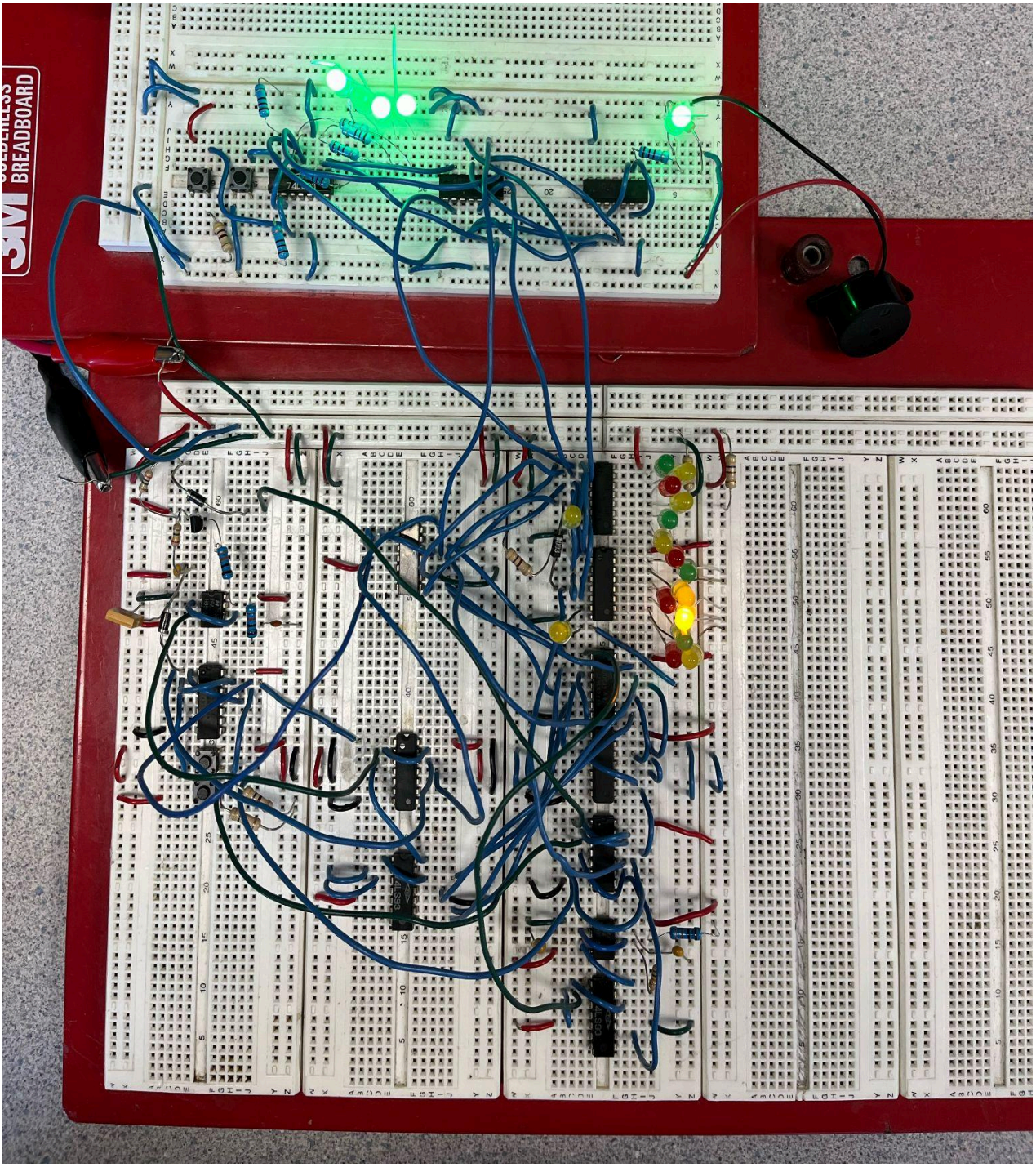
# Lab 8- Roulette Wheel




**Jason Cardente, Daniel Cardone, Ethan Herron, Jackson Staheli, Nick Vaughan, Henry Warfield**

**(11/28/22-12/6/22)**

**Objectives:**

- Build a 16 LED roulette wheel. This wheel will start with a push of a button then will slow down until it stops on a number. If the right number is picked the buzzer will go off as a winning number.

**Materials List:**

- Multisim
- Ultiboard
- Autodesk Inventor
- 5v DC Power Supply
- 7408 Quad 2-input AND Gate (2)
- 74266 Quad 2-input XNOR Gate (2)
- 7493N Counter (4)
- 74138N 3-Line to 8-Line Demultiplexer (2)
- 7402 Quad 2-input NOR Gate (2)
- 555 Timer (2)
- 1N4001 Diode (3)
- 2N3906 Transistor
- 10 uF capacitor
- 1 uF capacitor
- 0.1 uF capacitor (2)
- 0.01 uF capacitor
- 10M Ω Resistor
- 3.6M Ω Resistor
- 10k Ω Resistor (3)
- 2k Ω Resistor
- 470 Ω Resistor (5)
- 220 Ω Resistor (2)
- Buzzer
- LED (21)
- N/O switches (4)

**Procedure:**

**Multisim:**

1. To start creating a roulette wheel, we needed a circuit that would be able to slow down flashing LEDs so it can land on the winning LED, so we created a decay circuit as shown in figure 1-1 using a 555 timer.

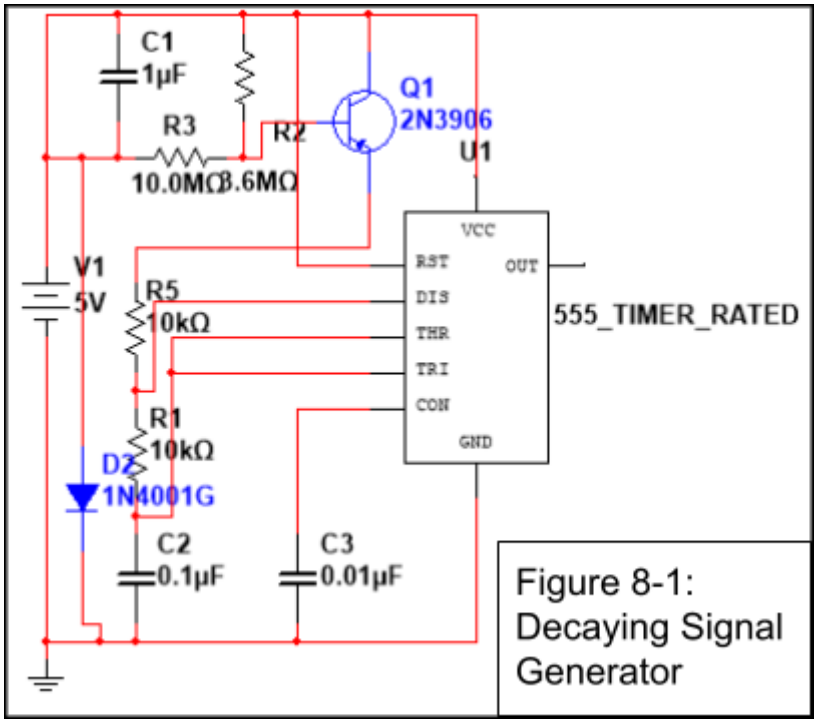



Figure 8-1: Decaying Signal Generator

2. We then needed to use an SR flip-flop to have a set and reset for the roulette wheel to reset the LED counter and spin the wheel again. This circuit can be shown in figure 1-2.

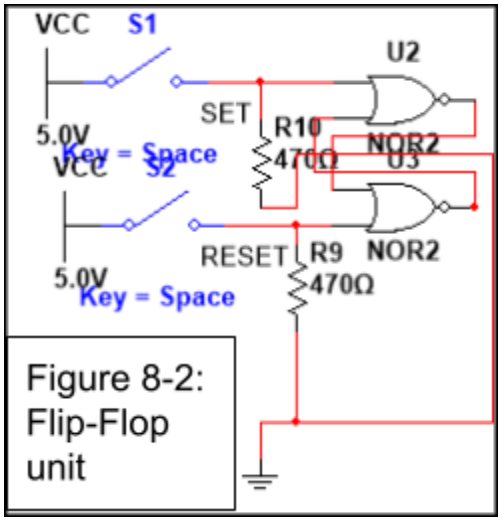
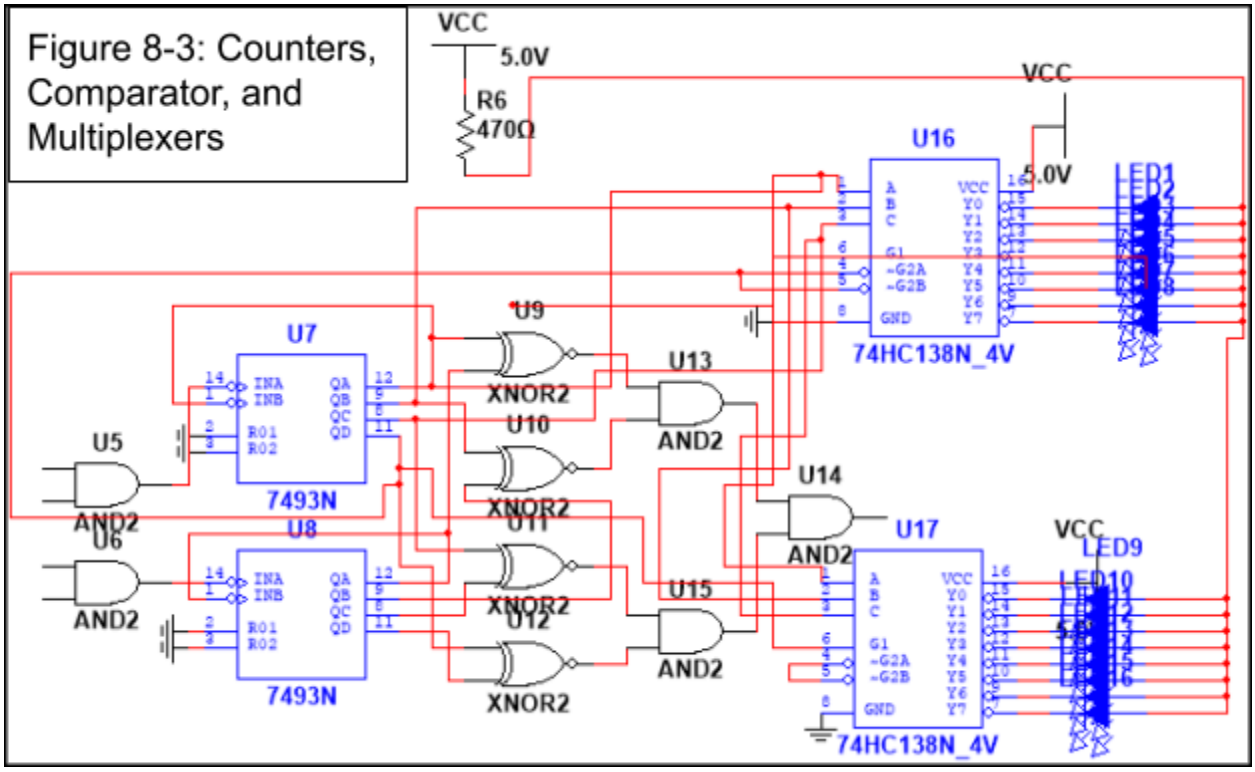
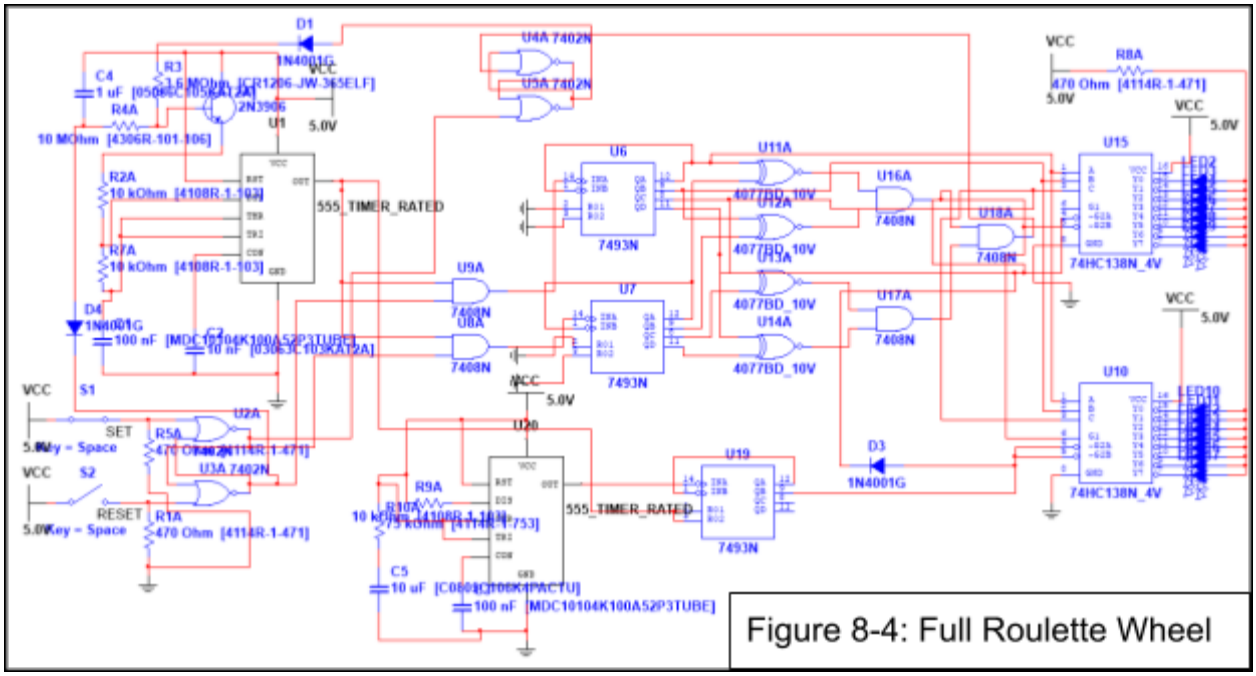



Figure 8-2: Flip-Flop unit

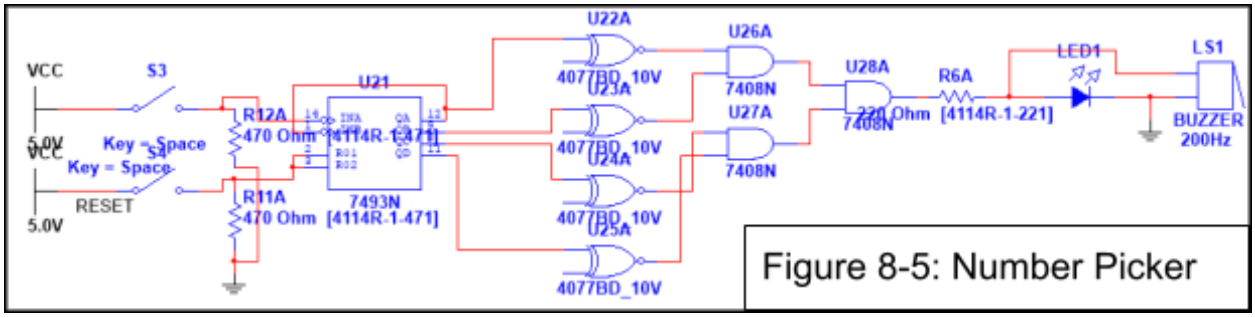
3. After the flip-flop we added a MOD-16 counter to count from 0-15 in binary when the clock pulse and Q are to select the LED. We also put another counter that would be active only when the clock pulse and Q-not were high that flows into a digital comparator with the first clock. By connecting R2 in figure 1 to this comparator instead of power it causes a random delay in the decaying signal generator
4. Then we added multiplexers to the output of the first counter to send the binary signal into 1 of 16 possible LEDs




5. We then combined all of the pieces above to create the roulette wheel circuit shown below. We also added a secondary 555 and a counter with the rest pin wired to the clock pulse so that when the LED will lock after remaining on an LED for over two seconds. We also added a flip-flop to save this condition until reset.




6. Once we had the full roulette wheel operational, we made the circuit shown in figure 8-5 which allows a person to set and reset a numerical guess, which is then compared with the output of the roulette wheel to activate a LED and buzzer if the correct number is picked




7. We then attached the number picker to the roulette wheel to make the final circuit shown in figure 8-67

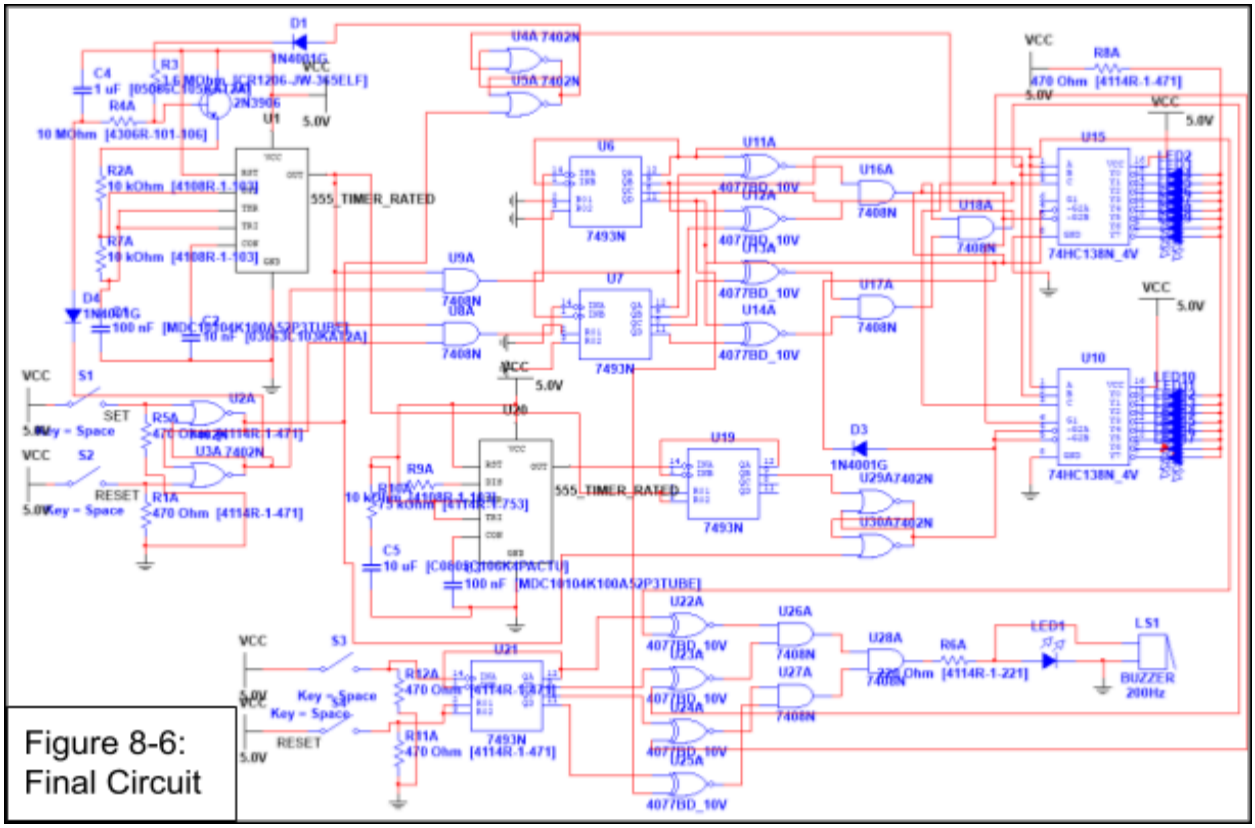



Figure 8-6:
Final Circuit

[Multisim File Link]

**Ultiboard:**

1. Next, using the transfer tool on Multisim, we transferred the file to Ultiboard.
2. We then changed the component packages to match the components we used on the breadboard in order to fit the needs of our board parameters which were 4 inches by 4 inches.
3. The final Ultiboard can be shown in figure 8-7.

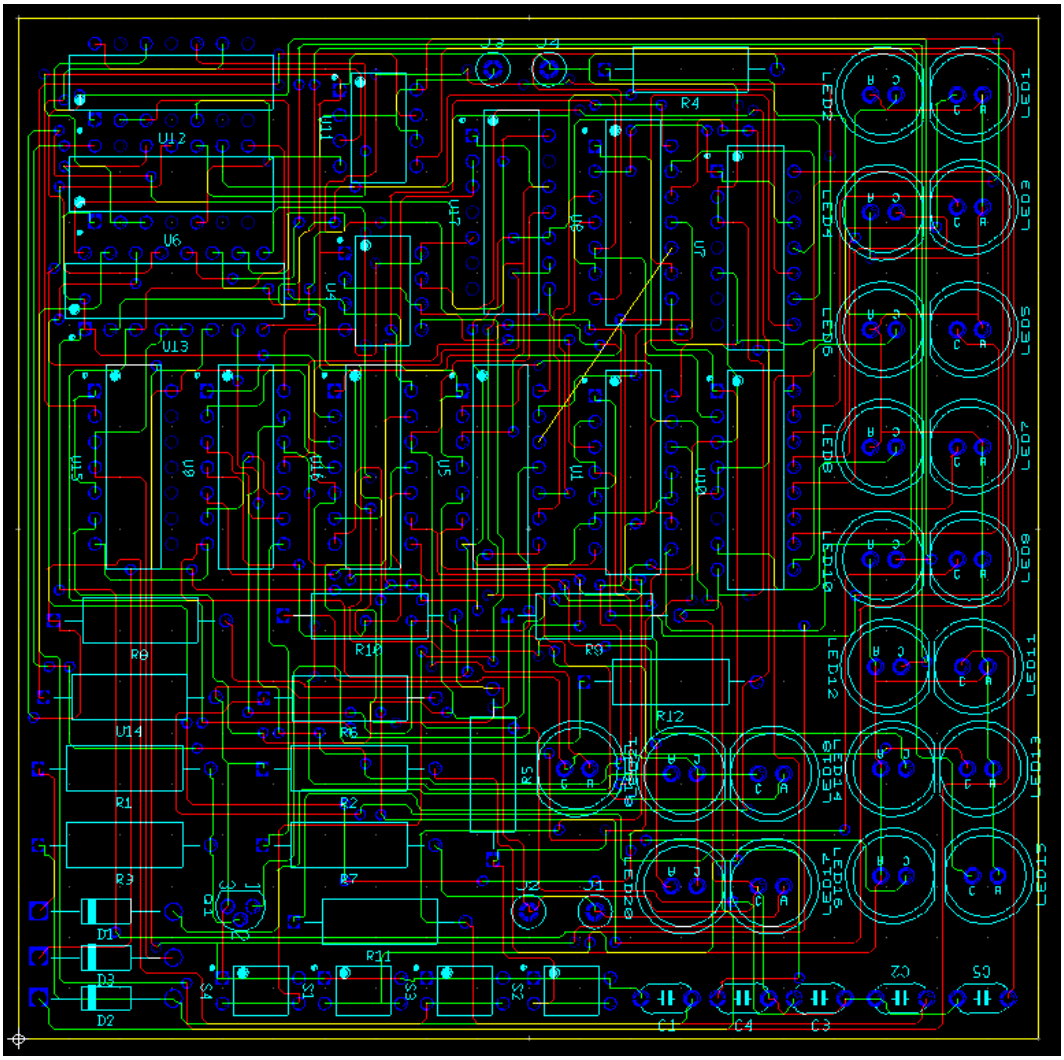


*Figure 8-7: Finalized Ultiboard*

[Ultiboard File Link]

**Autodesk:**

1. To make the enclosure for the circuit, start by making a 5.5” x 4.5” x 1.25” rectangle and using the shell tool to hollow it out, leaving 0.125” thick walls on all sides. While in the shell tool, select the top face in order to remove it completely. Fillet the outer edges of the enclosure.
2. To add holes for screws, start a 2D sketch on the top of the box and put 4 circles centered on the inner corners. These should be slightly wider than the diameter of the screws to account for tolerance. Add 4 more circles centered on the previous ones and make these slightly wider. Extrude both the outer and inner circles to fill downwards 1.25”, then extrude just the inner circle to cut downwards for the length of the screw.

3. Put two 0.02” diameter holes through one wall of the box in order to put power wires through them. The box can be shown in figure 8-8.

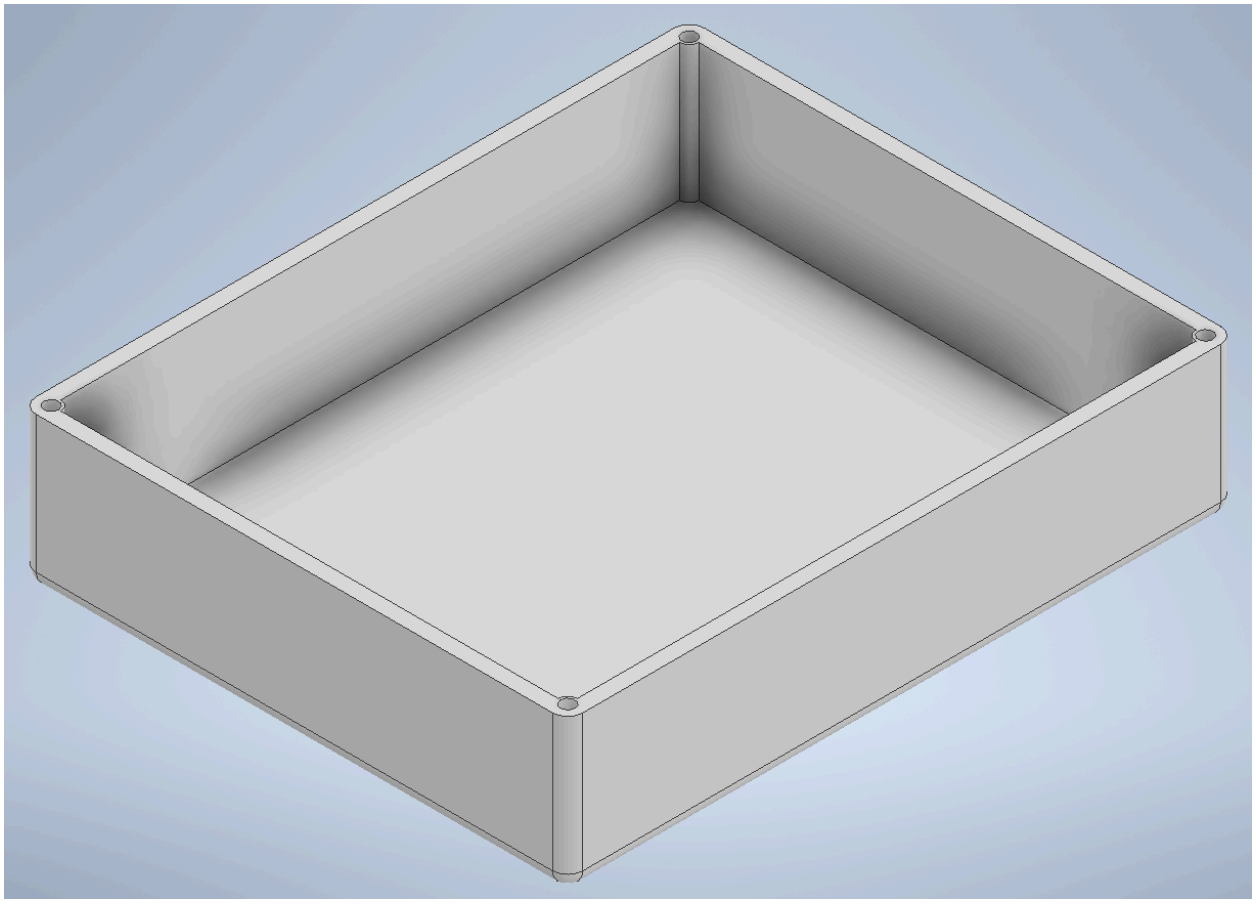


***Figure 8-8: Finalized Roulette Wheel Box***

**Roulette Wheel Box File Download**

4. Make a new part to become the lid and start by creating a 5.5” x 4.5” x 0.2” rectangle. Add the four screw holes 0.125” inwards from the outer edges of each corner and extrude these through the lid.
5. Start another 2D sketch on the top face of the lid and make a 2.5” diameter circle centered on the 4.5” axis of the lid. Make sure that its position is relatively central, as other sketches will go around it. Make a 0.24” diameter circle centered on where the centerline of the lid and the edge of the large circle intersect. Use the rotate tool to revolve this circle around the center of the large circle, and put 16 copies of it spaced evenly across 360 degrees.

6. Within the circle of 16 circles, add a 0.24” diameter circle centered along the centerline of the lid, but above the center of the 2.5” circle created in Step 4. Extrude only the 0.24” circles created in Steps 4 and 5, cutting them fully through the lid.
7. To make holes for the buttons add four holes below and to the sides of the large collection of circles. These holes should be square and 0.36” across diagonally.
8. For the four indicator lights, put four holes arranged horizontally below the large circle of holes. These should be circular with a 0.24” diameter. The finalized Roulette Wheel Lid can be shown in figure 8-9.

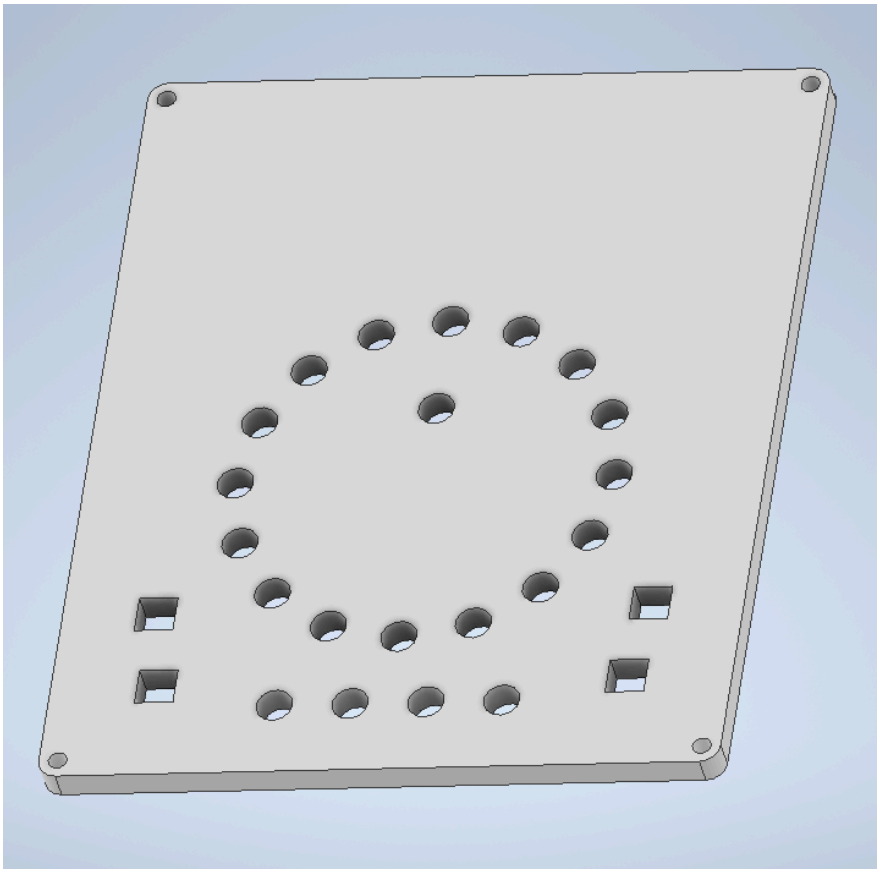


**Figure 8-9: Finalized Roulette Wheel Lid**

[Roulette Wheel Lid Download]

**Discussion:**

Symptom: The evenly numbered LED's were exhibiting blinking when the wheel landed on them.

Troubleshooting: We started at the secondary 555 timer because the frequency of the timing and the blinking were similar and worked our way up through the counter and the flip-flop

Solution: We moved the connection to the main counter from the Q of the flip-flop to the Q-not of the flip flop and the blinking was removed.

Symptom: The resistors on our multisim appeared as IC chips when we imported the file to Ultiboard.

Troubleshooting: We started checking the packages of the resistors and discovered that they were from an IC chip.

Solution: We changed the resistors used in multisim to axial resistors and they appeared as a resistor on multisim.

**Conclusion:**

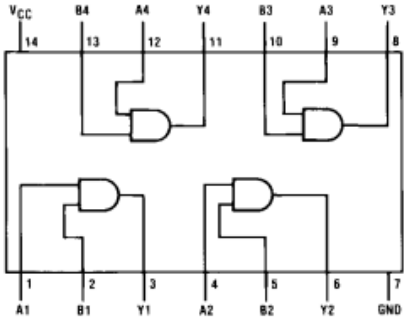
- The selector circuit consists of a counter to allow you to choose your number, four LEDs to show your displayed number, four XNOR gates to compare the number selected by the roulette number with the number you selected, which has those four outputs going into two AND gates, with those two outputs going to another AND gate to make sure all four bits are the same for both numbers.
- Using a 555 timer, we made a decaying circuit that would increase the time between pulses overtime. The capacitors would charge and discharge overtime, causing the highs and lows. There are two of these circuits in the roulette wheel, one for the spinning of the wheel and another for the random delay for the spinning.
- To make it so we can both spin and reset the roulette wheel, we used an SR flip-flop. This allowed us to set, or start, the spinning of the wheel, and reset the wheel so it can spin again after it has stopped.

- Using a MOD-16 counter, we could count from 0-15. The outputs from these counters would go into XNOR gates, then through two AND gates with the outputs of those going to another AND gate, which then goes to the 555 signal decay circuit, causing a random decay at some point in time during the process of counting. At the same time, these counters are connected to the demultiplexers, which are then connected to the 16 LEDs, lighting them up one by one as the counter counts from 0-15.

**Datasheets:**

☐ **7408 Datasheet AND**

**Connection Diagram**



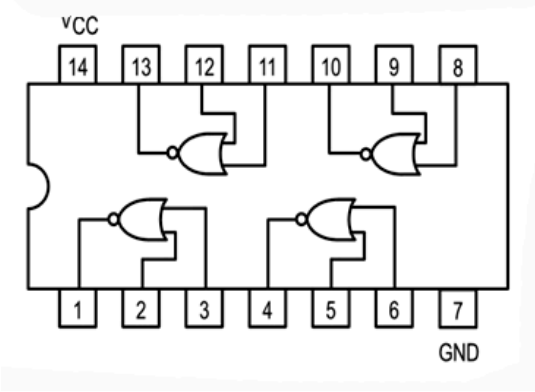

**Function Table**

$Y = AB$

| Inputs | | Output |
|---|---|---|
| A | B | Y |
| L | L | L |
| L | H | L |
| H | L | L |
| H | H | H |

H = HIGH Logic Level
L = LOW Logic Level

☐ **7402 Datasheet NOR**



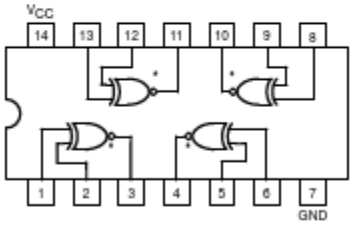

**Function Table**

$Y = \overline{A + B}$

| Inputs | | Output |
|---|---|---|
| A | B | Y |
| L | L | H |
| L | H | L |
| H | L | L |
| H | H | L |

H = HIGH Logic Level
L = LOW Logic Level

☐ **74266 Datasheet XNOR**

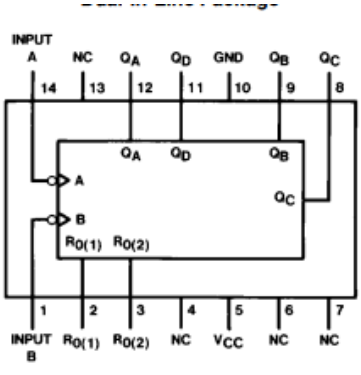



*OPEN COLLECTOR OUTPUTS

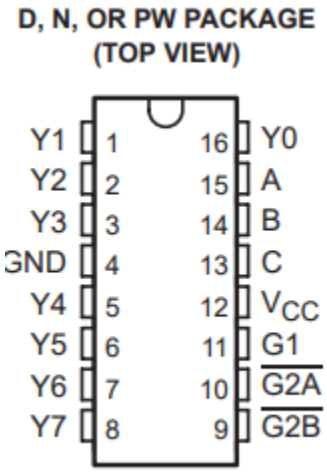
**TRUTH TABLE**

| IN | | OUT |
|---|---|---|
| A | B | Z |
| L | L | H |
| L | H | L |
| H | L | L |
| H | H | H |

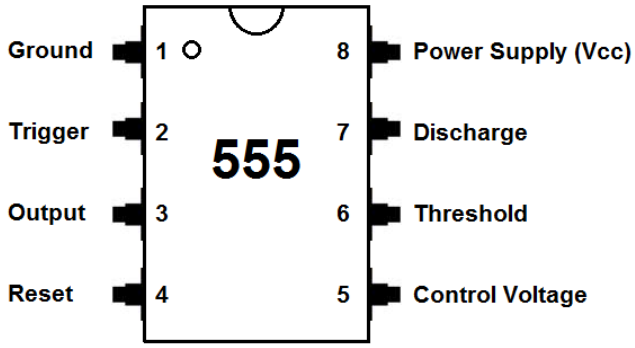
☐ **7493 Counter Datasheet**



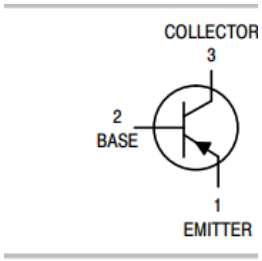

☐ **74138 Demultiplexer Datasheet**

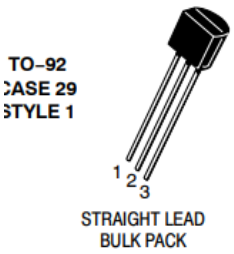



☐ **555 Timer Datasheet**



## ☐ 2N3906 Transistor Datasheet




## ☐ 1N4001 Diode

| **MAXIMUM RATINGS** ($T_A$ = 25 °C unless otherwise noted) | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **PARAMETER** | | **SYMBOL** | **1N4001** | **1N4002** | **1N4003** | **1N4004** | **1N4005** | **1N4006** | **1N4007** | **UNIT** |
| Maximum repetitive peak reverse voltage | | $V_{RRM}$ | 50 | 100 | 200 | 400 | 600 | 800 | 1000 | V |
| Maximum RMS voltage | | $V_{RMS}$ | 35 | 70 | 140 | 280 | 420 | 560 | 700 | V |
| Maximum DC blocking voltage | | $V_{DC}$ | 50 | 100 | 200 | 400 | 600 | 800 | 1000 | V |
| Maximum average forward rectified current 0.375" (9.5 mm) lead length at $T_A$ = 75 °C | | $I_{F(AV)}$ | 1.0 | | | | | | | A |
| Peak forward surge current 8.3 ms single half sine-wave superimposed on rated load | | $I_{FSM}$ | 30 | | | | | | | A |
| Non-repetitive peak forward surge current square waveform $T_A$ = 25 °C (fig. 3) | $t_p$ = 1 ms | $I_{FSM}$ | 45 | | | | | | | A |
| | $t_p$ = 2 ms | | 35 | | | | | | | |
| | $t_p$ = 5 ms | | 30 | | | | | | | |
| Maximum full load reverse current, full cycle average 0.375" (9.5 mm) lead length $T_L$ = 75 °C | | $I_{R(AV)}$ | 30 | | | | | | | μA |
| Rating for fusing (t < 8.3 ms) | | $I^2t$ (1) | 3.7 | | | | | | | $A^2s$ |
| Operating junction and storage temperature range | | $T_J$, $T_{STG}$ | -50 to +150 | | | | | | | °C |