ELT-217, **Experiment AD-3**: Logical Implementation of Comparator Using NAND gates

*EQUIPMENT NEEDED:*
    Analog Discovery unit and Computer
    7400, quad 2-input NANDs
    7410, triple 3-input NANDs
    7404 Hex inverters
    Breadboard and wire as needed

Note: **Before** wiring each circuit you must have a neat logic diagram showing pin numbers used
    and functional interconnections. (This circuit diagram must be included in your report.) **It is
    strongly recommended that you arrange all switches and LEDs so that they are in order,
    on the same line, with the MSB at the left and LSB on the right.** Also do not forget to
    hook up power and ground to each IC you use.

*DESIGN OBJECTIVE*

Two 2-bit numbers are to be automatically compared. (See figure 1.) The inputs are four bits $A_1$,
$A_0$, $B_1$, and $B_0$ and are arranged into two 2-bit numbers as follows:

                MSB-LSB
    Number A:  $A_1$  $A_0$
                MSB-LSB
    Number B:  $B_1$  $B_0$
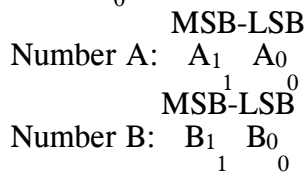
As a result of the comparison process we desire **three distinct simultaneous outputs**. Each of
    these 3 should be implemented by a separate circuit; with the common inputs fed into all
    circuits simultaneously.

        F1 = 1  if Number A equals Number B, zero otherwise

        F2 = 1  if Number A is greater than Number B, zero otherwise

        F3 = 1  if Number A is less than Number B, zero otherwise

```
                   ┌─» |‾‾‾‾‾‾‾‾‾‾‾|
                   │─» |   YOUR    |          O
                   │─» | DESIGN    |─»F1       U
                   └─» |_____|          T
      I                 ‾‾‾‾‾‾‾‾‾‾‾           P
      N    A1─┌─» |                 |          U
      P    A0─│─» |   YOUR          |─»F2       T
      U    B1─│─» | DESIGN          |          S
      T    B0─└─» |_____|
      S                 ‾‾‾‾‾‾‾‾‾‾‾
                   ┌─» |                 |
                   │─» |   YOUR    |─»F3
                   │─» | DESIGN    |
                   └─» |_____|
```
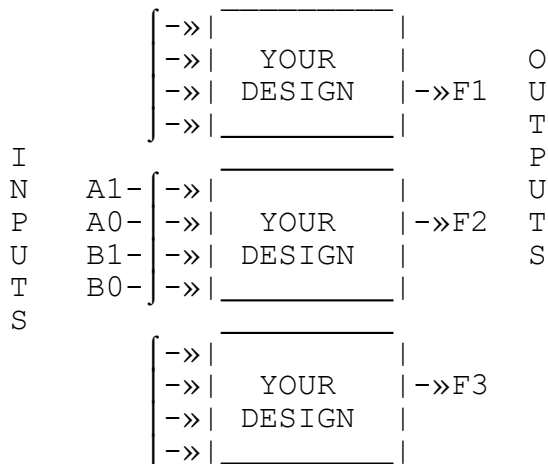
    Figure 1, Design Objectives

Experiment AD-3

*Design Procedure:*

Note: Your lab report should provide full documentation covering the design, construction, and verification of an acceptable solution to the design problem. While several solutions are possible each acceptable solution must provide the required truth table for the three separate outputs F1, F2, and F3.

(1) Develop the logical expressions for F1, F2, and F3 using a truth table and then simplifying the derived Boolean expressions. A suggested form of the truth table is given in figure 2. (Show this process clearly in your report.)

(2) Draw the SOP (Sum of Products) logic diagram for each of the three outputs using AND, OR, and INVERTERS. Note that you can derive the third output from the other two outputs if it would simplify the circuits.

(3) Redraw the logic using only NAND gates (and inverters); use only:
7404, Inverters
7400, 2-input NANDs or
7410, 3-input NANDs.
Your solution should not require more than a total of **five** (5) ICs of the above types. (The best solution to date required only four ICs.)

(4) Add the pin numbers to the logic diagram and construct your design using four virtual data switches as inputs and three virtual LEDs to indicate the outputs. On your wiring diagram, in addition to the pin numbers, be sure to *clearly* indicate which I/O pins of the AD2 are being used as switches or LEDs.

(5) Test your design by making it and filling in an *observed data* table.

(6) Troubleshoot and/or redesign your circuit if your measured values do not agree with the expected ones! Analyze the results by interpreting the input bits as numbers and comparing the observed values of the outputs with the theoretical truth table you developed in step 1.

(7) Take a "screen shot" of any one set of data and include it in your report. In your report be sure you clearly indicate which set of data is being tested and the meaning of each relevant switch and LED.

Figure 4, Suggested Data Table

| I N P U T S | | | | | | O U T P U T S | | |
|---|---|---|---|---|---|---|---|---|
| *Equivalent Decimal* | | Binary Data Used | | | | F1 | F2 | F3 |
| # A | # B | A1 | A0 | B1 | B0 | (A=B) | (A>B) | (A<B) |
| 0 | 0 | 0 | 0 | 0 | 0 | | | |
| 0 | 1 | 0 | 0 | 0 | 1 | | | |
| 0 | 2 | 0 | 0 | 1 | 0 | | | |
| 0 | 3 | 0 | 0 | 1 | 1 | | | |
| 1 | 0 | 0 | 1 | 0 | 0 | | | |
| 1 | 1 | 0 | 1 | 0 | 1 | | | |
| 1 | 2 | 0 | 1 | 1 | 0 | | | |
| 1 | 3 | 0 | 1 | 1 | 1 | | | |
| 2 | 0 | 1 | 0 | 0 | 0 | | | |
| 2 | 1 | 1 | 0 | 0 | 1 | | | |
| 2 | 2 | 1 | 0 | 1 | 0 | | | |
| 2 | 3 | 1 | 0 | 1 | 1 | | | |
| 3 | 0 | 1 | 1 | 0 | 0 | | | |
| 3 | 1 | 1 | 1 | 0 | 1 | | | |
| 3 | 2 | 1 | 1 | 1 | 0 | | | |
| 3 | 3 | 1 | 1 | 1 | 1 | | | |

11/6/17-dw