

# The Interactive-Graphic Man-Computer Dialogue in Computer-Aided Circuit Design

ROBERT SPENCE, SENIOR MEMBER, IEEE, AND MARK APPERLEY

**Abstract**—The crucial role played by the man-computer dialogue in computer-aided circuit design, and particularly within an interactive graphic medium, is demonstrated by reference to a working circuit design facility implemented on a minicomputer. The detailed nature of this facility and the techniques illustrated are related to the circuit design process and the behavioral characteristics of the human designer. Special attention is given to the command dialogue, and to the introduction of flexibility allowing a user to modify the facility to suit his own needs.

## I. INTRODUCTION

INDUSTRIAL circuit designers have long been aware of the computer's enormous potential, and the capabilities of a wide range of circuit analysis programs. Nevertheless, the use of such programs in design is not as extensive as it could be or as it was, perhaps, originally anticipated. Our principal thesis is that the major reason for this underexploitation is the lack of attention paid to one of the most important aspects of a computer-aided design (CAD) tool—the man-computer dialogue. How the designer uses a CAD tool, and how the tool communicates with the designer, is of paramount importance.

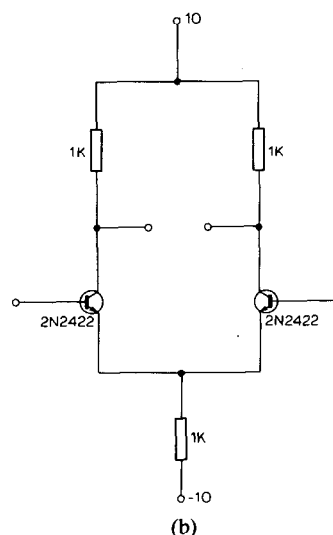
Two brief examples, more fully discussed later, will temporarily serve to illustrate this thesis, and simultaneously hint at the potential of interactive computer graphics as a medium in which to conduct the major part of the man-computer dialogue associated with circuit design [1], [2].

In the first example (Fig. 1) the circuit description format (a) is inferior to (b) in many respects. Fig. 1(a) requires reference to an already *conceived* and *drawn* circuit diagram, needs prior *labeling*, is nowhere as easy as (b) to “*read*” and *understand*, and is not as easy to *check* in view of the difficulty of detecting errors (e.g., *C* instead of *R* in row 2). Moreover, format (b) provides an excellent *basis* for displaying component-based circuit properties (as the second example will show), and needs *no relearning* following a substantial absence from the CAD system using this format.

The second example (Fig. 2) concerns the presentation of sensitivity information to enable the designer to gain a

```
B1 N(0,1), R=1000
B2 N(0,2), C=1000
B3 N(3,0), R=1000
B4 N(1,3), R=40000
T1 B(7,4), GM=.15
B5 N(2,3), R=40000
T2 B(6,5), BETA=50
```

(a)



(b)

Fig. 1. Alternative formats (a), (b) for circuit description.

feel for the significance in the frequency domain of the components of a circuit. In format (a) the sensitivity of voltage gain to small changes in component values is described by a numeric matrix whose rows and columns correspond to components and frequencies, respectively. In format (b) circles whose areas are proportional to sensitivity are superimposed on component symbols in a display which changes in discrete steps, at roughly 1/10-s intervals, as a pointer (at right) cycles through a frequency range of interest. Since the pointer can also be swept manually by light pen, the identification of components having substantial effect in one or more frequency ranges, or at some critical frequency, is far easier with (b). Format (a) has the additional disadvantage of requiring reference to a circuit description such as (a) of Fig. 1. Fig. 2 also provides a glimpse of the advantages of dynamic interaction and display.

The primary object of this paper is to describe, by reference to a working circuit design facility, the potential

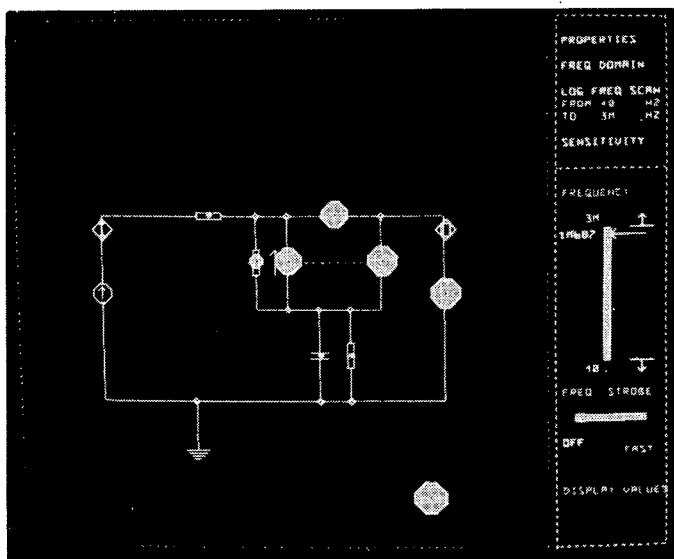
Manuscript received October 3, 1975. This work was supported by the U. K. Science Research Council.

R. Spence is with the Department of Electrical Engineering, Imperial College, London, England.

M. Apperley was with the Department of Electrical Engineering, Imperial College, London, England. He is now with the Department of Computer Science, University of Waikato, Hamilton, New Zealand.

0.30	0.44	0.15	0.62	0.52	0.52	0.77	0.35	0.51	0.61
0.17	0.27	0.18	0.81	0.32	0.81	0.63	0.32	0.82	0.21
0.32	0.32	0.95	0.55	0.42	0.41	0.92	0.77	0.63	0.92
0.36	0.15	0.89	0.75	0.67	0.44	0.35	0.36	0.61	0.99
0.56	0.39	0.35	0.89	0.35	0.32	0.74	0.62	0.81	0.91
0.39	0.17	0.25	0.45	0.33	0.61	0.66	0.22	0.11	0.15
0.47	0.29	0.85	0.44	0.81	0.82	0.13	0.17	0.63	0.21
0.45	0.48	0.25	0.27	0.19	0.98	0.91	0.31	0.37	0.45

(a)



(b)

Fig. 2. Alternative means of presenting sensitivity information.

of interactive graphics for circuit design and, by implication, for a much wider range of creative activities. It is therefore directed towards circuit designers who use CAD tools as well as the system architects who create them. The second objective is to focus attention upon the complex interaction between hardware, software, and the behavioral characteristics of the human designer. In so doing we underline the responsibilities of the system architect and identify quite a wide vista of research problems whose solution would immediately be relevant to the improvement of computer aided design.<sup>1</sup>

A suitable framework for the discussion is the simple model of an interactive-graphic CAD system shown in Fig. 3. The model serves the sole purpose of identifying the principal system components and attributes to which we shall refer, though it certainly places emphasis appropriate to our thesis. For example, it focuses attention upon the flow of data from designer to computer (as in circuit description) and vice versa (as in the display of circuit response). Similarly, it identifies control of the computer by the designer (as in choosing a response of interest), as well as control in the opposite direction (e.g.,

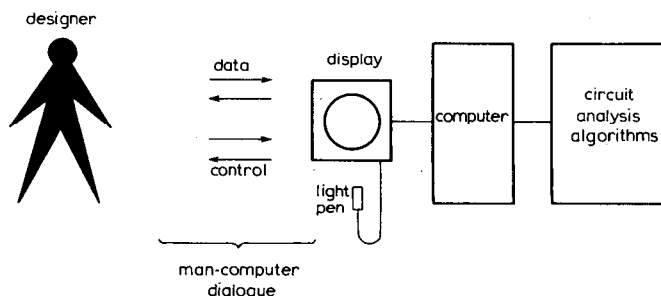


Fig. 3. Simple model of interactive-graphic system for circuit design.

prompts and default options). Conventional wisdom would have us inscribe, in the algorithm "box," large programs such as ECAP, ASTEC, or SCEPTRE. We do not do so, for the simple reason that an algorithm originally intended for use in batch-mode may be totally unsuited to design within an interactive graphic medium, and vice versa. A successful interactive-graphic circuit design facility is unlikely to result from the mere association of input and output graphics with a conventional batch-mode analysis program.

## II. DESIGN AND THE DESIGNER

It would be hazardous and irresponsible to embark upon the implementation of a CAD tool without first considering the intended use and user of that tool. For this reason we first examine the nature of the circuit design process, and then certain behavioral characteristics of the human being who practices the art of design.

### 2.1) The Circuit Design Process

The object of circuit design is to define an interconnection of available components to form a circuit whose performance meets some specification. The simplicity of this description is, of course, in stark contrast to the means by which the design objective is eventually achieved. For the vast majority of circuits the design process is extremely complex, involving elements of speculation, memory recall, theory, rules of thumb, simulation, search, intermediate goal formulation, and other processes. Nevertheless, though design is a complex process, pertinent comments by practitioners [3]–[8] and observers of this art provide useful pointers, not only to present day design practice, but also to aspects of design which are relevant to the fashioning of a new CAD tool. We summarize some of these comments below, loosely arranged under headings, and label them for future reference.<sup>2</sup>

#### The Problem:

- Design requirements are often ill-defined.

<sup>2</sup>By bracketed references.

<sup>1</sup>Open reference brackets, i.e. [ ], are used extensively in this paper to indicate the many problems on which research is still required.

- b) Only a finite set of component types, and often of values, is available.
- c) Circuit design is only one of the many responsibilities of the designer.

*The Approach:*

- d) Initial uncertainty exists about the approach to be adopted.
- e) Designers often ask "what if?" questions.
- f) Designers frequently speculate.
- g) Many changes in topology and values are examined.
- h) Component models must be proposed.
- i) "Scratch-pad" calculations are frequently made.

*Nature of Interest:*

- j) Qualitative aspects of behavior are often of greater interest than quantitative aspects.
- k) Designers seek insight into circuit behavior.
- l) Many circuit properties are simultaneously of interest.
- m) A feel for the direction in which search should proceed is useful.

Can any useful, general, and simply stated conclusions be drawn from these observations? We propose the following three.

- 1) In view of the many alternative circuits that it may be necessary to examine [a), d)–h)], an exceedingly simple and fast means of defining the new circuit and its property of interest is essential.
- 2) The format in which information concerning circuit behavior is described by the computer must be compatible with the designer's real needs, and easily interpreted [j), l), m)].
- 3) The calculation of the circuit property of interest should be rapid [e), f), k)].<sup>3</sup>

## 2.2) Behavioral Characteristics of the Designer

The man-computer dialogue associated with interactive-graphic circuit design is complex and little understood. However, certain behavioral characteristics of the human being have been established, either qualitatively or (less frequently) quantitatively, which appear to be directly relevant to the use of a CAD tool, and which suggest useful guidelines for the specification of a CAD system. Some of these characteristics are listed below: to economize on space the illustrative examples are necessarily terse, but a fuller discussion is provided by the quoted references.

A) *Short-Term Memory* [9], [10]: For example, the

<sup>3</sup>With respect to the speed with which both circuit definition can, and analysis does, proceed in the design facility to be described, concern has been expressed over the supposedly resulting tendency of the designer to "try this" rather than think. Two comments are, for the moment, relevant. One is that any designer must develop an intelligent approach to a new tool that simultaneously offers advantages and the potential for misuse. The other is that the subject of the effect of time delay is complex and as yet not wholly understood, and is discussed at greater length later.

human experiences difficulty in retaining in his memory a long telephone number, so that any interruption during dialing may either annoy excessively or cause loss of (short-term) memory.

B) *Psychological Closure* [9]–[11]: In the telephone example just quoted, the moment when dialing is complete and the short-term memory can be *unbuffered*, is referred to as psychological closure. Following closure, the human is less sensitive to otherwise potential irritants.

C) *Computer Response Time* [7], [9], [10], [12]: The speed with which the computed result appears after being requested can exert a considerable influence over the success with which the design problem is tackled, as well as the fatigue experienced by the designer [11], [13], [14].

D) *Uncertainty of Response to Control Actions* [7], [9], [11]: If an analysis is requested and there is no indication of reaction or progress by the computer, irritating uncertainty is caused: often a longer but predictable time delay is preferable to a possible shorter but less predictable delay. In the shorter term, some indication that a key has been depressed (e.g., click) or a light-button hit (e.g., brightening) is essential.

E) *Interpretation of Data* [9]: The human being excels at recognizing patterns amid a large volume of data.

F) *Familiarity of Terminology* [9], [11]: The dialogue between a designer and a computer should employ symbols and names that are familiar to him, not those which may be familiar to the programmer. For example, he wants to *plot* a circuit *property*, not "*execute*" a circuit "*analysis*."

We do not attempt to draw general conclusions from the above list. Rather, we emphasize the importance to CAD system design of those characteristics of design and the designer identified in 2.1) and 2.1), by reference to a working design facility. Bracketed references are used to refer the reader to items in sections 2.1) and 2.2). The system chosen for purposes of illustration is the MINNIE system for circuit design, [15], [16], implemented on a large minicomputer<sup>4</sup> with interactive graphic display.

## III. CIRCUIT DESCRIPTION

Familiarity, and the advantages enumerated in Section I, led to the choice of a more-or-less conventional circuit diagram as the format in which the component types and the manner of their interconnection are defined. Connecting wires follow a tracking-cross positioned by the activated<sup>5</sup> light pen, and are drawn much as with pencil on paper. A component selected by a light-pen hit from

<sup>4</sup>A PDP-15 with 32K core, 1.2M word disk, VT-15 display and printer. The MINNIE system was originally implemented to act as a test bed for innovation and exploration in man-computer dialogues, but is now also recognized as a useful design facility in its own right: it is continually being extended.

<sup>5</sup>For example, with the shutter opened.

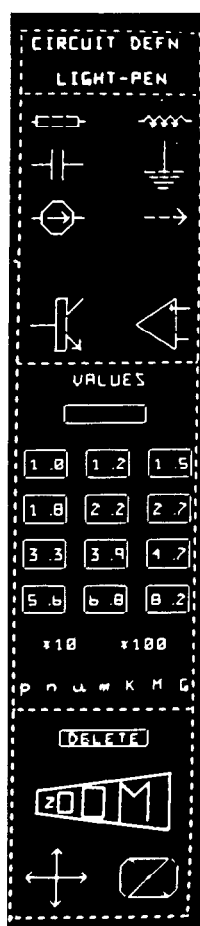


Fig. 4. Menu during circuit definition.

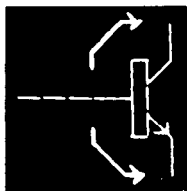


Fig. 5. Temporarily displayed arrows permitting rotation and interchange of connection.

the menu (Fig. 4, top) on the right-hand side of the screen appears at the last cross position in the direction of the last drawing action. If the component selected is a transistor or operational amplifier, the base or positive input, respectively, is connected, but rotational arrows (Fig. 5) permit other terminals to be connected instead. All wires and components are automatically adjusted, during drawing, to lie on an invisible reference grid (Fig. 6). Perhaps surprisingly, this does not degrade the drawing action, and it eases the data/display structure organization (see Section VI).

A value can be assigned to a component as soon as its symbol has been placed in the circuit diagram, allowing the component's complete identity to be specified at one time if desired: in this way the designer's flow of thought is less likely to suffer artificial interruption [A], (F), b)].

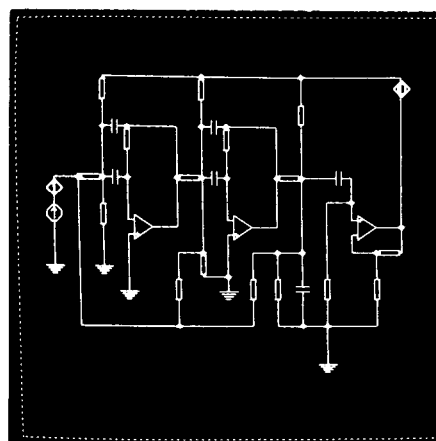


Fig. 6. Drawn circuit diagram.

Values in the 10 percent tolerance range are simply input by a maximum of three selections from the value menu [b)], (Fig. 4, middle). Other values and device type numbers can be entered via a keyboard. A selected value is assigned to a symbol just added or identified as soon as drawing recommences: in this way the definition of a circuit can take place smoothly [A)]. Assignment of a selected value to more than one component is achieved by pointing at the components one after the other.

Conventional mathematical models are associated with the resistor, capacitor, and inductor symbols. The voltage sensor can easily be linked with the current source symbol (otherwise denoting an independent source) to provide a voltage-controlled current source. A hybrid-pi model, with parameters  $r_x$ ,  $r_\pi$ ,  $C_\pi$ ,  $r_\mu$ ,  $g_m$ , and  $r_o$ , is associated with the transistor symbol, specific values of these parameters being associated with predefined type numbers which can be entered via the keyboard. The operational amplifier model includes the effects of gain, input and output resistance, and high-frequency rolloff. The ability of the designer to define new component symbols and their associated mathematical models is regarded as essential [ ], and research to this end is in progress. In the former case a light-pen "sketch" of the symbol, followed by computer reduction and justification, is an attractive objective. In the latter, the definition of the internal structure by a lumped model composed of menu items is one possibility.

Following the initiation of a bistable DELETE button, a single action suffices to delete components or wires. Movement to accommodate circuits larger than the screen size is possible; in this event zooming to permit viewing of the complete diagram—with an indication of the maximum drawing area (Fig. 7)—is possible. The present limit of 20 nodes and 50 branches (about to be increased to 40 nodes and 60 branches) is dictated by the storage demands of analysis programs rather than by the size of the data/display structure, but is soon to be further extended by exploiting sparsity.

A significant number of designers have wished to *compute* the value to be assigned to a component [i)]. To

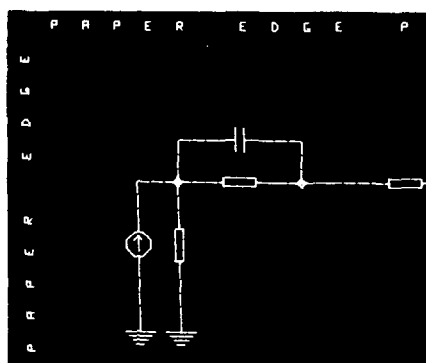


Fig. 7. Zooming permits viewing of large circuit.

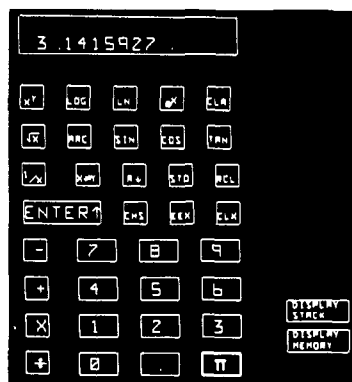


Fig. 8. Light calculator.

permit this activity most conveniently a "light" version of a popular pocket calculator (Fig. 8) has been implemented. A computed value appearing in its store can be assigned with the light pen to one or more components upon return to the displayed circuit diagram. Similarly, one of the inputs to the light calculator could be selected from the circuit diagram (e.g., a component value) or from a plotted circuit property.

Storage to eliminate the need for repeated circuit drawing is essential. This is automatic during examination of the circuit's properties—when the designer may wish to move frequently and rapidly between displays of circuit and response—since the display file relevant to the circuit diagram is retained in core. At the termination of a session, storage on paper tape or disk is possible, and hard-copy documentation of the circuit description and properties is easily generated for later reference. Nevertheless, there may be a need for some means of transferring a pencil-on-paper circuit diagram directly to the graphics screen, as an alternative input method [ ].

#### IV. CIRCUIT PROPERTIES

Clearly, a circuit property cannot be computed until it has been identified by the location of input and output nodes and by the numerical values of such parameters as the frequency range. However, because the effort of specifying these parameters can be so considerable as to



Fig. 9. Menu during examination of circuit property: at bottom are PLOT and CIRCUIT DISPLAY buttons.

detract [A), B), f)] from the design process, the procedure of the *default option* is employed extensively. In the simplest case this means that during the first circuit analysis of a given type (e.g., frequency domain), and unless the designer indicates otherwise, 1) input and output node symbols are associated with the left-most and right-most nodes, respectively, 2) the frequency range is set to 10 Hz to 1 MHz, and 3) the magnitude and phase of the voltage gain are the properties selected for display. This default choice is, however, very easy to override.<sup>6</sup> Moreover, the parameters specified either explicitly or by default automatically become the *new* default option until a change is made. The consequent "biasing" of the design facility to the designer's interest [A), B)] can save a great deal of time and irritation.

Following circuit definition and the identification of a property of interest, a light-pen hit on the PLOT button (Fig. 9, lower left) initiates the corresponding analysis. If

<sup>6</sup>For example, the input terminal location is moved by pointing the activated light pen, first, at the input node symbol in the menu and second, at the desired new location.

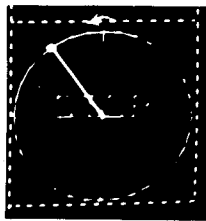


Fig. 10. Count-down clock showing progress of analysis.

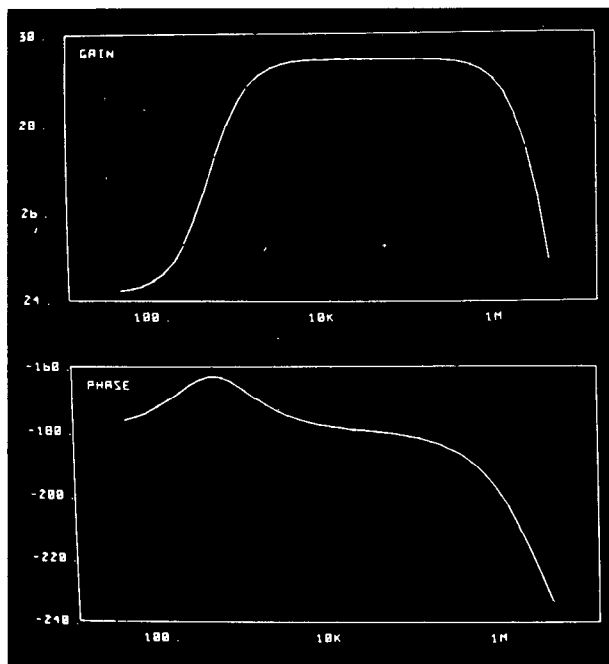


Fig. 11. Plots of magnitude and phase of voltage gain for circuit.

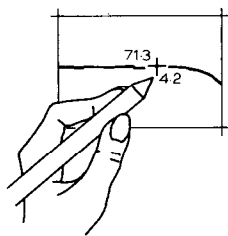


Fig. 12. Light coordinate meter.

the analysis is predicted to take longer than a second, a "count-down clock," (Fig. 10) whose hand performs one anti-clockwise revolution during the analysis, is displayed to remove the extreme irritation [D]) that results from a blank or static screen. A light-pen hit on the STOP button (Fig. 10) halts an analysis, though a second hit restarts it. At the termination of an analysis the circuit diagram is replaced by suitably scaled and dimensioned plots of the requested circuit property(ies) (Fig. 11). A detailed examination of the curves can be carried out by a coordinate meter option (Fig. 12) achieved simply by pointing the light pen at the plot.

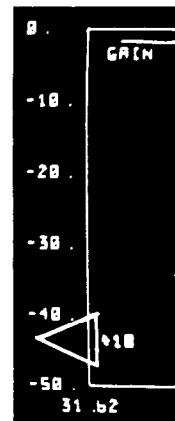


Fig. 13. Indication of desired 10 times reduction in lowest frequency.

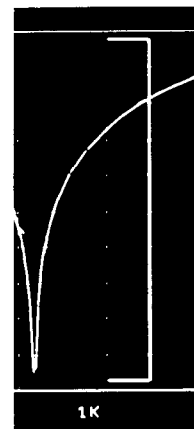


Fig. 14. Windowing to define new upper frequency limit.

It is frequently necessary to redefine the frequency range of interest following examination of the nominal response. Activation of the "lower frequency" button in the control area, followed by a light-pen hit on the bottom left-hand corner of the plot leads to a simple indication (Fig. 13) that the lower frequency has been reduced by a factor of ten. A further hit results in a double arrow indicating a 100 times reduction. The upper limit can similarly be increased; if, however, a decrease is desired, a light-pen hit on the high-frequency button can be followed by the dynamic<sup>7</sup> light pen positioning of the desired limit on the actual plot (Fig. 14).

If the circuit response changes very rapidly in some frequency range, with the consequent danger that the sampling frequencies are too widely spaced to indicate the required detail, additional resolution can be obtained by hitting the DETAIL button and positioning the limits of the range wherein additional analyses are required (Fig. 15). Additional sample points will then be computed in this range.

<sup>7</sup>The limit can be moved to and fro; only when the pen is deactivated is the limit accepted as being defined.

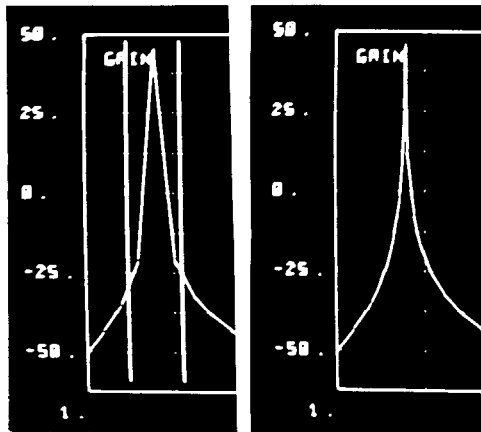


Fig. 15. Positioning of detail limits and resulting plot.

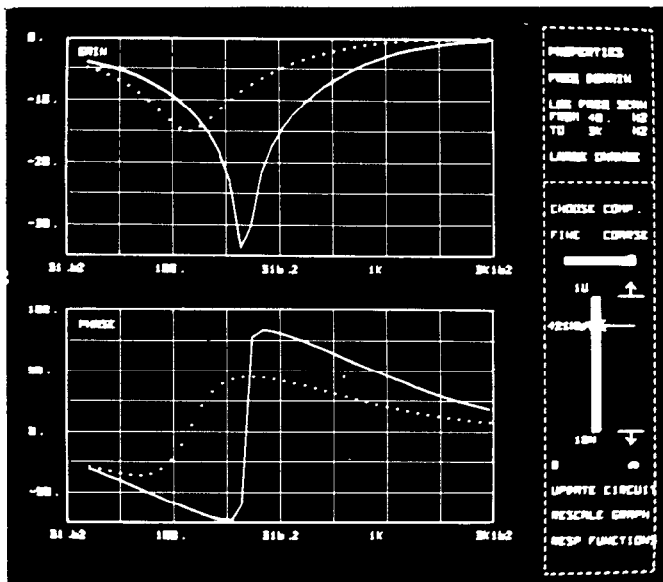


Fig. 16. New response due to changed component value.

#### 4.1) Large Component Changes

Once a circuit property has been plotted, the effect upon it of large variations in any component [17] may easily be explored dynamically [e)–g), m)]. Following identification of the variable component and a subsequent calculation, a light potentiometer is displayed. Its pointer can be moved by light pen to simulate changes in component value up to 10 times and down to 0.1 times its nominal (central) value: other scale factors are easily selected, as are zero and infinite component values. The light-pen movement is followed, *within about half-a-second*, by the display of the new response curve superimposed on the nominal plot (Fig. 16). Such a short response time, which is *independent of circuit size*, easily enables a “feel” to be gained for the effect of variations in a component [j), k)], allows smooth dynamic adjustment to optimize circuit response [m)], and allows speculative ideas easily to be tested [e), f)]. The short response time

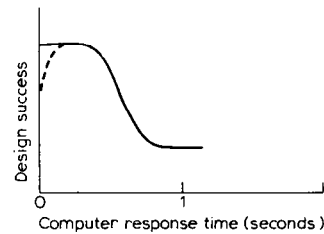


Fig. 17. Conjectured relation between some measure of design success and computer response time in interactive-graphic design facility.

does not simply speed up the design process; there are some ideas that would just not be tested with a longer response time, say 4 s, and which would be associated with a conscious “*trial-wait-observe error-decide-trial* . . .” activity as opposed to dynamic exploration [A)]. We suspect that the designer is prepared to tolerate the initial calculation involved if he can then avail himself of the dynamic exploration facility [C)].

The effect of computer response time on problem solving is highly relevant to the design of a CAD tool and, concomitantly, to the design of analysis algorithms. Research on this topic [9], [10], [12] has previously been associated largely with the use of interactive alphanumeric keyboards, and with response times ranging from seconds to minutes. Specific attention should now be given to the effect of response times of the order of one-tenth to two seconds within the interactive graphic medium [ ]. For example as a result of experience and observation we tentatively entertain the hypothesis, now being explored, that some measure of design success exhibits that form of variation with response time shown by the full curve in Fig. 17 though, with justification, some workers have suggested a variant indicated by the dashed line [ ]. Verification of such a hypothesis is not a simple matter, involving as it does the measure adopted for design success, the device employed to simulate component change, previous training [m)], and other factors. For example, use of the light pen can be criticized from a human factors viewpoint, since it is difficult to devote visual attention both to the potentiometer (to ensure proper registration of the light pen) and to the new response curve. It is for this reason that a manual potentiometer with software reset and scaling is to be provided, though the light potentiometer will be retained to provide easy value readout and a qualitative indication of location within a range [j)].

Efficient algorithms for computing the effect of *multiple* component changes [18], either one at a time or simultaneously, are available and have been incorporated in the interactive graphic design facility.

#### 4.2) Small Component Changes

The dynamic display of small-change sensitivity<sup>8</sup> [19],

<sup>8</sup>In the present context, “small-change sensitivity” implies the ratio of small percentage changes in response and component value.

[20] (Fig. 2(b)) can provide the designer with a "feel" [E] for the significance of each component in various parts of the frequency range [j), k)], an approximate indication of possible relative component tolerances, clues as to which components might be varied [m)] in order to achieve a desired circuit response, and an indication of components involved in resonance. If required, information concerning actual sensitivity values can be accessed at two levels. First, the reference circle (Fig. 2(b), bottom right) has an inscribed value. Second, a light-pen hit on the DISPLAY VALUES button will replace the circles by the corresponding numerical values. The approach illustrated in Fig. 2(b) is, perhaps, best summarized by Negroponte's [21] remark in the context of architectural design that

"One does not sketch with a 9H pencil, or make working drawings with a felt pen."

Fig. 2(b) is, of course, only one example of the visual encoding of data in symbolic form [22], an aspect of man-computer communication that has not been extensively explored [ ] in the context of both *interactive* and *dynamic* graphics. An interesting variant—in one sense a discrete example of the idea just discussed—is *selective display enhancement* [23], in which components possessing a common property (e.g., those carrying the majority of the current) are brightened. Effective application of this technique could greatly enhance the designer's insight [k)] into circuit behavior [ ].

#### 4.3) Tracking Sensitivity

If a group of (or all) components exhibit the same functional dependence upon some global variable such as temperature or radiation, this effect can be explored dynamically in the same way as for large component changes, (sec. 4.1)), following the assignment to each variable component of a coefficient value. The algorithm [24], [25] is such that, following an initial calculation, the time delay between the choice of a new value for the global variable and the appearance of points on the new response curve (Fig. 18) is about 1/3 s.

#### 4.4) Hazardous Parasitics

Since the designer is examining only a circuit model, he risks ignoring significant parasitic components. Potentially hazardous internode parasitic capacitances are brought to the designer's attention in the following way. At the designer's request, any such capacitance less than some specified value (1/2 pF by default), which individually would cause some specified error in predicted response, is automatically displayed in dashed form on the circuit diagram (Fig. 19). The potential applications of this information are many and varied, and well worth closer examination [ ].

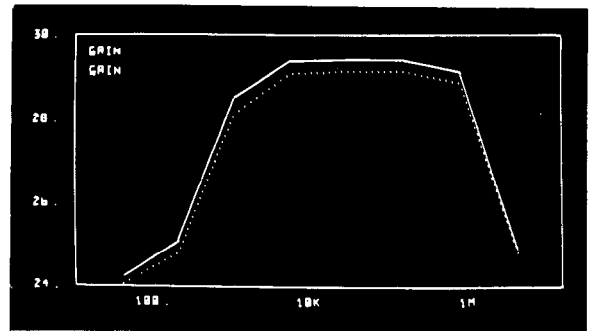


Fig. 18. New response due to temperature dependence of group of components.

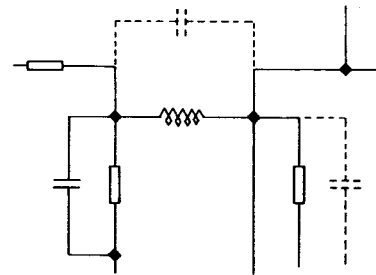


Fig. 19. Automatic display of hazardous parasitic capacitances.

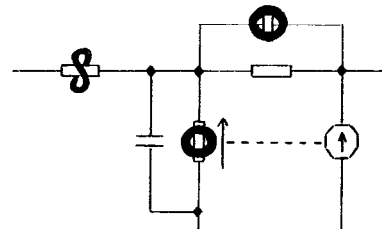


Fig. 20. Indication of redundant model components (S=short, O=open).

#### 4.5) Model Pessimization

A procedure known as model pessimization [26] will indicate (Fig. 20) those components of a designated device model which can be removed (by open or short circuit) without causing a deviation in circuit response greater than some specified tolerance. In fact, the tolerance may be varied manually in order to explore its effect upon component "redundancy." This procedure can provide the designer with insight [k)] into the relative importance of device model parameters [h)], and perhaps thereby reduce the cost of measurement prior to selection in manufacture. The nature and effectiveness of model pessimization particularly within an interactive-graphic medium, needs further exploration [ ].

#### 4.6) Performance Contours

A performance contour [27] shows how two components in a circuit may vary in value simultaneously—all



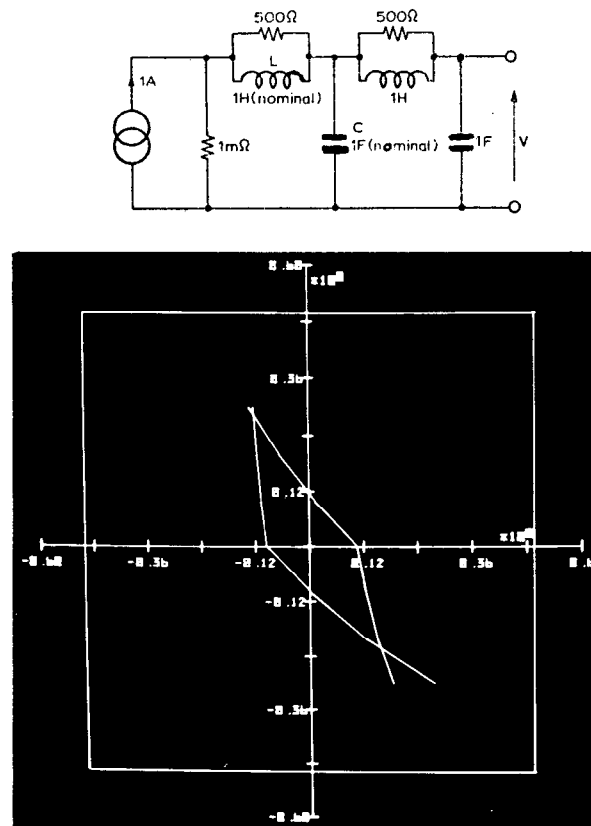


Fig. 21. A circuit, and performance contour for tolerance of  $\pm 10$  percent on  $|V|$  at frequencies of 30 and 300 Hz. Square shows  $\pm 50$  percent deviations in  $L$  (horizontal) and  $C$  (vertical).

other components remaining at their nominal values—if the circuit response must remain within a specified tolerance. Following the identification, by light pen, of components of interest, and provided circuit response specifications have already been entered, an analysis is carried out which then permits the performance contour for any identified pair of components to be immediately displayed. The display (Fig. 21) may assist in the assignment of component tolerances and in design centering [28].

#### 4.7) Other Circuit Properties

Other options already implemented within the MINNIE design facility will only be mentioned to illustrate both the range of man-computer dialogue problems that remain to be solved and the potential that can be explored.

For example, in automatic optimization [38], it is possible for the designer to specify, with the light pen, the variable components, the desired response curve and the weights associated with each of ten sample frequencies. On the display screen he can then observe progress as indicated by the actual response curve, a graph of cost function versus iteration step, the error gradients, and the current component values.

Recently discovered algorithms [29], [30], [39] have been implemented to allow statistical circuit analysis, such that

good approximations to the frequency and cumulative distributions of a circuit function can be displayed with little delay. In this type of analysis, the possibility of designing a display that would allow the critical components or component combinations [30] causing failure to be identified is attractive [ ]. The same type of algorithm used in statistical analysis has been extended to provide a rapid display of worst-case behavior [40].

Particularly critical in the context of a minicomputer is the man-computer dialogue associated with the control and display of a time-domain analysis, though the problem is no less important when, for example, the analysis is performed on a large machine accessed by a storage-tube graphic terminal [ ]. Finally, various proposals for the automatic or manually assisted growth or topological modification of a circuit to meet some specification pose many problems both in numerical methods [31] and man-computer dialogue [ ].

## V. COMMAND DIALOGUE

The command dialogue is the means by which a user communicates with a CAD tool. In an interactive graphics system, control is usually effected by pointing (with light pen or other device) to labels or light buttons displayed on the screen, or to objects or graphs produced as a result of

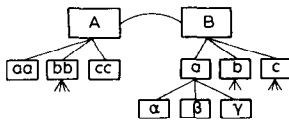


Fig. 22. Tree structure of command dialogue.

some previous action. The selection and sequence of these actions constitutes the dialogue within the discourse language of the system. In any CAD system the dialogue should ideally be completely asynchronous, so that any command is valid at any time. However, in most available graphic design systems the dialogue is inflexible, offering a limited set of operations to the user and presenting responses in a rigid format [32].

Interactive graphic systems are limited in that, at any time, it is possible to display only a limited number of light buttons and, therefore, to offer only a limited number of options. This leads to the inevitability of a "tree structure," such as that shown in Fig. 22, where at each state the user has open to him a number of choices, each of which will lead to a new state of the design facility. For example, *A* and *B* may be "circuit definition" and "circuit properties," respectively. At the next level, *aa*, *bb*, and *cc* may permit input by light pen, paper tape or disk, while *a*, *b*, and *c* may refer to the frequency, time, and quiescent domains. At the next lower level associated with circuit properties in the frequency domain, the designer may choose one of three analyses, for example nominal ( $\alpha$ ), large-change sensitivity ( $\beta$ ), or small-change sensitivity ( $\gamma$ ). Problems arise because the paths between two similar states in a typically large tree structure can be very complex, and the user may become lost. Based on a tree structure it is all too easy to design a dialogue which is clumsy and not easily learned [11].

In the MINNIE system these problems are overcome quite simply, by careful design of the visual indication of states and the control of movement between them. Suppose that the system is in state *A* (e.g., circuit definition). Light buttons (Fig. 23.1) offer a choice between 3 states (*aa*, *bb*, *cc*) within *A*. State *cc* (e.g., light-pen input) is chosen by light-pen-hit (Fig. 23.2) and the display changes (Fig. 23.3) to provide a summary (e.g., "circuit definition by light-pen input") of the state of the design system. After the use of state *A* is complete (e.g., when circuit definition is complete), button *A* is hit (Fig. 23.4). If the only other option at this level is *B* (e.g., circuit properties), then *B* will automatically appear (Fig. 23.5) together with buttons allowing choice at the next lower level. In this way, the designer may easily place the system in state *B*, *a*,  $\gamma$  (Fig. 23.9); this may, for example, allow small-change sensitivity analysis ( $\gamma$ ) in the frequency domain (*a*) of circuit properties (*B*). At all times, a horizontal bar separates selected states from presently available choices (Fig. 24).

Having used state *B*, *a*,  $\gamma$  the designer may wish to return to state *A*, *cc*—for example, to make a small change to the circuit diagram. All he needs to do is cancel button *B* (Fig. 23.10): since the only alternative to *B* is *A*,

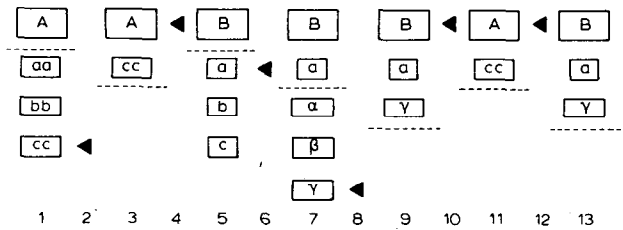


Fig. 23. Display and selection of states of design facility.

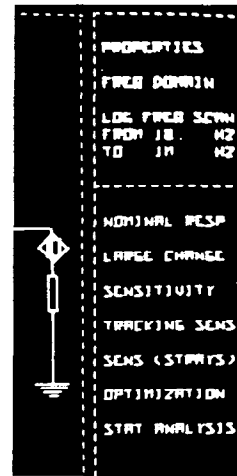


Fig. 24. Typical display in control area, showing summary of selected state and, below the line, available choices.

the system would in any case go to *A*. Additionally, however, since a record is kept of the option chosen at each state, this option is again chosen within *A*, so that the system is taken straight to state *A*, *cc* (circuit definition by light pen) (Fig. 23.11). Again, when *A* is cancelled, the default choice based on the record of previous choices is automatically selected to yield *B*, *a*,  $\gamma$  (Fig. 23.13).

By the scheme just described, movement through the structure can be rapid and straightforward, and has the additional benefits of a displayed summary and default choices based on past use of the facility [33].

## VI. FLEXIBILITY

The value of a tool resides partly in its flexibility: that is, the ease with which it can be modified to carry out a broader range of tasks than that to which it might nominally or conventionally be constrained [7]. Thus it is likely that many potential users of a circuit design facility will want to compute a circuit property that has not previously been programmed, or perhaps even to include a new component type in their circuit. Since the aim of a CAD system designer should be to allow the user to do what he wants rather than what the system designer *thought* he might want, it is important that the design facility should possess some measure of flexibility. Such flexibility is required at several levels: it may be necessary simply to change the expression for the property being computed, to

```

FREQUENCY RESPONSE FUNCTIONS

1  GAIN←10*LOG((VDR+VDR)*(VDI+VDI)/(VIR+VIR)*VII+VII)

2  PHASE←DEG(VDR ATAN VDI)-VIR ATAN VII

3  REAL←VIR

4  IMAG←VII

```

Fig. 25. Expressions defining normally calculated circuit properties.

expand the available range of symbols and models, or perhaps to provide for the design of a completely new type of network.

One measure of flexibility that has been implemented within the MINNIE facility [34] permits modification of the circuit property being computed, and was suggested by our experience [35] of the APL notation [36] and its interactive terminal implementation. One advantage of APL is its orientation towards operations upon arrays, since arrays pervade the process of circuit analysis. For example, the relation between scattering and admittance parameters of a circuit, conventionally written as [37]

$$S = -I + 2 \times (I + Y)^{-1}$$

may be expressed in APL notation<sup>9</sup> as

$$S \leftarrow (-I) + 2 \times \boxminus (I + Y).$$

An APL expression is also the program for its execution at an APL terminal. A third advantage is that APL can act as a concise, fluent, and productive link between a specialist designer and the programmer who maintains the CAD facility. Thus there is a real possibility that, on request, a design facility such as MINNIE can be modified, *within a few seconds*, to compute and display a new circuit property.

The response function flexibility implemented within MINNIE can be illustrated by the example of a designer of simulated inductances who wishes to examine a frequency plot of the effective inductance presented at the input terminal of his circuit. Either on his own, or with the assistance of the applications programmer [3], he performs the following sequence of actions. First, there is called to the screen a display, in APL-like<sup>10</sup> notation, of the expressions defining the circuit properties for which the facility is normally programmed (Fig. 25). The result of each

<sup>9</sup>A complex arithmetic implementation of APL is assumed.  $I$  is the unit matrix and can easily be arranged to be of the appropriate dimension.

<sup>10</sup>While the normal syntax and array orientation of APL are preserved and the available primitives are extended to handle complex numbers, differences in some symbols and a limitation on available primitives are imposed by the lack of an APL interpreter for the PDP-15 and the exploratory nature of the experiment.

```

EDIT      REDEFINE
LEQ_VII/2*PI*F

```

Fig. 26. Definition of new circuit property.

```

INPUT NODE  ↕
OUTPUT NODE ↕
RESP FUNCTIONS
GAIN  LEQ
PHASE IMAG

```

Fig. 27. Appearance of newly defined property in control menu.

expression is a variable whose name (GAIN, PHASE, REAL, or IMAG) is associated with the property, and which also appears in the control menu as a light button by means of which a plot of the property may be requested. The expressions contain certain global variables (e.g., VIR and VII, the real and imaginary input voltage vectors) as well as some predefined functions and constants (e.g., DIFF, DEG, PI).

The definition of a new circuit property is entered from the user's keyboard in the pseudo-APL notation, whereupon it appears in an editing buffer (Fig. 26). For a circuit driven by a unit current source at the input terminal, the appropriate expression for effective inductance is

$$LEQ\_VII/2*PI*F.$$

When this expression has been entered,<sup>11</sup> the typing of a terminator causes the expression to be compiled, a process taking less than one second. The new expression replaces a selected existing expression, and the name of the property also appears in the control menu (Fig. 27). Now the designer of simulated inductances can call his circuit to the screen, check that the input terminal is properly located, and select the LEQ button. After an analysis the effective inductance will be displayed as a function of frequency.

The global variables on which the function definitions are based are common to most of the analysis routines in MINNIE. Therefore, once a new function is defined, it becomes available, not only in the nominal analysis option but also, for example, in the dynamic exploration (large-change sensitivity), statistical analysis, and tracking sensitivity options.

At other levels, the modification of MINNIE is not so straight-forward, but is by no means a difficult task. A version of MINNIE to analyze digital filter networks, and called ICICLE, was implemented by a systems programmer in a few hours. This involved the provision of new sym-

<sup>11</sup>Full editing facilities are available, including the insertion and deletion of characters.

bols necessary for digital filters and used previously written analysis routines.

## VII. THE DATA STRUCTURE

A common fault of interactive graphic systems is that their data structure not only makes them cumbersome, but is generally so deeply imbedded that modifications at any level are difficult to undertake for anyone other than the system designer. For circuit diagrams MINNIE uses a data structure which is essentially simple and which lends itself to rapid modification for other types of network. As mentioned in Section III, a circuit is automatically adjusted to lie on an invisible reference grid. One unit of this grid is the length of a single component. Each element of the grid is described by a reference word which gives details of the item displayed: it may be a component, a wire, or nothing. The relation of positional information (such as is selected by the light pen) to circuit information is therefore very simple, and the implementation of circuit editing features such as deletion is not a difficult task.

Once drawn, the circuit is completely characterized by its display file and a conventional description table. Any positional information obtained during the course of an analysis definition (perhaps the positioning of an input node) can be rapidly interpreted by a simple routine. Similarly, the provision of positional information—such as component locations for the sensitivity display—is equally straight-forward.

## VIII. HARDWARE AND OPERATING SYSTEM

MINNIE is implemented on a PDP-15 computer with a VT-15 interactive display. The PDP-15 is an 18-bit machine with 32K of core, and the facility is dependent on the 1.2 million word disk for core overlays and data storage. The display shares the core of the machine on a cycle-stealing basis. The operating system (DOS V2A) is a single user disk system, providing a good framework for the overlay type of execution. A similar system could be implemented on any other minicomputer (16-bit words create no problem) and interactive display. The important requirements are 32K of core (for a stand-alone system) and a disk operating system.

An overlay structure has been devised for MINNIE which is closely related to the command dialogue [33]; indeed, the tree diagram of one is the tree diagram of the other. It is important that this relationship be maintained, since a delay during an interactive action, and caused by disk operation, can spoil the smooth action of a system.

## IX. DISCUSSION

Since the design of a CAD tool is largely an art, and our understanding of man-computer dialogues is in its

infancy, no "conclusions" in the conventional sense can be added to the above discussion. Our main objective has been to advocate an approach and describe some useful techniques. Partial support for our thesis is provided by strong industrial interest in the MINNIE system.

## ACKNOWLEDGMENT

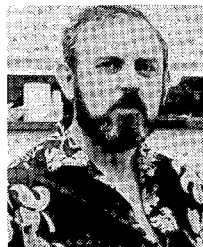
The programming assistance and advice of J. Dunn is very gratefully acknowledged. We also wish to thank Dr. D. Agnew, Prof. P. Bryant, and Dr. K. H. Leung for their collaboration. Financial support was provided by the U. K. Science Research Council.

## REFERENCES

- [1] H. C. So, "Analysis and interactive design of networks using on-line simulation," in *System Analysis by Digital Computer*, F. F. Kuo and J. F. Kaiser, Eds. New York: Wiley, 1966, ch. 2.
- [2] W. G. Magnuson, Jr., F. F. Kuo, and W. J. Walsh, "GINA—A computer graphics analysis system," *IEEE Trans. on Circuit Theory*, vol. CT-16, pp. 389–391, Aug. 1969.
- [3] S. W. Director and B. J. Karafin, Editorial (Special Issue on Computer-aided Design) *IEEE Trans. on Circuit Theory*, vol. CT-20, pp. 625–626, Nov. 1973.
- [4] D. W. G. Ingram, "Man-computer interaction in the design of telecommunication systems," in *IEE Conference Publication No. 68*, Sept. 1970, pp. 7–11.
- [5] Z. Fazarinc, "Designer-oriented CAD," *IEEE Trans. on Circuit Theory*, vol. CT-20, pp. 673–683, Nov. 1973.
- [6] A. Newell and H. A. Simon, *Human Problem Solving*, Englewood Cliffs, NJ: Prentice-Hall, 1972.
- [7] J. Hatvany, "The engineer's creative activity in a CAD environment," in *Computer-Aided Design*, J. Vlietstra and R. F. Wielinga, Eds. Amsterdam, The Netherlands: North-Holland, 1973, pp. 113–125.
- [8] R. Spence, "Interactive graphics in electrical circuit design: the user's requirements," *Convegno sulla Interazione Uomo-Macchina*, Sorrento: Informatica, II, 1, supplemento pp. 196–206, Apr. 1971.
- [9] J. Martin, *Design of Man-Computer Dialogues*. Englewood Cliffs, NJ: Prentice-Hall, 1973.
- [10] R. B. Miller, "Response time in man-computer conversational transactions," in *1968 AFIPS Conference Proceedings*. Washington, D.C.: Thompson, 1968, pp. 267–277.
- [11] J. D. Foley and V. L. Wallace, "The art of natural graphic man-machine conversation," *Proc. IEEE*, vol. 62, pp. 462–471, Apr. 1974.
- [12] H. Sackman, "Experimental investigation of user performance in time-shared computing systems," System Development Corporation, Santa Monica, CA, May 1967.
- [13] M. A. Sabin, comment in discussion of [7].
- [14] R. Spence and A. J. Drew, "Graphical exploration in electrical circuit design and modelling," in *Proc. 2nd Man-Computer Communication Seminar*, Ottawa, Canada, NRC, 1971.
- [15] R. Spence and M. D. Apperley, "On the use of interactive graphics in circuit design," Digest of papers, in *IEEE Int. Symp. on Circuits and Systems*, 1974, pp. 558–563.
- [16] M. D. Apperley, "User-oriented interactive graphics," in *DECUS Europe Conference Proceedings*, Zurich, Switzerland, 1974.
- [17] P. J. Goddard, P. A. Villalaz, and R. Spence, "Method for the efficient computation of the large-change sensitivity of linear non-reciprocal networks," *Electronics Letters*, pp. 112–113, Feb. 25, 1971.
- [18] K. H. Leung and R. Spence, "Multiparameter large-change sensitivity analysis, and systematic exploration," *IEEE Trans. on Circuits and Systems*, vol. CAS-22 pp. 796–804, Oct. 1975.
- [19] S. W. Director and R. A. Rohrer, "Automated network design—the frequency domain case," *IEEE Trans. on Circuit Theory*, vol. CT-16, pp. 330–337, Aug. 1969.
- [20] S. W. Director, "LU factorization in network sensitivity computations," *IEEE Trans. on Circuit Theory*, vol. CT-18, pp. 184–185, Jan. 1971.
- [21] N. Negroponte, *The Architecture Machine*. Cambridge, MA: MIT Press, 1970.

- [22] J. A. Coekin, "A versatile presentation of parameters for rapid recognition of total state," *IEE Conference Publication No. 68*, Sept. 1970.
- [23] R. B. Miller, "Man-machine interaction in the learning mode," *Convegno sulla Interazione Uomo-Macchina*, Sorrento: Informatica, II, I, supplemento, pp. 35-41, Apr. 1971.
- [24] K. H. Leung and R. Spence, "Tracking sensitivity: an efficient algorithm for linear nonreciprocal circuits," *Electronics Letters*, pp. 377-378, Sept. 5, 1974.
- [25] P. R. Bryant, "Tracking sensitivity: an alternative algorithm for linear nonreciprocal circuits," *Electronics Letters*, pp. 114-116 Mar. 6, 1975.
- [26] P. A. Villalaz and R. Spence, "A scheme for the elimination of redundant model complexity," *Electronics Letters*, pp. 38-40, Jan. 27, 1972.
- [27] E. M. Butler, "Realistic design using large-change sensitivities and performance contours," *IEEE Trans. on Circuit Theory*, vol. CT-18, pp. 58-66, Jan. 1971.
- [28] J. F. Pintel and K. A. Roberts, "Tolerance assignment in linear networks using nonlinear programming," *IEEE Trans. on Circuit Theory*, vol. CT-19, pp. 475-579, Sept. 1972.
- [29] K. H. Leung and R. Spence, "Efficient statistical circuit analysis," *Electronics Letters*, pp. 360-362, Aug. 22, 1974.
- [30] T. R. Scott and T. P. Walker, Jr., "Regionalization: A method for generating joint density estimates," in *Proc. 1975 IEEE Int. Symp. on Circuits and Systems*, 1975, pp. 325-328.
- [31] O. P. D. Cutteridge, "Computer synthesis of lumped linear networks of arbitrary structure," in *Network and Signal Theory*, J. O. Scanlan and J. K. Skwirzynski, Eds. London, England: Pergamon, pp. 105-111.
- [32] J. F. O'Callaghan, "Graphical communication in interactive computer graphics," *Australian Computer Journal*, pp. 12-19, Feb. 1971.
- [33] M. D. Apperley, "A dialogue structure for interactive computer systems," to be published.
- [34] M. D. Apperley and R. Spence, "Flexibility in a minicomputer CAD system," *Computer Aided Design*, Oct. 1975.
- [35] R. Spence, *Resistive Circuit Theory*, New York: McGraw-Hill, 1974.
- [36] A. D. Falkoff and K. E. Iverson, "The design of APL," *IBM Journal of Research and Development*, pp. 324-334, 1973.
- [37] H. J. Carlin and A. B. Giordano, *Network Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1964, p. 242.
- [38] D. G. Agnew, "Efficient use of the Hessian matrix for circuit optimization," *Proc. 1976 Int. Symp. on Circuits and Systems*, 1976, pp. 324-327.
- [39] K. H. Leung and R. Spence, "Idealized statistical models for low-cost linear circuit yield analysis," *IEEE Trans. on Circuits and Systems*, this issue pp. 62-66.
- [40] D. G. Agnew, "Systematic search and worst-case analysis," in *Proc. 1976 European Conference on Circuit Theory and Design*, Genova, Italy.

✦



**Robert Spence** (S'55-M'59-SM'76) is a Reader in electrical engineering in the Department of Electrical Engineering, Imperial College, London, England. Research interests have included circuit theory, linear active networks, injection-locked oscillators, and the use of computers in teaching, but are now concentrated on sensitivity analysis and the man-computer dialogue associated with CAD. His research group was responsible for the MINNIE interactive-graphic circuit design facility implemented on a large minicomputer. He is the author of *Linear Active Networks* (1970) and *Resistive Circuit Theory* (1974), and a co-author of *Tellegen's Theorem and Electrical Networks* (1970). He is now acting as General Editor of, and a contributor to, a new series of volumes on the CAD of electronic circuits, to begin in 1977. He is also co-editor of a new journal, *Electronic Circuits & Systems*, published by the Institution of Electrical Engineers (IEE).

Dr. Spence is a Fellow of the IEE.

✦

**Mark Apperley**, photograph and biography not available at time of publication.