

assignment 2 final

by Joshua Davies

FILE	8847_JOSHUA_DAVIES_ASSIGNMENT_2_FINAL_158487_14094654.DOC X		
TIME SUBMITTED	03-MAY-2018 06:03PM (UTC+0100)	WORD COUNT	4858
SUBMISSION ID	88273883	CHARACTER COUNT	34 129

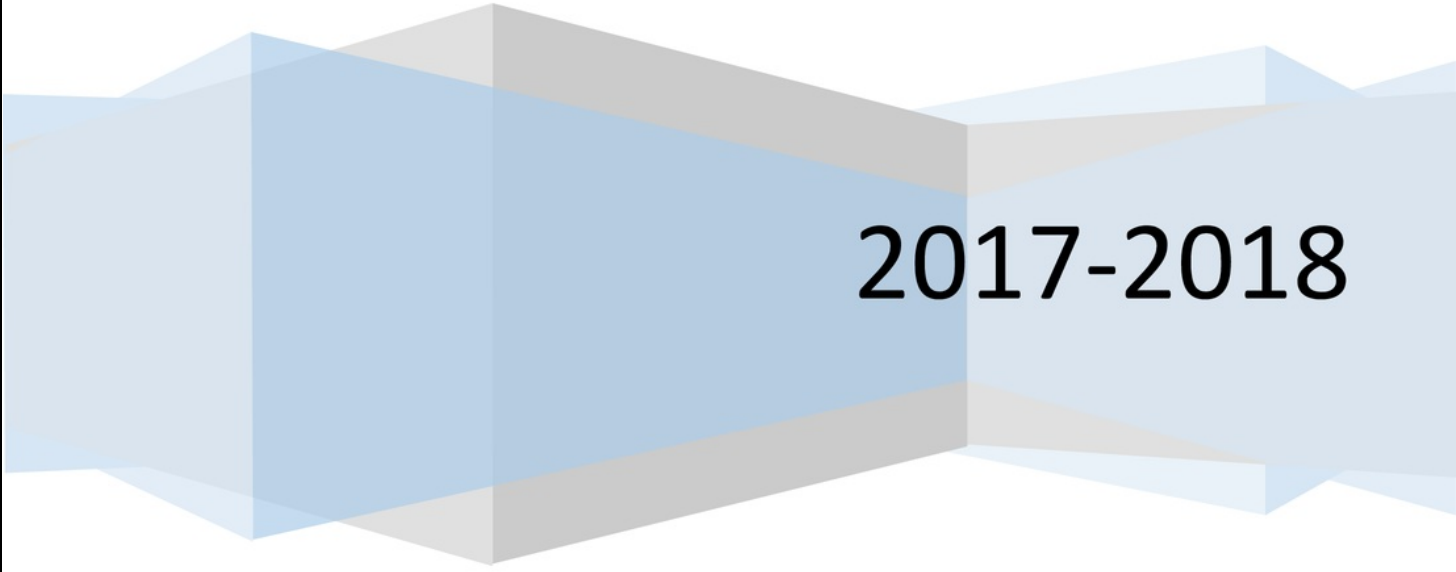


colegsirgâr

Fault Modelling

PSDE 6018 – Electronic Production and Test

Joshua Davies



2017-2018

Contents

Table of Figures	3
1. Introduction.....	5
2. Research.....	6
2.1. IC Fabricated Faults	6
2.1.1. Single Stuck-at Faults	7
2.2. Electro Static Discharge (ESD).....	8
2.3. ESD Damage and Prevention Methods.....	10
3. MultiSIM Design	13
3.1. Fault Modelling.....	13
4. Analysis of NI MultiSim Design	18
4.1. MultiSim Test 1 – A Stuck at 0.....	19
4.2. MultiSim Test 2 – D Stuck at 0.....	20
4.3. MultiSim Test 3 – F Stuck at 0	21
5. 6Bit Pattern Generator Design and Test	22
5.1. 6Bit Pattern Generator Design	22
5.2. 6Bit Pattern Generator Testing	26
6. NI LabVIEW Design and Practical Build.....	28
6. Analysis of LabVIEW Practical Build.....	30
6.2. Implementing Faults	34
6.2.1. Practical Test 1 – A Stuck at 0	34
6.2.2. Practical Test 2 – D Stuck at 0	35
6.2.3. Practical Test 3 – F Stuck at 0.....	36
7. Conclusion	37
8. Reference List	38
9. Appendix.....	39

Table of Figures

Figure 1 - Stuck-at fault example	7
Figure 2 - Electrostatic Discharge.....	8
Figure 3 - Magnitude of Voltage Developed in ESD	9
Figure 4 - Figures of ESD Susceptibility of Devices.....	10
Figure 5 - ESD Event with a Transistor	10
Figure 6 - ESD Damage of the Transistor Bond	11
Figure 7 - ESD Protected work area	12
Figure 8 - Technicians Earthing Strap.....	12
Figure 9 - MultiSim Design	13
Figure 10 - 3 Input And Gate Test Vectors.....	13
Figure 11 - 2 Input And Gate Test Vectors.....	14
Figure 12 - 2 Input Or Gate Test Vectors	14
Figure 13 - 2nd 2 Input Or Gate Test Vectors	14
Figure 14 – Test table for the Test Circuit.....	15
Figure 15 - Reduced Test Table	15
Figure 16 - Further Reduced Test Table.....	16
Figure 17 - MultiSim simulation of Test circuit against Piece wise tools	16
Figure 18 - Test vectors inputted into Piece Wise Tool	17
Figure 19 - No Faults MultiSim.....	18
Figure 20 - Test 1, A stuck at 0	19
Figure 21 – Test 1, A Stuck at 0 Logic Analyzer Result	19
Figure 22 - Test 2, D stuck at 0.....	20
Figure 23 - Test 2, D stuck at 0 Logic analyzer results	20
Figure 24 - Test 3, F stuck at 0	21
Figure 25 - Test 3, F Stuck at 0 Logic analyzer results.....	21
Figure 26 – Present and Next State values of the 6 bit register	22
Figure 27 - A+ Karnaugh Map	22
Figure 28 - B+ Karnaugh Map.....	23
Figure 29 - C+ Karnaugh Map.....	23
Figure 30 - D+ Karnaugh Map	24
Figure 31 - E+ Karnaugh Map.....	24
Figure 32 - F+ Karnaugh Map	25
Figure 33 - 6 Bit Register in MultSim	25
Figure 34 - 6 Bit pattern generator Logic analyzer Results	26
Figure 35 - NI LabView 6Bit Pattern Generator Design	28
Figure 36 - Test Circuit built on NI ELVIS Board	29
Figure 37 - LabView Test 0	30
Figure 38 - LabView Test 1	30
Figure 39 - LabView Test 2	31
Figure 40 - LabView Test 3	31

Figure 41 - LabView Test 4	32
Figure 42 - LabView Test 5	32
Figure 43 - LabView Test 6	33
Figure 44 - LabView test 7	33
Figure 45 - NI ELVIS board at Test 1	34
Figure 46 - Fault 1, A Stuck at 0	34
Figure 47 - Fault 2, D Stuck at 0	35
Figure 48 - NI ELVIS Board at Test 2	35
Figure 49 – NI ELVIS board at Test 3	36
Figure 50 - Fault 3, F Stuck at 0	36

Good list of contents.

1. Introduction

The aims of this assignment is to investigate into IC Fabrication and the utilization of 'Single Stuck at Faults' (SSF) fault model to test a simple industrial control process PCB design. To Research into the effects of Electro Static Discharge (ESD) and the counter measures taken to prevent the dangers of ESD in electronic design and manufacturing. To Implement SSF fault modelling techniques on combinational logic circuits to define the minimum amount of test vectors to achieve maximum fault coverage.

The objectives of this assignment contain building a circuit prototype of a given combinational logic circuit using MultiSim that consists of four gates, with a total of 6 inputs and a single output. To design a 6 bit pattern generator using the minimum amount of test vectors to test against the prototype circuit. To design and build the prototype circuit on NI LabVIEW software utilizing an NI ELVIS board applying the same tests between the circuits and analyzing the results obtained.

2. Research

2.1. IC Fabricated Faults

There are a number of fabricated fault tests that are carried out but modelling faults is preferred because input/output tests are insufficient for manufacturing, real mechanical defects in circuits are too frequent and analysis of the fault is tough.

Fault modelling is preferred as the designer will be generating known faults to test their circuits where they can test a circuit against a table of expected faults that makes the testing stage easier. Using fault modelling, an analysis can be achieved as the circuit under test can be compared with generated faults.

Common fault models are single stuck at faults, transistor open and short faults, memory faults, PLA faults functional tests of processors, delay and analog faults.

Real defects in chips include processing defects such as missing contact windows parasitic transistors and oxide breakdown. Material defects such as bulk defects like cracks, crystal imperfections and surface impurities from ion migration. Time dependant failures like dielectric breakdown and electro migration.

Defect classes	Occurrence frequency (%)
Shorts	51
Opens	1
Missing components	6
Wrong components	13
Reversed components	6
Bent leads	8
Analog specifications	5
Digital logic	5
Performance (timing)	5

Figure 1 - PCB Defects

(web.eecs.umich.edu, 2002)

*Good
use of
tables
for
example
s.*

2.1.1. Single Stuck-at Faults

The most common modelling technique used is the Single stuck at fault for fault modelling. The model assumes that a fault within a logic gate results in either an input or output is fixed to logic 1 or logic 0.

There are three properties that define a single stuck-at fault. These are:

- Only one faulty line
- Faulty line is permanently fixed to 0 or 1
- The fault can be at an input or output of a gate. (web.eecs.umich.edu, 2002)

Figure 2 shows an example of a stuck-at 0 fault applied to a logic circuit.

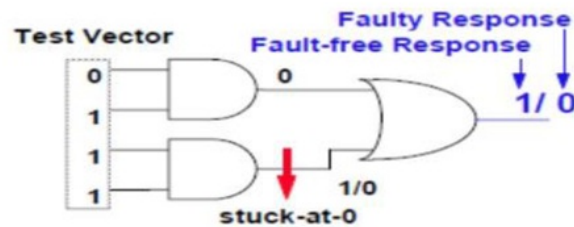


Figure 2 - Stuck-at fault example

(Slideshare.net, 2014)

Good example. There are other scenarios to consider.

Why are single stuck at fault models used? Stuck at fault models are used as complexity of testing is greatly reduced as an array of physical defects are able to be modeled using one logical stuck at fault. The stuck at fault is independent technology, it can be utilized for TTL, ECL, CMOS and more.

2.2. Electro Static Discharge (ESD)

ESD is the discharge of static electricity between two objects when they become in contact with each other. ESD examples include the shock people receive when walking across a carpet and touches metal doorknobs. A severe case of ESD is a lightning bolt. The majority of ESD shocks are inoffensive, ESD can be a serious problem for industrial environments.

For ESD, a build of electrostatic charge is required for it to take place. Electrostatic charge is built up between frictions of two materials rubbing together where one of the materials becomes positively charged whilst the other becomes negatively charged. When the material with the positive charge comes into contact with a suitable material, electrostatic charge is transferred thus, an ESD event.

Heat caused from ESD can melt electronic components causing the electronic circuit to fail however, a latent defect of ESD involves an ESD event on a device but the device continues to operate as normal. This is hard to detect and ominously shortens the lifespan of the device. (whatis.techtarget.com, 2011)

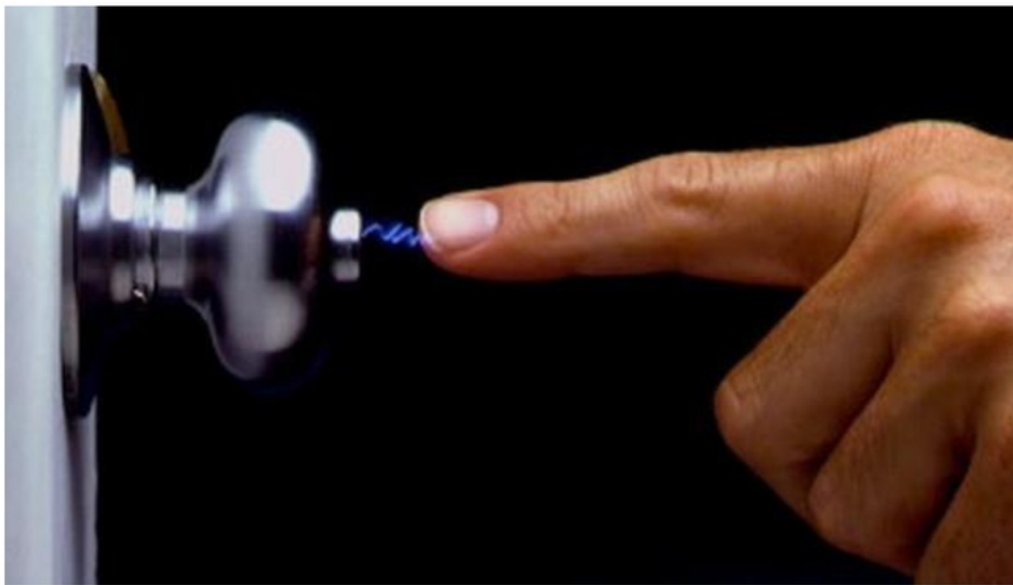


Figure 3 - Electrostatic Discharge

(we-online.com, 2018)

Many synthetic materials like carpets, clothes and furniture are good insulators especially when in dry weather. A charge will be built up on the materials and not have the capability to flow away due to leakage but will be discharged by coming into contact with another material.

The materials generate such a high static voltage energy when rubbed together such as when walking over vinyl floor or moving about on a plastic chair. The way our bodies wearing these synthetic materials move, rubbing, causes humans to receive a mild electric shock causing us discomfort as the material worn charges to a high potential compared to other objects.

Minimal discomfort occurs to people but ESD significantly damages modern day electronic components. Figure 6 shows a the electro static voltage magnitude that can be obtained from sources of ESD voltage

(Phil. Storr, 1998)

Source of ESD voltage	Typical Electro Static Voltage in KV at a relative humidity of 20%
Walking across Vinyl floor	12 KV
Walking across synthetic carpet	35 KV
Opening a plastic bag	20 KV
Arising from a foam cushion	18 KV
Sliding plastic box on a carpet	18 KV
Removing plastic tape for a PC board	12 KV
Removing shrink film from PC board	12 KV
Triggering a vacuum solder remover	8 KV
Spraying aerosol freezer spray	15 KV

Figure 4 - Magnitude of Voltage Developed in ESD

Modern integrated circuits are gradually using more silicon, internal devices are getting much smaller thus meaning that the amount of voltage and current from ESD is becoming less, creating a larger window for ESD damage.

Electro static discharge does not always result in ‘Instant Death’ of an electronic device. ‘Latent Damage’ is common where as it only appears in degradation of performance and intermittent failing of the device.

Catastrophic failure is known as ‘Instant Death’ of an electronic device. This happens 10 to 20% of electronic devices that receive an ESD event. This type of damage is preferred as the electronic device will be known to completely fail rather than a ‘Latent Damaged’ device of intermittently working or downgrading of performance over time. Figure 7 shows the ESD susceptibility of electronic devices

Device type	ESD susceptibility
Bipolar transistors	380 to 7000 Volt
CMOS logic devices	250 to 3000 Volt
EPROM devices	100 volt
Film resistors	300 to 3000 Volt
TTL logic devices	1000 to 2500 Volt
Microprocessor chips	as low as 10 Volt

Table of examples are good.

Figure 5 - Figures of ESD Susceptibility of Devices

(Phil. Storr, 1998)

2.3. ESD Damage and Prevention Methods.

The following figures will show ESD events and the damage caused to electronic components

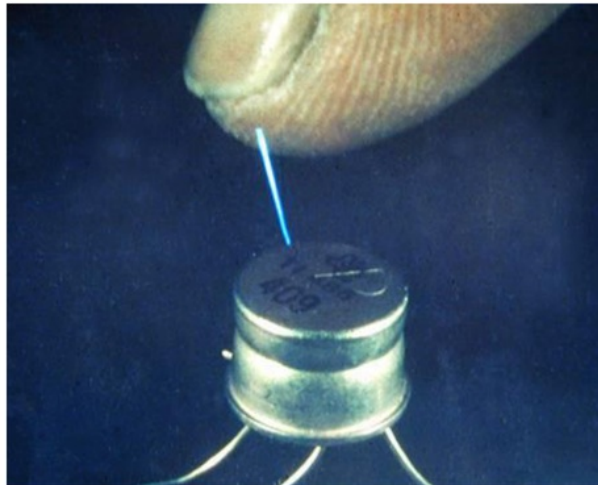


Figure 6 - ESD Event with a Transistor

Figure 8 shows a real event of a technician causing an ESD event with an electronic component

(DESCO, 2013)

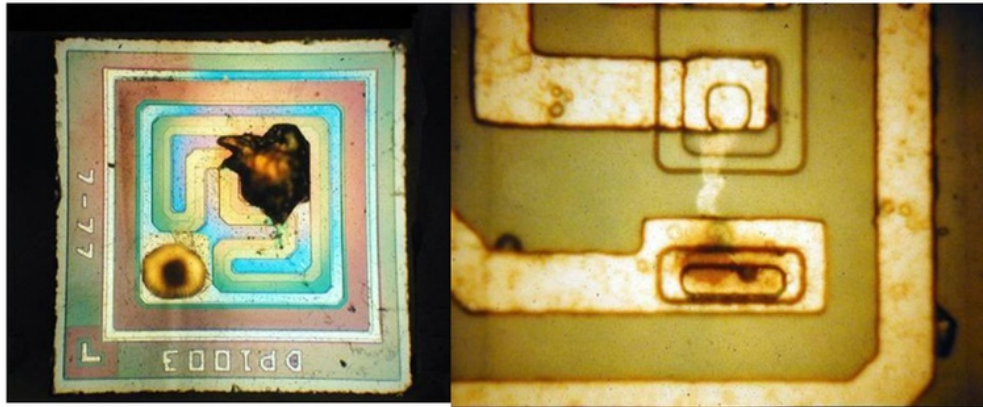


Figure 7 - ESD Damage of the Transistor Bond

Figure 9 shows the extent of the damage caused on the transistor from the ESD event. The bond within the transistor has melted from the inductive surge from the ESD. ESD arcing of this type is common within testing of integrated circuits.

(DESCO, 2013)

Methods of ESD protection include ESD sensitive devices using the methods of grounding, isolation and neutralization. All methods of protection is obtained through connecting everything involved within the test to a known grounding point. This includes the work area, operator and the electronics itself. This can be done by using clipping to a grounding mat but it is important to test for a good earth prior to testing.

Typical static sensitive equipment must be handled in safe working areas and prevention methods applied. These include, Bench top ground points, ESD grounding mat. Technician grounding strap to ground the technician to the grounding point, protective clothing and hair protection and ESD safe floor area and Ionized air blowers.

(Phil. Storr, 1998)

Figure 10 and Figure 11 show typical ESD prevention methods of safe working area and technician's wrist earth point.

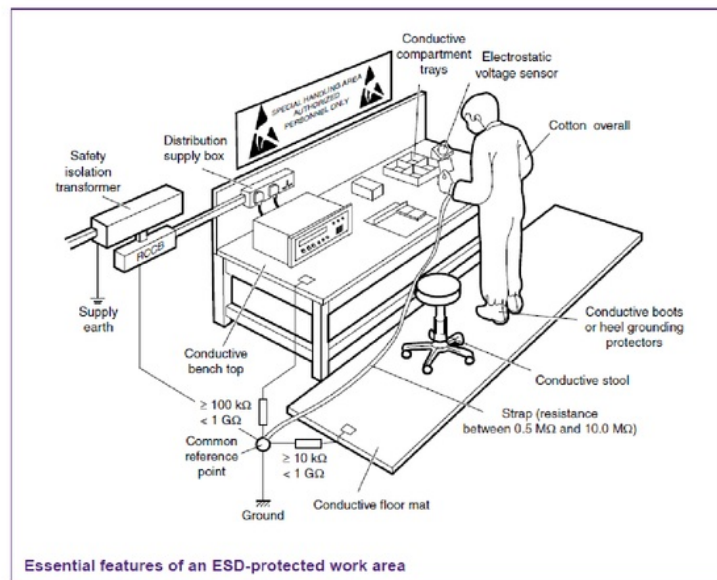


Figure 8 - ESD Protected work area

(electronics.stackexchange.com, 2011)



Figure 9 - Technicians Earthing Strap

(instructables.com, 2009)

Research is good and using relevant sources. Bridgin

3. MultiSIM Design

Figure 12 shows the given test circuit. The circuit has 6 inputs A-F and 4 logic gates. The gate include a 3 input and gate, 2 input and gate and 2 or gates.

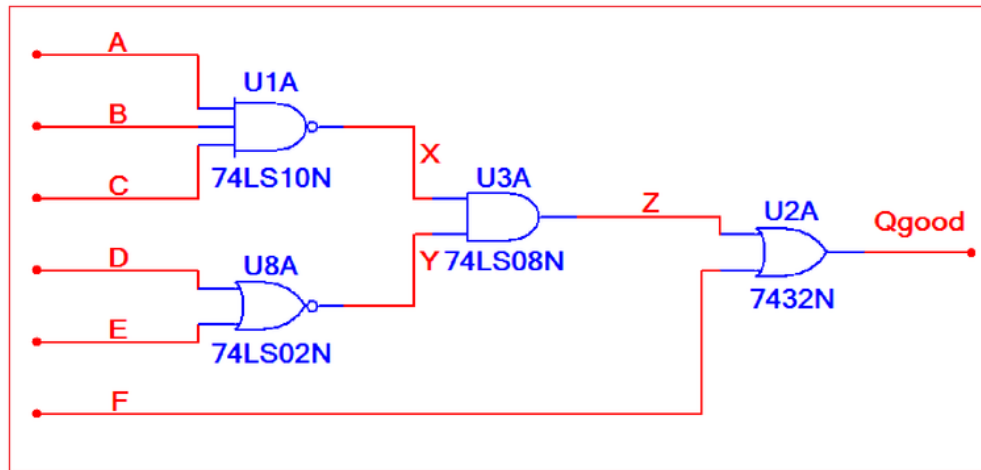


Figure 10 - MultiSim Design

The test circuit shown was to be modelled using 'piece wise' tools to replicate the test circuit where the final outputs of each circuit can be compared using a Xor gate comparator to identify any differences between the two circuits on a logic analyzer.

3.1. Fault Modelling

To begin the modelling of the test circuit, the test vectors of each gate had to be determined to produce an overall test table of test vectors. The number of test vectors of each gate was determined through following the formula of $M+1$ where M represents that number of inputs of the gate. Therefore, for the 3 input and gate, 4 test vectors were obtained.

Test No	A	B	C	X
1	1	1	1	0
2	0	1	1	1
3	1	0	1	1
4	1	1	0	1

This is NAND gate model

Figure 11 - 3 Input and Gate Test Vectors

Figure 14 shows the truth table of test vectors for the 3 input and gate. As it can be seen, there are 4 test vectors as determined from the formula

Test No	D	E	Y
5	0	0	1
6	1	0	0
7	0	1	0

This is 2 i/p NOR gate model

Figure 12 - 2 Input and Gate Test Vectors

Figure 14 shows the truth table of test vectors for the 3 input and gate. Using the formula $M+1$, 3 test vectors were determined for the 2 input and gate.

Test No	X	Y	Z
8	1	1	1
9	0	1	0
10	1	1	0

This is AND gate model

Figure 13 - 2 Input or Gate Test Vectors

Figure 15 shows the truth table of test vectors for the 2 input or gate. 3 test vectors were determined using the formula $M+1$.

Test No	F	Z	Qgood
11	0	0	0
12	1	0	1
13	0	1	1

Figure 14 - 2nd 2 Input or Gate Test Vectors

Figure 16 shows the truth table of test vectors for the second 2 input or gate, taking the total test numbers to 13. All four of the truth tables were compiled into one test vector table.

Some confusion with gates here. The tables are correct but the names are not. Something to be careful of in the future.

Test No	A	B	C	D	E	F	Qgood	HEX	X	Y	Z	Fault
1	1	1	1	0	0	0	0	70	0	1	0	A+B+C SA0
2	0	1	1	0	0	0	1	31	1	1	1	X SA1
3	1	0	1	0	0	0	1	51	1	1	1	A SA1
4	1	1	0	0	0	0	1	61	1	1	1	B SA1
5	0*	0*	0*	0	0	0	1	01	1	1	1	C SA1
6	0*	0*	0*	1	0	0	0	08	1	0	0	D+E SA1
7	0*	0*	0*	0	1	0	0	04	1	0	0	X+Y SA0, Z SA0
8	0*	0*	0*	0	0	0	1	01	1	1	1	D SA0
9	1	1	1	0	0	0	0	70	0	1	0	E SA0
10	0*	0*	0*	1**	1**	0	0	0C	1	0	0	X+Y SA0
11	1	1	1	1**	1**	0	0	7C	0*	0*	0	X SA1
12	1	1	1	1**	1**	1	1	7F	0*	0*	0	Y SA1
13	0*	0*	0*	0	0	0	1	01	1	1	1	F+Z SA1

Figure 15 – Test table for the Test Circuit

Figure 17 shows the complete test table with all test vectors from all four gates with the binary inputted.

To minimize the amount of testing, the test table vectors can be reduced from 13 using these through removing redundant tests using the hex values of the tests and by taking out values that are equal to prevent repetitive testing. Tests 13, 9 and 8 were removed for duplicate hex values of 01, and 70

A reduced test table was then produced and is shown in Figure 18

Test No	A	B	C	D	E	F	Qgood	HEX	X	Y	Z	Fault
1	1	1	1	0	0	0	0	70	0	1	0	A+B+C SA0
2	0	1	1	0	0	0	1	31	1	1	1	X SA1
3	1	0	1	0	0	0	1	51	1	1	1	A SA1
4	1	1	0	0	0	0	1	61	1	1	1	B SA1
6	0*	0*	0*	1	0	0	0	08	1	0	0	C SA1
7	0*	0*	0*	0	1	0	0	04	1	0	0	D SA0
9	1	1	1	1**	1**	0	0	7C	0*	0*	0	E SA0
10	1	1	1	1**	1**	1	1	7F	0*	0*	0	F+Z SA1

Figure 16 - Reduced Test Table

Test No	A	B	C	D	E	F	Qgood	HEX	X	Y	Z	Fault
1	1	1	1	0	0	0	0	70	0	1	0	A+B+C SA0
2	0	1	1	0	0	0	1	31	1	1	1	X SA1
3	1	0	1	0	0	0	1	51	1	1	1	A SA1
4	1	1	0	0	0	0	1	61	1	1	1	B SA1
5	0*	0*	0*	0	0	0	1	01	1	1	1	C SA1
6	0*	0*	0*	1	0	0	0	08	1	0	0	D+E SA1
7	0*	0*	0*	0	1	0	0	04	1	0	0	X+Y SA0, Z SA0
8	0*	0*	0*	1**	1**	0	0	0C	1	0	0	D SA0
9	1	1	1	1**	1**	0	0	7C	0*	0*	0	E SA0
10	1	1	1	1**	1**	1	1	7F	0*	0*	0	Y SA1

Good reduction method used here.

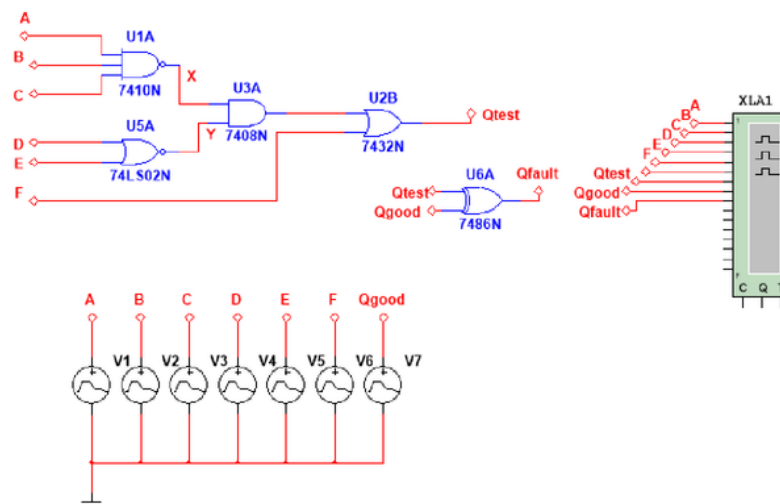
Figure 17 - Further Reduced Test Table

Figure 19 shows the final reduce test table of all test vectors that can be applied to the test circuit after the reduction methods. Two rules can be applied to the test table that can further reduce the amount of test vectors in the test table. These rules are:

* Indicates all or at least 1 must be a logic 0 for correct result.

** Indicates all or at least 1 must be a logic 1 for correct result.

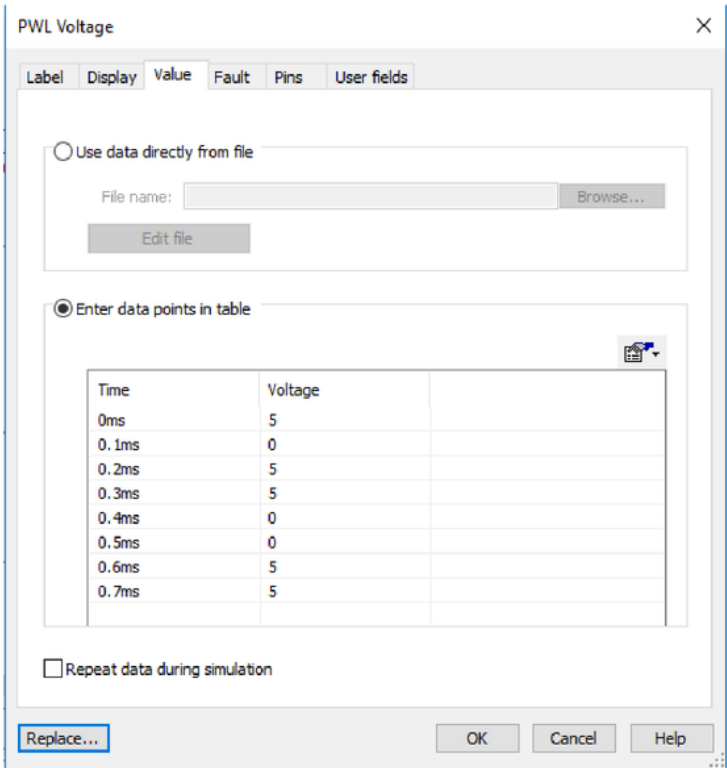
The manipulation process of the rules is used to alter the logic condition to further reduce the number of test vectors. Test 5 was altered to match test 4, A and B were changed to a 1 and Test 8 is altered to match test 6 where E was altered to logic 0. Removing these two tests produced a final further reduced test table.



The tables are correct despite the wrong gate identity. Reduction methods are good and demonstrates attention to detail.

Figure 18 - MultiSim simulation of Test circuit against Piece wise tools

Figure 20 shows the two test circuits on MultiSim, the given test circuit and the test circuit modelling using Piece wise tools. The two final outputs of the logic circuits are compared using a Xor gate. The logic is then monitored through using a logic analyzer.



Good example of the PLV which shows innovation.

Figure 19 - Test vectors inputted into Piece Wise Tool

Figure 21 shows how the test vector was inputted into each piece wise tool. The screenshot was taken from ‘A’ where the voltage replicates the binary from the truth table. 5 volts represents 1 and 0 volts represents 0. Time represents the tests that are applied 0.1ms increments. Three faults were picked from the test table from different gates and applied in the simulation.

4. Analysis of NI MultiSim Design

Testing the MultiSim Piece Wise tool model against the good test circuit was carried out on the MultiSim circuitry shown in Figure 20. Three test vectors of the further reduced test vector table were selected to apply single stuck at faults to the test circuit model for comparison with the good test circuit.

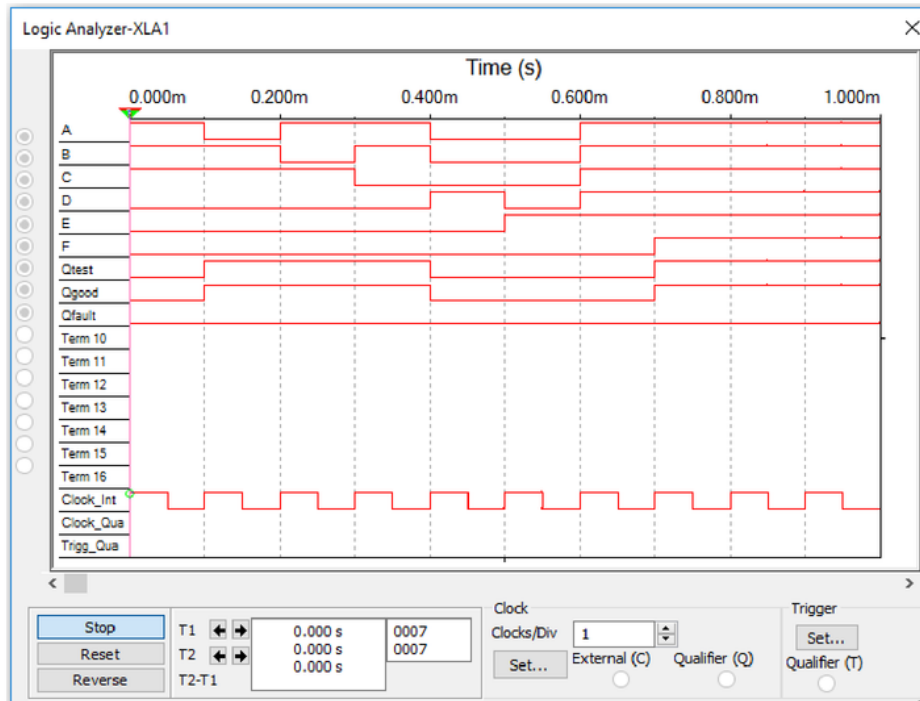


Figure 20 - No Faults MultiSim

Figure 22 shows the logic analyzer result of a test between the good test circuit and the modelled circuit with no faults fabricated. The logic analyzer shows the 6 bits A-F, the output of the test circuit QTest, the output of the good circuit Qgood. Qfault is the comparator, comparing the two circuits output to detect a fault.

4.1. MultiSim Test 1 – A Stuck at 0

The first test carried out was to keep A stuck at 0. This was done by an interactive digital constant set to 0 and connected in place of the piece wise tool as shown in Figure 24.

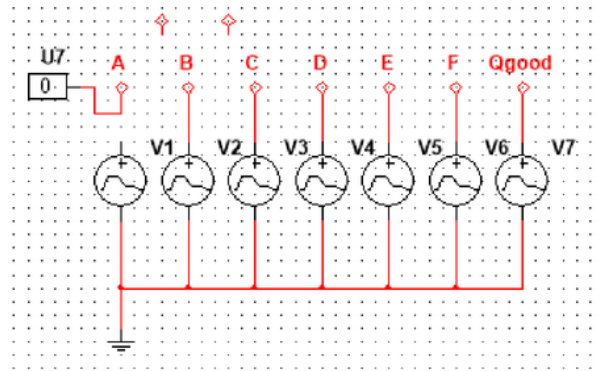


Figure 21 - Test 1, A stuck at 0

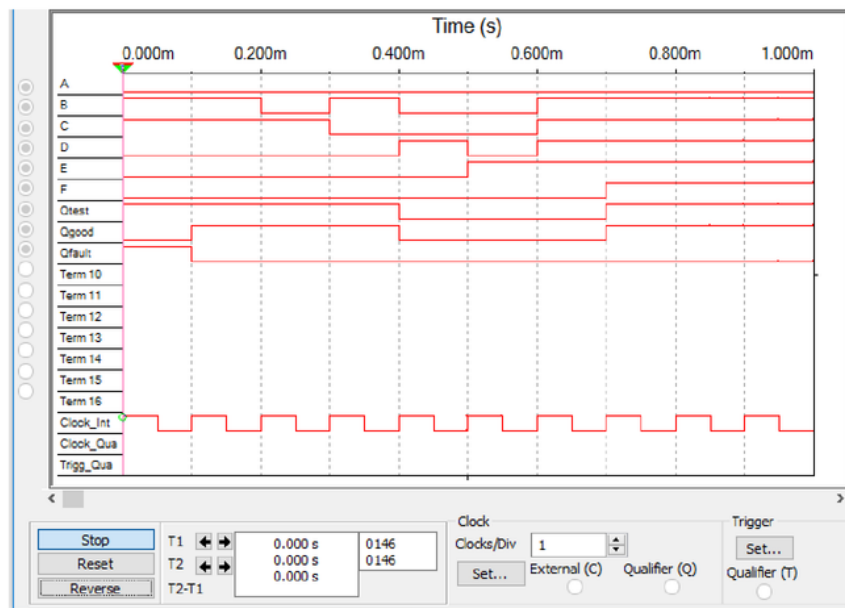


Figure 22 – Test 1, A Stuck at 0 Logic Analyzer Result

Figure 24 shows the logic analyzer result with A stuck at 0. It can be seen that A has remained low throughout the test. The logic analyzer results also show that there is a difference in output between the two circuits as the Xor comparator has saturated high when there was a difference between outputs at the start.

4.2. MultiSim Test 2 – D Stuck at 0

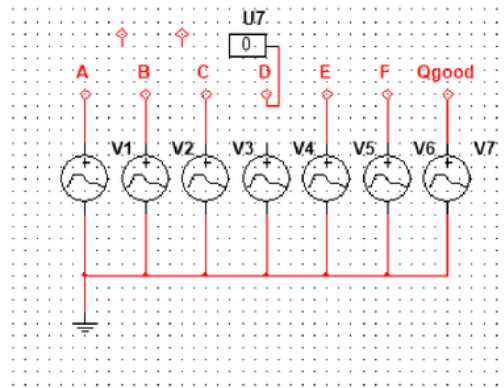


Figure 23 - Test 2, D stuck at 0

Figure 25 shows the second test fault modelled where D is stuck at zero using the interactive digital constant.

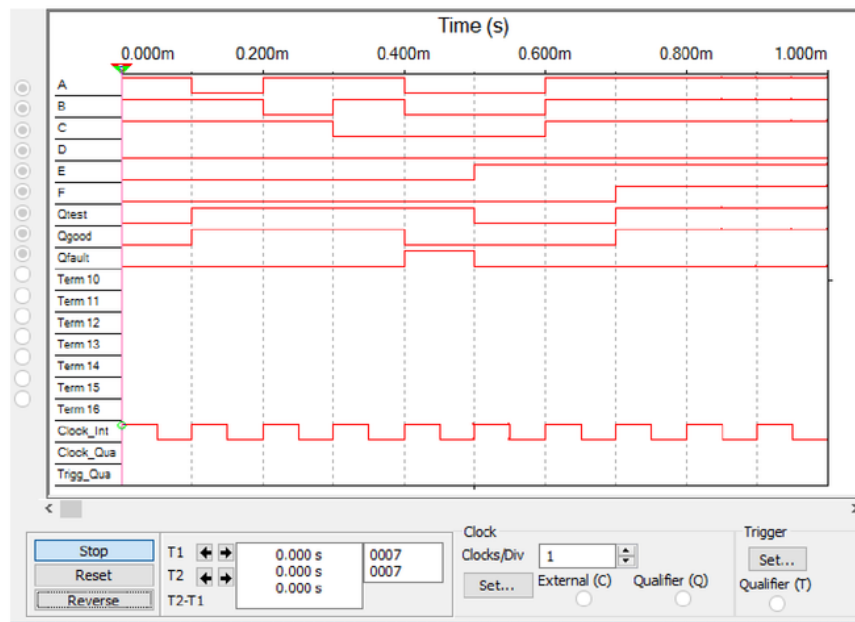


Figure 24 - Test 2, D stuck at 0 Logic analyzer results

Figure 26 shows the logic analyzer results of test 2. It can be seen that D has remained low throughout the whole test as expected. The logic analyzer results also show that a fault was detected through a dissimilar output from the two gates at the comparator within the test. This is shown in the fifth stage of the pattern by Qfault saturating high where there was a fault present.

4.3. MultiSim Test 3 – F Stuck at 0

Figure 27 shows the third test modelled where F is stuck at 0 using the interactive digital constant.

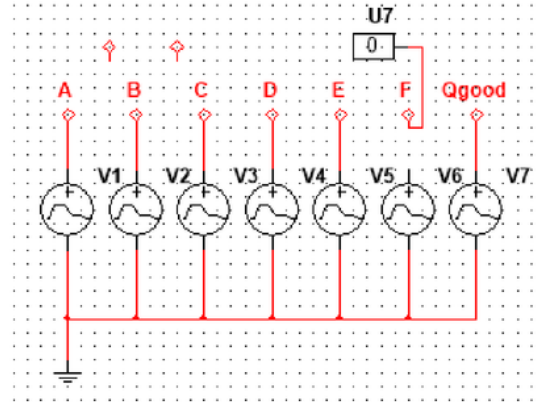


Figure 25 - Test 3, F stuck at 0

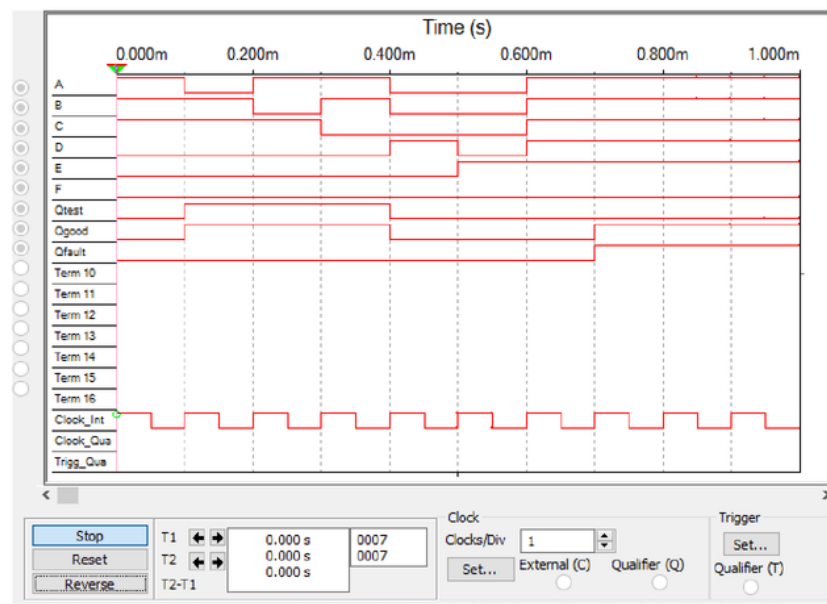


Figure 26 - Test 3, F Stuck at 0 Logic analyzer results

Figure 28 shows the logic analyzer results of test 3. It can be seen that F remains at 0 throughout the entirety of the test as expected. The logic analyzer results show that the fault was detected by the comparator at the final stage of the test where the difference between the two test circuits occurs. This is shown by Qfault going saturating high. The three fault modelled tests were a success as the expected outcome occurred during each stuck at fault applied.

5. 6Bit Pattern Generator Design and Test

5.1. 6Bit Pattern Generator Design

An alternative method to fault model would be to use a 6 bit register. Using the same binary values of the test vectors test table, the values were applied showing the present state of the register and the next state.

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	0	0	0

Figure 27 – Present and Next State values of the 6 bit register

Figure 29 shows the present and next state values of A to F, these values will be utilized as inputs and outputs to the positive edged triggered D type flip flops of the 6 bit register. To produce the Boolean expressions for the input and outputs of the register, karnaugh mapping is required to derive the expressions for each bit.

	ABC							
DEF	00	01	11	10	100	101	111	110
00	0	0	1	0	0	1	0	0
01	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0
10	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

Figure 28 - A+ Karnaugh Map

Figure 30 shows the karnaugh map of A+. The Boolean expression derived from the map is expressed as:

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

	ABC							
DEF	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

Figure 29 - B+ Karnaugh Map

Figure 31 shows the karnaugh map of B+. The Boolean expression derived from the map is expressed as:

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

	ABC							
DEF	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

Figure 30 - C+ Karnaugh Map

Figure 32 shows the karnaugh map of C+. The Boolean expression derived from the map is expressed as:

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

	ABC							
DEF	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

Figure 31 - D+ Karnaugh Map

Figure 33 shows the karnaugh map of D+. The Boolean expression derived from the map is expressed as:

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

	ABC							
DEF	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

Figure 32 - E+ Karnaugh Map

Figure 34 shows the karnaugh map of E+. The Boolean expression derived from the map is expressed as:

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

	ABC							
DEF	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

Figure 33 - F+ Karnaugh Map

Good use of K-maps and use of the mirror method for grouping.

Figure 35 shows the karnaugh map of C+. The Boolean expression derived from the map is expressed as:

$$F+ = A.B.C.D.E.NF$$

This is for F+ I think! The figure numbering are offset.

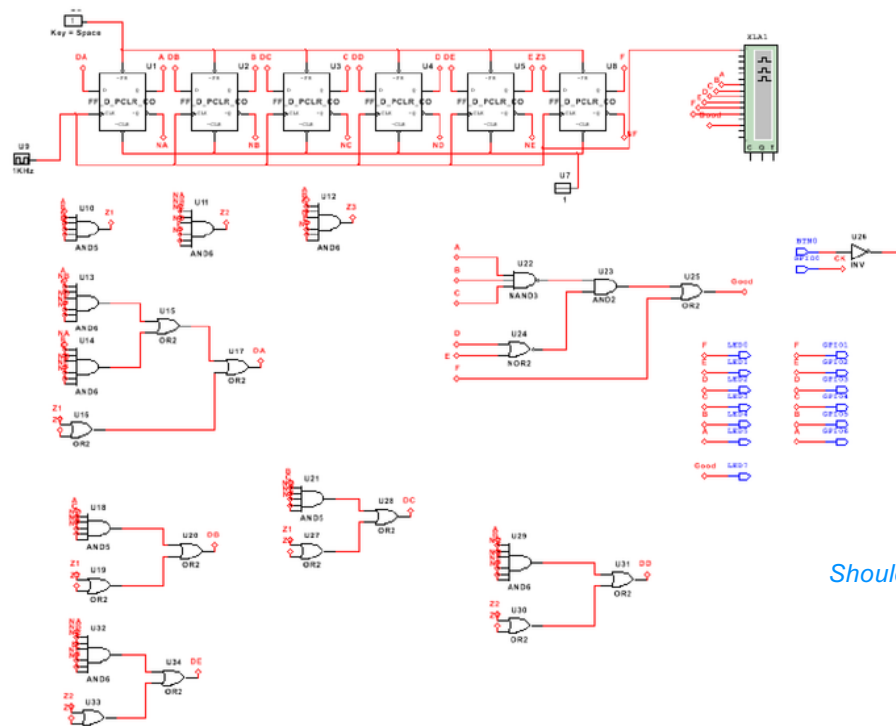


Figure 34 - 6 Bit Register in MultSim

Should include the x-or gate.

Figure 36 shows the alternative 6 bit register circuit on MultiSim.

At the top of the circuit the 6 bit register can be seen that includes 6 d type flip flops. The logic circuits were produced from the Boolean expressions of the karnaugh maps. The output of each logic circuit for e.g. A+, B+, C+ etc. has the output of DA, DB and DC etc. and is connected to the individual inputs of the 6 flip flops, where A,B,C etc. is connected to the output of each flip flop. Z1, Z2 and Z3 represent Boolean algebra that is common to most logic circuits where Z3 is also used as DF.

The output from the 6bit register and the test circuit is compared using the logic analyzer. Inputs A-F are also inputted into the logic analyzer to observe the input logic.

5.2. 6Bit Pattern Generator Testing

The logic analyzer results from testing the 6 bit pattern generator is shown in Figure 37

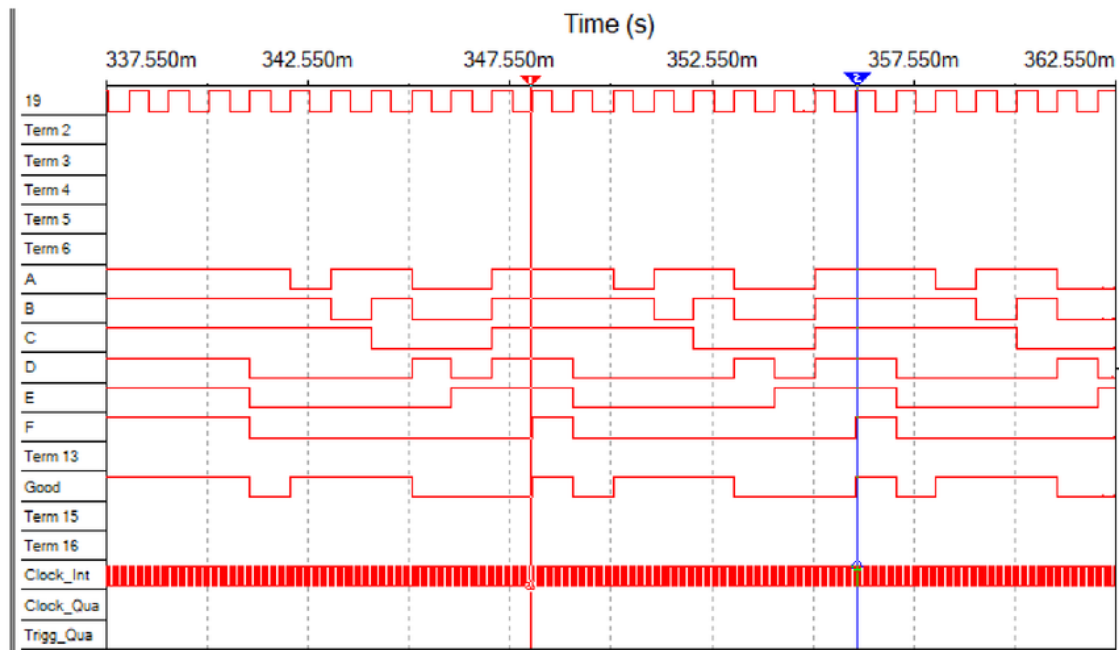


Figure 35 - 6 Bit pattern generator Logic analyzer Results

The two cursors represent a full 6 bit pattern where after each positive pulse from the clock presents the next state of the pattern generator. The first state shows all of the inputs A-F as logic 1.

*Test is good
and shows the
test vectors
between the
cursors.*

The second state shows the binary as 111 000, third state shows the binary as 011 000, fourth state is shown as the binary value is 101 000, the fifth state is shown as 110 000, sixth state is shown as 000 100, the seventh state shown as 000 010 and the final eighth state is shown as 111 110.

The 6 bit pattern generated replicates the combination logic within the state diagram shown in Figure 29

Good analysis here which shows the 8 test vectors. The figures are offset at this stage of report.

6. NI LabVIEW Design and Practical Build

A prototype circuit was designed and built practically using a NI ELVIS prototyping board and simulation software NI LabVIEW. The NI LabVIEW design is shown in Figure 38.

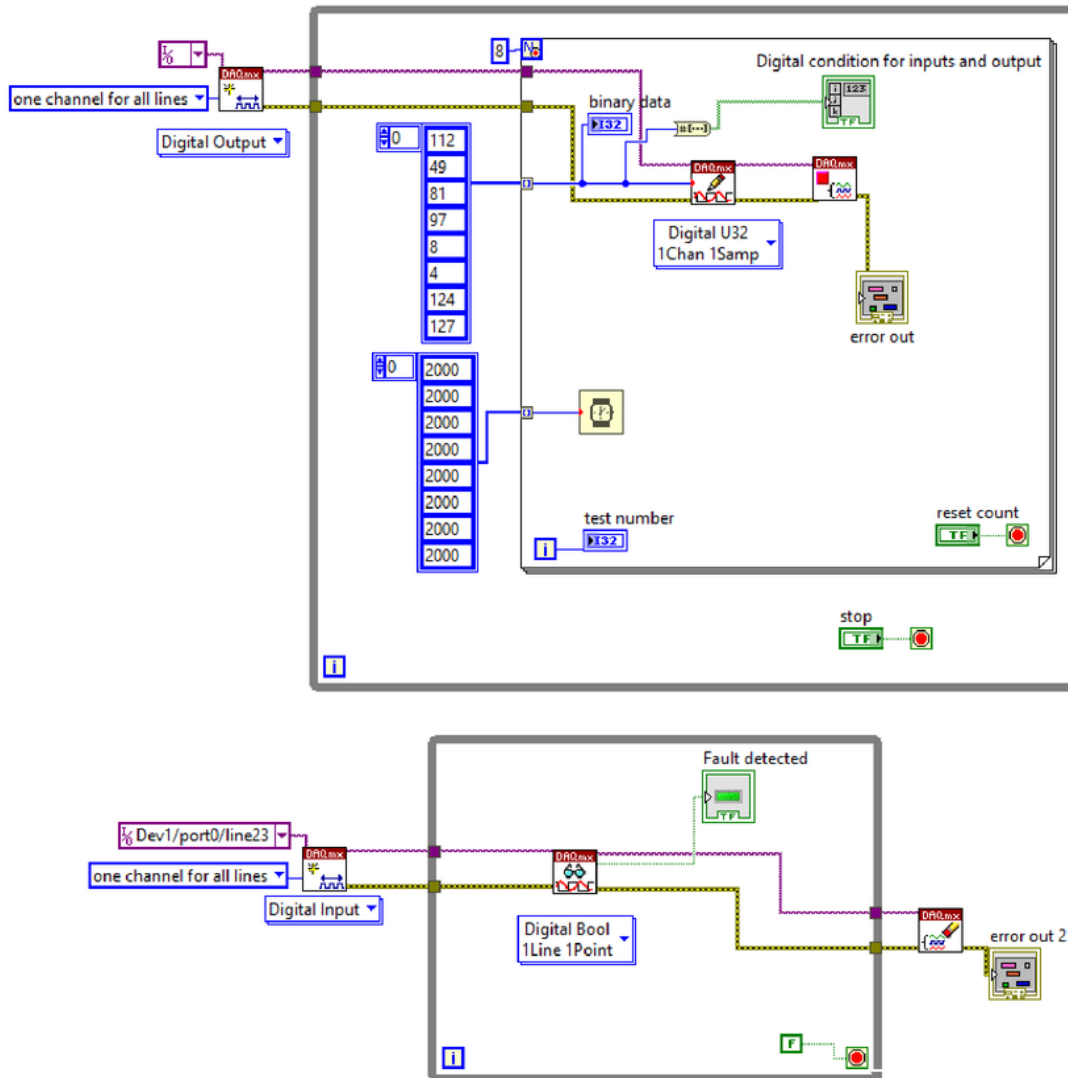


Figure 36 - NI LabVIEW 6Bit Pattern Generator Design

The NI LabVIEW 6 bit pattern generator consists of two while loops, one for the writing data loop and another for the reading data loop, a loop count is also used within the top writing circuit to count the 8 stages of the pattern generator.

A DAQmx write and DAQmx read blocks are used to write and read data of the 6 bit pattern generator where DAQmx digital input and DAQmx digital output blocks are used to transmit data to the practical circuit on the NI ELVIS prototyping board. A DAQmx stop block is also used to the pattern generator where a DAQmx erase block is used to delete data.

Input lines are 0-6 for A-Qtest with the output line as 23 for Qgood from the circuit on the NI ELVIS. Qtest input from the test circuit will always be healthy where a or gate will be used as a comparator to compare the two test circuit outputs for a fault. Any differences in output will cause the xor gate to flag a fault.

Two 8 bit arrays are utilized, one array connected to the DAQmx write block contains the number pattern generated whereas the array below is a timer for each test state where each state will last 2000ms.

There is also a test counter and stop buttons applied as well as indication of fault detection and binary values of each test.

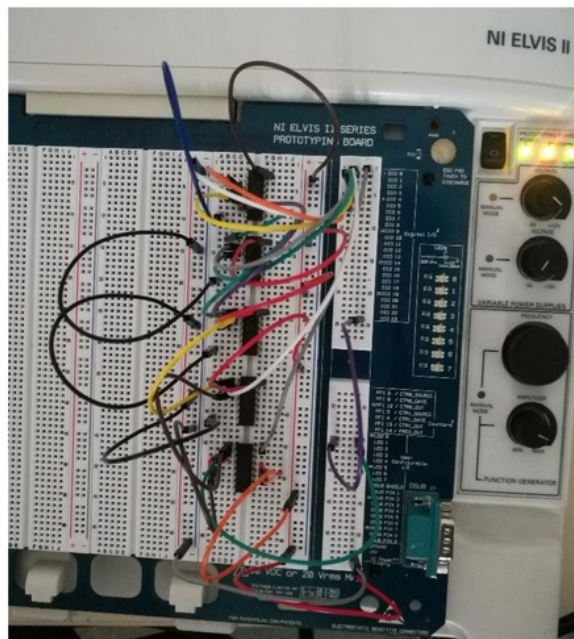


Figure 37 - Test Circuit built on NI ELVIS Board

Figure 39 shows the practical build on the NI ELVIS of the test circuit using the four test gates.

6. Analysis of LabVIEW Practical Build

The 6bit pattern generator was set to run where the circuit operated correctly through the pattern with no faults being detected.

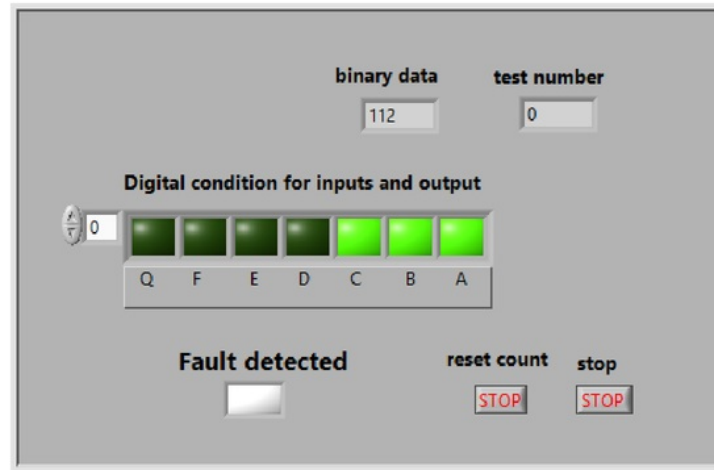


Figure 38 - LabVIEW Test 0

Figure 40 shows the result of test 0 on the LabVIEW front panel. Binary data of the test was shown as 112 where A, B and C are at logic 1 where D E F and Q are at logic 0.

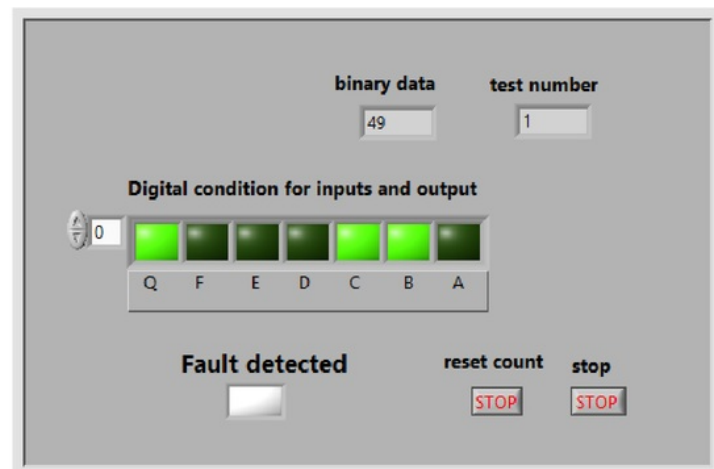


Figure 39 - LabVIEW Test 1

Figure 41 shows the result of test 1 on LabVIEW. Test 1s' binary data was 49 where B, C and Q are logic 1 and A, D, E and F are at logic 0

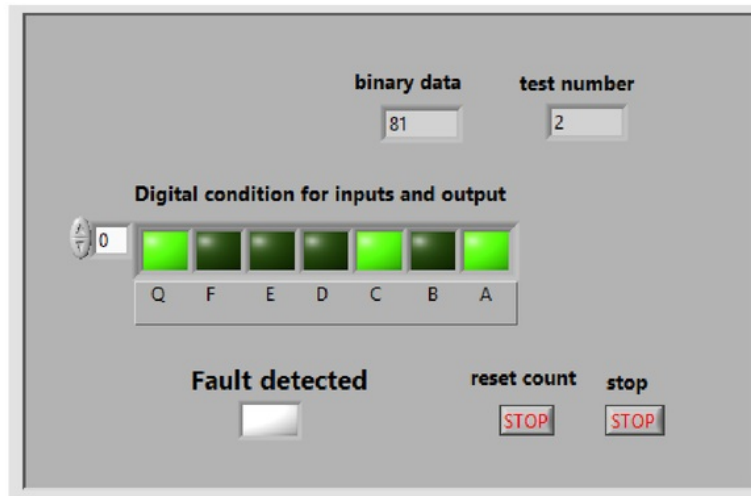


Figure 40 - LabVIEW Test 2

Figure 42 shows the results of test 2. Binary data of test 2 is 81 where A, C and Q are at logic 1 and B, D, E and F are at logic 0 with no fault detected.

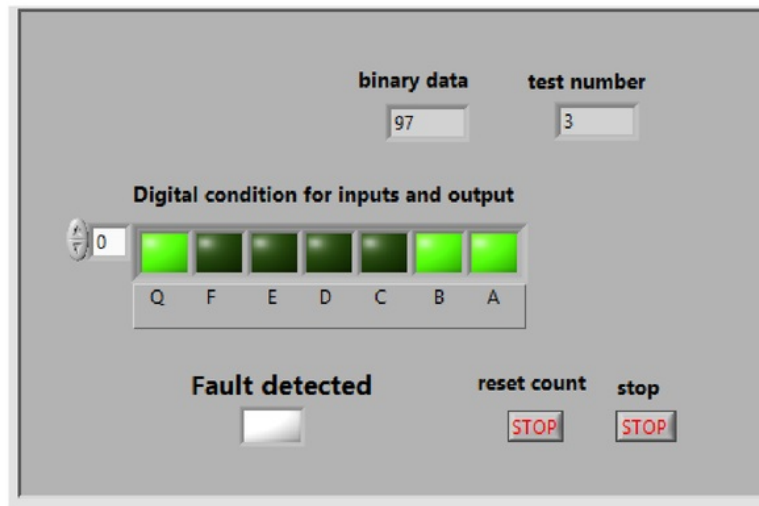


Figure 41 - LabVIEW Test 3

Figure 43 shows the results of test 3. Binary data of test 3 is 97 where logic inputs and outputs are shown as A, B and Q at logic 1 and C, D, E and F are at logic 1 with no faults detected.

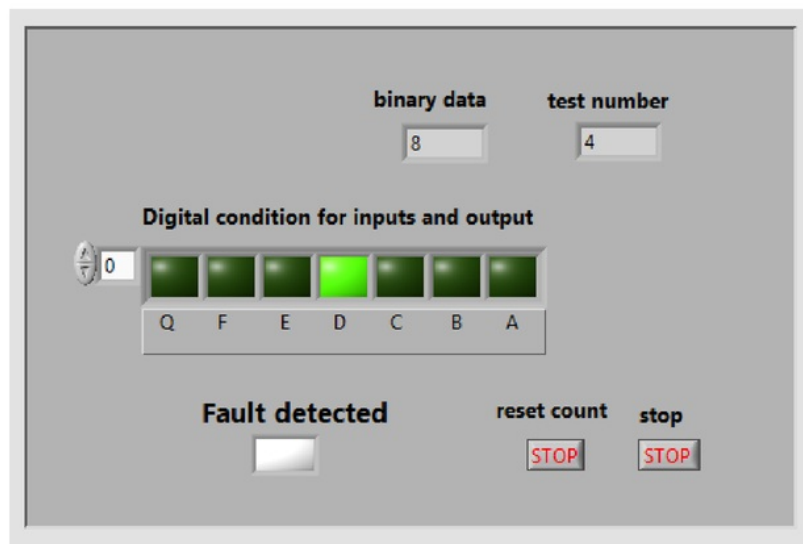


Figure 42 - LabVIEW Test 4

Figure 44 shows the results of test 4. Test 4 has a binary data of 8 where only D is at logic 1 and A, B, C, E, F and Q at logic 0 with no fault detected.

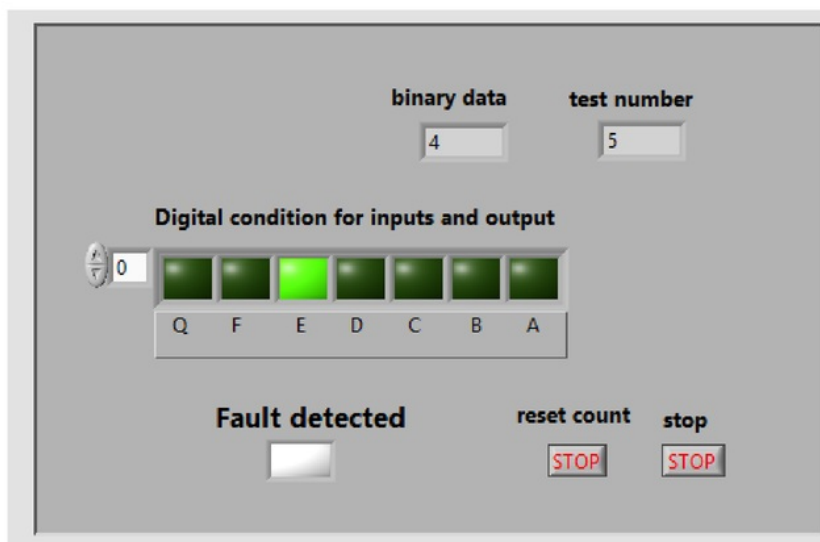


Figure 43 - LabVIEW Test 5

Figure 45 shows the result of test 5. Test 5 has a binary data of 4 where only E is at logic 1 where all A, B, C, D, F and Q are logic 0 with no fault indicated.

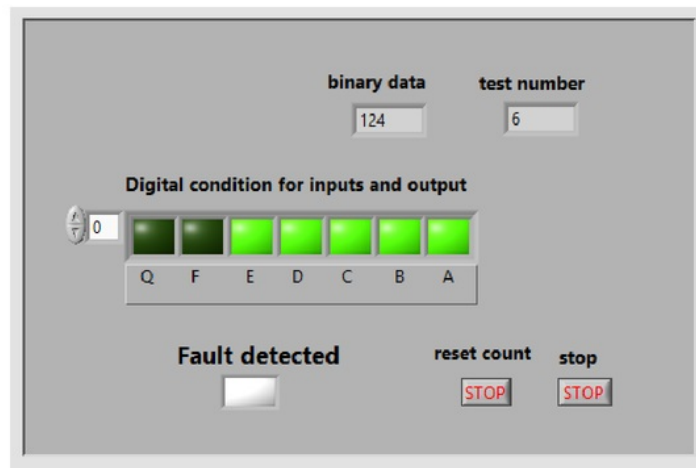


Figure 44 - LabVIEW Test 6

Figure 46 shows the result of test 7. Test 7 has binary data of 124 where A, B, C, D and E are at logic 1 and F and Q are at logic 0 with no faults detected.

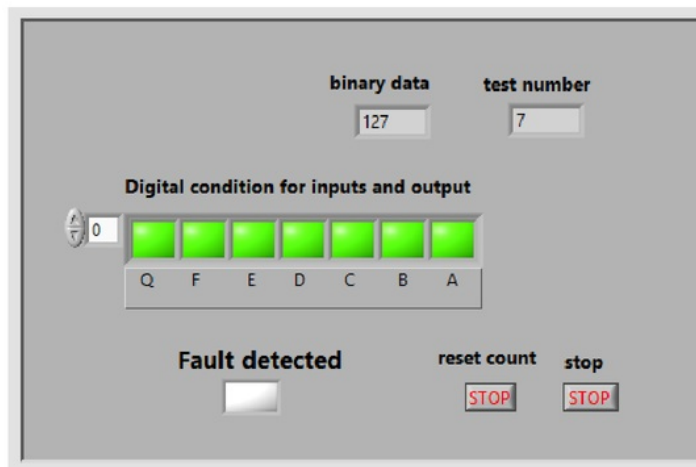


Figure 45 - LabVIEW test 7

Figure 47 shows the final LabVIEW test. Test 7 shows a total binary count of 127 with all digital conditions at logic 1 with no faults detected. Figure 40 to Figure 47 show the entire pattern of the 6 bit pattern generator operating with no faults.

6.2. Implementing Faults

Identical Stuck at faults used in the Piece Wise tool modelled circuit were implemented on to the practical circuit on the NI ELVIS board. These faults were, Test 0 – A stuck at 0, Test 4 – D stuck at 0 and Test 6 – F stuck at 0.

6.2.1. Practical Test 1 – A Stuck at 0

Figure 49 shows the LabVIEW panel under test 1 where Figure 48 shows the NI ELVIS circuit under test 1

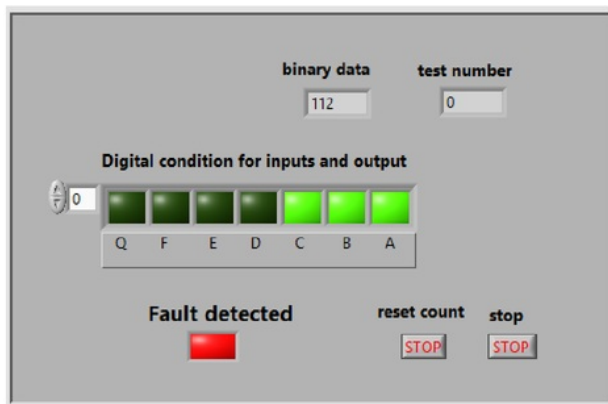


Figure 47 - Fault 1, A Stuck at 0

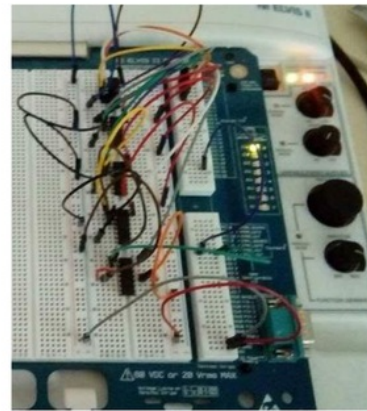


Figure 46 - NI ELVIS board at Test 1

Test 1 was carried out by connecting A input of the and gate to 0v on the NI ELVIS prototyping board. The pattern generator was run where the pattern operated through its sequence until a difference between the good circuit and test circuit was found where the output LED illuminated indicating a fault.

The fault was detected in test 0. Within test 0, A is supposed to be logic 1 as shown in the LabVIEW panel result but as A is stuck at 0V on the NI ELVIS board, a difference in output between the two circuits was detected by the comparator. The fault LED illuminated red when a fault was detected.

6.2.2. Practical Test 2 – D Stuck at 0

Figure 50 shows the LabVIEW panel under test 1 where Figure 51 shows the NI ELVIS circuit under test 1

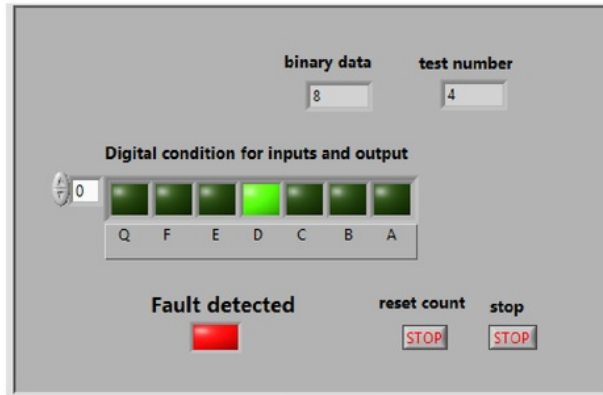


Figure 48 - Fault 2, D Stuck at 0

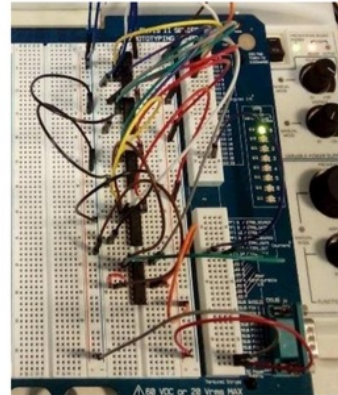


Figure 49 - NI ELVIS Board at Test 2

Test 2 was implemented by connecting the D input to 0V on the NI ELVIS prototyping board. The pattern generator was started where the system sequentially run through each state where the fault detected LED illuminated red when a discrepancy occurred between the outputs of the two circuits. The Fault occurred in test 4 where D was supposed to be at logic 1 as shown in the LabVIEW results but as D was connected to 0V on the NI ELVIS board, a difference in output was detected by the comparator therefore a fault was detected between the two circuits.

A Fault LED on the NI ELVIS board also illuminated when a difference occurred between the two circuits as shown in Figure 51

6.2.3. Practical Test 3 – F Stuck at 0

Figure 50 shows the LabVIEW panel under test 1 where Figure 51 shows the NI ELVIS circuit under test 1

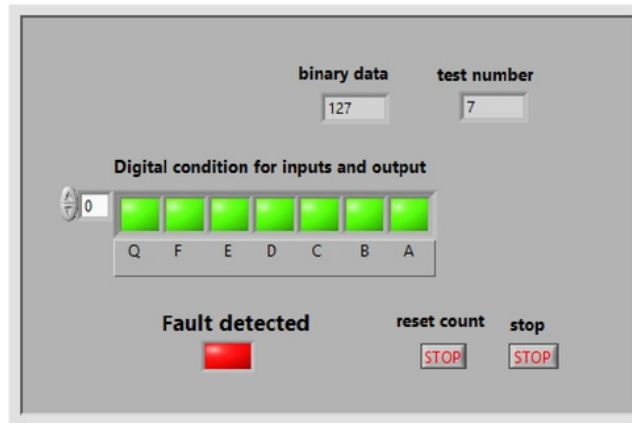


Figure 51 - Fault 3, F Stuck at 0

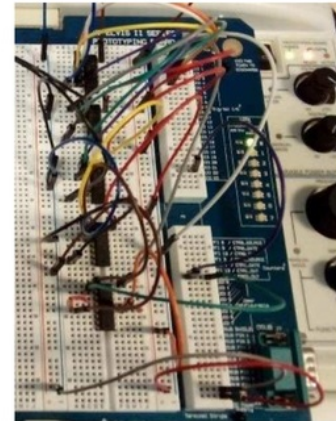


Figure 50 – NI ELVIS board at Test 3

Test 3 was implemented by connecting the F input to 0V on the NI ELVIS prototyping board. With the pattern generator started, the system operated through the states where a fault was detected when in Test 7 where F supposed to be logic 1 condition for both test circuits. But as F was connected to 0V on the NI ELVIS circuit and is shown at logic 1 one the LabVIEW circuit, a fault was detected by the comparator and indicated a fault between the two circuits.

A Fault LED on the NI ELVIS board also illuminated when a difference occurred between the two circuits as shown in Figure 52

Good analysis using LabVIEW and DAQMX with reference to the table of test vectors.

7. Conclusion

To conclude this assignment, the aims and objectives set out within the introduction have been met as background research was conducted into IC Fabrication and the utilization of single stuck at faults and fault modelling to use in testing a simple industrial control process PCB design.

An investigation of Electro Static Discharge was carried out to learn about its characteristics, impact on electronics and counter measures that are utilized throughout the industry to protect electronic boards from an ESD event.

A 6bit pattern generator was produced on MultiSim using Piece wise tools following the fault modelling process and testing against the known good test circuit. An alternative LSFR 6bit pattern generator was also produced on MultiSim to prove an alternative method could have been used through a state diagram and karnaugh mapping to produce Boolean expressions to create the logic circuit of the test circuit to compare with the known test circuit.

A practical implementation of the test circuit was built using the NI ELVIS prototyping board and a NI LabVIEW circuit was designed to represent the good test circuit with successful test results.

The Piece wise circuit and practical circuits were tested using identical test vectors of a stuck at 0, D stuck at 0 and F stuck at 0 that were determined within the fault modelling process. All testing was a success where the system operated through all 8 states of the 6bit pattern generator with no faults indicating. When three separate faults were implemented, a fault was detected by the comparator indicating that there was a difference between the two outputs of the circuit proving the fault modelling technique.

The alternative method wasn't tested using the test vectors but was proven to operate throughout the 8 state of the pattern generator.

*G
o
o
d
c
o
n
cl
u
si
o
n
th
at
id*

8. Reference List

DESCO, 2013. *Images of ESD Damage*. [Online]

Available at: <https://desco.blog/2013/07/09/images-of-esd-damage-2/>

[Accessed 02 May 2018].

electronics.stackexchange.com, 2011. *How to avoid ESD with you are working on components*. [Online]

Available at: <https://electronics.stackexchange.com/questions/22474/to-avoid-esd-when-you-are-working-on-computers-what-should-you-attach-your-wris>

[Accessed 02 May 2018].

instructables.com, 2009. *how to make a grounding wrist band*. [Online]

Available at: <http://www.instructables.com/id/How-To-Make-A-Grounding-Wrist-Band/>

[Accessed 02 May 2018].

Phil. Storr, 1998. *Phil Storrs PC Hardware book*. s.l.:s.n.

Slideshare.net, 2014. *slideshare.net*. [Online]

Available at: <https://www.slideshare.net/agyemanh/fa-31428491>

[Accessed 28 April 2018].

web.eecs.umich.edu, 2002. [Online]

Available at: <http://web.eecs.umich.edu/~mazum/F02/lectures/lec mazum3.pdf>

[Accessed 28 April 2018].

we-online.com, 2018. *How to Avoid ESD and EMI Issues*. [Online]

Available at: http://www.we-online.com/web/en/passive_components_custom_magnetics/blog_pbcmblog_detail_electronics_in_action_35519.php

[Accessed 28 April 2018].

whatis.techtarget.com, 2011. *electrostatic-discharge-ESD*. [Online]

Available at: <https://whatis.techtarget.com/definition/electrostatic-discharge-ESD>

[Accessed 28 April 2018].

9. Appendix

assignment 2 final

GRADEMARK REPORT

FINAL GRADE

GENERAL COMMENTS

Instructor

67/100

PAGE 1

PAGE 2

PAGE 3

PAGE 4

Text Comment. Good list of contents.

PAGE 5

PAGE 6

Text Comment. Good use of tables for examples.

PAGE 7

Text Comment. Good example. There are other scenarios to consider.

PAGE 8

PAGE 9

PAGE 10

Text Comment. Table of examples are good.

PAGE 11

PAGE 12

Text Comment. Research is good and using relevant sources. Bridging faults and feedback faults are other types of SA faults that are possible and could have been covered in

the research.

PAGE 13

Text Comment. This is NAND gate model

PAGE 14

Text Comment. This is 2 i/p NOR gate model

Text Comment. This is AND gate model

Text Comment. Some confusion with gates here. The tables are correct but the names are not. Something to be careful of in the future.

PAGE 15

PAGE 16

Text Comment. Good reduction method used here.

Text Comment. The tables are correct despite the wrong gate identity. Reduction methods are good and demonstrates attention to detail.

PAGE 17

Text Comment. Good example of the PLV which shows innovation.

PAGE 18

PAGE 19

PAGE 20

PAGE 21

Text Comment. Good analysis with reference to tables. All 3 fault example scenarios are shown and explained.

PAGE 22

PAGE 23

PAGE 24

Text Comment. Good use of K-maps and use of the mirror method for grouping.

Text Comment. This is for F+ I think! The figure numbering are offset.

Text Comment. Should include the x-or gate.

Text Comment. Test is good and shows the test vectors between the cursors.

Text Comment. Good analysis here which shows the 8 test vectors. The figures are offset at this stage of report.

Text Comment. Good analysis using LabVIEW and DAQMX with reference to the table of test vectors.

Text Comment. Good conclusion that identifies what has been achieved. Recommendations could have been included to emphasise the reliability and what the advantages of the fault model process offers for larger more complex circuits.

GRADING FORM: CSG-RUBIC

JOSHUA DAVIES

0

RESEARCH-INFORMED LITERATURE



Good research with good use of relevant sources IC
fabrication - 3/5 ESD - 3.5/5 SAF - 3.5/5

10/15

KNOWLEDGE AND UNDERSTANDING



Good understanding demonstrated, but there are
errors with gate identity and some offsetting with
figures. Minimum test vectors derived for design is
good - 7/10 Design of LabVIEW test vectors is
good - 7/10 Design of ECAD for PLD alternative -
6/10

20/30

ANALYSIS



Analysis is good generally, but there are some minor
issues such as gate identity and figure offset. ECAD
results of tests - 6.5/10 Comparison to theory is
identified - 3.5/5

10/15

PRACTICAL APPLICATION AND DEP.



This is good especially for LabVIEW with DAQMX.-
11/15 The resultant outputs from the logic analyser (ECAD) could have been pointed out using the cursors which would have made a difference. There are minor errors with the figures in the report, but the tests are good. Prototyping of test circuit and LabVIEW application is very good. - 7/10

18/25

SKILLS FOR PROFESSIONAL PRAC.



Good report despite some minor issues.
Deployment of appropriate software is good - 3.5/5
Report layout - 3/5 despite some errors with figures
Conclusion - 2.5/5

9/15