Charge redistribution SAR ADC in CMOS technology

Luca Pacher

INFN and University of Turin

Follow-up presentation for *Digital Electronics* course

Turin, 19/04/2013

pacher@to.infn.it

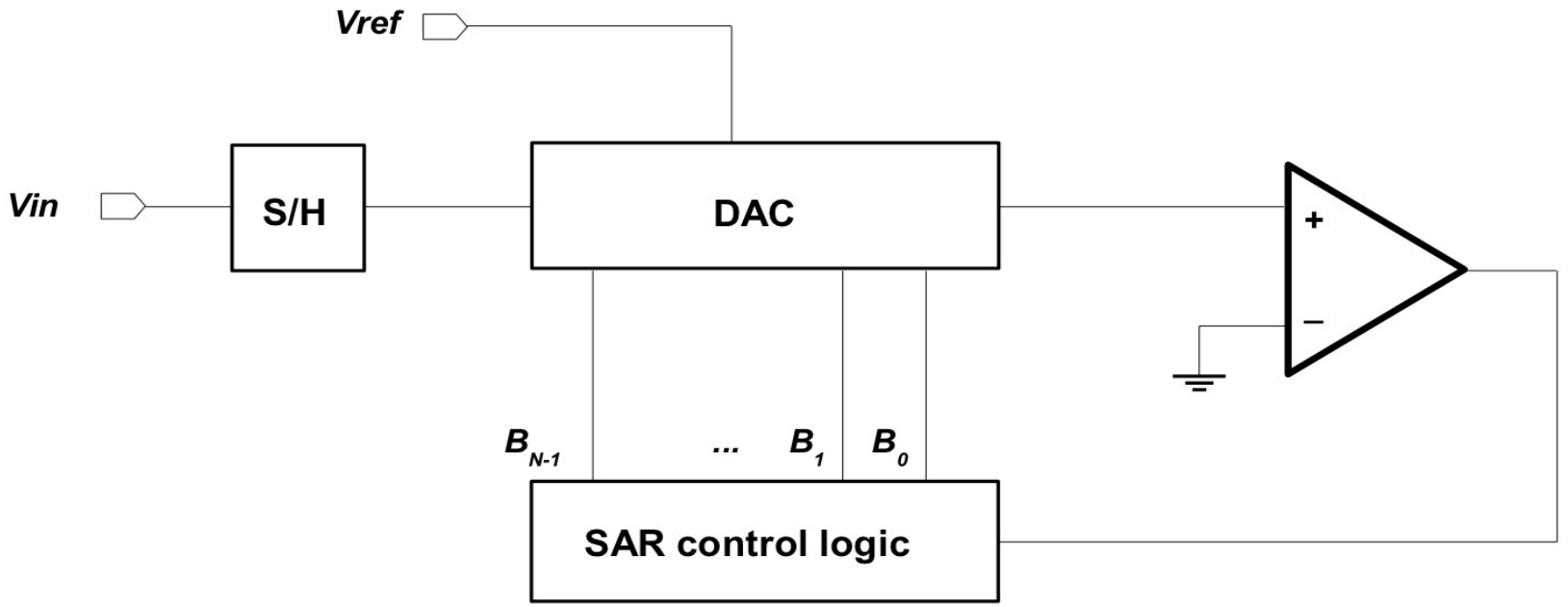
Outline

- *introduction*
- *general concept*
- *SAR algorithm*
- *CMOS full-design example*
- *non-linearities*
- *ADC characterization*

Introduction

- many different architectures of ADCs available
- specifications and requirements determine the choice of a particular topology
- ***Flash-ADC***
  - OK to convert a few bits
  - fastest topology, 1 cycle/overall conversion
  - HW requirements become relevant (N bits → 2^N -1 comparators + 1 encoder)
- ***multi-slope ADCs***, ***Wilkinson*** (e.g. TDC applications), ***pipeline***, etc.
- ***successive approximation register* (SAR)** ADCs offer the best trade-off between :
  - ***speed*** (medium speed applications, from tens of MS/s to a few GS/s)
  - ***resolution*** (5-10 bits)
  - ***HW requirements***
- largely employed in ***full-custom applications* (ASICs)** for particle tracking, energy and timing measurements in nuclear and particle physics electronics
  - 3-5 bit SAR ADCs can be used at the end of a ***front-end chain*** (e.g. ALICE SDD readout)
  - 8-10 bit SAR ADCs are used for ***on-chip monitoring*** (e.g. temperature, voltage supply etc.)

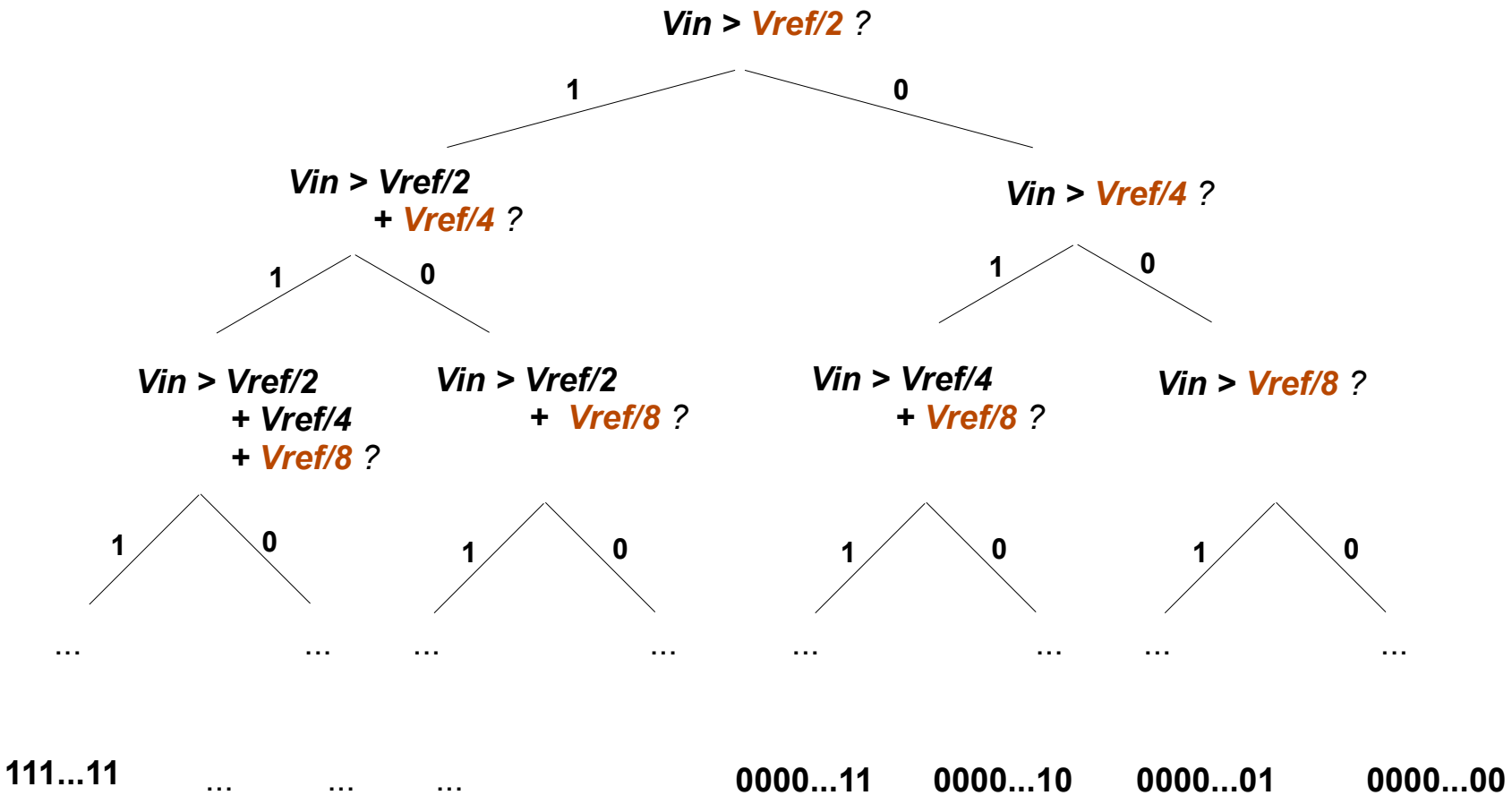
General concept



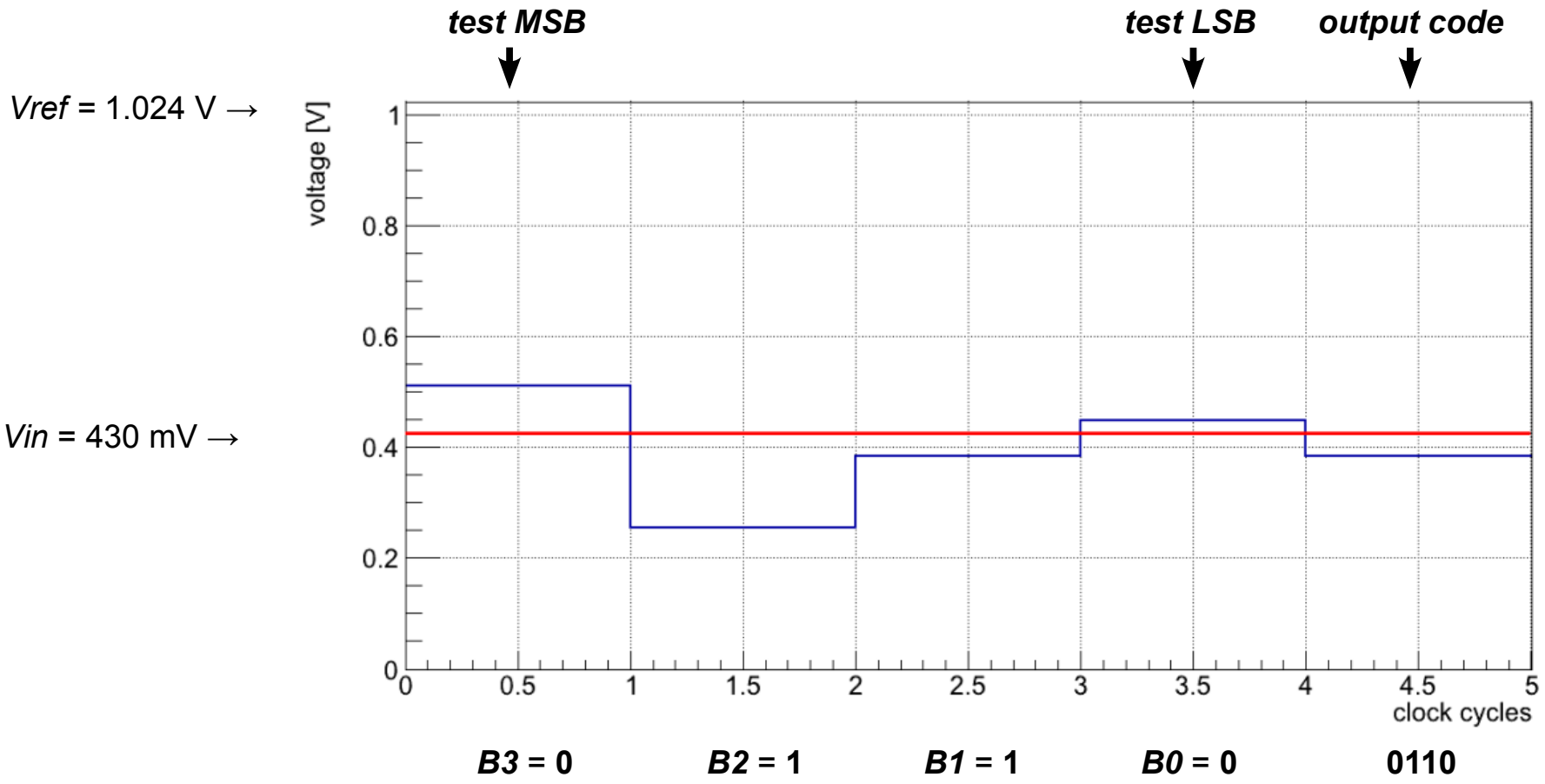
- ***mixed-mode*** system example (analog part + digital part)
- basic ***building blocks*** :
  - sample-and-hold (S/H) circuit
  - DAC
  - voltage comparator
  - SAR digital control logic
- ***charge-redistribution*** architectures use ***binary-weighted capacitor array**s*** to implement the DAC

Conventional SAR algorithm

trial-and-error search procedure ! (1bit /cycle)

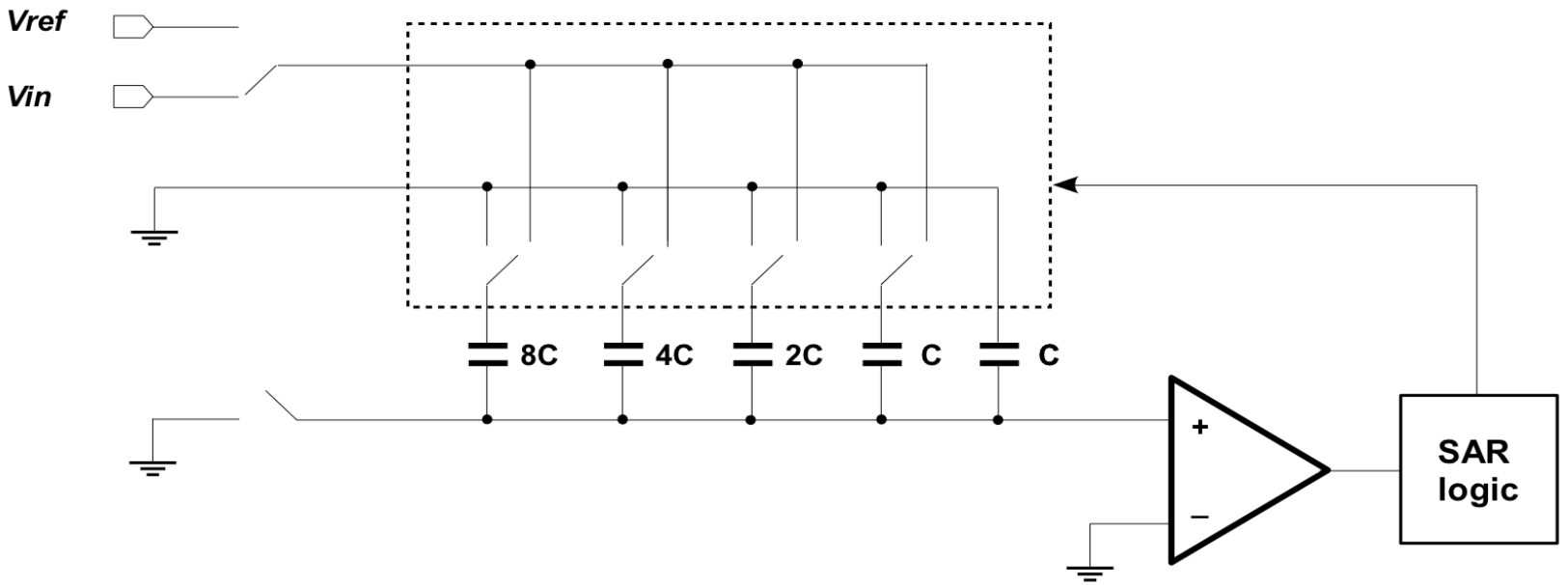


4 bit example



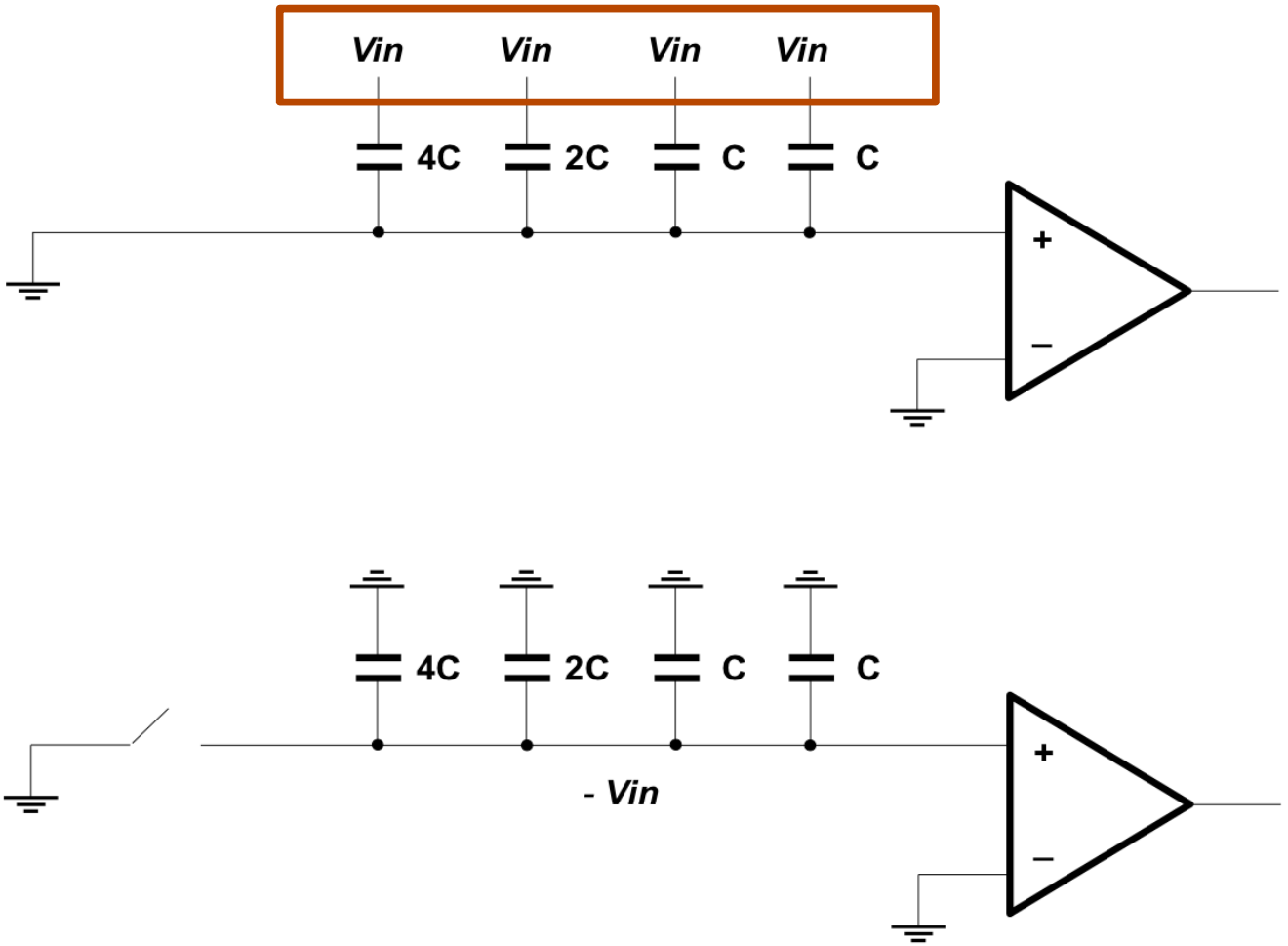
This is part of a set of high-level simulations developed in C/C++ and Python for my research work (see later)...

Charge-redistribution SAR ADCs

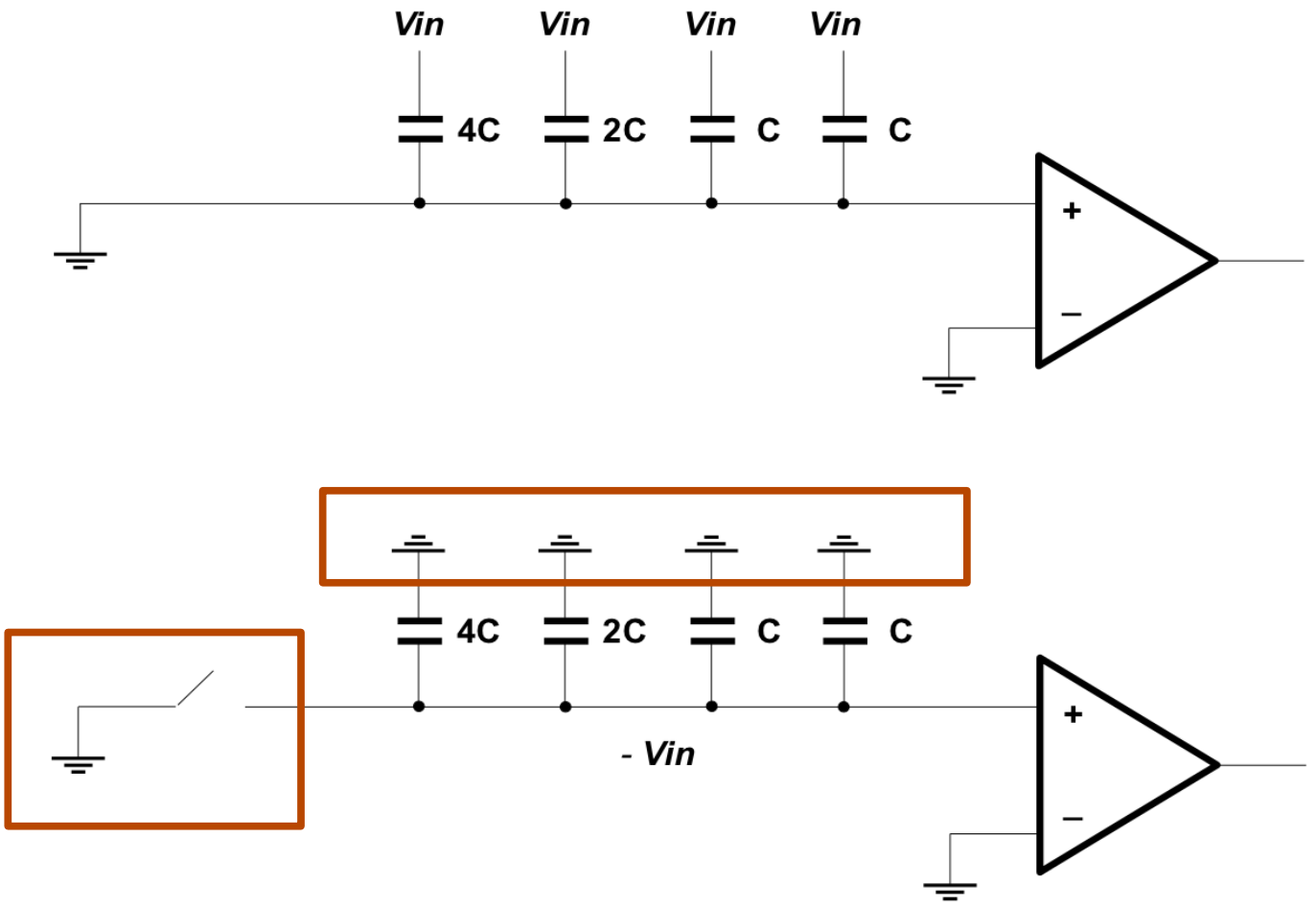


- charge redistribution SAR ADCs employ ***binary weighted capacitor arrays*** (C, C, 2C, 4C, ...)
- ***single-ended*** and ***fully-differential*** architectures are used
- in conventional architectures ***both the analog input and Vref are sampled on bottom plates***

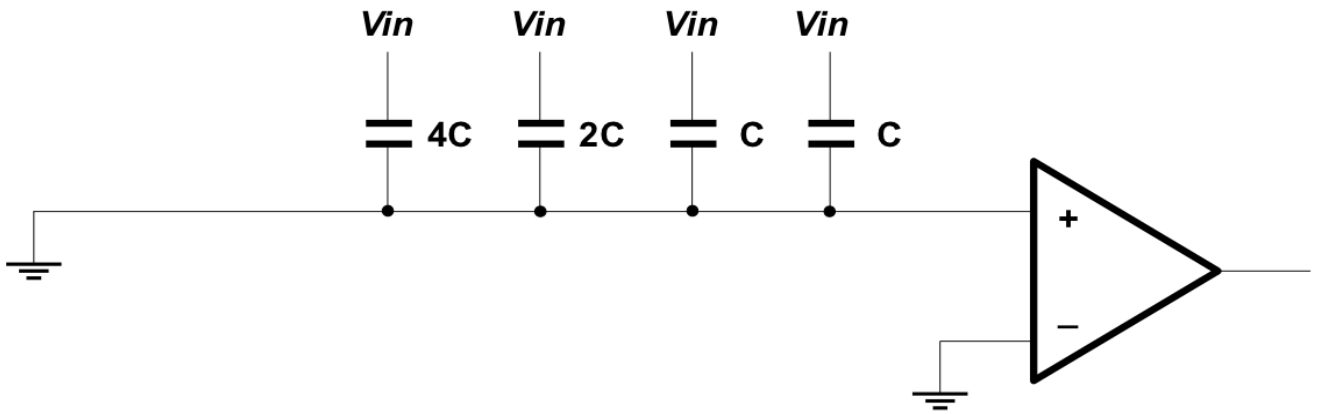
Sampling phase

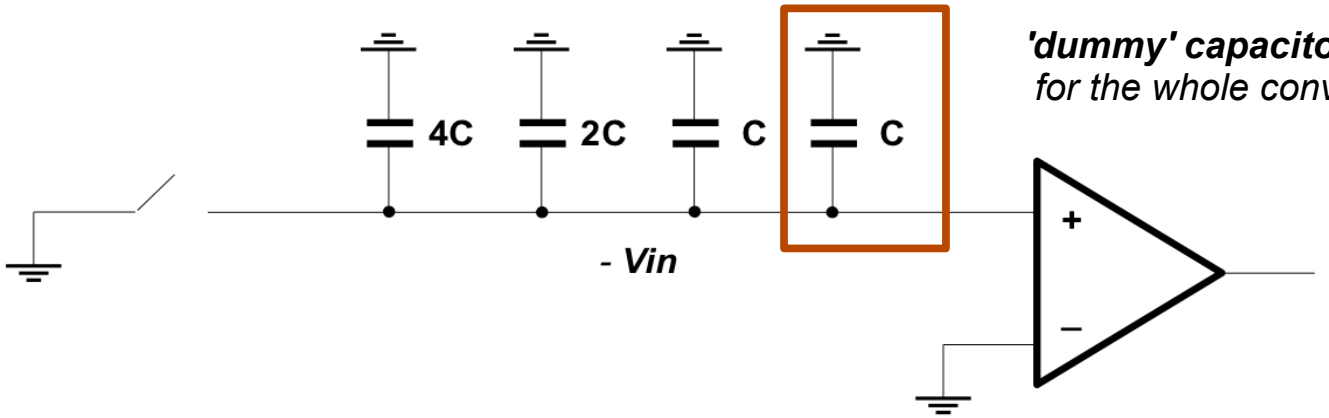


Sampling phase



Sampling phase

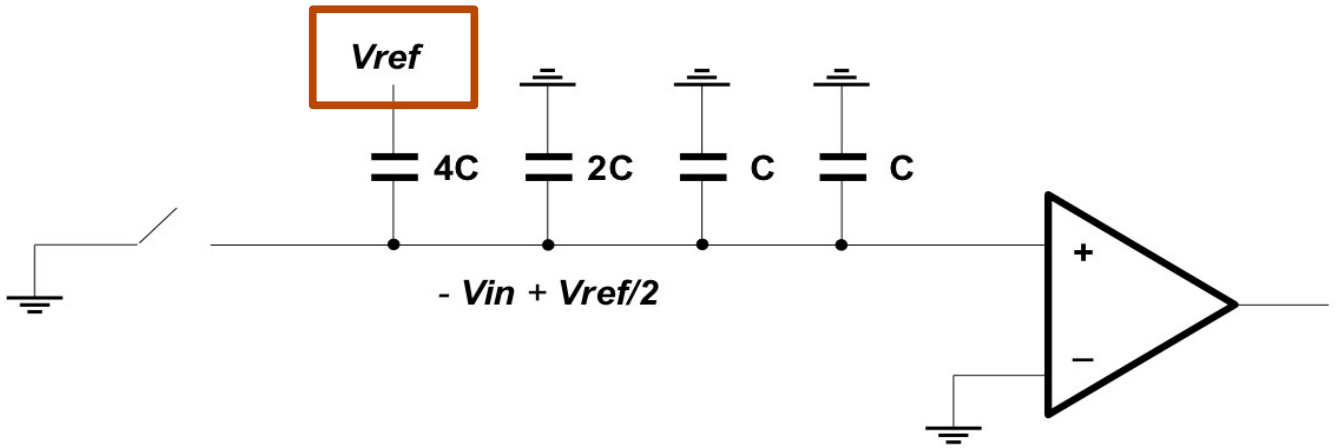




'dummy' capacitor *(tied to ground for the whole conversion)*

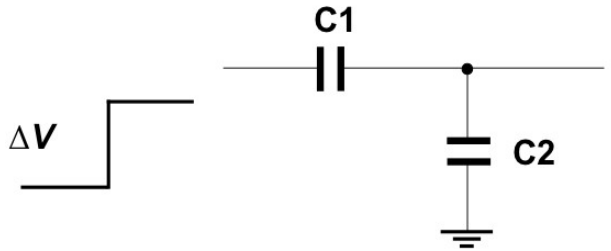
$$C_{TOT} = \underbrace{C + 2^0 C}_{2\times 2^0 C} + 2^1 C + 2^2 C + \ldots + 2^{N-1} C = 2 \times 2^{N-1} C = 2^N C$$

Charge redistribution



- Vin + Vref/2 > 0 ?

Refresh



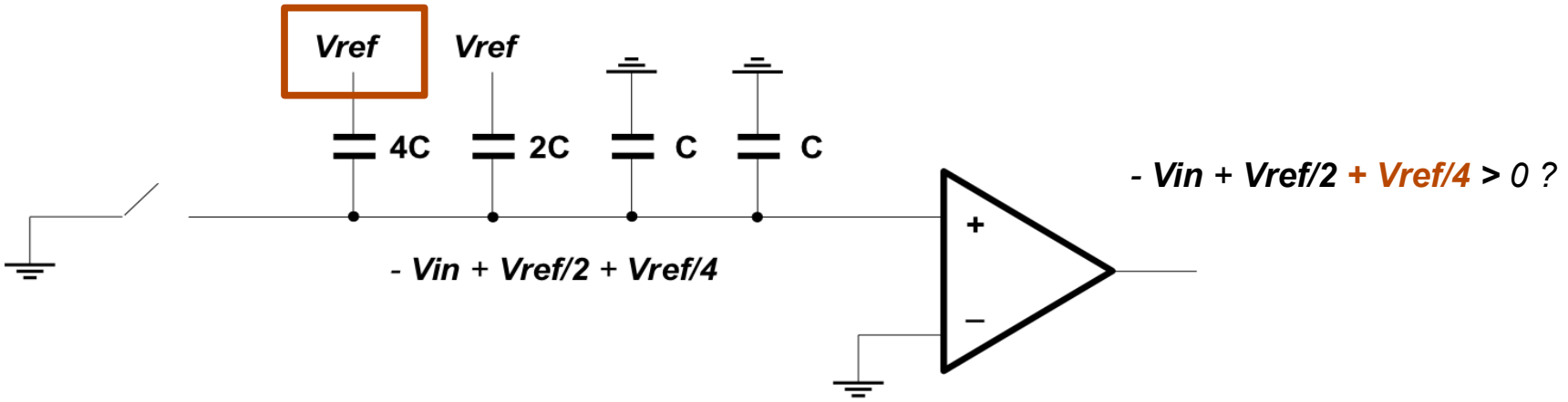
$$V' = V_0 + \frac{C1}{C1 + C2}\Delta V$$

capacitive divider formula

Conversion phase (trial-and-error)

Vin > Vref/2

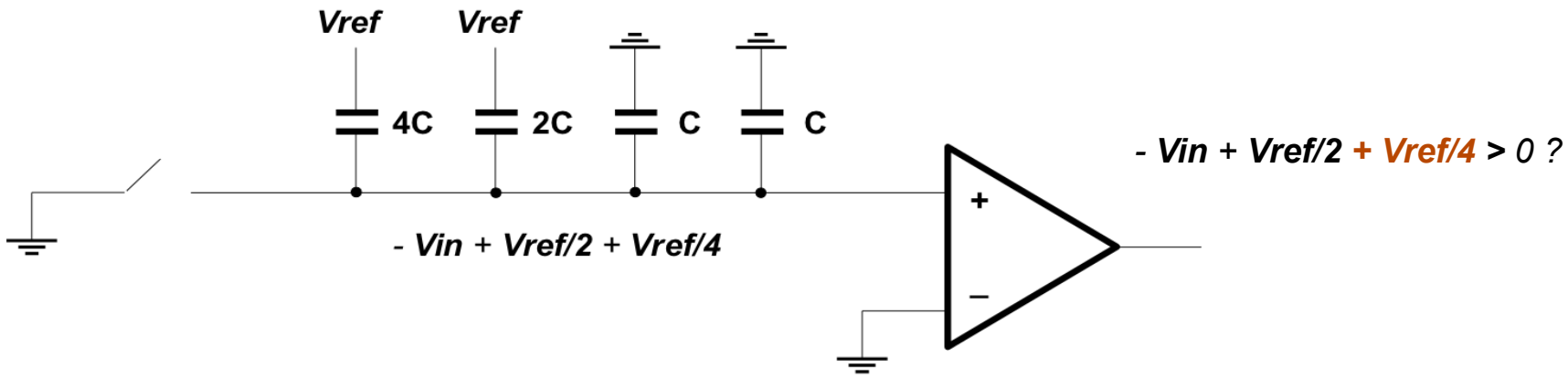
B = 1



Conversion phase (trial-and-error)

Vin > Vref/2

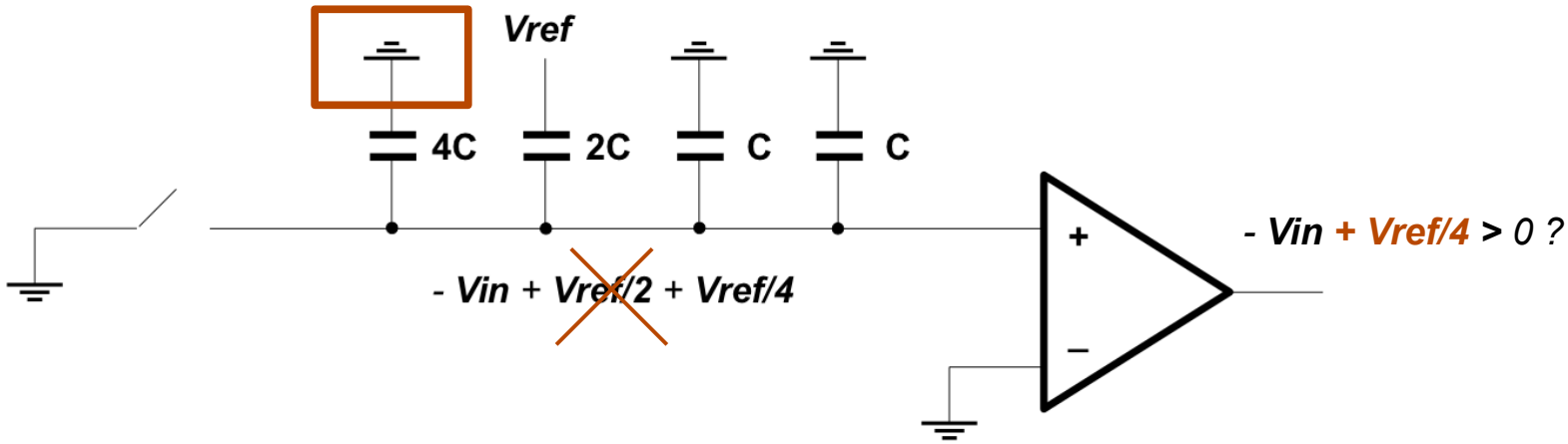
B = 1



- Vin + Vref/2 + Vref/4 > 0 ?

Vin < Vref/2

B = 0

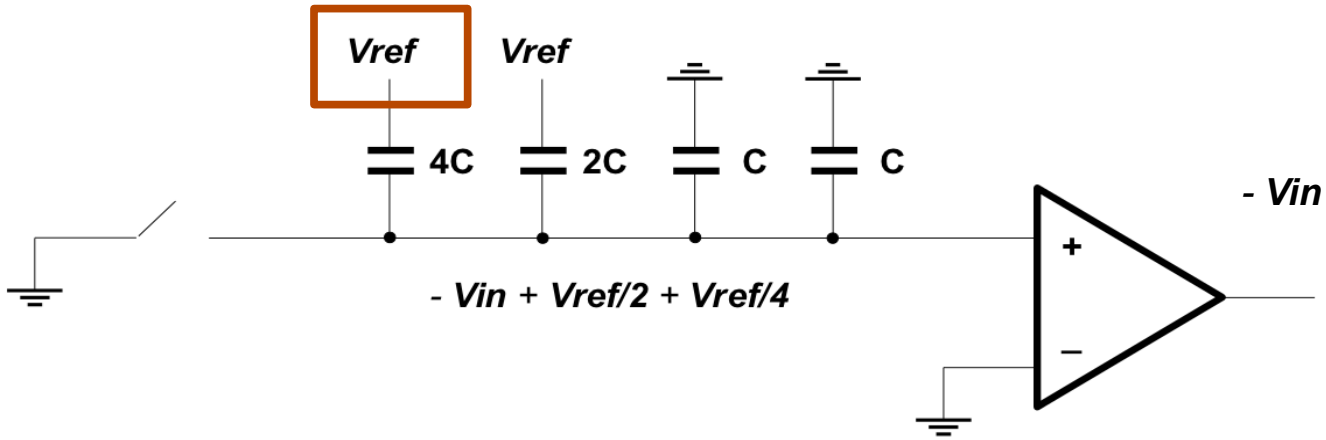


- Vin + Vref/4 > 0 ?

Conversion phase (trial-and-error)

Vin > Vref/2

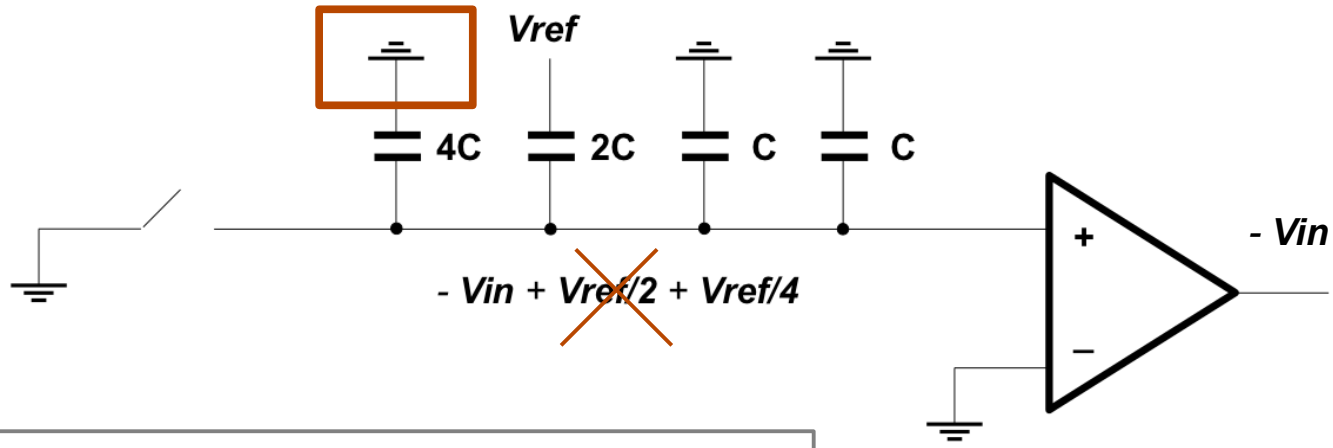
B = 1



- Vin + Vref/2 + Vref/4 > 0 ?

Vin < Vref/2

B = 0

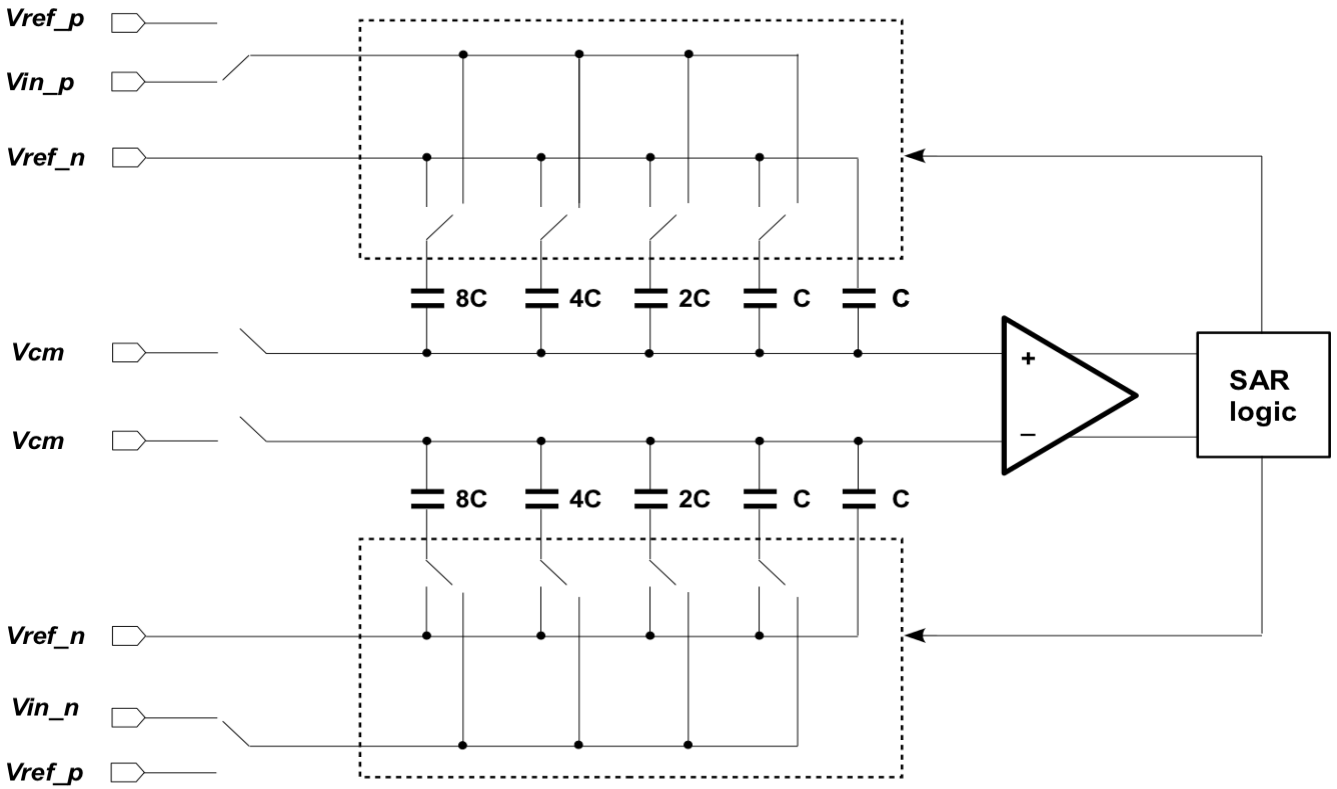


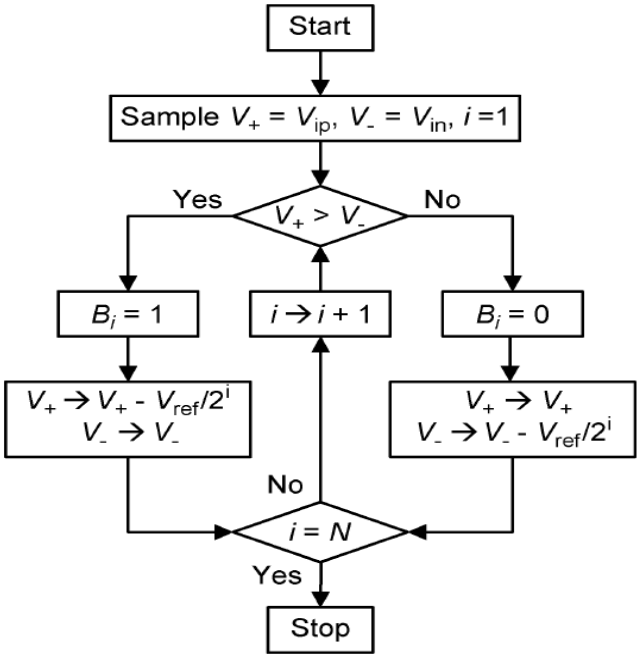
- Vin + Vref/4 > 0 ?

$$V_{DAC}(k) = -V_{in} + \underbrace{\left[\frac{1}{C_{TOT}} \sum_{i=N-1}^{N-k} B_i C_i\right]}_{B_{N-1}\frac{1}{2}+B_{N-2}\frac{1}{4}+\ldots} V_{ref}$$

the procedure is then repeated until the LSB is decided

Fully-differential architectures

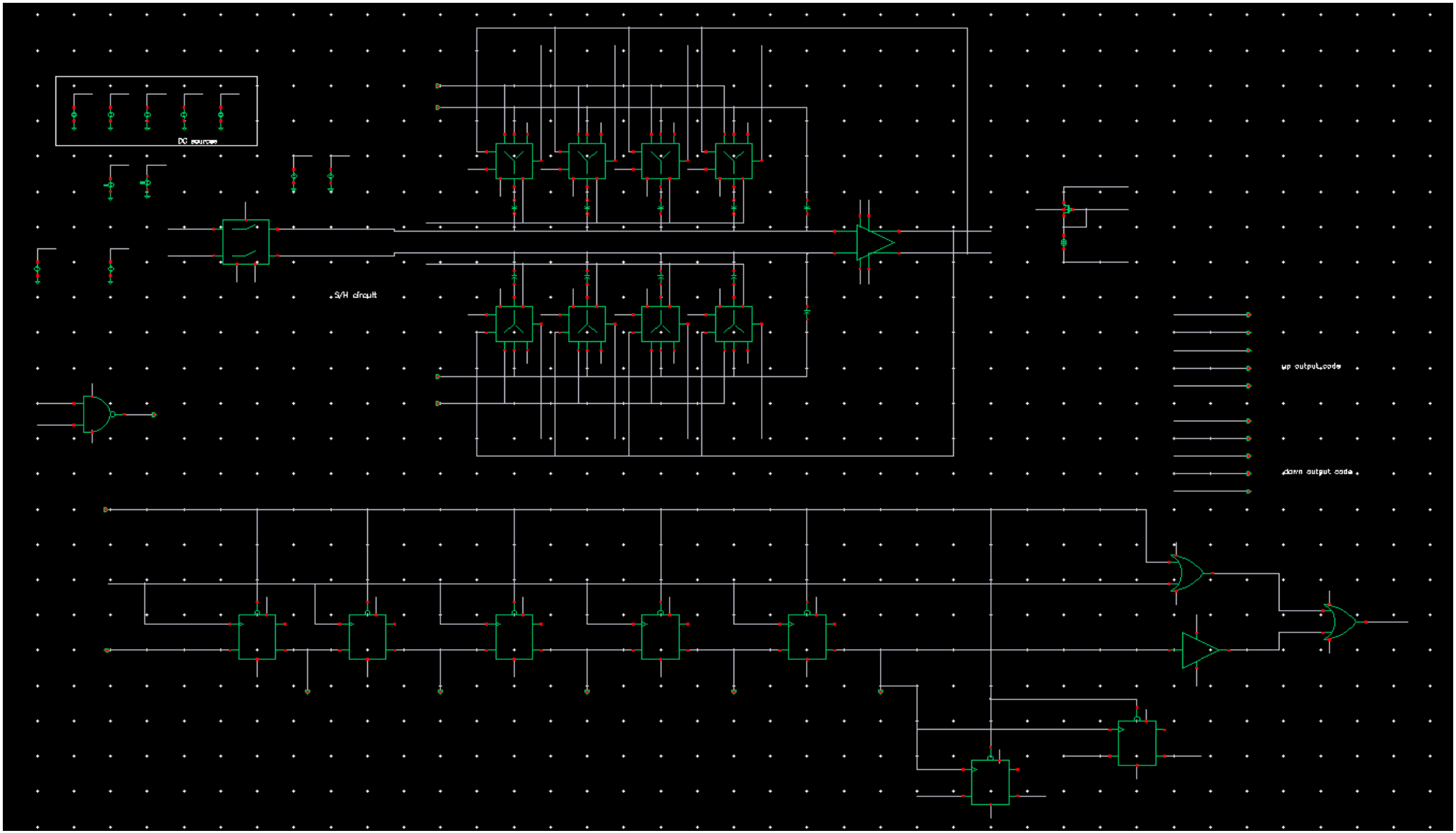




- two different reference voltages *Vref_p* and *Vref_n* (in principle *Vref_n* can be different from ground)
- high ***common-mode noise rejection power*** (substrate and voltage supply noise)
- the operations of the two sides is ***complementary***
- in a conventional architecture ***both input signals and reference voltages are sampled on the bottom plates***
- several energy-efficient switching schemes have been proposed (see later...)

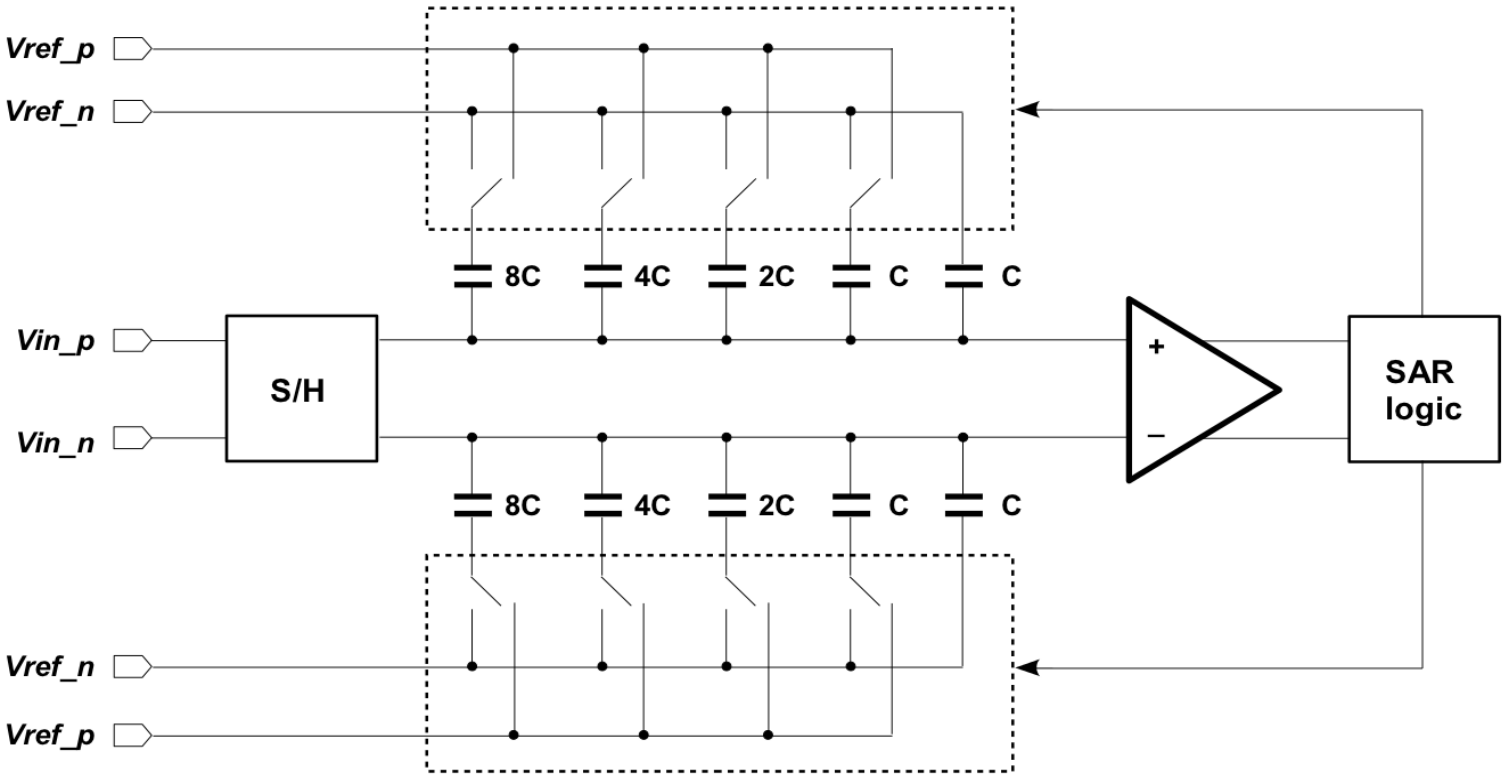
Full design example

Liu et al, *A 10-bit 50-MS/s SAR ADC with a Monotonic Capacitor Switching Procedure*
IEEE Journal of Solid State Circuits, Vol. 45, N. 4, Apr. 2010



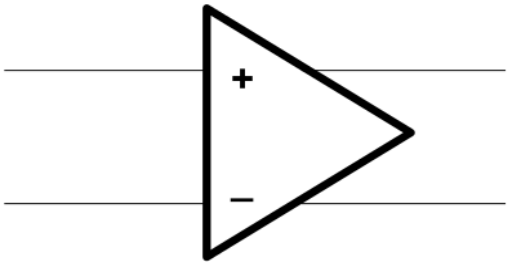
Full design example

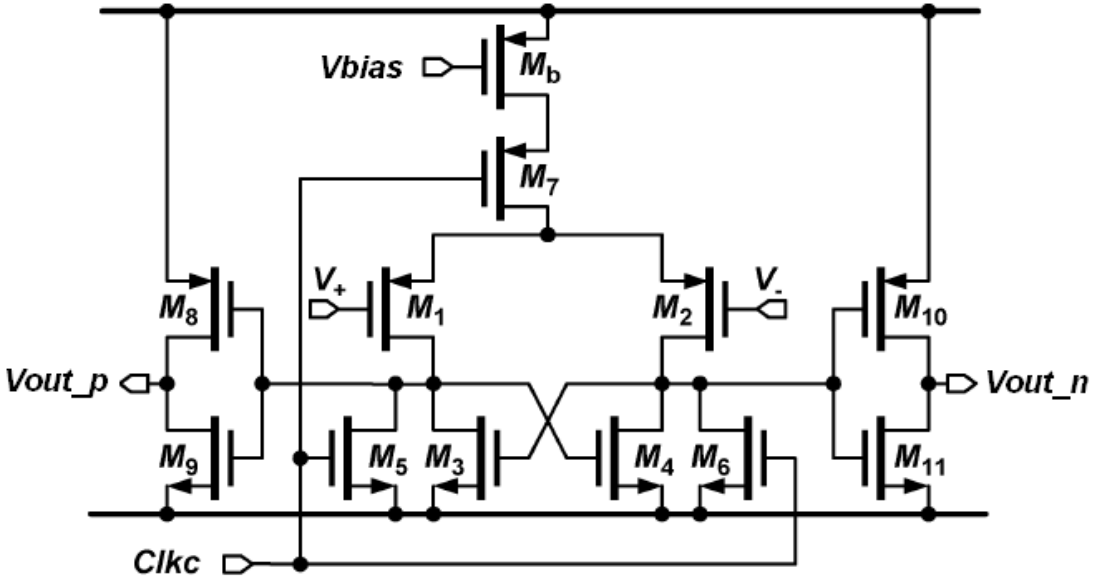
Liu et al, *A 10-bit 50-MS/s SAR ADC with a Monotonic Capacitor Switching Procedure*
IEEE Journal of Solid State Circuits, Vol. 45, N. 4, Apr. 2010



- 5 bit ADC design in 65nm CMOS (10 bit, 130 nm CMOS in the reference paper)
- 50 MS/s sampling rate
- ***set-and-down (monotonic) switching scheme***, slight different with respect to the conventional one
- ***asynchronous control logic***

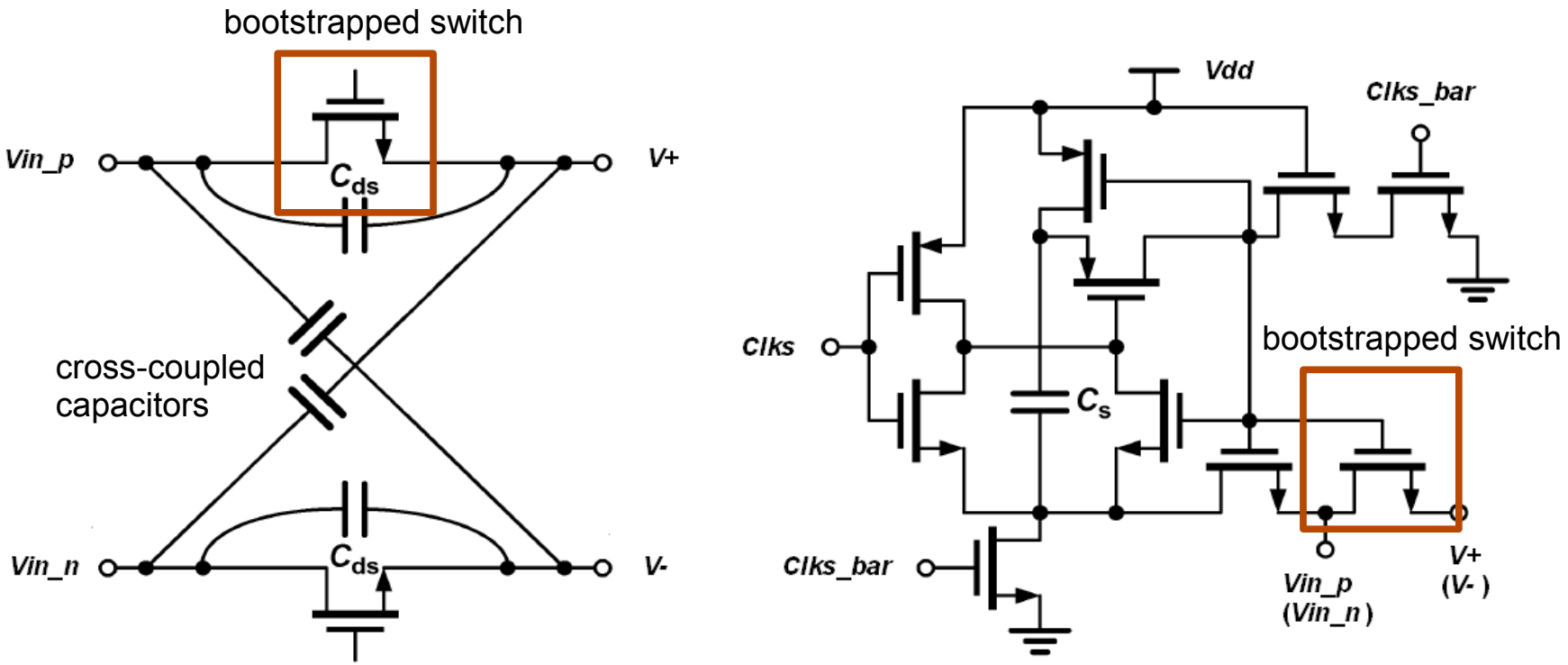
CMOS comparator





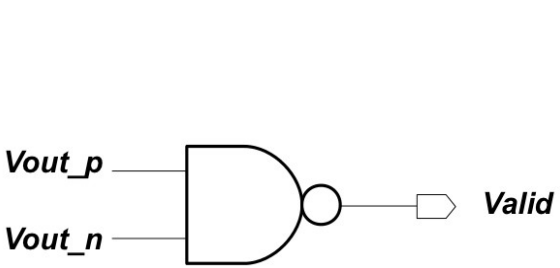
- ***synchronous*** voltage comparator
  - ***no static power consumption*** (no static current flows in the main branch)
  - comparator outputs are reset to high (*Vdd*) when *Clkc* is high
  - when *Clkc* goes low the differential pair compares analog inputs
- speed increases with the usage of ***regenerative (positive) feedback***
  - the ***latch*** regeneration forces one output to high and the other one to low according to the comparison result
- the input common-mode voltage gradually decreases from half *Vref_p* to ground (*Vref_n*)
  - ***PMOS input pair***

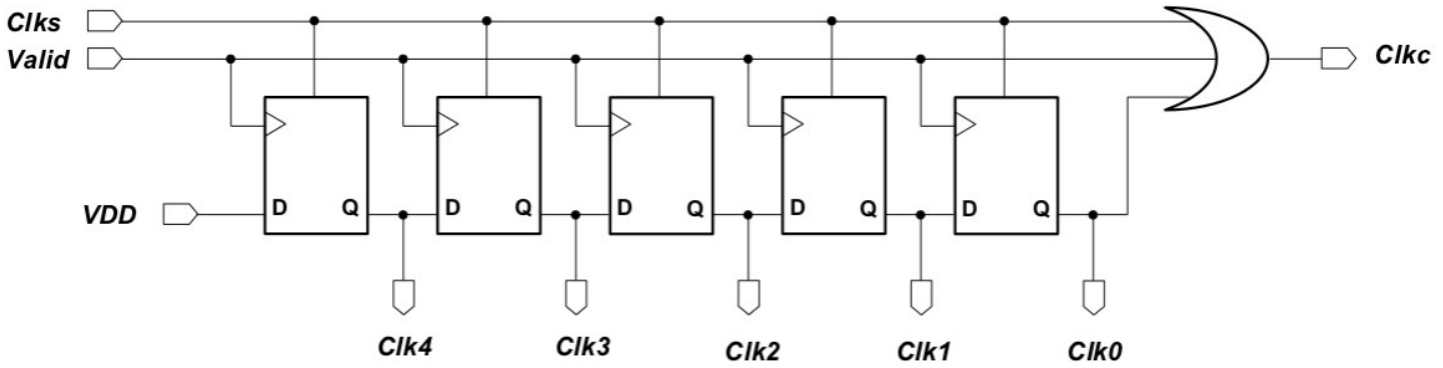
S/H circuit



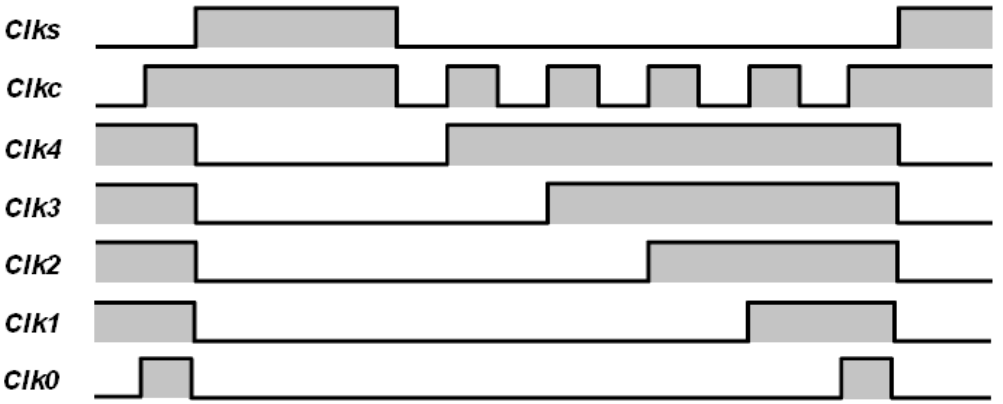
- ***bootstrapped switches*** are used to sample the analog input voltages
- Vgs of the sampling transistor is fixed to the supply voltage, improving the switch linearity
- cross-coupled capacitors are used to neutralize ***coupling effects*** between Cds and DAC sampling capacitors, which degrades the high frequency performance
- the ***sampling phase*** (*Clks* = 1) is about 20% of the whole conversion time

SAR control logic



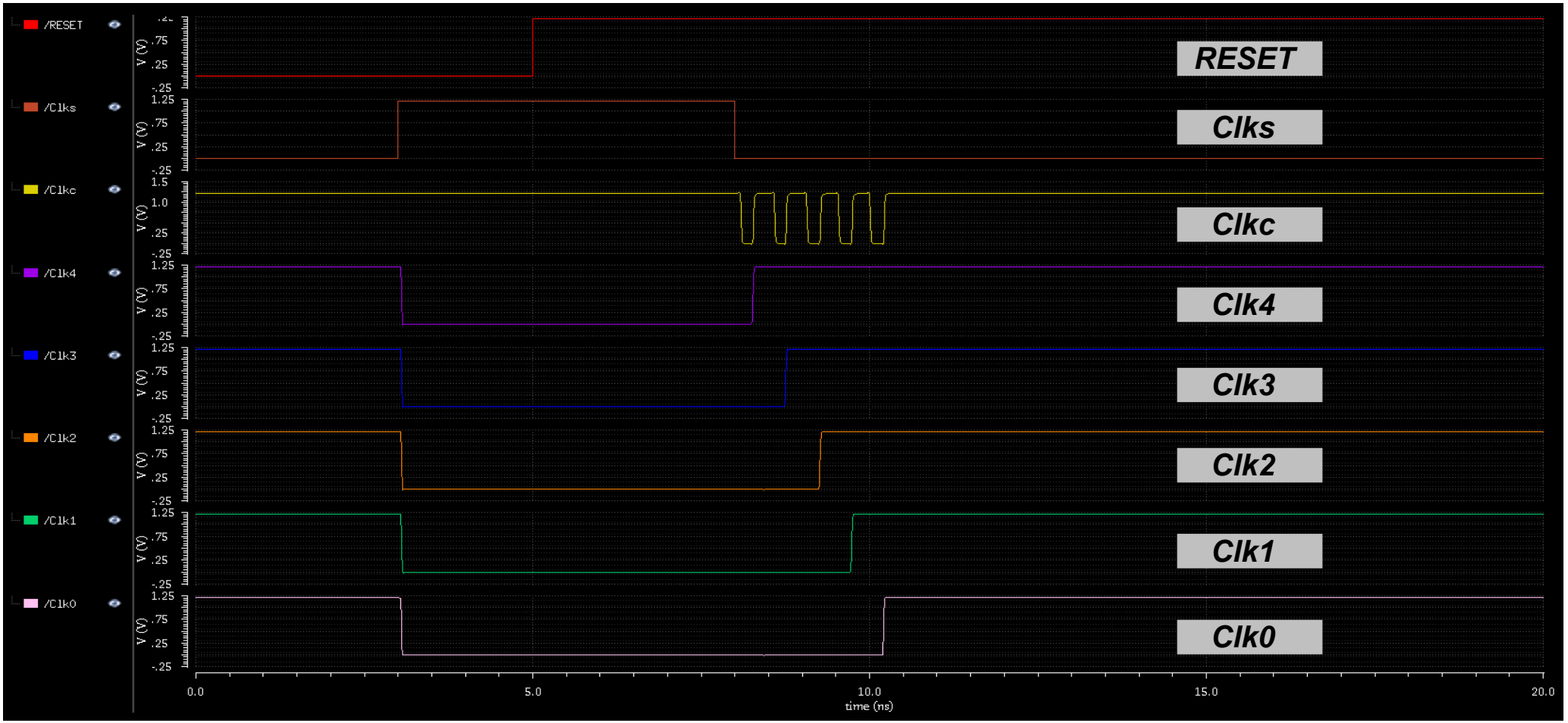


- ***asynchronous control logic***
  - control signals are ***internally generated***
  - no external clock is required

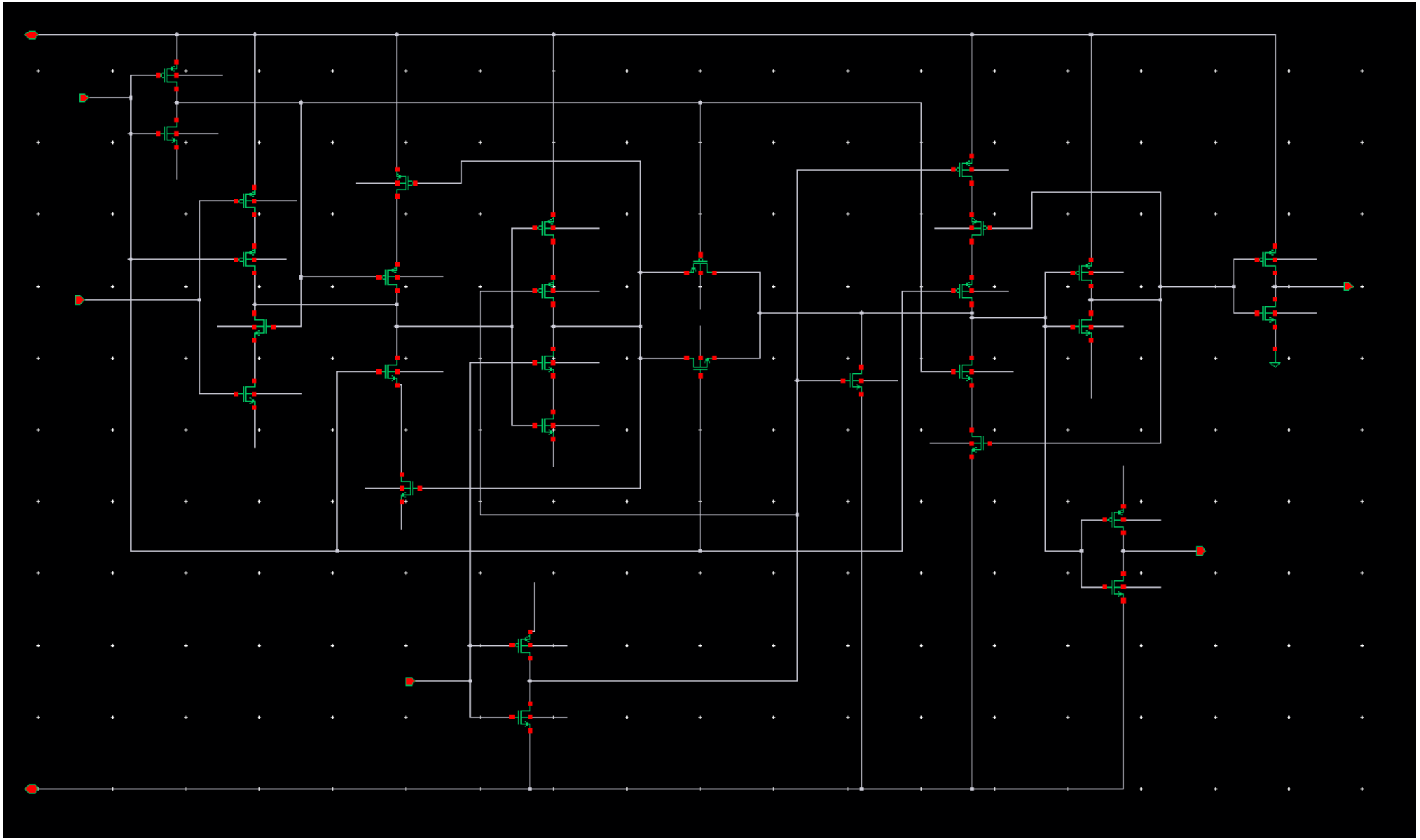


- *Clks* is the control signal of the sampling switches
- comparator outputs generate a *Valid* signal
- *Clkc* is the control signal of the dynamic comparator
  - when *Clkc* = 1 *Vout_p* and *Vout_n* are reset to 1 and *Valid* = 0
  - when *Clkc* = 0 *Vout_p* and *Vout_n* are complementary and *Valid* is pulled to high, enabling DFF shifts

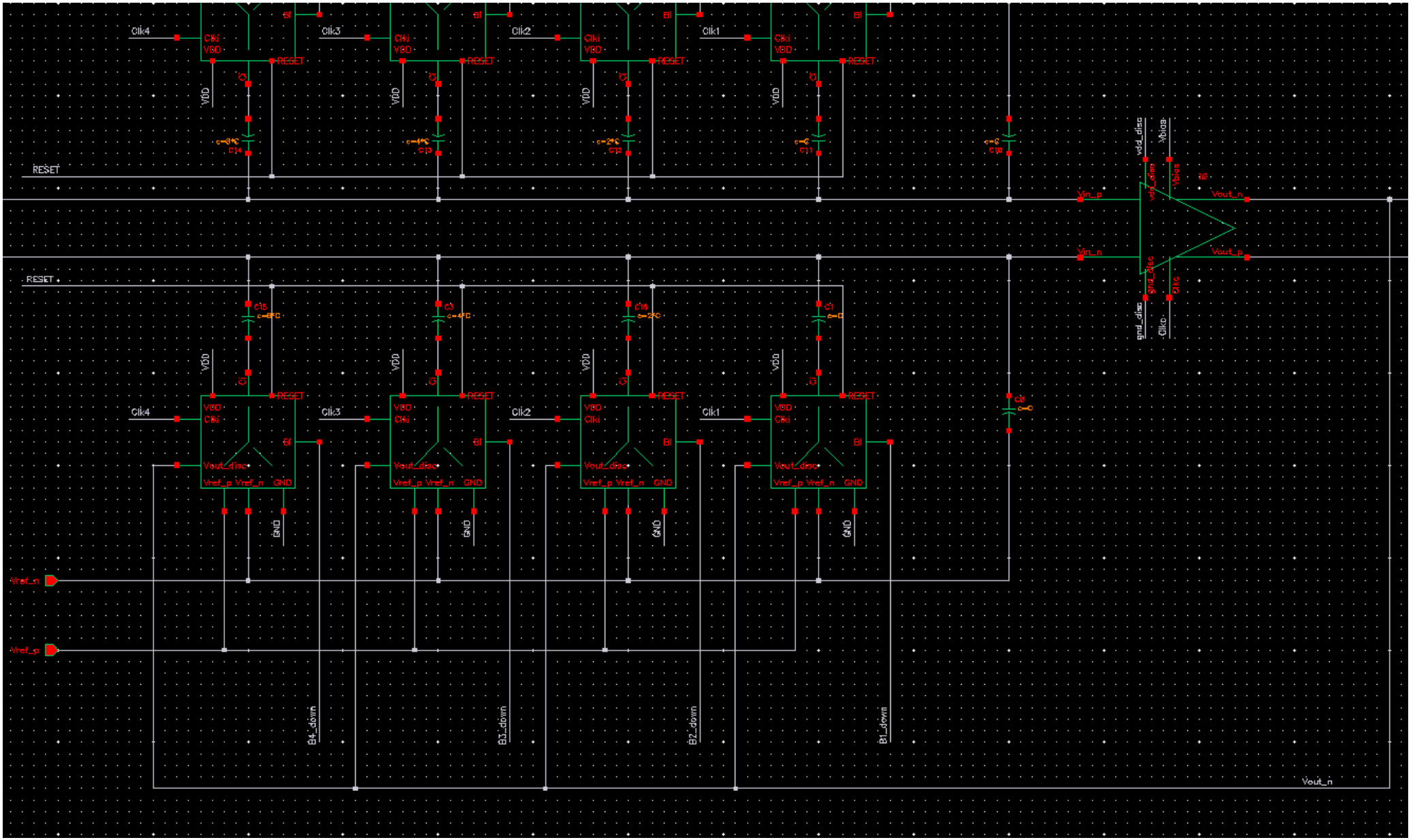
Transient analysis



CMOS DFF

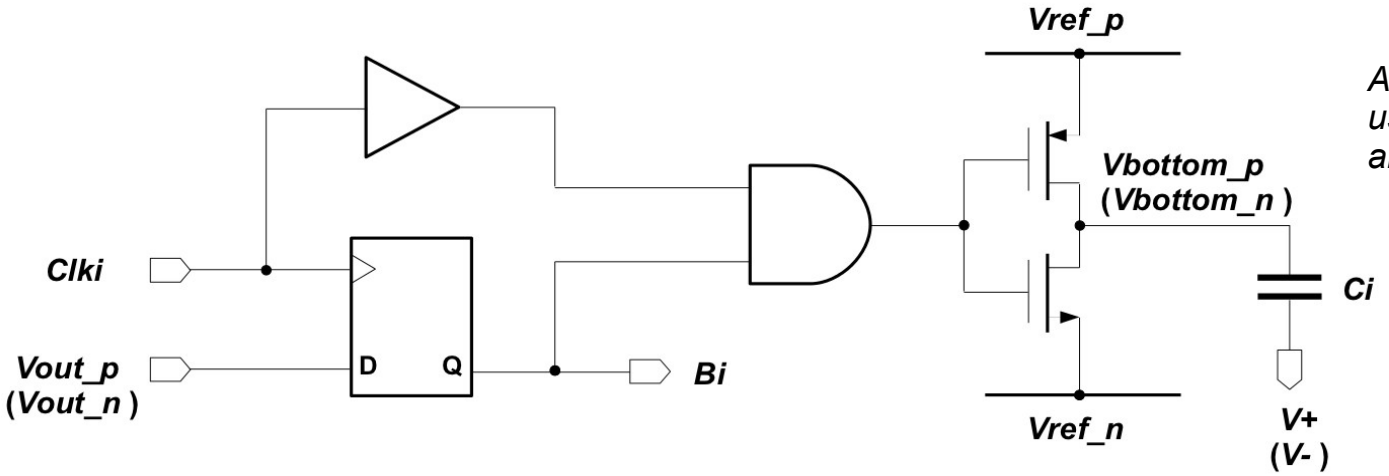


DAC control logic

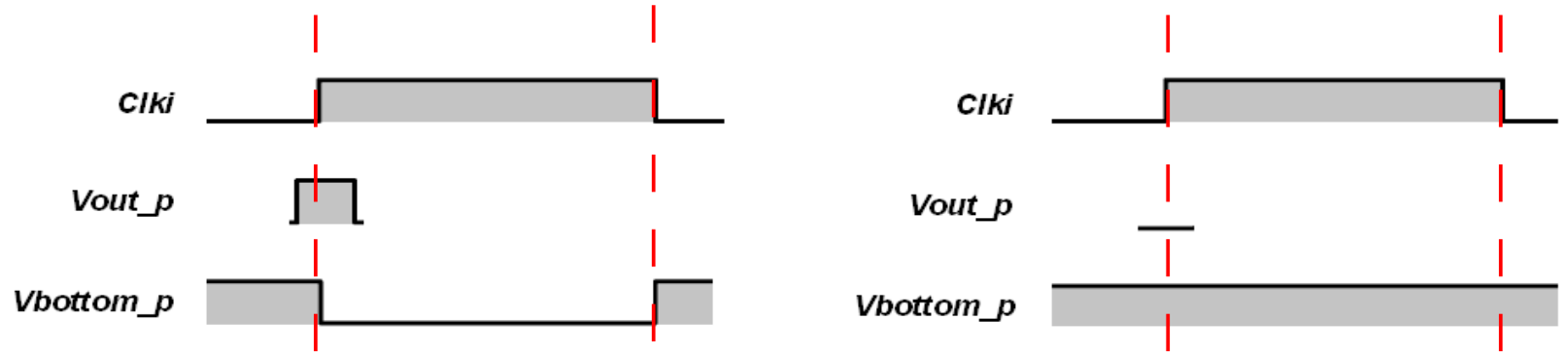


DAC control logic

Clk4 to Clk0 signals sample comparator output codes and control the switching activity of DAC capacitors

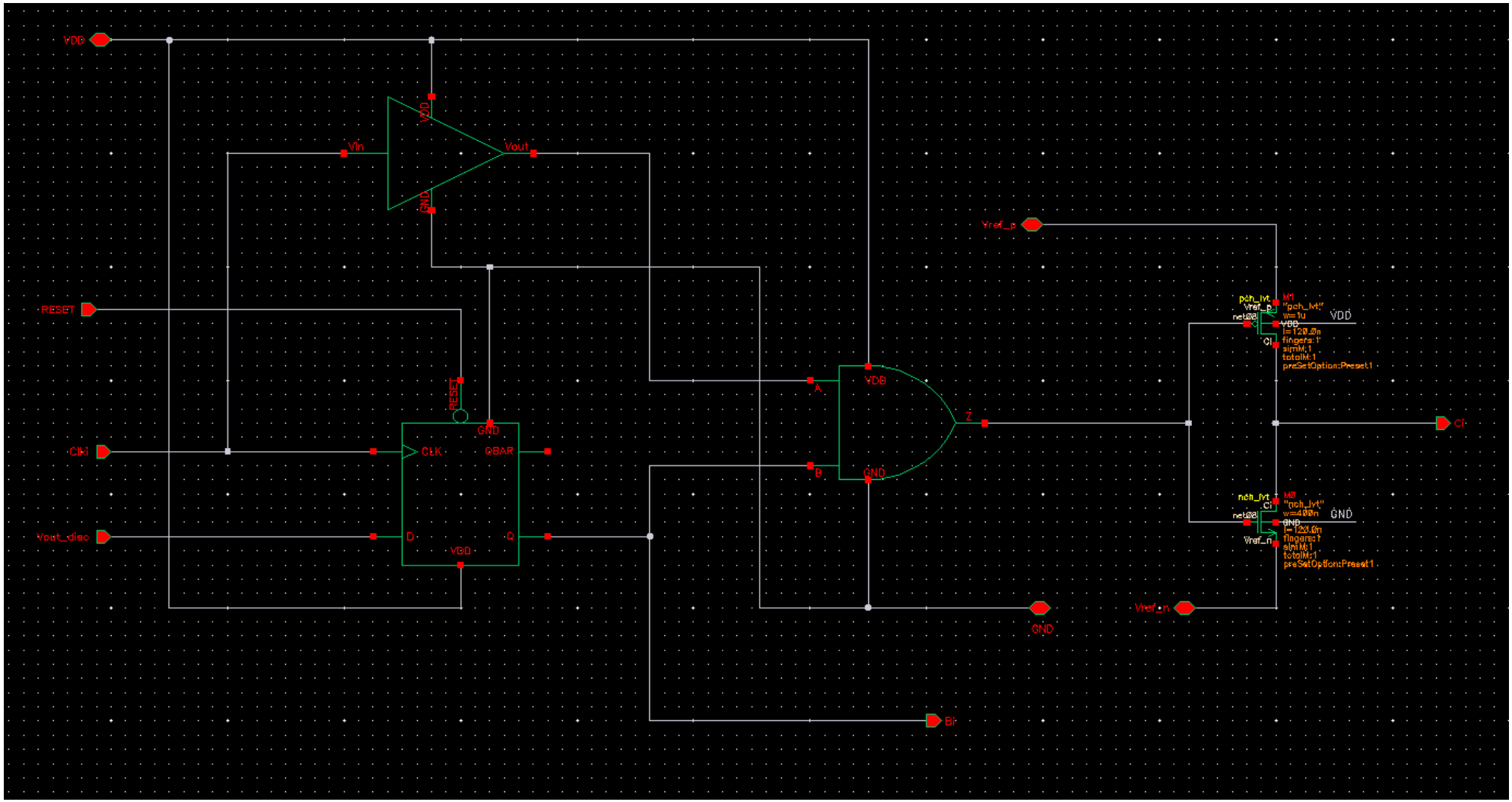


*A simple **CMOS inverter** is used to switch between positive and negative references*



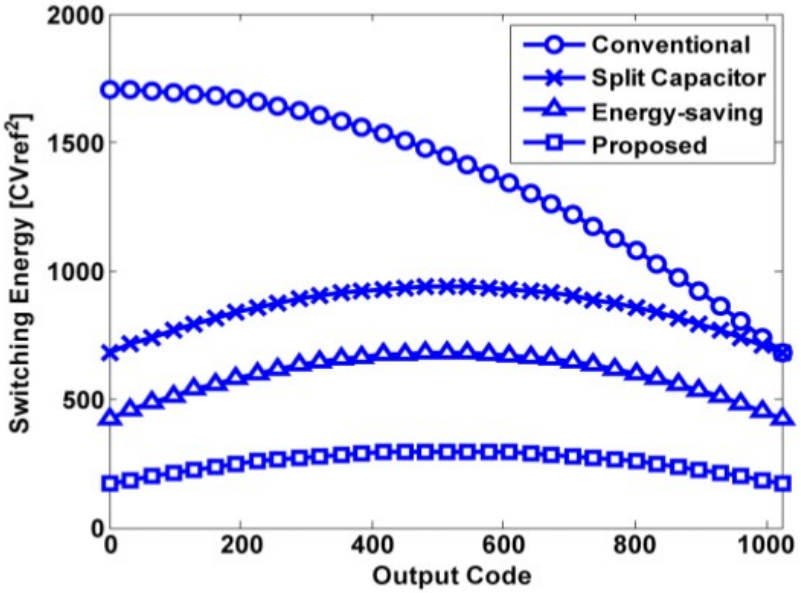
- at the rising edge of *Clki* a DFF samples the comparator output level
  - if *Vout_p* = 1 the bottom plate of the relative capacitor is switched from *Vref_p* to ground (*Vref_n*)
  - if *Vout_p* = 0 the bottom plate of the relative capacitor is kept to *Vref_p*
- at the falling edge of *Clki* all capacitor bottom plates are reconnected to *Vref_p*

DAC control logic



Set-and-down switching method

- several different ***switching procedures*** (SAR logic) have been proposed to ***reduce the switching power dissipation*** of the DAC capacitor array
  - conventional switching scheme
  - split-capacitor method
  - energy-saving method
  - ***set-and-down (monotonic)***



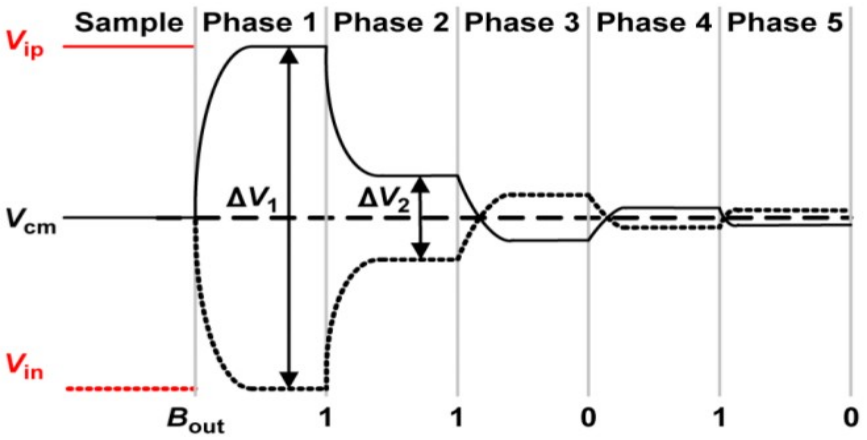
- ~80% energy reduction than a conventional switching scheme
  - the average switching energy is reduced !

- ***sampling phase***
  - top plates of capacitors sample the analog input signals *Vin_p* and *Vin_n*
  - bottom plates of capacitors are switched to *Vref_p*

- ***redistribution/conversion phase***
  - the comparator directly performs ***the first comparison without switching any capacitor !***
  - ***only one relevant capacitor is turned down to Vref_n each time !***
  - charge transfers and in the DAC network and transitions of the control logic are reduced
  - no energy is consumed before the first comparison (reduced power consumption)

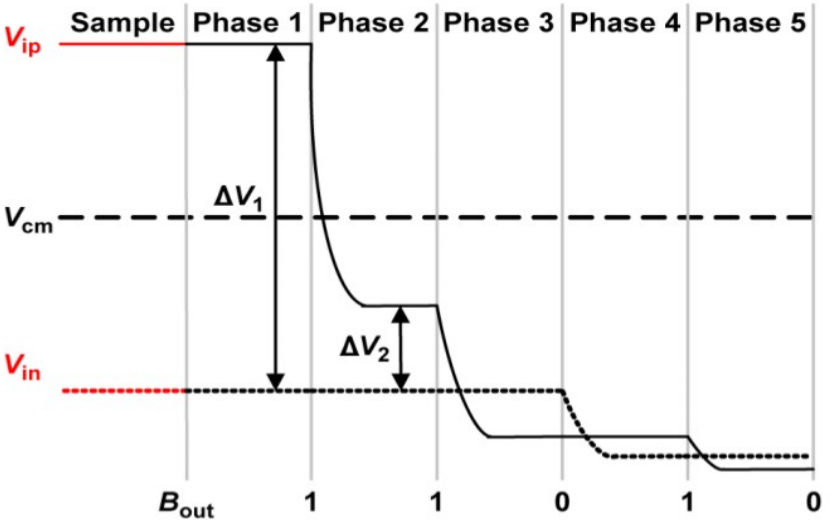
Conventional vs monotonic

conventional switching method



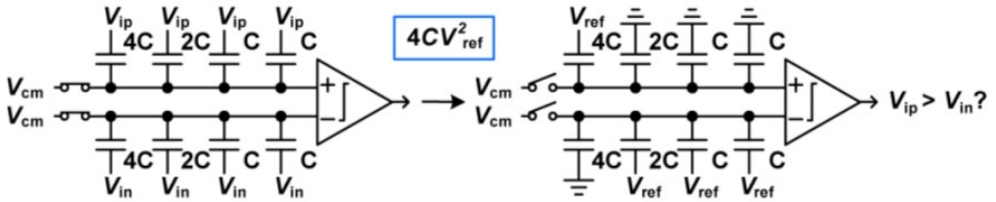
*the input common-mode voltage is **fixed***

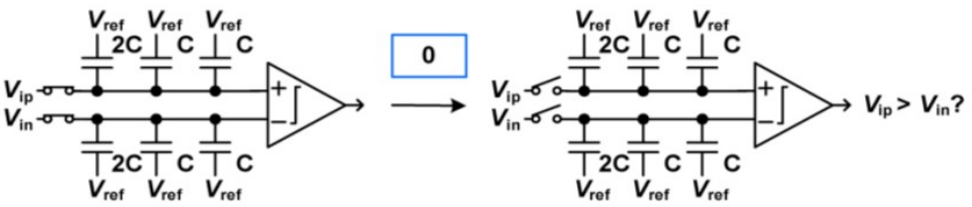
set-and-down (monotonic) switching method



*the input common-mode voltage gradually **decreases** from half Vref to ground*

3bit example

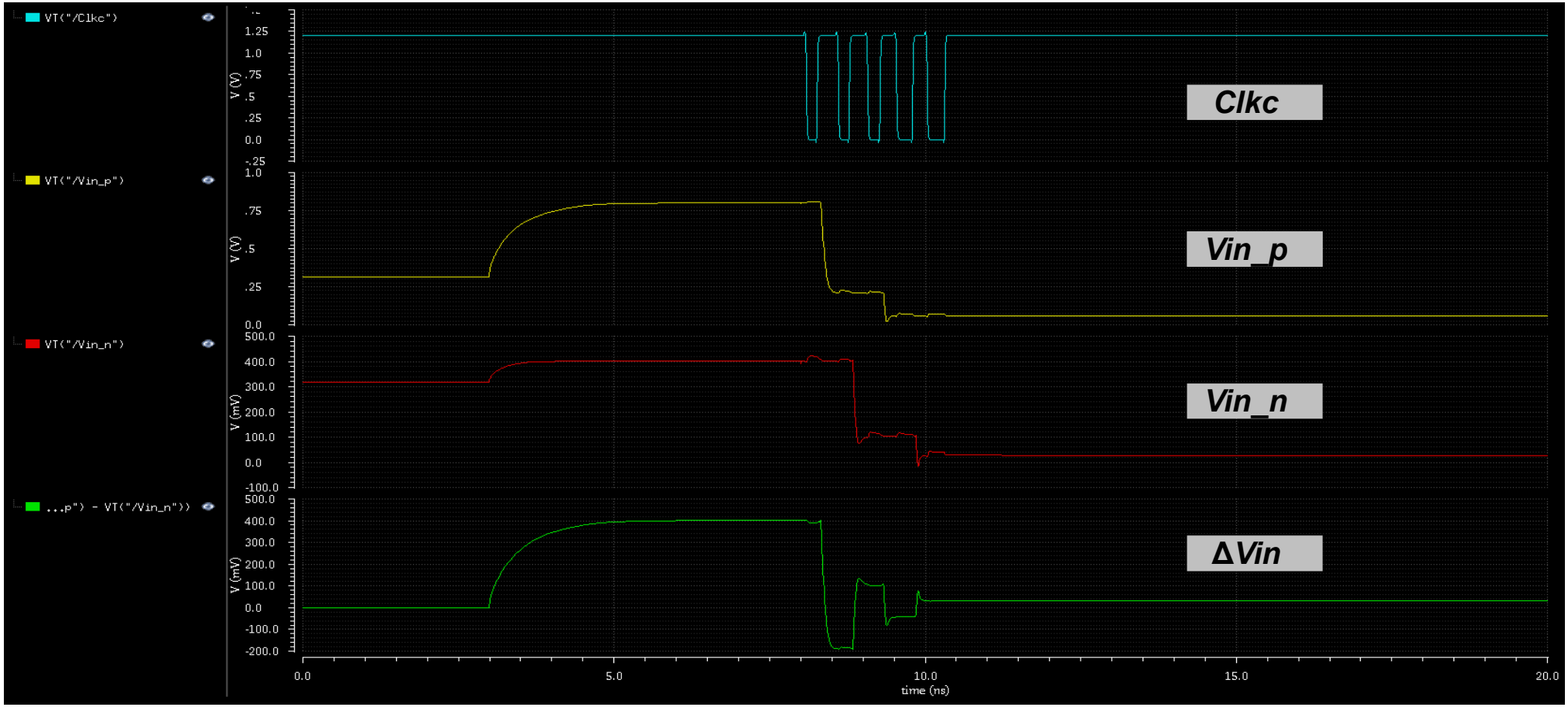




no energy is consumed before the first comparison !
*(the comparator directly performs the first comparison **without switching any capacitor**)*

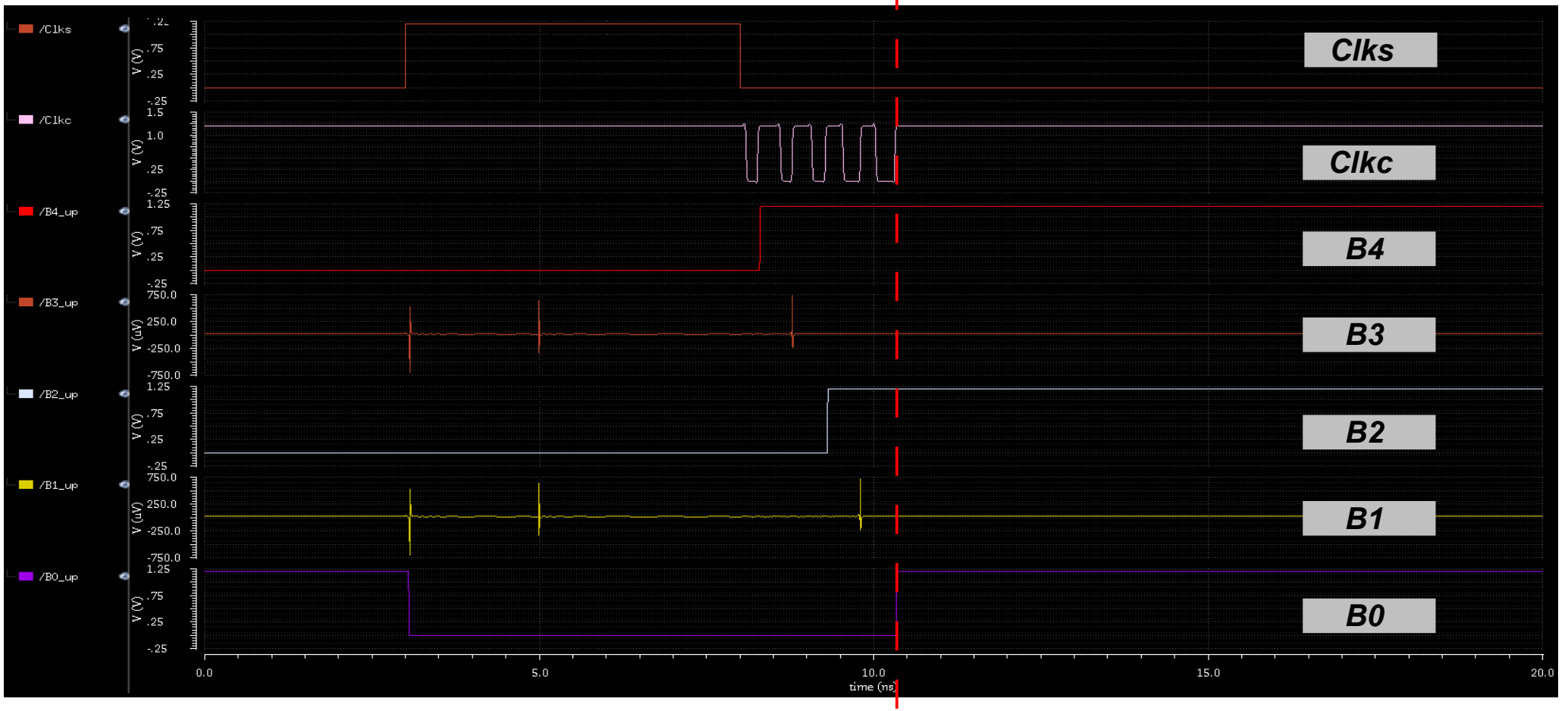
Conversion example (1)

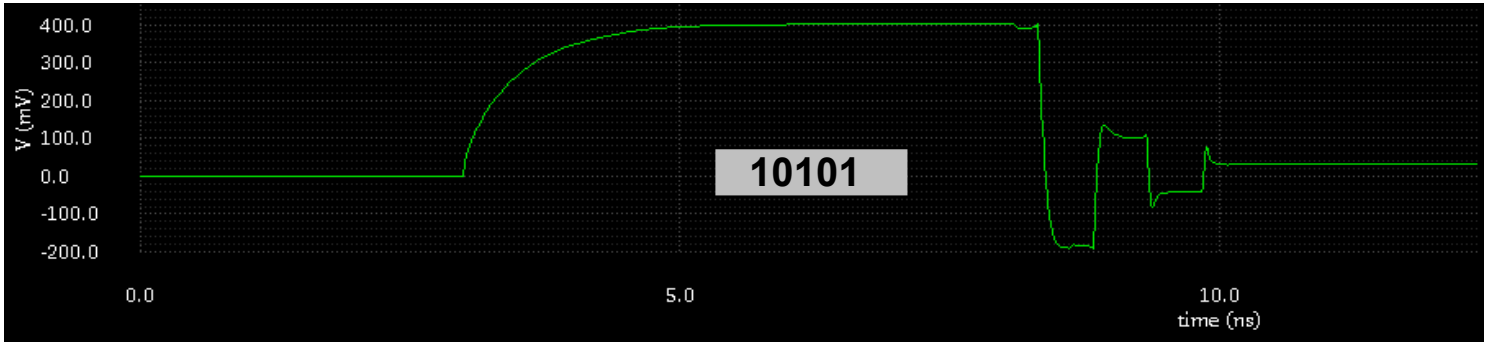
ΔVin = + 400mV



*less than **3ns** are required to complete a 5 bit conversion !*

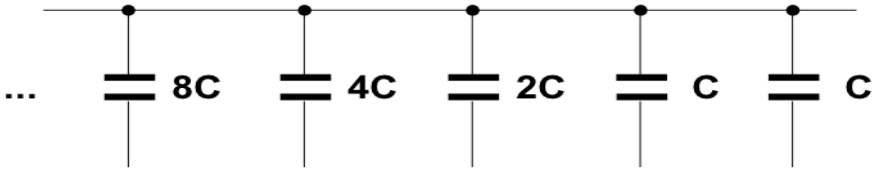
Conversion example (2)





Binary-weighted DAC array

- capacitance values are obtained in form of ***sums of unit capacitors***
  - 2C = C + C
  - 4C = C + C + C + C
  - etc.

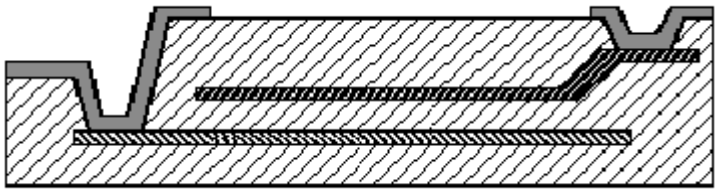


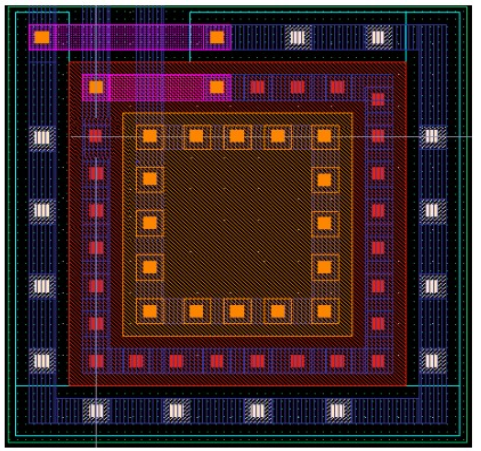
- repeated structure of ***unit capacitors***

- for an N-bit ADC the number of unit capacitors in the capacitor array is $2^{N-1} = 2^N/2$
  - ***half of a conventional architecture*** *!* (remind, 2^N in a conventional SAR ADCs)
  - the total capacitance is reduced by 50%

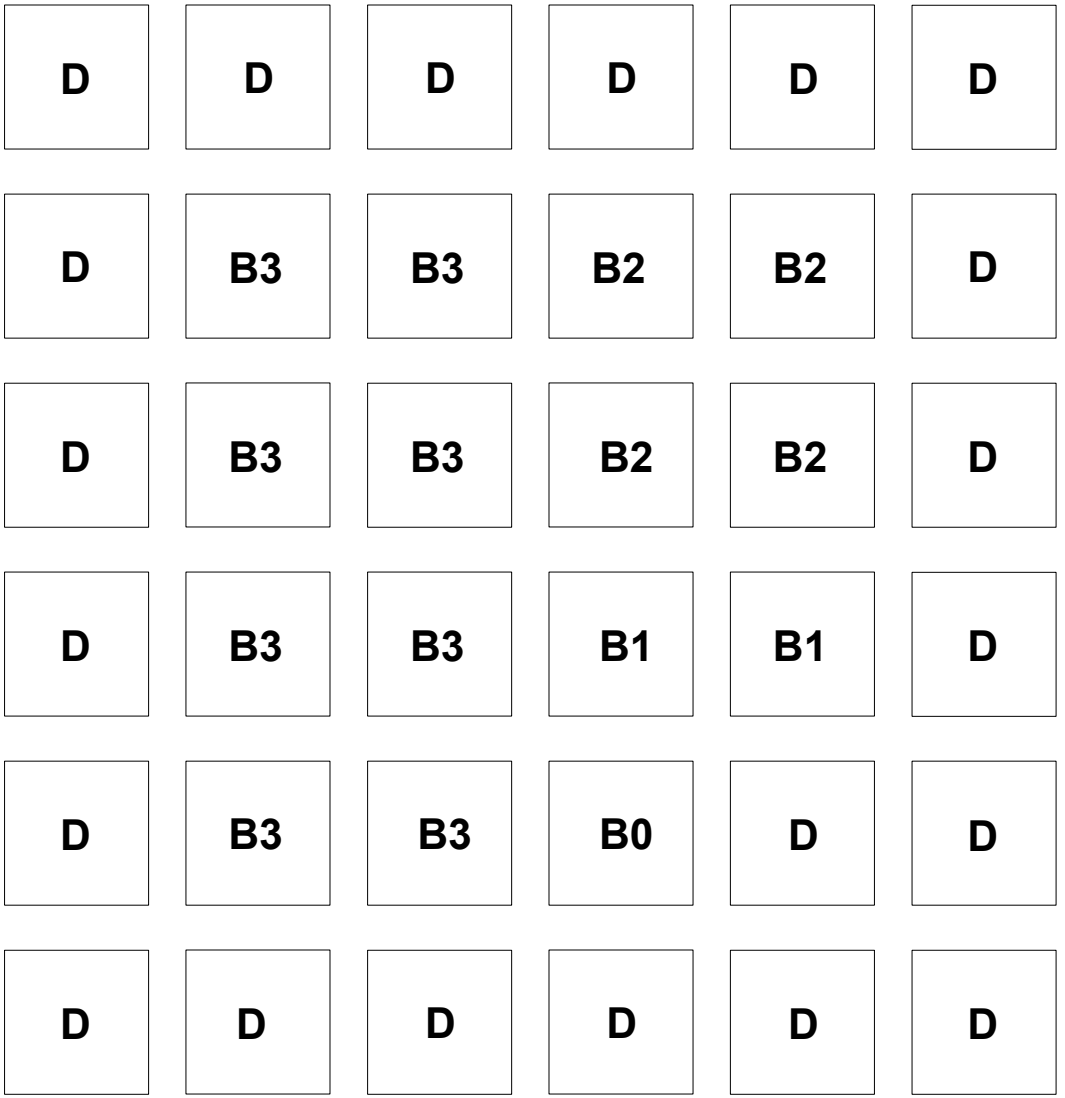
- the capacitors network occupies most of the total area (~70%)

- each unit capacitor can be implemented as a sandwich ***metal-oxide-metal (MOM) capacitor***

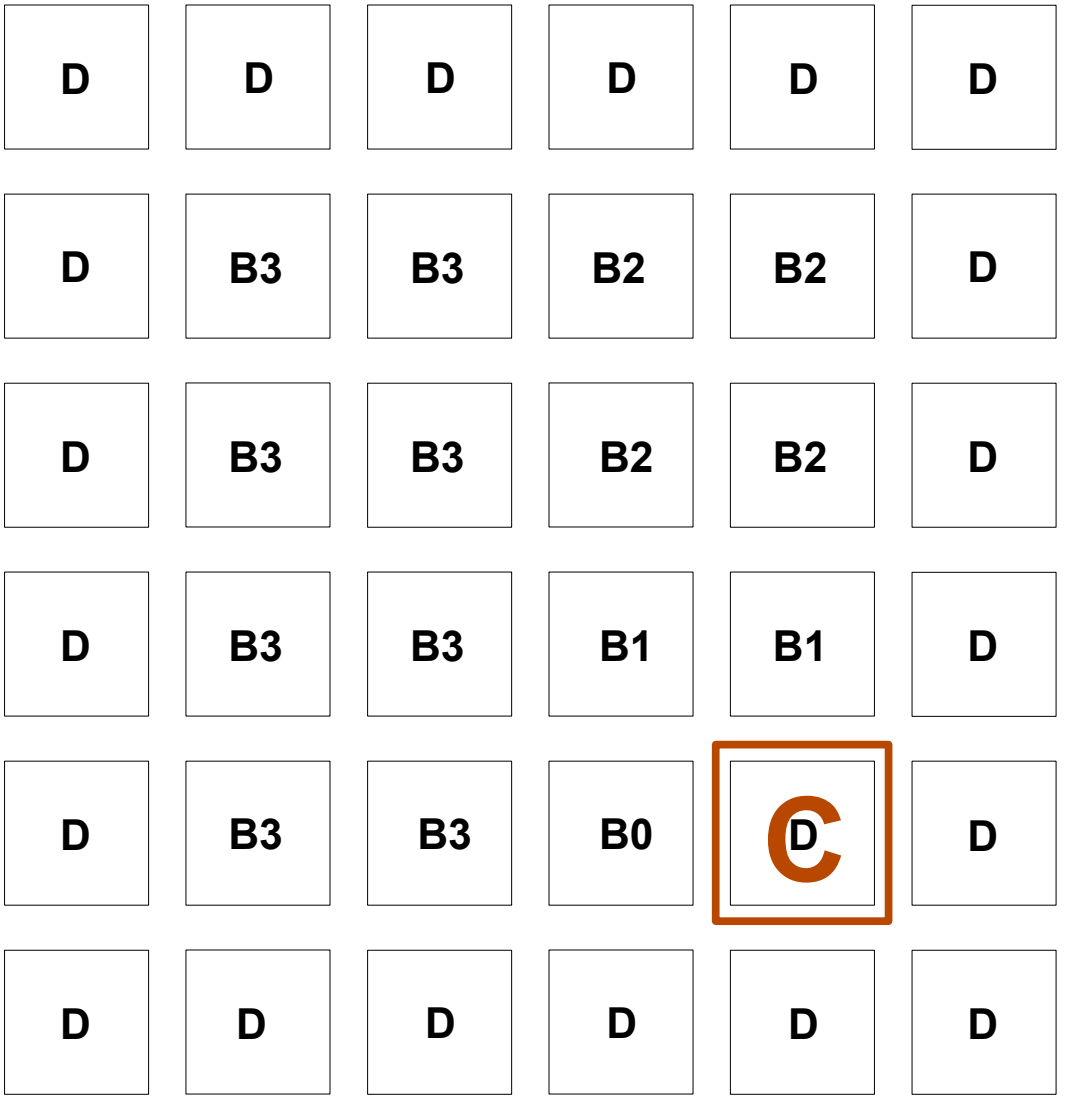




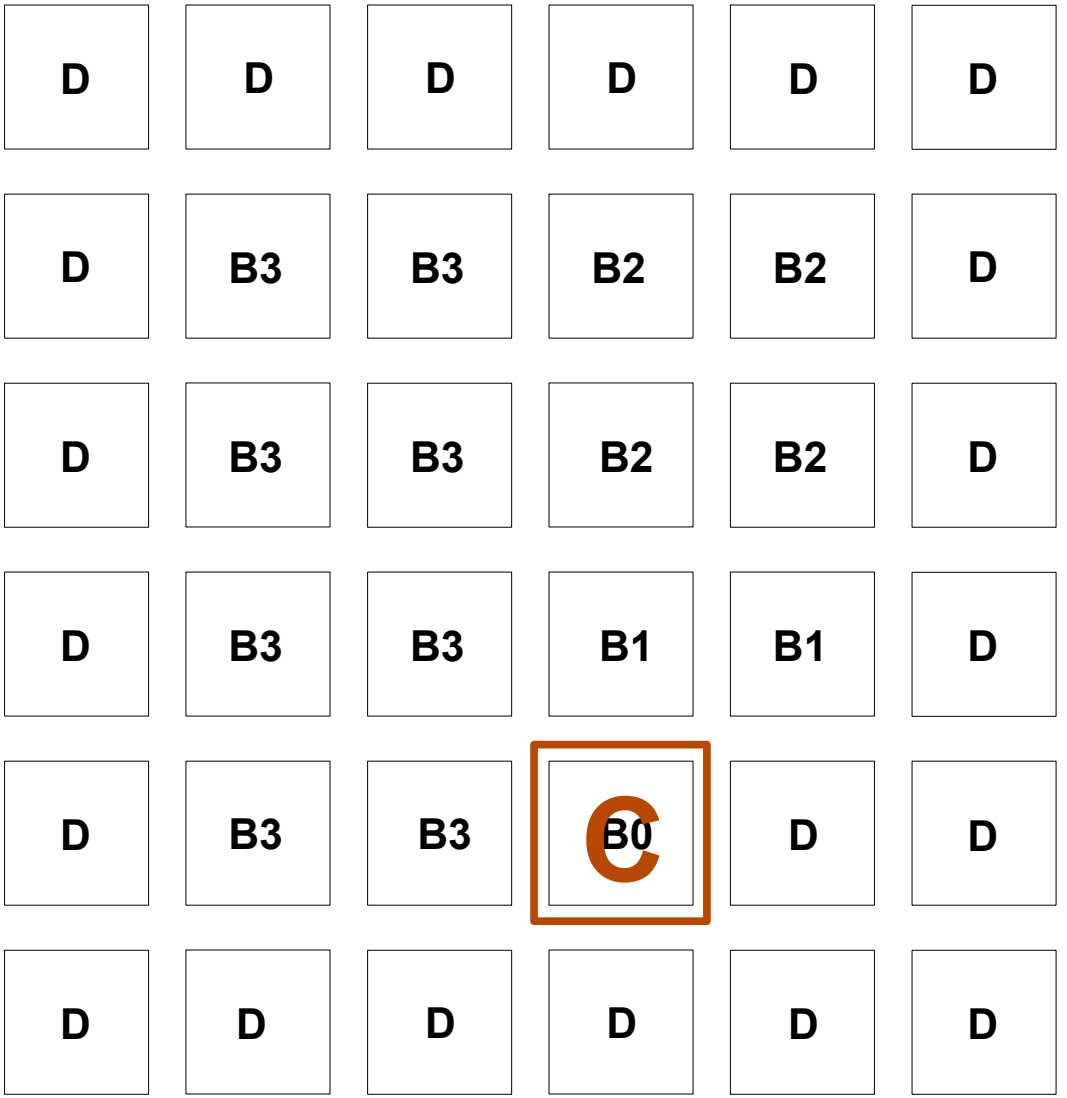
Layout floorplan (5 bit)



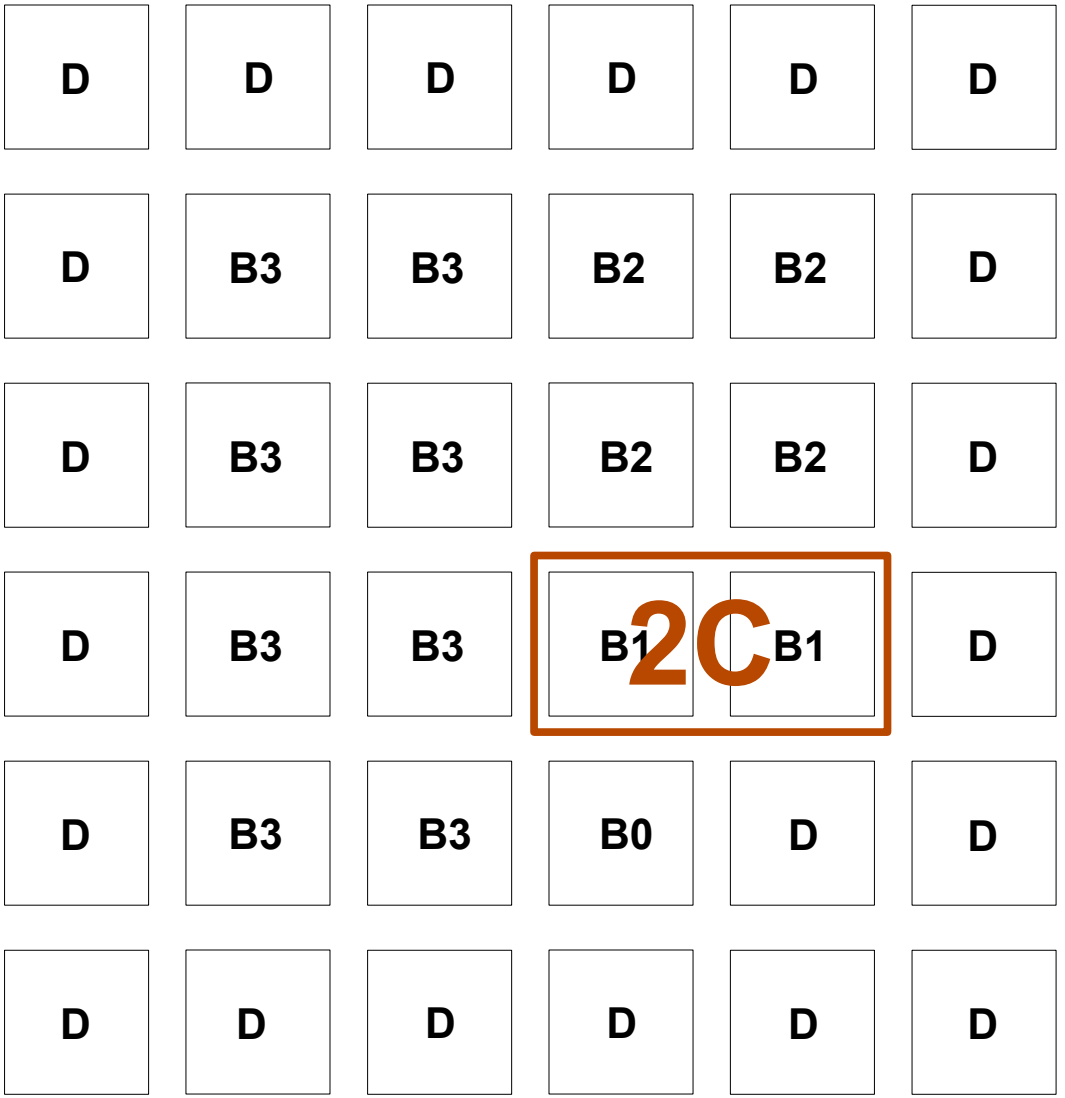
Layout floorplan (5 bit)



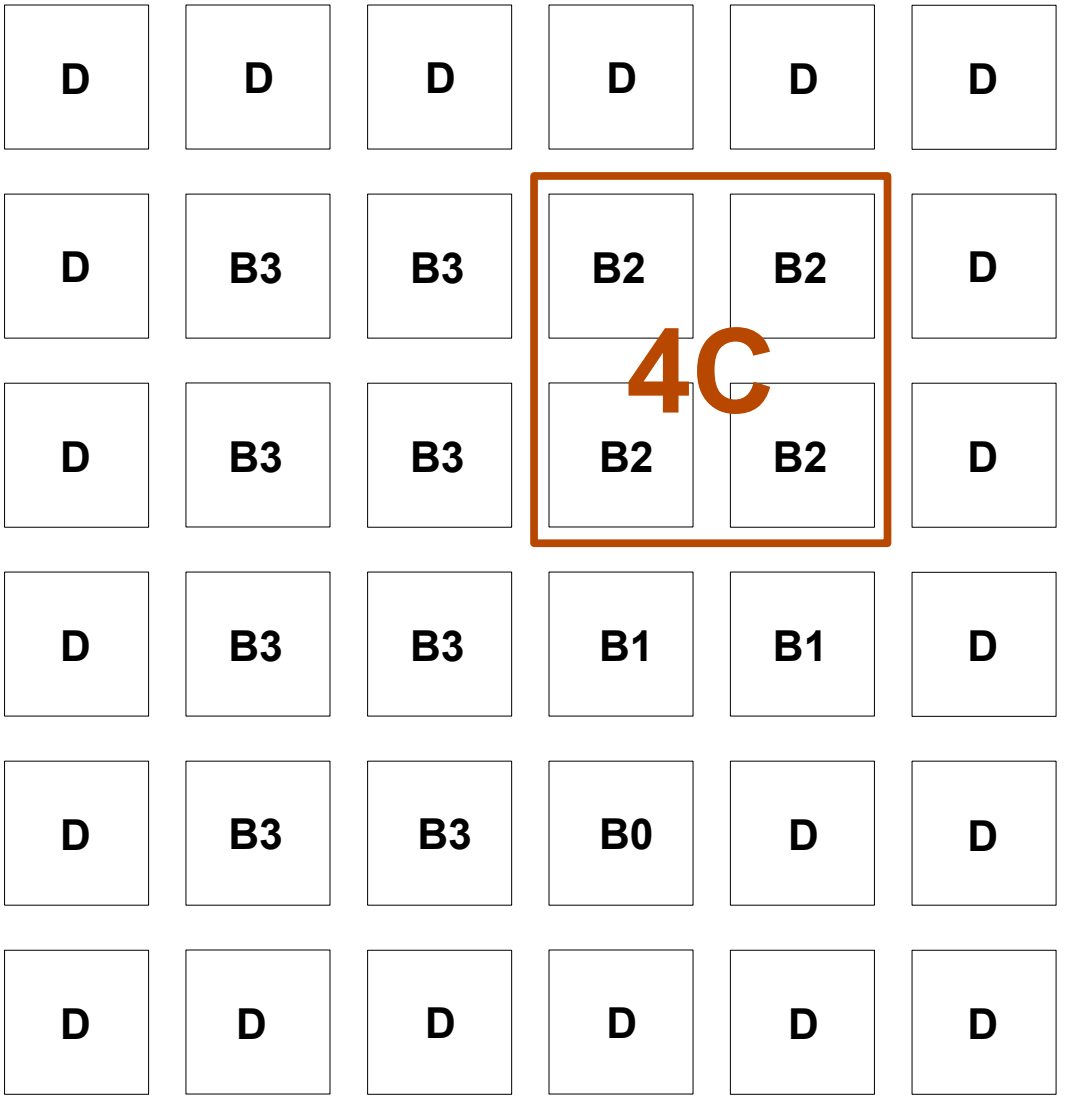
Layout floorplan (5 bit)



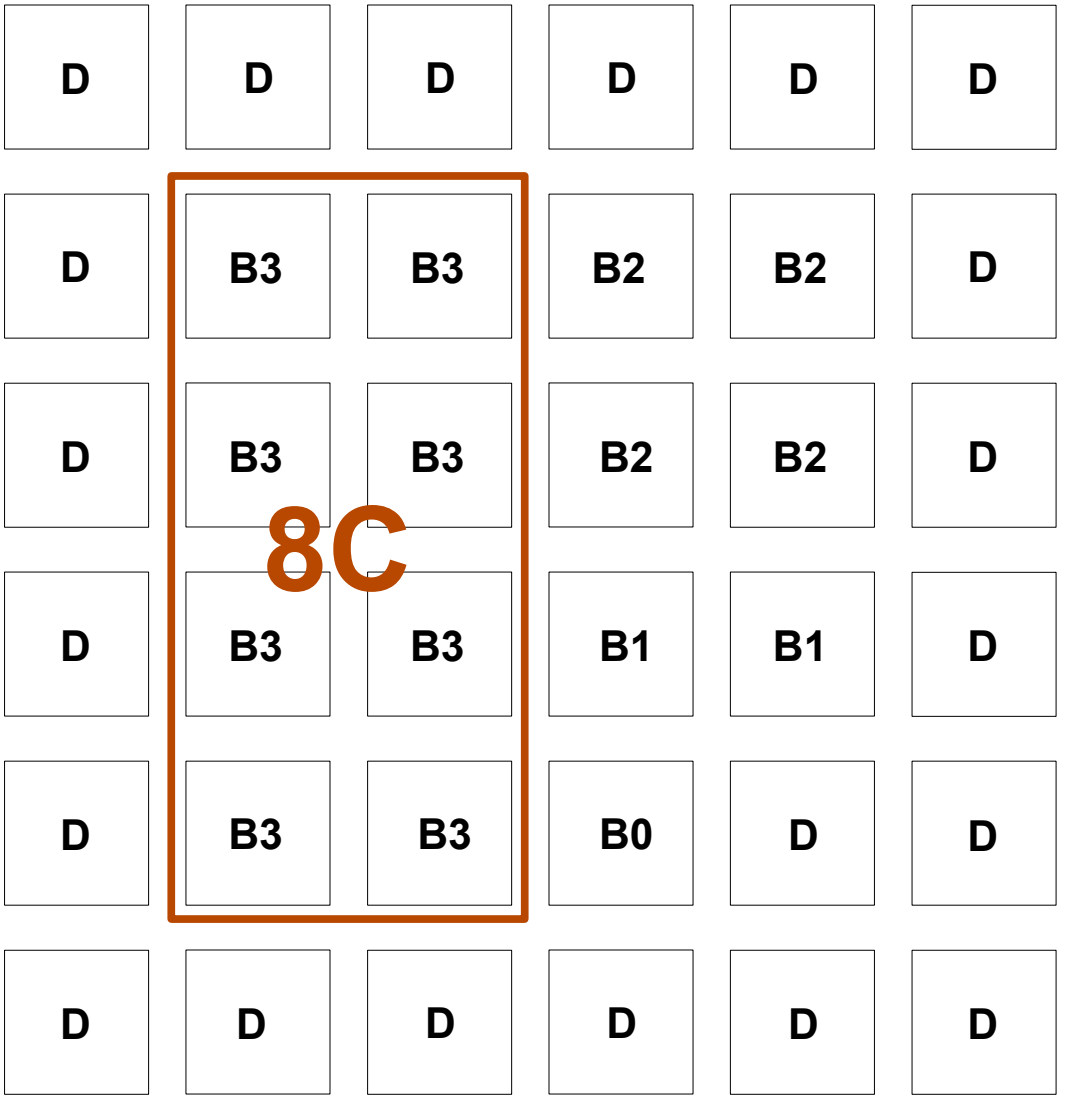
Layout floorplan (5 bit)



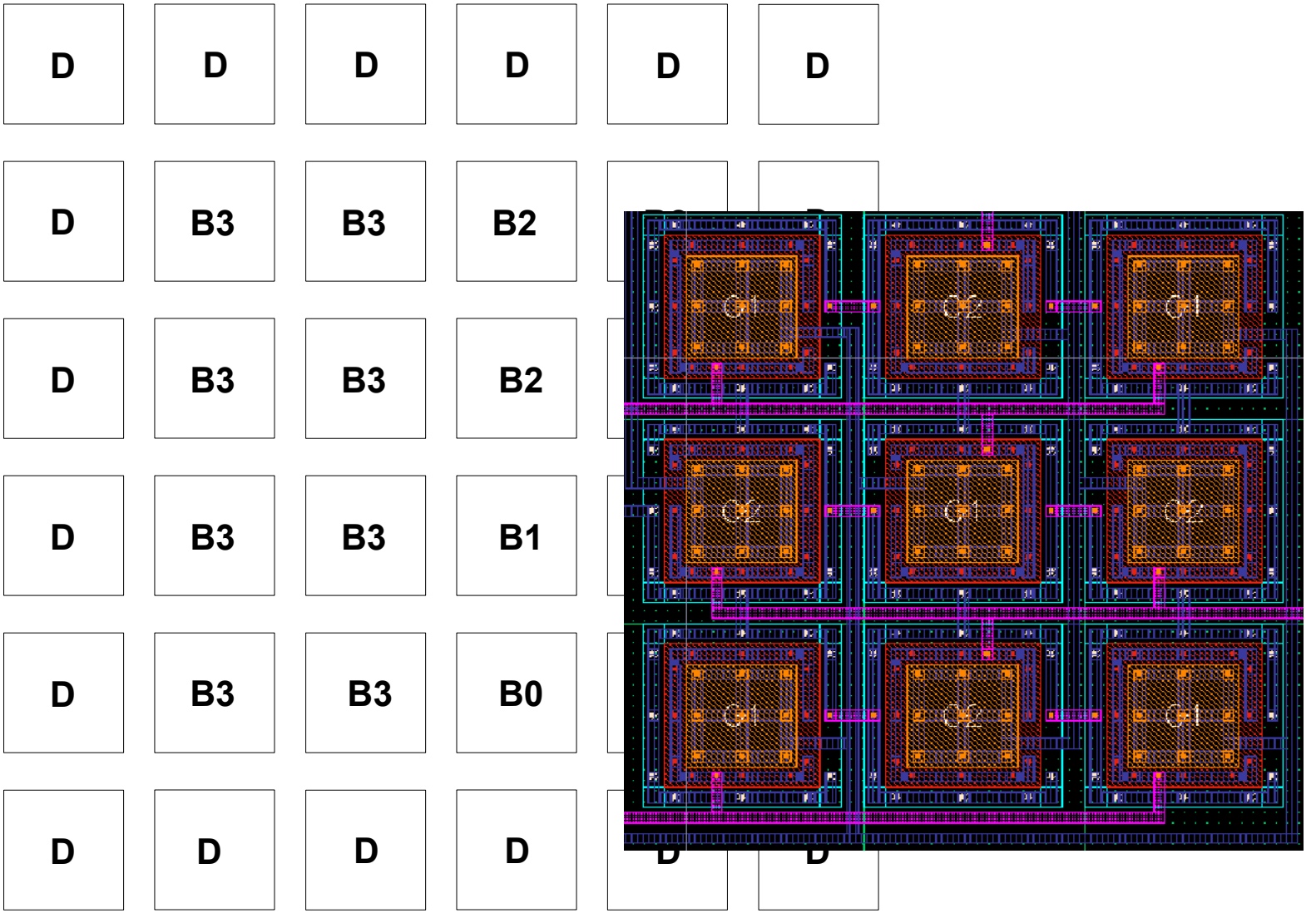
Layout floorplan (5 bit)



Layout floorplan (5 bit)

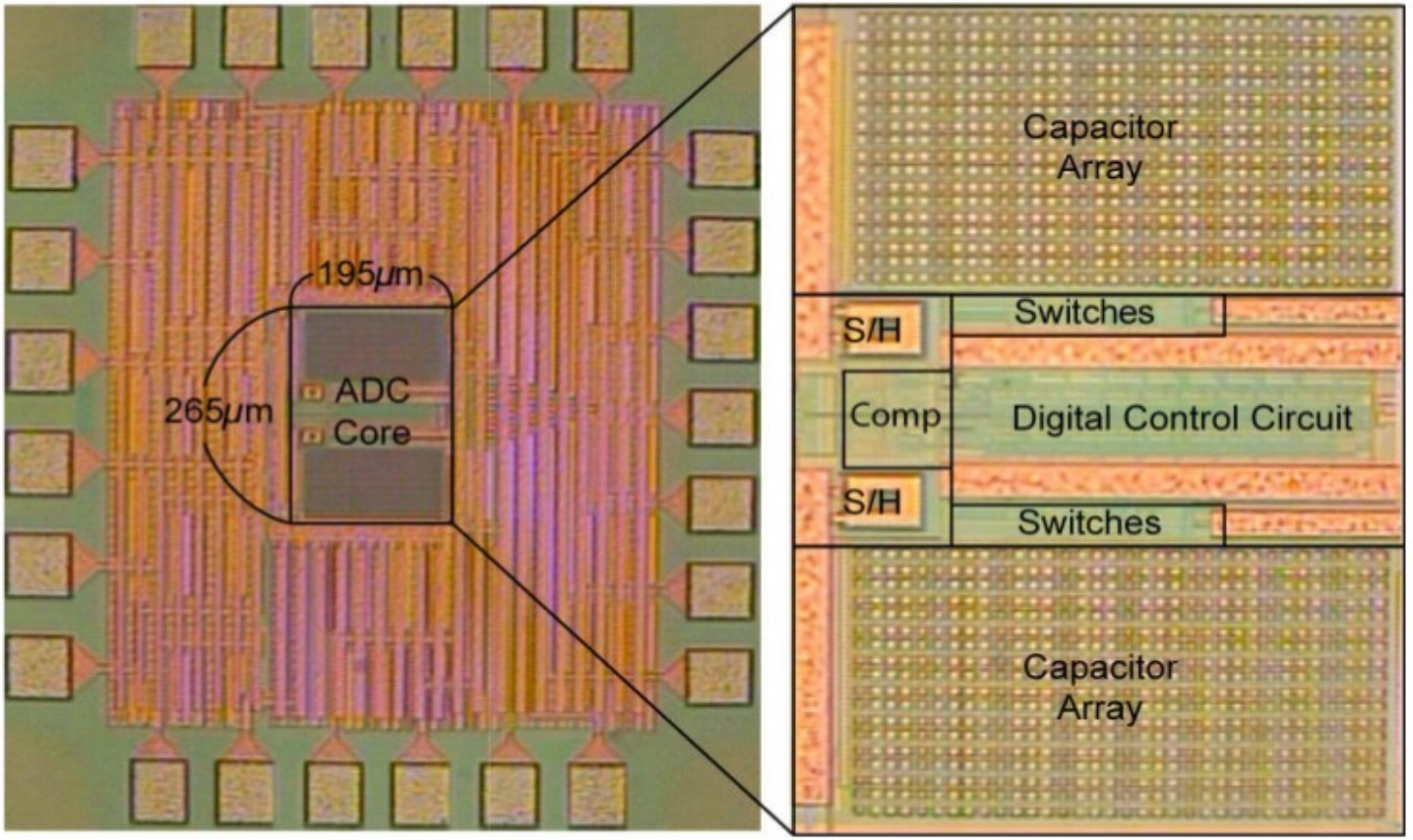


Layout floorplan (5 bit)



130nm CMOS references

- 1.2 V supply voltage
- 195 µm x 265 µm = 0.052 mm^2 total area
- 0.92 mW total power consumption
- layout not yet explored in 65nm



System non-linearity

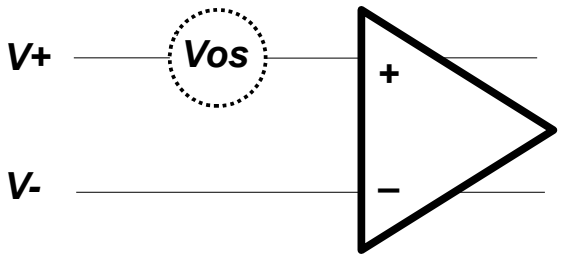
- each unit capacitor is affected by ***process variations*** and ***mismatches*** :

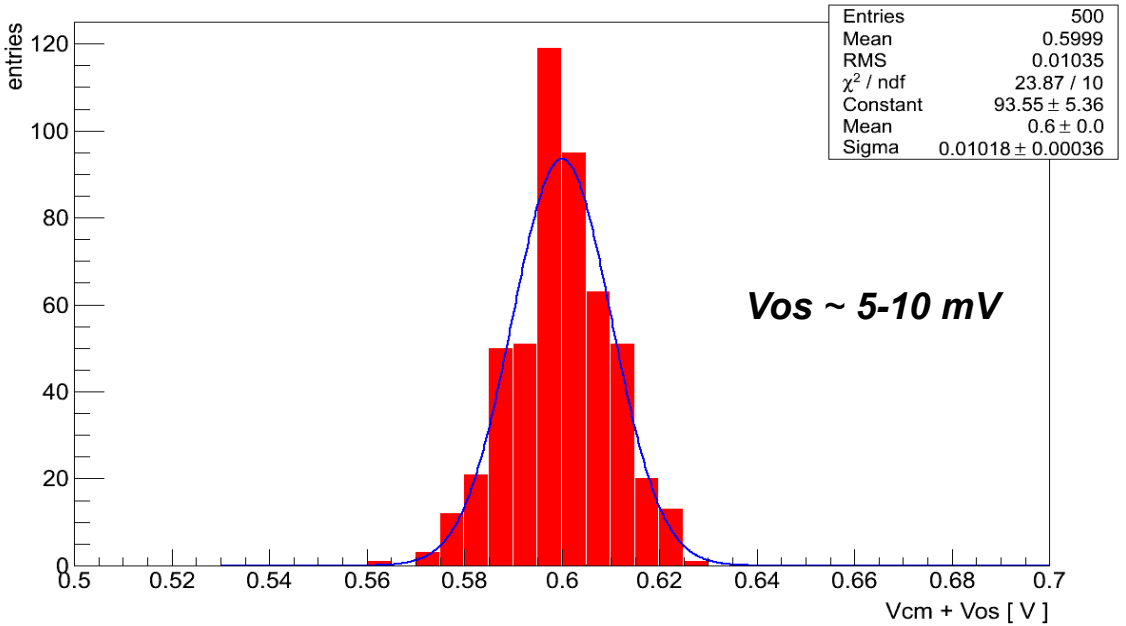
$$\left(\frac{\Delta C}{C}\right)^2 = \left(\frac{\Delta \varepsilon_{ox}}{\varepsilon_{ox}}\right)^2 + \left(\frac{\Delta t_{ox}}{t_{ox}}\right)^2 + \left(\frac{\Delta W}{W}\right)^2 + \left(\frac{\Delta L}{L}\right)^2$$

mismatches affect ***geometrical*** *quantitites*

- capacitor values are ***extracted*** during PEX, thus no SPICE models are available for MC simulations !
  - ***mismatches effect*** investigated with high level simulations (C++/Python, Verilog-A)
  - the whole DAC network works fine up to ~20% mismatches (5fF basic unit capacitor)
  - ***dummy capacitors*** are used in the layout to reduce mismatches

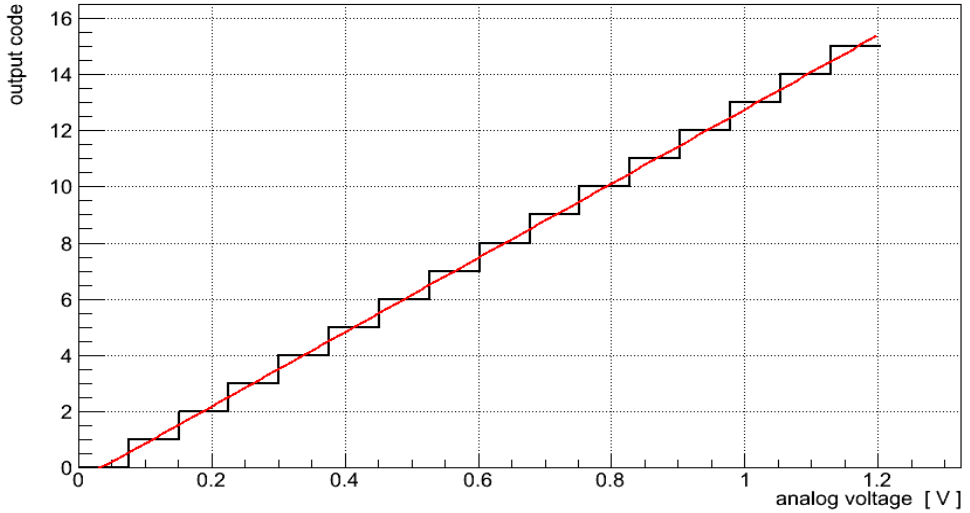
- comparator ***offset***



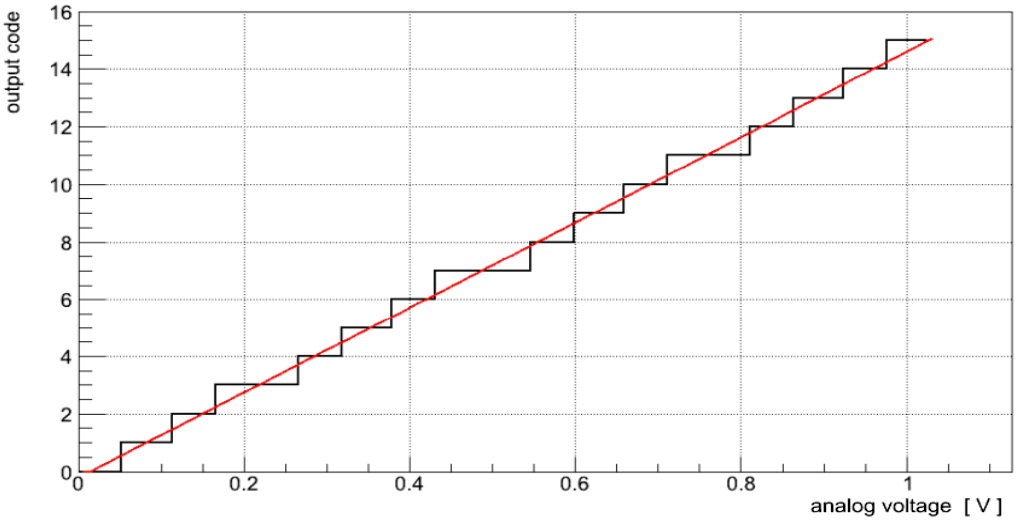


ADC characteristics

***ideal ADC* (4 bit example)**



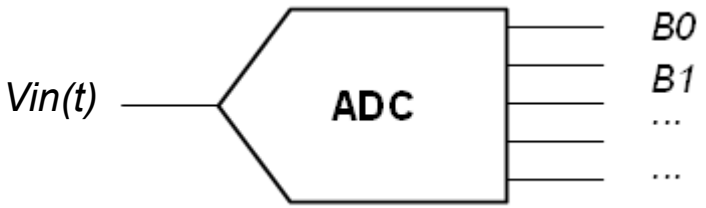
***20% unit capacitor mismatch* (4 bit example)**

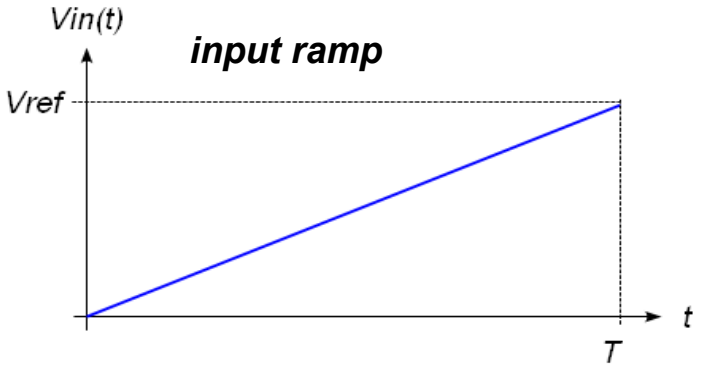


Process variations and mismatches introduce :

- ***differential non linearity (DNL)***
- ***integral non linearity (INL)***
- ***offset error***
- ***gain error***
- ***non-monotonicity error***
- ***missing codes***

Code density (static)



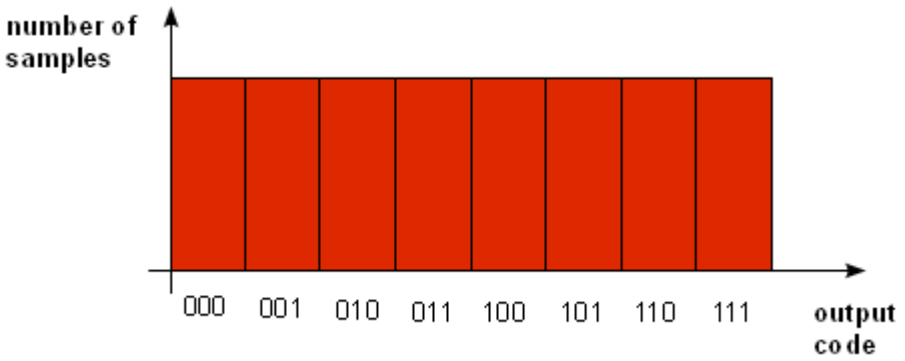


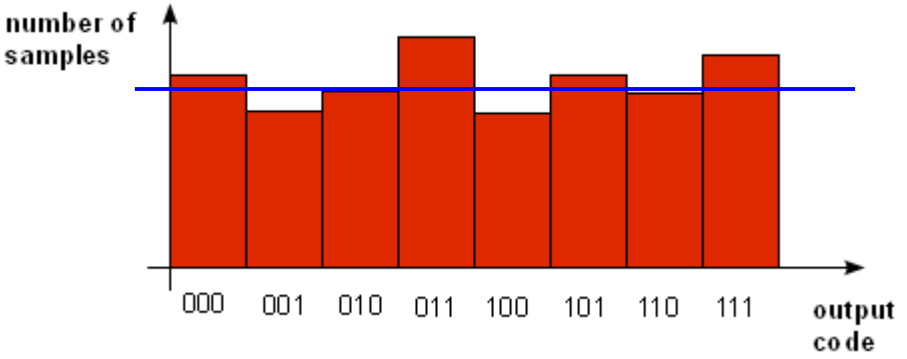
$$v_{in}(t) = k\,t$$

$$k = V_{ref}/T$$

$$f(t) = \begin{cases} 1/T & 0 < t < T \\ 0 & \text{otherwise} \end{cases} \quad \Longrightarrow \quad g(v_{in}) = \frac{1}{V_{ref}}$$

input voltage PDF

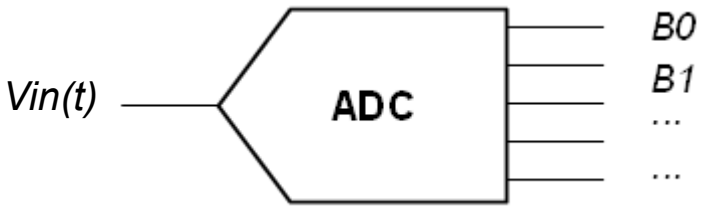


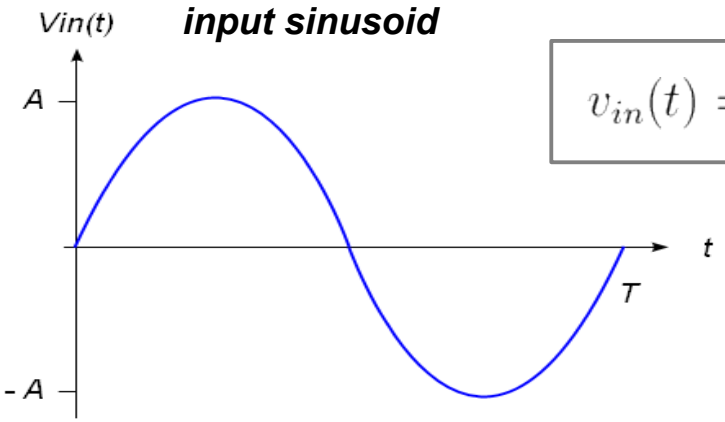


Refresh

$$v_{in} = v_{in}(t) \quad \Longrightarrow \quad g(v_{in}) = \left| \frac{\partial\, t(v_{in})}{\partial\, v_{in}} \right| f(v_{in})$$

Code density (dynamic)





input sinusoid

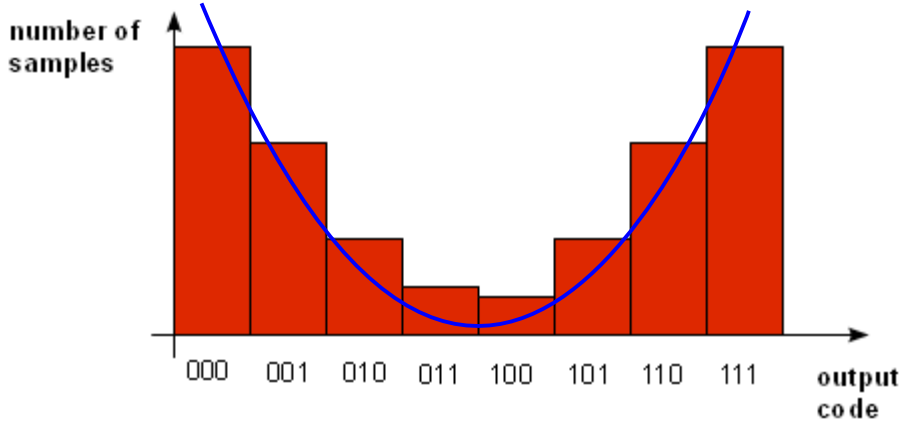
$$v_{in}(t) = A \sin \omega t$$

$$f_{sig} < f_s/2$$

$$f(t) = \begin{cases} 1/T & 0 < t < T \\ 0 & \text{otherwise} \end{cases} \Longrightarrow$$

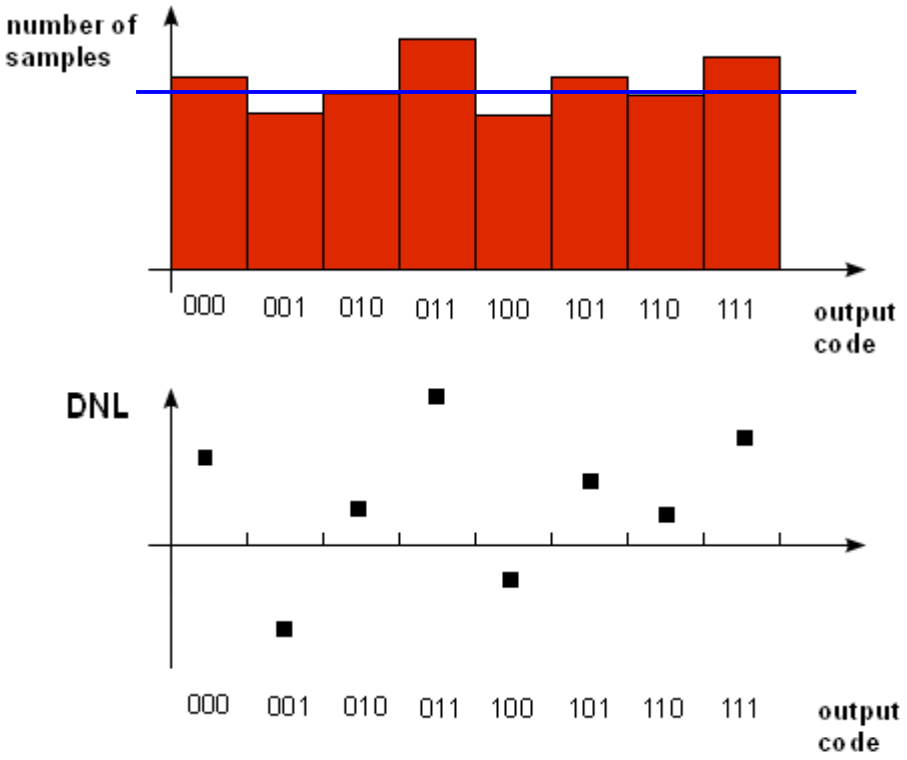
$$g(v_{in}) = \frac{1}{\pi \sqrt{A^2 - v_{in}^2}}$$

input voltage PDF

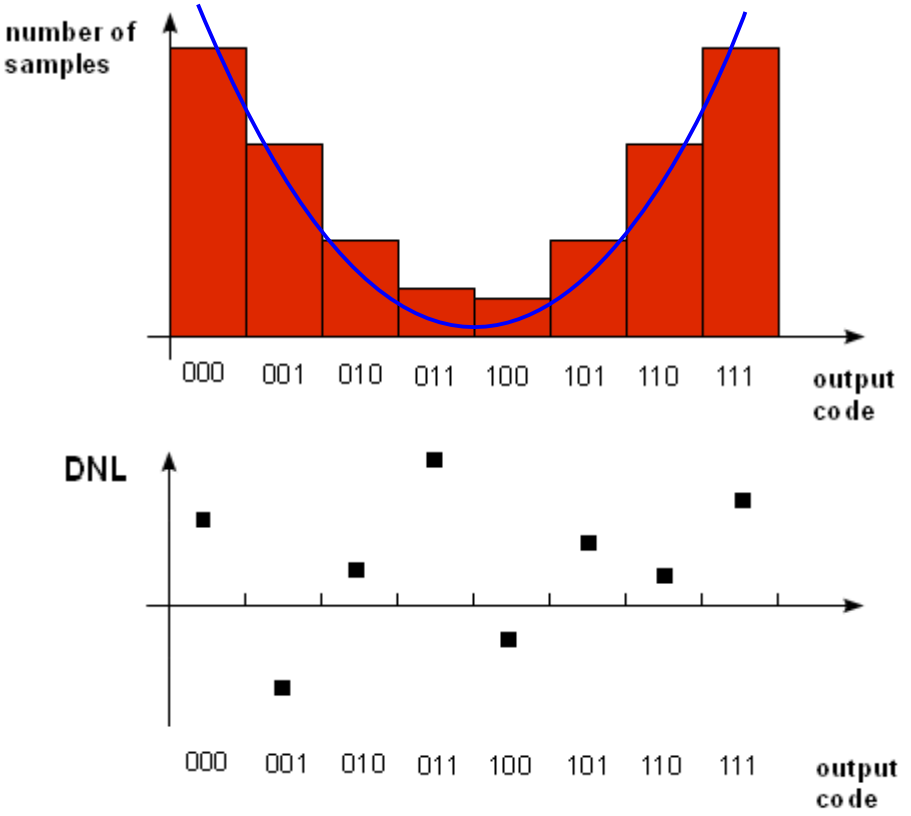


DNL and INL

static characteristic



dynamic characteristic

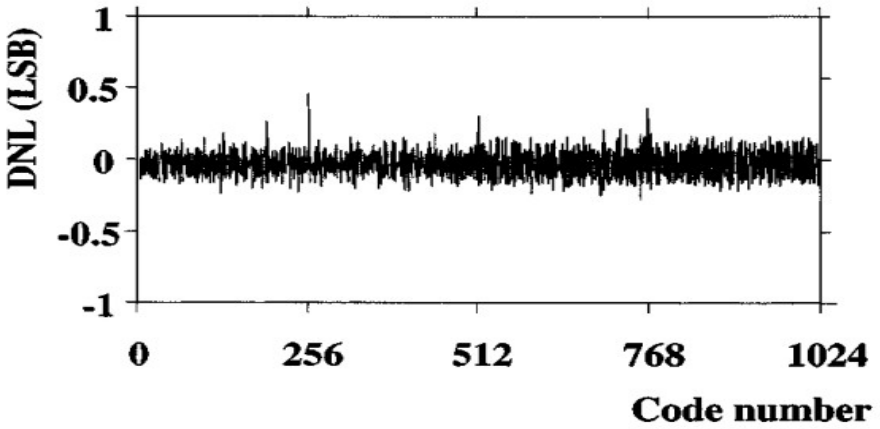


$$DNL_k = \frac{n_k - PDF(k)}{PDF(k)}$$

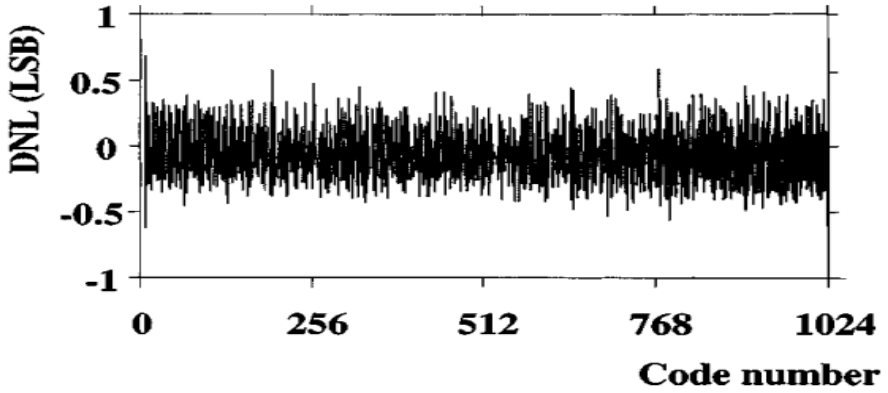
$$INL_k = \sum_{j=0}^{k} DNL_j$$

10 bit ADC example (ALICE SDD readout)

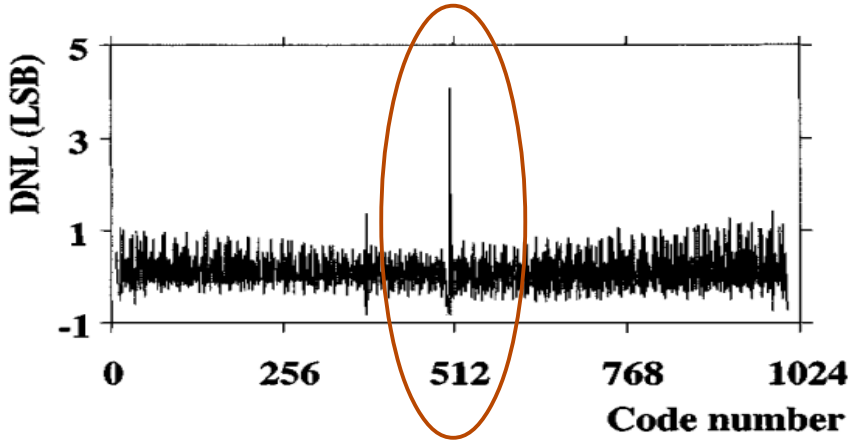
20 MHz clock



30 MHz clock

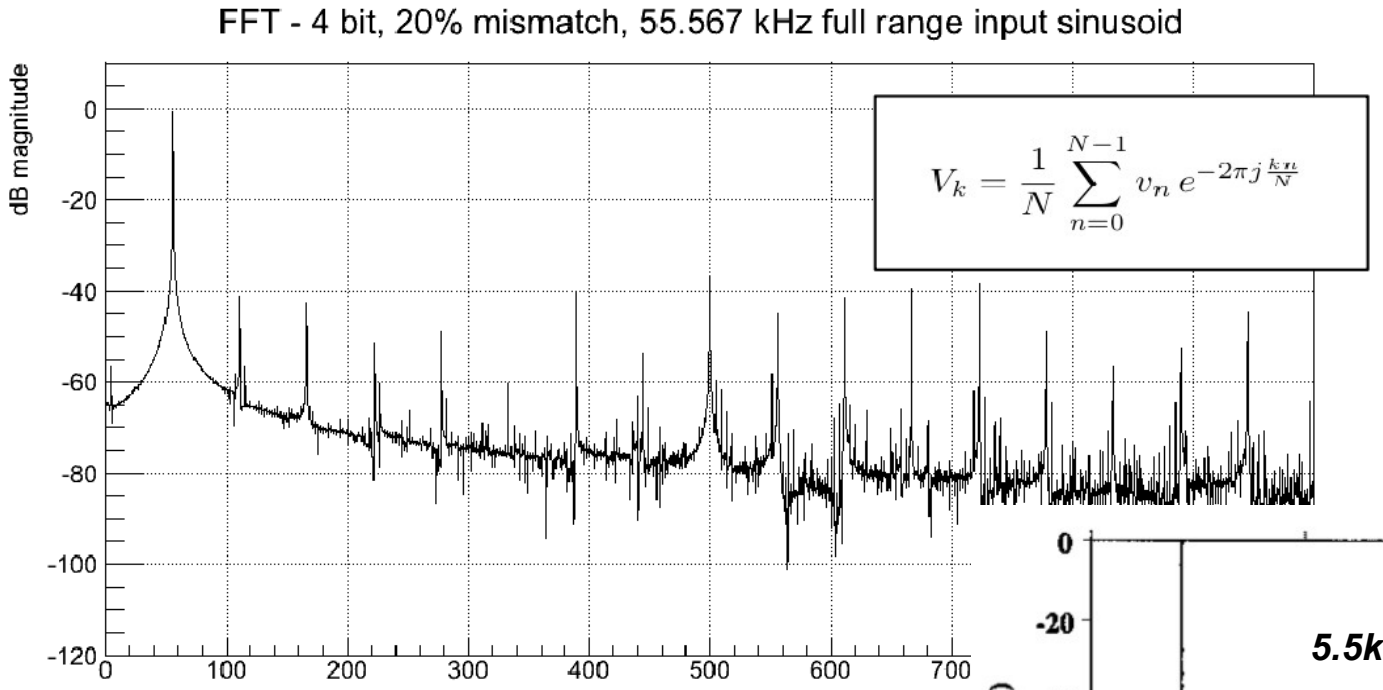


40 MHz clock



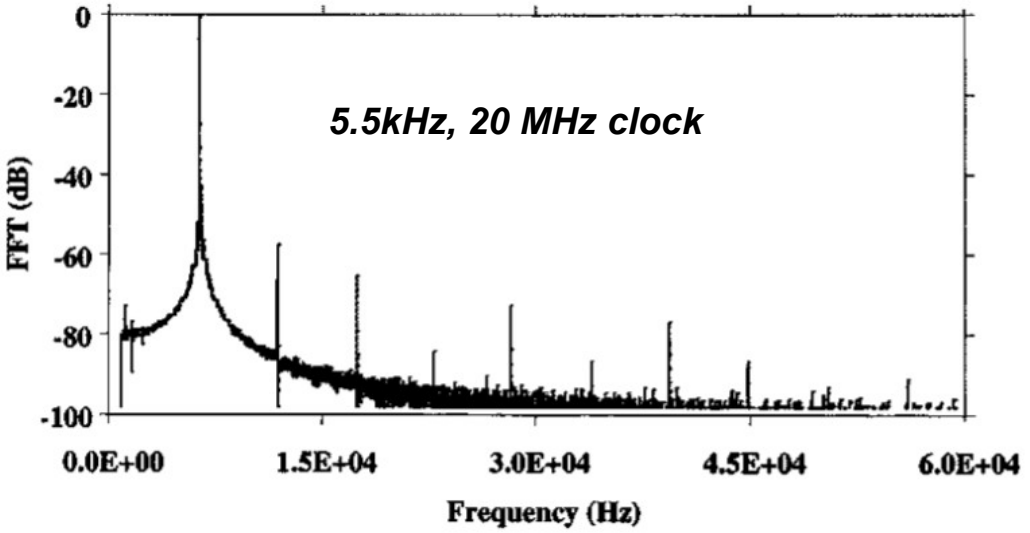
*A spike in the DNL pattern indicates a **missing code** !*

FFT analysis



$$V_k = \frac{1}{N}\sum_{n=0}^{N-1} v_n\, e^{-2\pi j \frac{kn}{N}}$$

$$f_{sig} < f_s/2$$



Performance metrics :

- ***total harmonic distortion (THD)***
- ***signal to noise ratio (SNR)***
- ***signal to noise and distortion ratio (SNDR/SINAD)***
- ***spurious free dynamic range (SFDR)***
- ***effective number of bits (ENOB)***

You can fine this terms in manufactures' data sheets to characterize ADCs and DACs performance

Any questions ?