**Digital Design and Computer Organisation Laboratory**

**3rd Semester, Academic Year 2025**

Date:25-08-2025

|  |  |  |
| --- | --- | --- |
| Name: Aakash Desai | SRN:PES1UG24CS006 | Section  3A |

Week Number: 3 Program Number: 1

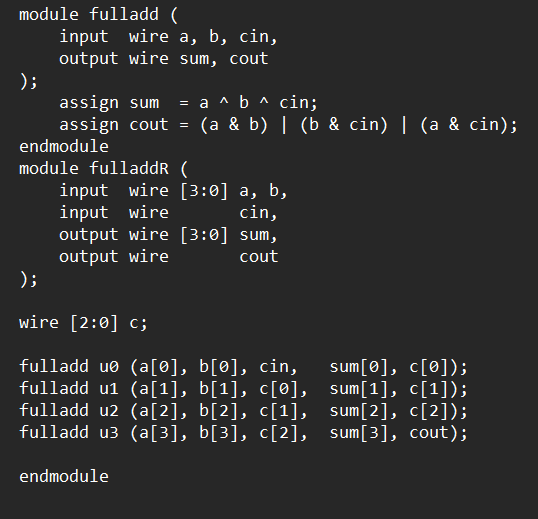
TITLE:Write a verilog program to model a 4 bit ripple carry adder. Generate the VVP output and simulation waveform using GTKWave.

Deliverables

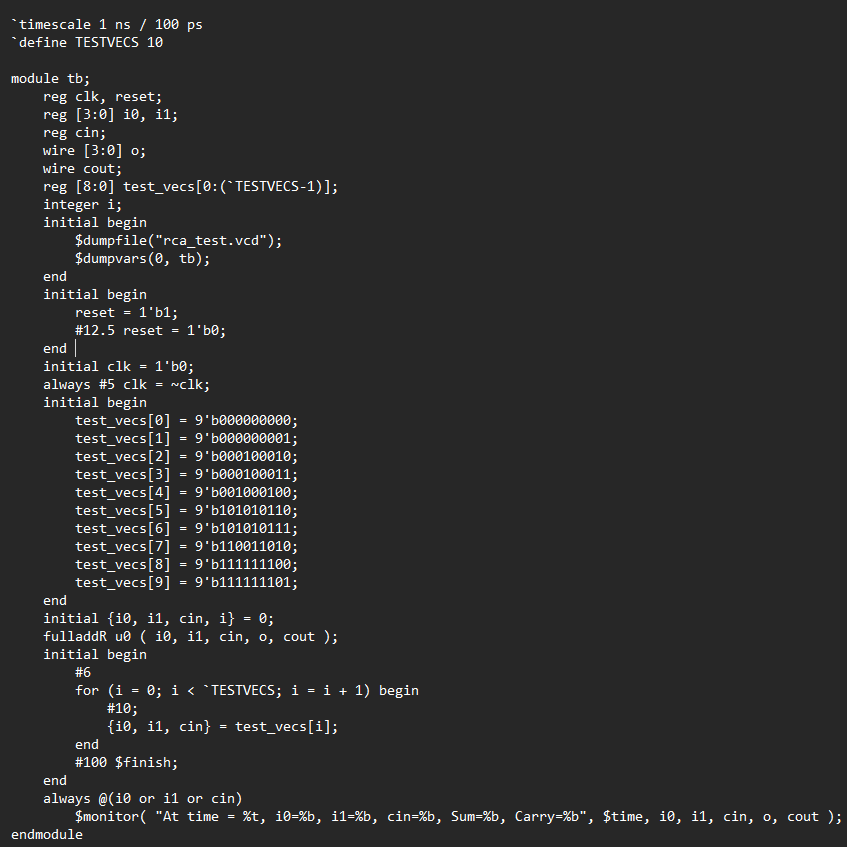
1. Verilog Code Screenshot
2. Verilog VVP Output Screen Shot
3. GTKWAVE Screenshot
4. Output Table to be completed and included

I.Verilog Code Snippet

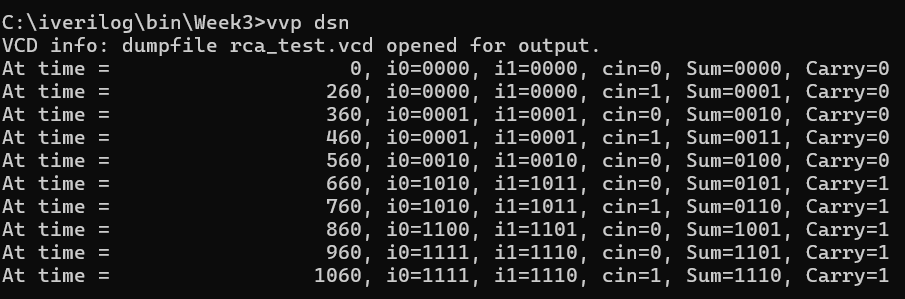
* Main code

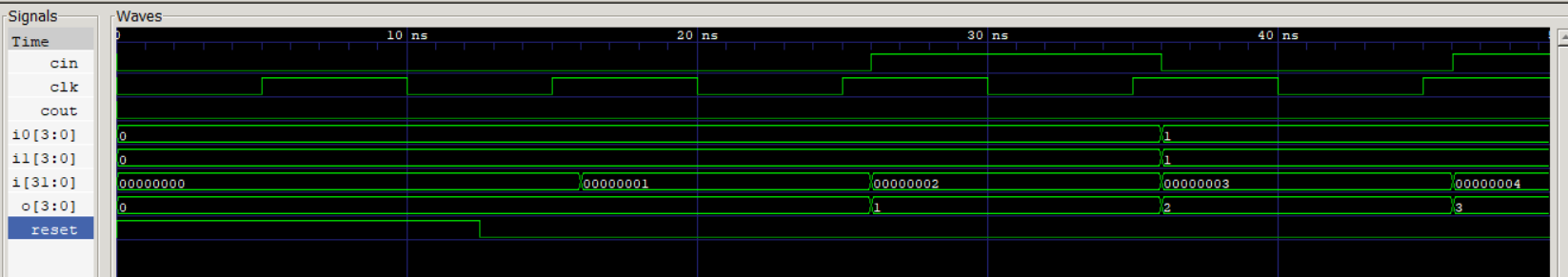


* Test Bench



II Verilog VVP output ScreenShot



III GTKWave ScrrenShot

IV Truth Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A | B | Cin | SUM | CARRY |
| 0000 | 0000 | 0 | 0000 | 0 |
| 0000 | 0000 | 1 | 0001 | 0 |
| 0001 | 0001 | 0 | 0010 | 0 |
| 0001 | 0001 | 1 | 0011 | 0 |
| 0010 | 0010 | 0 | 0100 | 0 |
| 1010 | 1011 | 0 | 0101 | 1 |
| 1010 | 1011 | 1 | 0110 | 1 |
| 1100 | 1101 | 0 | 1001 | 1 |
| 1111 | 1110 | 0 | 1101 | 1 |
| 1111 | 1110 | 1 | 1110 | 1 |

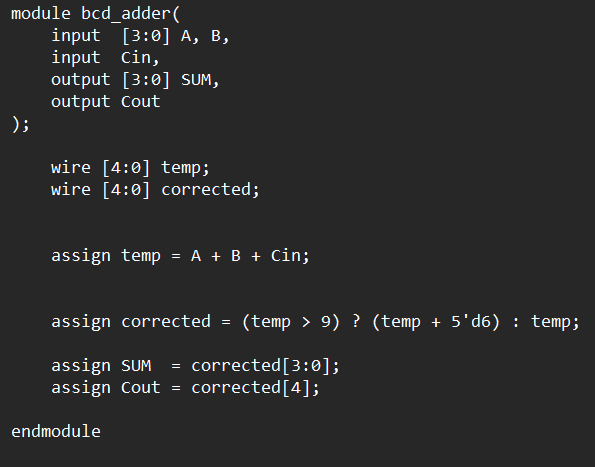
TITLE:Design a digital system that adds two decimal digits represented in BCD format and output results in BCD. Explain the need for correction in result and implement in verilog.What should be verified in simulation?

Deliverables

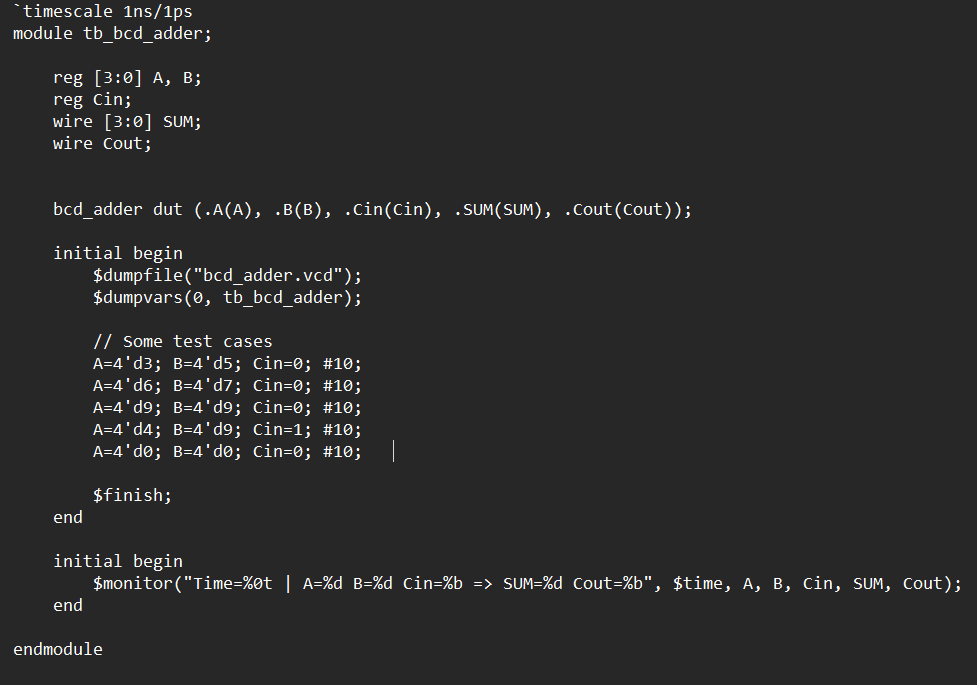
1. Verilog Code Screenshot
2. Verilog VVP Output Screen Shot
3. GTKWAVE Screenshot
4. Output Table to be completed and included
5. THE BCD Circuit

I.Verilog Code Snippet

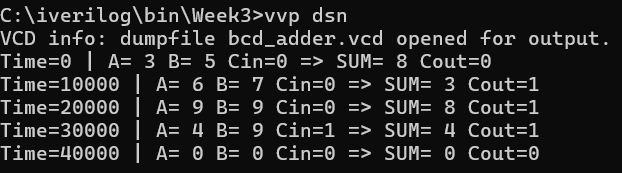
* Main Code



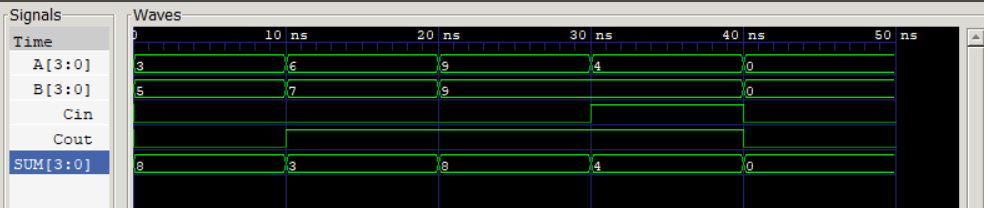
* TestBench



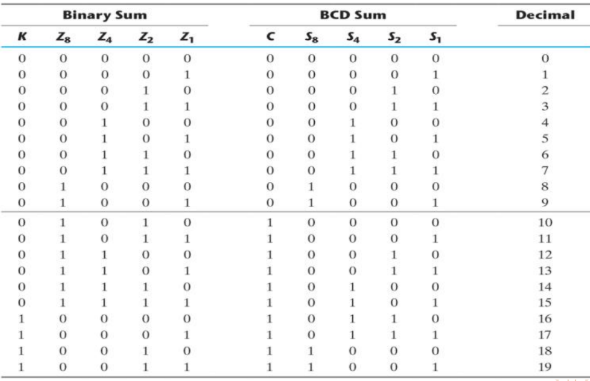
II Verilog VVP ScreenShot



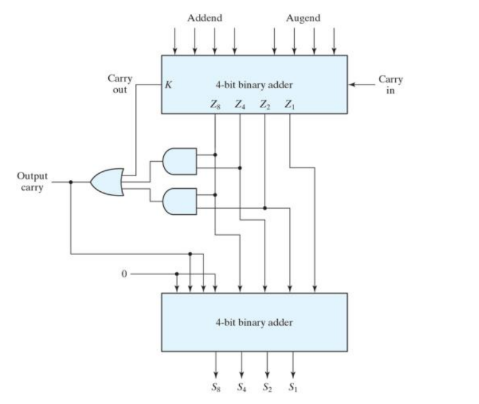
III GTK wave Screenshot



IV Truth Table



V BCD Adder Circuit



**Why Correction is Needed**

When adding two BCD digits (and possibly a carry-in), the sum is initially generated as a 4-bit binary number. However, this 4-bit result can sometimes exceed 1001 (decimal 9), which is not a valid BCD digit.

For example:

1. Adding 7 (0111) + 8 (1000) = 1111 (decimal 15).
2. But 1111 is not a valid BCD representation, since valid BCD digits stop at 1001.

To correct this situation, a correction factor of +6 (0110) is added.

* Because the binary system allows up to 15 (1111), but BCD allows only up to 9.
* The difference between 1010 (decimal 10) and 1111 (decimal 15) is 6.
* Adding 0110 ensures the result "wraps around" to a valid BCD digit and generates a carry for the tens place.

A correction is applied when:

1. The 4-bit sum is greater than 9 (>1001), OR
2. There is a carry out from the addition.

**Example**

1. 6 (0110) + 7 (0111) = 1101 (13) → invalid BCD
2. Apply correction: 1101 + 0110 = 1 0011
3. Final result: 0011 (3) with carry 1, which represents decimal 13 correctly.

**What Should be Verified in Simulation?**

1. Correct Sum Generation: Verify that the adder correctly adds two BCD digits with/without carry-in.
2. Need for Correction: Check that whenever the binary sum exceeds 9 (1001) or there is a carry, the circuit adds 0110 (6) to produce a valid BCD result.
3. Valid BCD Output: Ensure the output sum is always in the range 0000–1001.
4. Carry-out Handling: Verify that a carry is generated when the result ≥ 10, and the tens digit is represented correctly.
5. Edge Cases: Confirm correct results for boundary conditions (e.g., 0+0, 9+9, 8+7+1 with carry-in).