



| | |
|-------------------|--|
| Evento | Salão UFRGS 2018: SIC - XXX SALÃO DE INICIAÇÃO CIENTÍFICA DA UFRGS |
| Ano | 2018 |
| Local | Campus do Vale - UFRGS |
| Título | Comparação de métodos de síntese de funções Booleanas |
| Autor | GABRIEL AMMES PINHO |
| Orientador | RENATO PEREZ RIBAS |

Comparação de métodos de síntese de funções Booleanas
Aluno: Gabriel Ammes Pinho Orientador: Renato Perez Ribas
Universidade Federal do Rio Grande do Sul

Uma função Booleana pode ser sintetizada em uma estrutura de dois níveis, como somas-de-produtos (SOP) ou produto-de-somas (POS), ou em uma expressão fatorada de estrutura multi-nível. Cada método de síntese apresenta vantagens e desvantagens que devem ser consideradas. A escolha de qual método deve ser utilizado está fortemente relacionada com a tecnologia e a topologia do circuito desejado, além de considerar o seu custo computacional. Os métodos abordados serão a síntese baseada em multiplexadores, o algoritmo de Quine-McCluskey, uma expansão ao algoritmo de Quine-McCluskey e classes de funções XOR.

Em vários métodos a entrada será uma tabela verdade de n variáveis. As formas mais diretas de representar uma tabela verdade em uma expressão são: na forma de SOP, na qual é feita a soma dos mintermos, e na forma POS, na qual é feita o produto dos maxtermos. Um multiplexador, ou MUX, é um bloco combinacional que utiliza n sinais de seleção para conectar um dos 2^n sinais de entrada a uma única saída. Um MUX pode ser logicamente representado por uma SOP onde cada cubo corresponde a uma condição de seleção e a respectiva entrada selecionada. É possível implementar uma função de n variáveis utilizando um MUX com 2^{n-1} sinais de entrada e $n-1$ sinais de seleção. Para isto, $n-1$ variáveis são utilizadas como sinais de seleção e os sinais de entrada podem ser a variável restante, com polaridade positiva ou negativa, além de 0 e 1. O algoritmo de Quine-McCluskey é um dos mais difundidos métodos de síntese de funções Booleanas. Este algoritmo encontra uma SOP com o menor número de cubos de uma dada tabela verdade de n variáveis. O algoritmo consiste em dois passos principais: combinar os mintermos, ou cubos, de uma tabela verdade e usá-los para cobrir os mintermos desta função. Os cubos que possuem $n-1$ literais com a mesma polaridade podem ser combinados, gerando um novo cubo que não contém este literal com polaridade diferente. Este novo cubo cobre os dois cubos anteriores, que não precisam ser mais considerados na solução. Esta combinação é feita até que não seja possível combinar nenhum cubo. A cobertura dos mintermos é feita combinando os cubos resultantes para cobrir os mintermos originais. A cobertura que possui o menor número de cubos é a saída do algoritmo. É possível expandir o algoritmo de Quine-McCluskey para considerar operações OU-exclusivo (XOR) na combinação de cubos. Nesta expansão, além de combinar como no algoritmo original onde apenas um literal tem a polaridade diferente, também são combinados cubos que possuem dois literais com polaridade diferente. Caso estes literais tenham polaridades inversas em cada um dos cubos, será feita uma XOR entre estas duas variáveis. Caso contrário, será feita uma XNOR destas duas variáveis. Existem classes de expressões que utilizam operações XOR, onde apresentarei 5 delas. Cada uma destas classes apresenta características diferentes e podem ser utilizadas para sintetizar uma expressão já existente. Para isso, são aplicadas expansões nesta expressão original, o que pode simplificá-la. As classes abordadas são: Expressão de Reed-Muller de Polaridade Positiva, Expressão de Reed-Muller de Polaridade Fixa, Expressão de Kronecker, Expressão Pseudo Reed-Muller e Expressão Pseudo Kronecker. As expansões utilizadas nesta síntese são a Expansão de Davio Positiva, a Expansão de Davio Negativa, e a Expansão de Shannon.

Os resultados iniciais mostram que o método baseado em MUX apresenta um ganho comparando com a SOP e mantendo o custo computacional baixo. O método de Quine-McCluskey consegue o melhor resultado de síntese de dois níveis, mas o custo computacional é mais alto que os métodos anteriores. Entre os métodos que utilizam XOR, destaca-se a extensão do algoritmo de Quine-McCluskey, pois tem um custo computacional menor que algumas das classes e apresenta os melhores resultados entre todos os métodos analisados. Como continuação deste trabalho, será necessário implementar otimizações no algoritmo de Quine-McCluskey e nas classes de expressões XOR, provavelmente na Expressão Pseudo Kronecker, para poder comparar os métodos utilizando funções maiores.