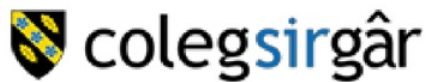


assignment 2 - james stallard

by Frederick Stallard

FILE	8702_FREDERICK_STALLARD_ASSIGNMENT_2_-_JAMES_STALLARD_158487_604049842.DOCX		
TIME SUBMITTED	24-APR-2018 02:46PM (UTC+0100)	WORD COUNT	3657
SUBMISSION ID	87332805	CHARACTER COUNT	29137



Assignment 2

Fault Modelling

PSDE6018

Course: BSc Hons Electrical and Electronic
Engineering

Student - James Stallard

Date – 26/4/18

Table of Contents

Figures List	3
1. Introduction	4
2. Research.....	5
2.1 IC Fabrication	5
2.1.1 IC Fabrication Faults.....	5
2.2 Stuck at fault Modelling	6
2.3 Electrostatic discharge	8
3. Design of assignment requirements	9
4. Analysis of MultiSim results	13
5. Analysis of results using 6 bit pattern generator	16
6. Analysis of Labview results	21
6.1 Fault Detection testing.....	22
6.1.1 Fault 1	23
6.1.2 Fault 2	23
6.1.3 Fault 3	23
7. Conclusion and Recommendations	24
8. References	25

Figures List

Figure 2-1 - Example of gate	6
Figure 2-2 - Example of Stuck at fault	6
Figure 2-3 - Error on Y explained	7
Figure 2-4 - Electrostatic discharge band.....	8
Figure 3-1 - Logic Circuit.....	9
Figure 3-2 – Component Tables	9
Figure 3-3 – First Test Table	10
Figure 3-4 - Reduced test table.....	10
Figure 3-5 - Further Reduced Test Table.....	11
Figure 3-6 - Full Circuit	11
Figure 3-7 - PWL Source.....	11
Figure 3-8 - Test Results.....	12
Figure 4-1 - Faults to be done	13
Figure 4-2 - Test 1 circuit	13
Figure 4-3 – Test 1 Result.....	13
Figure 4-4 - Test 2 Circuit	14
Figure 4-5 - Fault 2 testing	14
Figure 4-6 - Test 3 Circuit	15
Figure 4-7 - Fault 3 results	15
Figure 5-1 - Truth table for generator.....	16
Figure 5-2 - K Map for A+	16
Figure 5-3 - K Map for B+	17
Figure 5-4 – K Map for C+	17
Figure 5-5 - K Map for D+.....	18
Figure 5-6 - K Map for E+	18
Figure 5-7 - K Map for F+	19
Figure 5-8 – Compared Testing.....	19
Figure 5-9 - Compared Circuit	20
Figure 6-1 - Labview Circuit.....	21
Figure 6-2 – Labview testing	22
Figure 6-3 – Labview Fault 1	23
Figure 6-4 – Labview Fault 2	23
Figure 6-5 – Labview Fault 3	23

1. Introduction

For this assignment the aims will be to understand and research common faults that occur with integrated circuit boards. This research will have to look into IC fabrication faults, also there will have to be research performed into stuck at faults. There will then have to be research performed into the dangers of electrostatic discharge ESD, this will have to be done to look into the risks of it to electrical components.

There will then have to be work done to implement SSF fault modelling methods, this will be performed on the combinational logic circuit. This has to be done to get a minimum set of testing vectors to get full fault coverage.

*Introduction
is a little
brief*

2. Research

2.1 IC Fabrication

IC fabrication is the process of constructing integrated circuits, this process is made up of the following steps:

Lithography – this is the name given to the process of defining the pattern on the board, this is done using a layer of fluid that is photo resist on the surface of the board. The fluid is hardened by baking using an oven and is selectively removed using a light projection system.

Etching – this is the process of removing the unwanted material off the board's surface. In this process the photo- resist fluid is transferred to the board by using an etching agent.

Deposition – this is the name given to the process of adding different layers to the boards. There are two types of processes for this to be done, the first being physical vapour deposition (PVD). Second is chemical vapour deposition (CVD).

Chemical Mechanical Polishing – this is the process of Applying layers of slurry fluid with etching agent in it and using it to polish the surface of the board's surface.

Oxidation – for this process oxygen is used as either in a dry form or as water for wet oxidation. To convert the silicon layers on top of the board to silicon dioxide.

Ion Implantation – this is the most common method used to introduce impurities into the semi conductive devices, this is done by accelerating ionized particles through the electrical field of the component, and this is directed towards the board's surface.

Diffusion – this step would follow on from the ion implantation, this step is used to heat and cool the board to bombardment-induced lattice defects.

(LEC22, 2009) (Ravindran, 2017) (Jeng, 2002)

2.1.1 IC Fabrication Faults

There are many faults that can occur on IC fabricated boards. These faults are shown below:

Stuck at fault – this fault occurs when a line or gate is stuck at logic 1 or 0. This fault can occur anywhere on the circuit.

Stuck open faults – this fault can occur when a transistor becomes faulty, this means that the transistor can no longer control the current. The component can either be stuck open, where it will not allow any current through. Or it could be stuck closed, as in it is allowing current through even when it shouldn't.

Bridge fault – there are many things that can cause this fault to occur. The first being a direct short across to two pins, this can occur from bad soldering as an excessive use of solder can cause it to run and cross over to other lines. Another reason for the bridge fault to occur is an etching fault, this occurs when the etching machine incorrectly etches the tracks on the surface of the board and causes to lines to connect when they shouldn't.

(Williams, 2007) (Barua, 2005)

2.2 Stuck at fault Modelling

When you are fault modelling on logic circuits you are performing a structural test. This means that you are not just testing every combination of logic 1 and 0 inputs to the circuit, but you will be testing with a reduced number of tests that cover all possible combinations. For this testing the inputs or outputs for the gates can be stuck at logic 1 or 0, see figure 2.1 for example of logic gates input or output positions.

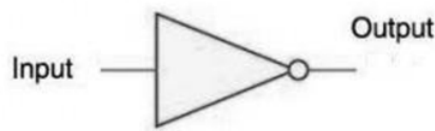


Figure 2-1 - Example of gate

(Meixner, 2018)

This type of modelling has many names for example Boolean, Logic, Abstract and Behavioural model. A stuck at fault is a fault on a Boolean logic circuit, it is not an actual defect with the circuit. This does not necessarily mean that the circuit has been shorted to the supply or to the ground. It is only showing for a stuck high fault that when there is a logic 0 being applied to the circuit the output is still high and is not reacting to the change. An example of a stuck on fault can be seen below in figure 2.2. (Lee, 2010)

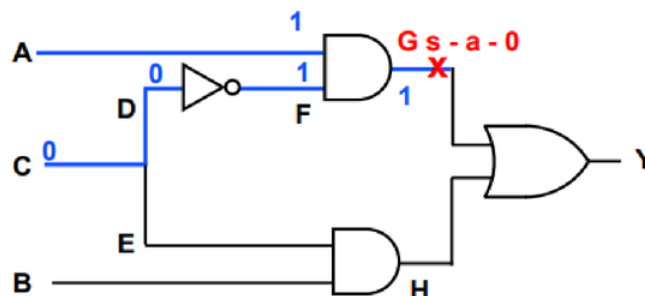
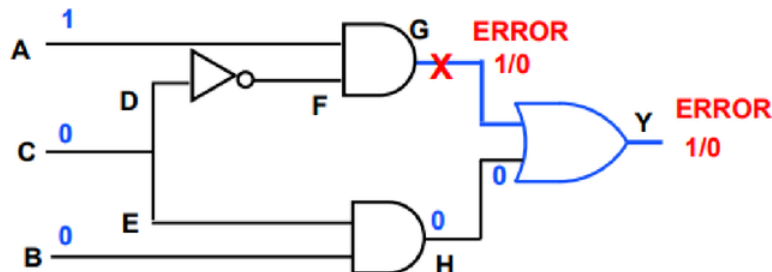


Figure 2-2 - Example of Stuck at fault

(Patel, 2005)

This figure shows a circuit where the output to the and gate G is stuck at logic 0 even though the inputs to the gate are both logic 1, this shows that the logic for G is stuck as it is not reacting to the change in state from the inputs.

This fault will then cause the output from Y to be incorrect depending on it's the input states. This can be seen in the figure 2.2 below.



Good examples of SAF scenarios.

Figure 2-3 - Error on Y explained

(Patel, 2005)

As can be seen in figure 2.2 the logic seen on Y is now no longer the logic expected and this is all due to the stuck at fault on G, this is the basis for the testing technic used for stuck at fault modelling.

There are two types of faults than can occur with the gates in the circuits. The first being internal faults to the gate, this is where the gate is stuck on or open. These faults could be due to internal leakages or shorts between internal connections. The second type of fault is external faults on the gates. This is where the gate has become disconnected or has been shorted. (Bedi, 2017)

2.3 Electrostatic discharge

Electrostatic discharge (ESD) is the name given to the sudden flow of electricity between electrically charged components. This can be caused by contact, electrical shorts or dielectric breakdown. The build up of static electricity is caused by the electrostatic induction.

There are many forms of ESD. On the larger scale, lightening is a form of ESD. On the lower scale ESD can cause damage to sensitive electronic devices. For electrical sparks to occur a field with a strength of 40KV/cm is needed in the air. There are two forms of discharge, the first being corona discharge. This comes from sharp edged electrodes. The second being brush discharge, this comes from blunt edged electrodes. (Rouse, 2018)

ESD can cause failure to solid state electronic components, for example integrated boards. To try and prevent this issue, manufactures do try to ensure that areas where components are handled are static free areas. To try to prevent the electrostatic build up many companies use grounding bands that go around the operator's wrist and is attached to ground. This is used to discharge the static before it is transferred to the components. (Circuits, 2015)



Figure 2-4 - Electrostatic discharge band

(Vastar, 2018)

*Research
is quite
good in
parts, but a
little short in
ESD and
some
examples
with images
would have
been
useful.*

3. Design of assignment requirements

As can be seen in figure 3.1, this is the logic circuit that will have to be modelled.

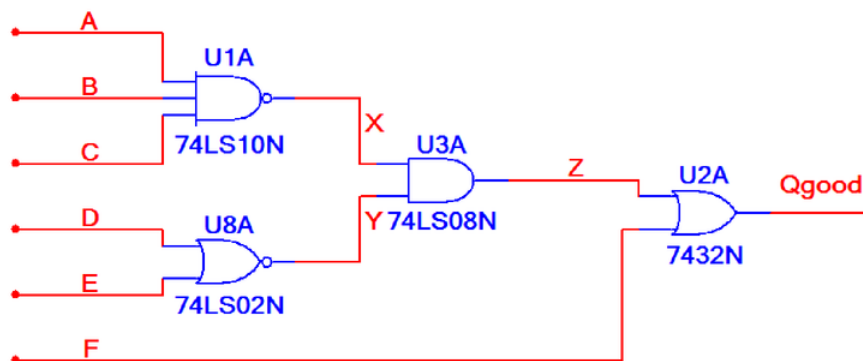


Figure 3-1 - Logic Circuit

From this circuit the following tables that can be seen within below have been made, see figure 3.2. These tables have been for all four components.

NAND				AND		
A	B	C	X	X	Y	Z
1	1	1	0	1	1	1
0	1	1	1	0	1	0
1	0	1	1	1	0	0
1	1	0	1			
NOR				OR		
D	E	Y		Z	F	QGOOD
0	0	1		0	0	0
1	0	0		0	1	1
0	1	0		1	0	1

Figure 3-2 – Component Tables

From the gate tables above for each component the truth table has been made so that it shows the minimum variants for the least testing, this has to be done to prevent any tests being repeated.

Fault models are clear for each gate.

The gate tables for each components have been used to create the test table that can be seen in figure 3.3.

Test No	A	B	C	D	E	F	Q good	HEX	X	Y	Z	Fault
1	1	1	1	0	0	0	0	70	0	1	0	A+B+C SA0 X
2	0	1	1	0	0	0	1	31	1	1	1	A SA1
3	1	0	1	0	0	0	1	51	1	1	1	B SA1
4	1	1	0	0	0	0	1	61	1	1	1	C SA1
5	0*	0*	0*	0	0	0	1	1	1	1	1	D+E SA1 X+Y SA0,
6	0*	0*	0*	1	0	0	0	8	1	0	0	D SA0
7	0*	0*	0*	0	1	0	0	4	1	0	0	E SA0
8	0*	0*	0*	0	0	0	1	1	1	1	1	X+Y SA0
9	1	1	1	0	0	0	0	70	0	1	0	X SA1
10	0*	0*	0*	1**	1**	0	0	0C	1	0	0	Y SA1
11	1	1	1	1**	1**	0	0	7C	0*	0*	0	F+Z SA1
12	1	1	1	1**	1**	1	1	7F	0*	0*	0	F SA0
13	0*	0*	0*	0	0	0	1	1	1	1	1	Z SA0

Figure 3-3 – First Test Table

The derived test table seen above in figure 3.3 shows all the test vector. If we look at the table we can see that the red values are taken straight from the inputs of the gate tables. The blue values are the outputs from the gate tables. The fault field displays the fault that would be tested for. Finally the hex column is used to show if a test is being repeated, using this the table has been reduced by removing the repeated tests so only one is performed. This can be seen below in figure 3.4.

Test No	A	B	C	D	E	F	Q good	HEX	X	Y	Z	Fault
1	1	1	1	0	0	0	0	70	0	1	0	A+B+C SA0 X SA1
2	0	1	1	0	0	0	1	31	1	1	1	A SA1
3	1	0	1	0	0	0	1	51	1	1	1	B SA1
4	1	1	0	0	0	0	1	61	1	1	1	C SA1
5	0*	0*	0*	0	0	0	1	1	1	1	1	D+E SA1 X+Y SA0, Z SA0
6	0*	0*	0*	1	0	0	0	8	1	0	0	D SA0
7	0*	0*	0*	0	1	0	0	4	1	0	0	E SA0
8	0*	0*	0*	1**	1**	0	0	0C	1	0	0	Y SA1
9	1	1	1	1**	1**	0	0	7C	0*	0*	0	F+Z SA1
10	1	1	1	1**	1**	1	1	7F	0*	0*	0	F SA0

Figure 3-4 - Reduced test table

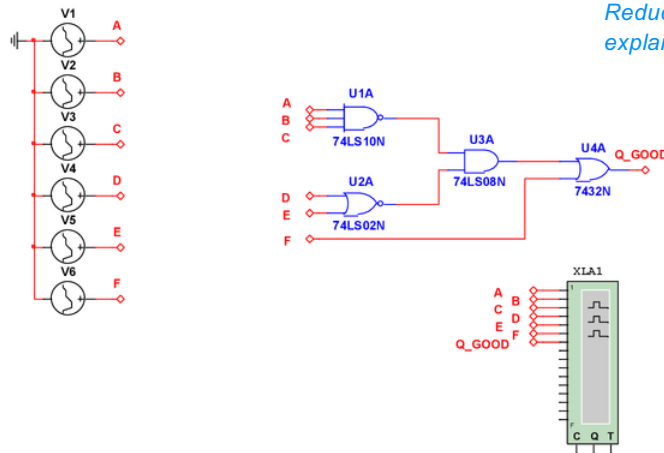
The above table (figure 3.4) shows the reduced table, this table can still be reduced further by changing the don't care values that are identified with * and **, these qualify are don't care values as changing these inputs will not have an effect on their outputs, this means they can be either logic 1 or 0. The reason to change the inputs is to change the hex value to match another

one. One this has been done the matching hex value tests can be removed so only one is on the table. The further reduced table can be seen in figure 3.5.

Test No	A	B	C	D	E	F	Q good	HEX	X	Y	Z	Fault
1	1	1	1	0	0	0	0	70	0	1	0	A+B+C SA0 X SA1
2	0	1	1	0	0	0	1	31	1	1	1	A SA1
3	1	0	1	0	0	0	1	51	1	1	1	B SA1
4	1	1	0	0	0	0	1	61	1	1	1	C SA1
6	0*	0*	0*	1	0	0	0	8	1	0	0	D SA0
7	0*	0*	0*	0	1	0	0	4	1	0	0	E SA0
9	1	1	1	1**	1**	0	0	7C	0*	0*	0	F+Z SA1
10	1	1	1	1**	1**	1	1	7F	0*	0*	0	F SA0

Figure 3-5 - Further Reduced Test Table

Figure 3.6 is the test circuit that has been created from the test table so that all the test can be performed. To generate the inputs, Pulse Width Length (PWL) sources have been used. For the Multisim programme the PWL sources have to be set up this can be seen in figure 3.7. The full circuit can be seen in figure 3.6.



Reduction methods are good and explained well.

This should include the x-or gate.

Figure 3-6 - Full Circuit

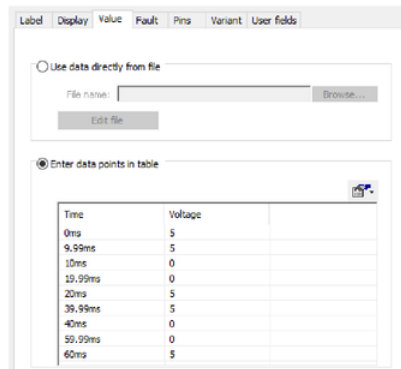


Figure 3-7 - PWL Source

As can be seen below in figure 3.8, the circuit has been tested and the results can be seen. If we compare the test results to the reduced test table we can see that both match, this proves that the testing has been done correctly and the circuit has been designed correctly.

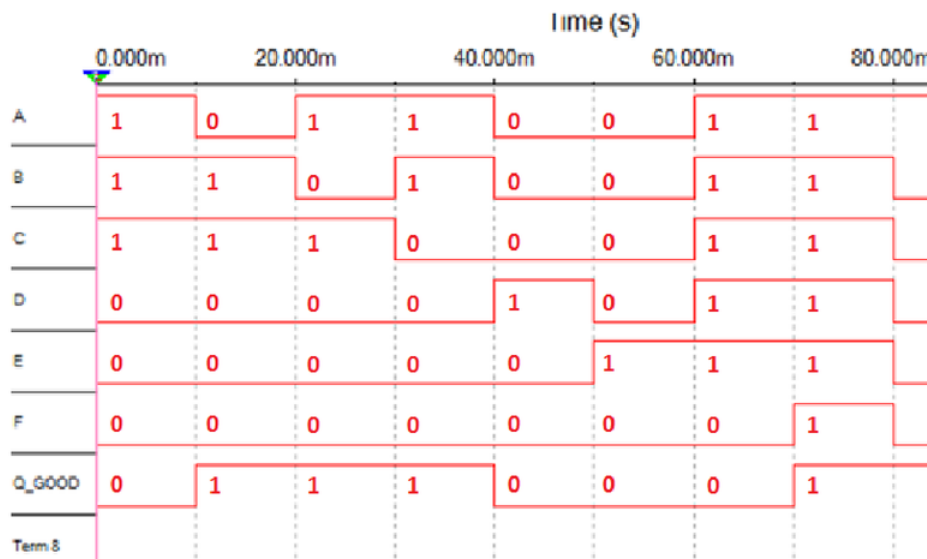


Figure 3-8 - Test Results

The tests are correct.

4. Analysis of MultiSim results

For this section of the assignment there will be three faults entered into the circuit, this will be done to test the circuit. The faults will be seen in tests 1, 7 and 10. The faults have been highlighted in the test table and this can be seen in figure 4.1.

Test No	A	B	C	D	E	F	Q good	HEX	X	Y	Z	Fault
1	0	1	1	0	0	0	0	70	0	1	0	A+B+C SA0 X
7	0*	0*	0*	0	0	0	0	4	1	0	0	E SA0
10	1	1	1	1**	1**	0	1	7F	0*	0*	0	F SA0

Figure 4-1 - Faults to be done

We can see below that the circuit has been modified for the fault testing, see figure 4.2. As can be seen the A input has been grounded so will be stuck at logic 0

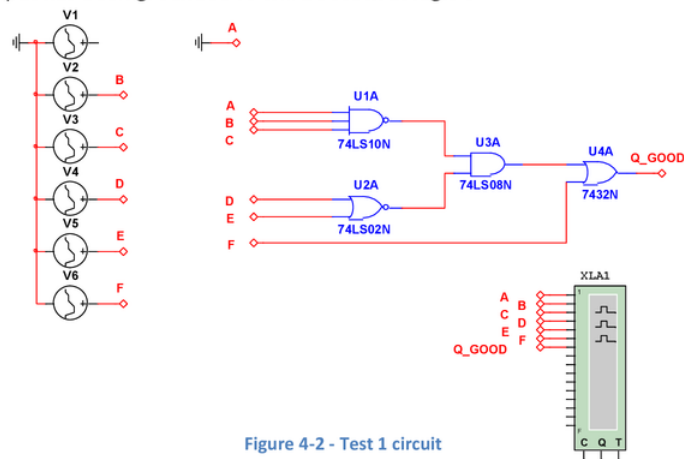


Figure 4-2 - Test 1 circuit

From this circuit the following test was performed, this can be seen in figure 4.3. If we look at the test graph we can see that the input A is stuck at logic 0. If we compare the Q_GOOD from the fault test and the good test we can see that it is at Logic 1 instead of logic 0. This proves that a fault changes the output of the circuit.

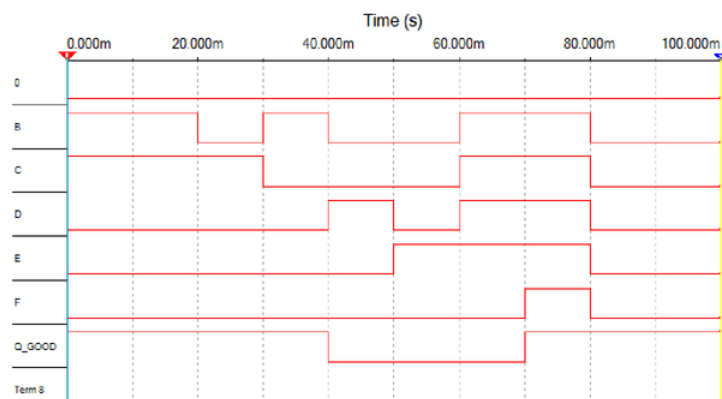


Figure 4-3 – Test 1 Result

This test is correct but it should include the x-or test.

We can see below in figure 4.4, the circuit has been modified so the fault for this test is on input E. This fault has been grounded so that it is stuck at logic 0.

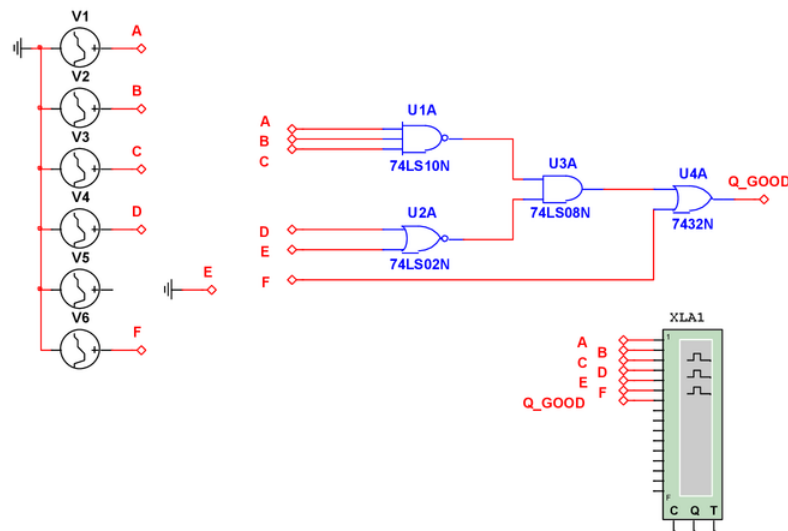


Figure 4-4 - Test 2 Circuit

From this circuit the following test has been performed, this result can be seen in figure 4.4. If we look at the test results we can see that input E is stuck at logic 0. If we compare the Q_GOOD on the good test and result from the test 2, we can see that there is a change with the Q_GOOD between 50 and 60ms. At this point the Q_GOOD is at logic 1 when it should be at logic 0.

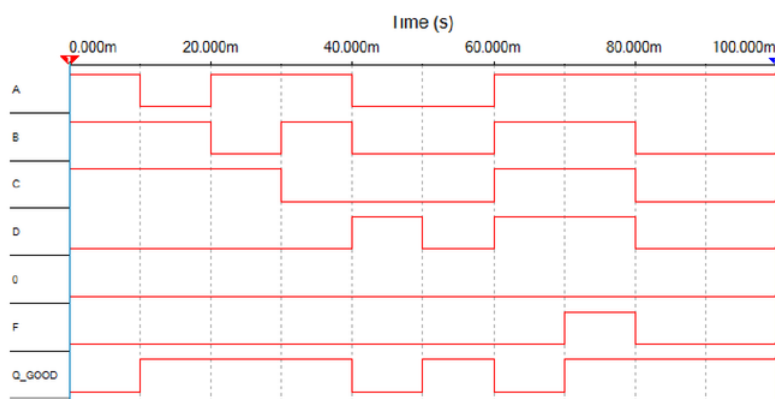


Figure 4-5 - Fault 2 testing

We can see below in figure 4.6, the circuit has been modified so the fault for this test is on input F. This fault has been grounded so that it is stuck at logic 0.

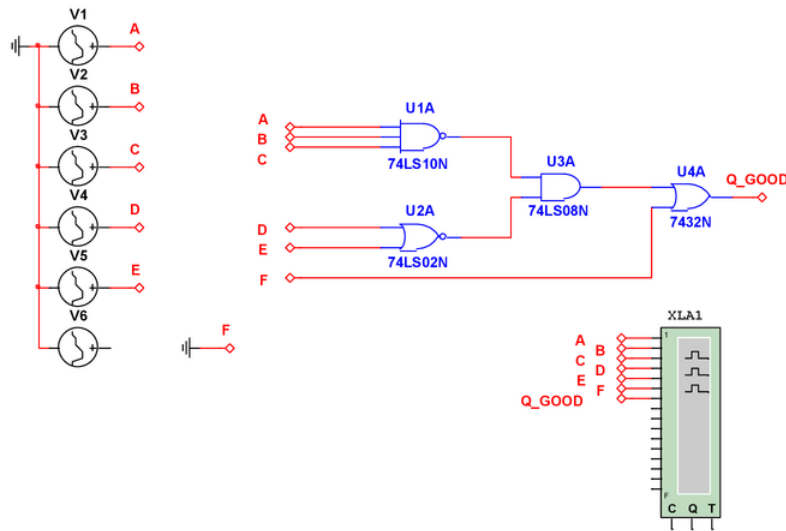


Figure 4-6 - Test 3 Circuit

From this circuit the following test has been performed, this result can be seen in figure 4.7. If we look at the test results we can see that the input F is stuck at logic 0. If we compare the Q_GOOD logic on the good test and the result from test 3, we can see that there is a change with Q_GOOD. The Q_GOOD goes to logic 1 at 10ms and at 80ms, and is at logic 0 at 70ms. This proves that the input F being stuck at logic 0 has an effect on the Q_GOOD logic

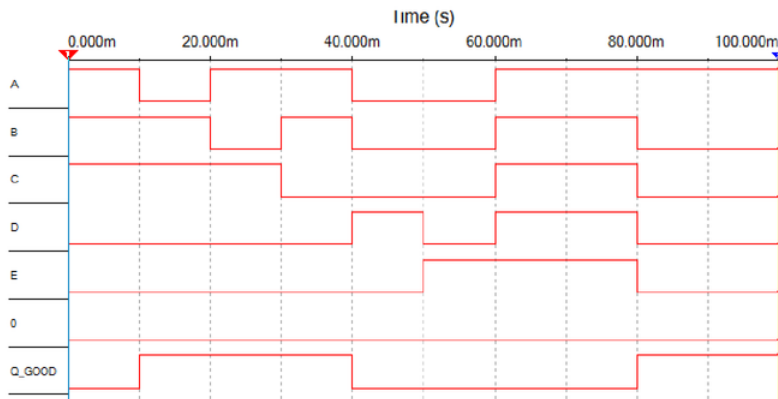


Figure 4-7 - Fault 3 results

Testing using this method is good and shows the fault at the output. The X-OR gate comparator should be implemented here and explained.

5. Analysis of results using 6 bit pattern generator

Figure 5.1 shows the truth table for the 6 Bit pattern generator, the table has been created to show the present state and the new state values.

Present State						Next State					
A	B	C	D	E	F	A+	B+	C+	D+	E+	F+
1	1	1	0	0	0	0	1	1	0	0	0
0	1	1	0	0	0	1	0	1	0	0	0
1	0	1	0	0	0	1	1	0	0	0	0
1	1	0	0	0	0	0	0	0	1	0	0
0	0	0	1	0	0	0	0	0	0	1	0
0	0	0	0	1	0	1	1	1	1	1	0
1	1	1	1	1	0	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	0	0

Figure 5-1 - Truth table for generator

Figure 5.2 shows the K map for the input A from the truth table. The logic 1's have been grouped and are highlighted in red. From these the Boolean expression has been created. This will be used to create the Multisim circuit.

A+									
DEF	ABC								
		000	001	011	010	100	101	111	110
	000	0	0	1	0	0	1	0	0
	001	0	0	0	0	0	0	0	0
	011	0	0	0	0	0	0	0	0
	010	1	0	0	0	0	0	0	0
	100	0	0	0	0	0	0	0	0
	101	0	0	0	0	0	0	0	0
	111	0	0	0	0	0	0	1	0
	110	0	0	0	0	0	0	1	0

$$N = A.B.C.D.E + A.NB.C.ND.NE.NF + NA.B.C.ND.NE.NF + NA.NB.NC.ND.E.NF$$

Figure 5-2 - K Map for A+

Figure 5.3 shows the K map for the input B from the truth table. The logic 1's have been grouped and are highlighted in red. From these the Boolean expression has been created. This will be used to create the Multisim circuit.

B+									
DEF	ABC								
		000	001	011	010	100	101	111	110
	000	0	0	0	0	0	1	1	0
	001	0	0	0	0	0	0	0	0
	011	0	0	0	0	0	0	0	0
	010	1	0	0	0	0	0	0	0
	100	0	0	0	0	0	0	0	0
	101	0	0	0	0	0	0	0	0
	111	0	0	0	0	0	0	1	0
	110	0	0	0	0	0	0	1	0

$$N = A.C.ND.NE.NF + A.B.C.D.E + NA.NB.NC.ND.E.NF$$

Figure 5-3 - K Map for B+

Figure 5.4 shows the K map for the input C from the truth table. The logic 1's have been grouped and are highlighted in red. From these the Boolean expression has been created. This will be used to create the Multisim circuit.

C+									
DEF	ABC								
		000	001	011	010	100	101	111	110
	000	0	0	1	0	0	0	1	0
	001	0	0	0	0	0	0	0	0
	011	0	0	0	0	0	0	0	0
	010	1	0	0	0	0	0	0	0
	100	0	0	0	0	0	0	0	0
	101	0	0	0	0	0	0	0	0
	111	0	0	0	0	0	0	1	0
	110	0	0	0	0	0	0	1	0

$$N = B.C.ND.NE.NF + A.B.C.D.E + NA.NB.NC.ND.E.NF$$

Figure 5-4 – K Map for C+

Figure 5.5 shows the K map for the input D from the truth table. The logic 1's have been grouped and are highlighted in red. From these the Boolean expression has been created. This will be used to create the Multisim circuit.

D+									
DEF	ABC								
		000	001	011	010	100	101	111	110
	000	0	0	0	0	0	0	0	1
	001	0	0	0	0	0	0	0	0
	011	0	0	0	0	0	0	0	0
	010	1	0	0	0	0	0	0	0
	100	0	0	0	0	0	0	0	0
	101	0	0	0	0	0	0	0	0
	111	0	0	0	0	0	0	0	0
	110	0	0	0	0	0	0	1	0

$$N = NA.NB.NC.ND.E.F + A.B.NC.ND.NE.NF + A.B.C.D.E.NF$$

Figure 5-5 - K Map for D+

Figure 5.6 shows the K map for the input E from the truth table. The logic 1's have been grouped and are highlighted in red. From these the Boolean expression has been created. This will be used to create the Multisim circuit.

E+									
DEF	ABC								
		000	001	011	010	100	101	111	110
	000	0	0	0	0	0	0	0	0
	001	0	0	0	0	0	0	0	0
	011	0	0	0	0	0	0	0	0
	010	1	0	0	0	0	0	0	0
	100	1	0	0	0	0	0	0	0
	101	0	0	0	0	0	0	0	0
	111	0	0	0	0	0	0	0	0
	110	0	0	0	0	0	0	1	0

$$N = NA.NB.NC.ND.E.NF + NA.NB.NC.D.NE.NF + A.B.C.D.E.NF$$

Figure 5-6 - K Map for E+

Figure 5.7 shows the K map for the input E from the truth table. The logic 1's have been grouped and are highlighted in red. From these the Boolean expression has been created. This will be used to create the Multisim circuit.

F+									
DEF	ABC	000	001	011	010	100	101	111	110
	000	0	0	0	0	0	0	0	0
	001	0	0	0	0	0	0	0	0
	011	0	0	0	0	0	0	0	0
	010	0	0	0	0	0	0	0	0
	100	0	0	0	0	0	0	0	0
	101	0	0	0	0	0	0	0	0
	111	0	0	0	0	0	0	0	0
	110	0	0	0	0	0	0	1	0

N = A.B.C.D.E.NF

Figure 5-7 - K Map for F+

*Good work with k-maps
for circuit reduction and
use of the mirror method.*

If we look at figure 5.8 we can see the results from the testing of the alternative method using Multisim. If we compare the two results from the two methods we can see that they are the same, this proves that it is possible to obtain the circuit from the assignment's spec using multiply methods.

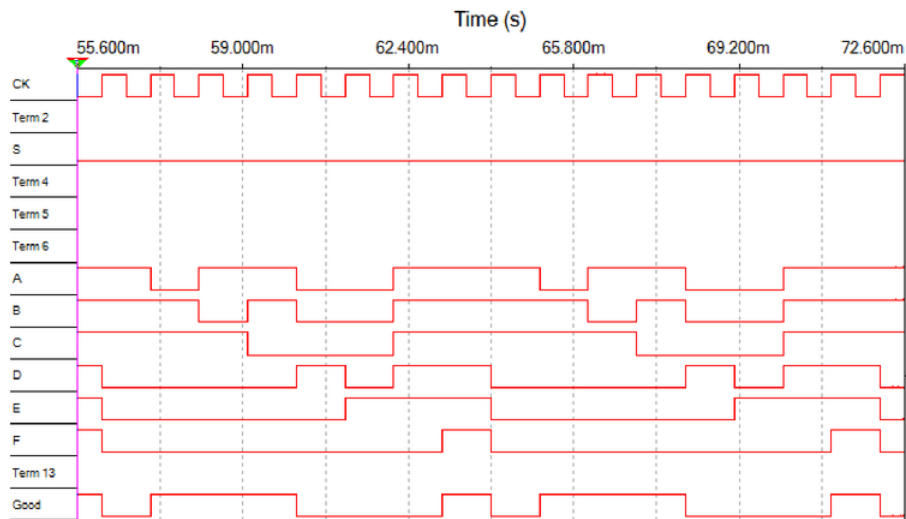
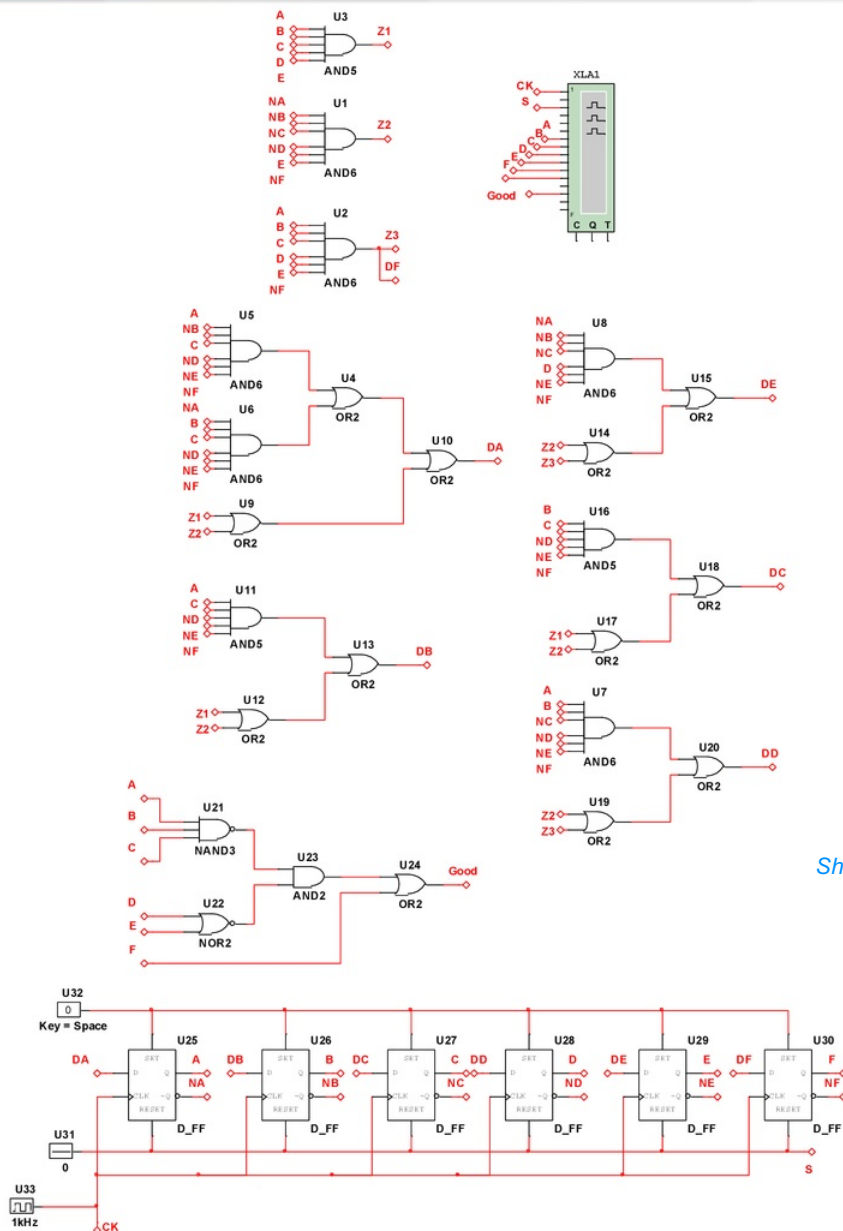


Figure 5-8 – Compared Testing



Should include x-or gate.

Figure 5-9 - Compared Circuit

Figure 5.9 shows the circuit for the circuit that has been created from the K Maps using the Multisim software. The circuit is based around D type flip flops.

The design looks correct and the tests are good which show the 8 vectors. The X-OR should be implemented here also and explained.

6. Analysis of Labview results

The circuit for the Labview can be seen above in figure 7.1. The outer while loop is used to house the for loop. There is within this an array that houses the digital values for the six inputs. The decimal values are converted from the hex values seen earlier. The digital condition loop is used to show the conditions of the inputs and the outputs. This loop will run 8 steps, the other array is used to control how long each step will last. For this circuit they are all set to 2 seconds. During each step the decimal value is sampled and is outputted to the output task.

The other while loop is connected to the XOR logic gate. In here the value from the Labview is compared to the value from the IC, if there is any differences then a fault bit would be brought on. This is then connected to a fault light on the operation screen.

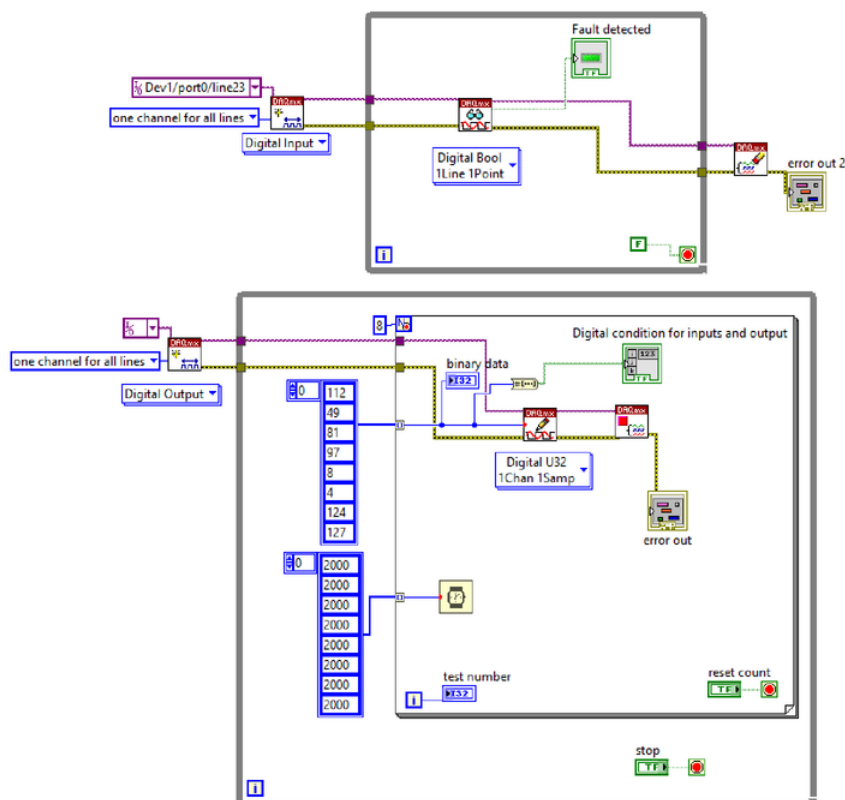


Figure 6-1 - Labview Circuit

If we look at figure 7.2, we can see the operator screen for the Labview circuit. All of the images show the programme running through all 8 steps. The lights will show the condition of the inputs, if

the logic is 1 or 0. The binary value is then displayed on the binary data. If there was a fault then the fault detected light will come on.

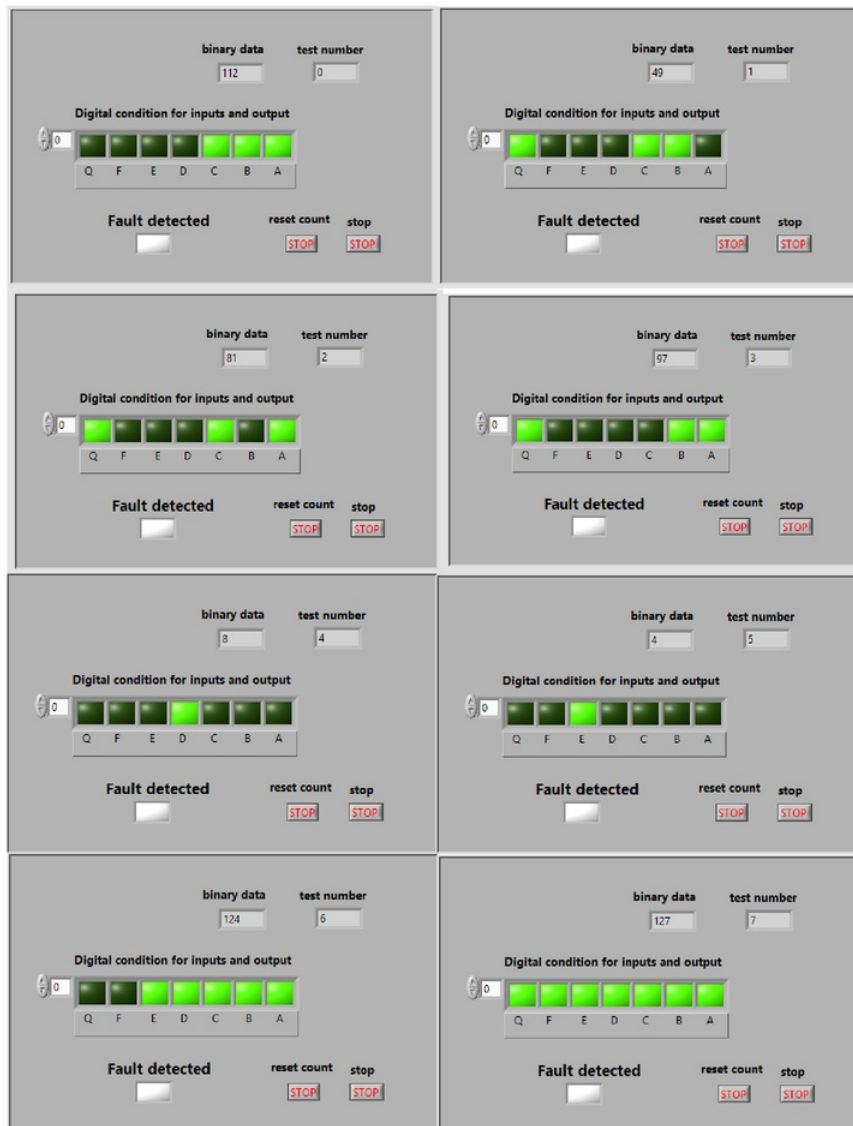


Figure 6-2 – Labview testing

Test vectors are good and cover all 8 tests.

6.1 Fault Detection testing

For the fault testing there has been three faults placed on the circuit, this is being done to prove that the fault detect light works.

6.1.1 Fault 1

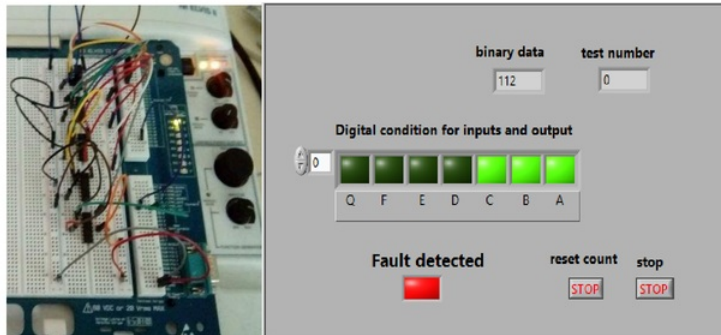


Figure 6-3 – Labview Fault 1

For this fault the input A has been connected to the ground, this means that it is stuck at logic 0.

Figure 7.3 shows that the fault light is on when input A is at logic 0, we can see why this is as the Labview is looking for the input to be at logic 1, this can be seen in the image.

6.1.2 Fault 2

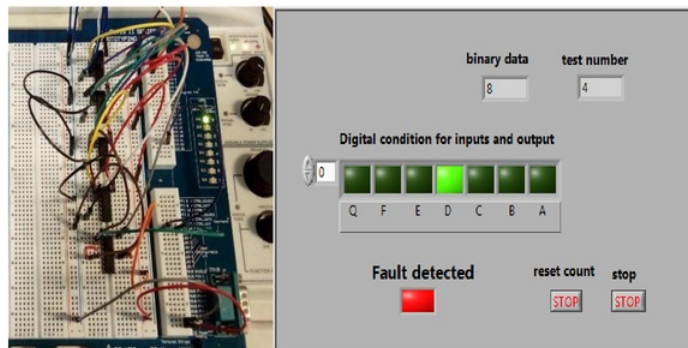


Figure 6-4 – Labview Fault 2

For this fault the input D has been connected to the ground, this means that it is stuck at logic 0.

Figure 7.4 shows that the fault light is on when input D is at logic 0, we can see why this is as the Labview is looking for the input to be at logic 1, this can be seen in the image.

6.1.3 Fault 3

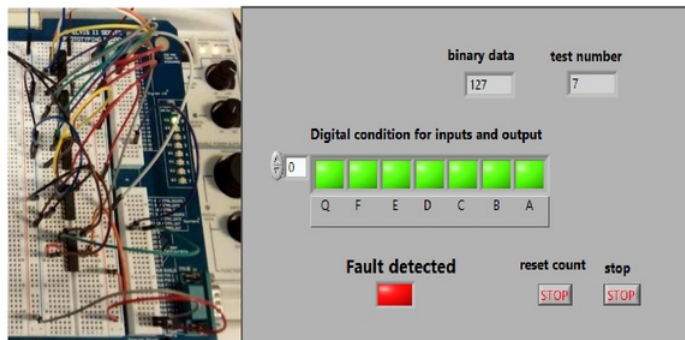


Figure 6-5 – Labview Fault 3

For this fault the input F has been connected to the ground, this means that it is stuck at logic 0.

Figure 7.5 shows that the fault light is on when input F is at logic 0, we can see why this is as the Labview is looking for the input to be at logic 1, this can be seen in the image.

Tests are good and the design in LabVIEW is explained well.

7. Conclusion and Recommendations

In this assignment there has been research performed into common faults that occur with integrated circuit boards. For the research there has been IC fabrication faults, this includes the common faults that can occur during the building process or by handling the components. Also there has been research performed into stuck at faults, this includes how the faults occur and all the types of faults that come under this title. Final there has been research performed into how ESD (electrostatic discharge) affects or damages electrical components.

There has been work performed into implementing SSF fault modelling, this has been done using two different method on the Multisim and Labview software. These have been used to create the combinational logic circuit needed for testing. This were done to achieve the minimum amounts of test required to cover all faults. This has been done and the testing can be seen with many software to achieve the same results.

*The conclusion
should have some
discussion on the
comparison of the
methods used.
Needs more detail
of what the overall
analysis has
demonstrated.*

8. References

- Barua, P., 2005. *Fault Diagnosis of intergated circuit*. [Online]
Available at: <file:///C:/Users/James/Downloads/9780387257426-c2.pdf>
[Accessed 22/4/18].
- Bedi, H., 2017. *What is stuck at fault in DFT?*. [Online]
Available at: <https://www.quora.com/What-is-stuck-at-fault-in-DFT>
[Accessed 22/4/18].
- Circuits, M., 2015. *The Prevention and Control of electrostatic discharge (ESD)*. [Online]
Available at: <https://ww2.minicircuits.com/app/AN40-005.pdf>
[Accessed 22/4/18].
- EEWeb, 2012. *Prevention and Control of Electrostatic Discharge*. [Online]
Available at: <https://www.eeweb.com/app-notes/prevention-and-control-of-electrostatic-discharge>
- Jeng, M. D., 2002. *IEEE xplore*. [Online]
Available at: <https://ieeexplore.ieee.org/document/625761/>
[Accessed 22/4/18].
- LEC22, 2009. *Integrated circuit fabrication*. [Online]
Available at: <http://nptel.ac.in/courses/113106062/Lec22.pdf>
[Accessed 22/4/18].
- Lee, E., 2010. *Testing for Fault Detection/Diagnosis*. [Online]
Available at: <https://embedded.eecs.berkeley.edu/eecs44/fall2011/lectures/StuckAtTesting.pdf>
[Accessed 22/4/18].
- Meixner, A., 2018. *Digital Circuits and the Stuck at Fault Model*. [Online]
Available at: <https://accendoreliability.com/digital-circuits-stuck-fault-model/>
- Patel, J. H., 2005. *Stuck-At Fault: A Fault Model for the next Millennium?*. [Online]
Available at: http://web.stanford.edu/class/ee386/public/stuck_at_fault_6per_page
- Ravindran, T., 2017. *Mindmeister*. [Online]
Available at: <https://www.mindmeister.com/1004901723/ic-fabrication>
[Accessed 22/4/18].
- Rouse, M., 2018. *electrostatic discharge (ESD)*. [Online]
Available at: <https://whatis.techtarget.com/definition/electrostatic-discharge-ESD>
[Accessed 22/4/18].
- SRI, 2005. *ESD*. [Online]
Available at: <http://www.sourceresearch.com/newsletter/ESD.cfm?emART>
- Vastar, 2018. *Vastar ESD Wrist Strap alligator clip Anti Static Discharge Band*. [Online]
Available at: <https://www.aliexpress.com/item/Vastar-ESD-Anti-Static-Wrist-Straps-Elastic-Band-with-Clip-for-Sensitive-Electronics-Repair-Work-Tools/32814788142.html>

Williams, H., 2007. *Failures in Integrated Circuits*. [Online]
Available at: http://ece-research.unm.edu/jimp/vlsi_test/slides/html/defects1.html
[Accessed 22/4/18].

*The sources are
relevant and
appropriate number
used for the task.*

assignment 2 - james stallard

GRADEMARK REPORT

FINAL GRADE

GENERAL COMMENTS

Instructor

62/100

PAGE 1

PAGE 2

PAGE 3

Text Comment. Good list of contents

PAGE 4

Text Comment. Introduction is a little brief

PAGE 5

PAGE 6

PAGE 7

Text Comment. Good examples of SAF scenarios.

PAGE 8

Text Comment. Research is quite good in parts, but a little short in ESD and some examples with images would have been useful.

PAGE 9

Text Comment. Fault models are clear for each gate.

PAGE 10

PAGE 11

Text Comment. Reduction methods are good and explained well.

Text Comment. This should include the x-or gate.

Text Comment. The tests are correct.

Text Comment. This test is correct but it should include the x-or test.

Text Comment. Testing using this method is good and shows the fault at the output. The X-OR gate comparator should be implemented here and explained.

Text Comment. Good work with k-maps for circuit reduction and use of the mirror method.

Text Comment. Should include x-or gate.

Text Comment. The design looks correct and the tests are good which show the 8 vectors. The X-OR should be implemented here also and explained.

Text Comment. Test vectors are good and cover all 8 tests.

Text Comment. Tests are good and the design in LabVIEW is explained well.

Text Comment. The conclusion should have some discussion on the comparison of the methods used. Needs more detail of what the overall analysis has demonstrated.

Text Comment. Good number of relevant sources.

Text Comment. The sources are relevant and appropriate number used for the task.

GRADING FORM: CSG-RUBIC

FREDERICK STALLARD

0

RESEARCH-INFORMED LITERATURE



Research is a little brief, but relevant sources have been implemented. IC fabrication - 2.5/5 ESD - 2.5/5 SSF - 3/5

8/15

KNOWLEDGE AND UNDERSTANDING



Minimum test vectors for design has been implemented and explained - 7/10 LabVIEW design and explanation is good - 7/10 ECAD design as alternative should have included the X-OR gate comparator with an explanation of its importance. This is left out of this part, but the test is still valid because the output shows the fault. - 6/10

20/30

ANALYSIS



ECAD tests are good but should show the X-OR comparator function. - 6/10 Comparison to theory has been achieved This would have been better with the comparator implemented. - 3/5

9/15

PRACTICAL APPLICATION AND DEP.



Prototyping of test circuit has been demonstrated and explained. The X-OR function would have explained the testing of the prototype. - 6/10
LabVIEW testing is good and operation of design is explained. - 11/15

17/25

SKILLS FOR PROFESSIONAL PRAC.



Use of appropriate software - 3/5 Report layout is good - 3/5 Conclusion is a little weak - 2/5

8/15