

1. INTRODUCTION – DIGITAL PROTOTYPING

There is practically no aspect of our live today, which is not influenced by the work of engineers. The buildings and equipment we use, the vehicles we travel in and the roads and rails they use are all direct products of engineering activity. The food we eat is grown and processed with the assistance of engineering products. Engineers design and construct the equipment, which prints books, manufactures medicines and produces television images and fast communications.

Engineering forms the largest single economic activity of most countries, and provides the basis for human prosperity.

It can be said that one of the basic paradigms of the contemporary industrial society is producing huge variety of products and services. Many of the products are of great complexity and are subjected to rapid changes and innovations. In such case keeping the producing going has been achieved by organizing large teams of specialists to collaborate in the products' development and manufacture. Today, these teams work under increasing pressure to develop products of high performance and reliability at low cost and in shorter and shorter time-scales.

In view of this pressure, it is not surprising that engineers have turned to machines to assist them in the task of product development and manufacture. The machines involved are computers, and their task is information processing: they are used to assist in the definition and processing of information connected with the design of the product, and with the organization and management of the manufacturing systems.

It is commonly recognized that the product lifecycle passes through three basic stages as shown in Figure 1-1:

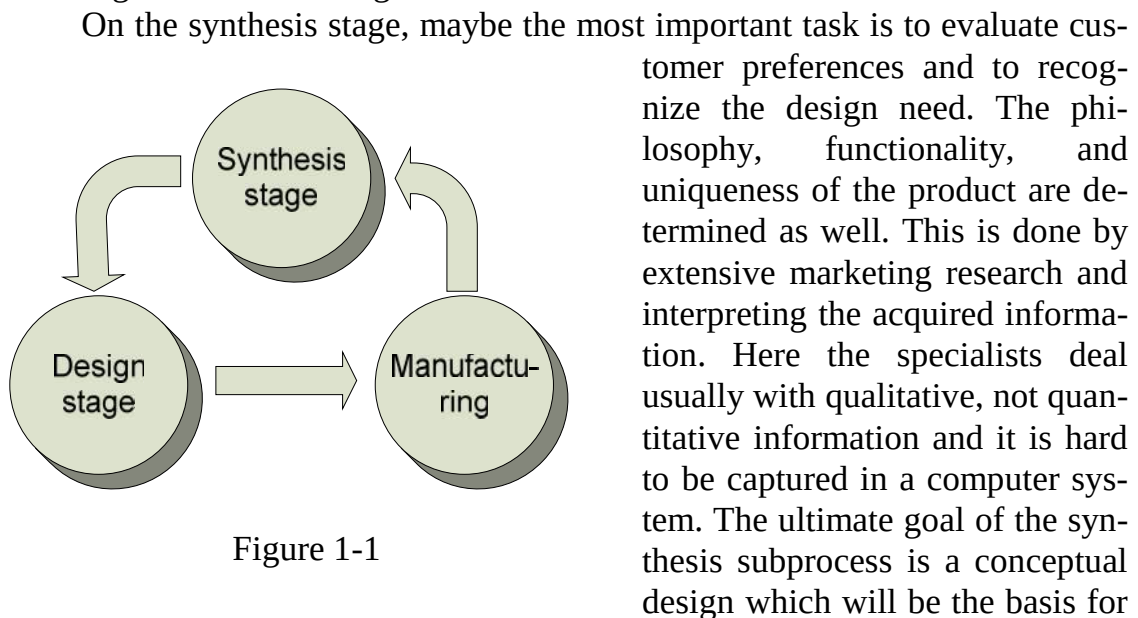


Figure 1-1

the subsequent product development.

The design stage is crucial for the product performance, cost, reliability, and even feasibility. The efforts and funds spent here can make the product much more competitive which could be a base for generation of successful modifications or other types of products development.

In most industries, product requirements and design criteria drive the engineering process. As its name implies, functional design favors function over geometry. Rather than modeling geometry first and then hoping that the form satisfies all the design criteria, in the functional design method, the designer or engineer makes sure the product operates correctly given the design criteria prior to finalizing the product's shape. If functional design is done well, the geometry will be the result of the design process rather than the input to it.

Typical Design Requirements

- o **Strength** (material, size, weight, load conditions, mechanical behavior, safety factors);
- o **Power** (speed, torque, momentum, power transmission, lubrication, safety);
- o **Temperature** (cooling, heat dissipation);
- o **Vibration and motion** (frequency response/eigenvalues/damping)
- o Wear resistance (surface treatments, plating, tolerances/life cycle/wear/durability/lubrication);
- o **Sound restrictions/considerations** (insulation/packaging);
- o **Optical characteristics** (color, surface characteristics, refraction, transparency, chromatic aberration, aesthetics);
- o **Electrical and magnetic characteristics;**
- o **Cost** (materials, packaging, eco-sustainability, spare parts, stock parts, standard sizes vs. custom made, maintenance, manufacturing and assembly methods, time restrictions).

Oftentimes, these requirements will conflict with each other; for example, improving strength or durability often will increase cost, and so on.

To answer the constant demands of shortening the product lifecycle, and ensuring quality, considerable efforts have been made to develop suitable software allowing the best part of the activities on all three stages to be done in virtual reality. Significant success has been achieved by introducing knowledge based and expert systems used in the synthesis stage, or computer aided process planning (CAPP) and machining simulating software for the manufacturing. In any case, the software created to aid the design process has been progressing steadily through the last three decades and now there is substantial number of integrated, highly developed parametric CAD systems on the market. This al-

allows information technologies to be used not only for drafting and modeling, but on all the phases of the design activities.

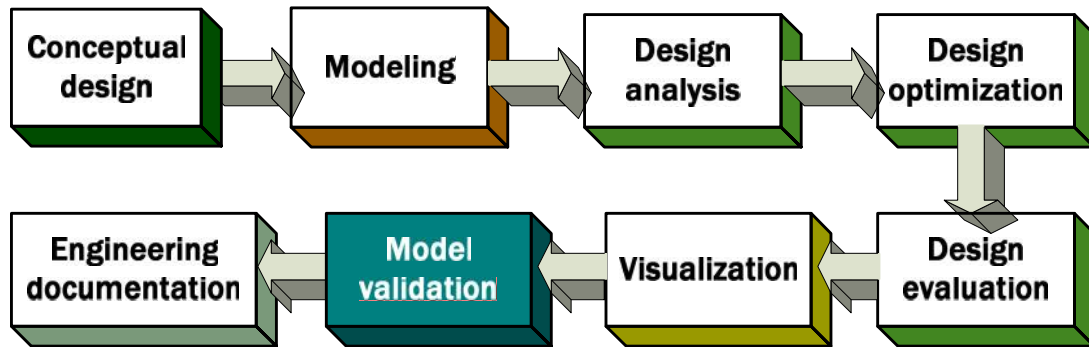


Figure 1-2

Figure 1-2 shows a flowchart the of the design process. Through all the steps on this stage models are used. These models could be of different type – form engineering 2D sketches and drafts to physical prototypes build for different purposes of the investigation and with various degree of approximation to the final product. With the developing of CAD/CAE products such models are mostly created, analyzed and evaluated in the computer environment with several great benefits:

- o Considerable decrease of the developing time;
- o Possibility to rapidly investigate number of alternative solutions and design alternations;
- o Improved accuracy of the design and eliminating errors;
- o Using libraries with standard parts and reusing parts and assemblies;
- o Reducing engineering personal requirements;
- o Reducing the cost of the design process by diminishing the necessity of physical prototypes building

The analysis subprocess begins with an attempt to put the conceptual design in the context of the abstracted engineering science, to evaluate the performance of the expected product. This needs design modeling as well as simulation.

The design analysis may involve stress — strain calculations, heat transfer computations, or the use of differential equations to describe the dynamic behavior of the system being designed. The computer can be used to aid in this analysis work. It is often necessary that specific programs be developed internally by the engineering analysis group to solve a particular problem of design analysis. In other situations, commercially available general purpose programs can be used. One integrated CAD/CAM systems often include or can be interfaced to engineering analysis software which can be called to operate on

the current design model.

Two of the most important examples of design analysis are:

- o analysis of mass properties;
- o finite-element stress and fatigue analysis.

The analysis of mass properties is the analysis feature of a CAD system that has probably the widest application. It provides properties of a solid object being analyzed, such as the surface area, weight, center of gravity, and moment of inertia (inertia tensor). For a plane surface (or a cross section of a solid object), the corresponding computations include the perimeter, area, and inertia properties.

Probably the most powerful analysis feature of a CAD system is the finite-element method. With this technique, the object is divided into a large number of finite elements (usually rectangular or triangular shapes – Fig 1-3a), which form an interconnecting network of concentrated nodes. By using a computer with significant computational capabilities, the entire object can be analyzed for stress — strain, heat transfer, and other characteristics by calculating the behavior of each node. By determining the interrelating behaviors of all the nodes in the system, the behavior of the entire object can be assessed.

Most of the CAD systems have the capability to define automatically the nodes and the network structure for the given object. The user simply defines certain parameters for the finite-element model (such as loads and types of supports) and the CAD system proceeds with the computations.

The output of the finite-element analysis is often best presented by the system in graphical format on the CRT screen for easy visualization by the user. For example, in stress — strain analysis of an object, the output may be shown in the form (Fig. 1-3b) of a deflected shape sometimes superimposed over the unstressed object. Color graphics can also be used to accentuate the comparison before and after deflection of the object. If the finite-element analysis indicates behavior of the design which is undesirable, the designer can modify the shape and recompute the finite-element analysis for the revised design.

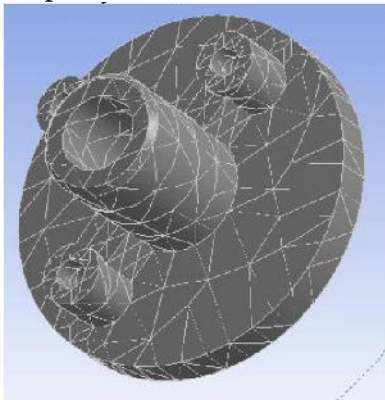


Figure 1-3a

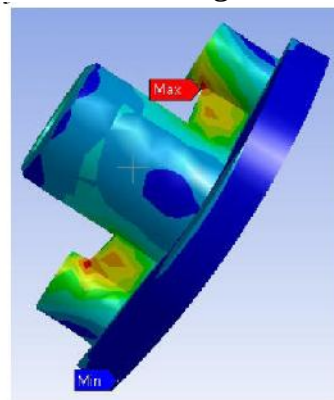


Figure 1-3b

Design optimization is an important phase of the analysis process as the optimum design is an important factor for product competitiveness on the market. Optimization can include a particular part as well as the construction as a whole. One particular construction could be optimized using different criteria such as weight, form, dynamic stress, fuel consumption, productivity and many others. For this purpose a goal function which includes all the criteria and factors influencing these criteria must be created. There are number of computer oriented methods for finding extremes of such a goal function. For example a shape optimization could be achieved using FEA on the basis of equal stress.

Once the major elements of the design have been analyzed and their nominal dimensions determined, the **design evaluation** phase starts. Here some basic tasks are performed:

- **Checking the accuracy of the design.** Many CAD systems include semiautomatic dimensioning and tolerancing routines which assign size specifications to objects indicated by the user. This help to reduce the possibility of dimensioning errors.
- **Interference checking.** This involves the analysis of an assembled structure in which there is a risk that the components of the assembly may occupy the same space. This risk occurs in the design of large complex structures (planes, automobiles, engines) or chemical devices, and other complicated piping structures.
- **Kinematics and dynamic simulating.** The available kinematics packages provide the capability to animate the motion of simple designed mechanisms such as hinged components and linkages. This capability enhances the designer's visualization of the operation of the mechanism and helps to ensure against interference with other components. Commercial software packages are available to perform kinematical and dynamic analysis. This type of programs can be very useful to the designer in constructing the required mechanism to accomplish a specified motion and/or force.

The last phase of the design process is the design communication and documentation, which involves preparations of drawings, reports, and presentations. Automated drafting involves the creation of hard-copy engineering drawings directly from the CAD database. Drawings are utilized to produce blueprints to be passed to the manufacturing process. Some of the graphics features of computer-aided design systems utilize features which include automatic dimensioning, generation of crosshatched areas, scaling of the drawing, and the capability to develop sectional views and enlarged views of particular part details. The ability to rotate the part or to perform other transformations of the image (e.g., oblique, isometric, or perspective views), can be of significant

assistance in drafting. Engineering drawings can be made to adhere to company drafting standards by programming the standards into the CAD system.

In the view of the above, we can divide the product development as consisting of two phases:

- o The digital phase with computer-aided design (CAD), computer-aided engineering (CAE), product simulation, product information management (PIM), and a bill of materials;
- o Physical phase with prototype manufacturing, physical testing, and final refinement and validation of the design).

Obviously, when the digital phase can cover most of the steps as shown in Figure 1-2 (shown on white background), we can speak of digital prototyping. Digital Prototyping is relatively new term which attempts to summarize the software, procedures and organization giving manufacturers the ability to virtually explore a complete product before it is built. The concept of Digital Prototyping has been around for over a decade, particularly since software companies such as Autodesk, PTC, Siemens PLM , and Dassault began offering computer-aided design combined with computer aided engineering software capable not only of creating accurate 3D models, but providing possibilities to analyze, optimize and evaluate these models. Digital Prototyping is a viable alternative for small-to-midsized manufacturers. It seeks to link otherwise unconnected, siloed activities shown in Figure 1-2. The realm of Digital Prototyping ends when the digital product and the engineering bill of materials are complete.

In the CAD environment the conceptual design can be presented as geometrical model inside the computer and thus the conceptual design must also be part of digital prototyping. Here, together with the establishing the general requirements and part layout, one of the most important task is to provide vision for the overall shape and form of the product.

Table 1-1 summarizes the activities performed on each step of the design stage and the relevant software provided by four major vendors on the market.

Table 1-1

	Autodesk	Dassault	Siemens	PTC
Conceptual Design Parts layout; Shape design; Performance requirements.	Alias Studio, AutoCAD	CATIA Live CATIA Concept design	NX Industrial design & styling	Pro/CONCEPT
Modeling Part modeling; Assembly model-	Inventor	CATIA Solid Works	NX6 Solid Edge	Pro/ENGINEER

ing. Sheet metal				
Design Analysis				
Mass properties; Interference check- ing; Finite-element analysis.	Inventor Stress analysis	CATIA structural analysis, Cosmos works	Solid edge FEA Simulation	Pro/ENGINEER Mechanism Dy- namics
Design evaluation				
Dimensioning and tolerancing; Kinematics and dynamic analysis; Fluid flow and thermal analysis.	Inventor design accelerator, Inventor Dynamic simulator	CATIA Toleranc- ing, CATIA systems dynamic behavior, Cosmos motion, Solid works flow simulator	Solid edge Simu- lation, NX Mechanical simulation, NASTRAN	Pro/ENGINEER Tolerance Analysis Exten- sion Pro/ENGINEER Advanced Mechanica Pro/ENGINEER Mechanism Dy- namics
Electrical design	AutoCAD Electrical	CATIA Systems LogicaElectrical design	NX Electrome- chanical Design	Pro/ENGINEER Spark Analysis
Cable and Routing	Cable and Har- ness Design, Tube, Pipe and Flexible Hose Design	CATIA Wire and Harness design CircuitWorks SolidWorks Rout- ing	Xpres Route, Solid Edge Wir- ing	Pro/ENGINEER Piping and Ca- bling Extension
Visualization	Inventor studio, 3D Studio Max, Showcase	SolidWorks Ani- mator FeatureWorks	Virtual studio +	

Table 1 shows considerable variety of software products capable of aiding the most tasks of the design process. The right product must be chosen carefully on the basis of the concrete and future projects, cost, availability of trained engineering personnel, possibilities to train new staff members, possibilities to organize the information flow etc. One additional question which arises here is: Is the design possible with the available software? If not, additional third party software has to be sought for (getting a global minimum or maximum of the multivariable function in order to optimize the design, or solving the set of differential equations to study the dynamic behavior of a product are two examples). The necessary computational routines are to be found in mathematical packages such as MATLAB, MathCAD, Mathematica etc. Additional way to deal with the problem is developing customized programs and packages. All the software listed above can host application routines written in some high level programming languages (AutoCAD and Inventor for example use Visual basic for Application or Visual C++). This approach however imposes additional costs upon the project and must be utilized carefully.

It is clear that even relatively small products as kitchen robot or automobile or airplane seat have their internal complexity and require different kind of specialists to work together on the same project - mechanical and electrical engineers, ergonomists, designers and others. That leads to the necessity to organize the information flow in the best possible way. Here of great importance are recently, on a mass scale introduced systems of data management giving possibilities to control the data flow from the beginning to the end of the product lifecycle. Such systems are known as PDM (Product Data Management) or PLM (Product Lifestyle Management).

As the digital prototyping could be considered as a small part of PLM it can be managed by such a system, but it can also have its own system of data organization which is better solution for small-to-midsized enterprises. Typical example here is the solution provided by Autodesk to manage the data flow on all stages of the digital prototyping process. Figure 1-4 represents a diagram of all activities on the design stage together with the software solutions involved. The core of the system is the Vault package which manages all engineering files, regardless of the file type. Files can be Autodesk Inventor, AutoCAD, FEA, CAM, Microsoft Word, Microsoft Excel, or any other file used in the design process.

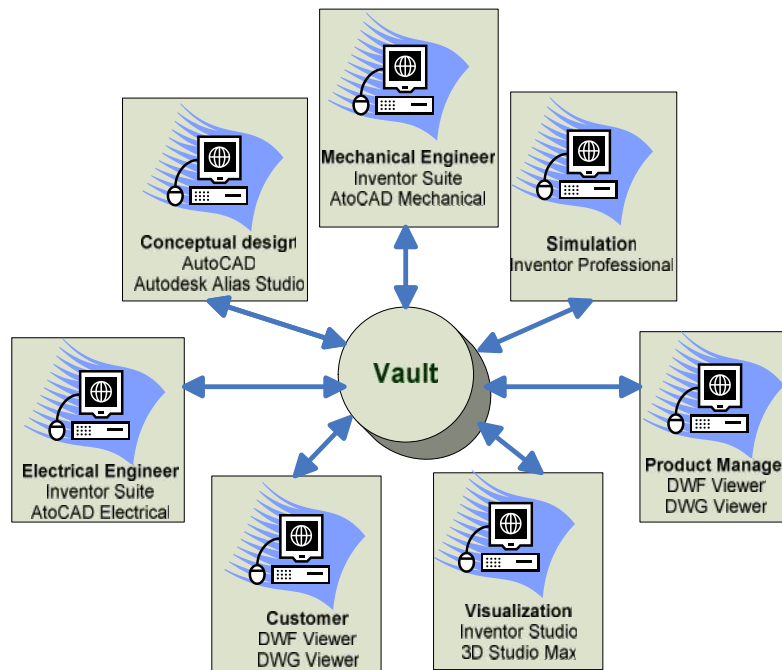


Figure 1-4

The system manager can organize all files and keep them in one location for easy access. All file versions are retained, so one never misplaces or replaces past versions. The vault stores each version of a file, along with all the files dependencies, giving a living history of the project as a team works on it. The Vault also stores file properties such as the user name, date, and comments

for rapid searching and retrieving. Each member of a team must have a unique login name and password. Team members check out files to prevent more than one member from editing the same file at the same time. Once a file is checked back into the vault, team members can refresh their local copies of the model file to get the latest version from the vault, so all members of the design team work together.

This is of course a very broad-brush description of the engineering process, and the detail at every stage will vary considerably according to the numbers of people involved and the nature and complexity of the product. In this book we are particularly concerned with the process of design, and in this respect, aircraft engines or computer systems (for example) are highly complex products in which the design involves large teams and is tightly constrained by technical factors. Conversely, in some areas a design may be the product of a single designer or small team, or factors such as aesthetics or fashion may have a dominant role.

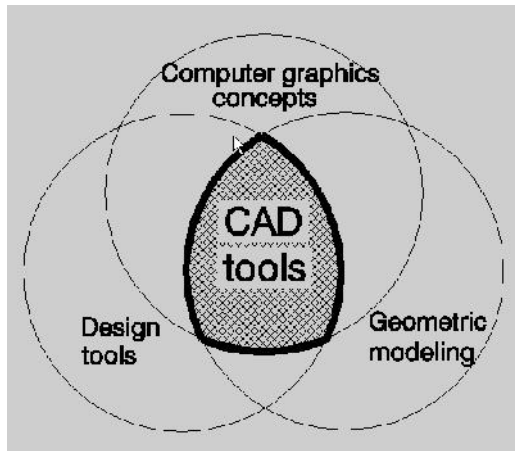


Figure 1-5

Summarizing, CAD tools can be defined as the intersection of three sets: geometric modeling, computer graphics and design tools. Fig. 1-5 shows such definition. As can be perceived from this figure, the abstracted concepts of geometric modeling and computer graphics must be applied to serve the design process. Based on implementation in a design environment, CAD tools can be defined as the design tools (analysis codes, heuristic procedures, design practices, etc.) being augmented

by computer hardware and software throughout its various phases to achieve the design goal efficiently and competitively. The level of augmentation determines the design capabilities of the various CAD/CAM systems and the effectiveness of the CAD tools they provide. Designers will always require tools that provide them with fast and reliable solutions to design situations that involve iterations and testing of more than one alternative. CAD tools can vary from geometric tools, such as manipulations of graphics entities and interference checking, on one extreme, to customized applications programs, such as developing analysis and optimization routines, on the other extreme. In between these two extremes, typical tools currently available include tolerance analysis, mass property calculations, and finite element modeling and analysis.

Questions:

1. Specify the basic stages of the product lifecycle;

2. List some typical design requirements;
3. What kind of activities are included in the analysis stage of the design process?
4. What kind of activities are included in the design evaluation stage of the design process?
5. What is digital prototyping?

2. CAD/CAM HARDWARE

2.1 INTRODUCTION

CAD/CAM systems are usually known for their fast interactive response and graphics display. It is a common question to ask: what are the differences between a CAD/CAM system and a conventional computer system? The answer is related to the two basic elements of the system, that is, hardware and software. The differences in hardware lie, in general, in the specialized input and output devices typically required by a CAD/CAM system to handle interactive graphics inputs and displays.

The first system that permitted a designer to graphically interact with a computer was SKETCHPAD, developed by Ivan Sutherland in 1962 - 63. Graphical interaction was achieved by the use of a light pen to directly touch items on the screen. During the 1970s CAD systems were developed so that mainframe computers could be used for designing large projects such as aircraft or buildings. During the 1980s professional CAD software was built up for Workstations and gradually for personal computers, so that now computer-aided design is available in a wide range of levels and prices. Designers can choose from a variety of software for different stages of the design cycle.

The majority of today's CAD/CAM systems utilize open hardware architecture and standard operating systems. Open hardware architecture implies that CAD/CAM vendors no longer design and manufacture their own hardware platforms. Instead the CAD/CAM industry relies upon the giant general-purpose computer companies and smaller firms that specialize in engineering workstations.

CAD/CAM systems based on either the microcomputer or workstation concept represent a distinct philosophy or trend in hardware technology, which is based on a distributed (stand-alone) but networked (linked) environment..

2.2 INPUT DEVICES

The user of a CAD/CAM system spends much time sitting at a workstation communicating and interacting with the computer to develop a particular engineering design. The user utilizes both input and output devices that comprise a workstation to achieve the design task. These devices are universal and mostly independent of the types of CAD/CAM. Software drivers are required to enable the host application program, i.e., the CAD/CAM software, to interpret the information received from input devices as well as send information to output devices.

A number of input devices are available. Input devices may include alphanumeric keyboards, programmed-function keyboards, digitizing tablets, lightpens, electronic styluses, trackballs, mouse systems, and touch input devices. These devices are used to input the two possible types of information: text and graphics.

There are three classes of graphics input devices:

- locating devices,
- digitizers,
- image-input devices.

Text-input devices are the alphanumeric (character-oriented) keyboards.

Locating devices, or locators, provide a position or location on the screen. These include lightpens, mice, digitizing tablets and styluses, joysticks, trackballs, thumbwheels, touchscreens, and touchpads. The keyboard arrow (cursor direction) keys are inadequate for most graphics applications and therefore are not considered here. Locating devices typically operate by controlling the position of a cursor on the screen. Thus, they are also referred to as cursor-control devices. The popularity of window and icon-oriented user interfaces has stimulated the further development and enhancement of locating devices. Normally, locators do not only have positioning functionality; they also provide other functions and graphical input modes such as picking and choosing, tracing, and sketching. Picking or pointing means selecting a displayed item or entity on the screen. All locating devices, with the exception of the joystick, which may provide three-dimensional input, are two-dimensional input devices. Three-axis input devices have been demonstrated using acoustic and mechanical techniques. There is great improvement in these designs during the last decade.

Image-input devices such as video frame grabbers and scanners comprise the third class of graphics-input devices. Electronic imaging is an area of relevance to image processing. This area may become significant to the CAD/CAM field if robot vision systems are to be driven by CAD/CAM databases. Video digitizers is another area where digitizers are connected to a video source, whether it is a video camera or recorder. The resolution of video digitizers is determined by that of the original source.

Image scanners are another form of image input devices. They are used to convert flat paper drawings or plots into digital bit-map form. They range in size and have been applied to help in the conversion of existing engineering blueprints into CAD databases. Some scanners incorporate optical character recognition capability for common typewriter fonts.

2.2.1 KEYBOARDS

Conventional keyboards are text-only devices and form an essential and basic input device. They are typically employed to create/edit programs or to

perform word processing functions. These keyboards have been modified to perform graphics tasks by adding special function keys or attaching graphics-input devices such as mice to them. The programmable function keyboard (PFK) is another type that typically has pushbuttons that are programmed to eliminate extensive typing of commands or entering coordinate information. The pushbuttons are controlled by the software and may be assigned different functions at different phases of the software. PFK may be built as a separate unit, or buttons may just be integrated with a conventional keyboard.

In CAD/CAM systems, information entered through keyboards should be displayed back to the user on a screen for verification, as in the case with conventional computers. However, the situation is different in CAD/CAM due to the simultaneous display of text and interactive graphics. Various vendors have devised different alternatives. A workstation may have a dual screen. One is used as an exclusive alphanumeric terminal to display input text, commands, and user prompts. The other screen is used to display graphics. Thus, no loss of graphics display area is suffered. Other systems such as the Computervision workstation utilizes one screen for both text and graphics by reserving a small portion at the bottom of the display for text (This is also typical for AutoCAD program system). The user has the option to display more text by letting it overlap with the graphics. Irrespective of the text/graphics display organization, users prefer to have a means to obtain hard copies of the text, as it is input for various reasons. Hardcopy printers may be provided for this purpose. If they are not available, software functions may be available to generate session files or hard files of sessions' inputs, which may be printed later. Users should be warned that these files are usually hard to read, especially when functions such as backspaces, control keys, or digitizes are encountered in the files.

2.2.2 MOUSE SYSTEMS

The mouse was invented in the late 1960s as a location device but has only recently become fairly popular due to its convenient use with icons and pop-up and pull-down menus. Unlike the digitizing tablet, the mouse measures its relative movement from its last position, rather than where it is in relation to some fixed surface. There are two basic types of mice available: mechanical and optical. The mechanical mouse is a box with two metal wheels or rollers on the bottom whose axes are orthogonal in order to record the mouse motion in the *X* and *Y* directions. The roll of the mouse on any flat surface causes the rotation of the wheel, which is encoded into digital values via potentiometers or other means to sense the rotation.

Developed by Agilent Technologies and introduced to the world in late 1999, the optical mouse actually uses a tiny camera to take 1,500 pictures every second. Able to work on almost any surface, the mouse has a small, red light-emitting diode (LED) that bounces light off that surface onto a complimentary

metal-oxide semiconductor sensor (usually 16x16 pixel light sensitive matrix).

The sensor sends each image to a digital signal processor (DSP) for analysis. The DSP, operating at 18 MIPS (million instructions per second), is able to detect patterns in the images and see how those patterns have moved since the previous image. Based on the change in patterns over a sequence of images, the DSP determines how far the mouse has moved and sends the corresponding coordinates to the computer. The computer moves the cursor on the screen based on the coordinates received from the mouse. This happens hundreds of times each second, making the cursor appear to move very smoothly.

Optical mice have several benefits over wheeled mice:

No moving parts means less wear and a lower chance of failure.

There's no way for dirt to get inside the mouse and interfere with the tracking sensors.

Increased tracking resolution means smoother response.

They don't require a special surface, such as a mouse pad.

Working in 3D environment requires a lot of activities to manipulate the 3D model – panning, rotating, zooming etc. One quite recent development in this area are so called Space Mouse or Space Balls. These are input devices for powerful, two handed work style. They simultaneously pan, zoom, rotate, and position 2D and 3D objects with the controller in one hand, while the other hand selects, creates and edits with the standard mouse. Figure 2.1 shows such devices.



Figure 2.1

2.2.3 DIGITIZING TABLETS

Another class of graphics input devices, besides locating devices, is digitizer boards or tablets, or simply digitizers. They are considered as electronic drafting boards. A digitizer consists of a large synthesized electronic board with

a movable stylus called the cursor. It is a two-dimensional input device with high resolution and accuracy. Typical sizes are 36 x 48 and 48 x 72 inches. Available resolution and accuracy are up to 0.001 and 0.003 inch respectively.

A digitizing tablet is considered to be a locating as well as a pointing device. It is a small, low-resolution digitizing board often used in conjunction with a graphics display. The tablet is a flat surface over which a stylus or a puck (a hand-held cursor to differentiate it from a display screen cursor) can be moved by the user. The close resemblance of the tablet and stylus to paper and pencil contributes to its popularity as an input device in computer graphics. The

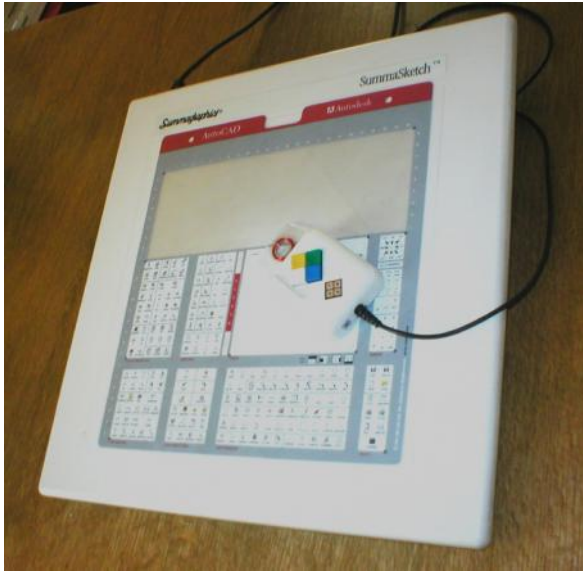


Figure 2.2

stylus is shaped like a pen, and a puck is a little hand-held box. The puck contains a rectile and at least one pushbutton. The rectile is engraved cross-hairs to help locate a point for digitizing (Fig. 2.2). Pressing the pushbutton sends the coordinates at the cross-hairs to the computer. Additional buttons may be available on the puck and may be programmed by the software for other functions than digitizing locations such as selecting alphanumeric font sizes or electronic symbols. Sizes of digitizing tablets range from 11 x 11 to 36 x 36 inches and per-

haps larger. A tablet's typical resolution is 0.005 inch or 200 dots per inch.

The tablet operation is based on sensitizing its surface area to be able to track the pointing element (stylus or puck) motion on the surface. Several sensing methods and technologies are used in tablets. The most common sensing technology is electromagnetic, where the pointing element generates an out-of-phase magnetic field sensed by a wire grid in the tablet surface (the pad). Other tablets are based on magnetostrictive techniques, in which a current pulse sent from the wire grid in the pad is picked up by the pointing element to calculate the x and y coordinates of the desired position. A third type of tablet is based on sonic or acoustic sensing. The pointing element in these tablets has a sound generator with strip microphones along the edges of the tablet to pick up the periodic sound to determine the x and y values. Fig 2.3 shows such a tablet.

The operational concept of the digitizing tablet is simple. The user moves the pointing element to the desired position and then interrupts the computer to accept the coordinate value of this position. In the case of a stylus, the user presses it against the tablet surface or depresses a pushbutton near its tip. This, in turn, activates a switching mechanism inside the stylus that picks (or sends) signals from (to) the tablet position and sends them to the stylus decoding logic

which stores the corresponding coordinates in the tablet's buffer registers. These registers are read by the application program as soon as it is interrupted by the depression of the stylus or its pushbutton. All digitizing tablets provide an origin, typically in the lower left corner, for the x and y coordinates, the x axis being horizontal and the y axis being vertical. Some tablets let the user move the origin to any point on the tablet's area.

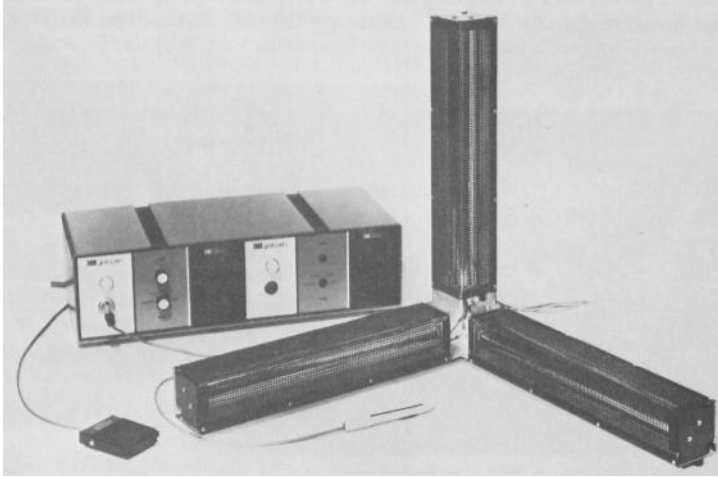


Figure 2.3

There are four relevant parameters to measure the performance of graphics input devices. These are **resolution**, **accuracy**, **repeatability**, and **linearity**. Some may be more significant to some devices than others.

The resolution of a device is defined as the smallest distance the device requires to recognize

two adjacent points. For example, if a digitizer has a resolution of 0.001 inch, the user cannot digitize or represent two points that are less than 0.001 inch apart. Resolution becomes important if a driver for the device is to be written.

Accuracy is defined as the error in the measurement of actual data by the input device. This parameter applies more to digitizers. The accuracy of a digitizer is measured by how closely a point is reported as to its actual location on the digitizer grid.

Repeatability measures the device ability to return to a given position. For example, if the same point is digitized many times, how close are the coordinates of the resulting point?

Linearity measures the response of a device to the user hand movements. How does the device, for example, increase or decrease the input coordinates in accordance with the user hand movements? These parameters are frequently used very loosely and should be carefully defined within the context of each input device of interest.



Figure 2.4

Figure 2.4 shows 3D digitizer and 3D laser scanner with 6 degrees of freedom based on the combination of an articulated arm and a laser head. Using optical triangulation, object shapes are captured in 3D by sweeping a laser profile across the surface.

A 3D scanner (Figure 2.4a) is a device that analyses a real-world object or environment to collect data on its shape and possibly its appearance (e.g. colour). The collected data can then be used to construct digital three-dimensional models.

Many different technologies can be used to build these 3D-scanning devices; each technology comes with its own limitations, advantages and costs. Many limitations in the kind of objects that can be digitized are still present, for example, optical technologies encounter many difficulties with shiny, mirroring or transparent objects. For example, industrial computed tomography scanning can be used to construct digital 3D models, applying non-destructive testing.

Collected 3D data is useful for a wide variety of applications. These devices are used extensively by the entertainment industry in the production of movies and video games. Other common applications of this technology include industrial design, orthotics and prosthetics, reverse engineering and prototyping, quality control/inspection and documentation of cultural artifacts.

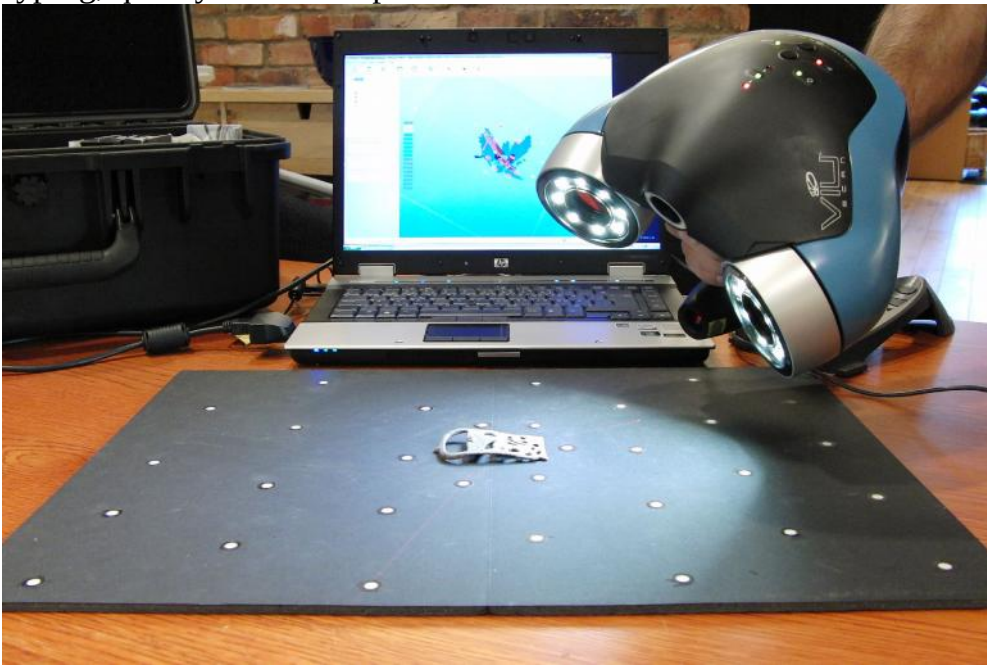


Figure 2.4a

There are a variety of technologies for digitally acquiring the shape of a 3D object. A well established classification divides them into two types: contact and non-contact. Non-contact solutions can be further divided into two main categories, active and passive. There are a variety of technologies that fall under each of these categories.

Contact: Contact 3D scanners (Fig. 2.4) probe the subject through physical touch, while the object is in contact with or resting on a precision flat surface plate, ground and polished to a specific maximum of surface roughness. Where the object to be scanned is not flat or can not rest stably on a flat surface, it is supported and held firmly in place by a fixture.

The scanner mechanism may have different forms.

Non-contact active: Active scanners emit some kind of radiation or light and detect its reflection or radiation passing through object in order to probe an

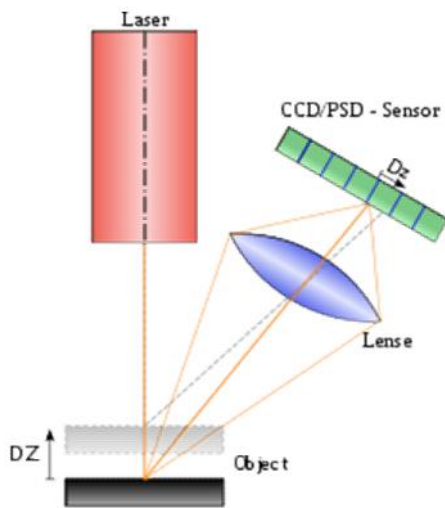
object or environment. Possible types of emissions used include light, ultrasound or x-ray.

The time-of-flight 3D laser scanner is an active scanner that uses laser light to probe the subject. At the heart of this type of scanner is a time-of-flight laser range finder. The laser range finder finds the distance of a surface by timing the round-trip time of a pulse of light. A laser is used to emit a pulse of light and the amount of time before the reflected light is seen by a detector is measured. Since the speed of light c is known, the round-trip time determines the travel distance of the light, which is twice the distance between the scanner and the surface. If t is the round-trip time, then distance is equal to $c \cdot t / 2$. The accuracy of a time-of-flight 3D laser scanner depends on how precisely we can measure the t time: 3.3 picoseconds (approx.) is the time taken for light to travel 1 millimetre.

The laser range finder only detects the distance of one point in its direction of view. Thus, the scanner scans its entire field of view one point at a time by changing the range finder's direction of view to scan different points. The view direction of the laser range finder can be changed either by rotating the range finder itself, or by using a system of rotating mirrors. The latter method is commonly used because mirrors are much lighter and can thus be rotated much faster and with greater accuracy. Typical time-of-flight 3D laser scanners can measure the distance of 10,000~100,000 points every second.

Time-of-flight devices are also available in a 2D configuration. This is referred to as a time-of-flight camera.

Triangulation based 3D laser scanners are also active scanners that use laser light to probe the environment. With respect to time-of-flight 3D laser scanner the triangulation laser shines a laser on the subject and exploits a camera to look for the location of the laser dot. Depending on how far away the laser strikes a surface, the laser dot appears at different places in the camera's field of view. This technique is called triangulation because the laser dot, the camera and the laser emitter form a triangle. The length of one side of the triangle, the distance between the camera and the laser emitter is known. The angle of the laser emitter corner is also known. The angle of the camera corner can be determined by looking at the location of the laser dot in the camera's field of view. These three pieces of information fully determine the shape and size of the triangle and give the location of the laser dot corner of the triangle. In most cases a laser stripe, instead of a single laser dot, is swept across the object to speed up the acquisition process.



2.3 OUTPUT DEVICES

2.3.1 GRAPHICS DISPLAYS

Output devices may include conventional plotters, hardcopy printers, and more importantly graphics display terminals. These terminals are quite different from the conventional video display, or video terminals, usually referred to as computer terminals which generally display only text. Graphics display terminals possess various local processors and controllers that can perform various graphics functions, such as transformations and graphics generation, locally at the hardware level to reduce the interactive time response between the user and the system.

Various display technologies are now available to the user to choose from. Among them it could be mentioned:

- liquid crystal display LCD;
- light emitting diodes LED;
- cathode ray tube CRT.

All these are based on the principle of converting computer signals, controlled by corresponding digital information into visible images at high speed. In recent times CRT was the most dominant technology but is gradually replaced by LCD (TFT) monitors for number of disadvantages of the former. Although today the CRT are very good as the computer graphics display of choice (because of its relatively low price and sufficient resolution), it is bulky, heavy, fragile, and currently limited in size to about 30 inches diagonally. For these as well as other reasons, fiat panel displays are becoming of increasing importance. All flat panel displays are raster refresh displays.

A **raster graphics device** can be considered a matrix of discrete cells each of which can be made bright. Thus, it is a point-plotting device. It is not possible except in special cases to directly draw a straight line from one addressable point, or pixel, in the matrix to another addressable point or pixel. The line can only be approximated by a series of dots (pixels) close to the path of the line. Only in the special cases of completely horizontal, vertical, or 45° lines will a straight line of dots or pixels result. All other lines will appear as a series of stair steps. This is called aliasing or the 'jaggies.' Antialiasing is addressed further on.

The most common method of implementing a raster graphics device utilizes a frame buffer. A frame buffer is a large, contiguous piece of computer memory. As a minimum there is one memory bit for each location or pixel (picture element) in the raster. This amount of memory is called a bit plane. A 512×512 element square raster requires 2^{18} ($2^9 = 512$; $2^{18} = 512 \times 512$) or 262,144 memory bits in a single bit plane. The picture is built up in the frame buffer 1 bit at a time. Since a memory bit has only two states (binary 0 or 1), a single bit plane yields a black-and-white display. Since the bit plane is a digital device,

while the raster CRT is an analog device, which requires an electrical voltage, conversion from a digital representation to an analog signal must take place when information is read from the frame buffer and displayed on the raster graphics device. This is accomplished by a digital-to-analog converter (DAC) for analog CRT devices. Each pixel in the frame buffer must be accessed and converted before it is visible on the raster. A schematic diagram of a single-bit-plane black-and-white frame buffer raster graphics device is shown in Fig. 2.5.

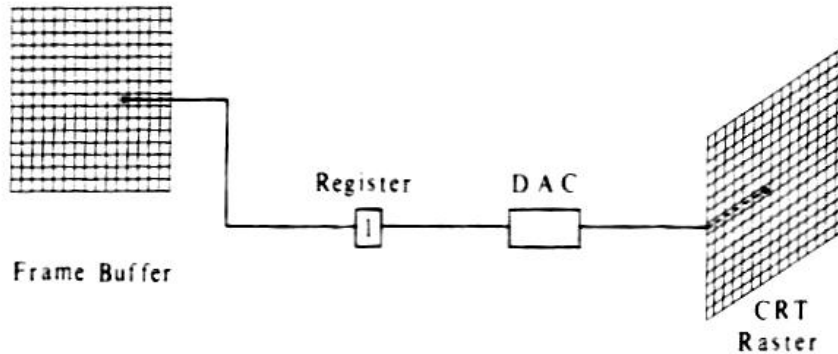


Figure 2.5

Color or gray levels can be incorporated into a frame buffer raster graphics device by using additional bit planes. Figure 2.10 schematically shows an N-bit-plane gray level frame buffer. The intensity of each pixel on the CRT is controlled by a corresponding pixel location in each of the N bit planes. The binary value (0 or 1) from each of the N bit planes is loaded into corresponding positions in a register. The resulting binary number is interpreted as an intensity level between 0 and $2^N - 1$. This is converted into a voltage between (dark) and $2^N - 1$ (full intensity) by the DAC. A total of 2^N intensity levels can be achieved. Figure 2.10 illustrates a system with three bit planes for a total of 8 (2^3) intensity levels. Each bit plane requires the full complement of memory for a given raster resolution; e.g., a three-bit-plane frame buffer for a 512 x 512 raster requires 786,432 ($3 \times 512 \times 512$) memory bits.

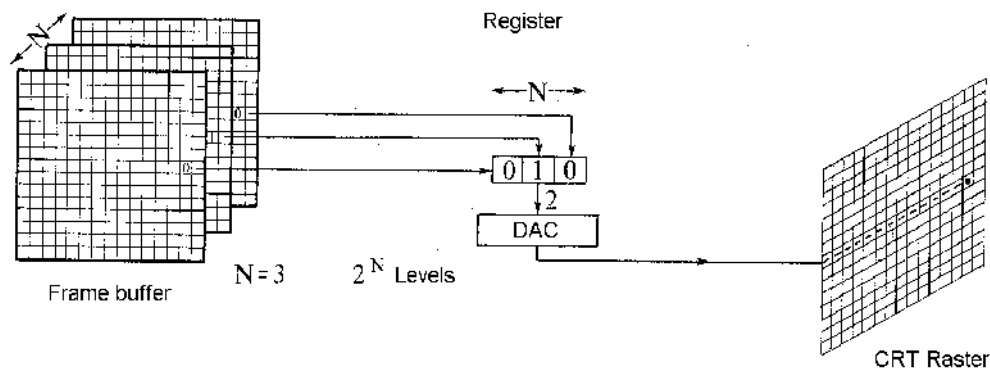


Figure 2.6

Table 2.1.

	Red	Green	Blue
Black	0	0	0
Red	1	0	0
Green	0	1	0
Blue	0	0	1
Yellow	1	1	0
Cyan	0	1	1
Magenta	1	0	1
White	1	1	1

Since there are three primary colors, a simple color frame buffer can be implemented with three bit planes, one for each primary color. Each bit plane drives an individual color gun for each of the three primary colors used in color video. These three colors are combined at the raster display to yield eight colors. The eight obtainable colors and appropriate binary codes are shown

in Table 2.1. A simple color raster frame buffer is shown schematically in Fig. 2.6 and 2.7

Additional bit planes can be used for each of the three-color guns. Figure 2.12 shows a schematic of a multiple-bit-plane color frame buffer with 8 bit planes per color, i.e., a 24-bit-plane frame buffer. Each group of bit planes drives an 8-bit DAC. Each group can generate 256 (2^8) shades or intensities of red, green, or blue. These can be combined into 16,777,216 [$(2^8)^3 = 2^{24}$] possible colors. This is a 'full' color frame buffer.

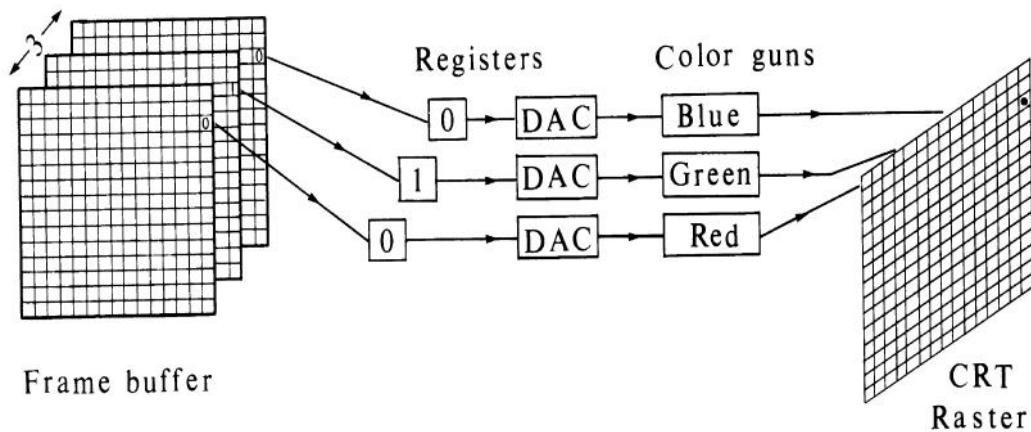


Figure 2.7

Because of the large number of pixels in a raster scan graphics device, achieving real-time performance and acceptable refresh or frame rates can be difficult. For example, if pixels are accessed individually with an average access time of 200 nanoseconds (200×10^{-9} second), then it requires 0.0524 second to access each of the pixels in a 512×512 frame buffer. This is equivalent to a refresh rate of 19 frames (pictures) per second, well below the required minimum refresh rate of 30 frames per second. A 1024×1024 frame buffer contains slightly more than 1 million bits (1 megabit) and at 200 nanoseconds average access time requires 0.21 second to access each of the pixels. This is 5 frames per second. A 4096×4096 frame buffer contains 16.78 million bits per memory

plane! At a 200-nanosecond access time per pixel it requires 0.3 second to access each of the pixels. To achieve a refresh rate of 30 frames per second a 4096 x 4096 raster requires an average effective access rate of 2 nanoseconds per pixel. Real-time performance with raster scan graphics devices is achieved by accessing pixels in groups of 16, 32, or 64 or more simultaneously. In the case of color frame buffers each pixel may contain up to 32 bits; i.e., all bit planes for an individual pixel are accessed together. With an average access time of 1600 nanoseconds for each group of pixels, real-time performance for 512 x 512 and 1024 x 1024 frame buffers is possible.

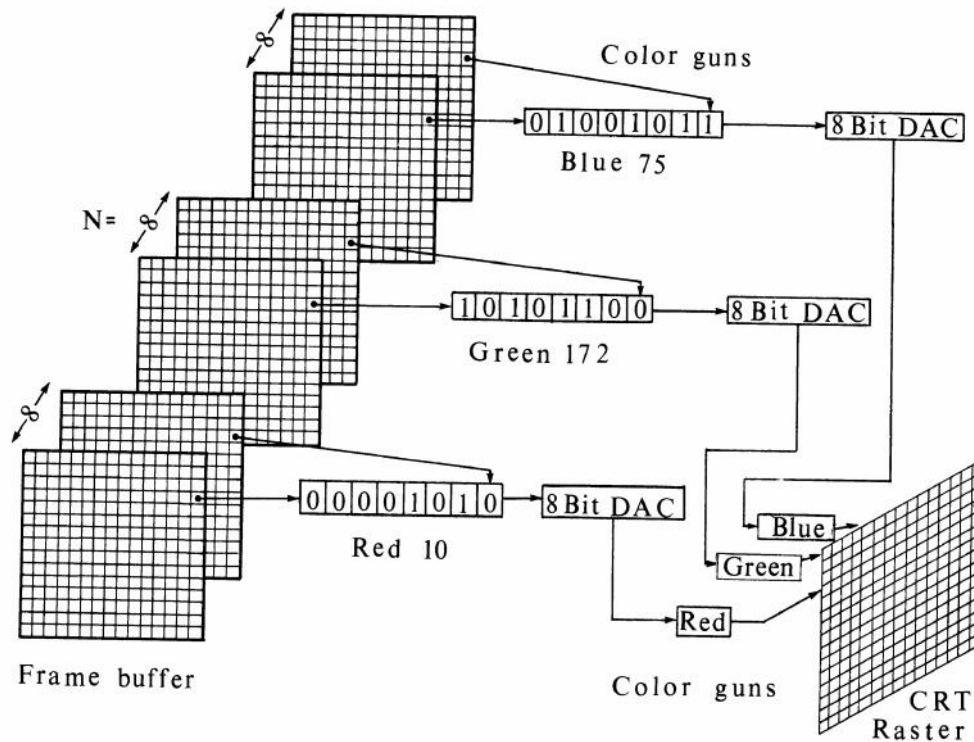


Figure 2.8

2.3.2 FLAT PANEL DISPLAYS

Flat panel displays are broadly divided into those based on active (light-emitting) and passive (light-modulating) technologies. Among the active technologies are flat CRTs, plasma-gas discharge, electroluminescent (EL) and vacuum fluorescent displays.

Of the active flat panel display technologies, plasma-gas discharge is currently most suitable for the relatively large sizes and high resolutions required by computer graphics applications.

Plasma-gas discharge, and liquid crystal displays have several operating characteristics in common. Each consists of a matrix of individual pixel locations on a raster. Each pixel must contain some mechanism, or material,

activated by application of either a voltage or a current, that either emits light or modulates incident light.

The more common, technique that activates a particular pixel uses row-column addressing of the raster. This technique requires that the display material have a switching threshold. Only when the switching threshold is exceeded does emission or modification of light occur. Part of the voltage or current required to activate an individual pixel is applied through the appropriate row and the other part through the appropriate column. The individual row or column voltage or current supplied is below the switching threshold. Together, it is above the switching threshold. Consequently, unwanted pixels along either the row or column are not activated. Only the desired pixel at the intersection of the row and column receives enough voltage or current to exceed the switching threshold and hence is activated.

The basic technology of a plasma or gas discharge display is quite simple. Essentially the display consists of a matrix of cells (the raster) in a glass envelope. Each cell is filled with a gas (usually neon or a neon/argon mixture) at low pressure (below atmospheric). When a sufficiently high voltage is applied, the gas dissociates, i.e., electrons are stripped from the atoms. The dissociated gas is called a plasma, hence the name — plasma display. When the electrons recombine, energy is released in the form of photons; and the gas glows with the characteristic bright orange-red hue.

The cell matrix is usually a panel which separates two additional ones — that is front and back panels embedded with wires. Wires in a front panel are vertical, wires in the back panel are horizontal. The vertical wires of the front panel represent the X coordinates on the screen while the horizontal wires on the back panel represent the Y coordinates. To light a particular pixel (x_0, y_0) the voltage difference between the vertical wire representing x_0 and horizontal wire representing y_0 is risen enough to cause a current flow between the two wires and thus exceeding the switching threshold. The schematic representation

of the plasma panel is given on Fig. 2.9.

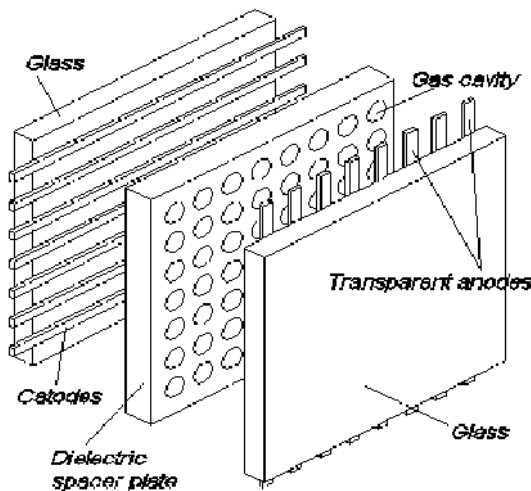


Figure 2.9

Plasma panels require relatively high voltage to operate and are relatively very expensive, so they are not normally used for portable machines.

When using row-column addressing, bistable pixel memory is highly desirable. With bistable memory, once activated a pixel remains activated until explicitly turned off. Bistable pixel memory eliminates the necessity of constantly refreshing

pixels. Consequently, no external memory is required to refresh the display. In addition, the display controller is simplified.

While plasma and electroluminescent displays are examples of active flat panel technologies, the liquid crystal display is an example of a passive technology. A liquid crystal display either transmits or reflects incident light. The polarizing characteristics of certain organic compounds are used to modify the characteristics of the incident light.

In the LCD displays, the middle panel contains pockets of clear liquid crystals which could be aligned by current passing between the front and the rear panels. The front panel of the LCD display is polarised glass, so the noncoherent light entering the system is polarised in xy plane, while in many cases, the back panel is a simple mirror. Here the middle panel of the LCD panel is used as second polarizer. If the crystals of the second polarizer are aligned in the same direction (xy plane), the polarised light is reflected off the back panel without having its plane of polarisation modified. The light passes back through the middle panel with very little loss of intensity. The pixel appears light (Figure 2.10).

Most of the LCD pannels have a backlignht and the rear panel is the first polarizer. In this case the light from the lamp passes or not through the liquid cristal middle panel or is stopped accornig to the cristals orientation – Figure 2.10.

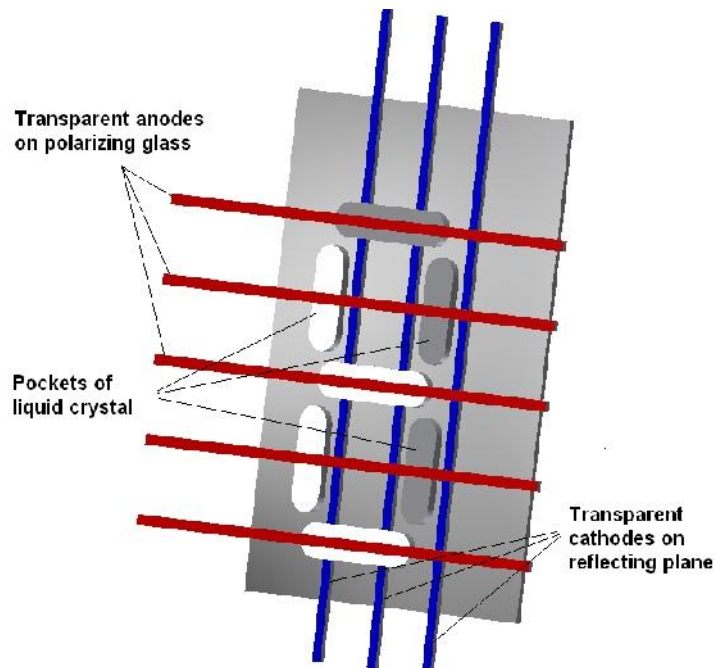


Figure 2.10

If the crystals are oriented the other way, they modify the direction of the polarisation and the light can no longer pass through the front panel. The pixel is dark.

A dark spot on a point (x_1, y_1) is created via matrix addressing. The point is selected by applying negative voltage $-V$ to the horizontal grid wire y_1 and

positive voltage $+V$ to the vertical grid wire x_1 . Neither $-V$ or $+V$ are big enough to cause the crystal to line up, but the difference is big enough to do so.

To display dots at (x_1, y_1) and (x_2, y_2) simultaneously we can not simply apply positive voltage to x_1 and x_2 as well as negative voltage to y_1 and y_2 . That would cause dots to appear also at the points (x_2, y_1) and (x_1, y_2) . Rather the points must be selected in succession one after the other and this selection process must be repeated to refresh the activation process.

The voltage difference needed in LCD displays is very small and the screen is lighted with the room light. The low energy requirements make LCD panels applicable for portable machines as well as for stand alone terminals. Low contrast image and slow respond time are two main disadvantages of the LCD panels. One solution is to light the back of the screen.

Liquid crystal displays are divided in two major types – **passive matrix and Active matrix (TFT)** and can contain active matrix elements, e.g., individual transistors or diodes at each pixel. Resolutions up to 100 pixels per inch are currently available.

In passive matrix systems, each row and each column of the display has its own driver, and to create an image, the matrix is rapidly scanned to enable every pixel to be switched on or off as required. As the current required to brighten a pixel increases (for higher brightness displays), and as the display gets larger, this process becomes more difficult since higher currents have to flow down the control lines. Also, the controlling current has to be present whenever the pixel is required to light up. As a result, passive matrix displays tend to be used mainly where small, simpler displays are required.

Active matrix displays solve the problem of efficiently addressing each pixel by incorporating a transistor (TFT) in series with each pixel which provides control over the current and hence the brightness of individual pixels. Lower currents can now flow down the control wires since these have only to programme the TFT driver, and the wires can be finer as a result. Also, the transistor is able to hold the current setting, keeping the pixel at the required brightness, until it receives another control signal. Future demands on displays will in part require larger area displays so the active matrix market segment will grow faster.

2.4.3 HARDCOPY PRINTERS AND PLOTTERS

Output devices of both printers and plotters are available to CAD/CAM systems for purposes such as creating checkplots for offline editing, and producing final drawings and documentation on paper. Relative to the drawing rate on screen, they are slow. Printers usually provide hard copies of text as well as graphics. Hardcopy devices, in general, employ one of two methods of plotting: vector or raster plotting. The two methods are very close in concept to refresh and raster displays respectively. Vector plotting can employ either absolute or

incremental plotting depending on whether the coordinates of the current point are measured relative to the absolute origin of the plot or to the last point plotted. In general, absolute plotting is preferred because it eliminates any errors or noise from the surrounding environment. Raster (or dot matrix) plotting is based on using the presence or absence of dots to draw lines and other geometric information present in the document. The resolution of the plotter may cause jaggedness in the hard copy similar to raster displays. For many years, plotters were the only means of obtaining high-quality outputs from a computer. Currently, various printers are available that provide either black and white or multicolor pictures.

The hardcopy devices that are available include **impact** and **nonimpact** devices. Impact devices include dot matrix printers and typically produce an image on paper by hammering a ribbon onto the surface of the paper – hence the name ‘impact printers’. Nonimpact devices utilize other methods and include pen, photographic, electrophotographic, electrostatic, thermal-transfer, and ink-jet plotters and/or printers. While most of these devices may be central and shared by all workstations, each workstation usually has its own hardcopy unit. The basic operations of some of these devices are described below.

Pen plotters. There are two common types of conventional pen plotters: flat-bed and drum. In the flat-bed plotter, the paper is stationary and the pen-holding mechanism can move in two axes. In the drum plotter, the paper is attached to a drum that rotates back and forth, thereby providing movement in one axis. The pen mechanism moves in the transverse direction to provide movement along the other axis. Pen plotters are considered vector plotters and employ multiple pens (four to eight) to provide varying line widths or colors. Pens may be of wet ink, ball-point, or felt-tip type. The plotting speed is typically limited by the speed of the writing pen used. Pen plotters require supervision to

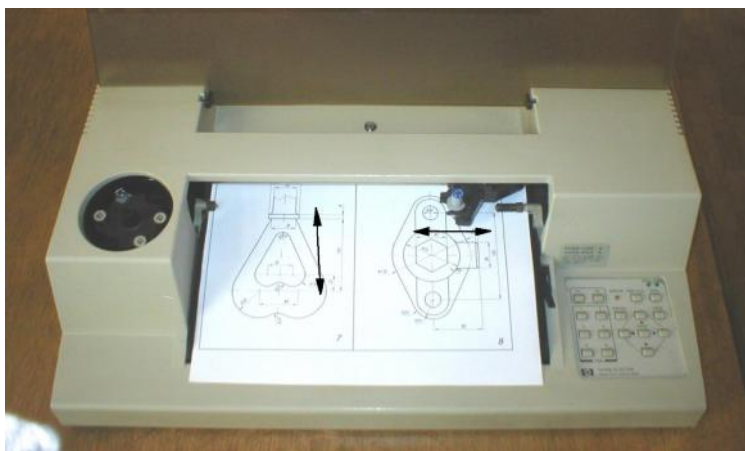


Figure 2.11

watch pen performance and are relatively inexpensive to acquire and operate. A view of a drum plotter is shown in Fig. 2.11

Pen plotters can produce high-quality line drawings for CAD applications. Prices start at approximately \$500, but typically range from \$3,000 to \$8,000 for

plotters having eight pens and handling paper up to A0 size. Although pen plotters can fill areas by putting lines close together, such fill requires considerable time, so pen plotters work best on line drawings. Pen plotters operate with vec-

tor graphics; that is, a line is plotted by placing the pen at a starting point and drawing to the end point, just as you would do by hand. The pen motion can be directly programmed, for example, by embedding into your program pen commands in HPGL or HPGL/2 (Hewlett-Packard Graphics Languages). Most CAD systems can directly output drawings to a plotter by using HPGL or HPGL/2.

Because pen plotters are vector devices, with lines drawn one after another, plotting time is proportional to the number of lines in the drawing. In contrast, for raster output devices the actual drawing time is relatively independent of the drawing complexity because in every case the entire paper is scanned with dots placed at appropriate positions. Of course, in plotting with CAD software, particularly for 3-D drawings with hidden lines, much of the plotting time may be taken up by software computations that are strongly dependent on the complexity of the drawing. When using a pen plotter you should not plot too many lines through a single point or in a very small region on the paper because the pen can eat through the paper if a point is drawn over too frequently. This situation can occur easily with CAD software, in which a complex subpart of the drawing may shrink to nearly a point on the scale of the overall drawing; therefore, you should carefully consider which portions of the drawing are to appear on the final pen plot. Some pen plotters also accept pencils, which are convenient when the user knows that part of the drawing will be erased later and modified by hand. Overall, pen plotters are very useful for providing line drawings of all sizes in a limited number (about eight) of colors.

Pen plotters are rapidly being replaced by raster plotters.

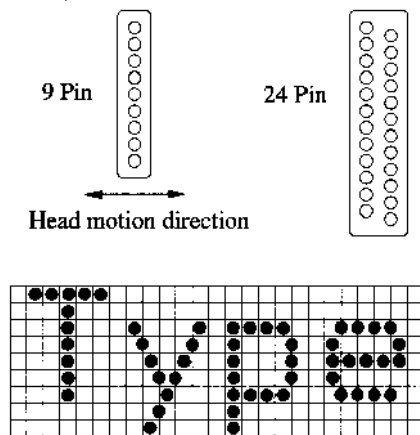


Figure 2.12

Dot-matrix printers. Dot-matrix printers are the least expensive output devices. Following common usage, here dot-matrix printer refers to the dot-matrix **impact printer**, even though essentially all other printers and plotters except pen plotters are dot-matrix (raster) devices. Although used mainly for text, the dot-matrix printer can produce graphics. Most dot-matrix impact printers have one or two lines of pins (also called wires, needles, or hammers) that are carried across the paper by the motion of the printhead. Most dot-matrix printers have either 9 or 24 pins,

with the 24-pin printer producing near 'letter quality' text. The term letter quality references the quality to a daisy wheel printer that prints in the same manner as a typewriter by impacting raised characters onto a ribbon above the paper. The dot-matrix printer works in a similar fashion, with each impact of a pin producing a single dot (an electromagnet is used to move the pins to the paper and back).

Figure 2.12 illustrates how the 9-pin printhead forms letters as it moves horizontally across the paper. The top 7 pins form the capital letters, which are above the baseline; the 2 bottom pins form the tails of lowercase letters such as y and p. Because dots are positioned horizontally at half (dot) spaces as well as at full spaces, all letters can fit within five horizontal spaces. Note that the 24-pin printhead in Fig. 2.12 is shown with two staggered lines of pins so that printed dots will overlap for increased resolution and smoothness in forming characters. In addition to forming characters, the dots can represent a drawing or picture. Typically, the 9-pin printer will produce graphics at 70-dpi resolution and the 24-pin printer at 180 to 300 dpi. Most dot-matrix impact printers are monochrome, although some models add one or more colors to black by shifting to additional ribbons. Because of the speed at which the pins hit the ribbon, dot-matrix printers are usually noisy; the other printers and plotters described in this section are relatively quiet.

Black and white laser printers. In recent years due to their increasing performance and dropping of the costs laser printers are gradually replacing most daisy wheel and dot-matrix printers used for high-quality text. The standard resolution for a laser printer is 600 dpi, although 1200-dpi printers are now available. Laser printer prices start at approximately \$300. Practically most laser printers are monochrome at the present time; however, color models (at relatively low prices) are becoming more common.

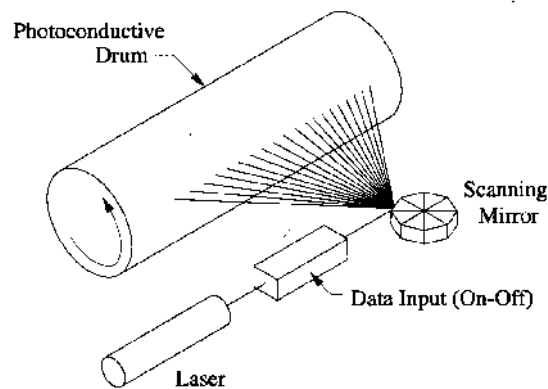


Figure 2.13

Figures 2.13 and 2.15 illustrate the optical system for a laser printer. A laser beam is modulated by data to be printed, and then reflected off a rotating polygonal mirror so that the beam scans along the axis of a rotating drum. The photoconductive coating of the drum is electrically charged, but loses the charge at points hit by the laser beam. In the remainder of the process, a black plastic powder (toner) is attracted to the remaining charge on the drum. The ink particles (positively charged) adhere to negatively charged (by the drum charge) paper, which moves across the back side of the drum. The toner is then melted into the paper by applying high temperature (fusser) to make the image permanent – Figure 2.14.

Figure 2.14 shows a color laser printer. A laser printer that prints in color uses four toner cartridges (CMYK) of cyan, magenta, yellow and black. The colors are applied one at a time to the drum and are then adhered to the paper. The toner cartridges deliver the toner to the photo conducting belt, which delivers it to the drum and then the paper.

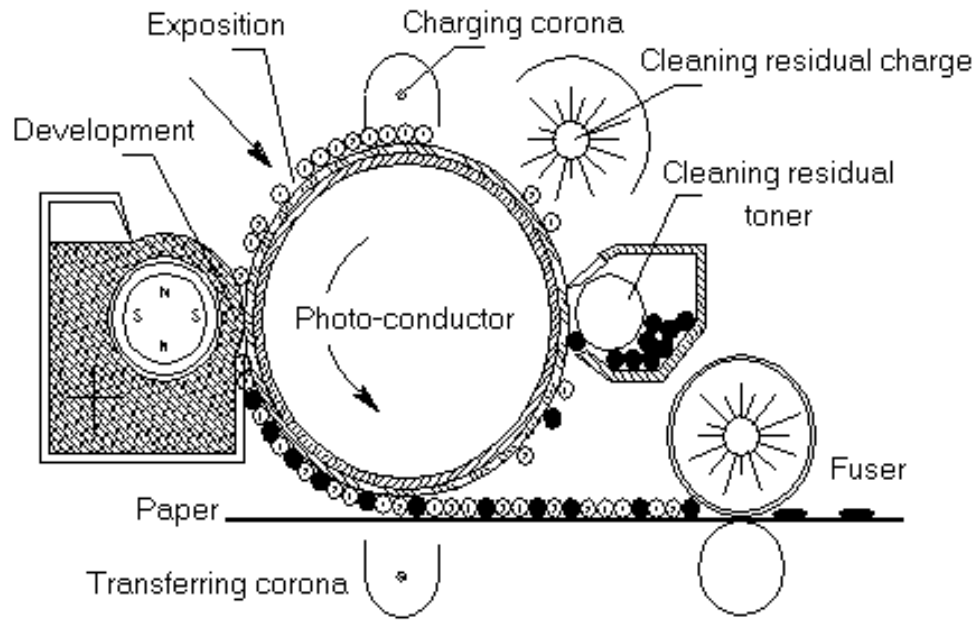


Figure 2.14

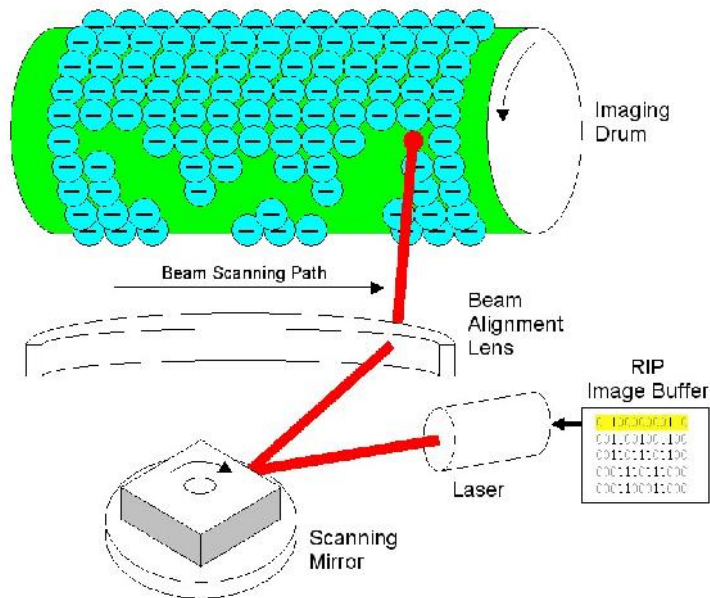


Figure 2.15

From Computer Desktop Encyclopedia
© 2004 The Computer Language Co. Inc.

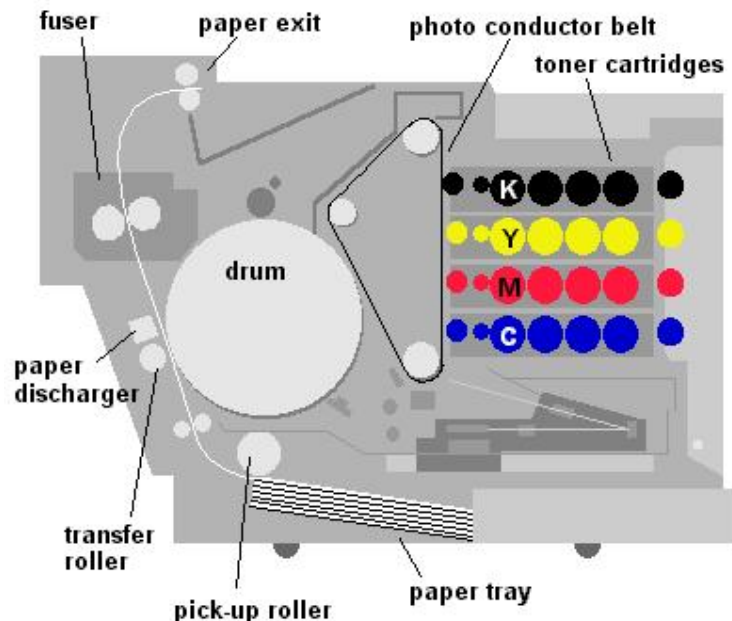


Figure 2.16

Inkjet printers. Monochrome ink-jet printers generally lie between dot-matrix and laser printers in price and resolution, whereas color ink jets are normally the least expensive option for full-color area fill applications. Most ink-jet printers for personal computers and workstations are drop-on-demand (impulse) systems in which a drop of ink is ejected from a small nozzle in response to a pressure pulse. Early systems were continuous devices in which a stream of ink breaks up into drops, which are then charged, and are either allowed to strike the paper or are deflected into a gutter, depending on whether or not a printed dot is desired at a given position on the paper.

Two common printer configurations for moving paper exist: the printhead containing ink-jet nozzles may follow the axial direction of a rapidly spinning drum to which the paper is attached, or the printhead may move back and forth across the paper, which moves up line by line. The latter configuration is most common and is the same as that used for a dot-matrix impact printer, except that the pin printhead is replaced by an ink-jet printhead. An ink-jet printhead typically contains 30 to 100 nozzles so that it can print rapidly as it sweeps across the page. The ink-jet printer is fast because the printhead can move continuously across the paper while ejecting drops; impact printers must move the pins to the paper and back before the printhead moves to the next position. In a third ink-jet configuration, a plotter pen is replaced by an ink-jet printhead; thus no time is needed to bring the pen to the paper at the beginning of a line and pick it up at the end.

The pressure pulse for drop-on-demand ink jets may be produced in two ways: the channel supplying ink may be squeezed (normally by a piezoelectric

crystal), or a short power pulse may rapidly heat a resistor and produce a bubble that forces a drop of ink out of the nozzle – Figure 2.17

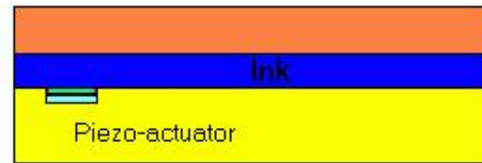


Figure 2.17

This latter method is called the thermal ink jet or bubble jet and is illustrated in Fig. 2.18. Monochrome ink-jet printers offer almost the same resolution as laser printers at somewhat less cost.

Figure 2.18 shows stages of the thermal ink-jet cycle, showing a cross section along the axis of a nozzle region:

(a) An ink meniscus forms across the nozzle orifice under equilibrium conditions (no ink motion) before the power pulse is applied to the heater resistor;

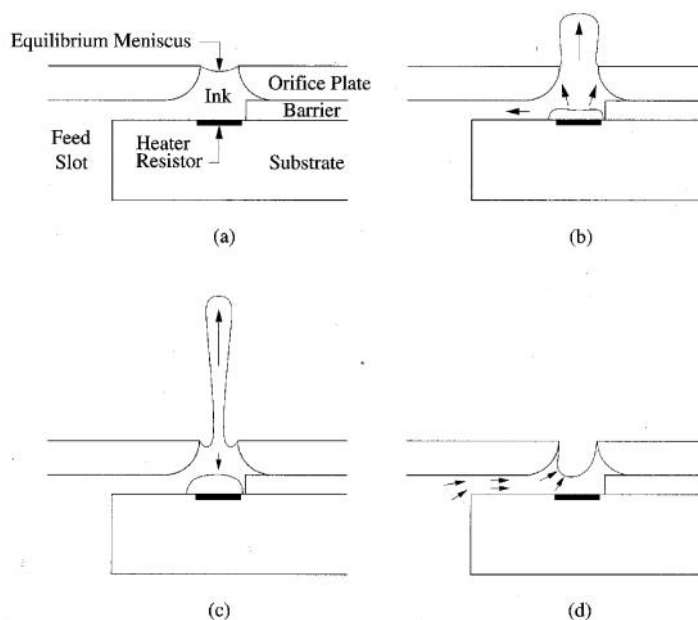


Figure 2.18

(b) A bubble grows over the resistor a short time after the power pulse is applied. A drop starts to eject from the orifice, and there is some backflow toward the feed slot (arrows show the ink motion);

(c) The ejected drop is nearly at the break-off point (when the menisci near the orifice join and the drop separates from

the remaining ink). Because the power pulse is over and the resistor is cooling, the bubble starts to collapse;

(d) The capillary forces exerted by the orifice meniscus (after drop ejection and bubble collapse) initiate refill. The refill period is the longest part of the cycle; it therefore determines the rate at which drops can be fired and thus the speed of the printer.

Color ink-jet printers mix dots of cyan, yellow, and magenta dyes to produce a relatively full range of colors by dithering (see below). Normally, a

separate cartridge of black ink is provided to ensure a good black color for text and graphics (in theory black can be produced by equally mixing cyan, yellow, and magenta). Currently, color ink-jet printers are priced from \$100 to \$1000 and produce resolutions of 300 to 1200 dpi. Sometimes quality is reduced by the ‘banding’ that results from the separate passes of the printhead across the paper and by lowered resolution for color shades produced by dithering. Color ink-jet printers often require special paper for the best output, whereas monochrome ink-jet printers often operate well with plain paper. The above discussion applies to liquid ink jets that use water-based inks. At the high end of the ink-jet market are the solid or phase-change ink-jet printers, which can produce fully saturated colors on plain paper. These ink jets start with color wax sticks, which are melted and squirted as drops onto the paper. When the drops strike the paper, they cool and solidify.

3D Printers

3D printing or additive manufacturing is a process of making three dimensional solid objects from a digital file. The creation of a 3D printed object is achieved using additive processes. In an additive process an object is created by laying down successive layers of material until the entire object is created. Each of these layers can be seen as a thinly sliced horizontal cross-section of the eventual object.

It all starts with making a virtual design of the object you want to create. This virtual design is made in a CAD (Computer Aided Design) file using a 3D modeling program (for the creation of a totally new object) or with the use of a 3D scanner (to copy an existing object). A 3D scanner makes a 3D digital copy of an object.

Not all 3D printers use the same technology. There are several ways to print and all those available are additive, differing mainly in the way layers are build to create the final object.

Some methods use melting or softening material to produce the layers. Selective laser sintering (SLS) and fused deposition modeling (FDM) are the most common technologies using this way of printing. Another method of printing is when we talk about curing a photo-reactive resin with a UV laser or another similar power source one layer at a time. The most common technology using this method is called stereolithography (SLA).

Since 2010, the American Society for Testing and Materials (ASTM) group “ASTM F42 – Additive Manufacturing”, developed a set of standards that classify the Additive Manufacturing processes into 7 categories according to Standard Terminology for Additive Manufacturing Technologies. These seven processes are:

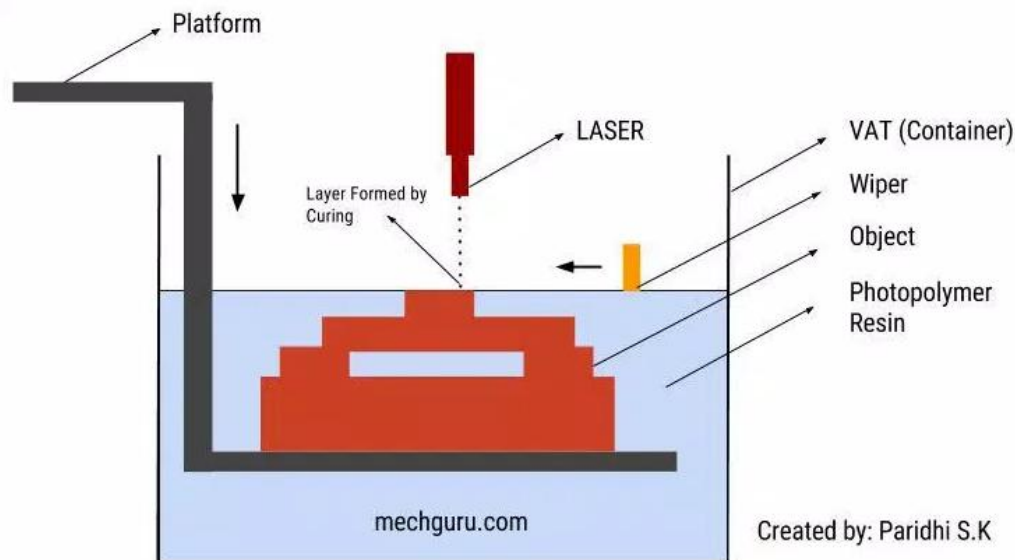
- ❖ Vat Photopolymerisation
- ❖ Material Jetting

- ❖ Binder Jetting
- ❖ Material Extrusion
- ❖ Powder Bed Fusion
- ❖ Sheet Lamination
- ❖ Directed Energy Deposition

Vat Photopolymerisation

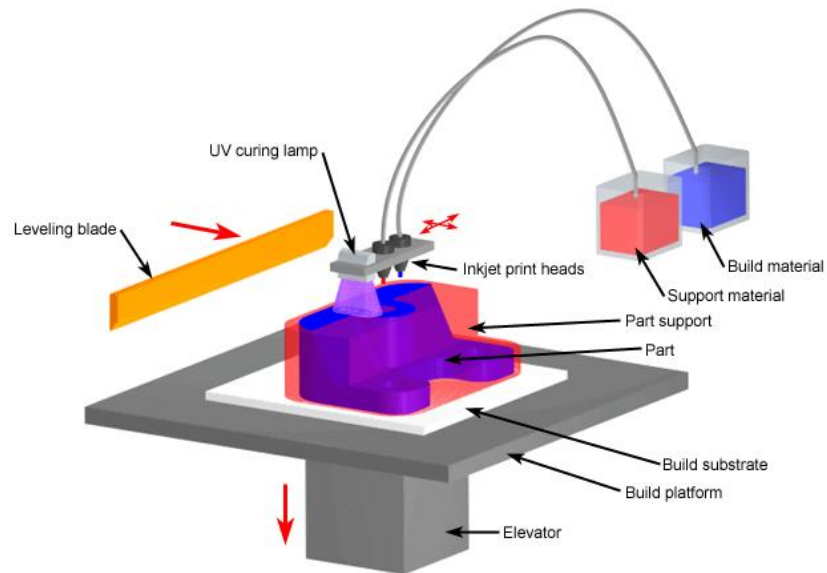
The most commonly used technology in this processes is Stereolithography (SLA). This technology employs a vat of liquid ultraviolet curable photopolymer resin and an ultraviolet laser to build the object's layers one at a time. For each layer, the laser beam traces a cross-section of the part pattern on the surface of the liquid resin. Exposure to the ultraviolet laser light cures and solidifies the pattern traced on the resin and joins it to the layer below.

After the pattern has been traced, the SLA's elevator platform descends by a distance equal to the thickness of a single layer, typically 0.05 mm to 0.15 mm (0.002" to 0.006"). Then, a resin-filled blade sweeps across the cross section of the part, re-coating it with fresh material. On this new liquid surface, the subsequent layer pattern is traced, joining the previous layer. The complete three dimensional object is formed by this project. Stereolithography requires the use of supporting structures which serve to attach the part to the elevator platform and to hold the object because it floats in the basin filled with liquid resin. These are removed manually after the object is finished.



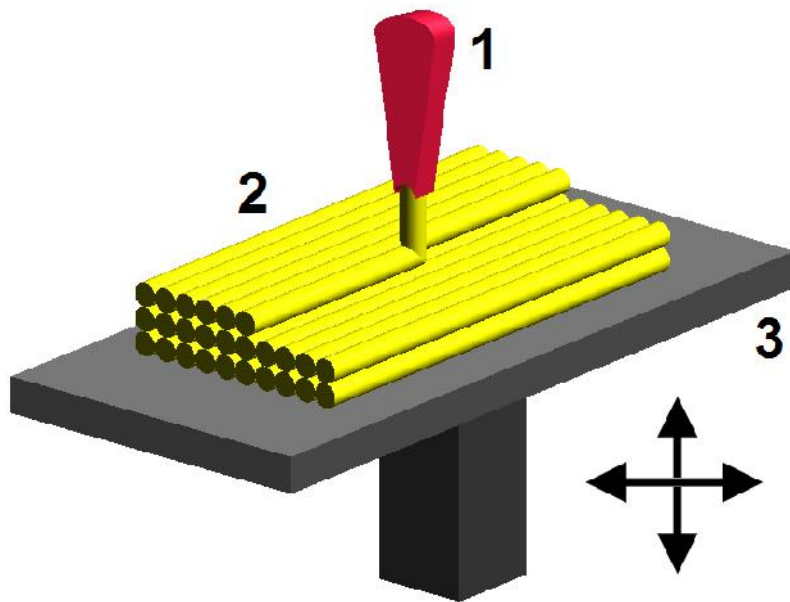
Material Jetting

In this process, material is applied in droplets through a small diameter nozzle, similar to the way a common inkjet paper printer works, but it is applied layer-by-layer to a build platform making a 3D object and then hardened by UV light.



Material Extrusion

The most commonly used technology in this process is Fused deposition modeling (FDM)

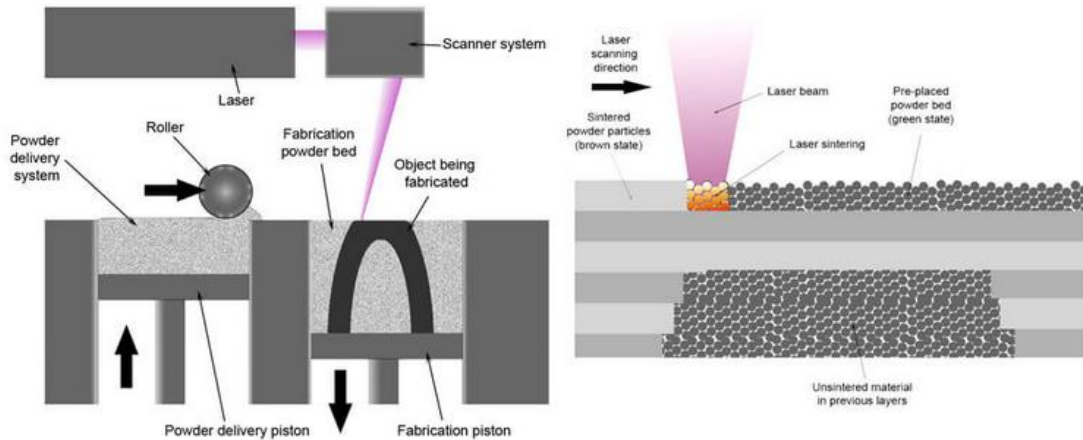


The FDM technology works using a plastic filament or metal wire which is unwound from a coil and supplying material to an extrusion nozzle which can turn the flow on and off. The nozzle is heated to melt the material and can be moved in both horizontal and vertical directions by a numerically controlled mechanism, directly controlled by a computer-aided manufacturing (CAM) software package. The object is produced by extruding melted material to form layers as the material hardens immediately after extrusion from the nozzle. This technology is most widely used with two plastic filament material types: (Acry-

lonitrile Butadiene Styrene) and (Polylactic acid) but many other materials are available ranging in properties from wood filled, conductive, flexible etc.

Powder Bed Fusion

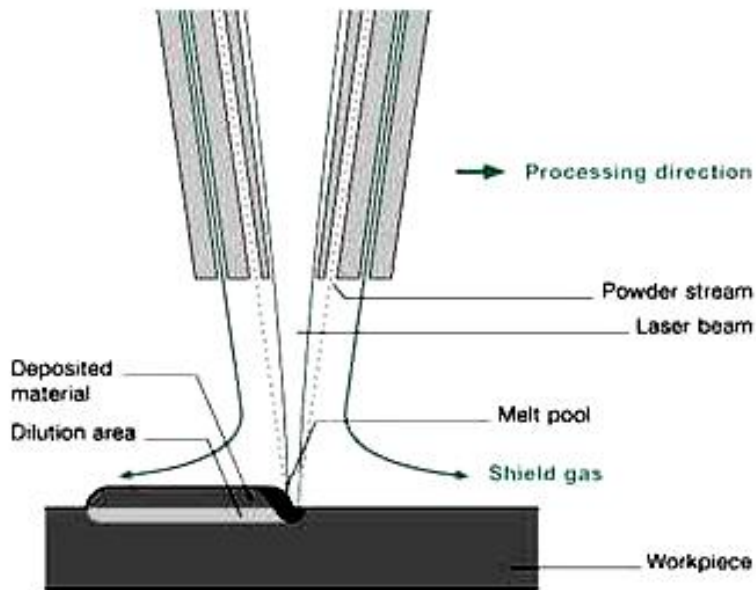
The most commonly used technology in this processes is **Selective laser sintering (SLS)**



This technology uses a high power laser to fuse small particles of plastic, metal, ceramic or glass powders into a mass that has the desired three dimensional shape. The laser selectively fuses the powdered material by scanning the cross-sections (or layers) generated by the 3D modeling program on the surface of a powder bed. After each cross-section is scanned, the powder bed is lowered by one layer thickness. Then a new layer of material is applied on top and the process is repeated until the object is completed.

All untouched powder remains as it is and becomes a support structure for the object. Therefore there is no need for any support structure which is an advantage over SLS and SLA. All unused powder can be used for the next print.

Directed Energy Deposition



This process is mostly used in the high-tech metal industry and in rapid manufacturing applications. The 3D printing apparatus is usually attached to a multi-axis robotic arm and consists of a nozzle that deposits metal powder or wire on a surface and an energy source (laser, electron beam or plasma arc) that melts it, forming a solid object.

2.5 DITHERING

The printing industry uses a halftone process in which different gray scales or shades are produced by dots of different sizes. Although the halftone process can be simulated on a computer, most computer graphics software employs a dither pattern. Dither patterns are based on square cells composed of a matrix of n (n pixels, where the cell size n determines the number of gray shades that can be produced. (Dithering can also be used to increase the number of colors available with monitors and printers).

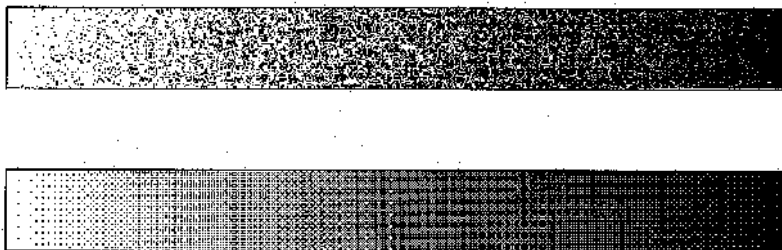


Figure 2.19

The shade of gray produced by a dither pattern depends on the number of black pixels relative to the number of white ones in a cell. Gray values between white and black can be obtained by randomly distributing black

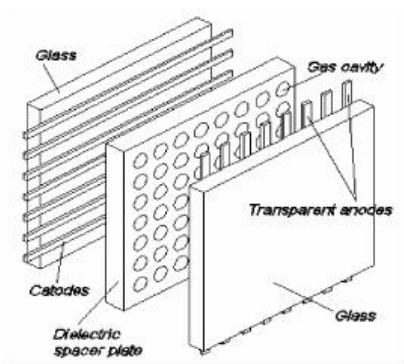
pixels on a white background, as illustrated at the top of Fig. 2.19. Although this procedure can produce appropriate gray values, the overall effect of a random dither is to produce a 'dirty' or 'foggy' effect on the scene being rendered. A much better procedure is to use an ordered dither, which produces cleaner shading, as shown at the bottom of Fig. 2.19. In an ordered dither, pixels are turned on in a prescribed pattern to produce desired gray values.



Figure 2.20

Questions:

1. Optical mouse working principle is based on:
 - a/ Using the interference between the falling and reflected light;
 - b/ Scanning the surface and using motion prediction algorithms;
 - c/ Using the light attenuation of the reflected light.
2. What is resolution?
3. What is accuracy?
4. Calculate the necessary video memory capacity for a 600 x 800 pixel screen with three primary colors and 8 bit - planes per single color. Calculate the number of possible colors for a such frame buffer.
5. In order to display images in eight colours on a graphic display the amount of the video memory in bits which must be present is:
 - a/ Equal to the number of the screen pixels;
 - b/ Four times the number of the screen pixels;
 - c/ Three times the number of the screen pixels;
 - d/ Nothing of the above
6. On the figure at the left is shown a schematic diagram of a plasma display. What causes the gas into the cavities to glow?



7. The frame buffer is used to:
 - a/ To store the information for the raster graphics display;
 - b/ To perform vector graphical calculations;

- c/ To store information needed for the CPU;
- d/ Nothing of the above.

8. What is the difference between the passive and active matrixes?
9. Draw schematic diagram and explain the working principle of the laser jet printer.
10. Draw schematic diagram and explain the working principle of the bubble jet printer.
11. What is "DITHERING"?

3. CAD/CAM SOFTWARE

3.1 INTRODUCTION

Similar to hardware, CAD/CAM software has progressed steadily since the development of interactive computer graphics in the 1960s. As in the case of computing in general, software has emerged from working out techniques, followed by developing sound algorithms, to placing more emphasis on basic principles and less on techniques that were quite inadequate when compared to the demands of actual graphics and engineering applications. As a result, the need to develop sound algorithms to support both graphics and applications was apparent.

An investigation of existing CAD/CAM software in general reveals that it has common characteristics regardless of the hardware it runs on. It is an interactive program typically written in a standard programming language FORTRAN, Pascal, or C. It is hardware-dependent and seems different to the

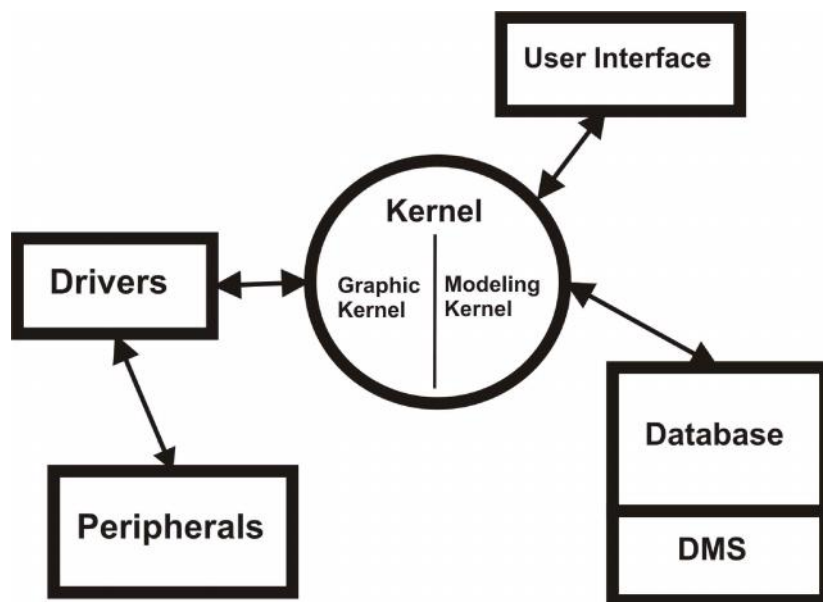


Figure 3.1

user from conventional software due to the user interface. The database structure and database management system of the software determines its quality, speed and ease of information retrieval. The overall structure of CAD system is given on Figure 3.1

The most important characteristic of CAD/CAM

software is its fully three-dimensional, associative, centralized and integrated database. Such database is always rich in information needed for both the design and manufacturing processes.

The associativity concept implies that working in multi-view environment any change in or addition to a geometric model in one of its views is automatically reflected in the existing views or any views that may be defined later. This concept is very important with the feature based parametric systems where associativity is maintained between the part models and assembly models as well as between part and assembly models and two dimensional drawings. In

this case the change in any part models is reflected in the assembly model(s) containing that part and corresponding drawings.

The integrated concept implies that a geometric model of an object can be utilized in all various phases of a product cycle.

The centralized concept implies that input information can be retrieved in various forms. For example, if the two endpoints of a line are input, the line length, projections on the coordinate axes and its direction can be output.

Users of CAD/CAM software can be classified into four groups:

- software operators;
- applications programmers;
- system programmers;
- system administrators.

The majority of users including engineers and designers fall into the **operator** category. The main concern of this group is to master using the software so that the anticipated productivity increases are achieved. A typical operator tends to specialize in one or two modules of the software. For example, a designer may master geometric modeling and finite element modeling and analysis modules while a draftsman concentrates primarily on the drafting and detailing module. Operators are usually assisted by their system manager and vendors' customer response centers and hot lines to resolve any problems and answer any questions they may encounter in their day-to-day use of the software. The post training support is very crucial because the trainee's early productivity on the CAD/CAM system to complete a design task may be less than is needed to complete the same task using the conventional ways.

Users of the software must learn:

- related **semantics** - specifies , how the software functions and what information is needed for each operation of an object. For example, a block requires three lengths and an orientation to create. Semantics are usually based on the principles and theories underlying a given field;
- **syntax** - defines the format of inputs and outputs. It is considered the grammar of the software. It specifies the rules that users must follow to achieve the desired semantics.

Applications programmers can develop new programs and link them with the software, but they are not allowed to modify the existing source code. Such a need arises when a CAD/CAM system is to be customized by adding special applications and means to its software. These programmers are also experienced operators of the software with extensive programming background. They are usually familiar with the programming module of the software. In contrast to the application programmers, system programmers have the privilege to change the source code. In essence they are the developers of the software itself.

Administration of a CAD/CAM system is similar to the administration of the software as a whole. System administrators are very important factor for

the day-to-day performance of the system and they are responsible for a number of activities such as:

- to keep track of all noticed bugs of in the program system and to report them back to the vendors. The system managers relay on the interaction with the system users (operators, application programmers) for system evaluation;
- to develop a backup procedures;
- to organize replacing the old versions of the system with new ones with minimum interruption of the work;
- to develop procedures and naming conventions for files and other named objects (layers text and dimension styles etc.) which are compulsory for the operators;
- to maintain libraries for reusing parts and assemblies, developing strict rules for library content access. For example a strict order in designing parts of an assembly could be of great help as some users can have access to the parts which a ready before starting the geometrical modeling of new parts. The order of the part creation could be of importance for assembly hierarchy as well.

3.2 DATABASE

The term ‘database’ is commonly used and may mean different things to different users. Casually, it is synonymous with the terms ‘files’ and ‘collection of files.’ Formally, a database is defined as an organized collection of graphics and nongraphics data stored on secondary storage in the computer. It could, therefore, be viewed as the art of storing or the implementation of data structure into the computer. Hence, it is a repository for stored data. From a software development point of view, a decision on the data structure has to be made first, followed by a choice of a database to implement such a structure. There may exist more than one alternative of database to implement a given data structure.

There must be at least four different types of databases in CAD/CAM environment: administrative, technological database modeling and working database.

Administrative database is mostly an archive, which contains different documents such as drawings, descriptions or technological specifications. This database replaces the conventional archive and implements functions of storing, searching and reproducing the required documentation.

Technological database consists of data used to describe technological processes. It can also contain national and international standards and specifications, material consumption norms and other technological documents.

The modeling database stores all the data necessary for the description of the models – mostly geometrical as well as some technological information concerning mostly production methods in conjunction with the desired geomet-

rical parameters and other requirements (such as surface quality). This database can also contain model of standard mechanical and other parts according to some national, international or industrial standards and norms.

The working database is intended to store some temporary intermediate data, created during the work of some program modules.

In general the objective of a database is to collect and maintain data in a central storage so that it will be available for operations and decision-making.

The advantages that accrue from having integrated control of the data, or a integrated database, is manifold:

1. Eliminate redundancy. This is important for integrated CAD/CAM functions and CIM applications. The database should be rich enough to support all various phases of product design and manufacturing. If both design and manufacturing departments, for example, have access to the same database, inconsistent and conflicting decisions are inherently eliminated, and data is shared by all applications. Thus, engineering 'assets' and experiences of a company can be captured in a database and modified for a new product designs.

2. Enforce standards. With central control of the database, both national and international standards are followed. Dimensioning and tolerancing are examples. In addition, a company can develop its own internal standards required by various departments. Standards are desirable for data interchange or migration between systems.

3. Apply security restrictions. Access to sensitive data and projects can be checked and controlled by assigning each user the proper access code (read, write, delete, copy, and/or none) to various parts of the database.

4. Balance conflicting requirements. Compromises can easily be made when designing a model of the integrated database to provide its overall best performance. If, for example, the software is designed solely for design and modeling, one would expect inadequate performance in manufacturing functions.

CAD/CAM databases must be able to store pictorial data in addition to textural and alphanumeric data typically stored in conventional databases. There are several popular database models:

RELATIONAL DATABASE - Data is stored in tables, called relations that are related to each other. The relations are stored in files which can be accessed sequentially or in a random access mode. Sequential access files are widely used. One of the disadvantages of the relational database is that it requires substantial sorting, which might result in slowing the system response to user commands.

HIERARCHICAL DATABASE - In this model, data is represented by a tree structure. The top of the tree is usually known as the 'root' and the superiority, or hierarchy, of the tree levels relative to each other descends from the root down. One of the drawbacks of the hierarchical approach is the asymmetry of the tree structure, which forces database programmers to devote time and ef-

fort to solving problems, introduced by the hierarchical approach, which are not intrinsic to the object modeling itself.

OBJECT-ORIENTED DATABASE

Unlike conventional database processing, CAD/CAM applications require object-oriented accessing and manipulation; that is units of retrieval and storage are design objects and not individual records in files. These design objects also form the basis for ensuring database integrity upon the insertion, deletion, or modification of component objects.

In object-oriented programming (OOP) systems the data structures are combined with the procedural elements (in OOP called **methods**) that manipulate them within objects. In such a case one cannot get at a piece of the object without having the whole object available. The whole program is then built around a set of such objects, which interact with each other by passing messages.

Objects have specific meanings in computer science, and from that perspective there are three features that make up an object-oriented programming system:

- inheritance of properties and methods;
- encapsulation of properties and methods;
- polymorphism across objects.

Inheritance of properties and methods means that objects that are derived from parent objects contain the same elements as the parent objects. This is seen in the airplane object in which both the propeller and jet airplanes inherit the ability to fly because they are of the airplane object class. Inheritance helps save programming time by sharing program features between similar modules. It also aids in establishing and maintaining a hierarchy of objects.

Inheritance is directly connected to the **class** concept. The class is a template from which an object is created. The class defines the object's properties and methods (discussed later in this section), that control the behavior of the object. Any object created from a class represents an instance of the same class. One example can be given with Visual Basic programming for AutoCAD. When a line is built, the user creates a unique Line object, which is a member of a class called AcadLine. The same class is used to create other lines with different characteristics, but all are based on the same class. The following code demonstrates such a principle:

```
Dim lineObj As AcadLine
Dim startPoint(0 To 2) As Double
Dim endPoint(0 To 2) As Double

Set lineObj = ThisDrawing.ModelSpace.AddLine
(startPoint, endPoint)
```

Encapsulation of properties and methods means that an object combines both data items (properties) and subroutines (methods). In addition, objects can contain references to other objects. Encapsulation means that all these related items could be referenced using a single name. It also means that data in an object is maintained by the object and that external program modules cannot access the information in an object without the entire object being available.

In the CAD environment a **property** is an attribute, or characteristic, of an object that defines the object's size, color, screen location, or some other aspect of its behavior, such as whether it is enabled or visible. To change the characteristics of an object, you change the values of its properties. To set the value of a property in a program code window, one generally uses the format: *Object.Property = Value*. Typical example can be given considering as above Visual Basic programming for AutoCAD. The following code changes the color of the line to red:

```
Dim lineObj As AcadLine  
lineObj.Color = acRed
```

Usually not all the properties can be set. Those are referred to as read-only properties.

A **method** is a function that performs an action on an object. For example, any CAD object can be moved, copied, or erased. Following the previous example we will use the *Copy* method to create a copy of the same object called *newObj*. The new object is exactly the same as the first object and is located on top of the old object.

```
Set newObj = lineObj.Copy
```

Another common task accomplished with VBA is the *creation* of entities based on user input. Most of the methods that accomplish this begin with the letters 'Add,' such as AddArc, AddCircle, AddLine, and AddText, etc.

Polymorphism, which means *many forms*, is applied to programming in the messaging system. To a programmer, it means that objects that share similar characteristics also share similar messaging formats. That is, you can send the same message to different objects to get the results unique to that object. Through polymorphism, objects use the same name to accomplish related tasks such as obtaining the layer name of an entity object or computing the intersection between two objects.

Object-oriented databases seem to be ideal for CAD/CAM applications. Hybrid database models may also be useful.

3.3 DATABASE MANAGEMENT SYSTEM (DBMS)

A DBMS is defined as the software that allows access to use and/or modify data stored in a database. The DBMS forms a layer of software between the physical database itself (i.e., stored data) and the users of this database. DBMS protects the database from user's abuse. It also shields users from having to deal with hardware-level details by interpreting their input commands and requests from the database. For example, a command such as retrieve a line could involve few lower-level steps to execute. In general, a DBMS is responsible for all database-related activities such as creating files, checking for illegal users of the database, and synchronizing user access to the database.

The requirements of a DBMS for CAD/CAM are fundamentally different from those required by commercial data processing applications.

Some large companies will have hundreds of thousands of model files, such as drawing files, in store, and in some cases they will be continually reusing these over a period of many years (consider for example the design of a large passenger aircraft or of a generating station). They therefore need systems to index and manage these large quantities of data in a sort of electronic drawing vault. These systems might be used by a designer to enquire what drawings are available on a particular project, which are on line, when were they drawn and by whom, and so on. Designers spend an enormous amount of time searching for data of all sorts: some estimates suggest that they spend as much as 30% of their time actually searching for or through information that is contained in existing drawings. A system to assist a design team in the indexing and search of its design data may be termed an **engineering data management system (EDMS)**.

3.4 USER INTERFACE

The user interface is defined in general as collection of actions that users can implement with a particular CAD/CAM system. User interface or man-machine dialogue represents the only means of communication between users and CAD/CAM software.

CAD systems may be considered as comprising a large number of functions for creating or manipulating the design model. For example a function might create a line in the database parallel to another line at a given distance (OFFSET command). To carry out this action, the user would have to:

- indicate to the system that this function is required;
- select an existing object in the model;
- enter the required offset, and indicate the side on which the new line is required.

These actions are examples of the user interface operations of program control – command entry, entity (entities) selection and data entry, which are among the activities generally used by the operator.

The general sequence of operations in utilizing the CAD program is for the user to select the function to be applied, and the particular way it is to be operated (usually the system offers several ways to accomplish a particular task – a circle can be built by specifying its center and radius or by entering three points), to enter the necessary information (i.e. via supplying the coordinates of end points for the line), and then for the program to apply this function using data provided by the user. When a particular entity or selection set of entities are to be modified, the user must apply some selection method to specify desired objects.

Traditionally, there are two ways to invoke and carry out the command:

- by command entry, in which the user enters a series of characters, essentially as a line of text, which is then interpreted by the system to determine the required action;
- by menu entry, in which the user is presented with a list of possible options and invited to select from them.

3.4.1 COMMAND-BASED SYSTEMS

They operate by reading a command and its parameters entered by the user at the command line or command window, carrying out the required actions, then waiting for the next command. The commands themselves may often comprise English-like words. In this case the language of the interface should be simple enough for user to understand. It should also be efficient and complete and should have natural grammar i.e. minimum number of easy to grasp words. This helps minimize user training process and allows the user to concentrate on the task to be solved.

The main commands are indicated by the permitted major words, and the options relating to the command by minor words. Many systems also allow commands to be abbreviated, either through the use of an alternative form (for example 'dimension' might be abbreviated by 'dm'), or by allowing the user to type only those characters of the command required for its unique identification ('dimension' in this case might be entered as 'dim'). The command is itself sometimes followed by the name of an object on which to carry out the command. The general form of command may thus be summarized as:

Command: [optional parameters] target_object

Command systems are generally more taxing to the user than menu-driven systems. They are more prone to erroneous entry, and they need comprehensive facilities to guide the user ('help systems'), which are, however, difficult

to provide effectively. For effective operation, commands should be memorized, and complex systems can therefore require substantial training and experience. On the positive side, however, in trained hands they are fast and extremely flexible, and they are also easy to customize because commands can be collected together into command files.

3.4.2 MENU-DRIVEN SYSTEMS

The menu-driven approach contrasts considerably with the command approach. The basic principle is that the user is at any time presented with a list or menu of the functions that are available to be selected. The user selects from the list, and then perhaps from further lists of available sub-functions until the function is specified fully, at which point data is entered, or items selected for the operation. In many systems the menus may be regarded as forming an inverted tree, with the main command groupings at the first level (e.g. in a CAD context, construct, modify, delete, move), and more specific functions at lower levels. Figure 3.2 shows a part of a hypothetical menu tree for a CAD system, with the selection route for the example of a line drawn parallel to another line at a distance highlighted.

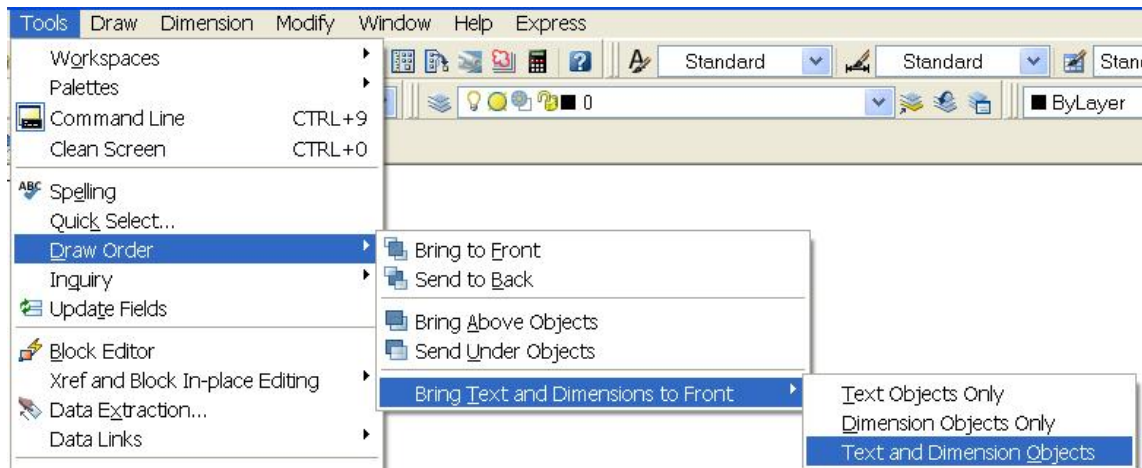


Figure 3.2

In many systems there are two groups of menus: a main or primary group for model construction and modification, with a secondary menu giving general functions that may be applied at any time, even in the midst of another operation. Such general functions include display manipulation - zoom and pan - and construction colors and linestyles.

Traditionally, menus have been displayed as a series of text items on the screen, and the appropriate text item selected in a variety of ways, for example by entering a number or letter (shortcuts) on the keyboard or by pointing to the menu item with a graphics or text cursor. Some systems allow both keystroke and cursor entry. A recent trend is for graphical symbols, known as icons, to be used for menu items, with selection usually by means of a graphics device such as a mouse combined with a

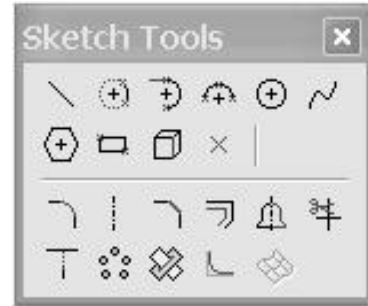


Figure 3.3

screen cursor. These icons, also referred as tools, are combined in 'toolboxes' which can be manipulated in different way (shown, hidden, arranged on the screen and so on). Figure 3.3 shows typical toolbox arranged in 'floating' position (movable location within the viewport). Normally the toolboxes are 'anchored' at the four sides around the drawing area.

Well-designed menu systems are generally easy to learn and require little training, although the very complexity of modern CAD systems often leads to several levels in the 'menu tree', and novices may find it difficult to remember where specific options are located. Menu-driven systems also require little typing skill (in particular when graphical menus or cursor selection are used), and pose little problem of erroneous selection. They are, however, rather slow to use (although this may be alleviated by allowing the user to 'type-ahead' menu selections), and are rather less flexible than command systems.

The efficiency of CAD/CAM implementation in design process depends heavily on their interactive features. While CAD/CAM system may look different due to the differences of the syntaxes used and the design and functionality that the user interface offers, they all propose similar sets of graphical assistance and manipulations.

The recognition of the basic graphical functions allows the user to think a particular way while achieving his goal. Utilizing of such functions must be made a part of the planning strategy of creating the design.

3.4.3 COMMANDS AND GRAPHICS AIDS

There are some common commands and graphics aids useful for creating geometric model on most of the existing CAD system. Most of them are described bellow.

Commands for creating 2D objects.

These objects are drawn in two dimensions on some kind of a plane. In typical drafting systems this is usually a XY plane of the world coordinate system, and the combination of two-dimensional objects form the drawing itself. In parametric systems such tools are used to produce sketches for successive add-

ing the third dimension, or to add additional graphics elements to the drawings. Contemporary CAD systems offer usually the same amount of 2D drafting tools (with different modifications). Usually the command names (in English) are equal to the graphics object it create (Line, Circle, Rectangle, Arc, Spline etc.). Very often the drawing commands support so called **Command modifiers**.

A command modifier is a word that changes the mode of input in a command. They are used in several occasions:

— **For invoking object snap modes:**

That kind of modifications as described above helps the designer to place new entities with strict accordance to other, existing entities in the drawing area. These modes are also referred as ‘object snaps’. There could be more than one active modifier working at the same time automatically.

The following AutoCAD dialog builds a horizontal line which is positioned with accordance with an existing line: the start point X coordinate is equal to the X coordinate of the midpoint of the existing line, while Y coordinate of the same start point is equal to the Y coordinate of the endpoint of the existing line. The other endpoint of the horizontal line is coincident with the endpoint of the existing line.

In this dialog the user entry in **bold**, system response in *italics*, user action between <angle brackets>:

```
Command: LINE
Specify first point: .X
of MID
of <select point 1> (need YZ): END
of <select point 2>
Specify next point or [Undo]: END
of <select point 3>
Specify next point or [Undo]:
<press the Enter key>
Command:
```

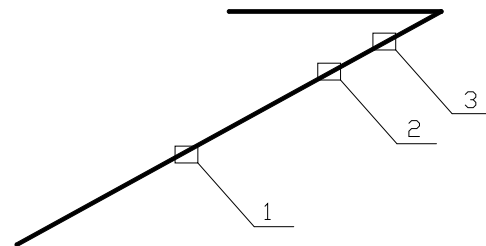


Figure 3.4

The result is shown on Fig. 3.4.

In the dialog above, after invoking the LINE command, the operator uses the **.X** command modifier at the *Specify first point:* prompt. That changes the action of the command in such a way that the X coordinate of the new point will be the same as X coordinate of some point, which already exists in the drawing (midpoint of the slanted line).

The three common geometric modifiers are the ‘endpoint’, ‘center’ (‘midpoint’) and ‘intersection’ (referred as ‘MID’, ‘END’ in the above dialog). The ‘endpoint’ modifier signifies an endpoint of an entity. Each entity has two endpoints. If the entity is a closed curve such as a circle, the two endpoints are coincident. The ‘center’ (‘midpoint’) modifier identifies the center of a closed

or midpoint of an open entity. The 'intersection' modifier indicates the intersection point of two entities.

— **For transparent commands:**

Another kind of command modifiers is associated with so called 'transparent commands'. That is, the operator is allowed to enter a new command while implementing a particular one. The current command execution is suppressed and the new command becomes active. Such modifications are widely used for visualization purposes – zooming, panning or shadowing images.

Names.

Entities (curves, surfaces, and solids) created by users and stored in a model database can be named or tagged for future reference purposes. Users can

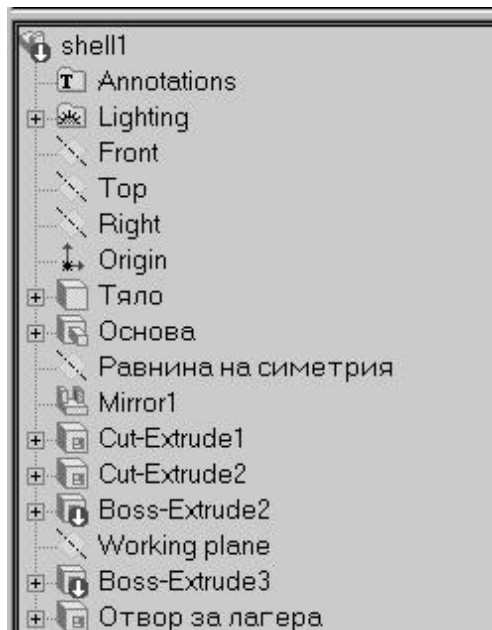


Figure 3.5

use names (tags) of entities to refer to them instead of digitizing (selecting from the drawing area of the screen) them. This is very useful when writing interactive programs with minimum user interaction. Consider, for example, referencing the center of an existing circle. The user can use the origin modifier followed by digitizing the circle or specifying its name. Using names of entities enables the user to access entities that are not displayed but reside in the model database. Naming entities usually involves either existing entities or entities to be created. For existing entities, the user requests the CAD/CAM system to name them. Each system has its own naming scheme, which is usually self-explanatory.

For example, lines are named LINXXX where XXX is a number. The number usually reflects the order in which the entities of one kind, say lines, are created. An entity can be named at the time it is created by assigning a name to it.

For example, the pseudo command:

"CREATE line P, P1 name L1"

creates a line between points *P*, and *P1* with the name *L1*. Names of entities are not usually displayed by the system unless the user requests so. To display entity names, the user must turn the status of name display on via the appropriate command. The status of name display can be turned on or off at the user control.

Naming entities is essential in feature modeling where each feature has a unique name. Usually the systems utilize standard naming convention consisting of the name of the operation which is performed to produce a feature, fol-

lowed by successive number. Thus 'Extrude 1' means a feature build by extruding some planar region. The operator can always supply meaningful names for features which is much preferable when dealing with large number of features. Figure 3.5 shows typical feature manager tree.

Layers.

Users of CAD/CAM systems may often want to group and/or separate certain types of information related to the models or parts they create on these systems. For example, a user may want to separate the dimensions and other drafting information from the geometry of a given model. This can be accomplished by using layers (some systems refer to them as classes or levels). Most CAD/CAM systems provide their users with the infinite number of layers. A layer can be thought of as a sheet of transparency. Users can mix and organize these sheets, as they desire to deal with and/or present their models interactively.

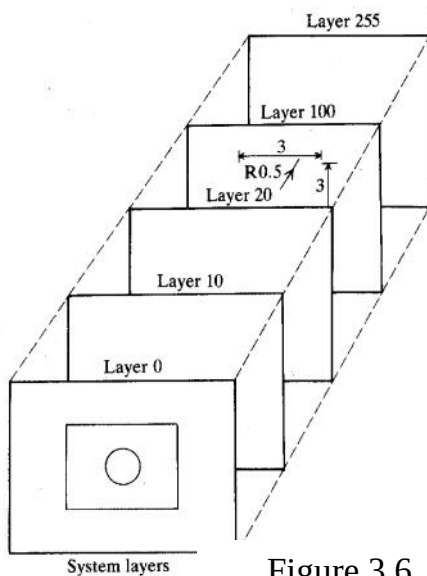
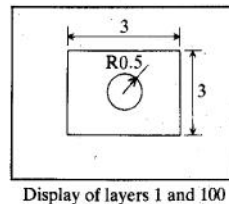


Figure 3.6



Display of layers 1 and 100

Typically, a user can assign geometric entities to layers, display on and off layers, and assign colors to layers. Figure 3.6 shows an example of using layers.

The assignment of entities to layers can be done either globally or locally. One of typical ways of assignment is as follows: If the user selects a particular layer number (make

this layer active), any geometric or graphics entities created afterwards are assigned to this layer until the user selects another layer. Users can always change (modify) their layer assignments of entities by using the proper layer commands. Layer assignment can be done locally to an entity. When the user is creating the entity, a layer modifier can be used. The resulting entity resides on the specified layer regardless of the current selectable one. While only one layer can be active at any one time, more than one layer can be displayed simultaneously. CAD/CAM systems usually provide their users with layer commands that can display one layer, any combination, or all the layers. Various colors can be assigned to layers - typically one color per layer. If no color assignments are chosen, all layers have the same color (the default color). Entities created on a given layer take the color attribute of such a layer. When more than one layer is displayed, the effect of different colors becomes obvious to users.

There are number of other object properties which could be assigned to a layer. These are linetypes (continuous, dotted, dash-dotted objects) as well as lineweight (or the thickness of the lines), whether the layer is printable or not and so on. There is always ways to change any assignment to the layer at any time.

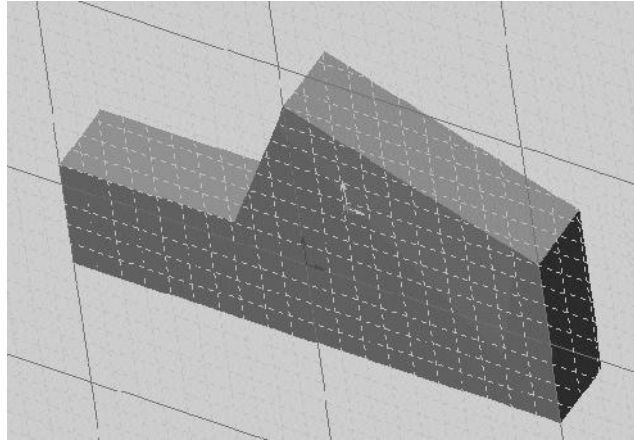


Figure 3.7

Layers are especially useful in constructing assembly and architecture drawings. Different parts (sub-assemblies) of the assembly could be placed on certain layers and manipulated or studied. The entire assembly could also be seen and operated on by displaying all layers. In an architectural drawing, for example, of a room, the shell of the room could be placed on one layer with the structural, piping, wiring, and furniture arrangements placed on other layers. In typical industrial CAD/CAM installations, standard layering schemes are usually developed by company system management and system users are required to adhere to these schemes. One example would be to use layer 1 for indexing, the next fifty layers for model geometry, the following fifty for manufacturing notes, another fifty for drafting (dimensioning and text), another fifty for analysis and technical illustrations, and the last fifty for construction aids and intermediate construction steps.

Colors.

Colors are helpful in distinguishing entities of geometric models from each other. A typical color command on a typical CAD/CAM system affects the color with which entities are displayed on the color graphics display screen. Most systems provide seven basic colors. These are red, green, blue, yellow, magenta, cyan, and gray. Other colors may be available directly or by using the RGB model to create other colors. Some systems refer to colors by their names (e.g., red for the color red) while others refer to them by numbers (1 for red). If color is assigned by layer, all entities assigned to a particular layer are displayed in the color assigned to that layer. If no color assignments are made, the default color is used for all layers.

Colors are widely used in plotting the drawings while use 'color tables'. In this case a user can always attach a color to a particular pen in the peripheral device (plotter or hardcopy printer). Such a pen could be physical (in pen plotters) or virtual (on raster devices). If a color is attached to a pen, all entities having the same color will be plotted with the same pen (with equal thickness or color) regardless of the particular layer they are assigned to.

Grids. A grid is a network of uniformly spaced points superimposed (overlaid) on the screen. Grids are useful for operations such as sketching, planning layouts, placing text at a specific location, and using freehand digitizing to indicate geometry location. When the grid is activated in a view, digitizes in that view are snapped (moved) to the nearest grid point. This mode however could be canceled or the system could provide capabilities to 'snap' to the grid nodes even when the grid is not displayed on the screen.

There are two types of grids available, rectangular and radial. Grids are displayed as series of dots (in rectangular or radial patterns), one dot for each grid point. They can be also displayed as number of intersecting lines, forming nodes at their intersection points as it is shown on Fig. 3.7.

A grid can be selected for each view and can be turned on (activated) or off (deactivated) in any view at any time. The default origin for grids is usually the origin of the view. If a grid size (spacing between grid points) is specified as being too small for the grid points to be easily distinguished, the grid is not displayed on the screen; also, the grid is not visible if the grid points are further apart than the width and/or height of the view. Changing the magnification (zooming the view in or out) of the view will cause the grid points to become visible. A rectangular grid is particularly useful in constructing equally spaced entities, enabling the user to enter exact points by using digitizing rather than explicit coordinate entry. Use of rectangular grids also greatly facilitates sketching, moving entities from point to point, measuring between points, and trimming (using grid points as trimming boundaries). The user can specify the distance between grid points, the origin of the grid, and the angle (orientation) of the grid. Different views can contain different grids of different sizes. A grid command typically activates or deactivates the rectangular grid in a specific view. Various commands can be used to control the origin, size, orientation, and/or the local area in which the grid is to be displayed. The default orientation of a rectangular grid is parallel to the X and Y axes of the active user coordinate system (UCS). Changing the orientation of the grid can be achieved by using an orientation angle as a modifier for the grid command or by orienting the UCS in the desired direction. In modeling packages grids are displayed on 'sketch planes' or $Z = 0$ of the current UCS as it is shown on Fig. 3.7.

Groups. Groups are useful when it is required to treat a selected number of entities temporarily as one (compound) entity. A group can be created by using the proper command, which usually requires the user to digitize the individual entities that are to make up the group. A group must have a name and can be manipulated by using its name or digitizing an entity that belongs to it. A command executed on a group affects all the entities in the group. More than one group can be constructed if required. Groups eliminate the need for continually constructing a window about entities that are to be manipulated several times, as in the case of animation. Consider the motion of a mechanism (a robot for example) consisting of few links, which may rotate and/or translate in space. It will be efficient to combine the entities of each link into a group, which can be

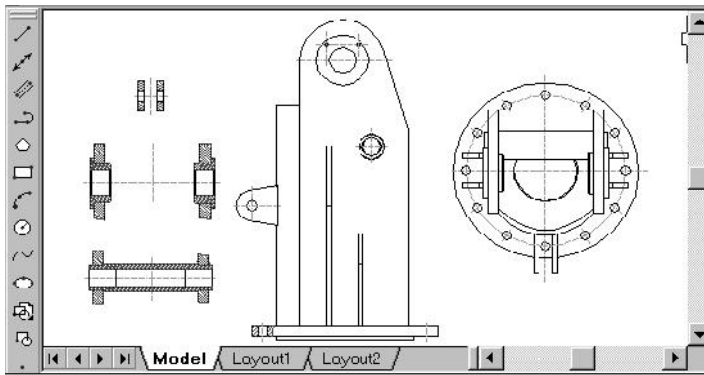


Figure 3.8a

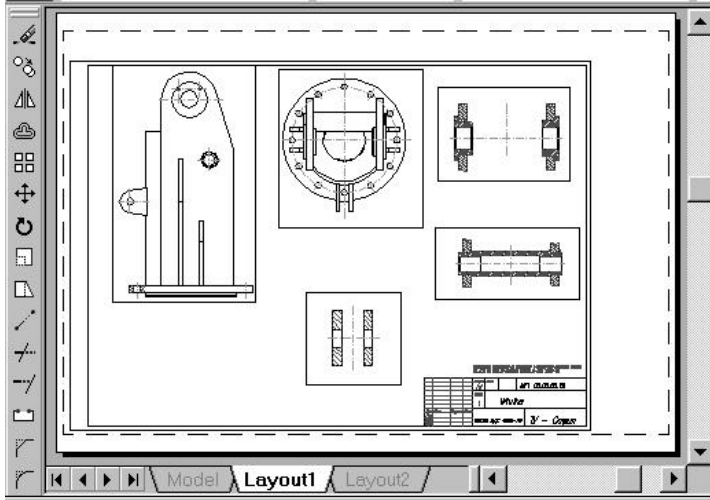


Figure 3.8b

rotated or translated as one (compound) entity from one position to another. Once a group is no longer needed, the user can delete (disassociate or degroup) it, that is, release its entities so that each can be treated individually again. A user may not be allowed to delete an entity that is a member of a group before deleting the group first.

Viewports.

Viewports are rectangular bounded areas in the drawing windows that display some portion of the model space. The operator can create as many viewports as needed. Each viewport can show different

view of the model as well as different scale or shaded representation. There is always one active viewport where the changes in the model can be done. Because of the associability concept, all changes in the active viewport are automatically reflected in other existing viewports or any viewport that may be defined later.

Viewports can also be used in preparing the document drawings, by combining and arranging different parts of the model in one drawing sheet. This is shown on Fig. 3.8 where the original drawing (Fig. 3.8 a) is arranged on a sheet

of paper, according to some drawing conventions using viewports with different scale factors.

Managing the view

Because the model could be very complex, large with small significant details, the user interface provides means for view manipulation. These manipulations consist primarily of three tools – zoom, pan and rotate.

The PAN tool does not change the location or magnification of objects on the drawing; it changes only the view.

You can change the magnification of a view by zooming in and out, which is similar to zooming in and out with a camera. ZOOM tool does not change the absolute size of objects in the drawing; it changes only the magnification of the view. When the designer works with minute parts in the drawing (3D model), he may need to zoom out frequently to see an overview of the work. Usually the ZOOM tool provides a Previous option to return quickly to the prior view. This can be several steps backwards.

The rotate view tool is a 3D navigation tool which allows viewing objects in a drawing from different angles, heights, and distances. This tool very often is called also 3D ORBIT. It moves around a target. The target of the view stays stationary while the camera location, or point of view, moves. The center of the viewport, not the center of the objects you're viewing, is the target point.



There usually are preset views (camera locations) such as views perpendicular to the coordinate planes, isometric or trimetric views which are easily set and help the designer to orient himself in three dimensional space. Such views can be set by selecting from menu or by special icon in the graphics window. Figure 3.9 shows the view cube of the Autodesk Inventor, together with the Compass icon of CATIA package.

Questions:

1. Draw a diagram showing the overall structure of CAD system.
2. What does the centralized concept of CAD/CAM software implies?
3. What does the associativity concept of CAD/CAM software implies?
4. List some type of CAD/CAM software users.
5. List some types of databases;
6. What are methods and properties of objects in the OOP?
7. Specify some properties for a graphics objects.
8. What is inheritance of properties and methods?
9. What is encapsulation of properties and methods?

10. What are the ways to invoke and carry out the command in a CAD user interface.
11. What command modifiers are used for
12. Explain the usage of names for objects in the user interface;
13. Explain the usage of layers. Properties of the layers;
14. Explain the PAN and ZOOM commands.