Electronic Systems

Direct Digital Frequency Synthesizer

Ing. Luca Zulberti – luca.zulberti@phd.unipi.it
Prof. Massimiliano Donati – massimiliano.donati@unipi.it
Prof. Luca Fanucci – luca.fanucci@unipi.it

Agenda

1) DDFS System Description
2) Frequency Quantization
3) Amplitude Quantization
4) LUT optimization

Agenda

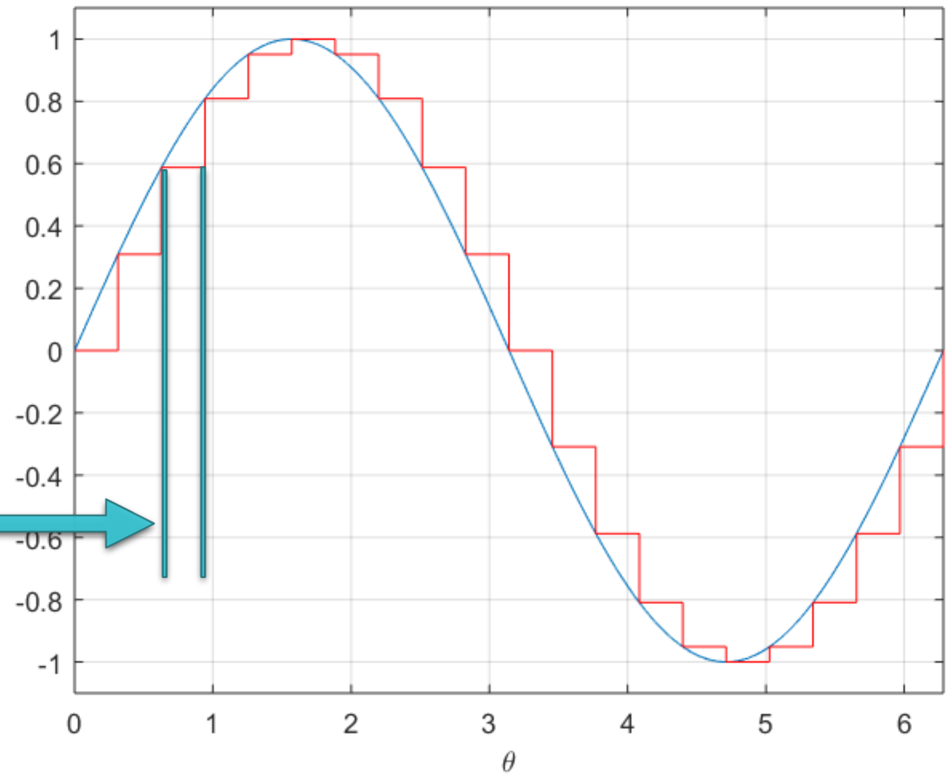
1) **DDFS System Description**
2) Frequency Quantization
3) Amplitude Quantization
4) LUT optimization

DDFS System Description

The DDFS (Direct Digital Frequency Synthesizer) is a digital system able to generate a quantized *sin* waveform, with frequency control.

Suppose we want to generate a single tone at frequency f:

- Ideal output: $\sin(2\pi f t) = \sin(\theta)$
- Discrete time: $\sin(2\pi f k T_{ck}) = \sin(k\Delta\theta)$
- Phase resolution: $\Delta\theta = 2\pi \frac{f}{f_{ck}}$
    - Depends on f
    - Depends on f_{ck}



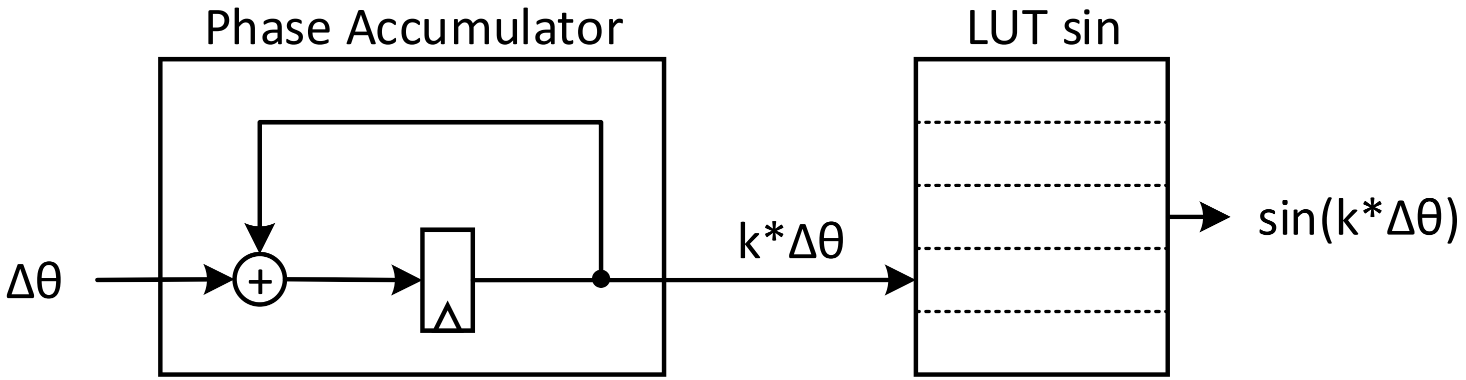
DDFS System Description

The delta-theta, $\Delta\theta$, is accumulated each clock cycle to sweep the phase θ.

Phase accumulator: $\theta = k\Delta\theta$

sin values are stored and accessed from a ROM (or simply a Look Up Table), using the phase $k\Delta\theta$ as address.

LUT for sin values: $k\Delta\theta \rightarrow \sin(k\Delta\theta)$



Agenda

1) DDFS System Description
2) Frequency Quantization
3) Amplitude Quantization
4) LUT optimization

Frequency Quantization

Ideal $\theta \in [0, 2\pi)$. We remap the interval using N-bit into the $[0{:}\,2^N - 1]$ as unsigned

$$LSB_\theta = \frac{2\pi}{2^N}$$

We call the remapped unsigned value the ***frequency word***:

$$fw \in [0{:}\,2^N - 1] \qquad \Delta\theta = fw \times LSB_\theta$$

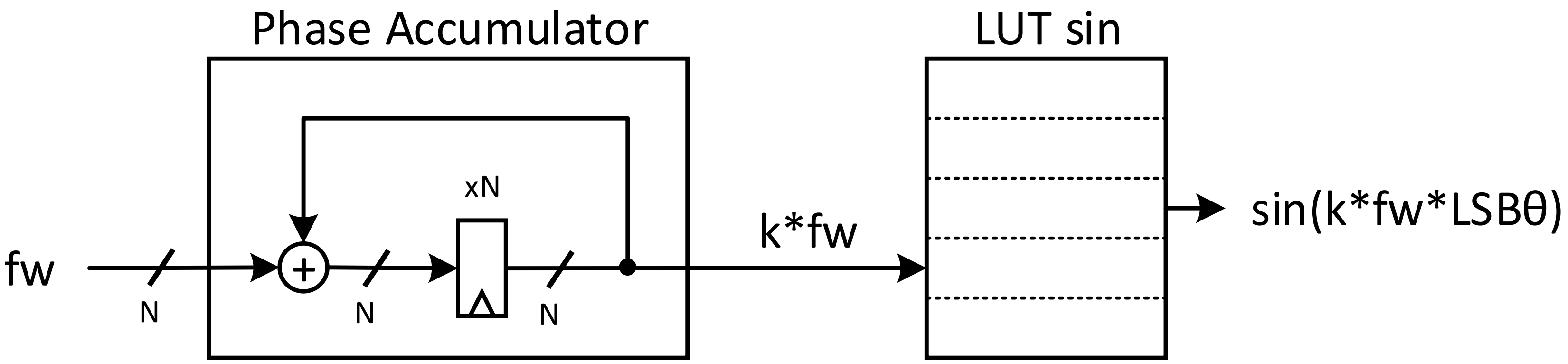
fw	$\Delta\theta$
00	0
01	$1/2\ \pi$
10	π
11	$3/2\ \pi$

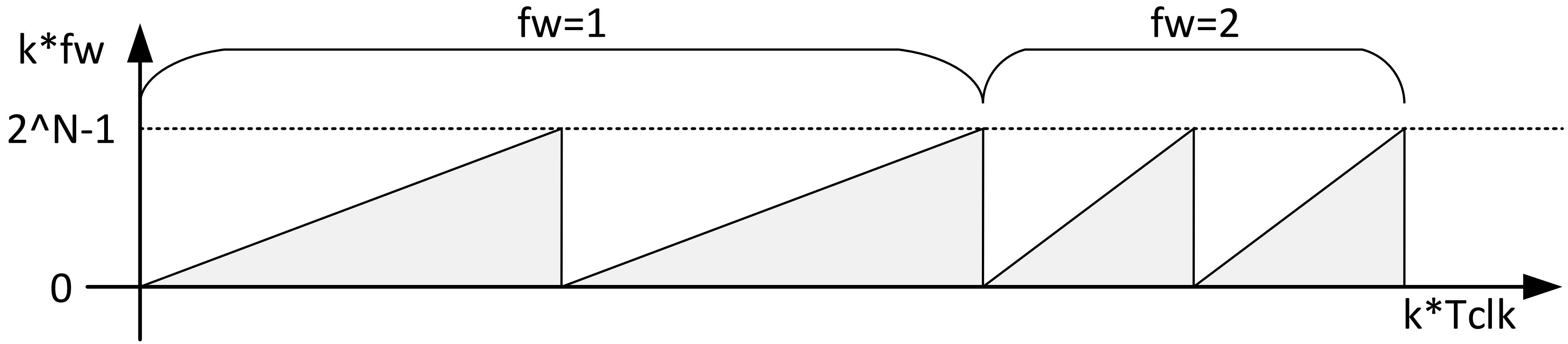
Since $f = \frac{\Delta\theta}{2\pi} f_{clk}$ is the formula for the frequency resolution.
The minimum output frequency corresponds to $\Delta\theta = LSB_\theta$ ($fw = 1$) and it depends on N:

$$f_1 = \frac{f_{clk}}{2^N}$$

Frequency Quantization

Phase Accumulator Datapath





Frequency Quantization

Which is the value N-bit to choose?
It depends on the frequency resolution (and error) **requirements**.

Suppose to have $f_{clk} = 125\ MHz$
We want to sweep the DDFS frequency range with an accuracy less than $40\ kHz$:

$$f_1 = \frac{f_{clk}}{2^N} < 40\ kHz \quad N = \left\lceil \log_2 \frac{125\ MHz}{40\ kHz} \right\rceil = 12\ bit$$

$$f_1 = \frac{f_{clk}}{2^N} \approx 30.5\ kHz$$

Agenda

1) DDFS System Description
2) Frequency Quantization
3) Amplitude Quantization
4) LUT optimization

Amplitude Quantization

We need the LUT for the *sin* values with N inputs (2^N lines): $\sin\left(k\frac{2\pi}{2^N}\right) \in [-1, 1]$

But we need to quantize also the *sin* values!

Suppose to use M-bit for the signed quantization, **using a C2 balanced representation**:

$$[-1, 1] \rightarrow [-2^{M-1} + 1 : 2^{M-1} - 1] \times LSB_A$$

$$LSB_A = \frac{1}{2^{M-1} - 1}$$

M depends on the signal resolution requirements

Amplitude Quantization

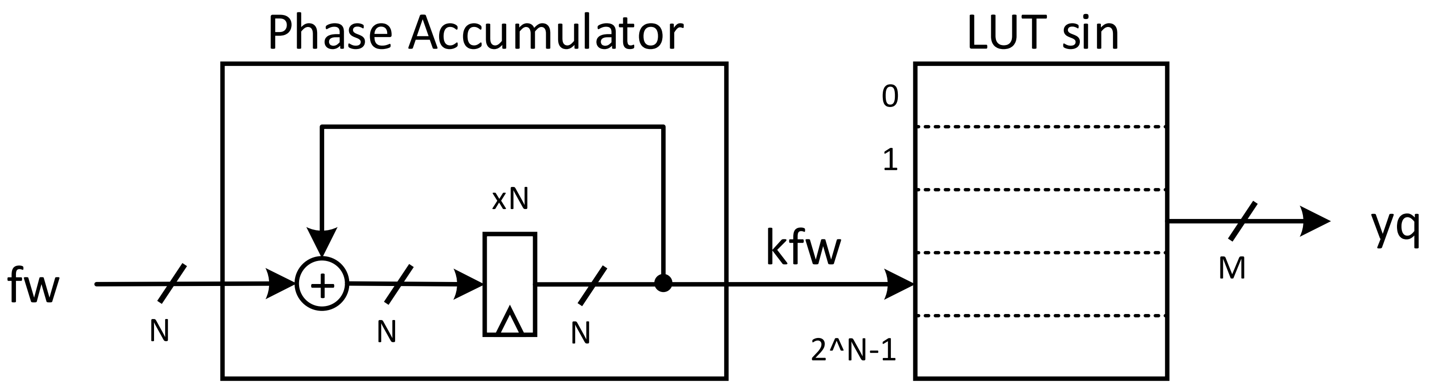
How to generate the quantized LUT values?

Using the standard quantization formula with the $LSB_A = \frac{1}{2^{M-1}-1}$

$$y(k) = \sin\left(k\frac{2\pi}{2^N}\right) \quad k \in [0{:}\,2^N - 1]$$

$$y_q(k) = LSB_A \underbrace{\left[\frac{y(k)}{LSB_A}\right]} \quad k \in [0{:}\,2^N - 1]$$

Integer to convert in C2 and to store in the LUT



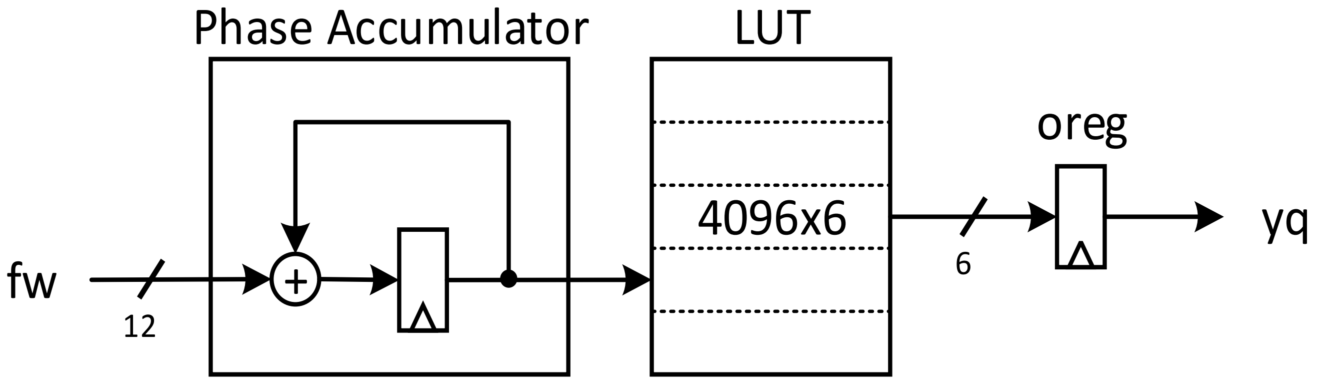
Exercise

Implement a DDFS in VHDL with the following interface:

```vhdl
entity DDFS is
  port (
    clk   : in  std_logic;                   -- suppose 125 MHz
    reset : in  std_logic;                   -- asynchronous reset, active high
    fw    : out std_logic_vector(11 downto 0);   -- frequency word
    yq    : out std_logic_vector(5 downto 0)     -- quantized sin
  );
end entity;
```

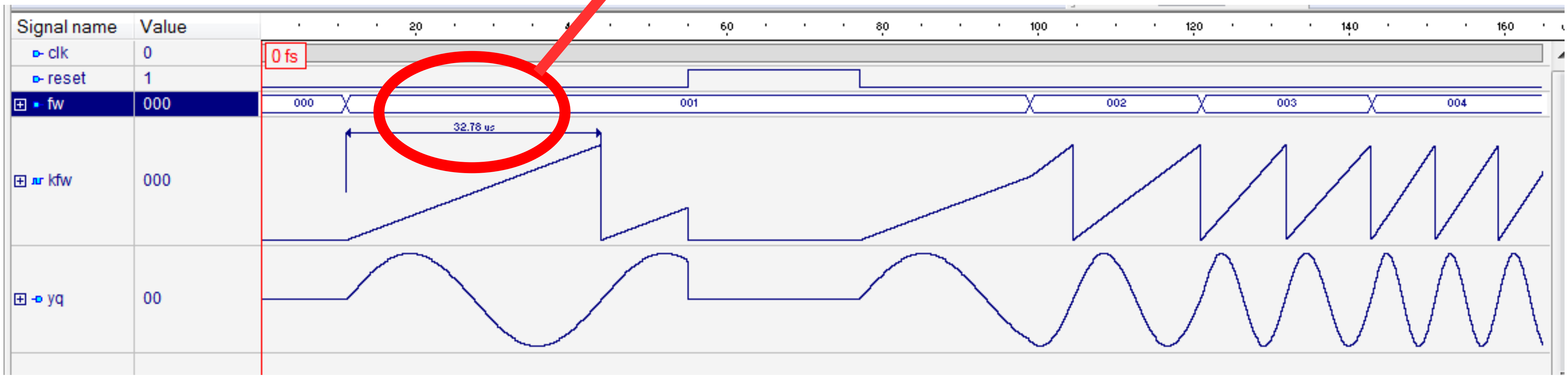
Write a testbench to verify the functionality.

- use `clk` @125MHz
- verify f_1 frequency resolution
- change `fw` on-the-fly
- test the reset



Exercise

Simulation $clk\ @125\ MHz$ $32.768\ \mu s \rightarrow 30.521\ kHz = f_1$



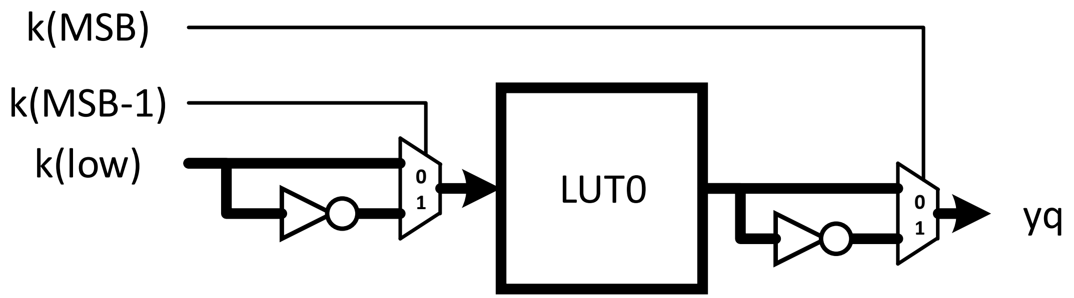
Agenda

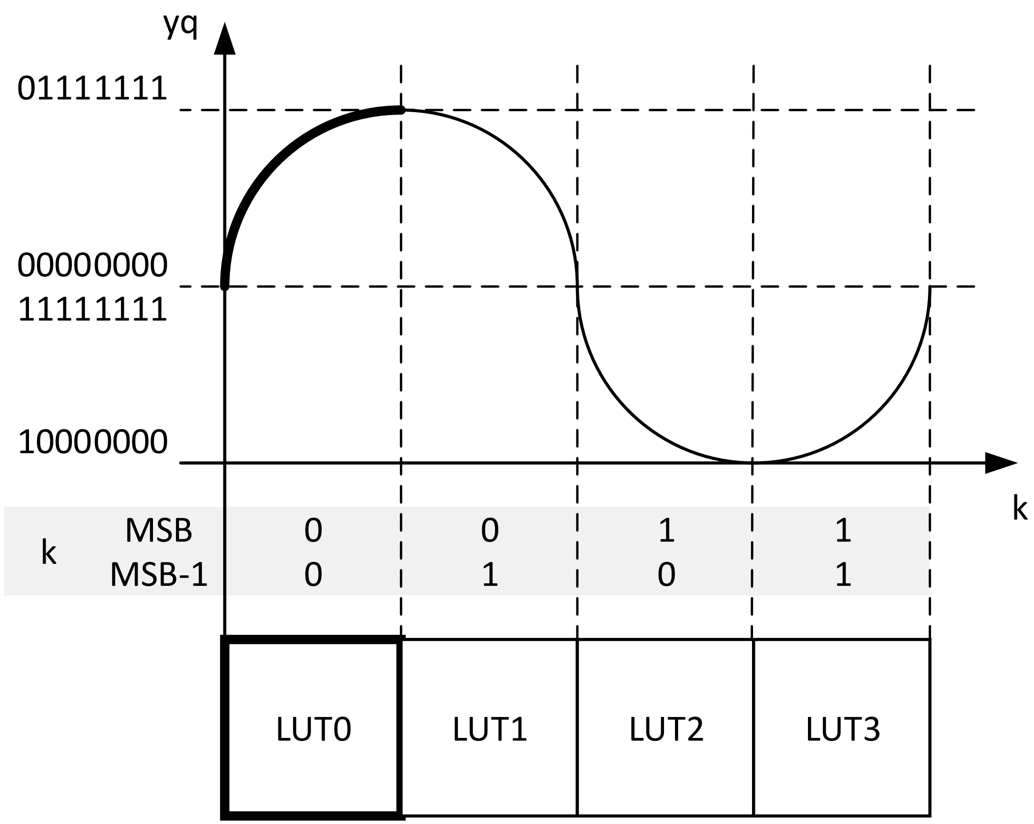
1) DDFS System Description
2) Frequency Quantization
3) Amplitude Quantization
4) LUT optimization

LUT Optimization

Sine wave has quadrangular-symmetrical profile
We can use just the first quarter, LUT0, adjusting

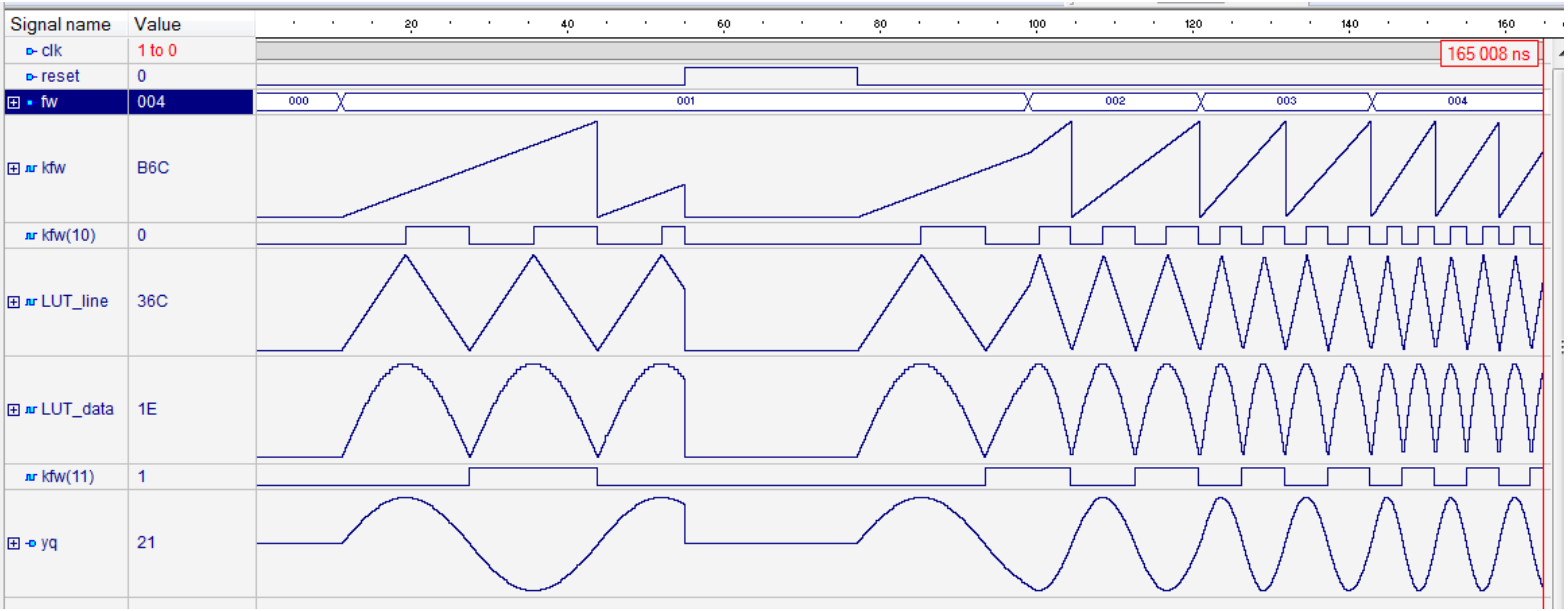
- access modality: normal or reverse (complement)
- amplitude polarity: plain or complemented





This will save a lot of area (% utilization) for the implementation

LUT Optimization



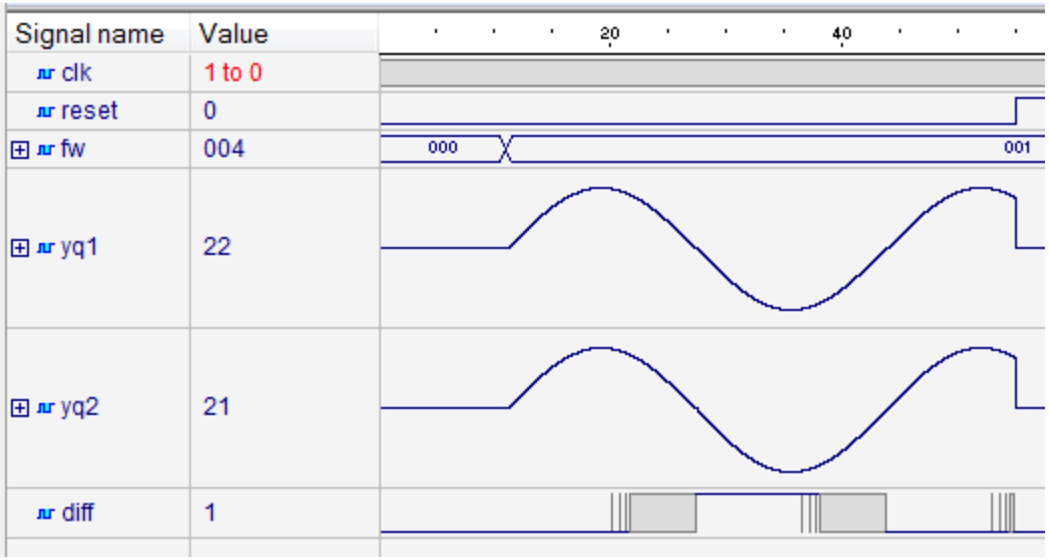
Exercise

To check if the DDFS and DDFS_QLUT are equal, we can write a comparison Test Bench

```vhdl
i_dut_std : DDSF
port map (
  clk   => clk,
  reset => reset,
  fw    => fw,
  yq    => yq_std
);

i_dut_opt : DDFS_QLUT
port map (
  clk   => clk,
  reset => reset,
  fw    => fw,
  yq    => yq_opt
);

diff <= '1' when yq_std /= yq_opt else '0';
```



Different!!??

Exercise

The DDFS_QLUT is different!!! But the differences are maximum of 1 LSB (C1 negation)

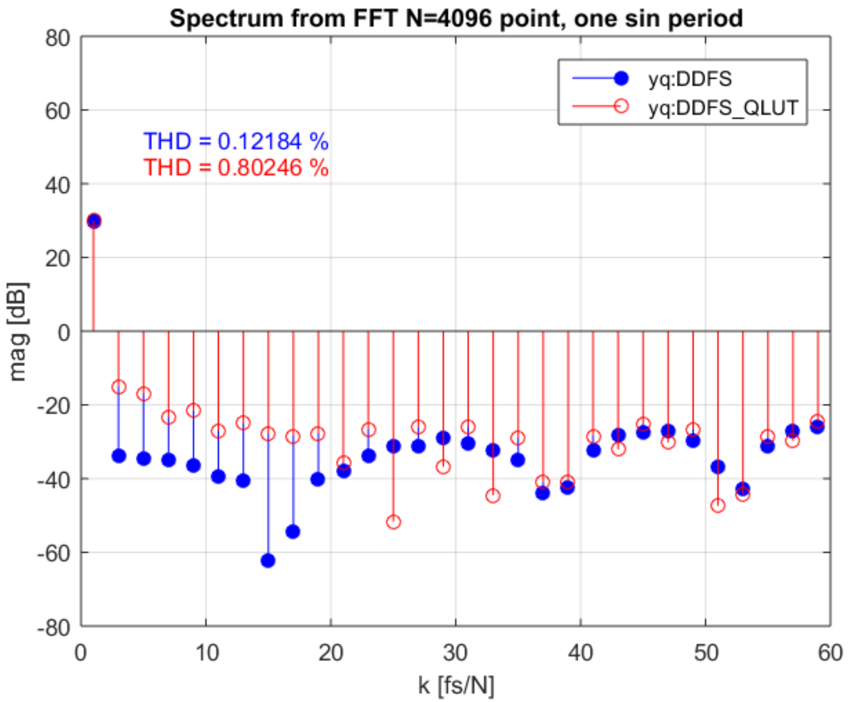
We can analyze the spectrum of one sin period in both cases

THD here is evaluated with 3rd 5th 7th 9th 11th harmonics

Usually even less harmonics are used

DDFS_QLUT is worst but both THDs are still acceptable

$$THD = \frac{\sqrt{V_2^2 + V_3^2 + V_4^2 + \cdots}}{V_1}$$



End

1) DDFS System Description
2) Frequency Quantization
3) Amplitude Quantization
4) LUT optimization