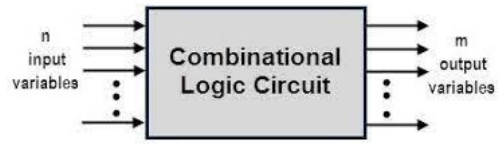
PHY 307: Electronics-II

Tutorial-6

Combinational Logic Circuits

## Combinational Logic Circuits:




Combinational Circuit

Combinational (combinatorial) circuits realize Boolean functions and deal with digitized signals, usually denoted by 0s and 1s. The behavior of a combinational circuit is memoryless, given a stimulus to the input of a combinational circuit, a response appears at the output after some propagation delay, but the response is not stored or fed back. The output of these circuits depends solely on its most recent input and is independent of the circuit's past history.

Design of a combinational circuit begins with a behavioral specification and selection of the implementation technique. These are then followed by simplification, hardware synthesis, and verification. Combinational circuits can be specified via Boolean logic expressions, structural descriptions, or truth tables. Combinational circuits implemented with fixed logic tend to be more expensive in terms of design effort and hardware cost, but they are often both faster and denser and consume less power.

They are suitable for high-speed circuits and/or high-volume production. Implementations that use memory devices or programmable logic circuits, on the other hand, are quite economical for low-volume production and rapid prototyping, but may not yield the best performance, density, or power consumption. Simplification is the process of choosing the least costly implementation from among feasible and equivalent implementations with the targeted technology. For small combinational circuits, it might be feasible to do manual simplification based on manipulating or rewriting logic expressions in one of several equivalent forms. In most practical cases, however, automatic hardware synthesis tools are employed that have simplification capabilities built in. Such programmed simplifications are performed using a mix of algorithmic and heuristic transformations. Verification refers to the process of ascertaining, to the extent possible, that the implemented circuit does in fact behave as originally envisaged or specified.

A few popular combinational circuits that are more complex than single gates include,

Adder circuits, Multiplexers, Parity generators, Decoders, and Encoders.

PHY 307: Electronics-II

Tutorial-6

K-map, Combinational Logic Circuits

1. **Simplification of Boolean expression using K-map:** Simplify the following Boolean expressions,

   *(i).* $f=\Sigma(0,1,4,6,8)+d(3,12,14)$, *(ii).* $f=\Sigma(1,2,4,)+d(3,5,6,7,11,13,14,15)$.

2. **Alarm circuit using Boolean expression:** The ALARM output is 1 if the PANIC input is 1, or if the ENABLE input is 1, the EXITING input is 0, and the house is not sure; the house is secure if the WINDOW, DOOR, and GARAGE inputs are all 1. Find a suitable Boolean expression and draw a combinational circuit.

3. **Course pass or fail using Boolean expression:** There are three components in a course: homework (H), lab (L), and examination (E). You pass the course (P) only if you pass two or more components. Design a combinational circuit for this application.

4. **Two-level AND-OR circuit:** For the given truth table, find a Boolean expression, draw a logic circuit for the same. Express it through two-level AND-OR circuit.

| Row | X | Y | Z | F |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 |
| 2 | 0 | 1 | 0 | 1 |
| 3 | 0 | 1 | 1 | 0 |
| 4 | 1 | 0 | 0 | 0 |
| 5 | 1 | 0 | 1 | 1 |
| 6 | 1 | 1 | 0 | 0 |
| 7 | 1 | 1 | 1 | 1 |

5. **Combinational logic:** Design a circuit that counts the number of 1's present in 3 inputs A, B and C. Its output is a two-bit number $X_1X_0$, representing that count in binary. Assume active-HIGH logic.

6. **Half and Full subtractor:** List the difference and the borrow-out outputs for, (i) the half, and (ii) the full subtractor, for each set of input pulses (figure 1).

(i).

(ii).

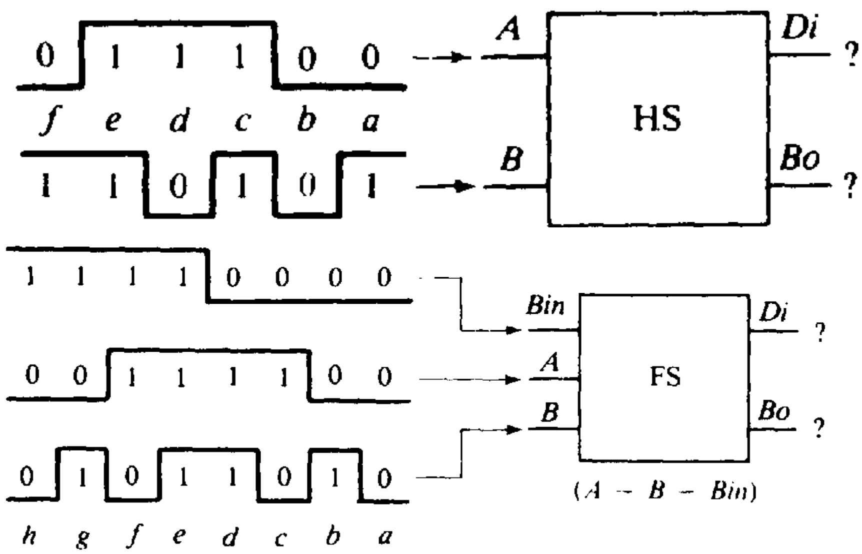



Figure 1