

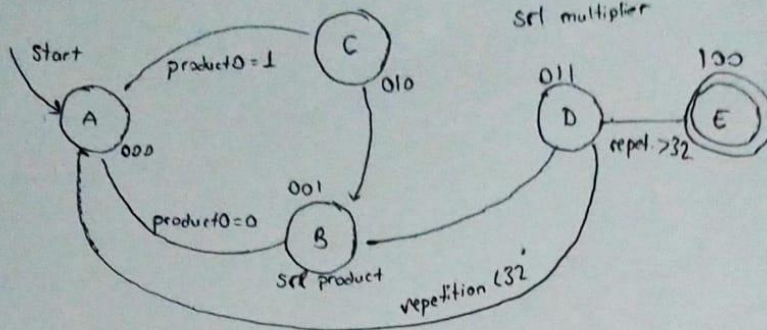
CSE 331
COMPUTER ORGANIZATION

HW 3 REPORT

- OGUZ MUTLU 1801042624

1.State Diagram and Table

1. State Diagram for Multiplication



Oguz MUTLU
1801042624

[Signature]

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

```
Transcript
# Loading work.one_bit_and
add wave -position end  sim:/and32_testbench/A
add wave -position end  sim:/and32_testbench/B
add wave -position end  sim:/and32_testbench/R
VSIM 22> run
# A=00001010000000000000111101010011,
# B=00101010000001110000000110101000
# Result: 00001010000000000000000100000000
#
VSIM 23>

Now: 100 ps Delta: 1 sim:/and32_testbench
```

2. or32

```
Transcript
# Loading work.one_bit_or
add wave -position end  sim:/or32_testbench/A
add wave -position end  sim:/or32_testbench/B
add wave -position end  sim:/or32_testbench/R
VSIM 45> run
# A=10001010101010101010111101010011,
# B=00101010010101010101000010101000
# Result: 1010101011111111111111111111011
#
VSIM 46>

Now: 100 ps Delta: 1 sim:/or32_testbench
```

3.xor32

```
Transcript
add wave -position end  sim:/xor32_testbench/A
add wave -position end  sim:/xor32_testbench/B
add wave -position end  sim:/xor32_testbench/R
add wave -position end  sim:/xor32_testbench/clk
VSIM 28> run
# A=00001010001100001100111101010011,
# B=00101110111001110000000110101111
# Result: 00100100110101111100111011111100
#
VSIM 29>
```

4.nor32

```
Transcript
# Loading work.one_bit_nor
add wave -position end  sim:/nor32_testbench/A
add wave -position end  sim:/nor32_testbench/B
add wave -position end  sim:/nor32_testbench/R
VSIM 50> run
# A=11100010010011111100111101010011,
# B=00101001101010110000000110110000
# Result: 00010100000100000011000000001100
#
VSIM 51>
```

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: 111111111111111111111111111111
#

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=011010011011111111011000110110000
# Cin=1
# Cout=1
# Result: 010011000000011110111010100000100
#

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=000010100000000000000111101010011,
# B=001010100000001110000000110101000,
# Result: 00110100000001110001000011111011
# , ALUop: 000
#
# A=000010100000000000000111101010011,
# B=001010100000001110000000110101000,
# Result: 00100000000001110000111011111011
# , ALUop: 001
#
# A=000010100000000000000111101010011,
# B=001010100000001110000000110101000,
# Result: 1101111111110010000110110101011
# , ALUop: 010
#
# A=000010100000000000000111101010011,
# B=001010100000001110000000110101000,
# Result: 00000000000000000000000000000000
# , ALUop: 011
#
# A=000010100000000000000111101010011,
# B=001010100000001110000000110101000,
# Result: 11111111111111111111111111111111
# , ALUop: 100
#
# A=000010100000000000000111101010011,
# B=001010100000001110000000110101000,
# Result: 11010101111110001111000000000100
# , ALUop: 101
#
# A=000010100000000000000111101010011,
# B=001010100000001110000000110101000,
# Result: 00101010000001110000111111111011
# , ALUop: 111
#
VSIM 57>
```