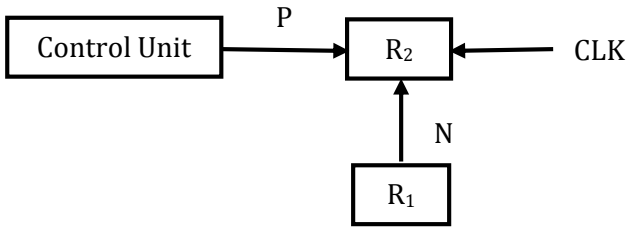
Chapter – 2

Register Transfer and Micro-operations

Register Transfer:

As we know that the digital computer has many registers and while performing certain tasks the content of one register may change or transfer to another. The operations or micro-operation which transfers content of one register to another is known as register transfer.

The register transfer is represented as: $R_2 \leftarrow R_1$. It implies that the content of register R_1 is transfer to R_2. The control signal can also be known as P: $R_2 \leftarrow R_1$. The important hardware implementation is:



The data transfer (Register to Register) occurs in a single clock pulse known as T-state. A T-state is the smallest unit of time in which smallest operation is performing.

Register Transfer Language (RTL):

It is a symbolic notation used to describe the operation in a register transfer, which describes the elementary operation of digital computer system. Language is totally computer architecture dependent and hence RTL defines symbol or various types of micro-operations.

Micro-operation is a primitive action performed by a machine on the data stored in one or more-registers. We can say that micro-operations are functional or automatic operation of processor.

Types of Micro-operations:

1. Register Transfer Micro-operation:

These micro-operations transfer information from one register to another. The information contained in the register does not change during micro-operation. It can written as: **$R_2 \leftarrow R_1$**. Here, the content of R_1 is transfer to R_2 and for this operation there must be a data path for data transfer from source register to destination register.

2. Arithmetic Micro-operation:

These micro-operations performs on arithmetic operations on numeric data stored in register. The basic operation may be addition, subtraction, increment, decrement, etc.

Example:

R3 ← R1+R2
R2 ← R1- R3
R1 ← R2 + 1
R3 ← R1 – 1

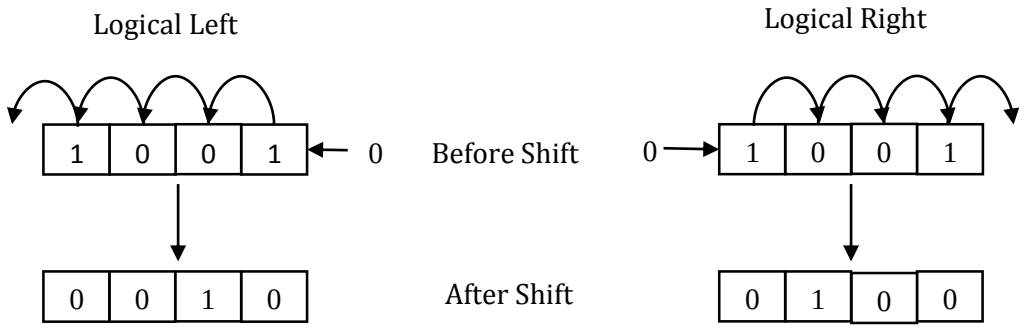
3. Logical Micro-operation:

They basically perform binary operation on the bits of a string stored in a register. For a logical micro-operation each bit of a register is considered as a variable. Some of the common logical micro-operations are: AND, OR, NAND, NOT, etc.

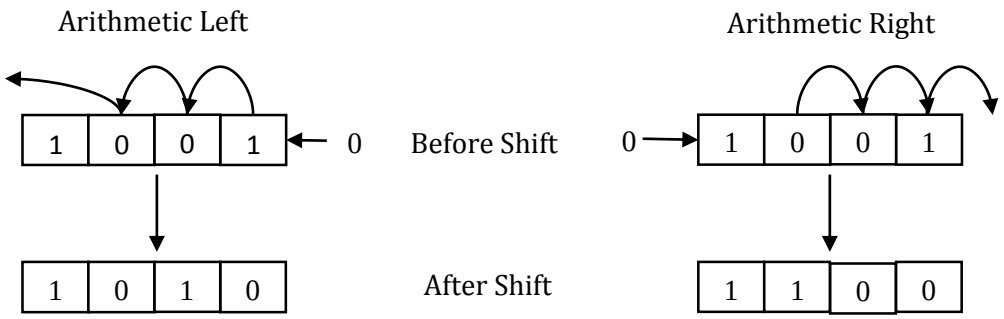
4. Shift Micro-operation:

Shift is a useful operation which can be shifted as a serial data transfer and they can also be along with arithmetic and logical operations.

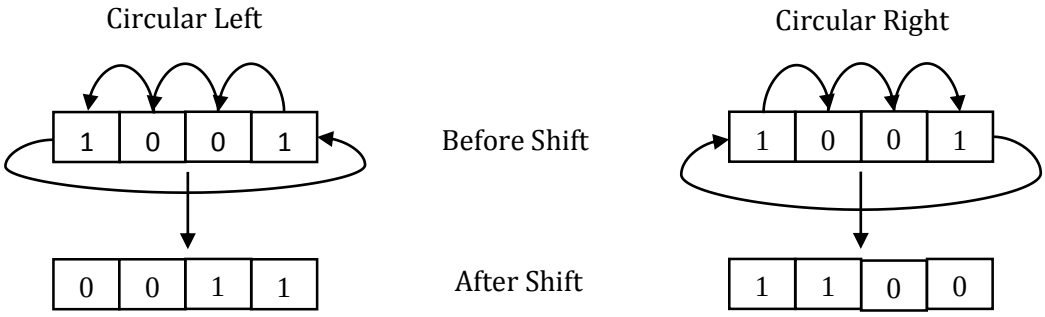
a. Logical Shift:



b. Arithmetic Shift (Sign Bit Should Not Change):



c. Circular Shift:



INTRODUCTION TO HDL:

In electronics, a hardware description language (HDL) is a specialized computer language used to describe the structure and behavior of electronic circuits, and most commonly, digital logic circuits.

A hardware description language enables a precise, formal description of an electronic circuit that allows for the automated analysis and simulation of an electronic circuit. It also allows for the synthesis of a HDL description into a netlist (a specification of physical electronic components and how they are connected together), which can then be placed and routed to produce the set of masks used to create an integrated circuit.

A hardware description language looks much like a programming language such as C; it is a textual description consisting of expressions, statements and control structures. One important difference between most programming languages and HDLs is that HDLs explicitly include the notion of time.

INTRODUCTION TO VHDL:

VHDL stands for very high-speed integrated circuit hardware description language. Which is one of the programming language used to model a digital system by dataflow, behavioral and structural style of modeling. This language was first introduced in 1981 for the department of Defense (DoD) under the VHSIC programe. In 1983 IBM, Texas instruments and Intermetrics started to develop this language. In 1985 VHDL 7.2 version was released. In 1987 IEEE standardized the language.

This language is hardware dependent, where any digital system can be simulated or can be represented in VHDL. This language is only used in digital system. Other languages such as ABEL (Advanced Boolean Expression Language) and VEILOG are also used for hardware description but among these VHDL is more popular and widely used.

The program structure for VHDL is:

```
entity-name is;
[PORT] (interface-signal-declaration);
end (entity) entity-name;
```

architecture architecture-name of entity is;
[declaration]
begin;
architecture-body;
end [architecture] architecture-name;

Example:

NOR = OR + NOT
Entity-NOR is;
[PORT] (input A, B; Output Y)
End NOR;
Architecture NOR of NOR is;
Input A, B; X-output;
Input X; Y-output;
Begin;
A, B; Y
End NOR