Background

At the time of the Star Wars Program early 1980's it became increasingly apparent to the US Department of Defence that in writing requirement specifications for complex systems involving both hardware and software it would make good sense to have a standard language for describing how a specific hardware unit performed.

Usually hardware was described by a circuit diagram and to help with the description some text and maybe a timing diagram. Often the text was ambiguous and the timing diagrams lacked sufficient detail. For large projects this implied that interfacing the work of various groups caused problems.

A standard language for describing how the circuit worked, used by all members of the consortium would significantly reduce the interfacing problems. From this the idea of Hardware Description Languages (HDLs) evolved. A development from this was that the circuit diagram was no longer the definition of the circuit it's main use was now in debugging the physical circuit.

With the introduction of large scale integrated circuits, chips containing 1000 plus gates, it became apparent with the relevant software tools, using the HDL text as a source the design could be directly mapped onto the silicon chip. i.e. The source code is compiled and the final output is the instructions, in whatever format, to produce the necessary information to make the required hardware system/sub-system.

This information could be in one or more of the following formats:
PCB Printed Circuit Board using a specified logic family/families.
FPLD Field Programmable Gate Array
FPGA Field Programmable Gate Array
ASIC Application Specific Integrated Circuit.

An obvious extension of the software tools associated with the HDL would be a Simulator/Debugger to test the design before committing it to real hardware. Associated with the Simulator will be test harnesses, to test the design.

Two main HDLs now exist VHDL and VERILOG the two languages are not that dissimilar and most compilers/simulators will now cope with both languages. We will be using VHDL.

A large number of simulator/debuggers are available some may be targeted for specific types of design i.e. ASIC or FPGA, some are universal. The Simulator/Debugger we will be using is V-System from Model Technologies.

VHDL

A condensed acronym for VHSICHDL
where:

V	Very High Speed Integrated Circuit
H	Hardware
D	Description
L	Language

The language is derived from or modelled on Ada!

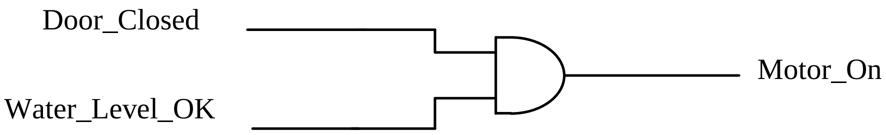
From this
Read Chap 1 of VHDL Training Manual.
Understand the idea of top-down design
You will only be concerned with the Behavioural Model and the Dataflow (RTL) Model.
Terminology and Conventions

How to describe a circuit using an HDL.

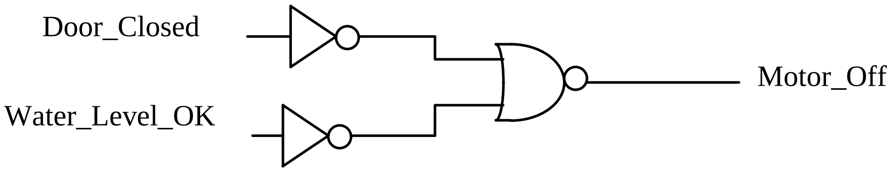
Easy to describe combinational circuits
i.e. For a Washing Machine.

Motor On <= Water_Level_OK and Door_Closed

c.f. The equivalent circuit diagram:



another equally valid circuit would be



NOTE
The designer has used a NOR gate plus Inverters to implement this function. Not so easy to see what was intended.

Considering the problem of how you would design a controller to sequence through the complete wash cycle is more complex but again can easily be described in an HDL, the circuit diagram is, however, far more complex.