IEEE International Symposium on Intelligent Signal Processing and Communication Systems 2017

NOVEMBER 6-9, 2017, XIAMEN, CHINA

Nov. 8 WP-L2 14:30-15:50

Study of Jitter Generators For High-Speed I/O Interface Jitter Tolerance Testing

Y. Ozawa, T. Arafune, N. Tsukiji,

H. Kobayashi, R. Shiota

Gunma University, Socionext Inc.,

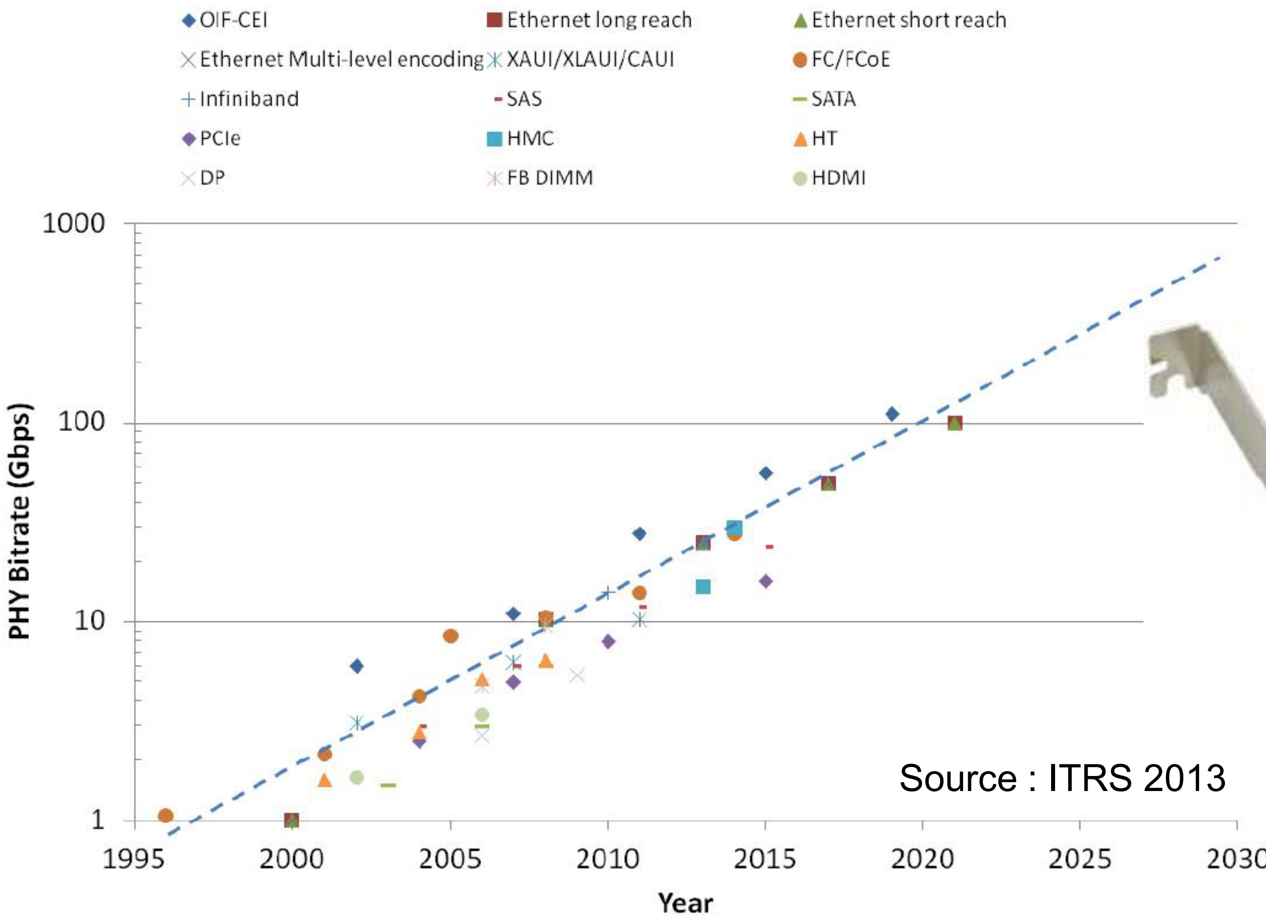
Purpose

- To develop jitter generation methods for high-speed I/O interface testing
  - Low cost
  - Flexible jitter generation
  - Easy to design
  - Suitable for BOST

BOST: Built-Out Self-Test

Application

Jitter generator for high speed I/O interface tolerance testing



Data rate increasing ➡ **Jitter margin decreasing**

Jitter tolerance testing is necessary at low cost

Innovation

Proposed method Ⅰ
Using "inter-symbol interference"

Proposed method Ⅱ
Using "digital modulator"

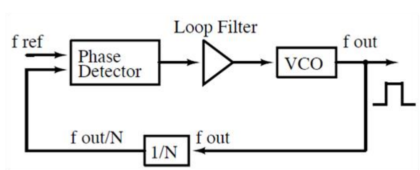
Mostly digital implementation

Low cost & Easy to design

Conventional methods


BERT(Bit Error Ratio Tester)

Expensive



Analog PLL

Difficult to Design

OUTLINE

- Proposed Method Ⅰ
  - Inter-symbol Interference (ISI)
  - Jitter Generator with ISI
  - Simulation
  - Summary
- Proposed Method Ⅱ
  - Jitter Generator with ⊿Σ Modulator
  - Multi-bit
  - Simulation
  - Summary
- Conclusion

OUTLINE

- Proposed Method Ⅰ
  - Inter-symbol Interference (ISI)
  - Jitter Generator with ISI
  - Simulation
  - Summary
- Proposed Method Ⅱ
  - Jitter Generator with ⊿Σ Modulator
  - Multi-bit
  - Simulation
  - Summary
- Conclusion

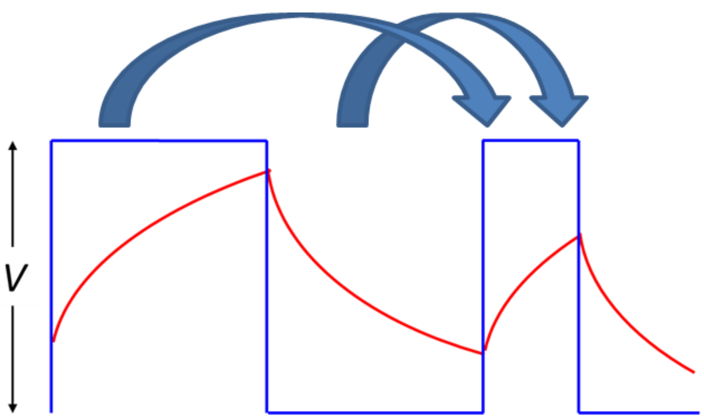
Inter-symbol Interference

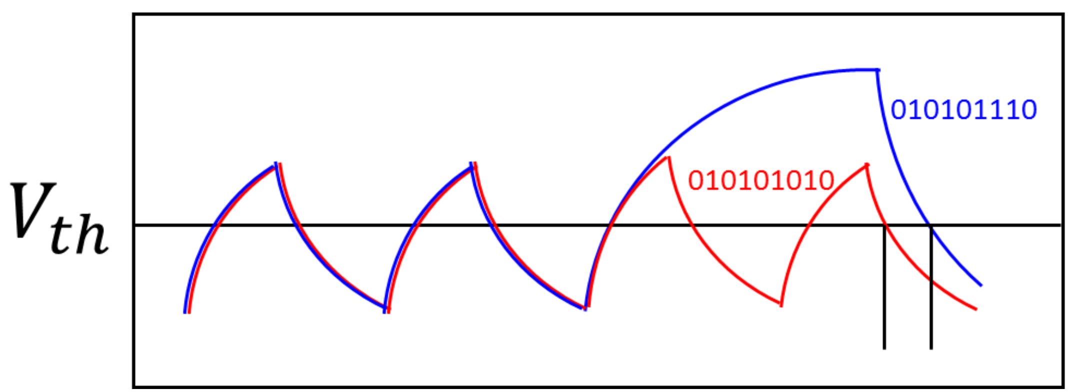
Jitter Classification

- **Timing**
  - **Random**
  - **Deterministic**
    - **Data Dependent**
      - **Inter-symbol interference**
      - **Duty Cycle Distortion**
    - **Periodic**

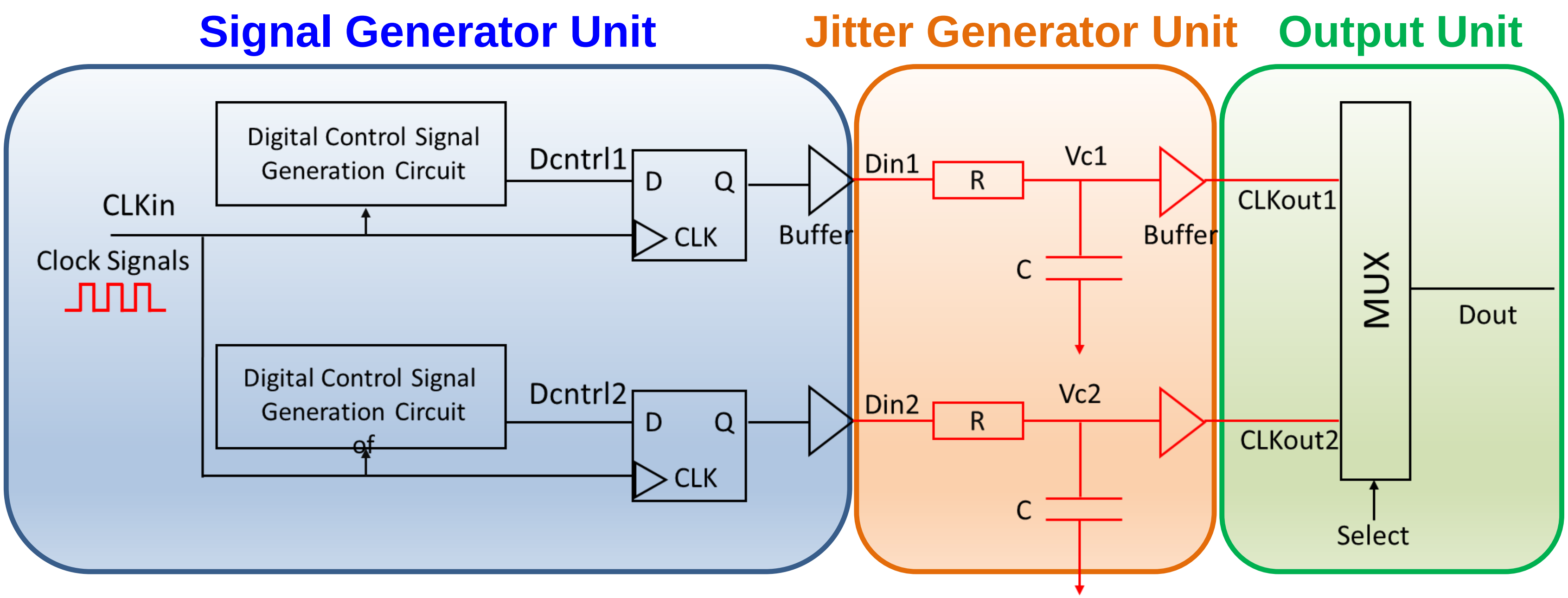
Preceding signals → Subsequent signals

affect





Proposed Circuit

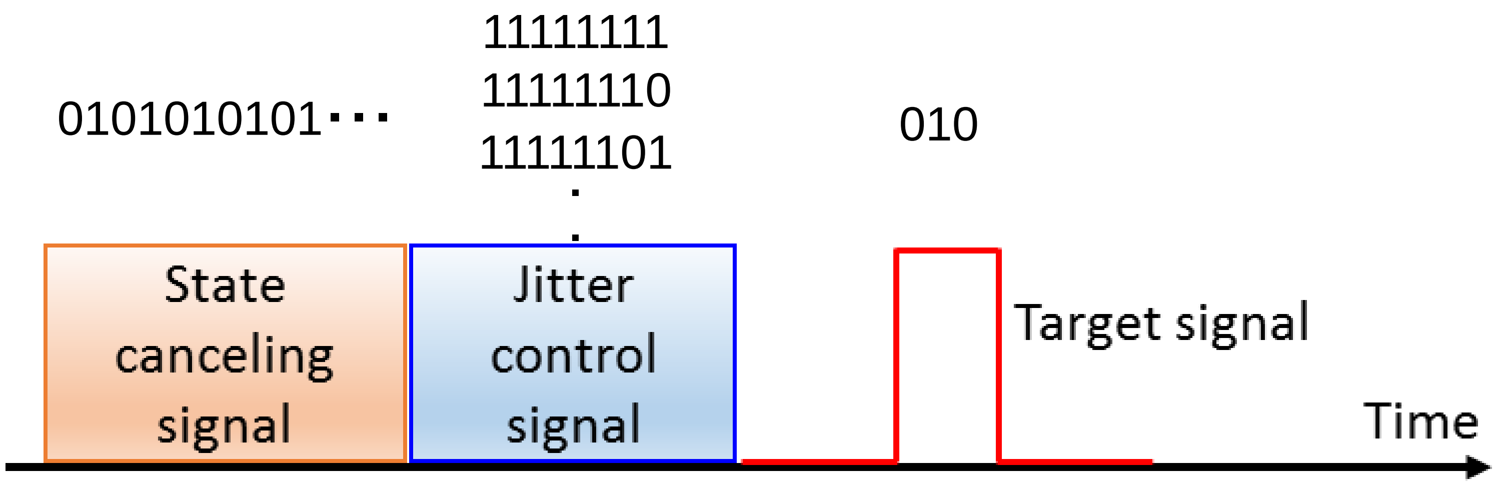


- **Signal Generator Unit**
  Digital signals ➡ control ➡ Target jitter amount

- **Jitter Generator Unit**
  Delay & jitter generation

- **Output Unit**
  Output jitter generated at 1st or 2nd channel

Operation Principle

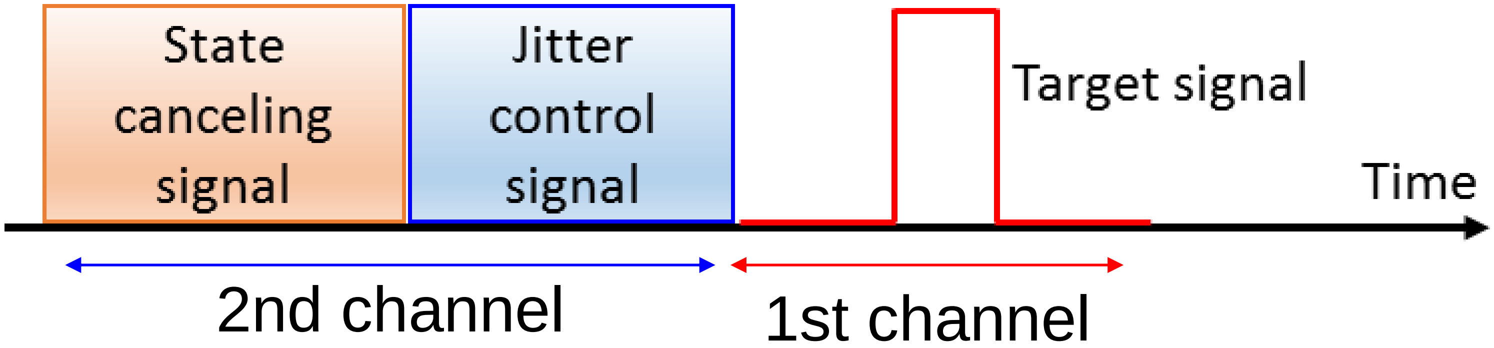


- **State canceling signal**
  Previous signal influence cancellation

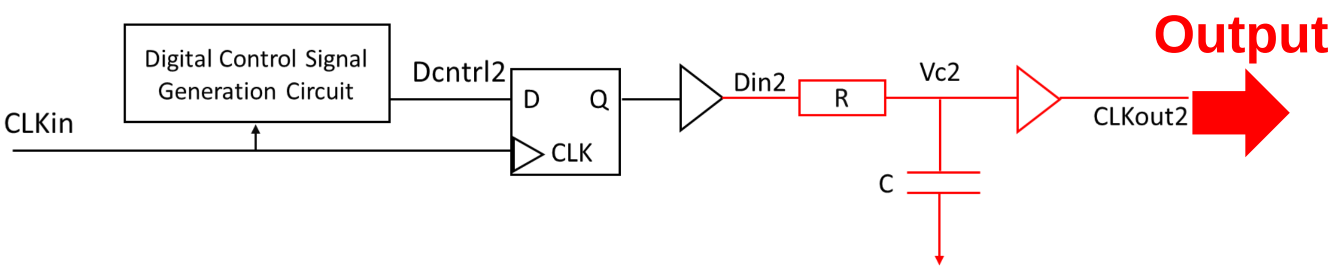
- **Jitter control signal**
  Jitter amount control for target signal

- **Target signal**
  Shifted by jitter control signal

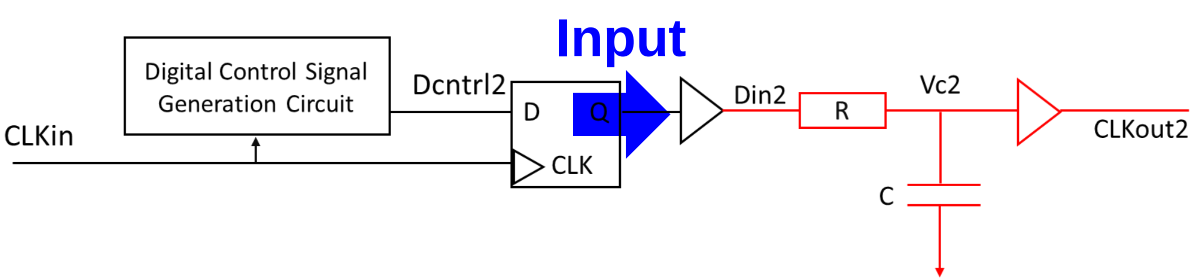
Parallel Operation①



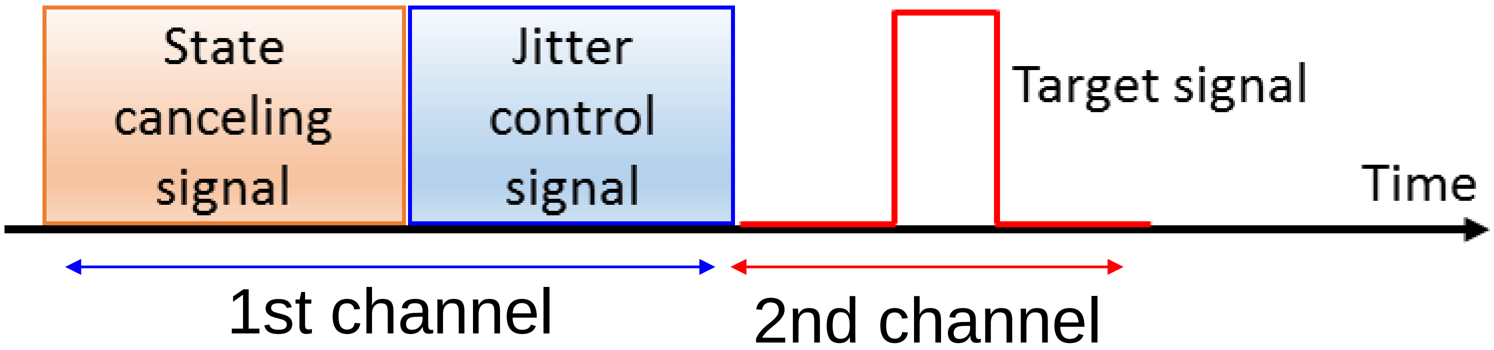
1st channel



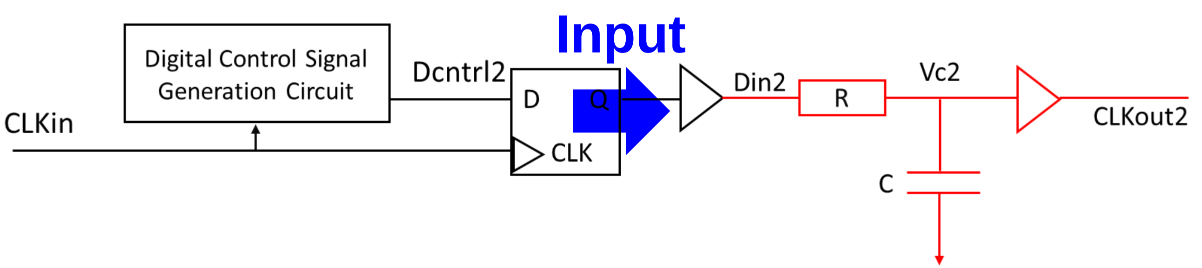
2nd channel



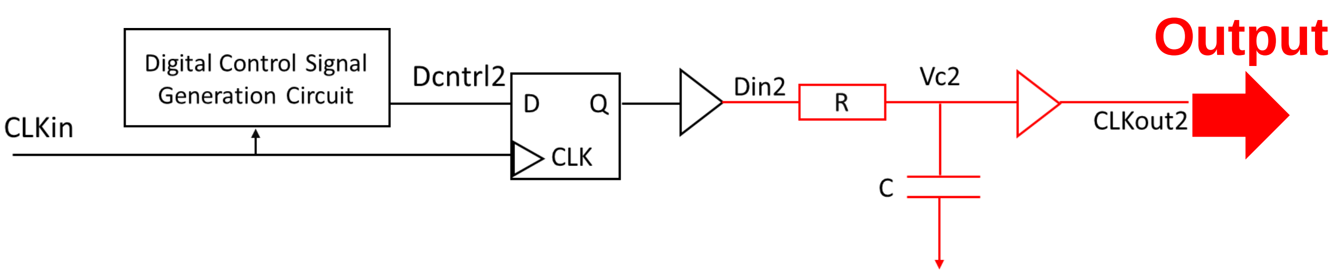
Parallel Operation②



1st channel



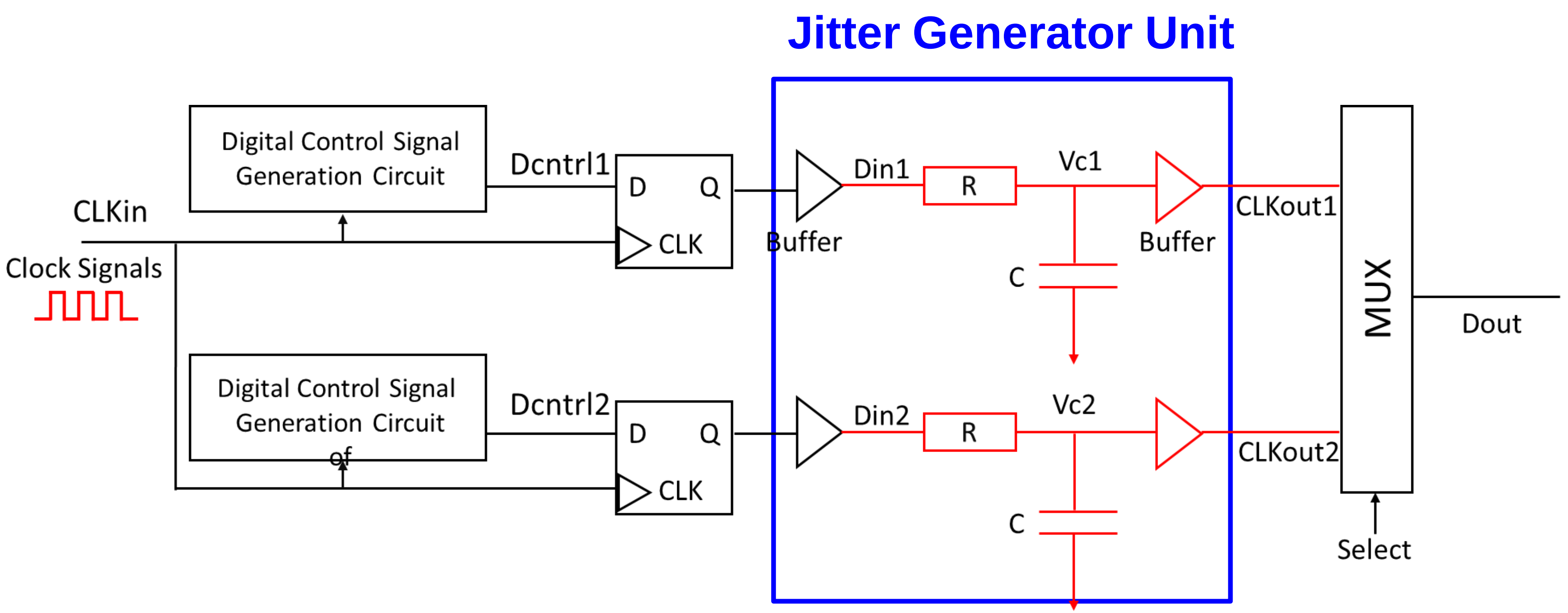
2nd channel



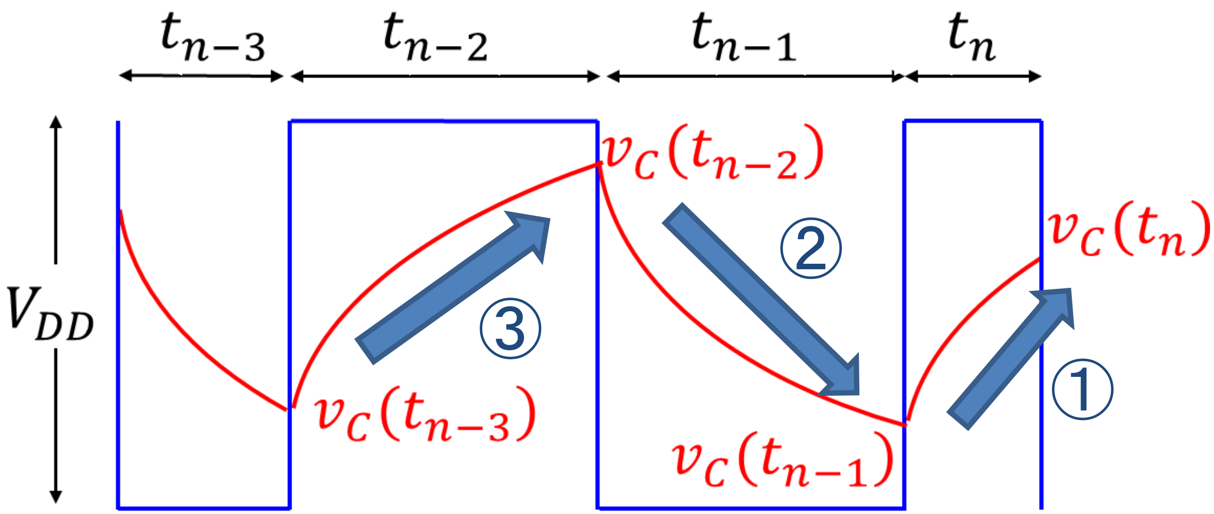
Simulation of RC LPF & Buffer

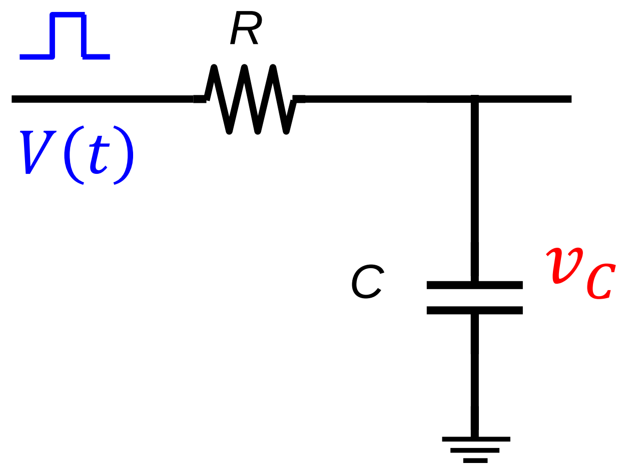
Confirmation

Input digital signals ➡ Expected jitter



Transient Equation (Input = VDD case)





When $V(t_n) = V_{DD}$

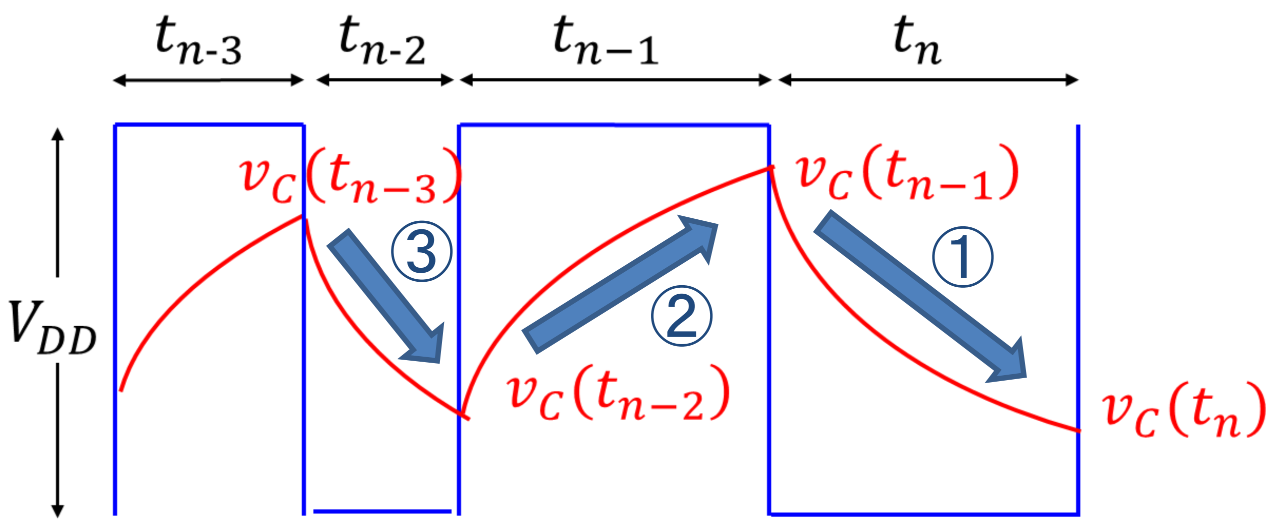
① $v_C(t_n) = V_{DD} + (v_C(t_{n-1}) - V_{DD})\exp(-\frac{t_n}{RC})$

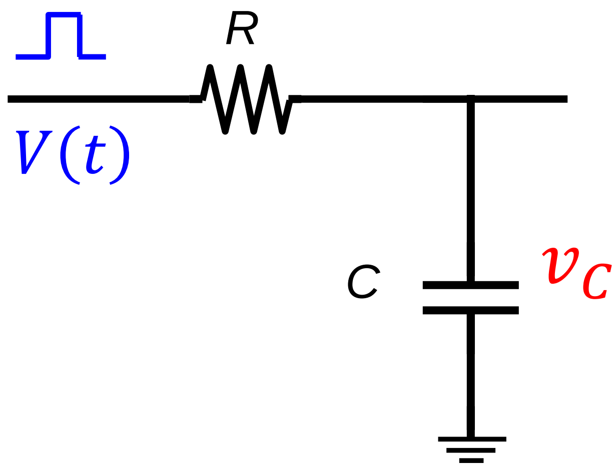
② $v_C(t_n) = \{V_{DD} + (v_C(t_{n-2}) - V_{DD})\exp(-\frac{t_{n-2}}{RC})\}\exp(-\frac{t_n}{RC})$

③ $v_C(t_n) =$

-
-
-
-
-

Transient Equation (Input = 0 case)





When $V(t_n) = \mathbf{0}$

① $v_C(t_n) = v_C(t_{n-1})\exp(-\frac{t_n}{RC})$

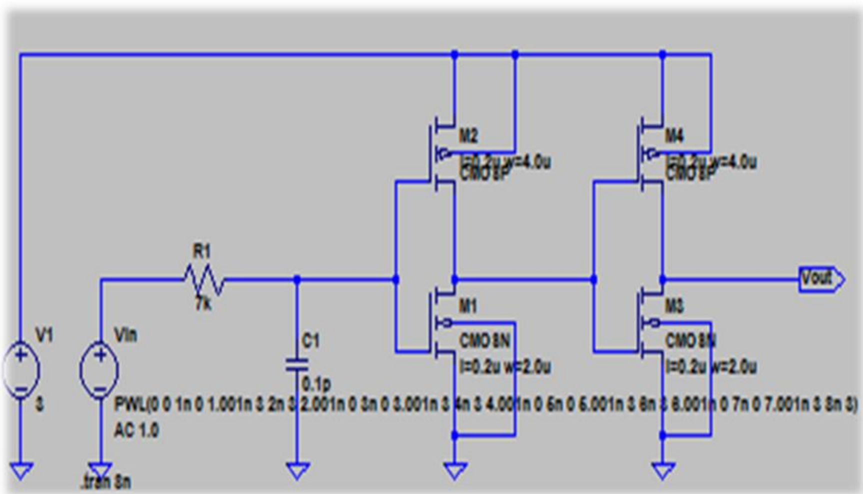
② $v_C(t_n) = V_{DD} + (v_C(t_{n-2})\exp(-\frac{t_{n-2}}{RC}) - V_{DD})\exp(-\frac{t_n}{RC})$

③ $v_C(t_n) =$

-
-
-
-
-

Validation Policy

Circuit simulation (Spice) ⟺ **Numerical simulation (Scilab)**



$$v_C(t_n) = V_{DD} + (v_C(t_{n-1}) - V_{DD})\exp(-\frac{t_n}{RC})$$

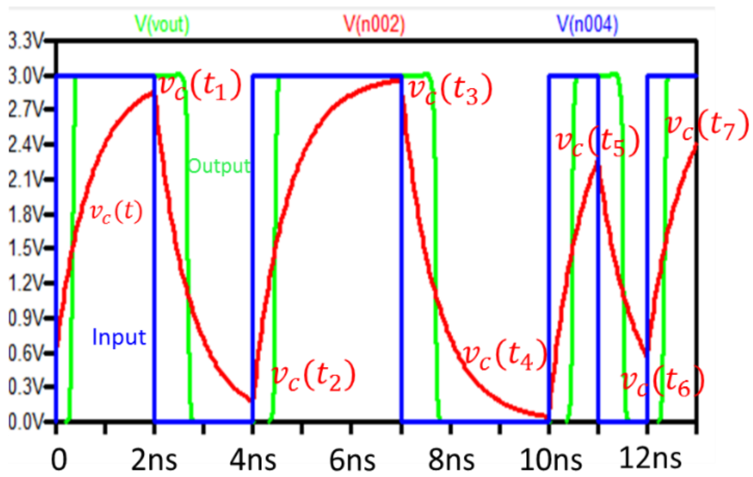
$$v_C(t_n) = \{V_{DD} + (v_C(t_{n-2}) - V_{DD})\exp(-\frac{t_{n-2}}{RC})\}\exp(-\frac{t_n}{RC})$$

$$v_C(t_n) = v_C(t_{n-1})\exp(-\frac{t_n}{RC})$$

$$v_C(t_n) = \{V_{DD} + (v_C(t_{n-2}) - V_{DD})\exp(-\frac{t_{n-2}}{RC})\}\exp(-\frac{t_n}{RC})$$

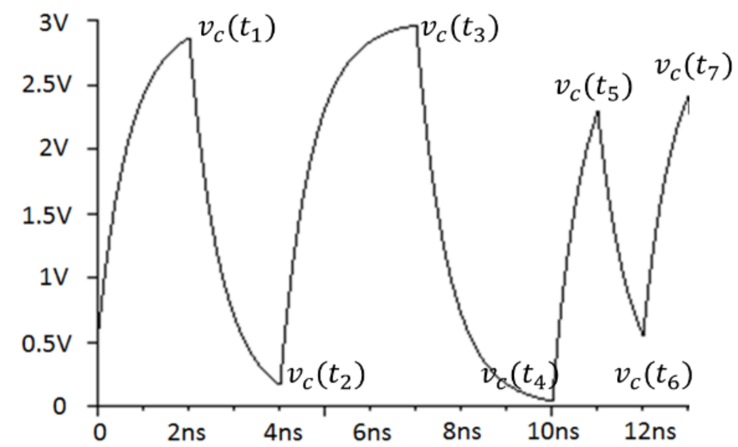
$$\vdots$$

Circuit & Numerical Simulations



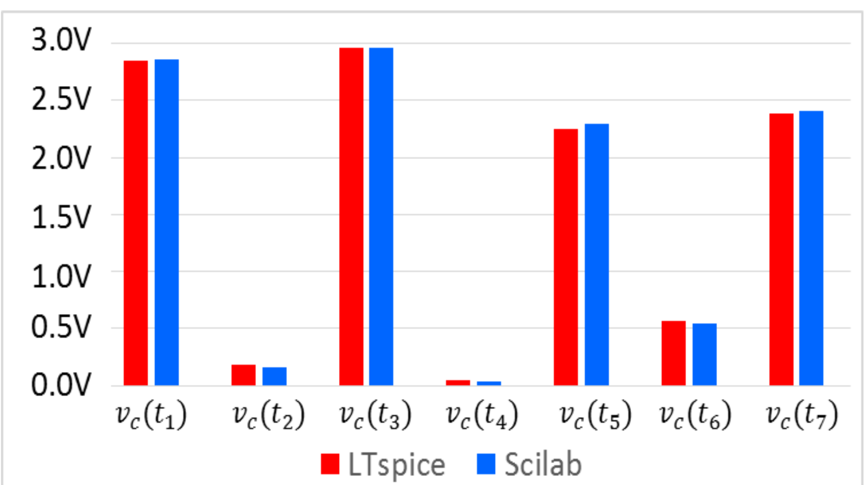
Circuit simulation (LTspice)

Comparison



Numerical simulation (Scilab)

Numerical vs. Circuit simulations



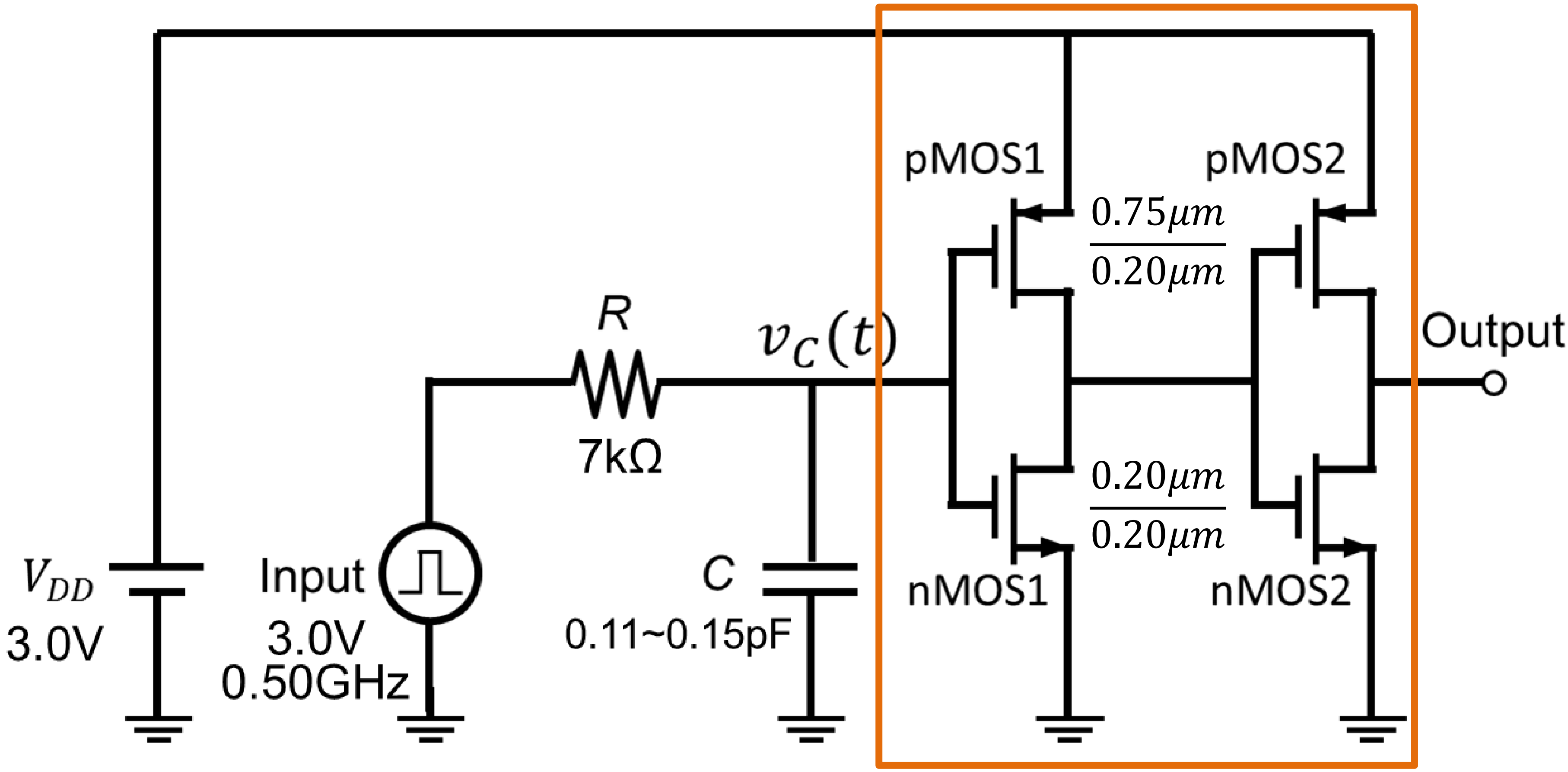
Error average 4.2%

Consistent with derived equation

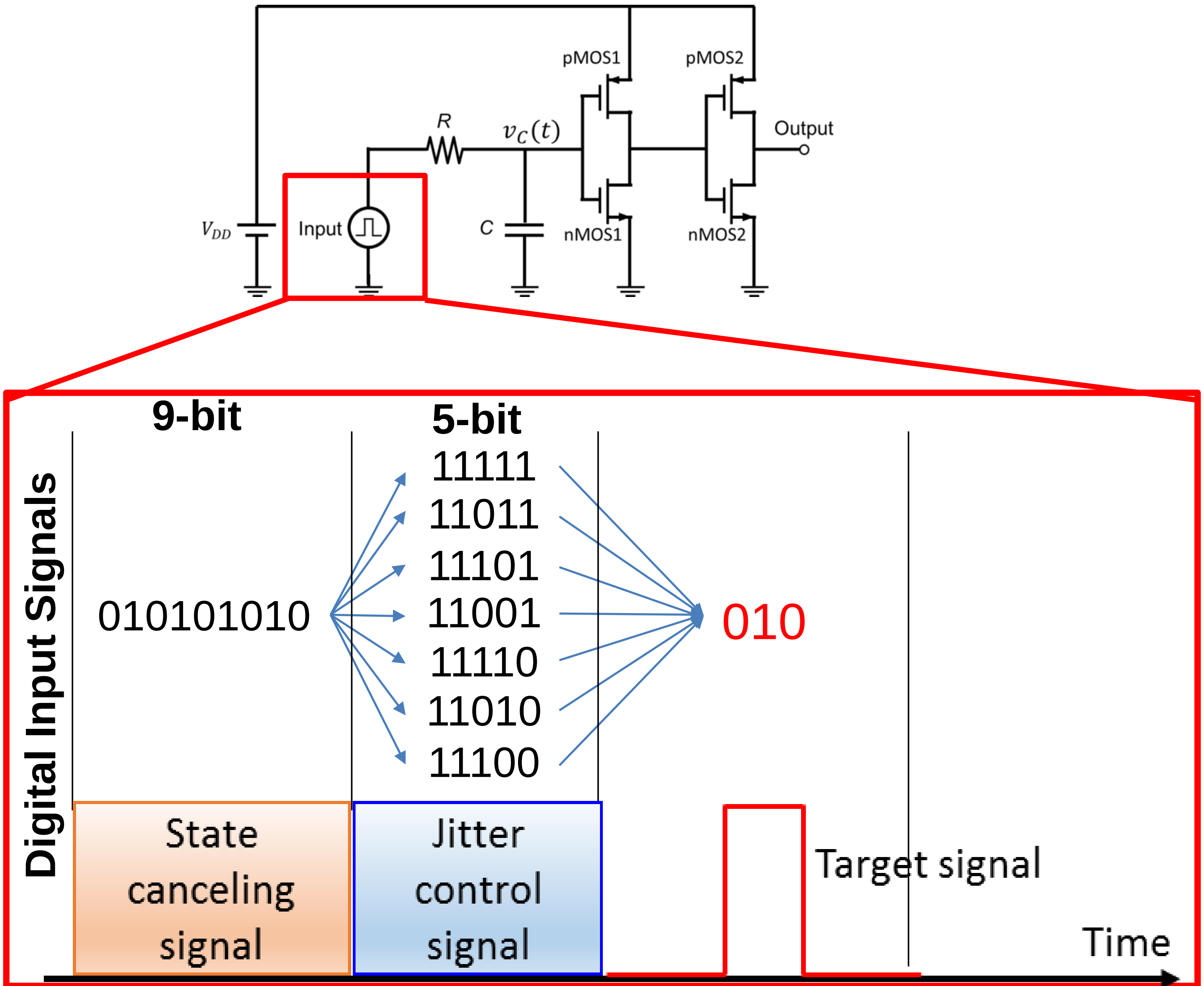
Circuit Simulation Condition

- Simulation software : **Spice**
- Model parameter : TSMC180nm

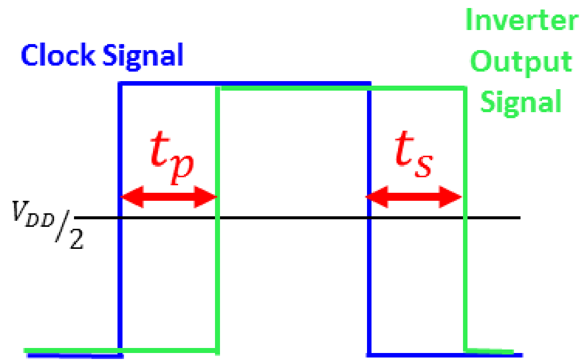
Inverter threshold $= {}^{V_{DD}}/_{2}$



Input Signals of Simulation Circuit



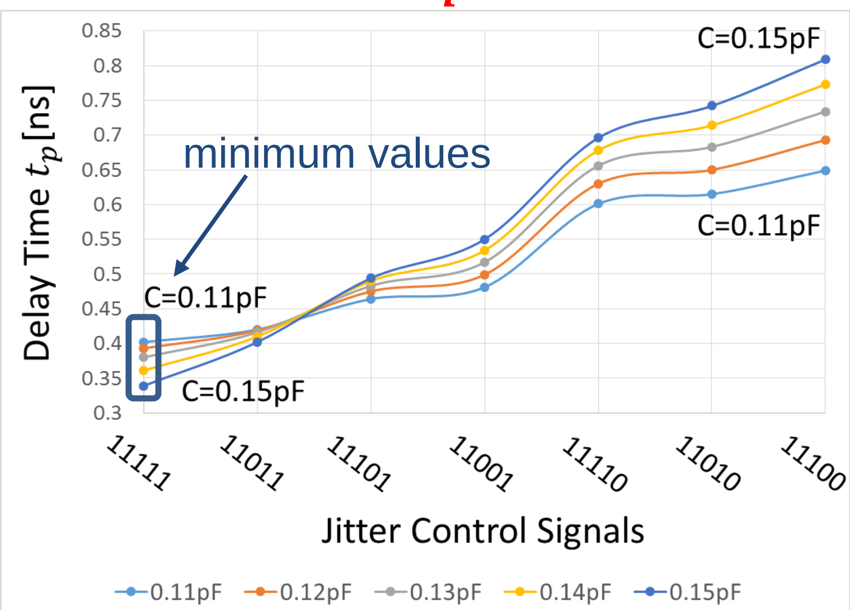
SPICE Simulation Results



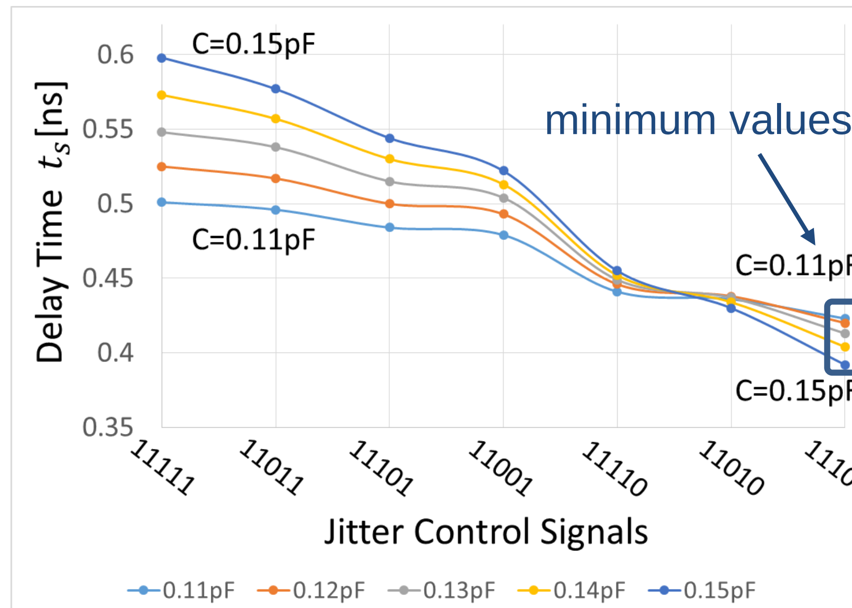
Changing "%" of "t_p"
based on minimum value
Max : 238.6% (0.15pF)
Min : 148.8% (0.11pF)

Changing "%" of "t_s"
based on minimum value
Max :152.6% (0.15pF)
Min :113.0% (0.11pF)

t_p



t_s



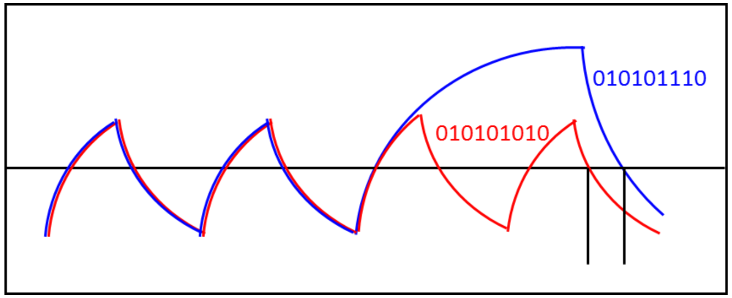
Jitter amounts can be digitally <u>controlled</u>

Summary

Proposed Method Ⅰ
Jitter Generator with ISI

◆ “Inter-symbol interference” positively using method

V_{th}



➢ Jitter amount can be controlled with full digital

➢ Circuit operation was confirmed

by numerical simulation / circuit simulation

Scilab Spice

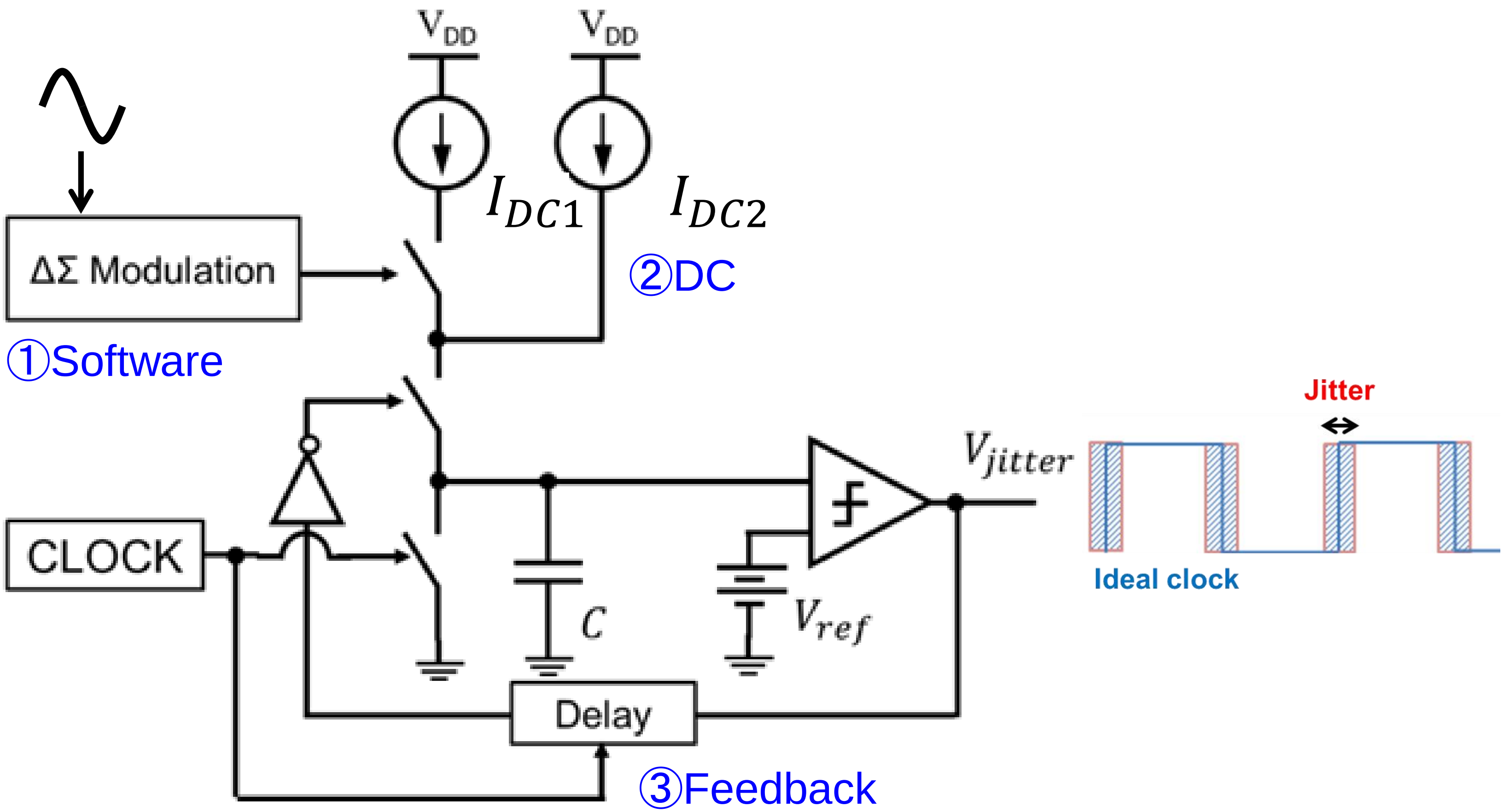
OUTLINE

- Proposed Method Ⅰ
  - Inter-symbol Interference (ISI)
  - Jitter Generator with ISI
  - Simulation
  - Summary
- Proposed Method Ⅱ
  - Jitter Generator with ⊿Σ Modulator
  - Multi-bit
  - Simulation
  - Summary
- Conclusion

Proposed Circuit

Digital ΔΣ modulator + Small analog circuit ⇒ Jitter generator

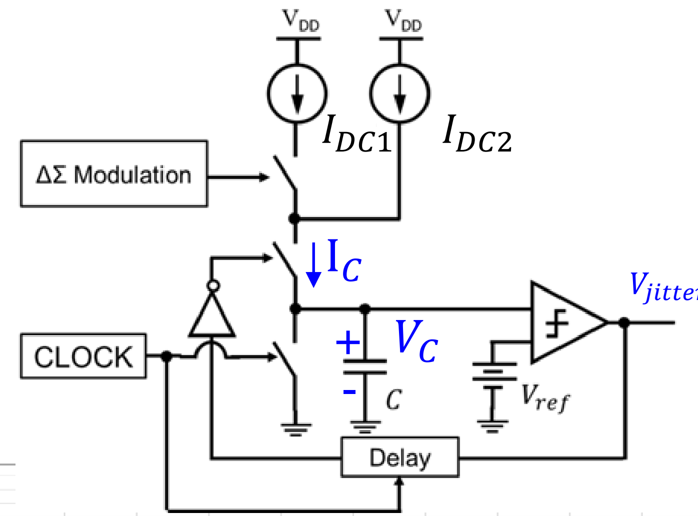
①ΔΣ modulator can be realized by software on ATE
②I_{DC1} is switched by ΔΣ modulator output
③Power reduction by feedback output



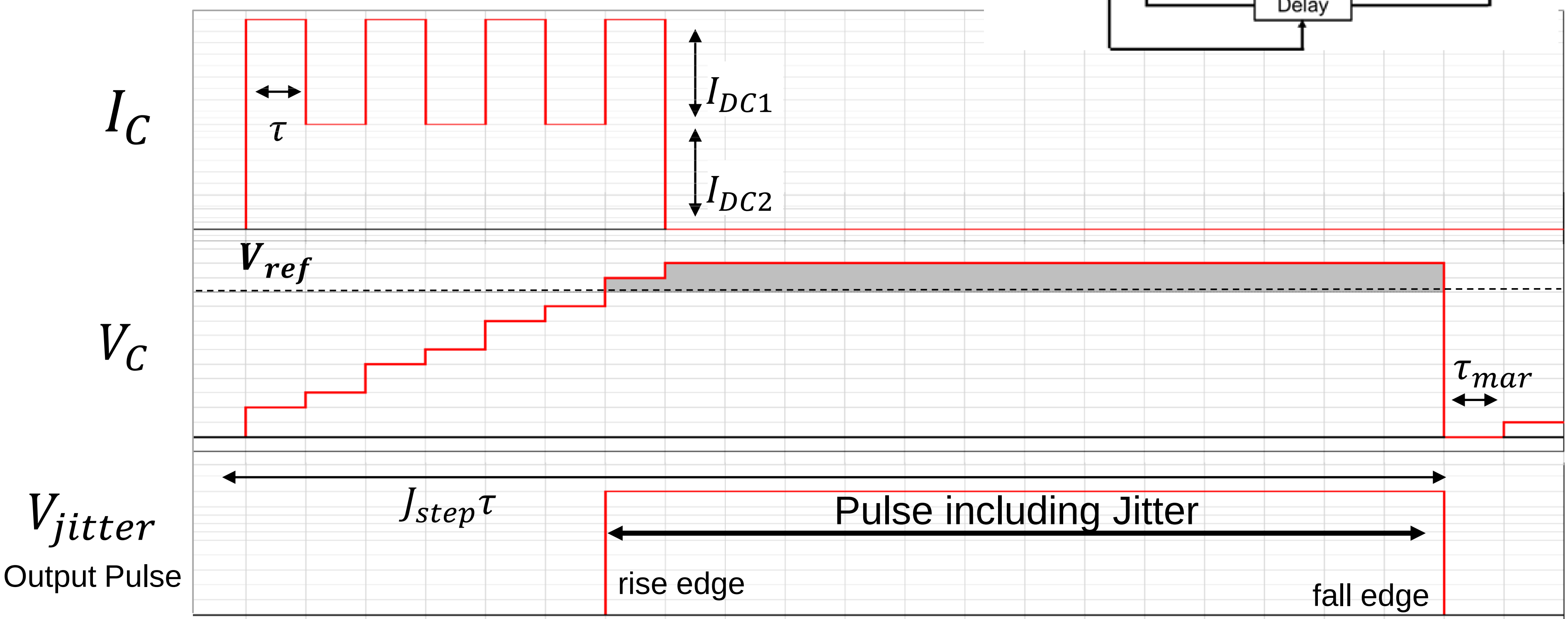
Operation Example

Operation

1. I_{DC1} is switched by ΔΣ modulator output
2. Current accumulation in capacitance
3. Output rises when V_C exceeds V_{ref}
4. Clock resets the capacitor.



1 cycle of pulse including jitter



Jitter Equation

Jitter deviation range

$$W_{pulse\_ref} \leq jitter \leq W_{pulse\_max}$$

$$\left(J_{step}\tau - \frac{CV_{ref}}{I_{DC2}}\right) \leq jitter \leq \left(J_{step}\tau - \frac{CV_{ref}}{(I_{DC1}+I_{DC2})}\right)$$

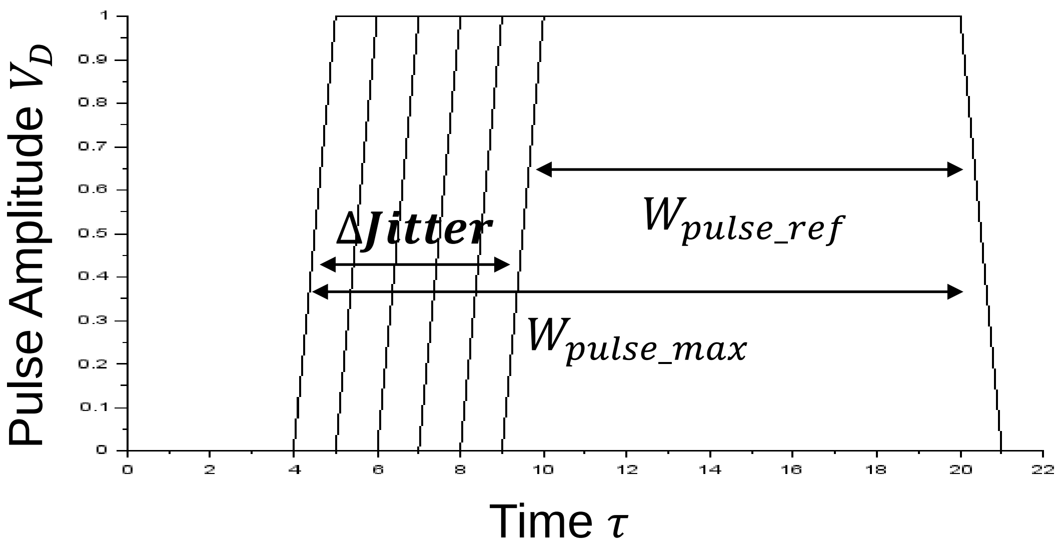
$$\Delta Jitter = \frac{I_{DC1}}{I_{DC2}}\left(\frac{C}{(I_{DC1}+I_{DC2})}\right)V_{ref}$$

Jitter resolution J_{step}

$$J_{step} = f_{sampling}/f_{CLK}$$

Increase jitter resolution････

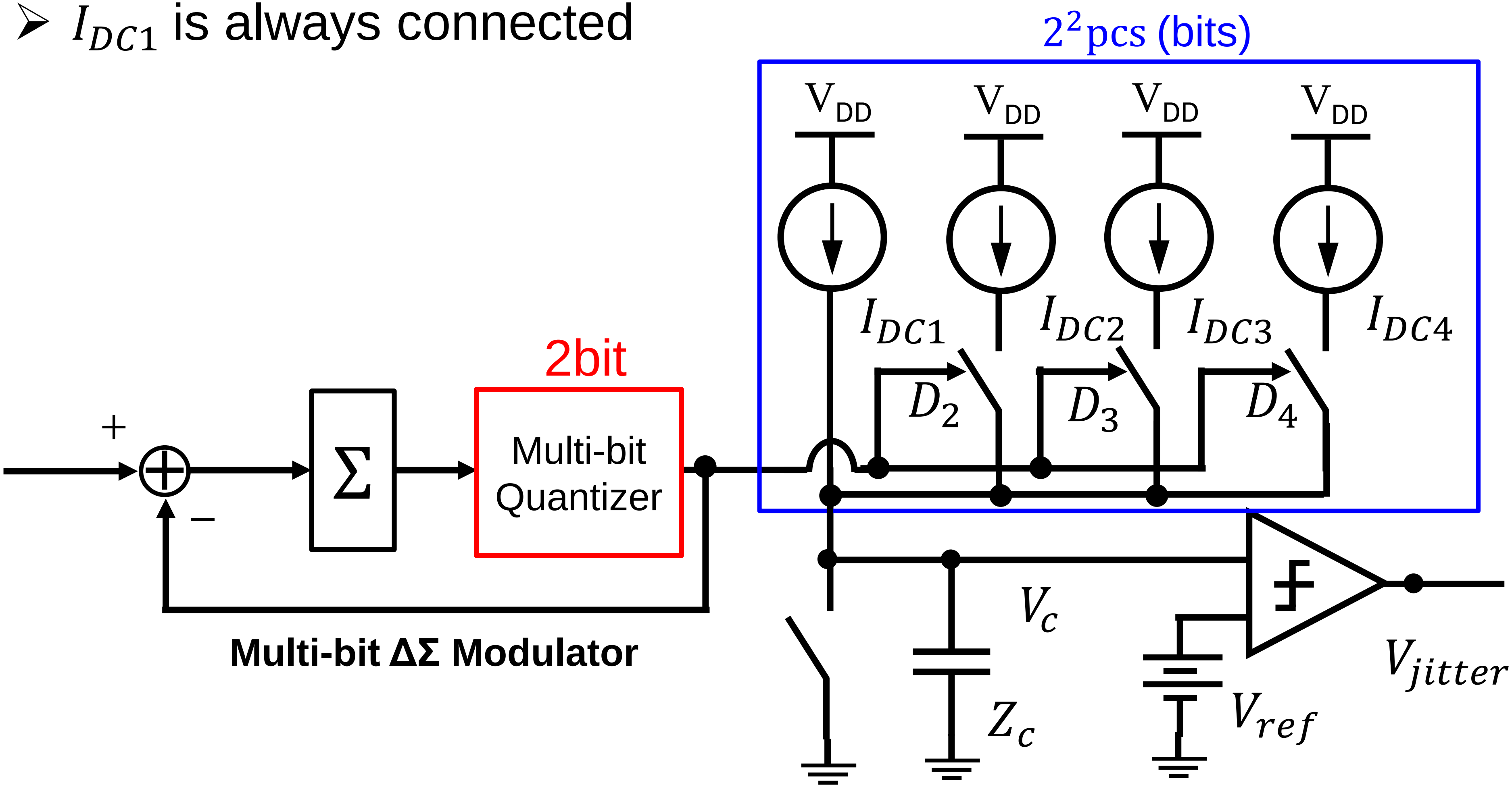
- $f_{sampling}$; UP
- f_{CLK} ; DOWN



Jitter amount of each pulse

Multi-bit ΔΣ Jitter Generator

- Changing single-bit quantizer to multi-bit
- Each current value is constant.
- I_{DC1} is always connected

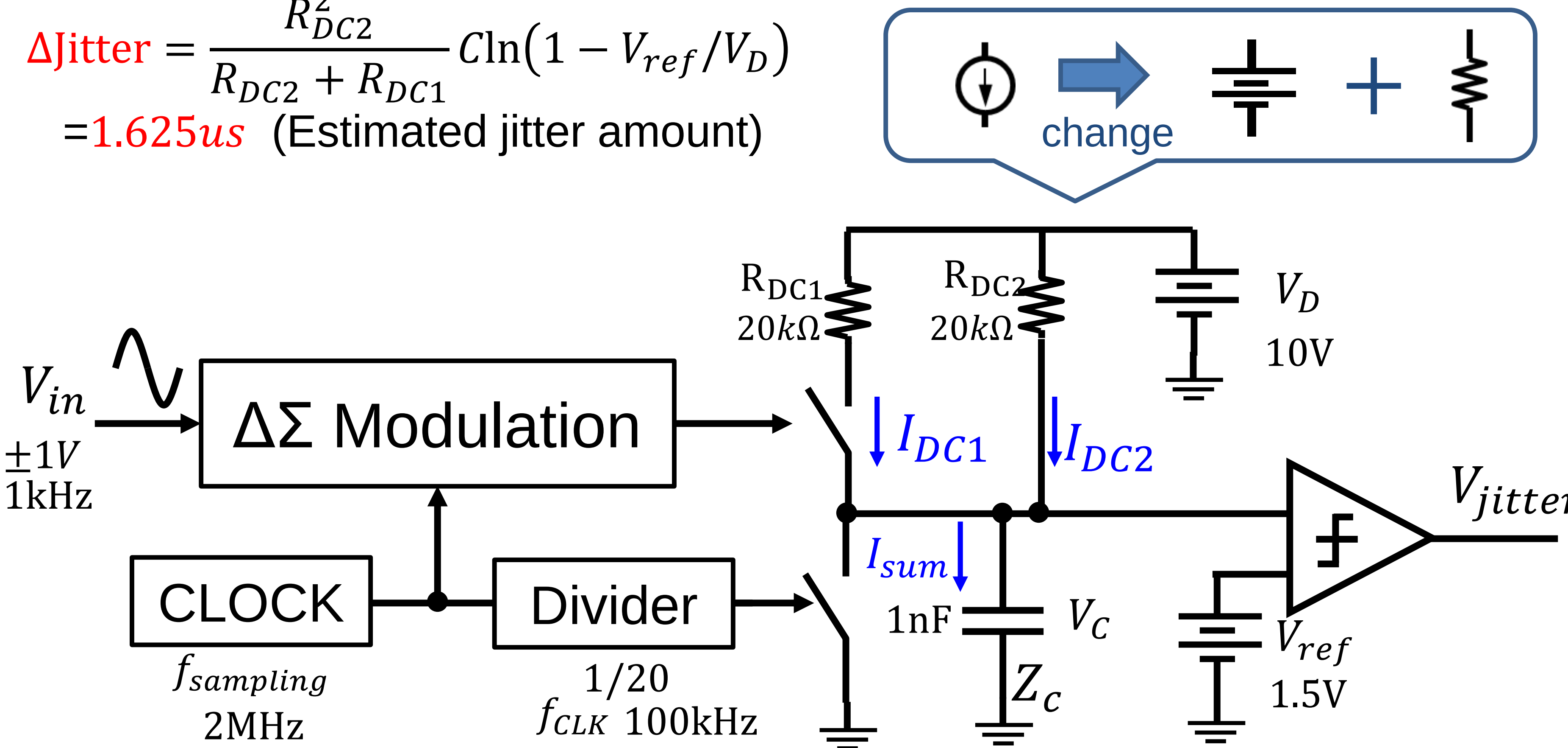


Circuit Simulation Condition

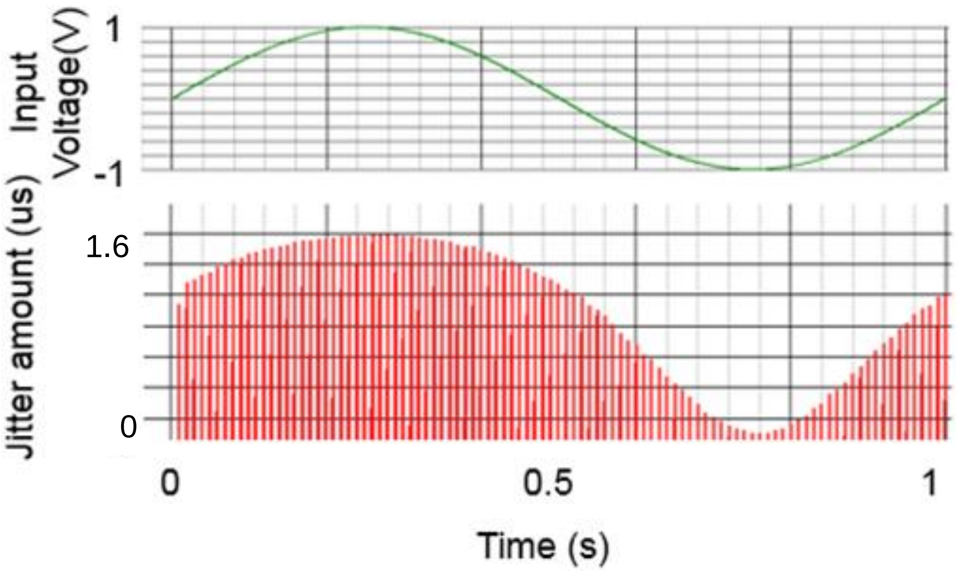
Simulation software : **SIMetrix**

$$\Delta \text{Jitter} = \frac{R_{DC2}^2}{R_{DC2} + R_{DC1}} C \ln\left(1 - V_{ref}/V_D\right)$$

$= 1.625us$ (Estimated jitter amount)



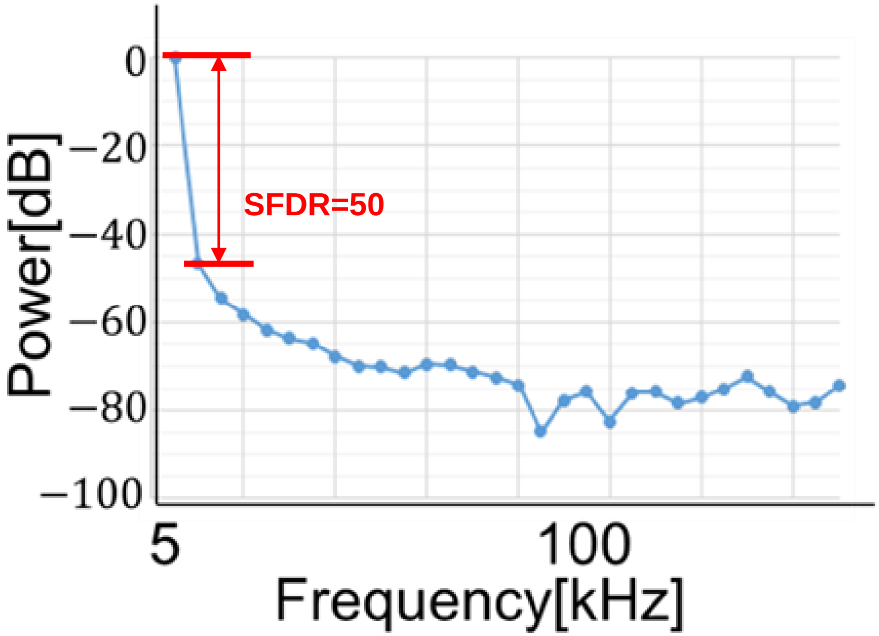
Circuit Simulation Results



Jitter amount superimposed on each pulse

$$\Delta \text{Jitter} = \frac{R_{DC2}^2}{R_{DC2} + R_{DC1}} C \ln\left(1 - V_{ref}/V_D\right)$$

$=1.625us$ (Estimated jitter amount)



Spectrum of jitter amount

Jitter amount 1.6us

⇒Pulse width 6.8us~8.4us

⇒Slightly distorted

SFDR≈50dB

⇒SFDR is improved

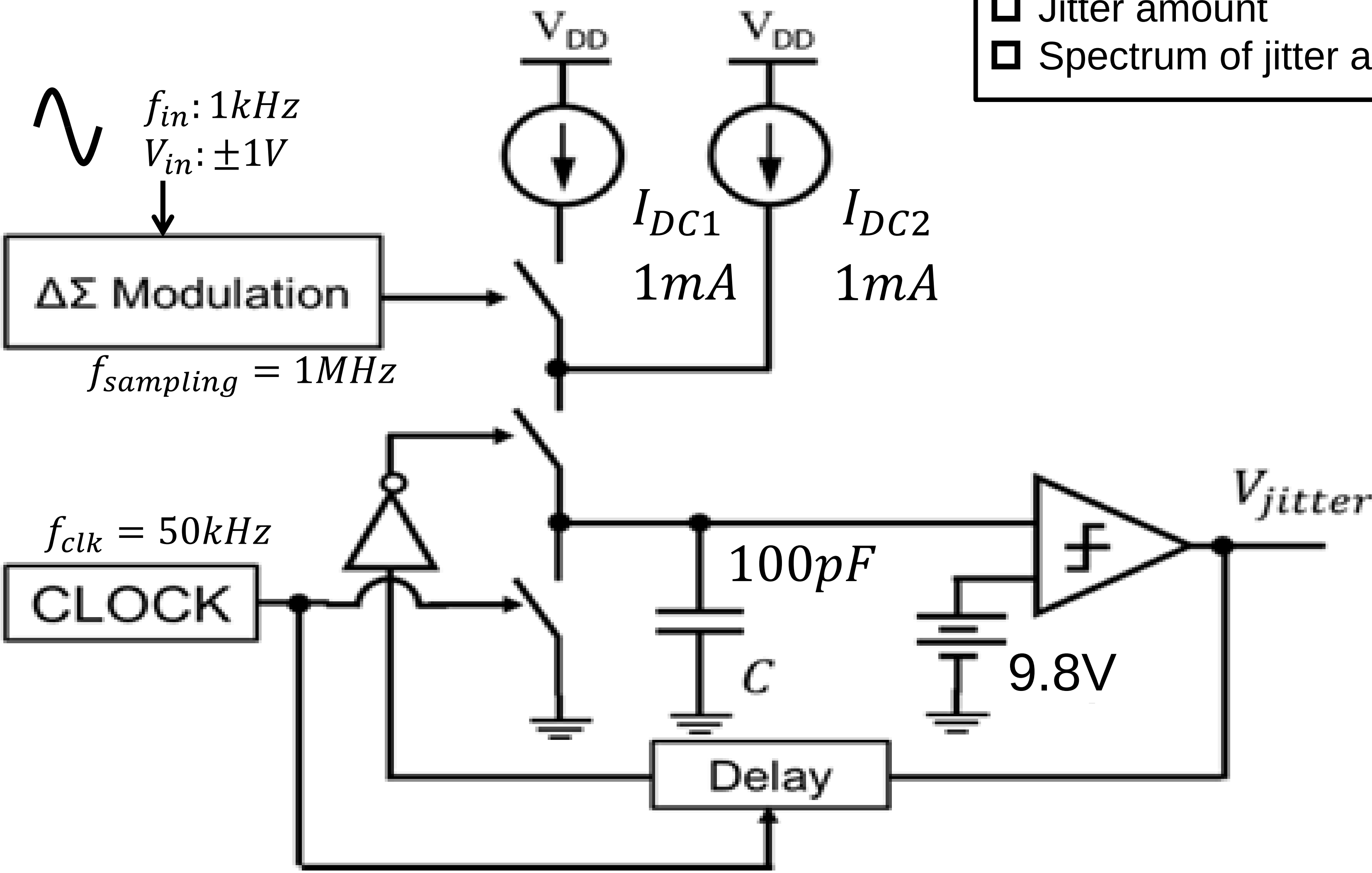
multi-bit conversion or digital filter

Numerical Simulation Condition

Simulation software: **Scilab**

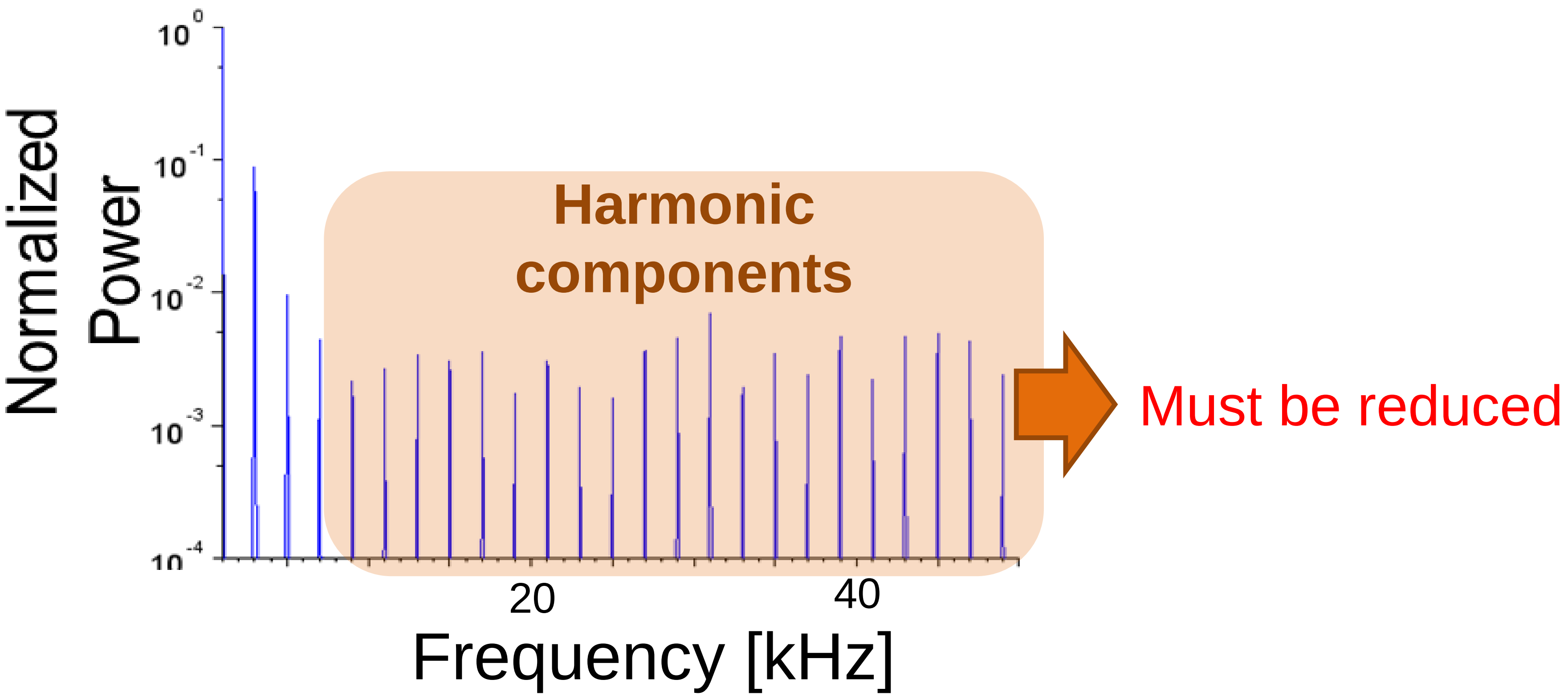
Validation Items

- Generated pulse trains
- Jitter amount
- Spectrum of jitter amount



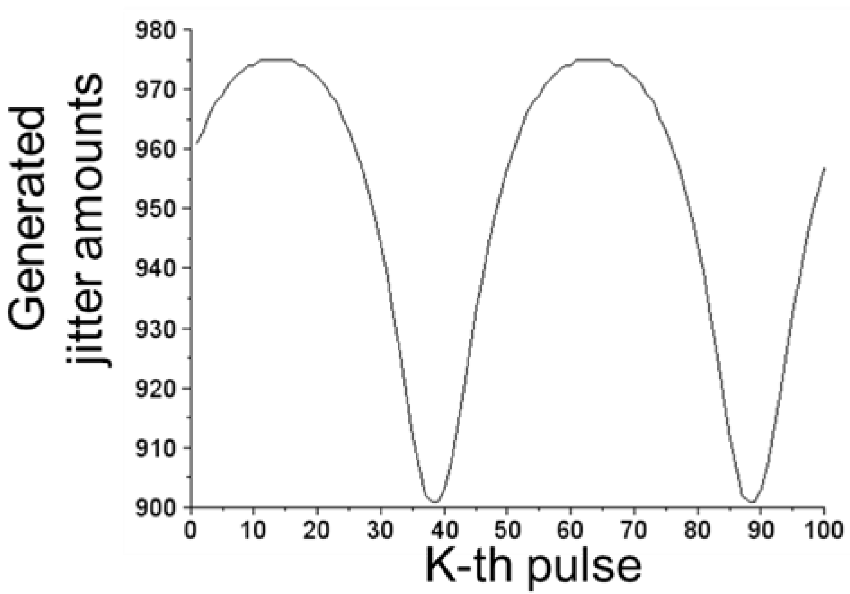
Numerical Simulation Results (Single-bit)

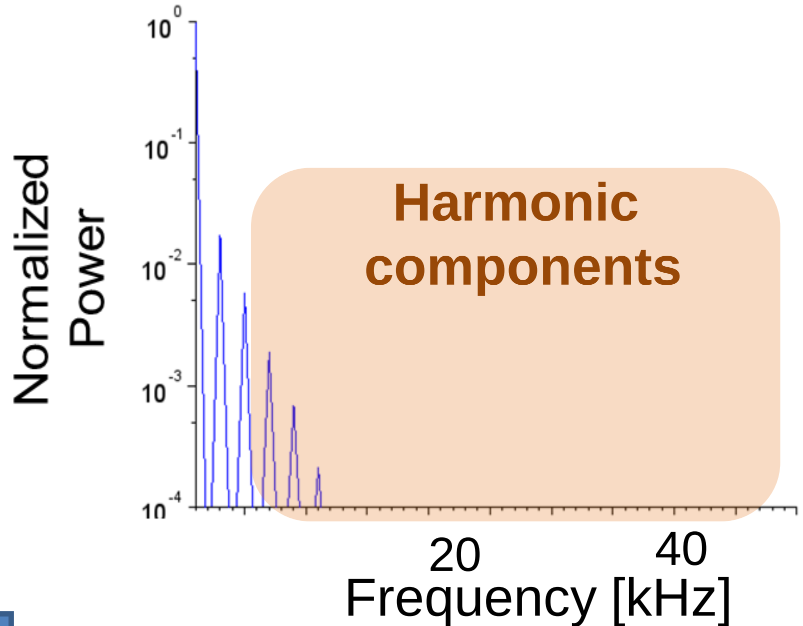
- Spectrum of jitter amount
  - Input frequency : 1kHz



Numerical Simulation Results (Multi-bit)

- **2-bit** ΔΣ jitter modulator





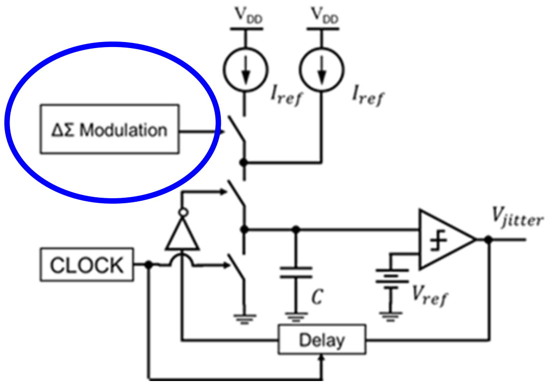
Harmonic components **<u>greatly reduced</u> due to multi-bit**

Summary

Proposed jitter generator Ⅱ
Jitter generator with ⊿Σ modulator

◆ "ΔΣ modulator" using method

Realized by software on ATE



➢ Jitters can be generated with digital circuit & small analog circuit

➢ Circuit operation was confirmed
by numerical simulation / circuit simulation
Scilab SIMetrix

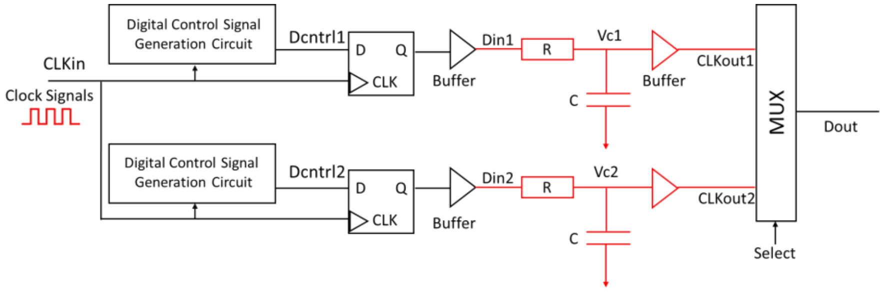
OUTLINE

- Proposed Method Ⅰ
  - Inter-symbol Interference (ISI)
  - Jitter Generator with ISI
  - Simulation
  - Summary
- Proposed Method Ⅱ
  - Jitter Generator with ⊿Σ Modulator
  - Multi-bit
  - Simulation
  - Summary
- Conclusion

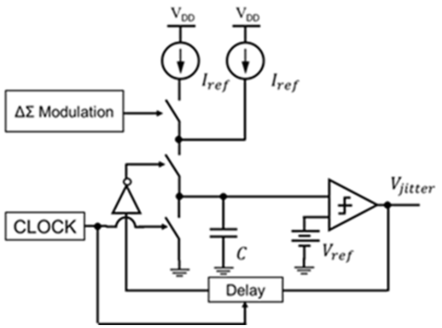
Conclusion

Proposed jitter generation methods

1. Using "inter-symbol interference"



2. Using "ΔΣ modulator"



Mostly digital implementation

Jitter generators become low cost & easy to design

Future works

Implementation & Experiment

Final Statement

Nature prefers fluctuations.

Albert Einstein