

SGEE6002 Assignment 1

by Jonathon Williams

GOOD RESEARCH WITH APPROPRIATE SOURCES AND CORRECT CITATIONS. PRACTICAL WORK WAS EXPLAINED WELL, HOWEVER THE ABC/DEF ITEMS ON THE K-MAPS WERE REVERSED. ANALYSIS OF RESULTS, PARTICULARLY THE LOGIC ANALYSIS GRAPH COULD HAVE BEEN MORE EXPLANATORY. REPORT WAS WELL WRITTEN AND WORD COUNT NOT EXCEEDED.

MARK AGREED


FILE	11737_JONATHON_WILLIAMS_SGEE6002_ASSIGNMENT_1_172472_528073 542.PDF (1.53M)		
TIME SUBMITTED	19-FEB-2021 01:52PM (UTC+0000)	WORD COUNT	5503
SUBMISSION ID	145097303	CHARACTER COUNT	28785

Course: BSc (Hons) Electrical and Electronic Engineering

Module: SGEE6002 Electronic Production and Test

Assignment: Fault Modeling

Name: Jonathon Williams

University number: 1708159

Date submitted: 19/02/2021

Word count: 3186

CONTENTS

Contents.....	1
Table of Figures.....	2
Table of Tables.....	3
1 Executive Summary.....	4
2 Research.....	5
2.1 IC Fabricated faults	5
2.2 Single Stuck at Faults.....	6
2.3 Electrostatic Discharge Dangers	7
2.4 Prevention methods for ESD	9
3 Design of assignment requirements	10
3.1 Stuck at fault model method	10
3.2 Derived test vectors	12
3.3 Multisim design test of CUT using Word Generator.....	14
4 Analysis of NI MultiSim results	15
5 The 6 – bit register design.....	18
6 Test analysis for the 6 bit register design	23
7 Practical build and test using Labview	26
8 CAM design and build of prototype	29
9 Conclusions and recommendations.....	33
10 Bibliography	34

TABLE OF FIGURES

Figure 2.1 - Bridging faults	5
Figure 2.2 - SSF example	6
Figure 2.3 – ESD damage.....	8
Figure 2.4 – ESD prevention	9
Figure 3.1 – CUT (Circuit Under Test)	10
Figure 3.2 – NAND gate test vectors.....	10
Figure 3.3 – NOR gate test vectors	11
Figure 3.4 – AND gate test vectors	11
Figure 3.5 – OR gate test vectors.....	11
Figure 3.6 – Simulation test with no fault condition	14
Figure 3.7 – No fault test	14
Figure 4.1 – Input A SA1 fault condition	15
Figure 4.2 – Test result for input A SA1	15
Figure 4.3 – Input C SA1 fault condition	16
Figure 4.4 – Input C SA1 fault condition	16
Figure 4.5 – Input D SA1 fault condition	17
Figure 4.6 – Input D SA1 fault condition	17
Figure 5.1 – No fault condition	21
Figure 5.2 – No fault condition	22
Figure 6.1 – Input A SA1 fault condition set	23
Figure 6.2 – Input A SA1 fault result	23
Figure 6.3 – Input C SA1 fault condition set.....	24
Figure 6.4 – Input C SA1 fault result	24
Figure 6.5 – Input D SA1 fault condition set	25
Figure 6.6 – Input D SA1 fault result.....	25
Figure 7.1 - Data pattern generator to drive inputs of CUT via DAQ outputs	26
Figure 7.2 - Result from X-OR gate sent to LabVIEW via DAQ input.....	26
Figure 7.3 – Full set of test vectors with no fault condition	27
Figure 7.4 – DAQ CU.....	27
Figure 7.5 – Fault indicated for A SA1 condition	28
Figure 7.6 – Fault indicated for C SA1 condition.....	28
Figure 7.7 – Fault indicated for D SA1 condition	28
Figure 7.8 - PCB CUT Test using LabVIEW and Elvis DAQ.....	29
Figure 7.9 – Fault indicated for A SA1 condition.....	29
Figure 8.1 - Schematic Design in Eagle.....	30
Figure 8.2 - Board design in Eagle	30
Figure 8.3 - Gerber Files	30
Figure 8.4 - Gerber View - Copper - Top	31
Figure 8.5 - Gerber View - Copper – Bottom	31
Figure 8.6 - Gerber View - Drill Holes (underlined for visibility)	31
Figure 8.7 - Gerber View – Solder Positions	32

TABLE OF TABLES

Table 2.1 - PCB Defects	6
Table 2.2 - Device ESD withstand levels from human body model	8
Table 3.1 – Initial derived table.....	12
Table 3.2 – Reduced table using redundancy rules	13
Table 3.3 – Further reduced table	13
Table 5.1 – Next state conditions table	18
Table 5.2 – Output QA	18
Table 5.3 – Output QB.....	19
Table 5.4 – Output QC.....	19
Table 5.5 - Output QD	20
Table 5.6 – Output QE.....	20
Table 5.7 – Output F.....	21

1 EXECUTIVE SUMMARY

This report explores the different types of IC fabricated faults but focuses on the most common modelling technique used, the SSF fault model. This assumes that the fault within a logic gate results in either an input or output being fixed to logic high or low. The report also investigates electrostatic discharge (ESD) - the release of static electricity when two objects come into contact. The electrostatic charge is created by two insulator surfaces rubbing together or pulling apart. As electrostatic discharge can permanently damage devices, prevention methods such as grounding wrist straps and antistatic instruments were researched. Simulation software was then used for a 4-logic gate circuit with 6-inputs and then tested using test vector tables. The results were then analysed by using a logic analyser. A 6-bit pattern generator was built by using Karnaugh maps.

Using industry-accepted software Eagle, a PCB board was design to replicate the 6-bit generator.

Using LabVIEW and Elvis DAQ, the board could be used for testing and then using the Graphical User Interface on LabView, displaying the fault.

2 RESEARCH

2.1 IC FABRICATED FAULTS

There are many methods of fabricated fault tests that can be carried out, but the preferred method is fault modelling method because input/output testing is not sufficient for manufacturing testing.

A preferred technique is Fault Modelling which enables the designer to generate known faults where the circuit can be tested against a derived table of expected faults. This can be done using ATE methods of test, which can be very fast.

This method allows analysis of the CUT (Circuit Under Test), which can be compared with the generated faults.

The most common types of fault models are:

SSF – Single Stuck at Faults

Transistor fault – open and short circuit faults

Delay faults

Memory faults

PLA faults

Bridging Faults

Transistor Stuck-On/Open Faults

Functional Faults

Memory Faults

The bridging fault is demonstrated in Figure 2.1 below:

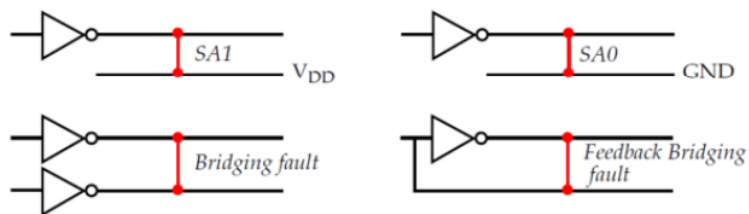


Figure 2.1 - Bridging faults

(Plusquellic, 2004)

Defects in IC's include processing defects such as missing contacts and transistor breakdown. Table 2.1 below shows the PCB fault classifications:

Defect Classes	Frequency of occurrence
shorts	51
opens	1
missing components	6
wrong components	14
reverse components	6
bent leads	8
timing	5

WHAT DOES THIS
REFER TO?

Table 2.1 - PCB Defects

(Hayes, 2002)

2.2 SINGLE STUCK AT FAULTS

The most common modelling technique used is the SSF fault model. This assumes that the fault within a logic gate results in either an input or output being fixed to logic high or low.

Three properties that define a SSF are:

- Only one faulty line
- Fault is permanently fixed to logic high or low
- Fault can be at an input or output of a gate

(Hayes, 2002)

Figure 2.2 below demonstrates a test condition at the input and a SSF at logic 1 at the output.

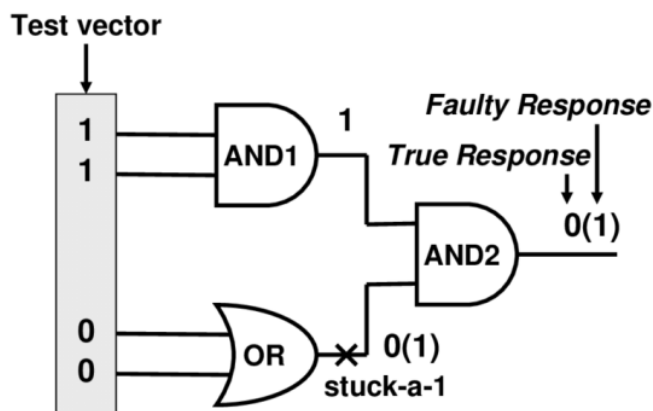


Figure 2.2 - SSF example

(Han, 2008)

SSF methods are used to reduce the complexity of test procedures due to an array of physical defects are able to be modelled using one logical stuck at fault. The SSF model is independent technology and can be used for TTL and CMOS.

In VLSI design verification and test, failure mode is used in reference to the manifestation of a defect at the electrical level (Plusquellic, 2004)

These defects can present themselves as short or open as mentioned earlier except in this case it VLSI which is on a large scale.

Faults at this level can be attributed to:

- Unwanted material or etching in the material
- Chemical deposits left from processing
- Dust particles due to contaminated clean room

When a fault shows up in tests for other faults it is known as fault dominance. This means only tests that pick up the other faults need to be carried out as the dominant fault will always be present regardless and not need a specific test.

When two faults are equally dominant where only one test needs to be carried out to detect the other, it is known as fault equivalence (Ubar, et al., 2017)

2.3 ELECTROSTATIC DISCHARGE DANGERS

Electrostatic discharge (ESD) is the release of static electricity when two objects come into contact (Rouse, 2011). The electro static charge is created by two insulator surfaces rubbing together or pulling apart. This will mean that once surface will gain electrons and the other will lose electrons which cause a potential between the surfaces.

ESD can be generated by mechanical means such as rubbing and sliding or fields of high energy nearby.

When ESD occurs in IC's the voltage can be high enough to cause the oxide dielectric breakdown, and high currents can cause melting of silicon, creating short circuits. This can permanently damage the device. Latency can be increased by ESD, where the IC is damaged but not to the point of failure. The effects of this get worse during the life of the device.



Figure 2.3 – ESD damage

(Technofaq, 2018)

A large amount of damage can be caused by ESD from human contact. Table 2.2 shows the levels of ESD components can withstand using the human body model.

Device type	ESD withstand voltage sensitivity (V) HBM
MR heads, RF FETs	10 - 100
Power MOSFETs PIN diodes, laser diodes	100 - 300V
Pre - 1990 VLSI	400 - 1000V
Modern VSLI	1000 - 3000V
HCMOS	1500 - 3000V
CMOS B Series	2000 - 5000 V
Linear MOS	800 - 4000 V
Small geometry older bipolar	600 - 6000 V
Small geometry modern bipolar	2000 - 8000 V
Power bipolar	7000 - 25000 V
Film resistor	1000 - 5000 V

Table 2.2 - Device ESD withstand levels from human body model

(Electrostatic Solutions Ltd, 2012)

With new development of IC's more silicon is used and the devices are becoming more complex VLSI. As a result, the amount of voltage needed to damage components is getting much smaller.

ESD will not necessarily instantly destroy circuits, it is far more frequent for the device to be damaged which will affect long term performance eventually leading to failure.

2.4 PREVENTION METHODS FOR ESD

An effective method used to protect IC's against ESD from manufacture to implementation are earth straps which can be worn on the ankle or wrist and are directly connected to an earth point, usually locally. The earth path provides a good path for the static voltage.

Other prevention methods used are antistatic instruments, carrier tape, antistatic bags, which eliminate any potential across a circuit and antistatic mats, which are usually used in industry clean rooms.

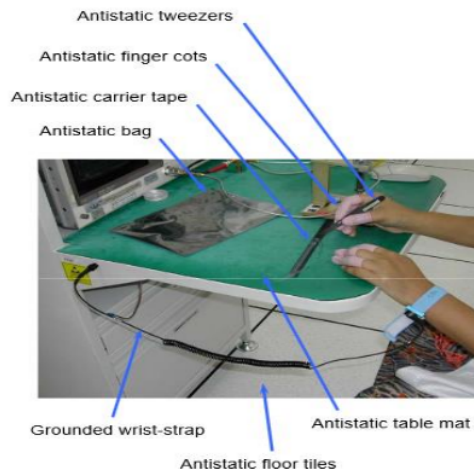


Figure 2.4 – ESD prevention

(Mini-circuits, 2015)

3 DESIGN OF ASSIGNMENT REQUIREMENTS

The objective of the assignment requires a combinational circuit for the CUT analysis. The circuit requirements were 6 input logic circuit which is shown in Figure 3.1.

The test sequence data was derived by using the SSF model which identifies and eliminates any redundant data in the sequence. The data sequence was greatly reduced using this method and produced the correct order of sequence for the test and analysis.

This circuit will be tested initially using simulation only methods using NI Multisim ECAD software. Final test analysis was carried out using a combination of a PCB design of the CUT using discrete component IC's with virtual instrumentation design methods using NI LabVIEW and DAQ mx interface.

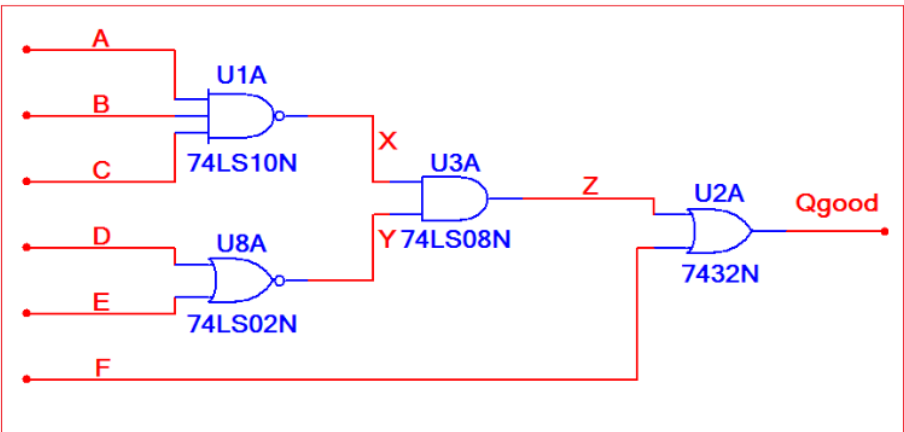


Figure 3.1 – CUT (Circuit Under Test)

3.1 STUCK AT FAULT MODEL METHOD

The 'stuck at' fault model technique was implemented to identify the SA0 and SA1 faults on the CUT.

The number of test vectors required for each individual gate in the CUT was reduced using the $m + 1$ formula where m being the number of gate inputs.

Figure 3.2 shows the 3 input NAND gate with table of test vectors with no redundancy.

	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

Figure 3.2 – NAND gate test vectors

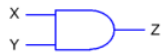
Figure 3.3 shows the 2 input NOR gate with table of test vectors with no redundancy.



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

Figure 3.3 – NOR gate test vectors

Figure 3.4 shows the 2 input AND gate with table of test vectors with no redundancy.



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

Figure 3.4 – AND gate test vectors

Figure 3.5 shows the 2 input OR gate with table of test vectors with no redundancy.



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

Figure 3.5 – OR gate test vectors

3.2 DERIVED TEST VECTORS

Table 3.1 shows the derived table of results for all possible stuck at fault tests and all conditions required to test each stage. The fault condition is set for each test so that the fault at the output of each gate follows the final output 'Qgood' in the 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	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	01	1	1	1	D+E SA1
6	0*	0*	0*	1	0	0	0	08	1	0	0	D SA0
7	0*	0*	0*	0	1	0	0	04	1	0	0	E SA0
8	0*	0*	0*	0	0	0	1	01	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	01	1	1	1	Z SA0

Table 3.1 – Initial derived table

Key:

* Rule that allows one condition must be set at logic 0

** Rule that allows one condition must be at logic 1

The tests in Table 3.1 identify the redundancies by the red and green rectangles. Test 1 and 9 in red are duplicates and tests 5, 8 and 13 are duplicates. Only one test is needed and subsequently the redundant tests can be eliminated.

The result of applying the duplicate redundancy rules produce a table of reduced test vectors which is shown in Figure 3.7.

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, 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	01	1	1	1	D+E SA1, X+Y SA0, Z SA0
6	0*	0*	0*	1	0	0	0	08	1	0	0	D SA0
7	0*	0*	0*	0	1	0	0	04	1	0	0	E SA0
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

Table 3.2 – Reduced table using redundancy rules

The number of tests in Table 3.2 can be further reduced using the wildcard equivalence rule on certain test vectors which are identified by '*' and '**' in the table.

Test 5 can be altered so that it becomes the same as test 4 which creates a redundant test vector. Only one of these is required and therefore one can be eliminated.

The same rule can be applied to test number 6 and 10 where the redundant test can be removed.

The result of this was a further reduced test table consisting of 8 test vectors instead of the initial 13.

This is shown in the table of Table 3.3 and it indicates that the fault condition of the redundant tests are added to the existing test vector.

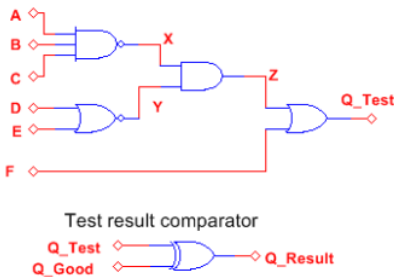
Test No	A	B	C	D	E	F	Qgood	HEX	X	Y	Z	Fault
1	1	1	1	0	0	0	0	70	0	1	0	A+B+C SA0, X SA1
2	0	1	1	0	0	0	1	31	1	1	1	A SA1
3	1	0	1	0	0	0	1	51	1	1	1	B SA1
4	1	1	0	0	0	0	1	61	1	1	1	C SA1, D+E SA1, X+Y SA0, Z SA0
6	0*	0*	0*	1	0	0	0	08	1	0	0	D SA0, Y SA1
7	0*	0*	0*	0	1	0	0	04	1	0	0	E SA0
11	1	1	1	1**	1**	0	0	7C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

Table 3.3 – Further reduced table

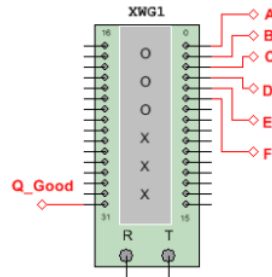
3.3 MULTISIM DESIGN TEST OF CUT USING WORD GENERATOR

The test demonstrated in Figure 3.6 below has no fault condition set and the results are recorded using the analyser.

Circuit Under Test



Word Generator



Logic analyser

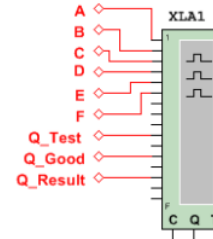


Figure 3.6 – Simulation test with no fault condition

The resultant waveform sequence of the test with no fault is shown in Figure 3.7 and the result waveform has no positive pulses which indicate that the output of the X-OR gate is at logic 0.

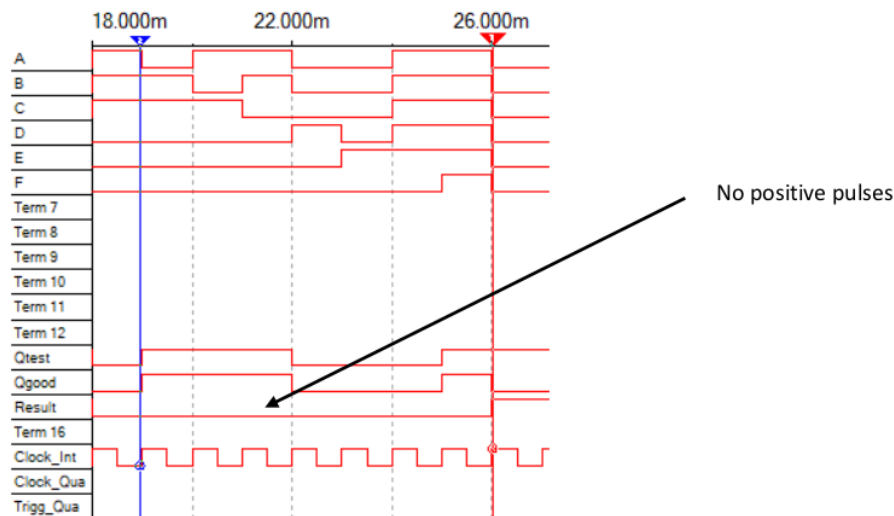


Figure 3.7 – No fault test

If there is a difference at the input of the X-OR gate then this would indicate a fault and the output will go to logic 1.

4 ANALYSIS OF NI MULTISIM RESULTS

The following tests were carried out with a set fault condition taken from the derived table of test vectors. The CUT was set with three separate fault conditions to prove the theory and design principle.

Figure 4.1 shows the CUT modified for a stuck-at fault of input A stuck at logic 1: A SA1

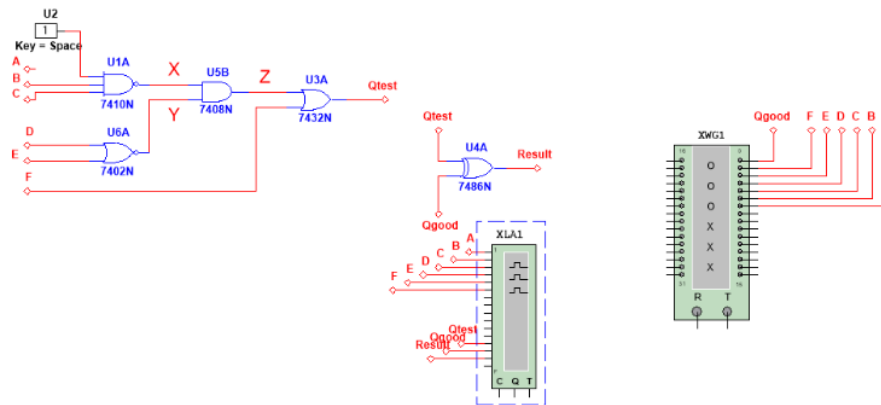


Figure 4.1 – Input A SA1 fault condition

Figure 4.2 shows the resultant output for the fault condition where the output 'Result' of the X-OR gate has a positive pulse when there is a discrepancy at its inputs. The input from the CUT is showing a fault and is sensed at the output of the X-OR gate.

The two cursors indicate the start and finish points of the test sequence.

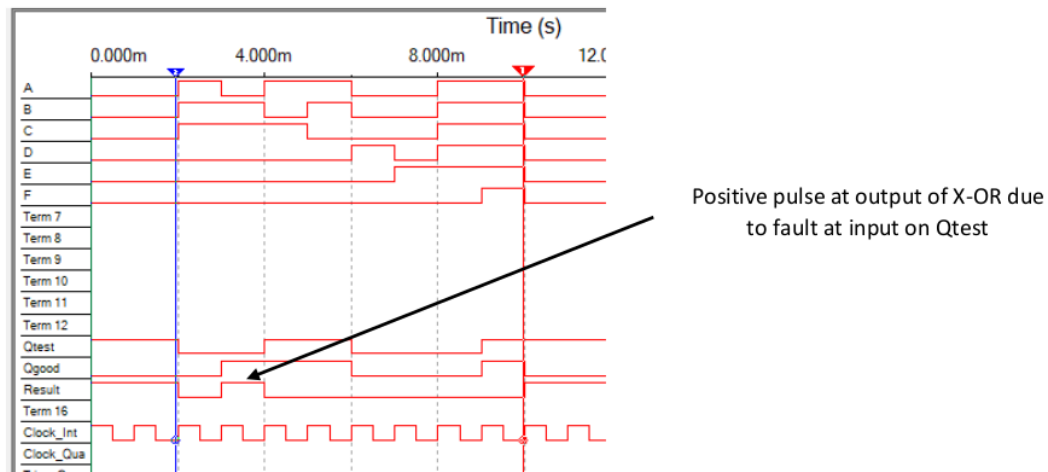


Figure 4.2 – Test result for input A SA1

Figure 4.3 shows the CUT modified for a stuck at fault of input C stuck at logic 1: C SA1

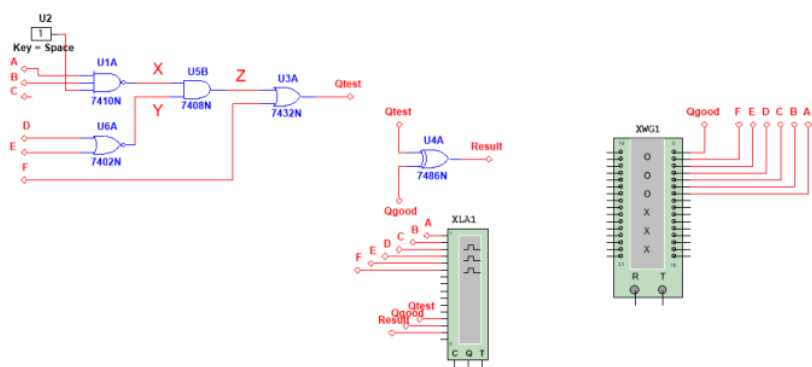


Figure 4.3 – Input C SA1 fault condition

Figure 4.4 identifies the fault condition with a positive pulse at the X-OR gate output ‘Result’ due to the error implemented in the CUT ‘Qtest’ of input C SA1.

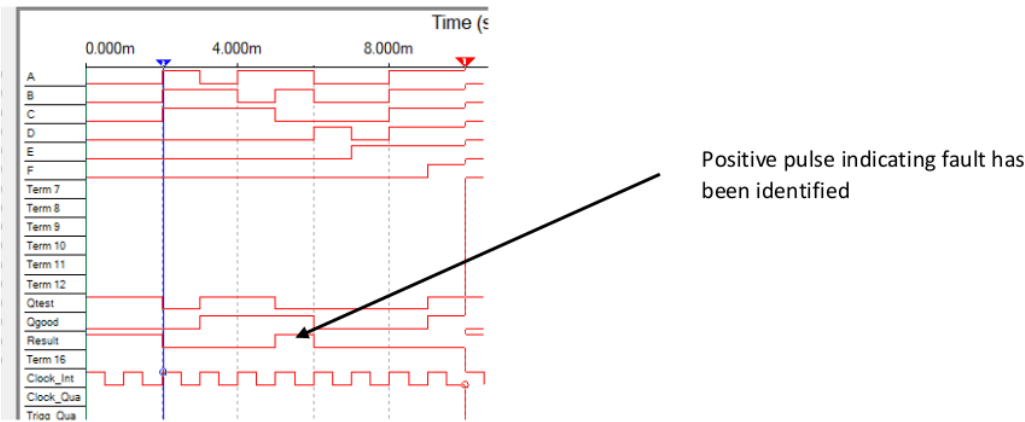


Figure 4.4 – Input C SA1 fault condition

Key = Space

Inputs: A, B, C, D, E, F

ICs: U1A (7410N), U5B (7408N), U3A (7432N), U4A (7486N)

Outputs: Qtest, Qgood, Result

Truth Table:

A	B	C	D	E	F	Qtest	Qgood	Result
0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0
0	0	0	0	1	0	0	0	0
0	0	0	0	1	1	0	0	0
0	0	0	1	0	0	0	0	0
0	0	0	1	0	1	0	0	0
0	0	0	1	1	0	0	0	0
0	0	0	1	1	1	0	0	0
0	0	1	0	0	0	0	0	0
0	0	1	0	0	1	0	0	0
0	0	1	0	1	0	0	0	0
0	0	1	0	1	1	0	0	0
0	0	1	1	0	0	0	0	0
0	0	1	1	0	1	0	0	0
0	0	1	1	1	0	0	0	0
0	0	1	1	1	1	0	0	0
0	0	1	1	1	1	0	0	0
0	1	0	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0
0	1	0	0	1	0	0	0	0
0	1	0	0	1	1	0	0	0
0	1	0	1	0	0	0	0	0
0	1	0	1	0	1	0	0	0
0	1	0	1	1	0	0	0	0
0	1	0	1	1	1	0	0	0
0	1	0	1	1	1	0	0	0
0	1	1	0	0	0	0	0	0
0	1	1	0	0	1	0	0	0
0	1	1	0	1	0	0	0	0
0	1	1	0	1	1	0	0	0
0	1	1	1	0	0	0	0	0
0	1	1	1	0	1	0	0	0
0	1	1	1	1	0	0	0	0
0	1	1	1	1	1	0	0	0
0	1	1	1	1	1	0	0	0
0	1	1	1	1	1	0	0	0
1	0	0	0	0	0	0	0	0
1	0	0	0	0	1	0	0	0
1	0	0	0	1	0	0	0	0
1	0	0	0	1	1	0	0	0
1	0	0	1	0	0	0	0	0
1	0	0	1	0	1	0	0	0
1	0	0	1	1	0	0	0	0
1	0	0	1	1	1	0	0	0
1	0	0	1	1	1	0	0	0
1	0	0	1	1	1	0	0	0
1	0	1	0	0	0	0	0	0
1	0	1	0	0	1	0	0	0
1	0	1	0	1	0	0	0	0
1	0	1	0	1	1	0	0	0
1	0	1	1	0	0	0	0	0
1	0	1	1	0	1	0	0	0
1	0	1	1	1	0	0	0	0
1	0	1	1	1	1	0	0	0
1	0	1	1	1	1	0	0	0
1	0	1	1	1	1	0	0	0
1	1	0	0	0	0	0	0	0
1	1	0	0	0	1	0	0	0
1	1	0	0	1	0	0	0	0
1	1	0	0	1	1	0	0	0
1	1	0	1	0	0	0	0	0
1	1	0	1	0	1	0	0	0
1	1	0	1	1	0	0	0	0
1	1	0	1	1	1	0	0	

Figure 4.6 identifies the fault condition with a positive pulse at the X-OR gate output 'Result' due to the error implemented in the CUT 'Qtest' of input D SA1.



5 THE 6 – BIT REGISTER DESIGN

The 6-bit register ATE test method was designed and implemented for the purpose of testing the CUT. The same fault conditions were used for fault detection and an analysis of the results was carried out.

Table 5.1 demonstrates a next state conditions table for the finite state machine that produces the test sequence for the inputs A through F.

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

Table 5.1 – Next state conditions table

The next stage was to use Karnaugh Maps to generate the logic state conditions required for the feedback logic for the input sequence.

Table 5.2 demonstrates the Karnaugh Map for output QA of the 6 bit register

ABC/DEF	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

Table 5.2 – Output QA

Boolean expression derived from the Karnaugh Map:

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

Table 5.3 demonstrates the Karnaugh Map for output QB of the 6-bit register

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

Table 5.3 – Output QB

Boolean expression derived from the Karnaugh Map:

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

Table 5.4 demonstrates the Karnaugh Map for output QC of the 6-bit register

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

Table 5.4 – Output QC

Boolean expression derived from the Karnaugh Map:

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

Table 5.5 demonstrates the Karnaugh Map for output QD of the 6-bit register

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

Table 5.5 - Output QD

Boolean expression derived from the Karnaugh Map:

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

Table 5.6 demonstrates the Karnaugh Map for output QE of the 6-bit register

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

Table 5.6 - Output QE

Boolean expression derived from the Karnaugh Map:

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

Not a group due to 6 variable K-MAP rules

Table 5.7 demonstrates the Karnaugh Map for output QF of the 6-bit register

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

Table 5.7 – Output F

Boolean expression derived from the Karnaugh Map:

A.B.C.D.E.NF

The design of the 6-bit register with combination feedback circuit was taken from the results of the Karnaugh Map procedure.

The design circuit is shown in Figure 5.1 below and demonstrates the no fault condition on the CUT.

The circuit diagram identifies the CUT and the good template circuit and the outputs of both at the inputs of the X-OR gate which acts as a comparator.

As demonstrated in the analysis there are no positive pulses at the output of the X-OR gate which indicates no fault condition.

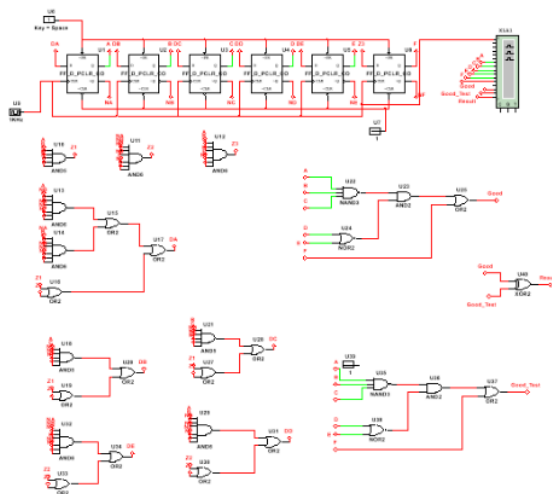


Figure 5.1 – No fault condition

Figure 5.2 shows the test result analysis for no fault condition

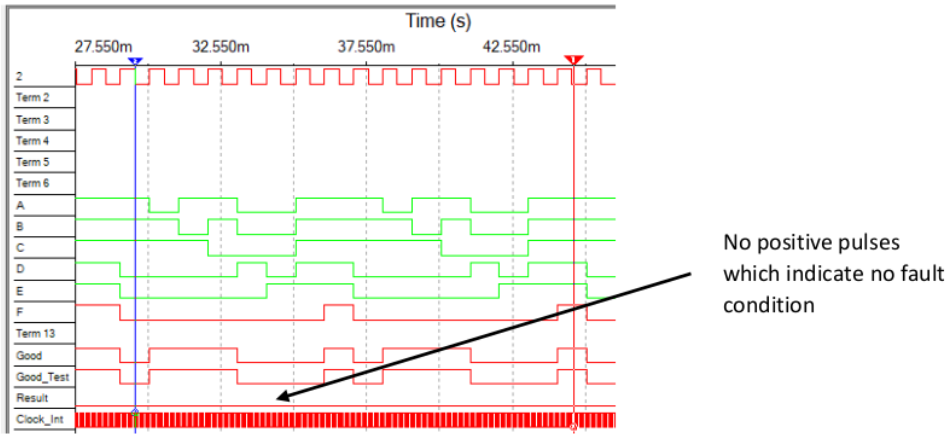


Figure 5.2 – No fault condition

6 TEST ANALYSIS FOR THE 6 BIT REGISTER DESIGN

Identical fault conditions from previous tests using Multisim Word Generator were set for the 6-bit register.

The faults condition set:

Input A SA1

Input C SA1

Input D SA1

Figure 6.1 shows the CUT with stuck at fault input A SA1

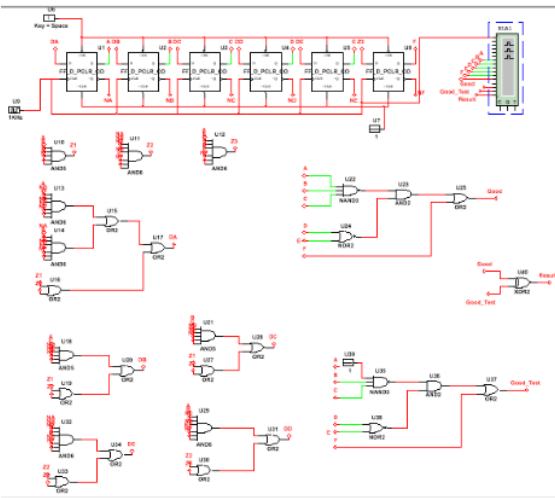


Figure 6.1 – Input A SA1 fault condition set

Figure 6.2 shows the test result analysis for the stuck at fault input A SA1

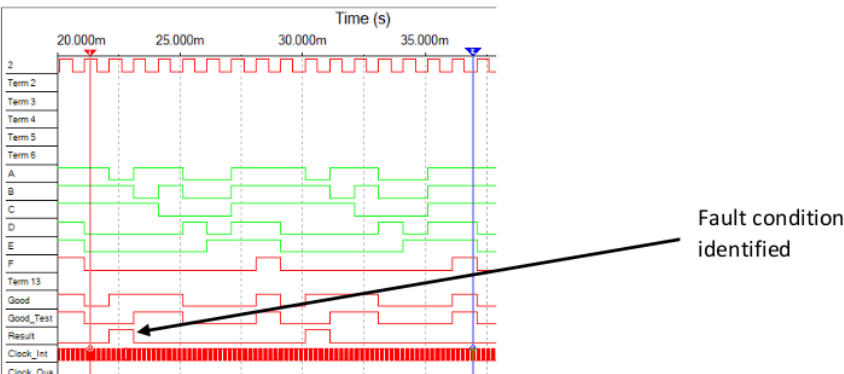


Figure 6.2 – Input A SA1 fault result

Figure 6.3 shows the CUT with stuck at fault input C SA1

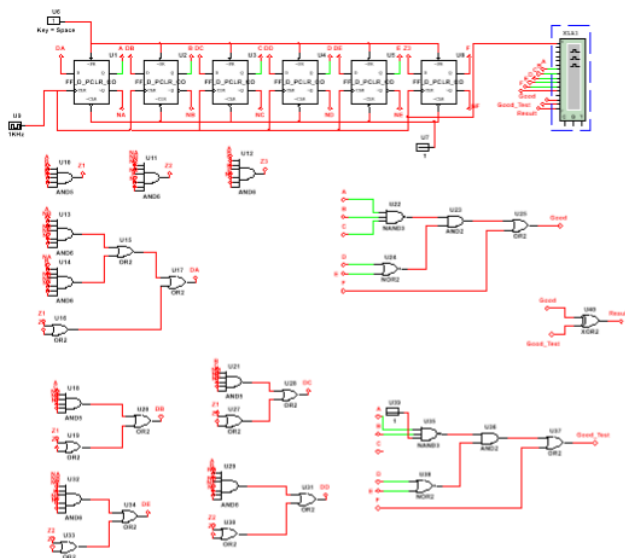


Figure 6.3 – Input C SA1 fault condition set

Figure 6.4 shows the test result analysis for the stuck at fault input C SA1

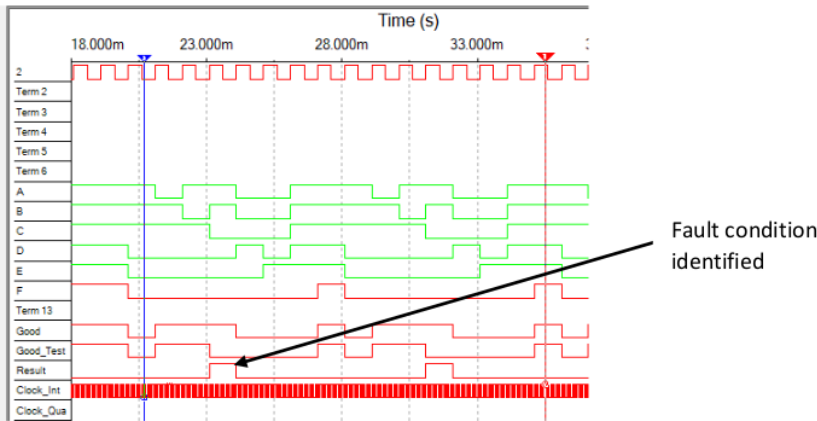


Figure 6.4 – Input C SA1 fault result

Figure 6.5 shows the CUT with stuck at fault input D SA1

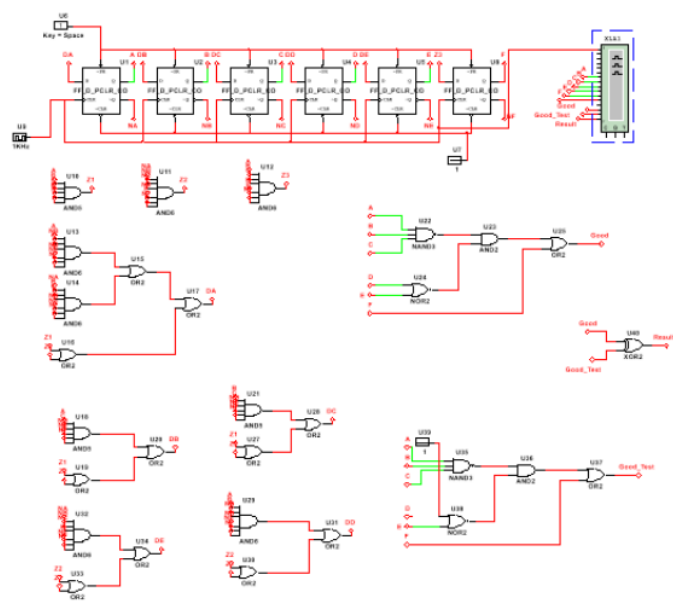


Figure 6.5 – Input D SA1 fault condition set

Figure 6.6 shows the test result analysis for the stuck at fault input D SA1

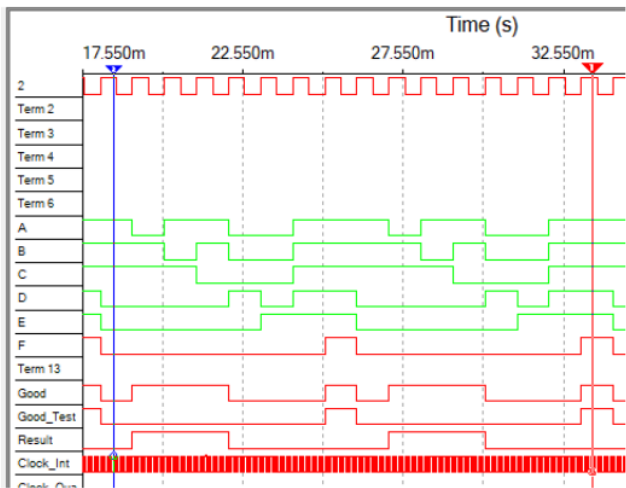


Figure 6.6 – Input D SA1 fault result

7 PRACTICAL BUILD AND TEST USING LABVIEW

The software NI LabVIEW was used as an alternative method for ATE to test the identical CUT. The interface was provided by NI Elvis II DAQ development board and the associated interface library that is added to LabVIEW as a plugin option.

The Figure below shows the G-code program which was designed to run the data stream to drive the inputs of the CUT.

The output of the CUT is fed into an X-OR gate input along with a generated data pattern of a good set of results. This enable the output of the CUT to be compared with the good data stream generated from LabVIEW.

The X-OR gate provides the comparison result by feeding its output into LabVIEW via the DAQ input.

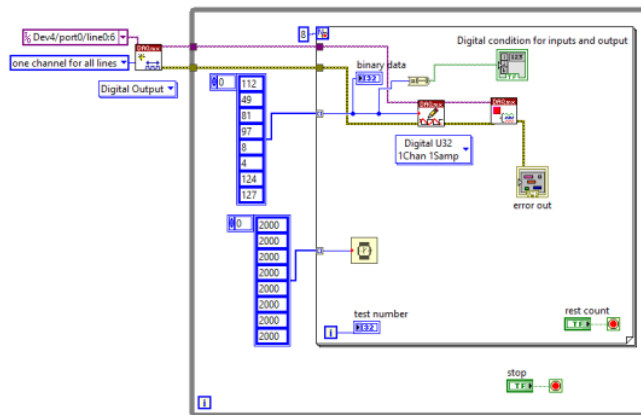


Figure 7.1 - Data pattern generator to drive inputs of CUT via DAQ outputs

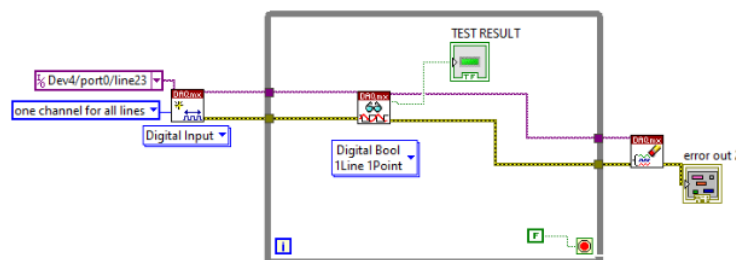


Figure 7.2 - Result from X-OR gate sent to LabVIEW via DAQ input

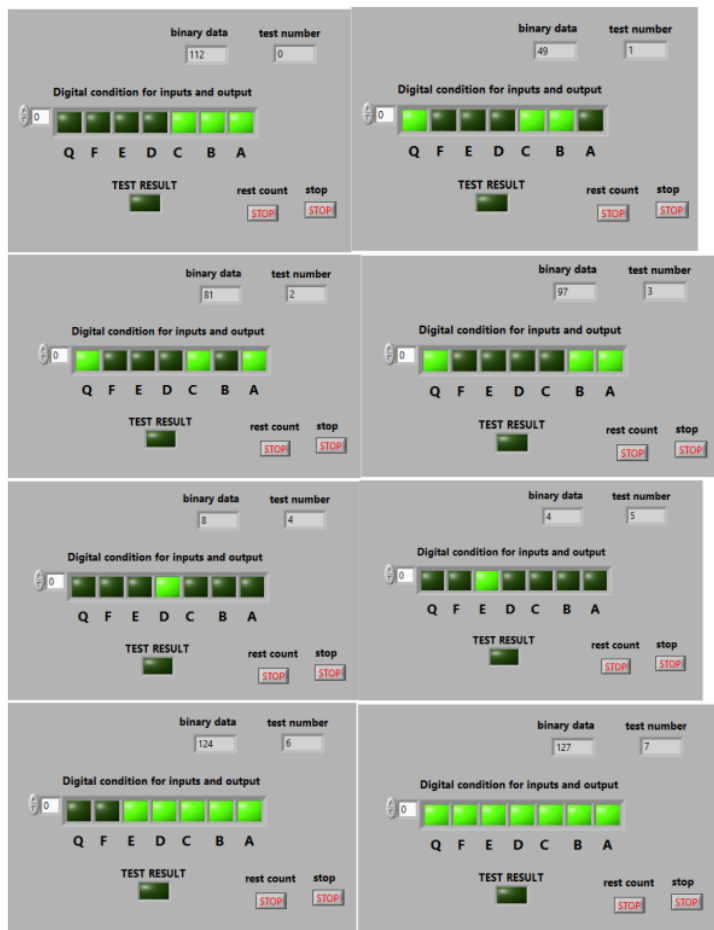


Figure 7.3 – Full set of test vectors with no fault condition

Figure 7.3 above shows the result of running the test sequence on the practical CUT setup on the Elvis DAQ development board. This setup is shown in Figure 7.4 below.

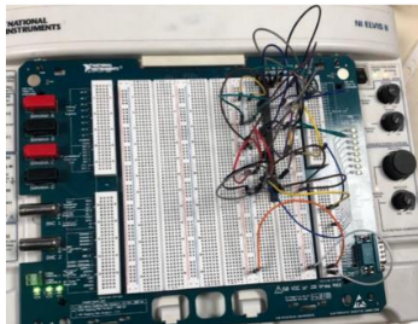


Figure 7.4 – DAQ CU

Fault conditions set on the practical CUT identical to the ECAD test:

- A SA1
- C SA1
- D SA1

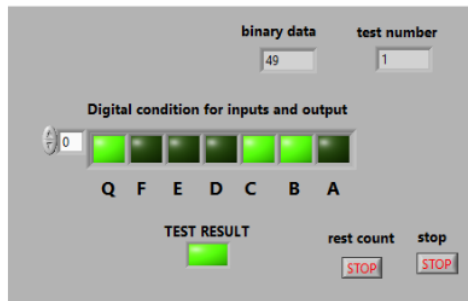


Figure 7.5 – Fault indicated for A SA1 condition

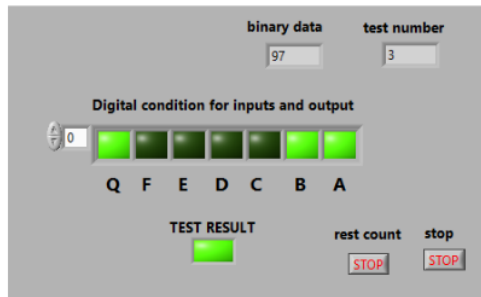


Figure 7.6 – Fault indicated for C SA1 condition

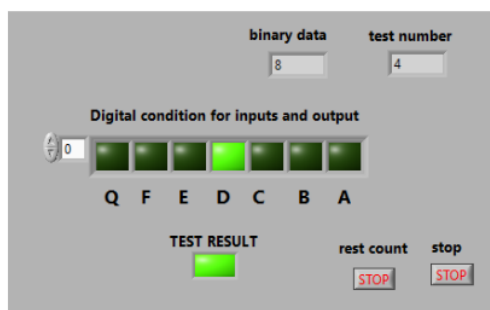


Figure 7.7 – Fault indicated for D SA1 condition

Figure 7.8 shows the CUT PCB connected to the elvis DAQ.

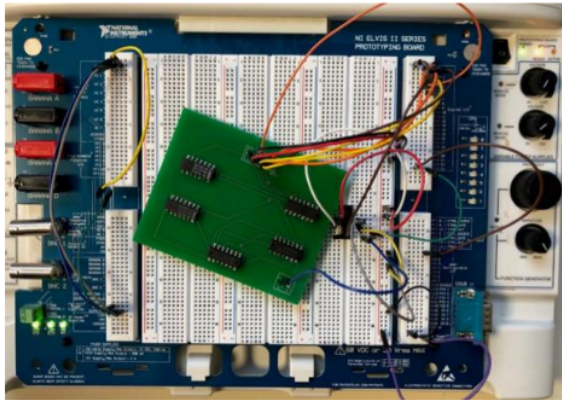


Figure 7.8 - PCB CUT Test using LabVIEW and Elvis DAQ

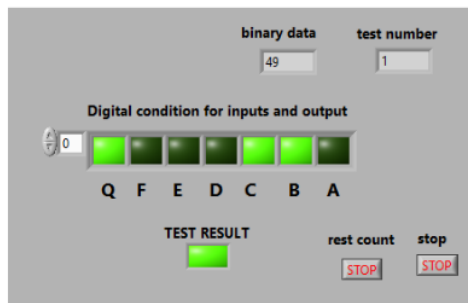


Figure 7.9 – Fault indicated for A SA1 condition

The PCB test indicated no faults with no fault condition set which was identical to the practical CUT using DAQ. With the same fault condition set; proves that the PCB behaves in the same way as the identical practical CUT with the DAQ development board.

8 CAM DESIGN AND BUILD OF PROTOTYPE

Using industry approved CAM software – Eagle, the test circuit was built on the software so that the Gerber files could be sent to a supplier to manufacture the circuit PCB. Figure 8.1 show the schematic design created and Figure 8.2 shows the board design, including tracks and drill holes. Figure 8.3 contains is the content of the zipped circuit designed in Eagle. Figure 8.4 - Figure 8.7 shows the content of the guber files.

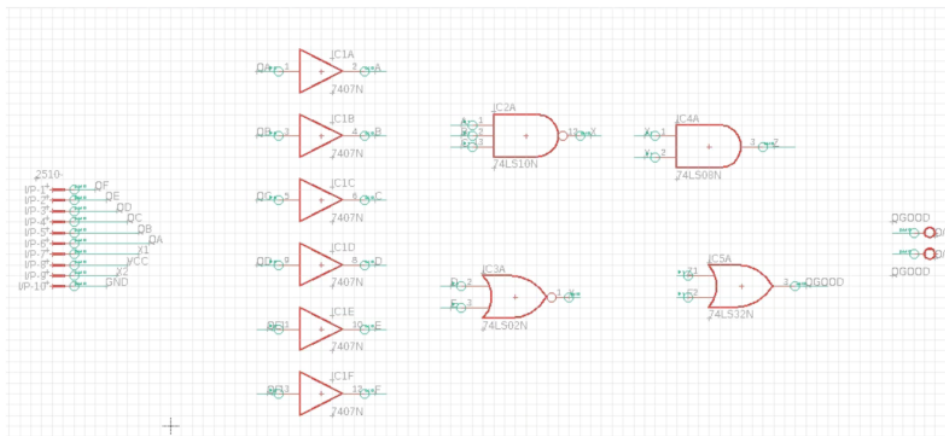


Figure 8.1 - Schematic Design in Eagle

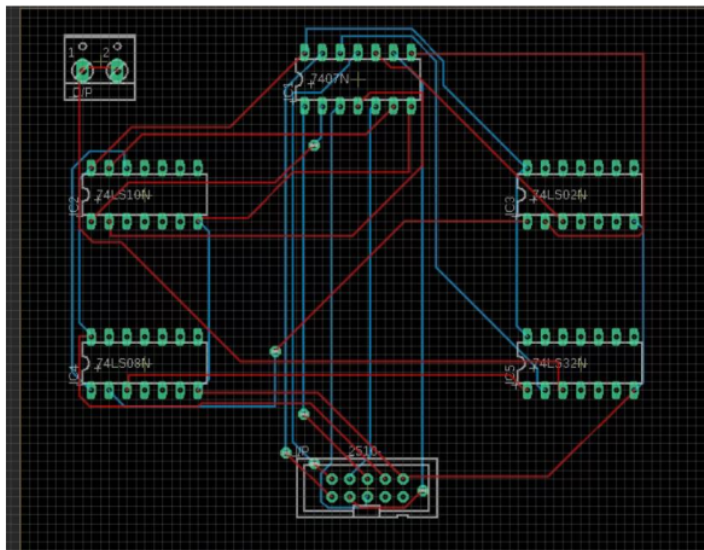


Figure 8.2 - Board design in Eagle

comp.cmp	03/02/2019 21:27	CMP File	7 KB
comp.drd	03/02/2019 21:27	DRD File	2 KB
comp.dri	03/02/2019 21:27	DRI File	1 KB
comp.gpi	03/02/2019 21:27	GPI File	2 KB
comp	03/02/2019 21:27	S7wsvapx.ViewDo...	38 KB
comp.pls	03/02/2019 21:27	PLS File	1 KB
comp.sol	03/02/2019 21:27	SOL File	6 KB
comp.stc	03/02/2019 21:27	STC File	4 KB
CUT2_2019	24/01/2020 12:25	File	7 KB
CUT2_2019.gpi	24/01/2020 12:25	GPI File	2 KB
CUT2_2019	15/02/2019 15:02	WinRAR ZIP archive	28 KB
ept1.b#1	23/01/2019 12:59	B#1 File	36 KB
ept1	24/01/2020 11:59	EAGLE board	36 KB
ept1.pro	24/01/2020 12:00	PRO File	1 KB
ept1	14/01/2019 01:07	EAGLE schematic	59 KB

Figure 8.3 - Gerber Files

The files were then tested on Gerber viewer.

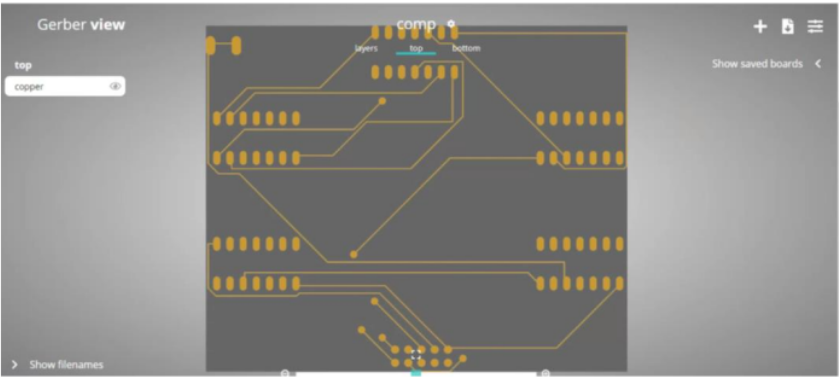


Figure 8.4 - Gerber View - Copper - Top

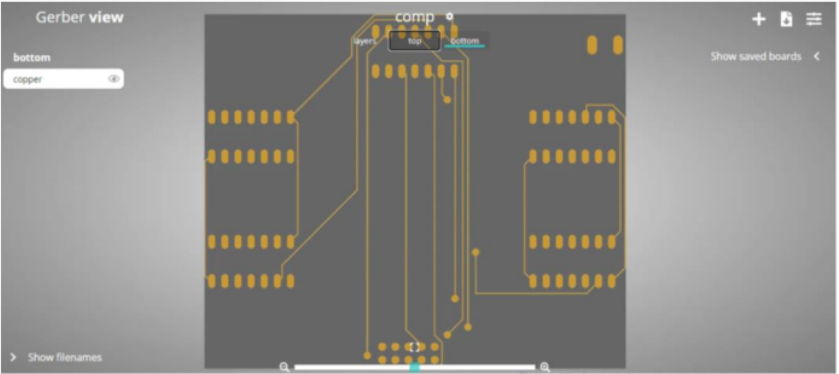


Figure 8.5 - Gerber View - Copper – Bottom



Figure 8.6 - Gerber View - Drill Holes (underlined for visibility)

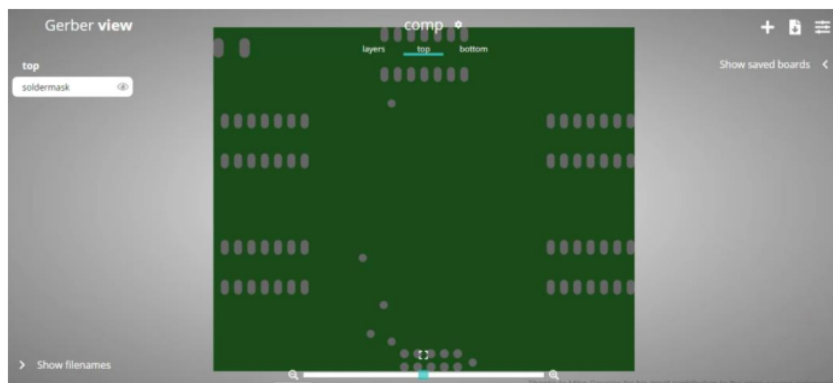


Figure 8.7 - Gerber View – Solder Positions

9 CONCLUSIONS AND RECOMMENDATIONS.

It can be concluded that the CUT met the design requirement of 6 input combinational circuit. The Stuck at fault model was used to reduce the number of test vectors required to carry out a full test of the CUT from 13 to 8. The process of test vector reduction was carried out using tables to quantify the redundant tests by identifying similarities and using wildcard methods to change values within the model rules.

The resultant test vectors were implanted using Multisim word generator to generate the inputs needed for the CUT. Tests were carried out for the no fault condition initially to prove the CUT integrity. Following this, 3 faults from the resultant table of test vectors were chosen and set in the CUT. The CUT was then tested with these fault condition set in turn and the resultant output was identified by the using the X-OR gate for comparison. From comparing the CUT output to that of a good knowing working output, it made it possible to identify the fault conditions.

The next stage was testing the CUT under the exactly the same conditions using a 6-bit pattern generator (shift registers) ATE design implementation. The results were taken and compared with the word generator.

The final testing was using LabVIEW and DAQ interface board to drive the 6 inputs of the CUT and carry the same test procedure. The advantages of LabVIEW is that it provides a Graphical User Interface (GUI) of the real time test process. The test with no fault condition was carried out initially to prove the CUT using LabVIEW and then the identical fault conditions were set and tested.

The result of the procedures carried out was that all 3 gave the same fault indications for the same fault conditions set on the CUT, meaning it can be concluded that the circuit functioned as the theory predicted it would.

In future, the Multisim PLD and the LabVIEW could be implemented as ATE for production circuits and the fault reduction model can be implemented to reduce the test time

10 BIBLIOGRAPHY

Electrostatic Solutions Ltd, 2012. *ESD Sensitivity of electronic devices*. [Online]

Available at: https://www.electrostatics.net/ESD_Guide/technical/ESD_sensitivity.htm

[Accessed 16th January 2021].

Han, K., 2008. *A Parallel Implementation of Fault Simulation on a Cluster of Workstations*. Beijing, IEE International.

Hayes, J. P., 2002. *Digital System Testing*. [Online]

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

[Accessed 16th January 2021].

Mini-circuits, 2015. *Scientific components Corporation*. [Online]

Available at: www.minicircuits.com/homepage/terms_of_use.html

[Accessed 16th January 2021].

Plusquellic, J., 2004. *VLSI Design Verification and Test*. [Online]

Available at: http://ece-research.unm.edu/jimp/vlsi_test/slides/defects.pdf

[Accessed 27th January 2021].

Rouse, M., 2011. *Techtarget*. [Online]

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

[Accessed 16th January 2021].

Technofaq, 2018. *Understanding the Nature of How ESD Damages Your Components*. [Online]

Available at: <https://technofaq.org/posts/2018/05/understanding-the-nature-of-how-esd-damages-your-components/>

[Accessed 16th January 2021].

Ubar, R., Jurimagi, L., Orasson, E. & Raik, J., 2017. *Fault Collapsing in Digital Circuits Using Fast Fault*. [Online]

Available at: <http://hal.inria.fr/hal-01578613/document>

[Accessed 27th January 2021].

SGEE6002 Assignment 1

GRADEMARK REPORT

FINAL GRADE

GENERAL COMMENTS

Instructor

76/100

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13

PAGE 14

PAGE 15

PAGE 16

PAGE 17

PAGE 18

PAGE 19

PAGE 20

PAGE 21

PAGE 22

PAGE 23

PAGE 24

PAGE 25

PAGE 26

PAGE 27

PAGE 28

PAGE 29

PAGE 30

PAGE 31

PAGE 32

PAGE 33

PAGE 34

PAGE 35

GRADING FORM: CSG-RUBIC

JONATHON WILLIAMS

0

RESEARCH-INFORMED LITERATURE



Research covers significant detail of the requirements and has numerous relevant sources.

12/15

Research carried out shows extensive reading around the subject area and a good variety of examples are included from the research. A good number of relevant sources have been researched.

KNOWLEDGE AND UNDERSTANDING



A sound knowledge has been demonstrated acquiring the required minimum set of test vectors . 8.5/10 The design of the LabVIEW test vectors shows a great deal of study has ben carried out in creating the test program, and implementing the test sequence. 8.5/10 The PLD design alternative underpins the knowledge gained from your study and the alternative designs demonstrate a good knowledge of the subject area with regard to data handling. 4/5 The CAM design of the CUT has been carried out successfully using an approved industrial software and the Gerber design files have been created. 3/5

24/30

ANALYSIS



Good extensive ECAD simulation testing has been carried out using very good designs. Good explanations and supporting images are included in your results. Relevant reference points are made to the test vector table from previous task. 7.5/10 A good comparison to theory has been carried out with the test results and explained in the analysis and in the conclusion. Recommendations have been highlighted. 3.5/5

11/15

PRACTICAL APPLICATION AND DEP.



Prototype design of CUT was carried out successfully and Gerber files created for PCB manufacture. The resulting PCB design was error free and fit for ATE testing. 7.5/10 The LabVIEW ATE testing carried out has shown to be successful with a good range of test scenarios selected from test vector table. 11.5/15

19/25

SKILLS FOR PROFESSIONAL PRAC.



A good use of a range of software packages demonstrated with programming skills. 3.5/5 Report layout is very good and has the correct sequence of tasks required. 3/5 The conclusion covers all the good points and has included some critical analysis. 3.5/5

10/15