Individual post lab assignment PLA1_1 to be discussed next Lab1_2.

1.4.1. Analysis method I: Proteus simulation (virtual laboratory) for truth tables and deducing logic circuits

1.4.1.1. SPICE algorithms

1.4.1.1.1. Proteus ISIS from Labcenter Electronics (EETAC cloud licence available)

1. Specifications

Our goal is to deduce *Circuit_W* truth table W = *f* (D1, D0, A, B) by means of running Proteus simulations. Proteus tutorial and a video rec. on how to proceed with Proteus to deduce a circuit truth table. Circuit's components can be from TTL-LS or CMOS libraries, but do not mix components from different libraries in the same schematic.

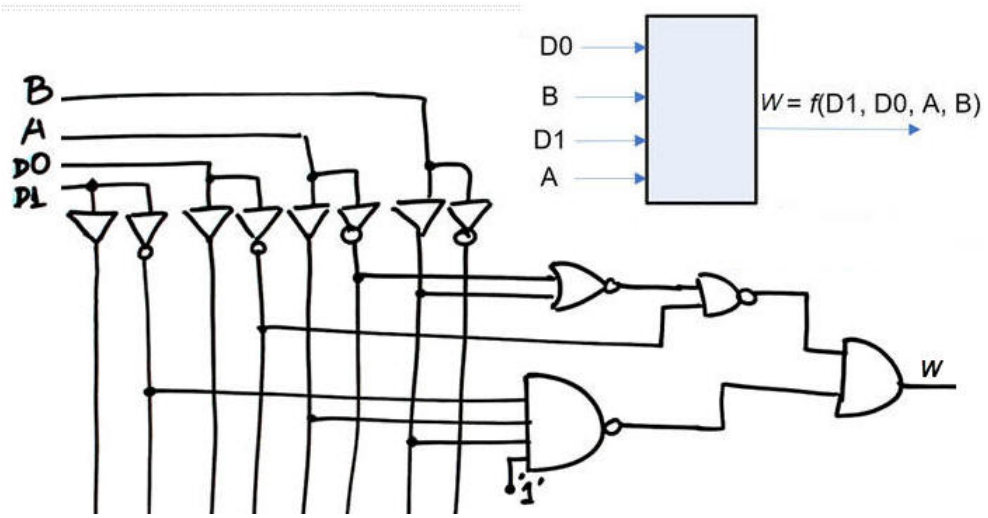


Fig. 1. Symbol and internal architecture of *Circuit_W*.

2. Planning

Draw a plan for this analysis method I. It is possible to model logic gates using CMOS or TTL-LS libraries. Draw all gates using the same library.

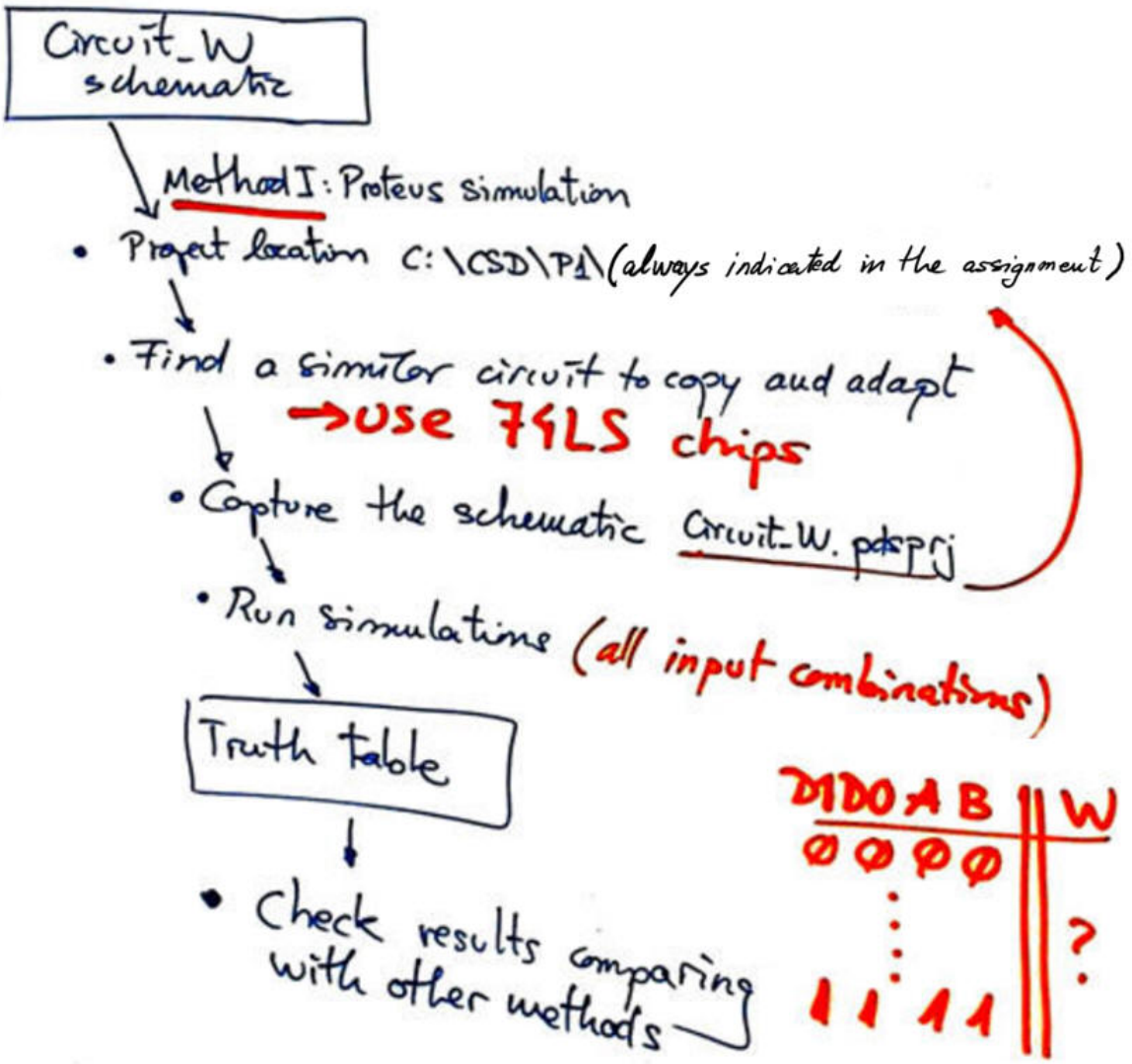


Fig. 2. This sketch explains the plan to follow to deduce the Circuit_W truth table.

This is the project location:

C:\CSD\P1\Lab1_1\Circuit_W\Proteus\(Circuit_W.pdsprj and other files)

3. Development

Select TTL-LS or CMOS 4000 components to build your circuit. For instance: a 3-input NOR gate in classic CMOS 4000 technology is the 4025; the same 3-input NOR in TTL-LS technology is 74LS27.

Use a mouse with your portable computer to make it easier to draw the circuit and navigating through the many tabs.

Find a similar circuit to copy and adapt from this *DIGSYS* web. change its name and save it in the given directory.

In this tutorial we will use TTL-LS (**74LS**) library of classic components.

This is an example Circuit_W.pdsprj capture. Print the captured circuit to demonstrate that you are able to draw circuits as shown in Fig. 3.

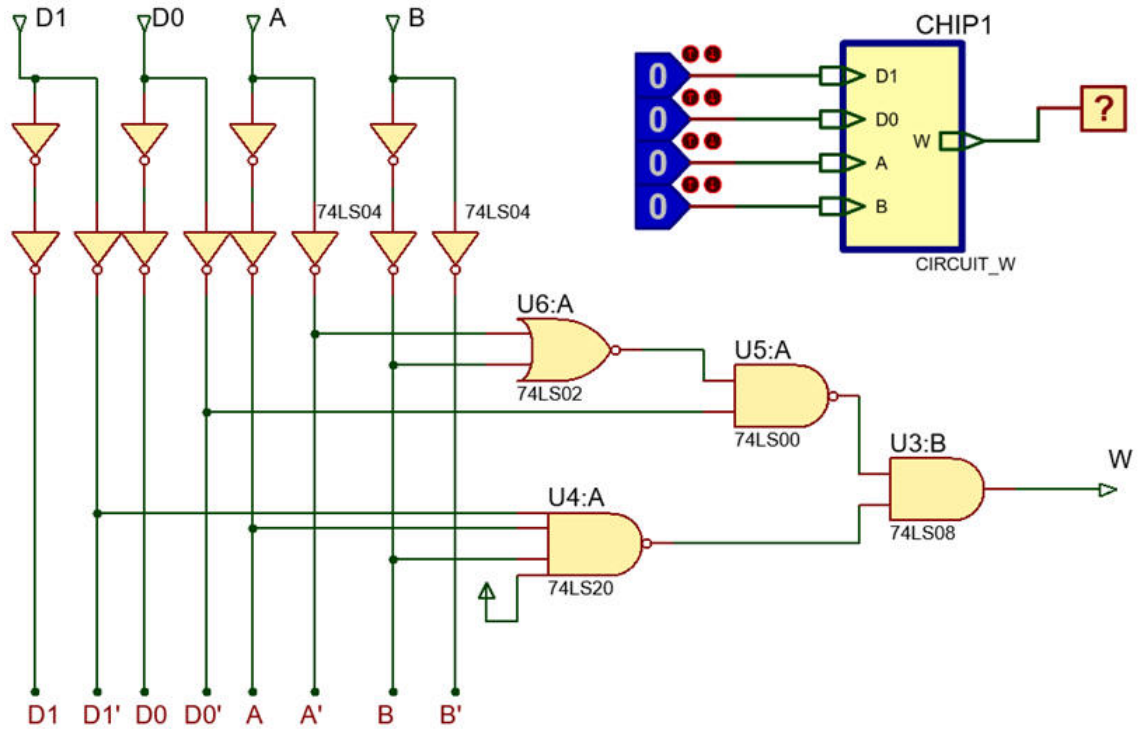


Fig. 3. This sketch explains the plan to follow to deduce the Circuit_W truth table.

Run simulations applying all the binary combinations in order to complete your truth table.

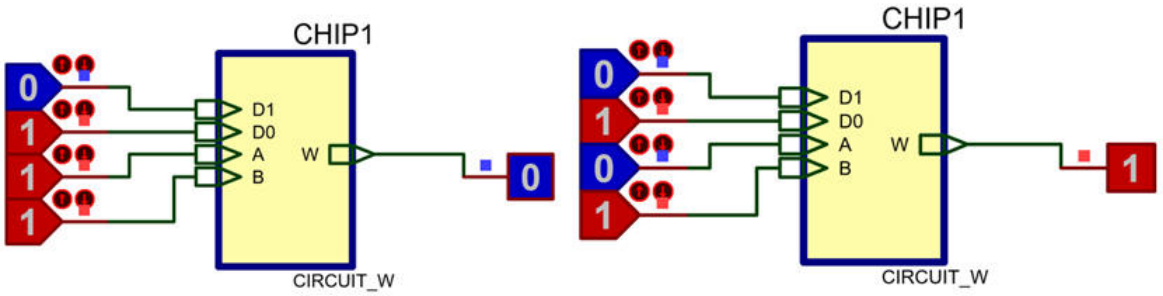


Fig. 4. Simulation results when applying input binary combinations "0111" and "0101". We can observe that "0111" generates the maxterm M_7 and "0101" generates the minterm m_5.

Write down your solution as a truth table or its equivalent canonical equations sum of minterms or product of maxterms.

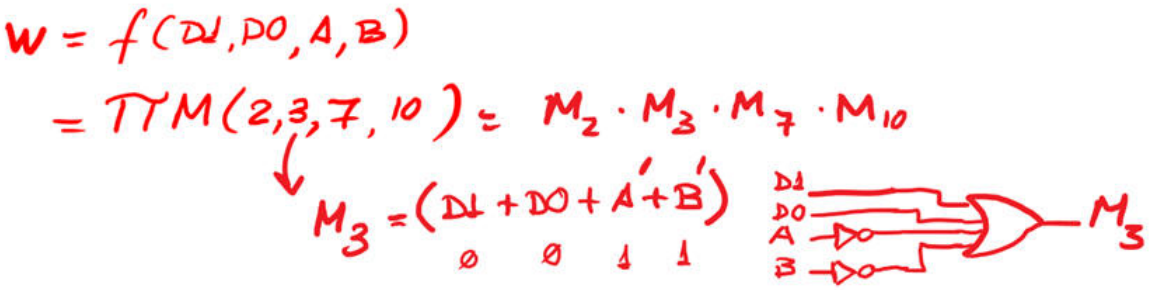


Fig. 5. Problem solution in form of product of maxterms.

4. Testing

Verifying your solutions means self-assessing your results. This is why comparing results from other methods is the best idea. For instance method III solved in class L1.3 or method II to be solved in the project below.

In this way, this verification section has to say something as simple as:

> "My *Circuit_W* analysed using method I (Proteus) generates the same truth table that was obtained using method III".

5. Reporting

- Follow this rubric for writing reports. This is an example that shows how to write a report of a circuit analysis using the method I based on Proteus simulations.

1.4.1.2. (Optional) Digital simulators

1.4.1.2.1. HADES: interactive simulation framework based on JAVA applets (Univ. of Hamburg) (not covered).

1.4.2.1.2. DEEDS: Digital electronics education and design suite (Univ. of Genoa) (not covered). This site promoted by prof. G. Donzellini is spectacular, recommended for learning digital circuits.

Planning	**Laboratory 1_1: Analysis of simple circuits based on logic gates**	Lab1_2
Laboratory	[P1 - Section A] method II: Cirucit equation. WolframAlpha engine	**[22/9]**

1.4.2. Analysisi method II: WolframAlpha numerical engine for calculating truth tables and deducing logic circuits

1. Specifications

Our goal is to deduce *Circuit_W* truth table W = *f* (D1, D0, A, B) using WolframAlpha numerical engine. Video rec. on how to proceed with WolframAlpha.

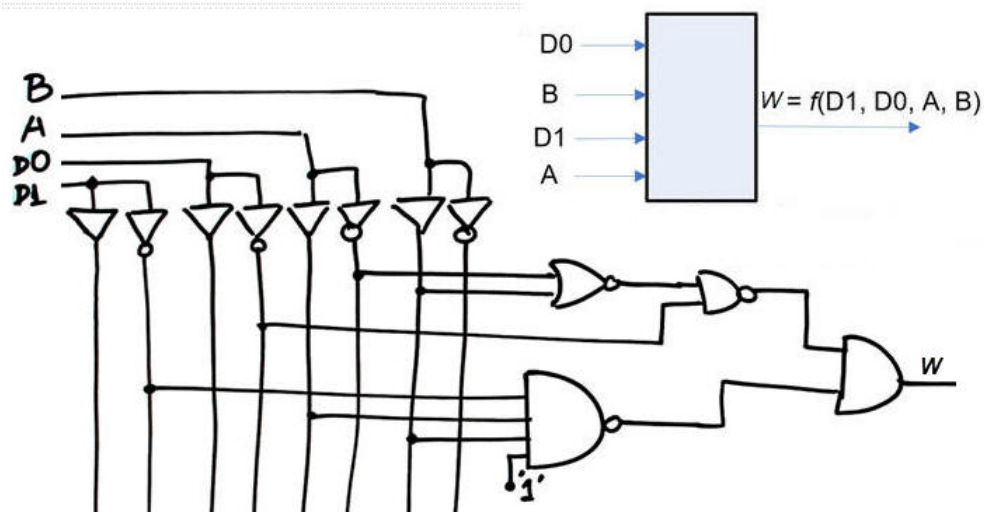


Fig. 1. Symbol and internal architecture of *Circuit_W*.

2. Planning

Draw a plan, a flowchart, a bullet list, for obtaining and checking your solution. For this method, the key point is to input the circuit equation step by step while running the engine to check that there are no errors when interpreting the equation.

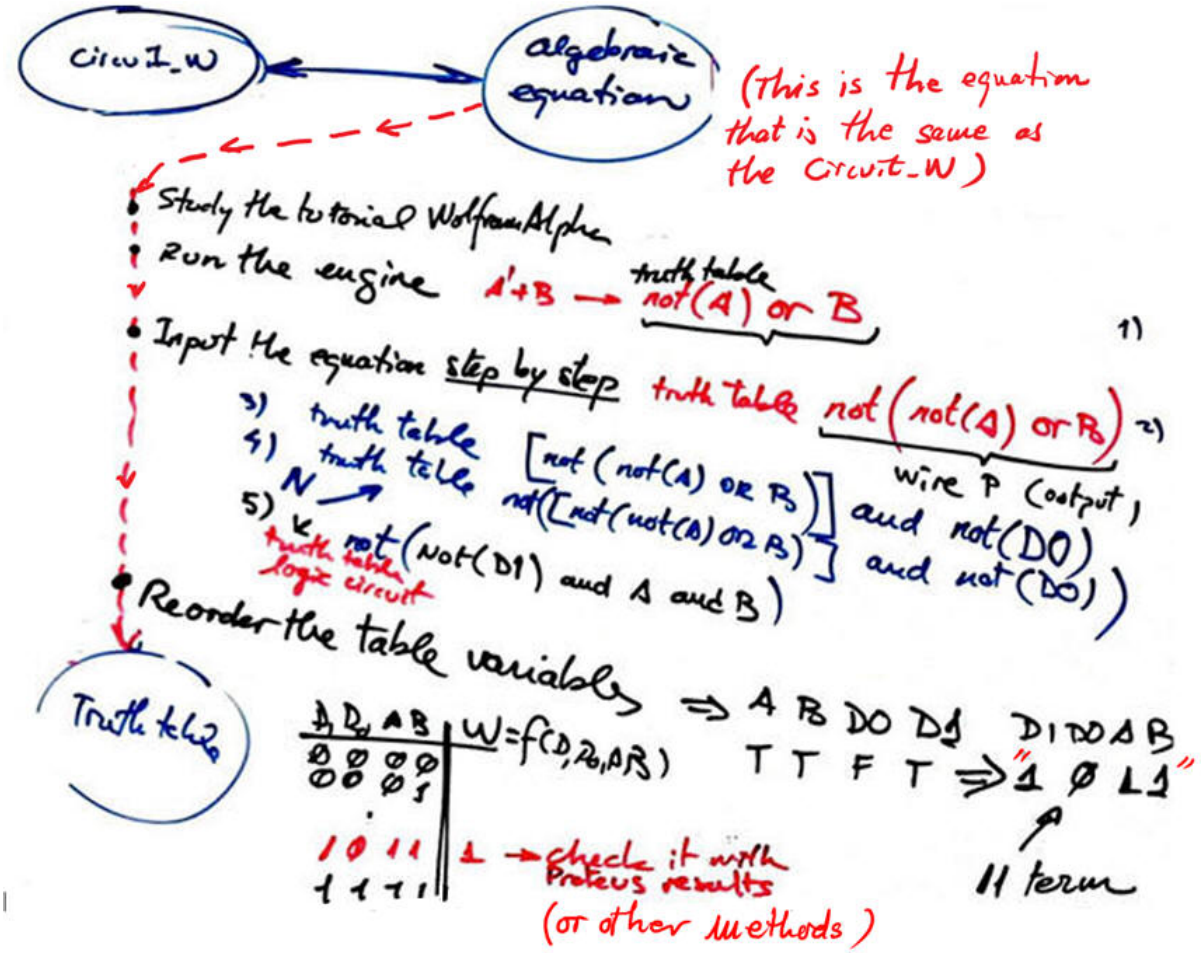


Fig. 2. Proposed plan to carry out this analysis project.

This is a convenient location for this project:

C:\CSD\P1\Lab1_1\Circuit_W\Wolfram\

Files may inlcude your written equations Circuit_W_equ.txt and other files like printed result images.

3. Development

Analyse the circuit and obtain its logic equation, as shown in Fig. 3.

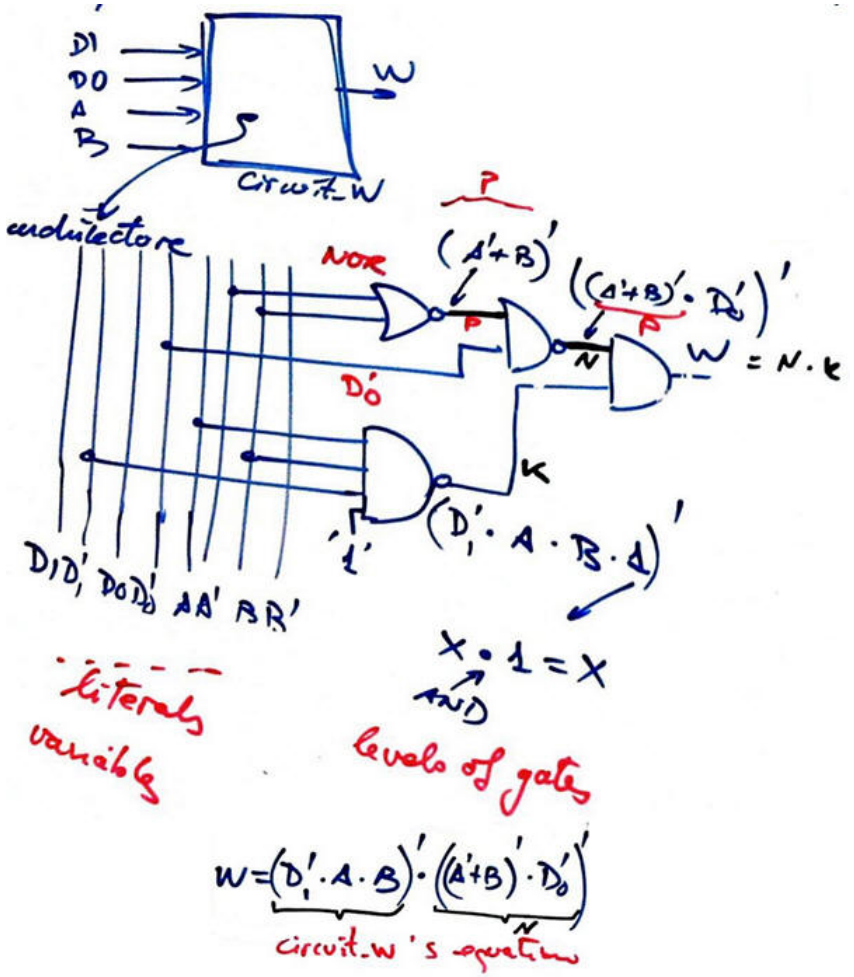


Fig. 3. Example handwritten circuit analysis gate by gate in order to find the circuit algebraic equation.

Start writing simpler equations in a text file and try them in WolframAlpha. This file Circuit_W_equ.txt contains some equations to copy and paste into WolframAlpha. We use Notepad++ as enriched text editor for writing text equations.

Run WolframAlpha engine to get results: "truth table", "logic circuit".

Print WolframAlpha truth table and logic circuit in a sheet of paper.

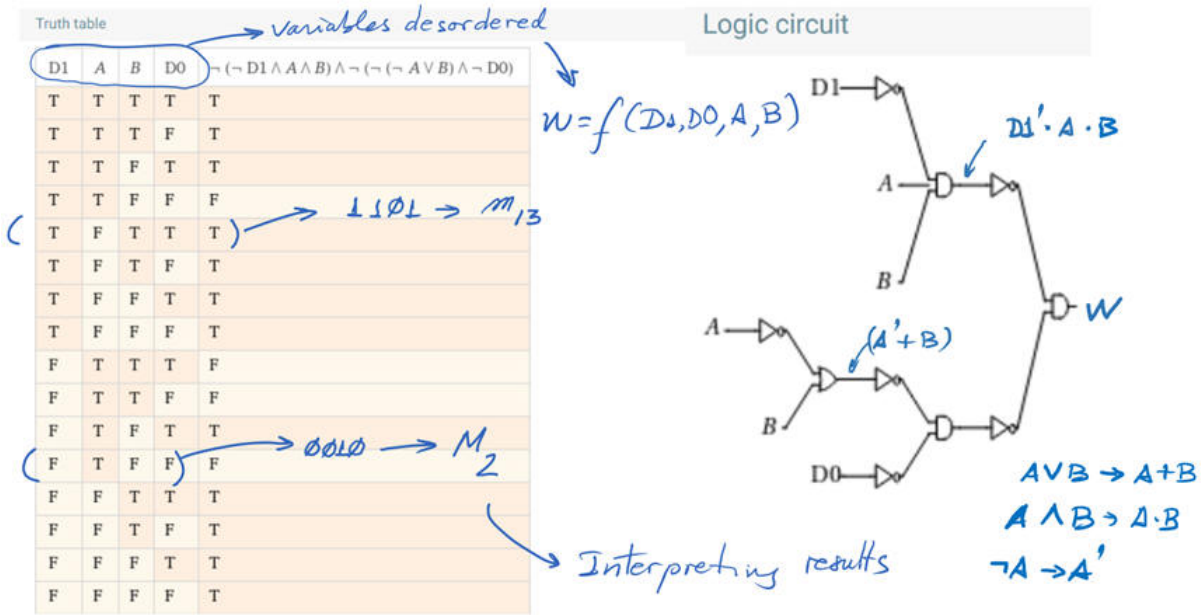


Fig. 4. Example of printed and interpreted results from WolframAlpha.

Reorder columns W = *f* (D1, D0, A, B) and complete the circuit's truth table identifying all minterns and maxterms.

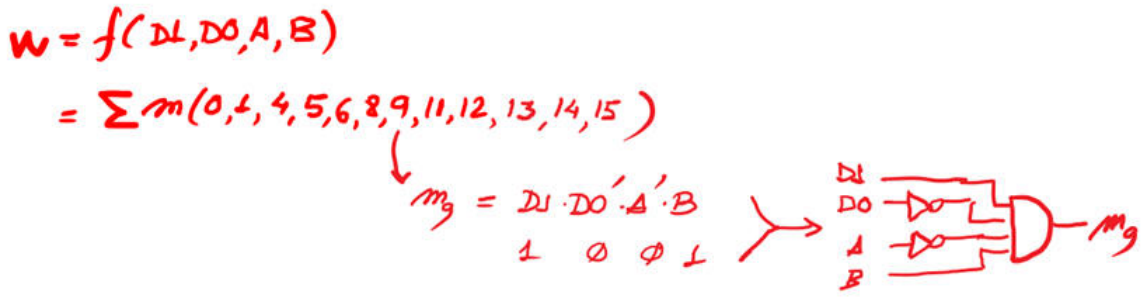


Fig. 5. Problem solution presented as a sum of minterms.

4. Test

Compare results with other methods. For instance method III solved in class L1.3 or method I solved in the project above.

In this way, this sections has to say something as simple as

"My *Circuit_W* generates the same truth table for methods II and method III".

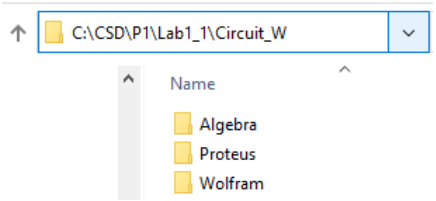
5. Report

- Follow this rubric for writing reports.

Summary of the complete discussions on *Circuit_W* analysis:

1. Specifications	2. Planning	3. Development	4. Test
Find *Circuit_W* truth table	Method I	Proteus tool	Verify whether you are obtaining the same truth table for all methods.
	Method II	WolframAlpha tool	
	Method III	Handwritten analysis (L1.3)	
	Method IV	VHDL EDA tools and VHDL testbech simulation (next Lab1_2)	

Furthermore, because most of the analysis and designs will require computer tools and applications, a folder is required for storing files from each project in your harddrive.



In next Lab1_2 we will study the same *Circuit_W* yet again with another method: