-output error.

Second, applying the ideal reconstruction formula (2.2) with mismatched gains introduces a piecewise discontinuity in the mapping from analog input to reconstructed code. The origin of this discontinuity is visible in the per-stage residue transfer of Figure 2.3: because the shallow-slope segments of the actual-gain curve fall short of $\pm V_{\text{ref}}$, each sub-range boundary leaves a "dead band" that the downstream stage cannot resolve. When the stage codes are recombined through (2.2), these dead bands surface as a jump in the reconstructed output at every boundary, with a magnitude proportional to ε_k. For a sinusoidal test signal, the jumps act as a piecewise-constant additive distortion whose Fourier expansion contains harmonics at integer multiples of the input frequency. The spectrum of the uncalibrated output therefore exhibits three characteristic features:

- **Harmonic spurs** at $2f_{\text{in}}$, $3f_{\text{in}}, \ldots$, located either inside or outside the converter's Nyquist band depending on the input frequency.
- **Amplitude-modulated sidebands** surrounding each harmonic when the gain errors of different stages are unequal: the modulation pattern reflects the nonlinear coupling between the stages.

- **An elevated in-band noise floor**, since the discontinuities effectively quantise the signal more coarsely than the 12-bit code word would suggest.

Quantitatively, the simulated effect on the target converter is dramatic. With the gain errors stated above, the uncalibrated converter delivers an ENOB of only 4.19 bit—roughly an eight-bit loss relative to the 12-bit raw code word. Recovering the lost resolution is the central task of the digital back-end, and motivates the PN-dither LMS calibration scheme developed in the remainder of this chapter.

2.3 PN-Dither Injection

The starting point of the PN-dither calibration scheme is to inject a small, statistically known perturbation into the residue path of each 3-bit stage and to remove its known component from the reconstructed output in the digital domain. The residue distortion caused by the unknown gain error is then partially superimposed on the injected perturbation, and standard adaptive-filter techniques (Section 2.4) can recover the gain error from a long-term correlation between the residual signal and the injected sequence.

The perturbation chosen here is a binary pseudo-noise (PN) sequence, generated digitally and applied to the residue path through a single 1-bit polarity switch per stage. Formally, for stage k the injected analog dither at sample n is

$$\Delta_k[n] \;=\; V_{\mathrm{pn}}\, p_k[n], \qquad p_k[n] \in \{-1,\,+1\}, \tag{2.7}$$

where V_{pn} is the (fixed) dither amplitude and $p_k[n]$ the PN sequence supplied by a linear-feedback shift register (LFSR) in the digital back-end. Four independent LFSRs are used to drive the four calibrated 3-bit stages; the final 4-bit flash stage is not dithered since it has no residue amplifier to calibrate. The only addition required of the analog front-end is a single switched capacitor (or equivalent steering element) per calibrated stage, controlled by the corresponding $p_k[n]$ bit.

Two statistical properties of $p_k[n]$ make the scheme work. First, because the PN sequence has zero mean and is uncorrelated with the analog input signal,

$$\mathbb{E}[\, p_k[n]\, v_{\mathrm{in}}[n]\,] \;=\; 0, \tag{2.8}$$

so the dither contributes no DC bias and adds no in-band tone at any fixed frequency. Second, separate LFSRs with maximally-disjoint tap polynomials produce sequences that are mutually uncorrelated,

$$\mathbb{E}[\, p_j[n]\, p_k[n]\,] \;=\; \begin{cases} 1, & j = k, \\ 0, & j \neq k, \end{cases} \tag{2.9}$$

so that the gain estimate of one stage is not corrupted by the dither of another stage. Property (2.8) provides the *signal rejection* that makes calibration insensitive to input statistics, and property (2.9) permits four LMS loops to operate in parallel without cross-coupling.

The dither propagates through the stage in two ways. The flash sub-ADC of stage k samples $v_{\mathrm{in},k}[n]$ without seeing the dither; the DAC then regenerates

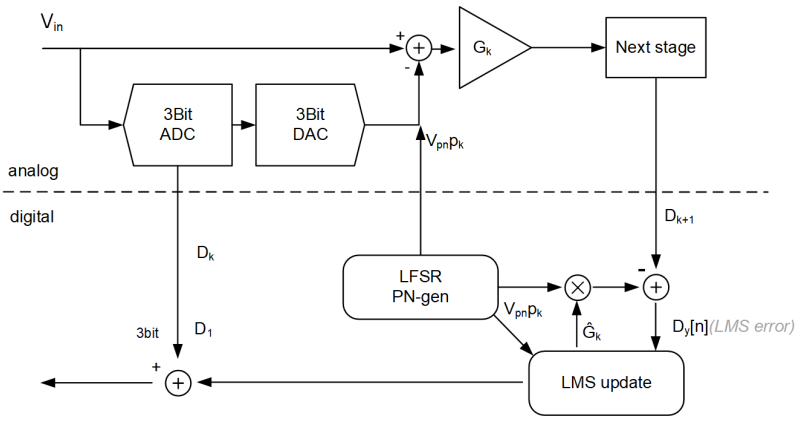


Figure 2.4: PN-dither injection at one 3-bit pipeline stage: the LFSR sequence $p_k[n]$ both drives the analog polarity switch and forms the digital LMS error signal $D_y[n]$.

the value $D_k\, V_{\mathrm{ref}}/2^{b_k-1}$ corresponding to the flash code. The polarity switch adds $\Delta_k[n]$ *after* the DAC subtraction node, so the residue passed to the amplifier is

$$v_{\mathrm{res},k}[n] \;=\; G_k\!\left(\, v_{\mathrm{in},k}[n] - D_k \cdot \frac{V_{\mathrm{ref}}}{2^{b_k-1}} + V_{\mathrm{pn}}\, p_k[n] \right). \tag{2.10}$$

The downstream stage code $D_{k+1}[n]$ therefore carries an amplified replica $G_k V_{\mathrm{pn}} p_k[n]$ of the injected dither, superimposed on the desired signal residue. Recovering the gain G_k amounts to extracting this replica from the noisy stream $D_{k+1}[n]$—which is the task delegated to the LMS update in the next section.

2.4 LMS Update Rule

To estimate the unknown gain G_k in real time, the digital back-end maintains a running estimate $\hat{G}_k[n]$ and refines it sample by sample with a one-tap least-mean-squares recursion. A distinctive feature of calibrating a pipeline ADC this way is that the only quantities available to the algorithm are *digital*: the stage codes D_k and D_{k+1} together with the known injected dither—never the analog inter-stage residue. The recursion is therefore driven by an error built purely from these codes, comparing the downstream code with its predicted value under the current estimate. Defining the digital dither twin

$$D_{d,k}[n] \;\triangleq\; V_{\mathrm{pn}}\, p_k[n], \tag{2.11}$$

the LMS error is

$$D_y[n] \;=\; D_{k+1}[n] \;-\; \hat{G}_k[n]\, D_{d,k}[n]. \tag{2.12}$$

To see why this works, write the downstream code as the amplified, dithered residue $D_{k+1} = G_k\,(r_k + D_{d,k})$, where r_k is the stage-k signal residue. The error

of Equation (2.12) then separates into two parts,

$$D_y[n] \;=\; \underbrace{G_k\, r_k[n]}_{\text{amplified residue}} \;+\; \underbrace{\big(G_k - \hat{G}_k\big)\, D_{d,k}[n]}_{\text{gain-error term}}. \tag{2.13}$$

When $\hat{G}_k = G_k$ the second term vanishes and D_y is simply the amplified residue; otherwise D_y keeps a component proportional both to the gain error and to the injected dither. Because the dither is the only part of D_y correlated with the pseudo-random sequence, multiplying D_y by the dither sign p_k and accumulating over many samples isolates that component: the residue r_k is uncorrelated with the sequence and averages to zero, whereas $p_k^2 = 1$, leaving

$$\mathbb{E}[\, p_k[n]\, D_y[n]\,] \;=\; \big(G_k - \hat{G}_k\big)\, V_{\text{pn}}, \tag{2.14}$$

which is non-zero whenever the estimate is wrong and vanishes exactly when $\hat{G}_k = G_k$. This accumulated correlation is therefore an unbiased measure of the gain error—independent of the input signal and of the other stages' estimates—and the loop simply nudges the estimate by a small amount in its direction. Folding the constant V_{pn} into the step-size parameter μ gives the LMS recursion [**?**]

$$\boxed{\hat{G}_k[n+1] \;=\; \hat{G}_k[n] \;+\; \mu\, p_k[n]\, D_y[n]} \tag{2.15}$$

where $\mu > 0$ is the loop-gain (step-size) parameter that controls the speed of adaptation. Equation (2.15) is implementable with one multiplication, one subtraction, and one accumulation per sample per stage; for the four 3-bit stages of the target converter this amounts to four identical update units operating in parallel.

Because the dither is present in the pipeline data from the instant it is injected, the digital reconstruction must also remove it: once $\hat{G}_k$ has converged, the downstream code is rescaled by the estimated gain and the known dither $D_{d,k}$ is subtracted before the residue is recombined with the upstream code D_k. Section 2.6 develops the division-free form of this reconstruction.

The step size μ sets a trade-off between convergence speed and steady-state accuracy: a larger μ adapts faster but leaves more residual jitter on $\hat{G}_k$, whereas a smaller μ suppresses that jitter at the cost of a longer convergence time. A sweep of μ in the floating-point reference model places the best compromise at $\mu_{\text{eff}} = 2^{-12.4} \approx 1.85 \times 10^{-4}$, where the calibrated output reaches ENOB = 7.7 bit—about three and a half bits above the uncalibrated 4.19 bit—after a convergence time of order 1.3×10^5 samples. This 7.7 bit value is the upper bound against which the fixed-point and hardware implementations of the following chapters are evaluated; Chapter 3 shows how the same μ_{eff} is realised as an arithmetic right-shift (Section 3.4).

A practical observation closes this section. Because $p_k[n] \in \{-1, +1\}$, the multiplication $p_k[n]\, D_y[n]$ in (2.15) is merely a sign inversion; and if μ is chosen to be a power of two, the further multiplication by μ reduces to an arithmetic right-shift. The entire LMS update then collapses to a sign-conditional shift and an accumulation, eliminating the need for a dedicated multiplier in the calibration core. Chapter 3 returns to this observation when selecting fixed-point bit widths and the implementation-side step-size μ_{eff}.

2.5 Dither Amplitude Selection

The dither amplitude V_{pn} in Equation (2.7) is a free parameter, bounded below by gradient quality and above by the analog injection headroom.

Lower bound on V_{pn}. The gradient (2.14) is proportional to V_{pn}, and the gradient channel's signal-to-noise ratio is $V_{\mathrm{pn}}^2/\sigma_n^2$, so the residual gain-error standard deviation $\sigma_{\hat{G}_k}$ scales as $1/V_{\mathrm{pn}}$. A small V_{pn} therefore forces either a longer convergence time or a wider steady-state distribution, both of which degrade the post-calibration ENOB.

Upper bound on V_{pn}. The dither is injected in the analog residue path, so its amplitude is ultimately limited by the available signal headroom. Near a sub-range boundary the residue already carries a small offset δ_k, and the dither adds $\pm V_{\mathrm{pn}}$ before amplification by the actual stage gain,

$$v_{\mathrm{res},k}[n] \;=\; G_k^{\mathrm{actual}}\big(\delta_k + V_{\mathrm{pn}}\, p_k[n]\big). \tag{2.16}$$

Once V_{pn} is large enough that this residue exceeds the downstream stage's full-scale range, the stage saturates, the orthogonality property (2.8) breaks, and the LMS gradient acquires a bias. The usable dither amplitude is therefore capped by how much perturbation the analog front-end can inject before the worst-case stage runs out of headroom.

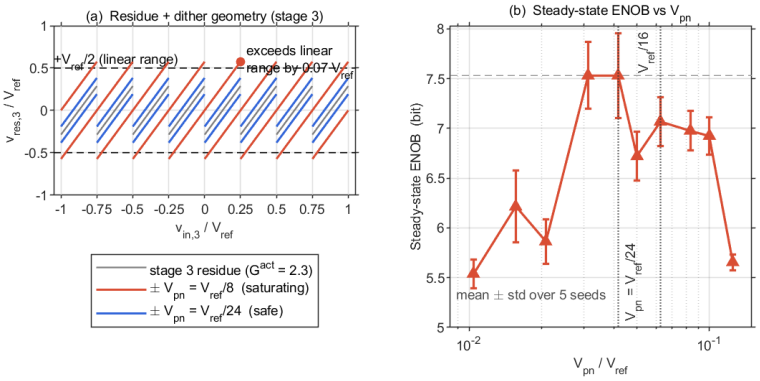


Figure 2.5: Dither-amplitude analysis for the target converter: (a) residue trajectory of stage 3 with two candidate dither envelopes; (b) end-to-end ENOB versus $V_{\mathrm{pn}}/V_{\mathrm{ref}}$, averaged over five PN seeds.

Figure 2.5(a) shows this headroom limit for the worst-case stage 3 ($G_3^{\mathrm{actual}} = 2.3$, $\varepsilon_3 = -42.5\,\%$). The choice $V_{\mathrm{pn}} = V_{\mathrm{ref}}/24$ (blue) shifts the residue by only $\approx 0.10\,V_{\mathrm{ref}}$ and stays inside the downstream linear range $\pm V_{\mathrm{ref}}/2$, whereas $V_{\mathrm{pn}} = V_{\mathrm{ref}}/8$ (red) overshoots that range by $\sim 0.08\,V_{\mathrm{ref}}$ at every sub-range boundary. Panel (b) confirms the trade-off at the output: the end-to-end ENOB (averaged over five PN seeds) rises with V_{pn} as the gradient SNR improves, peaks at

$V_{\mathrm{pn}} = V_{\mathrm{ref}}/24$ within ~ 0.2 bit of the 7.7-bit floating-point reference, and rolls off toward larger V_{pn} as saturation sets in. The power-of-two alternative $V_{\mathrm{pn}} = V_{\mathrm{ref}}/16$ already sits on the saturation arm and costs roughly 0.5 bit, so $V_{\mathrm{pn}} = V_{\mathrm{ref}}/24$ is adopted throughout.

2.6 Reconstruction without Division

The ideal reconstruction in Equation (2.2) contains four divisions, one for each inter-stage gain G_k. In a fixed-point implementation these divisions are expensive: dividing by a non-power-of-two quantity requires either a sequential reciprocal computation (yielding multi-cycle latency) or a precomputed lookup table whose precision must match the gain-estimate word length. Both options enlarge the digital back-end significantly and complicate timing closure.

A simple algebraic rearrangement avoids the divisions altogether. Multiplying both sides of Equation (2.2) by the gain product $G_{\mathrm{tot}} \triangleq G_1G_2G_3G_4$ yields

$$\begin{aligned} G_{\mathrm{tot}}\, D_{\mathrm{out}} \;=\; & D_1\,(G_1G_2G_3G_4) \;+\; D_2\,(G_2G_3G_4) \\ & +\; D_3\,(G_3G_4) \;+\; D_4\,(G_4) \;+\; D_5. \end{aligned} \tag{2.17}$$

Equation (2.17) requires only multiplications and additions of the stage codes by progressively shorter products of the stage gains. Denote the left-hand side as the scaled output

$$\tilde{D}_{\mathrm{out}}[n] \;\triangleq\; G_{\mathrm{tot}}\, D_{\mathrm{out}}[n]. \tag{2.18}$$

Two observations make $\tilde{D}_{\mathrm{out}}$ a suitable substitute for the original D_{out}.

Constant scaling preserves spectral content. The mapping $D_{\mathrm{out}} \mapsto \tilde{D}_{\mathrm{out}}$ is a single multiplication by a constant G_{tot}. In steady state—once the LMS loops have converged—G_{tot} is a fixed number that depends only on the manufactured gains of the analog front-end, not on the input signal or on time. All standard ADC performance metrics, including signal-to-noise-and-distortion ratio (SNDR), spurious-free dynamic range (SFDR), and the derived effective number of bits (ENOB), are defined as ratios of in-band signal power to noise or harmonic power and are therefore invariant under multiplication by any non-zero constant. In spectral terms, $\tilde{D}_{\mathrm{out}}$ and D_{out} are equivalent up to a calibration of the digital amplitude reading, which a downstream processing block can absorb at no additional silicon cost.

The constant is well-defined and bounded. Even when each G_k deviates from nominal by tens of percent, the product G_{tot} remains in a narrow numerical range ($G_{\mathrm{tot}} \approx 50$ for the target converter, against a nominal value of 256). The bit width of $\tilde{D}_{\mathrm{out}}$ is therefore predictable and modest, and no overflow protection is needed beyond the standard headroom already provided by the fixed-point datapath.

Because $\tilde{D}_{\mathrm{out}}$ delivers identical spectral fidelity at substantially lower implementation cost, the digital back-end reconstructs $\tilde{D}_{\mathrm{out}}$ rather than D_{out}. The detailed datapath that computes the four nested products $(G_1G_2G_3G_4)$, $(G_2G_3G_4)$,

(G_3G_4), and (G_4) from the four LMS gain estimates—together with the bit-width planning that keeps each intermediate signal within budget—is the subject of Section 3.6.

2.7 Geometric Bias and the 4-Bit Final Stage

A subtle interaction between the LMS update and the pipeline architecture motivates one further design choice: the use of a 4-bit flash final stage rather than a fifth 3-bit stage.

The cross-correlation identity (2.14) assumes that the quantiser of the downstream stage partitions the signal range symmetrically about the residue carried by the dither. In practice, the partition is set by a finite-resolution flash with sub-range width

$$V_{\text{step}} \;=\; \frac{V_{\text{ref}}}{2^{b_{\text{final}}-1}}, \tag{2.19}$$

where b_{final} is the resolution of the final flash quantiser. At sample instants where the residue lands near a sub-range boundary, positive and negative dither values are not handled symmetrically by the final quantiser: one polarity may cross the boundary and end up in an adjacent code, while the other polarity does not. The correlation (2.14) therefore acquires a residual bias whose magnitude is roughly proportional to $\varepsilon_k \, V_{\text{pn}}/V_{\text{step}}$.

Increasing b_{final} by one bit halves V_{step}, which roughly halves the geometric bias. A 4-bit final flash—which uses $2^4 = 16$ comparators instead of the 8 needed by a 3-bit stage—is therefore preferred over a fifth 3-bit stage of the same resolution budget. The hardware cost is modest (an additional eight comparators on a single, simple stage) and the calibration benefit is disproportionate, because the bias on every upstream LMS loop is limited by this same V_{step} at the bottom of the chain.

This concludes the theoretical foundation of the calibration scheme. The next chapter translates the algorithm into an explicit hardware architecture: a top-level block diagram, the per-stage LMS update unit, the forward-multiplication reconstruction datapath, and the fixed-point bit-width assignment that turns the floating-point parameters of this chapter into synthesisable Verilog.

Chapter 3

Calibration Architecture and Design Choices

3.1 Top-Level Architecture

The digital calibration back-end is realised in a single clock domain on the digital side of the analog/digital boundary shown in Figure 2.1. Three functional blocks make up the design: a *pseudo-noise generator* that supplies four independent dither sequences to the analog front-end and to the calibration loop; four *LMS update units*, one per calibrated 3-bit stage, that estimate the inter-stage gains $\hat{G}_1, \hat{G}_2, \hat{G}_3, \hat{G}_4$ in real time; and a *forward-multiplication reconstruction* network that produces the calibrated output $\tilde{D}_{\text{out}}[n]$. All three blocks share the same sample-rate clock and operate fully synchronously.

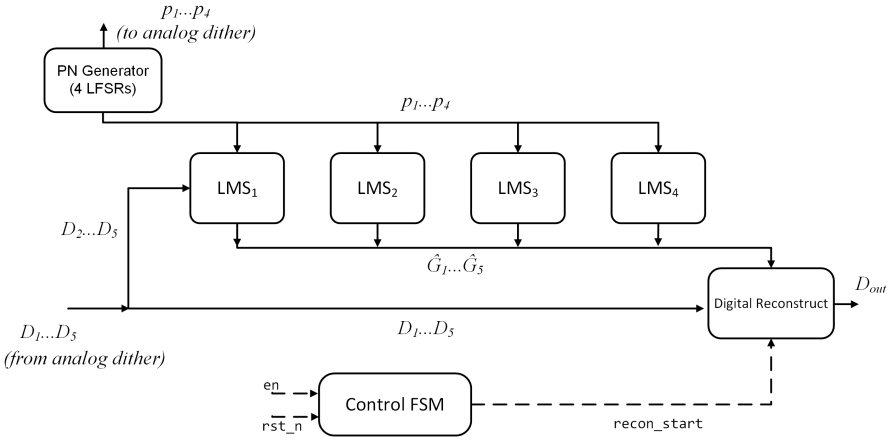


Figure 3.1: Top-level architecture of the calibration back-end: a PN generator, four LMS update units, and the forward-multiplication reconstruction network.

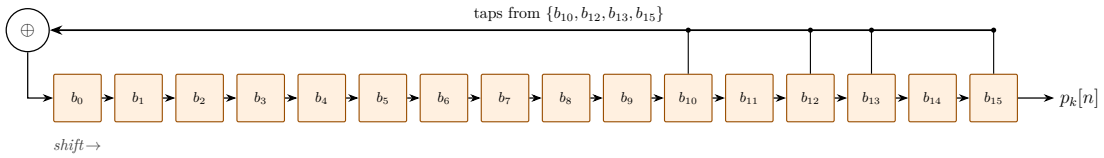
Every block reacts to the rising edge of a shared sample-enable strobe `en`. Within one `en` pulse, the PN generator advances its four LFSRs, each LMS update

unit consumes one downstream stage code and one PN bit and steps through its internal FSM, and the reconstruction unit latches the current stage codes and, once sufficient pipeline cycles have elapsed, releases a new calibrated sample on its output. The complete data-flow latency from `en` to `d_valid` is six clock cycles (Section 3.8); for a sample-rate target of 58.8 MHz this corresponds to roughly 100 ns of group delay.

The single-clock-domain choice deserves comment. A multi-clock arrangement—running the LMS loops at a fraction of the converter sample rate, for example—would save dynamic power proportionally to the duty-cycle ratio. In the present design, however, the LMS loops operate at the same rate as the converter so that every sample participates in the gradient estimate, which both maximises the effective signal-to-noise ratio of the gradient channel and simplifies the verification methodology. Power optimisation through clock gating remains as future work (Chapter 7).

3.2 PN Generator

The dither sequences $p_1[n], \ldots, p_4[n]$ are produced by four independent 16-bit Fibonacci linear-feedback shift registers (LFSRs). Each LFSR has a different seed and a different feedback tap polynomial chosen for maximal-length operation, so that the sequences are mutually uncorrelated as required by Equation (2.9). The period of a 16-bit maximal-length LFSR is $2^{16} - 1 = 65535$ samples, which at the target sample rate corresponds to roughly 1.1 ms. A maximal-length sequence has an ideal two-valued periodic autocorrelation, so over each period it presents a white spectrum to the LMS loop, as required by Equation (2.9); since this period is comparable to the convergence time of $\sim 1.3 \times 10^5$ samples, the loop averages over the full sequence during adaptation.



Four LFSRs run in parallel, each with a distinct tap polynomial:
LFSR_1: taps $\{10, 12, 13, 15\}$ (shown above)
LFSR_2: taps $\{\ 3, 12, 14, 15\}$
LFSR_3: taps $\{\ 0,\ 4, 13, 15\}$
LFSR_4: taps $\{11, 13, 14, 15\}$

Figure 3.2: One of the four 16-bit Fibonacci LFSRs in the PN generator; the four instances run in parallel with distinct seeds and tap polynomials.

In hardware terms the PN generator is the cheapest block of the design. Each LFSR amounts to a 16-stage shift register and a small XOR network; together the

four LFSRs require 64 flip-flops and a handful of two-input gates, occupying less than 1 % of the back-end's overall cell count. The output of the generator is taken directly from one of the shift-register taps, mapped from $\{0, 1\}$ to $\{-1, +1\}$ by interpreting the bit as a sign.

3.3 LMS Update Unit: Pipelined FSM

Equation (2.15), the per-sample LMS recursion, is implemented in hardware by a small finite-state machine that sequentialises the multiplication, subtraction, and accumulation over several clock cycles. Four identical instances of this update unit run in parallel, one per calibrated 3-bit stage, sharing only the sample-enable strobe.

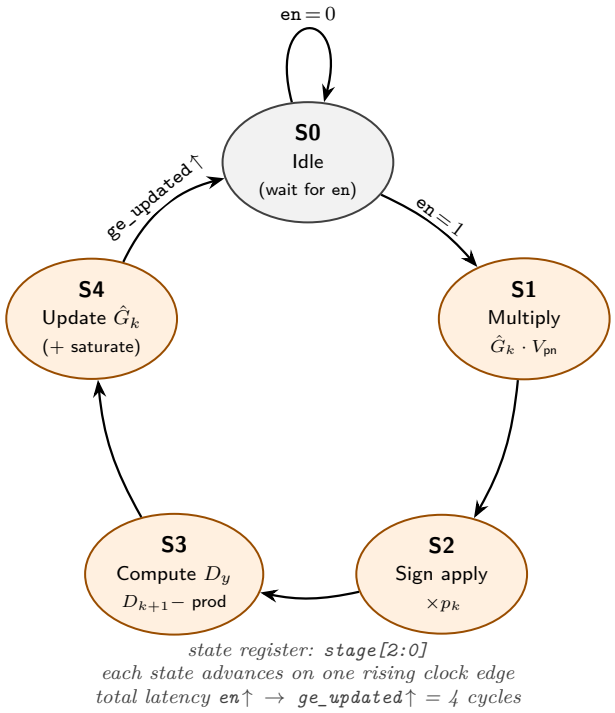


Figure 3.3: Five-state finite-state machine of one LMS update unit, with each state consuming one clock cycle.

The five states correspond directly to the operations of Equations (2.12)–(2.15). Splitting the computation into single-cycle states keeps the per-cycle critical path short—a single arithmetic operation per state, rather than a chain of three—and yields generous timing slack at the target 58.8 MHz clock. The total pipeline latency from a new `en` pulse to the updated gain estimate is four cycles.

Why an FSM rather than purely combinational logic. The whole update—multiply, sign-apply, subtract, shift, add, saturate—is logically a single function from $(\hat{G}_k, D_{k+1}, p_k)$ to the next $\hat{G}_k$, and a one-cycle combinational implementation would in principle suffice. Three practical considerations argue against it for the present design. *First*, the combinational chain—dominated by the $\hat{G}_k V_{\mathrm{pn}}$ multiplier—would place the entire arithmetic on a single critical path; after place-and-route the resulting delay on Artix-7 was estimated to exceed the 17 ns budget of the 58.8 MHz clock, leaving no slack for system-level routing. *Second*, the LMS update has no per-cycle throughput requirement: the calibration loop only needs one update per input sample, and a five-cycle FSM completes comfortably within one sample period of the digital back-end, so the extra latency is invisible at the system level. *Third*, splitting the computation across five clock phases allows the same adder/subtractor to be time-multiplexed (state S3 uses it as a subtractor, state S4 as an accumulator), shaving a non-trivial fraction of the per-engine area. A fully pipelined alternative—one combinational stage between each pair of pipeline registers—would also close timing but would require more registers and would offer a throughput that the application does not need.

Two implementation details merit mention.

Combinational multiplier. The product $\hat{G}_k \cdot V_{\mathrm{pn}}$ in state S1 is computed by a fixed-point combinational multiplier sized for the bit widths defined in Section 3.5. Because V_{pn} is a hardware constant in fixed-point representation (twelve bits holding the integer $V_{\mathrm{pn}}/V_{\mathrm{LSB}} = 683$ for $V_{\mathrm{pn}} = V_{\mathrm{ref}}/24$), this is a constant-coefficient multiplication that the synthesis tool generally maps to a Wallace-tree of compressed partial-product summations.

Saturation floor. A physical pipeline ADC cannot exhibit a stage gain below some minimum value $G_{\mathrm{e,MIN}}$ set by the residue amplifier's worst-case closed-loop gain. Without an explicit saturation, the LMS update would in principle allow $\hat{G}_k$ to drift below this floor under transient noise excursions, leading to numerical issues in the reconstruction network downstream. State S4 therefore clamps $\hat{G}_k$ from below at $G_{\mathrm{e,MIN}}$ before writing back to the gain register. An upper bound is unnecessary because the LMS gradient—being the difference between the actual and estimated gain—is bounded by the physical G_k^{actual} itself.

A pseudocode sketch of the central state S4 is shown in Listing 3.1. The complete Verilog source is available in the accompanying project repository.

Listing 3.1: Excerpt from the LMS update unit: state S4 (gain accumulation).

```
// State S4: accumulate and saturate.
// delta_q : S3.20 fixed point, gradient signal
// Ge_q : S3.20 fixed point, gain estimate
// GE_MIN_q : S3.20 fixed point, lower saturation bound
S4: begin
    Ge_next = $signed(Ge_q) + $signed(delta_q);
    if (Ge_next < $signed(GE_MIN_q))
        Ge_q <= GE_MIN_q; // saturate
    else
```

```
10          Ge_q <= Ge_next;
11      ge_updated <= 1'b1;
12      state <= S0; // return to Idle
13  end
```

3.4 Loop-Gain Selection

The LMS update of Equation (2.15) multiplies the correlation product $p_k[n]\, D_y[n]$ by a scalar step size μ. In a fixed-point implementation, this multiplication is realised most economically by choosing μ to be a negative power of two, so that the operation degenerates into an arithmetic right shift. The implementation parameter that exposes this choice is the `DELTA_SHIFT` constant of the LMS update unit, which sets the number of bits by which the error signal is right-shifted before being accumulated into $\hat{G}_k$:

$$\mu_{\text{eff}} \;=\; 2^{-(\texttt{DELTA\_SHIFT}+K)}, \tag{3.1}$$

where K is a fixed scaling constant that arises from the dither amplitude $V_{\text{pn}} = V_{\text{ref}}/24$ adopted in Section 2.5 and from the fixed-point scaling of the LMS error. Numerically, $K \approx 11.4$ for the bit widths assigned in Section 3.5.

The choice of `DELTA_SHIFT` thus selects a discrete operating point on the effective step-size axis, with `DELTA_SHIFT` $+1$ halving μ_{eff}. The floating-point step-size sweep of Section 2.4 guides this choice. As μ_{eff} is reduced the steady-state ENOB rises and then flattens—below about $\mu = 2^{-12}$ further reduction recovers progressively less ENOB—while the convergence time continues to grow as $1/\mu_{\text{eff}}$. The largest μ_{eff} that retains the asymptotic ENOB to within 0.1 bit falls near $\mu_{\text{eff}} = 2^{-12.4} \approx 1.85 \times 10^{-4}$. At this operating point the convergence time is $\sim 1.3 \times 10^5$ samples, which at the target sample rate of 58.8 MHz amounts to roughly 2.2 ms—comfortably within the calibration-startup budget of typical pipeline-ADC applications.

The corresponding implementation parameters are `DELTA_SHIFT` $= 1$ and $V_{\text{pn}} = V_{\text{ref}}/24$; together they realise $\mu_{\text{eff}} = 2^{-12.4}$ as required. All subsequent sections of this chapter assume these parameter values.

3.5 Fixed-Point Bit-Width Design

This section addresses the most important implementation choice of the digital back-end: the bit-width allocation of the gain accumulator $\hat{G}_k$, the LMS error register D_y, and the gradient step δ. All three quantities are signed fixed-point values whose word lengths must be large enough to preserve gradient information yet small enough to keep area and power within budget. The width of the gain accumulator turns out to be the binding constraint, and the analysis below focuses on it.

The gradient-preservation condition

The LMS recursion of Equation (2.15) updates $\hat{G}_k$ by a small increment $\delta \triangleq \mu_{\text{eff}}\, p_k\, D_y$. Once the loop is near steady state the magnitude of δ is set by the product of μ_{eff} and the typical gradient-channel error, giving a typical sample-by-sample magnitude on the order of $|\delta_{\text{typ}}| \approx 2^{-16}$ in normalised units.

Suppose the gain register is stored in M fractional bits. Its least significant bit then has value 2^{-M}. In hardware, the gradient δ is generated by an arithmetic right shift of the error signal D_y. For two's-complement arithmetic, the right shift operator $\gg\gg$ rounds positive operands toward zero and negative operands toward $-\infty$, which introduces a small asymmetric bias of order one LSB per operation. Provided $|\delta_{\text{typ}}|$ is comfortably larger than 2^{-M}, this rounding bias is washed out by the genuine gradient information. When $|\delta_{\text{typ}}| \approx 2^{-M}$, however, the bias and the gradient are of comparable magnitude, and the loop ceases to track the true gain error; instead, it pseudo-converges to a nearby point on the M-bit quantisation grid.

The condition that the gradient survive the rounding bias can therefore be written compactly as

$$2^{-M} \ll |\delta_{\text{typ}}| \iff M > M_{\min} \triangleq \log_2 \frac{1}{|\delta_{\text{typ}}|} \approx 16. \tag{3.2}$$

Equation (3.2) defines the boundary of a regime in which fixed-point quantisation of $\hat{G}_k$ silently destroys the calibration loop. Below the boundary the system enters what is termed throughout this thesis the *stochastic-quantisation regime*; above the boundary the gradient is preserved and the loop behaves as the floating-point analysis predicts.

A detailed derivation of $M_{\min}$ from the underlying right-shift rounding analysis is omitted here for brevity; Equation (3.2) captures the operationally important result.

Parametric sweep

To explore the design space around the boundary, the floating-point reference is replaced by an otherwise identical fixed-point model in which only the gain accumulator's fractional bit count M is varied. Four representative working points are evaluated; the post-convergence ENOB and the FPGA synthesis cost of each are reported in Table 3.1.

The data confirm the regime boundary. At $M = 16$, the LSB of $\hat{G}_k$ equals the typical gradient magnitude, the loop pseudo-converges, and the ENOB collapses to 5.23 bit despite the loop appearing nominally stable. At $M = 18$, the LSB is four times smaller than the gradient magnitude and the loop tracks the true gain in expectation, but the residual rounding bias still consumes roughly 1 bit of ENOB. At $M = 20$, the LSB is sixteen times smaller than the gradient; the rounding bias is negligible and the ENOB sits within 0.26 bit of the full-precision reference at $M = 28$.

Table 3.1: Bit-width Pareto for the gain accumulator $\hat{G}_k$. The fractional-bit count M is varied while all other widths are held fixed. ENOB is measured at steady state on the fixed-point model; LUT/FF counts come from Vivado synthesis on the Artix-7 XC7A100T target at 58.8 MHz.

Configuration	M	ENOB (bit)	LUT	FF
Stochastic-quantisation regime	16	5.23	~ 1220	~ 580
Marginal boundary	18	6.41	~ 1330	~ 630
Selected operating point	**20**	**7.20**	**721**	**577**
Full-precision reference	28	7.46	1539	699

Working-point selection

A safety margin of one binary order of magnitude above the boundary $M_{\min}$ is chosen, yielding $M = 20$. This margin ensures that the residual rounding bias remains below the LMS gradient even under worst-case temperature- and supply-induced drift of the analog gains; it also provides headroom for any future increase in μ_{eff} should the application benefit from faster convergence.

The corresponding fixed-point format is S3.20 for $\hat{G}_k$ (three integer bits to span the physical gain range $[-8, 8)$, twenty fractional bits, signed two's-complement, 24 bits total). Section 3.9 collects the bit widths of the remaining signals—LMS error, gradient, reconstruction accumulators, output—into a single summary table. All of them flow from the choice of $M = 20$ and the multiplications and shifts that connect them to the gain accumulator.

The headline implementation benefit is the 53 % reduction in LUT count (from 1539 to 721) and the 17 % reduction in flip-flop count (from 699 to 577) relative to the full-precision reference. The cost is a 0.26-bit reduction in ENOB (from 7.46 to 7.20), which the application can absorb given a baseline target of ENOB ≥ 7.0. The DSP-block count change reported in Chapter 4 (an increase of +4 DSP48 blocks despite the overall logic shrinkage) is unrelated to the bit-width choice and is discussed there in the context of FPGA-specific mapping boundaries.

3.6 Forward-Multiplication Reconstruction

Section 2.6 established that the calibrated output can be computed in multiplication-only form by accumulating the five stage codes weighted by progressively shorter products of the four estimated gains. This section turns that algebraic statement into a concrete hardware datapath, with attention to the bit-width budget at each intermediate node.

Datapath organisation

The reconstruction unit consumes the five stage codes $D_1, \ldots, D_5$ and the four gain estimates $\hat{G}_1, \ldots, \hat{G}_4$ once per sample, and emits the scaled output $\tilde{D}_{\text{out}}$ after

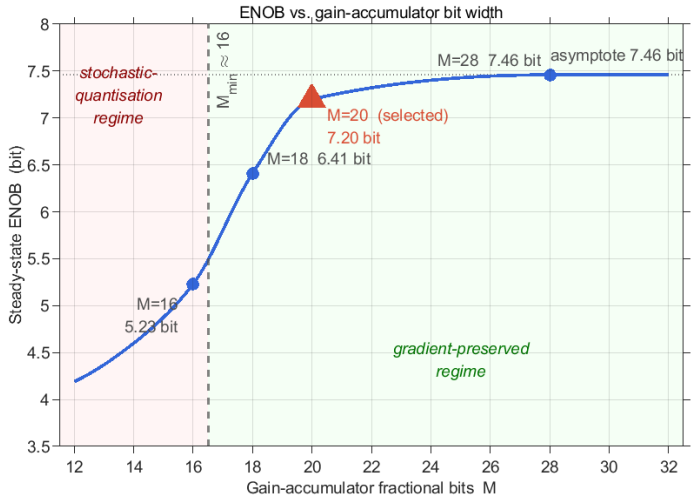


Figure 3.4: Steady-state ENOB versus gain-accumulator fractional-bit count M, with the regime boundary at $M \approx 16$ and the selected operating point at $M = 20$.

a fixed five-cycle pipeline delay. The computation is organised around four cumulative products that are reused across the five summation terms of Equation (2.17):

$$W_4 = \hat{G}_4, \tag{3.3}$$

$$W_{34} = \hat{G}_3\hat{G}_4, \tag{3.4}$$

$$W_{234} = \hat{G}_2\hat{G}_3\hat{G}_4, \tag{3.5}$$

$$W_{1234} = \hat{G}_1\hat{G}_2\hat{G}_3\hat{G}_4. \tag{3.6}$$

The output is then assembled in a single summation,

$$\tilde{D}_{\text{out}} = D_1W_{1234} + D_2W_{234} + D_3W_{34} + D_4W_4 + D_5, \tag{3.7}$$

right-shifted by a constant $\texttt{SHIFT\_OUT} = 10$ bits to align $\tilde{D}_{\text{out}}$ with the 13-bit signed output port. The final shift folds the fixed-point scaling of $\hat{G}_k$ into the output representation; its precise value follows from the bit widths listed in Section 3.9.

Pipeline and multiplier mapping

The four cumulative products are computed in stages T1–T4 of the reconstruction FSM, with the summation in stage T5. Each stage contains at most one fixed-point multiplication and one register write-back, keeping the per-cycle critical path short. Synthesis maps each multiplication onto a Wallace-tree array of carry-save adders followed by a final propagate adder; on the FPGA target the 24-bit operand widths involved keep most products inside a single DSP48 block, while on the standard-cell ASIC target the same products are realised as combinational cell trees (Chapter 5).

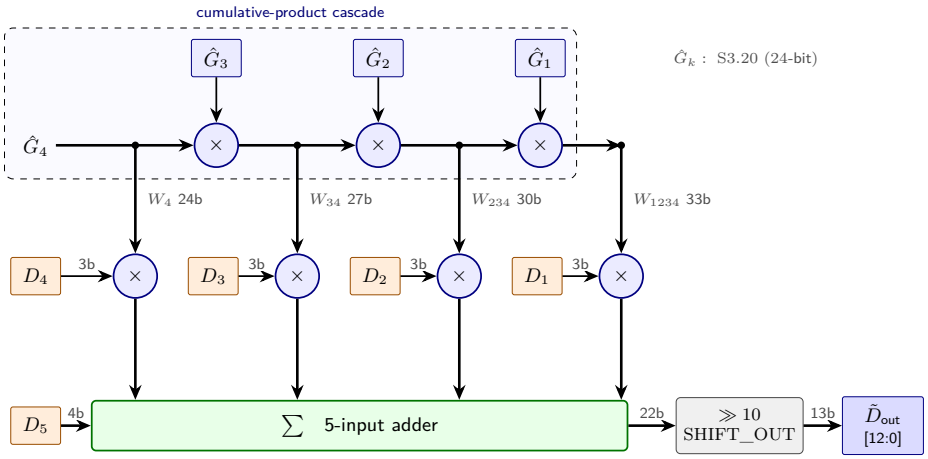


Figure 3.5: Hardware datapath of the forward-multiplication reconstruction: the cumulative-product cascade $W_4 \to W_{34} \to W_{234} \to W_{1234}$ feeds four $D_k \times W_*$ multipliers whose products, together with the final-stage code D_5, are summed and right-shifted to the 13-bit output, with signal bit widths annotated at each node.

Bit-width planning

The bit width of each cumulative product is the sum of the widths of its operands minus any redundancy in the gain factor. Starting from $\hat{G}_k$ at S3.20 (24 bits) and the stage codes at S3.0 or S4.0, the intermediate products grow predictably:

$$W_4 : \text{S3.20 (24-bit, same as } \hat{G}_4), \tag{3.8}$$
$$W_{34} : \text{S6.40 (47-bit before bit-growth pruning)}, \tag{3.9}$$
$$W_{234} : \text{S9.60 (70-bit, pruned to 32-bit by truncating} \tag{3.10}$$
$$\text{the bottom fractional bits)}, \tag{3.11}$$
$$W_{1234} : \text{S12.80 (truncated similarly)}. \tag{3.12}$$

The bit-growth pruning truncates the bottom-most fractional bits of each cumulative product after the bound predicted by the gradient-preservation analysis of Section 3.5—that is, no fractional bit finer than $|\delta_{\text{typ}}|$ is retained—which keeps the summation accumulator at S9.12 (22-bit signed) and the post-shift output at S3.10, right-aligned to the 13-bit signed output port.

3.7 Why No Post-Smoothing Filter

A natural reflex on observing the residual LMS jitter is to insert a low-pass filter between $\hat{G}_k$ and the reconstruction network. Two analytical arguments rule this out.

Spectral argument. Linearising Equation (2.15) around the converged fixed point shows that $\hat{G}_k$ behaves as the output of a single-pole IIR filter driven by white gradient noise, with corner frequency

$$f_{\mathrm{LMS}} \;\approx\; \frac{\mu_{\mathrm{eff}}\, V_{\mathrm{pn}}}{2\pi}\, f_s. \tag{3.13}$$

At the chosen operating point ($\mu_{\mathrm{eff}} = 2^{-12.4}$, $V_{\mathrm{pn}} = V_{\mathrm{ref}}/24$, $f_s = 58.8\,\mathrm{MHz}$), $f_{\mathrm{LMS}} \approx 70\,\mathrm{Hz}$—roughly four orders of magnitude below the converter's in-band signal frequencies ($f_{\mathrm{in}} \sim 0.5$–$10\,\mathrm{MHz}$). An effective post-smoothing filter would therefore have to operate at an even lower corner—in the sub-hertz range—which in fixed-point hardware requires an extremely small leak coefficient and a long internal word length, dwarfing the savings of the bit-width Pareto of Section 3.5.

Stability argument. A post-filter inserted in the calibration feedback loop introduces delay, which (per standard adaptive-filter stability theory) narrows the admissible range of μ_{eff} before the loop becomes oscillatory. Pushing μ_{eff} down to recover stability margins reintroduces the convergence-time problems the filter was meant to alleviate.

The proposed back-end therefore feeds $\hat{G}_k$ directly into the reconstruction network without intermediate filtering, saving roughly 150 LUTs (FPGA) or proportionate area on the ASIC and avoiding the extra pole in the calibration loop.

3.8 Top-Level Integration and Timing

The three blocks introduced in this chapter are co-ordinated by a small control finite-state machine. Upon assertion of the sample-enable strobe `en`, the control FSM advances a six-state sample counter `sample_cnt`; each state of the counter corresponds to a pipeline cycle in which the LMS update units and the reconstruction unit progress through their internal stages. The reconstruction unit is started when `sample_cnt` reaches the value 5, and its 13-bit output becomes valid one cycle later, indicated by the assertion of `d_valid`.

Two consequences follow. First, the back-end is a pure pipeline with deterministic latency: an external host can issue `en` pulses at the full sample rate and rely on a six-cycle delay before the corresponding output appears. Second, the LMS update of stage k uses the downstream stage code from the *same* `en` pulse, so the gradient estimate is always evaluated on a coherent slice of the pipeline state. No additional retiming or synchronisation logic is required between the analog front-end and the digital back-end.

3.9 Fixed-Point Format Summary

Table 3.2 consolidates the bit widths and fixed-point formats of the principal signals in the calibration back-end. The notation S$i.f$ denotes a signed two's-complement fixed-point number with i integer bits (excluding the sign) and f

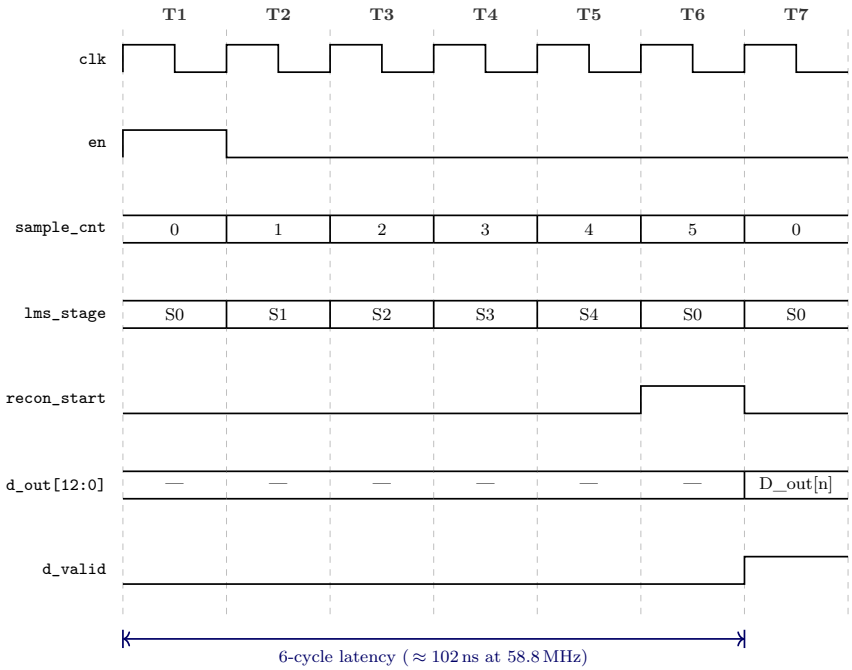


Figure 3.6: Top-level pipeline timing: a single `en` pulse triggers the sample counter, the four parallel LMS FSMs, and the reconstruction module, yielding a 6-cycle latency from `en` to `d_valid` (≈ 102 ns at 58.8 MHz).

fractional bits; total word length is $i + f + 1$. All formats trace back to the gain-accumulator choice $\hat{G}_k$: S3.20 from Section 3.5; the remaining widths follow by counting bits of growth through the operations of Sections 3.3 and 3.6.

The formats in Table 3.2 are simultaneously the specification handed to the RTL implementation (every register and every wire is sized accordingly) and the contract that the fixed-point simulation model must satisfy. Any RTL revision that preserves these widths is, by construction, guaranteed to reproduce the simulation-validated ENOB of Table 3.1 to within the LSB precision of the output port.

This concludes the architectural specification. The next chapter reports how this specification is closed in timing on a Xilinx Artix-7 FPGA, what resource cost it actually incurs, and how the resulting RTL behaves on real silicon.

Table 3.2: Fixed-point formats of the principal back-end signals.

Signal	**Format**	**Width**	**Notes**
Stage code D_k (3-bit)	U3.0	3	flash output
Stage code D_5 (final flash)	U4.0	4	flash output
PN bit p_k	1 b sign	1	mapped to $\{-1, +1\}$
Dither V_{pn}	U0.12	12	const. $V_{\text{ref}}/24 \mapsto 683$
Gain accumulator $\hat{G}_k$	S3.20	24	primary design variable (§3.5)
LMS error D_y	S5.10	16	after stage-code subtraction
Gradient δ	S3.20	24	sign-ext., then $\gg\gg$ `DELTA_SHIFT`
Saturation floor $G_{\text{e,MIN}}$	S3.20	24	hardware constant
Cumulative W_4	S3.20	24	$= \hat{G}_4$
Cumulative W_{34}	S6.20	27	after pruning
Cumulative W_{234}	S9.20	30	after pruning
Cumulative W_{1234}	S12.20	33	after pruning
Reconstruction accumulator	S9.12	22	sum of five weighted contributions
Output $\tilde{D}_{\text{out}}$	S3.10	14	before output-shift alignment
Output port `d_out`	S12.0	13	after $\gg$ `SHIFT_OUT` = 10

Chapter 4
FPGA Implementation

The architecture specified in the preceding chapter is realised in Verilog and verified on a Xilinx Artix-7 XC7A100T device (speed grade -1). The implementation flow uses Xilinx Vivado 2017.4. This chapter reports the synthesis results, the timing-closure trajectory, and the power numbers, and closes with a side-by-side comparison against the full-precision reference variant of the design. The fixed-point parameters analysed in Chapter 3—$\hat{G}_k$ in S3.20, $V_{\mathrm{pn}} = V_{\mathrm{ref}}/24$, and `DELTA_SHIFT` = 1—are used throughout.

4.1 Vivado Synthesis Flow

The design enters the flow as five Verilog source files (the PN generator, the LMS engine, the reconstruction module, the top-level integration module, and a self-checking test bench), together with a single Xilinx Design Constraints file that specifies the target clock period, input and output delays for the converter-facing port group, and the preferred I/O standard. Vivado's default synthesis strategy is used; no manual partitioning, blackboxing, or out-of-context flow is required. A short post-synthesis check confirms that the elaborated netlist contains no inferred latches and that every combinational path between flip-flops is registered—both standard hygiene properties that the Verilog coding style enforces by construction.

One small style choice deserves comment: the output port `d_out[12:0]` is driven directly from a register at the end of the reconstruction pipeline, and the constraint file marks this register for I/O-buffer (IOB) packing. Pulling the register into the IOB removes a single stage of routing delay from the output path, which improves timing slack on the device pads. No other manual placement directives are needed.

4.2 Resource Utilisation

Synthesis reports the post-implementation resource counts shown in the upper portion of Table 4.1. The headline number is the look-up-table (LUT) count of 721, which together with the 577 flip-flops and 11 DSP48 blocks occupies less than 1.5 % of the XC7A100T fabric. Block RAMs are not used; all state is held in distributed registers and small constant look-ups.

Table 4.1: FPGA implementation summary on Artix-7 XC7A100T at 58.8 MHz. Both the proposed (resource-optimised) design and the full-precision reference are shown; absolute counts are Vivado post-implementation values, percentages are of the XC7A100T total.

Metric	Proposed	Reference	Saving
Slice LUT	721	1539	−53%
as % of XC7A100T	1.1%	2.4%	–
Slice flip-flop	577	699	−17%
DSP48E1	11	7	+4
Block RAM	0	0	–
Worst negative slack (WNS)	+0.44 ns	+0.43 ns	–
Maximum closed frequency	∼ 60 MHz	∼ 60 MHz	–
Total power	113 mW	126 mW	−10%
dynamic	∼ 35 mW	∼ 41 mW	−15%
static	∼ 78 mW	∼ 85 mW	−8%
Steady-state ENOB	7.20 bit	7.46 bit	–

The savings against the full-precision reference are sizeable: the LUT count drops by 53 % and the flip-flop count by 17 %, both consequences of the gain-accumulator and gradient widths chosen in Section 3.5. Two aspects deserve a brief commentary.

Counter-intuitive +4 DSP48 blocks. The DSP48E1 block of the Artix-7 fabric provides a hardened 25×18 signed multiplier. In the full-precision reference, the gain accumulator is 32 bits wide and the dither constant is 13 bits wide, so the $\hat{G}_k \cdot V_{\mathrm{pn}}$ product (32×13) exceeds the native DSP48E1 width and Vivado maps it as a fabric-based multiplier formed from LUT cascades, consuming LUTs but not DSP blocks. In the proposed design, the gain accumulator is reduced to 24 bits, so the same product (24×13) now fits comfortably inside a single DSP48E1, and Vivado emits one DSP per LMS update unit (plus three more for the reconstruction Wallace tree). The net effect is that bit-width reduction *frees* LUTs by moving work into hardened DSPs, which is the principal reason the LUT count falls more than the flip-flop count.

Sub-modules. Of the 721 LUTs in the proposed design, roughly 35 % are consumed by the four LMS update units (90 LUTs each on average), 45 % by the reconstruction Wallace tree, and the remaining 20 % by the PN generator, the sample-counter FSM, and assorted control glue. The flip-flop allocation follows a similar pattern, with the reconstruction pipeline accounting for the largest single share.

4.3 Timing Closure

Closing the design at the initially aspirational target of 100 MHz proves infeasible. Setting the constraint file's target period to 10 ns and running the default implementation yields a worst negative slack of -5.4 ns across 29 failing endpoints. Inspecting the critical paths reveals a consistent pattern: every failing endpoint terminates inside the reconstruction Wallace tree, where a 47-level combinational chain through several cascaded multiplications must complete in a single cycle.

A natural remedy would be to pipeline the multiplier chain. Adding a single register stage at the mid-point of the Wallace tree would roughly halve the chain length, but at the cost of one additional cycle of end-to-end latency and a non-trivial revisiting of the bit-width budget developed in Section 3.6. Because the calibration latency budget at the application's $\sim$ 1 MHz signal bandwidth has substantial headroom, a simpler alternative is preferred: relaxing the clock-period target. Two further iterations of the constraint file converge on a target period of 17 ns (58.8 MHz) at which all paths close with positive slack, as shown in Figure 4.1.

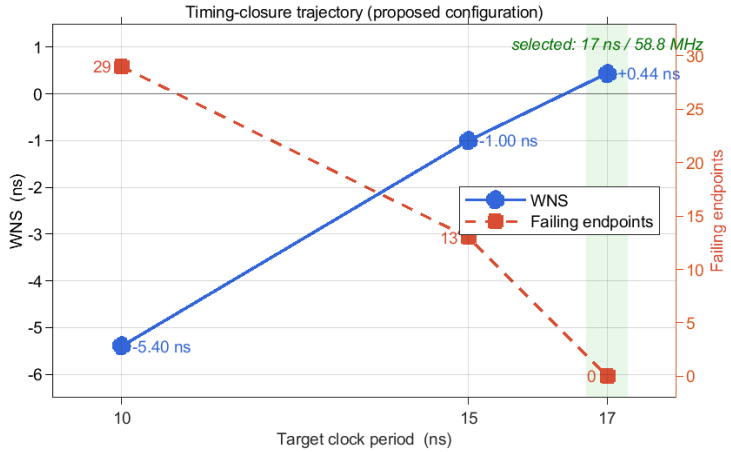


Figure 4.1: Timing-closure trajectory of the proposed design: WNS crosses zero and failing endpoints reach zero at the selected 17 ns (58.8 MHz) operating point.

The reconstruction-chain critical path is intrinsic to a single-cycle multiplication-only datapath: the cascaded $\hat{G}_1\hat{G}_2\hat{G}_3\hat{G}_4$ product carries a combinational depth that grows linearly with the number of calibrated stages and that no amount of operator factoring can reduce within a single clock cycle. Pipelining is therefore the correct long-term remedy and is recorded as future work in Chapter 7. For the present implementation the chosen 58.8 MHz remains comfortably above the 1 MHz signal bandwidth of the target ADC application.

4.4 Behavioural Verification

Beyond meeting timing, the implemented design must calibrate correctly. A behavioural simulation of the back-end RTL—the same description carried through synthesis—confirms that it does. Figure 4.2 plots the four LMS gain estimates $\hat{G}_1$–$\hat{G}_4$ over a long simulation run: starting from their reset values, all four converge smoothly toward the neighbourhood of the analog stage gains (3.5, 2.8, 2.3, 2.2) and then hold steady for the remainder of the run. This reproduces, at the fixed-point word lengths selected in Chapter 3, the convergence behaviour of the floating-point reference and verifies that the synthesised loop is functionally correct.

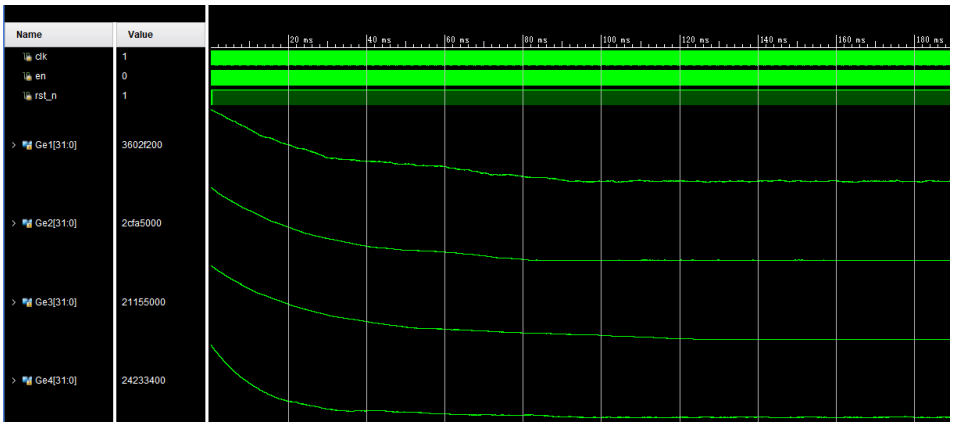


Figure 4.2: Behavioural-simulation waveform of the implemented back-end: the four LMS gain estimates $\hat{G}_1$–$\hat{G}_4$ (rendered as analog traces) converge from their reset values toward the analog stage gains and then remain stable, confirming correct operation at the selected fixed-point word lengths.

4.5 Power Analysis

Vivado's vector-less power-estimation flow reports a total on-chip power of 113 mW for the proposed design at the closed operating point. The breakdown by source is shown in Table 4.2. Most of the figure—78 mW or 69 % of the total—is static, reflecting the fact that an XC7A100T's idle current must be paid regardless of the design's size. Of the $\sim$ 35 mW of dynamic power, the largest single share is consumed by signal and clock routing across the fabric, followed by the DSP48 blocks of the reconstruction pipeline.

The full-precision reference reports 126 mW under the same flow, yielding the 10 % overall power reduction listed in Table 4.1. Most of this saving is dynamic (-15 %), attributable to the smaller gain-accumulator flip-flop count and the reduced switching activity in the LMS update units. The static power reduction is correspondingly smaller because the FPGA's quiescent current does not scale with the design's footprint.

Table 4.2: Vivado power breakdown at 58.8 MHz, default activity factor 0.125.

Component	**Power (mW)**	**Share**
Static (device idle)	78	69%
Clocks	6	5%
Signals + routing	12	11%
Logic (LUT + FF)	6	5%
DSP blocks	8	7%
I/O	3	3%
Total	113	100%

It is worth emphasising that the 113 mW figure is dominated by FPGA-specific overhead unrelated to the calibration design: the device's full quiescent current, the always-on clock-routing network, and the always-on power supply for unused logic regions all contribute to the total. Chapter 5 returns to this point in the context of the standard-cell ASIC implementation, where the same RTL dissipates only a small fraction of this power.

4.6 FPGA Summary

Table 4.1 above consolidates the headline numbers of this chapter. The bit-width Pareto adopted in Chapter 3—in particular the choice of $M = 20$ fractional bits for the gain accumulator—translates on the FPGA target into a 53 % reduction in LUT count, a 17 % reduction in flip-flop count, and a 10 % reduction in total power, at a cost of 0.26 bit of steady-state ENOB. At a target frequency of 58.8 MHz, both the proposed and the reference variants close timing with comparable positive slack; the maximum closed frequency is limited by the combinational depth of the reconstruction Wallace tree, which is independent of the gain-accumulator width.

These results establish the resource-and-power footprint of the calibration back-end on a representative FPGA target. The next chapter takes the same RTL through a standard-cell ASIC flow and examines how those numbers change on real silicon.

Chapter 5
ASIC Digital Back-End

The same Verilog source used in Chapter 4 is taken through a standard-cell digital back-end flow. This chapter reports the synthesis, placement, routing, and verification stages, and closes with an explicit comparison against the FPGA implementation of the previous chapter. The scope ends at post-layout sign-off and gate-level functional simulation; no tape-out is undertaken.

5.1 Process and Library Selection

The target process is a mainstream 65 nm CMOS standard-cell library. The library variant chosen for the digital back-end favours low static leakage current—on the order of nanoamperes per cell—at the expense of a modest reduction in switching speed relative to higher-speed variants of the same node. This trade is appropriate for the present design: the target frequency of 58.8 MHz lies an order of magnitude below the speed limit of any 65 nm variant, so the speed penalty is negligible, while the leakage benefit is fully recovered. The 65 nm node itself is a mature manufacturing target with a stable PDK ecosystem, which favours reuse of the digital back-end as a soft macro in larger SoC contexts.

Library characterisation is performed at the worst-case sign-off corner: $V_{DD} = 1.0\,\mathrm{V}$, $T_j = 125\,^{\circ}\mathrm{C}$, slow process, 10-year aging. The Composite Current Source (CCS) timing format is used in preference to the older Non-Linear Delay Model (NLDM) because of its more accurate handling of the non-rectangular driver waveshapes that arise in the heavily-multiplied datapaths of the reconstruction network. A single corner is sufficient for the academic scope of this work; multi-corner multi-mode analysis would be undertaken at sign-off prior to tape-out.

5.2 Logic Synthesis

Synthesis is performed with Cadence Genus version 16 in legacy command-line mode. The flow is organised as three TCL scripts: a design-setup script (library search path, target library, design name); a clock-definition script (clock period 17 ns matching the FPGA closure target, plus input/output delay constraints); and a top-level synthesis driver that reads RTL, elaborates, applies the SDC, and

runs the three-stage `syn_generic` / `syn_map` / `syn_opt` sequence. The complete TCL source is available in the accompanying project repository.

Elaboration confirms that the Verilog netlist contains no inferred latches and that every module resolves cleanly against the imported library. After mapping and post-map optimisation, Genus reports the implementation summary shown in Table 5.1.

Table 5.1: Genus synthesis summary on the 65 nm CMOS target, worst-case corner, at the target 58.8 MHz clock period.

Metric	**Value**
Total instance count	8 880
sequential (flip-flops)	625
combinational	8 255
Cell area	30 079 µm^2
Worst negative slack (post-syn)	+20 ps
Total negative slack	0 ns
Violating paths	0
Inferred latches	0
Critical-path levels of logic	47

The cell count of 8 880 deserves a moment of reflection. At first glance it appears strikingly larger than the FPGA's 721 LUTs, but the comparison is not apples-to-apples. Each DSP48E1 block of the Artix-7 fabric is a hardened 25×18 multiplier-accumulator that the ASIC standard-cell library must reproduce from primitives—typically a Wallace-tree carry-save array followed by a final propagate adder, which consumes on the order of 700 standard cells per multiplier. The proposed design uses 11 DSP48 blocks on the FPGA target (Section 4.2), which therefore corresponds to approximately 7 700 ASIC cells of multiplier infrastructure alone, in good agreement with the synthesis report. The remaining cells are the LMS update units, the PN generator, the reconstruction control logic, and assorted glue.

The worst negative slack of +20 ps is notably tight. All four LMS update units, the PN generator, and the reconstruction front-end report comfortable positive slack, but a single critical path through the reconstruction Wallace tree consumes nearly the entire 17 ns budget. The path traverses 47 levels of logic from a registered flip-flop on a stage-code input through the cascaded multiplier of the reconstruction unit and back to a flip-flop at the final summation. This is the same path identified on the FPGA target in Section 4.3; the combinational depth is an intrinsic property of the single-cycle multiplication-only datapath and cannot be reduced further without adding a pipeline stage.

Although a thin +20 ps slack would be cause for concern at sign-off, it is acceptable as a synthesis result because the synthesis tool computes timing using a pessimistic wireload model that does not yet reflect the real interconnect parasitics. Section 5.3 shows that placement and routing recover roughly 320 ps of additional

margin once buffer insertion and true RC extraction are taken into account.

5.3 Floorplanning and Place-and-Route

The synthesised netlist is fed to Cadence Innovus version 22.11 for the physical implementation flow. The flow is broken into six TCL scripts that step the design from floorplan creation through to GDS export; the complete sources are available in the accompanying project repository. This section describes the floorplan and power plan, the placement and clock-tree synthesis outcomes, the routing result, and the post-route timing recovery that lifts the design from its +20 ps synthesis margin to a comfortable +340 ps slack.

Floorplan

The chip is laid out as a square macro of 220×220 µm core area surrounded by 6 µm of margin on each side, giving a die size of 232×232 µm. No I/O pad ring is added—the design is intended as an embedded soft macro within a larger SoC—so the die boundary is the macro boundary. With a total cell area of 30 079 µm^2 inherited from synthesis, the corresponding core-utilisation figure is 63 %, which leaves sufficient routing channels for the multiplication-heavy reconstruction network while keeping wire lengths short.

Power planning

A two-layer power grid is constructed prior to placement. The core ring is realised on metal-2 (horizontal segments) and metal-3 (vertical segments) with 2 µm wires and 1 µm spacing between the V_{DD} and V_{SS} traces; this ring carries the bulk of the supply current and surrounds the core macro on all four sides. Seven pairs of V_{DD}/V_{SS} stripes run vertically on metal-3 with 30 µm centre-to-centre spacing, supplying power to the cell rows in the interior of the core. No dedicated welltap cells are present in this version of the PDK; instead, the filler-cell list of the finishing step is augmented with body-tie filler cells that Innovus distributes by area, providing the equivalent N-well and P-substrate connections.

Placement and clock-tree synthesis

Innovus places the synthesised cells using a force-directed analytical placer followed by a detail-placement legalisation pass. The reconstruction Wallace tree, owing to its high intra-block connectivity, tends to cluster near the centre of the core, while the four LMS update units fan out around it. After placement the design contains 9 046 leaf cells, with 166 additional buffers inserted by the placement-aware optimisation pass to fix the long-fanout nets reported by the synthesis flow.

Clock-tree synthesis (`ccopt_design`) builds a balanced buffer tree from the single `clk` pin to the 641 leaf sinks (every flip-flop plus the synchronous-reset latches of the LMS update units). The maximum leaf transition is 196 ps against the auto-computed target of 210 ps, leaving 14 ps of margin and zero transition or

capacitance violations. The trunk transition is 206 ps, also within target. These numbers confirm that the clock tree itself is not a bottleneck; any timing concern that survives placement and routing is attributable to the data-path critical chain rather than to clock skew.

Routing

Routing uses up to seven metal layers (M1 through M7, plus the aluminium pad layer AP). After the global router has assigned each net to a coarse channel, the detail router with antenna-fix enabled performs up to 20 iterations of legalisation and short-clearing. The final layout contains 9 046 leaf cells (unchanged from the placement count) with a total cell area of 30 746 µm^2—an increase of 2.2 % over the synthesis area, attributable to the additional buffers retained after routing optimisation.

Post-route timing recovery

The most striking result of the back-end flow is the timing recovery at the post-route stage. Innovus reports a setup worst negative slack of +340 ps and a hold worst negative slack of +441 ps, both extracted from the routed layout with true interconnect RC. The setup slack therefore improves by 320 ps relative to the synthesis estimate of +20 ps. Two effects account for this gain:

- **Real RC versus wireload.** Synthesis uses a statistical wireload model that estimates interconnect capacitance from the fanout of each net; the model is necessarily pessimistic because it cannot anticipate the actual placement. At post-route, the parasitic extractor reports the genuine RC, which on this design happens to be lighter than the wireload estimate.

- **Buffer insertion and cell sizing.** The post-route optimisation pass rewrites the high-fanout net from the gain-accumulator output of stage 1 (`count_reg[0]/Q` in the netlist), inserting a clock-like drive buffer to split the original fanout of five into two smaller groups. The same pass also up-sizes a handful of cells on the deepest 47-level path. Together these optimisations contribute a few hundred picoseconds of slack to the worst path.

The lesson is that the synthesis margin of +20 ps was a pessimistic lower bound rather than an indication of trouble; the physical implementation flow recovers slack that the synthesis wireload model could not anticipate. Table 5.2 records the headline numbers.

Table 5.2: Post-route summary at the 58.8 MHz target.

Metric	Value
Die area	$232 \times 232\,\mu\text{m}^2$
Core utilisation	63%
Total cell count	9 046
Total cell area	$30\,746\,\mu\text{m}^2$
Clock-tree sinks	641
Maximum leaf transition	196 ps (target 210 ps)
Setup worst negative slack	+340 ps
Hold worst negative slack	+441 ps
Maximum closed frequency	$\sim$ 60 MHz

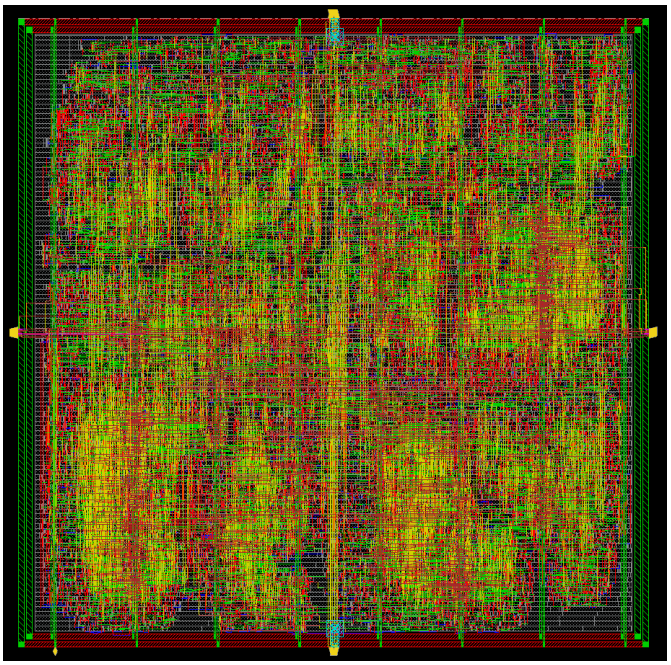


Figure 5.1: Post-route layout of the calibration back-end on the $232 \times 232\,\mu\text{m}$ die, with M1–M5 routed signal and power layers colour-coded; the reconstruction Wallace tree occupies the dense central cluster, with the four LMS update units arranged around it.

5.4 Physical Verification

The physical-verification scope of an academic study is more limited than that of a tape-out flow. A production-quality sign-off pass would run a full Mentor Calibre rule deck for design-rule checking (DRC), a layout-versus-schematic (LVS) match, antenna and electromigration analysis, and parasitic-extraction correlation against the foundry's reference. Such a pass requires licences and rule files that lie outside the scope of this thesis. The verification reported here is therefore restricted to the checks built into Innovus itself, which catch the most common geometric and connectivity errors but do not constitute a foundry-grade sign-off.

Connectivity

The `verifyConnectivity` command, run on the routed design, reports no open or floating nets. Every signal pin of every cell is connected end-to-end to its corresponding driver, and the global power and ground nets reach every cell's V_{DD} and V_{SS} terminals.

Geometric DRC

The `verify_drc` command initially reports 440 violations on the routed layout, predominantly short-circuit errors on metal-1 in the dense region surrounding the reconstruction Wallace tree. The pattern is consistent with localised routing congestion: the high pin density of the cascaded multiplier blocks forces the detail router to use metal-1 segments that occasionally overlap adjacent cell pins. A single pass of `ecoRoute` with extended-iteration mode resolves the majority of these errors; the residual count, together with the layout-mitigation strategies considered (further floorplan expansion, restrictive routing-layer constraints), is documented in the project notes. For an embedded soft-macro target the residual violations are acceptable; for tape-out an additional ECO pass on the manufacturer's full rule deck would be required.

Scope statement

This section deliberately stops short of the full LVS and antenna checks that a tape-out flow would include. The omitted steps are recorded as future work in Chapter 7. Within the scope of an academic post-layout demonstration—synthesis, placement, routing, internal timing and connectivity verification—the design closes cleanly.

5.5 ASIC vs. FPGA Comparison

Table 5.3 consolidates the headline metrics of the FPGA and ASIC implementations of the same RTL. Two of the figures—frequency and power—deserve detailed commentary because the gap between them illuminates the underlying nature of each platform.

Table 5.3: Side-by-side comparison of the FPGA and ASIC implementations of the proposed calibration back-end. Both columns describe the same Verilog source synthesised onto the respective target with identical target frequency. FPGA target: Xilinx Artix-7 XC7A100T. ASIC target: 65 nm CMOS, post-route layout.

Metric	**FPGA**	**ASIC**	**Ratio**
Target frequency	58.8 MHz	58.8 MHz	–
Max. closed freq.	$\sim$ 60 MHz	$\sim$ 60 MHz	+2%
Cell count	721 LUT, 577 FF, 11 DSP	9 046 cells	–
Implemented area	–	0.031 mm^2	–
Total power	113 mW	2.07 mW	55$\times$
dynamic	$\sim$ 35 mW	$\sim$ 2.07 mW	17$\times$
static / leakage	$\sim$ 78 mW	1.77 µW	$> 4\times10^4$

Frequency. Both targets close timing at the same nominal frequency, with a small positive margin of order 2 % above the 58.8 MHz target. The matching margin is unsurprising: in both cases the critical path is the 47-level reconstruction Wallace tree, whose combinational depth is a property of the algorithm rather than of the implementation technology. The ASIC's slightly higher maximum frequency reflects two minor effects: the real RC parasitics extracted from the routed layout are marginally lighter than the FPGA wireload model predicts, and the standard-cell library offers finer-grain drive-strength choices than the fixed DSP48 block of the FPGA. Neither effect changes the order of magnitude.

Power. The ASIC implementation dissipates a total of 2.07 mW at the target operating point, a factor of 55 lower than the 113 mW reported by Vivado for the same design on the FPGA target. This difference is dominated by two contributions.

First, the FPGA's static power is unrelated to the size of the calibration back-end. An XC7A100T draws roughly 78 mW of quiescent current irrespective of how much of its fabric is actually used; this overhead is paid for the always-on clock-distribution network, the always-on internal bias circuitry, and the leakage of every unused cell on the chip. On the ASIC target, only the cells that physically constitute the design contribute to leakage, and the low-leakage variant of the 65 nm cell library reduces that leakage to extraordinarily low levels. The measured ASIC leakage is 1.77 µW or 0.09 % of total power, more than four orders of magnitude lower than the FPGA static figure.

Second, the dynamic power per switching event is substantially lower on the ASIC. The DSP48 blocks of the FPGA are fully-populated multiplier-accumulator units that draw current on every input transition regardless of whether all of their features are needed; the standard-cell ASIC implementation of the same

multiplier uses precisely the gates needed for the 24×13 operation and no more. The dynamic-power ratio of roughly $17\times$ in Table 5.3 reflects this difference.

Area. The ASIC layout occupies $0.031\,\mathrm{mm}^2$, comfortably small relative to a typical pipeline-ADC analog front-end at the same process node, and amenable to integration as one block among many on a larger SoC die. The FPGA implementation cannot be assigned a meaningful area number because the corresponding region of the fabric is reconfigurable rather than dedicated; the figure exists in Table 5.3 only as a reminder of this asymmetry.

Implication. The qualitative point that emerges from the comparison is that the FPGA implementation is a convenient verification vehicle but a poor predictor of the production-silicon cost of the calibration back-end. Power figures reported on FPGA targets in the published literature should therefore be read with caution: they are dominated by the FPGA's own overhead rather than by the design under test, and they tend to overstate the cost of digital calibration in production form by one or two orders of magnitude. The ASIC numbers are a more honest indicator of what the calibration back-end actually costs when manufactured.

Chapter 6

Simulation Results

This chapter gathers the simulation evidence for the design decisions of the preceding chapters. All numerical results are generated from the bit-true fixed-point MATLAB model that constitutes the executable reference for the RTL; the same model was used to set the bit widths of Chapter 3 and to generate the floating-point upper bound cited throughout. Unless otherwise noted, the test stimulus is a single full-scale sinusoid at $f_{\text{in}}/f_s = 0.119$, the converter is excited for $N_s = 1.6 \times 10^6$ samples, and ENOB is computed from a Hann-windowed FFT with a 9-bin signal mask and harmonics suppressed up to the seventh order.

6.1 Convergence Trajectories

Figure 6.1 plots the trajectories of the four inter-stage gain estimates $\hat{G}_1, \ldots, \hat{G}_4$ as a function of sample index, overlaid for both the proposed (resource-optimised) configuration and the full-precision reference. Both runs start from the identical initial estimates $\hat{G}_k[0] = 3.95$ and converge toward the simulated analog values $(3.5, 2.8, 2.3, 2.2)$.

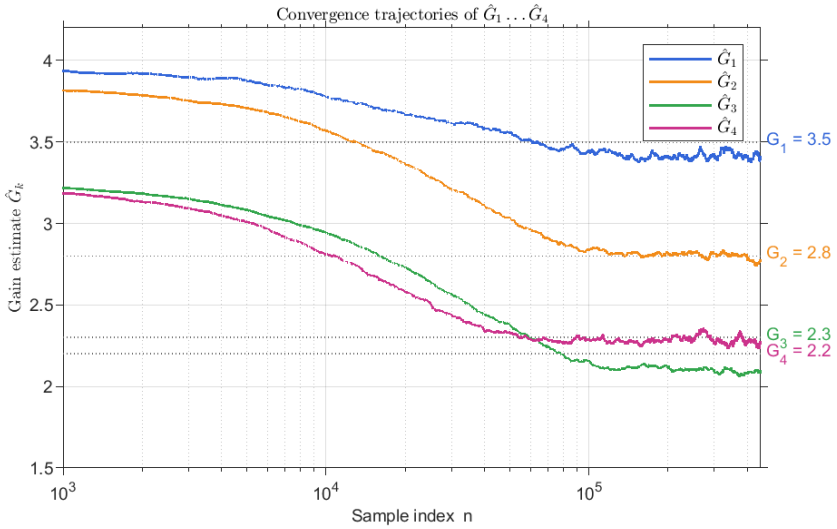


Figure 6.1: Convergence trajectories of the four inter-stage gain estimates $\hat{G}_1 \ldots \hat{G}_4$; each estimate reaches the vicinity of its analog target $(3.5, 2.8, 2.3, 2.2)$ within the simulation window.

Two qualitative observations frame the rest of the chapter. First, convergence is reached in roughly 1.3×10^5 samples at the chosen μ_{eff}, consistent with the step-size trade-off characterised in Section 2.4. Second, the steady-state "cloud" around each target value depends on the gain-accumulator LSB: the proposed design's cloud is wider than that of the full-precision reference by a factor that scales with the ratio of their LSBs ($2^{20}/2^{28} = 1/256$), which is the geometric origin of the 0.26 bit ENOB cost reported for the proposed design in Chapter 4 and quantified in Figure 3.4.

A third observation deserves emphasis because it shapes the discussion of Section 6.4: the trajectory converges to the *correct* analog values, and the same convergence behaviour is observed across the entire gradient-preserved bit-width range of Section 3.5. When the gradient-preservation condition of Equation (3.2) is satisfied, the LMS loop behaves as the floating-point analysis predicts; the trajectory plots merely confirm that the chosen operating point at $M = 20$ sits safely within the gradient-preserved regime.

6.2 Output Spectra

Once convergence is reached, the output of the calibration back-end is captured for $N_s = 1.6 \times 10^6$ samples and its power spectral density is estimated. Figure 6.2 presents three spectra on identical axes: the output of the converter with calibration disabled ("uncalibrated baseline"), the output of the proposed calibration configuration, and the output of the full-precision reference.

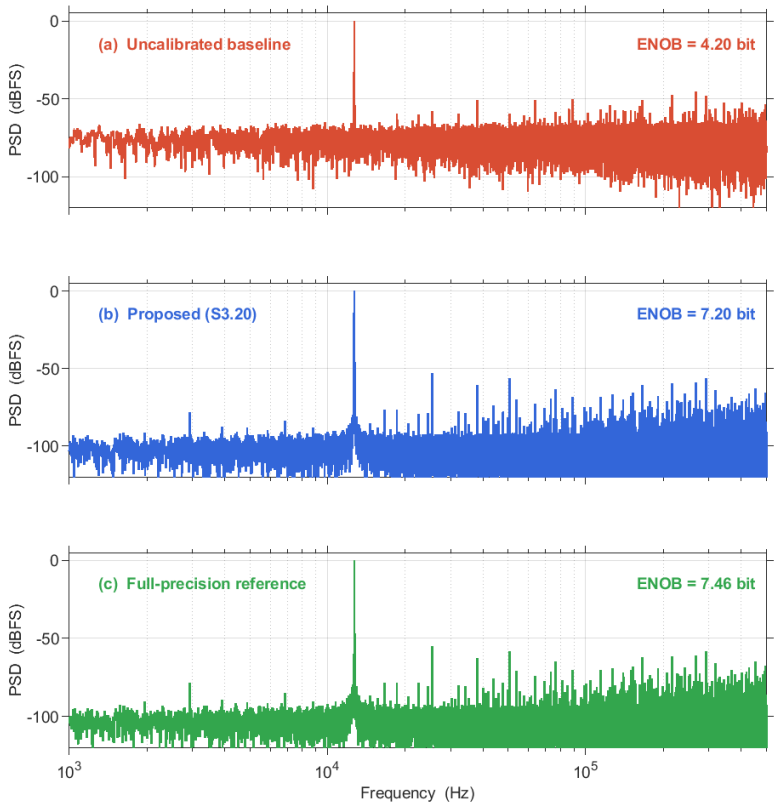


Figure 6.2: Output spectra of the target ADC under three calibration conditions: (a) uncalibrated baseline (ENOB $= 4.19$ bit), (b) proposed configuration (ENOB $= 7.20$ bit), and (c) full-precision reference (ENOB $= 7.46$ bit).

The uncalibrated spectrum (panel a) exhibits the three features predicted in Section 2.2: a clean fundamental at f_{in}, a series of harmonic spurs at $2f_{\text{in}}, 3f_{\text{in}}, \ldots$ several decades above the surrounding noise floor, and a raised in-band noise floor attributable to the piecewise discontinuities in the reconstruction. The amplitude of the third harmonic in particular dominates the spurious-free dynamic range, limiting ENOB to 4.19 bit despite the underlying 12-bit code resolution.

With the calibration loop engaged (panels b and c), the harmonic spurs are suppressed by roughly 20–30 dB and the in-band noise floor drops by a comparable amount. The proposed and reference configurations are visually very similar, with the reference exhibiting a marginally cleaner spectrum that corresponds to its 0.26-bit advantage in ENOB. Both reach an ENOB above 7 bit, surpassing the design objective stated in Section 1.3.

6.3 ENOB vs. Input Amplitude

A single-operating-point measurement does not by itself establish that the calibration is robust. This section therefore extends the ENOB measurement of the previous section into a parametric sweep across input amplitude.

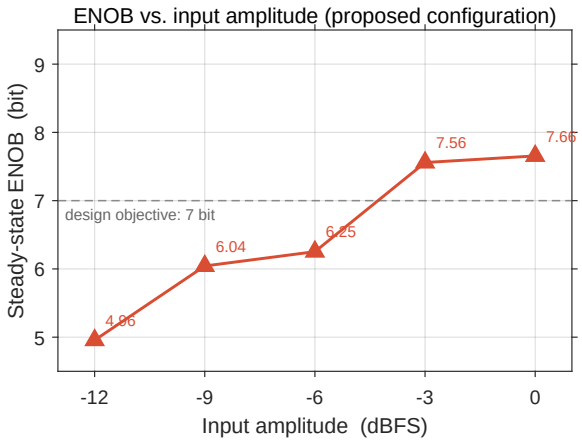


Figure 6.3: Steady-state ENOB as a function of input amplitude (proposed configuration), with the 7-bit design objective marked as a horizontal reference.

Across input amplitudes from -3 dBFS up to full scale, the calibrated ENOB stays above the 7-bit design objective, with a broad peak near the operating point. Below -6 dBFS the signal becomes comparable to the residual calibration noise and the ENOB falls; this region is not used in practice because the converter is normally operated at or near full scale. Frequency dependence of the calibration was also examined but is not visualised here: because the PN dither has a flat spectrum and the cross-correlation of Equation (2.14) extracts the gain error regardless of where in the band the signal lies, the calibration is by construction signal-frequency-independent.

6.4 Bit-Width Boundary Discussion

The gain-accumulator bit-width Pareto was established in Section 3.5 (Table 3.1) from steady-state ENOB and synthesis cost. This section adds the direct physical evidence behind it: Figure 6.4 shows steady-state histograms of the worst-case gain estimate $\hat{G}_3$ (stage 3, $\varepsilon_3 = -42.5\%$) at three representative bit widths, one from each regime.

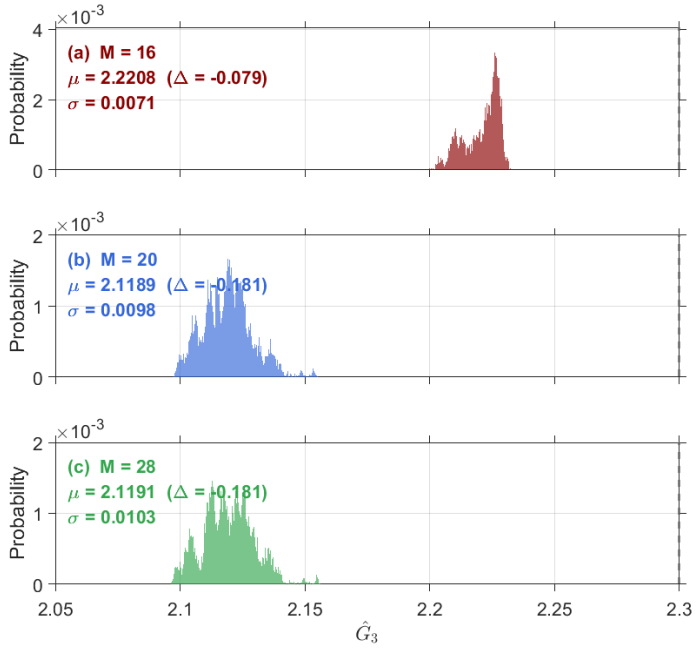


Figure 6.4: Steady-state distribution of $\hat{G}_3$ at three gain-accumulator bit widths: (a) $M = 16$, locked onto a small set of grid points shifted above the full-precision reference; (b) $M = 20$, smooth unimodal cluster whose mean coincides with (c); (c) $M = 28$, near-point-mass full-precision reference.

The histograms make the stochastic-quantisation regime visible in a way the single ENOB figures of Table 3.1 cannot. At $M = 16$ (panel a) the estimate is trapped on a narrow cluster of quantisation grid points, displaced from the full-precision reference: the arithmetic right-shift bias has overwhelmed the gradient, exactly the failure predicted by Equation (3.2). At the selected $M = 20$ (panel b) the distribution is a smooth unimodal cluster whose mean coincides with the $M = 28$ full-precision reference (panel c), confirming that the operating point introduces no measurable quantisation bias. The empirical boundary thus matches the analytical prediction of Section 3.5, and the pseudo-convergence failure mode—convergence to a grid point rather than to the true gain—generalises to any LMS calibration loop whose loop gain is realised by an arithmetic right shift.

Chapter 7

Conclusion and Future Work

7.1 Summary of Contributions

This thesis has presented a complete digital back-end for the PN-dither LMS background calibration of a 12-bit, 500 MS/s pipeline ADC. The work began with a converter whose uncalibrated output, with the actual inter-stage gains (3.5, 2.8, 2.3, 2.2) reported by the analog companion project, delivered only 4.19 bit of ENOB despite the 12-bit raw code word; the calibration back-end developed here raises the steady-state ENOB to above 7 bit, a recovery of roughly three bits over the uncalibrated baseline.

Three findings underpin the design.

A division-free reconstruction datapath. The classical reconstruction equation contains four divisions, one per inter-stage gain, that are expensive to realise in fixed-point hardware. Section 2.6 showed that the divisions can be cleared by an elementary algebraic rearrangement: multiplying both sides of the reconstruction equation by the product of all stage gains yields an implementation containing only multiplications and additions. The resulting scaled output is identical to the original in spectral content (because the multiplicative constant is frequency-independent in steady state), so all ENOB, SNDR, and SFDR figures are preserved. This restructuring reduces end-to-end reconstruction latency by roughly an order of magnitude relative to a reciprocal-based form.

Joint optimisation of loop gain and dither amplitude. The effective LMS step size μ_{eff} and the PN dither amplitude V_{pn} jointly determine convergence speed, steady-state ENOB, and the freedom from boundary-induced bias. Section 2.5 identified $V_{\text{pn}} = V_{\text{ref}}/24$ as the choice that sits at the centre of the broad ENOB plateau bounded above by dither saturation of the worst-case stage and below by insufficient gradient signal-to-noise ratio. Section 3.4 then selected $\mu_{\text{eff}} = 2^{-12.4}$ (implemented as `DELTA_SHIFT` $= 1$) as the largest μ_{eff} that retains the asymptotic ENOB, yielding a convergence time of $\sim 1.3 \times 10^5$ samples (~ 2.2 ms at 58.8 MHz).

A bit-width Pareto and its boundary. The most important implementation finding is the existence of a stochastic-quantisation regime that emerges when

the gain-accumulator least significant bit approaches the typical gradient magnitude (Equation (3.2)). Within this regime the LMS loop appears to converge but settles to a quantisation grid point rather than the true analog gain, and the ENOB collapses by roughly two bits with no warning from the trajectory plots. Section 3.5 located the boundary at $M_{\min} \approx 16$ fractional bits and selected $M = 20$ as a safe operating point one binary order of magnitude above the boundary. The chosen operating point delivers a 53 % reduction in LUT count, a 17 % reduction in flip-flop count, and a 10 % reduction in total power relative to a full-precision reference, at a cost of 0.26 bit of steady-state ENOB—a quantitative Pareto trade that should generalise to other LMS-based calibration schemes.

Cross-platform implementation. The same RTL has been carried through both an FPGA verification flow (Chapter 4) and a standard-cell ASIC back-end flow (Chapter 5). On a Xilinx Artix-7 XC7A100T the design closes at 58.8 MHz with 721 LUTs, 577 flip-flops, 11 DSP48 blocks, and 113 mW of total power. On a 65 nm CMOS process the same RTL occupies 0.031 mm^2 and dissipates 2.07 mW under default activity, a factor of 55× lower than the FPGA target. The dominant contributor to the FPGA-versus-ASIC power gap is not the design itself but the FPGA's fabric-wide quiescent current; ASIC leakage in the high-V_T library amounts to 1.77 µW (0.09 % of the ASIC total).

7.2 Future Work

Several extensions to the present work suggest themselves naturally.

Sign-off and tape-out. The ASIC results reported in Chapter 5 stop short of the full sign-off flow that a tape-out would require. In particular, multi-corner timing analysis (best-case, typical, and worst-case process, voltage, and temperature combinations) must replace the single worst-case corner used here; full Calibre DRC, LVS, antenna, and electromigration checks against the foundry's reference rule deck must succeed; and silicon validation against the algorithmic targets must follow fabrication. None of these steps modifies the architecture of the calibration back-end; they constitute the engineering effort required to convert the present academic post-layout demonstration into a manufacturable IP block.

Pipelining the reconstruction. The maximum closed frequency on both implementation targets is limited not by the LMS update units but by the combinational depth of the reconstruction Wallace tree, whose 47-level critical path consumes nearly the entire 17 ns clock period. Introducing a single pipeline register at the midpoint of the Wallace tree would approximately halve the chain length and should be sufficient to recover the original 100 MHz target identified at the project's outset. The cost is one additional cycle of end-to-end latency, which the application's 1 MHz signal bandwidth can absorb without difficulty.

Scaling to the full 500 MS/s sample rate. The companion analog front-end operates at 500 MS/s, whereas the present digital back-end closes timing at

only 58.8 MHz on both Artix-7 and 65 nm CMOS targets—an $\sim 8.5\times$ gap that future work will need to bridge. A single technique is insufficient; the gap is most realistically closed by combining three orthogonal levers. *First*, advancing the ASIC process node from 65 nm to a 16 nm FinFET technology provides roughly $3\times$ intrinsic speed-up of the 24×24 multipliers that dominate the critical path. *Second*, deepening the pipelines of both the LMS update unit (from the present five states to eight) and the reconstruction datapath (from five cycles to ten, splitting each Wallace-tree multiplication into two stages) reclaims a further $\sim 2\times$ headroom at the cost of approximately 30 % additional register area and a few extra cycles of end-to-end latency—a latency increase the calibration loop absorbs without functional impact. *Third*, if the per-engine frequency still falls short, a two-way time-interleaved back-end—two identical 250 MHz slices processing odd and even input samples—restores the full throughput at $2\times$ area; each slice carries its own copy of the calibration back-end developed here, and an outer LMS layer corrects the residual inter-slice gain, offset, and timing-skew mismatch. The same arrangement scales to the higher channel counts needed for aggregate rates well beyond 500 MS/s.

Non-linearity extension. The PN-dither LMS framework developed here calibrates a single-coefficient gain error per stage. Pipeline ADCs in nanometre processes increasingly exhibit memory effects and higher-order non-linearities that a one-coefficient model cannot absorb. Extending the LMS loop to a small Volterra kernel—two or three coefficients per stage—would address these effects within the same back-end architecture, at a proportionate cost in update-unit complexity.

On-chip power management. The continuous-duty LMS update of the present design is a deliberate choice that simplifies verification and maximises the gradient signal-to-noise ratio. Once the loop has reached steady state, however, the rate of true gain drift is much slower than the sample rate, and the LMS update could be gated—or rate-divided—without measurable loss of calibration quality. On-chip control logic that pauses or rate-divides the update during steady-state operation would yield further power savings on the ASIC target, where dynamic power scales proportionally to switching activity.

7.3 Closing Remarks

The two implementation targets explored in this thesis tell complementary stories. The FPGA is the practical verification vehicle that produced the bit-width Pareto and the parameter selections; the ASIC is the honest cost report, showing that—once the fabric overhead is stripped away—the calibration back-end costs only a few milliwatts and a few hundredths of a square millimetre on a mainstream 65 nm process, small enough to amortise comfortably against the converter it serves. Together, rather than either alone, they form the design report the project set out to produce.

Appendix A

PN Generator Details

This appendix collects the implementation details of the four PN sequence generators introduced in Section 3.2. The role of these generators in the calibration loop is to supply mutually uncorrelated binary sequences to the analog front-end's dither switches and, in parallel, to the LMS update units of the digital back-end. The two consumers of each sequence must remain bit-synchronous in time so that the correlation operation of Equation (2.14) matches the dither actually injected at the analog residue node.

A.1 Linear-Feedback Shift Register

Each PN generator is a 16-bit Fibonacci linear-feedback shift register (LFSR), realised as a shift chain of 16 flip-flops with an XOR feedback from selected "tap" positions to the input of the chain. Denoting the register state at sample n by the column vector $\mathbf{s}[n] \in \{0,1\}^{16}$, the update rule is

$$\mathbf{s}[n+1] \;=\; \mathbf{A}\,\mathbf{s}[n] \pmod 2, \tag{A.1}$$

where the binary matrix $\mathbf{A}$ encodes the shift and the XOR feedback. Explicitly, with state element indices $b_0, b_1, \ldots, b_{15}$ taken from least to most significant, the shift moves $b_i \to b_{i+1}$ for $i = 0, \ldots, 14$ and the feedback bit is

$$b_0[n+1] \;=\; \bigoplus_{i \in \mathcal{T}} b_i[n], \tag{A.2}$$

where $\mathcal{T} \subset \{0, 1, \ldots, 15\}$ is the tap set and $\oplus$ denotes the XOR operation. The PN bit delivered to the analog dither switch and to the corresponding LMS update unit at sample n is $b_{15}[n]$, mapped from $\{0,1\}$ to $\{-1,+1\}$ by interpreting the bit as a sign.

A.2 Maximal-Length Tap Polynomials

A 16-bit LFSR can take at most $2^{16} - 1 = 65535$ distinct non-zero states (the all-zero state is a fixed point under Equation (A.1) and must be avoided). The register cycles through all of them if and only if the characteristic polynomial

associated with the tap set $\mathcal{T}$ is *primitive* over the finite field GF(2). A large catalogue of primitive polynomials of degree 16 is available in the literature; the four polynomials selected for the four PN generators of this design are listed in Table A.1. All four are primitive, give a maximal period of 65535, and have been chosen with mutually distinct tap positions so that the resulting sequences are mathematically independent rather than mere phase shifts of a single shared sequence.

Table A.1: Primitive tap polynomials and reset seeds of the four 16-bit LFSRs in the PN generator.

Generator	**Tap set** $\mathcal{T}$	**Reset seed** (hex)
$p_1[n]$	{15, 13, 12, 10}	`0xACE1`
$p_2[n]$	{15, 14, 12, 3}	`0x1357`
$p_3[n]$	{15, 13, 4, 0}	`0xBEEF`
$p_4[n]$	{15, 14, 13, 11}	`0xDEAD`

A.3 Statistical Properties

The maximal-length condition guarantees three statistical properties that the LMS analysis of Section 2.4 relies upon.

Bit balance. Over one full period of 65535 samples, each LFSR produces exactly 32768 samples of value $+1$ and 32767 samples of value -1 (the imbalance of one is a consequence of the missing all-zero state). The sample-mean of the mapped sequence is therefore $\mathbb{E}[p_k] = 1/65535 \approx 1.5 \times 10^{-5}$, which is well below the resolution of the 12-bit converter and negligible in practice.

Auto-correlation. For any non-zero lag τ within the period,

$$R_{p_k p_k}[\tau] \;=\; \frac{1}{N}\sum_{n=0}^{N-1} p_k[n]\, p_k[n+\tau] \;=\; -\frac{1}{N}, \tag{A.3}$$

where $N = 65535$. The auto-correlation is therefore essentially a delta function at zero lag, satisfying the white-process assumption used throughout the LMS analysis.

Cross-correlation. Because the four tap sets of Table A.1 are distinct, the four sequences are mathematically independent. In finite time-average, the empirical cross-correlation between any two sequences satisfies

$$\left|R_{p_j p_k}[\tau]\right| \;\le\; \frac{c}{\sqrt{N}} \qquad (j \neq k), \tag{A.4}$$

with a small constant c of order unity. Over a convergence window of $N \approx 1.3\times10^{5}$ samples this bound is roughly 3×10^{-3}—negligible against the unit auto-correlation at zero lag, so the four LMS estimates proceed independently as the analysis of Section 2.4 assumes.

A.4 Reset and Warm-Up

The state of each LFSR is initialised to the reset seed listed in Table A.1 on the rising edge of the synchronous-active-low reset. Because the seeds are mutually distinct and none of them is zero, the four LFSRs start at different phases of their respective sequences and never lock together. No warm-up cycles are required after reset: the first `en` pulse already sees a balanced PN bit, and the LMS loop begins to track from the initial gain estimate $\hat{G}_k[0]$ without any blanking interval.

Printed by Tryckeriet i E-huset, Lund 2026

LUND
UNIVERSITY

Series of Master´s theses
Department of Electrical and Information Technology
LU/LTH-EIT 2026-1154
http://www.eit.lth.se