PLA1. Analysis and design using minimised equations

The aim of this exercise is firstly to analyse *Circuit_T* in Fig. 1 to obtain its truth table T = *f*(S1, S0, A, B); and secondly to design another equivalent circuit using minimised equations (SoP or PoS).

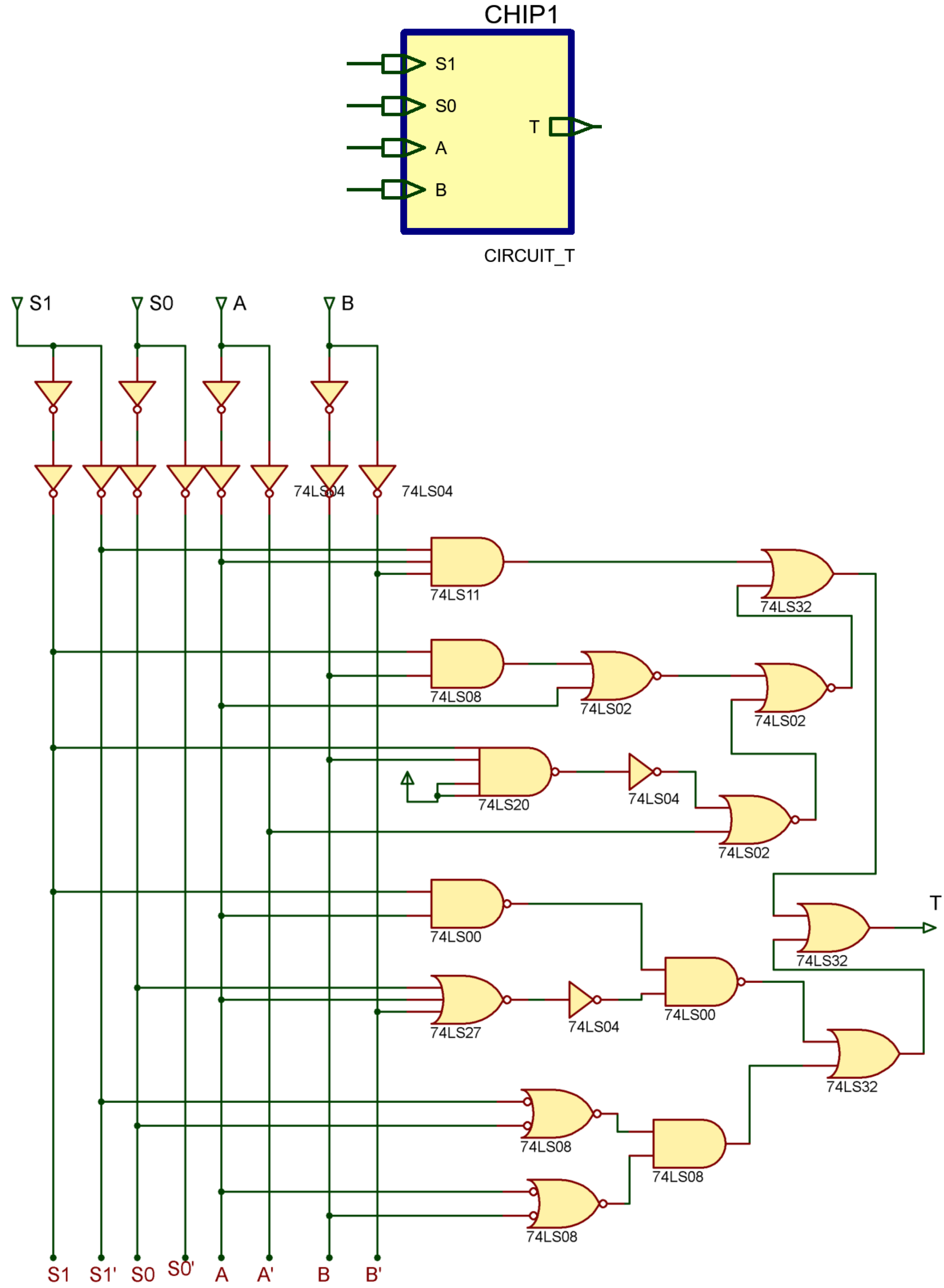
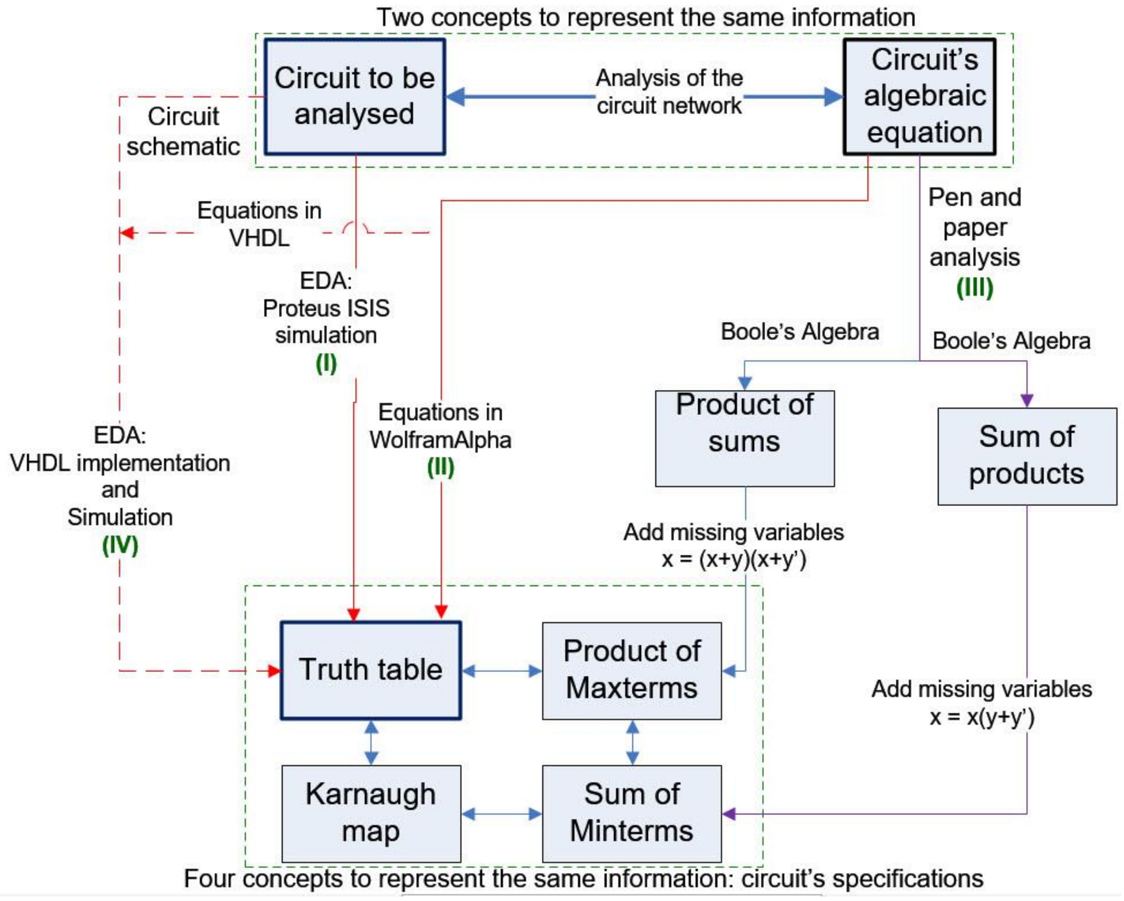


Fig. 1 *Circuit_T* composed of a network of logic gates.

Let us establish a plan to solve this problem. Fig. 2 shows several ways to plan how to determine the truth table of a given simple combinational circuit composed of logic gates.

Fig. 2 Multiple planning paths to analyse a circuit of logic gates.



Two concepts to represent the same information
Circuit to be analysed
Analysis of the circuit network
Circuit's algebraic equation
Circuit schematic
Equations in VHDL
EDA: Proteus ISIS simulation (I)
Equations in WolframAlpha (II)
Pen and paper analysis (III)
Boole's Algebra
Boole's Algebra
Product of sums
Sum of products
EDA: VHDL implementation and Simulation (IV)
Add missing variables x = (x+y)(x+y')
Add missing variables x = x(y+y')
Truth table
Product of Maxterms
Karnaugh map
Sum of Minterms
Four concepts to represent the same information: circuit's specifications

Phase A: ANALYSIS. Deduce and verify the circuit's truth table using the following analysis methods:

1. Method I. Draw the Circuit_T, capture the schematics in Proteus and run simulations to obtain its truth table. There are up to 16 input combinations to complete the circuit truth table. Logic family to be used for Proteus simulation is LS-TTL (low-power Schottky transistor-transistor logic). Find examples in this tutorial.

2. Method II. Deduce the logic equation that exactly matches the circuit and use numerical engine WolframAlpha to obtain the circuit's truth table. Type in the equation and analyse computer results to obtain the circuit's truth table. Find examples in this tutorial.

   Hint: To make it easy, the equation must be subdivided into smaller parts. Perhaps, if the equation is too large and it does not work, try to rename an input such S1 = P.

3. Method III. Apply Boolean algebra to determine the truth table (which is equivalent to the sum of minterms or the product of maxterms). In this way, the SoP or PoS expressions will be obtained as a step towards the final truth table.

Phase B: DESIGN. Invent several circuits from the given truth table in phase A, as shown in in Fig. 3 map.

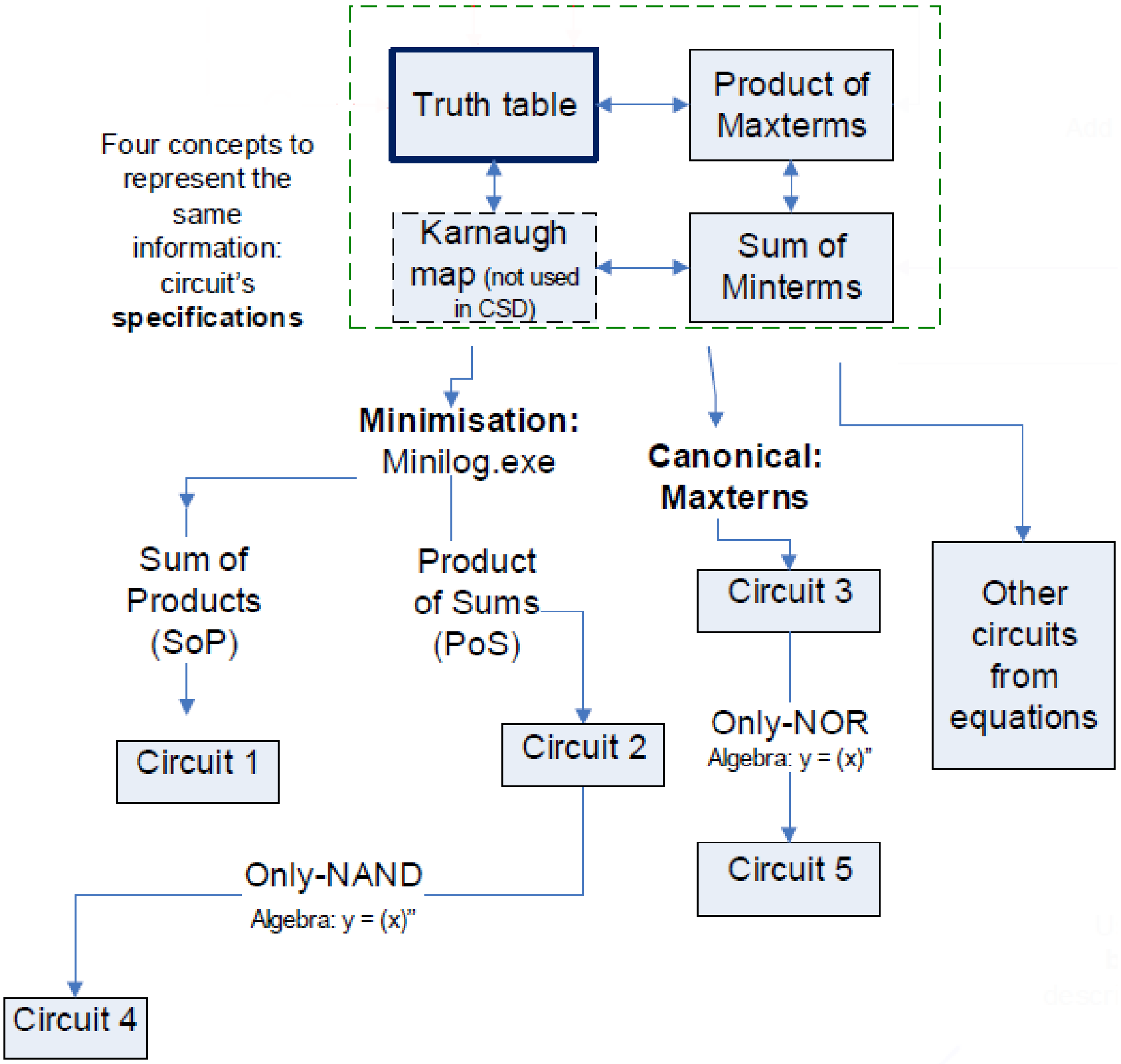


Fig. 3 Invention of circuits from the same initial truth table.

4. Create and verify a new *Circuit_2* by minimising the truth table and choosing PoS.

5. From *Circuit_2*, derive and verify a new *Circuit_4* based on 2-input NAND gates.

Each section has to be handled as a project, therefore, solution much follow CSD layout: 1 – 2 – 3 – 4

Each section mark is 2p.