**Registers**

Register – is a group of flip-flops. Its basic function is to hold information within a digital system so as to make it available to the logic units during the computing process. However, a register may also have additional capabilities associated with it.

As we know, Register is a collection of flip flops. A flip flop is used to store single bit digital data. For storing a large number of bits, the storage capacity is increased by grouping more than one flip flops. If we want to store an n-bit word, we have to use an n-bit register containing n number of flip flops.

The register is used to perform different types of operations. For performing the operations, the CPU use these registers. The result returned by the system will store in the registers. There are the following operations which are performed by the registers:

Fetch:

It is used

o To take the instructions given by the users.

o To fetch the instruction stored into the main memory.

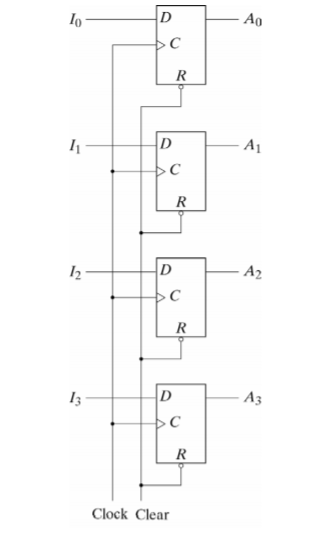
Decode:

The decode operation is used to interpret the instructions. In decode, the operation performed on the instructions is identified by the CPU. In simple words, the decode operation is used to decode the instructions.

Execute:

The execution operation is used to store the result produced by the CPU into the memory. After storing this result, it is displayed on the user screen.

A simple 4-bit register is shown below

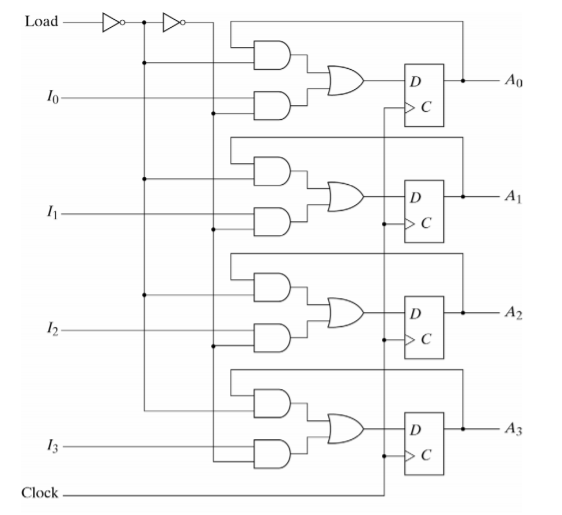


The common clock input triggers all flip-flops and the binary data available at the four inputs are

transferred into the register. The clear input is useful for clearing the register to all 0’s output.

**Register with Parallel Load**

If all the bits in a register are loaded at the same time, the loading is done in parallel. A 4-bit register with a load control input is shown below: This register will be built around four edge triggered D flip-flops. The register has four data input bits: I0, I1, I2, and I3. In addition, the register also has four data output bits: A0, A1, A2, and A3. There is a Load input which, when at logic level 1, sets the four flip-flop data inputs to the register's four input bits. When Load is at logic level 0, the four flip-flops have their outputs fed into their inputs. This effectively disables loading from the four input bits. In schematic form we have the following:



The Load input determines the action to be taken with each clock pulse. The feedback connection

from output to input is necessary because the D flip-flop does not have a “no change” condition.

**Types of Registers**

There are various types of registers which are as follows:

**MAR or Memory Address Register**

The MAR is a special type of register that contains the memory address of the data and instruction. The main task of the MAR is to access instruction and data from memory in the execution phase. The MAR stores the address of the memory location where the data is to be read or to be stored by the CPU.

**Program Counter(PC)**

The program counter is also called an instruction address register or instruction pointer. The next memory address of the instruction, which is going to be executed after completing the execution of current instruction is contained in the program counter. In simple words, the program counter contains the memory address of the location of the next instruction.

**Accumulator Register**

The CPU mostly uses an accumulator register. The accumulator register is used to store the system result. All the results will be stored in the accumulator register when the CPU produces some results after processing.

**MDR or Memory Data Register**

Memory Data Register is a part of the computer's control unit. It contains the data that we want to store in the computer storage or the data fetched from the computer storage. The MDR works as a buffer that contains anything for which the processor is ready to use it. The MDR contains the copied data of the memory for the processor. Firstly, the MDR holds the information, and then it goes to the decoder.

The data which is to be read out or written into the address location is contained in the Memory Data Register.

The data is written in one direction when it is fetched from memory and placed into the MDR. In write instruction, the data place into the MDR from another CPU register. This CPU register writes the data into the memory. Half of the minimal interface between the computer storage and the microprogram is the memory data address register, and the other half is the memory data register.

**Index Register(IR)**

The Index Register is the hardware element that holds the number. The number adds to the computer instruction's address to create an effective address. In CPU, the index register is a processor register used to modify the operand address during the running program.

**Memory Buffer Register (MBR)**

Memory Buffer Register is mostly called MBR. The MBR contains the Metadata of the data and instruction written in or read from memory. In simple words, it is used to store the upcoming data/instruction from the memory and going to memory.

**Data Register(DR)**

The data register is used to temporarily store the data. This data transmits to or from a peripheral device.