RESEARCH ON TESTING AND EVALUATION PROCEDURES FOR INTERACTIVE MAN-COMPUTER SYSTEMS

Final Report

July 1969

This report is provided in the interest of information exchange and should not be construed as endorsement by NASA of the material presented. Responsibility for the contents resides with the organization that prepared it.

Prepared under Contract No. NAS12-2002 by
DUNLAP AND ASSOCIATES, INC.
Darien, Connecticut

Electronics Research Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Dr. J. H. Hill
Technical Monitor
NAS12-2002
Electronics Research Center
575 Technology Square
Cambridge, Massachusetts 02139

RESEARCH ON TESTING AND EVALUATION PROCEDURES FOR INTERACTIVE MAN-COMPUTER SYSTEMS

Final Report

July 1969

Prepared under Contract No. NAS 12-2002 by
DUNLAP AND ASSOCIATES, INC.
Darien, Connecticut

Electronics Research Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

TABLE OF CONTENTS

RESEARCH ON TESTING AND EVALUATION PROCEDURES FOR INTERACTIVE MAN-COMPUTER SYSTEMS

Dunlap and Associates, Inc.
Darien, Connecticut

SUMMARY

Active work on the project started on 16 September 1968. The system's displays, input devices and operator station(s) were inspected and evaluated. Human engineering interface problems involving the CRT displays and operator station(s) were identified. Design concepts for an Interactive Computer Graphics (ICG) research station were developed and engineering sketches and descriptive specifications were prepared. Later, under a modification to the original contract, two adjustable graphic-input-device stands for the ICG station were designed, fabricated and delivered to NASA. In addition, drawings were prepared of alternative configurations or arrangements for all current and planned equipment in the Laboratory's main computer facility.

In preparation for developing a research program of test and evaluation procedures, we (1) learned about the design concept, current status and planned development of the Laboratory's computer system, (2) surveyed the literature on interactive computer systems, computer graphics and computer-aided decision making, (3) categorized man-computer interface capabilities, information coding dimensions, computer applications and user problems and (4) developed a systematic research approach toward man-computer interface test and evaluation. A short-term and long-term research program were outlined. However, the scope and rate of progress in conducting the research program was drastically curtailed by its dependency on the evolving resources and changing environment of the computer research facility.

Procedures and an experimental method were developed and evaluated for testing the discriminability of the colors generated on the color CRT. Because of scheduling problems, equipment down time, the temporary removal of the display generator in April for modification, and the time required to revise the system software to be compatible with the hardware modifications, none of the other planned interface performance tests could be conducted within the period of this contract. Instead, a preliminary survey and analysis was made of the human factors aspects of program debugging.

INTRODUCTION

The objective of the Computer Research Laboratory's human engineering project, as stated in the contract is "to conceive, design and carry out procedures for testing and evaluating the human performance effectiveness of computer I/O devices and software. Based on the results there from a further objective is to formulate engineering guidelines for modification and/or development of computer I/O devices and software."

Dunlap and Associates, Inc., contracted to provide the Man-Computer Research Branch of the Computer Research Laboratory with essentially two types of human engineering support, a) consulting service as requested to test and evaluate by inspection alternative man-computer interfaces and to develop design recommendations and specifications for interactive Man-Computer research stations, and b) to design and carry out a program of performance studies to evaluate the effectiveness of man-computer interactions at these research stations.

The direction and pace of the human engineering research activities of necessity reflected the changing capabilities and limitations of the immediate system environment, its schedule and rate of development, and aimed to avoid duplicating the research efforts of other groups, in government, universities or industry. The selection of interface variables for experimental investigation, and the design and scale of the planned experiments were largely determined by the interface capability currently provided.

Our progress and expenditure of effort have been adjusted to make the most efficient use of the assigned personnel in relation to changes in their availability and in the status of the system. The scope of the work has included CRT console design, equipment and space layout, the development of test requirements and software for a test of the color CRT, collection of test data and the elaboration of research program plans including an analysis of the human factors problems of program debugging.

This report is intended to be comprehensive. It describes all the work performed during the period of the contract and draws on material already delivered as Quarterly Reports. Some additional material describing work in the last quarter is included also. The human engineering evaluation and design activities are described first, then the research planning activities and finally the research actually accomplished.

APPROACH

In order to advance toward our ultimate research objectives and to work productively within an ever-changing system and research environment, we followed a systematic approach. Firstly, we defined the immediate system environment in which the research must proceed. This involved learning the system concept, design, components, and functional capabilities as they existed and were planned. This task also involved the evaluation by inspection of elements of the system, such as the CRT displays and the configuration of equipment in the facility and the preparation of human engineering design specifications and recommendations.

Next, we surveyed the general context of human factors research into interactive man-computer systems with special emphasis on the use of graphics. Broadly speaking, we identified the principal centers for this kind of research, the general problems being studied, and the major categories of applications for computer graphics. The results of this survey were then expressed in terms of a general categorical description of man-computer interfaces that relates devices, formats, languages, users and problems or applications. This analysis and structuring of the research and application activities in this area provided a basis for identifying significant research problems, and certain general compatabilities among I/O devices, formats and uses, and suggested situations for measuring the effectiveness of man-computer interactions that have practical significance for NASA's current and projected missions.

Then, by relating this general identification of significant research problems and research contexts or user applications to the specific system environment of the Computer Research Laboratory, we "cut our coat to fit the cloth" and developed a program of studies phased to the planned development of the system's capabilities. Our program had to be provisional and was expected to be revised and developed to reflect the actual progress in evolving an experimental small scale time-shared general purpose interactive computer system.

The preparation for and the conduct of research activities provided the core or continuity for the project, but human engineering inspection and design activities were also performed as requested throughout the whole of the contract period.

VISUAL INSPECTION AND DESIGN ACTIVITIES

An understanding of the system concept and the current and planned system environment was necessary both to an evaluation of the designs of the systems consoles and I/O devices and to the preparation of an appropriate and realistically scheduled research program. Apart from studying the system documentation and questioning system engineers and programmers, learning the system involved project personnel being checked out in operating the computer and the writing and debugging of programs designed to test the capabilities of the system.

System Environment

The objective of current activity at the Man-Computer Research Branch was to bring into being a facility for the study of advanced designs in man-computer interactive systems. In particular, the goal was to enable a small community of users to converse with the computer and indirectly with each other in a highly interactive fashion through the use of graphic as well as alphanumeric representation. The intention was to evolve an interactive graphics subsystem embedded as a satellite in a powerful multiprocessor, resource-sharing system. However, prior to the realization of the full interactive graphics capability, as presently planned, a sequence of more limited interactive configurations were available. Earlier versions were operable only on a dedicated basis. Later ones functioned as part of a time-sharing environment.

A user may view the system at any time as a particular array of devices furnishing him a certain combination of capabilities. Table I lists the full set of planned terminal devices in the Fall of 1968 and indicates four subsets that constitute current and near future configurations of the system that were expected to be available on a dedicated or time-shared basis. Similarly, Table II lists the hardware resources invisible to the user (central processor, storage, and communication facilities), which, together with the terminal hardware configurations of Table I, their linkages, and appropriate operating system software, make possible the interface capabilities to be evaluated. In general, the system in the Fall of 1968 provided a manual input capability consisting of a teletype and an operable but unprogrammed light pen on the IDI display. The system display capability consisted of the teletype, an IDI monochrome CRT display with a graphic and alphanumeric character capability, and an ITT color CRT with only a graphic image capability.

It was anticipated that early in 1969 the system would have additional manual input capabilities resulting from the hardware and software development needed to operate the BBN and Sylvania graphic tablets, and the software for implementing particular light pen functions.

The then current and anticipated system interface capabilities are itemized in the following.

TABLE I
TERMINAL FACILITIES

	Dedicated (Fall 1968)	Dedicated (Early 1969)	Time-Shared (Initial)	Time-Shared (Later)
Input Devices				
Push buttom and toggle switch console	1	1	0	0
64-Character local teletype keyboard	1	>1	>1	>1
128-Character local teletype keyboard	0	1	?	1
Remote teletype keyboard	0	0	>1	>1
Local teletype paper tape reader	1	>1	>1	>1
Remote teletype paper tape reader	0	0	1	1
High speed paper tape reader	1	1	0	1
Function push buttom array	1	1	0	1
Light pen	1	1	0	1
RAND tablet	0	1	0	>1
SYLVANIA tablet	0	1	0	1
Control lever	0	1	0	1
Output Devices				
Console indicators panel	1	1	1	1
64-Character local teletype printer	1	>1	>1	>1
128-Character local teletype printer	0	1	?	1
Remote teletype printer	0	0	>1	>1
Local teletype paper tape punch	1	>1	>1	>1
Remote teletype paper tape punch	0	0	>1	>1
High speed paper tape punch	1	1	0	1
Refreshed monochrome CRT	1	>1	0	>1
Refreshed color CRT	1	1	0	1
Storage monochrome CRT	0	?	0	?
Hard copy device	0	1	1	1

TABLE II
CENTRAL PROCESSOR, STORAGE AND
COMMUNICATION FACILITIES

	Fall 1968	Early 1969	Future
DDP-516 Central processor	X	X	X
Paging hardware	X	X	X
16 Priority interrupt system	X	X	X
16K core memory module 1	X	X	X
16K core memory module 2		X	X
Magnetic drum		X	X
Magnetic disk 1	X	X	X
Magnetic disk 2		X	X
Magnetic disk 3			X
16 DMC Channels	X	X	X
Bit banger	X		
Teletype multiplexor		X	X
Input devices unit 1		X	X
Input devices unit 2			X
A-D/D-A	X	X	X

Current. -- Existing physical links and accessible system software (programs and library subroutines) make it possible for a user to:

1. Change the state of the sense switches, other flip-flops, live registers, and memory locations by operating console controls and display the state of these points via the console indicators;

2. Transmit and display alphanumeric strings on a single, locally connected, 64 ASCTT character teletype;

3. Display alphanumeric strings on the monochrome CRT, selected from a 128-character set and specifiable by the user as to position, orientation, size, and brightness;

4. Display geometric entities (visible dots, straight line segments, circles) on the monochrome CRT, specifiable by the user as to position, dimension, line structure, and brightness;

5. Display geometric entities on the color CRT, specifiable by the user as to position, dimension, line structure, and chromatic proprieties.

Future. --It was anticipated that in the near future (early 1969) and in addition to current capabilities, it would be possible for a user to:

1. Transmit and display alphanumeric strings on a single, locally connected, 128 ASCTT character teletype;

2. Transmit and display alphanumeric strings concurrently on multiple, locally connected teletypes;

3. "Pick" entities displayed on the monochrome or color CRT by means of the RAND tablet;

4. "Pick" entities displayed on the monochrome or color CRT by means of the SYLVANIA tablet;

5. Originate graphic entities by tracing or sketching on the RAND tablet;

6. Originate graphic entities by tracing or sketching on the SYLVANIA tablet;

7. Initiate any of a user-programmed set of functions by operating a pushbutton array;

8. Control the direction and rate of frame flow on the monochrome or color CRT by operating a lever device;

9. Record individual frames or frame sequence on the monochrome or color CRT by operating a "hard copy" device.

It was not then clear what the successive configurations of capability under time-sharing would be. It was thought that, most probably, the earliest available version of the system would permit the concurrent communication of alphanumeric strings via two or more remotely connected teletypes.

Interface Problems

The initial inspection and evaluation of the system's CRT displays and operator work station revealed two types of perceptual instabilities of the alphanumeric displays on the IDI monochrome CRT (flicker and jump phenomena); uncertainties over the perceptual qualities of the colors generated by the ITT color CRT, and certain human engineering inadequacies of the operator station (including the lack of a separate color CRT research station).

Monochrome CRT flicker phenomenon. --Flicker is a sensory effect produced when the eye is stimulated by flashes of light at regular intervals. The rate of stimulation at which the flashes fuse together and appear as a steady light is known as the critical fusion frequency (CFF). The CFF is proportional to the logarithm of the illumination intensity. When intensity is very low, flashes as infrequent as 3 or 4 per second appear as continuous light. When intensity is high (above 100 footlamberts), the CFF occurs at about 50 to 55 cycles per second. The CFF value also increases as the illuminated area increases and decreases as the ratio of light to dark periods increases. Consequently, the appearance of flicker on a CRT display will be a function of both the pulse repetition frequency (i. e., refresh rate) and persistence (i. e., decay rate) of the trace.

The IDI display has a P-31 phosphor which provides a bright short persistence image. Published data show that for sine modulated stimuli the CFF for a P-31 phosphor is 34 cps at a brightness of 10 footlamberts, 46 cps at 32 footlamberts, and 54 cps at 100 footlamberts.

The present IDI display is refreshed initially at 60 cps, but as more characters are presented, the refresh rate drops to 30 cps and a marked flicker occurs. With further loading of the display, the refresh rate drops to 20 cps.

Monochrome CRT jump phenomenon. --"Jump" is a phenomenon observed on bright phosphor tubes with a character repetition frequency (or "frame rate") of 30-60 cps, a character "on" time on the order of 10-50 microseconds, and a character decay to 50% brightness in less than 100 microseconds. All characters on the scope face appear to "jump" simultaneously in magnitude and direction each time viewers shift their gaze over the display or even about the room. Laboratory tests have confirmed that the phenomenon is related to the short persistence and high brightness of certain phosphors, such as the P-31 and the standard TV-tube phosphors. An increase in scope persistence (to the order of 10 milliseconds for 50% decay) or a decrease in brightness contrast will substantially reduce or will eliminate the phenomenon, which is due to the stroboscopic effect of the pulsing phosphor on the small, quick (20 millisecond duration) eye movements normally made by a viewer shifting his gaze over a display. (A much less pronounced, but identical effect may be perceived with any standard television receiver during rapid vertical eye movements.)

Both the flicker and the jump phenomena are associated with the bright short persistence P-31 phosphor used on the IDI display. The effects can be alleviated by decreasing display brightness, the brightness contrast ratio, and refreshing the display at about 60 cps. Both effects could be eliminated by changing to a slightly longer persistence phosphor.

Color CRT problems. --Apart from hardware problems that affect the quality and consistency of the color display images, there was a need to evaluate and define the qualities of the colors generated by the currently used color coordinate transformation programming system. The system designers needed to know the validity of the present color programming system and how it might need to be modified. Also, it was most desirable that the quality of the programmed colors be established before we attempted, in later experiments, to evaluate the effectiveness of color as a means to encode information on computer displays in the context of particular man-computer interactions.

Human engineering design tasks. --The system operator work station is far from being optimally arranged, but as yet no color display station exists and we, therefore, concentrated our initial human engineering design effort on providing alternative concepts for a color display station and, more generally, for an Interactive Computer Graphic research station. Figures 1 and 2 were prepared to show alternative configurations of the ICG station,

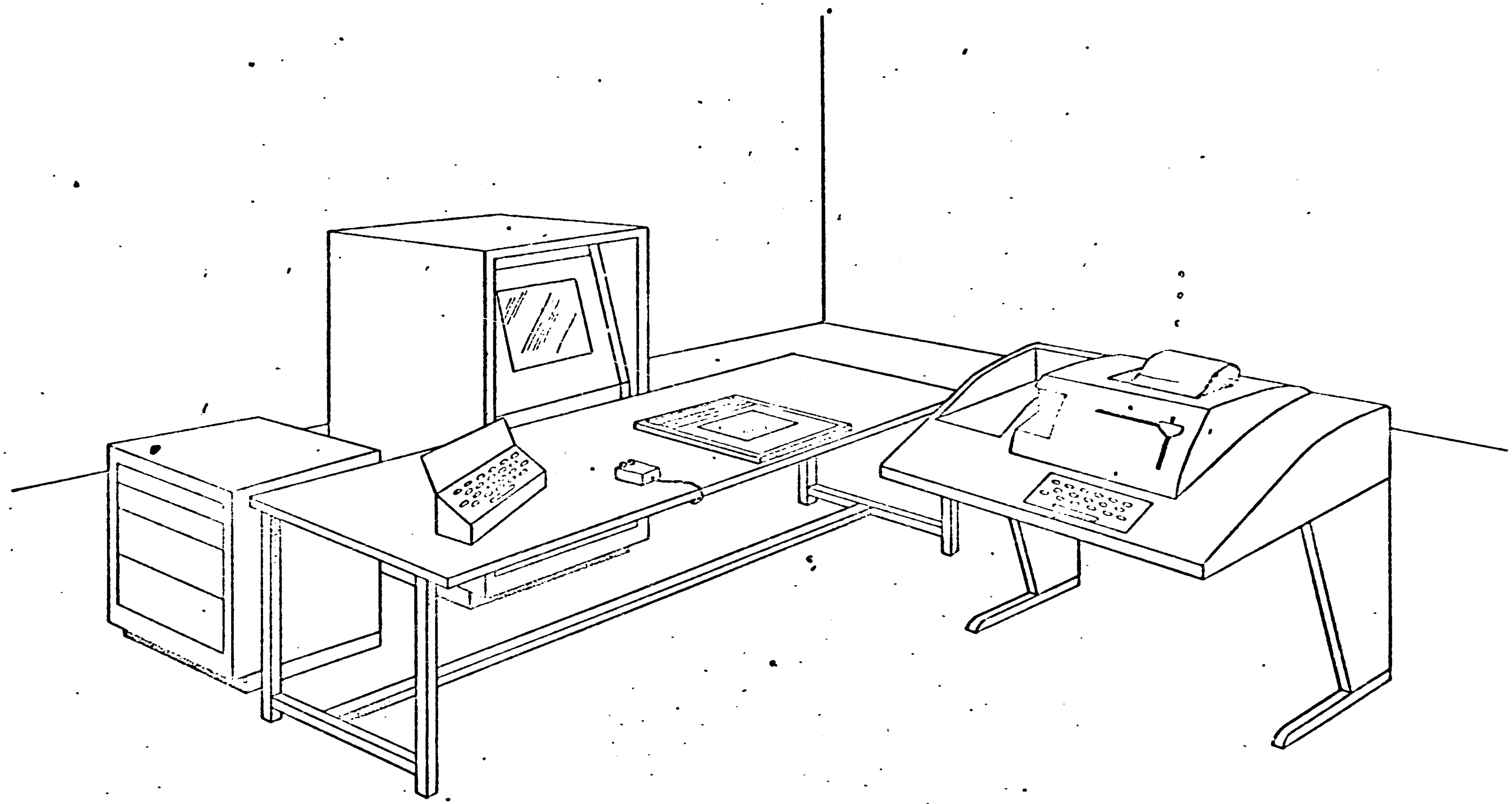


Figure 1. ICG Research Station - Configuration 1.

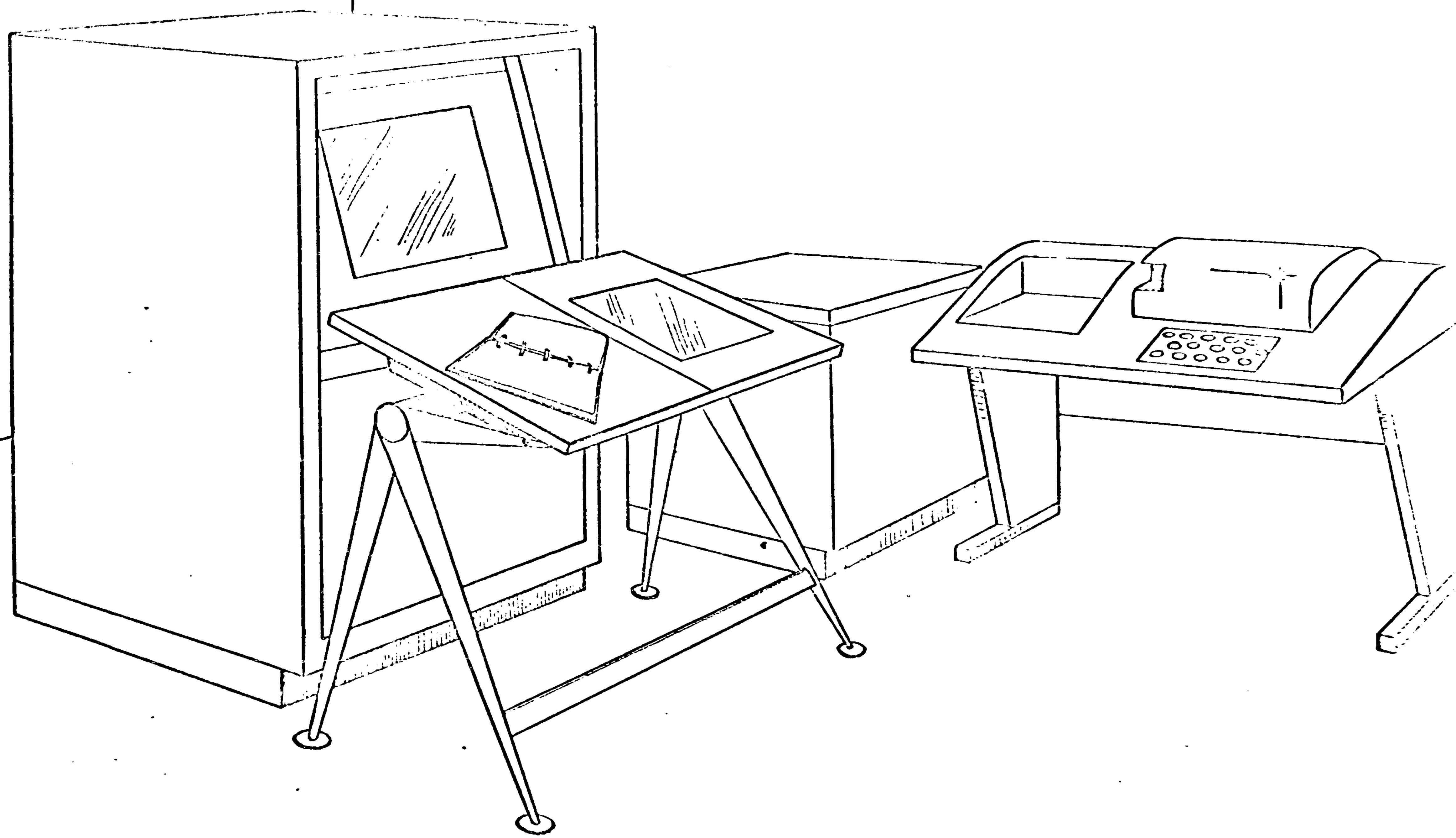


Figure 2. ICG Research Station - Configuration 2

using a table or an adjustable input equipment stand to support the input devices; Figure 3 illustrated alternative arrangements of input devices that could be mounted on the adjustable equipment stand; and Figure 4 was meant to demonstrate the possibility that a graphic tablet mounted on the adjustable equipment stand could be used with a projection type CRT to provide an Integrated Input/Output Station. The work station design concepts illustrated in these Figures were intended to reflect the research nature of this computer system and the system designers' concern for maximum configuration flexibility through design modularization. Specific descriptive specifications and engineering sketches were provided for a color CRT console cabinet, the Input Devices Unit (IDU) equipment rack, and the modular adjustable equipment stand of the Input Devices Station (IDS) shown in Figure 2.

Subsequently, we revised and prepared more detailed recommendations for a console cabinet for the color CRT and for an adjustable stand (Input Devices Station) for two graphic input devices. Finally, under a modification to our original contract, we fabricated two adjustable Input Devices Stations; one to accommodate the BBN Tablet and the other, the Sylvania Tablet.

A second human engineering task involved the preparation of different arrangements of the computer equipment, laboratory furniture and peripheral gear to provide an effective working environment for the operators and users of the computer research facility. The original four arrangements submitted for review and comment represented alternative trade-off solutions for the various requirements and considerations identified during discussion with the Laboratory's staff members. Some three months later, two recommended Laboratory arrangements were prepared. These reflected comments received from NASA on the original layouts, and revised lists of equipments to be accommodated now and projected for the near future.

SURVEY OF ICG RESEARCH

A limited survey of the literature on interactive computer graphic systems and research was conducted. It was concerned with identifying only major areas of current research and development activity that bear directly on the effectiveness of man-computer interactions, and specifically any projects similar to our own that were concerned with the formal experimental testing and evaluation of the effectiveness of the man-computer interaction.*

*An excellent survey of Computer CRT Displays (graphics, alphanumeric and applications) appeared recently in Modern Data's "Technology Profile," Parts 1, 2 and 3 in the July, August and September issues, 1968. See also the Selected Bibliography in this report.

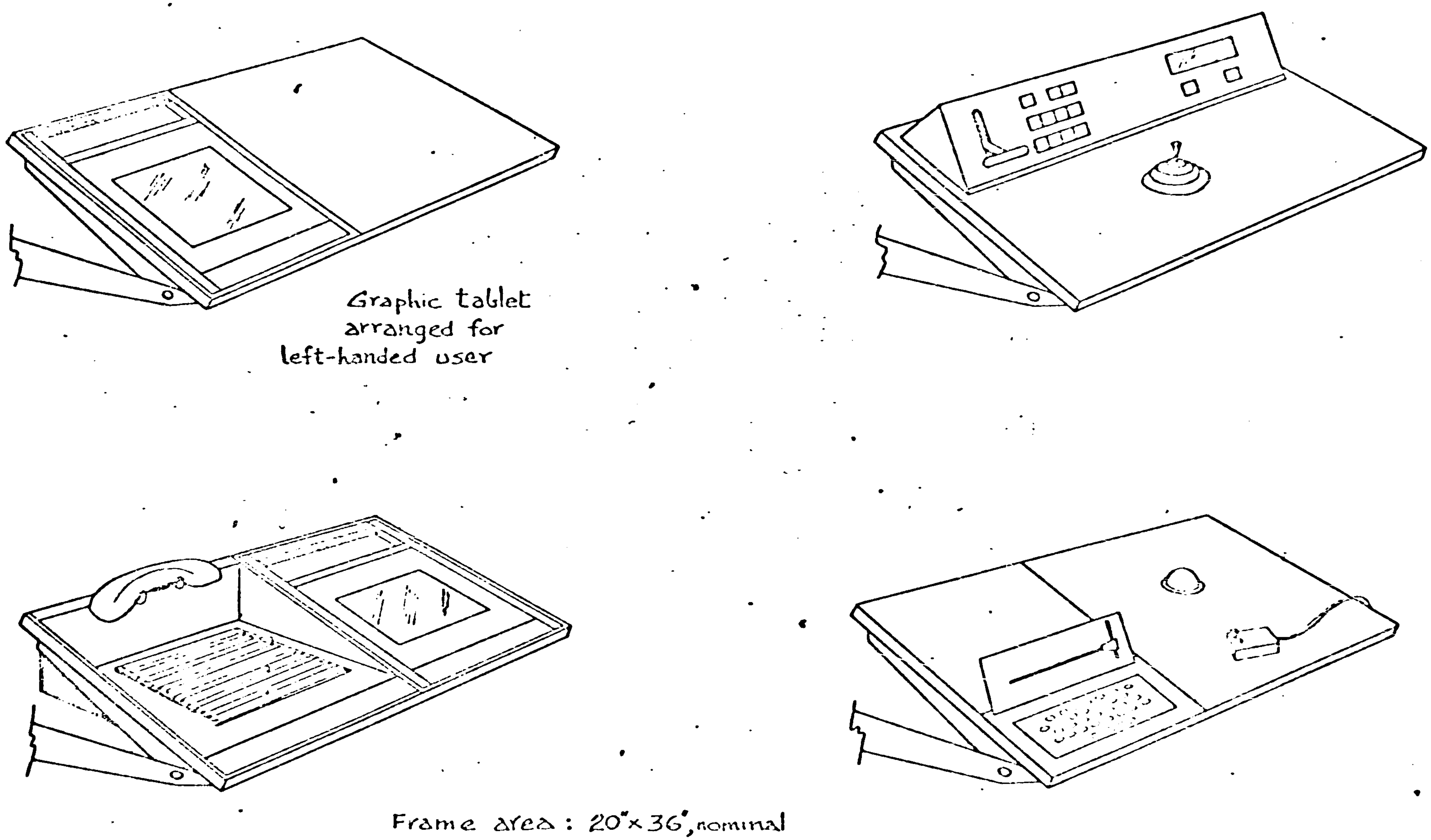
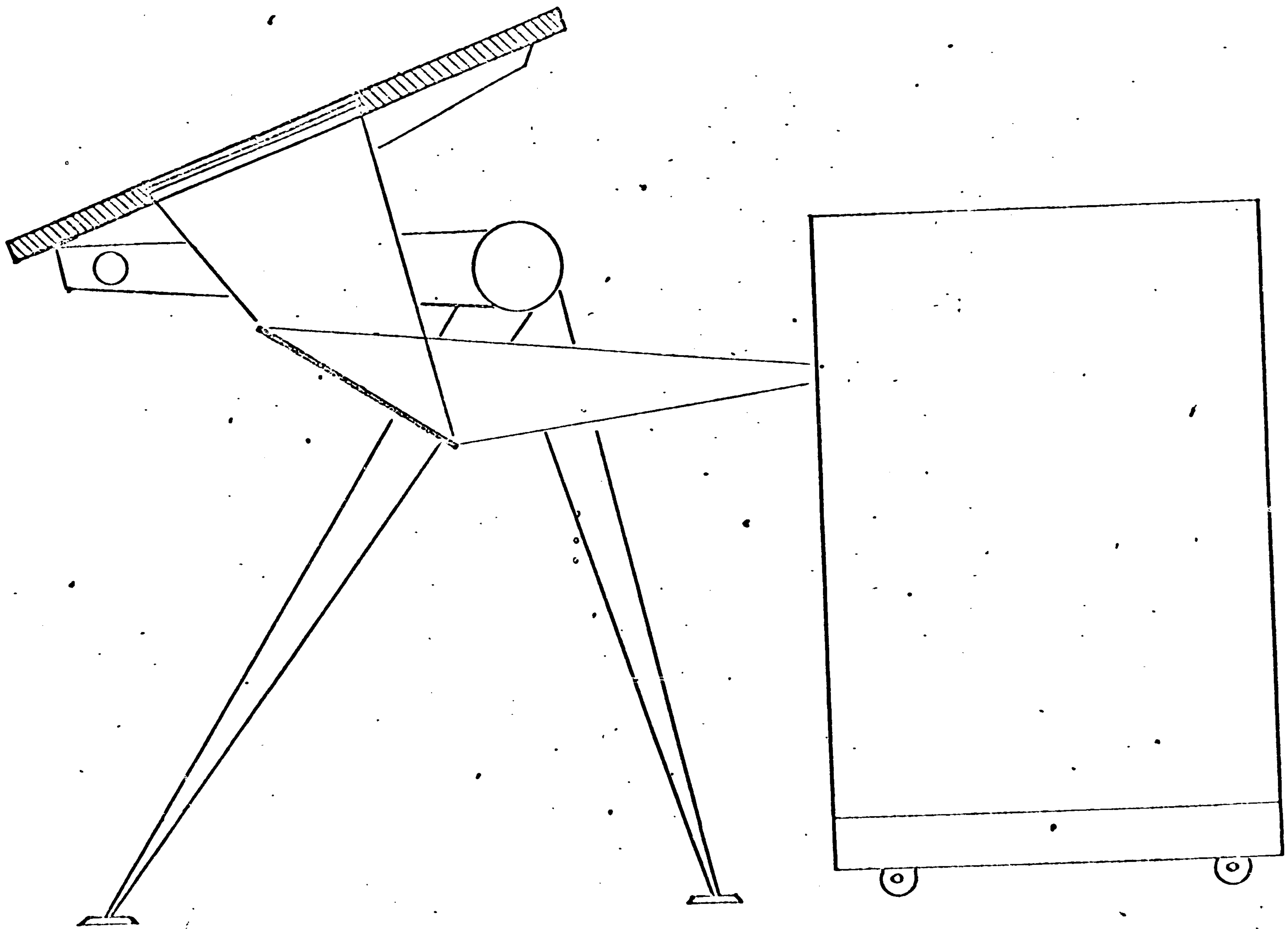


Figure 3. Input Devices Station - Arrangements

Figure 4. Schematic - Integrated I/O Station



Most developmental endeavors in computer graphics stem from underlying human factors considerations, but the acceptance of most new developments in interface hardware and software appears to rest on their "obvious" superiority over existing techniques for communicating with computers, rather than on systematic human engineering tests and evaluations of their usefulness.

This lack of formal testing is, however, understandable in view of the "primitive" levels or modes of communication that have existed until very recently between man and computer. Consequently, any increases in the facility and richness of communication resulting from both hardware and software developments are welcomed by computer users. Much of the hardware development in ICG research has concerned the production of brighter, higher resolution, flicker-free more sophisticated yet lower cost and more reliable CRT displays. Most of the innovations in display hardware have come from equipment manufacturers. Considerable emphasis is being given to the production of graphic CRT displays for remote terminal applications and studies of the adequacy of storage-tube based displays, such as at MIT Electronic Systems Laboratory and RCA's Princeton Laboratories, are typical of current interest in this problem.

The hardware of an interactive computer work station has typically consisted of a CRT display and a typewriter or teletype input device. Consequently, man's ability to communicate in a graphic mode with a computer has been severely constrained by the encoding limitations of his keyboard input device. The development of CRT light pen devices and recently of RAND tablet type graphic input devices (e. g., BBN's Grafacon and Sylvania's Data Tablet) represent significant advances in man-computer interface hardware. With these techniques both man and the computer can potentially communicate in similar, compatible alphanumeric, symbolic and graphic terms.

A variety of two dimensional "pointing" devices have been developed for use with CRT displays, such as joystick and "track-ball" cursor controls, and more recently SRI's "mouse" device. The Lincoln Laboratory's "Wand" represents an extension of the cursor control concept to entering three-dimensional coordinate information in a directly encodible form. A number of experimental studies were found that compared the speed and accuracy of using a light pen, joystick and track-ball for acquiring targets on CRT's. Generally, the light pen was reported superior for "picking" a target, and a track-ball for following a moving target. In a study at SRI reported by Englebart in 1965, he compared a light pen with such devices as the "mouse," grafacon, joystick and "knee control" for target positioning. He found the "mouse" followed by the light pen had an advantage over the other input devices in speed and error rate. However, since no tests of statistical significance were provided, it is impossible to judge the reliability of these findings. In a "Survey of Graphic Input Devices" published in Machine Design (August 1967), David Keast identified three uses for these devices: 1) inputting two- or three-dimensional pictorial material, 2) raw sketching of design concepts, and 3) selection of an item from an array or set of possible choices. He applied four criteria to the input devices: 1) they should be simple and natural to use, 2) they should be easy to interface to a computer, 3) the required software support must be as simple as possible, and 4) the device should have a low cost/performance ratio. His summary of a qualitative evaluation of nine input devices are given in Table III.

In a well-designed experimental study of two data entry methods, reported in "Perceptual and Motor Skills," 1965, Earl and Goff found that a point-in data entry method was a more accurate technique than either a type-in or mixed point-in, type-in data entry methods.

The development of a variety of graphic input devices has extended the range of methods available for encoding information, but the flexibility and capability of all these devices for entering data depend on the development of adequate software. Consequently, software developments to provide the display and input capabilities of the interactive man-computer interface represent the bulk of contributions to ICG research. Similarly, the communication potential of the CRT display depends on the availability of a range of display formats and methods of encoding information. In a previous project dealing with the design of computer-based displays for NASA prelaunch checkout activities, we surveyed published studies on the relative effectiveness of alternative display formats. We concluded that the generally equivocal findings in the literature related to differences in the tasks and measures of effectiveness used by different investigators. Subsequently, we analyzed the appropriateness of major format categories for displaying different kinds of information and conducted preliminary performance evaluations of various graphic formats for monitoring system test performances. We also analyzed the potential role of computer-driven displays as a thinking or decision aid for engineers in responding to non-normal operational test situations. Formats were designed in these later studies that attempted to take advantage of man's pattern recognition capabilities. We tried to present information in graphical forms that enabled viewers to perceive intuitively relationships and patterns in the underlying data that they might miss if the data were encoded alphanumerically.

The history of these studies shows that format requirements are not confined simply to the display; as issues of format are pursued, one comes to realize that the concept extends to the "format" of the interaction, the give-and-take between the man and the computer. The language of this interaction is a most important determinant of the effectiveness that is

TABLE III

COMPARISONS OF ON-LINE GRAPHIC INPUT DEVICES

Device	Principal Advantages	Principal Disadvantages	Resolution	Input From Hard Copy	Cost*	Comments
Light Pen	"Naturalness" of use with CRT display	Tracking required no hard-copy input	Low	No	Low	
Analog Sheet Encoder	Hard-copy input; no tracking required	Analog drift, low accuracy	Low	Yes	Low	Some versions not commercially available
Digital Pressure-Sensing Encoder	Uses ordinary pencil or pen	Potential mechanical failure	High	Yes	Likely to be high	Not commercially available
Rand Tablet	Hard-copy input; no tracking required	Small size	High	Yes	High	
Lincoln Wand	Three-dimensional	No hard-copy input	Medium	No	Likely to be medium	Not commercially available
Rho-Theta Transducer	Can accommodate large drawings	Requires polar-to-Cartesian transform	Medium	Yes	Low	
Datacoder	Low cost	Mechanical encumbrance	Medium	Yes	Low	
Track-Ball & Joy Stick	No tracking required	No built-in positional feedback; no hard-copy input	Medium+	No	Medium	
SRI Mouse	Speed and naturalness	No hard-copy input	Medium+	Adaptable	Low	Not commercially available

*Not including display and/or A/D converters, if required. + Basically cursor-positioning devices.

achieved. The development of conversational mode languages, such as JOSS, "problem-oriented" systems such as AMTRAN which can be used with both teletype and graphic terminals, and graphic language systems such as ALPINE and DAC-1 and more recently the RAND company's GRAIL system illustrate the high degree to which the effectiveness of a man-computer interface depends on having a language appropriate for a population of users or for the problems or applications of interest. The significance of developments in computer input/output hardware can be measured indirectly by the potential reduction in software costs that could be achieved to still attain a given interactive man-computer capability. Tests of the effectiveness of alternative interface configurations must include, therefore, not only measures of performance effectiveness but also some measure of the software "cost" of providing specified capabilities.

Most published studies of the effectiveness of input/output devices concern only partial tasks, limited input/output functions, and apply specific proximal criteria of effectiveness. An exception is the work of Landis and co-authors at the Franklin Research Institute on the evaluation of large-scale displays and their ingenious attempts to develop a Decision-Quality-Metric. However, their method of establishing the aggregate "pay-off" of a succession of cost/benefit type decisions in a highly structured problem does not provide a general test method as it is inappropriate for many tasks that involve inductive as well as deductive thought processes.

Many authors have published surveys of developments in interactive computer systems and each has had his own method for classifying the technological advances and system applications. Our purpose has been to develop a general method for classifying the various developments and applications that will help identify and select for research those human factor's problems that most often limit the effectiveness of man-computer interactions. We developed a general concept of the man-computer interface that relates computer applications to the language or format of the information exchange and to the interface input/output hardware.

MAN-COMPUTER INTERFACE DIMENSIONS

Research and development activities concerned with the man-computer interface are centered in three major areas, input/output devices, languages and information formats, and system applications. We have found it helpful to think of a progression for output devices ranging from event/status lights through teletypes, high speed printers, plotters to CRT displays; for input devices ranging from push buttons and switches through keyboards, and light pens to RAND tablets; and for languages and formats ranging from rigid alphanumeric operator-operands syntac, through algebraic/mathematical macro languages to the new high level graphic self-expanding languages with a parallel progression in formats from line texts, through matrices and scalar formats to diagrams and sketches. System applications reflect types of problems, levels of information and distinctions between using information and generating information; we can think of applications as located variously on a dimension ranging from Information, Storage and Retrieval (such as a library), through Bookkeeping and Inventory (such as seat reservation systems) Scientific Computation, and Engineering Design to large hybrid Management systems. This multi-dimensional description of the man-computer interface is represented diagramatically in Figure 5.

We have observed that for different types of application there seems to be sets of input/output devices, languages and formats that enable man and computer to interact in the most natural and efficient manner to deal with particular classes of problems. We would generalize from our observations of successful applications and say that for man and computer to interact effectively it is necessary to have compatible forms of information transformation as encoding from problem to man, man to computer, computer to man and man to problem.

To the extent that man-computer interface hardware and software enable a man to address information about a problem to a computer in a natural language and the computer to respond with information in a format appropriate to its characteristics and to the problem, they thereby contribute directly toward effective interactions. For different applications or sets of problems, some different combinations of devices and interface software would seem most natural and effective. For example, in a computerized library man and computer could interact effectively via a teletype terminal using alphanumeric English-type language with a text format. In the case of a scientific data processing system, a man needs an algebraic language, a capability of entering special symbols in text and matrix (array x column) formats and of seeing the computer output encoded variously in algebraic, alphanumeric and graphic symbols in text, matrix and scalar formats. In this latter case, the man-computer interactions could not be fully effective, if a teletype were the only interface device provided. The opportunity that a CRT and light pen provides for entering data into a computer by a pointing action avoids the need for a more complex learned skill such as typing. Similarly, the newer graphic input devices permit a man to write algebraic expressions in conventional multi-line formats, as well as make diagrams and line drawings for direct entry into the computer without the need for encoding the information in some more artificial abstract and complex language.

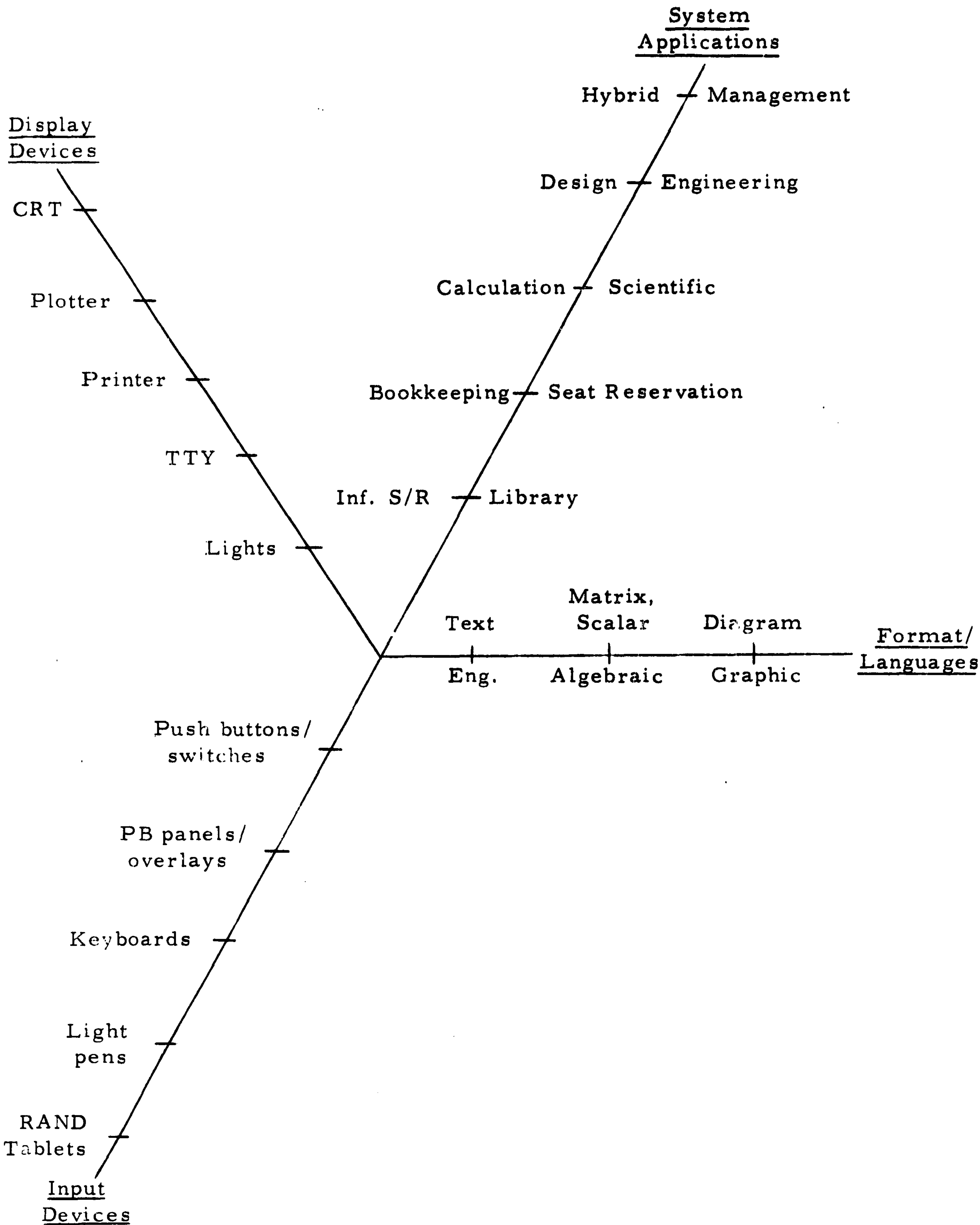


Figure 5. Dimensions of the Man-Computer Interface

RESEARCH PROGRAM PLAN

Our research test and evaluation program was concerned directly with the general problem of the effect of information transformations, information coding and formating, on the timing and effectiveness of the interactions between man and computer. More specifically, we planned to explore and test the effects of different input devices, different display coding variables (such as color) and display formats on the effectiveness of interactions, in such major categories of applications as information search and retrieval (library search), calculation or computation and design problem solution. In particular, we planned to be involved deeply with the problem of devising appropriate methods of observing, recording and quantifying man-computer interactions and developing relevant and valid criteria for evaluating the effectiveness of these interactions. We planned to obtain whatever objective measures of performance effectiveness were possible, as for example, times to solution, number of interactions, number of "false" leads or incorrect display requests, but we would certainly not disregard the comments, criticisms and subjective evaluations of the men using the system.

Experimental Test Designs

Basically, two forms of test design were envisaged. One form of test would be concerned primarily with a figure of merit of the outcome; the other form of test would be concerned primarily with the fluency of the process for gaining an outcome. Provisionally, we associated the first form with the categories of "calculation" and "library search," and the second form with "problem solving." However, this distinction is not absolute; it is more a question of emphasis. The nature of the test and evaluation procedures is summarized in Table IV under the titles of Form A Design and Form B Design.

As far as possible, each test and evaluation would be made quantitative and thus accessible for statistical reduction and analysis. The analysis of this data would serve to guide us toward more efficient and more effective test procedures.

Both forms of test and evaluation procedures were to be conducted with each of two or three I/O systems differing in equipment and programmed capabilities, e.g., black and white versus color tubes, RAND tablets versus light pen, light pen versus keyboard, etc. All subjects would perform on each of the systems so that they serve as their own "experimental controls." Normal experimental procedures for identifying or controlling sources of

TABLE IV
SUMMARY OF TEST DESIGNS

Design	Objects of Measurement	Measurement Procedures	Appropriate Criteria	Basis of Evaluation	Relevant Statistical Techniques
Form A	Performance of tasks of "calculation" and "library search" type	. Objective Documentation and measurement of performance in terms of . Speed . Accuracy . Economy of effort . Etc.	. Effectiveness and efficiency in gaining the solution	. Comparison of performance measures associated with different devices; analysis of individual differences	. Analysis of variance and/or regression analysis
		. Subjective Subjects account of use procedures	. Presence of difficulties in use	. Consolidation and analysis of ratings and comments to gain profile of user opinion	. Non-parametric techniques to examine relative frequencies and associations among data categories; and content analysis of comments

Table IV (continued)

Summary of Test Designs

Design	Objects of Measurement	Measurement Procedures	Appropriate Criteria	Basis of Evaluation	Relevant Statistical Techniques
Form B	Performance of tasks of "problem solving" type	. Objective Time history record of operations including: . Nature of step . Apparent purpose . Time . Errors . Adequacy of outcome	. Fluency of progress toward solution in form of absence of impasses, absence of repeat operations, evenness of operations, etc., also adequacy of outcome as judged against specification or by experts	. Comparison of "fluency" measures associated with different devices; analysis of individual differences	. Mainly non-parametric techniques to examine relative frequencies and associations among data categories
		. Subjective Subjects account of use	. Presence of difficulties in use	. Consolidation and analysis of ratings and comments to gain profile of user opinion	. Non-parametric techniques to analyze comments; and content analysis of comments

error such as counter-balancing, etc., were to be taken. The main results would be normative in character so that they might lead toward specifying I/O systems which were satisfactory for all users, but we suspected that considerable individual differences might occur. The consequences of these individual differences would be a function of their nature. For example, they might indicate the need for extensive training in order to enable some persons, otherwise competent, to use the systems effectively. Alternatively, they might indicate that different persons have different "styles," which could be equally effective but which were most compatible with some particular form of the I/O interface. These matters would be, therefore, a subject for analysis.

The experiences gained from each preliminary experiment were to be used iteratively to improve our provisional test and evaluation procedures. The revised and improved procedures would, in effect, define a complete test and evaluation program.

The basic forms of test described above represent in a sense "ideal" tests. They assume a well-developed system and a substantial data base with which experimental subjects could interact in executing computational problems, interrogating files and executing specific data search and retrieval tasks or in the sometimes iterative, open-ended process of problem solving.

But because of the evolving nature of the system environment in the Computer Laboratory's research facilities, the above test designs were not possible. We planned a program of tests and evaluations that would be responsive to the immediate needs of the system's designers for specific parametric testing and could develop progressively in scope and research significance from tests using simple laboratory type part-tasks to complex concrete problems in a simulated operational situation.

It was, in our opinion, quite unrealistic at the time of this contract to design performance tests that simulated complex operational interactive man-computer systems. The computer research facility did not have the necessary resources for such studies and is unlikely to have them in the near future.

Preliminary Research Program

The broad outline of our human engineering research program included the three phases depicted in our statement of work. Phase I of the work was essentially the human engineering design support provided on request, i.e., to inspect and evaluate the systems I/O devices and their arrangement and to make design recommendations. Phase I testing by inspection continued throughout most of the period of the present contract. This was because the Laboratory's computer research facilities are more experimental and at an earlier stage of development than had been presumed when the project started. Phase II test planning and evaluation work commenced as originally scheduled and continued as the major effort of the project. The Phase III validatory testing was smaller than originally planned because of budget and time limitations.

The general concept of our test program plan was to progress from simple short tests based on laboratory type part-tasks to complex longer tests based on concrete "real world" type problem situation. For example, one series of tests that was considered was as follows:

1. A test of the stimulus characteristics of the color CRT display.

2. Performance tests and evaluation of a keyboard, light pen and graphic tablet for constructing a graphic CRT display (monochrome or color).

3. Performance test and evaluation of the interactive effectiveness of three input devices and a CRT for an abstract design problem.

4. Performance test and evaluation of above I/O devices for a simple simulated "real" problem.

5. Ditto as in 4 but for complex problem.

6. Performance test and evaluation of use of color in interactive "real" library, scientific computation and design graphics applications.

7. Use of operational problem situations.

The tests planned for this research program were short and relatively simple, but they would require progressively more man-computer interactive capability. In this way, we hoped to encourage the development of the system's interactive mode of operation without over-extending its resources in any one test. All three phases of this human engineering test program should continue as the system develops. At later stages of system development, more demanding and more valid performance tests could be developed and more significant measures of the man-computer interactive process obtained.

Modified Research Program

The first test in the program, an evaluation of the colors generated on the ITT color CRT was planned to consist of three experiments. Programming, debugging and scheduling problems were encountered during preparation that delayed the start of testing, but we hoped to complete pilot tests of Experiments I and II during February, and to obtain sufficient data on which to specify the stimuli for Experiment III.

In March, while the display generator was expected to be "out" for modification, we planned to analyze the pilot data, develop the Experiment III stimulus programs, and start exploring the test requirements for possible performance effectiveness experiments.

In April, when the display generator was expected to be back in service, we had hoped to conduct the main data collection trials for Experiments I and II of the color test, debug the Experiment III programs and then collect data under the conditions of Experiment III. The time taken to collect the color test data under stable experimental conditions on five or six observers with color-normal vision was expected to depend on the effect of schedules for hardware and software modification and debugging activities on access to the display scopes for periods of approximately two hours at a time. We hoped to complete the color test data collection trials in April. In any event, we anticipated that during April we would continue to develop detailed requirements for a performance effectiveness test.

The first performance test could have evaluated the effectiveness and the amount of programming support required for using a keyboard, and a light pen (and possibly a graphic tablet also) to generate on the CRT on separate occasions an alpha-numeric statement, an algebraic formula and a geometric figure. Alternatively, if the software requirements for an input devices evaluation test appeared too great to offer any reasonable expectation of completing the study in May, we planned to design a test based on existing system programs. For example, we could have developed experimental conditions for observing and recording how one or more individuals working at separate input consoles use the "editor" or "debugging" system programs to solve simulated problems. Our objectives in this test would be to identify the compatibility of the program(s) with the user's modes of thought, any particular difficulties or shortcomings that the users experienced in using the program(s), but particularly we would hope to develop a better understanding of how users were working with the computer and what objective measures of their performances were sensitive to any problems or difficulties they reported.

In May, we planned to complete the analysis and report of the color identification test, and conduct at least preliminary trials of a performance effectiveness test. By adhering to this schedule we expected to be able to deliver a draft Final Report to NASA by June 30 in compliance with the terms of the contract.

Such simple performance tests as these would be meaningful and useful only if they were regarded as essential first steps in developing a more comprehensive and valid set of tests for evaluating the effectiveness of the system's interactive capability. By preparing tests such as these we expected to learn a great deal about the system and contribute to the growth of the system's software. These short performance tests would represent simple tasks from the point of view of a potential user of the system. Nevertheless, they would relate to some basic software development problems involving control programs, display generator programs, input device support programs, language considerations and to some degree the development of data structures. By conducting such short tests, we also expected to start experimenting with the techniques of performance measurement that could be used in later and more realistic tests. A program of realistic tests of the effectiveness of the system's interactive capabilities could be conducted only as quickly as those capabilities were themselves developed, and as we learned to design appropriate and sensitive performance measurement procedures.

Long Term Research Program

The research effort planned for the first half of 1969 would have increased in value if it were followed by a series of user performance tests involving more complex tasks and increasingly sophisticated programming support for the system's interactive graphic capabilities. In a continuing program of performance tests, we planned to compare the effectiveness of users interacting with the computer by means of the CRT displays and two or more different input devices to perform a range of representative user tasks. The tasks would have been selected from each of the three principal categories of system applications identified in the literature survey: namely, information storage and retrieval, scientific (algebraic) computations and engineering design application. In addition, we would have planned tasks that exercised the system's interactive capabilities in both a character only, and a graphic mode of operation.

During all tests of the effectiveness of man-computer interactions, we would have collected data bearing on the fluency of the interactive process. Some of the aspects of man-computer interactions that have been reported as being subjectively important to a user are the computer's response time and response time variability, the length of messages exchanged, the consistency of the frequency or duration of exchanges and the receipt of acknowledgments from the computer when a user enters a message, or status information when he is awaiting a response. Some of these aspects of an interaction between man and computer become particularly significant when a system is operating in a time-sharing mode. Arbitrary or unpredictable time constraints on the interactive process may then occur as a result of the total "work load" on the system. However, for planning purposes we assumed that the full interactive capabilities required for the first three or four "realistic" performance tests over the next twelve to eighteen months would be available only in a dedicated mode of system operation. But after the system's interactive graphic capabilities had been developed and evaluated in a dedicated mode, it would be logical and desirable to extend the program to test the capabilities and effects on human performance of operating the system on a time-sharing basis. The following general ideas about possible performance tests and the tentative schedule that was outlined were offered as stimuli to further discussion with the system designers about directions of growth for the system and concepts of the characteristics of potential user populations.

An extension of the initial limited program to include tests demanding progressively more interactive and graphic capabilities could have been composed from tasks or simulated system applications such as the following.

1. An information storage and retrieval task resembling an automated library search problem. In its simplest form this task could be developed so that it would use only the system's alphanumeric display capability. Users could be provided with tutorial-type and menu-type displays to help them search prepared files for specific information. Display requests or other messages would be entered on separate trials with a keyboard and light pen. Possibly more complex versions of the test could be developed later which might include a graphic or color capability. A user might, for example, request to see data presented in several formats' e.g., test, matrix, scalar or diagramatic, or the location and type of data might be identified by the use of color codes (if a character generator were developed for the color CRT).

2. A computational task or series of tasks involving the use of algebraic equations, reference to tables of data or charts of parametric values and dependent variables and the selection of alternative operations at a series of choice points. Again, the complexity of the problem and the range of possible interactions could be varied. Initially, the test might compare the use of a keyboard and a light pen to solve a problem involving only alphanumerically encoded data. Subsequently, a graphic display capability could be included, as well as a graphic tablet as a third input device.

3. An engineering design task involving information on the functional, temporal, and spational relationships of some construction that could be encoded graphically in two or three dimensional displays. More specifically,

we are thinking of a task involving a variety of design activities that might include the preparation and/or use of block diagrams, circuit diagrams, wiring plans, engineering drawings and perspective views. The scope of the test and the complexity of the design activities required could be varied and might be divided into at least two separate tests. The first would be the simpler and might involve only two dimensional figures, as, for example, block diagrams, circuit diagrams, schematic floor plans or equipment layout drawings. A later test might require the use of more detailed line drawings, and the use of perspective views from a variety of angles. The input devices used in these tests of the value of graphics in the performance of design tasks would be the light pen and graphic tablets. More than one test in each of the three categories of application would be desirable in order to develop the system's interactive capabilities incrementally, and to test and revise the experimental and measurement procedures as necessary.

Following completion of the separate tests in the three application categories we might have planned to evaluate the effect of running two or more of these tasks simultaneously on a time-sharing basis during a single test. By selecting particular pairs or groups of tasks for several users to work on simultaneously it might have been possible to create a graded series of combined tasks in which the computer system became progressively over loaded. Our interest in creating this situation would have been to observe what effects it had on how well users could continue to perform their tasks with varying levels of system degradation.

Until we had had experience in preparing and conducting user performance tests and detailed joint planning sessions with the system designers, it was impossible to estimate with a high degree of confidence the probable duration of such a research program. But if two tests were conducted in each application category they might together require about five to six months to complete. Consequently, the long range program outlined above was estimated to have a probable duration of from about 18 to 24 months. This tentative schedule of tests for both the short and long term programs is summarized in Table V. However, mainly because of limited access to the system, equipment failures and the closing of the Laboratory on two occasions because of heavy snow storms, it was not possible to conduct test runs of the "color test" and collect reliable data from six observers until late March and early April. Additional test runs in the later half of April were cancelled because of scheduling problems and the imminent removal of the system's Display Generator to the manufacturer for modification.

With the temporary loss of the Display Generator for the month of May, no further performance tests were possible. Instead, a preliminary analysis was made of the human factors problems of debugging computer programs in an attempt to map their general dimensions and develop a program for research on specific problems.

TABLE V

TENTATIVE SCHEDULE OF SHORT AND LONG TERM TEST PROGRAMS

Year	Jan.	Feb.	Mar.	Apl.	May	June	July	Aug.	Sept.	Oct.	Nov.	Dec.
1969	Prepare for Color Test	Pilot Color Tests (Expt. 1 and 2)	Program Color Test Expt. 3. --- Develop Requirements for Performance Test	Color Test --- Prepare Performance Test	Input Devices Tests; or Editor/Debugging Program Tests ------ Prepare Draft of Final Report		Final Report	Prepare for Info. Storage/ Retrieval Tests		Info. Storage/ Retrieval Test (1)	Prepare for Info. Storage/ Retrieval Test (2)	
1970	Info. Storage/ Retrieval Test (2)	Prepare for Computation Tests		Computation Test (1)	Prepare for Computation Test (2)		Computation Test (2)	Prepare for Engineering Design Tests		Design Test (1)	Prepare for Engineering Design Test (2)	
1971	Design Test (2)	Prepare for Time-Shared Test		Time-Shared Test		→	→	→	→			

AN EXPERIMENTAL TEST OF COLORS ON THE ITT CRT DISPLAY

This test was planned in response to a direct request from NASA engineers for an evaluation of the colors generated on the ITT color scope by the Wolf Associates color coordinate transformation programming scheme. The test (experiment) investigated whether a selected sample of colors generated on the Color CRT could be uniquely identified. In addition, the experiment sought to determine along what dimensions (i. e., hue, saturation, and intensity) difficulty in making absolute color judgments occur.

The Wolf Associates' color coordinate transformation programming system is represented schematically by three planes through an inverted "color cone," in which altitude varies with intensity and radius varies with saturation (see Figure 6). The circular sections of the cone represent three levels of stimulus intensity, and the concentric circles of each level represent degrees of saturation. The cells within the circles represent different colors, differing in hue, saturation, and intensity. The complete schema represents 64 colors which can be generated theoretically from the various possible combinations of the three primaries each varying in intensity. The number code used by Wolf Associates consists of a set of three digits; hue is encoded by digit position, and intensity by digit value (0 to 3).

General Test Design

The test was planned to consist of three experiments. Experiment I, made up of three categories, investigates whether an observer can identify the sample of generated colors for hue. The color stimuli of Category Ia, Experiment I, vary according to the Wolf model in hue and saturation. The color stimuli of Category Ib vary in hue and intensity and the stimuli of Category Ic vary in hue, saturation, and intensity.

Experiment II, also made up of three categories utilizes the same colors as Experiment I. Experiment II investigates whether an observer can identify the sample of colors for both hue and their degree of saturation and intensity. The observer is instructed to identify the color's hue in all three categories, along with rating the degree of saturation in Category IIa, and the degree of intensity in Category IIb. The observer is instructed to identify the color stimulus for hue along with rating the degrees of saturation and intensity in Category IIc.

Experiment III seeks to determine the fineness of absolute discrimination for a sample of generated colors varying in hue, saturation, and intensity.

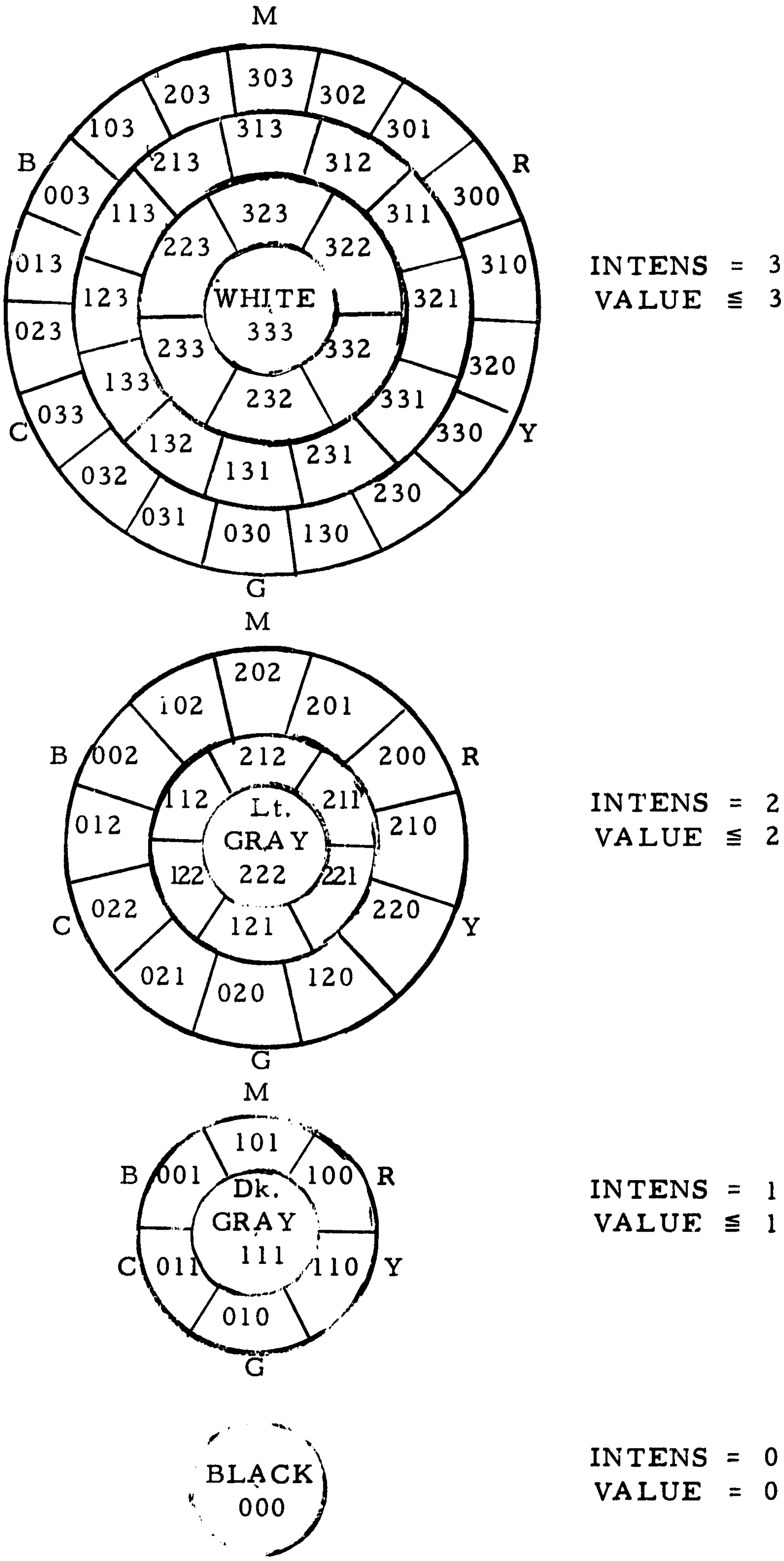


Figure 6. Wolf Schema

This stimulus sample would be selected on the basis of the analyses of Experiments I and II.

Method

Apparatus and display conditions.--A DPP 516 computer system controlled and sequenced the selected sample of colors shown on the ITT color CRT. The form of the color stimuli was a one (1) inch diameter filled circle. Each stimulus was presented at the center of the CRT screen.

The stimuli luminances are dependent on the calibration of the three intensity guns of the primary colors (i. e., red, blue, and green).

The calibration of the intensity guns was determined by the ERC engineers. Their procedure for calibrating involved the use of two calibration displays. The first presented twelve one (1) inch filled circles arranged in four columns, each consisting of three circles. Each column displayed a primary hue at three levels of intensity and the fourth column displayed the combinations of the three primaries. With this display before them, they adjusted the intensity of the red and blue guns to a setting just before the point where the green column displayed erratic characteristics (i. e., thin loops of additional green color appeared on the scope surface).

The second calibration display was then used which presented a "rainbow" display of spectral colors. The intensity controls for the red and blue guns were then adjusted to create the best spectrum of hues. All three intensity controls were then considered calibrated and taped over to protect against tampering.

Stimuli luminances were measured with a Gamma Scientific Photometer, model 700-2, with photopic filter and a photometric Telescope (700-2). The size of the telescope aperture was .030" - 1/2 degree. The distance between the telescope and screen was 6-1/2'. The diameter of the color stimulus being measured was approximately 3/4". The recorded stimuli luminances are listed in Table VI.

The ceiling luminaries of the test room were turned off, and an average level of indirect ambient illumination of 20 foot-candles was establishing with five moveable neon lamps. The reflectors on the lamps were angled to avoid or minimize glare on the face of the color CRT. Light levels were measured with a Gamma Scientific Photometer, model 700-2, with an incident cosine receptor head (700-4). The photometer was calibrated with a Gamma Scientific 202 Calibrator.

TABLE VI
STIMULI LUMINANCES

Stimulus	Luminance (foot-Lambert)	Stimulus	(ft.-L)	Stimulus	(ft.-L)	Stimulus	(ft.-L)	Stimulus	(ft.-L)	Stimulus	(ft.-L)	Stimulus	(ft.-L)
303	15	300	9.5	330	36.5	030	28	033	36	003	6.5		
313	26	311	27.5	331	42	131	37	133	36.5	113	22	333	36
3[illegible]3	33	322	35	332	41.5	232	37.5	233	38	223	32.5		
202	12	200	6.5	220	30	020	26.5	022	29	002	7		
212	28	211	[illegible]	221	30	121	29	122	28	112	23	222	31
101	9	100	4	110	21	110	9	011	24	001	6.5	111	26

The reflected ambient illuminance and the internally produced background luminance were not of sufficient magnitude to be detected with the Gamma Scientific Photometer.

The use of a monochrome CRT for listing the choices of response and transmitting the subject's responses to the central processor as mentioned in Memorandum 317-5 was disregarded due to programming difficulties and time limitations for solving these difficulties. A tape recorder was used to record the subject's responses and these responses were later manually transcribed.

Procedure.--Each subject reported to the laboratory 15 minutes before testing and remained in the test room for the purpose of adapting to the controlled level of illumination. During the adaptation period each subject was seated in front of the ITT CRT and the chair was adjusted for a 28" viewing distance.

During the adaptation, the meanings of hue, saturation, and brightness were explained. Each subject was then told the choices for hue responses and the methods of rating saturation and brightness. The subject was instructed to respond by speaking into the microphone which was held by the subject during the test.

The subject was then shown the primary colors plus their first order intermediaries at intensity Level 2, maximumly saturated. Light gray was also shown. The experimenter identified the hue along with rating the saturation and intensity for these stimuli. The purpose of this presentation was to acquaint the subject with the range of stimuli and responses to be used.

Each subject was then asked if he had any questions, and if ready, the test followed.

Only one color stimulus was presented on the CRT at a time for a duration of five seconds. The screen was then blanked. The subject made his response to the stimulus while it was presented or while the screen was blanked. A 10-second blank interval between stimuli was scheduled.

Two different experiments were administered to each subject. Each experiment had a number of categories. The color generation program automatically halted after each category of an experiment and after each experiment. There was a one to two minute rest period between the categories of each experiment, and a three to five minute rest between the experiments.

The experiments and their categories were as follows.

Experiment 1

Category (Ia)

Stimuli - The primary colors, Red (R), Green (G), and Blue (B) and their first order intermediates (300, 330, 030, 033, 003, 303), varying in saturation and with intensity held constant (Intens=3). See Figure 7.

The stimuli (19) labeled per Wolf schema were:

	300	330	030	033	003	202
	311	331	131	133	113	313
333	322	332	232	233	223	323

Response - The possible responses for the subject upon presentation of a stimulus were a forced choice among six hues plus gray (i.e., red, yellow, green, cyan, blue, magenta, and white).

Experiment I

Category Ia

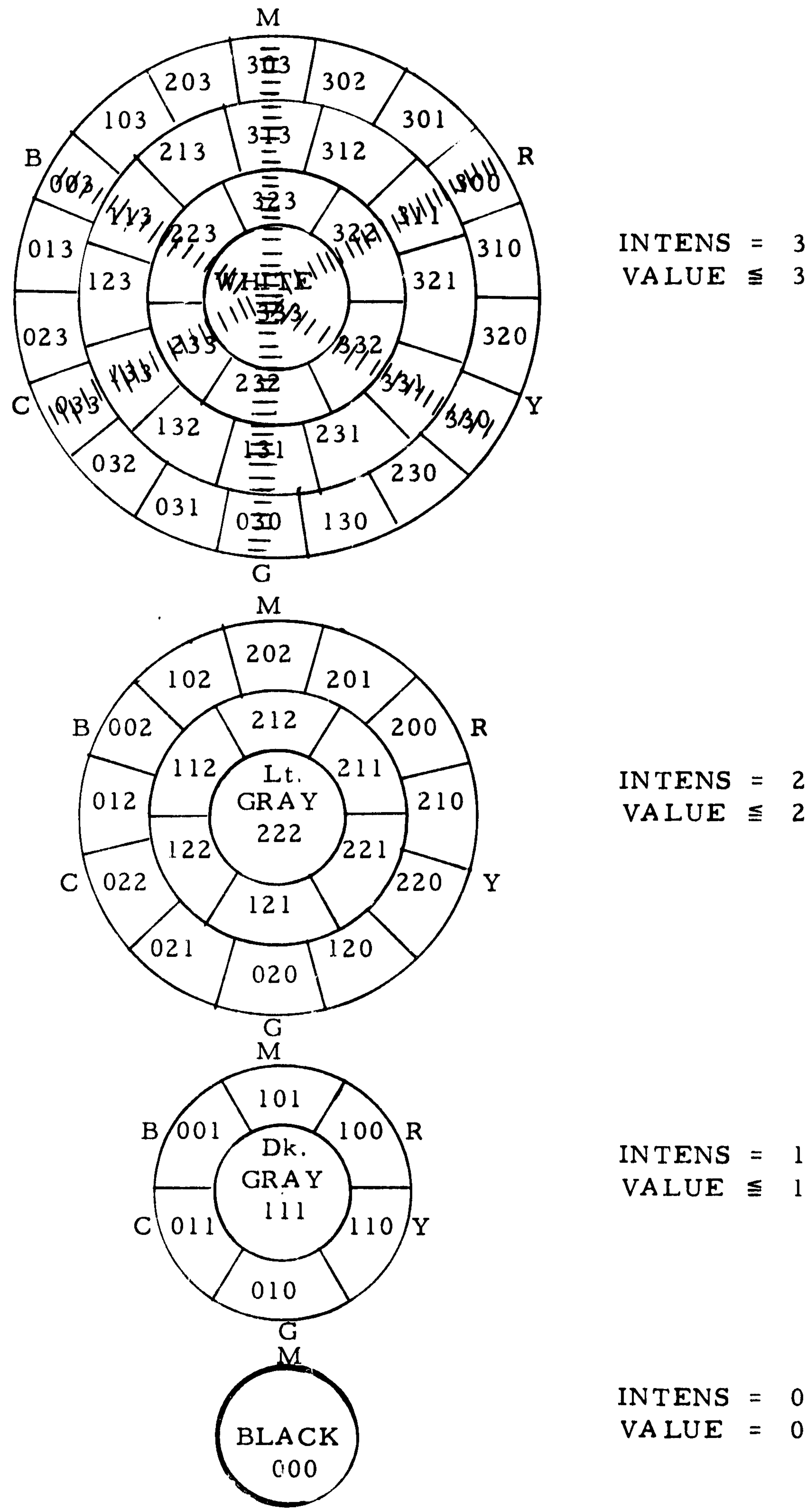


Figure 7. Shaded Colors were the Selected Color Stimuli

Category (Ib)

Stimuli - Primary colors plus their first (1st) order intermediates varying in intensity but constant saturation (see Figure 8). The stimuli, labeled per Wolf schema, were:

333	303	300	330	030	033	003
222	202	200	220	020	022	002
111	101	100	110	010	011	001

Response - Same as Category Ia.

Category (Ic)

Stimuli - Primary colors and first (1st) order intermediates varying in saturation and intensity (see Figure 9). The stimuli, labeled per Wolf schema, were:

303	300	330	030	033	003
313	311	331	131	133	113
323	322	332	232	233	223
		333			
202	200	220	020	022	002
212	211	221	121	122	112
		222			
101	100	110	010	011	001
		111			

Response - Same as Cateogry Ia.

Experiment I
Category Ib

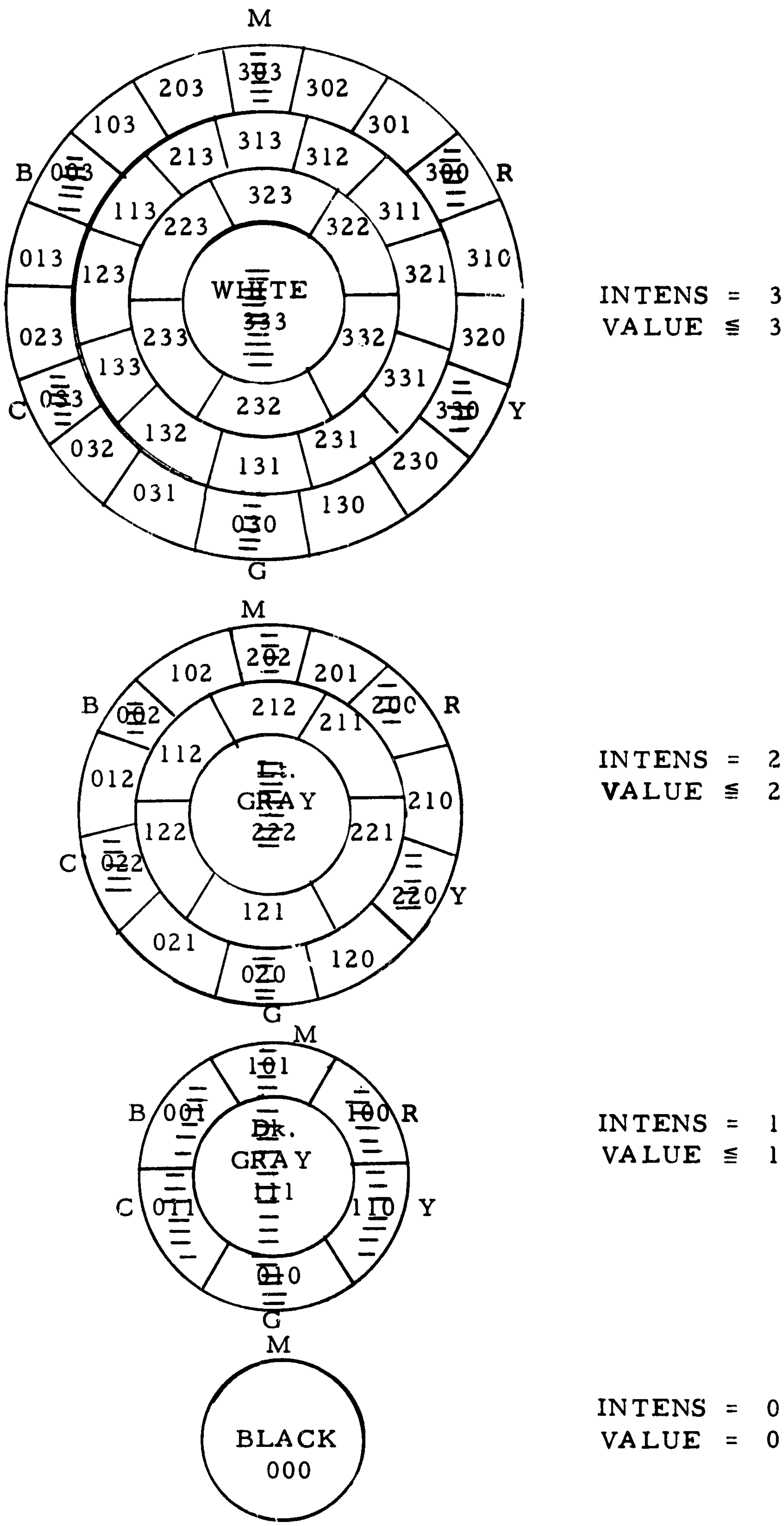
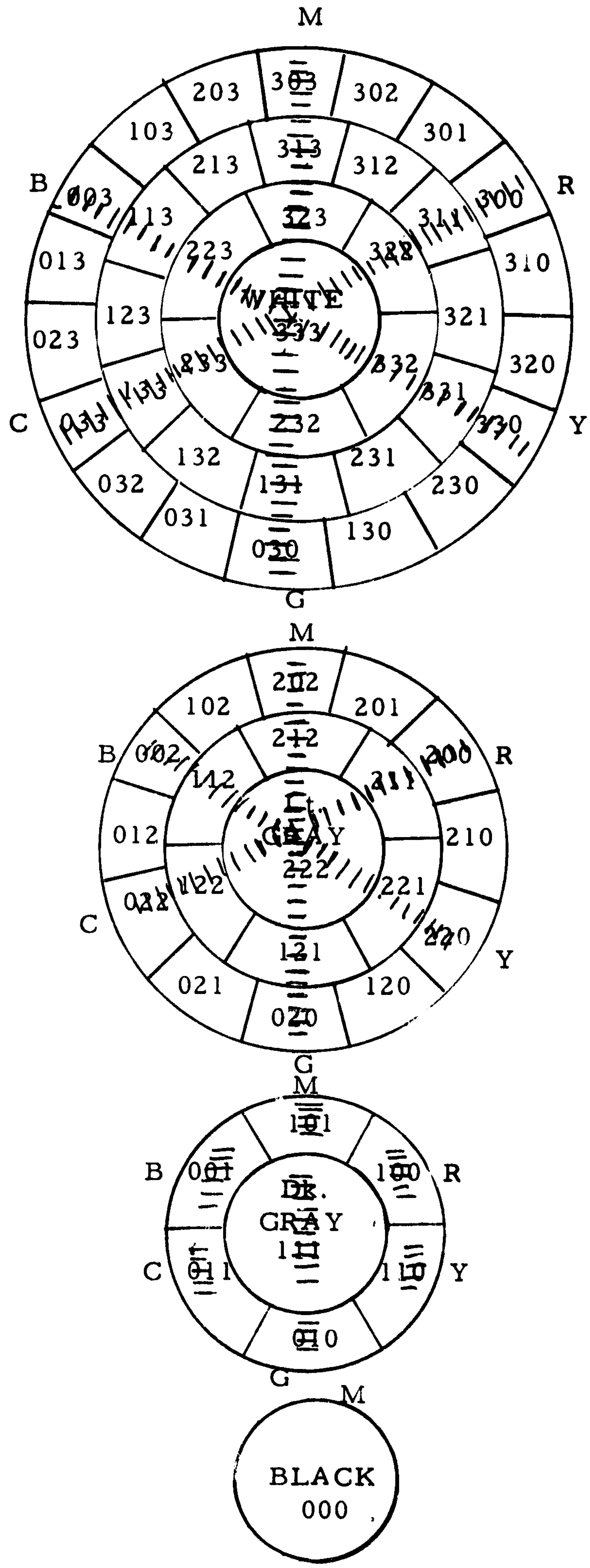


Figure 8. Shaded Colors were the Selected Color Stimuli

Experiment I
Category Ic



INTENS = 3
VALUE ≦ 3

INTENS = 2
VALUE ≦ 2

INTENS = 1
VALUE ≦ 1

INTENS = 0
VALUE = 0

Figure 9. Shaded Colors were the Selected Color Stimuli

Experiment II

Category (IIa)

Stimuli - (see Figure 10) Primary colors and first (1st) order intermediates varying in saturation and with intensity constant (Itens = 3). The stimuli labeled per Wolf schema, were:

300	330	030	033	003	303
311	331	131	133	113	313
322	332	232	233	223	323
		333			

Response - The possible responses for the subject upon presentation of a stimulus were a forced choice among six hues plus gray (i.e., red, yellow, green, cyan, blue, magenta, and gray) along with rating the degree of saturation (as 1, 2 and 3).

Category (IIb)

Stimuli - Primary colors and first (1st) order intermediates varying in intensity and with saturation constant (see Figure 11). The stimuli, labeled per Wolf schema, were:

333	303	300	330	030	033	003
222	202	200	220	020	022	002
111	101	100	110	010	011	001

Response - The possible responses for the subject upon presentation of a stimulus were a forced choice among six hues plus gray (i.e., red, yellow, green, cyan, blue, magenta, and gray) along with rating the degree of intensity (as 1, 2, and 3).

Experiment II

Category IIa

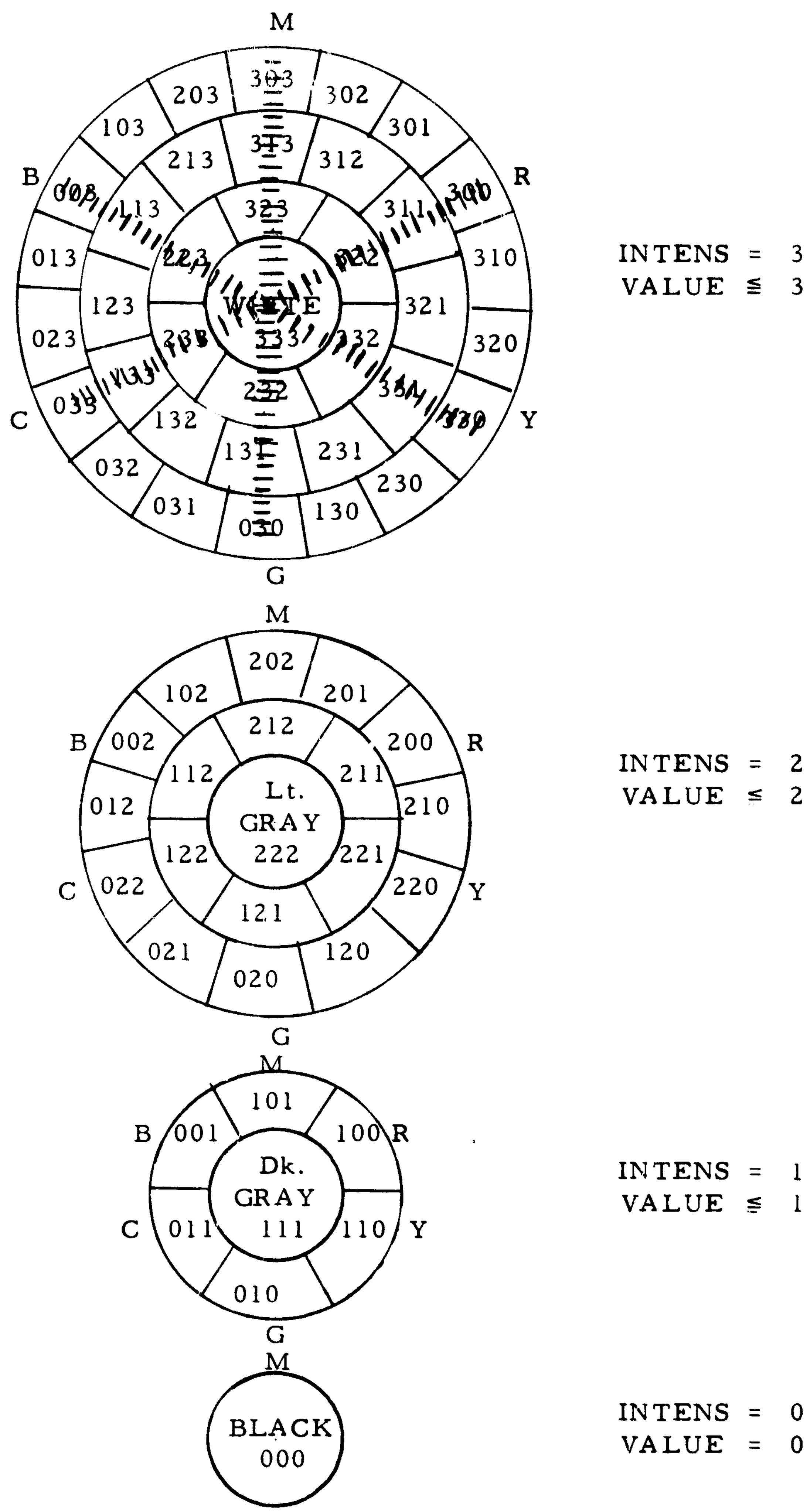


Figure 10. Shaded Colors were the Selected Stimuli

Experiment II

Category IIb

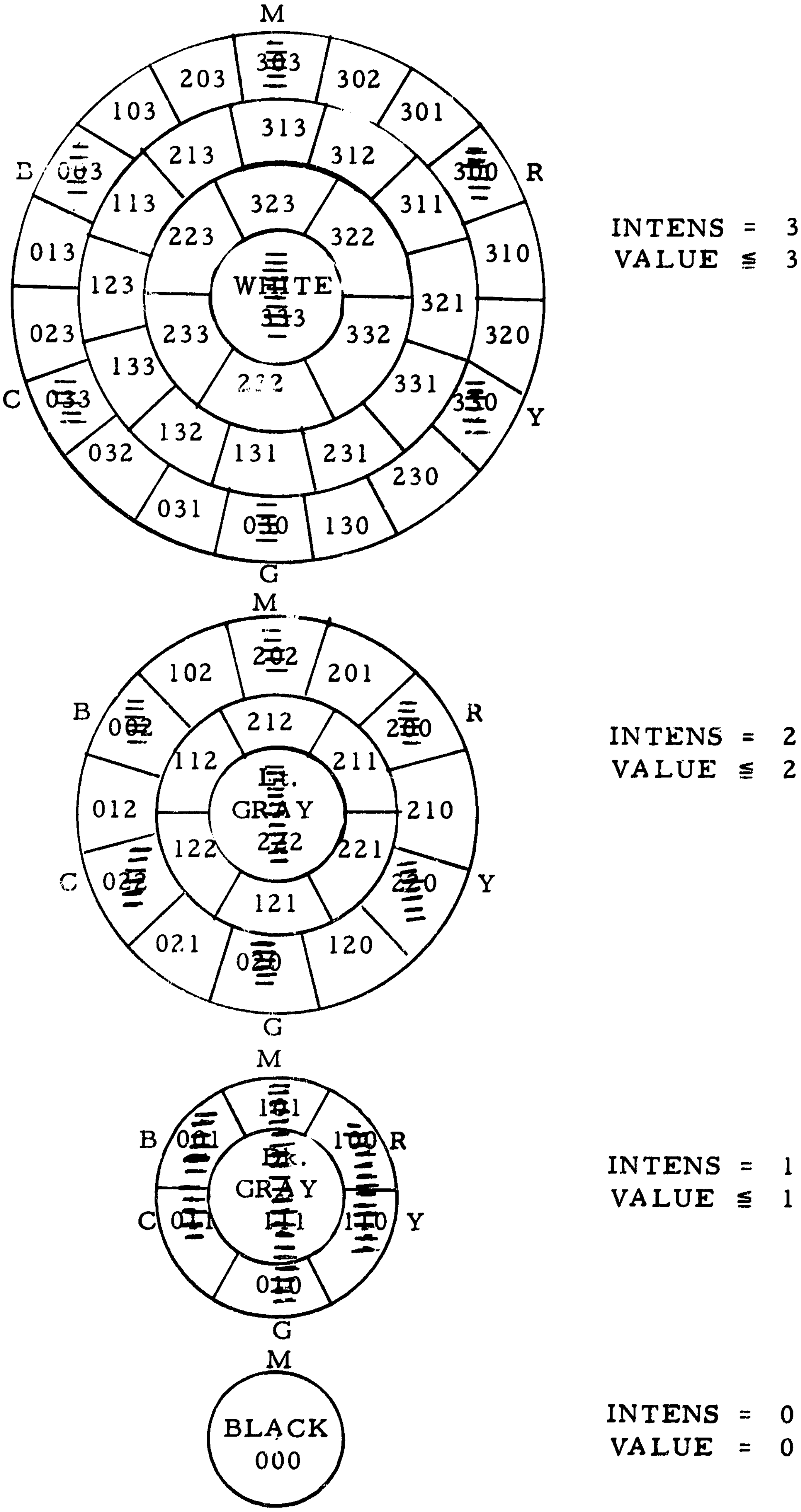


Figure 11. Shaded Colors were the Selected Color Stimuli

Category (IIc)

Stimuli - Primary colors and first (1st) order intermediates varying in saturation and intensity (See Figure 12). The stimuli, labeled per Wolf Schema, were:

303	300	330	030	033	003	
313	311	331	131	133	113	333
323	322	332	232	233	223	
202	200	220	020	022	002	222
212	211	221	121	122	112	
101	100	110	010	011	001	111

Response - The possible responses for the subject upon presentation of a stimulus were a forced choice among six hues plus gray (i.e., red, yellow, green, cyan, blue, magenta, and white) along with rating the degree of saturation (as 1, 2, and 3) and rating the degree of intensity (as 1, 2, and 3).

Subjects.--The test program used six male subjects. All had normal or corrected to normal visual acuity. All were tested with the Dvorine Color Discrimination Test and were found to be normal.

Design.--The order of presentation of Experiments I and II were counterbalanced among the subjects. The first two categories of each experiment were also counterbalanced, and always preceded category three of Experiments I and II. The stimuli of each category were randomized. Two sets of randomized stimuli were drawn up with three subjects receiving the first list, the other three subjects receiving the second list.

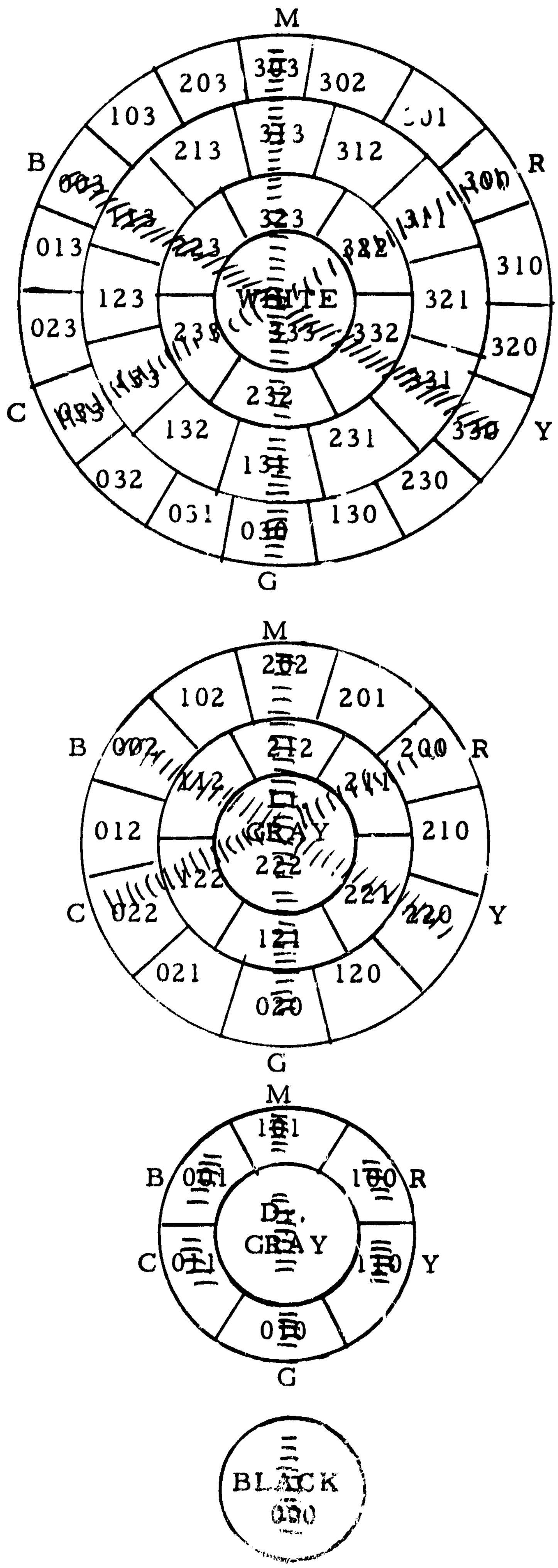
Each subject performed Experiments I and II during one experimental session which lasted for approximately one and one-half hours.

The subjects performances in identifying the colors were measured in terms of the frequency with which color was identified correctly.

Results.--The experimental results are grouped into two categories for discussion. Category I includes the results from the first four (4) subjects;

Experiment II

Category IIc



INTENS = 3
VALUE ≤ 3

INTENS = 2
VALUE ≤ 2

INTENS = 1
VALUE ≤ 1

INTENS = 0
VALUE = 0

Figure 12. Shaded Colors were the Selected Color Stimuli

Category II combines the results of all six (6) subjects. The data was grouped in this way because of possible differences in stimulus conditions. After the first four (4) subjects were tested, the CRT failed and was repaired. However, there is some doubt that the CRT calibration setting and stimulus conditions for the last two subjects after the failure were exactly the same as for the first four (4) subjects.

Tables VII (a and b) and VIII (a and b) tabulate the frequency and percentage that a stimulus was identified as a particular hue by the first four (4) subjects, and by all six (6) subjects, respectively.

Those stimuli that were consistently identified as a particular hue for 75% and 90% of the observations are posted on Figures 13, 14, 15, and 16 to bring these results into the perspective of the Wolf model.

Table IX is the stimulus-response matrix tabulating the frequency of responses of both hue and intensity rating. Only responses that have correct hue identification have been included in this matrix. Figure 17, the graph of the physical intensities for the maximumly saturated primary and first order intermediate colors, is included for discussion of the relationship between the physical intensities of the colors and the subjects' intensity ratings.

<u>Discussion.</u> --Tables VII (a and b) and VIII (a and b), show that absolute hue discrimination is possible only at maximum saturation for all levels of intensity. The primary hues (red, green and blue) at maximum saturation were consistently identified correctly for more than 90% of the observations for all three levels of intensity. This was the result for all six subjects.

The responses of the first four (4) subjects show the first order intermediary colors of magenta and yellow at maximum saturation to be consistently identified 90% of the time for all levels of intensity. Cyan failed to be consistently identified at intensity Level 1.

The last two subjects performance differed in that they failed to discriminate yellow from green at intensity Level 1.

Since it was found that there was no hue discrimination for desaturated colors, it was needless to analyze the results for saturation discriminations.

Intensity discrimination was analyzed only for the fully saturated colors since hue discrimination only existed at that level of saturation. Table IX illustrates that only for red is intensity discrimination demonstrated. Even in the case of red, there is much uncertainty. Failure to discriminate

TABLE VIIa
THE FREQUENCY AND PERCENTAGE A STIMULUS WAS IDENTIFIED AS A PARTICULAR HUE (n=4)

STIMULUS		RESPONSE							
Hue	Wolf Code	Red Frequency/ Percent	Yellow Frequency/ Percent	Green Frequency/ Percent	Cyan Frequency/ Percent	Blue Frequency/ Percent	Magenta Frequency/ Percent	White-Gray Frequency/ Percent	Freq. Total
R	100	16/100							16
Y	110		15/94	1/6					16
G	010			16/100					16
C	011				8/44	10/56			18
B	001					14/100			14
M	101				6/38		10/62		16
W-Gr	111				13/81			3/19	16
R	200	18/100							18
Y	220		17/100						17
G	020			18/100					18
C	022				15/88	2/12			17
B	002					18/100			18
M	202						18/100		18
R	211				2/25		4/50	2/25	8
Y	221				3/33			6/67	9
G	121				7/88	1/12			8
C	122				8/100				8
B	112				6/75	1/13	1/12		8
M	212						6/75	2/25	8
W-Gr	222				2/12		5/29	10/59	17

TABLE VIIb

THE FREQUENCY AND PERCENTAGE A STIMULUS WAS IDENTIFIED AS A PARTICULAR HUE (n=4)

STIMULUS		RESPONSE							
Hue	Wolf Code	Red Frequency/ Percent	Yellow Frequency/ Percent	Green Frequency/ Percent	Cyan Frequency/ Percent	Blue Frequency/ Percent	Magenta Frequency/ Percent	White-Gray Frequency/ Percent	Freq. Total
R	300	24/100							24
Y	330		24/100						24
G	030			21/100					21
C	033				22/92	2/8			24
B	003					24/100			24
M	303				1/5		20/95		21
R	311						14/88	2/12	16
Y	331						4/27	11/73	15
G	131				12/86	1/7		1/7	14
C	133				13/81	1/6	1/6	1/7	16
B	113			1/6	11/69	1/6	2/12	1/6	16
M	313				1/7		10/71	3/22	14
R	322						7/44	9/56	16
Y	332						3/23	10/77	13
G	232				4/25		1/19	9/56	14
C	233				5/31	1/6	1/16	9/57	16
B	223				4/29			10/71	14
M	323				9/69			4/31	13
W-Gr	333				1/4		7/29	16/67	24

TABLE VIIIb
THE FREQUENCY AND PERCENTAGE A STIMULUS WAS IDENTIFIED AS A PARTICULAR HUE (n=6)

STIMULUS		RESPONSE							
Hue	Wolf Code	Red Frequency/ Percent	Yellow Frequency/ Percent	Green Frequency/ Percent	Cyan Frequency/ Percent	Blue Frequency/ Percent	Magenta Frequency/ Percent	White-Gray Frequency/ Percent	Freq. Total
R	300	35/100							35
Y	330		36/100						36
G	030			33/100					33
C	033				28/78	8/22			36
B	003				1/97	35/3			36
M	303				1/30		32/97		33
R	311				4/17		16/67	4/16	24
Y	331				3/13		6/26	14/61	23
G	131				16/73	5/23		1/4	22
C	133				17/71	5/21	1/4	1/4	24
B	113			1/4	15/63	4/17	2/8	2/8	24
M	313				1/5	3/14	14/64	4/17	22
R	322				4/17		8/33	12/50	24
Y	332				4/19		5/24	12/57	21
G	232				7/32	4/18	1/5	10/45	22
C	233				10/42	4/17	1/4	9/37	24
B	223				7/32	2/9		13/59	22
M	323				4/19		10/48	7/33	21
W-Gr	333				8/22		9/25	19/53	36

TABLE VIIIa
THE FREQUENCY AND PERCENTAGE A STIMULUS WAS IDENTIFIED AS A PARTICULAR HUE (n=6)

STIMULUS		RESPONSE							
Hue	Wolf Code	Red Frequency/ Percent	Yellow Frequency/ Percent	Green Frequency/ Percent	Cyan Frequency/ Percent	Blue Frequency/ Percent	Magenta Frequency/ Percent	White-Gray Frequency/ Percent	Freq. Total
R	100	24/100							24
Y	110		16/70	7/30					23
G	010			24/100					24
C	011				12/44	15/56			27
B	001					21/100			21
M	101				8/33	4/17	12/50		24
W-Gr	111				17/71	4/17		3/12	24
R	200	26/100							26
Y	220		24/96	1/4					25
G	020			26/100					26
C	022				18/72	7/28			25
B	002					25/100			25
M	202						26/100		26
R	211				5/42		4/33	3/25	12
Y	221				5/36	1/7		8/57	14
G	121				9/75	3/25			12
C	122				10/83	2/17			12
B	112				8/67	3/25	1/8		12
M	212				2/17		6/50	4/33	12
W-Gr	222				4/17	1/4	7/29	12/50	24

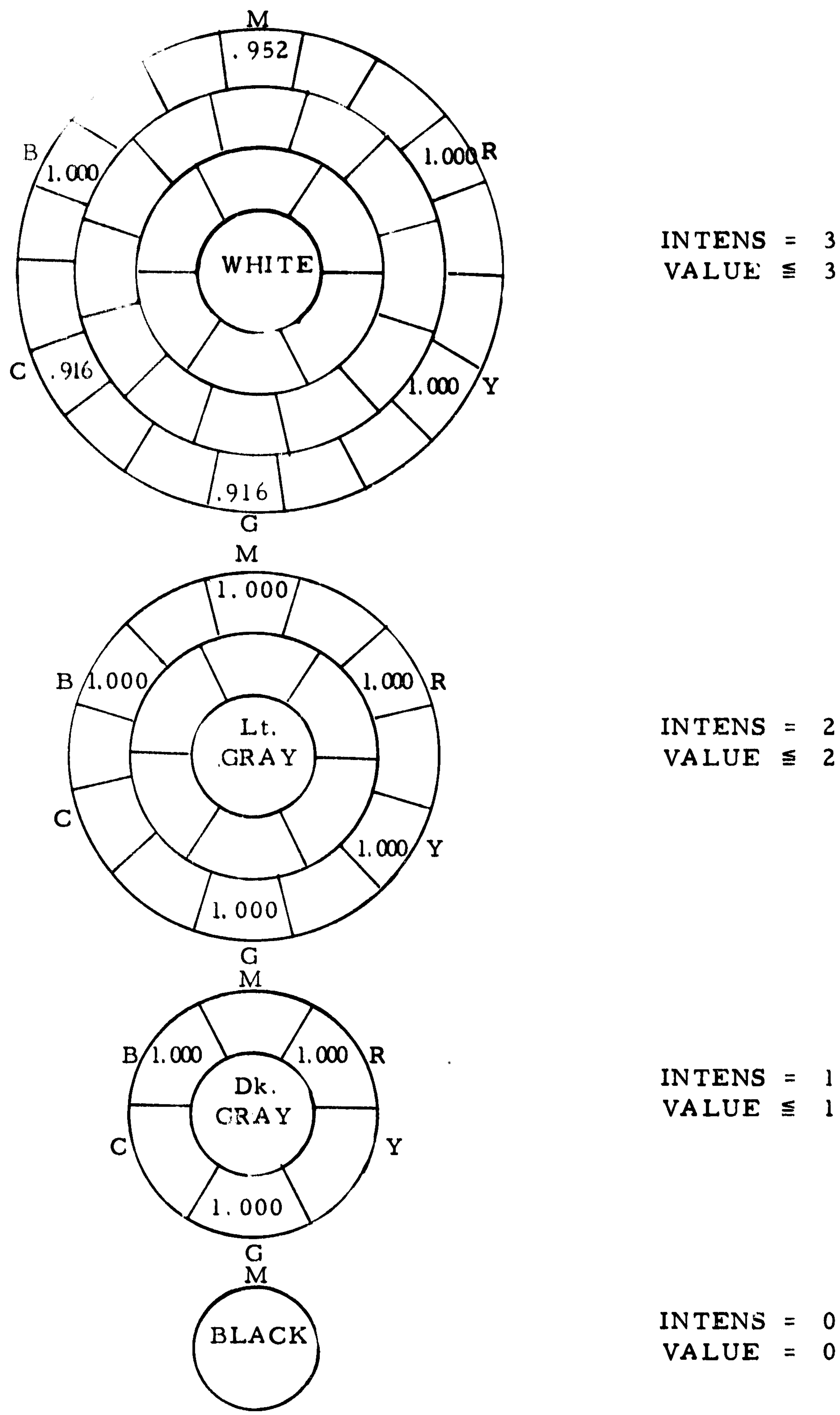


Figure 13. Pr (discriminating hue) ≥ .900
n = 4

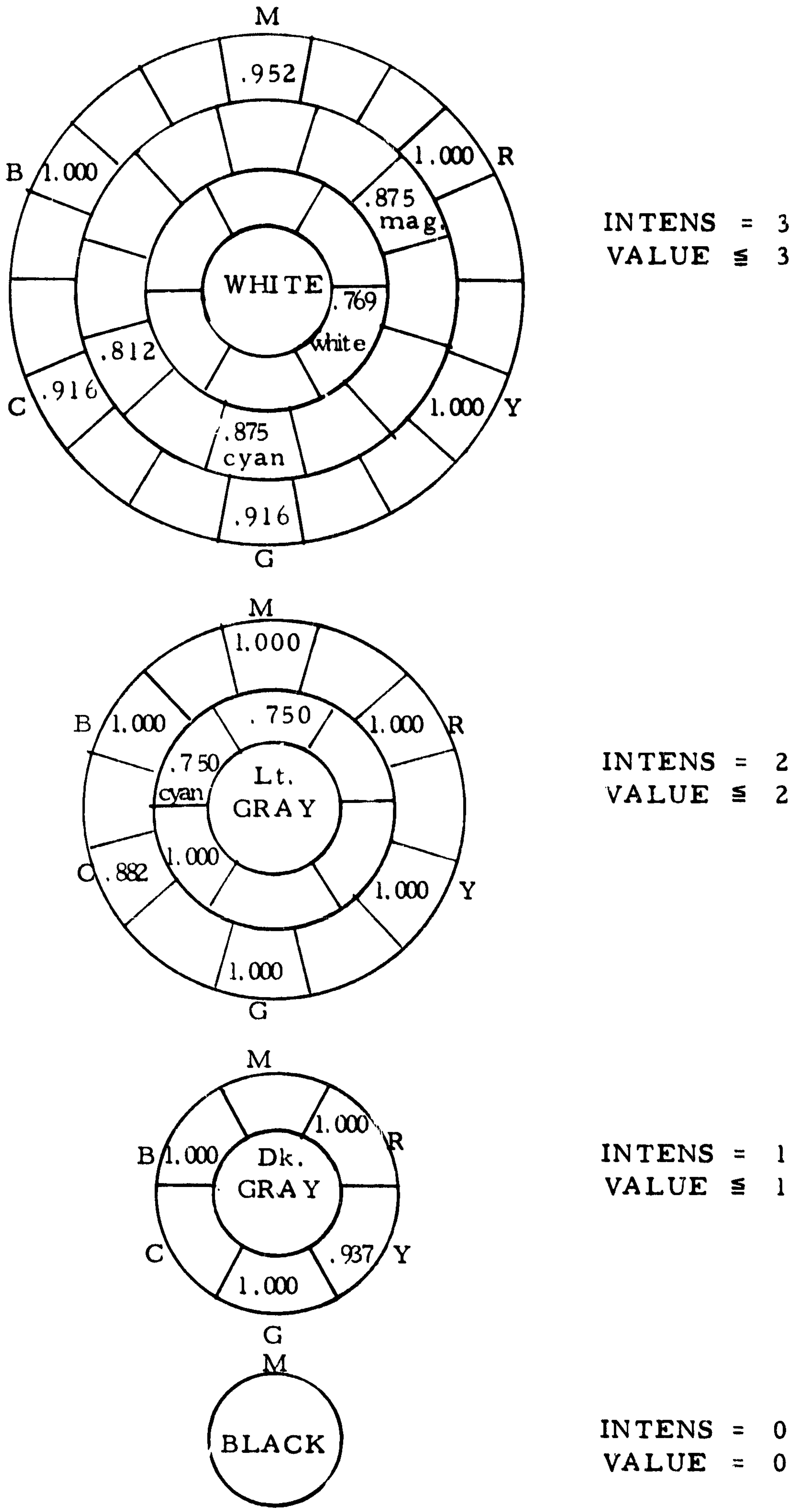


Figure 14. Pr (discriminating hue) ≥ .750
n = 4

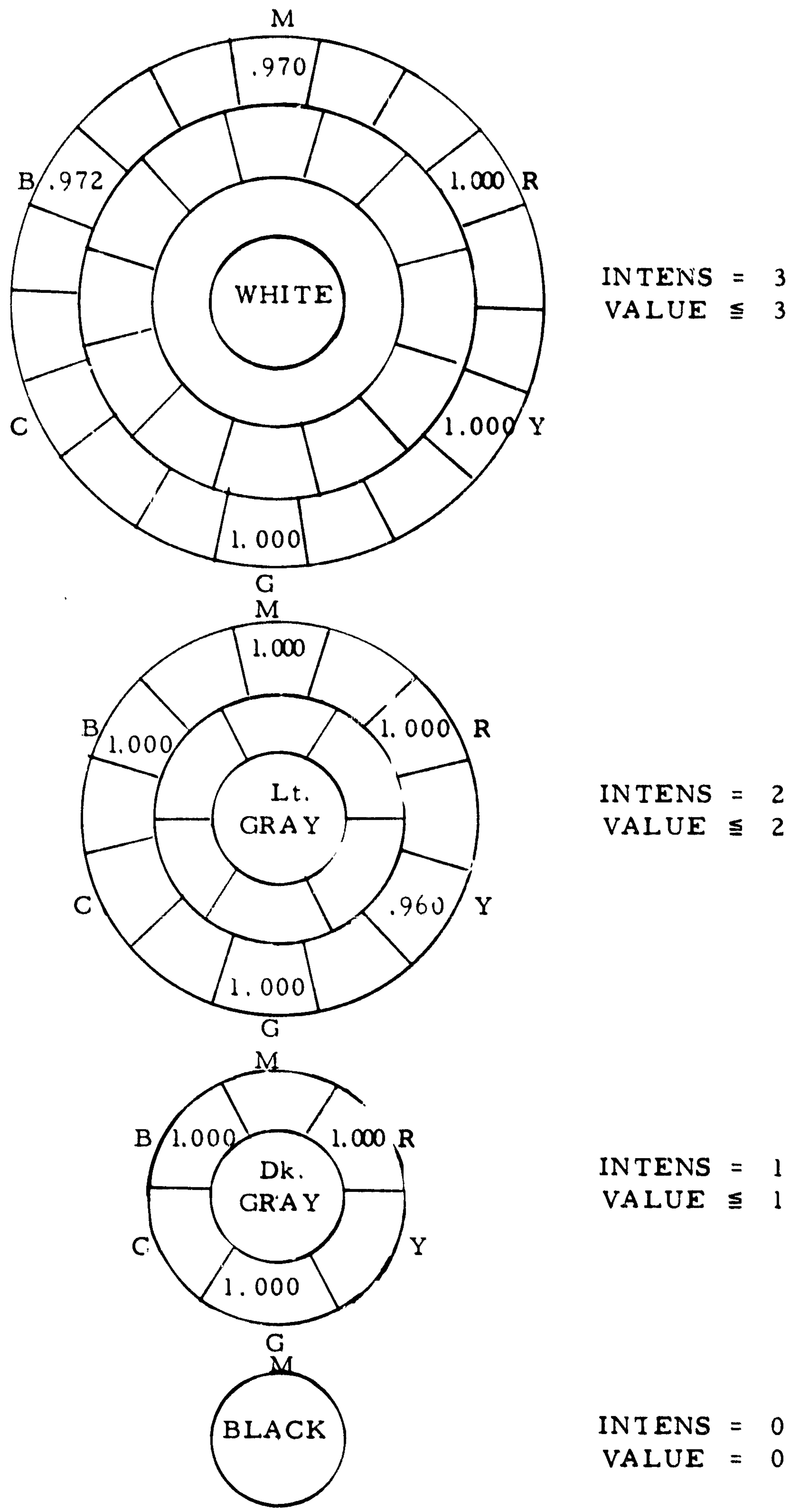
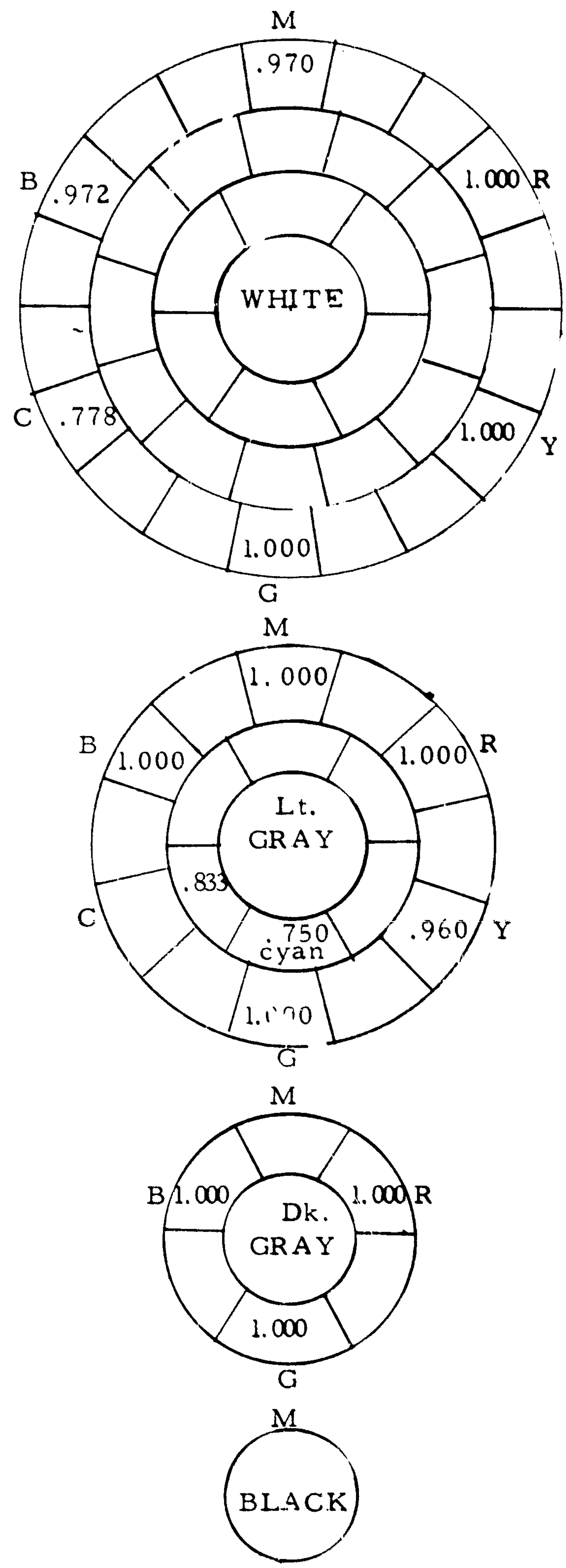


Figure 15. Pr(discriminating hue) ≥ .90
n = 6



INTENS = 3
VALUE ≤ 3

INTENS = 2
VALUE ≤ 2

INTENS = 1
VALUE ≤ 1

INTENS = 0
VALUE = 0

Figure 15. Pr (discriminating hue) ≥ .75
n = 6

TABLE IX

STIMULUS RESPONSE MATRIX FOR THE RESPONSE OF HUE AND RATING OF INTENSITY (n=6)

STIMULUS		RESPONSE																	
Hue	Wolf Code	Hue: Red			Yellow			Green			Cyan			Blue			Magenta		
		Intensity Rating: 1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
R	100	7	4	1															
R	200		7	5															
R	300	1	4	7															
Y	110				2	6	1												
Y	220					9	3												
Y	330					4	8												
G	010							1	8	2									
G	020								10	2									
G	030								3	8									
C	011										1	6							
C	022											3	6						
C	033											1	8						
B	001													1	5	3			
B	002													1	6	4			
B	003													1	6	4			
M	101																1	2	2
M	202																3	3	6
M	303																	3	7

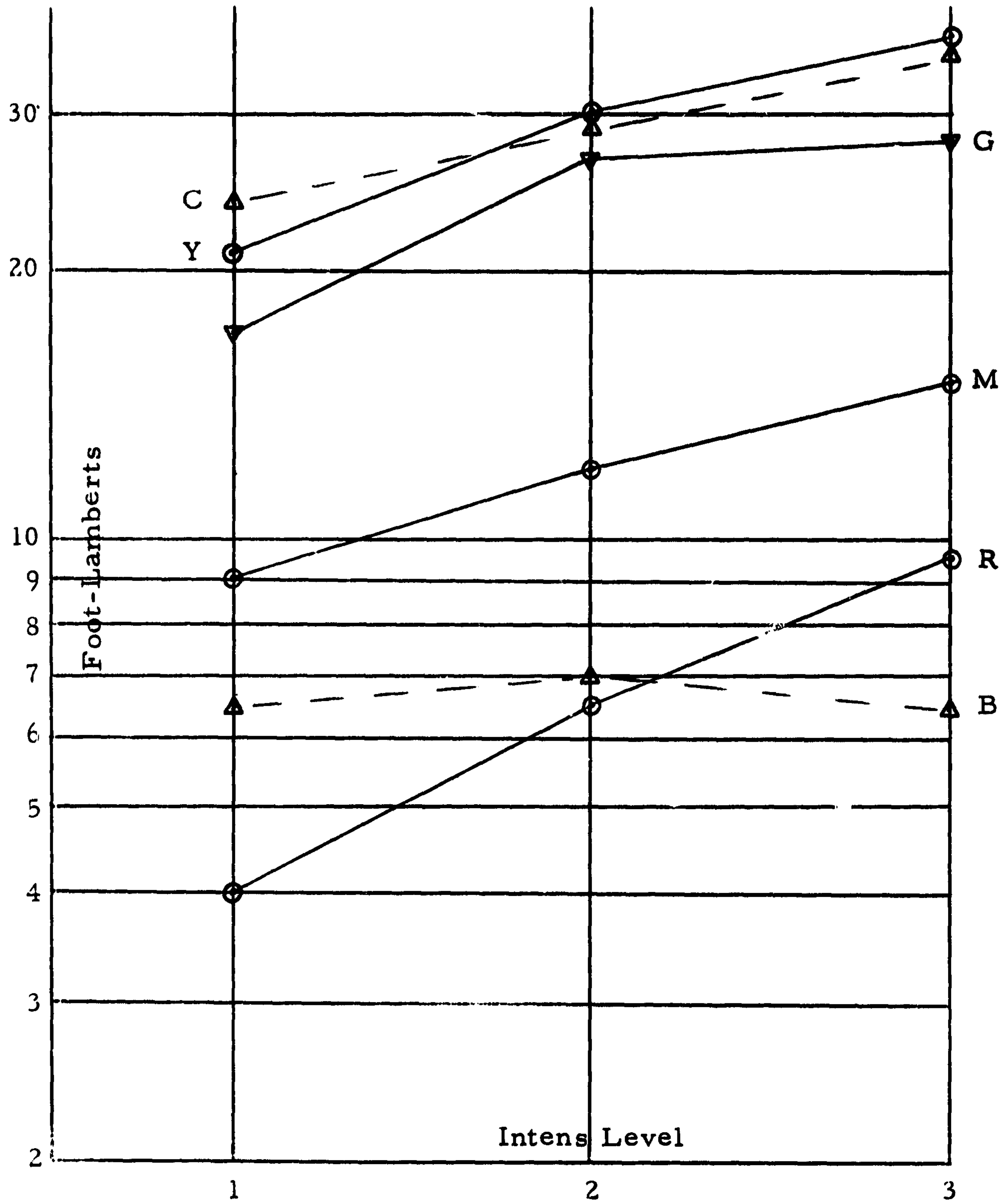


Figure 17. Intensity of Maximumly Saturated Primary and 1st Order Intermediate Colors vs. Intens Level.

maximum saturated colors along the dimension of intensity is understandable when the graph of the physical intensities is inspected. This graph shows very small increments of physical intensity from Level 1 to Level 3.

No attempt was made to perform Experiment III as had been originally planned.

With discrimination of the sample colors only along the dimension of hue, only six of the thirty-nine (39) colors sampled were uniquely identified. The results of this test do not confirm the prediction of the Wolf schema that sixty-four (64) colcrs can be generated on the color CRT from the various possible combinations of the three primaries each varying in intensity.

Reservations must be made on the adequacy of this experiment as a valid test of the Wolf schema, because of the previously mentioned restriction on the calibration of the intensities of the three color guns of the CRT, and the inadequate range of intensity control which precluded variation in the primary hues over a range of intensities required by the Wolf programming schema.

HUMAN FACTORS PROBLEMS OF PROGRAM DEBUGGING

Program design, development and installation has become a critical phase of system implementation that is hindering exploitation of the full potential of modern digital computers. Yet, surprisingly little systematic study and experimentation has been undertaken to better define the dimensions of the problem: what are the abilities and skills of "good" programmers; how do you teach program debugging; what are the probabilities of differing kinds of errors, and the difficulties of diagnosing and correcting errors in differing kinds of programs, program languages, or operating systems; and what are the relative efficiencies of on-line or off-line debugging or of available or proposed debugging aids (programs, displays and procedures). Debugging covers a variety of operations and tasks involved in detecting, isolating and identifying program faults, in preparing and making corrections or modifications, and in testing the revised program over a range of critical system conditions, problems and parametric values. The difficulty of any or all of these debugging operations is likely to vary with the type of error (logical, system specific or coding error), the language used for writing the program, the program function and the design and condition of the system environment. A variety of specific debugging aids (programs and output formats) are available to help a programmer isolate and identify software faults, but little or nothing seems available to help a programmer select an efficient procedure, guide him through a logical analytic process, and evaluate and interpret the information generated at each step in the process.

We conducted a preliminary exploration of the human factors problems of debugging and attempted to map their general dimensions and to develop an approach and program for research on specific problems. First, we reviewed and classified a sample of the debugging aids currently available to programmers. Our aim in making this survey was to identify 1) the kinds of information that programmers sought, 2) the procedures they had to follow to acquire the information, and 3) the needs that programmers perceived for improved techniques as indicated by the successive developments in debugging aids. Secondly, we analyzed the programming and debugging processes to identify a programmer's opportunities for making various types of errors, for detecting and correcting errors at successive stages of program generation and execution, and the stages at which debugging systems could aid him localize, identify and correct program faults. Thirdly, we selected a limited number of critical problems for which human factors research suggests certain improved aids or debugging techniques and outlined a program of research to develop and test these potential "solutions."

Survey of Current Debugging Aids

The method used by past surveys for classifying existing debugging systems has usually been Assembly-Language debugging and Higher-Level-Language debugging. For our purposes, this description of available debugging aids was expanded to have the following four descriptors: 1) Numerics (Bits and Bytes), 2) Instruction Mnemonics, 3) Full Symbolic Reference, and 4) Higher Level. Table X illustrates our classification of some current debugging aids using these four categories. The lists of debugging aids in each category are examplary only and by no means inclusive or complete.

The Numeric category includes all those aids in which instructions, output and modification of the program are accomplished at the numeric level (octal in most cases). In general, the functions provided by the aids in the Numeric category are:

1. type contents of register(s)
2. change contents of selected registers
3. list those registers having given contents in a given field
4. execute instructions
5. print "traces" of machine operations

Included in the category of Instruction Mnemonics are those aids in which examination and modification of a program is also accomplished at the

TABLE X

A TENTATIVE CLASSIFICATION OF SOME CURRENT DEBUGGING AIDS

NUMERICS	INSTRUCTION MNEMONICS	FULL SYMBOLIC REFERENCE	HIGHER LEVEL
TYC	TX-O D.I.U.S	FLIT	MADBUG
FIND AND CHANGE, ESD	UT-3	SUPERFLIT	BUGTRAN
TRACE, ESD	DB/BBN DEBUG	DDT	DISPLAYTRAN
EXTENDED TRACE, PDP-9	HAMBUG	DDT IMPROVED	FTN SOURCE STATEMENTS
DEBUG/ERC	OPAK	DDT EXTENDED	FTN OPER. SYS. OPTIONS
X16 DEBUG		DDT ESD	TALK
GBUG (Hexadecimal)		DDT 340	
COLOR DEBUGGER		DDT-4	
KUS		DDT-8	
		DDT-UP	
		XDDT	
		DDT-I.C.	
		DDT-BASIC	
		DDT-9	
		DDT-10	
		DDT-940	
		FAPDBG	
		TIC	
		DEBUG/AFCRL	
		ADT/AFCRL	
		DBUG/SPC	

numeric level (octal or decimal) but with instructions expressed in some form of symbolic mnemonics. The principal functions performed by mnemonic debugging aids are:

1. examine a word or block of words
2. change a word or block of words
3. conduct word searches
4. execute dumps (post mortem - snap)
5. print traces as specified
6. accept breakpoints or traces

In general, the functions and information provided by the aids in the Mnemonic category are much the same as in the Numeric category with the addition of the important function of specifying breakpoints.

The category of Full Symbolic Reference includes those debugging aids in which the user examines and modifies his program in terms of the symbols used in his source program and changes the contents of registers in a form almost identical to that used in the corresponding assembly language. The Full Symbolic Reference category provides the same functions and information as the Mnemonic category with the addition of conditional breakpoints.

The category of Higher Level debugging aids includes those debugging aids in which the user examines and modifies his program directly rather than the computer's states by means of commands and outputs in a higher level language similar to his source program language. The Higher Level category provides many functions and kinds of information similar to the Full Symbolic Reference category. Along with those functions mentioned in the other categories there are conditional terminations of the program along with an option to print the error comment of the source program. The user may incorporate debug statements in his source "deck" or accompany his source deck with separate debug statements and may command the processor to provide for debug operations.

These four categories of debugging aids can be viewed as representing a progression in methods of encoding instructions and information output from a code that is all numeric (octal and decimal) to an all symbolic code completely compatible with the source program language. The overall functions provided within each of these four categories of aids do not show a comparable progression of increased capabilities. Major increases in capability relate to the introduction of specific breakpoints in the Mnemonic category, of conditional breakpoints in the Full Symbolic category, and of associated increased degrees of freedom to edit and/or patch a program arbitrarily. The development of the High Level debugging aids does, however represent a significant change in approach to debugging. It enables a programmer (or user) to interact directly with his source language program instead of having to trace the effect of an assembly language statement on the states of machine registers and contents of storage locations.

The relative stability in the range of functions provided by the different categories of debugging aids is possibly due to the fact that in debugging assembly language programs the programmer has always needed the same kinds of information about machine states irrespective of differing system environments. The development of particular debugging aids for newer computer systems has resulted primarily in progressively easier ways of formulating requests for information and in reading or interpreting the information provided on machine states (i.e., as a result of increased compatibility between the codes of programming and debugging languages). The developments in functional capability referred to above reflect attempts to meet in part at least the programmer's need to get as "close as possible" to his program while it is being executed, and to be able to modify it or insert corrections with a minimum of effort, i.e., without having to recompile his whole program.

The development of high level debugging aids reflects the needs of a growing body of relatively unsophisticated computer users, who are capable of writing their own programs in some Problem Oriented Language, but lack the programming and computer experience needed to debug their programs in assembly level or machine language. The POL programming languages, their users' limited computer training, and the associated (usually large time-shared) computer systems represent a new and qualitatively different environment in which program "bugs" can thrive and for which new computer assisted debugging procedures needed to be designed.

In surveying available debugging aids and articles on recent developments in the art of debugging, we were impressed by the lack of discussion about procedures for employing available aids most effectively. Where are the guidelines for the tyro on how to approach his debugging problems or discussions of the relative power (bugs per unit cost) of differing diagnostic approaches or debugging aids, or analyses of the differences in the problems of debugging programs differing in size, complexity, language, operating environment, single or multiple authorship, etc.? What kinds of "bugs" are most likely to occur in different program/system/language environments? What are the relative probabilities of success for different debugging approaches--team debugging, probabilistic debugging (best guess diagnosis), incremental debugging, on-line or off-line debugging? In fact, little research has been published that analyzes the problem of human error in programming nor of detecting, diagnosing and correcting faults in computer software.

Programming Bugs, Debugging Tasks and Human Factors Problems

Programming bugs are almost always the result of human error. An awareness of this fact and an acceptance that it holds true even for oneself is probably one of the greatest aids to successful programming and program debugging. Bugs occur during programming, and skilled programmers differ from less skilled individuals not in being less fallible than other men, but rather in their early detection and correction of their mistakes. Skilled programmers are alert to errors at all stages of programming and succeed in correcting many of their mistakes before run time.

The analysis of a problem and its reformulation into related sequences of intermediate steps provides an opportunity for making errors of logic. The expression of these logically related steps in some form of flow-chart macro-statements of function or of machine operations provides a medium in which errors of logic can be detected and corrected. The next major opportunity for error comes when the flow-chart statements are translated or coded into a source program in accordance with the rules of a particular language and the additional specific requirements and constraints imposed by a particular computing system's operating environment (both hardware and software). This coding operation can give rise to two kinds of "bugs", coding or language errors and system specific errors, i.e., statements that are inadmissable for the system as presently configured. A final opportunity to err, comes when the source program is key-punched or typed in a form ready for compilation or assembly. The wary programmer who has checked his coded program for language and system errors will edit his punching or typing as he goes along. Also he will list his punched program and then carefully recheck his listing or cards with his original hand-written source program.

Experimental studies have shown that the speed and accuracy of checking or scanning paired lists of alphanumeric code words are enhanced if the lists are presented in similar formats and close together to minimize the extent of the observer's eye movements. Furthermore, this checking task can be performed most efficiently if the viewing conditions permit an observer to search for anomolies in the appearance or forms of the words in each pair rather than force him to read every character sequentially one at a time. The applicability of laboratory findings such as these to the effect of viewing conditions on the efficiency of programmers in checking programming sheets against program listings or punched cards should be verified.

By the time that a program is punched and ready for compilation, a programmer has had an opportunity to make (and correct) at least four different kinds of mistakes--errors of logic, language errors system specific errors and punching errors. The chances that he will make any or all these kinds of errors will vary for programs differing in size and complexity, written in different languages and run in system environments differing in complexity and alternative modes of operation. Similarly, the magnitude of the problem a programmer will face in isolating, diagnosing and correcting errors that remain in his program after it is ready for compilation will vary correspondingly. More over, the magnitude of his debugging problem will also be affected by the amount and kind of assistance he receives from the computer system.

Most compilers will not only not compile incorrect code but will identify the kind and location of certain classes of inadmissable statements. But after such errors have been corrected and the program compiled, some coding punching, logical and system specific errors can still remain undetected. The experienced programmer expects this. and will include in his program temporary statements that can be easily removed which call for additional information, such as echo prints of all input data prints of significant intermediate results as well as final results and possibly special printouts upon entering sub-programs. He anticipates errors and prepares in advance to acquire such additional information about the operation of his program as he believes will help him locate and correct program bugs. At execution time program bugs make their presence known by one or the other of the following signs: the program does not execute comes to an unprogrammed halt, cycles through a segment of program continuously, produces an incorrect solution to a problem, prints output in incorrect formats or as unintelligible garbage. The general problem then facing the programmer is to acquire and interpret additional information a) about the operation of his program (expressed as higher level language statements or as a chronology of machine states and operations) and b) the status of the system environment that will enable him to correctly locate and diagnose the source of his program's failure. In general terms he needs three categories of information--a) detailed program performance information, b) system status information (past and present) and c) reference information against which he can interpret the performance and status information. Reference information will include several categories, e.g., program requirements, the design characteristics and operating requirements and constraints of the computer and its operating system, the program languages employed and their characteristics and interface requirements, knowledge of the program development and the corrections already made, previous experience in writing related programs or of running programs in this particular system environment, available debugging aids, etc. The programmer will develop one or more hypotheses about the probable location and kind of fault in his program, and proceed to test each against the information available. To the extent that the available information is inadequate to test his hypotheses, the programmer will use the debugging aids available with the system to acquire the necessary additional information.

Questions about how to encode and format different kinds of information for display on CRT scopes or printouts so that a programmer can interpret the information most easily and use the information effectively for debugging, represent basic and significant human factors problems. Under separate contracts to NASA, Office of Manned Space Flight, Dunlap and Associates, Inc., have already investigated and developed computer graphic display formats to assist system test engineers detect and diagnose failures during prelaunch checkout. Comparable human factors research needs to be done to develop and evaluate new methods of using computer displays to assist programmers assimilate and use effectively all the information that debugging aids can now provide.

However, the most critical problem in debugging that requires intensive human factors investigation concerns the development and testing of alternative debugging strategies. Nowhere in the literature have we found any comprehensive guidelines, nor even discussion of how debugging aids can be used most effectively to locate and diagnose program faults in different kinds of programs and system environments.

Program Debugging Human Factors Research

Research is needed to 1) better define the scope of the debugging problem, the nature of the process and the factors that determine its relative difficulty, 2) identify common approaches to debugging and the kinds and amounts of information requested, 3) test and evaluate alternative debugging strategies, and 4) explore the effectiveness of revised methods of encoding and displaying computer generated information during debugging.

The first two sets of objectives listed above lend themselves to a research approach based on survey type methods. These methods could include a questionnaire mailed to a relatively large number of programmers working in a number of different organizations with differing kinds of system applications, or alternatively structured interviews conducted with a much smaller but similar sample of programmers. An alternative or supplementary approach for exploring the second set of objectives would be experimental. A group of available programmers could be asked to perform their normal programming and debugging activities under conditions which provided complete records of all errors committed, corrected, overlooked and subsequently debugged with the aid of the computer, and of all information requested and actions taken at the computer during debugging.

The third objective clearly requires an experimental approach in which measures of effectiveness, such as time to solution, requests of information, use of debugging aids, length of program, etc., and measures of costs, such as terminal operating cost, CPU cost, labor cost, etc., would be obtained from a sample of programmers working under controlled conditions. The resources available in personnel, computing facilities and time would dictate the form and scope of the experimental design. But in principle, programmers would use different diagnostic strategies to identify different kinds of faults in differing kinds of programs operating in differing system environments.

Similarly, the fourth objective of evaluating the effectiveness of improved display formats for assisting programmers to debug their programs could best be investigated experimentally. This research would include a number of small experiments in which the value of different formats would be tested for performing particular part-tasks, such as cross-checking lists of codes, scanning dumps of data for anomolous values, or for tracing the sequences of operations in parallel machines or parts of a machine, etc. It might then provide an overall evaluation of either a recommended "package" of display formats for specified tasks or the use that programmers make of a requestable repertoire of display formats during the performance of a representative range of debugging tasks.

BIBLIOGRAPHY

1. Auerbach Corporation. Visual Information Display Systems, Chapter V. Interactive Display Systems, NASA SP-5094, Office of Technical Utilization. NASA, Washington, 1968.

2. Chasen, S.H., and Seity, R.N. "On-Line Systems and Man-Computer Graphics," Computers and Automation, November 1967.

3. "CRT Displays." Modern Data, Technology Profile, Parts 1, 2, and 3, July, August, September 1968.

4. Earl and Goff. "Experimental Study of Two Data Entry Methods," Perceptual and Motor Skills, 1965.

5. Garrett, Gerald V., Thornton, Carl L. and Cabe, Patrick A. "Human Factor Evaluation of a Computer-Based Information Storage and Retrieval System," Human Factors, 10 (14), pp. 431-436, 1968.

6. Gould, John D. "Visual Factors in the Design of Computer-Controlled CRT Displays," Human Factors, 10 (14), pp. 359-376, 1968. (IBM, Yorktown Heights)

7. Grace, Gloria. "Application of Empirical Methods to Computer-Based System Design," Journal of Applied Psychology, Vol. 50, No. 6, 1966. (SDC)

8. Keast, David N. "Survey of Graphic Input Devices for Computer-Aided Design," Machine Design, August 3, 1967.

9. Landis, Daniel; Slivka, Robert; Jones, James and Silver, Carl. Experiments in Display Evaluation. Prepared for ONR, Contract NONR-4832(00), Franklin Institute Research Laboratories, Technical Report 1-194, 1967.

10. Landis, Daniel, Slivka, Robert, Jones, James; Harrison, Sylvia, and Silver, Carl. Evaluation of Large Scale Displays. Prepared for USAF Rome Air Development Center, Contract No. AF30(602)-3901, Franklin Institute Research Laboratories, Technical Report 1-182, 1966.

11. Pepler, R.D. and Wohl, J.G. Display Requirements for Prelaunch Checkout of Advanced Space Vehicles. Prepared for the Rand Corporation. Dunlap and Associates, Inc., DRD-64-76, May 1964.

12. Pepler, R.D. and Wohl, J.G. Integration of Man and Computer in Prelaunch Checkout of Advanced Space Vehicles. Memo. RM-4506-NASA, April 1965.

13. Ridings, H.J., "The Grafacon Man-Machine Interface--A Commercial Rand Tablet," Datamation, July 1967.

14. Silver, Carl; Landis, Daniel and Jones, James. Development of Criteria for Evaluation of Large-Screen Displays. Prepared for USAF Rome Air Development Center, Contract No. AF30(602)-3302, Franklin Institute Research Laboratories, Technical Report 1-052, 1965.

15. Weisberg, David E. "Graphic Displays: Matching Man to Machine for On-Line Control," Control Engineering, November 1968.

16. Prelaunch Checkout Computers as Decision Aids:

    I. Software Implications of Test Planning and Control Functions, (SSD-65-226, 510-TM-14).

    II. Data Conditioning and Display for Test Monitoring, (SSD-65-226, 510-TM-15).

    III. Data Organization and Display for Fault Isolation, (SSD-65-227, 510-TM-16).

    Submitted to OMSF, Apollo Program Office, NASA by Dunlap and Associates, Inc., 1965.

17. Data Conditioning and Display for Apollo Prelaunch Checkout:

    - Signal Flow Display Technique, 675-TM-5, SSD-66-344.
    - Servo Response Display Technique, 675-TM-7, SSD-66-353.
    - Single Parameter Data Compression and Display, 675-TM-11, SSD-373.
    - Multi-Parameter Monitoring Display, 675-TM-12, SSD-374.

    Submitted to OMSF, Apollo Program Office, NASA by Dunlap and Associates, Inc., 1966.

NEW TECHNOLOGY APPENDIX

"After a deligent review of the work performed under this contract, no new innovation, discovery, improvement or invention was made."