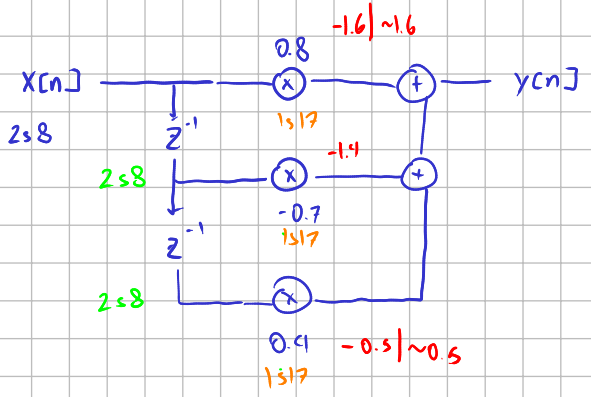
Consider the following filter:

3-tap FIR filter: h[n] = [0.8 -0.7 0.4]



let x be a 2s8 number, then find the signal formals for the circuit

Need to analyze not from a sinusoidal perspective but the absolute worse case scenario since that is more realisitic

If x[n] is 2s8 number, x[n] can be from -2:2-(2^-8)

we can represent the coeffiencients in the multiplier as a 1s17 number since typically we will be dealing with 18 bit numbers and 1s17 can represent numbers from -1 to ~1

we know that the output of the multiplier will be a 3s25, then we would add these numbers to get our output.

Thus to capture y[n] accurately, we may need a 5s25 signal when summing these values

But thats pessimistic and instead we can determine the required signal format based on analyzing the worst case output.

Assuming the worst case input is -2, and we know that we are multiplying by 0.8, the largest value of this multiplier would be -1.6 or ~1.6

again we repeat this process for the other multipliers

What is the largest possible output of this filter, given these coefficient values?

We can find the largest value by assuming that all 3 multipliers simultaneously gave us either the largest negative or positive value

worse positive case for y[n]: 1.6 + 1.4 + 0.8 = 3.8

worse negative case for y[n]: -1.6 -1.4 - .8 = -3.8

Thus the worse case value for y is either 3.8 or -3.8

Based on the worse case value, we can determine the 18 bit signal format that can handle the worse case output. Thus +/-3.8 can fit in a 3s15 signal format

We are asked to scale the filter coefficients so that the output can always fit a 1s17 signal format. Now that we have found that worse case output, we can scale the coefficients so that we get a 1s17

We are also asked to identify the worse case sequence of input values and set up a testbench that generates the worst case sequence to test our filter.

So what is the worst case sequence of input values that would generate the worse case value for the above filter?

From the above example, we would need to generate the worse case input into the multipliers simultaneously. For the worse case positive output, we would need to generate a 2 for the first multiplier, -2 for the 2nd , and 2 for the third.

This sequence is unlikely to happened BUT given that the inputs are randomized and enough time has passed, we may run into this worst case