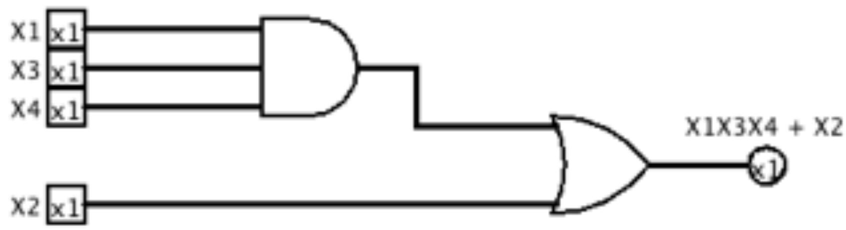
CSC 333 Lab 5

Samuel Jentsch

To begin the project, I implemented the minimized circuit in figure 7.19 to use for testing my implementation of the circuit for table 7.9:

X1	X3	X4	X2	F(X1X3X4X2)
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

Truth table for circuit implement in Logism.

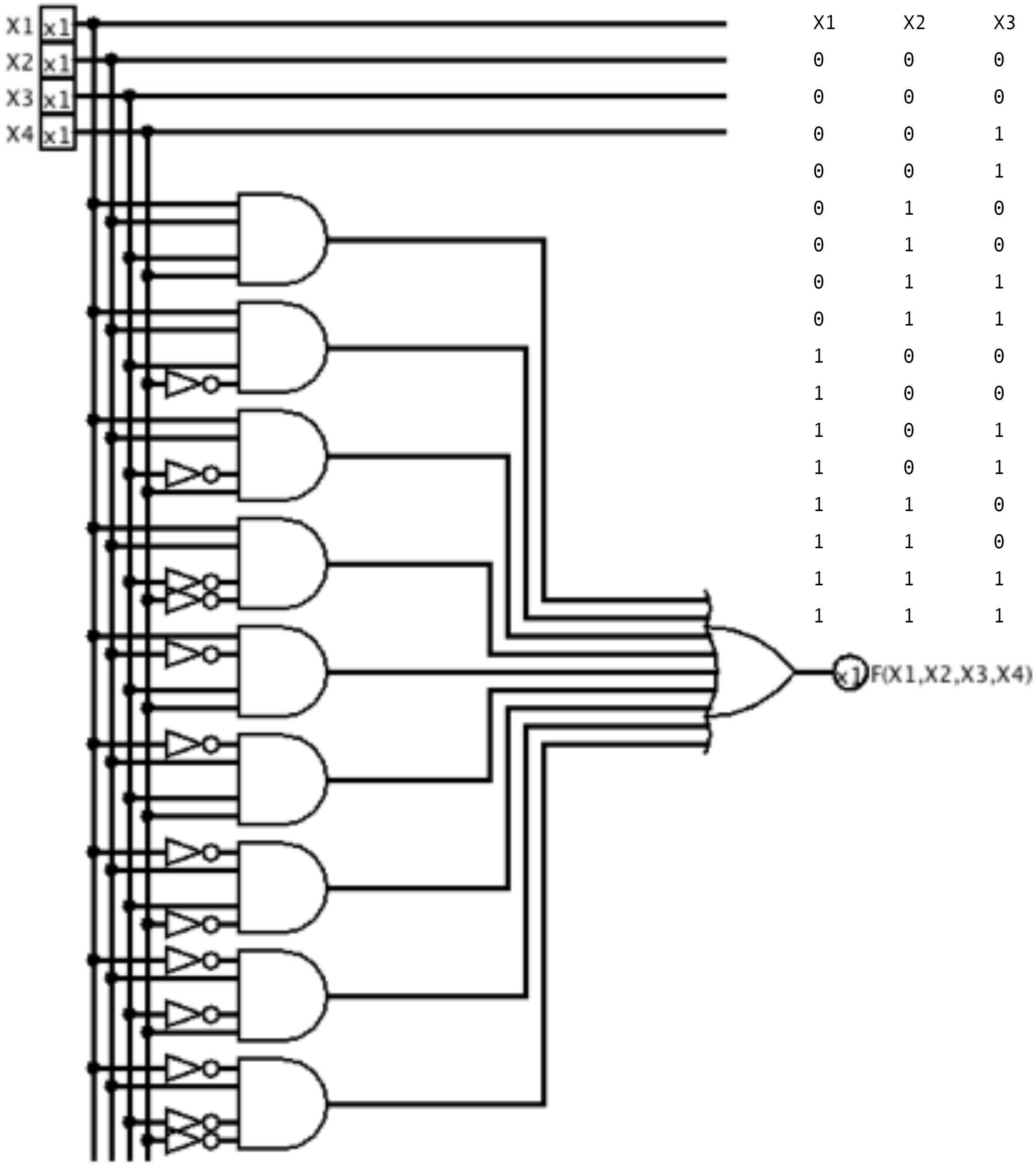


Circuit 7.19 implemented in Logism.

Next, I worked on finding the canonical sum of products for table 7.9 to use as a means of implementing the table as a circuit:

X1X2X3X4 + X1X2X3(X4') + X1X2(X3')X4 + X1X2(X3')(X4') + X1(X2')X3X4 + (X1')X2X3X4 + (X1')X2X3(X4') + (X1')X2(X3')X4 + (X1')X2(X3')(X4')

Following is the circuit implemented directly using the unminimized sum of products and the truth table for the circuit:



Unminimized Circuit implemented in Logism

Truth table for unminimized circuit

X1	X2	X3	X4	FX1X2X3X4
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

To simplify the circuit before implementation to obtain the minimized circuit, I'll use a K-map populated with the results from finding the sum of products form for the truth table:

	X1X2	X1(X2')	(X1')(X2')	(X1')X2
X3X4	1	1		1
X3(X4')	1			1
(X3')(X4')	1			1
(X3')X4	1			1

The yellow squares will wrap around and the square fading into the gray square will be used (overlapped). When there are 8 adjacent squares in a four variable K-map, 3 variables can be eliminated. The 8 yellow squares simplify to: X2. The 2 gray squares simplify to: X1X3X4. So the entire sum of products can be reduced to X1X3X4 + X2. This is the same expression used for the circuit in 7.19, and when implemented in Logism would be identical to the minimized circuit already implemented: