**CS220 Computer Architecture**

**Experiment Reports**

It is our intention to correct your practical work on screen before the end of the session. You should therefore come ON TIME and aim to finish about 20 mins before the end of the session. In some cases, for more difficult practicals where you may not finish with time to spare and may have other classes to go to, it may not be possible to get to everyone and correct them in time.

You should therefore write a report on your activity as you are going along so that you can hand something up to be corrected. The following items should be included.

a. A simple verbal description of the circuit being constructed.
b. Details of a formal circuit design. Provide any tables, K-maps or diagrams used in deriving the circuit input/output equations and a demonstration of how the equations were derived.
c. A logic schematic for the circuit. This can be printed from the TQGate application but a hand drawn schematic will also suffice. You may also use the Ubuntu Screenshot application to take a snapshot of your circuit on the screen.
d. Verify that the circuit worked according to the state table/functional equations. This is done by observing a simulation and testing a circuit under various input conditions and noting any comments or results of these observations.

A report is expected to be ready for examination towards the end of each practical session.

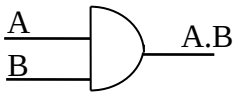
**Example Experiment Report**

a. To verify the operation of an AND gate.

b. An AND gate operates according to the following truth table.

| A | B | A.B |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

c. Logic Schematic




d. Verification of Experiment and Observations
The circuit worked in accordance with the truth table for all input combinations.

**CS220 Computer Architecture**
**Digital Logic Design Practicals**

## Practical 1

Please read the introduction to the TQGate Simulator before beginning this practical.

Part A. Verify the operation of the two input logic gates AND, OR and NOT.

Part B. Construct a circuit which functions as an EXOR gate using only AND, OR and NOT gates.

Part C. Make a circuit composed only of AND and NOT gates which functions as a two input OR gate. Note the equality below.

$$A + B = \overline{\overline{A}.\overline{B}}$$

The inputs to all the above circuits should be provided from I/O switches and the outputs of all circuits should be connected to an I/O LED device. By default the LED shows bright RED when representing logic 1 and light pink when representing logic 0.