2. Deriving a Boolean Expression from a Truth Table

2.1 Introduction

We have been looking at several different representations for digital logic circuits:

- Circuit diagrams
- Truth tables
- Boolean algebra

We have also started to consider how we translate between these representations (in Lab Two we translated Boolean algebra expressions to circuit diagrams and vice versa). Today we will look at an approach to converting a truth table into a Boolean algebra expression. In a week's time we will look at the method by which we can use the laws of Boolean algebra to simplify the expressions we derive.

2.2 Sum of Products

Here is a truth table.

A	B	X
0	0	0
0	1	0
1	0	1
1	1	0

To derive the Sum of Products expressions we must look at the rows in the table where the output is 1. Clearly in this case there is only one such row, namely the one where A=1 and B=0. We may say that "X=1 if A=1 and B=0". We translate this into Boolean algebra notation exactly as you might expect: $X = A.\overline{B}$. In cases where there are more than one 1s in the output column, we use an OR in the expression:

A	B	X
0	0	1
0	1	0
1	0	0
1	1	1

Here the expression is "X=1 if A=0 and B=0 or A=1 and B=1", or $X = \overline{A}.\overline{B} + A.B$

2.3 Exercises

1. Derive the Sum of Products expression from the truth table below.

A	B	X
0	0	1
0	1	1
1	0	0
1	1	0

2. Derive the Sum of Products expression from the truth table below.

A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

2.4 Three Input Truth Tables

Three input truth tables are treated just the same.

A	B	C	Z
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

The expression would be $Z = \overline{A}.\overline{B}.C + A.B.\overline{C} + A.B.C$

2.5 Exercises

3. Derive the Sum of Products expression from the truth table below.

A	B	C	Z
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

4. Derive the Sum of Products expression from the truth table below.

A	B	C	Z
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

2.6 Combinational Circuit Exercises

5. Refer back to the lecture notes and consider the truth table for the 2 to 4 Decoder. Derive expressions for the outputs, draw the resulting expressions in LogiSim and compare with the circuit in the lecture.

6. Refer back to the lecture notes and consider the truth table for the half adder. Derive expressions for the sum and carry outputs, draw the resulting expressions in LogiSim and compare with the circuit in the lecture.

7. In Logisim draw the circuit $X = \overline{A \oplus B}$ and test its operation until you are certain you understand how it works.

8. Draw a truth table for the circuit.

9. Under what circumstances does this circuit output a 1?

10. What is the name of this circuit and what is its purpose?

11. In Logisim draw the circuit $Y = A.\overline{C} + B.C$ and test its operation until you are certain you understand how it works.

12. Draw a truth table for the circuit.

13. Through examining the truth table you should be able to see that the circuit always outputs certain inputs, depending on the value of one particular input. What can be truth table therefore be reduced to?

14. What is the name of this circuit and what is its purpose?

Extension Work

This example is based on a four-input truth-table that defines a *2x2 bit multiplier*. A_1A_0 represents the first operand. B_1B_0 represents the second. The table below shows all the numbers that can be represented using two binary inputs. Naturally we could draw an identical table for B_1B_0.

A_1	A_0	Value
0	0	0
0	1	1
1	0	2
1	1	3

To determine the output we compute A_1A_0 x B_1B_0. We need to consider every combination of A_1A_0 and B_1B_0, so we need four inputs.

We need four outputs because the highest number that can be output is 9 (3 x 3 = 9, which is 1001 in binary). The outputs are labelled E, F, T and W in the truth-table and these stand for Eights, Fours, Twos and Ones. So, 3 x 3 = 9 and 9 = (1 x 8) + (0 x 4) + (0 x 2) + (1 x 1). Another example: 2 x 3 = 6 and 6 = (0 x 8) + (1 x 4) + (1 x 2) + (0 x 1). The completed truth-table is shown below.

A_1	A_0	B_1	B_0	Calculation	E	F	T	W
0	0	0	0	0 x 0	0	0	0	0
0	0	0	1	0 x 1	0	0	0	0
0	0	1	0	0 x 2	0	0	0	0
0	0	1	1	0 x 3	0	0	0	0
0	1	0	0	1 x 0	0	0	0	0
0	1	0	1	1 x 1	0	0	0	1
0	1	1	0	1 x 2	0	0	1	0
0	1	1	1	1 x 3	0	0	1	1
1	0	0	0	2 x 0	0	0	0	0
1	0	0	1	2 x 1	0	0	1	0
1	0	1	0	2 x 2	0	1	0	0
1	0	1	1	2 x 3	0	1	1	0
1	1	0	0	3 x 0	0	0	0	0
1	1	0	1	3 x 1	0	0	1	1
1	1	1	0	3 x 2	0	1	1	0
1	1	1	1	3 x 3	1	0	0	1

Draw a circuit in LogiSim which will implement this truth table.