# 7bit_1.2G_SAR

**Author: 王靖翔**

## 1. post-simulation

corner: tt_mc

| | THIS WORK |
|---|---|
| architecture | SAR |
| Technology | 28nm |
| Sampling rate | 1.2GS/s |
| Resolution | 7bits |
| Supply | 0.9V |
| Input Swing | 0.3V |
| SNDR | 38.75dB |
| Power | 2.03mW |

In other corners, the structure can work normally.

## 2. Details of layout

| CELL NAME | LAYOUT |
|---|---|
| 7bit_SAR_DEBUG_core_V2 | layout |
| SAR_Cap_DAC_7bits | layout_5:3_INV |
| Logic_SAR_3_3 | layout |
| Logic_SAR_3_4 | layout |
| Logic_SAR_dynamic_layout_3 | layout |
| Comparator_7bit_cal_debug | layout_h2 |

> - *SAR_Cap_DAC_7btis: schematic needs checking LVS and schematic_ideal is used in just schematic*

## 3. Extra messages

The input swing of this project has just 300mV, it makes the Vref smaller than high input swing. It will make the conversion speed slowly. For dealing with the problem, a high parasitic capacitor is added in CDAC. For N bits SAR ADC, the efficient voltage on the nth conversion is as formula 1. efficient

$$V_n = \frac{C_{unit} * 2^{n-1}}{C_{unit} * 2^{N-1} + C_{parasitic}}$$

After that, SNDR will be changed, as formula 2.

$$SNDR = 6.02 * N + 1.76 + 20 * lg\frac{input\ swing * C_{total\ cap}}{Vref * C_{effcient\ cap}}$$

In code part, the code will be shown.

## 4. Code

```
import numpy as np

Nbit = float(input("please input Nbit  "))
VREF = float(input("please input Vref mV  "))*1e-3
input_swing = float(input("please input input swing mV "))*1e-3
total_cap = float(input("please input total cap fF "))
effiency_cap = float(input("please input effiency cap fF "))
SNDR =
6.02*Nbit+1.76+20*np.log10(input_swing/(VREF*effiency_cap/total_cap
))
print("the SNDR is %.2f dB" %SNDR)
```

## 5. version

| VERSION | CHANGE | TIME AND AUTHOR |
| --- | --- | --- |
| V1 | add basic information | 1.21.2024 Jingxiang |
| V2 | change some name of layout | 1.22.2024 Jingxiang |