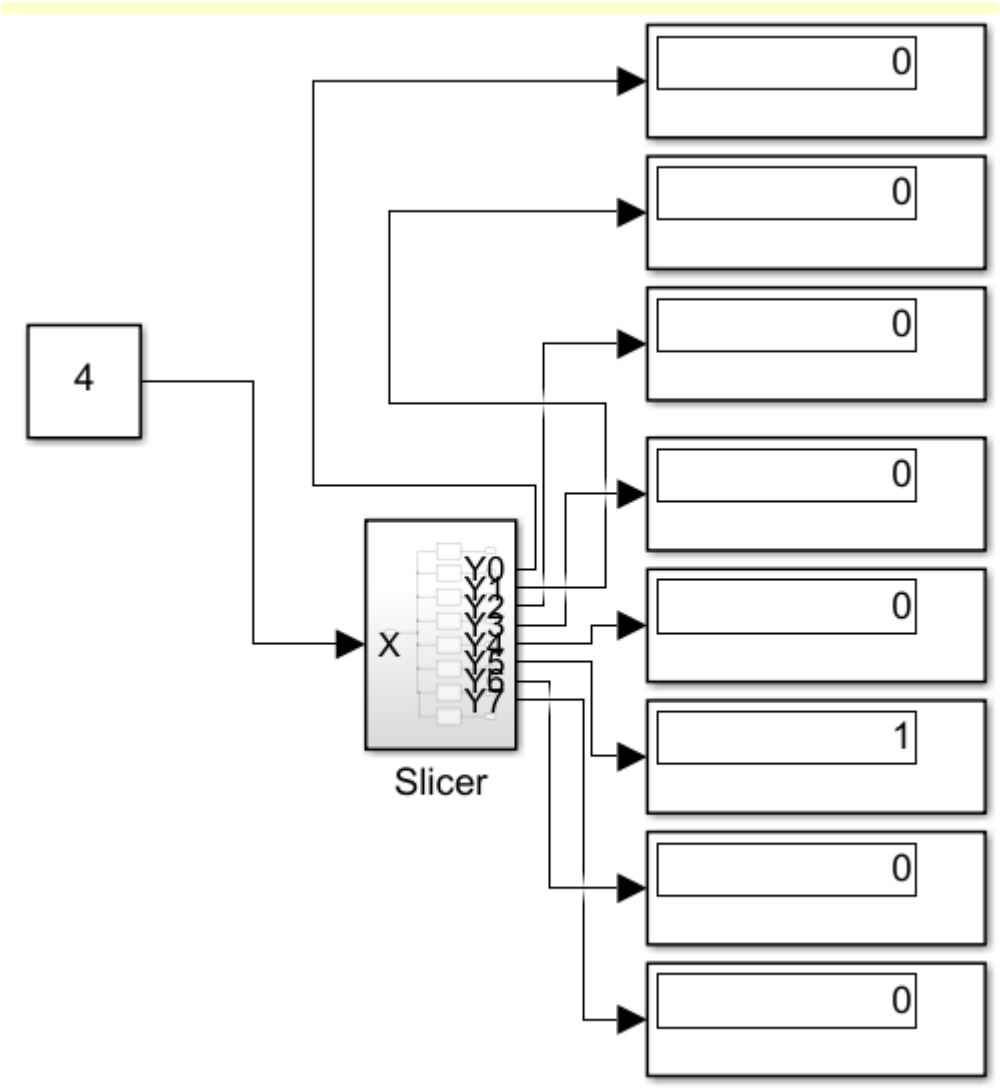
2160 Lab 7 Report

Patrick Taylor and James Wachala

Section Number: 1

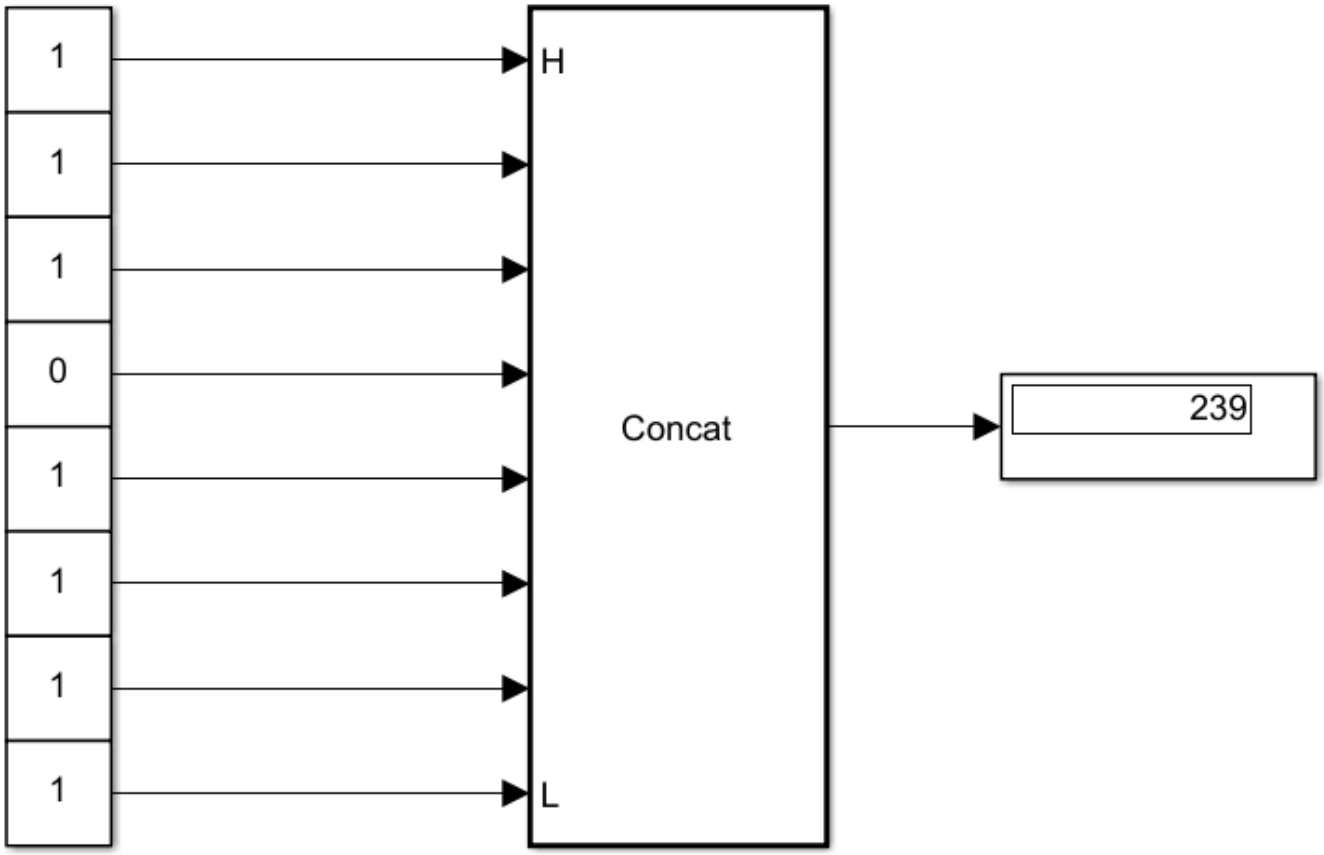
March 13th, 2019

Assignment 1



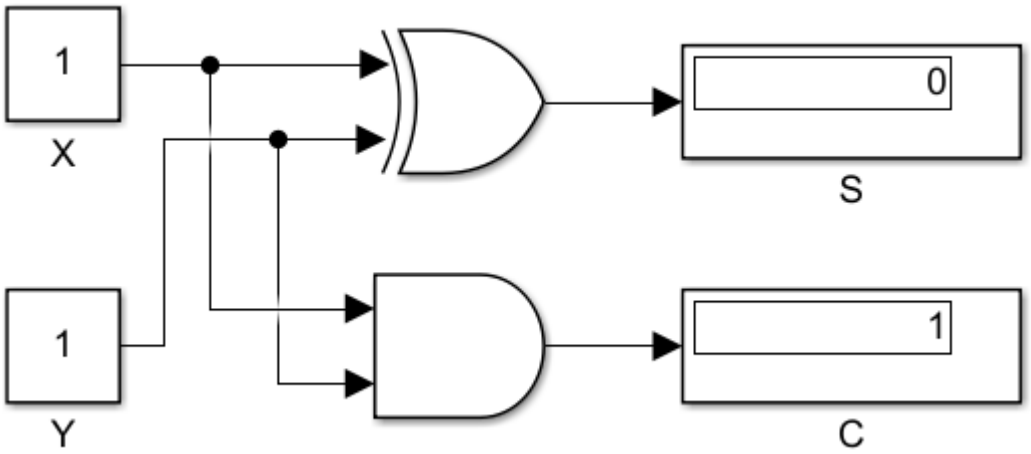
The output displayed here is meant to be a binary representation of the number put into the splicer, split into individual bits. This output is correct, as the binary form of 4 is 100. So if you read from top to bottom, this is that result.

Assignment 2



Displayed here is the opposite behavior of the above, this takes 8 bits and returns a number in decimal form. Each bit is given place value and concatenated to create a binary number, which the output displays in decimal. The displayed is correct as 11101111 in base two is 239 in base ten.

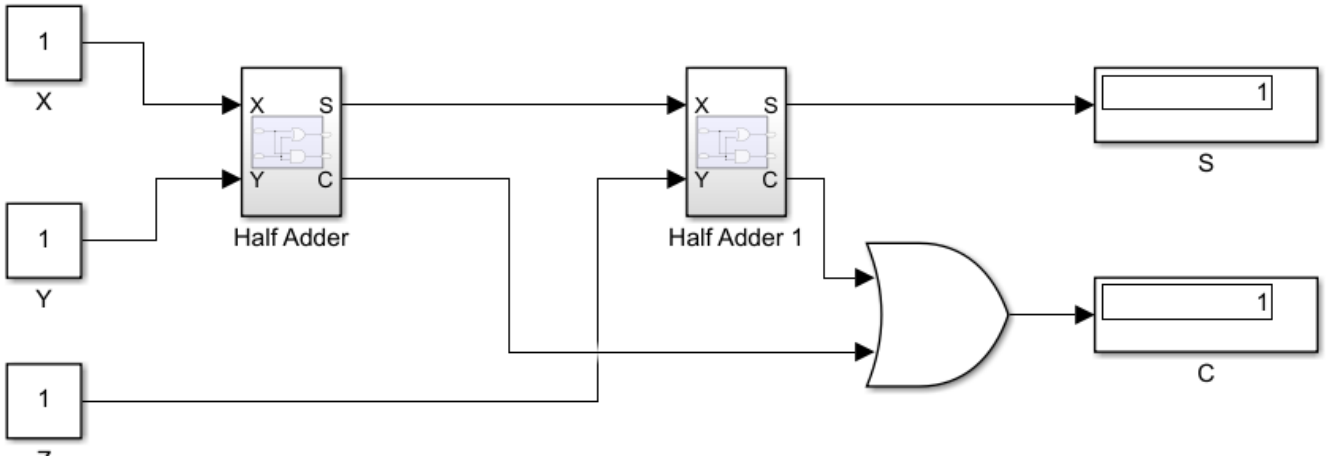
Assignment 3



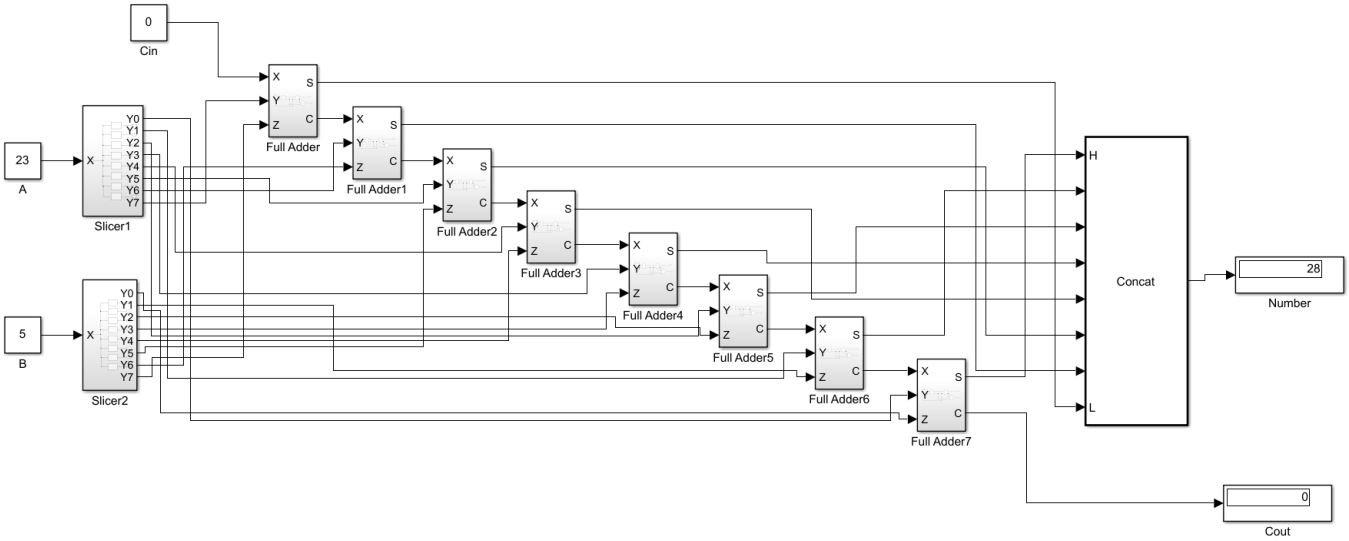
As s represents the least significant digit and c is the most significant (and carry) digit, the half adder can add 2 bits, as it takes 2 bits as input, and therefore, can have up to two bits as an output. In three cases, the sum is one bit, so is represnted entirely in s. (0 or 1). However, when the sum is two, one bit is not enough. Instead, the two is represented in c, carrying higher place value. In this case, c should be one and s should be zero. To create this behavior, an XOR gate does the job for x and an AND gate does the job for c.

x	y	s	c
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

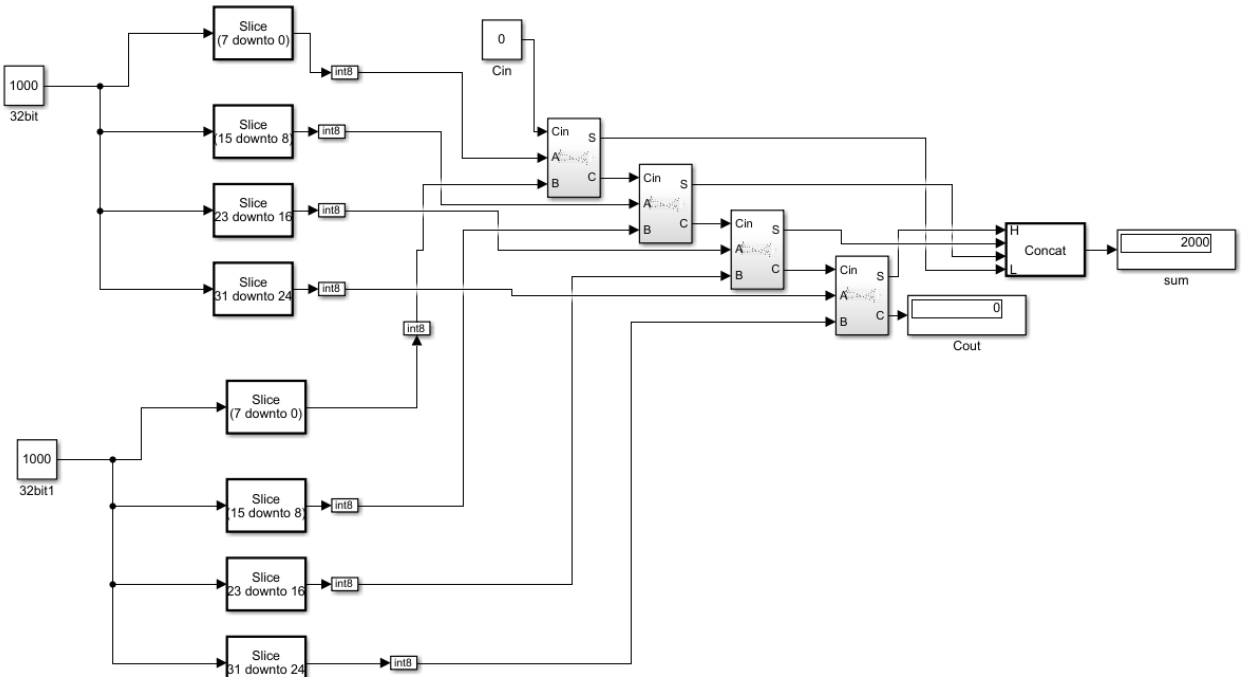
Assignment 4



Assignment 5



Extra Credit


1000
32bit
Slice
(7 downto 0)
Slice
15 downto 8
Slice
23 downto 16
Slice
31 downto 24
int8
0
Cin
Cin
A
B
S
C
Concat
H
L
2000
sum
0
Cout
1000
32bit1