1. Arithmetic Circuits

(a) Extend the basic 4-bit CLA to a 5-bit CLA. For this give the equations for the single-bit (small) \(p_i, g_i\)’s, the equations for the big \(P, G\) going out of the 5-bit CLA (this is the \(P, G\) generation unit), and the equations for generating the carries to modified full adders # 2-5 (this is the carry-generation unit).

For your reference the first 3 carry equations of a 4-bit CLA (there are 4 carries \(c_1, \ldots, c_4\) generated in a 4-bit CLA) are:

\[
c_1 = g_0 + p_0c_0
\]

\[
c_2 = g_1 + p_1c_1 = g_1 + p_1(g_0 + p_0c_0) = g_1 + p_1g_0 + p_1p_0c_0
\]

\[
c_3 = g_2 + p_2c_2 = g_2 + p_2g_1 + p_2p_1g_0 + p_2p_1p_0c_0
\]

(b) Use the above 5-bit CLA (as many as required), and a top-level 4-carry-generation unit (a unit that generates 4 carries) to design a 20-bit tree CLA as follows:

(i) Give the equations of the 4-carry-generation unit (the inputs to this unit are the \(P, G\)’s coming from the 5-bit CLAs you are using, and the carry-in \(C_{in}\)).

(ii) Draw the schematic of the 20-bit tree CLA with the above components showing the the various clearly labeled inputs and outputs of the components and their interconnections.

(c) Clearly derive the worst-case delay of your 20-bit tree CLA assuming that a \(k\)-input AND/OR/NOT/XOR/XNOR gate has a delay of \(k\) units.
2. **State Equivalency**

For the FSM shown below, determine which state pairs are equivalent and give the proof of equivalency of one of these state pairs (the proofs for the other pairs, if any, will follow by symmetry for this case) using the basic definition of state equivalency. Start your proof by clearly stating this basic definition of state equivalency.
3. **State Minimization of Incompletely Specified Machines**

(a) Define an *applicable input sequence* for a state $A$ in an incompletely specified machine. Give an example of an applicable input sequence and an inapplicable input sequence (you need to start your example with an appropriate state transition table).

(b) Give the basic definition of compatibility between two states $A, B$.

(c) For the following portion of a state transition table, specify what the entry will be for the square corresponding to state pair $(A, C)$ in the implication table.

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A$</td>
<td>B/0</td>
<td>--/--</td>
</tr>
<tr>
<td>$C$</td>
<td>D/0</td>
<td>E/0</td>
</tr>
</tbody>
</table>

Give clear reasonings for your answer.

**Hint:** Utilize the concept of applicable input sequence as used in the basic definition of state compatibility.
4. Moore Machine and One-Hot Design

For the Mealy FSM shown below do the following:

(a) Obtain the corresponding Moore FSM.
(b) Obtain the 1-hot design of this Moore FSM using D-FFs.
(c) Determine the delay of the combinational logic portion of your 1-hot design assuming that a \( k \)-input AND/OR/NOT gate has a delay of \( 4k \) ns.