

PCEE4006

Electronics

Digital Systems

Billy Heaven

Task 1 - Boolean Algebra and De Morgan's Theorem

i, Boolean Expression = $A.B.C + A.B'.C + A.B.C'$

To simplify the expression, a series of laws are followed;

$A.C.(B+B') + A.B.C'$	Distributive Law
$A.C.1 + A.B.C'$	Complement Law
$A.C + A.B.C'$	Identity Law
$A.(C+B.C')$	Distributive Law
$A.(B) + C$	Absorption Law
$A.B + A.C$	Distribution Law

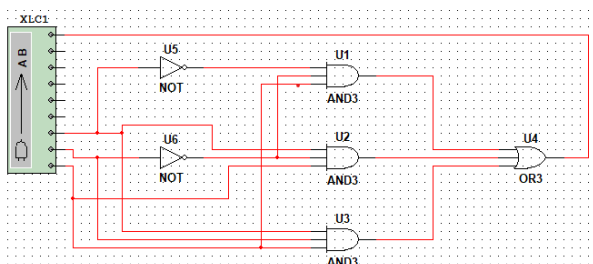
The expression has now been simplified. For the expression to be converted using only NAND gates, De Morgan's theorem must be used;

$$Q = A.B + A.C$$

$$Q'' = (A.B + A.C)''$$

$$Q = ((A.B)' . (A.C)')'$$

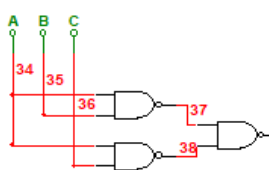
To prove that the converted expression is correct, both the original expression (Figure A) and new expression (Figure B) can be built and simulated. An identical truth table should be produced despite only NAND gates being used for the newly converted expression.



000	0	0	0	0
001	0	0	1	0
010	0	1	0	0
011	0	1	1	0
100	1	0	0	0
101	1	0	1	0
110	1	1	0	1
111	1	1	1	1

(Figure A – Simulated Original Boolean Expression - $A.B.C + A.B'.C + A.B.C'$)

Commented [RB1]: Well done.



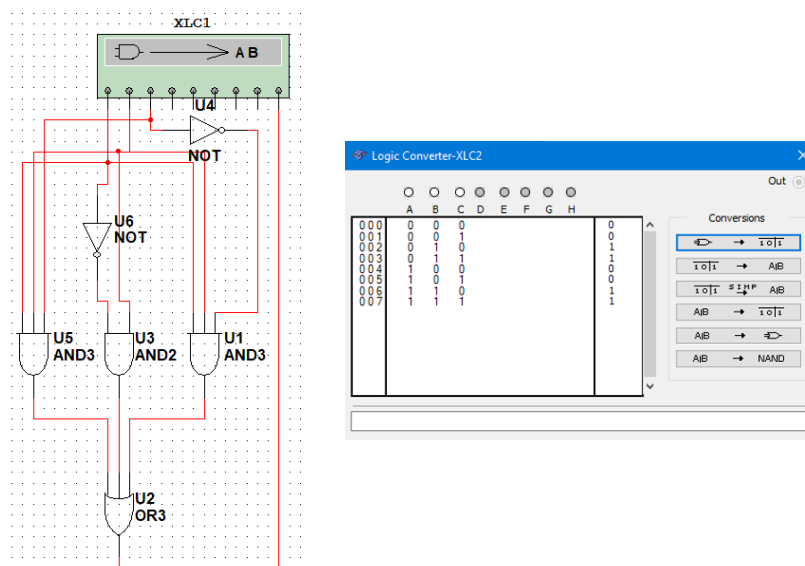
000	0	0	0	0
001	0	0	1	0
010	0	1	0	0
011	0	1	1	0
100	1	0	0	0
101	1	0	1	0
110	1	1	0	1
111	1	1	1	1

(Figure B – Simulated Simplified Boolean Expression - $((A.B)'.(A.C)')'$)

ii, Boolean expression = $A.B.C + A'B + A.B.C'$

$Q = A.B.(C.C') + A'B$	Distributive Law
$Q = A.B.1 + A'.B$	Complement Law
$Q = A.B + A'.B$	Identity Law
$Q = B.(A + A')$	Distributive Law
$Q = B.1$	Complement Law
$Q = B$	Identity Law

Due to the expression being simplified to $Q = B$, this means that the produced output equals 1 whenever B equals 1. Resulting in no need for the use of De Morgan's theorem and no need for it to be converted into a NAND gate configuration. The expression $Q = B$ would result in the logic circuit being a straight connection from B to Q.



(Figure C – Simulated Original Boolean Expression - $A.B.C+A'B+A.B.C'$)

iii) Boolean Expression = $Q = (A+B)(A+B')(A'+C)$

$(A.(B+B')).(A'+C)$	Distributive Law
$A.1.(A'+C)$	Complement Law
$A.(A'+C)$	Identity Law
$A.A' + A.C$	Distributive Law
$0+AC$	Complement Law
A.C	Identity Law

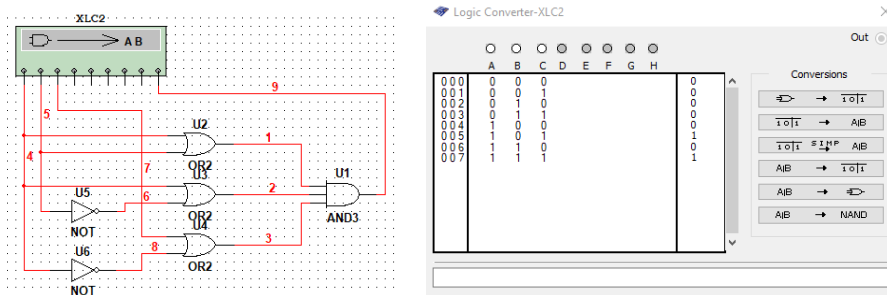
Commented [RB2]: Not right?

Billy Heaven - 166855

The expression has now been simplified. For the expression to be converted using only NAND gates, De Morgan's theorem must be used;

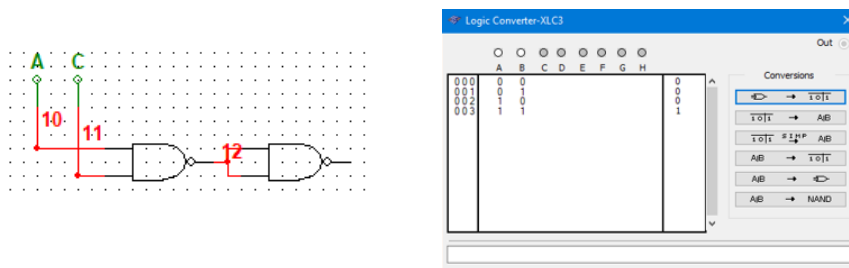
$$Q = A.C$$
$$Q = (A.C)''$$

Shown below in figure D is the original Boolean expression and associated logic table for $Q=(A+B)(A+B')(A'+C)$.



(Figure D – Simulated Original Boolean Expression - $(A+B)(A+B')(A'+C)$)

Shown below in figure, is the logic circuit and truth table for the simplified expression that was produced through the use of different laws and De Morgan's First Law = $(A.C)''$. The logic tables in both figure D and figure E show the same outcomes, proving that the simplified expression of $(A.C)''$ is correct.



(Figure E – Simulated Simplified Boolean Expression - $(A.C)''$)



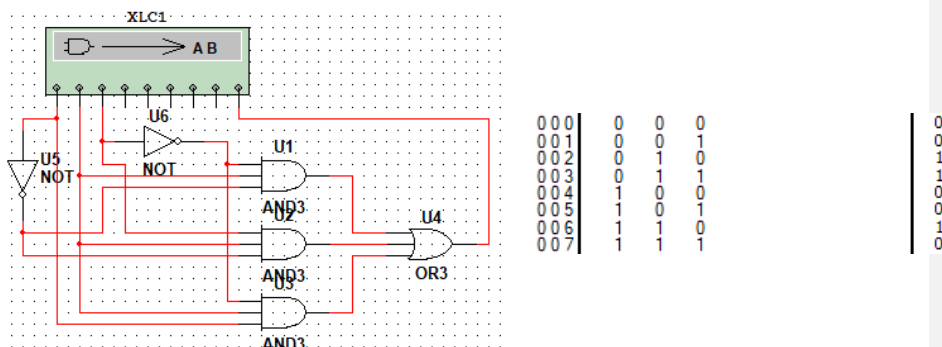
Task 2 – Using Combinational logic to solve real world problems

Truth table

A	B	C	Q
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

$A'BC'$ ✓
 $A'BC$ ✓
 ABC' ✓

- I. From the truth table above, the Boolean expression of $A'BC' + A'BC + ABC'$ can be formed. In the figure below, a logic circuit has been simulated representing the Boolean expression from the truth table. The simulated logic circuit produced a truth table that matches the original truth table shown above, proving that the logic circuit is correct.



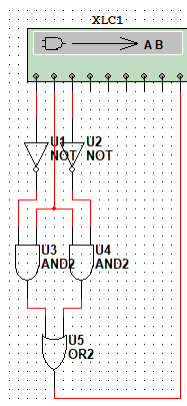
(Figure E – Simulated Boolean Expression - $A'BC' + A'BC + ABC'$)

Boolean expression $Q = A'BC' + A'BC + ABC'$

$A'B.(C+C') + ABC'$	Distributive Law
$A'B1 + ABC'$	Complement Law
$A'B + ABC'$	Identity Law
$B.(A'+AC')$	Distributive Law
$B.(A'+C')$	Absorption Law
$A'B + BC'$	Distribution Law

The original Boolean expression can be simplified to an expression of $A'B + BC'$. As shown below, a logic circuit has been produced and simulated. The truth table produced from this simulation

matches the truth table produced from the original Boolean expression proving that the simplified expression is correct.



(Figure F – Simulated Simplified Boolean Expression - $A'B + BC'$)

Logic Converter-XLC1

	A	B	C	D	E	F	G	H	
000	0	0	0						0
001	0	0	1						1
002	0	1	0						1
003	0	1	1						0
004	1	0	0						0
005	1	0	1						1
006	1	1	0						0
007	1	1	1						1

Conversions

Out

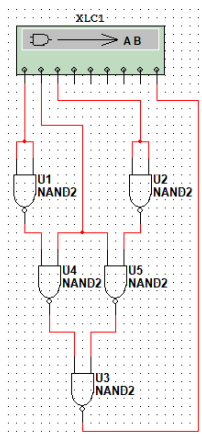
The use of De Morgan laws can be used to convert the simplified expression of $A'B + BC'$ to an expression that can be shown as a logic circuit through the use of only NAND gates;

$$Q = A'B + BC'$$

$$Q'' = (A'B + BC')''$$

$$Q = ((A'B)'.(BC')')'$$

The simplified expression of $Q = ((A'B)'.(BC')')'$ has been represented and simulated through a logic circuit using only NAND gates. The truth table shown below was produced from the simulated circuit and it matches the original Boolean expression.



Logic Converter-XLC1

	A	B	C	D	E	F	G	H	
000	0	0	0						0
001	0	0	1						1
002	0	1	0						1
003	0	1	1						0
004	1	0	0						0
005	1	0	1						1
006	1	1	0						0
007	1	1	1						1

Conversions

Out

(Figure G – Simulated Simplified Boolean Expression - $A'B + BC'$)

Task 3 - Multiplexer

A multiplexer is a combinational logic circuit designed to switch one of several input lines through to a single common output line by the application of a control signal. Multiplexers can also be used to implement Boolean functions of multiple variables.

Figure 'H' is a truth table for a 4-1 multiplexer. C0 to C3 represent four inputs which can be directed to the output, S1 and S0 represent the inputs that select which input is directed to the output and Q is the single output.

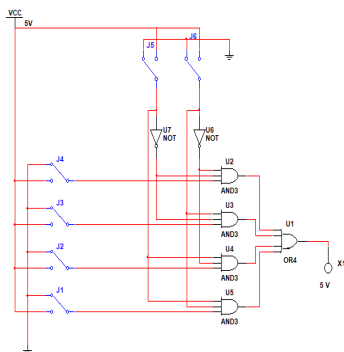
S1	S0	C0	C1	C2	C3	Q
0	0	1	X	X	X	C0
0	1	X	1	X	X	C1
1	0	X	X	1	X	C2
1	1	X	X	X	1	C3

(Figure H – Truth Table for 4-1 multiplexer)

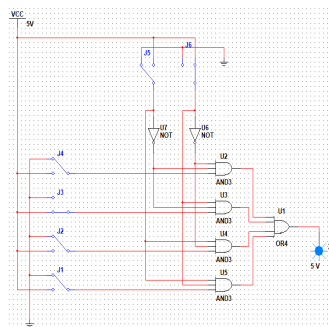
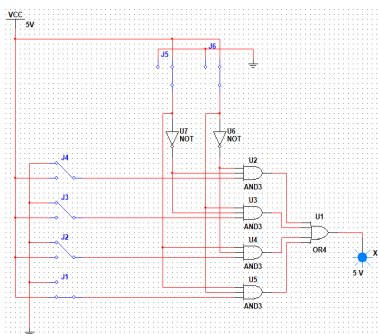
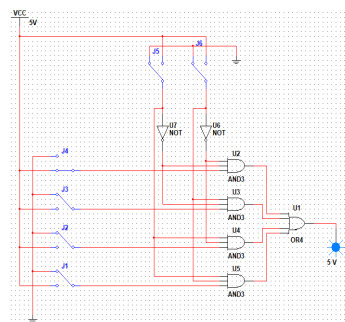
A Boolean expression can be formed from the information within figure H;

$$Q = S1'.S0'.C0 + S1'.S0.C1 + S1.S0'.C2 + S1.S0.C3$$

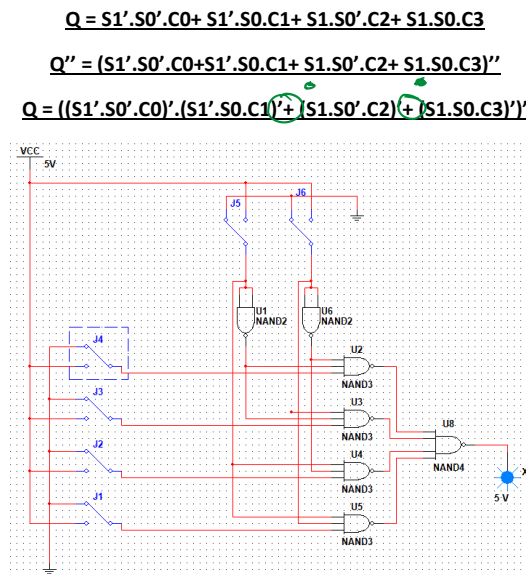
Using the formed Boolean expression above, a 4 to 1 multiplexer circuit was built and simulated. Screenshots below provide the particular expected outcomes when different inputs are applied. All results correlate with figure H.



(Figure I – 4-1 Multiplexer)



Through the use of De Morgans theorem, the formed Boolean expression produced from figure H can be used to convert the logic into NAND gates. This meant a circuit could be constructed and tested using only NAND gates.



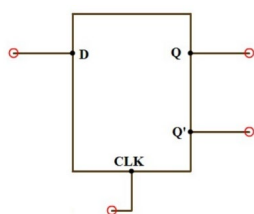
(Figure J – NAND gate 4-1 Multiplexer)

Task 4 – Sequential Logic

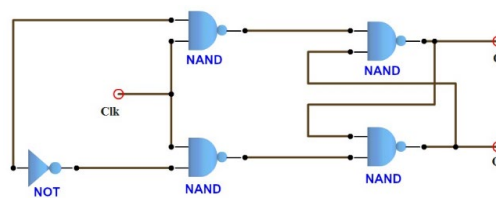
Flip-flops are formed using logic gates, which are formed using transistors. Flip-flops are basic building blocks in the memory of electronic devices. Each flip-flop can store one bit of data, possessing two stable states; high (logic 1) and low (logic 0). Flip-flops can be used for registering, counting, data transferring and frequency dividing. (Hub, 2012)

Commented [RB3]: Good description.

One of the most commonly used flip-flops is a D flip flop. This is a flip flop where the inputs are clock and data, with it possessing two outputs that complement each-other. The symbol for a D flip flop and a circuit using NAND gates has been shown below. A truth table of a D flip flop has also been included below.



(Figure K – D flip flop symbol)

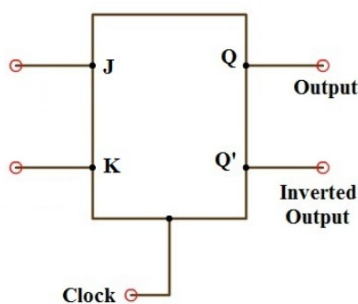


(Figure L – Circuit Diagram)

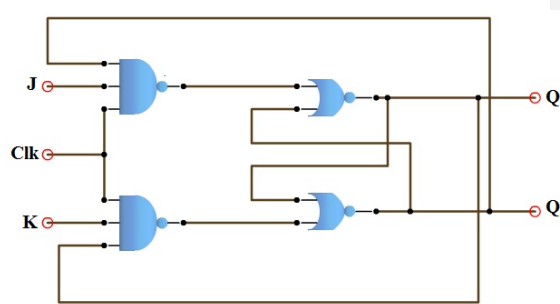
Clk	D	Q	Q'	State
0	0	Q	Q'	No change in state
1	0	0	1	Resets Q to 0
1	1	1	1	Sets Q to 1

(Figure M – Truth Table for D flip flop)

The JK flip flop is a universal flip flop. The two inputs of 'J' and 'K' do not represent an abbreviated word, however they are letters chosen by the inventor Jack Kilby to distinguish the flip flop design from others. The JK flip flop has two data inputs, J and K along with a clock input. The flip flop has two outputs Q and Q'. Included below is the symbol for a JK flip flop, a circuit diagram and a truth table. (Tutorials, 2015)



(Figure N – Symbol for JK flip flop)



(Figure O – Circuit Diagram)

Clk	J	K	Q	Q'	State
1	0	0	Q	Q'	No change in state
1	0	1	0	1	Resets Q to 0
1	1	0	1	0	Sets Q to 1
1	1	1	-	-	Toggles

(Figure P – Truth Table for JK flip flop)

(Hub, 2012)

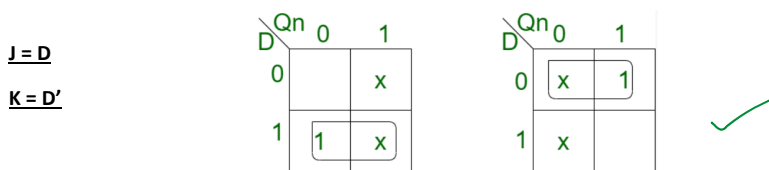
To convert a JK flip flop into a D flip flop, a characteristic truth table of a D flip flop and excitation table of a JK flip flop must be created.

Commented [RB4]: You could show both truth tables first before utilizing them.

<u>D</u>	<u>Q_n</u>	<u>Q_{n+1}</u>	<u>J</u>	<u>K</u>
<u>0</u>	<u>0</u>	<u>0</u>	<u>0</u>	<u>X</u>
<u>0</u>	<u>1</u>	<u>0</u>	<u>X</u>	<u>1</u>
<u>1</u>	<u>0</u>	<u>1</u>	<u>1</u>	<u>X</u>
<u>1</u>	<u>1</u>	<u>1</u>	<u>X</u>	<u>0</u>

(Figure Q – Truth and excitation table)

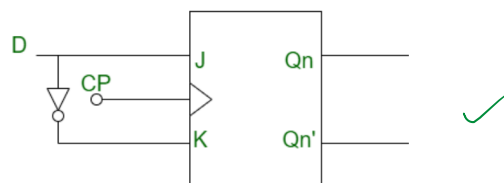
With the use of Karnaugh Maps, a Boolean expression can be formed of J and K in terms of D;



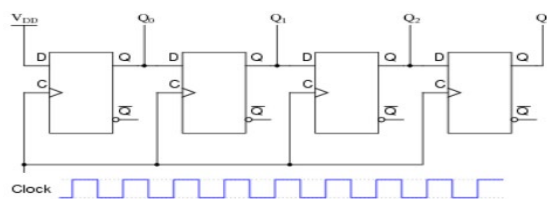
(Figure R –Karnaugh Maps)

(GeeksforGeeks, 2013)

From knowing the outcomes provided by the Karnaugh Maps, a circuit diagram can be constructed of the conversion of JK flip flop to D flip flop;



(Figure S –Final converted circuit diagram)



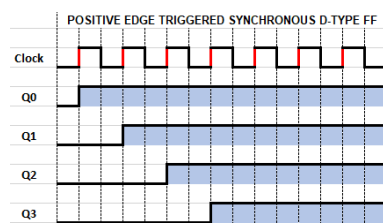
(Shift Register)

As all Q inputs were set to 0, and we needed all Q outputs to be set to 1 in a sequence of clock pulses. The clock input has a single arrow at its connection point which indicates the output sees the input on the rising edge of the clock pulse.

Clock	Data	Q0	Q1	Q2	Q3
initial	1	0	0	0	0
1	1	1	0	0	0
2	1	1	1	0	0
3	1	1	1	1	0
4	0	1	1	1	1

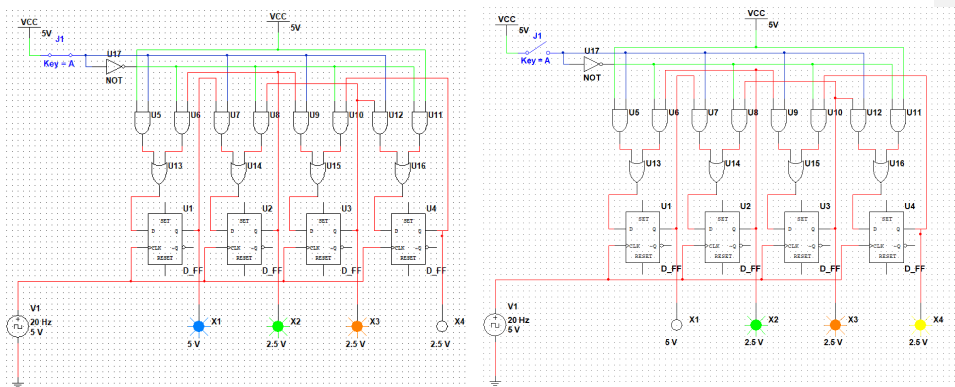
(Truth table)

Truth table for the above shift register, with this a timing diagram can be formed to show the process of which the shift register takes and the desired outputs that would be achieved.



(Timing diagram)

Modified bi-directional shift register was constructed and simulated. Below are screenshots of the lights being triggered in both directions. The bidirectional shift register shifts data either right or left depending on the mode selected. If the mode selected is 1(high), the data will be shifted towards the right direction and if the mode selected is 0(low), the data will be shifted towards the left direction. Figure T shows the circuit shifting from left to right with the switch at VCC on and shows the circuit shifting from right to left with the switch at VCC off.



(Figure T – Bi-directional shift register simulations)

Commented [RB5]: Well done, timing diagrams nicely explained and well presented

Commented [RB6]: Good, however, you could also explain how additional gates being used there are helping you to achieve the bidirectional logic.

Shift Registers are used for data storage or for the movement of data and are therefore commonly used inside calculators or computers to store data such as two binary numbers before they are added together, or to convert the data from either a serial to parallel or parallel to serial format. (Tutorials, 2010)

Commented [RB7]: Good application.

To improve this circuit, the introduction of an additional switch to control switching between high and low. Another improvement that could be made would be for the SET and RESET of the circuit would be connected as needed in a real-life scenario to avoid any unwanted confusion or connections being made. For the purposes of simulation on Multi-sim, this was not an issue.

Commented [RB8]: nice suggestions.

Task 5 - Using Sequential Logic to Solve Real Problems

To design a 3-stage synchronous binary counter using JK-flip flops that produce odd numbers from 0 to 8 (1, 3, 5, 7). The binary equivalent of these numbers have been identified through a binary to decimal conversion. A truth table has been created from this conversion, with the next state being included too.

A	B	C	A+	B+	C+
0	0	0	0	0	1
0	0	1	0	1	1
0	1	1	1	0	1
1	0	1	1	1	1
1	1	1	0	0	0

(Figure U – Next states truth table)

The next state truth table is used to confirm the desired outcome of each J or K input to change or maintain its current state in order to produce the binary counter 0, 1,3,5,7.

A	B	C	A+	B+	C+	JA	KA	JB	KB	JC	KC
0	0	0	0	0	1	0	X	0	X	1	X
0	0	1	0	1	1	0	X	1	X	X	0
0	1	1	1	0	1	1	X	X	1	X	0
1	0	1	1	1	1	X	0	1	X	X	0
1	1	1	0	0	0	X	1	X	1	X	1

(Figure V – Next State Table combined with excitation table)

Q	Q+	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	0	X	1

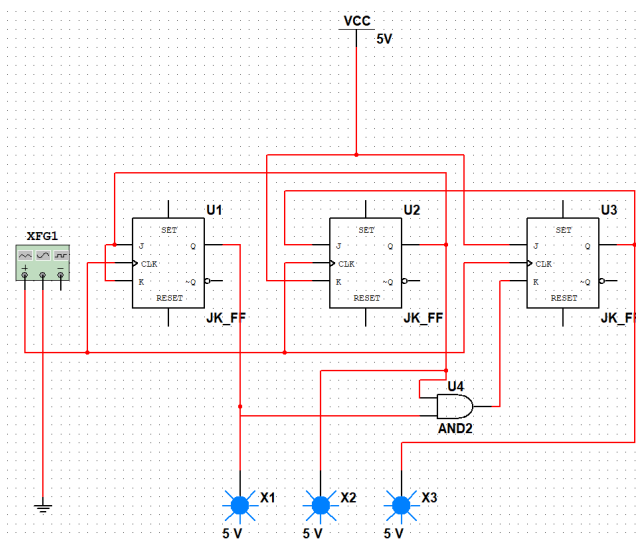
(Figure W – Excitation Truth Table)

From the tables provided above, Karnaugh maps can be formed for JA, KA, JB, KB, JC and KC. The outcomes provided from these maps will configure the layout of connections required to successfully create the counter.

A/BC	00	01	11	10	
0	0	0	1	X	JA = B
1	X	X	X	X	
A/BC	00	01	11	10	
0	X	X	X	X	KA = B
1	X	0	1	X	
A/BC	00	01	11	10	
0	0	1	X	X	JB = C
1	X	1	X	X	
A/BC	00	01	11	10	
0	X	X	1	X	KB = 1
1	X	X	1	X	
A/BC	00	01	11	10	
0	1	X	X	X	JC = 1
1	X	X	X	X	
A/BC	00	01	11	10	
0	X	0	0	X	KC = AB
1	X	0	1	X	

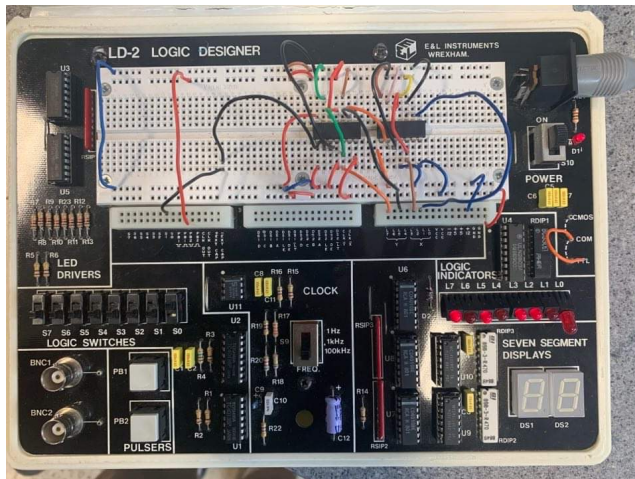


(Figure X – Karnaugh Maps)



(Figure Y – Designed and simulated circuit from Karnaugh map outcomes)

Commented [RB9]: Could explain the functionality of the circuit by taking one case into account.



(Built circuit on breadboard)

The simulated circuit in figure Y was then built on breadboard using 74HC74 ICs to replicate and produce the outputs provided on Multi-sim. The circuit functioned as required, with the only issue of the circuit being an incorrect use of IC.

Commented [RB10]: This was shift register circuit?

Task 6 - Logic Families and Their Characteristics

Digital systems use digital ICs, and when a system is constructed the ICs used in the design should all come from the same logic family to maintain simplicity and compatibility in the design. The different logic families must be understood so that the most suitable combinations of ICs are used. A logic family are specific circuit configurations that possess identical electrical parameters. (Electrical4u, 2014) A logic family is designed by considering the electronic components involved such as resistors, diodes, MOSFET and transistors. (Athshala, 2009)

Commented [RB11]: Please use the Full stop after citation.

Examples of a logic family are;

- TTL (Transistor Transistor Logic)
- CMOS (Complementary Metal Oxide Semiconductor Logic)

Characteristics of TTL and CMOS

Power Dissipation

This is the power supplied required to operate the gate. It is expressed in mill watt (mW) and represents actual power dissipated in the gate. It is the number that represents power delivered to gate from the power supply. The total power dissipated in the digital system is sum of power dissipated in each digital IC. TTL average power dissipation is 22.6 mW and relatively high in comparison to CMOS. This is because CMOS dissipates power only when switching, and nearly zero power when idle making CMOS very efficient.

Commented [RB12]: Interesting finding.

Propagation Delay

Propagation delay is the time required for a digital signal to travel from the input of a logic gate to the output. There are propagation delays present in both TTL and CMOS. On average, the propagation delays of TTL are usually 10ns whereas the propagation delays for the CMOS lies between 20 to 50 ns. (Components101, 2017)

Fan in/fan out

The fan in of a logic gate is defined as the number of inputs that the gate is designed to handle. The fan out of a logic gate is defined as the maximum number of standard loads that the output of the gate can drive without impairing its normal operation

The number of standard loads that could be connected to the output of the gate under the normal operation that is the fan-out is 10 for TTL whereas it is 50 for the CMOS. The number of standard inputs that can be connected to the gate is the fan in, which is approximately 12-14 for the TTL and for the CMOS it is 10 only. (Components101, 2017)

Current and voltage parameter

Voltage threshold at which the logic decides either logic 1 or 0. There is a substantial difference in the voltage level range for both. CMOS technology is more economical and preferred more as compared to the TTL logic. For TTL it is 4.75 V to 5.25 V while for CMOS it ranges between 0 to 1/3 VDD (Voltage at drain) at a low level and 2/3VDD to VDD at high levels. (TheGoldenSpoon, 2021)

Speed Power Product

This is the amount of speed produced by the lowest power consumed, with the lower the speed power product the better. The unit of speed-power product is the picojoule (pJ). For example, CMOS has a speed-power product of 1.2 pJ at 100 kHz while LS TTL has a value of 22 pJ. (Inflibnet, n.d.)

Characteristic	Logic family			
	74	74LS	74HC	40BE
Maximum supply voltage	5.25V	5.25V	5.5V	18V
Minimum supply voltage	4.75V	4.75V	4.5V	3V
Static power dissipation (mW per gate at 100kHz)	10	2	negligible	negligible
Dynamic power dissipation (mW per gate at 100kHz)	10	2	0.2	0.1
Typical propagation delay (ns)	10	10	10	105
Maximum clock frequency (MHz)	35	40	40	12
Speed-power product (pJ at 100kHz)	100	20	1.2	11
Minimum output current (mA at $V_{OUT} = 0.4V$)	16	8	4	1.6
Fan-out (LS loads)	40	20	10	4
Maximum input current (mA at $V_{IN} = 0.4V$)	-1.6	-0.4	0.001	-0.001

(Figure Z – Table comparing various members of the TTL family with buffered CMOS logic)

(Tooley, 2003)

Commented [RB13]: Good study.

Commented [RB14]: Well done Billy, each characteristic is well investigated, however a conclusion could have helped.

References

Athshala, 2009. *Logic Families*. [Online]

Available at: [Microsoft Word - p4m01_etext_final.docx \(inplibnet.ac.in\)](#)

[Accessed 2021].

Components101, 2017. *An Overview of TTL and CMOS ICs and How to Choose Between Them*.

[Online]

Available at: <https://components101.com/articles/ttl-vs-cmos-ics-and-how-to-choose-between-them#:~:text=On%20average%2C%20the%20propagation%20delays,expensive%20with%20the%20CMOS%20chips.&text=The%20current%20requirements%20of%20the,thus%20power%20consumption%20is%20li>

[Accessed 2021].

Electrical4u, 2014. *Logic Families Significance and Types of Logic Families*. [Online]

Available at: [Logic Families Significance and Types of Logic Families | Electrical4U](#)

[Accessed 2021].

GeeksforGeeks, 2013. *Conversion of J-K Flip-Flop into D Flip-Flop*. [Online]

Available at: [Conversion of J-K Flip-Flop into D Flip-Flop - GeeksforGeeks](#)

Hub, E., 2012. *Introduction to Flip Flops*. [Online]

Available at: [Introduction to Flip Flops \(electronics-hub.org\)](#)

[Accessed 2021].

Inplibnet, n.d. *Speed Power Product*. [Online]

Available at: [Microsoft Word - p4m01_etext_final.docx \(inplibnet.ac.in\)](#)

[Accessed 2021].

TheGoldenSpoon, 2021. *CMOS Logic Levels*. [Online]

Available at: <https://thegoldenspoon.com.vn/wp-content/uploads/vanilla-beans-ndjuk/viewtopic.php?a86609=cmos-logic-levels>

[Accessed 2021].

Tooley, M., 2003. *Logic Families*. [Online]

Available at: [Logic families \(key2study.com\)](#)

[Accessed 2021].

Tutorials, E., 2010. *The Shift Register*. [Online]

Available at:

https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwjYttGJrdiwAhXYtqQKHVjTBbwQFjAFegQIDBAD&url=https%3A%2F%2Fwww.electronics-tutorials.ws%2Fsequential%2Fseq_5.html&usg=AOvVaw1wxpE9eJCGSlioobcm8lq1

Tutorials, E., 2015. *JK Flip Flop and the Master-Slave JK Flip Flop Tutorial*. [Online]

Available at: [JK Flip Flop and the Master-Slave JK Flip Flop Tutorial \(electronics-tutorials.ws\)](#)

Bibliography

- [JK Flip Flop and the Master-Slave JK Flip Flop Tutorial \(electronics-tutorials.ws\)](#)
- [Difference Between TTL and CMOS ICs and How to Choose Between Them \(components101.com\)](#)
- <https://www.youtube.com/watch?v=ApJ972OYyXQ>
- https://www.electronics-tutorials.ws/sequential/seq_2.html
- <https://www.electrical4u.com/logic-families-significant-and-types/#:~:text=Significance%20of%20Logic%20Families,and%20the%20goal%20is%20achieved.>
- <https://www.geeksforgeeks.org/conversion-of-j-k-flip-flop-into-d-flip-flop/>
- <https://learnabout-electronics.org/Digital/dig31.php>
- <https://www.coursehero.com/file/p7o5kgp/Speed-Power-Product-The-speed-power-product-provides-a-basis-for-the-comparison/#:~:text=The%20unit%20of%20speed%2Dpower,a%20value%20of%2022%20pJ.>

STUDENT FEEDBACK FORM

This section details the *extent* to which the assessment criteria are demonstrated by you, which in turn determines your mark. The marks available for each category of skill are shown. Lecturers will use the space provided to comment on the achievement of the task(s), including those areas in which you have performed well and areas that would benefit from development/improvement.

Generic Assessment Criteria	Marks available	Marks awarded
1. Engagement with Literature Skills	10	
You have used a wide range of the literature including books and relevant website to provide supporting evidence for your explanations throughout the assignment except in task 1 and 2. Referencing conventions are consistent and right with few inaccuracies.	64% 10	6.4
2. Knowledge and Understanding Skills	30	
Your work demonstrates your competent knowledge and understanding of the basic underlying concepts of digital electronics. You have used appropriate approach and theories to complete all set tasks throughout the assignment. All your calculation work is correct and clearly laid out by stating the laws of Boolean algebra being used at first place. Each characteristic of logic family is well investigated – providing the evidence of your reading and excellent understanding of the concept. Practical applications of various concepts are also well considered. However, at times, more efforts could be made to explain the functionality of the various circuits (see comments given in the assessment).	72% of 30	21.6
3. Cognitive and Intellectual Skills	10	
You have performed a sound evaluation of underlying concepts and principles, showing your ability to devise an argument from the evidence. Most of your work is supported by a good evaluation with suggestions to improve in place. However, at times, your evaluation could be expanded, for instance, in task 1 and 2 you could explain the benefits of using Boolean algebra and De-Morgan's theorem by analysing your own results.	65% of 10	6.5

4. Practical Application Skills	40	
You have demonstrated an excellent ability to perform simulations and practical work by verifying the simulated and practical results against your theoretical findings and relating it to the real-world situations. All the practical and simulated results are correct, very well presented and explained with suggestions to improve in place.	80% 40	32
5. Transferable for Life and Professional Practice	10	
Overall, your work is accurate, coherent, fluent well-structured and organised, except that few times, your sentences could be restructured. Your usage of equation editor and Multisim screen shots when required to explain the concept deeply is worth appreciation. Your work clearly evident your ability to work effectively and independently or as a part of the group.	70% 10	7

Assessment Mark (Assessment marks are subject to ratification at the Exam Board. These comments and marks are to give feedback on module work and are for guidance only until they are confirmed.)	Late Submission Penalties (tick if appropriate)		73.5%
	Up to 1 week late (40% Max)		
	Over 1 week late (0%)		