Verilog Assignment 4
Tanishq Prasad (21CS30054)
Krish Khimasia (21CS10037)
Group No. 17

Question 1:

We are given the task to take an input x which is of 32 bits (4 bytes), and compute the value of y = x/255, without using any multiplier or division unit. The circuit would comprise of only an 8-bit subtractor and shifters as the main data-path elements.

We are using the approach that follows the given hint that is y = 256y - x.

Now lets break down 32-bit y as $y_3\, y_2\, y_1\, y_0$ each 8 bits long and similarly for x as $x_3\, x_2\, x_1\, x_0$
Implementing 256y as an 8-bit left shift of y we can see that

y		y_3	y_2	y_1	y_0
256y	y_3	y_2	y_1	y_0	z
x		x_3	x_2	x_1	x_0
(-)					
		y_3	y_2	y_1	y_0

The logic is that when we shift y by 8 bits to the left then we get the least significant 8 bits to be zero, those are represented by z in the schematic and then using the approach we can just get y_0 to be $z\text{-}x_0$ and then use the borrow out if any for the calculation of y_1 using $y_0\text{-}x_1$-borrow of the previous stage and continue this until we calculate the value for y_3.

Point to note : Using this logic we would give correct outputs for the multiples of 255 and if we don't give a multiple the output might not be as desired as then doing a bit shift might just lead to loss in data that could have been used otherwise.

To implement this we have designed a finite state machine whose working is as follows :

- S0: The start state. Goes to s1 when rst1 is set.
- S1: Reads in the x value into the X register.
- S2: Initialises the x3,x2,x1,x0 to 0,0,X[15:8],X[7:0] and y0,y1,y2,y3 bytes to 0,0,0,0 respectively.

- S3: Assigns the difference of zero and x0 to y0.
- S4: Assigns the difference of y0 and x1 to y1.
- S5: Assigns the difference of y1 and x2 to y2.
- S6: Assigns the difference of y2 and x3 to y3.
- S7: Writes {y1,y0} to y. If rst2 is set, we go back to the starting state.

The circuit diagram for the Divby255 module is given here :-