**Date: 16-04-2021**

**Branch: CSE (III Year), EE (III Year) and MCA (I Year)**

**Topic: Register Transfer language and Bus and Memory Transfers**

**Time: 08:00 AM -9:00 AM**

In symbolic notation, it is used to describe the micro-operations transfer among registers. It is a kind of intermediate representation (IR) that is very close to assembly language, such as that which is used in a compiler. The term “Register Transfer” can perform micro-operations and transfer the result of operation to the same or other register.

**Micro-operations:**

The operation executed on the data store in registers are called micro-operations. They are detailed low-level instructions used in some designs to implement complex machine instructions.

**Register Transfer:**

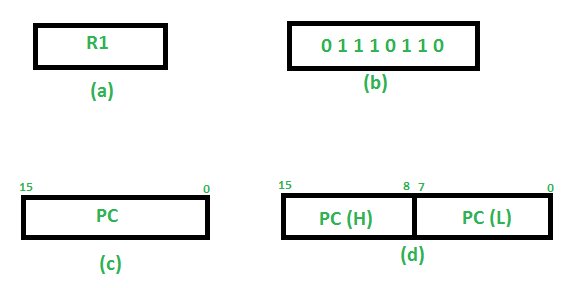
The information transformed from one register to another register is represented in symbolic form by replacement operator is called Register Transfer.

**Replacement Operator :**

In the statement, R2 <- R1, **<-** acts as a replacement operator. This statement defines the transfer of content of register R1 into register R2.

There are various methods of RTL –

1. General way of representing a register is by the name of the register in closed in a rectangular box as shown in (a).
2. Register is numbered in a sequence of 0 to (n-1) as shown in (b).
3. The numbering of bits in a register can be marked on the top of the box as shown in (c).
4. A 16-bit register PC is divided into 2 parts- Bits (0 to 7) are assigned with lower byte of 16-bit address and bits (8 to 15) are assigned with higher bytes of 16-bit address as shown in (d).



**Basic symbols of RTL:**

| Symbol | Description | Example |
| --- | --- | --- |
| Letters and Numbers | Denotes a Register | MAR, R1, R2 |
| ( ) | Denotes a part of register | R1(8-bit)  R1(0-7)  R1 (8-15) |
| <- | Denotes a transfer of information | R2 <- R1 |
| , | Specify two micro-operations of Register Transfer | R1 <- R2  R2 <- R1 |
| : | Denotes conditional operations | P : R2 <- R1  if P=1 |
| Naming Operator (:=) | Denotes another name for an already existing register/alias | Ra := R1 |

**Register Transfer Operations:**

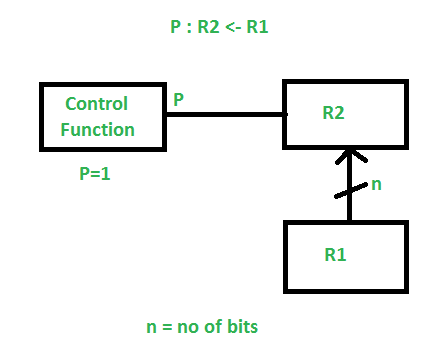
The operation performed on the data stored in the registers are referred to as register transfer operations.

There are different types of register transfer operations:

**1. Simple Transfer – R2 <- R1**

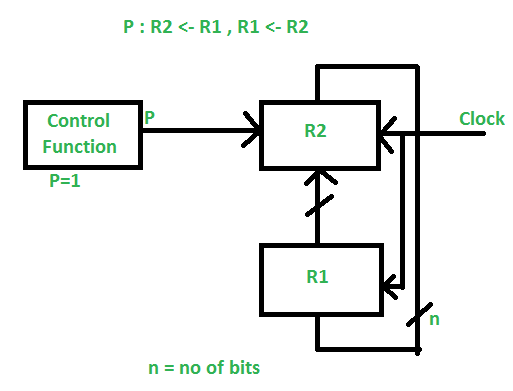
The content of R1 are copied into R2 without affecting the content of R1. It is an unconditional type of transfer operation.

**2. Conditional Transfer –**



It indicates that if P=1, then the content of R1 is transferred to R2. It is a unidirectional operation.

**3. Simultaneous Operations –**   
If 2 or more operations are to occur simultaneously then they are separated with comma **(,)**.



If the control function P=1, then load the content of R1 into R2 and at the same clock load the content of R2 into R1.

Bus and Memory Transfers

### 4.3.1. BUS

If a computer has 16 registers, each holding 32 bits, how many wires are needed to connect every register to every other? (162) For a large number of wide registers, the wires could end up taking most of the space in the circuit.

A BUS is a single shared set of wires connecting all registers.

#### 4.3.1.1. BUS with Multiplexers

Figure 4-3, bus system for 4 registers using multiplexers.

Bus, 4 8-bit registers connected for input and output. Use decoder/demux to drive load input on registers. Wire one MUX. Clock speed is limited by the propagation delay through the MUX and wires.

Only one register's contents can be on the bus for a given clock cycle. Which of the following are legal?

* A ← B, C ← D
* A ← C, D ← C
* B ← D, B ← A

Book error: No R1 in the diagram.

BUS ← C, A ← BUS

Can be written as:

A ← C

Multiplexers must be driven to select register C t time units before the clock pulse that loads A, where t is the propagation delay of the multiplexers.

#### 4.3.1.2. BUS with 3-state Buffers

3-state gates

3-state buffers can be used instead of multiplexers. A 3-state buffer is a combinational circuit that acts like a simple switch: It either passes the input signal to the output (C=1), or blocks the electrical current (C=0). The latter case is called high impedance state.

I ------|>-------- O

|

C -------+

3-state buffers can be used to connect or disconnect register outputs to/from the bus. A single decoder can control a large number of 3-state buffers. The outputs of the 3-state buffers can be tied directly together, provided that the circuit guarantees that all but one will be in high-impedance state at any given time. Using a decoder to control them guarantees this.

### 4.3.2. Memory Transfer

The internal bus connects only registers within the CPU, so how do we get data to and from memory?

The address register (AR) is used to select a memory address, and the data register (DR) is used to send and receive data. Both these registers are connected to the internal bus. DR is a bridge between the internal BUS and the memory data BUS.

Memory can also be connected directly to the internal BUS in theory.

Diagram showing connections to memory unit.

M[AR] ← DR

DR ← M[AR]

Hence, accessing memory outside the CPU requires at least two clock cycles. First we load AR with the desired memory address, and then transfer to or from DR. In most typical computer systems, memory transfers take many clock cycles, known as wait states. Solutions for reducing wait states are discussed in