ENGG4080 Project

Cole Mayke, 1127413 (Group 16)
ENGG*4080 Micro and Nano Electronics
Professor: Dr. Stefano Gregori

December 5, 2025

1 Background

The target system is a mixed-signal front end for a smart-watch ECG application. Gizmonic Inc. specifies a single-chip solution in TSMC 65 nm CMOS, powered from a 1 V battery with 10 Ω internal resistance and a common analog ground. The watch must interface to an ECG conditioning circuit with 10 kΩ output resistance, 0.5 V common-mode level, and a maximum signal amplitude of 20 mV. The analog front end therefore must resolve small differential signals around mid-supply while consuming little power and die area.

On the analog side the chip comprises three main blocks: a three-channel analog multiplexer, a programmable-gain amplifier (PGA) with integer gains from 1 to 4, and a 6-bit successive-approximation ADC. Digital control and signal processing are handled by a separate microprocessor block. The mixed-signal interface must operate over the 0 to 70 °C commercial temperature range and for ±5% battery voltage variation, so the analog blocks were designed with some gain and headroom margin rather than being tuned only at nominal conditions.

2 Overview

The overall architecture follows the block diagram in the project brief. The ECG sensor and two auxiliary inputs are routed to a 3:1 CMOS analog multiplexer. The selected channel drives the PGA input. The PGA implements gain-of-{1,2,3,4} modes using a switched-capacitor configuration around a single-ended operational amplifier. Gain selection is performed by a 4-bit programmable capacitor array (PCA) controlled by digital bits B1 to B4 from the microcontroller. The PGA output is then low-pass filtered and presented to the sampling network of the 6-bit SAR ADC.

At the transistor level, the design is constrained by the 1 V supply and by the minimum geometries in the specification. All MOSFET widths are integer multiples of 350 nm and all lengths are multiples of 60 nm. Minimum devices are used where possible to reduce input capacitance and power consumption; larger devices are reserved for the op-amp gain stages and for switches that must pass near-rail voltages with low on-resistance.

Non-overlapping clocks ϕ_1 and ϕ_2 with 5% rise/fall times are used throughout the switched-capacitor network so that critical nodes are never shorted between phases. The same clock pair is reused in the SAR ADC charge-redistribution DAC to keep the timing architecture simple.

3 Design Constraints

Key design constraints from the project handout are:

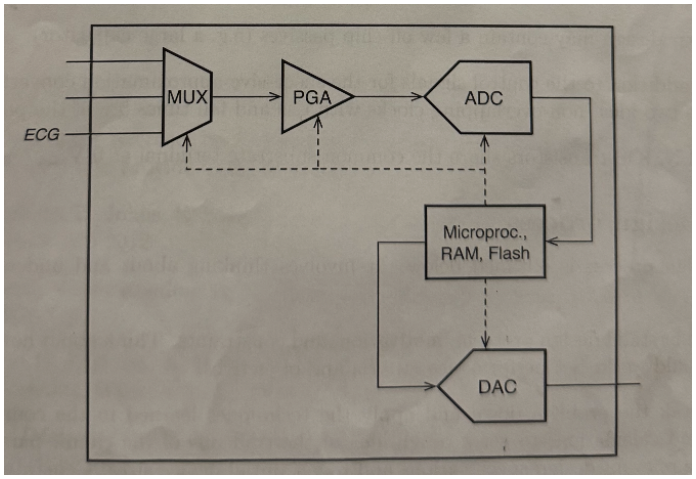

Figure 1: Top-level block diagram of the ECG front end: 3:1 MUX, switched-capacitor PGA, and 6-bit SAR ADC.

- **Process and supply.** TSMC 65 nm CMOS, $V_{\mathrm{DD}} = 1\,\mathrm{V}$, single 0 V substrate for all NMOS devices.
- **Device sizing.** MOSFET widths are integer multiples of 350 nm, lengths are integer multiples of 60 nm; the smallest device is $W/L = 350\,\mathrm{nm}/60\,\mathrm{nm}$.
- **Capacitors.** Minimum usable capacitor is 5 fF. Feedback and gain-setting capacitors must use exact integer ratios.
- **PGA performance.** The op-amp inside the PGA must provide at least 40 dB open-loop gain so that closed-loop integer gains of 1 to 4 are met with acceptable error for a 6-bit converter.
- **ADC performance.** The SAR ADC must achieve at least 10^5 samples/s with 6-bit resolution over a 1 V full-scale range, implying an LSB of 15.625 mV.
- **Clocks.** In addition to the SAR control signals, only two ideal non-overlapping clocks with rise/fall times of 5% of the period may be used in the analog sections.
- **System metrics.** Power dissipation and die area should be minimized while meeting the above specifications over supply and temperature.

4 Design Plan

The design flow followed the general process from the assignment:

1. Translate the system-level specification into block-level targets (gain, bandwidth, input range, output swing, and load for each circuit) and sketch candidate architectures for the MUX, PGA and ADC.
2. For each block, perform hand calculations for device sizes and component ratios, using small-signal models to check that gain and bandwidth are achievable under the 1 V supply.

3. Implement minimum schematic versions in Cadence Virtuoso with ideal sources and loads. Verify basic behavior (MUX selection, PGA gain stepping, op-amp biasing) before adding non-idealities.

4. Refine device sizes based on DC operating points, AC gain plots, and transient simulations. At this stage some choices were simplified (for example, replacing dense transmission-gate networks inside the PCA with ideal switches) to make system behavior clearer.

5. Integrate the verified blocks into a top-level testbench that applies an ECG-like input, steps through all gain codes, and drives a load representative of the SAR ADC input.

5 Analog Multiplexer (MUX)

5.1 Architecture and Sizing

The multiplexer selects one of three analog channels under digital control from the microcontroller. A 3:1 MUX is implemented as two cascaded 2:1 cells. Each 2:1 cell uses a pair of complementary CMOS transmission gates driven by a local select signal and its complement. This provides rail-to-rail swing and lower on-resistance than a simple NMOS pass transistor, which would otherwise clip the ECG signal at 1 V supply.

Each transmission gate consists of an NMOS and a PMOS device with $W = 350\,\text{nm}$, $L = 60\,\text{nm}$ (the minimum) so that the MUX input capacitance is small compared with the $10\,\text{k}\Omega$ source resistor. The inverter that generates the complementary select signal uses $W = 200\,\text{nm}$, $L = 60\,\text{nm}$ for both NMOS and PMOS. The input is fully analog and speed is not critical, so symmetric sizing is sufficient.

5.2 Control Strategy

The three input channels, labelled `d0`, `d1`, and `d2` in the testbench, are combined using one front MUX that selects between `d0` and `d1`, followed by a second MUX that selects between the front-MUX output and `d2`. The microcontroller provides a 2-bit select code (S_1, S_0). Here S_0 drives the first MUX and S_1 selects either the first-stage output or the third channel. This reuses the same 2:1 cell and avoids a separate 3:8 decoder.

6 Transmission Gate

Transmission gates are used inside each 2:1 MUX cell and as generic analog switches in the PCA and PGA sampling network. The `tgate` schematic implements a bidirectional transmission gate between nodes `v1` and `v2`. The gate consists of parallel NMOS and PMOS devices, each $W = 350\,\text{nm}$, $L = 60\,\text{nm}$, which gives a reasonable trade-off between on-resistance and gate capacitance at 1 V.

A local inverter generates the complementary control signal $\overline{\texttt{sel}}$ from `sel`. The PMOS gate is driven by $\overline{\texttt{sel}}$ and the NMOS gate by `sel`. When `sel`=1 the gate conducts with low resistance over most of the common-mode range; when `sel`=0 both devices are off and leakage between `v1` and `v2` is negligible in simulation.

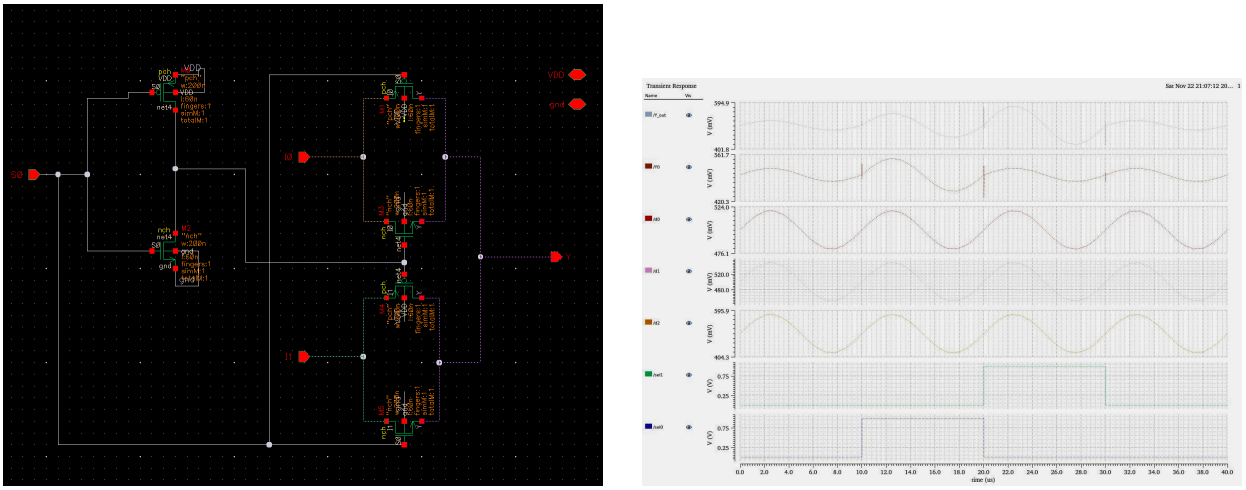

Figure 2: MUX schematic (left) and transient response showing correct selection of the three input channels (right).

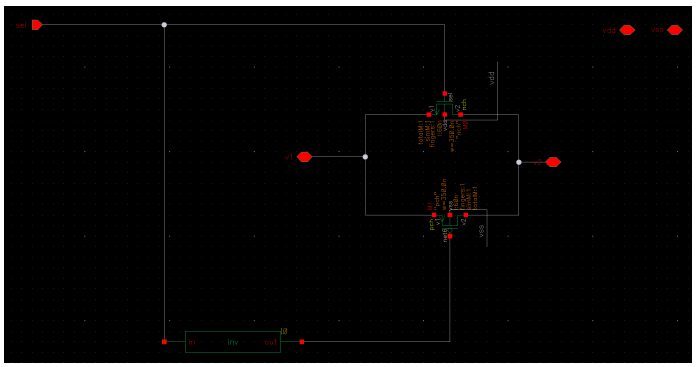

Figure 3: CMOS transmission gate used in the MUX and other blocks.

7 Inverter

A simple CMOS inverter is used as the basic logic element in the design, mainly to generate complementary select signals for the MUX and transmission gates. The `inverter` schematic follows the textbook topology: a PMOS between V_{DD} and the output, and an NMOS between the output and ground, with gates tied together.

To keep input capacitance small, both devices use $W = 200\,\text{nm}$, $L = 60\,\text{nm}$. With these dimensions the inverter switches well within the 100 kHz range of the analog test signals used in the MUX testbench and provides clean rail-to-rail logic levels for the transmission-gate inputs.

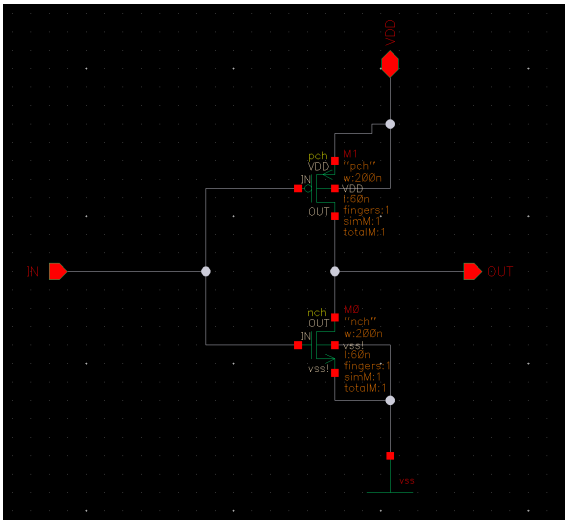

Figure 4: CMOS inverter used to generate complementary control signals.

8 Operational Amplifier

8.1 Topology Selection

The PGA requires an op-amp with at least 40 dB open-loop gain while driving the switched-capacitor network and the SAR ADC input. A two-stage CMOS operational amplifier was used. The first stage is a PMOS differential pair with NMOS active loads; the second stage is a common-source NMOS amplifier that provides additional gain and drives `VOUT`.

The PMOS input pair improves the common-mode input range around 0.5 V, which is set by the ECG sensor and PGA sampling network. The NMOS devices in the second stage benefit from higher mobility and help recover gain that would otherwise be lost at 1 V and minimum length.

8.2 Biasing and Device Sizing

Bias currents are generated by a current-mirror branch (M1 and current source I0), which establishes a reference bias on node `net2`. This bias is mirrored into M0 to feed the differential-pair load devices. M1 and M4, which act as load devices for the output branch, use $W = 700\,\text{nm}$, $L = 120\,\text{nm}$ to provide sufficient output resistance without excessive headroom loss.

The differential-pair PMOS transistors M2 and M3 use $W = 4.2\,\mu\text{m}$, $L = 160\,\text{nm}$ to increase input gm and first-stage gain while keeping input-referred noise acceptable for a 6-bit system. The

NMOS tail devices M7 and M6 are minimum width ($W = 350\,\text{nm}$, $L = 160\,\text{nm}$), since they mainly set tail current and common-mode gain.

The output NMOS M5 ($W = 700\,\text{nm}$, $L = 360\,\text{nm}$) and PMOS load M4 form the second stage. These devices are larger to reduce output resistance modulation when driving the load, which includes the PGA feedback network and an emulated ADC sampling capacitor. The bias current source is $10\,\mu\text{A}$, so the static power of the op-amp is on the order of a few tens of μW.

8.3 Frequency Compensation and AC Gain

Miller compensation uses a series R_0–C_0 network between the intermediate node and the output node. In the final schematic $R_0 = 15\,\text{k}\Omega$ and $C_0 = 300\,\text{fF}$. The capacitor introduces a dominant pole at the first-stage output and a high-frequency zero that is controlled by the series resistor. The values were chosen by sweeping the AC response until a reasonable phase margin was obtained for an expected load of roughly 20 to 50 pF, without overly limiting the closed-loop bandwidth of the PGA.

The simulated open-loop AC magnitude response with a 1 V AC input shows roughly $100\,\text{V/V}$ gain at low frequencies. Converting to decibels gives

$$A_{\text{OL,dB}} = 20\log_{10}(100) \approx 40\ \text{dB},$$

which meets the project requirement.

8.4 Transient Response

To check large-signal behavior, the op-amp was simulated in unity-gain with a 0.5 V common-mode and a small sinusoidal excitation. The transient output tracks the input without slewing and stays within the linear output swing range for the expected ECG-sized signals, confirming that the chosen bias current can drive the load capacitance at the intended operating frequency.

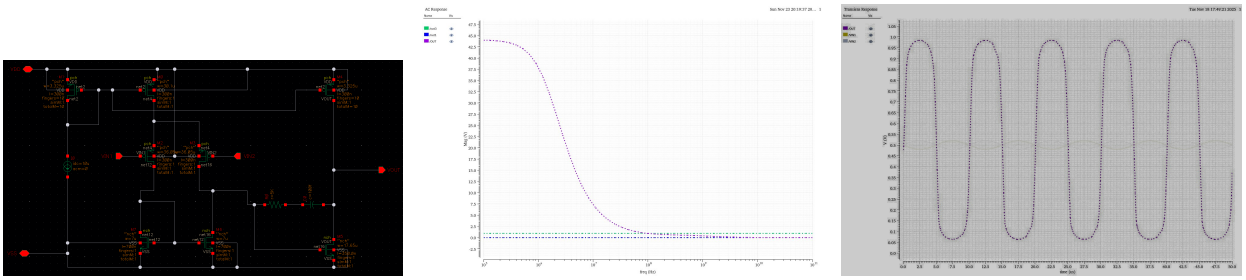

Figure 5: Two-stage CMOS op-amp: schematic (left), open-loop AC response (middle), and unity-gain transient response (right).

9 Programmable Capacitor Array (PCA)

The programmable capacitor array enables integer gain steps in the PGA and implements binary-weighted charge redistribution for the 6-bit SAR converter. The `pca` schematic uses four binary-weighted capacitors of 8 pF, 4 pF, 2 pF, and 1 pF (C_0 to C_3). The right plates of all capacitors are tied together to form the PGA summing node and ADC input node, while their left plates are individually connected either to the input signal or to ground via switches.

Each branch is controlled by a digital bit B1 to B4, which drives a local switch (`W0` to `W3`). When a bit is high, the corresponding capacitor connects to the input node; when low, the capacitor is discharged to VSS. Because integer ratios are used, the binary weighting directly produces the desired gain steps when combined with the sampling capacitor on the op-amp input.

9.1 Ideal Switch Implementation

An earlier version of the PCA, PGA sampling network, and ADC DAC used full transistor-level transmission gates for every switch. That is physically realistic but created issues at 1 V:

- With small overdrive, transmission-gate resistance varies strongly with common-mode level, making settling time signal-dependent.
- Detailed transistor models produced visible glitches on the summing node each time a switch changed state, masking the ideal charge-redistribution behavior.
- The dense network often led to "shorted output" warnings and convergence problems in Spectre.

The final design keeps transistor-level transmission gates in the front-end MUX but replaces most internal PCA, PGA, and ADC switches with ideal switch elements. This preserves the correct charge-redistribution equations and makes the impact of op-amp gain and capacitor ratios easier to see.

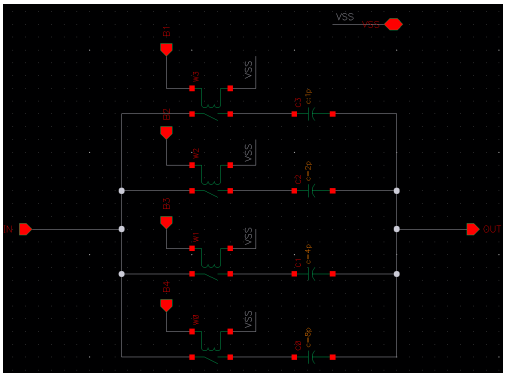

Figure 6: Programmable capacitor array (binary-weighted 1 to 8 pF capacitors) shared by the PGA and SAR ADC.

10 Programmable-Gain Amplifier (PGA)

10.1 Switched-Capacitor Topology

The PGA combines the op-amp and PCA into a non-inverting switched-capacitor configuration with gains of 1, 2, 3, and 4. The `pga` schematic shows a sampling capacitor at the op-amp input that alternately samples the ECG signal and is then connected in series with selected PCA capacitors during the amplification phase. The effective closed-loop gain is set by the ratio between the total PCA capacitance connected to the input and the fixed feedback/sampling capacitor.

In addition to the main binary-weighted capacitors, a second set of small "glitch" capacitors is connected to the same summing node. These have values of 100 fF, 200 fF, 400 fF, and 800 fF

and are driven by the same digital bits as the main branches. When the gain code changes, these smaller capacitors momentarily absorb or supply charge and spread the transition in time. They barely change the dc gain because they are much smaller than the main feedback capacitor, but in transient they reduce spikes at the op-amp input and make the gain steps more monotonic.

Because the main capacitors still share a common unit size and exact integer ratios, the ideal gain expression $G = 1 + C_{\mathrm{PCA}}/C_{\mathrm{S}}$ is preserved. The glitch capacitors mainly affect waveform shape during switching rather than the final settled value of each gain level.

10.2 Clocking and Control

The PGA relies on strictly non-overlapping clocks. In the final `pga_tb` schematic, CLK1 and CLK2 come from VPULSE sources with period $T_{\mathrm{clk}} = 40\,\mu\mathrm{s}$. Rise and fall times are about 5% of the period, which leaves flat time in each phase for the capacitors to settle.

Within this 40 µs frame, gain control bits B1 to B4 are also driven by VPULSE sources. Each bit is active for roughly 10 µs and the four bits are offset so the digital word steps through the four gain codes in sequence. Over a 40 µs window the PGA output spends about 10 µs at each gain from 1 to 4. Bit transitions occur only during the short interval when both CLK1 and CLK2 are low, which keeps charge injection into the summing node small and lets the glitch capacitors do their job.

10.3 Measured Gain for Codes 1 to 4

The PGA was exercised with a 0.5 V common-mode and a 20 mV peak sinusoidal input (40 mV peak-to-peak). Figure 7 shows the output waveform as the digital gain word steps through the four codes. From the waveform viewer, the approximate peak-to-peak output amplitudes are:

Gain code	$V_{\mathrm{out,pp}}$ [mV]	Measured gain $G = V_{\mathrm{out,pp}}/V_{\mathrm{in,pp}}$
1x	≈ 40	≈ 1.0
2x	≈ 80	≈ 2.0
3x	≈ 120	≈ 3.0
4x	≈ 160	≈ 4.0

The gains are within about 5 to 10% of the ideal integers; the residual error is mainly due to finite op-amp open-loop gain and the output RC loading. This is more than sufficient for a 6-bit ADC with 15.6 mV LSB.

11 Analog to Digital Converter

11.1 Architecture Overview

The ADC is a 6-bit successive-approximation (SAR) converter using a charge-redistribution capacitive DAC. Its input range is 0 to 1 V, matching the PGA output swing. The topology is standard: a sample-and-hold and capacitive DAC connect either the input or the reference to a summing node; a clocked comparator resolves each bit; and a digital SAR state machine drives the switches and records the 6-bit word.

The same binary-weighted PCA used in the PGA is reused as the main DAC in the SAR ADC. During the sample phase all bottom plates are connected to the input, so the top plate tracks the PGA output. During the conversion phase the bottom plates are successively reconnected either to

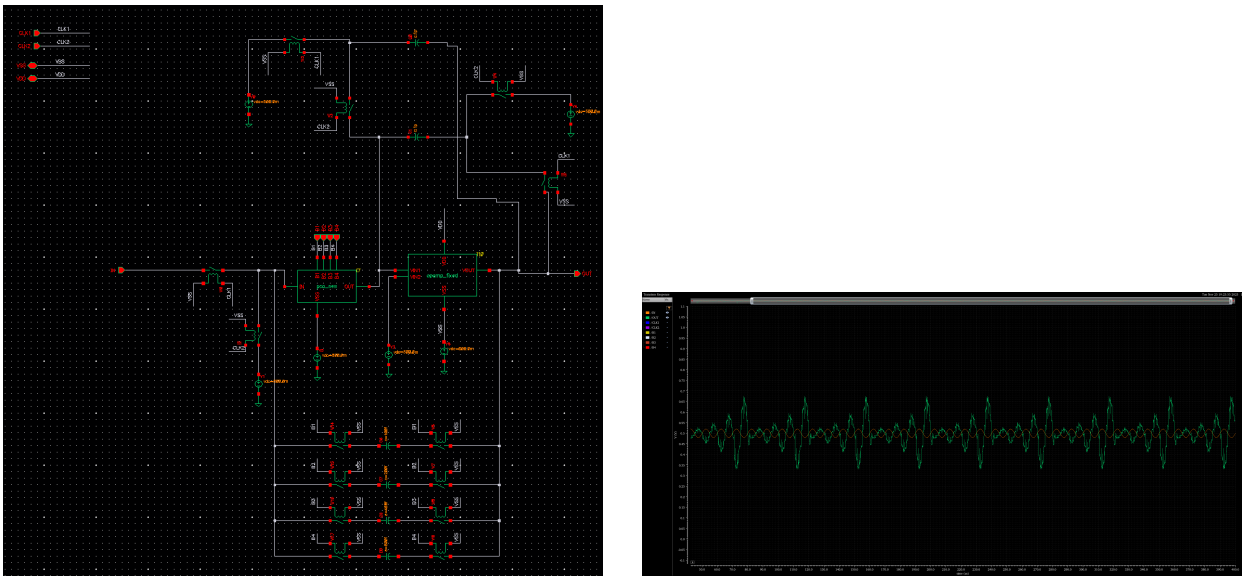

Figure 7: PGA schematic (left) and transient response for gain codes 1 to 4 (right).

ground or to $V_{\mathrm{REF}} = 1\,\mathrm{V}$ under SAR control, implementing the binary search on the summing-node voltage.

11.2 Capacitive DAC and Sampling Network

On the left of the ADC schematic, the binary-weighted capacitors are driven by switches labelled $S[5:0]$, SB, SA, and SO_prime, which match the Verilog-A SAR control pins. The top plates are shorted to form `comp_in`, feeding the comparator input.

During sampling, SA and SB close so that the top node connects to the PGA output and bottom plates are tied to the input and common-mode voltages. When the clock enters the conversion phase, SA opens and SB reconfigures the array into the DAC, while the SAR logic starts toggling $S[k]$ from the MSB down.

Ideal switches are used for the DAC for the same reason as in the PCA. This keeps the DAC transfer linear and leaves non-idealities dominated by op-amp gain and comparator offset instead of pass-device limitations.

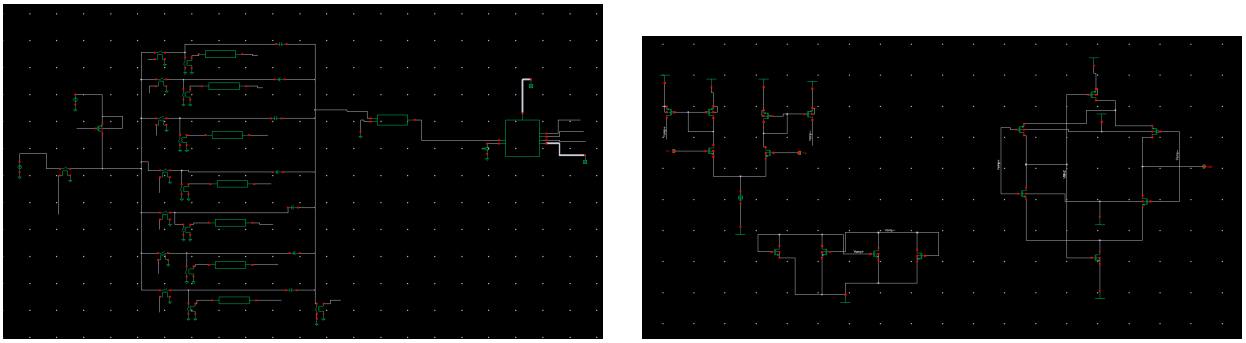

Figure 8: SAR ADC top-level schematic (left) and clocked comparator (right).

11.3 Comparator Design

The comparator consists of a small preamplifier followed by a regenerative latch. The preamplifier is an op-amp with differential inputs `Vcmp+` and `Vcmp-` connected to the DAC summing node and a common-mode reference. It amplifies the difference by roughly 20 to 30 dB while keeping input-referred noise low.

The second stage is a cross-coupled latch built from back-to-back inverters gated by the SAR clock. During reset the latch nodes are precharged to mid-rail. When the clock asserts, the latch is released and the small preamp output is regenerated into a full rail-to-rail logic level sent to the SAR block as the comparison result.

11.4 SAR Verilog-A Control Logic

The SAR state machine is implemented in Verilog-A. The module interface is `(clk, comp_in, SA, SB, SO_prime, S[5:0], dout[5:0])`; internal electrical nodes drive the capacitor switches through vectors `s_vals` and `dout_vals`. The code cycles through:

1. **Idle / reset.** All DAC bottom plates connected to the input (sample mode), outputs at zero, counters reset.
2. **Sample.** For `SAMPLE_CYCLES` clocks, SA and SB hold sampling mode so the top node tracks the PGA output.
3. **Conversion.** Starting from the MSB, each capacitor is tentatively connected to V_{REF} or ground, the comparator output is read, the SAR register is updated, and the next bit is tried.

Using Verilog-A instead of an ideal code converter lets realistic timing between DAC settling and comparator decision be modelled while staying technology independent. The full code is listed in the Appendix.

11.5 ADC Waveform Example

Figure 9 shows a typical ADC transient response. The PGA output is overlaid with the sampled comparator input and the resulting 6-bit SAR output word. As the input ramps, the staircase shows uniform 15.6 mV steps, which confirms that the charge-redistribution DAC and SAR logic operate correctly.

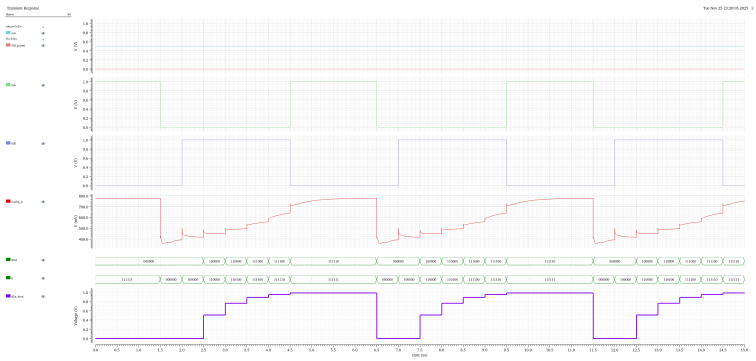

Figure 9: Example SAR ADC transient: comparator input and 6-bit output code versus time.

12 Performance Analysis

12.1 Power Consumption

The dominant static power consumer is the op-amp in the PGA (also used in the comparator preamplifier). With a 10 μA bias at 1 V, its quiescent power is about 10 to 20 μW. The MUX, PCA, and DAC use minimum-sized devices and ideal switches, so their static power is negligible. Dynamic power is dominated by charging and discharging the DAC and load capacitors at the SAR clock:

$$P_{\text{dyn}} \approx C_{\text{eff}} V_{\text{DD}}^2 f_{\text{clk}}.$$

For an effective switched capacitance of a few tens of picofarads and a 1 MHz clock, this is only a few tens of μW. Overall, the analog front end (MUX, PGA, comparator, DAC) is expected to consume well below 100 μW, which is compatible with smartwatch battery budgets.

12.2 Temperature and Operating Range

Correct operation is required from 0 to 70 °C. Over this range, mobility decreases and threshold voltage shifts with temperature. The op-amp bias was chosen so that even at high temperature the input pair stays in strong inversion and maintains enough transconductance for 40 dB gain. Because the PGA and ADC are switched-capacitor, gain accuracy depends mainly on capacitor ratios, which are relatively insensitive to temperature.

At low temperature the increased mobility improves gm and speeds up settling. The 0.5 V common-mode, 1 V supply, and chosen device sizes keep transistors in saturation during normal operation, even under ±5% supply variation. Op-amp transient simulations show the output between roughly 0.05 and 0.95 V, leaving margin to the rails for process and temperature shifts.

13 Top Level

13.1 Integration Strategy

The verified MUX, PGA (including op amp and PCA), and ADC model were instantiated in a top-level schematic that matches the project block diagram. A dedicated testbench provides:

- the ECG source model (10 kΩ source resistance, 0.5 V common-mode, up to 20 mV amplitude);
- the two non-overlapping clocks ϕ_1 and ϕ_2 with 5% edge times;
- the digital gain word `B[3:0]` stepping through the gains;
- SAR control logic and comparator clock for the ADC.

At this level, op-amp input connections were checked to ensure that the inverting node always receives feedback from the capacitor network and the non-inverting node stays at the common-mode reference. Mis-wiring at this stage produced flat outputs and had to be corrected before meaningful gain and conversion measurements could be taken.

13.2 Top-Level Results

Transient simulations were run over many clock periods and gain codes. After an initial start-up interval, the MUX, PGA, and ADC chain behaved as expected: the PGA output amplitude scaled approximately 1 to 4 times with the gain word, and the ADC codes covered a reasonable portion of the 6-bit range without clipping.

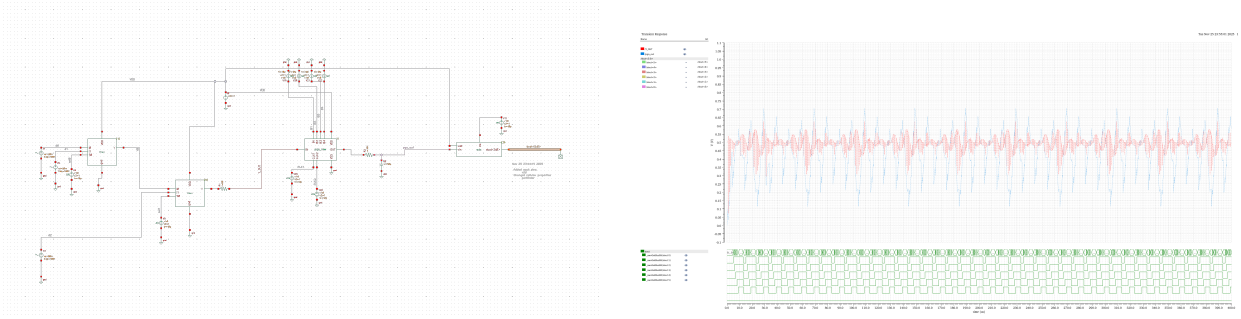

Figure 10: Top-level schematic (left) and simulation showing ECG input, PGA output for different gains, and ADC codes (right).

14 Discussion

The design meets the main functional requirements in the project outline. The MUX selects among three inputs, the PGA achieves roughly integer gains of 1 to 4 for the specified ECG signal, and the SAR ADC provides adequate dynamic range relative to the 15.6 mV LSB.

The main challenges were practical rather than architectural. Examples include:

- resolving shorted-output warnings from dense transmission-gate networks;
- enforcing non-overlap between ϕ_1 and ϕ_2 and aligning digital gain changes with the both-low window;
- obtaining sufficient op-amp gain and bandwidth with quantized device sizes at 1 V;
- recognizing and ignoring the first-cycle start-up transient of the switched-capacitor network once steady-state behavior was confirmed.

Power, area, and operating range were considered qualitatively during sizing, but a complete optimization and full corner analysis would require additional simulation time.

15 Appendix

15.1 Verilog-A SAR Listing

```
‘include "constants.vams"
‘include "disciplines.vams"

module SAR_renesas(clk, comp_in, SA, SB, SO_prime, S, dout);

input clk, comp_in;
output SA, SB, SO_prime;
output [5:0] S, dout;

electrical clk, comp_in;
electrical SA, SB, SO_prime;
electrical S[5:0], dout[5:0];

parameter real vth = 0.41;
parameter real vdd = 1.0;
parameter real trise = 100p;
parameter real tfall = 100p;
parameter integer SAMPLE_CYCLES = 3;

integer state;
integer bit_index;
integer sar_reg[5:0];
integer prev_sar_reg[5:0];
```

```
integer pending_bit_index;
integer phase_counter;

real sa_val, sb_val;
real s_vals[5:0];
real dout_vals[5:0];

analog begin

  @(initial_step) begin
    state = 0;
    bit_index = 5;
    pending_bit_index = -1;
    phase_counter = 0;

    sar_reg[0]=0; sar_reg[1]=0; sar_reg[2]=0;
    sar_reg[3]=0; sar_reg[4]=0; sar_reg[5]=0;

    sa_val = vdd; // feedback closed
    sb_val = 0; // bus at VIN

    s_vals[0]=0; s_vals[1]=0; s_vals[2]=0;
    s_vals[3]=0; s_vals[4]=0; s_vals[5]=0;

    dout_vals[0]=0; dout_vals[1]=0; dout_vals[2]=0;
    dout_vals[3]=0; dout_vals[4]=0; dout_vals[5]=0;
  end

  @(cross(V(clk) - vth, +1)) begin
    case (state)

      0: begin
        // SAMPLE
        sa_val = vdd;
        sb_val = 0;

        s_vals[0]=0; s_vals[1]=0; s_vals[2]=0;
        s_vals[3]=0; s_vals[4]=0; s_vals[5]=0;

        phase_counter = phase_counter + 1;
        if (phase_counter >= SAMPLE_CYCLES) begin
          state = 1;
          phase_counter = 0;
        end
      end

      1: begin
        // PREPARE: reset SAR and move bus to VREF
        sa_val = 0;
        sb_val = vdd;

        prev_sar_reg[0]=sar_reg[0]; prev_sar_reg[1]=sar_reg[1];
        prev_sar_reg[2]=sar_reg[2]; prev_sar_reg[3]=sar_reg[3];
        prev_sar_reg[4]=sar_reg[4]; prev_sar_reg[5]=sar_reg[5];

        sar_reg[0]=0; sar_reg[1]=0; sar_reg[2]=0;
        sar_reg[3]=0; sar_reg[4]=0; sar_reg[5]=0;

        bit_index = 5;
        pending_bit_index = -1;
        state = 2;
      end

      2: begin
        // CONVERSION
        sa_val = 0;
        sb_val = vdd;

        // STEP 1: latch comparator result for pending bit
        if (pending_bit_index >= 0) begin
          if (V(comp_in) > vth)
            sar_reg[pending_bit_index] = 1;
          else
            sar_reg[pending_bit_index] = 0;
        end

        // STEP 2: update DAC switch positions
        s_vals[5] = (sar_reg[5] == 1 || bit_index == 5) ? vdd : 0;
        s_vals[4] = (sar_reg[4] == 1 || bit_index == 4) ? vdd : 0;
        s_vals[3] = (sar_reg[3] == 1 || bit_index == 3) ? vdd : 0;
        s_vals[2] = (sar_reg[2] == 1 || bit_index == 2) ? vdd : 0;
        s_vals[1] = (sar_reg[1] == 1 || bit_index == 1) ? vdd : 0;
        s_vals[0] = (sar_reg[0] == 1 || bit_index == 0) ? vdd : 0;

        // STEP 3: prepare next bit
        pending_bit_index = bit_index;
        bit_index = bit_index - 1;

        // STEP 4: check if conversion finished
        if (bit_index < 0) begin
          state = 0;
          bit_index = 5;
```

```
        pending_bit_index = -1;

        sa_val = vdd;
        sb_val = 0;

        s_vals[0]=0; s_vals[1]=0; s_vals[2]=0;
        s_vals[3]=0; s_vals[4]=0; s_vals[5]=0;
      end
    end

   endcase
  end

  // Drive analog outputs with smoothed transitions
  V(SA) <+ transition(sa_val, 0, trise, tfall);
  V(SB) <+ transition(sb_val, 0, trise, tfall);
  V(SO_prime) <+ transition(0, 0, trise, tfall);

  V(S[5]) <+ transition(s_vals[5], 0, trise, tfall);
  V(S[4]) <+ transition(s_vals[4], 0, trise, tfall);
  V(S[3]) <+ transition(s_vals[3], 0, trise, tfall);
  V(S[2]) <+ transition(s_vals[2], 0, trise, tfall);
  V(S[1]) <+ transition(s_vals[1], 0, trise, tfall);
  V(S[0]) <+ transition(s_vals[0], 0, trise, tfall);

  dout_vals[5] = (sar_reg[5] ? vdd : 0);
  dout_vals[4] = (sar_reg[4] ? vdd : 0);
  dout_vals[3] = (sar_reg[3] ? vdd : 0);
  dout_vals[2] = (sar_reg[2] ? vdd : 0);
  dout_vals[1] = (sar_reg[1] ? vdd : 0);
  dout_vals[0] = (sar_reg[0] ? vdd : 0);

  V(dout[5]) <+ transition(dout_vals[5], 0, trise, tfall);
  V(dout[4]) <+ transition(dout_vals[4], 0, trise, tfall);
  V(dout[3]) <+ transition(dout_vals[3], 0, trise, tfall);
  V(dout[2]) <+ transition(dout_vals[2], 0, trise, tfall);
  V(dout[1]) <+ transition(dout_vals[1], 0, trise, tfall);
  V(dout[0]) <+ transition(dout_vals[0], 0, trise, tfall);

end

endmodule
```

Listing 1: Verilog-A code for the 6-bit SAR control logic.