



Computer Architecture

Dr. Dheya Ghazi

Assignment3: CPU

Team #5

Name	ID Number	Seat Number
Moath Yousef AL-Tahrawi	2141099	4
Omar Tareq AL-Shamasneh	2139919	69
Mohammed AL-Tamimi	2042979	58
Abdullah Nasha'at	1937045	36
Ahmed Maen Barham	2330132	1

Introduction

This project involves the design, construction, and testing of a basic Central Processing Unit (CPU) featuring a hardwired control unit integrated into the datapath from Assignment 2. The goal is to build a functional CPU capable of executing a defined set of benchmark machine instructions, with a particular focus on Register-Reference and Input-Output operations.

Objectives

- 1. Control Unit Design:** Create a hardwired control unit that ensures efficient instruction decoding and execution, seamlessly coordinating with the datapath.
- 2. Instruction Set Implementation:** Implement essential operations, including Clear, Complement, Shift, and Increment, alongside Input-Output instructions.
- 3. Testing and Validation:** Develop machine code benchmarks to thoroughly test the CPU's functionality and validate its performance.

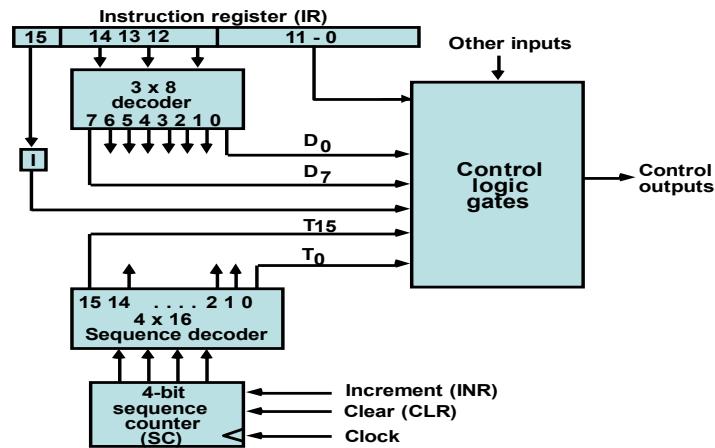
Methodology

- 1. Control Unit Design:** The control logic was developed using state transitions and micro-operations based on the instruction table. Control signals were derived from instruction decoding and timing sequences to ensure proper functionality.
- 2. Integration with Datapath:** The control unit was integrated into the datapath designed in Assignment 2, enabling the execution of various arithmetic, logical, and control flow instructions.
- 3. Benchmark Testing:** Machine code instructions were written to serve as benchmarks for validating CPU performance. These benchmarks tested both the control unit's logic and its integration with the datapath.

Results

The CPU successfully executed all specified Register-Reference and Input-Output instructions, meeting design specifications. Key operations such as Complement (CMA), Increment (INC), Input (INP), and Output (OUT) were verified for accuracy. Benchmark tests demonstrated the CPU's ability to handle these operations efficiently, confirming the seamless integration of the control unit with the datapath.

The Control Unit for the basic computer



Hardwired Control Organization

cpe 252: Computer Organization

25

The Control Unit for the Basic Computer

This diagram illustrates the architecture of a hardwired control unit for a basic computer. The control unit is a crucial component responsible for orchestrating the execution of instructions by generating and sequencing appropriate control signals.

Components and Functionality:

- 1. Instruction Register (IR):** Holds the current instruction being executed. The instruction's opcode, which determines the operation to be performed, is located in bits 15-12 of the IR.
- 2. 3x8 Decoder:** Decodes the opcode field (bits 15-12) into one of eight possible control signals (D₀-D₇). Each control signal corresponds to a specific operation or micro-operation within the instruction cycle.
- 3. Control Logic Gates:** These gates implement the logic functions that determine which control signals should be activated based on the decoded opcode and other inputs, such as flags and status signals.
- 4. Sequence Decoder:** This component generates a 4-bit sequence counter (SC) signal. The SC signal is used to control the timing and order of control signals during the instruction cycle.

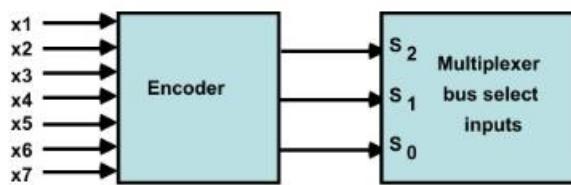
5. **Sequence Counter (SC):** This 4-bit counter increments on each clock cycle, generating a unique sequence for each instruction cycle. This allows for precise timing of control signals.
6. **Clear (CLR) and Increment (INR) Signals:** These signals control the operation of the sequence counter. CLR resets the counter to its initial state, while INR increments it by one.
7. **Clock Signal:** Provides the timing reference for the entire system, synchronizing the operation of all components.

Operation:

1. **Instruction Fetch:** The instruction is fetched from memory and loaded into the IR.
2. **Opcode Decoding:** The 3x8 decoder decodes the opcode field of the instruction into one of eight possible control signals.
3. **Control Signal Generation:** The control logic gates generate the appropriate control signals based on the decoded opcode and other inputs.
4. **Sequence Counter Control:** The sequence counter (SC) is incremented on each clock cycle, generating a unique sequence for each instruction cycle. The clear (CLR) signal can be used to reset the counter.
5. **Control Signal Timing:** The SC signal is used to time the generation of control signals, ensuring they are activated in the correct order and duration.

5-9 Design of Basic Computer^{cont.}

- Control of Common bus is accomplished by placing an encoder at the inputs of the bus selection logic and implementing the logic for each encoder input



In the given computer architecture, the encoder acts as a bus selector. It receives multiple input signals, each representing a different device or module. By processing these inputs, the encoder generates a set of output signals that control a multiplexer. This allows the multiplexer to select the desired device to connect to the common bus, ensuring efficient data transfer and preventing conflicts.

Essentially, the encoder translates device requests into signals that guide the bus traffic

Fetch	R'T0:	$AR \leftarrow PC$
	R'T1:	$IR \leftarrow M[AR], PC \leftarrow PC + 1$
Decode	R'T2:	$D_0, \dots, D_7 \leftarrow \text{Decode } IR(12 \sim 14), AR \leftarrow IR(0 \sim 11), I \leftarrow IR(15)$
Indirect	D7'IT3:	$AR \leftarrow M[AR]$
Interrupt:		
T0'T1'T2'(IEN)(FGI + FGO):		$R \leftarrow 1$
	RT0:	$AR \leftarrow 0, TR \leftarrow PC$
	RT1:	$M[AR] \leftarrow TR, PC \leftarrow 0$
	RT2:	$PC \leftarrow PC + 1, IEN \leftarrow 0, R \leftarrow 0, SC \leftarrow 0$
Memory-Reference:		
AND	D0T4:	$DR \leftarrow M[AR]$
	D0T5:	$AC \leftarrow AC . DR, SC \leftarrow 0$
ADD	D1T4:	$DR \leftarrow M[AR]$
	D1T5:	$AC \leftarrow AC + DR, E \leftarrow \underline{\text{Cout}}, SC \leftarrow 0$
LDA	D2T4:	$DR \leftarrow M[AR]$
	D2T5:	$AC \leftarrow DR, SC \leftarrow 0$
STA	D3T4:	$M[AR] \leftarrow AC, SC \leftarrow 0$
BUN	D4T4:	$PC \leftarrow AR, SC \leftarrow 0$
BSA	D5T4:	$M[AR] \leftarrow PC, AR \leftarrow AR + 1$
	D5T5:	$PC \leftarrow AR, SC \leftarrow 0$
ISZ	D6T4:	$DR \leftarrow M[AR]$
	D6T5:	$DR \leftarrow DR + 1$
	D6T6:	$M[AR] \leftarrow DR, \text{ if}(DR=0) \text{ then } (PC \leftarrow PC + 1), SC \leftarrow 0$

1. AND (D0T4 - D0T5)

D0T4: Load the Data Register (DR) with the value from memory at the address in AR.

D0T5: Perform a bitwise AND between AC and DR, storing the result in AC. Set SC to 0 to finish.

2. ADD (D1T4 - D1T5)

D1T4: Load DR from memory at the address in AR.

D1T5: Add DR to AC, storing the result in AC. Store the carry-out in E. Set SC to 0.

3. LDA (D2T4 - D2T5)

D2T4: Load DR from memory at AR.

D2T5: Transfer DR to AC. Set SC to 0.

4. STA (D3T4)

D3T4: Store AC into memory at the address in AR. Set SC to 0.

5. BUN (D4T4)

D4T4: Set PC to the value in AR, causing an unconditional branch. Set SC to 0.

Register-Reference:		
	D7IT3 = r	(Common to all register-reference instructions)
	IR(i) = Bi	(i = 0,1,2, ..., 11)
	r:	SC ← 0
CLA	rB11:	AC ← 0
CLE	rB10:	E ← 0
CMA	rB9:	AC ← AC'
CME	rB8:	E ← E'
CIR	rB7:	AC ← shr AC, AC(15) ← E, E ← AC(0)
CIL	rB6:	AC ← shl AC, AC(0) ← E, E ← AC(15)
INC	rB5:	AC ← AC + 1
SPA	rB4:	If(AC(15) = 0) then (PC ← PC + 1)
SNA	rB3:	If(AC(15) = 1) then (PC ← PC + 1)
SZA	rB2:	If(AC = 0) then (PC ← PC + 1)
SZE	rB1:	If(E=0) then (PC ← PC + 1)
HLT	rB0:	S ← 0
Input-Output:		
	D7IT3 = p	(Common to all input-output instructions)
	IR(i) = Bi	(i = 6,7,8,9,10,11)
	p:	SC ← 0
INP	pB11:	AC(0-7) ← INPR, FGI ← 0
OUT	pB10:	OUTR ← AC(0-7), FGO ← 0
SKI	pB9:	If(FGI=1) then (PC ← PC + 1)
SKO	pB8:	If(FGO=1) then (PC ← PC + 1)
ION	pB7:	IEN ← 1
IOF	pB6:	IEN ← 0

©PC 2022, Computer Organization

Register-Reference Instructions

These instructions operate directly on the Accumulator (AC), the E flag, and the Program Counter (PC). The common control for all these instructions is D7IT3 = r where r represents the specific operation.

CLA (rB11): Clears the AC. AC ← 0

CLE (rB10): Clears the E flag. E ← 0

CMA (rB9): Complements the AC. AC ← AC'

CME (rB8): Complements the E flag. E ← E'

CIR (rB7): Circular shift right on AC, with E shifted into AC(15). AC ← shr AC, AC(15) ← E, E ← ZC(0)

CIL (rB6): Circular shift left on AC, with E shifted into AC(0). AC ← shl AC, AC(0) ← E, E ← ZC(15)

INC (rB5): Increments AC. AC ← AC + 1

SPA (rB4): Skips next instruction if AC(15) = 0 (AC positive). PC ← PC + 1

SNA (rB3): Skips next instruction if AC(15) = 1 (AC negative). PC ← PC + 1

SZA (rB2): Skips next instruction if AC = 0. PC ← PC + 1

SZE (rB1): Skips next instruction if E = 0. PC ← PC + 1

Input-Output Instructions

These instructions manage data transfer between the system and external devices, and control interrupt behavior.

INP (pB11): Inputs data from INPR into the AC(0-7). Clears FGI. AC(0-7) ← INPR, FGI ← 0

OUT (pB10): Outputs AC(0-7) to OUTRAC. Clears FGO. OUTRAC ← AC(0-7), FGO ← 0

SKI (pB9): Skips next instruction if FGI = 1. PC ← PC + 1

SKO (pB8): Skips next instruction if FGO = 1. PC ← PC + 1

ION (pB7): Enables interrupts. IEN ← 1

IOF (pB6): Disables interrupts. IEN ← 0



```

0, AR: 000, PC: 012c, DR: 0000, AC: 0000, IR: 0000, TR: 0000, memo location:4132, E:&Hx
10, AR: 12c, PC: 012c, DR: 0000, AC: 0000, IR: 0000, memo location:4132, E:&Hx
20, AR: 12c, PC: 012d, DR: 0000, AC: 0000, IR: 11f4, TR: 0000, memo location:4132, E:&Hx
30, AR: 1f4, PC: 012d, DR: 0000, AC: 0000, IR: 11f4, TR: 0000, memo location:4132, E:&Hx
50, AR: 1f4, PC: 012d, DR: 1010, AC: 0000, IR: 11f4, TR: 0000, memo location:4132, E:&Hx
60, AR: 1f4, PC: 012d, DR: 1010, AC: 1010, IR: 11f4, TR: 0000, memo location:4132, E:&H0
70, AR: 12d, PC: 012d, DR: 1010, AC: 1010, IR: 11f4, TR: 0000, memo location:4132, E:&H0
80, AR: 12d, PC: 012e, DR: 1010, AC: 1010, IR: 01f5, TR: 0000, memo location:4132, E:&H0
90, AR: 1f5, PC: 012e, DR: 1010, AC: 1010, IR: 01f5, TR: 0000, memo location:4132, E:&H0
110, AR: 1f5, PC: 012e, DR: 0101, AC: 1010, IR: 01f5, TR: 0000, memo location:4132, E:&H0
120, AR: 1f5, PC: 012e, DR: 0101, AC: 0000, IR: 01f5, TR: 0000, memo location:4132, E:&H0
130, AR: 12e, PC: 012e, DR: 0101, AC: 0000, IR: 01f5, TR: 0000, memo location:4132, E:&H0
140, AR: 12e, PC: 012f, DR: 0101, AC: 0000, IR: 21f6, TR: 0000, memo location:4132, E:&H0
150, AR: 1f6, PC: 012f, DR: 0101, AC: 0000, IR: 21f6, TR: 0000, memo location:4132, E:&H0
170, AR: 1f6, PC: 012f, DR: 1234, AC: 0000, IR: 21f6, TR: 0000, memo location:4132, E:&H0
180, AR: 1f6, PC: 012f, DR: 1234, AC: 1234, IR: 21f6, TR: 0000, memo location:4132, E:&H0
190, AR: 12f, PC: 012f, DR: 1234, AC: 1234, IR: 21f6, TR: 0000, memo location:4132, E:&H0

```

step-by-step progression of machine states, capturing the values of various registers at each time point:

- **At time 0:** Initial state.
 - AR = 000, PC = 012c, DR = 0000, AC = 0000, IR = 0000, TR = 0000, memo location = 4132, E = &Hx.
- **At time 10:**
 - The AR remains 12c, and PC stays at 012c. The DR, AC, IR, and TR stay at their initial values.
- **At time 20:**
 - AR stays at 12c, but PC moves to 012d. The IR is updated to 11f4.
- **At time 30:**
 - AR shifts to 1f4, and PC remains at 012d. DR stays at 0000.
- **At time 50:**
 - AR is still 1f4, but now DR becomes 1010.

- **At time 60:**
 - AR stays at 1f4. DR remains 1010, and AC is updated to 1010. IR remains 11f4.
- **At time 70:**
 - AR moves to 12d, while PC, DR, and AC retain their values.
- **At time 80:**
 - PC increments to 012e. DR and AC stay at 1010, while IR updates to 01f5.
- **At time 90:**
 - AR becomes 1f5, and PC remains at 012e.
- **At time 110:**
 - DR is updated to 0101.
- **At time 120:**
 - The value in AC resets to 0000.
- **At time 130:**
 - AR remains at 12e. DR is still 0101, and IR continues at 01f5.
- **At time 140:**
 - PC moves to 012f, and the IR changes to 21f6.
- **At time 150:**
 - AR changes to 1f6, but the DR and AC remain at 0101 and 0000, respectively.
- **At time 170:**
 - DR becomes 1234, with the AC still 0000.

This progression illustrates how the values in various registers, such as the Address Register (AR), Program Counter (PC), Data Register (DR), Accumulator (AC), Instruction Register (IR), and others, evolve over time during the execution of the program. These changes reflect the machine's ongoing operations and updates in its memory and computations.

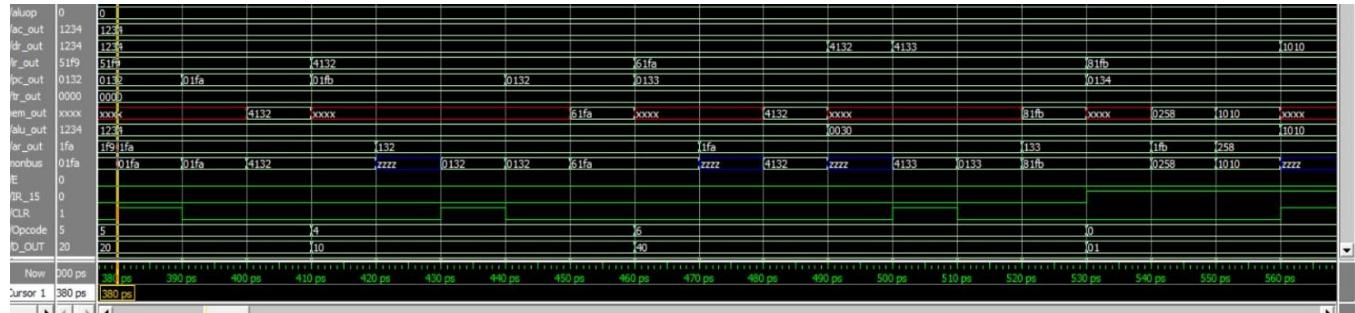


```

190, AR: 12f, PC: 012f, DR: 1234, AC: 1234, IR: 21f6, TR: 0000, memo location:4132, E:&H0
200, AR: 12f, PC: 0130, DR: 1234, AC: 1234, IR: 31f7, TR: 0000, memo location:4132, E:&H0
210, AR: 1f7, PC: 0130, DR: 1234, AC: 1234, IR: 31f7, TR: 0000, memo location:4132, E:&H0
240, AR: 130, PC: 0130, DR: 1234, AC: 1234, IR: 31f7, TR: 0000, memo location:4132, E:&H0
250, AR: 130, PC: 0131, DR: 1234, AC: 1234, IR: 4300, TR: 0000, memo location:4132, E:&H0
260, AR: 300, PC: 0131, DR: 1234, AC: 1234, IR: 4300, TR: 0000, memo location:4132, E:&H0
280, AR: 300, PC: 0300, DR: 1234, AC: 1234, IR: 4300, TR: 0000, memo location:4132, E:&H0
300, AR: 300, PC: 0301, DR: 1234, AC: 1234, IR: 4131, TR: 0000, memo location:4132, E:&H0
310, AR: 131, PC: 0301, DR: 1234, AC: 4131, TR: 0000, memo location:4132, E:&H0
330, AR: 131, PC: 0131, DR: 1234, AC: 4131, TR: 0000, memo location:4132, E:&H0
350, AR: 131, PC: 0132, DR: 1234, AC: 51f9, TR: 0000, memo location:4132, E:&H0
360, AR: 1f9, PC: 0132, DR: 1234, AC: 51f9, TR: 0000, memo location:4132, E:&H0
380, AR: 1fa, PC: 0132, DR: 1234, AC: 51f9, TR: 0000, memo location:4132, E:&H0

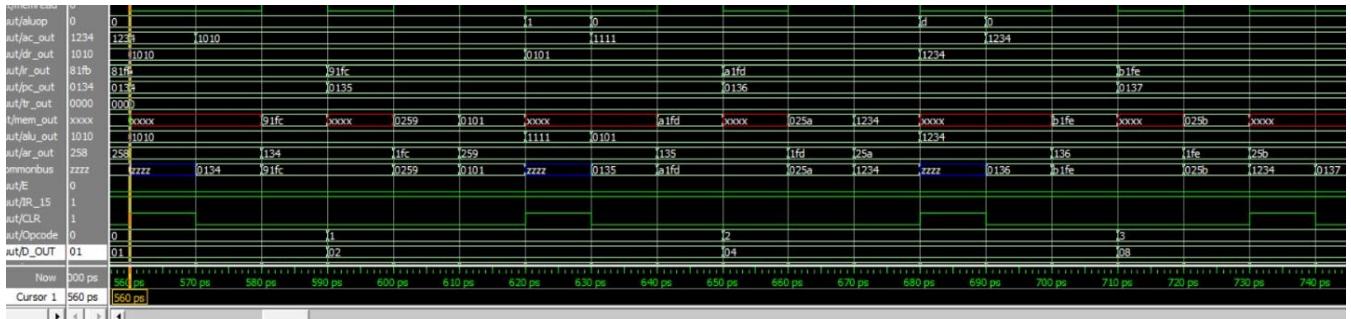
```

- At time 200: AR = 12f, PC = 0130, DR = 1234, AC = 1234, IR = 31f7.
- At time 210: AR = 1f7, PC = 0130, DR = 1234, AC = 1234, IR = 31f7.
- At time 240: AR = 130, PC = 0130, DR = 1234, AC = 1234, IR = 31f7.
- At time 250: AR = 130, PC = 0131, DR = 1234, AC = 1234, IR = 4300.
- At time 260: AR = 300, PC = 0131, DR = 1234, AC = 1234, IR = 4300.
- At time 280: AR = 300, PC = 0300, DR = 1234, AC = 1234, IR = 4300.
- At time 300: AR = 300, PC = 0301, DR = 1234, AC = 1234, IR = 4131.
- At time 310: AR = 131, PC = 0301, DR = 1234, AC = 1234, IR = 4131.



380, AR: 1fa, PC: 0132, DR: 1234, AC: 1234, IR: 51f9, TR: 0000, memo location:4132, E:&H0
 390, AR: 1fa, PC: 01fa, DR: 1234, AC: 1234, IR: 51f9, TR: 0000, memo location:4132, E:&H0
 410, AR: 1fa, PC: 01fb, DR: 1234, AC: 1234, IR: 4132, TR: 0000, memo location:4132, E:&H0
 420, AR: 132, PC: 01fb, DR: 1234, AC: 1234, IR: 4132, TR: 0000, memo location:4132, E:&H0
 440, AR: 132, PC: 0132, DR: 1234, AC: 1234, IR: 4132, TR: 0000, memo location:4132, E:&H0
 460, AR: 132, PC: 0133, DR: 1234, AC: 1234, IR: 61fa, TR: 0000, memo location:4132, E:&H0
 470, AR: 1fa, PC: 0133, DR: 1234, AC: 1234, IR: 61fa, TR: 0000, memo location:4132, E:&H0
 490, AR: 1fa, PC: 0133, DR: 4132, AC: 1234, IR: 61fa, TR: 0000, memo location:4132, E:&H0
 500, AR: 1fa, PC: 0133, DR: 4133, AC: 1234, IR: 61fa, TR: 0000, memo location:4132, E:&H0
 510, AR: 1fa, PC: 0133, DR: 4133, AC: 1234, IR: 61fa, TR: 0000, memo location:4133, E:&H0
 520, AR: 133, PC: 0133, DR: 4133, AC: 1234, IR: 61fa, TR: 0000, memo location:4133, E:&H0
 530, AR: 133, PC: 0134, DR: 4133, AC: 1234, IR: 81fb, TR: 0000, memo location:4133, E:&H0
 540, AR: 1fb, PC: 0134, DR: 4133, AC: 1234, IR: 81fb, TR: 0000, memo location:4133, E:&H0
 550, AR: 258, PC: 0134, DR: 4133, AC: 1234, IR: 81fb, TR: 0000, memo location:4133, E:&H0
 560, AR: 258, PC: 0134, DR: 1010, AC: 1234, IR: 81fb, TR: 0000, memo location:4133, E:&H0

- At time 390: AR = 1fa, PC = 01fa, DR = 1234, AC = 1234, IR = 51f9.
- At time 410: AR = 1fa, PC = 01fb, DR = 1234, AC = 1234, IR = 4132.
- At time 420: AR = 132, PC = 01fb, DR = 1234, AC = 1234, IR = 4132.
- At time 440: AR = 132, PC = 0132, DR = 1234, AC = 1234, IR = 4132.
- At time 460: AR = 132, PC = 0133, DR = 1234, AC = 1234, IR = 61fa.
- At time 470: AR = 1fa, PC = 0133, DR = 1234, AC = 1234, IR = 61fa.
- At time 490: AR = 1fa, PC = 0133, DR = 4132, AC = 1234, IR = 61fa.
- At time 500: AR = 1fa, PC = 0133, DR = 4133, AC = 1234, IR = 61fa.
- At time 510: AR = 1fa, PC = 0133, DR = 4133, AC = 1234, IR = 61fa.

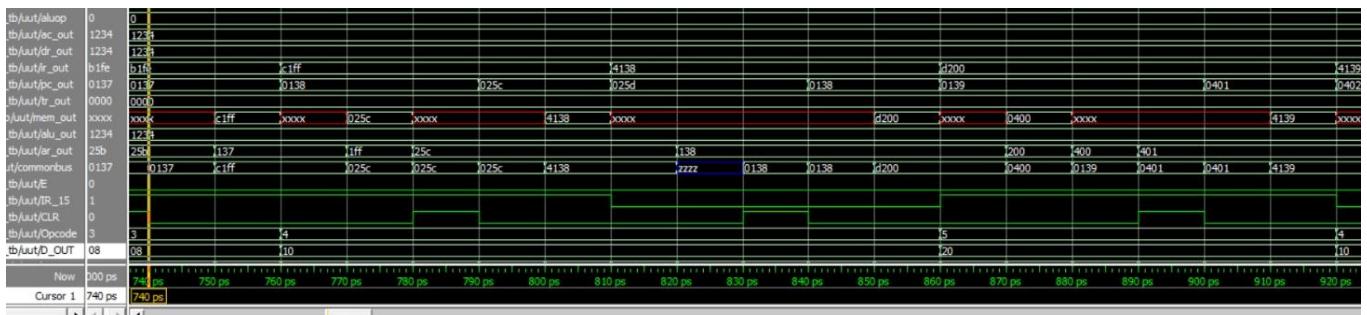


```

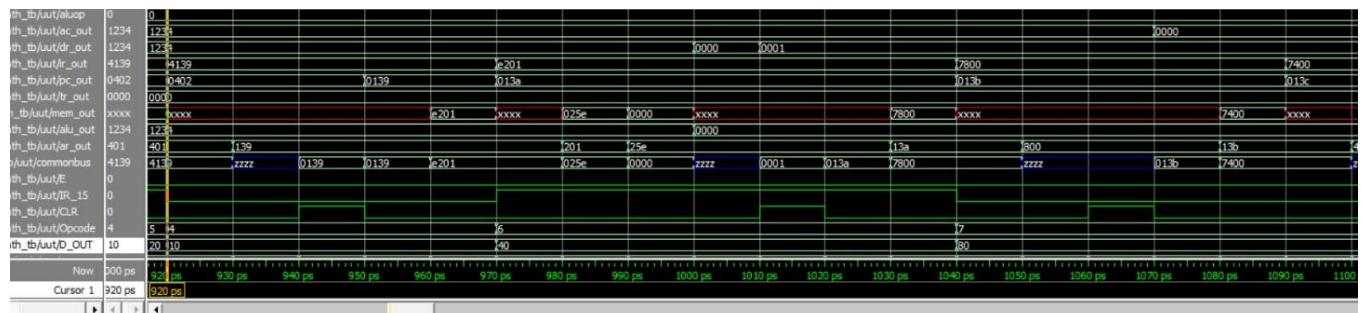
560, AR: 258, PC: 0134, DR: 1010, AC: 1234, IR: 81fb, TR: 0000, memo location:4133, E:&H0
570, AR: 258, PC: 0134, DR: 1010, AC: 1010, IR: 81fb, TR: 0000, memo location:4133, E:&H0
580, AR: 134, PC: 0134, DR: 1010, AC: 1010, IR: 81fb, TR: 0000, memo location:4133, E:&H0
590, AR: 134, PC: 0135, DR: 1010, AC: 1010, IR: 91fc, TR: 0000, memo location:4133, E:&H0
600, AR: 1fc, PC: 0135, DR: 1010, AC: 1010, IR: 91fc, TR: 0000, memo location:4133, E:&H0
610, AR: 259, PC: 0135, DR: 1010, AC: 1010, IR: 91fc, TR: 0000, memo location:4133, E:&H0
620, AR: 259, PC: 0135, DR: 0101, AC: 1010, IR: 91fc, TR: 0000, memo location:4133, E:&H0
630, AR: 259, PC: 0135, DR: 0101, AC: 1111, IR: 91fc, TR: 0000, memo location:4133, E:&H0
640, AR: 135, PC: 0135, DR: 0101, AC: 1111, IR: 91fc, TR: 0000, memo location:4133, E:&H0
650, AR: 135, PC: 0136, DR: 0101, AC: 1111, IR: alfd, TR: 0000, memo location:4133, E:&H0
660, AR: 1fd, PC: 0136, DR: 0101, AC: 1111, IR: alfd, TR: 0000, memo location:4133, E:&H0
670, AR: 25a, PC: 0136, DR: 0101, AC: 1111, IR: alfd, TR: 0000, memo location:4133, E:&H0
680, AR: 25a, PC: 0136, DR: 1234, AC: 1111, IR: alfd, TR: 0000, memo location:4133, E:&H0
690, AR: 25a, PC: 0136, DR: 1234, AC: 1234, IR: alfd, TR: 0000, memo location:4133, E:&H0
700, AR: 136, PC: 0136, DR: 1234, AC: 1234, IR: alfd, TR: 0000, memo location:4133, E:&H0
710, AR: 136, PC: 0137, DR: 1234, AC: blfe, TR: 0000, memo location:4133, E:&H0
720, AR: 1fe, PC: 0137, DR: 1234, AC: blfe, TR: 0000, memo location:4133, E:&H0
730, AR: 25b, PC: 0137, DR: 1234, AC: blfe, TR: 0000, memo location:4133, E:&H0
750, AR: 137, PC: 0137, DR: 1234, AC: 1234, IR: blfe, TR: 0000, memo location:4133, E:&H0

```

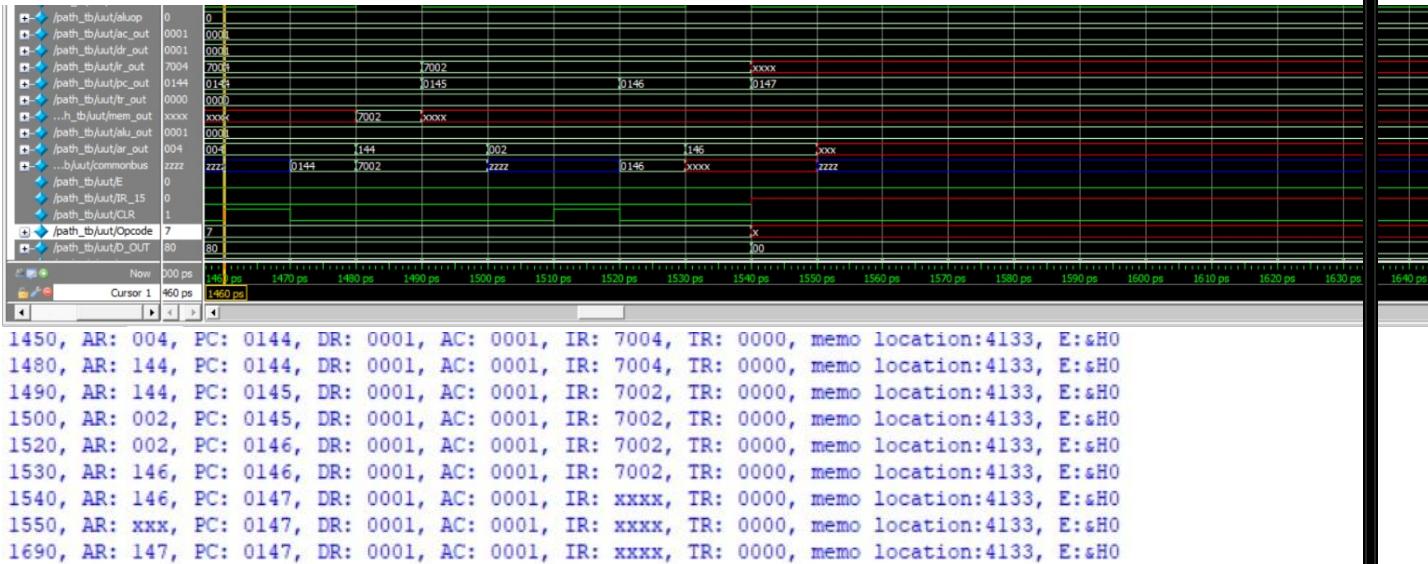
- **At time 560:** The **AR** is 258, and the **PC** is 0134. The **DR** is 1010, and the **AC** is 1234. The **IR** holds 81fb, and the **TR** is 0000. The memory at 4133 is being accessed.
- **At time 570:** The **AR** remains 258, and the **PC** is still 0134. The **DR** is 1010, and the **AC** has been updated to 1010. The **IR** is 81fb, and the **TR** is 0000.
- **At time 580:** The **AR** has been updated to 134. The **PC** is still at 0134, and the **DR** and **AC** remain at 1010.
- **At time 590:** The **PC** is incremented to 0135, and the **IR** contains 91fc. The **DR** and **AC** remain at 1010.
- **At time 600:** The **AR** is updated to 1fc, and the **PC** remains at 0135. The **DR** and **AC** still hold 1010.
- **At time 610:** The **AR** is now 259, and the **PC** remains at 0135. The **DR** is 0101, and the **AC** is 1010.
- **At time 620:** The **DR** has been updated to 0101, and the **AC** is now 1111. The **PC** is still at 0135.
- **At time 630:** The **AR** remains at 259, and the **PC** is still at 0135. The **DR** holds 0101, and the **AC** is now 1111.
- **At time 640:** The **AR** is updated to 135, and the **PC** remains at 0135. The **DR** is 0101, and the **AC** is 1111.
- **At time 650:** The **PC** is incremented to 0136, and the **IR** contains alfd. The **AC** is 1111, and the **DR** is 0101.
- **At time 660:** The **AR** is 1fd, and the **PC** is 0136. The **AC** and **DR** remain at 1111 and 0101, respectively.
- **At time 670:** The **AR** is updated to 25a, and the **PC** is still at 0136. The **DR** remains at 0101, and the **AC** is 1111.



750, AR: 137, PC: 0137, DR: 1234, AC: 1234, IR: blfe, TR: 0000, memo location:4133, E:@HO
760, AR: 137, PC: 0138, DR: 1234, AC: 1234, IR: clff, TR: 0000, memo location:4133, E:@HO
770, AR: 1ff, PC: 0138, DR: 1234, AC: 1234, IR: clff, TR: 0000, memo location:4133, E:@HO
780, AR: 25c, PC: 0138, DR: 1234, AC: 1234, IR: clff, TR: 0000, memo location:4133, E:@HO
790, AR: 25c, PC: 025c, DR: 1234, AC: 1234, IR: clff, TR: 0000, memo location:4133, E:@HO
810, AR: 25c, PC: 025d, DR: 1234, AC: 1234, IR: 4138, TR: 0000, memo location:4133, E:@HO
820, AR: 138, PC: 025d, DR: 1234, AC: 1234, IR: 4138, TR: 0000, memo location:4133, E:@HO
840, AR: 138, PC: 0138, DR: 1234, AC: 1234, IR: 4138, TR: 0000, memo location:4133, E:@HO
860, AR: 138, PC: 0139, DR: 1234, AC: 1234, IR: d200, TR: 0000, memo location:4133, E:@HO
870, AR: 200, PC: 0139, DR: 1234, AC: 1234, IR: d200, TR: 0000, memo location:4133, E:@HO
880, AR: 400, PC: 0139, DR: 1234, AC: 1234, IR: d200, TR: 0000, memo location:4133, E:@HO
890, AR: 401, PC: 0139, DR: 1234, AC: 1234, IR: d200, TR: 0000, memo location:4133, E:@HO
900, AR: 401, PC: 0401, DR: 1234, AC: 1234, IR: d200, TR: 0000, memo location:4133, E:@HO
920, AR: 401, PC: 0402, DR: 1234, AC: 1234, IR: 4139, TR: 0000, memo location:4133, E:@HO

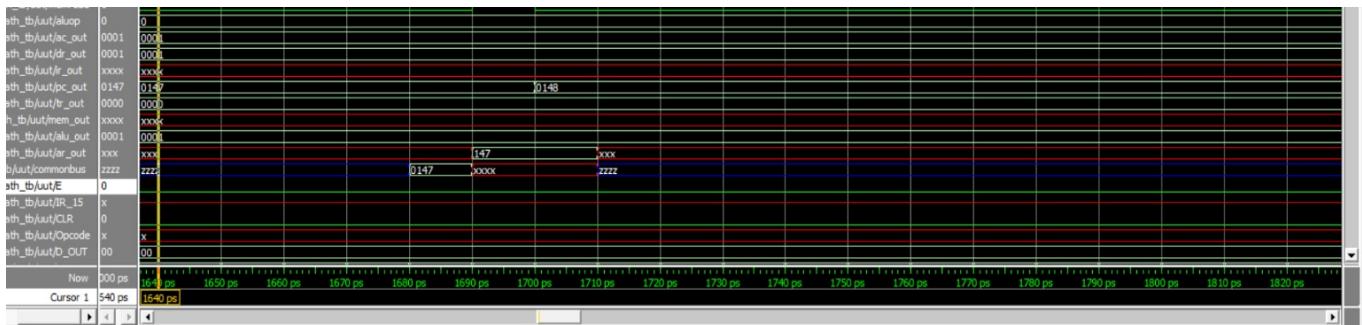


920, AR: 401, PC: 0402, DR: 1234, AC: 1234, IR: 4139, TR: 0000, memo location:4133, E:&HO
930, AR: 139, PC: 0402, DR: 1234, AC: 1234, IR: 4139, TR: 0000, memo location:4133, E:&HO
950, AR: 139, PC: 0139, DR: 1234, AC: 1234, IR: 4139, TR: 0000, memo location:4133, E:&HO
970, AR: 139, PC: 013a, DR: 1234, AC: 1234, IR: e201, TR: 0000, memo location:4133, E:&HO
980, AR: 201, PC: 013a, DR: 1234, AC: 1234, IR: e201, TR: 0000, memo location:4133, E:&HO
990, AR: 25e, PC: 013a, DR: 1234, AC: 1234, IR: e201, TR: 0000, memo location:4133, E:&HO
1000, AR: 25e, PC: 013a, DR: 0000, AC: 1234, IR: e201, TR: 0000, memo location:4133, E:&HO
1010, AR: 25e, PC: 013a, DR: 0001, AC: 1234, IR: e201, TR: 0000, memo location:4133, E:&HO
1030, AR: 13a, PC: 013a, DR: 0001, AC: 1234, IR: e201, TR: 0000, memo location:4133, E:&HO
1040, AR: 13a, PC: 013b, DR: 0001, AC: 1234, IR: 7800, TR: 0000, memo location:4133, E:&HO
1050, AR: 800, PC: 013b, DR: 0001, AC: 1234, IR: 7800, TR: 0000, memo location:4133, E:&HO
1070, AR: 800, PC: 013b, DR: 0001, AC: 0000, IR: 7800, TR: 0000, memo location:4133, E:&HO
1080, AR: 13b, PC: 013b, DR: 0001, AC: 0000, IR: 7800, TR: 0000, memo location:4133, E:&HO
1090, AR: 13b, PC: 013c, DR: 0001, AC: 0000, IR: 7400, TR: 0000, memo location:4133, E:&HO
1100, AR: 400, PC: 013c, DR: 0001, AC: 0000, IR: 7400, TR: 0000, memo location:4133, E:&HO



- At time 1480:
 - AR: 144, PC: 0144, DR: 0001, AC: 0001, IR: 7004, TR: 0000, Memo Location: 4133, E: 0
 - The BUN instruction is being executed. The value of AR (which is 144) is transferred to the PC, causing the program to jump to address 144.
- At time 1490:
 - AR: 144, PC: 0145, DR: 0001, AC: 0001, IR: 7002, TR: 0000, Memo Location: 4133, E: 0
 - The PC has been updated to 0145. This is the result of the jump, and the next instruction (7002) is being processed.
- At time 1500:
 - AR: 002, PC: 0145, DR: 0001, AC: 0001, IR: 7002, TR: 0000, Memo Location: 4133, E: 0
 - The AR now contains 002, which will potentially be the target for the next instruction. The PC remains at 0145.
- At time 1520:
 - AR: 002, PC: 0146, DR: 0001, AC: 0001, IR: 7002, TR: 0000, Memo Location: 4133, E: 0
 - The PC has advanced to 0146. The next instruction is processed, and AR continues to hold 002.
- At time 1530:
 - AR: 146, PC: 0146, DR: 0001, AC: 0001, IR: 7002, TR: 0000, Memo Location: 4133, E: 0
 - AR is updated to 146, but the PC remains at 0146. The process continues with instruction 7002.
- At time 1540:
 - AR: 146, PC: 0147, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, Memo Location: 4133, E: 0

- The PC moves to 0147 as per the control flow. The instruction is unclear (xxxx), but the program continues.
- At time 1550:
 - AR: xxx, PC: 0147, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, Memo Location: 4133, E: 0
 - The AR value is unknown (xxx), and the instruction (xxxx) is also unclear. The PC remains at 0147.
- At time 1690:
 - AR: 147, PC: 0147, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, Memo Location: 4133, E: 0
 - AR is updated to 147, and the PC remains at 0147. The program continues, but further instructions are not clear.



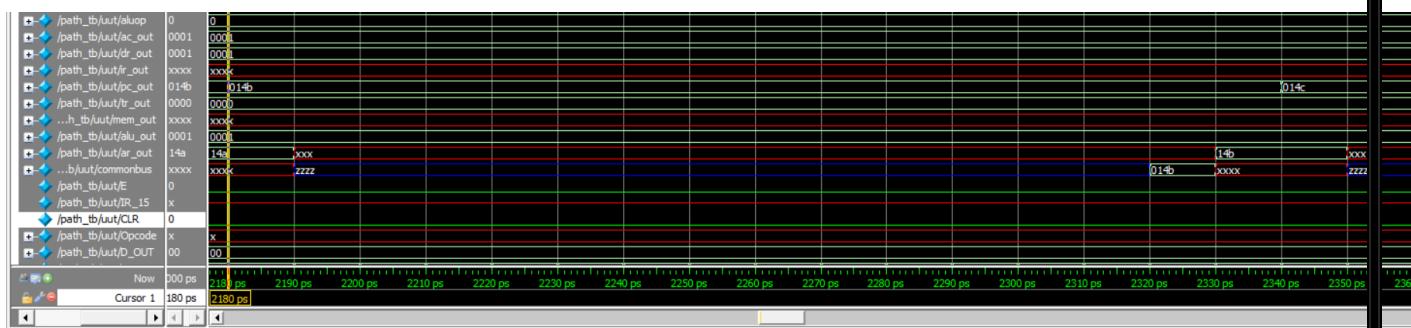
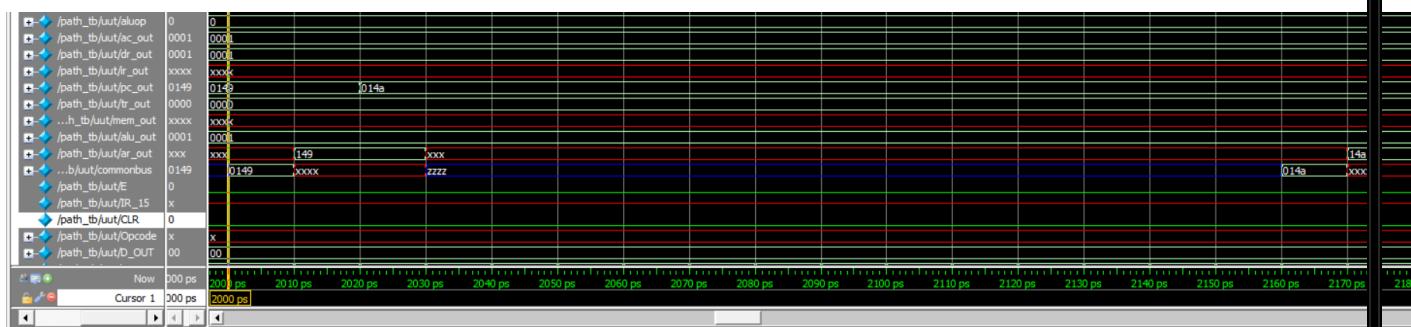
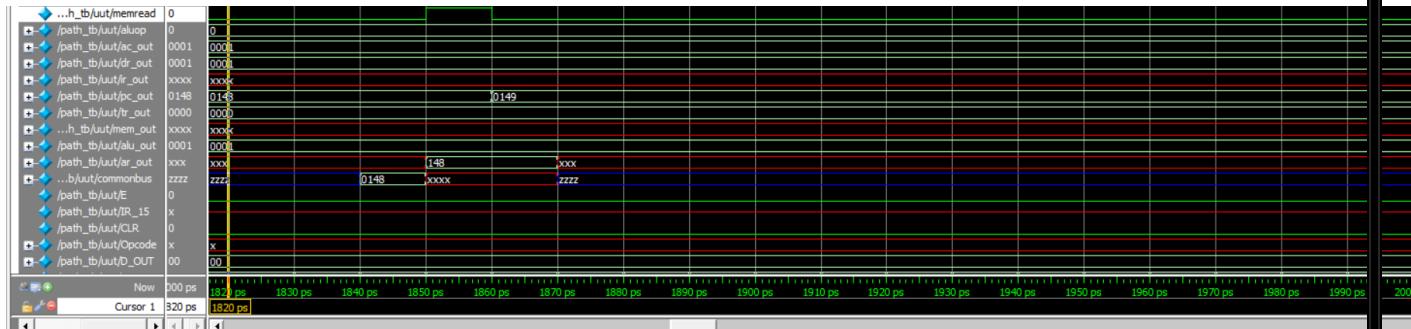
1690, AR: 147, PC: 0147, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
 1700, AR: 147, PC: 0148, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
 1710, AR: xxx, PC: 0148, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
 1850, AR: 148, PC: 0148, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
 1860, AR: 148, PC: 0149, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
 1870, AR: xxx, PC: 0149, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0

1850 – 2360

- At time 1850:
 - AR: 148, PC: 0148, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, Memo Location: 4133, E: 0
 - The memory value at address 148 is loaded into DR.
- At time 1860:
 - AR: 148, PC: 0149, DR: 0002 (after increment), AC: 0001, IR: xxxx, TR: 0000, Memo Location: 4133, E: 0
 - DR is incremented, and the new value (0002) is stored back in memory.

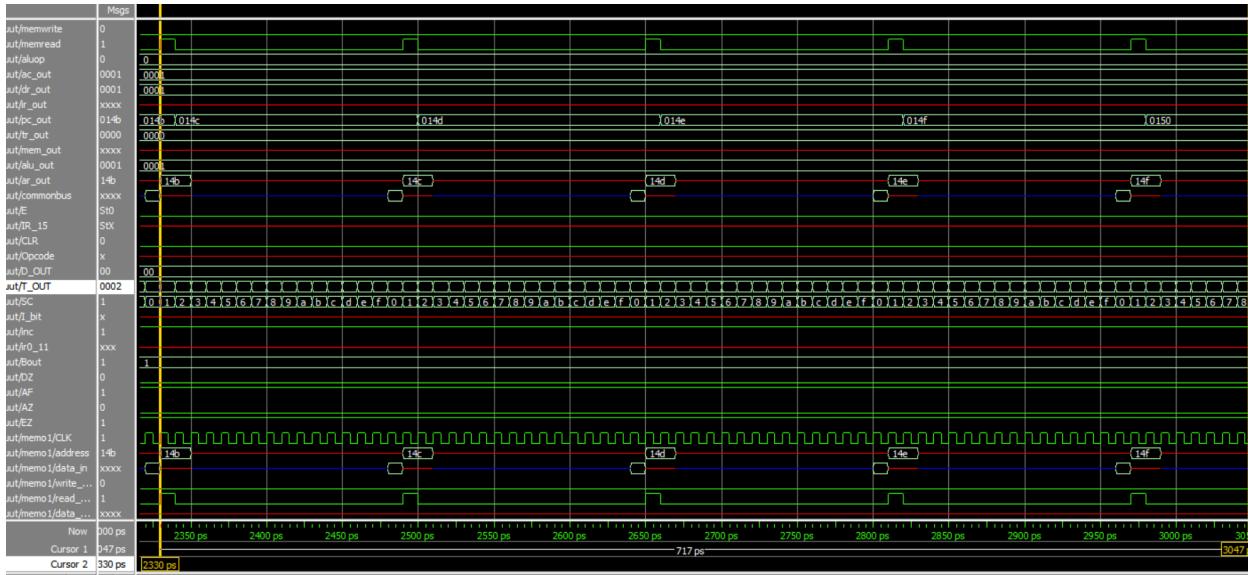
- **At time 1870:**
 - **AR: xxx, PC: 0149, DR: 0002, AC: 0001, IR: xxxx, TR: 0000, Memo Location: 4133, E: 0**
 - The AR becomes undefined, potentially indicating a transition state or undefined memory access.
- **At time 2010:**
 - **AR: 149, PC: 0149, DR: 0002, AC: 0001, IR: xxxx, TR: 0000, Memo Location: 4133, E: 0**
 - The AR is updated to 149, with the PC unchanged from the previous state.
- **At time 2020:**
 - **AR: 149, PC: 014A, DR: 0002, AC: 0001, IR: xxxx, TR: 0000, Memo Location: 4133, E: 0**
 - The PC is incremented to 014A, indicating the program continues its execution without skipping.
- **At time 2030:**
 - **AR: xxx, PC: 014A, DR: 0002, AC: 0001, IR: xxxx, TR: 0000, Memo Location: 4133, E: 0**
 - The AR becomes undefined, as before, marking another possible transition.
- **At time 2170:**
 - **AR: 14A, PC: 014A, DR: 0002, AC: 0001, IR: xxxx, TR: 0000, Memo Location: 4133, E: 0**
 - The AR is set to 14A, and the PC remains the same.
- **At time 2180:**
 - **AR: 14A, PC: 014B, DR: 0002, AC: 0001, IR: xxxx, TR: 0000, Memo Location: 4133, E: 0**
 - The PC increments to 014B, and the program continues execution.
- **At time 2190:**
 - **AR: xxx, PC: 014B, DR: 0002, AC: 0001, IR: xxxx, TR: 0000, Memo Location: 4133, E: 0**
 - The AR becomes undefined again, and the cycle proceeds.
- **At time 2330:**

- **AR: 14B, PC: 014B, DR: 0002, AC: 0001, IR: xxxx, TR: 0000, Memo Location: 4133, E: 0**
- The AR remains 14B, with the PC still at 014B.
- **At time 2340:**
 - **AR: 14B, PC: 014C, DR: 0002, AC: 0001, IR: xxxx, TR: 0000, Memo Location: 4133, E: 0**
 - The PC increments to 014C, continuing the program.
- **At time 2350:**
 - **AR: xxx, PC: 014C, DR: 0002, AC: 0001, IR: xxxx, TR: 0000, Memo Location: 4133, E: 0**
 - The AR is undefined again, indicating another transition or cycle.



```
1870, AR: xxx, PC: 0149, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
2010, AR: 149, PC: 0149, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
2020, AR: 149, PC: 014a, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
2030, AR: xxx, PC: 014a, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
2170, AR: 14a, PC: 014a, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
2180, AR: 14a, PC: 014b, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
2190, AR: xxx, PC: 014b, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
2330, AR: 14b, PC: 014b, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
2340, AR: 14b, PC: 014c, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
2350, AR: xxx, PC: 014c, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
2490, AR: 14c, PC: 014c, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
```

- **1850:** AR = 148, PC = 0148, DR = 0001, AC = 0001, IR = xxxx. Memory value at address 148 is loaded into DR.
- **1860:** AR = 148, PC = 0149, DR = 0002 (incremented), AC = 0001, IR = xxxx. DR is incremented, and the new value (0002) is stored in memory.
- **1870:** AR becomes undefined, indicating a transition state.
- **2010:** AR = 149, PC = 0149, DR = 0002, AC = 0001. AR updated to 149, PC remains the same.
- **2020:** AR = 149, PC = 014A. PC increments to 014A, and the program continues.
- **2030:** AR becomes undefined again, marking another transition.
- **2170:** AR = 14A, PC = 014A. AR set to 14A, PC remains unchanged.
- **2180:** AR = 14A, PC = 014B. PC increments to 014B.
- **2190:** AR becomes undefined again, continuing the cycle.
- **2330:** AR = 14B, PC = 014B. AR remains 14B, PC stays at 014B.
- **2340:** AR = 14B, PC = 014C. PC increments to 014C.
- **2350:** AR becomes undefined again, indicating a transition or cycle.

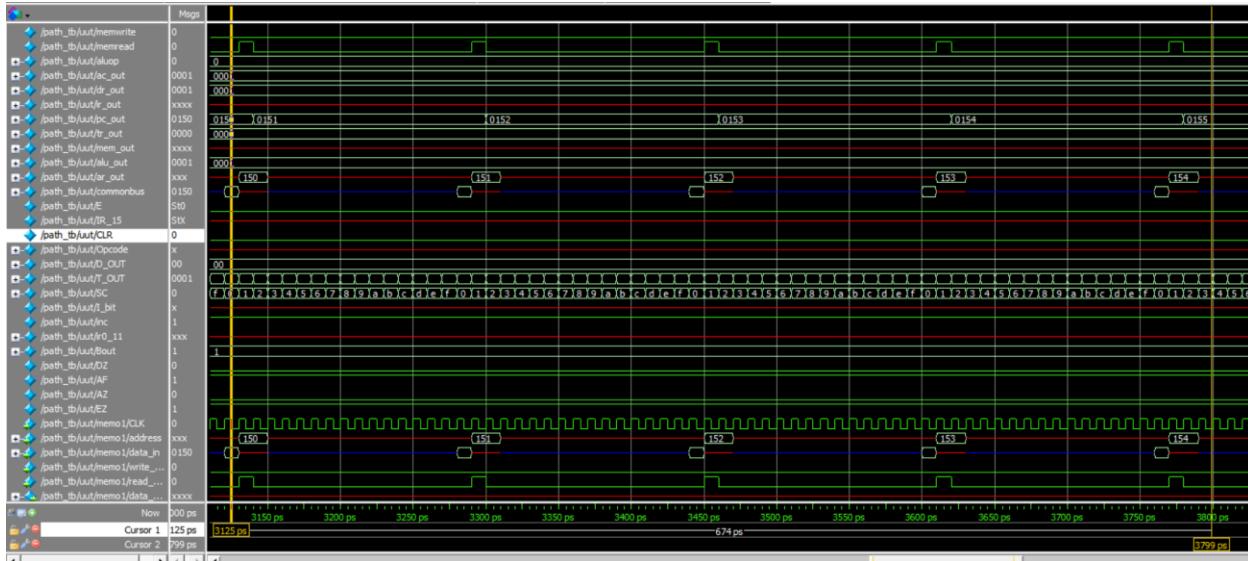


```

# At time 2350, AR: xxx, PC: 014c, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 2490, AR: 14c, PC: 014c, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 2500, AR: 14c, PC: 014d, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 2510, AR: xxx, PC: 014d, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 2650, AR: 14d, PC: 014d, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 2660, AR: 14d, PC: 014e, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 2670, AR: xxx, PC: 014e, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 2810, AR: 14e, PC: 014e, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 2820, AR: 14e, PC: 014f, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 2830, AR: xxx, PC: 014f, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 2970, AR: 14f, PC: 014f, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 2980, AR: 14f, PC: 0150, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 2990, AR: xxx, PC: 0150, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 3130, AR: 150, PC: 0150, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0

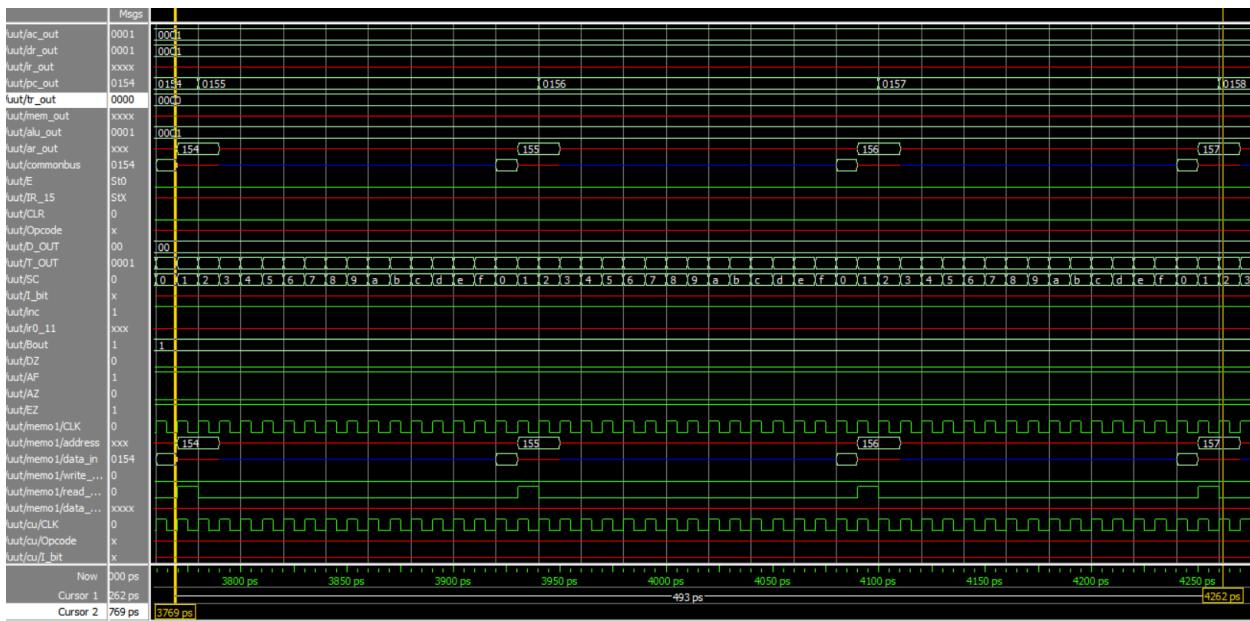
```

- 2350: AR=xxx, PC=014c.**
- 2490-2500: AR=14c, PC increments to 014d.**
- 2510: AR=xxx, PC remains 014d.**
- 2650-2660: AR=14d, PC increments to 014e.**
- 2670: AR=xxx, PC remains 014e.**
- 2810-2820: AR=14e, PC increments to 014f.**
- 2830: AR=xxx, PC remains 014f.**
- 2970-2980: AR=14f, PC increments to 0150.**



```
# At time 3125 ps, AR: 150, PC: 0150, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 3140, AR: 150, PC: 0151, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 3150, AR: xxx, PC: 0151, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 3290, AR: 151, PC: 0151, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 3300, AR: 151, PC: 0152, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 3310, AR: 152, PC: 0152, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 3450, AR: 152, PC: 0152, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 3460, AR: 152, PC: 0153, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 3470, AR: xxx, PC: 0153, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 3610, AR: 153, PC: 0153, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 3620, AR: 153, PC: 0154, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 3630, AR: xxx, PC: 0154, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 3770, AR: 154, PC: 0154, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 3780, AR: 154, PC: 0155, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
# At time 3790, AR: xxx, PC: 0155, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
```

- 3150: AR=xxx, PC=0151.**
- 3290-3300: AR=151, PC increments to 0152.**
- 3310: AR=xxx, PC remains 0152.**
- 3450-3460: AR=152, PC increments to 0153.**
- 3470: AR=xxx, PC remains 0153.**
- 3610-3620: AR=153, PC increments to 0154.**
- 3630: AR=xxx, PC remains 0154.**

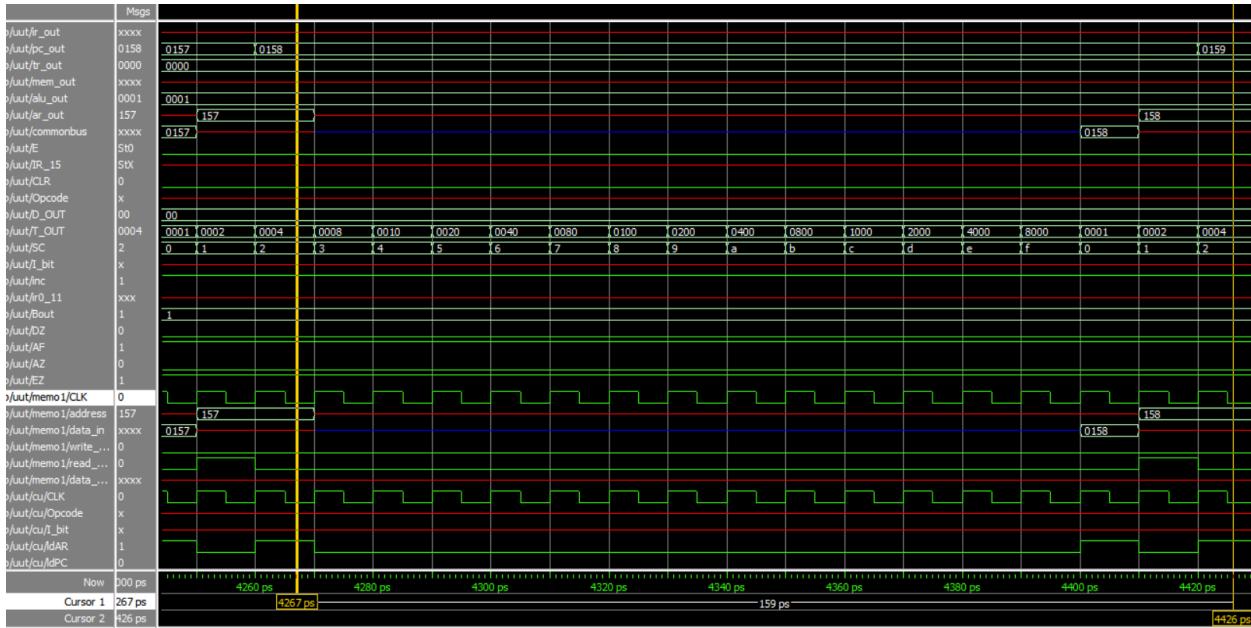


```

3770, AR: 154, PC: 0154, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
3780, AR: 154, PC: 0155, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
3790, AR: xxx, PC: 0155, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
3930, AR: 155, PC: 0155, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
3940, AR: 155, PC: 0156, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
3950, AR: xxx, PC: 0156, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
4090, AR: 156, PC: 0156, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
4100, AR: 156, PC: 0157, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
4110, AR: xxx, PC: 0157, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
4250, AR: 157, PC: 0157, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
4260, AR: 157, PC: 0158, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
4270, AR: xxx, PC: 0158, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0

```

- **3770-3780:** AR=154, PC increments from 0154 to 0155.
- **3790:** AR=xxx, PC remains 0155.
- **3930-3940:** AR=155, PC increments to 0156.
- **3950:** AR=xxx, PC remains 0156.
- **4090-4100:** AR=156, PC increments to 0157.
- **4110:** AR=xxx, PC remains 0157.
- **4250-4260:** AR=157, PC increments to 0158.
- **4270:** AR=xxx, PC remains 0158.



4260, AR: 157, PC: 0158, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
 4270, AR: xxx, PC: 0158, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
 4410, AR: 158, PC: 0158, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
 4420, AR: 158, PC: 0159, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
 4430, AR: xxx, PC: 0159, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0

□ Time 2180:

- **AR = 14A, PC = 014B**: Address Register is 14A, Program Counter is 014B.
- **DR = 0001, AC = 0001**: Data Register and Accumulator hold 1.
- **Memo = 4133, E = &H0**: Memory location 4133, E flag cleared.

□ Time 2190:

- **AR = XXX**: Address Register transitions to unknown.
- Other values remain unchanged.

□ Time 2330:

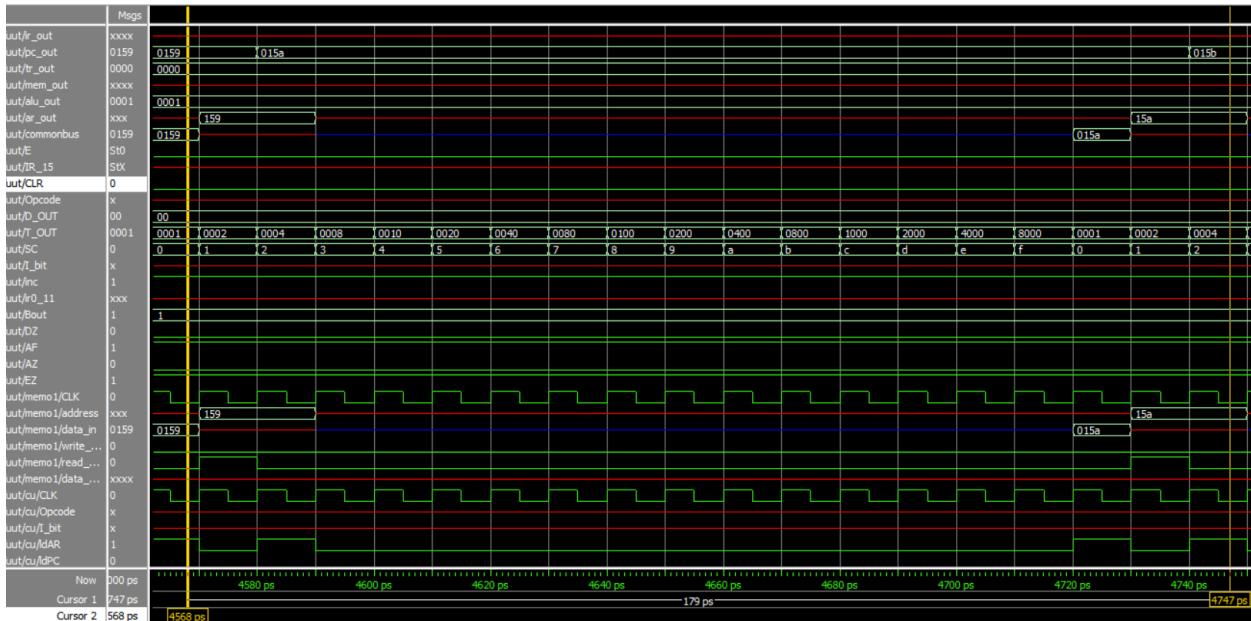
- **AR = 14B, PC = 014B**: Address Register and Program Counter align at 14B.

□ Time 2340:

- **PC \leftarrow PC + 1**: Program Counter increments to 014C.

□ Time 2350:

- **AR = XXX**: Address transitions to unknown.



```

4570, AR: 159, PC: 0159, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
4580, AR: 159, PC: 015a, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
4590, AR: xxx, PC: 015a, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
4730, AR: 15a, PC: 015a, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
4740, AR: 15a, PC: 015b, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0

```

□ Time 4570:

- AR = 159, PC = 0159: Address and Program Counter are at 159.
- DR = 0001, AC = 0001: Data Register and Accumulator hold 1.
- Memo = 4133, E = &H0: Memory at 4133 and E flag cleared.

□ Time 4580:

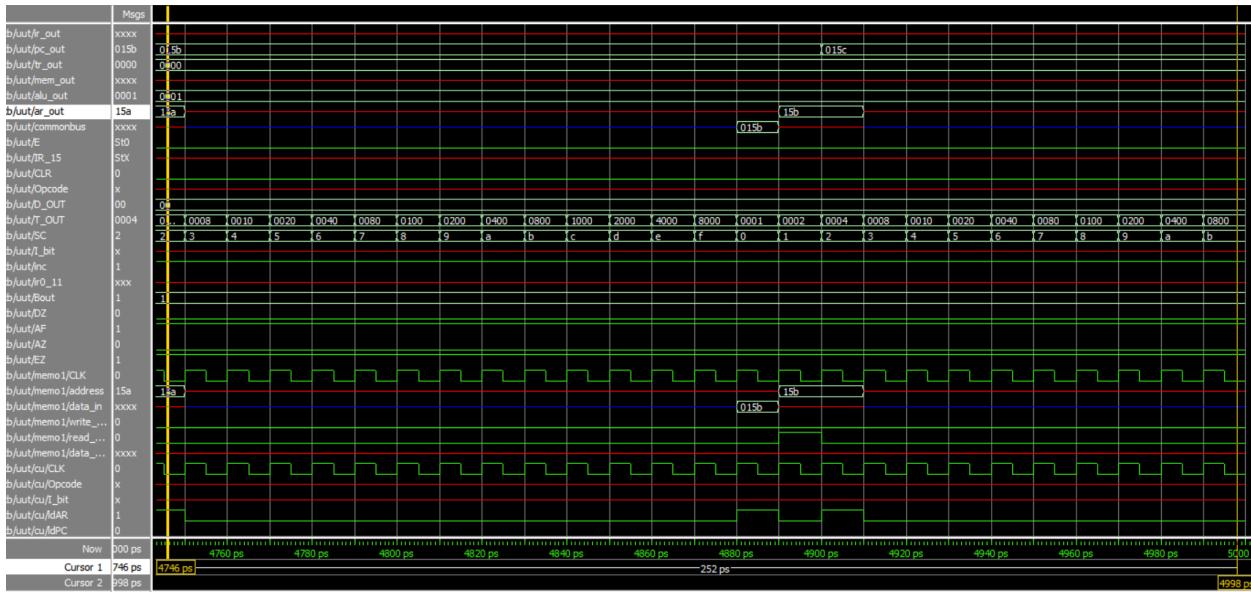
- AR = 159, PC \leftarrow PC + 1: Program Counter increments to 015A.

□ Time 4590:

- AR = XXX: Address transitions to unknown.
- Other values remain unchanged.

□ Time 4730:

- AR = 15A, PC = 015A: Address and Program Counter update.
- Registers and memory hold previous values.



4730, AR: 15a, PC: 015a, DR: 0001, AC: xxxx, IR: 0000, memo location:4133, E:&H0
 4740, AR: 15a, PC: 015b, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
 4750, AR: xxx, PC: 015b, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
 4890, AR: 15b, PC: 015b, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
 4900, AR: 15b, PC: 015c, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0
 4910, AR: xxx, PC: 015c, DR: 0001, AC: 0001, IR: xxxx, TR: 0000, memo location:4133, E:&H0

Initial State (Time = 4730):

- AR = 15A, PC = 015A:** The Address Register holds the value (15A), and the Program Counter points to the current instruction (015A).
- DR = 0001, AC = 0001:** The Data Register and Accumulator are initialized with a value of 1.
- IR, TR = XXXX, 0000:** The Instruction Register is not yet updated, while the Temporary Register is cleared.

Instruction Fetch and PC Increment (Time = 4740):

- AR = 15A:** The Address Register retains its value.
- PC \leftarrow PC + 1:** The Program Counter increments to 015B, preparing for the next instruction.

Memory Access and Transition (Time = 4750):

- AR = XXX:** The Address Register transitions to the operand address (value not provided).
- PC = 015B, DR = 0001, AC = 0001:** The Program Counter, Data Register, and Accumulator remain unchanged, holding their respective values.

