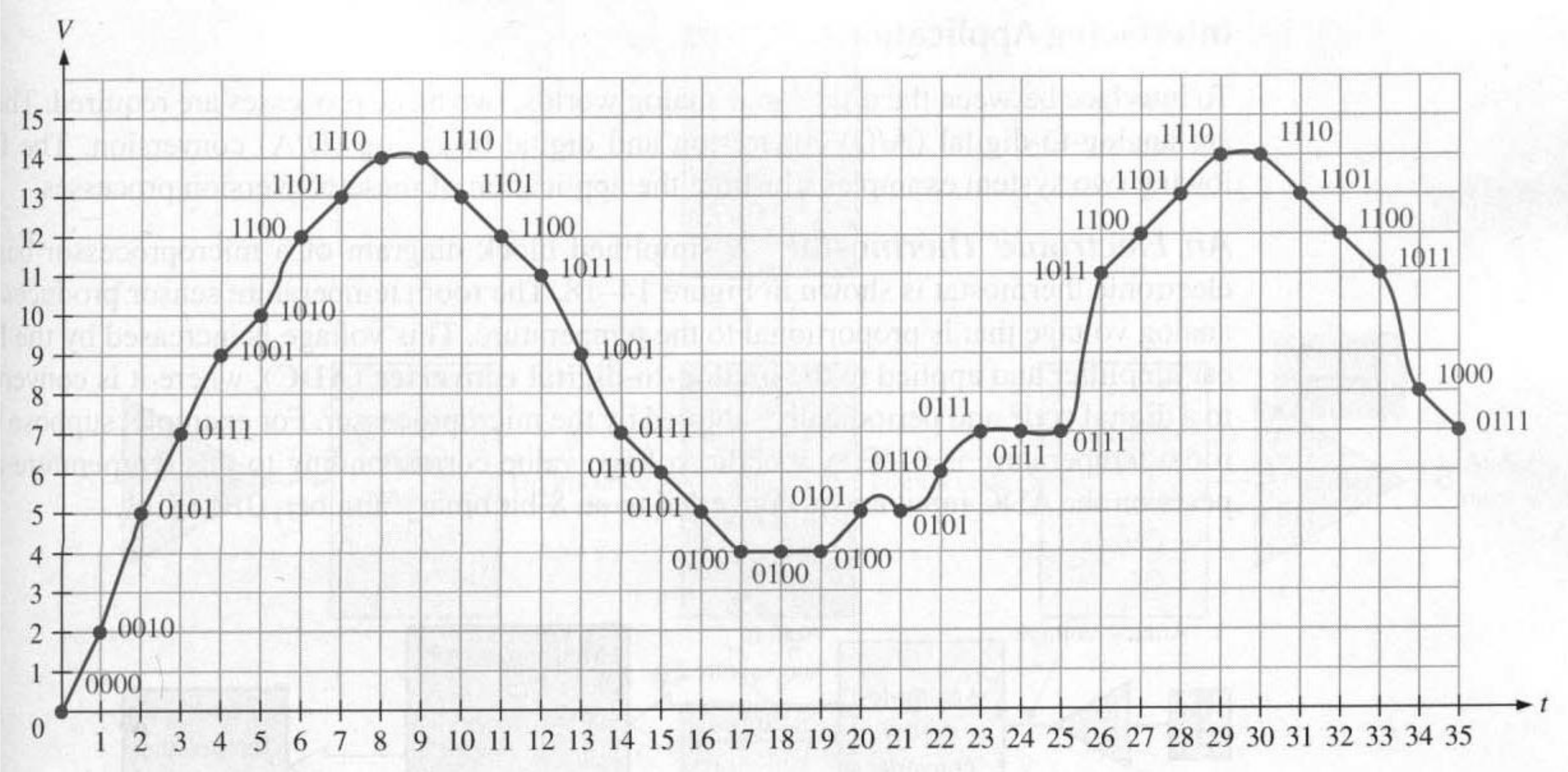
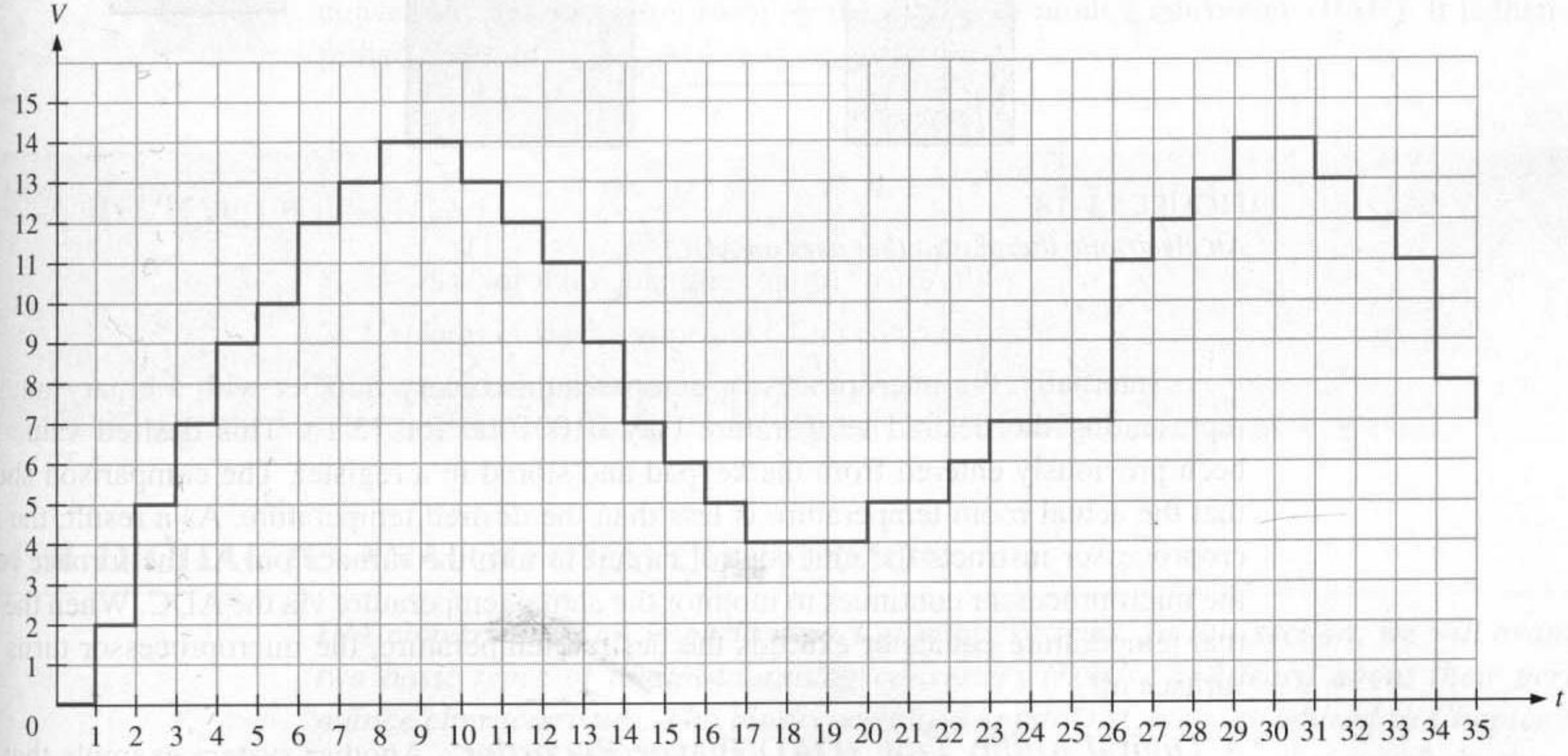
Lecture 25

EC103

Discrete (Digital) Points on an Analog Curve



Digital Representation of an Analog curve



Introduction

- Real signals (e.g., a voltage measured with a thermocouple or a speech signal recorded with a microphone) are analog quantities, varying continuously with time.

Introduction

- Real signals (e.g., a voltage measured with a thermocouple or a speech signal recorded with a microphone) are analog quantities, varying continuously with time.
- Digital format offers several advantages: digital signal processing, storage, use of computers, robust transmission, etc.

Introduction

- Real signals (e.g., a voltage measured with a thermocouple or a speech signal recorded with a microphone) are analog quantities, varying continuously with time.
- Digital format offers several advantages: digital signal processing, storage, use of computers, robust transmission, etc.
- An ADC (Analog-to-Digital Converter) is used to convert an analog signal to the digital format.

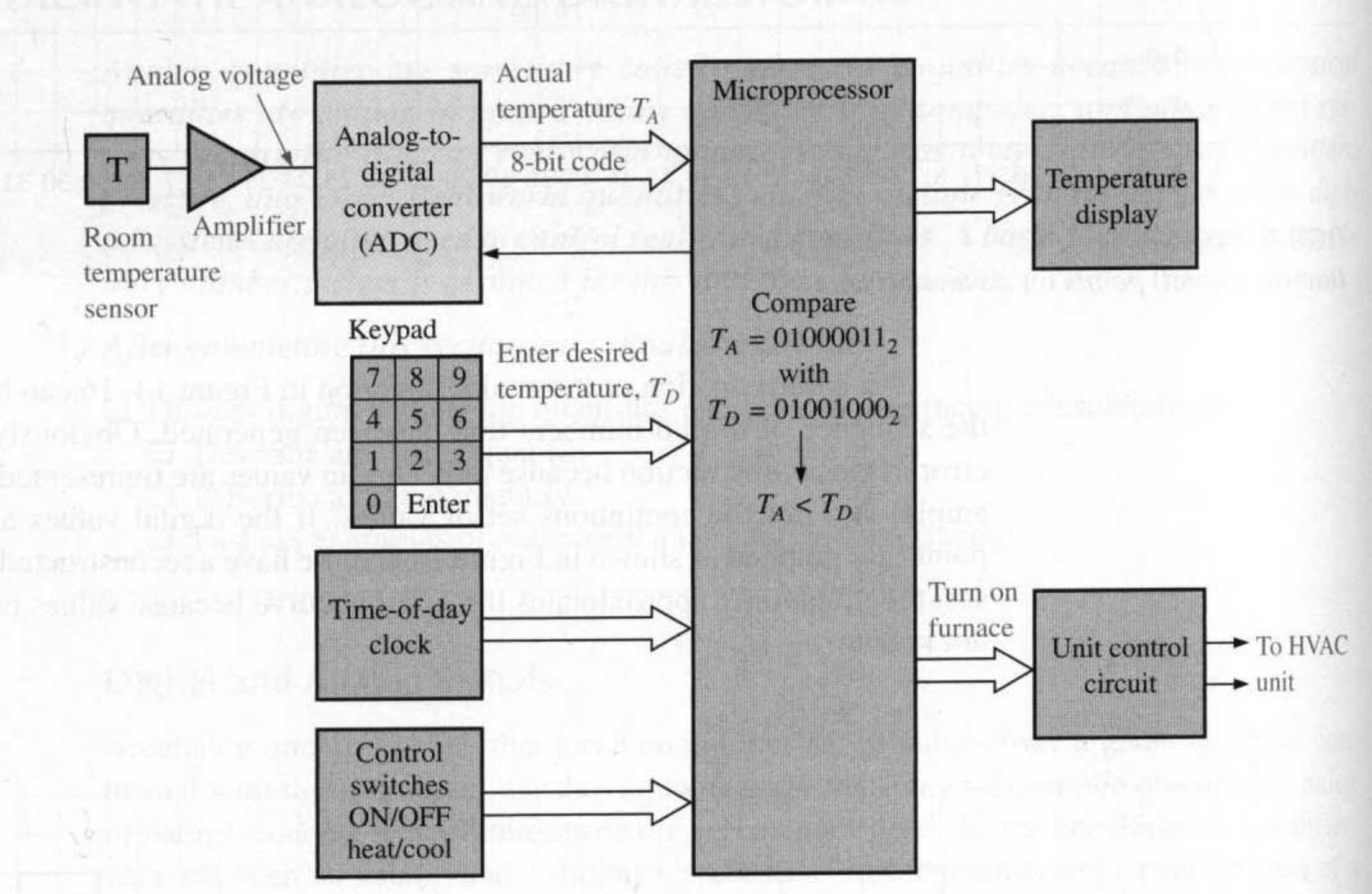
Introduction

- Real signals (e.g., a voltage measured with a thermocouple or a speech signal recorded with a microphone) are analog quantities, varying continuously with time.
- Digital format offers several advantages: digital signal processing, storage, use of computers, robust transmission, etc.
- An ADC (Analog-to-Digital Converter) is used to convert an analog signal to the digital format.
- The reverse conversion (from digital to analog) is also required. For example, music stored in a DVD in digital format must be converted to an analog voltage for playing out on a speaker.

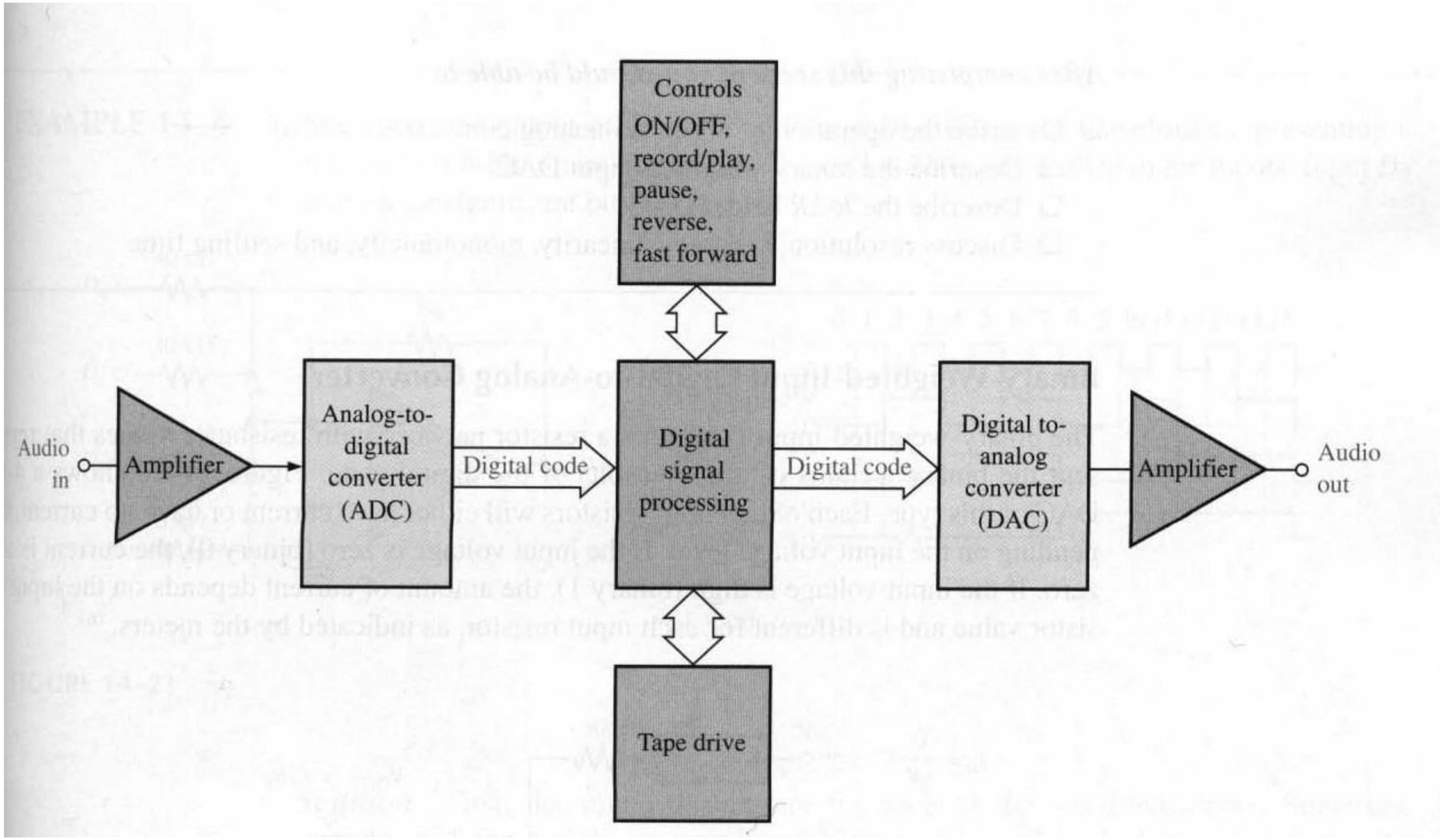
Introduction

- Real signals (e.g., a voltage measured with a thermocouple or a speech signal recorded with a microphone) are analog quantities, varying continuously with time.
- Digital format offers several advantages: digital signal processing, storage, use of computers, robust transmission, etc.
- An ADC (Analog-to-Digital Converter) is used to convert an analog signal to the digital format.
- The reverse conversion (from digital to analog) is also required. For example, music stored in a DVD in digital format must be converted to an analog voltage for playing out on a speaker.
- A DAC (Digital-to-Analog Converter) is used to convert a digital signal to the analog format.

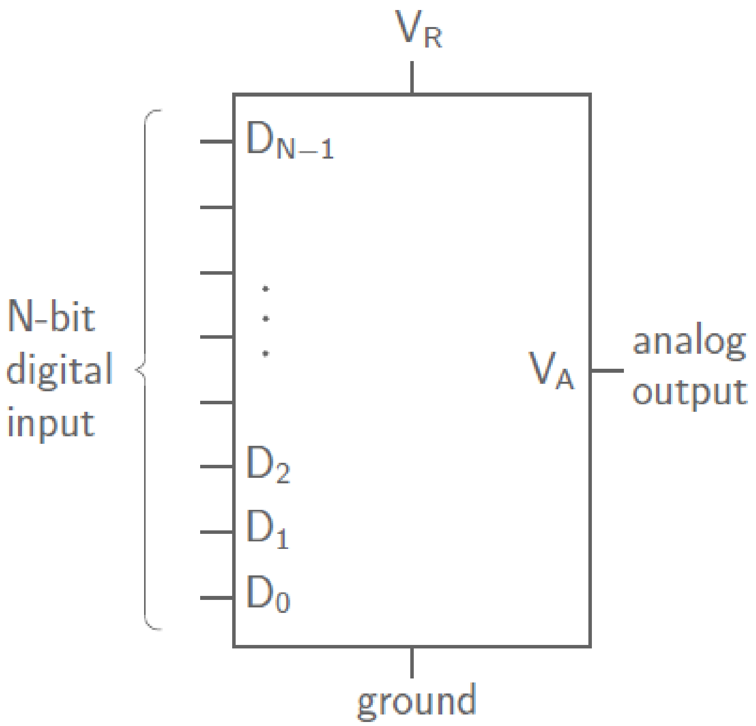
Interfacing the Analog and Digital Worlds: Example 1: An Electronic Thermostat



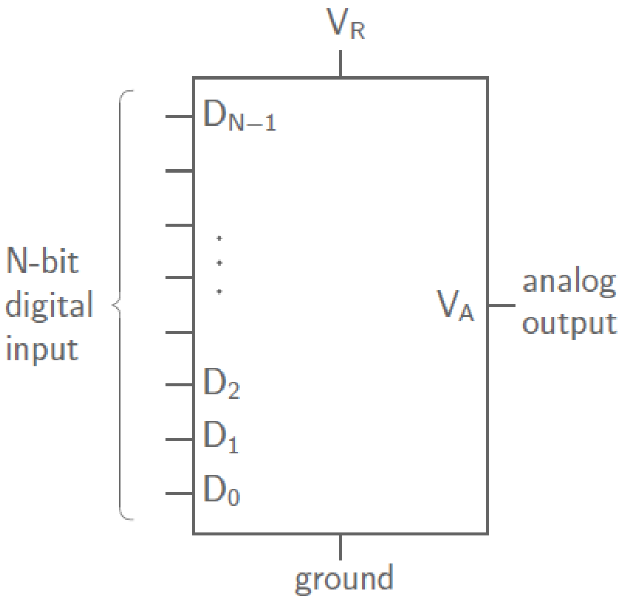
Interfacing the Analog and Digital Worlds: Example 2: A Digital Audio Recorder/Player



DAC

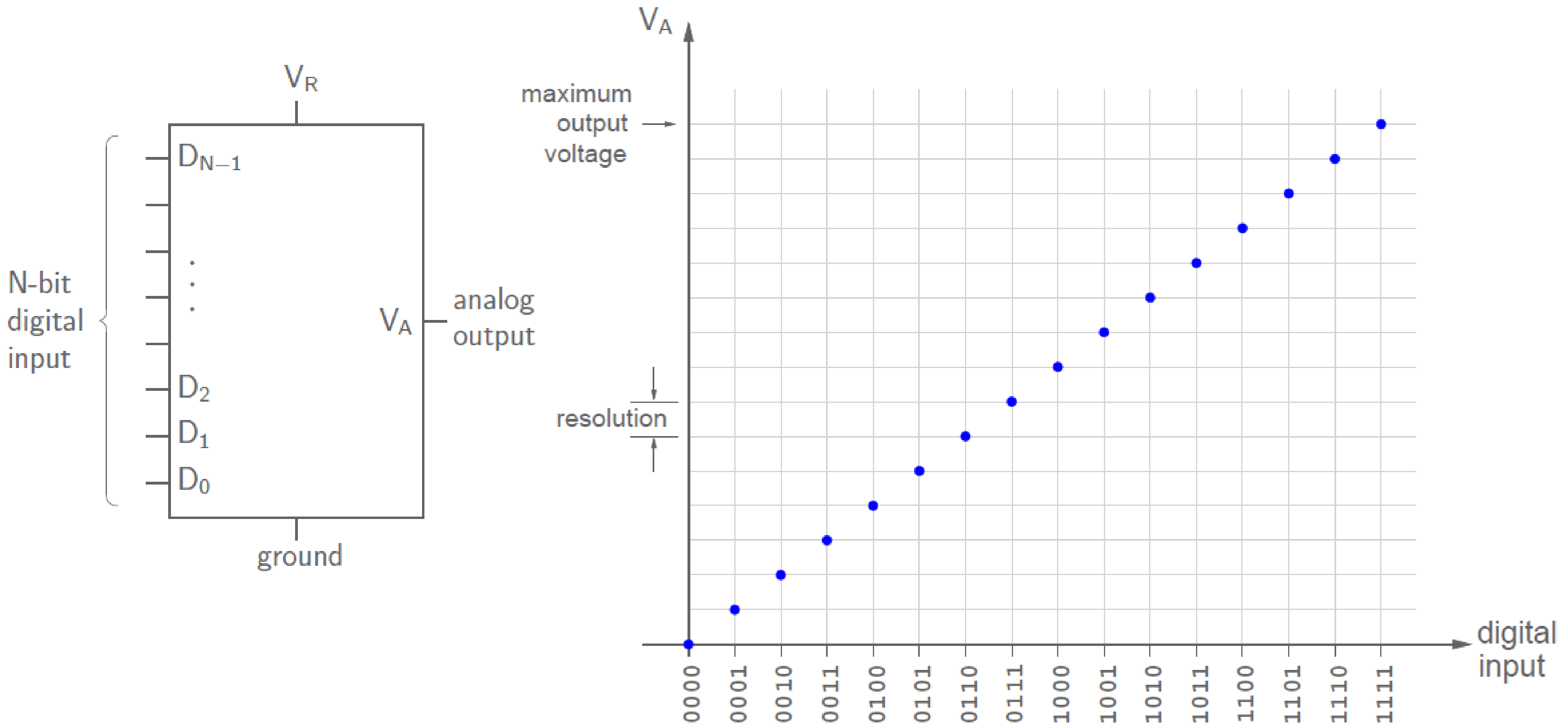


DAC



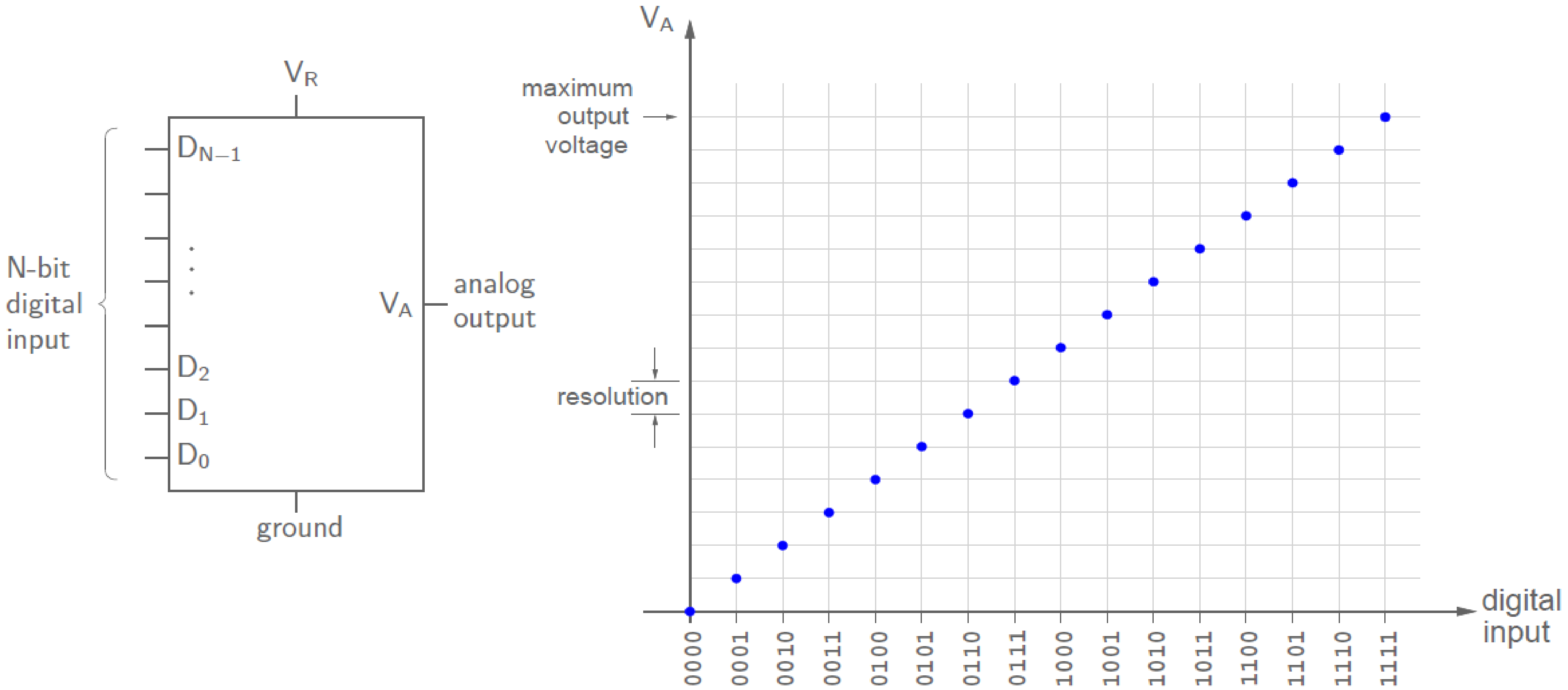
- ∗ For a 4-bit DAC, with input $S_3 S_2 S_1 S_0$, the output voltage is $V_A = K\left[(S_3 \times 2^3) + (S_2 \times 2^2) + (S_1 \times 2^1) + (S_0 \times 2^0)\right]$.
  In general, $V_A = K \sum_0^{N-1} S_k 2^k$.

DAC



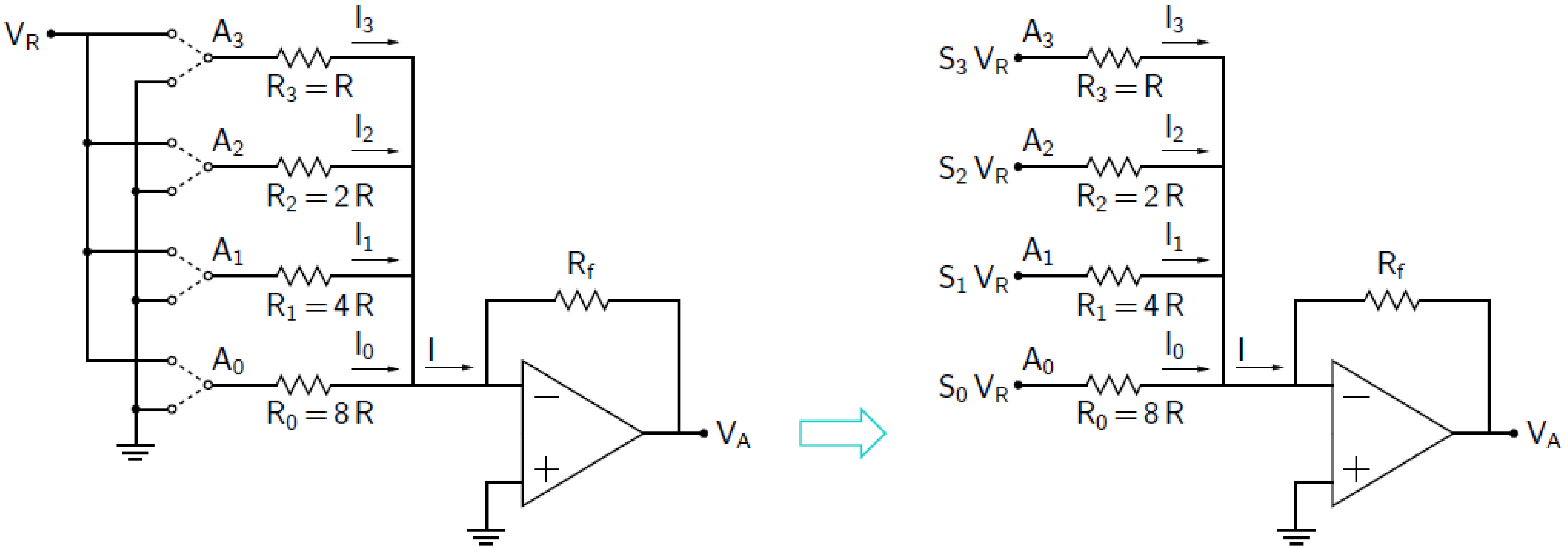
∗ For a 4-bit DAC, with input $S_3S_2S_1S_0$, the output voltage is $V_A = K\left[(S_3 \times 2^3) + (S_2 \times 2^2) + (S_1 \times 2^1) + (S_0 \times 2^0)\right]$.
In general, $V_A = K \sum_0^{N-1} S_k 2^k$.

DAC

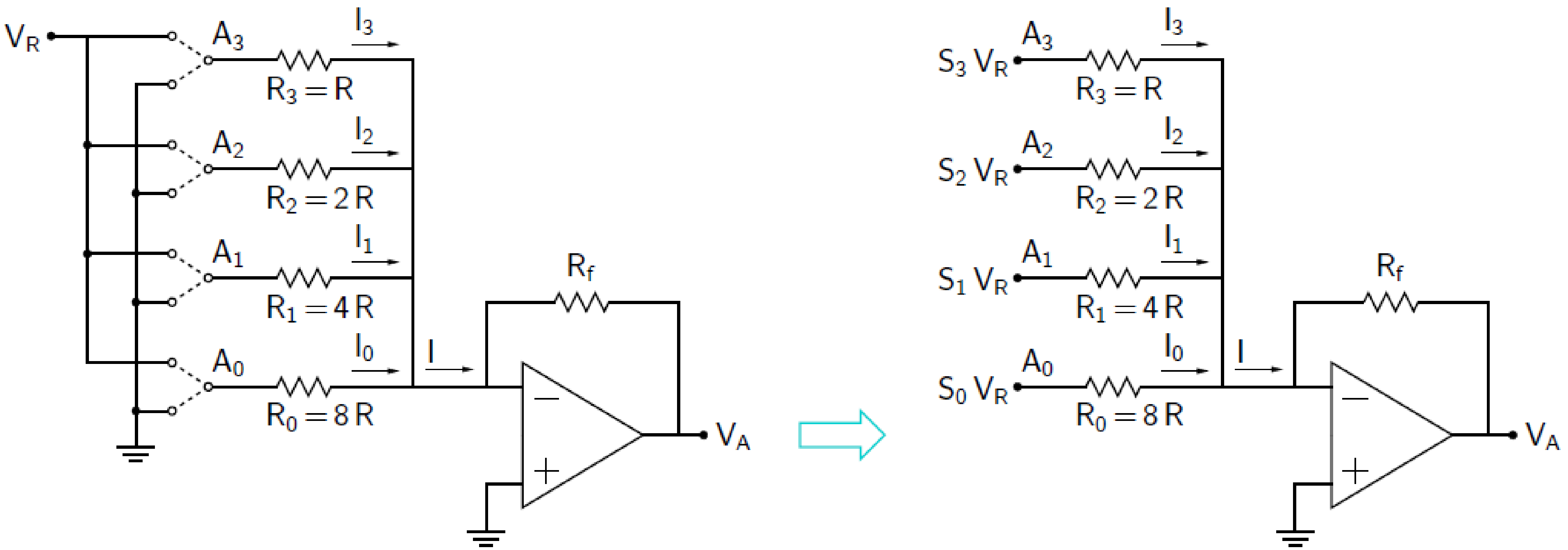


- For a 4-bit DAC, with input $S_3 S_2 S_1 S_0$, the output voltage is $V_A = K\left[(S_3 \times 2^3) + (S_2 \times 2^2) + (S_1 \times 2^1) + (S_0 \times 2^0)\right]$.
  In general, $V_A = K \sum_0^{N-1} S_k 2^k$.
- K is proportional to the reference voltage V_R. Its value depends on how the DAC is implemented.

DAC using binary-weighted resistors

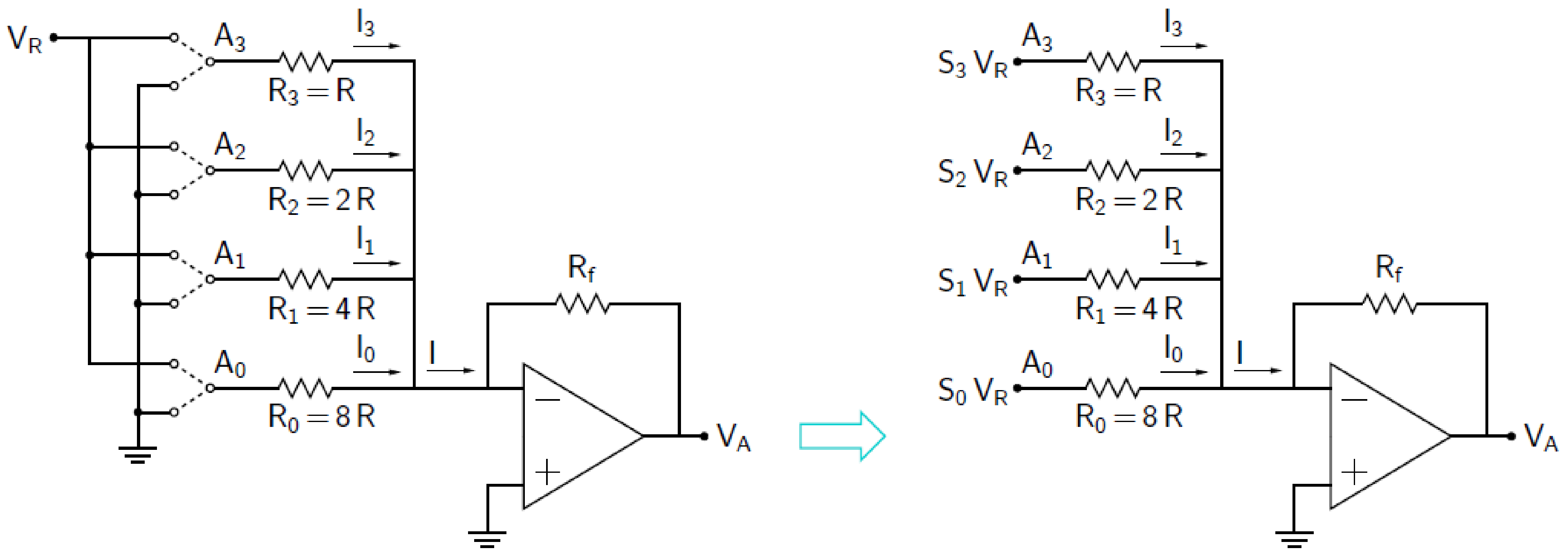


DAC using binary-weighted resistors



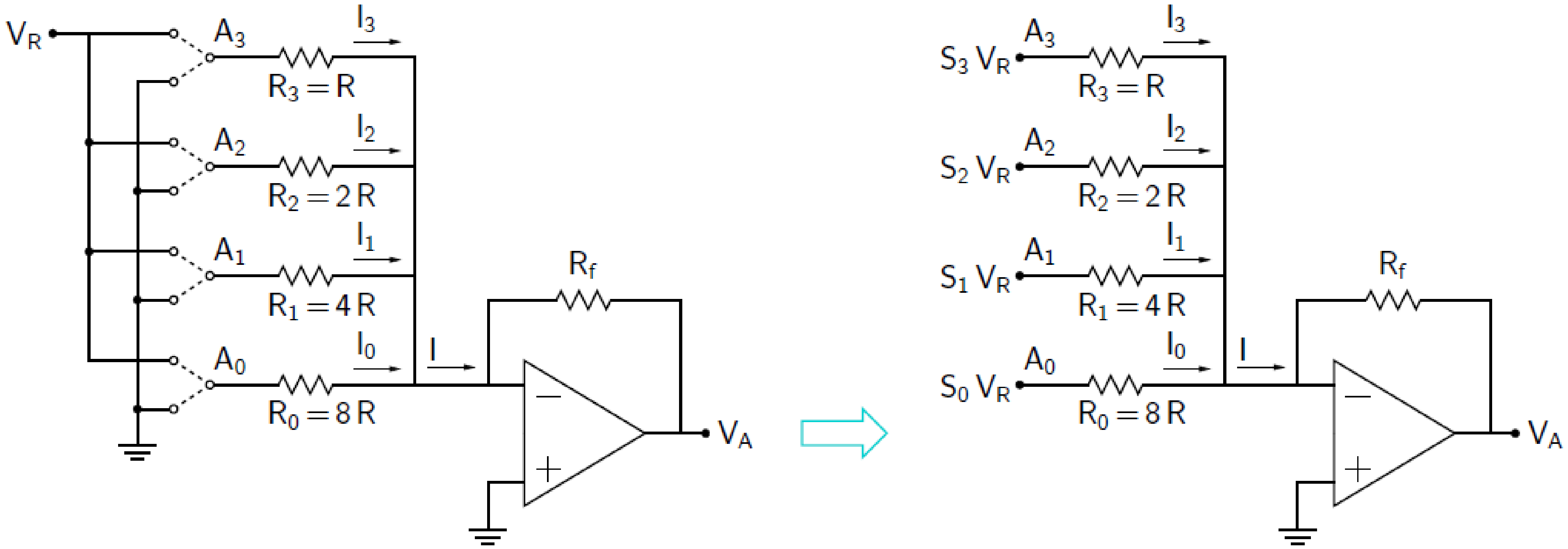
- ∗ If the input bit S_k is 1, A_k gets connected to V_R; else, it gets connected to ground.

DAC using binary-weighted resistors



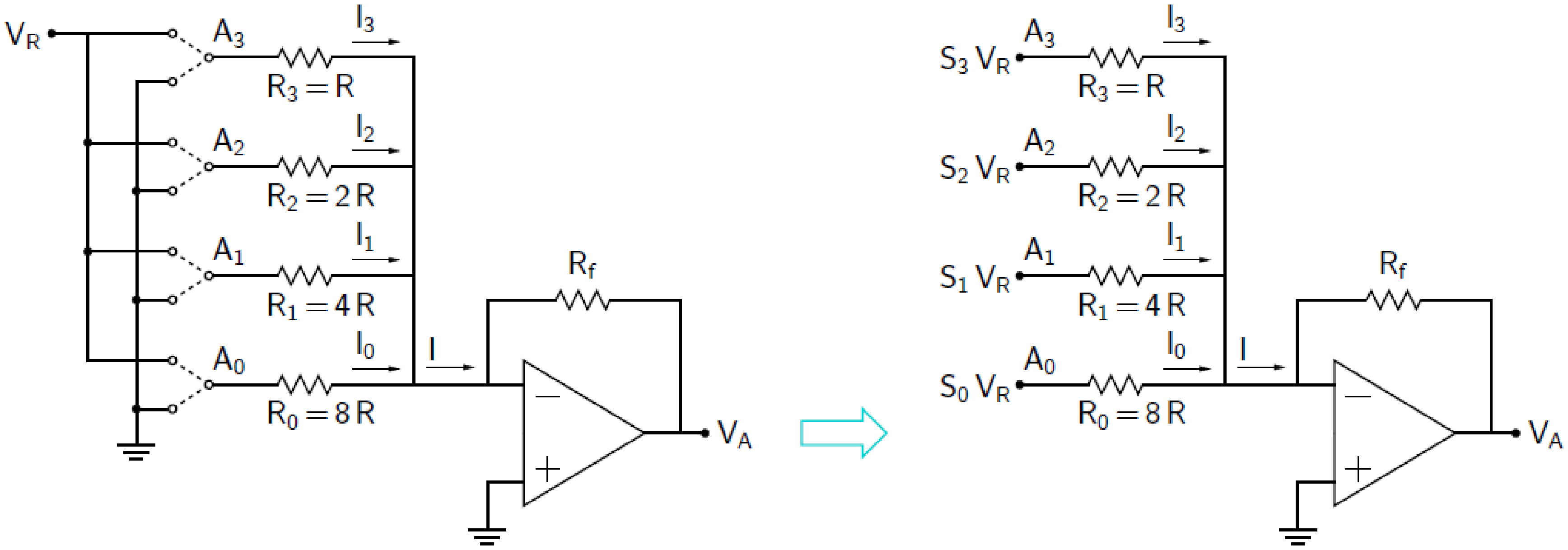
- If the input bit S_k is 1, A_k gets connected to V_R; else, it gets connected to ground.
  $\rightarrow V(A_k) = S_k \times V_R$.

DAC using binary-weighted resistors



- ∗ If the input bit S_k is 1, A_k gets connected to V_R; else, it gets connected to ground.
  $\rightarrow V(A_k) = S_k \times V_R$.
- ∗ Since the inverting terminal of the Op Amp is at virtual ground,
  $I_k = \dfrac{V(A_k) - 0}{R_k} = \dfrac{S_k\, V_R}{R_k}$.

DAC using binary-weighted resistors

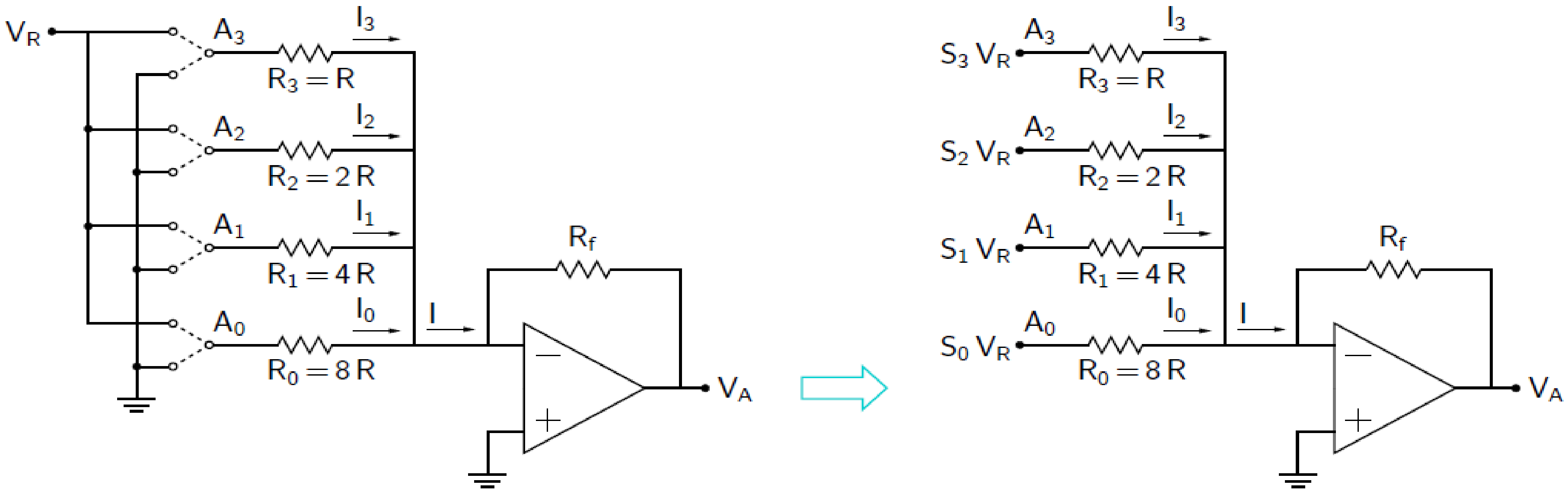


- If the input bit S_k is 1, A_k gets connected to V_R; else, it gets connected to ground.
  $\rightarrow V(A_k) = S_k \times V_R$.

- Since the inverting terminal of the Op Amp is at virtual ground,
  $I_k = \dfrac{V(A_k) - 0}{R_k} = \dfrac{S_k\, V_R}{R_k}$.

- Using $R_k = 2^{N-1}\, R/2^k$, we get $I = \dfrac{V_R}{2^{N-1} R} \displaystyle\sum_{0}^{N-1} S_k \times 2^k$ ($N = 4$ here).

DAC using binary-weighted resistors

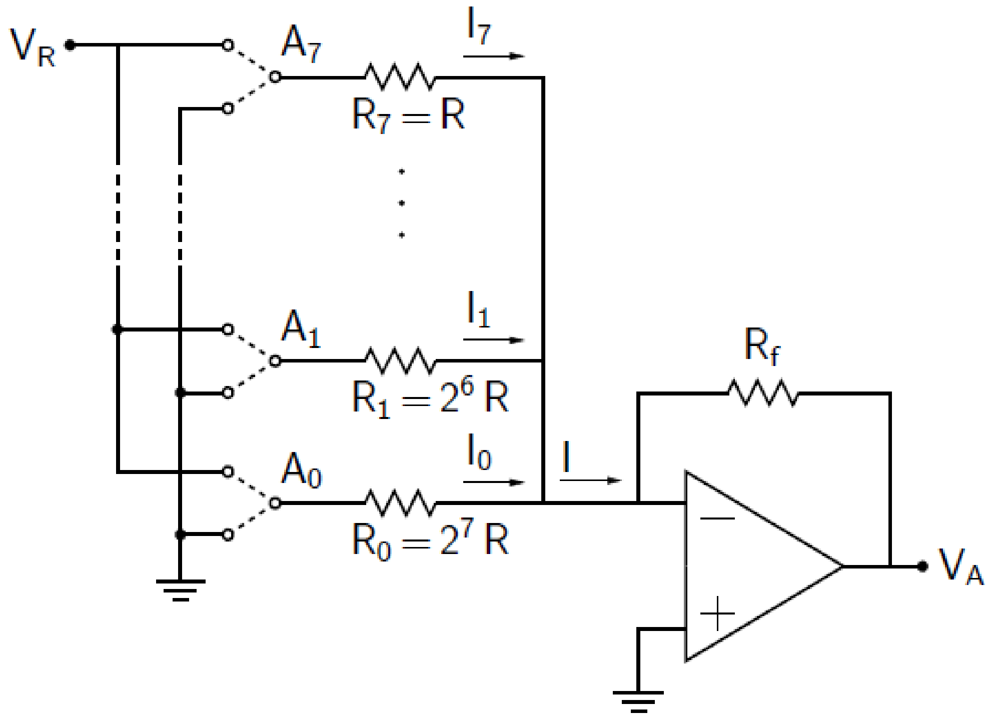


- If the input bit S_k is 1, A_k gets connected to V_R; else, it gets connected to ground. $\rightarrow V(A_k) = S_k \times V_R$.
- Since the inverting terminal of the Op Amp is at virtual ground, $I_k = \dfrac{V(A_k) - 0}{R_k} = \dfrac{S_k\, V_R}{R_k}$.
- Using $R_k = 2^{N-1}\, R/2^k$, we get $I = \dfrac{V_R}{2^{N-1} R} \displaystyle\sum_{0}^{N-1} S_k \times 2^k$ ($N = 4$ here).
- The output voltage is $V_o = -R_f\, I = -V_R\, \dfrac{R_f}{2^{N-1} R} \displaystyle\sum_{0}^{N-1} S_k \times 2^k$.

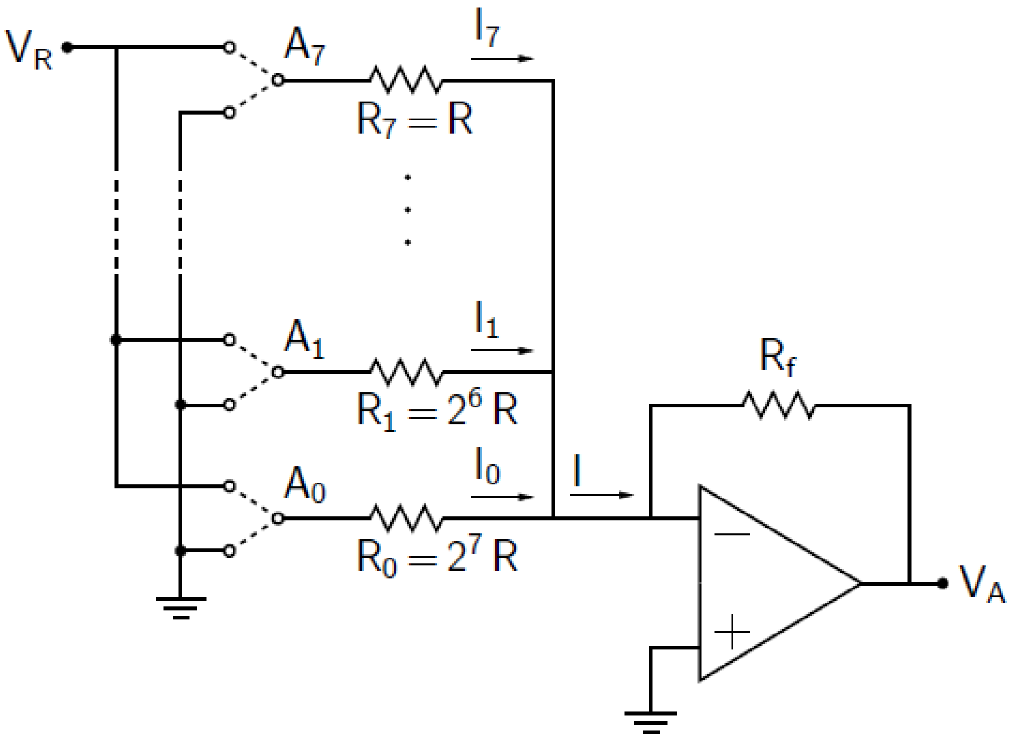
Lecture 26

EC103

DAC using binary-weighted resistors: Example

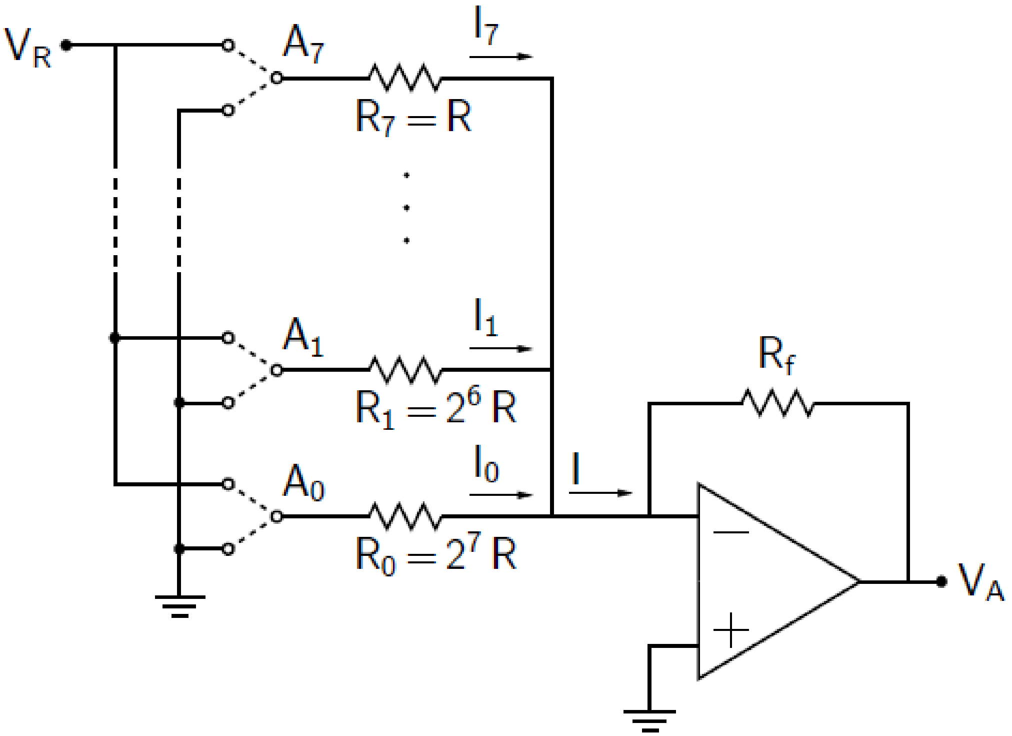


- Consider an 8-bit DAC with $V_R = 5\,\text{V}$. What is the smallest value of R which will limit the current drawn from the supply (V_R) to 10 mA?



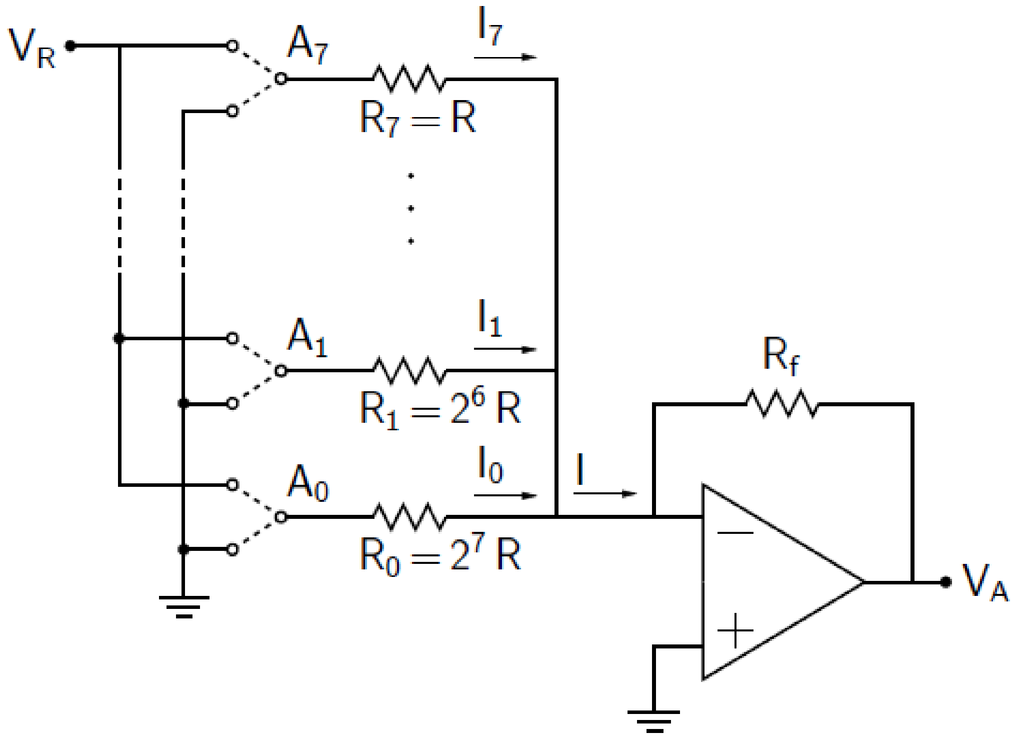
∗ Consider an 8-bit DAC with $V_R = 5\,\text{V}$. What is the smallest value of R which will limit the current drawn from the supply (V_R) to 10 mA?

Maximum current is drawn from V_R when the input is 1111 1111.



- Consider an 8-bit DAC with $V_R = 5\,\text{V}$. What is the smallest value of R which will limit the current drawn from the supply (V_R) to $10\,\text{mA}$?

  Maximum current is drawn from V_R when the input is 1111 1111.
  → All nodes A_0 to A_7 get connected to V_R.

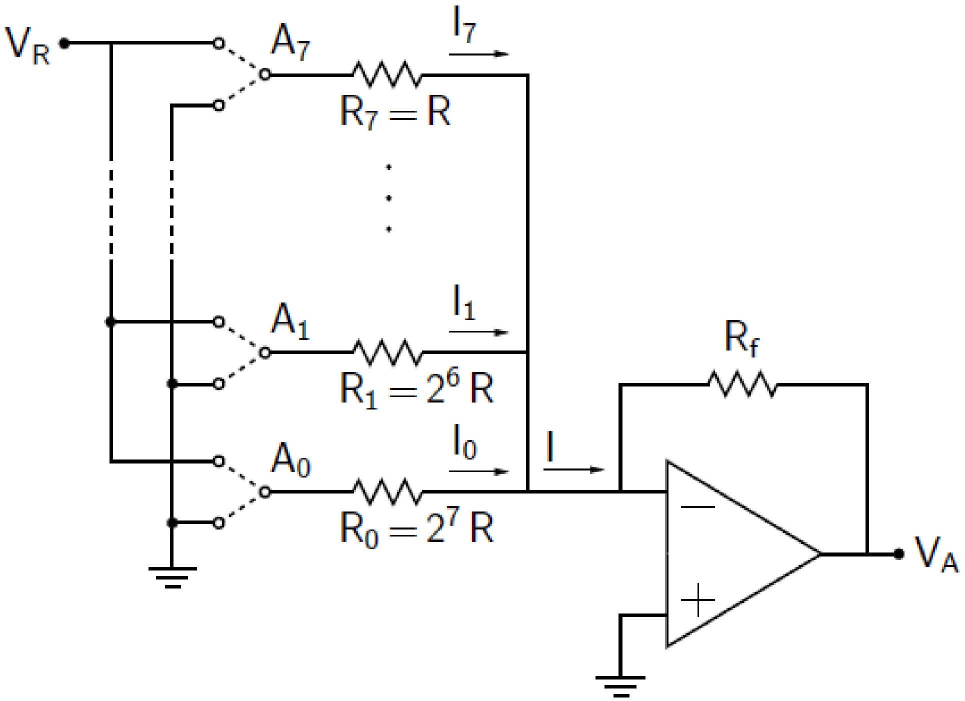


- ∗ Consider an 8-bit DAC with $V_R = 5\,\text{V}$. What is the smallest value of R which will limit the current drawn from the supply (V_R) to 10 mA?

Maximum current is drawn from V_R when the input is 1111 1111.
→ All nodes A_0 to A_7 get connected to V_R.

$$\rightarrow 10\,\text{mA} = \frac{V_R}{R} + \frac{V_R}{2R} + \cdots + \frac{V_R}{2^7 R} = \frac{1}{2^7}\,\frac{V_R}{R}\,\left(2^0 + 2^1 + \cdots + 2^7\right)$$

$$= \frac{1}{2^7}\,\frac{V_R}{R}\,\left(2^8 - 1\right) = \frac{255}{128}\,\frac{V_R}{R}$$



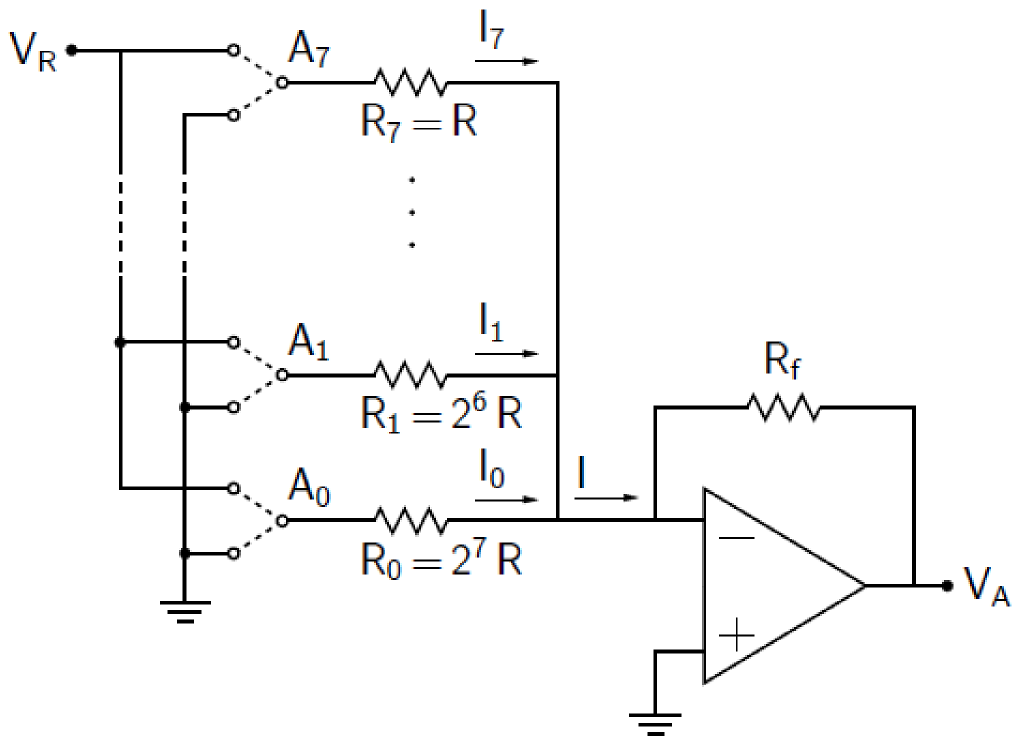
- * Consider an 8-bit DAC with $V_R = 5\,\text{V}$. What is the smallest value of R which will limit the current drawn from the supply (V_R) to $10\,\text{mA}$?

Maximum current is drawn from V_R when the input is 1111 1111.
→ All nodes A_0 to A_7 get connected to V_R.

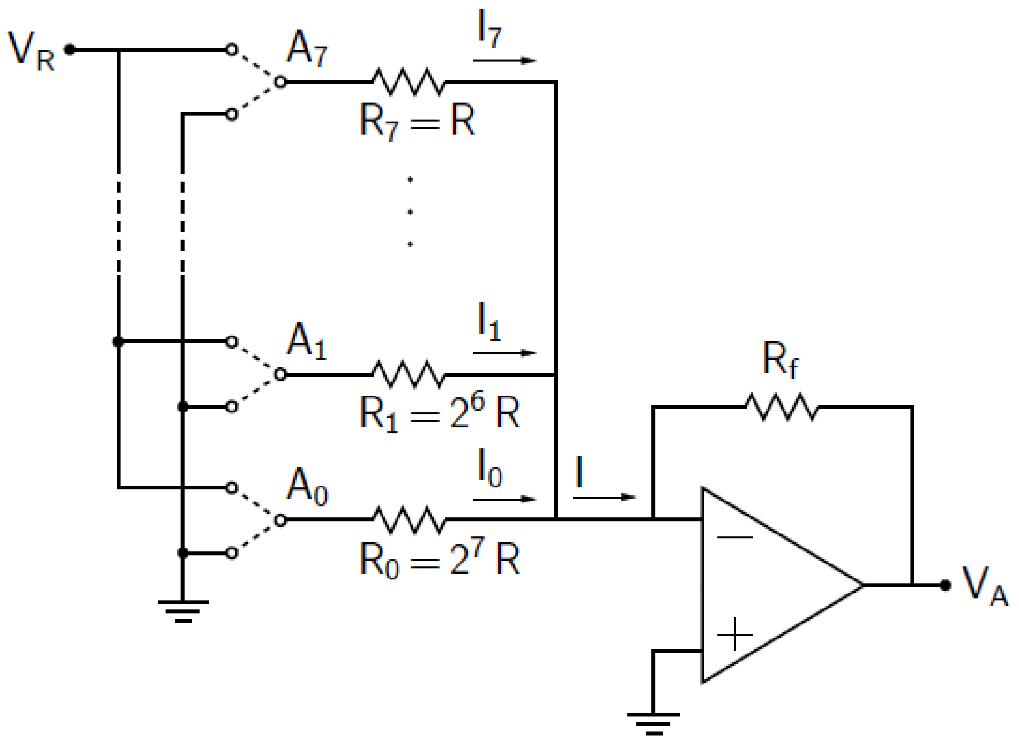
$$\rightarrow 10\,\text{mA} = \frac{V_R}{R} + \frac{V_R}{2R} + \cdots + \frac{V_R}{2^7 R} = \frac{1}{2^7}\,\frac{V_R}{R}\left(2^0 + 2^1 + \cdots + 2^7\right)$$

$$= \frac{1}{2^7}\,\frac{V_R}{R}\left(2^8 - 1\right) = \frac{255}{128}\,\frac{V_R}{R}$$

$$\rightarrow R_{\text{min}} = \frac{5\,\text{V}}{10\,\text{mA}} \times \frac{255}{128} = 996\,\Omega\,.$$

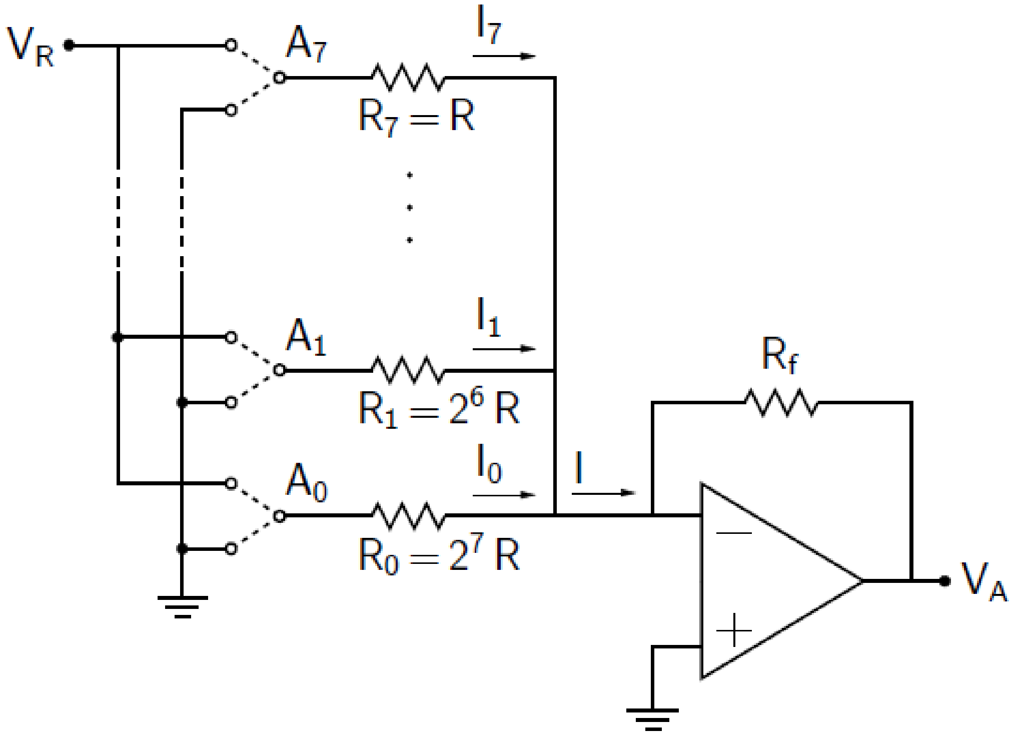


- ∗ If $R_f = R$, what is the resolution (i.e., ΔV_A corresponding to the input LSB changing from 0 to 1 with other input bits constant)?



* If $R_f = R$, what is the resolution (i.e., ΔV_A corresponding to the input LSB changing from 0 to 1 with other input bits constant)?

$$V_A = -V_R \,\frac{R_f}{2^{N-1}R}\,\left[S_7 2^7 + \cdots + S_1 2^1 + S_0 2^0\right]$$

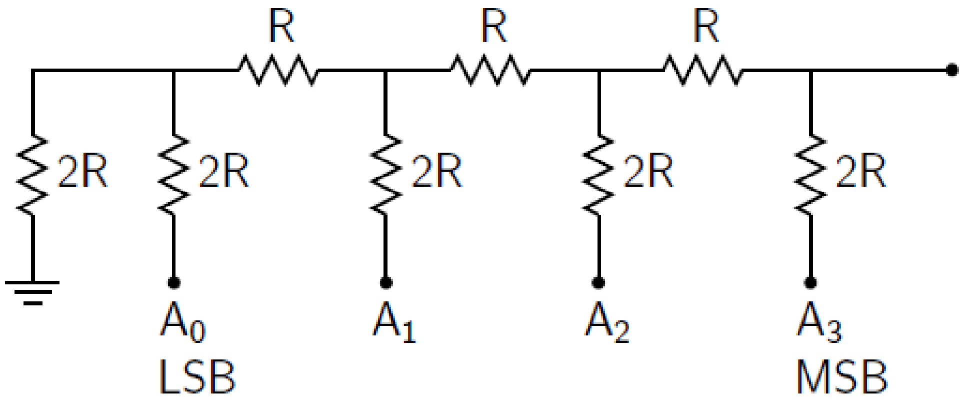


- ∗ If $R_f = R$, what is the resolution (i.e., ΔV_A corresponding to the input LSB changing from 0 to 1 with other input bits constant)?

$$V_A = -V_R \, \frac{R_f}{2^{N-1}R} \left[S_7 2^7 + \cdots + S_1 2^1 + S_0 2^0 \right]$$

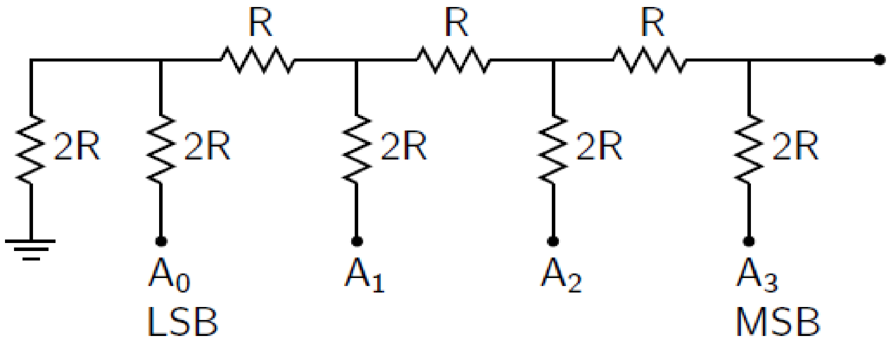
$$\rightarrow \Delta V_A = \frac{V_R}{2^{N-1}} \, \frac{R_f}{R} = \frac{5\,\text{V}}{2^{8-1}} \times 1 = \frac{5}{128} = 0.0391\,\text{V}.$$

R-2R ladder network



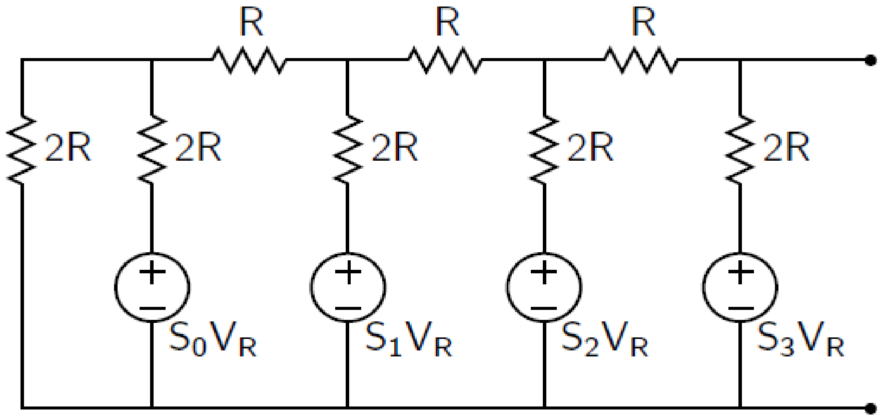
Node A_k is connected to V_R if input bit S_k is 1; else, it is connected to ground.

R-2R ladder network

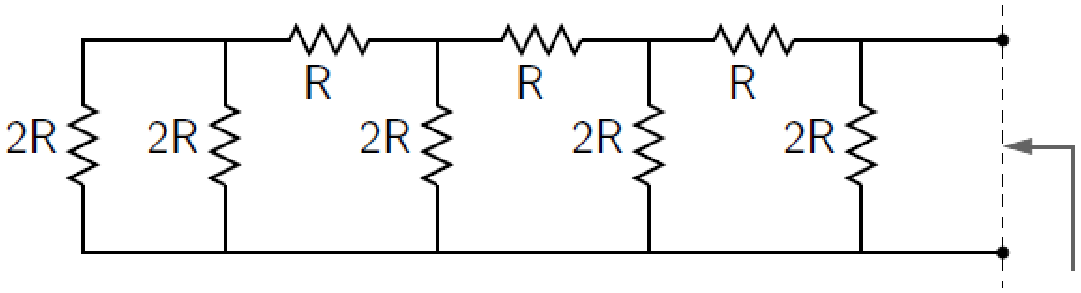


Node A_k is connected to V_R if input bit S_k is 1; else, it is connected to ground.

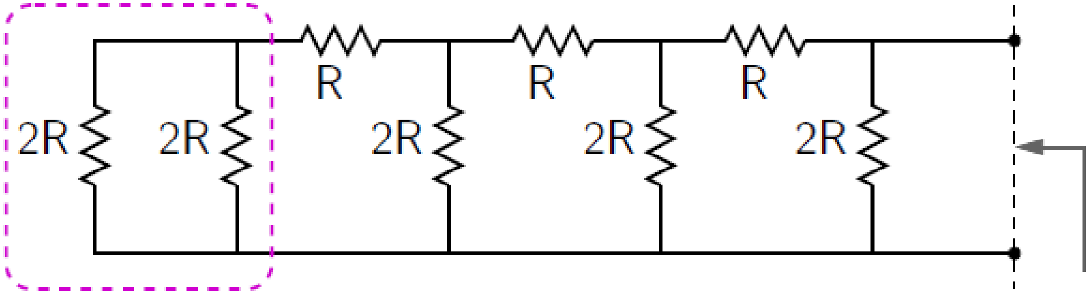
The original network is equivalent to



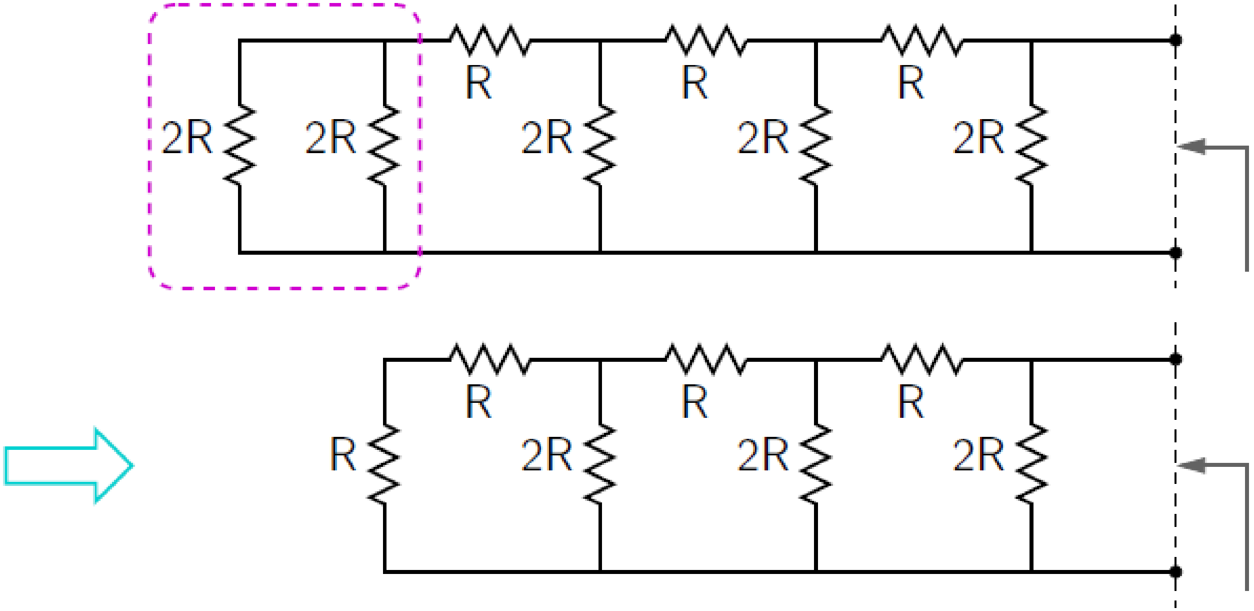
R-2R ladder network: Thevenin resistance



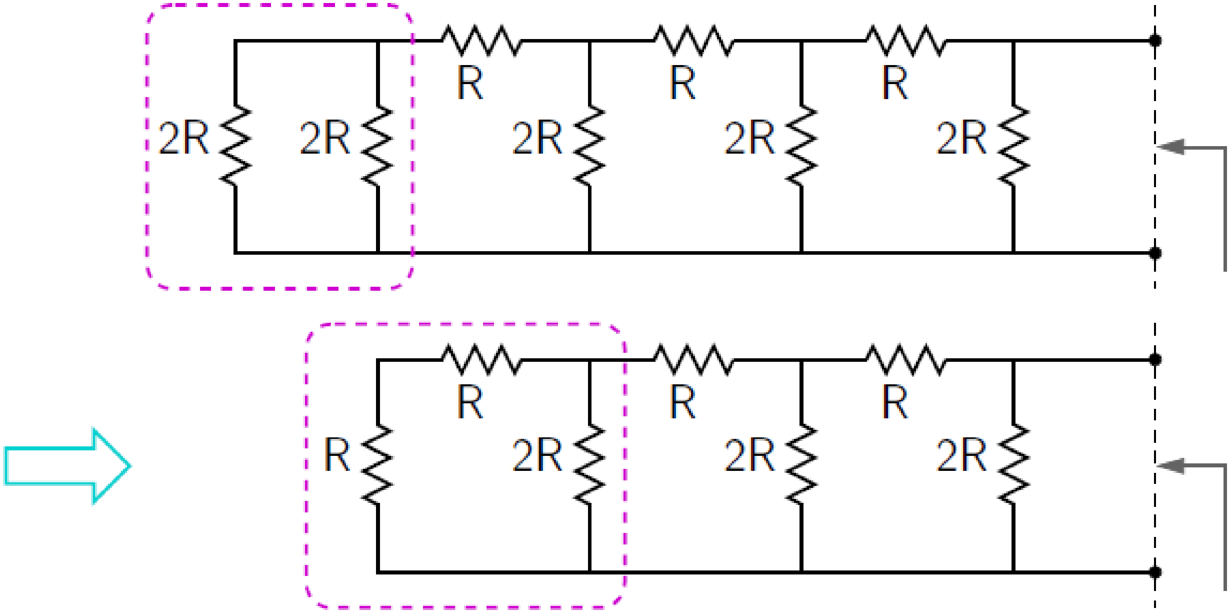
R-2R ladder network: Thevenin resistance



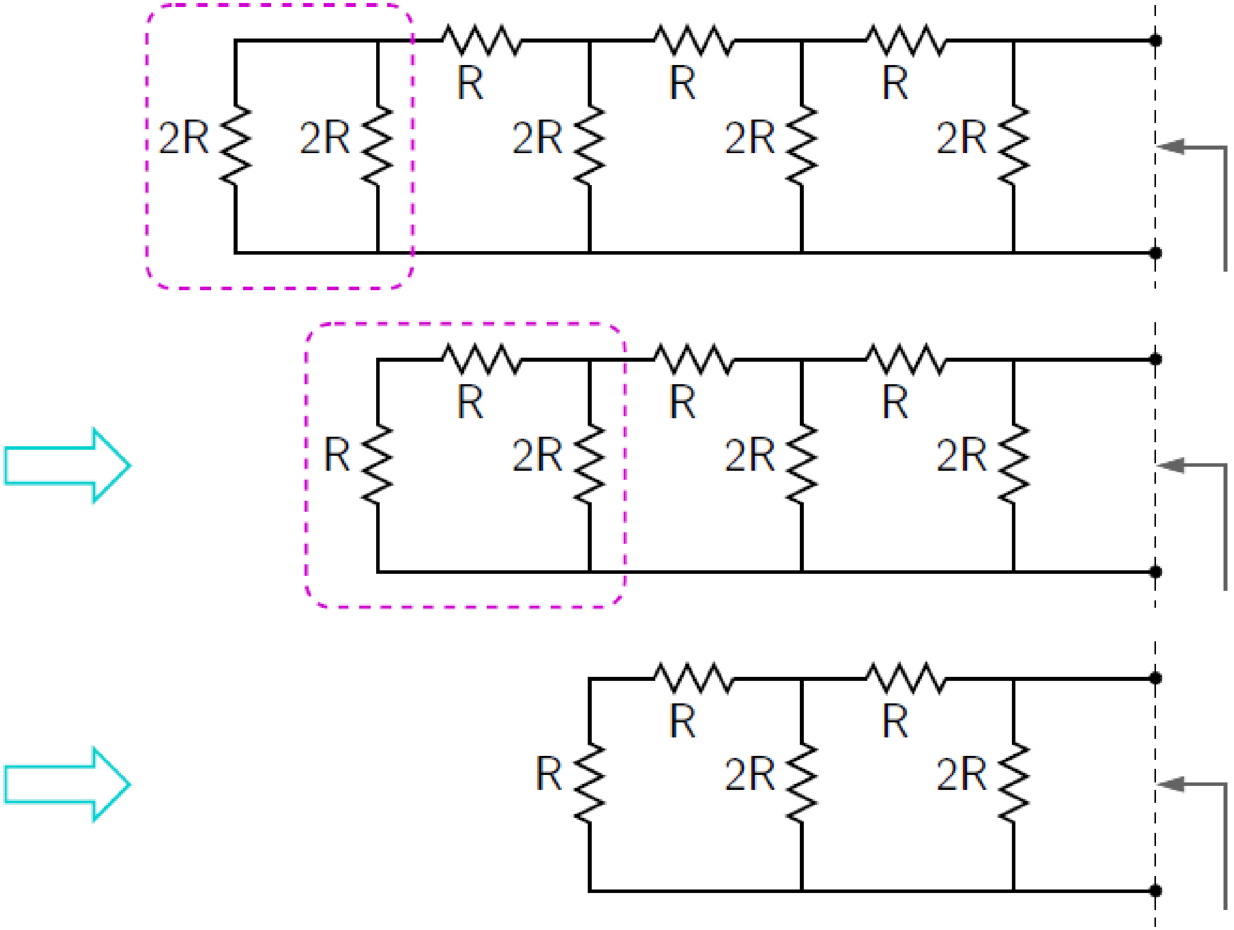
R-2R ladder network: Thevenin resistance



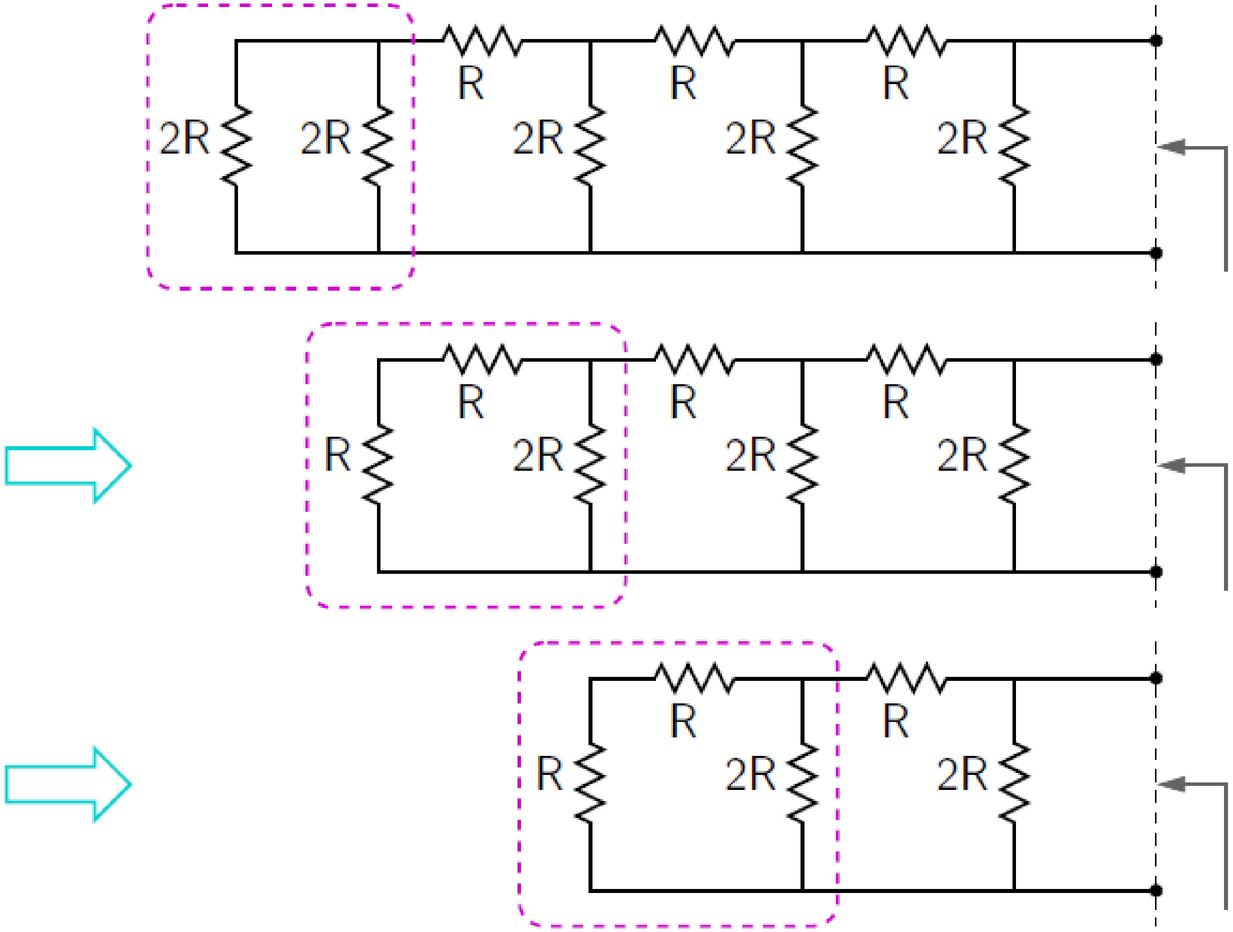
R-2R ladder network: Thevenin resistance



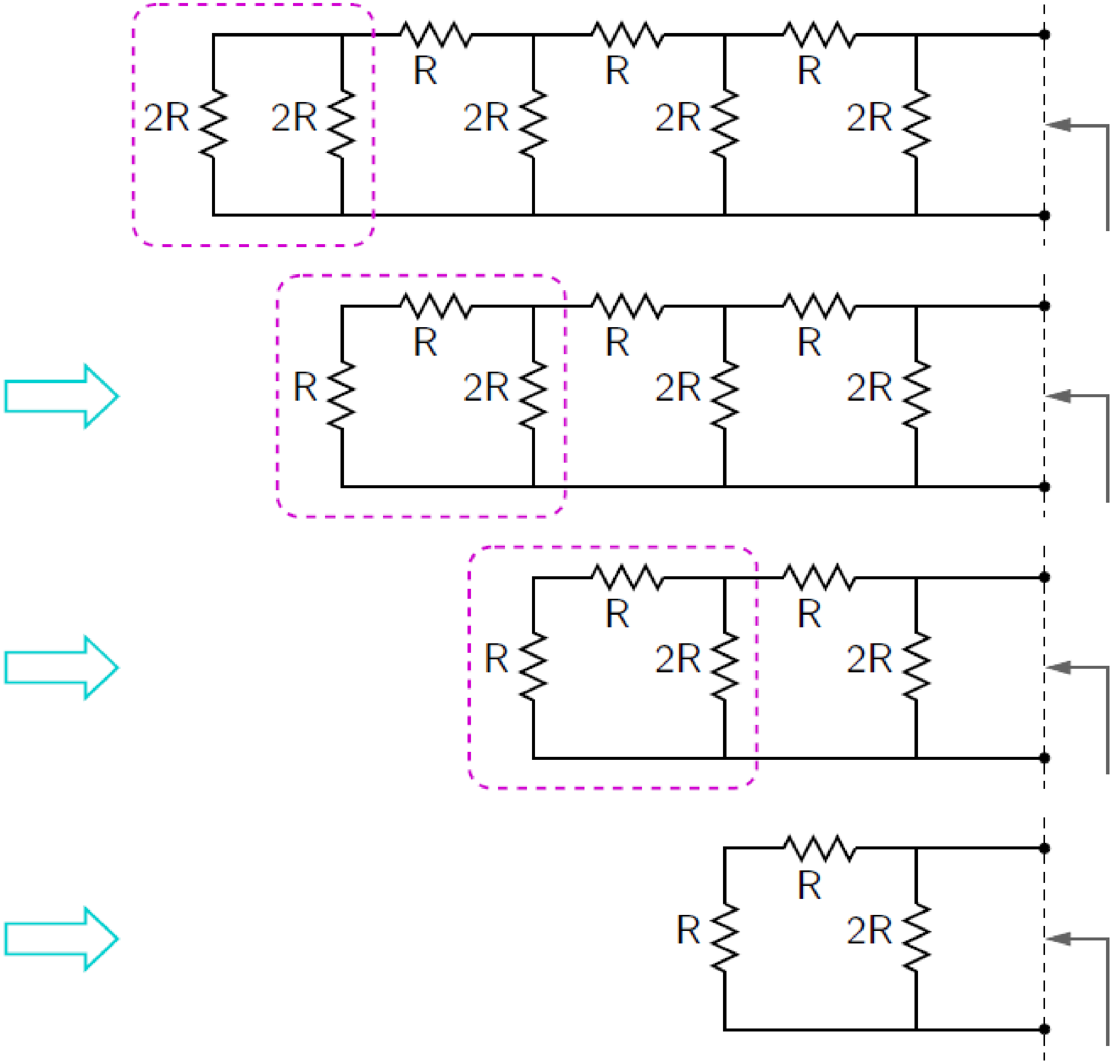
R-2R ladder network: Thevenin resistance



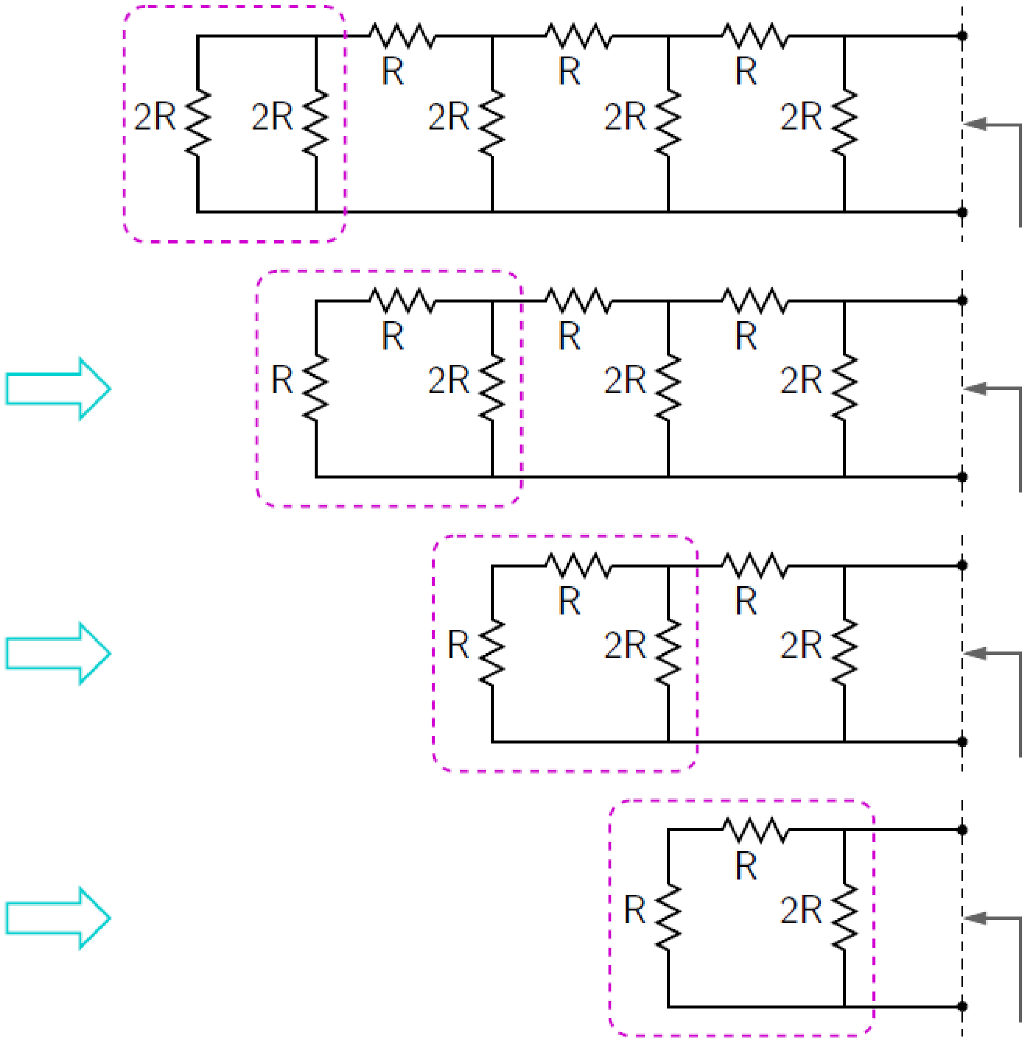
R-2R ladder network: Thevenin resistance



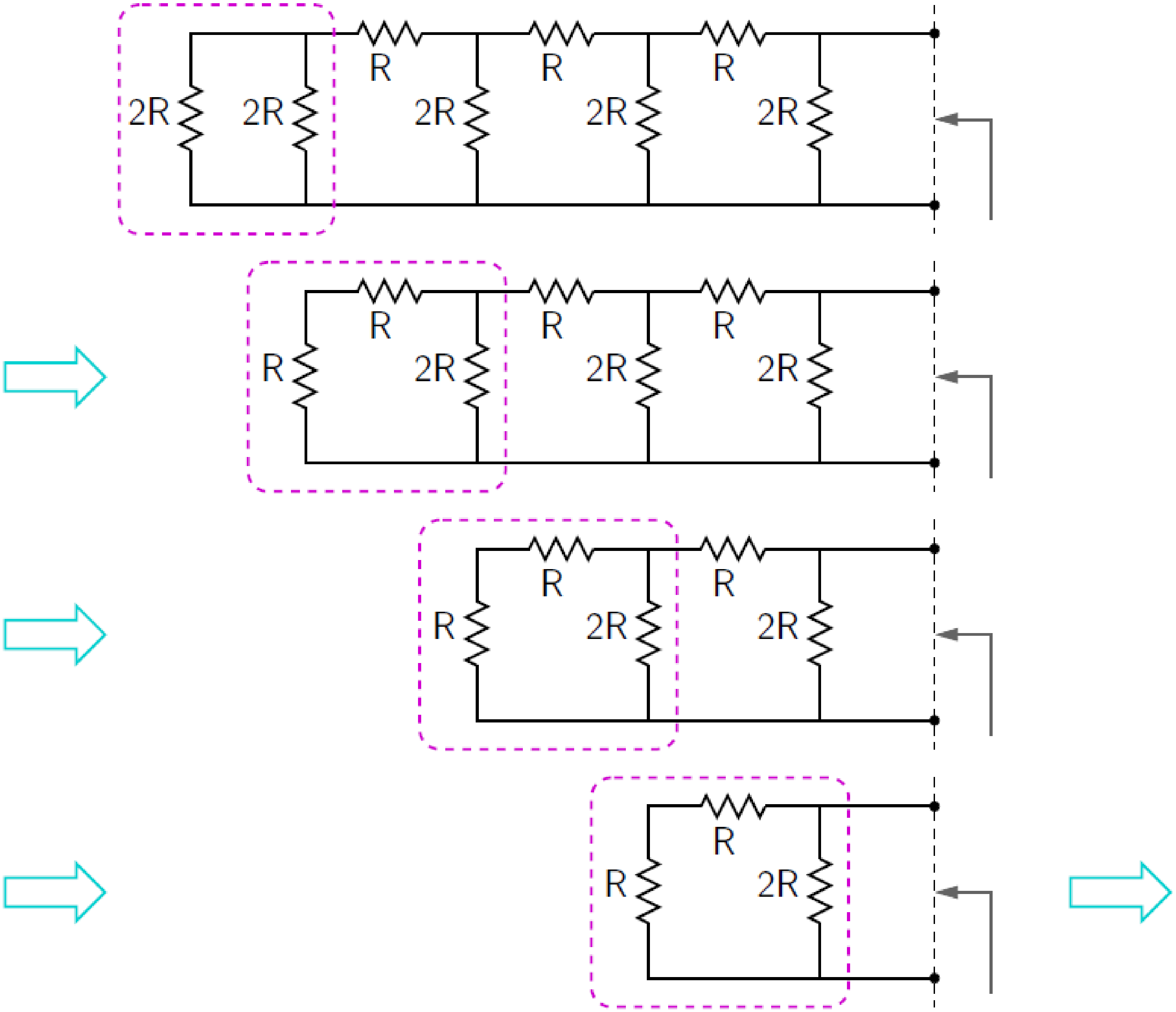
R-2R ladder network: Thevenin resistance



R-2R ladder network: Thevenin resistance

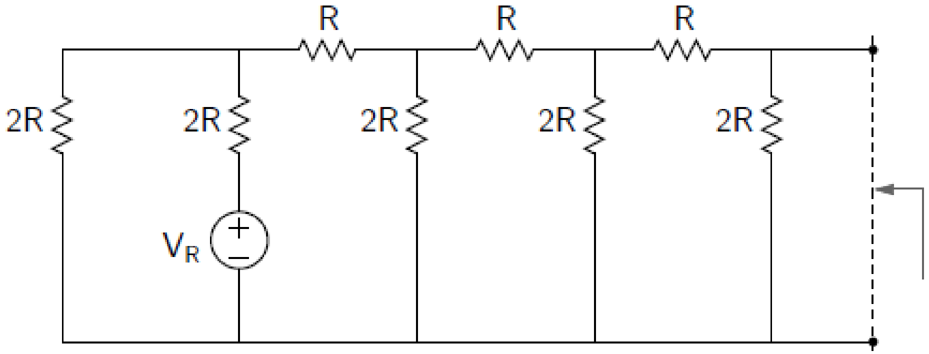


R-2R ladder network: Thevenin resistance

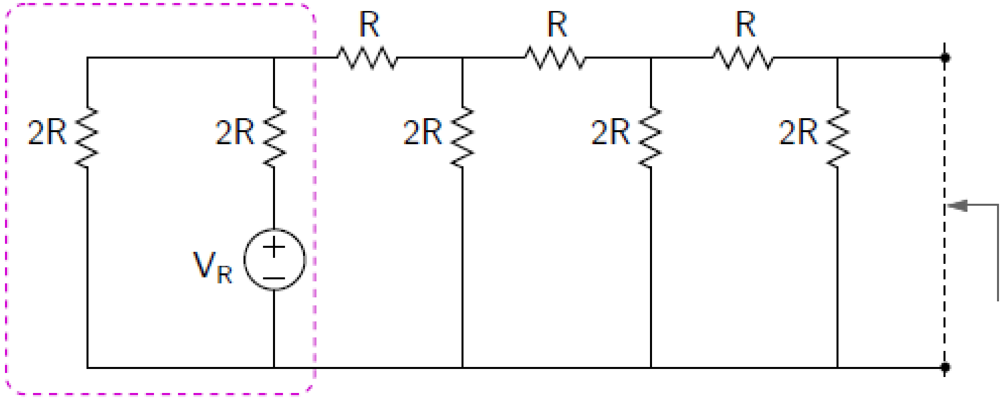


$\boxed{R_{Th} = R}$

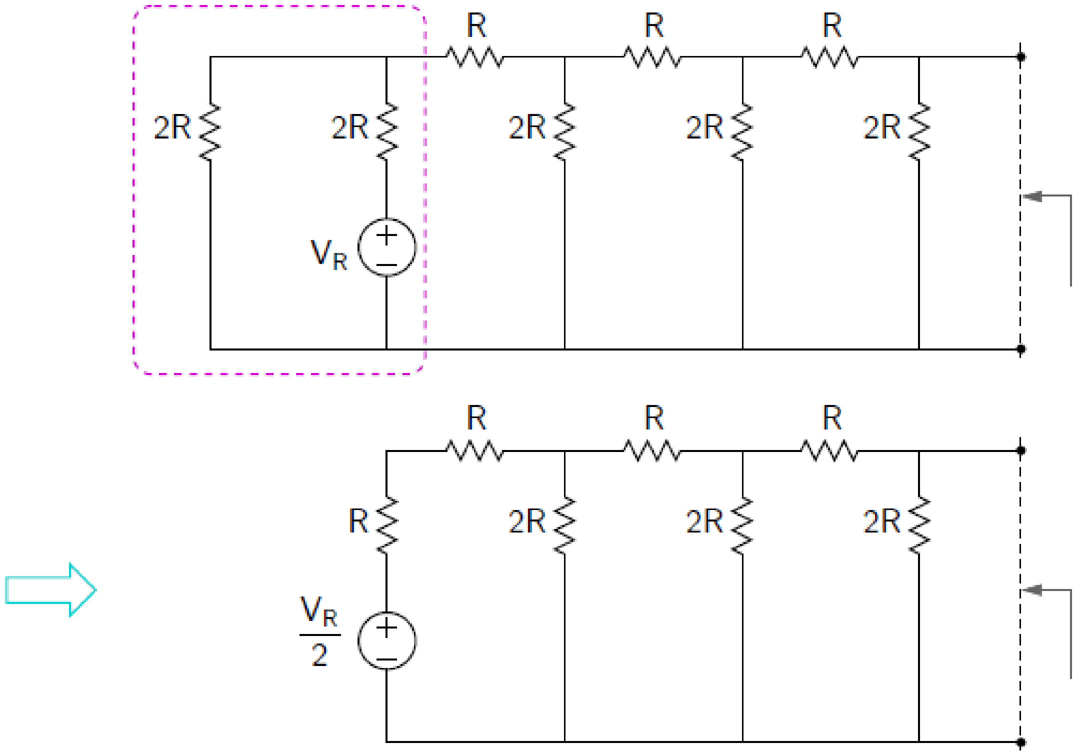
R-2R ladder network: V_{Th} for $S_0 = 1$



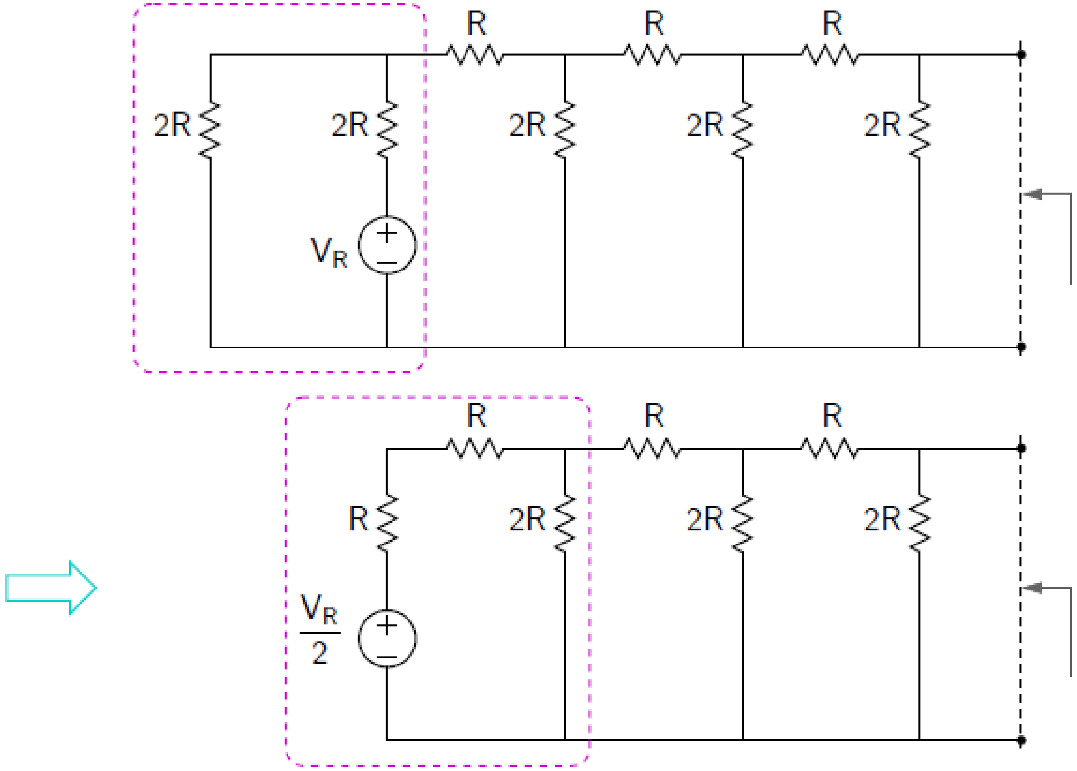
R-2R ladder network: V_{Th} for $S_0 = 1$



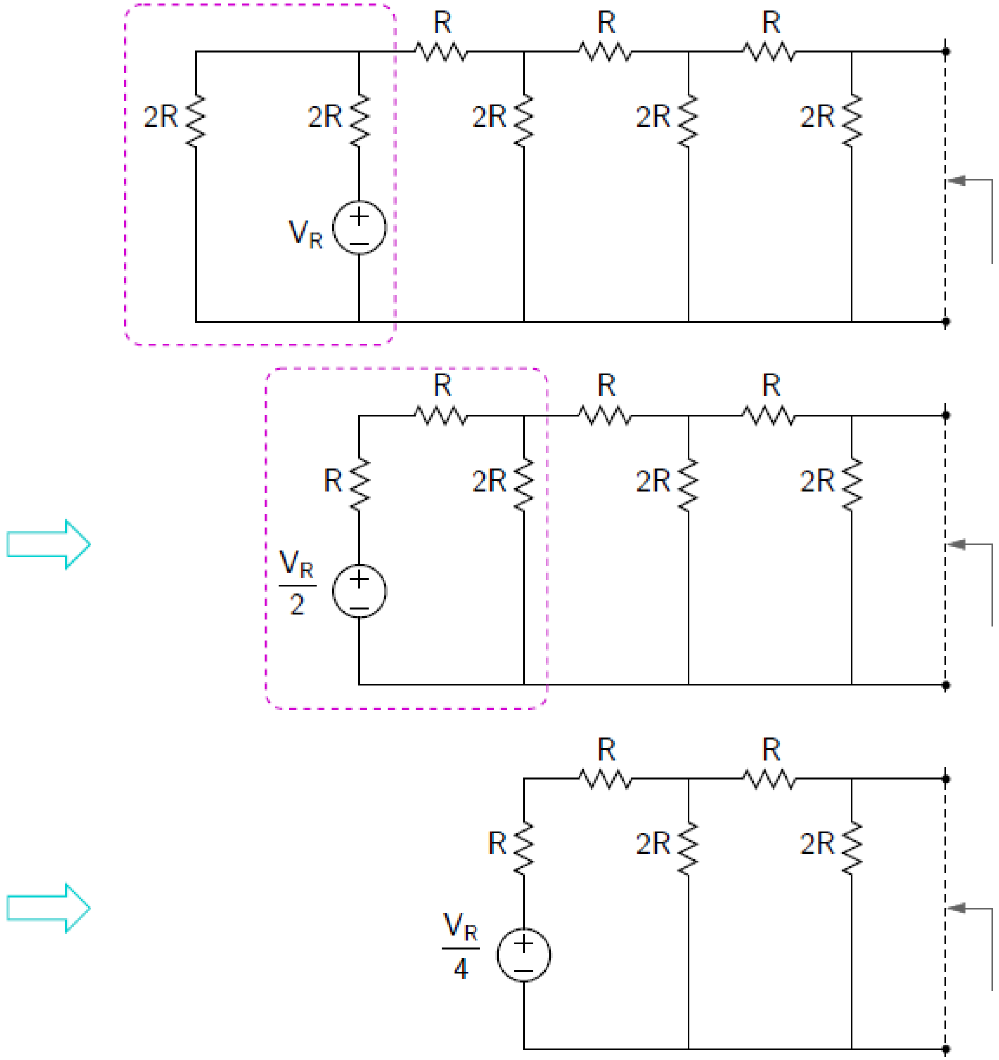
R-2R ladder network: V_{Th} for $S_0 = 1$



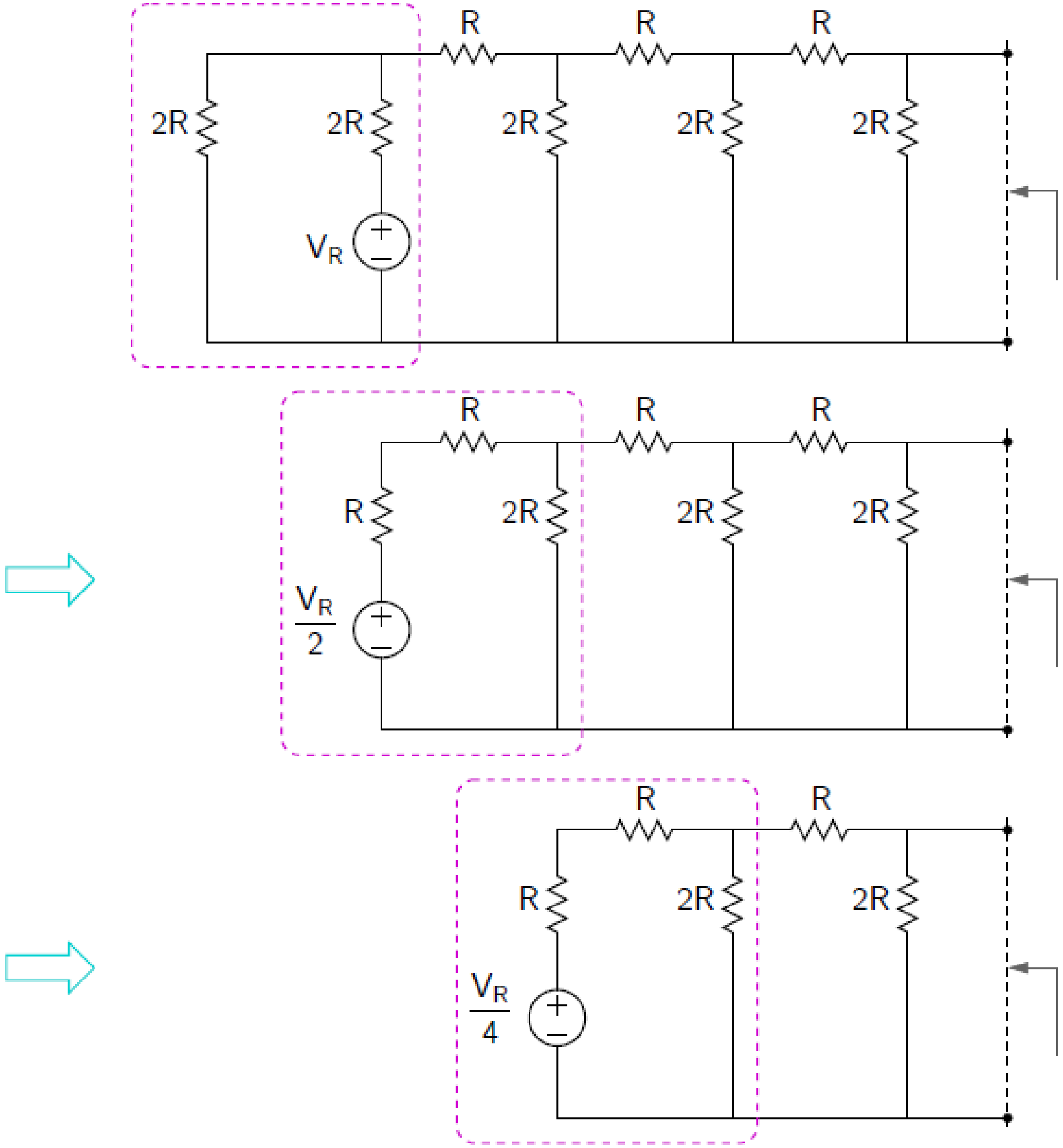
R-2R ladder network: V_{Th} for $S_0 = 1$



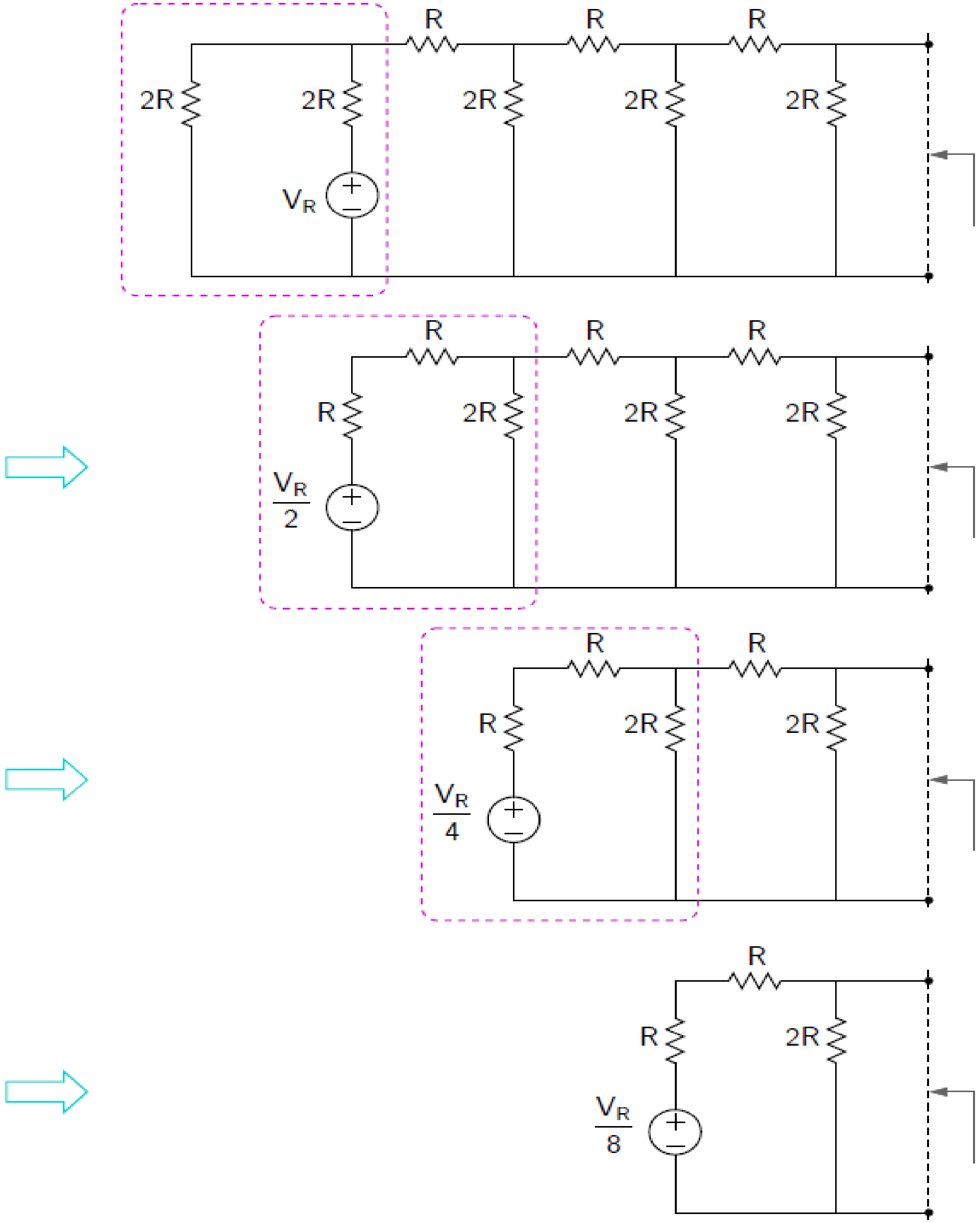
R-2R ladder network: V_{Th} for $S_0 = 1$



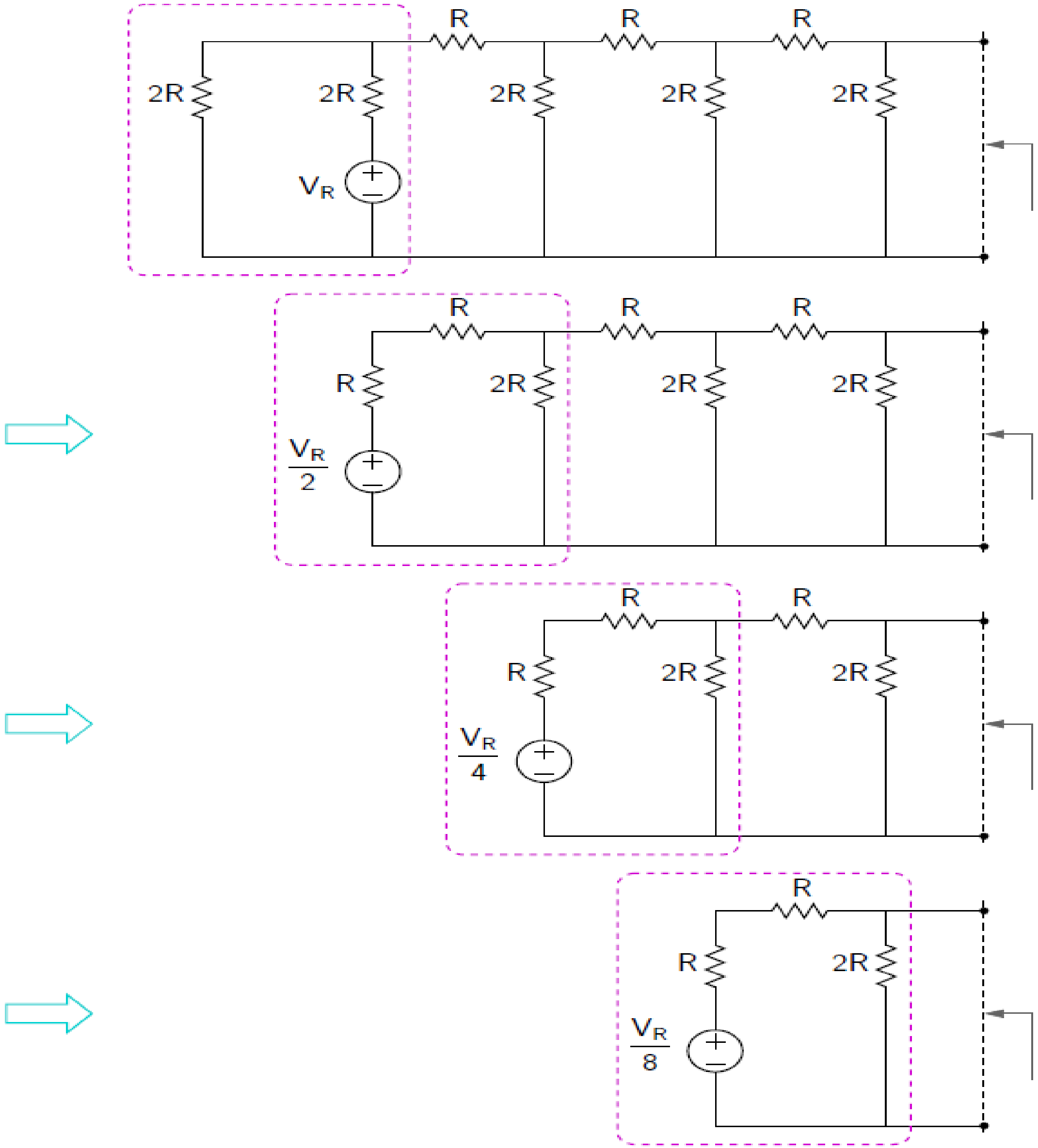
R-2R ladder network: V_{Th} for $S_0 = 1$



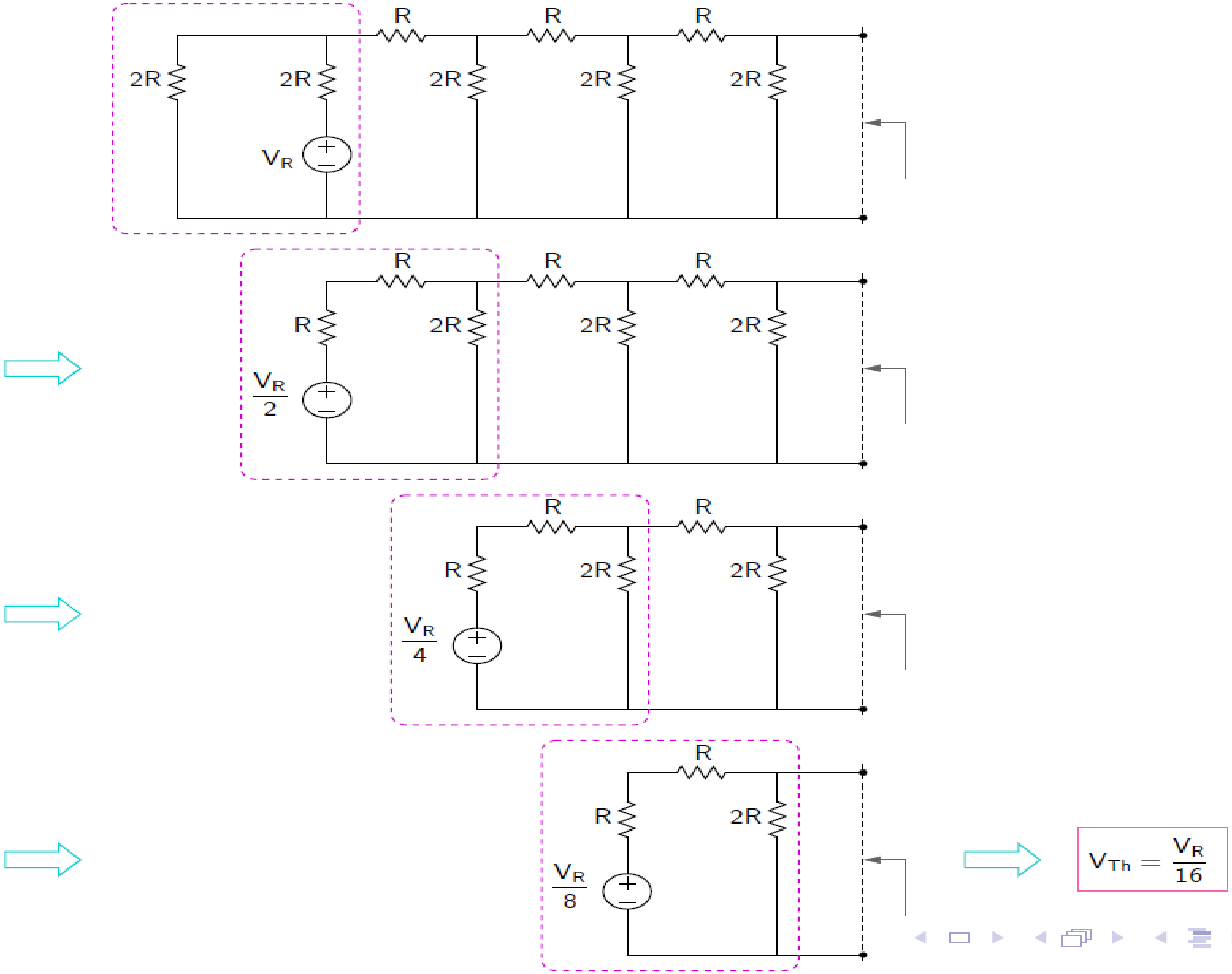
R-2R ladder network: V_{Th} for $S_0 = 1$



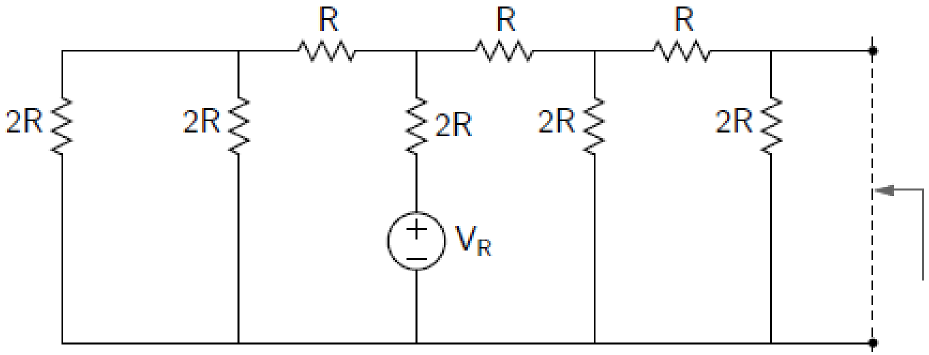
R-2R ladder network: V_{Th} for $S_0 = 1$



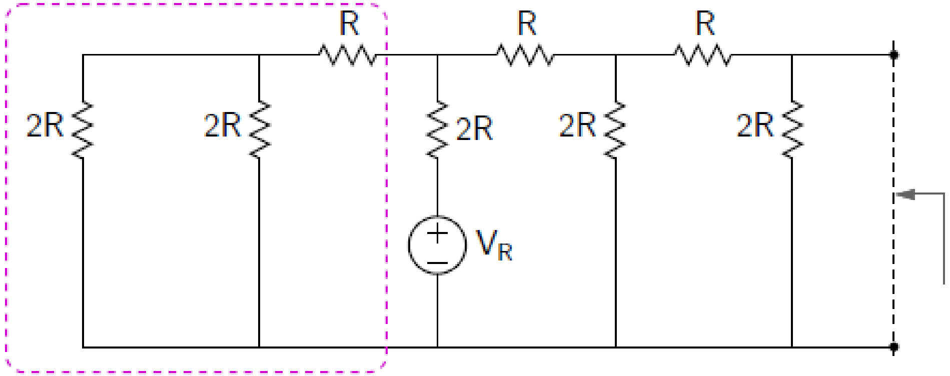
R-2R ladder network: V_{Th} for $S_0 = 1$



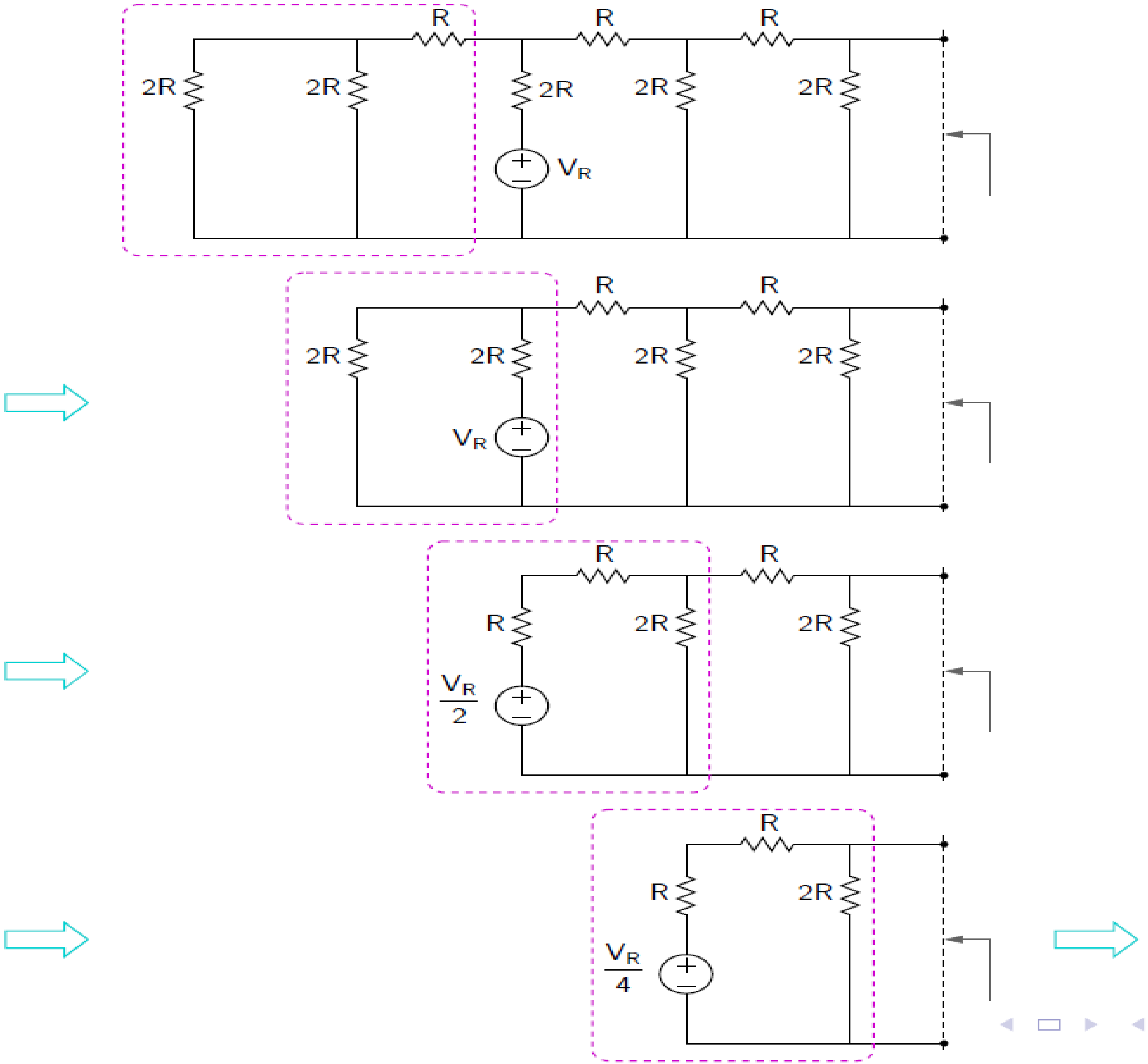
R-2R ladder network: V_{Th} for $S_1 = 1$



R-2R ladder network: V_{Th} for $S_1 = 1$

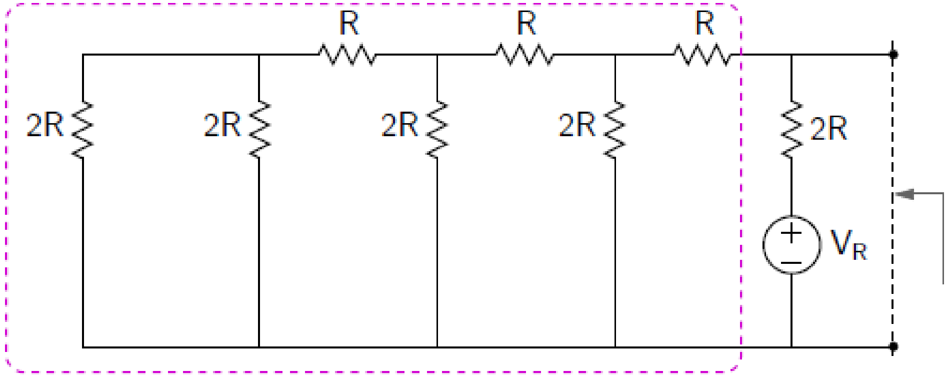


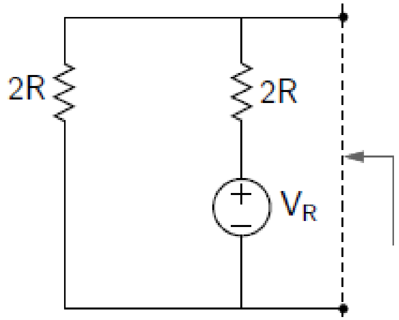
R-2R ladder network: V_{Th} for $S_1 = 1$



$$\boxed{V_{Th} = \frac{V_R}{8}}$$

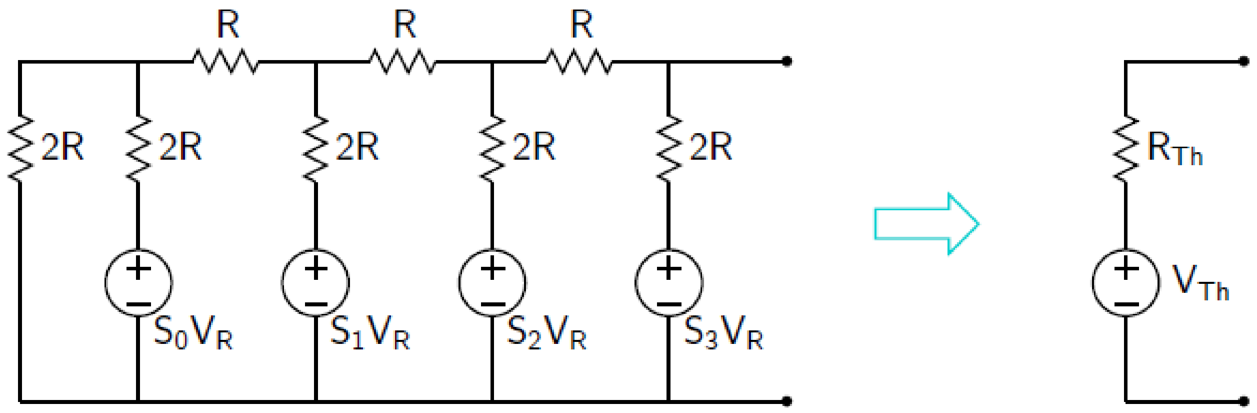
R-2R ladder network: V_{Th} for $S_3 = 1$





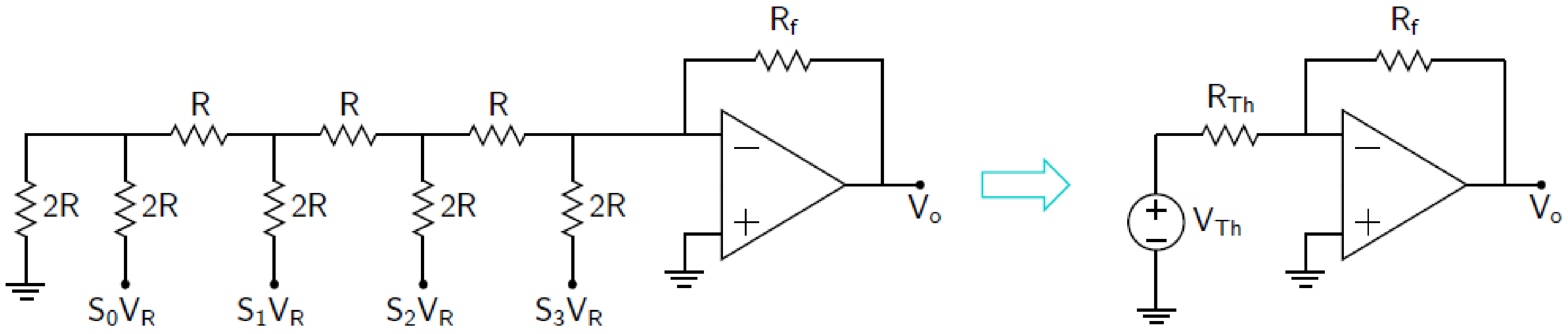
$$\boxed{V_{Th} = \frac{V_R}{2}}$$

R-2R ladder network: R_{Th} and V_{Th}



- $R_{Th} = R\,.$
- $V_{Th} = V_{Th}^{(S0)} + V_{Th}^{(S1)} + V_{Th}^{(S2)} + V_{Th}^{(S3)}$
  $$= \frac{V_R}{16}\left[S_0\,2^0 + S_1\,2^1 + S_2\,2^2 + S_3\,2^3\right]\,.$$
- We can use the *R*-2*R* ladder network and an Op Amp to make up a DAC → next slide.

DAC with R-2R ladder

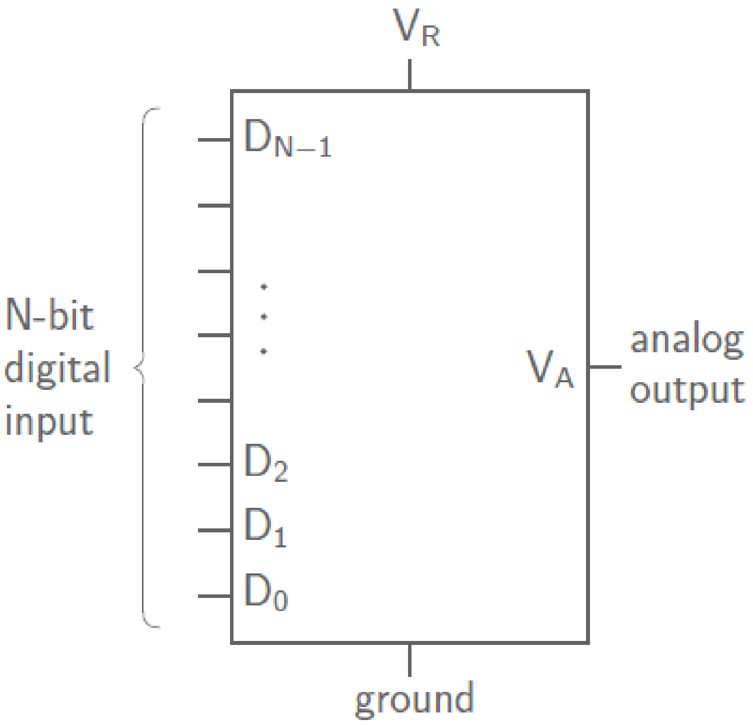


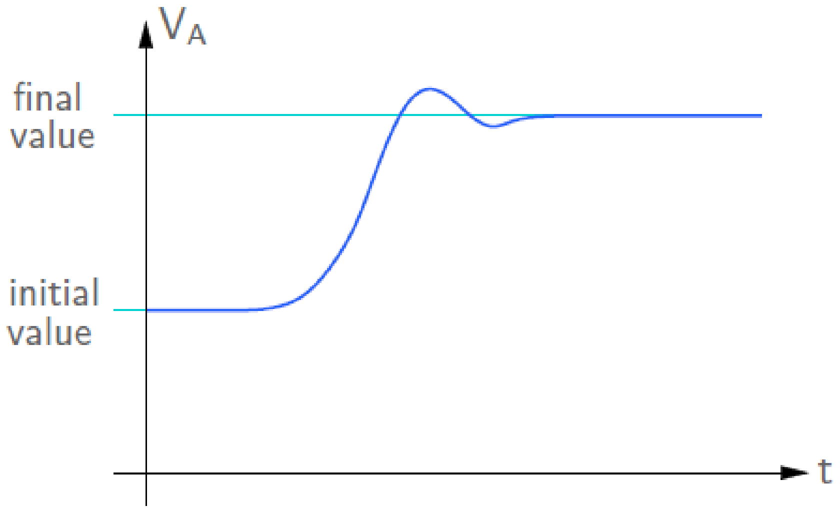
- * $V_o = -\dfrac{R_f}{R_{Th}}\, V_{Th} = -\dfrac{R_f}{R_{Th}}\, \dfrac{V_R}{16}\, \left[S_0\, 2^0 + S_1\, 2^1 + S_2\, 2^2 + S_3\, 2^3\right]$.
- * For an N-bit DAC, $V_o = -\dfrac{R_f}{R_{Th}}\, V_{Th} = -\dfrac{R_f}{R_{Th}}\, \dfrac{V_R}{2^N}\, \sum_0^{N-1} S_k 2^k$.
- * 6- to 20-bit DACs based on the R-2R ladder network are commercially available in monolithic form (single chip).
- * Bipolar, CMOS, or BiCMOS technology is used for these DACs.

Lecture 27

EC103

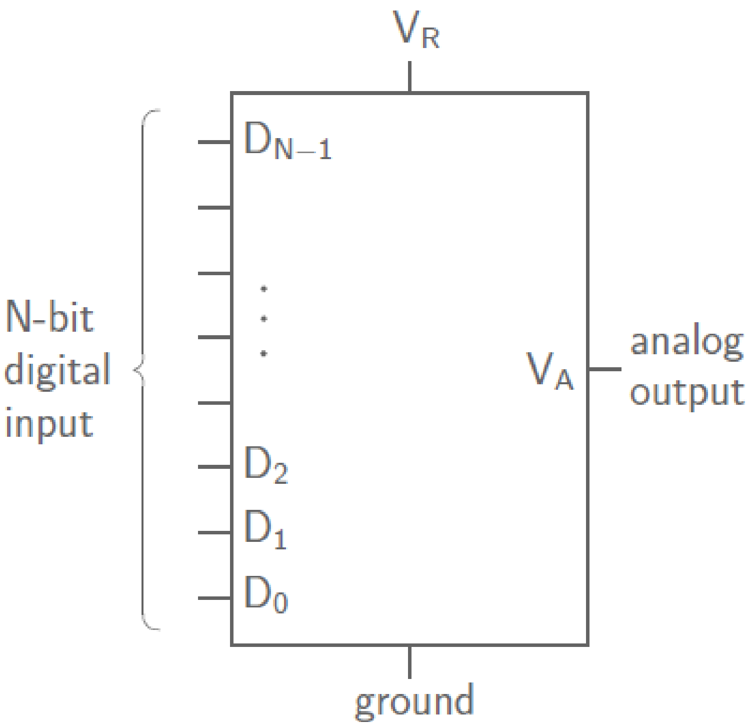
DAC: settling time

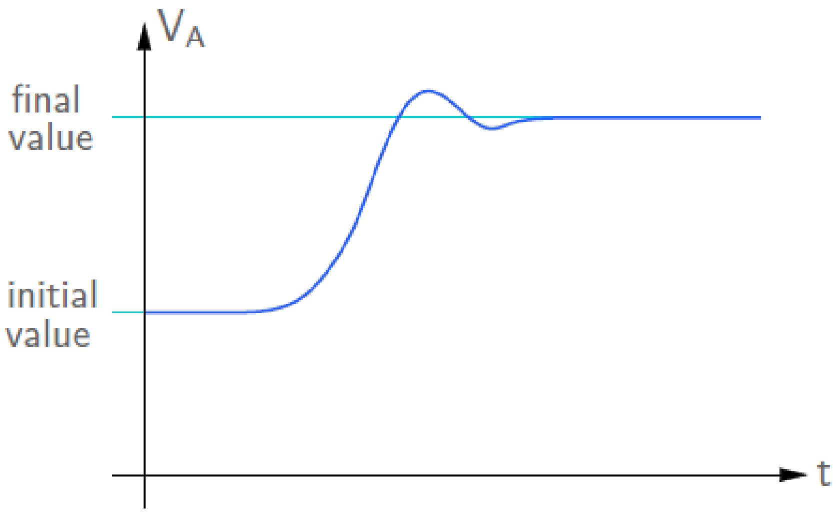




- ∗ When there is a change in the input binary number, the output V_A takes a finite time to settle to the new value.

DAC: settling time





- When there is a change in the input binary number, the output V_A takes a finite time to settle to the new value.
- The finite settling time arises because of stray capacitances and switching delays of the semiconductor devices used within the DAC chip.

DAC: Resolution:

- The resolution of a DAC is the reciprocal of the maximum number of discrete steps in the output.
- Resolution, of course, is dependent on the number of input bits.
- For example, a 4-bit DAC has a resolution of one part in fifteen, i.e., 1/15. That is 100/15% = 6.67%.
- The total number of discrete steps equals to $2^n - 1$, where n is the

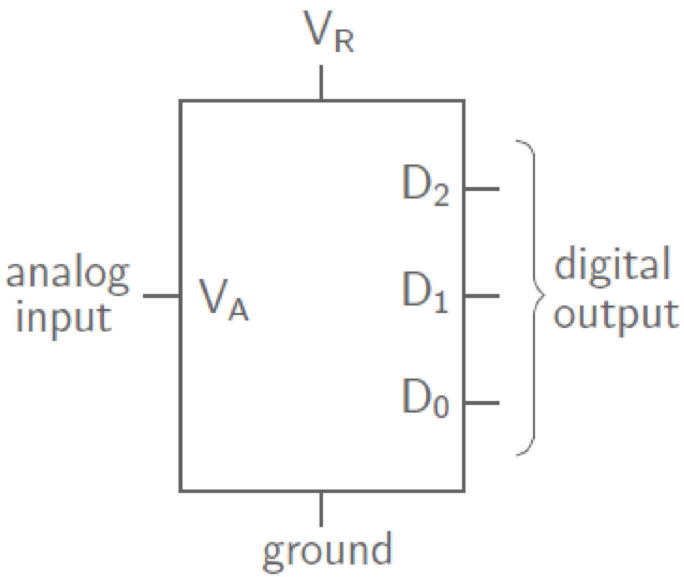
number of bits.

- Resolution can also be expressed as the number of bits that are converted.

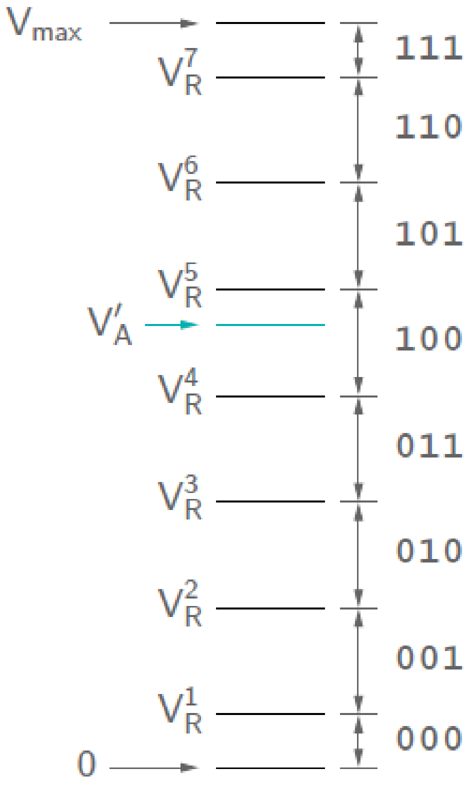
DAC: Accuracy

- Accuracy is a comparison of the actual output of a DAC with the expected output.
- It is expressed as a percentage of a full-scale, or maximum, output voltage.
- For example, if a DAC has a full-scale output of 10 V, and the accuracy is ±0.1%, then the maximum error for any output voltage is 10V X 0.001 = 10 mV.
- Ideally, the accuracy should be no worse than ±0.5 of an LSB.
- For a 8-bit converter, the LSB is 0.39% of the full scale. The accuracy should be approximately ±0.2%.

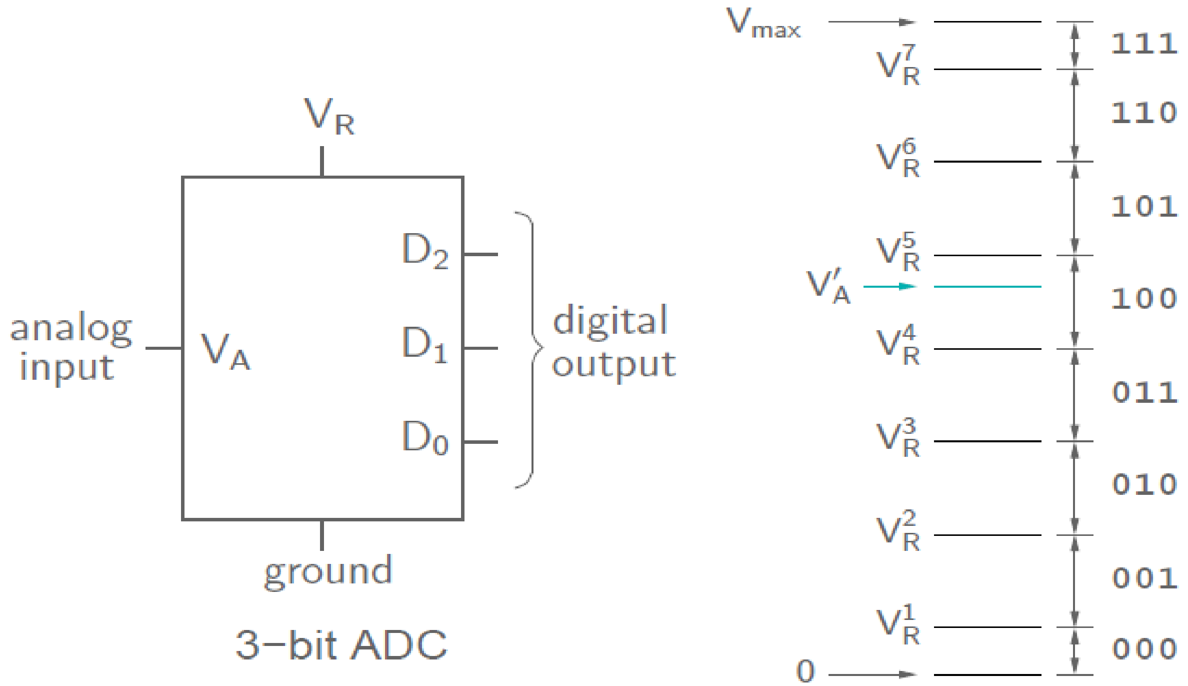
ADC: introduction



3−bit ADC

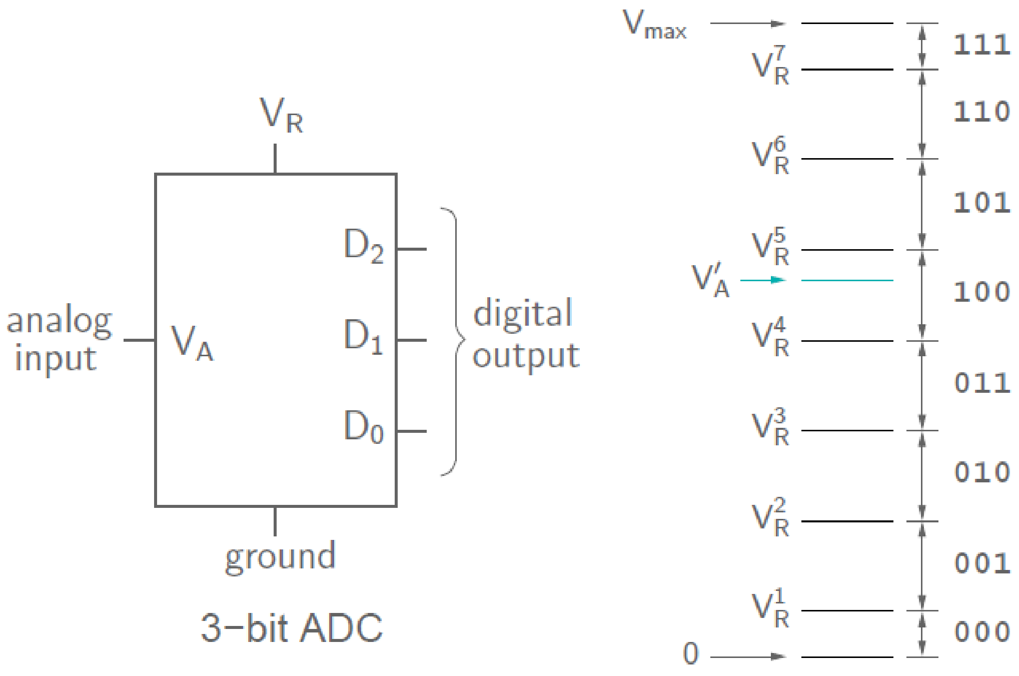


ADC: introduction



* If the input V_A is in the range $V_R^k < V_A < V_R^{k+1}$, the output is the binary number corresponding to the integer k. For example, for $V_A = V'_A$, the output is 100.

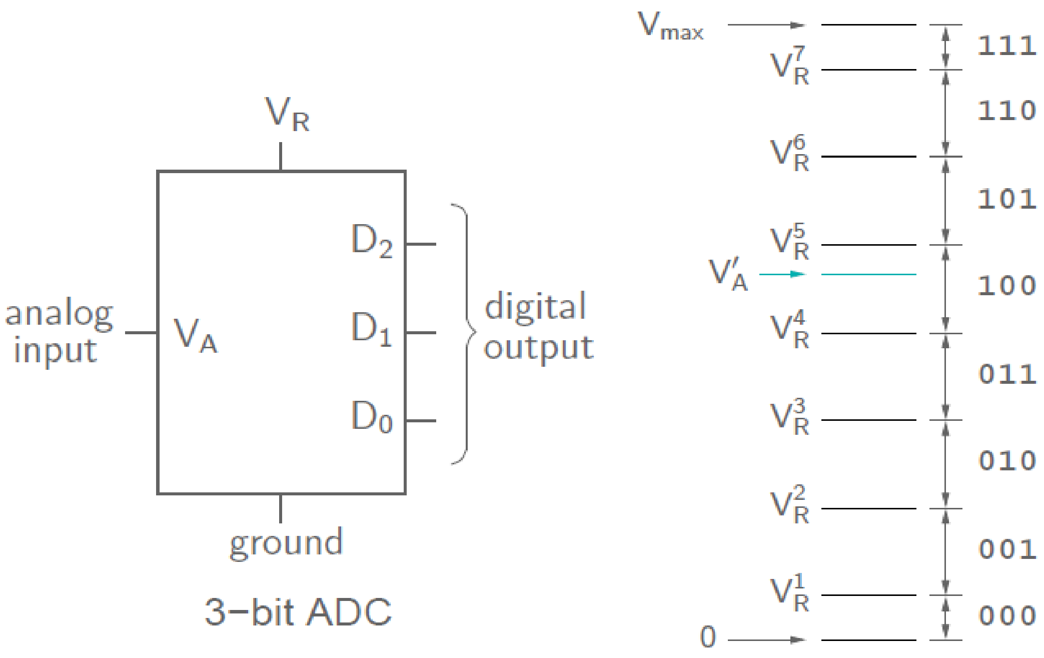
ADC: introduction



- If the input V_A is in the range $V_R^k < V_A < V_R^{k+1}$, the output is the binary number corresponding to the integer k. For example, for $V_A = V_A'$, the output is 100.

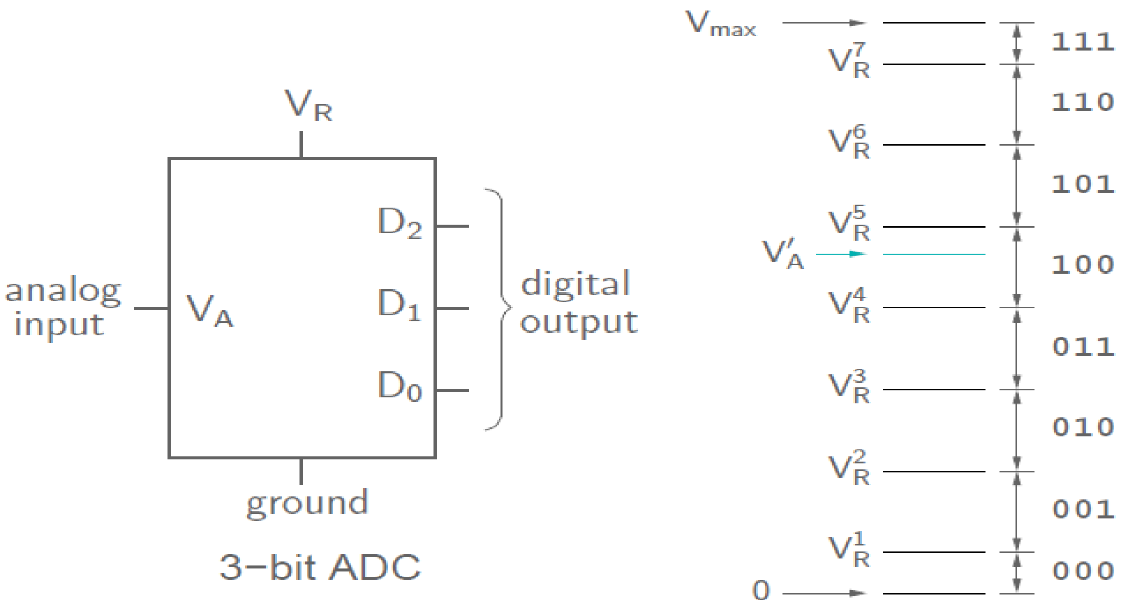
- We may think of each voltage interval (corresponding to 000, 001, etc.) as a "bin." In the above example, the input voltage V_A' falls in the 100 bin; therefore, the output of the ADC would be 100.

ADC: introduction



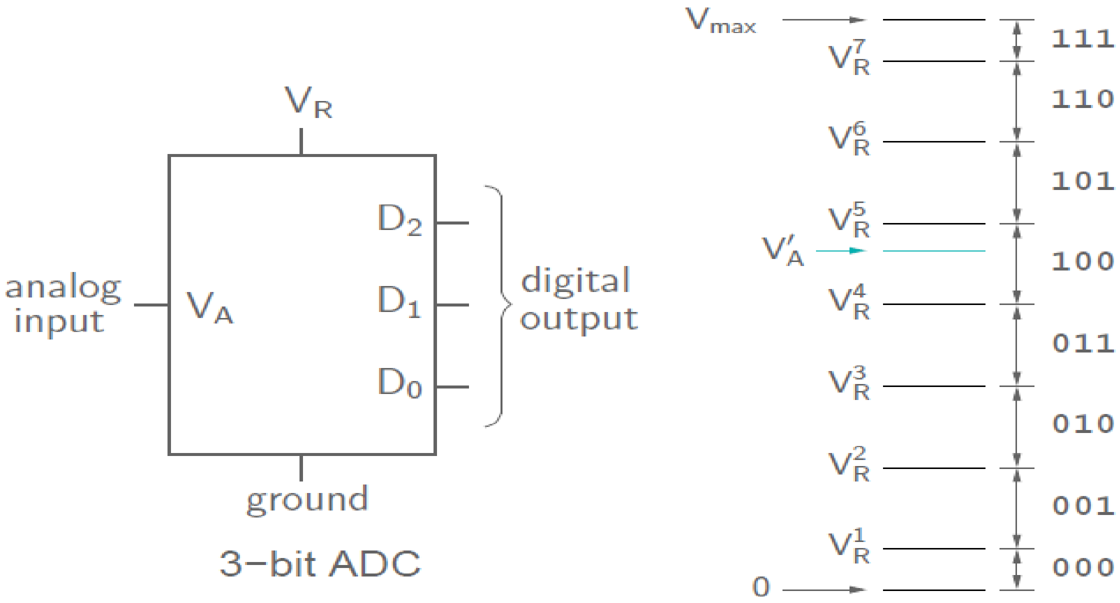
- ∗ If the input V_A is in the range $V_R^k < V_A < V_R^{k+1}$, the output is the binary number corresponding to the integer k. For example, for $V_A = V_A'$, the output is 100.
- ∗ We may think of each voltage interval (corresponding to 000, 001, etc.) as a “bin.” In the above example, the input voltage V_A' falls in the 100 bin; therefore, the output of the ADC would be 100.
- ∗ Note that, for an N-bit ADC, there would be 2^N bins.

ADC: introduction



- ∗ The basic idea behind an ADC is simple:

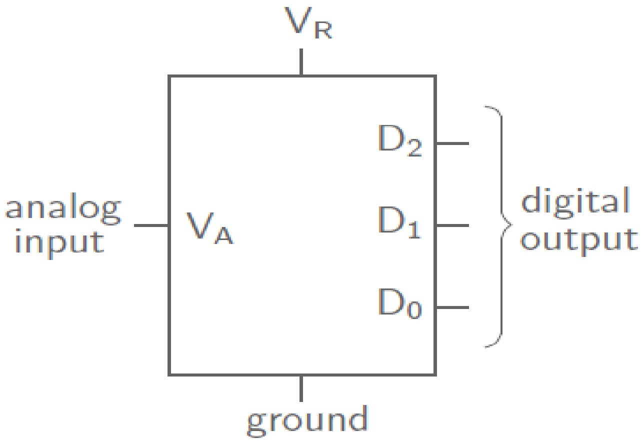
ADC: introduction



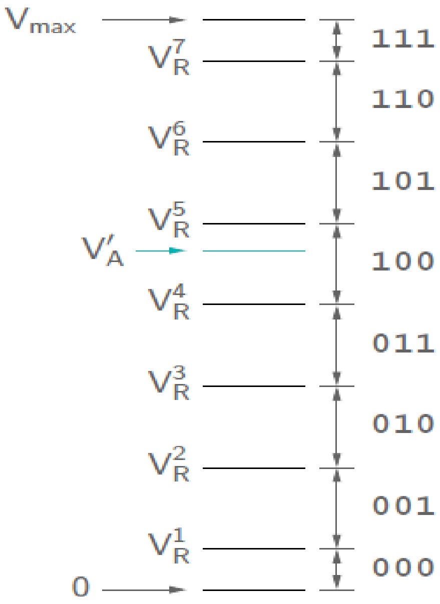
∗ The basic idea behind an ADC is simple:

- Generate reference voltages V_R^1, V_R^2, etc.

ADC: introduction



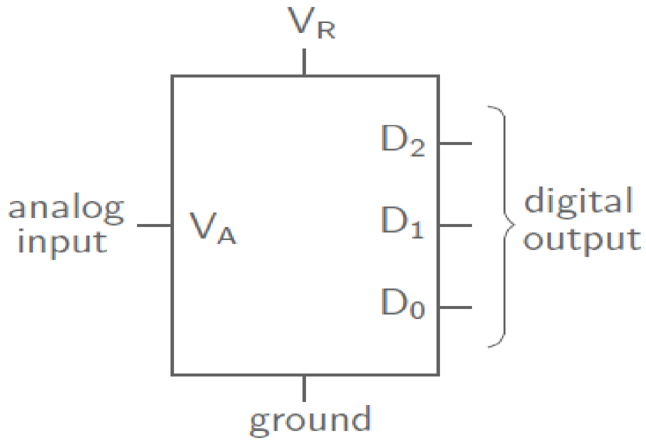
3−bit ADC



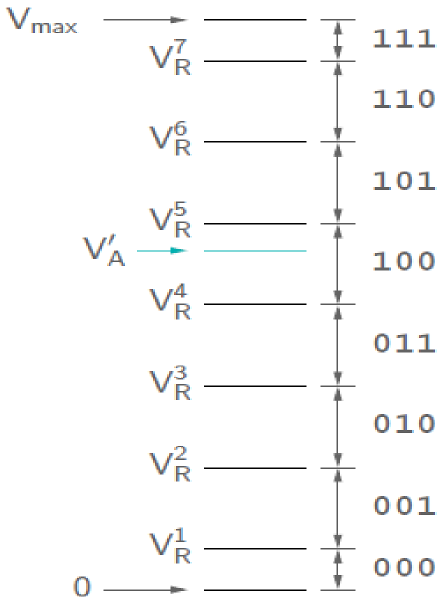
∗ The basic idea behind an ADC is simple:

- Generate reference voltages V_R^1, V_R^2, etc.
- Compare the input V_A with each of V_R^i to figure out which bin it belongs to.

ADC: introduction

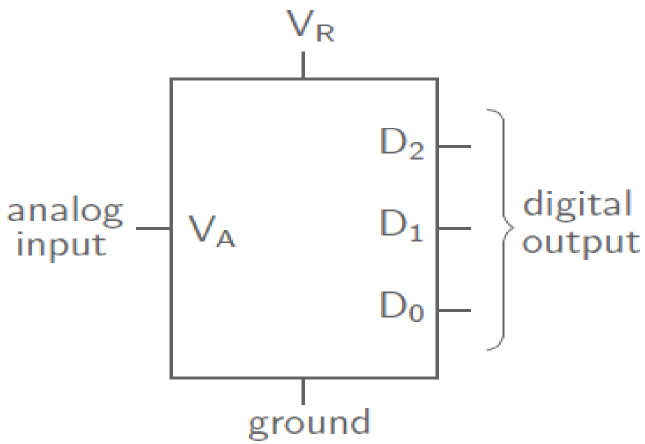


3−bit ADC

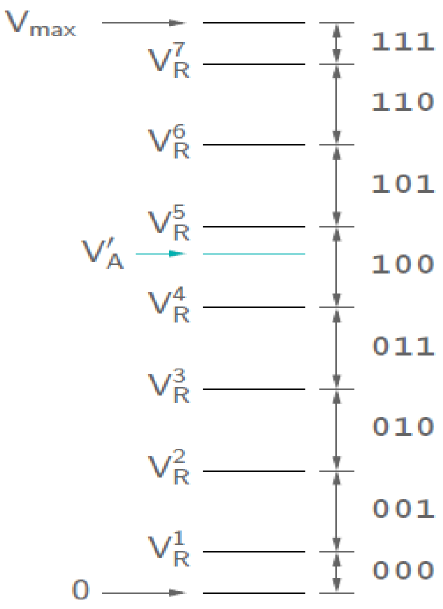


- The basic idea behind an ADC is simple:
  - Generate reference voltages V_R^1, V_R^2, etc.
  - Compare the input V_A with each of V_R^i to figure out which bin it belongs to.
  - If V_A belongs to bin k (i.e., $V_R^k < V_A < V_R^{k+1}$), convert k to the binary format.

ADC: introduction

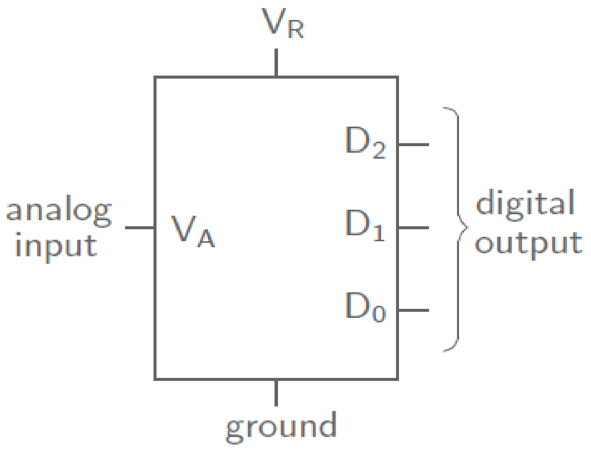


3−bit ADC

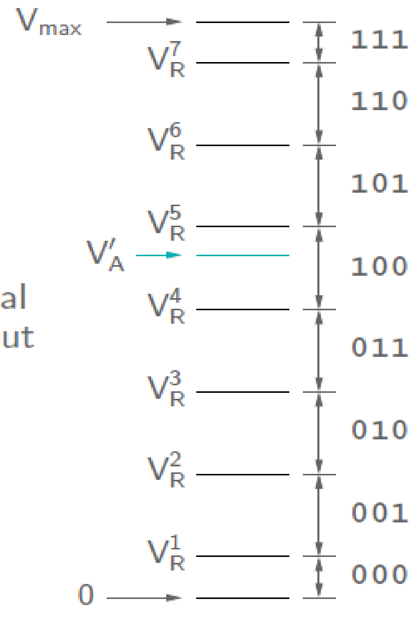


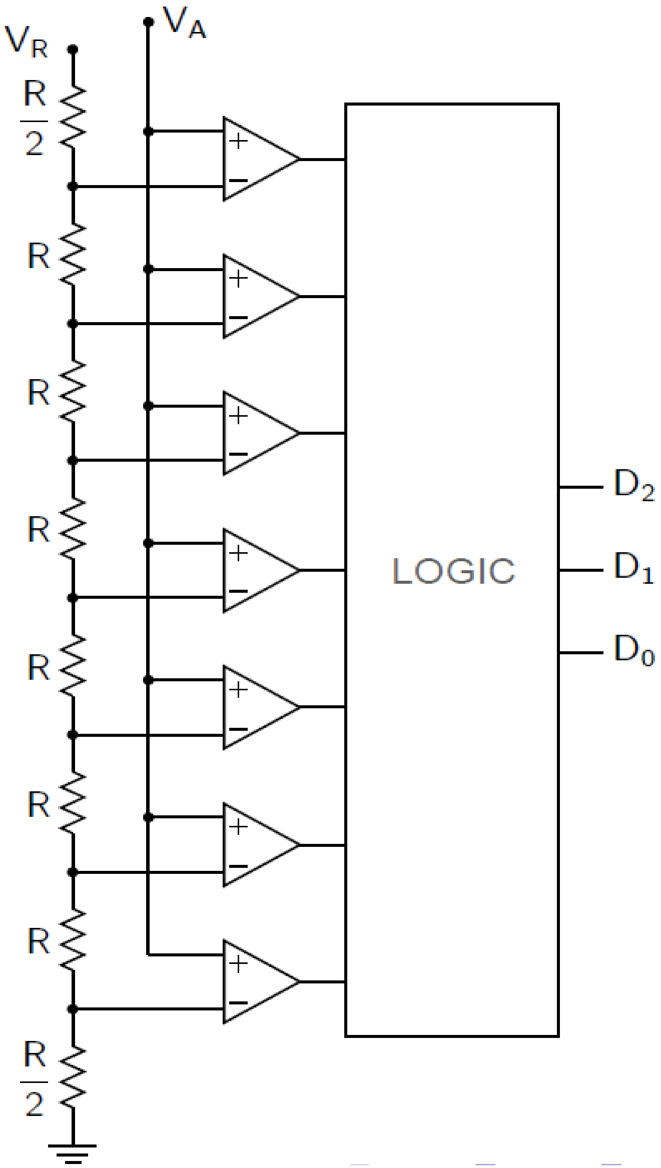
- ∗ The basic idea behind an ADC is simple:
  - Generate reference voltages V_R^1, V_R^2, etc.
  - Compare the input V_A with each of V_R^i to figure out which bin it belongs to.
  - If V_A belongs to bin k (i.e., $V_R^k < V_A < V_R^{k+1}$), convert k to the binary format.
    - ∗ A "parallel" ADC does exactly that → next slide.

3-bit parallel (flash) ADC

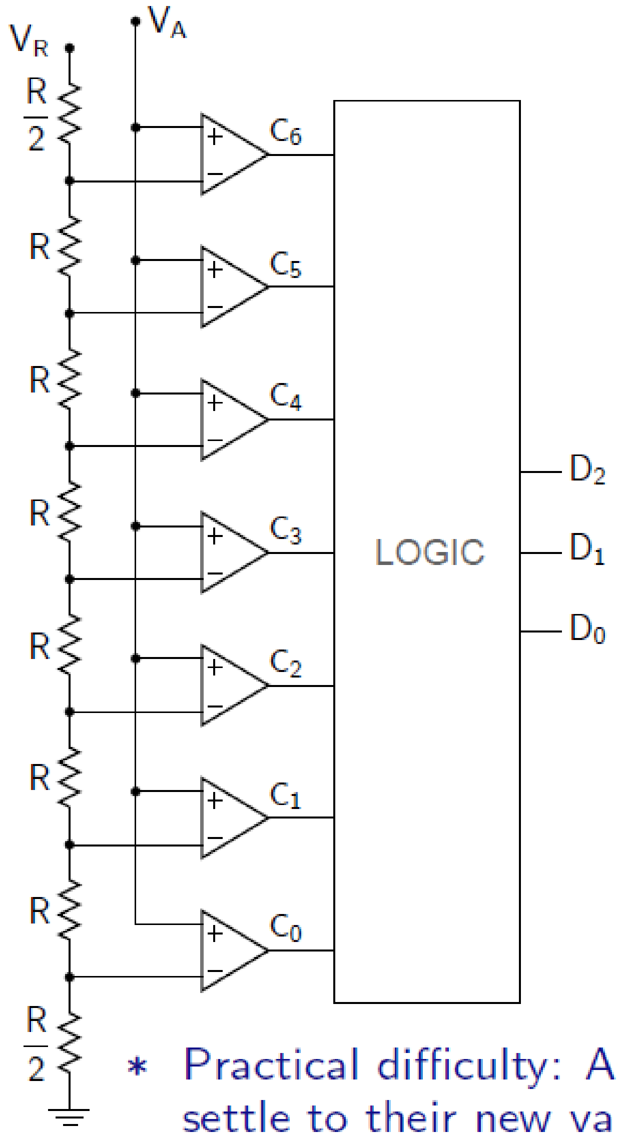


3−bit ADC



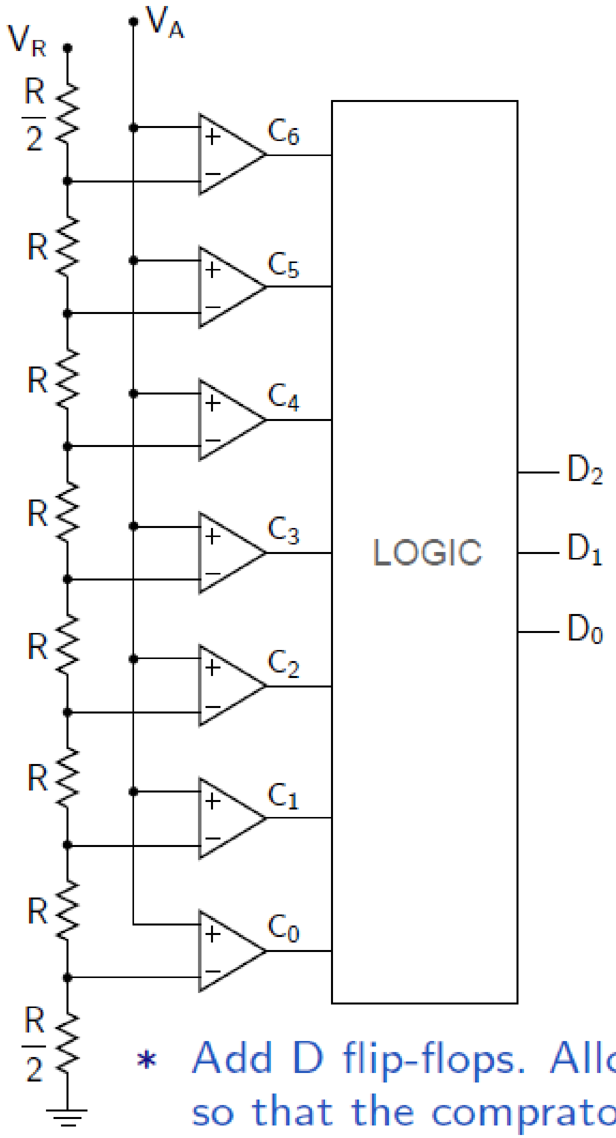


3-bit parallel (flash) ADC



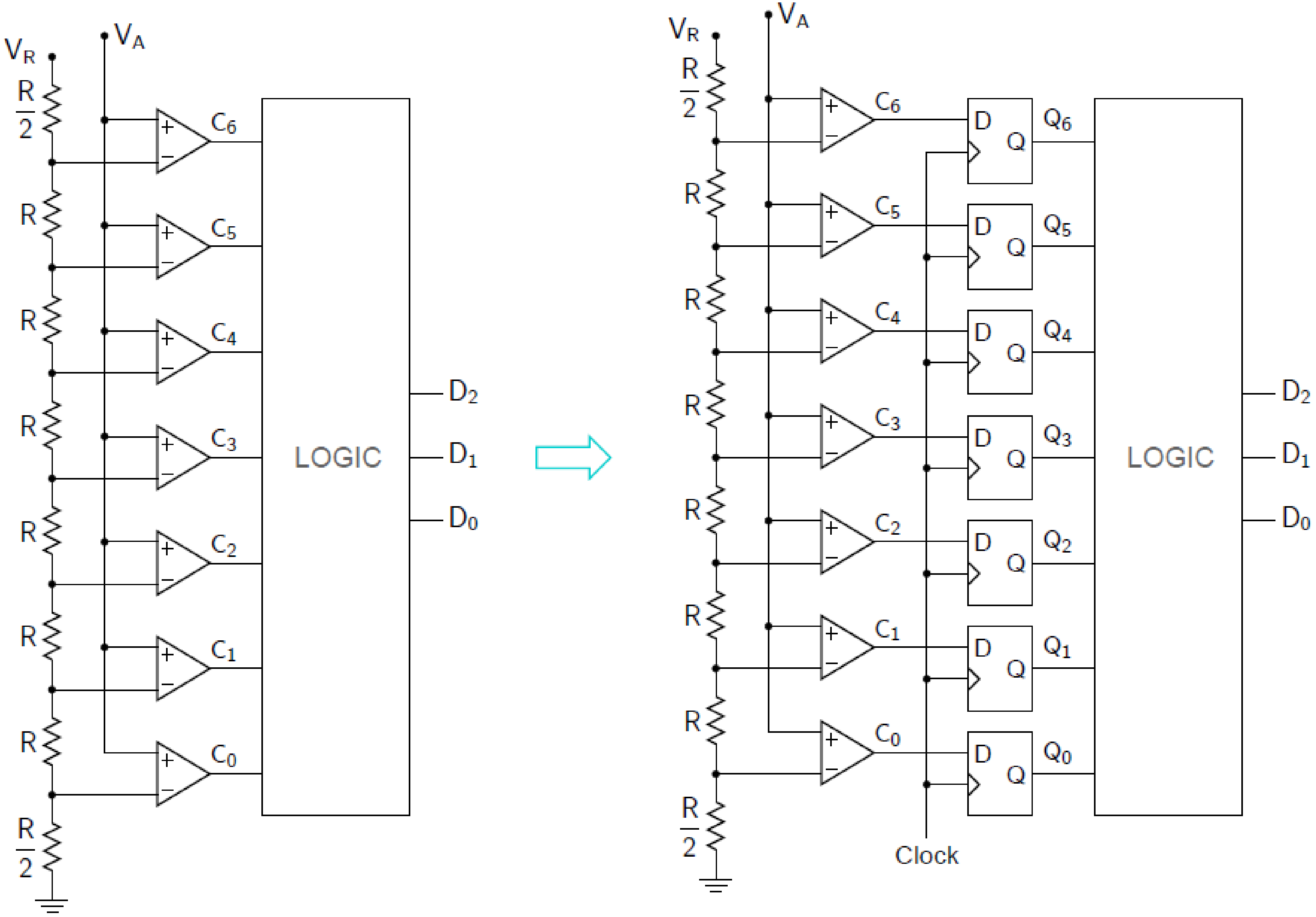
- Practical difficulty: As the input changes, the comparator outputs (C_0, C_1, etc.) may not settle to their new values at the same time.
  → ADC output will depend on when we sample it.

3-bit parallel (flash) ADC



- * Add D flip-flops. Allow sifficient time (between the change in V_A and the active clock edge) so that the comprator outputs have already settled to their new values before they get latched in.

3-bit parallel (flash) ADC



Parallel (flash) ADC

- In the parallel (flash) ADC, the conversion gets done “in parallel,” since all comparators operate on the same input voltage.

Parallel (flash) ADC

∗ In the parallel (flash) ADC, the conversion gets done “in parallel,” since all comparators operate on the same input voltage.

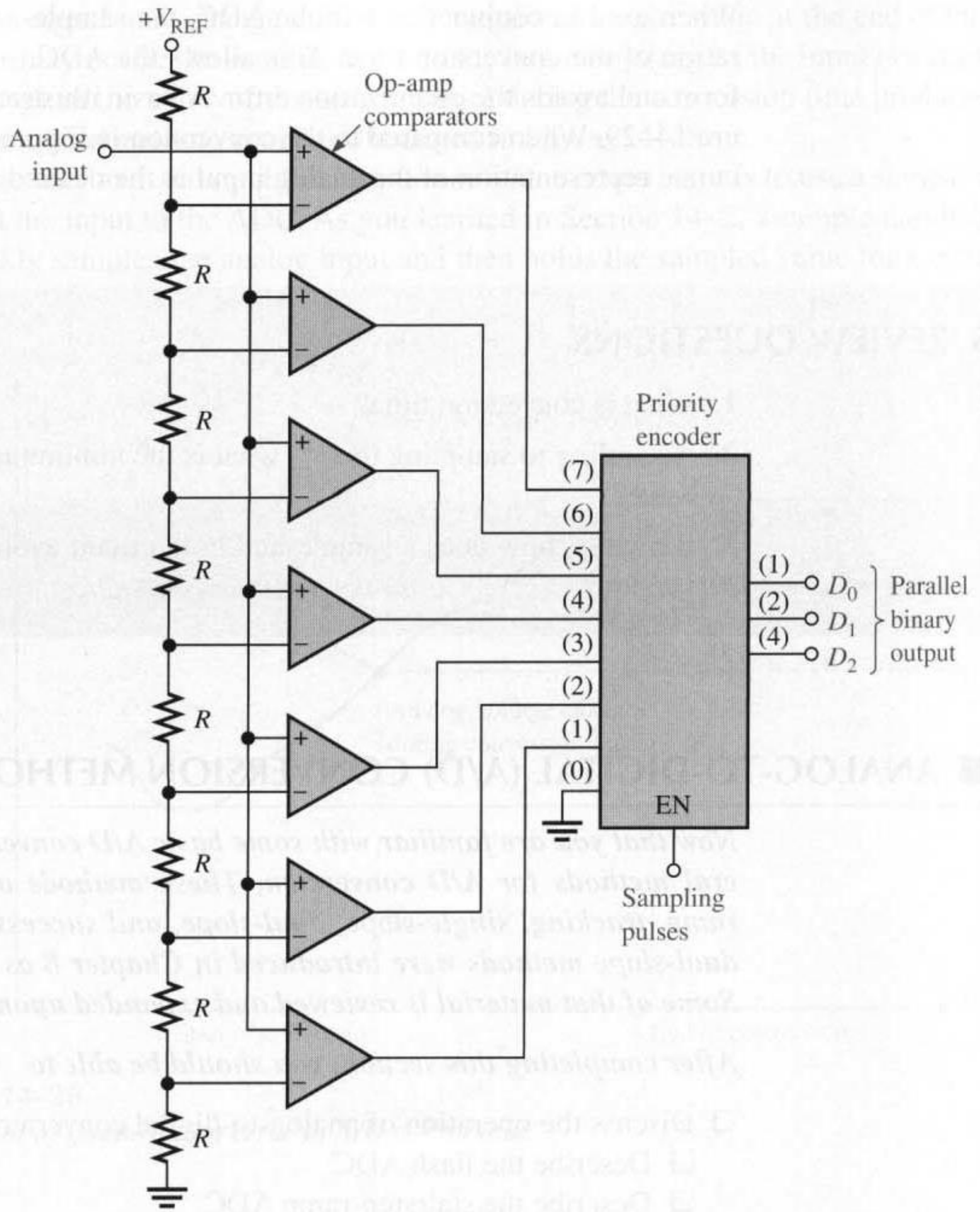
∗ Conversion time is governed only by the comparator response time → fast conversion (hence the name “flash” converter).

∗ Flash ADCs to handle 500 million analog samples per second are commercially available.

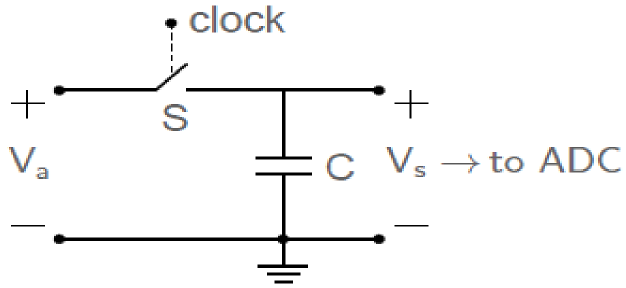
Parallel (flash) ADC

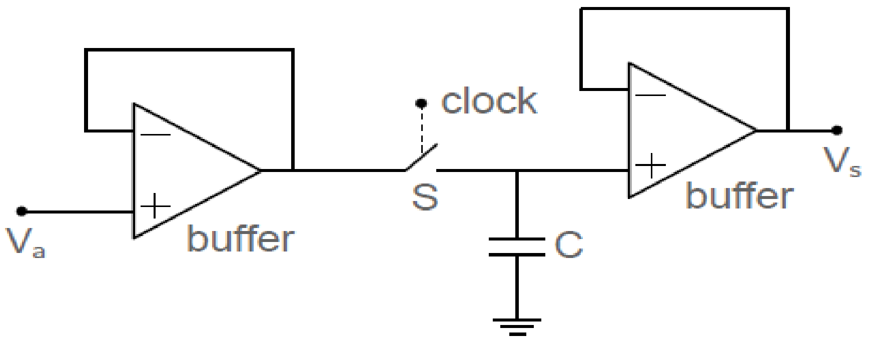
- In the parallel (flash) ADC, the conversion gets done “in parallel,” since all comparators operate on the same input voltage.
- Conversion time is governed only by the comparator response time → fast conversion (hence the name “flash” converter).
- Flash ADCs to handle 500 million analog samples per second are commercially available.
- 2^N comparators are required for N-bit ADC → generally limited to 8 bits.

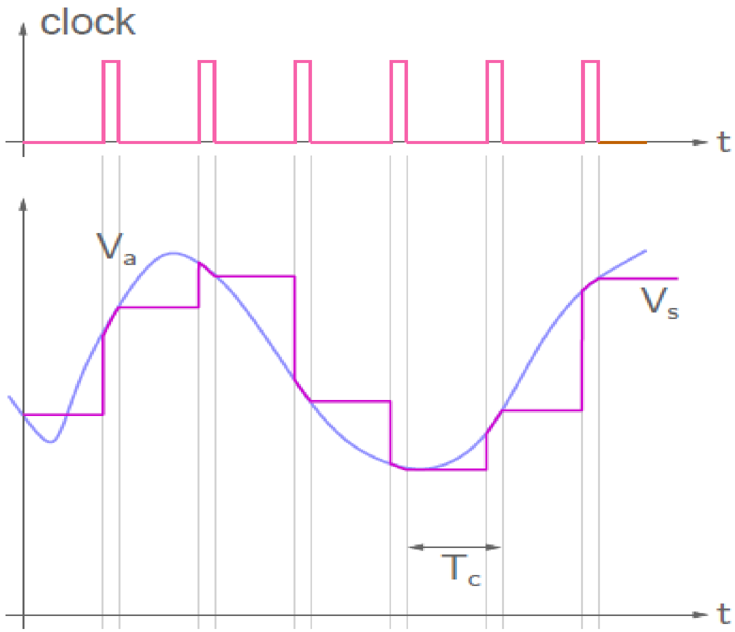
Alternative Structure for a 3-bit Parallel (Flash) ADC



ADC: sampling of input signal

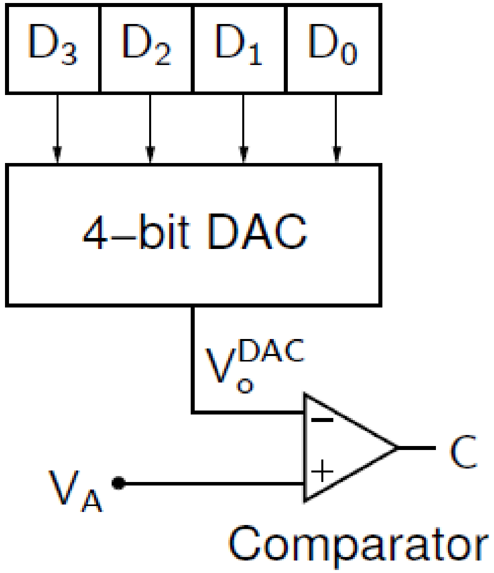






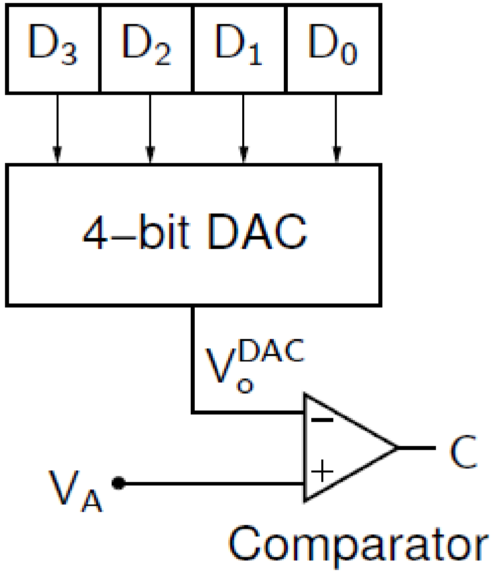
- An ADC typically operates on a “sampled” input signal ($V_s(t)$ in the figure) which is derived from the continuously varying input signal ($V_a(t)$ in the figure) with a “sample-and-hold” (S/H) circuit.
- The S/H circuit samples the input signal $V_a(t)$ at uniform intervals of duration T_c, the clock period.
- When the clock goes high, switch S (e.g., a FET or a CMOS pass gate) is closed, and the capacitor C gets charged to the signal voltage at that time. When the clock goes low, switch S is turned off, and C holds the voltage constant, as desired.

Successive Approximation ADC



- Suppose we have a 4-bit DAC. We can use it to perform A-to-D conversion by successively setting the four bits as follows.
  - Start with $D_3 D_2 D_1 D_0 = 0000$, $\texttt{I} = 3$.

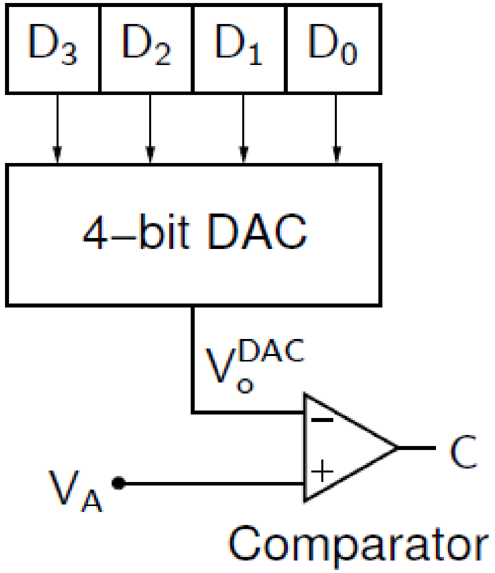
Successive Approximation ADC



∗ Suppose we have a 4-bit DAC. We can use it to perform A-to-D conversion by successively setting the four bits as follows.

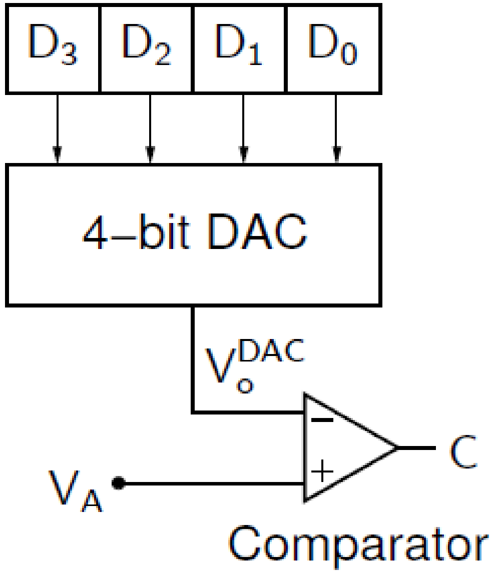
- Start with $D_3 D_2 D_1 D_0 = 0000$, $\mathtt{I} = 3$.
- Set $D[\mathtt{I}] = 1$ (keep other bits unchanged).

Successive Approximation ADC



- ∗ Suppose we have a 4-bit DAC. We can use it to perform A-to-D conversion by successively setting the four bits as follows.
  - Start with $D_3 D_2 D_1 D_0 = 0000$, $\mathtt{I} = 3$.
  - Set $D[\mathtt{I}] = 1$ (keep other bits unchanged).
  - If $V_o^{DAC} > V_A$ (i.e., $C = 0$), set $D[\mathtt{I}] = 0$; else, keep $D[\mathtt{I}] = 1$.

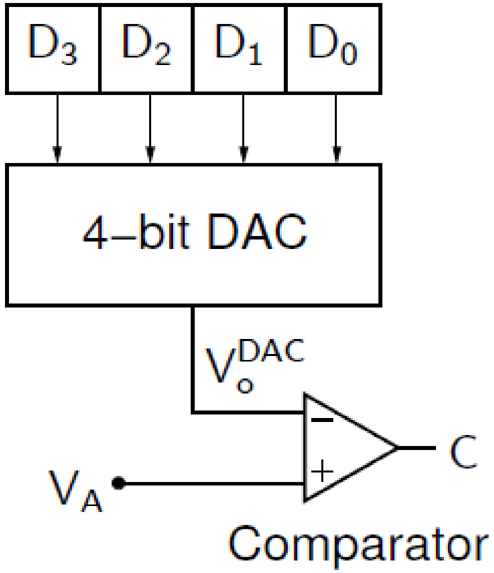
Successive Approximation ADC



∗ Suppose we have a 4-bit DAC. We can use it to perform A-to-D conversion by successively setting the four bits as follows.

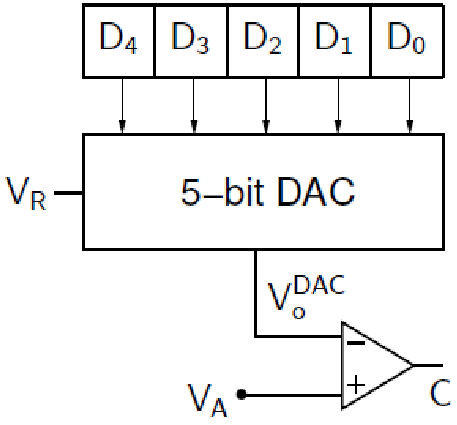
- Start with $D_3 D_2 D_1 D_0 = 0000$, $\mathtt{I} = 3$.
- Set $D[\mathtt{I}] = 1$ (keep other bits unchanged).
- If $V_o^{DAC} > V_A$ (i.e., $C = 0$), set $D[\mathtt{I}] = 0$; else, keep $D[\mathtt{I}] = 1$.
- $\mathtt{I} \leftarrow \mathtt{I} - 1$; go to step 1.

Successive Approximation ADC

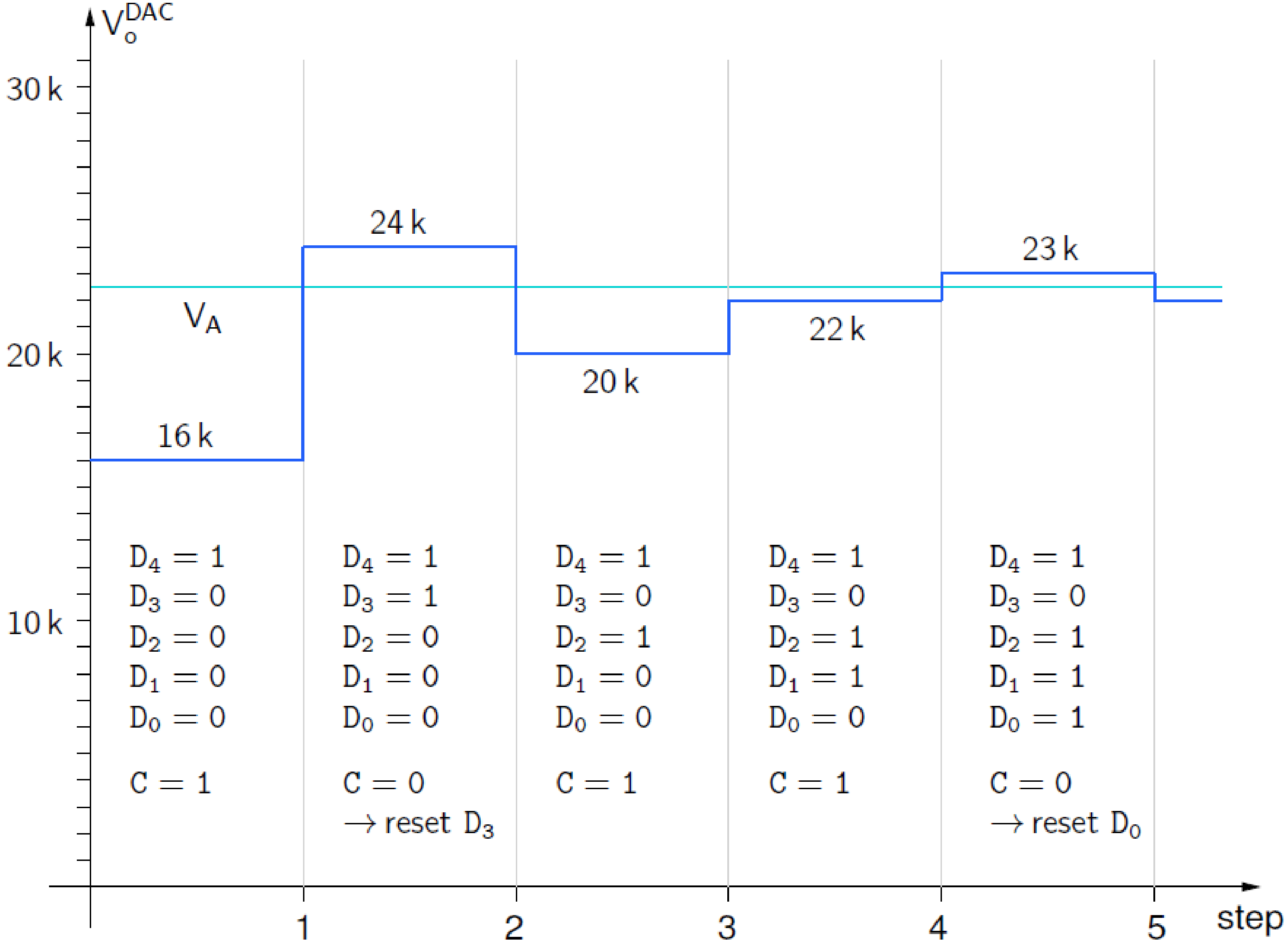


- * At the end of four steps, the digital output is given by $D_3 D_2 D_1 D_0$. Example → next slide.

Successive Approximation ADC



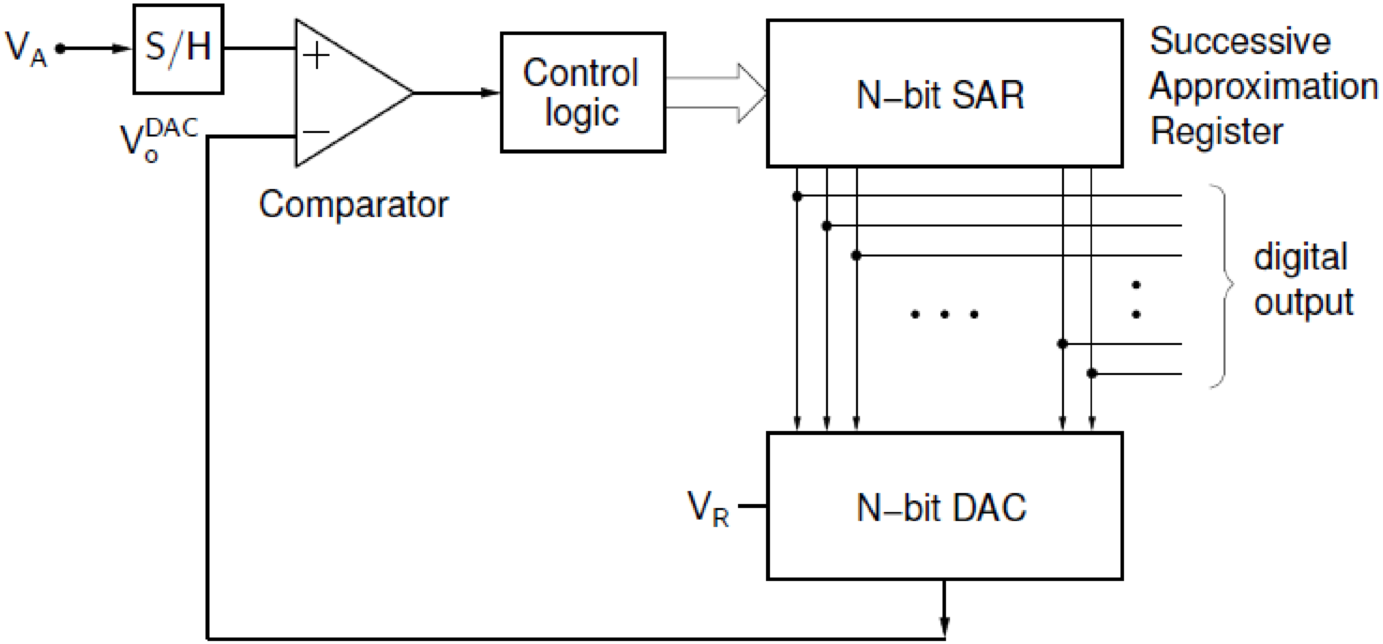
(Note: $k \propto V_R$)



	1	2	3	4	5
	$D_4 = 1$	$D_4 = 1$	$D_4 = 1$	$D_4 = 1$	$D_4 = 1$
	$D_3 = 0$	$D_3 = 1$	$D_3 = 0$	$D_3 = 0$	$D_3 = 0$
	$D_2 = 0$	$D_2 = 0$	$D_2 = 1$	$D_2 = 1$	$D_2 = 1$
	$D_1 = 0$	$D_1 = 0$	$D_1 = 0$	$D_1 = 1$	$D_1 = 1$
	$D_0 = 0$	$D_0 = 0$	$D_0 = 0$	$D_0 = 0$	$D_0 = 1$
	$C = 1$	$C = 0$ → reset D_3	$C = 1$	$C = 1$	$C = 0$ → reset D_0

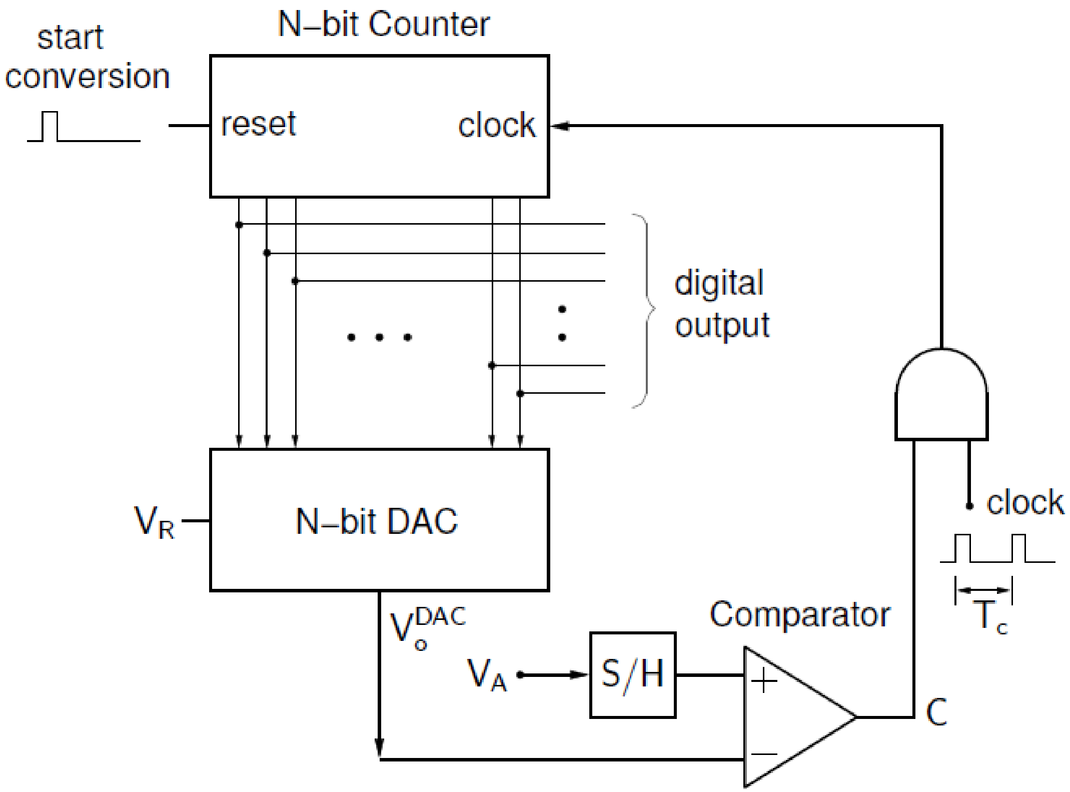
- At the end of the 5th step, we know that the input voltage corresponds to 10110.

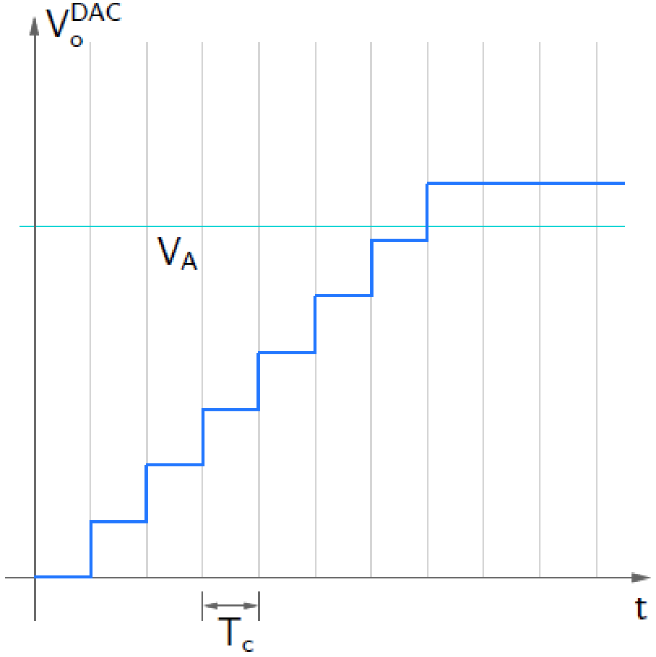
Successive Approximation ADC



- ∗ Each step (setting SAR bits, comparison of V_A and V_o^{DAC}) is performed in one clock cycle → conversion time is N cycles, irrespective of the input voltage value V_A.
- ∗ S. A. ADCs with built-in or external S/H (sample-and-hold) are available for 8- to 16-bit resolution and conversion times of a few μsec to tens of μsec.
- ∗ Useful for medium-speed applications such as speech transmission with PCM.

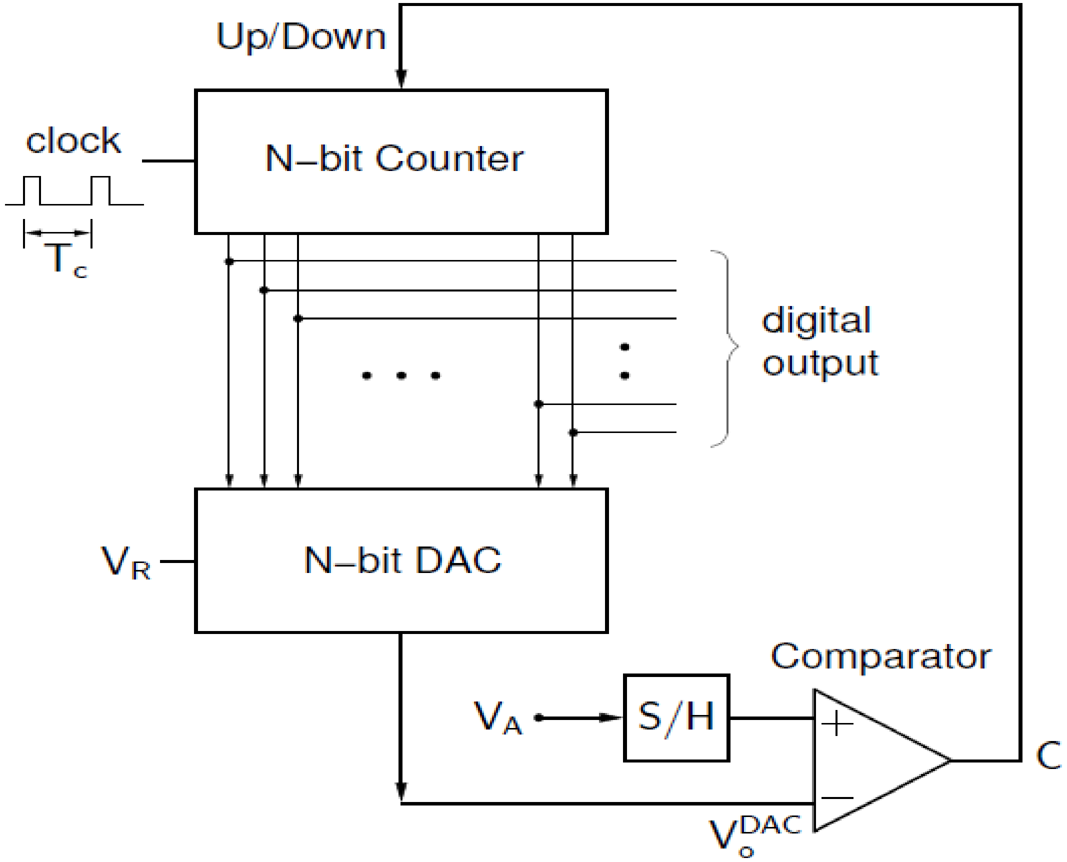
Counting ADC

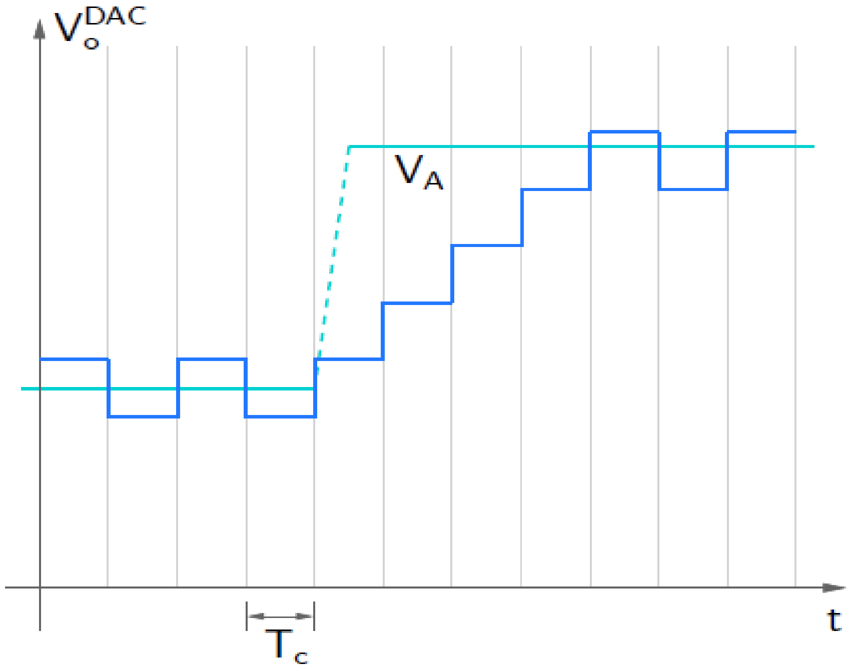




- ∗ The "start conversion" signal clears the counter; counting begins, and V_o^{DAC} increases with each clock cycle.
- ∗ When V_o^{DAC} exceeds V_A, C becomes 0, and counting stops.
- ∗ Simple scheme, but (a) conversion time depends on V_A, (b) slow (takes 2^N clock cycles in the worst case) → tracking ADC (next slide)

Tracking ADC





- ∗ The counter counts up if $V_o^{DAC} < V_A$; else, it counts down.
- ∗ If V_A changes, the counter does not need to start from $000\cdots0$, so the conversion time is less than that required by a counting ADC.
- ∗ used in low-cost, low-speed applications, e.g., measuring output from a temperature sensor or a strain gauge