Design Problem

Scanners for item returns are useful for determining whether the item was sold with a discount (and therefore should be refunded at the corresponding price) or if the item was potentially stolen. At Nordstrom, items are tracked by their UPC code, which determines whether the item is on a discount and whether it is an expensive item. Scanning the UPC code will determine whether the item is on discount, but determining whether it is stolen requires an extra input: a marker on expensive items that have been purchased. Therefore, if an expensive item is scanned without a marker, then it must have been stolen. This effectively reveals anyone attempting to return an expensive item that was stolen. This system design takes in four total inputs, three of which being the UPC code combination, and the fourth input being the secret marker placed on purchased expensive items. There are two outputs, one indicating whether the item is on discount or not, and the second output indicating whether the item was stolen or not. Overall, this system will indicate if a scanned item is on discount, and additionally, if it was a stolen expensive item.

Methods

The price scanner was designed using basic logic gates including AND, NAND, OR, and NOR. To implement the design, a truth table was constructed from the description of the assignment, where each combination of inputs has outputs determined to be true or false. Each combination of inputs, which are the different UPC codes, as well as whether the item is marked or not, has two outputs. One output is whether the item is on discount or not, and the second output is whether the item was stolen or not. From the truth table, two Karnaugh Maps were constructed, one for both outputs Discount and Stolen. Each Karnaugh Map has four inputs, being U, P, C, which altogether are the item code, and M, which represents whether the item is marked or not. The data for the Karnaugh Maps was transferred from the truth table, and simplifies the process to find an expression for the system. Ones (true values) in the Karnaugh Maps were grouped together, with the largest possible amount of elements within the smallest number of groups, and a Sum of Products expression was made from the groupings for both Karnaugh Maps, where the expression is built off of the true values in the Karnaugh Map. The expression was then simplified, and a gate-level schematic was constructed to illustrate the design. The gates were then programmed in Verilog in a module at the gate level with an accompanying testbench to test all possible combinations of inputs. The Verilog implementation was then verified with ModelSim using the testbench, before being uploaded into the Altera DE1-SoC Board to further simulate the system.

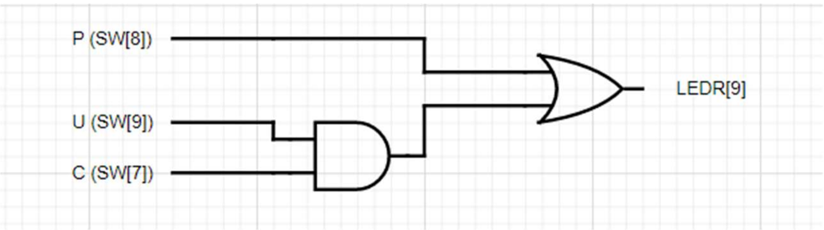
Karnaugh Map for Discounted Output

CM (SW[7]SW[0])		00	01	11	10
UP	00	0	0	X	0
(SW[9]SW[8])	01	X	X	X	X
	11	1	X	X	X
	10	0	0	1	1

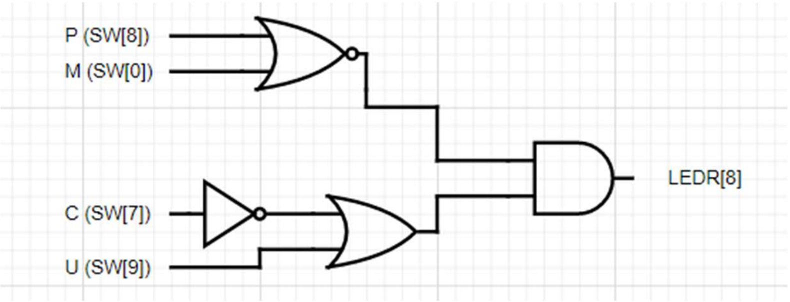
Karnaugh Map for Stolen Output

CM (SW[7]SW[0])		00	01	11	10
UP	00	1	0	X	0
(SW[9]SW[8])	01	X	X	X	0
	11	0	X	X	X
	10	1	0	0	1

Schematic for Discounted Output



Schematic for Stolen Output


P (SW[8])
M (SW[0])
C (SW[7])
U (SW[9])
LEDR[8]