CSE 331 – Project 3

Ozan Şelte – 161044061

alu_32_msb_tb:

Sums 2 32bit numbers. Numbers are hexadecimals.

001 22222222 + 33333333 = 55555555 # 001 12341234 + 66667777 = 789a89ab

conc_tb:

Concatenates 4 8bit numbers. Numbers are hexadecimals.

01,23,45,67 = 67452301 67,45,23,01=01234567

ctrl_unit_tb:

# op:20	alu:001	slct:00	regW:1	memW:0	sign:0
# op:24	alu:001	slct:00	regW:1	memW:0	sign:1
# op:21	alu:001	slct:01	regW:1	memW:0	sign:0
# op:25	alu:001	slct:01	regW:1	memW:0	sign:1
# op:0f	alu:001	slct:10	regW:1	memW:0	sign:1
# op:23	alu:001	slct:11	regW:1	memW:0	sign:1
# op:28	alu:001	slct:00	regW:0	memW:1	sign:1
# op:29	alu:001	slct:01	regW:0	memW:1	sign:1
# op:2b	alu:001	slct:11	regW:0	memW:1	sign:1

extender_16_32_tb:

Extendes 16bit number to 32bit number.

# zero	1010 1010 1010 1010	→	0000 0000 0000 0000 1010 1010 1010 1010
# sgnd	1010 1010 1010 1010	→	1111 1111 1111 1111 1010 1010 1010 1010

extender_upper_tb:

1010 1010 1010 1010 → 1010 1010 1010 1010 0000 0000 0000 0000

1110 1100 1100 1111 → 1110 1100 1100 1111 0000 0000 0000 0000

filler_8_32_tb: & filler_16_32_tb

Numbers are hexadecimals.

ab → 000000ab # 20 → 00000020

abcd → 0000abcd # 2043 → 00002043

mux32_4_1_tb:

32bit 4:1 Multiplexer

00 11111111 # 01 55555555 # 10 aaaaaaaa # 11 eeeeeeee

Mips32 Tests:

Initial Registers:

R0: 0000 0000 0000 0000 0000 0000 0000 0000

R1: 0000 0000 0000 0000 0000 0000 0000 0001

Initial Memory Data:

M0: 1000 0000 0000 0001 1000 0000 0000 0001

M1: 1001 1111 1110 0000 1100 0001 1001 0001

M2: 1010 1010 1010 1010 1010 1010 1010 1010

Instructions:

Operation	Opcode	RS	RT	IMM
LB	100000	00000	00010	0000 0000 0000 0001
LB	100000	00001	00011	0000 0000 0000 0001
LBU	100100	00000	00100	0000 0000 0000 0001
LBU	100100	00001	00101	0000 0000 0000 0001
LH	100001	00000	00110	0000 0000 0000 0001
LH	100001	00001	00111	0000 0000 0000 0001
LHU	100101	00000	01000	0000 0000 0000 0001

LHU	100101	00001	01001	0000 0000 0000 0001
LUI	001111	00000	01010	0000 0000 0000 0001
LUI	001111	00001	01011	0000 0000 0000 0001
LW	100011	00000	01100	0000 0000 0000 0001
LW	100011	00001	01101	0000 0000 0000 0001
SB	101000	00000	01001	0000 0000 0000 0011
SB	101000	00001	01010	0000 0000 0000 0011
SH	101001	00000	01011	0000 0000 0000 0101
SH	101001	00001	01100	0000 0000 0000 0101
SW	101011	00000	01101	0000 0000 0000 0111
SW	101011	00001	01110	0000 0000 0000 0111

Outputs:

Testbench modifies the registers and the memory. But it prints more than clocks.

Registers:

0000 0000 0000 0000 0000 0000 0000 0000

0000 0000 0000 0000 0000 0000 0000 0001

0000 0000 0000 0000 0000 0000 1001 0001

0000 0000 0000 0000 0000 0000 1010 1010

0000 0000 0000 0000 0000 0000 1001 0001

0000 0000 0000 0000 0000 0000 1010 1010

0000 0000 0000 0000 1100 0001 1001 0001

0000 0000 0000 0000 1010 1010 1010 1010

0000 0000 0000 0000 1100 0001 1001 0001

0000 0000 0000 0000 1010 1010 1010 1010

0000 0000 0000 0001 0000 0000 0000 0000

0000 0000 0000 0001 0000 0000 0000 0000

1001 1111 1110 0000 1100 0001 1001 0001

1010 1010 1010 1010 1010 1010 1010 1010

Memory Data:

1000 0000 0000 0001 1000 0000 0000 0001

1001 1111 1110 0000 1100 0001 1001 0001

1010 1010 1010 1010 1010 1010 1010 1010

0000 0000 0000 0000 0000 0000 1010 1010

0000 0000 0000 0000 0000 0000 0000 0000

0000 0000 0000 0000 0000 0000 0000 0000

0000 0000 0000 0000 1100 0001 1001 0001

1010 1010 1010 1010 1010 1010 1010 1010