Parameterisable Processor Breakdown

Contents

Parameterisable ALU

Simulation Screenshots

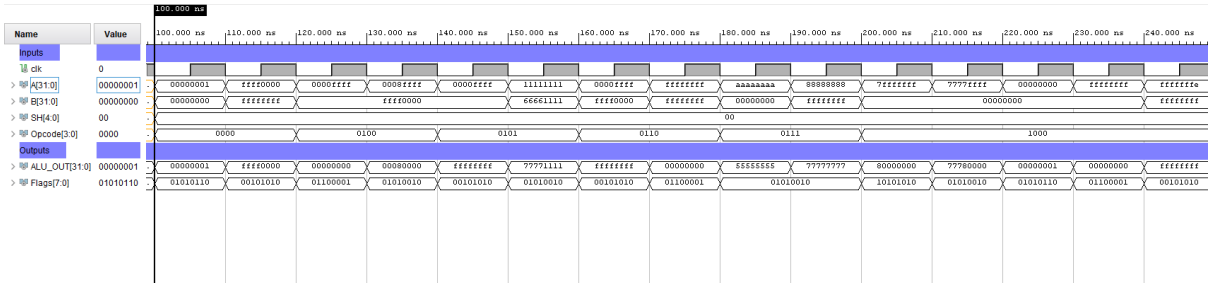


Figure 1

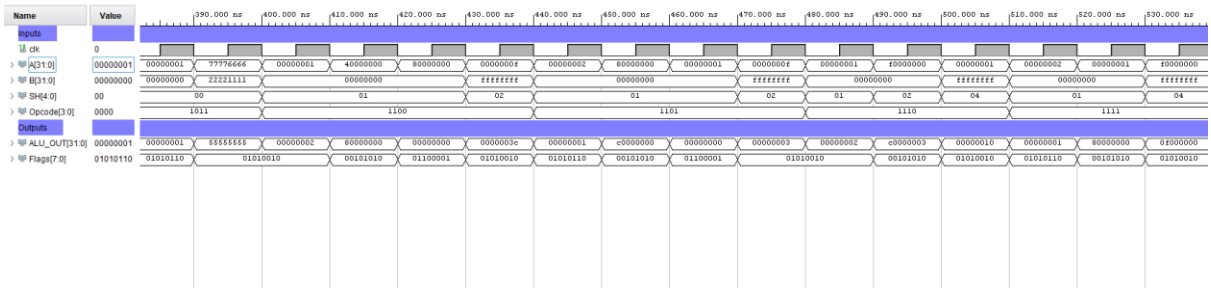


Figure 2

In Figures 1 and 2, various arithmetic logic operations can be seen. The inputs A and B are the pieces of input data which are acted upon. The input SH is for how far the input data is shifted left or right depending on the Opcode input. The Opcode input is the input which decides what operations happen to A and B. The output ALU_OUT is the result of A and/or B after the operation that Opcode corresponds to has been carried out. ALU_OUT is then assessed and flags are generated based off of this information. These flags identify things such as whether ALU_OUT is = 0, >0, <0 etc.

For example, in Figure 2 at 390ns, A = 77776666, B = 22221111, and Opcode = 1011. 1011 for opcode corresponds to subtraction, so ALU_OUT = A – B, and we can see this to be true because ALU_OUT = 55555555, which is equivalent to 77776666 – 22221111. We can also see the flags are 01010010, which means ALU_OUT is >0, >=0, and /=0 – which is all true.

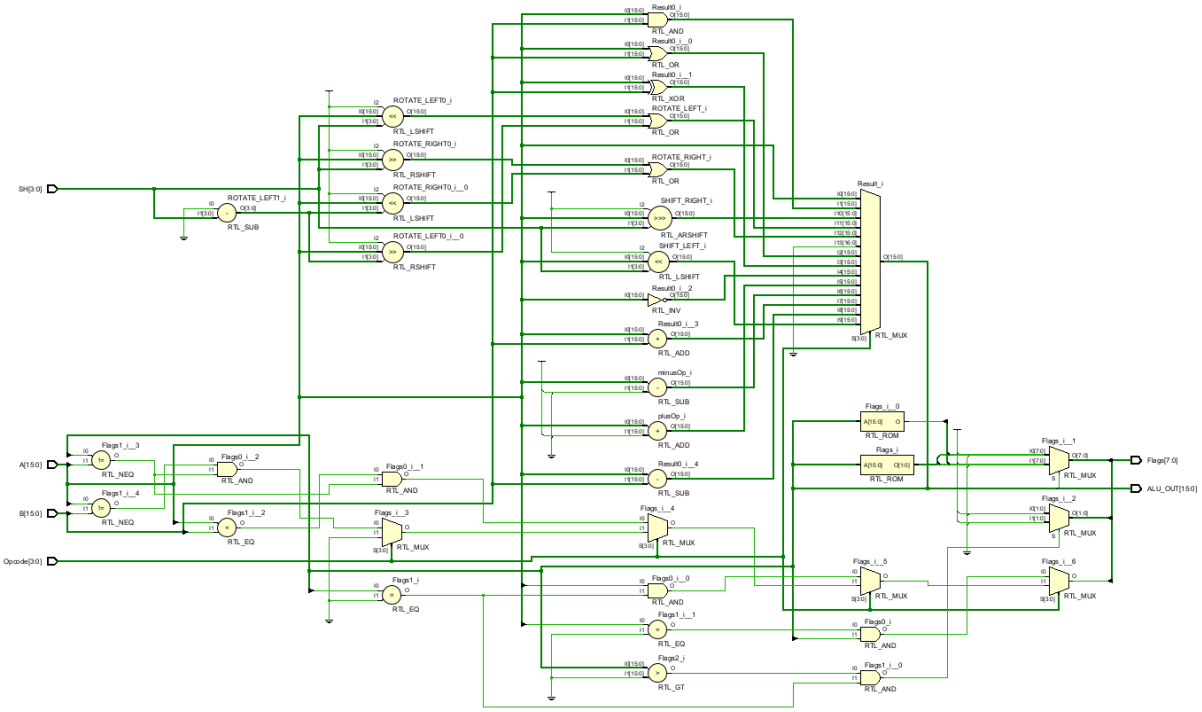
Console Screenshots

```
Time: 100 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 0 passed: expected values matched the actual values.
Time: 110 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 1 passed: expected values matched the actual values.
Time: 120 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 2 passed: expected values matched the actual values.
Time: 130 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 3 passed: expected values matched the actual values.
Time: 140 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 4 passed: expected values matched the actual values.
Time: 150 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 5 passed: expected values matched the actual values.
Time: 160 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 6 passed: expected values matched the actual values.
Time: 170 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 7 passed: expected values matched the actual values.
Time: 180 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 8 passed: expected values matched the actual values.
Time: 190 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 9 passed: expected values matched the actual values.
Time: 200 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 10 passed: expected values matched the actual values.
Time: 210 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 11 passed: expected values matched the actual values.
Time: 220 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 12 passed: expected values matched the actual values.
Time: 230 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 13 passed: expected values matched the actual values.
Time: 240 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 14 passed: expected values matched the actual values.
Time: 250 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 15 passed: expected values matched the actual values.
Time: 260 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 16 passed: expected values matched the actual values.
Time: 270 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 17 passed: expected values matched the actual values.
Time: 280 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 18 passed: expected values matched the actual values.
Time: 290 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 19 passed: expected values matched the actual values.
Time: 300 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 20 passed: expected values matched the actual values.
Time: 310 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 21 passed: expected values matched the actual values.
Time: 320 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 22 passed: expected values matched the actual values.
Time: 330 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 23 passed: expected values matched the actual values.
Time: 340 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 24 passed: expected values matched the actual values.
Time: 350 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 25 passed: expected values matched the actual values.
Time: 360 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 26 passed: expected values matched the actual values.
Time: 370 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 27 passed: expected values matched the actual values.
Time: 380 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 28 passed: expected values matched the actual values.
Time: 390 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 29 passed: expected values matched the actual values.
Time: 400 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 30 passed: expected values matched the actual values.
Time: 410 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 31 passed: expected values matched the actual values.
Time: 420 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 32 passed: expected values matched the actual values.
Time: 430 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 33 passed: expected values matched the actual values.
Time: 440 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 34 passed: expected values matched the actual values.
Time: 450 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 35 passed: expected values matched the actual values.
Time: 460 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 36 passed: expected values matched the actual values.
Time: 470 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 37 passed: expected values matched the actual values.
Time: 480 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 38 passed: expected values matched the actual values.
Time: 490 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 39 passed: expected values matched the actual values.
Time: 500 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 40 passed: expected values matched the actual values.
Time: 510 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 41 passed: expected values matched the actual values.
Time: 520 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 42 passed: expected values matched the actual values.
Time: 530 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test vector 43 passed: expected values matched the actual values.
Time: 540 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Error: Test vector 44 failed for inputs A = 0x00000000 and B = 0x00000000. Expected result = 0x00000000 and flags = 56; observed result = 0x00000001 and flags = 56
Time: 550 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Error: Test vector 45 failed for inputs A = 0x00000000 and B = 0x00000000. Expected result = 0x00000001 and flags = 00; observed result = 0x00000001 and flags = 56
Time: 560 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Error: Test vector 46 failed for inputs A = 0x00000000 and B = 0x00000000. Expected result = 0x00000000 and flags = 00; observed result = 0x00000001 and flags = 56
Time: 570 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
Note: Test completed with 3 errors.
Time: 570 ns  Iteration: 0  Process: /parameterizable_ALU_tb/test_process  File: D:/Lab4_TaskA2/Final Project/Final Project.srcs/sim_1/new/parameterizable_ALU_tb.vhd
```

Synthesis Report – RTL Component Statistics

```
---------------------------------------------------------------------------------
Start RTL Component Statistics
---------------------------------------------------------------------------------
Detailed RTL Component Info :
+---Adders :
       2 Input   16 Bit       Adders := 1
       3 Input   16 Bit       Adders := 1
+---XORs :
       2 Input     16 Bit         XORs := 1
+---Muxes :
      14 Input   16 Bit        Muxes := 2
       2 Input    8 Bit        Muxes := 1
       2 Input    2 Bit        Muxes := 1
       2 Input    1 Bit        Muxes := 2
       4 Input    1 Bit        Muxes := 1
---------------------------------------------------------------------------------
Finished RTL Component Statistics
---------------------------------------------------------------------------------
```

RTL analysis schematics



Parameterisable Register Bank

Simulation Screenshots

Write

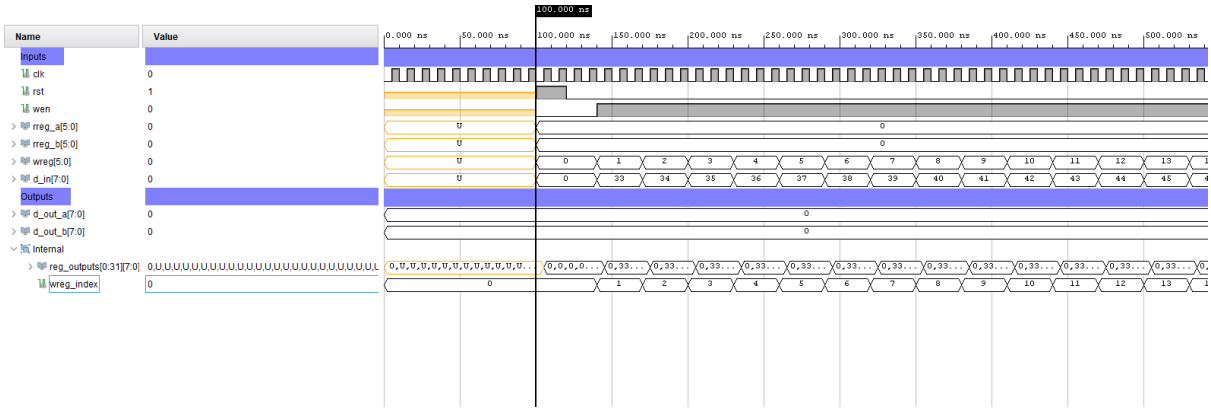


Figure 3

Read

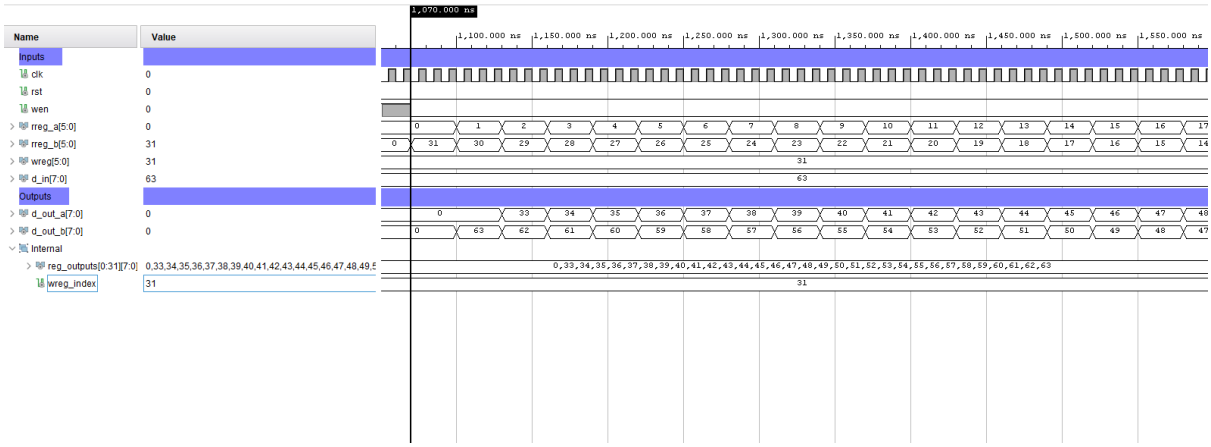


Figure 4

Read and Write

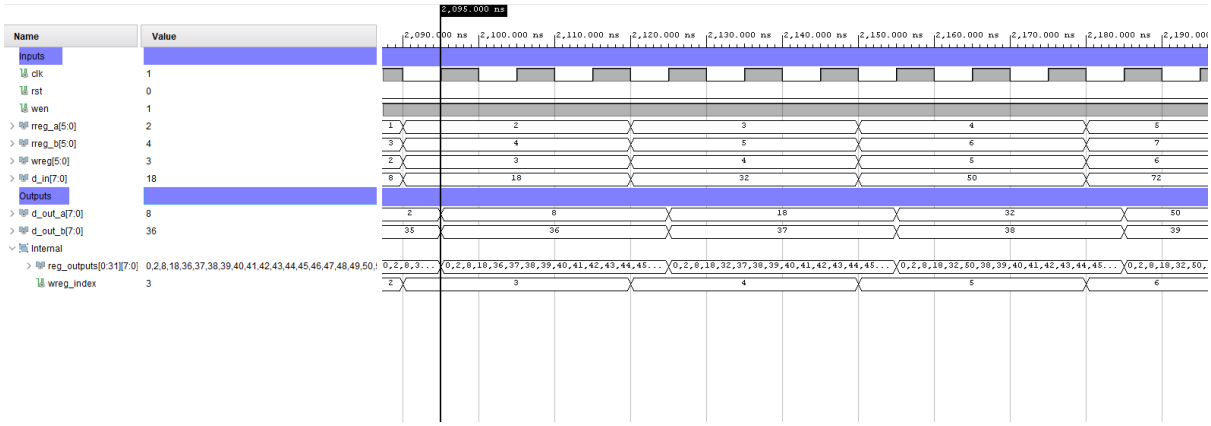


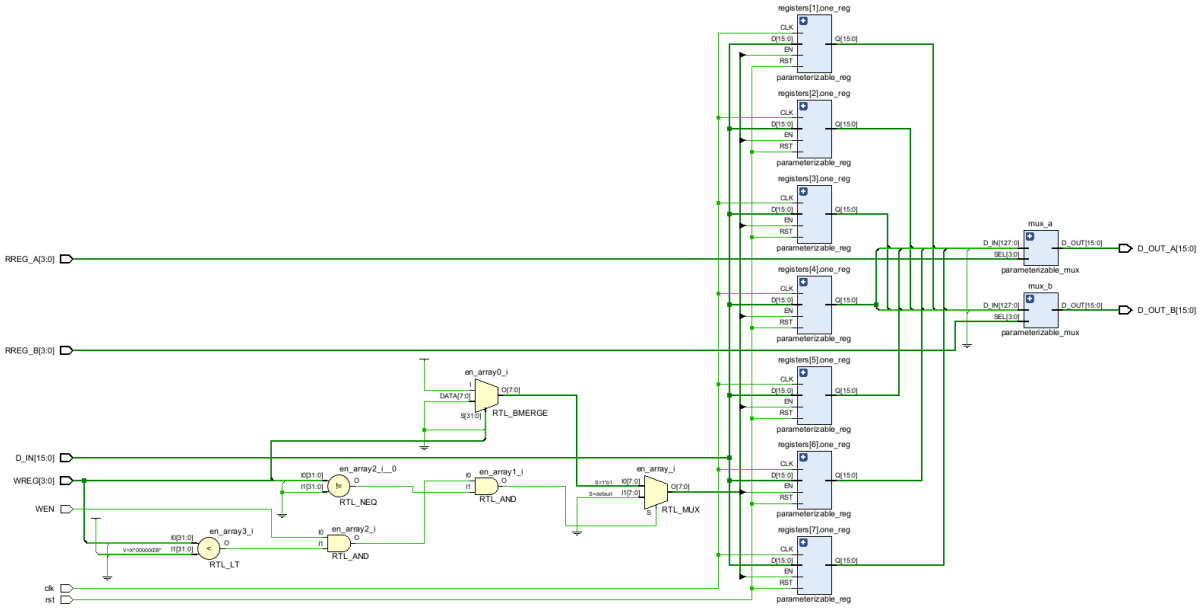
Figure 5

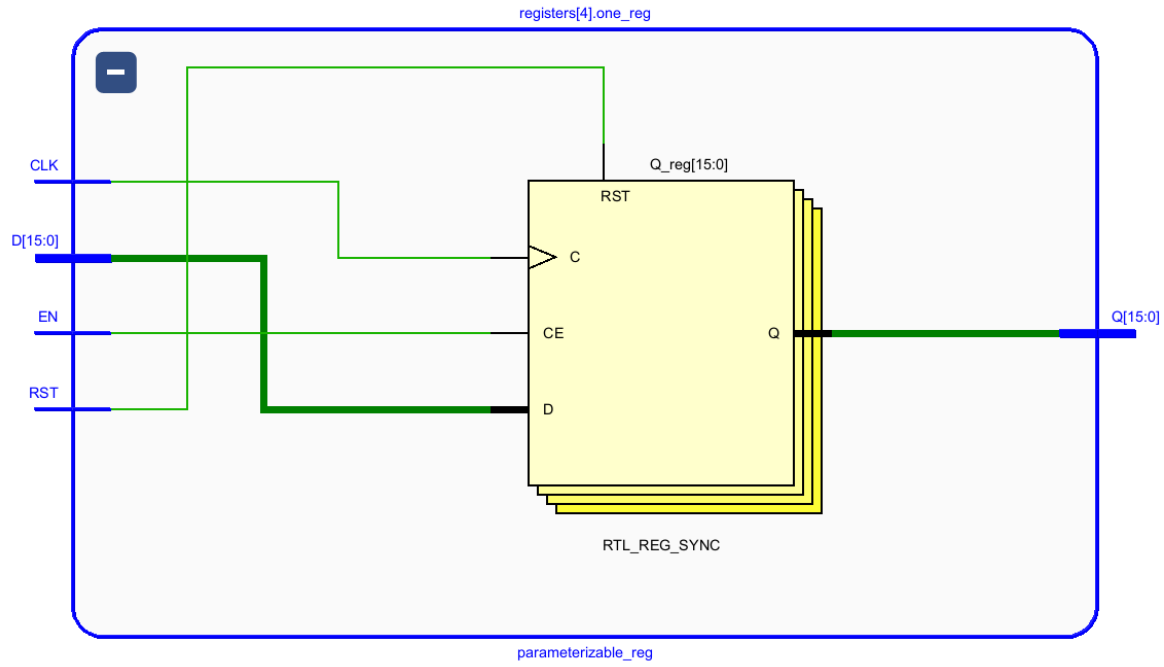
In Figures 3 (write), 4 (read), and 5 (read and write), the functionality of our register bank can be seen. When the WEN signal is high (such as in Figures 3 and 5), the value at D_IN is written to the register that WREG is pointing to. This can be seen clearer in Figure 5 as it is more zoomed in, and you can see the reg_outputs value update with the new value of 18 (D_IN) in the register 3 (WREG) slot. In Figure 4, it can be seen that when RREG_A or RREG_B holds a value, the value held at that number register in reg_outputs is output in the corresponding D_OUT_A or D_OUT_B (this can also be seen in Figure 5, but is clearer in Figure 4)

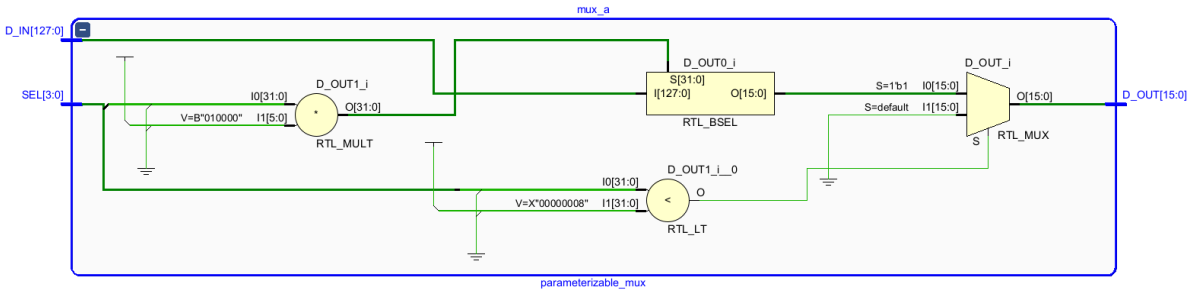
Synthesis Report – RTL Component Statistics

```
---------------------------------------------------------------------------------
Start RTL Component Statistics
---------------------------------------------------------------------------------
Detailed RTL Component Info :
+---Registers :
                  16 Bit    Registers := 7
+---Muxes :
       2 Input   16 Bit        Muxes := 2
---------------------------------------------------------------------------------
Finished RTL Component Statistics
---------------------------------------------------------------------------------
```

RTL Analysis Schematics







Single-cycle Processor Datapath

Table of outputs

State	OPCODE	SH	SEL	IMM	RREG_A	RREG_B	WEN	WREG	OUT_EN	RST_DP
IDLE	D	D	D	D	D	D	D	D	0	1
LOAD_R1	0000	D	1	DATA_IN*	D	D	1	001	0	0
WAIT_ST	D	D	D	D	D	D	D	D	0	0
LOAD_R2	0000	D	1	DATA_IN*	D	D	1	010	0	0
READ_R0	D	D	0	D	D	000	D	D	0	0
MASK_R6	1000	D	0	D	D	D	1	110	0	0
READ_R6	D	D	0	D	D	110	D	D	0	0
SET_R7	1100	0011	0	D	D	D	1	111	0	0
READ_R1_R6	D	D	0	D	001	110	D	D	0	0
AND_R1_R6	0100	D	0	D	D	D	1	D	0	0
READ_R3_R2	D	D	0	D	011	010	D	D	0	0
ADD_R3_R2	1010	D	0	D	D	D	1	011	0	0
READ_R1	D	D	0	D	D	001	D	D	0	0
SHIFT_R1	1101	0001	0	D	D	D	1	001	0	0
READ_R2	D	D	0	D	D	010	D	D	0	0
SHIFT_R2	1100	0001	0	D	D	D	1	010	0	0
READ_R7	D	D	0	D	D	111	D	D	0	0
UPDATE_R7	1001	D	0	D	D	D	1	111	0	0
LOAD_R3	0000	D	1	D	D	D	1	011	0	0
HOLD	D	D	D	D	D	D	D	D	1	0

D = ‘Don’t care’ values, which are set to 0 by default, and DATA_IN* is the DATA_IN input sign extended to 16 bits.

Simulation Screenshots

Overall Behaviour

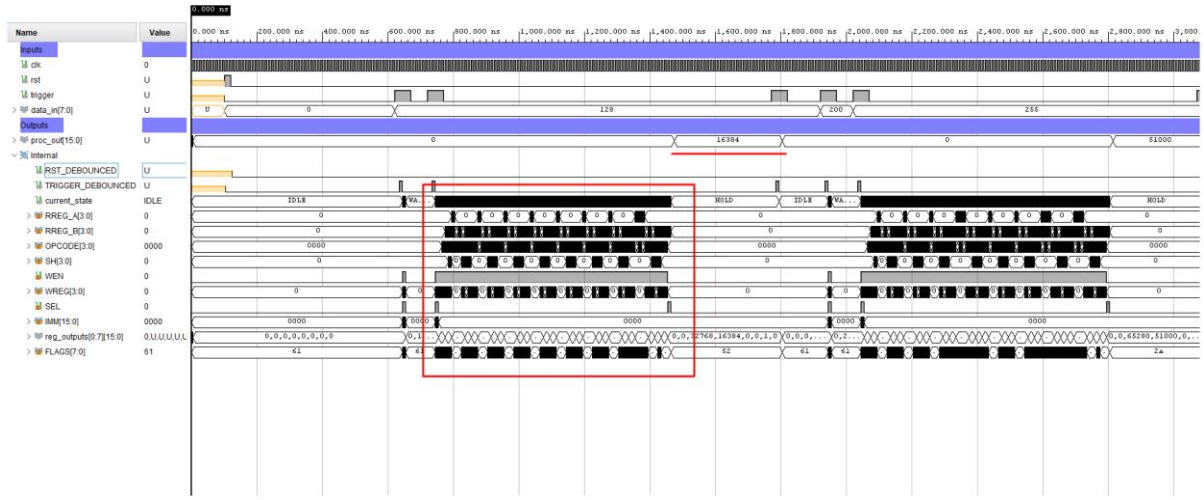


Figure 6

Figure 7

In Figures 6 and 7, the overall function of our processor can be verified. In Figure 6, a red line underlines a value from PROC_OUT, this value is the total of our inputs multiplied together. Given our inputs were 128 and 128, and the output is 16384, and then in Figure 7 our inputs are 255 and 128, and the output is 32640 – we can verify our processor works. Also in Figure 7, around 5000ns, a reset signal is sent as an input, this cancels everything and resets it back to 0 – verifying our reset button functionality. In Figure 6, there is a red rectangle which encloses the main function as to how these values are calculated, this will be explained in the following sets of photos.

Algorithm Setup

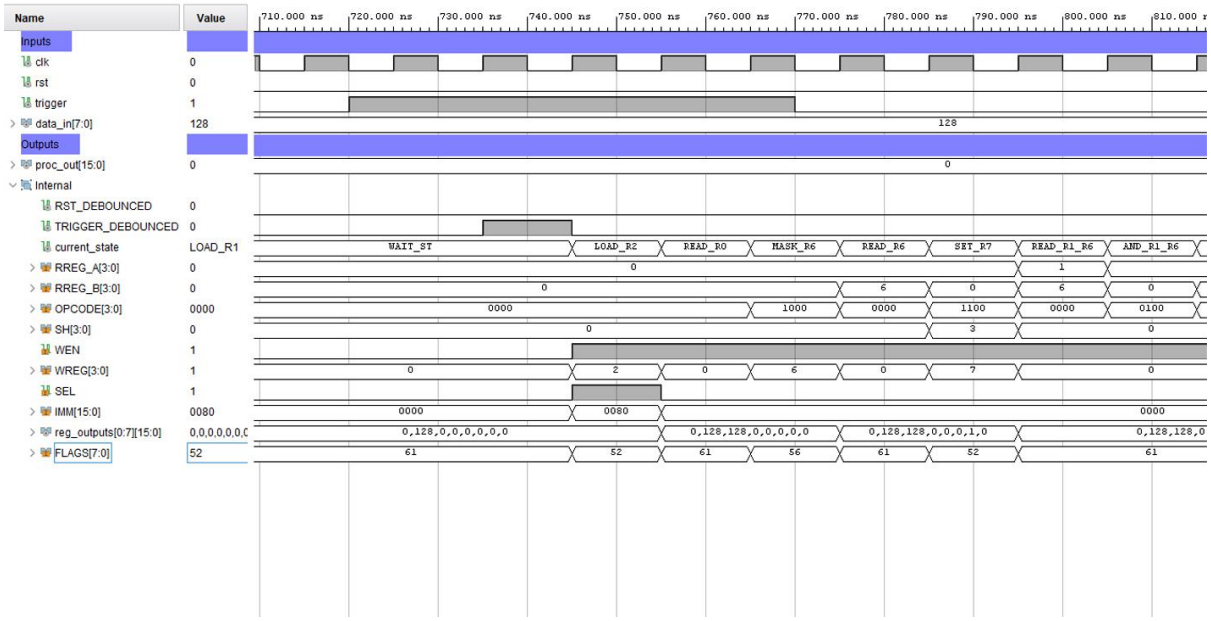


Figure 8

In Figure 8, the start of the algorithm can be seen (excluding the first load of R1 since this was unable to fit into a screenshot – however it functions the same as load R2). After R1 and R2 have been loaded with the data in value (which can be confirmed by looking at the reg_outputs values), the value of R0 is read, which should always be 0. This 0 value is then incremented using Opcode value 1000 and written to register 6 in the MASK_R6 state. Next, the value in R6, which we just set as 1, is read and left shift by 3 places using Opcode value 1100 and SH value 3, this value is then written to register 7. After this, the algorithm setup is complete and we begin to move into the main algorithm loop by reading the values of R1 and R6 and completing an AND operation on them by using Opcode value 0100 in order to decide which state to move to next.

Algorithm loop

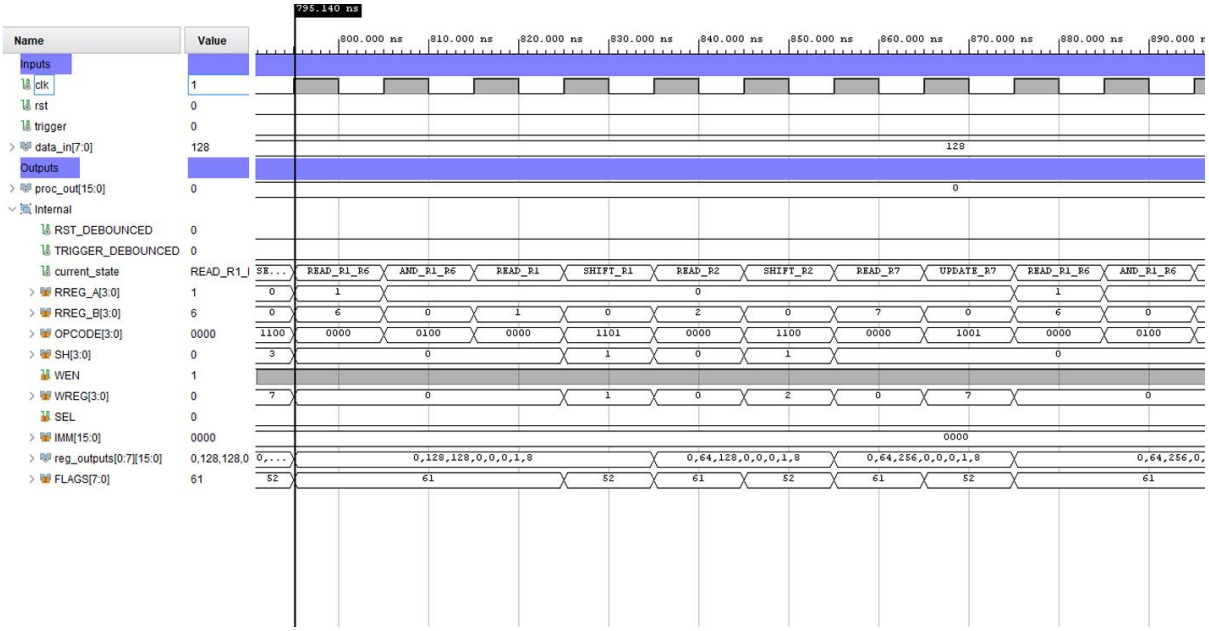


Figure 9

Now inside the main algorithm loop in Figure 9, we can see how things work. Once the R1 and R6 values have been 'ANDed', the value of FLAGS is checked to decide which state to move to. In figure 7, the next state was to read R1, R1 is then shifted to the right by 1 space which halves it, then R2 is shifted to the left by 1 space – doubling it. This can be again evidenced by looking at the reg_outputs values. After this, R7 is read, and decremented before flags are checked again to decide what to do. If the flags detect ALU_OUT is still >0, the loop starts again, this carries on until R7 decrements to 0 – this can be seen in Figure 10.

Algorithm end

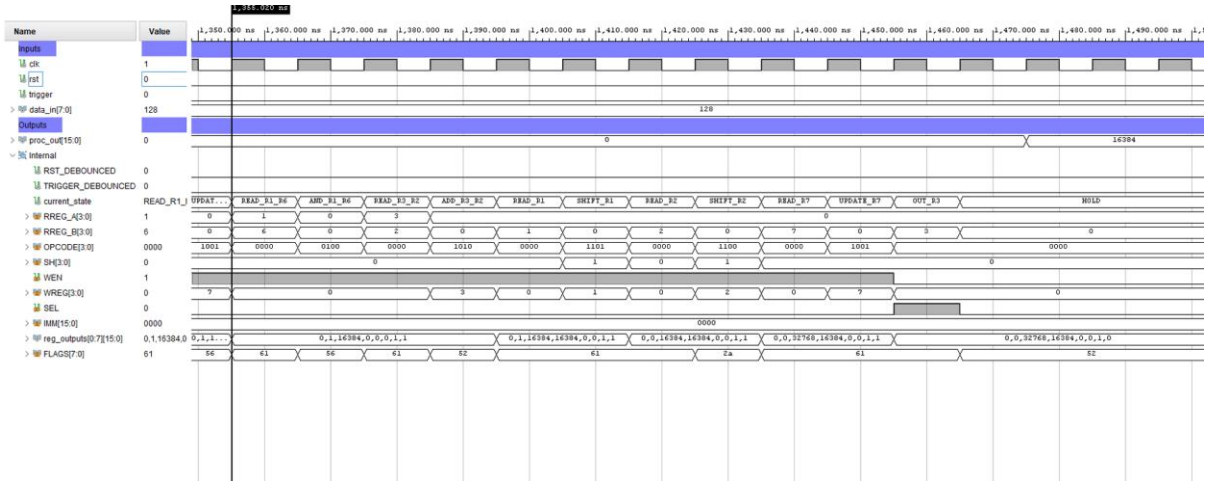


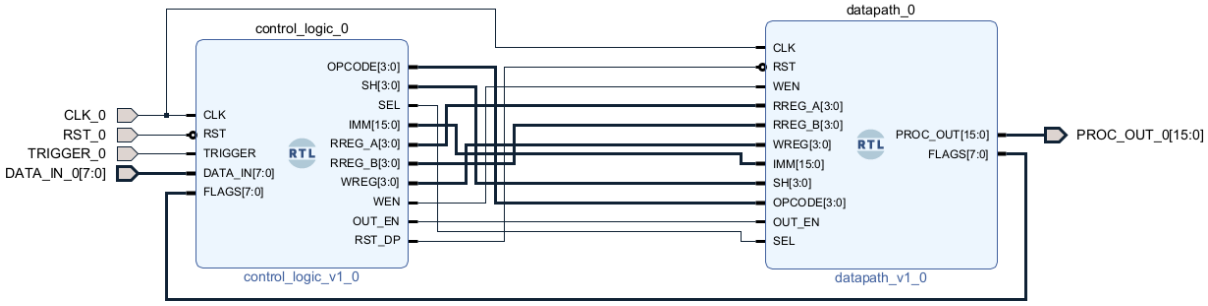
Figure 10

In Figure 10, we have a case of the algorithm coming to an end, everything functions the same up until the UPDATE_R7 state, because here the flag is different and it has detected that ALU_OUT = 0, this then moves into the OUT_R3 state, where the value which had accumulated in R3 is finally sent to PROC_OUT to be output, which we can see happen in Figure 10.

Synthesis Report – RTL Component Statistics

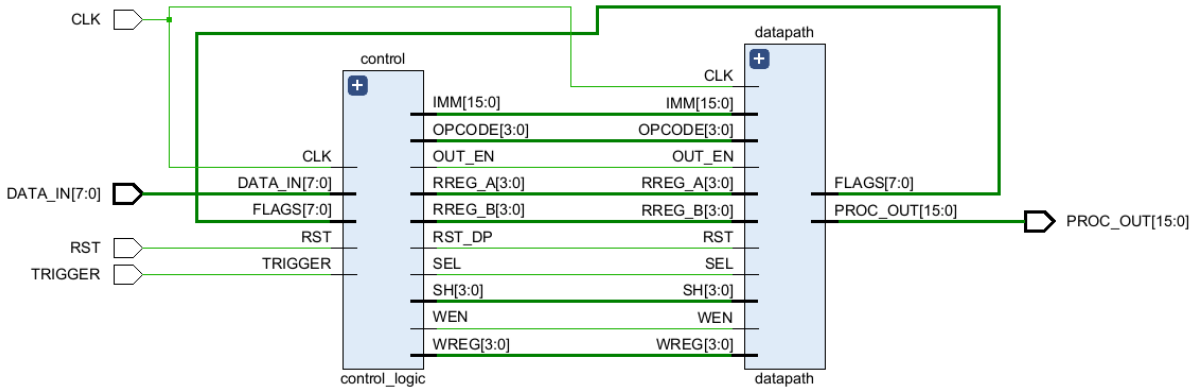
```
---------------------------------------------------------------------------------
Start RTL Component Statistics
---------------------------------------------------------------------------------
Detailed RTL Component Info :
+---Adders :
       2 Input   16 Bit       Adders := 1
       3 Input   16 Bit       Adders := 1
+---XORs :
       2 Input     16 Bit         XORs := 1
+---Registers :
                  16 Bit    Registers := 8
                   5 Bit    Registers := 1
                   1 Bit    Registers := 6
+---Muxes :
       2 Input   16 Bit        Muxes := 3
      14 Input   16 Bit        Muxes := 2
       2 Input    8 Bit        Muxes := 2
       2 Input    5 Bit        Muxes := 2
       3 Input    4 Bit        Muxes := 1
       2 Input    4 Bit        Muxes := 2
       5 Input    4 Bit        Muxes := 1
       2 Input    3 Bit        Muxes := 6
       3 Input    3 Bit        Muxes := 1
       2 Input    2 Bit        Muxes := 5
       3 Input    2 Bit        Muxes := 1
       2 Input    1 Bit        Muxes := 4
       4 Input    1 Bit        Muxes := 1
---------------------------------------------------------------------------------
Finished RTL Component Statistics
---------------------------------------------------------------------------------
```

Block design

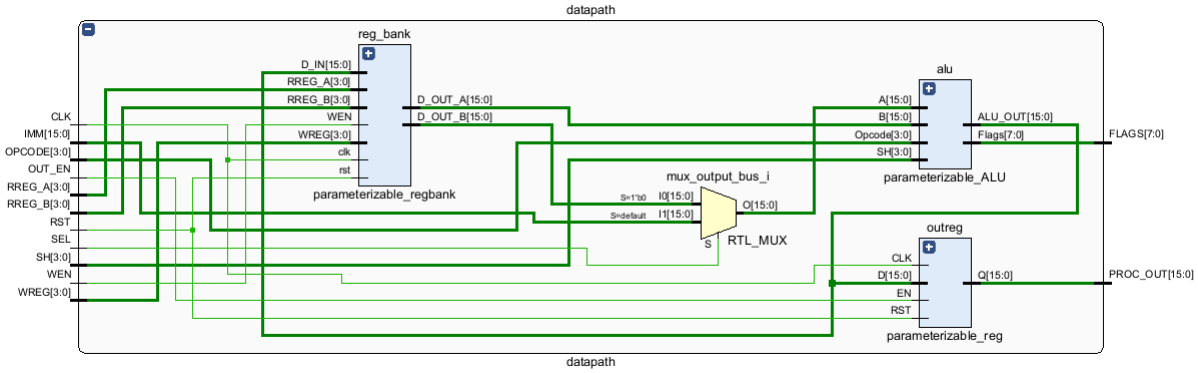


RTL Analysis Schematics

Processor



Datapath



Control Logic