Design and Implementation of Reconfigurable Mixer and PN Sequence Generator using GF180MCU on the MOSBius Platform

James Patrick
School of Electronics and Informatics Engineering
Institut Teknologi Bandung
Bandung, Indonesia
13222059@mahasiswa.itb.ac.id

Ibrahim Hanif Mulyana
School of Electronics and Informatics Engineering
Institut Teknologi Bandung
Bandung, Indonesia
13222111@mahasiswa.itb.ac.id

Goldwin Sonick Wijaya Thaha
School of Electronics and Informatics Engineering
Institut Teknologi Bandung
Bandung, Indonesia
13222067@mahasiswa.itb.ac.id

***Abstract*— This paper presents the design and physical implementation of a Frequency Hopping Spread Spectrum (FHSS) transceiver module on the MOSBius platform using GlobalFoundries 180nm MCU (GF180MCU) open-source technology. The system core features a Reconfigurable Double-Balanced Gilbert Cell Mixer designed to up-convert a 10.7 MHz Intermediate Frequency (IF) signal to the 61.7-68.7 MHz RF band. To align with the platform's educational objectives, a custom switching mechanism was implemented to allow dynamic toggling between internal biasing and an external MOSBius current mirror. Simulation results demonstrate functional up-conversion and down-conversion, with the standalone mixer core exhibiting a conversion loss of 10 dB. However, the integration of the reconfigurability switch introduced an additional 8 dB insertion loss, highlighting the design trade-offs in modular analog systems. A Pseudo-Random Noise (PN) Sequence Generator was also designed for frequency hopping control, but due to persistent routing congestion and pad frame pin limitations requiring 14-17 additional pins, it was excluded from the final tape out. Consequently, the final tape out consists of the DRC-clean Reconfigurable Mixer layout, with frequency hopping control offloaded to external GPIO connections**

Keywords: MOSBius, FHSS, Reconfigurable Mixer, PN Sequence Generator.

I. Introduction

Frequency Hopping Spread Spectrum (FHSS) is a wireless communication technique to transmit radio signals by rapidly changing the carrier frequency for large spectral band [1]. In an FHSS transceiver system, the frequency mixer is a key component. Its primary function is to translate signals between baseband and Radio Frequency (RF) by multiplying the input signal with a Local Oscillator (LO).

For the Mixer design, we selected the Double-Balanced Gilbert Cell topology [2]. This topology is chosen for its ability to suppress unwanted signals and provide high isolation between ports. To support the educational goals of this project, we modified the standard topology to be reconfigurable. This allows users to adjust the circuit's biasing settings dynamically, making it suitable for experimental learning. Complementing the analog mixer, we also designed a Pseudo-Random Noise (PN) Sequence Generator. This digital block is essential for controlling the dynamic frequency hopping patterns required by the FHSS system.

Currently, open-source silicon technology is a growing trend, notably with the release of the GlobalFoundries 180nm MCU (GF180MCU) Process Design Kit (PDK). This movement is further supported by accessible design environments like IIC-OSIC-TOOLS, which integrates open-source EDA software for streamlined circuit design. Within this expanding ecosystem, the IEEE Solid-State Circuits Society (SSCS) introduced the MOSBius platform. MOSBius serves as an open chip playground, designed to facilitate modular analog design and education.

In this work, we implement the Reconfigurable Mixer and a Pseudo-Random Noise (PN) Sequence Generator on the MOSBius platform. This paper documents the complete design process, covering the schematic design, physical layout, and the final tape out of the chip using the GF180MCU technology.

II. System Architecture

A. FHSS System Architecture

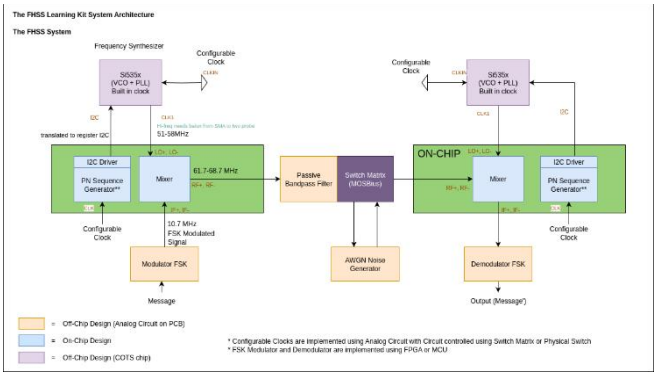

Fig. 1. FHSS System Architecture

The proposed FHSS learning kit emulates a complete communication link partitioned into custom On-Chip modules (GF180MCU) and Off-Chip COTS components. The transmission path processes an FSK-modulated Intermediate Frequency (IF) signal at 10.7 MHz, which is up-converted by the On-Chip Mixer to the 61.7–68.7 MHz RF band. To achieve spread spectrum behavior, the Local Oscillator (LO) is provided by a commercial Si535x synthesizer. This synthesizer is dynamically tuned by the On-Chip PN Sequence Generator via an I2C driver, enabling the pseudo-random frequency hopping required to synchronize the transmitter and receiver.

B. Reconfigurable Mixer Design

The core of the RF front-end is a Double-Balanced Gilbert Cell Mixer. This topology was selected for its superior port-to-port isolation, effectively suppressing LO feedthrough which is critical in low-IF architectures.

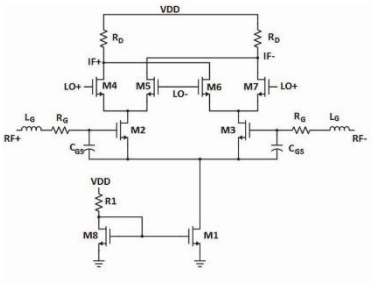

Fig. 2. The Gilbert-cell Mixer Topology

Fig. 2. Double Balanced Gilbert Cell [2]

To align with the educational objectives of the MOSBius platform, the mixer is designed with a unique reconfigurability feature. It incorporates a switching mechanism that allows the biasing network to be toggled between:

- Internal Biasing: Uses an on-chip current mirror for standalone operation.
- External Biasing: Connects to the MOSBius platform's configurable current mirror array.

This flexibility allows students to experimentally observe how variations in tail current (bias point) impact key performance metrics such as Conversion Gain, Linearity, and Noise Figure, bridging the gap between theory and practical circuit behavior.

C. PN Sequence Generator

The Pseudo-Random Noise (PN) Sequence Generator serves as the digital controller for the FHSS system. It is implemented as a Linear Feedback Shift Register (LFSR) on the custom chip.

Its primary function is to generate a pseudo-random sequence that determines the frequency hopping pattern. This digital sequence is mapped to specific frequency control words, which are transmitted to the off-chip Si535x Frequency Synthesizer via an integrated I2C driver. This ensures that both the transmitter and receiver LO frequencies "hop" in synchronization, securing the communication link against narrowband interference.

D. Pin Diagram

Finishing the plan for system architecture and design, we finalized the both On-Chip and Off-Chip system's pin list as shown below.

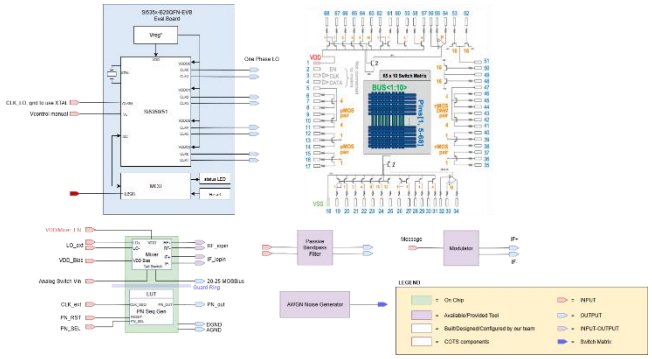

Fig. 3. Pin Diagram

TABLE I. PIN LIST

Pin #	Pin Name	Type	Description
Power & Ground			
1	VDD/Mixer_EN	Power	Positive power supply for the chip. Connect to load for gain modificaton.
2	DGND	Power	Ground reference for the PN (digital).
3	AGND	Power	Ground reference for the Mixer and Switch (analog).
RF, IF & LO Ports			
4	RF+	I/O	Non-inverting input & output for the Radio Frequency (RF) signal.
5	RF-	I/O	Inverting input & output for the Radio Frequency (RF) signal.
6	LO_ext+	I	Non-inverting input for the external Local Oscillator (LO) signal.
7	LO_ext-	I	Inverting Input for the external Local Oscillator (LO) signal.
8	IF+	I/O	Non-inverting input & output for the Intermediate Frequency (IF) signal.
9	IF-	I/O	Inverting input & output for the Intermediate Frequency (IF) signal.
Bias, Control & Digital I/O			
10	VDD_Bias	I	Analog input. External bias voltage port. Allows user to tune the VDD_Bias to control the current source. Add resistance to modify gain.
11	Analog Switch Vin	I	Digital input. Controls the internal multiplexer to select the mixer's bias source. (LOW for internal source, HIGH for external source).
12	MOSBius Current Mirror	O	Analog output. Allows user to select mixer's tail current from MOSBius when selected by Analog Switch Vin.
13	CLK_ext/PN_EN	I	Digital input. External clock signal for the PN Sequence Generator block.
14	PN_out	O	Digital output. Provides the Pseudo-random Number (PN) sequence generated by the internal LUT/generator.

15	PN_RST	I	Digital input. Reset PN Sequence to DFF
16	PN_SEL	I	Digital input. 2 MUX 1 for input XOR feedback. The XOR accepts Q of last DFF and one nearest or one furthest from the last DFF
17	Seed_SEL	I	Digital input. Selector for seed input in PN
18	Seed_IN	I	Digital input. Seed input for 1 DEMUX 2

III. Implementation and Results

A. Schematics and Pre-Layout Simulation

For PN Sequence Generator, we implemented several components, including logic gates (NAND 2-input, NAND 3-input) and logic circuits (Mux 2-to-1, and a D Flip Flop with a reset port. The circuits are designed and simulated using xschem and ngspice.

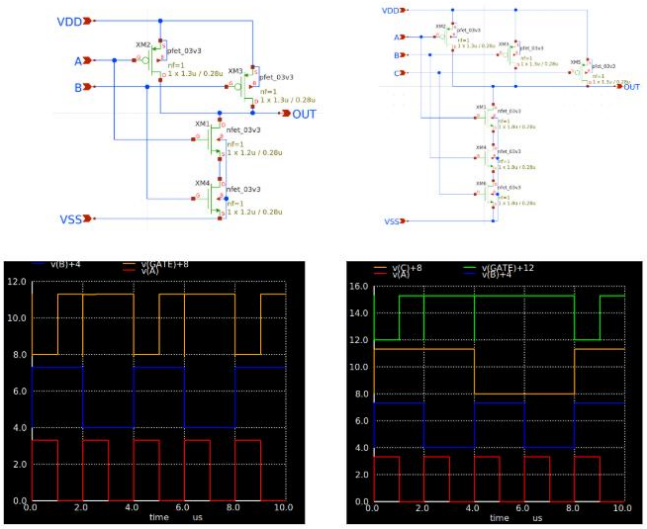

Fig. 4. NAND-2in, NAND-3-in Schematics and Simulation

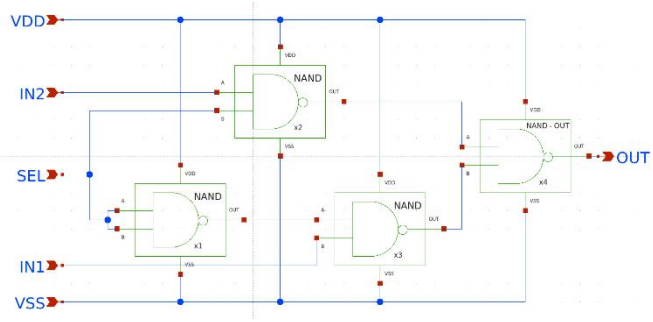

Fig. 5. MUX 2-to-1 Schematic

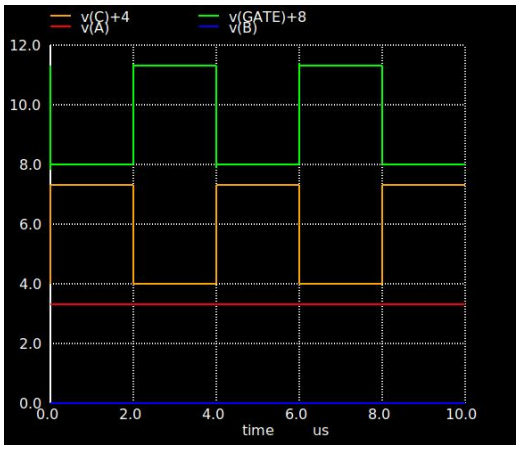

Fig. 6. MUX 2-to-1 Simulation

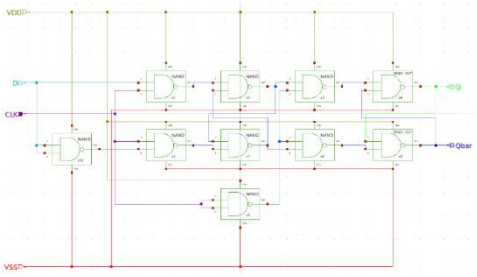

Fig. 7. D Flip Flop Schematic

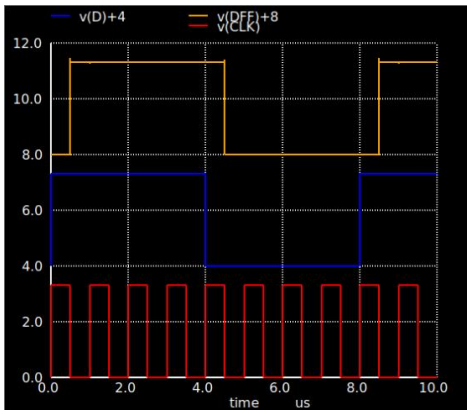

Fig. 8. D Flip Flop Simulation

The resulting simulations have shown that each component is working as intended, with minimal signal fluctuations, showing excellent cutoffs. With these results, we can then move into designing and simulating the PN Sequence Generator.

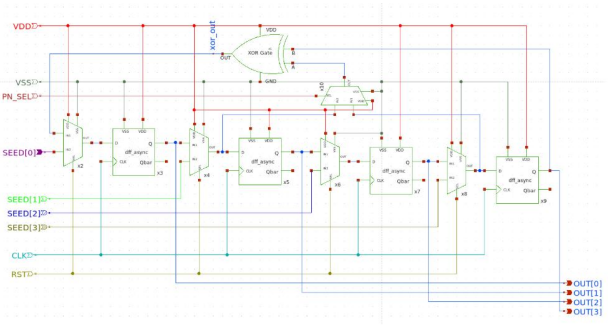

Fig. 9. PN Sequence Generator Schematic

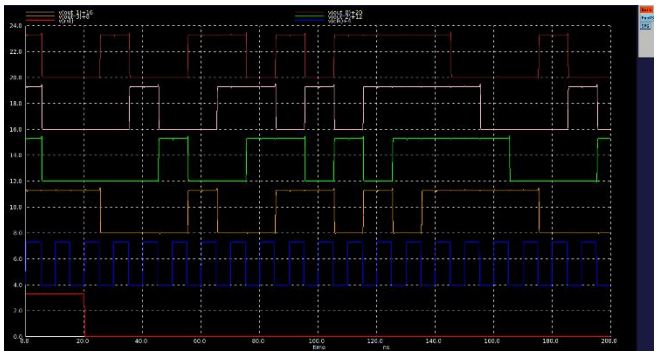

Fig. 10. PN Sequence Generator Simulation Result

The shown result is expected with each bit of the OUT signal (OUT[0..3]) successfully showing an LFSR pattern signal. Each clock cycle, the signal is then shifted into the next bit, forming a pseudo-randomized 4-bit number output. While the sequence will be perfectly deterministic (with the initial seed being the same), the number will “hop” around (e.g. 8, 4, 2, 9, 12), forming the pseudo-random trajectory. To ensure the output to not just be “0”, we set a SEED signal sized 4-bits to ensure the state transition sequence will start. If later developments show SEED port isn’t possible (to fit the number of pad frame pins), we can connect one of the ports to the VDD, making sure the SEED doesn’t have the value of “0”, ensuring the initialization (Hardwired SEED approach).

Aiming to make a learning experience as per our initial design, we also have set the number “hopping” to be configurable by making the equation configurable (whether by using normal LFSR (using 4 Flip-Flops) and simplified LFSR (using only 2-Flip-Flops) by configuring the input signal PN_SEL.

Using same toolset (xschem and ngspice) we then designed the schematic and simulated the RF Mixer.

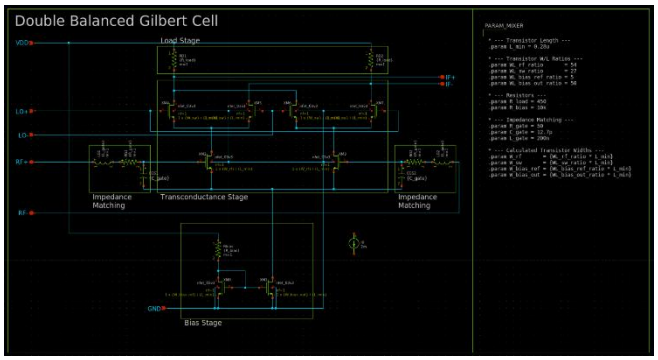

Fig. 11. Double Balanced Gilbert Cell Schematic

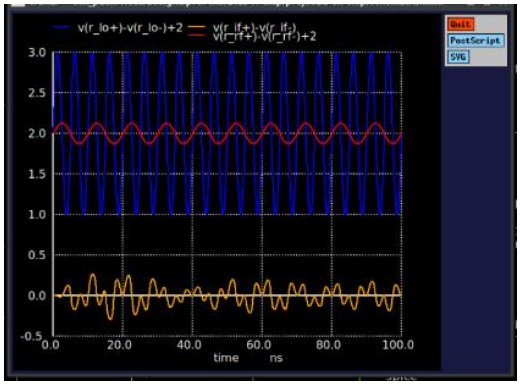

Fig. 12. Double Balanced Gilbert Cell Simulation Result

The mixer operates as a frequency translator, producing an output amplitude determined by the RF input signal strength and the circuit's conversion gain. In this simulation, the mixer demonstrates active gain, as the IF output amplitude (Yellow, 0.5Vpp) exceeds the RF input amplitude (Red, 0.2Vpp). While the mixer relies on the multiplication principle, the LO signal (Blue) functions as a switching drive. Therefore, the output amplitude scales linearly with the RF input but remains largely independent of the LO amplitude, provided the LO drive is sufficient to fully switch the mixer. However, the gain changes to loss when the circuit was implemented with a switch.

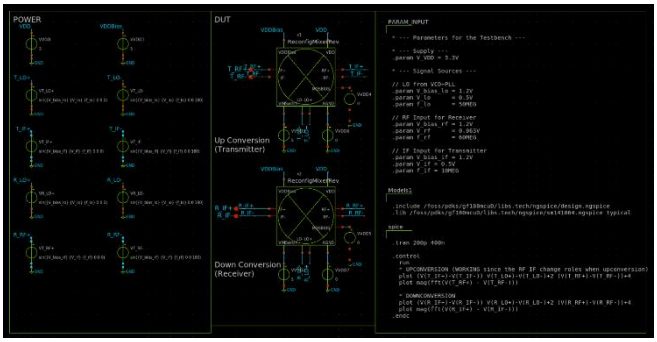

Fig. 13. Top-Level Mixer and Testbench Schematic

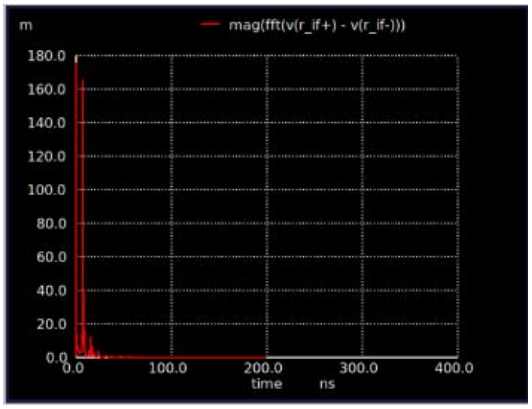

Fig. 14. Top-Level Mixer as Down-Conversion Simulation Result

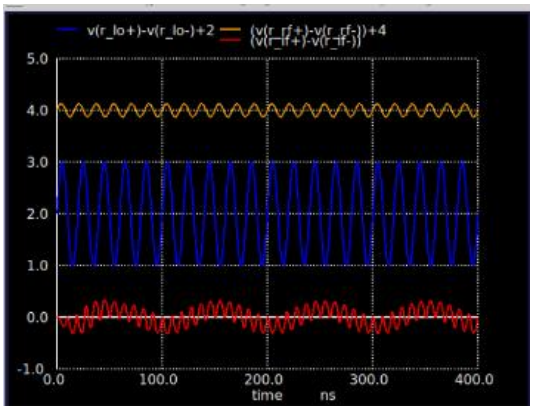

Fig. 15. Top-Level Mixer as Down-Conversion Simulation Result

The first two images illustrate the simulation results for down-conversion. The first image displays the FFT magnitude, effectively acting as a spectrum analyzer to identify the operating frequency. The dominant peak indicates the fundamental IF frequency is located at

approximately 250 MHz (based on the X-axis position), confirming the spectral content of the output. Meanwhile, the second simulation demonstrates the time-domain functionality. Since the output signal (Red trace) has a significantly lower frequency than the RF and LO inputs (Orange and Blue), this confirms the mixer is performing down-conversion. The mixer operates as a multiplier, where the interaction between the two high-frequency input signals produces a different frequency at the output, visible as the slower sinusoidal waveform.

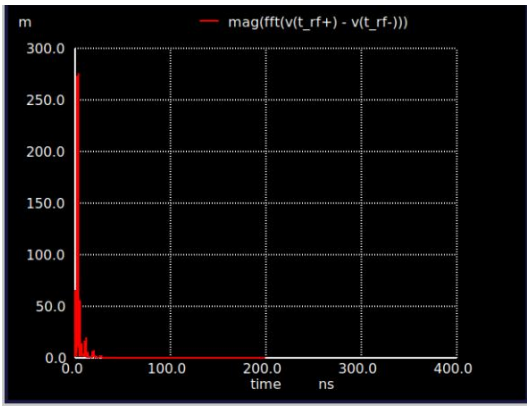

Fig. 16. Top-Level Mixer as Up-Conversion Simulation Result

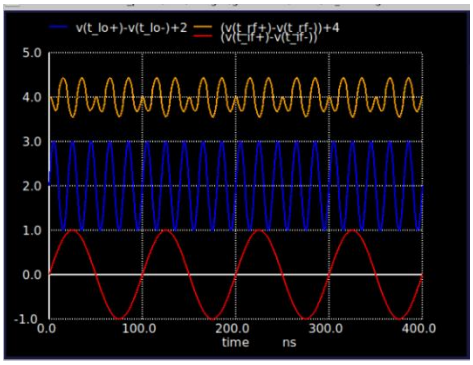

Fig. 17. Top-Level Mixer as Up-Conversion Simulation Result

The last two images demonstrate the mixer's frequency conversion performance. The first image displays the FFT magnitude, effectively acting as a spectrum analyzer. The dominant peak at '4' indicates the operating IF frequency is 1/4n MHz or 250MHz. The second simulation illustrates the time-domain behavior. The presence of two high-frequency phase-shifting inputs (Blue LO and Red IF) producing a slower oscillating output (Orange RF) confirms this is up-conversion. The LO signal (Blue) has a period of approximately 20ns, corresponding to 50 MHz. The mixer functions as a multiplier, mixing the IF and LO signals to generate the difference frequency (RF) observed in the Orange trace.

We then replaced the Bias Stage (mixer tail) of the Mixer with Reconfigurable Mixer Bias Stage by implementing the switch first.

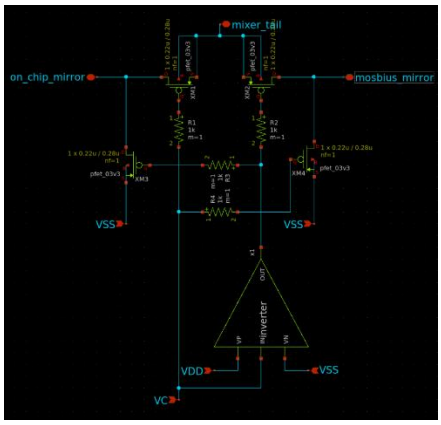

Fig. 18. Reconfigurable Mixer's Bias Stage Schematic

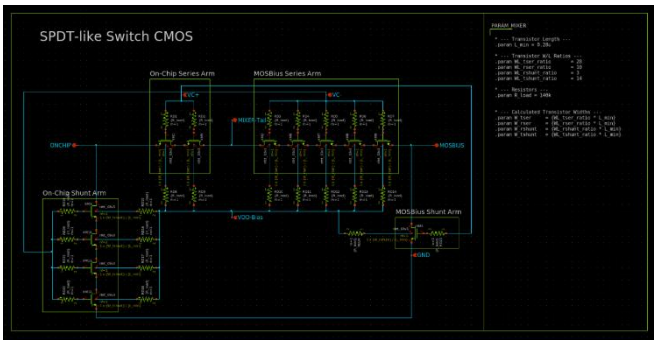

Fig. 19. Reconfigurable Mixer Switch Schematic

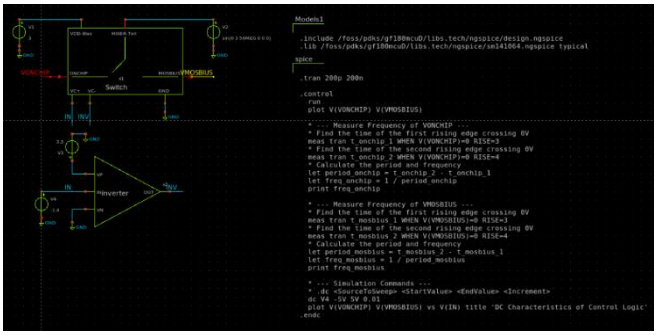

Fig. 20. Reconfigurable Mixer Switch Testbench Schematic

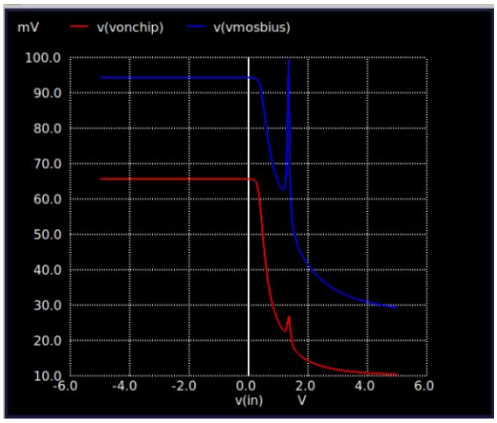

Fig. 21. Reconfigurable Mixer Switch Simulation Result

On 3.3V input to the switch, the signal from on chip that travels to the on chip current tail will be exchanged with current tail from MOSBius. This switch will allow students to integrate their circuit to the existing mixer on chip, making the mixer reconfigurable.

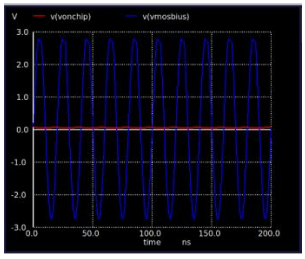

Fig. 22. Reconfigurable Mixer Switch Simulation Result

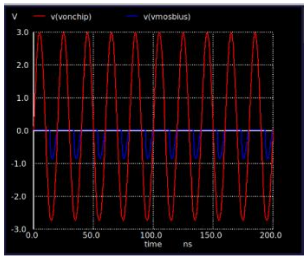

Fig. 23. Reconfigurable Mixer Switch Simulation Result

After successfully implementing the switch, the top-level mixer schematic is implemented and tested using a testbench. The Mixer is tested as a *reconfigurable Up-Conversion Mixer* and *Down-Conversion Mixer*.

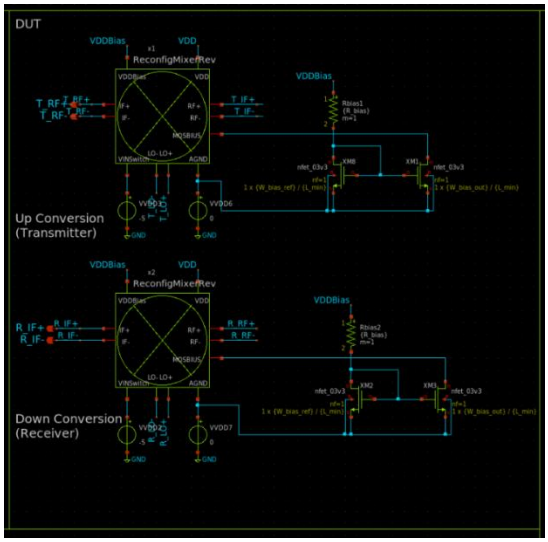

Fig. 24. Final Reconfigurable Mixer Testbench Schematic

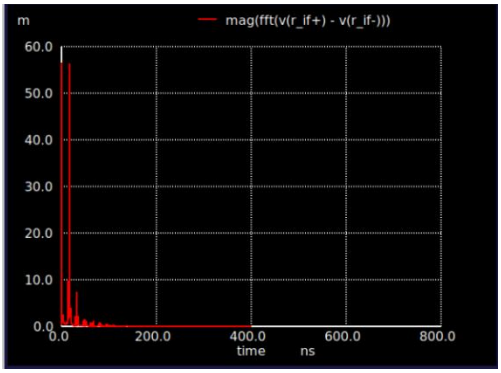

Fig. 25. Top-Level Mixer as Down-Conversion Simulation Result (with MOSBius Connection)

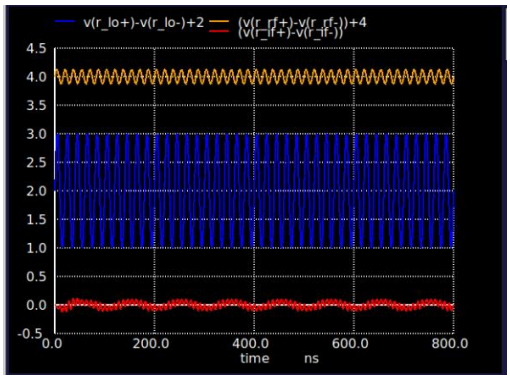

Fig. 26. Top-Level Mixer as Down-Conversion Simulation Result (with MOSBius Connection)

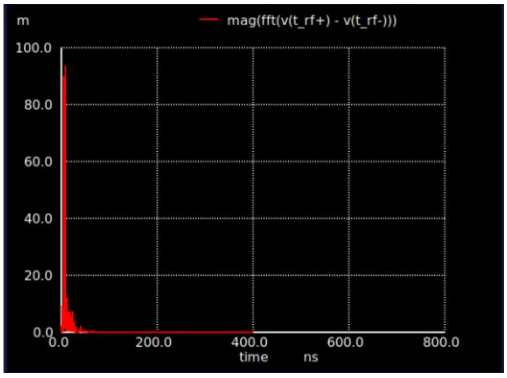

Fig. 27. Top-Level Mixer as Up-Conversion Simulation Result (with MOSBius Connection)

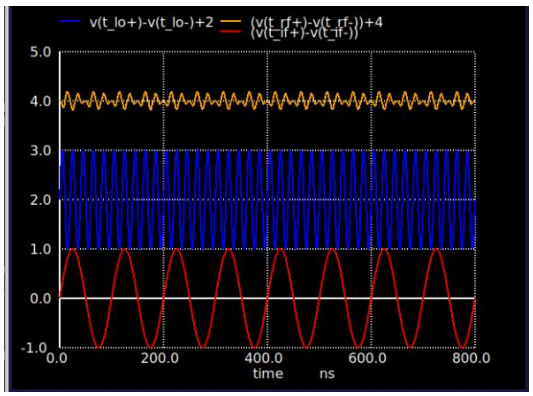

Fig. 28. Top-Level Mixer as Up-Conversion Simulation Result (with MOSBius Connection)

The mixer has insertion loss and additional loss from connecting to the switch. On the mixer itself, the conversion loss is 10 dB. Meanwhile, the additional insertion loss from connecting the mixer to the switch is 8 dB. The loss of insertion might come from the switch topology and sizing itself. This loss could be prevented by proper sizing of transistors and resistors while choosing a more appropriate topology for this use case. Also, the operating frequency drops from the intended 250MHz to 61.7 - 68.7 MHz

B. Layout

After implementing the schematic for the PN Sequence Generator and Reconfigurable Mixer, both circuit layouts are implemented. The implementation of the layout is done using open-source software, *Magic,* and *KLayout*.

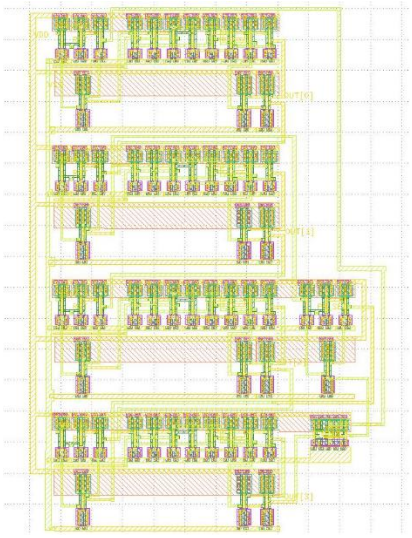

Fig. 29. PN-Sequence Generator Layout

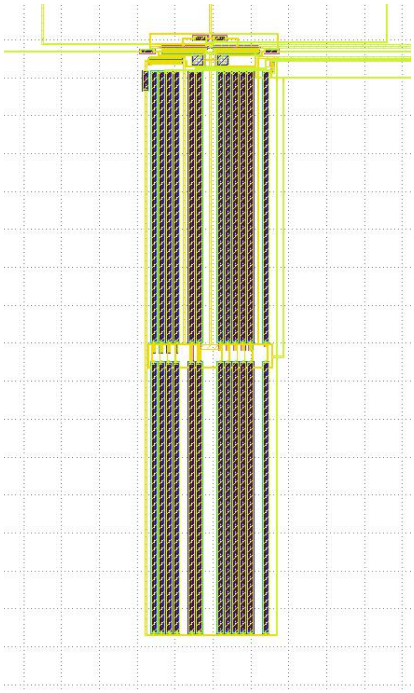

Fig. 30. Reconfigurable Mixer Layout

The core of the mixer is implemented on the top side of the chip, near the pad-ring. This is done to reduce the loss of insertion, maintaining the signal integrity of the mixer. On the lower side of the mixer core, the switch is implemented using a transistor and a long resistor. The long resistor is used for the switch to reduce RF leakage. The resistor acts as a high impedance barrier. However, this comes with a trade-off. The long resistor does lessen the gain and then the available frequency for the operation, limiting it to 61.7 - 68.7 MHz as optimal frequency.

C. Top-Level Tape-out Layout

After implementing the layout for PN Sequence Generator and Reconfigurable Mixer, it runs through post layout simulations (DRC and LVS).

The verification results indicated distinct outcomes for the two blocks. The Reconfigurable Mixer was confirmed to be DRC-clean, successfully satisfying all manufacturing constraints of the GF180MCU process. In contrast, the PN Sequence Generator layout had persistent DRC violations, mainly due to routing congestion within the standard cells (caused by the limitation of metal layers to just 2). For LVS verification, mismatches were found in both designs.

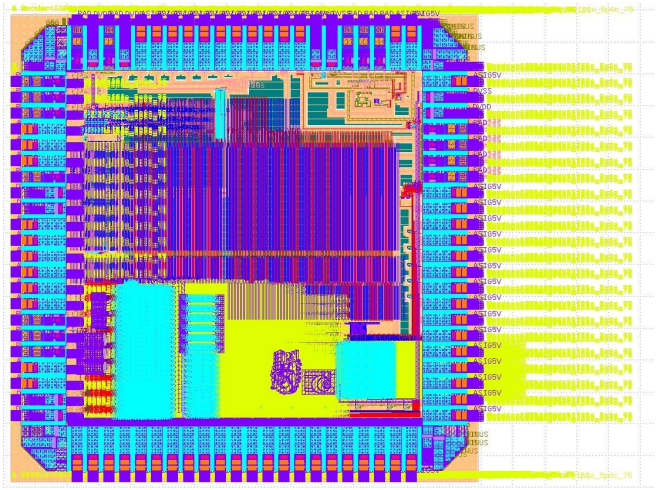

Fig. 31. Final Top-Level Layout for Tape-out

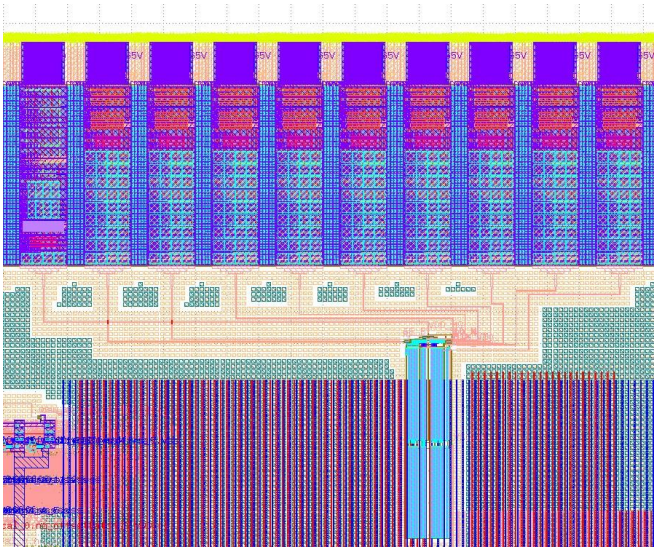

Fig. 32. Final Top-Level Layout for Tape-out (Zoomed to Mixer)

To ensure the chip is manufacturable and meets the strict zero-DRC requirement for submission, we decided to exclude the PN Sequence Generator. This was necessary due to the unresolved DRC errors. Not only that, as the Chipathon fabrication was done on predetermined pads, the numbers of pins available are limited. The inclusion of PN Sequence Generator will need an additional 14-17 pins, which is impossible given our given pin allocation. Therefore, only the Reconfigurable Mixer was included in the final tape out. The frequency hopping control will be handled externally using GPIO connections, so the system remains functional for educational purposes.

IV. CONCLUSIONS

This work successfully demonstrates the design and tape out-ready layout of a Reconfigurable Double-Balanced Gilbert Cell Mixer using GF180MCU technology. The mixer is designed to process an FSK-modulated IF signal at 10.7 MHz and up-convert it to the 61.7-68.7 MHz RF band.

Simulation results indicate that the standalone mixer core achieves a conversion loss of approximately 10 dB. However, the integration of the reconfigurability switch introduced an additional insertion loss of 8 dB, highlighting a trade-off between educational flexibility and signal integrity. Despite this, the Reconfigurable Mixer layout was verified as DRC-clean and passed LVS, qualifying it for the final tape out.

The physical implementation of the PN Sequence Generator encountered significant challenges. While schematic simulations confirmed correct logic functionality, the layout failed to meet manufacturing constraints due to persistent DRC violations caused by routing congestion and pad frame limits. Consequently, the PN Sequence Generator was excluded from the chip, and the frequency hopping control mechanism will be offloaded to external GPIO connections for the current iteration.

Future development will prioritize redesigning the PN Sequence Generator layout to resolve routing issues. Future development will also focus on improving the Reconfigurable Mixer design, specifically by optimizing the switch topology and component sizing to minimize insertion loss and enhance conversion gain. Furthermore, comprehensive layout optimization will be conducted to improve area efficiency and signal integrity, ensuring a more robust and fully integrated performance for the next tape out cycle.

REFERENCES

[1] "Frequency-hopping spread spectrum," *Wikipedia, The Free Encyclopedia*. [Online]. Available: https://en.wikipedia.org/wiki/Frequency-hopping_spread_spectrum.

[2] S. A. Z. Murad, "Design of a high-linearity Gilbert-cell mixer at 5 GHz," *J. Phys.: Conf. Ser.*, vol. 2129, no. 1, p. 012083, Jan. 2022.

[3] Y. Wang, Z. Wu, L. Wang, S. Xiang, and H. Hao, "High-speed pseudo-random bit sequence generation based on a single chaotic semiconductor laser with optical feedback," *Front. Optoelectron.*, vol. 13, no. 2, pp. 182–192, Jun. 2020.

APPENDIX

In the early development phase, a custom Voltage-Controlled Oscillator (VCO) was designed to serve as the frequency source for the Local Oscillator (LO) generation. The circuit was implemented using a 5-stage current-starved ring oscillator topology to allow for frequency tuning via voltage control.

The original intent was to couple this on-chip VCO with a Phase-Locked Loop (PLL) to drive the mixer. However, during the system analysis, we encountered significant challenges in matching the custom VCO's signal specifications with available Commercial Off-The-Shelf (COTS) PLL chips. Due to these integration complexities and the risk of specification mismatch, we decided to exclude the VCO from the final tapeout.

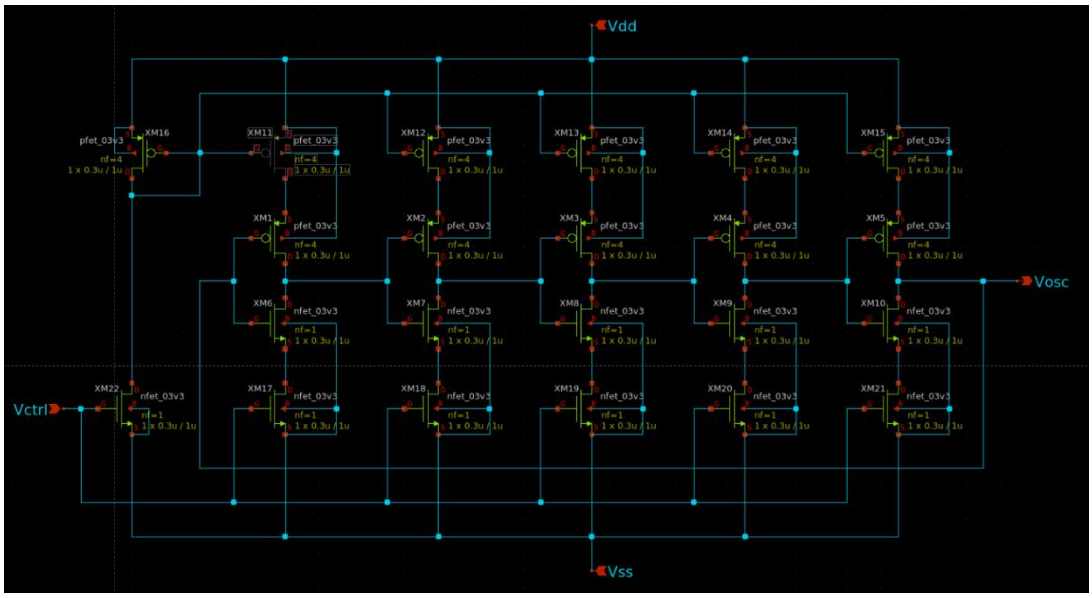

Fig. 33. VCO Schematic Implementation

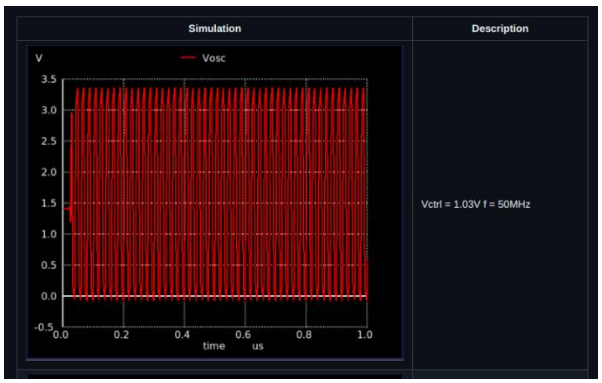

Fig. 34. VCO Schematic Simulation Result

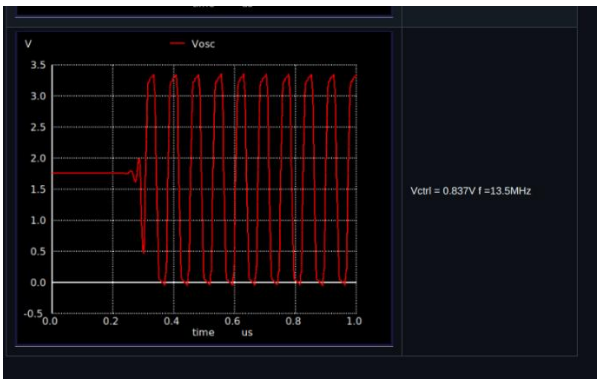

Fig. 35. VCO Schematic Simulation Result