

assignment 1

by Joshua Davies

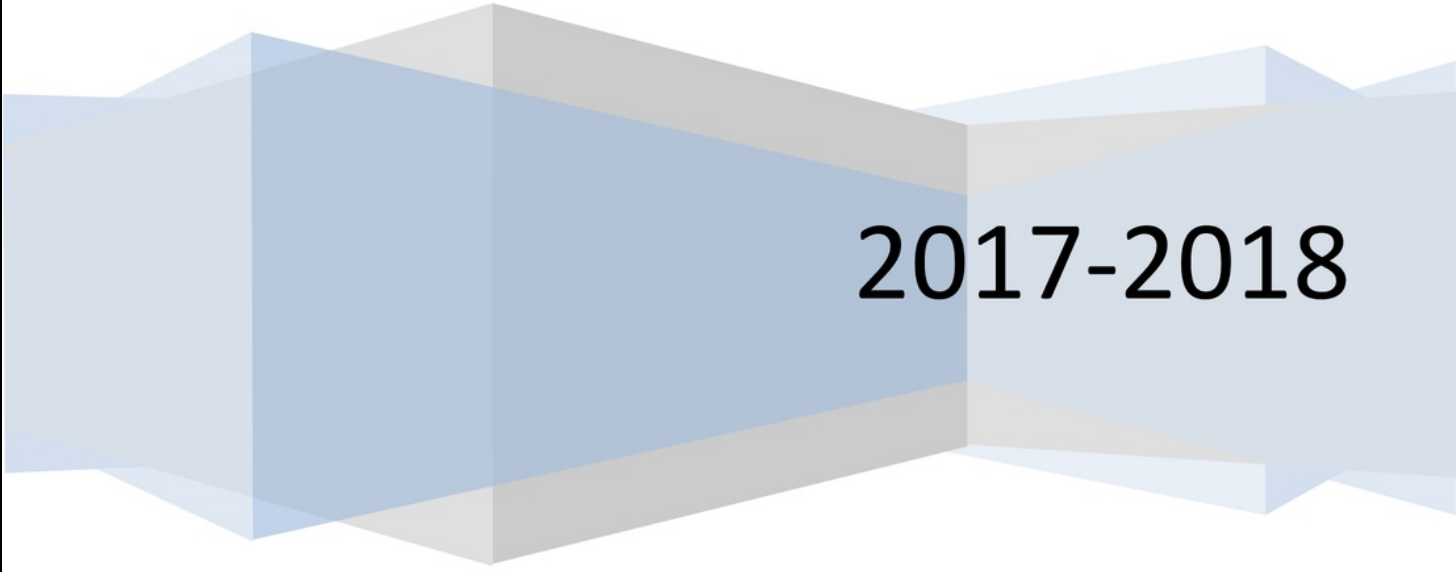
FILE	8847_JOSHUA_DAVIES_ASSIGNMENT_1_155126_956093049.DOCX (6.49M)		
TIME SUBMITTED	15-FEB-2018 12:29PM (UTC+0000)	WORD COUNT	6133
SUBMISSION ID	83212847	CHARACTER COUNT	42371



colegsirgâr

Pseudo Random Pattern Generators **PSDE6018 – Electronic Production & Test**

Joshua Davies – BSc. (Hons) Electrical And Electronic Engineering Year 3



2017-2018

Contents

Table of Figures	4
1. Introduction.....	6
2. Research.....	7
2.1. Surface Mounted Technology	7
2.2. Plated Through Hole	8
2.3. PCB Process.....	9
2.5. Automatic Optical Inspection	11
2.6. Automatic X-Ray Inspection	13
3. NI MultiSIM CAR Design.....	15
3.1. 4 Bit CAR 90 Rule Development	15
3.2. 4 Bit CAR 150 Rules Development.....	17
3.3. 4 Bit Hybrid CAR 90/150 Development.....	19
3.4. 6Bit Hybrid CAR 90/ 150 Development.....	21
3.5. Testing the NI MultiSIM 6Bit Hybrid 90/150 CAR	24
3.5.1. Test Circuit Fault 1	24
3.5.2. Test Circuit Fault 2	25
3.5.3. Test Circuit Fault 3	27
4. HDL FPGA Design.....	28
4.1. 4 Bit CAR 90 Development	28
4.2. 4 Bit CAR 150 Development	28
4.3. 4 Bit Hybrid CAR 90/150 Development.....	29
4.4. 6 Bit Hybrid CAR 90/150	31
4.5. Testing HDL FPGA 6Bit Hybrid CAR 90/150 on FPGA Board.....	34
4.6. Testing HDL FPGA 6Bit Hybrid CAR 90/150 Against Test Circuits	35
4.6.1. Test Circuit Fault 1	35
4.6.2. Test Circuit Fault 2	35
4.6.3. Test Circuit Fault 3	36
5. Practical PCB Build and Test.....	37
5.1. Testing PCB.....	40
6. Analysis	41

6.1 Analysis of NI MultiSim Design	41
6.2. Analysis of HDL FPGA Design.....	42
6.3. Analysis of PCB Design	42
7. Conclusions and Recommendations	43
8. References.....	44
10. Appendix.....	46

Table of Figures

Figure 1 – Surface Mounted Technology on PCB.....	7
Figure 2 - Characteristics of Surface Mount Technology.....	7
Figure 3 - Multiple Layers of PCB Board	8
Figure 4 - PTH PCB Board	8
Figure 5 - Wave Soldering Process.....	9
Figure 6 - Reflow Process.....	10
Figure 7 - Automatic Optical Inspection process	11
Figure 8 - Automatic Optical Inspection	12
Figure 9 - AXI Technology able to see Heel joint of PCB.....	13
Figure 10 - Comparison of detection systems.....	14
Figure 11 - Wolframs CAR 90 Rule	15
Figure 12 - MultiSIM CAR90 Circuit and Sequence	16
Figure 13 - MultiSIM CAR90 Logic Analyser Result	16
Figure 14 - Wolframs CAR 150 Rule	17
Figure 15 - MultiSIM CAR150 Circuit and Sequence.....	18
Figure 16 - MultiSIM CAR150 Logic Analyser Results	18
Figure 17 - MultiSIM Hybrid CAR90/150 Circuit and Sequence.....	19
Figure 18 – MultiSIM 4Bit Hybrid CAR90/150 Logic Analyser Results	20
Figure 19 - MultiSIM Circuit of 6Bit Hybrid CAR90/150.....	21
Figure 20 – 6 bit hybrid CAR90/150 Circuit with X_Ors.....	21
Figure 21 - MultiSIM 6Bit Hybrid CAR90/150 Logic Analyser Results.....	22
Figure 22 - Logic analyser Cursor bit count display	22
Figure 23 - MultiSIM Logic Analyzer Bit Count table	23
Figure 24 - Test Circuit 1 with Working Circuit	24
Figure 25 - Test 1 Logic Analyzer Comparison	25
Figure 26 - Test Circuit 2 with working circuit	26
Figure 27 – Test Circuit 2 Logic Analyzer Comparison	26
Figure 28 – Test Circuit 3 Logic Analyzer Comparison	27
Figure 29 - Test Circuit 3 with working Circuit.....	27
Figure 30 - 4Bit CAR 90 HDL Simulation Results	28
Figure 31 - 4Bit CAR 150 HDL Simulation Results	29
Figure 32 - 4Bit CAR 90/150 Hybrid HDL Simulation Results Part 1.....	29
Figure 33 - 4Bit CAR 90/150 Hybrid HDL Simulation Results Part 2.....	30
Figure 34 - Vivadoo Circuit.....	31
Figure 35 - 6Bit Hybrid CAR 90/150 HDL Code	32
Figure 36 – 6 Bit Hybrid CAR 90/150 Simulation Results.....	33
Figure 37 - 6Bit Hybrid CAR 90/150 tested on FPGA Board Pass.....	34
Figure 38 - 6Bit Hybrid CAR 90/150 Tested on FPGA Board Fail.....	34
Figure 39 - FPGA Test circuit Fault 1	35
Figure 40 - FPGA Test Circuit Fault 2.....	35

Figure 41 - FPGA Test Circuit Fault 3.....	36
Figure 42 - Logic Circuit produced on Eagle software	37
Figure 43 – PCB Layout and connection pads	37
Figure 44 - Bottom Layer tracks of PCB	38
Figure 45 - Top Layer tracks of PCB.....	38
Figure 46 - Completed PCB With All elements included.....	39
Figure 47 - Manufactured PCB of Test circuit	39
Figure 48 - Final Test stage of 6Bit Hybrid CAR with PCB Test circuit.....	40
Figure 49 - Final Test stage of 6Bit Hybrid CAR PCB Test with fault	40

1. Introduction

The aims of this assignment is to investigate into PCB design and soldering techniques for simple 2 layer PCBs' and the use of Automatic Optical Inspection and Automatic X-Ray Inspection within the PCB Production process for Automatic Test Equipment. To design and test a collection of random pattern generator models through simulation software MultiSIM and Vivadoo using hardware descriptive language. To produce a PCB circuit of a random pattern generator and test the circuit to ensure it meets the customers specification.

A little weak on objectives. The virtual ATE integration is possible here.

2. Research

2.1. Surface Mounted Technology

Surface mounted components are directly placed onto the PCB surface to produce an overall surface mount device. Surface mounted technology has extensively replaced through hole technology methods for building PCBs' because of its efficient and effective approach though both methods can be utilized on the same PCB.

Both methods can be utilized on the same PCB as not all components are suited for surface mounting, this includes transformers and heat sinks for semi-conductors. Surface mounted components are generally smaller in size than through hole components as they contain none or smaller leads allowing more electronics to be packed within a smaller space than through hole components. (Zentech-Manufacturing, 2014)

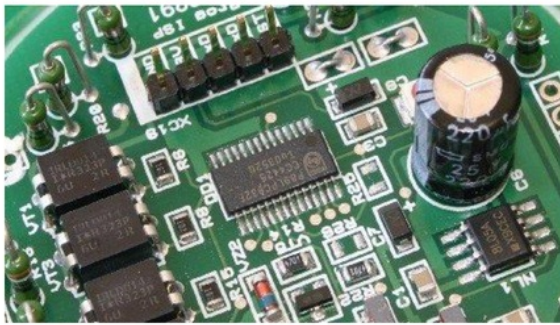


Figure 1 – Surface Mounted Technology on PCB

(airborn.com, 2015)

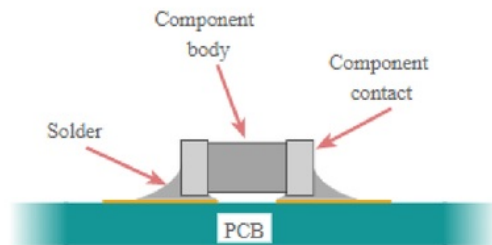


Figure 2 - Characteristics of Surface Mount Technology

(Electronics-Notes, 2016)

Surface mount technology (SMT) is virtually the exclusive way for manufacturing electronic circuits today as they are smaller, produce greater performance and can be applied through the use of automatic pick and place robots which are much faster assembling boards. This almost eliminate the need for human intervention within the assembly process unlike using through hole components as it is difficult to apply them automatically as the legs needed to be formed to fit the hole spacing.

The majority of connectors and components still need assisted placing, PCBs are usually developed to decrease the assisted placing components to a minimum, this could include altering the design to incorporate components than be placed automatically. Some components

manufacturers have developed their own surface mount components that will allow completely automatic assembly of PCBs (Electronics-Notes, 2016)

2.2. Plated Through Hole

Through hole technology is the mounting characteristic of electronic components onto a PCB. This includes the use of components that have leads, these leads are then inserted into pre-drilled holes in the PCB and then fixed by solder on the reverse side through manual assembly or with the aid of automatic assembly lines.

Through hole technology surpassed early electronics assembly techniques like the wire wrapping method. Before the introduction of SMT, through hole technology was the preferred method from the 1950s' second generation of computers until the late 1980s' where all components on a PCB were through-hole.

Printed circuit boards started off with tracks printed on a single side and as technology improved, both sides and then multiple-layer pcb boards were available where the through-holes later became Plated Through Holes (PTH). Components turned to PTH so that the components could make contact with the specific conductive layer of the PCB. However, with the introduction of SMT, PTH boards and components are being overlooked for producing component connections. Plated through hole technology is still being utilized for producing interconnections within PCBs'. (eurocircuits, 2015) (quora, 2015)

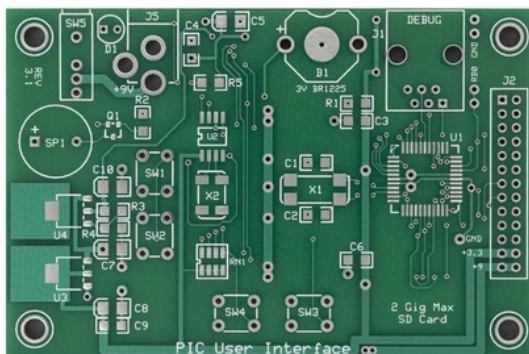


Figure 4 - PTH PCB Board

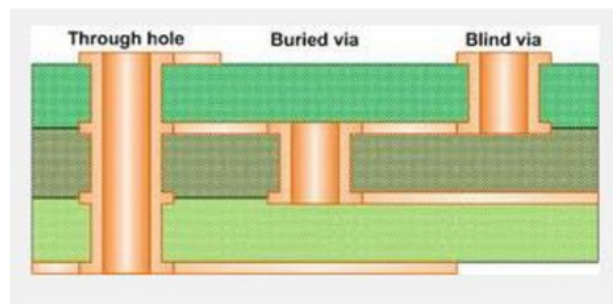


Figure 3 - Multiple Layers of PCB Board

2.3. PCB Process

PCBs' not only support mechanically but also electrically connect the components through conductive tracks and pads that are engraved on the copper layers of the board between a non conductive material. All PCBs are different as many functions can be produced, designers can utilized software such as Altium Designer, OrCAD and Eagle to create a pcb layout which will include conductive tracks and component holes. The design information will then be given to computerized drilling machines or automatic soldering machines to utilized within the manufacturing process.

When the designed circuit is laid out, negative imaging or a mask is printed out identically on clear plastic. (pcbcart.com, 2018) (Madehow.com, 2018)

Two techniques are used when producing large scale soldering. Wave soldering obtains its name through using waves of molten solder to bind components to the PCB that is held within a tank to hold the molten solder. The electronic components are inserted on the PCB where it is then passed through a pumped wave of solder where the solder exposes the metallic areas of the board which then produces a solid mechanical and electrical connection. This method is preferred to than manual soldering as its a lot faster and produces better connections. Wave soldering is used on both Plated through hole and surface mount technologies but in SMT the components are glued to the PCB before the solder wave is run. An advantage of Reflow soldering it can be used for both SMT and PTH components can be fixed on one PCB at the same time.

(epectec.com, 2018) (autodesk, 2016)

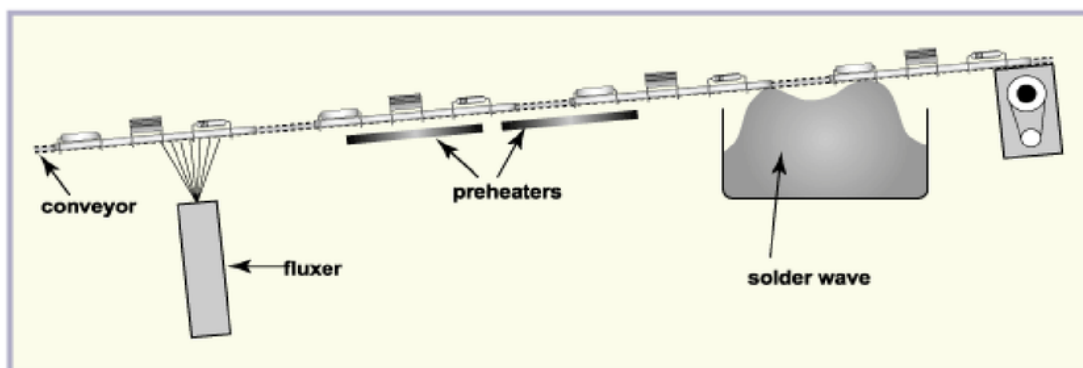


Figure 5 - Wave Soldering Process

Reflow soldering is the preferred method by manufacturers for fixing electronic components to PCBs and 'Reflow Soldering' is a part of the reflow process where the solder deviates from a solid to liquid state under high temperature to enable a solder joint onto the PCB.

There are four stages within reflow profiles, these four stages involve, pre-heating, thermal soaking, reflow soldering and then cooling down.

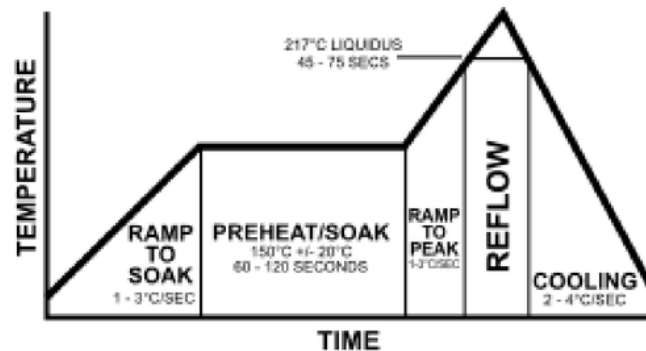


Figure 6 - Reflow Process

(deskdecode, 2017)

The pre-heating stage provides heat to the PCB to bring up the boards temperature allowing the solder flux to melt making the PCB ready to bake. The thermal soak involves rising the temperature to 180 degrees after a minute of preheating which brings the solder to near melting point.

During the reflow solder stage, the temperature becomes higher than the solders melting point which then changes the solder from solid to liquid, allowing the solder to fill the gaps and produce a connection between component and conductive track. After the reflow stages has complete, a cooling stage is then implemented which involves bringing the temperature down gradually thus changing the solder state from liquid back to solid to implement the connection permanently.

(deskdecode, 2017)

2.5. Automatic Optical Inspection

Automatic optical inspection (AOI) is an important technique utilized within the manufacturing and testing of electronic printed circuits or PCBs'. AOI allows quick and accurate scrutiny of the electronics assemblies to ensure the product has been put together with a high quality and that the PCBs' are built correctly and free from any faults. With the complexity of the boards increasing, manual inspection was no longer good enough as the market for electronics was growing rapidly in high volumes, it was no longer a viable option as the inspectors grew tedious not building the boards correctly and errors were easily missed.

With the introduction of AOI, quality products were readily available as it is a much faster technique where costs were reduced as low as possible as faults were detected early in the assembly process (electronics-notes.com, 2015)

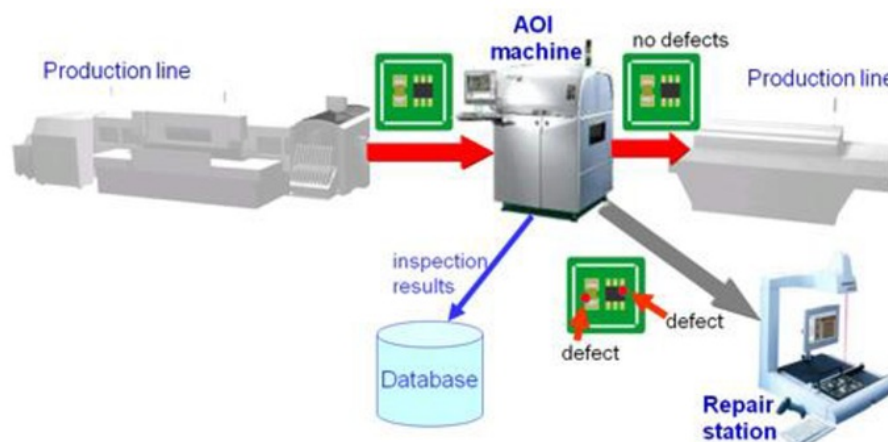


Figure 7 - Automatic Optical Inspection process

(miselectronics.com, 2017)

Detection systems are most commonly found directly after the soldering process as this will catch errors at the earliest stage and can be resolved. As errors found later in the production line its more costly to fix.

Template matching, Pattern Matching and Statistical pattern matching are the analysis techniques used to ensure the PCB has no defects. Template matching is where a perfect template or

'Golden Board' with no defects is passed through the AOI first so that the software can capture the relevant attributes and compare each board to the template. (electronics-notes.com, 2015)

Pattern matching technique stores the pass and fail information of the PCB production lines and then compares the obtained visual of the PCB to the pass/ fail information. Statistical pattern matching technique is a related technique to pattern matching although it uses a statistical method of dealing with problems. It accommodates minimal deviations without flagging the PCB as a fault. (electronics-notes.com, 2015)

Three different lighting options are used within AOI these are Fluorescent, LED and Infra-red where the fluorescent light breaks down over time meaning that the AOI will have compromise over the different lights. LED lights maintain a more stable light as they do not breakdown as much over time and Infrared lighting is used to heighten the visibility of a defect

(electronics-notes.com, 2015) (dcbautomation, 2014)



Figure 8 - Automatic Optical Inspection

2.6. Automatic X-Ray Inspection

The majority of PCBs' at present are using BGA (Ball grid array) integrated circuits and CSPs which means that the solder connections aren't visible. AXI technology has ascended due to the growing numbers of interconnections within the integrated circuits which leads to the increasing complexity of PCBs. As there is more interconnections, it has become necessary to use AXI as it can check the solder joints as well as detect defects with the solder that would prove difficult to see using optical inspection methods

The requirement of automatic x-ray inspection has risen considerably in recent years. Due to this rise, a broader array of equipment is available. This has allowed the techniques involved within the AXI equipment to improve and permitted even better levels of detection to be obtained for PCB manufacture. AXI inspection is not only limited to 2D techniques but machines are now utilizing 3D technology to improve performance.

(electronic-notes, 2015)

Just like the AOI method, AXI systems ideal place would be situated immediately after the soldering process. AXI technology has a prime advantage over AOI as it doesn't only see the solder joint, it is able to see the joints underneath ball grid array, CSPs where soldering is out of sight.

As AXI systems can see an internal view of the solder, it also means that it can give additional information with the soldering such as thickness of joints and sizes of joints and produce a template on specific boards. This technology can then used to compare data on the soldering process and see how well the soldering is functioning. Unlike AOI Systems, AXI systems can detect the "heel" of the solder joint where AOI systems aren't as the leads of the components are blocking.

(electronic-notes, 2015)

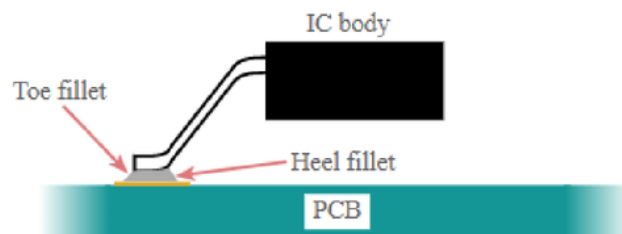


Figure 9 - AXI Technology able to see Heel joint of PCB

When AXI or AOI systems are utilized within the manufacturing of electronic PCBs, the defects and information can be analysed quickly where the process can be tweaked to limit the amount of defects, thus, improving the quality of the process. There is not only two methods AXI and AOI available for finding defects on PCBs. ICT (In-circuit testing) can also be utilized where the same if not similar information is acquired. A comparison table is shown below between the three methods used within ATE. (electronic-notes, 2015)

COMPARISON OF THE MAJOR DEFECT DETECTION CAPABILITIES OF AOI, AXI AND ICT			
DEFECT TYPE	AOI	AXI	ICT
Soldering defects			
Open circuits	Y	Y	Y
Solder bridges	Y	Y	Y
Solder shorts	Y	Y	Y
Insufficient solder	Y (not heel of joint)	Y	N
Solder void	N	Y	N
Excess solder	Y	Y	N
Solder quality	N	Y	N
Component defects			
Lifted lead	Y	Y	Y
Missing component	Y	Y	Y
Misaligned or misplaced component	Y	Y	Y
Incorrect component value	N	N	Y
Faulty component	N	N	Y
BGA and CSP defects			
BGA shorts	N	Y	Y
BGA open circuit connections	N	Y	Y

Figure 10 - Comparison of detection systems

(electronic-notes, 2015)

The research is comprehensive and has a good variety of ATE use in production industry.

3. NI MultiSIM CAR Design

This section of the report will demonstrate the development of a 4 Bit Cellular Automata Registers (CAR) using both 90 and 150 rules using Linear State Feedback regulators (LSFRs') and Xor gates. The two rules will then be combined to form a 4Bit hybrid 90/150 CAR and progressed to a 6 bit hybrid 90/150 CAR to achieve a random pattern generator at maximum length of 63 bits with no repeated values a. A Cellular Automata Register is a circuit involving flip flops that's next state is given by the previous value as well as the value of the neighboring flip flop. A circuit of a CAR can be found in Figure 12. For Pseudo-number generators to begin the random sequence, it requires a 'seed' value. A seed value is an integer number that is used to create a unique set of results for the same sequence, the seed number used through the assignment is 1 where each sequence begins at 1 to allow comparison

3.1. 4 Bit CAR 90 Rule Development

Studying Wolframs CAR90 Rule, The next state of a flip flop is obtained through the output of the neighbouring flip flops being inputted through an X_or gate. Thus, Giving the expression of i.e. $s_i(t+1) = s_{i+1}(t) \text{ xor } s_{i-1}(t)$ however, this rule is unable to meet the maximum length of bits of a random pattern generator.

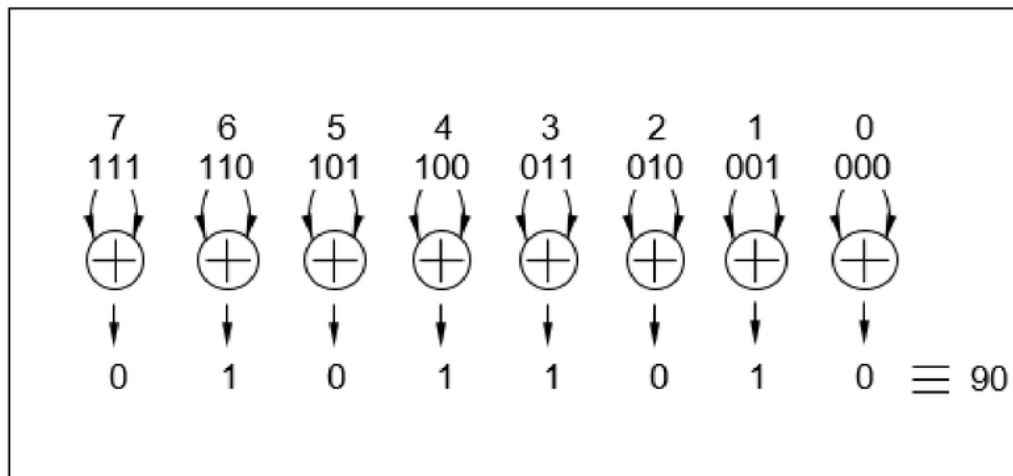


Figure 11 - Wolframs CAR 90 Rule

Figure 12 illustrates the CAR90 circuit on MultiSim

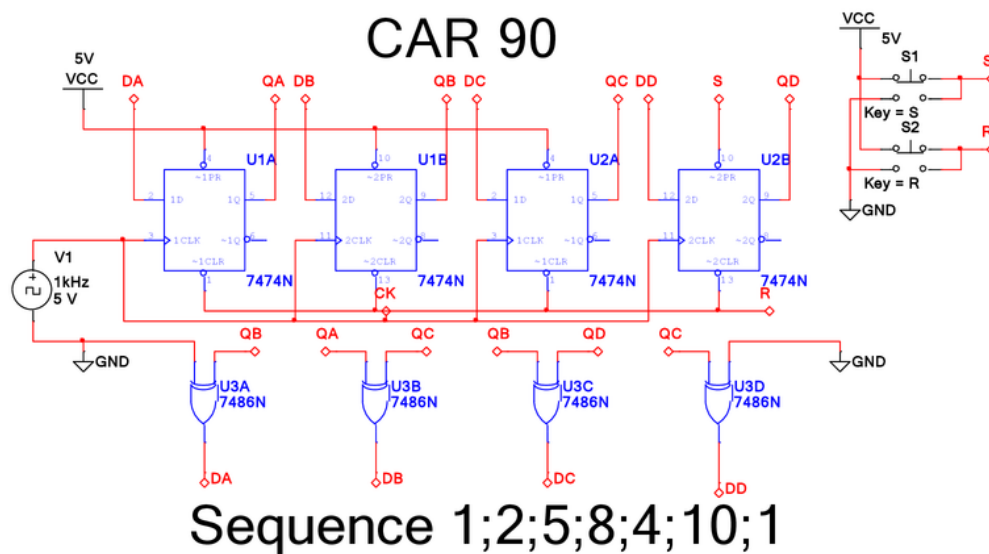


Figure 12 - MultiSIM CAR90 Circuit and Sequence

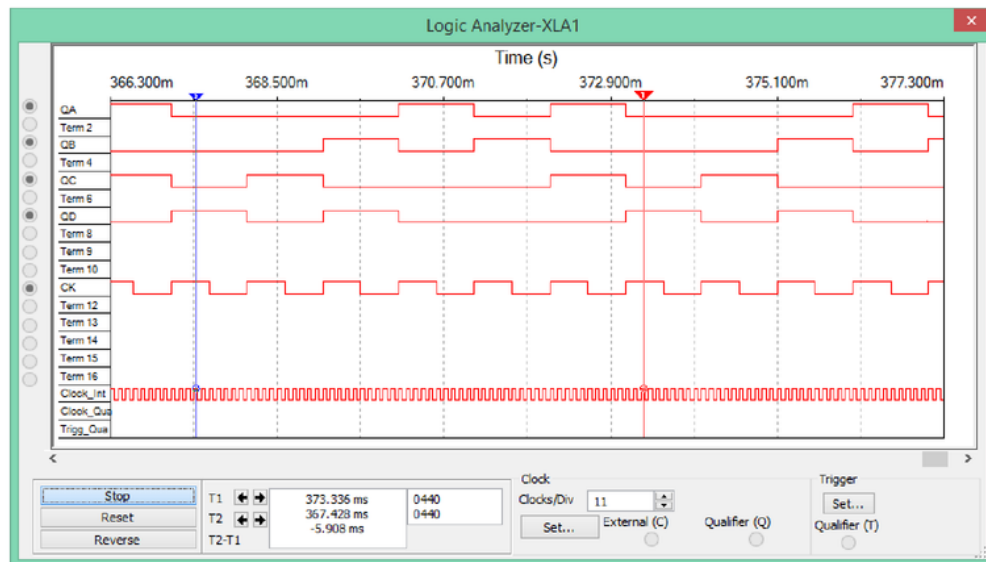


Figure 13 - MultiSIM CAR90 Logic Analyser Result

Figure 13 illustrates the sequence of the 4 Bit CAR90 with QD the least significant bit. It can be seen from the analyser result between the two cursors, the bit pattern generated is 1, 2, 5, 8, 4, 10 and 1 with the sequence restarting. This proves that the 4 bit CAR90 isn't capable of producing a random pattern sequence of maximum count of 15

3.2. 4 Bit CAR 150 Rules Development

Studying Wolframs CAR150 Rule, the next state of the Flip Flop is obtained through the output of neighboring flip flops and its own being inputted to an X_Or gate and then being fed back into its own input. Thus giving the expression of $s_i(t + 1) = s_i + 1(t) \text{ x_or } s_i(t) \text{ x_or } s_{i-1}(t)$. However, like the CAR 90 Rule, the CAR 150 is not able to produce maximum length random bit pattern sequence.

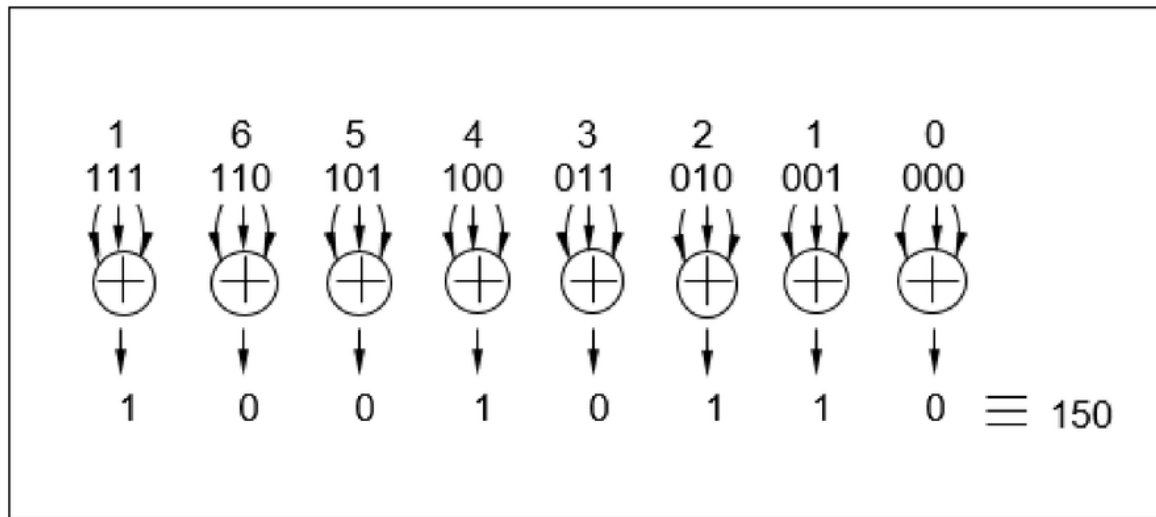
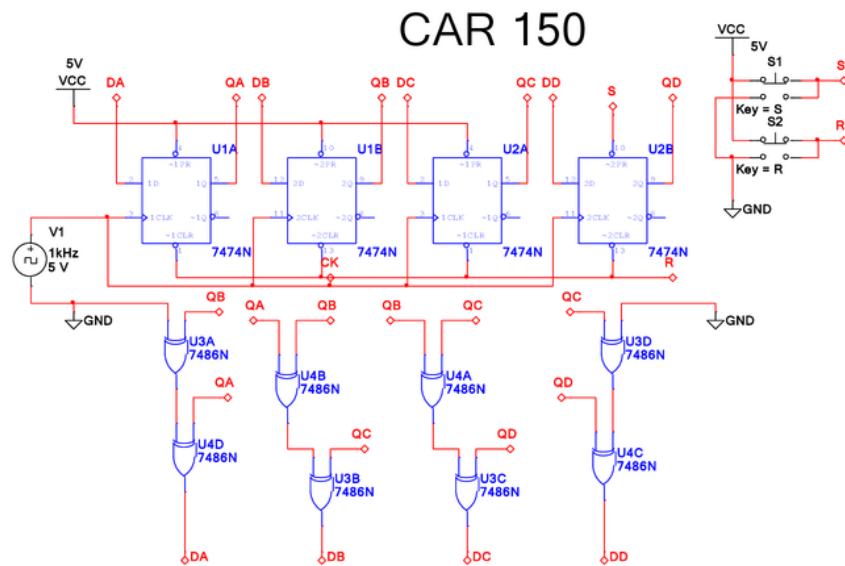


Figure 14 - Wolframs CAR 150 Rule



Sequence 1;3;4;14;5;13;1

Figure 15 - MultiSIM CAR150 Circuit and Sequence

Figure 15 conveys the MultiSIM circuit of the CAR 150 Rule. Figure 16 conveys the sequence of the 4Bit CAR150 from a logic analyser with QD the lsb. The two cursors show the total sequence where a bit pattern of 1, 3, 4, 14, 5, 13, is generated before starting again. The result shown proves that the 150 Rule is unable to produce a random pattern of maximum length 15 bits.

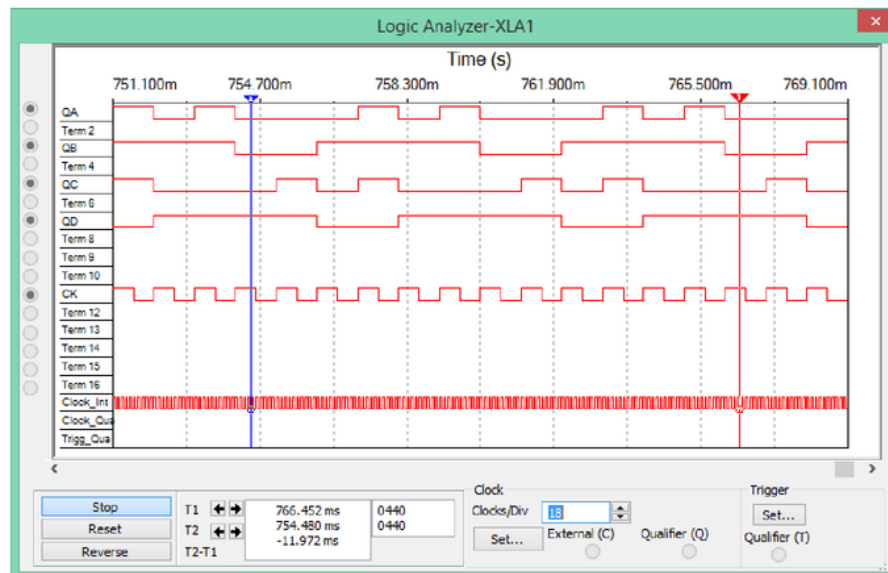


Figure 16 - MultiSIM CAR150 Logic Analyser Results

The theory is explained well and this is a good method of tying in the practical with theory. The point has been made here that the independent rules do not have max count practically.

3.3. 4 Bit Hybrid CAR 90/150 Development

As each rule singularly cannot produce a pattern of maximum length, to achieve a pattern of maximum length, both rules can be combined to create a 4 bit Hybrid CAR 90/150. This will be configured as CAR90 and CAR150 alternating in turn which is able to produce a random number generator of maximum length of 15. Figure 17 conveys the hybrid circuit.

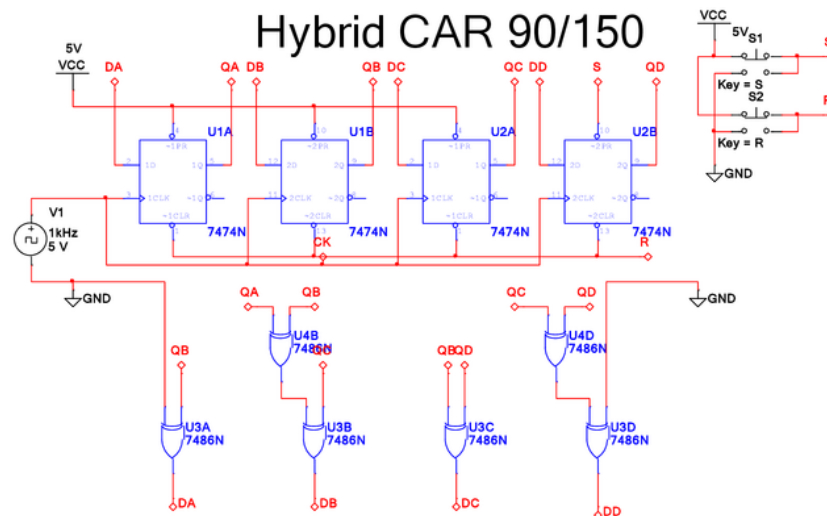


Figure 17 - MultiSIM Hybrid CAR90/150 Circuit and Sequence

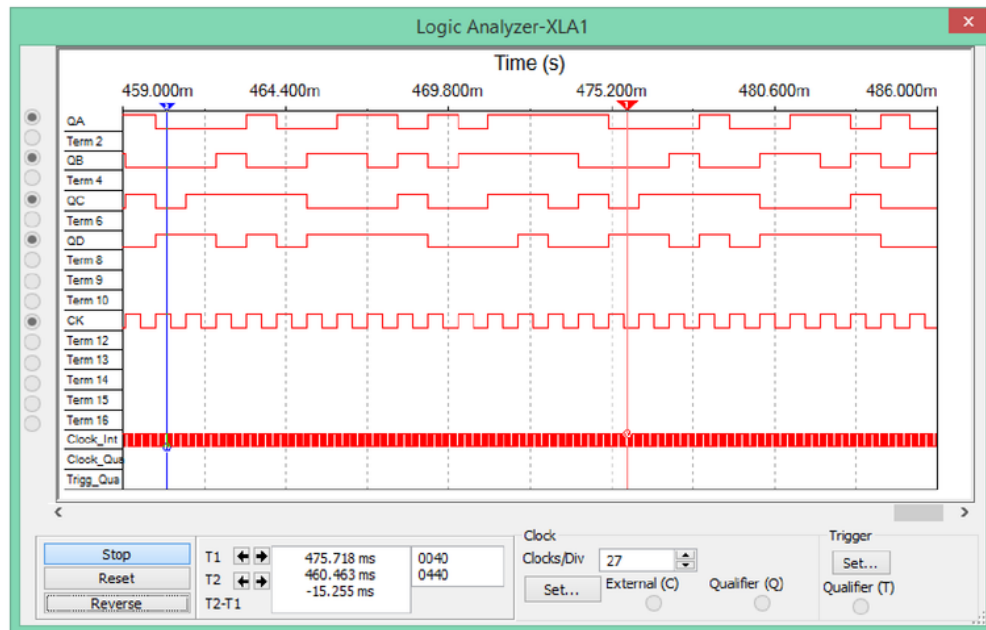


Figure 18 – MultiSIM 4Bit Hybrid CAR90/150 Logic Analyser Results

Figure 18 displays the sequence of the 4Bit hybrid CAR90/150 on the logic analyser. The two cursors illustrate the length of the random pattern generated before it repeats. The bit pattern generated is 1, 3, 6, 11, 2, 5, 13, 9, 7, 8, 4, 14, 15, 12, 10 before repeating.

The result displayed demonstrates that the two rules applied singularly is unable to produce a bit pattern of maximum length but when both rules are applied together then a bit pattern of maximum bit count of 15 can be achieved.

*Test shows the max
count achieved by the
4 bit hybrid circuit.*

3.4. 6Bit Hybrid CAR 90/ 150 Development

After a 4bit Hybrid was produced and simulated with results proving the combination between two rules is able to produce a maximum length random number generator, the next step was to design a 6 Bit Hybrid CAR90/150 to produce a random bit pattern at a maximum length of 63 bits without any repeating bit values.

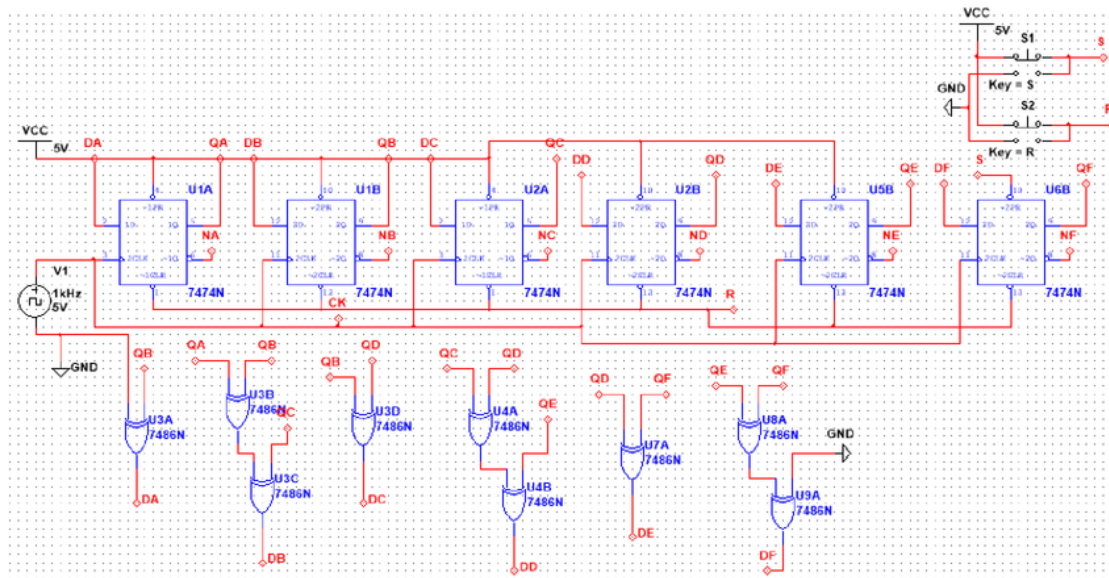


Figure 19 - MultiSIM Circuit of 6Bit Hybrid CAR90/150

Figure 19 illustrates the circuit for the 6Bit Hybrid CAR90/150, it also shows that a 6bit hybrid CAR90/150 can be produced by adding two additional flip flops and X_or gates using the hybrid rule.

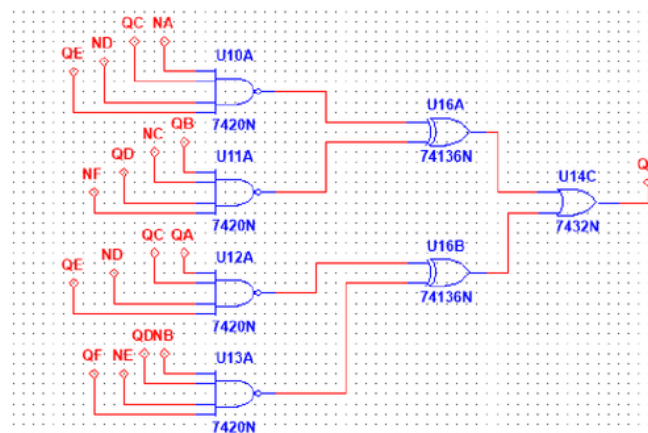


Figure 20 – 6 bit hybrid CAR90/150 Circuit with X_Ors

This is the test cct for comparison with DUT.

A Boolean expression was produced to give the desired outcome of a pattern of maximum length and no repeats so that the design can be tested against to ensure that it works correctly, the expression is shown below.

$$\text{Desired Circuit} = NA.QC.ND.QE + QB.NC.QD.NF + QA.QC.ND.QE + NB.QD.NE.QF$$

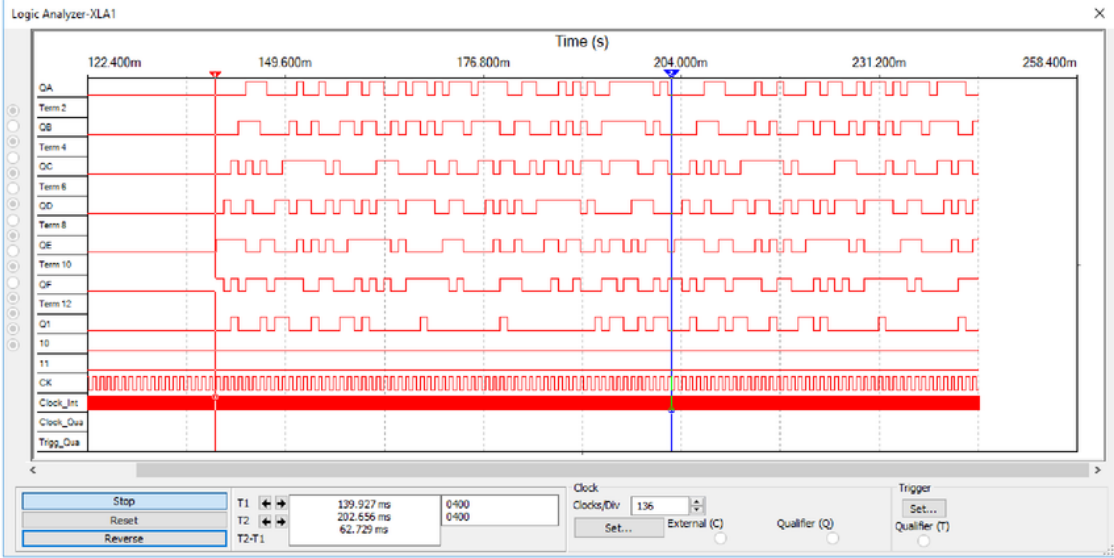


Figure 21 - MultiSIM 6Bit Hybrid CAR90/150 Logic Analyser Results

Figure 21 conveys the logic analyser results of the 6Bit hybrid CAR90/150 where QF is the least significant bit. The two cursors display the length of the random pattern generated by the 6bit hybrid. To ensure that the 6bit hybrid works correctly generating a pattern of 63 bits without repeats, a cursor was used to read the bit count. The value of the bit count was displayed on the logic analyser where it has been blown up and is shown in Figure 22

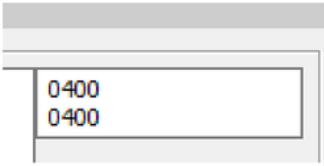


Figure 22 - Logic analyser Cursor bit count display

400 represents the first count of the bit sequence, the top row is for cursor 1 and bottom for cursor 2. Three digits are used to count the binary value, from left to right, the first digit is unused. 2nd digit represents the bit count of QF and QE, 3rd digit QC and QD and finally 4th digit QA and QB. When QF, QD and QB singularly are high on logic analyser, a value of 4 is given, when QE, QC and QA are singularly high a value of 1 is given and when both are high a joint value of 5 is given.

A spreadsheet was produced logging each bit change throughout the sequence with QF being the LSB (1) QE = 2, QD = 4, QC = 8, QB = 16, QA the MSB (32). Logic title refers to logic acquired, where bit value refers to the binary value where Qa+ is for QF & QE, Qb + is for QC & QD and Qc+ is for QA & QB where a total bit count is summed up in the total bit section,

This table is shown below in Figure 23.

Logic			Bit Value			Total bits	Logic			Bit Value			Total bits	Logic			Bit Value			Total bits	Logic			Bit Value			Total bits
qa	qb	qc	qa+	qb+	qc+		qa	qb	qc	qa+	qb+	qc+		qa	qb	qc	qa+	qb+	qc+		qa	qb	qc	qa+	qb+	qc+	
4	0	0	1	0	0	1	5	4	0	3	4	0	7	5	0	4	3	0	16	19	0	4	1	0	4	32	36
5	0	0	3	0	0	3	0	1	0	0	8	0	8	1	5	5	2	12	48	62	1	5	4	2	12	16	30
1	4	0	2	4	0	6	0	4	4	0	4	16	20	5	4	5	3	4	48	55	5	4	1	3	4	32	39
5	1	0	3	8	0	11	1	4	5	2	4	48	54	0	0	1	0	0	32	32	0	1	4	0	8	16	24
1	0	4	2	0	16	18	5	0	1	3	0	32	35	0	0	4	0	0	16	16	0	5	1	0	12	32	44
4	5	5	1	12	48	61	1	4	4	2	4	16	22	0	1	5	0	8	48	56	1	1	0	2	8	0	10
4	0	5	1	0	48	49	5	0	5	3	0	48	51	0	5	5	0	12	48	60	4	0	4	1	0	16	17
5	1	1	3	8	32	43	1	5	1	2	12	32	46	1	0	5	2	0	48	50	5	1	5	3	8	48	59
1	0	0	2	0	0	2	5	5	0	3	12	0	15	4	5	1	1	12	32	45	1	1	5	2	8	48	58
4	4	0	1	4	0	5	0	5	4	0	12	16	28	4	1	0	1	8	0	9	4	1	5	1	8	48	57
4	5	0	1	12	0	13	1	0	1	2	0	32	34	5	4	4	3	4	16	23	5	5	5	3	12	48	63
4	1	4	1	8	16	25	4	4	4	1	4	16	21	0	0	5	0	0	48	48	0	4	5	0	4	48	52
5	5	1	3	12	32	47	4	4	5	1	4	48	53	0	1	1	0	8	32	40	1	4	1	2	4	32	38
0	5	0	0	12	0	12	4	1	1	1	8	32	41	0	4	0	0	4	0	4	5	1	4	3	8	16	27
1	1	4	2	8	16	26	4	5	4	1	12	16	29	1	5	0	2	12	0	14	1	1	1	2	8	32	42
4	1	1	1	8	32	41	4	0	1	1	0	32	33	5	5	4	3	12	16	31	4	0	0	1	0	0	1

Figure 23 - MultiSIM Logic Analyzer Bit Count table

The results shown in Figure 23 display that the 6bit Hybrid CAR90/150 generates a random pattern to a maximum length of 63 bits without any repeats until after the maximum length has been achieved, where the sequence starts again.

Good
test of
maximum
count
for 6
bit
hybrid
circuit.

3.5. Testing the NI MultiSIM 6Bit Hybrid 90/150 CAR

With the 6bit hybrid operating correctly producing maximum length patterns without repeats, three identical circuits were produced however, each circuit had one fault placed amongst the logic gates. This testing stage will compare the outcome of the desired circuit logic and the three test circuits with individual faults in the logic where the two test circuits outcome will also be compared through an X_OR gate. The X_OR Gate will determine if the two outcomes are the same or if there is a difference between the two where a logic 1 present of X_OR will indicate there is a difference, logic 0 if outcomes are the same. No differences = pass, difference = fail.

3.5.1. Test Circuit Fault 1

Figure 24 conveys the original (Q1) against the first test circuit (Q2) with a comparison X_OR gate. The first test circuit was to change the original Boolean expression by adding the 4th input of the 4th And gate QF to 0V/Ground. The test circuit Boolean is shown below with '0' highlighting the 0V.

$$\text{Test Circuit 1} = NA.QC.ND.QE + QB.NC.QD.NF + QA.QC.ND.QE + NB.QD.NE.'0'$$

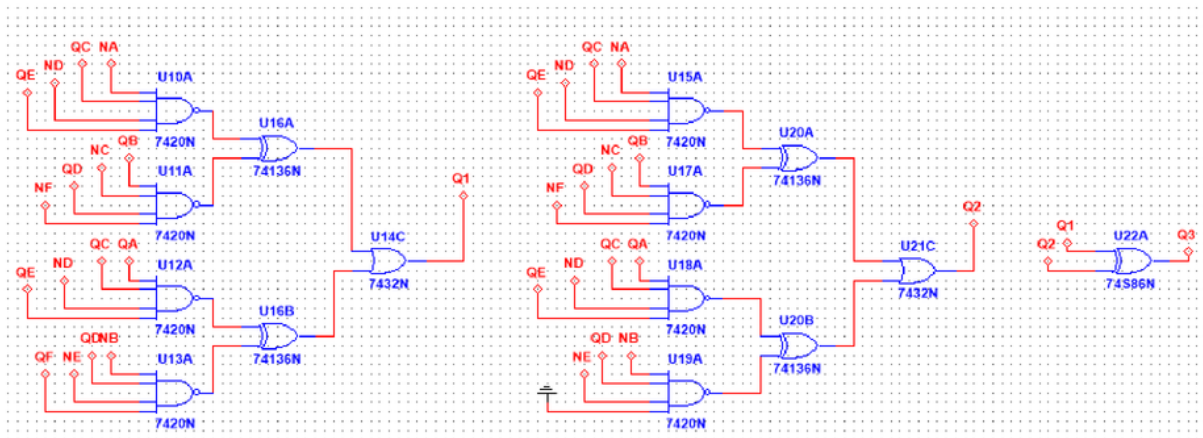


Figure 24 - Test Circuit 1 with Working Circuit

The test cct is correct and includes the all important x-or gate.

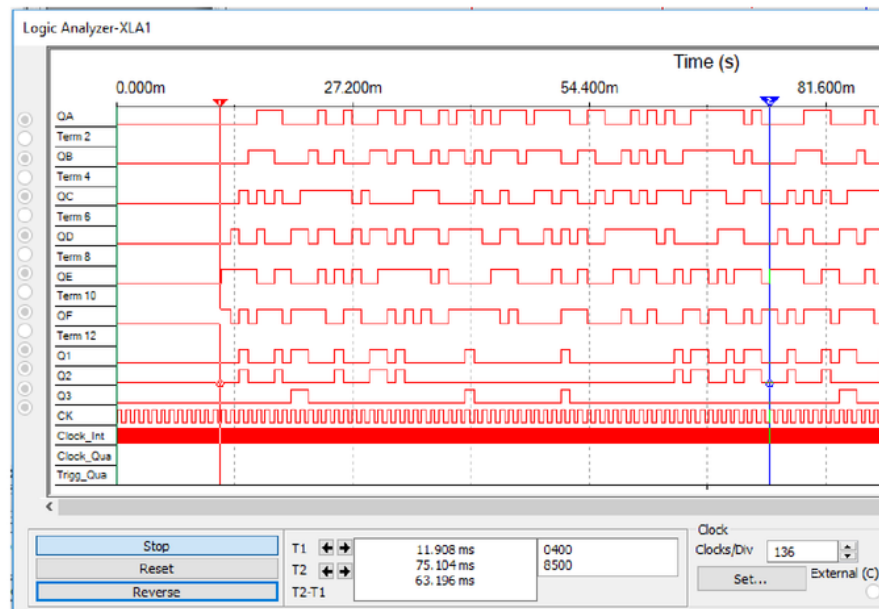


Figure 25 - Test 1 Logic Analyzer Comparison

Figure 25 conveys the result of test 1. The pattern remains the same and the two circuits are shown as Q1 and Q2. It is evident from the analyzer that test circuit 1 fails, this is because the change within the boolean expression causes a different outcome which is highlighted by the logic of Q3 this is shown as when a difference in outcome is detected, a pulse is given from the X_OR gate.

3.5.2. Test Circuit Fault 2

Figure 26 conveys the original (Q1) against the second test circuit (Q2) with a comparison X_OR gate. The second test circuit was to change the boolean expression again by adding first input of the third and gate QA to 0V/Ground. The test circuit Boolean is shown below with '0' highlighting the 0V.

$$\text{Test Circuit 2} = \text{NA.C.ND.E} + \text{B.NC.D.NF} + \text{'0'.C.ND.E} + \text{NB.D.NE.F}$$

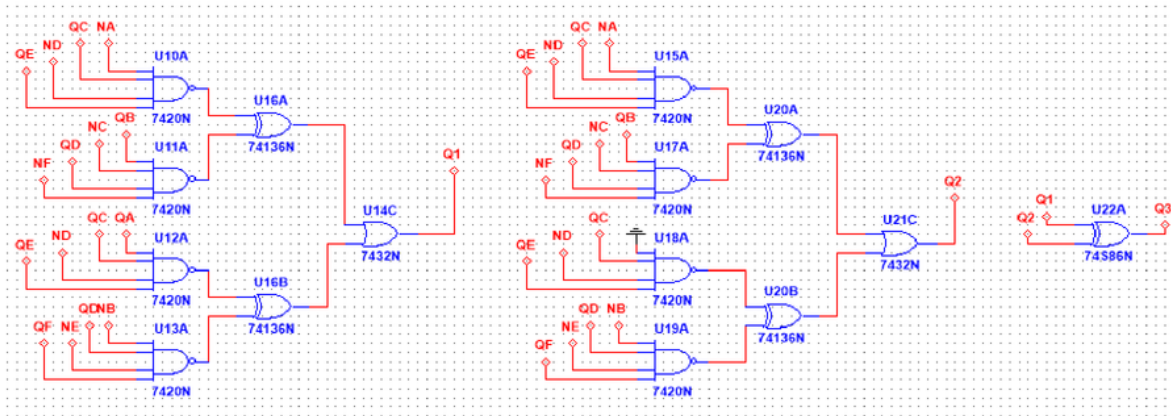


Figure 26 - Test Circuit 2 with working circuit

Figure 26 conveys the change within the test circuit where the first input of the third and gate has been connected to 0V.

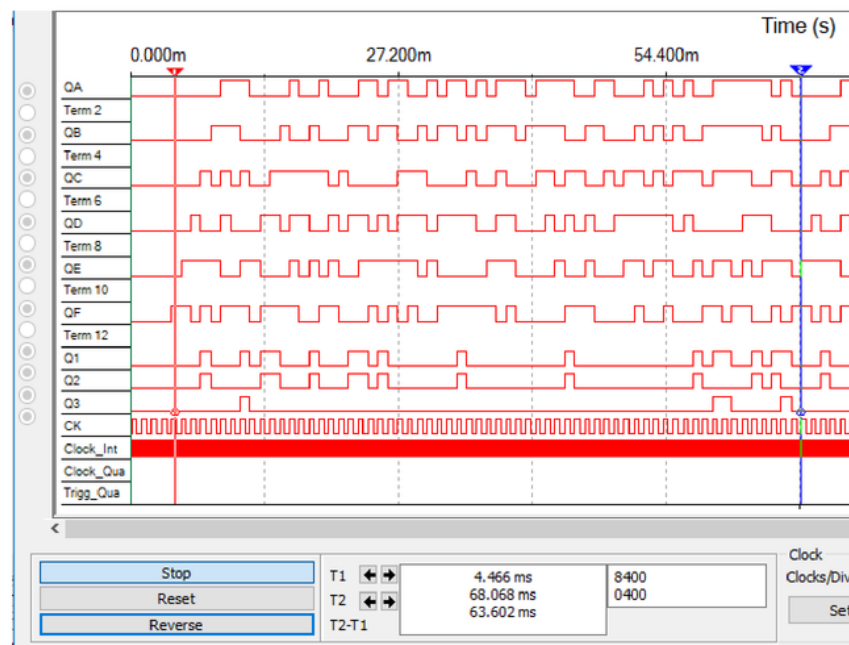


Figure 27 – Test Circuit 2 Logic Analyzer Comparison

Figure 27 shows the logic analyzer result of test circuit 2 in comparison with the desired outcome. The count between QA to QF remains the same however test circuit 2 results in a fail as a difference is highlighted by the X_OR gate highlighted that there is a difference in the outcome of the two circuits.

3.5.3. Test Circuit Fault 3

Figure 29 conveys the original (Q1) against the Final test circuit (Q2) with a comparison X_OR gate. The final test circuit was to change the boolean expression again by connecting last input of the first and gate Q5 to 5V. The test circuit Boolean is shown below with '1' highlighting the 5V.

$$\text{Test Circuit 3} = \text{NA.C.ND.'1'} + \text{B.NC.D.NF} + \text{A.C.ND.E} + \text{NB.D.NE.F}$$

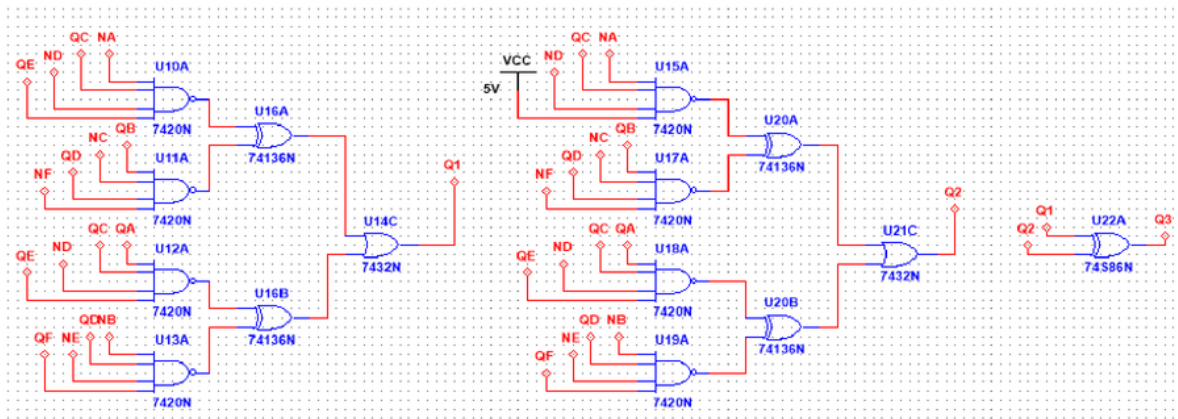


Figure 29 - Test Circuit 3 with working Circuit

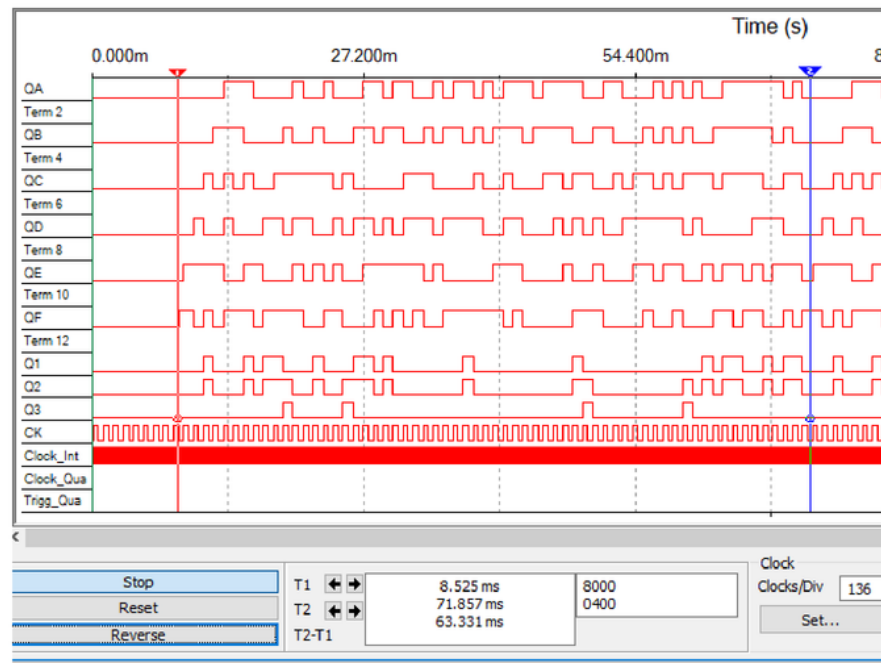


Figure 28 – Test Circuit 3 Logic Analyzer Comparison

Good fault analysis and explanation
of the 3 different scenarios.

Figure 28 illustrates the final test circuit logic analyzer outcome comparison with the desired circuit. The count between QA and QF remains the same however by connecting 5V to an input of an and gate results in the circuit failing as the X_OR gate has indicated that there is a difference between the two outcomes of the two circuits as there are pulses present in the logic where there is a difference.

4. HDL FPGA Design

This section of the assignment illustrates the progression and testing of the same 6Bit Hybrid CAR 90/150 through Vivadoo software using a hardware descriptive language (HDL) where results will be simulated to determine correct operation and then practically by downloading to a FPGA board.

4.1. 4 Bit CAR 90 Development

Figure 30 illustrates the simulation results of the CAR 90 on Vivadoo.

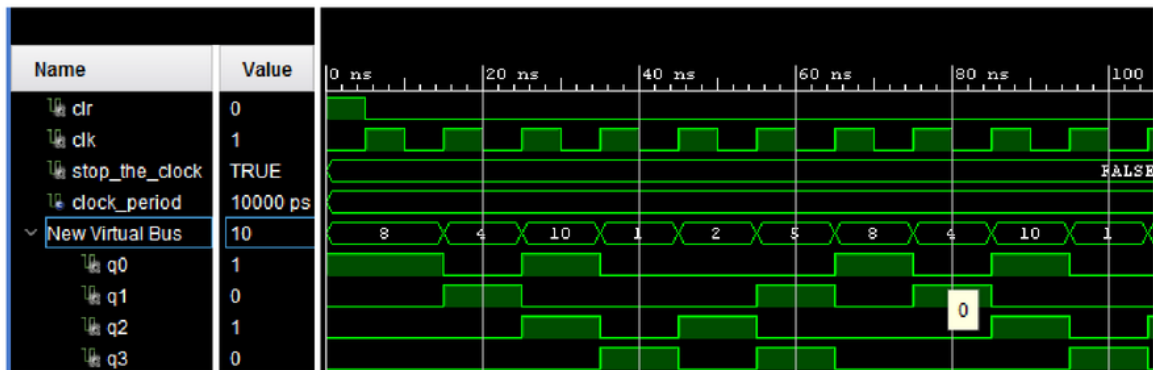


Figure 30 - 4Bit CAR 90 HDL Simulation Results

The simulation results show a reset pulse, clock pulse and then the random pattern bit count. A virtual bus was used which accumulates a total bit count and presents the logic as a decimal number. From the simulation, the pattern generated is identical to that of the MultiSIM CAR 90 in the sequence produced and that it doesn't achieve a pattern of maximum bit and there are repeated bits present.

4.2. 4 Bit CAR 150 Development

Figure 31 illustrates the simulation results of the CAR 150 on Vivadoo.

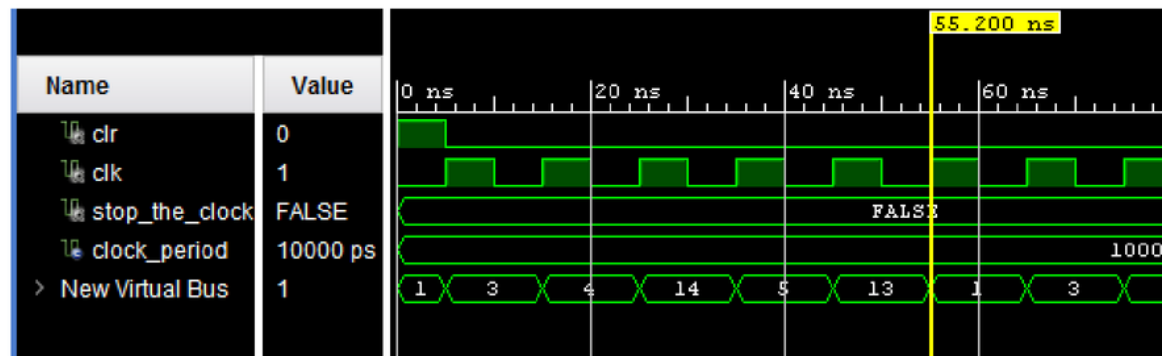


Figure 31 - 4Bit CAR 150 HDL Simulation Results

From the simulation results shown, the CAR 150 does not generate a random pattern of maximum length without repeating, the bit values have a greater range to meeting the maximum length count however a maximum length isn't generated produce a pattern identical to the CAR 150 of the MultiSIM Design.

The individual rules 90 and 150 show that the max count is not achieved using FPGA VHDL.

4.3. 4 Bit Hybrid CAR 90/150 Development

Figure 32 and Figure 33 illustrates the simulation results of the 4Bit Hybrid CAR 90/150 on Vivadoo.

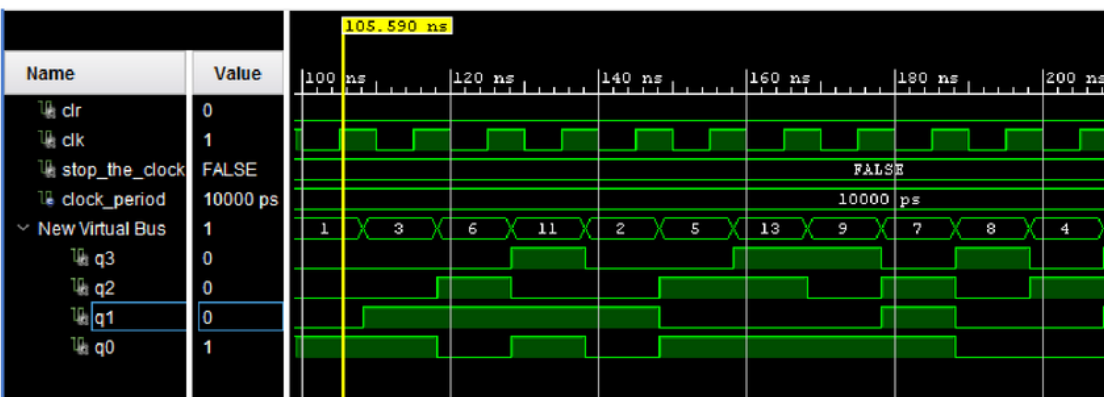


Figure 32 - 4Bit CAR 90/150 Hybrid HDL Simulation Results Part 1

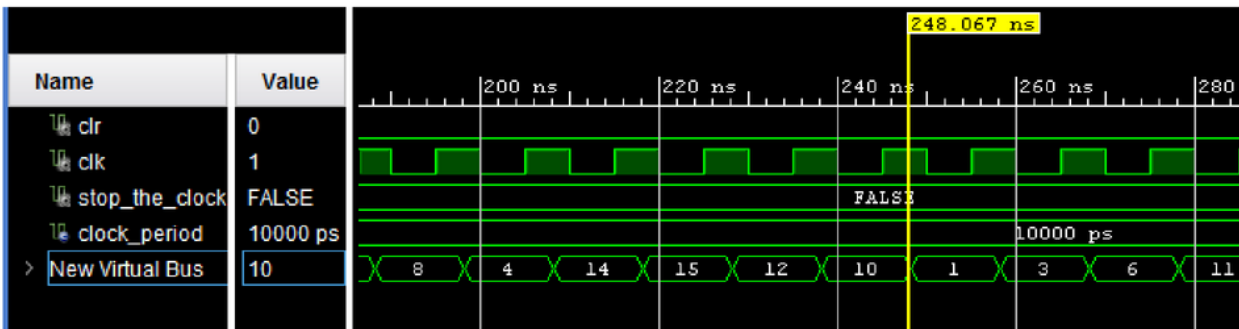


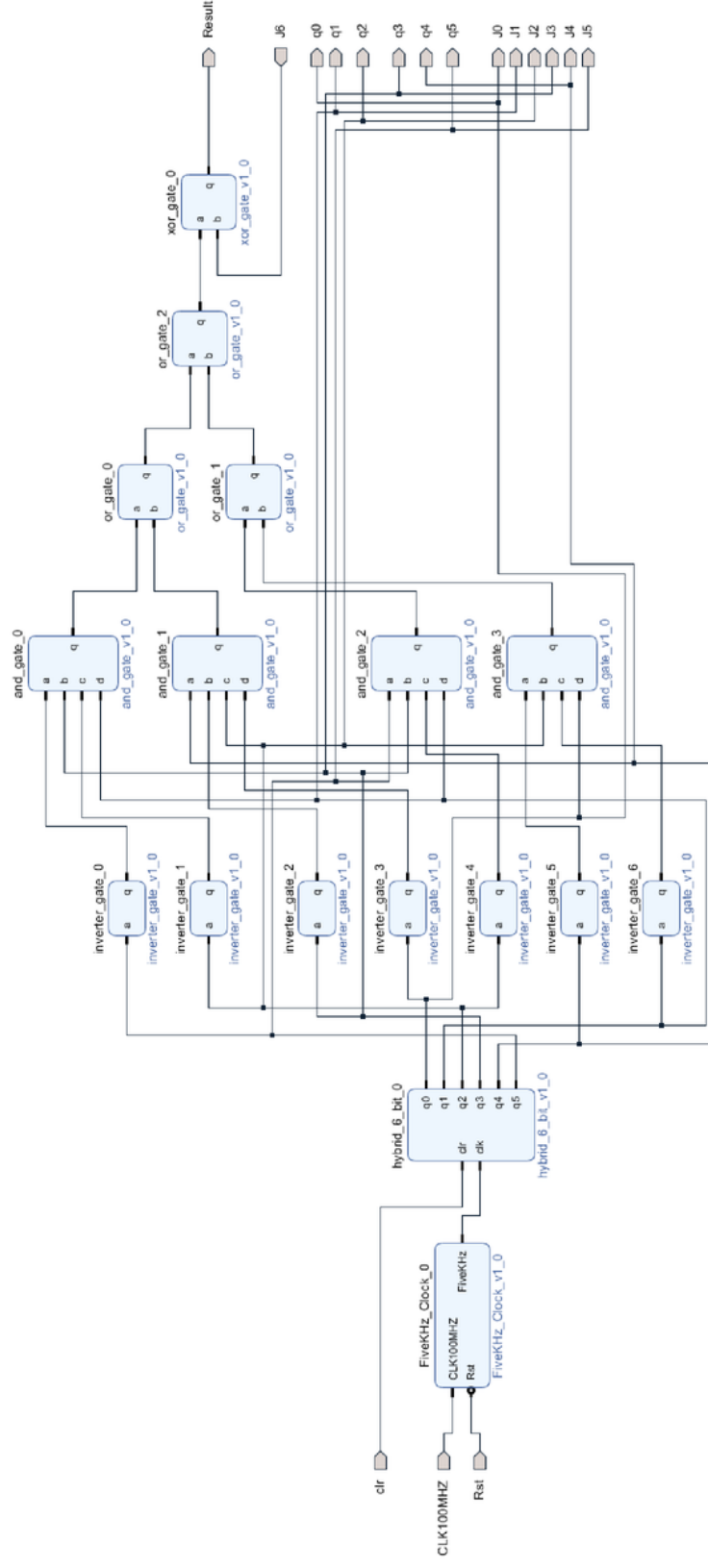
Figure 33 - 4Bit CAR 90/150 Hybrid HDL Simulation Results Part 2

The simulation results indicate that combining Wolframs CAR 90 and 150 rules together producing a hybrid CAR, is able to generate a random pattern sequence of maximum length without repeated bits matching that of the MultiSIM design. This is evident as the simulation shows the bits counting from 1 to 15 without repeating, proving that when both rules applied singularly is unable to produce this, and when both rules combined a operational CAR is achievable.

*The 4 bit
hybrid
CAR
shows that
max count
is
achieved.*

4.4. 6 Bit Hybrid CAR 90/150

The HDL code for the 4Bit Hybrid CAR was progressed from 4 bit to the final 6 Bit design where the Vivadoo test circuit was made. The IP Circuit was produced by placing the required components from a register on to the workspace. Below shows the complete circuit that includes the test circuit connecting to the 6 bit register and input/ output ports.



Design is correct and shows the output from the external x-or gate feedback to the VHDL virtual design.

Figure 34 - Vivadoo Circuit

```

-- :
16 : D(5) <= '0' xor qs(4);
17 : D(4) <= qs(5) xor qs(4) xor qs(3);
18 : D(3) <= qs(4) xor qs(2);
19 : D(2) <= qs(3) xor qs(2) xor qs(1);
20 : D(1) <= qs(2) xor qs(0);
21 : D(0) <= qs(1) xor qs(0) xor '0';
22 :

```

The VHDL of the hybrid CAR is identified. The 6 bit conditions are shown here.

Figure 35 - 6Bit Hybrid CAR 90/150 HDL Code

Figure 35 illustrates the HDL Code of the 6Bit Hybrid CAR 90/150. The architecture of the HDL shows 6bit signal where D is the bit signal and 0 to 5 is the signal reference. From lines 16 to 21 is the configuration of the CAR 90/150. The wolfram 90 rule indicates that the output of the flip flop is only affected by its neighboring flip flops where the 150 rule states that the output of the flip flop is dependent on its neighboring flip flop state and its own output as referred to in sections 3.1 and 3.2

It can be seen that the code is written where Signals D5, D3, and D1 are CAR 90 rule as is only X_Ored between two states, where signals D4, D2 and D0 are CAR 150 rule as the three states are X_Ored between three states. The code then progressed combining the two rules after each other.

Once the HDL code was complete, a synthesis was run to ensure the code was ok, and a simulation could be run to test the CAR Hybrid.

Figure 36 illustrates the simulation results of the 6 bit Hybrid CAR 90/150 on Vivadoo.



Figure 36 – 6 Bit Hybrid CAR 90/150 Simulation Results

Test shows that max count has been achieved by the 6 bit CAR.

As shown in the simulation results in Figure 36, the 6Bit Hybrid CAR 90/150 operates correctly as it generates a random pattern of maximum bit length without repeating bits as show by the decimal numbers within the virtual bus. The random pattern matches that of the 6Bit hybrid of the MultiSim Design.

4.5. Testing HDL FPGA 6Bit Hybrid CAR 90/150 on FPGA Board

With the 6Bit CAR 90/150 Hybrids operation proven to work correctly through simulation, the CAR could now be tested physically by downloading the HDL to an FPGA Board as shown in Figure 37.

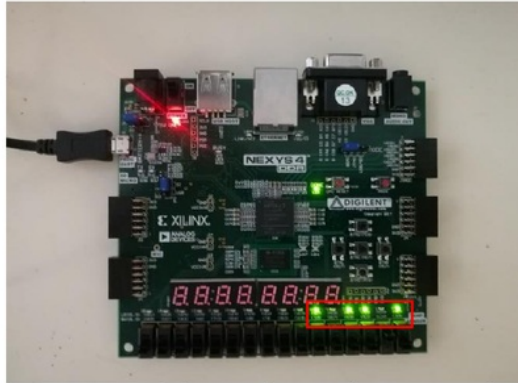


Figure 37 - 6Bit Hybrid CAR 90/150 tested on FPGA Board Pass

With the program set to run, the 6 LEDs in a row represented the patterns bit count with the right end LED representing QA (MSB) and left end LED representing QF (LSB). The combination of LEDs' illuminated would change as each bit count was generated, a fault detection LED was programmed as, this was the far most LED on the FPGA Board. Figure 37 illustrates the CAR has passed as no fault LED was present. The count shown on the board is 45.

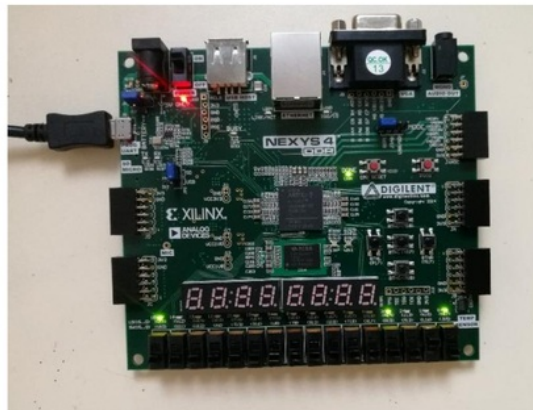


Figure 38 - 6Bit Hybrid CAR 90/150 Tested on FPGA Board Fail

Figure 38 illustrates the CAR highlighting that a fault has occurred within the pattern, this fault is shown by the far left fault LED being illuminated.

4.6. Testing HDL FPGA 6Bit Hybrid CAR 90/150 Against Test Circuits

With downloading the 6Bit hybrid CAR to the FPGA Board and testing its operation and fault detection passing, a test circuit was produced on breadboard that replicated that used in MultiSIM. Three tests were carried out where the same faults from the MultiSIM Testing boolean circuit were replicated in turn on the breadboard to test the operation of the CAR.

4.6.1. Test Circuit Fault 1

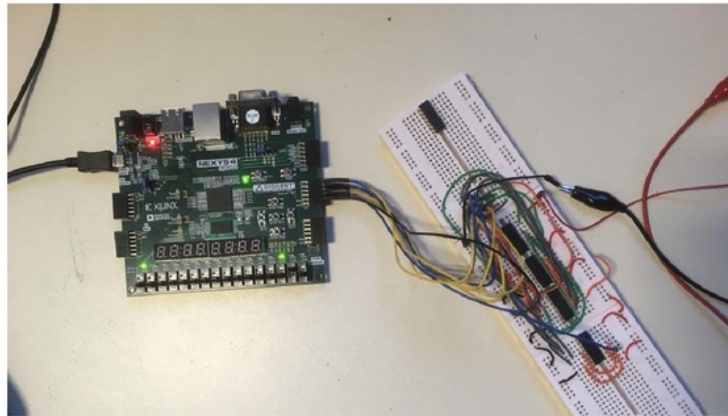


Figure 39 - FPGA Test circuit Fault 1

Figure 39 conveys the result of the CAR detecting a fault within the test circuit boolean as an input of an and gate is connected to 0V.

4.6.2. Test Circuit Fault 2

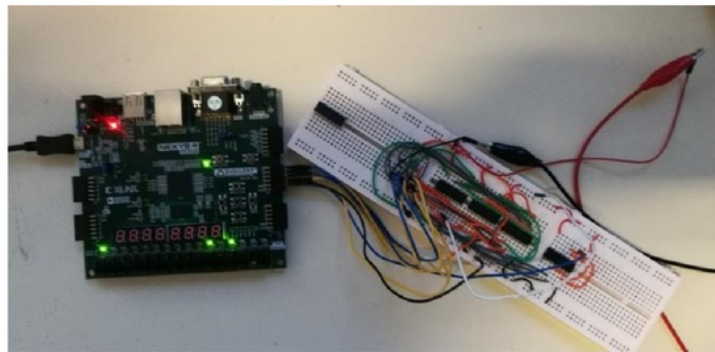


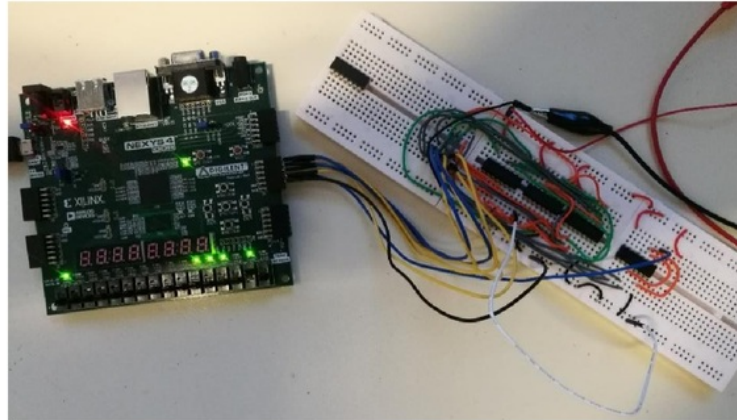
Figure 40 - FPGA Test Circuit Fault 2

6 BIT car test of the practical prototype CUT built on prototype board.

The FPGA shows the fault has been registered, thus proving the ATE 6 bit CAR as an auto test procedure.

Figure 40 displays the result when the test circuit is applied with the 2nd fault where a different and gate input is connected to 0V.

4.6.3. Test Circuit Fault 3



A different fault is registered.

Figure 41 - FPGA Test Circuit Fault 3

Figure 41 shows the result of the CAR when the third fault is placed within the test circuit, as it can be seen on the FPGA board, an input of the test circuit is connected to 5V where a count is displayed by the LEDs' but the fault LED is also present – highlighting a fault.

5. Practical PCB Build and Test

With the 6Bit Hybrid CAR operating correctly having been tested, the next step was to produce a PCB of the test circuit used in the previous tests on PCB where the components will be soldered onto the PCB and tested again to ensure the product meets the specification. Using PCB design software Eagle, the test circuit logic was built and connected using on-page connectors. This is shown below in Figure 42

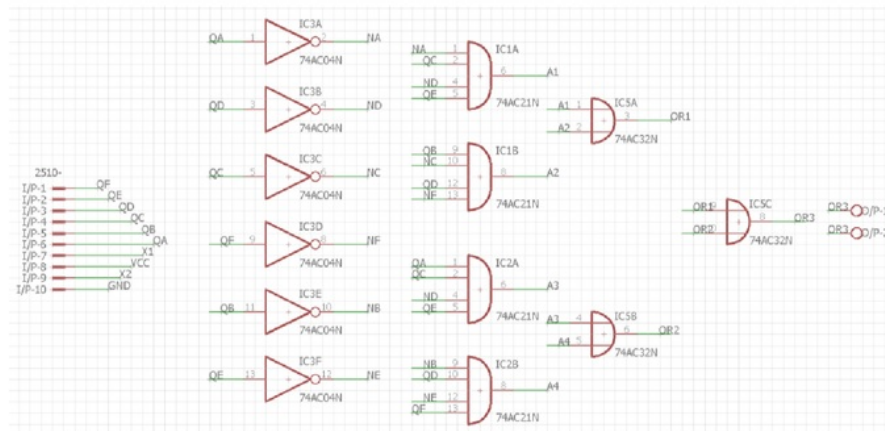


Figure 42 - Logic Circuit produced on Eagle software

The logic circuit shows the test circuit represented on the Eagle software, it can be seen that a 10 input port is present allowing the 6 bit inputs of the CAR to the chip also with a power supply and ground. The 6 not gates, 4 and gates and 3 or gates to 2 output ports.

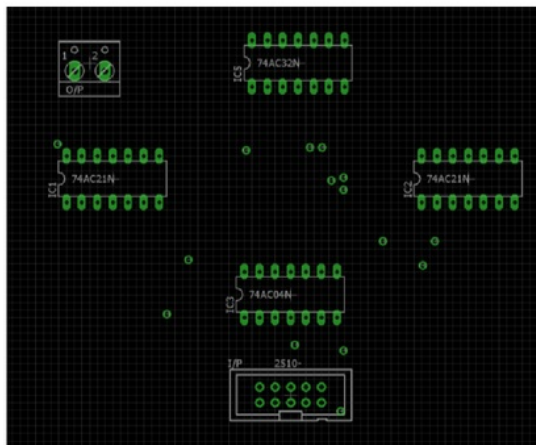


Figure 43 – PCB layout and connection pads

Figure 43 shows the PCB generated on Eagle from the given logic circuit. The circuit shows the layout of the components in white, vias, and connection pads of the components in green

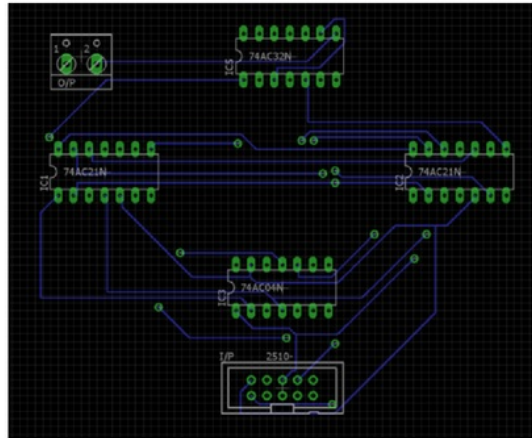


Figure 44 - Bottom Layer tracks of PCB

Figure 44 Shows bottom layer tracks of the PCB Board in blue produced through the on-page connectors within the logic circuit.

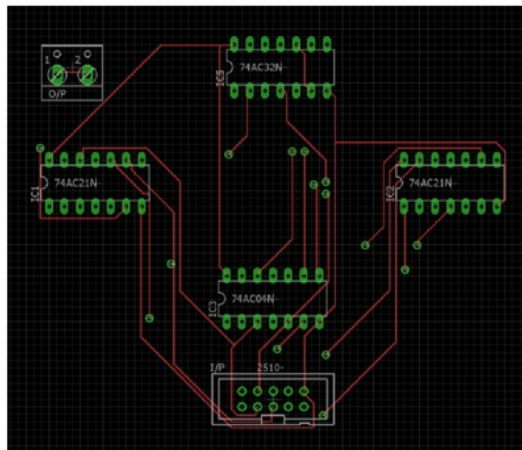
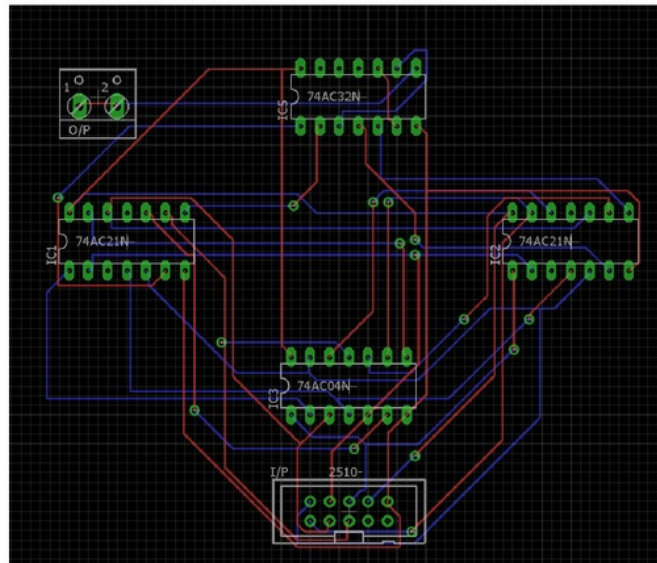


Figure 45 - Top Layer tracks of PCB

Figure 45 displays the top layer tracks on the PCB board in red produced through the on-page connectors within the logic circuit



PCB design is explained and looks correct.

Figure 46 - Completed PCB With All elements included

Figure 46 conveys the complete product all together with all components and track included. The complete PCB can now be built and soldered through a reflow oven.

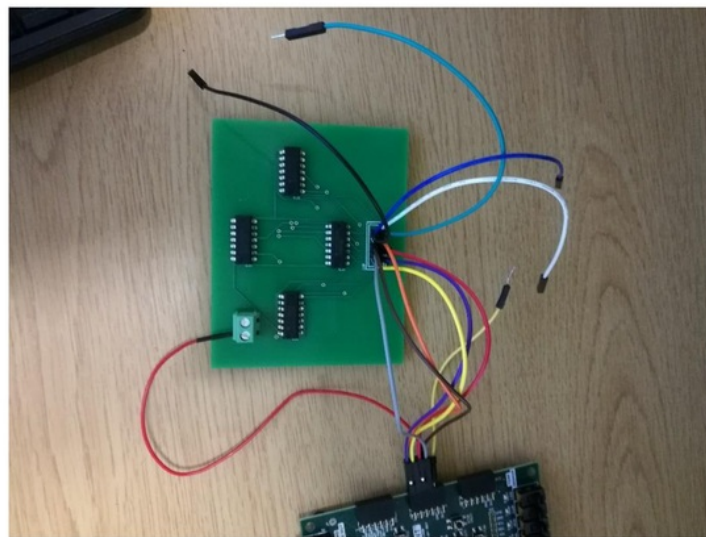


Figure 47 - Manufactured PCB of Test circuit

Figure 47 displays the manufacturers PCB circuit with all components fixed with solder with input and output wires visible to connect to FPGA.

5.1. Testing PCB

With the PCB Manufactured with components fixed down with solder and input/ output wires made, the PCB test circuit was ready to be connected to the FPGA board for final testing of the product.

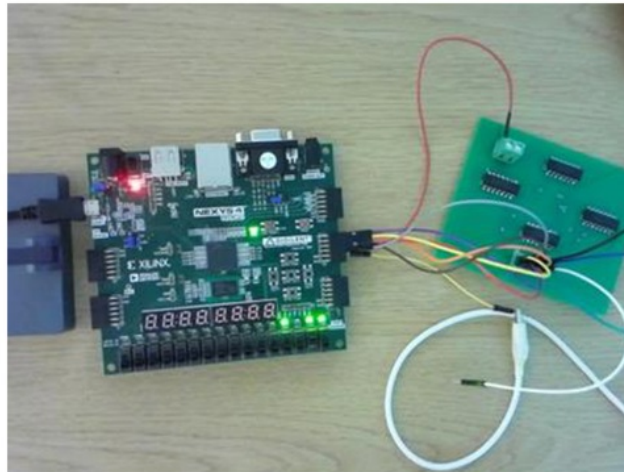


Figure 48 - Final Test stage of 6Bit Hybrid CAR with PCB Test circuit

Test here showing no fault condition.

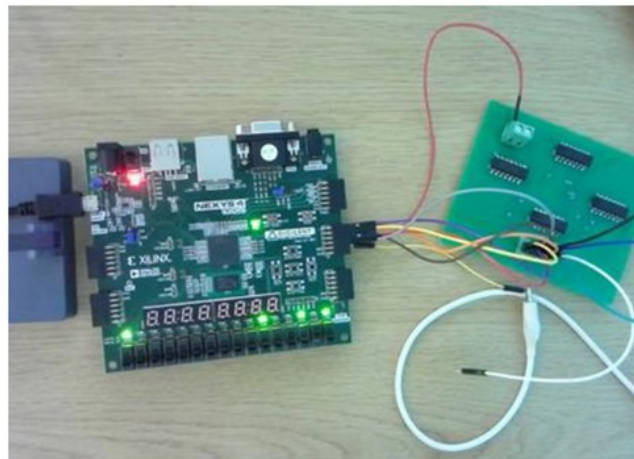


Figure 49 - Final Test stage of 6Bit Hybrid CAR PCB Test with fault

Good example of PCB CUT with fault conditions.

Figure 48 and Figure 49 display the testing of the CAR with the PCB test circuit with the first image showing the CAR Operating correctly with no fault with a bit count of 11. The second image illustrates that a fault has occurred, no change was made to the circuit as PCB components are fixed, fault was detected by turning off the power supply to the test circuit. This proves that the 6Bit Hybrid works correctly to the specification.

6. Analysis

6.1 Analysis of NI MultiSim Design

The three tests carried out on MultiSIM were carried out on the same test circuit however a different fault was placed in each test which included of two different inputs connected to 0V and then 1 input connected to 5V. These faults were used throughout all testing circuits within the assignment.

The progression of the multisim development showing the CAR 90 and 150 rules singularly proved background reading that they are unable to produce a random pattern of maximum length. However, by combining both rules to produce a hybrid showed that a random pattern generator of maximum length can be produced as shown in the results the hybrid produce a maximum count of 15. The results from the finished design of the 6Bit hybrid met the specification as a pattern of maximum length was achieved without repeats as shown in the results table in Figure 23 proving background reading of wolframs rules. The three fault circuits applied to the CAR were also detected by the comparison X_OR Gate when a difference was found. A Multisim comparison results table is shown below

Table 1 - MultiSIM Results Comparison

Boolean output from 6 bit CAR	Boolean required from good circuit	Sim fault result from test circuit	Fault detected?
NA.NB.NC.D.NE.F	NA.C.ND.E + B.NC.D.NF + A.C.ND.E + NB.D.NE.F	NA.C.ND.E + B.NC.D.NF + A.C.ND.E + NB.D.NE.'0'	Yes
A.B.C.ND.E.F	NA.C.ND.E + B.NC.D.NF + A.C.ND.E + NB.D.NE.F	NA.C.ND.E + B.NC.D.NF + '0'.C.ND.E + NB.D.NE.F	Yes
NA.NB.C.ND.NE.F	NA.C.ND.E + B.NC.D.NF + A.C.ND.E + NB.D.NE.F	NA.C.ND.'1' + B.NC.D.NF + A.C.ND.E + NB.D.NE.F	Yes

*Good
use of
table to
show
condition
s of fault
analysis.*

6.2. Analysis of HDL FPGA Design

The CAR development from a 4Bit CAR 90 through to the final 6Bit hybrid CAR 90/150 Product was identical to that of the MultiDesign as the same seed value was used. The simulation Results also proved that a combination between the two 90 and 150 rules will generate a pattern of maximum length.

The HDL FPGA hybrid CAR also met the specification as when applying the three test circuits with individual faults, a comparison X_OR gate was able to detect a difference between the two output logic which was evident within the images of the FPGA Board under test as a fault LED was illuminated. A comparison table of the test results are shown below showing the Boolean expressions and if faults were detected.

Table 2 - FPGA Design Comparison Table

Boolean output from 6 bit CAR	Boolean required from good circuit	Sim fault result from test circuit	Fault detected?
NA.NB.NC.D.NE.F	NA.C.ND.E + B.NC.D.NF + A.C.ND.E + NB.D.NE.F	NA.C.ND.E + B.NC.D.NF + A.C.ND.E + NB.D.NE.'0'	Yes
A.B.C.ND.E.F	NA.C.ND.E + B.NC.D.NF + A.C.ND.E + NB.D.NE.F	NA.C.ND.E + B.NC.D.NF + '0'.C.ND.E + NB.D.NE.F	Yes
NA.NB.C.ND.NE.F	NA.C.ND.E + B.NC.D.NF + A.C.ND.E + NB.D.NE.F	NA.C.ND.'1' + B.NC.D.NF + A.C.ND.E + NB.D.NE.F	Yes

6.3. Analysis of PCB Design

With the test circuit being designed and built on a PCB, the logic circuit was able to be connected to the FPGA Board and tested. As shown in Figure 48 the random pattern generator operated through the sequence without any repeats and no fault LED was illuminated proving the final product satisfied the specification. The power to the test circuit was turned off to generate a fault, which the CAR detected as the fault LED Illuminated as shown in Figure 49.

[Good analysis here and comparison.](#)

7. Conclusions and Recommendations

To conclude this assignment, the aims and objectives set out within the introduction were met as an investigation into PCB design and soldering techniques 2 layer pcbs was carried out. An investigation of automatic test equipment was carried out through learning about techniques of automatic optical inspection and x ray inspection and how they compare against each other when used in manufacturing.

The designed product of a 6bit random pattern generator met the customers specification through progressive simulation using MultiSIM and Vivadoo ensuring that the logic circuit designed operated successfully at maximum length. The design was then manufactured and tested again ensuring that the finished product met the customers specification where the test results met the specification.

*This could be much better
with the focus on the virtual
ATE integration.*

8. References

- airborn.com, 2015. *connector encyclopedia*. [Online]
Available at: <https://www.airborn.com/resources/connector-encyclopedia/smt>
[Accessed 22 January 2018].
- autodesk, 2016. *top 10 soldering issues can ruin pcb design*. [Online]
Available at: <https://www.autodesk.com/products/eagle/blog/top-10-soldering-issues-can-ruin-pcb-design/>
[Accessed 29 January 2018].
- dcbautomation, 2014. *introduction to AOI*. [Online]
Available at: <http://www.dcbautomation.com/wp-content/uploads/2014/09/1-Introduction-to-AOI.pdf>
[Accessed 24 January 2018].
- deskdecode, 2017. *reflow*. [Online]
Available at: <https://www.deskdecode.com/reflow/>
[Accessed 29 January 2018].
- electronic-notes, 2015. *axi x ray inspection*. [Online]
Available at: <https://www.electronics-notes.com/articles/test-methods/automatic-automated-test-ate/axi-x-ray-inspection.php>
[Accessed 29 January 2018].
- electronics-notes.com, 2015. *Automated Optical Inspection*. [Online]
Available at: <https://www.electronics-notes.com/articles/test-methods/automatic-automated-test-ate/aoi-optical-inspection.php>
[Accessed 24 January 2018].
- Electronics-Notes, 2016. *Surface Mount Technology*. [Online]
Available at: https://www.electronics-notes.com/articles/electronic_components/surface-mount-technology-smd-smt/what-is-smt-primer-tutorial.php
[Accessed 22 January 2018].
- epectec.com, 2018. *wave soldering defects*. [Online]
Available at: <http://www.epectec.com/pcb/wave-soldering-defects/>
[Accessed 29 January 2018].
- eurocircuits, 2015. *Plated through hole*. [Online]
Available at: <https://www.eurocircuits.com/holes-npth-non-plated-through-hole-and-ptb-plated-through-hole/>
[Accessed 22 January 2018].

Madehow.com, 2018. *Printed Circuit Board*. [Online]

Available at: <http://www.madehow.com/Volume-2/Printed-Circuit-Board.html>

[Accessed 29 January 2018].

miselectronics.com, 2017. *pcb aoi testing*. [Online]

Available at: <https://www.miselectronics.com/PCB-AOI-Testing.html>

[Accessed 24 January 2018].

pcbcart.com, 2018. *PCB manufacturing Process*. [Online]

Available at: <https://www.pcbcart.com/article/content/PCB-manufacturing-process.html>

[Accessed 29 January 2018].

quora, 2015. *What are the Pros and Cons of blind vias versus Through hole PCB Layout*. [Online]

Available at: <https://www.quora.com/What-are-the-pros-and-cons-of-blind-vias-versus-through-hole-in-PCB-layout-design>

[Accessed 22 January 2018].

Zentech-Manufacturing, 2014. *Surface Mounted Technology What is it all About*. [Online]

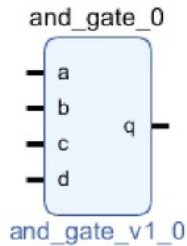
Available at: <http://info.zentech.com/blog/bid/310654/Surface-Mounted-Technology-What-Is-It-All-About>

[Accessed 22 January 2018].

*Refs are good and an
appropriate number for
this task.*

10. Appendix

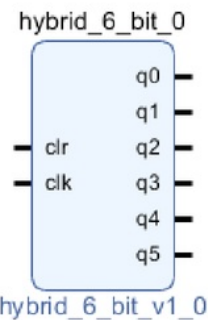
The following images show the HDL of the logic components used within Vivadoo.



```

6 entity and_gate is
7   -- Port ( );
8   Port ( a : in std_logic;
9         b : in std_logic;
10        c : in std_logic;
11        d : in std_logic;
12        q : out std_logic);
13 end and_gate;
14
15 architecture Behavioral of and_gate is
16
17 begin
18   q <= a and b and c and d;
19
20 end Behavioral;

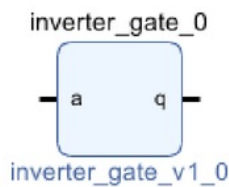
```



```

4 entity hybrid_6_bit is
5   port(
6     clr : in STD_LOGIC;
7     clk : in STD_LOGIC;
8     q0,q1,q2,q3,q4,q5 : out std_logic
9   );
10 end hybrid_6_bit;
11
12 architecture Behavioral of hybrid_6_bit is
13   signal D, qe: STD_LOGIC_VECTOR(5 downto 0);
14   begin
15
16   D(5) <= '0' xor qe(4);
17   D(4) <= qe(5) xor qe(4) xor qe(3);
18   D(3) <= qe(4) xor qe(2);
19   D(2) <= qe(3) xor qe(2) xor qe(1);
20   D(1) <= qe(2) xor qe(0);
21   D(0) <= qe(1) xor qe(0) xor '0';
22
23   -- Three D flip-flops
24   process(clk, clr)
25   begin
26     if clr = '1' then
27       qe <= "000001";
28     elsif clk'event and clk = '1' then
29       qe <= D;
30     end if;
31   end process;
32   q0 <= qe(0);
33   q1 <= qe(1);
34   q2 <= qe(2);
35   q3 <= qe(3);
36   q4 <= qe(4);
37   q5 <= qe(5);
38
39 end Behavioral;

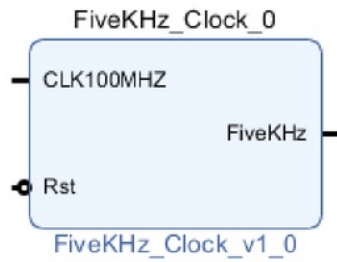
```



```

6 entity inverter_gate is
7   -- Port ( );
8   Port ( a : in std_logic;
9         q : out std_logic);
10 end inverter_gate;
11 architecture Behavioral of inverter_gate is
12
13 begin
14   q <= not a;
15
16 end Behavioral;

```



```

6 entity FiveKHz_Clock is
7
8     Port ( CLK100MHZ : in     STD_LOGIC;
9           Rst       : in     std_logic;
10          FiveKHz    : out    STD_LOGIC
11
12          );
13
14 end FiveKHz_Clock;
15
16 architecture Behavioral of FiveKHz_Clock is
17
18     signal cnt1      : integer := 0;
19
20     signal temp1     : std_logic := '0';
21
22
23     begin
24
25
26     process(CLK100MHZ,Rst)
27     begin
28         if Rst = '1' then
29             temp1 <= '0';
30             cnt1 <= 0;
31         elsif rising_edge (CLK100MHZ) then
32             if(cnt1= 50000000) then
33                 temp1 <= not temp1;
34                 cnt1 <= 0;
35             else
36                 cnt1 <= cnt1 +1 ;
37             end if;
38         end if;
39
40     end process;
41     FiveKHz <= temp1;
42
43 end Behavioral;

```

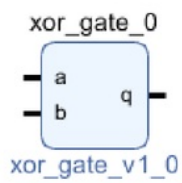
```

graph LR
    a --> or_gate_0
    b --> or_gate_0
    or_gate_0 --> q
    style or_gate_0 fill:#add8e6,stroke:#000,stroke-width:1px
    
```

```

6 entity or_gate is
7     -- Port ( );
8     Port ( a : in std_logic;
9           b : in std_logic;
10          q : out std_logic);
11 end or_gate;
12 architecture Behavioral of or_gate is
13
14     begin
15         q <= a or b;
16
17 end Behavioral;

```



```
6 entity xor_gate is
7   -- Port ( );
8   Port ( a : in std_logic;
9         b : in std_logic;
10        q : out std_logic);
11 end xor_gate;
12
13 architecture Behavioral of xor_gate is
14
15   begin
16     q <= a xor b;
17
18 end Behavioral;
```

The appendix includes all the VHDL component prototypes tested and used in the final design.

assignment 1

GRADEMARK REPORT

FINAL GRADE

GENERAL COMMENTS

Instructor

74/100

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

Text Comment. A little weak on objectives. The virtual ATE integration is possible here.

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13

PAGE 14

Text Comment. The research is comprehensive and has a good variety of ATE use in production industry.

PAGE 15

Text Comment. The theory is explained well and this is a good method of tying in the practical with theory. The point has been made here that the independent rules do not have max count practically.

Text Comment. Test shows the max count achieved by the 4 bit hybrid circuit.

Text Comment. This is the test cct for comparison with DUT.

Text Comment. Good test of maximum count for 6 bit hybrid circuit.

Text Comment. The test cct is correct and includes the all important x-or gate.

Text Comment. Good fault analysis and explanation of the 3 different scenarios.

Text Comment. The individual rules 90 and 150 show that the max count is not achieved using FPGA VHDL.

Text Comment. The 4 bit hybrid CAR shows that max count is achieved.

PAGE 31

Text Comment. Design is correct and shows the output from the external x-or gate feedback to the VHDL virtual design.

PAGE 32

Text Comment. The VHDL of the hybrid CAR is identified. The 6 bit conditions are shown here.

PAGE 33

Text Comment. Test shows that max count has been achieved by the 6 bit CAR.

PAGE 34

PAGE 35

Text Comment. 6 BIT car test of the practical prototype CUT built on prototype board.

Text Comment. The FPGA shows the fault has been registered, thus proving the ATE 6 bit CAR as an auto test procedure.

PAGE 36

Text Comment. A different fault is registered.

PAGE 37

PAGE 38

PAGE 39

Text Comment. PCB design is explained and looks correct.

PAGE 40

Text Comment. Test here showing no fault condition.

Text Comment. Good example of PCB CUT with fault conditions.

PAGE 41

Text Comment. Good use of table to show conditions of fault analysis.

PAGE 42

Text Comment. Good analysis here and comparison.

PAGE 43

Text Comment. This could be much better with the focus on the virtual ATE integration.

PAGE 44

PAGE 45

Text Comment. Refs are good and an appropriate number for this task.

PAGE 46

PAGE 47

PAGE 48

Text Comment. The appendix includes all the VHDL component prototypes tested and used in the final design.

GRADING FORM: CSG-RUBIC

JOSHUA DAVIES

0

RESEARCH-INFORMED LITERATURE



SMT technology is researched and includes good illustrations for support. 3/5 SMT and PTH are compared and evaluated. 3/5 ATE includes a variety of types used in production industry and are

14/20

explained in detail for this task. 8/10 Intro is a little weak on objectives. The focus should be more on virtual ATE integration.

KNOWLEDGE AND UNDERSTANDING



Design of 6 bit hybrid CAR with CAD covers all the prototype stages in a logical order which is easy to follow. This ties in with theory very well. 6.5/7
Design of PCB is good. The individual files necessary for manufacture are missing. 4.5/7
Design of 6 bit CAR using FPGA is explained very well and the prototypes are covered in appendix. 5/6

16/20

ANALYSIS



Test results are clear and explained. Good use of tables for analysis and comparison. The results tie in very well with theory. 8/10 Analysis of FPGA and PCB are covered in good detail. 7/10 Minor confusion with CAR and test circuit.

15/20

PRACTICAL APPLICATION AND DEP.



PCB manufacture process is good and the production a PCB is achieved. Good use of external software which required some research. 11/15
FPGA testing is logical using prototype method and in good order. 8/10 Good application and prototyping.

19/25

SKILLS FOR PROFESSIONAL PRAC.



Use of software is professional and external software for testing PCB design required some research. 3.5/5 Report layout is logical and easy to follow. 4/5 Conclusion is a little weak. 2.5/5

10/15