

# Switchable Counter & Variable Pulse Generator

Beyza YILDIRIM

Department of Electrical and Electronic  
Engineering, Abdullah Gül University  
Kayseri, Turkiye  
beyza.yildirim@agu.edu.tr

Nisanur TÜRKMEN

Department of Electrical and Electronic  
Engineering, Abdullah Gül University  
Kayseri, Turkiye  
nisanur.turkmen@agu.edu.tr

**Abstract**—This project presents the design and implementation of a switchable BCD counter integrated with a duty-cycle-variable pulse generator. The system consists of two 1-bit signal generators operating at distinct frequencies (1 Hz and 2 Hz), a CMOS-level 2-by-2 multiplexer, a gate-level BCD counter, and a duty-cycle controller. A1 was implemented using an NE555 timer circuit, while A0 was generated via a frequency divider. The counter output, controlled by a selection switch (SC), increments by 1 or 2 based on input logic, and is displayed using a two-digit 7-segment display. A transistor-level CMOS MUX was designed and simulated to route A0 and A1 signals to the counter and pulse generator, depending on selection inputs SM1 and SM0. The pulse generator modifies the duty cycle of the output O0. The system was verified through both SPICE simulations and physical implementation, and the results confirm accurate frequency switching, BCD counting, and duty-cycle control.

**Keywords**—NE555 timer, frequency divider, 2X2MUX, CMOS circuit, asynchronous flip flop counter circuit, potentiometer op-amps

## INTRODUCTION

This project report has been prepared to discuss in detail the design and development process of the switchable counter and variable pulse generator system. The design is organized into three main tasks. In Task 1, a 2 Hz clock signal was generated using a 555 Timer IC, which served as a base timing source. In Task 2, a second 1 Hz frequency signal was generated and routed through a selector circuit based on SM0 and SM1 control signals, producing outputs B0 and B1. These clock signals were connected to a 2×2 CMOS multiplexer designed using BS250 (PMOS) and BS170 (NMOS) transistors. In Task 3, a two-digit BCD counter was implemented using D-type flip-flops, capable of incrementing by either 1 or 2 depending on the SC control input. To enable control over pulse width, an op-amp-based duty cycle shaping circuit was designed. This shaping circuit modified the duty cycle of the B0 clock without changing its frequency, using a diode, RC components, and a potentiometer.

## DESIGN AND TEST PROCEDURES

### The 555 Timer-Based Clock Signal Design

In this project, the design process begins with the generation of a square wave clock signal using the NE555 timer IC in astable mode. The 555 timer is used to generate variable pulses and provide a clock signal to the counter circuit. The frequency can be controlled with adjustable resistors and capacitors so that the output signal can be shaped according to the requirements of the project. The NE555 timer is preferred due to its flexibility, efficiency, safety, simplicity, and ease of implementation on a breadboard or simulation environment. To achieve the desired frequency and duty cycle, an online 555 timer calculator was used.[1] Based on this tool, the required component values were selected to generate a frequency close to 2 Hz with a duty cycle near 50%. The selected components were: 5 kΩ for R1, 250 kΩ for R2, and 1.5 μF for C.

$$f = \frac{144}{(R_1 + 2R_2) \times C} \quad (1)$$

$$f = \frac{144 \times 10^{-2}}{(5+500) \times 10^3 \times 15 \times 10^{-9}} = 1.9009901 \text{ Hz} = 0.52604167 \text{ s} \quad (2)$$

$$T_H = 0.694 \times (5000 + 250000) \times 1.5 \times 10^{-6} = 0.265455 \text{ s} \quad (3)$$

$$\text{Duty-cycle: } \frac{T_H}{T} \times 100 = \frac{0.265455}{0.52604167} \times 100 = 50.4627324 \quad (4)$$

The frequency and pulse widths were calculated using the standard formulas for the NE555 in astable mode. With the chosen values, the resulting frequency was approximately 1.90 Hz and the duty cycle was about 50.46%, which was acceptable for the intended application.

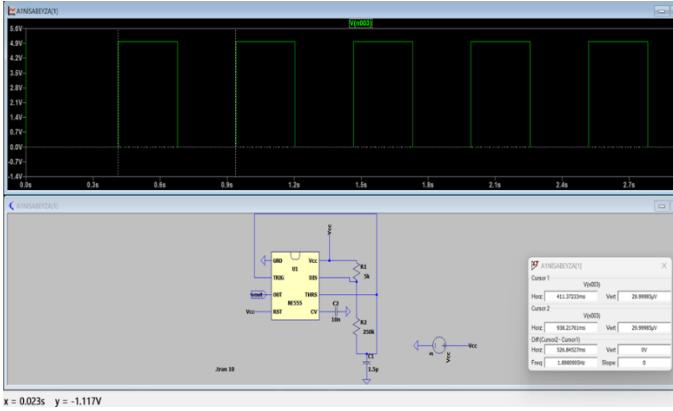


Figure 1: NE555 timer circuit and its square-wave graph

The output signal, determined as A1, served as one of the primary clock sources in the later stages of the project.

#### A. A0 and 2x2 MUX Design

In this stage, a 1 Hz clock signal labeled A0 was obtained from the previously generated 2 Hz clock signal (A1) by connecting it to a D-type flip-flop. A 74LS74 flip-flop was used in this configuration to ensure synchronized and stable frequency division. This method allowed the derivation of a clean, accurate A0 signal that could be reliably used alongside A1 as an input to the multiplexer system.

To control which clock signal would be forwarded to the counter, a custom-designed 2×2 CMOS multiplexer was implemented. The inputs of the multiplexer were A0 and A1, while the outputs were labeled as B0 and B1. Two digital control signals, SM0 and SM1, were used to determine which input would be routed to which output, according to a predefined truth table.

Table 1: The truth table which is used

Sm1	Sm0	B0	B1
1	1	A1	A1
0	0	A0	A0
1	0	A1	A0
0	1	A0	A1

The Boolean expressions for the outputs were derived from this truth table using Boolean algebra:

$$B_1 = (\overline{Sm0} \cdot A_0) + (Sm_0 \cdot A_1) \quad (5)$$

$$B_0 = (\overline{Sm1} \cdot A_0) + (Sm_1 \cdot A_1) \quad (6)$$

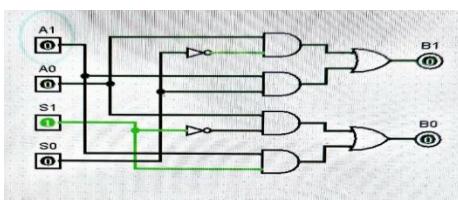


Figure 2: The design has been created accordingly Boolean algebra equations.

These formulas were implemented in LTSpice using logic gates and a custom AOI (AND-OR-Invert) gate model that was specifically defined for this circuit. The AOI gate performs the operation  $y = \overline{AB} + \overline{CD}$ , which matches the core structure of the multiplexer output expressions. However, because the AOI gate inherently outputs the inverted result of the desired logic, an additional CMOS inverter gate was placed at the output to correct the logic level. This configuration allowed for more compact implementation and reduced transistor count.

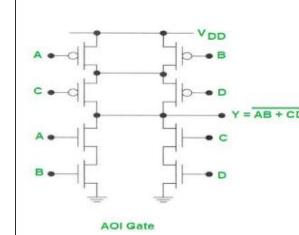


Figure 3:AOI gate design [2]

#### B. 2X2 MUX

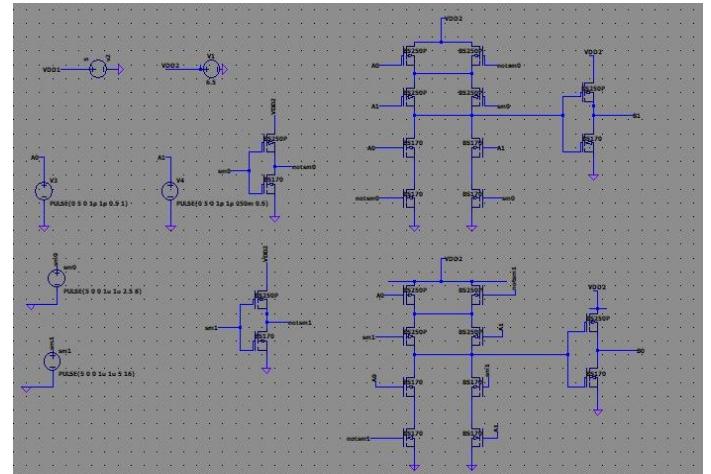


Figure 4:2X2 MUX circuit on LTspice

The multiplexer itself was constructed at the transistor level using BS250 (PMOS) and BS170 (NMOS) transistors. A total of 24 transistors were used for the full implementation. CMOS technology was chosen due to its low power consumption, high noise immunity, and compatibility with LTSpice modeling. The BS250 was selected for its suitability in low-current, high-efficiency PMOS applications, while BS170 was used due to its fast switching performance under low-voltage conditions. The necessary models for both transistors were manually added to the LTSpice library.

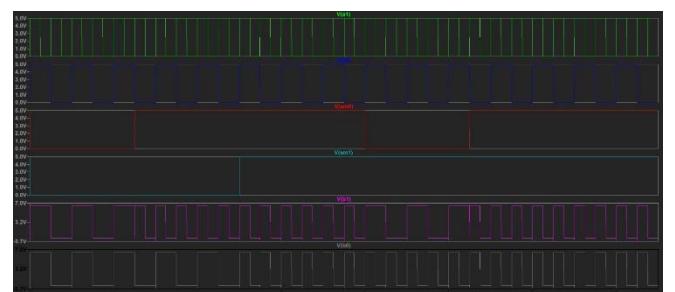


Figure 5: all outcomes for task 2

Following the simulation, the entire multiplexer circuit was also implemented on a breadboard for physical testing. The A0 signal was created from A1 using the flip-flop circuit, and two separate CMOS multiplexer boards were assembled. Switches were used to manually control SM0 and SM1, and LEDs were connected to A0, A1, B0, and B1 outputs to visually verify correct switching behavior. The observed output logic was confirmed to be fully compliant with the expected truth table.

### C. Switchable Counter and Duty Cycle

Switchable counter circuit represents a binary counter system designed to count in increments of 1 or 2, based on the input control signals. The system uses D flip-flops and multiplexers (MUX) to control the counting process and manage the frequency of the clock signal driving the counter. The B1 signal is given directly to this circuit and the frequency of this counter is required to operate directly in parallel with B1. The setup is started with the appropriate D flip flops.

The D flip-flops are arranged to count 1 by 1 or 2 by 2 depending on the SC signal. They receive the clock signal from the 2x1 MUX on the left side of the figure. The first AND gate (A9 on the left side of the Figure 4) triggers nQ0, which is incremented by 1 since Q0 is the least significant bit (LSB) of D0 and 2 to the power of 0 is equal to 1. The lower left of the figure triggers the second D flip-flop (DFF), and Q1 is the second LSB of D0. Since 2 to the power of 1 is 2, the first DFF is never activated and the system counts by 2. The D flip-flops are configured to be reset or preset according to the PRE and CLR lines connected to the flip-flops. These lines allow the counter to be reset or preset to specific values as needed. Since a two-digit number must be obtained, the maximum value a digit can take is 9. If both Q3 and Q1 of D0 are 1, the binary system will overflow the value 9. To prevent this, the Q3 and Q1 outputs are fed to a NAND gate. The 0 value from this NAND gate is then sent to the CLR pins of D0, and the next step is to reset the ones digit from 9 to 0. Meanwhile, the tens digit is sent to a NOT gate, which triggers the CLK pin of the tens digit, causing the tens digit to change. The counter works by incrementing the stored value on the rising edge of the clock signal. In real life, 74LS74 DFFs were used to build the circuit. The reason for choosing 74LS74 when building a circuit is that this flip flop is positive edge triggered and has a 2-flip flop structure, thus reducing the number of flip flops to be used.

The output of the counter is a BCD (binary coded decimal) value displayed in two digits. The two digits are usually shown on a 7-segment display for visual representation of the current value of the counter. In this case, a cathode 7-segment display was tried, but cathode 7-segment displays work with a logical 1 signal. In other words, the logical 0 signal generated by 74LS74 is not suitable for cathode displays. Since the 74LS74 only produces low levels of logical 0 signals, it cannot provide correct signals for cathode type displays. In this case, instead of replacing the 74LS74 ICs, an anode 7-segment display was used. As a result, the reason why the 74LS74 flip-flops work with anode 7-segment displays is because anode displays work with logical 0 signals, and the 74LS74 also produces logical 0 signals. Since cathode displays require logic 1 signals, the 74LS74 is not

directly compatible. Also, the 74LS74 flip-flop stores binary data and sends it to the 74LS47 BCD to 7-segment decoder. The 74LS47 converts the 4-bit binary input into a 7-bit output, controlling which segments of the 7-segment display light up. Each binary combination corresponds to a decimal digit, and the 74LS47 ensures the correct segments are activated to display the digit on a common anode 7-segment display.

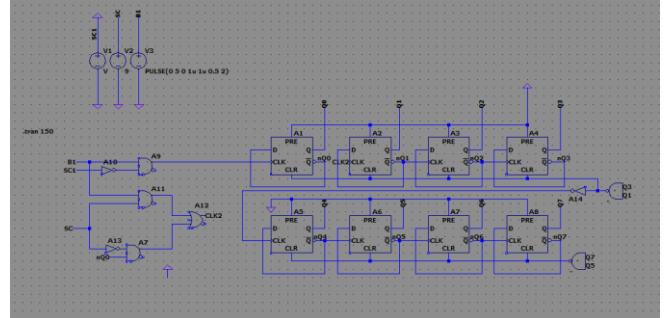


Figure 6: LtSpice of switchable counter circuit

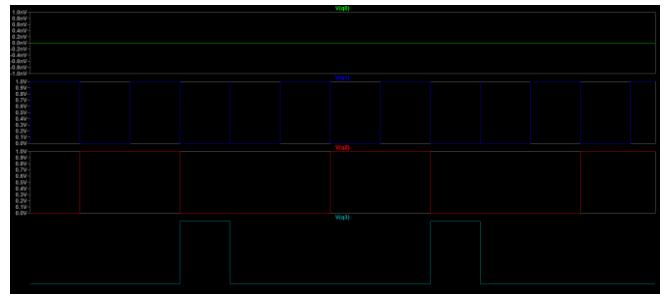


Figure 7: D0 values when the switchable counter circuit increments by 2

In second part of Task 3, the duty cycle of the O0 signal at the output comes from the B0 input, the B0 input signal is provided as a clock signal to the circuit that controls the duty cycle. This signal can be adjusted with the Rpot potentiometer in the circuit, since the LtSpice application does not provide instantaneous graphics, although the potentiometer model was added, it was named Rpot because instantaneous changes would not be seen and its accuracy was checked by changing the ratio of the resistance values. The potentiometer affects the duty cycle of the output signal by changing the voltage applied to the input of the op-amp U3 (UA741). The U3 op-amp is configured as a comparator and produces the output signal by comparing the voltage coming from the potentiometer with the reference voltage. The output signal provides an O0 signal with a pulse width changed according to the set voltage. In this way, the duty cycle of the O0 signal can be changed by changing the value of Rpot. Diode D1 (1N5819) and capacitor C2 (4.7 µF) in the circuit are used to shape and smooth the signal. The diode ensures that the current flows in the correct direction and the capacitor smooths the signal, resulting in a smoother output signal. By adjusting the potentiometer, the duty cycle of the O0 signal at the output can be adjusted to the desired level, while its frequency remains constant.

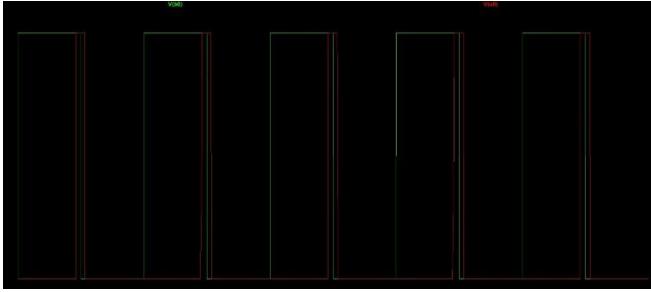


Figure 7: LTspice result of variable pulse generator

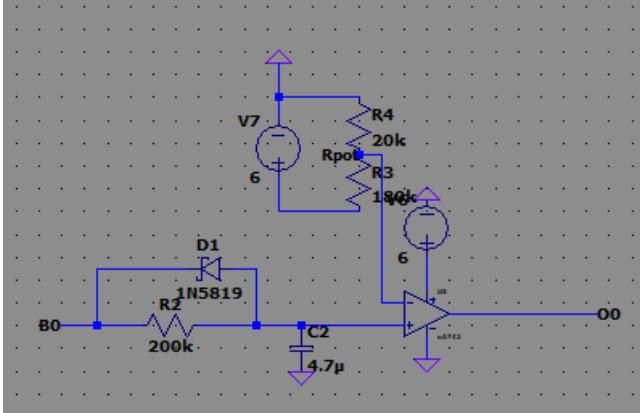


Figure 8: LTSpice of Variable pulse generator circuit

#### D. Entire System

In this project, the first step was generating a 2 Hz clock signal labeled A1. This signal was then fed into a D flip-flop (frequency divider) to produce a new clock signal, A0 (1 Hz). These two clock signals (A1 and A0) were then passed into a  $2 \times 2$  CMOS multiplexer, which, based on SM0 and SM1 control signals, determined which clock source would be used. SM0 and SM1 signals were controlled by switches, producing output signals B0 and B1.

The B1 output was connected to the BCD counter circuit, where its counting speed was synchronized with B1's frequency. When the switch controlling the frequency was changed, the counting process and the frequency of B1 alternated between 1 Hz or 2 Hz. Additionally, the counter's counting speed could be adjusted by a switch at the beginning of the counter. With the control switch closed, the counter incremented by 1; when open, the counter incremented by 2.

The B0 output was connected to the duty cycle control circuit. The switch that controlled B0 frequency also changed the frequency of O0 to match the selected B0 frequency, either 1 Hz or 2 Hz, and it was observed that B0 and O0 worked in parallel. Finally, the duty cycle of the O0 signal was adjusted with a potentiometer, keeping its frequency constant, and these adjustments were analyzed.

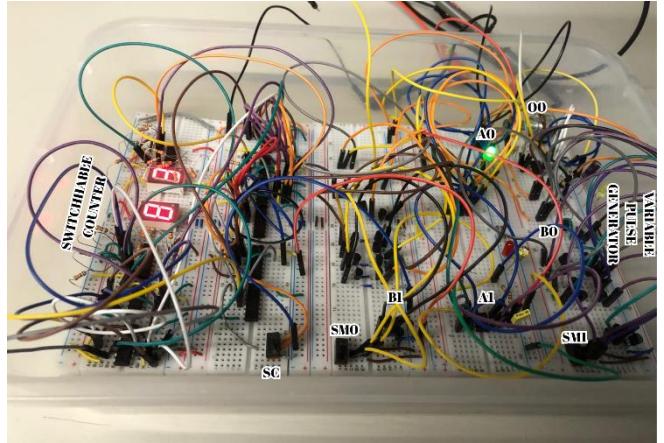


Figure 9: Entire system

## RESULT AND DISCUSSION

The results obtained in this project showed some minor differences between the simulation and real-world tests, but overall, the results were good to a degree that can be considered negligible. In the LTSpice simulation environment, all the circuit functions were validated; however, since the LTSpice circuit did not include the display and switches, the counter circuit's operation was demonstrated by setting up and testing the circuit in Proteus. A video was recorded to observe the operation of the circuit. The video of the tests performed in Proteus is provided as a link in the report.

Simulation results for the counter circuit were obtained in LTSpice, but some modifications were made. The 74LS74 flip-flop used in the circuit could not be integrated into the LTSpice simulation due to the lack of this model's capability in the system. A model that did not show any significant difference in performance was used instead; however, this model was not suitable for active-low operation, which caused the NAND gate to not work as expected. This issue was solved by using AND gates, and the version of the circuit with AND gates was used to generate the final graphs.

In addition to this, some differences were observed between the real-world tests and the simulations performed. In the real setup, the connections on the breadboard were made using more jumper cables and zinc wires. These connections introduced small variations due to the tolerances of the circuit components and the physical properties of the connecting elements. Factors such as the length of the wiring, the quality of the connections, and the placement of the components on the breadboard led to small resistance tolerances affecting signal transmission. Additionally, the tolerances of real components, such as potentiometer, resistors, and capacitors, caused slight value differences compared to the simulation. These variations did not prevent the circuit from operating properly in the real setup, but they introduced small differences from the simulation.

While ideal components were used in the simulation, real-world testing introduced possible connection errors and environmental factors (such as temperature and humidity), which caused small variations. However, these changes did not significantly affect the overall operation of the system. As a result, similar outcomes were achieved in both the simulation and the real tests. This demonstrates that the small

differences between the physical setup and the simulation environment are negligible and do not affect the overall accuracy of the circuit.

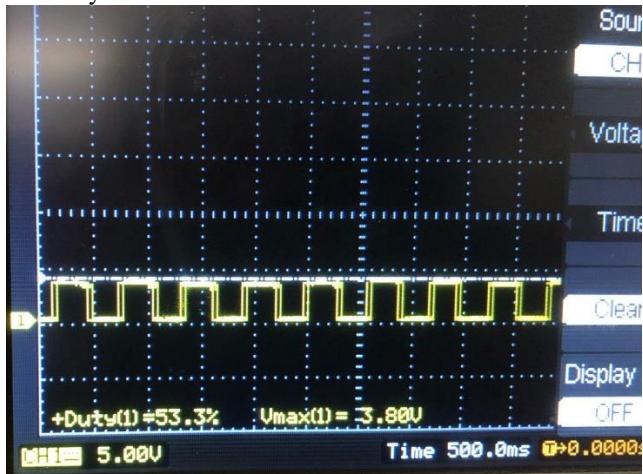


Figure 9: 53.3% duty cycle

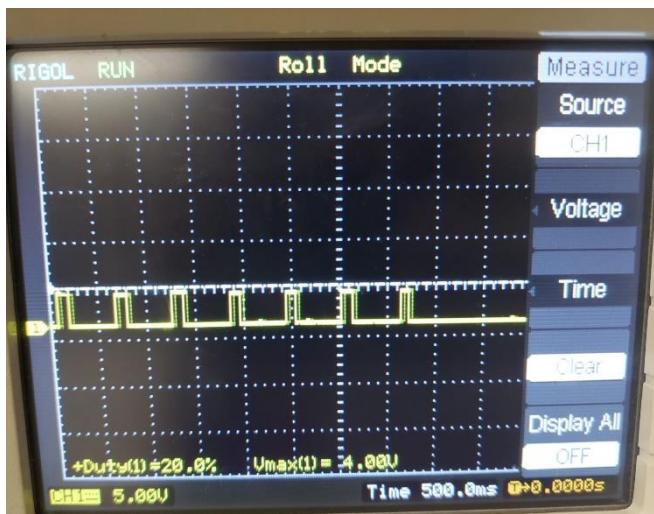


Figure 10: 20% duty cycle

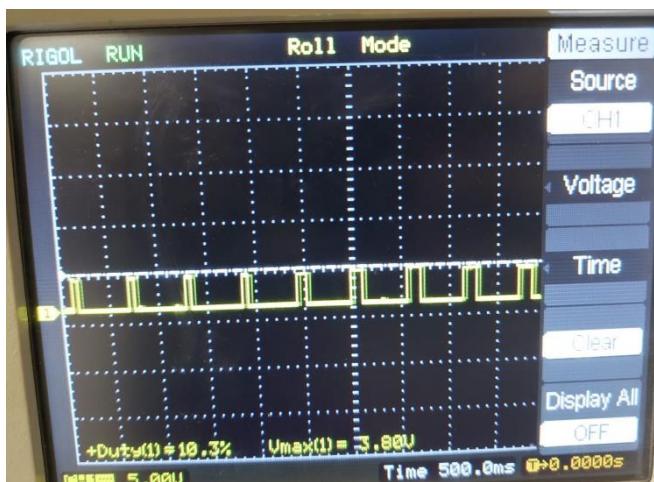


Figure 11: 10.3% duty cycle

## CONCLUSION

In this project, a modular digital system consisting of a clock generator, a CMOS-based multiplexer, and a switchable BCD counter was designed and successfully implemented. A 555 timer was used to generate stable clock signals, which were routed through a  $2 \times 2$  CMOS multiplexer to control the BCD counter. The system allowed the clock frequency to be switched between 1 Hz and 2 Hz based on the SM0 and SM1 control signals, and the counting speed could be adjusted with a switch, allowing either 1 or 2 increments.

The duty cycle of the clock signal was dynamically controlled using an op-amp-based circuit, with the duty cycle being adjusted via a potentiometer. These changes were verified through LTSpice and Proteus simulations. The results showed that the frequency of B0 and O0 remained synchronized, maintaining the same frequency while the duty cycle was modified.

In terms of the counting process, a reset logic using Q3 and Q1 outputs was integrated to ensure the counter only counted within the valid BCD range (0-9), resetting to 0000 once the binary value reached 1010.

Physical tests were also conducted on a breadboard, and the same results were obtained as in the simulation. A0, A1, B0, and B1 outputs, controlled by SM0 and SM1 via switches, correctly followed the expected logic. The O0 output was tested with duty cycle adjustments using a potentiometer, and these changes were consistent with the simulation results.

This project provided valuable experience in CMOS technology, 555 timer circuits, and op-amp design for digital system timing and signal generation. The accuracy and functionality of the system were confirmed through both simulation and physical testing on the breadboard. This work demonstrates the integration of digital components, logic gate usage, and timing control in a practical, real-world setting.

within the valid BCD range (0-9), resetting to 0000 once the binary value reached 1010.

Physical tests were also conducted on a breadboard, and the same results were obtained as in the simulation. A0, A1, B0, and B1 outputs, controlled by SM0 and SM1 via switches, correctly followed the expected logic. The O0 output was tested with duty cycle adjustments using a potentiometer, and these changes were consistent with the simulation results.

This project provided valuable experience in CMOS technology, 555 timer circuits, and op-amp design for digital system timing and signal generation. The accuracy and functionality of the system were confirmed through both simulation and physical testing on the breadboard. This work demonstrates the integration of digital components, logic gate usage, and timing control in a practical, real-world setting.

## Acknowledgements

We extend our sincere gratitude to Dooyoung Hah and Sinan Genç (Abdullah Gül University) for generously sharing their knowledge, which played a vital role in the successful completion of this study. We are also thankful to Umut Kavuncu's valuable support and insightful contributions throughout this project.

This work was carried out with the help of the facilities provided by the AGU URL Lab environment.

## REFERENCES

### Supporting Materials

Proteus Link: <https://youtu.be/w4lE5OwJWyk>

Lab Video Link:

<https://youtube.com/shorts/xt4uYdlcXKs?si=hGAkW6hEnbvgiTQh>

- [1]555 (NE555) Astable Circuit Calculator. (n.d.).  
<https://ohmslawcalculator.com/555-astable-calculator>

- [2](No date) *An efficient compound implementation of Aoi Gate and OAI Gate.* / download scientific diagram. Available at:  
[https://www.researchgate.net/figure/An-efficient-compound-implementation-of-AOI-gate-and-OAI-gate\\_fig5\\_364467351](https://www.researchgate.net/figure/An-efficient-compound-implementation-of-AOI-gate-and-OAI-gate_fig5_364467351)