技 術 解 說

산업설계에의 컴퓨터 응용

정 원 량*

Ⅰ. Introduction

Industrial production has gone through several stages of progress: (i) production by human labor and hand tools, (ii) mechanized production, (iii) electromechanical production, (iv) computerization, and (v) complete automation.

The start of each stage has been ignited by a revolution in science or technology. Specifically, the British industrial revolution has started the second stage, and the invention of electricity, vacuum tubes and transistors the third.

The advent of the computer in the middle of the twentieth century has had such a far reaching impact on industrial production and our life in general that it is considered to be the second industrial revolution.

Computer technology has set the stage for the ultimate goal of industrial automation envisioned by unmanned automatic factories (15). Although different industrial sectors are in different stages of production mode, the frontiers of industries are now entering the fifth stage.

Desing is one of the important processes in industrial production-planning/management, desing, and manufacturing.

The main theme of this article is how the desing process is affected by computing technologies. We will see that, although computer systems can enchance the overall design activity, human still plays an important role as a creative agent. This article is an extended and written version of the talk given by the author at KIEE (5). Technical details of the material may be found in the author's another work (3). The readers who are interested in the history of the computer aids for industrial desing are referred to an article (Section 3 of (7)) by the author.

The article is organized as follows: In Section 2, the functions involved in the total design process are discussed in terms of five phases and the term IGCAD (interactive graphic computer aided design) is introduced. Section 3 is devoted to the discussion of the IGCAD system and turns out to be the longest section. We discuss two particular examples of IGCAD applications and the benefits of IGCAD methodology in Section 4.

Ⅱ. Total Desing Process

Design is a process of systematically determining the form (shape), structure, material composition, and specification of a product to meet a set of design criteria.

The word "systematically" implies that there may be several stages to go through from conceptualization of an idea to production of a full documentation.

Indeed, we can identify five phases of this total design process: (i) preliminary design, (ii) detailed design, (iii) design analysis, (iv) design verification, and (v) design documentation.

In the following paragraphs we will review

* 正會員 : 韓國科學技術硏究所 電算開發센터先任硏究院(工博)

the detailed activities involved and functions to be supported for each of these phases. and then we will initiate the discussion of the main theme, i.e., use of computer system as a major design tool, by introducing the term "interactive graphic computer aided design" (IGACAD)..

Preliminary Design:

When a need is recognized for an industrial product, a designer has to translate the needs into functions (to be performed by a product). He will them determine various features to satisfy these function. These features consititute a basis for the criteria against which the acceptance of a designed product is tested in a later stage. The next step is to find an "idea" of the design. This is the very step that requires creativity and innovation. Two most common practices that a designer follows to help his idealization are literature search and brainstorming. The former is necessary to study other people,s ideas, to prevent reinventing the wheel, or to improve old designs.

The latter may be applied to single person or a group. Either case, a list of ideas, some crazy and some very inventive and useful, is obtained from this process. These design ideas are now screened to select the most desirable one. This is the step of feasibility analysis. Various factors or constraints are considered: technological, human, economic, social, safety, environmental. The analysis performed here is not usually quantitative like the one for the phase-three. Most of all, profit is the most critical factor for industrial products. Now that a design of a product is conceptualized, the designer moves on to the next stage.

Detailed Design:

All the specifics of the design idea conceived in the first stages (likely in the form of free hand sketch) are fixed in this phase. Except for monolithic products which consist of single piece, components for a product have to be selected, and their geometry, material composition, and color have to be determined according to the design criteria. A designer defines the structure: interconnection or topology, hierarchy, input/output relations, and interface between components. Specification of all parameters-dimensions, physical and logical-is also done.

Many drawings are produced in this phase. They may not be in such a neat form to be used as documentation but should contain all the information regarding a product or for analysis purposes.

Design Analysis:

Once a product specification is defined, it is subjected to a rigorous analysis. Analysis may be performed by a designer himself or by a group of specialists who possess the necessary expertise. What this means is that design analysis is bound to be application dependent. However, there are a number of types of analysis which are common to many different products and applications. To name a few, they are functional, cost, performance, reliability, stability, and durability analyses. There are also a set of techniques or methods that are often employed for quantitative analysis: medeling and macromodeling (17), simulation, optimization (linear or dynamic programming, nonlinear optimization), structural analysis techniques such as finite-element analysis, and prototype analysis (actual or scaleddown size models) (11).

Design Verification:

Design analysis phase produces a large amount of dataexperimental or computational. Interpretation and evaluation of these analysis results are the main tasks of design verification phase. Every aspect of a product and its environment becomes the target of scrutiny.

This includes even the styling or fashion. Compatibility with design criteria is checked and a judgement is made bases on the analysis results, experience, knowledge, and available information in general. The outcome of this process decides the approval or disapproval of the current design. When the current design is not accepted, the design process repeats itself by going back to phase-one or-two to effect design changes.

If the design is finalized, a designer or engineer goes ahead and generates manufacturing data. Manufacturing data are produced from the products definition and include machine tool instructions, work sequence, assembly and packaging instructions. It is sometimes required to generate test or diagnostic data.

Design Documentation:

By the end of design verification stage a designer tends to feel quite exhausted and has used much of his creative talent. Even so, the last remaining step should not be forgotten or neglected. Design documentation is considered to be the least creative of design functions but not the least important. It is often the most expensive.

Whether it is done by a designer or left to a professional technician care should be exercised in producing good documentation. The purposes of documentation are manifold: (i) For reportingoral or written, (ii) For publication, (iii) As a design record-in design library or archive, (iv) As production information (as manufacturing inferface), (v) For registration-patent, copyright, government, etc, (vi) As maintenance information (inventory, service), (vii) As operational information-such as operator's or training manual, and (viii) As consumer niformation. Design documents contain many pieces of drawings such as assembly, layout, and multiview orthogonal drawings. They also contain functional specification, component list, parameter values, part numbers, and price data. These indicate the needs for text editing capabilities as well as drawing functions and complex data representations.

Designers have used a variety of tools and aids which have not changed much for a long period of time. They are writing and drawing tools (pen, paper, rulers, scales, etc), calculating tools (tables, slide rule, calculator), knowledge aids (books, algorithms), and personal skills (experience, aesthetic faculty, imagination, etc). Without these tools, design becomes inefficient, error-prone, and almost unmanageable, particularly when the product is a complex system.

With the introduction of computers into design activities as a powerful multipurpose tool, design has evolved into a sophisticated man-machine processe. This is due to the division of labor and cooperation between man and machine. Various design functions are divided according to their best capabilities. Humans are good at decision making, innovation, and heuristic procedures. On the other hand, the computer system possesses extremely high speed of calculation, vast storage capacity, and automatic analysis and documentation capabilities. Therefore, most of design analysis and documentation functions can be computerized. In addition, many activities of other phases of design can be performed by a computer or greatly enhanced by computer aids.

The methodology of utilizing computer aids for design in the above fashion is given a name, "computer aided design (CAD)" or "design automation". These terms place emphasis on differing aspects of computer aids and design. But one thing is sure.

That is, complete automation of design is infeasible or against the basic objectives of design.

As such, a designer and computer system become a duo by a close partnership whose separation leads to an inevitable breakdown of design process itself.

For this reason, we will use the term "interactive graphic computer aided design (IGCAD)", bringing forward the significance of information exchange (interaction) between man and computer and aspects of graphical man-machine interface (3).

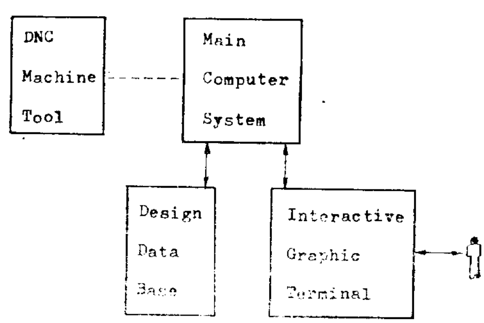


Fig. 1. Structure of an IGCAD System

Ⅲ. Igcad System

Our discussion of IGCAD methodology will be centered around an IGCAD system. After a concise description of components and structure of an IGCAD system, we will discuss the functional and systems requirements and problematic issues in eight problem areas: (i) interactive computer graphics, (ii) man-machine dialogues, (iii) programming systems, (iv) data base systems, (v) distributed processing, (vi) analysis packages, (vii) geometric modeling, and (viii) manufacturing interface.

Major subsystem components of an IGCAD system are the main computer system (CPU, main memory, and standard peripherals), design data base, interactive graphic terminal, and a human designer. The system configuration is shown in Fig. 1. When a design is entered via the graphic terminal, it will be stored on the design data base which will be altered and manipulated for analysis and display as the design process goes on. Everything that is input or output through the terminal, whether it be the geometry of product, analysis results, or documentation, should exist in the form of graphical data.

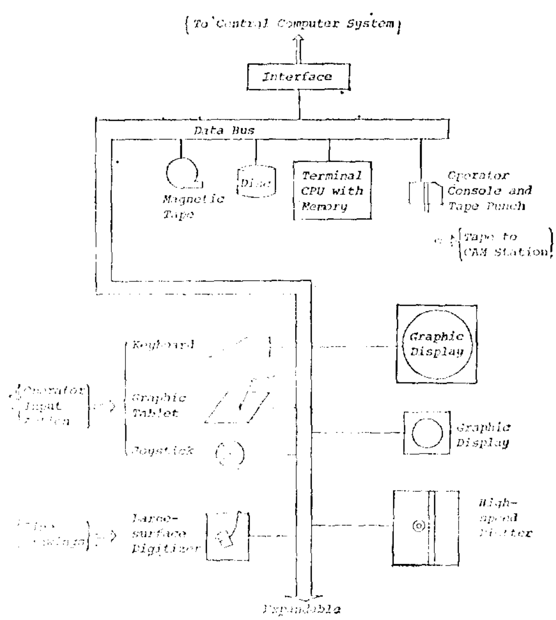


Fig. 2. Interactive graphic terminal configuration

In other words, the graphics terminal subsystem functions as a front-end or viewing window with respect to the rest of the computer system. An application proigram becomes the center of all activities and processes for design and invokes graphics, data base management, analysis, documentation, and NC programming modules.

These latter modules comprise the systems software for an IGCAD system that is made available to applications programmers.

Interactive Computer Graphics:

Interactive computer graphics is one of the most important areas for IGCAD methology as already pointed out in the above. It is concerned with organizing hairdware/software systems for syntheszing images and for accepting pictorial ipnut. IGCAD is one of many applications of interactive graphics (others include image processing, remote sensing, CAI, decision support systems, command/control systems). But, it is the application that requires the most sophisticated and high-performance configuration for interactive graphics systems. A typical hardware components forr an interactive graphic terminal are shown in Fig. 2. They are display processing unit (DPU) and local processor, refresh memory, display generators, one, or more video monitors, keyboard, cursor device (like joystick), graphical tablet, large surface digitizer (s), printer and plotter, computer microfilm output (COM) recorder (3,7,10,18).

Among the currently available display technologies, direct view storage tube (DVST) displays outnumber any other primarily due to its low lost and high resolution (Its low cost is attributed to the absence of a large amount of refresh memory).

As hardware cost drops including memory cost, refresh-type or raste-scan displays offer very attractive alternatives because they are capable of dynamic displays and thus provide more realism and interaction versatility. So it is not an overstatement that raster-scan displays, for which color comes almost free, will be taking place of DVST displays in the coming years at an accelerating rate. Hard evidences for this prediction exist: price of color raster-scan displays becomes competitive with DVST or random-scan refresh types (say, around $10,000, although screen size is somewhat smaller at that price yet); the resolution is getting higher which is a prerequisite for IGCAD applications (500×500 is the de facto standard for now, but 1000×1000 is already available). Also an important parameter is the screen size. Users prefer large-screen displays for good reasons. Trend is toward the large-screen display and 25"-diagonal screen has appeared (with resolution of 4000×4000, DVST, Tektronix). 21"-diagonal color rasterscan displays are available and 25"-version should be out in the market about this time. For the details of performance parameters and price ranges for graphics hardware readers are referred to the reference by auther (Chapter 1, (3)).

One more remark about hardware: Although conventional plotters are continued to be used for hard copy output (high-speed, high-resolution, multi-color, large-surface plotters are preferred in IGCAD, while electrostatic printer/plotters are becoming popular and useful in certain applications*), the potential of COM recorders is just being exploited. Of course, many of us think of and try for the paper-less environment (i,e., everything will be stored on the computer system), there still is and will be needs for some form of hardcopy output. As the amound of information grows rapidly, the problem of having to handle the paper output becomes serious. The lure of COM output as a replacement is hard to dispell. They are very cheap and extremely compact (a microfiche containing a 80-page report costs only a few cents). Furthermore, they can be produced with a high-resolution (many orders higher than that of display or plotters) and with colors. Also different from paper plot is that computer accessible storage device is available. One major drawback is the relatively high price of the COM recorder. But that price can be easily amortized if the device is shared among many users over a long period of time. Problems of interactive graphic software will be discussed in programming systems.

Man-machine Dialogues:

There are two issues pertinent to graphical man-machine dialgues: (i) ergonomics, and (ii) command language (see Section 2.4 of (3)).

Erogonomics concerns both hardware and software aspects of graphical interaction. Proper spatial arrangement of hardware components for a design station is crucial inincreasing the work efficiency and ultimately productivity of a designer. Graphics hardware characteristics reflectance, contrast, light output, and size of the screen, button lighting, back-lighting for digitizer-should be considered for effective visual perception and comfortable mortory actions. Characteristies desired from software aspects include the following (13) : (a) right amount of information contents in a single frame, (b) proper frame organization-overlays, multiple windows, (c) fast reponse time for continuity of thought, (d) responsive feedback for input actions, (e) flexibility of action choices-many different ways of input data specification, (f) adaptiveness appropriate responses according to user's maturity, (g) forgiveability-error correction and recovery features, (h) sufficient user information helps, diagnostics, training aids, (i) hierarchically structured input and output-naming and macro facilities, menu hierarchy (13).

Human-factors problems for command-language design were already discussed in the above. The remaining problems are the modeling (theory), programming language syntax, efficient processor implementation, and antomatic production of command languages. Automata model has been extensibly used and for good reasons. It provides us with valuable insights into the design of syntax and implementation. Other problems have received limited amount of attention thus far. Automatic command language generation system will be a convenient tool for applications programmers if it is available and incorperates all the ergonomic factors mentioned in the above.

Programming Systems:

The main issue here is the provision of a 'good' programming system: 'good' in the sense that the language is high-level and user-oriented, that there are many software modules for ready-use to eliminate the problem of re-programming, (these are the concerns to applications programmers), and that the system is implemented efficiently (since efficiency means fast reaponse for man-machine interaction and more importantly lowcost of operation and eventually of design and product).

Other than general software-engineering-type problems (structure controls, data abstraction, etc) a prominent problem in language design unique to IGCAD is the polymorphic nature of IGCAD. That is, the applications programming language should cover several problem domains graphics, geometric modeling, documentation, data base, NC part programming, as well as the specific application area delineated by a product (6). In order to have a proper level of abstraction (above that of general-purpose language like FORTRAN, ALGOL, APL, or PL/1) and to bring down the complexity of the problem to a manageable proportion, what we need to do is to obtain well defined models for these problem domains from which the language syntax can be readily derived. (These models should be application-and device independent as much as possible).

The collection of software modules useful for IGCAD may contain the following algorithms: curve and surface generation, clipping, hidden line or hidden-surface removal, shading, assembly and layout drawings (as used in drafting and documentation), geometric computations (intersections, volumes, arear, etc), cutter path generation, curve and surface fitting, contouring, 3-D reconstruction from 2-D data, mesh generation, symbolic differentiation/integration.

Efficient implementation of the programming system is a pragmatic problem to be coordinated with overall IGCAD system development. There fore, it is desirable to take a unified topdown approach to the system development rather than piecemeal methods. It seems to the author that the key to the solution lies in the implementation of efficient data organization throughout the system in both spatial and temporal domains. Then we may proceed with development of efficient algorithms in a local fashion (These algorithms include the above ones). Language optim ization could be one of many problems (4).

Data-base Systems:

Design data base contans a variety of information regarding a product and design plus design aids like knowledge. It is organized in such a way that it is constantly updated and presents up-to-date design data to a designer any time it is accessed through a graphics interface.

There are many commercially available general purpose data base management systems such as IMS (IBM), TOTAL (CDC), MARK IV (Informatics), SYSTEM-2000 (MRI). They are based on one of approaches-relational, hierarchical, and network. It is not impossible to use such a general-purpose system for constructing a design data base. However, it is not recommended to do so due to many unique characteristics of design data and procedures. They also suffer from an inefficiency problem which is critical for cost reduction and improvement of interaction speed.

Central to a design data base is the geometric description of a product together with application related specification data. All other data are derived from this geometric model. In order to maintain the consistency of a highly interrelated data representation, it is preferable to keep a single copy of 'values' within the geometric model. Then a change in the design specification is done through this single representation and the dependency of other data on this model guarantees the consistency. As a realistic example, when a designer modifies a design parameter (say, size) in a design session at a graphic terminal-he needs not worry about changing drafting data or recompiling input data for an analysis module. This automatic maintenance of consistency is effected because the geometric model functions as the "basis" for these data representations. Another problem in efficient implementation of design data is to optimize data representations which are repeatedly traced and used by many operations. Graphics data is only one example. This optimization should be tried on a global basis not just on a local scale. Therefore indiscriminatory use of very general structures like plex should be carefully examined to weigh the advantages of generality against the perils of inefficiency.

Data base management systems aimed at IGCAD users are not yet readily available as general purpose systems mentioned in the above. Most of current systems are proprietary and embedded with in turnkey stand-alone or large-scale CAD/CAM system (8,19).

Distributed Processing:

Distributed processing is based on the simulataneous (concurrent) processing of jobs by many processing facilities. Distributed computing system forms a network whose nodes are the processing units.

Many IGCAD applications are dependent on this distributed system organization due to their large processing requirement (caused by large design data base and complicated analysis needs) and trends toward intelligent satellite terminal configurations. The latter is greatly influenced by two factors: one is the declining hardware cost and avai lability of low cost micro-or mini-computer-based systems which can satisfy much of computing requirement for IGCAD in a standalone mode; the other is the intrinsic characteristics of IGCAD activities which call for a more or less dedicated computing power to handle interactive functions needing a large number of short bursts of attention processing while there arise occasional needs for large number crunching or data base processing.

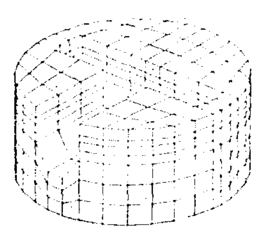
Distributed processing configuration in IGCAD takes a variety of forms depending on the power of local satellite system. They range from simple time-sharing terminals to full computer systems (i.e., network computers). Therefore, any IGCAD User should choose a system configuration that best serve his needs at a minimum cost. While the cost of communication decreases it always seems to lag behind that of hardware. This, together with other benefits of having a loccl comptuing system of their own, will further encourage the use of intelligent satellite design stations.

We will point out only a few problems related to distributed processing in IGCAD. There are of course many problems that are network related that of data communication and protocol. Another problem is addressed to data base management: maintaining the integrity and consistency of the large volume of data for multiple users under the presence of concurrent processing, servicing a large number of users (i.e., different terminals) in different application areas. The second problem poses the question of supporting a distributed data base organization for varying degrees of locality, i.e., ranging from no local data base to a case with almost complete local design data base.

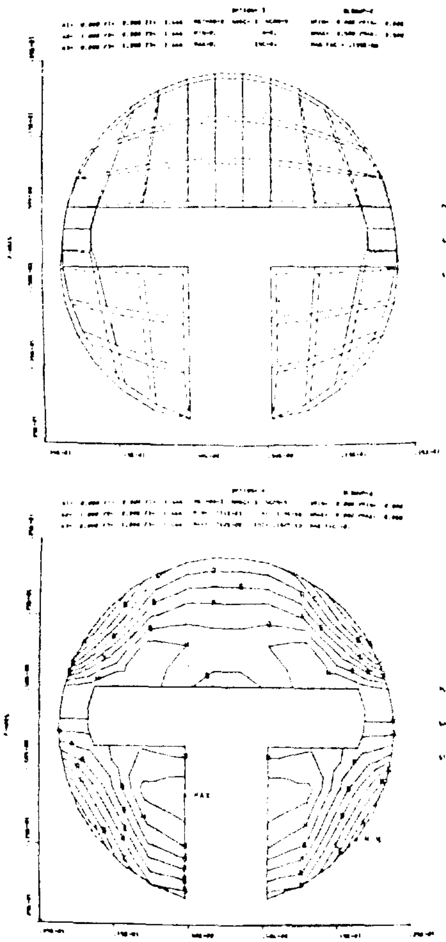
Analysis Packages:

The types of analysis techniques were introduced in the previous section. Specific analysis packages used for particular applications will be mentioned later in the next section.

Algorithms and packages for design analysis have been developed by many people from multi disciplinary areas. Many of them concern more than one design areas. One example is the network analysis package which is used for electronic circuits, methanical or civil engineering networks, or even the social or economic network models. Once an anlysis package is developed (and the software is certified) by a computer scientist, numerical analyst, engineer, or social scientist, it is going to be used by nonspecialists. Therefore, the most important problem to be resolved here is that of providing proper input and output interfaces in such a way that the user (i.e., designer) should spend the least amount of efforts in prepa-



(a) Display of finite-element mesh



(b) Display of finite-element analysis results

Eig 3

ring input data and interpreting analysis results.

A good example for the above point is found in structural analysis. Much work has been done to provide user-oriented graphical interfaces for finite-element analysis packages in the from of graphical pre-and postprocessors (the former for mesh generation and the latter for results display).. ICON (Interactive Creation of NASTRAN) (1) and SAP6 are the prominent examples. Fig 3 shows an isometric drawing of a 3-D object with meshes for structural analysis and the display of analysis results.

Geometric Modeling:

Geometry, without and exaggeration, is the most important single attribute of a product in IGCAD. Not only the designer's activities from conceptualization to manufacturing data generation but also the data and procedure representation within an IGCAD system depend critically upon the geometric information of a design. Stimulated by the fascinating subject of geometry many intelligent people have contributed a great deal to the development of theory and practical systems for geometric modeling (2). They now claim a well-defined area entitled "computational geometry".

There are three recognized approaches to geometric modeling for IGCAD: they are based on curves, free-from surfaces, and solid volumes (2, 3, 11). These geometric models form a hierarchy in terms of geometric complexity and computer representation. What this says is that the above classification is based on the geometric unit used by a designer as a building block in construction of geometric structures. Mathematical background for geometric modeling is provided by analytical geometry (both plane and solid) and approximation theory.

Manufacturing Interface:

Integrated IGCAD systems have evolved from the achivements in several related areas-NC, APT, geometric modeling, and interactive graphics, whose origins date back to early sixties (7).

It is only very recent that integrated IGCAD systems are developed by industries and laboratories. In this process, many problems in integration were identified and solved. one of the them was the recognition of the fact that APT or APT-like part programming languages are no longer suitable for proper manufacturing interface. The basic reason for this is that these languages are not powerful enough to describe a variety of geometric classes or convenient for designers from the human-factors viewpoints. Thus, there is now a strong tendency to move away from APT as a manufacturing interface standard, especially on the side of academic community. GNC (Graphical Numerical Control) and POLYSURF systems at CAD Center of Cambridge University are viewed as typical efforts in that direction (14). However, IGCAD systems based on APT are widespread and well-rooted in industrial sectors.

Extensions of old APT intended to support a wide class of geometries have been tried by the APT Long Range Program Committee at IIT. It will be interesting and worth while to watch where these competitions and current research activities will lead to.

IV. APPLICATIONS AND ECONOMICS OF IGCAD

We will discuss two major applications of IGCAD methodogy: (i) electrical/electronic design and (ii) geometric form design. For each area of applicaton the discussion will focus on the class of geometries and corresponding display requirements and analysis problems involved.

Electrical/electronic Design:

Typical products that are designed by IGCAD systems are discrete component circuits, LSI, IC, VLSI chips, PCB layouts, digital systems including microprocessors, electric power systems, generators, motors, lighting systems, and applionces. We will concentrate on the electrical parts of these products although they contain nonelectrical parts too.

Design of these products consists of two distinct stages: i.e., logical design followed by physical design. Despite their predecessor-successor relation-ship, there are rather few integrated IGCAD systems supporting both. One of many reasons explaining this is the fact that they require quite distinct analysis techniques as will be seen below.

The class of geometries required of logical design is that of 2-D line drawings encompassing relatively restricted shapes. This class is designated by the component geometry for logical schematic diagrams. As Fig 4—1 shows straight line and conic segments are sufficient to represent most components and circuits. The geometric class relevant to physicl design is larger than the above (see Fig 4—2). It is necessary to include 2-D polygons or other areal patterns for proper description of PCB artwork master and IC/LSI lithographic mask patterns. Piecewise cubic

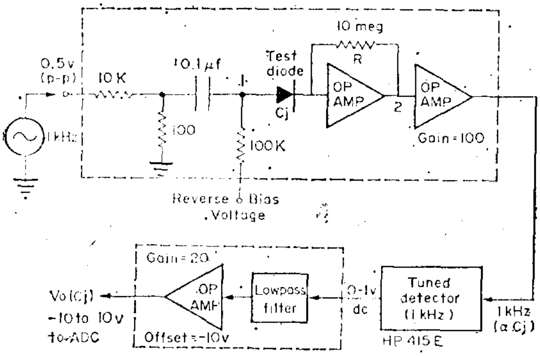
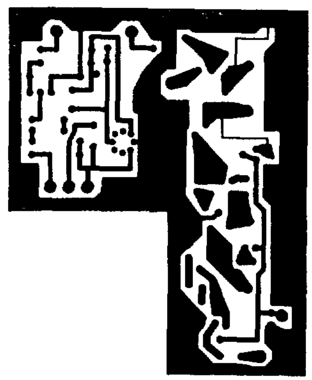
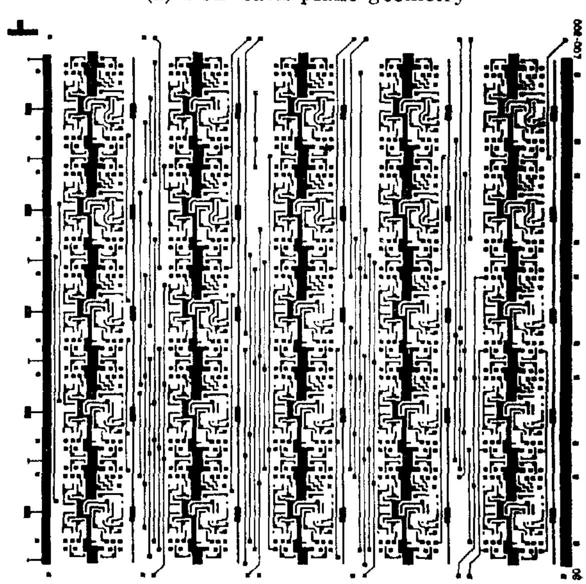


Fig. 4-1. Schematic diagram for logical design of a circuit



(a) PCB back plane geometry



(b) Art work mask foran IC

Fig 4-2

polynomials will be more than adequate for description of boundary curves of these patterns. However, there are othr geometric forms that are needed by an IGCAD system for logical or physical design. They are necessary to represent the electrical properites of components and circuits and take the form of 2-D graphs. They may occur in design specification or display of analysis results.

Analysis of logical and physical designs call for different appr and techniqus. the former requires (non) linear circuit analysis, network analysis and modeling/simulation techniques. Examples of analysis packages are ASTAP (IBM), CIRCUS (Boeing), ECAP2 (IBM), NET2 (G.E.), SNAP (Purdue Univ.). Analysis functions required by physical layout design include placement (of parts and pins), routing, test and diagnostics generation. Almost all of packages for these functions require human interactions and thus are not fully automatic yet. Programs like XYMAD (Bell Labs.), AEWRAP, CADAT, WIRE, NOMAD (Bell Labs.) are examples. The final output from an IGCAD system for physical design consists of PCB artwork master generation instructions (say, for a photoplotter), NC drill instructions to produce holes on a PCB, NC component insertion instructions, and IC/LSI lithographic mask production instructions.

We will discuss briefly the display requirements of IGCAD systems for electrical design: High complexity of modern electronic circuits demands, first of all, that sufficient resolution be available (at least 1000×10000 pixels, and more is desirable for high-density microelectronics Use of color will be valuable because it makes design). it possible to color-code components, to dispplay a family of electrically similar components in an identical color, and to use different colors for different diagrams in simultaneous or overlay displays (e.g., overlay of logical sche matic and physical layout diagrams). Most systems these days depend on nonrefresh DVST displays. As hardware cost comes down further, refresh capability (with color) will be introduced since redrawing of complex circuits is inhibitly show and since windowing and scrolling of large PCB's or IC's will add to flexibility and effectiveness of graphical interaction.

Geometric form Design:

Geometric forms can be classified according to their dimensionality. 2-D forms are found in flat patterns (textile, shoe-making, furniture industries)or sheet metal parts. Exampless of $2^1/_2$-D objects are numerous in mechanical parts and machinery industries. 3-D forms are extensively used in aerospace, shipbuilding, and automobile industries for description of sculptured surfaces comprising the exterior of vehicles and ships. Designs of geometric forms by means of IGCAD systems are commonplace in the above industries and also in construction/plant engineering. Civil and architectural construction-highways, cities, bridges, buildings, damsrequires $2^1/_2$-D or 3—D geometries. Piping network design needed in plant engineering depends heavily on 3-D geemetry and IGCAD systems (e.g., ADL PIPE*).

Methematical and computational representations of these forms are parametric curves/surfaces, blended surfaces, ruled or lofted surfaces, Bezier curves/surfaces, and Bspline surfaces.

Analysis techniques used for geometric from design include finite-element analysis (packages like NASTRAN, SAP*, STRUDL), nesting algorithms (or cutting stock algorithms), impact and load analysis, aerodynamic analysis, thermal analysis among others.

In contrast to electrical/electronic design displaying the shape information itself with high realism is of utmost importance in geometric design. For instance, displays of curved surfaces with continuous shading are not so uncommon. Color is also one of critical attributes for the types of products that we are concerned with here. Results displays are much more elaborate than those for electrical design. Contours or 3-D surfaces appear quite often in this context. Color can be a valuable asset in results display such as overlays of original and distorted structures using different colors.

As an example of use of color, display of a complex piping network for a nuclear power plant or ship can be made more readable by coloring the pipe subnetworks carrying certain kinds of chemical or fluid differently from others. Displays are also useful for interactive generation and editing of NC programs. Even the operations of NC machine tools and assembling machines can be graphically simulated and verified (14).

Refresh displays are essential for this purpose and color can add more to the power of interactive graphics.

Economics of IGCAD:

There are many yardsticks used for discussing the cost effectiveness of IGCAD. Two most common ones are the total design cost and terminal usage cost/hour. In the first case(9), different defintions of the total design cost are used depending on the desiggn environment: Most often the costo of design analysis and documentation are accounted for a fixed design, and both manpower and computing expenses are included. The costs of design with and without an IGCAD system are then compared in terms of their absolute amount. When the second measure is used, the comparison is made by the amound of cost per hour. For an IGCAD system it is also useful for comparing various system configurations-small vs large systems, time-shared vs. standalone systems, the number of terminals attached. As an example, the number may go down to under $5/hr, for an IGCAD system with a large CPU (scale of IBM 3032) and tens of time-shared graphical terminals. (It should be noted that the cost/hour dropps as the number of terminals increases and that the large number of terminals can only be supported by a large scale computer.).

There are other figures of merit that are rather difficult to compare. They are the cost of design changes (within a single design or over a long period of time during the life cycle of a product) and the benefits or losses incurred by the reduction of lead-time or overall design time. We note that a single design modification will necessitate the redoing of entire analysis and/or documentation phases, which can be efficiently handled only by on IGCAD system. Turning out a product in a competitive market on a timely basis is an advantage that may decide the success or failure of the product not just the cost savings.

IGCAD methodology offers many other benefits:

(i) improved product quality due to the use of better design tools, thorough testing, and high precision made possible by the computer technology, (ii) reduced design and product cost, (iii) improved efficiency from increased productivity, shortened design time, and elimination of human deficiencies such as errors, imprecision, sickness, training, and labor dispute, and (iv) standing at the cutting edge of technology because of the fact that incorperation of new design technology into an IGCAD system is easier than any other means.

REFERENCES

1. Armit, A. P. and Lemke, H.U. ICON: The interactive creation of NASTRAN data. A system description. CAD 7-3 (July 1975), 145-150.
2. Barnhill, R.E. and Riesenfeld, R.F. (eds.) Computer Aided Geometric Design. Academic Press, New york, 1974.
3. Chung, W.L. Interactive Graphic Computer Aided Design. Notes for a professional seminar with the same title at KIST (Feb. 22-24, 1979), Jan. 1979. KIST, 242 pp.
4. Chung, W.L. Graphic languages: Semantics and code optimization. Proc. of Int'l Conf. on Interactive Techniques in CAD, Sept. 1978, Bologna, Italy, 255-264.
5. Chung, W.L. Use of computers in industrial design. Invited Lecture for Korea Institute of Electrical Engineers, April 7, 1979, in commemoration of science month.
6. Chung, W.L. Geometric modeling: The interface between CAD and CAM. Spring Conf. of Korea Information Science Society, April 21, 1979, to appear in J. KISS.

〈p. 48에 계속〉