LE9.1

Bookmark this page

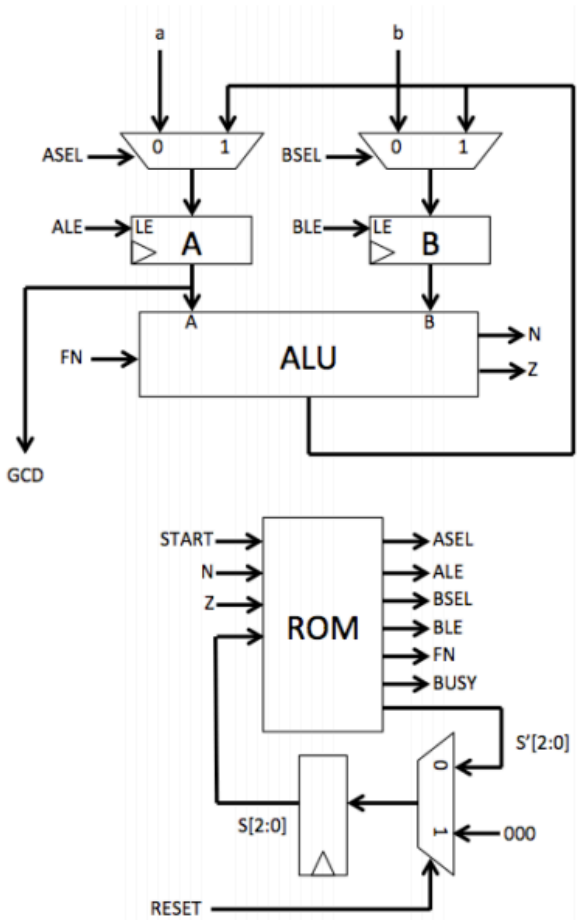
LE9.1.1: GCD

1/1 point (ungraded)

Note: you may find it helpful to view the first Worked Example in this chapter before working on this problem.

Euclid Electronic Computing Systems (EECS) has hired you as a summer intern to work on their latest product: a datapath and FSM controller that computes the greatest common divisor of two *positive* arguments represented in 32-bit two's complement form. EECS is using a variant of Euclid's algorithm, which can be written as

$$\gcd(a,b) = \begin{cases} a & a = b \\ \gcd(a-b,b) & a > b \\ \gcd(a,b-a) & a < b \end{cases}$$



The EECS circuits team has devised the datapath shown to the right, which computes GCD iteratively, holding the current values of the arguments "a" and "b" in the A and B registers respectively. The computation is initiated by placing two values of the "a" and "b" inputs to the datapath, then setting START to 1 until the BUSY output becomes 1. When BUSY becomes 0 again, the answer is available on the GCD output.

Here are the control signals for the datapath:

ASEL 0 selects the "a" input, 1 selects the output of the ALU

ALE load enable signal for the A register (1 = load)

BSEL 0 selects the "b" input, 1 selects the output of the ALU

BLE load enable signal for the B register (1 = load)

FN 0: ALU computes A-B, 1: ALU computes B-A

And here are the status signals from the datapath ALU. When the ALU function is "A-B" the N and Z status signals can be used to determine if A equals B (Z=1), A is less than B (Z=0, N=1), or A is greater than B (Z=0, N=0).

N 1 if ALU result is negative, 0 otherwise

Z 1 if ALU result is zero, 0 otherwise

An intern from the previous summer left the following unfinished table describing the contents of the ROM. Please complete the blank entries in the last two rows by entering either 0 or 1 so that the datapath correctly computes GCD according to the formula given above. If a value doesn't matter and could be either 0 or 1, enter "X", to indicate a don't-care condition.

S[2:0]	Start	N	Z	S'[2:0]	ASEL	ALE	BSEL	BLE
000	0	-	-	000	0	0	0	0
000	1	-	-	001	0	0	0	0
001	-	-	-	010	0	1	0	1
010	-	0	0	011	1	0	1	0
010	-	1	0	100	1	0	1	0
010	-	-	1	000	1	0	1	0
011	-	-	-	010	1 ✔ Answer: 1	1 ✔ Answer: 1	X ✔ Answer: X	0 ✔ Answer: 0
100	-	-	-	010	X ✔ Answer: X	0 ✔ Answer: 0	1 ✔ Answer: 1	1 ✔ Answer: 1

Explanation

The contents of the ROM are basically a truth table that describes the transitions from state to state given

Calculator

particular inputs. It also specifies the value of each output for all states. Note that all outputs associated with current state 010 are the same regardless of the inputs. This suggests that this is a Moore type of FSM (one where the outputs are purely a function of the current state). In order to fill in the values of state 011 and 100, we need to understand what each of those states is supposed to represent. In state 010, the operation A - B is being performed as specified by the ALU function = 0. The results of this subtraction are used to determine which state to go to next.

If the results had N = 0 and Z = 0, this means that A > B so in state 011 gcd(a-b, b) should be computed. This means that the A register should get the results of A - B, so ASEL = 1 and ALE = 1. Since the B register already has B in it from before, BLE = 0, so BSEL = don't care since nothing will be loaded into the B register.

If the results of the original subtraction of A - B resulted in a negative number (N = 1, Z = 0), that means that A < B so in state 100 gcd(a, b-a) should be computed. This means that the B register should get the results of B - A, so BSEL = 1, BLE = 1, and FN = 1 (to calculate B - A instead of A - B). Since the A register already has A in it from before, ALE = 0, so ASEL = don't care since nothing will be loaded into the A register.

Finally, if the results of the original subtraction resulted in zero (Z = 1), then one should go back to state 000 to indicate that you are done with the computation.

Submit

Answers are displayed within the problem

Discussion

Topic: 9. Designing an Instruction Set / LE9.1

Hide Discussion

Add a Post

Show all posts by recent activity

How to understand changes in the datapath when state transition occurs?	22
register select in relation to register load first try i reloaded the registers with their respective input not realizing that once the register is loaded, if the value need not be ch...	1
Exact meaning of ALE or BLE? At first I mixed it with CLK of a register, whose positive transition (0→1) load the inputs of a register to its output. But it seems that i...	6
Update a and b value every iteration In the datapath diagram, it seems a or b is only the initial value of input value, should we update the value of a and b for next iterati...	3
State meanings Something is eluding me with this state diagram. The main question - why do we not assert ALE or BLE going from S2 to either S3 ...	3
Don't get how second diagram relates to first Are is the value of s[2:0] meant to be a, and the value of s'[2,0] = b? ROMs just store things? What is the ROM output BUSY? What ...	3
[STAFF] Clarification in the gcd fsm table	3
LE and Worked Examples : Practice or Graded? Multiple Attempts? Will the LE and Worked Example problems allow multiple attempts? If so, will it be marked as so? If so, when? Do LE and Worked Ex...	2

Previous Next

edX

About
Affiliates

Calculator