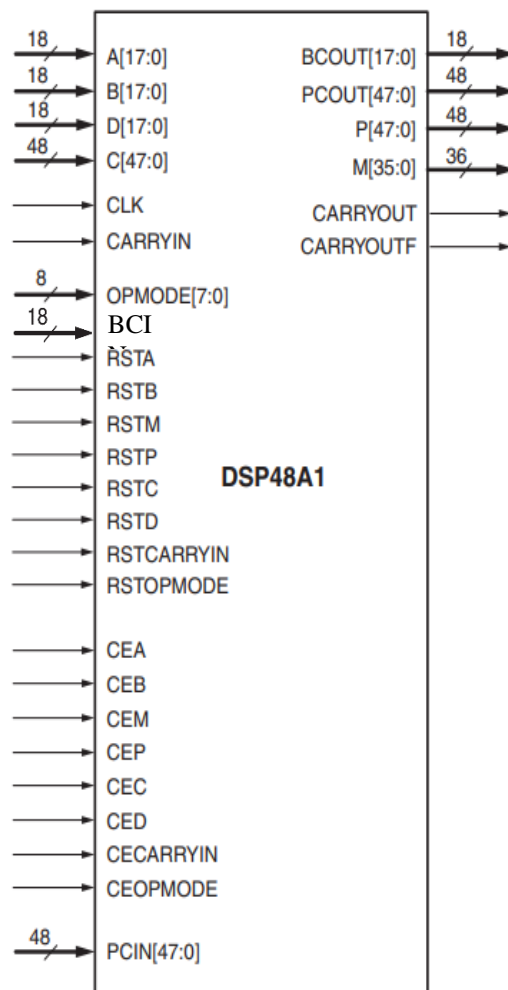




V15 Digital Design Diploma

Prepared by Magdy Ahmed & Revised by Kareem Waseem

DSP48A1 Testbench Stimulus



Testbench for DSP48A1 design will be as follows:

1. The parameters will remain unchanged, and the design will operate with the following default values:

- A0REG = 0
- A1REG = 1
- B0REG = 0
- B1REG = 1
- CREG = 1
- DREG = 1
- MREG = 1
- PREG = 1
- CARRYINREG = 1
- CARRYOUTREG = 1
- OPMODEREG = 1
- CARRYINSEL = "OPMODE5"
- B_INPUT = "DIRECT"
- RSTTYPE = "SYNC"

2. Stimulus Generation (Initial Block)

- 2.1. Verify Reset Operation

- Assert all active-high reset signals by setting them to 1.
- Drive remaining inputs with arbitrary (random) values.
- Wait for the negative edge of the clock.
- Add a condition for self-checking to verify that all outputs are zero.
- Deassert all reset signals and assert all clock enable signals to validate the functionality of the subsequent DSP paths.

- 2.2. Verify DSP Path 1

- This test path as shown in figure 1 evaluates the pre-subtractor, allowing it to propagate and post-subtractor stages by enabling the multiplier output to route through Mux X and the C-port through Mux Z, corresponding to OPMODE = 8'b11011101.
- Apply the following input values:
A = 20, B = 10, C = 350, and D = 25.
- Drive BCIN, PCIN, and CARRYIN with arbitrary (random) values.
- Wait for four negative clock edges, as the data propagates through four flip-flops (DREG, B1REG, MREG, and PREG), as illustrated in Figure (1).
- The expected outputs are: BCOUT = 'hf, M = 'h12c, P = PCOUT = 'h32, and CARRYOUT = CARRYOUTF = 0.
- Add a condition for self-checking to verify that the design outputs with the expected outputs.

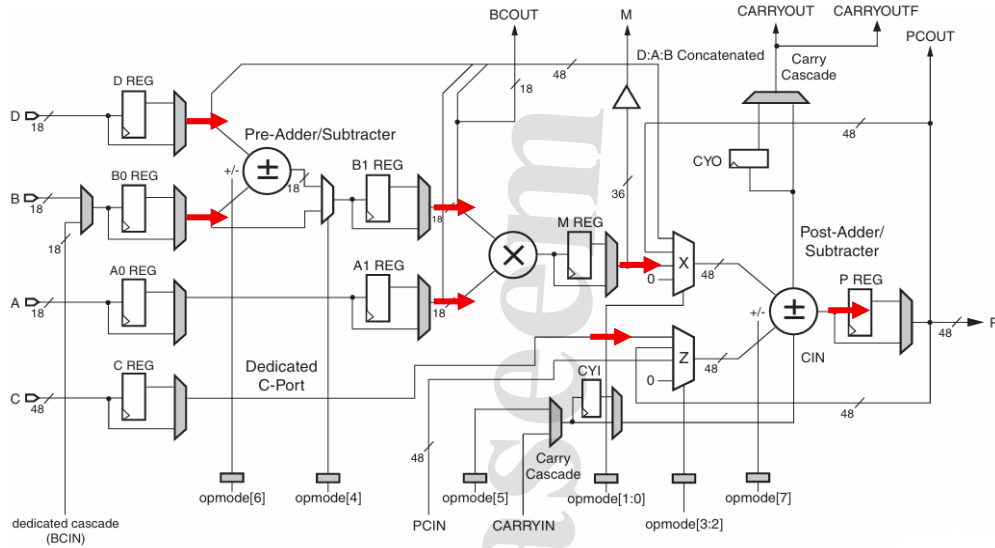


Figure 1: Path 1 Data Flow

2.3. Verify DSP Path 2

- This test path evaluates the pre-addition allowing it to propagate and post-addition stages by enabling the zeros to route through Mux X and Mux Z, corresponding to OPMODE = 8'b00010000.
- Apply the following input values:
A = 20, B = 10, C = 350, and D = 25.
- Drive BCIN, PCIN, and CARRYIN with arbitrary (random) values.
- Wait for three negative edges, as the data propagates through three flip-flops (DREG, B1REG, MREG) as longest path and also pass to PREG in parallel, as illustrated in Figure (2).
- The expected outputs are: BCOUT = 'h23, M = 'h2bc, P = PCOUT = 0, and CARRYOUT = CARRYOUTF = 0.
- Add a condition for self-checking to verify that the design outputs with the expected outputs.

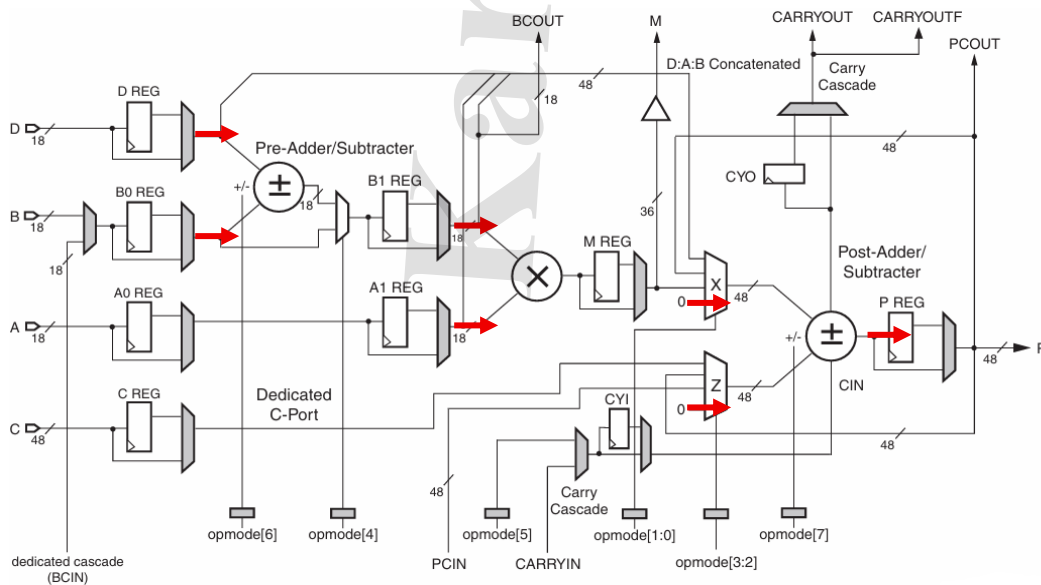


Figure 2: Path 2 Data Flow

2.4. Verify DSP Path 3

- This test path has no pre-addition/subtractor (doesn't allow it to propagate) and allows post-addition stages by enabling the P feedback to route through Mux X and Mux Z, corresponding to `OPMODE = 8'b00001010`.
- Apply the following input values:
A = 20, B = 10, C = 350, and D = 25.
- Drive BCIN, PCIN, and CARRYIN with arbitrary (random) values.
- Wait for three negative edges, as the data propagates through three flip-flops (B1REG, MREG, PREG) as longest path and also pass to PREG in parallel, as illustrated in Figure (3).
- The expected outputs are: BCOUT = 'ha, M = 'hc8, P = PCOUT which is the past Value of P, and CARRYOUT = CARRYOUTF which is the past value of CARRYOUT.
- Add a condition for self-checking to verify that the design outputs with the expected outputs.

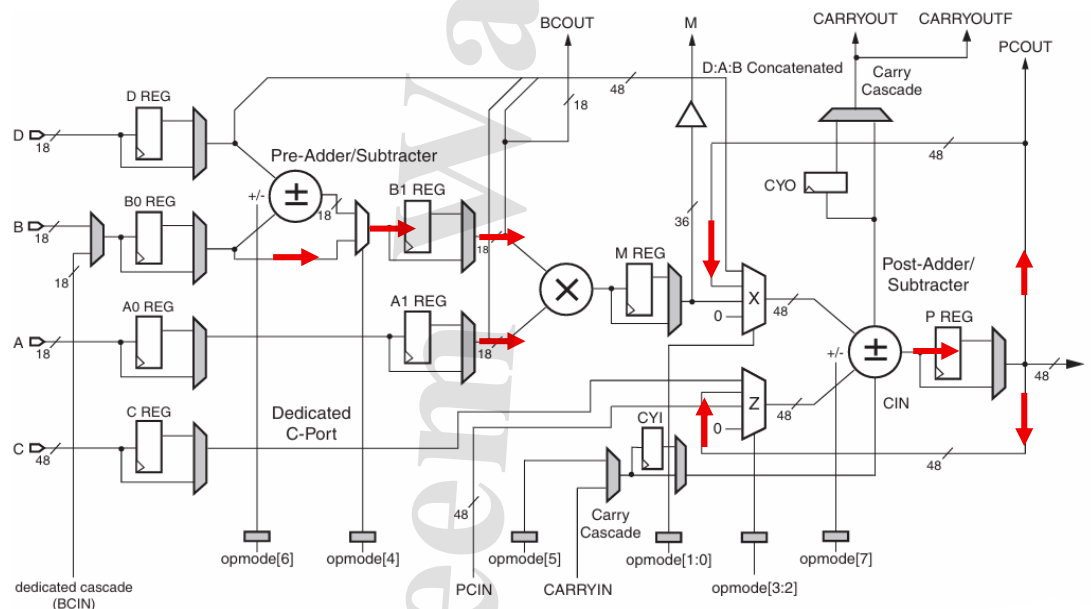


Figure 3: Path 3 Data Flow

2.5. Verify DSP Path 4

- This test path has no pre-addition/subtractor (doesn't allow it to propagate) and allows post-subtractor stages by enabling D:A:BC Concatenated to route through Mux X and PCIN to Mux Z, corresponding to OPMODE = 8'b10100111.
- Apply the following input values:
A = 5, B = 6, C = 350, D = 25 and PCIN = 3000
- Drive BCIN, and CARRYIN with arbitrary (random) values.
- Wait for three negative edges, as the data propagates through three flip-flops (B1REG, MREG, PREG) as longest path as illustrated in Figure (4).
- The expected outputs are: BCOUT = 'h6, M = 'h1e, P = PCOUT = 'hfe6ffffec0bb1, and CARRYOUT = CARRYOUTF = 1.

- Add a condition for self-checking to verify that the design outputs with the expected outputs.

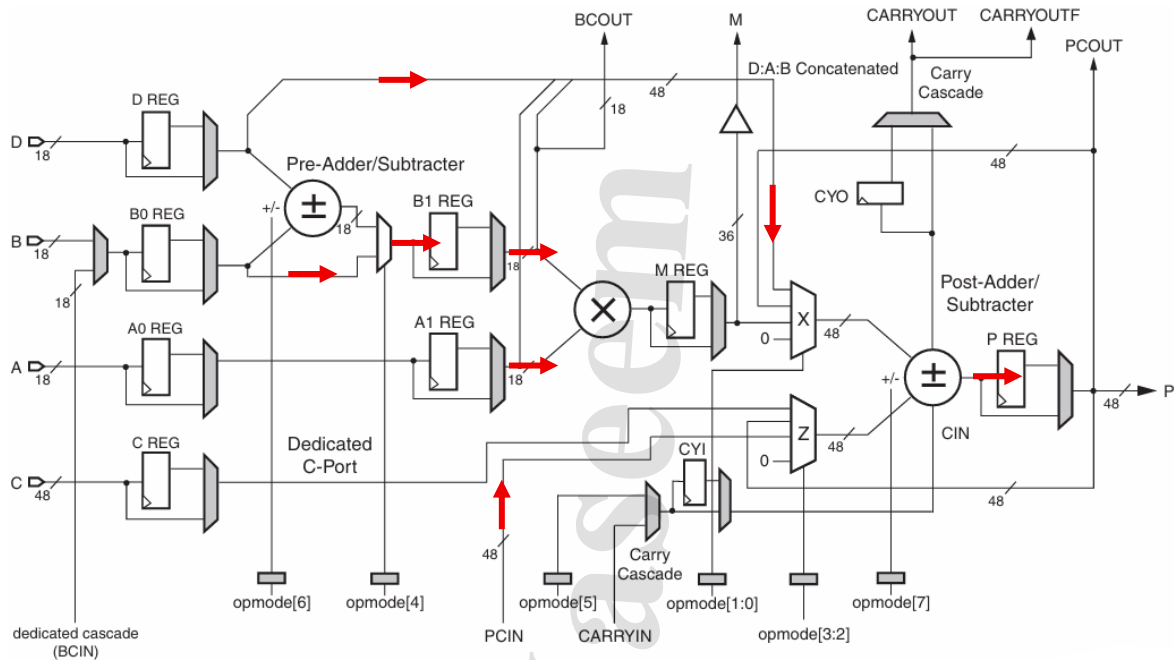


Figure 4: Path 4 Data Flow

