

Microcomputer measurement and analysis of newborn sucking

D. C. MARTIN

University of Washington, Seattle, Washington

D. BORG-BREEN

National Oceanic and Atmospheric Administration, Seattle, Washington

and

J. C. MARTIN and A. P. STREISSGUTH

University of Washington, Seattle, Washington

An on-line Basic computer program with on-line monitor prompts and read-out collects, stores, and digitizes pressure and latency of sucking responses for later analyses. The program and analog equipment are described, and approximate costs are noted.

A defective sucking response is one of the earliest and most reliable indicators of neurological distress in the newborn human (Cowett, Lipsitt, Vohr, & Oh, 1978); sucking pressure, latency, and frequency may all be negatively affected (Kron, Stein, & Goddard, 1963; D. C. Martin, J. C. Martin, & Streissguth, 1979). The inability to ingest sufficient nutrients may compromise an already weakened system in high-risk infants, such as those born prematurely (Cowett et al., 1978) and those who are exposed prenatally to drugs (Kron, Litt, Phoenix, & Finnegan, 1976; Stock, Streissguth, & D. C. Martin, 1985). It is important to identify a sucking problem early, since failure to thrive can predict later neurobehavioral deficits (Lipsitt, 1979).

In 1979, we published a sucking conditioning study of infants born to smokers and social drinkers, in which we utilized then state-of-the-art equipment: preamplifiers and differential pressure transducers, transducer read-out equipment for signal amplification, and instrumentation tape-recorders. The schedule was programmed with an LVB Logic Box, and the signals were digitized and laboriously analyzed on a PDP-11 minicomputer (J. C. Martin, D. C. Martin, Lund, & Streissguth, 1977). Today, however, microcomputer technology allows the entire instrumentation and computer system that we described in our previous study to be replaced with a small portable package. Yet when we recently attempted to replace the old system, we were unable to find readily available hardware or software, so we had to design our own.

Hardware Design

Three NEC Powermate Port/40, "lunch-box" microcomputers were selected for data acquisition and analysis. Two of them had 20-MB disks; the third unit had a 40-MB disk, which was used for software development. A DAS-8 analog-to-digital conversion card (MetraByte, Taunton, MA) was installed in each machine. A Calex 160-MK bridge sensor (Calex, Pleasant Hill, CA) amplifier interface was connected to a DTXtm (TNF-R) disposable pressure transducer (Spectramed, Oxnard, CA), which was connected to 6 in. of flexible .5-cm-diameter tubing, with a Luer fitting (Upchurch Scientific Inc., Oak Harbor, WA). The orthopedic type of nipple did not form a tight seal with the tubing, so a standard nipple was used (Evenflo). An opening sized for either a newborn or a prematurely born infant worked equally well. Nipples were discarded after each use, and the tubing and pressure transducer were sterilized with gas for repeated use. An X-Caliber calibrator (Spectramed) was used for calibration prior to the start of Trial 1. One entire package was assembled at a cost of \$4,325.00 (excluding disposable

Table 1
Digitized Printout of Sucking Session for One Baby

Trial:01	Avgpressure:00.2	#Sucks:5	Latency:08.10	07:11	A.M.
Trial:02	Avgpressure:02.5	#Sucks:1	Latency:18.62	07:11	
Trial:03	Avgpressure:03.5	#Sucks:5	Latency:00.50	07:12	
Trial:04	Avgpressure:01.6	#Sucks:2	Latency:13.55	07:12	
Trial:05	Avgpressure:01.5	#Sucks:0	Latency:20.05	07:12	
Trial:06	Avgpressure:12.6	#Sucks:15	Latency:12.89	07:13	
Trial:07	Avgpressure:01.2	#Sucks:0	Latency:20.05	07:13	
Trial:08	Avgpressure:27.5	#Sucks:36	Latency:01.71	07:15	
Trial:09	Avgpressure:26.9	#Sucks:29	Latency:00.83	07:16	
Trial:10	Avgpressure:21.6	#Sucks:25	Latency:00.22	07:17	
Trial:11	Avgpressure:08.2	#Sucks:12	Latency:00.83	07:18	
Trial:12	Avgpressure:06.6	#Sucks:10	Latency:00.94	07:19	
Trial:13	Avgpressure:11.6	#Sucks:14	Latency:00.94	07:20	
Trial:14	Avgpressure:17.4	#Sucks:27	Latency:01.87	07:21	
Trial:15	Avgpressure:25.9	#Sucks:36	Latency:00.44	07:22	A.M.

This research was supported by NIDA/ADAMHA Grant, DA05365 to A. P. Streissguth. Reprints may be obtained from D. C. Martin, Department of Biostatistics, SC-32, University of Washington, Seattle, WA 98195.

items): NEC, \$2600; math coprocessor, \$250; calibrator, \$500; cable, \$100; bridge sensor, \$215; 8-channel analog input board, \$560; and Turbo BASIC driver, \$100. The current cost would be less. Laptop computers are now available that can accept the analog input board. They would be more convenient than the NEC Powermate.

Procedure

This apparatus was developed for newborns who were 1 day old. They were tested between feedings on a vari-

ety of tasks, which had been chosen to assess neurological integrity. The sucking task consisted of 15 20-sec trials with 5-sec intertrial intervals, for a total test period of 6-7 min. At the start of the trial, the nurse placed the non-nutritive nipple in the infant's mouth, after which the technician pressed the Enter key on the computer. The computer then began the count of elapsed time, which was displayed in seconds on the screen. At the end of the trial, the nurse removed the nipple from the baby's mouth, counted 5 sec, and reinserted the nipple; Trial 2 then be-

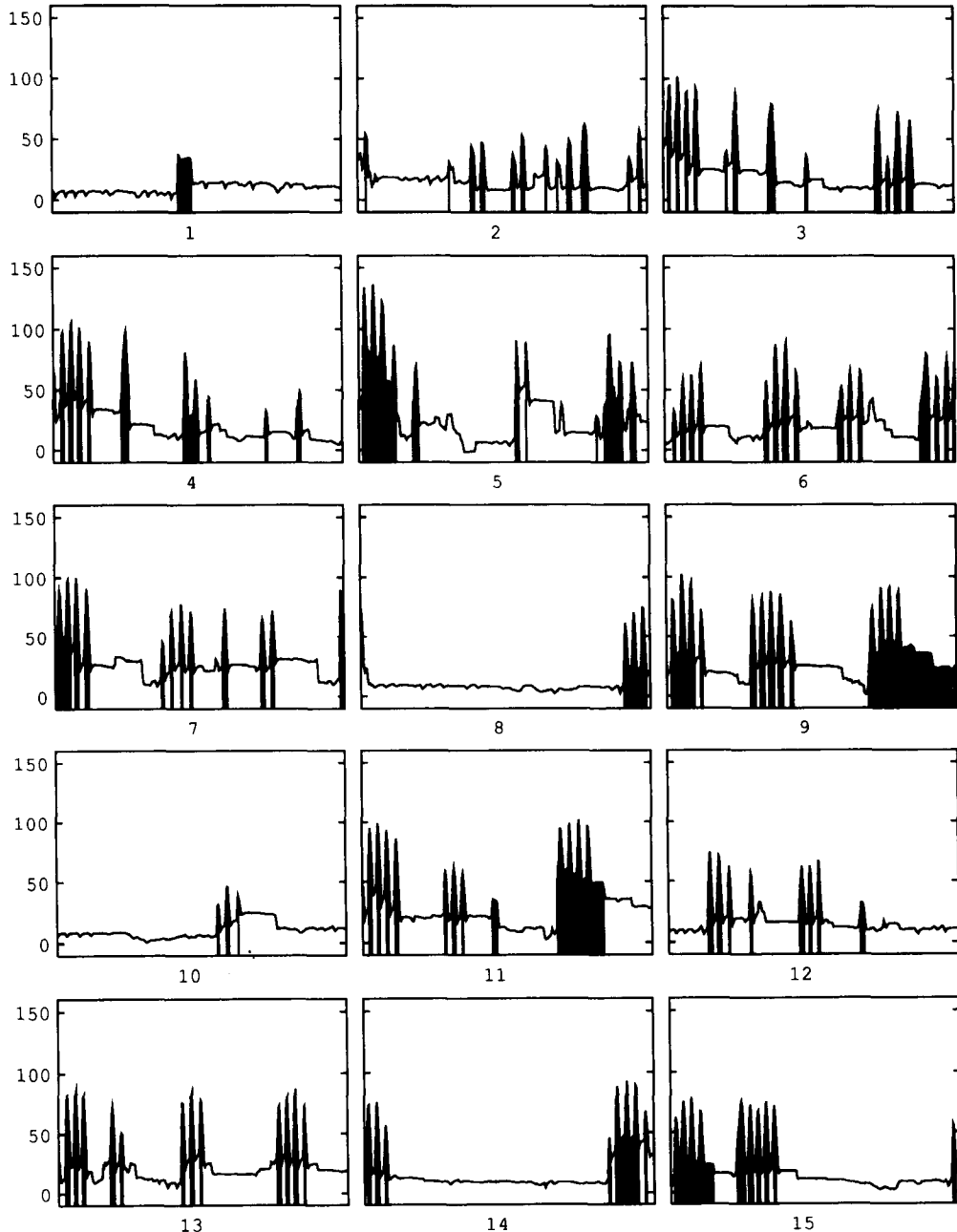


Figure 1. Analog record of sucking session for one baby. Each numbered block represents one 20-sec trial. The horizontal line is the floating baseline; the vertical excursions are sucks. The horizontal axis represents time; the vertical axis represents pressure in millimeters of mercury.

gan. Sucking itself was displayed on the screen in real time as a moving graph that served as a visual monitor.

Software Design

The program is written in TurboBasic (Borland, Scott's Valley, CA).¹ MetraByte LOAD-DAS-8 and DAS-8 subroutines carry out the analog-to-digital conversion. The graphics routines are device-independent but use a VGA (videographics adaptor) emulation mode.

Table 1 is a typical digitized printout of a sucking session and Figure 1 is an analog printout. The two are not records of the same infant.

Dependent Measures

The following measurements are taken: (1) the average pressure in millimeters of mercury for each of the 15 trials, (2) the number of sucking episodes as defined by a floating baseline (see below), and (3) the latency in seconds to the first suck for each of the 15 trials.

Floating Baseline

A shifting baseline to define an individual suck is mandatory, since some infants do not completely release the pressure between sucks but maintain it and raise the pressure for the next episode.

The floating baseline, b_t , is computed when in the non-sucking state by $b_{t+1} = 0.9b_t + 0.1p_t$, where p_t is the pressure at time point t . A suck state is assumed to begin once p_{t+1} exceeded b_t by 10 mm and is terminated when p_t decreases below the threshold that starts the suck (the baseline at the beginning of the suck plus 10 mm). Once the suck state is terminated, the floating baseline update is started again. See subroutine TESTSUCK (Appendix A) for the algorithm.

Program Description

1. The baby number and related data entry are specific to this study and not included.

2. The program requests a -50 mm and a 0-mm calibration pressure. The calibration is displayed graphically and can be repeated until the correct pressure is achieved.

3. The program takes 15 trials of 20 sec at 18 samples per second. This display is useful for verification of correct operation. The sucks and timing are analyzed in real time. The display is switched to reverse video at the end of the trial. This is noticeable even when the examiner is not looking directly at the display.

4. All of the raw values and the derived data are written to disk at the end of each trial.

The key subroutines—TESTSUCK, CALIBRATE, and SUCK—are included in Appendices A, B, and C. SUCK samples the pressure data, plots data, and stores data on disk. CALIBRATE establishes conversion to millimeters of pressure from analog to digital units. TESTSUCK determines whether a suck has occurred.

Program Availability

These subroutines are available by mail. Send a double-sided, double-density 3.5- or 5.25-in. diskette and a re-

usable diskette mailer to Donald C. Martin, Department of Biostatistics, SC-32, University of Washington, Seattle, WA 98195.

REFERENCES

- COWETT, R. M., LIPSITT, L. P., VOHR, B., & OH, W. (1978). Aberrations in sucking behavior of low-birthweight infants. *Developmental Medicine & Child Neurology*, **20**, 701-709.
- KRON, R. E., LITT, M., PHOENIX, M. D., & FINNEGAN, L. P. (1976). Neonatal narcotic abstinence: Effects of pharmacotherapeutic agents and maternal drug usage on nutritive sucking behavior. *Journal of Pediatrics*, **88**, 637-641.
- KRON, R. E., STEIN, M., & GODDARD, K. E. (1963). A method of measuring sucking behavior of newborn infants. *Psychosomatic Medicine*, **25**, 181-191.
- LIPSITT, L. P. (1979). Critical conditions in infancy: A psychological perspective. *American Psychologist*, **34**, 973-980.
- MARTIN, D. C., MARTIN, J. C., & STREISSGUTH, A. P. (1979). Sucking frequency and amplitude in newborns as a function of maternal smoking and drinking. In V. M. Galanter (Ed.), *Currents in alcoholism* (Vol. 5, pp. 359-366). New York: Grune & Stratton.
- MARTIN, J. C., MARTIN, D. C., LUND, C. A., & STREISSGUTH, A. P. (1977). Maternal alcohol ingestion and cigarette smoking and their effects on newborn conditioning. *Alcoholism: Clinical & Experimental Research*, **1**, 243-247.
- STOCK, D. L., STREISSGUTH, A. P., & MARTIN, D. C. (1985). Neonatal sucking as an outcome variable: Comparison of quantitative and clinical assessments. *Early Human Development*, **10**, 273-278.

NOTE

1. Borland's TurboBasic is now distributed as PowerBASIC through Spectra Publishing, 1030D East Duane Ave., Sunnyvale, CA 94086.

Appendix A TESTSUCK

```

* TESTSUCK.BAS
* Borland Turbo BASIC
* Compiled and run from this listing 6/4/90
* Subroutine to test state of suck. To use:
*
* CALL TESTSUCK (P!,NP,STATE)
*
* P! Pressure (in -MM HG).
* NP Point number in trial.
* STATE Returned as 1 during suck, 0 between sucks.
*
defint a-z

sub testsuck (p!,np,state)

static baseline!,test1,test2!,alpha!

v! = p!: if p! < 0 then v! = 0

if np = 1 then
    test1 = 10      ' Threshold above baseline for suck.
    dtest2 = 0     ' Threshold above test1 for end-of-suck.
    alpha! = 0.1   ' Parameter for updating baseline.
    baseline! = v! ' Floating baseline.
    state = 0
    exit sub
end if

```

Appendix A (Continued)

```

d! = v!-baseline!
if state = 0 then
  if d! > test1 then
    test2! = d!+dtest2: state = 1
  else
    baseline! = baseline!*(1-alpha!) + v!*alpha!
  end if
else
  if d! < test2! then
    state = 0
    baseline! = v!*(1-alpha!) + baseline!*alpha!
  end if
end if
end sub

```

Appendix B
SUCK

```

' DATA-SAMPLING ROUTINE DURING BABY SUCKING.
' by David Borg-Breen, modified by DCM 11/17/89

DEFINT A-Z

DIM AVPRESSURE!(15),SUCKCOUNT(15),LATENCY!(15)
DIM PSAVE!(600)
DIM D(10),TSUCK!(50),adend(50),tsend!(50)

babyid$ = "12345" 'set up fake id for test runs
open "junk.jnk" for output as #1
examline = 18
locate 10,1

SUCK:
PRINT USING " ##",EXAMLIN;
PRINT ". Sucking results:"+SPACE$(50);
GOSUB EXAMTIME
' set up output file for pressures
SUCKFILE$ = "B_" +BABYID$+".SUK"
OPEN SUCKFILE$ FOR OUTPUT AS #2

CALL LOAD_DAS8 ' LOAD DAS8 DRIVER. SEE DAS8 HANDBOOK

'
' INITIALIZE TO SAMPLE FROM CHANNEL 0.
' SET FOR ONE CHANNEL AND GET ERROR FLAG
'
MODE = 0: BASADR% = &H300: FLAG = 0
CALL DAS8 (MODE,BASADR%,FLAG)
IF FLAG <> 0 THEN
  PRINT "ERROR INITIALIZING DAS8 BOARD.": END
END IF
MODE = 1: D(0) = 0: D(1) = 0: FLAG = 0
CALL DAS8 (MODE,D,FLAG)
IF FLAG <> 0 THEN
  PRINT "ERROR SETTING DAS8 CHANNEL.": END
END IF
'
' SET GAIN RANGE TO +/-5 VOLTS.
'
MODE = 19: IGAIN% = 8: FLAG = 0
CALL DAS8 (MODE,IGAIN%,FLAG)
IF FLAG <> 0 THEN PRINT "ERROR SETTING GAIN.": END

```

Appendix B (Continued)

```

VLOW! = -5.0: VHI = +5.0
IADLOW = -1200: IADHI = 300
ADPLOT RANGE! = IADHI - IADLOW

,
, SET UP PLOT.
,
IX0 = 20: IX1 = 590
IY0 = 140: IY1 = 10

SCREEN 2
LINE (IX0,IY0) - (IX0,IY1)
LINE (IX1,IY0) - (IX1,IY1)
LINE (IX0,IY0) - (IX1,IY0)
LINE (IX0,IY1) - (IX1,IY1)
YFACT! = (IY1-IY0)/ADPLOT RANGE!

GOSUB CALIBRATE
IF IAD0MM = IAD50MM THEN IAD50MM = IAD0MM - 100

A! = 50.0/(IAD50MM-IAD0MM)
B! = -50.0*IAD0MM/(IAD50MM-IAD0MM)

FOR NTRIAL = 1 TO 15

CLS
LINE (IX0,IY0) - (IX0,IY1)
LINE (IX1,IY0) - (IX1,IY1)
LINE (IX0,IY0) - (IX1,IY0)
LINE (IX0,IY1) - (IX1,IY1)

IY = IY0 + (IAD50MM-IADLOW)*YFACT!
FOR IX = IX0 TO IX1 STEP 6: PSET (IX,IY): NEXT IX
IY = IY0 + (IAD0MM-IADLOW)*YFACT!
FOR IX = IX0 TO IX1 STEP 6: PSET (IX,IY): NEXT IX

LOCATE 23,5

MSG01$=" (or Q, then [ENTER] to abort sucking trials.)"
PRINT " "+MSG01$;
LOCATE 21,9
PRINT "Press [ENTER] to begin trial";
PRINT USING " ##: ";NTRIAL;
INPUT ,RESP$;
IF RESP$ = "Q" OR RESP$ = "q" THEN GOTO OUTSUCK
LOCATE 21,1: PRINT SPACE$(60)
PRINT SPACE$(60): PRINT SPACE$(60)

T0! = TIMER
PT! = TIMER: WHILE PT!<=T0!: PT! = TIMER: WEND
T! = PT!
NPOINT = 0

WHILE T!-T0! <= 20.0

NAVG = 0
ADTOT& = 0
DO
'GET ONE SAMPLE POINT (MODE = 4)
MODE = 4: FLAG = 0
CALL DAS8 (MODE,IAD,FLAG)
IF FLAG <> 0 THEN PRINT "ERROR SAMPLING DATA."
NAVG = NAVG+1
ADTOT& = ADTOT&+IAD
T! = TIMER

```

Appendix B (Continued)

```

LOOP UNTIL T! <> PT!
PT! = T!
IAD = ADTOT&/NAVG
IY = IY0 + (IAD-IADLOW)*YFACT!
IX = IX0 + (IX1-IX0)*(T!-T0!)/20.0
NPOINT = NPOINT+1
IF NPOINT = 1 THEN PSET(IX,IY)
LINE -(IX,IY)

PRESSURE! = A!*IAD + B!
PSAVE!(NPOINT) = PRESSURE!

WEND

TSUK1! = T!-T0!
DT! = (T!-T0!)/NPOINT
NSUCKS = 0
ISTATEPREV = 0
FOR I = 1 TO NPOINT
  CALL TESTSUCK (PSAVE!(I),I,ISTATE)
  IF ISTATE = 1 AND ISTATEPREV = 0 THEN
    NSUCKS = NSUCKS+1
    IF NSUCKS = 1 THEN TSUK1! = I*DT!
  END IF
  ISTATEPREV = ISTATE
NEXT I

LASTTRIAL = NTRIAL
TP! = 0
FOR I = 1 TO NPOINT: TP!=TP!+PSAVE!(I): NEXT I
AVPRESSURE!(NTRIAL) = TP!/NPOINT
SUCKCOUNT(NTRIAL) = NSUCKS
LATENCY!(NTRIAL) = TSUK1!

LOCATE 22,1
PRINT " Trial:"; PRINT USING " ##";NTRIAL;
PRINT " Avg pressure:";
PRINT USING "####.#";AVPRESSURE!(NTRIAL);
PRINT " # Sucks:";
PRINT USING "####";SUCKCOUNT(NTRIAL);
PRINT " Latency:";
PRINT USING "###.##";LATENCY!(NTRIAL);
GOSUB GRABLINE

SCREEN 0: COLOR 0,7: CLS

PRINT #2, NTRIAL ", " NPOINT ", " T!-T0!
FOR I = 1 TO NPOINT STEP 10
FOR J = I TO I+9
  IF J <= NPOINT THEN PRINT #2, USING "###.##";PSAVE!(J);
  IF J < (I+9) AND J < NPOINT THEN PRINT #2, ", ";
NEXT J
PRINT #2, ""
NEXT I

DELAY 1.0
SCREEN 2

NEXT NTRIAL

```

Appendix B (Continued)

OUTSUCK:

```
PRINT #2, 999 ", " 0 ", " 0.0
CLOSE #2
```

```
SCREEN 0: COLOR 0,7: CLS
```

```
END
,
```

**Appendix C
CALIBRATE**

```
, Calibrate A/D data using 50 mm calibration level.
,
```

```
DIM D%(10)
CALIBRATE:
```

```
TPLOT! = 5*60      ' TOTAL PLOT TIME.
PLOTTIME0! = TIMER
NPT& = 0
ISTATE = 1
IAD0MM = 19999
TRUE = 1
```

```
LOCATE 25,4
PRINT "Calibration can be restarted by typing R";
```

```
WHILE TRUE
```

```
CHAR$ = INKEY$
IF CHAR$ = "R" THEN CLS: GOTO CALIBRATE
IF LEN(CHAR$) > 0 AND ASC(CHAR$) = 13 THEN
  ENTERKEY = 1
ELSE
  ENTERKEY = 0
END IF
```

```
SELECT CASE ISTATE
```

```
CASE 1
```

```
LOCATE 21,4
PRINT "Dial calibration to -50 and press [ENTER]: ";
ISTATE = 2
```

```
CASE 2
```

```
IF ENTERKEY THEN      ' CHECK IF [ENTER] KEY TYPED.
  IAD50MM = IAD      ' SAVE A/D VALUE FOR 50 MM.
  JY = IY0 + (IAD-IADLOW)*YFACT!
  FOR JX = IX0 TO IX1 STEP 6
    ' PLOT LINE AT -50 MM LEVEL.
    PSET (JX,JY)
  NEXT JX
  PSET (IX,IY)
  LOCATE 22,4
  PRINT "Dial calibration to 0 and press [ENTER]: ";
  ISTATE = 3
```

```
END IF
```

```
CASE 3
```

```
IF ENTERKEY THEN      ' CHECK IF [ENTER] KEY TYPED.
  IAD0MM = IAD      ' SAVE A/D VALUE FOR 0 MM.
  JY = IY0 + (IAD-IADLOW)*YFACT!
  FOR JX = IX0 TO IX1 STEP 6
    ' PLOT LINE AT 0 MM LEVEL.
    PSET (JX,JY)
```

Appendix C (Continued)

```

NEXT JX
PSET (IX,IY)
LOCATE 21,1: PRINT SPACES$(60): PRINT SPACES$(60)
LOCATE 23,4
PRINT "Press [ENTER] to proceed to trial 1. ";
ISTATE = 4
END IF
CASE 4
IF ENTERKEY THEN RETURN
END SELECT
,
, SAMPLE POINT AND PLOT IT.
,
NPT& = NPT&+1
MODE = 4: FLAG = 0
CALL DAS$(MODE,IAD,FLAG)
T! = TIMER - PLOTTIME0!
IF FLAG <> 0 THEN PRINT "ERROR SAMPLING DATA.": END
IY = IY0 + (IAD-IADLOW)*YFACT!
IX = IX0 + (IX1-IX0)*T!/TPLOT!
IF NPT& = 1 THEN PSET(IX,IY)
LINE -(IX,IY)

IF T!-PLOTTIME0! > TPLOT! THEN GOTO QUITCAL

WEND

QUITCAL:

LOCATE 23,4
PRINT "Sucking trial complete. Press [ENTER]: ";
INPUT "",R$
IF R$ = "R" OR R$ = "r" THEN GOTO CALIBRATE

RETURN
END

EXAMTIME:
EXAMTIMES$ = LEFT$(TIMES$,5)
SAVELINE = CSRLIN
GOSUB GRABLINE
'Scroll up if necessary to allow at least 6
'lines at the bottom of the screen.
LOCATE SAVELINE+1,1
FOR N = 1 TO 6: PRINT SPACES$(79): NEXT N
LOCATE SAVELINE+1,1
IF SAVELINE = 18 THEN LOCATE SAVELINE,1
EXAMLIN$ = EXAMLIN$+1
RETURN
END

GRABLINE:
LOCATE CSRLIN,75: PRINT EXAMTIMES$;
NCRT = NCRT+1
CRTLIN$(NCRT) = MID$(" ",1,0)
FOR NC = 1 TO 79
ASCII$ = SCREEN(CSRLIN,NC)
CRTLIN$(NCRT) = CRTLIN$(NCRT)+CHR$(ASCII$)
NEXT NC

```

Appendix C (Continued)

```
PRINT #1, CRTLINE$(NCRT)
RETURN
END
```

```
$INCLUDE "DAS8.INC"
' Use if subroutines are in separate files
'$INCLUDE "TESTSUCK.BAS"
'$INCLUDE "CALIBRATE.BAS"
```

(Manuscript received January 22, 1990;
revision accepted for publication June 13, 1990.)