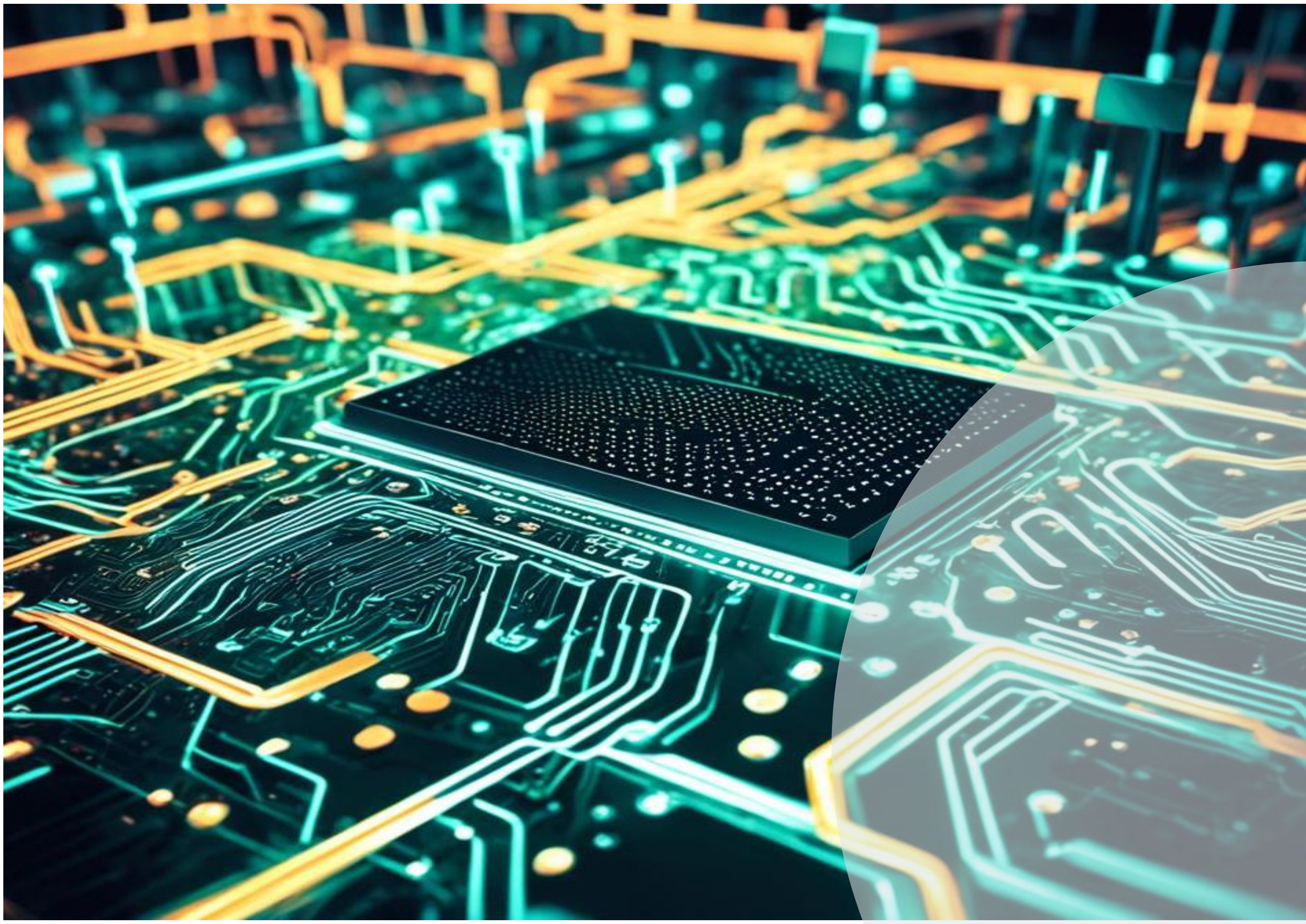
Faculty of Computing



NETWORK PACKET TRANSMISSION MONITORING SYSTEM

Lecturer: DR. FARKHANA BINTI MUCHTAR

Prepared by

PARTHIV GUNALAN	A24CS0178
GAVENESH A/L BATUMALAI	A24CS0076
HENG ZHI QIANG	A24CS0081
IDA YATULLAILIYEH BINTI AMRUN	A24CS0084

TABLE OF CONTENTS

DEDICATION & ACKNOWLEDGEMENT

This project is dedicated to our lecturer, Dr. Farkhana Binti Muchtar. We would like to express our heartfelt appreciation to Dr. Farkhana for her invaluable guidance in Digital Logic, which has been instrumental in the successful completion of this project. Her teaching not only allowed us to gain knowledge about digital components and concepts, but also helped us understand their practical applications, giving us a comprehensive grasp of Digital Logic subject.

We also wish to extend our special thanks to our dedicated group members. The collective effort of our team has been the cornerstone of our success. Each member contributed a unique set of skills and perspectives, creating a dynamic and innovative environment.

This collaboration not only motivated us, but also enhanced our critical thinking, problem-solving and teamwork abilities. Thank you, Dr. Farkhana and our team for an extraordinary learning experience.

BACKGROUND

In this project, a group of computers in Lab A are connected to a group of computers in Lab B via a cable. We want to transmit a number of data packet (bytes) (define from user) from one source of computer in Lab A to another destination computer in Lab B.

The full circuit is drawn using deeds and includes combinational circuits and sequential circuits.

Combinational Circuit Components include:

- 4-bit comparator
- 8x1 Multiplexer
- 1x8 DeMultiplexer
- Basic gates (AND, OR, NOT)
- Test LED
- Input switches
- 1-bit output display
- 4-bit Hex input

Sequential Circuit Components include:

- T flip-flop
- clock generator

Extra features made:

- power indicator
- security module
- fun concept display

PROBLEM STATEMENT

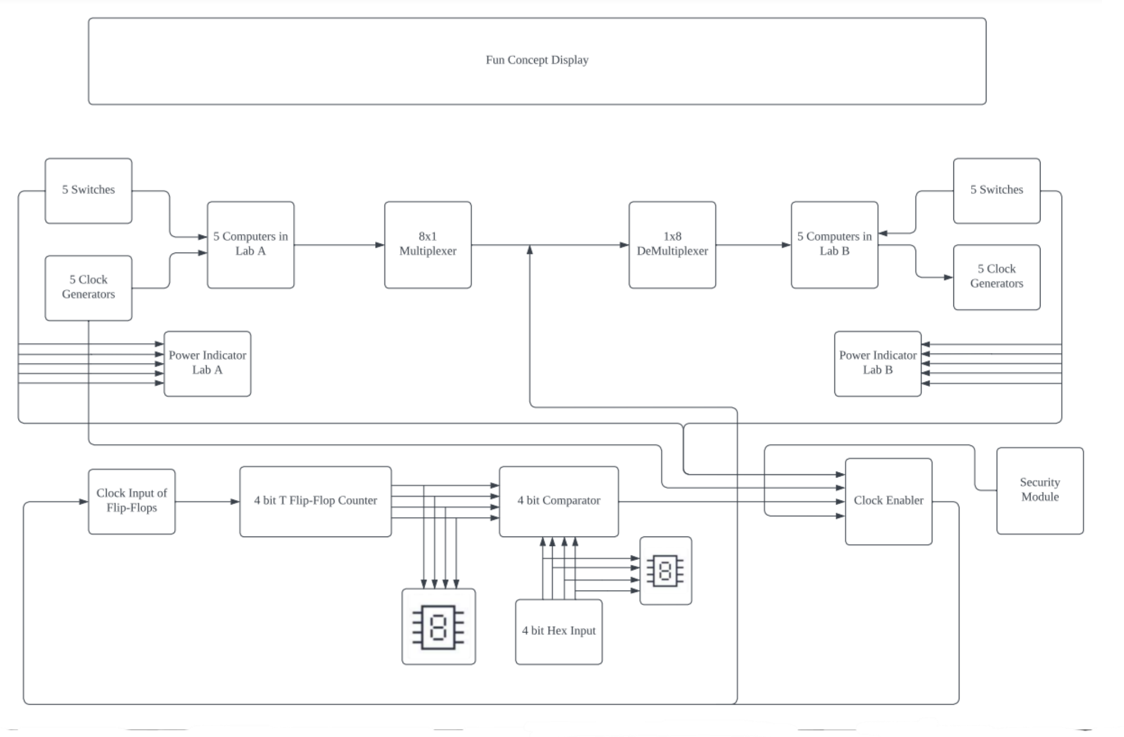
A group of computers are connected from Lab A to a group of computers in Lab B via one cable. We are required to build a circuit that will solve this issue. On top of that, we are also needed to count the amount of data transferred. Furthermore, the system must also ensure that the data transfer process halts once the pre-set maximum number of packets is reached. This is to prevent excessive data packet transmission and to manage the data packet efficiently.

Whilst completing the circuit, we are also required to add certain features such as comparators and clock enabler to smoothen the process of data packet transmission

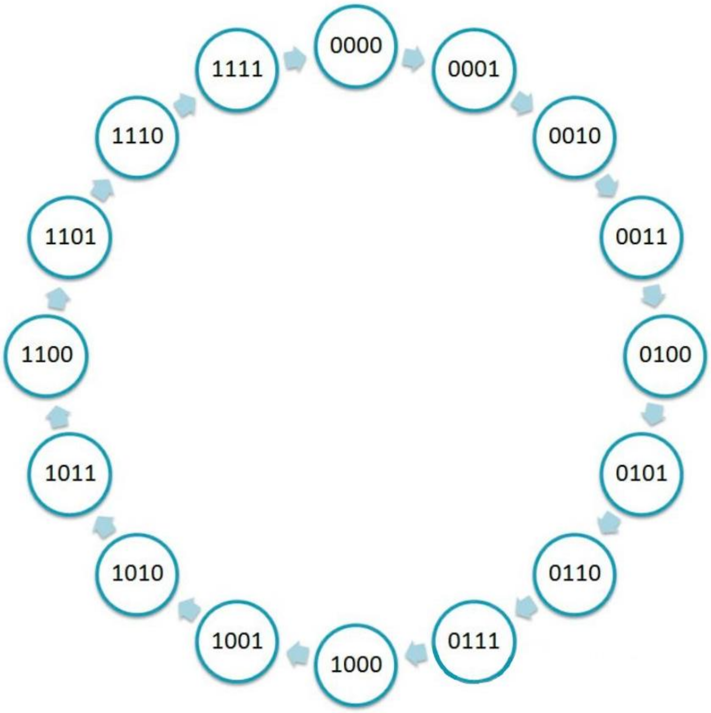
SUGGESTED SOLUTION

The system employs a series of mechanisms to prevent unauthorized or continuous data transfers. The AND gate ensures that data is only transferred when both the corresponding computer in Lab A is turned ON and the receiving computer in Lab B is also turned ON. The 4-bit Synchronous T flip-flop counter is used to keep track of the number of packets transferred. The flip-flops are connected to a singular clock generator, synchronizing the counting process. The counter's output is compared with a 4-bit Hex input, which represents the maximum packet count set by the user. The comparator compares the counter value with the input, and once both values match, the system cuts off the data packet transfer using the count enabler. The count enabler, which is driven by an AND gate, takes three inputs: the output of the comparator (indicating the counter has reached the max value), the combined clock signal from the computers in Lab A (checked by an OR gate), and the status of the computers being ON in both labs (checked by an AND gate). When all conditions are met, the count enabler allows the counter to continue counting; otherwise, it blocks the counting process, stopping further data transfer. To ensure that the transfer halts as soon as the max packet limit is reached, the NOT gate inverts the comparator's output, which then turns off the AND gate in the count enabler, cutting the connection between the Multiplexer and Demultiplexer. This entire process ensures that no further data packets are transferred once the limit is reached, effectively controlling the system's data transfer flow.

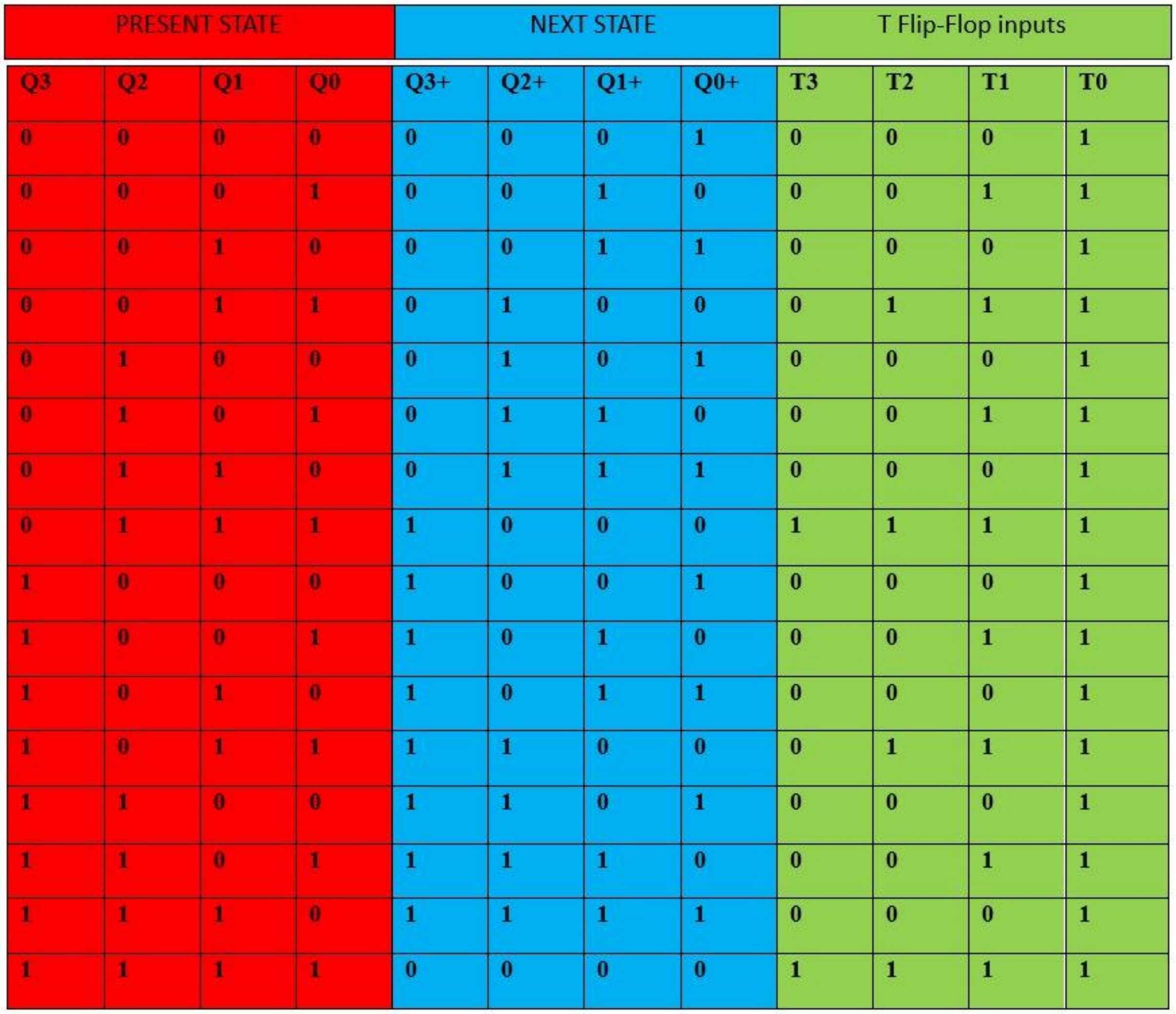
Block Diagram



State Diagram



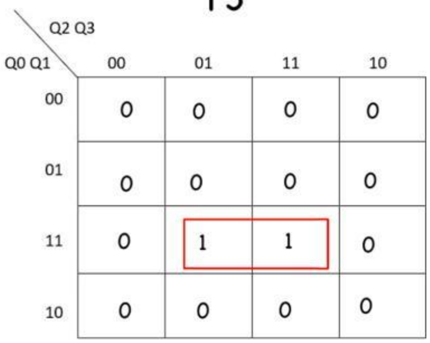
Flip-flop Excitation Table



PRESENT STATE				NEXT STATE				T Flip-Flop inputs			
Q3	Q2	Q1	Q0	Q3+	Q2+	Q1+	Q0+	T3	T2	T1	T0
0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	1	0	0	1	0	0	0	1	1
0	0	1	0	0	0	1	1	0	0	0	1
0	0	1	1	0	1	0	0	0	1	1	1
0	1	0	0	0	1	0	1	0	0	0	1
0	1	0	1	0	1	1	0	0	0	1	1
0	1	1	0	0	1	1	1	0	0	0	1
0	1	1	1	1	0	0	0	1	1	1	1
1	0	0	0	1	0	0	1	0	0	0	1
1	0	0	1	1	0	1	0	0	0	1	1
1	0	1	0	1	0	1	1	0	0	0	1
1	0	1	1	1	1	0	0	0	1	1	1
1	1	0	0	1	1	0	1	0	0	0	1
1	1	0	1	1	1	1	0	0	0	1	1
1	1	1	0	1	1	1	1	0	0	0	1
1	1	1	1	0	0	0	0	1	1	1	1

Karnaugh-Map

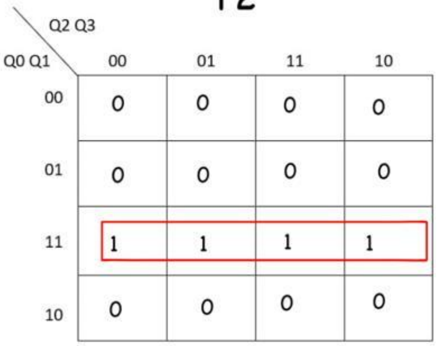
T3



Q0 Q1 \ Q2 Q3	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	0	1	1	0
10	0	0	0	0

T3=Q0.Q1.Q2

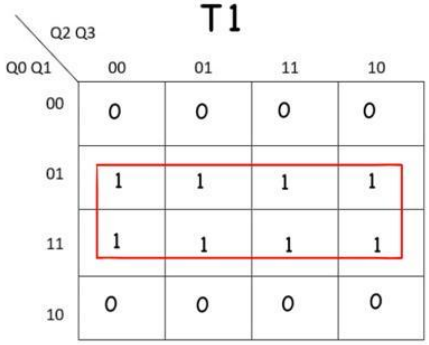
T2



Q0 Q1 \ Q2 Q3	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	1	1	1	1
10	0	0	0	0

T2=Q0.Q1

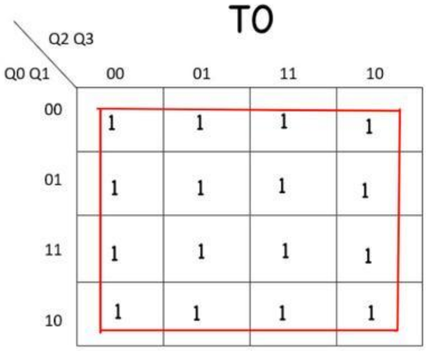
T1



Q0 Q1 \ Q2 Q3	00	01	11	10
00	0	0	0	0
01	1	1	1	1
11	1	1	1	1
10	0	0	0	0

T1=Q0

T0



Q0 Q1 \ Q2 Q3	00	01	11	10
00	1	1	1	1
01	1	1	1	1
11	1	1	1	1
10	1	1	1	1

T0=1

REQUIREMENTS

Connection Computers from Lab A to Lab B

- 1x8 Multiplexer
- 8x1 DeMultiplexer

Count Up Counter Function

- 4-bit Synchronous T Flip-Flop Counter

Comparator Function

- Comparator Module & NOT Gate

Clock Enabler Function

- AND gate

Monitoring Packet Transfer Function

- 1x8 Multiplexer
- 8x1 DeMultiplexer
- 1-bit Output Display

Display Controller (Fun Concept) Function

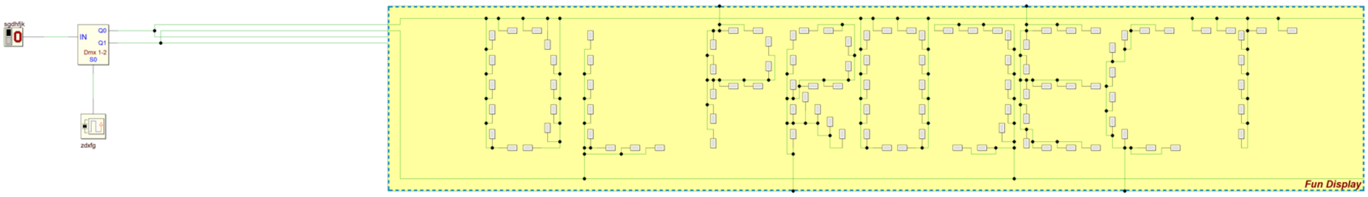
- 1x2 DeMultiplexer
- Clock Generator
- Input Switch

SYSTEM IMPLEMENTATION

By adhering to the requirements, we have used 1x8Multiplexer and 8x1DeMultiplexer for the connection between the computers from Lab A to Lab B. For the count up counter function, we have used synchronous counter and JK flip-flop to build the 4-bit synchronous T flip-flop counter (since there’s no T flip-flop in deeds). Comparator module is used for comparator functions while AND gate is used for clock enabler function. Finally, we have used MUX, DEMUX and basic gates to enable the function of monitoring the packet transfer.

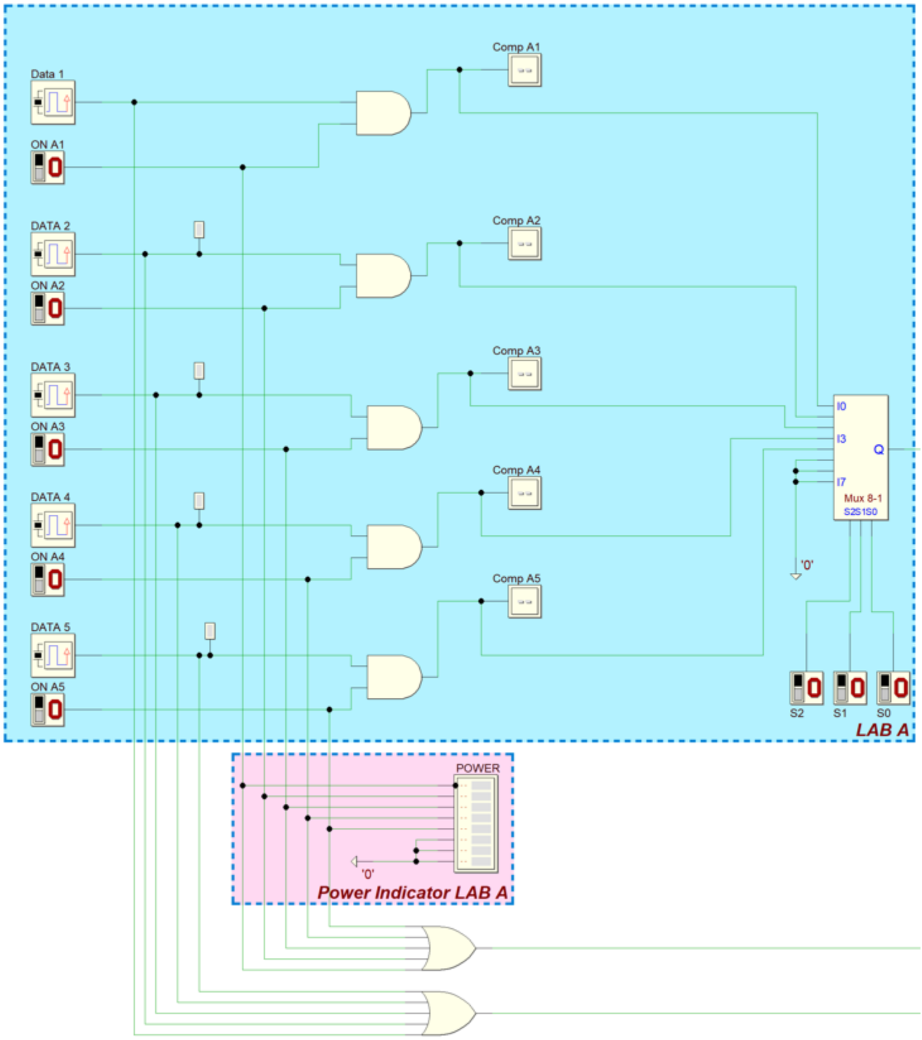
The following is our system implementation on network packet transmission monitoring system (with deeds and explanations).

Fun Concept Display



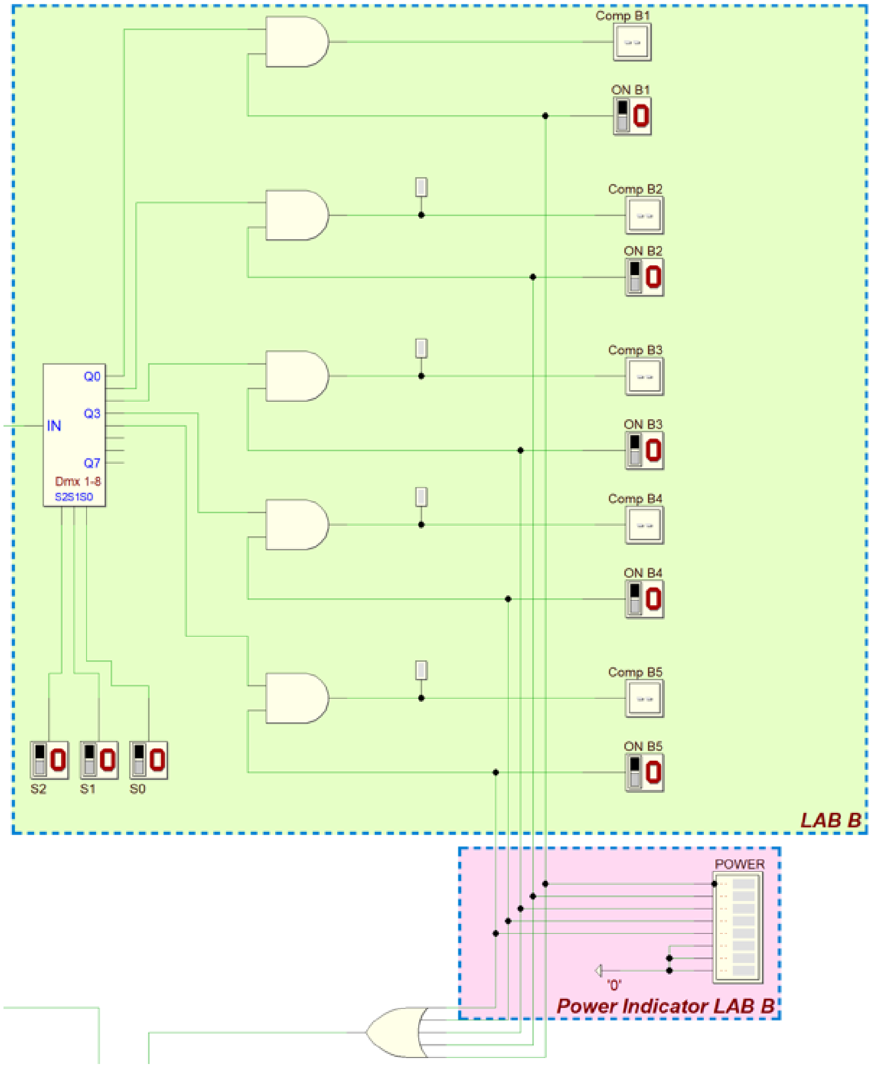
The fun concept display is built using a 1x8 DEMUX, clock generator, switches and the LEDs as the output. This fun concept display with the title "DL PROJECT" was built to make the deeds circuit more appealing.

Computers in Lab A



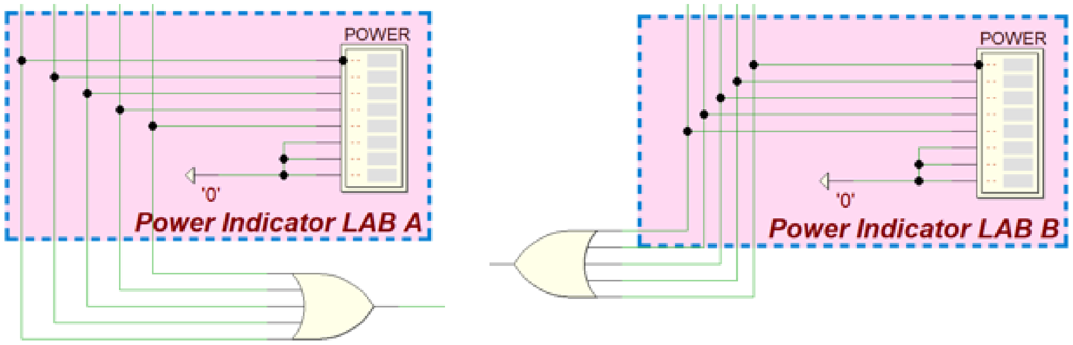
In Lab A, all 5 computers are connected to an 8x1 multiplexer (MUX). Each of them is then connected with a button and a clock. The button is used to control the ON and OFF for the computer while the clock indicates the data packet transfer of the computer. There is an AND gate connecting the button and the clock for each computer. It ensures that the data packets are transferred if and only if the computer is ON. The output display indicates the status of the data packet transferred (HIGH- data packet successfully transferred, LOW- no data packet transferred). LED will light up once there is any data packet transfer or in another words, the clock is activated.

Computers in Lab B



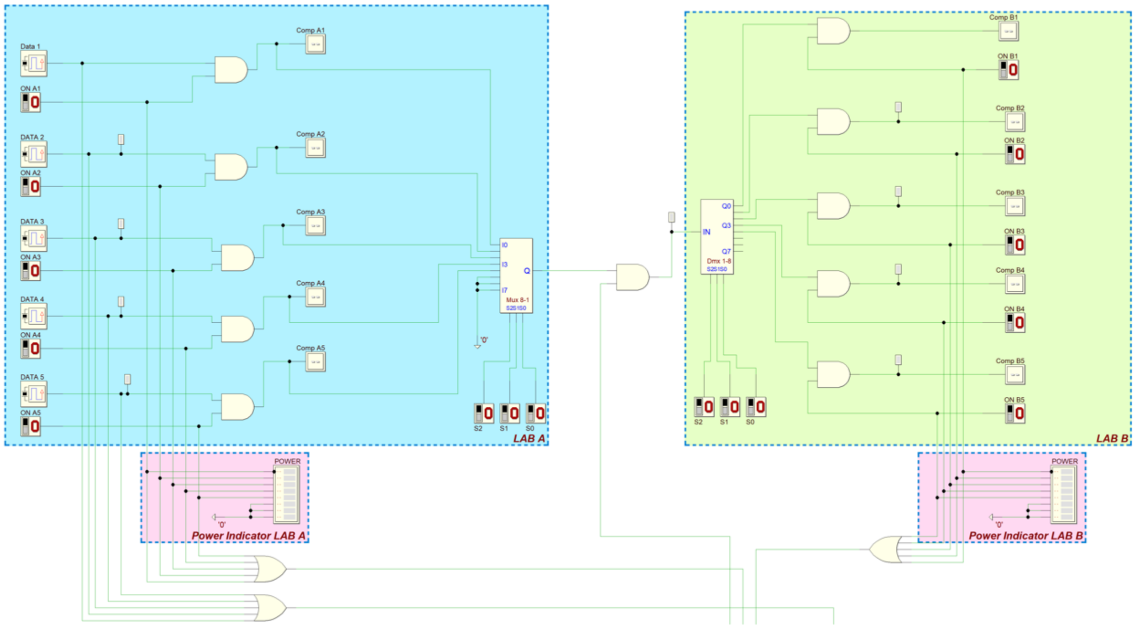
In Lab B, all 5 computers are connected to a 1x8 demultiplexer (DEMUX). Each of them is then connected with a button and an output display. The button is used to control the ON and OFF for the computer while the output display shows whether the computer received the data packet or not. For each computer, there is an AND gate connecting the button and distinct output from DEMUX. It ensures that the data packets are received by the computer if and only if the computer is ON. The output of the AND gate is connected to the output display (HIGH- data packet successfully received, LOW- no data packet received). LED will light up once there is any data packet received by the computer or in another words, the output is 1.

Power Indicator for Lab A & Lab B



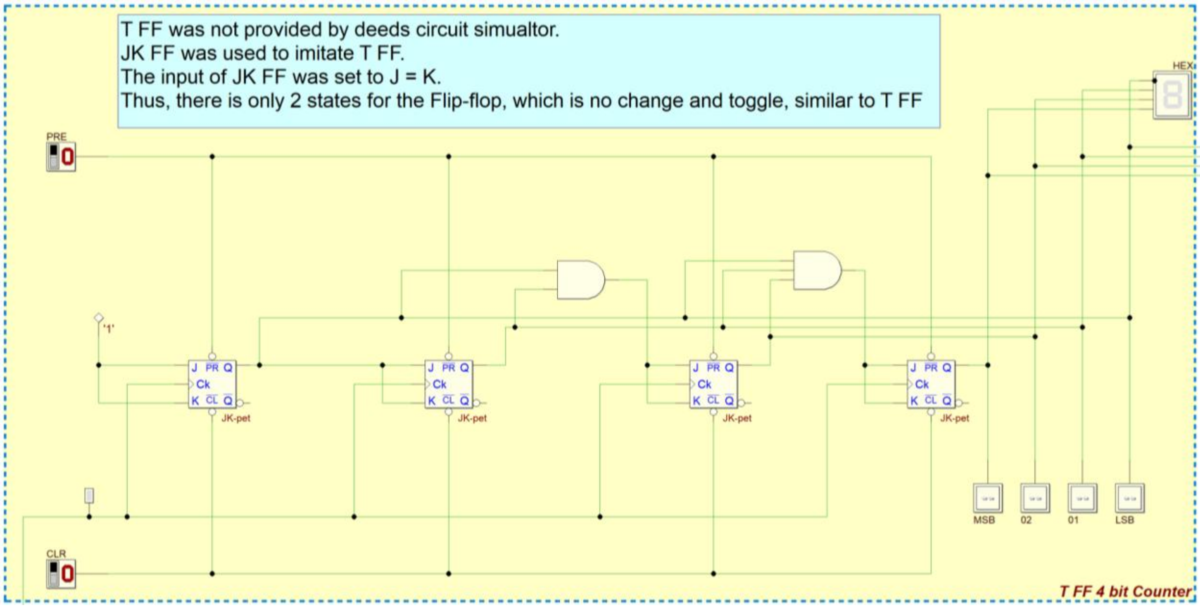
There is a power indicator for each Lab A & Lab B which is built using an 8-bit output LED array display. The power indicator is connected to the switch of all 5 computers in each respective lab. The power indicator is used to show the status of which computer is currently turned ON. It helps the user to know which computer can be used for data packet transmission.

Connection between Lab A & Lab B



For the full diagram involving Lab A and Lab B, both multiplexer (MUX) and demultiplexer (DEMUX) are used and there is a connection between Lab A and Lab B. The 8x1 MUX in Lab A is connected to 1x8 DEMUX in Lab B by coaxial cable and between the 8x1 MUX and 1x8 DEMUX, there is an AND gate having an input from MUX and an input from clock enabler. It is used to cut off the data packet transfer once the counter has reached maximum data packet amount (HIGH- data packet continues to transfer to DEMUX, LOW- no more data packet transfer to DEMUX). At the same time, the user can know which computer can be used for data transmission by looking at the status of computer whether it is turned ON or turned OFF.

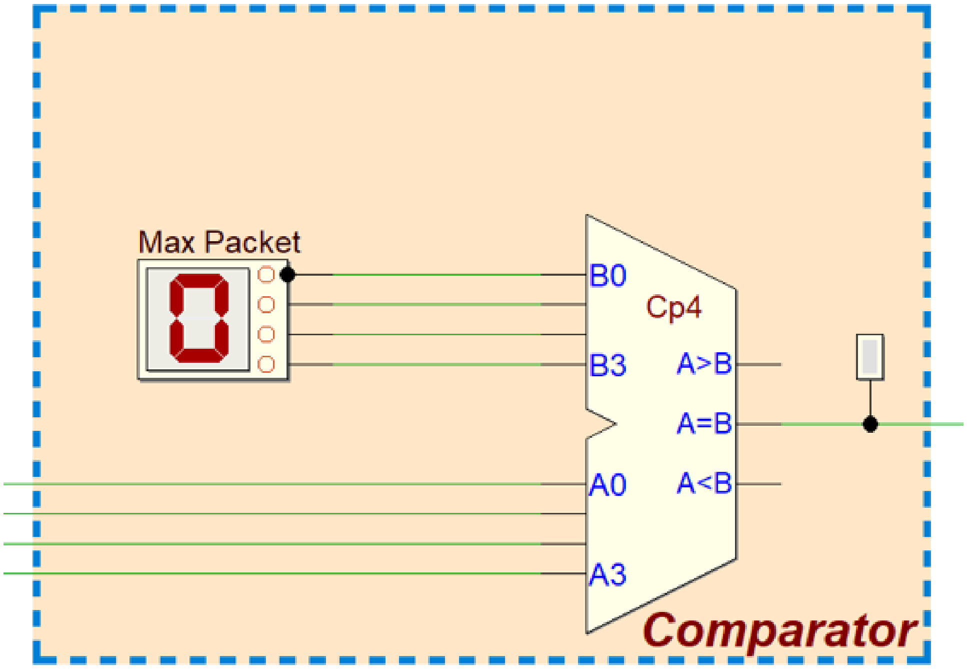
4-bit Synchronous T Flip-flop Counter



T FF was not provided by deeds circuit simualtor.
JK FF was used to imitate T FF.
The input of JK FF was set to J = K.
Thus, there is only 2 states for the Flip-flop, which is no change and toggle, similar to T FF

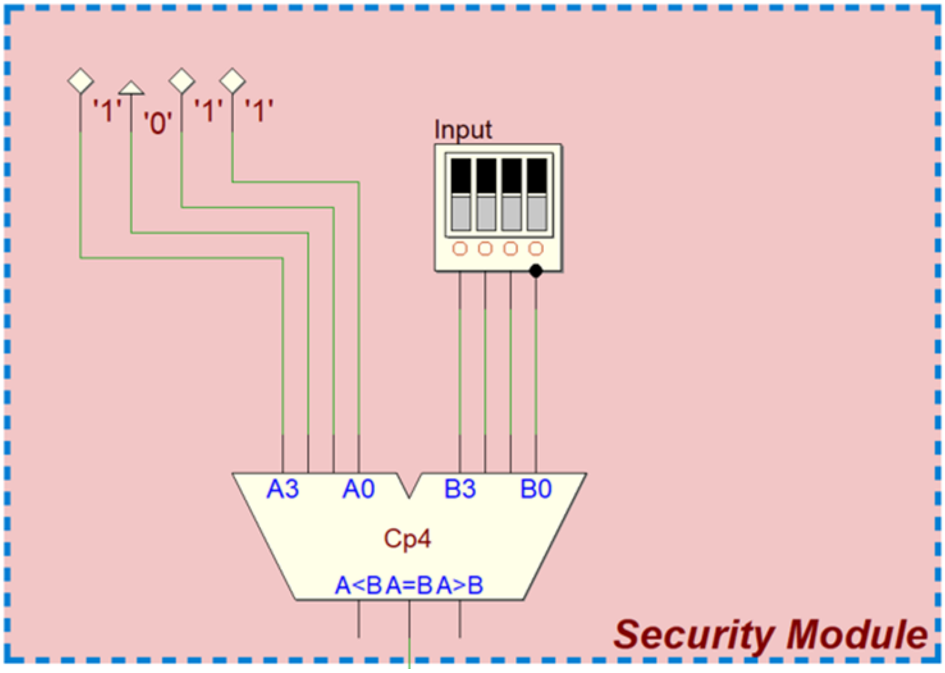
All the clock input for the T flip-flops comes from a common clock- the clock enabler. The flip-flops only work if and only if the clock input is 1.

Comparator



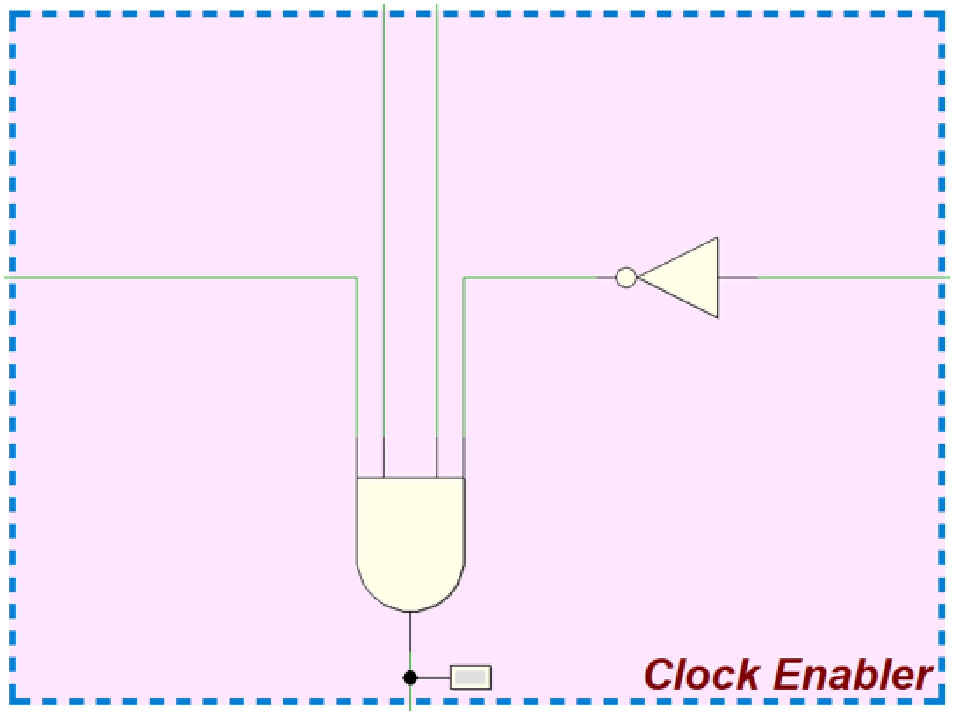
The first 4 inputs of comparator (B0-B3) are connected to 4-bit HEX input while the second 4 inputs (A0-A4) are connected to 4 outputs of counter from each flip-flop. The user can control the maximum data packet to be transferred, and the comparator will work to compare between maximum data packet and the counter. Once the counter reaches the same amount as the maximum data packet, it will cut the connection between MUX and DEMUX at coaxial cable. Hence, there will be no more data packet transferred.

Security Module



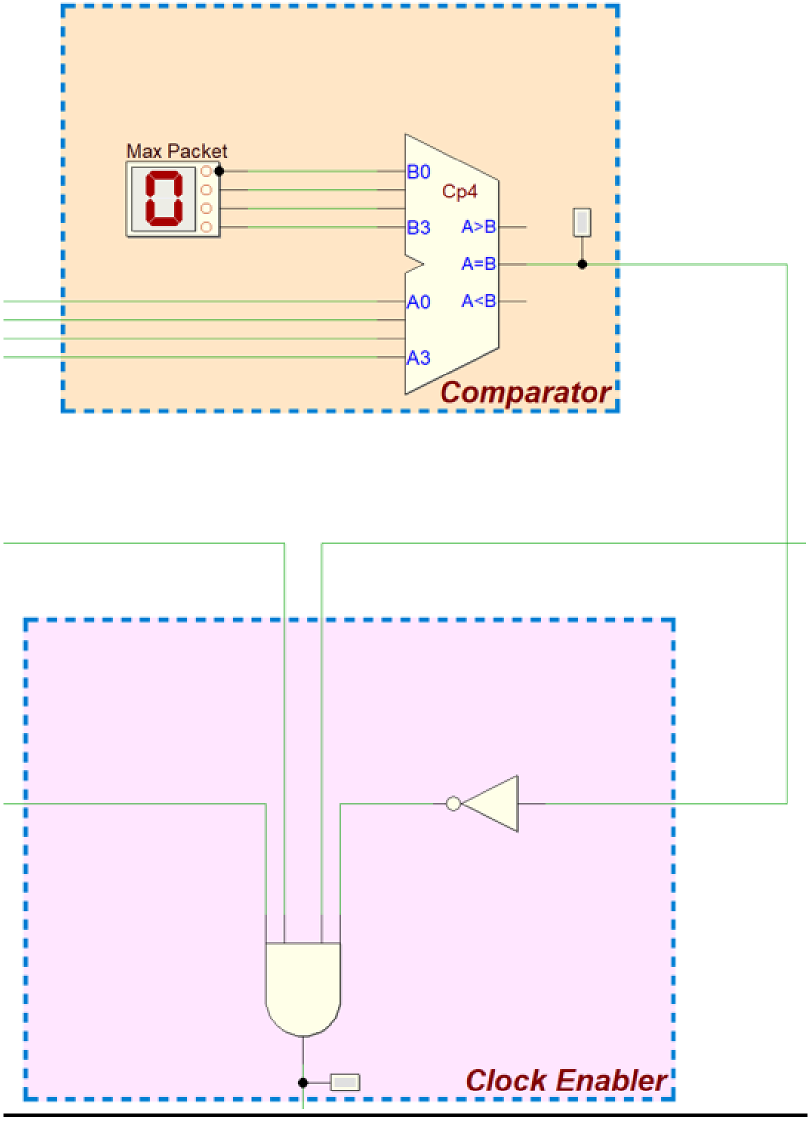
The security module is built using a 4-bit comparator and 4-bit input dip-switch. The security module is added as an extra feature for the computers. The password of the security module is set to 1011. If the user inputs any other value, thus the security module will not enable the clock enabler. Hence, there will be no data packet transmission.

Clock Enabler



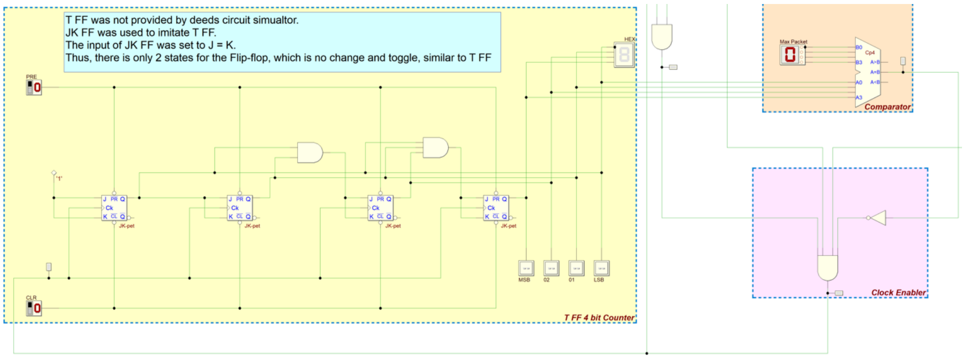
For clock enabler, 4-input AND gate is used. One of the inputs is from comparator in which once it reached the same amount as maximum data packet, another three inputs are from OR gate that connects or combines all the clocks of computers, from AND gate that is used to check if the button to corresponding computers in Lab A and the data receiving computers in Lab B are both turned ON, and lastly from the security module that read the password entered by the user, if the password is correct (1011), then it enables the clock enabler. Once all 4 conditions are met (output of 4-input AND gate for clock enabler = HIGH, in which output from OR gate is HIGH & output from AND gate is HIGH & output from security module is HIGH & output from comparator is LOW and after it inverts, it becomes HIGH), the clock enabler will start counting up. If not, the clock enabler will stop and block the connection between MUX and DEMUX at coaxial cable which stops the transmission.

Connection between Clock Enabler & Comparator



Once the 4-bit HEX input and the counter are the same, the comparator gives an output of 1. Since there is a NOT gate (inverter) at the wire connecting to the clock enabler, it inverts the input of 1 for the clock enabler which originates from the output of comparator to 0. With the AND gate there, the clock enabler will stop the counter and cut the data packet transfer at coaxial cable between MUX and DEMUX.

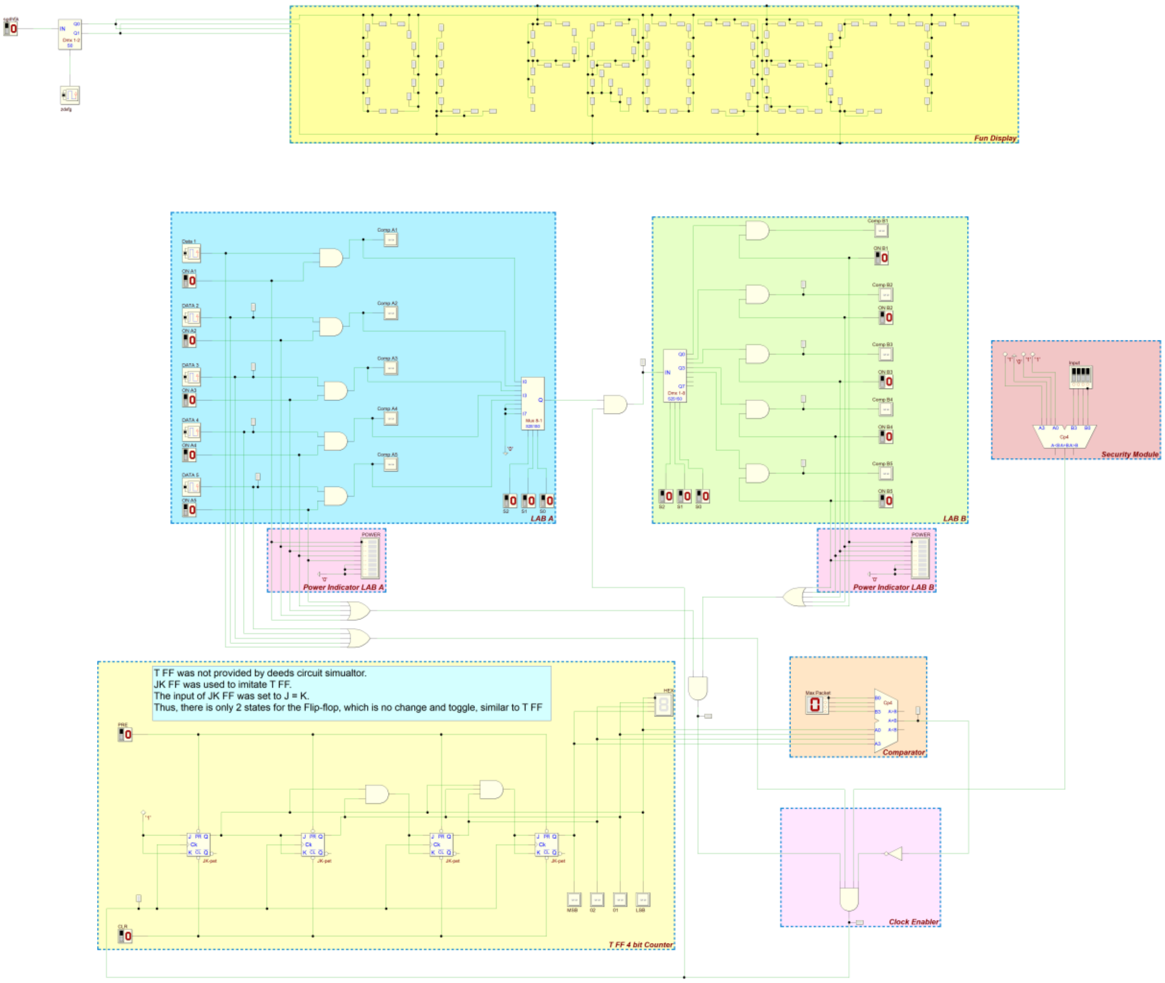
Connection between Clock Enabler, Comparator & 4-Bit Synchronous T Flip-Flop Counter



T FF was not provided by deeds circuit simualtor.
JK FF was used to imitate T FF.
The input of JK FF was set to J = K.
Thus, there is only 2 states for the Flip-flop, which is no change and toggle, similar to T FF

The output of the counter from each flip-flop is connected to the second input of the comparator. At the same time, the output of the clock enabler is connected to the clock input of each T flip-flop since common clock is used. If all 4 conditions in clock enabler are not met, there will be no changes for the clock input of each flip-flop, and it won’t activate the flip-flop. Hence, the counter will stop.

Full Circuit Diagram


Fun Display
LAB A
LAB B
Security Module
Power Indicator LAB A
Power Indicator LAB B
T FF was not provided by deeds circuit simualtor.
JK FF was used to imitate T FF.
The input of JK FF was set to J = K.
Thus, there is only 2 states for the Flip-flop, which is no change and toggle, similar to T FF
PRE
HEX
CLR
MSB
LSB
Comparator
Max Packet
Clock Enabler
T FF 4 bit Counter

The system will eventually stop the data packet transfer once the comparator reached the same amount as the maximum data packet defined by the user or the computers in Lab A and Lab B are all turned OFF. Inactivated clocks for computers in Lab A or wrong password entered by the user at security module will also stop the transmission. In this case, it causes the clock enabler to generate an output of LOW which is brought to the counter as clock input of each flip-flop. Hence, due to the input of LOW, the clock is not triggered to activate the flip-flops, and the counter doesn’t work anymore. With that, the connection between MUX and DEMUX at coaxial cable will be cut off, causing the data packet unable to be transmitted to the receiving computers through DEMUX.

CONCLUSION & REFLECTION

In a nutshell, the project about network packet transmission monitoring system is a huge success. The monitoring system is user-friendly and efficient in transmitting the network packet since the user can define the maximum data packet to be transferred. The circuit for the system mainly consists of a 4-bit comparator, 8x1 multiplexer, 1x8 demultiplexer, 4-bit Hex input, 4-bit decoder, T flip-flops (using JK flip-flops to construct since there's no T flip-flop in deeds), clock generator, test LEDs, switches, count enabler and some basic gates. While doing the project, we encountered some challenges, mainly in communication, in which we had different ideas on the system implementation, circuit design and the extra features. However, after a peaceful discussion among us, all the problems are resolved perfectly and we come up with the final system implementation and circuit design. We also added some extra features that made the system more functional, secure and appealing. For instance, the power indicator, security module, fun concept display.

Again, we would like to express our gratitude to Dr Fo'ad for the briefing of this project and Dr Farkhana for giving us an opportunity to do the project and in guiding us throughout the project. Not to mention, thanks to our dedicated members who work hard in this project.

In future projects, we hope to implement even more complex circuits with more exciting options so that we can improve our critical thinking and problem-solving skills. We also look forward to using our knowledge in related courses, for example, Computer Organization and Architecture in the upcoming semester, and applying it to real-life situations.

Thank you for your patience and time!

REFERENCES

Mohd Fo'ad, R. (2024). *Digital Logic Project Manual*. Johor Bahru, Johor: Faculty of Computing, University of Technology Malaysia.

Abd. Bahrim, Y. (2024). *Digital Logic 5th Edition*. Johor Bahru, Johor: Faculty of Computing, University of Technology Malaysia.

Tocci, Ronald J., Widmer, Neal S. and Moss, Gregory L. (2017). *Digital Systems: Principles and Applications 12th Edition*. Boston: Pearson Education.

Floyd, T. L. (2015). *Digital Fundamentals 11th Edition*. Boston: Pearson Education.

APPENDICES

Name		Parthiv Gunalan	Ida Yatullailiyeh Binti Amrun	Heng Zhi Qiang	Gavenesh A/L Batumalai
Tasks	Designing	Full Circuit Diagram using Deeds Simulator			
				Cover Page	
		Extra Features for Circuit		Presentation Slides	
		Block Diagram	State Diagram	T Flip-flop Excitation Table	Karnaugh-Map
		Script Preparation & Video Editing			
	Report Writing	Problem Statement& Suggested Solution	Background	System Implementation	
		Conclusion & Reflection			Acknowledgement & Dedication
		References	Requirements	Final Editing	