# Plans for designing simple circuits

Four concepts to represent the same information: circuit's **specifications**

Truth table ↔ Product of Maxterms

Truth table ↕ Karnaugh map (not used in CSD)

Karnaugh map (not used in CSD) ↔ Sum of Minterms

Product of Maxterms ↕ Sum of Minterms

**Minimisation**
*Minilog.exe*

**Canonical**

**minterms**

**maxterms**

Circuits from other equations

Sum of Products (SoP)

Product of Sums (PoS)

Circuit 1

Circuit 2

Circuit 3

Circuit 4

Only-NAND
Algebra: y = (x)''

Only-NOR
Algebra: y = (x)''

Only-NAND

Only-NOR

Circuit 9

Like *Circuit_C* or *Circuit_K* in P1

Using a VHDL **behavioural** description of the truth table (plan B)

Section B

**Circuit design**

Circuit 5

Circuit 6

Circuit 7

Circuit 8

Only 2-input NAND

Circuit 10

Examples of plans A to obtain circuit structures depending on the equations (there are infinite circuits because any combination is possible)

Any kind of circuit (the VHDL synthesiser decides)
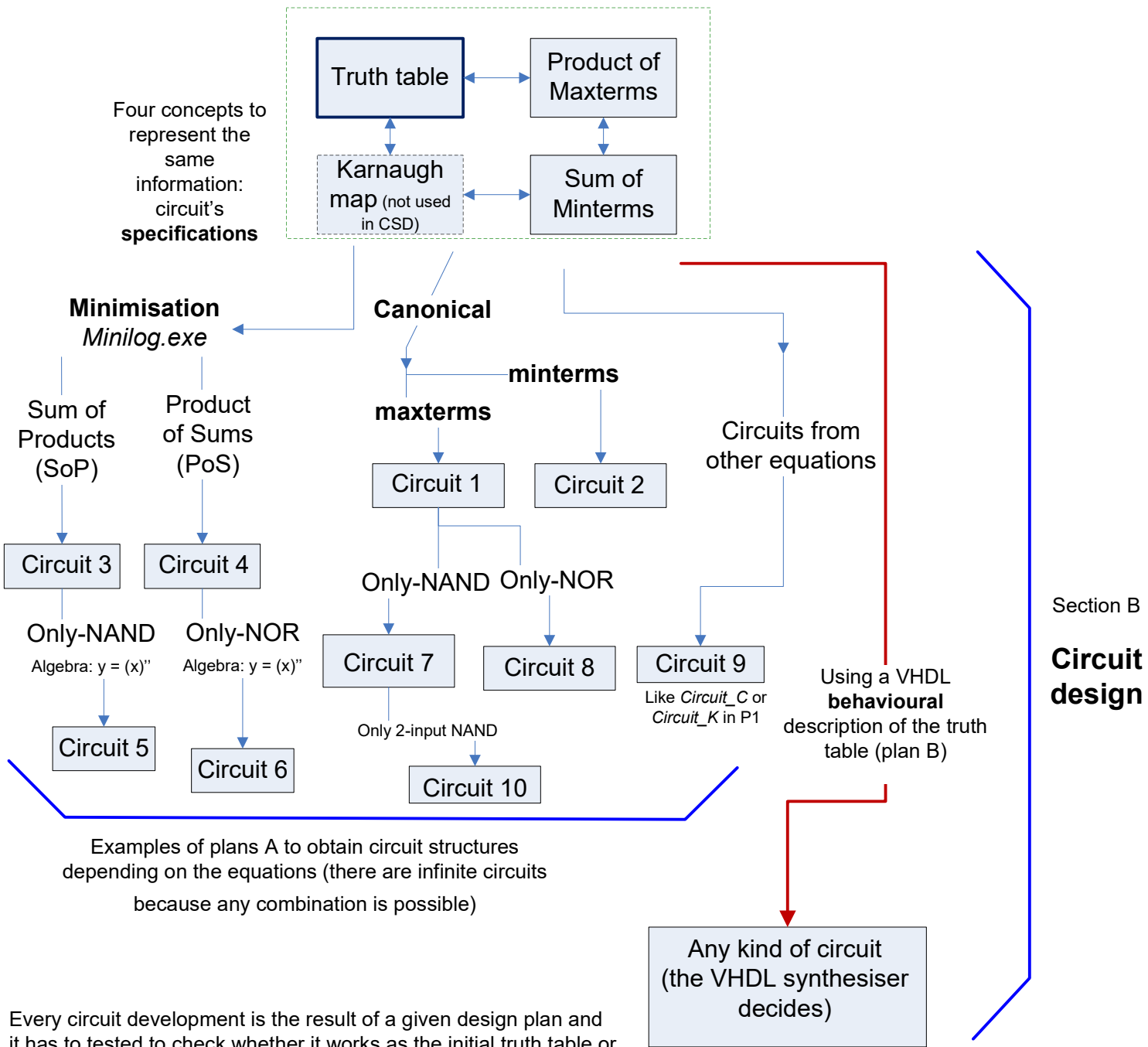
Every circuit development is the result of a given design plan and it has to tested to check whether it works as the initial truth table or not.

For instance, Circuit 1 may be tested using Proteus and the Circuit 2 using WolframAlpha

Specify → Plan → Develop → Test → Does it work? — no → Plan

Does it work? — yes → end