

**Activity 1.2.1 Combinational Logic Design**

Introduction

Combinational and sequential logic are the fundamental building blocks of digital electronics. Combinational logic, which is sometimes referred to as "combinatorial logic”, is characterized by its output being a function of the current input value.

A variety of different logic gates can be used to implement combinational logic circuits. Many of these gates will be studied in future units of this course. In this introductory unit, we will limit our designs to AND, OR, and INVERTER gates for the sake of simplicity.

In this activity you will use the Circuit Design Software (CDS) to build and test your first combinational logic circuits.

Equipment

* Circuit Design Software (CDS)

Procedure

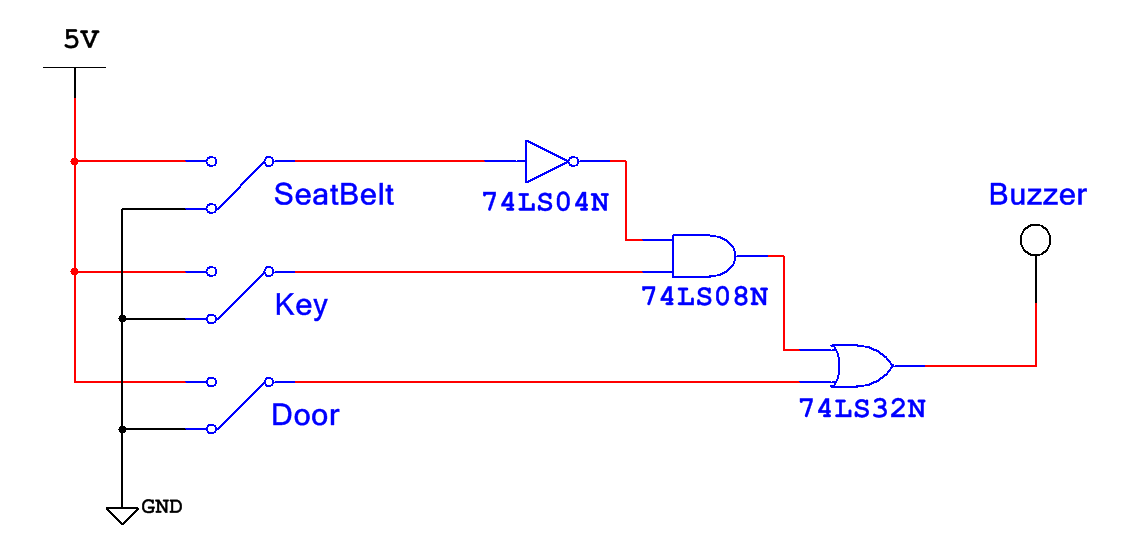
Now it’s time for you to implement your first AOI combinational logic circuit. The circuit that we will use for this purpose is a Car Safety Buzzer design.The design specifications are as follows:

The buzzer is on whenever the door is open or when the key is in the ignition and the seat belt is not buckled.

1. Create a table that describes these design specification in terms of “highs” (1) and “lows” (0). This is when sensor or indicator is active or not active.

|  |  |
| --- | --- |
| Seat Belt | 0 = |
| 1 = |
| Key | 0 = |
| 1 = |
| Door | 0 = |
| 1 = |
| Buzzer | 0 = **BUZZER** is **OFF** |
| 1 = **BUZZER** is **ON** |

1. Using the Circuit Design Software (CDS), enter the Car Buzzer circuit shown below. Use switches for the inputs ***Seat Belt, Key*** & ***Door*** and a probe for the output ***Buzzer***.



1. Based on the design specification you defined, enter a “1” when the **BUZZER** should be **ON** in the *Expected Buzzer Output* column. Toggle the input switches to complete the *Actual Buzzer Output* column in the truth table shown below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Inputs | | | Buzzer Output | |
| Seat Belt | Key | Door | Actual | Expected |
| 0 | 0 | 0 |  |  |
| 0 | 0 | 1 |  |  |
| 0 | 1 | 0 |  |  |
| 0 | 1 | 1 |  |  |
| 1 | 0 | 0 |  |  |
| 1 | 0 | 1 |  |  |
| 1 | 1 | 0 |  |  |
| 1 | 1 | 1 |  |  |

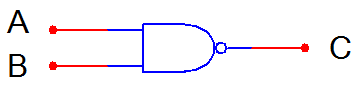
If the Actual Buzzer Output column matches the Expected Output column, then your first combinational logic circuit works. Congratulations! Print a copy of your circuit and put it in your engineering notebook. If the Actual Output does not match the Expected Output, review your circuit diagram and the design specifications that you defined. Make any necessary revision and retest.

**Conclusion**

1. Combinational logic circuits surround us everywhere in our daily lives. Identify 3-5 examples of circuits that contain combinational logic that you interact with almost daily.
2. In this activity we used switches for the circuit inputs and a probe for the circuit output. Though this works fine for testing purposes, it does not reflect the actual sensors and indicator used in real-world applications of combinational logic circuits. List three input and three output devices that could be used with real world applications of combinational logic.

**Going Further – Optional**

As mentioned in the purpose section of this activity, combinational logic circuits can be implemented with a variety of different logic gates. One such gate is introduced in the previous lesson and is called the NAND gate (see below). Functionally, the NAND gate is an AND gate with its output inverted.



Using what you know about the AND gate and INVERTER gates, complete the truth table for the NAND gate.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | **A** | **B** | **C** | | 0 | 0 |  | | 0 | 1 |  | | 1 | 0 |  | | 1 | 1 |  | |  |

What does a NOR gate look like? What do you think it’s truth table would look like? (NAND and NOR gates are sometimes referred to as “Universal Gates”. We will find out why later.