

1605160

by Steffan Davies

Submission date: 18-Jan-2022 07:35PM (UTC+0000)

Submission ID: 169516237

File name: 15323_Steffan_Davies_1605160_179500_16639916.docx (665.92K)

Word count: 2732

Character count: 11146

Course: HNC Electrical and Electronic Engineering

Module: Electronics II

Assignment: Finite state machine

Name: Steffan Wyn Davies

University number: 1605160

Date submitted: 18/01/2022

Word count: 2744

70-75

- ① SHORT RESEARCH, DIAGRAMS USED. CITATION LIST WAS APPROPRIATE.
- ② DESIGN METHOD WAS EXPLAINED IN DETAIL
- ③ TESTING WAS CARRIED OUT USING THE LOGIC ANALYSER VERY EFFECTIVELY.
- ④ SOME ANALYSIS WAS EVIDENT.
- ⑤ MODIFICATION NOT ATTEMPTED
- ⑥ GOOD CONCLUSIONS WITH CONSIDERATION OF SUSTAINABLE IMPLEMENTATION.

Contents

Research.....	3
Designing a traffic light control system	5
5 second timer	5
15 second timer	9
RAG Controller	14
Conclusion.....	19
Bibliography	20

Figure 1- 5 second timer state diagram	5
Figure 2- 5 second counter state table	5
Figure 3-5 second timer circuit	7
Figure 4-5 second timer logic analyser	8
Figure 5-15 second state diagram.....	9
Figure 6-15second timer truth table.....	10
Figure 7-0-15 counter final circuit	12
Figure 8-0-15 counter logic analyser	13
Figure 9-15s timer logic analyser	Error! Bookmark not defined.
Figure 10-RAG controller truth table	15
Figure 12-RAG controller final circuit.....	17
Figure 11-RAG controller logic analyser.....	17
Figure 13-Traffic light final circuit	18
Figure 14-Final circuit logic analyzer	18

Executive summary

The purpose of this task is to design and test a traffic light control system. Research was carried out into the various design methods and development options with regards to cost drivers and sustainability. A full analysis of the system was carried out and found that the traffic light system worked as intended. However, a pressure pad could be introduced as an improvement to the system in order to improve road safety. A key point gathered from this task was that FPGA's are suitable for prototyping and ASICs are suitable for applications where mass production is required. *SHOULD COVER THE WHOLE REPORT.*

Research

Before beginning the design process, research was required to understand the types of pedestrian crossings available in the UK and how they work. The 2 most common pedestrian crossings can be explained below:

1. **Pelican crossing-** pelican crossings use lights, sound and buttons to allow pedestrians to cross safely. When the pedestrian presses the wait button, traffic is then brought to a stop. Timers and logic within the control panel are then used to ensure pedestrians can cross safely before the traffic lights are changed back to green and traffic can flow again. The green walking person symbol lights will change to red which is determined by a timer, which according to legislation is between 9 and 16 seconds long depending on the size of the crossing. (office, 1995)
2. **Puffin crossing-** puffin crossings work in a very similar way to the pelican crossing. However, puffin crossings also use sensors called PCD (pedestrian crossing detector) and PKD (pedestrian kerb detector). This means that the time the traffic is on stop for can be increased if a pedestrian takes a long time to cross.

There are several new technologies that can be used for the control circuitry. One example is the introduction of smart traffic lights. The technology combines wide-angle cameras with artificial intelligence in order to detect the number of pedestrians that require to cross the road. When a certain amount of people are waiting, the traffic lights will then change to red, allowing the pedestrians to cross. This therefore reduces the frequency that drivers are made to stop. The AI technology is used to predict if pedestrians have intentions of crossing the road from up to 15M away. (Ahuja, 2021)

The control circuitry of a traffic light system can be controlled using an FPGA. An FPGA is a field programmable gate array which is a hardware circuit that can be programmed to carry out logic operations (Touger, 2018). The FPGA works by loading a configuration into the FPGA and then it will begin to act like the circuit you wanted (C, 2022). FPGA's are relatively

simplistic compared to ASIC's, but are allot more flexible as they can be redesigned by changing the software. Another advantage of FPGA's is that its development costs are low in comparison to ASIC. This makes FPGA's suitable for prototyping as you can download, test and re-download until the design is correct.

ASIC's is an application specific integrated circuit, meaning it is designed for a specific application. ASIC's are commonly used for high volume products due to the cost of the design process. Although ASIC's have a complex design flow, their performance is better than FPGA's and the production unit cost is smaller.

When designing a system such as a traffic light system, careful consideration must be taken to promote sustainable development. One example of this is the transition from halogen to LED lightbulbs within traffic lights. This was done around 1981 in order to improve efficiency. By improving the efficiency of the traffic light system, the battery on mobile units will last longer. Furthermore, if one LED fails on a traffic light there are still hundreds of other ones that are still lit. Not only do LED's reliability contribute to sustainable development, but it also improves road safety.

BASIC RESEARCH, WOULD BENEFIT FROM DIAGRAMS

Designing a traffic light control system

5 second timer

The first part of designing the traffic light control system required finding out the specification for the 5 second timer. The 5 second timer was required to count from 0-5, start when B=1 and W to go high when the count is 5. This was then put into the state diagram seen below:

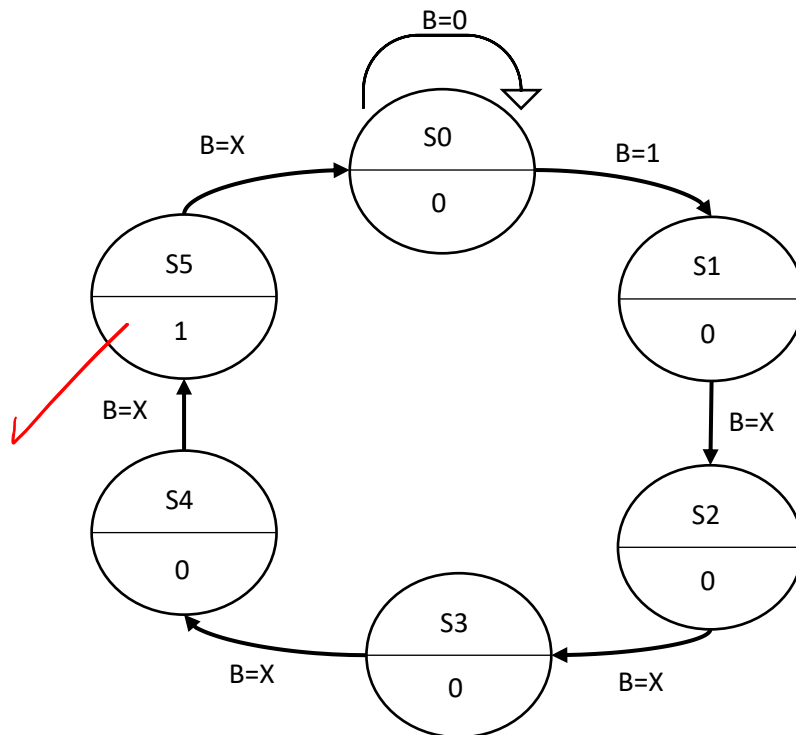


Figure 1- 5 second timer state diagram

After creating the state diagram, the next part in the design process was to create a state table, as seen below:

Current state	Next state		Current			Next(B=0)			Next(B=1)			Output (W)
	B=0	B=1	QA	QB	QC	QA+	QB+	QC+	QA+	QB+	QC+	
S0	S0	S1	0	0	0	0	0	0	0	0	1	0
S1	S1	S2	0	0	1	0	1	0	0	1	0	0
S2	S2	S3	0	1	0	0	1	1	0	1	1	0
S3	S3	S4	0	1	1	1	0	0	1	0	0	0
S4	S4	S5	1	0	0	1	0	1	1	0	1	0
S5	S0	S0	1	0	1	0	0	0	0	0	0	1
S6	X	X	1	1	0	X	X	X	X	X	X	X
S7	X	X	1	1	1	X	X	X	X	X	X	X

Figure 2- 5 second counter state table

Karnaugh mapping was then required to reduce the Boolean expressions. The Karnaugh mapping process and reduced Boolean expressions can be seen below:

BQA\QBQC	00	01	11	10
00	0	0	1	0
01	1	0	X(1)	X(1)
11	1	0	X(1)	X(1)
10	0	0	1	0

$$QA^+ = QA\overline{Q}\overline{C} + QBQC$$

BQA\QBQC	00	01	11	10
00	0	1	0	1
01	0	0	X(0)	X(1)
11	0	0	X(0)	X(1)
10	0	1	0	1

$$QB^+ = \overline{QA}\overline{Q}BQC + QB\overline{Q}\overline{C}$$

BQA\QBQC	00	01	11	10
00	0	0	0	1
01	1	0	X(0)	X(1)
11	1	0	X(0)	X(1)
10	1	0	0	1

$$QB^+ = \overline{QA}Q\overline{C} + QB\overline{Q}\overline{C} + B\overline{Q}\overline{C}$$

QA\QBQC	00	01	11	10
0	0	0	0	0
1	0	1	X(1)	X(0)

$$W = QAQC$$



The next step was to build the circuit using multisim, the 0-5 counter circuit can be seen below:

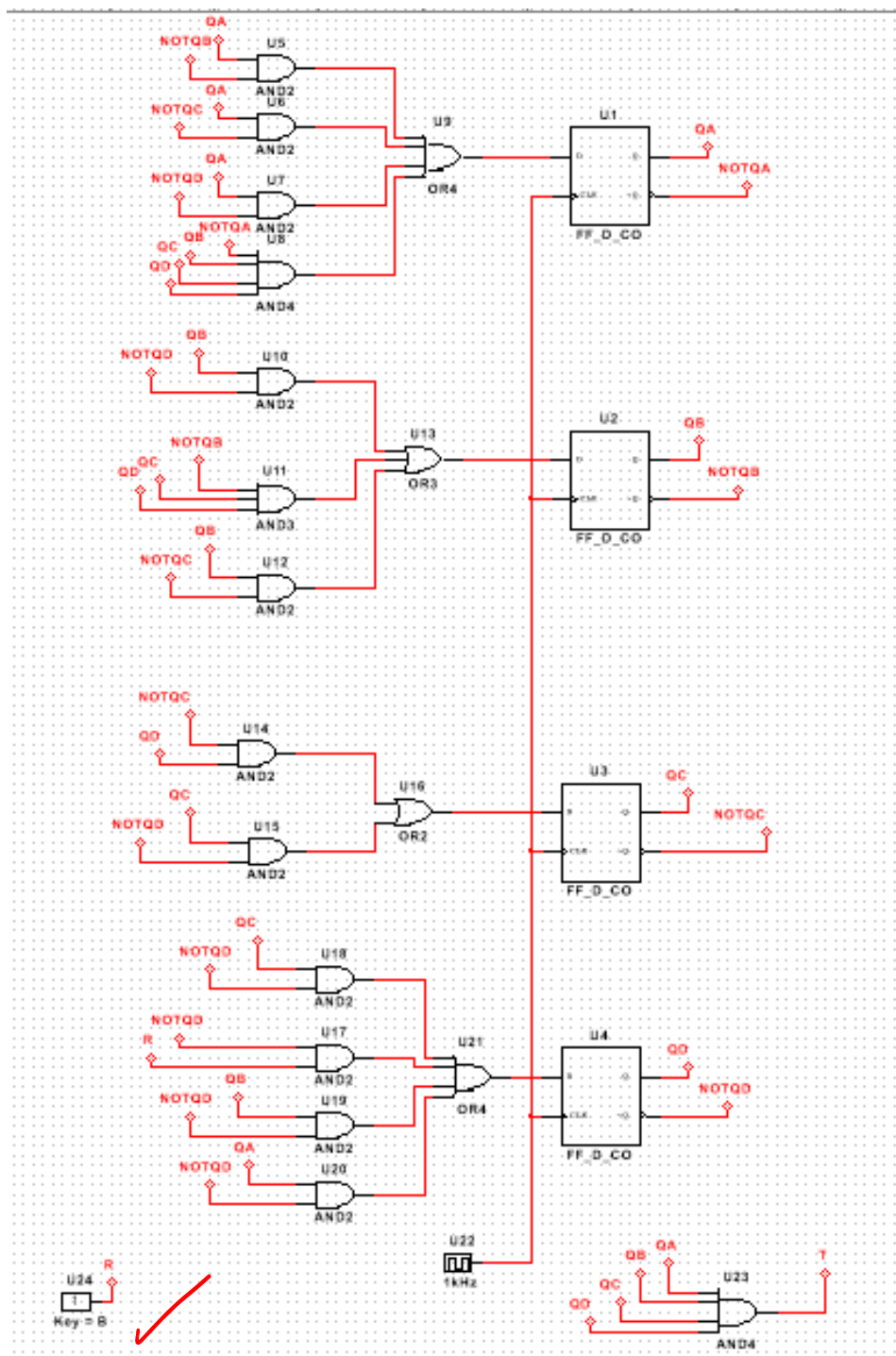


Figure 3-5 second timer circuit

In order to test the 5 second timer, a logic analyser was used to analyse when each output would become high. By looking at the logic analyser and the truth table, we can see that the correct outputs come on at the correct time. For example, the output 'W' is high when QA and QC are high, and QB is low.

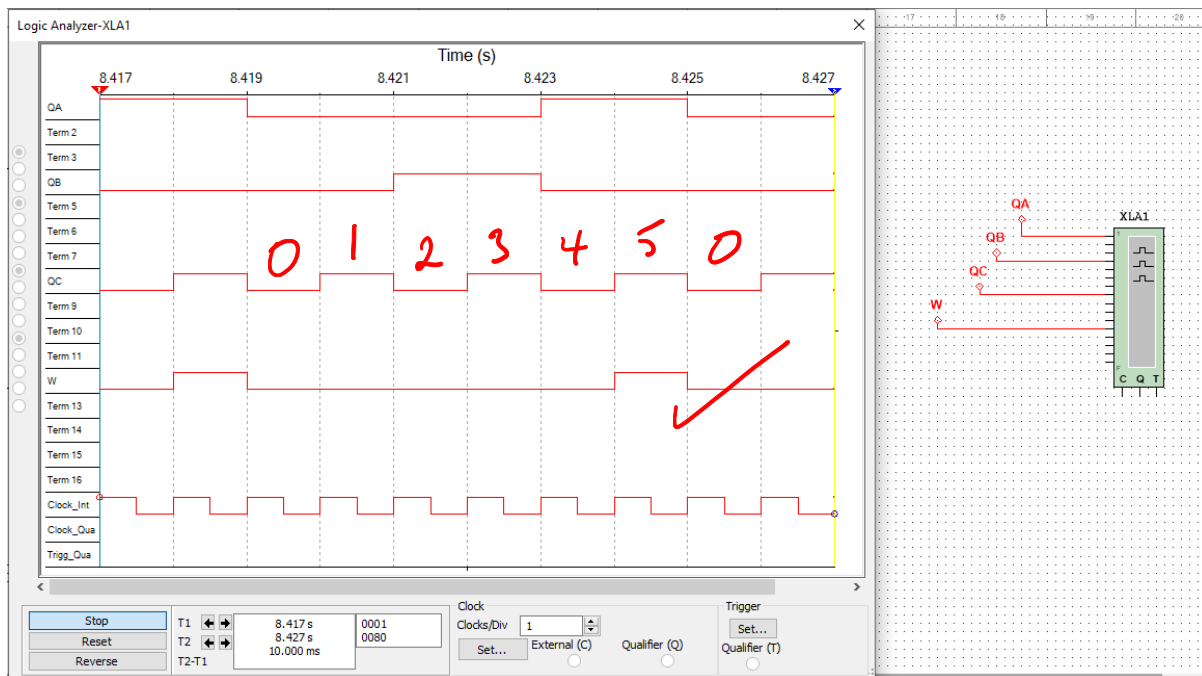


Figure 4-5 second timer logic analyser

15 second timer

The first part of designing the 15 second timer was to determine the specification of the circuit. The circuit was required to count from 0-15 when R=1 and when 15 seconds has passed, output T will become 1. The state diagram for this can be seen below:

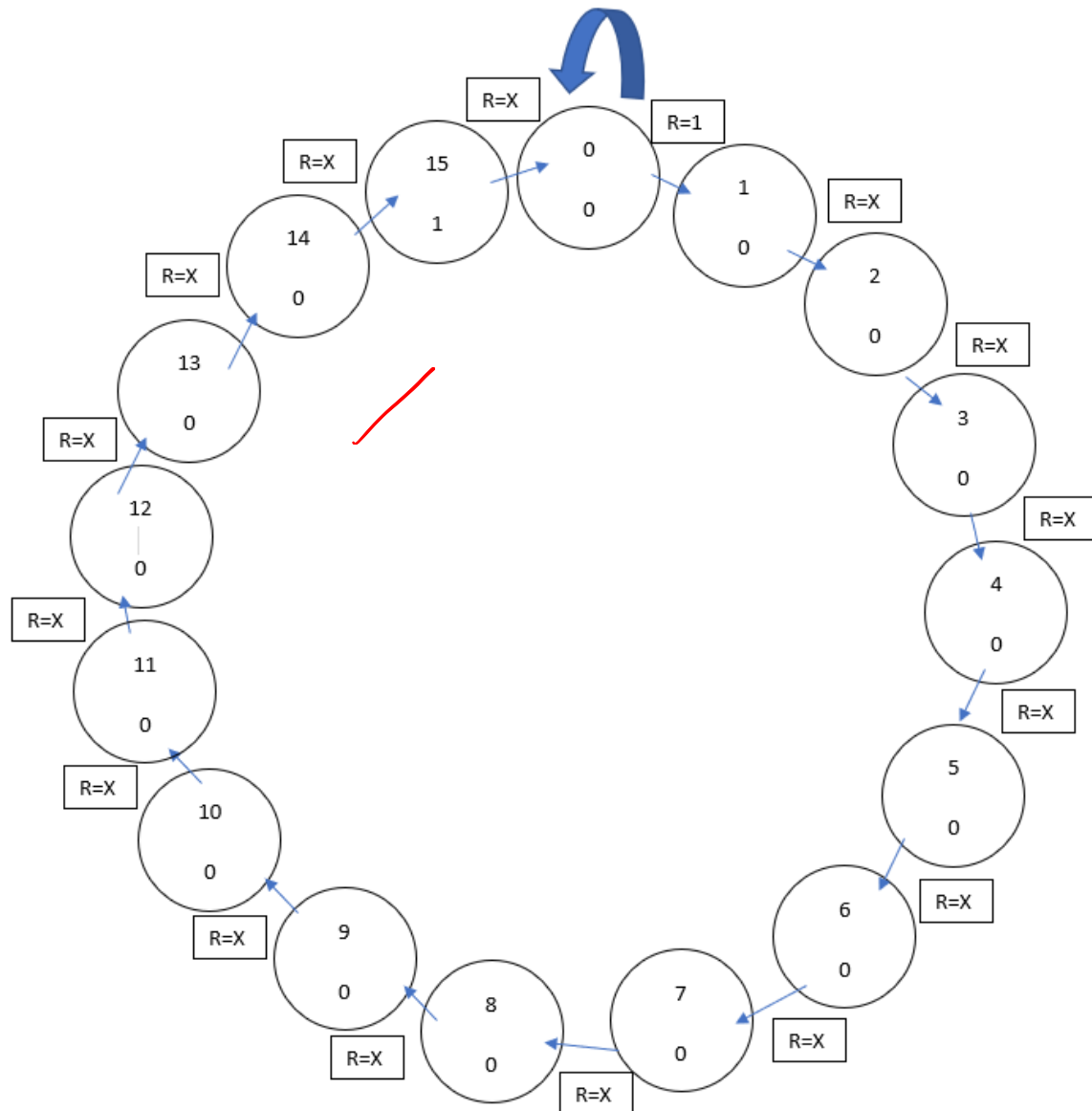


Figure 5-15 second state diagram

Current state	Next state		Current state				R=0				R=1				Output (T)
	R=0	R=1	QA	QB	QC	QD	QA+	QB+	QC+	QD+	QA+	QB+	QC+	QD+	
S0	S0	S1	0	0	0	0	0	0	0	0	0	0	0	1	0
S1	S2	S2	0	0	0	1	0	0	1	0	0	0	1	0	0
S2	S3	S3	0	0	1	0	0	0	1	1	0	0	1	1	0
S3	S4	S4	0	0	1	1	0	1	0	0	0	1	0	0	0
S4	S5	S5	0	1	0	0	0	1	0	1	0	1	0	1	0
S5	S6	S6	0	1	0	1	0	1	1	0	0	1	1	0	0
S6	S7	S7	0	1	1	0	0	1	1	1	0	1	1	1	0
S7	S8	S8	0	1	1	1	1	0	0	0	1	0	0	0	0
S8	S9	S9	1	0	0	0	1	0	0	1	1	0	0	1	0
S9	S10	S10	1	0	0	1	1	0	1	0	1	0	1	0	0
S10	S11	S11	1	0	1	0	1	0	1	1	1	0	1	1	0
S11	S12	S12	1	0	1	1	1	1	0	0	1	1	0	0	0
S12	S13	S13	1	1	0	0	1	1	0	1	1	1	0	1	0
S13	S 14	S14	1	1	0	1	1	1	1	0	1	1	1	0	0
S14	S15	S15	1	1	1	0	1	1	1	1	1	1	1	1	0
S15	S0	S0	1	1	1	1	0	0	0	0	0	0	0	0	1

Figure 6-15second timer truth table

Karnaugh mapping was then required to reduce the Boolean expressions. The Karnaugh mapping and reduced Boolean equations can be seen below:

QA	00	01	11	10
00	0	0	0	0
01	0	0	1	0
11	1	1	0	1
10	1	1	1	1

QA	00	01	11	10
00	0	0	0	0
01	0	0	1	0
11	1	1	0	1
10	1	1	1	1

$$QA^+ = QA\overline{Q}\overline{D} + QA\overline{Q}\overline{C} + QA\overline{Q}\overline{B} + \overline{Q}AQBQCQD$$

QB	00	01	11	10
00	0	0	1	0
01	1	1	0	1
11	1	1	0	1
10	0	0	1	0

QB	00	01	11	10
00	0	0	1	0
01	1	1	0	1
11	1	1	0	1
10	0	0	1	0

$$QB^+ = QB\overline{Q}\overline{D} + QB\overline{Q}\overline{C} + \overline{Q}BQCQD$$

QC	00	01	11	10
00	0	1	0	1
01	0	1	0	1
11	0	1	0	1
10	0	1	0	1

QC	00	01	11	10
00	0	1	0	1
01	0	1	0	1
11	0	1	0	1
10	0	1	0	1

$$QC^+ = QC\overline{Q}\overline{D} + \overline{Q}CQD$$

QD	00	01	11	10
00	0	0	0	1
01	1	0	0	1
11	1	0	0	1
10	1	0	0	1

QD	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	1	0	0	1
10	1	0	0	1

$$QD^+ = QC\overline{Q}\overline{D} + QB\overline{Q}\overline{D} + QA\overline{Q}\overline{D} + \overline{Q}DQR$$

The next step was to build the circuit using multisim. The final circuit for the 0-15 second counter can be seen below:

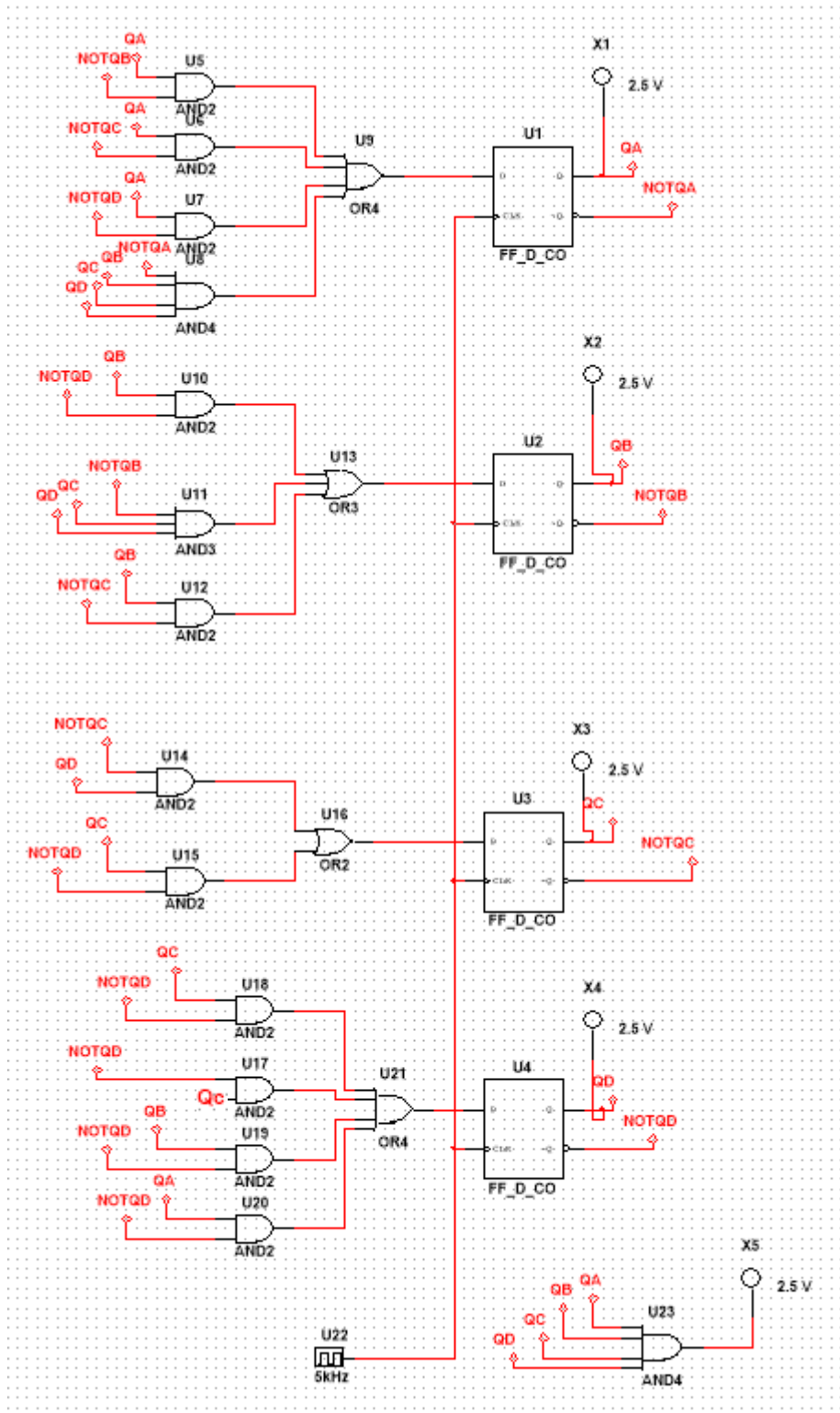


Figure 7-0-15 counter final circuit

After building the circuit a logic analyser was used to test it. A screenshot of the logic analyser can be seen below:

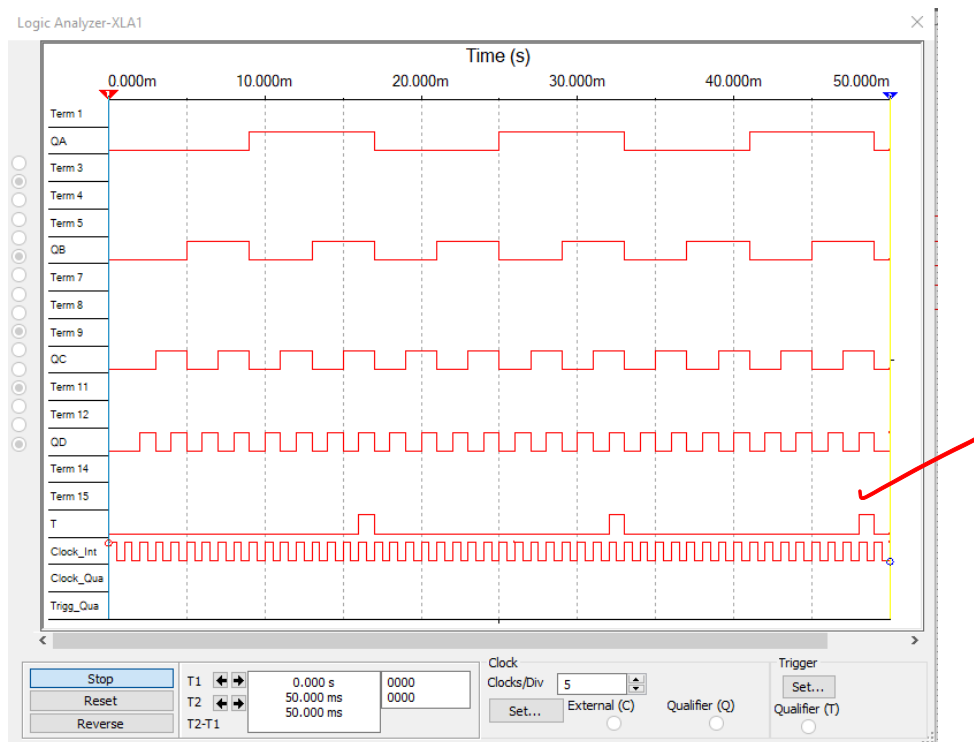


Figure 8-0-15 counter logic analyser

The logic analyser shows that the correct inputs come on at the correct time. For example, T shows a logic 1 when QA, QB, QC and QD are high meaning the count has reached 15 seconds and output T will become a logic 1.

RAG Controller

The first step of designing the RAG controller, I had to determine the design specification and state allocation. The research carried out helped to determine the state allocation seen below:

Count 0 = Green

Count 1 = Amber

Count 2 = Amber

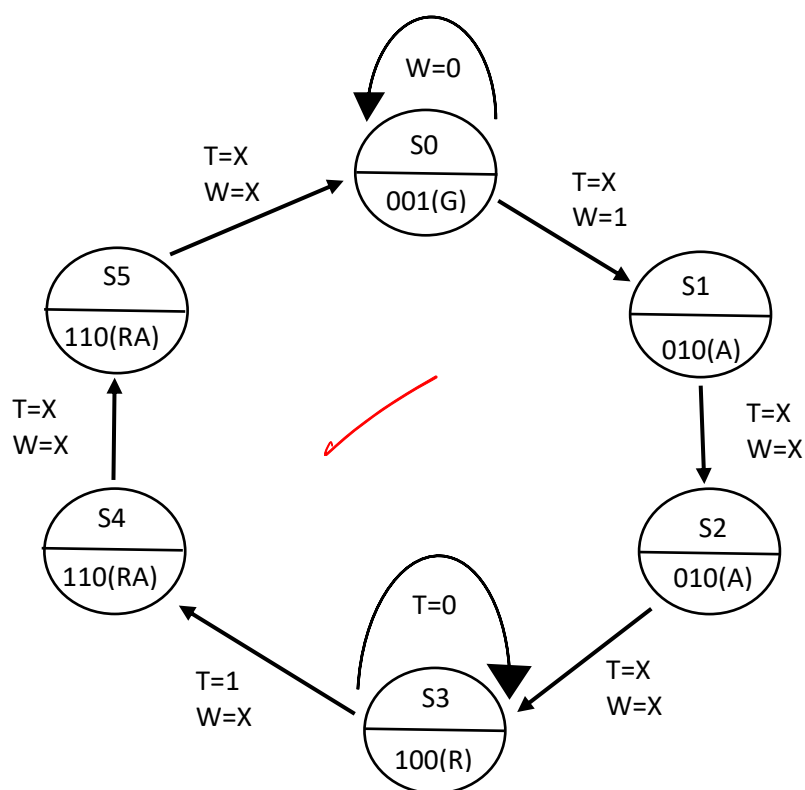
Count 3 = Red

Count 4 = Amber & Red

Count 5 = Amber & Red

Count 6 = Green

The state diagram for the RAG controller can be seen below:



The truth table for the RAG controller can be seen below:

Present state	Next state				Next state												Ouputs		
	W=0		W=1		W=0						W=1								
					T=0			T=1			T=0			T=1					
	T=0	T=1	T=0	T=1	QA+	QB+	QC+	QA+	QB+	QC+	QA+	QB+	QC+	AQ+	QB+	QC+	R	A	G
S0	S0	S0	S1	S1	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1
S1	S2	S2	S2	S2	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0
S2	S3	S3	S3	S3	0	1	1	0	1	1	0	1	1	0	1	1	0	1	0
S3	S3	S3	S3	S3	0	1	1	1	0	0	0	1	1	1	0	0	1	0	0
S4	S5	S5	S5	S5	1	0	1	1	0	1	1	0	1	1	0	1	1	1	0
S5	S0	S0	S0	S0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
S6	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
S7	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Figure 9-RAG controller truth table

The Karnaugh maps used to reduce Boolean equations can be seen below:

QA	00	01	11	10
00	0	0	0	0
01	1	0	X	X(1)
11	1	0	X(1)	X(1)
10	0	0	1	0

QA	00	01	11	10
00	0	0	0	0
01	1	0	X	X(1)
11	1	0	X(1)	X(1)
10	0	0	1	0

$$QA^+ = QA\overline{Q}C + TQBQC$$

QB	00	01	11	10
00	0	1	1	1
01	0	0	X	X(1)
11	0	0	X	X(1)
10	0	1	0	1

QB	00	01	11	10
00	0	1	1	1
01	0	0	X	X(1)
11	0	0	X	X(1)
10	0	1	0	1

$$QB^+ = QB\overline{Q}C + \overline{T}QB + \overline{QA}QBQC$$

QC	00	01	11	10
00	0	0	1	1
01	1	0	X(1)	X(1)
11	1	0	X	X(1)
10	0	0	0	1

QC	00	01	11	10
00	1	0	1	1
01	1	0	X(1)	X(1)
11	1	0	X	X(1)
10	1	0	0	1

$$QC^+ = QB\overline{Q}C + \overline{T}QB + QA\overline{Q}C + \overline{Q}CW$$

QA\QBQC	00	01	11	10
0	0	0	1	0
1	1	1	X(1)	X(1)

QA\QBQC	00	01	11	10
0	0	1	0	1
1	1	1	X(1)	X(1)

$$R = QA + QBQC$$

$$A = QA + \overline{Q}BQC + QB\overline{Q}C$$

QA\QBQC	00	01	11	10
0	1	0	0	0
1	0	0	X	X

$$G = \overline{QA}QBQC$$

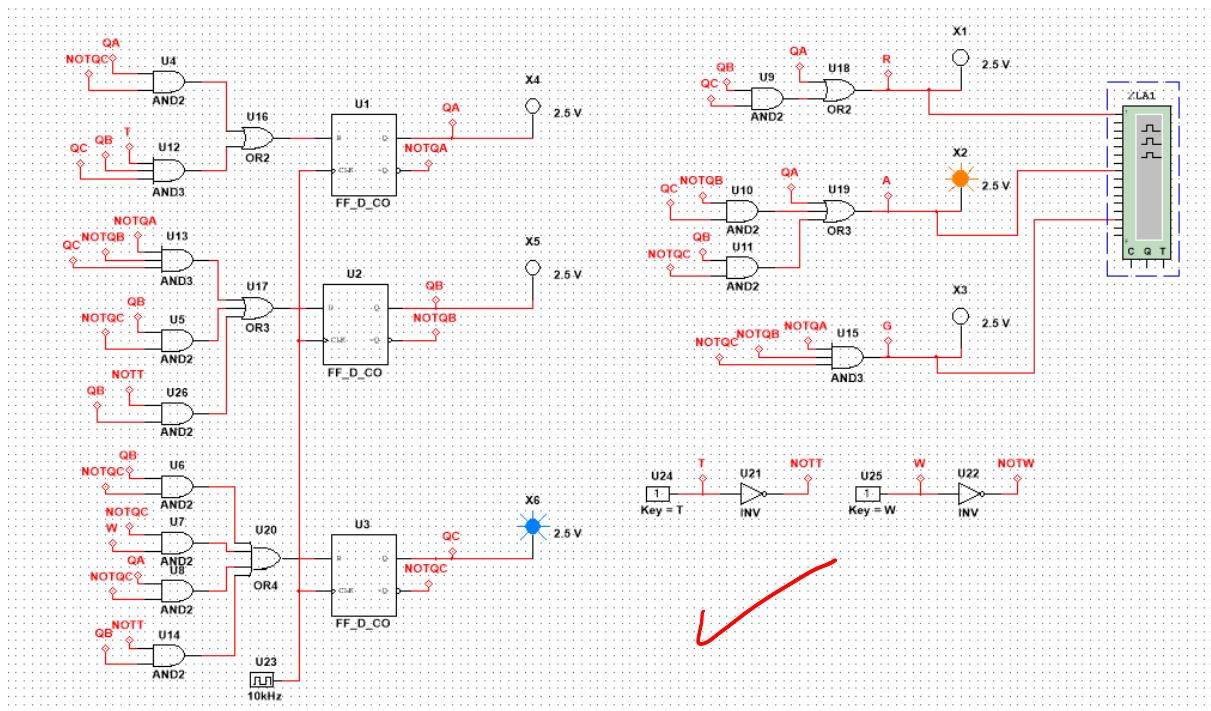


Figure 10-RAG controller final circuit

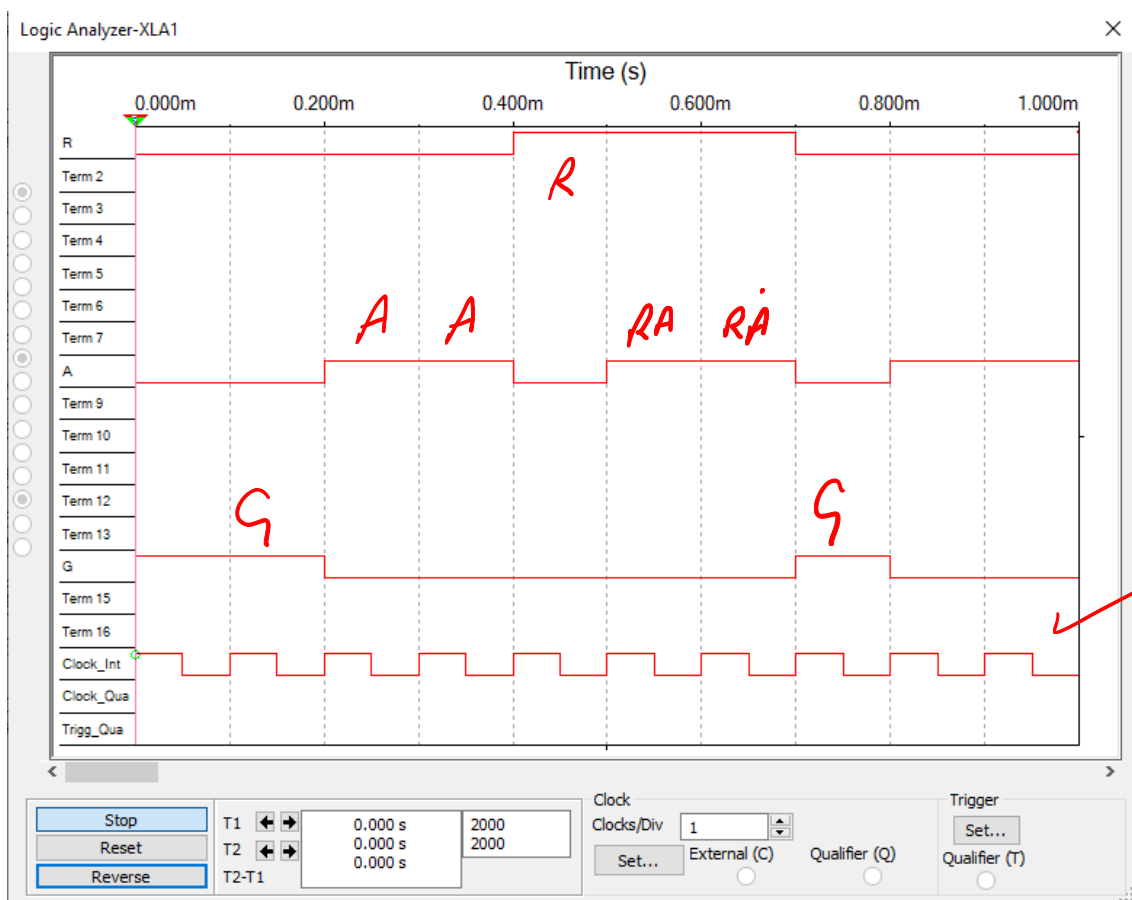


Figure 11-RAG controller logic analyser

The final circuit for the traffic light system can be seen below:

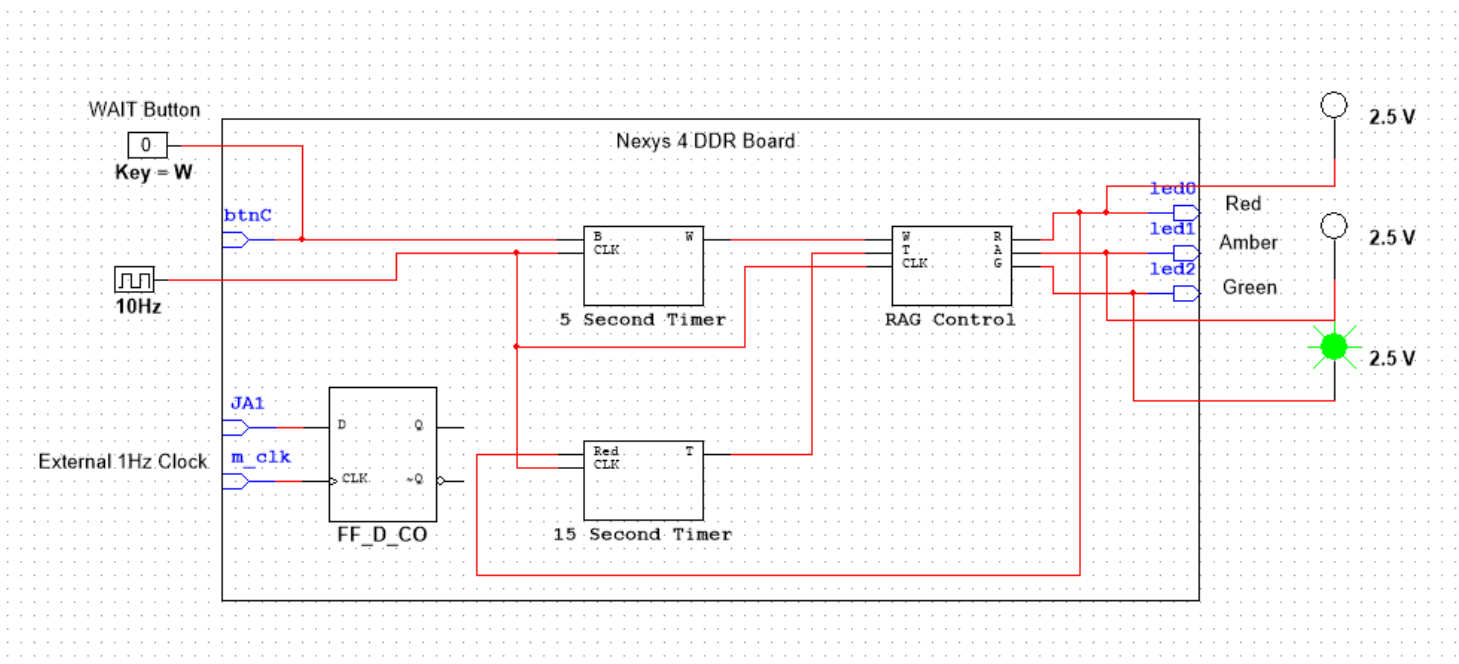


Figure 12-Traffic light final circuit

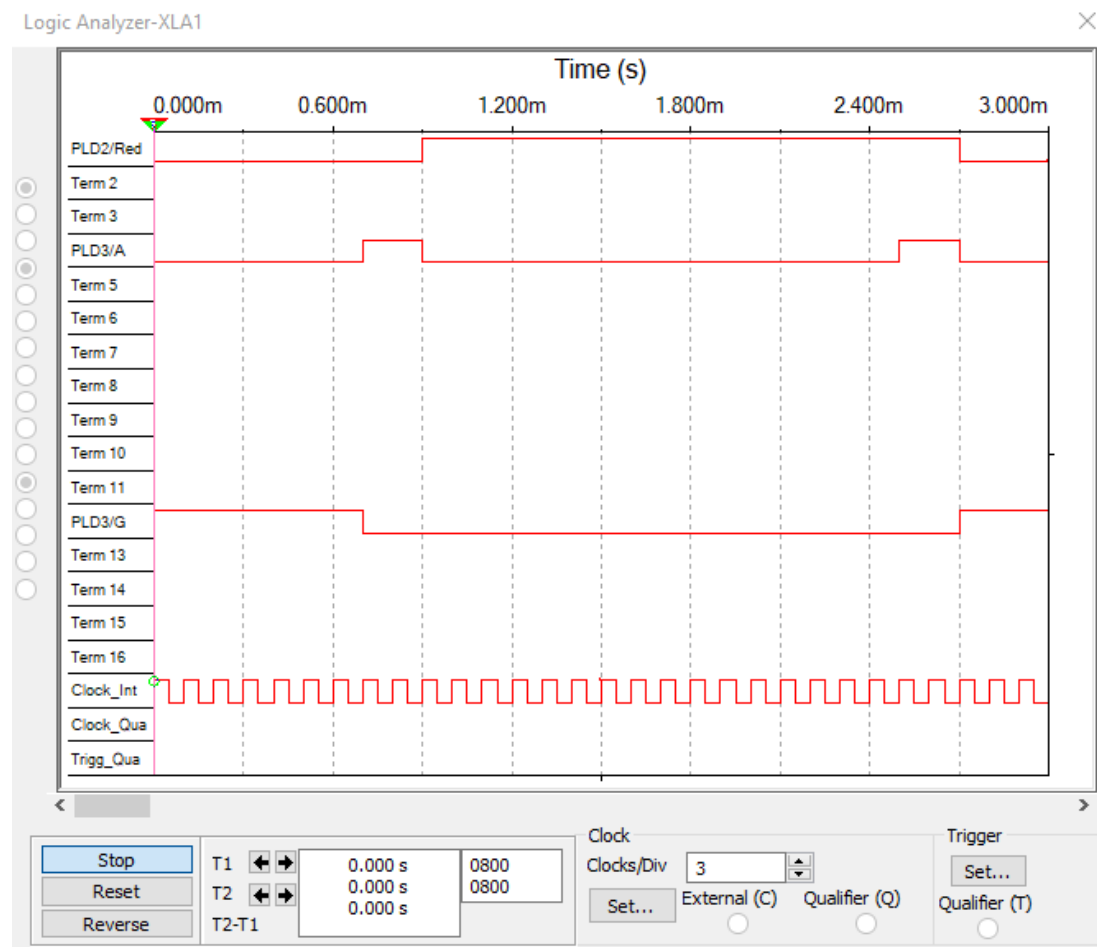


Figure 13-Final circuit logic analyzer

Conclusion

The final circuit worked as it should. All circuits were tested visually using LED's before using a logic analyser to check the output of each circuit was coming on when they should. By testing each circuit individually before being placed in the final circuit, it prevented any problems from being encountered at the final stages of the design. The same design process was used for all circuits, which included state allocation, state diagram, state table, Karnaugh mapping, multisim circuit.

Although the traffic light system worked as intended, there are a few improvements that could be introduced to the system. One example of this would be to add a pressure pad as an additional input to the final circuit. This would mean that if someone was to press the wait button and walk away, then the lights wouldn't change colour unless someone is stood on the pressure pad. Another possible solution would be to have pedestrian crossing detectors on the other side of the road so that pedestrians can take as long as they need to cross the road without the problem of the lights changing colour.

This design would be best implemented using an FPGA due to FPGA's flexibility. If any modifications would be needed with this circuit, then it could be changed easily with FPGA's. However, with respect to sustainable development, ASIC would be better suited due to its better efficiency and performance compared to FPGA's. Also, FPGA's can be cloned so it's important that steps are put in place to protect intellectual property. When it comes to cost, the non-recurring engineering cost (NRE) of ASIC is high compared to FPGA, but FPGA's can cost more overall since individual costs are higher per unit than ASIC. (Cadence PCB solutions, 2021)

In conclusion, how the circuit should be implemented is determined by what the system needs to do. For example, a FPGA is suitable for prototyping applications as its design flow allows it to be reprogrammable and able to reach the market faster. However, ASIC would be suitable where sustainable development is considered due to its performance and efficiency. Furthermore, the total cost of an ASIC would be decreased the more quantity you required, meaning this system would need to be mass produced in order to compete financially with FPGA's.

If this system was to be designed and implemented as an ASIC, then careful consideration must be taken of the non-recurring engineering (NRE) which covers the cost of the design and specification. Another cost driver to consider is the production costs which include testing wafers, cutting them into individual dice and packaging as required. The 3 main factors that determine the price of an ASIC are testing time, die area and packaging, which as previously mentioned is dependant on volume. (Sigenics inc, 2021)

Bibliography

Ahuja, K., 2021. *The sunday times*. [Online]

Available at: <https://www.driving.co.uk/news/roads/smart-traffic-lights-fewer-stops-motorists/>
[Accessed 09 January 2022].

Cadence PCB solutions, 2021. *Cadence*. [Online]

Available at: <https://resources.pcb.cadence.com/blog/2019-fpga-vs-asic-differences-and-choosing-best-for-your-business#:>
[Accessed 18th January 2022].

C, E., 2022. *sparkfun*. [Online]

Available at: <https://www.sparkfun.com/users/871489?>
[Accessed 16th January 2022].

Hafizudding, 2021. *Microcontrollerslab.com*. [Online]

Available at: <https://microcontrollerslab.com/density-based-auto-traffic-signal-control/>
[Accessed 09 January 2022].

office, T. s., 1995. *The design of pedestrian crossings*. [Online]

Available at:
https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/330214/ltn-2-95_pedestrian-crossings.pdf
[Accessed 04 January 2022].

Sigenics inc, 2021. *Sigenics*. [Online]

Available at: <https://www.sigenics.com/page/asics-c#:~>
[Accessed 18th January 2022].

Touger, E., 2018. *PROWESS*. [Online]

Available at: <https://www.prowesscorp.com/what-is-fpga/>
[Accessed 16th January 2022].

FINAL GRADE

GENERAL COMMENTS

Instructor

68/100

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

GRADING FORM: CSG-RUBIC

STEFFAN DAVIES**68****RESEARCH-INFORMED LITERATURE**

There was evidence of further reading and research and some in text referencing. Your references were mainly appropriate to the task, however the reference list was short and you should increase the breadth of your research for future assignments.

5**KNOWLEDGE AND UNDERSTANDING**

The knowledge and understanding of the subject demonstrated by this assignment was reasonably extensive. There were very few gaps in your knowledge of the basic principles and your understanding of the higher level concepts was demonstrated to a limited degree. You have demonstrated some critical understanding of the main concepts and expanded your knowledge beyond that contained in the course notes, however your explanations need to be more

22

detailed and written with a greater emphasis on critical evaluation rather than description.

ANALYSIS



Your analysis of the task was generally very good, both in class and as shown in this assignment. You have demonstrated sound critical analysis skills and your approach to the task was logical. You carried out a detailed evaluation of the results and included a good conclusion relating the practical results to researched theory. The conclusions demonstrated very good cognitive skills and the ability to interpret quantitative and qualitative data without error.

18

PRACTICAL APPLICATION AND DEP.



Your application of practical skills for this task was generally excellent. You used both the basic and advanced features of the resources appropriately when designing the circuit and you used appropriate advanced options for testing and evaluating the circuit performance. You managed to design an effective circuit that functioned well within the specification and achieved this by rarely asking for assistance thus demonstrating some advanced problem solving skills. Your results were comprehensive and allowed for a detailed conclusion to be written.

16

SKILLS FOR PROFESSIONAL PRAC.



In this assignment you have demonstrated skills in working individually and also collaboratively. Your written report was generally good and contained most of the required detail, it was also written in a coherent and logical way. The available word count was in the main effectively used.

7