EE446 PROJECT REPORT

Arda Ünver - 2444081

Deniz Karakay - 2443307

SUBMISSION DATE:

14.06.2023

Introduction

The effective execution of instructions is essential for improving the performance of contemporary processors in the constantly developing field of computer architecture. To increase instruction throughput, pipelining, a method that facilitates the simultaneous execution of several instructions, has gained popularity. The smooth flow of instructions across the pipeline, however, may be hampered by risks, which are a new problem brought on by pipelining.

By adding a **hazard unit** and a **branch predictor**, this experiment seeks to improve the functionality of a 32-bit pipelined processor created in a prior Laboratory Work. Two categories of hazards will be detected and handled by the hazard unit, ensuring that pipeline instructions are carried out correctly. On the other side, the branch predictor will boost processor speed by correctly anticipating the results of branch instructions.

The hazard unit and branch predictor must be implemented on a working pipelined processor. This project makes use of the earlier work that was completed in the fourth lab to increase the processor's capabilities. By including the hazard unit and branch predictor into the current **Datapath** and **Control Unit**, we hope to create a processor design that is more dependable and efficient.

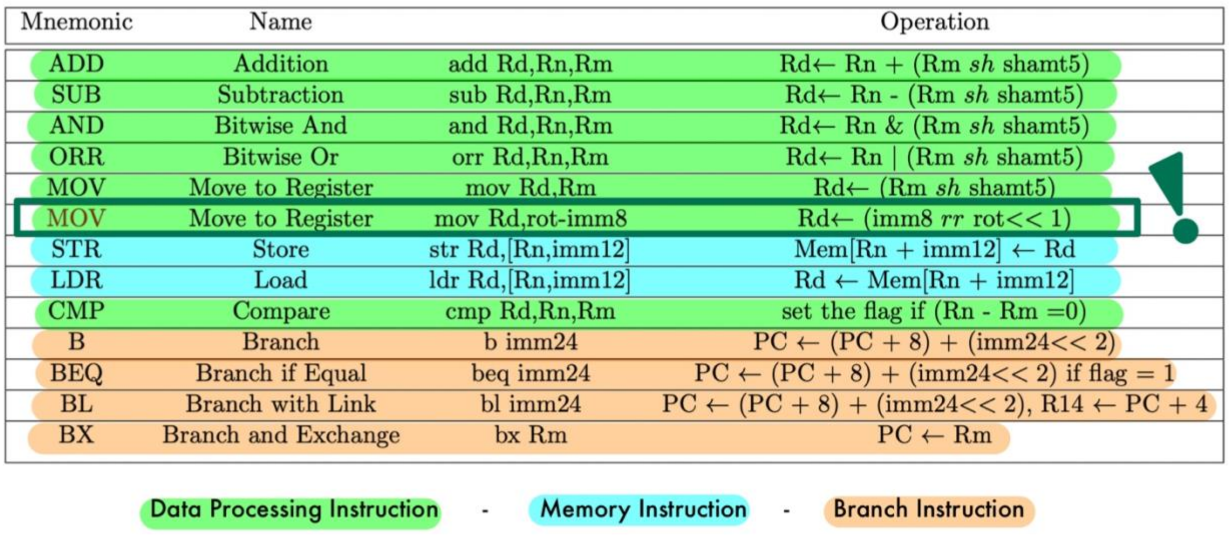
Three mechanisms—**flush**, **stall**, and **forward**—will be the main ones used by the hazard unit. A hazard is recognized, and the flush mechanism clears the stage register linked to it, deleting its output. The pipeline is filled with a bubble to eliminate the risk, and the stall mechanism holds the value of a stage register. Finally, the forward mechanism enables the transmission of calculated values to earlier stages, facilitating the elimination of hazards and uninterrupted instruction execution.

The branch target buffer (BTB), global branch history register (GHR), and pattern table make up the branch predictor, an essential aspect of contemporary CPUs. To predict the target of upcoming branches, the branch target buffer holds the program counters (PCs) and associated branch target addresses of recent branch instructions. The global branch history record keeps track of previous branch outcomes, which helps with forecast accuracy. Based on past trends, the pattern table serves as a lookup table to help determine the most likely branch outcome.

By bringing these improvements into practice, we want to increase the pipelined processor's effectiveness at handling risks and forecasting branch outcomes. This project provides an excellent chance to delve further into the intricate details of computer architecture while investigating workable solutions to problems posed by current processor architectures.

ISA to be Implemented

Figure 1 illustrates how the processor design for this project will be concentrated on a limited set of ARM instructions. Not all ARM instructions will be supported by this processor. However, it will provide support for conditional logic, specifically the EQ (equal), NE (not equal), and AL (always) conditions, with the Z flag being the only significant condition. If the condition is EQ, the processor will assess the zero flag to decide whether to carry out the instruction or not.



Mnemonic	Name		Operation
ADD	Addition	add Rd,Rn,Rm	Rd← Rn + (Rm *sh* shamt5)
SUB	Subtraction	sub Rd,Rn,Rm	Rd← Rn - (Rm *sh* shamt5)
AND	Bitwise And	and Rd,Rn,Rm	Rd← Rn & (Rm *sh* shamt5)
ORR	Bitwise Or	orr Rd,Rn,Rm	Rd← Rn | (Rm *sh* shamt5)
MOV	Move to Register	mov Rd,Rm	Rd← (Rm *sh* shamt5)
MOV	Move to Register	mov Rd,rot-imm8	Rd← (imm8 *rr* rot<< 1)
STR	Store	str Rd,[Rn,imm12]	Mem[Rn + imm12] ← Rd
LDR	Load	ldr Rd,[Rn,imm12]	Rd ← Mem[Rn + imm12]
CMP	Compare	cmp Rd,Rn,Rm	set the flag if (Rn - Rm =0)
B	Branch	b imm24	PC ← (PC + 8) + (imm24<< 2)
BEQ	Branch if Equal	beq imm24	PC ← (PC + 8) + (imm24<< 2) if flag = 1
BL	Branch with Link	bl imm24	PC ← (PC + 8) + (imm24<< 2), R14 ← PC + 4
BX	Branch and Exchange	bx Rm	PC ← Rm

Data Processing Instruction - Memory Instruction - Branch Instruction

Figure 1. ISA to be Implemented.

The new MOV instruction with an immediate operand is a major addition to the instruction set. The rotate field, a 4-bit unsigned integer describing a shift operation on an 8-bit immediate value, is part of the immediate operand. The immediate value is rotated right by twice the value in the rotate field after being zero-extended to 32 bits. This method makes it possible to generate common constants quickly and effectively, increasing the processor's adaptability. It is essential to note that this processor will carry out instructions involving the transfer of another register to the PC, such as BX. These instructions will not be handled by the branch predictor, nevertheless.

Datapath Changes

As it can be seen from *Figure 2*, additional MUXs (4x1 MUX) are added to utilize the forwarding signals generated at the Hazard Unit. However, the Datapath design in *Figure 2* **does not** contain all the changes, for the overall project. It is only added to visualize the 5-stage Pipelined Computer Architecture with the Hazard Unit.

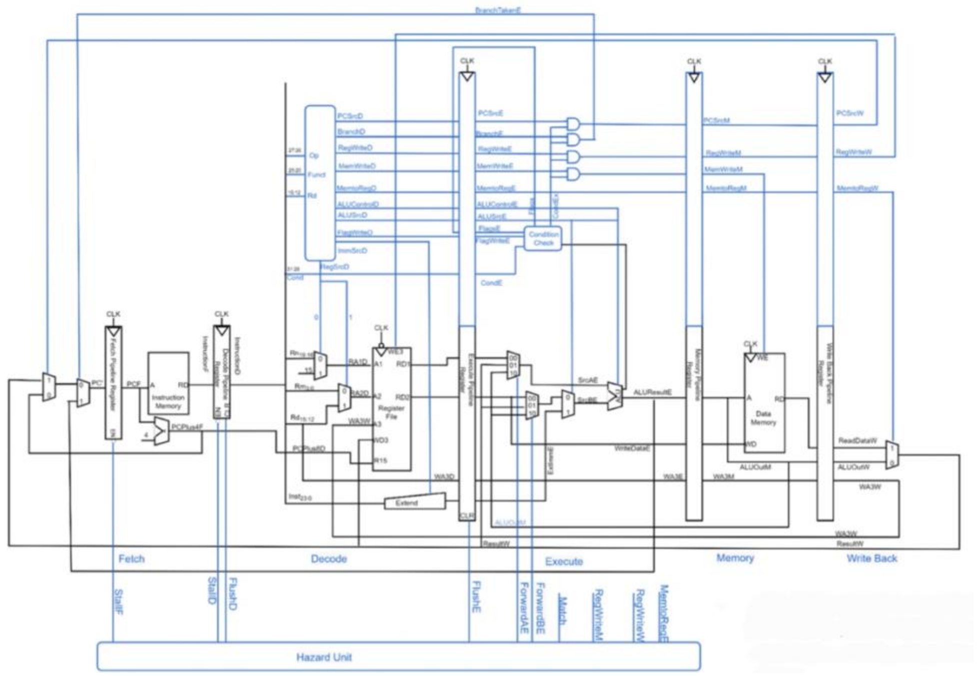


Figure 2. Datapath of the 5-stage Pipelined Computer Architecture with the Hazard Unit added.

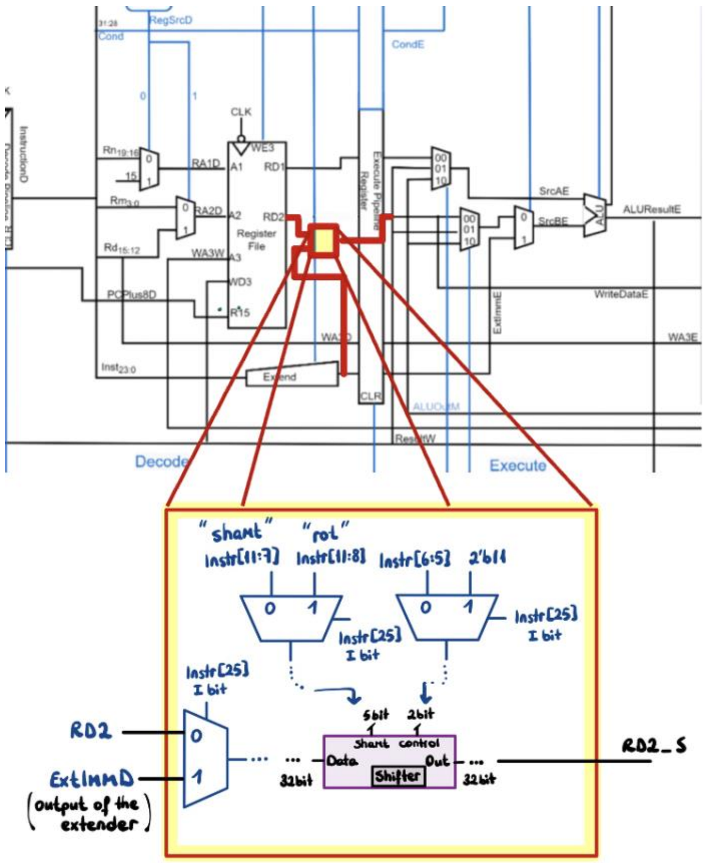


Figure 3. Datapath change regarding the altered MOV operation.

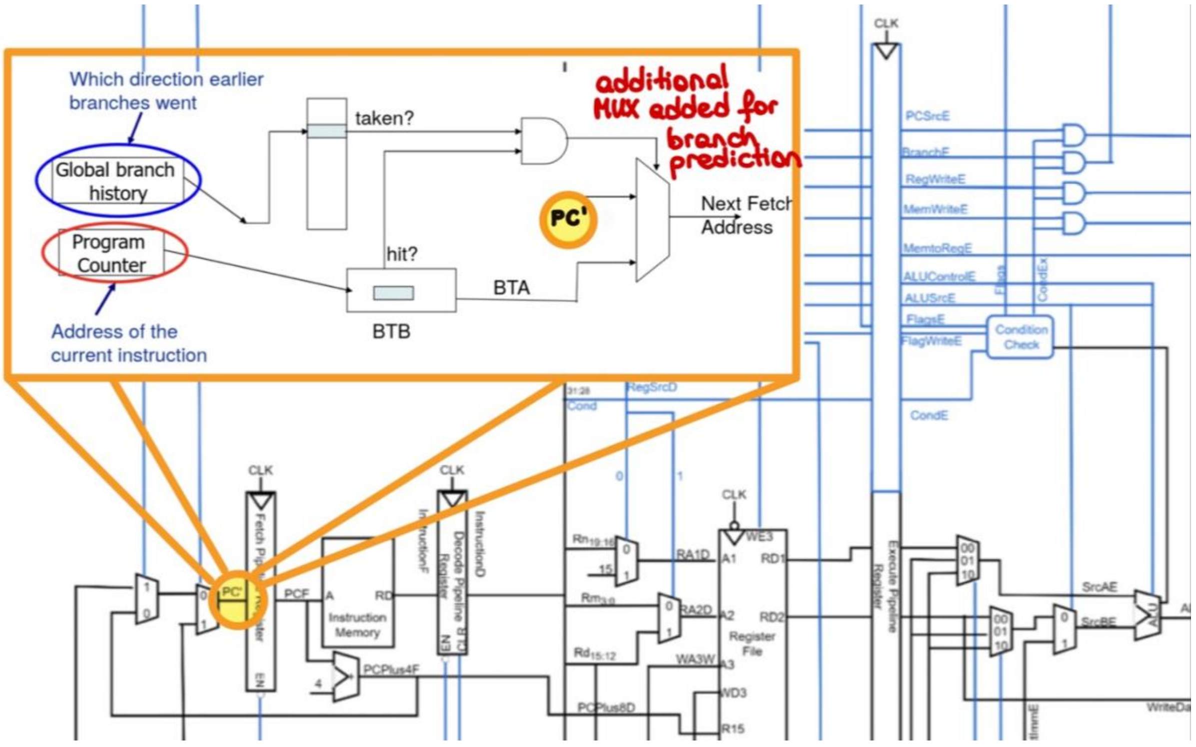
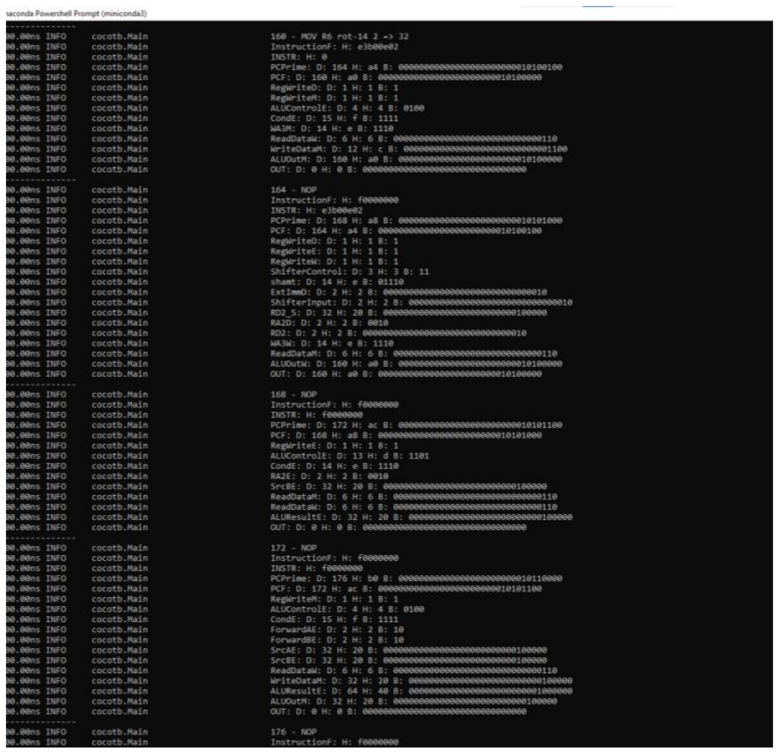
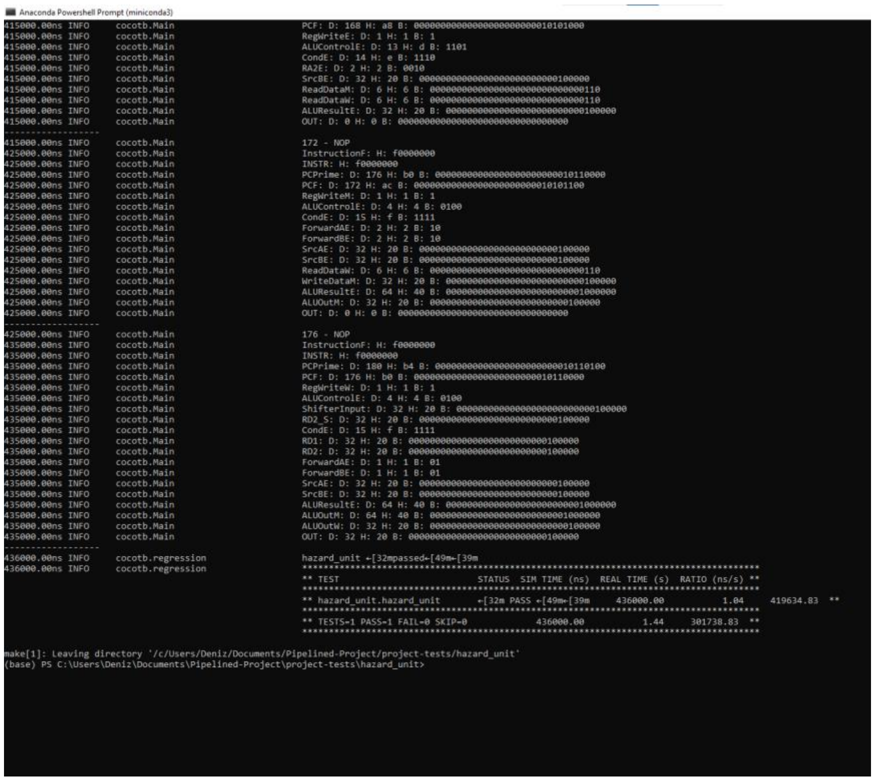


Figure 4. Datapath changes regarding branch prediction involving Global Branch History and Branch Target Buffer.


naconda Powershell Prompt (miniconda3)

```
--------------
00.00ns INFO     cocotb.Main                    160 - MOV R6 rot-14 2 -> 32
00.00ns INFO     cocotb.Main                    InstructionF: H: e3b00e02
00.00ns INFO     cocotb.Main                    INSTR: H: 0
00.00ns INFO     cocotb.Main                    PCPrime: D: 164 H: a4 B: 00000000000000000000000010100100
00.00ns INFO     cocotb.Main                    PCF: D: 160 H: a0 B: 00000000000000000000000010100000
00.00ns INFO     cocotb.Main                    RegWriteD: D: 1 H: 1 B: 1
00.00ns INFO     cocotb.Main                    RegWriteM: D: 1 H: 1 B: 1
00.00ns INFO     cocotb.Main                    ALUControlE: D: 4 H: 4 B: 0100
00.00ns INFO     cocotb.Main                    CondE: D: 15 H: f B: 1111
00.00ns INFO     cocotb.Main                    WA3M: D: 14 H: e B: 1110
00.00ns INFO     cocotb.Main                    ReadDataW: D: 6 H: 6 B: 00000000000000000000000000000110
00.00ns INFO     cocotb.Main                    WriteDataM: D: 12 H: c B: 00000000000000000000000000001100
00.00ns INFO     cocotb.Main                    ALUOutM: D: 160 H: a0 B: 00000000000000000000000010100000
00.00ns INFO     cocotb.Main                    OUT: D: 0 H: 0 B: 00000000000000000000000000000000
--------------
00.00ns INFO     cocotb.Main                    164 - NOP
00.00ns INFO     cocotb.Main                    InstructionF: H: f0000000
00.00ns INFO     cocotb.Main                    INSTR: H: e3b00e02
00.00ns INFO     cocotb.Main                    PCPrime: D: 168 H: a8 B: 00000000000000000000000010101000
00.00ns INFO     cocotb.Main                    PCF: D: 164 H: a4 B: 00000000000000000000000010100100
00.00ns INFO     cocotb.Main                    RegWriteD: D: 1 H: 1 B: 1
00.00ns INFO     cocotb.Main                    RegWriteE: D: 1 H: 1 B: 1
00.00ns INFO     cocotb.Main                    RegWriteW: D: 1 H: 1 B: 1
00.00ns INFO     cocotb.Main                    ShifterControl: D: 3 H: 3 B: 11
00.00ns INFO     cocotb.Main                    shamt: D: 14 H: e B: 01110
00.00ns INFO     cocotb.Main                    ExtImmD: D: 2 H: 2 B: 00000000000000000000000000000010
00.00ns INFO     cocotb.Main                    ShifterInput: D: 2 H: 2 B: 00000000000000000000000000000010
00.00ns INFO     cocotb.Main                    RD2_S: D: 32 H: 20 B: 00000000000000000000000000100000
00.00ns INFO     cocotb.Main                    RA2D: D: 2 H: 2 B: 0010
00.00ns INFO     cocotb.Main                    RD2: D: 2 H: 2 B: 00000000000000000000000000000010
00.00ns INFO     cocotb.Main                    WA3W: D: 14 H: e B: 1110
00.00ns INFO     cocotb.Main                    ReadDataM: D: 6 H: 6 B: 00000000000000000000000000000110
00.00ns INFO     cocotb.Main                    ALUOutW: D: 160 H: a0 B: 00000000000000000000000010100000
00.00ns INFO     cocotb.Main                    OUT: D: 160 H: a0 B: 00000000000000000000000010100000
--------------
00.00ns INFO     cocotb.Main                    168 - NOP
00.00ns INFO     cocotb.Main                    InstructionF: H: f0000000
00.00ns INFO     cocotb.Main                    INSTR: H: f0000000
00.00ns INFO     cocotb.Main                    PCPrime: D: 172 H: ac B: 00000000000000000000000010101100
00.00ns INFO     cocotb.Main                    PCF: D: 168 H: a8 B: 00000000000000000000000010101000
00.00ns INFO     cocotb.Main                    RegWriteE: D: 1 H: 1 B: 1
00.00ns INFO     cocotb.Main                    ALUControlE: D: 13 H: d B: 1101
00.00ns INFO     cocotb.Main                    CondE: D: 14 H: e B: 1110
00.00ns INFO     cocotb.Main                    RA2E: D: 2 H: 2 B: 0010
00.00ns INFO     cocotb.Main                    SrcBE: D: 32 H: 20 B: 00000000000000000000000000100000
00.00ns INFO     cocotb.Main                    ReadDataM: D: 6 H: 6 B: 00000000000000000000000000000110
00.00ns INFO     cocotb.Main                    ReadDataW: D: 6 H: 6 B: 00000000000000000000000000000110
00.00ns INFO     cocotb.Main                    ALUResultE: D: 32 H: 20 B: 00000000000000000000000000100000
00.00ns INFO     cocotb.Main                    OUT: D: 0 H: 0 B: 00000000000000000000000000000000
--------------
00.00ns INFO     cocotb.Main                    172 - NOP
00.00ns INFO     cocotb.Main                    InstructionF: H: f0000000
00.00ns INFO     cocotb.Main                    INSTR: H: f0000000
00.00ns INFO     cocotb.Main                    PCPrime: D: 176 H: b0 B: 00000000000000000000000010110000
00.00ns INFO     cocotb.Main                    PCF: D: 172 H: ac B: 00000000000000000000000010101100
00.00ns INFO     cocotb.Main                    RegWriteM: D: 1 H: 1 B: 1
00.00ns INFO     cocotb.Main                    ALUControlE: D: 4 H: 4 B: 0100
00.00ns INFO     cocotb.Main                    CondE: D: 15 H: f B: 1111
00.00ns INFO     cocotb.Main                    ForwardAE: D: 2 H: 2 B: 10
00.00ns INFO     cocotb.Main                    ForwardBE: D: 2 H: 2 B: 10
00.00ns INFO     cocotb.Main                    SrcAE: D: 32 H: 20 B: 00000000000000000000000000100000
00.00ns INFO     cocotb.Main                    SrcBE: D: 32 H: 20 B: 00000000000000000000000000100000
00.00ns INFO     cocotb.Main                    ReadDataW: D: 6 H: 6 B: 00000000000000000000000000000110
00.00ns INFO     cocotb.Main                    WriteDataM: D: 32 H: 20 B: 00000000000000000000000000100000
00.00ns INFO     cocotb.Main                    ALUResultE: D: 64 H: 40 B: 00000000000000000000000001000000
00.00ns INFO     cocotb.Main                    ALUOutM: D: 32 H: 20 B: 00000000000000000000000000100000
00.00ns INFO     cocotb.Main                    OUT: D: 0 H: 0 B: 00000000000000000000000000000000
--------------
00.00ns INFO     cocotb.Main                    176 - NOP
00.00ns INFO     cocotb.Main                    InstructionF: H: f0000000
```

Figure 5. COCOTB results from the Fetch, Decode, Execute and Memory stages of the altered MOV instruction.



```
Anaconda Powershell Prompt (miniconda3)
415000.00ns INFO     cocotb.Main                        PCF: D: 168 H: a8 B: 00000000000000000000000010101000
415000.00ns INFO     cocotb.Main                        RegWriteE: D: 1 H: 1 B: 1
415000.00ns INFO     cocotb.Main                        ALUControlE: D: 13 H: d B: 1101
415000.00ns INFO     cocotb.Main                        CondE: D: 14 H: e B: 1110
415000.00ns INFO     cocotb.Main                        RA2E: D: 2 H: 2 B: 0010
415000.00ns INFO     cocotb.Main                        SrcBE: D: 32 H: 20 B: 00000000000000000000000000100000
415000.00ns INFO     cocotb.Main                        ReadDataM: D: 6 H: 6 B: 00000000000000000000000000000110
415000.00ns INFO     cocotb.Main                        ReadDataW: D: 6 H: 6 B: 00000000000000000000000000000110
415000.00ns INFO     cocotb.Main                        ALUResultE: D: 32 H: 20 B: 00000000000000000000000000100000
415000.00ns INFO     cocotb.Main                        OUT: D: 0 H: 0 B: 00000000000000000000000000000000
------------------
415000.00ns INFO     cocotb.Main                        172 - NOP
425000.00ns INFO     cocotb.Main                        InstructionF: H: f0000000
425000.00ns INFO     cocotb.Main                        INSTR: H: f0000000
425000.00ns INFO     cocotb.Main                        PCPrime: D: 176 H: b0 B: 00000000000000000000000010110000
425000.00ns INFO     cocotb.Main                        PCF: D: 172 H: ac B: 00000000000000000000000010101100
425000.00ns INFO     cocotb.Main                        RegWriteM: D: 1 H: 1 B: 1
425000.00ns INFO     cocotb.Main                        ALUControlE: D: 4 H: 4 B: 0100
425000.00ns INFO     cocotb.Main                        CondE: D: 15 H: f B: 1111
425000.00ns INFO     cocotb.Main                        ForwardAE: D: 2 H: 2 B: 10
425000.00ns INFO     cocotb.Main                        ForwardBE: D: 2 H: 2 B: 10
425000.00ns INFO     cocotb.Main                        SrcAE: D: 32 H: 20 B: 00000000000000000000000000100000
425000.00ns INFO     cocotb.Main                        SrcBE: D: 32 H: 20 B: 00000000000000000000000000100000
425000.00ns INFO     cocotb.Main                        ReadDataW: D: 6 H: 6 B: 00000000000000000000000000000110
425000.00ns INFO     cocotb.Main                        WriteDataM: D: 32 H: 20 B: 00000000000000000000000000100000
425000.00ns INFO     cocotb.Main                        ALUResultE: D: 64 H: 40 B: 00000000000000000000000001000000
425000.00ns INFO     cocotb.Main                        ALUOutM: D: 32 H: 20 B: 00000000000000000000000000100000
425000.00ns INFO     cocotb.Main                        OUT: D: 0 H: 0 B: 00000000000000000000000000000000
------------------
425000.00ns INFO     cocotb.Main                        176 - NOP
435000.00ns INFO     cocotb.Main                        InstructionF: H: f0000000
435000.00ns INFO     cocotb.Main                        INSTR: H: f0000000
435000.00ns INFO     cocotb.Main                        PCPrime: D: 180 H: b4 B: 00000000000000000000000010110100
435000.00ns INFO     cocotb.Main                        PCF: D: 176 H: b0 B: 00000000000000000000000010110000
435000.00ns INFO     cocotb.Main                        RegWriteW: D: 1 H: 1 B: 1
435000.00ns INFO     cocotb.Main                        ALUControlE: D: 4 H: 4 B: 0100
435000.00ns INFO     cocotb.Main                        ShifterInput: D: 32 H: 20 B: 00000000000000000000000000100000
435000.00ns INFO     cocotb.Main                        RD2_S: D: 32 H: 20 B: 00000000000000000000000000100000
435000.00ns INFO     cocotb.Main                        CondE: D: 15 H: f B: 1111
435000.00ns INFO     cocotb.Main                        RD1: D: 32 H: 20 B: 00000000000000000000000000100000
435000.00ns INFO     cocotb.Main                        RD2: D: 32 H: 20 B: 00000000000000000000000000100000
435000.00ns INFO     cocotb.Main                        ForwardAE: D: 1 H: 1 B: 01
435000.00ns INFO     cocotb.Main                        ForwardBE: D: 1 H: 1 B: 01
435000.00ns INFO     cocotb.Main                        SrcAE: D: 32 H: 20 B: 00000000000000000000000000100000
435000.00ns INFO     cocotb.Main                        SrcBE: D: 32 H: 20 B: 00000000000000000000000000100000
435000.00ns INFO     cocotb.Main                        ALUResultE: D: 64 H: 40 B: 00000000000000000000000001000000
435000.00ns INFO     cocotb.Main                        ALUOutM: D: 64 H: 40 B: 00000000000000000000000001000000
435000.00ns INFO     cocotb.Main                        ALUOutW: D: 32 H: 20 B: 00000000000000000000000000100000
435000.00ns INFO     cocotb.Main                        OUT: D: 32 H: 20 B: 00000000000000000000000000100000
------------------
436000.00ns INFO     cocotb.regression                  hazard_unit ←[32mpassed←[49m←[39m
436000.00ns INFO     cocotb.regression                  ******************************************************************************************
                                                        ** TEST                            STATUS  SIM TIME (ns)  REAL TIME (s)  RATIO (ns/s) **
                                                        ******************************************************************************************
                                                        ** hazard_unit.hazard_unit      ←[32m PASS ←[49m←[39m     436000.00           1.04     419634.83  **
                                                        ******************************************************************************************
                                                        ** TESTS=1 PASS=1 FAIL=0 SKIP=0             436000.00           1.44     301738.83  **
                                                        ******************************************************************************************

make[1]: Leaving directory '/c/Users/Deniz/Documents/Pipelined-Project/project-tests/hazard_unit'
(base) PS C:\Users\Deniz\Documents\Pipelined-Project\project-tests\hazard_unit>
```

Figure 6. COCOTB results from the Memory and Write Back stages of the altered MOV instruction.

Hazard Unit

The implemented hazard unit for this project will deal with two different types of dangers to handle the difficulties brought on by risks in the pipeline.

Forwarding method will be used to mitigate data hazards and assure effective execution without adding extra cycles. Forwarding will be used to deliver the necessary data immediately from the executing instruction to the dependent instruction, removing the requirement for pausing, when a data risk occurs as a result of activities involving registers.

On the other hand, control hazards arise when the decision regarding the next instruction to fetch has not yet been determined at the time of the fetch operation. A **branch predictor** will be used by branch procedures that use immediate values, such as B, BL, and their conditional variations. To enhance overall performance, the branch predictor seeks to accurately predict how these branches will turn out. The predictor could, however, make a mistake in its forecasts. In these circumstances, it is necessary to **flush** the pipeline with the goal of discarding the incorrectly obtained instructions and resume execution using the results of the correct branch.

Matching and Forwarding of Operand AE:

- **Match_1E_M** checks if the register being read in the first execute stage (RA1E) matches the register being written to in the third memory stage (WA3M).
- **Match_1E_W** checks if RA1E matches WA3W, the register being written to in the third writeback stage (WA3W).
- If **Match_1E_M** is true and there is a register write in the memory stage (RegWriteM), the value from ALUOutM is forwarded to the first execute stage as the source operand AE (ForwardAE = 10).
- If **Match_1E_W** is true and there is a register write in the writeback stage (RegWriteW), the value from ResultW is forwarded to the first execute stage as the source operand AE (ForwardAE = 01).
- If neither **Match_1E_M** nor **Match_1E_W** is true, the source operand AE is obtained from the register file (ForwardAE = 00).

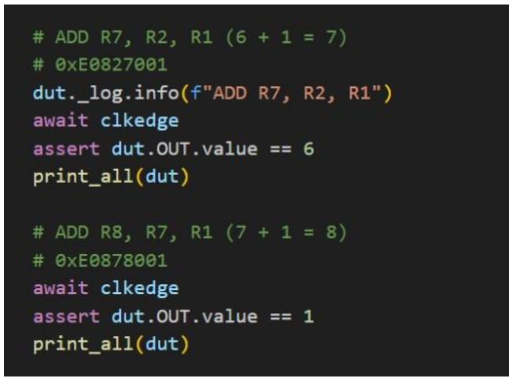
Matching and Forwarding of Operand BE:

- **Match_2E_M** checks if the register being read in the second execute stage (RA2E) matches the register being written to in the third memory stage (WA3M).
- **Match_2E_W** checks if RA2E matches WA3W, the register being written to in the third writeback stage (WA3W).
- If **Match_2E_M** is true and there is a register write in the memory stage (RegWriteM), the value from ALUOutM is forwarded to the second execute stage as the source operand BE (ForwardBE = 10).
- If **Match_2E_W** is true and there is a register write in the writeback stage (RegWriteW), the value from ResultW is forwarded to the second execute stage as the source operand BE (ForwardBE = 01).
- If neither **Match_2E_M** nor **Match_2E_W** is true, the source operand BE is obtained either from ExtImmE or the register file (depending on another multiplexer).

Hazard Unit Operations:

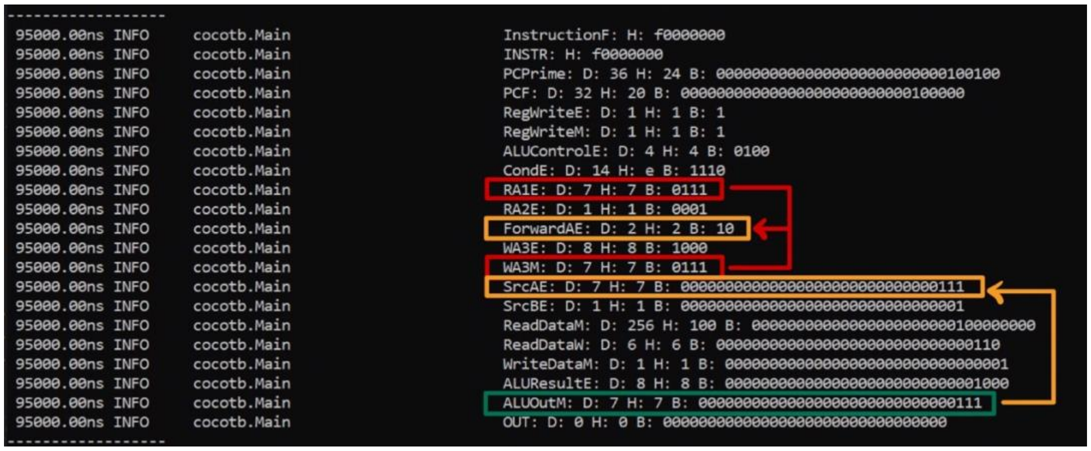
- **Match_12D_E** checks if either RA1D or RA2D matches WA3E, the register being written to in the third execute stage. It determines if there is a hazard in the decode stage due to instructions reading the same register that is being written in the execute stage.
- **LDRstall** is asserted if there is a match in **Match_12D_E** and MemtoRegE is true, indicating a data hazard caused by a memory operation. It causes a minimal amount of stalling to resolve the hazard.
- **BranchTakenE** is determined by BranchE and CondEx, indicating if a branch instruction is taken in the execute stage.
- **PCWrPendingF** is the combination of PCSrcD, PCSrcE, and PCSrcM, determining if there is a pending PC write in the fetch stage. This is used to stall the fetch stage and flush the decode stage.
- **StallF** is asserted when there is **LDRstall** or **PCWrPendingF** to stall the fetch stage. However, it is not asserted during PCSrcW to allow the write. It remains asserted continuously for three cycles once PCS=1 is decoded.
- **StallD** is asserted if there is **LDRstall**, causing stalling in the decode stage.
- **FlushD** is asserted when there is **PCWrPendingF**, PCSrcW, or **BranchTakenE**, indicating that the decode stage needs to be flushed.

- **FlushE** is asserted when there is **LDRstall** or **BranchTakenE**, signifying the need to flush the execute stage.



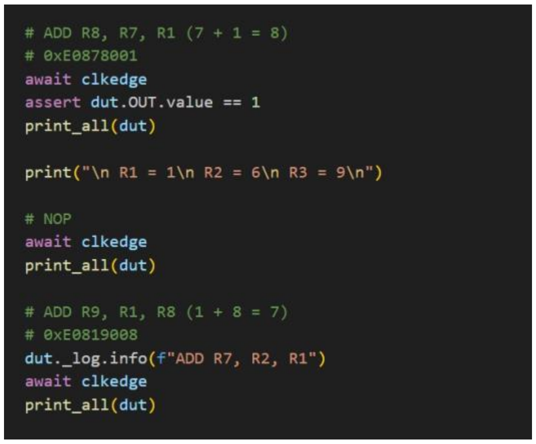
```
# ADD R7, R2, R1 (6 + 1 = 7)
# 0xE0827001
dut._log.info(f"ADD R7, R2, R1")
await clkedge
assert dut.OUT.value == 6
print_all(dut)

# ADD R8, R7, R1 (7 + 1 = 8)
# 0xE0878001
await clkedge
assert dut.OUT.value == 1
print_all(dut)
```

Figure 7. Testing forwarding (ForwardAE) with 2 ADD instructions run consecutively.



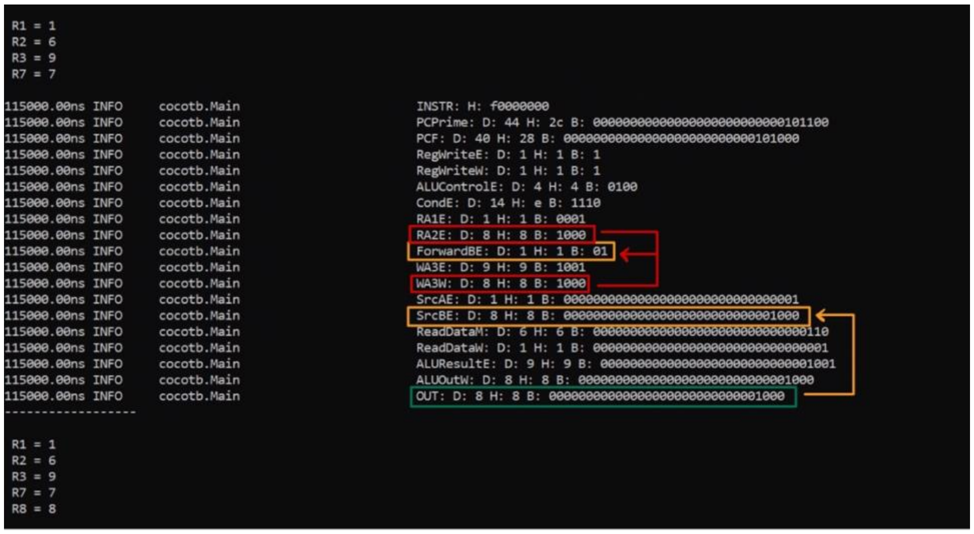
```
------------------
95000.00ns INFO     cocotb.Main                        InstructionF: H: f0000000
95000.00ns INFO     cocotb.Main                        INSTR: H: f0000000
95000.00ns INFO     cocotb.Main                        PCPrime: D: 36 H: 24 B: 00000000000000000000000000100100
95000.00ns INFO     cocotb.Main                        PCF: D: 32 H: 20 B: 00000000000000000000000000100000
95000.00ns INFO     cocotb.Main                        RegWriteE: D: 1 H: 1 B: 1
95000.00ns INFO     cocotb.Main                        RegWriteM: D: 1 H: 1 B: 1
95000.00ns INFO     cocotb.Main                        ALUControlE: D: 4 H: 4 B: 0100
95000.00ns INFO     cocotb.Main                        CondE: D: 14 H: e B: 1110
95000.00ns INFO     cocotb.Main                        RA1E: D: 7 H: 7 B: 0111
95000.00ns INFO     cocotb.Main                        RA2E: D: 1 H: 1 B: 0001
95000.00ns INFO     cocotb.Main                        ForwardAE: D: 2 H: 2 B: 10
95000.00ns INFO     cocotb.Main                        WA3E: D: 8 H: 8 B: 1000
95000.00ns INFO     cocotb.Main                        WA3M: D: 7 H: 7 B: 0111
95000.00ns INFO     cocotb.Main                        SrcAE: D: 7 H: 7 B: 00000000000000000000000000000111
95000.00ns INFO     cocotb.Main                        SrcBE: D: 1 H: 1 B: 00000000000000000000000000000001
95000.00ns INFO     cocotb.Main                        ReadDataM: D: 256 H: 100 B: 00000000000000000000000100000000
95000.00ns INFO     cocotb.Main                        ReadDataW: D: 6 H: 6 B: 00000000000000000000000000000110
95000.00ns INFO     cocotb.Main                        WriteDataM: D: 1 H: 1 B: 00000000000000000000000000000001
95000.00ns INFO     cocotb.Main                        ALUResultE: D: 8 H: 8 B: 00000000000000000000000000001000
95000.00ns INFO     cocotb.Main                        ALUOutM: D: 7 H: 7 B: 00000000000000000000000000000111
95000.00ns INFO     cocotb.Main                        OUT: D: 0 H: 0 B: 00000000000000000000000000000000
------------------
```

Figure 8. Testing forwarding (ForwardAE = 10) with COCOTB.



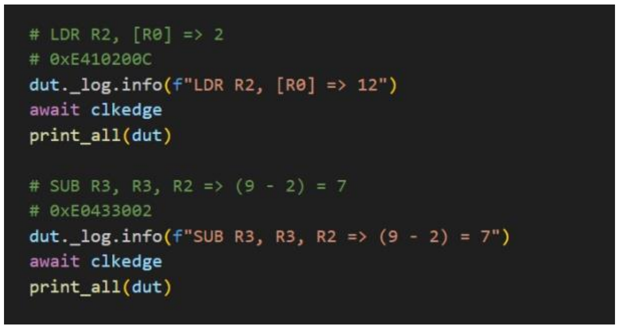
```
# ADD R8, R7, R1 (7 + 1 = 8)
# 0xE0878001
await clkedge
assert dut.OUT.value == 1
print_all(dut)

print("\n R1 = 1\n R2 = 6\n R3 = 9\n")

# NOP
await clkedge
print_all(dut)

# ADD R9, R1, R8 (1 + 8 = 7)
# 0xE0819008
dut._log.info(f"ADD R7, R2, R1")
await clkedge
print_all(dut)
```

Figure 9. Testing forwarding (ForwardBE) with 2 ADD instructions run consecutively.



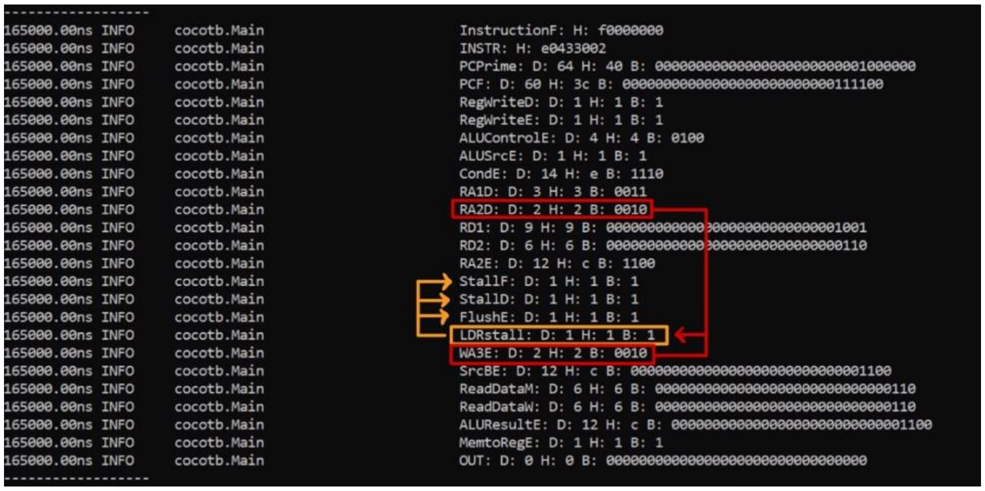
```
 R1 = 1
 R2 = 6
 R3 = 9
 R7 = 7

115000.00ns INFO     cocotb.Main                         INSTR: H: f0000000
115000.00ns INFO     cocotb.Main                         PCPrime: D: 44 H: 2c B: 00000000000000000000000000101100
115000.00ns INFO     cocotb.Main                         PCF: D: 40 H: 28 B: 00000000000000000000000000101000
115000.00ns INFO     cocotb.Main                         RegWriteE: D: 1 H: 1 B: 1
115000.00ns INFO     cocotb.Main                         RegWriteW: D: 1 H: 1 B: 1
115000.00ns INFO     cocotb.Main                         ALUControlE: D: 4 H: 4 B: 0100
115000.00ns INFO     cocotb.Main                         CondE: D: 14 H: e B: 1110
115000.00ns INFO     cocotb.Main                         RA1E: D: 1 H: 1 B: 0001
115000.00ns INFO     cocotb.Main                         RA2E: D: 8 H: 8 B: 1000
115000.00ns INFO     cocotb.Main                         ForwardBE: D: 1 H: 1 B: 01
115000.00ns INFO     cocotb.Main                         WA3E: D: 9 H: 9 B: 1001
115000.00ns INFO     cocotb.Main                         WA3W: D: 8 H: 8 B: 1000
115000.00ns INFO     cocotb.Main                         SrcAE: D: 1 H: 1 B: 00000000000000000000000000000001
115000.00ns INFO     cocotb.Main                         SrcBE: D: 8 H: 8 B: 00000000000000000000000000001000
115000.00ns INFO     cocotb.Main                         ReadDataM: D: 6 H: 6 B: 00000000000000000000000000000110
115000.00ns INFO     cocotb.Main                         ReadDataW: D: 1 H: 1 B: 00000000000000000000000000000001
115000.00ns INFO     cocotb.Main                         ALUResultE: D: 9 H: 9 B: 00000000000000000000000000001001
115000.00ns INFO     cocotb.Main                         ALUOutW: D: 8 H: 8 B: 00000000000000000000000000001000
115000.00ns INFO     cocotb.Main                         OUT: D: 8 H: 8 B: 00000000000000000000000000001000
------------------

 R1 = 1
 R2 = 6
 R3 = 9
 R7 = 7
 R8 = 8
```

Figure 10. Testing forwarding (ForwardBE = 01) with COCOTB.



```
# LDR R2, [R0] => 2
# 0xE410200C
dut._log.info(f"LDR R2, [R0] => 12")
await clkedge
print_all(dut)

# SUB R3, R3, R2 => (9 - 2) = 7
# 0xE0433002
dut._log.info(f"SUB R3, R3, R2 => (9 - 2) = 7")
await clkedge
print_all(dut)
```

Figure 11. Testing LDRstall.



```
------------------
165000.00ns INFO     cocotb.Main                         InstructionF: H: f0000000
165000.00ns INFO     cocotb.Main                         INSTR: H: e0433002
165000.00ns INFO     cocotb.Main                         PCPrime: D: 64 H: 40 B: 00000000000000000000000001000000
165000.00ns INFO     cocotb.Main                         PCF: D: 60 H: 3c B: 00000000000000000000000000111100
165000.00ns INFO     cocotb.Main                         RegWriteD: D: 1 H: 1 B: 1
165000.00ns INFO     cocotb.Main                         RegWriteE: D: 1 H: 1 B: 1
165000.00ns INFO     cocotb.Main                         ALUControlE: D: 4 H: 4 B: 0100
165000.00ns INFO     cocotb.Main                         ALUSrcE: D: 1 H: 1 B: 1
165000.00ns INFO     cocotb.Main                         CondE: D: 14 H: e B: 1110
165000.00ns INFO     cocotb.Main                         RA1D: D: 3 H: 3 B: 0011
165000.00ns INFO     cocotb.Main                         RA2D: D: 2 H: 2 B: 0010
165000.00ns INFO     cocotb.Main                         RD1: D: 9 H: 9 B: 00000000000000000000000000001001
165000.00ns INFO     cocotb.Main                         RD2: D: 6 H: 6 B: 00000000000000000000000000000110
165000.00ns INFO     cocotb.Main                         RA2E: D: 12 H: c B: 1100
165000.00ns INFO     cocotb.Main                         StallF: D: 1 H: 1 B: 1
165000.00ns INFO     cocotb.Main                         StallD: D: 1 H: 1 B: 1
165000.00ns INFO     cocotb.Main                         FlushE: D: 1 H: 1 B: 1
165000.00ns INFO     cocotb.Main                         LDRstall: D: 1 H: 1 B: 1
165000.00ns INFO     cocotb.Main                         WA3E: D: 2 H: 2 B: 0010
165000.00ns INFO     cocotb.Main                         SrcBE: D: 12 H: c B: 00000000000000000000000000001100
165000.00ns INFO     cocotb.Main                         ReadDataM: D: 6 H: 6 B: 00000000000000000000000000000110
165000.00ns INFO     cocotb.Main                         ReadDataW: D: 6 H: 6 B: 00000000000000000000000000000110
165000.00ns INFO     cocotb.Main                         ALUResultE: D: 12 H: c B: 00000000000000000000000000001100
165000.00ns INFO     cocotb.Main                         MemtoRegE: D: 1 H: 1 B: 1
165000.00ns INFO     cocotb.Main                         OUT: D: 0 H: 0 B: 00000000000000000000000000000000
------------------
```

Figure 12. Testing LDRstall COCOTB results.

Branch Prediction

Branch Target Buffer (BTB):

- The BTB is a cache-like structure that stores the Program Counter (PC) values and their corresponding branch target addresses (BTAs).
- During the Fetch cycle, the PC value of the current instruction is used as an index to access the BTB.
- If a match is found in the BTB, the predicted target address is obtained from the corresponding entry.
- The predicted target address is used to fetch the next instruction in the Fetch cycle, assuming the branch is taken.

Global Branch Register (GBR):

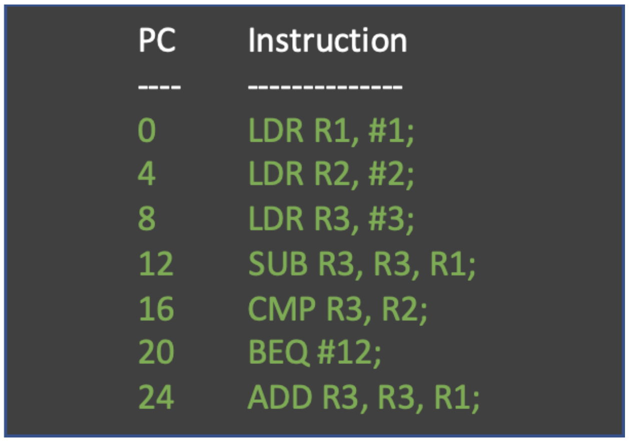
- The GBR is a register that maintains the historical outcomes of previous branch instructions.
- It is updated during the Execute cycle when the actual outcome of a branch instruction is determined.
- The GBR stores the branch history as a pattern of 1s and 0s, representing taken and not taken outcomes, respectively.
- The GBR is used in conjunction with the BTB to improve the accuracy of branch predictions.

Fetch and Execute Cycle:

- In the Fetch cycle, the PC value is used to fetch the next instruction.
- If the predicted target address is available from the BTB, the next instruction is fetched from that address.
- In the Execute cycle, the branch instruction is evaluated, and the actual outcome is determined.
- If the prediction is accurate (branch taken), the subsequent instructions are fetched from the predicted target address.
- However, if the prediction is incorrect (branch not taken), measures need to be taken to retrieve the correct BTA or PC + 4.

Measures for Incorrect Predictions:

- If the prediction is incorrect, the pipeline needs to be corrected to ensure the correct execution flow.
- One measure is to flush the incorrect instructions fetched based on the wrong prediction.
- The pipeline stages that follow the branch instruction, including the Fetch, Decode, and Execute stages, are flushed, discarding the incorrectly fetched instructions.
- The correct BTA or the PC + 4 (PC value incremented by 4) is then used to fetch the correct instructions.
- The pipeline continues execution from the correct target address, maintaining the correct program flow.



```
PC      Instruction
----    -------------
0       LDR R1, #1;
4       LDR R2, #2;
8       LDR R3, #3;
12      SUB R3, R3, R1;
16      CMP R3, R2;
20      BEQ #12;
24      ADD R3, R3, R1;
```

Figure 13. Sample subroutine for testing branch predictor.

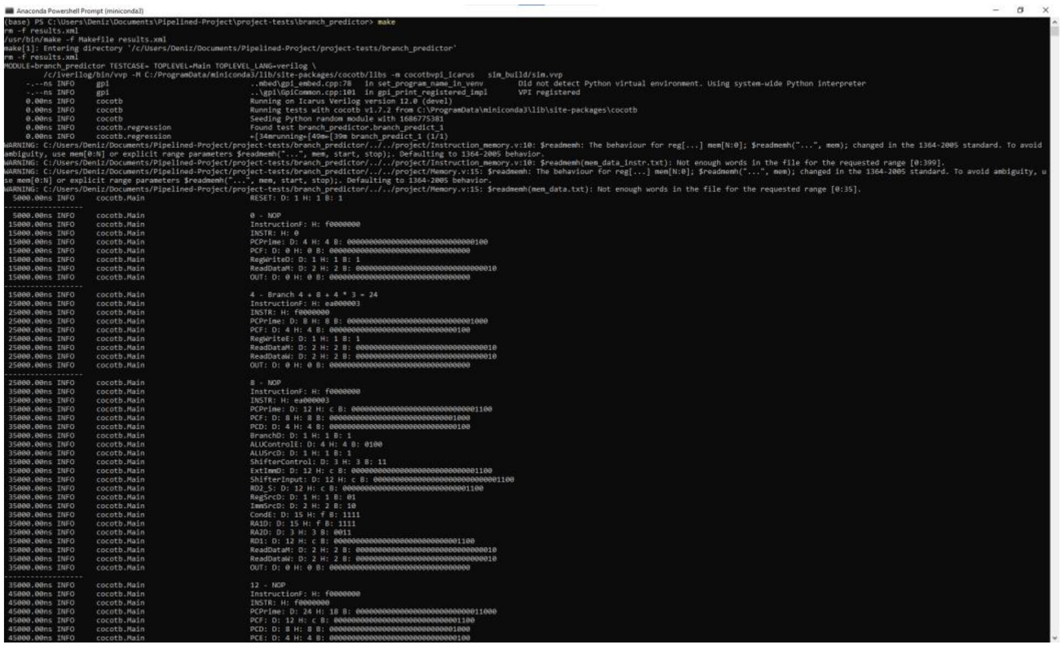
The sample subroutine in *Figure 13*, demonstrates the case when the branch predictor predicts that the branch will be taken, and the prediction is incorrect. Based on this prediction, the pipeline proceeds to fetch and decode instructions assuming the branch is taken, i.e., it will fetch the instruction at PC 12 next.

However, during the Execute stage, it is determined that the branch condition is not met, and the branch instruction is actually "not taken." The execution will continue with the subsequent instruction after the branch, which is ADD R3, R3, R1 at **PC 24**.

Since the branch prediction was incorrect, the pipeline needs to be corrected. The instructions fetched based on the wrong prediction need to be flushed. The correct program counter (PC) value, in this case, PC + 4 (PC 24 + 4 = 28), is used to fetch the correct instruction.

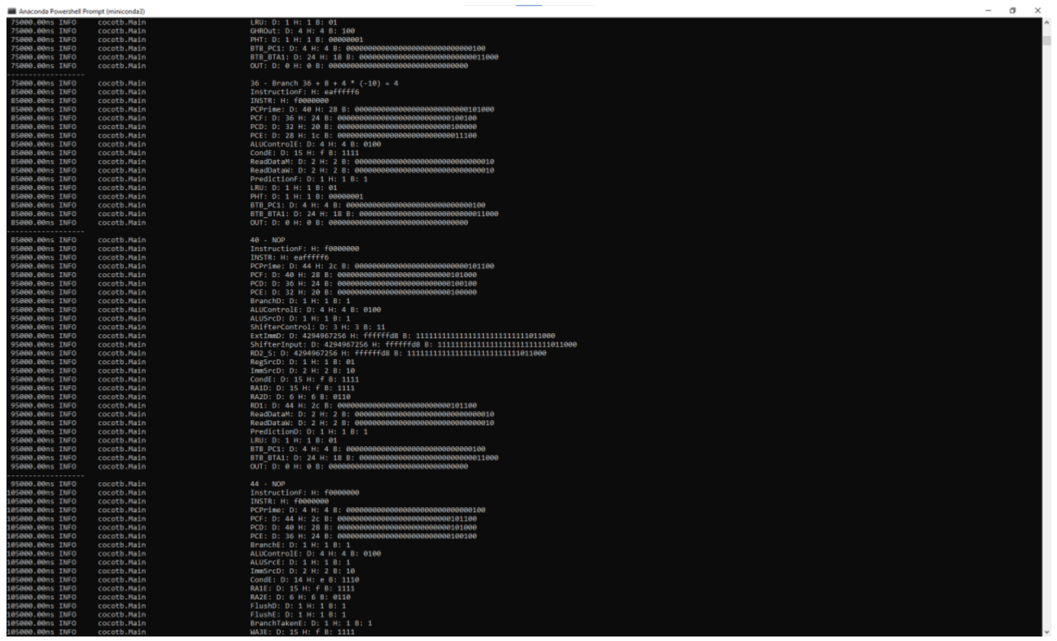
By flushing the pipeline and fetching from the correct PC value, the pipeline will resume execution with the correct instruction (ADD R3, R3, R1). This ensures that the program execution continues correctly based on the actual outcome of the branch instruction.

Sample Branch Predictor subroutine can be found in Figures 14-18. You can examine the Branch Target Buffer Output, BTB PC addresses that saved for Branch Predictor, BTB BTA addresses, GHR Outputs and Prediction signal generated from PTH. Note that: Only non-zero values are printed in this test environment.



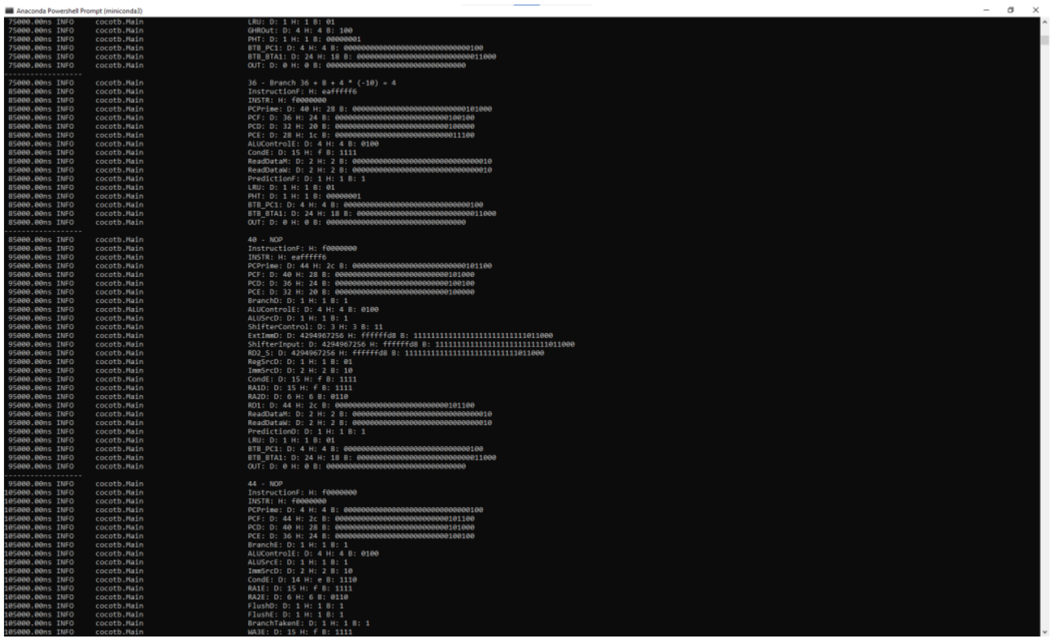
```
Anaconda Powershell Prompt (miniconda3)
(base) PS C:\Users\Deniz\Documents\Pipelined-Project\project-tests\branch_predictor> make
rm -f results.xml
/usr/bin/make -f Makefile results.xml
make[1]: Entering directory '/c/Users/Deniz/Documents/Pipelined-Project/project-tests/branch_predictor'
rm -f results.xml
MODULE=branch_predictor TESTCASE= TOPLEVEL=Main TOPLEVEL_LANG=verilog \
        /c/iverilog/bin/vvp -M C:/ProgramData/miniconda3/lib/site-packages/cocotb/libs -m cocotbvpi_icarus   sim_build/sim.vvp
     -.--ns INFO     gpi                                ..mbed\gpi_embed.cpp:78   in set_program_name_in_venv        Did not detect Python virtual environment. Using system-wide Python interpreter
     -.--ns INFO     gpi                                ..\gpi\GpiCommon.cpp:101  in gpi_print_registered_impl       VPI registered
     0.00ns INFO     cocotb                             Running on Icarus Verilog version 12.0 (devel)
     0.00ns INFO     cocotb                             Running tests with cocotb v1.7.2 from C:\ProgramData\miniconda3\lib\site-packages\cocotb
     0.00ns INFO     cocotb                             Seeding Python random module with 1686775381
     0.00ns INFO     cocotb.regression                  Found test branch_predictor.branch_predict_1
     0.00ns INFO     cocotb.regression                  ←[34mrunning←[49m←[39m branch_predict_1 (1/1)
WARNING: C:/Users/Deniz/Documents/Pipelined-Project/project-tests/branch_predictor/../../project/Instruction_memory.v:10: $readmemh: The behaviour for reg[...] mem[N:0]; $readmemh("...", mem); changed in the 1364-2005 standard. To avoid
ambiguity, use mem[0:N] or explicit range parameters $readmemh("...", mem, start, stop);. Defaulting to 1364-2005 behavior.
WARNING: C:/Users/Deniz/Documents/Pipelined-Project/project-tests/branch_predictor/../../project/Instruction_memory.v:10: $readmemh(mem_data_instr.txt): Not enough words in the file for the requested range [0:399].
WARNING: C:/Users/Deniz/Documents/Pipelined-Project/project-tests/branch_predictor/../../project/Memory.v:15: $readmemh: The behaviour for reg[...] mem[N:0]; $readmemh("...", mem); changed in the 1364-2005 standard. To avoid ambiguity, u
se mem[0:N] or explicit range parameters $readmemh("...", mem, start, stop);. Defaulting to 1364-2005 behavior.
WARNING: C:/Users/Deniz/Documents/Pipelined-Project/project-tests/branch_predictor/../../project/Memory.v:15: $readmemh(mem_data.txt): Not enough words in the file for the requested range [0:35].
  5000.00ns INFO     cocotb.Main                        RESET: D: 1 H: 1 B: 1
------------------
  5000.00ns INFO     cocotb.Main                        0 - NOP
 15000.00ns INFO     cocotb.Main                        InstructionF: H: f0000000
 15000.00ns INFO     cocotb.Main                        INSTR: H: 0
 15000.00ns INFO     cocotb.Main                        PCPrime: D: 4 H: 4 B: 00000000000000000000000000000100
 15000.00ns INFO     cocotb.Main                        PCF: D: 0 H: 0 B: 00000000000000000000000000000000
 15000.00ns INFO     cocotb.Main                        RegWriteD: D: 1 H: 1 B: 1
 15000.00ns INFO     cocotb.Main                        ReadDataM: D: 2 H: 2 B: 00000000000000000000000000000010
 15000.00ns INFO     cocotb.Main                        OUT: D: 0 H: 0 B: 00000000000000000000000000000000
------------------
 15000.00ns INFO     cocotb.Main                        4 - Branch 4 + 8 + 4 * 3 = 24
 25000.00ns INFO     cocotb.Main                        InstructionF: H: ea000003
 25000.00ns INFO     cocotb.Main                        INSTR: H: f0000000
 25000.00ns INFO     cocotb.Main                        PCPrime: D: 8 H: 8 B: 00000000000000000000000000001000
 25000.00ns INFO     cocotb.Main                        PCF: D: 4 H: 4 B: 00000000000000000000000000000100
 25000.00ns INFO     cocotb.Main                        RegWriteE: D: 1 H: 1 B: 1
 25000.00ns INFO     cocotb.Main                        ReadDataM: D: 2 H: 2 B: 00000000000000000000000000000010
 25000.00ns INFO     cocotb.Main                        ReadDataW: D: 2 H: 2 B: 00000000000000000000000000000010
 25000.00ns INFO     cocotb.Main                        OUT: D: 0 H: 0 B: 00000000000000000000000000000000
------------------
 25000.00ns INFO     cocotb.Main                        8 - NOP
 35000.00ns INFO     cocotb.Main                        InstructionF: H: f0000000
 35000.00ns INFO     cocotb.Main                        INSTR: H: ea000003
 35000.00ns INFO     cocotb.Main                        PCPrime: D: 12 H: c B: 00000000000000000000000000001100
 35000.00ns INFO     cocotb.Main                        PCF: D: 8 H: 8 B: 00000000000000000000000000001000
 35000.00ns INFO     cocotb.Main                        PCD: D: 4 H: 4 B: 00000000000000000000000000000100
 35000.00ns INFO     cocotb.Main                        BranchD: D: 1 H: 1 B: 1
 35000.00ns INFO     cocotb.Main                        ALUControlE: D: 4 H: 4 B: 0100
 35000.00ns INFO     cocotb.Main                        ALUSrcD: D: 1 H: 1 B: 1
 35000.00ns INFO     cocotb.Main                        ShifterControl: D: 3 H: 3 B: 11
 35000.00ns INFO     cocotb.Main                        ExtImmD: D: 12 H: c B: 00000000000000000000000000001100
 35000.00ns INFO     cocotb.Main                        ShifterInput: D: 12 H: c B: 00000000000000000000000000001100
 35000.00ns INFO     cocotb.Main                        RD2_S: D: 12 H: c B: 00000000000000000000000000001100
 35000.00ns INFO     cocotb.Main                        RegSrcD: D: 1 H: 1 B: 01
 35000.00ns INFO     cocotb.Main                        ImmSrcD: D: 2 H: 2 B: 10
 35000.00ns INFO     cocotb.Main                        CondE: D: 15 H: f B: 1111
 35000.00ns INFO     cocotb.Main                        RA1D: D: 15 H: f B: 1111
 35000.00ns INFO     cocotb.Main                        RA2D: D: 3 H: 3 B: 0011
 35000.00ns INFO     cocotb.Main                        RD1: D: 12 H: c B: 00000000000000000000000000001100
 35000.00ns INFO     cocotb.Main                        ReadDataM: D: 2 H: 2 B: 00000000000000000000000000000010
 35000.00ns INFO     cocotb.Main                        ReadDataW: D: 2 H: 2 B: 00000000000000000000000000000010
 35000.00ns INFO     cocotb.Main                        OUT: D: 0 H: 0 B: 00000000000000000000000000000000
------------------
 35000.00ns INFO     cocotb.Main                        12 - NOP
 45000.00ns INFO     cocotb.Main                        InstructionF: H: f0000000
 45000.00ns INFO     cocotb.Main                        INSTR: H: f0000000
 45000.00ns INFO     cocotb.Main                        PCPrime: D: 24 H: 18 B: 00000000000000000000000000011000
 45000.00ns INFO     cocotb.Main                        PCF: D: 12 H: c B: 00000000000000000000000000001100
 45000.00ns INFO     cocotb.Main                        PCD: D: 8 H: 8 B: 00000000000000000000000000001000
 45000.00ns INFO     cocotb.Main                        PCE: D: 4 H: 4 B: 00000000000000000000000000000100
```

Figure 14. COCOTB Test procedure for the Branch Predictor No.1.



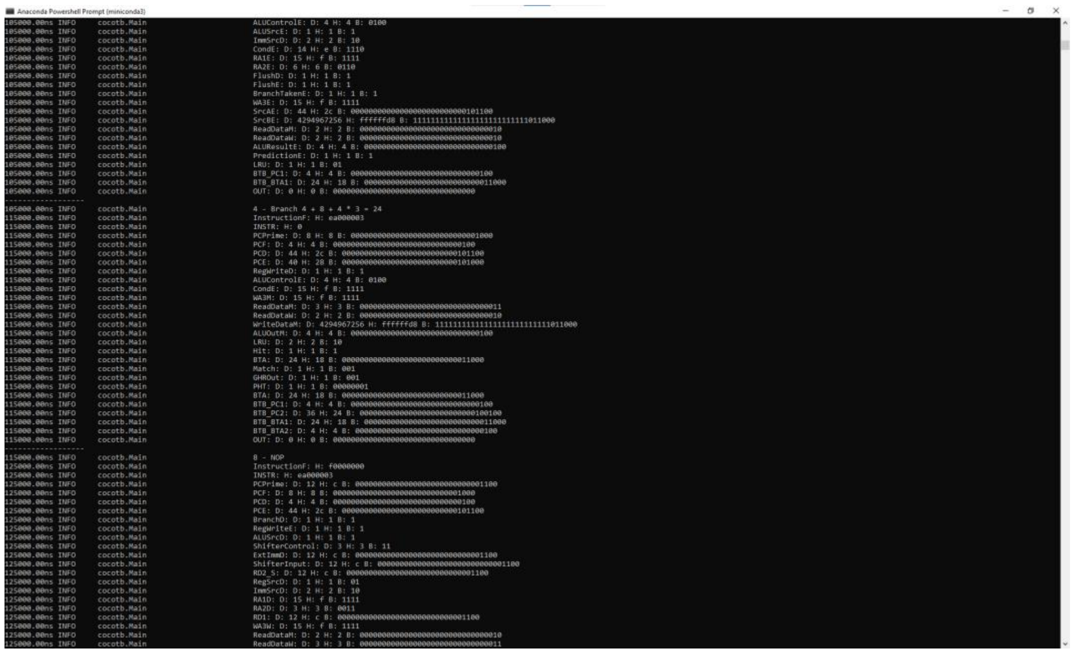
```
Anaconda Powershell Prompt (miniconda3)
 75000.00ns INFO     cocotb.Main                        LRU: D: 1 H: 1 B: 01
 75000.00ns INFO     cocotb.Main                        GHROut: D: 4 H: 4 B: 100
 75000.00ns INFO     cocotb.Main                        PHT: D: 1 H: 1 B: 00000001
 75000.00ns INFO     cocotb.Main                        BTB_PC1: D: 4 H: 4 B: 00000000000000000000000000000100
 75000.00ns INFO     cocotb.Main                        BTB_BTA1: D: 24 H: 18 B: 00000000000000000000000000011000
 75000.00ns INFO     cocotb.Main                        OUT: D: 0 H: 0 B: 00000000000000000000000000000000
------------------
 75000.00ns INFO     cocotb.Main                        36 - Branch 36 + 8 + 4 * (-10) = 4
 85000.00ns INFO     cocotb.Main                        InstructionF: H: eafffff6
 85000.00ns INFO     cocotb.Main                        INSTR: H: f0000000
 85000.00ns INFO     cocotb.Main                        PCPrime: D: 40 H: 28 B: 00000000000000000000000000101000
 85000.00ns INFO     cocotb.Main                        PCF: D: 36 H: 24 B: 00000000000000000000000000100100
 85000.00ns INFO     cocotb.Main                        PCD: D: 32 H: 20 B: 00000000000000000000000000100000
 85000.00ns INFO     cocotb.Main                        PCE: D: 28 H: 1c B: 00000000000000000000000000011100
 85000.00ns INFO     cocotb.Main                        ALUControlE: D: 4 H: 4 B: 0100
 85000.00ns INFO     cocotb.Main                        CondE: D: 15 H: f B: 1111
 85000.00ns INFO     cocotb.Main                        ReadDataM: D: 2 H: 2 B: 00000000000000000000000000000010
 85000.00ns INFO     cocotb.Main                        ReadDataW: D: 2 H: 2 B: 00000000000000000000000000000010
 85000.00ns INFO     cocotb.Main                        PredictionF: D: 1 H: 1 B: 1
 85000.00ns INFO     cocotb.Main                        LRU: D: 1 H: 1 B: 01
 85000.00ns INFO     cocotb.Main                        PHT: D: 1 H: 1 B: 00000001
 85000.00ns INFO     cocotb.Main                        BTB_PC1: D: 4 H: 4 B: 00000000000000000000000000000100
 85000.00ns INFO     cocotb.Main                        BTB_BTA1: D: 24 H: 18 B: 00000000000000000000000000011000
 85000.00ns INFO     cocotb.Main                        OUT: D: 0 H: 0 B: 00000000000000000000000000000000
------------------
 85000.00ns INFO     cocotb.Main                        40 - NOP
 95000.00ns INFO     cocotb.Main                        InstructionF: H: f0000000
 95000.00ns INFO     cocotb.Main                        INSTR: H: eafffff6
 95000.00ns INFO     cocotb.Main                        PCPrime: D: 44 H: 2c B: 00000000000000000000000000101100
 95000.00ns INFO     cocotb.Main                        PCF: D: 40 H: 28 B: 00000000000000000000000000101000
 95000.00ns INFO     cocotb.Main                        PCD: D: 36 H: 24 B: 00000000000000000000000000100100
 95000.00ns INFO     cocotb.Main                        PCE: D: 32 H: 20 B: 00000000000000000000000000100000
 95000.00ns INFO     cocotb.Main                        BranchD: D: 1 H: 1 B: 1
 95000.00ns INFO     cocotb.Main                        ALUControlE: D: 4 H: 4 B: 0100
 95000.00ns INFO     cocotb.Main                        ALUSrcD: D: 1 H: 1 B: 1
 95000.00ns INFO     cocotb.Main                        ShifterControl: D: 3 H: 3 B: 11
 95000.00ns INFO     cocotb.Main                        ExtImmD: D: 4294967256 H: ffffffd8 B: 11111111111111111111111111011000
 95000.00ns INFO     cocotb.Main                        ShifterInput: D: 4294967256 H: ffffffd8 B: 11111111111111111111111111011000
 95000.00ns INFO     cocotb.Main                        RD2_S: D: 4294967256 H: ffffffd8 B: 11111111111111111111111111011000
 95000.00ns INFO     cocotb.Main                        RegSrcD: D: 1 H: 1 B: 01
 95000.00ns INFO     cocotb.Main                        ImmSrcD: D: 2 H: 2 B: 10
 95000.00ns INFO     cocotb.Main                        CondE: D: 15 H: f B: 1111
 95000.00ns INFO     cocotb.Main                        RA1D: D: 15 H: f B: 1111
 95000.00ns INFO     cocotb.Main                        RA2D: D: 6 H: 6 B: 0110
 95000.00ns INFO     cocotb.Main                        RD1: D: 44 H: 2c B: 00000000000000000000000000101100
 95000.00ns INFO     cocotb.Main                        ReadDataM: D: 2 H: 2 B: 00000000000000000000000000000010
 95000.00ns INFO     cocotb.Main                        ReadDataW: D: 2 H: 2 B: 00000000000000000000000000000010
 95000.00ns INFO     cocotb.Main                        PredictionD: D: 1 H: 1 B: 1
 95000.00ns INFO     cocotb.Main                        LRU: D: 1 H: 1 B: 01
 95000.00ns INFO     cocotb.Main                        BTB_PC1: D: 4 H: 4 B: 00000000000000000000000000000100
 95000.00ns INFO     cocotb.Main                        BTB_BTA1: D: 24 H: 18 B: 00000000000000000000000000011000
 95000.00ns INFO     cocotb.Main                        OUT: D: 0 H: 0 B: 00000000000000000000000000000000
------------------
 95000.00ns INFO     cocotb.Main                        44 - NOP
105000.00ns INFO     cocotb.Main                        InstructionF: H: f0000000
105000.00ns INFO     cocotb.Main                        INSTR: H: f0000000
105000.00ns INFO     cocotb.Main                        PCPrime: D: 4 H: 4 B: 00000000000000000000000000000100
105000.00ns INFO     cocotb.Main                        PCF: D: 44 H: 2c B: 00000000000000000000000000101100
105000.00ns INFO     cocotb.Main                        PCD: D: 40 H: 28 B: 00000000000000000000000000101000
105000.00ns INFO     cocotb.Main                        PCE: D: 36 H: 24 B: 00000000000000000000000000100100
105000.00ns INFO     cocotb.Main                        BranchE: D: 1 H: 1 B: 1
105000.00ns INFO     cocotb.Main                        ALUControlE: D: 4 H: 4 B: 0100
105000.00ns INFO     cocotb.Main                        ALUSrcE: D: 1 H: 1 B: 1
105000.00ns INFO     cocotb.Main                        ImmSrcD: D: 2 H: 2 B: 10
105000.00ns INFO     cocotb.Main                        CondE: D: 14 H: e B: 1110
105000.00ns INFO     cocotb.Main                        RA1E: D: 15 H: f B: 1111
105000.00ns INFO     cocotb.Main                        RA2E: D: 6 H: 6 B: 0110
105000.00ns INFO     cocotb.Main                        FlushD: D: 1 H: 1 B: 1
105000.00ns INFO     cocotb.Main                        FlushE: D: 1 H: 1 B: 1
105000.00ns INFO     cocotb.Main                        BranchTakenE: D: 1 H: 1 B: 1
105000.00ns INFO     cocotb.Main                        WA3E: D: 15 H: f B: 1111
```

Figure 15. COCOTB Test procedure for the Branch Predictor No.2.



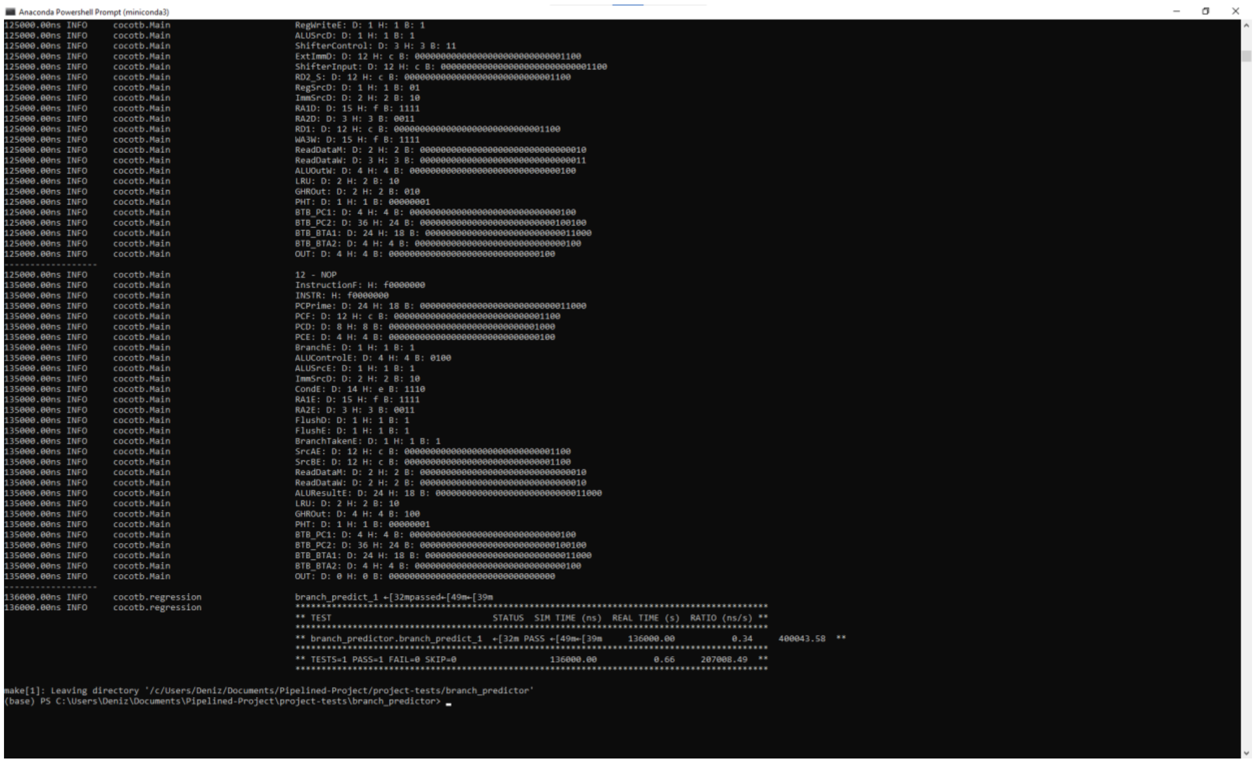
```
 75000.00ns INFO     cocotb.Main                          LRU: D: 1 H: 1 B: 01
 75000.00ns INFO     cocotb.Main                          GHROut: D: 4 H: 4 B: 100
 75000.00ns INFO     cocotb.Main                          PHT: D: 1 H: 1 B: 00000001
 75000.00ns INFO     cocotb.Main                          BTB_PC1: D: 4 H: 4 B: 00000000000000000000000000000100
 75000.00ns INFO     cocotb.Main                          BTB_BTA1: D: 24 H: 18 B: 00000000000000000000000000011000
 75000.00ns INFO     cocotb.Main                          OUT: D: 0 H: 0 B: 00000000000000000000000000000000
------------------
 75000.00ns INFO     cocotb.Main                          36 - Branch 36 + 8 + 4 * (-10) = 4
 85000.00ns INFO     cocotb.Main                          InstructionF: H: eafffff6
 85000.00ns INFO     cocotb.Main                          INSTR: H: f0000000
 85000.00ns INFO     cocotb.Main                          PCPrime: D: 40 H: 28 B: 00000000000000000000000000101000
 85000.00ns INFO     cocotb.Main                          PCF: D: 36 H: 24 B: 00000000000000000000000000100100
 85000.00ns INFO     cocotb.Main                          PCD: D: 32 H: 20 B: 00000000000000000000000000100000
 85000.00ns INFO     cocotb.Main                          PCE: D: 28 H: 1c B: 00000000000000000000000000011100
 85000.00ns INFO     cocotb.Main                          ALUControlE: D: 4 H: 4 B: 0100
 85000.00ns INFO     cocotb.Main                          CondE: D: 15 H: f B: 1111
 85000.00ns INFO     cocotb.Main                          ReadDataM: D: 2 H: 2 B: 00000000000000000000000000000010
 85000.00ns INFO     cocotb.Main                          ReadDataW: D: 2 H: 2 B: 00000000000000000000000000000010
 85000.00ns INFO     cocotb.Main                          PredictionF: D: 1 H: 1 B: 1
 85000.00ns INFO     cocotb.Main                          LRU: D: 1 H: 1 B: 01
 85000.00ns INFO     cocotb.Main                          PHT: D: 1 H: 1 B: 00000001
 85000.00ns INFO     cocotb.Main                          BTB_PC1: D: 4 H: 4 B: 00000000000000000000000000000100
 85000.00ns INFO     cocotb.Main                          BTB_BTA1: D: 24 H: 18 B: 00000000000000000000000000011000
 85000.00ns INFO     cocotb.Main                          OUT: D: 0 H: 0 B: 00000000000000000000000000000000
------------------
 85000.00ns INFO     cocotb.Main                          40 - NOP
 95000.00ns INFO     cocotb.Main                          InstructionF: H: f0000000
 95000.00ns INFO     cocotb.Main                          INSTR: H: eafffff6
 95000.00ns INFO     cocotb.Main                          PCPrime: D: 44 H: 2c B: 00000000000000000000000000101100
 95000.00ns INFO     cocotb.Main                          PCF: D: 40 H: 28 B: 00000000000000000000000000101000
 95000.00ns INFO     cocotb.Main                          PCD: D: 36 H: 24 B: 00000000000000000000000000100100
 95000.00ns INFO     cocotb.Main                          PCE: D: 32 H: 20 B: 00000000000000000000000000100000
 95000.00ns INFO     cocotb.Main                          BranchD: D: 1 H: 1 B: 1
 95000.00ns INFO     cocotb.Main                          ALUControlE: D: 4 H: 4 B: 0100
 95000.00ns INFO     cocotb.Main                          ALUSrcD: D: 1 H: 1 B: 1
 95000.00ns INFO     cocotb.Main                          ShifterControl: D: 3 H: 3 B: 11
 95000.00ns INFO     cocotb.Main                          ExtImmD: D: 4294967256 H: ffffffd8 B: 11111111111111111111111111011000
 95000.00ns INFO     cocotb.Main                          ShifterInput: D: 4294967256 H: ffffffd8 B: 11111111111111111111111111011000
 95000.00ns INFO     cocotb.Main                          RD2_S: D: 4294967256 H: ffffffd8 B: 11111111111111111111111111011000
 95000.00ns INFO     cocotb.Main                          RegSrcD: D: 1 H: 1 B: 01
 95000.00ns INFO     cocotb.Main                          ImmSrcD: D: 2 H: 2 B: 10
 95000.00ns INFO     cocotb.Main                          CondE: D: 15 H: f B: 1111
 95000.00ns INFO     cocotb.Main                          RA1D: D: 15 H: f B: 1111
 95000.00ns INFO     cocotb.Main                          RA2D: D: 6 H: 6 B: 0110
 95000.00ns INFO     cocotb.Main                          RD1: D: 44 H: 2c B: 00000000000000000000000000101100
 95000.00ns INFO     cocotb.Main                          ReadDataM: D: 2 H: 2 B: 00000000000000000000000000000010
 95000.00ns INFO     cocotb.Main                          ReadDataW: D: 2 H: 2 B: 00000000000000000000000000000010
 95000.00ns INFO     cocotb.Main                          PredictionD: D: 1 H: 1 B: 1
 95000.00ns INFO     cocotb.Main                          LRU: D: 1 H: 1 B: 01
 95000.00ns INFO     cocotb.Main                          BTB_PC1: D: 4 H: 4 B: 00000000000000000000000000000100
 95000.00ns INFO     cocotb.Main                          BTB_BTA1: D: 24 H: 18 B: 00000000000000000000000000011000
 95000.00ns INFO     cocotb.Main                          OUT: D: 0 H: 0 B: 00000000000000000000000000000000
------------------
 95000.00ns INFO     cocotb.Main                          44 - NOP
105000.00ns INFO     cocotb.Main                          InstructionF: H: f0000000
105000.00ns INFO     cocotb.Main                          INSTR: H: f0000000
105000.00ns INFO     cocotb.Main                          PCPrime: D: 4 H: 4 B: 00000000000000000000000000000100
105000.00ns INFO     cocotb.Main                          PCF: D: 44 H: 2c B: 00000000000000000000000000101100
105000.00ns INFO     cocotb.Main                          PCD: D: 40 H: 28 B: 00000000000000000000000000101000
105000.00ns INFO     cocotb.Main                          PCE: D: 36 H: 24 B: 00000000000000000000000000100100
105000.00ns INFO     cocotb.Main                          BranchE: D: 1 H: 1 B: 1
105000.00ns INFO     cocotb.Main                          ALUControlE: D: 4 H: 4 B: 0100
105000.00ns INFO     cocotb.Main                          ALUSrcE: D: 1 H: 1 B: 1
105000.00ns INFO     cocotb.Main                          ImmSrcD: D: 2 H: 2 B: 10
105000.00ns INFO     cocotb.Main                          CondE: D: 14 H: e B: 1110
105000.00ns INFO     cocotb.Main                          RA1E: D: 15 H: f B: 1111
105000.00ns INFO     cocotb.Main                          RA2E: D: 6 H: 6 B: 0110
105000.00ns INFO     cocotb.Main                          FlushD: D: 1 H: 1 B: 1
105000.00ns INFO     cocotb.Main                          FlushE: D: 1 H: 1 B: 1
105000.00ns INFO     cocotb.Main                          BranchTakenE: D: 1 H: 1 B: 1
105000.00ns INFO     cocotb.Main                          WA3E: D: 15 H: f B: 1111
```

Figure 16. COCOTB Test procedure for the Branch Predictor No.3.



```
105000.00ns INFO     cocotb.Main                          ALUControlE: D: 4 H: 4 B: 0100
105000.00ns INFO     cocotb.Main                          ALUSrcE: D: 1 H: 1 B: 1
105000.00ns INFO     cocotb.Main                          ImmSrcD: D: 2 H: 2 B: 10
105000.00ns INFO     cocotb.Main                          CondE: D: 14 H: e B: 1110
105000.00ns INFO     cocotb.Main                          RA1E: D: 15 H: f B: 1111
105000.00ns INFO     cocotb.Main                          RA2E: D: 6 H: 6 B: 0110
105000.00ns INFO     cocotb.Main                          FlushD: D: 1 H: 1 B: 1
105000.00ns INFO     cocotb.Main                          FlushE: D: 1 H: 1 B: 1
105000.00ns INFO     cocotb.Main                          BranchTakenE: D: 1 H: 1 B: 1
105000.00ns INFO     cocotb.Main                          WA3E: D: 15 H: f B: 1111
105000.00ns INFO     cocotb.Main                          SrcAE: D: 44 H: 2c B: 00000000000000000000000000101100
105000.00ns INFO     cocotb.Main                          SrcBE: D: 4294967256 H: ffffffd8 B: 11111111111111111111111111011000
105000.00ns INFO     cocotb.Main                          ReadDataM: D: 2 H: 2 B: 00000000000000000000000000000010
105000.00ns INFO     cocotb.Main                          ReadDataW: D: 2 H: 2 B: 00000000000000000000000000000010
105000.00ns INFO     cocotb.Main                          ALUResultE: D: 4 H: 4 B: 00000000000000000000000000000100
105000.00ns INFO     cocotb.Main                          PredictionE: D: 1 H: 1 B: 1
105000.00ns INFO     cocotb.Main                          LRU: D: 1 H: 1 B: 01
105000.00ns INFO     cocotb.Main                          BTB_PC1: D: 4 H: 4 B: 00000000000000000000000000000100
105000.00ns INFO     cocotb.Main                          BTB_BTA1: D: 24 H: 18 B: 00000000000000000000000000011000
105000.00ns INFO     cocotb.Main                          OUT: D: 0 H: 0 B: 00000000000000000000000000000000
------------------
105000.00ns INFO     cocotb.Main                          4 - Branch 4 + 8 + 4 * 3 = 24
115000.00ns INFO     cocotb.Main                          InstructionF: H: ea000003
115000.00ns INFO     cocotb.Main                          INSTR: H: 0
115000.00ns INFO     cocotb.Main                          PCPrime: D: 8 H: 8 B: 00000000000000000000000000001000
115000.00ns INFO     cocotb.Main                          PCF: D: 4 H: 4 B: 00000000000000000000000000000100
115000.00ns INFO     cocotb.Main                          PCD: D: 44 H: 2c B: 00000000000000000000000000101100
115000.00ns INFO     cocotb.Main                          PCE: D: 40 H: 28 B: 00000000000000000000000000101000
115000.00ns INFO     cocotb.Main                          RegWriteD: D: 1 H: 1 B: 1
115000.00ns INFO     cocotb.Main                          ALUControlE: D: 4 H: 4 B: 0100
115000.00ns INFO     cocotb.Main                          CondE: D: 15 H: f B: 1111
115000.00ns INFO     cocotb.Main                          WA3M: D: 15 H: f B: 1111
115000.00ns INFO     cocotb.Main                          ReadDataM: D: 3 H: 3 B: 00000000000000000000000000000011
115000.00ns INFO     cocotb.Main                          ReadDataW: D: 2 H: 2 B: 00000000000000000000000000000010
115000.00ns INFO     cocotb.Main                          WriteDataM: D: 4294967256 H: ffffffd8 B: 11111111111111111111111111011000
115000.00ns INFO     cocotb.Main                          ALUOutM: D: 4 H: 4 B: 00000000000000000000000000000100
115000.00ns INFO     cocotb.Main                          LRU: D: 2 H: 2 B: 10
115000.00ns INFO     cocotb.Main                          Hit: D: 1 H: 1 B: 1
115000.00ns INFO     cocotb.Main                          BTA: D: 24 H: 18 B: 00000000000000000000000000011000
115000.00ns INFO     cocotb.Main                          Match: D: 1 H: 1 B: 001
115000.00ns INFO     cocotb.Main                          GHROut: D: 1 H: 1 B: 001
115000.00ns INFO     cocotb.Main                          PHT: D: 1 H: 1 B: 00000001
115000.00ns INFO     cocotb.Main                          BTA: D: 24 H: 18 B: 00000000000000000000000000011000
115000.00ns INFO     cocotb.Main                          BTB_PC1: D: 4 H: 4 B: 00000000000000000000000000000100
115000.00ns INFO     cocotb.Main                          BTB_PC2: D: 36 H: 24 B: 00000000000000000000000000100100
115000.00ns INFO     cocotb.Main                          BTB_BTA1: D: 24 H: 18 B: 00000000000000000000000000011000
115000.00ns INFO     cocotb.Main                          BTB_BTA2: D: 4 H: 4 B: 00000000000000000000000000000100
115000.00ns INFO     cocotb.Main                          OUT: D: 0 H: 0 B: 00000000000000000000000000000000
------------------
115000.00ns INFO     cocotb.Main                          8 - NOP
125000.00ns INFO     cocotb.Main                          InstructionF: H: f0000000
125000.00ns INFO     cocotb.Main                          INSTR: H: ea000003
125000.00ns INFO     cocotb.Main                          PCPrime: D: 12 H: c B: 00000000000000000000000000001100
125000.00ns INFO     cocotb.Main                          PCF: D: 8 H: 8 B: 00000000000000000000000000001000
125000.00ns INFO     cocotb.Main                          PCD: D: 4 H: 4 B: 00000000000000000000000000000100
125000.00ns INFO     cocotb.Main                          PCE: D: 44 H: 2c B: 00000000000000000000000000101100
125000.00ns INFO     cocotb.Main                          BranchD: D: 1 H: 1 B: 1
125000.00ns INFO     cocotb.Main                          RegWriteE: D: 1 H: 1 B: 1
125000.00ns INFO     cocotb.Main                          ALUSrcD: D: 1 H: 1 B: 1
125000.00ns INFO     cocotb.Main                          ShifterControl: D: 3 H: 3 B: 11
125000.00ns INFO     cocotb.Main                          ExtImmD: D: 12 H: c B: 00000000000000000000000000001100
125000.00ns INFO     cocotb.Main                          ShifterInput: D: 12 H: c B: 00000000000000000000000000001100
125000.00ns INFO     cocotb.Main                          RD2_S: D: 12 H: c B: 00000000000000000000000000001100
125000.00ns INFO     cocotb.Main                          RegSrcD: D: 1 H: 1 B: 01
125000.00ns INFO     cocotb.Main                          ImmSrcD: D: 2 H: 2 B: 10
125000.00ns INFO     cocotb.Main                          RA1D: D: 15 H: f B: 1111
125000.00ns INFO     cocotb.Main                          RA2D: D: 3 H: 3 B: 0011
125000.00ns INFO     cocotb.Main                          RD1: D: 12 H: c B: 00000000000000000000000000001100
125000.00ns INFO     cocotb.Main                          WA3W: D: 15 H: f B: 1111
125000.00ns INFO     cocotb.Main                          ReadDataM: D: 2 H: 2 B: 00000000000000000000000000000010
125000.00ns INFO     cocotb.Main                          ReadDataW: D: 3 H: 3 B: 00000000000000000000000000000011
```

Figure 17. COCOTB Test procedure for the Branch Predictor No.4.


Anaconda Powershell Prompt (miniconda3)

```
125000.00ns INFO     cocotb.Main                        RegWriteE: D: 1 H: 1 B: 1
125000.00ns INFO     cocotb.Main                        ALUSrcD: D: 1 H: 1 B: 1
125000.00ns INFO     cocotb.Main                        ShifterControl: D: 3 H: 3 B: 11
125000.00ns INFO     cocotb.Main                        ExtImmD: D: 12 H: c B: 00000000000000000000000000001100
125000.00ns INFO     cocotb.Main                        ShifterInput: D: 12 H: c B: 00000000000000000000000000001100
125000.00ns INFO     cocotb.Main                        RD2_S: D: 12 H: c B: 00000000000000000000000000001100
125000.00ns INFO     cocotb.Main                        RegSrcD: D: 1 H: 1 B: 01
125000.00ns INFO     cocotb.Main                        ImmSrcD: D: 2 H: 2 B: 10
125000.00ns INFO     cocotb.Main                        RA1D: D: 15 H: f B: 1111
125000.00ns INFO     cocotb.Main                        RA2D: D: 3 H: 3 B: 0011
125000.00ns INFO     cocotb.Main                        RD1: D: 12 H: c B: 00000000000000000000000000001100
125000.00ns INFO     cocotb.Main                        WA3W: D: 15 H: f B: 1111
125000.00ns INFO     cocotb.Main                        ReadDataM: D: 2 H: 2 B: 00000000000000000000000000000010
125000.00ns INFO     cocotb.Main                        ReadDataW: D: 3 H: 3 B: 00000000000000000000000000000011
125000.00ns INFO     cocotb.Main                        ALUOutW: D: 4 H: 4 B: 00000000000000000000000000000100
125000.00ns INFO     cocotb.Main                        LRU: D: 2 H: 2 B: 10
125000.00ns INFO     cocotb.Main                        GHROut: D: 2 H: 2 B: 010
125000.00ns INFO     cocotb.Main                        PHT: D: 1 H: 1 B: 00000001
125000.00ns INFO     cocotb.Main                        BTB_PC1: D: 4 H: 4 B: 00000000000000000000000000000100
125000.00ns INFO     cocotb.Main                        BTB_PC2: D: 36 H: 24 B: 00000000000000000000000000100100
125000.00ns INFO     cocotb.Main                        BTB_BTA1: D: 24 H: 18 B: 00000000000000000000000000011000
125000.00ns INFO     cocotb.Main                        BTB_BTA2: D: 4 H: 4 B: 00000000000000000000000000000100
125000.00ns INFO     cocotb.Main                        OUT: D: 4 H: 4 B: 00000000000000000000000000000100
-----------------
125000.00ns INFO     cocotb.Main                        12 - NOP
135000.00ns INFO     cocotb.Main                        InstructionF: H: f0000000
135000.00ns INFO     cocotb.Main                        INSTR: H: f0000000
135000.00ns INFO     cocotb.Main                        PCPrime: D: 24 H: 18 B: 00000000000000000000000000011000
135000.00ns INFO     cocotb.Main                        PCF: D: 12 H: c B: 00000000000000000000000000001100
135000.00ns INFO     cocotb.Main                        PCD: D: 8 H: 8 B: 00000000000000000000000000001000
135000.00ns INFO     cocotb.Main                        PCE: D: 4 H: 4 B: 00000000000000000000000000000100
135000.00ns INFO     cocotb.Main                        BranchE: D: 1 H: 1 B: 1
135000.00ns INFO     cocotb.Main                        ALUControlE: D: 4 H: 4 B: 0100
135000.00ns INFO     cocotb.Main                        ALUSrcE: D: 1 H: 1 B: 1
135000.00ns INFO     cocotb.Main                        ImmSrcD: D: 2 H: 2 B: 10
135000.00ns INFO     cocotb.Main                        CondE: D: 14 H: e B: 1110
135000.00ns INFO     cocotb.Main                        RA1E: D: 15 H: f B: 1111
135000.00ns INFO     cocotb.Main                        RA2E: D: 3 H: 3 B: 0011
135000.00ns INFO     cocotb.Main                        FlushD: D: 1 H: 1 B: 1
135000.00ns INFO     cocotb.Main                        FlushE: D: 1 H: 1 B: 1
135000.00ns INFO     cocotb.Main                        BranchTakenE: D: 1 H: 1 B: 1
135000.00ns INFO     cocotb.Main                        SrcAE: D: 12 H: c B: 00000000000000000000000000001100
135000.00ns INFO     cocotb.Main                        SrcBE: D: 12 H: c B: 00000000000000000000000000001100
135000.00ns INFO     cocotb.Main                        ReadDataM: D: 2 H: 2 B: 00000000000000000000000000000010
135000.00ns INFO     cocotb.Main                        ReadDataW: D: 2 H: 2 B: 00000000000000000000000000000010
135000.00ns INFO     cocotb.Main                        ALUResultE: D: 24 H: 18 B: 00000000000000000000000000011000
135000.00ns INFO     cocotb.Main                        LRU: D: 2 H: 2 B: 10
135000.00ns INFO     cocotb.Main                        GHROut: D: 4 H: 4 B: 100
135000.00ns INFO     cocotb.Main                        PHT: D: 1 H: 1 B: 00000001
135000.00ns INFO     cocotb.Main                        BTB_PC1: D: 4 H: 4 B: 00000000000000000000000000000100
135000.00ns INFO     cocotb.Main                        BTB_PC2: D: 36 H: 24 B: 00000000000000000000000000100100
135000.00ns INFO     cocotb.Main                        BTB_BTA1: D: 24 H: 18 B: 00000000000000000000000000011000
135000.00ns INFO     cocotb.Main                        BTB_BTA2: D: 4 H: 4 B: 00000000000000000000000000000100
135000.00ns INFO     cocotb.Main                        OUT: D: 0 H: 0 B: 00000000000000000000000000000000
-----------------
136000.00ns INFO     cocotb.regression                  branch_predict_1 ←[32mpassed←[49m←[39m
136000.00ns INFO     cocotb.regression                  ******************************************************************************************
                                                        ** TEST                                STATUS  SIM TIME (ns)  REAL TIME (s)  RATIO (ns/s) **
                                                        ******************************************************************************************
                                                        ** branch_predictor.branch_predict_1  ←[32m PASS ←[49m←[39m     136000.00           0.34     400043.58  **
                                                        ******************************************************************************************
                                                        ** TESTS=1 PASS=1 FAIL=0 SKIP=0                     136000.00           0.66     207008.49  **
                                                        ******************************************************************************************

make[1]: Leaving directory '/c/Users/Deniz/Documents/Pipelined-Project/project-tests/branch_predictor'
(base) PS C:\Users\Deniz\Documents\Pipelined-Project\project-tests\branch_predictor> _
```

Figure 18. COCOTB Test procedure for the Branch Predictor No.5.