**CSCE 337 - Digital Design II**

**Dr. Mohamed Shalan**

**GTA: Marwan Adel**

**Project 2: Static-Timing-Analysis**

**By:**

**Yehia Hesham**

**Ahmad Abdalla**

**Tasneem Ismail**

**Overall Progress:**

* We used the Gate-Level Netlist parser and the DAG structure from the previous project, after doing modifications.
* We were able to parse all files needed for the timing analysis. That includes: Net Capacitance file, Clock Skew file, Initial Input transition file, Clock period and input delay file.
* We parsed the Liberty file and were able to extract all needed values from the tables (Gate delay, output transition and pin capacitance) using Interpolation and Extrapolation.
* We were able to extract full Paths from every input, passing by the respective pins, to every output, and identified all the needed timing paths (REG-REG, IN-REG , IN-OUT & REG-OUT).
* We were able to get the output capacitance on the output of every gate and input transition of every input pin of every gate.
* We can then get the Cell delays and output transitions for all gates.
* We were able to create functions that do timing analysis for the paths, sort them based on their slacks and display a table for each path describing the combinational delay, arrival time, required time and slack.
* We created 6 test cases with different complexities. But, we only had time to do analysis by hand for 3 of them.

**Problems:**

* We admittedly wasted time trying to integrate the liberty parser with the other data structures and class.
* We wasted even more time getting around the code and how to use the needed functions (particularly the getCellDelay() and the Cell\_Set\_Hold() functions).
* We have 2 unsolved problems now, which are identifying and calculating the critical paths in each timing path.
* That is because we discovered at a very late phase that we had to change the structure of the Directed Acyclic Graph to account for pins in our timing paths.
* Moreover, the identification of the timing paths that was initially working and was displayed in the first demo is now faulty when iterating across wire branching (several input pins connected to the same wire).
* Since we were not able to complete the critical section part, we assumed that the data structure containing the timing paths was available with all the data acquired from the critical path analysis, and created functions to do timing analysis for all the paths and display the results.