گزارش کار پروژه

پیاده‌سازی و شبیه‌سازی رمز جریان مبتنی بر LFSR با استفاده از ModelSim

1.هدف پروژه

هدف از این پروژه، پیاده‌سازی یک رمز جریان (Stream Cipher) ساده مبتنی بر ثبات‌های انتقال با فیدبک خطی (Linear Feedback Shift Register – LFSR)و بررسی صحت عملکرد آن از طریق شبیه‌سازی در محیط ModelSimمی‌باشد. در این پروژه تلاش شده است با استفاده از چند LFSR مستقل و ترکیب غیرخطی خروجی آن‌ها، یک keystream شبه‌تصادفی تولید شده و عملیات رمزگذاری و رمزگشایی پیام به‌صورت صحیح انجام شود.

2.معرفی رمز جریان وLFSR

رمزهای جریان نوعی از الگوریتم‌های رمزنگاری هستند که در آن‌ها پیام ورودی به‌صورت بیت‌به‌بیت با یک دنباله کلید (keystream) ترکیب می‌شود. یکی از روش‌های رایج برای تولیدkeystream ، استفاده ازLFSR ها است. LFSRیک مدار ترتیبی است که با هر سیکل کلاک، محتویات آن شیفت شده و بیت جدید بر اساس تابع فیدبک تولید می‌شود. استفاده از seed اولیه وtap های مناسب باعث تولید توالی شبه‌تصادفی می‌گردد.

3. ساختار کلی سیستم

سیستم طراحی‌شده در این پروژه از سه بخش اصلی تشکیل شده است:

1. ماژول‌هایLFSR
2. ماژول) Cipher رمزگذاری و رمزگشایی(
3. Testbenchجهت شبیه‌سازی و صحت‌سنجی

هر یک از این بخش‌ها به‌صورت ماژولار و مستقل طراحی شده‌اند.

4. پیاده‌سازی LFSR ها

در این پروژه، سه LFSR مستقل با طول‌های متفاوت پیاده‌سازی شده‌اند:

- LFSRاول با طول 5 بیت
- LFSRدوم با طول 7 بیت
- LFSRسوم با طول 9 بیت

هر LFSR دارای seed اولیه (کلید) مخصوص به خود بوده و به‌صورت همزمان با سیگنال کلاک به‌روزرسانی می‌شود. استفاده از طول‌های متفاوت برایLFSR ها باعث افزایش دوره تناوب keystream و کاهش هم‌بستگی بین خروجی‌ها می‌شود که از نظر امنیتی اهمیت دارد.

5. تولیدKeystream

برای افزایش غیرخطی بودن سیستم، خروجی سه LFSR با استفاده از تابع ترکیب غیرخطی زیر به یک keystreamواحد تبدیل شده است:

$$\text{Keystream} = (a \wedge b) \oplus c$$

که در آن:

- a، bو cخروجی بیت‌های سه LFSR هستند.

این روش ترکیب نسبت به ترکیب‌های خطی ساده، پیچیدگی بیشتری ایجاد کرده و کیفیت keystream را بهبود می‌بخشد.

6. فرآیند رمزگذاری و رمزگشایی

در رمز جریان پیاده‌سازی‌شده:

- رمزگذاری پیام با استفاده از عملگر XOR بین plaintext و keystream انجام می‌شود:

- $\text{Ciphertext} = \text{Plaintext} \oplus \text{Keystream}$
- $\text{Decrypted} = \text{Ciphertext} \oplus \text{Keystream}$

- رمزگشایی نیز با XOR مجدد ciphertext و همان keystream صورت می‌گیرد:

به دلیل خاصیت معکوس‌پذیری عملگرXOR ، پیام اولیه به‌طور کامل بازیابی می‌شود.

7. Testbenchو شبیه‌سازی

برای بررسی صحت عملکرد سیستم، یک Testbench کامل طراحی شده است که وظایف زیر را بر عهده دارد:

- تولید سیگنال‌های clock وreset
- مقداردهی اولیه کلیدهایLFSR
- اعمال یک پیام نمونه ۸ بیتی به سیستم
- بررسی خودکار تطابق خروجی رمزگشایی‌شده با پیام اصلی

در صورت تطابق کامل، پیام زیر در Transcript نمایش داده می‌شود:

```
PASS: decrypted matches plaintext for all bits.
```

پس از پایان تست، شبیه‌سازی با دستور `$stop` متوقف می‌شود تا امکان مشاهده دقیقwaveform ها فراهم گردد.

8.نتایج شبیه‌سازی

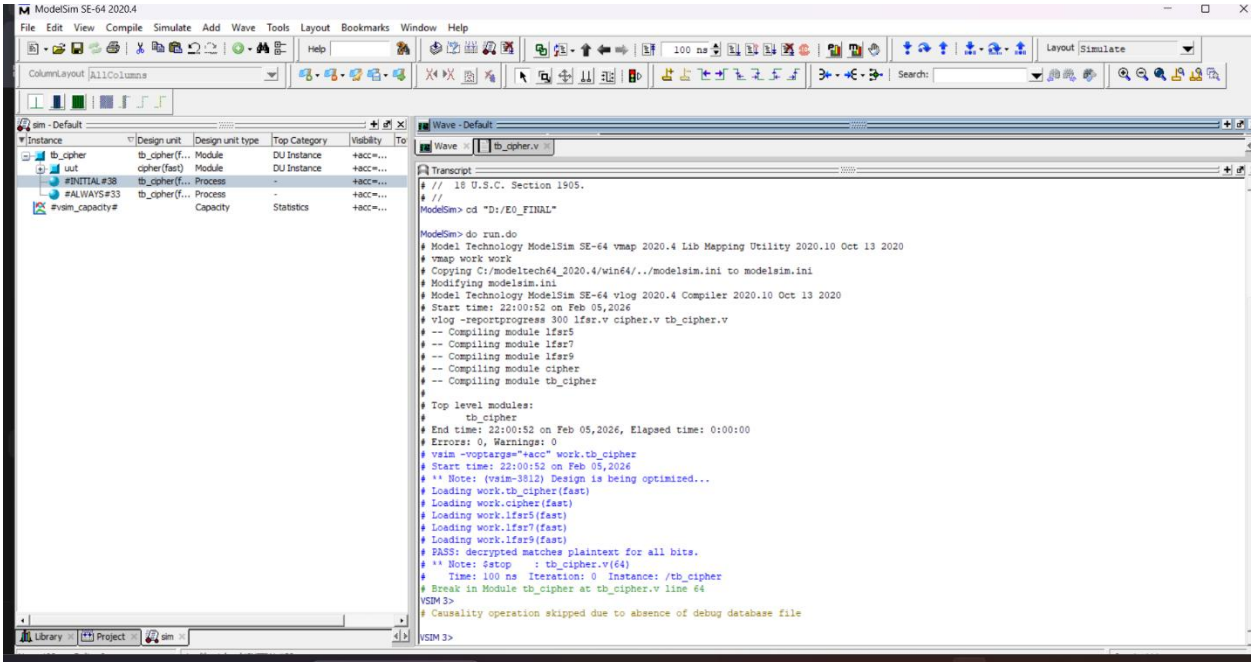
نتایج شبیه‌سازی در محیط ModelSim نشان می‌دهد:

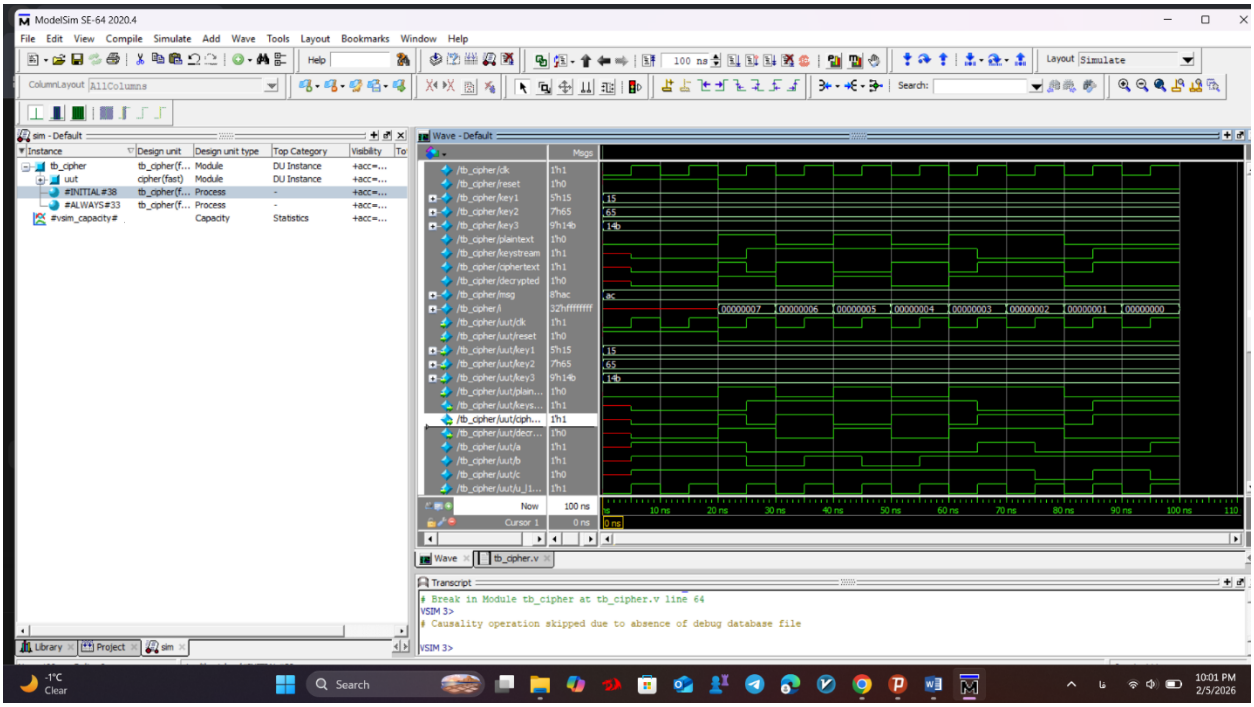
- keystreamبه‌درستی تولید شده است.
- عملیات رمزگذاری و رمزگشایی بدون خطا انجام می‌شود.
- پیام رمزگشایی‌شده در تمام بیت‌ها دقیقاً برابر پیام اولیه است.

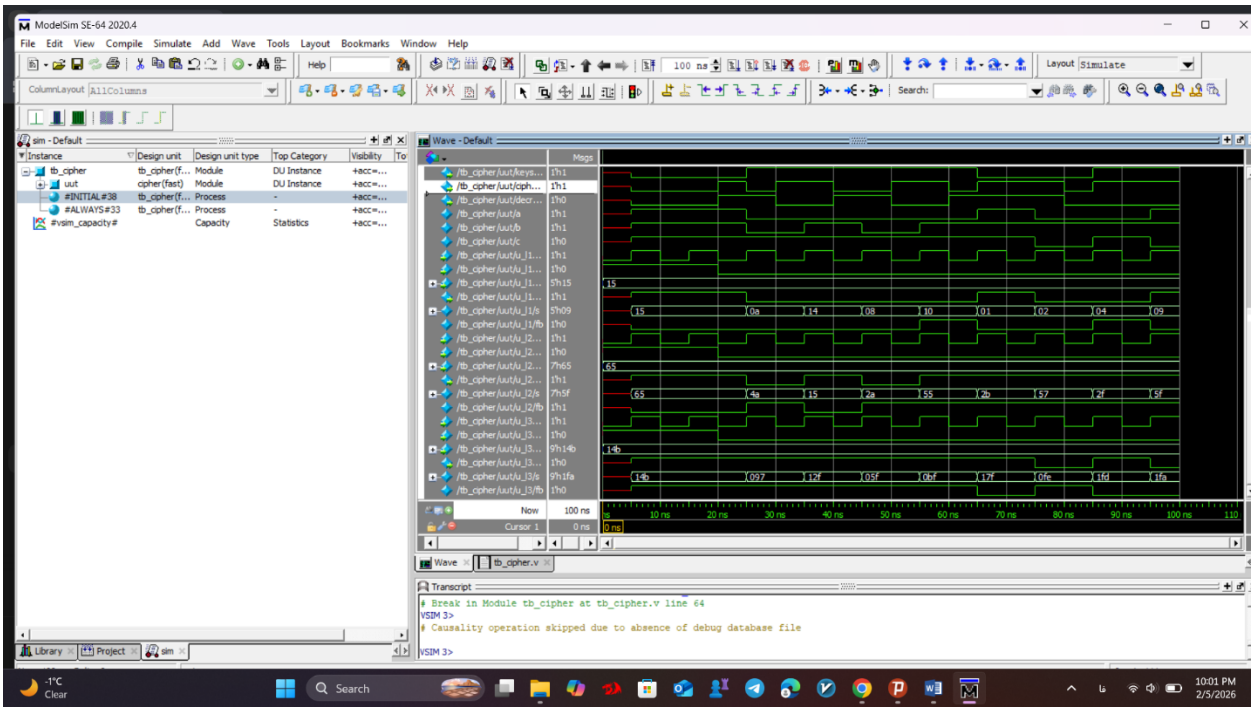
Waveformهای ثبت‌شده تغییرات سیگنال‌های کلیدی شاملplaintext ، keystream، ciphertextو decryptedرا به‌وضوح نمایش می‌دهند و صحت عملکرد سیستم را تأیید می‌کنند.

9.جمع‌بندی

در این پروژه، یک رمز جریان مبتنی بر چند LFSR مستقل با موفقیت پیاده‌سازی و شبیه‌سازی شد. نتایج شبیه‌سازی نشان داد که سیستم طراحی‌شده قادر به رمزگذاری و رمزگشایی صحیح پیام بوده و اهداف پروژه به‌طور کامل تحقق یافته است. استفاده ازLFSR های با طول متفاوت و ترکیب غیرخطی خروجی‌ها باعث بهبود کیفیت keystreamتولیدی شده است.




ModelSim SE-64 2020.4
File Edit View Compile Simulate Add Wave Tools Layout Bookmarks Window Help
sim - Default
Wave - Default
Transcript
Break in Module tb_cipher at tb_cipher.v line 64
VSIM 3>
Causality operation skipped due to absence of debug database file
VSIM 3>
Search
10:01 PM
2/5/2026


ModelSim SE-64 2020.4
File Edit View Compile Simulate Add Wave Tools Layout Bookmarks Window Help
sim - Default
Wave - Default
Transcript
Break in Module tb_cipher at tb_cipher.v line 64
VSIM 3>
Causality operation skipped due to absence of debug database file
VSIM 3>
Search
10:01 PM
2/5/2026