

# Fast Fourier Transform

## RTL Project

Team 2. 유재훈, 유승범

# 목차

## Fast Fourier Transform

- 프로젝트 개요
  - 목표, 사용 부품 및 스펙
- Fourier Transform
- DFT, FFT
- 개발과정
  - DFT, FFT
- DSP(FFT) IP
  - Block Diagram, Flow Chart
- 검증
  - 테스트 장비, 테스트 과정
- 프로젝트 결과분석
  - 오류, 문제점, 해결 과정
  - 하드웨어 자원 사용량
  - 작품 활용 방안
  - 프로젝트를 통해 얻은 지식

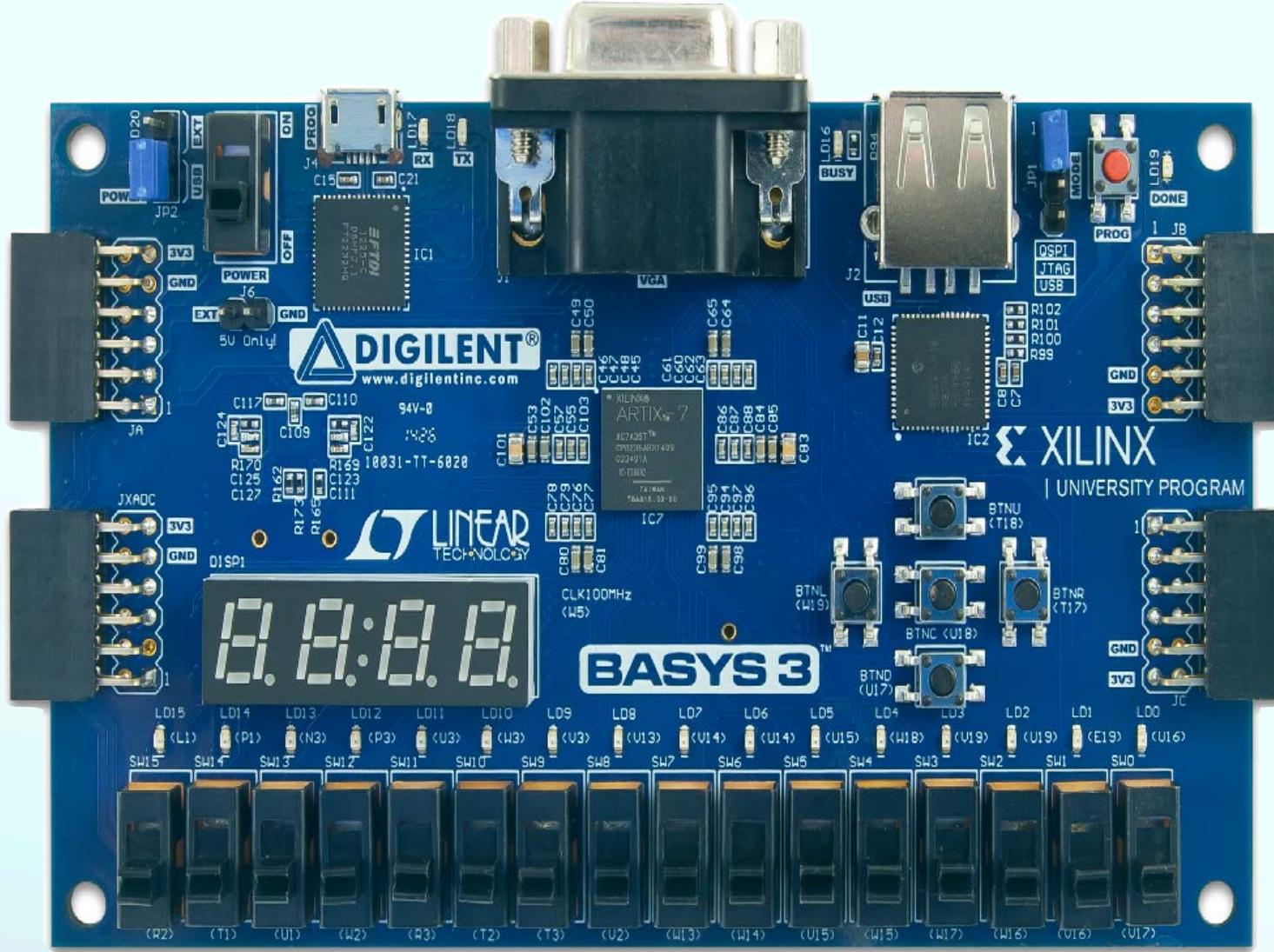
# 개발 목표

Fast Fourier transform

- FFT 연산 전용 디지털 신호 처리기(DSP) 개발
  - 16 Point Sample
  - 16bit input resolution
- STM32 를 이용한 Test rig 개발
  - 다양한 샘플 주파수 생성
- FFT DSP 검증
  - Test bench 를 이용한 검증
  - Test rig를 이용한 검증
- 다양한 사용처 탐구



# 사용 부품 및 스펙



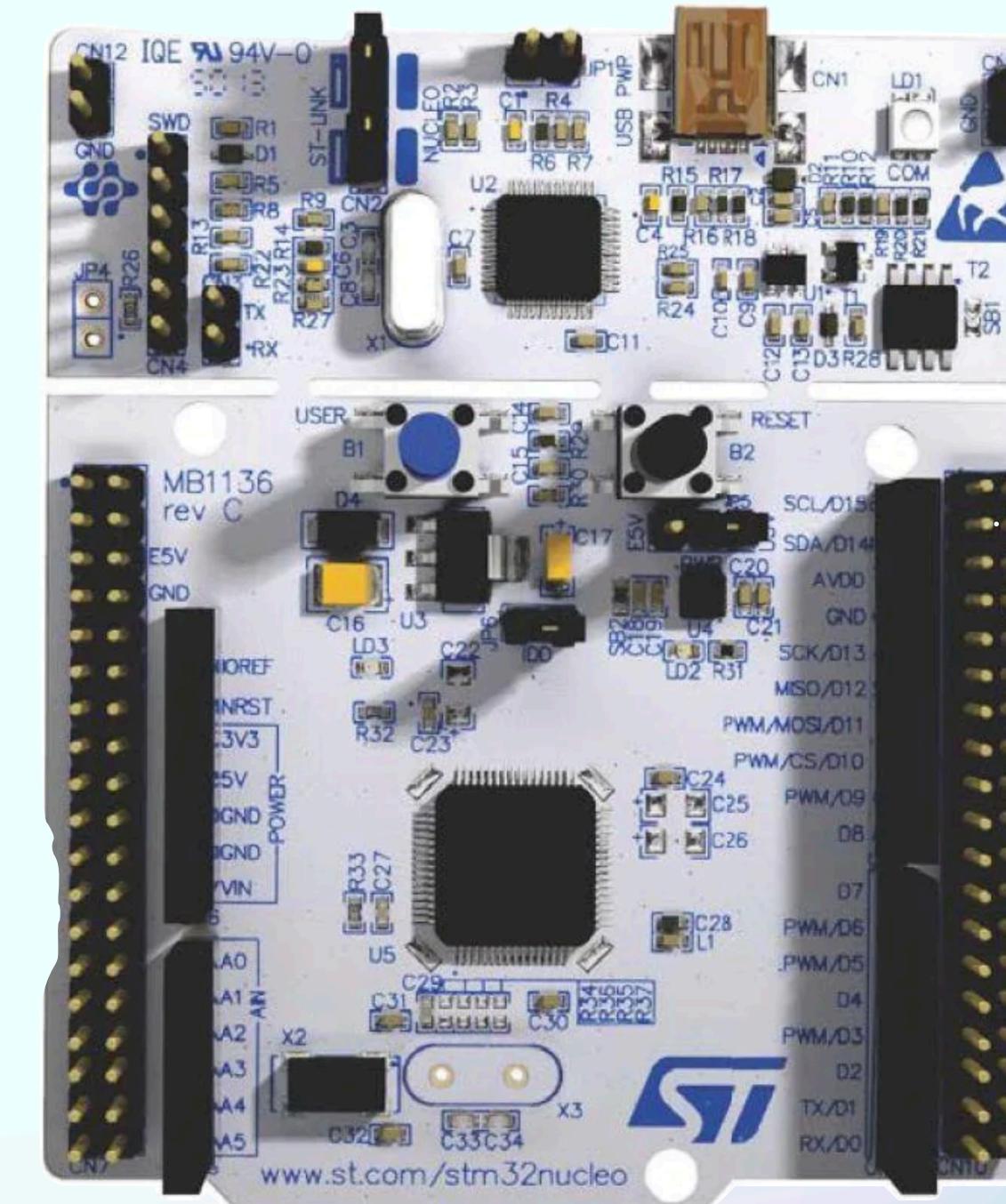
XC7A35T

Logic Cell : 33800

BRAM : 1800Kb

DSP : 90ea

Basys3



STM32 - F411

ARM Cortex M4

3단계 파이프라인

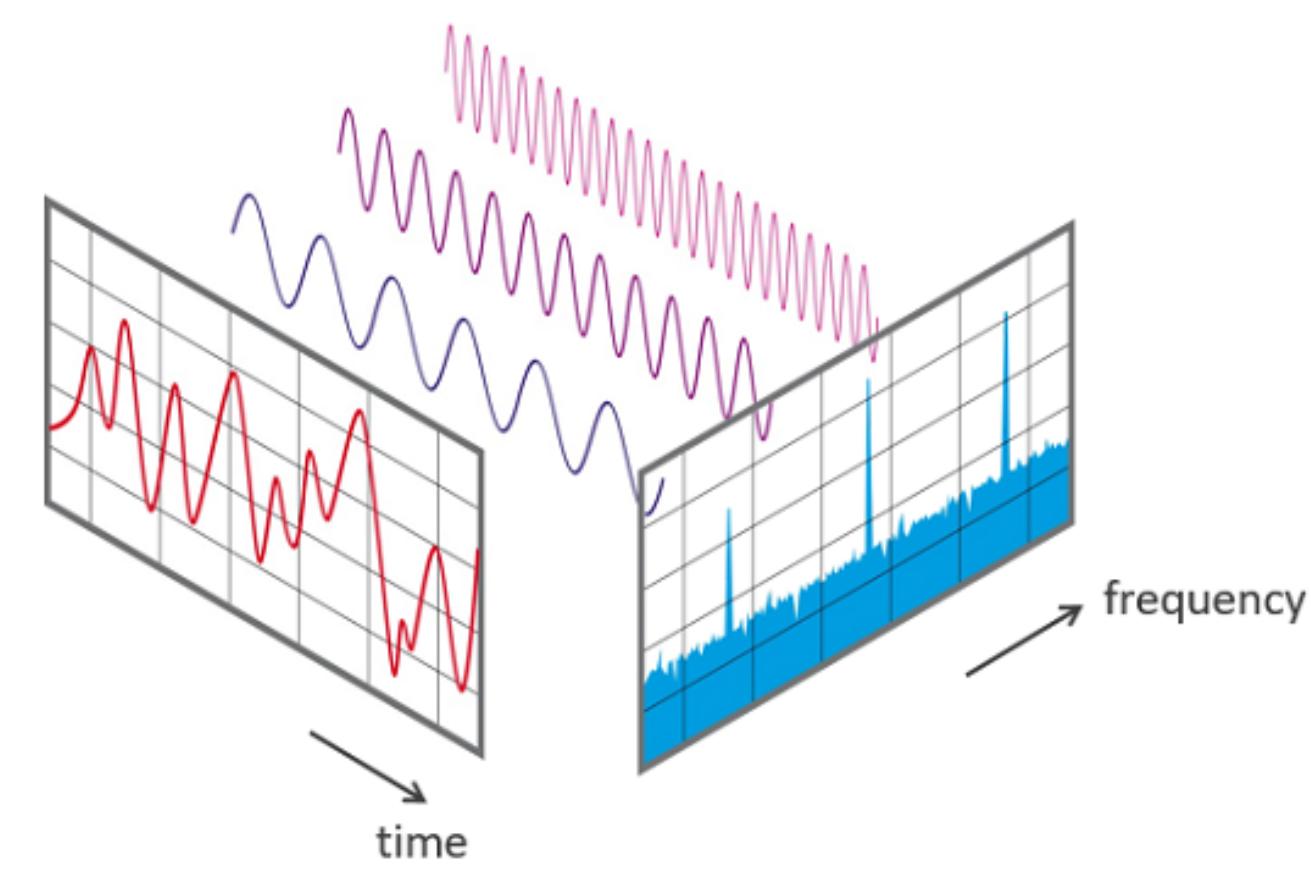
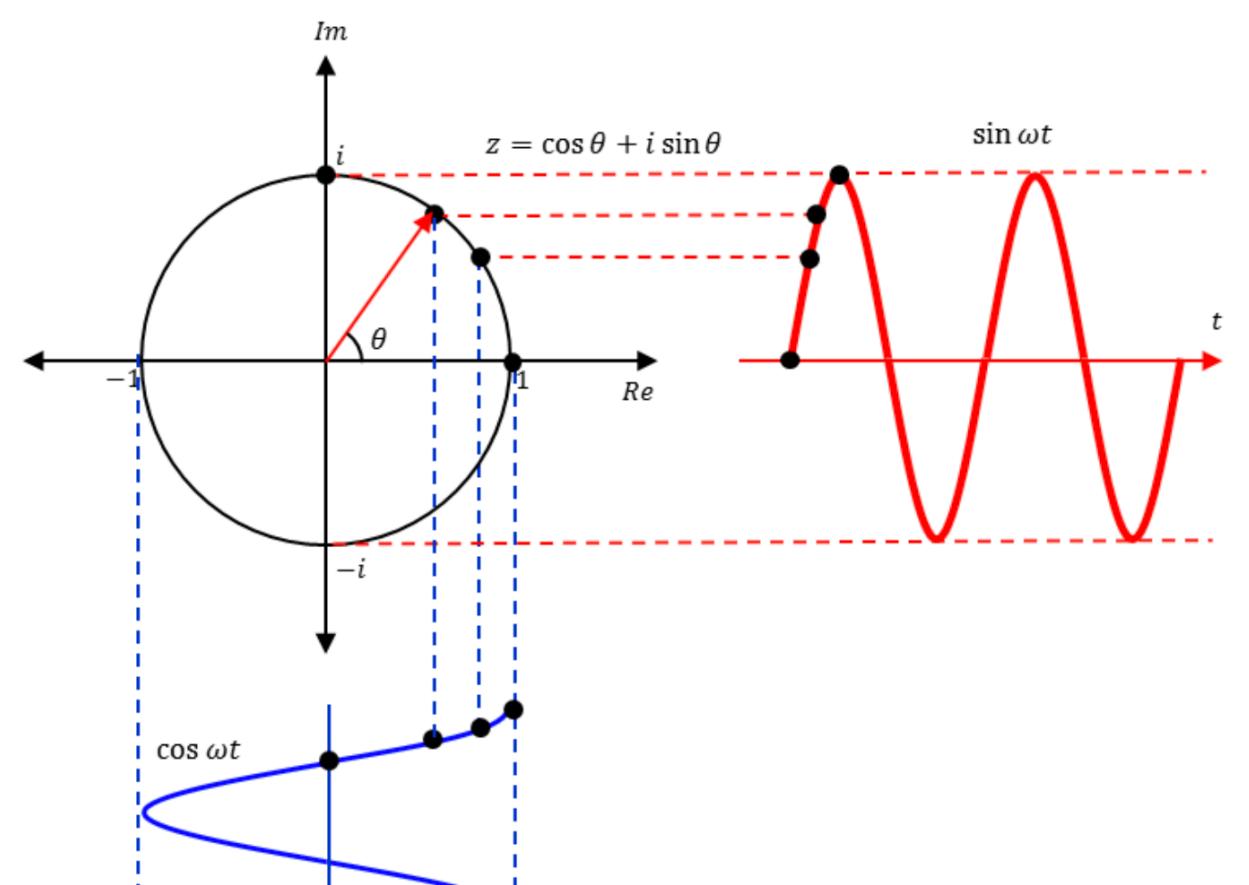
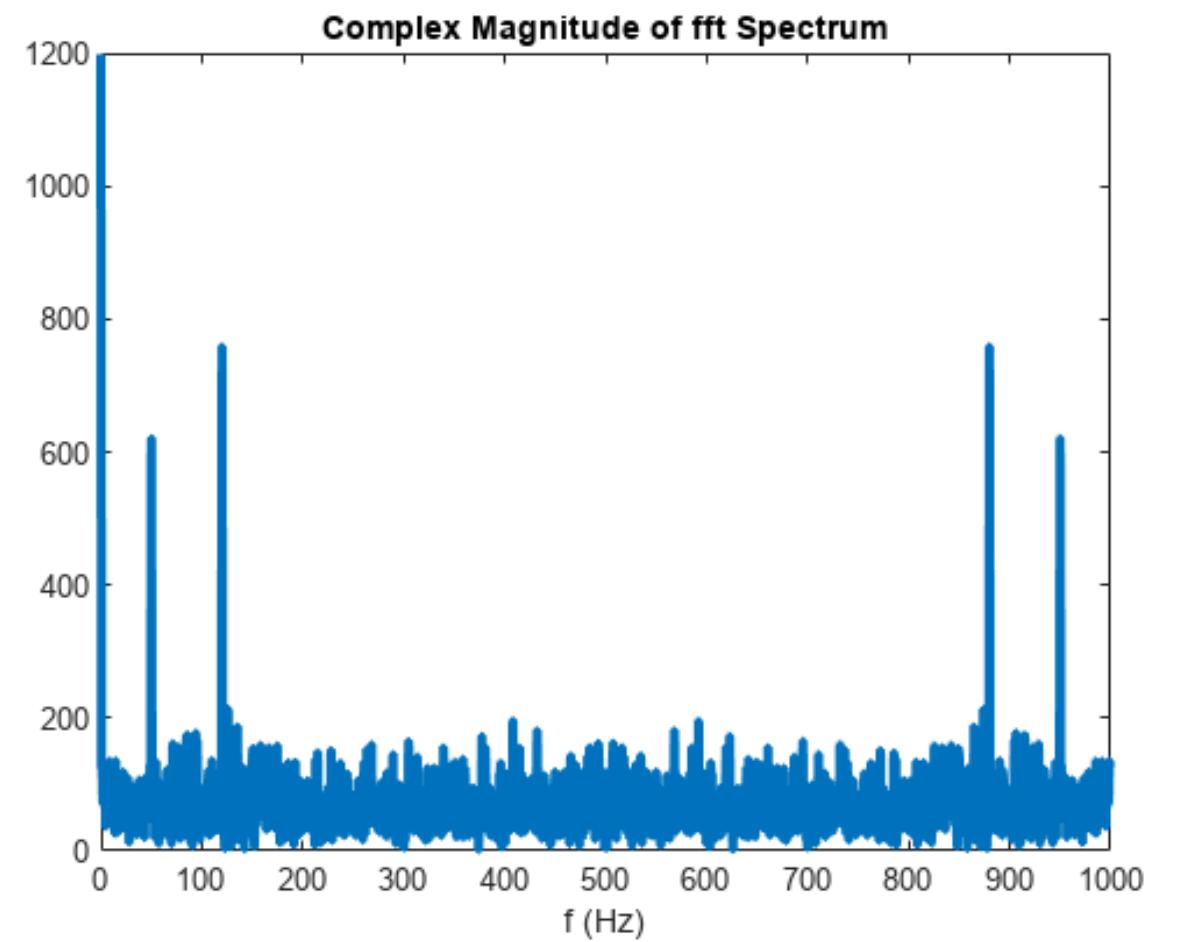
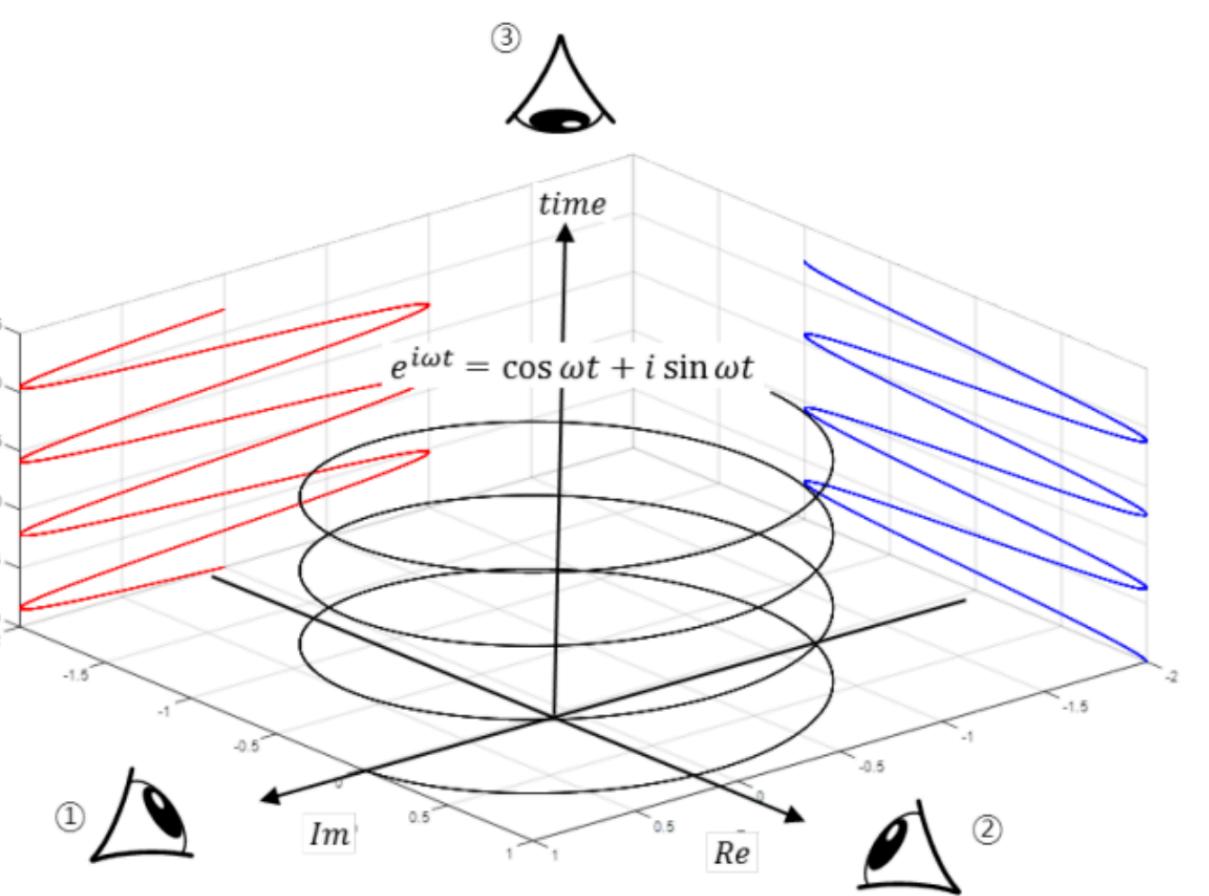
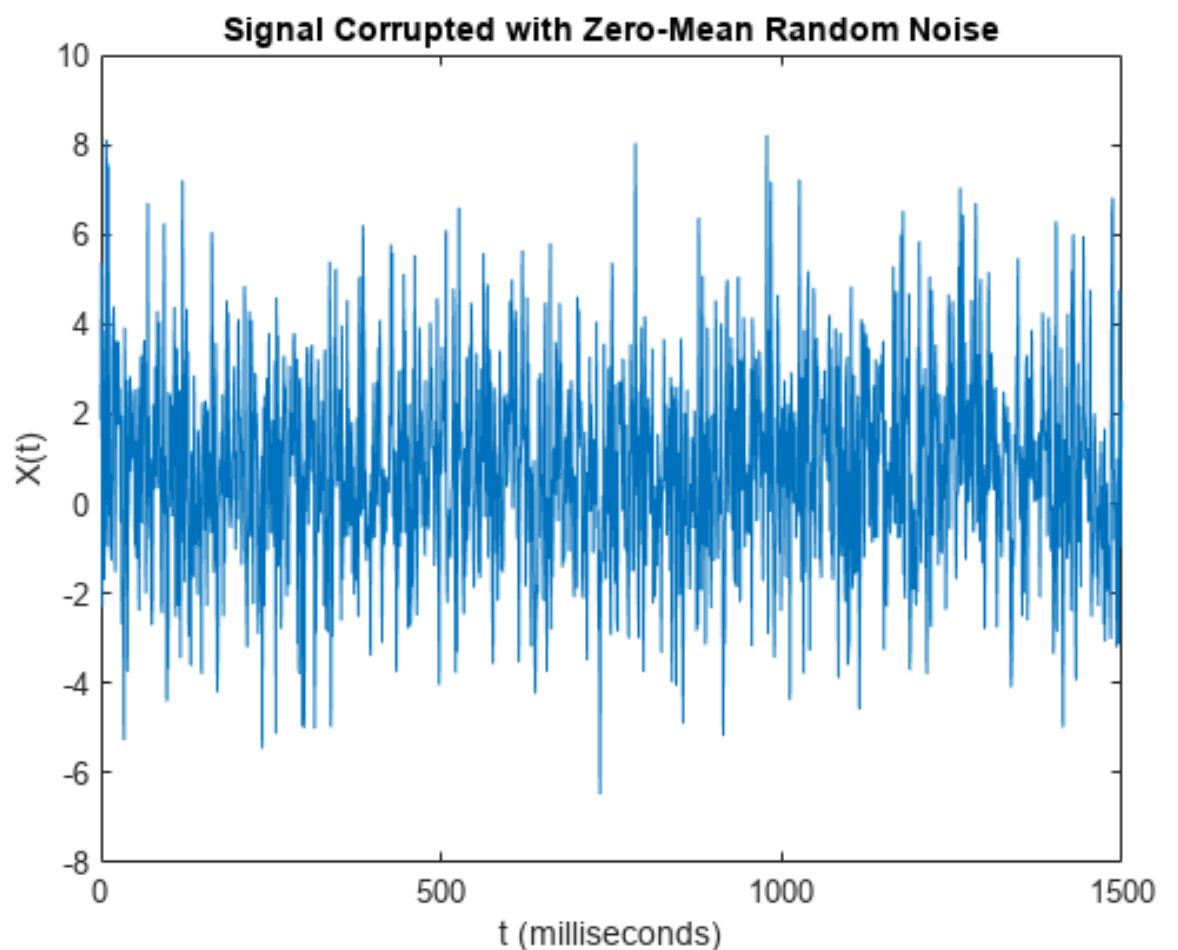
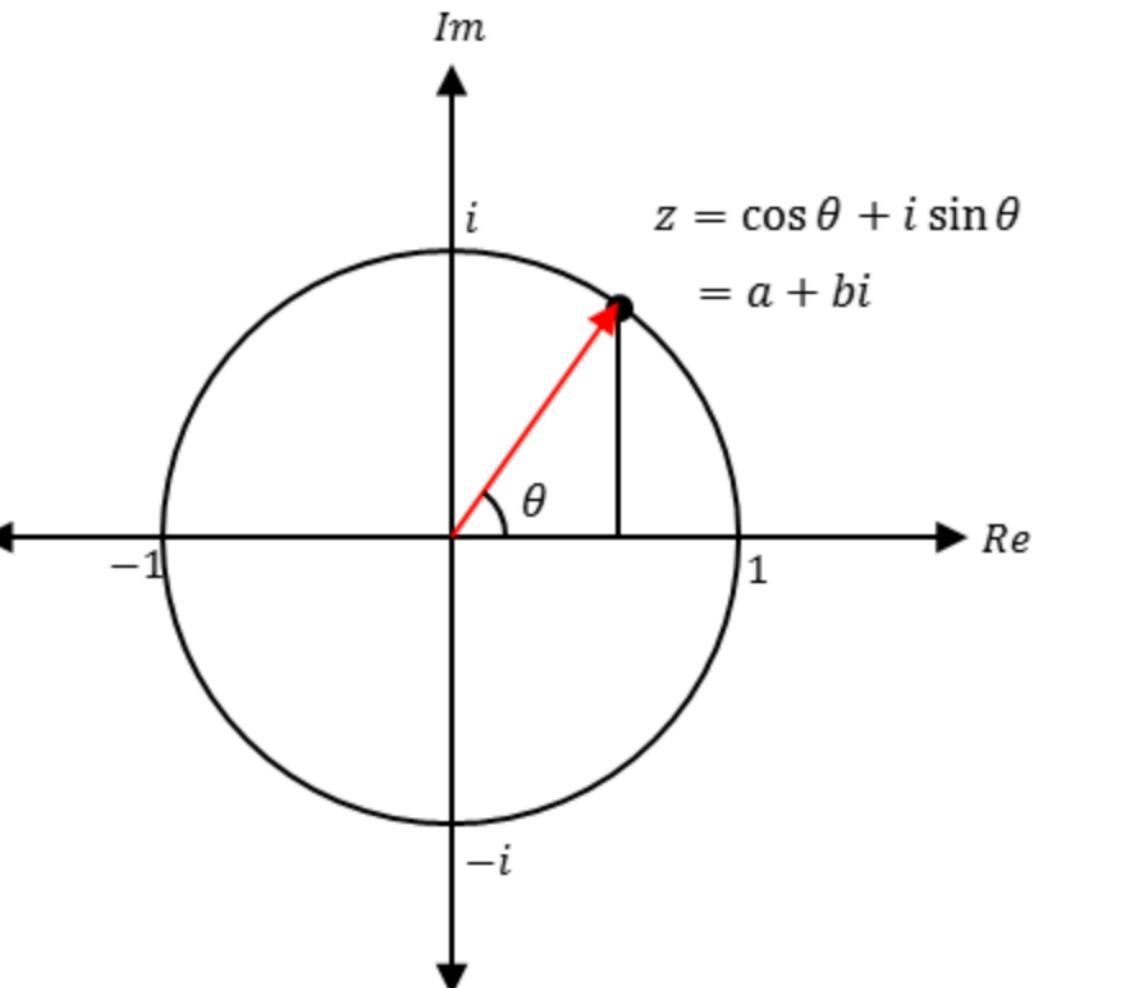
Thumb, Thumb2 명령어

3x AMBA AHB-lite

# Project overview

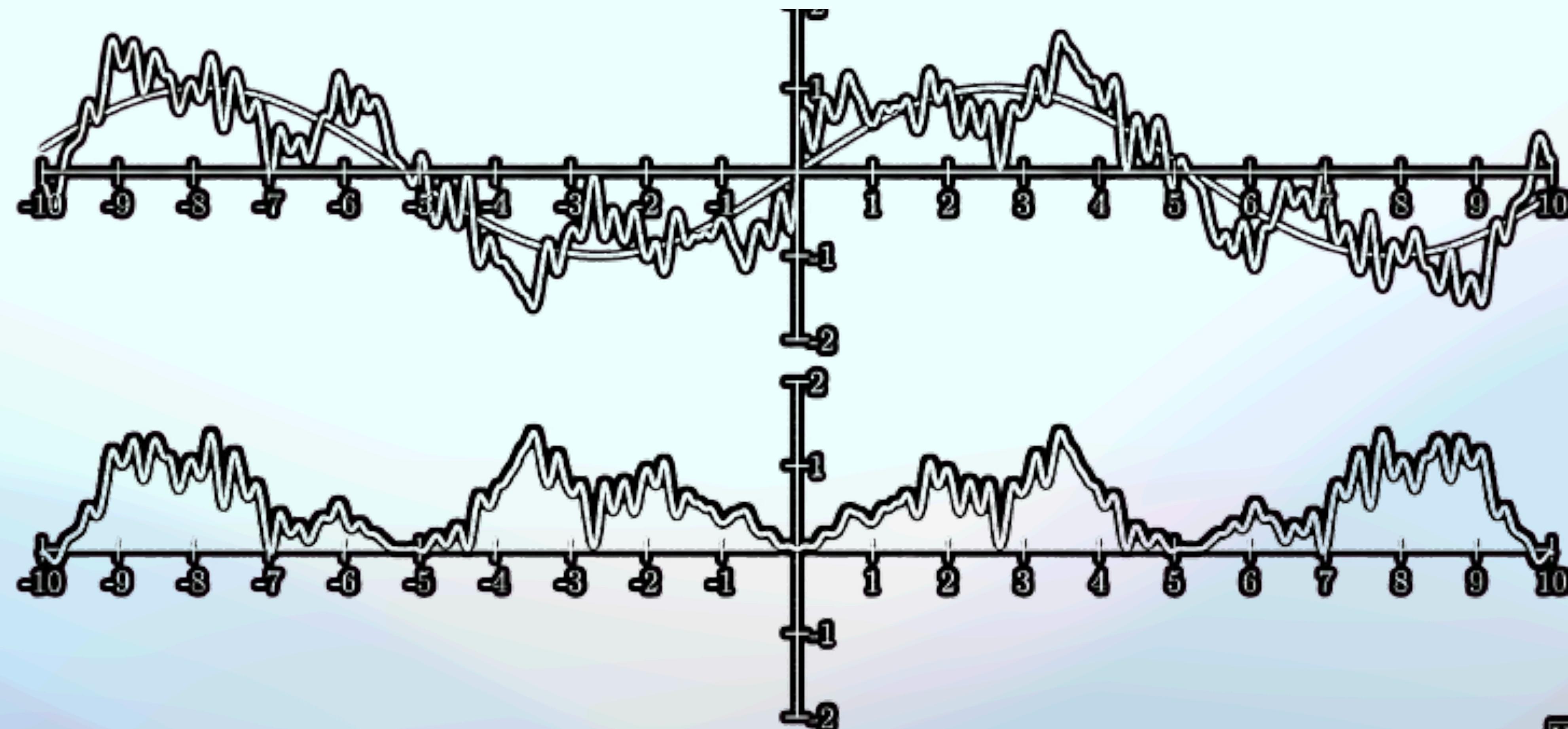
## Fourier Transform

- 시간 영역 -> 주파수 영역
  - 오일러 공식 이용
- 다양한 주파수가 혼합된 신호를 각각의 주파수 성분으로 분류
- 각각의 주파수를 실수(Cos) 허수(Sin)로 분류
- Test bench를 통한 검증
- Test rig를 이용한 검증(STM32, SoC 활용)



# Discrete Fourier Transform

Example



# Discrete Fourier Transform

## Algorithm

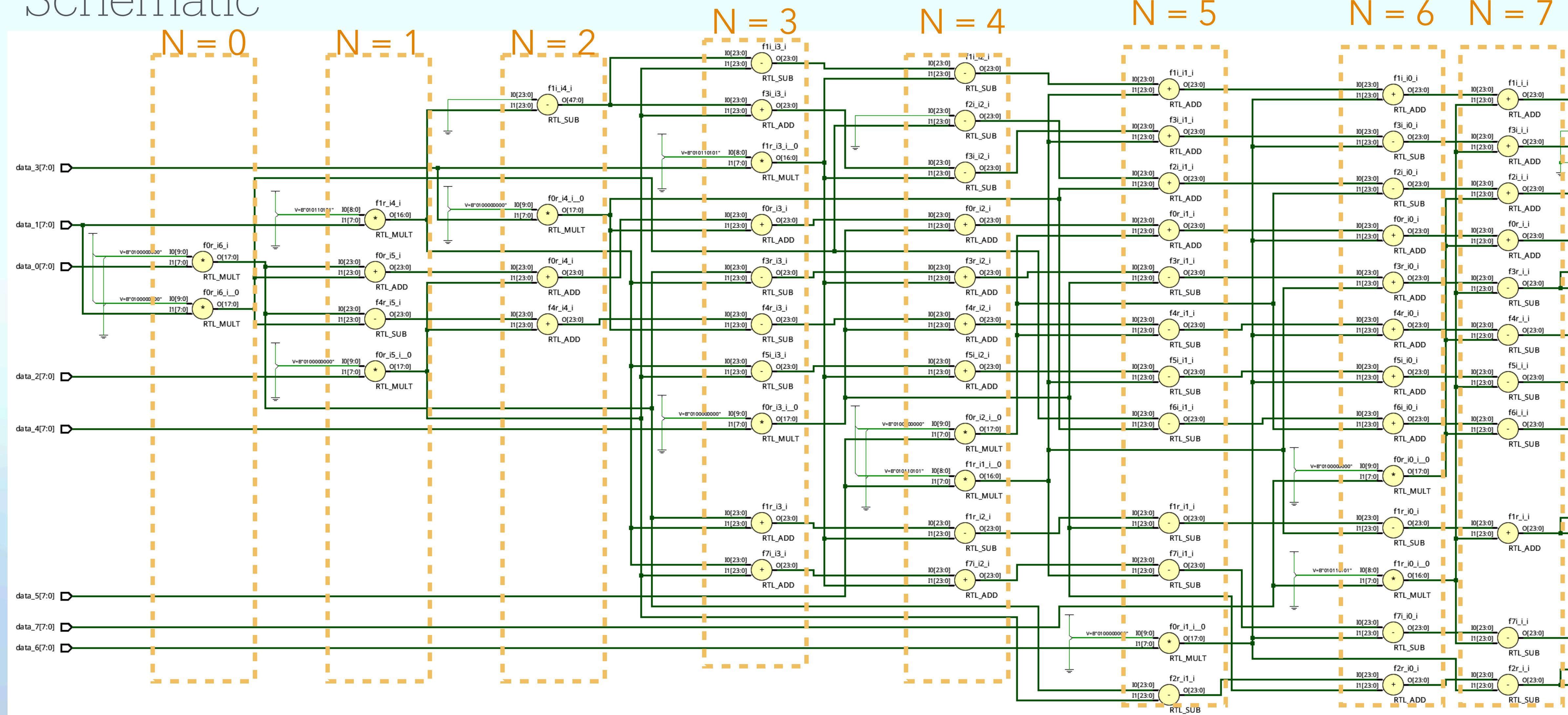
$$W_N = e^{-2\pi i / N}.$$

$$\begin{bmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \\ \vdots \\ \vdots \\ f_{N-1} \end{bmatrix} = \begin{bmatrix} W^0 & W^0 & W^0 & \cdots & W^0 \\ W^0 & W^1 & W^2 & \cdots & W^{N-1} \\ W^0 & W^2 & W^4 & \cdots & W^{2(N-1)} \\ W^0 & W^3 & W^6 & \cdots & W^{3(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ W^0 & W^{N-1} & W^{2(N-1)} & \ddots & W^{(N-1)^2} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ \vdots \\ \vdots \\ x_{N-1} \end{bmatrix}$$

시간 복잡도 :  $O(N^2)$

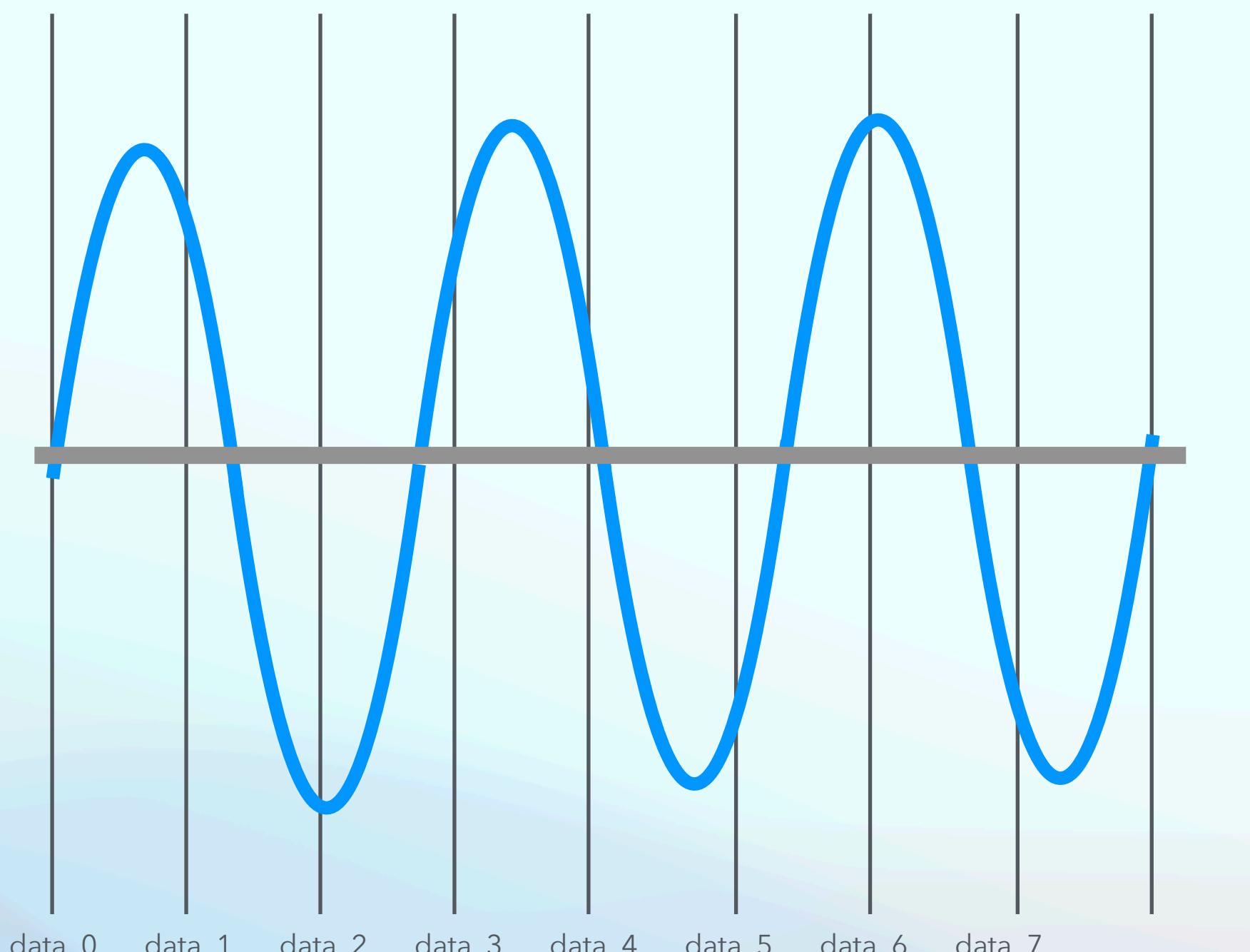
# Discrete Fourier Transform

## Schematic



# Discrete Fourier Transform

## 8point Simulation



Input Data (Sin 3x //8 point)

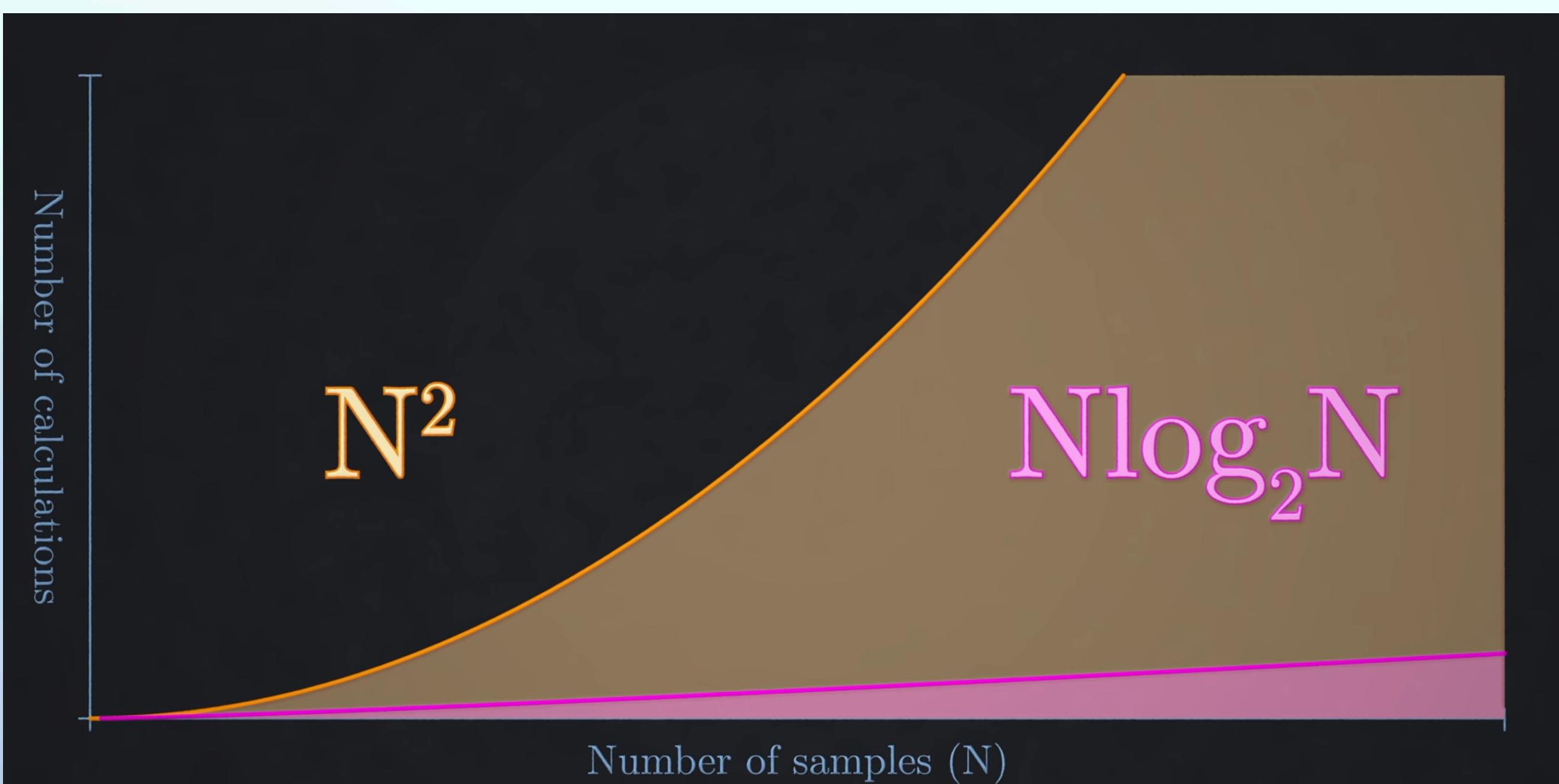
- data\_0 : 0
- data\_1 : 90
- data\_2 : -127
- data\_3 : 90
- data\_4 : 0
- data\_5 : -90
- data\_6 : 127
- data\_7 : -90

> data_0[7:0]	0	0
> data_1[7:0]	90	90
> data_2[7:0]	-127	-127
> data_3[7:0]	90	90
> data_4[7:0]	0	0
> data_5[7:0]	-90	-90
> data_6[7:0]	127	127
> data_7[7:0]	-90	-90
> o_f0_r[15:0]	0	0
> o_f0_i[15:0]	0	0
> o_f1_r[15:0]	0	0
> o_f1_i[15:0]	-1	-1
> o_f2_r[15:0]	0	0
> o_f2_i[15:0]	0	0
> o_f3_r[15:0]	0	0
> o_f3_i[15:0]	-509	-509
> o_f4_r[15:0]	0	0
> o_f4_i[15:0]	0	0
> o_f5_r[15:0]	0	0
> o_f5_i[15:0]	508	508
> o_f6_r[15:0]	0	0
> o_f6_i[15:0]	0	0
> o_f7_r[15:0]	0	0
> o_f7_i[15:0]	0	0

# Fast Fourier Transform

## Algorithm

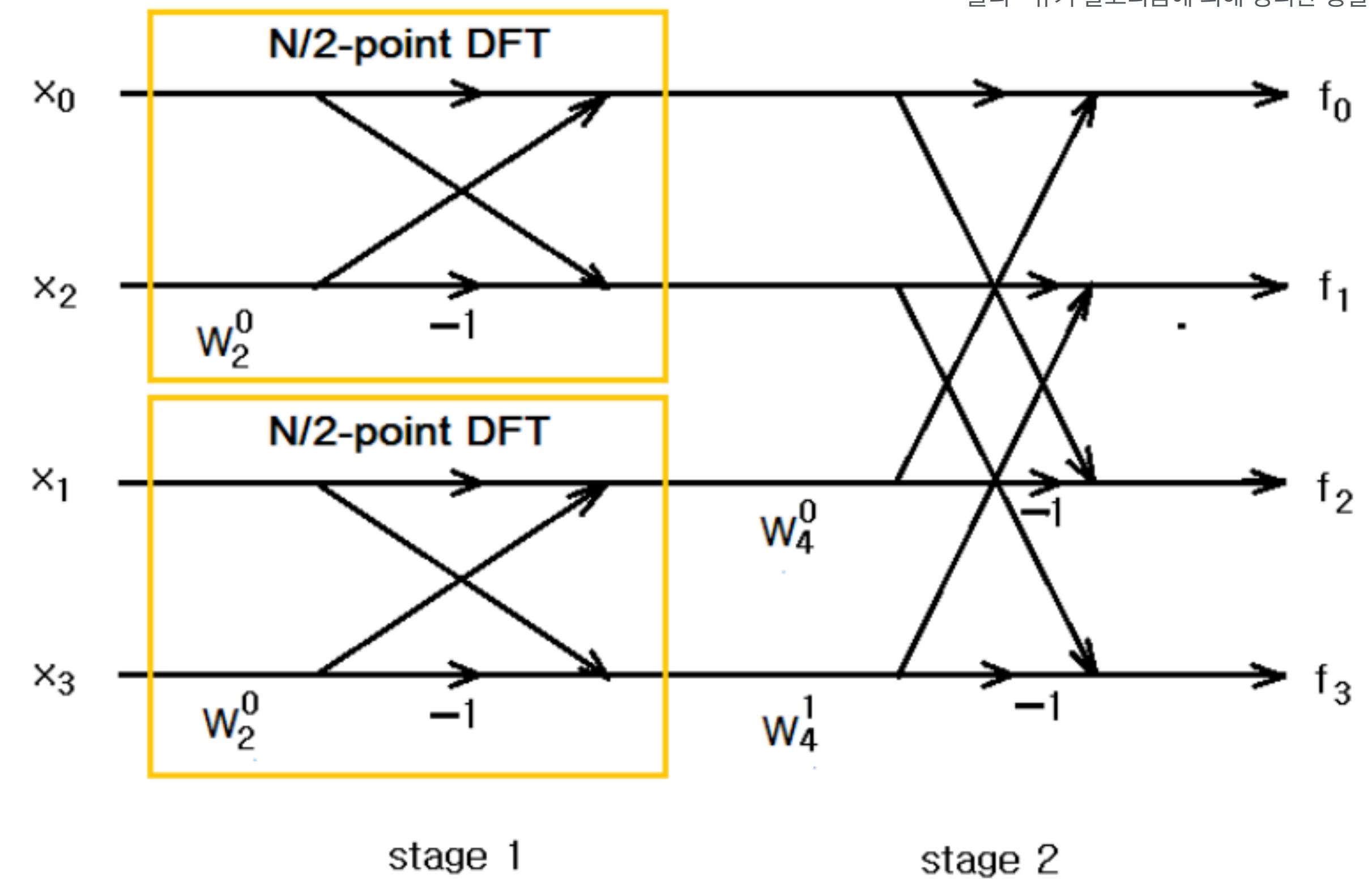
- DFT 연산에서 중복 연산 과정을 재활용
  - 쿠리 - 튜키 알고리즘 이용
  - 시간 복잡도  $N \log_2 N$ 로 축소
- DFT(N Stage)에서 FFT( $\log_2 N$  Stage)로 축소



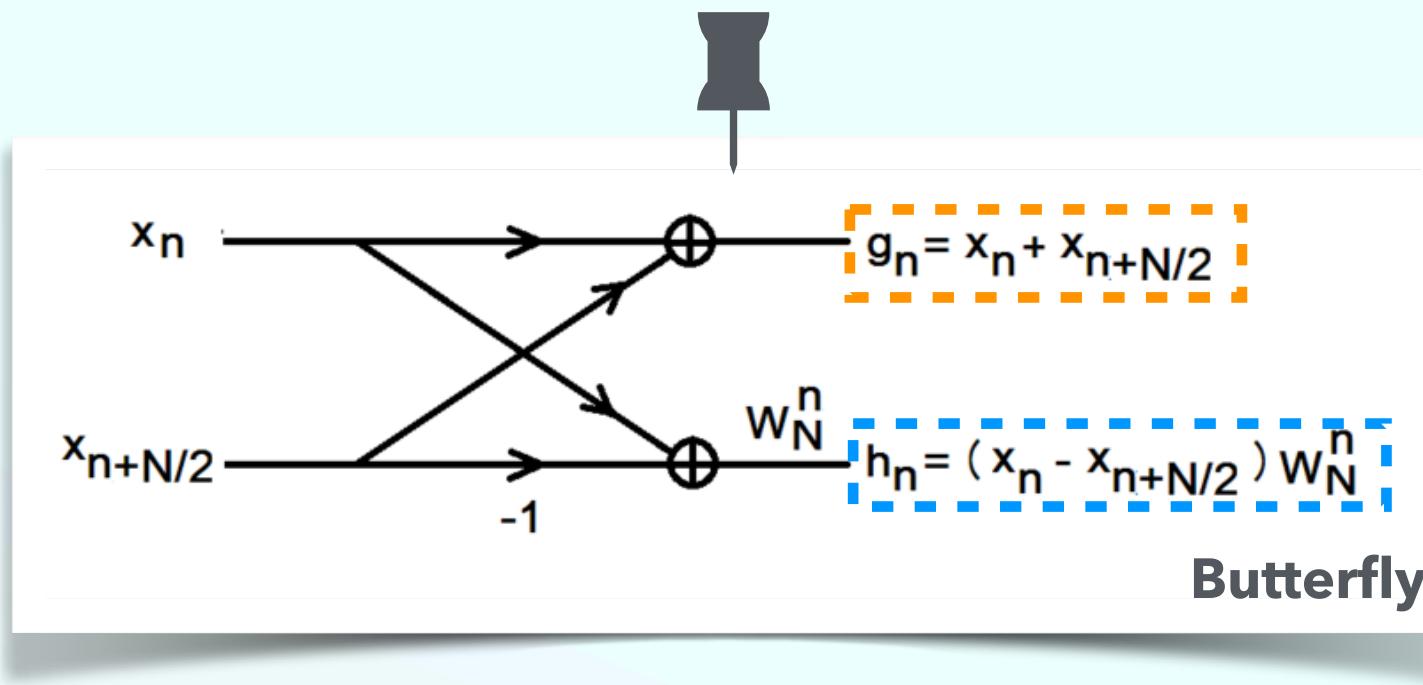
Butterfly Diagram

$$\begin{bmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & W_4^0 & 0 \\ 0 & 1 & 0 & W_4^1 \\ 1 & 0 & -W_4^0 & 0 \\ 0 & 1 & 0 & -W_4^1 \end{bmatrix} \begin{bmatrix} W_2^0 & W_2^0 \\ W_2^0 & -W_2^0 \\ W_2^0 & W_2^0 \\ W_2^0 & -W_2^0 \end{bmatrix} \begin{bmatrix} x_0 \\ x_2 \\ x_1 \\ x_3 \end{bmatrix}$$

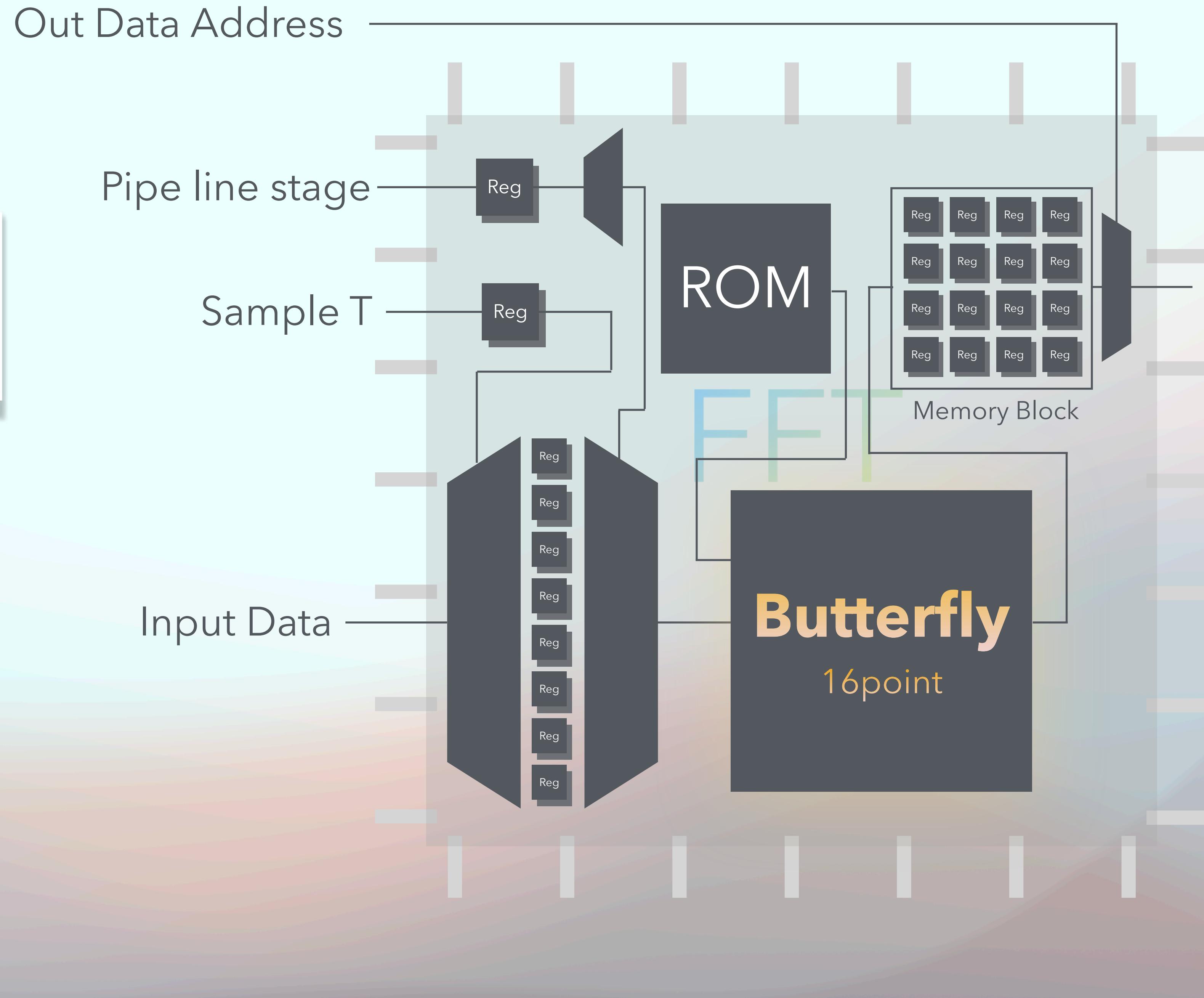
쿠리 - 튜키 알고리즘에 의해 정리한 행렬



# FFT algorithm



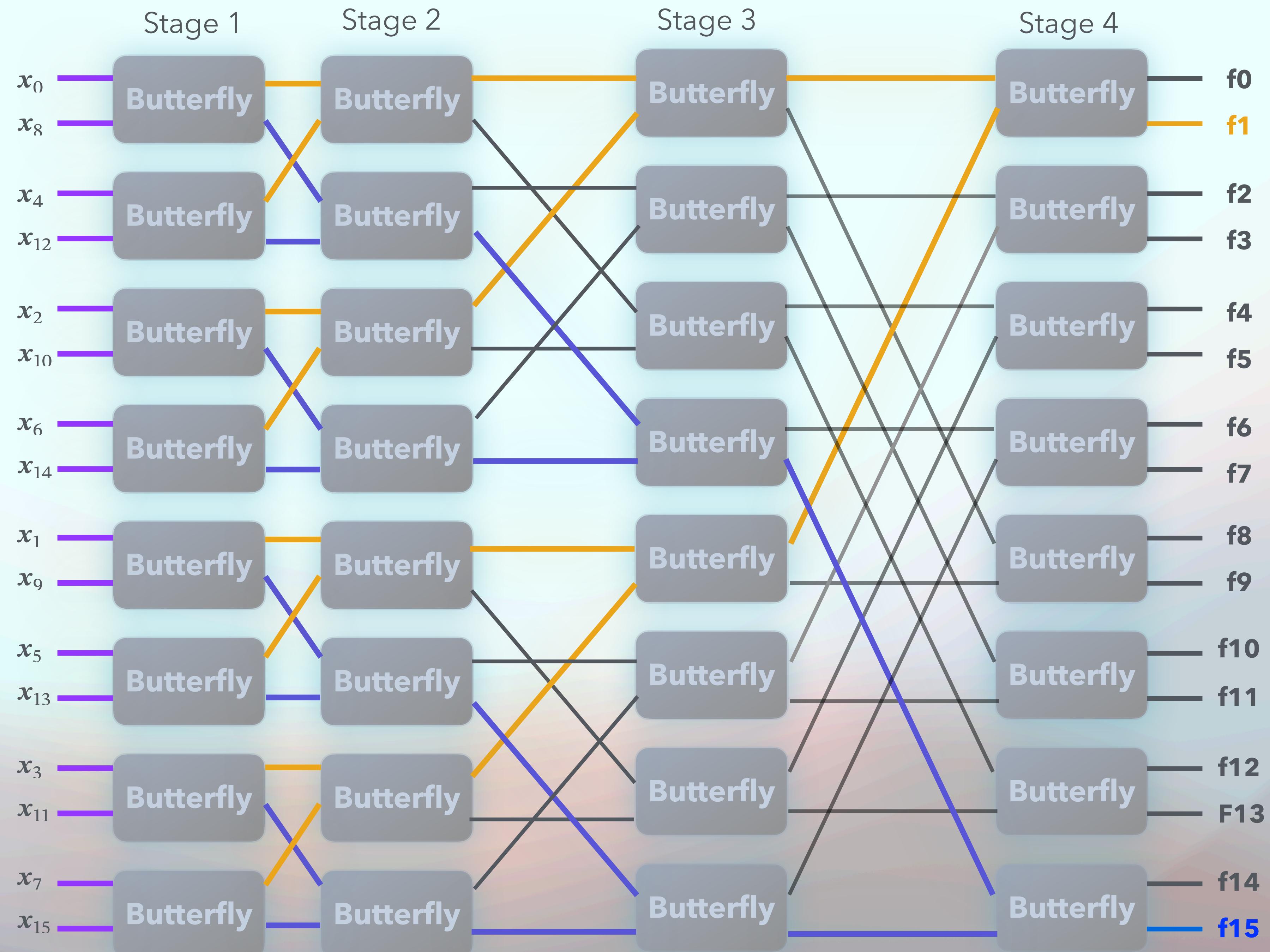
$$\begin{aligned}
 X_{n+N/2} \text{ Real} &= W_{\text{real}} \times Y_{\text{real}} & \text{Temp0} \\
 X_{n+N/2} \text{ Imag} &= W_{\text{real}} \times Y_{\text{imag}} + W_{\text{imag}} \times Y_{\text{real}} & \text{Temp1} \\
 X_{n+N/2} \text{ Real} &= W_{\text{real}} \times Y_{\text{real}} + W_{\text{imag}} \times Y_{\text{imag}} & \text{Temp2} \\
 X_{n+N/2} \text{ Imag} &= W_{\text{real}} \times Y_{\text{imag}} & \text{Temp3} \\
 X_{\text{real}} &= x_n \text{ Real} + X_{n+N/2} \text{ Real} \\
 X_{\text{imag}} &= x_n \text{ Imag} + X_{n+N/2} \text{ Imag} \\
 Y_{\text{real}} &= y_n \text{ Real} - X_{n+N/2} \text{ Real} \\
 Y_{\text{imag}} &= y_n \text{ Imag} - X_{n+N/2} \text{ Imag}
 \end{aligned}$$



# FFT 16point radix2 Block Diagram

짝수

홀수

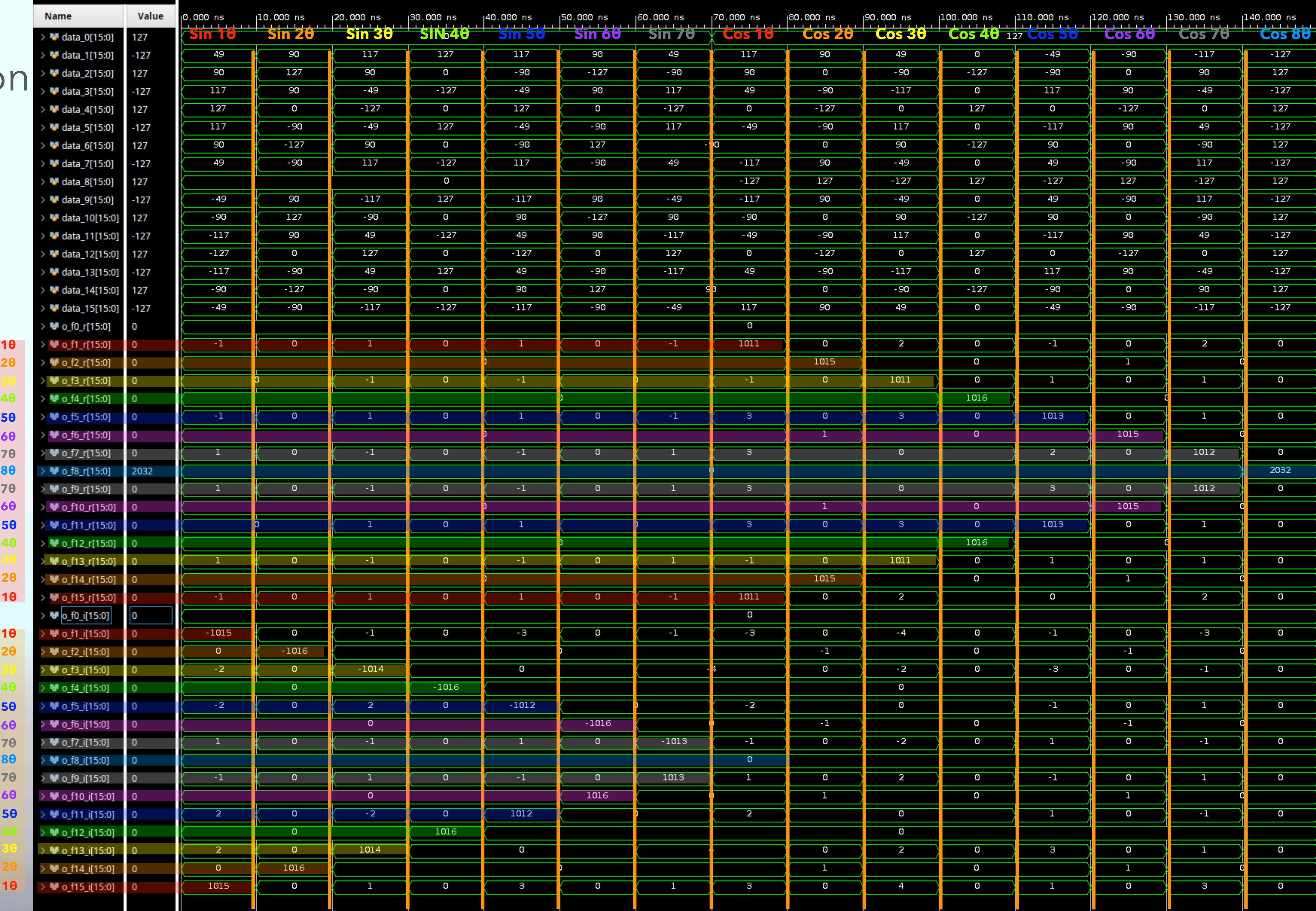


# FFT Verification

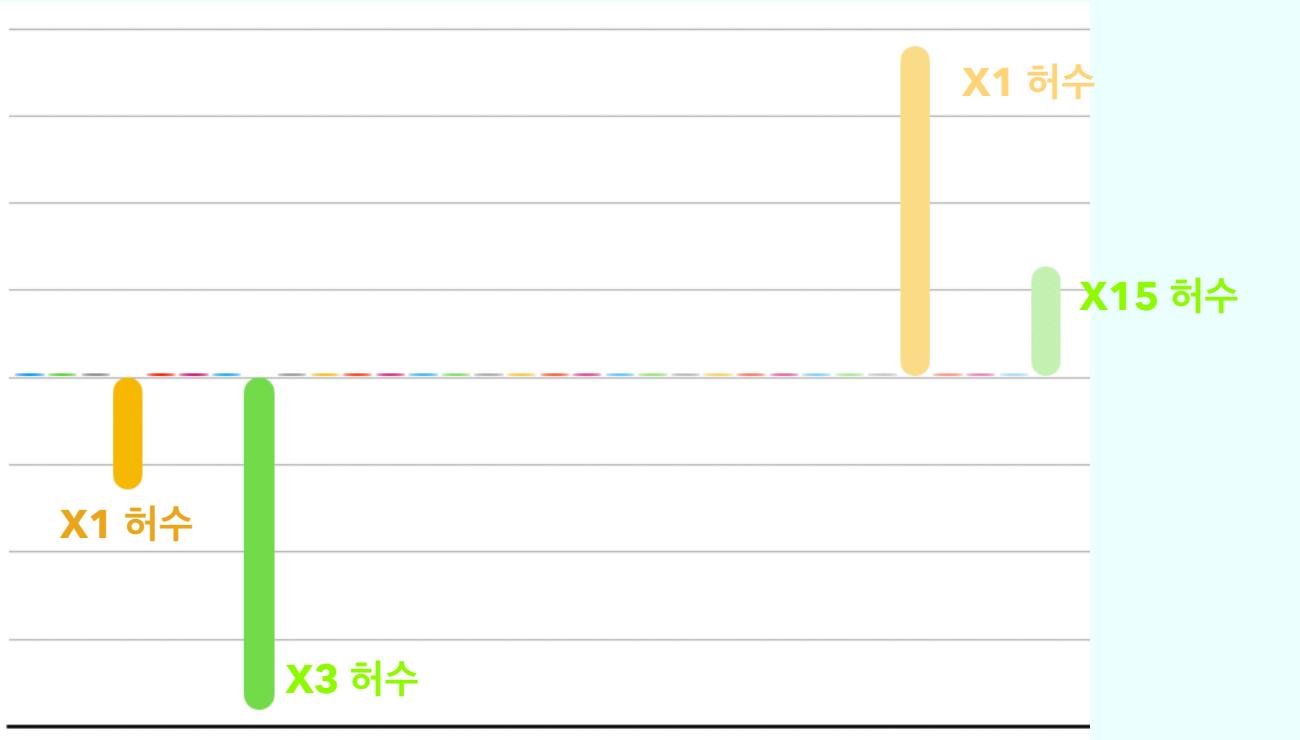
## Test bench

Cos (실수)

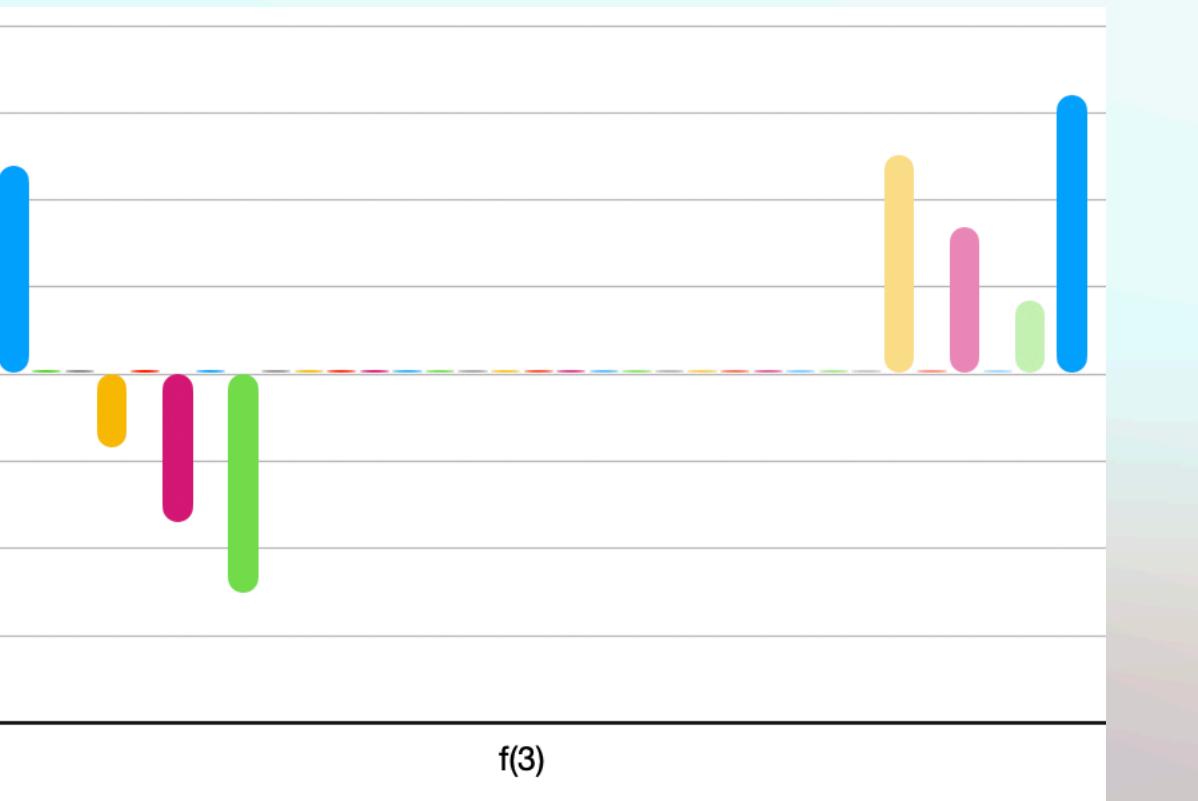
Sin (허수)



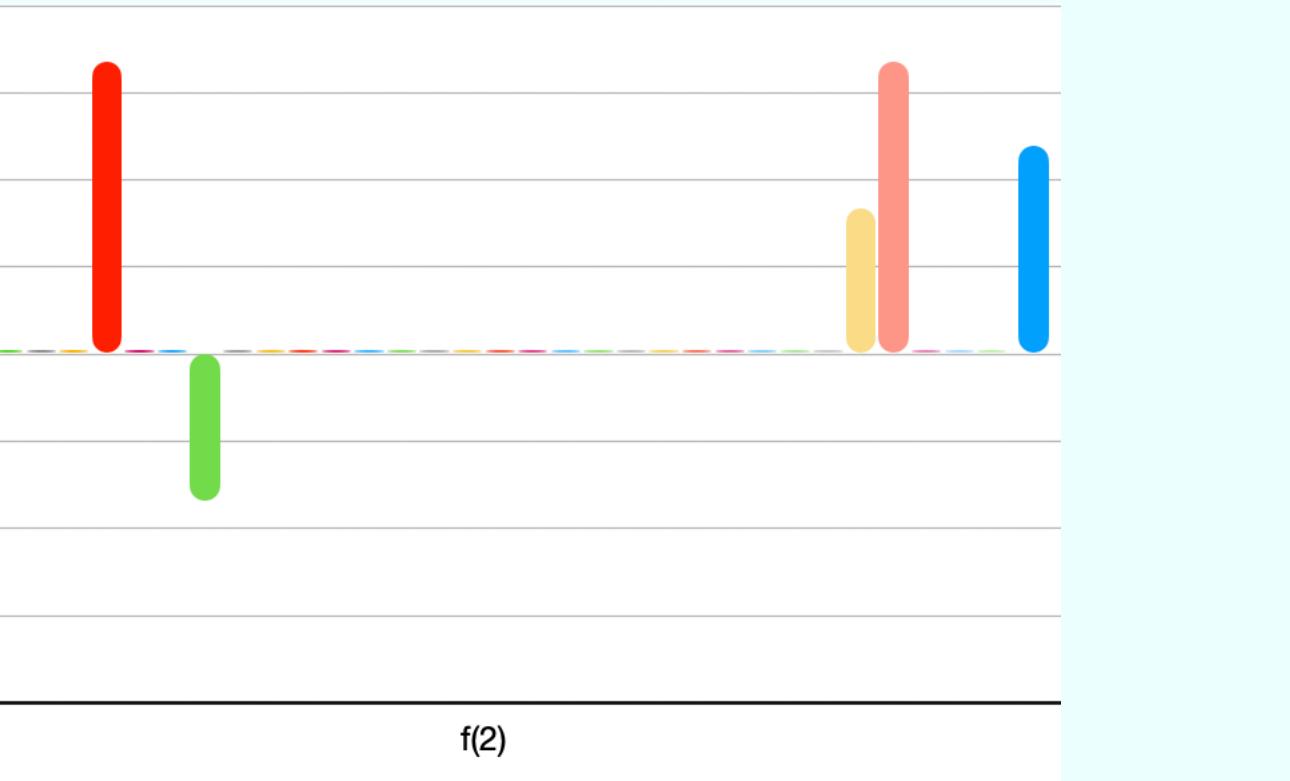
# FFT 16p Test bench 2



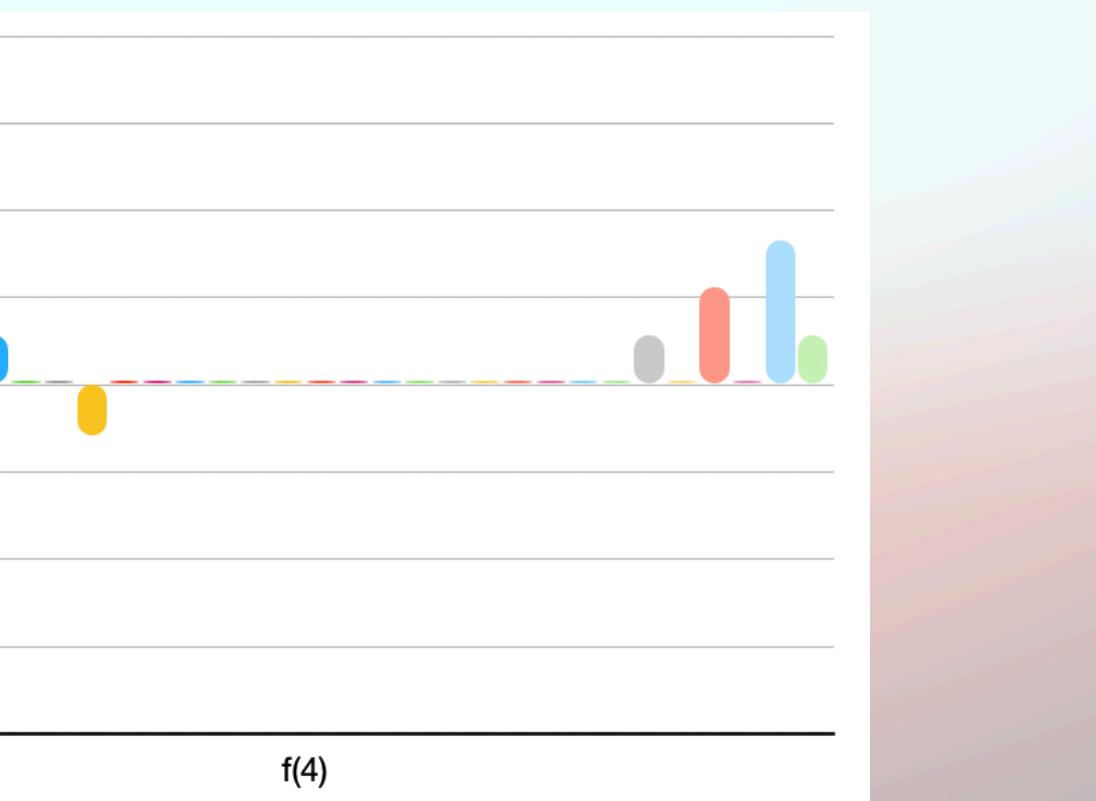
$$f(1) = 3\sin 3\theta + \sin \theta$$



$$f(3) = \sin \theta + 2\sin 2\theta + 3\sin 3\theta$$



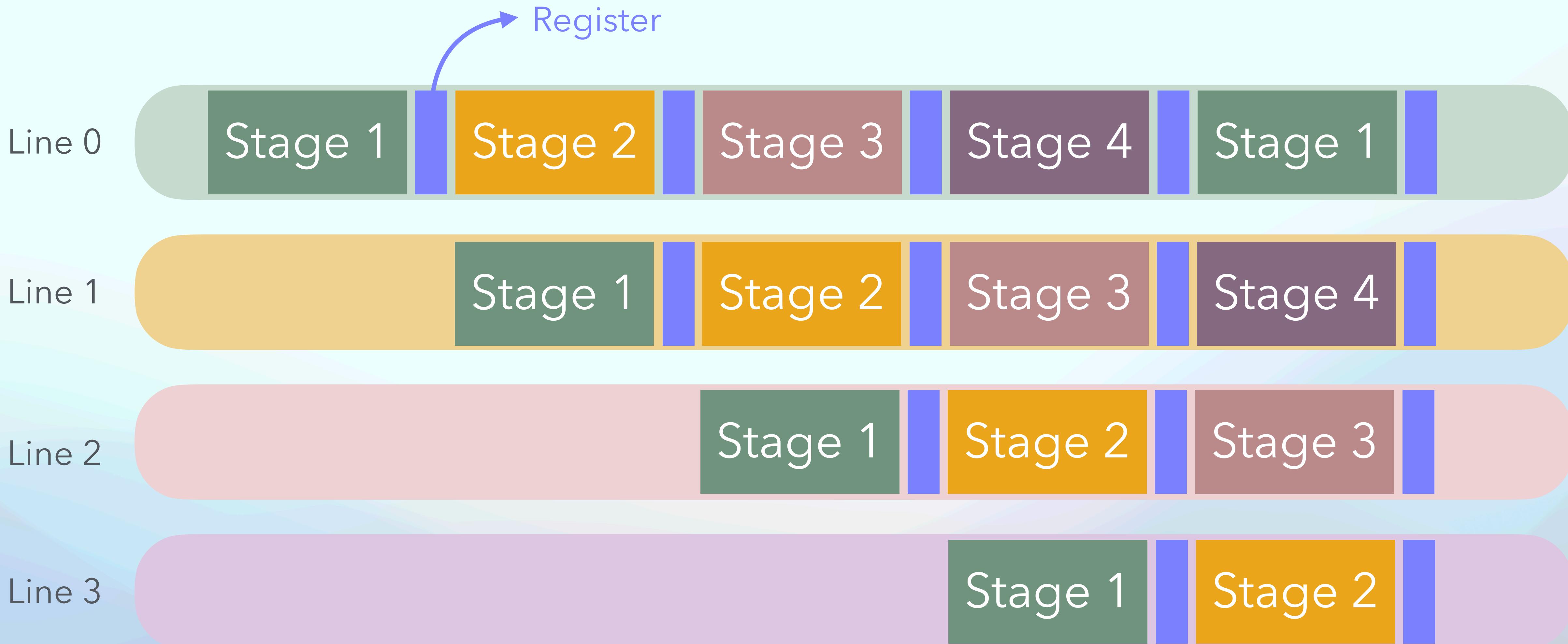
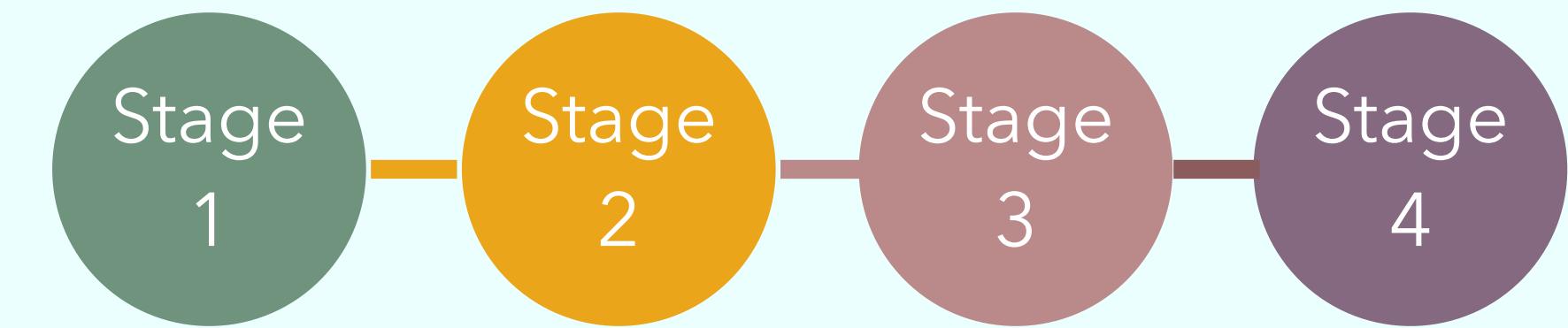
$$f(2) = 2\cos 2\theta + \sin 3\theta$$

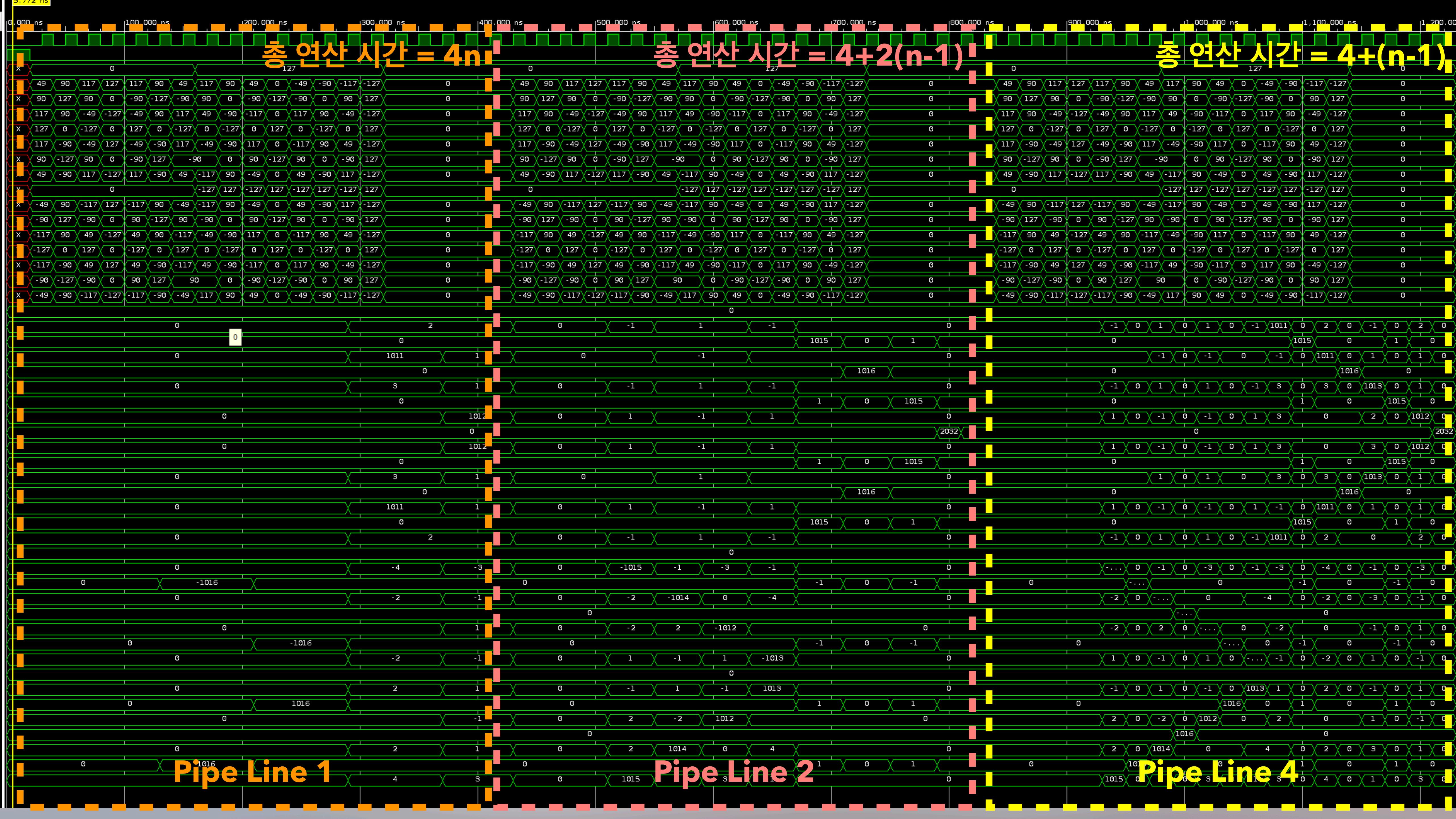


$$f(4) = 3\cos \theta + 2\cos 2\theta + \cos 3\theta + \sin 3\theta$$

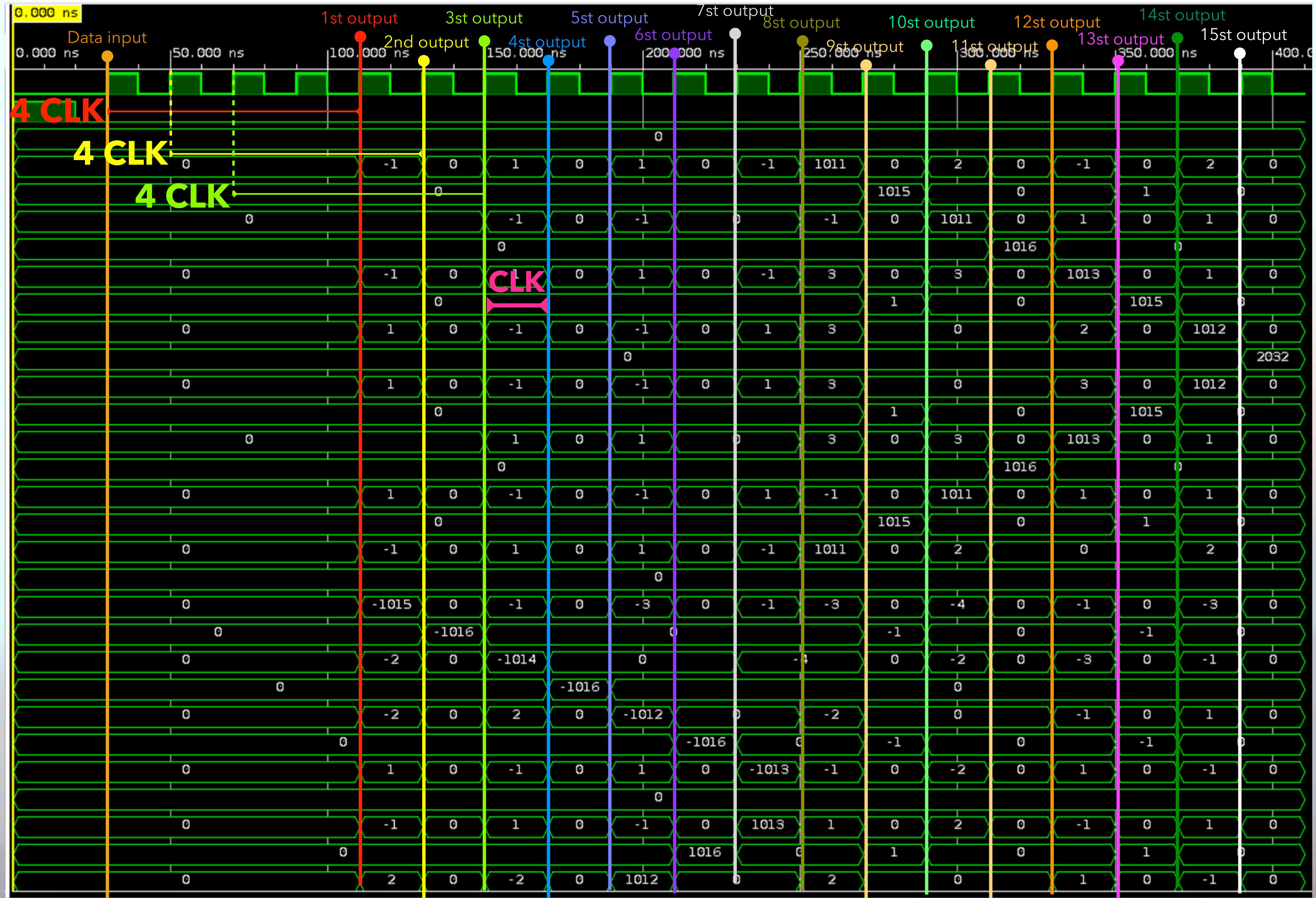
Name	Value	0.000 ns	10.000 ns	20.000 ns	30.000 ns
>  data_0[15:0]	124	0	84	30	124
>  data_1[15:0]	117	100	98	126	117
>  data_2[15:0]	69	90	29	132	69
>  data_3[15:0]	18	-7	-76	55	18
>  data_4[15:0]	-2	-64	-127	-12	-2
>  data_5[15:0]	12	-7	-76	-4	12
>  data_6[15:0]	31	90	29	47	31
>  data_7[15:0]	29	100	98	66	29
>  data_8[15:0]	12	0	84	30	12
>  data_9[15:0]	3	-100	20	-6	3
>  data_10[15:0]	11	-90	-29	13	11
>  data_11[15:0]	23	7	-43	64	23
>  data_12[15:0]	26	64	-42	72	26
>  data_13[15:0]	29	7	-43	5	29
>  data_14[15:0]	49	-90	-29	-72	49
>  data_15[15:0]	91	-100	20	-66	91
>  o_f0_r[15:0]	642	0	-3	480	642
>  o_f1_r[15:0]	331	1			331
>  o_f2_r[15:0]	223	0	670	0	223
>  o_f3_r[15:0]	114	-1	0		114
>  o_f4_r[15:0]	0	0	-1	0	
>  o_f5_r[15:0]	4	1	0	0	4
>  o_f6_r[15:0]	1	0	4	0	1
>  o_f7_r[15:0]	-1	-1	0		-1
>  o_f8_r[15:0]	-2	0	1	0	-2
>  o_f9_r[15:0]	-1				-1
>  o_f10_r[15:0]	1	0	4	0	1
>  o_f11_r[15:0]	4	1	0		4
>  o_f12_r[15:0]	0	0	-1	0	
>  o_f13_r[15:0]	114	-1	0		114
>  o_f14_r[15:0]	223	0	670	0	223
>  o_f15_r[15:0]	331	1	0	1	331
>  o_f0_i[15:0]	0				0
>  o_f1_i[15:0]	-2	-255	3	-169	-2
>  o_f2_i[15:0]	-1	0	-1	-338	-1
>  o_f3_i[15:0]	-115	-761	-336	-502	-115
>  o_f4_i[15:0]	0	0	-2	0	
>  o_f5_i[15:0]	-1	3	-2	2	-1
>  o_f6_i[15:0]	-1	0	-1	2	-1
>  o_f7_i[15:0]	0	-3	-1	-1	0
>  o_f8_i[15:0]	0			0	
>  o_f9_i[15:0]	0	3	1	0	
>  o_f10_i[15:0]	1	0	1	-2	1
>  o_f11_i[15:0]	1	-3	2	-2	1
>  o_f12_i[15:0]	0	0	2	0	
>  o_f13_i[15:0]	115	761	336	502	115
>  o_f14_i[15:0]	1	0	1	-338	1
>  o_f15_i[15:0]	2	255	-3	169	2

# Pipe Line



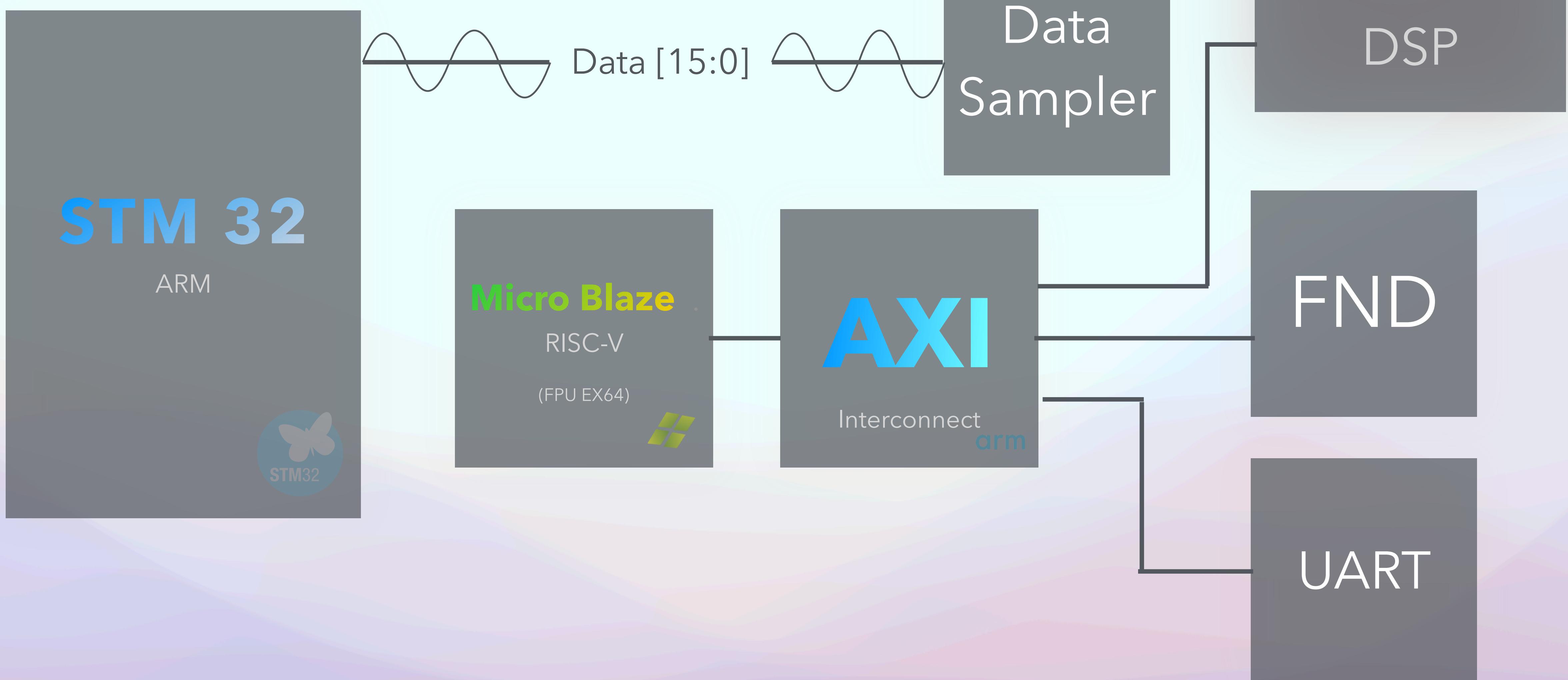


# FFT Pipeline



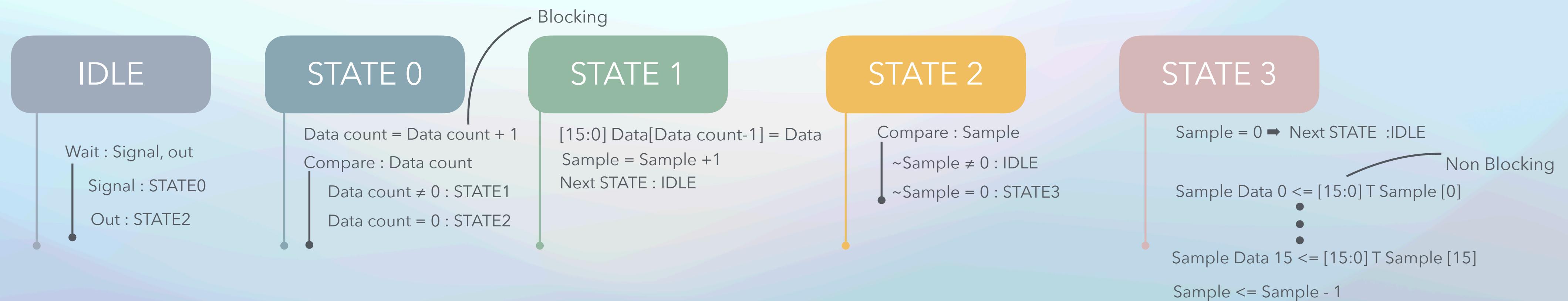
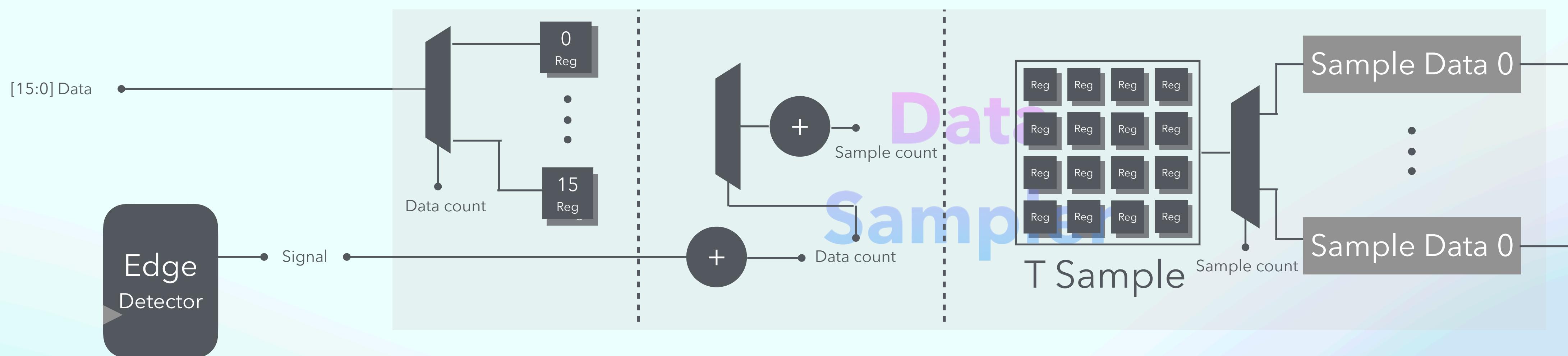
# FFT Verification

## SoC Block Diagram



# Data Sampler

## FFT Verification



# FFT Verification

# Result

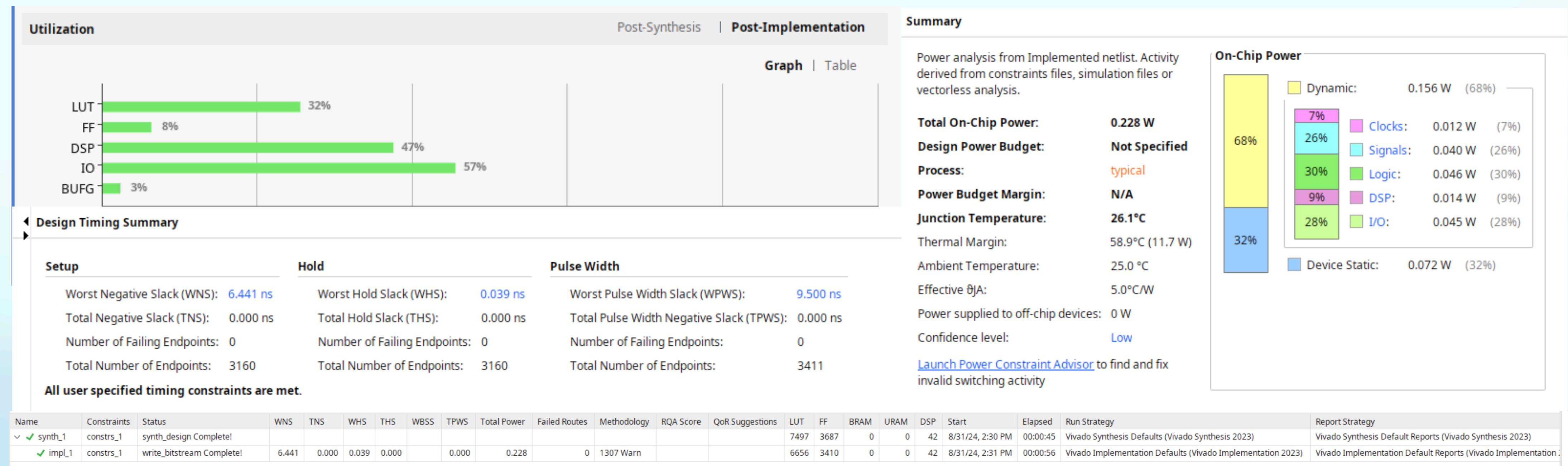
# 프로젝트 결과

오류, 문제점, 해결 과정

- Signed 곱셈과정
  - Signed 변수 간 곱셈에서 내부처리 과정 중 Unsigned 처리되는 버그
- Adder, Multiplier, FPU 설계 (내부 연산 유닛블록)
  - 시간 부족으로 내부 DSP로 대체
- STM32 Data 출력 주기가 일정하지 않음, GPIO 속도 문제
  - GPIO 출력 중 데이터를 읽는 문제가 발생, Data Sampler 모듈 추가
- Negative Slack 발생
  - 크리티컬 패스 확인, 클럭 조정

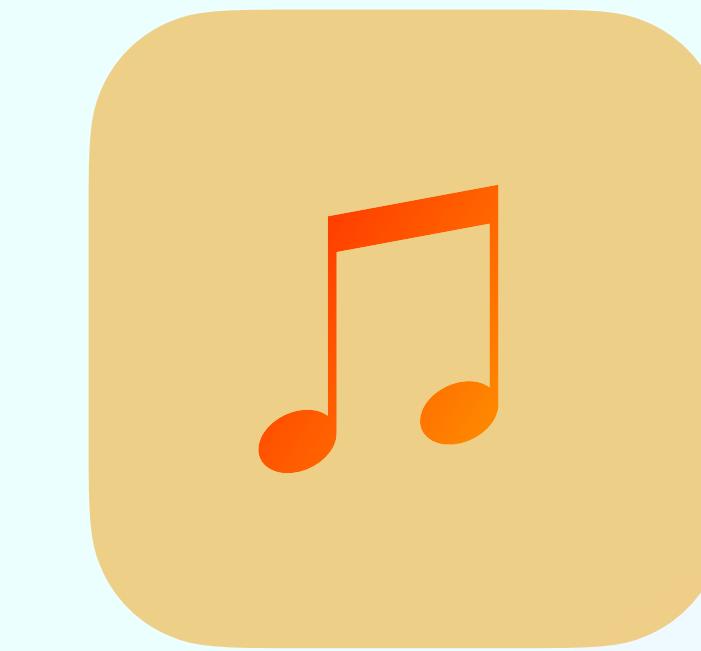
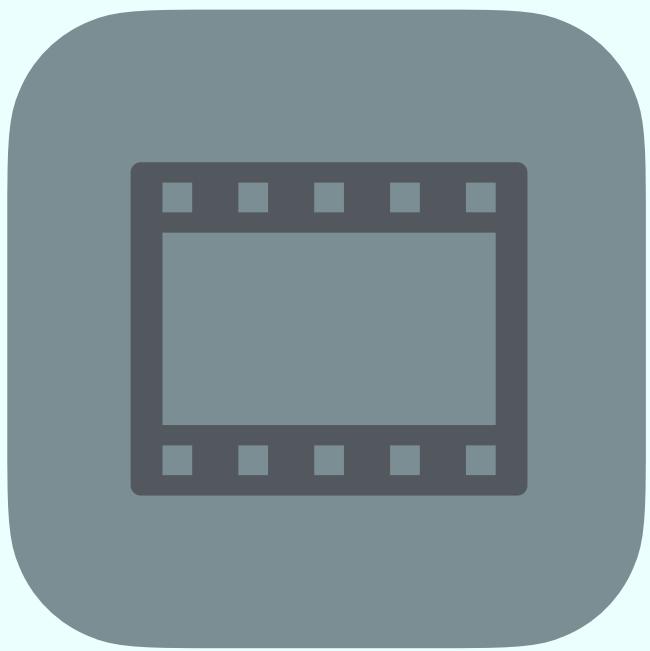
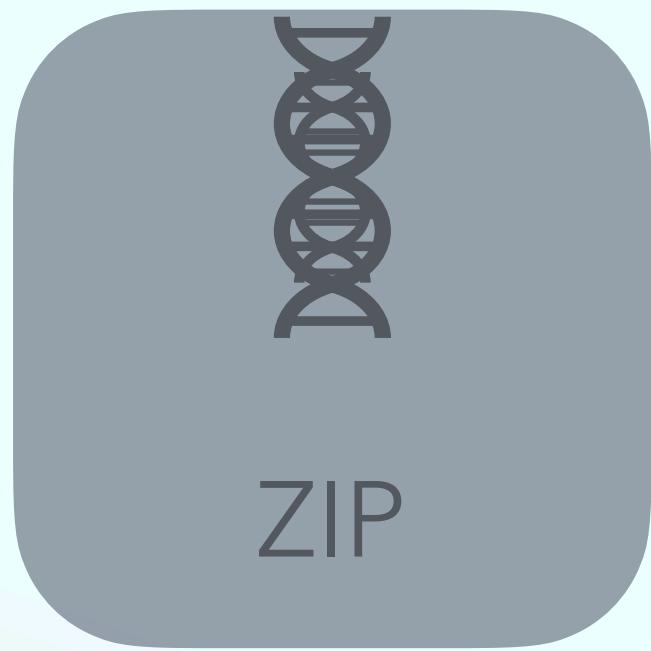
# 프로젝트 결과

하드웨어 사용량,

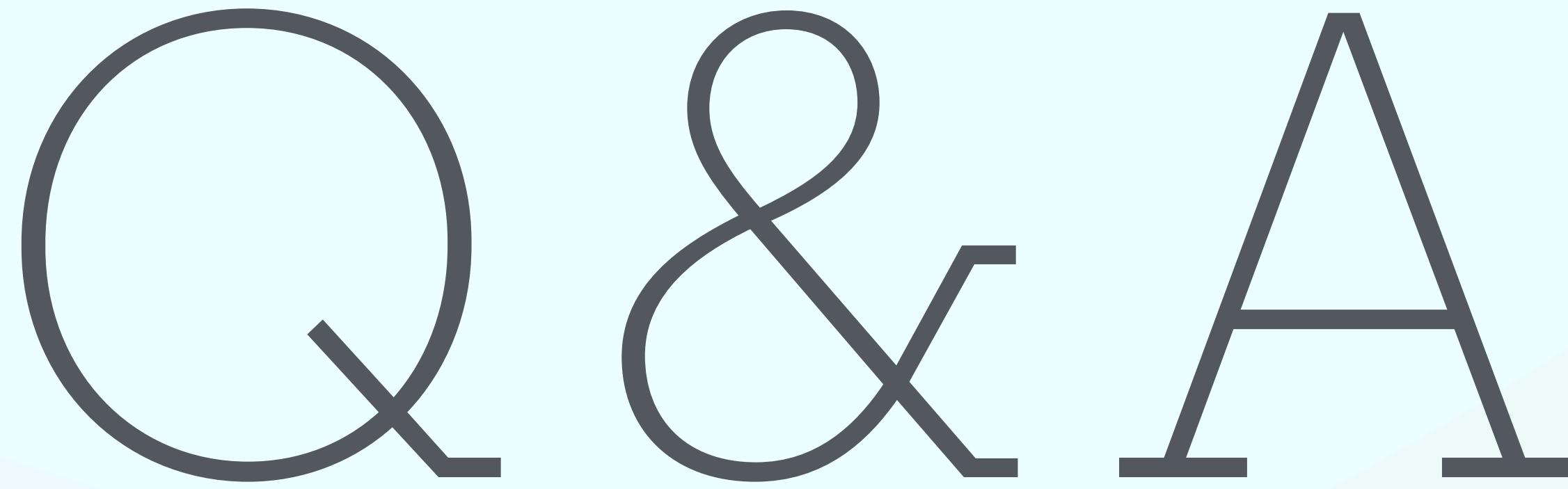


# 프로젝트 작품 활용방안

FFT Proceses unit



# Pipe Line



유재훈 - [o\\_ayjh@naver.com](mailto:o_ayjh@naver.com), 유승범 - [ca0602@naver.com](mailto:ca0602@naver.com)



Code



**Portfolio**

유재훈



유승범