Edit code - EDA Playground | EPWave Waveform Viewer

edaplayground.com

Gmail YouTube Maps

EDA playground | New | Run | Save* | Resources | Community | Help | Playgrounds | Profile

Brought to you by DOULOS

Doulos does not endorse training material from other suppliers on EDA Playground.

Languages & Libraries

Testbench + Design

SystemVerilog/Verilog

UVM / OVM

None

Other Libraries

None
OVL
SVUnit

- [ ] Enable TL-Verilog
- [ ] Enable Easier UVM
- [ ] Enable VUnit

Tools & Simulators

Icarus Verilog 12.0

Compile Options

-Wall -g2012

Run Options

Run Options

- [ ] Use **run.bash** shell script
- [x] Open **EPWave** after run
- [ ] Show output file after run
- [ ] Download files after run

Examples

testbench.sv

```
//Program 1 to be runned on the designed Processor, (testbench)
//Bipriti Deb Sarma
module test_mips32;
    reg clk1, clk2;
    integer k;

    // Instantiate the MIPS32 processor module
    pipe_MIPS32 mips (clk1, clk2);

    initial begin
        clk1 = 0;
        clk2 = 0;

        // Generating two-phase clock
        repeat (20) begin
            #5 clk1 = 1;
            #5 clk1 = 0;
            #5 clk2 = 1;
            #5 clk2 = 0;
        end
    end

    initial begin
        // Initialize register values
        for (k = 0; k < 32; k = k + 1)
```

design.sv

```
// RISC PROCESSOR - MIPS32 DESIGN CODE
//Bipriti Deb Sarma

module pipe_MIPS32 (clk1, clk2);        // Two-phase clock
    input clk1, clk2;

    reg [31:0] PC, IF_ID_IR, IF_ID_NPC;
    reg [31:0] ID_EX_IR, ID_EX_NPC, ID_EX_A, ID_EX_B, ID_EX_Imm;
    reg [2:0] ID_EX_type, EX_MEM_type, MEM_WB_type;
    reg [31:0] EX_MEM_IR, EX_MEM_ALUOut, EX_MEM_B;
    reg EX_MEM_cond;
    reg [31:0] MEM_WB_IR, MEM_WB_ALUOut, MEM_WB_LMD;
    reg [31:0] regbank [0:31];
    reg [31:0] mem [0:1023];
    // Instruction opcodes and parameter declarations
    parameter ADD = 6'b000000, SUB = 6'b000001, AND = 6'b000010, OR = 6'b000011,
              SLT = 6'b000100, MUL = 6'b000101, HLT = 6'b111111, LW = 6'b001000,
              SW = 6'b001001, ADDI = 6'b001010, SUBI = 6'b001011, SLTI = 6'b001100,
              BNEQZ = 6'b001101, BEQZ = 6'b001110;

    parameter RR_ALU = 3'b000, RM_ALU = 3'b001, LOAD = 3'b010, STORE = 3'b011,
              BRANCH = 3'b100, HALT = 3'b101;

    reg HALTED; // Set after HLT instruction is completed (in WB stage)
    reg TAKEN_BRANCH; // Required to disable instructions after branch
```

Log | Share

```
R0 =  0
R1 = 10
R2 = 20
R3 = 25
R4 = 30
R5 = 55
Finding VCD file...
./mips.vcd
[2024-09-20 16:52:44 UTC] Opening EPWave...
Done
```

Edit code - EDA Playground | EPWave Waveform Viewer

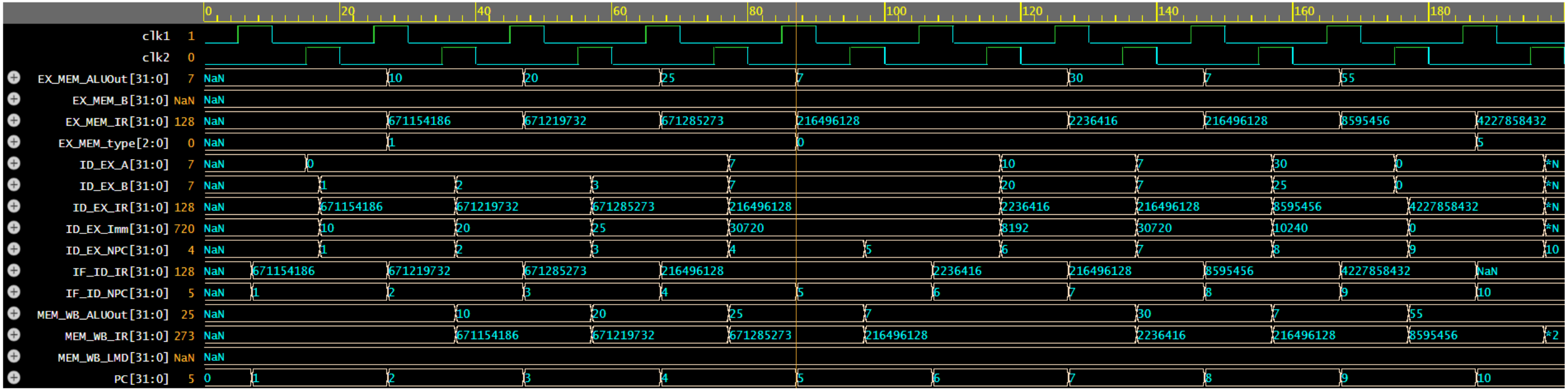
edaplayground.com/launchEpwave

Gmail YouTube Maps

EPWave *beta* | Load | Save | Examples | About | Apps | Rcc1bba34-2697-4b85-be74-d3e9e

EDA Playground | https://www.edaplayground.com/ | From: 0s | To: 400s

Get Signals | Radix | 100% | Δ 499,999,999,913s



Brought to you by DOULOS

Edit code - EDA Playground | EPWave Waveform Viewer

edaplayground.com

Gmail YouTube Maps

EDA playground | New | Run | Save* | Resources | Community | Help | Playgrounds | Profile

Brought to you by DOULOS

Doulos does not endorse training material from other suppliers on EDA Playground.

Languages & Libraries

Testbench + Design: SystemVerilog/Verilog

UVM / OVM: None

Other Libraries: None, OVL, SVUnit

Enable TL-Verilog

Enable Easier UVM

Enable VUnit

Tools & Simulators

Icarus Verilog 12.0

Compile Options: -Wall -g2012

Run Options

Use **run.bash** shell script

Open **EPWave** after run

Show output file after run

Download files after run

Examples

using EDA Playground

VHDL

Verilog/SystemVerilog

UVM

testbench.sv

```
//Program 2 to be runned on the designed Processor, (testbench)
//Bipriti Deb Sarma
module test_mips32;
 reg clk1, clk2;
 integer k;

// Instantiate the MIPS processor module
pipe_MIPS32 mips (clk1, clk2);

// Clock generation block: generate two-phase clock
initial
begin
 clk1 = 0;
 clk2 = 0;
 repeat(50)   // Generating two-phase clock for 50 cycles
 begin
   #5 clk1 = 1;
   #5 clk1 = 0;
   #5 clk2 = 1;
   #5 clk2 = 0;
 end
```

design.sv

```
// RISC PROCESSOR - MIPS32 DESIGN CODE
//Bipriti Deb Sarma

module pipe_MIPS32 (clk1, clk2);        // Two-phase clock
    input clk1, clk2;

    reg [31:0] PC, IF_ID_IR, IF_ID_NPC;
    reg [31:0] ID_EX_IR, ID_EX_NPC, ID_EX_A, ID_EX_B, ID_EX_Imm;
    reg [2:0] ID_EX_type, EX_MEM_type, MEM_WB_type;
    reg [31:0] EX_MEM_IR, EX_MEM_ALUOut, EX_MEM_B;
    reg EX_MEM_cond;
    reg [31:0] MEM_WB_IR, MEM_WB_ALUOut, MEM_WB_LMD;
    reg [31:0] regbank [0:31];
    reg [31:0] mem [0:1023];
    // Instruction opcodes and parameter declarations
    parameter ADD = 6'b000000, SUB = 6'b000001, AND = 6'b000010, OR = 6'b000011,
              SLT = 6'b000100, MUL = 6'b000101, HLT = 6'b111111, LW = 6'b001000,
              SW = 6'b001001, ADDI = 6'b001010, SUBI = 6'b001011, SLTI = 6'b001100,
              BNEQZ = 6'b001101, BEQZ = 6'b001110;

    parameter RR_ALU = 3'b000, RM_ALU = 3'b001, LOAD = 3'b010, STORE = 3'b011,
```

Log | Share

```
[2024-09-20 16:55:59 UTC] iverilog '-Wall' '-g2012' design.sv testbench.sv  && unbuffer vvp a.out
VCD info: dumpfile mips.vcd opened for output.
Time:    0 | R2:    2
Time:   67 | R2:    1
Time:  147 | R2:    7
Time:  247 | R2:   42
Time:  347 | R2:  210
Time:  447 | R2:  840
Time:  547 | R2: 2520
Time:  647 | R2: 5040
Time:  747 | R2: 5040
Mem[200] =  7, Mem[198] =   5040
testbench.sv:68: $finish called at 3000 (1s)
Finding VCD file...
./mips.vcd
[2024-09-20 16:56:00 UTC] Opening EPWave...
```

Edit code - EDA Playground | EPWave Waveform Viewer

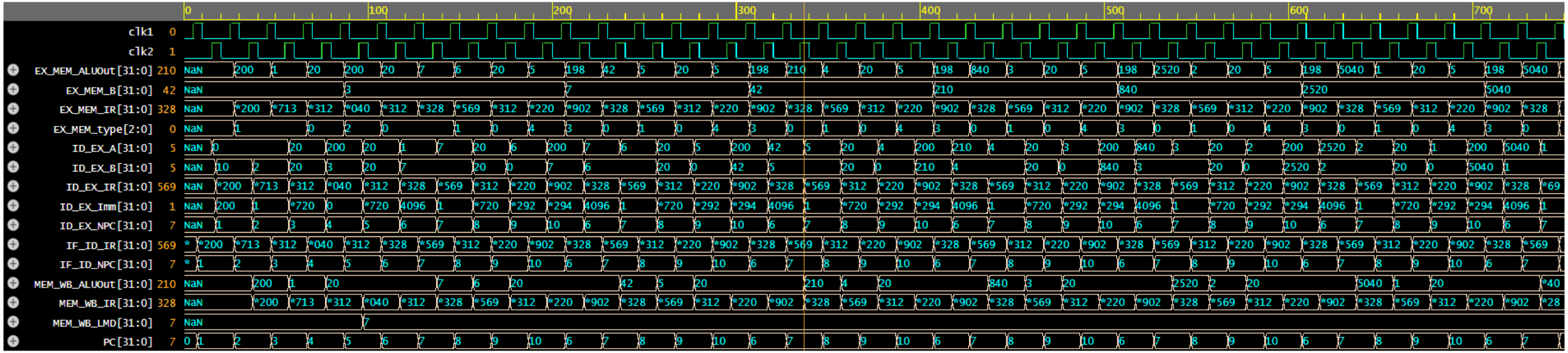
edaplayground.com/launchEpwave

Gmail YouTube Maps

EPWave beta | Load | Save | Examples | About | Apps | Rcc1bba34-2697-4b85-be74-d3e9e

EDA Playground | https://www.edaplayground.com/ | From: 0s | To: 3,000s

Get Signals | Radix | 100% | Δ 499,999,999,663s



Brought to you by DOULOS