ECE150 Project 3:

A Mini Game

Dan Brody, Jonghyeok Kim

The Cooper Union for the Advancement of Science and Art

ECE-150-1
Digital Logic Design

Professor Lisa Shay

14 May 2021

Contents

Section 1: Introduction

For this assignment, the task was to create a minigame project utilizing wires, breadboards, and chips found at our CU@Home kit. The specific game that was assigned was an LED collision game. The aim was to move the LEDS in the LED matrix, over a 4x5 grid, left to right by user input or down every clock pulse to collide with a high LED in the target row, a row of 4 LEDs that should only update on the first row of the LED matrix lighting up. An example game can be shown in Fig.1.

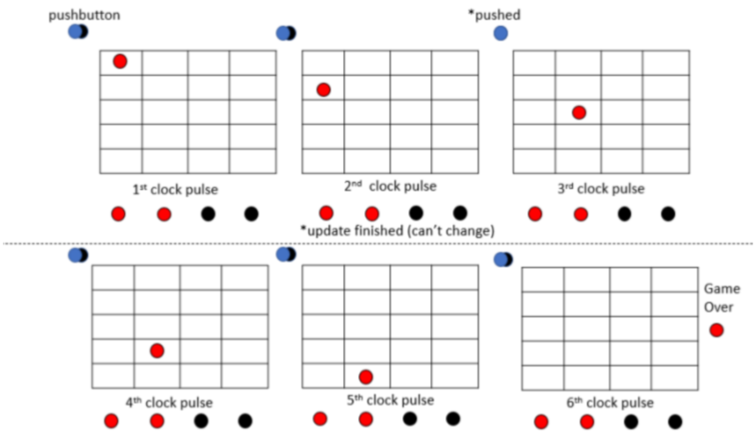


Fig.1. Simulation of Minigame Project Over 6 Clock Pulses

Regarding the clock pulses, the pulses were created by what is known as 555 timers. A 555 timer (in astable) possesses what is known as period and duty cycle. In a 555 timer there is a certain pattern that iterates every time the period is over. That pattern is what is known as the duty cycle. In a period only a certain percent of the period will be high and the other percent will be low. The percent of the period that remains high is the duty cycle. In this way users can use the 555 timer to go high or low after a certain duration. In order to control the period and duty cycle we can edit the values for R1, R2, and C that we use for the chip as in Fig. 2.

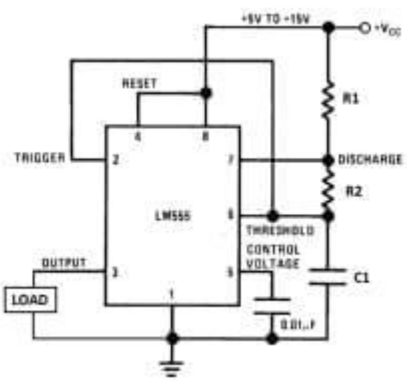


Fig. 2. Pinout for Astable 555 Timer

The equations related to the duty cycle and period are as follows:

1. $t_{HIGH} = 0.693*C*(R1+R2)$
2. $t_{LOW} = 0.693*C*R2$
3. Period = tHIGH + tLOW
4. Duty Cycle = t_{HIGH} /(t_{HIGH} + t_{LOW}) = (R1+R2)/(R1+2*R2)

Using these equations once can now change the duty cycle. Since R1 is the only term different from (1) and (2) we can conclude that increasing R1 will increase t_{HIGH} and therefore the duty cycle. Additionally, this will increase the period such that the LED will light slower. The reason why using 555 timers were essential to this project is because there need to be two different 555 timers for the down functionality and updating the target row since these two functions operate at different periods (target row being faster than down functionality). Some more recommended devices for the project include counters, multiplexers, and demultiplexers. Counters do just as their name suggests, count. In our circuit we used the CD4029 as shown in Fig. 3. Counters count up to a certain number in binary before they move back down to 0. They are activated by clock pulses such that the counter only counts once per clock pulse based on the edge trigger. A function that we used in the project is a pin known as PRESET ENABLE. When this pin is set high, the counter starts over from the numbers on the jam inputs. for example, if we set jam 1-4 low and set PRESET ENABLE high on the chip in Fig. 3. the chip's count would start over from 0.

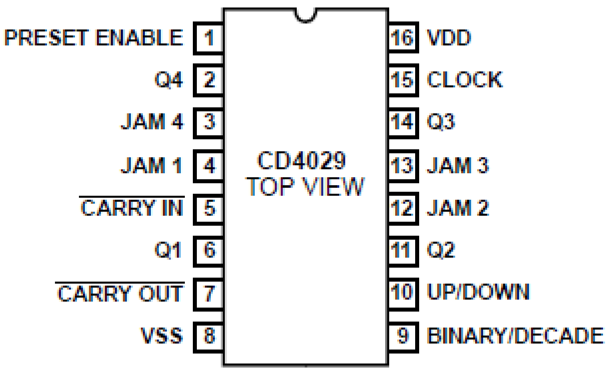


Fig. 3. Pinout for CD4029 Counter

Concerning the demultiplexer (demux) and multiplexer (mux) these two chips go hand in hand. The demux takes one input, as a number in binary, and outputs the output from the corresponding pin. For example, if we put as input to the demux 010 then the output at pin 2 is sent. The mux, on the other hand, is the opposite selecting inputs as opposed to outputs. If we, for example, had 8 inputs and 010 was sent to the data selector then only the input at pin 2 would be sent to the output. Over the course of this project there were two implementations, an upgraded packet and base packet. The upgraded packet includes collision detection to call a game over while the base packet does not. For this project both the base packet and upgraded packet were created.

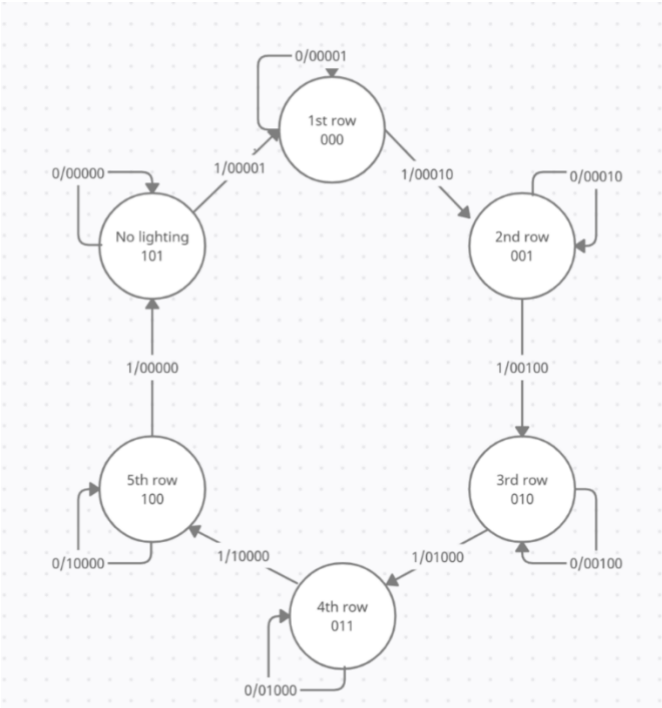
Section 2: Theory

Steps Taken

First, state diagrams for down functionality, step functionality, and generating target LEDs were designed. Up counter CD4029BE and demultiplexer CD4051BE were used to have down functionality on the LED matrix; two D flip-flops of CD4013BE were used to have step functionality; up counter CD4029BE was used to generate target LEDs when LEDs of the first row light up. Then, truth tables were made for each of the functions. Karnaugh map was only needed to be designed and solved for the step functionality since CD4029BE was always set to count up except for the reset. With the use of Logisim evolution, a simulation for the base packet was designed first, and an upgraded packet was then designed based on it. After the successful simulations, Dan Brody built the hardware circuit for the base packet, and Jonghyeok Kim built the hardware circuit for the upgraded circuit. Datasheets for the chips were achieved from Ti.com. Once the designed circuits function as intended, the bill of materials used in the circuit was estimated by finding the price of each part used in the project.

State Diagram

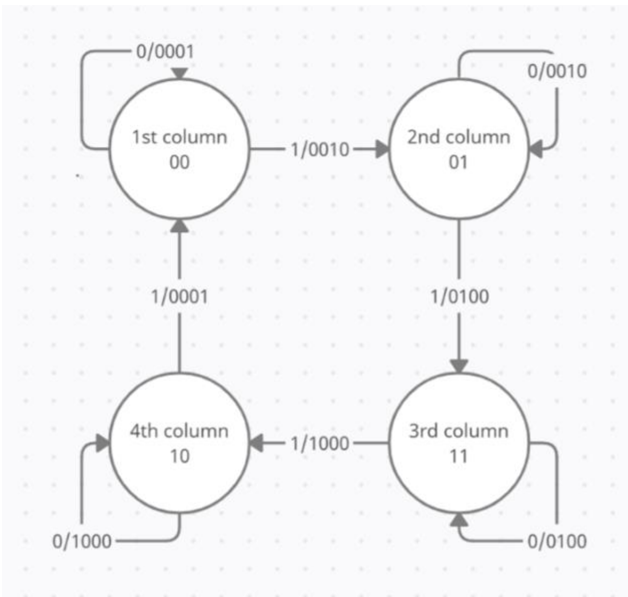
For Down Functionality (CD4029BE + Demultiplexer CD4051BE)



*Outputs of CD4029BE become inputs of the demultiplexer.

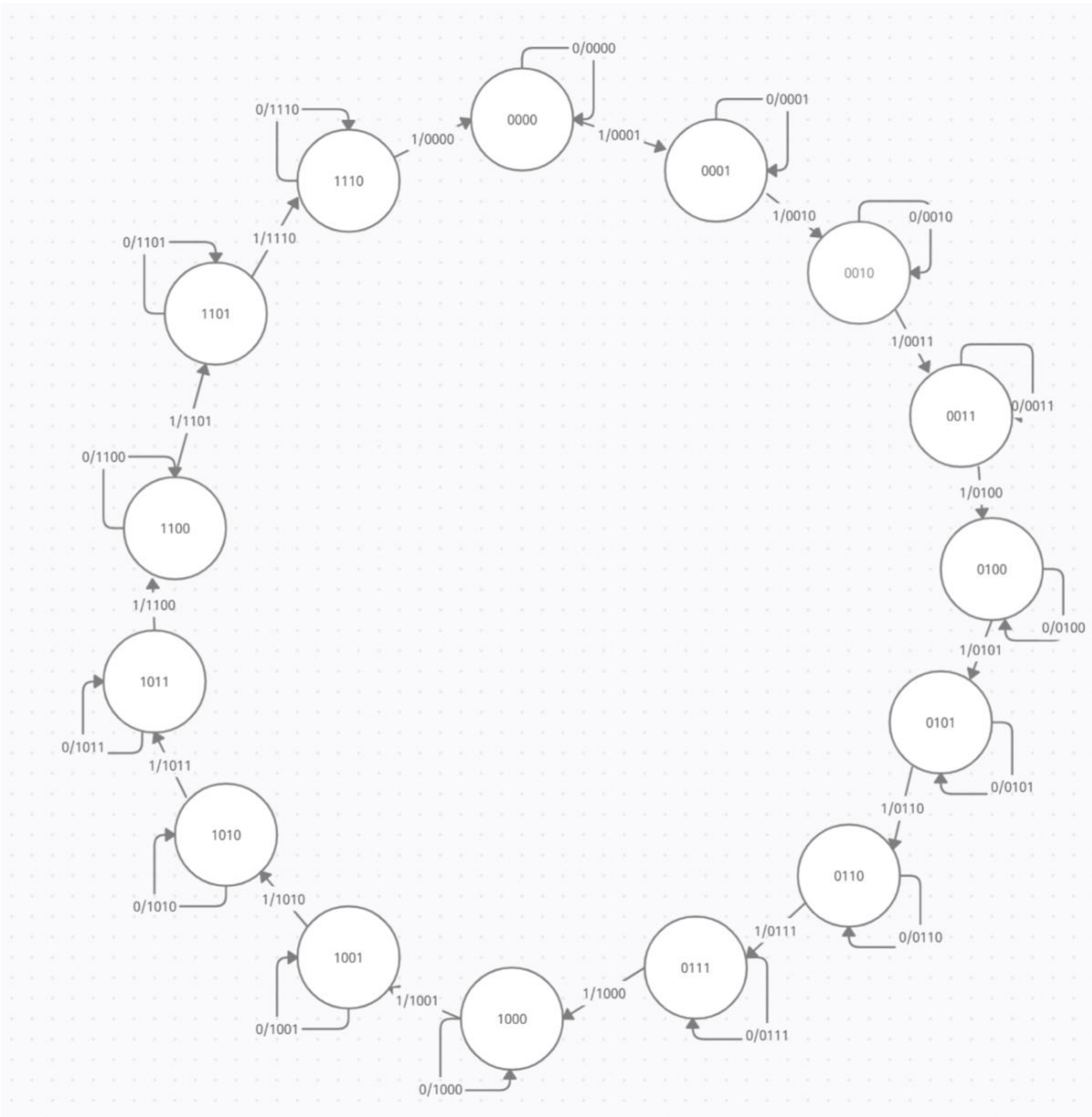
*Y_4, Y_3, Y_2, Y_1, Y_0 are the outputs of the demultiplexer and connected to the Row 5, 4, 3, 2, 1 of the LED Matrix.

For Step Functionality (CD4013BE)



*Outputs (Y_8, Y_7, Y_6, and Y_5) are Column 4, 3, 2, 1 of the LED Matrix.

<u>For Generating Target LEDs (CD4029BE)</u>



*Outputs (Y_{12}, Y_{11}, Y_{10}, and Y_9) are four target LEDs.

Truth Table

For Down Functionality (CD4029BE + Demultiplexer CD4051BE)

Input	Current State			Next State			Outputs				
X_0	Q2	Q1	Q0	Q'(2)	Q'(1)	Q'(0)	Y_4	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	0	0	0	0	0	0	1
1	0	0	0	0	0	1	0	0	0	1	0
0	0	0	1	0	0	1	0	0	0	1	0
1	0	0	1	0	1	1	0	0	1	0	0
0	0	1	1	0	1	1	0	0	1	0	0
1	0	1	1	0	1	0	0	1	0	0	0
0	0	1	0	0	1	0	0	1	0	0	0
1	0	1	0	1	1	0	1	0	0	0	0
0	1	1	0	1	1	0	1	0	0	0	0
1	1	1	0	1	0	0	0	0	0	0	0
0	1	0	0	1	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0	0	1

*Outputs of CD4029BE become inputs of the demultiplexer.

*Y_4, Y_3, Y_2, Y_1, Y_0 are the outputs of the demultiplexer and connected to the Row 5, 4, 3, 2, 1 of the LED Matrix.

*Input X_0 receives the signal from 555 Timer built for down functionality.

For Step Functionality (CD4013BE)

2 D Flip-flops were used to step the columns rightward.

Input	Current State		Next State		Outputs			
X_1	Q4	Q3	Q'(4)	Q'(3)	Y_8	Y_7	Y_6	Y_5
0	0	0	0	0	0	0	0	1
1	0	0	0	1	0	0	1	0
0	0	1	0	1	0	0	1	0
1	0	1	1	1	0	1	0	0
0	1	1	1	0	0	1	0	0
1	1	1	1	0	1	0	0	0
0	1	0	1	0	1	0	0	0
1	1	0	0	0	0	0	0	1

*Y_8, Y_7, Y_6, and Y_5 are Column 4, 3, 2, 1 of the LED Matrix.

*Input X_1 receives the signal from the push button switch.

For Generating Target LEDs (CD4029BE)

Input	Current State				Next State				Outputs			
X_2	Q8	Q7	Q6	Q5	Q'(8)	Q'(7)	Q'(6)	Q'(5)	Y_{12}	Y_{11}	Y_{10}	Y_9
0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	1	0	0	0	1	0	0	0	1
1	0	0	0	1	0	0	1	0	0	0	1	0
0	0	0	1	0	0	0	1	0	0	0	1	0
1	0	0	1	0	0	0	1	1	0	0	1	1
0	0	0	1	1	0	0	1	1	0	0	1	1
1	0	0	1	1	0	1	0	0	0	1	0	0
0	0	1	0	0	0	1	0	0	0	1	0	0
1	0	1	0	0	0	1	0	1	0	1	0	1
0	0	1	0	1	0	1	0	1	0	1	0	1
1	0	1	0	1	0	1	1	0	0	1	1	0
0	0	1	1	0	0	1	1	0	0	1	1	0
1	0	1	1	0	0	1	1	1	0	1	1	1
0	0	1	1	1	0	1	1	1	0	1	1	1
1	0	1	1	1	1	0	0	0	1	0	0	0
0	1	0	0	0	1	0	0	0	1	0	0	0
1	1	0	0	0	1	0	0	1	1	0	0	1
0	1	0	0	1	1	0	0	1	1	0	0	1
1	1	0	0	1	1	0	1	0	1	0	1	0
0	1	0	1	0	1	0	1	0	1	0	1	0
1	1	0	1	0	1	0	1	1	1	0	1	1
0	1	0	1	1	1	0	1	1	1	0	1	1
1	1	0	1	1	1	1	0	0	1	1	0	0
0	1	1	0	0	1	1	0	0	1	1	0	0
1	1	1	0	0	1	1	0	1	1	1	0	1
0	1	1	0	1	1	1	0	1	1	1	0	1
1	1	1	0	1	1	1	1	0	1	1	1	0
0	1	1	1	0	1	1	1	0	1	1	1	0
1	1	1	1	0	0	0	0	0	0	0	0	0

*Y_{12}, Y_{11}, Y_{10}, and Y_9 are four target LEDs.

*Input X_2 receives the signal from the 555 timer built to generate target LEDs.

Karnaugh Maps

*The use of CD4029BE does not need generation of Karnaugh maps.

For Step Functionality (CD4013BE)

Q4 | Q3	0	1
0	1	1
1	0	0

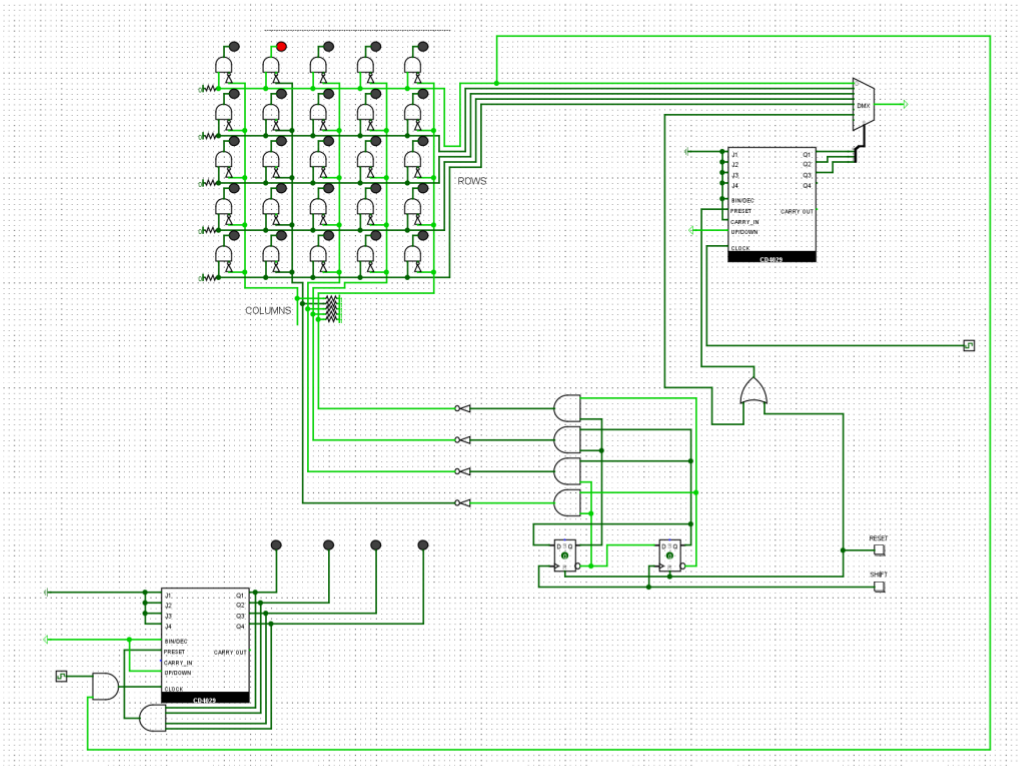
Q'(3) = $\overline{Q4}$

Q4 | Q3	0	1
0	0	1
1	0	1

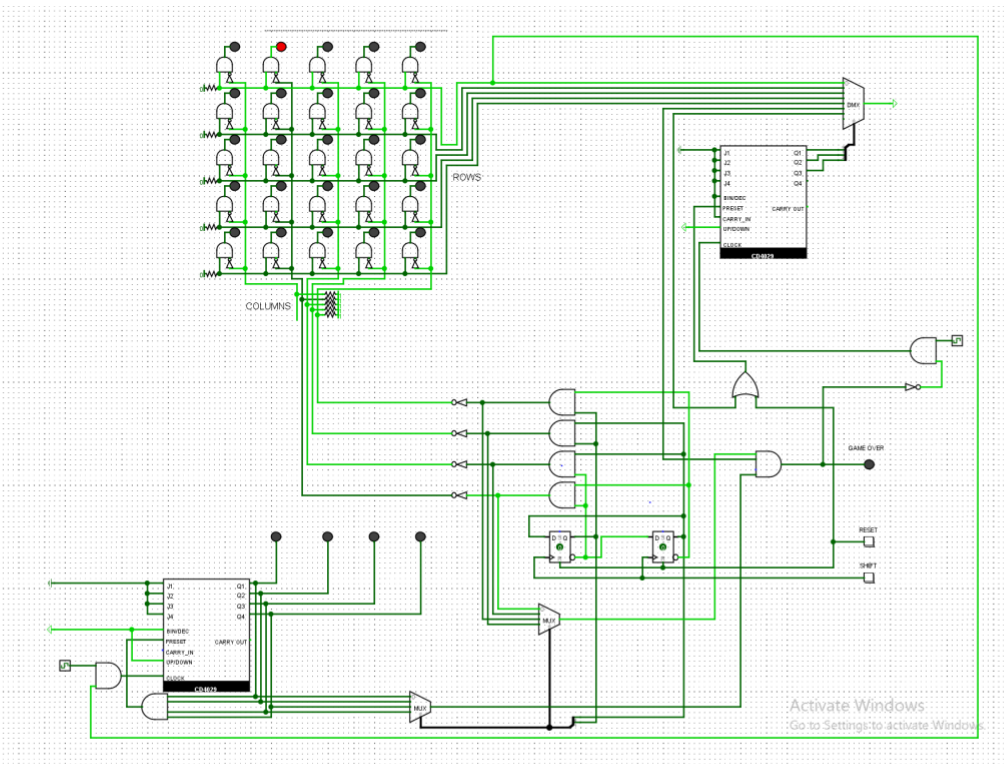
Q'(4) = Q3

Circuit Diagrams

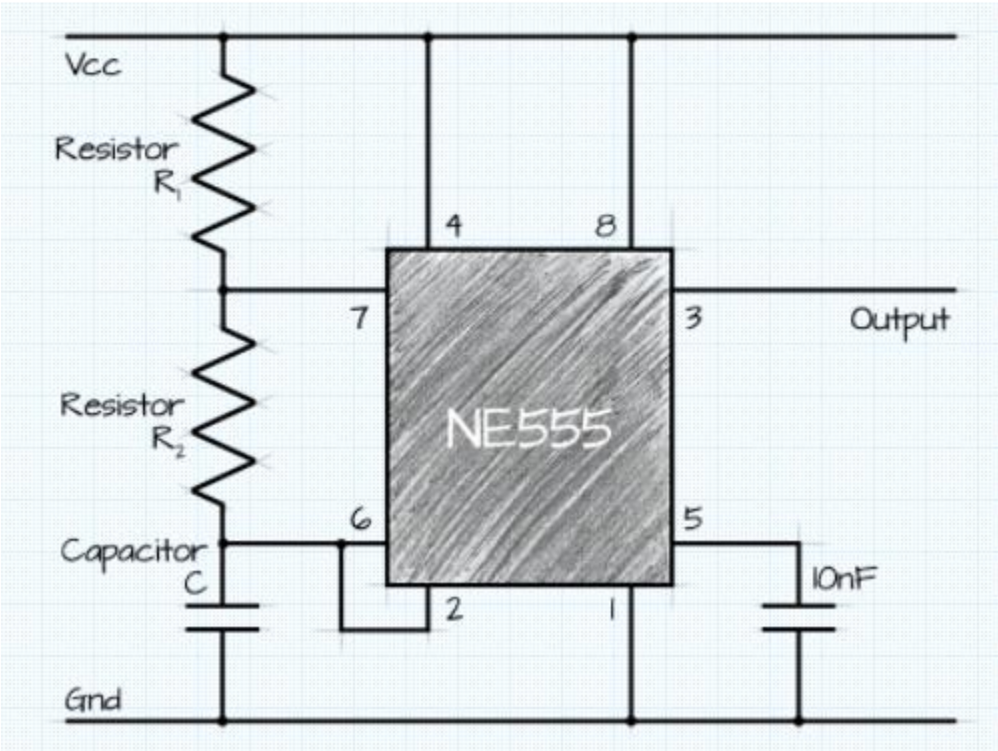
Brody's Base Packet



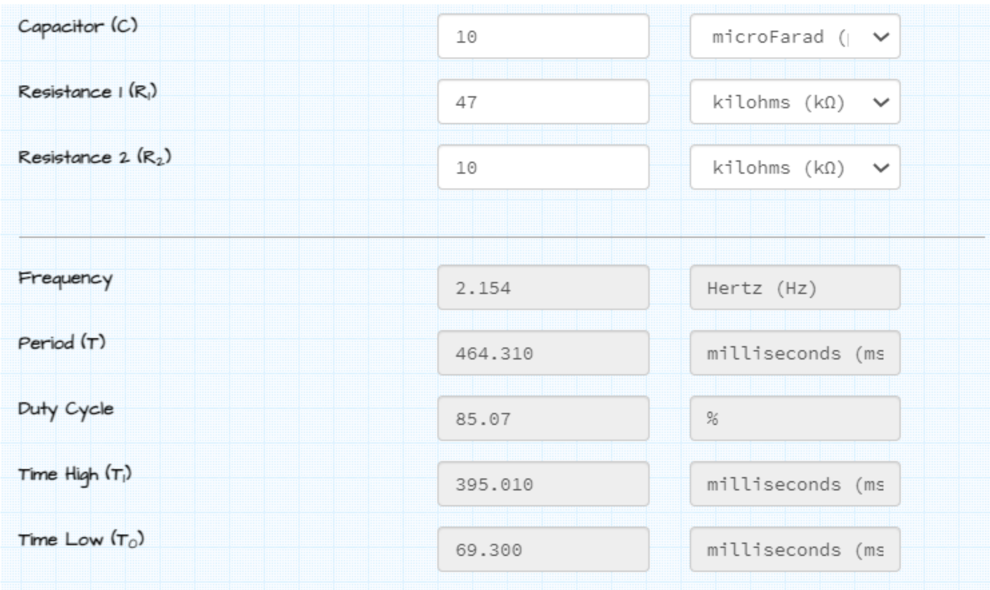
Kim's Updated Packet



555 Timers Set Up



Brody’s 555 Timer for Down Functionality



Capacitor (C)	10	microFarad (µ
Resistance 1 (R_1)	47	kilohms (kΩ)
Resistance 2 (R_2)	10	kilohms (kΩ)
Frequency	2.154	Hertz (Hz)
Period (T)	464.310	milliseconds (ms
Duty Cycle	85.07	%
Time High (T_1)	395.010	milliseconds (ms
Time Low (T_0)	69.300	milliseconds (ms

Kim's 555 Timer for Down Functionality

Capacitor (C)	10	microFarad (µ
Resistance 1 (R_1)	47	kilohms (kΩ)
Resistance 2 (R_2)	47	kilohms (kΩ)
Frequency	1.023	Hertz (Hz)
Period (T)	977.130	milliseconds (ms
Duty Cycle	66.67	%
Time High (T_1)	651.420	milliseconds (ms
Time Low (T_0)	325.710	milliseconds (ms

Kim's and Brody's 555 Timer Used to Generate Target LEDs

Capacitor (C)	10	microFarad (µ
Resistance 1 (R_1)	10	kilohms (kΩ)
Resistance 2 (R_2)	330	ohms (Ω)
Frequency	13.537	Hertz (Hz)
Period (T)	73.874	milliseconds (ms
Duty Cycle	96.90	%
Time High (T_1)	71.587	milliseconds (ms
Time Low (T_0)	2.287	milliseconds (ms

Section 3: Bill of Materials

Name	Quantity	Price ($)	Quantity Used (Dan)*	Quantity Used (Burt)**
CD4013	1	0.60 ("CD4013BE Texas Instruments: Mouser.")	1	1
CD4081	1	0.60 ("CD4081BE 4 Texas Instruments: Mouser.")	2	2
CD4069	1	0.61 ("CD4069UBE Texas Instruments: Mouser.")	1	1
CD4029	1	0.61 ("CD4029BE Texas Instruments: Mouser.")	2	2
CD4082	1	0.61 ("CD4082BE Texas Instruments: Mouser.")	1	1
CD4051	1	0.61 ("CD4051BE Texas Instruments: Mouser.")	1	3
CD4071	1	0.60 ("CD4071BE Texas Instruments: Mouser.")	1	1
555 Timer	1	1.31 ("LM555CN/ NOPB Texas	2	2

		Instruments: Mouser.”)		
LED Matrix	1	1.66	1	1
Push Button	1	3.19 (“TPA11CGP C6 TE Connectivity / Alcoswitch: Mouser.”)	2	2

*implemented base packet
** implemented upgraded packet

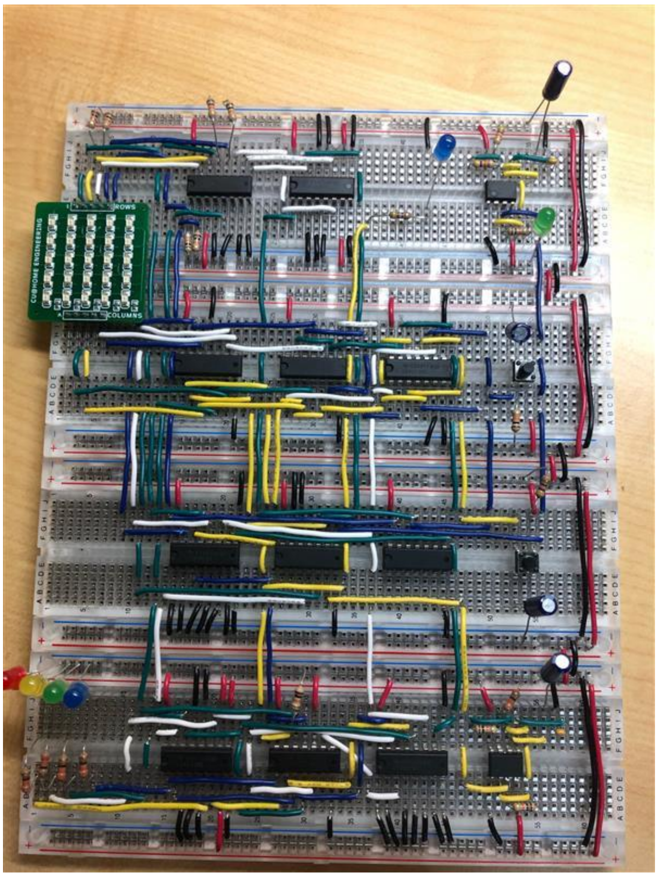
Total Cost of Each Implementation :

Dan’s = (1)($0.60) + (2)($0.60) + (1)($0.61)+ (2)($0.61)+ (1)($0.61) + (1)($0.61) + (1)($0.60) + (2)($1.31)+$1.66 + (2)($3.19) = $16.11

Burt’s = (1)($0.60) + (2)($0.60) + (1)($0.61)+ (2)($0.61)+ (1)($0.61) + (3)($0.61) + (1)($0.60) + (2)($1.31)+$1.66 + (2)($3.19) = $17.33

Section 4: Results

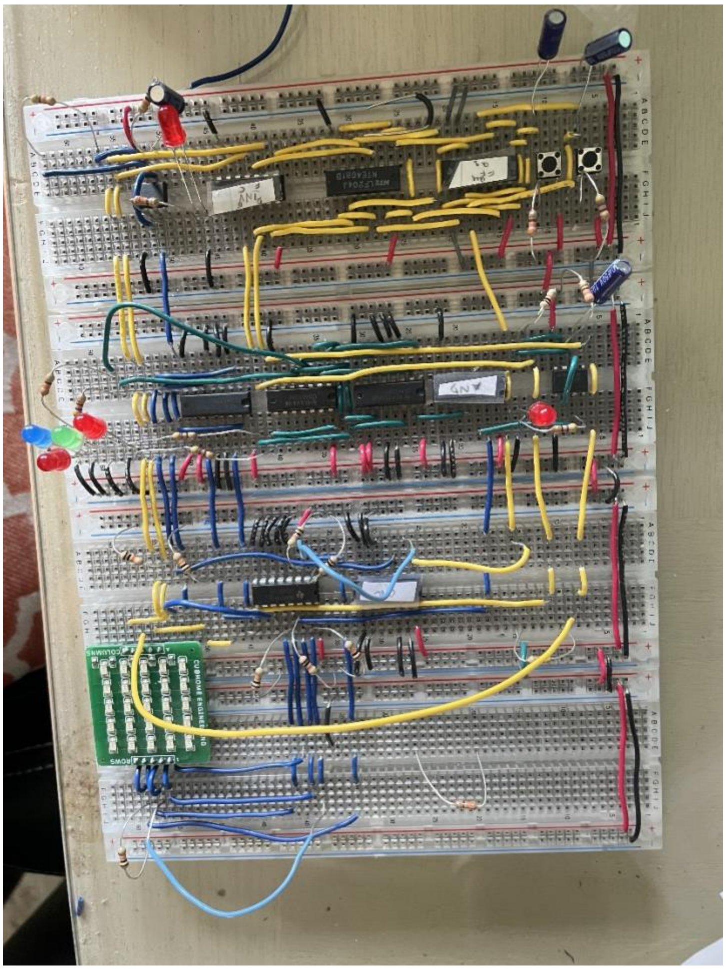
A Picture of Kim's Breadboard



Comparison Between Theoretical and Empirical Results

The empirical results obtained from the built circuit exactly matched with theoretical results achieved from the Logisim simulation. In other words, the state tables for the actual circuit and the simulation were identical. Thus, the truth table for the actual hardware can be seen on the truth table page in Section 2.

A Picture of Dan's Breadboard



*reset button on top right
**left to right push button next to reset

Comparison Between Theoretical and Empirical Results

The empirical results obtained from the built circuit almost exactly matched with theoretical results achieved from the Logisim simulation. In other words, the state tables for the actual circuit and the simulation were almost identical. Unfortunately, there was one minor glitch in the push button moving the LED from left to right. Whenever the flip flop was in 00 state the state was skipped if one released the push button. Upon changing resistors connected to the push button in question we found that this was not a switch debouncing issue but a flip flop issue.

Section 5: Conclusions

Personal Statement

Jonghyeok Kim: Unlike previous projects, this project was building a game that the user can control. I have always wanted to build a game like this and am glad that I built the upgraded packet in a brief time. While I spent lots of time on 1st and 2nd projects, I spent a relatively small amount of time on this project. After building hardware, I had troubleshot because of the floating issues and debouncing switches. Fortunately, the issues were solved within few hours with the use of a digital multimeter. I think this project was easier than I expected it to be.

Dan Brody: Overall this project was fun because, unlike other projects, we were not given specific instructions on how to set up our functionality, left open to choose any logic that suited the project. Given this creative freedom, we were able to make some innovative designs. A big challenge in this project, however, was troubleshooting. With the introduction of push buttons so came switch debouncing, changing multiple states at one push of the button. I learned a lot from this project including how to solve switch debouncing and how to better think about solutions. While my implementation only had one small glitch the project was successful.

Project Conclusion

This project required the understanding of concepts of debouncing switches and floating input/outputs. Overall, this project was a valuable experience of learning how to use multiplexer/demultiplexer and button press switch. It was tough working on a group project in hardware remotely, but we overcame that issue.

Section 6: References

"CD4013BE Texas Instruments: Mouser." *Mouser Electronics*, Texas Instruments, www.mouser.com/ProductDetail/Texas-Instruments/CD4018BE?qs=%2Fha2pyFadugnwXjopnZ5OIw6oQ4YpHOebRxJ%252B21A5TY%3D.

"CD4029BE Texas Instruments: Mouser." *Mouser Electronics*, Texas Instruments, www.mouser.com/ProductDetail/Texas-Instruments/CD4029BE?qs=q2XTDbzbm6C3NEJn1CP1SA%3D%3D.

"CD4051BE Texas Instruments: Mouser." *Mouser Electronics*, Texas Instruments, www.mouser.com/ProductDetail/Texas-Instruments/CD4051BE?qs=q2XTDbzbm6DxulBsMcV7tA%3D%3D.

"CD4069UBE Texas Instruments: Mouser." *Mouser Electronics*, Texas Instruments, www.mouser.com/ProductDetail/Texas-Instruments/CD4069UBE?qs=gqbMQSs93zN4MVMbMFTI6g%3D%3D.

"CD4071BE Texas Instruments: Mouser." *Mouser Electronics*, Texas Instruments, www.mouser.com/ProductDetail/Texas-Instruments/CD4071BE?qs=D5pVkbrsqqJag2QszZHFMA%3D%3D.

"CD4081BE Texas Instruments: Mouser." *Mouser Electronics*, Texas Instruments, www.mouser.com/ProductDetail/Texas-Instruments/CD4018BE?qs=%2Fha2pyFadugnwXjopnZ5OIw6oQ4YpHOebRxJ%252B21A5TY%3D.

"CD4082BE Texas Instruments: Mouser." *Mouser Electronics*, Texas Instruments, www.mouser.com/ProductDetail/Texas-Instruments/CD4082BE?qs=KaAwwOlwapuurJwzCPQBUQ%3D%3D.

"LM555CN/NOPB Texas Instruments: Mouser." *Mouser Electronics*, Texas Instruments, www.mouser.com/ProductDetail/Texas-Instruments/LM555CN-NOPB?qs=QbsRYf82W3GjZpQL%2FakZOg%3D%3D.

"TPA11CGPC6 TE Connectivity / Alcoswitch: Mouser." *Mouser Electronics*, Texas Instruments, www.mouser.com/ProductDetail/TE-Connectivity-Alcoswitch/TPA11CGPC6?qs=kwkLQ26gqZBf8HC9paKrrg%3D%3D.

555 Astable Circuit Calculator. 23 Apr. 2021, https://ohmslawcalculator.com/555-astable-calculator.