EEL3712L - Logic Design I Lab
One Credit, Two and a half hours, Engineering Topic.

Instructor: Dr. Nonnarit O-Larnnithipong.

Textbook: Open source materials are used as instruction materials

Specific Course Information:
Laboratory experiments, using gates, combinational networks, SSI, MSI, LSI. Sequential logic design.

Specific Goals for the Course

a. Specific outcomes of instruction
Upon successful completion of this course, the student will:
1. Develop truth tables.
2. Design combinational and sequential logic circuits using Xilinx Vivado
3. Implement their design on to the target FPGA platform.
4. Verify the characteristics of logic gates and combinational logic circuits.

b. Explicitly indicate which of the student outcomes listed in Criterion 3 or any other outcomes are addressed by the course.
In this course the student will have to show
(a) an ability to apply knowledge of mathematics, science, and engineering (X)
(b) an ability to design and conduct experiments (simulations), as well as to analyze, interpret data (N/A)
(c) an ability to design a system, component, or process to meet desired needs (N/A)
(d) an ability to function in multi-disciplinary teams (N/A)
(e) an ability to identify, formulate, and solve engineering problems (homework) (X)
(f) an understanding of professional and ethical responsibility (N/A)
(g) an ability to communicate effectively (through project reports) (N/A)
(h) the broad education necessary to understand the impact of engineering solutions in a global and societal context (N/A)
(i) a recognition of the need, and an ability to engage in life-long learning (N/A)
(j) a knowledge of contemporary issues (N/A)
(k) an ability to use the techniques, skills, and modern engineering tools necessary for engineering practice (N/A)
(l) a knowledge of probability and statistics (N/A)

Brief list of the topics to be covered
1. AND Gates & OR Gates
2. Inverting Logic: NOT, NAND, & NOR
3. Boolean Laws & Rules and DeMorgan’s Theorem
4. XOR and XNOR Gates with Applications
5. Use of XOR/XNOR Gates to Generate & Check Parity
6. Binary Adders
7. Decoders & Applications
8. Encoder & Application to a 7-Segment Display Driver
9. Multiplexer & De-Multiplexer

**GRADING:**

<table>
<thead>
<tr>
<th>Course Requirements</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lab Reports</td>
<td>60%</td>
</tr>
<tr>
<td>Final Exam</td>
<td>20%</td>
</tr>
<tr>
<td>Midterm Exam</td>
<td>20%</td>
</tr>
<tr>
<td>Overall Grade</td>
<td>100%</td>
</tr>
</tbody>
</table>

**Conversion of Numerical Grade to Letter Grade**

<table>
<thead>
<tr>
<th>95&lt;=A&lt;=100</th>
<th>83&lt;=B&lt;86</th>
<th>70&lt;=C&lt;76</th>
</tr>
</thead>
<tbody>
<tr>
<td>90&lt;=A&lt;94</td>
<td>80&lt;=B&lt;82</td>
<td>60&lt;=D&lt;69</td>
</tr>
<tr>
<td>87&lt;=B+&lt;89</td>
<td>77&lt;=C+&lt;79</td>
<td>F: Below 60</td>
</tr>
</tbody>
</table>