Nyquist-Rate A/D Converters

劉 尙 大 敎授

전 자 공 학 부

Kyungpook National University

Nyquist-Rate CMOS A/D Converters

- ❏ Nyquist-rate : oversampling ADC.
- ❏ A/D converter architectures.

Low-to-medium speed High accuracy	Medium speed Medium accuracy	High speed Low-to-medium accuracy
Integrating Oversampling	Successive approximation Algorithmic	Flash Two-step Interpolating Folding Pipelined Time-interleaved

Integrating A/D Converters

- A popular approach for high accuracy and low speed.
- Very low offset and gain errors.
- Highly linear.
- Small amount of circuitry in implementation.
- Usage in voltage or current meters.
- Dual-slope integrating converters.

Dual-Slope Integrating Converters

- Phase I: For a *fixed interval* $T_1 = 2^N T_{clk}$, switch S_1 is connected to $-V_{in}$ such that V_x is given by

$$V_x(t) = \int_0^t \frac{V_{in}}{R_1 C_1} dt = \frac{V_{in}}{R_1 C_1} t$$

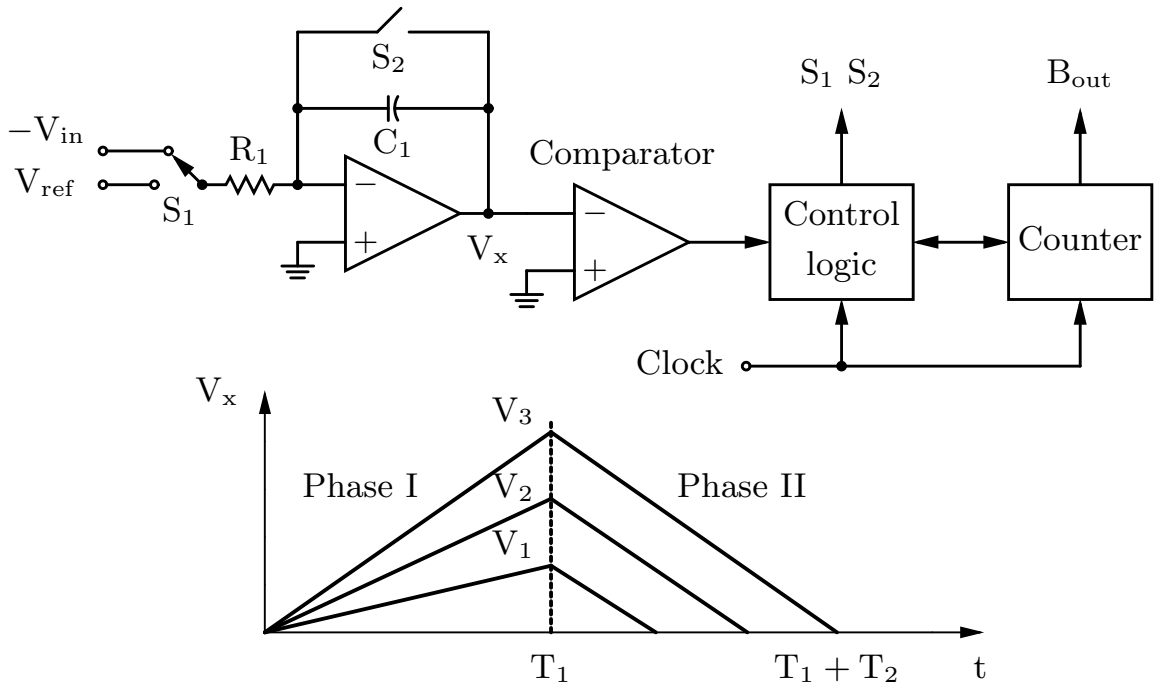
- Phase II: For a variable interval T_2, switch S_1 is connected to V_{ref} such that V_x is decaying in a *constant slope*.

$$V_x(t) = V_x(T_1) - \int_{T_1}^t \frac{V_{ref}}{R_1 C_1} dt = \frac{V_{in}}{R_1 C_1} T_1 - \frac{V_{ref}}{R_1 C_1}(t - T_1)$$

If V_x equals to 0 when $t = T_1 + T_2$, T_2 is related to T_1 by

$$\frac{T_2}{T_1} = \frac{V_{in}}{V_{ref}} = \frac{B_{out} 2^N T_{clk}}{2^N T_{clk}} = b_1 2^{-1} + b_2 2^{-2} + \cdots + b_N 2^{-N}$$

❏ A dual slope A/D converter.



❏ Design of R_1 and C_1 for a large peak value of V_x to reduce noise.

❏ A dual-slope converter does not suffer from gain error, but it can have an offset error ⇒ quad-slope: first ground, next signal.

❏ Low speed: 2^{N+1} clock cycles in the worst-case conversion.

❏ T_1 = an integer multiple of 1/(60 Hz) or (16.67 ms) to attenuate the power line noise.

$$V_{in} = V_{in(ideal)} + V_{in(60\ Hz)} = V_{in(ideal)} + A\sin(120\pi t + \phi)$$

$$V_x(t) = \int_0^{T_1} \frac{V_{in}}{R_1 C_1} dx = \int_0^{T_1} \frac{V_{in(ideal)}}{R_1 C_1} dx + \int_0^{T_1} \frac{V_{in(60\ Hz)}}{R_1 C_1} dx$$

❏ Effective input filtering for integrating-then-reset behavior ⇒ $h(t)$ = a square pulse of length T_1.

$$|H(f)| = \left| \frac{\sin(\pi f T_1)}{\pi f T_1} \right| = -20 \text{ dB/decade for side lobes}$$

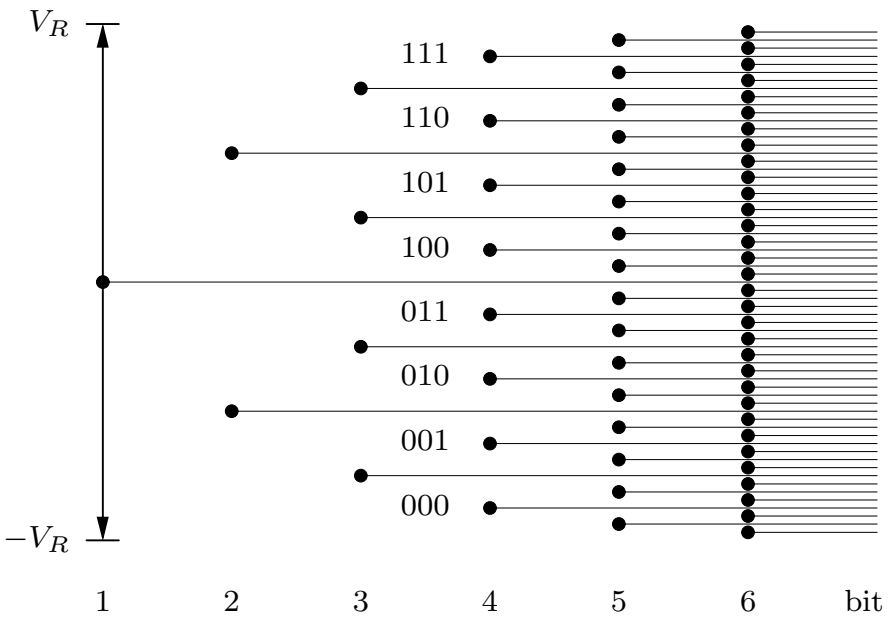
Successive-Approximation Converters

- ❏ One of the most popular approaches.
- ❏ Quick conversion time and moderate circuit complexity.
- ❏ Operation by binary search algorithm.
- ❏ Determination of the closest digital word to match an input signal.
- ❏ DAC-based successive-approximation ADC.
- ❏ Charge-redistribution ADC.
- ❏ Resistor-capacitor hybrid ADC.

Bipolar Successive-Approximation Algorithm

```
void SaAdc( n, b, vin, vref )
int   n,     /* resolution of ADC            */
      b[];   /* offset binary code of result */
float vin,   /* input voltage (± vref)       */
      vref;  /* reference voltage            */
{   int   i; float vda = 0.0; /* output voltage of D/A converter */
    for ( i = 0; i < n; i++ ) {
        if ( vin >= vda ) {
            b[i] = 1;
            vda  = vda + vref / pow( 2, i+1 );
        } else {
            b[i] = 0;
            vda  = vda - vref / pow( 2, i+1 );
        }   // equivalent statements
    }       // vda  = vda - vref / pow( 2, i   );  // b[i] -> 0
}           // vda  = vda + vref / pow( 2, i+1 );
```


Successive-Approximation Diagram

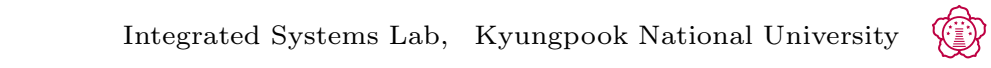


Unipolar Successive-Approximation Algorithm

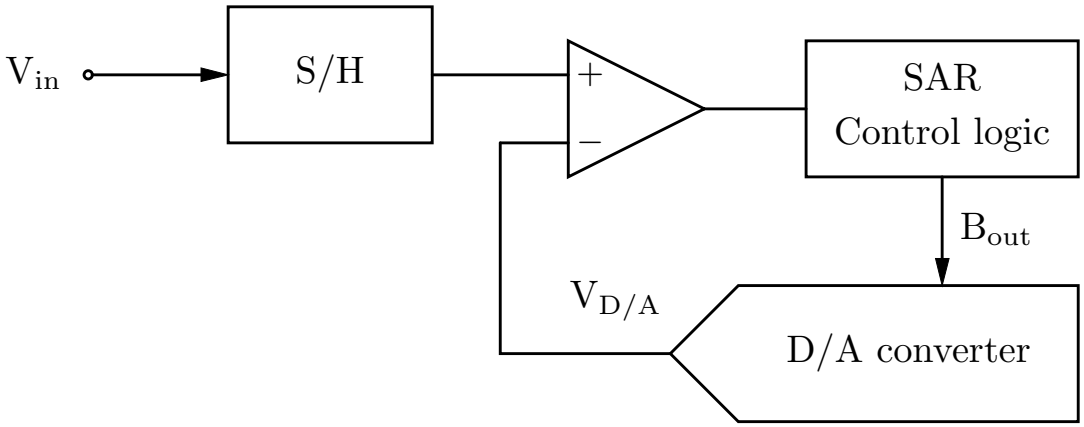
```
void UniSaAdc( n, b, vin, vref )
int   n,     /* resolution of ADC            */
      b[];   /* offset binary code of result */
float vin,   /* input voltage (< vref)       */
      vref;  /* reference voltage            */
{   int i; float vda = vref / 2; /* output voltage of D/A converter */
    for ( i = 0; i < n; i++ ) {
        if ( vin >= vda ) {
            b[i] = 1;
            vda  = vda + vref / pow( 2, i+2 );
        } else {
            b[i] = 0;
            vda  = vda - vref / pow( 2, i+2 );
        }   // equivalent statements
    }       // vda  = vda - vref / pow( 2, i+1 );  // b[i] -> 0
}           // vda  = vda + vref / pow( 2, i+2 );
```

DAC-Based Successive-Approximation ADC

- ❏ Unipolar input signal.
- ❏ S/H for unchangeable input signal during one-bit conversion.
- ❏ Comparator + successive-approximation register (SAR).
- ❏ Control logic + DAC.
- ❏ DAC determines the accuracy and speed of the ADC.
- ❏ N clock cycles to complete an N-bit conversion.



- ❏ A DAC-based successive-approximation A/D converter.



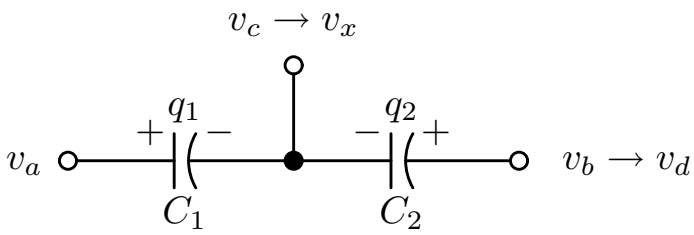
Charge Redistribution in Switched Capacitors

- Initial charges: $q_1 = C_1(v_a - v_c), \quad q_2 = C_2(v_b - v_c)$.
- Charge redistribution: $v_b \rightarrow v_d$.

$$q_1' = q_1 + \Delta q = C_1(v_a - v_x), \quad q_2' = q_2 - \Delta q = C_2(v_d - v_x)$$

- Charge conservation and voltage change: $q_1 + q_2 = q_1' + q_2'$.
  $(v_x \neq f(v_a),\ v_b = 0,\ v_c = \mathrm{V_{OS}} - \mathrm{V_{in}},\ v_d = \mathrm{V_{ref}},\ C_1 = C_2)$

$$v_x = v_c + \frac{C_2}{C_1 + C_2}(v_d - v_b) = \mathrm{V_{OS}} - \mathrm{V_{in}} + \frac{\mathrm{V_{ref}}}{2}$$

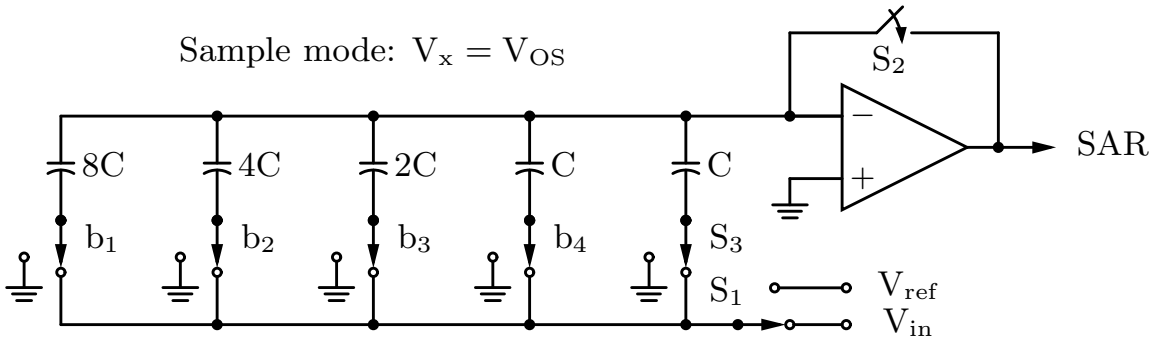


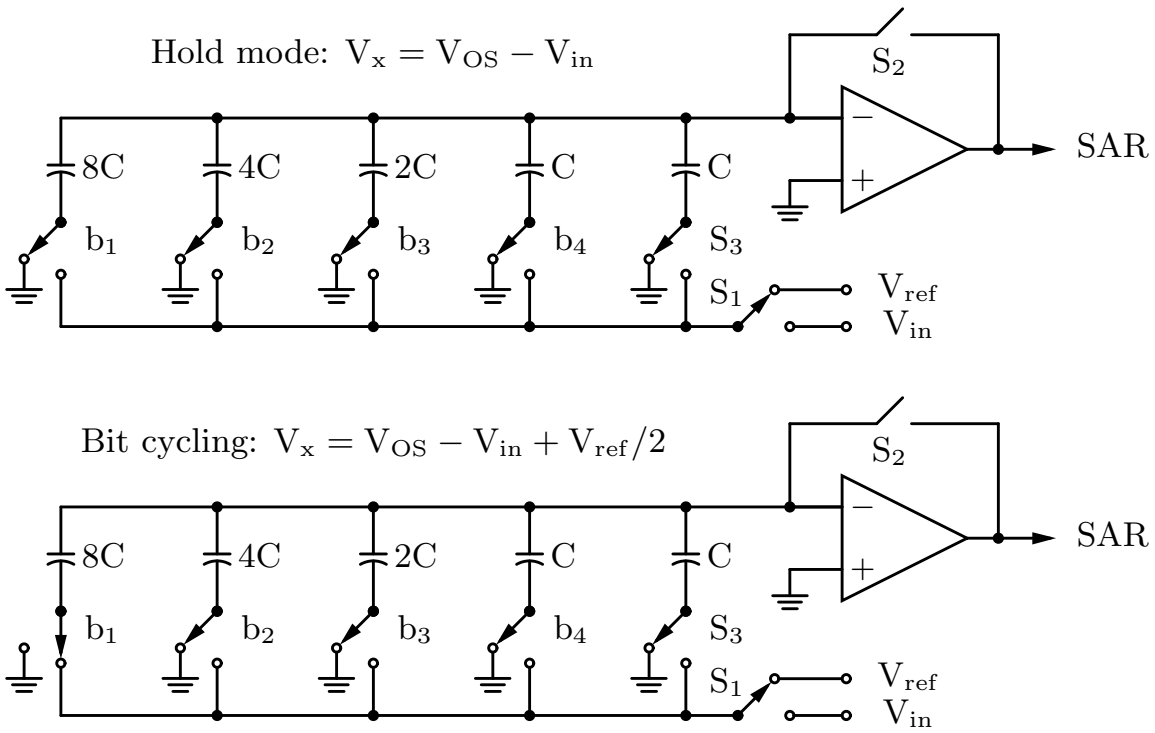
Unipolar Charge-Redistribution A/D Converter

- A single circuit: S/H + DAC + subtraction.
- Input signal: $0 \leq \mathrm{V_{in}} < \mathrm{V_{ref}}$.
- Error signal $\mathrm{V_x} \equiv -(\mathrm{V_{in}} - \mathrm{V_{D/A}})$: always compared to ground.
- *Sample mode*: All capacitors are charged to $\mathrm{V_{in}}$ while the comparator is being reset to its offset voltage.
- *Hold mode*: The comparator is taken out of reset by opening $\mathrm{S_2}$, and then all capacitors are switched to ground. Finally, $\mathrm{S_1}$ is switched so that $\mathrm{V_{ref}}$ can be applied to the capacitor array during bit cycling.
- *Bit cycling*: The largest capacitor 8C is switched to $\mathrm{V_{ref}}$. $\mathrm{V_x}$ now goes to $\mathrm{V_{OS}} - \mathrm{V_{in}} + \mathrm{V_{ref}}/2$. If $\mathrm{V_x} \leq \mathrm{V_{OS}}$ ($\mathrm{V_{in}} \geq \mathrm{V_{ref}}/2$), then this capacitor is left connected to $\mathrm{V_{ref}}$ and $\mathrm{b_1} = 1$.

Otherwise, this capacitor is reconnected to ground ($V_x = V_{OS} - V_{in}$) and $b_1 = 0$. This process is repeated for the smaller capacitors to finish the conversion → offset voltage cancellation.

- ❏ Parasitic capacitance at node x does not cause conversion errors.
- ❏ Bottom plates of capacitors should be connected to V_{ref} side.
- ❏ A 4-bit unipolar charge-redistribution ADC.





Bipolar Successive-Approximation by Divided Remainder

```
void DiSaAdc( n, b, vin, vref )
int   n,     /* resolution of ADC            */
      b[];   /* offset binary code of result */
float vin,   /* input voltage (± vref)       */
      vref;  /* reference voltage            */
{   int   i;
    float vda = - vin; /* divided remainder  */
    for ( i = 0; i < n; i++ ) {
        if ( vda <= 0.0 ) {
            b[i] = 1;
            vda  = vda + vref / pow( 2, i+1 );
        } else {
            b[i] = 0;
            vda  = vda - vref / pow( 2, i+1 );
        }   // equivalent statements
    }       // vda  = vda - vref / pow( 2, i   );  // b[i] -> 0
}           // vda  = vda + vref / pow( 2, i+1 );
```

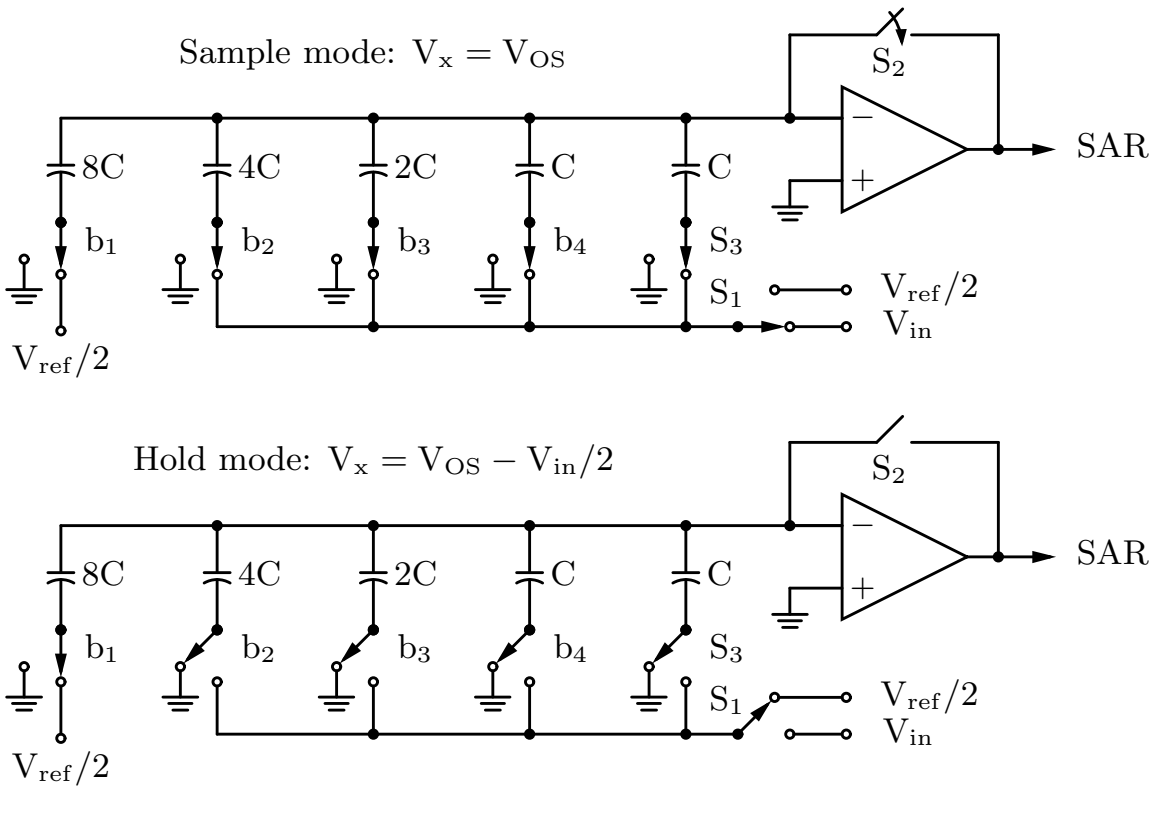

Signed Charge-Redistribution A/D Converter

- Input signal: $-V_{ref}/2 \leq V_{in} < V_{ref}/2$.
- Using only a single reference voltage.
- *Sample mode*: All capacitors, except for the largest capacitor, are charged to V_{in} while the comparator is being reset to its offset voltage. The largest capacitor is now connected to $V_{ref}/2$.
- *Hold mode*: The comparator is taken out of reset by opening S_2, and then all capacitors, except for the largest capacitor, are switched to ground. The sign of the input signal is determined by looking at the comparator output. Finally, S_1 is switched so that $V_{ref}/2$ can be applied to the capacitor array during bit cycling.

- *Bit cycling*: If $V_x \leq V_{OS}$, then V_{in} is positive and b_1 is set to 1, and conversion proceeds as in the unipolar case, starting with b_2.

  The largest capacitor 8C is switched to ground if V_{in} is negative. This is *level shift* operation causing V_x to become $V_{OS} - V_{in}/2 - V_{ref}/4$ (which is a negative value since $V_{in} \geq -V_{ref}/2$). And conversion proceeds as in the unipolar case, starting with b_2.

- Reduced S/N ratio by using $V_{in}/2$ in the conversion process.
- Any error in the MSB capacitor causes both an offset and a sign-dependent gain error due to level shift for $V_{in} < 0$.
- Conversion results expressed in offset binary code → digital recoding.
- A 4-bit signed charge-redistribution A/D converter.



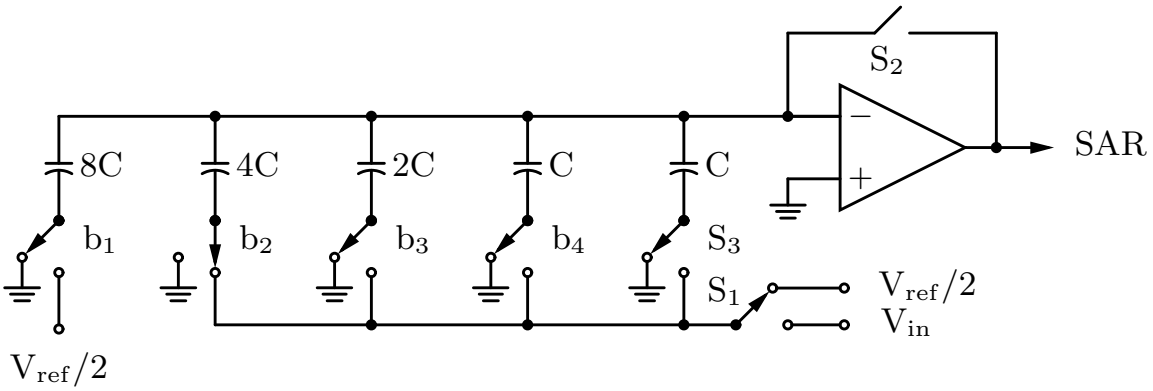
❑ $V_{in} \geq 0$

Bit cycling: $V_x = V_{OS} - V_{in}/2 + V_{ref}/8$

❑ $V_{in} < 0$

Level shift ($b_1 \rightarrow 0$): $V_x = V_{OS} - V_{in}/2 - V_{ref}/4$

Bit cycling: $V_x = V_{OS} - V_{in}/2 - V_{ref}/4 + V_{ref}/8 = V_{OS} - V_{in}/2 - V_{ref}/8$



Resistor-Capacitor Hybrid A/D Converter

❑ Combination of resistor string and capacitor array.

❑ Step 1: All the capacitors are charged to V_{in} while the comparator is being reset.

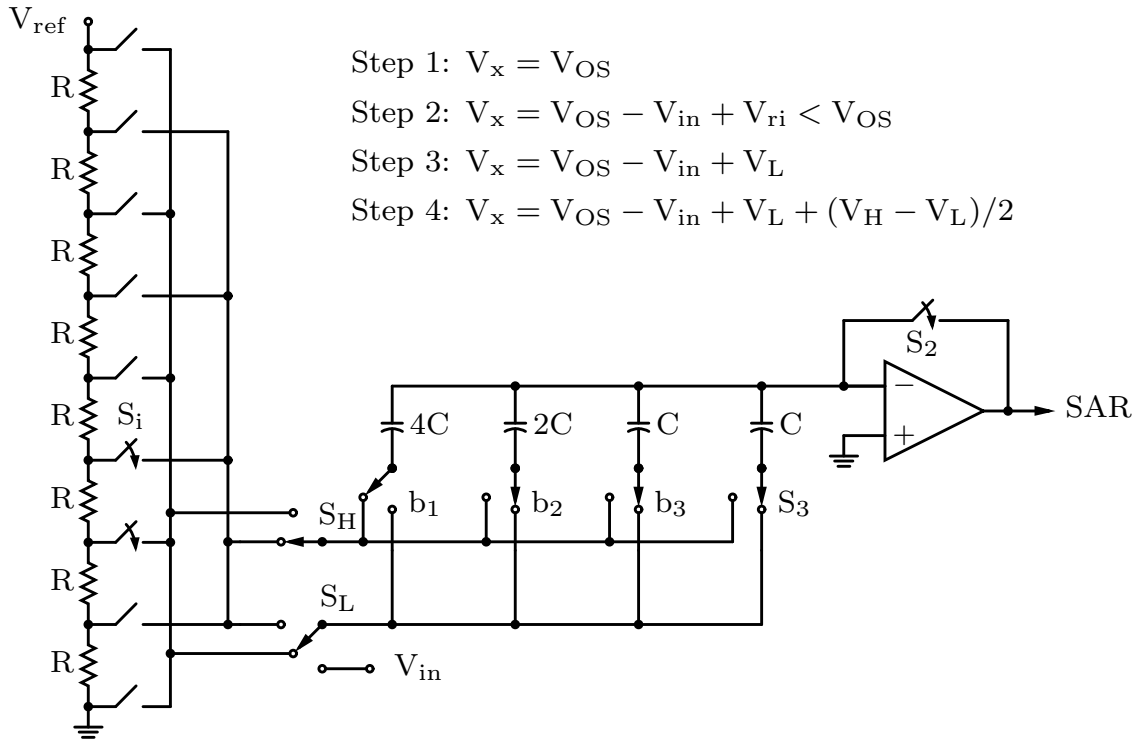
❑ Step 2: The operation is performed to find the two adjacent resistor nodes that have voltages larger and smaller than V_{in}.

To find the smaller voltage node, charging all the capacitors to the voltage of each resistor-string node, check the comparator output for ($V_x = V_{OS} - V_{in} + V_{ri} < V_{OS}$) from the top node.

❑ Step 3: All the capacitors are charged to the lower voltage.

❑ Step 4: The successive approximation is performed by switching capacitors to the bus having the larger voltage.

❑ A 6-bit resistor-capacitor hybrid A/D converter.



Step 1: $V_x = V_{OS}$

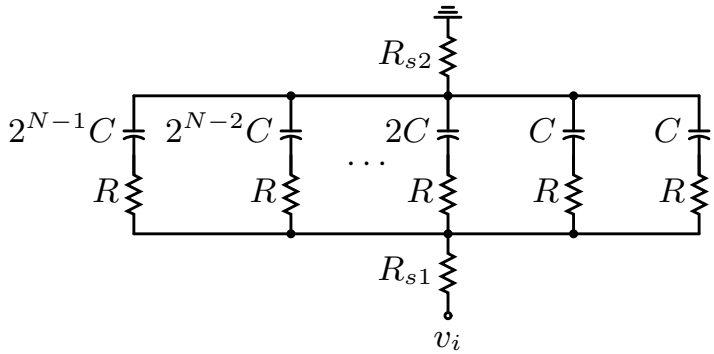
Step 2: $V_x = V_{OS} - V_{in} + V_{ri} < V_{OS}$

Step 3: $V_x = V_{OS} - V_{in} + V_L$

Step 4: $V_x = V_{OS} - V_{in} + V_L + (V_H - V_L)/2$

Speed Estimate for Charge-Redistribution Converters

❑ Simplified model of a capacitor array during sampling time.



❑ Open-circuit time constant and charging time for better than 0.5-LSB accuracy ($e^{-T/\tau} < 0.5/2^N$) → rough estimate for maximum sampling rate.

$$\tau = \sum_i R_{io}C_i = (R_{s1} + R + R_{s2})2^N C \simeq \frac{1}{\omega_H}, \quad T > 0.69(N+1)\tau$$

Algorithmic A/D Converter

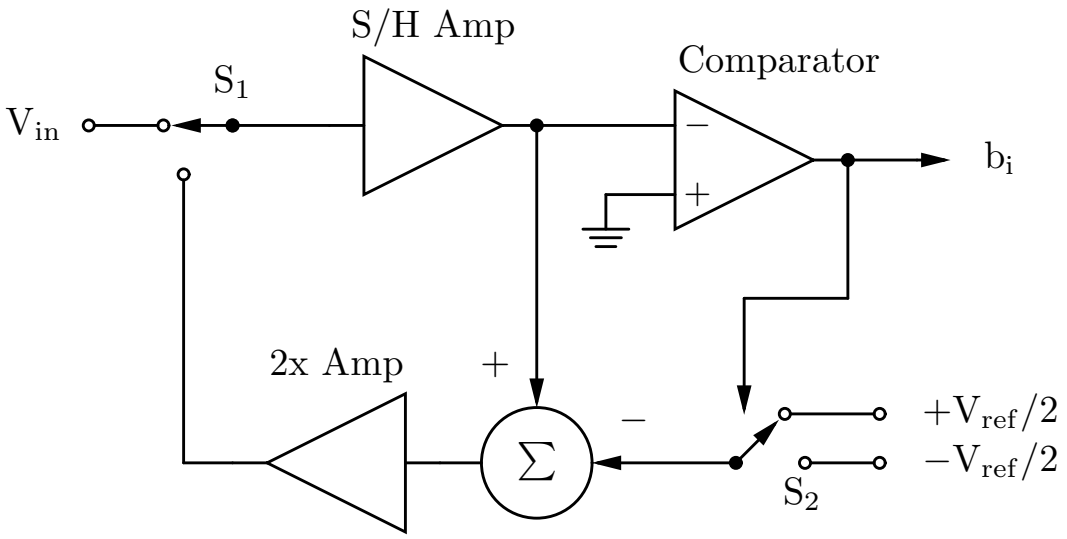
- ❏ Cyclic or recirculating A/D converter.
- ❏ A small amount of analog circuitry.
- ❏ S/H amp + 2x amp + comparator + subtraction circuit.
- ❏ An accurate multiply-by-two gain amp: sample the input signal twice using the same capacitor.
- ❏ Similar operation as SA converters: Whereas an SA converter halves the reference voltage in each cycle, an algorithmic converter *doubles the error voltage* while leaving the reference voltage unchanged.

$$8[V_{in} - (b_1 2^{-1} + b_2 2^{-2} + b_3 2^{-3})V_{ref}] = 2[2(2V_{in} - b_1 V_{ref}) - b_2 V_{ref}] - b_3 V_{ref} \rightarrow 0$$

C Code for A Bipolar Cyclic ADC

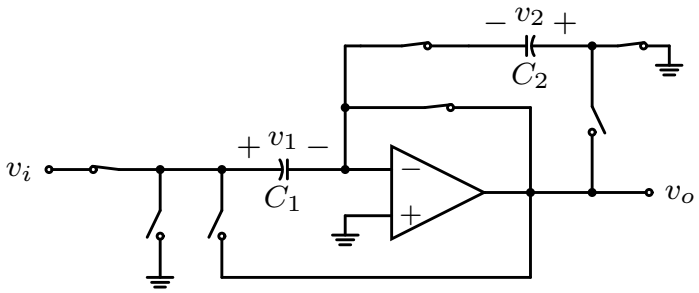
```
void CyclicAdc( n, b, vin, vref )
/* bipolar cyclic or algorithmic A/D converter  */
int   n,        /* resolution of ADC            */
      b[];      /* offset binary code of result */
float vin,      /* input voltage (± vref)       */
      vref;     /* reference voltage            */
{   int i; float vda = vin;
    for ( i = 0; i < n; i++ ) {
        if ( vda >= 0 ) {
            b[i] = 1;
            vda  = 2.0 * vda - vref;  // 2.0 * ( vda - vref/2 )
        } else {
            b[i] = 0;
            vda  = 2.0 * vda + vref;  // 2.0 * ( vda + vref/2 )
        }
    }
}
```


❏ A bipolar algorithmic A/D converter.

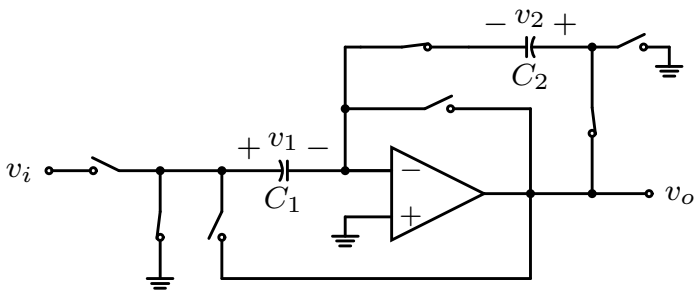


Multiply-by-Two Gain Amp for Cyclic Converters

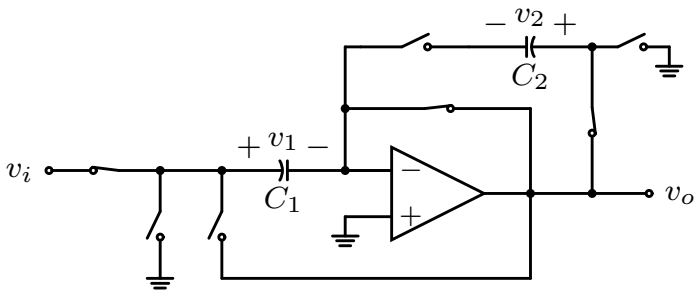
❏ Phase 1: sample v_i and V_{OS}: $v_1 = v_i - V_{OS}$, $v_2 = -V_{OS}$, $v_o = V_{OS}$.



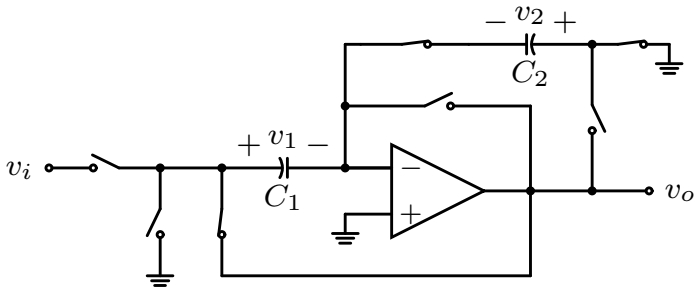
❏ Phase 2: transfer charge Δq_1 to C_2: $v_1 = -V_{OS}$, $v_2 = -V_{OS} + \frac{C_1}{C_2} v_i$, $v_o = \frac{C_1}{C_2} v_i$.



❑ Phase 3: sample v_i again: $v_1 = v_i - V_{OS}$, $v_2 = -V_{OS} + \frac{C_1}{C_2} v_i$, $v_o = V_{OS}$.



❑ Phase 4: combine q_1 and Δq_2 on C_1: $v_2 = -V_{OS}$, $\Delta q_2 = C_2 \Delta v_2 = C_1 v_i$,
$v_1 = v_i - V_{OS} + \frac{\Delta q_2}{C_1} = 2v_i - V_{OS}$, $\boxed{v_o = 2v_i}$ → independent of C_1, C_2, and V_{OS}.



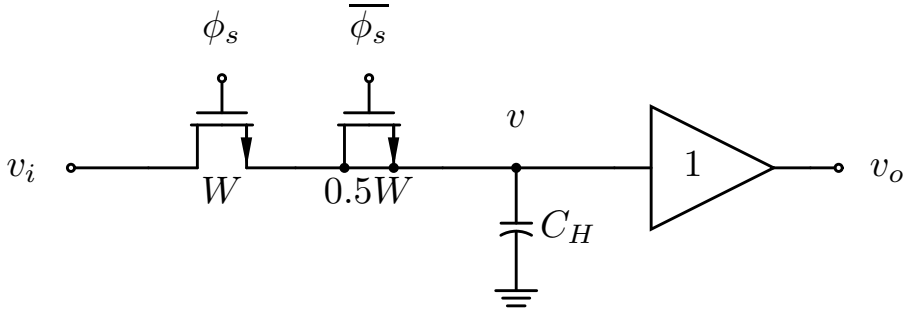
Major Error Sources for SC A/D Converters

❑ Offset voltage of op amps ⇒ offset-cancellation method.

❑ Capacitor mismatches ⇒ ratio-independent technique.

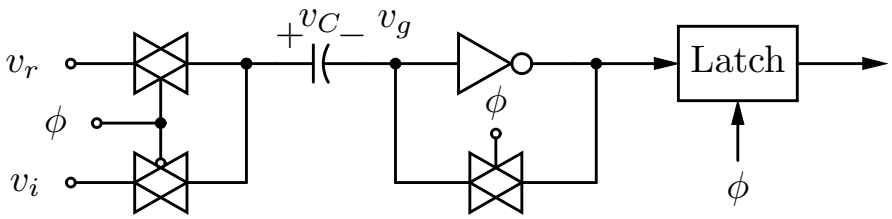
❑ Finite gain of op amps ⇒ gain-insensitive technique.

❑ Clock feedthrough and charge injection ⇒ dummy switches, CMOS switches, and fully differential structures.



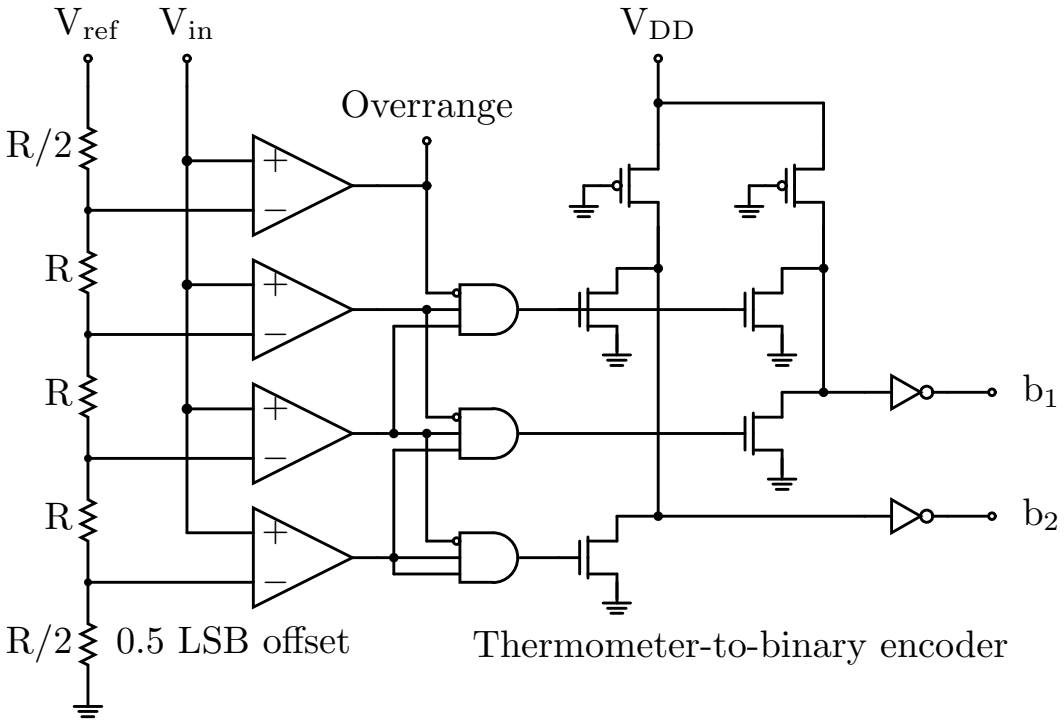
Flash A/D Converters

- ❏ Parallel A/D converters.
- ❏ Standard approach for realizing very high speed: 8-b 500-MHz.
- ❏ A resistor string + 2^N comparators + an encoder.
- ❏ Fast but large area and power → small clocked comparator using a CMOS inverter ($\phi = 1$: $v_C = v_r - V_{inv}$, $\phi = 0$: $v_g = v_i - v_r + V_{inv}$).



- ❏ Output code = thermometer code: need of a code converter.
- ❏ Top and bottom R/2 resistors for 0.5 LSB offset.

- ❏ A 2-bit flash A/D converter.



Design Issues for Flash A/D Converters

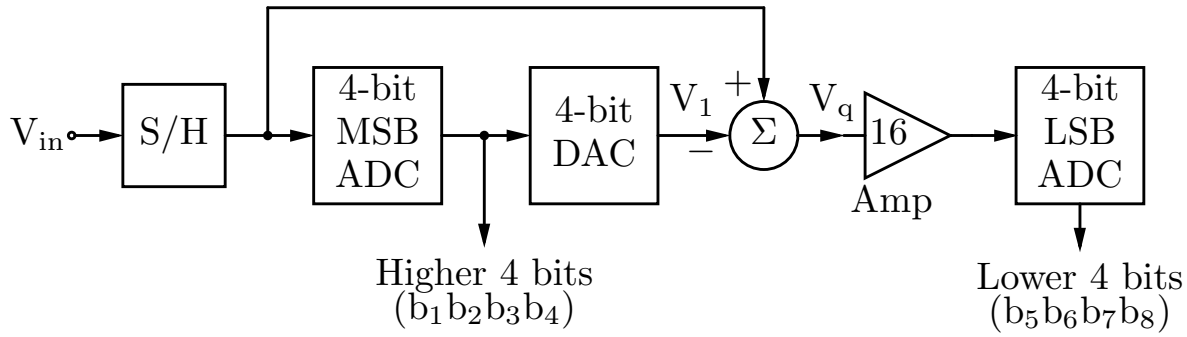
- ❏ Input capacitive loading: A large parasitic load at the node V_{in} due to the large number of comparators → speed limit.
- ❏ Resistor-string bowing: Errors in the voltages of the nodes of the resistor string due to the input currents of bipolar comparators.
- ❏ Signal and clock delay: Small differences in the arrival of clock or input signals at each comparators. It would only take 5 ps to change through 1 LSB for the case where a 250-MHz 1-V peak-input sinusoid is being encoded by an 8-bit A/D converter with $V_{ref} = 2$ V ⇒ usage of a high-speed S/H circuit which can be more difficult to realize than the flash converter itself.

- ❏ Substrate and power-supply noise: For $V_{ref} = 2$ V and an 8-bit converter, only 7.8 mV of noise injection would cause a 1 LSB error ⇒ clock shielding, running differential clocks closely together, separation of analog power supply from digital supply, on-chip power-supply bypassing (small resistors in series with bypass capacitors).
- ❏ Bubble error removal: A bubble of lone 1 or 0 will occur within a thermometer code due to comparator metastability, noise, crosstalk, limited bandwidth, etc → 3-input NAND gate to remove a bubble.
- ❏ Flashback: When latched comparators are switched from track to latch mode, or vice-versa, there is charge glitch at the input to the latch. If there is no preamplifier, this will cause major errors due to the unmatched impedances at the comparator inputs (one input: resistor string – other input: input signal).

Two-Step A/D Converters

- ❏ Two-step or subranging A/D converters.
- ❏ The most popular approach for high speed medium accuracy.
- ❏ Less silicon area, less power, less capacitive loading: $2 \times 2^{N/2}$ comparators (2^N comparators in N-bit flash converters).
- ❏ Less stringent resolution of comparators: $\frac{N}{2}$ bit accuracy.
- ❏ Larger latency delay for the first output.
- ❏ $\frac{N}{2}$-bit MSB ADC + $\frac{N}{2}$-bit DAC + $2^{N/2}$ gain amp + $\frac{N}{2}$-bit LSB ADC.
- ❏ A 10-b 75-MHz 2-W subranging A/D converter in 1990.

- ❏ An 8-bit two-step A/D converter.



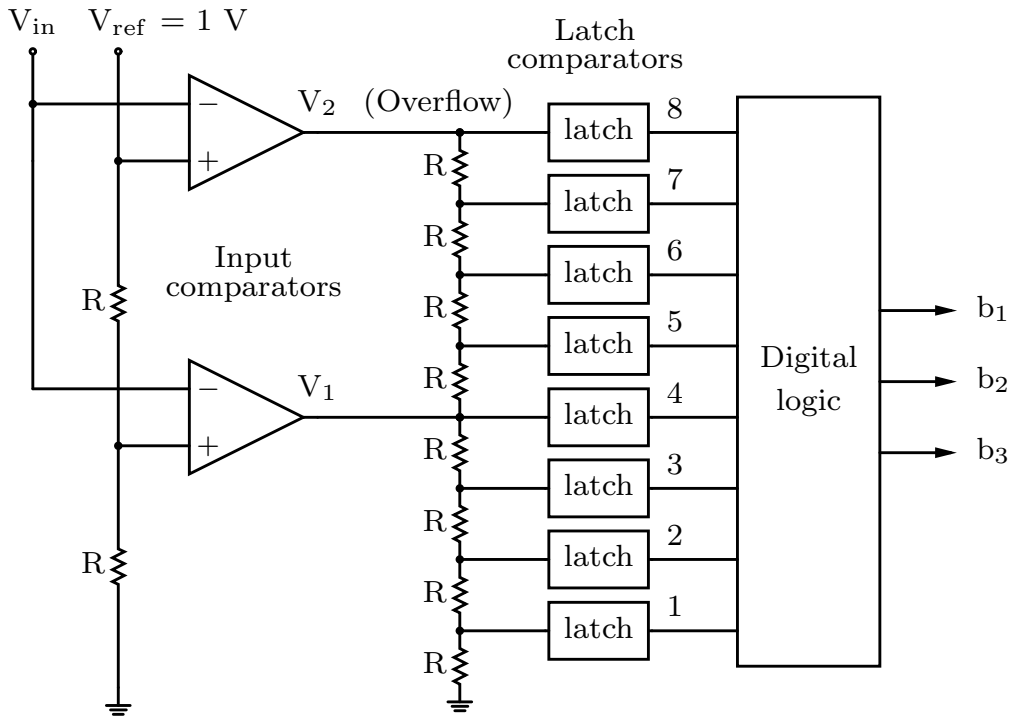
- ❏ The reason for digital error correction is to significantly ease the requirements on the 4-bit MSB A/D converter (8 → 4-bit accuracy). V_{in} is found by properly combining the digital values of V_1 and V_q.

$$V_{in} = V_1(4 \text{ bits}) \oplus V_q(5 \text{ bits}) \quad (\text{gain} = 8)$$

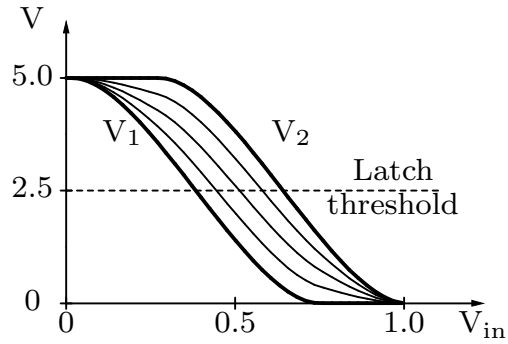
Interpolating A/D Converters

- ❏ Input comparators as linear amplifiers near threshold voltages.
- ❏ Interpolation between output voltages of two input comparators by a resistor string ⇒ interpolating factor of M.
- ❏ Reduction in the number of input comparators → lower input capacitance, reduced power dissipation, lower number of accurate reference voltages created by a resistor string.
- ❏ Adding series resistors to equalize delay times to latch comparators.
- ❏ Interpolation by capacitors or current mirrors instead of resistor string.
- ❏ A 10-b 20-MHz 30-mW CMOS ADC in 1993.
- ❏ A 8-b 100-MHz 1.5-μm CMOS ADC in 1993.

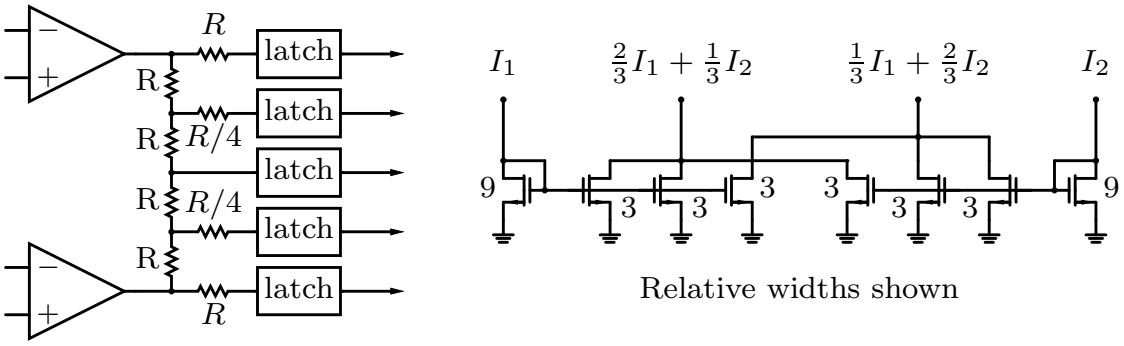
- ❏ A 3-bit interpolating A/D converter with interpolating factor of 4.



❏ Possible transfer responses for input-comparator output signals.



❏ Delay equalization by adding resistors and current interpolation by mirrors.



Folding A/D Converters

❏ Folding architecture by analog preprocessing of folding blocks (FBs). The *folding rate* is the number of output transitions for single FB as V_{in} is swept over input range → number of bits in the converter.

❏ Reduction in the number of latch comparators $< 2^N$ (interpolating)

❏ Folding blocks realized using cross-coupled differential pairs.

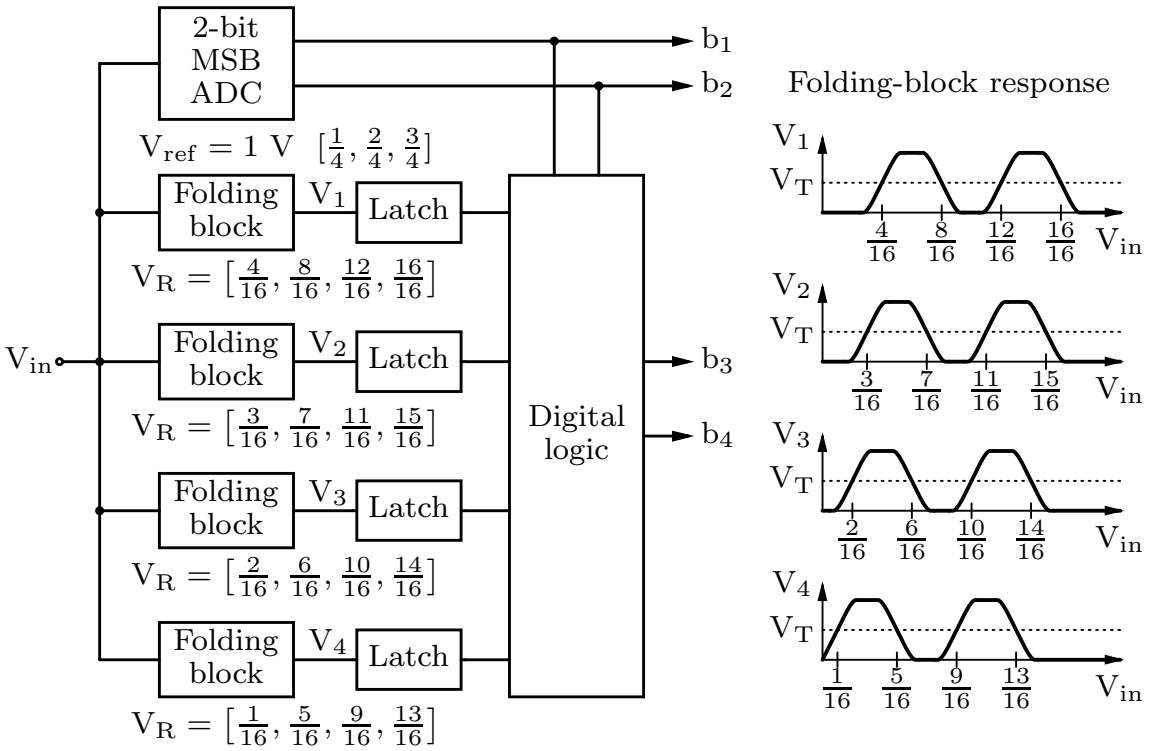
❏ The MSB converter would usually be realized by combining FB signals ($b_2 = V_1$, $b_1 = V_c$ of transistor connected to V_{r2} in FB V_1)

❏ Frequency-multiplying effect of FBs limits the practical folding rate.

❏ Folding and interpolating ADC: reduction in the number of FBs.

❏ An 80-MHz, 80-mW, 8-b CMOS folding A/D converter in 1996.

❑ A 4-bit folding A/D converter with a folding rate of 4.



Pipelined A/D Converters

❑ Pipelined ADC: The first stage operates on the most recent sample while all other stages operate on residues from previous samples.

❑ Each stage: SHA, ADSC, DAC, subtracter.

❑ N clock latency for the first output: offset-binary code.

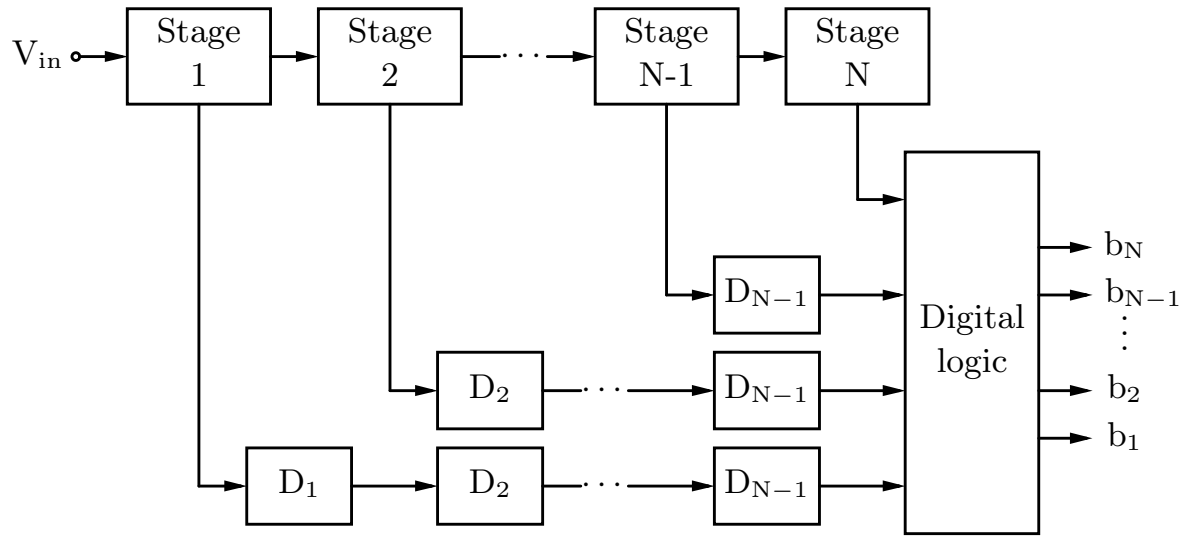
❑ High throughput rate: two clocks per conversion, low hardware cost.

❑ Gain amplifiers in the first few stages have most stringent accuracy.

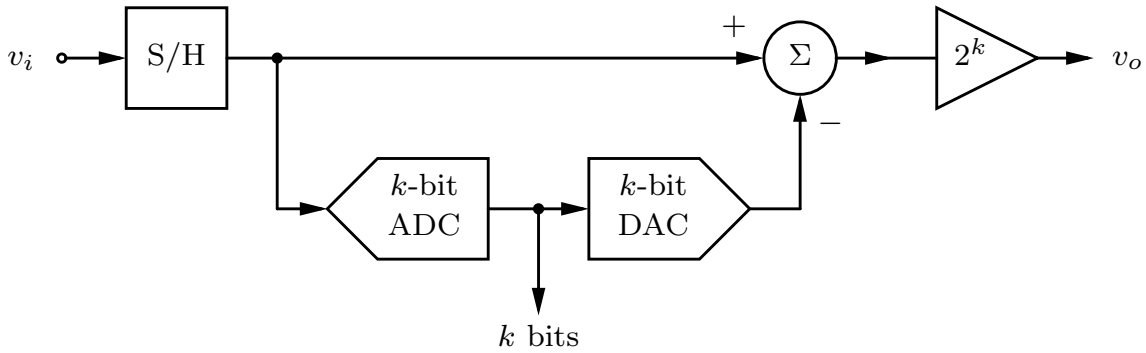
❑ Digital error correction: redundancy (more bit per stage).

❑ Implementation by switched-capacitor S/H/Gain amplifiers.

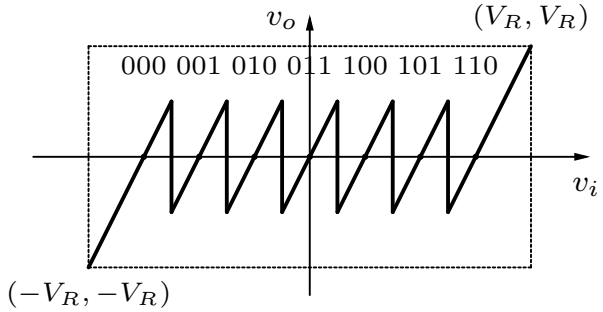
❑ A pipelined A/D converter.



❑ One stage of a pipelined A/D converter.

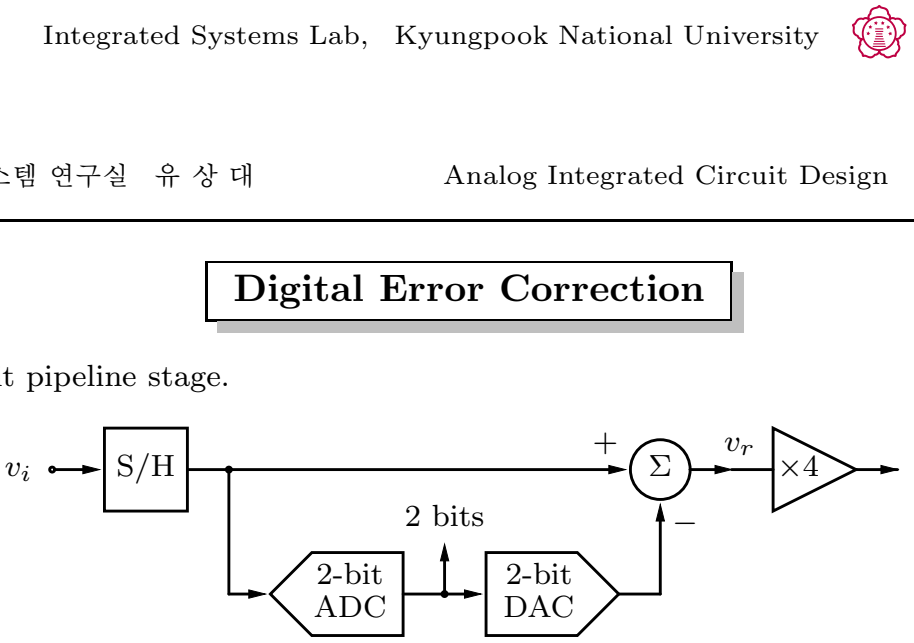


❑ Transfer function of a 2.8-bit pipeline stage: $\log_2 7 \simeq 2.8$ b.



Error Sources in Pipelined ADCs

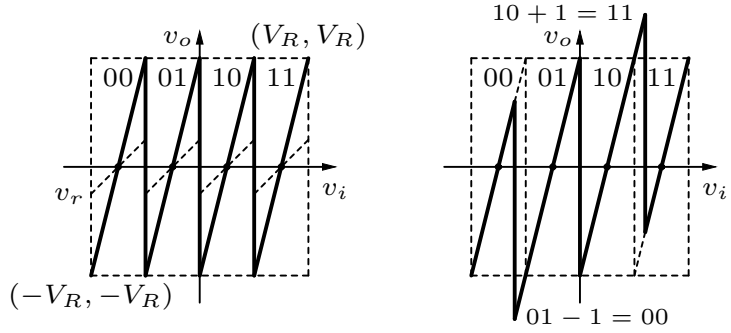
- ❏ Offset and gain errors in S/H circuits and amplifiers.
- ❏ A/D subconverter nonlinearity.
- ❏ D/A subconverter nonlinearity.
- ❏ Op amp settling-time errors.
- ❏ Effects of gain, offset, and A/D subconverter nonlinearity are reduced or eliminated with digital error correction → D/A subconverter nonlinearity and op amp settling-time errors limit the performance.
- ❏ The digital error correction *postpones decisions* on inputs that are near the A/D subconverter decision levels until the residues from these inputs are amplified to the point where similar nonlinearity in later A/D subconverters is insignificant.



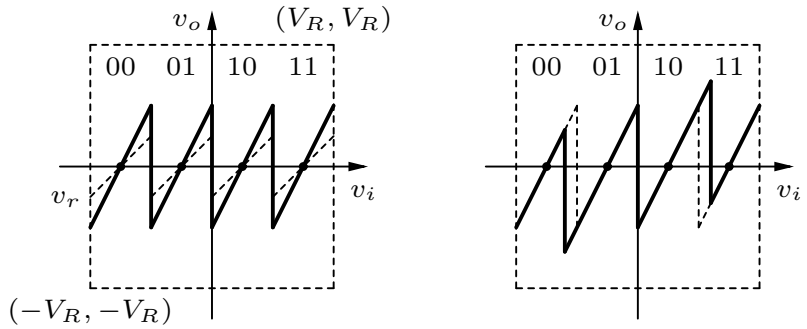
Digital Error Correction

- ❏ A 2-bit pipeline stage.

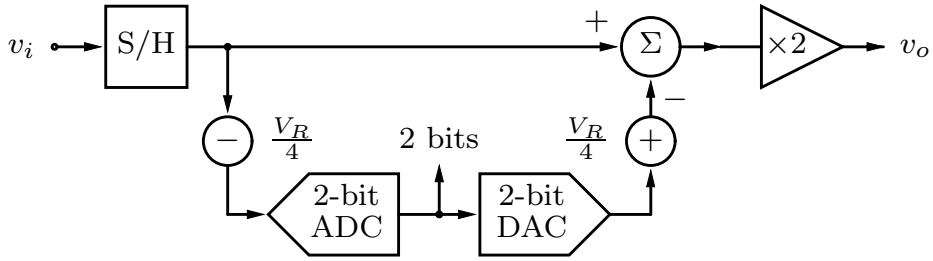
- ❏ Transfer functions: ideal and practical, error detection and correction (overrange: $10+1=11$, $v_o - V_{\text{LSB}}$, underrange: $01-1=00$, $v_o + V_{\text{LSB}}$)



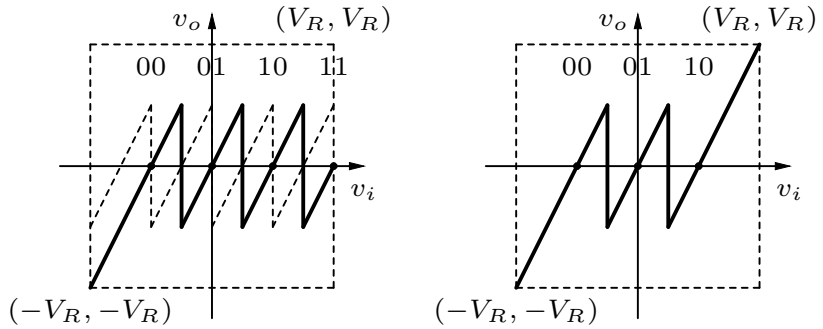
❏ Reduction of the stage gain to detect the overrange: a factor of 2.
($\times 4 \rightarrow \times 2$, correction range $= V_R/4$)



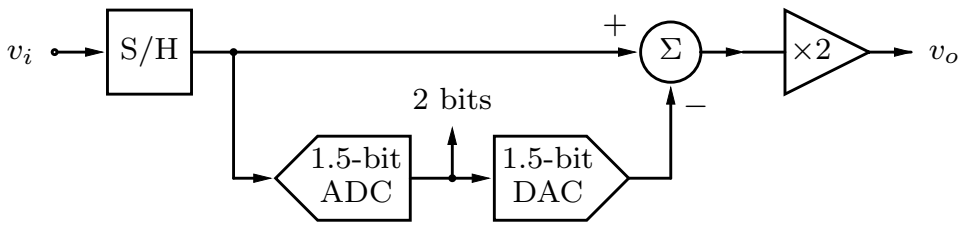
❏ Addition of systematic offsets to eliminate the subtraction (ADC offset shifts decision levels to the right, DAC offset shifts transfer function down)



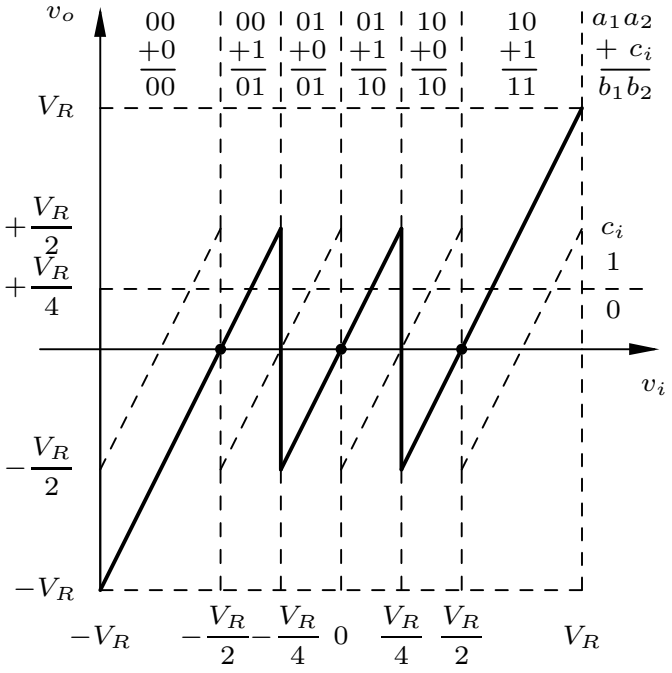
❏ Transfer functions with offsets and without the top decision level
(a correctable error because correction range $= V_R/4$)



❏ A 1.5-bit pipeline stage: The digital correction and output coding is done by adding $(i+1)$th stage output to ith stage output with one bit overlap.
(ADC decision levels $= -\frac{V_R}{4}, +\frac{V_R}{4}$; DAC voltage levels $= -\frac{V_R}{2}, 0, +\frac{V_R}{2}$)



❑ Correction table for 6 regions of the transfer function: dotted line = last stage.
(raw output a_1a_2, correction c_i from stage $i+1$, corrected output b_1b_2)



raw outputs	final output
00 00 00	0000
00 00 01	0001
00 00 10	0010
00 01 01	0011
00 01 10	0100
00 10 01	0101
01 00 10	0110
01 01 01	0111
01 01 10	1000
01 10 01	1001
10 00 10	1010
10 01 01	1011
10 01 10	1100
10 10 01	1101
10 10 10	1110
10 10 11	1111

Time-Interleaved A/D Converters

❑ Very high speed A/D conversion.

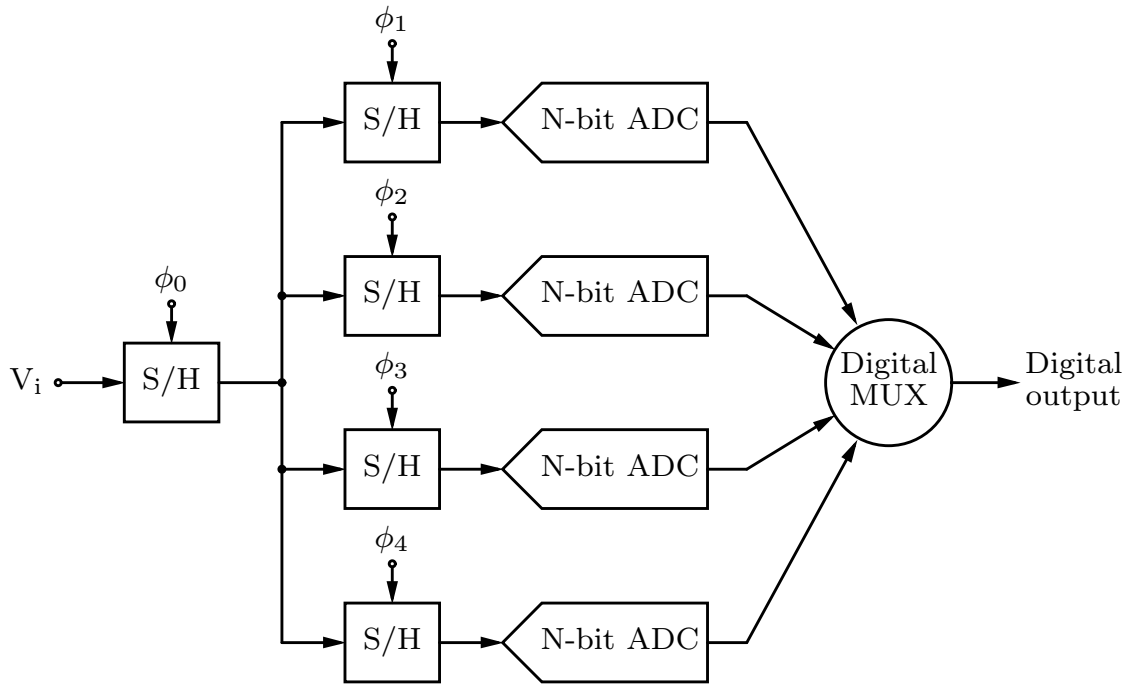
❑ *Parallel operation* of many A/D converters.

❑ The input S/H amplifier is critical, sometimes realized in a different technology (GaAs S/H circuits).

❑ Mismatches in the channels will produces tones at f_s/m.
For example, consider a dc input signal in a time-interleaved converter where one converter has a dc offset of 100 mV. This converter will produce every fourth digital word different from other three.

❑ A 1-GHz 6-bit ADC in 1987.

❑ A four-channel time-interleaved A/D converter.



Homework

❑ Modeling of the n-bit pipelined A/D converter with 1.5-bit stage by the C language. What is the correction range for ADC decision levels?

❑ Problems 13.2, 13.4, 13.7, 13.12, 13.14, 13.21.

References

[1] S. H. Lewis, H. S. Fetterman, G. F. Gross, Jr., R. Ramachandran, and T. R. Viswanathan, "A 10-b 20-Msample/s analog-to-digital converter", *IEEE J. of Solid-State Circuits*, vol. 27, no. 3, pp. 351–358, 1992.

[2] G. Chien, *"High Speed, Low Power, Low Voltage Pipelined Analog-to-Digital Converter"*, MS Thesis, U. C. Berkeley, 1996.