

Fig. 1.1.1 The First Generation: The first generation computers made use of: Vacuum tube technology,

1.2.1. **The First Generation:** The first generation computers made use of: Vacuum tube technology,

- Punched cards for data input,
- Punched cards and paper tape for output,
- Machine Language for writing programs,
- Magnetic tapes and drums for external storage.

The computers of the first generation were very bulky and emitted large amount of heat which required air conditioning. They were large in size and cumbersome to handle. They had to be manually assembled and had limited commercial use. The concept of operating systems was not known at that time. Each computer had a different binary coded program called a machine language that told it how to operate.

1.2.2 The Second Generation:

In the second generation computers:

- Vacuum tube technology was replaced by transistorized technology,
- Size of the computers started reducing,
- Assembly language started being used in place of machine language,
- Concept of stored program emerged,
- High level languages were invented.

This was the generation of Transistorized Computers. Vacuum tubes were replaced by transistors. As a result, the size of the machines started shrinking. These computers were smaller, faster, more reliable and more energy efficient. The first transistorized computer was TX-0. The first large scale machines that took advantage of the transistor technology were the early supercomputers, Stretch by IBM and LARC by Sperry Rand. These machines were mainly developed for atomic energy laboratories. Typical computers of the second generation were the IBM 1400 and 7000 series, Honeywell 200 and General Electric.

1.2.3 The Third Generation:

The third generation computers were characterized by:

- Use of Integrated circuits,
- Phenomenal increase in computation speed,
- Substantial reduction in size and power consumption of the machines,
- Use of magnetic tapes and drums for external storage,
- Design of Operating systems and new higher level languages,
- Commercial production of computers.

This generation was characterized by the invention of Integrated Circuits (ICs). The IC combined electronic components onto a small chip which was made from quartz.

Later, even more components were fitted onto a single chip, called a semiconductor. This reduced the size even further. The weight and power consumption of computers decreased and the speed increased tremendously. Heavy emphasis was given to the development of software. Operating systems were designed which allowed the machine to run many different programs at once. A central program monitored and co-ordinate the computer's memory. Multiprogramming

was made possible, whereby the machine could perform several jobs at the same time. Computers achieved speeds of executing millions of instructions per second. Commercial production became easier and cheaper. Higher level languages like Pascal and Report Program Generator (RPG) were introduced and applications oriented languages like FORTRAN, COBOL, and PL/I were developed.

1.2.4 The Fourth Generation:

The general features of the fourth generation computers were:

- Use of Very Large Scale Integration,
- Invention of microcomputers,
- Introduction of Personal Computers,
- Networking,
- Fourth Generation Languages.

The third generation computers made use of 'Integrated Circuits that had 10- 20 components on each chip, this was Small Scale Integration (SSI). The Fourth Generation realized Large Scale Integration (LSI) which could fit hundreds of components on one chip and Very Large Scale integration (VLSI) which squeezed thousand of components on one chip. The Intel 4004 chip, located all the components of a computer (central processing unit, memory, input and output controls) on a single chip and microcomputers were introduced. Higher capacity storage media like magnetic disks were developed. Fourth generation languages emerged and applications software's started becoming popular.

1.2.5 The Fifth Generation:

Defining the fifth generation computers is somewhat difficult because the field is still in its infancy. The computers of tomorrow would be characterized by Artificial Intelligence (AI). An example of AI is Expert Systems. Computers could be developed which could think and reason in much the same way as humans. Computers would be able to accept spoken words as input (voice recognition).

Many advances in the science of computer design and technology are coming together to enable the creation of fifth generation computers. Two such advances are parallel processing where many CPUs work as one and advance in superconductor technology which allows the flow of electricity with little or no resistance, greatly improving the speed of information flow.

1.3 CLASSIFICATION OF COMPUTERS

Computers are broadly classified into two categories depending upon the logic used in their design as:

1.3.1 Analog computers:

In analog computers, data is recognized as a continuous measurement of a physical property like voltage, speed, pressure etc. Readings on a dial or graphs are obtained as the output, ex. Voltage, temperature; pressure can be measured in this way

1.3.2 Digital Computers:

These are high speed electronic devices. These devices are programmable. They process data by way of mathematical calculations, comparison, sorting etc. They accept input and produce output as discrete signals representing high (on) or low (off) voltage state of electricity. Numbers, alphabets, symbols are all represented as a series of 1s and 0s.

Digital Computers are further classified as General Purpose Digital Computers and Special Purpose Digital Computers. General Purpose computers can be used for any applications like accounts, payroll, data processing etc. Special purpose computers are used for a specific job like those used in automobiles, microwaves etc.

Another classification of digital computers is done on the basis of their capacity to access memory and size like:

Small Computers:

- i. **Microcomputers:** Microcomputers are generally referred to as Personal Computers (PCs). They have Smallest memory and less power. They are widely used in day to day applications like office automation, and professional applications, ex. PCAT, Pentium etc.
- ii. **Note Book and Laptop Computers:** These are portable in nature and are battery operated. Storage devices like CDs, floppies etc. and output devices like printers can be connected to these computers. Notebook computers are smaller in physical size than lap top computers. However, both have powerful processors, support graphics, and can accept mouse driven input.
- iii. **Hand Held Computers:** These types of computers are mainly used in applications like collection of field data.

They are even **smaller** than the note book computers.

- **Hybrid Computers:** Hybrid Computers are a combination of Analog and Digital computers. They combine the speed of analog computers and accuracy of digital computers. They are mostly used in specialized applications where the input data is in an analog form i.e. measurement. This is converted into digital form for further processing. The computers accept data from sensors and produce output using conventional input/output devices.

- **Mini Computers:** Mini computers are more powerful than the micro computers. They have higher memory capacity and more storage capacity with higher speeds. These computers are mainly used in process control systems. They are mainly used in applications like payrolls, financial accounting, Computer aided design etc. ex. VAX, PDP-11

- **Mainframe Computers:** Main frame computers are very large computers which process data at very high speeds of the order of several million instructions per second. They can be linked into a network with smaller computers, micro computers and with each other. They are typically used in large organizations, government departments etc. ex. IBM4381, CDC

- **Super Computers:** A super computer is the fastest, most powerful and most expensive computer which is used for complex tasks that require a lot of computational power. Super computers have multiple processors which process multiple instructions at the same time. This is known as parallel processing. These computers are widely used in very advanced applications like weather forecasting, processing geological data etc. ex. CRAY-2, NEC - 500, PARAM.

1.4 APPLICATIONS OF COMPUTERS

Today computers find widespread applications in all activities of the modern world. Some of the major application areas include:

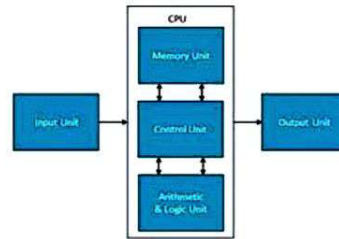
1.4.1 Scientific, Engineering and Research:

This is the major area where computers find vast applications. They are used in areas which require lot of experiments, mathematical calculations weather forecasting, and complex mathematical and engineering applications. Computer Aided Design (CAD) and Computer

1.6 COMPONENTS OF A COMPUTER SYSTEM

The basic parts of computer system are:

- Input Unit
- The Central Processing Unit
- Output Unit



1.6.1 The Input Unit:

Input devices are the devices which are used to feed programs and data to the computer. The input system connects the external environment with the computer system. The input devices are the means of communication between the user and the computer system. Typical input devices include the keyboard, floppy disks, mouse, microphone, light pen, joy stick, magnetic tapes etc. The way in which the data is fed into the computer through each of these devices is different. However, a computer can accept data only in a specific form. Therefore these input devices transform the data fed to them, into a form which can be accepted by the computer. These devices are a means of communication and interface station between the user and the computer systems.

Thus the functions of the input unit are :

- Accept information (data) and programs.
- convert the data in a form which the computer can accept.
- provide this converted data to the computer for further processing.

1.6.2 The Central Processing Unit:

This is the brain of any computer system. The central processing unit or CPU is made of three parts:

- The control unit.
- The arithmetic logic unit
- The primary storage unit

The Control Unit :

The Control Unit controls the operations of the entire computer system. The control unit gets the instructions from the programs stored in primary storage unit interprets these instructions and subsequently directs the other units to execute the instructions. Thus it manages and coordinates the entire computer system.

The Arithmetic Logic Unit:

The Arithmetic Logic Unit (ALU) actually executes the instructions and performs all the calculations and decisions. The data is held in the primary storage unit and transferred to the ALU whenever needed. Data can be moved from the primary storage to the arithmetic logic unit a number of times before the entire processing is complete. After the completion, the results are sent to the output storage section and the output devices.

The Primary Storage Unit:

This is also called as Main Memory. Before the actual processing starts the data and the instructions fed to the computer through the input units are stored in this primary storage unit. Similarly, the data which is to be output from the computer system is also temporarily stored in the primary memory. It is also the area where intermediate results of calculations are stored. The main memory has the storage section that holds the computer programs during execution. Thus the primary unit:

- Stores data and programs during actual processing
- Stores temporary results of intermediate processing
- Stores results of execution temporarily

1.6.3 Output Unit:

The output devices give the results of the process and computations to the outside world. The output units accept the results produced by the computer, convert them into a human readable form and supply them to the users. The more common output devices are printers, plotters, display screens, magnetic tape drives etc.

1.7 Execution Cycle

A program residing in the memory unit of a computer consists of a sequence of instructions. These instructions are executed by the processor by going through a cycle for each instruction.

In a basic computer, each instruction cycle consists of the following phases:

1. Fetch instruction from memory.
2. Decode the instruction.
3. Read the effective address from memory.
4. Execute the instruction.

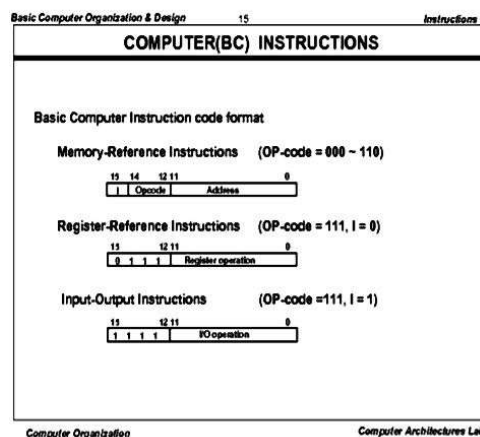
1.8 Instruction categories:

Computer Instructions: The basic computer has three instruction code formats, as shown in Fig. Each format has 16 bits. The operation code (opcode) part of the instruction contains three bits and the meaning of the remaining 13 bits depends on the operation code encountered. A memory-reference instruction uses 12 bits to specify an address and one bit to specify the addressing mode *I*. *I* is equal to 0 for direct address and to 1 for indirect address. The register reference instructions are recognized by the operation code *III* with a 0 in the leftmost bit (bit 15) of the instruction.

A register-reference instruction specifies an operation on or a test of the AC register. An operand from memory is not needed; therefore, the other 12 bits are used to specify the operation or test to be executed. Similarly, an input-output instruction does not need a reference to memory and is recognized by the operation code *III* with a 1 in the leftmost bit of the instruction. The remaining 12 bits are used to specify the type of input-output operation or test performed.

The type of instruction is recognized by the computer control from the four bits in positions 12 through 15 of the instruction. If the three opcode bits in positions 12 through 14 are not equal to *III*, the instruction is a memory-reference type and the bit in position 15 is taken as

the addressing mode *I*. If the 3-bit opcode is equal to *III*, control then inspects the bit in position 15. If this bit is 0, the instruction is a register-reference type. If the bit is 1, the instruction is an input-output type. Only three bits of the instruction are used for the operation code. It may seem that the computer is restricted to a maximum of eight distinct operations. However, since register-reference and input-out instructions use the remaining 12 bits as part of the operation code, the total number of instructions can exceed eight.



The instructions for the computer are listed in Table. The symbol designation is a three-letter word and represents an abbreviation intended for programmers and users. The hexadecimal code is equal to the equivalent hexadecimal number of the binary code used for the instruction. By using the hexadecimal equivalent we reduced the 16 bits of an instruction code to four digits with each hexadecimal digit being equivalent to four bits. A memory-reference instruction has an address part of 12 bits.

The address part is denoted by three x's and stand for the three hexadecimal digits corresponding to the 12-bit address. The last bit of the instruction is designated by the symbol *I*. When *I* = 0, the last four bits of an instruction have a hexadecimal digit equivalent from 0 to 6 since the last bit is 0. When *I* = 1, the hexadecimal digit equivalent of the last four bits of the instruction ranges from 8 to E since the last bit is 1.

The symbol designation is a three-letter word and represents an abbreviation intended for programmers and users. The hexadecimal code is equal to the equivalent hexadecimal number of the binary code used for the instruction. By using the hexadecimal equivalent we reduced the 16 bits of an instruction code to four digits with each hexadecimal digit being equivalent to four bits. A memory-reference instruction has an address part of 12 bits.

The address part is denoted by three x's and stand for the three hexadecimal digits

1.9 Memory

A memory is just like a human brain. It is used to store data and instructions. Computer memory is the storage space in the computer, where data is to be processed and instructions required for processing are stored. The memory is divided into large number of small parts called cells. Each location or cell has a unique address, which varies from zero to memory size minus one. For example, if the computer has 64k words, then this memory unit has $64 * 1024 = 65536$ memory locations. The address of these locations varies from 0 to 65535.

Memory is primarily of three types –

- Cache Memory
- Primary Memory/Main Memory
- Secondary Memory

Cache Memory

Cache memory is a very high speed semiconductor memory which can speed up the CPU. It acts as a buffer between the CPU and the main memory. It is used to hold those parts of data and program which are most frequently used by the CPU. The parts of data and programs are transferred from the disk to cache memory by the operating system, from where the CPU can access them.

Advantages

The advantages of cache memory are as follows –

- Cache memory is faster than main memory.
- It consumes less access time as compared to main memory.
- It stores the program that can be executed within a short period of time.
- It stores data for temporary use.

Disadvantages

The disadvantages of cache memory are as follows –

- Cache memory has limited capacity.
- It is very expensive.

Primary Memory (Main Memory)

Primary memory holds only those data and instructions on which the computer is currently working. It has a limited capacity and data is lost when power is switched off. It is generally made up of semiconductor device. These memories are not as fast as registers. The data and instruction required to be processed resides in the main memory. It is divided into two subcategories RAM and ROM.

Characteristics of Main Memory


- These are semiconductor memories.
- It is known as the main memory.
- Usually volatile memory.
- Data is lost in case power is switched off.
- It is the working memory of the computer.
- Faster than secondary memories.
- A computer cannot run without the primary memory.

Secondary Memory

This type of memory is also known as external memory or non-volatile. It is slower than the main memory. These are used for storing data/information permanently. CPU directly does not access these memories, instead they are accessed via input-output routines. The contents of secondary memories are first transferred to the main memory, and then the CPU can access it. For example, disk, CD-ROM, DVD, etc.

Characteristics of Secondary Memory

- These are magnetic and optical memories.
- It is known as the backup memory.
- It is a non-volatile memory.
- Data is permanently stored even if power is switched off.
- It is used for storage of data in a computer.
- Computer may run without the secondary memory.



Access time in RAM is independent of the address, that is, each storage location inside the memory is as easy to reach as other locations and takes the same amount of time. Data in the RAM can be accessed randomly but it is very expensive.

RAM is volatile, i.e. data stored in it is lost when we switch off the computer or if there is a power failure. Hence, a backup Uninterruptible Power System (UPS) is often used with computers. RAM is small, both in terms of its physical size and in the amount of data it can hold.

RAM is of two types –

- Static RAM (SRAM)
- Dynamic RAM (DRAM)

Static RAM (SRAM)

The word **static** indicates that the memory retains its contents as long as power is being supplied. However, data is lost when the power gets down due to volatile nature. SRAM chips use a matrix of 6-transistors and no capacitors. Transistors do not require power to prevent leakage, so SRAM need not be refreshed on a regular basis.


There is extra space in the matrix, hence SRAM uses more chips than DRAM for the same amount of storage space, making the manufacturing costs higher. SRAM is thus used as cache memory and has very fast access.

Characteristic of Static RAM

- Long life
- No need to refresh
- Faster
- Used as cache memory
- Large size
- Expensive
- High power consumption

Dynamic RAM (DRAM)

DRAM, unlike SRAM, must be continually **refreshed** in order to maintain the data. This is done by placing the memory on a refresh circuit that rewrites the data several hundred times per second.



DRAM is used for most system memory as it is cheap and small. All DRAMs are made up of memory cells, which are composed of one capacitor and one transistor.

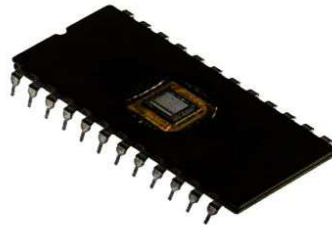
Characteristics of Dynamic RAM

- Short data lifetime
- Needs to be refreshed continuously
- Slower as compared to SRAM
- Used as RAM
- Smaller in size
- Less expensive
- Less power consumption

2. ROM stands for **Read Only Memory**. The memory from which we can only read but cannot write on it. This type of memory is non-volatile. The information is stored permanently in such memories during manufacture. A ROM stores such instructions that are required to start a computer. This operation is referred to as **bootstrap**. ROM chips are not only used in the computer but also in other electronic items like washing machine and microwave oven.



- Smaller in size
 - Less expensive
 - Less power consumption
2. ROM stands for **Read Only Memory**. The memory from which we can only read but cannot write on it. This type of memory is non-volatile. The information is stored permanently in such memories during manufacture. A ROM stores such instructions that are required to start a computer. This operation is referred to as **bootstrap**. ROM chips are not only used in the computer but also in other electronic items like washing machine and microwave oven.



Let us now discuss the various types of ROMs and their characteristics.

MROM (Masked ROM)

The very first ROMs were hard-wired devices that contained a pre-programmed set of data or instructions. These kind of ROMs are known as masked ROMs, which are inexpensive.

PROM (Programmable Read Only Memory)

PROM is read-only memory that can be modified only once by a user. The user buys a blank PROM and enters the desired contents using a PROM program. Inside the PROM chip, there are small fuses which are burnt open during programming. It can be programmed only once and is not erasable.

EPROM (Erasable and Programmable Read Only Memory)

EPROM can be erased by exposing it to ultra-violet light for a duration of up to 40 minutes. Usually, an EPROM eraser achieves this function. During programming, an electrical charge is trapped in an insulated gate region. The charge is retained for more than 10 years because the charge

has no leakage path. For erasing this charge, ultra-violet light is passed through a quartz crystal window (lid). This exposure to ultra-violet light dissipates the charge. During normal use, the quartz lid is sealed with a sticker.

EEPROM (Electrically Erasable and Programmable Read Only Memory)

EEPROM is programmed and erased electrically. It can be erased and reprogrammed about ten thousand times. Both erasing and programming take about 4 to 10 ms (millisecond). In EEPROM, any location can be selectively erased and programmed. EEPROMs can be erased one byte at a time, rather than erasing the entire chip. Hence, the process of reprogramming is flexible but slow.

Advantages of ROM

The advantages of ROM are as follows –

- Non-volatile in nature
- Cannot be accidentally changed
- Cheaper than RAMs
- Easy to test
- More reliable than RAMs
- Static and do not require refreshing
- Contents are always known and can be verified

1.10 Input/output devices

The functioning of a computer system is based on the combined usage of both input and output devices. Using an input device we can give instructions to the computer to perform an action and the device reverts back to our action through an output device.

We shall discuss the various input and output devices which can be connected to a computer, along with their functions. Also, some sample questions based on this topic have been given further below in this article.

Let us first discuss the exact definition of an input and output device:

Input Device Definition: A piece of equipment/hardware which helps us enter data into a computer is called an input device. For example keyboard, mouse, etc.

Output Device Definition: A piece of equipment/hardware which gives out the result of the entered input, once it is processed (i.e. converts data from machine language to a human-understandable language), is called an output device. For example printer, monitor, etc.

List of Input Devices

Given below is the list of the most common input devices along with brief information about each of them.

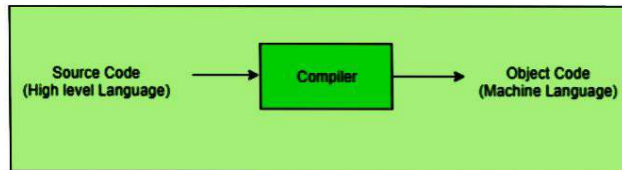
1. Keyboard

- A simple device comprising keys and each key denotes either an alphabet, number or number commands which can be given to a computer for various actions to be performed
- It has a modified version of typewriter keys

The language processor that reads the complete source program written in high level language as a whole in one go and translates it into an equivalent program in machine language is called as a Compiler.

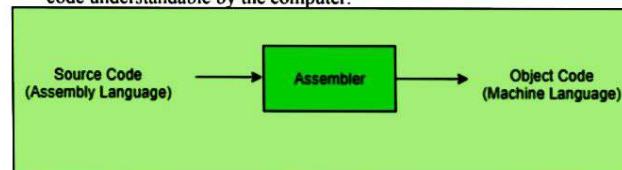
Example: C, C++, C#, Java

In a compiler, the source code is translated to object code successfully if it is free of errors. The compiler specifies the errors at the end of compilation with line numbers when there are any errors in the source code. The errors must be removed before the compiler can successfully recompile the source code again.>



1.14 Assembler –

The Assembler is used to translate the program written in Assembly language into machine code. The source program is a input of assembler that contains assembly language instructions. The output generated by assembler is the object code or machine code understandable by the computer.



Assembler divides these tasks in two passes:

- **Pass-1:**
 1. Define symbols and literals and remember them in symbol table and literal table respectively.
 2. Keep track of location counter
 3. Process pseudo-operations
- **Pass-2:**
 1. Generate object code by converting symbolic op-code into respective numeric op-code
 2. Generate data for literals and look for values of symbols

1.14 Interpreter:

The translation of single statement of source program into machine code is done by language processor and executes it immediately before moving on to the next line is called an

interpreter. If there is an error in the statement, the interpreter terminates its translating process at that statement and displays an error message. The interpreter moves on to the next line for execution only after removal of the error. An Interpreter directly executes instructions written in a programming or scripting language without previously converting them to an object code or machine code.

Example: Perl, Python and Matlab.

Difference between Compiler and Interpreter –

Compiler

A compiler is a program which converts the entire source code of a programming language into executable machine code for a CPU.

Compiler takes large amount of time to analyze the entire source code but the overall execution time of the program is comparatively faster.

Compiler generates the error message only after scanning the whole program, so debugging is comparatively hard as the error can be present anywhere in the program.

Generates intermediate object code.

Examples: C, C++, Java

Interpreter

interpreter takes a source program and runs it line by line, translating each line as it comes to it.

Interpreter takes less amount of time to analyze the source code but the overall execution time of the program is slower.

Its Debugging is easier as it continues translating the program until the error is met

No intermediate object code is generated.

Examples: Python, Perl

Zero-address instructions in a stack-organized computer are implied-mode instructions since the operands are implied to be on top of the stack.

Immediate Mode: In this mode the operand is specified in the instruction itself. In other words, an immediate-mode instruction has an operand field rather than an address field. The operand field contains the actual operand to be used in conjunction with the operation specified in the instruction. Immediate-mode instructions are useful for initializing registers to a constant value.

Register Mode: In this mode the operands are in registers that reside within the CPU. The particular register is selected from a register field in the instruction. A k-bit field can specify any one of 2^k registers.

Register Indirect Mode: In this mode the instruction specifies a register in the CPU whose contents give the address of the operand in memory. In other words, the selected register contains the address of the operand rather than the operand itself. Before using a register indirect mode instruction, the programmer must ensure that the memory address of the operand is placed in the processor register with a previous instruction. A reference to the register is then equivalent to specifying a memory address. The advantage of a register indirect mode instruction is that the address field of the instruction uses fewer bits to select a register than would have been required to specify a memory address directly.

Auto increment or Auto decrement Mode: This is similar to the register indirect mode except that the register is incremented or decremented after (or before) its value is used to access memory. When the address stored in the register refers to a table of data in memory, it is

necessary to increment or decrement the register after every access to the table. This can be achieved by using the increment or decrement instruction. However, because it is such a common requirement, some computers incorporate a special mode that automatically increments or decrements the content of the register after data access.

Obtain the operand from memory. Sometimes the value given in the address field is the address of the operand, but sometimes it is just an address from which the address of the operand is calculated. To differentiate among the various addressing modes it is necessary to distinguish between the address part of the instruction and the effective address used by the control when executing the instruction. The effective address is defined to be the memory address obtained from the computation dictated by the given addressing mode. The effective address is the address of the operand in a computational-type instruction. It is the address where control branches in response to a branch-type instruction.

Direct Address Mode: In this mode the effective address is equal to the address part of the instruction. The operand resides in memory and its address is given directly by the address field of the instruction. In a branch-type instruction the address field specifies the actual branch address.

Indirect Address Mode: In this mode the address field of the instruction gives the address where the effective address is stored in memory. Control fetches the instruction from memory and uses its address part to access memory again to read the effective address.

A few addressing modes require that the address field of the instruction be added to the content of a specific register in the CPU. The effective address in these modes is obtained from the following computation:

effective address = address part of instruction + content of CPU register

The CPU register used in the computation may be the program counter, an index register, or a base register. In either case we have a different addressing mode which is used for a different application.

Relative Address Mode: In this mode the content of the program counter is added to the address part of the instruction in order to obtain the effective address. The address part of the instruction is usually a signed number (in 2^k 's complement representation) which can be either positive or negative. When this number is added to the content of the program counter, the result produces an effective address whose position in memory is relative to the address of the next instruction. To clarify with an example, assume that the program counter contains the number 825 and the address part of the instruction contains the number 24. The instruction at location 825 is read from memory during the fetch phase and the program counter is then incremented by one to 826. The effective address computation for the relative address mode is $826 + 24 = 850$. This is 24 memory locations forward from the address of the next instruction. Relative addressing is often used with branch-type instructions when the branch address is in the area surrounding the instruction word itself. It results in a shorter address field in the instruction format since the relative address can be specified with a smaller number of bits compared to the number of bits required to designate the entire memory address.

Indexed Addressing Mode: In this mode the content of an index register is added to the address part of the instruction to obtain the effective address. The index register is a special CPU register that contains an index value. The address field of the instruction defines the beginning address of a data array in memory. Each operand in the array is stored in memory relative to the beginning address. The distance between the beginning address and the address of the operand is