FPGA BASED SYSTEM DESIGN(2EC202)

PROJECT REPORT

Institute of Technology, Nirma University

Department of Electronics and Communication engineering

FM MODULATION

SEMESTER-IV

Submitted by :-

HIRPARA YAGNESH R.

22BEC051

INDEX

ABSTRACT:-

- In this project, I have tried to implement Frequency Modulation (FM) using Verilog and ModelSim. The design employs Numerically Controlled Oscillators (NCOs) to generate both the message and carrier signals.. FM modulation is a key technique in communication systems, facilitating efficient transmission of analog information over radio frequency carriers. The Verilog modules designed for FM modulation are rigorously simulated using ModelSim, enabling comprehensive testing and verification under various scenarios and input conditions.

- The architecture of the design includes modules for message signal generation, carrier signal generation, and the FM modulation process. NCOs are instrumental in producing sine waves for both the message and carrier signals, contributing to the accuracy and reliability of the modulation process. The simulation results affirm the successful implementation of FM modulation using Verilog meeting the requirements of real-world communication systems..

KEYWORD:-

1. Verilog
2. FM Modulation
3. NCO(Numerically Controlled Oscillators)
4. Data Flow modelling
5. Multiplier
6. Subtractor
7. Massage signal
8. Carrier signal

LITERATURE SURVEY: -

Frequency modulation is a type of angle modulation in which output signal's width changes according to message signal and amplitude as per carrier signal. Frequency modulation is of two types: Narrow Band Frequency modulation and White Band modulation. Narrow Band Frequency modulation is one type amplitude modulation.

The basic principle behind FM modulation can be described as follows:

- **Carrier Signal**: A high-frequency sinusoidal signal (carrier) is used as a carrier wave.
- **Modulating Signal**: A low-frequency signal (modulating signal) that carries the information to be transmitted is used to modulate the frequency of the carrier wave.
- **Frequency Deviation**: The amplitude of the modulating signal determines the frequency deviation of the carrier wave. Higher amplitude results in greater frequency deviation.

Mathematical Representation of FM Modulation

Mathematically, FM modulation can be represented by the following equation:

$$s(t) = A_c \cdot \cos(2\pi f_c t + 2\pi k_f \int_0^t m(\tau)d\tau)$$

Where:

- $s(t)$ is the modulated signal.
- Ac is the amplitude of the carrier wave.
- fc is the frequency of the carrier wave.
- kf is the frequency sensitivity or modulation index, representing how much the frequency changes per unit amplitude of the modulating signal.
- $m(t)$ is the modulating signal.

In FM modulation, the frequency of the carrier wave is varied based on the integral of the modulating signal. The modulation index kf determines the extent of frequency deviation from the carrier frequency.

LIMITATIONS OR DRAWBACKS OF CURRENTLY AVAILABLE TECHNOLOGY: -

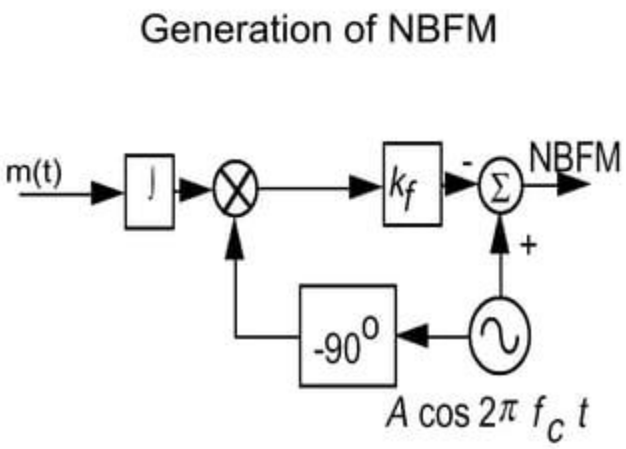
1. Direct Digital Synthesis (DDS):
    - Limited frequency resolution.
2. Field-Programmable Gate Arrays (FPGAs):
    - Limited processing resources may constrain the complexity of modulation algorithms.
    - Power consumption can be higher compared to dedicated hardware implementations.
3. Digital Signal Processors (DSPs):
    - Finite word length can introduce quantization errors.
    - Limited processing capability for real-time applications with high data rates.
4. Integrated Circuits (ICs):
    - Limited flexibility for customization compared to programmable solutions.
    - Design complexity and cost can be high for specialized FM modulation ICs.

Proposed solution/methodology:

1. Firstly, I have implemented Narrow band frequency modulation which type of frequency modulation.
2. I have to give signal from function generator to Analog to digital convertor which lengthy process, therefore I used inbuilt NCO IP-core for signal generation (Message signal and Carrier signal)
3. The signal which I generated; I called that signal submodule in my top module.
4. I took sine of both signal and multiplied them

5. I generated Sinusoidal (cos) from NOC which has length less than 1 bit from the multiplied signal (because I have to ignore my MSB bit of multiplication)
6. I subtracted cos signal from multiplied signal.
7. The difference which we got from above is inverted from desired output.
8. Therefore, i invert the difference in order to get desired signal.
9. I verified the above steps in Model sim with different cases and senerios.

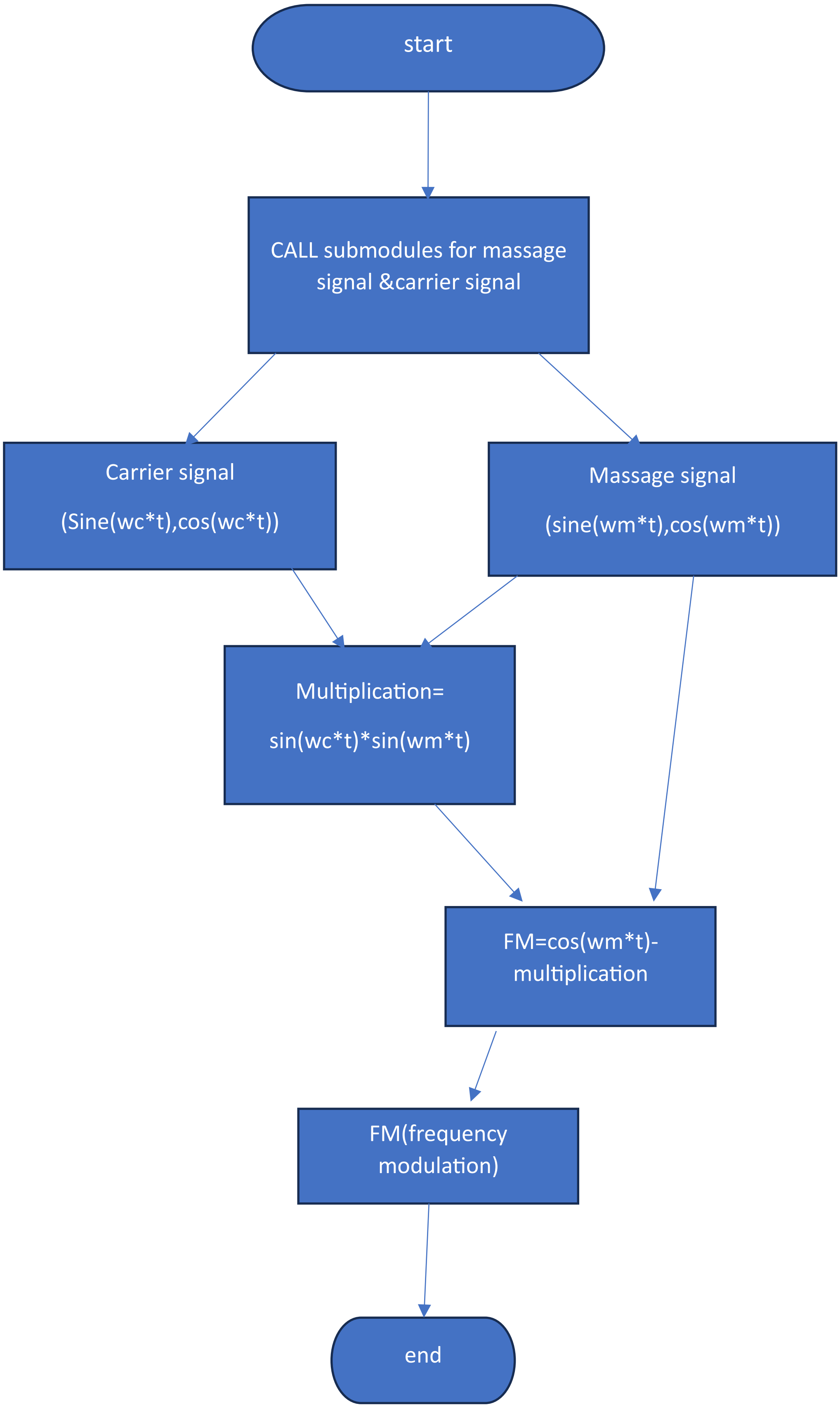
Circuit diagram: -



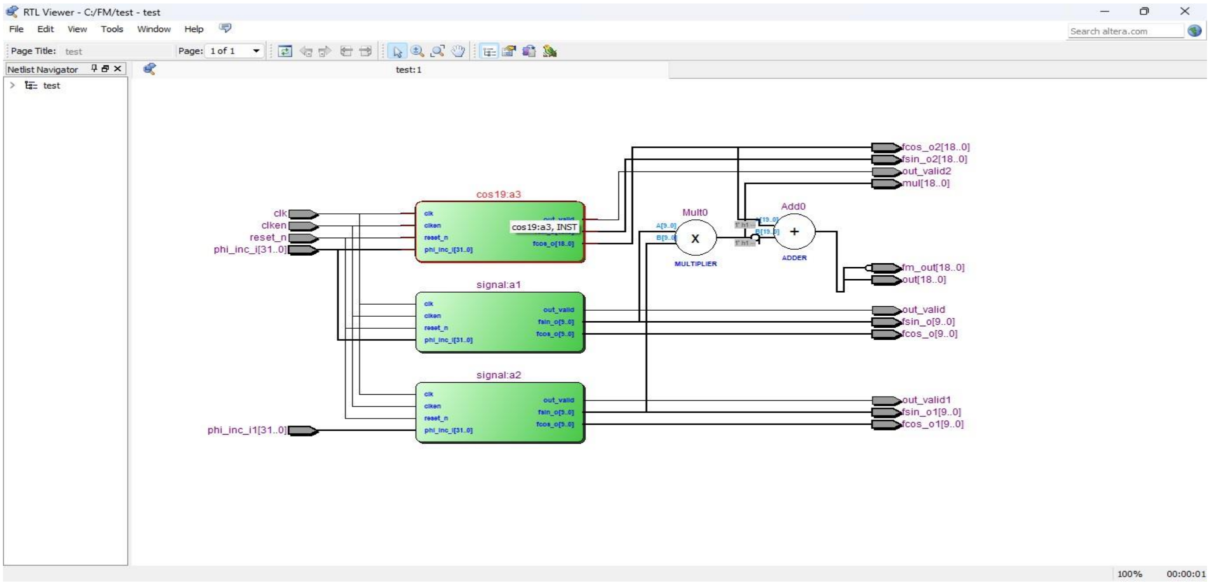
https://images.app.goo.gl/tTEiqGTHdt9MraCu7

Here,I assumed that Kf=Fm/Am. So, Beta will become 1.

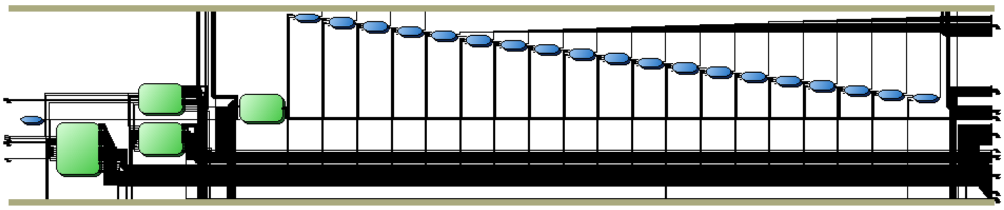
Flow chart of the code:



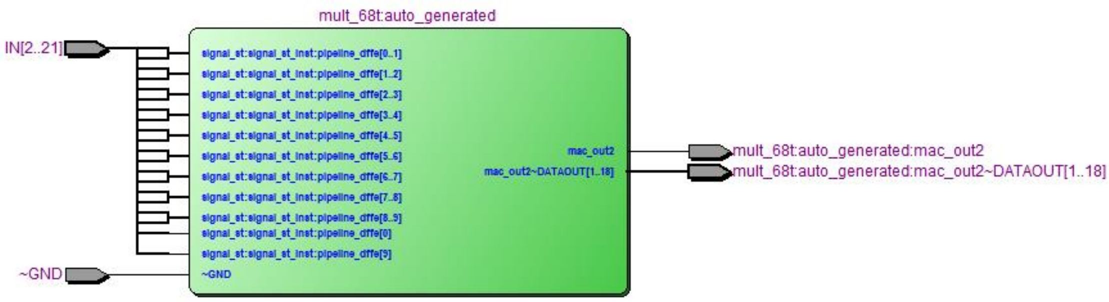
RTL VIEW:-



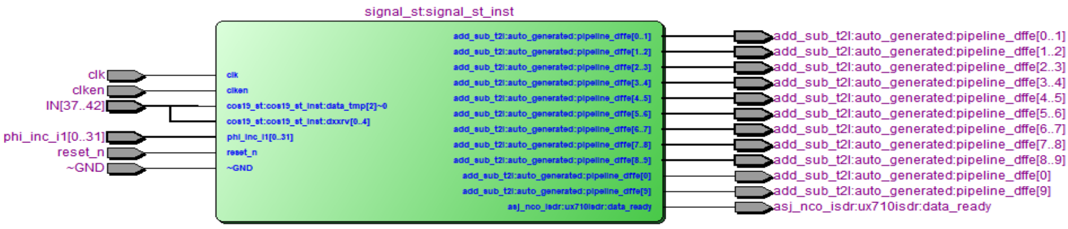
TTL VIEW:-



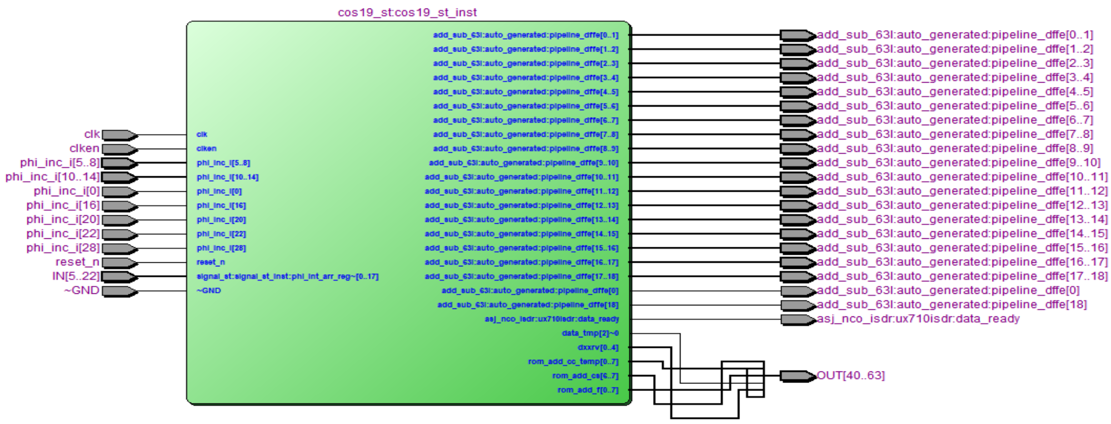
TTL OF MULTIPLIER: -



TTL OF 10bit output of signal: -



TTL OF 19bit output of signal: -



SIMULATION: -

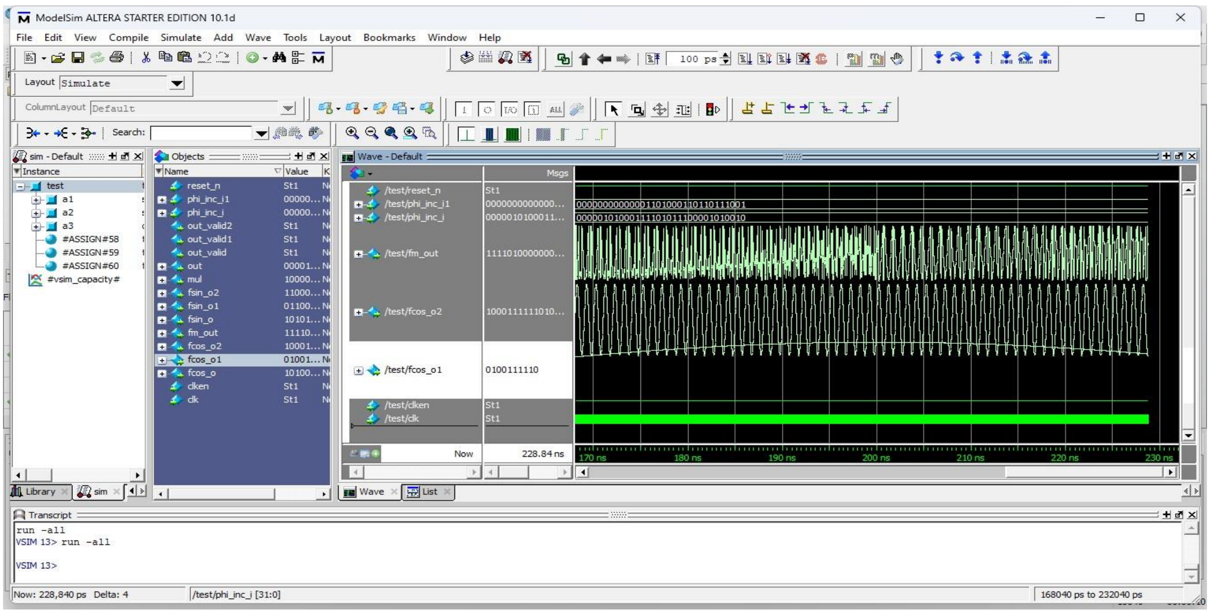


Figure-1 : Desired output

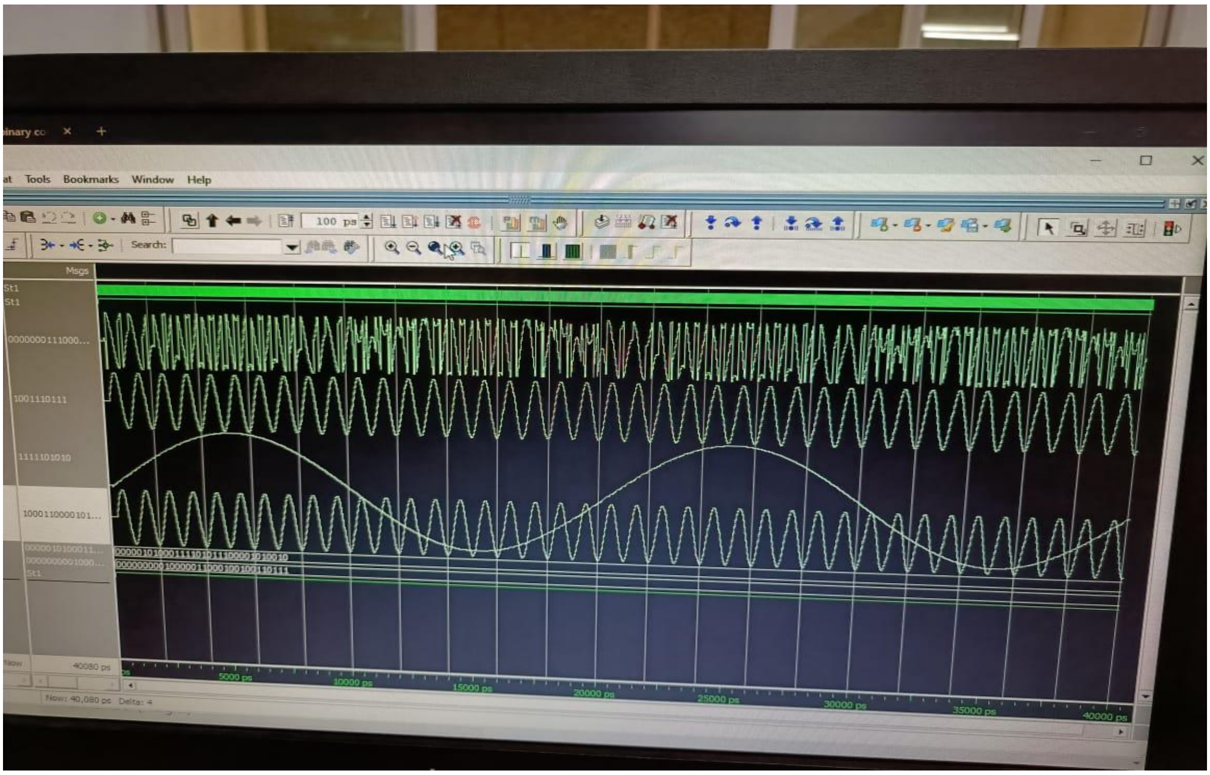


Figure-2 : Desired output

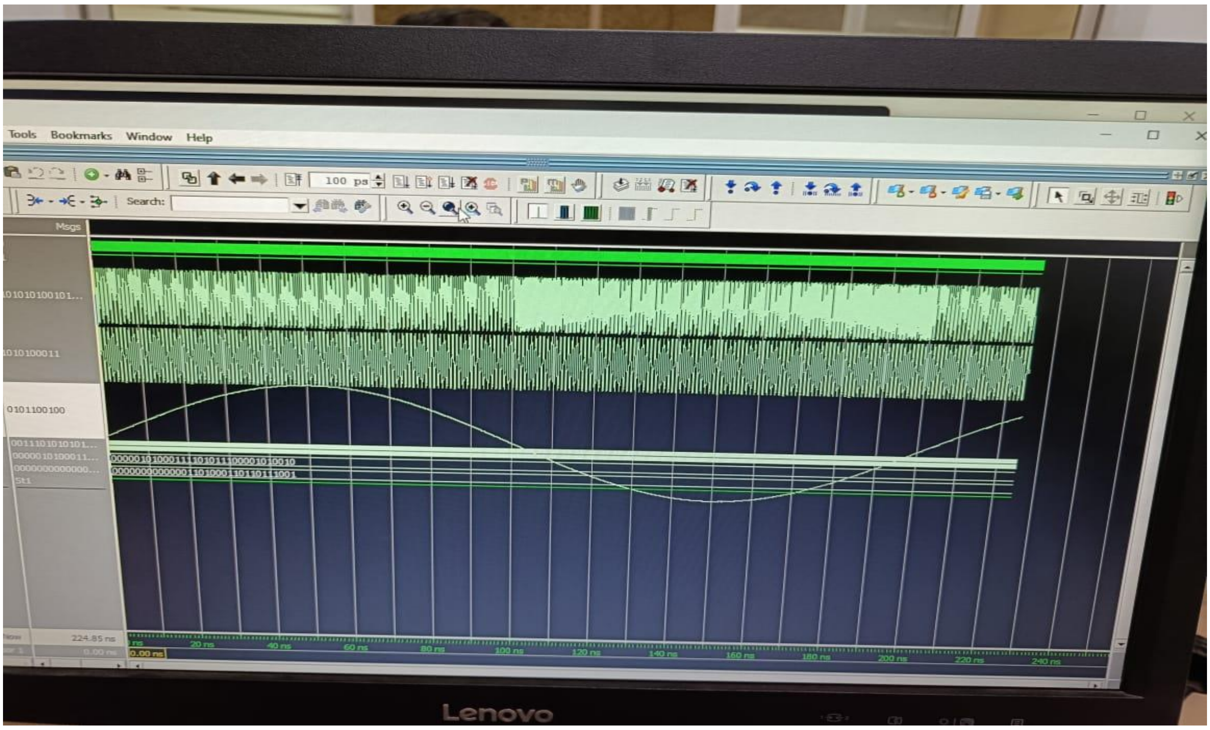


Figure-3 :without inverted output

Conclusion:

I done this project using Verilog and I learned how to implement Narrowband Frequency Modulation using Verilog by mathematical step. I also learned that How signal generate by NCO. I understand how communication is done in digitally and how it is possible that signal is transmitting from one point to other point. And I also learned how other tools are also use to watch signal.

References: -

1. Reference for literature Survey:-

http://contents.kocw.net/KOCW/document/2011/korea/koyoungchai/lecturenote16may3.pdf

”

2. Reference for narrowband frequency modulation:- https://youtu.be/ZyiHw6z2weY?si=9T1e4tPsFcUh83Vw.

3. Reference for NCO Ip core (Intel handbook):- https://cdrdv2-public.intel.com/666679/ug_nco-683406-666679.pdf

4. Reference for DSP builder:- https://www.intel.com/content/www/us/en/docs/programmable/683337/21-1/introduction.html

5. Reference for NBFM: - “Analog Communication System By Abhishek Yadav · 2008”

6. Reference for NCO Ip core (research paper):- https://www.researchgate.net/publication/261419474_Design_and_implementation_of_numerical_controlled_oscillator_on_FPGA

Code in the Appendix: -

```
module test(phi_inc_i,
  phi_inc_i1,
clk,
reset_n,
clken,
fsin_o,
fcos_o,
fsin_o2,
fcos_o2,
fsin_o1,
fcos_o1,
mul,
out_valid,
out_valid1,
fm_out,
out,
out_valid2);//my top module


input [31:0] phi_inc_i,phi_inc_i1;
input clk;
input reset_n;
input clken;
output [9:0] fsin_o,fsin_o1;
output [9:0] fcos_o,fcos_o1;
output [18:0]fcos_o2,fsin_o2;
output out_valid,out_valid1,out_valid2;
output [18:0]mul,fm_out,out;

// input carrier signal of 10 bit
signal a1(
.phi_inc_i(phi_inc_i),
.clk(clk),
.reset_n(reset_n),
.clken(clken),
.fsin_o(fsin_o),
.fcos_o(fcos_o),
.out_valid(out_valid));
```

```
// input message signal of 10 bit
signal a2(
.phi_inc_i(phi_inc_i1),
.clk(clk),
.reset_n(reset_n),
.clken(clken),
.fsin_o(fsin_o1),
.fcos_o(fcos_o1),
.out_valid(out_valid1));

// to get carrier signal output of 19 bit we generate this module
cos19 a3(
.phi_inc_i(phi_inc_i),
.clk(clk),
.reset_n(reset_n),
.clken(clken),
.fsin_o(fsin_o2),
.fcos_o(fcos_o2),
.out_valid(out_valid2));
assign mul=fsin_o*fsin_o1;
assign out=fcos_o2[18:0]-mul[18:0];// here we ingonre msb(signed 1 bit) for fm_out
assign fm_out=~out;
endmodule
```