**ESET 219 Digital Electronics**

Laboratory Report

Lab 2

Minterm and Maxterm Logic Design

Ogochukwu Ezuma

9/30/2023

All of the information contained in this report is my own work that I completed as part of this lab assignment. I have not used results or content from any unauthorized sources or fellow students.

Ogochukwu Ezuma Date: 9/30/2023

**Introduction**

In this lab, the development of SOP (sum of products) and POS (Product of sums) circuits were utilized to paraphrase more complex circuits.

**Background**

Binary outputs produced by a circuit can be categorized as either minterms or maxterms. Minterms are the occurrence of 1 (denoted as mn) as an output, maxterms being the occurrence of 0 (denoted as Mn). Both of these outputs have the capacity to culminate into a new circuit that yields the same outputs as the original. For each minterm, multiplying each input that outputs a minterm together, then taking these products’ Boolean sum (ORing them together) gives as SOP circuit. For each Maxterm, summing each input that outputs a Maxterm, then taking these sums’ Boolean product (ANDing them together) gives as POS circuit. SOPs tend to be more readable and intuitive for descaling a complex circuit, regardless both SOP and POS circuits have the same outputs as the parent circuit for each type of input.

**Implementation**

Task one of this lab provided the details of a circuit. It’s inputs are a 4 bit unsigned number and has three outputs, the first output being high if the given number is a prime, second being high when the input number is divisible by 3, and the third output is high if the number is a square number. The objective was to provide a truth table of the main circuit, as well as schematics and simulations of a circuit for P (prime number output) using minterms, S (perfect square output), and Divis3(divisible by three output). First, the main circuit was developed, and each property (output) for each number was assigned the appropriate binary value.

Table 1 Main ciruit truth table

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **A** | **B** | **C** | **D** |  | **P** | **S** | **Divis3** |
| 0 | 0 | 0 | 0 |  | 0 | 1 | 1 |
| 0 | 0 | 0 | 1 |  | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 |  | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 |  | 1 | 0 | 1 |
| 0 | 1 | 0 | 0 |  | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |  | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 |  | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 |  | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 |  | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 |  | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 |  | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |  | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 |  | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 |  | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 |  | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 |  | 0 | 0 | 1 |

To create a circuit for P using minterms, an SOP equation would have to be developed. This was done by multiplying each input that outputs a minterm (value of 1) for P together, then taking these products’ Boolean sum (Oring them together), as seen below.

To create a circuit for S using Maxterms, a POS equation needs to be developed. This was done by summing each input that outputs a Maxterm (value of o) for S, then taking these sums’ Boolean product (ANDing them together), as follows. The length of this POS to Ps SOP was also noted.

The same process which gave an SOP equation for P was then implemented to find Divis3s SOP equation.

For each of these equations, a schematic was developed of them in Quartus as well as their waveform simulations and simulated on the FPGA board.

In task 2 A traffic logic controller has 4 inputs A, B, C, and D. The controller has two outputs, which are the intersection lights. There is an east-west (EW) light and north-south (NS) light. Inputs A and B are a 2-bit binary number with A being the MSB, and C and D are another 2-bit binary number with C being the MSB. If the number AB is greater than the number CD, the EW light is off, and NS is on. If the number CD is greater than the number AB, the NS is off, and the EW is on. If the number AB is equal to number CD, EW light is off, and NS is on.

A truth table of the logic of the controller was constructed.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **A** | **B** | **C** | **D** |  | **EW** | **NS** |
| 0 | 0 | 0 | 0 |  | 0 | 1 |
| 0 | 0 | 0 | 1 |  | 1 | 0 |
| 0 | 0 | 1 | 0 |  | 1 | 0 |
| 0 | 0 | 1 | 1 |  | 1 | 0 |
| 0 | 1 | 0 | 0 |  | 0 | 1 |
| 0 | 1 | 0 | 1 |  | 0 | 1 |
| 0 | 1 | 1 | 0 |  | 1 | 0 |
| 0 | 1 | 1 | 1 |  | 1 | 0 |
| 1 | 0 | 0 | 0 |  | 0 | 1 |
| 1 | 0 | 0 | 1 |  | 0 | 1 |
| 1 | 0 | 1 | 0 |  | 0 | 1 |
| 1 | 0 | 1 | 1 |  | 1 | 0 |
| 1 | 1 | 0 | 0 |  | 0 | 1 |
| 1 | 1 | 0 | 1 |  | 0 | 1 |
| 1 | 1 | 1 | 0 |  | 0 | 1 |
| 1 | 1 | 1 | 1 |  | 0 | 1 |

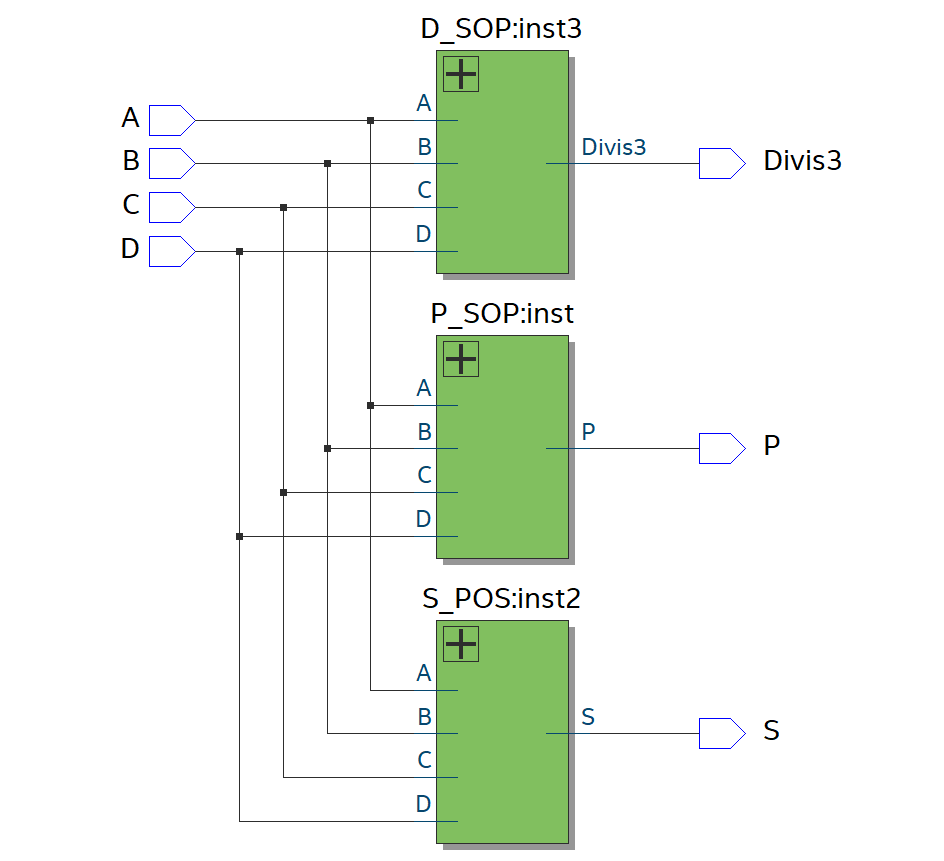
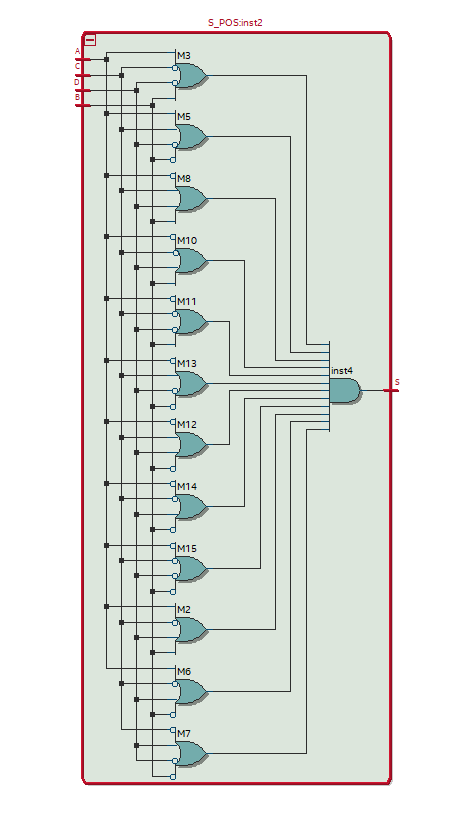
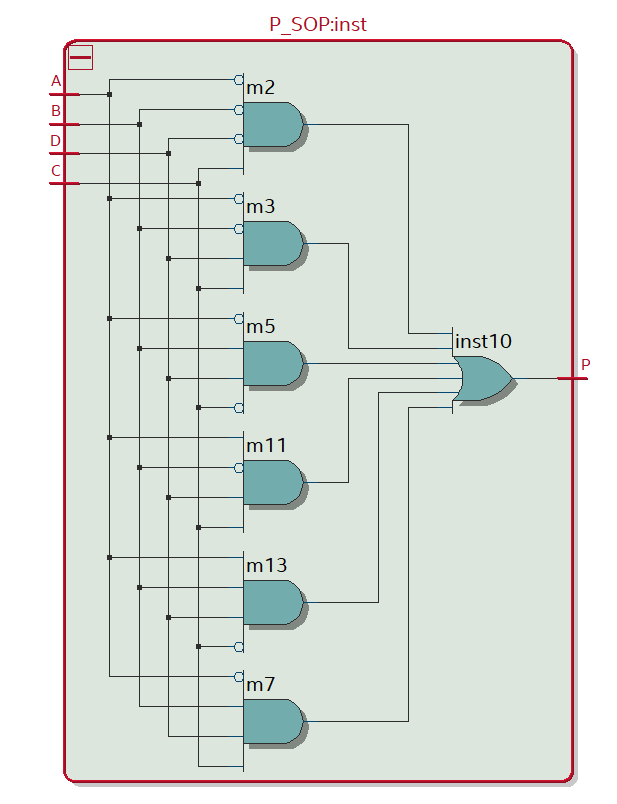
A requirement of this task was to provide a circuit for EW using minterms. The equation below is an SOP circuit for EW, developed through the same process as shown in task 1.

requirement of this task was to provide a circuit for NS using Maxterms. The equation below is an POS circuit for NS, developed through the same process as shown in task 1 just as before.

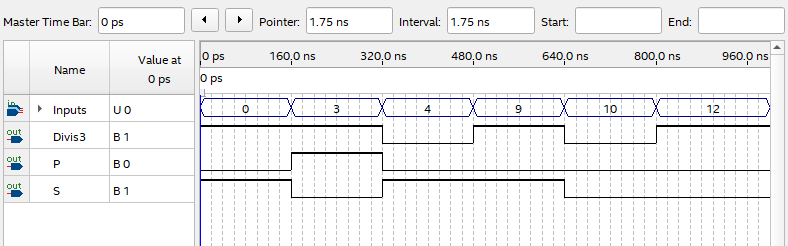
Schematics for this these equations were developed and simulated on the FPGA board.

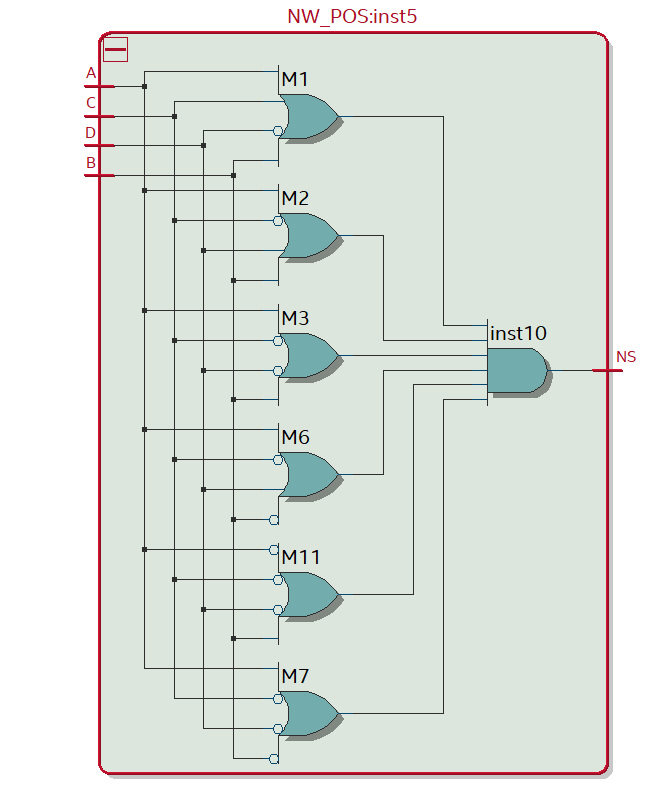
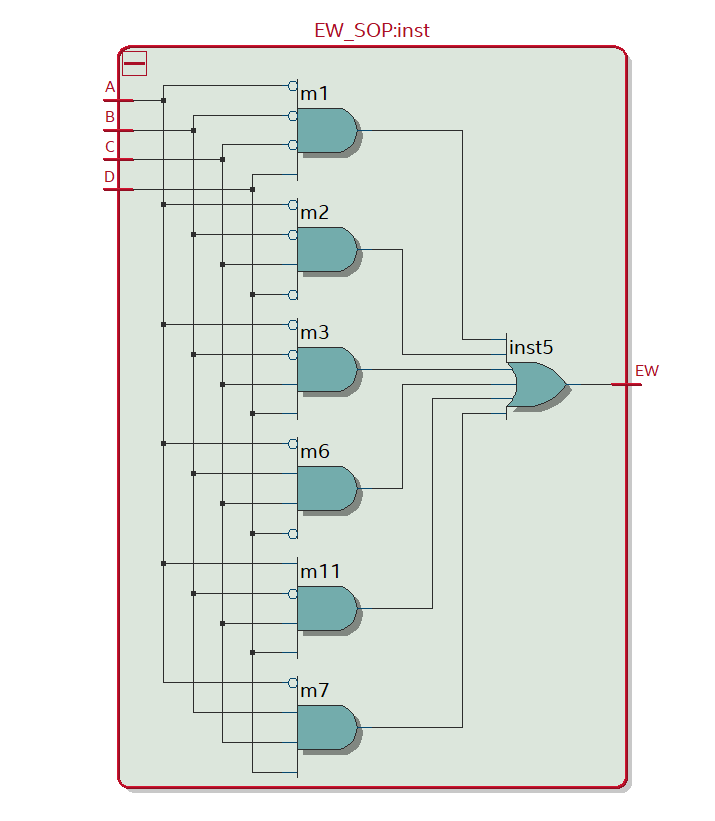
**Results**

A diagram of a computer

Description automatically generatedAll circuits were created in high level block diagrams and inserted into the same bdf file.

The following numbers were to be simulated as well: 0(not prime, square, divisible by 3), 3(prime, not square, divisible by 3),4(not prime, square, not divisible by 3), 9(not prime, square, divisible by 3), 10(not prime, not square, not divisible by 3), 12(not prime, not square, divisible by 3).





**Conclusion**

SOPs and POSs provide a means to paraphrase circuits into smaller designs and equations. Because they have the same outputs, there are no tradeoffs in utilizing such a method of simplification. It should be noted however, that SOPs tend to be far more readable for an outside observer of the simpler circuit design.