Trabalho 2 - Modelagem de controle de tráfego em SystemC

José Reinaldo da C.S.A.V. da Silva neto
Matrícula: 14/0169148

24 de Abril de 2018

1 Objetivos

neste projeto iremos modelar um cruzamento entre duas pistas (autoestrada e via lateral) sendo a preferência ditada por dois semáforos. O controle do sistema dos dois semáforos será modelado em systemc.

2 Introdução

Um sensor na via lateral nos alerta quando há um carro esperando para atravessar a autoestrada e, neste caso, queremos alterar os sinais dos semáforos permitindo aos carros da via lateral passagem enquanto os carros da autoestrada esperam. Percebe-se que este problema é facilmente modelado por uma máquina de espaço de estados finita e determinística.

Tal máquina é representada na figura 1. Onde os estados $S0$, $S1$, $S2$ e $S3$ representam:

Estados	Sem. Autoestrada	Sem. Via lateral
S0	verde	vermelho
S1	amarelo	vermelho
S2	vermelho	verde
S3	vermelho	amarelo

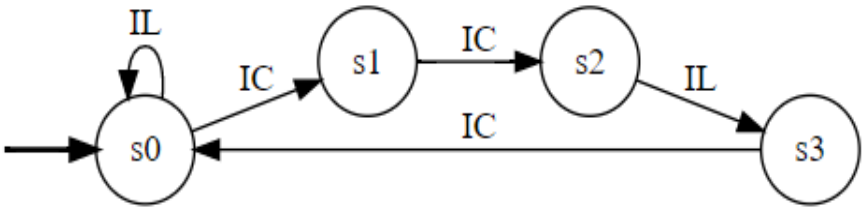

Figure 1: Máquina Espaco de Estados

A transição de $S0$ para $S1$ só é feita quando há carro esperando na via lateral, ou seja, quando o sensor da via lateral estiver com nível lógico 1. Todas as outras transições dependem apenas do tempo em que a máquina ficou em tal estado.

3 Modelagem

Após montada a máquina de estados (arquivos semaforo.h e semaforo.cpp), foi feito um módulo temporizador (timer.h e timer.cpp) que após receber um sinal de ativação da máquina de estados, devolve um pulso após um tempo IL(intervalo longo) ou IC(intervalo curto) dependendo do estado atual. Note que devido ao modo como foi feita a implementação, o intervalo que a máquina permanece em um estado é sempre o tempo que ela de fato deveria ficar neste estado somado de um ciclo de clock. Isto ocorre pois a função de atualização de estados sempre espera um ciclo de clock para devolver o nível lógico zero ao sinal de ativação.

4 Validação do Modelo

Para verificar que o modelo está operando de acordo com as especificações desejadas utilizamos um testbench definido em $semaforo\_tb.h$ e $semaforo\_tb.cpp$. Podemos verificar os resultados pelo resultado apresentado em prompt, assim como pelo software GTKWave. A figura 2 apresenta o resultado do prompt para a implementação com 3 métodos. O tempo IL e IC utilizado para esta simulação foi de 2 e 1 ciclos de clock respectivamente, pode-se notar de acordo com a figura 2 que o tempo de permanência em cada estágio foi um ciclo de clock superior ao que de fato deveria ocorrer, como explicitado na seção Modelagem.

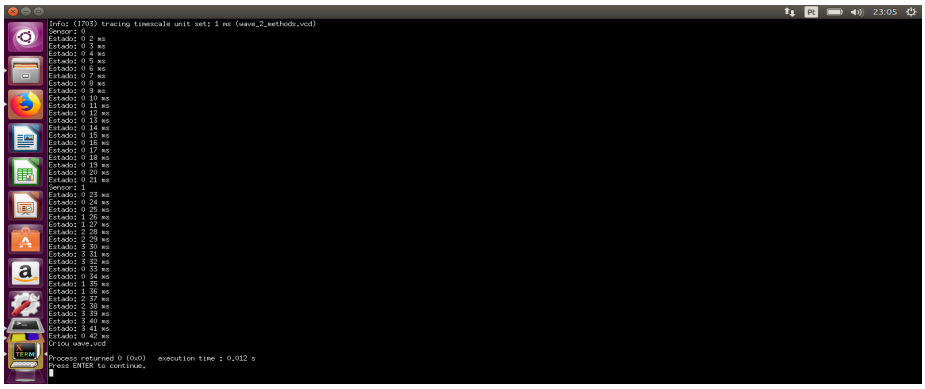

Figure 2: Resultado do Prompt após a simulação com 3 métodos