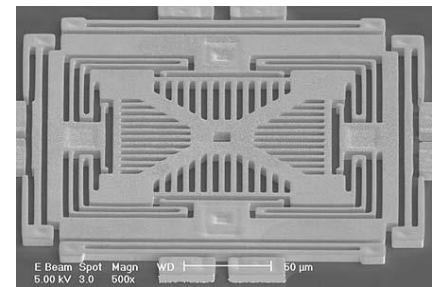
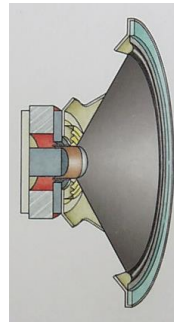


# ADC

## (Analog Digital to Converter)

# Dunia adalah Analog

- Semua yang ada di dunia ini adalah sinyal analog
  - Suara, cahaya, suhu, gaya gravitasi
- Harus diubah menjadi sinyal listrik
  - Transducer: mengubah tipe energi dari satu ke lainnya
    - *Electro-mechanical, Photonic, Electrical, ...*
  - Contoh
    - *Microphone/speaker*
    - *Thermocouple*
    - *Accelerometer*



# Sinyal Analog

Analog signals – Mengukur langsung kuantitas  
berdasar kuantitas lain

Misal:

- Speedometer Mobil – Jarum bergerak lebih cepat jika ada percepatan
- Stereo – Volume naik jika knob diputar.

# Sinyal Digital

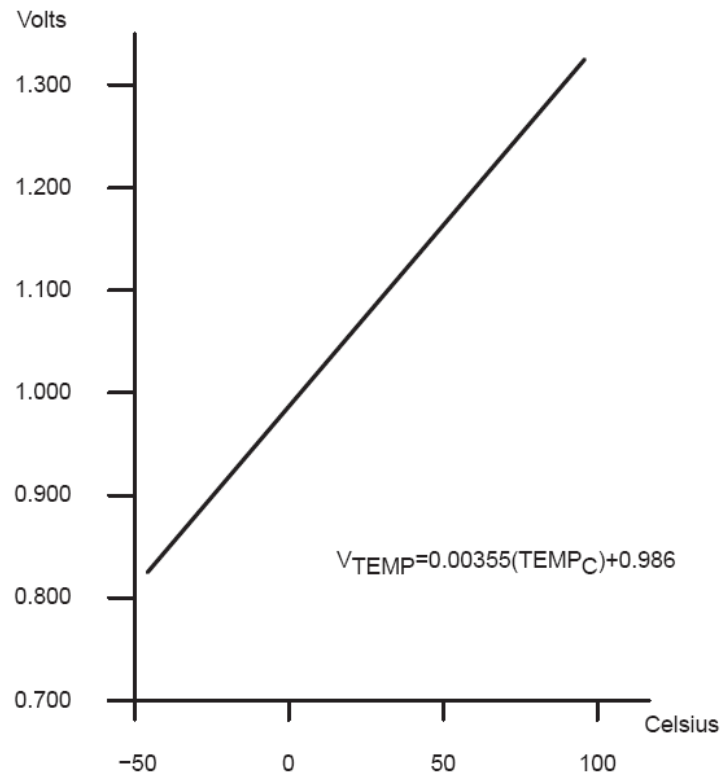
Sinyal Digital → hanya punya dua keadaan (Biner untuk komputer digital), yaitu 0 dan 1. “1” bisa ON, “0” bisa OFF.

Misal:

- Saklar Lampu bisa ON atau OFF
- Pintu bisa tertutup atau terbuka

# Tranduser

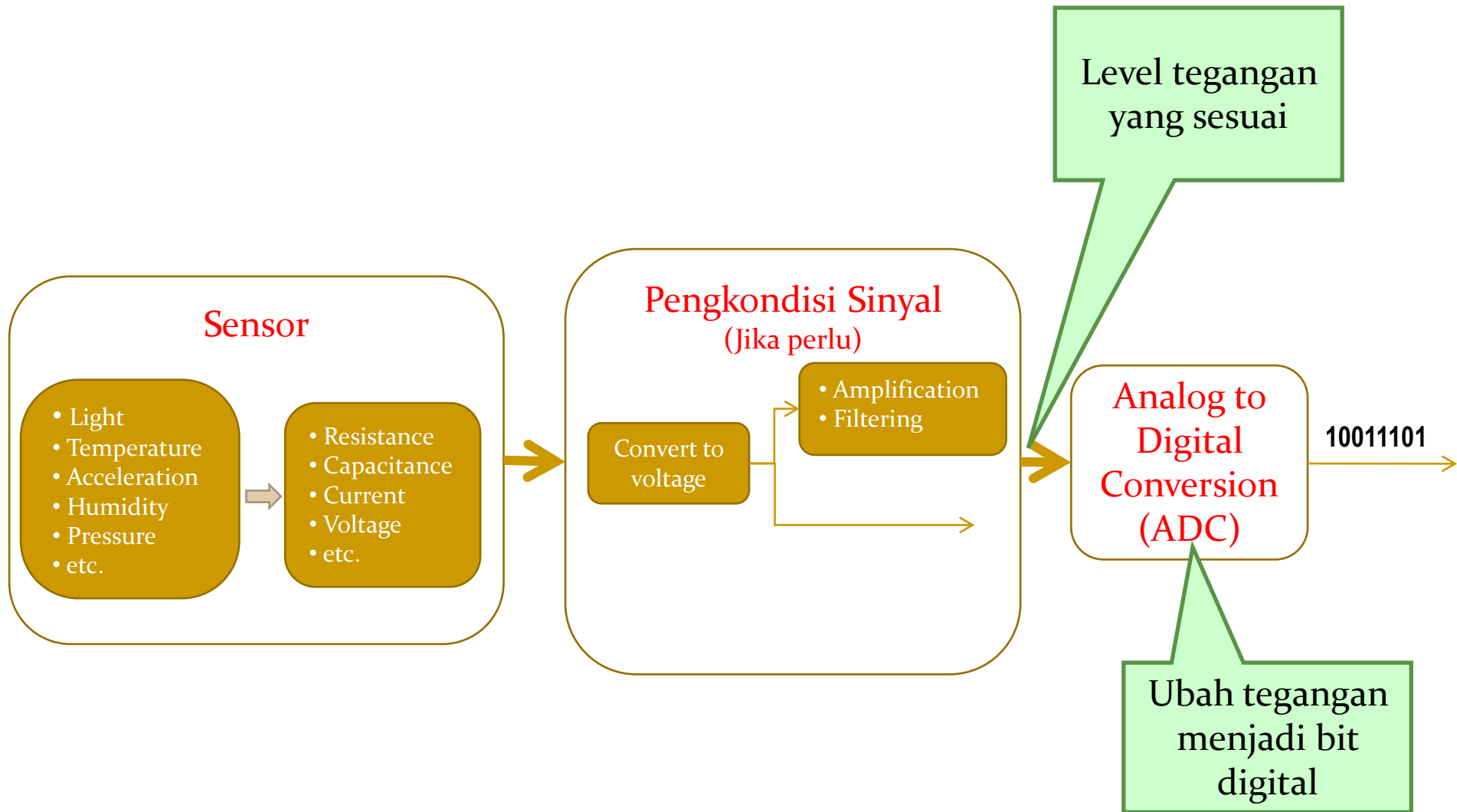
- Mengubah fenomena fisika menjadi tegangan potensial.



# Contoh Transduser

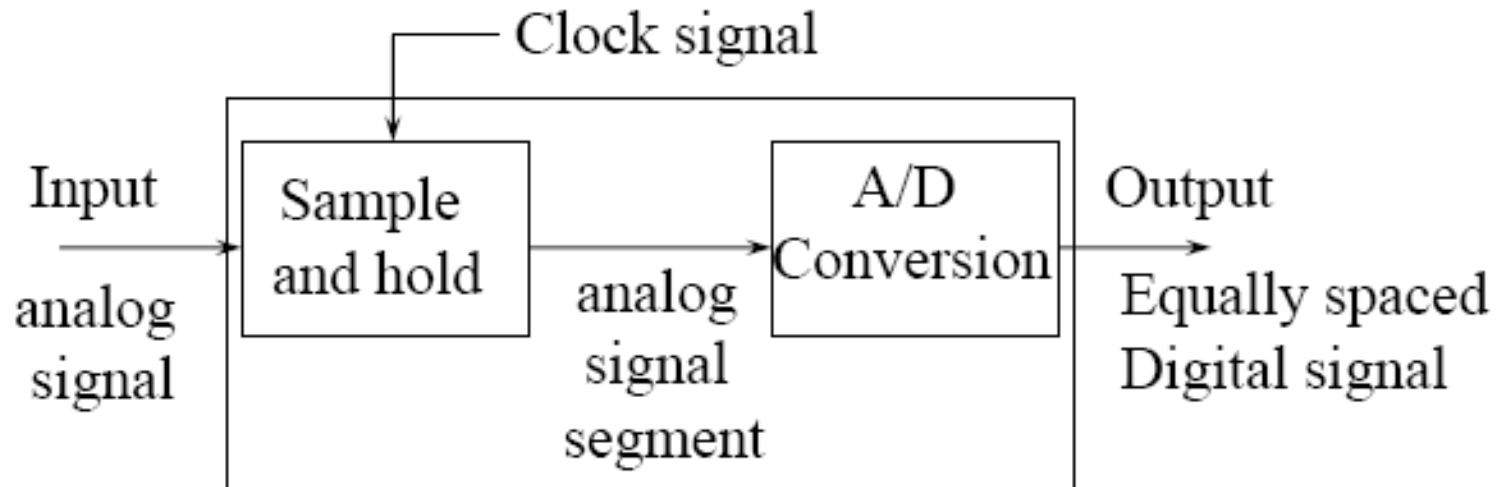
- **Microphones** – Mengambil variasi gelombang tekanan suara di udara dan diubah menjadi variasi sinyal listrik
- **Strain Gages** – Menentukan besarnya regangan /*strain* (berbentuk dimensi) ketika ada tekanan
- **Thermocouple** – Alat pengukur suhu dengan mengubah energi panas menjadi energi listrik
- **Voltmeters**
- **Digital Multimeters**

# ADC



# Apa yang dilakukan A/D converter ?

- Mengubah sinyal analog menjadi kode biner





# ADC



**Sample and Hold** : Sinyal analog harus di Hold sebelum bisa di sample, jika tidak maka sinyal akan berubah terlebih dahulu sebelum di sample

---

## 2-langkah proses pada ADC :

- **Quantizing** – Memecah sinyal analog menjadi beberapa bagian (a set of finite states)
  - **Encoding** – mengubahnya menjadi sebuah kode atau angka digital dan mencocokkannya dengan input
-

# Step 1: Quantizing

Misal:

Sinyal 0-10V. Bagi menjadi beberapa bagian dengan kenaikan **1.25V**. (caranya? **next slide** ...)

| Output States | Kisaran Output (V) |
|---------------|--------------------|
| 0             | 0.00-1.25          |
| 1             | 1.25-2.50          |
| 2             | 2.50-3.75          |
| 3             | 3.75-5.00          |
| 4             | 5.00-6.25          |
| 5             | 6.25-7.50          |
| 6             | 7.50-8.75          |
| 7             | 8.75-10.0          |

# Quantizing

**Jumlah Bagian dihitung dengan:**

$$N = 2^n$$

n = jumlah bit pada ADC

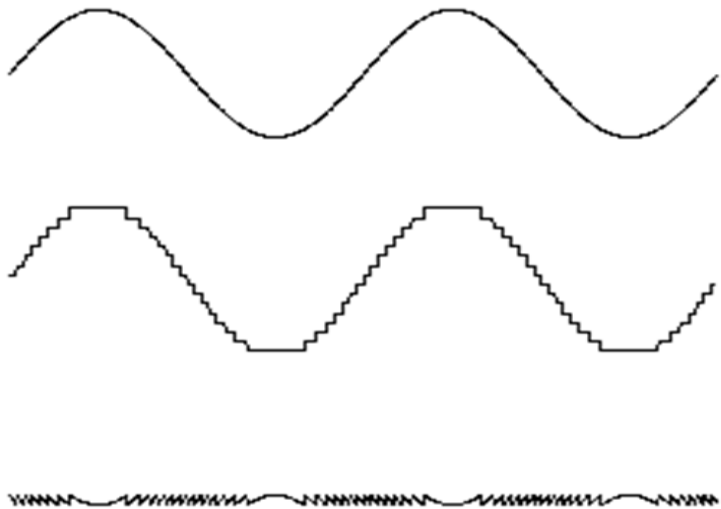
Misal: 3 bit A/D converter,  $N = 2^3 = 8$  bagian.

**Analog quantization size:**

$$Q = (V_{\max} - V_{\min}) / N = (10V - 0V) / 8 = 1.25V$$

# Quantizing Error

- Ketika sinyal diubah menjadi digital, Presisi terbatas oleh bit.
- Gambar di bawah menunjukkan sinyal analog yang diubah dengan ke digital – dalam kasus ini , 8 bit.
- Sinyal analog yang smooth, hanya bisa direpresentasikan dalam bentuk sinyal undak karena terbatas presisinya dengan bit yang digunakan.

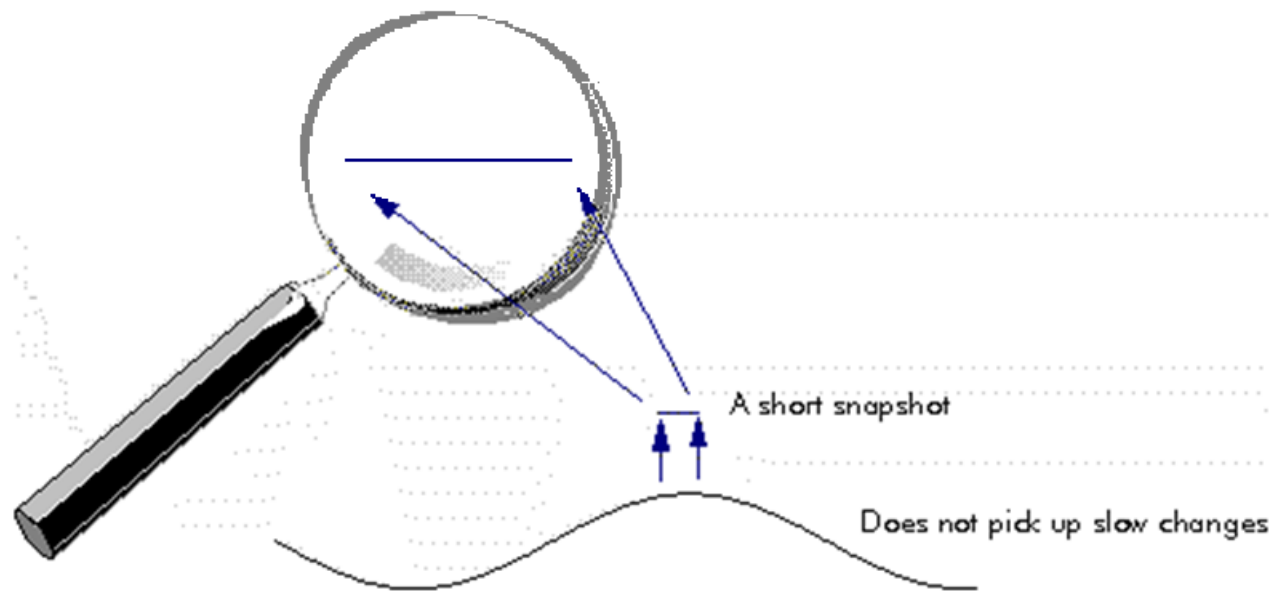


limited precision leads to errors...

which are signal dependent

# Sampling frequency

- Kita tidak dapat melihat perubahan (slow change) jika waktu kurang lama.
- Kita harus mensample cukup lama untuk frekuensi rendah dan perubahan kecil
- Sample harus cepat dan lama untuk mencapai resolusi frekuensi yang bagus → banyak sample → banyak komputasi -> waktu cukup lama → ini harus di optimasi



# Encoding

- Disini kita kodekan tiap bagian menjadi kode/angka digital (biner) agar bisa dibaca oleh komputer /microcontroller

| Output States | Ekuivalensi Output biner |
|---------------|--------------------------|
| 0             | 000                      |
| 1             | 001                      |
| 2             | 010                      |
| 3             | 011                      |
| 4             | 100                      |
| 5             | 101                      |
| 6             | 110                      |
| 7             | 111                      |

# Akurasi ADC

Dua cara memperbaiki akurasi ADC:

- ❖ Menaikkan resolusi → memperbaiki akurasi pengukuran amplitudo sinyal analog.
- ❖ Menaikkan waktu pengambilan sampel (sampling rate) → menaikkan frekuensi maksimum yang dapat diukur



# Resolusi

- ❖ Resolusi (Jumlah nilai diskrit yang bisa dihasilkan oleh konverter) = Analog Quantization size (Q)  
 $(Q) = V_{range} / 2^n$ , dimana  $V_{range}$  = kisaran tegangan analog yang dapat di representasikan
- ❖ Pada contoh:  $Q = 1.25V \rightarrow$  resolusi tinggi.  
Jika 2-bit  $\rightarrow$  resolusi lebih rendah  $\rightarrow Q = 10/2^2 = 2.50V$ .

# Type-type ADC

- ❑ Flash ADC
- ❑ Counter-Ramp Converter
- ❑ Dual Slope (integrating) ADC
- ❑ Successive Approximation ADC
- ❑ Delta-Sigma ADC
- ❑ Sub-Ranging ADCs
- ❑ Pipelined ADCs
- ❑ etc

---

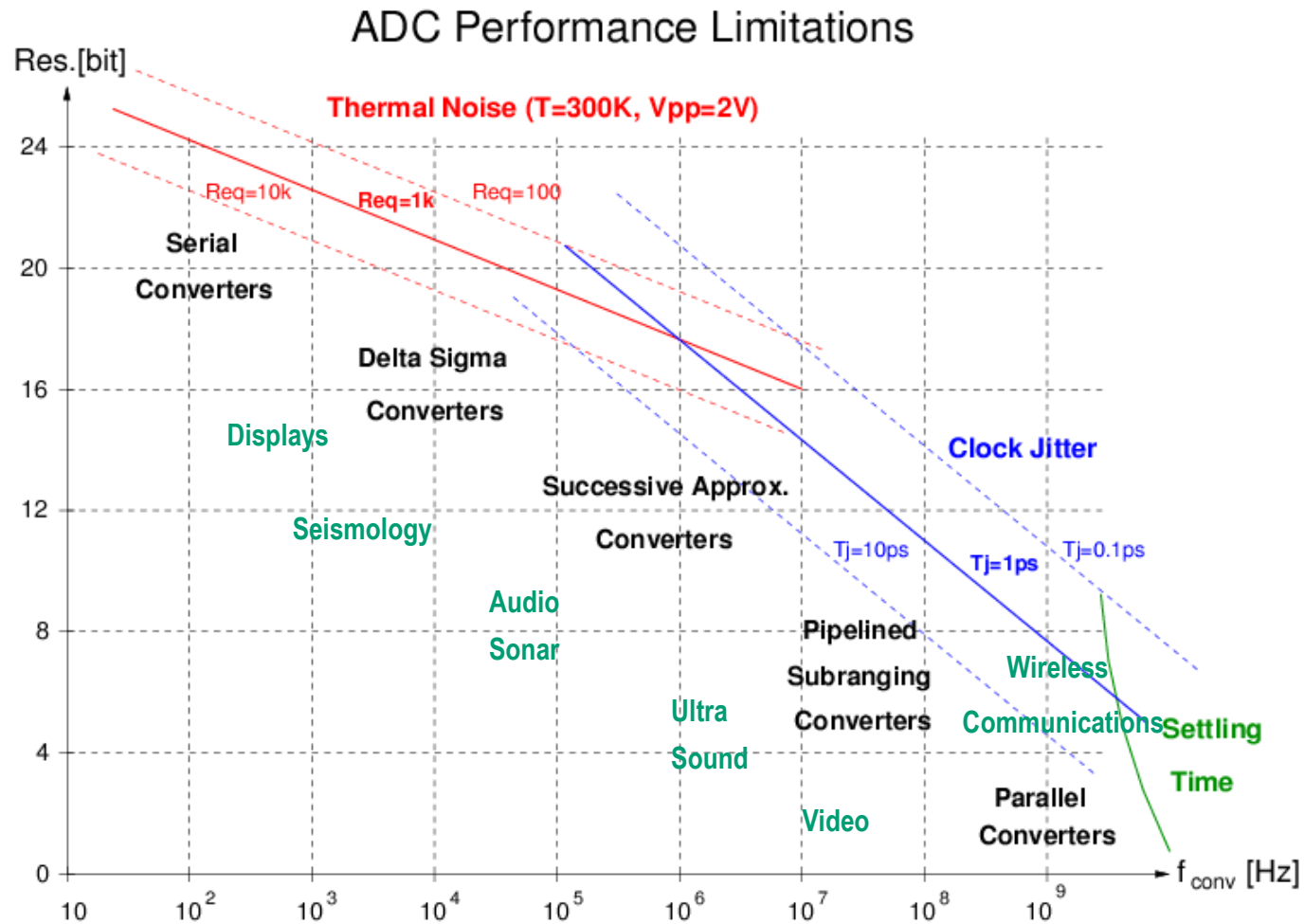
# Type-type ADC

- ❑ **Flash ADCs:** High speed, but large area and high power dissipation. Suitable for low-medium resolution (6-10 bit).
  - ❑ **Integrating ADCs or Ramp ADCs:** Low speed but high resolution. Simple circuitry.
  - ❑ **Successive Approximation ADCs:** Moderate speed with medium-high resolution (8-14 bit). Compact implementation.
  - ❑ **Delta-Sigma based ADCs:** Moderate bandwidth due to oversampling, but very high resolution thanks to oversampling and noise shaping.
  - ❑ **Sub-Ranging ADCs:** Require exponentially fewer comparators than Flash ADCs. Hence, they consume less silicon area and less power.
  - ❑ **Pipelined ADCs:** Medium-high resolution with good speed. The trade-offs are latency and power.
-

# Berdasar prinsip konversi

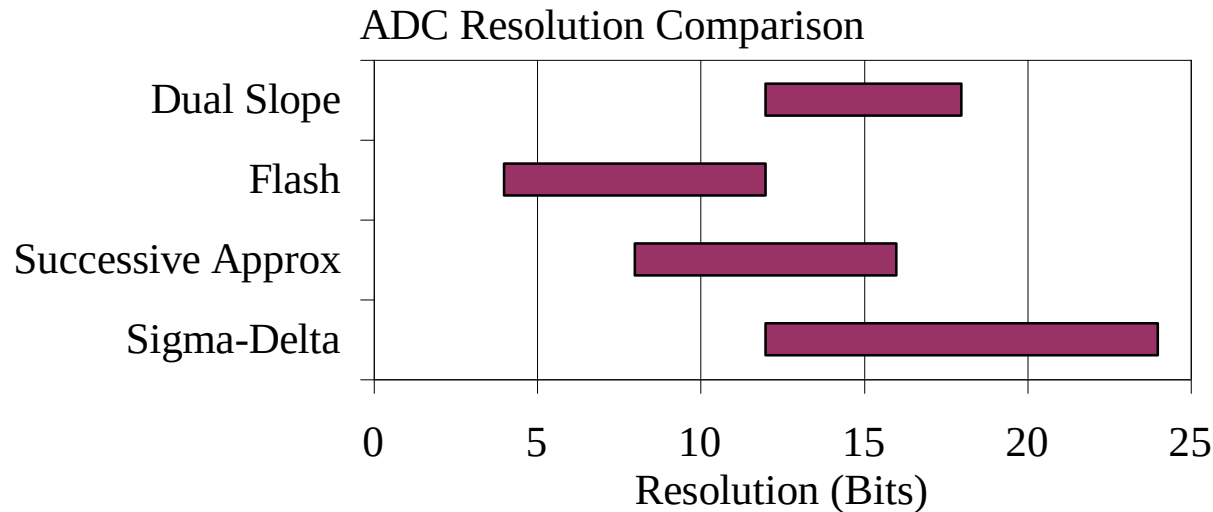
| Principle                | Resolution         | Speed               | Cost                 |
|--------------------------|--------------------|---------------------|----------------------|
| Serial Conversion        | High<br>>12 Bit    | Low<br><1kHz        | Low                  |
| Successive Approximation | Medium<br>8-14 Bit | Medium<br><10MHz    | Medium               |
| Parallel Conversion      | Low<br>6-10 Bit    | High<br><100MHz     | High                 |
| Delta Sigma              | High<br>>12 Bit    | Low-Medium<br><1MHz | Low<br>(analog only) |

# Aplikasi ADC



→ Selection of ADC Architecture is driven by Application

# Perbandingan empat tipe ADC



| Type             | Speed (relative) | Cost (relative) |
|------------------|------------------|-----------------|
| Dual Slope       | Slow             | Med             |
| Flash            | Very Fast        | High            |
| Successive Appox | Medium – Fast    | Low             |
| Sigma-Delta      | Slow             | Low             |

# Flash ADC

---

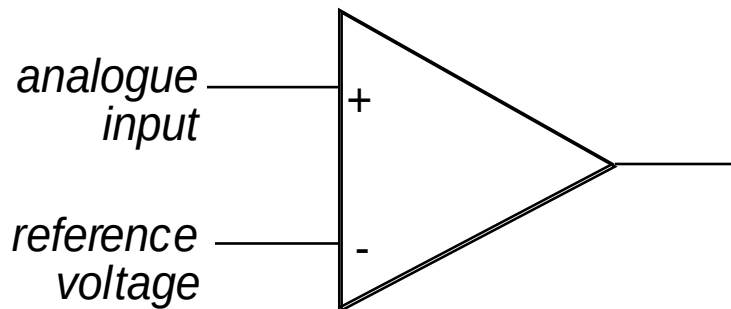
# Flash ADC

- ❑ Terdiri dari deretan komparator dimana masing-masing komparator membandingkan input dengan tegangan referensi tertentu.
  - ❑ Jumlah komparator = Jumlah level kuantisasi.
  - ❑ Output komparator dihubungkan dengan enkoder yang akan mengubahnya menjadi bilangan / kode biner.
-

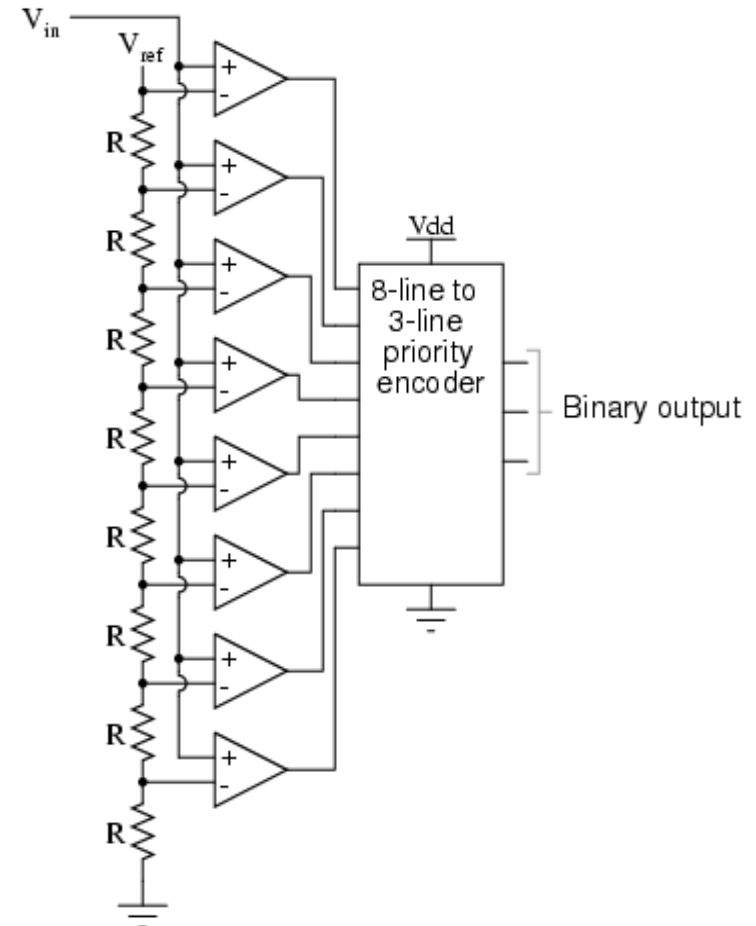
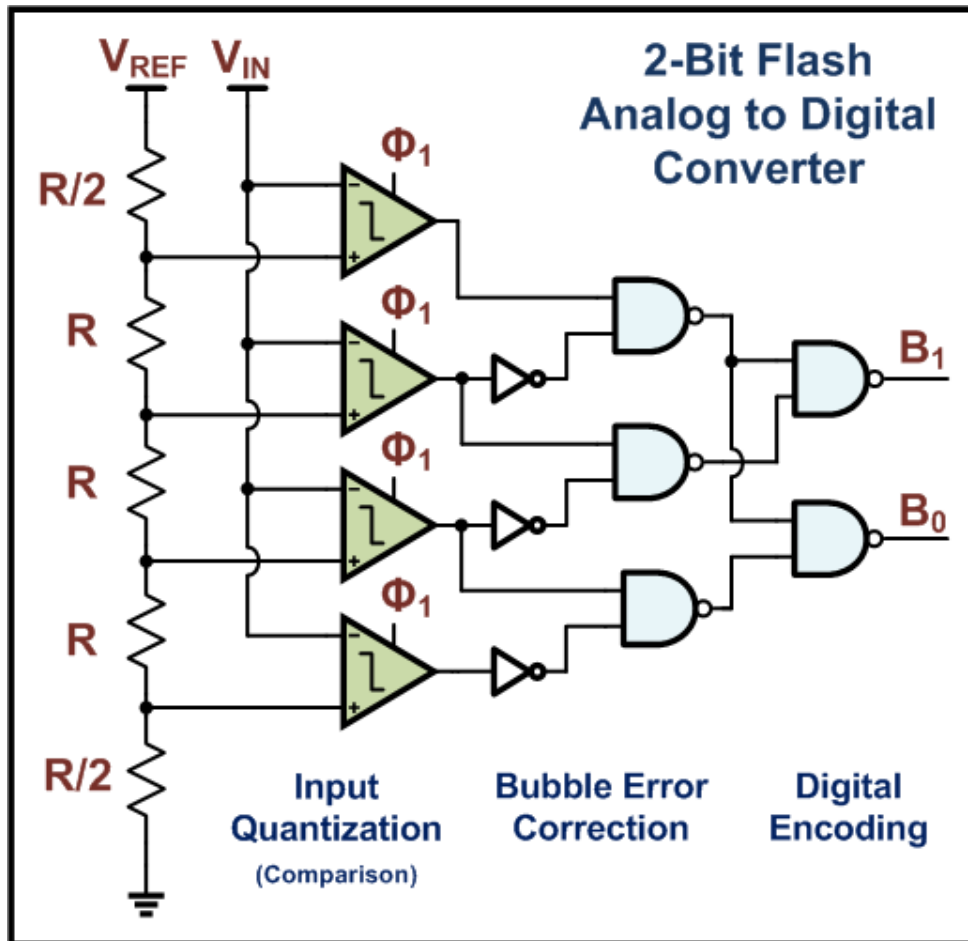


# Komparator

- ❑ Kebanyakan A-D converter menggunakan komparator sebagai bagian proses konversi
- ❑ Komparator membandingkan 2 sinyal A dan B
  - jika  $A > B \rightarrow$  output salah satu logika (mis. 0)
  - Jika  $B > A \rightarrow$  kebalikannya (mis. 1)
- ❑ Komparator bisa dibuat dari op-Amp tanpa feedback.



# Rangkaian Flash ADC



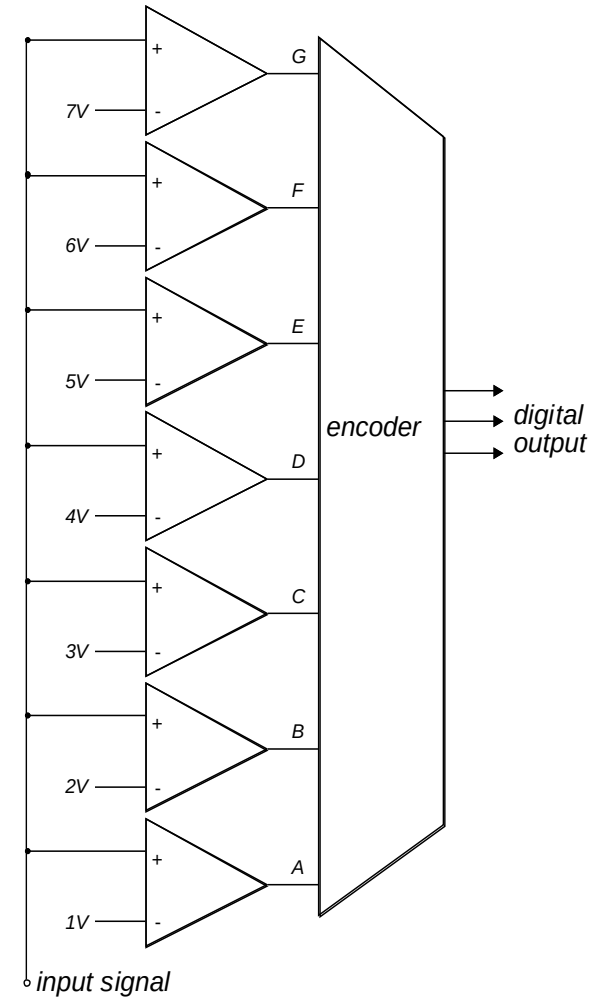
---

# Bagaimana cara kerja Flash?

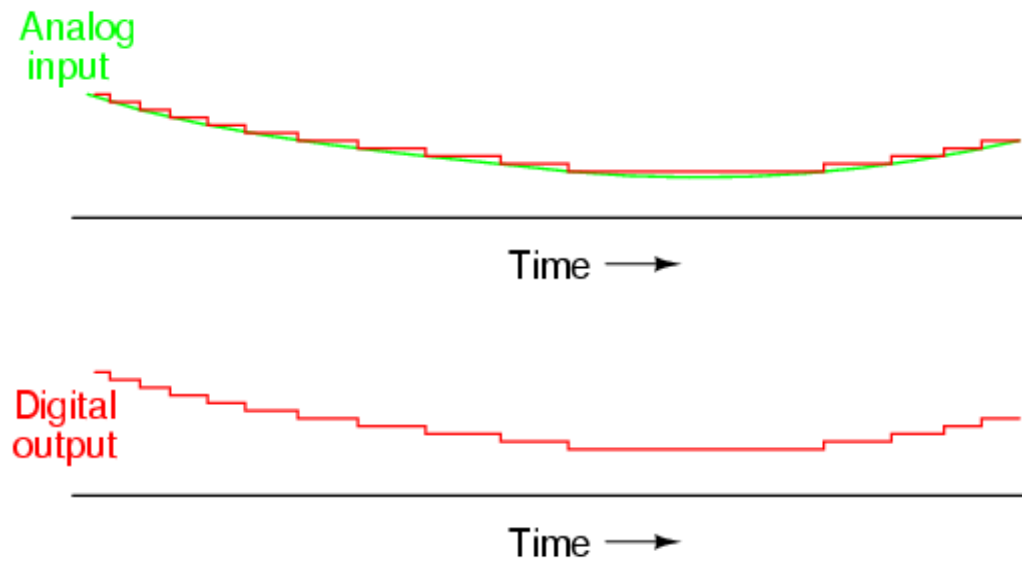
- Ketika input analog melewati tegangan referensi → Output HIGH
  - Enkoder mengeluarkan bilangan biner berbasis pada input yang jadi prioritas (*highest-order active input*), input lain (yang tidak prioritas) diabaikan.
-

# Flash Converter

| Converter<br>input<br>range (V) | Comparator Outputs |   |   |   |   |   |   | Encoder Output |
|---------------------------------|--------------------|---|---|---|---|---|---|----------------|
|                                 | A                  | B | C | D | E | F | G |                |
| <1                              | 0                  | 0 | 0 | 0 | 0 | 0 | 0 | 000            |
| >1-2                            | 1                  | 0 | 0 | 0 | 0 | 0 | 0 | 001            |
| >2-3                            | 1                  | 1 | 0 | 0 | 0 | 0 | 0 | 010            |
| >3-4                            | 1                  | 1 | 1 | 0 | 0 | 0 | 0 | 011            |
| >4-5                            | 1                  | 1 | 1 | 1 | 0 | 0 | 0 | 100            |
| >5-6                            | 1                  | 1 | 1 | 1 | 1 | 0 | 0 | 101            |
| >6-7                            | 1                  | 1 | 1 | 1 | 1 | 1 | 0 | 110            |
| >7                              | 1                  | 1 | 1 | 1 | 1 | 1 | 1 | 111            |



# ADC Output



# Flash

## Kelebihan

- Simple (operational theory)
- Paling Efisien (*speed*, Sangat cepat)
  - Hanya terbatas delay dari komparator dan gerbang

## Kekurangan

- Resolusi Rendah
- Mahal
- Tiap penambahan satu bit → jumlah komparator dobel
  - mis. untuk 8 bit, 256 perlu 256 komparator

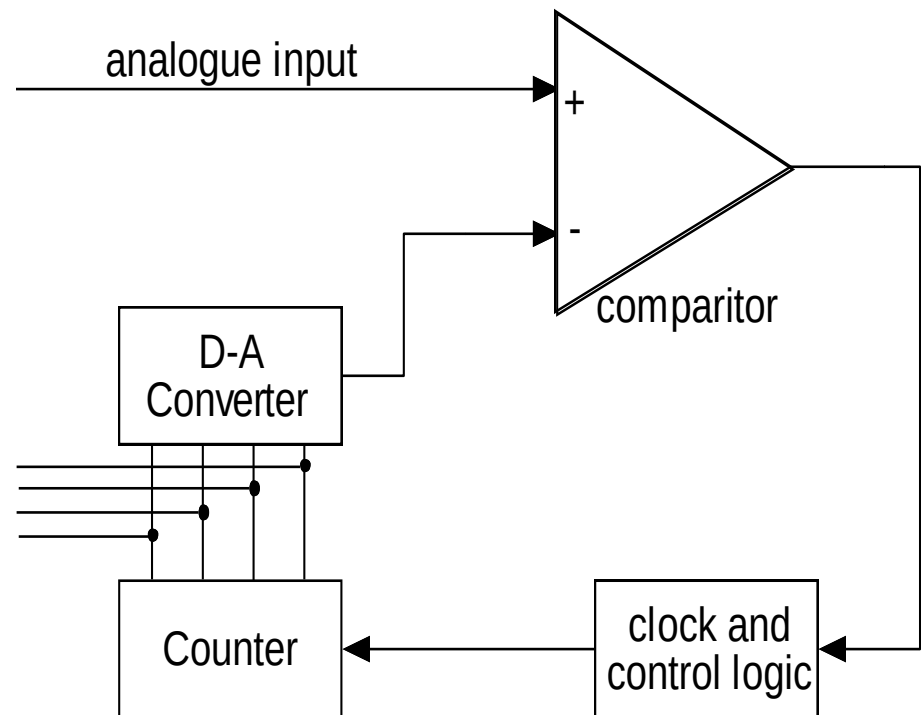
# Integrating/Ramp ADC

# Counter-ramp Converter

- Terdiri dari D-A converter, komparator, pencacah, *clock* dan kontrol logika.

- Langkah Konversi

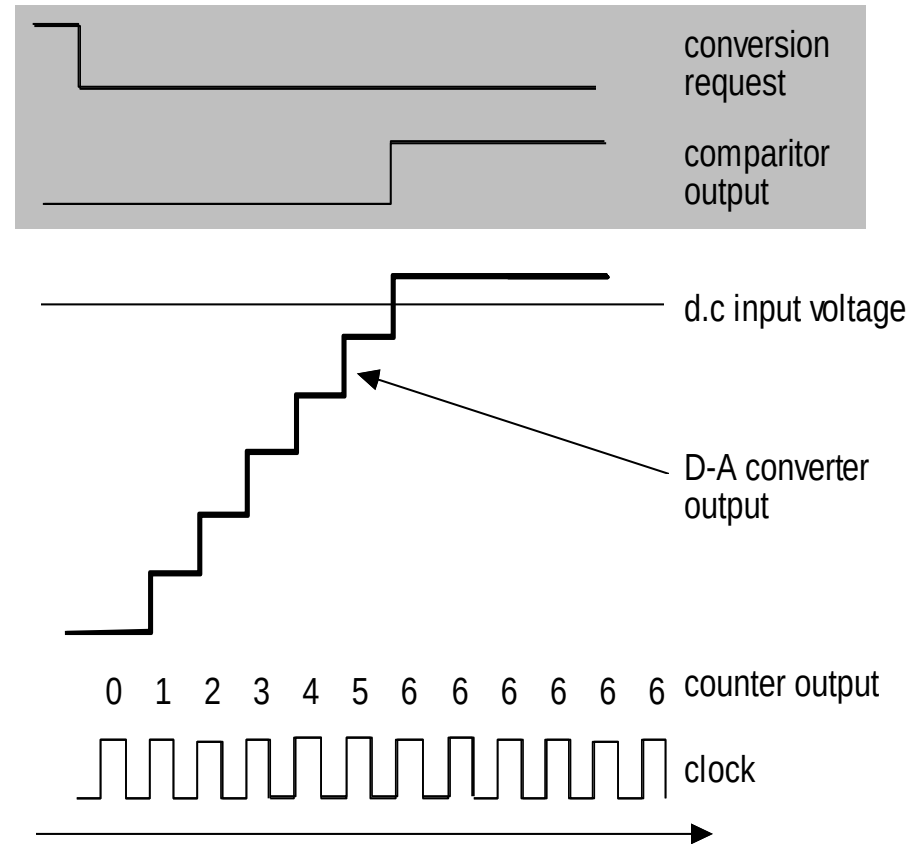
- Sinyal (*conversion request*) dikirimkan ke konverter dan pencacah di reset ke NOL.
- Sinyal *clock* membuat pencacah mulai menghitung sampai tegangan referensi yang dihasilkan DAC, lebih besar dari input analog → output menjadi logika 1 → kontrol logika membuat pencacah berhenti
- Nilai pencacah = nilai digital



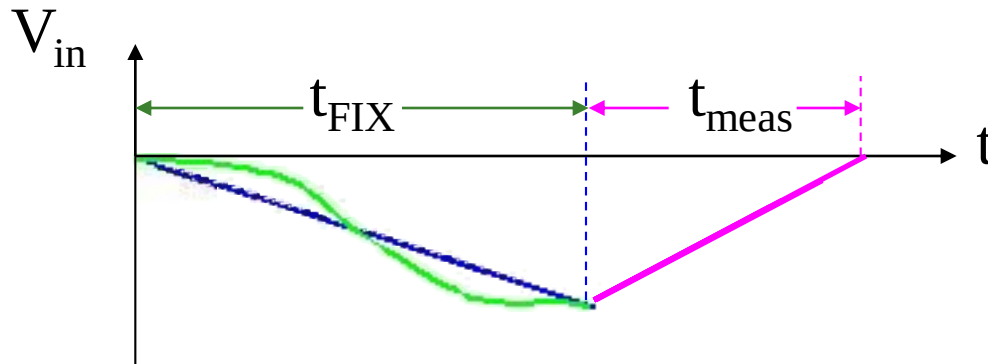


# Counter-ramp Converter

- ❑ Waktu antara awal dan akhir konversi = *conversion time*
- ❑ Akan memerlukan waktu yg lama jika tegangan input sangat besar → ambil *worst case* ketika menghitung waktu konversi

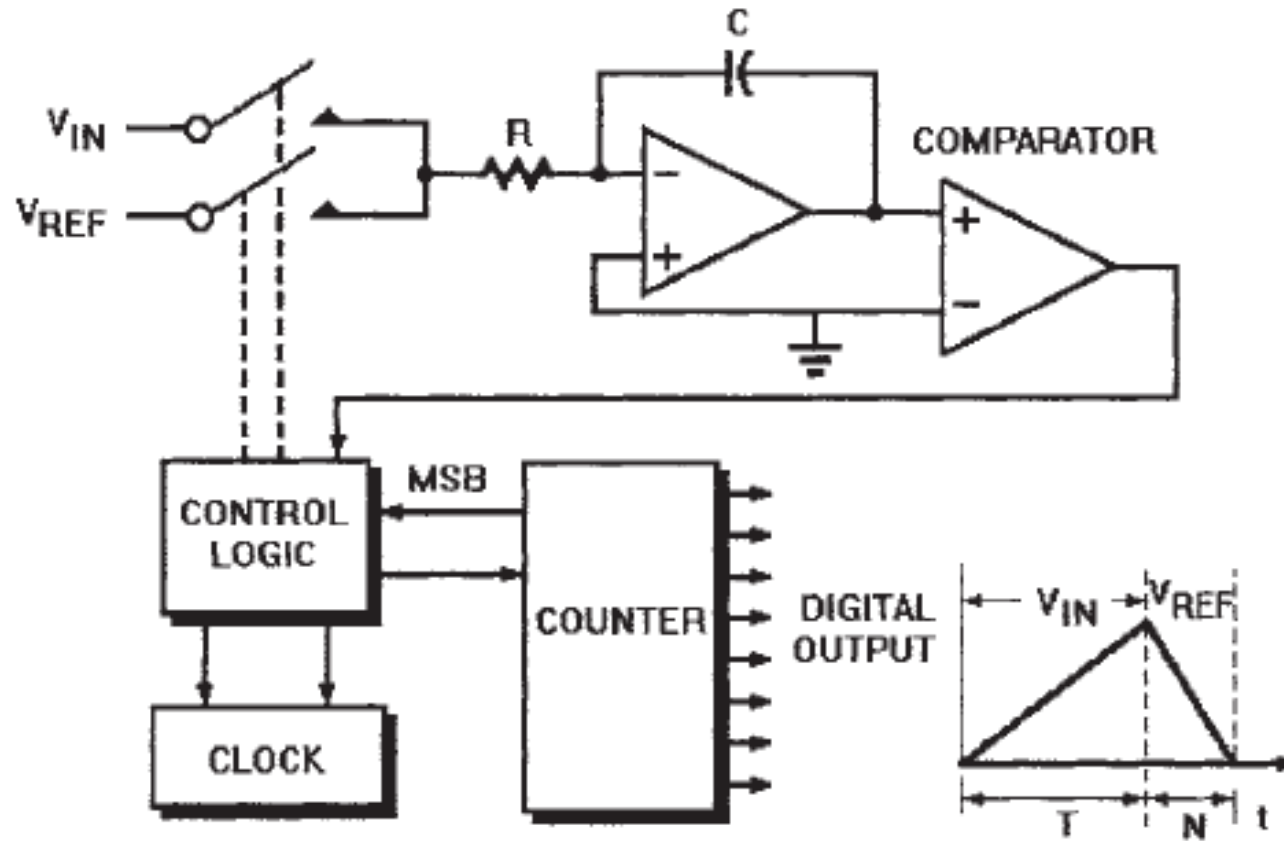


# Dual Slope Converter



- Sinyal yg di sampel mengisi kapasitor untuk waktu tertentu
- kemudian ADC membuang muatan kapasitor dan menghitung waktunya → output berupa bit, semakin lama waktu → semakin tinggi nilai bit

# *Dual-slope integrating converters*



---

# *Dual Slope Converter*

## Keuntungan

- ❖ Sinyal input rata-rata
- ❖ Ketahanan terhadap derau lebih tinggi daripada jenis lainnya
- ❖ Akurasi tinggi

## Kekurangan

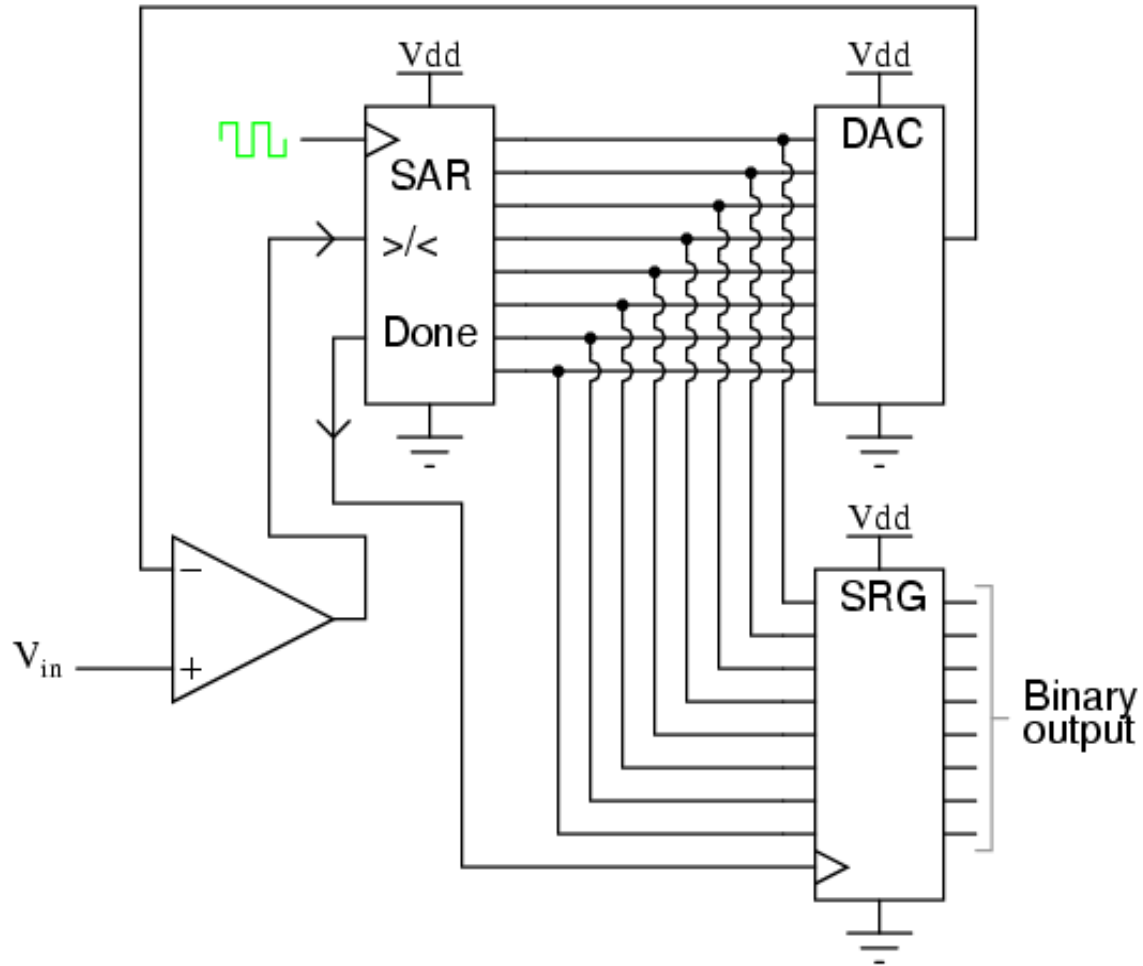
- ❖ lambat
  - ❖ Perlu komponen eksternal presisi tinggi untuk mencapai akurasi yang tinggi
-

# Successive Approximation ADC

# Successive Approximation ADC

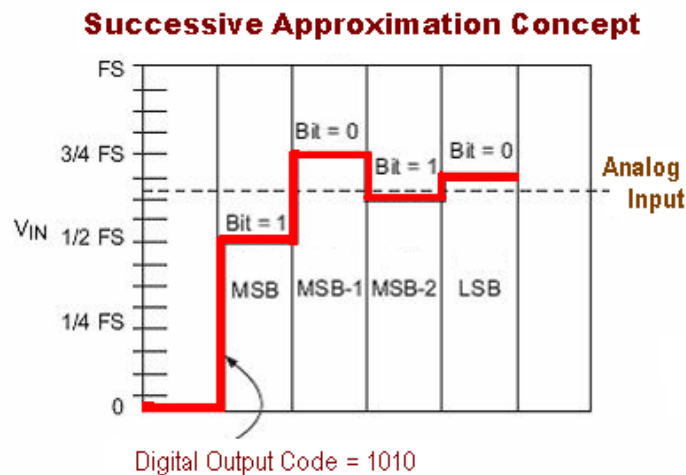
- mempunyai Successive Approximation Register (SAR) sebagai pengganti Kapasitor
- Dari pada menghitung deret biner, register ini membandingkan semua nilai bit dimulai dari MSB s/d LSB.
- Register memonitor output komparator untuk melihat apakah biner hitungan lebih besar atau lebih kecil dari sinyal input analog → hasil merubah bit

# Rangkaian SAR ADC

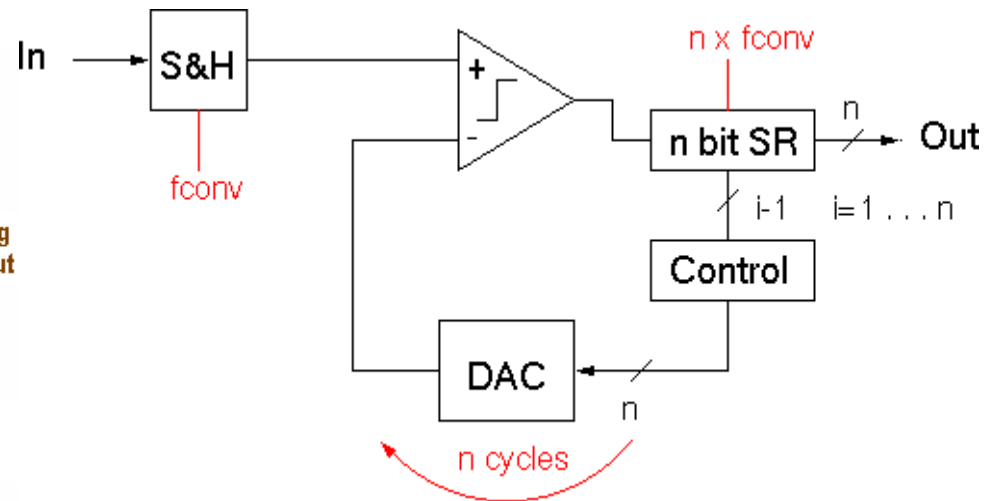


# Successive Approx. ADCs

## Concept

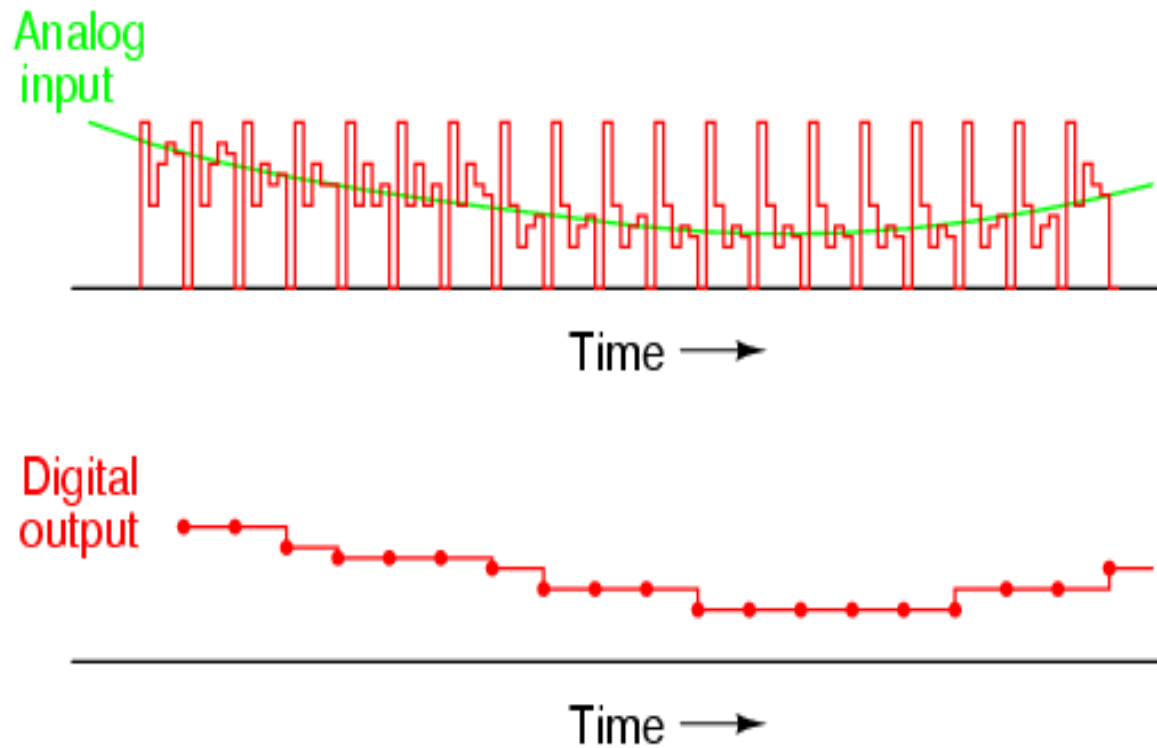


## Implementation





# Output



# Contoh SAR ADC

- Resolusi 10 bit atau 0.0009765625V
- $V_{in} = 0.6$  volts
- $V_{ref} = 1$  volts
- Berapa nilai digitalnya?

| Bit | Voltage     |
|-----|-------------|
| 9   | .5          |
| 8   | .25         |
| 7   | .125        |
| 6   | .0625       |
| 5   | .03125      |
| 4   | .015625     |
| 3   | .0078125    |
| 2   | .00390625   |
| 1   | .001952125  |
| 0   | .0009765625 |



# Successive Approximation

- Selanjutnya hitung MSB-1 (bit 8)
  - Bandingkan  $V_{in}=0.6\text{ V}$  to  $V=V_{ref}/2 + V_{ref}/4= 0.5+0.25 =0.75\text{V}$
  - karena  $0.6<0.75$ , MSB = 0
- Hitung MSB-2 (bit 7)
  - Ambil tegangan yg membuat ON (Bit 9) dan tambahkan ke  $V_{ref}/8$ , kemudian bandingkan dg  $V_{in}$
  - Bandingkan  $V_{in}$  dg  $(0.5+V_{ref}/8)=0.625$
  - Karena  $0.6<0.625$ , MSB = 0 (off)

|   |   |   |  |  |  |  |  |  |  |
|---|---|---|--|--|--|--|--|--|--|
| 1 | 0 | 0 |  |  |  |  |  |  |  |
|---|---|---|--|--|--|--|--|--|--|

# Successive Approximation

- Hitung MSB-3 (bit 6)
  - Ambil tegangan yg membuat ON (bit 9) dan tambahkan ke  $V_{\text{ref}}/16$ , kemudian bandingkan dg  $V_{\text{in}}$
  - Bandingkan  $V_{\text{in}}$  dg  $V = 0.5 + V_{\text{ref}}/16 = 0.5625$
  - Karena  $0.6 > 0.5625$ , MSB-3=1 (on)

| MSB | MSB-1 | MSB-2 | MSB-3 | ... |  |  |  |  |  |
|-----|-------|-------|-------|-----|--|--|--|--|--|
| 1   | 0     | 0     | 1     |     |  |  |  |  |  |

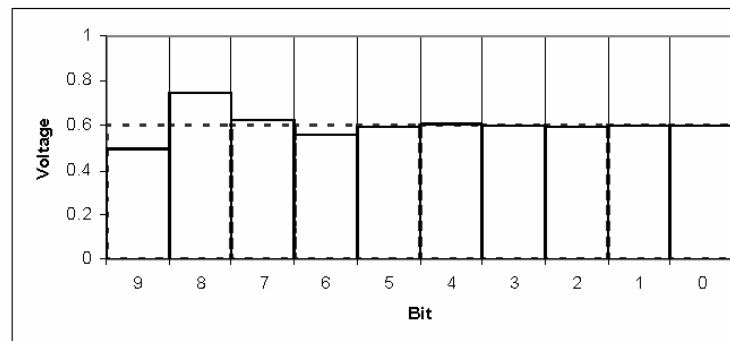
# Successive Approximation

- Proses berlanjut sampe LSB.

•Digital Results:

| MSB | MSB-1 | MSB-2 | MSB-3 | ... |   |   |   |   | LSB |
|-----|-------|-------|-------|-----|---|---|---|---|-----|
| 1   | 0     | 0     | 1     | 1   | 0 | 0 | 1 | 1 | 0   |

•Results:  $\frac{1}{2} + \frac{1}{16} + \frac{1}{32} + \frac{1}{256} + \frac{1}{512} = .599609375 \text{ V}$



# Successive Approximation

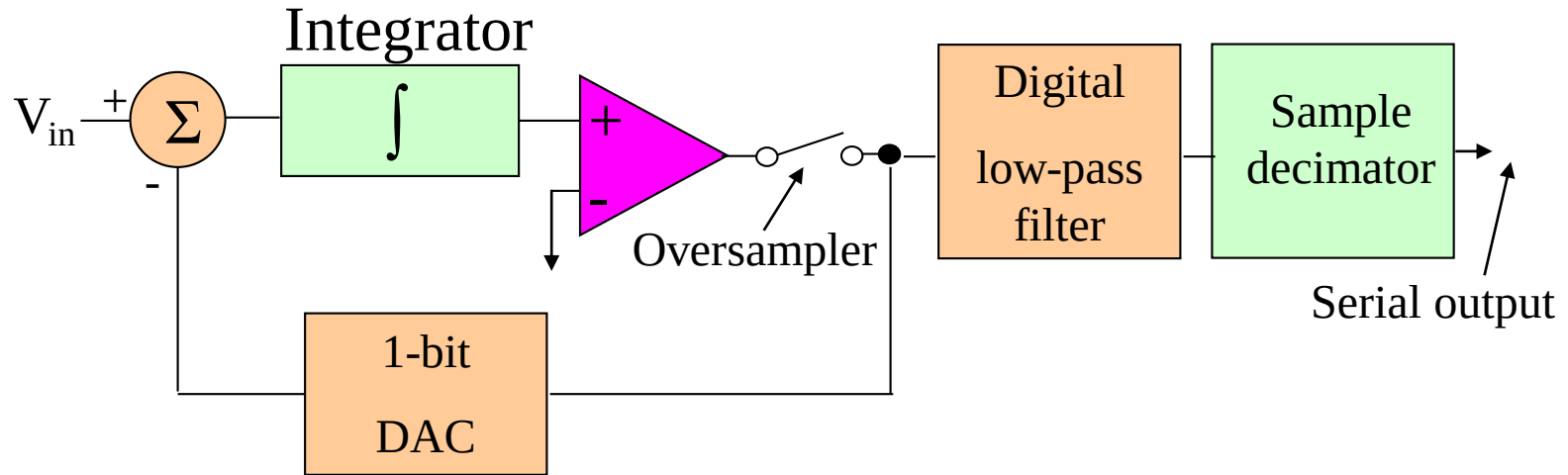
## Keuntungan

- Mampu berkecepatan tinggi dan handal
- Akurasi menengah dibanding tipe lain
- Tradeoff antara kecepatan dan harga
- Mampu mengeluarkan angka biner secara seial (satu bit per waktu).

## kerugian

- Resolusi tinggi akan menyebabkan successive approximation ADC's menjadi lambat
- Kecepatan terbatas s/d ~5Mbps

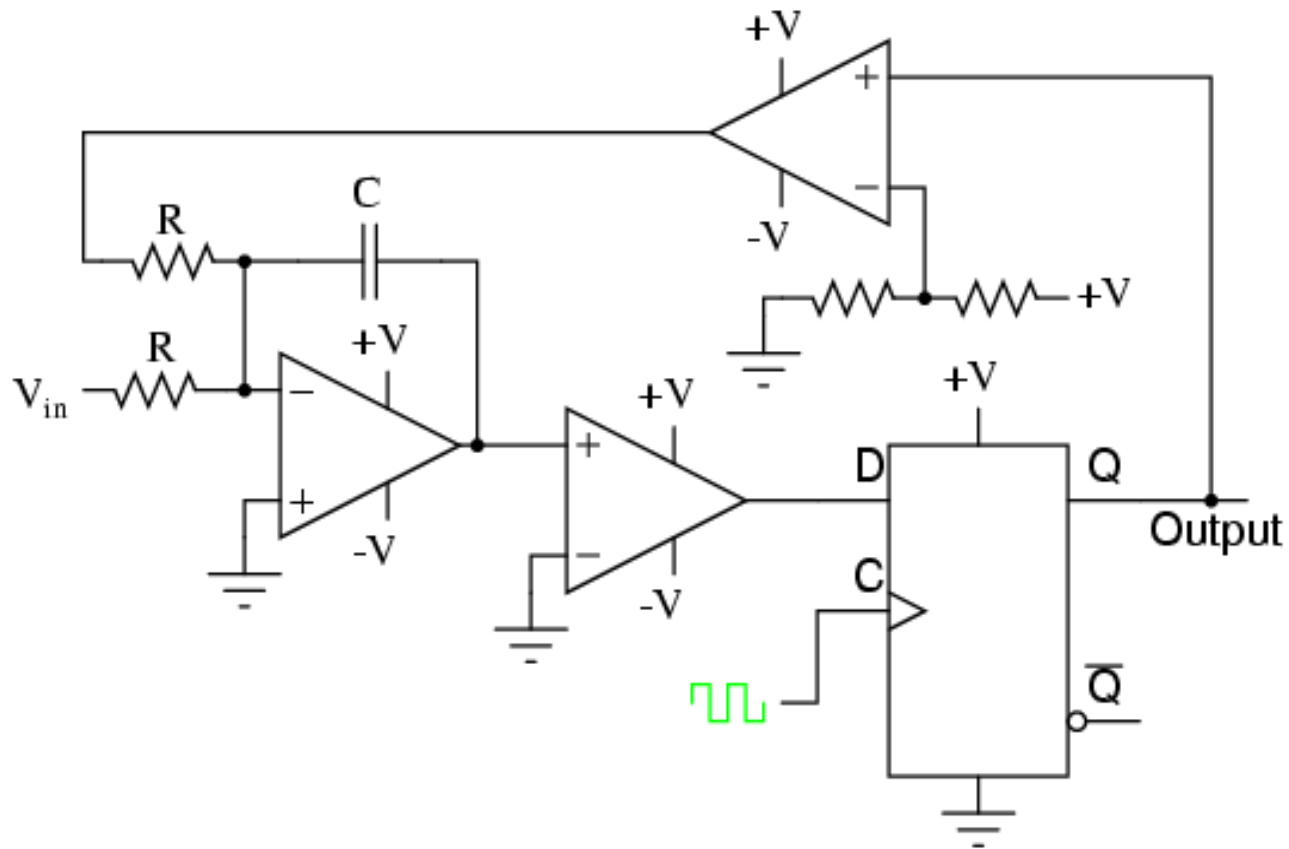
# Sigma-delta ADC



- Oversampled input signal masuk ke integrator
- Output integrator dibandingkan dg GND
- Iteratasi agar menghasilkan serial bitstream
- Output berupa serial bit stream dg jumlah 1 proporsional terhadap  $V_{in}$



# Sigma Delta ADC



# Sigma Delta ADC

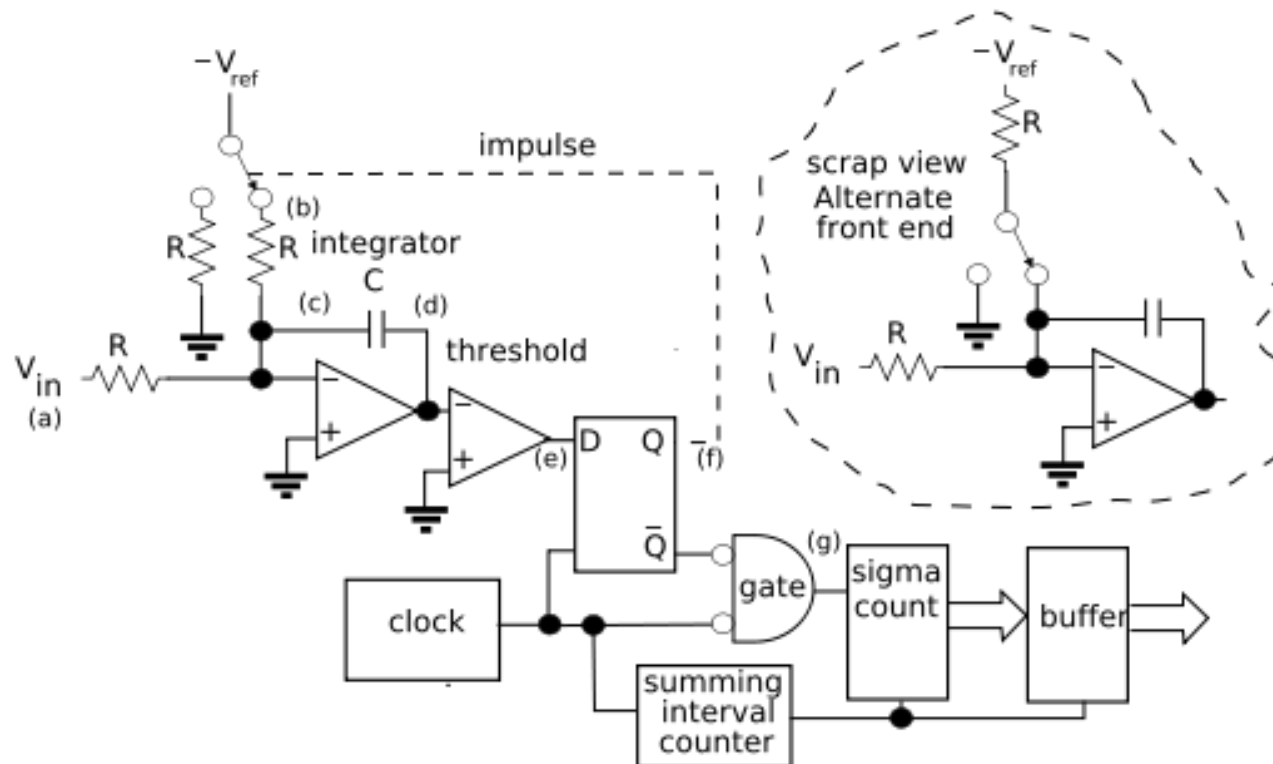


Fig. 1b - Circuit Diagram

# Sigma Delta ADC

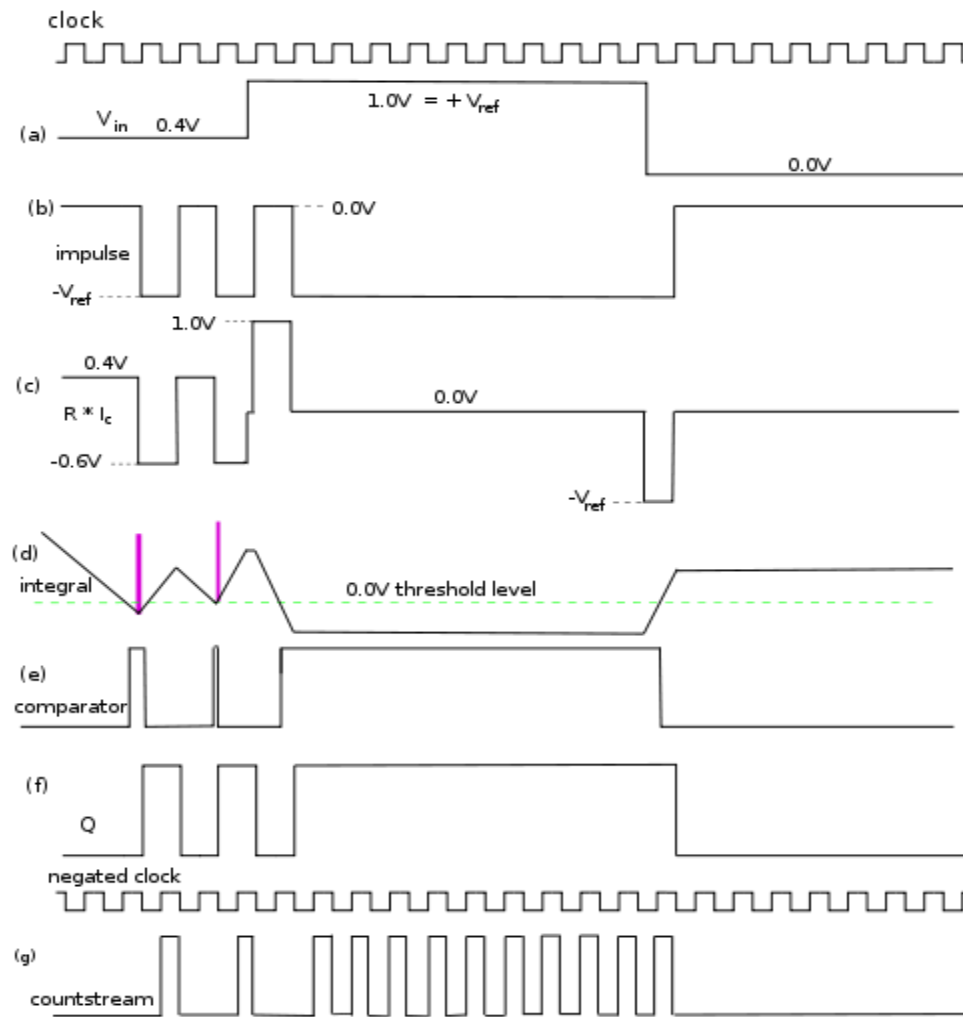
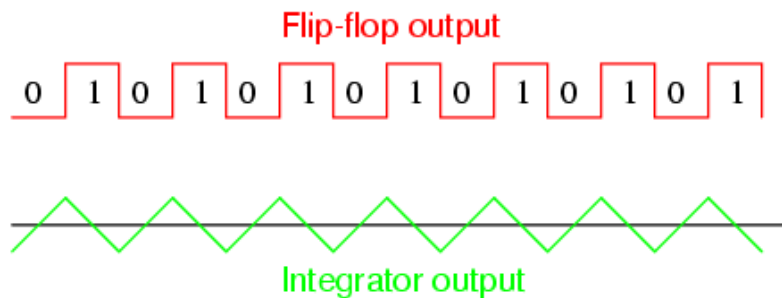


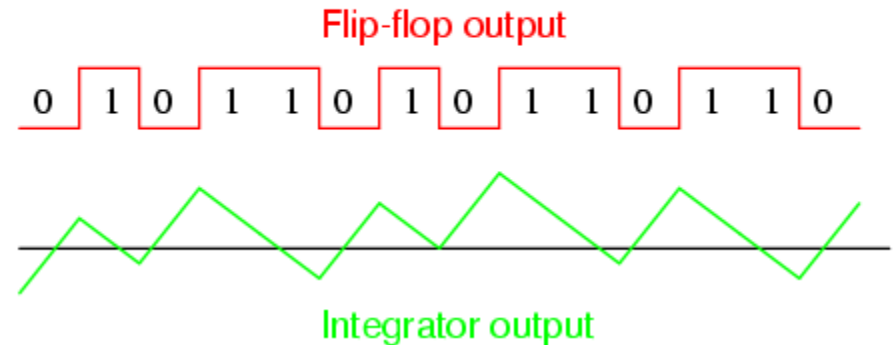
Fig. 1c - ADC waveforms

# Outputs of Delta Sigma

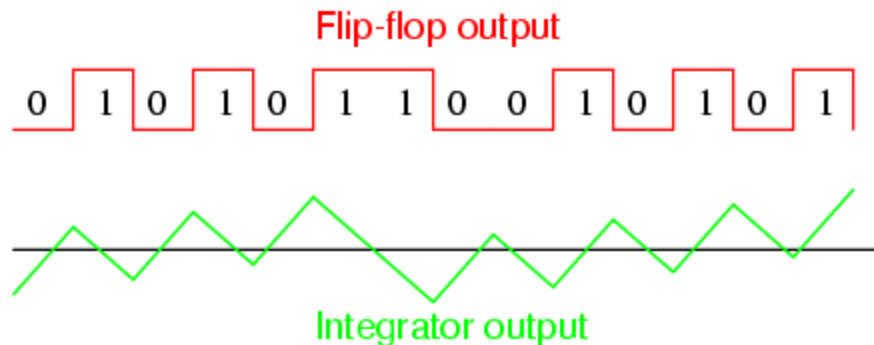
$\Delta\Sigma$  converter operation with  
0 volt analog input



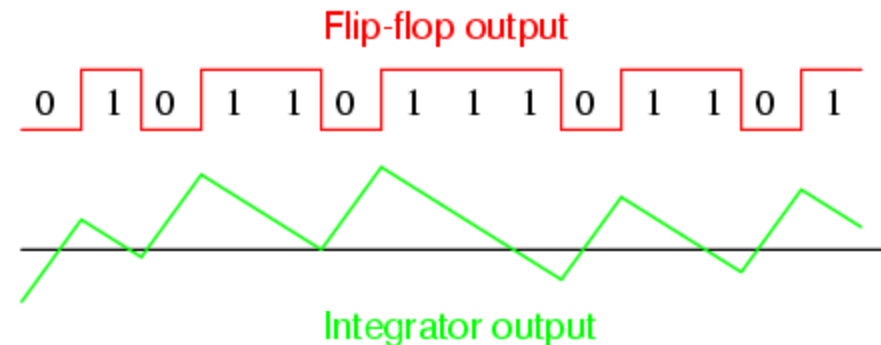
$\Delta\Sigma$  converter operation with  
medium negative analog input



$\Delta\Sigma$  converter operation with  
small negative analog input



$\Delta\Sigma$  converter operation with  
large negative analog input



---

# Sigma-Delta

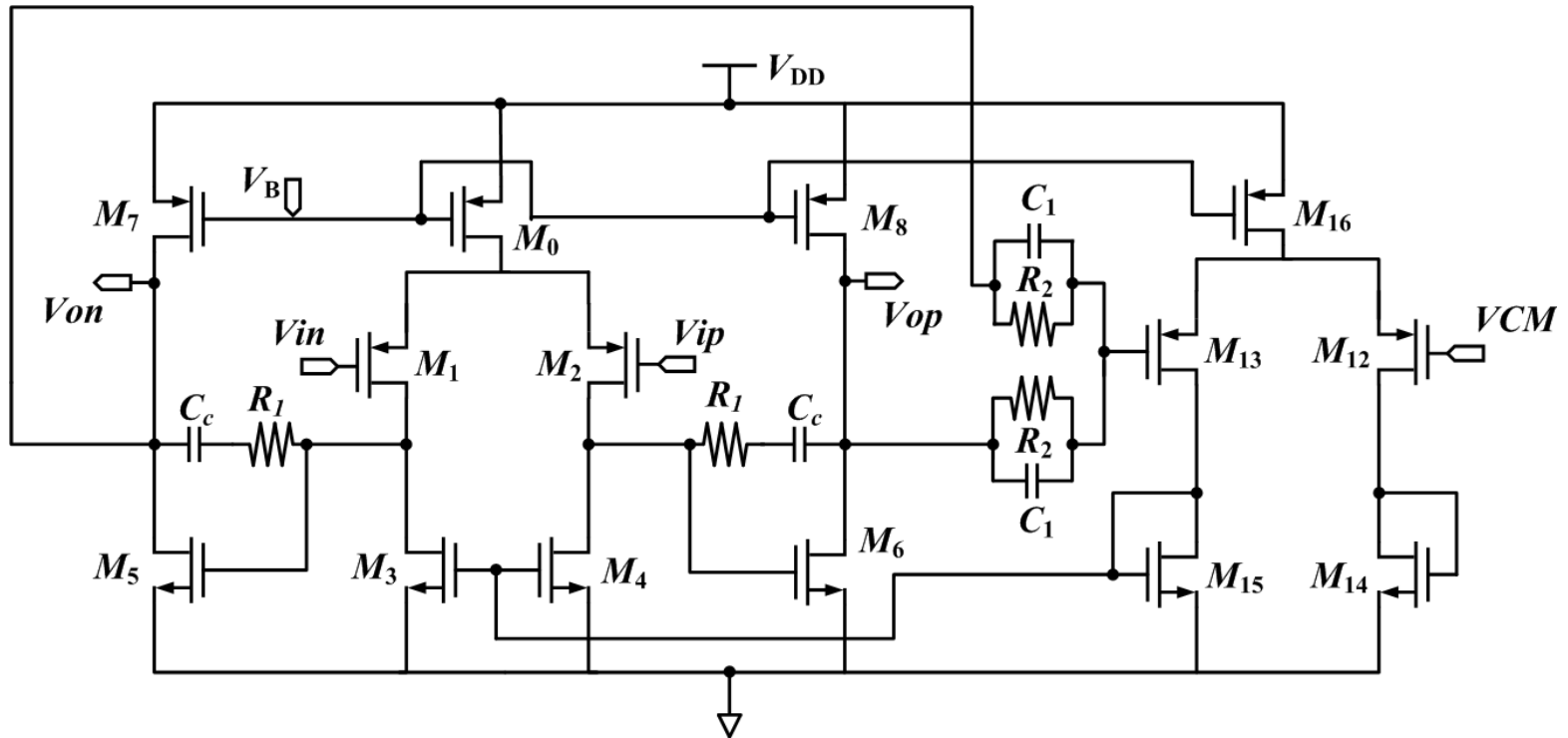
## Keuntungan

- Resolusi Tinggi
- Tidak diperlukan komponen tambahan yang presisi

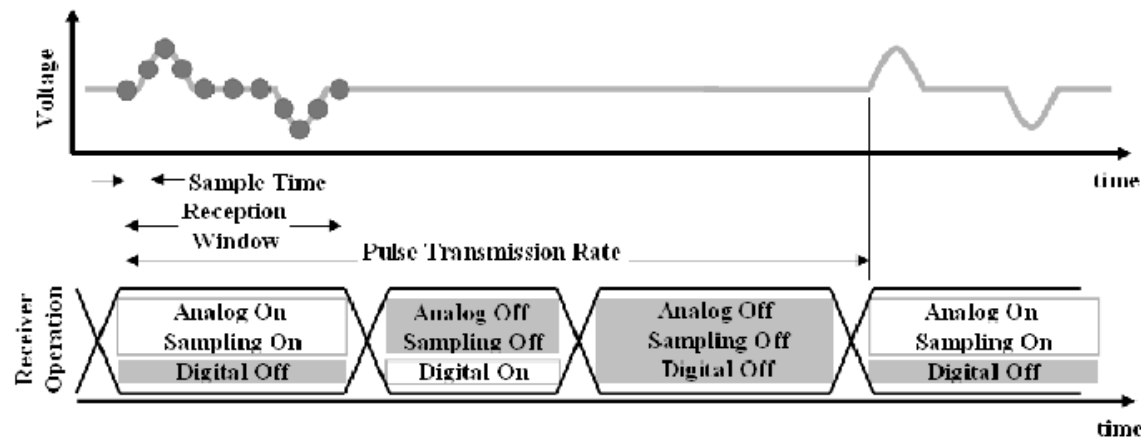
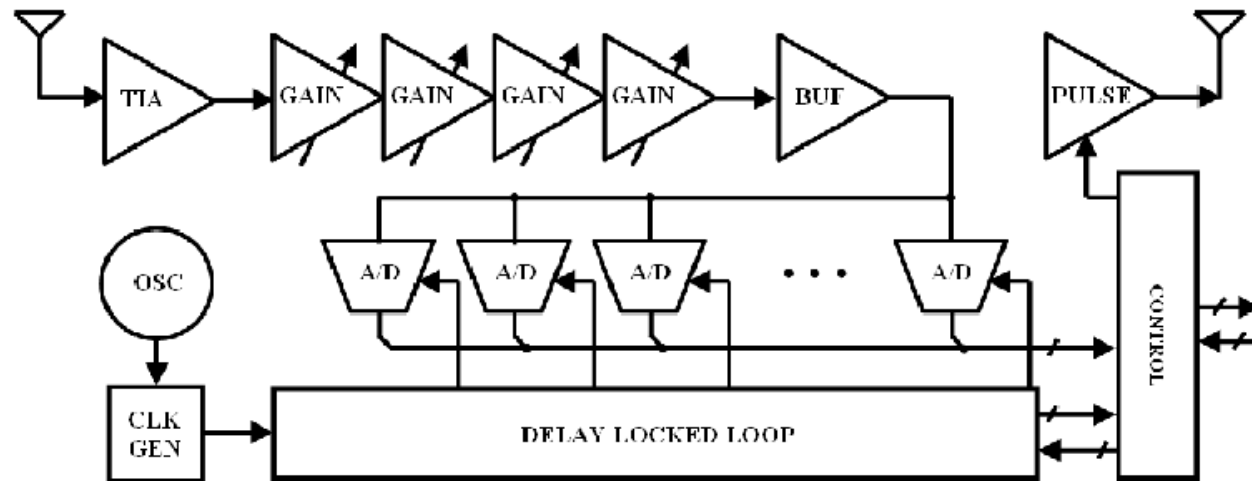
## Kekurangan

- Lambat karena *oversampling*

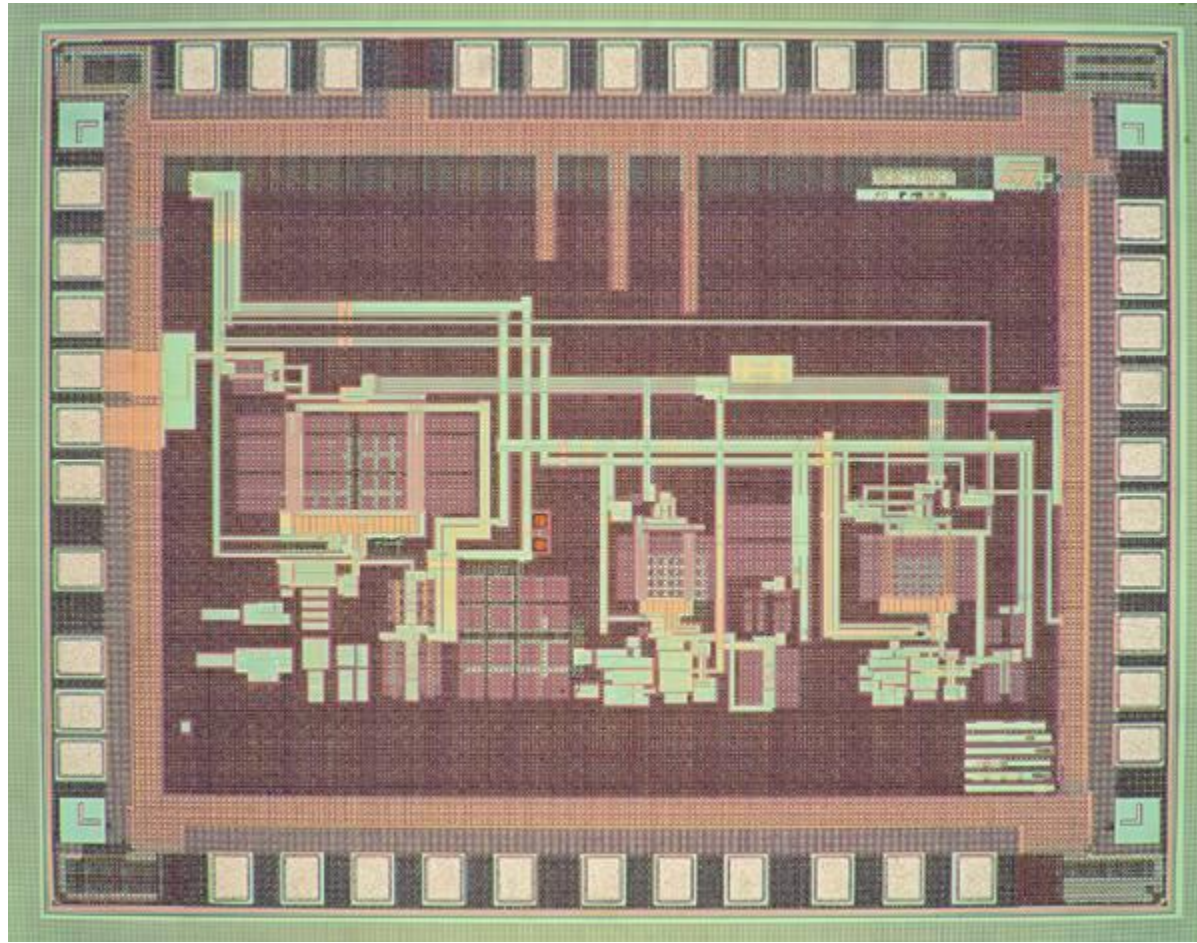
# Sigma-Delta



# Sigma-Delta



# Sigma-Delta

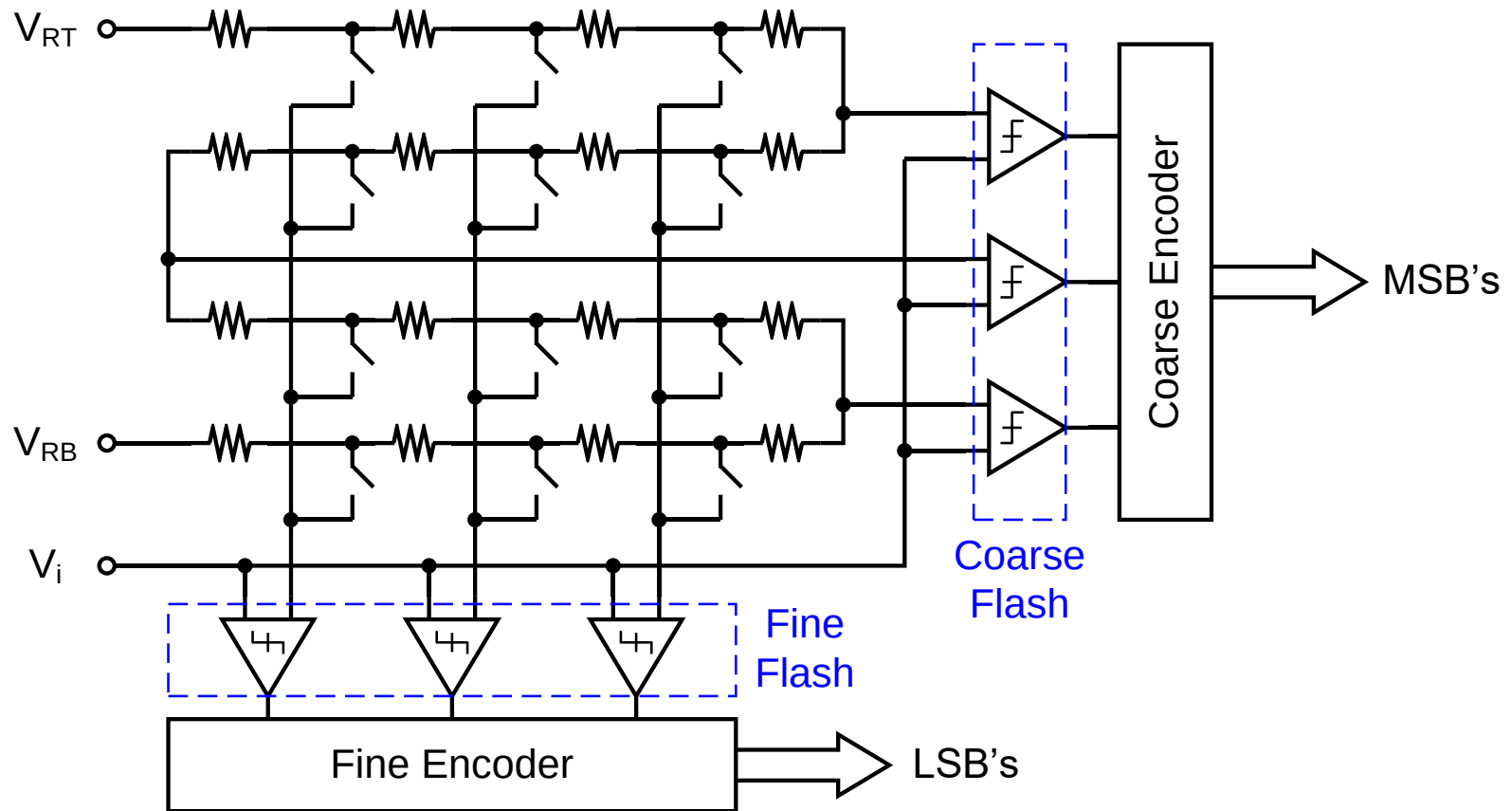




# Subranging ADC

---

# Subranging ADC Architecture



# Subranging ADC

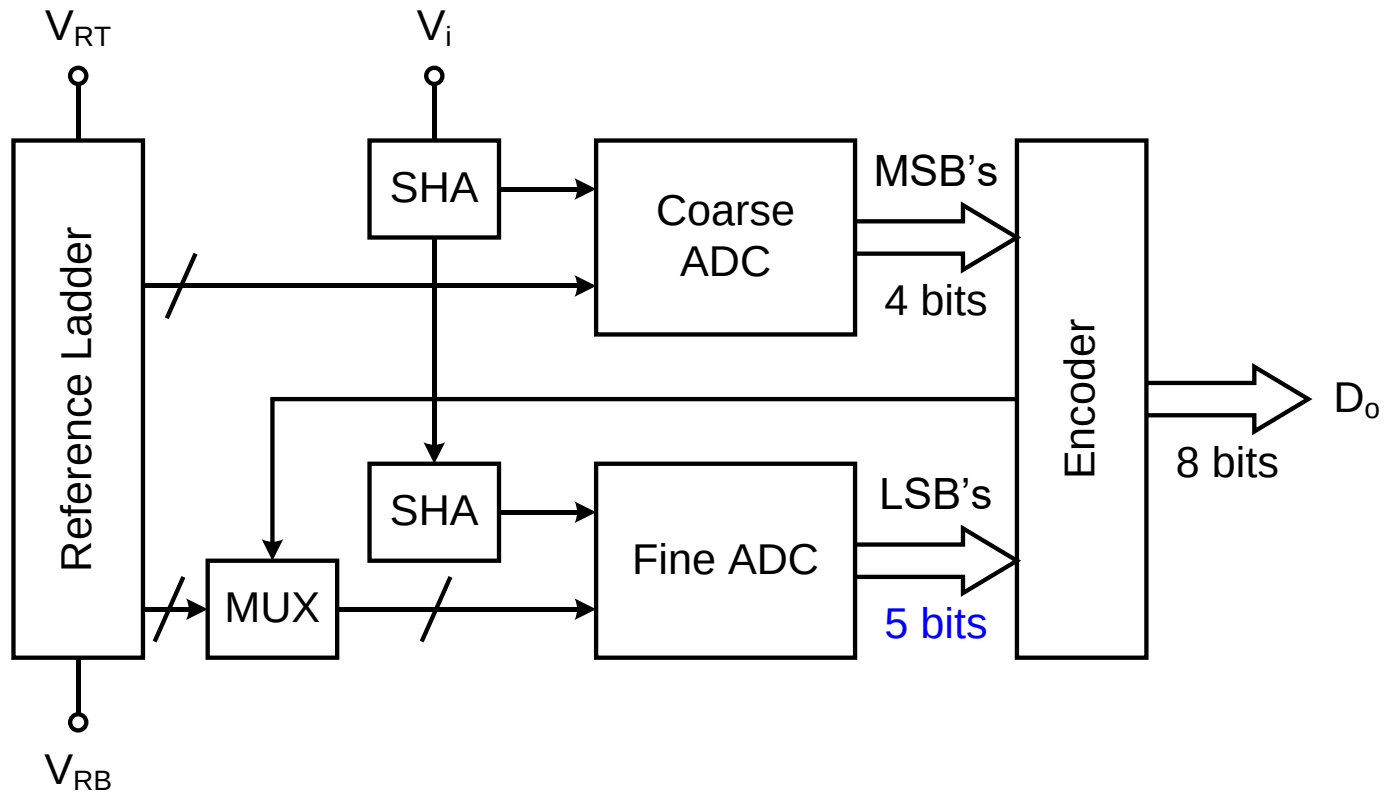
## Features

- Reduced complexity –  $2 \cdot (2^{N/2} - 1)$  comparators – relative to flash
- Reduced  $C_{in}$ , area, and power consumption
- No residue amplifier required (compare to pipelined ADC)

## Limitations

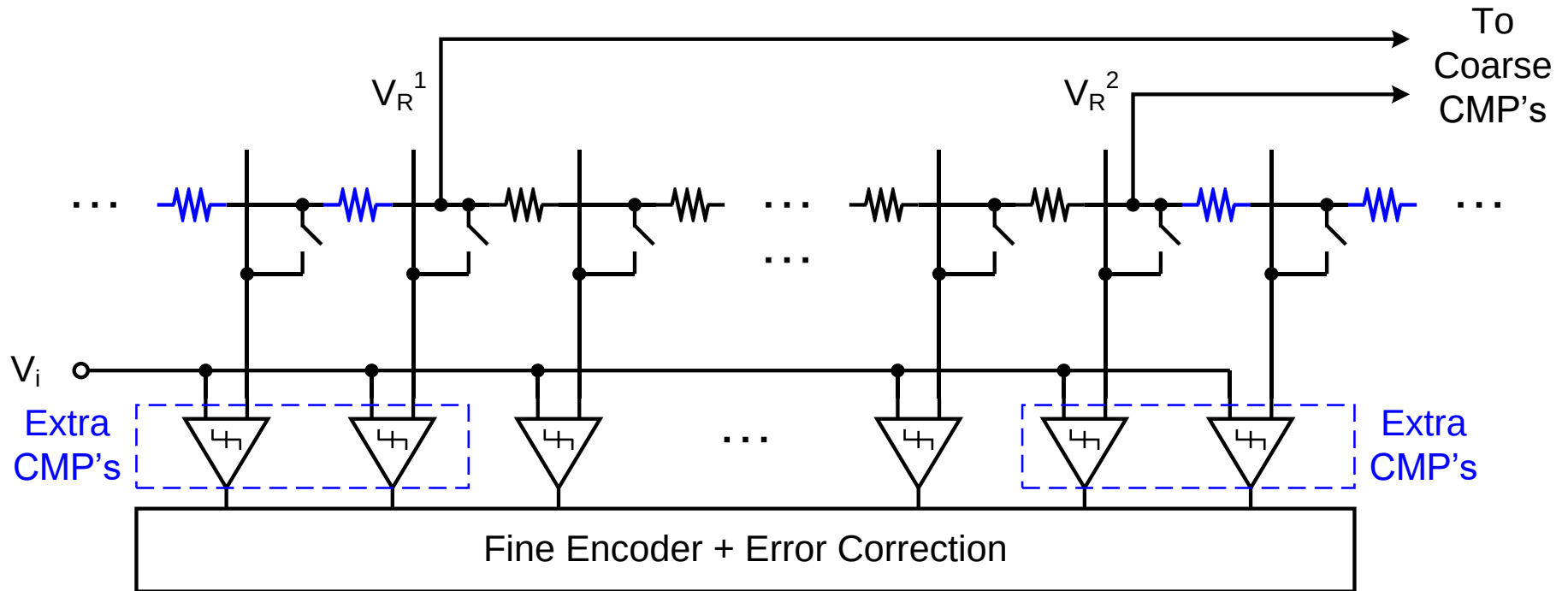
- Typically 3 clock phases per conversion
  - Sample
  - Coarse comparison
  - Fine comparison
- Typically two SHAs are required for the coarse and fine ADCs
- Fine comparator offset must be controlled to N-bit level
- Offset tolerance on coarse comparators can be relaxed with digital redundancy

# Typical Subranging Block Diagram



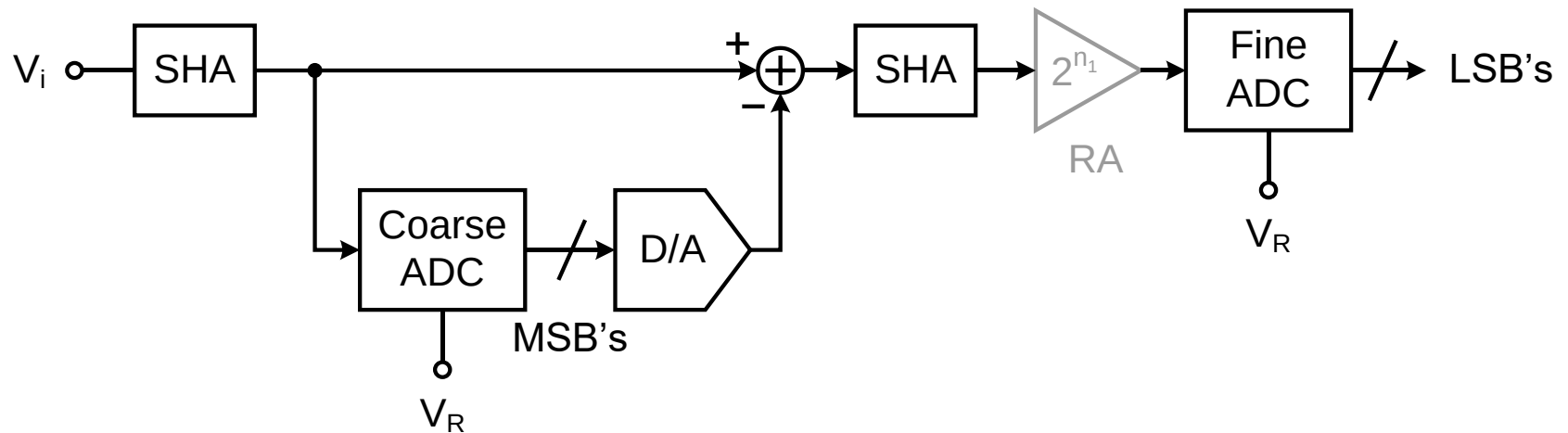
Redundancy in fine ADC provided by over- and under-range comparators

# Digital Redundancy in Fine ADC



The range of fine search extended on both sides

# Two-Step Subranging/Pipelined ADC



- Coarse-fine two-step subranging architecture
- Conversion residue produced instead of switching reference taps
- Residue gain can be provided to relax offset tolerance in fine ADC
- Very similar to the pipelined architecture