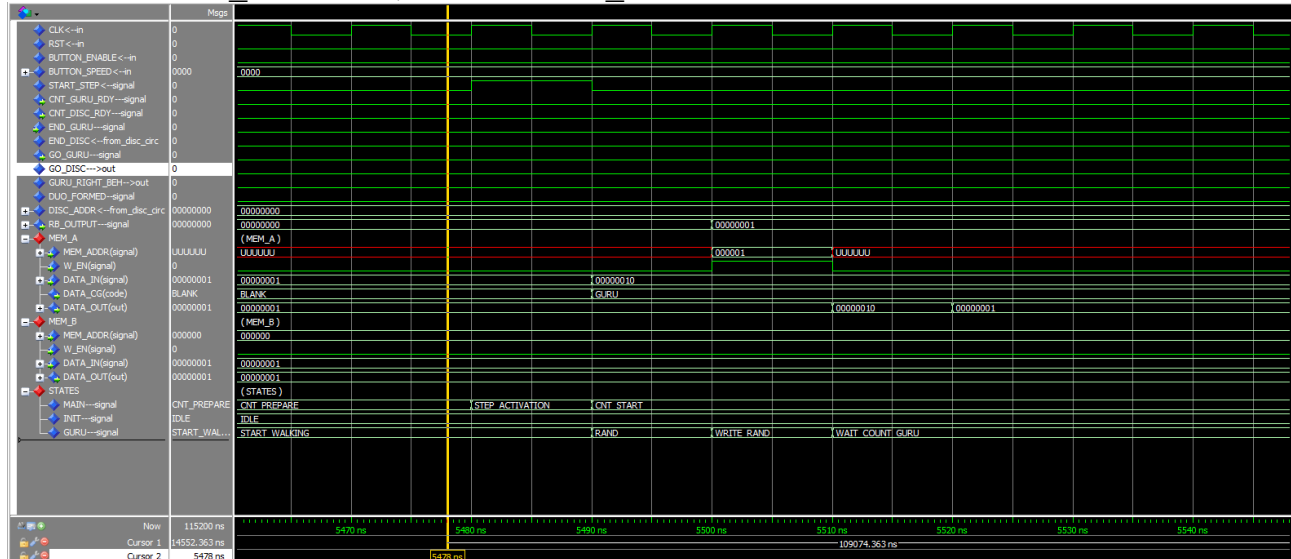


# Relatório Aula 11 – Gabriel Lujan Bonassi – 11256816

1c) captura: imagem do Wave onde fiquem evidente os 3 estados de espera da FSM Guru e de suas relações com os sinais respectivos que os liberam

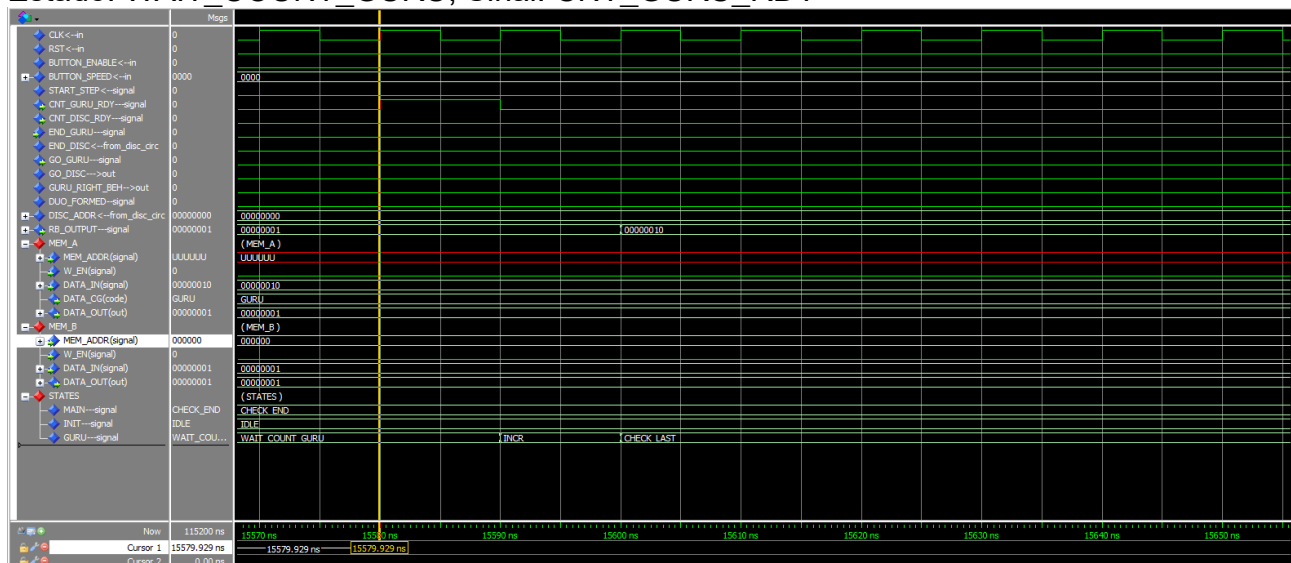
1t) no texto: para cada evento acima, explicar o ocorrido de acordo com a interação do circuito com o ambiente externo

Estado: START\_WALKING; Sinal: START\_STEP



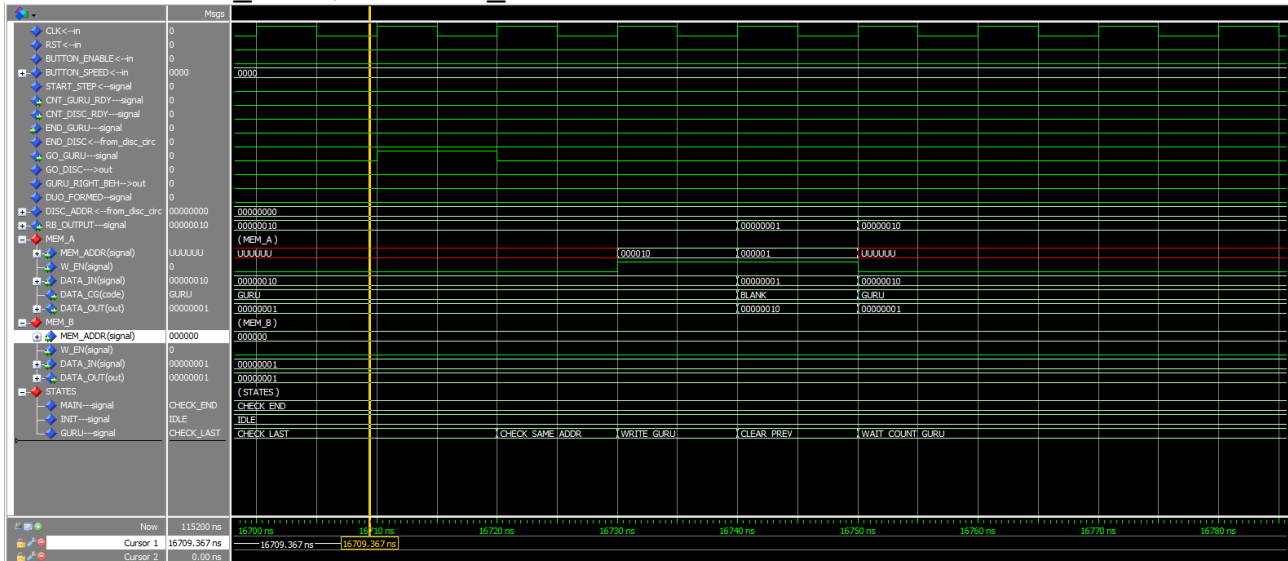
Quando o tabuleiro termina de inicializar, é enviado um sinal START\_STEP, por parte da FSM\_MAIN, que, ao final de um ciclo de clock, joga a FSM\_GURU para o estado RAND, onde ele gera um valor aleatório, e depois para o sinal WRITE\_RAND, onde o sinal W\_EN (enable da memória A). O processo termina com a FSM\_GURU indo para o estado WAIT\_COUNT\_GURU. Do ponto de vista do ambiente externo, o tabuleiro ainda está em branco.

Estado: WAIT\_COUNT\_GURU; Sinal: CNT\_GURU\_RDY



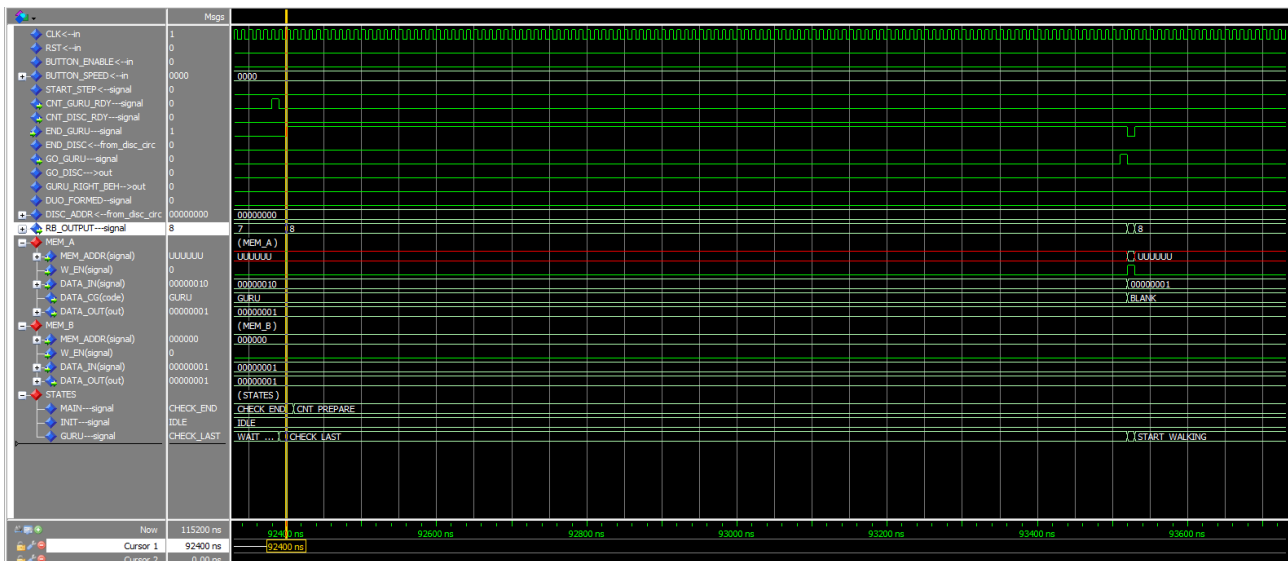
Ao receber o sinal CNT\_GURU\_RDY, a FSM\_GURU muda para o estado INCR, onde ela incrementa o contador e assim muda a posição do guru. Ao fim do processo, a FSM\_GURU vai para o estado CHECK\_LAST. Do ponto de vista do ambiente externo, o guru ainda está em uma posição aleatória.

Estado: CHECK\_LAST; Sinal: GO\_GURU

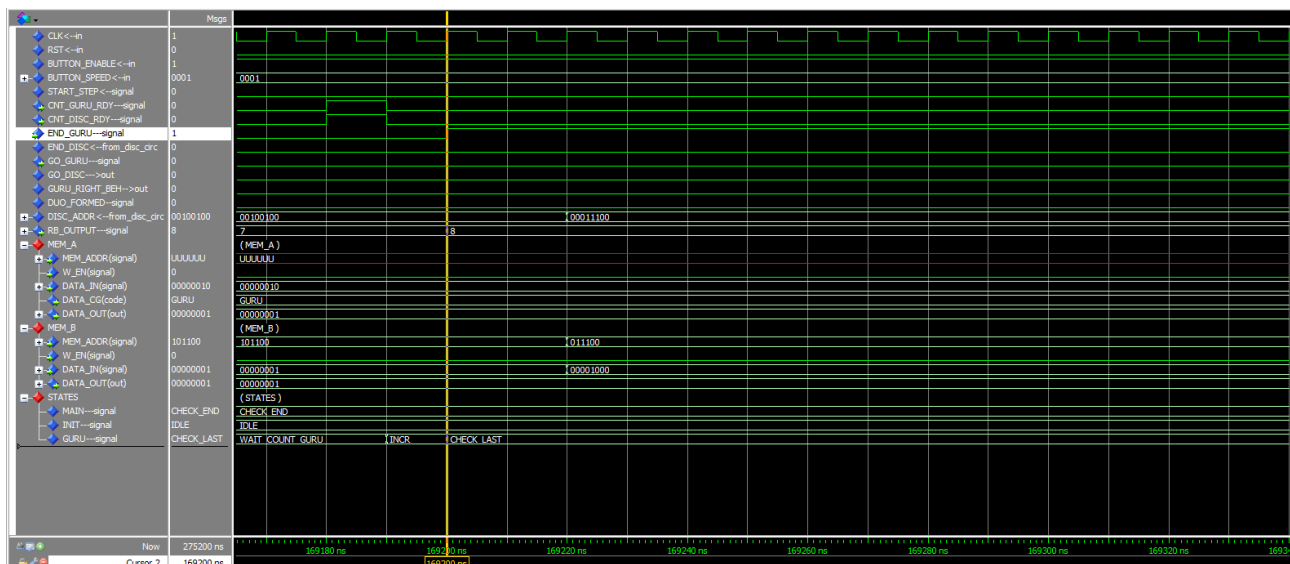


Ao receber o sinal GO\_GURU, a FSM\_GURU decide qual o próximo estado a partir desse sinal e do sinal END\_GURU. As opções são: escrever o guru na posição incrementada, escrever duo caso haja encontro ou, caso o guru tenha ultrapassado a borda do tabuleiro, inicia uma nova rodada. Do ponto de vista do ambiente externo, essa transição trata dos passos do guru, se ele andou uma casa, se encontrou com o discípulo ou se ultrapassou a borda do tabuleiro.

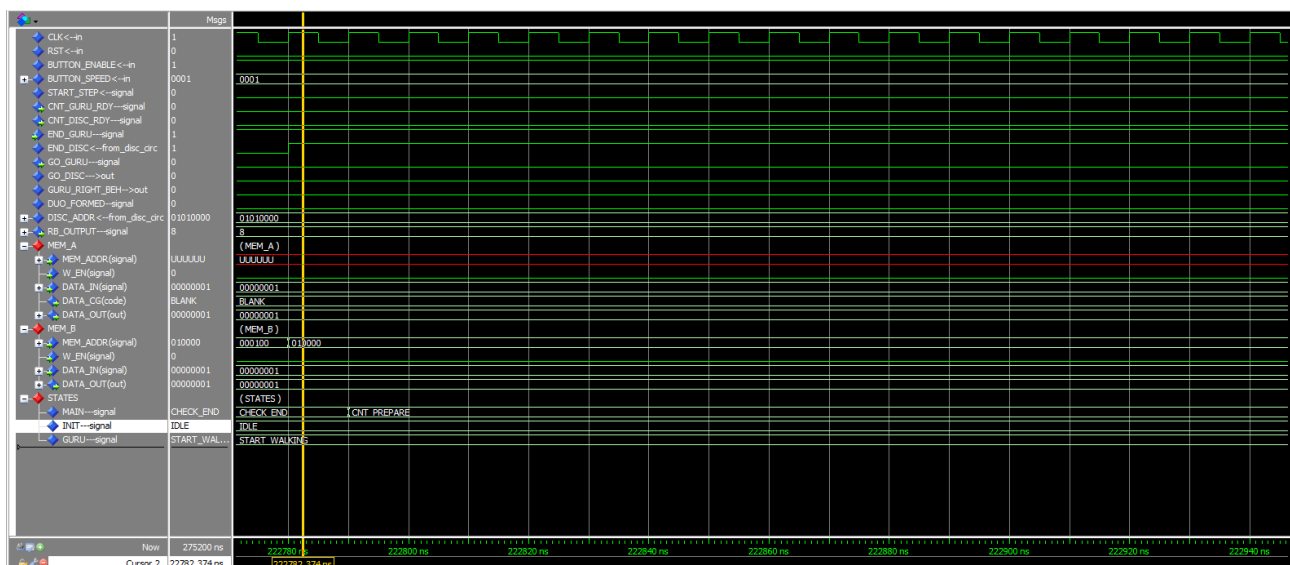
2c) captura: imagem dos eventos em SIM-2 e SIM-3 com os estados de finalização da iteração e o sinais que os determinam.



Momento em que o contador do GURU chega em 8 e seta o sinal END\_GURU = 1, fazendo com que a FSM\_GURU passe ao estado LAST, indicando que o guru passou pela borda do tabuleiro sem que o discípulo tenha sido ativado. Uma nova rodada deve iniciar.



Momento em que o sinal `END_GURU = 1`, e esperamos a `FSM_DISC` chegar ao fim



Momento em que o sinal `END_GURU = 1`, e a `FSM_DISC` chegou ao fim da rodada após passar pela casa 4. Nesta rodada o guru terminou antes do discípulo, então a `FSM_GURU` deve aguardar o discípulo terminar para iniciar uma nova rodada.

2t) no texto: Inclua o trecho do código VHDL da FSM Main onde estes eventos são determinados e explicar o ocorrido

```
when CHECK_END    => if (end_of_guru = '1') AND (end_of_disc = '1') AND (duo_formed = '1') AND
(en_disc = '1') then
    NEXT_STATE <= HIT_POINT;
    elsif ((end_of_guru = '1') AND (end_of_disc = '1')
AND (duo_formed = '0') AND (en_disc = '1'))
    OR ((end_of_guru = '1') AND (en_disc = '0') AND
(duo_formed = '0')) then
    NEXT_STATE <= CNT_PREPARE;
    else
    NEXT_STATE <= CHECK_END;
```

```
end if;
```

Em SIM-2, o guru termina e teremos `end_of_guru='1'` AND `en_disc='0'` AND `duo_formed='0'`. Assim, para esse caso, o próximo estado é CNT\_PREPARE para uma nova rodada.

Em SIM-3, o guru termina indicando `END_GURU = 1` e a FSM\_GURU aguarda enquanto o discípulo não termina. Quando o discípulo envia o sinal de término `END_DISC = 1` e `DUO_FORMED = 0` teremos `end_of_guru='1'` AND `end_of_disc='1'` AND `duo_formed='0'` AND `en_disc='1'`. Assim, a FSM\_MAIN vai do estado CHECK\_END para o estado CNT\_PREPARE, para uma nova rodada.