

EXECUÇÃO DA PRIMEIRA INSTRUÇÃO SW:

Primeiro, analisamos a execução da primeira instrução `sw` que escreve no Registro de habilitação de GPIO o valor `0x0000FFFF`.

- Instrução Assembly: `sw t3,0(t4)`
- Instrução em código de máquina: `0x01cea023`

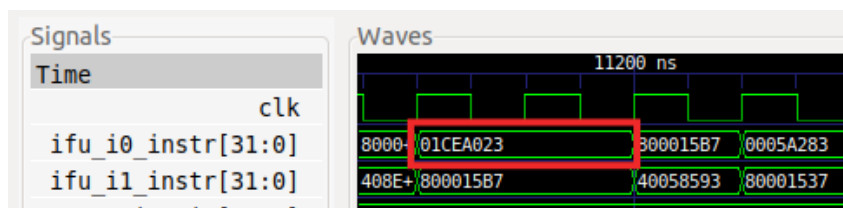
Defina o intervalo de tempo como 11100ns-11800ns



A instrução `store` (`0x01cea023`) é carregada por volta de 11200ns, conforme mostrado no sinal `ifu_i0_instr` (consulte a Figura 1). O prefixo do sinal indica que ele faz parte da Instruction Fetch Unit (`ifu`). Ela está na Via 0 do processador superescalar de 2 vias (`_i0`), e o sinal é a instrução que está sendo carregada (`_instr`).

Após vários ciclos (durante os quais a instrução é decodificada, executada... no CPU), a solicitação de escrita é enviada ao sistema de E/S, conforme mostrado na Figura 1. Especificamente, por volta de 11500ns:

- O CPU envia o endereço para escrita (`wb_m2s_io_adr=0x00001408`) por meio do barramento Wishbone. O endereço é fornecido ao multiplexer usando o sinal `wb_io_adr_i=0x00001408`.
- Com base no endereço `0x00001408`, o multiplexer seleciona o GPIO Slave (`match = 0000010` e `wb_gpio_cyc_0=1`), conectando todos os seus sinais ao barramento Wishbone que se conecta à CPU. Especificamente:
 - `wb_gpio_dat_o = wb_io_dat_i = 0x0000FFFF` (valor fornecido ao GPIO pela instrução de escrita)
 - `wb_gpio_adr_o = wb_io_adr_i = 0x00001408` (Endereço fornecido ao GPIO, que corresponde ao registro de habilitação).
- Finalmente, um ciclo após o multiplexer fazer sua seleção, o registro de habilitação (`ext_padoe_o`) é atualizado com o valor fornecido pela escrita: `ext_padoe_o=0x0000FFFF`.



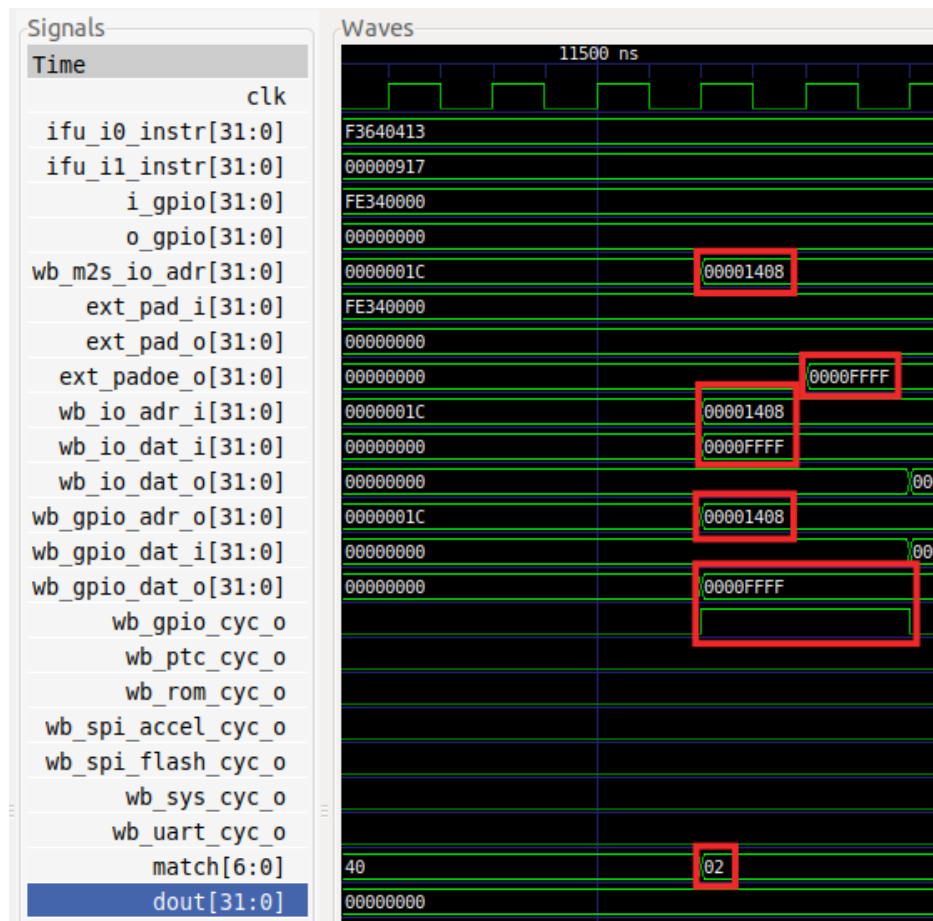


Figura 1 . Simulação da escrita do Enable Register

EXECUÇÃO DA INSTRUÇÃO LW:

Agora analisamos a execução da instrução `lw` que lê o valor dos Switches.

- Instrução Assembly: `lw t0,0(a1)`
- Instrução em código de máquina: `0x0005a283`

Defina o mesmo intervalo de tempo de 11100ns-12000ns. A instrução de leitura (`0x0005a283`) é carregada em torno de 11200ns em `ifu_i0_instr` (consulte Figura 2).

Após vários ciclos (durante os quais a instrução é decodificada e executada no CPU), o pedido de leitura é enviado ao sistema de E/S. Especificamente,

- O valor dos interruptores é fornecido ao módulo GPIO por meio dos sinais `i_gpio` e `ext_pad_i` (pode rever o código da Figura 2). pode verificar na figura que o valor simulado para os interruptores no testbench é 0xFE34, pois esse é o valor contido nos sinais `i_gpio[31:16]` e `ext_pad_i[31:16]`.
- O CPU envia o endereço para escrita (`wb_m2s_io_adr=0x80001400`) por meio do barramento Wishbone. O endereço é fornecido ao multiplexador usando o sinal `wb_io_adr_i=0x80001400`.

- Com base no endereço 0x00001400, o multiplexer seleciona o escravo GPIO (*match* = 0000010 e *wb_gpio_cyc_0*=1), conectando todos os seus sinais ao barramento Wishbone que se conecta ao CPU. Em particular, na simulação, pode ver que:
 - ***wb_io_dat_o* = *wb_gpio_dat_i* = 0xFE340000** (valor fornecido pelo GPIO devido à instrução de leitura).
 - ***wb_gpio_adr_o* = *wb_io_adr_i* = 0x00001400** (Endereço fornecido ao GPIO, que corresponde ao registo de leitura).
- Por fim, observe que, vários ciclos depois, o registo t0 (sinal *dout* na simulação) é atualizado com o valor lido dos interruptores: ***dout*[31:16]=0xFE34**.

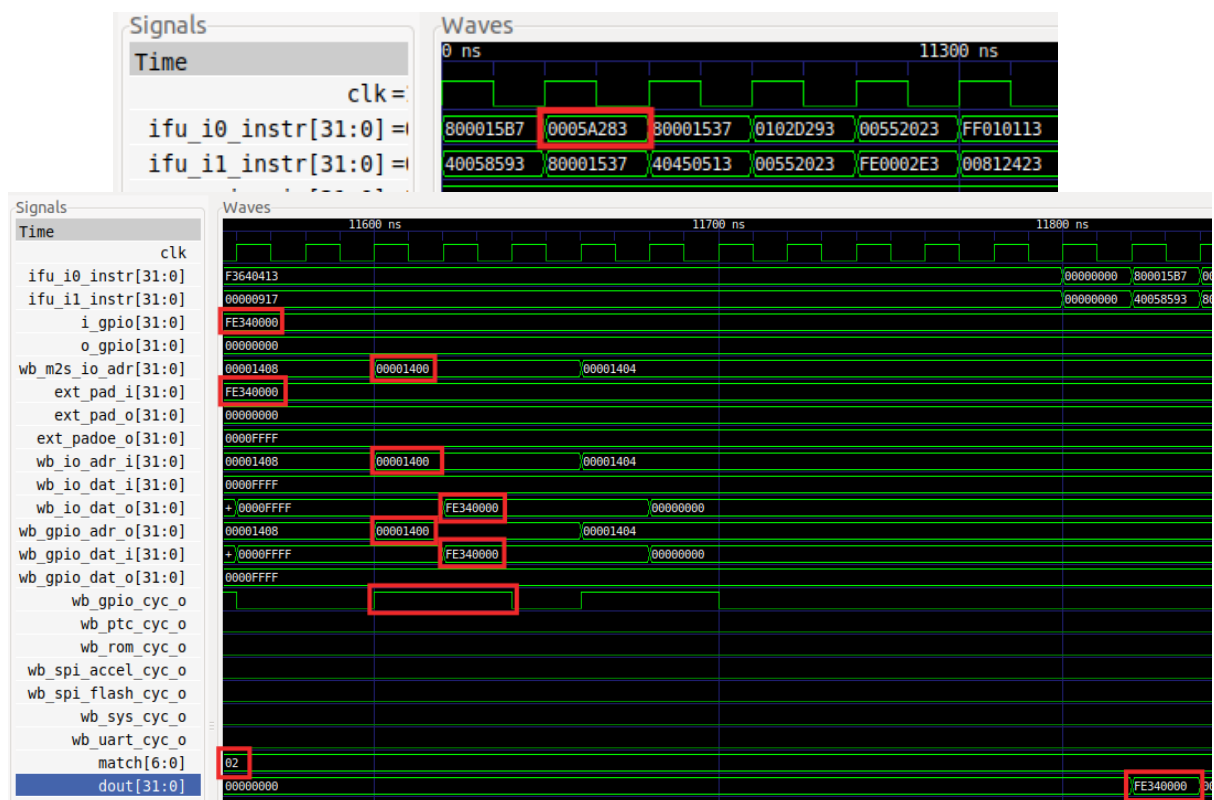


Figura 2 . Simulação da leitura dos interruptores (Switches)

EXECUÇÃO DA SEGUNDA INSTRUÇÃO SW:

Por fim, analisamos a execução da segunda instrução *sw* que define os valores do LED.

- Instrução Assembly: ***sw t0,0(a0)***
- Instrução em código de máquina: **0x00552023**

Defina o intervalo de tempo como 11200ns-12300ns. A instrução store (0x00552023) é carregada em *ifu_i1_instr* por volta de 11300ns (ver Figura 3).

Após vários ciclos (durante os quais a instrução é decodificada e executada no CPU), o

pedido de escrita é enviado ao sistema de E/S. Analise a Figura 3 guiado pelas seguintes etapas:

- O CPU envia o endereço para escrita ($wb_m2s_io_adr=0x80001404$) por meio do barramento Wishbone. O endereço é fornecido ao multiplexer usando o sinal $wb_io_adr_i=0x80001404$.
- Com base no endereço $0x00001404$, o multiplexer seleciona o GPIO Slave ($match = 0000010$ e $wb_gpio_cyc_0=1$), conectando todos os seus sinais ao barramento Wishbone que se conecta ao CPU. Especificamente:
 - **$wb_gpio_dat_o = wb_io_dat_i = 0x0000FE34$** (Valor fornecido ao GPIO pela instrução de escrita)
 - **$wb_gpio_adr_o = wb_io_adr_i = 0x00001404$** (Endereço fornecido ao GPIO, que corresponde ao registro de habilitação).
- Finalmente, um ciclo após o multiplexer fazer sua seleção, ext_pad_o é atualizado com o valor fornecido pela escrita: **$ext_padoe_o=0x0000FE34$** . Esse valor é fornecido aos LEDs por meio do sinal **$o_gpio=0x0000FE34$** .

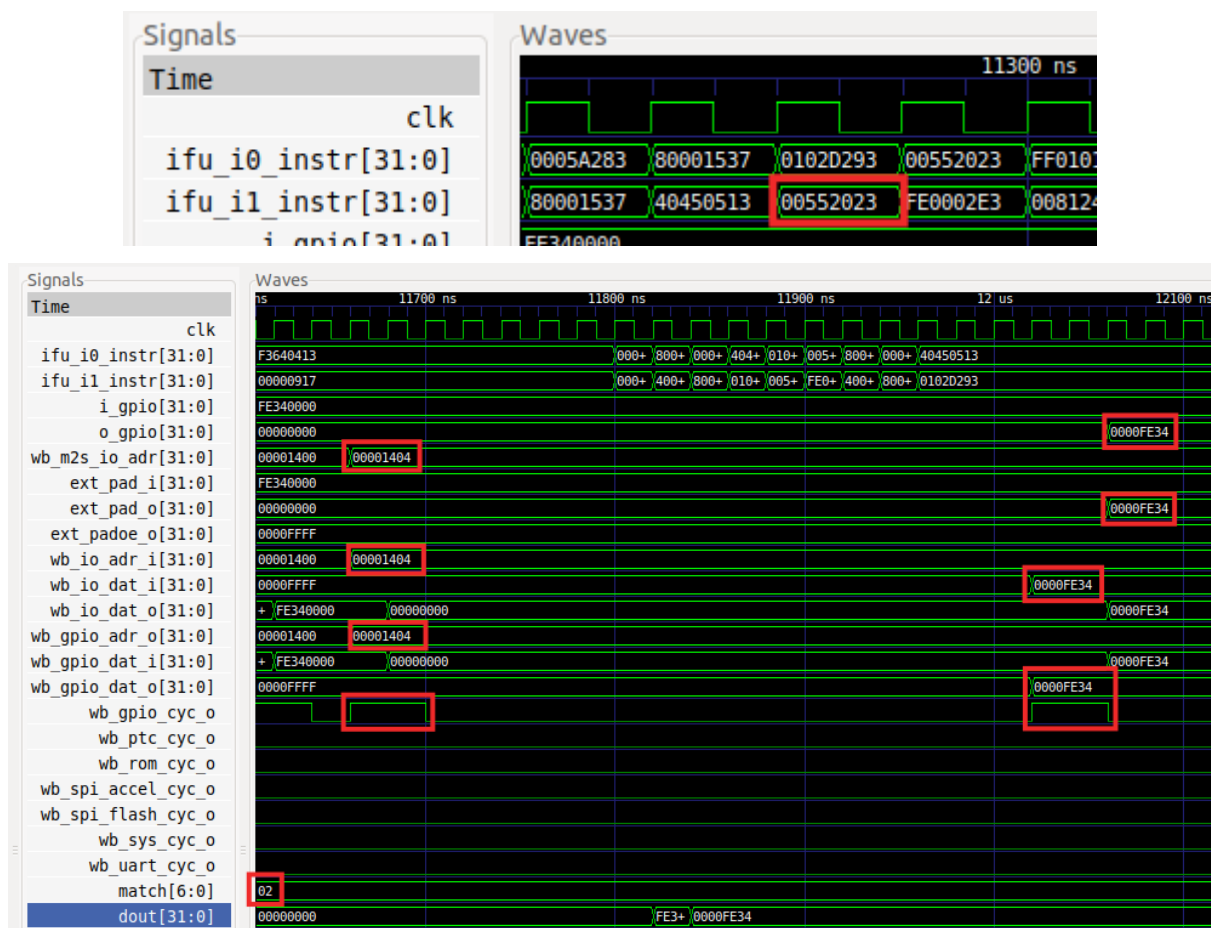


Figura 3 . Simulação da escrita nos LEDs