

6.1. COMPUTER REGISTERS SET (REGISTER ORGANIZATION)

Registers are referred as extremely fast memory spaces within the CPU that are used to store the results of execution of the instructions in CPU. Different computers have different set of register. They differ in the number of registers, register types, and the length of each register. They also differ in the usage of each register.

General-purpose registers can be used for multiple purposes and assigned to a variety of functions by the programmer.

Special-purpose registers are restricted to only specific functions. In some cases, some registers are used only to hold data and cannot be used in the calculations of operand addresses. The length of a data register must be long enough to hold values of most data types. Some machines allow two contiguous registers to hold double-length values.

Address registers may be dedicated to a particular addressing mode or may be used as address general purpose. Address registers must be long enough to hold the largest address. The number of registers in a particular architecture affects the instruction set design. A very small number of registers may result in an increase in memory references.

Small Concept

All CPU registers has their unique identity and special role in computer system.

Another type of registers is used to hold processor status bits, or flags. These bits are set by the CPU as the result of the execution of an operation. The status bits can be tested at a later time as part of another operation.

- Memory Access Registers
- Condition Registers
- Instruction Fetching Registers
- Special-Purpose Address Registers

6.1.1. Memory Access Registers

There are two registers referred as Memory Access Registers

1. MDR (Memory Data Register)
2. MAR (Memory Address Register)

Two basic operations of memory are write operation and read operation, the registers memory data register (MDR) and memory address register (MAR) are respectively used for these operations. The MDR and MAR are used exclusively by the CPU. These registers are not directly accessible by developers. In order to perform a write operation into a specified memory location, these are used as following way:

1. The word to be stored into the memory location is first loaded by the CPU into MDR.
2. The address of the location into which the word is to be stored is loaded by the CPU into a MAR.
3. A write signal is issued by the CPU.

Similarly, to perform a memory read operation, the MDR and MAR are used as following way:

1. The address of the location from which the word is to be read is loaded into the MAR.
2. A read signal is issued by the CPU.
3. The required word will be loaded by the memory into the MDR ready for use by the CPU.

6.1.2. Instruction Fetching Registers

There are two main registers involved for fetching an instruction for execution:

1. PC (Program Counter)
2. IR (Instruction Register)

The PC contains the address of the next instruction to be fetched. The fetched instruction is loaded in the IR for execution. After a successful instruction fetch, the PC is updated to point to the next instruction to be executed. In the case of a branch operation, the PC is updated to point to the branch target instruction after the branch is resolved, that is, the target address is known.

6.1.3. Condition Registers

Condition registers, or flags, are used to maintain status information. Some architectures contain a special program status word (PSW) register. The PSW contains bits that are set by the CPU to indicate the current status of an executing program. These indicators are typically for arithmetic operations, interrupts, memory protection information, or processor status.

The condition register indicates the status of the CPU as well as controls its operations. A condition register is a flag register which indicates some conditions produced by the execution of an instruction or controls certain operations of the CPU. A 16-bit flag register in the CPU contains nine active flags.

There are two types of flags:

1. Conditional or Status Flag
2. Machine control flags

Small Concept

The condition register indicates the status of CPU as well as controls its operations.

Conditional or status flags: Six flags are conditional flags. They are set or reset by the EU on the basis of the results of some arithmetic operation.

- **Carry Flag (CF):** indicates a carry after addition or a borrow after subtraction, also indicates error conditions.
- **Parity Flag (PF):** is a logic "0" for odd parity and a logic "1" for even parity.
- **Auxiliary Carry Flag (AF):** important for BCD addition and subtraction; holds a carry (borrow) after addition (subtraction) between bits position 3 and 4. Only used for DAA and DAS instructions to adjust the value of AL after a BCD addition (subtraction).
- **Zero Flag (ZF):** indicates that the result of an arithmetic or logic operation is zero.
- **Sign Flag (SF):** indicates arithmetic sign of the result after an arithmetic operation.
- **Overflow Flag (OF):** a condition that occurs when signed numbers are added or subtracted. An overflow indicates that the result has exceeded the capacity of the machine.

Machine control flags: The three remaining flags in the flags register are used to control certain operations of processor. They are called the control flags.

RED MI NOTE 6 PRO

MI DUAL CAMERA

The control flags are deliberately set or reset with specific instructions you put in your program. The three control flags are:

- **Trap Flag (TF)**: used for single stepping through a program;
- **Interrupt Flag (IF)**: used to allow or prohibit the interruption of a program;
- **Direction Flag (DF)**: used with string instructions.

6.1.4. Special-Purpose Registers

There are three main special purpose registers:

1. Index Register
2. Segment Pointers
3. Stack Pointers

1. Index Register

Index register used in index addressing mode (discussed in Chapter -7), to obtain the address of the operand by adding the content of the register. The index register holds an address displacement. Index addressing is indicated in the instruction by including the name of the index register in parentheses and using the symbol X to indicate the constant to be added.

2. Segment Pointers

In order to support segmentation, the address issued by the processor should consist of a segment number (base) and a displacement (or an offset) within the segment. A segment register holds the address of the base of the segment.

3. Stack Pointer

A stack is a data organization mechanism in which the last data item stored is the first data item retrieved. Two specific operations can be performed on a stack. These are the Push and the Pop operations. A specific register, called the stack pointer (SP), is used to indicate the stack location that can be addressed. In the stack push operation, the SP value is used to indicate the location (called the top of the stack). After storing (pushing) this value, the SP is incremented (in some architectures, e.g. X86, the SP is decremented as the stack grows low in memory).

6.2. DATAPATH

The CPU itself can be also segregated into two sections:

- Data section and
- Control section.

The Data section (Datapath), contains the registers and the ALU. The datapath is capable of performing certain operations on data items.

Small Concept

Data path is used for transferring data contents among the components of computer.

The Control section is basically the control unit, which issues control signals to the datapath. Internal to the CPU data moves from one register to another and between ALU and registers.

REDMI NOTE 6 PRO
MI DUAL CAMERA

Internal data movements are performed via local buses, which may carry data, instructions, and addresses. Externally, data move from registers to memory and I/O devices, often by means of a system bus.

The bus organization may use one-bus, two-bus, or three-bus architecture.

6.2.1. One-Bus Organization

In this organization only one bus is used to move outgoing and incoming data. Since a bus can handle only a single data movement within one clock cycle, two-operand operations will need two cycles to fetch the operands for the ALU. Additional registers may also be needed to buffer data for the ALU.

Small Concept

One-bus organization uses only single bus for all inputs and outputs.

This bus organization is the cheapest and simple, but it limits the amount of data transfer that can be done in the same clock cycle. The process of data transfer in same clock will slow down the overall performance of the system.

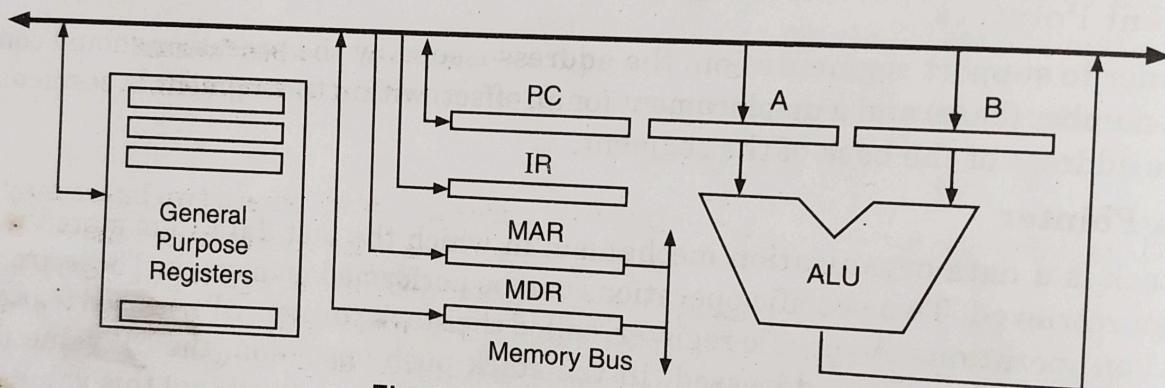


Fig. 6.2. One bus datapath

6.2.2. Two-Bus Organization

The two bus organization has fast performance than the one-bus organization. In this organization, general-purpose registers are directly connected to both buses. The content of two different registers can be transferred into the input point of the ALU at the same time. Therefore, two different operands of an operation can be fetched in the same clock cycle. An additional buffer register may be needed to hold the output of the ALU when the two buses are busy carrying the two operands.

Small Concept

Two-bus organization uses two buses one for input data and one for output data.

Figure 6.3(a) shows a two-bus organization. In some cases, one of the buses may be dedicated for moving data into registers (input-bus), while the other is dedicated for transferring data out of the registers (output-bus). In this case, the additional buffer register may be used, as one of the ALU inputs, to hold one of the operands. The ALU output can be connected directly to the in-bus, which will transfer the result into one of the registers. Figure 6.3(b) shows a two-bus organization with in-bus and out-bus.

REDMI NOTE 6 PRO
MI DUAL CAMERA

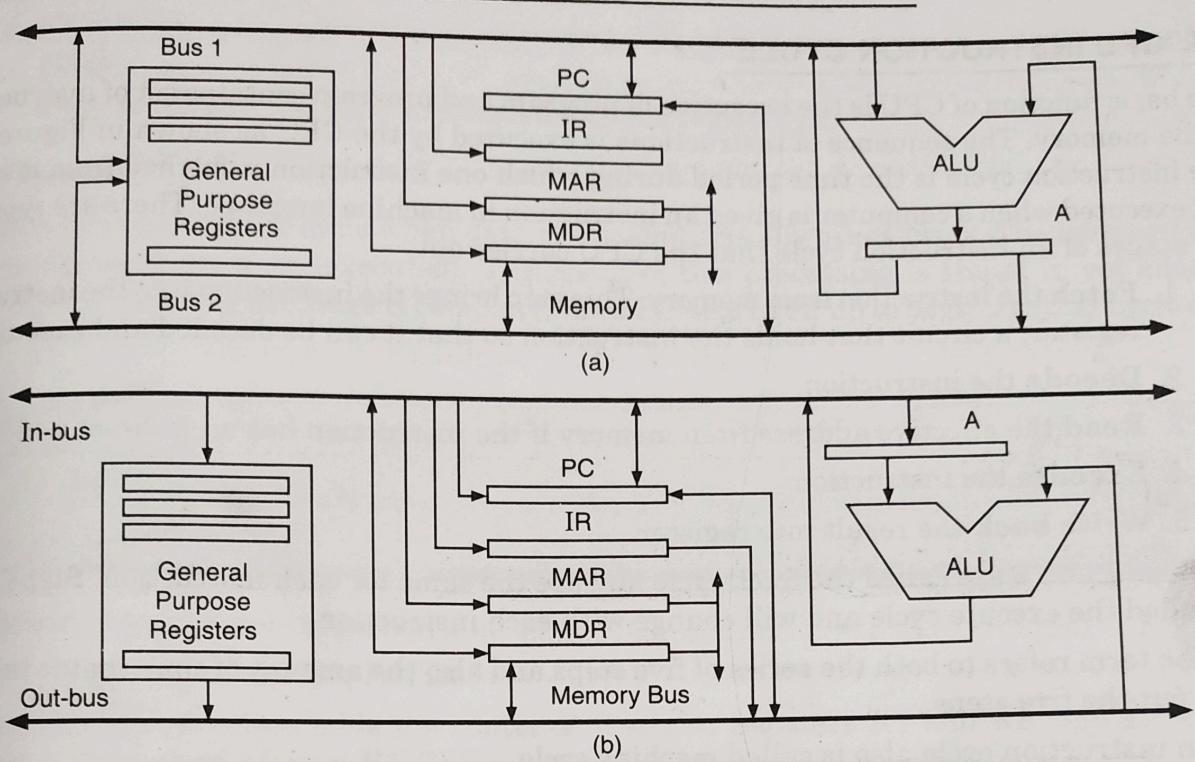


Fig. 6.3. Two-bus organizations. (a) An Example of Two-Bus Datapath
 (b) Another Example of Two-Bus Datapath with in-bus and out-bus

6.2.3. Three-Bus Organization

In a three-bus organization, three different buses are used, where two buses may be used as source buses and the third is used as destination bus. The source buses move data out of registers (out-bus), and the destination bus may move data into a register (in-bus). Each of the two out-buses is connected to an ALU input point. The output of the ALU is connected directly to the in-bus. As can be expected, the more buses we have, the more data we can move within a single clock cycle. However, increasing the number of buses will also increase the complexity of the hardware. Figure 6.4 shows an example of a three-bus datapath.

Small Concept

Three-bus organization uses three buses two may use for inputs two may use for output, but one fully dedicated for inputs and one for output.

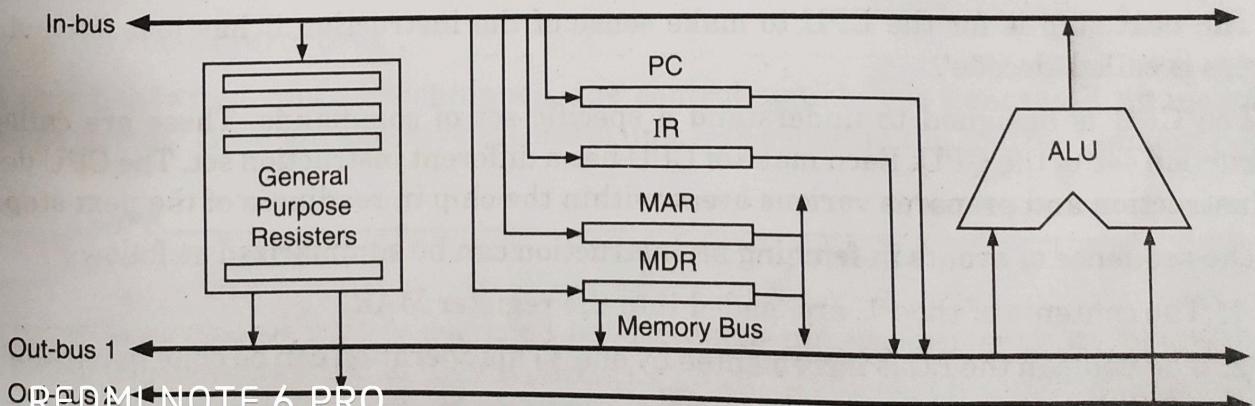


Fig. 6.4. Three-bus datapath