Profa. Michele Nogueira DCC/ ICEx / UFMG

DCC114 - Introdução aos Sistemas Lógicos 2023.2

Trabalho Prático em Verilog

Lucca Alvarenga de Magalhães Pinto - lucca.alvarenga@dcc.ufmg.br

Matrícula: 2021036736

1. Introdução

Este trabalho aborda a implementação em Verilog de um técnica de criptografia chamada “One-Time Pad”, derivado da cifra de Vernam. Esse método utiliza a representação binária de uma mensagem a ser codificada, e uma chave binária de mesmo tamanho, e a codificação é realizada pela operação XOR (ou exclusivo) bit a bit. Dessa forma, no caso em que a chave tiver um número de bits menor que a mensagem, é necessário criar uma nova chave de mesmo tamanho que a mensagem, para que cada elemento do texto a ser criptografado tenha um correspondente na chave, assim podendo realizar a operação XOR.

O projeto foi feito por meio da plataforma “EDAPlayground”. Para testar a codificação e decodificação do método, foram escolhidas algumas mensagem de texto que foram transformada em binário no formato ASCII por meio do site “https://www.rapidtables.com/convert/number/ascii-to-binary.html”. Isso permitiu verificar a corretude da implementação.

2. Implementação

Abaixo é explicado como as principais funções e métodos do projeto foram estruturadas:

- **Flip Flop D:**

Foi implementado um flip-flop D síncrono com uma entrada adicional para reset. Abaixo estão as explicações para os parâmetros e operações:

Entradas:

- “input wire clk”: Sinal de clock utilizado para sincronizar as operações do dispositivo.
- “input wire en”: Sinal de habilitação ou desabilitação do flip-flop que controla seu funcionamento.
- “input wire reset”: Sinal que redefine o estado do flip-flop.

Saída:

- “output reg q”: Saída de dados.

Comportamento:

O flip-flop D armazena o valor de entrada "d" na saída "q" em cada transição de subida do sinal de clock "clk", quando o sinal de habilitação "en" está ativo. Caso o sinal de reset esteja ativo, a saída "q" é redefinida para o valor lógico "0".

Especificação Comportamental:

Sempre que ocorrer uma transição de subida no sinal de clock "clk", os seguintes passos serão executados:

1. Verifique se o sinal de "reset" está ativo.
    - Se estiver ativo, define a saída "q" como "0".
    - Caso contrário, continua para o próximo passo.
2. Verifique se o sinal de habilitação "en" está ativo.
    - Se estiver ativo, define a saída "q" como o valor presente na entrada "d".
    - Caso contrário, mantém o valor atual da saída "q".

A implementação do código em Verilog está disponível em:

https://edaplayground.com/x/pxdV

- **Porta XOR:**

Módulo xor_gate (Porta XOR):

- O módulo define uma porta XOR com entradas “a” e “b” e saída “y”.
- A implementação estrutural da porta XOR é feita usando portas NOT, AND e OR.
- Outra forma alternativa seria usando a implementação comportamental, operação XOR (assign y = a ^ b).

Módulo xorgate_tb (Testbench para a Porta XOR):

- Declaração de sinais de teste (A, B, Y) para verificar a porta XOR.
- Instância da porta XOR (xor_gate).
- Inicialização do simulador, definindo o arquivo de saída VCD ($dumpfile) para visualização no GTKWave.
- Monitoramento de variáveis durante a simulação usando $monitor.
- Sequência de testes de entrada (A e B) com delays (#n) entre eles.
- Exibição do tempo de simulação e valores de entrada/saída usando $display.
- A simulação termina após um atraso de 5 unidades de tempo (#5), e uma mensagem "fim" é exibida.

O teste verifica o comportamento da porta XOR para diferentes combinações de entradas a e b ao longo do tempo de simulação. A implementação do código em Verilog está disponível em: https://edaplayground.com/x/vCVa.

- **Shift_Register:**

Foi implementado em Verilog um registrador de deslocamento (shift register) e um módulo de teste associado. Abaixo é explicado as partes principais do código:

Módulo ShiftRegister:

Este módulo descreve um registrador de deslocamento de 8 bits. O registrador é sensível à borda de subida do sinal de clock (posedge clock). Quando

há uma transição de 0 para 1 no sinal de clock, o novo bit (novoBit) é carregado no bit mais significativo (sequencia[0]). Em seguida, os bits restantes (sequencia[1] até sequencia[7]) são deslocados para a direita, mantendo a informação anterior.

Módulo TesteShiftRegister:

Neste módulo de teste um sinal de clock e um sinal de novo bit são declarados como registradores, e é definida uma saída de 8 bits. O módulo “ShiftRegister” é instanciado e conectado aos sinais de entrada e saída.

O bloco “initial” contém duas iterações. Na primeira, novoBit é configurado como 0, e o clock é acionado 8 vezes. Na segunda iteração, novoBit é configurado como 1, e novamente o clock é acionado 8 vezes. Em cada iteração, o valor atual da saída é monitorado a cada duas unidades de tempo (#2) usando a tarefa “minotor”. A implementação do código em Verilog está disponível em: https://edaplayground.com/x/aQPx.

- **One_Time_Pad 1 (chave e palavra com 32 bits):**

Primeiramente foi feito um código mais simples em Verilog implementando o "One-Time Pad" (OTP) com a chave e palavra de tamanhos iguais:

Módulo testeOneTimePad:

Este módulo de teste define dois módulos, um para cifrar e outro para decifrar. A mensagem original e seu tamanho são definidos em “definicao.v”. O resultado cifrado e decifrado é exibido usando o “$display”. A espera de um ciclo (#1) é adicionada para garantir que as operações dentro dos módulos OneTimePad tenham tempo para serem concluídas.

Módulo OneTimePad:

Este módulo implementa a lógica do One-Time Pad. A mensagem original é fornecida como entrada (mensagem), e a mensagem cifrada é produzida através da operação XOR (^) com uma chave dada. O resultado é atribuído à saída mensagemCifrada.

A chave (cifra) é uma constante pré-definida, no entanto, na realidade uma chave OTP deveria ser verdadeiramente aleatória. Neste exemplo, a chave é definida diretamente no código em “definicao.v” para facilitar a simulação do usuário. No código fornecido, não há explicitamente um uso direto de shift registers ou flip-flops que serão mostrados de forma conjuntas mais à frente.

A implementação do código em Verilog está disponível em: https://edaplayground.com/x/8bBJ.

- **One_Time_Pad 2 (chave com 8 bits e mensagem com 96 bits):**

A implementação é igual ao OTP 1 mas com a chave menor que a mensagem. Foi criado uma “chave_repetida” em que a cifra é repetida até o momento em que cada elemento da mensagem tenha um correspondente na chave. O código em Verilog está disponível em: https://edaplayground.com/x/Nrd4.

- **One_Time_Pad 3 (chave com 8 bits e mensagem com 328 bits):**

A implementação é igual ao OTP 2 mas com um exemplo de mensagem ainda maior. O código em Verilog está disponível em: https://edaplayground.com/x/Pm4Y.

- **One Time Pad FINAL:**

Com base nos códigos anteriores, nesse OTP foi implementado um cifrador e decifrador com base em um deslocador de bits (shifter) para gerar a chave:

design.sv - Módulo OneTimePad:

Este módulo realiza a operação XOR entre a mensagem e a chave OTP. Possui 3 entradas: “clk”(sinal do relógio), “otp”(Chave One-Time Pad) e a “mensagem” a ser cifrada ou decifrada. A saída é o “resultadoCifrado”, ou seja, o resultado da operação XOR, em que cada bit da mensagem é combinado com o bit correspondente da chave usando o operador “ou exclusivo”.

testbench.sv - Módulo TesteOneTimePad

Este módulo de teste é o qual instância o shifter e dois módulos OneTimePad (um para cifrar e outro para decifrar). Possui 5 sinais de controle: “clk”(sinal de

relógio), “reset” (sinal que redefine o estado), “enable” (sinal de habilitação), “seed” (semente inicial para o deslocador) e a mensagem de entrada. Além disso, possui 3 sinais de saída: “resultadoCifrado”, “resultadoDescifrado” e “otp_out” (saída do deslocador, chave OTP).

shifter.v - Módulo shifter

Este módulo é referente ao deslocador de bits que gera a chave OTP. Possui 4 entradas: “clk” (sinal de clock), “reset”, “en” (sinal de habilitação), “seed” (semente geradora); uma saída gerada pelo deslocamento “out”, e uma variável “shift_reg” referente ao registrador de deslocamento. O registrador é inicializado com a semente quando há um sinal de reset. A cada borda de subida do “clock”, se habilitado, o registrador é deslocado em 1 byte.

definicao.v

Este arquivo contém definições de tamanho e textos codificados em ASCII usados para testar o método. O código em Verilog está disponível em: https://edaplayground.com/x/Egys.

- **One Time Pad wave:**

Para melhor visualização do gráfico gerado pelo OTP em Verilog: https://edaplayground.com/x/UJuw

3. Conclusão

O desenvolvimento deste projeto proporcionou uma visão nos conceitos da lógica combinatória, sequencial relacionados para implementação de um algoritmo de criptografia. O trabalho permitiu a visualização prática das matérias vistas na disciplina por meio da linguagem Verilog. O objetivo principal de implementar um One Time Pad para codificação e descodificação de mensagens binárias foi alcançado e, ao final, o código elaborado obteve êxito ao executar de forma eficiente a manipulação dessas mensagens. Nesse cenário, a divisão do trabalho em pequenas partes e a simulação de cada uma por meio de vários testes se mostrou bem vantajoso, garantindo a correção do código.

Referências

AMD. Efficient Shift Registers, LFSR Counters, and Long Pseudo-Random Sequence Generators. Disponível em: https://docs.xilinx.com/v/u/en-US/xapp052. Acesso em: 05/12/2023.

Crypto Museum. One Time Pad Encryption: the unbreakable encryption method. Disponível em: https://www.cryptomuseum.com/manuf/mils/files/mils_otp_proof.pdf. Acesso em: 04/12/2023.

GeeksforGeeks. Implementation of Vernam Cipher or One Time Pad Algorithm. Disponível em: https://www.geeksforgeeks.org/implementation-of-vernam-cipher-or-one-time-pad-algorithm/. Acesso em: 04/12/2023.

Michele Nogueira. Slides virtuais da disciplina de Introdução aos Sistemas Lógicos. Disponibilizado via moodle. Departamento de Ciência da Computação. Universidade Federal de Minas Gerais. Belo Horizonte.

Wikipedia. Linear Feedback Shift Register. Disponível em: https://en.wikipedia.org/wiki/Linear-feedback_shift_register. Acesso em: 05/12/2023.

Wikipedia. One Time Pad. Disponível em: https://pt.wikipedia.org/wiki/One-time_pad. Acesso em: 05/12/2023.

Wiki IME/USP. One-Time Pad. Disponível em: https://wiki.imesec.ime.usp.br/books/ctf-starter-pack/page/one-time-pad. Acesso em: 04/12/2023.

Apêndice - Anexos

Todos os links com as implementações dos códigos em Verilog:

- **Flip_Flop_D:**

https://edaplayground.com/x/pxdV

- **XOR_Gate:**

https://edaplayground.com/x/vCVa

- **Shift_Register:**

https://edaplayground.com/x/aQPx

- **One_Time_Pad (chave e palavra com 32 bits):**

https://edaplayground.com/x/8bBJ

- **One_Time_Pad (chave com 8 bits e mensagem com 96 bits):**

https://edaplayground.com/x/Nrd4

- **One_Time_Pad (chave com 8 bits e mensagem com 328 bits):**

https://edaplayground.com/x/Pm4Y

- **One Time Pad_Final:**

https://edaplayground.com/x/Egys

- **One Time Pad wave:**

https://edaplayground.com/x/UJuw