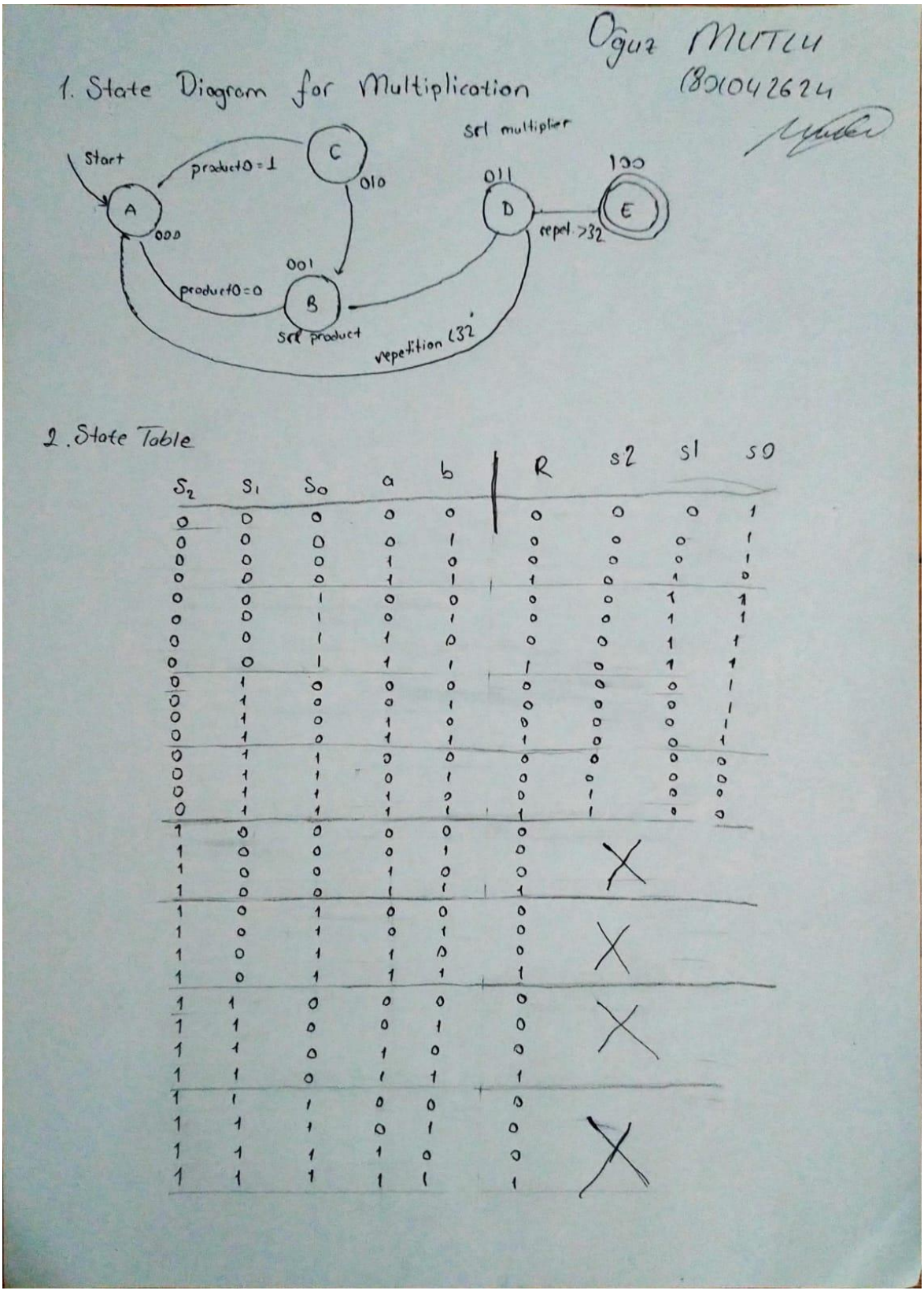
CSE 331

COMPUTER ORGANIZATION

HW 3 REPORT

- OGUZ MUTLU 1801042624

1.State Diagram and Table



Oğuz MUTLU
180104624

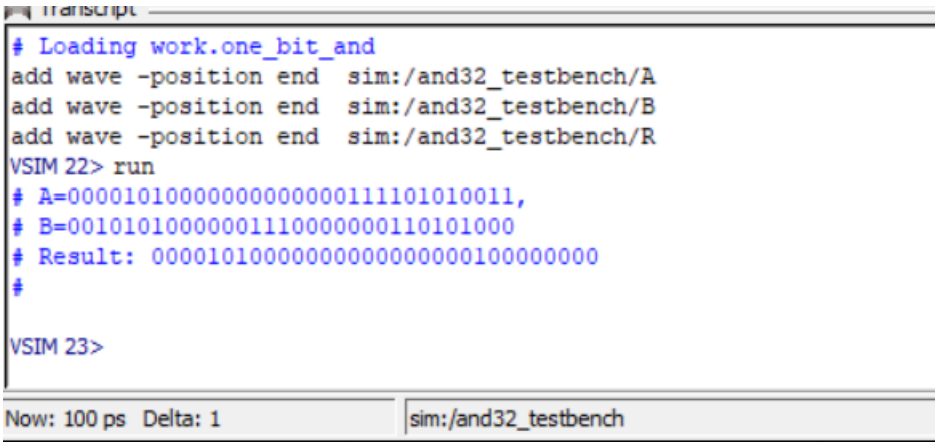
1. State Diagram for Multiplication

2. State Table

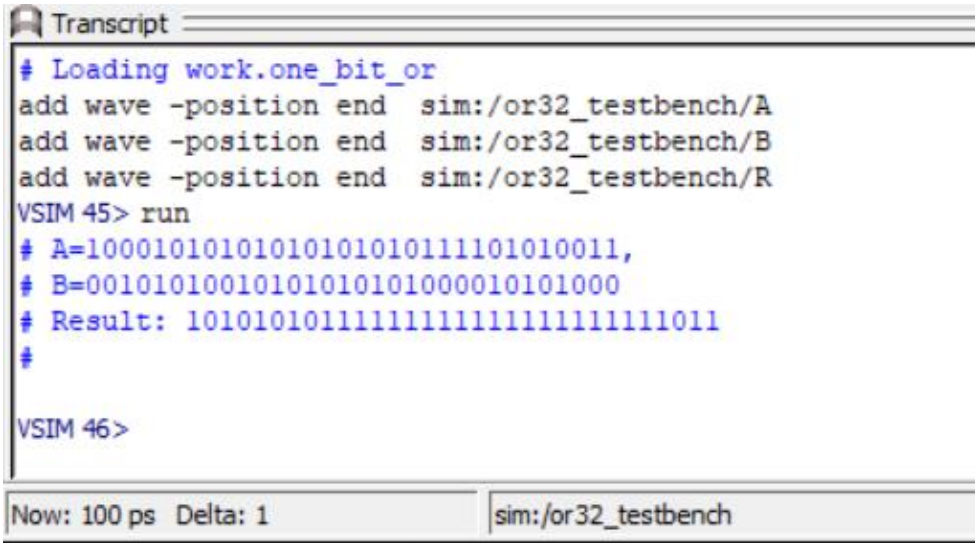
S_2	S_1	S_0	a	b	R	s2	s1	s0
0	0	0	0	0	0	0	0	1
0	0	0	0	1	0	0	0	1
0	0	0	1	0	0	0	0	1
0	0	0	1	1	1	0	1	0
0	0	1	0	0	0	0	1	1
0	0	1	0	1	0	0	1	1
0	0	1	1	0	0	0	1	1
0	0	1	1	1	1	0	1	1
0	1	0	0	0	0	0	0	1
0	1	0	0	1	0	0	0	1
0	1	0	1	0	0	0	0	1
0	1	0	1	1	1	0	0	1
0	1	1	0	0	0	0	0	0
0	1	1	0	1	0	0	0	0
0	1	1	1	0	0	1	0	0
0	1	1	1	1	1	1	0	0
1	0	0	0	0	0			
1	0	0	0	1	0			
1	0	0	1	0	0			
1	0	0	1	1	1			
1	0	1	0	0	0			
1	0	1	0	1	0			
1	0	1	1	0	0			
1	0	1	1	1	1			
1	1	0	0	0	0			
1	1	0	0	1	0			
1	1	0	1	0	0			
1	1	0	1	1	1			
1	1	1	0	0	0			
1	1	1	0	1	0			
1	1	1	1	0	0			
1	1	1	1	1	1			

2.Test Results

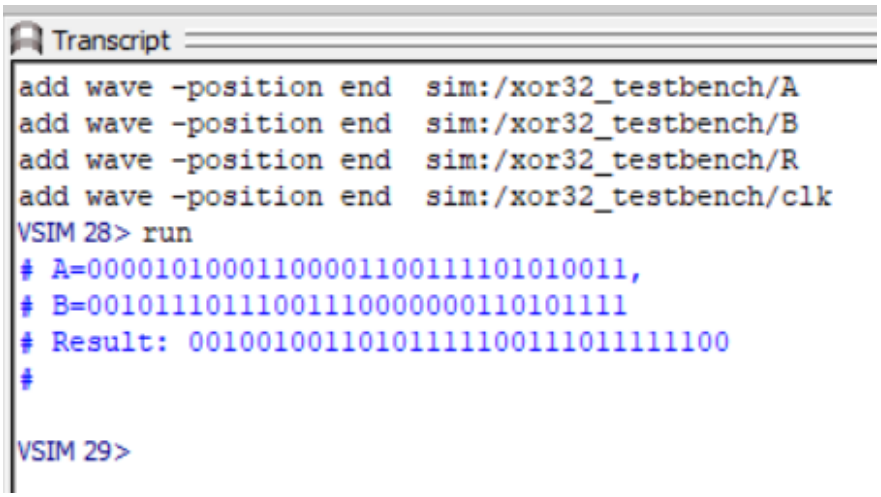
1. and32



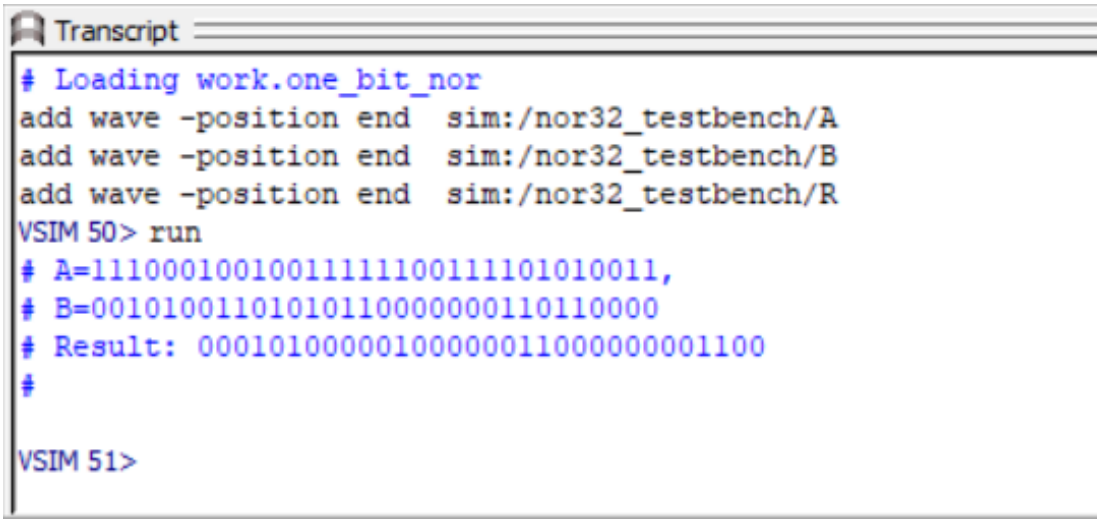
2. or32



3.xor32



4.nor32



5. slt32

```
Transcript
#         Region: /slt32_testbench/slt32_inst/g0
add wave -position end  sim:/slt32_testbench/A
add wave -position end  sim:/slt32_testbench/B
add wave -position end  sim:/slt32_testbench/R
VSIM 40> run
# A=11111011001101000111001101010001,
# B=01100010101101010111110110101111
# Result: 11111111111111111111111111111111
#

VSIM 41>
```

6. add32

```
Transcript
add wave -position end  sim:/adder32_testbench/R
add wave -position end  sim:/adder32_testbench/Cout
VSIM 17> run
# A=11100010010011111100001101010011,
# B=01101001101111111101100011011000O
# Cin=1
# Cout=1
# Result: 01001100000011110111010100000100
#

VSIM 18>
```

Now: 100 ps Delta: 1 | sim:/adder32_testbench

7. sub32

Transcript

```
add wave -position end  sim:/subtractor32_testbench/R
add wave -position end  sim:/subtractor32_testbench/Bout
VSIM 35> run
# A=11100010010011111100001101010011,
# B=01101001101111111011000110110000
# Bin=1
# Bout=0
# Result: 01111000100100000001000110100010
#

VSIM 36>
```

8.ALU32

Transcript

```
# A=00001010000000000000111101010011,
# B=00101010000001110000000110101000,
# Result: 00110100000001110001000011111011
# , ALUop: 000
#
# A=00001010000000000000111101010011,
# B=00101010000001110000000110101000,
# Result: 00100000000001110000111011111011
# , ALUop: 001
#
# A=00001010000000000000111101010011,
# B=00101010000001110000000110101000,
# Result: 11011111111111001000110110101011
# , ALUop: 010
#
# A=00001010000000000000111101010011,
# B=00101010000001110000000110101000,
# Result: 00000000000000000000000000000000
# , ALUop: 011
#
# A=00001010000000000000111101010011,
# B=00101010000001110000000110101000,
# Result: 11111111111111111111111111111111
# , ALUop: 100
#
# A=00001010000000000000111101010011,
# B=00101010000001110000000110101000,
# Result: 11010101111111000111100000000100
# , ALUop: 101
#
# A=00001010000000000000111101010011,
# B=00101010000001110000000110101000,
# Result: 00101010000001110000111111111011
# , ALUop: 111
#

VSIM 57>
```