```
Screen:    0
 0 11
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:    1
 0 ( DISK 5"/8" DISK READ/WRITE FUNCTIONS )   HEX
 1
 2 : (WRITE) SELECT DCMD [ 0A00 , ] ?ERROR ;
 3 : (BUFFER) DISK GET OBTAIN (WRITE) DISK RELEASE ;
 4
 5 : (READ) SELECT DCMD [ 0800 , ] ?ERROR ;
 6 : (BLOCK)  OFFSET @ + PAUSE CORE BUFFER DISK GRAB
 7    OVER (READ)  SWAP IDENTIFY DISK RELEASE ;   DECIMAL
 8
 9 ' (BLOCK) 'BLOCK !
10 ' (BUFFER) 'BUFFER !
11
12
13
14
15

Screen:    2
 0 ( N-BUFFER MANAGER )  DECIMAL
 1 CODE CORE   W POP   W SHL   I 2 MOV   PREV 4 + # I MOV
 2    NB 1 MOV   BEGIN   LODS   0 SHL   W 0 CMP   -LOOP
 3       2 I MOV   0= NOT IF   W SHR   W PUSH   NEXT
 4    THEN   1 COM   NB 1 ADD   1 SHL
 5 BEGIN    1 PREV MOV   1 1 HI XCHG B   1 SHL
 6    FIRST 1 ADD   1 PUSH   ' EXIT JMP
 7
 8 CODE OBTAIN   PREV 2+ 1 MOV   BEGIN   1 SHR   1 INC   NB 1 CMP
 9       0= IF   1 1 SUB   THEN   1 SHL   PREV 1 CMP   0= NOT END
10    1 PREV 2+ MOV   1 W MOV   32767 # 0 MOV
11    PREV 4 + W) 0 XCHG   0 SHL   CS IF   0 SHR
12       1 PREV MOV   1 1 HI XCHG B   1 SHL   FIRST 1 ADD
13       1 PUSH   0 PUSH   'BUFFER 2+ STA   NEXT
14    THEN   0 # DISK MOV   JMP
15
```

```
Screen:    3
 0 ( SELECT LUN AND LOGIVCAL SECTOR FOR 5" WINI 8 " FLOPPY )
 1 DECIMAL
 2 CODE SELECT ( address blk# )    IOPB # W MOV   0 POP
 3    10097 # 0 CMP 0> IF   8 #B 4 W) MOV   32 #B 1 W) MOV
 4       10098 # 0 SUB
 5       0 SHL   0 SHL   0 SHL
 6                 ELSE   4 #B 4 W) MOV   0 #B 1 W) MOV
 7    0 SHL   0 SHL   THEN   0 0 HI XCHG B   0 2 W) MOV
 8    0 POP   0 PUSH   0 6 W) MOV   NEXT
 9
10
11
12
13
14
15

Screen:    4
 0 ( DISK 5"/8" DISK READ/WRITE FUNCTIONS )   HEX
 1
 2 : (WRITE) SELECT DCMD [ 0A00 , ] ?ERROR ;
 3 : (BUFFER) DISK GET OBTAIN (WRITE) DISK RELEASE ;
 4
 5 : (READ) SELECT DCMD [ 0800 , ] ?ERROR ;
 6 : (BLOCK)  OFFSET @ + PAUSE CORE BUFFER DISK GRAB
 7    OVER (READ)  SWAP IDENTIFY DISK RELEASE ;   DECIMAL
 8
 9 ' (BLOCK) 'BLOCK !
10 ' (BUFFER) 'BUFFER !
11
12
13
14
15

Screen:    5
 0 ( PATCH FOR 8 INCH FLOPPY )
 1 HEX
 2 CREATE SA800-INIT 0 C, 8 C, 4C C, 0F C, 0 , 0 C, C0 C,
 3   C0 C, 0 C,
 4
 5 : 8INCH-INIT IOPB 0A ERASE S0 @ IOPB 6 + !
 6   20 IOPB 1+ C! DCMD [ C000 , ]
 7 SA800-INIT S0 @ 0A MOVE DCMD [ C200 , ] ?ERROR ;
 8
 9 : LUN IOPB 0A ERASE 20 IOPB 1+ C! ;
10
11 : ZAP8 IOPB 0A ERASE 20 IOPB 1+ C! DCMD [ 0400 , ] ?ERROR ;
12
13 : SEEK LUN DUP 100 / IOPB 2+ C! IOPB 3 + C! DCMD [ 0B00 , ]
14    ?ERROR ;
15 DECIMAL
```

```
Screen:    6
 0 ( 8" DISK PATCHES )
 1
 2 5 LOAD
 3 8INCH-INIT
 4 8INCH-INIT
 5 2 LOAD 3 LOAD 4 LOAD
 6
 7
 8
 9
10
11
12
13
14
15

Screen:    7
 0 ( UTILITIES FOR COPYING FLOPPIES )
 1
 2 : BEEP 7 EMIT ;
 3
 4 : SUCK ( drive# _ read from floppy and store in drive # )
 5    0 DRIVE  250 * 250 0 DO I . CR DUP I + 10098 I +
 6    SWAP COPY UPDATE LOOP DROP FLUSH BEEP BEEP ;
 7
 8 : BELCH ( drive# - transfer from drive# to floppy )
 9    0 DRIVE 250 * 250 0 DO I . CR DUP I + 10098 I +
10    COPY UPDATE LOOP DROP FLUSH BEEP BEEP ;
11
12
13
14
15

Screen:    8
 0 ( INITIALIZATION OFOR DOUBLE SIDED 5 1/4" DISK ) HEX
 1
 2 CREATE TANDON 0 C, 7 C, 4F C, 16 C, 0 , 0 C, 80 C, 0 ,
 3
 4 : 5INCH-INIT IOPB 0A ERASE S0 @ IOPB 6 + !
 5    20 IOPB 1+ C! DCMD [ C087 , ]
 6    TANDON S0 @ 0A MOVE DCMD [ C200 , ] ;
 7
 8 : FORMAT IOPB 0A ERASE 20 IOPB 1+ C! DCMD [ 0400 , ] ?ERROR ;
 9 DECIMAL
10
11
12
13
14
15
```

```
Screen:    9
 0 ( SYSTEM DEFINITION)
 1 : KEY   CNT @  0 'S 1 EXPECT  SWAP CNT ! ;
 2 : EMIT   'S 1 TYPE DROP ;
 3 HERE ]  ABORT" TRAPPED" [   ASSEMBLER
 4    BEGIN   SWAP # I MOV   STI   NEXT   15 INTERRUPT
 5    BEGIN   0 0 SUB   CWD   IRET   0 INTERRUPT
 6
 7 ( DICTIONARY)  10 LOAD   ( 32-BIT)  30 LOAD  31 LOAD
 8    32 LOAD  33 LOAD   ( TASKS)  24 LOAD  ( 27 LOAD)
 9 ( EXTENTIONS)  36 LOAD  37 LOAD  38 LOAD
10 ( TIME)  57 LOAD  8320 CONSTANT TODAY  58 LOAD
11    : NOW   TODAY ! ;      SPACE TODAY @ .DATE  SPACE TIME
12
13 CONTEXT GOLDEN 20 MOVE  HERE H 2+ !
14    ( SECOND PROMPT)
15

Screen:   10
 0    ( VOCABULARIES)
 1 : IMMEDIATE   128  LAST @ @ +! ;
 2
 3 : DEFINITIONS   CONTEXT @ CURRENT ! ;
 4
 5 : FORGET   ' 8 -  DUP H !  1-  CURRENT CONTEXT 2+ DO
 6    I @  BEGIN  2DUP < IF  2+ 2+ @ AGAIN  I !  2 /LOOP DROP ;
 7
 8 : ATTACH   CONTEXT 2+ 16 MOVE ;
 9
10 39 CONSTANT DISKING   42 CONSTANT PRINTING
11 90 CONSTANT COMPILER
12
13
14
15

Screen:   11
 0    ( CHARACTER OPERATORS)
 1 | CODE -MATCH   ( D D# S S# -- T D')
 2    2 POP   0 POP   1 POP   W POP   U PUSH   I PUSH
 3    3 0 XCHG   3 ) 0 MOV B   2 DEC   2 1 SUB   0> IF   3 INC
 4    BEGIN   REP B SCAS B   0= IF   ( 1ST MATCH)   1 PUSH
 5          W PUSH   3 I MOV   2 1 MOV   REP CMPS B   0= NOT IF
 6             ( NO MATCH)   W POP   1 POP   ROT JMP   THEN
 7       ( MATCH)   0 POP   0 POP   0 0 SUB   0 2 MOV   THEN THEN
 8    I POP   U POP   0 PUSH   2 1 ADD   1 W ADD   W PUSH   NEXT
 9
10 CODE <CMOVE   I 0 XCHG   1 POP   W POP   I POP
11    1 W ADD   W DEC   1 I ADD   I DEC   STD
12    REP MOVS B   STD B   I 0 XCHG   NEXT
13
14
15
```

```
Screen:   12
 0 ( 8086 ASSEMBLER)   HEX
 1  0 CONSTANT ES   1 CONSTANT IS   2 CONSTANT SS   3 CONSTANT DS
 2  3 CONSTANT U    4 CONSTANT S    5 CONSTANT R    6 CONSTANT I CR
 3  7 CONSTANT W                   10 CONSTANT #   11 CONSTANT #B
 4 0C CONSTANT I)  0D CONSTANT W)  0E CONSTANT R)  0F CONSTANT 3)
 5               8008 CONSTANT (2)                               CR
 6 | : INST!   HERE PTR ! ;         | : ?#   OVER 2/ 8 = ;
 7 | : FIX   PTR @ 1+ +! ;          | : ?R   -8 AND 0= ;
 8 | : MICRO   PTR @ +! ;
 9                                                               CR
10 : B   -1 MICRO ;      : -B   1 MICRO ;
11 : V   2 MICRO ;       : HI   2+ 2+ ;
12
13 | : ,LIT   1 AND IF  C, B  ELSE ,  THEN ;
14 | : ,-LIT   1 AND IF  C, B  ELSE
15    ?CELL IF  ,  ELSE C, V  THEN THEN ;      DECIMAL

Screen:   13
 0    ( ADDRESSING MODES)   HEX
 1 | : R/M   DUP -10 AND IF  , 6  ELSE  DUP ?R IF  C0 +
 2    ELSE 8 -  SWAP OVER 6 =  OVER OR IF
 3       ?CELL IF  , 80 +  ELSE C, 40 +  THEN
 4    ELSE DROP  THEN THEN THEN  FIX ;
 5
 6 | : IMM   DUP ?R IF  ?# IF  R/M 1
 7       ELSE 8 * FIX  R/M 0  THEN
 8    ELSE R/M  DUP ?R IF  8 * FIX  B B 0
 9       ELSE 1  THEN THEN ;
10
11 FORTH : RM#   USER  DOES> C@  INST!  C,
12    ?# OVER 0= AND IF  DROP  V  ,LIT
13    ELSE  0 C,  IMM IF  PTR @ C@  81 OVER - MICRO
14       38 AND FIX  ,-LIT  THEN THEN ;   DECIMAL
15

Screen:   14
 0    ( INSTRUCTION TYPES)
 1 FORTH : RMR   2CONSTANT  DOES> 2@  INST!
 2    ROT DUP ?R IF  ROT + C,  DROP
 3       ELSE SWAP ,  SWAP DROP R/M  THEN ;
 4
 5 FORTH : RM   CONSTANT  DOES> @  INST!  ,  R/M ;
 6
 7 : OPC   USER  ;CODE   FORTH
 8 : OPC   USER  DOES> C@  INST!  C, ;
 9
10 FORTH : MA   USER  DOES> C@  INST!  C, , ;
11
12 FORTH : REG   USER  DOES> C@  SWAP 8 *  + C, ;
13
14 FORTH : I/O   USER  DOES> C@  INST!  C,
15    DUP 0< IF  MICRO  ELSE C,  THEN ;
```

```
Screen:   15
 0    ( INSTRUCTIONS)   HEX
 1 40 00FF RMR INC    48 08FF RMR DEC      A1 MA LDA
 2 50 30FF RMR PUSH   58 008F RMR POP      A3 MA STA            CR
 3 03 RM# ADD   0B RM# OR    13 RM# ADC   1B RM# SBB
 4              2B RM# SUB                3B RM# CMP           CR
 5 C4 RM# LES   C5 RM# LDS   8C RM# SSG   8E RM# LSG
 6 8D RM# LEA                     28FF RM LIS    18FF RM PIS     CR
 7 10F7 RM COM    18F7 RM NEG    20F7 RM MUL    28F7 RM IMUL
 8 30F7 RM DIV    38F7 RM IDIV   20FF RM LIP    10FF RM PIP     CR
 9 00D1 RM ROL    08D1 RM ROR    10D1 RM RCL    18D1 RM RCR
10 20D1 RM SHL    28D1 RM SHR                   38D1 RM SAR     CR
11 E4 I/O IN     E6 I/O OUT    C3 OPC RET    CF OPC IRET
12 F3 OPC REP    A5 OPC MOVS   A7 OPC CMPS   AB OPC STOS        CR
13 0AD OPC LODS  AF OPC SCAS
14 FB OPC STI    FD OPC STD    99 OPC CWD
15 26 REG SEG    07 REG POPS   06 REG PUSHS      DECIMAL

Screen:   16
 0    ( INSTRUCTIONS)   HEX
 1 : MOV   INST!  ?# OVER ?R AND IF  B8 + C,
 2       1 AND IF  C,  -8 MICRO  ELSE ,  THEN
 3    ELSE 8B ,  IMM IF  3C MICRO  ,LIT  THEN THEN ;
 4
 5 : XCHG   INST!  OVER ?R  OVER 0= AND IF  DROP  90 + C,
 6    ELSE 87 C,  8 * C,  R/M  THEN ;
 7
 8 : TEST   INST!  ?# OVER 0= AND IF  DROP  A9 C,  ,LIT
 9    ELSE 85 ,  IMM IF  72 MICRO  ,LIT  THEN THEN ;
10
11 : NEXT   LODS   W 0 XCHG   W ) LIP ;   ( 39)                 CR
12 : NOT   1 XOR ;
13
14 23 RM# AND               33 RM# XOR            DECIMAL
15

Screen:   17
 0    ( STRUCTURES)   HEX
 1 72 CONSTANT CS   74 CONSTANT 0=
 2 78 CONSTANT 0<   7F CONSTANT 0>   E2 CONSTANT 1NZ
 3
 4 : JMP   INST!  E9 C,  HERE 1+ -
 5    ?CELL IF  1- ,  ELSE C, V  THEN ;
 6 : CALL   E8 C,  HERE 2+ - , ;
 7                                                              CR
 8 : IF   NOT ,  HERE ;
 9 : THEN   HERE OVER -  SWAP 1- C! ;
10 : ELSE   EA IF  SWAP THEN ;
11
12 : END   INST!  NOT C,  HERE 1+ - C, ;
13 : LOOP   E3 END ;      : -LOOP   E1 END ;
14
15    DECIMAL
```

```
Screen:   18
 0 ( EDITOR)
 1 | : #I   PAD 65 + ;      | : #F   PAD 130 + ;
 2
 3 | : STRING   ( S -- S 1+)  >R  94 WORD  C@ IF  I 65 BLANK
 4       HERE I OVER C@ 1+ MOVE  THEN R> 1+ ;
 5 | : AT   ( -- D D#)  SCR @ BLOCK  R# @ +  64  R# @  63 AND - ;
 6 | : AT0   R# @  -64 AND  R# !  AT ;
 7                                                              CR
 8 | : DELETE   ( D D# S#)  OVER MIN >R  I -  2DUP OVER I +
 9       ROT ROT MOVE  + R> BLANK  UPDATE ;
10 | : INSERT   ( S S# D D#)  ROT  OVER MIN >R  I -
11       OVER DUP I + ROT <CMOVE  R> MOVE  UPDATE ;
12
13 : -TRAILING   BEGIN 1-  2DUP + C@  32 -  OVER 0< + END  1+ ;
14 : >TYPE   >R  PAD I MOVE  PAD R> TYPE ;
15

Screen:   19
 0    ( LINE EDITOR)
 1 : LINE   CR SPACE  AT 64 SWAP -  >R I -  R> >TYPE
 2       ." ^"  AT >TYPE  R# @ 64 /MOD . DROP  EDITOR ;
 3 | : HOLD   AT0 #I  2DUP C!  1+ SWAP MOVE ;
 4
 5 : T   15 AND  64 * R# !  LINE ;
 6 : P   #I STRING  AT0 MOVE  UPDATE ;
 7 : X   HOLD  AT0  1024 R# @ -  SWAP DELETE ;                  CR
 8 : U   #I STRING  64 R# +!  AT0 SWAP  1024 R# @ -  INSERT ;
 9 : M   HOLD  SCR 2@ >R  64 + >R  64 * SWAP SCR 2!  U
10       R> R> SCR 2! ;
11
12 : LIST   PAGE  16 0 DO  CR I 2 U.R SPACE  DUP BLOCK
13       I 64 * +  64 -TRAILING >TYPE  LOOP SPACE  SCR ! ;
14 : L   SCR @  DUP .  LIST ;
15 : COPY   SWAP BLOCK DROP  OFFSET @ + IDENTIFY  UPDATE ;

Screen:   20
 0    ( STRING EDITOR)
 1 : TOP   0 R# ! ;      : N   1 SCR +! ;      : B   -1 SCR +! ;
 2
 3 | : -FOUND   #F STRING DROP  BEGIN  AT  OVER >R  #F COUNT -MATCH
 4       R> - R# +!  R# @ 1024 =  OVER 0= +  NOT IF  DROP AGAIN ;
 5 | : SEARCH   -FOUND IF  TOP
 6       #F HERE 65 MOVE  1 ABORT" NONE"  THEN ;
 7
 8 : TILL   R# @  SEARCH  R# @ SWAP  DUP R# !
 9       -  AT ROT DELETE  LINE ;                               CR
10 : E   #F C@  DUP MINUS R# +!  AT ROT DELETE  LINE ;
11 : I   #I STRING  #I C@  AT INSERT  #I C@ R# +!  LINE ;
12 : D   SEARCH E ;      : F   SEARCH LINE ;      : R   E I ;
13
14 : S   DUP SCR @ DO  -FOUND IF  N TOP
15       ELSE  LINE SCR ?  LEAVE  DUP THEN  LOOP DROP ;
```

```
Screen:   21
 0 88
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   22
 0 88
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   23
 0 88
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:   24
 0 ( TASK DEFINITION)
 1 : BACKGROUND   ( #S #R #U)  CREATE  >R  OVER +
 2       DUP HERE 1+ +  15 AND 15 XOR  DUP D+
 3    DUP HERE 2+ + ,  SWAP HERE + ,  I C,  R> + 3 - ALLOT ;
 4 : BUILD   OPERATOR @  OVER DUP @  SWAP 2+ 2+ C@ MOVE
 5       2@  DUP OPERATOR @ 2+ !  8 + ! ;
 6
 7 : TERMINAL   CREATE  25 LOAD ;      ' 'CR C@
 8 : CONSTRUCT   DUP BUILD  DUP 5 +  SWAP @ IN-LINE +  16 MOVE ;
 9
10 : PROMPT   ACTIVATE  EMPTY  PAGE ." up" CR  0 QUIT [
11       ' OFFSET C@   ' S0 C@
12 : SEND   DUP @   S0 @  OVER IN-LINE + @  80 MOVE
13    OFFSET  SWAP IN-LINE +  66 MOVE  >IN @ CNT !
14       ACTIVATE  0 INTERPRET  QUIT [
15

Screen:   25
 0    ( DEFINE TERMINAL TASK)   FORTH
 1 BASE @ >R  DECIMAL   HERE 2+ >R
 2       DUP I + 1-  15 AND 15 XOR +   ( MOD 16 ADJUSTMENT)
 3    DUP I +  DUP 128 + ,  ,  96 C,
 4    ' (CR) ,  ' (PAGE) ,  ' (TYPE) ,  ROT ,  SWAP ,
 5    ' (EXPECT) ,  I ,  R> ,   224 + ALLOT   R> BASE !
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   26
 0 99
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:   27
 0 ( SYSTEM TASKS)   HEX
 1 C8 CA 800 TERMINAL SECOND
 2 ASSEMBLER   <TYPE> 8 INTERRUPT  SECOND @ 8 BIAS
 3    <EXPECT> 9 INTERRUPT  SECOND @ 9 BIAS   FORTH
 4
 5 DECIMAL   SECOND CONSTRUCT
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   28
 0 1010
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   29
 0 1010
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:   30
 0 ( 32-BIT OPERATORS)
 1 : 2CONSTANT   CONSTANT ,  ;CODE
 2       4 W) PUSH   2 W) PUSH   NEXT
 3 0 0 2CONSTANT 0.
 4
 5 CODE 2SWAP   2 POP   W POP   0 POP   1 POP
 6    W PUSH   2 PUSH   1 PUSH   0 PUSH   NEXT
 7 CODE 2OVER   S W MOV   6 W) PUSH   4 W) PUSH   NEXT
 8
 9
10
11
12
13
14
15

Screen:   31
 0    ( 32-BIT ARITHMETIC)
 1 CODE D+   2 POP   0 POP   S W MOV   0 2 W) ADD
 2    2 W ) ADC   NEXT
 3 CODE DMINUS   2 POP   0 POP   0 NEG   0 # 2 ADC   2 NEG
 4    0 PUSH   2 PUSH   NEXT
 5 : D-   DMINUS D+ ;
 6
 7 : D0=   0= SWAP 0= AND ;
 8 : D<   D- 0<  SWAP DROP ;
 9 : DABS   DUP 0< IF  DMINUS THEN ;
10 : DMIN   2OVER D-  DUP 0< IF  D+  ELSE 2DROP  THEN ;
11 : DMAX   2OVER 2OVER DMIN  D- D+ ;
12
13
14
15

Screen:   32
 0    ( MULTIPLY, DIVIDE)
 1 CODE MOD   W POP   2 2 SUB   0 POP   W DIV   2 PUSH   NEXT
 2 CODE */MOD   W POP   2 POP   0 POP   2 MUL   W DIV
 3    2 PUSH   0 PUSH   NEXT
 4 CODE */   W POP   2 POP   0 POP   2 IMUL   W IDIV
 5    0 PUSH   NEXT
 6 CODE /   W POP   0 POP   CWD   W IDIV   0 PUSH   NEXT
 7
 8 CODE M*   0 POP   2 POP   2 IMUL   0 PUSH   2 PUSH   NEXT
 9 CODE M/   W POP   2 POP   0 POP   W IDIV   0 PUSH   NEXT
10
11 CODE U*   0 POP   2 POP   2 MUL   0 PUSH   2 PUSH   NEXT
12 CODE M/MOD   W POP   2 POP   0 POP   W DIV
13    2 PUSH   0 PUSH   NEXT
14
15
```

```
Screen:   33
 0    ( MIXED ARITHMETIC)
 1 CODE M+   0 POP   CWD   ' D+ 2+ JMP
 2
 3 CODE T*   I -2 R) MOV   U -4 R) MOV   U POP   I POP   W POP
 4    W 0 MOV   U MUL   0 PUSH   2 1 MOV   I 0 MOV   U MUL
 5    0 1 ADD   0 # 2 ADC   I I OR   0< IF   U 2 SUB   THEN
 6       U U OR   0< IF   W 1 SUB   I 2 SBB   THEN
 7    1 PUSH   2 PUSH   -4 R) U MOV   -2 R) I MOV   NEXT
 8
 9 CODE T/   W POP   2 POP   0 POP   1 POP   W W OR   156 C,
10    0< IF   W NEG   THEN   2 2 OR   0< IF   W 2 ADD   THEN
11    W DIV   1 0 XCHG   W DIV   157 C,   0< IF   0 NEG
12       0 # 1 ADC   1 NEG   THEN   0 PUSH   1 PUSH   NEXT
13
14 : M*/   >R T*  R> T/ ;
15

Screen:   34
 0 ( CHARACTER OPERATORS)
 1 CODE -TEXT   I 0 XCHG   W POP   1 POP   I POP   1 SHR
 2       REP CMPS   0= NOT IF   1 # 1 MOV   CS IF   1 NEG
 3    THEN   THEN   1 PUSH   I 0 XCHG   NEXT
 4
 5 : <TEXT   -TEXT 0< ;
 6
 7 : TEXT   PAD 72 BLANK  WORD COUNT  PAD SWAP <CMOVE ;
 8
 9
10
11
12
13
14
15

Screen:   35
 0 1212
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:   36
 0 ( POWERFUL WORDS)
 1 : (ASSIGN)   R> 2+ SWAP ! ;
 2 : ASSIGN   BEGIN  \ (ASSIGN)  \ [ 2 - @ , ] ;   IMMEDIATE
 3
 4 CODE INPUT   2 POP   0 0 SUB   (2) IN   0 PUSH   NEXT
 5 CODE OUTPUT   2 POP   0 POP   (2) OUT   NEXT
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   37
 0 ( 32-BIT OUTPUT)
 1 CODE /DIGIT   0 POP   1 POP   2 2 SUB   BASE U) DIV
 2    1 0 XCHG   BASE U) DIV   0 PUSH   1 PUSH   2 PUSH   NEXT
 3
 4 : #   /DIGIT  NUMERAL HOLD ;
 5 : #S   BEGIN  #  2DUP D0= END ;
 6
 7 : (D.)   SWAP OVER DABS  <# #S SIGN #> ;
 8 : D.   (D.) TYPE SPACE ;
 9 : D.R   >R (D.)  R> OVER - SPACES  TYPE ;
10
11
12
13
14
15

Screen:   38
 0 ( 32-BIT INPUT)
 1 CODE *DIGIT   W POP   2 POP   0 POP   1 POP   2 PUSH
 2    BASE U) MUL   1 0 XCHG   BASE U) MUL   W 0 ADD   2 1 ADC
 3    2 POP   0 PUSH   1 PUSH   2 PUSH   NEXT
 4 CODE WITHIN   2 POP   1 POP   0 POP   W W SUB
 5    2 0 CMP   0< IF   1 0 CMP   0< NOT IF   W INC
 6    THEN   THEN   W PUSH   NEXT
 7 CODE +I'   2 R) INC   NEXT
 8
 9 : (NUMBER)   0 >R  DUP 1+ C@  45 = DUP >R  +  0.
10    ROT  BEGIN BEGIN  DIGIT IF  *DIGIT AGAIN  DUP C@
11       DUP 58 =  SWAP 44 48 WITHIN  + IF  +I' AGAIN
12    C@ 32 - ABORT" ?"  R> IF DMINUS THEN  R> 0= IF DROP THEN ;
13 ' (NUMBER)  'NUMBER !
14
15
```

```
Screen:   39
 0 ( DISK UTILITY)   EMPTY   DECIMAL
 1 34 LOAD  40 LOAD  41 LOAD
 2
 3 : BLOCKS   0 DO  I NB @ MOD  0= IF FLUSH  THEN
 4      OVER I +  OVER I + COPY  LOOP 2DROP FLUSH ;
 5 : LEFT   2DUP  OVER -  OVER 250 +  SWAP BLOCKS  MATCH ;
 6 : BACKUP   0 250 LEFT ;
 7 : PROGRAM   0 60 LEFT ;
 8
 9
10
11
12
13
14
15

Screen:   40
 0    ( DISK DIAGNOSTIC)
 1 : TEST   DUP BLOCK DROP  DISK 2+ C@ ?DUP IF
 2       OVER . ." BLOCK " .  THEN DROP ;
 3 : SWEEP   SWAP DO  I TEST  LOOP ;
 4
 5 : MATCH   SWAP DO  I 250 + BLOCK  PAD 1024 MOVE
 6       PAD 1024 I BLOCK  -TEXT IF  I .  THEN LOOP ;
 7
 8
 9
10
11
12
13
14
15

Screen:   41
 0    ( FORMATTING)   HEX
 1 CREATE IOPB  80 C,  32 C,  1A C,  0 C,  1 C,  HERE 2+ ,  20 C,
 2
 3 CODE TRACK   0 POP   IOPB 3 + STA B   IOPB #B 0 MOV   E1 OUT
 4    IOPB 100 / #B 0 MOV   E2 OUT   WAIT JMP      DECIMAL
 5
 6 : INITIALIZE   DISK GET  77 0 DO  I TRACK  LOOP  DISK RELEASE ;
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:   42
 0 ( PROGRAM DOCUMENTATION)   DECIMAL   EMPTY
 1 : LFS   0 DO  CR LOOP ;
 2 : C   ."    Proprietary to FORTH, Inc.  "  TODAY @ .DATE
 3       SPACE TIME  ."      86/12-201 PROM polyFORTH" ;
 4 : LINE   BLOCK + 64  -TRAILING >TYPE ;
 5 : INDEX   SWAP  OVER SWAP DO  PAGE CR  DUP I 60 + MIN  I DO
 6       CR I 10 U.R SPACE  8 I < IF  I BLOCK @ IF
 7          0 I LINE  THEN THEN LOOP
 8       2 LFS  C CR  60 /LOOP DROP ;
 9 : TRIAD   3 / 3 *  DUP 3 + SWAP 2DUP DO
10       I BLOCK @ LOOP  + + IF  PAGE CR CR  DO  I 7 U.R  ."  LIST"
11    CR  I BLOCK @ IF  I  16 0 DO CR  I 10 U.R SPACE
12       I 64 * OVER LINE  LOOP DROP  3
13    ELSE 19  THEN LFS  LOOP  C CR  ELSE 2DROP  THEN ;
14 : SHOW   SWAP  3 / 3 * DO  I TRIAD  3 /LOOP ;   EXIT
15         pF8086/201 polyFORTH

Screen:   43
 0 1515
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   44
 0 1515
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:   45
 0 ( FILES)
 1 : SAVE   R>  R# 2@ >R >R  >R ;
 2 : RESTORE   R>  R> R> R# 2!  >R ;
 3
 4 : FILE   CONSTANT  DROP ,  DUP ,  1024 OVER / * ,  DOES> F# ! ;
 5 : (FILE)   CONSTANT  DOES> @  F# @ + ;
 6 0 (FILE) ORG   2 (FILE) LIM   4 (FILE) B/R   ( : R/B B/R 2@ / ;)
 7
 8 : SAFE   DUP LIM @ <  NOT ABORT" OUTSIDE FILE" ;
 9 : READ   SAFE R# ! ;
10 : RECORD   B/R 2@  SWAP */MOD  ORG @ +  BLOCK + ;
11
12
13
14
15

Screen:   46
 0    ( ALLOCATE RECORDS)
 1 : AVAILABLE   ORG @ BLOCK ;
 2 : CHUNK   >R  AVAILABLE @ DUP
 3    BEGIN  I +  LIM @ MOD  2DUP = ABORT" FILE FULL"
 4       DUP RECORD  DUP @ IF  DROP AGAIN
 5    DUP  B/R @ R> *  ERASE  -1 SWAP ! UPDATE
 6    DUP AVAILABLE ! UPDATE  SWAP DROP ;
 7 : SLOT   1 CHUNK ;
 8
 9 : SCRATCH   RECORD  0 SWAP !  UPDATE ;
10
11 : RECORDS   AVAILABLE @ 1+  1 ;
12
13
14
15

Screen:   47
 0    ( PRIVILEGED FILE OPERATORS)
 1 : #B   ( #R B/R)  1024 SWAP /  DUP 1- ROT +  SWAP / ;
 2 : #R   ( #B B/R)  1024 SWAP / * ;
 3
 4 : STOPPER   -1  1 RECORD 2+ !  UPDATE ;
 5 : INITIALIZE   ORG @  LIM @  B/R @ #B +  ORG @ DO
 6       I BLOCK 1024 ERASE UPDATE  LOOP  STOPPER ;
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:   48
 0    ( ORDERED UPDATES)
 1 CODE RSWAP   ( # S D -- # S)   I 2 MOV   I POP   W  POP   1 POP
 2    1 PUSH   W PUSH   1 SHR   BEGIN   W ) 0 MOV
 3    I ) 0 XCHG   STOS  I INC   I INC   LOOP    2 I  MOV  NEXT
 4
 5 : DIRECTION   AVAILABLE +! UPDATE  ENTIRE  AVAILABLE @ 2+
 6       SAFE R# @ ;
 7 : +ORDERED   1 DIRECTION DO  I RECORD RSWAP UPDATE LOOP
 8       2DROP  ORDERED RELEASE ;
 9 : -ORDERED   ENTIRE SWAP ERASE  -1 DIRECTION SWAP DO
10       I RECORD RSWAP UPDATE  -1 +LOOP  2DROP  ORDERED RELEASE ;
11
12
13
14
15

Screen:   49
 0    ( ORDERED SEARCH)
 1 : -BINARY   SWAP  AVAILABLE @ 2/ 1+  DUP READ  ORDERED GET
 2    BEGIN  DUP 1+ 2/  2OVER OVER ADDRESS  -TEXT 1 < IF
 3       MINUS  THEN R# +!  2/ DUP 0= END  DROP
 4    2DUP OVER ADDRESS  -TEXT 0 >  R# +!  OVER ADDRESS -TEXT ;
 5
 6 : BINARY   -BINARY  ORDERED RELEASE  ABORT" NOT IN FILE"
 7    LINK N@ ;
 8
 9
10
11
12
13
14
15

Screen:   50
 0    ( CHAINED RECORDS)
 1 : SNATCH   ( F #)  OVER N@  SWAP ROT N! ;  ( #')
 2 : FIRST   HEAD @ READ ;
 3 : -NEXT   LINK N@  0 OVER < IF  READ 1  THEN 1- ;
 4
 5 : -LOCATE   1+ FIRST BEGIN  1- DUP IF [ SWAP ]  -NEXT END THEN ;
 6    ( FINDS NTH OR LAST, TRUE IFF NOT FOUND)
 7
 8 : CHAIN   -LOCATE DROP  ( NTH RECORD OR END)
 9    SLOT  LINK OVER SNATCH  SWAP READ  LINK N! ;
10
11 : UNCHAIN   DUP 0= ABORT" WON'T"  -LOCATE ABORT" CAN'T"
12    SAVE  LINK N@ READ  LINK 0 SNATCH  RESTORE  LINK N! ;
13
14
15
```

```
Screen:   51
 0 ( FIELDS)
 1 : 1BYTE   DUP 1+  SWAP CONSTANT  DOES> @  WORKING + ;
 2 : NUMERIC   1BYTE 1+ ;      : DOUBLE   NUMERIC 2+ ;
 3 : BYTES   CREATE  2DUP +  ROT C, ( DISP.)  SWAP C, ( #)
 4       DOES>  DUP 1+ C@  SWAP C@  WORKING + ;
 5
 6 : ENTIRE   B/R @ WORKING ;
 7 : FILLER   + ;
 8 0   NUMERIC LINK   DROP
 9
10 : N?   N@ . ;
11 : D?   D@ D. ;
12 : 1?   1@ . ;
13 : B?   2DUP B@ S. ;
14
15

Screen:   52
 0    ( FIELD OPERATORS)
 1 : ADDRESS   R# @ RECORD  WORKING -  + ;
 2
 3 CODE >MOVE<   ( S D #)   I 2 MOV   1 POP   W POP   I POP   1 SHR
 4    BEGIN   LODS   0 0 HI  XCHG B   STOS   LOOP  2 I MOV   NEXT
 5
 6 : S@   PAD ROT >MOVE< ;      : S!   PAD SWAP  ROT >MOVE< ;
 7 : S.   DROP PAD SWAP -TRAILING TYPE ;
 8
 9 : N@   ADDRESS @ ;       : N!   ADDRESS ! UPDATE ;
10 : D@   ADDRESS 2@ ;      : D!   ADDRESS 2! UPDATE ;
11 : B@   ADDRESS S@ ;      : B!   ADDRESS S! UPDATE ;
12 : 1@   ADDRESS C@ ;      : 1!   ADDRESS C! UPDATE ;
13
14
15

Screen:   53
 0 ( SUBTOTALING)
 1 : REGISTER   H 2+ @ ;   ( WORKING)
 2 : D+!   >R  I 2@ D+  R> 2! ;
 3 : REG   REGISTER  DUP 2@ MOD  2+ 2+  DUP REGISTER 2+ !  + ;
 4
 5 : ZERO   2* 2*  REGISTER OVER 2+ ERASE  REGISTER ! ;
 6 : SUM   2* 2* REGISTER + D+! ;
 7 : FOOT   2DUP REG D+! ;
 8
 9 : SUB   REG >R  I 2@  2DUP REGISTER @ I + D+!  0. R> 2! ;
10 : GRAND   REGISTER DUP @ >R  2+ 2+  DUP I +  SWAP R> MOVE ;
11
12
13
14
15
```

```
Screen:   54
 0 ( REPORT FORMATTING)
 1 : 0COL   RPT @ 5 +  C# ! ;
 2 : +L   CR  ( 1 L# C+!)  0COL ;
 3 : COL   1 C# +!  C# @ ;
 4 : COLS   COL C@  ?DUP 0=  IF  +L  COL C@ THEN ;
 5
 6 : .TEXT   WORD  COUNT 1+ TYPE ;
 7 : TITLE   >IN 2@  RPT @ 2@  >IN 2!  92 ( \) .TEXT
 8    +L  93 ( ]) .TEXT  +L  >IN 2! ;
 9 : HEADING   RPT !  TITLE ;
10
11 : [R   HERE  >IN 2@ , ,  ['] TITLE ,  93 ( ]) WORD DROP
12    >IN @  OVER @  >IN !  92 ( \) WORD DROP
13    >IN @ 1+  BEGIN  32 WORD DROP  >IN @  DUP ROT - C,
14       OVER >IN @ < END  2DROP  -1 HERE 1- C+!  0 C, ;
15

Screen:   55
 0    ( PAGE FORMATTING)
 1 : SKIP   COLS SPACES ;
 2 : SKIPS   0 DO  SKIP LOOP ;
 3 : RIGHT   COLS OVER -  DUP 0< IF 1-  C# @ >R I C@ OVER - I C!
 4    I 1+ C@ + 0 MAX  R> 1+ C!  1 THEN  SPACES TYPE ;
 5
 6 : .PAGE   PAGE  0 L# C!  1 P# +!
 7 ." FORTH, Inc.                                          Page "
 8    P# ?  2 SPACES  TODAY @ .DATE  +L
 9    24 SPACES  RPT @ 2+ 2+ @ EXECUTE ;
10
11 : ?PAGE   L# C@ +  L/P C@ > IF  L/P C@ 60 < IF
12       KEY 0= ABORT" ?"  THEN .PAGE  THEN ;
13
14 : +CR   +L 1 ?PAGE ;
15 : REPORT   RPT !  0 P# !  .PAGE ;

Screen:   56
 0    ( REPORT GENERATOR OUTPUT)
 1 : .N   (.) RIGHT ;
 2 : .D   (D.) RIGHT ;
 3 : ?N   N@ .N ;
 4 : ?1   1@ .N ;
 5 : .S   DROP PAD  SWAP RIGHT ;
 6 : ?B   2DUP B@  PAD C@ IF .S  ELSE 2DROP SKIP  THEN ;
 7
 8 : .M/D/Y   ?DUP IF (DATE) RIGHT  ELSE SKIP  THEN ;
 9
10
11
12
13
14
15
```

```
Screen:   57
 0    ( CALENDAR - MAY 1976)
 1 365 4 * 1+ CONSTANT D/Y   VARIABLE JAN0   VARIABLE LEAP
 2 : MTH   CONSTANT DOES>  @ 58 OVER <  LEAP @ * +  JAN0 @ +  + ;
 3   0 MTH JAN      31 MTH FEB      59 MTH MAR      90 MTH APR
 4 120 MTH MAY     151 MTH JUN     181 MTH JUL     212 MTH AUG
 5 243 MTH SEP     273 MTH OCT     304 MTH NOV     334 MTH DEC
 6 : AD   1949 -  D/Y 4 */MOD  366 - JAN0 !  3 = LEAP ! ;
 7 : .YR   731 +  4 D/Y */MOD  1948 + SWAP  4 /MOD 1+
 8    DUP ROT 0= IF  DUP 60 > -  SWAP DUP 59 > -  THEN ROT ;
 9 : .MTH   ['] JAN  11 0 DO  2DUP 10 + @ >  IF 10 + 1
10    ELSE 11  THEN /LOOP  SWAP DROP  DUP 8 - >R  @ - R> ;
11 : ''   32 HOLD ;
12 : (DATE)   <# .YR  0 # # # # 2DROP ''  .MTH
13    -3 PTR +!  COUNT PTR @ SWAP MOVE  ''  0 #S #> ;
14 : .DATE   ?DUP IF  (DATE) TYPE SPACE
15       ELSE ." DY MTH NOW "  THEN ;      1979 AD

Screen:   58
 0 ( CLOCK, 800 HZ = 1.25 MS)
 1 8322 CONSTANT TICKS      69,120,000 2CONSTANT 1DAY
 2 ASSEMBLER BEGIN   TICKS 2+ INC   0= IF
 3       TICKS INC   THEN   IRET   10 INTERRUPT
 4 CODE CLOCK   0 0 SUB   194 OUT   NEXT
 5 CODE -CLOCK   4 #B 0 MOV   194 OUT   NEXT
 6 CODE COUNTER   TICKS 2+ PUSH   NEXT
 7 : TIMER   COUNTER SWAP -  10 8 */ . ;
 8 : MS   COUNTER  SWAP 8 10 */ +
 9    BEGIN PAUSE  DUP COUNTER < END  DROP ;
10
11 : PST   -CLOCK  100 M/MOD 60 * +  48000 U*  TICKS 2!  CLOCK ;
12 : @TIME   -CLOCK  1DAY  TICKS 2@  BEGIN  2OVER D-  DUP 0<
13    NOT IF  1 TODAY +!  AGAIN  D+  2DUP TICKS 2!  CLOCK ;
14 : .TIME   0 <# #  6 BASE ! #  DECIMAL 58 HOLD  # # #> ;
15 : TIME   @TIME 48000 M/MOD  .TIME TYPE SPACE  DROP ;

Screen:   59
 0 2020
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:   60
 0 ( DATA BASE MANAGEMENT SYSTEM)      EMPTY
 1 : WORKING   H 2+ @ ;
 2 CREATE ORDERED  0  ,
 3
 4 ( USER VARIABLES)   26 LOAD
 5 ( STRINGS) 34  LOAD
 6 ( FILES) 45  LOAD 46 LOAD
 7 ( FIELDS)  52 LOAD  51 LOAD
 8 ( ORDERED INDEXES) 48 LOAD  49 LOAD
 9 ( CHAINS) 50  LOAD   ( SUBTOTALS)  53  LOAD
10 ( REPORTS)   54 LOAD  55 LOAD  56 LOAD
11 ( APPLICATIONS)  61 LOAD
12
13 63 CONSTANT DOCUMENTER
14
15 CONTEXT GOLDEN 20 MOVE   HERE H 2+ !

Screen:   61
 0 ( BYTES  RECORDS    BLOCKS  ORG      NAME)
 1      26      144         2  180  FILE (GLOSSARY)
 2     340      144        48  182  FILE GLOSSARY
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   62
 0 2121
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:   63
 0 ( GLOSSARIZER)    EMPTY   40 ALLOT
 1 ( FILE)   64 LOAD
 2 : .STACK   0 <# #  45 HOLD  # #>   RIGHT  ;
 3 : (SHOW)   +L WORD ?B  VOC ?B  SOURCE  ?N  STACKS  N@ .STACK
 4    4 0 DO   PHRASE I  )  DUP  N@  IF +L  5  SPACES   B?
 5       ELSE 2DROP  THEN  LOOP +L ;
 6 [R FORTH GLOSSARY\ WORD         VOCABULARY  BLOCK   STACK]
 7    CONSTANT TITLE
 8 ( ENTRY)  65 LOAD
 9 : GLOSS   (GLOSSARY)  READ  LINK N@ DUP IF GLOSSARY READ
10       1  THEN ;
11 : SUMMARY   TITLE REPORT  (GLOSSARY)  RECORDS DO  6 ?PAGE
12      I GLOSS IF (SHOW) THEN  LOOP  SPACE ;
13 : /VOCABULARY  VOCABULARY TITLE  REPORT  (GLOSSARY)  RECORDS DO
14    6 ?PAGE  I GLOSS  IF  VOC  SWAP  OVER ADDRESS -TEXT  NOT IF
15      (SHOW) THEN THEN LOOP  SPACE ;

Screen:   64
 0     ( GLOSSARY FILE)
 1 2 ( LINK)  12 BYTES  WORD  12 BYTES VOC
 2    NUMERIC SOURCE  NUMERIC STACKS  70 BYTES PHRASE
 3    210 FILLER ( 4 LINES)  32 FILLER  ( 340 B/R, 3/BLOCK ) DROP
 4 2 24 BYTES WORD-VOC   DROP
 5 : )   >R  OVER  R> * + ;
 6
 7 : -FIND  32 TEXT WORD  S!  (GLOSSARY)  WORD+VOC  -BINARY ;
 8 : (ENTER)  DROP -FIND IF
 9      SAVE  GLOSSARY SLOT  DUP LINK !  RESTORE  +ORDERED
10          WORD+VOC  B@  WORD+VOC  S!
11      GLOSSARY READ  WORD+VOC  B!  STACKS N!  SOURCE @ SOURCE N!
12      ELSE  ORDERED RELEASE  LINK N@ GLOSSARY READ  DROP THEN
13          TODAY  @ 1 MAX LINK N! ;
14 : VOCABULARY   32 TEXT VOC S! ;
15

Screen:   65
 0    ( GLOSSARY MAINTENANCE)
 1 VARIABLE L
 2
 3 : (FIND)  -FIND ABORT"  NOT IN FILE"  LINK  N@ ;
 4 : INITIAL  0  L !  CR  70 SPACES ;
 5 : ENTER  DROP  (ENTER) INITIAL ;
 6 : DELETE   (FIND)  -ORDERED  GLOSSARY  READ  0. LINK D!  ;
 7
 8 : AT SOURCE N!  ;
 9 : STACK   DROP  STACKS  N! ;
10 : T   3 MIN  L !  CR SPACE SPACE PHRASE L @ ) B? ;
11 : P   1 TEXT  PHRASE L @ ) B!  L @ 1+ T ;
12 : F   TITLE HEADING  (SHOW)  0 L ! ;
13 : FIND  (FIND) ORDERED RELEASE  GLOSSARY  READ  F ;
14 : X   PHRASE L @ )   0 SWAP N!  DROP ;
15
```

```
Screen:   66
 0 2323
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   67
 0 2323
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   68
 0 2323
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:   69
 0 2424
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   70
 0 2424
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   71
 0 2424
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:   72
 0 2525
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   73
 0 2525
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   74
 0 2525
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 75
 0 2626
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 76
 0 2626
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 77
 0 2626
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen:   78
 0 2727
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   79
 0 2727
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   80
 0 2727
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:   81
 0 2828
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   82
 0 2828
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   83
 0 2828
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:   84
 0 2929
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   85
 0 2929
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   86
 0 2929
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:   87
 0 3030
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   88
 0 3030
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:   89
 0 ( TARGET COMPATIBILITY)   EMPTY   DECIMAL
 1 : STRING   WORD C@ 1+ ALLOT ;
 2
 3    VARIABLE 'R      CREATE WDS 8 ,
 4 : THERE   'R @ ;      : ALLOT   'R +! ;      : ORG   H ! ;
 5 : HOST   FORTH ;   IMMEDIATE      : FORTH   FORTH ;   IMMEDIATE
 6 : TARGET   IMMEDIATE ;
 7
 8 : CVARIABLE   THERE CONSTANT  1 ALLOT ;
 9 : VARIABLE   CVARAIBLE  1 ALLOT ;
10 : LABEL   HERE 10 + CONSTANT  ASSEMBLER ;
11
12 : THRU   1+ SWAP DO  I LOAD  LOOP ;
13
14
15
```

```
Screen:   90
 0 ( TARGET COMPILER)   EMPTY   DECIMAL
 1 CREATE HEAD  16 ALLOT   CREATE WDS  4 ALLOT   CREATE ?\  1 ,
 2 VARIABLE W0   VARIABLE 'H   VARIABLE 'R   VARIABLE VOC
 3    ( META)  91 LOAD
 4 : LOG   ( BASE @ HEX  OVER U.  BASE !
 5       >IN @  32 WORD  COUNT 1+ TYPE  >IN !) ;
 6
 7 : ORG   'H ! ;      : GAP   'H +! ;
 8 : HERE   'H @ ;     : THERE   'R @ ;      HEX
 9 : WINDOW   DUP ORG  W0 !  0 'R !  000B VOC !  HEAD 10 ERASE ;
10 : |   2 WDS ! ;      DECIMAL
11    'CR @ >R   ' SPACE 'CR !      ( ACCESS)  93 LOAD
12    ( ASSEMBLER)  95 LOAD      R> 'CR FORTH !
13 ( : LOAD   CR DUP . LOAD ;)      : CR   ( CR SPACE) ;
14 : THRU   1+ SWAP DO  FORTH I HOST LOAD  LOOP ;
15    120 EQU 8086      : ALLOT   GAP ;

Screen:   91
 0    ( HOST COMPUTER)   HEX
 1 : TEXT   LAST @ @ ;      : PREVIOUS   TEXT 8 + @ ;
 2 : -'   CONTEXT @  VOC @ CONTEXT !  -'  ROT CONTEXT ! ;
 3
 4       ' DOES> 2 -
 5 : DOES>   [ , ]  FORTH ' 2 - ,  5179 CONTEXT ! ;  IMMEDIATE
 6
 7 0571 VOCABULARY FORTH   IMMEDIATE
 8 0517 VOCABULARY HOST   IMMEDIATE
 9 5179 VOCABULARY ASSEMBLER      DECIMAL
10
11 : HASH   TEXT 1+ C@  VOC @ +  14 AND  CONTEXT 2+ + ;
12 : TARGET   LAST @  DUP @  DUP 2+ 2+ @  ROT ROT DUP
13       HASH  DUP LAST !  DUP @  ROT 2+ 2+ !  !  ! ;
14 HOST DEFINITIONS
15

Screen:   92
 0    ( COMPILE TO DISK)
 1 : DICTIONARY   250 240 DO  I BLOCK  -1 OVER !
 2    DUP 2+ 1022 MOVE  UPDATE  LOOP  FLUSH  DUP WDS 2! ;
 3 : 'P   W0 @ -  0 MAX 10239 MIN  1024 /MOD  240 +  BLOCK + ;
 4
 5 : C@   'P C@ ;      : C!   'P C! UPDATE ;
 6 : @   DUP PTR - IF  DUP C@  SWAP 1+ C@  256 * +  ELSE @  THEN ;
 7 : !   >R 256 /MOD  I 1+ C!  R> C! ;
 8 : ,   HERE !  2 GAP ;      : C,   HERE C!  1 GAP ;
 9 : !   DUP PTR - IF  !  ELSE  FORTH !  THEN ;
10 : 2!   SWAP  OVER !  2+ ! ;
11 : +!   SWAP  OVER @ +  SWAP ! ;
12
13 : DUMP   OVER + SWAP DO  CR I 5 U.R SPACE  I 16 + I' MIN I DO
14    I 7 AND 0= 2* SPACES  I C@ 3 U.R  LOOP  16 /LOOP  SPACE ;
15
```

```
Screen:   93
 0    ( COMPILE TO RAM)
 1 CREATE RAM  8192 ALLOT      : +RAM   W0 @ -  8191 AND  RAM + ;
 2 : DICTIONARY   -1 RAM !  RAM DUP 2+ 8190 MOVE  DUP WDS 2! ;
 3
 4 : C@   +RAM C@ ;      : C!   +RAM C! ;
 5 : C,   HERE C!  1 GAP ;      : ,   HERE +RAM !  2 GAP ;
 6 : @   DUP PTR - IF  +RAM  THEN @ ;
 7 : !   DUP PTR - IF  +RAM  THEN ! ;
 8 : 2!   +RAM 2! ;      : +!   +RAM +! ;
 9
10 : DUMP   OVER + SWAP DO  CR I 5 U.R SPACE  I 16 + I' MIN I DO
11    I 7 AND 0= 2* SPACES  I C@ 3 U.R  LOOP  16 /LOOP  SPACE ;
12
13 : FLUSH   RAM  248 240 DO  DUP I BLOCK 1024 MOVE
14    UPDATE  1024 +  LOOP  FLUSH DROP ;
15

Screen:   94
 0 3232 ( COMPILE TO TARGET SYSTEM, 8080 CODE)   HEX
 1 : DICTIONARY   DUP WDS 2! ;      CREATE BUF  4 ALLOT
 2 : DONE   F7 OPERATOR @ C!  BEGIN  STOP  0 END [
 3 : SEND   BUF 2!  PRINTER ACTIVATE  BUF 3 TYPE  DONE [
 4 : GET   BUF !  PRINTER ACTIVATE  BUF 0 CTR 2!  (TYPE)
 5    BUF 1+ C@ EMIT  DONE [
 6
 7 : C@   GET STOP  BUF 1+ C@ ;      : C!   SEND STOP ;
 8 : @   DUP PTR - IF  DUP C@  SWAP 1+ C@  256 * +  ELSE @  THEN ;
 9 : !   >R 256 /MOD  I 1+ C!  R> C! ;
10 : ,   HERE !  2 GAP ;      : C,   HERE C!  1 GAP ;
11 : !   DUP PTR - IF  !  ELSE  FORTH !  THEN ;
12 : 2!   SWAP  OVER !  2+ ! ;
13 : +!   SWAP  OVER @ +  SWAP ! ;
14 : DUMP   OVER + SWAP DO  CR I 5 U.R SPACE  I 16 + I' MIN I DO
15    I 7 AND 0= 2* SPACES  I C@ 3 U.R  LOOP  16 /LOOP  SPACE ;

Screen:   95
 0    ( TARGET ASSEMBLER)
 1 : EMPLACE   LOG  CONSTANT  DOES> @ 2 - , ;
 2 : CREATE   HERE  WDS FORTH @ +  EMPLACE  WDS @ 2 - IF
 3       TEXT 2@  DUP HOST ,  SWAP ,  256 /  VOC FORTH @ 10 - +
 4       14 AND  HEAD +  DUP @ HOST ,  HERE 6 -  SWAP FORTH !
 5    THEN  WDS 2+ @ WDS !  HOST HERE 2+ , ;
 6 : CODE   CREATE TARGET  ASSEMBLER ;
 7 : EQU   CONSTANT ;      : LABEL   HERE LOG EQU  ASSEMBLER ;
 8 : STRING   WORD  DUP FORTH C@ 1+
 9    OVER + SWAP DO  I C@  HOST C,  LOOP ;
10 : MOVE   OVER + SWAP DO  DUP FORTH C@
11    I HOST C!  1+  LOOP  DROP ;
12 FORTH  HERE 3 -  15 C,  0 C,  14 C,  12 C,  13 C,
13 ASSEMBLER DEFINITIONS      : )   0  SWAP IN-LINE +  FORTH C@ ;
14 ASSEMBLER   12 LOAD  13 LOAD  14 LOAD  15 LOAD  16 LOAD  17 LOAD
15    : BEGIN   HERE ;   3) CONSTANT U)      HOST DEFINITIONS
```

```
Screen:   96
 0    ( TARGET COMPILER)
 1 : \   -' ABORT" ?"  2 - FORTH , ;   IMMEDIATE
 2 : '   -' ABORT" ?"  FORTH @ ;
 3
 4 : IN-LINE   \ LITERAL , ;   TARGET
 5 : [']   '  \ LITERAL  , ;   TARGET
 6 : ]   BEGIN  -' IF  NUMBER  ?CELL IF
 7       \ LITERAL ,  ELSE  \ LIT C,  THEN ELSE  EXECUTE
 8    ?STACK ABORT" STACK EMPTY"  THEN (ELSE) [ (BACK)
 9 : [   R> DROP ;   TARGET
10
11 : (;CODE)   R> FORTH @  PREVIOUS 2 - HOST ! ;
12 : ;CODE   FORTH \ HOST (;CODE)  HERE FORTH ,
13       R> DROP  ASSEMBLER ;   IMMEDIATE
14 : DOES>   FORTH \ HOST (;CODE)  HERE FORTH ,
15       R> DROP  HOST  232 C, (DOES>) HERE - ,  ] ;   IMMEDIATE

Screen:   97
 0    ( COMPILER DIRECTIVES)
 1 : (BACK)   HERE 1+ - C, ;
 2 : BEGIN   HERE ;   TARGET
 3 : END   \ (IF)  (BACK) ;   TARGET
 4
 5 : (THEN)   HERE 1- OVER -  SWAP C! ;
 6 : IF   \ (IF)  HERE 0 C, ;   TARGET
 7 : THEN   (THEN) ;   TARGET
 8 : ELSE   \ (ELSE)  HERE 0 C,  SWAP (THEN) ;   TARGET
 9 : AGAIN   \ (ELSE)  SWAP (BACK)  (THEN) ;   TARGET
10
11 : DO   \ (DO)  HERE ;   TARGET
12 : LOOP   \ (LOOP)  (BACK) ;   TARGET
13 : /LOOP   \ (/LOOP)  (BACK) ;   TARGET
14 : (   41 WORD DROP ;   TARGET
15 : ;   \ EXIT  R> DROP ;   TARGET

Screen:   98
 0    ( DEFINING WORDS)
 1 : \CONSTANT   FORTH ?\ @ IF  >IN @  OVER CONSTANT  >IN !
 2       THEN DROP ;   HOST
 3 HERE  (:) ORG
 4 FORTH : :   CREATE  FORTH CURRENT @ CONTEXT !
 5             HOST ] TARGET  ;CODE
 6       (CONSTANT) ORG
 7 FORTH : CONSTANT   DUP \CONSTANT  CODE ,  ;CODE
 8 FORTH : CVARIABLE   THERE CONSTANT  1 ALLOT ;   ( PROM VERSION)
 9 FORTH : VARIABLE   CVARIABLE  1 ALLOT ;   ( PROM VERSION)
10       (CREATE) ORG
11 FORTH : CREATE   HERE WDS FORTH @ +  HOST \CONSTANT  CODE  ;CODE
12       (USER) ORG
13 FORTH : USER   DUP \CONSTANT  CODE C,  ;CODE            ORG
14 FORTH : CVARIABLE   CREATE  1 ALLOT ;   ( RAM VERSION)
15 FORTH : VARIABLE   CVARIABLE  1 ALLOT ;   ( RAM VERSION)
```

```
Screen:   99
 0 3434
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  100
 0 3434
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  101
 0 3434
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  102
 0 3535
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  103
 0 3535
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  104
 0 3535
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  105
 0 3636
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  106
 0 3636
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  107
 0 3636
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  108
 0 3737
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  109
 0 3737
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  110
 0 3737
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  111
 0 3838
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  112
 0 3838
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  113
 0 3838
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 114
 0 3939
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 115
 0 3939
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 116
 0 3939
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen:  117
 0 4040
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  118
 0 4040
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  119
 0 4040
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  120
 0 ( 8086 METAFORTH)   HEX   HOST DEFINITIONS
 1 FORGET ALLOT   : ALLOT   'R FORTH +! ;   HOST
 2 2000 EQU PROM   PROM 10 / MINUS EQU ZERO
 3    8 DICTIONARY  0 WINDOW  1 ORG  PROM 40 + ALLOT
 4 CODE X   HERE 8 - HEAD FORTH !   HEAD DUP 2+ 0E MOVE
 5    2080 PREVIOUS 8 - HOST !   DECIMAL
 6
 7 ( NUCLEUS)  123 131 THRU   ( COMPILER)  96 98 THRU
 8    FORGET CVARIABLE
 9 ( LEVEL 1)  132 134 THRU   ( MULTI-TASK)  135 136 THRU
10 ( TERMINAL)  138 LOAD   ( OUTPUT)  144 146 THRU
11 ( INTERPRETER)  147 150 THRU
12 ( DISK)  153 154 THRU   ( COMPILER)  159 164 THRU
13    HEX  CR  HERE U.  THERE U.  DECIMAL
14 ( EDITOR)  171 LOAD   ( ASSEMBLER)  172 LOAD
15       121 EQU 86/12

Screen:  121
 0    ( 86/12)
 1 HEX   4 NB !   A000 400 NB @ * -  DUP FIRST !
 2    CREATE OPERATOR  60 - ,   DECIMAL
 3
 4 ( TERMINAL)  137 LOAD  139 LOAD  141 142 THRU
 5 ( DISK)  156 158 THRU
 6 ( INITIALIZE)  165 167 THRU   HEX
 7
 8 ( LINKS)  HOST  HEAD RAM 2+ 10 MOVE
 9 ( VECTOR SEED)  ZERO <IRET>  PROM 4 - 2!
10    CR  HERE U.  THERE U.  DECIMAL   FLUSH
11
12
13
14
15

Screen:  122
 0 4141
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  123
 0 ( NUCLEUS)
 1 CODE LITERAL   LODS   0 PUSH   NEXT   ( 63)
 2 | CODE LIT   LODS B   CWD B   0 PUSH   NEXT   ( 63)
 3
 4 LABEL ':'   26 GAP
 5 LABEL (:)   W INC   W INC   R DEC   R DEC
 6    I R ) MOV   W I MOV   NEXT   ( 67)
 7 CODE EXIT   R ) I MOV   R INC   R INC   NEXT   ( 60)
 8
 9
10
11
12
13
14
15

Screen:  124
 0    ( DEFINING EXECUTION)
 1 LABEL 'CONSTANT'   14 GAP
 2 LABEL (CONSTANT)   2 W) PUSH   NEXT   ( 66)
 3
 4 LABEL 'CREATE'   44 GAP
 5 LABEL (CREATE)   W INC   W INC   W PUSH   NEXT   ( 53)
 6                                                          CR
 7 HERE 2 - EQU (DOES>)   R DEC   R DEC   I R ) MOV   I POP
 8    W INC   W INC   W PUSH   NEXT   ( 83)
 9
10 LABEL 'USER'   14 GAP
11 LABEL (USER)   2 W) 0 MOV B   CWD B   U 0 ADD
12    0 PUSH   NEXT   ( 71)
13
14
15

Screen:  125
 0    ( STRUCTURES)
 1 CODE (ELSE)
 2    BEGIN   LODS B   CWD B   0 I ADD   NEXT   ( 56)
 3 | CODE (IF)   0 POP   0 0 OR   0= NOT END   ( 83)
 4    I INC   NEXT   ( 56)
 5
 6 | CODE (DO)   4 # R SUB   R ) POP   2 R) POP   NEXT   ( 95)
 7
 8 | CODE (LOOP)   R ) 0 MOV   0 INC
 9 HERE   2 R) 0 CMP   CS IF   ( 98)
10    BEGIN   0 R ) MOV   LODS B   CWD B   0 I ADD   NEXT   ( 115)
11 | CODE (+LOOP)   0 POP   0 2 MOV   R ) 0 ADD
12       0 W MOV   2 R) W SUB   W 2 XOR   0< NOT END   ( 141)
13    THEN   4 # R ADD   I INC   NEXT   ( 100)
14 | CODE (/LOOP)   0 POP   R ) 0 ADD   JMP   ( LOOP+22)
15
```

```
Screen:  126
 0    ( LOGICAL OPERATORS)
 1 CODE AND   2 POP   0 POP   2 0 AND   0 PUSH   NEXT   ( 68)
 2 CODE OR   2 POP   0 POP   2 0 OR   0 PUSH   NEXT   ( 68)
 3 CODE XOR   2 POP   0 POP   2 0 XOR   0 PUSH   NEXT   ( 68)
 4
 5 CODE 0=   0 POP   0 0 OR   HERE   0 # 0 MOV
 6    0= IF   0 INC   THEN   0 PUSH   NEXT   ( 70T,80F)
 7 CODE =   2 POP   0 POP   2 0 CMP   JMP   ( 93T,103F)
 8
 9 CODE 0<   0 POP   0 0 OR   HERE   0 # 0 MOV
10    0< IF   0 INC   THEN   0 PUSH   NEXT   ( 70T,80F)
11 CODE <   2 POP   0 POP   2 0 SUB   DUP JMP   ( 93T,103F)
12 CODE >   0 POP   2 POP   2 0 SUB   JMP   ( 93T,103F)
13
14
15

Screen:  127
 0    ( ACCESS OPERATORS)
 1 CODE @   W POP   W ) PUSH   NEXT   ( 70)
 2 CODE !   W POP   0 POP   STOS   NEXT   ( 67)
 3 CODE +!   W POP   0 POP   0 W ) ADD   NEXT   ( 80)
 4
 5 CODE C@   W POP   0 0 SUB   W ) 0 MOV B   0 PUSH   NEXT   ( 73)
 6 CODE C!   W POP   0 POP   STOS B   NEXT   ( 65)
 7
 8 CODE >R   R DEC   R DEC   R ) POP   NEXT   ( 69)
 9 CODE R>   R ) PUSH   R INC   R INC   NEXT   ( 68)
10
11
12
13
14
15

Screen:  128
 0    ( STACK OPERATORS)
 1 CODE DUP   S W MOV   W ) PUSH   NEXT   ( 62)
 2 CODE DROP   S INC   S INC   NEXT   ( 43)
 3 CODE SWAP   2 POP   0 POP   2 PUSH   0 PUSH   NEXT   ( 75)
 4
 5 CODE OVER   S W MOV   2 W) PUSH   NEXT   ( 66)
 6 CODE ROT   W POP   2 POP   0 POP
 7     2 PUSH   W PUSH   0 PUSH   NEXT   ( 93)
 8
 9 CODE I   R ) PUSH   NEXT   ( 64)
10
11
12
13
14
15
```

```
Screen:  129
 0    ( ARITHMETIC OPERATORS)
 1 CODE +   2 POP   0 POP   2 0 ADD   0 PUSH   NEXT   ( 68)
 2 CODE -   2 POP   0 POP   2 0 SUB   0 PUSH   NEXT   ( 68)
 3
 4 CODE 1+   0 POP   0 INC   0 PUSH   NEXT   ( 59)
 5 CODE 1-   0 POP   0 DEC   0 PUSH   NEXT   ( 59)
 6 CODE 2+   0 POP   0 INC   0 INC   0 PUSH   NEXT   ( 61)
 7
 8 CODE 2*   0 POP   0 SHL   0 PUSH   NEXT   ( 59)
 9 CODE 2/   0 POP   0 SAR   0 PUSH   NEXT   ( 59)
10
11
12
13
14
15

Screen:  130
 0    ( MULTIPLY, DIVIDE)
 1 CODE *   W POP   0 POP   W MUL   0 PUSH   NEXT   ( 189)
 2
 3 CODE /MOD   W POP   0 POP   2 2 SUB
 4    W DIV   2 PUSH   0 PUSH   NEXT   ( 233)
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  131
 0    ( ARRAY OPERATORS)
 1 CODE MOVE   1 POP   W POP   0 POP   1NZ IF   I 0 XCHG   W ROR
 2    W ROL   CS IF   MOVS B   1 DEC   THEN   1 SHR   REP MOVS
 3    CS IF   MOVS B   THEN   I 0 XCHG   THEN   NEXT   ( 120+12.5*)
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  132
 0 ( LEVEL 1 WORDS)
 1 CODE MINUS   0 POP   BEGIN   0 NEG   0 PUSH   NEXT   ( 60)
 2 CODE ABS   0 POP   0 0 OR   0< NOT END   ( 79)
 3    0 PUSH   NEXT   ( 64)
 4 CODE NOT   ' 0= HERE 2 - !   ( 70T,80F)
 5
 6 CODE MIN   0 POP   2 POP   2 0 CMP
 7    HERE   0< IF   2 0 XCHG   THEN   2 PUSH   NEXT   ( 80)
 8 CODE MAX   0 POP   2 POP   0 2 CMP   JMP   ( 95)
 9
10 0 CONSTANT 0   1 CONSTANT 1
11
12
13
14
15

Screen:  133
 0    ( MISCELLANEOUS)
 1 CODE ERASE   0 0 SUB   1 POP   W POP   1NZ IF   W ROR
 2    W ROL   CS IF   STOS B   1 DEC   THEN   1 SHR   REP STOS
 3    CS IF   STOS B   THEN   THEN   NEXT   ( 100+8*)
 4
 5 CODE 'S   S 0 MOV   0 PUSH   NEXT   ( 51)
 6 CODE ?DUP   0 POP   0 0 OR   0= NOT IF
 7    0 PUSH   THEN   0 PUSH   NEXT   ( 75)
 8
 9 CODE I'   2 R) PUSH   NEXT   ( 64)
10 CODE J   4 R) PUSH   NEXT   ( 64)
11
12 CODE LEAVE   R ) 0 MOV   0 2 R) MOV   NEXT   ( 74)
13
14
15

Screen:  134
 0    ( BASIC 32-BIT)
 1 CODE 2@   W POP   2 W) PUSH   W ) PUSH   NEXT   ( 97)
 2 CODE 2!   W POP   0 POP   STOS   0 POP   STOS   NEXT   ( 87)
 3
 4 CODE 2DUP   S W MOV   2 W) PUSH   W ) PUSH   NEXT   ( 87)
 5 CODE 2DROP   4 # S ADD   NEXT   ( 43)
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  135
 0 ( TERMINAL)
 1 0 USER STATUS   8 USER S0   10 USER CTR   12 USER PTR
 2 14 USER 'CR   16 USER 'PAGE   18 USER 'TYPE                    CR
 3 20 USER DEVICE   24 USER 'EXPECT   26 USER H
 4 30 USER OFFSET   32 USER CNT   34 USER BASE   36 USER >IN    CR
 5 38 USER BLK   40 ( - 57) USER CONTEXT   58 USER CURRENT
 6 60 USER LAST   62 USER SCR   64 USER R#                      CR
 7
 8 VARIABLE GOLDEN  18 ALLOT
 9 VARIABLE 'BLOCK   VARIABLE 'BUFFER  2 ALLOT
10 VARIABLE 'NUMBER   VARIABLE 'CREATE                          CR
11 VARIABLE DISK  2 ALLOT   VARIABLE PREV  18 ALLOT
12
13 PROM 8 - CONSTANT NB   PROM 6 - CONSTANT FIRST
14 PROM CONSTANT PROM
15

Screen:  136
 0    ( MULTI-PROGRAMMER)   HEX
 1 D7FF EQU WAKE
 2
 3 ASSEMBLER HERE   U POP   U DEC   U DEC   EA2E # STATUS U) MOV
 4    STATUS 6 + U) S MOV   I POP   R POP   NEXT
 5
 6 CODE PAUSE   WAKE # STATUS U) MOV   ( 194)
 7 LABEL WAIT   R PUSH   I PUSH   S STATUS 6 + U) MOV
 8    # W MOV   STATUS 2+ U) LIS
 9
10 CODE STOP   WAIT HERE 2 - !   ( 179)      DECIMAL
11
12
13
14
15

Screen:  137
 0    ( GET, RELEASE)
 1 | CODE GET   W POP   W ) U CMP   0= NOT IF   W ) 1 MOV
 2    1NZ IF   4 # I SUB   W PUSH   NEXT   ( 101+PAUSE)
 3       THEN   U W ) MOV   THEN   ' EXIT JMP
 4 : GET   PAUSE GET [
 5 CODE RELEASE   W POP   W ) U CMP   0= IF
 6    0 # W ) MOV   THEN   NEXT
 7 CODE GRAB   W POP   U W ) MOV   NEXT
 8
 9 CODE ACTIVATE   U 0 XCHG   W POP   W ) U MOV
10    WAKE # STATUS U) MOV   0 # CTR U) MOV
11    S0 U) W MOV   4 # W SUB   I W ) MOV   U 2 W) MOV
12    W STATUS 6 + U) MOV   U 0 XCHG   ' EXIT JMP
13
14
15
```

```
Screen:  138
 0    ( VECTORED TERMINAL)
 1 CODE EXECUTE   W POP   W DEC   W DEC   W ) LIP   ( 34)
 2
 3 CODE TYPE   0 POP   PTR U) POP   0 0 OR   0> IF
 4       0 CTR U) MOV   'TYPE U) W MOV   ' EXECUTE 1+ JMP
 5    THEN   BEGIN   NEXT
 6
 7 CODE EXPECT   0 POP   PTR U) POP   0 NEG   0< END
 8       0 CTR U) MOV   0 # CNT U) MOV   'EXPECT U) W MOV
 9    ' EXECUTE 1+ JMP
10
11 : CR   'CR @ EXECUTE ;
12
13 : PAGE   'PAGE @ EXECUTE ;
14
15

Screen:  139
 0    ( TERMINAL I/O)   HEX
 1 CODE (TYPE)   DEVICE U) 2 MOV   2 INC   2 INC   26 #B 0 MOV
 2    (2) OUT   CTR U) INC   0 INC   (2) OUT   WAIT JMP
 3
 4 CODE (EXPECT)   WAIT HERE 2 - !
 5
 6 MSG (CR)  2 C,  0D C,  0A C,
 7
 8 MSG (PAGE)  1 C,  0C C,      DECIMAL
 9
10
11
12
13
14
15

Screen:  140
 0 4747
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  141
 0    ( TYPE INTERRUPT)
 1 CREATE <TYPE>   ASSEMBLER
 2    0 PUSH   U PUSH   DS PUSHS   ZERO # 0 MOV   0 DS LSG
 3    U IS SSG   U SHL   U SHL   U SHL   U SHL
 4    CTR U) 0 MOV   0 DEC   0< NOT IF   0 CTR U) MOV
 5    0= IF   WAKE # STATUS U) MOV   ELSE
 6       2 PUSH   PTR U) I XCHG   DEVICE U) 2 MOV
 7       LODS B   PTR U) I XCHG   (2) OUT   2 POP
 8    THEN   THEN   DS POPS   U POP   0 POP   IRET
 9
10 HERE 1- EQU <IRET>
11
12
13
14
15

Screen:  142
 0    ( EXPECT INTERRUPT)
 1 CREATE <EXPECT>   ASSEMBLER
 2    0 PUSH   2 PUSH   U PUSH   DS PUSHS   ZERO # 0 MOV
 3    0 DS LSG   U IS SSG   U SHL   U SHL   U SHL   U SHL
 4    DEVICE 2+ U) 2 MOV   (2) IN   128 #B CTR 1+ U) TEST
 5    0< IF   PTR U) I XCHG   127 #B 0 AND
 6       127 #B 0 CMP   0= IF   7 #B 0 MOV   CNT U) DEC
 7          0< IF   CNT U) INC   ELSE   0 INC   I DEC   CTR U) DEC
 8       SWAP ELSE   13 #B 0 CMP   0= NOT IF   0 I ) MOV B
 9          I INC   CNT U) INC   CTR U) INC   0= IF
10             SWAP ELSE   32 #B 0 MOV   THEN   1 # CTR U) MOV
11       THEN   THEN   THEN   PTR U) I XCHG   (2) OUT
12    THEN   DS POPS   U POP   2 POP   0 POP   IRET
13
14
15

Screen:  143
 0 4848
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  144
 0    ( OUTPUT)
 1 : HERE   H @ ;      : PAD   H @ 33 + ;
 2
 3 CODE COUNT   W POP   W ) 0 MOV B   CWD B   W INC
 4    W PUSH   0 PUSH   NEXT
 5 CODE BLANK   8224 # 0 MOV   ' ERASE 2+ JMP
 6                                                              CR
 7 HERE  8224 , 8224 , 8224 , 8224 ,
 8 | CODE ?ANY   W W SUB   0 POP   0 0 OR   0< NOT IF   8 # W MOV
 9       W 0 SUB   0 PUSH   SWAP # 2 MOV   2 PUSH   0< IF   0 W ADD
10    THEN   W PUSH   W INC   THEN   W PUSH   NEXT
11 : SPACES   BEGIN  ?ANY IF  TYPE  AGAIN ;
12 : SPACE   1 SPACES ;
13
14 CODE HOLD   PTR U) DEC   PTR U) W MOV
15    0 POP   STOS B   NEXT

Screen:  145
 0    ( NUMBERS)
 1 CODE NUMERAL   0 POP   10 #B 0 CMP   CS NOT IF
 2    7 #B 0 ADD   THEN   48 #B 0 ADD   0 PUSH   NEXT
 3
 4 : <#   PAD PTR ! ;
 5 : #>   2DROP  PTR @  PAD OVER - ;
 6 : SIGN   ROT 0< IF  45 HOLD  THEN ;
 7 : #   BASE @ /MOD  SWAP NUMERAL HOLD ;
 8 : #S   BEGIN  #  DUP 0= END ;                                CR
 9 : (.)   DUP ABS DUP  <# #S SIGN #> ;
10 : .   (.) TYPE SPACE ;      : ?   @ . ;
11
12 : U.   DUP  <# #S #>  TYPE SPACE ;
13 : U.R   SWAP DUP  <# #S #>  ROT OVER - SPACES  TYPE ;
14 : DUMP   OVER + SWAP DO  CR I 5 U.R SPACE  I 16 + I' MIN I DO
15    I 7 AND 0= 2* SPACES  I C@ 3 U.R  LOOP  16 /LOOP  SPACE ;

Screen:  146
 0    ( MESSAGES)
 1 | CODE R@   R ) W MOV   W ) 0 MOV B   CWD B   0 INC   0 R ) ADD
 2    0 POP   0 0 OR   0= IF   ' EXIT JMP   THEN   W PUSH   NEXT
 3 | CODE RESET   S0 U) S MOV   0 0 SUB   0 PUSH
 4 LABEL 'QUIT   0 # I MOV   NEXT
 5
 6 | : (ABORT")   R@  BLK @ IF  BLK ?  THEN
 7    HERE COUNT 1+ TYPE  COUNT TYPE  CR  RESET [
 8 | : (.")   1 R@  COUNT TYPE ;
 9
10 FORTH : ABORT"   \ (ABORT")  34 STRING ;   TARGET
11 FORTH : ."   \ (.")  34 STRING ;   TARGET
12
13
14
15
```

```
Screen:  147
 0 ( INTERPRETER)
 1 | CODE SOURCE   1024 # 1 MOV   BLK U) W MOV   W W OR   0= IF
 2       CNT U) 1 MOV   S0 U) W MOV   I INC   I INC   THEN
 3    1 PUSH   W PUSH   NEXT
 4
 5 | CODE (WORD)   W POP   1 POP   0 POP
 6    >IN U) W ADD   >IN U) 1 SUB   1 >IN U) ADD
 7    I I SUB   REP SCAS B   W I MOV   0= NOT IF
 8       I DEC   REP B SCAS B   0= IF   W DEC   THEN   THEN
 9    I W SUB   1 >IN U) SUB   W 1 MOV   H U) W MOV   W PUSH
10    1 0 MOV   STOS B   REP MOVS B   8224 # 0 MOV
11    STOS   STOS B   ' EXIT JMP
12
13 : WORD   SOURCE SOURCE  (WORD) [
14
15

Screen:  148
 0    ( DICTIONARY SEARCH)
 1 | CODE HASH   2 POP   2 0 MOV   15 # 0 AND   0= IF
 2       I PUSH   ' EXIT JMP   THEN   4 # 1 MOV   2 SHR V
 3    W POP   W PUSH   1 W) 0 ADD B   14 #B 0 AND
 4    CONTEXT 2+ #B 0 ADD   U 0 ADD   0 PUSH   NEXT
 5
 6 | CODE -FIND   2 -2 R) MOV   W POP   W ) W MOV   I 0 XCHG
 7    I POP   I PUSH   I ) 1 MOV   1 SHL B   2 I) 2 MOV   I 0 XCHG
 8    BEGIN   2 W) 2 CMP   0= NOT IF   BEGIN
 9          4 W) W MOV   W W OR   ROT 0= END   ( 62*)
10       -2 R) PUSH   4 # I SUB   NEXT
11       SWAP THEN   W ) 0 MOV   0 SHL B   1 0 SUB   0= END
12    I POP   8 # W ADD   W PUSH   0 PUSH   ' EXIT JMP
13
14 : -'   32 WORD  CONTEXT @  HASH -FIND [
15

Screen:  149
 0    ( NUMBER INPUT)
 1 CODE DIGIT   2 2 SUB   W POP   W INC   W PUSH   W ) 0 MOV B
 2    58 #B 0 CMP   CS NOT IF   65 #B 0 CMP   CS NOT IF
 3       7 #B 0 SUB   SWAP THEN   48 #B 0 SUB   CS NOT IF
 4       BASE U) 0 CMP B   CS IF   CWD B   0 PUSH   2 INC
 5    THEN   THEN   THEN   2 PUSH   NEXT
 6
 7 | : (NUMBER)   DUP 1+ C@  45 =  DUP >R  +
 8    0 BEGIN SWAP  DIGIT IF  ROT BASE @ * + AGAIN
 9    C@ 32 - ABORT" ?"  R> IF MINUS  THEN ;
10
11 CODE NUMBER   'NUMBER W MOV   (:) 2+ JMP
12
13
14
15
```

```
Screen:  150
 0    ( CONTROL)
 1 CODE ?STACK   S0 U) S CMP   0 0 SBB   0 COM   0 PUSH   NEXT
 2
 3 : '   -' ABORT" ?" ;
 4 : INTERPRET   BEGIN  -' IF  NUMBER  ELSE  EXECUTE
 5    ?STACK ABORT" STACK EMPTY"  THEN  (ELSE) [ (BACK)
 6
 7 | CODE CLEAR   U R MOV   0 0 SUB
 8       0 BLK U) MOV   0 >IN U) MOV   NEXT
 9 : QUIT   BEGIN  CLEAR  S0 @ 80 EXPECT
10       INTERPRET  ." ok" CR  (ELSE) [ (BACK)
11
12 ' QUIT  'QUIT 1+ !   ' EXIT  ' X 2 - !
13
14
15

Screen:  151
 0 5151
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  152
 0 5151
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  153
 0 ( N-BUFFER MANAGER)
 1 | CODE CORE   W POP   W SHL   I 2 MOV   PREV 4 + # I MOV
 2    NB 1 MOV   BEGIN   LODS   0 SHL   W 0 CMP   -LOOP
 3       2 I MOV   0= NOT IF   W SHR   W PUSH   NEXT
 4    THEN   1 COM   NB 1 ADD   1 SHL
 5 BEGIN   1 PREV MOV   1 HI 1 XCHG B   1 SHL
 6    FIRST 1 ADD   1 PUSH   ' EXIT JMP
 7
 8 | CODE OBTAIN   PREV 2+ 1 MOV   BEGIN   1 SHR   1 INC   NB 1 CMP
 9       0= IF   1 1 SUB   THEN   1 SHL   PREV 1 CMP   0= NOT END
10    1 PREV 2+ MOV   1 W MOV   32767 # 0 MOV
11    PREV 4 + W) 0 XCHG   0 SHL   CS IF   0 SHR
12       1 PREV MOV   1 HI 1 XCHG B   1 SHL   FIRST 1 ADD
13       1 PUSH   0 PUSH   'BUFFER 2+ STA   NEXT
14    THEN   0 # DISK MOV   JMP
15

Screen:  154
 0    ( VECTORED BLOCK)
 1 CODE UPDATE   PREV W MOV   128 #B PREV 5 + W) OR   NEXT
 2 CODE IDENTIFY   PREV 2+ W MOV   PREV 4 + W) POP   NEXT
 3
 4 CODE BUFFER   'BUFFER W MOV   (:) 2+ JMP
 5 CODE BLOCK   'BLOCK W MOV   (:) 2+ JMP
 6 ' BLOCK 2 -  ' WORD 2+ !
 7                                                              CR
 8 : FLUSH   NB @ 0 DO  BUFFER DROP  LOOP ;
 9 : EMPTY-BUFFERS   PREV NB @ 2+ 2* ERASE  FLUSH ;
10
11
12
13
14
15

Screen:  155
 0    ( RAM BLOCKS)
 1 : (BUFFER)   OBTAIN DROP ;
 2 : (BLOCK)   CORE BUFFER  SWAP IDENITFY ;
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  156
 0    ( INTEL-201: 250 BLOCKS, 2 UNITS)   HEX
 1 THERE EQU IOPB  8 ALLOT
 2
 3 | CODE FDC   IOPB # W MOV   80 #B 0 MOV   STOS B   0 POP
 4    0 SHL   0 SHL   0 SHL   CWD   7D0 # 1 MOV   1 DIV
 5    0 0 HI MOV B   LODS B   30D5 ,   STOS   2 0 XCHG
 6    1A #B 1 MOV   1 DIV B   100 # 0 ADD   STOS   1 POP
 7    1 PUSH   1 0 XCHG   ( RAM BIAS   0 # 0 SUB)   STOS
 8    1B08 # 0 MOV   STOS B   1 HI 0 HI SUB B   0 0 HI CMP B
 9       CS IF   0 HI 0 MOV B   THEN   IOPB 2+ STA B
10    0 0 SUB   DISK 2+ STA B   IOPB #B 0 MOV   E1 OUT
11    IOPB 100 / #B 0 MOV   E2 OUT   WAIT JMP
12
13 DECIMAL
14
15

Screen:  157
 0    ( DISK INTERRUPT)   HEX
 1 LABEL FLOPPY   0 PUSH   DS PUSHS   0 IS SSG   0 DS LSG
 2    E1 IN   0 0 OR B   E3 IN   0= NOT IF   DISK 3 + STA B   ELSE
 3       0 DISK 2+ OR B   W PUSH   IOPB # W MOV   IOPB 7 + LDA B
 4       2 W) 0 SUB B   0= IF   DISK W MOV   WAKE # W ) MOV   ELSE
 5          IOPB 7 + STA B   1A #B 0 CMP   CS NOT IF   1A #B 0 MOV
 6             THEN   2 W) 0 XCHG B   3 W) INC B   1 #B 4 W) MOV
 7          0 SHR   0 6 W) ADD B   W 0 XCHG   E1 OUT   0 HI 0 MOV B
 8          E2 OUT   THEN   W POP   THEN   DS POPS   0 POP   IRET
 9 DECIMAL
10
11
12
13
14
15

Screen:  158
 0    ( DISK BLOCKS)   HEX
 1 : (BUFFER)   DISK GET  OBTAIN  FDC [ 6 C, ]  DISK RELEASE ;
 2
 3 : (BLOCK)   OFFSET @ +  PAUSE CORE  BUFFER  DISK GRAB
 4    OVER FDC [ 4 C, ]  SWAP IDENTIFY  DISK RELEASE ;   DECIMAL
 5
 6 : DRIVE   250 * OFFSET ! ;
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  159
 0 ( COMPILER)
 1 FORTH : IMMEDIATE   128 PREVIOUS 8 - +!  0 TEXT FORTH ! ; HOST
 2
 3 | CODE ?ALLOT   0 POP   0 H U) ADD   H U) 0 MOV
 4    250 # 0 ADD   S 0 CMP   CS IF   ' EXIT JMP
 5       THEN   I PUSH   NEXT
 6 : ALLOT   ?ALLOT ABORT" DICTIONARY FULL" [
 7 : C,   HERE C!  1 ALLOT ;      : ,   HERE !  2 ALLOT ;
 8
 9 CODE \   H U) W MOV   MOVS   W H U) MOV   NEXT
10 CODE ?CELL   0 POP   0 PUSH   128 # 0 ADD
11     -256 # 0 AND   0 PUSH   NEXT
12 : ]   BEGIN  -' IF  (NUMBER)  ?CELL IF
13       \ LITERAL ,  ELSE  \ LIT C,  THEN
14    ELSE DUP 9 - @  0< IF EXECUTE  ?STACK ABORT" STACK EMPTY"
15       ELSE 2 - ,  THEN THEN  (ELSE) [ (BACK)

Screen:  160
 0    ( DEFINING WORDS)
 1 CODE [   ' EXIT HERE 2 - !   IMMEDIATE
 2
 3 : IN-LINE   \ LITERAL , ;   IMMEDIATE
 4 : [']   '  \ LITERAL  , ;   IMMEDIATE
 5
 6 | : (;CODE)   R>  LAST @ @ 6 + ! ;
 7 FORTH : ;CODE   \ (;CODE)  R> DROP ;   TARGET
 8 FORTH : DOES>   \ (;CODE)  232 C, (DOES>) HERE - , ;   TARGET
 9
10 | : SMUDGE   32768 LAST @ @ +! ;
11
12
13
14
15

Screen:  161
 0    ( DEFINING WORDS)
 1 CODE CREATE   'CREATE W MOV   (:) 2+ JMP
 2
 3 HERE   'CREATE' ORG
 4 : (CREATE)   32 WORD  4 ALLOT  CURRENT @ HASH
 5    DUP LAST !  DUP @ ,  0 ,  !  ;CODE
 6 'USER' ORG       : USER   CREATE C,  ;CODE
 7 'CONSTANT' ORG   : CONSTANT   CREATE ,  ;CODE                CR
 8 ':' ORG          : :   CURRENT @ CONTEXT !  CREATE
 9                    SMUDGE ] SMUDGE  ;CODE
10 ORG
11 : CVARIABLE   CREATE  1 ALLOT ;
12 : VARIABLE   CVARIABLE  1 ALLOT ;
13
14 : MSG   CREATE  ;CODE      FORTH
15 : MSG   CREATE  DOES>  COUNT TYPE ;
```

```
Screen:  162
 0    ( COMPILER DIRECTIVES)
 1 : (BACK)   HERE 1+ - C, ;
 2 : BEGIN   HERE ;   IMMEDIATE
 3 : END   \ (IF)  (BACK) ;   IMMEDIATE
 4
 5 | : (THEN)   HERE 1- OVER -  SWAP C! ;
 6 : IF   \ (IF)  HERE 0 C, ;   IMMEDIATE
 7 : THEN   (THEN) ;   IMMEDIATE                                    CR
 8 : ELSE   \ (ELSE)  HERE 0 C,  SWAP (THEN) ;   IMMEDIATE
 9 : AGAIN   \ (ELSE)  SWAP (BACK)  (THEN) ;   IMMEDIATE
10
11 : DO   \ (DO)  HERE ;   IMMEDIATE
12 : LOOP   \ (LOOP)  (BACK) ;   IMMEDIATE
13 : /LOOP   \ (/LOOP)  (BACK) ;   IMMEDIATE
14 : +LOOP   \ (+LOOP)  (BACK) ;   IMMEDIATE
15 : ;   \ EXIT  R> DROP ;   IMMEDIATE

Screen:  163
 0    ( GENERIC WORDS)   HEX
 1 : VOCABULARY   CONSTANT  ;CODE      FORTH
 2 : VOCABULARY   CODE ,  DOES>  @ CONTEXT ! ;
 3
 4 0513 VOCABULARY ASSEMBLER
 5 0051 VOCABULARY FORTH
 6
 7 : CODE   CREATE  HERE DUP 1- 1- !  ASSEMBLER ;
 8 : ;CODE   \ (;CODE)  R> DROP  ASSEMBLER ;   IMMEDIATE
 9       (DOES>)
10 : DOES>   \ (;CODE)  E8 C,  IN-LINE HERE - , ;   IMMEDIATE
11 DECIMAL
12
13
14
15

Screen:  164
 0    ( MISC.)
 1 : EMPTY   H 2+ @ H !  GOLDEN CONTEXT 20 MOVE ;
 2
 3 | : STRING   34 WORD  C@ 1+ ALLOT ;
 4 : ABORT"   \ (ABORT")  STRING ;   IMMEDIATE
 5 : ."   \ (.")  STRING ;   IMMEDIATE
 6
 7 : DECIMAL   10 BASE ! ;      : HEX   16 BASE ! ;
 8 : LOAD   >IN 2@ >R >R  0 >IN 2!  INTERPRET  R> R> >IN 2! ;
 9 : (   41 WORD DROP ;   IMMEDIATE
10
11
12
13
14
15
```

```
Screen:  165
 0 ( INITIAL RAM)   HEX
 1 | VARIABLE TEST  6 ALLOT   ( A55A, TODAY, TICKS)
 2
 3 | CREATE RAM   51 ,  0 ,  0 ,  0 ,  0 ,  0 ,  0 ,  0 ,  0 ,
 4    51 ,  ' (BLOCK) ,  ' (BUFFER) ,  0 ,  ' (NUMBER) ,
 5    ' (CREATE) ,  0 ,
 6
 7 | CREATE 'OPERATOR   EA2E ,  OPERATOR @ ,  ZERO ,  0 ,
 8    OPERATOR @ 80 - ,  0 ,  0 ,  ' (CR) ,  ' (PAGE) ,  ' (TYPE) ,
 9    D8 ,  D8 ,  ' (EXPECT) ,  THERE  DUP ,  ,  0 ,  0 ,  0A ,
10
11 FORTH : BIAS   0  ZERO 10 M*  D+  OPERATOR @ 0 D-
12    OPERATOR @ SWAP  10 M/ SWAP ;      DECIMAL
13
14 | CREATE INTERRUPTS
15    <TYPE> BIAS , ,  <EXPECT> BIAS , ,  ZERO FLOPPY , ,

Screen:  166
 0    ( INITIALIZE I/O)   HEX
 1 | CREATE OWN   ( USART)  01DA , 40DA , 4EDA , 36DA ,
 2    ( PIT)  34D6 , 00D0 , 06D0 , 70D6 , B6D6 , 08D4 , 00D4 ,
 3    ( PPI)  BFCE ,  ( PIC)  11C0 , 08C2 , 00C2 , 0FC2 , 04C2 ,
 4
 5 | CODE PORTS   0 ,   HERE ' PORTS 2 - !
 6    1 POP   0 POP   I PUSH   I 0 XCHG   BEGIN
 7       2 W) IMUL   LODS B   2 0 XCHG   LODS B   (2) OUT   LOOP
 8    STI   I POP   NEXT
 9 DECIMAL
10
11
12
13
14
15

Screen:  167
 0    ( POWER UP)   HEX
 1       PROM 2C +   PROM 4 -
 2 LABEL START ]   RAM GOLDEN 20 MOVE  IN-LINE PROM 40 MOVE
 3    INTERRUPTS IN-LINE 0C MOVE  'OPERATOR STATUS  DUP 60 ERASE
 4    24 MOVE  OWN 11 PORTS  TEST @ A55A - IF  EMPTY-BUFFERS
 5       A55A TEST !  THEN  EMPTY  PAGE ." up" CR  RESET [
 6
 7 ASSEMBLER   HERE
 8    0 ES LSG   0 SS LSG   0 DS LSG   START # I MOV
 9    OPERATOR @   DUP # U MOV   U R MOV   80 - # S MOV
10    STD B   NEXT
11
12 HERE  PROM 10 - ORG   ZERO # 0 MOV   EA C,  SWAP ,  ZERO ,
13 ORG      DECIMAL
14
15
```

```
Screen:  168
 0 5757
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  169
 0 5757
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  170
 0 5757
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  171
 0 ( PROM EDITOR)   HEX
 1 0015 VOCABULARY EDITOR      00BF VOC FORTH !   HOST
 2
 3    DECIMAL   11 LOAD  18 20 THRU   HEX
 4
 5 000B VOC FORTH !   HOST      CR  HERE U.   DECIMAL
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  172
 0 ( PROM ASSEMBLER)   HEX
 1 00BD VOC FORTH !   0 ?\ !      ASSEMBLER DEFINITIONS
 2 FORTH : 2CONSTANT   CONSTANT C, ;
 3
 4       HERE 3 -  0F C,  0 C,  0E C,  0C C,  0D C,
 5    : )   0  SWAP IN-LINE +  C@ ;
 6       DECIMAL   12 17 THRU   HEX                              CR
 7    WAIT CONSTANT WAIT      WAKE CONSTANT WAKE
 8    ZERO CONSTANT ZERO      : U)   STATUS - 3) ;
 9    : INTERRUPT   ZERO ROT  ROT 2* 2*  PROM + 2! ;
10    CODE BIAS   W POP   W SHL   W SHL   PROM # W ADD   HERE 1+
11       10 # 0 MOV   2 W) MUL   W ) 0 ADD   0 #B 2 ADC   1 POP
12       1 0 SUB   0 #B 2 SBB   STOS   1 0 XCHG   DIV   STOS   NEXT
13
14 000B VOC FORTH !   1 ?\ !      HOST DEFINITIONS
15    CR  HERE U.   DECIMAL

Screen:  173
 0 5858
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 174
 0 5959
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 175
 0 5959
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 176
 0 5959
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen:  177
 0 6060
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  178
 0 6060
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  179
 0 6060
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  180
 0 6161
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  181
 0 6161
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  182
 0 6161
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  183
 0 6262
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  184
 0 6262
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  185
 0 6262
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  186
 0 6363
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  187
 0 6363
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  188
 0 6363
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  189
 0 6464
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  190
 0 6464
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  191
 0 6464
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  192
 0 6565
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  193
 0 6565
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  194
 0 6565
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  195
 0 6666
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  196
 0 6666
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  197
 0 6666
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  198
 0 6767
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  199
 0 6767
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  200
 0 6767
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  201
 0 6868
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  202
 0 6868
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  203
 0 6868
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 204
 0 6969
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 205
 0 6969
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 206
 0 6969
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen:  207
 0 7070
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  208
 0 7070
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  209
 0 7070
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  210
 0 7171
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  211
 0 7171
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  212
 0 7171
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  213
 0 7272
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  214
 0 7272
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  215
 0 7272
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  216
 0 7373
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  217
 0 7373
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  218
 0 7373
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  219
 0 7474
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  220
 0 7474
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  221
 0 7474
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  222
 0 7575
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  223
 0 7575
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  224
 0 7575
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  225
 0 7676
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  226
 0 7676
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  227
 0 7676
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  228
 0 7777
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  229
 0 7777
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  230
 0 7777
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  231
 0 7878
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  232
 0 7878
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  233
 0 7878
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  234
 0 7979
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  235
 0 7979
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  236
 0 7979
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  237
 0 8080
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  238
 0 8080
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  239
 0 8080
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  240
 0 8181
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  241
 0 8181
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  242
 0 8181
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  243
 0 8282
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  244
 0 8282
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  245
 0 8282
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  246
 0 8383
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  247
 0 8383
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15


Screen:  248
 0 8383
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  249
 0 8484
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  250
 0 ( ERROR REPORING AND CONTROL ROUTINES )
 1
 2 VARIABLE ELEVEL   VARIABLE WLEVEL
 3 VARIABLE ENUMBER   VARIABLE WNUMBER
 4
 5 : .LINE  BLOCK  SWAP 64 * +  64 -TRAILING >TYPE SPACE  ;
 6
 7 : B&L#  2*  16 /MOD  7926 + ;
 8
 9 : .WARN  WNUMBER !  WLEVEL @  ?DUP IF  CR  ." WARNING: "
10    WNUMBER @  B&L# .LINE   2 = IF  TALKER ACTIVATE
11    ." WARN ING   "  WNUMBER @  B&L# SWAP 1+ SWAP .LINE
12    CR  0 QUIT  THEN  THEN ;
13
14 2 WLEVEL !     2 ELEVEL !
15

Screen:  251
 0    ( ERROR REPORTING CONT. )
 1
 2 : (.ERROR)  TALKER ACTIVATE  ." AIROR  "  ENUMBER @ B&L#
 3    SWAP 1+ SWAP  .LINE CR 0 QUIT  [
 4
 5 : .ERROR  ENUMBER !  ELEVEL @  ?DUP  IF  2 = IF
 6    (.ERROR)   THEN  CR ." ERROR: "  ENUMBER @ B&L#
 7    .LINE  1 ABORT"  "   THEN ;
 8
 9
10
11
12
13
14
15

Screen:  252
 0 ( FREEDOM 100 FUNCTION KEY ROUTINES )
 1
 2 : fcompute  75 *  750 /MOD  7932 +  ;
 3
 4 : FKEY  CREATE  DUP 1+ C,  fcompute ,  274 + ,
 5    DOES>  8 EMIT  8 EMIT ." F"  DUP C@ .  1+  2@
 6    BLOCK +  S0 @ 75 MOVE  75 CNT ! 0. >IN 2!  INTERPRET  ;
 7
 8 : FKEYDEF  CONSTANT  DOES> @  fcompute  BLOCK
 9    274 + +  DUP 75 BLANK  92 WORD COUNT  ROT SWAP MOVE  UPDATE ;
10
11 : FKEYS  PAGE  2 7932 .LINE  CR
12    20 0 DO CR ." F" I 1+ . ." = "  I  fcompute
13    BLOCK +  274 +  75 -TRAILING TYPE   LOOP CR ;
14
15
```

```
Screen:  253
 0    ( FUNCTION KEY ASSIGNING WORDS )
 1
 2  0 FKEYDEF F1=     1 FKEYDEF F2=     2 FKEYDEF F3=
 3  3 FKEYDEF F4=     4 FKEYDEF F5=     5 FKEYDEF F6=
 4  6 FKEYDEF F7=     7 FKEYDEF F8=     8 FKEYDEF F9=
 5  9 FKEYDEF F10=   10 FKEYDEF F11=   11 FKEYDEF F12=
 6 12 FKEYDEF F13=   13 FKEYDEF F14=   14 FKEYDEF F15=
 7 15 FKEYDEF F16=   16 FKEYDEF F17=   17 FKEYDEF F18=
 8 18 FKEYDEF F19=   19 FKEYDEF F20=
 9
10
11
12
13
14
15

Screen:  254
 0    ( FUNCTION KEY DEFINITIONS )
 1
 2 ( ctrl-A @ )  0 FKEY @           ( ctrl-A ` )  10 FKEY `
 3 ( ctrl-A A )  1 FKEY A           ( ctrl-A a )  11 FKEY a
 4 ( ctrl-A B )  2 FKEY B           ( ctrl-A b )  12 FKEY b
 5 ( ctrl-A C )  3 FKEY C           ( ctrl-A c )  13 FKEY c
 6 ( ctrl-A D )  4 FKEY D           ( ctrl-A d )  14 FKEY d
 7 ( ctrl-A E )  5 FKEY E           ( ctrl-A e )  15 FKEY e
 8 ( ctrl-A F )  6 FKEY F           ( ctrl-A f )  16 FKEY f
 9 ( ctrl-A G )  7 FKEY G           ( ctrl-A g )  17 FKEY g
10 ( ctrl-A H )  8 FKEY H           ( ctrl-A h )  18 FKEY h
11 ( ctrl-A I )  9 FKEY I           ( ctrl-A i )  19 FKEY i
12
13
14
15

Screen:  255
 0 8686
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  256
 0 8787
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  257
 0 8787
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  258
 0 ( BIT MANIPULATION ROUTINES FOR 8086/88  CARL MYERHOLTZ 9/15/82)
 1
 2 CODE +BIT  ( bit# add )   W POP   1 POP   1 #B 0 MOV
 3    0 SHL V   0 W ) OR B   NEXT
 4
 5 CODE -BIT  ( bit# add )   W POP   1 POP   254 #B 0 MOV
 6    0 ROL B V   0 W ) AND B   NEXT
 7
 8 CODE ?BIT   ( bit# add )   W POP   1 POP   1 # 0 MOV
 9    0 SHL V   W ) 0 AND B   0 SAR V   0 PUSH   NEXT
10
11
12
13
14
15
```

```
Screen:  259
 0 ( :CASE FOR 8086/88   CARL MYERHOLTZ  9/15/82 )
 1
 2 : :CASE  CREATE  ]  ;CODE   W INC   W INC   0 POP   0 SHL
 3    0 W ADD   W ) W MOV    W ) LIP
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  260
 0 ( COLLECTIVE CASE CONSTRUCT FOR 8086/88 )
 1 : [COMPILE] ' 2- , ;  IMMEDIATE
 2 ( cases)  HERE DUP 2+ , ASSEMBLER   I INC   I INC   NEXT
 3 : CASES   \ [ , ] HERE 0 , ;    IMMEDIATE
 4
 5 ( <case)  HERE DUP 2+ , ASSEMBLER   0 POP   2 POP   2 0 SUB
 6    0= NOT IF  2 PUSH   I ) I MOV  NEXT  THEN  I INC  I INC NEXT
 7 : <CASE   \ [ , ] HERE 0 , ;   IMMEDIATE
 8
 9 : ENDCASE  HERE SWAP ! ;   IMMEDIATE
10 ( case>)  HERE DUP 2+ , ASSEMBLER   I ) I MOV  I ) I MOV  NEXT
11 : ECASE>   \ [ , ] OVER , [COMPILE] ENDCASE ;  IMMEDIATE
12
13 ( also)   HERE DUP 2+ , ASSEMBLER   I ) I MOV   NEXT
14 : ALSO   \ [ , ] HERE 0 , SWAP [COMPILE] ENDCASE ;
15                                                  IMMEDIATE

Screen:  261
 0 ( ARRAY ROUTINES FOR 8086/88    CARL MYERHOLTZ 9/14/82 )
 1
 2 : CARRAY  CREATE  ALLOT  ;CODE   W INC   W INC   0 POP
 3    W 0 ADD   0 PUSH   NEXT
 4
 5 : ARRAY  CREATE  2* ALLOT  ;CODE   W INC   W INC   0 POP
 6    0 SHL   W 0 ADD   0 PUSH   NEXT
 7
 8 : 2ARRAY  CREATE  2* 2* ALLOT  ;CODE   W INC   W INC   0 POP
 9    0 SHL   0 SHL   W 0 ADD   0 PUSH   NEXT
10
11
12
13
14
15
```

```
Screen:  262
 0 ( 2-D MATRIX DEFINITIONS FOR 8086/88  CARL MYERHOLTZ  9/15/82 )
 1
 2 : CMATRIX  ( #r #c )  CREATE  2DUP , ,  * ALLOT  ;CODE
 3   W INC  W INC   0 POP   W ) MUL   4 # W ADD   0 W ADD
 4   0 POP  W 0 ADD   0 PUSH   NEXT
 5
 6 : MATRIX  ( #r #c )  CREATE  2* DUP , * ALLOT  ;CODE
 7    W INC   W INC   1 POP   0 POP   W ) MUL   W INC   W INC
 8    W 0 ADD   1 SHL   1 0 ADD   0 PUSH   NEXT
 9
10 : 2MATRIX  ( #r #c )  CREATE  2* DUP , 2* * ALLOT  ;CODE
11    W INC   W INC   0 POP   0 SHL   0 SHL   W ) MUL   4 # W ADD
12    0 W ADD   0 POP   0 SHL   0 SHL   W 0 ADD   0 PUSH   NEXT
13
14
15

Screen:  263
 0 ( INPUT CONVERSION FOR ONE DECIMAL PLACE )
 1
 2 CODE ?#R ( add - add cnt ) 1 1 SUB  W POP   W PUSH   W ) 1 MOV B
 3    W INC   46 #B 0 MOV   REP B   SCAS B   1 PUSH   NEXT
 4
 5 : number  ?#R  1 = IF  (NUMBER) DROP  ELSE  (NUMBER)  THEN ;
 6
 7 ' number 'NUMBER !
 8
 9
10
11
12
13
14
15

Screen:  264
 0 8989
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  265
 0 ( :CASE FOR 8086/88   CARL MYERHOLTZ  9/15/82 )
 1
 2 FORTH : :CASE  CREATE  ]  ;CODE   W INC   W INC   0 POP
 3    0 SHL   0 W ADD   W ) W MOV   W ) LIP
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  266
 0 ( ARRAY ROUTINES FOR 8086/88     CARL MYERHOLTZ 9/14/82 )
 1
 2 FORTH : CARRAY   CREATE  ALLOT  ;CODE   W INC   W INC
 3    0 POP   W 0 ADD   0 PUSH   NEXT
 4
 5 FORTH : ARRAY  CREATE  2* ALLOT  ;CODE   W INC   W INC
 6    0 POP   0 SHL   W 0 ADD   0 PUSH   NEXT
 7
 8 FORTH : 2ARRAY  CREATE  2* 2* ALLOT  ;CODE   W INC
 9    W INC   0 POP   0 SHL   0 SHL   W 0 ADD   0 PUSH   NEXT
10
11
12
13
14
15

Screen:  267
 0 ( 2-D MATRIX DEFINITIONS FOR 8086/88  CARL MYERHOLTZ 9/15/82 )
 1
 2 FORTH : CMATRIX ( #r #c )   CREATE  DUP ,  * ALLOT
 3    ;CODE   W INC   W INC   1 POP   0 POP   W ) MUL   W INC
 4       W INC   W 0 ADD   1 0 ADD   0 PUSH   NEXT
 5
 6 FORTH : MATRIX ( #r #c )   CREATE  2* DUP ,  * ALLOT
 7    ;CODE   W INC   W INC   1 POP   0 POP   W ) MUL   W INC
 8       W INC   W 0 ADD   1 SHL   1 0 ADD   0 PUSH   NEXT
 9
10 FORTH : 2MATRIX ( #r #c )   CREATE 2* 2* DUP ,   * ALLOT
11    ;CODE   W INC   W INC   1 POP   0 POP   W ) MUL
12    W INC   W INC   W 0 ADD   1 SHL   1 SHL   1 0 ADD
13    0 PUSH   NEXT
14
15
```

```
Screen:  268
 0 9191
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  269
 0 9191
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  270
 0 9191
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  271
 0 9292
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  272
 0 9292
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  273
 0 9292
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  274
 0 9393
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  275
 0 9393
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  276
 0 9393
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  277
 0 9494
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  278
 0 9494
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  279
 0 9494
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  280
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  281
 0 9595
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  282
 0 9595
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  283
 0 9696
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  284
 0 9696
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  285
 0 9696
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  286
 0 9797
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  287
 0 9797
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  288
 0 9797
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  289
 0 9898
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  290
 0 9898
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  291
 0 9898
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  292
 0 9999
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  293
 0 9999
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  294
 0 9999
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  295
 0 100100
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  296
 0 100100
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  297
 0 100100
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  298
 0 101101
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  299
 0 101101
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  300
 0 ( CONSTANTS FOR INTERPROCESSOR HARDWARE  COOR/COMM )
 1
 2 HEX
 3 F000 CONSTANT HSTAT     F008 CONSTANT SSTAT
 4
 5 F010 CONSTANT CB1       F011 CONSTANT CB2
 6 F012 CONSTANT CB3       F013 CONSTANT CB4
 7
 8 F014 CONSTANT BC-REG
 9
10 F015 CONSTANT STATOUT
11
12 F016 CONSTANT INTSL#     F017 CONSTANT INTDATA
13
14 DECIMAL
15
```

```
Screen:  301
 0 ( COMMUNICATOR INTERPROCESSOR MEMORY FUNCTIONS )
 1    ( SLAVE SELECTION FUNCTIONS )
 2
 3 VARIABLE SEG-REG   2 ALLOT
 4
 5 HEX
 6 CODE #SLAVE  ( slave#)  0 POP   0 0 OR   0= NOT IF   0C # 1 MOV
 7    0 SHL V   8000 # 0 ADD    THEN   0 SEG-REG MOV   NEXT
 8
 9 DECIMAL
10
11 : >SLAVE   ( source slave#, destination slave# )
12        #SLAVE   SEG-REG @ SEG-REG 2+ !   #SLAVE ;
13
14
15

Screen:  302
 0    ( COMMUICATOR INTERPROCESSOR ACCESS OPERATORS )
 1
 2 CODE I@   W POP   DS PUSHS   SEG-REG DS LSG   W ) 0 MOV
 3    DS POPS  0 PUSH   NEXT
 4
 5 CODE I!   W POP   0 POP  ES PUSHS   SEG-REG ES LSG   STOS
 6    ES POPS   NEXT
 7
 8 CODE IC@   W POP   0 0 SUB   DS PUSHS   SEG-REG DS LSG
 9    W ) 0 MOV B   DS POPS   0 PUSH NEXT
10
11 CODE IC!   W POP   0 POP   ES PUSHS   SEG-REG ES LSG   STOS B
12    ES POPS   NEXT
13
14
15

Screen:  303
 0    ( INTERPROCESSOR ARRAY OPERATORS )
 1
 2 CODE IPMOVE   1 POP   W POP   0 POP   DS PUSHS   ES PUSHS
 3    SEG-REG 2+ ES LSG  SEG-REG DS LSG  1NZ IF   I 0 XCHG   W ROR
 4    W ROL   CS IF   MOVS B   1 DEC   THEN   1 SHR   REP MOVS
 5    CS IF   MOVS B   THEN   I 0 XCHG   THEN
 6    ES POPS   DS POPS   NEXT
 7
 8 CODE SMOVE   0 0 SUB   0 INC   0 BC-REG MOV B
 9    1 POP   W POP   0 POP   ES PUSHS   DS PUSHS
10    SEG-REG 2+ ES LSG  SEG-REG DS LSG  1NZ IF   I 0 XCHG   W ROR
11    W ROL   CS IF   MOVS B   1 DEC   THEN   1 SHR   REP MOVS
12    CS IF   MOVS B   THEN   I 0 XCHG   THEN   0 0 SUB   1 SEG
13    0 BC-REG MOV B   HSTAT 0 MOV B   0 SEG   HSTAT 0 MOV B
14    DS POPS   ES POPS   NEXT
15
```

```
Screen:  304
 0 ( COMMAND BUFFER DRIVERS )     HEX
 1   VARIABLE SLAVE#
 2
 3 CODE ?FULL   SLAVE# W MOV   HSTAT # W ADD   W ) 0 MOV B
 4    10 #B 0 TEST   0= NOT IF   I DEC   I DEC   ' PAUSE JMP
 5    THEN   NEXT
 6
 7 : CBUFFER  ( value slave# )  DUP  SLAVE# !  CB4 C!
 8    ( ?FULL )  CB1 !  ;
 9
10 : SPUSH  ( data sl# )  CBUFFER  0B3 CB3 C!  ;
11
12 : SCMD   ( add sl# )  CBUFFER  0B1 CB3 C!  ;
13
14 DECIMAL
15

Screen:  305
 0 ( SLAVE HOLD AND RESET )     HEX
 1
 2 : HLD  ( slave# )   CB4 C!  0F0 CB3 C! ;
 3
 4 : RESET  ( slave# )   CB4 C!  03 CB3 C! ;
 5
 6 DECIMAL
 7
 8 HERE  ]  ABORT" BUS TIMEOUT ERROR" [  ASSEMBLER
 9    BEGIN   SWAP # I MOV    00 #B 0 MOV   0 BC-REG MOV B
10    00 # 0 MOV   0 DS LSG   0 ES LSG   STI   NEXT   09 INTERRUPT
11
12
13
14
15

Screen:  306
 0    ( INTERPROCESSOR MEMORY PRINT )
 1    ( NORMALLY LOADED FOR DEBUGGING ONLY )
 2
 3 : I?   I@  . ;
 4
 5 : IDUMP   OVER + SWAP DO  CR I 5 U.R SPACE  I 16 + I' MIN I DO
 6    I 7 AND 0= 2* SPACES  I IC@ 3 U.R  LOOP  16 /LOOP  SPACE ;
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  307
 0 ( SLAVE SEARCH AND LABEL ROUTINES )
 1 CODE +I   0 POP   0 R ) ADD   NEXT
 2
 3 : SEARCH   32 WORD  2@   ROT 8046 +  BLOCK  >R
 4    BEGIN  6 +I  2DUP  I 2@  DUP NOT
 5       ABORT" Slave command not found in SCAT "
 6    D- D0=  END  2DROP   R> 4 + @  ;
 7
 8 : label  CREATE  C,  DOES>  C@ DUP  SEARCH 2+  CREATE
 9    SWAP  , , DOES>  2@  #SLAVE ;
10
11 1 label 1LABEL   ( slave-name  new-name )
12 2 label 2LABEL   ( slave-name  new-name )
13 3 label 3LABEL   ( slave-name  new-name )
14 4 label 4LABEL   ( slave-name  new-name )
15

Screen:  308
 0 ( SLAVE COMMAND )
 1
 2 CODE [SCMD]   LODS   0 PUSH   0 0 SUB   LODS B   0 PUSH
 3    ' SCMD 2- # W MOV  W ) LIP
 4
 5 : SLn  CREATE C,  DOES>  C@ DUP SEARCH  \ [SCMD] , C, ;
 6
 7 1  SLn SL1     IMMEDIATE
 8 2  SLn SL2     IMMEDIATE
 9 3  SLn SL3     IMMEDIATE
10 4  SLn SL4     IMMEDIATE
11
12
13
14
15

Screen:  309
 0 ( SLAVE PUSH )
 1
 2 CODE [SPUSH]   0 0 SUB   LODS B  0 PUSH   ' SPUSH 2- # W MOV
 3     W ) LIP
 4
 5 : PUSHES  CREATE  C,  DOES>  C@  \ [SPUSH]  C,  ;
 6
 7 1 PUSHES 1PUSH     IMMEDIATE
 8 2 PUSHES 2PUSH     IMMEDIATE
 9 3 PUSHES 3PUSH     IMMEDIATE
10 4 PUSHES 4PUSH     IMMEDIATE
11
12
13
14
15
```

```
Screen:  310
 0 105105
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  311
 0 105105
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  312
 0 ( SLAVE COMPILER )
 1
 2 CREATE SLAVE-COMPILER-CONTROL ( complier, system, application)
 3                                390 ,     420 ,   421 ,
 4                                390 ,     420 ,   422  ,
 5                                390 ,     420 ,   423 ,
 6                                390 ,     420 ,   423 ,
 7
 8
 9 : SCOMPILE   DUP 2DUP 2DUP  >R #SLAVE  SLAVE# !  HLD
10    0 DRIVE DUP 8046 + BLOCK  1024 ERASE  UPDATE FLUSH
11    CR  ." COMPILING SLAVE " .  1- 6 * SLAVE-COMPILER-CONTROL +
12    DUP @ LOAD  2+ DUP @ LOAD  2+ @ LOAD  R> RESET  EMPTY
13    ['] ?CREATE  'CREATE !  DEPTH ?DUP IF  CR  .
14    ."  Items on stack "  THEN ;
15
```

```
Screen:  313
 0 ( SLAVE COMMAND ACCESS TABLE DISLAY )
 1
 2 : SCAT  8046 +  BLOCK DUP @ 0 DO  6 + DUP  CR  DUP C@ 2 U.R
 3    2 SPACES  DUP 1+ 3 TYPE 4 SPACES 4 + @  HEX U.
 4    DECIMAL LOOP ;
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  314
 0 106106
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  315
 0 106106
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  316
 0 ( STATUS LINE)
 1 EXIT
 2 : PLEASE   ." Please be nice to me Unkil Mikey " CR  ;
 3
 4 .status PLEASE
 5
 6 FORGET PLEASE
 7
 8
 9
10
11
12
13
14
15

Screen:  317
 0 107107
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  318
 0 107107
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  319
 0 108108
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  320
 0 108108
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  321
 0 108108
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 322
 0 109109
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 323
 0 109109
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 324
 0 109109
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen:  325
 0 110110
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  326
 0 110110
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  327
 0 110110
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  328
 0 111111
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  329
 0 111111
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  330
 0 111111
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  331
 0 112112
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  332
 0 112112
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  333
 0 112112
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  334
 0 113113
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  335
 0 113113
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  336
 0 ( Slaves - Extensions )
 1
 2 CODE INPUT   2 POP   0 0 SUB   (2) IN   0 PUSH   NEXT
 3 CODE OUTPUT   2 POP   0 POP   (2) OUT   NEXT
 4
 5 : DEPTH   'S 2+ S0 @ SWAP - 2/ ;
 6
 7 : BIN 2 BASE ! ;
 8 : OCTAL 8 BASE ! ;
 9
10 CODE WITHIN   2 POP   1 POP   0 POP   W W SUB
11    2 0 CMP   0< IF   1 0 CMP   0< NOT IF   W INC
12    THEN   THEN   W PUSH   NEXT
13
14
15
```

```
Screen:  337
 0 ( Slaves - Powerful Words )
 1
 2 : (ASSIGN)   R> 2+ SWAP ! ;
 3
 4 : ASSIGN   BEGIN \ (ASSIGN) \ [ 2 - @ , ] ;   IMMEDIATE
 5
 6 : S? ( non-destructive stact print )   CR DEPTH ?DUP IF  0 DO
 7    'S I 2* + @ . CR  LOOP  ELSE  ." Empty" CR  THEN ;
 8
 9
10
11
12
13
14
15

Screen:  338
 0 ( Slaves - 32-bit Output )
 1
 2 CODE /DIGIT   0 POP   1 POP   2 2 SUB   BASE U) DIV
 3    1 0 XCHG   BASE U) DIV   0 PUSH   1 PUSH   2 PUSH   NEXT
 4
 5 : #   /DIGIT  NUMERAL HOLD ;
 6 : #S   BEGIN  #  2DUP D0= END ;
 7
 8 : (D.)   SWAP OVER DABS  <# #S SIGN #> ;
 9 : D.   (D.) TYPE SPACE ;
10 : D.R   >R (D.)  R> OVER - SPACES  TYPE ;
11
12
13
14
15

Screen:  339
 0 ( Slaves - 32-bit Input )
 1
 2 CODE *DIGIT   W POP   2 POP   0 POP   1 POP   2 PUSH
 3    BASE U) MUL   1 0 XCHG   BASE U) MUL   W 0 ADD   2 1 ADC
 4    2 POP   0 PUSH   1 PUSH   2 PUSH   NEXT
 5
 6 CODE +I'   2 R) INC   NEXT
 7
 8 : (NUMBER)   0 >R  DUP 1+ C@  45 = DUP >R  +  0 0
 9    ROT  BEGIN BEGIN  DIGIT IF  *DIGIT AGAIN  DUP C@
10       DUP 58 =  SWAP 44 48 WITHIN  + IF  +I' AGAIN
11    C@ 32 - ABORT" ?"  R> IF DMINUS THEN  R> 0= IF DROP THEN ;
12
13
14
15
```

```
Screen:  340
 0 115115
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  341
 0 115115
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  342
 0 115115
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  343
 0 116116
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  344
 0 116116
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  345
 0 116116
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  346
 0 117117
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  347
 0 117117
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  348
 0 117117
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  349
 0 118118
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  350
 0 118118
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  351
 0 118118
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  352
 0 119119
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  353
 0 119119
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  354
 0 119119
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 355
 0 120120
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 356
 0 120120
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 357
 0 120120
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen:  358
 0 121121
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  359
 0 121121
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  360
 0 121121
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 361
 0 122122
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 362
 0 122122
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 363
 0 122122
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 364
 0 123123
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 365
 0 123123
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 366
 0 123123
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 367
 0 124124
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 368
 0 124124
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 369
 0 124124
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen:  370
 0 125125
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  371
 0 125125
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  372
 0 125125
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 373
 0 126126
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 374
 0 126126
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 375
 0 126126
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 376
 0 127127
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 377
 0 127127
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 378
 0 127127
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen:  379
 0 128128
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  380
 0 128128
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  381
 0 128128
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  382
 0 129129
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  383
 0 129129
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  384
 0 129129
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  385
 0 130130
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  386
 0 130130
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  387
 0 130130
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  388
 0 131131
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  389
 0 131131
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  390
 0 ( TARGET COMPILER)      DECIMAL ' <CREATE> 'CREATE !
 1 CREATE HEAD  16 ALLOT   CREATE WDS  4 ALLOT   CREATE ?\  1 ,
 2 VARIABLE W0   VARIABLE 'H   VARIABLE 'R   VARIABLE VOC
 3    ( META)  91 LOAD
 4  ( : LOG   BASE @ HEX  OVER U.  BASE !
 5       >IN @  32 WORD  COUNT 1+ TYPE  >IN !   ; )
 6    :  LOG ;
 7 : ORG   'H ! ;      : GAP   'H +! ;
 8 : HERE   'H @ ;     : THERE   HERE ;      HEX
 9 : WINDOW   DUP ORG  W0 !  0 'R !  000B VOC !  HEAD 10 ERASE ;
10 : |   2 WDS ! ;      DECIMAL  392 LOAD
11   ' NULL 'CR !     ( ACCESS TO DISK)  394 LOAD
12 ( ASSEMBLER)  95 LOAD
13 (  : LOAD   CR  DUP . LOAD ;    : CR     CR SPACE ; )
14 : THRU   1+ SWAP DO  FORTH I HOST LOAD  LOOP ;
15 : ALLOT   GAP ;
```

```
Screen:  391
 0 132132
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  392
 0 ( SLAVE COMMAND )
 1
 2 : COMMAND  SLAVE# @  8046 + BLOCK  DUP DUP   1 SWAP +!  @  6 * +
 3  >R  TEXT 2@  I 2!  PREVIOUS  2-  R> 4 + !  UPDATE ;  IMMEDIATE
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  393
 0 132132
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  394
 0     ( COMPILER TO SLAVE MEMORY )
 1 : DICTIONARY   DUP WDS 2! ;
 2
 3
 4 : C@   IC@ ;      : C!  IC! ;
 5 : @   DUP PTR - IF  I@  ELSE @  THEN ;
 6 : !  I! ;
 7 : ,   HERE !  2 GAP ;      : C,   HERE C!  1 GAP ;
 8 : !   DUP PTR - IF  !  ELSE  FORTH !  THEN ;
 9 : 2!   SWAP  OVER !  2+ ! ;
10 : +!   SWAP  OVER @ +  SWAP ! ;
11
12 : DUMP   OVER + SWAP DO  CR I 5 U.R SPACE  I 16 + I' MIN I DO
13    I 7 AND 0= 2* SPACES  I C@ 3 U.R  LOOP  16 /LOOP  SPACE ;
14
15

Screen:  395
 0 133133
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  396
 0 133133
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  397
 0 134134
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  398
 0 134134
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  399
 0 134134
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  400
 0 135135
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  401
 0 135135
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  402
 0 135135
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  403
 0 136136
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  404
 0 136136
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  405
 0 136136
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  406
 0 137137
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  407
 0 137137
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  408
 0 137137
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  409
 0 138138
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  410
 0 138138
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  411
 0 138138
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  412
 0 139139
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  413
 0 139139
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  414
 0 139139
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  415
 0 140140
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  416
 0 140140
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  417
 0 140140
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  418
 0 141141
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  419
 0 ( Extras for Slave Processors )
 1
 2 28 LOAD ( Unsigned )
 3 29 LOAD 31 33 THRU ( 32-bit and string stuff )
 4 337 339 THRU ( 32-bit I/O )
 5 424 LOAD ( Tasks )
 6 265 267 THRU ( Misc. Utilities )
 7
 8
 9
10
11
12
13
14
15

Screen:  420
 0 ( 8086 Slave Metaforth )   HEX   HOST DEFINITIONS
 1 : ALLOT GAP ;   HOST
 2 100 EQU PROM   0 EQU ZERO
 3    8 DICTIONARY  0 WINDOW  100 ORG  (   400 ALLOT)
 4 CODE X   HERE 8 - HEAD FORTH !   HEAD DUP 2+ 0E MOVE
 5    2080 PREVIOUS 8 - HOST !   DECIMAL
 6
 7 ( NUCLEUS)  123 131 THRU  ( COMPILER)  96 98 THRU
 8 ( LEVEL 1) 132 134 THRU
 9 ( MULTI-TASK)  435 436 THRU   ( Extensions )   336 LOAD
10 ( TERMINAL)  138 LOAD   ( OUTPUT)  144 146 THRU
11 ( INTERPRETER) 147 150 THRU   ( SLAVE)   446 448 THRU
12 ( COMPILER)  159 164 THRU
13 (   HEX  CR  HERE U.  THERE U.  DECIMAL )
14 ( EDITOR  171 LOAD)  ( ASSEMBLER  172 LOAD )
15
```

```
Screen:  421
 0 ( Ion Path - System Load Block )
 1
 2 HEX  4000   CREATE OPERATOR  60 - ,  DECIMAL
 3
 4 ( Terminal )  137 LOAD  139 LOAD  141 142 THRU
 5 ( Extras )  419 LOAD
 6 601 LOAD   ( Mass Spec Applications Load Block )
 7 462 LOAD   ( Startup )
 8 ( Initialize ) 465 467 THRU  HEX
 9
10 ( LINKS )  HOST  HEAD RAM 2+ 10 MOVE
11 ( VECTOR SEED )  ZERO <IRET>  PROM 4 - 2!
12 DECIMAL HERE 8 U.R {  bytes used } CR   FLUSH
13
14 EMPTY ' (CR) 'CR !
15

Screen:  422
 0 ( Reduction - System Load Block )
 1
 2 HEX   0 NB !  C000 400 NB @ * -  DUP FIRST !
 3    CREATE OPERATOR  60 - ,  DECIMAL
 4 ( TERMINAL ) 137 LOAD  139 LOAD  141 142 THRU
 5 ( Extras )    419 LOAD   35 LOAD ( Scaling )
 6 ( Graphics )   480 LOAD
 7 ( MATH         540 LOAD )
 8 ( Mass Spec Applications Load Block )     602 LOAD
 9 ( Startup)  463 LOAD
10 ( Initialize ) 468 470 THRU  HEX
11 ( LINKS )  HOST  HEAD RAM 2+ 10 MOVE
12 ( VECTOR SEED )  ZERO <IRET>  PROM 4 - 2!
13 DECIMAL HERE 8 U.R {  bytes used } CR  FLUSH
14 EMPTY ' (CR) 'CR !
15

Screen:  423
 0 ( Detection - System Load Block )
 1
 2 HEX  4000   CREATE OPERATOR  60 - ,  DECIMAL
 3
 4 ( Terminal )   137 LOAD  139 LOAD  141 142 THRU
 5 ( Extras )   419 LOAD
 6 ( Mass Spec Applications Load Block )  603 LOAD
 7
 8 ( Startup)   464 LOAD
 9 ( INITIALIZE ) 471 473 THRU  HEX
10
11 ( LINKS )  HOST  HEAD RAM 2+ 10 MOVE
12 ( VECTOR SEED )  ZERO <IRET>  PROM 4 - 2!
13 DECIMAL HERE 8 U.R {  bytes used } CR  FLUSH
14 EMPTY ' (CR) 'CR !
15
```

```
Screen:  424
 0 ( Slaves - Background Task Definition )
 1
 2 FORTH : BACKGROUND ( #S #R #U )   CREATE >R  OVER +
 3    DUP  THERE 1+  +  15 AND 15 XOR  DUP D+
 4    DUP  THERE 2+ + ,  SWAP THERE + ,  I C,  R> + 3 - ALLOT ;
 5
 6 : BUILD   OPERATOR @ OVER DUP @  SWAP 2+ 2+ C@ MOVE
 7    2@  DUP OPERATOR @ 2+ !  8 + ! ;
 8
 9 HEX : KILL ( taskname )   @  EA2E SWAP ! ; DECIMAL
10
11
12
13
14
15

Screen:  425
 0 143143
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  426
 0 143143
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  427
 0 144144
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  428
 0 144144
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  429
 0 144144
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 430
 0 145145
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 431
 0 145145
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 432
 0 145145
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen:  433
 0 146146
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  434
 0 146146
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  435
 0 ( TERMINAL)
 1 0 USER STATUS   8 USER S0   10 USER CTR   12 USER PTR
 2 14 USER 'CR   16 USER 'PAGE   18 USER 'TYPE                  CR
 3 20 USER DEVICE   24 USER 'EXPECT   26 USER H
 4 30 USER OFFSET   32 USER CNT   34 USER BASE   36 USER >IN    CR
 5 38 USER BLK   40 ( - 57) USER CONTEXT   58 USER CURRENT
 6 60 USER LAST   62 USER SCR   64 USER R#   95 USER ?KEY
 7 (  VARIABLE IOPB  8 ALLOT)
 8 VARIABLE GOLDEN  18 ALLOT
 9 (  VARIABLE 'BLOCK   VARIABLE 'BUFFER  2 ALLOT )
10 VARIABLE 'NUMBER   VARIABLE 'CREATE                          CR
11 ( VARIABLE Disk  2 ALLOT   VARIABLE PREV  18 ALLOT)
12  VARIABLE TEST  VARIABLE TODAY  VARIABLE TICKS  2 ALLOT
13 ( PROM 8 - CONSTANT NB   PROM 6 - CONSTANT FIRST )
14 PROM CONSTANT PROM
15 (   FORTH HEX 3000 'R ! DECIMAL HOST)
```

```
Screen:  436
 0    ( MULTI-PROGRAMMER)   HEX
 1 D7FF EQU WAKE
 2
 3 ASSEMBLER HERE   U POP   U DEC   U DEC   EA2E # STATUS U) MOV
 4    STATUS 6 + U) S MOV   I POP   R POP   NEXT
 5
 6 CODE PAUSE   WAKE # STATUS U) MOV   ( 194)
 7 LABEL WAIT   R PUSH   I PUSH   S STATUS 6 + U) MOV
 8    # W MOV   STATUS 2+ U) LIS
 9
10 CODE STOP   WAIT HERE 2 - !   ( 179)      DECIMAL
11
12
13
14
15

Screen:  437
 0    ( GET, RELEASE)
 1 | CODE GET   W POP   W ) U CMP   0= NOT IF   W ) 1 MOV
 2    1NZ IF   4 # I SUB   W PUSH   NEXT   ( 101+PAUSE)
 3       THEN   U W ) MOV   THEN   ' EXIT JMP
 4 : GET   PAUSE GET [
 5 CODE RELEASE   W POP   W ) U CMP   0= IF
 6    0 # W ) MOV   THEN   NEXT
 7 CODE GRAB   W POP   U W ) MOV   NEXT
 8
 9 CODE ACTIVATE   U 0 XCHG   W POP   W ) U MOV
10    WAKE # STATUS U) MOV   0 # CTR U) MOV
11    S0 U) W MOV   4 # W SUB   I W ) MOV   U 2 W) MOV
12    W STATUS 6 + U) MOV   U 0 XCHG   ' EXIT JMP
13
14
15

Screen:  438
 0 147147
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  439
 0 148148
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  440
 0 148148
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  441
 0 148148
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  442
 0 149149
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  443
 0 149149
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  444
 0 149149
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  445
 0 150150
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  446
 0 ( CONSTANTS FOR INTERPROCESSOR HARDWARE  COOR/COMM )
 1
 2 HEX
 3 F000 CONSTANT HSTAT     F008 CONSTANT SSTAT
 4
 5 F010 CONSTANT CB1       F011 CONSTANT CB2
 6 F012 CONSTANT CB3       F013 CONSTANT CB4
 7
 8 F015 CONSTANT STATOUT
 9
10 F016 CONSTANT INTS#     F017 CONSTANT INTDATA
11
12 DECIMAL
13
14
15

Screen:  447
 0 ( SLAVE INTERPRETER )  HEX
 1
 2 CODE RELOAD   EA C,  PROM 10 - ,  ZERO ,   NEXT  COMMAND
 3
 4 CODE ACTION   ( SSTAT 1+ 0 MOV B   01 #B 0 OR   0 STATOUT MOV B)
 5    CB3 0 MOV B   01 #B 0 TEST   0= IF
 6    0 POP   ELSE   02 #B 0 TEST   0= IF
 7    W POP   W ) LIP   THEN   THEN   NEXT
 8
 9 CODE SRESET   S0 U) S MOV   0 0 SUB   0 PUSH   NEXT
10
11 DECIMAL
12
13
14
15
```

```
Screen:  448
 0 ( SLAVE INTERPRETER )  HEX
 1
 2 CODE ?FIFO   CB4 0 MOV B   01 #B 0 TEST   0= IF   I DEC I DEC
 3   ( SSTAT 1+ 0 MOV B   0FE #B 0 AND   0 STATOUT MOV B )
 4    ' PAUSE JMP   THEN   NEXT
 5
 6 : MONITOR   BEGIN  ?FIFO  CB1 @  ACTION  ?STACK IF  SRESET  THEN
 7    (ELSE) [ (BACK)
 8
 9 : SLAVE  CLEAR  MONITOR ;
10
11 DECIMAL
12
13
14
15

Screen:  449
 0 151151
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  450
 0 151151
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  451
 0 152152
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  452
 0 152152
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  453
 0 152152
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  454
 0 153153
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  455
 0 153153
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  456
 0 153153
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  457
 0 154154
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  458
 0 154154
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  459
 0 154154
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  460
 0 155155
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  461
 0 155155
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  462
 0 ( Ion Path - Startup )
 1    : STARTUP   0 0 ITABLE 128 ERASE  0 VALUES 64 ERASE
 2    10000 0 1 ITABLE !   -1 1 1 ITABLE !
 3    10000 2 1 ITABLE !   -1 3 1 ITABLE !
 4    scanner BUILD   64663 ALIVE !
 5    0 YF/F C!   0 AMUF/F C! SLAVE ;
 6 (    : STARTUP  0 0 ITABLE 128 ERASE 0 VALUES 64 ERASE
 7    10000 0 1 ITABLE !  -1 1 1 ITABLE !
 8    10000 2 1 ITABLE !  -1 3 1 ITABLE !
 9    scanner BUILD  CR ." HI MIKEY " CR RESET ; )
10 HEX
11 : ICIO   10 0 DO   22 FC40 C!   5000 0 DO LOOP
12    27 FC40 C!   5000 0 DO LOOP   LOOP ;  COMMAND
13 DECIMAL
14
15
```

```
Screen:  463
 0 ( Reduction - Startup )
 1     : STARTUP   0 0  ITABLE  128 ERASE
 2    10000 0 1 ITABLE !   -1 1 1 ITABLE !
 3    10000 2 1 ITABLE !   -1 3 1 ITABLE !
 4      GINIT CLEAR GDEMO  64663 ALIVE ! GRAPHICS
 5    LIN  0 ?HEADER !  Title 16 BLANK
 6    0 XF/F C!  0 YF/F C! SLAVE ;
 7 EXIT
 8    : STARTUP  0 0 ITABLE 128 ERASE
 9   10000 0 1 ITABLE !   -1 1 1 ITABLE !
10   10000 2 1 ITABLE !   -1 3 1 ITABLE !
11   CR ." HI MIKEY " CR RESET ;
12
13
14
15

Screen:  464
 0 ( Detection - Start Up )
 1      : STARTUP   64663 ALIVE !  00 #points !
 2        TEST-ACQUIRE 2DROP  ( kluge for DAQ freak-out
 3                            on cold power-up )
 4            0 XF/F C!  SLAVE ;
 5 EXIT
 6     : STARTUP  CR ." HI MIKE " CR RESET ;
 7
 8
 9
10
11
12
13
14
15

Screen:  465
 0 ( Ion Path - Initial RAM )   HEX
 1
 2 | CREATE RAM   51 ,  0 ,  0 ,  0 ,  0 ,  0 ,  0 ,  0 ,  0 ,
 3    51 ,
 4
 5 | CREATE 'OPERATOR   EA2E ,  OPERATOR @ ,  ZERO ,  0 ,
 6    OPERATOR @ 80 - ,  0 ,  0 ,  ' (CR) ,  ' (PAGE) ,  ' (TYPE) ,
 7    FE00 , FE00 , ' (EXPECT) , THERE 180 + DUP , ,  0 ,  0 , 0A ,
 8
 9 FORTH : BIAS   0  ZERO 10 M*  D+  OPERATOR @ 0 D-
10    OPERATOR @ SWAP  10 M/ SWAP ;      DECIMAL
11
12 | CREATE INTERRUPTS
13    <EXPECT> BIAS , ,  <TYPE> BIAS , ,
14
15
```

```
Screen:  466
 0 ( Ion Path - Initialize I/O)   HEX
 1 | CREATE OWN
 2 ( USART0)  FE01 , 01 C, FE01 , 40 C, FE01 , 4E C, FE01 , 36 C,
 3   ( PIC0)  FE80 , 11 C, FE81 , 08 C, FE81 , 00 C, FE81 , 03 C,
 4            FE81 , 00 C,
 5   ( PPI0)  FC43 , 80 C, FC40 , 22 C, FC41 , 04 C,
 6
 7 | CODE PORTS
 8    1 POP   0 POP   I PUSH   I 0 XCHG   BEGIN
 9        LODS    W 0 XCHG   LODS B   0 W ) MOV B   LOOP
10    STI   I POP   NEXT
11
12 DECIMAL
13
14
15

Screen:  467
 0 ( Ion Path - Power Up )   HEX 28  PROM 4 -
 1 LABEL START ]   RAM GOLDEN 14 MOVE ['] (NUMBER) 'NUMBER !
 2  ['] (CREATE) 'CREATE !   IN-LINE 2@ 0 2!  0 4 60 MOVE
 3    INTERRUPTS IN-LINE 08 MOVE  'OPERATOR STATUS  DUP 60 ERASE
 4    24 MOVE  OWN 0C PORTS  TEST @ A55A - IF
 5  A55A TEST ! TODAY 6 ERASE  THEN  EMPTY
 6   STARTUP [
 7 ASSEMBLER   HERE
 8    0 ES LSG   0 SS LSG   0 DS LSG   START # I MOV
 9    OPERATOR @  DUP # U MOV   U R MOV   80 - # S MOV
10    STD B   NEXT
11
12 HERE  00F0  ORG   ZERO # 0 MOV   EA C,  00C0 ,  ZERO ,
13 00C0 ORG 9090 , 9090 , 9090 , 9090 , 9090 , 9090 , 9090 ,
14    9090 ,  EA C, SWAP , ZERO ,
15 ORG      DECIMAL

Screen:  468
 0 ( Reduction - Initial RAM )   HEX
 1 ( | VARIABLE TEST  6 ALLOT   ( A55A, TODAY, TICKS)
 2
 3 | CREATE RAM   51 ,  0 ,  0 ,  0 ,  0 ,  0 ,  0 ,  0 ,  0 ,
 4    51 ,
 5
 6
 7 | CREATE 'OPERATOR   EA2E ,  OPERATOR @ ,  ZERO ,  0 ,
 8    OPERATOR @ 80 - , 0 , 0 , ' (CR)  , ' PAGE , ' (TYPE)  ,
 9    FE00 , FE00 , ' (EXPECT) , THERE 180 + DUP , ,  0 ,  0 , 0A ,
10
11 FORTH : BIAS   0  ZERO 10 M*  D+  OPERATOR @ 0 D-
12    OPERATOR @ SWAP  10 M/ SWAP ;      DECIMAL
13
14 | CREATE INTERRUPTS
15    <EXPECT> BIAS , ,  <TYPE> BIAS , ,
```

```
Screen:  469
 0 ( Reduction - Initialize I/O)   HEX
 1 | CREATE OWN
 2 ( USART0)  FE01 , 01 C, FE01 , 40 C, FE01 , 4E C, FE01 , 36 C,
 3   ( PIC0)  FE80 , 11 C, FE81 , 08 C, FE81 , 80 C, FE81 , 03 C,
 4            FE81 , 00 C,
 5
 6 | CODE PORTS
 7    1 POP   0 POP   I PUSH   I 0 XCHG   BEGIN
 8        LODS    W 0 XCHG   LODS B   0 W ) MOV B   LOOP
 9    STI   I POP   NEXT
10 DECIMAL
11
12
13
14
15

Screen:  470
 0 ( Reduction - Power Up )   HEX  28   PROM 4 -
 1 LABEL START ]   RAM GOLDEN 14 MOVE ['] (NUMBER) 'NUMBER !
 2  ['] (CREATE) 'CREATE !   IN-LINE 2@ 0 2!  0 4 60 MOVE
 3    INTERRUPTS IN-LINE 08 MOVE  'OPERATOR STATUS  DUP 60 ERASE
 4    24 MOVE  OWN 09 PORTS  TEST @ A55A - IF
 5  A55A TEST ! TODAY 6 ERASE  THEN  EMPTY
 6     STARTUP  [
 7 ASSEMBLER   HERE
 8    0 ES LSG   0 SS LSG   0 DS LSG   START # I MOV
 9    OPERATOR @  DUP # U MOV   U R MOV   80 - # S MOV
10    STD B   NEXT
11
12 HERE  00F0  ORG   ZERO # 0 MOV   EA C,  00C0 ,  ZERO ,
13 00C0 ORG 9090 , 9090 , 9090 , 9090 , 9090 , 9090 , 9090 ,
14    9090 ,  EA C, SWAP , ZERO ,
15 ORG      DECIMAL

Screen:  471
 0 ( Detection - Initial RAM )   HEX
 1 ( | VARIABLE TEST  6 ALLOT   ( A55A, TODAY, TICKS)
 2
 3 | CREATE RAM   51 ,  0 ,  0 ,  0 ,  0 ,  0 ,  0 ,  0 ,  0 ,
 4    51 ,
 5
 6
 7 | CREATE 'OPERATOR   EA2E ,  OPERATOR @ ,  ZERO ,  0 ,
 8    OPERATOR @ 80 - ,  0 ,  0 ,  ' (CR) ,  ' (PAGE) ,  ' (TYPE) ,
 9    FE00 , FE00 , ' (EXPECT) , THERE 180 + DUP , ,  0 ,  0 , 0A ,
10
11 FORTH : BIAS   0  ZERO 10 M*  D+  OPERATOR @ 0 D-
12    OPERATOR @ SWAP  10 M/ SWAP ;      DECIMAL
13
14 | CREATE INTERRUPTS
15    <EXPECT> BIAS , ,  <TYPE> BIAS , ,
```

```
Screen:  472
 0 ( Detection - Initialize I/O )   HEX
 1 | CREATE OWN
 2 ( USART0)  FE01 , 01 C, FE01 , 40 C, FE01 , 4E C, FE01 , 36 C,
 3    ( PIO)  FA03 , 89 C, FA00 , 00 C, FA01 , 00 C,
 4   ( PIC0)  FE80 , 11 C, FE81 , 08 C, FE81 , 00 C, FE81 , 03 C,
 5            FE81 , 00 C,
 6
 7 | CODE PORTS
 8    1 POP   0 POP   I PUSH   I 0 XCHG   BEGIN
 9        LODS    W 0 XCHG   LODS B   0 W ) MOV B   LOOP
10    STI   I POP   NEXT
11 DECIMAL
12
13
14
15

Screen:  473
 0 ( Detection - Power Up )   HEX  28   PROM 4 -
 1 LABEL START ]   RAM GOLDEN 14 MOVE ['] (NUMBER) 'NUMBER !
 2  ['] (CREATE) 'CREATE !   IN-LINE 2@ 0 2!  0 4 60 MOVE
 3    INTERRUPTS IN-LINE 08 MOVE  'OPERATOR STATUS  DUP 60 ERASE
 4    24 MOVE  OWN 0C PORTS  TEST @ A55A - IF
 5  A55A TEST ! TODAY 6 ERASE  THEN  EMPTY
 6    STARTUP [
 7 ASSEMBLER   HERE
 8    0 ES LSG   0 SS LSG   0 DS LSG   START # I MOV
 9    OPERATOR @  DUP # U MOV   U R MOV   80 - # S MOV
10    STD B   NEXT
11
12 HERE  00F0  ORG   ZERO # 0 MOV   EA C,  00C0 ,  ZERO ,
13 00C0 ORG 9090 , 9090 , 9090 , 9090 , 9090 , 9090 , 9090 ,
14    9090 ,  EA C, SWAP , ZERO ,
15 ORG      DECIMAL

Screen:  474
 0 159159
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  475
 0 160160
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  476
 0 160160
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  477
 0 160160
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  478
 0 161161
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  479
 0 161161
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  480
 0 ( Graphics - GDC 7220 Graphics Load Block )
 1
 2 486 498  THRU   ( Basic Graphics )
 3 504 512  THRU   ( General Applications )
 4     516  LOAD   ( Graphics Demo )
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 481
 0
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162
 1
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162
 2
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162
 3
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162
 4
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162
 5
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162
 6
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162
 7
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162
 8
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162
 9
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162
10
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162
11
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162
12
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162162162162162162162162162
162162162162162162162162162162162162162162162162

```
13
16316316316316316316316316316316316316316316316316316316316316316316316316316316
16316316316316316316316316316316316316316316316316316316316316316316316316316316
16316316316316316316316316316316316316316316316316316316316316316316316316316316
14
16316316316316316316316316316316316316316316316316316316316316316316316316316316
16316316316316316316316316316316316316316316316316316316316316316316316316316316
16316316316316316316316316316316316316316316316316316316316316316316316316316316
15
16316316316316316316316316316316316316316316316316316316316316316316316316316316
16316316316316316316316316316316316316316316316316316316316316316316316316316316
16316316316316316316316316316316316316316316316316316316316316316316316316316316

Screen:  482
 0 163163
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  483
 0 163163
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  484
 0 164164
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  485
 0 164164
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  486
 0 ( Graphics - GDC 7220 Graphics )
 1 HEX ( 0019 VOCABULARY GRAPHICS     GRAPHICS DEFINITIONS )
 2 FF40 CONSTANT GDATA     FF40 CONSTANT GSTAT
 3 FF41 CONSTANT GCMD     DECIMAL
 4
 5 1024 CONSTANT XMAX     780 CONSTANT YMAX
 6
 7 VARIABLE XYLAST   2 ALLOT
 8
 9 CREATE gmode-pram  0 , YMAX 16 * , 0 , YMAX 16 * ,
10    ( graphics,interlaced,dynamic ram,refresh durng blank )
11 CREATE sync-format 31 C, XMAX 16 / 2- C, 166 C, 20 C, 5 C, 7 C,
12 FORTH   YMAX 2/ 9216 OR HOST ,
13
14 CREATE cursor-char 00 C, 192 C, 00 C,
15
```

```
Screen:  487
 0 ( Graphics - GDC 7220 I/O Routines )
 1
 2 ASSEMBLER CREATE GDCWAIT   BEGIN   GSTAT LDA B   2 #B 0 AND
 3    0= END   RET
 4
 5 CODE GWAIT   GDCWAIT CALL   NEXT
 6
 7 CODE PSEND ( add count ) 1 POP   0 POP   I PUSH   0 I XCHG
 8    BEGIN   GDCWAIT CALL   LODS B   GDATA STA B   LOOP
 9    I POP   NEXT
10
11 : W>GDC  'S 2 PSEND DROP ;
12 : B>GDC  'S 1 PSEND DROP ;
13
14
15

Screen:  488
 0 ( Graphics - GDC Commands )   HEX
 1 FORTH : cmnd  CREATE  C,  ;CODE   GDCWAIT CALL   W INC   W INC
 2    W ) 0 MOV B   GCMD STA B   NEXT
 3
 4 0E cmnd SYNC      6F cmnd VSYNC
 5 4B cmnd cchar     6B cmnd START     0C cmnd +BLANK
 6 0D cmnd -BLANK    46 cmnd zoom      49 cmnd curs
 7 47 cmnd pitch     22 cmnd ZEROS     21 cmnd COMPLEMENT
 8 23 cmnd ONES      4A cmnd mask      4C cmnd SFIG
 9 6C cmnd DFIG      68 cmnd GCHRD     A0 cmnd rdat
10
11 : MASK  mask W>GDC ;
12 : PITCH  pitch B>GDC ;
13
14 : PRAM  ( add count start ) 70 OR GWAIT  GCMD C! PSEND ;
15 DECIMAL

Screen:  489
 0 ( Graphics - GDC 7220 Functions )
 1 VARIABLE XYLAST   2 ALLOT
 2 VARIABLE (CSIZE)
 3 VARIABLE (ZOOM)
 4 VARIABLE (PATTERN)
 5
 6 : CSIZE  DUP (CSIZE) !  (ZOOM) @ 16 *  OR  zoom B>GDC ;
 7
 8 : PATTERN  'S 2 8 PRAM  (PATTERN) ! ;
 9
10 : GRESET  0 GCMD C! ;
11
12 : PAN ( y x )  16 / SWAP XMAX 16 / * + 'S 2 0 PRAM DROP ;
13
14
15
```

```
Screen:  490
 0 ( Graphics - GDC 7220 Line Drawing )     HEX
 1 CREATE dir 1 C, 0 C, 1 C, 2 C, 0 , 80 C, 80 C, 83 C, 0 C, 2 C,
 2    0 , 0 C, 83 C, 84 C, 87 C, 0 C, 6 C, 6 C, 0 , 87 C, 0 C,
 3    5 C, 0 C, 5 C, 0 , 0 C, 84 C, 0 C,
 4
 5 CODE LPARAMS  ( dy,dx - dy,dx,dir )   0 0 SUB   W POP   W ROL
 6    0 RCL   W SAR   0< IF   W NEG   THEN   2 POP   2 ROL   0 RCL
 7    2 SAR   0< IF   2 NEG   THEN  2 W CMP   0 RCL   F9 C,
 8    0= IF   F8 C,   THEN   0 RCL   0 ROL   W W OR   0= IF
 9    0 INC   THEN   2 2 OR   0= IF   0 INC   THEN   U PUSH
10    dir # U MOV   D7 C,   U POP   0 0 OR B   0< IF   W 2 XCHG
11    THEN   2 PUSH   W PUSH   7 # 0 AND   0 PUSH   NEXT
12
13
14 DECIMAL
15

Screen:  491
 0 ( Graphics - Line Drawing )
 1
 2 : (CUR)  ( y x )  2DUP XYLAST 2!  curs  16 /MOD ROT XMAX 16 /
 3    * +  W>GDC  16 * B>GDC ;
 4
 5 : (LINE)  ( y x ) SFIG  2DUP XYLAST 2@  2SWAP ROT - >R  SWAP -
 6    R> LPARAMS  8 OR B>GDC  2DUP W>GDC  2* OVER - W>GDC
 7    OVER SWAP - 2* W>GDC  2* W>GDC   DFIG 2DUP (CUR) XYLAST 2! ;
 8
 9 : (RECT) ( y x )  SFIG  64 B>GDC  3 W>GDC  SWAP 1- DUP W>GDC
10    SWAP 1- W>GDC  -1 W>GDC  W>GDC  DFIG ;
11
12 : (DOT) ( y x ) 2DUP (CUR) SFIG 0 DUP DUP 2DUP B>GDC W>GDC
13    W>GDC W>GDC W>GDC  DFIG (CUR) ;
14
15

Screen:  492
 0 ( Graphics - GDC 7220 Functions )
 1 ASSEMBLER CREATE READGDC   BEGIN   GSTAT LDA B
 2    0 SAR B   CS END   GCMD LDA B    RET
 3
 4 CODE (RDAT)   READGDC CALL   0 2 MOV B   READGDC CALL
 5    0 2 HI MOV B   16 # 1 MOV    BEGIN   2 RCR   0 RCL   LOOP
 6    0 PUSH   NEXT
 7
 8 : RDAT  SFIG 2 B>GDC 1 W>GDC  rdat (RDAT) ;
 9
10 : CLEAR   +BLANK  0 0 (CUR)  -1 MASK 4 0 DO  SFIG 2 B>GDC
11    16383 W>GDC  ZEROS  1 W>GDC LOOP  -BLANK  ONES ; COMMAND
12 : ON   +BLANK  0 0 (CUR)  -1 MASK 4 0 DO  SFIG 2 B>GDC
13    16383 W>GDC  ONES  1 W>GDC LOOP  -BLANK ; COMMAND
14
15 CREATE CHARACTER-SET
```

```
Screen:  493
 0        000 C, 000 C, 000 C, 000 C, 000 C, 000 C, 000 C, 000 C,
 1        000 C, 000 C, 000 C, 000 C, 079 C, 000 C, 000 C, 000 C,
 2        000 C, 000 C, 000 C, 007 C, 000 C, 007 C, 000 C, 000 C,
 3        000 C, 000 C, 020 C, 127 C, 020 C, 127 C, 020 C, 000 C,
 4        000 C, 000 C, 018 C, 042 C, 127 C, 042 C, 036 C, 000 C,
 5        000 C, 000 C, 099 C, 100 C, 008 C, 019 C, 099 C, 000 C,
 6        000 C, 000 C, 080 C, 038 C, 089 C, 078 C, 048 C, 000 C,
 7        000 C, 000 C, 000 C, 001 C, 002 C, 004 C, 000 C, 000 C,
 8        000 C, 000 C, 000 C, 065 C, 034 C, 028 C, 000 C, 000 C,
 9        000 C, 000 C, 000 C, 028 C, 034 C, 065 C, 000 C, 000 C,
10        000 C, 000 C, 020 C, 008 C, 062 C, 008 C, 020 C, 000 C,
11        000 C, 000 C, 008 C, 008 C, 062 C, 008 C, 008 C, 000 C,
12        000 C, 000 C, 000 C, 000 C, 000 C, 048 C, 064 C, 000 C,
13        000 C, 000 C, 008 C, 008 C, 008 C, 008 C, 008 C, 000 C,
14        000 C, 000 C, 000 C, 000 C, 000 C, 000 C, 064 C, 000 C,
15        000 C, 000 C, 003 C, 004 C, 008 C, 016 C, 096 C, 000 C,

Screen:  494
 0        000 C, 000 C, 062 C, 069 C, 073 C, 081 C, 062 C, 000 C,
 1        000 C, 000 C, 064 C, 064 C, 127 C, 066 C, 068 C, 000 C,
 2        000 C, 000 C, 070 C, 073 C, 081 C, 081 C, 098 C, 000 C,
 3        000 C, 000 C, 034 C, 085 C, 073 C, 065 C, 065 C, 000 C,
 4        000 C, 000 C, 016 C, 127 C, 018 C, 020 C, 024 C, 000 C,
 5        000 C, 000 C, 057 C, 069 C, 069 C, 069 C, 039 C, 000 C,
 6        000 C, 000 C, 049 C, 073 C, 073 C, 074 C, 060 C, 000 C,
 7        000 C, 000 C, 003 C, 005 C, 009 C, 113 C, 001 C, 000 C,
 8        000 C, 000 C, 054 C, 073 C, 073 C, 073 C, 054 C, 000 C,
 9        000 C, 000 C, 030 C, 041 C, 073 C, 073 C, 070 C, 000 C,
10        000 C, 000 C, 000 C, 000 C, 072 C, 000 C, 000 C, 000 C,
11        000 C, 000 C, 000 C, 000 C, 052 C, 000 C, 064 C, 000 C,
12        000 C, 000 C, 065 C, 065 C, 034 C, 020 C, 008 C, 000 C,
13        000 C, 000 C, 020 C, 020 C, 020 C, 020 C, 020 C, 000 C,
14        000 C, 000 C, 008 C, 020 C, 034 C, 065 C, 065 C, 000 C,
15        000 C, 000 C, 006 C, 009 C, 081 C, 001 C, 002 C, 000 C,

Screen:  495
 0        000 C, 000 C, 030 C, 085 C, 093 C, 065 C, 062 C, 000 C,
 1        000 C, 000 C, 126 C, 009 C, 009 C, 009 C, 126 C, 000 C,
 2        000 C, 000 C, 054 C, 073 C, 073 C, 073 C, 127 C, 000 C,
 3        000 C, 000 C, 034 C, 065 C, 065 C, 065 C, 062 C, 000 C,
 4        000 C, 000 C, 028 C, 034 C, 065 C, 065 C, 127 C, 000 C,
 5        000 C, 000 C, 065 C, 073 C, 073 C, 073 C, 127 C, 000 C,
 6        000 C, 000 C, 001 C, 009 C, 009 C, 009 C, 127 C, 000 C,
 7        000 C, 000 C, 122 C, 073 C, 065 C, 065 C, 062 C, 000 C,
 8        000 C, 000 C, 127 C, 008 C, 008 C, 008 C, 127 C, 000 C,
 9        000 C, 000 C, 000 C, 065 C, 127 C, 065 C, 000 C, 000 C,
10        000 C, 000 C, 001 C, 063 C, 065 C, 064 C, 032 C, 000 C,
11        000 C, 000 C, 065 C, 034 C, 020 C, 008 C, 127 C, 000 C,
12        000 C, 000 C, 064 C, 064 C, 064 C, 064 C, 127 C, 000 C,
13        000 C, 000 C, 127 C, 002 C, 012 C, 002 C, 127 C, 000 C,
14        000 C, 000 C, 127 C, 048 C, 008 C, 006 C, 127 C, 000 C,
15        000 C, 000 C, 062 C, 065 C, 065 C, 065 C, 062 C, 000 C,
```

```
Screen:  496
 0        000 C, 000 C, 006 C, 009 C, 009 C, 009 C, 127 C, 000 C,
 1        000 C, 000 C, 094 C, 033 C, 081 C, 065 C, 062 C, 000 C,
 2        000 C, 000 C, 070 C, 041 C, 025 C, 009 C, 127 C, 000 C,
 3        000 C, 000 C, 049 C, 073 C, 073 C, 073 C, 070 C, 000 C,
 4        000 C, 000 C, 001 C, 001 C, 127 C, 001 C, 001 C, 000 C,
 5        000 C, 000 C, 063 C, 064 C, 064 C, 064 C, 063 C, 000 C,
 6        000 C, 000 C, 007 C, 024 C, 096 C, 024 C, 007 C, 000 C,
 7        000 C, 000 C, 127 C, 032 C, 016 C, 032 C, 127 C, 000 C,
 8        000 C, 000 C, 099 C, 020 C, 008 C, 020 C, 099 C, 000 C,
 9        000 C, 000 C, 007 C, 008 C, 120 C, 008 C, 007 C, 000 C,
10        000 C, 000 C, 067 C, 069 C, 073 C, 081 C, 097 C, 000 C,
11        000 C, 000 C, 000 C, 000 C, 065 C, 065 C, 127 C, 000 C,
12        000 C, 000 C, 096 C, 016 C, 008 C, 004 C, 003 C, 000 C,
13        000 C, 000 C, 127 C, 065 C, 065 C, 000 C, 000 C, 000 C,
14        000 C, 000 C, 002 C, 001 C, 001 C, 001 C, 002 C, 000 C,
15        000 C, 000 C, 064 C, 064 C, 064 C, 064 C, 064 C, 000 C,

Screen:  497
 0 ( Graphics - Graphics Text )
 1 VARIABLE (GTDIR)
 2 : GCLOAD   32 - DUP 63 > IF 32 - THEN  63 AND 8 *
 3    CHARACTER-SET + 8 8 PRAM ;
 4 : GTDIR  7 AND 16 OR (GTDIR) ! ;
 5
 6 : (GTYPE) (PATTERN) @ PTR @ DUP CTR @ + SWAP DO SFIG (GTDIR) @
 7    B>GDC 7 W>GDC  I C@ GCLOAD GCHRD LOOP  0 CTR ! PATTERN ;
 8
 9 : (GCR) XYLAST 2@ DROP (CSIZE) @ 1+ 8 * + 0 (CUR) ;
10
11 : GRAPHICS ['] (GTYPE) 'TYPE ! ['] (GCR) 'CR ! ['] CLEAR
12    'PAGE ! ;
13 : TERMINAL  ['] (TYPE) 'TYPE ! ['] (CR) 'CR !
14    ['] (PAGE) 'PAGE ! ;
15

Screen:  498
 0 ( Graphics - GDC Initialization )
 1
 2 : GINIT
 3 GRESET     sync-format 8 PSEND    VSYNC
 4 cchar cursor-char 3 PSEND
 5 gmode-pram 8 0 PRAM   XMAX 16 / PITCH  ONES
 6 -1 PATTERN   -1 MASK
 7 0 (ZOOM) ! 0 CSIZE 0 GTDIR
 8 START
 9 GCMD 1+ C@ DROP ;     COMMAND
10
11
12
13
14
15
```

```
Screen:  499
 0 169169
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  500
 0
 1 ( Softknobs Interrupt Handler )
 2
 3 ASSEMBLER CREATE <KNOBS>   W PUSH   1 PUSH   0 PUSH
 4    ' KNOBVALUES # W MOV   4 # 1 MOV   INKNOBS LDA B
 5    BEGIN   0 SAR B   CS IF   W ) INC   THEN
 6            0 SAR B   CS IF   W ) DEC   THEN   W INC   W INC
 7    LOOP   0 POP   1 POP   W POP   IRET
 8
 9 <KNOBS> 14 INTERRUPT
10
11 DECIMAL
12
13
14
15

Screen:  501
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  502
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  503
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  504
 0 ( 32-BIT INPUT)
 1
 2 CODE WITHIN   2 POP   1 POP   0 POP   W W SUB
 3    2 0 CMP   0< IF   1 0 CMP   0< NOT IF   W INC
 4    THEN   THEN   W PUSH   NEXT
 5
 6 : (ASSIGN)   R> 2+ SWAP ! ;
 7    ' (ASSIGN) 2 - @
 8 FORTH : ASSIGN   \ (ASSIGN) IN-LINE , ;  TARGET
 9
10
11
12
13
14
15

Screen:  505
 0 ( Graphics - Field Controls )
 1
 2 VARIABLE FIELD#
 3
 4 5 12 MATRIX GPARAMS
 5
 6 FORTH : gparams   CREATE C,  DOES>  C@ FIELD# @ SWAP GPARAMS ;
 7
 8 0 gparams LPX     1 gparams UPX     2 gparams DPX
 9 3 gparams LPY     4 gparams UPY     5 gparams DPY
10 6 gparams LLX     7 gparams ULX     8 gparams DLX
11 9 gparams LLY    10 gparams ULY    11 gparams DLY
12
13 : FIELD ( n)   DUP  0 5 WITHIN IF  FIELD# !  ELSE  DROP  THEN ;
14    COMMAND
15
```

```
Screen:  506
 0 ( Graphics - Clipping, Scaling, and Windowing )
 1
 2 : CLIP ( y x - y x )   SWAP  LPY 2@ ROT MAX MIN  SWAP
 3    LPX 2@ ROT MAX MIN ;
 4
 5 : XTRANSLATE ( x )   LLX @ - DPX @ DLX @ */  LPX @ + ;
 6 : YTRANSLATE ( y )   LLY @ - DPY @ DLY @ */  UPY @ SWAP - ;
 7
 8 : XYTRANSLATE ( y x )   SWAP YTRANSLATE  SWAP XTRANSLATE ;
 9
10 : WINDOW ( y1 y2 x1 x2 )   2DUP SWAP - DPX !  SWAP LPX 2!
11    2DUP  SWAP - DPY !  SWAP LPY 2! ;   COMMAND
12
13 : SCLSET ( y1 y2 x1 x2 )   2DUP SWAP - DLX !  SWAP LLX 2!
14    2DUP  SWAP - DLY !  SWAP LLY 2! ;   COMMAND
15

Screen:  507
 0 ( Graphics - Cursor and Plotting Routines )
 1
 2 : CUR ( y x )   XYTRANSLATE  CLIP  (CUR) ;   COMMAND
 3
 4 VARIABLE GMODE
 5
 6 : (PLOT)   GMODE @ EXECUTE ;   COMMAND
 7
 8 : DOT   GMODE ASSIGN (DOT) ;   COMMAND
 9 : VECTOR   GMODE ASSIGN (LINE) ;   COMMAND
10 : HIST   GMODE ASSIGN  UPY @ OVER (CUR) (LINE) ;   COMMAND
11 : RECT   GMODE ASSIGN  LPY @ LPX @ (CUR)  (RECT) ;   COMMAND
12 : PLOT   XYTRANSLATE CLIP  (PLOT) ;   COMMAND
13
14
15

Screen:  508
 0 ( Graphics - Axes and Tics )
 1
 2 : AXES   GMODE @ VECTOR  UPY @ LPX @ (CUR)  LPY @ LPX @ (PLOT)
 3    LPY @ UPX @ (PLOT)  GMODE ! ;   COMMAND
 4
 5 : FRAME   GMODE @ RECT  UPY @ LPY @ -  UPX @ LPX @ -  (PLOT)
 6    GMODE ! ;  COMMAND
 7
 8 : XTICS ( start inc )   GMODE @ >R  VECTOR  ULX @ 1+ ROT
 9    DO  I XTRANSLATE DUP  UPY @ DUP  8 + ROT (CUR)  SWAP (PLOT)
10    DUP +LOOP  DROP  R> GMODE ! ;   COMMAND
11
12 : YTICS ( start inc )   GMODE @ >R  VECTOR  ULY @ 1+ ROT
13    DO  LPX @ DUP 5 -  I YTRANSLATE  DUP ROT (CUR)  SWAP (PLOT)
14    DUP +LOOP  DROP  R> GMODE ! ;   COMMAND
15
```

```
Screen:  509
 0 ( Graphics - Text Justification )
 1 VARIABLE GJMODE
 2
 3 : JUSTIFY   GJMODE @ EXECUTE ;
 4
 5 : GLEFT   GJMODE ASSIGN DROP ;
 6 : GRIGHT   GJMODE ASSIGN 8 (CSIZE) @ 1+ * * - ;
 7 : GCENTER   GJMODE ASSIGN 4 (CSIZE) @ 1+ * * - ;
 8
 9 : GTYPE   DUP XYLAST 2@  ROT JUSTIFY  (CUR)  TYPE ;
10
11 : G.   (.) GTYPE ;
12
13 : (G.")   1 R@ COUNT GTYPE ;
14 : G."   \  (G.")  STRING ;  IMMEDIATE
15

Screen:  510
 0 ( Graphics - Tic Tags )
 1
 2 VARIABLE GLMODE
 3
 4 : XTAGS   GCENTER  ULX @ 1+  ROT DO  UPY @ 15 +  I XTRANSLATE
 5    (CUR)  I GLMODE @ EXECUTE  GTYPE  DUP +LOOP  DROP ;   COMMAND
 6
 7 : YTAGS   GRIGHT  ULY @ 1+  ROT DO  I YTRANSLATE 5 -  LPX @ 10 -
 8    (CUR)  I GLMODE @ EXECUTE  GTYPE  DUP +LOOP  DROP ;   COMMAND
 9
10
11
12
13
14
15

Screen:  511
 0 ( Graphics - Unsigned Graphics Plotting )
 1
 2 : UXTRANS   LLX @ - DPX @ DLX @  U*/  LPX @ + ;
 3 : UYTRANS   LLY @ - DPY @ DLY @  U*/  LPY @ + ;
 4 : UXYTRANS   SWAP UYTRANS  SWAP UXTRANS ;
 5
 6 : UCUR   UXYTRANS (CUR) ;
 7 : UPLOT   UXYTRANS (PLOT) ;
 8
 9
10
11
12
13
14
15
```

```
Screen:  512
 0 ( Graphics - Unsigned Tics and Tags )
 1
 2 : UXTICS   ( start inc )  GMODE @ >R  VECTOR  ULX @ 1+ ROT
 3      DO  I UXTRANS DUP  UPY @ DUP 8 +   ROT  (CUR)  SWAP (PLOT)
 4         DUP /LOOP  DROP  R> GMODE ! ;
 5
 6 : UYTICS   ( start inc )  GMODE @ >R  VECTOR  ULY @ 1+  ROT
 7      DO  LPX @ DUP 5 -  I UYTRANS DUP  ROT  (CUR)  SWAP (PLOT)
 8         DUP /LOOP  DROP  R> GMODE  ! ;
 9
10 : UXTAG  GCENTER  ULX @ 1+ ROT  DO  UPY @ 25 +  I UXTRANS
11      (CUR)  I  GLMODE @ EXECUTE  GTYPE  DUP /LOOP  DROP ;
12 : UYTAG   GRIGHT  ULY @ 1+ ROT  DO  I UYTRANS 5 -  LPX @ 15 -
13       (CUR)  I  GLMODE @ EXECUTE  GTYPE  DUP /LOOP  DROP ;
14
15

Screen:  513
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  514
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  515
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  516
 0 ( Graphics - Graphics Test )
 1
 2 : 3TEST   GRAPHICS  ['] (.) GLMODE !
 3    3 0 DO  I FIELD  I 250 * 10 + DUP 200 + 100 950
 4    WINDOW  0 100 0 1000 SCLSET FRAME  I CSIZE
 5    0 25 YTICS  0 100 XTICS
 6    0 100 XTAGS  0 25 YTAGS  LOOP  TERMINAL ;
 7
 8 : PPP 0 0 CUR  100 0 DO I I 10 * PLOT  LOOP ;
 9
10 : GDEMO 3TEST 0 FIELD DOT PPP  1 FIELD VECTOR PPP
11    2 FIELD HIST PPP ;  COMMAND
12
13
14
15

Screen:  517
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  518
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  519
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  520
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  521
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  522
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  523
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 524
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 525
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 526
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen:  527
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  528
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  529
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  530
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  531
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  532
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  533
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  534
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  535
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 536
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 537
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 538
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen:  539
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  540
 0 ( 8087 Math Co-processor Support Option)
 1 ( : THRU 1+ SWAP DO I I . LOAD DECIMAL LOOP ; )
 2 ASSEMBLER DEFINITIONS
 3    ( 8087 Assembler)  543 546 THRU
 4 FORTH DEFINITIONS    DECIMAL    547 LOAD
 5 ( Operators)          548 549 THRU
 6 ( Constants)          550 LOAD
 7 ( Conditions)         551 LOAD  DECIMAL
 8 ( Input Conversion)   552 555 THRU   DECIMAL
 9 ( Output Conversion)  556 558 THRU   DECIMAL   2 FIX
10 ( Memory Access)      559 560 THRU   DECIMAL
11 ( Floating Literal)   561 562 THRU   DECIMAL   LONG
12 ( Utilities          563 564 THRU   DECIMAL   ( optional)
13 ( 8087 Interrupt)     565 LOAD       FINIT     ( 4354 bytes)
14 EXIT
15 CONTEXT GOLDEN 20 MOVE  HERE H 2+ !

Screen:  541
 0 HELP       Displays these 8087  MATH instructions.
 1
 2 n F.       Fixed point output of n fractional digits.
 3 s n F.R    Fixed point output right justified s spaces.
 4 n S.       Scientific output of n+1 digits.
 5 n E.       Engineering output of n+3 digits with exponent
 6              adjusted to powers of 3.
 7
 8 N.         Programmable format numeric output specified by:
 9            n FIX  or  n SCI  or  n ENG
10
11 ?N         Displays status of the 8087 coprocessor.
12
13 SHORT      Sets compiler to compile 32-bit real literals.
14 LONG       Sets compiler to compile 64-bit real literals.
15 INTEGER    Sets compilation of reals as 16/32-bit integers.
```

```
Screen:  542
 0 ( Real & integer number format examples)    EXIT
 1
 2 INTEGER numbers:       ( parameter stack)
 3
 4   37   -1000                 ( 16-bit)
 5
 6   43.  -1,000.     3:45      ( 32-bit: a blank must follow
 7   2,345   6/14/82   +9                   any period)
 8
 9 REAL numbers:         ( numeric stack)
10
11   .2   1.0   -2.3   +3:45.0    12.5E12    12.5e12    -12.5E-12
12
13   2.:  5.E78   +47.E+09   +1,000.e+300    2/00/0.5   +1:00.e-34
14
15   1000..   89.E+0    +19,999,999,999,999,999.9

Screen:  543
 0 ( 8087 Coprocessor assembler)   HEX
 1 00531 VOCABULARY forth    IMMEDIATE
 2  6 CONSTANT I16       2 CONSTANT I32
 3  0 CONSTANT R32       4 CONSTANT R64
 4 12 CONSTANT N)        CREATE 'WAIT  101 ,
 5
 6 : FWAIT   9B C, ;
 7 : NW   0 'WAIT C! ;
 8
 9 : AUTOWAIT   forth  'WAIT C@ IF  FWAIT  ELSE  'WAIT  1+ C@
10    'WAIT C! THEN ;
11
12 : INH ( n)   USER DOES> C@ AUTOWAIT D9 C, C, ;
13 : INH3 ( n)   USER DOES> C@ AUTOWAIT DB C, C, ;
14
15

Screen:  544
 0    ( Inherent opcodes)    HEX
 1   0FA INH FSQRT     0FD INH FSCALE     0F8 INH FPREM
 2   0FC INH FRNDINT   0F4 INH FXTRACT    0E1 INH FABS
 3   0E0 INH FCHS      0F2 INH FPTAN      0F3 INH FPATAN
 4   0F0 INH F2XM1     0F1 INH FYL2X      0F9 INH FYL2XP1
 5   0E4 INH FTST      0E5 INH FXAM       0EE INH FLDZ
 6   0E8 INH F1LD      0EB INH FLDPI      0E9 INH FLDL2T
 7   0EA INH FL2ELD    0EC INH FLG2LD     0ED INH FLN2LD
 8 ( 0F7 INH FINCSTP   0F6 INH FDECSTP    0D0 INH FNOP)
 9
10   0E3 INH3 FINIT    0E0 INH3 FENI      0E1 INH3 FDISI
11   0E2 INH3 FCLEX
12
13 : FCOMPP   AUTOWAIT D9DE , ;
14
15
```

```
Screen:  545
 0    ( Processor control class opcodes)    HEX
 1 : NPC ( n)  CONSTANT DOES>  @ AUTOWAIT INST! , R/M ;
 2
 3   28D9 NPC FCWLD       38D9 NPC FSTCW       38DD NPC FSWST
 4 ( 30D9 NPC FSTENV      20D9 NPC FLDENV      30DD NPC FSAVE)
 5 ( 20DD NPC FRSTOR      28DF NPC FILD)       28DB NPC FTLD
 6 ( 20DF NPC FBLD        38DF NPC FISTP)      38DB NPC FTSTP
 7 ( 30DF NPC FBSTP)
 8
 9 : stm ( n)   CONSTANT  DOES>  @ PTR @ +! ;
10
11   -2 stm -POP    -800 stm REV    -6 stm N0
12
13
14
15

Screen:  546
 0    ( Data transfer class opcodes)   HEX
 1 : DT   CONSTANT , DOES>  AUTOWAIT INST! SWAP DUP 12 =  forth
 2    IF  DROP @ , PTR @ 1+  ELSE  SWAP 2+ @ , R/M PTR @  THEN +! ;
 3
 4   00D9 C0D9 DT FLD    10D9 D0DD DT FST      18D9 D8DD DT FSTP
 5   10D8 D0D8 DT FCOM   18D8 D8D8 DT FCOMP    00D8 C0DE DT FADD
 6   28D8 E8DE DT FSUB   08D8 C8DE DT FMUL     38D8 F8DE DT FDIV
 7
 8 : sti   CONSTANT DOES> @ SWAP 100 * + AUTOWAIT , ;
 9
10        C8D9 sti FXCH    ( C0DD sti FFREE)
11
12
13
14
15

Screen:  547
 0 ( 8087 COMPATIBILITY BLOCK )
 1
 2 CODE cell   LODS   0 PUSH   NEXT   ( 63)
 3 CODE byte   LODS B   CWD B   0 PUSH   NEXT   ( 63)
 4
 5 CODE COMPILE   H U) W MOV   MOVS   W H U) MOV   NEXT
 6
 7 : [COMPILE]  ' 2-  , ;   IMMEDIATE
 8
 9 : DEPTH  'S 2+ S0 @ SWAP - 2/ ;
10
11 : ?DIGIT  1- DIGIT DUP IF  ROT 1+ ROT ROT  THEN ;
12
13 CODE >NUMERAL   0 POP   10 #B 0 CMP   CS NOT IF
14    7 #B 0 ADD   THEN   48 #B 0 ADD   0 PUSH   NEXT
15
```

```
Screen:  548
 0 ( Floating point arithmetic)
 1 CODE F+ ( )  ( r r - r)   1 N) FADD   NEXT
 2 CODE F- ( )  ( r r - r)   1 N) FSUB   NEXT
 3 CODE F* ( )  ( r r - r)   1 N) FMUL   NEXT
 4 CODE F/ ( )  ( r r - r)   1 N) FDIV   NEXT
 5 CODE FSQRT ( )  ( r - r)   FSQRT   NEXT
 6 CODE FABS ( )   ( r - r)   FABS    NEXT
 7 CODE FNEGATE ( )  ( r - r)    FCHS  NEXT
 8 CODE EXTRACT ( )  ( r - x s)  FXTRACT   NEXT
 9 CODE (TAN) ( )    ( r - y x)  FPTAN     NEXT
10 CODE (ARCTAN) ( ) ( y x - r)  FPATAN    NEXT
11 CODE 2**X-1 ( )   ( x - r)    F2XM1     NEXT
12 CODE Y*LOG2(X) ( ) ( y x - r)   FYL2X   NEXT
13 CODE Y*LOG2(X+1) ( ) ( y x - r)   FYL2XP1   NEXT
14 CODE >FIX ( ) ( r - r)   FRNDINT   NEXT
15 CODE 1/N ( ) ( x - r)   F1LD   1 N) FDIV REV   NEXT

Screen:  549
 0 ( Floating point stack operators)
 1 CODE FSWAP ( ) ( r r - r r)   1 FXCH   NEXT
 2 CODE FDUP  ( ) ( r - r r)   0 N) FLD   NEXT
 3 CODE F2DUP ( ) ( r r - r r r r)   1 N) FLD   1 N) FLD   NEXT
 4 CODE FOVER ( ) ( r r - r r r)   1 N) FLD   NEXT
 5 CODE FDROP ( r)   0 N) FSTP  NEXT
 6 CODE FROT  ( ) ( r r r - r r r)   1 FXCH   2 FXCH   NEXT
 7
 8 CODE >N ( n) ( - r)   S W MOV   I16 W ) FLD   0 POP   NEXT
 9 CODE N> ( - n) ( r)   0 PUSH   S W MOV   I16 W ) FSTP   FWAIT
10    NEXT
11 CODE 2>N ( d) ( - r)   2 POP   0 POP   2 PUSH   0 PUSH   S W MOV
12   I32 W ) FLD   0 POP   0 POP   NEXT
13 CODE 2N> ( - d) ( r)   0 PUSH   0 PUSH   S W MOV   I32 W ) FSTP
14   FWAIT   ' SWAP JMP
15

Screen:  550
 0 ( Floating point constants)
 1 CODE 0.0 ( ) ( r)   FLDZ   NEXT
 2 CODE 1.0 ( ) ( r)   F1LD   NEXT
 3 CODE PI  ( ) ( r)   FLDPI   NEXT
 4 CODE LN2 ( ) ( r)   FLDNL2   NEXT
 5
 6 CODE 2(E)LOG  ( ) ( r)   FL2ELD   NEXT
 7 CODE 2(10)LOG ( ) ( r)   FLDL2T   NEXT
 8 CODE 10(2)LOG ( ) ( r)   FLG2LD   NEXT
 9
10 : FINTEGER ( n)   CONSTANT   ;CODE   I16 2 W) FLD   NEXT
11
12 -1 FINTEGER -1.0    -2 FINTEGER -2.0     2 FINTEGER 2.0
13 10 FINTEGER 10.0
14
15
```

```
Screen:  551
 0 ( Floating point conditionals)   HEX
 1 HERE ASSEMBLER    FTST   0 N) FSTP
 2 HERE SWAP   0 PUSH   S W MOV   W ) FSWST   FWAIT   0 POP
 3    1 1 SUB   4100 # 0 AND   2 0 CMP   0= IF   1 INC   THEN
 4    1 PUSH   NEXT
 5
 6 CODE F0= ( - t) ( r)   4000 # 2 MOV   DUP JMP
 7 CODE F0< ( - t) ( r)   100 # 2 MOV   DUP JMP
 8 CODE F0> ( - t) ( r)   2 2 SUB   JMP
 9 CODE F= ( - t) ( r r)   FCOMPP   4000 # 2 MOV   DUP JMP
10 CODE F> ( - t) ( r r)   FCOMPP   100 # 2 MOV   DUP JMP
11 CODE F< ( - t) ( r r)   FCOMPP   2 2 SUB   JMP
12
13 : FMIN ( ) ( r r - r)   F2DUP F> IF FSWAP THEN FDROP ;
14 : FMAX ( ) ( r r - r)   F2DUP F< IF FSWAP THEN FDROP ;
15 : F?DUP ( - t) ( r - , r)   FDUP F0= DUP IF FDROP THEN 0= ;

Screen:  552
 0 ( Powers of 10 Table)
 1 CREATE POWERS  152 ALLOT
 2
 3 CODE [POWERS]   I16 ' 10.0 FLD   POWERS # W MOV   19 # 1 MOV
 4    F1LD   BEGIN   R64 W ) FST   1 N) FMUL N0   8 # W ADD
 5    LOOP   NEXT
 6
 7 [POWERS]  FORGET [POWERS]
 8
 9 CODE T10** ( +n) ( - r)   W POP   W SHL   W SHL   W SHL
10    POWERS # W ADD   R64  W ) FLD   NEXT
11
12
13
14
15

Screen:  553
 0 ( Computed power of 10)   HEX
 1 CODE (I+F) ( ) ( r - x s)   0 PUSH   S W MOV   NW W ) FSTCW
 2    W ) 0 MOV   F3FF # 0 AND   400 # 0 OR   W ) 0 XCHG   F1LD
 3    FCHS   1 N) FLD   W ) FCWLD   FRNDINT   0 W ) MOV   W ) FCWLD
 4    2 FXCH   0 POP   2 N) FSUB N0 REV   FSCALE   F2XM1
 5    1 N) FSUB REV   0 N) FMUL -POP   NEXT
 6
 7 CODE raise ( t) ( r x s - r)    1 FXCH   2 FXCH   0 POP
 8    0 0 OR   0= IF   1 N) FMUL   ELSE   1 N) FDIV REV   1 FXCH
 9    FCHS   1 FXCH   THEN   FSCALE   1 FXCH   0 N) FSTP   NEXT
10
11 : >10** ( n) ( r - r)   DUP 0<  SWAP  ABS DUP 13 < IF  T10**
12    EXTRACT  ELSE  >N 2(10)LOG F* (I+F)  THEN  raise ;
13
14
15
```

```
Screen:  554
 0 ( Real number conversion)  DECIMAL
 1 CODE ?REAL ( a - a t)   1 1 SUB   W POP   W PUSH   W ) 1 MOV B
 2    W INC   46 #B 0 MOV   REP B  SCAS B    1 PUSH    NEXT
 3
 4 CODE *REAL ( n) ( r - r)   I16  ' 10.0 FLD   1 N) FMUL
 5    S W MOV   I16  W ) FLD   0 POP   1 N) FADD    NEXT
 6
 7 : >REAL BEGIN  ?DIGIT  IF  *REAL 1 PTR +! AGAIN ;
 8
 9 : ?EXPONENT ( a)    DUP C@ DUP 32 - IF  DUP 69 = SWAP
10    101 = + 0= ABORT" ???"
11    (NUMBER)  (  PTR @ 0< NOT  IF DROP THEN)
12    >10**  ELSE 2DROP THEN ;
13
14
15

Screen:  555
 0 ( Floating point input conversion)   DECIMAL
 1 : (REAL) ( a) ( - r)   0 PTR !  1+ DUP C@ 45 = DUP >R +  0.0
 2   BEGIN  >REAL  DUP C@  DUP  43 48 WITHIN  SWAP 58 =  +
 3   IF 1+ 0 PTR ! AGAIN PTR @  T10** F/  ?EXPONENT
 4   R> IF  FNEGATE  THEN ;
 5
 6 : xnumber ( a - n, d) ( - r)  ?REAL IF  (REAL) 1  ELSE  (NUMBER)
 7    0   THEN  PTR ! ;
 8
 9   ' xnumber   'NUMBER !
10
11
12
13
14
15

Screen:  556
 0 ( Floating point fractional output )   HEX
 1 CODE F/DIGIT ( - n) ( r - q)    0 PUSH    S W MOV
 2    I16  ' 10.0 FLD   1 N) FLD   FPREM   2 FXCH   1 N) FDIV REV
 3    W ) FSTCW   W ) 0 MOV   0C00 # 0 OR   W ) 0 XCHG
 4    W ) FCWLD   FRNDINT   W ) 0 XCHG   W ) FCWLD   1 FXCH
 5    I16  W ) FSTP   FWAIT   NEXT
 6 : <#. ( ) ( r - r)   >FIX  <# ;
 7 : #. ( ) ( r - q)   F/DIGIT  >NUMERAL HOLD ;
 8 : #S. ( ) ( r - r)   BEGIN  #.  FDUP F0= END ;
 9 : .#> ( ) ( r)   FDROP  PTR @ PAD OVER - ;
10 : FSIGN ( n)   IF  2D ( -)  ELSE  20 ( bl)   THEN  HOLD ;
11 : (F.) ( n) ( r)   0 MAX 12 MIN  FDUP F0<  >R  FABS  DUP >10**
12    <#.  ?DUP IF  0 DO  #. LOOP  THEN  2E ( .) HOLD  #S.
13       R> FSIGN  .#> ;
14 : F. ( n) ( r)   (F.)  >TYPE SPACE ;
15 : F.R ( n n) ( r)   (F.)  ROT OVER - SPACES >TYPE ;
```

```
Screen:  557
 0 ( Floating point scientific output)   HEX
 1 : -MAGNITUDE ( - n) ( r - r)   EXTRACT FDROP FDUP  F0< FABS
 2    10(2)LOG  F*  0= IF  FNEGATE  THEN ;
 3
 4 : (ADJ) ( n - n') ( r - r)   FABS FDUP  10.0 F< NOT IF  FDROP
 5    10.0 F/ 1-  ELSE  1.0 F< IF  10.0 F* 1+  THEN  THEN ;
 6
 7 : XSCALE ( n - n)   FDUP F0= IF  0  ELSE  FDUP -MAGNITUDE N> DUP
 8    >10** FDUP (ADJ)  FDUP OVER >10** >FIX  OVER MINUS
 9    >10** (ADJ)  MINUS  THEN ;
10
11 : .EXP ( n)   45 EMIT  DUP ABS  0  <#  #S  PAD PTR @ -  1 =
12    IF  30 HOLD  THEN  ROT 0<  IF 2D ELSE 2B THEN  HOLD  #>
13    >TYPE SPACE ;
14
15 : S. ( n) ( r)   XSCALE SWAP (F.)   >TYPE .EXP ;

Screen:  558
 0 ( Floating point engineering output)
 1 : ADJUST ( n x - x n) ( r - r)   DUP ABS 3 MOD ?DUP IF  OVER
 2    0< IF  1- 2- MINUS  THEN  >R  I - SWAP I -  R>  >10**
 3    ELSE  SWAP  THEN ;
 4
 5 : E. ( n) ( r)   0 MAX 2+  XSCALE ADJUST  (F.)  >TYPE .EXP ;
 6
 7 ( Programmable format output)
 8 0 CONSTANT 'POINTS    VARIABLE 'FORMAT
 9
10 : N. ( ) ( r)   'POINTS  'FORMAT @ EXECUTE ;
11
12 : [N]   CREATE  ' ,  DOES> @ 'FORMAT !  ['] 'POINTS ! ;
13
14   [N] FIX F.    [N] SCI S.    [N] ENG E.
15

Screen:  559
 0 ( Floating point memory access)   HEX
 1 CODE short ( ) ( r)  R32  I ) FLD    4 # I ADD   NEXT
 2 CODE long  ( ) ( r)  R64  I ) FLD    8 # I ADD   NEXT
 3
 4 CODE s! ( a) ( r)   W POP   R32  W ) FSTP   NEXT
 5 CODE l! ( a) ( r)   W POP   R64  W ) FSTP   NEXT
 6
 7 CODE s@ ( a) ( - r)   W POP   R32  W ) FLD   NEXT
 8 CODE l@ ( a) ( - r)   W POP   R64  W ) FLD   NEXT
 9
10 : S, ( ) ( r)   HERE s!  4 ALLOT ;
11 : L, ( ) ( r)   HERE l!  8 ALLOT ;
12
13 : S+! ( a) ( r)   DUP s@ F+ s! ;
14 : L+! ( a) ( r)   DUP l@ F+ l! ;
15
```

```
Screen:  560
 0 ( Floating point constant & variable)
 1 : SVARIABLE   CREATE  0.0 S, ;
 2 : LVARIABLE   CREATE  0.0 L, ;
 3
 4 : SCONSTANT ( ) ( r)   CREATE  S,  ;CODE  R32 2 W) FLD   NEXT
 5 : LCONSTANT ( ) ( r)   CREATE  L,  ;CODE  R64 2 W) FLD   NEXT
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  561
 0 ( Floating point literal)
 1 VARIABLE 'FLITERAL
 2 : FLITERAL ( ) ( r)   'FLITERAL @ EXECUTE ;         IMMEDIATE
 3
 4 : SHORT   'FLITERAL ASSIGN  COMPILE short  S, ;    IMMEDIATE
 5 : LONG    'FLITERAL ASSIGN  COMPILE long   L, ;    IMMEDIATE
 6
 7 HERE ]  BEGIN -' IF  number  PTR @ IF  [COMPILE] FLITERAL  ELSE
 8    ?CELL  IF  COMPILE cell ,  ELSE  COMPILE byte C,
 9    THEN  THEN  ELSE  1+ @ 0<  IF  EXECUTE  DEPTH 0<
10    ABORT" Stack empty"  ELSE  2- , THEN  THEN  AGAIN [
11
12 HERE SWAP ASSEMBLER   R DEC  R DEC   I R ) MOV   # I MOV   NEXT
13 FORTH  (  ' ] 2- !)
14
15

Screen:  562
 0 ( Floating Integer literal)
 1 CODE integer ( ) ( - r)   I16  I ) FLD   I INC   I INC   NEXT
 2 CODE double ( ) ( - r)   I32  I ) FLD   4 # I ADD   NEXT
 3
 4 CODE i! ( a) ( r)   W POP   I16  W ) FSTP   NEXT
 5 CODE d! ( a) ( r)   W POP   I32  W ) FSTP   NEXT
 6
 7 CODE i@ ( a) ( - r)   W POP   I16  W ) FLD   NEXT
 8 CODE d@ ( a) ( - r)   W POP   I32  W ) FLD   NEXT
 9
10 : I, ( ) ( r)   HERE i!  2 ALLOT ;
11 : D, ( ) ( r)   HERE d!  4 ALLOT ;
12
13 : INTEGER ( ) ( r)   'FLITERAL ASSIGN  FDUP FABS  double
14    [ 32768.0 D, ]  F< IF  COMPILE integer I, ELSE COMPILE double
15    D,  THEN ;    IMMEDIATE
```

```
Screen:  563
 0 ( Numeric processor utilities)   HEX
 1 : BINARY   2 BASE ! ;
 2 CODE ?NDP ( - n)   0 PUSH   S W MOV   W ) FSWST   FWAIT   NEXT
 3
 4 : .INV   ." Invalid" ;          : .DENORM   ." Denormal" ;
 5 : .ZD   ." ZeroDivide" ;        : .OF   ." OverFlow" ;
 6 : .UF   ." UnderFlow" ;         : .PRC  ." Precision" ;
 7
 8 CREATE 'EXCEPTIONS  ] .INV .DENORM .ZD .OF .UF .PRC [
 9
10 : .EXCEPTIONS ( n - n)   CR  ." Exceptions: " 6 0  DO
11      DUP 1 AND IF  I 2* 'EXCEPTIONS + @ 2+ EXECUTE  SPACE THEN
12   2/ LOOP DROP ;
13 : .CC ( n - n)   DUP 100 /  7 AND  OVER  4000 AND  IF 8 + THEN
14 BASE @ >R  CR  ." Condition code: "  BINARY  0  <# # # # # #>
15 TYPE R> BASE ! ;

Screen:  564
 0 ( 8087 status display)   HEX
 1
 2 : .NSTACK  ( n - n)    DUP  800 /  7 AND  CR ." N-stack @ " . ;
 3
 4 : .IRQ  ( n)   80 AND  IF  CR ." Interrupt pending"  THEN  CR ;
 5
 6 : ?N   ?NDP .NSTACK  DUP .EXCEPTIONS  .CC .IRQ ;
 7
 8
 9
10
11
12
13
14
15

Screen:  565
 0 ( NDP Exception processing)   HEX
 1 CODE FINIT   NW FINIT  33E # 0 MOV   0 PUSH   S W MOV
 2    W ) FCWLD   FWAIT   0 POP   STI   NEXT
 3 FORTH
 4 HERE  ]  FINIT  1 ABORT" ? Numeric error"  [  >R
 5
 6 ASSEMBLER
 7 BEGIN   0 PUSH   S W MOV   NW W ) FSWST   0 POP   0 1 MOV
 8    80 # 0 AND   0= NOT IF   1 # 1 AND   0= NOT  IF
 9    ( Invalid)  R> # I MOV   NEXT    THEN
10    NW FCLEX   THEN   STI   IRET   0E INTERRUPT
11
12 DECIMAL
13 FORTH
14
15
```

```
Screen:  566
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  567
 0 ( Math option: Logarithmic functions)
 1 : 2** ( ) ( r - r)   FDUP F0< FABS  1.0 FSWAP (I+F) raise ;
 2
 3 : Y**X ( ) ( y x - r)   FSWAP Y*LOG2(X) 2** ;   ( y to the x)
 4
 5 : EXP ( ) ( x - r)   2(E)LOG F* 2** ;           ( e to the x)
 6
 7 : 10** ( ) ( x - r)   2(10)LOG F* 2** ;         ( 10 to the x)
 8
 9 ( Natural log of x)
10 CODE LN ( ) ( x - r)   FLN2LD   1 FXCH   FYL2X   NEXT
11
12 ( Log base 10 of x)
13 CODE LOG ( ) ( x - r)   FLG2LD   1 FXCH   FYL2X   NEXT
14
15

Screen:  568
 0 ( Math option: Temperature conversion)   INTEGER
 1
 2 9.0 5.0 F/ SCONSTANT TSCALE
 3
 4 : F>C ( ) ( f - c)   32.0 F- TSCALE F/ ;
 5
 6 : C>F ( ) ( c - f)   TSCALE F* 32.0 F+ ;
 7
 8 LONG
 9
10
11
12
13
14
15
```

```
Screen:  569
 0 ( Math option: Percent)
 1        HERE 100 ,
 2 CODE % ( ) ( x % - x r)  1 N) FMUL N0   I16  SWAP ( 100) FLD
 3    1 N) FDIV   NEXT
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  570
 0 ( Math option: Matrix defining words)
 1 : SMATRIX ( #r #c)   CREATE  2* 2* DUP , * ALLOT  ;CODE   W INC
 2    W INC   1 POP   0 POP   W ) MUL B   W INC   W INC   W 0 ADD
 3    1 SHL   1 SHL   1 0 ADD   0 PUSH   NEXT
 4
 5 : LMATRIX ( #r #c)   CREATE  8 * DUP , * ALLOT  ;CODE   W INC
 6    W INC   1 POP   0 POP   W ) MUL B   W INC   W INC   W 0 ADD
 7    1 SHL   1 SHL   1 SHL   1 0 ADD   0 PUSH   NEXT
 8
 9 ( Short matix display)
10 : SMD ( #r #c)   SWAP ' 2+ SWAP  0 DO  CR OVER
11    0 DO  DUP S@ N. 2+ 2+ LOOP  LOOP  CR 2DROP ;
12
13  ( Long matrix display)
14 : LMD ( #r #c)   SWAP ' 2+ SWAP  0 DO  CR OVER
15    0 DO  DUP L@ N. 8 + LOOP  LOOP  CR 2DROP ;

Screen:  571
 0 ( Math example: Determinant and Inverse of a 2x2 matrix)
 1 ( 123 LOAD)
 2
 3   2 2 SMATRIX DATA     SHORT     5 FIX
 4
 5 : DET ( )  ( - r)   1 1 DATA S@  0 0 DATA S@  F*  0 1 DATA S@
 6    1 0 DATA S@ F* F- ;
 7
 8 : INV   DET  1 1 DATA DUP S@  0 0 DATA DUP S@  SWAP
 9    S! S!  1 0 DATA DUP S@ FNEGATE  0 1 DATA DUP S@ FNEGATE
10    S! S!  4 0 DO  0 I DATA DUP S@  FOVER F/ S!  LOOP  FDROP ;
11
12 : !DATA   5.123  0 0 DATA S!
13           3.47   0 1 DATA S!
14          -1.8    1 0 DATA S!
15            PI    1 1 DATA S!  ;    !DATA
```

```
Screen:  572
 0 ( STUFF )
 1
 2 : ROOT  ( x root of y    Y X )
 3   1/N  FSWAP LOG  F* 10.0 FSWAP Y**X ;
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  573
 0 ( FACTORIAL )
 1
 2 : X! 1+  1 >N  1 DO I  >N F* LOOP  N. ;
 3
 4  6 SCI
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  574
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  575
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  576
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  577
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  578
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  579
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  580
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  581
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  582
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  583
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 584
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 585
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 586
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen:  587
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  588
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  589
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  590
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  591
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  592
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  593
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  594
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  595
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  596
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  597
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  598
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  599
 0 ( Start-up Test for Slave Viability - MJK 8/20/85 ) HEX
 1 1LABEL ALIVE 1ALIVE
 2 2LABEL ALIVE 2ALIVE
 3 3LABEL ALIVE 3ALIVE
 4
 5 :CASE ALIVE  1ALIVE  2ALIVE  3ALIVE ;
 6
 7 : ?ALIVE CR  3 0 DO I DUP 1+ DUP ." Slave " . #SLAVE  ALIVE I@
 8      FC97 =  IF ." is alive. "  ELSE ." is dead. " THEN
 9      CR LOOP ;
10
11 ?ALIVE
12
13 DECIMAL
14
15

Screen:  600
 0 ( Master - Mass Spec Applications Load Block )
 1  ' <CREATE> 'CREATE !
 2   CR 599 LOAD ( Slave Viability )
 3   28 LOAD     ( Slave Utilities for Master )
 4  606 LOAD     ( Basic MS/MS Control )
 5  732 LOAD     ( Fast Scans and Splits )
 6  540 LOAD     ( Mathemagics )
 7  840 LOAD     ( Data Operations )
 8  960 LOAD     ( Miscellaneous Functions )
 9 1080 LOAD     ( Scans and Sweeps )
10 1200 LOAD      ( MRM and SIM )
11 1260 LOAD      ( Calibration Software )
12 ' ?CREATE 'CREATE !
13
14 CONTEXT GOLDEN 20 MOVE  HERE H 2+ !
15 CR FREE {  bytes free }  BELL

Screen:  601
 0 ( Ion Path - Mass Spec Applications Load Block )
 1
 2  612  625 THRU ( Device mnemonics, addresses, and control )
 3  684      LOAD ( SIM/MRM Variable Definitions )
 4  708  714 THRU ( Scanning Support Words )
 5  720  725 THRU ( Fast Scans )
 6  744  746 THRU ( Splits )
 7  768      LOAD ( Rate setting )
 8 1086 1089 THRU ( Sweeps )
 9 1140 1146 THRU ( Scans )
10 1206 1212 THRU ( SIM/MRM )
11 1266      LOAD ( Calibration scans )
12 VARIABLE ALIVE  COMMAND
13 : STAT-TEST  15 STATOUT C! ; COMMAND
14
15
```

```
Screen:  602
 0 ( Reduction - Mass Spec Applications Load Block )
 1
 2  630  639 THRU     ( Device Access and Interpolations )
 3  774      LOAD     ( Peak Finding Parameters )
 4  912  932 THRU     ( Mass Spec Graphics )
 5 1098 1107 THRU     ( Sweeps )
 6 1149 1157 THRU     ( Scans )
 7 1218 1221 THRU     ( SIM/MRM )
 8 1272      LOAD     ( Calibration )
 9 VARIABLE ALIVE  COMMAND
10
11 : STAT-TEST  15 STATOUT C! ; COMMAND
12 : HELLO  0 0 (CUR) ." HI MIKEY " ; COMMAND
13 : STROBE  RSTROBE C@ DROP ; COMMAND
14
15

Screen:  603
 0 ( Detection - Mass Spec Applications Load Block )
 1
 2  780  781  THRU ( Data Acquisition Addresses )
 3 1116 1120  THRU
 4 1273 LOAD         ( Calibration Scanning )
 5 VARIABLE ALIVE  COMMAND
 6 EXIT
 7     2706 LOAD   ( LINK AND SYNC )
 8     2707 LOAD   ( CONSTANTS )
 9     2708 2709 THRU   ( ACQUIRE )
10     2710 LOAD   ( CONVERSIONS )
11
12
13
14
15

Screen:  604
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  605
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  606
 0 ( Master - Load Block for MS/MS Control )
 1
 2 CR { Loading Device Control Routines }
 3
 4 607 LOAD     ( Screen Operations Vectors )
 5 642 656 THRU ( Device Control )
 6 660 666 THRU ( Parameter Editor )
 7 672 678 THRU ( Softknobs )
 8 690 700 THRU ( SIMED and RED )
 9
10
11
12
13
14
15

Screen:  607
 0 ( Master - Screen Operations Vectors )
 1
 2 VARIABLE 'SCRNEXIT
 3 VARIABLE 'PED
 4 VARIABLE 'SPLITS
 5 VARIABLE 'KNOBS
 6
 7 : PED   'PED @ EXECUTE ;
 8 : SPLITS   'SPLITS @ EXECUTE ;
 9 : KNOBS   'KNOBS @ EXECUTE ;
10 : SCRNEXIT   'SCRNEXIT @ EXECUTE ;
11
12 : >NULL   'SCRNEXIT ASSIGN NULL ;
13 : >PED   'SCRNEXIT ASSIGN R> DROP PED ;
14 : >SPLITS   'SCRNEXIT ASSIGN R> DROP SPLITS ;
15 : >KNOBS   'SCRNEXIT ASSIGN R> DROP KNOBS ;
```

```
Screen:  608
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  609
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  610
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  611
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  612
 0 ( Ion Path - Data Space Definitions )
 1 32 ARRAY VALUES  COMMAND
 2 32 ARRAY SCRATCH
 3
 4 VARIABLE SWEEPDEV
 5 VARIABLE #DEVICE  COMMAND
 6 CODE !DEVICE# ( n)   0 POP   #DEVICE STA   NEXT
 7
 8 CODE VALUE   #DEVICE W MOV   W SHL   ' VALUES # W ADD   W PUSH
 9    NEXT
10 CODE scratch  #DEVICE W MOV   W SHL   ' SCRATCH # W ADD  W PUSH
11    NEXT
12 VARIABLE STEP
13 VARIABLE FQ3  2 ALLOT ( fractional Q3 value )
14 VARIABLE FRAC 2 ALLOT ( fractional step for Q3 )
15

Screen:  613
 0 ( Ion Path - Device Name Definitions )
 1
 2 FORTH : device   CREATE ,  ;CODE   W INC   W INC   W ) 0 MOV
 3    #DEVICE STA   NEXT
 4
 5  0 device EV      1 device REP     2 device CIV
 6  3 device EIV     4 device EXT     5 device L1
 7  6 device L2      7 device Q1      8 device L3
 8  9 device Q2     10 device L4     11 device Q3
 9 12 device MHV
10                  16 device M1     17 device M2
11 18 device M3     19 device DM1    20 device DM3
12 21 device RS1    22 device RS3
13 24 device P2
14
15
```

```
Screen:  614
 0 ( Ion Path - Device Address Table )   HEX
 1 CREATE device-address
 2  ( 00 EV )  FC00 ,  ( 01 REP)  FC02 ,  ( 02 CIV)  FC04 ,
 3  ( 03 EIV)  FC06 ,  ( 04 EXT)  FC08 ,  ( 05 L1 )  FC0A ,
 4  ( 06 L2 )  FC0C ,  ( 07 Q1 )  FC82 ,  ( 08 L3 )  FC9C ,
 5  ( 09 Q2 )  FC8A ,  ( 10 L4 )  FC9E ,  ( 11 Q3 )  FC92 ,
 6  ( 12 MHV)  FC0E ,  ( 13 NUL)    -1 ,  ( 14 NUL)    -1 ,
 7  ( 15 NUL)    -1 ,  ( 16 M1 )  FC80 ,  ( 17 M2 )  FC88 ,
 8  ( 18 M3 )  FC90 ,  ( 19 DM1)  FC84 ,  ( 20 DM3)  FC94 ,
 9  ( 21 RS1)  FC86 ,  ( 22 RS3)  FC96 ,  ( 23 NUL)    -1 ,
10  ( 24 P2 )  FC8E ,  ( 25 NUL)    -1 ,  ( 26 NUL)    -1 ,
11  ( 27 NUL)    -1 ,  ( 28 NUL)    -1 ,  ( 29 NUL)    -1 ,
12  ( 30 NUL)    -1 ,  ( 31 NUL)    -1 ,
13 DECIMAL
14 CODE DEVICE-ADDRESS ( - a)   #DEVICE W MOV   W SHL
15    ' device-address W) 0 MOV   0 PUSH   NEXT

Screen:  615
 0 ( Ion Path - Interpolation )
 1
 2 CODE INTERPOLATE ( n from to - n )
 3    2 POP   1 POP   0 POP   I PUSH   2 I MOV   1 W MOV
 4    ( W=from, I=to, 0=value )
 5    -2 # 2 MOV   BEGIN   2 INC   2 INC   SCAS   CS END
 6    2 DEC   2 DEC   ( 2=index)   1 W MOV   2 W ADD   2 I ADD
 7    2 I) 1 MOV   I ) 1 SUB   ( d-c)
 8    W ) 0 SUB   ( x-a)   1 MUL   ( x-a * d-c)
 9    2 W) 1 MOV   W ) 1 SUB   ( b-a)   1 DIV
10    1 SHR   2 1 SUB   CS IF   0 INC   THEN   ( round off )
11    I ) 0 ADD   I POP   0 PUSH   NEXT
12
13 ( round-off has been aded for symmetry of operation )
14
15

Screen:  616
 0 ( Ion Path - Interpolation Functions for Ior Quad Masses )
 1
 2  4 16 MATRIX ITABLE COMMAND
 3 ( COLUMN   0=Q1MASS   1=Q1DAC    2=Q2MASS )
 4 (          3=Q2DAC    4=Q3MASS   5=Q3DAC  )
 5
 6 : >Q1DAC   0 0 ITABLE  1 0 ITABLE  INTERPOLATE ;
 7 : >Q1MASS  1 0 ITABLE  0 0 ITABLE  INTERPOLATE ;
 8 : >Q3DAC   2 0 ITABLE  3 0 ITABLE  INTERPOLATE ;
 9 : >Q3MASS  3 0 ITABLE  2 0 ITABLE  INTERPOLATE ;
10 : >Q2DAC   >Q1DAC ;
11 : >Q2MASS  >Q1MASS ;
12
13
14
15
```

```
Screen:  617
 0 ( Ion Path - DAC Conversions )
 1
 2 : RND   >R OVER < IF R> 2DROP 1+ ELSE R> < IF 1- THEN THEN ;
 3 : >VDAC ( n - n)   2000 MIN -2000 MAX MINUS 2047 2000 N*/
 4    1000 -1000 RND ;
 5 : >VOLTS ( n - n)   2000 2047 N*/ 1023 -1023 RND MINUS ;
 6 : >q34dac ( n - n)   2000 MIN -2000 MAX MINUS  2047 1000 N*/
 7    1000 -1000 RND ;
 8 : >Q&L34 ( n - n)   1000 2047 N*/  1023 -1023 RND MINUS ;
 9 : >HVAMP ( n - n) 2000 MIN -2000 MAX MINUS 2047 2000 N*/
10      1000 -1000 RND ;
11 : HVAMP> ( n - n) 2000 2047 N*/ 1023 -1023 RND MINUS ;
12
13
14
15

Screen:  618
 0    ( Ion Path - DAC Conversions, continued )
 1
 2 : >EVDAC ( n - n)   1000 MIN 0 MAX 2047 1000 N*/ 1000 -1 RND ;
 3 : >EV    ( n - n)   1000 2047 N*/  1023 -1 RND ;
 4
 5 : >MHVDAC ( u - n)   0 UMAX 3500 UMIN  2047 5000 */MOD
 6    2500 ROT U< IF  1+  THEN ;
 7 : >MHV ( u - n)   5000 2083 */MON SWAP 1041 > IF 1+ THEN 10 * ;
 8
 9 : >q13dac ( n - n)   1000 MIN -1000 MAX MINUS 2047 3000 N*/
10    500 -500 RND ;
11 : >Q13 ( n - n)   3000 2047 N*/ 1023 -1023 RND MINUS ;
12
13
14
15

Screen:  619
 0 ( Ion Path - Output Drivers )
 1
 2 CODE NULL   NEXT
 3
 4 : SETTLE   250 0 DO  LOOP ;       ( quad settling time )
 5
 6 : !OUTPUT ( n ) DEVICE-ADDRESS ! ;
 7
 8 : !MASS ( n)  DEVICE-ADDRESS  !   SETTLE ;
 9
10
11 EXIT
12 CODE !OUTPUT ( n ) #DEVICE W MOV    W SHL   ' device-address
13   W) 0 MOV   0 W MOV   0 POP   0 W ) MOV B   0 HI 1 W) MOV B
14   NEXT
15
```

```
Screen:  620
 0 ( Ion Path - Variable Device Control Table )
 1
 2 CREATE VDCT
 3 (  0 EV  )  ' >EVDAC  ,  ' >EV     ,  ' !OUTPUT ,
 4 (  1 REP )  ' >VDAC   ,  ' >VOLTS  ,  ' !OUTPUT ,
 5 (  2 CIV )  ' >VDAC   ,  ' >VOLTS  ,  ' !OUTPUT ,
 6 (  3 EIV )  ' >VDAC   ,  ' >VOLTS  ,  ' !OUTPUT ,
 7 (  3 EXT )  ' >VDAC   ,  ' >VOLTS  ,  ' !OUTPUT ,
 8 (  3 L1  )  ' >VDAC   ,  ' >VOLTS  ,  ' !OUTPUT ,
 9 (  3 L2  )  ' >VDAC   ,  ' >VOLTS  ,  ' !OUTPUT ,
10 (  3 Q1  )  ' >q13dac ,  ' >Q13    ,  ' !OUTPUT ,
11 (  3 L3  )  ' >HVAMP  ,  ' HVAMP>  ,  ' !OUTPUT  ,
12 (  3 Q2  )  ' >q34dac ,  ' >Q&L34  ,  ' !OUTPUT ,
13 (  3 L4  )  ' >HVAMP  ,  ' HVAMP>  ,  ' !OUTPUT  ,
14
15

Screen:  621
 0    ( Ion Path - Variable Device Control Table, continued )
 1
 2 ( 11 Q3  )  ' >q13dac ,  ' >Q13    ,  ' !OUTPUT ,
 3 ( 12 MHV )  ' >MHVDAC ,  ' >MHV    ,  ' !OUTPUT ,
 4 ( 13 NUL )  ' NULL    ,  ' NULL    ,  ' DROP    ,
 5 ( 14 NUL )  ' NULL    ,  ' NULL    ,  ' DROP    ,
 6 ( 15 NUL )  ' NULL    ,  ' NULL    ,  ' DROP    ,
 7 ( 16 M1  )  ' >Q1DAC  ,  ' >Q1MASS ,  ' !MASS   ,
 8 ( 17 M2  )  ' >Q2DAC  ,  ' >Q2MASS ,  ' !MASS   ,
 9 ( 18 M3  )  ' >Q3CAC  ,  ' >Q3MASS ,  ' !MASS   ,
10 ( 19 DM1 )  ' >q13dac ,  ' >Q13    ,  ' !OUTPUT  ,
11 ( 20 DM3 )  ' >q13dac ,  ' >Q13    ,  ' !OUTPUT  ,
12 ( 21 RS1 )  ' >q13dac ,  ' >Q13    ,  ' !OUTPUT  ,
13 ( 22 RS3 )  ' >q13dac ,  ' >Q13    ,  ' !OUTPUT  ,
14 ( 23 NUL )  ' NULL    ,  ' NULL    ,  ' DROP    ,
15

Screen:  622
 0    ( Ion Path - Variable Device Control Table, continued )
 1
 2 ( 24 P2  )  ' >VDAC   ,  ' >VOLTS  ,  ' !OUTPUT ,
 3 ( 25 NUL )  ' NULL    ,  ' NULL    ,  ' DROP    ,
 4 ( 26 NUL )  ' NULL    ,  ' NULL    ,  ' DROP    ,
 5 ( 27 NUL )  ' NULL    ,  ' NULL    ,  ' DROP    ,
 6 ( 28 NUL )  ' NULL    ,  ' NULL    ,  ' DROP    ,
 7 ( 29 NUL )  ' NULL    ,  ' NULL    ,  ' DROP    ,
 8 ( 30 NUL )  ' NULL    ,  ' NULL    ,  ' DROP    ,
 9 ( 31 NUL )  ' NULL    ,  ' NULL    ,  ' DROP    ,
10
11
12
13
14
15
```

```
Screen:  623
 0 ( Ion Path - Variable Device Control Table Access )
 1
 2 FORTH : vdct   CREATE  2* VDCT + ,  DOES>  @ #DEVICE @   6 * +
 3    @ EXECUTE ;
 4
 5 0 vdct >DAC
 6 1 vdct >UNITS
 7 2 vdct !DATA
 8
 9
10 : ?INSTALLED ( - f)   DEVICE-ADDRESS  -1 = NOT ;
11
12
13
14
15

Screen:  624
 0 ( Ion Path - Device Access and Control )
 1
 2 : !SCRATCH ( n)   scratch ! ;
 3 : @SCRATCH ( - n)   scratch @ ;
 4 : >SCRATCH   VALUE @ !SCRATCH ;
 5
 6 : SET ( value device# )   !DEVICE#  DUP  VALUE !
 7    >DAC DUP !DATA  !SCRATCH ; COMMAND
 8
 9 : RESTORE   VALUE @ >DAC DUP    !DATA    !SCRATCH ; COMMAND
10
11 : DINIT   #DEVICE @  32 0 DO  I !DEVICE#  ?INSTALLED
12    IF    RESTORE   THEN   LOOP  !DEVICE# ;   COMMAND
13
14 : SCRATCHOUT   #DEVICE @  32 0 DO  I !DEVICE#  ?INSTALLED
15    IF  @SCRATCH !DATA  THEN  LOOP  !DEVICE# ;

Screen:  625
 0 ( Ion Path - Special Address Definitions )
 1 HEX
 2
 3 FC80 CONSTANT M1DAC
 4 FC88 CONSTANT M2DAC
 5 FC8C CONSTANT HDAC   ( scope x-axis dac )
 6 FC90 CONSTANT M3DAC
 7
 8 10 2* ' SCRATCH +  CONSTANT M1SCRATCH
 9 12 2* ' SCRATCH +  CONSTANT M3SCRATCH
10
11 FC40 CONSTANT EI/CI
12 FC41 CONSTANT SCOPE-MUX
13
14 DECIMAL
15
```

Screen: 626
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 627
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 628
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen:  629
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  630
 0 ( Reduction - Data Space Definitions )
 1
 2 VARIABLE #DEVICE COMMAND
 3
 4 CODE !DEVICE# ( n)   0 POP   #DEVICE STA   NEXT
 5
 6 CODE NULL   NEXT
 7
 8
 9
10
11
12
13
14
15

Screen:  631
 0 ( Reduction - Device Name Definitions )
 1
 2 FORTH : device   CREATE ,  ;CODE   W INC   W INC   W ) 0 MOV
 3    #DEVICE STA   NEXT
 4
 5  0 device EV      1 device REP     2 device CIV
 6  3 device EIV     4 device EXT     5 device L1
 7  6 device L2      7 device Q1      8 device L3
 8  9 device Q2     10 device L4     11 device Q3
 9 12 device MHV
10                  16 device M1     17 device M2
11 18 device M3     19 device DM1    20 device DM3
12 21 device RS1    22 device RS3
13 24 device P2
14
15
```

```
Screen:  632
 0 ( Reduction - Interpolation )
 1
 2 CODE INTERPOLATE ( n from to - n )
 3    2 POP   1 POP   0 POP   I PUSH   2 I MOV   1 W MOV
 4    ( W=from, I=to, 0=value )
 5    -2 # 2 MOV   BEGIN   2 INC   2 INC   SCAS   CS END
 6    2 DEC   2 DEC   ( 2=index)   1 W MOV   2 W ADD   2 I ADD
 7    2 I) 1 MOV   I ) 1 SUB   ( d-c)
 8    W ) 0 SUB   ( x-a)   1 MUL   ( x-a * d-c)
 9    2 W) 1 MOV   W ) 1 SUB   ( b-a)   1 DIV
10    1 SHR   2 1 SUB   CS IF   0 INC   THEN   ( round off )
11    I ) 0 ADD   I POP   0 PUSH   NEXT
12
13 ( round-off has been aded for symmetry of operation )
14
15

Screen:  633
 0 ( Reduction - Interpolation Functions for Ior Quad Masses )
 1
 2 4 16 MATRIX ITABLE  COMMAND
 3 ( COLUMN   0=Q1MASS   1=Q1DAC    2=Q2MASS )
 4 (          3=Q2DAC    4=Q3MASS   5=Q3DAC  )
 5
 6 : >Q1DAC   0 0 ITABLE  1 0 ITABLE  INTERPOLATE ;
 7 : >Q1MASS  1 0 ITABLE  0 0 ITABLE  INTERPOLATE ;
 8 : >Q3DAC   2 0 ITABLE  3 0 ITABLE  INTERPOLATE ;
 9 : >Q3MASS  3 0 ITABLE  2 0 ITABLE  INTERPOLATE ;
10 : >Q2DAC  >Q1DAC ;
11 : >Q2MASS  >Q1MASS ;
12
13
14
15

Screen:  634
 0 ( Reduction - DAC Conversions )
 1
 2 : RND   >R OVER < IF R> 2DROP 1+ ELSE R> < IF 1- THEN THEN ;
 3 : >VDAC ( n - n)   2000 MIN -2000 MAX MINUS 2047 2000 N*/
 4    1000 -1000 RND ;
 5 : >VOLTS ( n - n)   2000 2047 N*/ 1023 -1023 RND MINUS ;
 6 : >q34dac ( n - n)   2000 MIN -2000 MAX MINUS  2047 1000 N*/
 7    1000 -1000 RND ;
 8 : >Q&L34 ( n - n)   1000 2047 N*/  1023 -1023 RND MINUS ;
 9 : >HVAMP ( n - n ) 2000 MIN -2000 MAX MINUS 2047 2000 N*/
10    1000 -1000 RND ;
11 : HVAMP> ( n - n ) 2000 2047 N*/ 1023 -1023 RND MINUS ;
12   EXIT : >Q1DAC   10000 MIN 0 MAX 131 20 */ ;
13 : >Q2DAC   >Q1DAC ;  : >Q3DAC  >Q1DAC ;
14 : >Q1MASS  20 131 */ ;
15 : >Q2MASS  >Q1MASS ;  : >Q3MASS >Q1MASS ;
```

```
Screen:  635
 0    ( Reduction - DAC Conversions, continued )
 1
 2 : >EVDAC ( n - n)   1000 MIN 0 MAX 2047 1000 N*/ 1000 -1 RND ;
 3 : >EV    ( n - n)   1000 2047 N*/  1023 -1 RND ;
 4
 5 : >MHVDAC ( u - n)   0 UMAX 3500 UMIN  2083 5000 */MOD
 6    2500 ROT U< IF  1+  THEN ;
 7 : >MHV ( u - n)   5000 2083 */MON SWAP 1041 > IF 1+ THEN 10 * ;
 8
 9 : >q13dac ( n - n)   1000 MIN -1000 MAX MINUS 2047 3000 N*/
10    500 -500 RND ;
11 : >Q13 ( n - n)   3000 2047 N*/ 1023 -1023 RND MINUS ;
12
13
14
15

Screen:  636
 0 ( Reduction - Variable Device Control Table )
 1
 2 CREATE VDCT
 3 (  0 EV  )  ' >EVDAC  ,  ' >EV     ,
 4 (  1 REP )  ' >VDAC   ,  ' >VOLTS  ,
 5 (  2 CIV )  ' >VDAC   ,  ' >VOLTS  ,
 6 (  3 EIV )  ' >VDAC   ,  ' >VOLTS  ,
 7 (  4 EXT )  ' >VDAC   ,  ' >VOLTS  ,
 8 (  5 L1  )  ' >VDAC   ,  ' >VOLTS  ,
 9 (  6 L2  )  ' >VDAC   ,  ' >VOLTS  ,
10 (  7 Q1  )  ' >q13dac ,  ' >Q13    ,
11 (  8 L3  )  ' >HVAMP ,  ' HVAMP>  ,
12 (  9 Q2  )  ' >q34dac ,  ' >Q&L34  ,
13 ( 10 L4  )  ' >HVAMP ,  ' HVAMP>  ,
14
15

Screen:  637
 0    ( Reduction - Variable Device Control Table, continued )
 1
 2 ( 11 Q3  )  ' >q13dac ,  ' >Q13    ,
 3 ( 12 MHV )  ' >MHVDAC ,  ' >MHV    ,
 4 ( 13 NUL )  ' NULL    ,  ' NULL    ,
 5 ( 14 NUL )  ' NULL    ,  ' NULL    ,
 6 ( 15 NUL )  ' NULL    ,  ' NULL    ,
 7 ( 16 M1  )  ' >Q1DAC  ,  ' >Q1MASS ,
 8 ( 17 M2  )  ' >Q2DAC  ,  ' >Q2MASS ,
 9 ( 18 M3  )  ' >Q3DAC  ,  ' >Q3MASS ,
10 ( 19 DM1 )  ' >q13dac ,  ' >Q13    ,
11 ( 20 DM3 )  ' >q13dac ,  ' >Q13    ,
12 ( 21 RS1 )  ' >q13dac ,  ' >Q13    ,
13 ( 22 RS3 )  ' >q13dac ,  ' >Q13    ,
14 ( 23 NUL )  ' NULL    ,  ' NULL    ,
15
```

```
Screen:  638
 0    ( Reduction - Variable Device Control Table, continued )
 1
 2 ( 24 P2  )  ' >VDAC   ,  ' >VOLTS  ,
 3 ( 25 NUL )  ' NULL    ,  ' NULL    ,
 4 ( 26 NUL )  ' NULL    ,  ' NULL    ,
 5 ( 27 NUL )  ' NULL    ,  ' NULL    ,
 6 ( 28 NUL )  ' NULL    ,  ' NULL    ,
 7 ( 29 NUL )  ' NULL    ,  ' NULL    ,
 8 ( 30 NUL )  ' NULL    ,  ' NULL    ,
 9 ( 31 NUL )  ' NULL    ,  ' NULL    ,
10
11
12
13
14
15

Screen:  639
 0 ( Reduction - Variable Device Control Table Access )
 1
 2 FORTH : vdct   CREATE  2* VDCT + ,  DOES>  @ #DEVICE @ 4 * +
 3    @ EXECUTE ;
 4
 5 0 vdct >DAC
 6 1 vdct >UNITS
 7
 8
 9
10
11
12
13
14
15

Screen:  640
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  641
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  642
 0 ( Master - Variable Device Parameter Table )
 1
 2 4 32 MATRIX VDPT  768 ALLOT
 3 VARIABLE #DEVICE     VARIABLE IMODE     VARIABLE #PSET
 4
 5 : vdpt   CREATE  C,  DOES>  C@ #DEVICE @ VDPT
 6    IMODE @ 256 * + ;
 7
 8 0 vdpt 'CURRENT
 9 1 vdpt 'START
10 2 vdpt 'END
11 3 vdpt 'STEP
12
13 : !DEVICE# ( n)   #DEVICE ! ;
14
15 :CASE params   'CURRENT  'START  'END  'STEP ;

Screen:  643
 0 ( Master - Device Address Table )
 1
 2 CREATE device-address
 3    ( EV  )  0 ,  ( REP )  1 ,  ( CIV )  2 ,  ( EIV )  3 ,
 4    ( EXT )  4 ,  ( L1  )  5 ,  ( L2  )  6 ,  ( Q1  )  7 ,
 5    ( L3  )  8 ,  ( Q2  )  9 ,  ( L4  ) 10 ,  ( Q3  ) 11 ,
 6    ( MHV ) 12 ,  ( NUL ) -1 ,  ( NUL ) -1 ,  ( NUL ) -1 ,
 7    ( M1  ) 16 ,  ( M2  ) 17 ,  ( M3  ) 18 ,  ( DM1 ) 19 ,
 8    ( DM3 ) 20 ,  ( RS1 ) 21 ,  ( RS3 ) 22 ,  ( NUL ) -1 ,
 9    ( P2  ) -1 ,  ( NUL ) -1 ,  ( TIM ) -1 ,  ( NUL ) -1 ,
10    ( NUL ) -1 ,  ( NUL ) -1 ,  ( NUL ) -1 ,  ( NUL ) -1 ,
11
12 : DEVICE-ADDRESS ( - n)   #DEVICE @ 2* device-address + @ ;
13
14 : ?INSTALLED ( - f)   DEVICE-ADDRESS  -1 = NOT ;
15
```

```
Screen:  644
 0 ( Master - Data Output Routines )
 1
 2  : IONOUT (  value)    ?INSTALLED IF  1PUSH  DEVICE-ADDRESS
 3    1PUSH  SL1 SET  ELSE  DROP  THEN ;
 4
 5  1LABEL #DEVICE SL-DEVICE
 6  1LABEL VALUES VALUES
 7
 8 EXIT
 9 : IONOUT ( value) 1 #SLAVE  ?INSTALLED
10     IF  VALUES  #DEVICE @ DUP SL-DEVICE I! 2* + I!
11         SL1 RESTORE  ELSE  DROP  THEN  ;
12
13
14
15

Screen:  645
 0 ( Master - LABELS for Devices and Modes )
 1
 2 : LABELS ( line# blk# size)   CREATE  1+ C,  ,  64 * ,  DOES>
 3    DUP C@  ROT OVER *  ROT 1+  2@ BLOCK  +  + SWAP 1- ;
 4
 5  2 7934 3 LABELS DNAME
 6  6 7934 6 LABELS MNAME
 7 12 7934 3 LABELS PMNAME
 8
 9 : .DNAME ( n)   DNAME >TYPE ;
10 : .MNAME ( n)   MNAME >TYPE ;
11 : .PMNAME ( n)   PMNAME >TYPE ;
12
13
14
15

Screen:  646
 0 ( Master - Numeric Formatting )
 1
 2 : (N.) ( n - a cnt)   DUP ABS  0 <# # 46 HOLD #S SIGN #> ;
 3 : (U.) ( u - a cnt)           0 <# # 46 HOLD #S #> ;
 4
 5 : N.1 ( n)   (N.)  6 OVER - SPACES  TYPE ;
 6 : U.1 ( u)   (U.)  6 OVER - SPACES  TYPE ;
 7
 8 : N.1R ( n)   (N.)  RIGHT ;
 9 : U.1R ( u)   (U.)  RIGHT ;
10
11 : N7 ( n)   (.)  6 OVER - SPACES  TYPE ;
12
13 : N7R ( n)   (.)  RIGHT ;
14
15
```

```
Screen:  647
 0 ( Master - Limit Checking )
 1
 2 : BOUNDS   1+ 2CONSTANT  DOES>  2DUP  2@ WITHIN IF  DROP  ELSE
 3    2@ 1- ROT  MIN MAX  1 .WARN  WLEVEL @ IF  ." - "
 4    #DEVICE @ .DNAME  SPACE  THEN  THEN ;
 5
 6    0.0  1000.0 BOUNDS MRANGE  ( mass range )
 7 -200.0   200.0 BOUNDS VRANGE  ( voltage range )
 8    0    3500   BOUNDS HRANGE  ( high voltage range )
 9    0.0   100.0 BOUNDS ERANGE  ( eV range )
10 -100.0    35.0 BOUNDS CRANGE  ( civ and rep range )
11 -100.0   100.0 BOUNDS QRANGE  ( quad range )
12
13
14
15

Screen:  648
 0 ( Master - Variable Device Control Table )
 1
 2 CREATE VDCT
 3
 4 (  0 EV  )  ' ERANGE , ' N.1 , ' N.1R , ' IONOUT ,
 5 (  1 REP )  ' CRANGE , ' N.1 , ' N.1R , ' IONOUT ,
 6 (  2 CIV )  ' CRANGE , ' N.1 , ' N.1R , ' IONOUT ,
 7 (  3 EIV )  ' VRANGE , ' N.1 , ' N.1R , ' IONOUT ,
 8 (  4 EXT )  ' VRANGE , ' N.1 , ' N.1R , ' IONOUT ,
 9 (  5 L1  )  ' VRANGE , ' N.1 , ' N.1R , ' IONOUT ,
10 (  6 L2  )  ' VRANGE , ' N.1 , ' N.1R , ' IONOUT ,
11 (  7 Q1  )  ' QRANGE , ' N.1 , ' N.1R , ' IONOUT ,
12 (  8 L3  )  ' QRANGE , ' N.1 , ' N.1R , ' IONOUT ,
13 (  9 Q2  )  ' QRANGE , ' N.1 , ' N.1R , ' IONOUT ,
14 ( 10 L4  )  ' QRANGE , ' N.1 , ' N.1R , ' IONOUT ,
15 ( 11 Q3  )  ' QRANGE , ' N.1 , ' N.1R , ' IONOUT ,

Screen:  649
 0    ( Master - Variable Device Control Table, continued)
 1
 2 ( 12 MHV )  ' HRANGE , ' N7  , ' N7R  , ' IONOUT ,
 3 ( 13 NUL )  ' VRANGE , ' N.1 , ' N.1R , ' DROP   ,
 4 ( 14 NUL )  ' VRANGE , ' N.1 , ' N.1R , ' DROP   ,
 5 ( 15 NUL )  ' VRANGE , ' N.1 , ' N.1R , ' DROP   ,
 6 ( 16 M1  )  ' MRANGE , ' N.1 , ' N.1R , ' IONOUT ,
 7 ( 17 M2  )  ' MRANGE , ' N.1 , ' N.1R , ' IONOUT ,
 8 ( 18 M3  )  ' MRANGE , ' N.1 , ' N.1R , ' IONOUT ,
 9 ( 19 DM1 )  ' QRANGE , ' N.1 , ' N.1R , ' IONOUT ,
10 ( 20 DM3 )  ' QRANGE , ' N.1 , ' N.1R , ' IONOUT ,
11 ( 21 RS1 )  ' QRANGE , ' N.1 , ' N.1R , ' IONOUT ,
12 ( 22 RS3 )  ' QRANGE , ' N.1 , ' N.1R , ' IONOUT ,
13 ( 23 NUL )  ' VRANGE , ' N.1 , ' N.1R , ' DROP   ,
14 ( 24 P2  )  ' VRANGE , ' N.1 , ' N.1R , ' DROP   ,
15 ( 25 NUL )  ' VRANGE , ' N.1 , ' N.1R , ' DROP   ,
```

```
Screen:  650
 0    ( Master - Variable Device Control Table, continued )
 1
 2 ( 26 TIM )  ' VRANGE , ' U.1 , ' U.1R , ' DROP   ,
 3 ( 27 NUL )  ' VRANGE , ' N.1 , ' N.1R , ' DROP   ,
 4 ( 28 NUL )  ' VRANGE , ' N.1 , ' N.1R , ' DROP   ,
 5 ( 29 NUL )  ' VRANGE , ' N.1 , ' N.1R , ' DROP   ,
 6 ( 30 NUL )  ' VRANGE , ' N.1 , ' N.1R , ' DROP   ,
 7 ( 31 NUL )  ' VRANGE , ' N.1 , ' N.1R , ' DROP   ,
 8
 9
10
11
12
13
14
15

Screen:  651
 0 ( Master - Variable Device Control Table Access )
 1
 2 : vdct   CREATE  2* VDCT + ,  DOES>  @ #DEVICE @ 8 * +
 3    @ EXECUTE ;
 4
 5 0 vdct ?RANGE
 6 1 vdct .NUM
 7 2 vdct NUM.
 8 3 vdct !DATA
 9
10
11
12
13
14
15

Screen:  652
 0 ( Master - Device Definitions )
 1
 2 : device   CREATE  ,  DOES>  @ !DEVICE# ;
 3
 4  0 device EV      1 device REP     2 device CIV
 5  3 device EIV     4 device EXT     5 device L1
 6  6 device L2      7 device Q1      8 device L3
 7  9 device Q2     10 device L4     11 device Q3
 8 12 device MHV
 9                  16 device M1     17 device M2
10 18 device M3     19 device DM1    20 device DM3
11 21 device RS1    22 device RS3
12 24 device P2                      26 device TIM
13
14 -1 device NUL
15
```

```
Screen:  653
 0 ( Master - Setting Device Parameters )
 1
 2 : SET    ( n)   ?RANGE  DUP  'CURRENT !   !DATA    ;
 3 : !START ( n)   ?RANGE  'START ! ;
 4 : !END   ( n)   ?RANGE  'END   ! ;
 5 : !STEP  ( n)   ?RANGE  'STEP  ! ;
 6
 7 : DATAOUT   'CURRENT @ !DATA ;
 8
 9 : ?VALUES   #DEVICE @  32 0 DO  I !DEVICE#  ?INSTALLED IF
10    4 0 DO  I params DUP @  ?RANGE  SWAP !  LOOP  THEN  LOOP
11    !DEVICE# ;
12
13  : DINIT  1 #SLAVE  0 0 VDPT IMODE @ 256 * + 64 0 DO  DUP I +
14     @ VALUES I + I! 2 +LOOP DROP SL1 DINIT ;
15

Screen:  654
 0 ( Master - Parameter Set Mode Control )
 1 HEX : EI/CI ( n)   1 #SLAVE FC40 IC! ; DECIMAL
 2
 3 : imodes  CREATE  C, C,  DOES> DUP C@ IMODE !  1+ C@ EI/CI
 4    ?VALUES  DINIT ;
 5
 6 ( ICIO cntrl , imode # )    34 0 imodes EI
 7                             37 1 imodes +CI
 8                              5 2 imodes -CI
 9                             38 3 imodes USR
10
11 :CASE IMODES   EI  +CI  -CI  USR ;   ( assign mode by number )
12  0 0 VDPT 1024 ERASE   EI     0 #PSET !
13
14 : IONMODE   IMODE @ DUP IMODES SCUR  0 76 CURSOR .PMNAME RCUR ;
15

Screen:  655
 0 ( Master - Special Set Up Words )
 1
 2 : MS1 ( n)   DUP  2/ DUP M3 SET  M2 SET  M1 SET ;
 3
 4 : MS3 ( n)   DUP  2/ DUP M1 SET  M2 SET  M3 SET ;
 5
 6 : LOSS ( n)   M1 'START @  SWAP - M3 !START ;
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  656
 0 ( Feedback Modes for Quad Controller Testing) HEX
 1
 2 FC98 CONSTANT  Q1DIGITAL
 3 FC99 CONSTANT  Q2DIGITAL
 4 FC9A CONSTANT  Q3DIGITAL
 5
 6 : 1FEEDBACK   1 #SLAVE 81 Q1DIGITAL IC! ;
 7 : 2FEEDBACK   1 #SLAVE 81 Q2DIGITAL IC! ;
 8 : 3FEEDBACK   1 #SLAVE 81 Q3DIGITAL IC! ;
 9
10 ( Supposedly, in feedback mode, 10 volts minus the mass command
11   voltage will appear at the mux output - giving the user
12   confidence that the controller is responding appropriately to
13   the mass command )
14
15 DECIMAL

Screen:  657
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  658
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  659
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  660
 0 ( Master - Parameter Set Title Operations )
 1
 2 [R PARAMETER SET TITLES\ ]
 3
 4 : PDIR ( display parameter block titles )
 5    IN-LINE REPROT  16 0 DO  2 SPACES  I 2 U.R SPACE
 6    7935 BLOCK  I 64 * +  64 -TRAILING >TYPE  CR  LOOP ;
 7
 8 : 'PTITLE ( - a)   #PSET @ 64 *  7935 BLOCK + ;
 9
10 : .PTITLE   ." #" #PSET @ 2 U.R SPACE  'PTITLE 64 >TYPE
11    ." MODE: "  IMODE @ .PMNAME  +CR ;
12
13
14
15

Screen:  661
 0 ( Master - Parameter Set Save and Get Commands )
 1
 2 : PGET ( n)   DUP  0 16 WITHIN IF  DUP #PSET !  8051 + BLOCK
 3    0 0 VDPT 1024 MOVE  DINIT  ELSE  DROP  2 .ERROR  THEN ;
 4
 5 : PSAVE ( n)   DUP  0 16 WITHIN IF  DUP #PSET !  8051 + BLOCK
 6    0 0 VDPT SWAP 1024 MOVE  UPDATE FLUSH  ELSE  DROP
 7    2 .ERROR  THEN ;
 8
 9  0 PGET
10
11 : TUNESAVE ( n)   DUP  0 16 WITHIN IF  8051 + BLOCK  0 0 VDPT
12    IMODE @ 256 *  DUP D+ SWAP  32 <CMOVE  UPDATE FLUSH
13    ELSE  DROP 2 .ERROR  THEN ;
14
15 : ALLTUNE   16 0 DO  I TUNESAVE  LOOP ;
```

```
Screen:  662
 0 ( Master - STAT Display )
 1
 2     [R                                \ # DEV CURRENT  START    EN
 3 D   STEP        # DEV CURRENT  START    END   STEP ] DUP 2+ 2+
 4
 5 : STAT-TITLE   LITERAL [ , ] ASSIGN  +L .PTITLE
 6    RPT @ HEADING ;   ( type STAT title )
 7
 8 : STAT STAT-TITLE  LITERAL [ , ] REPORT   #DEVICE @   16 0 DO
 9    17 0 DO   J I + DUP DUP  !DEVICE# (.) RIGHT  ?INSTALLED IF
10    DNAME RIGHT  4 0 DO  I params @ NUM.  LOOP
11    ELSE  DROP  5 SKIPS  THEN
12    16 +LOOP +L  LOOP  !DEVICE#  CR ;
13    ( type STAT / VDP Table )
14
15 : .PED   STAT ;

Screen:  663
 0 ( Master - Parameter Editor  Cursor Positioning )
 1 VARIABLE #COL     VARIABLE PED-BUFFER
 2 : +COL   #COL @ +  4 MOD  #COL ! ;
 3 : +DEVICE   #DEVICE @ +  31 AND  !DEVICE# ;   ( next device )
 4 : +MODE   IMODE @ 1+  3 AND  IMODE ! ;
 5 : PUTCUR   7 #COL @ * 8 +  #DEVICE @ 16 /MOD 42 * ROT +
 6    SWAP   5 + SWAP CURSOR ;
 7
 8 : DMOVE   BEGIN   1 +DEVICE  ?INSTALLED  END ;
 9 : UMOVE   BEGIN  -1 +DECICE  ?INSTALLED  END ;
10
11 : movecur   2CONSTANT  DOES>  2@ +COL  #COL @ = IF
12    16 +DEVICE  ?INSTALLED NOT IF  UMOVE  THEN  THEN ;
13
14 0  1 movecur RMOVE
15 3 -1 movecur LMOVE

Screen:  664
 0 ( Master - Parameter Editor Functions )
 1 : PTABLE   #COL @ params ;
 2 : HIGHLIGHT   REVERSE  PUTCUR  PTABLE @ .NUM ;
 3 : res   NORMAL  PUTCUR  PTABLE @ .NUM SPACE ;
 4
 5 : NUMIN   DUP EMIT  +INPUT ;
 6 : MODIFY   REVERSE PUTCUR 6 SPACES PUTCUR NUMIN PTABLE ! ;
 7
 8 : .PED-BUFFER   20 0 CURSOR  15 SPACES  13 EMIT
 9    ." BUFFER= " PED-BUFFER ? ;
10
11 : TRANSFER   PTABLE @ PED-BUFFER ! .PED-BUFFER ;
12 : INSERT#   PED-BUFFER @ PTABLE ! ;
13
14 : ED-TITLE   2 4 CURSOR SCUR  64 SPACES  RCUR  I/O>
15    'PTITLE DUP  64 BLANK  64 EXPECT  >I/O  UPDATE FLUSH ;
```

```
Screen:  665
 0 ( Master - Parameter Editor Commands )
 1
 2 : PED-CMDS   CASES          11  <CASE  UMOVE             ECASE>
 3                             22  <CASE  DMOVE             ECASE>
 4                             12  <CASE  RMOVE             ECASE>
 5                              8  <CASE  LMOVE             ECASE>
 6                              0  <CASE  0 #COL !  DMOVE   ECASE>
 7                        ( C) 67  <CASE  TRANSFER          ECASE>
 8                        ( I) 73  <CASE  INSERT#           ECASE>
 9                        ( K) 75  <CASE  DROP  >KNOBS  81  ECASE>
10                        ( M) 77  <CASE  +MODE  STAT       ECASE>
11                        ( R) 82  <CASE  STAT 0 !DEVICE#   ECASE>
12                        ( S) 83  <CASE  DROP >SPLITS  81  ECASE>
13                        ( T) 84  <CASE  ED-TITLE          ECASE>
14                        ( Z) 90  <CASE  0 PTABLE !        ECASE>
15       8 EMIT SPACE 8 EMIT DROP    ALSO   HIGHLIGHT    ENDCASE ;

Screen:  666
 0 ( Master - Parameter Editor - PED )
 1
 2 : KEYHIT   KEY DUP  27 = IF  DROP KEY DROP KEY  30 - THEN ;
 3
 4 : (PED)   >NULL  IMODE @ STAT  0 #COL !  0 !DEVICE#  HIGHLIGHT
 5    BEGIN  KEYHIT res DUP DUP
 6       45 58 WITHIN IF  MODIFY  HIGHLIGHT  ELSE  PED-CMDS  THEN
 7    81 = END res SPACE  21 0 CURSOR  IMODE !  ?VALUES  DINIT
 8    SCRNEXIT ;
 9
10  ' (PED) 'PED !
11
12
13
14
15

Screen:  667
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 668
 0
 1
 2
 3
 4
 5
 6 **
 7 **
 8 **
 9 **
10
11
12
13
14
15

Screen: 669
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 670
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen:  671
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  672
 0 ( Master - Softknobs Interrupt Service )   HEX
 1 4 ARRAY KNOBVALUES     0 KNOBVALUES 8 ERASE
 2
 3 FBC0 CONSTANT INKNOBS
 4
 5 ( Softknobs Interrupt Handler )
 6
 7 ASSEMBLER CREATE <KNOBS>   W PUSH   1 PUSH   0 PUSH
 8    ' KNOBVALUES # W MOV   4 # 1 MOV   INKNOBS LDA B
 9    BEGIN   0 SAR B   CS IF   W ) INC   THEN
10            0 SAR B   CS IF   W ) DEC   THEN   W INC   W INC
11    LOOP   0 POP   1 POP   W POP   IRET
12
13 <KNOBS> 14 INTERRUPT
14
15 DECIMAL

Screen:  673
 0 ( Master - Softknobs Support )
 1
 2 4 ARRAY KDEVICE     ( device# associated with knob )
 3
 4 : KUPDATE   4 0 DO  I KDEVICE @ DUP -1 = NOT
 5    IF  !DEVICE#  I KNOBVALUES @ ?DUP
 6       IF  'CURRENT @ + SET  THEN
 7    ELSE  DROP  THEN  0 I KNOBVALUES !  LOOP  13 MS ;
 8
 9 60 60 20 BACKGROUND KNOBSTASK
10
11
12
13
14
15
```

```
Screen:  674
 0 ( Master - Knobs manual on/off )
 1
 2 : KNOBSON   KNOBSTASK ACTIVATE  BEGIN  KUPDATE  0 END  STOP ;
 3
 4 : KNOBSOFF   KNOBSTASK KILL  KUPDATE ;
 5
 6 KNOBSTASK BUILD
 7
 8
 9
10
11
12
13
14
15

Screen:  675
 0 ( Master - Softknobs Display Table Functions )
 1 : HLINE   11 SPACES  57 0 DO  95 EMIT  LOOP  CR ;
 2
 3 : .KDEV ( n)   5 SPACES  KDEVICE @ DUP -1 = NOT
 4    IF  .DNAME  ELSE  DROP  3 SPACES  THEN  5 SPACES ;
 5
 6 : .13-SP ( n)   DROP  13 SPACES ;
 7
 8 : .K# ( n)   DUP  2 SPACES  KDEVICE @ -1 = NOT
 9    IF  KDEVICE @ !DEVICE#  'CURRENT @ .NUM
10    ELSE  DROP 6 SPACES  THEN  5 SPACES ;
11
12 : .KSEG ( n)   DROP  13 0 DO  95 EMIT  LOOP ;
13
14 :CASE KBOX   .KDEV  .13-SP  .K#  .KSEG ;
15

Screen:  676
 0 ( Master - Softknobs Display Table )
 1
 2 : KREPORT   4 0 DO  10 SPACES
 3    4 0 DO  124 EMIT  I J KBOX  LOOP  124 EMIT  CR  LOOP  CR
 4    7 SPACES ." Strike keys 0-9 to select default assignment, str
 5 ike Q to exit."  CR ;
 6
 7 : .KNOBS   7936 SCREEN  CR  KREPORT ;
 8
 9
10
11
12
13
14
15
```

```
Screen:  677
 0 ( Master - Knobs Setup )
 1
 2 : KLINE ( n)   >R  BLK @ I/O>  7936 BLK !  R>
 3    3 + 64 * 12 +  >IN !  40 CNT !  4 0 DO  ' EXECUTE
 4    #DEVICE @ I KDEVICE !  LOOP  >I/O  BLK ! ;
 5
 6 : KSET   DUP 0 10 WITHIN NOT IF  3 .ERROR  THEN
 7    3 + 64 * 13 + 7936 BLOCK +  4 0 DO  32 TEXT  DUP I 14 * +
 8    PAD SWAP 3 MOVE  LOOP  DROP  UPDATE  FLUSH ;
 9
10 : KSAVE   FLUSH EMPTY-BUFFERS  7936 7937 COPY FLUSH ;
11 : KGET    FLUSH EMPTY-BUFFERS  7937 7936 COPY FLUSH ;
12
13 1 0 KDEVICE !  2 1 KDEVICE !  3 2 KDEVICE !  12 3 KDEVICE !
14
15

Screen:  678
 0 ( Master - Knobs )
 1
 2 : knob-keys   CASES
 3 ( M )   77 <CASE  +MODE  IONMODE    ECASE>
 4 ( P )   80 <CASE  DROP  >PED     81 ECASE>
 5 ( S )   83 <CASE  DROP  >SPLITS  81 ECASE>  DROP  ENDCASE ;
 6
 7 : (KNOBS)   >NULL  .KNOBS IONMODE -CURSOR  WLEVEL @  0 WLEVEL !
 8    BEGIN  19 10 CURSOR  4 0 DO  124 EMIT  I .K#  LOOP
 9    KUPDATE  ?TERMINAL 0 ?KEY C!  DUP DUP  48 - 0 10 WITHIN
10    IF  48 - KLINE  17 0 CURSOR KREPORT  ELSE  knob-keys  THEN
11    81 = END  WLEVEL !  +CURSOR  22 78 CURSOR SCRNEXIT ;
12
13 ' (KNOBS) 'KNOBS !   INKNOBS C@ DROP
14
15

Screen:  679
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  680
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  681
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  682
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  683
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  684
 0 ( Ion Path - SIM/MRM Variable Definitions )
 1
 2 VARIABLE #SIMS   COMMAND
 3
 4 VARIABLE MASSES 62 ALLOT   COMMAND
 5
 6 VARIABLE PNEXT
 7 VARIABLE PVALUES  510 ALLOT   COMMAND
 8
 9
10
11
12
13
14
15

Screen:  685
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  686
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  687
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  688
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  689
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  690
 0 ( Master - Variables for MRM/SIM Operations )
 1
 2 1LABEL #SIMS #sims     1LABEL MASSES MASSES
 3 1LABEL PVALUES PVALUES
 4
 5 VARIABLE #SIMS
 6 : !#SIMS ( n)   DUP #sims I!  #SIMS ! ;
 7
 8 2 8 MATRIX SIM-TABLE     0 0 SIM-TABLE 32 ERASE
 9 4 8 MATRIX MRM-TABLE     0 0 MRM-TABLE 64 ERASE
10
11 : SCHK ( n)   1 9 WITHIN NOT IF  4 .ERROR  THEN ;
12
13 0 !#SIMS
14
15

Screen:  691
 0 ( Master - Screen Editor for SIM - Display )
 1
 2 : .SIMTITLE   25 SPACES  ." Selected Ion Monitoring Editor"
 3    +CR ;
 4
 5 [R  \                      NUMBER        ION      PARAMETER SET
 6  ]  DUP 2+ 2+
 7 : SIMED-TITLE   LITERAL [ , ] ASSIGN +L .SIMTITLE
 8    RPT @ HEADING ;
 9
10 : .IONS   8 0  DO  I DUP DUP  25 SPACES  1+ .
11                  7 SPACES  0 SWAP SIM-TABLE @ N.1
12                  9 SPACES  1 SWAP SIM-TABLE @ 3 U.R  +CR  LOOP ;
13
14 : .SIMED   SIMED-TITLE  LITERAL [ , ] REPORT  +CR  .IONS ;
15
```

```
Screen:  692
 0 ( Master - Screen Editor for SIM - Utilities )
 1
 2 : sim.   #COL @ IF  3 U.R  ELSE  N.1  THEN ;
 3
 4 : SCURSOR   #SIMS @ 6 +  #COL @ IF  49  ELSE  34  THEN  CURSOR ;
 5
 6 : Stable ( - a)   #COL @ #SIMS @ SIM-TABLE ;
 7
 8 : SHILITE   REVERSE  SCURSOR  Stable @ sim. ;
 9
10 : SLOLITE   NORMAL  SCURSOR  Stable @ sim.  SPACE ;
11
12 : SMODIFY   REVERSE  SCURSOR  #COL @ IF  3 SPACES  ELSE
13    6 SPACES  THEN  SCURSOR  NUMIN  Stable ! ;
14
15


Screen:  693
 0 ( Master - Screen Editor for SIM - Keyboard Monitor )
 1
 2 : +SIM ( n)   #SIMS @ + 7 AND !#SIMS  SCURSOR ;
 3 : SCOL   #COL @ 1 XOR #COL !  SCURSOR ;
 4
 5 : SKEYS   CASES  11 <CASE  -1 +SIM  ( up )    ECASE>
 6                  22 <CASE   1 +SIM  ( down )  ECASE>
 7                  12 <CASE   SCOL    ( right ) ECASE>
 8                   8 <CASE   SCOL    ( left )  ECASE>
 9                   0 <CASE   1 +SIM  ( cr )    ECASE>
10                  90 <CASE   0 Stable ! ( Z )  ECASE>
11   8 EMIT SPACE 8 EMIT  DROP  ALSO  SHILITE  ENDCASE ;
12
13
14
15


Screen:  694
 0 ( Master - Screen Editor for SIM )
 1
 2 : SLEAVE   8 !#SIMS  8 0 DO  0 I SIM-TABLE @ 0=
 3    1 I SIM-TABLE @ 0=  AND IF  I !#SIMS  LEAVE  THEN  LOOP ;
 4
 5 : SIMED   NORMAL  .SIMED  0 #COL !  0 !#SIMS  SHILITE
 6    BEGIN  KEYHIT  SLOLITE  DUP DUP  46 58 WITHIN
 7       IF  SMODIFY  SHILITE  ELSE  SKEYS  THEN  81 = END
 8    SLOLITE  SPACE  SLEAVE  15 78 CURSOR ;
 9
10
11
12
13
14
15
```

```
Screen:  695
 0 ( Master - Get and Save SIM Tables )
 1
 2 : SIMSAVE ( n)   DUP  0 32 WITHIN IF  32 * 8067 BLOCK +
 3    0 0 SIM-TABLE  SWAP  32 <CMOVE  8067 IDENTIFY UPDATE FLUSH
 4    ELSE DROP  5 .ERROR  THEN ;
 5
 6 : SIMGET ( n)   DUP  0 32 WITHIN IF  32 * 8067 BLOCK +
 7    0 0 SIM-TABLE 32 <CMOVE  SLEAVE  ELSE  DROP  5 .ERROR  THEN ;
 8
 9 0 SIMGET
10
11
12
13
14
15

Screen:  696
 0 ( Master - Screen Editor for MRM - Display )
 1 : .RTITLE   22 SPACES  ." Multiple Reaction Monitoring Editor"
 2    +CR ;
 3 [R  \      REACTION     PARENT       QUAD 2 RF     DAUGHTER
 4 PARAMETER SET ] DUP 2+ 2+
 5
 6 : RED-TITLE   LITERAL [ , ] ASSIGN  +L .RTITLE RPT @ HEADING ;
 7
 8 : .RXNS   8 0 DO  I DUP DUP 2DUP  10 SPACES  1+ .
 9                 7 SPACES  0 SWAP MRM-TABLE @ N.1
10                 8 SPACES  1 SWAP MRM-TABLE @ N.1
11                 8 SPACES  2 SWAP MRM-TABLE @ N.1
12                 8 SPACES  3 SWAP MRM-TABLE @ 3 U.R  +CR  LOOP ;
13
14 : .RED   RED-TITLE  LITERAL [ , ] REPORT  +CR  .RXNS ;
15

Screen:  697
 0 ( Master - Screen Editor for MRM - Utilities )
 1
 2 : RX.   #COL @  3 = IF  3 U.R  ELSE  N.1  THEN ;
 3
 4 : RCURSOR   #SIMS @ 6 +  #COL @ 14 * 19 + CURSOR ;
 5
 6 : MTABLE ( - a)   #COL @ #SIMS @ MRM-TABLE ;
 7
 8 : RXHILITE   REVERSE  RCURSOR  MTABLE @ RX. ;
 9
10 : RXLOLITE  NORMAL  RCURSOR  MTABLE @ RX.  SPACE ;
11
12 : RXMODIFY   REVERSE  RCURSOR  #COL @ 3 < IF  3 SPACES  THEN
13    3 SPACES RCURSOR  NUMIN  MTABLE ! ;
14
15
```

```
Screen:  698
 0 ( Master - Screen Editor for MRM - Keyboard Monitor )
 1
 2 : +RXN ( n)   #SIMS @ + 7 AND !#SIMS  RCURSOR ;
 3 : +REDCOL   +COL  RCURSOR ;
 4
 5 : REDKEYS   CASES    11 <CASE  -1 +RXN     ( up)      ECASE>
 6                      22 <CASE   1 +RXN     ( down)    ECASE>
 7                      12 <CASE   1 +REDCOL  ( right)   ECASE>
 8                       8 <CASE  -1 +REDCOL  ( left)    ECASE>
 9                       0 <CASE   1 +RXN     ( cr)      ECASE>
10                      90 <CASE   0 MTABLE ! ( Z)       ECASE>
11          8 EMIT  SPACE  8 EMIT  DROP  ALSO  RXHILITE  ENDCASE ;
12
13
14
15

Screen:  699
 0 ( Master - Screen Editor for MRM )
 1
 2 : REDLEAVE   8 !#SIMS
 3    8 0 DO  0 I MRM-TABLE @ 0=  2 I MRM-TABLE @ 0= AND
 4       IF  I !#SIMS  LEAVE  THEN  1 I MRM-TABLE @ 0=
 5       IF  0 I MRM-TABLE @  2 I MRM-TABLE @  MIN 2/
 6          1 I MRM-TABLE !  THEN  LOOP ;
 7
 8 : RED   NORMAL  .RED  0 #COL !  0 !#SIMS  RXHILITE
 9    BEGIN  KEYHIT  RXLOLITE DUP DUP  46 58 WITHIN
10       IF  RXMODIFY  RXHILITE  ELSE  REDKEYS  THEN  81 =
11    END  RXLOLITE  SPACE  REDLEAVE  15 78 CURSOR ;
12
13
14
15

Screen:  700
 0 ( Master - Get and Save MRM Tables )
 1
 2 : RSAVE ( n)   DUP 0 16 WITHIN
 3    IF  64 * 8068 BLOCK +  0 0 MRM-TABLE  SWAP 64 <CMOVE
 4       8068 IDENTIFY UPDATE FLUSH
 5    ELSE  DROP  6 .ERROR  THEN ;
 6
 7 : RGET ( n)   DUP 0 16 WITHIN
 8    IF  64 * 8068 BLOCK +  0 0 MRM-TABLE  64 <CMOVE
 9       REDLEAVE  ELSE  DROP 6 .ERROR  THEN ;
10
11 0 RGET
12
13
14
15
```

```
Screen:  701
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  702
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  703
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  704
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  705
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  706
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  707
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  708
 0 ( Ion Path - DC/RF Relay Control )
 1 FORTH : dc/rf ( value address)   CREATE  , , DOES>  2@ C! ;
 2 HEX
 3  0 FC98 dc/rf RF1
 4  0 FC99 dc/rf RF2
 5  0 FC9A dc/rf RF3
 6 80 FC98 dc/rf DC1
 7 80 FC99 dc/rf DC2
 8 80 FC9A dc/rf DC3     DECIMAL
 9
10 :CASE DC   NULL  DC1  DC2  DC3 ;   COMMAND
11 :CASE RF   NULL  RF1  RF2  RF3 ;   COMMAND
12 : RRR   RF1  RF2  RF3 ;  COMMAND
13 : RRD   RF1  RF2  DC3 ;   COMMAND
14 : DRR   DC1  RF2  RF3 ;   COMMAND
15 : DRD   DC1  RF2  DC3 ;   COMMAND

Screen:  709
 0 ( Ion Path - Device Increments )
 1
 2 CODE +DEVICE   SWEEPDEV W MOV   W SHL   STEP 2 MOV
 3    2 ' SCRATCH W) ADD   ' SCRATCH W) 0 MOV
 4    device-address W) W MOV   0 W ) MOV   NEXT
 5
 6 CODE +Q1   M1SCRATCH # W MOV   W ) 0 MOV   STEP 0 ADD
 7    0 W ) MOV   M1DAC STA   0 SHR   M2DAC STA   NEXT
 8
 9 CODE +Q3   M3SCRATCH # W MOV   W ) 0 MOV   STEP 0 ADD
10    0 W ) MOV   M3DAC STA   0 SHR   M2DAC STA   NEXT
11
12
13
14
15
```

```
Screen:  710
 0 ( Ion Path - Device Increments, Continued )
 1
 2 CODE +Q1>3   M1SCRATCH # W MOV   W ) 0 MOV   STEP 0 ADD
 3    0 W ) MOV   M1DAC STA   0 SHR   M2DAC STA   M3DAC STA   NEXT
 4
 5 CODE +Q3>1   M3SCRATCH # W MOV   W ) 0 MOV   STEP 0 ADD
 6    0 W ) MOV   M3DAC STA   0 SHR   M2DAC STA   M1DAC STA   NEXT
 7
 8 CODE +Q13   M1SCRATCH # W MOV   W ) 0 MOV   STEP 0 ADD
 9    0 W ) MOV   M1DAC STA   FQ3 2 MOV   FQ3 2+ LDA   FRAC 2 ADD
10    FRAC 2+ 0 ADC   2 FQ3 MOV   FQ3 2+ STA   M3DAC STA   0 SHR
11    M2DAC STA   NEXT
12
13
14
15

Screen:  711
 0 ( Ion Path - Oscilloscope Control )
 1
 2 VARIABLE HAXIS
 3 VARIABLE HSTEPSIZE
 4
 5 HEX CODE -HAXIS   E00 # 1 MOV   1 HDAC MOV
 6    E000 # 1 MOV   1 HAXIS MOV   NEXT   DECIMAL
 7
 8 : !HSTEP ( #steps)   16384 SWAP U/  HSTEPSIZE ! ;
 9
10 CODE +HAXIS   HAXIS LDA   HSTEPSIZE 0 ADD   HAXIS STA
11    0 SHR   0 SHR   0 SHR   0 SHR   HDAC STA   NEXT
12
13 : SCOPE ( n)   5 MOD   SCOPE-MUX C! ;   COMMAND
14
15

Screen:  712
 0 ( Ion Path - Scanning Support Words )
 1
 2 : 3>DAC ( n n' n" - m m' m")   >DAC >R  >DAC >R  >DAC  R>
 3     R> ;
 4
 5 : #STEPS ( step end start - #steps start )   DUP >R
 6    - OVER U/  SWAP STEP !  R> ;
 7
 8 : Q1INIT ( stdac)   DUP (2/)  M2 !DATA
 9    DUP  M1  !DATA  !SCRATCH ; COMMAND
10
11 : Q3INIT ( stdac)   DUP  (2/)  M2 !DATA
12    DUP  M3  !DATA  !SCRATCH ; COMMAND
13
14
15
```

```
Screen:  713
 0 ( Ion Path - Setup for Neutral Loss Scanning )
 1
 2 CODE 3DUP   S W MOV   4 W) PUSH   2 W) PUSH   W ) PUSH   NEXT
 3
 4 : NSET   >R   3DUP  2DUP -  R> OVER +  DUP ROT +  >DAC
 5    SWAP  >DAC  2DUP  2/ M2DAC !   0 SWAP FQ3 2!  - >R
 6    M1 3>DAC -  R> SWAP >R  U* I M/MOD FRAC !  65535 R> */MOD
 7    FRAC 2+ !  DROP ;
 8
 9
10
11
12
13
14
15

Screen:  714
 0 ( Sweep DAC&STEP Function - MJK 11/22/85 )
 1
 2 VARIABLE DIRECTION
 3
 4 : >DAC&STEP  ( step, end , start -> #dacsteps, dacstart )
 5     DUP >DAC >R - DUP 0< IF -1 ELSE 1 THEN
 6     DIRECTION ! SWAP  DUP >DAC  0 >DAC  -
 7     STEP !   / ABS  R> ;
 8
 9
10
11
12
13
14
15

Screen:  715
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  716
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  717
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  718
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  719
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  720
 0 ( Ion Path - Fast Scanning Support )
 1 VARIABLE FPARAMS  2 ALLOT
 2 VARIABLE fspeed   COMMAND
 3 VARIABLE FSMODE   ( scan routine vector addr )
 4
 5 60 60 20 BACKGROUND scanner
 6
 7 : FSTOP   scanner KILL  DINIT  -HAXIS  4 SCOPE ;  COMMAND
 8     ( stop fast scan )
 9
10 : FINIT ( dacstep enddac startdac )   #STEPS OVER  !HSTEP
11     FPARAMS 2!  ;
12
13 : PAUSES ( n)   0 DO  PAUSE  LOOP ;
14 : FPAUSE   fspeed @ PAUSES ;
15

Screen:  721
 0 ( Ion Path - Fast Scanning Primatives )
 1
 2 : (F1SCAN) ( #steps startdac )   ( fast scan quad 1 )
 3     Q1INIT  0 DO  +Q1>3  +HAXIS  FPAUSE  LOOP ;
 4
 5 : (F3SCAN) ( #steps startdac )   ( fast scan quad 3 )
 6     Q3INIT  0 DO  +Q3>1  +HAXIS  FPAUSE  LOOP ;
 7
 8 : (FPSCAN) ( #steps startdac )   ( fast scan parents )
 9     Q1INIT  0 DO  +Q1  +HAXIS  FPAUSE  LOOP ;
10
11 : (FDSCAN) ( #steps startdac )   ( fast scan daughters )
12     Q3INIT  0 DO  +Q3  +HAXIS  FPAUSE  LOOP ;
13
14 : (FNSCAN) ( #steps startdac )   ( fast scan neutrals )
15    DUP  Q1INIT Q3INIT  0 DO  +Q13  +HAXIS  FPAUSE  LOOP ;
```

```
Screen:  722
 0 ( Ion Path - Fast Scanning Task Initialization )
 1
 2 : (FSCAN) ( dacsteps dacstart)   FSMODE @ EXECUTE ;
 3
 4 : [FSCAN]  ( step end start )  FINIT   scanner ACTIVATE
 5    BEGIN  -HAXIS  FPARAMS 2@  (FSCAN)  0 END  STOP ;
 6
 7 FORTH : fscans  CREATE  '  ,  DOES>  @  FSMODE !  [FSCAN]  ;
 8 fscans  [F1SCAN]  (F1SCAN)
 9 fscans  [F3SCAN]  (F3SCAN)
10 fscans  [FPSCAN]  (FPSCAN)
11 fscans  [FDSCAN]  (FDSCAN)
12 fscans  [FNSCAN]  (FNSCAN)
13
14 : [F1SCAN]  [F1SCAN] ; COMMAND
15 : [F3SCAN]  [F3SCAN] ; COMMAND

Screen:  723
 0 ( Ion Path - Fast Scanning Commands )
 1
 2 : F1SCAN  ( step end start in units )
 3    DRR  M1 3>DAC    [F1SCAN] ;     COMMAND
 4
 5 : F3SCAN  ( step end start in units )
 6    RRD  M3 3>DAC    [F3SCAN] ;     COMMAND
 7
 8 : FPSCAN  ( step end start in units )
 9    DRD  M1 3>DAC    [FPSCAN] ;     COMMAND
10
11 : FDSCAN  ( step end start in units )
12    DRD  M3 3>DAC    [FDSCAN] ;     COMMAND
13
14 : FNSCAN  ( step end start M3 offset in units )
15    DRD  NSET  M1 3>DAC    [FNSCAN] ;     COMMAND

Screen:  724
 0 ( Ion Path - Fast Sweep Functions )
 1
 2 : (FSWEEP)  ( #steps startdac )    SWEEPDEV @ !DEVICE#
 3    DUP !DATA !SCRATCH    0 DO  +DEVICE  +HAXIS
 4    FPAUSE LOOP ;
 5
 6 : [FSWEEP]  ( #steps startdac )   OVER !HSTEP  FPARAMS 2!
 7    scanner ACTIVATE
 8    BEGIN  -HAXIS  FPARAMS 2@  (FSWEEP)  0 END  STOP ;
 9
10 : FSWEEP   ( step end start #device )
11    DUP !DEVICE# SWEEPDEV !  >DAC&STEP  [FSWEEP] ; COMMAND
12
13
14
15
```

```
Screen:  725
 0 ( Ion Path - Xscans )
 1 : xparams   ROT  DROP  1  ROT  ROT ;
 2 : X1SCAN ( step end start in units )   DRR  M1
 3    3>DAC  xparams  [F1SCAN] ;     COMMAND
 4
 5 : X3SCAN ( step end start in units )   RRD  M3
 6    3>DAC  xparams  [F3SCAN] ;     COMMAND
 7
 8 : XPSCAN ( step end start in units )   DRD  M1
 9    3>DAC  xparams  [FPSCAN] ;     COMMAND
10
11 : XDSCAN ( step end start in units )   DRD  M3
12    3>DAC  xparams  [FDSCAN] ;     COMMAND
13
14 : XNSCAN ( step end start in units )   DRD  NSET  M1
15    3>DAC  xparams  [FNSCAN] ;     COMMAND

Screen:  726
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  727
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  728
 0 ( Master - Variables for MS Graphics )
 1
 2 2LABEL  DSTART        DSTART
 3 2LABEL  D#FIELDS      D#FIELDS
 4 2LABEL  DUNITS/FIELD  DUNITS/FIELD
 5 2LABEL  DNORM         DNORM
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  729
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  730
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  731
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  732
 0 ( Master - Load Block for Fast Scanning )
 1
 2 CR { Loading Fast Scans and Splits }
 3
 4 734 738 THRU ( Fast Scans )
 5 750 759 THRU ( Splits )
 6 792 794 THRU ( ASET )
 7 798 799 THRU ( SSET )
 8
 9
10
11
12
13
14
15

Screen:  733
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  734
 0 ( Master - QPS RF/DC Control )
 1
 2 : DC ( n)   1PUSH  SL1 DC ;
 3 : RF ( n)   1PUSH  SL1 RF ;
 4
 5 : RRD   SL1 RRD ;
 6 : DRR   SL1 DRR ;
 7 : DRD   SL1 DRD ;
 8 : RRR  SL1 RRR ;
 9
10
11
12
13
14
15

Screen:  735
 0 ( Master - Scanning Support )
 1
 2 : @PARAMS   'STEP @ 1PUSH  'END @ 1PUSH  'START @ 1PUSH ;
 3
 4 : NSET   M3 'START @  M1 'START @  - 1PUSH ;
 5
 6 HEX FC41 CONSTANT scope DECIMAL
 7 : SCOPE ( n)   5 MOD  1 #SLAVE  scope IC! ;     4 SCOPE
 8
 9 1LABEL fspeed fspeed
10
11 : FSPEED ( n)   fspeed I! ;     1 FSPEED
12
13
14
15

Screen:  736
 0 ( Master - Fast Scan Commands )
 1
 2 : FSTOP SL1 FSTOP ;
 3
 4 : F1SCAN   FSTOP M1 @PARAMS  SL1 F1SCAN ;
 5
 6 : F3SCAN   FSTOP M3 @PARAMS  SL1 F3SCAN ;
 7
 8 : FPSCAN   FSTOP M1 @PARAMS  SL1 FPSCAN ;
 9
10 : FDSCAN   FSTOP M3 @PARAMS  SL1 FDSCAN ;
11
12 : FNSCAN   FSTOP M1 @PARAMS  NSET  SL1 FNSCAN ;
13
14 : FSWEEP   FSTOP @PARAMS  #DEVICE @ 1PUSH  SL1 FSWEEP ;
15
```

```
Screen:  737
 0 ( Master - Xscans )
 1
 2 : X1SCAN   M1 @PARAMS  SL1 X1SCAN ;
 3
 4 : X3SCAN   M3 @PARAMS  SL1 X3SCAN ;
 5
 6 : XPSCAN   M1 @PARAMS  SL1 XPSCAN ;
 7
 8 : XDSCAN   M3 @PARAMS  SL1 XDSCAN ;
 9
10 : XNSCAN   M1 @PARAMS  NSET  SL1 XNSCAN ;
11
12
13
14
15

Screen:  738
 0 ( Master - Tuning Aids )
 1
 2 : RANGESETUP ( mass)   50 - DUP 100 +  !END  !START ;
 3
 4 : X1 ( mass)   M1 RANGESETUP  X1SCAN ;
 5
 6 : X3 ( mass)   M3 RANGESETUP  X3SCAN ;
 7
 8 : XP ( m1 m3)   M3 SET  M1 RANGESETUP  XPSCAN ;
 9
10 : XD ( m1 m3)   M3 RANGESETUP  M1 SET  XDSCAN ;
11
12 : XN ( m1 n1)   OVER  M1 RANGESETUP - M3 RANGESETUP  XNSCAN ;
13
14
15

Screen:  739
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  740
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  741
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  742
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  743
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  744
 0 ( Ion Path - Split Screen Operations )
 1
 2 VARIABLE #SPEAKS
 3 5 ARRAY SMASS   COMMAND
 4 5 ARRAY SGAINS  COMMAND
 5 5 2ARRAY SPARAMS
 6
 7 : SINIT   655 !HSTEP  0 #SPEAKS !  5 0 DO  I SMASS @ ?DUP
 8    IF  I SPARAMS !  1 #SPEAKS +!  THEN  LOOP  #SPEAKS @ DUP
 9    STEP !  DUP 655 SWAP / SWAP  0 DO  DUP  I SPARAMS @ 25 -
10    >DAC  I SPARAMS 2!  LOOP  DROP ;
11
12 : [SPLIT]   SINIT  scanner ACTIVATE  BEGIN  -HAXIS  #SPEAKS @
13    0 DO  I SGAINS @ SCOPE  I SPARAMS 2@ (FSCAN)  LOOP
14    0 END  STOP ;
15

Screen:  745
 0 ( Ion Path - Split Screen Task Initialization )
 1
 2 FORTH : splits   CREATE  ' ,  DOES>  @ FSMODE ! [SPLIT] ;
 3
 4 splits [1SPLIT]  (F1SCAN)
 5 splits [3SPLIT]  (F3SCAN)
 6 splits [PSPLIT]  (FPSCAN)
 7 splits [DSPLIT]  (FDSCAN)
 8 splits [NSPLIT]  (FNSCAN)
 9
10
11
12
13
14
15
```

```
Screen:  746
 0 ( Ion Path - Split Screen Commands )
 1
 2 : 1SPLIT   DRR  M1 [1SPLIT] ;     COMMAND
 3
 4 : 3SPLIT   RRD  M3 [3SPLIT] ;     COMMAND
 5
 6 : PSPLIT   DRD  M1 [PSPLIT] ;     COMMAND
 7
 8 : DSPLIT   DRD  M3 [DSPLIT] ;     COMMAND
 9
10 : NSPLIT   DRD  NSET  M1 [NSPLIT] ;     COMMAND
11
12 :CASE SPLITS   1SPLIT  3SPLIT  PSPLIT  DSPLIT  NSPLIT  NULL ;
13    COMMAND
14
15

Screen:  747
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  748
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  749
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  750
 0 ( Master - Split Screen Window Table )
 1 6 5 MATRIX SSWT     ( matrix for window amu's )
 2 6 5 MATRIX SSGT     ( matrix for window gains )
 3
 4 VARIABLE #SPLIT     ( # for type of scan { row # } )
 5    0 #SPLIT !
 6
 7 1LABEL SMASS SMASS
 8 1LABEL SGAINS SGAINS
 9
10 : >SPARAMS   #SPLIT @ 0 SSWT SMASS 10  0 1 >SLAVE IPMOVE
11    #SPLIT @ 0 SSGT SGAINS 10 0 1 >SLAVE IPMOVE ;
12
13 14 7938 9 LABELS STNAME   ( scan type name )
14 : .SNAME ( n)   STNAME >TYPE ;
15

Screen:  751
 0 ( Master - Split Screen Commands )
 1
 2 : splits   CREATE  C,  DOES>  C@ DUP  #SPLIT !  >SPARAMS
 3    1PUSH  SL1 SPLITS ;
 4
 5 0 splits 1SPLIT
 6 1 splits 3SPLIT
 7 2 splits PSPLIT
 8 3 splits DSPLIT
 9 4 splits NSPLIT
10
11
12
13
14
15

Screen:  752
 0 ( Master - Split Screen Gain Control Words )
 1
 2 CREATE AMP#   256 , 64 , 16 , 4 , 1 ,
 3
 4 : .XN ( n)   0 <# #S 120 HOLD #> .LEFT ;
 5
 6 : .GAIN ( col# row#)   SSGT @ 2* AMP# + @ .XN ;
 7
 8 : ATOGL   #SPLIT @ #COL @ SSGT DUP  @ 1+ 5 MOD  DUP  ROT !
 9    2* AMP# + @ 120 EMIT . ;
10
11
12
13
14
15
```

```
Screen:  753
 0 ( Master - Split Screen Table Display )
 1
 2 : .STITLE   ( type split screen table display title)  5 SPACES
 3  ." SPLIT SCREEN SCAN - WINDOW CENTERS (AMU) & RELATIVE AMPLIFIC
 4 ATIONS     " IMODE C@ .PMNAME  +CR ;
 5 [R  \SCAN-TYPE     FIRST gain  SECOND gain   THIRD gain  FOURTH
 6 gain   FIFTH gain] DUP 2+ 2+      ( column headers )
 7
 8 : SPLIT-TITLE   LITERAL [ , ] ASSIGN +L .STITLE RPT @ HEADING ;
 9
10 : WINDOWS   ( types out split screen windows table )
11    SPLIT-TITLE LITERAL [ , ] REPORT
12    76 0 DO  45 EMIT  LOOP   CR     6 0 DO  I STNAME RIGHT
13    5 0 DO  J I 2DUP SSWT  @ N.1R .GAIN  LOOP  +L LOOP  CR
14  19 SPACES ." DAUGHTER (PSPLIT) ="  M3 'CURRENT @ N.1 10 SPACES
15    ." PARENT (DSPLIT) ="  M1  'CURRENT @ N.1   CR   7938 LOAD ;

Screen:  754
 0 ( Master - Split Screen Window Table Editor & Cursor )
 1
 2 : +COLS ( n)   #COL @  +  5 MOD  #COL ! ;   ( chng col)
 3 : +ROWS ( n)   #SPLIT @  + 6 MOD  #SPLIT ! ;   ( chng row)
 4
 5 : MAPCUR    #SPLIT @  4 +   #COL @ 13 *  13 +  CURSOR ;
 6
 7 : WTABLE   (  -> window addr )   #SPLIT @  #COL @  SSWT ;
 8
 9 : .AMU   MAPCUR  WTABLE @  N.1 ;   ( print number from table )
10
11 : HILITE   REVERSE  .AMU ;   ( highlight number on screen)
12 : LOLITE   NORMAL   .AMU  SPACE ;   ( print number on screen)
13
14 : WMODIFY   REVERSE  MAPCUR  6 SPACES  MAPCUR  NUMIN  WTABLE ! ;
15             ( put terminal in rev video, accept, enter number)

Screen:  755
 0 ( Master - Split Screen Window Table Editor & Cursor )
 1 6 CARRAY ACTIND   ( ACTivity INDicator )   0 ACTIND 6 BLANK
 2 : ACTFILL  ( update ACTIND )   5 0 DO  I  #SPLIT @  =
 3       IF  42 ( * )  ELSE  32 ( sp )  THEN  I ACTIND !  LOOP
 4          32  5 ACTIND !  ;
 5 : ?ACTIVE     6 0 DO  I 4 + 11 CURSOR I ACTIND @  EMIT  LOOP ;
 6
 7 : GOSPLIT  #SPLIT @   >SPARAMS  1PUSH  SL1 SPLITS  ACTFILL ;
 8
 9 : TUNES  #SPLIT @ DUP DUP  5 #SPLIT ! >SPARAMS 1PUSH SL1 SPLITS
10    #SPLIT ! ACTFILL   5 = IF  32  ELSE  42  THEN  5 ACTIND ! ;
11   ( do split screen for current editor line, with "tune" amus)
12 : DEL-AMU   ( n ->  )   WTABLE +!  GOSPLIT ; ( inc/dec amu)
13 : LINEACT?   #SPLIT @ ACTIND C@ 42 = IF NORMAL  0 ACTIND
14    6 BLANK  ?ACTIVE THEN ;
15
```

```
Screen:  756
 0 ( Master - Set Parent/Daughter Ion Within SPLITS )
 1
 2 VARIABLE ?CUR    2 ALLOT     ( cursor location)
 3
 4 : ?ION   REVERSE   ?CUR 2@ CURSOR  6 SPACES  ?CUR 2@ CURSOR
 5      #INPUT  DUP  SET  ?CUR  2@ CURSOR NORMAL N.1 ;
 6
 7 : SIBLING   11 39 ?CUR 2!   #DEVICE @   M3   ?ION   !DEVICE# ;
 8
 9 : MOTHER    11 72 ?CUR 2!   #DEVICE @   M1   ?ION   !DEVICE# ;
10
11
12
13
14
15

Screen:  757
 0 ( Master - Split Screen Window Editor - Cases )
 1 : SPLITKEYS   CASES        11 <CASE   5 +ROWS ( up )     ECASE>
 2                            22 <CASE   1 +ROWS ( down )   ECASE>
 3                            12 <CASE   1 +COLS ( right )  ECASE>
 4                             8 <CASE   4 +COLS ( left )   ECASE>
 5                     ( < ) 60 <CASE   2 DEL-AMU          ECASE>
 6                     ( > ) 62 <CASE  -2 DEL-AMU          ECASE>
 7                     ( T ) 84 <CASE  TUNES               ECASE>
 8                     ( G ) 71 <CASE  GOSPLIT             ECASE>
 9                     ( R ) 82 <CASE  WINDOWS             ECASE>
10                     ( Z ) 90 <CASE  0 WTABLE !          ECASE>
11                     ( A ) 65 <CASE  ATOGL   LINEACT?    ECASE>
12                     ( D ) 68 <CASE  SIBLING             ECASE>
13  80 <CASE DROP >PED 81 ECASE> 70 <CASE  MOTHER ECASE>
14  75 <CASE DROP >KNOBS 81 ECASE> 77 <CASE  +MODE IONMODE ECASE>
15  8 EMIT  SPACE  8 EMIT   DROP  ALSO  ?ACTIVE  HILITE  ENDCASE ;

Screen:  758
 0 ( Master - Split Screen Windows Editor )
 1
 2 : (SPLITS)   NORMAL WINDOWS  0 #COL !  0 #SPLIT !  ?ACTIVE
 3    HILITE  >NULL  BEGIN  KEYHIT LOLITE DUP DUP  45 58 WITHIN
 4    IF  WMODIFY LINEACT? HILITE  ELSE  SPLITKEYS  THEN
 5    81 ( Q ) = END  LOLITE  23 70 CURSOR  SCRNEXIT ;
 6
 7 : .SPLITS   WINDOWS ;
 8
 9 ' (SPLITS) 'SPLITS !
10
11
12
13
14
15
```

```
Screen:  759
 0 ( Master - Split Screen Table Save and Restore )
 1
 2 : SSAVE  ( splits save)   8069 BLOCK 0 0 SSWT OVER 60 <CMOVE
 3          60 +  0 0 SSGT SWAP  60 <CMOVE  UPDATE  FLUSH ;
 4
 5 : SGET  ( get splits)  8069 BLOCK DUP 0 0 SSWT  60 <CMOVE
 6    60 +  0 0 SSGT  60 <CMOVE  ;
 7
 8 SGET
 9
10
11
12
13
14
15

Screen:  760
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  761
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  762
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  763
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  764
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  765
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  766
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  767
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  768
 0 ( Ion Path - Initialize Amu Timer )
 1
 2 HEX
 3
 4 CREATE RATE-TABLE   0 C, 1 C, 2 C, 4 C, 5 C, 6 C, 8 C, 9 C,
 5    0A C, 0C C, 0D C, 0E C, 10 C, 11 C, 12 C,
 6
 7 VARIABLE rate COMMAND
 8
 9 CODE RATE ( n)   U 1 XCHG   RATE-TABLE # U MOV   0 POP
10    rate STA   D7 C, ( XLAT )   FCC2 STA B   FCC0 STA B
11    1 U XCHG   NEXT
12    ( takes # from RATE-TABLE )    COMMAND
13
14 DECIMAL
15

Screen:  769
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  770
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  771
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  772
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  773
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  774
 0 ( Reduction - Parameters and Variables for Peak Finding )
 1
 2 VARIABLE threshold COMMAND
 3 VARIABLE pwidth    COMMAND
 4 VARIABLE mwidth    COMMAND
 5 VARIABLE xmax  6 ALLOT
 6          xmax  2+ CONSTANT xlast
 7          xmax 4 + CONSTANT max-peak
 8 VARIABLE ylast
 9 VARIABLE xvalue
10 VARIABLE (thld)
11 VARIABLE kflag
12 VARIABLE yprev  2 ALLOT
13 VARIABLE UPFLAG
14 VARIABLE +LAST
15

Screen:  775
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  776
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  777
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  778
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  779
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  780
 0 ( Detection - Data Acquisition Address Definitions )
 1
 2 HEX
 3
 4           F800 CONSTANT daqbase2
 5 daqbase2       CONSTANT #points
 6 daqbase2  1 +  CONSTANT startdaq
 7 daqbase2 1C8 + CONSTANT daqf/fclr
 8 daqbase2 40 +  CONSTANT daqrst
 9 daqbase2  C0 + CONSTANT daqstat
10
11 DECIMAL
12
13
14
15

Screen:  781
 0 ( Detection - Rate Selection )
 1
 2 CREATE averages   3 C,  5 C,  6 C,  7 C,  8 C, 10 C, 11 C, 12 C,
 3                  12 C, 12 C, 12 C, 12 C, 12 C, 12 C, 12 C, 12 C,
 4
 5 HEX
 6 CODE AVERAGES ( n)   U 1 XCHG   averages # U MOV   0 POP
 7    D7 C, ( XLAT )   #points STA   #points STA  ( kluge )
 8    1 U XCHG   NEXT
 9 DECIMAL
10
11 VARIABLE rate
12
13 : RATE ( n)   DUP  0 16 WITHIN
14    IF  DUP  rate !  AVERAGES  ELSE  DROP  THEN ;   COMMAND
15

Screen:  782
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  783
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  784
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  785
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  786
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  787
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  788
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  789
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  790
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  791
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  792
 0 ( Master - Variables for Peak Finding )
 1
 2 2LABEL threshold threshold
 3 2LABEL pwidth    pwidth
 4 2LABEL mwidth    mwidth
 5
 6 VARIABLE (THRESHOLD)
 7 VARIABLE (PWIDTH)
 8 VARIABLE (MWIDTH)
 9 VARIABLE (RATE)
10
11 :CASE AVAR   (THRESHOLD)  (PWIDTH)  (MWIDTH)  (RATE) ;
12
13
14
15

Screen:  793
 0 ( Master - Store Peak Finding Variables )
 1
 2 : Threshold   (THRESHOLD) @ threshold I! ;
 3 : Pwidth      (PWIDTH)    @ pwidth    I! ;
 4 : Mwidth      (MWIDTH)    @ mwidth    I! ;
 5 : Rate        (RATE) @ DUP  1PUSH SL1 RATE  3PUSH SL3 RATE ;
 6
 7 : !THRESHOLD ( n)   (THRESHOLD) !  Threshold ;
 8 : !PWIDTH    ( n)   (PWIDTH)    !  Pwidth    ;
 9 : !MWIDTH    ( n)   (MWIDTH)    !  Mwidth    ;
10 : RATE       ( n)   (RATE)      !  Rate      ;
11
12 : APARAMS   Threshold  Pwidth  Mwidth  Rate ;
13
14
15

Screen:  794
 0 ( Master - Set Data Acquisition Attributes )
 1
 2 6 7939 20 LABELS ANAME   ( acquisiton attributes )
 3 : .ANAME ( n)   ANAME >TYPE ;
 4
 5 : .ASET   PAGE  7940 SCREEN  7939 LOAD
 6    4 0 DO  CR  I .ANAME  I AVAR @ 12 U.R  LOOP  20 SPACES ;
 7
 8 : ASET   .ASET  4 0 DO  20 I + 36 CURSOR  63 EMIT ( ?)
 9    8 EMIT ( BS)  #INPUT  ?DUP IF  I AVAR !  THEN  LOOP  CR
10    APARAMS ;
11
12  300 !THRESHOLD  2 !PWIDTH  5 !MWIDTH  0 RATE
13
14
15
```

```
Screen:  795
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  796
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  797
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  798
 0 ( Master - SIM/MRM Parameter Setting Functions )
 1
 2 VARIABLE RRATE      4 RRATE !
 3 VARIABLE RRANGE     16 RRANGE !
 4 VARIABLE RGAIN      65535 RGAIN !
 5 VARIABLE RTMAX      0 RTMAX !
 6
 7 VARIABLE RRECS
 8
 9 CREATE rxperiods   1 C, 2 C, 4 C, 10 C, 20 C, 40 C,
10
11 : RATECHK   DUP 1 7 WITHIN NOT IF  7 .ERROR  THEN ;
12
13 : STEPTIM ( - step*10)   RRATE @ RATECHK  1- rxperiods + C@ ;
14
15

Screen:  799
 0 ( Master - SIM/MRM Parameter Setting )
 1
 2 6 7941 32 LABELS RNAME
 3 : .RNAME ( n)   RNAME >TYPE ;
 4
 5 :CASE RVAR   RRATE  RRANGE  RGAIN  RTMAX ;
 6
 7 : .SSET   PAGE  7942 SCREEN  7941 LOAD
 8    4 0 DO  CR 5 SPACES  I .RNAME  I RVAR @ 6 U.R  LOOP
 9    20 SPACES ;
10
11 : SSET   .SSET  4 0 DO  20 I + 47 CURSOR  63 EMIT ( ?)
12    8 EMIT ( BS)  #INPUT ?DUP IF  I RVAR !  THEN  LOOP  CR
13    RRATE @ RATECHK  ;
14
15

Screen:  800
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 801
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 802
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 803
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen:  804
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  805
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  806
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  807
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  808
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  809
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 810
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 811
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14   !NOJOON
15

Screen: 812
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen:  813
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  814
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  815
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  816
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  817
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  818
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  819
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  820
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  821
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  822
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  823
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  824
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 825
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 826
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 827
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen:  828
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  829
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  830
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  831
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  832
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  833
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  834
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  835
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  836
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  837
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  838
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  839
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  840
 0 ( Master - Load Block for Data Operations )
 1 CR { Loading Data Operations }
 2 842 854 THRU ( Databuffer and File Definition )
 3 855 864 THRU ( Report Generation )
 4 870 880 THRU ( Data File Maintainence )
 5 882 LOAD     ( Parameter Retrieval )
 6 888 891 THRU ( Add, Sum, and Subtract Spectra )
 7 900 905 THRU ( Data Listings )
 8 941 949 THRU  955 959 THRU  ( DISP )
 9
10
11
12
13
14
15

Screen:  841
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  842
 0 ( INTERFACE EXTENDED MEMORY ADDRESSING )
 1 HEX VARIABLE iface   1000 iface !
 2 ASSEMBLER DEFINITIONS
 3      1000 CONSTANT IFACE    ( 1000 = segment reg. value)
 4         DECIMAL
 5 : [IFACE  STI B ( CLI )   IFACE # W MOV    W DS LSG       ;
 6 : IFACE]                   ZERO # W MOV    W DS LSG   STI  ;
 7 FORTH DEFINITIONS
 8
 9 CODE E@   [IFACE   W POP   W ) PUSH   IFACE]   NEXT
10 CODE E!   [IFACE   W POP   0 POP   0 W ) MOV   IFACE]   NEXT
11 CODE E+!  [IFACE   W POP   0 POP   0 W ) ADD   IFACE]   NEXT
12
13 CODE EC@  [IFACE   W POP   0 0 SUB   W ) 0 MOV B
14                    0 PUSH   IFACE]   NEXT
15 CODE EC!  [IFACE   W POP   0 POP   0 W ) MOV B   IFACE]  NEXT
```

```
Screen:  843
 0 ( INTERFACE EXTENDED MEMORY ADDRESSING )
 1
 2 CODE E2@  [IFACE  W POP    2 W) PUSH   W ) PUSH   IFACE] NEXT
 3
 4 CODE E2!  [IFACE  W POP   W ) POP   2 W) POP   IFACE] NEXT
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  844
 0 ( DATA BUFFER DEFINTION )
 1
 2                 VARIABLE #RECORDS   12 ALLOT
 3     #RECORDS 2+ CONSTANT #POINTS
 4      #POINTS 2+ CONSTANT TIC
 5       TIC 2+ 2+ CONSTANT MAX-INTEN
 6 MAX-INTEN 2+ 2+ CONSTANT rate
 7
 8 ( The databuffer resides in extended memory starting at )
 9 ( address 0H. There is 16K available -- 2048 data points. )
10 ( The actual address is included in the segment reg. value. )
11 HEX
12 0 CONSTANT DATABUFFER
13 DECIMAL
14
15

Screen:  845
 0 ( DATA BUFFER ACCESS )
 1 : 'buffer  CREATE  ,  ;CODE   W INC   W INC   0 POP   0 SHL
 2    0 SHL   0 SHL   W ) 0 ADD   DATABUFFER # 0 ADD   0 PUSH
 3    NEXT
 4
 5 0 'buffer 'X          0 'buffer  '1MASS
 6 2 'buffer 'X1         2 'buffer  '3MASS
 7 2 'buffer 'WIDTH
 8 3 'buffer 'FLAGS
 9 4 'buffer 'Y
10
11 : @X  'X E@ ; : !X 'X E! ;    : @1MASS  '1MASS E@ ;
12 : @X1  'X1 E@ ;               : @3MASS  '3MASS E@ ;
13 : @WIDTH  'WIDTH EC@ ;        : !1MASS  '1MASS E! ;
14 : @FLAGS  'FLAGS EC@ ;        : !3MASS  '3MASS E! ;
15 : @Y  'Y E2@ ;                : !Y  'Y E2! ;
```

```
Screen:  846
 0 ( Master - Data Buffer Definition ) EXIT
 1
 2                 VARIABLE DATABUFFER     8204 ALLOT
 3   DATABUFFER 2+ CONSTANT #POINTS
 4      #POINTS 2+ CONSTANT TIC
 5       TIC 2+ 2+ CONSTANT MAX-INTEN
 6 MAX-INTEN 2+ 2+ CONSTANT rate
 7         rate 2+ CONSTANT databuffer
 8      DATABUFFER CONSTANT #RECORDS
 9
10
11
12
13
14
15

Screen:  847
 0 ( Master - Data Buffer Access ) EXIT
 1 : 'buffer   CREATE  ,  ;CODE   W INC   W INC   0 POP   0 SHL
 2    0 SHL   0 SHL   W ) 0 ADD   databuffer # 0 ADD   0 PUSH
 3    NEXT
 4
 5 0 'buffer 'X            0 'buffer '1MASS
 6 2 'buffer 'X1           2 'buffer '3MASS
 7 2 'buffer 'WIDTH
 8 3 'buffer 'FLAGS
 9 4 'buffer 'Y
10
11 : @X   'X @ ;     : !X   'X ! ;      : @1MASS   '1MASS @ ;
12 : @X1   'X1 @ ;                      : @3MASS   '3MASS @ ;
13 : @WIDTH   'WIDTH C@ ;               : !1MASS   '1MASS ! ;
14 : @FLAGS   'FLAGS C@ ;               : !3MASS   '3MASS ! ;
15 : @Y   'Y 2@ ;    : !Y   'Y 2! ;

Screen:  848
 0 ( Master - File Utilities )
 1
 2 : ?DISK   LIM @  Available @  DUP CR U. ."  Records Used  "
 3    - U. ."  Records Free  " ;
 4
 5 : INITIALIZE   ORG @ BLOCK 1024 ERASE  UPDATE FLUSH
 6    @AVAILABLE ;
 7
 8 : MSCHUNK ( n)   Available @ DUP ROT + LIM @ OVER
 9    U< IF  9 .ERROR  ELSE  Available !   1+ THEN ;
10
11 : MSSLOT   1 MSCHUNK ;
12
13
14
15
```

```
Screen:  849
 0 ( Master - Experiment and Scan Header Field Definitions )
 1
 2 ( Experiment Header )
 3 0 NUMERIC Rflag  16 BYTES Title       NUMERIC Date  DOUBLE Time
 4   NUMERIC #Records  NUMERIC Pheader   NUMERIC #Scans
 5   NUMERIC 1st-scan     DROP
 6
 7 ( Scan Header )
 8 28 NUMERIC Scan#  NUMERIC Stype   NUMERIC Device   NUMERIC Start
 9    NUMERIC End    NUMERIC Step    NUMERIC 1Mass    NUMERIC 3Mass
10    NUMERIC Sec    NUMERIC Svalue  NUMERIC #Points  DOUBLE Tic
11    DOUBLE Max-inten  NUMERIC Srate   ( 62 bytes used ) DROP
12
13 : Imode   Stype 1+ ;
14
15

Screen:  850
 0 ( Master - Experiment Header Support Functions )
 1
 2 VARIABLE CEXPT  1 CEXPT !
 3
 4 : LEXPT   Available 2+ ;
 5
 6 : !LEXPT   LEXPT ! ;
 7
 8 : CEREAD   CEXPT @ READ ;
 9
10 : +EXPT   CEXPT @ DUP  LEXPT @  U< IF  READ  #Records N@
11    CEXPT +!  CEREAD  ELSE  DROP  LEXPT @ CEXPT !  THEN ;
12
13 : -EXPT   CEXPT @ DUP  1- IF  READ  Pheader N@  CEXPT +!
14    CEREAD  ELSE  1 CEXPT !  THEN ;
15

Screen:  851
 0 ( Master - Scan Header Support Functions )
 1
 2 VARIABLE CSCAN
 3
 4 : LSCAN   Available 2+ 2+ ;
 5
 6 : !LSCAN   LSCAN ! ;
 7
 8 : CSREAD   CSCAN @ READ ;
 9
10 : +SCAN   CEREAD #Scans N@  CSREAD Scan# N@  SWAP < IF  CSCAN
11    #Records N@ SWAP +!  CSREAD  THEN ;
12
13 : -SCAN   CSREAD Scan# N@ 1 > IF  CSCAN  Pheader N@ SWAP +!
14    CSREAD  THEN ;
15
```

```
Screen:  852
 0 ( Master - Data File Utilities )
 1
 2 :CASE datafiles   0FILE  1FILE  2FILE  3FILE ;
 3
 4 VARIABLE CFILE     0 CFILE !     0 datafiles   @AVAILABLE
 5
 6 : DATAFILES   !AVAILABLE  DUP CFILE !
 7    datafiles  @AVAILABLE ;
 8
 9 : FILENAME   Available 6 + ;
10
11 : #EXPTS   Available 30 + ;
12
13 ( test if cartridge drive available )
14 : -CARTRIDGE   IOPB 10 ERASE  32 IOPB 1+ C!  DCMD [ 0000 , ]
15    I/O  Disk 2+ @ 3 AND ;

Screen:  853
 0 ( Master - Data File Support Functions )
 1
 2 : #FILES   -CARTRIDGE IF  2  ELSE  4  THEN ;
 3
 4 ( assign a name to a data file )
 5 ( syntax - n NAMEFILE name )
 6 : NAMEFILE ( n)   DUP 0 #FILES WITHIN
 7    IF  DATAFILES  ELSE  DROP 10 .ERROR  THEN
 8    92 TEXT PAD FILENAME 24 MOVE  UPDATE FLUSH ;
 9
10 ( select a data file by name )
11 : FILE   92 TEXT PAD WORKING 24 MOVE  0 CFILE !
12    BEGIN  CFILE @ DATAFILES  FILENAME 24 WORKING -TEXT
13       IF  CFILE @ 3 = IF  11 .ERROR  ELSE  1 CFILE +! 0 THEN
14       ELSE  1  THEN  END ;
15

Screen:  854
 0 ( Master - File Directory )
 1 [R                        \FILE#   ----------FILE---------
 2          #EXPTS   #RECORDS   AVAILABLE]   CONSTANT 'FDIR'
 3 : FTITLE   'FDIR' 2+ 2+ ASSIGN  +L 25 SPACES
 4    ." MSU TQMS FILE DIRECTORY" +L  RPT @ HEADING ;
 5
 6 : .FHEADER   CFILE @ 3 U.R  5 SPACES  FILENAME 24 TYPE
 7    15 SPACES  #EXPTS @ 4 U.R  6 SPACES  Available @ DUP 5 U.R
 8    6 SPACES LIM @ SWAP - 5 U.R ;
 9
10 : FDIR   CFILE @  FTITLE   'FDIR' REPORT  EMPTY-BUFFERS
11    #FILES 0 DO  I DATAFILES  .FHEADER  +CR  LOOP  DATAFILES ;
12
13 : ?FILE   'FDIR' HEADING  .FHEADER  CR ;
14 : .FILE   FILENAME 24 TYPE ;
15
```

```
Screen:  855
 0 ( EXTENDED MEMORY INTERPROCESSOR ARRAY OPERATORS )
 1 CODE E>IMOVE   1 POP   W POP   0 POP   DS PUSHS   ES PUSHS
 2   SEG-REG 2+ ES LSG   iface DS LSG   1NZ IF   I 0 XCHG   W ROR
 3   W ROL   CS IF   MOVS B  1 DEC   THEN   1 SHR   REP MOVS
 4   CS IF   MOVS B   THEN   I 0 XCHG   THEN
 5   ES POPS   DS POPS   NEXT
 6 CODE E<IMOVE   1 POP   W POP   0 POP   DS PUSHS   ES PUSHS
 7   iface ES LSG   SEG-REG DS LSG   1NZ IF   I 0 XCHG W ROR
 8   W ROL   CS IF   MOVS B   1 DEC   THEN   1 SHR   REP MOVS
 9   CS IF   MOVS B   THEN   I 0 XCHG   THEN
10   ES POPS   DS POPS   NEXT
11 CODE E<MMOVE  1 POP   W POP   0 POP      ES PUSHS
12   iface ES LSG      1NZ IF   I 0 XCHG   W ROR
13   W ROL   CS IF   MOVS B   1 DEC   THEN   1 SHR   REP MOVS
14   CS IF   MOVS B   THEN I 0 XCHG   THEN
15   ES POPS      NEXT

Screen:  856
 0 ( EXTENDED MEMORY RECORD TRANSFER )
 1
 2 CODE E<DISK   ( Diskaddr,Extaddr --> )
 3        IFACE # W MOV    W ES LSG
 4        W POP   2 POP   I PUSH   2 I MOV   32 # 1 MOV
 5        REP MOVS   ZERO # W MOV   W ES LSG   I POP   NEXT
 6
 7 CODE E>DISK   ( Extaddr,Diskaddr --> )
 8        IFACE # W MOV    W DS LSG
 9        W POP   2 POP   I PUSH   2 I MOV   32 # 1 MOV
10        REP MOVS   ZERO # W MOV   W DS LSG    I POP   NEXT
11
12
13
14
15

Screen:  857
 0 ( Master - Data Transfer to Graphics Slave )
 1
 2 2LABEL DATABUFFER 2DATABUFFER
 3 2LABEL SCNHDR SCNHDR
 4
 5 : SCNHDR>SL   CSCAN @ RECORD SCNHDR 0 2 >SLAVE 64 IPMOVE ;
 6
 7 HEX A000 CONSTANT graphbuf
 8     8000 CONSTANT 2databuffer     DECIMAL
 9 : >graphics SL2 >graphics ;
10
11 : data>SLAVE   #RECORDS 2DATABUFFER 0 2 >SLAVE 12 IPMOVE
12     DATABUFFER 2databuffer  0 2 >SLAVE
13     #POINTS @ 2* 2* 2* E>IMOVE   >graphics ;
14
15
```

```
Screen:  858
 0 ( Master - Time and Date Output for Report Generator )
 1
 2 : .RDATE   (DATE) RIGHT ;
 3
 4 : .RTIME   60000 M/MOD  .TIME RIGHT DROP ;
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  859
 0 ( Master - Header Value Printing Functions )
 1
 2 : ?NUM   N@ NUM. ;
 3
 4 10 7934 4 LABELS RF/SCAN
 5
 6 : ?MASS   -2 0 WITHIN NOT ;
 7
 8 : ?M.0   N@ DUP ?MASS IF  N.1R  ELSE  ABS RF/SCAN RIGHT  THEN ;
 9
10 : ?M   N@ DUP   ?MASS IF  N.1  ELSE  ABS RF/SCAN RIGHT  THEN ;
11
12 : ?M/T   N@ DUP  ?MASS IF  N.1  ELSE  ABS RF/SCAN TYPE  THEN ;
13
14 : .TITLE   Title 2DUP  B@  DROP PAD SWAP  TYPE ;
15

Screen:  860
 0 ( Master - Experiment Header Display )
 1 [R                       \-------TITLE-------                  DA
 2 TE       TIME      #SCANS      #RECORDS]  CONSTANT 'EDIR'
 3
 4 : ETITLE   'EDIR' 2+ 2+ ASSIGN  +L 14 SPACES
 5    ." Experiment Summary - File: " .FILE  +L  RPT @ HEADING ;
 6
 7 : .EHEADER   READ  Title ?B  Date N@ .RDATE  Time D@ .RTIME
 8    #Scans ?N  #Records ?N ;
 9
10 : EDIR   CEXPT @  ETITLE  'EDIR' REPORT  1 CEXPT !
11    LEXPT @ IF  BEGIN  CEXPT @ DUP  .EHEADER +CR  +EXPT
12    LEXPT @ = END  CEXPT !  THEN ;
13
14 : ?EXPT  'EDIR' HEADING  CEXPT @ .EHEADER  CR ;
15
```

```
Screen:  861
 0 ( Master - Scan Header Display Utilities )
 1 : .SDEVICE   Sec N@ DUP  1 32 WITHIN IF  DUP !DEVICE#
 2    DNAME RIGHT  Svalue ?NUM  ELSE  DROP  2 SKIPS  THEN ;
 3
 4 : .SHEADER   READ  Scan# ?N  Stype 1@ MNAME RIGHT  Device N@ DUP
 5    !DEVICE#  DNAME RIGHT  Start ?NUM  End ?NUM  Imode 1@
 6    PMNAME RIGHT  1Mass ?M.0  3Mass ?M.0  #Points ?N
 7    Tic ?D  .SDEVICE ;
 8
 9 [R                        \SCN#  STYPE DEV  START    END   MODE
10     Q1     Q3  #PNTS        TIC SEC SVALUE]  CONSTANT 'SDIR'
11
12 : STITLE   'SDIR' 2+ 2+ ASSIGN  CEREAD  +L  5 SPACES
13    ." File: " .FILE  ." Expt: "  .TITLE
14    Time D@ 60000 M/MOD .TIME TYPE DROP
15    5 SPACES  Date N@ .DATE +L  RPT @ HEADING ;

Screen:  862
 0 ( Master - Transfer Data to/from Disk File )
 1
 2 : >FILE   #RECORDS @ ?DUP IF  0 DO  DATABUFFER I 64 * + MSSLOT
 3    RECORD E>DISK  UPDATE  LOOP  FLUSH  THEN ;
 4
 5 : >DATABUFFER   CSREAD  #Records N@ 2-
 6    0 DO  CSCAN @ I + 2+  RECORD  DATABUFFER I 64 * +
 7    E<DISK  LOOP  #Points ADDRESS #POINTS 10 MOVE ;
 8
 9 : @RXREC ( rec#)   RECORD DATABUFFER 64 MOVE ;
10 : !RXREC ( rec#)   RECORD DATABUFFER SWAP 64 MOVE  UPDATE ;
11
12 VARIABLE stype
13 VARIABLE start
14
15

Screen:  863
 0 ( Master - Scan Directory, Selection, and Headers )
 1 : SDIR   CEREAD #Scans N@ IF  STITLE 'SDIR' REPORT  CEXPT @ DUP
 2    READ  1st-scan N@ + CSCAN !  #Scans N@ 0 DO  CSCAN @
 3    .SHEADER  +CR +SCAN  LOOP  LSCAN @ CSCAN !
 4    ELSE  CR  19 B&L# .LINE  CR  THEN ;
 5
 6 : .DHEADER   ." #" Scan# N@ 4 U.R SPACE  .TITLE  SPACE
 7    Stype 1@ .MNAME SPACE  Device N@ .DNAME 2 SPACES
 8    ." M1:" 1Mass ?M/T  SPACE ." M3:" 3Mass ?M/T  SPACE
 9    ." TIC:"  Tic D@ 9 D.R SPACE  Time D@ .SECONDS ;
10
11 : SS ( n)   CEXPT @ DUP  READ  1st-scan N@ +  CSCAN !  1- #Scans
12    N@ MIN DUP 1 < IF  DROP  ELSE  0 DO  +SCAN  LOOP  THEN
13    CSREAD  Stype 1@ DUP stype !  7 < IF  >DATABUFFER
14    data>SLAVE THEN ;
15 : ?SCAN   CR  CSREAD  .DHEADER  CR ;
```

```
Screen:  864
 0 ( Master - Header for CDISP Plot )
 1
 2 : .RHEADER   ." #"  Scan# N@ 4 U.R  SPACE .TITLE  SPACE
 3    Stype 1@ .MNAME  SPACE ." Length:" End N@ U.1  SPACE
 4    ." Sec"  SPACE  Time D@ .TIME TYPE  SPACE  Date N@ .DATE ;
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  865
 0 ( FLOPPY DIRECTED EXPERIMENT BACKUP )
 1
 2  ( enter file and select an experiment --- max 4000 records )
 3
 4  : ZEROREC  (   ---> addr 5050 ) CEREAD Scan# N@ SS CSCAN @
 5             5050 BLOCK UPDATE SWAP OVER 2+ 2+ ! DUP DUP
 6             CEREAD #Records N@ SWAP ! 2+ 1 SWAP ! ;
 7
 8
 9
10  : FIRBLK  ( addr 5050 --->    )  64 + 15 0 DO DUP 64 I * +
11            CEXPT @ I + RECORD SWAP 64 MOVE LOOP
12            DROP FLUSH ;
13
14 866 LOAD
15

Screen:  866
 0  ( FLOPPY DIRECTED EXPERIMENT BACKUP CONTINUED )
 1
 2   : ADDREC  (  ---   )  #Records N@ 16 DO 5050 I
 3             16 / + BLOCK UPDATE 16 0 DO DUP
 4             64 I * + I J + RECORD SWAP 64 MOVE
 5             LOOP DROP FLUSH 16 +LOOP ;
 6
 7
 8
 9   : EXP>FLOPPY ( before executing insert floppy--       )
10                ( choose a file and select an experiment )
11                 ZEROREC FIRBLK ADDREC 7 EMIT ;
12
13   : FLOPPY>EXP ( to BACKUP-FILE starting at 4751 )
14                5050 4751 250 BLOCKS ;
15
```

```
Screen:  867
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  868
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  869
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  870
 0 ( Master - Experiment Header Create & Update Utilities )
 1
 2 : !TIME   @DATE Date N!   @TIME Time D! ;
 3
 4 : +RECORDS ( #recs)   LEXPT @ READ  #Records N+! ;
 5
 6 : EXPT-UPDATE   LEXPT @ READ  #RECORDS @ 2+ +RECORDS
 7    1 #Scans N+! ;
 8
 9 : @RECORD ( rec#)   RECORD WORKING 64 MOVE ;
10 : !RECORD ( rec#)   RECORD WORKING SWAP 64 MOVE ;
11
12 : ?RECORDS   8 /MOD  SWAP IF  1+  THEN ;
13
14
15

Screen:  871
 0 ( Master - Experiment Entry )
 1
 2 : EXPT   MSSLOT  DUP DUP 2DUP  READ  LEXPT @ SWAP -
 3    Pheader N!  CEXPT !  !LSCAN  !LEXPT  1 Rflag N!
 4    0 #Scans N!  1 #Records N!  !TIME  92 TEXT Title B!
 5    4 MSCHUNK DROP  4 +RECORDS  Available  @ 1+ DUP
 6    CEXPT @ - 1st-scan N!  CSCAN ! UPDATE
 7    1 #EXPTS +!  !AVAILABLE  FLUSH ;
 8
 9
10
11
12
13
14
15

Screen:  872
 0 ( Master - Experiment Note Support )
 1
 2 : notes   CREATE  ,  DOES>  @  CEXPT @ +  RECORD  92 TEXT
 3    PAD SWAP  64 MOVE  UPDATE ;
 4
 5 1 notes 1NOTE
 6 2 notes 2NOTE
 7 3 notes 3NOTE
 8 4 notes 4NOTE
 9
10 : ?NOTES   0  CEXPT @ 1+ DUP  4 + SWAP  DO  CR 1+ DUP .
11    I RECORD 64 >TYPE  LOOP  DROP ;
12
13
14
15
```

```
Screen:  873
 0 ( Master - Experiment Selection )
 1
 2 : SELECT   92 TEXT  PAD WORKING 16 >MOVE<  1 CEXPT !
 3    BEGIN  CEREAD  CEXPT @   1st-scan N@ +  CSCAN !
 4       WORKING Title ADDRESS  -TEXT
 5       IF  CEXPT @ LEXPT @ =
 6          IF  12 .ERROR  ELSE  +EXPT  0 THEN
 7       ELSE  1  THEN
 8    END ;
 9
10
11
12
13
14
15

Screen:  874
 0 ( Master - Disk Compress )
 1
 2 : EXCHANGE ( rec#)   RECORD DUP  PAD B/R @ MOVE
 3    WORKING SWAP B/R @ MOVE  PAD WORKING B/R @ MOVE  UPDATE ;
 4
 5 : COMPRESS ( source dest #recs)   >R SWAP R>  0
 6    DO  WORKING 64 ERASE  2DUP I + EXCHANGE I + EXCHANGE
 7    LOOP  2DROP ;
 8
 9
10
11
12
13
14
15

Screen:  875
 0 ( Master - Experiment Deletion )
 1
 2 : DELETE   SELECT  #Records N@  CEXPT @ LEXPT @ U<
 3    IF  CEXPT @  Pheader N@  +EXPT  Pheader N!  CEXPT @ DUP
 4       ROT Available @ 1+  ROT -  ( src dest #recs) COMPRESS
 5       DUP MINUS DUP  LEXPT +!  LSCAN +!
 6    ELSE  -EXPT CEREAD  CEXPT @ LEXPT !   #Scans N@ ?DUP
 7       IF  SS CSCAN @ LSCAN !  ELSE  LEXPT @ LSCAN !  THEN
 8    THEN  MINUS  Available +!  -1 #EXPTS +!  UPDATE
 9    !AVAILABLE FLUSH  1 CEXPT !  BELL ;
10
11
12
13
14
15
```

```
Screen:  876
 0 ( Master - Scan Header Create Utilities )
 1
 2 : !DEVICE-PARAMS   #DEVICE @ Device N!  'START @ Start N!
 3    'END @ End N!  'STEP @ Step N! ;
 4
 5 : NEW-SHEADER   MSSLOT DUP READ  LSCAN @  OVER - Pheader !
 6    DUP CSCAN !  !LSCAN ;
 7
 8 : !PARAMS   0 0 VDPT Imode 1@ 256 * +  MSSLOT RECORD 64 MOVE
 9    UPDATE  1 #Records N+! ;
10
11
12
13
14
15

Screen:  877
 0 ( Master - Quad Value Control for Scan Type )
 1
 2 CREATE quad-value   ( Q1    Q3 )
 3 ( 1SCAN )             -2 ,  -1 ,
 4 ( 3SCAN )             -1 ,  -2 ,
 5 ( PSCAN )             -2 ,   0 ,
 6 ( DSCAN )              0 ,  -2 ,
 7 ( NSCAN )             -2 ,   1 ,
 8 ( SWEEP )              0 ,   0 ,
 9 ( LSWEEP )             0 ,   0 ,
10 ( 1SIM )              -2 ,  -1 ,
11 ( 3SIM )              -1 ,  -2 ,
12 ( PSIM )              -2 ,   0 ,
13 ( DSIM )               0 ,  -2 ,
14 ( MRM )               -2 ,  -2 ,
15

Screen:  878
 0 ( Master - Scan Type Storage )
 1
 2 : Q13OFFSET   #DEVICE @  M3 'START @ M1 'START @ - SWAP
 3    !DEVICE# ;
 4
 5 : !SECDEV ( scan type# - scan type#)   DUP  6 =
 6    IF  ELSE  -1 Sec N!  THEN ;
 7
 8 : !SCANTYPE ( scan type#)   #DEVICE @ SWAP  !SECDEV
 9    DUP  IMODE @ 256 * +  Stype N! 2* 2*  quad-value + 2@  ?DUP
10    IF  ELSE  M1 'CURRENT @  THEN  1Mass N!  ?DUP
11    IF  DUP 0 > IF  DROP  Q13OFFSET  THEN
12    ELSE  M3 'CURRENT @  THEN  3Mass N!  !DEVICE# ;
13
14
15
```

```
Screen:  879
 0 ( Master - Write New Scan/SIM/MRM Header )
 1
 2 : >SHEADER ( scan type#)
 3    #POINTS @ ?RECORDS #RECORDS !  EXPT-UPDATE
 4    LEXPT @ @RECORD NEW-SHEADER  CSCAN @ !RECORD  !TIME
 5    !DEVICE-PARAMS  #POINTS #Points ADDRESS 10 MOVE
 6    2 Rflag N!  #RECORDS @ 1+ #Records N!  !SCANTYPE
 7    (RATE) @ Srate N!  !PARAMS  UPDATE ;
 8
 9 : >RHEADER ( scan type#)   0 #RECORDS !  EXPT-UPDATE
10    LEXPT @ @RECORD  NEW-SHEADER  CSCAN @ !RECORD  !TIME
11    !DEVICE-PARAMS  #SIMS @ #Points N!  0. 2DUP  Tic D!
12    Max-inten D!  3 Rflag N!  1 #Records N!  !SCANTYPE
13    (RATE) @ Srate N!  !PARAMS  UPDATE FLUSH ;
14
15

Screen:  880
 0 ( Master - Scan Selection )
 1
 2 : +S   +SCAN >DATABUFFER ;
 3
 4 : -S   -SCAN >DATABUFFER ;
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  881
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  882
 0 ( Master - Recover Values from Data File )
 1
 2 : GETPARAMS   CSREAD  Imode 1@ 256 * 0 0 VDPT +
 3    CSCAN @ 1+ RECORD  SWAP 64 MOVE ;
 4
 5 : SCLR   0 0 SIM-TABLE 32 ERASE  0 #SIMS ! ;
 6
 7 : RCLR   0 0 MRM-TABLE 64 ERASE  0 #SIMS ! ;
 8
 9 : @MRMS   RCLR CSREAD  #Points N@ DUP  #SIMS !  SCHK
10    CSCAN @ 2+ @RXREC  #SIMS @ 0 DO  I DUP 2DUP  @1MASS
11    0 ROT MRM-TABLE !  @3MASS 2 ROT MRM-TABLE !  LOOP  REDLEAVE ;
12
13
14
15

Screen:  883
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  884
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  885
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  886
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  887
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  888
 0 ( Master - Add Buffer Access )
 1 ASSEMBLER  HEX ZERO 1000 + CONSTANT ABUF-SEG  DECIMAL
 2
 3 CODE A2@   W POP   W SHL   W SHL   0 DS SSG   ABUF-SEG # 2 MOV
 4    2 DS LSG   2 W) PUSH   W ) PUSH   0 DS LSG   NEXT
 5
 6 CODE A2!   W POP   W SHL   W SHL   0 DS SSG   ABUF-SEG # 2 MOV
 7    2 DS LSG   W ) POP   2 W) POP   0 DS LSG   NEXT
 8
 9 CODE A2+!   W POP   W SHL   W SHL   0 POP   2 POP   DS PUSHS
10    ABUF-SEG # 1 MOV   1 DS LSG   2 2 W) ADD   0 W ) ADC
11    DS POPS   NEXT
12
13 CODE AERASE   ABUF-SEG # 0 MOV   0 DS LSG   0 ES LSG   0 0 SUB
14    0 W MOV   4096 # 1 MOV   BEGIN   STOS   LOOP   ZERO # 0 MOV
15    0 DS LSG   0 ES LSG   NEXT

Screen:  889
 0 ( Master - Spectrum Adding Utilities )
 1
 2 : ?MAX-INTEN ( d)   2DUP TIC 2@ D+ TIC 2!   MAX-INTEN 2@ DMAX
 3    MAX-INTEN 2! ;
 4
 5 VARIABLE SAME 4 ALLOT
 6
 7 : ?SAME   SAME @ SAME 2+ 2@ + + 0= IF  Stype N@ SAME !
 8    1Mass D@ SAME 2+ 2!  1 ELSE  Stype N@ SAME @ =  1Mass D@
 9    SAME 2+ 2@  ROT = ROT ROT = AND AND  THEN ;
10
11
12
13
14
15

Screen:  890
 0 ( Master - Add Buffer Transfers )
 1 VARIABLE AXMAX
 2
 3 : NOMINAL ( nominal mass)   5 +  10 / ;
 4
 5 : >ABUF ( move current scan to add buffer)   AERASE
 6    #POINTS @ 0 DO  I @Y  I @X NOMINAL  A2+!  LOOP ;
 7
 8 : (ABUF>)   0 #POINTS !  0. MAX-INTEN 2!  0. TIC 2!
 9    2048 0 DO  I A2@ D0= NOT IF  #POINTS @ DUP I 10 * SWAP 'X !
10    I A2@ 2DUP ?MAX-INTEN  ROT 'Y 2!  1 #POINTS +!
11    I 10 * AXMAX !  THEN  LOOP ;
12
13 : ABUF>   DATABUFFER 8192 ERASE  (ABUF>)  CSREAD
14    Device N@ !DEVICE#  'START @  0 !START  'END @ AXMAX @ !END
15    Stype N@ >SHEADER >FILE  !END !START ;
```

```
Screen:  891
 0 ( Master - Add Buffer Transfers )
 1
 2 : (ADD) ( add current scan to add buffer)
 3    #POINTS @ 0 DO  I @Y  I @X NOMINAL  A2+!  LOOP ;
 4
 5 : ADD   SS >ABUF  SS (ADD)  ABUF> ;
 6
 7 : SUB   SWAP  SS >ABUF  SS
 8    #POINTS @ 0 DO  I @Y DMINUS  I @X NOMINAL  DUP >R  A2+!
 9    I A2@ 0. D< IF  0. I A2!  THEN  R> DROP  LOOP  ABUF> ;
10
11 : SUM ( start end)   1+ SWAP  AERASE  SAME 6 ERASE
12    DO  I SS ?SAME IF  (ADD)  THEN  LOOP  ABUF> ;
13
14
15

Screen:  892
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  893
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 894
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 895
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 896
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen:  897
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  898
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  899
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  900
 0 ( Master - Normalization)
 1 : NORMALIZE  ( d - n*1000 )   MAX-INTEN 2@  2DUP OR 0= NOT
 2   IF 2SWAP 2>N 2>N F/  1000 >N  F* N>
 3   ELSE 2DROP 2DROP 0 THEN ;
 4 EXIT
 5 1000 CONSTANT 1000.0
 6 HEX
 7 CODE NORMALIZE ( d - n*1000)   S W MOV
 8    9B C, DB C, 05 C,                                        ( 2>N)
 9    0 POP   0 POP   9B C, DB C, 06 C, MAX-INTEN ,      ( Max >N)
10    9B C, DE C, F9 C,                                        ( F/ )
11    9B C, DF C, 06 C, ' 1000.0 ,                     ( 1000.0 >N)
12    9B C, DE C, C9 C,                                        ( F* )
13    0 PUSH   S W MOV   9B C, DF C, 1D C, 9B C,              ( N>)
14 NEXT
15 DECIMAL

Screen:  901
 0 ( Master - Scan Data Reporting Utilities )
 1
 2 [R   \  #      X FLAGS WIDTH        Y  NORM    #      X FLAGS WI
 3 DTH         Y  NORM] CONSTANT 'DLIST'
 4
 5 : .POINT ( point #)   DUP .N  DUP @X NUM.  DUP @FLAGS .N
 6    DUP @WIDTH .N  @Y 2DUP .D  NORMALIZE  N.1R ;
 7
 8 : P/P ( - points/page)   L/P C@ 7 - ;
 9
10 : DLIST-TITLE   'DLIST' 2+ 2+ ASSIGN  +L  5 SPACES  ." FILE: "
11    .FILE 10 SPACES ." EXPT: " .TITLE  +L +L ." #" Scan# N@
12    4 U.R SPACE Stype 1@ .MNAME  SPACE Device N@ .DNAME  SPACE
13    ."  M1: " 1Mass ?M/T  ."  M3: " 3Mass ?M/T  3 SPACES
14    ." TIC:" Tic D@ 9 D.R  ."  MAX: " Max-inten D@ 9 D.R  SPACE
15    Time D@ .SECONDS  +L RPT @ HEADING ;

Screen:  902
 0 ( Master - Scan Data Reporting )
 1
 2 : dlist   CSREAD  #Points N@ 0=
 3    IF  17 B&L# .LINE
 4    ELSE  DLIST-TITLE  'DLIST'  REPORT  #POINTS @ 1+ 2/ P/P /MOD
 5       ( type as many full pages as needed)
 6       DUP  ?DUP  IF  0
 7       DO  P/P 0 DO  I' J * 2* I + DUP .POINT  P/P + .POINT +CR
 8                LOOP
 9       LOOP  THEN  P/P * 2* 2DUP + SWAP  OVER 0 >
10       ( type a partial page if needed)
11       IF  DO  I .POINT  DUP I +  #POINTS @ OVER >
12              IF  .POINT  ELSE  DROP  THEN  +CR
13           LOOP  DROP  THEN  THEN ;
14
15
```

```
Screen:  903
 0 ( Master - MRM/SIM Data Report Heading )
 1 : ?MASS/RF   DUP 0< IF  DROP  ."  RF "
 2    ELSE  5 + 10 / 4 U.R  THEN ;
 3 : .RXN ( rxn#)   DUP @1MASS ?MASS/RF ." ->"  @3MASS ?MASS/RF ;
 4
 5 : .CPAGE   PAGE  0 L# C!  1 P# +!
 6 ." MSU Chemistry Department                                  Page "
 7    P# ?  2 SPACES  TIME  2 SPACES  TODAY @ .DATE  +L +L
 8    5 SPACES ." FILE: " .FILE  5 SPACES ." EXPT: " .TITLE  +L
 9    ." # " Scan# N@ 4 U.R  SPACE stype @ .MNAME  3 SPACES
10    ." Length:  "  End N@ U.1 ." Seconds"  4 SPACES  Time D@
11    .SECONDS  3 SPACES  Date N@ .DATE  ."   Time  "
12    CSCAN @ 2+ @RXREC
13    #SIMS @ 0 DO  I .RXN  LOOP  +L  2 SPACES  ." ----"  2 SPACES
14    #SIMS @ 0 DO  ."  --------"  LOOP  +L ;
15

Screen:  904
 0 ( Master - MRM/SIM Data Reporting )
 1
 2 : ?CPAGE   L# C@ + L/P C@  > IF  L/P C@ 60 < IF  KEY  0= IF
 3    QUIT  THEN  THEN  .CPAGE  THEN ;
 4 : CR+   +L  1 ?CPAGE ;
 5
 6 : .RXDATA   DUP 1-  STEPTIM U* DROP  U.1  SPACE  CSCAN @ +
 7    @RXREC  #SIMS @ 0 DO  I @Y 9 D.R  LOOP  CR+ ;
 8
 9 : RLIST   CSREAD  @MRMS  0 P# !  .CPAGE  #Points N@ #SIMS !
10    #Records N@  DUP 2 > IF  2 DO  I .RXDATA  LOOP  ELSE  DROP
11    17 B&L# .LINE  THEN ;
12
13
14
15

Screen:  905
 0 ( Master - Data Listing )
 1
 2 : DLIST   CSREAD  Device N@ !DEVICE#
 3    Stype 1@ 7 < IF  dlist  ELSE  RLIST  THEN ;
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  906
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  907
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  908
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  909
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  910
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  911
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  912
 0 ( Graphics - Y-Axis Scaling )
 1
 2 VARIABLE YMODE
 3 VARIABLE YSTART  2 ALLOT   VARIABLE YEND
 4
 5 : LIN  0  250 YSTART 2!  1000 YEND !  0 YMODE ! ;  COMMAND
 6 : LOG  0 1000 YSTART 2!  3000 YEND !  1 YMODE ! ;  COMMAND
 7
 8 : MSXTAG  10 / (.) ;
 9 : LOGTAG  100 /  0 <#  # 46 HOLD #  #> ;
10
11 : MSYTAG  YMODE @ IF ['] LOGTAG  ELSE  ['] MSXTAG  THEN
12    GLMODE ! ;
13
14 : MRMTAG   10 U/ 0 <# #S #> ;
15

Screen:  913
 0 ( Graphics - Scan Buffer Header and Access )
 1
 2 VARIABLE SCNHDR 62 ALLOT  COMMAND
 3
 4 FORTH : 'scnhdr   CREATE  ,  ;CODE   W INC   W INC
 5    SCNHDR # 0 MOV   W ) 0 ADD   0 PUSH   NEXT
 6
 7  2 'scnhdr Title       18 'scnhdr Date      20 'scnhdr Time
 8 24 'scnhdr #Records    28 'scnhdr Scan#     30 'scnhdr Stype
 9 32 'scnhdr Device      34 'scnhdr Start     36 'scnhdr End
10 38 'scnhdr Step        40 'scnhdr 1Mass     42 'scnhdr 3Mass
11 48 'scnhdr #Points     50 'scnhdr Tic       54 'scnhdr Max-inten
12
13
14
15

Screen:  914
 0 ( Graphics - Graphics Buffer Definition and Access )     HEX
 1
 2 A000 CONSTANT graphbuf
 3
 4 FORTH : 'gbuf   CREATE  ,  ;CODE   W INC   W INC   0 POP   0 SHL
 5    0 SHL   0 SHL   W ) 0 ADD   graphbuf # 0 ADD   0 PUSH   NEXT
 6
 7 0 'gbuf 'GX                0 'gbuf 'G1MASS
 8 2 'gbuf 'G3MASS            4 'gbuf 'GY
 9
10 : @GX   'GX @ ;             : @G1MASS   'G1MASS @ ;
11 : @G3MASS   'G3MASS @ ;     : @GY   'GY 2@ ;
12
13 DECIMAL
14
15
```

```
Screen:  915
 0 ( Graphics - Parameters and Variables for Graphics )
 1
 2 VARIABLE DSTART        COMMAND
 3 VARIABLE D#FIELDS      COMMAND
 4 VARIABLE DUNITS/FIELD  COMMAND
 5 VARIABLE DNORM         COMMAND
 6
 7 VARIABLE dstart  COMMAND
 8 VARIABLE d#fields COMMAND
 9 VARIABLE dunits/field COMMAND
10
11 VARIABLE TSTART  COMMAND
12 VARIABLE TEND    COMMAND
13
14 5 CARRAY sellist  COMMAND
15

Screen:  916
 0 ( Graphics - Display Normalization Utilities )
 1
 2 8 2ARRAY MAX'S   COMMAND
 3
 4 : CLRMAX'S   0 MAX'S 32 ERASE ;
 5
 6 : ?FIELD ( x - field)   -1  5 0 DO  OVER  I FIELD LLX @ ULX @ 1+
 7    WITHIN IF  DROP I  LEAVE  THEN  LOOP  SWAP DROP ;
 8
 9 : MSMAXSET   CLRMAX'S  #Points @ 0 DO  I @GX ?FIELD DUP
10    0< IF  DROP 2DROP  ELSE  MAX'S DUP 2@ I @GY DMAX  ROT 2! THEN
11    LOOP ;
12
13
14
15

Screen:  917
 0 ( Graphics - Rounding Functions )
 1
 2 ( round up )
 3 : +ROUND ( n factor - n)  DUP ROT 1- SWAP / 1+ * ;
 4
 5 ( round down )
 6 : -ROUND ( n factor - n)   DUP ROT SWAP / * ;
 7
 8
 9 VARIABLE MICHAEL COMMAND
10 : 1KRISTO  Start @ dstart ! ; COMMAND
11 : 2KRISTO  Start @  End @  - MICHAEL ! ; COMMAND
12
13
14
15
```

```
Screen:  918
 0 ( Graphics - SIM/MRM Plotting Utilities )
 1
 2 : @SIM# ( sel# - sim#)   sellist C@ ;
 3
 4 : ?MASS/RF   DUP 0< IF  DROP  ."  RF "
 5    ELSE  5 + 10 / 4 U.R  THEN ;
 6
 7 : .RXN ( rxn#)   DUP @G1MASS ?MASS/RF ." ->"  @G3MASS ?MASS/RF ;
 8
 9
10
11
12
13
14
15

Screen:  919
 0 ( Graphics - Set up Tic Mark Intervals )
 1
 2 VARIABLE ticint          VARIABLE ticstart
 3
 4 : TICINT   DLX @    10 U/          10 ticint !
 5            DUP      20 > IF        20 ticint !  THEN
 6            DUP      40 > IF        50 ticint !  THEN
 7            DUP     100 > IF       100 ticint !  THEN
 8            DUP     200 > IF       200 ticint !  THEN
 9            DUP     400 > IF       500 ticint !  THEN
10            DUP    1000 > IF      1000 ticint !  THEN
11                   2000 > IF      2000 ticint !  THEN
12            LLX @  DUP 0= IF  ticstart !  ELSE
13                            ticint @  +ROUND ticstart !  THEN ;
14
15

Screen:  920
 0 ( Graphics - Draw Fields )
 1
 2 : ?#FIELDS ( delta amu - amu/field #fields )
 3    DUP 3 /  500 +ROUND  DUP 2000 > IF  DROP 2000  THEN
 4    DUP ROT SWAP  /MOD  SWAP  IF  1+  THEN ;
 5
 6 : MSAXES   d#fields @ 0 DO  I FIELD FRAME  TICINT
 7    ticstart @ ticint @ 2DUP XTICS  ['] MSXTAG GLMODE !  2* XTAGS
 8    YSTART 2@ 2DUP  YTICS  MSYTAG YTAGS  LOOP ;
 9
10 : CAXES   TICINT  ticstart @ ticint @  d#fields @ 0
11    DO  I FIELD FRAME  2DUP UXTICS  YSTART 2@ 2DUP YTICS
12       MSYTAG YTAGS  LOOP  0 FIELD ['] MRMTAG GLMODE !
13         2* UXTAG ;
14
15 : 3KRISTO Start @ End @ - ?#FIELDS d#fields ! ; COMMAND
```

```
Screen:  921
 0 ( Graphics - Draw Sweep Axes )
 1
 2 : rwaxes   ( start end)
 3    YSTART 2+ @ YEND @ 2SWAP  SCLSET
 4    TICINT  ticstart @ ticint @ 2DUP XTICS  ['] MSXTAG GLMODE !
 5    2* XTAGS  YSTART 2@ 2DUP YTICS  MSYTAG YTAGS  GLEFT
 6    20 880 (CUR)   Max-inten 2@ D. ;
 7
 8 : RWAXES ( start end)   0 FIELD  40 740 80 900 WINDOW
 9    FRAME  rwaxes ;
10
11
12
13
14
15

Screen:  922
 0 ( Graphics - Define Field Parameters )
 1
 2 VARIABLE PPNTS/FIELD
 3
 4 : CFIELDS ( #select )   5 MIN  DUP DUP DUP  d#fields !
 5    50 * 730 SWAP -  SWAP /  PPNTS/FIELD !  740 SWAP 0
 6    DO  I FIELD  DUP PPNTS/FIELD @ - DUP  ROT 60 930 WINDOW
 7       20 - YSTART 2+ @ YEND @ TSTART @ TEND @  SCLSET
 8    LOOP  DROP ;
 9
10 : MSFIELDS ( amu/field #fields)   2DUP d#fields ! dunits/field !
11    740 OVER 50 * - OVER / SWAP 0 DO  I FIELD
12    DUP DUP  50 + I *  740 SWAP - DUP ROT - SWAP 60 930 WINDOW
13    OVER YSTART 2+ @ YEND @ ROT DUP I * dstart @ + DUP ROT +
14    SWAP  12 -  SWAP SCLSET  LOOP  2DROP ;
15

Screen:  923
 0 ( Graphics - Label the Fields )
 1
 2 : CLABELS   d#fields @ 0
 3    DO  I FIELD  LPY @ 20 - UPX @ 20 - (CUR)  I @SIM# MAX'S 2@
 4       D.  LPY @ 5 + LPX @ 5 + (CUR)  I @SIM# .RXN  LOOP ;
 5
 6 : MAXFILL   Max-inten 2@  d#fields @ 0 DO  2DUP I MAX'S 2!
 7    LOOP  2DROP ;
 8
 9 : ?MAX'S   DNORM @ IF  MSMAXSET  ELSE  MAXFILL  THEN ;
10
11 : MSLABELS   ?MAX'S  d#fields @ 0
12   DO I FIELD  LPY @ 20 - UPX @ 20 - (CUR)  I MAX'S 2@ D. LOOP ;
13
14
15
```

```
Screen:  924
 0 ( Graphics - DISP Displays)
 1
 2 : CSCREEN   1 CSIZE CFIELDS  CAXES CLABELS ;
 3
 4 : MSSCREEN   End @  Start @ -  ( to give delta amu needed )
 5    1 CSIZE DSTART @ ?DUP
 6    IF  dstart !  ELSE  Start @ dstart !  THEN  D#FIELDS @
 7    IF  DROP DUNITS/FIELD @ D#FIELDS @
 8    ELSE  ?#FIELDS  THEN  MSFIELDS  MSAXES  MSLABELS ;
 9
10 : RWSCREEN  1 CSIZE  Start @  DUP dstart !  End @
11    2DUP  - dunits/field !  RWAXES ;
12
13
14
15

Screen:  925
 0 ( Graphics - Logarithms)
 1 1000 CONSTANT 1000.0
 2
 3 HEX
 4 CODE 10LOG ( n - 1000*logn)   9B C, D9 C, EC C,     ( FLG2LD)
 5    S W MOV
 6    9B C, DF C, 05 C,   0 POP                        ( 2>N )
 7    9B C, D9 C, F1 C,                                ( FYL2X )
 8    9B C, DF C, 06 C,   ' 1000.0 ,                   ( 1000.0>N )
 9    9B C, DE C, C9 C,                                ( F* )
10    0 PUSH   S W MOV   9B C, DF C, 1D C, 9B C,       ( N> )
11 NEXT
12 DECIMAL
13
14 : ?LOG   YMODE @ IF  1 MAX 10LOG  THEN ;
15

Screen:  926
 0 ( Graphics - Normalization)
 1
 2 HEX
 3
 4 CODE NORMALIZE ( d - n*1000)   S W MOV
 5    9B C, DB C, 05 C,                                    ( 2>N)
 6    0 POP   0 POP   9B C, DB C, 06 C, Max-inten ,      ( Max >N)
 7    9B C, DE C, F9 C,                                    ( F/ )
 8    9B C, DF C, 06 C, ' 1000.0 ,                     ( 1000.0 >N)
 9    9B C, DE C, C9 C,                                    ( F* )
10    0 PUSH   S W MOV   9B C, DF C, 1D C, 9B C,            ( N>)
11 NEXT
12
13 DECIMAL
14 : NORMALIZE   1000 T*  Max-inten 2@ T/D ;
15
```

```
Screen:  927
 0 ( Graphics - Mass Spectral Plotting )
 1
 2 : >MAX-INTEN ( n)   MAX'S 2@ Max-inten 2! ;
 3
 4 : MSPLOT ( y x)   DUP dstart @ - dunits/field @ /
 5    DUP >MAX-INTEN  FIELD  ROT ROT NORMALIZE ?LOG SWAP
 6    PLOT ; COMMAND
 7
 8
 9
10
11
12
13
14
15

Screen:  928
 0 ( Graphics - Data Plotting )
 1
 2 : CPLOT ( x y)    NORMALIZE ?LOG  SWAP UPLOT ;  COMMAND
 3
 4 : DPLOT    LPY @ LPX @ (CUR)  VECTOR  #Points @ 0
 5    DO  I @GY  NORMALIZE ?LOG  I @GX PLOT  LOOP ; COMMAND
 6
 7 : MPLOT    HIST  LPY @ LPX @ (CUR)   #Points @ 0
 8    DO  I DUP  @GY ROT @GX MSPLOT  LOOP ;  COMMAND
 9
10
11
12
13
14
15

Screen:  929
 0 ( Primitives for Display Headers - MJK 10/15/85 )
 1
 2  : 60BASE  #  6 BASE !  #  DECIMAL  58 HOLD ;
 3
 4 : .SECONDS   1 1000 M*/  <# 60BASE  60BASE # # #>  TYPE ;
 5
 6  FORTH : [A ( compiles ASCII text - no more than 72 chars. )
 7        93 ( ] ) STRING ;
 8
 9    7950 LOAD
10
11 : (N.) ( n -> addr length) DUP ABS 0 <# # 46 HOLD #S SIGN #> ;
12 : N.1  ( n -> )  (N.)  6 OVER - SPACES TYPE ;
13
14
15
```

```
Screen:  930
 0 ( Labels for Devices and Modes - MJK 10/15/85 )
 1 FORTH : LABELS ( line# size )   CREATE 1+ C, DUP 1+ SWAP
 2    64 * + ,  DOES> DUP C@  ROT OVER *  ROT 1+ @  ASCII-text + +
 3    SWAP  1- ;
 4
 5   0 3 LABELS DNAME        ( Device Names )
 6   2 6 LABELS MNAME        ( Mode Names )
 7   4 3 LABELS PMNAME       ( Parameter Mode Names )
 8   3 4 LABELS RF/SCAN      ( RF/SCAN Name )
 9
10  : >TYPE  >R  PAD I MOVE  PAD R>  TYPE ;
11
12 : .DNAME   ( dev# -> )  DNAME  >TYPE ;
13 : .MNAME   ( dev# -> )  MNAME  >TYPE ;
14 : .PMNAME   ( dev# -> )  PMNAME  >TYPE ;
15

Screen:  931
 0 ( Header Value Printing Functions - MJK 10/15/85 )
 1 : ?MASS  -2 0 WITHIN NOT ;
 2 : ?M/T   @  DUP  ?MASS IF  N.1  ELSE  ABS  RF/SCAN TYPE THEN ;
 3
 4 : .S  DROP PAD SWAP TYPE ;
 5
 6 CODE >MOVE<   ( S D # )  I 2 MOV   1 POP   W POP   I POP
 7   1 SHR   BEGIN LODS   0 0 HI XCHG B   STOS   LOOP   2 I MOV
 8   NEXT
 9
10 : S@  PAD  ROT  >MOVE< ;
11
12 : ?B  2DUP  S@  PAD  C@  IF  .S  ELSE 2DROP  16 SPACES THEN ;
13
14 : .TITLE 16 Title ?B ;
15

Screen:  932
 0 ( Displays with Headers - MJK 10/15/85 )
 1
 2 : .DHEADER   0 200 (CUR) ." #" Scan# @ 4 U.R SPACE  .TITLE
 3   SPACE Stype  C@ .MNAME SPACE  Device @  .DNAME 15 100 (CUR)
 4   ."  M1:" 1Mass  ?M/T  ." M3:" 3Mass  ?M/T  ."  TIC:" Tic 2@
 5   9 D.R SPACE  Time 2@ .SECONDS ;
 6
 7 VARIABLE ?HEADER COMMAND
 8 : .DHEADER ?HEADER @ 0= IF .DHEADER THEN ;
 9
10  : MSSCREEN  MSSCREEN  .DHEADER ; COMMAND
11  : RWSCREEN  MSSCREEN  .DHEADER ; COMMAND
12  : CSCREEN  CSCREEN  .DHEADER ; COMMAND
13
14
15
```

```
Screen:  933
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  934
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  935
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 936
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 937
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 938
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen:  939
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  940
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  941
 0 ( Master - Mass Spec Graphics Variables and Labels )
 1
 2 2LABEL  DSTART        DSTART
 3 2LABEL  D#FIELDS      D#FIELDS
 4 2LABEL  DUNITS/FIELD  DUNITS/FIELD
 5 2LABEL  DNORM         DNORM
 6
 7 :CASE GSET   DSTART  DUNITS/FIELD  D#FIELDS  DNORM ;
 8
 9 6 7943 20 LABELS GNAME   ( graphics display attributes )
10
11 : .GNAME ( n)   GNAME >TYPE ;
12
13 1 #SLAVE
14 0 D#FIELDS I! 0 DUNITS/FIELD I! 0 DNORM I! 0 DSTART I!
15
```

```
Screen:  942
 0 ( Master - Graphics Parameter Setting )
 1
 2 : dstat   IF  ." Preset"  ELSE  ."  Auto "  THEN ;
 3
 4 : .DSET   PAGE  7943 LOAD  2 0 DO  CR I .GNAME  I GSET I@ U.1
 5    3 SPACES  I GSET I@ dstat  LOOP CR  2 .GNAME  D#FIELDS I@
 6    DUP N7  3 SPACES  dstat CR  3 .GNAME  DNORM I@ DUP N7
 7    7944 LOAD ;
 8
 9 : DSET   .DSET  4 0 DO  4 I + 42 CURSOR  63 EMIT ( ?)
10    8 EMIT ( BS)  1#INPUT DUP -1 = NOT IF  I GSET I!
11    ELSE  DROP SPACE I EMIT SPACE  THEN  LOOP  .DSET ;
12
13
14
15

Screen:  943
 0 ( Master - Rounding Functions )
 1
 2 ( round up )
 3 : +ROUND ( n factor - n)   SWAP 1- SWAP DUP */ ;
 4
 5 ( round down )
 6 : -ROUND ( n factor - n)   DUP */ ;
 7
 8
 9
10
11
12
13
14
15

Screen:  944
 0 ( Master - SIM/MRM Display Normalization )
 1
 2 2LABEL MAX'S MAX'S
 3 8 2ARRAY SMAXIMA
 4
 5 : CLRSMAX    0 SMAXIMA 32 ERASE ;
 6
 7 VARIABLE PSTART
 8 VARIABLE PEND
 9
10 : SMAXSET   CLRSMAX  PEND @ 1+ PSTART @ 1 MAX
11    DO  CSCAN @ 1+ I + @RXREC
12       #SIMS @ 0 DO  I SMAXIMA DUP 2@  I @Y DMAX  ROT 2!  LOOP
13    LOOP  0 2 >SLAVE  0 SMAXIMA MAX'S 32 IPMOVE ;
14
15
```

```
Screen:  945
 0 ( Master - CDISP Parameter Display )
 1
 2 VARIABLE #SELECT            0 #SELECT !
 3 CREATE SELTABLE 8 ALLOT     SELTABLE 8 ERASE
 4
 5 : #chosen ( - n)   0  #SIMS @ 0 DO  I SELTABLE + C@
 6    IF  1+  THEN  LOOP  DUP #SELECT ! ;
 7
 8 : chosenchk   #chosen 0 6 WITHIN NOT IF  18 .WARN  THEN ;
 9
10 : .RXTABLE   CSCAN @ 2+ @RXREC  #SIMS @ 0 DO  I DUP  5 SPACES
11    1+ .  8 SPACES  I .RXN  10 SPACES I SELTABLE + C@
12    IF  ." yes"  ELSE  ." no"  THEN  CR  LOOP  chosenchk  CR ;
13
14
15

Screen:  946
 0 ( Master - Display CDISP Parameter Settings )
 1
 2 2LABEL TSTART TSTART     0 TSTART I!
 3 2LABEL TEND   TEND       0 TEND I!
 4
 5 6 7945 10 LABELS CNAME
 6 : .CNAME ( n)   CNAME >TYPE ;
 7
 8 :CASE CVAR   TSTART  TEND ;
 9
10 : .CSET   PAGE 7945 LOAD  .RXTABLE  CR CR  2 0
11    DO  CR 5 SPACES I .CNAME  I CVAR I@ 6 SPACES U.1  LOOP  CR ;
12
13
14
15

Screen:  947
 0 ( Master - CDISP Parameter Setting - Utilities )
 1
 2 : ?SELECT   KEY ?DUP IF  223 AND 89 = ( Y or y)
 3    IF  I' 1+ I' SELTABLE + C!  ELSE  0 I' SELTABLE C!  THEN
 4    THEN ;
 5
 6 : ?#SELECT   #chosen 5 > IF  2 SPACES  18 .WARN
 7    6 I' + 49 CURSOR  ?SELECT  THEN ;
 8
 9 : TIMCHK   End N@ DUP  TSTART @ UMIN TSTART !  TEND @ DUP 0=
10    IF  DROP -1 THEN  UMIN DUP TEND !  TSTART @ U<
11    IF  20 .ERROR  THEN ;
12
13
14
15
```

```
Screen:  948
 0 ( Master - CDISP Parameter Selection )
 1 2LABEL sellist sellist
 2 5 CARRAY SELLIST
 3 : @SIM# ( sel# - sim# )   SELLIST C@ ;
 4
 5 : >SELLIST   0  #SIMS @ 0 DO  I SELTABLE + C@ ?DUP
 6    IF  1- OVER SELLIST C!  1+  THEN  LOOP  #SELECT !
 7    0 2 >SLAVE  SELLIST sellist 5 IPMOVE ;
 8
 9 : TPARAM ( n)   DUP 9  + #SIMS @ + 35 CURSOR  63 EMIT ( ?)
10    8 EMIT ( BS)  1#INPUT DUP  -1 = NOT IF  SWAP CVAR I!  ELSE
11    DROP SPACE 8 EMIT SPACE  THEN ;
12
13 : CSET   .CSET  #SIMS @ 0 DO  6 I + 49 CURSOR  63 EMIT 8 EMIT
14    ?SELECT ?#SELECT  LOOP  0 TPARAM  1 TPARAM  TIMCHK
15    >SELLIST  CR ;

Screen:  949
 0 ( Master - CDISP Utilities )
 1
 2 VARIABLE DTIME
 3
 4 : T>PTS ( time*10 - #pnts)   STEPTIM U/ ;
 5
 6 : CHXSET   Rate N@ 8 - RRATE !  TEND I@ DUP  T>PTS PEND !
 7    TSTART I@ DUP  T>PTS PSTART !  - DTIME ! ;
 8
 9
10 : #CHOSEN   #chosen DUP  1 6 WITHIN NOT IF  18 .ERROR  THEN ;
11
12
13
14
15

Screen:  950
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 951
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 952
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 953
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen:  954
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  955
 0 ( Master - Commands to Graphics Slave )
 1
 2 : CSCREEN   SL2 CSCREEN ;
 3
 4 : MSSCREEN   SL2 MSSCREEN ;
 5
 6 : RWSCREEN   SL2 RWSCREEN ;
 7
 8 : MPLOT   SL2 MPLOT ;
 9
10 : DPLOT   SL2 DPLOT ;
11
12
13
14
15

Screen:  956
 0 ( Master - Basic Graphics Commands )
 1
 2 : FIELD ( n)   DUP  0 5 WITHIN IF  2PUSH  SL2 FIELD
 3    ELSE  DROP  THEN ;
 4
 5 : CLEAR SL2 CLEAR ;
 6 : GDEMO SL2 GDEMO ;
 7
 8 : LIN   SL2 LIN ;
 9 : LOG   SL2 LOG ;
10
11 2LABEL ?HEADER ?HEADER
12 : HEADER   2 #SLAVE 0 ?HEADER I! ;
13 : NO-HEADER   2 #SLAVE -1 ?HEADER I! ;
14
15
```

```
Screen:  957
 0 ( Master - SIM/MRM Plotting Routine)
 1
 2 : CPLOT   PEND @ 1+ PSTART @  #SELECT @ 0
 3    DO  I FIELD  2DUP
 4       DO  CSCAN @ 2+ I + @RXREC  J @SIM# @Y  I STEPTIM * 2PUSH
 5          SWAP 2PUSH 2PUSH  SL2 CPLOT  LOOP  LOOP  2DROP ;
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  958
 0 ( Master - Data Type Specific Display Functions )
 1
 2 : CDISP   #CHOSEN  CSREAD #Records N@ 2 >
 3    IF  SCNHDR>SL  SMAXSET  #SELECT @ CSCREEN  CPLOT
 4    ELSE  17 .ERROR  THEN ;
 5
 6 : MDISP   #Points N@ 0 > IF  SCNHDR>SL  MSSCREEN  MPLOT
 7    ELSE  17 .ERROR  THEN ;
 8
 9 : RDISP   #Points N@ 0 > IF  SCNHDR>SL  RWSCREEN  DPLOT
10    ELSE  17 .ERROR  THEN ;
11
12
13
14
15

Screen:  959
 0 ( Master - Display Functions )
 1
 2 : DISP   CLEAR stype @ DUP 5 = IF  DROP  RDISP  ELSE  DUP
 3    0 5 WITHIN IF  DROP  MDISP  ELSE
 4    7 12 WITHIN IF  CDISP  THEN  THEN  THEN ;
 5
 6 : NEXT   DTIME @ DUP  TSTART +!  TEND +!  TIMCHK  CDISP ;
 7
 8 : OPLOT   stype @ DUP 5 = IF  DROP DPLOT  ELSE  DUP
 9    0 5 WITHIN IF  DROP MPLOT  ELSE
10    7 12 WITHIN IF  CPLOT  THEN  THEN  THEN ;
11
12
13
14
15
```

```
Screen:  960
 0 ( Master - Miscellaneous Functions Load Block )
 1
 2 CR { Loading Miscellaneous Functions }
 3
 4 ( 966      LOAD     ( Manifold Control )
 5  972  973 THRU     ( Upload to PDP-11 )
 6  984  988 THRU     ( Methods )
 7  990  993 THRU     ( Sequences )
 8 ( 1011 1013 THRU     ( Disk Utilities )
 9   966  968 THRU       ( Manifold Control)
10
11
12
13
14
15

Screen:  961
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  962
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  963
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  964
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  965
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  966
 0 ( Master - Manifold Control )       HEX
 1 F9D8 CONSTANT 'MANIFOLD
 2  80 'MANIFOLD 3 + C! ( initialize PIO )
 3 VARIABLE MANIFOLD    -1 MANIFOLD !  -1 'MANIFOLD !
 4 : GCHECK 0 8 0 DO I MANIFOLD ?BIT IF 1+ THEN LOOP
 5          7 < IF 3 ELSE 0 THEN
 6         0 8 0 DO I MANIFOLD 1+ ?BIT IF 1+ THEN LOOP
 7         7 < IF 3 ELSE 0 THEN + ;
 8 : MSTORE  DUP  MANIFOLD !  GCHECK  2 >
 9          IF  0F .WARN  THEN  100 /MOD SWAP
10        'MANIFOLD C!  'MANIFOLD 1+ C! ;
11
12 : COLLISION   100 * ;
13 : ON  FFFF SWAP - MANIFOLD @  AND MSTORE ;
14 : OFF  MANIFOLD @ OR  MSTORE ;
15 : GAS-OFF   FFFF  OFF ;

Screen:  967
 0 ( Master Manifold Control )
 1 ( 01CONSTANT )
 2    02 CONSTANT AGAS       02 CONSTANT N2O
 3   04 CONSTANT BGAS
 4   08 CONSTANT CGAS
 5   ( 10 CONSTANT )
 6   20 CONSTANT CIROUGH
 7   ( 40 CONSTANT )
 8   ( 80 CONSTANT )
 9   ( 100 CONSTANT )
10   ( 200 CONSTANT )
11   400 CONSTANT FGAS    400 CONSTANT SF6
12    800 CONSTANT DGAS
13   1000 CONSTANT EGAS    1000 CONSTANT ARGON
14   2000 CONSTANT COLROUGH
15

Screen:  968
 0 ( Master Manifold control - continued )
 1 ( 4000 CONSTANT )
 2 ( 8000 CONSTANT )    DECIMAL
 3   6 8001 3 LABELS   VNAME
 4   10 8001 9 LABELS GNAME
 5  : .GNAME  GNAME >TYPE ;    : .VNAME   VNAME >TYPE ;
 6  : GSTATUS PAGE 5 0 DO CR 3 SPACES 8000 DUP BLOCK I 64 *
 7       + 64 -TRAILING >TYPE DROP LOOP CR
 8       16 0 DO 3 SPACES
 9       I DUP DUP DUP 5 U.R   12 SPACES .VNAME  12 SPACES
10      .GNAME 9 SPACES  DUP 7 > IF 8 -  MANIFOLD 1+
11       ELSE MANIFOLD THEN ?BIT
12       IF ." OFF " ELSE ." ON "   THEN CR LOOP ;
13    DECIMAL
14  : PARTY  N2O ON ;
15   : CROSS-OVER  CGAS ON DGAS ON ;
```

```
Screen:  969
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  970
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  971
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  972
 0 ( Master - Primatives for Upload to PDP-11 )
 1
 2 : CHECKSUM   0 WORKING DUP 64 + DUP >R SWAP DO I C@ + LOOP
 3    R> ! ;
 4 HEX
 5 : ZAP-8255   8C F803 C!  7 F803 C!  5 F803 C! ;
 6 ZAP-8255
 7
 8 CODE BELCH   I PUSH   21 # 1 MOV   WORKING # I MOV
 9    BEGIN   F802 LDA B   10 #B 0 AND   0= END
10    BEGIN   BEGIN   F802 LDA B   0 SAR B   CS END   LODS
11       F800 STA B   0 0 HI XCHG B   F801 STA B   LOOP
12    I POP   NEXT
13
14 DECIMAL
15

Screen:  973
 0 ( Master - Upload to PDP-11 )
 1
 2 : SEND-FILE   #Records N@ 0 DO  I CEXPT @ + RECORD
 3    WORKING 64 MOVE  CHECKSUM BELCH  LOOP ;
 4
 5
 6 : SEND-TO   1 WORKING C!  32 WORD COUNT DUP 2+ WORKING + >R
 7    DUP WORKING 1+ C!
 8    WORKING 2+ SWAP MOVE  2 I C!  CEREAD #Records N@ I 1+ !
 9    3 I 3 + C!  -2 I 4 + C!  4 I 5 + C!  5 I 6 + C!  6 I 7 + C!
10    R> CHECKSUM BELCH SEND-FILE ;
11
12
13
14
15

Screen:  974
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  975
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  976
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  977
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  978
 0 ( Master - Help Utility )
 1
 2 VARIABLE HSCR  ( current help screen )
 3
 4 : HFIND ( blk addr - f) ( f=0 => found)   1024 0
 5    DO  DUP I + DUP  8 PAD -TEXT
 6       IF  DROP  ELSE  9 + NUMBER HSCR !  DROP 0 LEAVE  THEN  16
 7    +LOOP ;
 8
 9 : HELP   0 HSCR !  CNT @ >IN @ - 1 > NOT
10    IF  3579 HSCR !
11    ELSE  32 TEXT  3600 3580
12       DO  I BLOCK  HFIND NOT IF  LEAVE  THEN  LOOP
13       CR  HSCR @ ?DUP  IF  PAGE  SCREENTYPE  ELSE  CR  THEN
14    ." HELP '" PAD 64 -TRAILING TYPE ." ' NOT FOUND"  THEN  CR ;
15

Screen:  979
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  980
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 981
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 982
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 983
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen:  984
 0 ( Master - Freedom 100 Editor )
 1
 2 HEX 0157 VOCABULARY FR100ED DECIMAL
 3
 4 FR100ED DEFINITIONS
 5
 6 : LINE#   R# @ 64 /MOD ;
 7
 8 : FRLINE   SCUR  DUP 2+ 3 CURSOR  64 * SCR @ BLOCK +
 9    64 TYPE  RCUR  FR100ED ;
10
11 : FRREFRESH   16 SWAP DO  [ FORTH ] I [ FR100ED ] FRLINE  LOOP ;
12
13
14
15

Screen:  985
 0 ( Master - Freedom 100 Editor, cont. )
 1 : F   F FR100ED ;          : T   T FR100ED ;
 2 : P   P LINE# FRLINE ;    : I   I LINE# FRLINE ;
 3 : D   D LINE# FRLINE ;
 4 : R   E FR100ED I ;
 5 : E   E LINE# FRLINE ;
 6 : TILL   TILL LINE# FRLINE ;
 7
 8 : U   U LINE# FRRREFESH ;     : X   X LINE# FRREFRESH ;
 9
10 : WIPE   SCR @ BLOCK 1024 BLANK UPDATE 0 FRREFRESH ;
11 : LIST   PAGE  LIST  FR100ED  19 0 CURSOR ;
12 : Q   FLUSH  FORTH  PAGE ;
13
14 FORTH DEFINITIONS
15

Screen:  986
 0 ( Master - Methods )
 1
 2 3480 CONSTANT methods
 3
 4 : +METHOD   DUP 0 120 WITHIN IF methods + ELSE 13 .ERROR THEN ;
 5
 6 : METHOD   +METHOD LOAD ;
 7
 8 : MLIST   +METHOD LIST ;
 9
10 : MED ( method editor)   DUP +METHOD [ FR100ED ] LIST
11    SCUR 0 0 CURSOR  ." METHOD  " .  RCUR ; FORTH
12
13 : MCOPY ( src dest)   FLUSH EMPTY-BUFFERS SWAP +METHOD
14    SWAP +METHOD  COPY  FLUSH ;
15
```

```
Screen:  987
 0 ( Master - Method Directory )
 1
 2 [R METHOD DIRECTORY]
 3
 4 : MDIR ( method directory)   IN-LINE REPORT  120 0 DO
 5    7 I U. SPACE  0 I methods + .LINE +CR  LOOP ;
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  988
 0 ( Master - Redirected I/O form Methods )
 1
 2 : <INPUT>   ' >R  I/O>  0. >IN 2!  0 CNT !
 3    ."  ? "  S0 @ 80 EXPECT
 4    R>  EXECUTE  >I/O ;
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  989
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  990
 0 ( Master - Sequence Support Words )
 1
 2 VARIABLE STARTING-TIME 2 ALLOT
 3
 4 : ?DURATION ( sec)   1000 U* STARTING-TIME 2@ D+ @TIME D< ;
 5
 6 : ?QUIT   ?TERMINAL 81 = IF  FLUSH  BELL CR
 7    ABORT" Sequence Terminated "  THEN ;
 8
 9 VARIABLE #SEQ
10
11 : (sequence) ( n)   2* 2* 8070 BLOCK #SEQ @ 64 * + + ;
12
13 : @SEQ ( n)   (sequence) 2@ ;
14
15

Screen:  991
 0 ( Master - Sequence )
 1
 2 : ?SEQ   DUP  0 16 WITHIN NOT IF  14 .ERROR  THEN ;
 3
 4 : SEQUENCE ( n)   ?SEQ  #SEQ @ SWAP #SEQ !  0 @SEQ  16 1
 5    DO  @TIME STARTING-TIME 2!
 6       BEGIN  2DUP METHOD ?QUIT ?DURATION  END  2DROP
 7       I @SEQ OVER 0= IF  LEAVE  THEN
 8    LOOP  2DROP  #SEQ ! ;
 9
10
11
12
13
14
15

Screen:  992
 0 ( Master - Sequence Editor - Utilities )
 1
 2 : SEDCUR   #DEVICE @ 16 /MOD  SWAP 4 + SWAP 8 * 4 + CURSOR ;
 3
 4 : 'SEQ   #DEVICE @ 16 /MOD SWAP (sequence) SWAP 2* + ;
 5 : SEDHILITE   REVERSE SEDCUR  'SEQ @ 6 U.R ;
 6 : SEDRESTORE   NORMAL SEDCUR  'SEQ @ 6 U.R SPACE ;
 7
 8 : SLIST ( seq#)   ?SEQ DUP #SEQ !  PAGE ." SEQUENCE # " . CR
 9    CR  ."     METHOD   DURATION" CR CR  16 0
10    DO  I @SEQ  I 2 U.R  8 U.R  8 U.R  CR  LOOP ;
11
12
13
14
15
```

```
Screen:  993
 0 ( Master - Sequence Editor )
 1
 2 : SEDKEYS   CASES    11 <CASE  -1 +DEVICE SEDCUR    ECASE>
 3                      22 <CASE   1 +DEVICE SEDCUR    ECASE>
 4                      12 <CASE  16 +DEVICE SEDCUR    ECASE>
 5                       8 <CASE  16 +DEVICE SEDCUR    ECASE>
 6                       0 <CASE   1 +DEVICE SEDCUR    ECASE>
 7    8 EMIT SPACE 8 EMIT DROP  ALSO  SEDHILITE  ENDCASE ;
 8
 9 : SED ( n)   ?SEQ SLIST  #DEVICE @  0 !DEVICE#  SEDHILITE
10    BEGIN  KEYHIT SEDRESTORE DUP DUP  48 58 WITHIN
11       IF  REVERSE SEDCUR 6 SPACES SEDCUR NUMIN 'SEQ !
12          SEDHILITE
13       ELSE  SEDKEYS  THEN  81 ( Q) = END  SEDRESTORE
14    !DEVICE#  UPDATE FLUSH  22 0 CURSOR ;
15

Screen:  994
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  995
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  996
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  997
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen:  998
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen:  999
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen: 1000
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1001
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1002
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 1003
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1004
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1005
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1006
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1007
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1008
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen: 1009
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1010
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1011
 0 ( Master - Disk Utilities for MS Users )
 1
 2 : +COPY ( src dest)   FLUSH EMPTY-BUFFERS  COPY FLUSH ;
 3 : +cd1 ( n - n)   5049 + ;
 4
 5 : +params ( n - cfix crem mfix mrem )   DUP 0 16 WITHIN
 6    IF  >R  I 8051 + DUP +cd1  R> 64 * 7935 BLOCK UPDATE
 7       OVER +  SWAP 8051 +cd1 BLOCK UPATE +
 8    ELSE  DROP  2 .ERROR  THEN ;
 9
10 : PARAM} ( n)   FLUSH +params 64 MOVE  +COPY ;
11 : }PARAM ( n)   FLUSH +params  SWAP 64 MOVE  SWAP +COPY ;
12
13 : PARAMS}   16 0 DO  I PARAM}  LOOP ;
14 : }PARAMS   16 0 DO  I }PARAM  LOOP ;
15
```

```
Screen: 1012
 0 ( Master - More Disk Utilities )
 1
 2 : SIM}   8067 DUP +cd1 +COPY ;
 3 : }SIM   8067 DUP +cd1 SWAP +COPY ;
 4
 5 : MRM}   8068 DUP +cd1 +COPY ;
 6 : }MRM   8068 DUP +cd1 SWAP +COPY ;
 7
 8 : SPLITS}   8069 DUP +cd1 +COPY ;
 9 : }SPLITS   8069 DUP +cd1 SWAP +COPY ;
10
11 : SEQUENCES}   8070 DUP +cd1 +COPY ;
12 : SEQUENCES    8070 DUP +cd1 SWAP +COPY ;
13
14
15

Screen: 1013
 0 ( Master - More Disk Utilities )
 1 : METHOD} ( n)   3480 + DUP +cd1 +COPY ;
 2 : }METHOD ( n)   3480 + DUP +cd1 SWAP +COPY ;
 3
 4 : METHODS}   120 0 DO  I METHOD}  LOOP ;
 5 : }METHODS   120 0 DO  I }METHOD  LOOP ;
 6
 7 : #B ( #recs - #blks)   16 DUP 1- ROT + SWAP U/ ;
 8
 9 : +blocks   FLUSH EMPTY-BUFFERS  0 DO  I NB @ MOD 0=
10    IF  FLUSH  THEN  OVER I + OVER I + COPY  LOOP  2DROP  FLUSH ;
11
12 : MOVEFILE ( src dest)   datafiles ORG @  SWAP datafiles ORG @
13    SWAP  AVAILABLE @ #B  +blocks  BELL ." complete "  CR
14    @AVAILABLE ;
15

Screen: 1014
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen: 1015
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1016
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1017
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 1018
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1019
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1020
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1021
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1022
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1023
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen: 1024
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1025
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1026
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 1027
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1028
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1029
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1030
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1031
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1032
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1033
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1034
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1035
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1036
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1037
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1038
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1039
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1040
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1041
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1042
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1043
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1044
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1045
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1046
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1047
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1048
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1049
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1050
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1051
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1052
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1053
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1054
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1055
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1056
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1057
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1058
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1059
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1060
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1061
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1062
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1063
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1064
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1065
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1066
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1067
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1068
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1069
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1070
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1071
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1072
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1073
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1074
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1075
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1076
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1077
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen: 1078
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1079
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1080
 0 ( Master - Load Block for Data Acquisition Routines )
 1
 2 CR { Loading Data Acquisition Routines }
 3
 4 1128 1129 THRU     ( Device Sweeps )
 5 1164 1166 THRU     ( Scans )
 6
 7 EXIT
 8
 9
10
11
12
13
14
15
```

Screen: 1081
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1082
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1083
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen: 1084
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1085
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1086
 0 ( Ion Path - Data Acquisition Address Definitions )
 1
 2 HEX
 3
 4 FEC0 CONSTANT XDATA             FEC8 CONSTANT XSTB
 5 FED0 CONSTANT xstable           FED8 CONSTANT AMUF/F
 6 FEE0 CONSTANT YF/F              FEE8 CONSTANT TESTMUX
 7
 8 DECIMAL
 9
10
11 : statCLR  0 STATOUT C! ; COMMAND
12 CODE KRISTO xstable STA NEXT
13
14
15
```

```
Screen: 1087
 0 ( Ion Path - Link and Sync Control )
 1
 2 ( see that detector is done with current step )
 3 ASSEMBLER CREATE ?YSAMPLED   0 0 SUB   TESTMUX STA B   wait
 4    YF/F STA B   ( clear detector flipflop )   RET
 5
 6 ( tell detector "ok to sample" )
 7 ASSEMBLER CREATE XSTABLE   1 # 0 MOV   TESTMUX STA B   wait
 8    ( amu timer )   AMUF/F STA B   xstable STA B   RET
 9
10 ( send x-value to reduction slave - value in reg. 2 )
11 ASSEMBLER CREATE >PEAK   2 XDATA MOV   XSTB STA B   RET
12 CODE TEST->PEAK  2 POP >PEAK CALL NEXT
13 : TESTER 52 TEST->PEAK ; COMMAND
14 CODE XOK   xstable STA    NEXT COMMAND
15 CODE ?YOK   ?YSAMPLED CALL   NEXT COMMAND

Screen: 1088
 0 ( Ion Path - Sweep initialization and termination )
 1
 2 : SWEEPINIT ( step end start dev - startdac dev-address #steps)
 3    0 STATOUT C!   FSTOP rate @ RATE
 4     !DEVICE#   >DAC&STEP DEVICE-ADDRESS
 5     ROT ; COMMAND
 6
 7 : SCANEND   ?YOK 0 XSTB C! 1 STATOUT C! DINIT ;
 8 EXIT
 9
10
11
12
13
14
15

Screen: 1089
 0 ( Ion Path - Perform a sweep )
 1
 2 CODE (SWEEP) ( startdac dev-address #steps )
 3    1 POP   W POP   2 POP
 4    2 W ) MOV   2500 # 0 MOV   BEGIN   0 DEC   0= END
 5    BEGIN
 6          ( put in any linked action here )
 7       ?YSAMPLED CALL   ( check if detection is done )
 8       2 W ) MOV        ( write out the new value )
 9       >PEAK CALL       ( write out x data to reduction )
10       xstable STA B    ( give ok to detection )
11 (      XSTABLE CALL     ( give go ahead to detection )
12       STEP 2 ADD       ( increment the device value )
13    LOOP    NEXT
14 : SWEEP ( step end start dev )   SWEEPINIT  (SWEEP)  SCANEND ;
15 COMMAND
```

Screen: 1090
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1091
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1092
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1093
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1094
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1095
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen: 1096
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1097
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1098
 0 ( Reduction - Data Buffer definitions )
 1
 2 HEX     8000  CONSTANT databuffer  COMMAND     DECIMAL
 3
 4               VARIABLE DATABUFFER  COMMAND
 5               10 ALLOT  ( for following pseudo-VARIABLEs )
 6   DATABUFFER  CONSTANT #RECORDS
 7   #RECORDS 2+ CONSTANT #POINTS
 8   #POINTS 2+  CONSTANT TIC
 9   TIC 2+ 2+   CONSTANT MAX-INTEN
10
11 : >graphics   databuffer graphbuf #POINTS @ 2* 2* 2* MOVE ;
12   COMMAND
13
14
15
```

```
Screen: 1099
 0 ( Reduction - Data Buffer Access )
 1
 2 FORTH : 'buffer  CREATE  ,  ;CODE   W INC   W INC   0 POP
 3    0 SHL   0 SHL   0 SHL   W ) 0 ADD   databuffer # 0 ADD
 4    0 PUSH   NEXT
 5
 6 0 'buffer 'X                    0 'buffer '1MASS
 7 2 'buffer 'X1                   2 'buffer '3MASS
 8 2 'buffer 'WIDTH                3 'buffer 'FLAGS
 9 4 'buffer 'Y
10
11 : !X   'X ! ;                   : !1MASS   '1MASS ! ;
12 : !X1  'X1 ! ;                  : !3MASS   '3MASS ! ;
13 : !WIDTH   'WIDTH ! ;           : !FLAGS   'FLAGS ! ;
14 : !Y   'Y 2! ;                  : @Y   'Y 2@ ;
15

Screen: 1100
 0 ( Reduction - Address definitions for data acquistion )
 1
 2 HEX
 3
 4 ( from Ion Path )
 5 FF08 CONSTANT XRDY                  FF10 CONSTANT XSTROBE
 6 FF18 CONSTANT XDATA                 FF20 CONSTANT XF/F
 7
 8 ( from Detection )
 9 FEC0 CONSTANT RSTROBE                FEC8 CONSTANT YRDY
10 FED0 CONSTANT YSTROBE                FED8 CONSTANT YDATA
11 FEE0 CONSTANT YF/F
12
13 DECIMAL
14
15

Screen: 1101
 0 ( Reduction - Link and Sync )
 1
 2 ( get the x value from Ion Path )
 3 ASSEMBLER CREATE ?XDATA      BEGIN RSTROBE LDA B
 4   XSTROBE LDA B   0 SHR   CS END
 5    XF/F LDA B   XDATA LDA   RET
 6
 7 ( get the y value from Detection )
 8 ASSEMBLER CREATE ?YDATA    BEGIN   YSTROBE LDA B
 9    0 SHR B   CS END   YF/F LDA B
10    YDATA 2 MOV   YDATA LDA
11    RET
12
13
14
15
```

```
Screen: 1102
 0 ( Reduction - Scan/Sweep Initialization )
 1
 2 : SCANINIT ( dev )
 3    !DEVICE#                ( device being scanned )
 4    0. MAX-INTEN 2!         ( clear maximum-intensity )
 5    0 UPFLAG ! 0 +LAST ! 0. yprev 2!
 6    xmax 8 ERASE           ( init peak finding parameters )
 7    DATABUFFER 12 ERASE       ( clear variables )
 8    0 STATOUT C!              ( clear its own status ) ;
 9    COMMAND
10
11
12
13
14
15

Screen: 1103
 0 ( Reduction - Data Buffer routines for data acquisition )
 1
 2 : ?MAX-INTEN   2DUP TIC 2@ D+ TIC 2!   MAX-INTEN 2@ DMAX
 3    MAX-INTEN 2! ;
 4
 5 : BUF>UNITS   0. 2DUP MAX-INTEN 2! TIC 2!   #POINTS @
 6    ?DUP IF  0 DO  I 'X DUP  @ >UNITS  SWAP !
 7    I @Y  ?MAX-INTEN  LOOP  ELSE 0. TIC 2!  0. MAX-INTEN 2!
 8    THEN ;
 9
10
11
12
13
14
15

Screen: 1104
 0 ( Reduction - Store a peak )
 1
 2 ASSEMBLER DEFINITIONS
 3
 4 FORTH : !XDATA   STOS   0 0 SUB   STOS ;
 5
 6 FORTH : !YDATA   2 0 XCHG   STOS   2 0 MOV   STOS ;
 7
 8 HOST DEFINITIONS
 9
10
11
12
13
14
15
```

```
Screen: 1105
 0 ( Reduction - Sweep data processing )
 1
 2 CODE (SWEEP)
 3    databuffer # W MOV
 4    BEGIN
 5       ?XDATA CALL   !XDATA
 6       ?YDATA CALL   !YDATA
 7       SSTAT 3 + LDA B   0 SHR   CS     ( test for end of scan )
 8    END
 9    databuffer # W SUB   W SHR   W SHR   W SHR   W #POINTS MOV
10 NEXT
11
12
13
14
15

Screen: 1106
 0 ( Reduction - Scan Termination )
 1
 2 : SCANEND      BUF>UNITS
 3    databuffer graphbuf  #POINTS @  2* 2* 2* MOVE
 4    xmax 20 ERASE
 5    1 STATOUT C! ;
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1107
 0 ( Reduction - Perform a sweep )
 1
 2 : SWEEP ( dev)   SCANINIT  (SWEEP)  SCANEND ;
 3 COMMAND
 4 CODE XGET ?XDATA CALL 0 PUSH NEXT
 5 CODE YGET ?YDATA CALL 0 PUSH 2 PUSH NEXT
 6
 7  : (SCAN)  SCANINIT 0 0 (CUR) ." HI MIKEY "
 8      BEGIN  XGET . YGET D. CR SSTAT 3 + C@
 9      1 = END SCANEND ;  COMMAND
10
11 : TRIER  XGET 0 0 (CUR) ." I DID IT " . ; COMMAND
12 : 2TRIER  YGET 0 0 (CUR) ." I DID IT " D. ; COMMAND
13
14
15
```

Screen: 1108
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1109
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1110
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen: 1111
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1112
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1113
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen: 1114
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1115
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1116
 0 ( Detection - Addresses for data acquisition )
 1
 2 HEX
 3
 4 F8C0 CONSTANT PIO
 5 F880 CONSTANT AMD9513
 6
 7 FEC0 CONSTANT YDATA              FEC8 CONSTANT YSTROBE
 8 FED0 CONSTANT YSAMPLED           FED8 CONSTANT PF/F
 9 FEE0 CONSTANT XF/F               FEE8 CONSTANT TESTMUX
10 DECIMAL
11
12
13
14
15
```

```
Screen: 1117
 0 ( Detection - Link and Sync )
 1 ASSEMBLER DEFINITIONS
 2 FORTH : >PEAK  ( writes out hi lo )
 3   YDATA STA   2 YDATA MOV
 4   1 #B TESTMUX MOV  20 # 0 MOV BEGIN 0 DEC 0= END
 5   wait  ( wait on peak finder )
 6   PF/F STA B   ( clear flipflop, then send data )
 7   YSTROBE STA B ;
 8
 9 FORTH : ?XSTABLE  0 #B TESTMUX MOV
10   wait     ( wait on Ion Path )
11   XF/F STA B  ( clear Ion Path F/F ) ;
12 HOST DEFINITIONS
13 CODE TEST->PEAK 0 POP 2 POP >PEAK NEXT
14 : SENDER  52 43 TEST->PEAK ; COMMAND
15

Screen: 1118
 0 ( Detection - Acquire a data point )
 1
 2 HEX
 3 ASSEMBLER DEFINITIONS
 4 FORTH : ACQUIRE ( - lo hi)
 5    0 0 SUB   0 DEC   startdaq STA B   ( start the daq )
 6    ( wait for daq buffer to be ready )
 7    BEGIN   daqstat LDA B   08 #B 0 TEST   0= NOT END
 8    ( get the data point )
 9    daqbase2 2 MOV B   daqbase2 1 + 2 HI MOV B
10    0 0 SUB   daqbase2 2 + LDA B
11    ( and reset the daq )
12    daqf/fclr STA B   ;
13 HOST DEFINITIONS
14 DECIMAL
15

Screen: 1119
 0 ( Detection - Data acquisition loop )
 1 CODE (SCAN)
 2    YSAMPLED STA B
 3    BEGIN
 4       ?XSTABLE              (  wait for ok from Ion Path )
 5       ACQUIRE                  ( leaves data point on stack )
 6       YSAMPLED STA B           ( send ok to Ion Path )
 7       >PEAK                    ( send data to Reduction )
 8       SSTAT 1+ LDA B   0 SHR   CS   ( test for end of scan )
 9    END 1 # 0 MOV STATOUT STA NEXT
10
11 CODE ?XOK  0 #B TESTMUX MOV  50 # 0 MOV BEGIN 0 DEC 0= END
12     wait NEXT
13 CODE YOK   YSAMPLED STA B NEXT
14
15
```

```
Screen: 1120
 0 ( Detection - Scanning )
 1 HEX
 2 CODE TEST->PEAK  >PEAK NEXT
 3 CODE TEST-ACQUIRE ACQUIRE NEXT
 4 DECIMAL
 5
 6 : SCANINIT    0 daqrst C!  0 #points C! 0 STATOUT C!
 7     TEST-ACQUIRE 2DROP ( cold start kluge ) rate @ RATE
 8     0 XF/F C! ; COMMAND
 9
10 : SCAN   SCANINIT   (SCAN) ;  COMMAND
11 : SWEEP   SCAN ;  COMMAND
12 : SIM/MRM   SCAN ;  COMMAND
13
14
15

Screen: 1121
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1122
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 1123
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1124
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1125
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen: 1126
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1127
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1128
 0 ( Master - Scan Termination )
 1
 2 : ?COMPLETE   BEGIN  PAUSE  SSTAT 2+ C@  1 = END ;
 3
 4
 5 : GET-DATA   2 0 >SLAVE   2DATABUFFER #RECORDS 12 IPMOVE
 6    2databuffer DATABUFFER  #POINTS @ 2* 2* 2* E<IMOVE ;
 7
 8
 9 : scanend ( n)   >SHEADER  >FILE  FLUSH  !AVAILABLE ;
10
11
12
13
14
15
```

```
Screen: 1129
 0 ( Master - Sweeps )
 1
 2 : SWEEPINIT
 3    SL1 statCLR
 4    #DEVICE @ DUP   2PUSH
 5    @PARAMS  1PUSH
 6    SL2 SWEEP   SL3 SWEEP   SL1 SWEEP ;
 7
 8 : SWEEPEND   ?COMPLETE  GET-DATA  5 DUP stype !  scanend ;
 9
10 : SWEEP  SWEEPINIT   SWEEPEND ;
11
12
13
14
15

Screen: 1130
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1131
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 1132
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1133
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1134
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1135
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1136
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1137
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen: 1138
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1139
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1140
 0 ( Ion Path - Scan Initialization )
 1
 2 : SCANINIT ( step end start - startdac #steps)
 3    0 STATOUT C!   FSTOP  rate @ RATE  0 AMUF/F C!
 4    3>DAC #STEPS ;
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen: 1141
 0 ( Ion Path - 1SCAN loop )
 1
 2 CODE (1SCAN)   ( #steps startdac )
 3    0 POP   1 POP   0 2 MOV
 4    M1DAC STA   0 SHR   M2DAC STA   M3DAC STA
 5    2500 # 0 MOV   BEGIN   0 DEC   0= END   ( settling time )
 6    BEGIN
 7       ?YSAMPLED CALL ( check if detection is done )
 8       2 0 MOV   M1DAC STA   0 SHR   M2DAC STA   M3DAC STA
 9       xstable STA B
10 ( XSTABLE CALL ( give ok to detection and sync w/ amutimer )
11       >PEAK CALL     ( send data in 2 to Reduction )
12       STEP 2 ADD     ( increment the quad 1 value )
13    LOOP   NEXT
14
15

Screen: 1142
 0 ( Ion Path - 3SCAN loop )
 1
 2 CODE (3SCAN)   ( #steps startdac )
 3    0 POP   1 POP   0 2 MOV
 4    M3DAC STA   0 SHR   M2DAC STA   M1DAC STA
 5    2500 # 0 MOV   BEGIN   0 DEC   0= END   ( settling time )
 6    BEGIN
 7       ?YSAMPLED CALL ( check if detection is done )
 8       2 0 MOV   M3DAC STA   0 SHR   M2DAC STA   M1DAC STA
 9       xstable STA B
10  ( XSTABLE CALL ( give ok to detection and sync w/ amutimer )
11       >PEAK CALL     ( send data in 2 to Reduction )
12       STEP 2 ADD     ( increment the quad 3 value )
13    LOOP   NEXT
14
15

Screen: 1143
 0 ( Ion Path - PSCAN loop )
 1
 2 CODE (PSCAN)   ( #steps startdac )
 3    0 POP   1 POP   0 2 MOV
 4    M1DAC STA   0 SHR   M2DAC STA
 5    2500 # 0 MOV   BEGIN   0 DEC   0= END   ( settling time )
 6    BEGIN
 7       ?YSAMPLED CALL ( check if detection is done )
 8       2 0 MOV   M1DAC STA   0 SHR   M2DAC STA
 9       xstable STA B
10    ( XSTABLE CALL   ( give ok to detection and sync w/ amutime)
11       >PEAK CALL     ( send data in 2 to Reduction )
12       STEP 2 ADD     ( increment the quad 1 value )
13    LOOP   NEXT
14
15
```

```
Screen: 1144
 0 ( Ion Path - DSCAN loop )
 1
 2 CODE (DSCAN)   ( #steps startdac )
 3    0 POP   1 POP   0 2 MOV
 4    M3DAC STA   0 SHR   M2DAC STA
 5    2500 # 0 MOV   BEGIN   0 DEC   0= END   ( settling time )
 6    BEGIN
 7       ?YSAMPLED CALL ( check if detection is done )
 8       2 0 MOV   M3DAC STA   0 SHR   M2DAC STA
 9       xstable STA B
10    ( XSTABLE CALL   ( give ok to detection and sync w/ amutime)
11       >PEAK CALL     ( send data in 2 to Reduction )
12       STEP 2 ADD     ( increment the quad 3 value )
13    LOOP   NEXT
14
15

Screen: 1145
 0 ( Ion Path - NSCAN loop )
 1
 2 CODE (NSCAN)   ( #steps startdac )
 3    0 POP   1 POP   0 2 MOV   FQ3 W MOV   I PUSH   FQ3 2+ I MOV
 4    M1DAC STA   W 0 MOV   M3DAC STA   0 SHR   M2DAC STA
 5    2500 # 0 MOV   BEGIN   0 DEC   0= END   ( settling time )
 6    BEGIN
 7       ?YSAMPLED CALL ( check if detection is done )
 8       2 M1DAC MOV   W 0 MOV   M3DAC STA   0 SHR   M2DAC STA
 9       xstable STA B
10    ( XSTABLE CALL   ( give ok to detection and sync w/ amutime)
11       >PEAK CALL     ( send data in 2 to Reduction )
12       STEP 2 ADD     ( increment the quad 1 value )
13       FRAC 2+ I ADD   FRAC W ADC   ( incr the quad 3 value )
14    LOOP   I POP   NEXT
15

Screen: 1146
 0 ( Ion Path - Scanning )
 1
 2 : 1SCAN ( step end start )   DRR  M1 SCANINIT  (1SCAN)
 3    SCANEND ;  COMMAND
 4
 5 : 3SCAN ( step end start )   RRD  M3 SCANINIT  (3SCAN)
 6    SCANEND ;  COMMAND
 7
 8 : PSCAN ( step end start )   DRD  M1 SCANINIT  (PSCAN)
 9    SCANEND ;  COMMAND
10
11 : DSCAN ( step end start )   DRD  M3 SCANINIT  (DSCAN)
12    SCANEND ;  COMMAND
13
14 : NSCAN ( step end start offset )   DRD  NSET  M1 SCANINIT
15    (NSCAN)  SCANEND ;  COMMAND
```

```
Screen: 1147
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1148
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1149
 0 ( Red. - Store a peak - for ultranew algo.- MJK 10/31/85 )
 1
 2 ASSEMBLER CREATE !PEAK   I PUSH
 3    xmax # I MOV   4 # 1 MOV   REP MOVS
 4    I POP   RET
 5 FORTH
 6
 7
 8
 9
10
11
12
13
14
15
```

```
Screen: 1150
 0 ( ULTRANEW PEAK-FINDING ALGORITHM - MJK ) HEX ASSEMBLER
 1 CREATE PEAK-up   UPFLAG INC    +LAST INC   RET
 2 CREATE PEAK-down    -1 # UPFLAG TEST   0=
 3   IF   max-peak 2+ 1 MOV   max-peak 2 MOV  threshold 1 SUB
 4         ( threshold 2+ ) 0 # 2 SBB  0< NOT
 5        IF   !PEAK CALL   THEN
 6        0 0 SUB   max-peak STA   max-peak 2+ STA
 7   ELSE   0 0 SUB   UPFLAG STA   THEN    RET
 8 CREATE PEAK-0   1 # UPFLAG CMP   CS NOT
 9   IF  yprev 2+ 1 MOV   yprev 2 MOV   max-peak 2+ 1 SUB
10      max-peak 2 SBB  0< NOT
11        IF   yprev LDA   max-peak STA
12             yprev 2+ LDA   max-peak 2+ STA
13             xvalue LDA   xmax STA
14        THEN THEN   0 0 SUB   +LAST STA   RET
15 DECIMAL

Screen: 1151
 0 ( ULTRANEW PEAK-FINDING ALGORITHM - MJK 10/31/85 )
 1
 2 ASSEMBLER CREATE ?PEAK   ( expects hi cell in 2, lo cell in 0 )
 3 ( also, regs. I and U must have been already pushed )
 4   2 U MOV   0 1 MOV  0 I MOV
 5   yprev 2+ 0 SUB   yprev 2 SBB   0<
 6
 7      IF   -1 # +LAST TEST   0=
 8         IF   PEAK-down CALL
 9         ELSE PEAK-0 CALL
10         THEN
11      ELSE   PEAK-up CALL
12      THEN
13
14   I yprev 2+ MOV   U yprev MOV      RET
15

Screen: 1152
 0 ( Reduction - Variable threshold control for peak finding )
 1 EXIT
 2 ( assumes lo cell of y in 1 and I, hi cell in 2 and U )
 3
 4 HEX
 5 ASSEMBLER CREATE NEWTHRESH
 6    max-peak 2+ 1 SUB   max-peak 2 SBB   0<
 7    IF   xvalue LDA   xmax STA
 8       I max-peak 2+ MOV   U max-peak MOV   U U OR   0=
 9       IF   I SHR   I (thld) MOV
10       ELSE   8000 # (thld) MOV
11       THEN
12    THEN
13 RET
14
15 DECIMAL
```

```
Screen: 1153
 0 ( Reduction - Store a peak )
 1 EXIT
 2 ASSEMBLER DEFINITIONS
 3
 4 FORTH : !PEAK   xmax # I MOV   4 # 1 MOV   MOVS REP
 5    0 0 SUB   xmax STA   xlast STA   max-peak STA
 6    max-peak 2+ STA   1 # kflag MOV ;
 7 VARIABLE FOUND-IT
 8 FORTH : !PEAK   56 # 0 MOV  FOUND-IT STA
 9   0 0 SUB  xmax STA  max-peak STA  max-peak 2+ STA
10   xlast STA ;
11 HOST DEFINITIONS
12
13
14
15

Screen: 1154
 0 ( Reduction - Check for a peak that's still rising )
 1 EXIT
 2 ( assumes lo cell of y in 1 and I, hi cell in 2 and U )
 3
 4 HEX
 5
 6 ASSEMBLER CREATE ?RISING
 7    yprev 2+ 1 SUB   yprev 2 SBB   0< NOT
 8    IF   I yprev 2+ MOV   U yprev MOV
 9    ELSE   0 # kflag MOV
10    THEN
11 RET
12
13 DECIMAL
14
15                                                              EXI

Screen: 1155
 0 ( Scanning - perform peak-finding   ) EXIT HEX
 1 CODE (SCAN)   U PUSH   I PUSH   databuffer # W MOV
 2    BEGIN   ?XDATA CALL  0 xvalue MOV   ?YDATA CALL
 3       2 POP   1 POP   2 U MOV   1 I MOV
 4       (thld) 1 SUB   0 # 2 SBB   0< NOT
 5       IF   ylast INC   kflag LDA   0 0 OR   0=
 6          IF   xlast INC   xlast LDA   0 mwidth CMP  CS
 7             IF   !PEAK   ELSE   NEWTHRESH CALL   THEN
 8          ELSE   ?RISING CALL   THEN
 9       ELSE   ylast 1 MOV   0 # ylast MOV   1 1 OR   0= NOT
10          IF   xlast LDA   0 pwidth CMP B   CS
11             IF   !PEAK   THEN
12             0 # kflag MOV   THEN
13       THEN   SSTAT 1+ LDA B   0 SHR   CS
14    END   I POP   U POP   databuffer # W SUB   W SHR   W SHR
15    W SHR   W #POINTS MOV   NEXT   DECIMAL
```

```
Screen: 1156
 0 ( Scanning - perform peak-finding   )  HEX
 1
 2 CODE (SCAN)   U PUSH   I PUSH   databuffer # W MOV
 3    BEGIN     ?XDATA CALL  xvalue STA   ?YDATA CALL
 4       ?PEAK CALL
 5       SSTAT 3 + LDA B   0 SHR   CS
 6    END   I POP   U POP   databuffer # W SUB   W SHR   W SHR
 7    W SHR   W #POINTS MOV   NEXT   DECIMAL
 8
 9
10
11
12
13
14
15

Screen: 1157
 0 ( Reduction - Scanning with peak finding )
 1
 2 : SCAN ( dev)   SCANINIT  (SCAN)  SCANEND ;  COMMAND
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1158
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 1159
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1160
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1161
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen: 1162
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1163
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1164
 0 ( Master - Scan Initialization )
 1
 2 : SCANINIT   SL1 statCLR #DEVICE @    2PUSH
 3    @PARAMS     SL2 SCAN  SL3 SCAN ;
 4
 5 : 1SCANINIT   M1 SCANINIT  SL1 1SCAN ;
 6
 7 : 3SCANINIT   M3 SCANINIT  SL1 3SCAN ;
 8
 9 : PSCANINIT   M1 SCANINIT  SL1 PSCAN ;
10
11 : DSCANINIT   M3 SCANINIT  SL1 DSCAN ;
12
13 : NSCANINIT   M1 SCANINIT  NSET  SL1 NSCAN ;
14
15
```

```
Screen: 1165
 0 ( Master - Scan termination )
 1
 2 : SCANEND ( stype)   ?COMPLETE  GET-DATA  DUP stype !  scanend ;
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1166
 0 ( Master - Scans )
 1
 2 : 1SCAN   1SCANINIT   0 SCANEND ;
 3 : 1SCANS   0 DO  1SCAN  LOOP ;
 4 : 3SCAN   3SCANINIT   1 SCANEND ;
 5 : 3SCANS   0 DO  3SCAN  LOOP ;
 6 : PSCAN   PSCANINIT   2 SCANEND ;
 7 : PSCANS   0 DO  PSCAN  LOOP ;
 8 : DSCAN   DSCANINIT   3 SCANEND ;
 9 : DSCANS   0 DO  DSCAN  LOOP ;
10 : NSCAN   NSCANINIT   4 SCANEND ;
11 : NSCANS   0 DO  NSCAN  LOOP ;
12
13
14
15

Screen: 1167
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 1168
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1169
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1170
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1171
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1172
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1173
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1174
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1175
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1176
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1177
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1178
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1179
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen: 1180
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1181
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1182
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 1183
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1184
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1185
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1186
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1187
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1188
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1189
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1190
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1191
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1192
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1193
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1194
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1195
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1196
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1197
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen: 1198
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1199
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1200
 0 ( Master - Load Block for MRM and SIM )
 1
 2 CR { Loading MRM and SIM routines }
 3
 4 1230  1237 THRU  ( MRM and SIM routines )
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 1201
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1202
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1203
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen: 1204
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1205
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1206
 0 ( Ion Path - Put masses into parameter tables )
 1 ASSEMBLER CREATE movmass   8 # 1 MOV
 2    BEGIN   MOVS   62 # W ADD   LOOP   RET
 3 CODE MRMS>PTABLE ( #sims)   1 POP   I PUSH   MASSES # I MOV
 4    PVALUES 32 + # W MOV   movmass CALL
 5    PVALUES 34 + # W MOV   movmass CALL
 6    PVALUES 36 + # W MOV   movmass CALL   I POP   NEXT
 7
 8 CODE 1SIMS>PTABLE ( #sims)   1 POP   I PUSH   MASSES # I MOV
 9    PVALUES 32 + # W MOV   BEGIN   LODS   0 W ) MOV   0 SHR
10    0 2 W) MOV   0 4 W) MOV   62 # 2 ADD   LOOP   I POP   NEXT
11
12 CODE 3SIMS>PTABLE ( #sims)   1 POP   I PUSH   MASSES # I MOV
13    PVALUES 32 + # W MOV   BEGIN   LODS   0 4 W) MOV   0 SHR
14    0 2 W) MOV   0 W ) MOV   62 # W ADD   LOOP   I POP   NEXT
15
```

```
Screen: 1207
 0 ( Ion Path - Convert param tables into DAC units )
 1
 2 : PARAMS>DACS   #DEVICE @  #SIMS @ 64 *  0
 3    DO  I PVALUES +   32 0 DO  I !DEVICE#  ?INSTALLED
 4       IF  DUP @  >DAC  SWAP !  ELSE  DROP  THEN  LOOP
 5    64 +LOOP  !DEVICE# ;
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1208
 0 ( Ion Path - Put constant masses into param tables )
 1
 2 : DAUGHTER>PARAMS   M3 VALUE @   PVALUES 36 +
 3    #SIMS @ 64 *  0 DO  2DUP I + !  64 +LOOP  2DROP ;
 4
 5 : PARENT>PARAMS   M1 VALUE @   PVALUES 32 +
 6    #SIMS @ 64 *  0 DO  2DUP I + !  64 +LOOP  2DROP ;
 7
 8
 9
10
11
12
13
14
15

Screen: 1209
 0 ( Ion Path - Fast parameter switching )
 1
 2 ASSEMBLER CREATE pswitch   I PUSH
 3    PNEXT LDA   ( index into parameters table )
 4    0 W MOV   W INC   W PNEXT MOV   ( increment and save )
 5    5 # 1 MOV   0 SHL V   ( convert index to offset )
 6    PVALUES # I MOV   0 I ADD   ( offset + base of source )
 7    device-address # W MOV   22 # 1 MOV
 8    BEGIN   LODS   W ) 2 MOV   -1 # 2 CMP   0= NOT
 9       IF   W 2 XCHG   0 W ) MOV   W 2 XCHG   THEN
10       W INC   W INC      ( point to ptr for next device )
11    LOOP   I POP   RET
12
13
14
15
```

```
Screen: 1210
 0 ( Ion Path - SIM/MRM Utilities )
 1
 2 : RXRATE ( rrate )   8 + RATE ;
 3
 4 VARIABLE SCYCLES   COMMAND
 5
 6 : INITSIM ( #sims #cycles rrate)   RXRATE  SCYCLES !
 7    DUP #SIMS !  0.  SSTAT 1+ C!  AMUF/F C! ;
 8
 9
10
11
12
13
14
15

Screen: 1211
 0 ( Ion Path - SIM/MRM run time code )
 1
 2 CODE SIM/MRM
 3    BEGIN   #SIMS 1 MOV   0 0 SUB   PNEXT STA
 4       BEGIN   ?YSAMPLED CALL   pswitch CALL   2500 # 0 MOV
 5          BEGIN   0 DEC   0= END   xstable STA B
 6          PNEXT 2 MOV   >PEAK CALL
 7       LOOP   1 # 0 MOV   TESTMUX STA B   wait   AMUF/F STA B
 8       SCYCLES DEC   0=
 9    END   NEXT
10
11
12
13
14
15

Screen: 1212
 0 ( Ion Path - SIM/MRM Commands )
 1
 2 : 1SIM ( #sims #cycles rrate)   INITSIM  1SIMS>PTABLE
 3    PARAMS>DACS  DRR  SIM/MRM  SCANEND ;   COMMAND
 4
 5 : 3SIM ( #sims #cycles rrate)   INITSIM  3SIMS>PTABLE
 6    PARAMS>DACS  RRD  SIM/MRM  SCANEND ;   COMMAND
 7
 8 : PSIM ( #sims #cycles rrate)   INITSIM  1SIMS>PTABLE
 9    PARAMS>DACS  DAUGHTER>PARAMS  DRD  SIM/MRM  SCANEND ; COMMAND
10
11 : DSIM ( #sims #cycles rrate)   INITSIM  3SIMS>PTABLE
12    PARAMS>DACS  PARENT>PARAMS  DRD  SIM/MRM  SCANEND ;  COMMAND
13
14 : MRM ( #sims #cycles rrate)   INITSIM  MRMS>PTABLE
15    PARAMS>DACS  DRD  SIM/MRM  SCANEND ;   COMMAND
```

Screen: 1213
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1214
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1215
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen: 1216
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1217
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1218
 0 ( Reduction - SIM/MRM graphics primitives )
 1
 2 VARIABLE #SIMS
 3
 4 : RXFIELDS-INIT   #SIMS @ 5 MIN  DUP DUP DUP  d#fields !
 5    20 * 750 SWAP - SWAP /  PPNTS/FIELD !  760 SWAP 0
 6    DO  I FIELD  DUP PPNTS/FIELD @ - DUP  ROT 35 1000 WINDOW
 7       20 - YSTART 2+ @ YEND @ TSTART @ TEND @  SCLSET
 8    LOOP  DROP ;
 9
10 : reset   CLEAR  d#fields @ 0 DO  I FIELD FRAME  0 100 XTICS
11    YEND @ DUP MSYTAG YTAGS  UPY @ 10 - LPX @ 5 + (CUR)  I .RXN
12    LOOP  DOT ;
13
14 : ?RESET   1000 MOD 0= IF  reset  THEN ;
15
```

```
Screen: 1219
 0 ( Reduction - collect data from slaves, plot, and store )
 1
 2 CODE mastersync   BEGIN   SSTAT 2+ LDA B   0= END   NEXT
 3
 4 VARIABLE #CYCLE
 5
 6 : COLLECT    #SIMS @ 0 DO  ?XDATA DROP  ?YDATA  I 5 <
 7    IF  2DUP NORMALIZE ?LOG  #CYCLE @ 1000 MOD  PLOT  THEN
 8    I !Y  LOOP ;
 9
10 CODE data-available   0 0 SUB   0 INC   SSTAT 2+ STA B   NEXT
11
12
13
14
15

Screen: 1220
 0 ( Reduction - SIM/MRM run loop )
 1
 2 : sim   BEGIN  mastersync   #CYCLE DUP @ 1+ DUP ROT !
 3    ?RESET  COLLECT  data-available  SSTAT 1+ C@  1 = END ;
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1221
 0 ( Reduction - SIM/MRM initialization and execution )
 1
 2 : INITSIM ( rrange rgain #sims )   0 SSTAT 2+ C!  0 #CYCLE !
 3    1 UMAX  SWAP 1 UMAX 16 UMIN  U* MAX-INTEN 2!  #SIMS !
 4    0 ?RESET ;
 5
 6 : SIM ( rrange rgain #sims )   INITSIM  sim ;  COMMAND
 7
 8
 9
10
11
12
13
14
15
```

Screen: 1222
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1223
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1224
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1225
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1226
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1227
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen: 1228
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1229
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1230
 0 ( Master - SIM/MRM Termination )
 1
 2 1LABEL SCYCLES SCYCLES
 3
 4 : simkill   1 SCYCLES I! ;
 5
 6 : RXEND   LSCAN @ READ  RRECS @ DUP DUP  #Records N+!
 7    Step N@ * End N!  UPDATE FLUSH
 8    LEXPT @ READ  #Records N+!  UPDATE FLUSH ;
 9
10 : WAIT   CR  ." press any key to begin data acquisition, or Q to
11  quit "  KEY 81 = IF  0 RRECS !  simkill  RXEND  CR
12    1 ABORT" Run aborted "  THEN  CR ;
13
14
15
```

```
Screen: 1231
 0 ( Master - SIM/MRM Utilities )
 1
 2 : simpoints ( - n)   RTMAX @ 10  STEPTIM */  1 MAX
 3    DUP RRECS ! ;
 4
 5 : memchk   DUP  LIM @ AVAILABLE @ -  SWAP U<
 6    IF  21 .ERROR  THEN ;
 7
 8 CODE ?datarec   BEGIN   SSTAT 2+ LDA B   1 #B 0 CMP   0= END
 9    NEXT
10
11 CODE dataack   0 0 SUB   SSTAT 2+ STA B   NEXT
12
13
14
15

Screen: 1232
 0 ( Master - more MRM/SIM utilities )
 1
 2 2LABEL  databuffer 2databuf
 3
 4 : @cycle   ?datarec  2 0 >SLAVE
 5    2databuf DATABUFFER 64 E<IMOVE  dataack ;
 6
 7 : COLLECT   BEGIN   ?TERMINAL 81 = IF  simkill  THEN
 8    @cycle  SSTAT 1+ C@ END ;
 9
10 : RCHECKS   SCHK  RATECHK  memchk ;
11
12
13
14
15

Screen: 1233
 0 ( Master - SIM/MRM - downloads to slave 1 )
 1
 2 : SMASSES>SL1   0 1 >SLAVE  0 0 SIM-TABLE MASSES 16 E>IMOVE ;
 3 : RMASSES>SL1   0 1 >SLAVE   0 0 MRM-TABLE MASSES 48 E>IMOVE ;
 4
 5 : PSET>BUF ( sim# pset)   8051 + BLOCK  IMODE @ 256 * +
 6    SWAP  64 * DATABUFFER + 64 E<MMOVE ;
 7
 8 : SPGETS   #SIMS @ 0 DO  I DUP  1 SWAP SIM-TABLE @ DUP
 9    0 16 WITHIN IF  PSET>BUF  ELSE  2DROP  2 .ERROR  THEN  LOOP ;
10
11 : RPGETS   #SIMS @ 0 DO  I DUP  3 SWAP MRM-TABLE @ DUP
12    0 16 WITHIN IF  PSET>BUF  ELSE  2DROP  2 .ERROR  THEN  LOOP ;
13
14 : PINIT   0 1 >SLAVE  DATABUFFER PVALUES #SIMS @ 64 * E>IMOVE ;
15
```

```
Screen: 1234
 0 ( Master - SIM/MRM ions to slave 2 )
 1 : IONS>SL2  0 2 >SLAVE  DATABUFFER 2databuf 60 E>IMOVE  ;
 2
 3 : 1ions   #SIMS @ 0 DO  0 I SIM-TABLE @ I !1MASS  -2 I !3MASS
 4    LOOP ;
 5 : 3ions   #SIMS @ 0 DO  0 I SIM-TABLE @ I !3MASS  -2 I !1MASS
 6    LOOP ;
 7
 8 : 1IONS   1ions  IONS>SL2 ;     : 3IONS   3ions  IONS>SL2 ;
 9 : PIONS   1ions  M3 'CURRENT @  #SIMS @ 0 DO  DUP I !3MASS
10    LOOP  DROP  IONS>SL2 ;
11 : DIONS   3ions  M1 'CURRENT @  #SIMS @ 0 DO  DUP I !1MASS
12    LOOP  DROP  IONS>SL2 ;
13 : MIONS   #SIMS @ 0 DO  0 I MRM-TABLE @ I !1MASS
14    2 I MRM-TABLE @ I !3MASS  LOOP  IONS>SL2 ;
15

Screen: 1235
 0 ( Master - SIM/MRM initialization )
 1
 2 : rxinit   PINIT  #SIMS @ DUP 1PUSH 2PUSH  simpoints 1PUSH
 3    RRATE @ 1PUSH  RRANGE @ 2PUSH  RGAIN @ 2PUSH  SL2 SIM
 4    SL3 SIM/MRM ;
 5
 6 : SINIT   SMASSES>SL1  SPGETS  rxinit ;
 7
 8 : MRMINIT   RMASSES>SL1  RPGETS  rxinit ;
 9
10
11
12
13
14
15

Screen: 1236
 0 ( Master - 1SIM, 3SIM, and PSIM )
 1
 2 : 1SIM   KNOBSOFF  RCHECKS  1IONS  SINIT  SL1 1SIM
 3    7 >RHEADER  WAIT  2COUNTER  COLLECT  2TIMER
 4    ." elapsed"  CR  ." 1SIM complete"  RXEND  CR  BELL ;
 5
 6 : 3SIM   KNOBSOFF  RCHECKS  3IONS  SINIT  SL1 1SIM
 7    8 >RHEADER  WAIT  2COUNTER  COLLECT  2TIMER
 8    ." elapsed"  CR  ." 3SIM complete"  RXEND  CR  BELL ;
 9
10 : PSIM   KNOBSOFF  RCHECKS  PIONS  SINIT  SL1 PSIM
11    9 >RHEADER  WAIT  2COUNTER  COLLECT  2TIMER
12    ." elapsed"  CR  ." PSIM complete"  RXEND  CR  BELL ;
13
14
15
```

```
Screen: 1237
 0 ( Master - DSIM and MRM )
 1
 2 : DSIM   KNOBSOFF  RCHECKS  DIONS  SINIT  SL1 DSIM
 3    10 >RHEADER  WAIT  2COUNTER  COLLECT  2TIMER
 4    ." elapsed"  CR  ." DSIM complete"  RXEND  CR  BELL ;
 5
 6 : MRM   KNOBSOFF  RCHECKS  MIONS  MRMINIT  SL1 MRM
 7    11 >RHEADER  WAIT  2COUNTER  COLLECT  2TIMER
 8    ." elapsed"  CR  ." MRM complete"  RXEND  CR  BELL ;
 9
10
11
12
13
14
15

Screen: 1238
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1239
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 1240
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1241
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1242
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1243
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1244
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1245
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1246
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1247
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1248
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1249
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen: 1250
 0 ( Master - More Disk Utilities )
 1 : METHOD} ( n)   3480 + DUP +cd1 +COPY ;
 2 : }METHOD ( n)   3480 + DUP +cd1 SWAP +COPY ;
 3
 4 : METHODS}   120 0 DO  I METHOD}  LOOP ;
 5 : }METHODS   120 0 DO  I }METHOD  LOOP ;
 6
 7 : #B ( #recs - #blks)   16 DUP 1- ROT + SWAP U/ ;
 8
 9 : +blocks   FLUSH EMPTY-BUFFERS  0 DO  I NB @ MOD 0=
10    IF  FLUSH  THEN  OVER I + OVER I + COPY  LOOP  2DROP  FLUSH ;
11
12 : MOVEFILE ( src dest)   datafiles ORG @  SWAP datafiles ORG @
13    SWAP  AVAILABLE @ #B  +blocks  BELL ." complete "  CR
14    @AVAILABLE ;
15

Screen: 1251
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1252
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 1253
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1254
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1255
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1256
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1257
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1258
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen: 1259
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1260
 0 ( Master - Load Block for Mass Calibration )
 1 CR  { Loading Mass Calibration Routines }
 2
 3 1278 1289 THRU
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1261
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 1262
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1263
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1264
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen: 1265
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1266
 0 ( Ion Path - Calibration Scans )
 1
 2 : 1CSCAN ( step end start )   DRR  M1 SCANINIT
 3    9 1 DO  0 STATOUT C!  2DUP  (1SCAN)  SCANEND
 4        BEGIN SSTAT 3 + C@  I =  END
 5        LOOP  2DROP ; COMMAND
 6
 7 : 3CSCAN ( step end start )   RRD  M3 SCANINIT
 8    9 1 DO  0 STATOUT C!  2DUP  (3SCAN)  SCANEND
 9        BEGIN  SSTAT  3 + C@  I = END
10        LOOP   2DROP ; COMMAND
11
12
13
14
15

Screen: 1267
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 1268
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1269
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1270
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen: 1271
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1272
 0 ( Reduction - Calibration Scan )
 1
 2 CODE Xcal  ?XDATA CALL xvalue STA NEXT
 3 CODE Ycal  U PUSH I PUSH    ?YDATA CALL  2 U MOV   0 I MOV
 4   max-peak 2+ 0 SUB   max-peak 2 SBB  0< NOT
 5   IF   xvalue LDA   xmax STA   I max-peak 2+ MOV
 6        U max-peak MOV   THEN  I POP   U POP   NEXT
 7
 8 : CSCAN   0 STATOUT C!  0 RSTROBE C!  0 xlast !
 9    9 1 DO     CR  0 xmax ! 0 0 max-peak 2!
10        BEGIN   Xcal Ycal   max-peak 2@ 2DROP
11        SSTAT 3 + C@   I = END
12        xmax I 1- 2* 2* 2* databuffer + 8 MOVE
13        LOOP   1 STATOUT C! ; COMMAND
14
15

Screen: 1273
 0 ( Detection - Calibration Scanning )
 1 CODE (CSCAN)   8 # 1 MOV ( index )
 2    BEGIN     YSAMPLED STA B   1 PUSH
 3    BEGIN
 4       ?XSTABLE              (  wait for ok from Ion Path )
 5       ACQUIRE                  ( leaves data point in regs 0,2 )
 6       YSAMPLED STA B           ( send ok to Ion Path )
 7       >PEAK                    ( send data to Reduction )
 8       SSTAT 1+ LDA B   0 SHR   CS   ( test for end of scan )
 9    END  0 0 SUB  SSTAT 3 + LDA B  0 INC  STATOUT STA B  1 POP
10    LOOP  NEXT
11
12 : CSCAN SCANINIT (CSCAN)  ; COMMAND
13
14
15
```

Screen: 1274
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1275
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1276
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen: 1277
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1278
 0 ( Master - Quad Interpolation Table )
 1
 2 4 16 MATRIX ITABLE
 3                      ( column   0=q1mass   1=q1dac )
 4                      (          2=q3mass   3=q3dac )
 5
 6 0 0 ITABLE 128 ERASE
 7
 8 : .ITABLE ( quad#)   CR  7946 LOAD  1 = IF  1  ELSE  3  THEN
 9    DUP .  2 SPACES  0 U.1 6 SPACES 0 7 U.R 1- CR 16 1
10    DO  DUP I ITABLE @ ?DUP
11       IF  9 SPACES  U.1  6 SPACES  DUP 1+ I ITABLE @ 7 U.R
12       ELSE  LEAVE  THEN  CR LOOP  DROP ;
13
14
15

Screen: 1279
 0 ( Master - Get and Save Interpolation Tables )
 1
 2 2LABEL ITABLE 2ITABLE     1LABEL ITABLE 1ITABLE
 3
 4 : IGET   8070 BLOCK  0 0 ITABLE  128 <CMOVE
 5          0 0 ITABLE  1ITABLE 2+  0 1 >SLAVE  128 IPMOVE
 6          0 0 ITABLE  2ITABLE 2+  0 2 >SLAVE  128 IPMOVE ;
 7
 8 : ISAVE   0 0 ITABLE  8070 BLOCK  128 <CMOVE  UPDATE FLUSH ;
 9
10 IGET
11
12
13
14
15
```

```
Screen: 1280
 0 ( Master - Calibration Table Get and Save )
 1
 2 16 ARRAY CAL
 3
 4 : CSAVE ( n)   DUP  0 32 WITHIN
 5    IF  0 CAL  SWAP 32 * 8072 BLOCK +  32 <CMOVE   UPDATE FLUSH
 6    ELSE  DROP  22 .ERROR  THEN ;
 7
 8 : CGET ( n)   DUP  0 32 WITHIN
 9    IF  32 * 8072 BLOCK +  0 CAL  32 <CMOVE
10    ELSE  DROP  22 .ERROR  THEN ;
11
12
13
14
15

Screen: 1281
 0 ( Master - Calibration Mass Display and Set-up )
 1
 2 : .CTITLE   CR  ." Calibration Masses" ;
 3
 4 : .CALMASSES   .CTITLE  CR  16 1
 5    DO  CR  I CAL @  ?DUP
 6       IF  5 SPACES U.1  ELSE  LEAVE  THEN  LOOP ;
 7
 8 : .CAL   CR .CALMASSES ;
 9
10 : CALSET   CR  .CTITLE  ."   ? "  CR  0 CAL 32 ERASE  CR  16 1
11    DO  CR  7 SPACES  63 EMIT ( ? )  8 EMIT ( BS )  #INPUT DUP
12       IF  ELSE  LEAVE  THEN  I CAL !
13    LOOP  PAGE  10 MS  ."        NEW"  .CALMASSES ;
14
15

Screen: 1282
 0 ( Master - Linear Calibration )
 1
 2 : LINCAL   0 0 ITABLE 128 ERASE
 3    10000 0 1 ITABLE !   -1 1 1 ITABLE !
 4    10000 2 1 ITABLE !   -1 3 1 ITABLE !
 5    0 0 ITABLE  1ITABLE 2+  0 1 >SLAVE  128 IPMOVE
 6    0 0 ITABLE  2ITABLE 2+  0 2 >SLAVE  128 IPMOVE ;
 7
 8
 9
10
11
12
13
14
15
```

```
Screen: 1283
 0 ( Master - Calibration Utilities )
 1
 2 :  callims ( mass) ( pushes step end start to slave 1)
 3    2 1PUSH ( step)   DUP  100 / 20 +  2DUP
 4    + 1PUSH ( end)  - 1PUSH ( start) ;
 5
 6 : ?REPEAT ( quad# - n f)   .ITABLE  CR
 7    ." Save (S), Repeat (R), or Quit (Q) "  KEY DUP  82 = NOT ;
 8
 9 : ?SAVE ( n)   83 = IF ISAVE IGET THEN ;
10
11
12
13
14
15

Screen: 1284
 0 ( Master - Calibration Scans )
 1
 2 : 1CALSCANS ( start end )   SL2 CSCAN  SL3 CSCAN
 3    SL1 1CSCAN
 4    BEGIN  SSTAT 2+ C@  END
 5    2 0 >SLAVE  2databuffer DATABUFFER 64 E<IMOVE ;
 6
 7 : 3CALSCANS ( start end )   SL3 CSCAN  SL2 CSCAN
 8    SL1 3CSCAN
 9    BEGIN  SSTAT 2+ C@  END
10    2 0 >SLAVE  2databuffer DATABUFFER 64 E<IMOVE ;
11
12
13
14
15

Screen: 1285
 0 ( Master - More Calibration Utilities )
 1
 2 VARIABLE #CPEAK
 3
 4 : ?ACCEPT ( - n f)   ." Accept (A), Repeat (R), or Specify (S) ?
 5  "  KEY DUP  CR  82 = NOT ;
 6
 7 : ?1CDONE ( n - f)  83 = IF  DROP  ." Value to use ? "  #INPUT
 8    THEN 1 #CPEAK @ 1- ITABLE !  #CPEAK @ 1- DUP CAL @ SWAP
 9    0 SWAP ITABLE !  #CPEAK @ CAL @ NOT ;
10
11 : ?3CDONE ( n - f)   83 = IF  DROP  ." Value to use ? "  #INPUT
12    THEN  3 #CPEAK @ 1- ITABLE !  #CPEAK @ 1- DUP CAL @ SWAP
13    2 SWAP ITABLE !  #CPEAK @ CAL @ NOT ;
14
15
```

```
Screen: 1286
 0 ( Master - Calibration Display )
 1
 2 VARIABLE XSUM  2 ALLOT
 3
 4 : 2+!   >R   I 2@  D+  R>  2! ;
 5
 6 : CALDISP ( mass - avg. dac)   PAGE CR  0. XSUM 2!
 7    ." Calibration Data for Mass " N.1  CR   8 0
 8    DO  5 SPACES  ." DAC value " I @X DUP 0  XSUM 2+!  6 U.R
 9        5 SPACES  ." Intensity " I @Y  8 D.R  CR
10    LOOP  XSUM 2@  8 M/  DUP  ." Average DAC value " 8 U.R
11    CR CR ;
12
13
14
15

Screen: 1287
 0 ( Master - Quad Calibration Loops )
 1
 2 : 1CALLOOP   1 #CPEAK !
 3    BEGIN  0  BEGIN  DROP   #CPEAK @  CAL @  DUP  callims
 4                     1CALSCANS   CALDISP   ?ACCEPT
 5              END  1 #CPEAK +!  ?1CDONE  END ;
 6
 7 : 3CALLOOP   1 #CPEAK !
 8    BEGIN  0  BEGIN  DROP   #CPEAK @  CAL @ DUP  callims
 9              1CALSCANS   CALDISP   ?ACCEPT
10              END  1 #CPEAK +!  ?3CDONE  END ;
11
12
13
14
15

Screen: 1288
 0 ( Master - Extrapolation Functions )
 1 ( y[65535] = [65535-xa]*{[yb-ya]/[xb-xa]} + ya )
 2  65535.0 SCONSTANT INVERSION
 3 : 1EXTRAPOLATE   #CPEAK @  0 OVER  1- ITABLE @ >N    ( yb)
 4    0 OVER 2- ITABLE @ DUP >N  ( ya)  F-  ( yb-ya)
 5    OVER 1- 1 SWAP ITABLE @ >N  ( xb)
 6    OVER 2- 1 SWAP ITABLE @  DUP >N ( xa)  F-  ( xb-xa)
 7    F/  ( yb-ya/xb-xa)  INVERSION  >N  F-  ( 65535-xa)   F*
 8    >N  ( ya)  F+  N>  0  ROT ITABLE ! -1 1 #CPEAK @ ITABLE ! ;
 9
10 : 3EXTRAPOLATE   #CPEAK @  2 OVER  1- ITABLE @ >N
11    2 OVER 2- ITABLE @  DUP >N   F-
12    OVER 1- 3 SWAP  ITABLE @  >N   ( xb)
13    OVER 2- 3 SWAP  ITABLE @  DUP >N  F-
14    F/   INVERSION >N F-  F*  >N  F+  N>  2 ROT ITABLE !
15   -1 3 #CPEAK @ ITABLE ! ;
```

```
Screen: 1289
 0 ( Master - Mass Calibration )
 1
 2 : 1CALIBRATE   0 BEGIN  DROP  1CALLOOP  1EXTRAPOLATE
 3    1 ?REPEAT  END  ?SAVE ;
 4
 5 : 3CALIBRATE   0 BEGIN  DROP  3CALLOOP  3EXTRAPOLATE
 6    3 ?REPEAT  END  ?SAVE ;
 7
 8 : CALIBRATE ( n)   3 = IF  3CALIBRATE  ELSE  1CALIBRATE  THEN ;
 9
10
11
12
13
14
15

Screen: 1290
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1291
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
```

Screen: 1292
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1293
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1294
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1295
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1296
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1297
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1298
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1299
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1300
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1301
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1302
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1303
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1304
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1305
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1306
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1307
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1308
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1309
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1310
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1311
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1312
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1313
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1314
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1315
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1316
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1317
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1318
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

```
Screen: 1319
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1320
 0 ( Master - Tests for graphics data transfers )
 1
 2 HEX
 3
 4 0 2 >SLAVE
 5
 6 : XX DATABUFFER 8000 2000 SMOVE ;
 7 : ZZ DATABUFFER 8000 2000 IPMOVE ;
 8
 9 DECIMAL
10
11 : XXS 0 DO CR I . XX LOOP ;
12 : ZZS 0 DO CR I . ZZ LOOP ;
13
14
15

Screen: 1321
 0 ( FLOATING POINT TEST )
 1
 2 1000 CONSTANT 1000.0
 3
 4 HEX
 5
 6 CODE NORMALIZE ( d - n*1000)
 7    S W MOV   I32 W ) FLD   0 POP   0 POP
 8    I32 MAX-INTEN FLD   1 N) FDIV   I16 ' 1000.0 FLD
 9    1 N) FMUL   0 PUSH   S W MOV   I16 W ) FSTP   FWAIT
10 NEXT
11
12 DECIMAL
13
14
15
```

Screen: 1322
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1323
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1324
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1325
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1326
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1327
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1328
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1329
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1330
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1331
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1332
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1333
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1334
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1335
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1336
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1337
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1338
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1339
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1340
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1341
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1342
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1343
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1344
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1345
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1346
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1347
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1348
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1349
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1350
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1351
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1352
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1353
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1354
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1355
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1356
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1357
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1358
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1359
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1360
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1361
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1362
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1363
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1364
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1365
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1366
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1367
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1368
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1369
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1370
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1371
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1372
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1373
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1374
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1375
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1376
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1377
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1378
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1379
 0
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15

Screen: 1380
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1381
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

```
Screen: 1382
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1383
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1384
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

Screen: 1385
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1386
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1387
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

```
Screen: 1388
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1389
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1390
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1391
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1392
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1393
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

Screen: 1394
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1395
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1396
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

```
Screen: 1397
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1398
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1399
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1400
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1401
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1402
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1403
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1404
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1405
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1406
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1407
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1408
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1409
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1410
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1411
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1412
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1413
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1414
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1415
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1416
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1417
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1418
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1419
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1420
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

Screen: 1421
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1422
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1423
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

```
Screen: 1424
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1425
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1426
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1427
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1428
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1429
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1430
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1431
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1432
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

Screen: 1433
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1434
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1435
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

```
Screen: 1436
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1437
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1438
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1439
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1440
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1441
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1442
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1443
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1444
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1445
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1446
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1447
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1448
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1449
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1450
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

Screen: 1451
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1452
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1453
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

```
Screen: 1454
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1455
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1456
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1457
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1458
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1459
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

Screen: 1460
0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1461
0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1462
0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1463
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1464
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1465
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1466
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1467
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1468
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

```
Screen: 1469
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1470
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1471
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1472
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1473
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1474
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1475
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1476
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1477
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1478
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1479
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1480
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1481
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1482
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1483
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1484
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1485
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1486
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1487
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1488
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1489
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

```
Screen: 1490
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1491
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1492
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

Screen: 1493
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1494
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1495
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1496
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1497
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Screen: 1498
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

```
Screen: 1499
 0 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 1 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 2 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 3 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 4 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 5 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 6 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 7 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 8 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
 9 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
10 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
11 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
12 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
13 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
14 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
15 eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
```