

## **ECE 745 – PROJECT REPORT**

### **Project Group 15**

NIKHIL KHATU (NGKHATU)

SAMARTH ASTHANA (SASTHAN3)

SONAM CHAKRAVARTI (SCHAKRA8)

**Data Def. used:** Data\_defs\_bug1.vp

**Bug Stage:** CONTROL STAGE

**Bug Title:** [CHECKER\_CONTROLLER\_BUG] control\_enable\_updatePC VALUE MISMATCH B/N DUT = 1  
GoldenRef = 0

#### **Description:**

Enable updatePC should be set to one along with enable execute and enable writeback being set to 0 only when a branch is taken. But here, the branch is not take and still it is set to high

#### **Score Board output:**

```
# INPUT DUT control_enable_fetch = 0
# INPUT DUT control_enable_updatePC = 1
# INPUT DUT control_enable_decode = 0
# INPUT DUT control_enable_execute = 0
# INPUT DUT control_enable_writeback = 0
# control_enable_fetch = 0
# control_enable_updatePC = 0
# control_enable_decode = 0
# control_enable_execute = 0
# output control_enable_writeback = 0
# output control_br_taken = 0
# output control_bypass_alu_1 = 0
# output control_bypass_alu_2 = 0
```

**Data Def. used:** Data\_defs\_bug2.vp

**Bug Stage:** CONTROL STAGE

**Bug Title:** 12110ns: [CHECKER\_CONTROLLER\_BUG] control\_bypass\_alu\_1 VALUE MISMATCH B/N DUT = 1 GoldenRef = 0

**Description:** Discrepancy in setting bypass\_alu\_1 in DUT. The condition says the bypass\_alu\_1 should only be set there is a dependency. The destination and src in IR execute and IR are not same and hence should not be set

**Score Board output:**

# 4970ns: [CHECKER\_CONTROLLER\_BUG] control\_bypass\_alu\_1 VALUE MISMATCH B/N DUT = 1 GoldenRef = 0

#

# INPUT control\_IR = 0f43

# INPUT control\_IR\_Exec = 1a6a

# INPUT control\_IMem\_dout = 0f43

# INPUT control\_complete\_data = 1

# INPUT control\_complete\_instr = 1

# INPUT control\_nzp = 0

# INPUT control\_psr = 1

# INPUT DUT Mem State = 3

# INPUT DUT control\_br\_taken = 0

**Data Def. used:** Data\_defs\_bug3.vp

**Bug Stage:** CONTROL STAGE

**Bug Title:** nables high at wrong time one cycle ahead for branch

**Description:** should take three cycles taking 2

**Score Board output:**

**Data Def. used:** Data\_defs\_bug4.vp

**Bug Stage:** EXECUTE STAGE

**Bug Title:** [CHECKER\_EXECUTE\_BUG] ex\_aluout VALUE MISMATCH B/N DUT = xxxxxxxxxxxxxxxx GoldenRef = 0000000000000010

**Description:** alu out in the LC3 microcontroller is output of execute stage and the calculation for same depends on instruction type IR[15:12], values at aluin1 and aluin2. When the bug appears, in the previous cycle, the values are as follows:

ex\_IR=0001010010000001

ex\_bypass\_alu\_1 value INPUT = 0

ex\_bypass\_alu\_2 value INPUT = 1

ex\_bypass\_mem\_1 value INPUT = 0

ex\_bypass\_mem\_2 value INPUT = 0

and Alu control from ex\_E\_control value INPUT = 000001 comes out to be 00. So ideally, the aluin2 pin should have aluout as the value as since ex\_bypass\_alu\_2 is high. But it appears that instead of aluout value, mem\_bypass\_value is selected for the instruction and hence the bug appears in the next cycle on comparison

### Score Board output:

A small part of buggy output is as below:

```
350[SCOREBOARD] Grabbing Data From both Driver and Receiver
#      350ns: [CHECKER] ex_IR value DUT = 0001010010000001
#      350ns: [CHECKER] ex_E_control value INPUT = 000001
#      350ns: [CHECKER] ex_IR value INPUT = 0001010010000001
#      350ns: [CHECKER] ex_npc_in value INPUT = 0011000000001100
#      350ns: [CHECKER] ex_VSR1 value INPUT = 0000000000000000
#      350ns: [CHECKER] ex_VSR2 value INPUT = 0000000000000000
#      350ns: [CHECKER] ex_bypass_alu_1 value INPUT = 0
#      350ns: [CHECKER] ex_bypass_alu_2 value INPUT = 1
#      350ns: [CHECKER] ex_bypass_mem_1 value INPUT = 0
#      350ns: [CHECKER] ex_bypass_mem_2 value INPUT = 0
#      350ns: [CHECKER] ex_Mem_control_in value INPUT = 0
#      350ns: [CHECKER] ex_W_Control_in value INPUT = 00
#      350ns: [CHECKER] ex_Mem_bypass_value value INPUT = xxxxxxxxxxxxxxxx
#      350ns: [CHECKER] ex_enable_execute value INPUT = 1
#      350ns: [CHECKER] buf_ex_aluout value Output = 0000000000000010
#      350ns: [CHECKER] buf_ex_IR_exec value Output = 0001010010000001
#      350ns: [CHECKER] buf_ex_W_Control_out value Output = 00
#      350ns: [CHECKER] buf_ex_Mem_control_out value Output = 0
#      350ns: [CHECKER] buf_ex_M_data value Output = 0000000000000010
#      350ns: [CHECKER] buf_ex_NZP value Output = 000
```

```

#      350ns: [CHECKER] buf_ex_dr value Output = 010
#      370[RECEIVER] Getting Payload
#      370[RECEIVER] Payload Obtained
#      370[DRIVER] Sending in new packet BEGIN
#      370ns: [DRIVER] Sending Payload Begin
#      370ns: [DRIVER] Sending in new packet END
#      370ns: [DRIVER] Number of packets sent =      15
#      370ns: [DRIVER] Number of packets in generator mailbox =      985
# [SCOREBOARD]Packet Found
#      370[SCOREBOARD] Grabbing Data From both Driver and Receiver
#      370ns: [CHECKER_EXECUTE_BUG] ex_aluout VALUE MISMATCH B/N DUT = xxxxxxxxxxxxxxxx GoldenRef =
0000000000000010
#

```

**Data Def. used:** Data\_defs\_bug5.vp

**Bug stage:** EXECUTE STAGE

**Bug Title:** [CHECKER\_EXECUTE\_BUG] ex\_NZP VALUE MISMATCH B/N DUT = 5 GoldenRef = 7

**Description:** Here NZP flag is being calculated wrongly by execute stage. The NZP value of 7 is calculated by golden reference for JMP/Branch instructions But whenever a JMP instruction comes, DUT calculates it to be 3'b101 instead of 3'b111. It appears that Bit 1 in NZP[2:0] is stuck at 0 since no other logic comes into play for calculating NZP flag for JMP instructions and they are directly set to 3'b111 based on instruction type.

**Score Board output:**

Small part of Score Board output is as follows:

```

# [SCOREBOARD]Packet Found
#      1610[SCOREBOARD] Grabbing Data From both Driver and Receiver
#      1610ns: [CHECKER] ex_IR value DUT = 0000111011011111
#      1610ns: [CHECKER] ex_E_control value INPUT = 000110
#      1610ns: [CHECKER] ex_IR value INPUT = 0000111011011111
#      1610ns: [CHECKER] ex_npc_in value INPUT = 0011000001000111
#      1610ns: [CHECKER] ex_VSR1 value INPUT = 1111111111110011
#      1610ns: [CHECKER] ex_VSR2 value INPUT = 0000000000000001
#      1610ns: [CHECKER] ex_bypass_alu_1 value INPUT = 0
#      1610ns: [CHECKER] ex_bypass_alu_2 value INPUT = 0
#      1610ns: [CHECKER] ex_bypass_mem_1 value INPUT = 0

```

```
#      1610ns: [CHECKER] ex_bypass_mem_2 value INPUT = 0
#      1610ns: [CHECKER] ex_Mem_control_in value INPUT = 0
#      1610ns: [CHECKER] ex_W_Control_in value INPUT = 00
#      1610ns: [CHECKER] ex_Mem_bypass_value value INPUT = 010001000010111
#      1630[DRIVER] Sending in new packet BEGIN
#      1630ns: [DRIVER] Sending Payload Begin
#      1630ns: [DRIVER] Sending in new packet END
#      1630ns: [DRIVER] Number of packets sent =      78
#      1630ns: [DRIVER] Number of packets in generator mailbox =    922
# [SCOREBOARD]Packet Found
#      1630[SCOREBOARD] Grabbing Data From both Driver and Receiver
#      1630ns: [CHECKER_EXECUTE_BUG] ex_NZP VALUE MISMATCH B/N DUT = 5 GoldenRef = 7
#
#      1630ns: [CHECKER] ex_IR value DUT = 0000111011011111
```