COMPUTER TECHNOLOGY

Apple tachistoscope

DON DIENER and WILSON P. SMEE
University of Nevada, Las Vegas, Nevada

This paper describes software for precise tachistoscopic presentation of stimuli on a raster-scan monitor controlled by an Apple IIe microcomputer. Routines that allow synchronization of the control program with the videoscanner are provided. The timing accuracy of stimulus presentation on this system was measured independently, by mounting phototransistors to the CRT. The tachistoscope is capable of millisecond accuracy in producing display-duration multiples of 16.7 msec. The listing of a program that implements a three-field (fixation point, target, mask) tachistoscope is included.

The problems associated with the use of computer-driven raster-scan cathode-ray-tube (CRT) monitors for tachistoscopic displays have been discussed in a number of published sources (e.g., Broyles, Prill, Marks, Salt-house, & Spencer, 1982; Lincoln & Lane, 1980; Reed, 1979; Stentis & Barowicz, 1978). Despite the high speed of commonly used microprocessors, a number of factors limit the temporal resolution of these devices. These factors include: (1) the long (10-15-msec) persistence of screen phosphors designed to produce a flicker-free display, (2) the time required to write information to video memory, and (3) the 60-Hz scanning rate (U.S.A.) of most raster-scan monitors. Fortunately, the Apple IIe computer embodies provisions for dealing with these limitations and allows millisecond control of display duration with no hardware modification. Most of the methods of display control available to Apple users can be easily implemented on other microcomputer systems.

The problems associated with phosphor decay rate can be easily overcome in most designs by following the target stimulus with a mask that writes an illuminated block over the display area. Thus, the interval between the target stimulus and mask onset, rather than phosphor decay, limits the display duration. The transparent access video display system of the Apple II series of computers (see Reed, 1979 for a discussion) also makes it possible to overcome the limitations associated with the transfer of information to the video display. Apple provides four graphics display "pages" (low-resolution graphics pages 1 and 2; high-resolution graphics pages 1 and 2) that permit the transfer of an entire screen of information to the video display between one videoscanner and the next.

The 60-Hz scanning rate of raster-scan CRTs imposes the most severe limitation on tachistoscopic presentations. Because the electron beam scans the screen only once every 1/60th of a second, display durations are limited to multiples of 16.67 msec. If multiples of 16.67 msec are acceptable, the remaining problem is to synchronize the

control program with the videoscanner. In tachistoscopic research involving a mask, this synchronization is necessary to ensure that the display data are not changed at the same time they are being displayed. Without synchronization, part of the target stimulus might be presented on one scan and the other part of the target and part of the mask on the next scan. The problem can be more severe if the delay between the onset of the display of video memory containing the target stimulus and that containing the mask is not a precise multiple of 16.67 msec. In this case, the number of scans during which the target stimulus is displayed can vary from trial to trial, depending on the point in the videoscanner at which the target display is initiated. Synchronization has the further advantage of allowing the precise recording of response latency in reaction time experiments.

It is possible to achieve synchronization with the Apple IIe entirely under software control. Decimal address -16359 (\$C019) of the Apple IIe contains a value that indicates the status of the vertical-blanking signal. This value is high (i.e., > 128) when the beam is scanning the screen and low for a period including vertical retrace. To synchronize the control program with the videoscanner, it is necessary to wait until this signal is low (i.e., < 128) and to continue with program execution only when the signal again is high. The WAIT command available to Apple IIe users makes this a relatively easy matter. Unfortunately, software access to the vertical-blanking signal is not available on the earlier Apple II or II+. Reed (1979) described hardware modifications for the Apple II and II+ computers that provide access to the vertical-synchronization pulse.

The demonstration program described in this paper will not run on the Apple IIc. Although the WAIT command has been implemented on the Apple IIc, we have been unable to locate the address of the vertical-blanking signal, or even to ascertain whether software access is available.

IMPLEMENTATION

A demonstration program is listed in the Appendix. The

Requests for reprints can be sent to D. Diener, Department of Psychology, University of Nevada, Las Vegas, 4505 Maryland Parkway, Las Vegas, NV 89154.

program requests the user to enter a display duration and number of trials. The display duration entered by the user is rounded to the nearest multiple of 16.7 msec. After the subject presses the space bar to begin the session, a fixation point is displayed, followed, in about .5 sec, by the target stimulus. After the specified display duration, the target is replaced by a mask consisting of a solid illuminated block. The target consists of either an "X" or an "O," selected on the basis of a random number. When displayed on an 11-in. video monitor, the target letters are approximately 11 mm in height. Following the stimulus presentation, the subject presses the appropriate key to indicate whether an "X" or an "O" was presented.

On a video monitor adjusted for normal text display, the target letter may be briefly visible after the onset of the mask. To eliminate this problem, it is necessary to readjust the monitor. The best results are obtained by increasing the brightness and decreasing the contrast relative to their normal settings.

To facilitate the programming necessary to tailor the tachistoscope to specific applications, a number of features of the program require further discussion.

Delay Routine

Control of the video display requires the use of a fairly accurate delay routine. The GOSUB 1000 on line 10 of the demonstration program creates a machine language delay routine with the following lines:

```
1000 POKE 768,206:POKE 769,14:POKE 770,3
1010 POKE 771,48:POKE 772,8
1020 POKE 773,169:POKE 774,a
1030 POKE 775,32:POKE 776,168:POKE 777,252
1040 POKE 778,76:POKE 779,0:POKE 780,3
1050 POKE 781,96
1060 POKE 782,n
```

The machine language program created by these BASIC lines calls the Apple monitor's built-in WAIT routine at address \$FCA8 (dec. 65,260) n times as determined by the value POKEd to 782. The program is located in an area of memory (dec. 768-975) which is not usable by BASIC programs or video display. The instructions on line 1000 decrement n by 1. Line 1010 branches on n = -1 to 781, which returns to the calling program. Line 1040 jumps back to 768 for another cycle. The instructions on line 1030 execute a jump-save-return to the WAIT routine, delaying for a period determined by the value (a) loaded in the accumulator in line 1030. The delay produced by a single call to the WAIT routine for a given value of a is: $(26 + 27a + 5a^2) / 2 \mu\text{sec}$ [see *Apple II Reference Manual* (Apple Computer, Inc., 1982b, p. 223)]. For a variety of applications, it is convenient to use a = 37 for a delay of approximately 4 msec (including overhead). This value allows accurate delay intervals from 4 to 508 msec (i.e., 127×4). Once the delay routine has been POKEd from BASIC, the commands "POKE 782,n: CALL 768" will result in a delay of 4n msec.

Delay Duration

In order to display a stimulus for a single videoscans, display memory must be changed after the scanning beam has traced the display location but before it returns on another cycle. The display portion of Apple's video cycle lasts for about 12.21 msec. The vertical-banking pulse occupies the remaining 4.46 msec of the 16.67-msec cycle. If the target figure occupied the entire screen, the delay between the execution of the command to display the target and the command to display the mask would have to be between 12.21 and 16.7 msec. Few applications, however, require anything like a full-screen display. For small displays using, for example, only the center of the screen, the timing problem is less critical. The choice of the delay interval used in the demonstration program is based on the results of testing discussed later in the paper.

Display Control

The following lines in the demonstration program control the display sequence fixation point-target-mask:

```
180 GOSUB 500: REM TARGET TO HIRES1
190 HOME:VTAB 12:PRINT TAB (20); "+"
200 POKE 782,125:CALL 768:REM HALF-SECOND DELAY
210 POKE 782,D
220 WAIT -16359,128:WAIT -16359,128,128:
    WAIT -16359,128:PK = PEEK(-16304):CALL 768:
    PK = PEEK(-16299)
```

Line 180 transfers control to the subroutine at line 500, which, based on the value of a random number, writes (but does not display) either an "X" or an "O" to HIRES1. Line 190 displays a fixation point (a "+" sign) in the text mode at the approximate center of the screen. Line 200 delays execution of the next line, displaying the fixation point for approximately .5 sec. Line 210 sets the delay between the target stimulus and mask. Line 220 synchronizes the control program with the videoscans, first waiting for the vertical blanking pulse to rise (when the electron beam is scanning the tube) and then waiting for it to fall and then rise again. Thus, the next statements will always be executed at the beginning of a videoscans. The last three statements on line 220 display HIRES1 (target) for a period determined by the machine language delay routine and then display HIRES2 (mask) until the subject responds with an "X" or an "O." It is important that all of the statements on line 250 appear on a single line to avoid the extra time required to process additional line numbers.

Hints for Programming

HIRES1 and HIRES2 are located pretty much in the middle of Apple's memory map (\$2000-\$FFFF; dec. 8192-24575). By default, BASIC program and variable space begins at \$0800 (dec. 2048), allowing only about 6K bytes of programming space before the program overlaps HIRES1. If a program longer than 6K bytes is RUN, the first HGR command erases all lines of the program that occupy memory locations above \$2000. The problem

can be avoided by locating the program in the 13+K bytes (\$6000-\$95FF) available above HIRES2. This is accomplished by setting the program pointer to \$6001 before the program is LOADED or RUN. The following commands, issued from the immediate mode or a "HELLO" program, set the program pointer above HIRES2 and clear the byte previous to the program to zero:

```
POKE 103, 1: POKE 104,96: REM START PROGRAM AT $9601
POKE 24576,0: REM SET BYTE PREVIOUS TO PROGRAM TO ZERO:
```

On some, but not all, Apple IIs, the default color for HIRES display (HCOLOR) is black. The HCOLOR must be set to 3 (white) or some other value 1-7 before the writing to a HIRES page. The statements HGR and HGR2 briefly display the contents of HIRES1 or HIRES2 and then clear the screen to black before accepting new input from HPlot statements. This results in a brief display of the HIRES contents. It is therefore advisable to issue the HGR and HGR2 commands early in the program. In the demonstration program, these commands are issued in a subroutine called from the first program line.

In the demonstration program, the text display is used for the fixation point, HIRES1 for the target stimulus, and HIRES2 for the mask. Other sequences are possible. For example the fixation point can be on HIRES1, the target on HIRES2, and the mask on low-resolution graphics page 1. The sequence HIRES1-TEXT-HIRES2 can be used for tachistoscopic display of text material. This requires that the text be written while HIRES1 is displayed. A list of the commands necessary to effect these sequences can be found in Volume 2 of *Apple II Applesoft BASIC Programmer's Reference Manual* (Apple Computer, Inc., 1982a, pp. 259-262).

TESTING

The timing accuracy of the interval between target stimulus and mask onset in the above-described tachistoscope was evaluated in the following manner. A 50 × 50 square beginning (top left) at row 75, column 50 was written to HIRES1, and a similar square, displaced 150 columns horizontally, was stored in HIRES2. Phototransistors (Radio Shack 276-130) were mounted on the screen in the top left corners of the squares. The output of the phototransistors was fed via a single-stage transistor amplifier to a Simpson Model 2725 digital counter, configured as an interval timer. The system (including CRT brightness) was carefully adjusted to trigger the timer on the first refresh cycle. A BASIC program including the above-described delay routine (a = 37) and line 250 of the demonstration program (exactly as shown) controlled the display sequence HIRES1-delay-HIRES2. In use, the target stimulus residing on HIRES1 and the mask residing on HIRES2 would be displayed at the same position on the screen. However, given the extremely fast horizontal scanning rate of CRT displays, the timing effects of

horizontal displacement of the mask in the test setup are within measurement error.

The tachistoscope was tested with delay intervals varying in steps of 4 msec from 4 to 260 msec. Any delay that allowed the contents of HIRES1 to have been displayed for one raster scan should have resulted in a 16.7-msec display duration. Theoretically (i.e., with perfect synchronization and timing accuracy), any delay between 4 and 16.7 msec should have resulted in a one-scan display. In practice, a delay of 16 msec resulted in a 16.7-msec display on 50% of the trials and a 33.3-msec display on 50% of the trials (tested in a run of 100). Similar results were obtained for other delays that were exact multiples of 16 msec. There are several implications of these findings. The variability in the number of display cycles suggests that either synchronization is not perfect or that the execution time of the display commands is slightly variable. The fact that this variability is produced by a delay of 16 msec indicates that, combined with the overhead of the display commands and the call to the delay routine, a nominal 16-msec delay is close to the 16.7-msec video cycle. The finding that larger multiples of 16 msec result in a similar degree of variability indicates that the delay routine produces accurate multiples of 4 msec. Given these findings, the delay interval that should result in the highest degree of stability is 8 msec shorter than the intended exposure duration. This value is computed on line 80 of the demonstration program. If material is to be displayed on the bottom quarter of the screen, a delay interval 12 msec shorter than the display duration should be chosen.

Testing revealed that the test configuration was very stable when used with delay intervals other than multiples of 16 msec. Runs of 100 trials were conducted with delay intervals of 8, 20, 44, 92, 140, and 244 msec, to produce display durations of 16.7, 33.3, 50, 100, 150, and 250 msec, respectively. The results of these tests, measured to the nearest .1 msec, are shown in Table 1. In no case did the measured display duration differ appreciably from the expected multiple of 16.7 msec. It is impossible to estimate the extent to which the small amount of variability in the measured durations reflects real display differences as opposed to measurement error (e.g., instability in the triggering mechanism).

Table 1
Measured Interstimulus Interval for
Some Common Display Durations

Nominal Duration	Number of Scans	Delay Interval	Mean Measurement	Range
16.7	1	8	16.6	16.3-17.0
33.3	2	20	33.6	33.3-34.2
50.0	3	44	50.4	49.6-51.0
100.0	6	92	100.4	99.7-100.7
150.0	9	140	150.2	149.8-150.9
250.0	15	244	250.4	250.0-251.1

Note—All time intervals are expressed in milliseconds. Results are based on runs of 100 trials.

The timing accuracy obtainable with raster-scan CRT video monitors seems perfectly acceptable for most tachistoscopic applications. Moreover, these devices have a number of advantages over alternative computer-driven display systems (e.g., x-y coordinate point-plots; LED or LCD displays), including cost, ease of interfacing, and the ability to display instructions to experimental subjects. Although a number of the procedures discussed in the present paper are specific to the Apple IIe, it should be possible to obtain similar results with most other microcomputers. Users of the Apple II or II+ should be able to apply the programs discussed with the addition of hardware modifications to access the vertical synchronization signal (see Reed, 1979).

REFERENCES

- APPLE COMPUTER, INC. (1982a). *Apple II Applesoft BASIC programmer's reference manual* (Vol. 1-2). Cupertino, CA: Author.
- APPLE COMPUTER, INC. (1982b). *Apple II Reference Manual*. Cupertino, CA: Author.
- BROYLES, J. W., PRILL, K. A., MARKS, M. H., SALTHOUSE, T. A., & SPENCER, K. L. (1982). An empirical approach to the timing limitations of the raster-scan CRT. *Bulletin of the Psychonomic Society*, 20, 287-289.
- LINCOLN, C. E., & LANE, D. M. (1980). Reaction time measurement errors resulting from the use of CRT displays. *Behavior Research Methods & Instrumentation*, 12, 55-57.
- REED, A. V. (1979). Microcomputer display timing: Problems and solutions. *Behavior Research Methods & Instrumentation*, 11, 572-576.
- STENTIS, K. P., & BAROWICZ, M. B. (1978). A CRT blanking circuit for video terminals. *Behavior Research Methods & Instrumentation*, 10, 856-857.

APPENDIX

Demonstration Program

```

10 GOSUB 1000: REM   CREATES DELAY ROUTINE; SEE TEXT

20 GOSUB 1100: REM   MASK TO HIRES2

30 HOME : PRINT "PLEASE ENTER EXPOSURE DURATION IN"

40 INPUT "APPROXIMATE MULTIPLE OF 16.7 MSEC ":D

45 REM   ROUND TO NEAREST MULTIPLE OF 16.67

50 A = INT (D / 16.67) / 2: IF A = 0 THEN A = .5

60 B = INT ( INT (A + .5) * 1666.67 ) / 100

70 PRINT : PRINT "EXPOSURE DURATION WILL BE :B: MSEC"

75 REM   D = NUMBER OF 4 MSEC CYCLES FOR DELAY ROUTINE

80 D = INT (B / 4) - 2

90 PRINT : INPUT "ENTER NUMBER OF TRIALS FOR THIS SESSION ":N

100 HOME : PRINT "YOUR TASK IS TO DISTINGUISH BETWEEN"

110 PRINT "X'S AND O'S."
```

```

120 PRINT : PRINT "PRESS THE X KEY IF YOU SEE AN X, OR"

130 PRINT "THE O KEY IF YOU SEE AN O."

140 PRINT : PRINT "POSITION YOUR FINGERS THEN PRESS THE"

150 PRINT "SPACE BAR TO BEGIN."

160 GET SB$: IF SB$ < > " " THEN 160

170 FOR I = 1 TO N

180 GOSUB 500: REM   TARGET TO HIRES1

190 HOME : VTAB 12: PRINT TAB( 20): "+"

200 POKE 782,125: CALL 768: REM   HALF-SECOND DISPLAY

210 POKE 782,0: REM   'D' SETS TARGET EXPOSURE DURATION

215 REM   VIDEOSCAN SYNCHRONIZATION; SEE TEXT

220 WAIT - 16359,128: WAIT - 16359,128: WAIT - 16359,128:

PK = PEEK ( - 16304): CALL 768: PK = PEEK ( - 16299)

230 GET R$: IF R$ < > "X" AND R$ < > "O" THEN 230

240 TEXT : HOME : VTAB 23

245 REM   WAS RESPONSE CORRECT?

250 IF SS = 1 AND R$ = "X" OR SS = 0 AND R$ = "O" THEN 270

260 PRINT TAB( 15): "INCORRECT": GOTO 280

270 PRINT TAB( 15): "CORRECT "

280 PRINT "PRESS SPACE BAR FOR ANOTHER TRIAL"

290 GET SB$: IF SB$ < > " " THEN 290

300 NEXT I

310 HOME : PRINT "TYPE 'RUN' <RETURN> FOR ANOTHER SESSION"

320 END

499 REM   TARGET TO HIRES1

500 CALL 62450: REM   CLEAR HIRES1

510 SS = INT ( RND (8) * 2): REM   RANDOM NUMBER BETWEEN 0 AND 1

520 IF SS = 0 THEN 560: REM   PLOT 'O' IF SS=0

530 HPLLOT 129,85 TO 142,98: REM   PLOTS 'X'
```

```
540 HPL0T 142,85 TO 129,98
550 RETURN
560 HPL0T 131,84 TO 140,84: REM THESE LINES PLOT '0'
570 HPL0T 131,99 TO 140,99
580 HPL0T 130,85 TO 130,98
590 HPL0T 141,85 TO 141,98
600 RETURN
995 REM DELAY ROUTINE: SEE TEXT
1000 POKE 768,206: POKE 769,14: POKE 770,3
1010 POKE 771,48: POKE 772,8
1020 POKE 773,169: POKE 774,37
1030 POKE 775,32: POKE 776,168: POKE 777,252
1040 POKE 778,76: POKE 779,0: POKE 780,3
1050 POKE 781,96
1060 POKE 782,1
1070 RETURN
1095 REM SOLID MASK TO HIRES2: COLOR=WHITE
1100 HGR2 : HCOLOR= 3
1110 FOR X = 124 TO 147: HPL0T X,80 TO X,103: NEXT X
1115 REM SET TO HIRES1, FULL-SCREEN DISPLAY
1120 HGR : POKE - 16302,0
1140 TEXT : HOME : RETURN
```

(Manuscript received July 24, 1984;
revision accepted for publication September 12, 1984.)