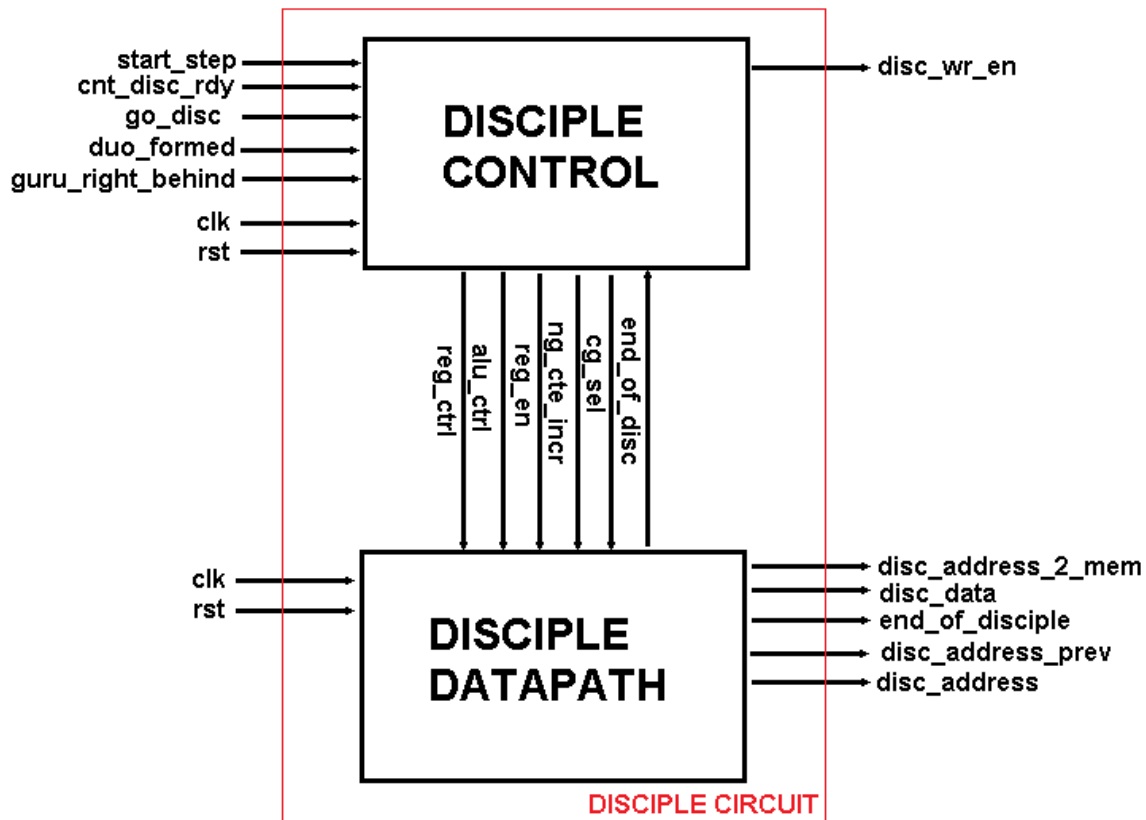


ESQUEMA DISCIPLE CIRCUIT



end_of_disc = Sinal dado pelo datapath indicando que o discípulo passou da borda do tabuleiro.

cg_sel = Sinal vindo do disciple control responsável por indicar o valor a ser escrito na memória (DISCIPLE, BLANCK ou DUO).

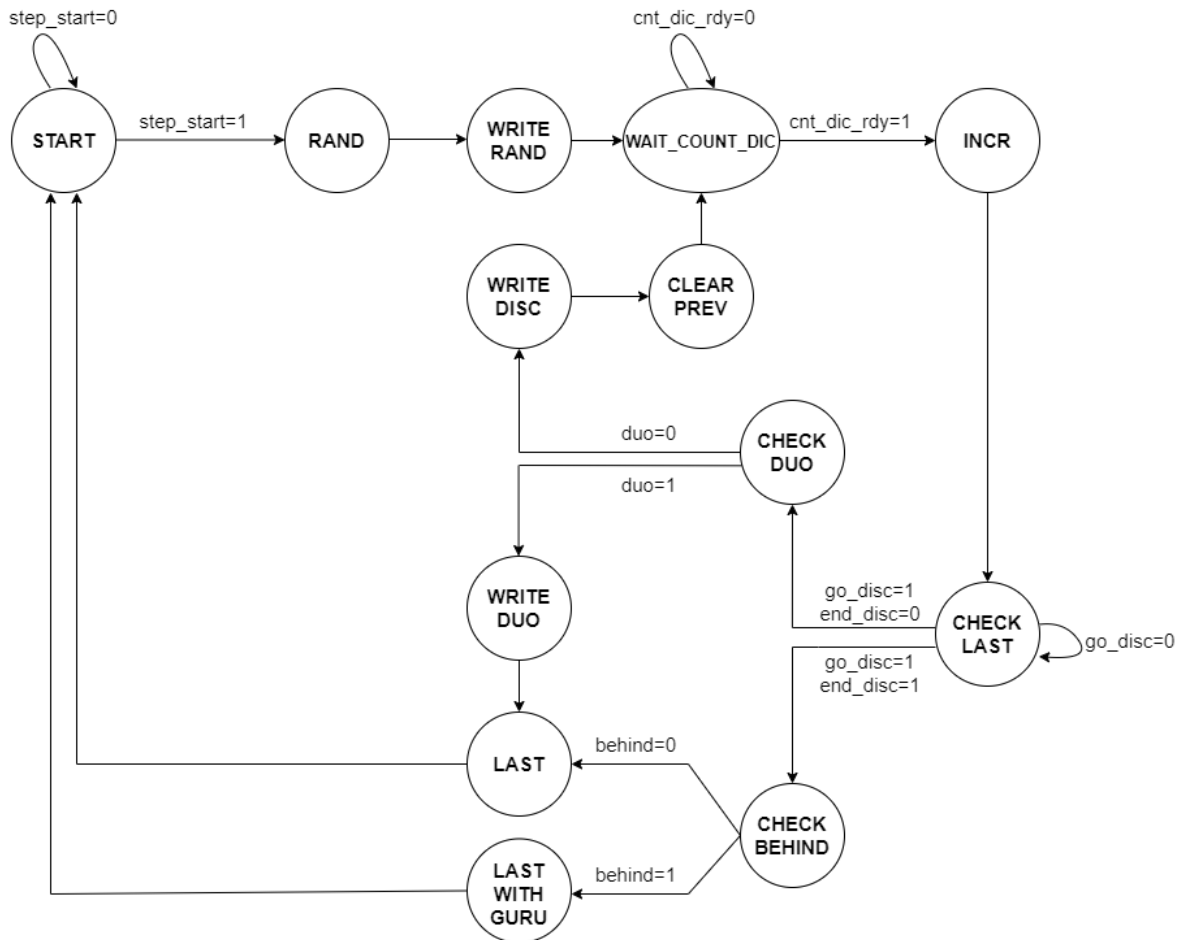
ng_cte_incr = Sinal para controlar o mux de NUM_GEN_DISC a fim de decidir a passagem do sinal aleatório ou da ULA.

alu_ctrl = Sinal utilizado para gerenciar o controle da ULA.

reg_en = Sinal para dar enable aos registradores de REG_BANK_DISC.

reg_ctrl = Sinal para controlar o mux que decide pela passagem do sinal dos registradores de REG_BANK_DISC.

MÁQUINA DE ESTADOS DISCIPLE CONTROL



START= Estado em que a máquina espera pelo sinal `start_step = 1` para poder iniciar o discípulo.

RAND = Estado em que a máquina solicita um numero aleatório ao `rand_num`, fazendo `neg_cte_incr = 0` e `reg_en = 1` para passar o numero aleatório para o `reg_disc`.

WRITE RAND = Estado em que a máquina escreve na memória de acordo com a posição aleatória gerada em `RAND` para o discípulo.

WAIT_COUNT_DISC= Estado em que a máquina espera pelo sinal de passo do discípulo `cnt_disc_rdy = 1`, para poder começar o processo de passo do discípulo.

INCR = Estado em que a máquina faz o incremento de posição do discípulo, fazendo `alu_ctrl = 1`, `ng_cte_incr = 1` e `reg_en = 1`.

CHECK_LAST= Estado em que a máquina averigua se chegou ao final e pode prosseguir ou se deve escrever na posição, faz isso esperando pelo sinal `go_disc=1` e averigua o sinal `end_of_disc` para selecionar o próximo estado.

CHECK_DUO = Estado em que a máquina averigua se o discípulo encontrou o guru, através do sinal `duo_formed`.

WRITE_DISC= Estado em que a máquina escreve `DISCIPLE` na memória no endereço indicado pelo valor de `reg_disc`, isso é feito com `disc_wr_en = 1`, `disc_address = reg_disc`, `reg_ctrl = 0` e `cg_sel = DISCIPLE` (`disc_data = DISCIPLE`).

CLEAR_PREV= Estado em que a máquina escreve `BLANCK` na posição indicada em `reg_disc_previous`, faz isso `reg_ctrl=1`, `disc_wr_en=1`, `disc_address = reg_disc_previous` e `cg_sel= BLANCK`(`disc_data = BLANCK`).

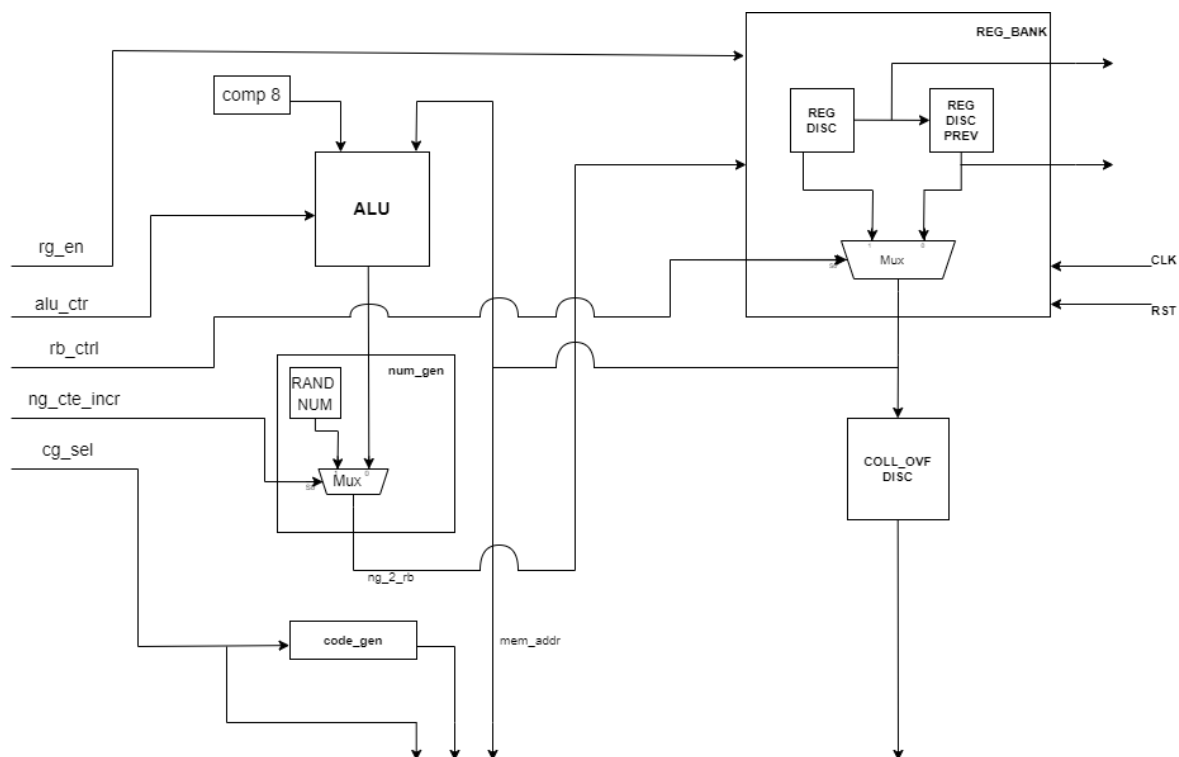
WRITE_DUO= Estado em que máquina escreve DUO na memória no endereço indicado pelo valor de reg_disc, isso é feito com disc_wr_en = 1, disc_address = reg_disc, reg_ctrl = 0 e cg_sel = DUO (disc_data = DUO).

CHECK_BEHIND= Estado em que a máquina avalia se o guru está logo atrás do discípulo através do sinal guru_right_behind, e decide pelo estado seguinte.

LAST_WITH_GURU= Estado em que a máquina dá fim a jogada e não faz nada pois o guru está logo atrás.

LAST= Estado em que a máquina dá fim a jogada e apaga o discípulo da posição anterior, isso é feito escrevendo BLANCK na posição indicada em reg_disc_previous, para isso faz-se reg_ctrl=1, disc_wr_en=1, disc_address = reg_disc_previous e cg_sel= BLANCK(disc_data = BLANCK).

DISCIPLE DATAPATH



ALU= Unidade lógica e aritmética, usada para calcular o passo do guru de acordo com `alu_ctrl`.

NUM_GEN_DISC= módulo que decide pela passagem do número aleatório ou pela passagem do número incrementado pela ALU, para os registradores do discípulo.

CODE_GEN= Módulo que traduz o código BLANCK, DISCIPLE ou DUO para binário.

REG_BANK= módulo de registradores, responsável por guardar o endereço do disciple e o endereço anterior para coordenar os passos do mesmo.

COL_OVF_DISC= Módulo responsável por indicar transbordo para o disciple, isto é, passou da borda superior.