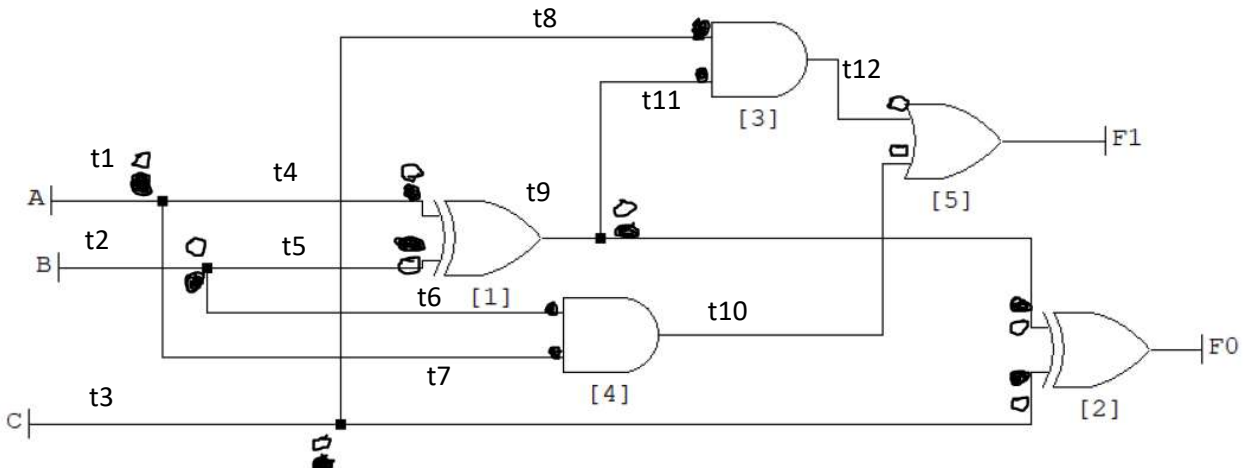
Justin Elassal
Tylor Cooks
Jorge Aguilar
11/29/20

CECS 361 – Lab 5

Stuck at Fault Detection

There are 12 possible locations for a stuck-at-one fault. These locations are labelled below with filled in circles:



From Simulator: SA1_tb.v					
A	**B**	**C**	**F1**	**F0**	
0	0	0	0	0	
0	**0**	**1**	**1**	1	***
0	1	0	0	1	
0	1	1	1	0	
1	0	0	0	1	
1	0	1	1	0	
1	1	0	1	0	
1	1	1	1	1	
gate one -> A^B = 0					
gate two -> (A^B)^C = 0					
gate three -> (A^B).C = 0					
gate four -> A.B = 0					
gate five -> ((A^B).C)+(A.B) = 0					
Use these equations to find stuck-at-one faults. They are included in the truth table below:					
Figure One Truth Table:					

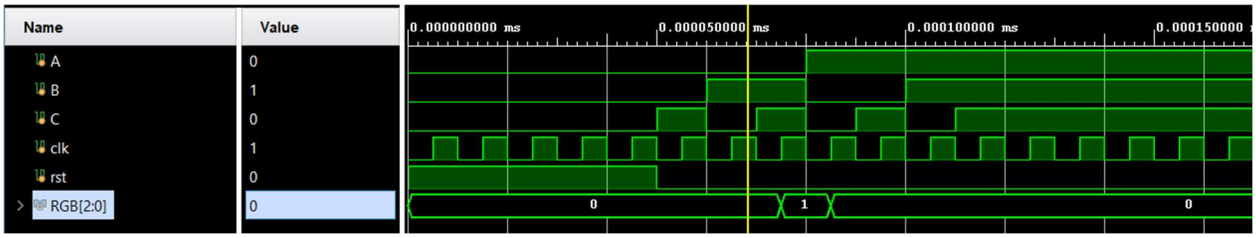
A	**B**	**C**	**one**	**two (F0)**	**three**	**four**	**five (F1)**	
0	0	0	0	0	0	0	0	
0	**0**	**1**	0	1	0	0	**0**	***inconsistent with simulator... fault detected here.
0	1	0	1	1	0	0	0	
0	1	1	1	0	1	0	1	
1	0	0	1	1	0	0	0	
1	0	1	1	0	1	0	1	
1	1	0	0	0	0	1	1	
1	1	1	0	1	0	1	1	

Stuck-at fault detected by F1 where {A,B,C} = 001

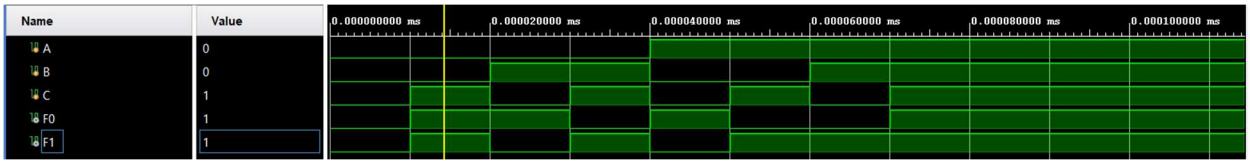
The stuck-at-one fault is detected with an inconsistency on gate five (F1) when the inputs are {A,B,C} = 001. We look for the AND-gate that leads to gate five. This is where our stuck-at-one fault is coming from (not necessarily where it originates). This happens to be on gate three. Since the detection occurs when C = 1, our focus now turns to the XOR-gate. The two inputs are A and B, and the output splits two ways. Our fault has to originate from one of these three locations. When C is zero, all combinations of A and B are consistent with the figure one truth table. This means that the origin of the stuck-at-one fault stems from where the output of gate one (XOR) splits. By looking at the test cases shown on the waveform for SA2 below, it can be seen that the issue lies in t11 (test case 11).

RGB onboard

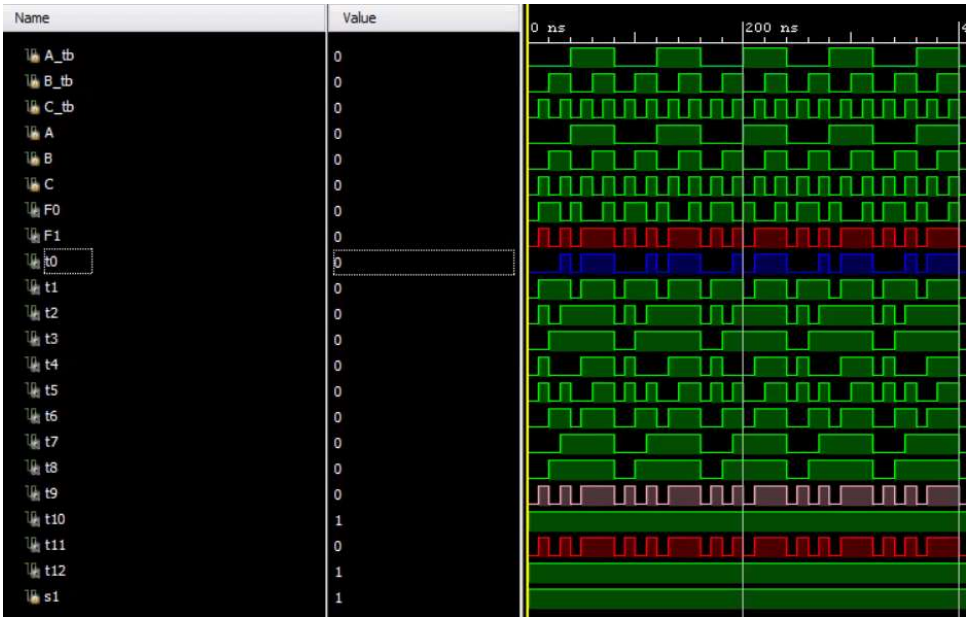
top waveform



SA1 waveform



SA2 test cases



ticker waveform