پروژه : پیاده‌سازی نسخهٔ ساده‌شدهٔ E0 Stream Cipher با Verilog

در این پروژه ، هدف آشنایی دانشجویان با مفاهیم رمزنگاری سخت‌افزاری ، تولید دنبالهٔ رمز (Keystream) و پیاده‌سازی یک Stream Cipherساده با زبان Verilog است .الگوریتم مورد استفاده نسخهٔ ساده‌شده ای از رمزنگار E0 در بلوتوث است که بر پایهٔ استفاده از چند LFSR و یک تابع غیرخطی ترکیبی کار می‌کند.

هدف پروژه

پیاده‌سازی یک رمزنگار جریانی ساده که :

- با استفاده از چند LFSR مستقل ، یک Keystream تولید می‌کند.
- این Keystream با دادهٔ اصلی XOR می‌شود و خروجی رمز شده (ciphertext) تولید می‌شود.
- با استفاده از همان Keystream ، مجدداً روی ciphertext عملیات XOR انجام می‌شود و پیام اصلی بازیابی می‌گردد. (decryption)

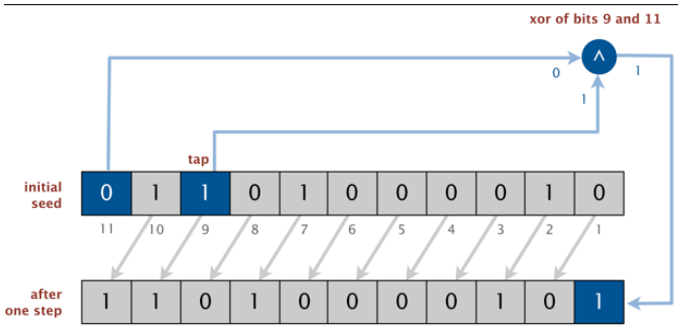
شرح کلی عملکرد

در این پروژه سه یا چهار LFSR مستقل در نظر گرفته می‌شود و هر LFSR دارای طول متفاوت و تابع بازخورد مخصوص به خود است . در هر سیکل کلاک ، هر LFSR یک بیت خروجی تولید می‌کند .خروجی این رجیستر ها وارد یک تابع ترکیب کننده می‌شود . برای ساده سازی، تابع غیرخطی پیشنهادی به صورت زیر در نظر گرفته می‌شود:

keystream_bit = (a & b) ^ c

که a, b و c بیت‌های خروجی از LFSR ها هستند .این تابع ساده است اما غیرخطی بوده و مفاهیم واقعی E0 را شبیه‌سازی می‌کند .بیت keystream تولید شده با بیت پیام اصلی XOR می‌شود تا ciphertext تولید گردد .در مرحلهٔ رمزگشایی نیز همین keystreamدوباره استفاده شده و پیام اصلی بازیابی می‌شود.

LFSR:



one step of an 11-bit LFSR with initial seed 01101000010

LFSR یک شیفت رجیستر است که در هر کلاک ، بیت جدید را از طریق XOR چند بیت قبلی تولید می‌کند. به جای اینکه ورودی از بیرون بیاید، خود رجیستر مقدار جدیدش را می‌سازد و به این دلیل می‌تواند یک دنباله‌ی شبه‌تصادفی تولید کند. اگر ضرایب XOR درست انتخاب شوند، دنباله بسیار طولانی و غیرتکراری خواهد بود. به همین دلیل LFSR در رمزنگاری، تولید کلید، تست سخت‌افزار و تولید اعداد تصادفی کاربرد دارد.

ورودی‌ها و خروجی‌ها

ورودی‌ها :
clock
reset
key - اولیه برای مقداردهی LFSR ها
plaintext - پیام ورودی برای رمز شدن

خروجی‌ها:
keystream
داده رمز شده (ciphertext)
داده رمزگشایی‌شده برای تست صحت

مراحل پیشنهادی

طراحی LFSR با امکان مقداردهی اولیه (seed/key)
طراحی سه یا چهار LFSR مستقل با پلی نوم های متفاوت
پیاده سازی تابع ترکیبی ساده برای ایجاد keystream
تولید cipher با XOR کردن keystream و داده ورودی
رمزگشایی و بازیابی دوباره پیام اصلی
ساخت Testbench در ModelSim جهت مشاهده keystream و بررسی صحت خروجی

خروجی نهایی پروژه

دانشجو باید فایل Verilog شامل LFSR ها، تابع ترکیبی و رمزنگار ، یک Testbench برای تست عملکرد ، و اجرای شبیه‌سازی در ModelSim را ارائه دهد .نتایج باید نشان دهد که پیام اصلی پس از رمزگذاری و سپس رمزگشایی با همان کلید، بازیابی می‌شود.

مثال ساده

plaintext = 10101100
keystream = 01100101
cipher = plaintext (XOR) keystream = 11001001
decrypt = cipher (XOR) keystream = 10101100

توجه : تعداد LFSR های استفاده شده و اندازه شان و همچنین تابع ترکیب استفاده شده بر عهده خودتان می باشد.

برای مطالعه بیشتر می توانید بررسی کنید که چرا از ترکیب های غیرخطی استفاده می شود؟ اگر از ترکیب های خطی استفاده شود چه حملاتی می شود اجرا کرد ؟
و تابع ترکیب پیاده سازی شده در E0 چطور کار می کند ؟