

by memory reference instructions, giving a program in memory exclusive control of the I/O. Since the DOS-M Disk Manager and Job Processor are always in core and protect Locations 1-0100, it was necessary to have a device for wresting I/O control from DOS-M.

4. For many purposes, this kind of pseudorandom generator will serve. However, for simulation work bias and failure of runs up and down to be random

may arise. For simulation we use linear shift-register methods (on related Tausworthe generators, see Tootill, Robinson, & Adams, 1971).

5. It is actually the address of a parameter that is stored in IBRIT and ILUX. The instruction of Line 11(c) LDA IBRIT,I loads the A register indirectly, that is, with the contents of the address stored in IBRIT.

6. Note 2 applies also to ILUX [see (b) Line 10].

peripheral to the computer have their own interrupt circuits and are plugged into a hard-wired bus, in which their location establishes their interrupt priority. Changing priorities is simply a matter of moving the interface cards around.

We have response levers and a Scientific Prototype tachistoscope, in addition to the above, hooked up to the computer system.

## COGLAB: A computer system designed for human research\*

RICHARD L. TAYLOR

Memorial University of Newfoundland, St. John's, Newfoundland, Canada

The COGLAB system, consisting of a modular on-line Executive program with equally modular utility routines, was designed to drive a large fast phosphor CRT as an n-channel tachistoscope and to service a human observer whose task during a session is to process displayed information. It is currently operational on a Hewlett-Packard 2114 computer in my information-processing laboratory.

The objectives in developing COGLAB were: (1) to permit studies requiring tachistoscopic timing but which would not be constrained during conception by a fixed number of available stimulus "channels," (2) to permit the presentation of both static and dynamic visual stimuli such as, for example, alphanumeric symbols and rotating two-dimensional projections of three-dimensional solids, (3) to permit both conventionally conceived studies in which values on stimulus parameters are predetermined before an experiment occurs, as well as less frequently employed paradigms in which important parameters are systematically varied, contingent on a S's performance during a task (an example here being the attempt to hold  $d'$  constant within and across sessions in a visual detection task by manipulating stimulus exposure duration, and (4) to enable experiments to proceed either fully S-paced, where the S is able to run as many trials for as long and as rapidly as he can without fatigue, or else to proceed under precise timing controls throughout a session. This paper describes my solutions to these four objectives and outlines some general principles which are important for similar systems.

\*This project is supported by the National Research Council of Canada through Operating Grant A-7811 and Special Equipment Grant E-3329 to the author.

### EQUIPMENT

The hardware for COGLAB consists of: a Hewlett-Packard Model 2114 computer with 8K of 16-bit-word core memory; an ASR-33 Teletype; a 600-character/sec paper-tape reader; a programmable 10-kHz crystal clock; an 8 x 10 in. CRT display subsystem equipped with a P31 aluminized phosphor, which decays to 10% of its peak luminance in less than 1 msec for the levels employed; and a 16-bit general-purpose interface card that is connected to a rack of solid-state BRS modular logic. All of the devices

### THE SOFTWARE

Each device that is driven by the computer has one or more interrupt drivers, which are subroutines designed to accomplish specific tasks. Each driver has one or more entry points which can be accessed by a simple FORTRAN subroutine call which specifies the name of the entry point and the parameters, if any, requiring transfer. For example, the 16-bit general-purpose interface card has an interrupt flip-flop which is connected through the solid-state external logic to levers available to a S such that, when a lever is pressed, the interface-interrupt flag is set. If one wishes to know when a S presses a lever, it is necessary only to include "CALL RYSET" in the proper place of the Executive program. When a response occurs, the subroutine RYSET, in addition to interrupt housekeeping chores, stores a constant

Table 1

PROGRAM LAB 14	
COMMON JJ (832), S (64)	
CALL NREAD	; Reads in pattern library
MSA = 3000	
MSB = 5000	
CALL CLOCK (1)	; Sets up session timer for ½ hr
9 CALL READY	; Displays the word "READY?" and waits for S to respond
N = N + 1	
CALL RESP (K1, K2, K3)	; Selects three digits as stimuli
CALL MESS (2, 0, 0, 0)	; Displays markers on screen where the key stimuli will appear later
CALL WAIT (MSA)	
IF (1 - K3) 10, 10, 11	
10 K2 = K3	
11 CALL MESS (3, N1, N2, 0)	; Displays words describing digits to appear
CALL STCLK (1)	; Starts millisecond timer
CALL PULL (K1, K2)	; Extracts digit patterns from library
12 CALL DIGIT (JA, JB)	; Selects one of four scanning directions
IF (5 - JA) 12, 12, 13	
13 CALL DIGIT (JD, JB)	; Selects one of four transformations
IF (5 - JA) 13, 13, 14	
14 CALL NFIND (JA, JB)	; Rearranges stored digit patterns
CALL RDCLK (NR)	; Reads current value of millisecond timer
MSB = MSB - NR	; Subtracts time for above from programmed delay
CALL WAIT (MSB)	
CALL MESS (4, 0, 0, 0)	; Displays actual digits selected above
CALL TRIAL (K3, JRT)	; Returns with error marker or S's RT
IF (1 - JRT) 16, 15, 16	; Tests to see if S made an error
15 CALL MESS (5, 0, 0, 0)	; Displays "ERROR" message on scope
CALL WAIT (MSA)	
GO TO 20	
16 CALL MESS (6, 0, 0, JRT)	; Displays an S's correct RT on screen
CALL WAIT (MSB)	
20 CALL DATUM (N, JRT, JA, JB, K3)	; Processes the data for this trial
CALL CLOCK (2)	; Tests to determine if S needs a rest
CALL TADJ (MSA, MSB, JRT)	; Resets time interval constants according to S's RT for this trial
GO TO 9	
END	

at NRESP, replacing zero, and subsequent tests of the content of that cell will tell us that the S has responded. These tasks require, at the most, .15 msec. In the meantime, because the system is operational in an hierarchical interrupt mode, the computer may be busy doing other things such as refreshing the CRT display.

The first priority is assigned to the programmable clock, and the system is set to respond to 1-msec interrupts after the clock is enabled at the beginning of a session. The clock runs continuously with crystal precision after being initiated and continues to deliver priority interrupts. Approximately 450 instructions of a program can be executed between clock interrupts. The second priority is assigned to the CRT digital-to-analog interface. Points can be plotted at an average rate of 200/msec. A total of 65,536 discrete points are available by converting 8 bits each of x- and y-axis output into different voltages between 0 and 10. A third axis (z) permits luminance for each point to be set independently from 0 to 100 fL. In order for a given display to appear constant, the points must be replotted about every 20 msec or so, depending on luminance. Accordingly, the CRT, when enabled, produces interrupts about every 20 msec. The third priority is assigned to the general-purpose interface card. A lever response by an S sets an external flip-flop, which causes an interrupt whenever the clock or CRT are not being serviced. In the conditions with the worst cases, a delay of 75 microsec may be encountered before the S's response is serviced. The error obtained in reaction-time experiments, for example, measured from stimulus onset to response initiation, will not exceed .1 msec. The fourth level of interrupt priority is assigned to the teletypewriter, which is turned on during a session only as a place to dump diagnostic error messages if something should go wrong and which is used primarily as the place where the data from a session is dumped at the end.

COGLAB is programmed in FORTRAN. An Executive program is written, compiled, and dumped in binary version onto paper tape. When an experiment is to be run, the paper tape is input by the high-speed paper-tape reader. If an experiment, including the Executive and all utility subroutines, required a full 8K of memory, the process of loading the full binary tape could require as much as 2 min.

An Executive is written largely as a series of subroutine calls. Table 1 is a somewhat simplified version of such a

program that has actually been used. The experiment involves n trials in which, on each, either one of two responses is possible but one is correct;

a stimulus may be any 1 of 35 items and to which any 1 of 16 different spatial transformations may be applied.

## The computer laboratory for instruction in psychological research, CLIPR

DANIEL E. BAILEY

University of Colorado, CLIPR, 1511 University Ave., Boulder, Colorado 80302

The Computer Laboratory for Instruction in Psychological Research (CLIPR) was established in the Department of Psychology at the University of Colorado in the fall of 1970. Three fundamental long-range goals of the laboratory are: (1) the development of undergraduate courses and course laboratories using the computer both as a primary object of study and as an instrument for teaching other material such as statistics and theory of learning, (2) as a training laboratory for graduate students working in areas of computer-assisted instruction, simulation, experiment control, and data acquisition, and (3) as a research laboratory for faculty and students of the department.

This paper presents a very brief overview of the function of the laboratory and the facilities of the laboratory. A more detailed description of the laboratory and its function is contained in Bailey (1970).

### TRAINING ASPECTS OF CLIPR

Much of the training achieved in the laboratory is by way of direct involvement of student and faculty with the computer in active research programs. However, a number of formal courses have been introduced as an integral part of the project. Two of these courses are at the undergraduate level: "Computer Applications in Social and Behavioral Sciences," and "Real-time Computers in Psychology." Two other courses are offered at the graduate level for students in psychology and computer science: "Computer Assisted Instruction," and "Laboratory Computers in Psychology."

A number of courses in the Departments of Psychology and Computer Science utilize the facilities of CLIPR for instructional purposes. Three psychology courses use on-line interactive capabilities for statistical analysis and computer-programming

training. One computer-science course uses the CLIPR facilities in the study of the design of real-time computer systems. Additional course work is being developed within the Department of Psychology to use CLIPR facilities, particularly the computer-assisted instruction portion of the project.

### RESEARCH APPLICATIONS OF CLIPR

A portion of the effort devoted to research is computer oriented. However, the primary thrust of the project is automation and sophistication of psychological research. Within the allotted space for this paper, I can only list a sampling of the research being developed within the framework of the laboratory. The list encompasses such subject areas as psycholinguistics, memory, verbal learning, concept formation, operant conditioning, behavioral genetics, visual pattern processing, electrophysiological bases of behavior, rapid-eye-movement involvement in learning consolidation, dynamic psychometric methods, cognitive functions in social psychology, and many others.

The current physical facilities limit the number of concurrently active research programs that can be supported in real time to three or four. However, as memory and peripheral equipment are added, the laboratory anticipates a broad expansion of this capability. On full development, the laboratory will be organized in terms of seven sublaboratories: Learning and Memory Laboratory, Automated Experiment Laboratory, Information Management Laboratory, Programming and Computer Assisted Instruction Laboratory, Visual Perception Laboratory, Psychobiology Laboratory, and a Linguistics Laboratory.