Uneven Seven-Sided Dice Roller

1st Mehmet Fatih Göğüş
Electrıcal-Electronics Engineering
Abdullah Gul University
Kayseri, Turkey
mehmetfatih.gogus@agu.edu.tr

2nd Davut Uysal
Electrıcal-Electronics Engineering
Abdullah Gul University
Kayseri, Turkey
davut.uysal@agu.edu.tr

ABSTRACT

This document outlines the design, simulation, and implementation of an uneven seven-sided dice roller system. The project integrates a 1-bit data generator, a CMOS-configured 1-bit full adder, two registers (intermediate and output), and a control system that manipulates a 400 Hz clock signal to produce a derived clock of 100 Hz. This system generates a three-bit number representing values from 1 to 7, randomized by the dynamics of digital signal processing and storage. Key challenges addressed include the design of a stable multivibrator for the data generator, the manipulation of binary addition with a constant carry, and the effective distribution of data signals to synchronize with computational needs.

INTRODUCTION

In the field of digital system design, the development of mechanisms that generate random outcomes is pivotal for applications ranging from gaming to secure communications. This project focuses on constructing an electronic system capable of producing a seven-sided dice roll, a novel approach that extends beyond traditional binary output systems. The design challenges traditional methods by implementing a system that not only generates random numbers but also manipulates clock signals and utilizes CMOS technology for critical computations. This system's innovation lies in its ability to consistently produce random and unbiased results through a complex interplay of electronic components and logic circuits.

DESCRIPTION OF THE ENTIRE SYSTEM

Thus, the uneven 7-sided dice roller system has been laid out as given below. It will come with a bunch of critical components that will work in a symbiotic relationship toward the development of the desired functionality.

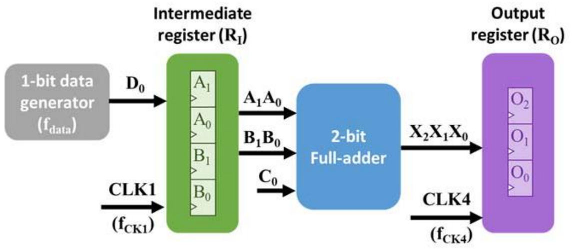


Fig. 1. Schematic diagram of an uneven seven-sided dice roller.

A. DATA (D0) GENERATOR

The pulse generator that is ensuring the randomness of the outputs is using a multivibrator circuit. It is designed to work outside the synchronization with the main clock so that the randomness is intrinsic in the signal being generated by it

CLOCK SIGNAL MANIPULATION

The circuitry designed manipulates the first clock (CLK1) (fck1) rate of 400 Hz to derive CLK4(fck4) at 100 Hz. This derived clock signal, thereafter, refreshes the output register at the frequency required to update the dice roll output accordingly. Configuration: The 2- bit full adder configuration used for processing the inputs of two registers is the CMOS configured 1-bit full adder and one more 1-bit full adder designed with full adder IC.

B. FULL ADDER CONFIGURATION

The computational logic within the system employs two distinct full adder configurations, each tailored to specific roles in the arithmetic process required for the dice roller operation. The first full adder, configured in CMOS technology, handles the least significant bits, A0 and B0. This adder operates with a carry input (Cin) permanently set to '1', simplifying the Boolean logic and reducing circuit complexity by minimizing the gate count and optimizing performance.

The second part of the arithmetic logic uses a standard full adder IC to process the more significant bits, A1 and B1. The carry output (C0) from the first CMOS-configured adder serves as the carry input for this second adder. This sequential carry propagation ensures accurate binary addition across both bits of the system, maintaining integrity and consistency in the computational process.

This dual-setup, with a CMOS full adder for initial bit addition and a traditional full adder IC for subsequent bit processing, provides a robust system that enhances both the precision and efficiency of the dice roller's electronic design. This configuration effectively leverages the strengths of both CMOS and standard IC technologies to achieve optimal results in the dice output generation.

C. *REGISTERS*

Two important registers pertaining to the system are Full adder circuit reduces the number of basic gates required to build the adder. It processes the two bits of input.

Intermediate Register (RI): temporarily stores the bits of data (D0) in timing order with respect to CLK1. Data appears at the outputs of the intermediate register at A1, A0, B1, and B0. This process is orchestrated so that D0 is delivered to A1 on the first clock pulse, to A0 on the second pulse, and so forth, with 'no change' (NC) occurring in the intervening periods.

Output Register (RO): Stores the final output, the binary equivalent of the dice, refreshed at the rate determined by CLK4.

DATA GENERATOR (D0)

The pulse generator, designed as a essential part of the dice roller system, produces a square wave output required for the generation of random data (D0). Multivibrator circuit is used to generate D0.

It is designed in such a way to avoid pulses incurring synchronization errors with high frequencies and, on the other hand, not producing too low frequencies that affect the quality of randomness.

A multivibrator circuit, Figure 1.1, has been designed to provide pulse signals for Data (D0). The circuit is designed to produce a proper square wave with a low voltage level of ground and a high voltage level of 5 volts. The circuit in Figure 1.0 was designed with the help of 2 1k resistors, 2 82k resistors, 2 1 nanometer capacitors and 2 2N2222 model transistors. In the produced circuit, care was taken to ensure that the frequency complies with the limits of

10fck1 < f_{data}<100fck1

A. *Circuit Diagram and Performance*

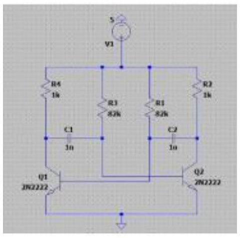


Figure 1.1

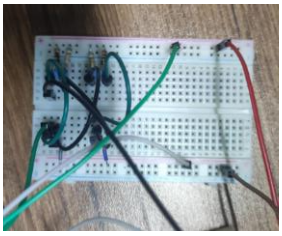


Figure 1.2

Figure 1.2 is the photograph of physical Data Generator (D0) multivibrator circuit built with transistors, capacitors and resistors.

$$The\ period\ of\ each\ half\ of\ the\ multivibrator\ is\ therefore\ given\ by\ t\ = \ln(2)\,RC$$

$$The\ total\ period\ of\ oscillation\ is\ given\ by:$$

$$T\ =\ t1\ +\ t2\ = \ln(2)\,R2\,C1\ + \ln(2)\,R3\,C2$$

$$f\ = \frac{1}{T} = \frac{1}{\ln(2) *\ (R2\,C1\ +\ R3\,C2)}$$

$$= \frac{1}{0.693\ *\ (R2\,C1\ +\ R3\,C2)}$$

For the special case where

- t1 = t2 (50% duty cycle)
- R2 = R3
- C1 = C2

$$f\ = \frac{1}{T} = \frac{1}{\ln(2) * RC * 2} = \frac{0.72}{RC}$$

$$fdata\ \ = \frac{1}{T} = \frac{1}{\ln(2) *\ (R2\,C1\ +\ R3\,C2)}$$

$$fdata\ =\ 8.798{,}78\ Hz\ =\ 8.7kHz$$

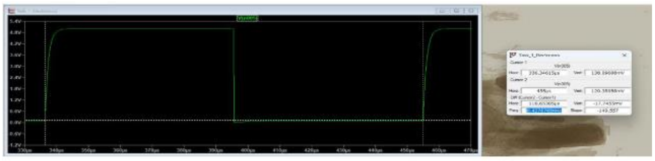


Figure 1.3

According to the simulation result in Figure 1.3, the f_{data} was found to be 8.4278768KHz.

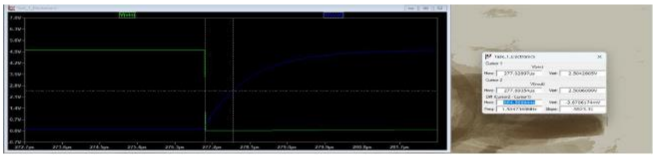


Figure 1.4

According to the simulation result in Figure 1.4, the t_{PHL} was found to be 664.56894ns

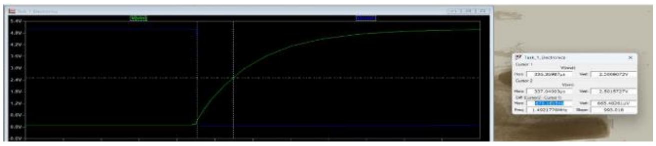


Figure 1.5

According to the simulation result in Figure 1.5, the t_{PLH} was found to be 670.1615ns.

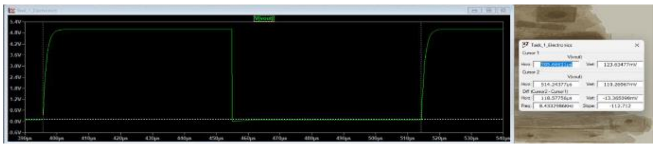
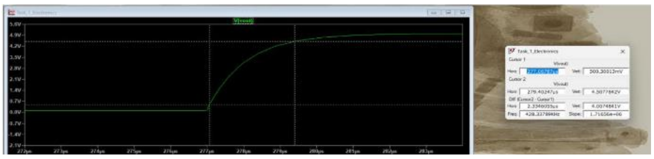


Figure 1.6

According to the simulation result in Figure 1.6, the t_{cycle} was found to be 118.57756µs.



Task_1_Electronics

Cursor 1

V(vout)

Horz: [illegible] Vert: 500.30013mV

Cursor 2

V(vout)

Horz: 279.40247µs Vert: 4.5077842V

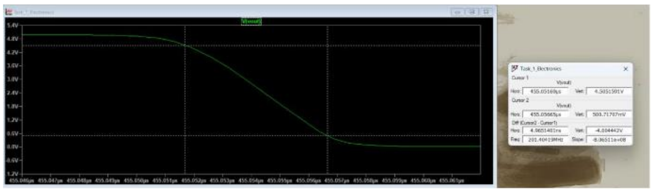
Diff (Cursor2 - Cursor1)

Horz: 2.3346055µs Vert: 4.0074841V

Freq: 428.33789kHz Slope: 1.71656e+06

Figure 1.7

According to the simulation result in Figure 1.7, the t_{rising} was found to be 2.3346055µs.



Task_1_Electronics

Cursor 1

V(vout)

Horz: 455.05160µs Vert: 4.5051501V

Cursor 2

V(vout)

Horz: 455.05665µs Vert: 500.71707mV

Diff (Cursor2 - Cursor1)

Horz: 4.9651401ns Vert: -4.004442V

Freq: 201.40419MHz Slope: -8.06511e+08

Figure 1.8

According to the simulation result in Figure 1.8, the t_{rising} was found to be 4.9651401ns.

Parameter	**Time**
t_{PHL}	664.56894 ns
t_{PLH}	670.1615 ns
t_{cycle}	118.57756 µs
t_{rising}	2.3346055 µs
$t_{falling}$	4.9651401 ns

Figure 1.9

Figure 1.9 shows the time results above.

While the manual calculation of f_{data} is 8,798.78 Hz, the f_{data} value obtained as a result of the simulation is 8,427.8768 Hz. As a result of the calculations, it was determined that there was an error of 4.215 percent. Also transition time is 1.97 percent of the cycle time. Transition time has been found to be less than 5 percent of the cycle time.

Voltage Levels: The circuit operates with a high of 5V and a low of ground, thus ensuring that transitions between signals are clear and stable.

Frequency Range: That means between 10 to 100 times the frequency of CLK1, the frequency of the data pulses meets the specification of random data generation.

B. Conclusion

The data generator contains, at its basic level, the anchor of a well-designed multivibrator circuit and serves as the cornerstone of randomness in the dice-rolling system. The data generator should be independent of the main clock and produce unsynchronized pulses, the most basic prerequisite for assuring the unpredictability of the data. This will provide a square wave in the required frequency range, making sure that the randomness is something which actually comes with the system and that the output is truly something stochastic by careful design and proper implementation of this component. Its performance serves to solidify the capability of systems to function as a random number generator, which is critical for project success.

DATA DISTRIBUTION (D0 TO A AND B)

INTERMEDIATE REGISTER(RI)

The data distribution subsystem is intricately designed to ensure precise control and timing of the D0 signal distribution to the inputs of the full adder, namely A1, A0, B1, and B0. To achieve this, the subsystem incorporates a counter that uses the CLK1 signal to generate two selector signals, S1 and S0, which dictate the specific timing of changes at the rising edges of CLK1.

The intermediate register receives the signal D0 at the incoming edge of CLK1. The data presence occurs at the outputs of the intermediate register A1, A0, B1, B0. This process is synchronized such that D0 is given to A1 during the first clock pulse, A0 during the second pulse, etc., with 'no change' (NC) states in between as seen in Figure 2.1.

CLK1	1st ↑	2nd ↑	3rd ↑	4th ↑	5th ↑	…
A_1	D_0	NC	NC	NC	D_0	…
A_0	NC	D_0	NC	NC	NC	…
B_1	NC	NC	D_0	NC	NC	…
B_0	NC	NC	NC	D_0	NC	…

Note. NC: no change

Figure 2.1

A. System Design And Implementation

Counter Implementation

The counter is configured to cycle through a sequence that generates S1 and S0. These selector signals are crucial for determining which of the register outputs (A1, A0, B1, B0) will receive the D0 signal at any given CLK1 cycle.

The counter's output, S1 and S0, effectively creates a controlled sequence that ensures each bit receives the D0 signal at the correct phase of the CLK1, facilitating accurate and synchronized data distribution.

Signal Routing:

Based on the outputs from the counter, a logic circuit routes the D0 signal to the appropriate register inputs. This routing logic is designed to change the state of A0, A1, B0, and B1 only at the rising edge of CLK1, aligning with the counter's output sequence.

B. Circuit Diagram and Timing Analysis

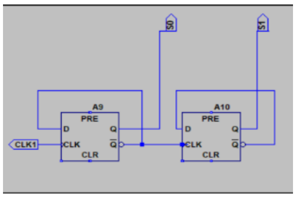


Figure 2.2

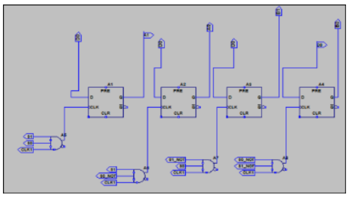


Figure 2.3

Figure 2.2: Displays the circuit diagram for the counter and the associated routing logic that controls the distribution of D0.

Figure 2.3 (Intermediate Register (RI)): A detailed schematic showing how the D0 signal is selectively routed to A0, A1, B0, and B1 based on the counter’s, Figure 2.2 (Counter), output.

Timing Diagram:

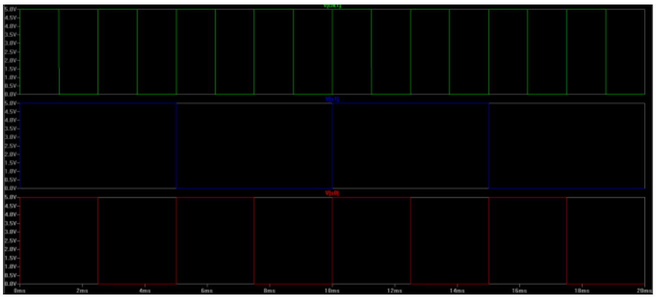


Figure 2.4

Figure 2.4: Presents a timing diagram illustrating the sequence of S1 and S0.

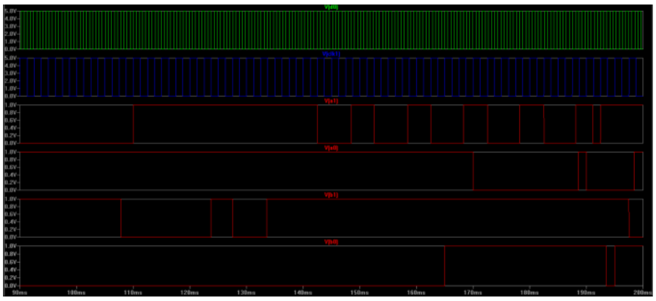


Figure 2.5

Figure 2.5: illustrates how the distribution of D0 to the full adder's inputs is. This diagram clearly shows the synchronization between CLK1, the counter outputs (S1 and S0), and the resulting changes in A1, A0, B1, and B0.

C. *Performance And Verification*

Simulation Results: These results confirm the effectiveness of the counter and routing logic in distributing the D0 signal. The simulation demonstrates that each input receives the D0 pulse accurately at the designated CLK1 rising edge, verifying the precision of the timing mechanism.

Practical Implementation: The implemented circuit was tested under operational conditions to ensure that the timing sequences match the design specifications, thus validating the functionality of the data distribution subsystem.

D. *Analysis And Discussion*

The data distribution system, enhanced by the innovative use of a counter to generate selector signals based on CLK1, provides a robust mechanism for managing the precise timing required for digital operations in the dice roller system. This setup not only simplifies the complexity of signal synchronization but also enhances the overall reliability and accuracy of the dice roll outcomes. The successful implementation and testing of this system demonstrate its effectiveness in a practical application, setting a precedent for similar digital logic applications requiring meticulous timing control.

FULL ADDER

The full adder in this system is pivotal, doing binary addition of two 2-bit numbers A (A1A0) and B (B1B0), with added carry (C0), to bring forth the sum. This result (X2X1X0) is stored in the output register (RO) and refreshed at a 100Hz frequency (fck4). To accomplish the above,

Sum and Carry outputs: It adds 2-bit numbers (A1A0) and (B1B0) and Cin is set to '1' for one of the 1-bit full adder. In the logical operation of the full adder, it provides both the output sum (S) and carry out (Cout) logic. This results in a constant carry, and further simplifies the logic needed to represent the numbers 1 through 7, which correspond to the seven-sided dice rolls. The circuit makes sure that the output is valid to be within the range of face values of the dice by using conditional logic to correct any overflow above the desired range.

Contribute to Randomness: The full adder contributes to randomness through the incorporation of input information from the intermediate register (RI), which has been contributed from the data generator through D0 with random distribution. Additional contributions of randomness are assured by not synchronizing f_{data} and fck4 from the project specifications.

A. *Circuit Details And Simplifications*

a. *First Bit (CMOS Configuration):*

This 1-bit full adder is designed and built in CMOS configuration. BS170 and BS250P mosfet models are used in this 1-bit full adder.

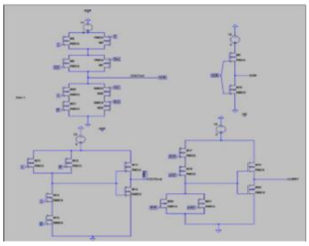


Figure 3.1

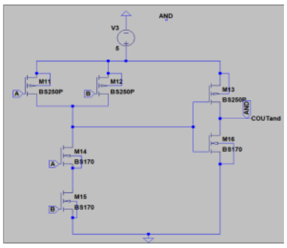


Figure 3.2

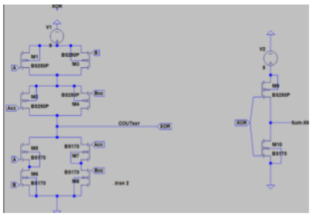


Figure 3.3

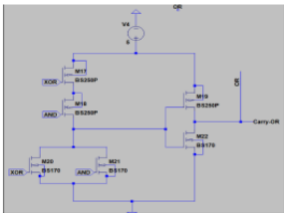


Figure 3.4

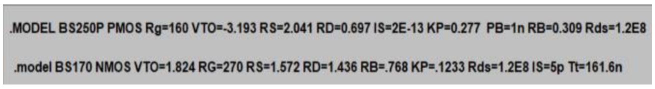
The CMOS full adder circuit was purposefully designed to execute binary addition tasks efficiently within the uneven seven-sided dice roller system. Its integration of PMOS and NMOS transistors in a CMOS configuration ensures robust performance while maintaining low power consumption. The circuit includes XNOR gate, AND gate and OR gate and they are designed and built in CMOS configuration. The sum is output of CMOS XOR gate circuit while the carry is output of CMOS OR gate circuit. As know Cin is set to '1'. A0 and B0 received from intermediate register are inputs for the full adder in Figure 3.1. The complements of A and B are also used in this configuration.

Labels are used to connect the circuits to each other as seen in Figure 3.1 – 3.2 – 3.3 – 3.4.

Figure 3.1 illustrates Circuit Diagram of The Selected Full Adder while Figure 3.2 illustrates Circuit Diagram of CMOS AND Gate Circuit.

Figure 3.3 illustrates Circuit Diagram of CMOS XOR Gate and SUM-XNOR Circuit while Figure 3.4 illustrates Circuit Diagram of CMOS OR Gate Circuit.

For the sum part, A0 and B0 pass through XOR gate and then the output is inverted. At the end of this process, the sum is gotten. For the carry, A0 and B0 pass through AND gate and then the output goes through OR gate. At the end of this process, the carryout is gotten.

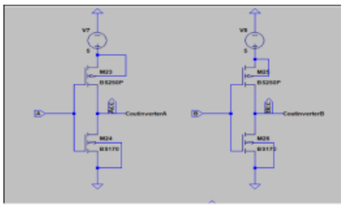


```
.MODEL BS250P PMOS Rg=160 VTO=-3.193 RS=2.041 RD=0.697 IS=2E-13 KP=0.277  PB=1n RB=0.309 Rds=1.2E8

.model BS170 NMOS VTO=1.824 RG=270 RS=1.572 RD=1.436 RB=.768 KP=.1233 Rds=1.2E8 IS=5p Tt=161.6n
```

Figure 3.5

As shown in Figure 3.5: VTO= -3. 193 and KP=0.277 for BS250P, VTO= 1.824 and KP=.1233 for BS170



Inverters for A0 and B0 Pulses in Figure 3.6

Figure 3.6

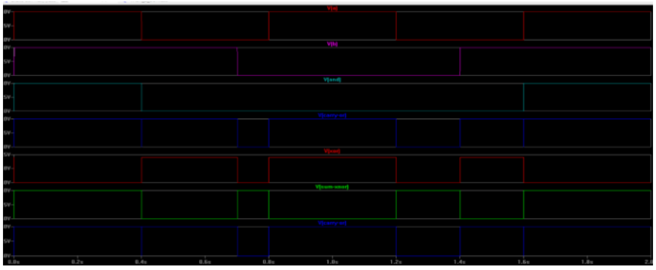


Figure 3.7

Representations: A: A0, B: B0, AND: AND gate output, Carry-OR: Both OR gate and Carry output, XOR: XOR gate output, SUM-XNOR: Both XNOR gate and SUM output.

As seen in Figure 3.7, the graph window shows us graphs of the outputs of CMOS logic gates in the circuit. According to Figure 3.8 Turuth Table, the outputs of the gates are as what they should be. In the graph, the outputs from XOR, AND, OR, AND gates are shown separately. It is obvious that the low voltage level is 0V while the high voltage level is 5V as they should be.

The sum output is equal to X0, and the carry output is equal to C0. C0 is the input of the other 1-bit full adder designed with IC. X0 is the input of the output register.

A	B	Cin	Sum (S)	Cout
0	0	1	1	0
0	1	1	0	1
1	0	1	0	1
1	1	1	1	1

(Figure 3.8 | Truth Table of the Full Adder designed with Mosfets)

b. Second Bit (Full Adder IC)

Extends the computational logic to include handling of the second bit based on the output from the first bit, ensuring comprehensive processing of all inputs.

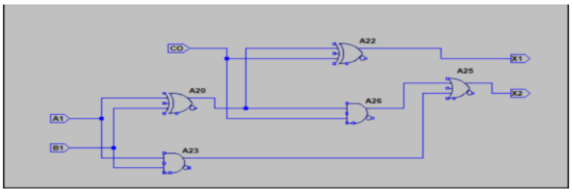


Figure 3.9

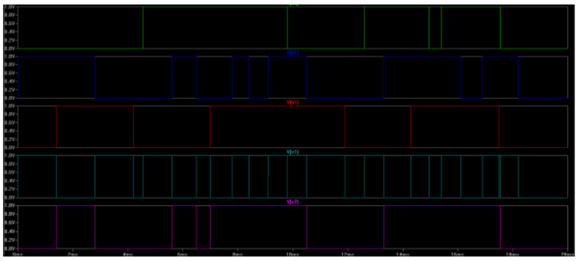


Figure 3.10

Representations: X1: Sum of A1 and B1, X2: C$_{out}$

As seen in Figure 3.10, the graph window shows us graphs of the outputs of 1- bit full adder designed with gates in LTspice. The full adder adds A1 and B1. Then it gives X1 and X2 outputs to the output register. It is obvious that the low voltage level is 0V while the high voltage level is 5V as they should be.

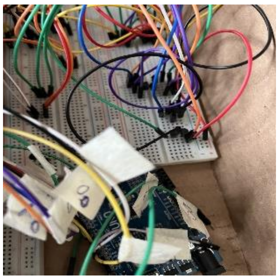


Figure 3.11

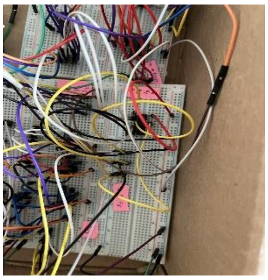


Figure3.12

Figure 3.11 is the photograph of physical full adder circuit built in CMOS configuration.

Figure 3.12 is the photograph of physical full adder circuit built with Full Adder IC.

B. Conclusion

The dual-configuration approach of a full adder design adeptly meets the requirements for binary addition in a dice roller system. The CMOS-configured first bit adder is a powerful configuration, as it simplifies the logic operations and reduces the circuit complexity by setting a constant 'carry-in' to high. The second-bit adder, constructed by using a standard full adder IC, processes the carry-forward element from the computation of the first adder. This unique blend of CMOS and IC technologies gives rise to precision not only in addition but also elaborates articulately on the synergistic integration of assorted design philosophies to effectively meet the demands of the system.

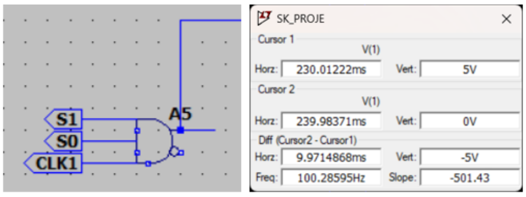
REGISTERS (RI AND RO)

Output Register (RO)

The 1-bit number values X2, X1 and X0, which are obtained by adding the A1, A0, B1, B0 and C0 values in the full adder circuit, are stored and repeated with the fck4 (100Hz) frequency value at the output of the output register (RO).

Obtaining CLK4

Saving and repeating of the output register was controlled with the CLK4 signal with a value of 100 Hz. The CLK4 signal is generated from the S0, S1 and CLK1 signals produced in the intermediate register to distribute the D0 value to the flip flops sequentially. S0, S1 and CLK1 signals were passed through the AND GATE integrated with serial number 74LS11 to obtain the CLK4 signal with a frequency of 100 Hz.



Cursor 1 V(1)
Horz: 230.01222ms Vert: 5V
Cursor 2 V(1)
Horz: 239.98371ms Vert: 0V
Diff (Cursor2 - Cursor1)
Horz: 9.9714868ms Vert: -5V
Freq: 100.28595Hz Slope: -501.43

Figure 4.2 CLK4 Circuit Diagram and Simulation Result values

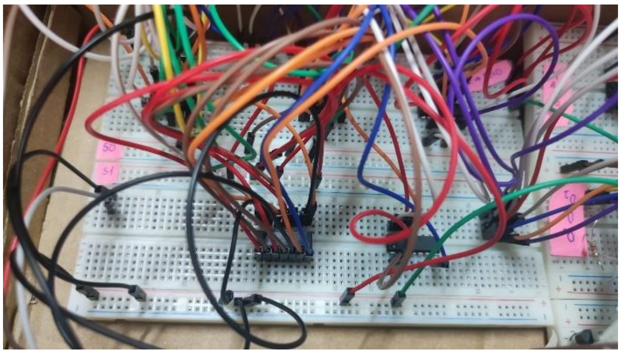


Figure 4.3- Physical circuits of CLK 4

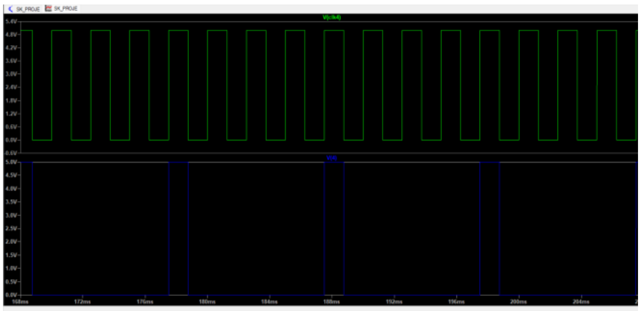


Figure 4.4 - Simulation Graphs of CLK1 and CLK4

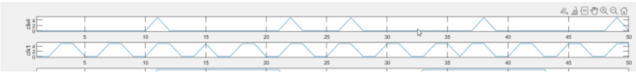


Figure 4.5 – Experimental Value of the CLK 4

As seen in Figure 4.6, the graphs of the CLK1 signal with a frequency of 400 Hz and the CLK4 signal with a frequency of 100 Hz are shown. The CLK4 signal was found to have one-fourth the frequency of CLK.

Output Register (RO) Circuit

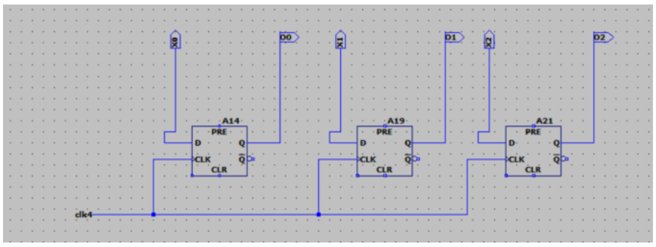


Figure 4.6 – Circuit Diagram of Output Register

74ls74 D-flip flop integrations are used in the output register circuit. The flipflops controlled by the CLK4 signal stored the X2, X1 and X0 1-bit number values from the full adder circuit by repeating them with a 100 Hz frequency value. The stored O2, O1 and O0 values were sent to the Arduino microcontroller and the analog values were read and converted from the binary number system to the decimal number value.

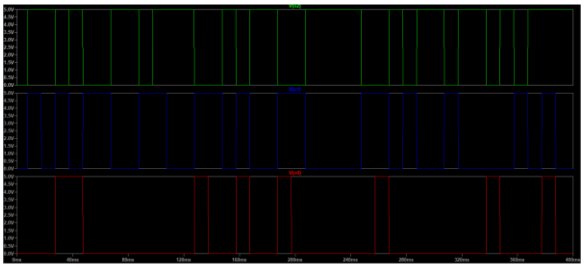


Figure 4.7 – Simulation Result Graphs of Output of the RO

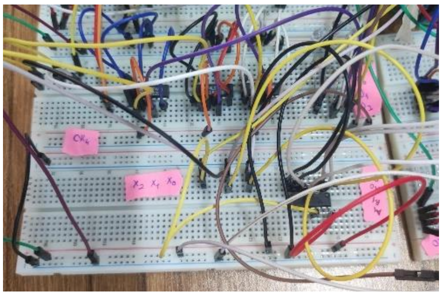


Figure 4.8 – Physical Circuit of Output Register

Reading $O_2O_1O_0$ Values by Arduino and Converting them to Decimal Values

The O2, O1 and O0 analog signals received from the output of the RO were sent to the Arduino microcontroller to be converted into a decimal number system and to compare the probability values with the theoretical values. The signal data obtained with Arduino is instantly plotted by MATLAB.

O_2	O_1	O_0	Decimal Value
0	0	0	**0**
0	0	1	**1**
0	1	0	**2**
0	1	1	**3**
1	0	0	**4**
1	0	1	**5**
1	1	0	**6**
1	1	1	**7**

(Table 1.2 – Conversion of output values to decimal values)

Dice Output Distribution and Probability Calculation

The accuracy table of the full adder scheme shown in Figure 3.6 was extracted and the probability calculation was made by calculating how often the O2, O1 and O0 values came. The accuracy table of the circuit is given in Table 1.3 and the number of dice outputs is theoretically given in Table 1.4.

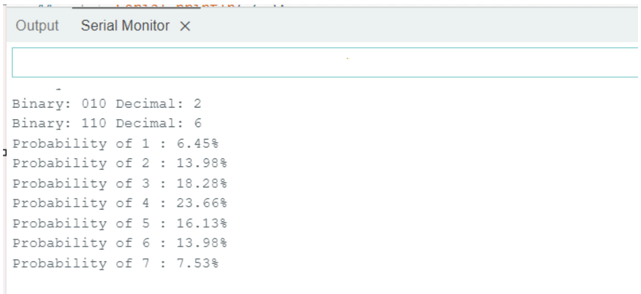
A1	B0	A1	B1	A22 (X_0)	A 23	A 11	A12 (X_1)	A 17	A 20	A21 (X_2)
0	0	0	0	1	0	0	0	0	0	0
0	0	0	1	1	0	1	1	0	0	0
0	0	1	0	1	0	1	1	0	0	0
0	0	1	1	1	0	0	0	0	1	1
0	1	0	0	0	1	0	1	0	0	0
0	1	0	1	0	1	1	0	1	0	1
0	1	1	0	0	1	1	0	1	0	1
0	1	1	1	0	1	0	1	0	1	1
1	0	0	0	0	1	0	1	0	0	0
1	0	0	1	0	1	1	0	1	0	1
1	0	1	0	0	1	1	0	1	0	1
1	0	1	1	0	1	0	1	0	1	1
1	1	0	0	1	1	0	1	0	0	0
1	1	0	1	1	1	1	0	1	0	1
1	1	1	0	1	1	1	0	1	0	1
1	1	1	1	1	1	0	1	0	1	1

Table 1.3 – Truth Table of the Full Adder Circuit

X2	X1	X0	Number of Dice	How many times?	Possibility (%)
0	0	0	0	0	0
0	0	1	1	1	6.75
0	1	0	2	2	12.75
0	1	1	3	3	18.75
1	0	0	4	4	25
1	0	1	5	3	18.75
1	1	0	6	2	12.5
1	1	1	7	1	6.25
			Sum	16	100

Table 1.4 – Theoretically Probabilities of Dice Outputs

The output values read in Arduino were collected according to the decimal values and the data were recorded by collecting the number of times each number came.



```
Binary: 010 Decimal: 2
Binary: 110 Decimal: 6
Probability of 1 : 6.45%
Probability of 2 : 13.98%
Probability of 3 : 18.28%
Probability of 4 : 23.66%
Probability of 5 : 16.13%
Probability of 6 : 13.98%
Probability of 7 : 7.53%
```

Figure 4.10 – Possibility Values from Arduino

D. Results

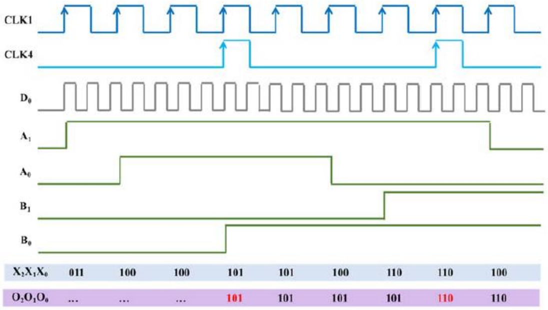


Figure 4.11 – Exemplary timing diagram.

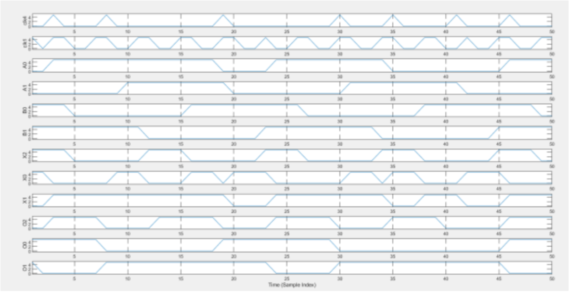


Figure 4.12 – Timing Diagram from MATLAB

It is observed that the sample time diagram given in the Uneven Seven-Sided Dice Roller project draft (Figure 4.11) and the graphs formed by reading analog signals from the circuit and projecting them simultaneously to the MATLAB (Figure 4.12) match each other to a large extent.

In the analysis performed, it was determined that the frequency ratios of the CLK1 signal and the CLK4 signal, which provide the basic functioning of the system, are appropriate. In addition, as a result of a detailed examination, the analysis of the A1, A0, B1 and B0 graphs has shown that with the positive edge trigger feature of our system, the D0 data information has been transmitted to the flip-flops without significant time delays, but in the correct sequential manner. This situation has confirmed the capacity of the system to provide the desired data flow. In addition, it has been observed that the intermediate register functions Decently.

Examination of the data from the output register has shown that the values of X2, X1 and X0 are stored by repeating the data from the full adder fed by the CLK1 signal with a frequency of 400 Hz with a CLK4 signal with a frequency of 100 Hz. This situation confirms the suitability of the system in data storage and processing processes. These results show that the basic clock signals in the system are properly synchronized, and the data processing processes take place correctly.

Number of Dice	Theoretical Possibility (%)	Possibility Values from Arduino (%)	Error (%)
0	**0**	**0**	**0**
1	**6.75**	**6.45**	**4.44**
2	**12.75**	**13.98**	**9.64**
3	**18.75**	**18.28**	**2.51**
4	**25**	**23.66**	**5.36**
5	**18.75**	**16.13**	**14.00**
6	**12.5**	**13.98**	**11.84**
7	**6.25**	**7.53**	**20.48**
Sum	**100**	**100**	

Table 1.5 – Theoretically Probabilities of Dice Outputs

In Table 1.5, the error margins are shown by comparing the data obtained from the output register with the probability of dice outputs and their theoretical calculations by reading them in Arduino. The average margin of error was calculated as 9.29%.

.

E. Conclusion

In this study, it has been observed that the time diagram of the Uneven Seven-Sided Dice Roller project and the graphics containing the projection of analog signals read through the circuit in MATLAB largely match each other. As a result of the analysis, it was determined that the frequency ratios of CLK1 and CLK4 signals, which provide the basic functioning of the system, are appropriate. In addition, as a result of a detailed examination, it was seen that the positive edge trigger feature of the A1, A0, B1 and B0 graphs and the D0 data information were transmitted to the flip-flops without a significant time delay and in the correct order. This confirmed the system's capability to provide the desired data flow. Additionally, it has been observed that the X2, X1 and This confirms that the system is suitable for data storage and processing. These results show that the underlying clock signals in the system are properly synchronized, and data processing processes occur correctly.

When we focus on the margins of error, we see that the margins of error for each number of dice are approximately 9.29% on average. These margins of error point to areas where further improvements are needed in the design and implementation of the system. More testing and optimization can be done in later stages.

CONCLUSION

An effort to make an uneven seven-sided dice roller system has resulted in cohesive combining of the electronic parts and logical sequencing, coming up to a fully functional prototype. Ingeniously, this system leverages the stability of a multivibrator circuit for data generation and the precision of CMOS and IC full adders for arithmetic logic, all orchestrated by a meticulously distributed clock signal.

This project is, therefore, successful in that it employs the adept manipulation of the CLK1 signal to derive the CLK4 frequency required in synchronizing the updating of the dice roll outputs. The efficient integration includes two full adders configured with a fixed carry-in for the low-order bit processing and CMOS, while the other uses standard IC for high-order bit processing and carry propagation.

From our testing, the dynamics of signal processing from the multivibrator-generated D0 to the output register, characterized with minimal time delay and accurate sequencing, are in adherence to the designed intent. Such results were indicative that the total error margin is around 9.29%. While this value can be regarded as acceptable for the prototype system, it signifies that there is still some room to improve reliability and accuracy of the system.

The result is not only an apparatus that meets the aim of generating a 3-bit number equivalent to that of a three-sided die but also how discrete electronic components can be realized to fulfill complex, nonstandard computational tasks. Future improvements will mainly be focused on minimizing the error margin, optimization of the component's efficiency, and expanding the system capabilities for an interface that works like a charm with all kinds of applications.

With this accomplishment, the project certainly lays a strong foundation for the future exploration of unconventional digital systems. The experience and the insights gained will definitely be very useful in the future for further innovative endeavors in this area.

References

[1] Hah, D. Lecture Notes

[2] Cd-Team, “Flip Flop Types, Truth Table, Circuit, Working, Applications,” *Electronics for You*, Aug. 23, 2023. https://www.electronicsforu.com/technology-trends/learn-electronics/flip-flop-rs-jk-t-d

[3] GeeksforGeeks, “Full Adder in digital logic,” *GeeksforGeeks*, Aug. 07, 2023. https://www.geeksforgeeks.org/full-adder-in-digital-logic/

[4] “CMOS nedir? | ElektrikPort Akademi.” https://www.elektrikport.com/teknik-kutuphane/cmos-nedir-guc-elektronigi-dersleri/11329#ad-image-0

[5] Wikipedia contributors, “Multivibrator,” *Wikipedia*, Mar. 11, 2024. https://en.wikipedia.org/wiki/Multivibrator

[6] A. R. Martin, “The Art of Electronics. By Paul Horowitz and Winfield Hill. Pp. 716, (Cambridge University Press, 1980.). £35.00 (hardback), £12.50 (paperback),” *Quarterly Journal of Experimental Physiology*, vol. 66, no. 4, pp. 543–544, Oct. 1981, doi: 10.1113/expphysiol.1981.sp002597.