

Trabalho Prático

Multiplicador Binário Sequencial

Aline Cristina Pinto
2020031412

Universidade Federal de Minas Gerais (UFMG)
Belo Horizonte - MG - Brasil

alinecristinapinto@ufmg.br

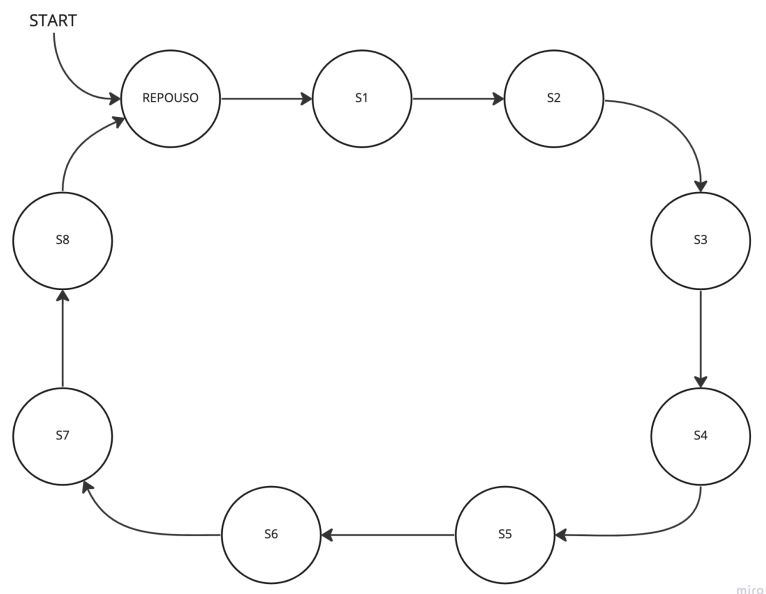
1. Introdução

Este trabalho possui como objetivo implementar um *Multiplicador Binário Sequencial* de 8x8 bits (unsigned), uma versão mais eficiente da multiplicação de números binários tradicional. Para descrever esse sistema digital, foram utilizadas a linguagem [Verilog e o EDA Playground](#) como ambiente de desenvolvimento. Nessa documentação abordaremos as decisões de projeto guiadas pelo diagrama de *Máquina de Estado Finita*, além da apresentação dos testes realizados.

2. Especificação

2.1. Diagrama de Máquina de Estado Finita

Dada a especificação do funcionamento de um multiplicador sequencial de 8x8 bits, o seguinte diagrama foi criado, utilizando a *Máquina de Moore*:



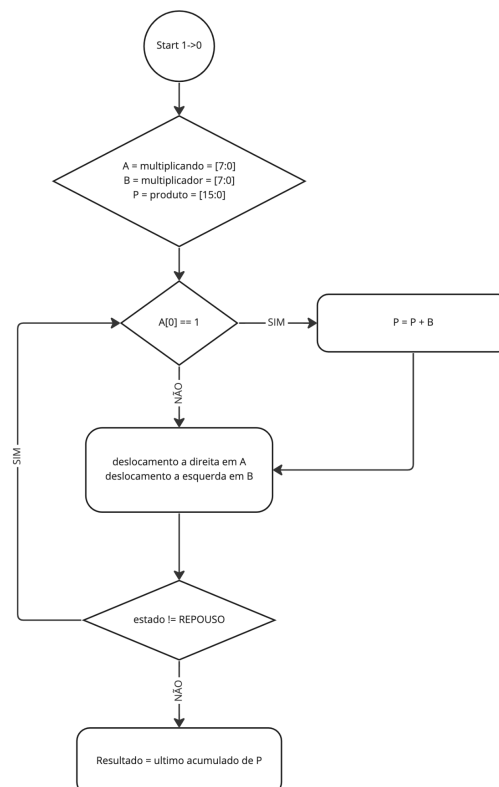
O estado de repouso é referente ao estado de espera do início dos cálculos, no qual os registradores são preenchidos com os valores das entradas do multiplicando e multiplicador e o produto é zerado. Esse estado é acessado após o término do estado S8, ou quando há o disparo da entrada *start*, que indica o início ou o reset de uma multiplicação.

Os estados seguintes, nomeados de S1 a S8, são referentes às mudanças do clock (que nesse projeto é sensível a borda de subida) para o cálculo do produto parcial e dos deslocamentos. Como o multiplicador é de 8 bits, foram necessários oito estados para garantir o deslocamento de todos os bits dos registradores. A mudança de estado entre S1 a S8 ocorre independente das entradas.

2.2. Implementação

A partir do diagrama FSM obtido e da criação do [fluxo de funcionamento](#) do sistema, a escolha de seguir com a primeira implementação de um multiplicador sequencial proposta pelo trabalho foi natural.

Durante o desenvolvimento, foram levantadas outras possíveis formas de implementação, tanto da parte de resumir a FSM a estados de "adição" e "deslocamento" ao invés dos S1 ... S2, quanto a outras abordagens a nível de código, como o uso de enums para a descrição dos estados. Entretanto, devido ao tempo e algumas [incertezas](#) quanto ao uso da linguagem, seguir com a implementação do multiplicador padrão se mostrou mais seguro e robusto para os casos de testes pensados.

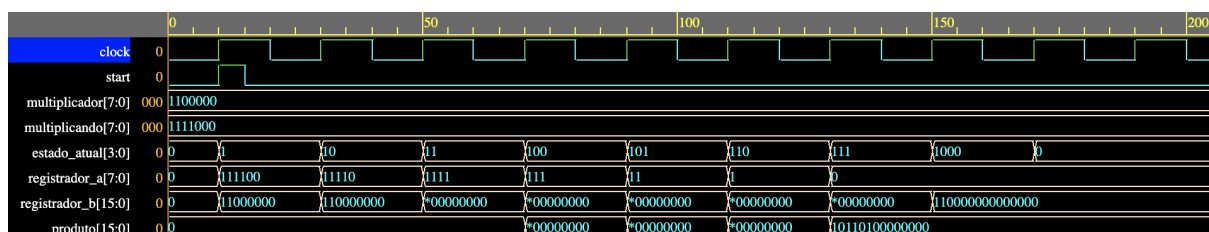


2.3. Casos de teste

Para validar o comportamento do multiplicador sequencial desenvolvido, alguns parâmetros foram observados:

- Corretude do produto após a oitava subida do clock;
- A inicialização da multiplicação **apenas** após um pulso sensível a subida para a entrada *start*;

Abaixo, podemos ver o caso da multiplicação de $120 \times 96 = 11520$:



Como pode ser observado, o multiplicador e o multiplicado são preenchidos com 1100000_2 (96_{10}) e 1111000_2 (120_{10}) respectivamente.

Enquanto o clock se encontra no nível baixo, o estado atual permanece em repouso (0000_2) e os registradores a e b não recebem os valores da multiplicação.

Após a primeira subida do clock e do start, o estado atual muda para S1. Os registradores são preenchidos com os valores já deslocados. Como os primeiros 4 bits menos significativos de a são 0, não há soma e o produto permanece zerado.

Após oito subidas do sinal de clock, o produto tem armazenado o resultado da operação 10110100000000_2 (11520_{10}) confirmando a validade da conta. Como não houve uma nova subida de sinal no start, os registradores mantêm os valores anteriores armazenados.

3. Referências

Slides virtuais da disciplina de Introdução a Sistemas Lógicos. Disponibilizado via moodle. Departamento de Ciência da Computação. Universidade Federal de Minas Gerais. Belo Horizonte. Acesso em: 12/12/2022

Exemplo prático: Verilog. Disponibilizado via moodle. Departamento de Ciência da Computação. Universidade Federal de Minas Gerais. Belo Horizonte. Acesso em: 12/12/2022