## ✱ 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 := 19;
3) Signal ⇒ signal gate_delay: time := 10 ns;
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 Micro_OP is (LOAD, STORE, ADD, SUB, MUL, DIV;

subtype Arith_OP is Micro_OP range ADD to DIV;

~~* Data Type (Pg: 59)~~

(2) Q] Find the type def. of STD_ULOGIC defined in std_logic_1164 package.

(2) Integer Types

```
type INDEX is range 0 to 15;
type WORD_LENGTH is range 31 downto 0;
```

(3) Floating Point Types

```
type Temp_Data is range 0.0 to 100.0;
```

(4) Physical Types

```
type CURRENT is range 0 to 1E9
    units
        nA;
        uA = 1000 nA;
        mA = 1000 uA;
        Amp = 1000 mA;
    end units;
subtype Filter_Current is CURRENT range 10uA to 5mA.
```

* Composite Types (Pg. 67)

It represents a collection of values

(1) Array Type

It is a collection of values all belonging to same a single type

Ex:

```
type Address_Word is array (0 to 63) of bits;
type Data_Word is array (7 downto 0) of std_logic;
type ROM is array (0 to 125) of Data_Word;
```

## (2) Record Types

An object of a record type is composed of elements of same or different types - (Analogus to struct of 'C'.)

```
type Pin_Type is range 0 to 10;
type Module is
  record
    size : integer range 20 to 200;
    delay: time;
    No_ips: Pin_Type;
    No_ops: Pin_Type;
  end record;
```

# ☆ Operators (Pg. 81)

1) Logical
2) Relational
3) Shift
4) Addition
5) Multiplication
6) Miscellaneous

## ① Logical:

and or nand nor xor xnor not

## ② Relational

= /= < <= > >=

## ③ Shift Operators

sll : shift left logical (shift & fill 0s)
sla : shift left arithmatic (shift & fill rightmost bit)

Similarly, srl & sra

Eg:

"10011011" sll 2 is "01101100"

"10011011" sra 3 is "11110011"

rol 4

ror 5

sla -2

srl -3

## (4) Addition Operators

+ − &

Ex: 'C' & 'A' & 'T' is "CAT"

## (5) Multiplication Operators

* / mod rem

| | |
|---|---|
| 9 mod 5 = 4 | 9 rem 5 = 4 |
| 9 mod -5 = -1 | 9 rem (-5) = 4 |
| -9 mod 5 = 1 | -9 rem 5 = -4 |
| -9 mod -5 = -4 | -9 rem (-5) = -4 |

A rem B = A − (A/B) × B ⟹ use sign of 1st operand

↳ A mod B = A − B × N ... for some integer N

↳ Ans will have sign of 2nd operand

* Homework: Difference b/w mod & rem.