Q] What is VHDL?

- VHSIC : Very High Speed Integrated Circuits
- VHDL = VHSIC HDL
- HDL = Hardware Description Language

Q] Why HDL, a language being taught to ECE & not CSE?

→ HDL has the power to capture descriptions of most complex chips to a complete electronic system.

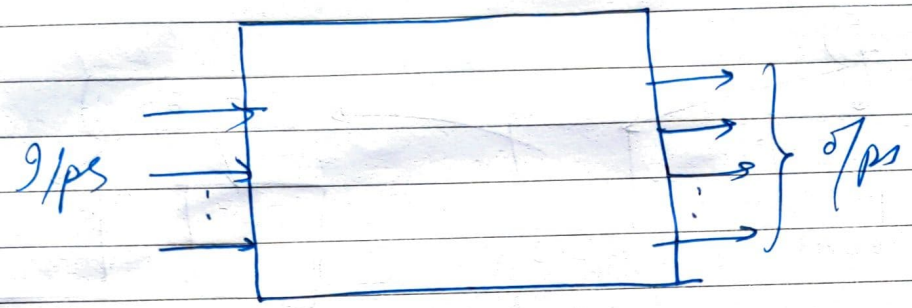
☆ History

* 1981: Under VHSIC prog., many US companies were involved in designing VHSIC chips for the department of defense (DoD)
  - Each company was using a different way to represent their design & ckt.
  - DoD wanted an uniform standard (language)
* 1983: IBM, TI & Intermetrics were awarded contract by DoD to develop a HDL
* 1985: Version 7.2 of VHDL was made public
* 1986: HDL standardization was given by IEEE

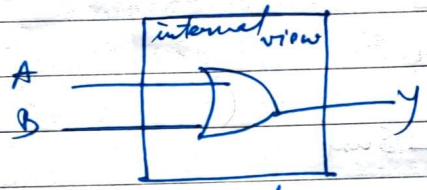
☆ Capabilities

* Exchange medium between chip vendors & CAD users.
* Hierarchy: Simplifies modelling (sub-blocks can be created)
* Top down / Bottom up approach
* HDL is not technology specific i.e. a HDL code can be implemented on 28nm, 14nm ~~et~~ technology
* Supports both synchronous & asynchronous timing modules
* Open source
* 3 styles: Structural, data flow & Behavioural
* Supports test benches
* Natural language

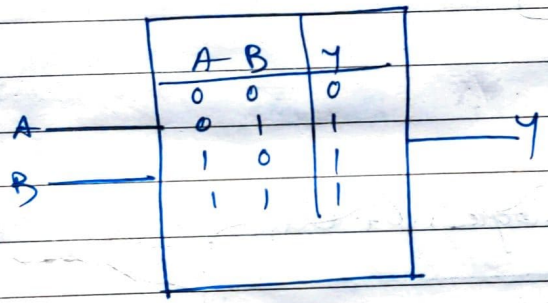
✱ Hardware Abstraction



Eg



external view



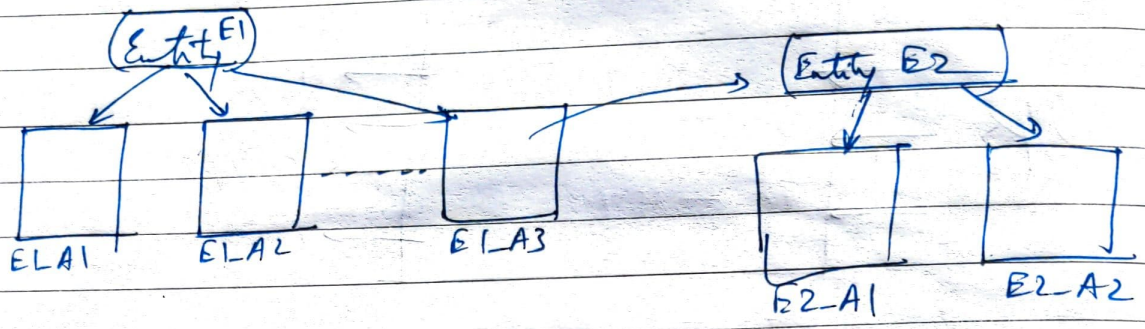
Outer box → ENTITY
Inner Description → ARCHITECTURE

✱ Definitions

* Entity: A h/w abstraction of a digital systems
* ~~Architecture~~:
* Describing an Entity:
  i) Entity declaration → External view of the entity
  ii) Architecture body → Internal description of the entity
  iii) Configuration declaration
  iv) Package declaration
  v) Package body

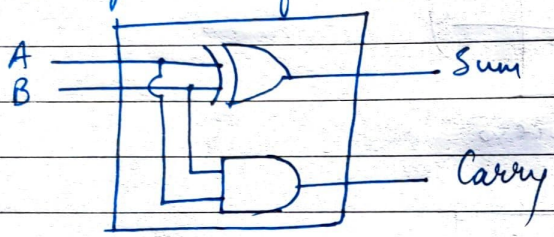
VHDL is NOT case sensitive!

* One entity can have many architectures



Lecture-2

✡ Example: Half Adder

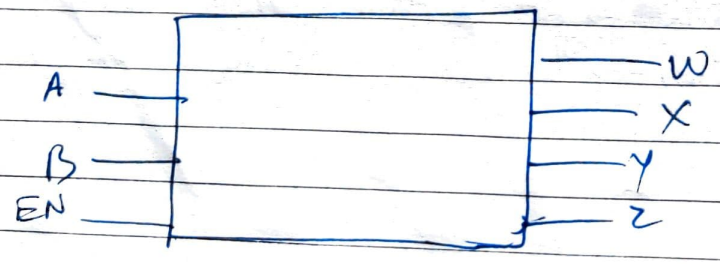



```
library ieee;
use ieee.std_logic_1164.all;
```

```
entity half_adder is
    port(A,B : in bit;
         sum, carry : out bit);
end half_adder;
```

architecture

✡ Example: 2:4 Decoder



```
entity decoder2_4 is
  port (a, b, EN : in bit;
        w, x, y, z : out bit);
end decoder2_4;
```

✡ Architecture Body (Pg. 34)

* Internal details of an entity are specified here
* 3 Styles of modelling:
  i) Structural: A set of interconnected components
  ii) Dataflow: A set of concurrent assignments
  iii) Behavioural: A set of sequential statements

* We will start with Behavioural ~~soft~~ style of modeling.

Lecture - 3

✡ Basic Language Elements (Pg. 53)

✡ Data Objects (Pg. 55)

A data object holds a value of a specified type. It is created by means of an object declaration

✡ Classification of Data Objects

1) Constant ⇒ constant rise_time := 10 ns;
2) Variable ⇒ variable sum: integer range 0 to 100 := 10;
3) Signal ⇒ signal gate_delay: time := 10ns;
4) File ⇒ file file_name: file_type_name [open mode] is string_expression;

✡ Data Types (Pg. 59)

① Enumeration: Set of user defined values:

```
type MVL is ('U', '0', '1', 'Z');
type MicroOP is (LOAD, STORE, ADD, SUB, MUL, DIV;
subtype ArithOP is MicroOP range ADD to DIV;
```